ഹയർ സെക്കന്ററി കോഴ്സ്

XII

കമ്പ്യൂട്ടർ സയൻസ്

ഭാഗം - II

കേരളസർക്കാർ

പൊതുവിദ്യാഭ്യാസ വകുപ്പ്

സംസ്ഥാന വിദ്യാഭ്യാസ ഗവേഷണ പരിശീലന സമിതി (SCERT); കേരളം

2019

ദേശീയഗാനം

ജനഗണമന അധിനായക ജയഹേ
ഭാരത ഭാഗ്യവിധാതാ,
പഞ്ചാബസിന്ധു ഗുജറാത്ത മറാഠാ
ദ്രാവിഡ ഉത്കല ബംഗാ,
വിന്ധ്യഹിമാചല യമുനാഗംഗാ,
ഉച്ഛല ജലധിതരംഗാ,
തവശുഭനാമേ ജാഗേ,
തവശുഭ ആശിഷ മാഗേ,
ഗാഹേ തവ ജയ ഗാഥാ
ജനഗണമംഗലദായക ജയഹേ
ഭാരത ഭാഗ്യവിധാതാ
ജയഹേ, ജയഹേ, ജയഹേ,
ജയ ജയ ജയ ജയഹേ!

പ്രതിജ്ഞ

ഇന്ത്യ എന്റെ രാജ്യമാണ്. എല്ലാ ഇന്ത്യക്കാരും എന്റെ സഹോദരീ സഹോദരന്മാരാണ്.

ഞാൻ എന്റെ രാജ്യത്തെ സ്നേഹിക്കുന്നു; സമ്പൂർണവും വൈവിധ്യപൂർണവുമായ അതിന്റെ പാരമ്പര്യത്തിൽ ഞാൻ അഭിമാനം കൊള്ളുന്നു.

ഞാൻ എന്റെ മാതാപിതാക്കളെയും ഗുരുക്കന്മാരെയും മുതിർന്നവരെയും ബഹുമാനിക്കും.

ഞാൻ എന്റെ രാജ്യത്തിന്റെയും എന്റെ നാട്ടുകാരുടെയും ക്ഷേമത്തിനും ഐശ്വര്യത്തിനും വേണ്ടി പ്രയത്നിക്കും.

Prepared by:
State Council of Educational Research and Training (SCERT)
Poojappura, Thiruvananthapuram 695012, Kerala
Website : www.scertkerala.gov.in *e-mail* : scertkerala@gmail.com
Phone : 0471 - 2341883, Fax : 0471 - 2341869
Typesetting and Layout : SCERT
© Department of Education, Government of Kerala

ആമുഖം

ഏതു വിജ്ഞാനവും മാതൃഭാഷയിൽ പഠിക്കാനും പ്രകാശനം ചെയ്യാനും സാധിക്കും. അതിനുള്ള അവസരം പഠിതാക്കൾക്ക് ഒരുക്കേണ്ടത്, ഏതൊരു പഠന സമ്പ്രദായത്തിന്റെയും അനിവാര്യതയാണ്. അതിന്റെ തുടക്കമെന്ന നിലയ്ക്കാണ് ഹയർസെക്കന്ററി തലത്തിൽ ഭാഷേതര വിഷയങ്ങളിലെ പാഠപുസ്തകങ്ങൾ മലയാളത്തിൽ പ്രസിദ്ധീകരിക്കുന്നത്.

മാതൃഭാഷയിലൂടെയുള്ള വിദ്യാഭ്യാസം, ജ്ഞാനസമ്പാദനത്തിനുള്ള സുഗമമാർഗം എന്നതിനോടൊപ്പം സാംസ്കാരികതനിമയുടെ തിരിച്ചറിയൽ കൂടിയാണ്. അതുകൊണ്ടാണ് വികസിതരാജ്യങ്ങൾ മാതൃഭാഷയെ മുഖ്യബോധന മാധ്യമമായി സ്വീകരിച്ചിരിക്കുന്നത്. ഇന്ത്യയിലാകട്ടെ, ദേശീയതലത്തിലുള്ള പ്രധാന പരീക്ഷകളെല്ലാം പ്രാദേശിക ഭാഷകളിൽക്കൂടി നടത്തുന്നതിനുള്ള സംവിധാനവും ഉണ്ടായി വരികയാണ്. ഈയൊരു സാഹചര്യത്തിൽ നമ്മുടെ കുട്ടികളും മാതൃഭാഷയുടെ ശക്തിസൗന്ദര്യങ്ങൾ തിരിച്ചറിഞ്ഞ് വിവിധ വിഷയങ്ങളിൽ ജ്ഞാനനിർമിതിയിൽ ഏർപ്പെടേണ്ടതുണ്ട്. അതിന് അവരെ സജ്ജരാക്കുകയാണ് ഈ പാഠപുസ്തകങ്ങളുടെ മുഖ്യ ലക്ഷ്യം.

പരിഭാഷപ്പെടുത്തിയ പുസ്തകങ്ങളിൽ അതത് വിഷയങ്ങളിലെ സാങ്കേതിക പദങ്ങൾ പരമാവധി മലയാളത്തിലാക്കിയിട്ടുണ്ട്. നമ്മുടെ ഭാഷയിൽ ചിരപരിചിതമായ ഇംഗ്ലീഷ് പദങ്ങളെ അതേപടി സ്വീകരിച്ചിട്ടുമുണ്ട്. വിവർത്തനത്തിന് തീർത്തും വഴങ്ങാത്ത പദങ്ങളെ അതേരീതിയിൽ തന്നെ ഉപയോഗിച്ചിരിക്കുന്നു. മാതൃഭാഷയിൽ പഠിക്കുന്നവർക്ക് ആശയഗ്രഹണം സുഗമമാക്കുന്ന വിധത്തിലാണ് പാഠപുസ്തകരചന നടത്തിയിരിക്കുന്നത്. അതോടൊപ്പം മലയാളഭാഷയുടെ വളർച്ചയ്ക്കും ഈ പ്രവർത്തനം സഹായകമാകുമെന്ന് കരുതുന്നു.

പാഠപുസ്തകവിവർത്തന രംഗത്ത് നമ്മുടെ രാജ്യത്ത് നടന്ന വലിയൊരു കാൽവെപ്പാണ് ഇത്. പ്രഥമ സംരംഭമെന്നനിലയിൽ പല പരിമിതികളും പരിഭാഷയിൽ വന്നിട്ടുണ്ടാകാം. ക്ലാസ്മുറിയിൽ പ്രയോഗത്തിൽ വരുമ്പോഴാണ് അവയെല്ലാം കൂടുതൽ ബോധ്യപ്പെടുക. തുടർന്ന് വരുന്ന ഘട്ടങ്ങളിൽ അവയൊക്കെ പരിഹരിക്കുന്നതിന് എല്ലാ അഭ്യുദയകാംക്ഷികളിൽ നിന്നും വിശിഷ്യ അധ്യാപകർ, വിദ്യാർത്ഥികൾ എന്നിവരിൽ നിന്നും അഭിപ്രായങ്ങളും നിർദേശങ്ങളും പ്രതീക്ഷിക്കുന്നു.

ഡോ.ജെ. പ്രസാദ്

ഡയറക്ടർ,

എസ്.സി.ഇ.ആർ.ടി. കേരളം

പാഠപുസ്തക നിർമാണ സമിതി

ശ്രീ. ജോയ് ജോൺ
എച്ച്.എസ്.എസ്.ടി.,
സെന്റ് ജോസഫ് എച്ച്.എസ്.എസ്. തിരുവനന്തപുരം

ശ്രീ. വിനോദ് വി.
എച്ച്.എസ്.എസ്.ടി., എൻ.എസ്.എസ്
എച്ച്.എസ്.എസ്.,പ്രാക്കുളം, കൊല്ലം.

ശ്രീ. എ. എസ്. ഇസ്മൈൽ
എച്ച്.എസ്.എസ്.ടി., പി.ജെ.എം.എസ്
ജി.എച്ച്.എസ്.എസ്., കണ്ടശ്ശാൻകടവ്, തൃശ്ശൂർ

ശ്രീ. പ്രശാന്ത് പി. എം
എച്ച്.എസ്.എസ്.ടി.,
സെന്റ് ജോസഫ് ബോയ്സ് എച്ച്.എസ്.എസ്.,
കോഴിക്കോട്

ശ്രീ. മുസ്ഫ ഷംസുൽ ഹക്ക് കെ. കെ.
എച്ച്.എസ്സ്.എസ്സ്.ടി.,
ജി.എച്ച്.എസ്.എസ്. കൂത്തുപറമ്പ്, കണ്ണൂർ

ശ്രീ. രാജ്മോഹൻ സി.
എച്ച്.എസ്സ്.എസ്സ്.ടി.,
നവമുകുന്ദ എച്ച്.എസ്.എസ്.,
തിരുനാവായ, മലപ്പുറം

ശ്രീ. നെജീവ് പി. പി.
എച്ച്.എസ്.എസ്.ടി.,
ഹിമായത്തുൽ ഇസ്ലാം എച്ച്.എസ്.എസ്.,
കോഴിക്കോട്

ശ്രീ. അബൂബക്കർ പി.
എച്ച്.എസ്.എസ്.ടി., ഗവ. ജി.എസ്.എസ്.ടി.,
ചാലപ്പുറം, കോഴിക്കോട്

വിദഗ്ധർ

ഡോ. ലജീഷ് വി. എൽ
അസ്സി. പ്രൊഫ. ഡിപ്പാർട്ട്മെന്റ് ഓഫ്
കമ്പ്യൂട്ടർ സയൻസ് , കോഴിക്കോട് സർവകലാശാല

മധു വി. ടി.
ഡയറക്ടർ, കമ്പ്യൂട്ടർ സെന്റർ, സർവകലാശാല
കോഴിക്കോട്.

ഡോ. സുഷീൽ കുമാർ
അസ്സി. പ്രൊഫ. ഡിപ്പാർട്ട്മെന്റ് ഓഫ് ഇംഗ്ലീഷ്,
ഡി.ബി.കോളജ്, ശാസ്താംകോട്ട

ഡോ. വിനയകുമാരൻ നായർ എൻ.
അസ്സി. കമാൻഡന്റ്, ഹൈ - ടെക് സെൽ,
പോലീസ് ഹെഡ് ക്വാർട്ടേർസ്, തിരുവനന്തപുരം

ഡോ. മധു എസ്സ്. നായർ
അസ്സി. പ്രൊഫ. ഡിപ്പാർട്ട്മെന്റ് ഓഫ്
കമ്പ്യൂട്ടർ സയൻസ് , കേരള സർവകലാശാല

ഡോ. ബിനു. പി. ചാക്കോ
അസ്സി. പ്രൊഫ. ഡിപ്പാർട്ട്മെന്റ് ഓഫ്
കമ്പ്യൂട്ടർ സയൻസ് , പ്രജ്യോതി നികേതൻ കോളജ്,
പുതുക്കാട്, തൃശ്ശൂർ

ഡോ. ദീപ എൽ. സി.
അസ്സി. പ്രൊഫ. ഡിപ്പാർട്ട്മെന്റ് ഓഫ് ഇംഗ്ലീഷ്,
ഗവ.വിമൻസ് കോളജ്, തിരുവനന്തപുരം

ഡോ. കബീർ. വി
അസ്സി. പ്രൊഫ. & ഹെഡ്, ഡിപ്പാർട്ട്മെന്റ് ഓഫ്
കമ്പ്യൂട്ടർ സയൻസ്, ഫറൂക്ക് കോളേജ്, കോഴിക്കോട്

ആർട്ടിസ്റ്റ്

സുധീർ വൈ   വിനീത് വി

അക്കാദമിക് കോർഡിനേറ്റർ

ഡോ. മീന എസ്.
അസ്സി. പ്രൊഫ., എസ്.സി.ഇ.ആർ.ടി

പാഠപുസ്തക പരിഭാഷ സമിതി
(മലയാളം)

ഡോ. ബിനു പി ചാക്കോ
അസോസിയേറ്റ് പ്രൊഫസർ, ഡിപ്പാർട്ട്മെന്റ് ഓഫ് കമ്പ്യൂട്ടർ സയൻസ്
പ്രജ്യോതി നികേതൻ കോളേജ്, പുതുക്കാട്

ഡോ. ഗ്ലാഡ്സ്റ്റൺ എസ്. രാജ്
അസിസ്റ്റന്റ് പ്രൊഫസർ & ഹെഡ് ഡിപ്പാർട്ട്മെന്റ് ഓഫ് കമ്പ്യൂട്ടർ സയൻസ്
ഗവ. കോളേജ്, നെടുമങ്ങാട്

ശ്രീ. എ. എസ്. ഇസ്മൈൽ
എച്ച്.എസ്.എസ്.ടി., ജി.എച്ച്.എസ്സ്.എസ്സ്,
മുല്ലശ്ശേരി, തൃശ്ശൂർ

ശ്രീ. ഹരി കെ
എച്ച്.എസ്.എസ്.ടി, ഗവ. വി.& എച്ച്.എസ്.എസ്.
കടപ്പുറം, ചാവക്കാട്, തൃശൂർ

ശ്രീ. ശ്രീജിത്ത് പി
എച്ച്.എസ്.എസ്.ടി., സി.ജെ. എച്ച്.എസ്.എസ്.,
ചെമ്മനാട്, കാസർകോട്

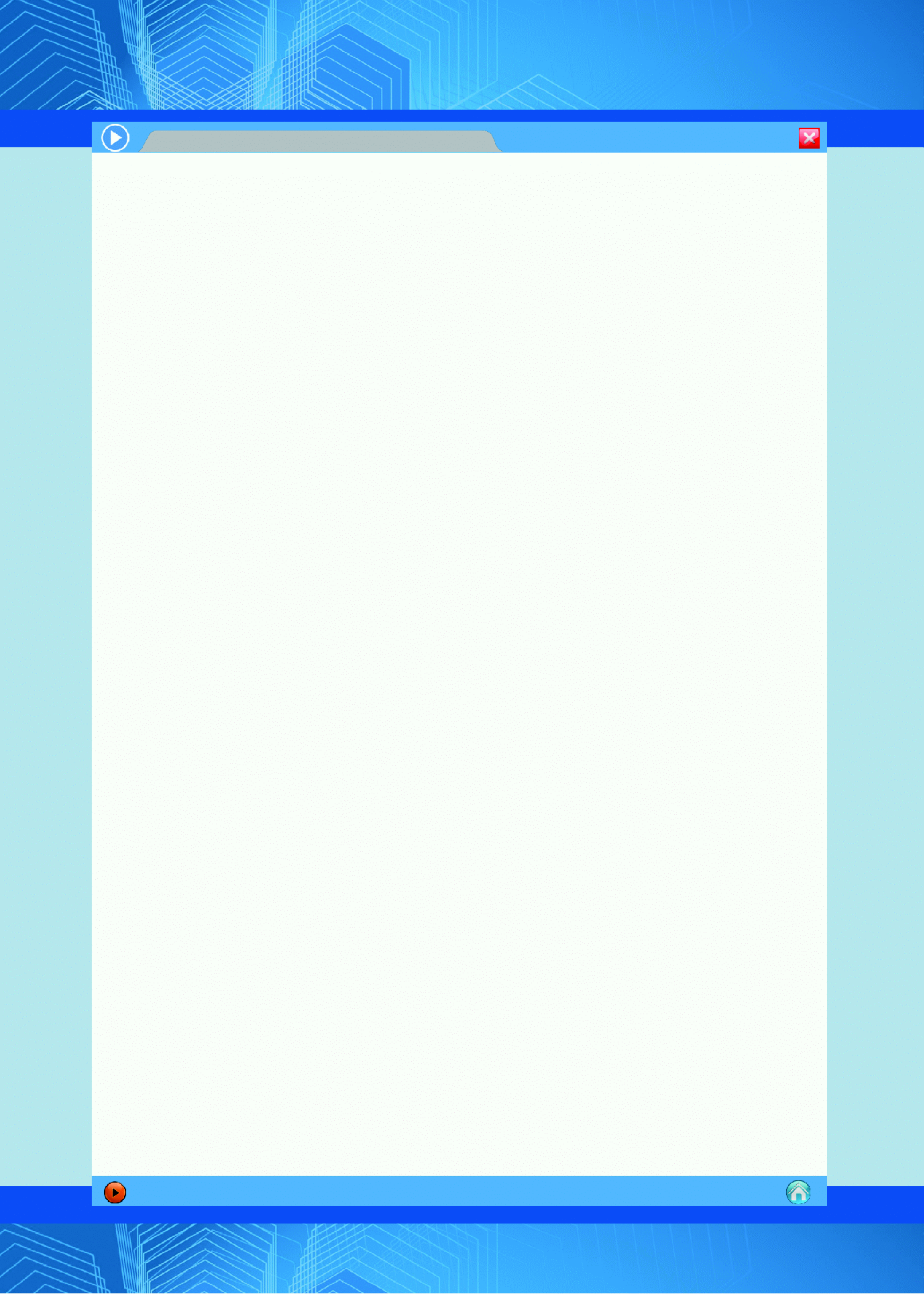
ശ്രീ. സായ് പ്രകാശ്
എച്ച്.എസ്.എസ്.ടി.,
സെന്റ്മേരീസ് എച്ച്.എസ്.എസ്. വെട്ടുകാട്

ശ്രീ. സുധീർ പി. എസ്.
എച്ച്.എസ്.എസ്.ടി., കാട്ടുകുളം,
ഹയർ സെക്കണ്ടറി

ശ്രീ. ഷിജു ആർ.
എച്ച്.എസ്.എസ്.ടി.,
ജി.എച്ച്.എസ്.എസ്. വടകര, പുത്തൂർ

അക്കാദമിക് കോർഡിനേറ്റർ

ശ്രീമതി റിയാന അൻസാരി
റിസർച്ച് ഓഫീസർ, എസ്.സി.ഇ.ആർ.ടി



ഉള്ളടക്കം

8.	ഡാറ്റാബേസ് നിർവഹണ സംവിധാനം	281
9.	സ്ട്രക്ചേർഡ്ക്വറി ലാംഗ്വേജ് (SQL)	309
10.	PHP ഉപയോഗിച്ചുള്ള സെർവർ സൈഡ് സ്ക്രിപ്റ്റിംഗ്	360
11.	കമ്പ്യൂട്ടിങ്ങിലെ മുന്നേറ്റങ്ങൾ	408
12.	വിവര ആശയവിനിമയ സാങ്കേതികവിദ്യയും സമൂഹവും	437

Z6A3F8

പാഠപുസ്തകത്തിൽ ഉപയോഗിച്ചിരിക്കുന്ന സൂചനകൾ

നമുക്ക് ചെയ്യാം

നിങ്ങളുടെ പുരോഗതി അറിയുക

ഇൻഫർമേഷൻ ബോക്സ്

നമുക്ക് പരിശീലിക്കാം

നമുക്ക് സംഗ്രഹിക്കാം

W8P6J5

8 ഡാറ്റാബേസ് നിർവഹണ സംവിധാനം

പ്രധാന പഠനനേട്ടങ്ങൾ

ഈ അധ്യായത്തിന്റെ പഠനം പൂർത്തിയാകുന്നതോടെ പഠിതാവ് ആർജിക്കേണ്ട പഠനനേട്ടങ്ങൾ

- ഫയലുകളുടെ ആവശ്യം മനസ്സിലാക്കുന്നു.
- പരമ്പരാഗത ഫയൽ നിർവഹണ സംവിധാനത്തിന്റെ പ്രധാന പരിമിതികൾ തിരിച്ചറിയുന്നു.
- ഡാറ്റാബേസ് നിർവഹണ സംവിധാനത്തിന്റെ (DBMS) വിവിധ ഗുണങ്ങൾ വിശദീകരിക്കാനറിയുന്നു.
- DBMS ന്റെ വിവിധ ഘടകങ്ങളും അവയുടെ ഉദ്ദേശ്യവും വിശദീകരിക്കുന്നു.
- DBMS ലെ വിവിധതരം ഉപയോക്താക്കളെയും അവരുടെ കടമകളും തിരിച്ചറിയുന്നു.
- DBMS ലെ ഡാറ്റ സംഗ്രഹത്തിന്റെയും (Abstraction) ഡാറ്റ സ്വാതന്ത്ര്യത്തിന്റെയും (Independence) വിവിധ തലങ്ങൾ വിശദീകരിക്കുന്നു.
- ഉദാഹരണങ്ങൾ ഉദ്ധരിച്ചുകൊണ്ട് റിലേഷണൽ ഡാറ്റ മോഡൽ വിശദീകരിക്കുന്നു.
- RDBMS ലെ വ്യത്യസ്ത പദങ്ങൾ അനുയോജ്യമായി ഉപയോഗിക്കുന്നു.
- റിലേഷണൽ ബീജഗണിതത്തിലെ വിവിധ പ്രവർത്തനങ്ങൾ പ്രയോഗിക്കുകയും വിലയിരുത്തുകയും ചെയ്യുന്നു.

ഇത് അറിവിന്റെയും വിവര സാങ്കേതിക വിദ്യയുടെയും കാലമാണ്. മത്സരയോട്ടങ്ങളുടെ ഈ ലോകത്തിൽ സ്ഥാപനങ്ങളുടെ നിലനിൽപ്പ് ഉയർന്ന കൃത്യതയോടും വേഗതയോടും ആവശ്യമുള്ള വിവരങ്ങൾ ലഭിക്കുന്നതിനെ ആശ്രയിച്ചിരിക്കും. വിവരങ്ങൾ ലഭ്യമാകുന്നത് ഡാറ്റ സംസ്കരണത്തിലൂടെയാണെന്ന് നമുക്ക് അറിയാം. വിവരങ്ങൾ രൂപപ്പെടുത്തുന്നതിനായി വളരെ അധികം ഡാറ്റ സ്വീകരിക്കുകയും സംഭരിക്കുകയും പ്രക്രിയകൾ നടത്തുകയും ചെയ്യുന്നു.

ബാങ്കുകൾ, ബിസിനസ്സ് സ്ഥാപനങ്ങൾ, സ്കൂളുകൾ മുതലായവയ്ക്ക് വിവരങ്ങൾ ആവശ്യമാണെന്ന് നിങ്ങൾക്ക് അറിയാമല്ലോ. പരമ്പരാഗതമായി ഈ സ്ഥാപനങ്ങൾ വളരെയധികം ഡാറ്റ കൈകാര്യം ചെയ്തത് എങ്ങനെയാണെന്ന് നിങ്ങൾക്ക് ഊഹിക്കാനാകുമോ? മുൻകാലങ്ങളിൽ ബുക്ക് കീപ്പിംഗ് മാതൃകയിൽ, അതായത് വിവരങ്ങൾ പുസ്തകങ്ങളിൽ കരകൃതമായി (manual processing) എഴുതി കൈകാര്യം ചെയ്യുന്ന രീതി അവലംബിച്ചിരുന്നു. ഈ രീതിയിൽ എഴുതിയ പുസ്തകങ്ങൾ സൂക്ഷിക്കുന്നതിന് ധാരാളം സ്ഥലം ആവശ്യമാണെന്നും ഡാറ്റ സംസ്കരണം വളരെ പ്രയാസകരമാണെന്നതും വ്യക്തമാണ്. കമ്പ്യൂട്ടറുകളുടെ ആവിർഭാവത്തോടെ ഡാറ്റ ഫലപ്രദമായി സംഭരിക്കുവാൻ സാധിക്കുന്നു, എന്നാൽ പകർപ്പ്, പൊരുത്തക്കേട്, അസാധാരണത്വം മുതലായവയ്ക്കുള്ള സാധ്യത നിലനിൽക്കുന്നു. ഈ പരിമിതികൾ മറികടക്കാനുള്ള ഫലപ്ര

മായ റെക്കോർഡ് കീപ്പിങ് സംവിധാനമെന്ന നിലയിൽ 'ഡാറ്റാബേസ് മാനേജ്മെന്റ് സിസ്റ്റം' (DBMS) എന്ന ആശയം ഈ അധ്യായത്തിൽ അവതരിപ്പിച്ചിരിക്കുന്നു. ഡാറ്റാബേസിൽ നിന്ന് ആവശ്യമുള്ളതും പ്രസക്തവുമായ വിവരങ്ങൾ വീണ്ടെടുക്കാനുള്ള വിവിധ പ്രവർത്തനങ്ങൾ ചർച്ച ചെയ്യുന്നുമുണ്ട്.

8.1 ഡാറ്റാബേസ് എന്ന ആശയം (Concept of database)

ഹയർ സെക്കൻഡറി വകുപ്പിന്റെ സ്കൂൾ പ്രവേശന ഏകജാലക സംവിധാനം പരിഗണിക്കുക. പതിനൊന്നാം ക്ലാസ്സ് പ്രവേശനത്തിനായി ഓരോ വർഷവും വിദ്യാർഥികൾ, കോഴ്സുകൾ, സ്കൂളുകൾ, ഗ്രേഡുകൾ എന്നീ ഡാറ്റയുടെ ഒരു വലിയ ശേഖരം (ഏകദേശം 5,00,000 അപേക്ഷകൾ അല്ലെങ്കിൽ 19 GB ഡാറ്റ) പരിപാലിക്കുന്നു. ഈ ഡാറ്റ നിരവധി വിദ്യാലയങ്ങളും വിദ്യാർഥികളും ഒരേ സമയം ഉപയോഗിക്കുന്നു. വിദ്യാർഥികളുടെയും സ്കൂളുകളുടെയും അലോട്ട്മെന്റിനെക്കുറിച്ചുള്ള ചോദ്യങ്ങൾക്ക് വേഗത്തിൽ ഉത്തരം നൽകണം. വ്യത്യസ്ത സ്കൂളുകളുടെ ഡാറ്റയിൽ വരുത്തുന്ന മാറ്റങ്ങൾ സ്ഥിരമായി ഉപയോഗിക്കണം. കൂടാതെ ഡാറ്റയുടെ ചില ഭാഗങ്ങൾ (ഉദാ. ഗ്രേഡുകൾ അല്ലെങ്കിൽ WGPA) ഉപയോഗിക്കുന്നത് നിയന്ത്രിക്കേണ്ടതുമുണ്ട്. പരമ്പരാഗത ഫയൽ നിർവഹണ സംവിധാനത്തിൽ ഡാറ്റ സൂക്ഷിച്ചുകൊണ്ട് നമുക്ക് അതിനെ നിയന്ത്രിക്കാനാവും. എന്നാൽ ഈ രീതിക്ക് പല ന്യൂനതകളുണ്ട്.

- വ്യത്യസ്ത ആപ്ലിക്കേഷനുകൾക്കായി ഒരേ ഡാറ്റയുടെ കൂടുതൽ പകർപ്പുകൾ സൂക്ഷിക്കേണ്ടതുണ്ട്. ഇത്തരം സംഭരണം ഡാറ്റ ആവർത്തിക്കപ്പെടുന്നതിന് ഇടയാക്കുന്നു.
- വ്യത്യസ്ത ഉപയോക്താക്കൾ ഒരേ സമയം ഉണ്ടാക്കുന്ന മാറ്റങ്ങളിൽ നിന്ന് ഡാറ്റയെ പരിരക്ഷിക്കുന്നതിനുള്ള ഒരു സംവിധാനവും ഇല്ല.
- മാറ്റങ്ങൾ വരുത്തുമ്പോൾ സിസ്റ്റം തകരാറിലായാൽ സ്ഥിരമായ ഒരു അവസ്ഥയിലേക്ക് ഡാറ്റ പുനസ്ഥാപിക്കുമെന്ന് ഉറപ്പുവരുത്താൻ ഒരു വഴിയും ഇല്ല.
- സുരക്ഷയ്ക്കായി ഓപ്പറേറ്റിംഗ് സംവിധാനങ്ങൾ ഒരു രഹസ്യകോഡ് സംവിധാനം മാത്രമാണ് നൽകുന്നത്. ഡാറ്റയിൽ സുരക്ഷാ നയങ്ങൾ നടപ്പിലാക്കാൻ ഇത് പര്യാപ്തമല്ല.
- ഡാറ്റയിൽ നിലവാരമൊന്നും ഇല്ല.

8.1.1 ഡാറ്റാബേസിന്റെ ആവശ്യകത (Need of database)

മുകളിൽ പറഞ്ഞിരിക്കുന്ന പിഴവുകൾ ഡാറ്റാബേസ് ഉപയോഗിച്ച് തരണം ചെയ്യാവുന്നതാണ്. വലിയ അളവിൽ ഡാറ്റ കൈകാര്യം ചെയ്യാൻ പരമ്പരാഗത ഫയൽ നിർവഹണ സംവിധാനം മതിയാകില്ല. ഇത്തരം സന്ദർഭങ്ങളിൽ ഡാറ്റാബേസ് ഉപയോഗിക്കുന്നു. ഒന്നിലധികം ആപ്ലിക്കേഷനുകൾക്ക് ഉപയോഗിക്കാൻ കഴിയുന്നവിധത്തിൽ ആവർത്തനം അധികമില്ലാതെ സംഭരിച്ച പരസ്പരബന്ധിതമായ വിവരങ്ങളുടെ ഒരു സംഘടിത ശേഖരമാണ് ഡാറ്റാബേസ്. ഹയർ സെക്കണ്ടറി സ്കൂൾ പ്രവേശനത്തിന് ഉപയോഗിക്കുനന്ന ഏകജാല സംവിധാനം ഡാറ്റയെ ഒരു ഡാറ്റാബേസ് നിർവഹണ സംവിധാനത്തിൽ സൂക്ഷിച്ചു കൊണ്ട് കൈകാര്യ ചെയ്യുന്നതിന് ഒരു ഉദാഹരണമാണ്. അടിസ്ഥാനപരമായി ഒരു ഡാറ്റാബേസ് നിർവഹണ സംവിധാനം എന്നത് ഡാറ്റാബേസിന്റെ സംഭരണം, വീണ്ടെടുക്കൽ, നിർവഹണം മുതലായവയ്ക്ക് സൗകര്യമൊരുക്കുന്ന ഒരു കൂട്ടം പ്രോഗ്രാമുകളാണ്. ഡാറ്റാബേസ് സംഭരിക്കുന്നതിനും വീണ്ടെടുക്കുന്നതിനും സൗകര്യപ്രദവും കാര്യക്ഷമവുമായ ഒരു അന്തരീക്ഷം പ്രദാനം ചെയ്യുക എന്നതാണ് DBMS ന്റെ പ്രാഥമിക ലക്ഷ്യം.

8.1.2 DBMS ന്റെ നേട്ടങ്ങൾ (Advantages of database)

ഡാറ്റാബേസ് സംവിധാനങ്ങൾ വലിയ അളവിലുള്ള ഡാറ്റ കൈകാര്യം ചെയ്യാൻ രൂപകല്പന ചെയ്തിരിയ്ക്കുന്നു. ഡാറ്റ സംഭരണത്തിനായുള്ള ഡാറ്റാബേസ് ഘടനകളുടെ നിർവചനവും, ഡാറ്റ കൈകാര്യം ചെയ്യുന്നതിനുള്ള സംവിധാനങ്ങളും DBMSൽ ഉൾപ്പെട്ടിരിക്കുന്നു. കൂടാതെ, സംഭരിച്ച ഡാറ്റയുടെ അനധികൃത ഉപയോഗം, സിസ്റ്റം പരാജയം മൂലം ഡാറ്റ നഷ്ടമാവുന്ന അവസ്ഥ തുടങ്ങിയ പ്രശ്നങ്ങളിൽ നിന്നുള്ള സുരക്ഷ DBMS ഉറപ്പുവരുത്തുന്നു. പല ഉപയോക്താക്കൾക്കിടയിൽ ഡാറ്റ പങ്കുവയ്ക്കേണ്ടതുണ്ടെങ്കിൽ അതുമൂലമുള്ള അപാകതയുള്ള ഫലങ്ങൾ സിസ്റ്റം ഒഴിവാക്കണം. പരമ്പരാഗത ഫയൽ നിർവഹണ സംവിധാനത്തെക്കാളും ഡാറ്റാബേസ് നിർവഹണ സംവിധാനത്തിന് അനേകം ഗുണങ്ങളുണ്ട്, അവ താഴെ ചർച്ചചെയ്യാം.

- **ഡാറ്റ റിഡൻഡൻസി നിയന്ത്രിക്കുന്നു:** പരമ്പരാഗത ഫയൽ നിർവഹണ സംവിധാനത്തിൽ, ഡാറ്റ നിരവധി ഫയലുകളിലായി സംഭരിക്കപ്പെടാം. ഒന്നിലധികം സ്ഥാനങ്ങളിൽ ഒരേ ഡാറ്റ സംഭരിക്കുക (ഒരേ ഫയലിൽ അല്ലെങ്കിൽ വ്യത്യസ്ത ഫയലുകളിൽ ആയിരിക്കാം) അല്ലെങ്കിൽ ഡാറ്റയുടെ തനിപ്പകർപ്പിന് ഡാറ്റ റിഡൻഡൻസി എന്ന് പറയുന്നു. റിഡൻഡൻസി മൂലം ഡാറ്റ സംഭരണത്തിനും ഡാറ്റ ഉപയോഗത്തിനും ഉയർന്ന ചെലവ് വരുന്നു. ഡാറ്റാബേസ് സംവിധാനത്തിൽ ഡാറ്റകളുടെ പകർപ്പ് സൂക്ഷിക്കുന്നില്ല, പകരം എല്ലാ ഡാറ്റയും കേന്ദ്രീകൃത രീതിയിൽ ഒരിടത്ത് സൂക്ഷിച്ചിരിക്കുന്നു. ഡാറ്റ ആവശ്യമുള്ള എല്ലാ ആപ്ലിക്കേഷനുകളും ഉപയോക്താക്കളും കേന്ദ്രീകൃതമായി പരിപാലിക്കപ്പെട്ട ഡാറ്റാബേസ് ഉപയോഗിക്കുന്നു. സാങ്കേതികമോ ബിസിനസ് പരമോ ആയ കാരണങ്ങൾകൊണ്ട് ചില സന്ദർഭങ്ങളിൽ ഒരേ ഡാറ്റയുടെ നിരവധി പകർപ്പുകൾ സൂക്ഷിക്കാറുണ്ട്. എന്നിരുന്നാലും, ഏത് സാഹചര്യത്തിലും ഡാറ്റയുടെ ആവർത്തനത്തെ ശ്രദ്ധാപൂർവം നിയന്ത്രിക്കുന്നു.

- **ഡാറ്റയുടെ പൊരുത്തം:** ഡാറ്റ റിഡൻഡൻസി ഡാറ്റയുടെ പൊരുത്തമില്ലായ്മയിലേക്ക് നയിച്ചേക്കാം. അതായത്, ഒരേ ഡാറ്റയുടെ വിവിധ കോപ്പികളിൽ വ്യത്യസ്ത വിലകൾ കാണിക്കാം. നിങ്ങളുടെ ക്ലാസ്സിലെ വിദ്യാർഥികളുടെ വിലാസം നിങ്ങളുടെ ക്ലാസ് അധ്യാപകനും പ്രിൻസിപ്പാളും പ്രത്യേകമായി പരിപാലിക്കുന്നുവെന്ന് കരുതുക. വിലാസത്തിൽ മാറ്റങ്ങൾ വരുത്തുന്നതിന് ചില വിദ്യാർഥികൾ ക്ലാസ് അധ്യാപകനെയും മറ്റു ചിലർ പ്രിൻസിപ്പാളിനെയും സമീപിക്കുന്നു. ഒരു നിശ്ചിത കാലയളവിനു ശേഷം, രണ്ടു പട്ടികകളും അപ്രസക്തവും പൊരുത്തപ്പെടാത്തതുമാകും. ഡാറ്റ റിഡൻഡൻസി നിയന്ത്രക്കുന്നതിലൂടെ ഡാറ്റയുടെ പൊരുത്തം ഉറപ്പാക്കാം. ഡാറ്റ ഒരിടത്ത് മാത്രം സൂക്ഷിക്കുകയാണെങ്കിൽ, അതിന്റെ വിലകളിലേക്കുള്ള ഏത് മാറ്റവും ആ സ്ഥലത്ത് മാത്രമെ ചെയ്യാനാകൂ, ഒപ്പം മാറ്റം വരുത്തിയ വിലകൾ എല്ലാ ഉപയോക്താക്കൾക്കും ഉടൻ ലഭ്യമാകുകയും ചെയ്യും.

- **കാര്യക്ഷമമായ ഡാറ്റ ഉപയോഗം:** ഡാറ്റ കാര്യക്ഷമമായി സംഭരിക്കാനും വീണ്ടെടുക്കാനും നിരവധി സാങ്കേതികവിദ്യകൾ DBMS ഉപയോഗപ്പെടുത്തുന്നു.

- **ഡാറ്റസമഗ്രത:** ഡാറ്റസമഗ്രത ഡാറ്റാബേസിലെ ഡാറ്റയുടെ പൂർണത, കൃത്യത, സ്ഥിരത എന്നിവയെ സൂചിപ്പിക്കുന്നു. ഒരു ഡാറ്റ റെക്കോർഡിന്റെ രണ്ട് മാറ്റങ്ങൾക്കിടയിലുള്ള ഡാറ്റയിൽ എന്തെങ്കിലും വ്യതിയാനമുണ്ടെങ്കിൽ അത് സൂചിപ്പിക്കാൻ ഇതിന് കഴിയും. ഡാറ്റാബേസ് രൂപകല്പന ചെയ്യുന്ന ഘട്ടത്തിൽ ശരിയായ നിയമങ്ങളും നടപടി

ക്രമങ്ങളും ഉപയോഗിച്ച് ഡാറ്റസമഗ്രത നടപ്പിൽ വരുത്തുന്നു. തെറ്റുകളുടെ പരിശോധനയും സാധൂകരണ പരിപാടികളും ഉപയോഗിച്ച് ഡാറ്റസമഗ്രത നിലനിർത്താൻ കഴിയും.

- **ഡാറ്റ സുരക്ഷ:** ഡാറ്റാബേസിലുള്ള വിവരങ്ങൾ ഏതൊരു കമ്പനിക്കും സ്ഥാപനത്തിനും മൂല്യമേറിയതാണ്. അതിനാൽ ഇത് സുരക്ഷിതമായും സ്വകാര്യവുമായും സൂക്ഷിക്കണം. അംഗീകൃതമല്ലാത്ത വ്യക്തികൾ നടത്തുന്ന അവിചാരിതമോ കരുതിക്കൂട്ടിയുള്ളതോ ആയ വെളിപ്പെടുത്തലുകൾ, അനധികൃത പരിഷ്കരണം, നശീകരണം എന്നീ പ്രശ്നങ്ങളിൽ നിന്നുള്ള സംരക്ഷണമാണ് ഡാറ്റ സുരക്ഷ കൊണ്ട് ഉദ്ദേശിക്കുന്നത്. വിവിധ പ്രോഗ്രാമുകളും ഉപയോക്താക്കളും പൊതുവായി ഡാറ്റ പങ്കുവയ്ക്കാം. എന്നാൽ ഉപയോഗിക്കുവാനുള്ള അവകാശങ്ങൾ ക്രമീകരിച്ചുകൊണ്ട് തിരഞ്ഞെടുത്ത ഉപയോക്താക്കൾക്ക് നിർദിഷ്ട വിവരങ്ങൾ മാത്രം ഉപയോഗിക്കുവാൻ കഴിയുന്ന രീതിയിൽ പരിമിതപ്പെടുത്താവുന്നതാണ്. പാസ്സ്‌വേഡിന്റെ സഹായത്താൽ, ഒരു ഡാറ്റാബേസിലെ വിവരങ്ങൾ അംഗീകൃത വ്യക്തികൾക്ക് മാത്രമായി ലഭ്യമാക്കാം.

- **ഡാറ്റ പങ്കുവയ്ക്കൽ:** ഡാറ്റാബേസിൽ സംഭരിച്ചിരിക്കുന്ന ഡാറ്റ നിരവധി ഉപയോക്താക്കൾക്കും പ്രോഗ്രാമുകൾക്കും ഒരേസമയം പങ്കിടാൻ കഴിയും. കൂടാതെ ഓരോരുത്തർക്കും വ്യത്യസ്ത ഉദ്ദേശ്യങ്ങൾക്കായും ഉപയോഗിക്കാം.

- **മാനദണ്ഡങ്ങളുടെ നടപ്പാക്കൽ:** ഡാറ്റാബേസിന്റെ കേന്ദ്ര നിയന്ത്രണം ഉപയോഗിച്ച്, ഒരു ഡാറ്റാബേസ് അഡ്മിനിസ്ട്രേറ്റർ (DBA) ആവശ്യമായ മാനദണ്ഡങ്ങൾ നിർവചിക്കുകയും നടപ്പിലാക്കുകയും ചെയ്യുന്നു. കംപ്യൂട്ടറുകൾ തമ്മിൽ ഡാറ്റ വിനിമയം നടത്തുന്നതിനുള്ള ഡാറ്റ ഫോർമാറ്റുകൾക്ക് മാനദണ്ഡങ്ങൾ നിർവചിക്കാം. നാമകരണ സമ്പ്രദായങ്ങൾ, പ്രദർശന ഫോർമാറ്റുകൾ, റിപ്പോർട്ട് ഘടനകൾ, പദാവലി, ഡോക്യുമെന്റേഷൻ നിലവാരങ്ങൾ, അപ്ഡേറ്റ് നടപടികൾ, വിനിയോഗ നിയമങ്ങൾ തുടങ്ങിയവ പ്രായോഗികമായ മാനദണ്ഡങ്ങളിൽ ഉൾപ്പെടുന്നു. വിവിധ വകുപ്പുകളുടെയും പ്രൊജക്റ്റുകളുടെയും ഉപയോക്താക്കളുടെയും ഇടയിൽ ആശയവിനിമയവും സഹകരണവും ഉണ്ടാക്കുന്നതിന് ഇത് സഹായിക്കുന്നു.

- **തകർച്ചയിൽ നിന്ന് വീണ്ടെടുക്കൽ:** ഒരു സിസ്റ്റം തകരുമ്പോൾ, മുഴുവൻ ഡാറ്റയും അല്ലെങ്കിൽ ഡാറ്റയുടെ ഒരു ഭാഗം ഉപയോഗശൂന്യമാകും. തകർച്ചയിൽ നിന്നും ഡാറ്റ വീണ്ടെടുക്കുന്നതിന് DBMS ചില സംവിധാനങ്ങൾ നൽകുന്നു.

നിങ്ങളുടെ പഠനപുരോഗതി അറിയുക

1. ഒരേ ഡാറ്റ വിവിധ സ്ഥലങ്ങളിൽ സൂക്ഷിക്കുന്നതിനെ ________ എന്നു പറയുന്നു.
2. ഒരു വിദ്യാർഥിയുടെ വിലാസം സ്കൂൾ റെക്കോർഡിൽ രണ്ടു രീതിയിൽ സൂക്ഷിച്ചിരിക്കുന്നു; ഈ സാഹചര്യം ________ എന്നറിയപ്പെടുന്നു.
3. ഡാറ്റയുടെ അംഗീകൃതമല്ലാത്ത ഉപയോഗം ________ ഉപയോഗിച്ചു കൊണ്ട് തടയുന്നു.
4. ഡാറ്റ പങ്കുവയ്ക്കുന്നതു ഡാറ്റ കൊണ്ട് റിഡൻഡൻസി കുറയ്ക്കും. ഈ പ്രസ്താവന ശരിയോ തെറ്റോ എന്ന് എഴുതുക.
5. ഡാറ്റ റിഡൻഡൻസി ഡാറ്റയുടെ സ്ഥിരത കുറയ്ക്കും. ശരിയോ തെറ്റോ എന്ന് എഴുതുക.

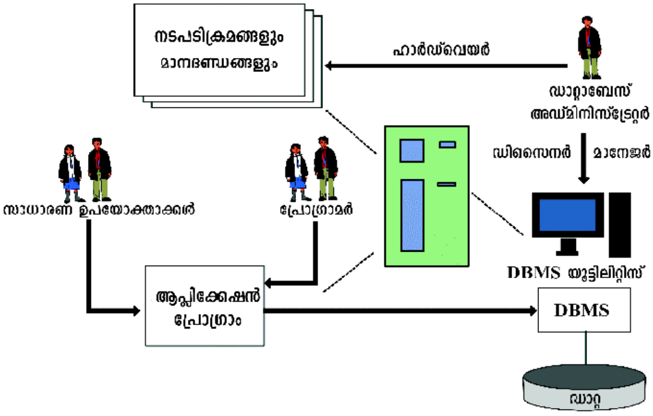
8.2 DBMS ഘടകങ്ങൾ (Components of the DBMS environment)

DBMS ന് ചുവടെ കൊടുത്തിരിക്കുന്ന വിധത്തിൽ നിരവധി ഘടകങ്ങൾ ഉണ്ട്. ഓരോന്നും അതിന്റെ പരിതസ്ഥിതിയിൽ വളരെ പ്രധാനമായ ചുമതലകൾ നിർവഹിക്കുന്നു.

- ഹാർഡ്‌വെയർ
- സോഫ്റ്റ്‌വെയർ
- ഡാറ്റ
- ഉപയോക്താക്കൾ
- നടപടിക്രമങ്ങൾ

ഹാർഡ്‌വെയർ: ഡാറ്റാബേസിന്റെ സംഭരണത്തിനും വീണ്ടെടുക്കലിനും ഉപയോഗിക്കുന്ന കമ്പ്യൂട്ടർ സിസ്റ്റം ആണ് ഹാർഡ് വെയർ. കമ്പ്യൂട്ടറുകൾ (പിസികൾ, വർക്ക്സ്റ്റേഷനുകൾ, സർവറുകൾ, സൂപ്പർകമ്പ്യൂട്ടറുകൾ), സംഭരണ ഉപകരണങ്ങൾ (ഹാർഡ് ഡിസ്കുകൾ, മാഗ്നെറ്റിക് ടേപ്പുകൾ), നെറ്റ്‌വർക്ക് ഉപകരണങ്ങൾ (ഹബ്, സ്വിച്ച്, റൗട്ടറുകൾ, ഫൈബർ ഓപ്റ്റിക്സ്), ഡാറ്റ സൂക്ഷിക്കുവാനുള്ള മറ്റു അനുബന്ധ ഉപകരണങ്ങൾ എന്നിവ ഇതിൽ ഉൾപ്പെടുന്നു.

സോഫ്റ്റ്‌വെയർ: DBMS, ആപ്ലിക്കേഷൻ പ്രോഗ്രാമുകൾ, യൂട്ടിലിറ്റികൾ എന്നിവ ഇതിൽ പെടും. ഉപയോക്താവിനും ഡാറ്റാബേസിനും ഇടയിലുള്ള ഒരു പാലമായി DBMS പ്രവർത്തിക്കുന്നു. മറ്റൊരുവിധത്തിൽ പറഞ്ഞാൽ, ഉപയോക്താക്കളുമായും ആപ്ലിക്കേഷൻ പ്രോഗ്രാമുകളുമായും ഡാറ്റാബേസുകളുമായും സംവദിക്കുന്ന സോഫ്റ്റ്‌വെയർ ആണ് DBMS. ഡാറ്റാബേസ് ഉപയോഗിക്കുന്ന സമയത്ത് ഉപയോക്താക്കളിൽ നിന്നുള്ള എല്ലാ അഭ്യർഥനകളും DBMS കൈകാര്യം ചെയ്യുന്നു. ഡാറ്റാനിർവചനം (data definition), ഡാറ്റ കൈകാര്യം ചെയ്യൽ (data manipulation), ഡാറ്റാ സുരക്ഷിതത്വം (data securtiy), ഡാറ്റ വിശ്വാസ്യത (data integrtiy), ഡാറ്റ വീണ്ടെടുക്കൽ (data recovery), പ്രകടനം മെച്ചപ്പെടുത്തൽ (performance optimization) തുടങ്ങിയ നിരവധി ചുമതലകൾ കൈകാര്യം ചെയ്യുന്ന നിരവധി സോഫ്റ്റ്‌വെയർ ഘടകങ്ങൾ DBMS സോഫ്റ്റ്‌വെയറിൽ ഉൾപ്പെടുന്നു. സങ്കീർണമായ ഹാർഡ്‌വെയർ വിശദാംശങ്ങളിൽ നിന്ന് ഡാറ്റാബേസിന് കവചം തീർക്കുന്ന ഒരു പൊതു സംവിധാനമാണ് DBMS. ഇത് ഡാറ്റയുടെ ഉപയോഗം നിയന്ത്രിക്കുകയും ഡാറ്റയുടെ സ്ഥിരത നിലനിർത്താൻ സഹായിക്കുകയും ചെയ്യുന്നു. റിപ്പോർട്ടുകൾ തയാറാക്കുക, തീരുമാനങ്ങൾ എടുക്കാൻ ആവശ്യമായ വിവരങ്ങൾ നൽകുക, ഡാറ്റാസമാഹരണം നടത്തുക എന്നിവയ്ക്കായി ഡാറ്റാബേസിൽ

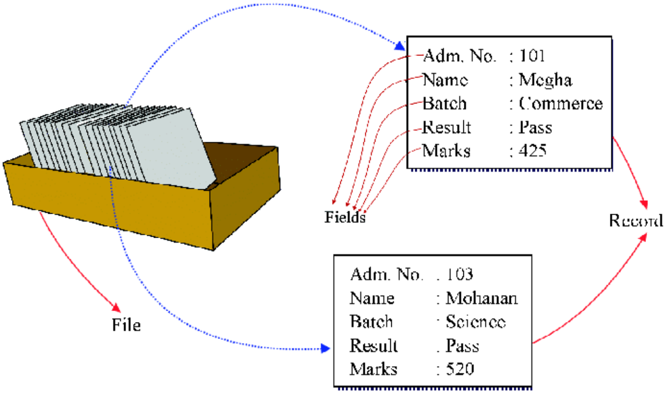


ചിത്രം 8.1: ഡാറ്റാബേസ് സിസ്റ്റം

ലഭ്യമായ ആപ്ലിക്കേഷൻ പ്രോഗ്രാമുകൾ ഉപയോഗിക്കുന്നു. ഡാറ്റാബേസ് സംവിധാനം കൈകാര്യം ചെയ്യുവാൻ സഹായിക്കുന്ന സോഫ്റ്റ്‌വെയറുകളാണ് യൂട്ടിലിറ്റികൾ. ഉദാഹരണത്തിന്, ഡാറ്റാബേസ് ഘടനകൾ നിർമ്മിക്കുക, ഡാറ്റാബേസ് ഉപയോഗം നിയന്ത്രിക്കുക, ഡാറ്റാബേസ് പ്രവർത്തനങ്ങൾ നിരീക്ഷിക്കുക തുടങ്ങിയവയ്ക്കായി എല്ലാ പ്രധാന DBMS കളും ഗ്രാഫിക്കൽ യൂസർ ഇന്റർഫേസുകൾ (GUIs) ഉപയോഗിക്കുന്നു.

ഡാറ്റ: അവസാനഘട്ട ഉപയോക്താക്കളുടെ കാഴ്ചപ്പാടിൽ DBMS പരിസ്ഥിതിയിലെ/ചട്ടക്കൂടിലെ ഏറ്റവും പ്രധാനപ്പെട്ട ഘടകം ഡാറ്റ തന്നെയാണ്. ഡാറ്റാബേസിൽ പ്രവർത്തനഡാറ്റയും മെറ്റാഡാറ്റയും (ഡാറ്റയെക്കുറിച്ചുള്ള ഡാറ്റ) അടങ്ങിയിരിക്കുന്നു. ഒരു സ്ഥാപനത്തിന് ആവശ്യമായ എല്ലാ ഡാറ്റയും ഡാറ്റാബേസിൽ ഉണ്ടായിരിക്കണം. യഥാർഥ ഡാറ്റയും ഡാറ്റ ഉപയോഗിക്കുന്ന പ്രോഗ്രാമുകളും പരസ്പരം വേർതിരിക്കപ്പെടുന്നു എന്നത് ഡേറ്റാബേസുകളുടെ പ്രധാന സവിശേഷതയാണ്. വിവരങ്ങൾ ഫലപ്രദമായി സംഭരിക്കുന്നതിനും വീണ്ടെടുക്കുന്നതിനും ഡാറ്റയെ ഫീൽഡുകൾ, റെക്കോർഡുകൾ, ഫയലുകൾ എന്നിങ്ങനെ ക്രമീകരിച്ചിരിക്കുന്നു.

ഒരു ക്ലാസിലെ വിദ്യാർഥികളുടെ അഡ്മിഷൻ നമ്പർ, പേര്, ബാച്ച്, പരീക്ഷാഫലം, മാർക്കുകൾ എന്നിവ ശേഖരിക്കുന്നതിനുള്ള കാർഡുകളുടെ ഒരു ശേഖരം സങ്കൽപ്പിക്കുക. ഓരോ കാർഡിനും സമാനഘടന ഉണ്ടായിരിക്കും, എന്നാൽ അവയിൽ രേഖപ്പെടുത്തിയിരിക്കുന്ന ഡാറ്റ ചിത്രം 8.2 ലെ പോലെ വ്യത്യസ്തമായിരിക്കും.



ചിത്രം 8.2: ഡാറ്റ സംസ്ഥാപന ആശയം

ഫീൽഡുകൾ: ശേഖരിച്ച ഡാറ്റയുടെ ഏറ്റവും ചെറിയ ഘടകമാണ് ഒരു ഫീൽഡ്. ഓരോ ഫീൽഡിലും ഒരു പ്രത്യേക തരം ഡാറ്റ അടങ്ങിയിരിക്കുന്നു. ചിത്രം 8.2 AdmNo, Name, Batch, Result, Marks എന്നിവ ഫീൽഡുകളാണ്.

റെക്കോർഡ്: അനുബന്ധ ഫീൽഡുകളുടെ ശേഖരമാണ് ഒരു റെക്കോർഡ്. മുകളിലുള്ള ചിത്രത്തിൽ ബോക്സിലെ ഓരോ കാർഡിലും ബന്ധപ്പെട്ട എല്ലാ ഫീൽഡുകളും ഉൾക്കൊള്ളുന്നു. ഉദാഹരണത്തിന്, 103, മോഹനൻ, സയൻസ്, പാസ്, 520 എന്നീ ഫീൽഡുകൾ ഉൾക്കൊള്ളുന്നതാണ് ഒരു റെക്കോർഡ്.

ഫയൽ: ഒരു ഫയൽ എന്നത് ഒരേ തരത്തിലുള്ള റെക്കോർഡുകളുടെ ഒരു ശേഖരമാണ്. ചിത്രം 8.2ൽ കാണിച്ചിരിക്കുന്ന ബോക്സ് ഒരു ഫയൽ ആയി കണക്കാക്കാം.

ഉപയോക്താക്കൾ: ഡാറ്റാബേസിലെ ഡാറ്റ ഉപയോഗിക്കുന്ന നിരവധി ഉപയോക്താക്കൾ ഉണ്ട്. DBMS ഉപയോഗിക്കുന്നവരുടെ ആശയവിനിമയത്തിന്റെ രീതി അനുസരിച്ച് ഒരു ഡാറ്റാബേസ് സംവിധാനത്തിന്റെ ഉപയോക്താക്കളെ തരം തിരിക്കാം. ഡാറ്റാബേസ് അഡ്മിനിസ്ട്രേറ്റർ (DBA), ആപ്ലിക്കേഷൻ പ്രോഗ്രാമർമാർ, സാങ്കേതിക പരിജ്ഞാനമുള്ള ഉപയോക്താക്കൾ (Sophisticated users), സാധാരണ ഉപയോക്താക്കൾ (Naive Users) എന്നിവയാണ് വിവിധ വിഭാഗങ്ങളിലെ ഉപയോക്താക്കൾ.

നടപടിക്രമങ്ങൾ: ഡാറ്റാബേസിന്റെ രൂപകല്പനയും ഉപയോഗവും നിയന്ത്രിക്കുന്ന നിർദേശങ്ങളും ചട്ടങ്ങളുമാണ് നടപടിക്രമങ്ങൾ. സംവിധാനത്തിന്റെ ഉപയോക്താക്കൾക്കും ഡാറ്റാബേസ് കൈകാര്യം ചെയ്യുന്ന വ്യക്തികൾക്കും സിസ്റ്റം എങ്ങനെ ഉപയോഗിക്കണം അല്ലെങ്കിൽ പ്രവർത്തിപ്പിക്കണമെന്ന് രേഖപ്പെടുത്തിയ പ്രമാണങ്ങൾ ആവശ്യമാണ്. ചുവടെ കൊടുത്തിരിക്കുന്ന പ്രവർത്തനങ്ങൾക്കാവശ്യമായ നിർദേശങ്ങൾ ഇത്തരം രേഖകളിൽ ഉണ്ടായിരിക്കും.

i) DBMS ലേക്ക് ലോഗിൻ ചെയ്യുക.

ii) ഒരു നിശ്ചിത DBMS സംവിധാനം അല്ലെങ്കിൽ ആപ്ലിക്കേഷൻ പ്രോഗ്രാം ഉപയോഗിക്കുക.

iii) DBMS പ്രവർത്തനം ആരംഭിക്കുകയും അവസാനിപ്പിക്കുകയും ചെയ്യുക.

iv) ഡാറ്റാബേസിന്റെ പകർപ്പ് എടുക്കുകയും ഹാർഡ്‌വെയറിന്റെയോ സോഫ്റ്റ്‌വെയറിന്റെയോ തകരാറുകൾ പരിഹരിക്കുകയും ചെയ്യുക.

v) വിവിധ ഡിസ്കുകളിലായി വ്യാപിച്ചിരിക്കുന്ന ഡാറ്റാബേസ് തിരിച്ചറിയുകയും പ്രവർത്തനം മെച്ചപ്പെടുത്തുകയും ദ്വിതീയ സംഭരണിയിലേക്ക് ചരിത്രരേഖയായി ഡാറ്റ സൂക്ഷിക്കുകയും ചെയ്യുക.

രണ്ടാം സംഭരണിയിലേക്ക് ആർക്കൈവ് ഡാറ്റയായി (Archive Daa) സൂക്ഷിക്കുകയും ചെയ്യുക.

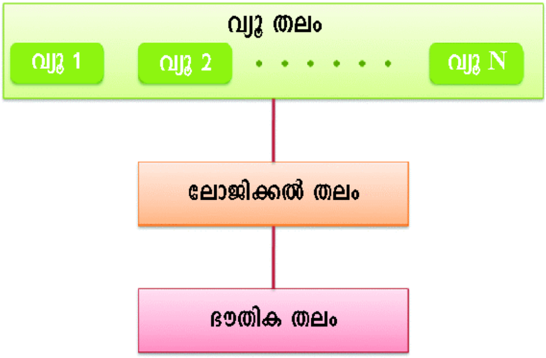
നിങ്ങളുടെ പഠനപുരോഗതി അറിയുക

1. ഡാറ്റയെ കുറിച്ചുള്ള ഡാറ്റ ____________ എന്നറിയപ്പെടുന്നു.
2. ഡാറ്റാബേസ് സംവിധാനത്തിന്റെ പ്രധാന ഘടകങ്ങളുടെ പേര് എഴുതുക?
3. താഴെ പറയുന്ന DBMS ന്റെ ഘടകങ്ങളെ തരം തിരിക്കുക:

   ഹാർഡ് ഡിസ്ക്ക്, സ്വിച്ച് , ഡിബിഎ, പേറോൾ സംവിധാനം, സാധാരണ ഉപയോക്താവ്, റെയിൽവേ റിസർവേഷൻ സംവിധാനം.

8.3 ഡാറ്റ സംഗ്രഹവും ഡാറ്റ സ്വാതന്ത്ര്യവും (Data abstraction and data independence)

ഒരു സിസ്റ്റം ഉപയോഗയോഗ്യമാക്കുവാൻ ഡാറ്റ കാര്യക്ഷമമായി വീണ്ടെടുക്കണം. കാര്യക്ഷമതയോടെ ഡാറ്റയെ ഡാറ്റാബേസിൽ പ്രതിനിധാനം ചെയ്യുന്നതിനായി ഡിസൈനർ (designer) സങ്കീർണമായ ഡാറ്റഘടനകൾ ഉപയോഗപ്പെടുത്തുന്നു. ഡാറ്റാബേസ് ഉപയോക്താക്കളിലധികവും കമ്പ്യൂട്ടർ പരിജ്ഞാനം ഇല്ലാത്തവരായതിനാൽ, കമ്പ്യൂട്ടർ പ്രോഗ്രാമുകൾ വികസിപ്പിക്കുന്നവർ ഡാറ്റാബേസ് സങ്കീർണത പലതലത്തിലുള്ള ഡാറ്റ സംഗ്രഹം വഴി ഉപയോക്താക്കളിൽ നിന്ന് മറയ്ക്കുന്നു.



ചിത്രം 8.3: ഡാറ്റാ സംഗ്രഹത്തിന്റെ തലങ്ങൾ

ചിത്രം 8.3 ൽ വിവരിച്ചിരിക്കുന്നതുപോലെ, ഒരു DBMS ലെ ഡാറ്റസംഗ്രഹം മൂന്നുതലങ്ങളിലായി വിവരിച്ചിരിക്കുന്നു ഭൗതിക തലം, ലോജിക്കൽ തലം, വ്യൂ തലം.

a. ഭൗതിക തലം (Physical level)

ഡിസ്കുകൾ, ടേപ്പുകൾ മുതലായ രണ്ടാംതര സംഭരണ ഉപകരണങ്ങളിൽ എങ്ങനെയാണ് ഡാറ്റ സൂക്ഷിച്ചിരിക്കുന്നത് എന്ന് ഏറ്റവും താഴ്ന്ന തലത്തിലുള്ള ഡാറ്റ സംഗ്രഹം സൂചിപ്പിക്കുന്നു. ഫിസിക്കൽ തലം താഴ്ന്ന തലത്തിലുള്ള സങ്കീർണ ഡാറ്റാഘടന വിശദീകരിക്കുന്നു. റിലേഷനുകൾ (പട്ടിക രൂപത്തിലുള്ള ഡാറ്റ) സൂക്ഷിക്കുന്നതിനും വേഗത്തിൽ ഡാറ്റ വീണ്ടെടുക്കുന്നതിനായി ഇൻഡക്സുകൾ എന്നറിയപ്പെടുന്ന സഹായക ഡാറ്റ ഘടന നിർമ്മിക്കുന്നതിനും ഏത് ഫയൽ വ്യവസ്ഥ ഉപയോഗിക്കണമെന്ന് നാം തീരുമാനിക്കേണ്ടതുണ്ട്. SWS (ഏകജാലക സംവിധാനം) ഡാറ്റാബേസിലെ ഭൗതിക ഡാറ്റ ഘടനയുടെ ഒരു മാതൃക താഴെക്കൊടുക്കുന്നു.

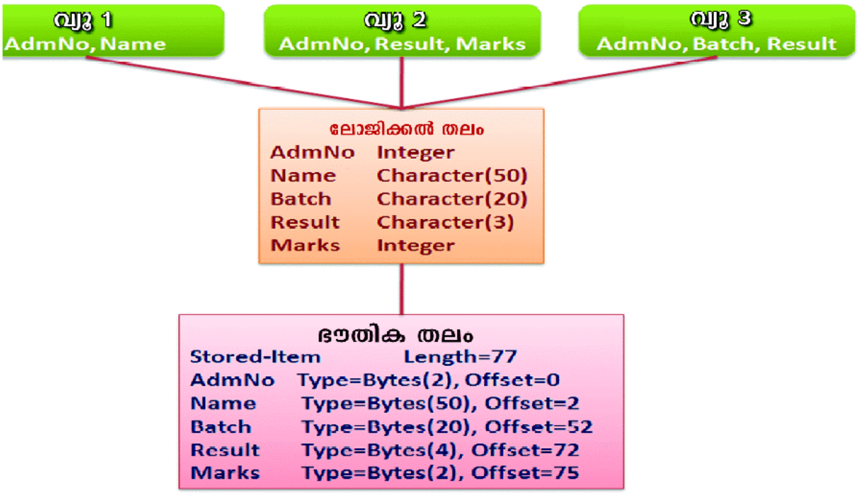
- എല്ലാ റിലേഷനുകളും ക്രമമല്ലാത്ത റെക്കോർഡുകൾ ഉള്ള ഫയലുകളായി സംഭരിക്കുക. (DBMSലെ ഒരു ഫയൽ ഡാറ്റയുടെയോ റെക്കോർഡുകളുടെയോ ശേഖരമാണ്):
- സ്റ്റുഡന്റ്, സ്കൂൾ, കോഴ്സ് എന്നീ ഫയലുകളിൽ ആദ്യ കോളത്തിനെ അടിസ്ഥാനമാക്കി ഇൻഡക്സുകൾ സൃഷ്ടിക്കുക.

b. ലോജിക്കൽ തലം (Logical level)

ഡാറ്റ സംഗ്രഹത്തിന്റെ അടുത്ത ഉയർന്ന തലമായ ലോജിക്കൽ തലം ഡാറ്റാബേസിൽ എന്ത് ഡാറ്റയാണ് സംഭരിക്കുന്നതെന്നും, ഡാറ്റകൾ തമ്മിൽ എന്ത് ബന്ധമാണ് നിലനിൽക്കുന്നതെന്നും വിശദമാക്കുന്നു. താരതമ്യേന വളരെ ലളിതമായ ഘടനകളെ അടിസ്ഥാനമാക്കിയാണ് ഡാറ്റാബേസിലെ ലോജിക്കൽ തലം വിശദീകരിക്കുന്നത്. ലോജിക്കൽ തലത്തിൽ ലളിതമായ ഘടനകൾ നടപ്പിലാക്കുന്നതിന്, ഭൗതിക തലത്തിൽ സങ്കീർണമായ ഘടനകൾ ഉപയോഗിക്കുന്നു. ലോജിക്കൽതലത്തിലുള്ള ഉപയോക്താവിന് ഈ സങ്കീർണത അറിഞ്ഞിരിക്കേണ്ട ആവശ്യമില്ല. ഡാറ്റാബേസിൽ എന്ത് വിവരമാണ് സൂക്ഷിക്കേണ്ടത് എന്ന് തീരുമാനിക്കുന്ന ഡാറ്റാബേസ് അഡ്മിനിസ്ട്രേറ്റർമാർ ഡാറ്റ സംഗ്രഹത്തിന്റെ ലോജിക്കൽ തലമാണ് ഉപയോഗിക്കുന്നത്. ലോജിക്കൽ ഡാറ്റാതലത്തെ കൺസെപ്ച്വൽ ഡാറ്റാതലം എന്നും പറയുന്നു.

c. വ്യൂ തലം (View level)

ഡാറ്റാബേസ് സംഗ്രഹത്തിന്റെ ഏറ്റവും ഉയർന്ന തലമായ വ്യൂ തലം, ഉപയോക്താക്കളോട് ഏറ്റവും അടുത്ത് നിൽക്കുന്നു. ഓരോ ഉപയോക്താക്കളും ഡാറ്റ കാണുന്ന രീതിയാണ് വ്യൂതലം. ഈ തലത്തിൽ ഡാറ്റാബേസിന്റെ ഒരു ഭാഗം മാത്രമേ വിവരിക്കുന്നുള്ളൂ. ഡാറ്റാബേസിലെ മിക്ക ഉപയോക്താക്കൾക്കും ഡാറ്റാബേസിൽ അടങ്ങിയിരിക്കുന്ന എല്ലാ വിവരങ്ങളും ആവശ്യമില്ല. പകരം അവർക്ക് ഡാറ്റാബേസിന്റെ അവശ്യമായ ഭാഗം മാത്രമേ ഉപയോഗിക്കേണ്ടതുള്ളൂ. ഇത് DBMSമായുള്ള അവരുടെ ഇടപെടൽ ലളിതമാക്കുന്നു. ഒരു ഡാറ്റാബേസിനായി നിരവധി വ്യൂ തലങ്ങൾ ഉണ്ടായേക്കാം. ചിത്രം 8.4ൽ, AdmNo, Name, Batch, Result, Marks എന്നീ ഫീൽഡുകളുള്ള STUDENT ഫയലിന്റെ ഡാറ്റാബേസ് സംഗ്രഹത്തിന്റെ മൂന്നു തലങ്ങൾ കാണിക്കുന്നു.



ചിത്രം 8.4: ഡാറ്റ സംഗ്രഹത്തിന്റെ തലങ്ങളുടെ ഉദാഹരണം

8.3.1 ഡാറ്റാസ്വാതന്ത്ര്യം (Data independence)

ഒരു ഡാറ്റാബേസിനെ അതിന്റെ മൂന്ന്തലത്തിലുള്ള ഡാറ്റാസംഗ്രഹം വഴി വീക്ഷിക്കുമ്പോൾ, ഒരു നിശ്ചിത തലത്തിലെ ഡാറ്റാബേസ് ഘടനയിൽ എന്തെങ്കിലും മാറ്റമുണ്ടാകുന്നത് മറ്റ് തലങ്ങളുടെ ഘടനയെ ബാധിച്ചേക്കാം. ഡാറ്റാബേസിൽ നടത്തുന്ന നിരന്തരമായ മാറ്റങ്ങൾ ഡാറ്റാബേസിന്റെ പുനക്രമീകരണത്തിന് കാരണമാകരുത്. തൊട്ടടുത്ത തലത്തിലെ ഡാറ്റാഘടനയുടെ നിർവചനത്തെ ബാധിക്കാതെ ഒരു തലത്തിലെ ഡാറ്റാഘടനയിൽ മാറ്റം വരുത്തുവാനുള്ള കഴിവിനെ ഡാറ്റാസ്വാതന്ത്ര്യം എന്ന് പറയുന്നു. ഡാറ്റാസ്വാതന്ത്ര്യത്തിന് രണ്ടു തലങ്ങളുണ്ട് - ഭൗതിക ഡാറ്റാസ്വാതന്ത്ര്യവും (physical data independence) ലോജിക്കൽ ഡാറ്റാസ്വാതന്ത്ര്യവും (logical data independence).

a. ഭൗതിക ഡാറ്റാസ്വാതന്ത്ര്യം (Physical data independence)

ലോജിക്കൽ തലത്തിലെ ഡാറ്റാഘടനയെ ബാധിക്കാതെ ഭൗതികതലത്തിലെ ഡാറ്റാഘടനയിൽ മാറ്റം വരുത്തുവാനുള്ള കഴിവിനെ ഭൗതിക ഡാറ്റാസ്വാതന്ത്ര്യം എന്നു പറയുന്നു. അതായത് ഭൗതിക തലത്തിലെ ഡാറ്റാഘടനയിൽ മാറ്റം വരുത്തിയാലും ആപ്ലിക്കേഷൻ പ്രോഗ്രാമുകൾ മാറ്റമില്ലാതെ ഒരേപോലെ തന്നെ തുടരും.

b. ലോജിക്കൽ ഡാറ്റാസ്വാതന്ത്ര്യം (Logical data independence)

ബാഹ്യതലത്തിലെ (വ്യൂ തലം) ഡാറ്റാഘടനയെ ബാധിക്കാതെ ലോജിക്കൽ തലത്തിലെ ഡാറ്റാഘടനയിൽ മാറ്റം വരുത്താനുള്ള കഴിവിനെ ലോജിക്കൽ ഡാറ്റ സ്വാതന്ത്ര്യം എന്നു പറയുന്നു. ആപ്ലിക്കേഷൻ പ്രോഗ്രാമുകളുടെ തൽസ്ഥിതി ഉറപ്പാക്കുന്നത് ലോജിക്കൽ ഡാറ്റാസ്വാതന്ത്ര്യമാണ്. ആപ്ളിക്കേഷൻ പ്രോഗ്രാമുകൾ ഡാറ്റാബേസിന്റെ ലോജിക്കൽ ഘടനയെ കൂടുതൽ ആശ്രയിച്ചിരിക്കുന്നതു കൊണ്ട്, ഭൗതിക ഡാറ്റാസ്വാതന്ത്ര്യത്തെക്കാൾ ലോജിക്കൽ ഡാറ്റാസ്വാതന്ത്ര്യം കൈവരിക്കാൻ വളരെ പ്രയാസമാണ്.

8.4 ഡാറ്റാബേസിന്റെ ഉപയോക്താക്കൾ (Users of database)

വൈദഗ്ധ്യത്തിന്റെ തോതനുസരിച്ച് അല്ലെങ്കിൽ DBMS കൈകാര്യം ചെയ്യുന്ന രീതിയുടെ അടിസ്ഥാനത്തിൽ ഒരു ഡാറ്റാബേസ് സംവിധാനത്തിന്റെ ഉപയോക്താക്കളെ ചുവടെ പ്പറയുന്ന ഗ്രൂപ്പുകളിൽ ഉൾപ്പെടുത്താം:

- ഡാറ്റാബേസ് അഡ്മിനിസ്ട്രേറ്റർ (DBA)
- ആപ്ലിക്കേഷൻ പ്രോഗ്രാമർമാർ
- സാങ്കേതിക പരിജ്ഞാനമുള്ള ഉപയോക്താക്കൾ
- സാധാരണ ഉപയോക്താക്കൾ

8.4.1 ഡാറ്റാബേസ് അഡ്മിനിസ്ട്രേറ്റർ (DBA)

കേന്ദ്രീകൃതവും പങ്കുവയ്ക്കപ്പെട്ടതുമായ ഡാറ്റാബേസിന്റെ നിയന്ത്രണച്ചുമതലയുള്ള വ്യക്തിയാണ് ഡാറ്റാബേസ് അഡ്മിനിസ്ട്രേറ്റർ (DBA). DBA പല സുപ്രധാന ജോലികളും നിർവഹിക്കേണ്ടതാണ്.

കൺസെപ്ചൽ, ഫിസിക്കൽ സ്കീമകളുടെ രൂപകല്പന: DBMS ൽ സംഭരിക്കേണ്ട ഡാറ്റ എന്താണെന്നും അത് എങ്ങനെ ഉപയോഗപ്പെടുത്താം എന്നും മനസ്സിലാക്കാൻ സംവിധാനത്തിന്റെ ഉപയോക്താക്കളുമായി ആശയവിനിമയം നടത്തുന്നത് DBA യാണ്. ഈ അറിവ് അടിസ്ഥാനമാക്കി, DBA കൺസെപ്ചൽ സ്കീമയും ഫിസിക്കൽ സ്കീമയും രൂപപ്പെടുത്തണം.

സുരക്ഷയും അംഗീകാരവും: ഡാറ്റയുടെ ശരിയായ ഉപയോഗം ഉറപ്പാക്കുന്നതിനുള്ള ഉത്തരവാദിത്വം DBAൽ നിക്ഷിപ്തമാണ്. ഉദാഹരണമായി, ഒരു സ്കൂളിൽ, അധ്യാപകർ വിദ്യാർഥികൾക്ക് പഠന മേഖലകൾ പരിചയപ്പെടാനും, വിദ്യാർഥിയുടെ പരീക്ഷാ ഫലങ്ങൾ അറിയാനും, ഒരു വിഷയം പഠിപ്പിക്കുന്ന അധ്യാപകരുടെ വിശദാംശങ്ങൾ കണ്ടെത്തുവാനും അനുവാദം നൽകുന്നു. അതേസമയം വിദ്യാർഥികൾക്ക് അധ്യാപകരുടെ ശമ്പളവിവരങ്ങളോ അല്ലെങ്കിൽ മറ്റ് വിദ്യാർഥികളുടെ ഗ്രേഡ് വിവരങ്ങളോ കാണാൻ അനുവാദമില്ല. പഠനമേഖലകളെ കുറിച്ചുള്ള ഒരു വ്യൂ മാത്രം ഉപയോഗിക്കാൻ വിദ്യാർഥികൾക്ക് അനുമതി നൽകിക്കൊണ്ട് DBA യ്ക്ക് ഈ നയം നടപ്പിലാക്കാൻ കഴിയും.

ഡാറ്റാലഭ്യതയും നഷ്ടത്തിൽ നിന്നുള്ള വീണ്ടെടുക്കലും: ഒരു ഇടപാട് പൂർത്തിയാക്കുന്നതിനോ തകരാർ പരിഹരിക്കുന്നതിനോ സംവിധാനം പരാജയപ്പെടുമ്പോൾ ഡാറ്റ ഒരു സ്ഥിരാവസ്ഥയിലേക്ക് പുനഃസ്ഥാപിക്കുന്നതിനായി DBA നടപടികൾ എടുക്കേണ്ടതാണ്. ഈ പ്രവർത്തനങ്ങൾക്കുള്ള സോഫ്റ്റ്വെയർ പിന്തുണ DBMS നൽകുന്നുണ്ട്, പക്ഷേ, അതാതു സമയം ഡാറ്റ ബാക്കപ്പ് ചെയ്യുവാനും സിസ്റ്റം പ്രവർത്തനത്തിന്റെ ലോഗുകൾ

(ഡാറ്റാബേസിലെ പ്രവർത്തനങ്ങളായ ഡാറ്റ കൂട്ടിച്ചേർക്കൽ, ഒഴിവാക്കൽ, പരിഷ്കരിക്കൽ, തുടങ്ങിയ പ്രക്രിയകൾ സൂക്ഷിക്കുന്ന ഫയലുകൾ) പരിപാലിക്കാനും ആവശ്യമായ നടപടിക്രമങ്ങൾ സ്വീകരിക്കേണ്ടത് DBA ആണ്.

8.4.2 ആപ്ലിക്കേഷൻ പ്രോഗ്രാമർമാർ (Application programmers)

ആപ്ലിക്കേഷൻ പ്രോഗ്രാമുകൾ വഴി DBMS മായി സംവദിക്കുന്ന കമ്പ്യൂട്ടർ പ്രൊഫഷണലുകളാണ് ആപ്ലിക്കേഷൻ പ്രോഗ്രാമർമാർ. വിഷ്വൽ ബേസിക്, സി, സി++, ജാവാ പോലുള്ള കമ്പ്യൂട്ടർ ഭാഷകൾ ഉപയോഗിച്ച് എഴുതുകയും ഡാറ്റ കൈകാര്യം ചെയ്യുന്ന ഭാഷ (DML) യിലൂടെ DBMS മായി സംവദിക്കുകയും ചെയ്യുന്ന പ്രോഗ്രാമുകളാണ് ആപ്ലിക്കേഷൻ പ്രോഗ്രാമുകൾ. ബാഹ്യസ്കീമയിലൂടെ ഡാറ്റയെ മികച്ച രീതിയിൽ ആപ്ലിക്കേഷൻ പ്രോഗ്രാമുകൾ ഉപയോഗിക്കുന്നു

8.4.3 സാങ്കേതിക പരിജ്ഞാനമുള്ള ഉപയോക്താക്കൾ (Sophisticated users)

എൻജിനീയർമാർ, ശാസ്ത്രജ്ഞർ, ബിസിനസ്സ് വിശകലന വിദഗ്ധർ, DBMS സൗകര്യങ്ങൾ നന്നായി പരിചയമുള്ളവർ എന്നിവരാണ് ഈ വിഭാഗത്തിൽപ്പെടുന്നത്. അവരുടെ സങ്കീർണമായ ആവശ്യകതകൾ നിറവേറ്റുന്നതിനായി സ്വന്തം ക്വറികളിലൂടെ (ഡാറ്റാബേസ് അഭ്യർഥന) അവർ DBMS-മായി സംവദിക്കുന്നു.

8.4.4 സാധാരണ ഉപയോക്താക്കൾ (Naive users)

മുമ്പേ എഴുതി തയാറാക്കിയ പ്രോഗ്രാമുകളിൽ കൂടി DBMS സംവിധാനവുമായി സംവദിക്കുന്നവരാണ് സാധാരണ ഉപയോക്താക്കൾ. DBMSന്റെ വിശദാംശങ്ങൾ അവർ അറിയുകയോ ശ്രദ്ധിക്കുകയോ ഇല്ല. സാധാരണ ഉപയോക്താക്കൾ ഉയർന്നതലത്തിലുള്ള ഡാറ്റ സംഗ്രഹം മാത്രം കൈകാര്യം ചെയ്യുന്നു. വെബിലൂടെ ഡാറ്റ ഉപയോഗിക്കുന്ന ആളുകൾ, ഒരു കാര്യാലയത്തിലെ ക്ളറിക്കൽ ജീവനക്കാർ, ഒരു സൂപ്പർമാർക്കറ്റിലെ അല്ലെങ്കിൽ ഹോട്ടലുകളിലെ ബില്ലിങ് ക്ലർക്ക്, ബാങ്ക് ക്ലാർക്ക് മുതലായവർ സാധാരണ ഉപയോക്താക്കളാണ്.

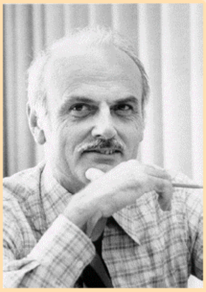
നിങ്ങളുടെ പഠനപുരോഗതി അറിയുക

1. ചോദ്യോത്തര രൂപത്തിൽ ഡാറ്റാബേസുമായി സംവദിക്കുന്ന ഉപയോക്താവ് ________ എന്നറിയപ്പെടുന്നു.
2. ഒരു സൂപ്പർമാർക്കറ്റിലെ ബില്ലിംഗ് ക്ലാർക്ക് ഏതുതരം ഉപയോക്താവാണ്?
3. ഒരു ഡാറ്റാബേസിന് ഡാറ്റാസുരക്ഷ നൽകുന്ന ഉപയോക്താവാരാണ്?
4. ഒരു ഡാറ്റാബേസിന്റെ ഘടനയെ മാറ്റുന്ന ഉപയോക്താവാരാണ് ?
5. ആപ്ളിക്കേൻ പ്രോഗ്രാം വഴി ഡാറ്റാബേസുമായി സംവദിക്കുന്ന ഉപയോക്താവ് ________ എന്നറിയപ്പെടുന്നു.

8.5 റിലേഷണൽ ഡാറ്റാമോഡൽ (Relational data model)

വ്യത്യസ്ത പേരുകളാൽ തിരിച്ചറിയാവുന്നതും റിലേഷനുകൾ എന്നറിയപ്പെടുന്നതുമായ ഒരു കൂട്ടം പട്ടികകളുടെ ശേഖരമായി ഡാറ്റാ ബേസിനെ പ്രതിനിധാനം ചെയ്യുന്ന

മാതൃകയാണ് റിലേഷണൽ ഡാറ്റാമോഡൽ റിലേഷണൽ ഡാറ്റാമോഡലിൽ ഡാറ്റയും അവ തമ്മിലുള്ള ബന്ധങ്ങളും പട്ടിക രൂപത്തിൽ അവതരിപ്പിക്കുവാൻ സാധിക്കും. ഈ അവതരണരീതി ഒരു ഡാറ്റാബേസിന്റെ ആശയങ്ങൾ എളുപ്പത്തിൽ മനസ്സിലാക്കാൻ സഹായിക്കുന്നു.



എഡ്ഗാർ ഫ്രാങ്ക് കോഡ് (19 ഓഗസ്റ്റ് 1923 - 18 ഏപ്രിൽ 2003) എന്ന കമ്പ്യൂട്ടർ ശാസ്ത്രജ്ഞനാണ് ഡാറ്റാ മാനേജ്മെന്റിനായുള്ള റിലേഷണൽ മോഡൽ കണ്ടുപിടിച്ചത്. ഇംഗ്ലണ്ടിലെ പോർട്ട്ലാൻഡ് ദ്വീപിൽ അദ്ദേഹം ജനിച്ചു. രണ്ടാം ലോകമഹായുദ്ധകാലത്ത് റോയൽ എയർ ഫോഴ്സിൽ പൈലറ്റ് ആയി സേവനമനുഷ്ഠിച്ചു. 1948 ൽ അദ്ദേഹം ഐ.ബി.എം.ൽ ചേർന്നു. 1981 ൽ അദ്ദേഹം ടൂറിംഗ് അവാർഡ് കരസ്ഥമാക്കി. 2003 ഏപ്രിൽ 18ന് 79-ാം വയസ്സിൽ ഫ്ളോറിഡയിലുള്ള വില്യംസ് ദ്വീപിലെ തന്റെ ഭവനത്തിൽ വച്ച് ഹൃദയാഘാതത്തെ തുടർന്ന് അദ്ദേഹം മരണമടഞ്ഞു.

ഇന്ന്, ഡാറ്റാബേസ് ഉൽപ്പന്നങ്ങളിൽ ഭൂരിഭാഗവും റിലേഷണൽ മോഡൽ അടിസ്ഥാനമാക്കിയുള്ളതാണ്. അവ റിലേഷണൽ ഡാറ്റാബേസ് നിർവഹണ സംവിധാനം (RDBMS) എന്നാണ് അറിയപ്പെടുന്നത്. ഡാറ്റാപ്രാതിനിധ്യവും സങ്കീർണമായ അന്വേഷണങ്ങളും ലളിതമായി അവതരിപ്പിക്കാൻ കഴിയുന്നു എന്നതാണ് മറ്റ് ഡാറ്റാമോഡലുകളിൽ നിന്ന് റിലേഷണൽ മോഡലിനെ വ്യത്യസ്തമാക്കുന്നത്. ഒറാക്കിൾ, മൈക്രോസോഫ്റ്റ് SQL സെർവർ, MySQL, DB2, ഇൻഫോമിക്സ്, ഇൻഗ്രസ് എന്നിവയാണ് ജനപ്രിയ RDBMS പാക്കേജുകൾ.

വാണിജ്യപരമായി ഏറ്റവുമധികം ഉപയോഗിക്കുന്ന റിലേഷണൽ മോഡൽ ഡാറ്റാബേസ് സംവിധാനങ്ങളിൽ സ്ട്രക്ചേർഡ് ക്വറി ലാംഗ്വേജ് (SQL), ക്വറി ബൈ എക്സാംബിൾ (QBE) അല്ലെങ്കിൽ ഡാറ്റലോഗ് എന്നിവ ഉൾപ്പെടുന്ന ഒരു ചോദ്യഭാഷ അടങ്ങിയിരിക്കുന്നു. വളരെ വ്യാപകമായി ഉപയോഗിക്കുന്ന SQL അടുത്ത അധ്യായത്തിൽ നമ്മൾ പഠിക്കും.

8.6 RDBMS ലെ പദാവലികൾ (Terminologies in RDBMS)

റിലേഷണൽ ഡാറ്റാബേസുകളിലെ പ്രവർത്തനങ്ങൾ ചർച്ച ചെയ്യുന്നതിനു മുൻപ് RDBMS മായി ബന്ധപ്പെട്ട ചില പദാവലികൾ പരിചയപ്പെടാം.

a). എന്റിറ്റി

മറ്റുള്ളവയിൽ നിന്ന് വേർതിരിച്ചറിയാവുന്ന ഒരു വ്യക്തിയെ അല്ലെങ്കിൽ ഒരു വസ്തുവിനെ എന്റിറ്റി എന്ന് പറയുന്നു. ഉദാഹരണത്തിന്, ഓരോ സ്കൂളും ഓരോ വിദ്യാർഥിയും ഓരോ വ്യത്യസ്ത എന്റിറ്റിയാണ്.

b) റിലേഷൻ

ഡാറ്റാശേഖരത്തെ വരികളും നിരകളുമായി ക്രമീകരിക്കുന്നതാണ് റിലേഷൻ. പട്ടിക എന്നും ഇത് അറിയപ്പെടുന്നു. STUDENT എന്ന് പേരുള്ള ഒരു റിലേഷൻ, പട്ടിക 8.1 ൽ കാണിച്ചിരിക്കുന്നു.

STUDENT relation

AdmNo	Roll	Name	Batch	Marks	Result
101	24	Sachin	Science	480	EHS
102	14	Rahul	Commerce	410	EHS
103	4	Fathima	Humanities	200	NHS
104	12	Mahesh	Commerce	180	NHS
105	24	Nelson	Humanities	385	EHS
106	8	Joseph	Commerce	350	EHS
107	24	Shaji	Humanities	205	NHS
108	2	Bincy	Science	300	EHS

പട്ടിക 8.1 STUDENT റിലേഷൻ

c) ടൂപിൾ

ഒരു റിലേഷനിലെ വരികളെ (റെക്കോർഡുകൾ) സാധാരണയായി ടൂപിൾ എന്ന് പറയുന്നു. ഒരു പ്രത്യേക എന്റിറ്റിയെ പ്രതിനിധീകരിക്കാൻ ഉപയോഗിക്കുന്ന മൂല്യങ്ങളുടെ ഒരു പൂർണഗണം ഒരു വരി ഉൾക്കൊള്ളുന്നു. പട്ടിക 8.1 ൽ, STUDENT റിലേഷനിലെ ഓരോ വരിയും ഓരോ വിദ്യാർഥിയുടെ പരീക്ഷാഫലത്തെ പ്രതിനിധീകരിക്കുന്നു.

d) ആട്രിബ്യൂട്ട്

ഒരു റിലേഷനിലെ നിരകളെ ആട്രിബ്യൂട്ടുകൾ എന്ന് വിളിക്കുന്നു. AdmNo, Roll, Name, Batch, Marks, Result എന്നിവ STUDENT റിലേഷന്റെ ആട്രിബ്യൂട്ടുകളാണ്. ഓരോ ആട്രിബ്യൂട്ടിന്റെയും മൂല്യങ്ങൾ ഡൊമെയ്ൻ എന്ന മൂല്യങ്ങളുടെ സാധുത പട്ടികയിൽ നിന്ന് എടുക്കുന്നു.

e) ഡിഗ്രി

ഒരു റിലേഷനിലെ നിരകളുടെ എണ്ണം ആ റിലേഷന്റെ ഡിഗ്രി നിർണയിക്കുന്നു. STUDENT എന്ന റിലേഷന് ആറ് നിരകൾ അല്ലെങ്കിൽ ആട്രിബ്യൂട്ടുകൾ ഉണ്ട്, അതിനാൽ STUDENT റിലേഷന്റെ ഡിഗ്രി ആറ് ആകുന്നു.

f) കാർഡിനാലിറ്റി

ഒരു റിലേഷനിലെ വരികളുടെ അല്ലെങ്കിൽ ടൂപിളുകളുടെ എണ്ണം കാർഡിനാലിറ്റി എന്നറിയപ്പെടുന്നു. STUDENT റിലേഷനിൽ എട്ടു വരികൾ ഉണ്ട്, അതിനാൽ STUDENT റിലേഷന്റെ കാർഡിനാലിറ്റി 8 ആണ്.

g) ഡൊമെയ്ൻ

ഒരു നിരയെ പ്രതിനിധാനം ചെയ്യുന്ന മൂല്യങ്ങളുടെ ഗണമാണ് ഒരു ഡൊമെയ്ൻ. ഉദാഹരണത്തിന്, പട്ടിക 8.1 ൽ കാണിച്ചിരിക്കുന്ന STUDENT ബന്ധത്തിൽ Batch എന്ന നിരയുടെ ഡൊമെയ്ൻ {Science, Humanities, Commerce} എന്ന മൂല്യങ്ങളുടെ ഗണമാണ്. അതായത്, ഈ ഗണത്തിൽ നിന്നുള്ള ഏതെങ്കിലുമൊരു മൂല്യം Batch നിരയിൽ ദൃശ്യമാകാം. അതുപോലെ തന്നെ, Result എന്ന നിരയുടെ ഡൊമെയ്ൻ ആണ് {EHS, NHS}.

h) സ്കീമ

ഒരു ഡാറ്റാബേസ് രൂപകല്പന ചെയ്യുന്ന സന്ദർഭത്തിൽ വ്യക്തമാക്കപ്പെടുന്ന ഡാറ്റാബേസിന്റെ ഘടനയെയോ വിവരണത്തെയോ ഡാറ്റാബേസ് സ്കീമ എന്നുവിളിക്കുന്നു. റിലേഷണൽ മോഡലിൽ, ഒരു റിലേഷന്റെ സ്കീമ അതിന്റെ പേര്, ഓരോ നിരയുടെയും പേര്, ഓരോ നിരയുടെയും ഡാറ്റ തരം എന്നിവ വ്യക്തമാക്കുന്നു. ഉദാഹരണമായി, ഒരു സ്കൂൾ ഡാറ്റാബേസിലെ വിദ്യാർഥികളുടെ വിവരങ്ങൾ ഇനിപ്പറയുന്ന ഘടനയുമായി ബന്ധപ്പെടുത്തി സൂക്ഷിക്കാം:

```
STUDENT     (Admno : integer,
             Roll   : integer,
             Name   : character(50),
             Batch  : character(20),
             Marks  : decimal,
             Result : character(4))
```

i) ഇൻസ്റ്റൻസ്

റിലേഷണൽ സ്കീമയിൽ നിർവചിച്ചിരിക്കും പ്രകാരം തുല്യയെണ്ണം നിരകളുള്ള ടുപ്പിളുകളുടെ ഒരു കൂട്ടത്തെയാണ് റിലേഷന്റെ ഇൻസ്റ്റൻസ് എന്നു പറയുന്നത്. നിരകളുടെ പേരും ഡാറ്റ തരവും സൂചിപ്പിക്കുന്ന 6 നിരകൾ STUDENT റിലേഷനിലെ ഓരോ വരിക്കും ഉണ്ടെന്ന് മേൽക്കാണുന്ന സ്കീമ വ്യക്തമാക്കുന്നു. STUDENT ബന്ധത്തിന്റെ ഉദാഹരണമാണ് പട്ടിക 8.1 ൽ കാണിച്ചിരിക്കുന്നത്.

നിങ്ങളുടെ പഠനപുരോഗതി അറിയുക

1. വരികളും നിരകളുമായി ഡാറ്റയെ ക്രമീകരിച്ചിരിക്കുന്നതിനെ __________ എന്ന് പറയുന്നു.
2. __________ ഒരു പട്ടികയിലെ ഒരു പ്രത്യേക എന്റിറ്റിയുടെ പൂർണമായ ഡാറ്റ നൽകുന്നു.
3. ഒരു റിലേഷനിലെ വരികളുടെ എണ്ണം __________ ആണ്.
4. ഒരു റിലേഷനിലെ __________ എണ്ണം റിലേഷന്റെ ഡിഗ്രി എന്നറിയപ്പെടുന്നു
5. റിലേഷണൽ മോഡലുകളിൽ ഡാറ്റ ക്രമീകരിച്ചിരിക്കുന്നത് __________ രീതിയിലാണ്.

8.6.1 കീകൾ (Keys)

ഒരു റിലേഷനെ ടൂപ്പിളുകളുടെ ഒരു ഗണമായി നിർവചിക്കുന്നു. അതായത് എല്ലാ നിരകളുടെയും മൂല്യങ്ങളുടെ സംയോജനം ഒന്നു തന്നെയാകുന്ന രണ്ടു ടൂപ്പിളുകൾ പാടില്ല. അതിനാൽ, ഒരു റിലേഷനിലെ ഓരോ ടൂപ്പിളും തിരിച്ചറിയുവാൻ ഒരു മാർഗം ഉണ്ടായിരിക്കണം. ഇത്തരം വ്യത്യാസങ്ങൾ സൃഷ്ടിക്കാൻ ഒരു കീയുടെ ആശയം നമ്മെ അനുവദിക്കുന്നു. ഒരു റിലേഷനിലെ ഓരോ ടൂപ്പിളും മറ്റ് ടൂപ്പിളുകളിൽ നിന്ന് അനന്യമായി വേർതിരിക്കുന്നതിനുള്ള ഒരു ആട്രിബ്യൂട്ടിന്റെയോ അല്ലെങ്കിൽ ഒരു കൂട്ടം ആട്രിബ്യൂട്ടുകളുടെയോ ശേഖരമാണ് കീ. ഒന്നിലധികം ആട്രിബ്യൂട്ടുകൾ ചേർന്നാണ് ഒരു കീ ഉണ്ടാകുന്നതെങ്കിൽ അതിനെ കോംപോസിറ്റ് (സംയുക്ത) കീ എന്നു പറയുന്നു. ഒരു റിലേഷനിലെ ഓരോ ടൂപ്പിളും വ്യത്യസ്തമായതിനാൽ അതിന്റെ മുഴുവൻ ആട്രിബ്യൂട്ടുകളും പരിഗണിച്ചാൽ അത് ഒരു കീ ആയിരിക്കും. എന്നിരുന്നാലും ചെറിയ കീകൾ ഉണ്ടെങ്കിൽ, പ്രായോഗിക കാരണങ്ങളാൽ, അവയെയാണ് പരിഗണിക്കുന്നത്.

a. കാൻഡിഡേറ്റ് കീ

ഒരു റിലേഷനിലെ ഒരു ട്യൂപ്പിളുകളെ അനന്യമായി തിരിച്ചറിയുന്ന ആട്രിബ്യൂട്ടുകളുടെ ചുരുങ്ങിയ ഗണമാണ് ഒരു കാൻഡിഡേറ്റ് കീ. പട്ടിക 8.1 ലെ STUDENT റിലേഷനിൽ, AdmNo ന് ഓരോ ട്യൂപ്പിളുകളും തിരിച്ചറിയുവാൻ കഴിയും. അതിനാൽ ഇത് ഒരു കാൻഡിഡേറ്റ് കീയായി കണക്കാക്കാം. ഒരു റിലേഷനിൽ ഒന്നിലധികം കാൻഡിഡേറ്റ് കീകൾ ഉണ്ടായിരിക്കാം. കൂടാതെ, ഒരു കാൻഡിഡേറ്റ് കീ ഒരു ആട്രിബ്യൂട്ട് മാത്രം ആകണമെന്നില്ല. അത് ഒരു സംയുക്ത കീയും ആകാം. ഉദാഹരണമായി, Roll, Batch, Year എന്നിവയുടെ സംയോജനവും ഒരു പ്രത്യേക വിദ്യാർഥിയെ തിരിച്ചറിയുവാൻ ഉപയോഗിക്കാം. അതിനാൽ, Roll + Batch + Year എന്നത് STUDENT റിലേഷന്റെ മറ്റൊരു കാൻഡിഡേറ്റ് കീയായി പരിഗണിക്കപ്പെടാം.

b. പ്രൈമറി കീ

ഡാറ്റാബേസ് സൃഷ്ടാവ് പട്ടികയിലെ ഒരു ട്യൂപ്പിൾ തിരിച്ചറിയാനായി തിരഞ്ഞെടുത്തിരിക്കുന്ന കാൻഡിഡേറ്റ് കീകളിൽ ഒന്നാണ് പ്രൈമറി കീ. ഒരു റിലേഷനിലെ വരികളെ തനതായി തിരിച്ചറിയാൻ കഴിയുന്ന ഒന്നോ അതിലധികമോ നിരകളുടെ ഒരു കൂട്ടമാണ് പ്രൈമറി കീ. ഓരോ എൻറിറ്റിയേയും തനതായി തിരിച്ചറിയുന്നതിനാൽ അത് ശൂന്യമോ, പകർപ്പോ ആകാൻ പാടില്ല.

പ്രൈമറി കീ സ്ഥാനത്തേക്കുള്ള സ്ഥാനാർഥികളായി കാൻഡിഡേറ്റ് കീകളെ കണക്കാക്കുന്നു. കാൻഡിഡേറ്റ് കീകളിൽ നിന്ന് ആട്രിബ്യൂട്ടുകളുടെ എണ്ണം ഏറ്റവും കുറവുള്ള സംയുക്തത്തെ പ്രൈമറി കീയായി തിരഞ്ഞെടുക്കാം. പട്ടിക 8.1 ൽ (STUDENT റിലേഷനിൽ) AdmNo പ്രൈമറി കീയായി ഉപയോഗിക്കാം. അതായത്, STUDENT റിലേഷനിലെ രണ്ട് വിദ്യാർഥികൾക്ക് ഒരേ AdmNo ഉണ്ടായിരിക്കില്ല. പട്ടിക 8.1 ൽ, Name ആട്രിബ്യൂട്ടിൽ തനതായ മൂല്യങ്ങൾ നമുക്ക് കാണാൻ സാധിക്കും. എന്നാൽ യഥാർഥ സാഹചര്യത്തിൽ വിദ്യാർഥികളിൽ കൂടുതൽ പേർക്ക് ഒരേ പേര് ഉണ്ടാകും.

c. ആൾട്ടർനേറ്റ് കീ

പ്രൈമറി കീ അല്ലാത്ത ഒരു കാൻഡിഡേറ്റ് കീയെ ആൾട്ടർനേറ്റ് കീ എന്നു വിളിക്കുന്നു. രണ്ടോ അതിലധികമോ കാൻഡിഡേറ്റ് കീകൾ ഉണ്ടെങ്കിൽ, അവയിലൊന്നിനു മാത്രമേ പ്രൈമറി കീയായി സജ്ജമാകാൻ കഴിയൂ. അവശേഷിക്കുന്ന ബാക്കി കീകളാണ് ആൾട്ടർനേറ്റ് കീ. തന്നിരിക്കുന്ന ഉദാഹരണത്തിൽ AdmNo പ്രൈമറി കീ ആയി എടുത്തിരിക്കുന്നതിനാൽ Roll + Batch + Year എന്ന സംയോജനമാണ് ആൾട്ടർനേറ്റ് കീ.

d. ഫോറിൻ കീ

ഒരു റിലേഷനിൽ മറ്റൊരു റിലേഷനിലെ പ്രൈമറി കീയുണ്ടെങ്കിൽ ആ കീയെ ഫോറിൻ കീ എന്നു വിളിക്കാം. രണ്ടോ അതിലധികമോ ടേബിളുകളുമായി ബന്ധപ്പെടുത്തുന്നതിനായി ഒരു ഫോറിൻ കീ ഉപയോഗിക്കുമ്പോൾ ഇത് റഫറൻസ് കീ എന്നും അറിയപ്പെടുന്നു. പട്ടിക 8.2 ൽ കാണിച്ചിരിക്കുന്നതുപോലെ Batch name നു പകരം Batch code ഉപയോഗിച്ചിട്ടുണ്ടെന്ന് കരുതുക. പട്ടിക 8.3ൽ കാണിച്ചിരിക്കുന്നതുപോലെ നമുക്ക് BATCH എന്ന ഒരു റിലേഷനുമുണ്ട്. BatchCode എന്നത് BATCH റിലേഷനിലെ പ്രൈമറി കീ ആയിരിക്കുമെന്നത് വ്യക്തമാണ്. എന്നാൽ ഇത് STUDENT പട്ടികയിൽ കീ അല്ലാത്ത നിരയായി ഉപയോഗിക്കുന്നു. അങ്ങനെ, BatchCode എന്നത് STUDENT റിലേഷന്റെ കാര്യത്തിൽ ഒരു ഫോറിൻ കീ ആയി പരാമർശിക്കപ്പെടുന്നു.

STUDENT relation

AdmNo	Roll	Name	BatchCode	Marks	Result
101	24	Sachin	S2	480	EHS
102	14	Rahul	C2	410	EHS
103	4	Fathima	H2	200	NHS
104	12	Mahesh	C2	180	NHS
105	24	Nelson	H2	385	EHS
106	8	Joseph	C2	350	EHS
107	24	Shaji	H2	205	NHS
108	2	Bincy	S2	300	EHS

പട്ടിക 8.2: പരിഷ്കരിച്ച STUDENT റിലേഷൻ

നിങ്ങളുടെ പഠനപുരോഗതി അറിയുക

1. ഒരു റിലേഷനിൽ ഒരു വരി അദ്വിതീയമായി തിരിച്ചറിയാൻ കഴിയുന്ന ആട്രിബ്യൂട്ട് __________ എന്നറിയപ്പെടുന്നു.
2. ഒരു റിലേഷനിൽ എത്ര പ്രൈമറി കീകൾ സാധ്യമാണ് ?
3. Employee എന്ന പട്ടികയിലെ കാൻഡിഡേറ്റ് കീകളാണ് Emp_code, Pan_no. Emp_code നെ പ്രൈമറി കീയായി നിശ്ചയിച്ചാൽ, Pan_no __________ ആയിരിക്കും
4. ഒരു കീയിൽ ഒന്നിൽ കൂടുതൽ ആട്രിബ്യൂട്ടുകളുണ്ടെങ്കിൽ അതിനെ __________ കീയെന്നുപറയുന്നു.
5. രണ്ട് റിലേഷനുകളെ ബന്ധിപ്പിക്കുന്നതിന് __________ കീ ഉപയോഗിക്കുന്നു.

BATCH relation

BatchCode	BatchName	Strength
S1	Science	150
S2	Science	150
C1	Commerce	100
C2	Commerce	100
H1	Humanities	100
H2	Humanities	100

പട്ടിക 8.3: BATCH റിലേഷൻ

8.7 റിലേഷണൽ ബീജഗണിതം (Relational algebra)

ഒരു ഡാറ്റാബേസിന്റെ ഘടനയും രൂപകല്പനയും നിർണയിക്കുന്നതിന് റിലേഷണൽ മോഡൽ പ്രദാനം ചെയ്യുന്ന ആശയങ്ങളെകുറിച്ച് നമ്മൾ ചർച്ചചെയ്തു. ഡാറ്റാബേസ് രൂപകല്പന ചെയ്ത് ഡാറ്റ സൂക്ഷിച്ചു കഴിഞ്ഞാൽ, ആവശ്യത്തിനനുസരിച്ച് വിവരം തിരിച്ചെടുക്കേണ്ടതാണ്. ഇതിനായി RDBMS വൈവിധ്യമാർന്ന പ്രവർത്തനങ്ങൾ കാഴ്ച വയ്ക്കുന്നു. ഒരു ഡാറ്റാബേസിലെ മുഴുവൻ റിലേഷനുകളെയും കൈകാര്യം ചെയ്യുന്ന പ്രവർത്തനങ്ങളുടെ ശേഖരം റിലേഷണൽ ബീജഗണിതം എന്നറിയപ്പെടുന്നു. ഈ പ്രവർത്തനങ്ങൾ റിലേഷണൽ മോഡലുമായി ബന്ധപ്പെട്ട ഒരു പ്രത്യേകതരം ഭാഷയായ അന്വേഷണ ഭാഷയുടെ (query language) സഹായത്തോടെ നടത്തപ്പെടുന്നു. ഈ പ്രവർത്തനങ്ങൾ നടത്തുന്നതിനുവേണ്ടി അടുത്ത അധ്യായത്തിൽ ആ ഭാഷ നമ്മൾ പഠിക്കും. റിലേഷണൽ ബീജഗണിതത്തിലെ വിവിധ പ്രവർത്തനങ്ങളിൽ ഒന്നോ രണ്ടോ റിലേഷനുകൾ ഇൻപുട്ട് ആയി സ്വീകരിച്ച് പുതിയ ഒരു റിലേഷൻ ഉണ്ടാക്കുന്നു. സെലക്ട് (SELECT), പ്രൊജക്റ്റ് (PROJECT), യൂണിയൻ (UNION), ഇന്റർസെക്ഷൻ (INTERSECTION), സെറ്റ് ഡിഫറൻസ് (SET DIFFERENCE), കാർട്ടീഷ്യൻ പ്രൊഡക്റ്റ് (CARTESIAN PRODUCT) തുടങ്ങിയവയാണ് റിലേഷണൽ ബീജഗണിതത്തിലെ അടിസ്ഥാനപരമായ പ്രവർത്തനങ്ങൾ. SELECT, PROJECT എന്നിവ ഒരു റിലേഷനിൽ മാത്രം പ്രവർത്തിക്കുന്നതിനാൽ അത് യൂനറി പ്രവർത്തനമാണ്. ബാക്കിയുള്ള പ്രവർത്തനങ്ങൾ രണ്ട് റിലേഷനുകളിൽ പ്രവർത്തിക്കുന്നതിനാൽ ബൈനറി പ്രവർത്തനങ്ങളാണ്.

8.7.1 സെലക്ട് പ്രവർത്തനം (SELECT operation)

നൽകിയിരിക്കുന്ന ഒരു ഉപാധി അനുസരിക്കുന്ന വരികൾ ഒരു റിലേഷനിൽ നിന്ന് തിരഞ്ഞെടുക്കുന്നതിന് സെലക്ട് പ്രവർത്തനം ഉപയോഗിക്കുന്നു. ഉപയോക്താവിന്റെ ഇഷ്ടാനുസരണം നിശ്ചിത വരികൾ തിരഞ്ഞെടുക്കാനുള്ള ഒരു ഉപയോക്തൃ നിർദിഷ്ട വ്യവസ്ഥയാണ് ഉപാധി (predicate). സിഗ്മ (σ) എന്ന അക്ഷരം ഉപയോഗിച്ച് ഈ പ്രവർത്തനം സൂചിപ്പിക്കുന്നു. സെലക്ടിന്റെ പൊതുവായ മാതൃക താഴെ കൊടുക്കുന്നു.

$$\sigma_{\text{condition}} (\text{Relation})$$

SELECT പ്രവർത്തനത്തിന്റെ ഫലമായുണ്ടാകുന്ന റിലേഷനിൽ തന്നിട്ടുള്ള വ്യവസ്ഥകൾ പാലിക്കപ്പെടുന്ന എല്ലാ വരികളും അടങ്ങിയിരിക്കുന്നു. റിലേഷണൽ ബീജഗണിതത്തിൽ താഴെപ്പറയുന്ന താരതമ്യ ഓപ്പറേറ്ററുകൾ ഉപയോഗിക്കുന്നു. < (കുറവ്), <= (കുറവോ അല്ലെങ്കിൽ തുല്യമോ), > (കൂടുതൽ), >= (കൂടുതലോ അല്ലെങ്കിൽ തുല്യമോ), = (തുല്യമാണ്), <> (തുല്യമല്ല) എന്നീ താരതമ്യ ഓപ്പറേറ്ററുകൾ ലളിതമായ വ്യവസ്ഥകൾ സജ്ജമാക്കുന്നതിന് ഉപയോഗിക്കുന്നു. അതുപോലെ ലോജിക്കൽ ഓപ്പറേറ്ററുകളായ ∨ (OR), ^ (AND), ! (NOT) എന്നിവ സമ്മിശ്ര വ്യവസ്ഥകൾ നിർമ്മിക്കാനും ഉപയോഗിക്കുന്നു,

SELECT പ്രവർത്തനം വിശദീകരിക്കുന്നതിന്, പട്ടിക 8.1 ൽ നൽകിയിട്ടുള്ള STUDENT റിലേഷൻ പരിഗണിക്കുക. എങ്ങനെയാണ് SELECT പ്രവർത്തനം റിലേഷണൽ ബീജഗണിതത്തിൽ ഉപയോഗിക്കുന്നതെന്നും അത് എന്ത് ഫലമാണ് ഉണ്ടാക്കുന്നതെന്നും താഴെ കൊടുത്തിരിക്കുന്ന ഉദാഹരണങ്ങൾ വ്യക്തമാക്കുന്നു

ഉദാഹരണം 8.1: ഉന്നതവിദ്യാഭ്യാസത്തിന് യോഗ്യതയുള്ള എല്ലാ വിദ്യാർഥികളെയും തിരഞ്ഞെടുക്കുക.

$$\sigma_{\text{Result="EHS"}}\ (\text{STUDENT})$$

പട്ടിക 8.4 ൽ കാണിച്ചിരിക്കുന്നതുപോലെയുള്ള ഒരു റിലേഷനാണ് ഈ പ്രവർത്തനത്തിന്റെ ഫലം.

AdmNo	Roll	Name	Batch	Marks	Result
101	24	Sachin	Science	480	EHS
102	14	Rahul	Commerce	410	EHS
105	24	Nelson	Humanities	385	EHS
106	8	Joseph	Commerce	350	EHS
108	2	Bincy	Science	300	EHS

പട്ടിക 8.4: ഉദാഹരണം 8.1 ന്റെ ഔട്ട്പുട്ട്

ഉദാഹരണം 8.2: കോമേഴ്സ് ബാച്ചിലെ പരാജയപ്പെട്ട വിദ്യാർഥികളെ തിരഞ്ഞെടുക്കുക

$$\sigma_{\text{Result="NHS"} \wedge \text{Batch="Commerce"}}\ (\text{STUDENT})$$

പട്ടിക 8.5 ൽ കാണിച്ചിരിക്കുന്നതു പോലെയുള്ള ഒരു റിലേഷനാണ് ഈ പ്രവർത്തനത്തിന്റെ ഫലം.

AdmNo	Roll	Name	Batch	Marks	Result
104	12	Mahesh	Commerce	180	NHS

പട്ടിക 8.5: ഉദാഹരണം 8.2 ന്റെ ഔട്ട്പുട്ട്

ഉദാഹരണം 8.3: സയൻസ് അല്ലെങ്കിൽ കൊമേഴ്സിലുള്ള എല്ലാ വിദ്യാർഥികളെയും തിരഞ്ഞെടുക്കുക.

$$\sigma_{\text{Batch="Science"} \vee \text{Batch="Commerce"}}\ (\text{STUDENT})$$

പട്ടിക 8.6 ലെ ഔട്ട്പുട്ട് ഈ ഓപ്പറേഷൻ കാണിക്കുന്നു

AdmNo	Roll	Name	Batch	Marks	Result
101	24	Sachin	Science	480	EHS
102	14	Rahul	Commerce	410	EHS
104	12	Mahesh	Commerce	180	NHS
106	8	Joseph	Commerce	350	EHS
108	2	Bincy	Science	300	EHS

പട്ടിക 8.6: ഉദാഹരണം 8.3 ന്റെ ഔട്ട്പുട്ട്

8.7.2 പ്രൊജക്റ്റ് പ്രവർത്തനം (PROJECT operation)

PROJECT പ്രവർത്തനം റിലേഷനിൽ നിന്ന് ചില നിരകൾ തിരഞ്ഞെടുക്കുകയും ഒരു പുതിയ റിലേഷൻ രൂപപ്പെടുത്തുകയും ചെയ്യുന്നു. ഏതെങ്കിലും പ്രവർത്തനത്തിന് ഒരു റിലേഷനിലെ ഏതാനും നിരകൾ മാത്രം തിരഞ്ഞെടുക്കണമെങ്കിൽ PROJECT പ്രവർത്തനം ഉപയോഗിക്കാം. പൈ (π) എന്ന അക്ഷരം ഉപയോഗിച്ച് ഈ പ്രവർത്തനം സൂചിപ്പിക്കുന്നു. PROJECT പ്രവർത്തനത്തിന്റെ പൊതുവായ മാതൃക താഴെ പറയുന്നു.

$$\pi_{A1, A2, \ldots, An} \text{(Relation)}$$

ഇവിടെ A1, A2,. . . ., എന്നത് തന്നിരിക്കുന്ന റിലേഷനുകളിലെ നിരകളാകുന്നു.

ഉദാഹരണം 8.4: STUDENT റിലേഷനിലെ പേര്, റിസൽട്ട്, മാർക്ക് എന്നീ നിരകൾ തിരഞ്ഞെടുക്കുക.

$$\pi_{\text{Name, Marks, Result}} \text{(STUDENT)}$$

ഈ പ്രക്രിയയുടെ ഫലം പട്ടിക 8.7 ൽ നൽകിയിരിക്കുന്നു

SELECT, PROJECT എന്നീ പ്രവർത്തനങ്ങളെ ഒരൊറ്റ പ്രസ്താവനയിൽ സംയോജിപ്പിക്കുവാൻ സാധിക്കും. ഇതിന്റെ വിശദീകരണം ഉദാഹരണം 8.5 ലും 8.6 ലും കൊടുത്തിട്ടുണ്ട്

Name	Marks	Result
Sachin	480	EHS
Rahul	410	EHS
Fathima	200	NHS
Mahesh	180	NHS
Nelson	385	EHS
Joseph	350	EHS
Shaji	205	NHS
Bincy	300	EHS

പട്ടിക 8.7: ഉദാഹരണം 8.4 ന്റെ ഔട്ട്പുട്ട്

ഉദാഹരണം 8.5: ഉന്നത പഠനത്തിനു യോഗ്യരായ വിദ്യാർഥികളുടെ അഡ്മിഷൻ നമ്പരും പേരും തിരഞ്ഞെടുക്കുക.

$$\pi_{\text{AdmNo, Name}} (\sigma_{\text{result="EHS"}} \text{(STUDENT)})$$

ഈ പ്രക്രിയയുടെ ഫലമായുണ്ടായ റിലേഷൻ പട്ടിക 8.8 ൽ കൊടുത്തിട്ടുണ്ട്. ഫലത്തിന്റെ കൃത്യത പരിശോധിച്ചുറപ്പിക്കുന്നതിനു പട്ടിക 8.4 ഉപയോഗിച്ച് താരതമ്യം ചെയ്യുക.

AdmNo	Name
101	Sachin
102	Rahul
105	Nelson
106	Joseph
108	Bincy

പട്ടിക 8.8: ഉദാഹരണം 8.5 ന്റെ ഔട്ട്പുട്ട്

ഉദാഹരണം 8.6: ഉന്നതവിദ്യാഭ്യാസത്തിന് യോഗ്യരല്ലാത്ത ഹ്യുമാനിറ്റീസ് ബാച്ചിലെ വിദ്യാർഥികളുടെ പേരും മാർക്കുകളും തിരഞ്ഞെടുക്കുക.

$$\pi_{\text{Name, Marks}} (\sigma_{\text{result="NHS"} \wedge \text{Batch="Humanities"}} \text{(STUDENT)})$$

കൂട്ടിയിണക്കിയ ഈ പ്രവർത്തനങ്ങളുടെ ഫലം പട്ടിക 8.9 ൽ കാണിച്ചിരിക്കുന്നു.

Name	Marks
Fathima	200
Shaji	205

പട്ടിക 8.9: ഉദാഹരണം 8.6 ന്റെ ഔട്ട്പുട്ട്

8.7.3 യൂണിയൻ പ്രവർത്തനം (UNION operation)

UNION പ്രവർത്തനം ഒരു ബൈനറി ഓപ്പറേഷനാണ്. അത് രണ്ട് നിർദിഷ്ട റിലേഷനുകളിലെ എല്ലാ വരികളും അടങ്ങുന്ന ഒരു പുതിയ റിലേഷൻ തരുന്നു. '∪' എന്ന ചിഹ്നം കൊണ്ട് UNION പ്രവർത്തനം സൂചിപ്പിക്കുന്നു. രണ്ടു റിലേഷനുകളും യൂണിയൻ അനുരൂപമായിരിക്കുകയും ഫലത്തിന്റെ സ്കീമ ആദ്യ റിലേഷന്റെ സ്കീമയ്ക്ക് സമാനമായിരിക്കുകയും ചെയ്യും. രണ്ട് റിലേഷനുകൾ യൂണിയൻ അനുരൂപമാണെങ്കിൽ, അവയ്ക്ക് ഒരേ എണ്ണം നിരകളും, സമാനമായ നിരകൾക്ക് സമാന ഡൊമെയ്നുമായിരിക്കണം. യൂണിയൻ അനുരൂപത നിർവചിക്കുന്നതിന് നിരകളുടെ പേരുകൾ പരിഗണിക്കാറില്ല എന്നത് ശ്രദ്ധിക്കുക.

പട്ടിക 8.10 ലും 8.11 ലും നൽകിയിരിക്കുന്ന ARTS, SPORTS എന്നീ രണ്ട് റിലേഷനുകൾ പരിഗണിക്കുക. ഇവയിൽ യഥാക്രമം കലാമേളയിൽ പങ്കെടുക്കുന്ന വിദ്യാർഥികളുടെ വിശദാംശങ്ങളും കായികമേളയിൽ പങ്കെടുക്കുന്ന വിദ്യാർഥികളുടെ വിശദാംശങ്ങളും അടങ്ങിയിരിക്കുന്നു. AdmNo, Name, BatchCode എന്നിവ ARTS, SPORTS എന്നീ റിലേഷനുകളിലെ നിരകളാണ്. ഈ രണ്ട് റിലേഷനുകളും യൂണിയൻ അനുരൂപമാണെന്നത് വ്യക്തമാണ്. അതായത്, ഈ രണ്ട് റിലേഷനുകളിലും ഒരേ എണ്ണം നിരകൾ ഉണ്ട്, സദൃശമായ നിരകളുടെ തരവും ഒരുപോലെയാണ്.

ARTS relation

AdmNo	Name	BatchCode
101	Sachin	S2
103	Fathima	H2
106	Joseph	C2
110	Nikitha	S1
132	Vivek	C1
154	Nevin	C1

പട്ടിക 8.10: ARTS

SPORTS relation

AdmNo	Name	BatchCode
102	Rahul	C2
103	Fathima	H2
105	Nelson	H2
106	Joseph	C2
108	Bincy	S2
132	Vivek	C1
164	Rachana	S1

പട്ടിക 8.11: SPORTS

ARTS ∪ SPORTS എന്ന റിലേഷൻ കലയിലോ, സ്പോർട്സിലോ അല്ലെങ്കിൽ രണ്ടിലുമോ പങ്കെടുക്കുന്ന വിദ്യാർഥികളുടെ വിശദാംശങ്ങൾ നൽകുന്നു. അതായത്, ARTS ∪ SPORTS പട്ടിക 8.12 ൽ കാണിച്ചിരിക്കുന്നതുപോലെയുള്ള ഒരു റിലേഷൻ നൽകുന്നു. ഈ ടേബിളിൽ ARTS അല്ലെങ്കിൽ SPORTS അല്ലെങ്കിൽ രണ്ടിലുമുള്ള റെക്കോർഡുകൾ ഉൾപ്പെടുന്നു. അതേ സമയം തനിപ്പകർപ്പ് ഒഴിവാക്കുകയും ചെയ്യുന്നു. പട്ടിക 8.12 ൽ 103, 106 എന്നീ അഡ്മിഷൻ നമ്പറുകളിലുള്ള വിദ്യാർഥികളുടെ രേഖകൾ ഒരിക്കൽ മാത്രമേ ലഭ്യമാകൂ.

AdmNo	Name	BatchCode
101	Sachin	S2
103	Fathima	H2
106	Joseph	C2
110	Nikitha	S1
132	Vivek	C1
154	Nevin	C1
102	Rahul	C2
105	Nelson	H2
108	Bincy	S2
164	Rachana	S1

പട്ടിക 8.12: ARTS ∪ SPORTS

8.7.4 ഇന്റർസെക്ഷൻ പ്രവർത്തനം (INTERSECTION operation)

INTERSECTION പ്രവർത്തനവും ഒരു ബൈനറി പ്രവർത്തനമാണ്. അത്, രണ്ട് നിർദിഷ്ട റിലേഷനുകളിലെ പൊതുവായ എല്ലാ വരികളും അടങ്ങുന്ന ഒരു പുതിയ റിലേഷൻ തരുന്നു. ∩ എന്നചിഹ്നം കൊണ്ട് INTERSECTION പ്രവർത്തനം സൂചിപ്പിക്കുന്നു. രണ്ടു റിലേഷനുകളും യൂണിയൻ അനുരൂപമായിരിക്കുകയും ഫലത്തിന്റെ സ്കീമ ആദ്യ റിലേഷന്റെ സ്കീമയ്ക്ക് സമാനമായിരിക്കുകയും ചെയ്യും. രണ്ട് റിലേഷനുകൾ യൂണിയൻ അനുരൂപമാണെങ്കിൽ, അവയ്ക്ക് ഒരേ എണ്ണം നിരകളും, സമാനമായ നിരകൾക്ക് സമാന ഡൊമെയ്നുമായിരിക്കണം. യൂണിയൻ അനുരൂപത നിർവചിക്കുന്നതിന് നിരകളുടെ പേരുകൾ പരിഗണിക്കാറില്ല എന്നത് ശ്രദ്ധിക്കുക.

നാം 8.10, 8.11 എന്നീ റിലേഷനുകളിൽ INTERSECT പ്രവർത്തനം നടത്തുകയാണെങ്കിൽ, ആർട്സിലും സ്പോർട്സിലും പൊതുവായി പങ്കെടുക്കുന്ന വിദ്യാർഥികളുടെ വിശദാംശങ്ങൾ ലഭിക്കും. അതായത് ARTS ∩ SPORTS റിലേഷനിൽ ARTS, SPORTS എന്നിവ രണ്ടിലുമുള്ള വരികൾ മാത്രമേ ഉണ്ടാകൂ (പട്ടിക 8.13)

AdmNo	Name	BatchCode
103	Fathima	H2
106	Joseph	C2
132	Vivek	C1

പട്ടിക 8.13: ARTS ∩ SPORTS

8.7.5 സെറ്റ് ഡിഫറൻസ് പ്രവർത്തനം (SET DIFFERENCE operation)

SET DIFFERENCE പ്രവർത്തനവും ഒരു ബൈനറി പ്രവർത്തനമാണ്. അത്, ആദ്യ റിലേഷനിൽ ഉള്ളതും രണ്ടാമത്തെ റിലേഷനിൽ ഇല്ലാത്തതുമായ എല്ലാ വരികളും അടങ്ങുന്ന ഒരു പുതിയ റിലേഷൻ തരുന്നു. '–' എന്നചിഹ്നം കൊണ്ട് SET DIFFERENCE, പ്രവർത്തനം സൂചിപ്പിക്കുന്നു. ഇവിടെയും രണ്ടു റിലേഷനുകളും യൂണിയൻ അനുരൂപമായിരിക്കണം.

AdmNo	Name	BatchCode
101	Sachin	S2
110	Nikitha	S1
154	Nevin	C1

പട്ടിക 8.14: ARTS - SPORTS

AdmNo	Name	BatchCode
102	Rahul	C2
105	Nelson	H2
108	Bincy	S2
164	Rachana	S1

പട്ടിക 8.15: SPORTS - ARTS

ARTS – SPORTS റിലേഷനിൽ, ആർട്സിൽ പങ്കെടുക്കുകയും എന്നാൽ സ്പോർട്സിൽ പങ്കെടുക്കാത്തതുമായ വിദ്യാർഥികളുടെ വിശദാംശങ്ങൾ ലഭിക്കും. അതായത് ARTS – SPORTS റിലേഷനിൽ പട്ടിക 8.14 ൽ കാണിച്ചിരിക്കുന്നതുപോലെ ARTS ൽ ഉള്ളതും, എന്നാൽ SPORTS ൽ ഇല്ലാത്തതുമായ വരികൾ ഉണ്ടാകും. അതുപോലെ SPORTS – ARTS റിലേഷനിൽ പട്ടിക 8.15 ൽ കാണിച്ചിരിക്കുന്നതു പോലെ SPORTS ൽ ഉള്ളതും, എന്നാൽ ARTS ൽ ഇല്ലാത്തതുമായ വരികളും ഉണ്ടാകും.

8.7.6 കാർട്ടീഷ്യൻ പ്രൊഡക്ട് പ്രവർത്തനം (CARTESIAN PRODUCT operation)

രണ്ട് റിലേഷനുകളിൽ നിന്നുള്ള വരികളുടെ സാധ്യമായ എല്ലാ കൂടിച്ചേരലുകളും ഉൾക്കൊള്ളുന്ന ഒരു റിലേഷൻ കാർട്ടീഷ്യൻ പ്രൊഡക്ട് നൽകുന്നു. പുതിയ റിലേഷന്റെ ഡിഗ്രി

(ആട്രിബ്യൂട്ടുകളുടെ എണ്ണം) രണ്ട് റിലേഷനുകളുടെയും ആകെ ഡിഗ്രിക്ക് തുല്യമാണ്. ഇതിന്റെ കാർഡിനാലിറ്റി (വരികളുടെ എണ്ണം) രണ്ട് റിലേഷനിലുമുള്ള വരികളുടെ എണ്ണത്തിന്റെ ഗുണനഫലമാണ്. കാർട്ടീഷ്യൻ പ്രൊഡക്ട് സൂചിപ്പിക്കുന്നതിന് 'x' (ക്രോസ്) ചിഹ്നം ഉപയോഗിക്കുന്നു. ഇത് ക്രോസ് പ്രൊഡക്ട് എന്നും അറിയപ്പെടുന്നു. ആദ്യത്തെ റിലേഷനിലെ എല്ലാ വരികളും രണ്ടാമത്തെ റിലേഷന്റെ വരികളുമായി കൂടിച്ചേർന്ന് പുതിയ ബന്ധത്തിന്റെ വരികൾ രൂപപ്പെടുത്തുന്നു.

TEACHER relation

TeacherId	Name	Dept
1001	Viswesaran	English
1002	Meenakshi	Computer

പട്ടിക 8.16: *TEACHER റിലേഷൻ*

പട്ടിക 8.16ൽ കാണിച്ചിരിക്കുന്ന TEACHER റിലേഷൻ നമുക്ക് പരിഗണിക്കാം. പട്ടിക 8.2 ലെ STUDENT റിലേഷനുമായുള്ള ഈ റിലേഷന്റെ കാർട്ടീഷ്യൻ പ്രൊഡക്ട് പ്രവർത്തനത്തിന്റെ ഫലം പട്ടിക 8.19 ൽ കാണിച്ചിരിക്കുന്നു. STUDENT റിലേഷനിലെ ഓരോ വരിയും TEACHER റിലേഷനിലെ വരികളുമായി കൂടിച്ചേർന്നതായി ഈ പട്ടിക കാണിക്കുന്നു.

Adm No	Roll	Name	Batch Code	Marks	Result	TeacherId	Name	Dept
101	24	Sachin	S2	480	EHS	1001	Viswesaran	English
101	24	Sachin	S2	480	EHS	1002	Meenakshi	Computer
102	14	Rahul	C2	410	EHS	1001	Viswesaran	English
102	14	Rahul	C2	410	EHS	1002	Meenakshi	Computer
103	4	Fathima	H2	200	NHS	1001	Viswesaran	English
103	4	Fathima	H2	200	NHS	1002	Meenakshi	Computer
104	12	Mahesh	C2	180	NHS	1001	Viswesaran	English
104	12	Mahesh	C2	180	NHS	1002	Meenakshi	Computer
105	24	Nelson	H2	385	EHS	1001	Viswesaran	English
105	24	Nelson	H2	385	EHS	1002	Meenakshi	Computer
106	8	Joseph	C2	350	EHS	1001	Viswesaran	English
106	8	Joseph	C2	350	EHS	1002	Meenakshi	Computer
107	24	Shaji	H2	205	NHS	1001	Viswesaran	English
107	24	Shaji	H2	205	NHS	1002	Meenakshi	Computer
108	2	Bincy	S2	300	EHS	1001	Viswesaran	English
108	2	Bincy	S2	300	EHS	1002	Meenakshi	Computer

പട്ടിക 8.19: *STUDENT X TEACHER*

ഒരു ഡാറ്റാബേസ് മോഡൽ ഡാറ്റയുടെ ലോജിക്കൽ രൂപരേഖ നിർവചിക്കുന്നു. ഡാറ്റയുടെ വിവിധ ഘടകങ്ങൾ തമ്മിലുള്ള ബന്ധത്തെ ഈ മോഡൽ വിശദീകരിക്കുന്നു. ഹൈറാർക്കിയൽ മോഡൽ, നെറ്റ്‌വർക്ക് മോഡൽ, റിലേഷണൽ മോഡൽ, ഒബ്ജക്റ്റ് ഓറിയന്റഡ് മോഡൽ എന്നീ വിവിധങ്ങളായ മോഡലുകൾ ഉപയോഗിച്ചു വരുന്നു. പഴയകാല മെയിൻഫ്രെയിം ഡാറ്റാബേസ് നിർവഹണ സംവിധാനമായ IBM ന്റെ ഇൻഫർമേഷൻ മാനേജ്മെന്റ് സിസ്റ്റം (IMS) പോലെയുള്ള ഹൈറാർക്കിയൽ മോഡൽ ഉപയോഗിച്ചിരുന്നു. സാധാരണയായി ഉപയോഗിച്ചിരുന്ന നെറ്റ്‌വർക്ക് മോഡലാണ് സിൽക്കോം സിസ്റ്റത്തിന്റെ ടോട്ടലും കുള്ളിനെറ്റിന്റെ IDMS ഉം.

നമുക്ക് സംഗ്രഹിക്കാം

DBMSലെ അടിസ്ഥാന ആശയങ്ങളെക്കുറിച്ചും അതിന്റെ ഘടകങ്ങളെക്കുറിച്ചും നമ്മൾ ചർച്ച ചെയ്തു. പരമ്പരാഗത ഫയൽ സംവിധാനത്തെ അപേക്ഷിച്ച് DBMSനുള്ള മേന്മകളും മനസ്സിലാക്കി. റിലേഷണൽ ഡാറ്റാ മോഡലിന്റെ പശ്ചാത്തലത്തിൽ ഡാറ്റാബേസുമായി ബന്ധപ്പെട്ട വിവിധ പദാവലികളെ കുറിച്ചുള്ള ഒരു സംക്ഷിപ്ത രൂപം പ്രതിപാദിച്ചു. ഡാറ്റാബേസിൽ ഒരിക്കൽ ഡാറ്റ വ്യവസ്ഥാപിതമായി ക്രമീകരിച്ചു കഴിഞ്ഞാൽ, ആവശ്യമുള്ള വിവരങ്ങൾ നിർമ്മിക്കുന്നതിനായി റിലേഷണൽ ബീജഗണിതത്തിലെ പ്രവർത്തനങ്ങൾ എങ്ങനെ ഉപയോഗിക്കാമെന്ന് ഉദാഹരണത്തിന്റെ സഹായത്തോടെ നമ്മൾ പഠിച്ചു.

ഈ അധ്യായത്തിൽ അവതരിപ്പിച്ച ആശയങ്ങളെക്കുറിച്ചുള്ള നല്ല ഒരു ധാരണ അടുത്ത അധ്യായം ഫലപ്രദമായി പഠിക്കുന്നതിന് അത്യാവശ്യമാണ്. ക്വറി ഭാഷ ഉപയോഗിച്ച് ഡാറ്റാബേസുകൾ നിർമ്മിക്കുന്നതും ശേഖരിച്ച വിവരങ്ങൾ കൈകാര്യം ചെയ്യുന്ന രീതിയും അടുത്ത അധ്യായത്തിൽ നമ്മൾ ചർച്ച ചെയ്യും.

നമുക്ക് വിലയിരുത്താം

1. ഡാറ്റാബേസുമായി ബന്ധപ്പെട്ട പ്രവർത്തനങ്ങൾ കൈകാര്യം ചെയ്യുന്നതും നിയന്ത്രിക്കുന്നതും ആരാണ്?

   a. ഡാറ്റാബേസ് അഡ്മിനിസ്ട്രേറ്റർ b. പ്രോഗ്രാമർ

   c. സാങ്കേതിക പരിജ്ഞാനമുള്ള ഉപയോക്താവ് d. സാധാരണ ഉപയോക്താവ്

2. റിലേഷണൽ മോഡലിൽ, കാർഡിനാലിറ്റി എന്നു കൊണ്ട് ഉദ്ദേശിക്കുന്നത് എന്താണ്?

   a. ടൂപ്പിളുകളുടെ എണ്ണം b. ആട്രിബ്യൂട്ടുകളുടെ എണ്ണം

   c. ടേബിളുകളുടെ എണ്ണം d. കൺസ്ട്രയ്ന്റുകളുടെ എണ്ണം

3. റിലേഷണൽ ബീജഗണിതത്തിലെ കാർട്ടീഷ്യൻ പ്രൊഡക്ട് എന്നത്,

   a. ഒരു യൂനറി ഓപ്പറേറ്റർ
   b. ഒരു ബൈനറി ഓപ്പറേറ്റർ
   c. ഒരു ടർണറി ഓപ്പറേറ്റർ
   d. നിർവചിക്കപ്പെട്ടിട്ടില്ല

4. ഡാറ്റാബേസ് സംഗ്രഹം എത്ര തലങ്ങളാണ്?

   a. ഒരു തലം
   b. രണ്ട് തലങ്ങൾ
   b. മൂന്നു തലങ്ങൾ
   d. നാലു തലങ്ങൾ

5. ഒരു റിലേഷണൽ മോഡലിൽ, റിലേഷൻ എന്ന് പറയുന്നത്

   a. ടൂപിളുകൾ
   b. ആട്രിബ്യൂട്ടുകൾ
   c. ടേബിളുകൾ
   d. വരികൾ

6. ഒരു ഡാറ്റാബേസ് സംഗ്രഹത്തിന്റെ ബാഹ്യതലം എന്നത്

   a. ഭൗതിക തലം
   b. ലോജിക്കൽ തലം
   c. കൺസെപ്ചൽ തലം
   d. വ്യൂ തലം

7. ഒരു ഡാറ്റാബേസിലെ ബന്ധപ്പെട്ട ഫീൽഡുകളുടെ കൂട്ടം അറിയപ്പെടുന്നത്.

   a. ഡാറ്റ ഫയൽ
   b. ഡാറ്റ റെക്കോർഡ്
   c. മെനു
   d. ബാങ്ക്

8. ഒരു ഡാറ്റാബേസ് ഡെവലപ്പർ ഒരു റെക്കോർഡ് എന്നത് കൊണ്ട് ഉദ്ദേശിക്കുന്നത്.

   a. മാനദണ്ഡം
   b. റിലേഷൻ
   c. ടൂപിൾ
   d. ആട്രിബ്യൂട്ട്

9. ഡാറ്റാബേസ് മാനേജ്മെന്റ് സമീപനത്തിന്റെ ഒരു ഗുണം

   a. ഡാറ്റ പ്രോഗ്രാമുകളെ ആശ്രയിച്ചിരിക്കുന്നു
   b. ഡാറ്റാ റിഡൻഡൻസി വർധിക്കുന്നു
   c. ഡാറ്റ സംയോജിപ്പിച്ച് ഒന്നിലധികം പ്രോഗ്രാമുകൾക്ക് ഉപയോഗിക്കുവാൻ കഴിയും
   d. ഇവ ഒന്നുമല്ല

10. ഡാറ്റ സ്വാതന്ത്ര്യം എന്നാൽ എന്താണ് ?

   a. ഡാറ്റ വെവ്വേറെ നിർവചിക്കുകയും പ്രോഗ്രാമുകളിൽ ഉൾപ്പെടുത്താതിരിക്കുകയും ചെയ്യുക.
   b. പ്രോഗ്രാമുകൾ ഡാറ്റയുടെ ഭൗതിക ആട്രിബ്യൂട്ടുകളെ ആശ്രയിക്കുന്നില്ല.
   c. പ്രോഗ്രാമുകൾ ഡാറ്റയുടെ ലോജിക്കൽ ആട്രിബ്യൂട്ടുകളെ ആശ്രയിക്കുന്നില്ല.
   d. (ബി) യും (സി) യും

11. റിലേഷനുകൾ തമ്മിലുള്ള ബന്ധം പ്രതിനിധാനം ചെയ്യുന്ന കീയെ എന്ത് വിളിക്കുന്നു?

a. പ്രൈമറി കീ
b. കാൻഡിഡേറ്റ് കീ
c. ഫോറിൻ കീ
d. ആൾട്ടർനേറ്റ് കീ

12. ടേബിളുകളിലെ ചില നിരകളിൽ മാത്രം മതിയെങ്കിൽ താഴെ പറയുന്ന ഏത് പ്രവർത്തനമാണ് ഉപയോഗിക്കുന്നത്?

a. PROJECTION
b. SELECTION
c. UNION
d. SELECT

13. താഴെ പറയുന്ന ഏത് പ്രവർത്തനത്തിനാണ് റിലേഷനുകൾ യൂണിയൻ അനുരൂപമാകേണ്ടത്.

a. UNION
b. INTERSECTION
c. SET DIFFERENCE
d. മുകളിൽ പറഞ്ഞവയെല്ലാം

14. ഏത് ഡാറ്റാബേസ് തലമാണ് ഉപയോക്താക്കൾക്ക് ഏറ്റവും അടുത്തുള്ളത്?

a. ബാഹ്യതലം
b. ആന്തരികതലം
c. ഭൗതികതലം
d. ലോജിക്കൽതലം

15. R1, R2 എന്നീ റിലേഷനുകളുടെ UNION ഓപ്പറേഷൻ നടത്തിയാൽ, അതിൽ

a. R1 ന്റെ എല്ലാ ടൂപ്പിളുകളും ഉൾപ്പെടും.
b. R2 ന്റെ എല്ലാ ടൂപ്പിളുകളും ഉൾപ്പെടും.
c. R1, R2 എന്നിവയിലുള്ള എല്ലാ ടൂപ്പിളുകളും ഉൾപ്പെടും.
d. R1, R2 എന്നിവയിലെ പൊതുവായ എല്ലാ ടൂപ്പിളുകളും ഉൾപ്പെടും.

16. ഒരു ഫയലിൽ നിന്ന് ചില റെക്കോർഡുകൾ ശേഖരിക്കുന്നതിനുള്ള ഒരു ഫയൽ മാനിപ്പുലേഷൻ പ്രവർത്തനം.

a. SELECT
b. PROJECT
c. JOIN
d. PRODUCT

17. R (A, B, C) എന്ന റിലേഷണൽ ഘടനയിൽ A യുടെ മൂല്യത്തിൽ NULL മൂല്യവും ഉൾപ്പെടുന്നു. താഴെ പറയുന്ന പ്രസ്താവനയിൽ ശരിയായിട്ടുള്ളത് ഏത്?

a. A ഒരു കാൻഡിഡേറ്റ് കീയാണ്
b. A ഒരു കാൻഡിഡേറ്റ് കീ അല്ല
c. A ഒരു പ്രൈമറി കീയാണ്
d. (a) യും (c) യും

18. കാർഡിനാലിറ്റി 22 ഉള്ള ഒരു റിലേഷനിൽ എത്ര ടൂപ്പിളുകൾ ഉണ്ട്?

a. 22 b. 11

c. 1 d. ഇവയൊന്നുമല്ല

19. ഒരു റിലേഷനിലെ നിരയുടെ സാധ്യമായ മൂല്യങ്ങളെ എന്ത് വിളിക്കുന്നു?

a. ആട്രിബ്യൂട്ട് b. ഡിഗ്രി

c. ടൂപ്പിൾ d. ഡൊമെയ്ൻ

20. പരമ്പരാഗത ഫയലുകളിൽ ഡാറ്റ സംഭരിക്കുന്നതിനുപകരം ഒരു ഡാറ്റാബേസ് സംവിധാനം നിങ്ങൾ തിരഞ്ഞെടുക്കുന്നത് എന്തുകൊണ്ട്?

21. DBMS ലെ വിവിധ തലത്തിലുള്ള ഡാറ്റാസംഗ്രഹങ്ങൾ വിശദീകരിക്കുക?

22. ലോജിക്കൽ, ഭൗതികം എന്നീ ഡാറ്റാസ്വാതന്ത്ര്യത്തിന്റെ ആശയങ്ങളുമായി സ്കീമയുടെ വിവിധ തലങ്ങൾ എങ്ങനെ ബന്ധപ്പെട്ടിരിക്കുന്നു?

23. താഴെ തന്നിരിക്കുന്ന EMPLOYEE റിലേഷന്റെ ആട്രിബ്യൂട്ടുകൾ, ഡിഗ്രി, കാർഡിനാലിറ്റി, Name ന്റെ ഡൊമെയ്ൻ, Emp_Code ന്റെ ഡൊമെയ്ൻ എന്നിവ എഴുതുക.

Emp_Code	Name	Department	Designation	Salary
1000	Sudheesh	Purchase	Manager	25000
1001	Dhanya	Sales	Manager	25000
1002	Fathima	Marketing	Clerk	12000
1003	Shajan	Sales	Clerk	13000

24. ചോദ്യം 23ൽ തന്നിരിക്കുന്ന EMPLOYEE റിലേഷന്റെ പ്രൈമറി കീ, കാൻഡിഡേറ്റ് കീകൾ, ആൾട്ടർനേറ്റ് കീകൾ എന്നിവ എഴുതുക.

25. താഴെ കൊടുത്തിരിക്കുന്ന STUDENT റിലേഷന്റെ പ്രൈമറി കീ Reg_no ആയാൽ.

a. STUDENT റിലേഷന്റെ കാൻഡിഡേറ്റ് കീകളും ആൾട്ടർനേറ്റ് കീകളും കണ്ടെത്തുക.

b. പ്രൈമറി കീയും കാൻഡിഡേറ്റ് കീയും തമ്മിലുള്ള ബന്ധം എന്താണ്?

Reg_no	Name	Batch	Result	Marks
101	Sachin	Science	Pass	480
103	Fathima	Humanities	Fail	200
106	Joseph	Commerce	Pass	350
108	Bincy	Science	Pass	300

26. ഒരു ഡാറ്റാബേസ് എന്നാൽ എന്ത്? DBMS ഉപയോഗിക്കുന്നതിന്റെ ഗുണങ്ങളും ദോഷങ്ങളും വിവരിക്കുക.

27. ഡാറ്റാസ്വാതന്ത്ര്യം എന്നാൽ എന്താണ്? ഭൗതിക ഡാറ്റാസ്വാതന്ത്ര്യവും ലോജിക്കൽ ഡാറ്റാസ്വാതന്ത്ര്യവും തമ്മിലുള്ള വ്യത്യാസം എഴുതുക.

28. ഡാറ്റാസ്റ്റാൻഡേർഡ് നിർവചിക്കുക എന്നത് ഒരു DBMS ന്റെ പ്രധാന സവിശേഷതയാണ്. ഒരു ഡാറ്റാബേസിൽ ഈ മാനദണ്ഡങ്ങൾ എങ്ങനെയാണ് ബാധകമാകുന്നത്?

29. യൂണിയൻ അനുരൂപമായ T1, T2 എന്നീ റിലേഷനുകളിൽ T1 ന്റെ കാർഡിനാലിറ്റി 10 ഉം T2 ന്റെ കാർഡിനാലിറ്റി 8 ഉം ആണ്. T1 $\cup$ T2 ന്റെ കാർഡിനാലിറ്റി 13 ആണെങ്കിൽ, T1 $\cap$ T2 ന്റെ കാർഡിനാലിറ്റി എന്തായിരിക്കും? നിങ്ങളുടെ ഉത്തരം സമർഥിക്കുക.

30. യൂണിയൻ അനുരൂപമായ T1, T2 എന്നീ റിലേഷനുകളിൽ T1 ന്റെ കാർഡിനാലിറ്റി 10 ഉം T2 ന്റെ കാർഡിനാലിറ്റി 8 ഉം ആണ്.

    a. T1 $\cup$ T2 ന്റെ സാധ്യമായ കാർഡിനാലിറ്റി എന്തായിരിക്കും?

    b. T1 $\cup$ T2 ന്റെ ഏറ്റവും കുറഞ്ഞ കാർഡിനാലിറ്റി എന്തായിരിക്കും?

31. Ctiy (ctiy_name, state), Hotel (name, address, ctiy_name) എന്നീ റിലേഷനുകൾ പരിഗണിച്ച് താഴെപ്പറയുന്നവയ്ക്കുള്ള റിലേഷണൽ ബീജഗണിത പ്രസ്താവനകൾ എഴുതുക.

    a. കൊച്ചി നഗരത്തിലെ ഹോട്ടലുകളുടെ പേരും വിലാസവും കണ്ടെത്തുക.

    b. കേരളത്തിലെ നഗരങ്ങളുടെ വിശദാംശങ്ങൾ.

    c. തൃശൂരിലെ ഹോട്ടലുകളുടെ പേരുകൾ കണ്ടെത്തുക.

    d. വിവിധ ഹോട്ടലുകളുടെ പേരുകൾ കണ്ടെത്തുക.

    e. കോഴിക്കോട് അല്ലെങ്കിൽ മൂന്നാറിലെ ഹോട്ടലുകളുടെ പേരുകൾ കണ്ടെത്തുക.

32. ചോദ്യം 23 ൽ കാണിച്ചിരിക്കുന്ന EMPLOYEE ബന്ധം ഉപയോഗിച്ചുകൊണ്ട്, താഴെ പറയുന്ന റിലേഷണൽ ബീജഗണിത എക്സ്പ്രഷനുകളുടെ ഫലം എഴുതുക.

    a. $\sigma_{\text{Department="Sales"}}$ (EMPLOYEE).

    b. $\sigma_{\text{salary>20000} \wedge \text{Department="Sales"}}$ (EMPLOYEE).

    c. $\sigma_{\text{salary>20000} \vee \text{Department="Sales"}}$ (EMPLOYEE).

    d. $\pi_{\text{name, salary}}$ (EMPLOYEE).

    e. $\pi_{\text{name, salary}}$ ($\sigma_{\text{Designation="Manager"}}$ (EMPLOYEE)).

    f. $\pi_{\text{name, Department}}$ ($\sigma_{\text{Designation="Clerk" ^ salary > 20000}}$ (EMPLOYEE)).

33. ഒരു ബാങ്കിലെ ഉപഭോക്താക്കളുടെ വിവരങ്ങൾ സൂക്ഷിച്ചിരിക്കുന്ന BORROWER, DEPOSITOR എന്നീ റിലേഷനുകൾ ഉപയോഗിച്ചുകൊണ്ട്, താഴെ പറയുന്ന റിലേഷണൽ ബീജഗണിത പ്രസ്താവനകൾ എഴുതുക.

a. നിക്ഷേപകരുടെയും വായ്പ എടുത്തവരുടെയും വിശദാംശങ്ങൾ പ്രദർശിപ്പിക്കുക.

b. നിക്ഷേപകരും വായ്പ എടുത്തവരുമായ ഉപഭോക്താക്കളുടെ പേര് പ്രദർശിപ്പിക്കുക.

c. വായ്പ എടുത്തിട്ടില്ലാത്ത നിക്ഷേപകരുടെ പേര് വിവരം പ്രദർശിപ്പിക്കുക.

d. നിക്ഷേപമില്ലാതെ വായ്പ എടുത്തവരുടെ പേരും വായ്പാ തുകയും പ്രദർശിപ്പിക്കുക.

BORROWER

Acc_No	Name	Amount
AC123	Albin	50000
AC103	Rasheeda	25000
AC106	Vishnu	25000
AC108	Aiswarya	30000

DEPOSITOR

Acc_No	Name	Amount
AC123	Albin	500
AC105	Shabana	25000
AC116	Vishnu	125000
AC108	Aiswarya	3000

34. താഴെകൊടുത്തിരിക്കുന്ന CUSTOMER, BRANCH എന്നീ റിലേഷനുകളുടെ കാർട്ടീഷ്യൻ പ്രൊഡക്ട് എഴുതുക.

CUSTOMER

Acc_No	Name	Branch_ID	Amount
AC123	Albin	B1001	50000
AC103	Rasheeda	B1001	25000
AC106	Vishnu	B1001	25000
AC108	Aiswarya	B1077	30000

BRANCH

Branch_ID	Name
B1001	Kochi
B1002	Guruvayur
B1077	Idukki

T6N8U2

9 സ്ട്രക്ചേർഡ് ക്വറി ലാംഗ്വേജ് (SQL)

പ്രധാന പഠനനേട്ടങ്ങൾ

ഈ അധ്യായത്തിന്റെ പഠനം പൂർത്തിയാകുന്നതോടെ പഠിതാവ് ആർജിക്കേണ്ട പഠനനേട്ടങ്ങൾ

- സ്ട്രക്ചേർഡ് ക്വറി ലാംഗ്വേജ് (SQL) എന്ന പ്രോഗ്രാമിങ് ഭാഷയുടെ പ്രാധാന്യവും വിശേഷഗുണങ്ങളും തിരിച്ചറിയുന്നു.
- SQL ന്റെ ഘടകങ്ങളെക്കുറിച്ച് വിശദമാക്കുന്നു.
- DDL, DML, DCL കമാൻഡുകളുടെ വ്യത്യാസം കണ്ടെത്തുന്നു.
- MySQL ന്റെ പ്രത്യേകതകൾ തിരിച്ചറിയുന്നു.
- വിവിധ ഡാറ്റ ഇനങ്ങളും അവയുടെ സവിശേഷതകളും പട്ടികപ്പെടുത്തുന്നു.
- SQL ലെ വിവിധ കൺസ്ട്രയിന്റുകളുടെ സ്വാധീനം വിശദമാക്കുന്നു.
- DDL കമാൻഡുകളായ CREATE, ALTER, DROP എന്നിവ ഉപയോഗിച്ച് വിവിധ പ്രവർത്തനങ്ങൾ ചെയ്യുന്നു.
- ഡാറ്റ കൈകാര്യം ചെയ്യുന്നതിനായി SELECT, INSERT, UPDATE, DELETE എന്നിവ പോലെയുള്ള DML കമാൻഡുകൾ ഉപയോഗിക്കുന്നു.
- SQL കമാൻഡുകളുടെ വിവിധ ഉപവാക്യങ്ങളും അവയുടെ ഉപയോഗവും മനസ്സിലാക്കുന്നു.
- വിവിധ നിബന്ധനകൾ ഉണ്ടാക്കുന്നതിന് ഓപ്പറേറ്ററുകൾ ഉപയോഗിക്കുന്നു.
- സംഗ്രഹ ഫങ്ഷനുകൾ (Aggregate Functions) പട്ടികപ്പെടുത്തുകയും അവയുടെ ഉപയോഗം വിവരിക്കുകയും ചെയ്യുന്നു.
- വിവരം ലഭ്യമാക്കുന്നതിന് നെസ്റ്റഡ് ക്വറി നിർമ്മിക്കുന്നു.

കഴിഞ്ഞ അധ്യായത്തിൽ നാം റിലേഷണൽ ഡാറ്റാബേസ് മാനേജ്മെന്റ് സിസ്റ്റത്തെക്കുറിച്ച് (RDBMS) ചർച്ച ചെയ്തിരുന്നു. റിലേഷണൽ ഡാറ്റാബേസ് എന്നത് പട്ടികകളിൽ (Tables) ശേഖരിച്ചുവച്ചിരിക്കുന്ന അനുബന്ധ വിവരങ്ങളുടെ ശേഖരണമാണെന്ന് നമുക്കറിയാം. ഇവയെ നമുക്ക് റിലേഷൻ (Relation) എന്നുവിളിക്കാം. വിവിധതരം പ്രവർത്തനങ്ങൾ റിലേഷനുകളിൽ ചെയ്യുന്നതിന് വേണ്ട റിലേഷണൽ ആൾജിബ്രയെക്കുറിച്ച് അടിസ്ഥാനാശയം നമുക്കുണ്ട്. പട്ടിക നിർമ്മിക്കുക, അതിലേക്ക് ഡാറ്റ ചേർക്കുക, അതിലെ ഡാറ്റ കൈകാര്യം ചെയ്യുക, ഡാറ്റ പുതുക്കുക, ഡാറ്റ നീക്കം ചെയ്യുക എന്നിവയെക്കുറിച്ച് ഇനി നമുക്ക് ഒരു വ്യക്തത വരേണ്ടത് ആവശ്യമാണ്. ഈ അധ്യായത്തിൽ സ്ട്രക്ചേർഡ് ക്വറി ലാംഗ്വേജിനെക്കുറിച്ചും (SQL) അതിന്റെ പ്രവർത്തനങ്ങളെക്കുറിച്ചും വിവരിക്കുന്നു. റിലേഷണൽ ഡാറ്റാബേസ് മാനേജ്മെന്റ് സിസ്റ്റം (RDBMS) പ്രോഗ്രാമുകളായ MySQL, ഒറാക്കിൾ, സൈബേസ്, ഇൻഫോമിക്സ്, പോസ്റ്റ്ഗ്രെസ്സ്, SQL സെർവർ, MS ആക്സസ് തുടങ്ങിയവയുടെ അടിസ്ഥാന ഡാറ്റാബേസ് SQL ആണ്. SQL പ്രാവർത്തികമാക്കുന്നതിന് ജനപ്രിയ സ്വതന്ത്ര RDBMS ആയ MySQL നാം ഉപയോഗിക്കുന്നു.

9.1 സ്ട്രക്ചേർഡ് ക്വറി ലാംഗ്വേജ് (Structured Query Language (SQL))

റിലേഷണൽ ഡാറ്റാബേസ് മാനേജ്മെന്റ് സിസ്റ്റത്തിലെ ഡാറ്റ കൈകാര്യം ചെയ്യാൻ SQL ഉപയോഗിക്കുന്നു. റിലേഷണൽ ഡാറ്റാബേസുമായി എളുപ്പത്തിലും കാര്യക്ഷമമായും സമ്പർക്കം പുലർത്താനുള്ള ഒരു വഴിയാണ് SQL. SQLന് ധാരാളം പതിപ്പുകൾ ഉണ്ട്. IBMന്റെ സാൻജോസ് ലബോറട്ടറിയിലെ (ഇപ്പോൾ അൽമാൻഡൻ റിസർച്ച് സെന്റർ) ഡൊണാൾഡ് ഡി ചാംബെർളിനും, റെയ്മണ്ട്.എഫ്.ബോയ്സും ചേർന്ന് 1970 ലാണ് ഇതിന്റെ ആദ്യ പതിപ്പ് വികസിപ്പിച്ചത്. ഈ ഭാഷയെ യഥാർഥത്തിൽ സ്ട്രക്ചർഡ് ഇംഗ്ലീഷ് ക്വറി ലാംഗ്വേജ് (sequel) എന്നാണ് വിളിച്ചിരുന്നത്. പിന്നീട് SQL എന്ന പേരിലേക്ക് മാറ്റി. 1986ൽ അമേരിക്കൻ നാഷണൽ സ്റ്റാൻഡേർഡ് ഇൻസ്റ്റിറ്റ്യൂട്ട് (ANSI) SQL സ്റ്റാൻഡേർഡ് പ്രസിദ്ധീകരിച്ചു.

റിലേഷണൽ ഡാറ്റാബേസ് സിസ്റ്റം എന്നത് പട്ടികകളുടെ (റിലേഷൻ) ഘടനാപരമായ ശേഖരമാണ്. ഡാറ്റ ഈ പട്ടികകളിലാണ് ശേഖരിക്കുന്നത്. ടേബിളുകൾ സവിശേഷമായി തിരിച്ചറിയുന്നത് പേരുകൾ ഉപയോഗിച്ചാണ്. അത് വരികളും നിരകളും ഉൾകൊള്ളുന്നു. ഒരു പട്ടികയിൽ ഒരു വരി പ്രതിനിധീകരിക്കുന്നത് ബന്ധപ്പെട്ട ഡാറ്റകളുടെ ശേഖരമാണ്. വരിയെ ടൂപ്പ്ൾ (Tuple) എന്നും നിരയെ ആട്രിബ്യൂട്ട് (Attribute) എന്നും വിളിക്കുന്നുവെന്ന് നമുക്കറിയാം.

നമുക്ക് ചെയ്യാം

student എന്ന പട്ടിക പരിശോധിക്കുക (പട്ടിക 9.1) തുടർന്ന് താഴെ കൊടുത്തിരിക്കുന്ന ചോദ്യങ്ങൾക്ക് ഡാറ്റാബേസുമായി ബന്ധപ്പെട്ട പദാവലിയുടെ അടിസ്ഥാനത്തിൽ ഉത്തരമെഴുതുക.

Adm_no	Name	Gender	Date_Birth	Income	Course
1001	Alok	M	02/10/1998	24000	Science
1002	Nike	M	26/01/1998	35000	Science
1003	Bharath	M	01/01/1999	45000	Commerce
1004	Virat	M	05/12/1998	22000	Science
1005	Meera	F	15/08/1998		Science
1006	Divakar	M	21/02/1998		Humanities

പട്ടിക 9.1: സ്റ്റുഡന്റ് പട്ടിക

i) ഈ പട്ടികയുടെ കാർഡിനാലിറ്റി ______________ ആണ്.

ii) ഈ പട്ടികയുടെ ഡിഗ്രി ______________ ആണ്.

iii) പട്ടികയിലെ വിവിധ വരികൾ (ടൂപ്പ്ൾ) എഴുതുക.

iv) പട്ടികയിലെ വിവിധ ആട്രിബ്യൂട്ടുകൾ എഴുതുക.

v) 'Course' എന്ന ആട്രിബ്യൂട്ടിന്റെ ഡൊമൈനിലെ വിലകൾ ഏതെല്ലാം?

RDBMS പ്രാവർത്തികമാക്കാനുള്ള ശക്തമായ ഒരു ഉപകരണമാണ് SQL. അത് പട്ടിക നിർമ്മിക്കുക, ഡാറ്റ ചേർക്കുക, ഡാറ്റ കൈകാര്യം ചെയ്യുക, ഡാറ്റ നീക്കം ചെയ്യുക, പട്ടികയുടെ ഘടന മാറ്റുക, പട്ടിക നീക്കം ചെയ്യുക തുടങ്ങിയവയ്ക്കുള്ള സൗകര്യങ്ങൾ നൽകുന്നു.

9.1.1 SQL ന്റെ സവിശേഷതകൾ (Features of SQL)

സ്ട്രക്ചേർഡ് ക്വറി ലാംഗ്വേജ് എന്നത് ANSI/ISO സ്റ്റാൻഡേർഡ് പ്രകാരമുള്ള, ഡാറ്റാബേസ് നിർമ്മിക്കുന്നതിനുള്ള ഒരു ഭാഷയാണ്. ക്വറി എന്നത് ഡാറ്റാബേസിനോടുള്ള അഭ്യർഥനയാണ്. മുൻപ് പറഞ്ഞിട്ടുള്ള എല്ലാ റിലേഷണൽ പ്രവൃത്തികളും അതിലൂടെ നമുക്ക് ചെയ്യാൻ സാധിക്കും. താഴെ പറയുന്ന സവിശേഷതകൾ കാരണം ക്വറികൾ നിർമ്മിക്കുന്നതിന് SQL ഫലപ്രദമാണ്.

- SQL എന്നത് ഒരു റിലേഷണൽ ഡാറ്റാബേസ് ഭാഷയാണ്. C, C++ എന്നിവ പോലെയുള്ള പ്രോഗ്രാമിങ് ഭാഷ അല്ല.
- അത് ലളിതവും, വഴക്കമുള്ളതും, സുശക്തവുമാണ്.
- അതിൽ പട്ടിക നിർമ്മിക്കുന്നതിനും, മാറ്റം വരുത്തുന്നതിനും, പട്ടികയിൽ ഡാറ്റ ചേർക്കുന്നതിനും, ഡാറ്റ കൈകാര്യം ചെയ്യുന്നതിനും ഉള്ള കമാൻഡുകൾ ഉണ്ട്.
- അത് ജനപ്രിയ RDBMS സോഫ്റ്റ് വെയറുകളായ Oracle, SQLServer, MySQL, MS Access, Sybase, Infomix തുടങ്ങിയവയ്ക്ക് മാർഗനിർദേശങ്ങൾ നൽകുന്നു.
- SQL ഒരു നോൺപ്രോസിഡറൽ ഭാഷയാണ്. ഏത് ഡാറ്റ തിരിച്ചെടുക്കണം, നീക്കം ചെയ്യണം, കൂട്ടിച്ചേർക്കണം അതിനുപരിയായി എങ്ങനെ ഈ പ്രവർത്തനങ്ങൾ ചെയ്യണം എന്ന് അത് വിവരിക്കുന്നു.
- ഡാറ്റയുടെ സുരക്ഷ ഉറപ്പാക്കുന്നതിന്റെ ഭാഗമായി ഡാറ്റാബേസിൽ നിന്നോ അല്ലെങ്കിൽ പട്ടികകളിലോ മാറ്റം വരുത്തുന്നതിന് വിവിധ ഉപയോക്താക്കൾക്ക് (Users) അനുമതി നൽകുന്നതിനും അത് റദ്ദ് ചെയ്യുന്നതിനും ഉള്ള സൗകര്യം SQL നൽകുന്നു.
- വ്യൂ എന്ന ആശയം ലഭ്യമാക്കുന്നു (ആശയം പിന്നീട് ഈ അധ്യായത്തിൽ ചർച്ച ചെയ്യും).

9.1.2 SQL ന്റെ ഘടകങ്ങൾ (Components of SQL)

SQL ന് മൂന്ന് ഘടകങ്ങളാണ് ഉള്ളത്. ഇത് ഡാറ്റ നിർവചന ഭാഷ (DDL), ഡാറ്റ കൈകാര്യം ചെയ്യുന്ന ഭാഷ (DML), ഡാറ്റ നിയന്ത്രണ ഭാഷ (DCL) എന്നിവയാണ്. ഇനി നമുക്ക് മൂന്ന് ഘടകങ്ങളെക്കുറിച്ചും RDBMS വികസിപ്പിക്കുന്നതിൽ ഇവയ്ക്കുള്ള പങ്ക് എന്താണെന്നും ചർച്ച ചെയ്യാം.

ഡാറ്റ നിർവചന ഭാഷ (DDL) (Data Definition Language)

പട്ടിക 9.1 പരിഗണിക്കുക (student പട്ടിക). ഇതുപോലെ ഒരു പട്ടിക നമുക്ക് എങ്ങനെ നിർമ്മിക്കാം? ഈ പട്ടികയിൽ പുതിയ നിര എങ്ങനെ കൂട്ടിച്ചേർക്കും? ഒരു ഡാറ്റാബേസിൽ നിന്ന് ഒരു പട്ടിക എങ്ങനെ ഒഴിവാക്കും? ഈ ചോദ്യങ്ങൾക്കെല്ലാം DDL പരിഹാരം നൽകും.

DDL എന്നത് RDBMSന്റെ സ്കീമാ (ഘടന) നിർവഹണം കൈകാര്യം ചെയ്യുന്നതിനുള്ള കമാൻഡുകൾ നൽകുന്ന SQL ന്റെ ഘടകമാണ്. DDL കമാൻഡുകൾ ഉപയോഗിക്കുന്നത് ഡാറ്റാബേസ് ഒബ്ജക്ടുകളായ പട്ടികകൾ, വ്യൂകൾ, കീകൾ എന്നിവ നിർമ്മിക്കാനും, പുതുക്കുവാനും, ഒഴിവാക്കുവാനും വേണ്ടിയാണ്. സാധാരണ DDL കമാൻഡുകൾ ആണ് CREATE, ALTER, DROP എന്നിവ.

ഡാറ്റ കൈകാര്യം ചെയ്യുന്ന ഭാഷ (DML) (Data Manipulation Language)

പട്ടിക 9.1 ൽ നമുക്ക് വിവിധതരം ടൂപ്പിളുകൾ (വരികൾ അല്ലെങ്കിൽ റെക്കോർഡുകൾ) കാണാൻ കഴിയും. ഈ ടൂപ്പിളുകൾ എങ്ങനെയാണ് ഒരു പട്ടികയിൽ ചേർക്കുന്നത്? ഒരു വിദ്യാർഥിയുടെ കുടുംബമാസവരുമാനം പുതുക്കണമെന്ന് വിചാരിക്കുക. ഇത് സാധ്യ മാണോ? പട്ടികയിൽനിന്ന് ഒരു കുട്ടിയുടെ റെക്കോർഡ് എങ്ങിനെ നീക്കം ചെയ്യും? ഡാറ്റ കൈകാര്യം ചെയ്യുന്ന ഭാഷ (DML) ഇത്തരം പ്രവർത്തനങ്ങൾക്കുള്ള കമാൻഡുകൾ നൽകുന്നു.

ഒരു കൂട്ടം കമാൻഡുകൾ നൽകുന്നതിലൂടെ ഡാറ്റാബേസ് സിസ്റ്റവുമായി കാര്യക്ഷമമായ ഉപയോക്തൃ ഇടപെടൽ നടത്തുന്നതിനുള്ള SQLന്റെ ഘടകമാണ് DML. പട്ടികകളിലേക്ക് ഡാറ്റ ചേർക്കാനും നിലവിലുള്ള ഡാറ്റ വീണ്ടെടുക്കാനും, പട്ടികകളിൽ നിന്നുള്ള ഡാറ്റ ഒഴിവാക്കുവാനും ശേഖരിച്ച ഡാറ്റ പരിഷ്ക്കരിക്കാനും ഉപയോക്താക്കളെ DML അനുവദിക്കുന്നു. സാധാരണമായി ഉപയോഗിക്കുന്ന DML കമാൻഡുകളാണ് SELECT, INSERT, UPDATE, DELETE എന്നിവ.

ഡാറ്റ നിയന്ത്രണ ഭാഷ (DCL) (Data Control Language)

ഡാറ്റ നിയന്ത്രണഭാഷ ഉപയോഗിക്കുന്നത് ഡാറ്റാബേസ് വിനിയോഗം നിയന്ത്രിക്കുന്നതിനാണ്. കാരണം ഡാറ്റാബേസിന്റെ സുരക്ഷിതത്വം വളരെ പ്രധാനപ്പെട്ടതാണ്. ഡാറ്റാകമ്മിറ്റിങ്ങും അഡ്മിനിസ്ട്രേറ്റിങ്ങ് പ്രിവിലേജും അടക്കമുള്ള ഡാറ്റ നിയന്ത്രണ കമാൻഡുകൾ അടങ്ങുന്നതാണ് DCL. GRANT, REVOKE എന്നിവ DCL ന്റെ ഭാഗമായി ഉപയോഗിക്കുന്നു.

GRANT : ഒരു ഡാറ്റാബേസിൽ ഉപയോക്താവിന് (User) പ്രത്യേക അവകാശങ്ങൾ നൽകുന്നു.

REVOKE : ഉപയോക്താവിന് GRANTഉപയോഗിച്ചു നൽകിയ പ്രത്യേക അവകാശങ്ങൾ പിൻവലിക്കുന്നു.

നിങ്ങളുടെ പുരോഗതി അറിയുക

1. SQL ന്റെ പൂർണരൂപം_____________________ ആണ്
2. SQL ന്റെ മൂന്നു ഘടകങ്ങൾ ഏതെല്ലാം?
3. SQL ഉപയോഗിക്കുന്നത്
   - a. ഡാറ്റാബേസിന്റെ ഘടന ഉണ്ടാക്കാൻ.
   - b. ഡാറ്റാബേസ് ക്വറി മാത്രം നിർമ്മിക്കാൻ.
   - c. ഡാറ്റാബേസിലെ ഡാറ്റയിൽ മാറ്റം വരുത്താൻ.
   - d. മുകളിൽ പറഞ്ഞവയെല്ലാം
4. SQL എന്നത്
   - a. ഒരു പ്രോഗ്രാമിംഗ് ഭാഷയാണ് b. ഒരു ഓപ്പറേറ്റിംഗ് സിസ്റ്റമാണ്
   - c. ഡാറ്റ കൈകാര്യം ചെയ്യാനുള്ള ഭാഷയാണ് d. ഒരു DBMS ആണ്
5. താഴെപ്പറയുന്നവയിൽ RDBMS പാക്കേജ് അല്ലാത്തത് ഏത് ?
   a. ORACLE b. SQL SERVER c. MySQL d. HTML

9.2 Working on MySQL

അമേരിക്കൻ നാഷണൽ സ്റ്റാൻഡേർഡ് ഇൻസ്റ്റിറ്റ്യൂട്ട് (ANSI) 1986 ലും ഇന്റർനാഷണൽ ഓർഗനൈസേഷൻ ഫോർ സ്റ്റാൻഡേർഡൈസേഷൻ (ISO) 1987 ലും SQL ന്റെ നിലവാരം നിശ്ചയിച്ചു. 1986 മുതൽ ഘടനയിൽ കുറച്ചു മാറ്റങ്ങൾ വരുത്തി SQL സ്റ്റാൻഡേർഡ് പരിഷ്ക്കരിച്ചു. പിന്നീട് ഈ സ്റ്റാൻഡേർഡ് പല പ്രാവശ്യമായി പരിഷ്ക്കരിക്കുകയും പുതിയ പതിപ്പുകൾ ഇറക്കുകയും ചെയ്തു. SQL ഡാറ്റാബേസ് ക്വറി ഭാഷയുടെ ISO ANSI സ്റ്റാൻഡേർഡിന്റെ ഏഴാമത്തെ പതിപ്പാണ് SQL:2011. ഡിസംബർ 2011 ലാണ് ഇത് ഔദ്യോഗികമായി അംഗീകരിക്കപ്പെട്ടത്. ഇത്തരം മാനദണ്ഡങ്ങൾ നിലനിൽക്കുന്നുണ്ടെങ്കിലും വ്യത്യസ്ത ഡാറ്റാബേസ് സോഫ്റ്റ്‌വെയർ പാക്കേജുകൾ ANSI SQL സ്റ്റാൻഡേർഡ് അനുസരിച്ചു അവരുടേതായ പതിപ്പുകൾ ഇറക്കുന്നുണ്ട്. അതിനാൽ മിക്ക SQL കോഡുകളും വ്യത്യസ്ത ഡാറ്റാബേസ് സോഫ്റ്റ്‌വെയറുകളിൽ ചെറിയ മാറ്റങ്ങളോടെയല്ലാതെ പ്രവർത്തിക്കുകയില്ല. ഈ അധ്യായത്തിൽ SQL പ്രവർത്തിക്കുന്നത് ഓപ്പൺ സോഴ്സ് സോഫ്റ്റ്‌വെയറായ MySQL ഉപയോഗിച്ചാണ്.

MySQL സൗജന്യവും, വളരെ വേഗം പ്രവർത്തിക്കുന്നതും, എളുപ്പത്തിൽ ഉപയോഗിക്കാൻ സാധിക്കുന്നതുമായ ഒരു RDBMS ആണ്

- അത് ഡാറ്റാബേസിന് ശക്തമായ സുരക്ഷ നൽകുന്നു.
- ഇത് എളുപ്പത്തിൽ ഉപയോഗിക്കാവുന്നതും പല ഓപ്പറേറ്റിംഗ് സിസ്റ്റങ്ങളിലും PHP, PERL, C, C++, JAVA തുടങ്ങിയ ഭാഷകളിൽ പ്രവർത്തിക്കുന്നതുമാണ്.

- MySQL വലിയ അളവിലുള്ള ഡാറ്റകളിൽ വേഗത്തിലും ഫലപ്രദമായും പ്രവർത്തിക്കുന്നു.
- ഇത് വെബ് ഡെവലപ്മെന്റിന് ഉപയോഗിക്കുന്ന ജനപ്രിയ പ്രോഗ്രാമിംഗ് ഭാഷകളിൽ ഒന്നായ PHP ക്ക് ഏറ്റവും അനുയോജ്യമാണ്.

1995 ൽ Michael "Monty" Widenius and David Axmark ചേർന്നാണ് MySQL വികസിപ്പിച്ചത്. യഥാർഥത്തിൽ ഇത് MySQL AB എന്ന സ്വീഡിഷ് കമ്പനിയുടെ ഉടമസ്ഥതയിലുള്ളതാണ്. പിന്നീട് 2008 ൽ സൺ മൈക്രോ സിസ്റ്റം ഇത് ഏറ്റെടുത്തു. സൺ മൈക്രോ സിസ്റ്റത്തെ 2010 ൽ ഒറാക്കിൾ കോർപ്പറേഷൻ ഏറ്റെടുത്തു. MySQL സാധാരണയായി Linux-Apache-MySQL-PHP, (LAMP), Windows-Apache-MySQL-PHP (WAMP), Mac-Apache-MySQL-PHP (MAMP) എന്ന രീതിയിലാണ് വിന്യസിച്ചിരിക്കുന്നത്. ഓപ്പറേറ്റിംഗ് സിസ്റ്റം ഉൾപ്പെടെ LAMP ലെ എല്ലാ ഘടകങ്ങളും സൗജന്യവും ഓപ്പൺ സോഴ്സുമാണ്. MySQLന്റെ ഔദ്യോഗിക വെബ് സൈറ്റാണ് www.mysql.com. റഫറൻസിനു MySQL ന്റെ സഹായ ഗ്രന്ഥം (Refence Manual) http://dev.mysq.co/doc ൽ ലഭ്യമാണ്.

9.2.1 MySQL തുറക്കുന്നു (Opening MySQL)

mysql> പ്രോംപ്റ്റിൽ കമാൻഡ് നൽകി നമുക്ക് MySQL പ്രവർത്തിപ്പിക്കാൻ സാധിക്കും. ഉബുണ്ടു ലിനക്സിൽ, ഈ പ്രോംപ്റ്റ് ലഭിക്കുന്നതിന് താഴെപറയുന്ന കമാൻഡ് ഉപയോഗിച്ച് ടെർമിനൽ ജാലകം തുറക്കണം.

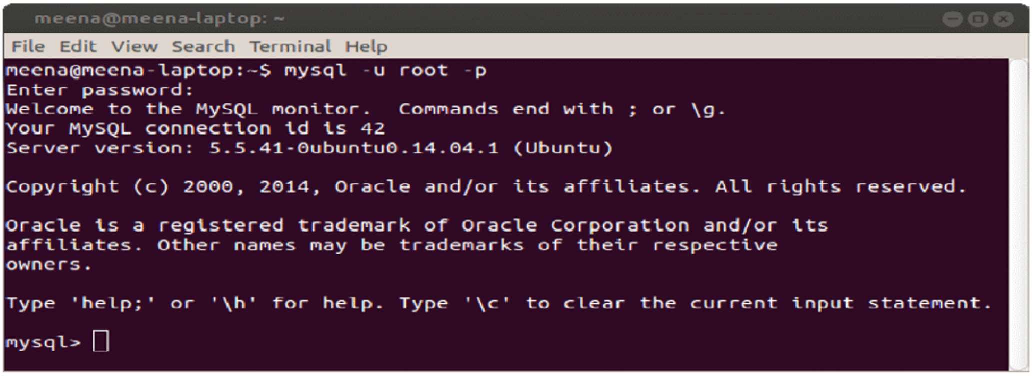
```
Applications -> Accessories -> Terminal
```

MySQL തുടങ്ങുന്നതിന് ടെർമിനൽ ജാലകത്തിൽ താഴെപറയുന്ന കമാൻഡ് നൽകണം

```
mysql -u root -p
```

വിൻഡോസ് ഓപ്പറേറ്റിംഗ് സിസ്റ്റത്തിൽ MySQL തുറക്കുന്നത് താഴെപ്പറയുന്ന രീതിയിലാണ്

Start -> Programs -> MySQL -> MySQL Server(Version Number) -> MySQL Commandline client.



ചിത്രം 9.1: ഉബുണ്ടു ലിനക്സ് ടെർമിനൽ ജാലകത്തിൽ MySQL പ്രോംപ്റ്റ്

MySQL തുറക്കുമ്പോൾ അത് പരിശോധനയ്ക്കായി പാസ്‌വേർഡ് ആവശ്യപ്പെടും. ഇൻസ്റ്റലേഷന്റെ സമയത്ത് ഉപയോഗിച്ച അതേ പാസ്‌വേർഡ് തന്നെ ആയിരിക്കണം ഇവിടെ നാം ഉപയോഗിക്കേണ്ടത്. പാസ്‌വേർഡ് പരിശോധനയ്ക്ക് ശേഷം MySQL ന്റെ പ്രോംപ്റ്റ് ചിത്രം 9.1 ൽ കാണുന്ന രീതിയിൽ ലഭിക്കുന്നു.

SQL കേസ് സെൻസിറ്റീവ് അല്ല. അതായത് ഇംഗ്ലീഷിലെ വലിയ അക്ഷരങ്ങൾ ഉപയോഗിച്ചോ, ചെറിയ അക്ഷരങ്ങൾ ഉപയോഗിച്ചോ, രണ്ടും കൂടിച്ചേർന്നോ കമാൻഡുകൾ നൽകാൻ സാധിക്കും. എന്നാൽ ഇതിനു ശേഷം മറ്റ് ടെക്സ്റ്റുകളിൽ നിന്ന് SQL കമാൻഡുകളെയും, കീ വേർഡുകളെയും വേർതിരിച്ചറിയുന്നതിനായി നാം ചില രീതികൾ ഉപയോഗിക്കും. കമാൻഡുകൾക്കും കീവേർഡുകൾക്കും വലിയ അക്ഷരങ്ങൾ (Upper Case Letters) ഉപയോഗിക്കുന്നു. എന്നാൽ ഉപയോക്താവ് നിഷ്കർഷിച്ച വാക്കുകളായ (User defined) ടേബിളിന്റെ പേര്, നിരകളുടെ പേര് തുടങ്ങിയവ ചെറിയ അക്ഷരത്തിൽ (Lower Case Letters) സൂചിപ്പിക്കുന്നു. കമാൻഡുകളും ഔട്ട്പുട്ടുകളും (പ്രതികരണങ്ങളും) ഒരു ടെക്സ്റ്റ് ഫയലിനകത്ത് **tee** എന്ന കമാൻഡ് നൽകി സൂക്ഷിക്കാവുന്നതാണ്. ഉദാഹരണത്തിന്, tee E:\output.txt എന്ന നിർദേശത്തിന്റെ പ്രവർത്തനത്തിന് ശേഷം സ്ക്രീനിൽ പ്രത്യക്ഷപ്പെടുന്നതെല്ലാം output.txt എന്ന ഫയൽ നിർമ്മിച്ച് E:ഡ്രൈവിൽ സംഭരിക്കുന്നു. ഈ അധ്യായത്തിൽ ഈ ഫയലിൽ സൂക്ഷിച്ചിട്ടുള്ള ഔട്ട്പുട്ടുകൾ ചിത്രരൂപേണ കൊടുത്തിരിക്കുന്നു.

ഉപയോക്താവിൽ നിന്ന് ഏതു തരം ക്വറിയും സ്വീകരിക്കാൻ MySQL തയാറാണ് എന്ന സന്ദേശമാണ് പ്രോംപ്റ്റ് നമുക്ക് നൽകുന്നത്. ഈ പ്രോംപ്റ്റിൽ നമുക്ക് ക്വറികൾ നൽകാവുന്നതാണ്.

MySQLൽ നിന്ന് പുറത്തുകടക്കാൻ QUIT അല്ലെങ്കിൽ EXIT എന്ന് കമാൻഡ് പ്രോംപ്റ്റിൽ നൽകണം.

```
mysql> EXIT;
```

9.2.2 MySQL ൽ ഒരു ഡാറ്റാബേസ് നിർമ്മിക്കുന്നു (Creating a database)

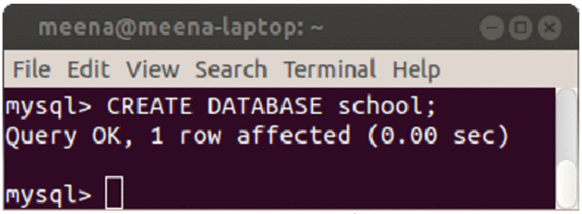
ഡാറ്റയിൽ പ്രവർത്തിക്കുന്നതിന് മുൻപ് നാം ഒരു ഡാറ്റാബേസ് സൃഷ്ടിക്കേണ്ടതുണ്ട്. ഡാറ്റാബേസ് എന്നത് പട്ടിക (Table) സംഭരിക്കുന്നതിനുള്ള ഒരു പെട്ടി പോലെയാണ്. MySQL ൽ ഡാറ്റാബേസ് ഉണ്ടാക്കുന്നതിന് CREATE DATABASE എന്ന കമാൻഡ് ഉപയോഗിക്കുന്നു. അതിന്റെ ഘടന താഴെപറയുന്ന രീതിയിലാണ്:

```
CREATE DATABASE <database_name>;
```

ഡാറ്റാബേസ് ഉണ്ടാക്കുമ്പോൾ താഴെ പറയുന്ന വസ്തുതകൾ ശ്രദ്ധിക്കേണ്ടതാണ്

- വാക്യഘടനയിൽ കൊടുത്ത `<database_name>` എന്നത് നിർമ്മിക്കേണ്ട ഡാറ്റാബേസിന്റെ പേരാണ്.
- ഡാറ്റാബേസിന്റെ പേര് സാധിക്കുന്നിടത്തോളം അർഥപൂർണവും വിവരണാത്മകവും ആയിരിക്കണം.
- `<database_name>` അനന്യമായിരിക്കണം. ഒരേ പേരിലുള്ള രണ്ട് ഡാറ്റാബേസുകൾ MySQL സെർവറിൽ ഉണ്ടാക്കാൻ സാധിക്കില്ല.

school എന്ന പുതിയ ഡാറ്റാബേസ് ഉണ്ടാക്കി നമ്മളുടെ ഡാറ്റാബേസ് പ്രവർത്തനങ്ങൾ ആരംഭിക്കാം. ചിത്രം 9.2ൽ കാണിച്ചിരിക്കുന്നത് ഈ കമാൻഡിന്റെ പ്രവർത്തനത്തിനുശേഷമുള്ള സ്ക്രീനാണ്. MySQL കമാൻഡ് പ്രോംപ്റ്റും, കമാൻഡും ആദ്യത്തെ വരിയിൽ കാണാൻ കഴിയും. ഈ കമാൻഡിനെ പ്രവർത്തിപ്പിച്ചശേഷം നൽകുന്ന സന്ദേശമാണ് *ചിത്രം 9.2ൽ കാണുന്നത്. (ഇനി മുതൽ മുകളിൽ കാണുന്ന രീതിയിലുള്ള സ്ക്രീൻ ഷോട്ടുകൾ ഒഴിവാക്കി പകരം നിർദിഷ്ട ഓപ്പറേഷനായുള്ള നിർദേശം പ്രത്യേക അക്ഷരങ്ങളിൽ (Seperate fonts) അവതരിപ്പിക്കുന്നതാണ്)*



ചിത്രം 9.2: ഒരു കമാൻഡിന്റെ പ്രവർത്തനത്തിനുശേഷം MySQL ജാലകം

9.2.3 ഡാറ്റാബേസ് തുറക്കുന്നു (Opening database)

ഒരു ഡാറ്റാബേസിൽ പ്രവർത്തനം നടത്തുന്നതിന് നാം അത് തുറക്കേണ്ടതുണ്ട്. നാം ഒരു ഡാറ്റാബേസ് തുറക്കുമ്പോൾ അത് MySQL സെർവറിലെ സജീവ ഡാറ്റാബേസ് *(Active database)* ആക്കി മാറ്റണം. ഇങ്ങനെ ഒരു ഡാറ്റാബേസ് തുറക്കുന്നതിന് USE എന്ന കമാൻഡാണ് MySQL ൽ നൽകേണ്ടത്. അതിന്റെ വാക്യഘടന താഴെ കൊടുത്തിരിക്കുന്ന രീതിയിലാണ്:

```
USE <database_name>;
```

ഇനി നമുക്ക് school എന്ന ഡാറ്റാബേസ് തുറക്കുന്നതിന് ചുവടെ കൊടുത്തിരിക്കുന്ന കമാൻഡ് നൽകാം.

```
USE school;
```

ഈ കമാൻഡിന്റെ പ്രവർത്തനത്തിന് ശേഷമുള്ള പ്രതികരണം താഴെ കൊടുത്തിരിക്കുന്നു:

```
Database changed
```

ഇപ്പോൾ school എന്ന ഡാറ്റാബേസാണ് സജീവ ഡാറ്റാബേസ്. ഇത് അർഥമാക്കുന്നത് വിവിധ DDL, DML, DCL കമാൻഡുകൾ ഇനി മുതൽ പ്രവർത്തിപ്പിക്കുന്നത് school ഡാറ്റാബേസിനെ അടിസ്ഥാനമാക്കിയാണ്. ഡാറ്റാബേസിന്റെ നിലനിൽപ്പ് ഇനി നമുക്ക് പരിശോധിക്കാം. ഒരു ഡാറ്റാബേസ് ഉണ്ടോ ഇല്ലയോ എന്നറിയാൻ SHOW DATA BASES എന്ന കമാൻഡ് ഉപയോഗിക്കാം. അത് കംപ്യൂട്ടറിലെ എല്ലാ ഡാറ്റാബേസുകളും പട്ടികപ്പെടുത്തുന്നു.

```
SHOW DATABASES;
```

ഈ കമാൻഡിന്റെ ഔട്ട്പുട്ട് ചിത്രം 9.3 ൽ കൊടുത്തിരിക്കുന്നു.

```
mysql> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| school             |
| test               |
+--------------------+
4 rows in set (0.00 sec)
```

ചിത്രം 9.3: SHOW DATABASES കമാൻഡിന്റെ ഒട്ട്പുട്ട്

9.2.4 SQL ലെ ഡാറ്റ ഇനങ്ങൾ (Data types in SQL)

ഒരു പട്ടികയിലെ നിരയിൽ നൽകുന്ന വിലയുടെ ഇനത്തെ ഡാറ്റ ഇനം നിർവചിക്കുന്നു. ഡാറ്റ ഇനം ഒരു ഡാറ്റയുടെ കൃത്യത ഉറപ്പുവരുത്തുന്നു. ഡാറ്റാബേസ് ഡിസൈനിങ് സമയത്ത് നിരകൾക്ക്

കൃത്യമായ ഡാറ്റ ഇനങ്ങൾ നൽകുവാൻ ശ്രദ്ധിക്കേണ്ടതാണ്. ഉദാഹരണത്തിന് 2 എന്ന സംഖ്യയെ ടെക്സ്റ്റ് ഡാറ്റ ഇനമായി (സ്ട്രിംഗ്) നൽകിയാൽ അതിനെ ഗണിത പ്രക്രിയയിൽ ഉപയോഗിക്കാൻ കഴിയില്ല. നേരെമറിച്ച് അതേ സംഖ്യ പൂർണസഖ്യ നിരയിൽ നൽകിയാൽ അതിനെ ഗണിതത്തിന് ഉപയോഗിക്കാം. അതുകൊണ്ട് SQL ഡാറ്റ ഇനങ്ങൾ എന്ന ആശയത്തെ അവ പ്രതിനിധാനം ചെയ്യുന്ന ഡാറ്റ ഇനങ്ങൾ, അവ ഓരോന്നും ഉൾക്കൊള്ളുന്ന വിലകളുടെ പരിധി മുതലായ ഘടകങ്ങളുടെ അടിസ്ഥാനത്തിൽ നമുക്ക് മനസ്സിലാക്കാം. SQL ന്റെ വ്യത്യസ്ത പതിപ്പുകളിൽ ഡാറ്റ ഇനങ്ങൾ വ്യത്യസ്തമായിരിക്കും. ഡാറ്റ ഇനം SQL ന്റെ വ്യത്യസ്ത പതിപ്പുകളിൽ വ്യത്യാസമുണ്ട്.

ഡാറ്റ ഇനത്തെ മൂന്നായി തരംതിരിക്കാം. സംഖ്യാഡാറ്റ ഇനം, സ്ട്രിംഗ്ഡാറ്റ ഇനം, തീയതി സമയം എന്നിവ സൂചിപ്പിക്കുന്ന ഡാറ്റ ഇനം എന്നിവയാണവ. സംഖ്യാപരമായ മൂല്യങ്ങളായ 7,100,234,-456,0 തുടങ്ങിയ സംഖ്യകളെ സംഖ്യാ ഡാറ്റ ഇനം ഉപയോഗിച്ച് പ്രതിനിധാനം ചെയ്യാം. "Aleena" (ഒരു വിദ്യാർഥിയുടെ പേര്), "Kerala" (ഒരു സംസ്ഥാനത്തിന്റെ പേര്), "F" (ലിംഗഭേദത്തെ സൂചിപ്പിക്കാൻ) എന്നിവ സ്ട്രിംഗ് ഡാറ്റ ഇനം ആണ്. '01-01-2020', '23:34:3' തുടങ്ങിയവ തീയതിയും സമയവും സൂചിപ്പിക്കാനുള്ള ഡാറ്റ ഇനം ഉപയോഗിച്ച് പ്രതിനിധാനം ചെയ്യാം.

a. സംഖ്യാഡാറ്റ ഇനങ്ങൾ

സംഖ്യാഡാറ്റ ഇനത്തിന്റെ മൂല്യങ്ങളെ ഏതെങ്കിലും സാധാരണ സംഖ്യ പോലെ ഉപയോഗിക്കാം. അവയിൽ ഗണിതക്രിയകളായ സങ്കലനം, വ്യവകലനം, ഗുണനം, ഹരണം എന്നിവ ചെയ്യാം. MySQL ൽ സാധാരണയായി ഉപയോഗിക്കുന്ന സംഖ്യാഡാറ്റ ഇനങ്ങളാണ് `INT` അല്ലെങ്കിൽ `INTEGER` ഉം, `DEC` അല്ലെങ്കിൽ `DECIMAL` ഉം.

(i) INT അല്ലെങ്കിൽ INTEGER

നമുക്കറിയാവുന്നതുപോലെ പൂർണസംഖ്യകൾ എന്നാൽ ദശാംശ ഭാഗം ഇല്ലാത്ത സംഖ്യകളാണ്. അവ അധിസംഖ്യയോ, പൂജ്യമോ, ന്യൂനസംഖ്യയോ ആവാം. MySQLൽ ഒരു പൂർണസംഖ്യയെ പ്രതിനിധാനം ചെയ്യാൻ `INT` അല്ലെങ്കിൽ `INTEGER` എന്ന ഡാറ്റ ഇനം ഉപയോഗിക്കാം. ഉദാഹരണത്തിന് 69, 0, 112 തുടങ്ങിയ ഡാറ്റകൾ INT ഡാറ്റ ഇനങ്ങളാണ്.

(ii) DEC അല്ലെങ്കിൽ DECIMAL

ദശാംശ ഭാഗത്തോടുകൂടിയ സംഖ്യകളെ `DEC` അല്ലെങ്കിൽ `DECIMAL` എന്ന ഡാറ്റ ഇനം കൊണ്ട് പ്രതിനിധീകരിക്കാം. ഈ ഡാറ്റ ഇനത്തിന്റെ മാതൃകാ രൂപം `DECIMAL(size,D)` അല്ലെങ്കിൽ `DEC(size,D)` എന്നാണ്. ഇവിടെ `size` എന്ന ഘടകം, ദശാംശ ഭാഗം ഉൾപ്പെടുന്ന മൂല്യത്തിന്റെ മൊത്തം എണ്ണം സൂചിപ്പിക്കുന്നു. `D` എന്നത് സൂചിപ്പിക്കുന്നത് പൂർണ സംഖ്യകൾക്കു ശേഷമുള്ള അക്കങ്ങളുടെ എണ്ണമാണ്. ഉദാഹരണത്തിന് `DEC (5,2)` അല്ലെങ്കിൽ `DECIMAL(5,2)` എന്നതിൽ 5 സംഖ്യയുടെ കൃത്യതയും (precision), 2 അതിന്റെ അളവും (Scale) സൂചിപ്പിക്കുന്നു. ഈ സവിശേഷതയോടുകൂടി ഒരു നിരയിൽ ഏറ്റവും കൂടിയത് അഞ്ച് അക്കം സൂക്ഷിക്കാം അതിൽ 2 അക്കങ്ങൾ ദശാംശ ബിന്ദുവിന് ശേഷമായിരിക്കും. ഈ ഡാറ്റ ഇനം ഉപയോഗിച്ച് ഒരു നിരയിൽ നൽകാൻ സാധിക്കുന്ന അക്കങ്ങളുടെ പരിധി -999.99 മുതൽ 999.99 വരെയാണ്. MySQL ലിലെ സംഖ്യാഡാറ്റ ഇനങ്ങളുടെ ഒരു അവലോകനമാണ് പട്ടിക 9.2 ൽ നൽകിയിരിക്കുന്നത്. ഇതിൽ നൽകിയിരിക്കുന്ന മൂല്യങ്ങൾ ഓരോ SQL പതിപ്പിനെയും ആശ്രയിച്ചിരിക്കും എന്ന് ഓർമിക്കുക.

ഡാറ്റാ തരം	ഉപയോഗം	ചിഹ്നത്തോടു കൂടിയത്	ചിഹ്നം ഇല്ലാതെ	സംഭരണ ശേഷി (ബൈറ്റിൽ)
TINY INT	വളരെ ചെറിയ പൂർണ സംഖ്യ	-128 മുതൽ 127	0 മുതൽ 255	1
SMALL INT	ചെറിയ പൂർണ സംഖ്യ	- 32768 മുതൽ 32767	0 മുതൽ 65535	2
MEDIUM INT	ഇടത്തരം വലുപ്പമുള്ള പൂർണസംഖ്യ	-8388608 മുതൽ 8388607	0 മുതൽ 1677215	3
INT	സാധാരണ പൂർണ സംഖ്യാമൂല്യം	- 2147483648 മുതൽ 2147483647	0 മുതൽ 4294967295	4
BIG INT	വലിയ പൂർണ സംഖ്യാമൂല്യം	19 അക്കങ്ങൾ വരെയുള്ള മൂല്യം	264 അക്കങ്ങൾ വരെയുള്ള മൂല്യം	8
FLOAT(M,D)	ദശാംശ സംഖ്യ	പൂർണസംഖ്യ ഭാഗം 24 ആകുന്നതുവരെ		4
DOUBLE(M,D)	ഇരട്ട കൃത്യത യുള്ള ദശാംശ സംഖ്യാമൂല്യം	പൂർണസംഖ്യ ഭാഗം 53 ആകുന്നതുവരെ		8
DECIMAL(M,D)	സൂക്ഷ്മമായ കൃത്യതയുള്ള ദശാംശ സംഖ്യാമൂല്യം	സംഖ്യാഡാറ്റ ഇനത്തിനു ഏറ്റവും കൂടിയത് 65 അക്കങ്ങൾ വരെ, അതിൽ 30 അക്കങ്ങൾ ദശാംശ ബിന്ദുവിന് ശേഷം		8

പട്ടിക 9.2: MySQL ലെ സംഖ്യ ഡാറ്റ ഇനങ്ങളും അവയുടെ പ്രത്യേകതകളും

b) സ്ട്രിംഗ് ഡാറ്റ ഇനങ്ങൾ

സ്ട്രിംഗ് എന്നാൽ അക്ഷരങ്ങളുടെ കൂട്ടമാണ് MySQLലിൽ ഏറ്റവും സാധാരണയായി ഉപയോഗിക്കുന്ന സ്ട്രിംഗ്ഡാറ്റ ഇനങ്ങളാണ് CHARACTER അല്ലെങ്കിൽ CHAR ഉം, VARCHAR ഉം.

(i) CHARACTER അല്ലെങ്കിൽ CHAR

അക്ഷരങ്ങൾ, ലിപികൾ, സംഖ്യകൾ, പ്രത്യേക ചിഹ്നങ്ങൾ മുതലായവ ഇതിൽ ഉൾപ്പെടുന്നു. CHAR ന് ഒരു നിശ്ചിത വലുപ്പമാണ് ഉണ്ടായിരിക്കുക. ഈ ഡാറ്റ ഇനത്തിന്റെ ഘടന CHAR(X) എന്നാണ്. X എന്നത് ഡാറ്റ നിർമ്മിക്കാൻ ഉപയോഗിക്കുന്ന അക്ഷരങ്ങളുടെ

പരമാവധി എണ്ണം സൂചിപ്പിക്കുന്നു X ന്റെ വില 0 നും 255 നും ഇടയിലായിരിക്കും. ഒരു നിരയിലെ നിശ്ചിത വലുപ്പമുള്ളതും, ചെറുതുമായ ഡാറ്റകൾക്കാണ് CHAR പ്രധാനമായും ഉപയോഗിക്കുന്നത്. ഉദാഹരണത്തിന് ഒരു പട്ടികയിലെ Gender എന്ന നിരയിൽ പുരുഷനെ സൂചിപ്പിക്കുന്നതിന് 'M' എന്നും സ്ത്രീയെ സൂചിപ്പിക്കുന്നതിന് 'F' എന്നും നമുക്ക് സംഭരിക്കണമെങ്കിൽ ഈ നിരയെ CHAR ഡാറ്റ ഇനമായി പ്രഖ്യാപിക്കുന്നതാണ് നല്ലത്. സംഭരിക്കേണ്ട ഡാറ്റയ്ക്ക് ആവശ്യമില്ലെങ്കിലും നിർദേശിക്കപ്പെട്ട അത്രയും സ്ഥലം എപ്പോഴും ഉപയോഗിക്കുന്നു. ഡാറ്റയിലുള്ള അക്ഷരങ്ങൾ നിരയുടെ പ്രഖ്യാപിത വലുപ്പത്തെക്കാൾ കുറവാണെങ്കിൽ ബാക്കിയുള്ള ഭാഗം ശൂന്യസ്ഥലം (spacebar character) ഉപയോഗിച്ചു പൂർണമാക്കുന്നതാണ്. എന്നാൽ പട്ടികയിൽ നിന്ന് ഈ മൂല്യം വീണ്ടെടുക്കുമ്പോൾ, ബാക്കിയുള്ള എല്ലാസ്ഥലങ്ങളും ഒഴിവാക്കപ്പെടുന്നു. CHAR തരത്തിലുള്ള നിരയുടെ വലുപ്പം 1 ആണെങ്കിൽ, വലുപ്പം സൂചിപ്പിക്കേണ്ടതില്ല, കാരണം CHAR ന്റെ തനത് വലുപ്പം 1 ആണ്.

(ii) VARCHAR (size)

VARCHAR എന്നത് വിവിധ നീളത്തിലുള്ള സ്ട്രിങ്ങുകളെ സൂചിപ്പിക്കുന്നു. അത് CHARന് സമാനമാണ്. എന്നാൽ ഡാറ്റയ്ക്ക് അനുവദിച്ച സ്ഥലം അതിന്റെ യഥാർഥ വലുപ്പത്തെ ആശ്രയിച്ചിരിക്കും അല്ലാതെ പ്രഖ്യാപിക്കപ്പെട്ട വലുപ്പം അനുസരിച്ചല്ല. ഉദാഹരണമായി ഒരു പട്ടികയിലെ Name എന്ന നിരയിൽ ഡാറ്റ സംഭരിക്കണമെങ്കിൽ ആ നിരയെ VARCHAR തരമായി പ്രഖ്യാപിക്കുന്നതാണ് നല്ലത്. കാരണം ആ നിരയിലെ ഡാറ്റയിൽ വ്യത്യസ്ത എണ്ണം അക്ഷരങ്ങൾ അടങ്ങിയിരിക്കാം. ഒരു സ്ട്രിങ്ങിന്റെ നീളം 0 മുതൽ 65535 വരെ ആകാം. VARCHAR, മെമ്മറി സ്ഥലം മിതമായി ഉപയോഗിക്കുന്നു. എന്തുകൊണ്ടെന്നാൽ, വിലകൾ സംഭരിക്കുമ്പോൾ അതിനോടുകൂടി ശൂന്യസ്ഥലങ്ങൾ ചേർക്കുന്നില്ല. മേൽവിലാസം, പേരുകൾ തുടങ്ങിയവ ഇത്തരം ഡാറ്റകൾക്ക് ഉദാഹരണകളാണ്

c. തീയതി, സമയം എന്നീ ഡാറ്റ ഇനങ്ങൾ

MySQL ലിൽ തീയതിയും, സമയവും ശേഖരിക്കുന്നതിന് ഡാറ്റ ഇനങ്ങളുണ്ട്. തീയതി സംഭരിക്കുന്നതിന് MySQL ൽ ഉപയോഗിക്കുന്ന ഡാറ്റ ഇനമാണ് DATE. അതുപോലെ സമയം സംഭരിക്കുന്നതിന് TIME ഡാറ്റാ ഇനം ഉപയോഗിക്കുന്നു.

i) DATE

DATE ഡാറ്റ ഇനം ഉപയോഗിക്കുന്നത് തീയതികൾ സംഭരിക്കുന്നതിനാണ്. YYYY-MM-DD എന്ന രൂപത്തിലാണ് MySQL തീയതിയെ സംഭരിക്കുന്നത്. ഇതുപയോഗിച്ച് 1000-01-01 മുതൽ 9999-12-31 വരെ രേഖപ്പെടുത്താവുന്നതാണ്. MySQL ൽ ഒരു രീതിയിൽ തീയതികൾ പ്രദർശിക്കപ്പെടുന്നു എന്നാൽ SQL പ്രസ്താവനകളിൽ നമുക്ക് വിവിധ തരം DATE രൂപഘടനകൾ ഉപയോഗിക്കാം. YYYY-MM-DD എന്നതാണ് അടിസ്ഥാന മാതൃക. എന്നാൽ നമുക്ക് ഏതെങ്കിലും വിരാമചിഹ്നം (Punctuation) തീയതികൾക്കിടയിൽ ഉപയോഗിക്കാൻ സാധിക്കും. ഉദാഹരണത്തിന് '2011-01-24', '2011/01/25', '20111026' തുടങ്ങിയവ സാധുവായ തീയതികളാണ്. മുകളിലുള്ള ഏതെങ്കിലും മാതൃകയിൽ DATE ഡാറ്റ തരത്തിലുള്ള നിരയിലേക്ക് വിലകൾ നമുക്ക് ചേർക്കാവുന്നതാണ്. MySQL ൽ DATE

എല്ലായ്പ്പോഴും year-month-day എന്ന ക്രമത്തിൽ തന്നെ കൊടുത്തിരിക്കണം (ഉദാഹരണത്തിന് '98-09-04')

ii) TIME

ഒരു നിരയിൽ സമയ ഡാറ്റ സംഭരിക്കുന്നതിനാണ് MySQL ൽ TIME ഡാറ്റ ഇനം ഉപയോഗിക്കുന്നത്. HH:MM:SS അംഗീകൃത മാതൃകയിലാണ് അത് വിലകൾ പ്രദർശിപ്പിക്കുന്നത്. TIME ഡാറ്റ ഇനം ഉപയോഗിച്ച് ഒരു പ്രത്യേക സമയം (10 മണിക്കൂർ 5 മിനിറ്റ് 2 സെക്കന്റ് എന്ന പോലെ) സംഭരിക്കുന്നതിനോടൊപ്പം, രണ്ടു സമയത്തിനിടയിലുള്ള ഇടവേള (23 മണിക്കൂറിനെക്കാൾ അധികമുള്ളതും) സംഭരിക്കുന്നതിനും അത് ഉപയോഗിക്കാം.

നമുക്ക് ചെയ്യാം

പട്ടിക 9.3 ലെ ഡാറ്റ ഇനം എന്ന നിര അനുയോജ്യമായ ഡാറ്റ ഇനം ഉപയോഗിച്ച് പൂരിപ്പിക്കുക.

മൂല്യം	ഡാറ്റ ഇനം
325.678	
'A'	
'Computer'	
'2016-01-01'	
450	
22:32:45	
456787	

പട്ടിക 9.3: വിലകളുടെ ഡാറ്റ ഇനം

നിങ്ങളുടെ പുരോഗതി അറിയുക

1. __________ കമാൻഡാണ് ഒരു ഡാറ്റാബേസിനെ സജീവമാക്കാൻ ഉപയോഗിക്കുന്നത്
2. കമ്പ്യൂട്ടറിൽ ഉള്ള ഡാറ്റാബേസിന്റെ പേര് എങ്ങനെ കാണാൻ കഴിയും?
3. CHAR, VARCHAR എന്നീ ഡാറ്റ ഇനങ്ങൾ തമ്മിലുള്ള വ്യത്യാസം എന്ത്?
4. DATE ഡാറ്റ ഇനം എങ്ങനെയാണ് മൂല്യങ്ങൾ സംഭരിക്കുന്നത്?
5. CHAR (5) എന്ന ഡാറ്റ ഇനത്തിൽ പ്രഖ്യാപിച്ചിട്ടുള്ള ഒരു നിരയിൽ 234 എന്ന സംഖ്യ നമുക്ക് സംഭരിക്കാൻ സാധിക്കുമോ?

9.3 SQL കമാൻഡുകൾ (SQL commands)

ഡാറ്റാബേസിൽ വിവിധ തരം പ്രവർത്തനങ്ങൾ ചെയ്യുന്നതിനുള്ള കമാൻഡുകൾ SQLൽ ലഭ്യമാക്കിയിരിക്കുന്നു. നാം മുൻപ് സൂചിപ്പിച്ചതു പോലെ കമാൻഡുകളെ DDL, DML, DCL കമാൻഡുകളെന്നും തരംതിരിച്ചിട്ടുണ്ട് ഏറ്റവും സാധാരണയായി ഉപയോഗിക്കുന്ന

DDL, DML കമാൻഡുകളെക്കുറിച്ച് നമുക്ക് ചർച്ച ചെയ്യാം. ഡാറ്റാബേസിന്റെ ഘടനാപരമായ കാര്യങ്ങൾ ചെയ്യുന്നതിനാണ് DDL കമാൻഡുകൾ ഉപയോഗിക്കുന്നത്. ഒരു പട്ടിക നിർമ്മിക്കുക, അത് പുതുക്കുക, പട്ടിക നീക്കം ചെയ്യുക തുടങ്ങിയ പ്രവർത്തനങ്ങൾ ഇതിൽ ഉൾപ്പെടുന്നു.

ഒരു പട്ടികയുടെ ഉള്ളടക്കവുമായി ബന്ധപ്പെട്ട പ്രവർത്തനങ്ങൾക്കാണ് DML കമാൻഡുകൾ ഉപയോഗിക്കുന്നത്. ഒരു പട്ടികയിൽ റെക്കോർഡുകൾ ചേർക്കുക, വീണ്ടെടുക്കുക, പരിഷ്ക്കരിക്കുക, ഒഴിവാക്കുക എന്നീ പ്രവർത്തനങ്ങൾ ഇതിൽ ഉൾപ്പെടുന്നു. ഒരു പ്രത്യേക വസ്തുവിനെ സംബന്ധിച്ച ഡാറ്റ സംഭരിക്കുന്നതിന് പട്ടിക നിർമ്മിക്കുക, ആവശ്യമായ വിവരം വീണ്ടെടുക്കുക, ആവശ്യമില്ലാത്തവ ഒഴിവാക്കുക എന്നീ പ്രവർത്തനങ്ങൾ ചെയ്യുന്നതിന് ഈ കമാൻഡുകൾ പരിചയപ്പെടുത്തുന്നു.

9.4 പട്ടികയുടെ നിർമ്മാണം (Creating tables)

ഒരു റിലേഷണൽ ഡാറ്റാബേസിന്റെ ഏറ്റവും പ്രധാനപ്പെട്ട ഘടകമാണ് പട്ടികകൾ. ഏതൊരു ഡാറ്റാബേസിന്റെയും പ്രാഥമികമായ ആവശ്യം പട്ടികകളിൽ സംഭരിച്ചിരിക്കുന്ന ഡാറ്റ പരിപാലിക്കുക എന്നതാണ്. ഇപ്പോൾ നമുക്ക് പട്ടിക 9.1 ലുള്ള student എന്ന പട്ടിക പരിഗണിക്കാം. ഒരു കൂട്ടം നിരകളുള്ള ഒരു പട്ടിക നാം എങ്ങനെയാണ് നിർമ്മിക്കുന്നത്? `CREATE TABLE` എന്ന DDL കമാൻഡ് ഉപയോഗിച്ചാണ് പട്ടിക നിർവചിക്കുന്നത്. ഇതിന് പട്ടികയുടെ പേര്, നിരയുടെ പേര്, ഡാറ്റ ഇനം, അവയുടെ വലുപ്പം മറ്റെന്തെങ്കിലും നിബന്ധനകൾ ഉണ്ടെങ്കിൽ അവയും ഉൾപ്പെടുത്തി പട്ടിക നിർവചിക്കേണ്ടതുണ്ട്. ഒരു പട്ടികയിൽ കുറഞ്ഞത് ഒരു നിരയെങ്കിലും ഉണ്ടായിരിക്കണം. `CREATE TABLE` കമാൻഡിന്റെ വാക്യഘടന താഴെക്കൊടുക്കുന്നു.

```
CREATE TABLE <table_name>
(<column_name> <data_type> [<constraint>]
[, <column_name> <data_type> [<constraint>,]
  ...............................
  ...............................   );
```

ഇവിടെ `<table_name>` എന്നത് പട്ടികയുടെ പേര് സൂചിപ്പിക്കുന്നു. `<column_name>` എന്നത് പട്ടികയിലെ ഒരു നിരയുടെ പേര് സൂചിപ്പിക്കുന്നു. `<constraint>` എന്നത് ഒരു നിരയിലെ വിലകളിൽ ഉപയോഗിക്കാവുന്ന നിയമങ്ങൾ ഏതെന്ന് വ്യക്തമാക്കുന്നു. എല്ലാ നിരകളും ഒരു ജോഡി ആവരണചിഹ്നങ്ങൾക്ക് (Parenthesis) അകത്താണ് എഴുതേണ്ടത്. ഓരോന്നും കോമ ഉപയോഗിച്ച് വേർതിരിക്കണം. നമുക്ക് എല്ലാ നിരകളും ഒരു വരിയിൽ എഴുതുവാൻ കഴിയും.

9.4.1 പട്ടികകളുടെയും, നിരകളുടെയും പേര് നൽകുന്നതിനുള്ള നിയമങ്ങൾ (Rules for naming tables and columns)

പട്ടികകയ്ക്കും, നിരയ്ക്കും പേരു നൽകുമ്പോൾ താഴെ പറയുന്ന കാര്യങ്ങൾ ശ്രദ്ധിക്കേണ്ടതാണ്

- പേരിൽ അക്ഷരങ്ങൾ (A-Z, a-z), സംഖ്യകൾ(0-9), അണ്ടർ സ്കോർ (_), ഡോളർ($) തുടങ്ങിയ ചിഹ്നങ്ങൾ ഉപയോഗിക്കാവുന്നതാണ്.
- പേരിനകത്ത് ഒരു അക്ഷരമെങ്കിലും ഉണ്ടായിരിക്കണം.
- പേരിൽ ശൂന്യസ്ഥലം, പ്രത്യേക ചിഹ്നങ്ങൾ എന്നിവ ഉപയോഗിക്കാൻ പാടില്ല.
- SQL ലിനകത്തെ കീവേർഡുകൾ ഉപയോഗിക്കാൻ പാടില്ല.
- ഒരു പട്ടികയുടെ പേര് മറ്റൊരു പട്ടികയ്ക്ക് നൽകാൻ പാടില്ല.

ചില MySQL പതിപ്പുകളിൽ പട്ടികയുടെ പേര് വിരാമ ചിഹ്നങ്ങൾ (" ' ") ഉപയോഗിച്ച് എഴുതാറുണ്ട്. പട്ടികയുടെ പേര് ഉദ്ധരണികൾക്കുള്ളിൽ ഉപയോഗിക്കാം. ഏത് പ്രത്യേക ചിഹ്നവും പട്ടികയുടെ പേരിൽ ഉൾപ്പെടുത്താം.

സീരിയൽ നമ്പർ	ആട്രിബ്യൂട്ടുകൾ	വിശദീകരണം
1	Admission number	പൂർണ സംഖ്യ
2	Name	20 അക്ഷരങ്ങളുള്ള സ്ട്രിംഗ്
3	Gender	ഒരക്ഷരം
4	Date of birth	തീയതി
5	Course	15 അക്ഷരങ്ങളുള്ള ഒരു സ്ട്രിംഗ്
6	Family income	പൂർണ സംഖ്യ

പട്ടിക 9.4: student പട്ടികയുടെ ആട്രിബ്യൂട്ടുകൾ

പട്ടിക 9.4 ൽ കൊടുത്തിരിക്കുന്ന വിശദാംശങ്ങൾക്ക് അനുസരിച്ച് ഒരു പട്ടിക നിർമ്മിക്കാനാവശ്യമായ പ്രസ്താവന താഴെ കൊടുക്കുന്നു.

ക്വറി 9.1

```
CREATE TABLE student
(adm_no INT,
name VARCHAR(20),
gender CHAR,
dob DATE,
course VARCHAR(15),
f_income INT);
```

ഇവിടെ CREATE TABLE എന്ന കമാൻഡ് adm_no, name, gender, dob, course, f_income എന്നീ ആറു നിരകളോടു കൂടിയ ഒരു പട്ടിക നിർമ്മിക്കുന്നു. adm_no എന്ന നിരയിൽ പൂർണ സംഖ്യകളും name എന്ന നിരയിൽ പരമാവധി 20 അക്ഷരങ്ങളും വിദ്യാർഥി ആൺകുട്ടിയാണോ പെൺകുട്ടിയാണോ എന്ന് സൂചിപ്പിക്കുന്നതിനുള്ള ഒരു അക്ഷരം gender ലും സൂചിപ്പിക്കുന്നു. course എന്ന നിരയിൽ വിദ്യാർഥിയുടെ

കോഴ്സും, `f_income` എന്നത് കുടുംബത്തിന്റെ വരുമാനം സംഭരിക്കുന്നതിനുള്ള നിരയും ആണ്. `name`, `course` എന്നീ നിരകൾ സ്ട്രിംഗ് സംഭരിക്കുന്നതിനുള്ള സ്ഥലം കരുതി വയ്ക്കും. ഒരു പട്ടിക നിർമ്മിച്ച് അതിലേക്ക് റെക്കോർഡുകൾ ചേർക്കുമ്പോൾ ചില നിരകളിൽ പ്രത്യേക നിബന്ധനകൾ പാലിക്കേണ്ടത് ആവശ്യമാണ്. ചില നിരകളിൽ ശൂന്യമാകാതിരിക്കുക, മറ്റു ചില നിരകളിൽ ആവർത്തനങ്ങൾ ഒഴിവാക്കുക തുടങ്ങിയവയാകാം ഈ നിബന്ധനകൾ. പട്ടികയുടെ നിർമ്മാണ വേളയിൽത്തന്നെ അത് സംബന്ധിച്ച അറിവുണ്ടായിരിക്കണം. Constraint എന്നു വിളിക്കുന്ന ചില കീവേർഡുകൾ ഈ ആവശ്യത്തിനായി MySQL ൽ ലഭ്യമാക്കിയിട്ടുണ്ട്.

9.4.2 കൺസ്ട്രെയിന്റുകൾ (Constraints)

ഒരു പട്ടികയിലെ നിരകളിൽ ചേർക്കേണ്ട ഡാറ്റകളിൽ നടപ്പിൽ വരുത്തുന്ന നിയമങ്ങളാണ് കൺസ്ട്രെയിന്റുകൾ. ഒരു പട്ടിക നിർമ്മിക്കുമ്പോൾ ഫീൽഡുകളിലേക്ക് നാം നൽകുന്ന വിലകളിന്മേൽ കൺസ്ട്രയിന്റുകൾ നമുക്ക് പ്രയോഗിക്കാം. ഇത് ഒരു നിരയുടെ നിർവചനത്തിൽ നൽകിയാൽ ഈ നിബന്ധന ലംഘിക്കുന്ന ഒരു വിലയും MySQL സ്വീകരിക്കില്ല. ഇത് ഡാറ്റാബേസിലെ ഡാറ്റയുടെ കൃത്യതയും, വിശ്വാസ്യതയും ഉറപ്പുവരുത്തുന്നു. ഈ നിബന്ധനകൾ ഡാറ്റാബേസിന്റെ സമ്പൂർണത ഉറപ്പുവരുത്തുന്നതിനാൽ അവയെ ഡാറ്റാബേസ് സമ്പൂർണതാ കൺസ്ട്രെയിന്റുകൾ (Integrity Constraints) എന്നു വിളിക്കുന്നു. കൺസ്ട്രെയിന്റുകൾ നിരയുടെ തലത്തിലോ പട്ടികയുടെ തലത്തിലോ ഉണ്ടാക്കാവുന്നതാണ്.

a. നിരകളിൻമേലുള്ള കൺസ്ട്രെയിന്റുകൾ

നിരകളിൻമേലുള്ള കൺസ്ട്രെയിന്റുകൾ ഓരോ നിരയിലും പ്രത്യേകമായി ഉപയോഗിക്കുന്നു. അവ ഒരു നിരയുടെ ഡാറ്റ ഇനം നൽകിയതിനുശേഷമാണ് എഴുതുക. താഴെ പറയുന്നവയാണ് നിരകളുടെ തലത്തിലുള്ള കൺസ്ട്രെയിന്റുകൾ.

i) NOT NULL

ഒരു നിരയിൽ മൂല്യം ശൂന്യമാക്കരുത് എന്നാണ് ഇത് പ്രസ്താവിക്കുന്നത്. NULL എന്നത് SQL ലെ ശൂന്യ സ്ഥലത്തെ സൂചിപ്പിക്കുന്ന കീവേർഡാണ്. അത് വിട്ടഭാഗമോ, പൂജ്യമോ അല്ല എന്ന് പ്രധാനമായും ഓർമ്മിക്കേണ്ടതാണ്. അത് തികച്ചും വ്യത്യസ്തമായ ഒന്നാണ്. വിട്ടഭാഗം മറ്റൊരു വിട്ടഭാഗത്തിനും, പൂജ്യം മറ്റൊരു പൂജ്യത്തിനും തുല്യമാണെങ്കിൽ NULL ഒരിക്കലും മറ്റൊരു NULL ന് തുല്യമല്ല. രണ്ടു NULL മൂല്യങ്ങളെ കൂട്ടാനോ, കുറയ്ക്കാനോ, താരതമ്യം ചെയ്യാനോ സാധ്യമല്ല.

ii) AUTO_INCREMENT

MySQL ലെ കീവേഡായ AUTO_INCREMENT ഉപയോഗിക്കുന്നത് സ്വയം വർധന നടപ്പിൽ വരുത്തുന്നതിനാണ്. AUTO_INCREMENT എന്ന കൺസ്ട്രയിന്റ് ഉള്ള ഒരു നിരയിൽ മൂല്യം നൽകിയിട്ടില്ലെങ്കിൽ MySQL സീരിയൽ നമ്പറുകൾ സ്വയമായി ക്രമീകരിക്കുകയും പുതിയ നിരയുടെ അനുബന്ധ റെക്കോർഡിൽ പുതുതായി നിശ്ചയിച്ചിട്ടുള്ള മൂല്യങ്ങൾ ചേർക്കുകയും ചെയ്യുന്നു. AUTO_INCREMENT ന്റെ തനത് മൂല്യം 1 ആണ്. അതിനാൽ

ഓരോ റെക്കോർഡിനും 1 വച്ച് മൂല്യം വർധിച്ചുകൊണ്ടേയിരിക്കും. ഓരോ നിരയിലേക്കും NULL മൂല്യം കൊടുക്കുകയാണെങ്കിൽ ഈ സവിശേഷ സ്വഭാവം അത് കാണിക്കും. AUTO_INCREMENT സവിശേഷത ഓരോ പുതിയ വരിക്കും ഒരു അനന്യമായ ID (Unique ID) നിശ്ചയിക്കുന്നത് എളുപ്പമാക്കുന്നു. കാരണം MySQL നമുക്കായി മൂല്യങ്ങൾ ഉണ്ടാക്കുന്നു. AUTO_INCREMENT നൽകിയിട്ടുള്ള നിര പട്ടികയുടെ പ്രാഥമിക കീ ആയി നിർവചിക്കേണ്ടതുണ്ട്. ഒരു പട്ടികയിലെ ഒരു നിരയിൽ മാത്രമേ AUTO_INCREMENT അനുവദിക്കൂ.

iii) UNIQUE

ഈ കൺസ്ട്രയിന്റ് ഉപയോഗിക്കുന്ന നിരയിലെ രണ്ടു വരികളിൽ ഒരേ മൂല്യം ഉണ്ടായിരിക്കുകയില്ല.

iv) PRIMARY KEY

ഈ കൺസ്ട്രയിന്റ് പട്ടികയുടെ പ്രൈമറി കീയായി ഒരു നിരയെ മാറ്റുന്നു. ഈ കൺസ്ട്രയിന്റ് UNIQUE ന് സമാനമാണ്. ഒരു നിരയിൽ അല്ലെങ്കിൽ നിരകളുടെ സംയോജനത്തിൽ കൺസ്ട്രയിന്റ് ഉപയോഗിക്കാൻ സാധിക്കും. പ്രൈമറി കീയിൽ NULL മൂല്യങ്ങൾ ഉണ്ടായിരിക്കരുത്. മറ്റൊരു തരത്തിൽ പറഞ്ഞാൽ ഇത് UNIQUE, NOT NULL എന്നീ കൺസ്ട്രെയിന്റുകളുടെ സംയോജനമായി കണക്കാക്കാം. പ്രൈമറി കീ നടപ്പിൽ വരുത്തുന്നതിന് ആ നിര അനന്യമായതും, ശൂന്യമല്ലാത്തതും ആയിരിക്കണം.

v) DEFAULT

ഈ കൺസ്ട്രയിന്റ് ഉപയോഗിച്ച് ഒരു നിരയ്ക്ക് തനത് മൂല്യം നൽകാവുന്നതാണ്. ഉപയോക്താവ് ഒരു നിരയ്ക്ക് മൂല്യം നൽകുന്നില്ലെങ്കിൽ ഈ കൺസ്ട്രയിന്റ് ഉപയോഗിച്ച് തനതു മൂല്യം കൊടുക്കുന്നതാണ്.

ഇനി നമുക്ക് student പട്ടികയ്ക്ക് ചില കൺസ്ട്രയിന്റുകൾ നൽകാം. ഇതിന് താഴെ പറയുന്ന രീതിയിൽ ക്വറി 9.1 പരിഷ്കരിക്കാം (ക്വറി 9.2 കാണുക).

ക്വറി 9.2

```
CREATE TABLE student
(adm_no INT PRIMARY KEY AUTO_INCREMENT,
name VARCHAR(20) NOT NULL,
gender CHAR DEFAULT 'M',
dob DATE,
course VARCHAR(15)
f_income INT);
```

ക്വറി 9.2 ൽ PRIMARY KEY, AUTO_INCREMENT എന്നി കൺസ്ട്രയിന്റുകൾ adm_no എന്ന നിരയിൽ കൊടുത്തിട്ടുണ്ട്. അതിനാൽ ഈ നിരയിൽ ആവർത്തിച്ചുവരുന്ന ഡാറ്റകൾ

സ്വീകരിക്കപ്പെടുകയില്ല. ഈ നിരയിലേക്ക് നാം മൂല്യങ്ങളൊന്നും നൽകിയില്ലെങ്കിൽ MySQL തന്നെ ഡാറ്റ യാന്ത്രികമായി സൃഷ്ടിക്കും. NOT NULL കൺസ്ട്രയിന്റ് name എന്ന നിരയിൽ നൽകിയിരിക്കുന്നതിനാൽ ആ നിര ശൂന്യമായിരിക്കാൻ അനുവദിക്കുന്നതല്ല. അതായത് ആ നിരയിൽ ഒരു ഡാറ്റ നിർബന്ധമാണ് അതുപോലെ gender എന്ന നിരയിൽ 'M' എന്ന് നാം നൽകിയില്ലെങ്കിൽ തനതു വിലയായി 'M' സംഭരിക്കപ്പെടും

b. പട്ടികയുടെ തലത്തിലുള്ള കൺസ്ട്രയിന്റുകൾ (Table Constraints)

പട്ടികയുടെ തലത്തിലുള്ള കൺസ്ട്രയിന്റുകൾ നിരയുടെ തലത്തിലുള്ള കൺസ്ട്രയിന്റുകൾക്ക് സമാനമാണ്. പട്ടികയുടെ തലത്തിലുള്ള കൺസ്ട്രയിന്റുകൾ ഒന്നിലധികം നിരകളിൽ ഉപയോഗിക്കാൻ സാധിക്കും എന്നതാണ് പ്രധാന വ്യത്യാസം. ഒരു കൺസ്ട്രയിന്റ് ഏതാനും നിരകളിൽ പ്രയോഗിച്ചാൽ അതിനെ പട്ടികയുടെ തലത്തിലുള്ള കൺസ്ട്രയിന്റ് എന്നു പറയുന്നു. പട്ടികയുടെ തലത്തിലുള്ള കൺസ്ട്രയിന്റ് ഒരു പട്ടികയുടെ നിർവചനത്തിൽ അവസാനമാണ് കൊടുക്കേണ്ടത്. ഉദാഹരണത്തിന് ക്വറി 9.3, stock എന്ന പട്ടിക നിർമിക്കുന്നു. icode, iname എന്നീ നിരകൾക്ക് UNIQUE എന്ന് സംയുക്തമായി നൽകിയിരിക്കുന്നു.

ക്വറി 9.3

```
CREATE TABLE stock
(icode CHAR(2) PRIMARY KEY AUTO_INCREMENT,
iname VARCHAR(30) NOT NULL,
dt_purchase DATE,
rate DECIMAL(10,2),
qty INT,
UNIQUE (icode, iname));
```

ക്വറി 9.3 icode, iname എന്നീ നിരകളിൽ ഒരേ ഡാറ്റ വന്നിട്ടില്ല എന്ന് ഉറപ്പു വരുത്തുന്നു.

9.4.3 ഒരു പട്ടികയുടെ ഘടന മനസ്സിലാക്കുന്നു (Viewing the structure of a table)

student, stock എന്നീ രണ്ടു പട്ടികകൾ നാം നിർമ്മിച്ചല്ലോ. ഒരു പട്ടിക നിർമ്മിച്ച ശേഷം അതിന്റെ ഘടന എങ്ങനെയാണ് കാണുന്നത്? DESCRIBE കമാൻഡ് ഉപയോഗിച്ച് ഒരു പട്ടികയുടെ ഘടന മനസ്സിലാക്കാൻ സാധിക്കും. ഇതിന്റെ വാക്യഘടന താഴെ കൊടുത്തിരിക്കുന്നു.

```
DESCRIBE <table_name>;
```

OR

```
DESC <table_name>;
```

student എന്ന പട്ടികയുടെ ഘടന കാണുന്നതിനുള്ള കമാൻഡാണ്

```
DESC student;
```

ചിത്രം 9.4 ഈ കമാൻഡിന്റെ ഔട്ട്പുട്ട് കാണിച്ചു തരുന്നു.

```
+-----------+-------------+------+-----+---------+----------------+
| Field     | Type        | Null | Key | Default | Extra          |
+-----------+-------------+------+-----+---------+----------------+
| adm_no    | int(11)     | NO   | PRI | NULL    | auto_increment |
| name      | varchar(20) | NO   |     | NULL    |                |
| gender    | char(1)     | YES  |     | M       |                |
| dob       | date        | YES  |     | NULL    |                |
| course    | varchar(15) | YES  |     | NULL    |                |
| f_income  | int(11)     | YES  |     | NULL    |                |
+-----------+-------------+------+-----+---------+----------------+
6 rows in set (0.02 sec)
```

ചിത്രം 9.4: *student* പട്ടികയുടെ ഘടന

```
+--------------------+
| Tables_in_school   |
+--------------------+
| student            |
+--------------------+
1 row in set (0.00 sec)
```

ചിത്രം 9.5: SHOW TABLES ന്റെ ഔട്ട്പുട്ട്

ക്വറി 9.2 ൽ adm_no, f_income എന്നീ രണ്ടു നിരകളുടെ വലുപ്പം പറഞ്ഞിട്ടില്ല എന്നത് ശ്രദ്ധിക്കുക. എന്നാൽ MySQL ഈ രണ്ടു നിരകളുടെ വലുപ്പം 11 ആയി സ്വീകരിക്കും adm_no എന്ന നിര PRIMARY KEY, AUTO_INCREMENT ആയി പ്രസ്താവിച്ചിട്ടുണ്ട്. അതിനാൽ ഈ നിരകളിൽ ശൂന്യമോ ആവർത്തിച്ചുള്ള മൂല്യങ്ങളോ വരുവാൻ അനുവദിക്കില്ല. ഈ നിരയിലേക്ക് മൂല്യങ്ങൾ നൽകാതിരുന്നാൽ ബന്ധപ്പെട്ട നിരയുടെ മൂല്യം 1 വർധിപ്പിച്ചു കൊണ്ട് പുതിയ മൂല്യം നിർമ്മിക്കും. ആദ്യ റെക്കോർഡിൽ ഈ നിരയുടെ മൂല്യം ഇല്ലാതിരുന്നാൽ MySQL തനതുമൂല്യമായി 1 നൽകുന്നു. gender എന്ന നിരയിലേക്ക് ഒരു സ്ഥിര മൂല്യം 'M' എന്ന് നൽകിയിരിക്കുന്നതായി നമുക്ക് കാണാൻ സാധിക്കും. ചിത്രം 9.4 ഈ സവിശേഷത വിവരിക്കുന്നു. നിലവിലുള്ള ഡാറ്റാബേസിലെ പട്ടികകൾ കാണുന്നതിന് SHOW TABLES എന്ന കമാൻഡ് ഉപയോഗിക്കാം (ചിത്രം 9.5).

നമുക്ക് ചെയ്യാം

ചിത്രം 9.4 ഉപയോഗിച്ച് stock എന്ന പട്ടികയുടെ ഘടന എഴുതുക. നിങ്ങളുടെ ഉത്തരം DESC stock; എന്ന കമാൻഡ് ഉപയോഗിച്ച് പരിശോധിക്കുക.

നിങ്ങളുടെ പുരോഗതി അറിയുക

1. ഒരു പട്ടികയുടെ ഘടന കാണുന്നതിനു വേണ്ടി താഴെ പറയുന്ന ഏതു കമാൻഡാണ് ഉപയോഗിക്കേണ്ടത് ?

   a. LIST b. SHOW c. DESCRIBE d. STRUCT

2. CREATE TABLE കമാൻഡിന്റെ വാക്യഘടന എഴുതുക.
3. നിരകളിൽ ഉപയോഗിക്കുന്ന കൺസ്ട്രയിന്റുകളുടെ പേരെഴുതുക.
4. PRIMARY KEY, UNIQUE എന്നീ കൺസ്ട്രയിന്റുകൾ തമ്മിലുള്ള വ്യത്യാസം എഴുതുക.

5. AUTO_INCREMENT എന്ന കൺസ്ട്രയിന്റിന്റെ സവിശേഷതകൾ എന്തൊക്കെ?
6. ഒരു പട്ടികയ്ക്ക് പേര് നൽകുന്നതിനുള്ള നിയമങ്ങൾ എഴുതുക.
7. ഒരു പട്ടികയുടെ PRIMARY KEY ആയി എത്ര നിരകൾ പ്രസ്താവിക്കാം?

9.5 ഒരു പട്ടികയിൽ ഡാറ്റ ചേർക്കുന്ന രീതി (Inserting data into tables)

ഒരു ഡാറ്റാബേസും അതിന്റെ പട്ടികകളും നാം നിർമ്മിച്ചു കഴിഞ്ഞു. ഈ പട്ടികയിൽ ഏതാനും റെക്കോർഡുകൾ ചേർക്കുന്നത് എങ്ങനെയെന്ന് നമുക്ക് ചർച്ച ചെയ്യാം. പട്ടികയിൽ ഡാറ്റ ചേർക്കുന്നതിന് INSERT INTO എന്ന DML കമാൻഡ് ഉപയോഗിക്കുന്നു. അതിന്റെ വാക്യഘടന താഴെ കൊടുക്കുന്നു.

```
INSERT INTO <table_name> [<column1>,<column2>,...,<columnN>]
VALUES(<value1>,<value2>,...,<valueN>);
```

ഇവിടെ <table_name> എന്നത് വരികൾ ഉൾപ്പെടുത്തേണ്ട പട്ടികയുടെ പേരാണ്. <column1>,<column2>,........,<column N> എന്നത് മൂല്യങ്ങൾ ചേർക്കേണ്ട നിരകളാണ്. <value1>,<value2>,....<value N> എന്നത് നിർദേശിച്ചിട്ടുള്ള നിരകളിൽ ചേർക്കേണ്ട വിലകളാണ്.

ഉദാഹരണത്തിന് *student* എന്ന പട്ടികയുടെ ഡാറ്റകളായ *1001, 'Alok', 'M', 1998/10/2, 'Science', 24000* എന്നിവ പുതിയ റെക്കോർഡായി adm_no, name, gender, dob, course, f_income എന്നീ നിരകളിലേക്ക് യഥാക്രമം ചേർക്കാം. (ക്വറി 9.4 ഉപയോഗിക്കുക)

ക്വറി 9.4

```
INSERT INTO student
VALUES (1001,'Alok','M','1998/10/2', 'Science',
  24000);
```

കമ്പ്യൂട്ടറിന്റെ പ്രതികരണം ഇതായിരിക്കും.

```
Query OK, 1 row affected (0.05 sec)
```

ഓരോ നിരയിലും ഓരോ മൂല്യങ്ങൾ നൽകി INSERT INTO കമാൻഡ് student എന്ന പട്ടികയിൽ ഒരു പുതിയ വരി ചേർക്കുന്നു. പട്ടികയിൽ ഒരു വരി ചേർക്കുമ്പോൾ എല്ലാ നിരയിലും മൂല്യങ്ങൾ ഉണ്ടെങ്കിൽ ക്വറിയിൽ നിരയുടെ പേര് പരാമർശിക്കേണ്ട ആവശ്യമില്ല. പക്ഷേ, മൂല്യങ്ങളുടെ ക്രമം പട്ടികയിലെ നിരകളുടെ ക്രമത്തിനനുസരിച്ചാണ് എന്ന് ഉറപ്പ് വരുത്തേണ്ടതാണ്. ഇനി ക്വറി 9.5ൽ കാണിച്ചിരിക്കുന്നതുപോലെ ചില മാറ്റങ്ങൾ വരുത്തി മറ്റൊരു വരി ചേർക്കാം.

ക്വറി 9.5

```
INSERT INTO student (name, dob, course, f_income)
VALUES ('Nike','1998/11/26','Science',35000);
```

ഈ പ്രസ്താവനയുടെ പ്രതികരണം ഇതായിരിക്കും.

```
Query OK, 1 row affected (0.01 sec)
```

ക്വറി 9.5ൽ adm_no,gender എന്നീ നിരകൾക്ക് വില നൽകിയിട്ടില്ല. ആദ്യത്തെ കുട്ടിയുടെ അഡ്മിഷൻ നമ്പർ 1001 ആയിരിക്കെ അടുത്ത കുട്ടിയുടെ അഡ്മിഷൻ നമ്പർ 1002 ആയിരിക്കും. gender ന് തനതു മൂല്യം 'M' എന്ന് നിശ്ചയിച്ചിട്ടുണ്ട്.

പട്ടികയിൽ ഡാറ്റ ചേർക്കുമ്പോൾ താഴെ പറയുന്ന കാര്യങ്ങൾ ഓർമ്മിക്കണം:

- ഒരു പുതിയ വരി ചേർക്കുമ്പോൾ ആ നിരയുടെ ഡാറ്റ ഇനം, ആ നിരയിലെ ഡാറ്റയുമായി യോജിക്കുന്നുണ്ടോ എന്ന് ഉറപ്പ് വരുത്തുക.
- ഒരു പട്ടികയ്ക്ക് നിശ്ചയിച്ച സമ്പൂർണതാ കൺസ്ട്രയിന്റ് ആ പട്ടിക പിന്തുടരുന്നുന്നുണ്ടോ എന്ന് ഉറപ്പ് വരുത്തുക.
- CHAR അല്ലെങ്കിൽ VARCHAR എന്നീ തരത്തിലുള്ള ഡാറ്റകൾ ഒറ്റ ഉദ്ധരണികൾക്കുള്ളിലോ ഇരട്ട ഉദ്ധരണികൾക്കുള്ളിലോ എഴുതണം.
- DATE തരത്തിലുള്ള നിരകൾക്ക് വില നൽകേണ്ടത് ഒറ്റ ഉദ്ധരണികൾക്കുള്ളിലാണ്. ഈ സ്ട്രിംഗ് ആന്തരികമായി DATE എന്ന ഡാറ്റ ഇനത്തിലേക്ക് മാറ്റം ചെയ്യപ്പെടുന്നു.
- NULL മൂല്യങ്ങളെ NULL ആയി പ്രസ്താവിക്കണം (ഉദ്ധരണികൾ ഇല്ലാതെ).
- എല്ലാ നിരകൾക്കും ഡാറ്റ ലഭ്യമല്ലെങ്കിൽ പട്ടികയുടെ പേരിന് ശേഷം നിരകളുടെ ലിസ്റ്റ് ഉൾപ്പെടുത്തണം.

ഒരൊറ്റ INSERT INTO കമാൻഡ് ഉപയോഗിച്ച് പല വരികൾ ഒരു പട്ടികയിൽ ചേർക്കാൻ MySQL അനുവദിക്കുന്നു. പൊതുവായ വാക്യഘടന താഴെ കൊടുക്കുന്നു.

```
INSERT INTO <table-name> VALUES(...), (...), ... ;
```

പട്ടിക 9.1 ൽ രണ്ട് റെക്കോർഡുകൾ കൂടി ക്വറി 9.6 ഉപയോഗിച്ച് നമുക്ക് ചേർക്കാം

ക്വറി 9.6

```
INSERT INTO student (name, dob, course, f_income)
VALUES('Bharath','1999/01/01','Commerce',45000),
      ('Virat','1998/12/05','Science',22000);
```

കമ്പ്യൂട്ടറിന്റെ പ്രതികരണം ഇതായിരിക്കും.

```
Query OK, 2 rows affected (0.02 sec)
Records: 2 Duplicates: 0 Warnings: 0
```

ഇവിടെ രണ്ട് റെക്കോർഡുകൾ ആവരണചിഹ്നത്തിനകത്തായി വെവ്വേറെ നൽകിയിട്ടുണ്ടെന്ന് നമുക്ക് കാണാം. രണ്ടു റെക്കോർഡുകൾ വിജയകരമായി ചേർത്തിട്ടുണ്ടെന്ന് പ്രതികരണം സൂചിപ്പിക്കുന്നു. ഒരു വിദ്യാർഥിയുടെ മാസവരുമാനത്തിന്റെ ഡാറ്റ നമുക്ക് ഇല്ല എന്ന് കരുതുക. ഈ റെക്കോർഡ് എങ്ങനെ ചേർക്കും? ക്വറി 9.7 ഇത് വിവരിക്കുന്നു.

ക്വറി 9.7

```
INSERT INTO student(name, dob, gender, course)
VALUES('Meera','1998/08/15','F','Science');
```

ഈ ക്വറിക്ക് മറുപടിയായി adm_no ന്റെ മൂല്യം കമ്പ്യൂട്ടർ തന്നെ ചേർക്കുന്നു. എന്നാൽ f_income എന്ന നിരയുടെ വില NULL ആയിരിക്കും. നിരയുടെ ക്രമവും ഈ ക്വറിയിൽ

മാറിയിട്ടുണ്ട്. ഒരു വരിയിലെ മൂല്യങ്ങളുടെ അഭാവത്തെ ക്വറി 9.8 ൽ കാണിച്ചിരിക്കുന്നതു പോലെ മറ്റൊരു തരത്തിൽ കൈകാര്യം ചെയ്യാം.

ക്വറി 9.8

```
INSERT INTO student(name, dob, gender, course, f_income)
VALUES('Divakar','1998/02/21','M' Science',NULL);
```

VALUES ഉപവാക്യത്തിൽ f_income ത്തിന് NULL കൊടുത്തിരിക്കുന്നത് ശ്രദ്ധിക്കുക.

നമുക്ക് ചെയ്യാം

student എന്ന പട്ടികയിൽ ഏതാനും റെക്കോർഡുകൾ കൂടി നമുക്ക് ചേർക്കാം. പട്ടിക 9.5 കാണിച്ചിരിക്കുന്നതു പോലെ വിദ്യാർഥികളുടെ വിശദാംശങ്ങൾ ചേർക്കുന്നതിന് ആവശ്യമായ ക്വറികൾ എഴുതുക.

adm_no	name	gender	dob	course	f_income
1025	Kaushi	M	1998/10/2	Commerce	17000
1026	Niveditha	F	1999/03/04	Humanities	52000
1027	Sreekumar	M	1998/06/06	Science	15000
1057	Chaithanya	F	1999/06/03	Science	

*പട്ടിക 9.5: **student** പട്ടികയിൽ കൂടുതൽ റെക്കോർഡുകൾ*

നിങ്ങളുടെ പുരോഗതി അറിയുക

1. ഒരു പട്ടികയിൽ ഒരു വരി ചേർക്കുന്നതിന് താഴെ കൊടുത്തിരിക്കുന്നതിൽ ഏത് കമാൻഡ് ഉപയോഗിക്കുന്നു ?

   a. ADD b. CREATE c. INSERT d. MAKE

2. ഒരു പട്ടികയിൽ പുതിയ ഡാറ്റ ചേർക്കുന്നതിനുള്ള പ്രസ്താവന ഏത്?

   a. ADD RECORD b. INSERT RECORD

   c. INSERT INTO d. INSERT ROW

3. INSERT കമാൻഡിന്റെ കൂടെ ഉപയോഗിക്കുന്ന കീവേർഡുകൾ എഴുതുക.

നമുക്ക് ചെയ്യാം

ക്വറി 9.3 ഉപയോഗിച്ച് ഉണ്ടാക്കിയ stock എന്ന പട്ടികയിൽ കൂടുതൽ റെക്കോർഡുകൾ ചേർക്കുന്നതിനുള്ള SQL കമാൻഡുകൾ എഴുതുക. നിരയിൽ വിലകൾ നൽകുമ്പോൾ AUTO_INCREMENT, UNIQUE എന്നീ കൺസ്ട്രയിന്റുകളുടെ സൗകര്യം ഉപയോഗിക്കുക.

9.6 പട്ടികയിൽ നിന്ന് വിവരങ്ങൾ വീണ്ടെടുക്കുന്ന രീതി (Retrieving information from tables)

school എന്ന ഡാറ്റാബേസും student എന്ന പട്ടികയും നിർമ്മിച്ച് അതിൽ 10 റെക്കോർഡുകൾ നാം ചേർത്തുവല്ലോ. ഈ പട്ടികകളിൽ ശേഖരിച്ച ഡാറ്റ എങ്ങനെ തിരിച്ചെടുക്കാമെന്ന് നമുക്ക് പഠിക്കാം. അത് ഒരു ഡാറ്റ കൈകാര്യം ചെയ്യൽ പ്രക്രിയയാണ്. ഈ ആവശ്യത്തിനായി SQL ലിൽ SELECT കമാൻഡ് ലഭ്യമാക്കിയിട്ടുണ്ട്. ഒരു പട്ടികയിലെ നിർദിഷ്ട നിരയിൽ നിന്ന് വിവരങ്ങൾ വീണ്ടെടുക്കാൻ ഇത് ഉപയോഗിക്കുന്നു. SELECT കമാൻഡിന് നിരവധി രൂപങ്ങൾ ഉണ്ട്. SELECT കമാൻഡിന്റെ ലളിതമായ രൂപമാണ് താഴെകൊടുത്തിരിക്കുന്നത്.

```
SELECT <column_name>[,<column_name>,<column_name>, ...]
FROM <table_name>;
```

ഇവിടെ <column_name> എന്നത്, ഏത് നിരയിൽ നിന്നാണ് ഡാറ്റ വീണ്ടെടുക്കേണ്ടത് എന്നും <table_name> എന്നത്, ഏത് പട്ടികയിൽ നിന്നാണ് വിവരങ്ങൾ വീണ്ടെടുക്കേണ്ടത് എന്നുമാണ് സൂചിപ്പിക്കുന്നത്. പട്ടികയുടെ പേര് നൽകുന്നത് FROM എന്ന കീവേർഡ് ഉപയോഗിച്ചാണ്. ഇത് SELECT കമാൻഡിന്റെ കൂടെ നിർബന്ധമായും ഉപയോഗിക്കേണ്ട ഉപവാക്യമാണ് (Clause). SELECT കമാൻഡിൽ കൊടുത്തിരിക്കുന്ന ക്രമത്തിൽ തന്നെ നിരകളായി ഡാറ്റ പ്രദർശിപ്പിക്കപ്പെടും.

ഇപ്പോൾ നമുക്ക് SELECT കമാൻഡിന്റെ നിർവഹണം വിവിധ ക്വറികൾ വഴി പരിചയപ്പെടാം. ക്വറി 9.9 പ്രവർത്തിക്കുമ്പോൾ ചിത്രം 9.6 ൽ കാണിച്ചിരിക്കുന്നതു പോലെ student പട്ടികയിലെ വിദ്യാർഥികളുടെ name, course എന്നിവ നമുക്ക് ലഭിക്കുന്നു.

ക്വറി 9.9

```
SELECT name, course
FROM student;
```

```
+------------+------------+
| name       | course     |
+------------+------------+
| Alok       | Science    |
| Nike       | Science    |
| Bharath    | Commerce   |
| Virat      | Science    |
| Meera      | Science    |
| Divakar    | Science    |
| Kaushi     | Commerce   |
| Niveditha  | Humanities |
| Sreekumar  | Science    |
| Chaithanya | Science    |
+------------+------------+
10 rows in set (0.00 sec)
```

ചിത്രം 9.6: Query 9.9 ന്റെ ഔട്ട്പുട്ട്

പട്ടിക മുഴുവനും നമുക്ക് പ്രദർശിപ്പിക്കണമെങ്കിൽ ആ റിലേഷനിലെ നിരകളുടെ ലിസ്റ്റ് മുഴുവൻ നൽകേണ്ടതില്ല, പകരം നിരകളുടെ പൂർണമായ ലിസ്റ്റ് ഉൾപ്പെടുത്തുന്നതിന് നക്ഷത്ര ചിഹ്നം (*) ഉപയോഗിക്കാം (ക്വറി 9.10). ഇതിന്റെ ഔട്ട്പുട്ട് ചിത്രം 9.7 ൽ കൊടുത്തിരിക്കുന്നു.

ക്വറി 9.10

```
SELECT * FROM student;
```

```
+---------+------------+--------+------------+------------+----------+
| adm_no  | name       | gender | dob        | course     | f_income |
+---------+------------+--------+------------+------------+----------+
|    1001 | Alok       | M      | 1998-10-02 | Science    |    24000 |
|    1002 | Nike       | M      | 1998-11-26 | Science    |    35000 |
|    1003 | Bharath    | M      | 1999-01-01 | Commerce   |    45000 |
|    1004 | Virat      | M      | 1998-12-05 | Science    |    22000 |
|    1005 | Meera      | F      | 1998-08-15 | Science    |     NULL |
|    1006 | Divakar    | M      | 1998-02-21 | Science    |     NULL |
|    1025 | Kaushi     | M      | 1998-10-02 | Commerce   |    17000 |
|    1026 | Niveditha  | F      | 1999-03-04 | Humanities |    52000 |
|    1027 | Sreekumar  | M      | 1998-06-06 | Science    |    15000 |
|    1057 | Chaithanya | F      | 1999-06-03 | Science    |     NULL |
+---------+------------+--------+------------+------------+----------+
10 rows in set (0.00 sec)
```

ചിത്രം 9.7: *student* പട്ടികയുടെ മുഴുവൻ ഉള്ളടക്കം

ചിത്രം 9.7 ൽ `f_income` എന്ന നിരയിലെ ചില വരികളിൽ NULL മൂല്യങ്ങൾ നമുക്ക് കാണാൻ സാധിക്കും. വരിയിലേക്ക് മൂല്യങ്ങൾ ചേർക്കുമ്പോൾ വിലകൾ വിട്ടുപോയ താണ് ഇതിന് കാരണം. (ക്വറി 9.7, 9.8 പട്ടിക 9.1, 9.5 പരിശോധിക്കുക).

9.6.1 DISTINCT ഉപയോഗിച്ച് നിരകളിലെ ആവർത്തനങ്ങൾ ഉള്ള മൂല്യങ്ങൾ നീക്കം ചെയ്യുന്ന രീതി (Eliminating duplicate values in columns using DISTINCT)

student എന്ന പട്ടികയിലെ വിവിധ കോഴ്സുകളുടെ പേരുകൾ നമുക്ക് അറിയാമെന്ന് കരുതുക, `SELECT course FROM student;` എന്ന ക്വറി നിർമ്മിക്കുകയാണെങ്കിൽ അത് `course` എന്ന നിരയിലെ എല്ലാ മൂല്യങ്ങളെയും ചിത്രം 9.7 ൽ കാണിച്ചിരിക്കുന്ന പോലെ പ്രദർശിപ്പിക്കും. റിലേഷനിലെ എല്ലാ വരികളിൽ നിന്നുമുള്ള ഡാറ്റ ഒന്നിൽ കൂടുതൽ തവണ ആവർത്തിക്കുന്നുണ്ടെങ്കിലും അവ പ്രദർശിപ്പിക്കും. ഈ ആവർത്തനം DISTINCT എന്ന കീവേർഡ് ഉപയോഗിച്ച് നമുക്ക് ഒഴിവാക്കാൻ സാധിക്കും (ക്വറി 9.11). ചിത്രം 9.8 ൽ ഇതിന്റെ ഔട്ട്പുട്ട് കൊടുത്തിരിക്കുന്നു.

ക്വറി 9.11

```
SELECT DISTINCT course
FROM student;
```

ഔട്ട്പുട്ടിൽ ആവർത്തനം ഇല്ല. കീവേർഡിനോടു കൂടി ഉപയോഗിച്ചിരിക്കുന്ന നിരയിൽ ഒന്നിലധികം NULL വില ഉണ്ടെങ്കിൽ അതിൽ ഒരു വില മാത്രമെ ഫലത്തിൽ കാണിക്കുകയുള്ളു. DISTINCT നു പകരം ALL എന്ന കീവേർഡ് ഉപയോഗിച്ചുവെന്നിരിക്കട്ടെ, ആ നിരയിലെ എല്ലാ ആവർത്തന മൂല്യങ്ങളും ഫലത്തിൽ അടങ്ങിയിരിക്കും. അതായത് DISTINCT, ALL എന്നിവ ഉപയോഗിക്കാത്തഅവസ്ഥയിലുള്ള അതേ ഫലം ലഭിക്കുന്നു.

```
+------------+
| course     |
+------------+
| Science    |
| Commerce   |
| Humanities |
+------------+
3 rows in set (0.25 sec)
```

ചിത്രം . 9.8: DISTINCT ന്റെ ഉപയോഗം

9.6.2 പ്രത്യേക വരികൾ WHERE ഉപവാക്യം ഉപയോഗിച്ച് തിരഞ്ഞെടുക്കുന്ന രീതി (Selecting specific rows using WHERE clause)

ചില സന്ദർഭങ്ങളിൽ നമുക്ക് പട്ടികയിലെ ഒരു ഉപഗണം മാത്രം തിരഞ്ഞെടുക്കേണ്ടതായി വരും. ഉദാഹരണമായി പെൺകുട്ടികളെക്കുറിച്ചുള്ള വിശദാംശങ്ങൾ മാത്രമോ, അല്ലെങ്കിൽ കുടുംബമാസവരുമാനം Rs 25000/- ന് താഴെയുള്ള വിദ്യാർഥികളുടെ വിശദാംശങ്ങളോ ആവശ്യമായി വന്നേക്കാം. ഇവിടെ റെക്കോർഡുകളുടെ വീണ്ടെടുക്കലിൽ ഒരു തിരഞ്ഞെടുക്കൽ ഉണ്ട്. തീർച്ചയായും ഇത് ഒരു നിബന്ധനയുടെ അടിസ്ഥാനത്തിലായിരിക്കും. SELECT കമാൻഡിന്റെ WHERE ഉപവാക്യം ഉപയോഗിച്ച് റെക്കോർഡുകളൾ വീണ്ടെടുക്കുന്നതിനുള്ള ചില മാനദണ്ഡങ്ങൾ ഉണ്ടാക്കാൻ SQL നമ്മെ സഹായിക്കുന്നു. WHERE ഉപവാക്യത്തോടുകൂടിയ SELECT കമാൻഡിന്റെ വാക്യഘടന താഴെ കൊടുക്കുന്നു.

```
SELECT <column_name>[,<column_name>,<column_name>, ...]
FROM <table_name>
WHERE <condition>;
```

WHERE ഉണ്ടെങ്കിൽ SELECT കമാൻഡ് പട്ടികയിലെ ഓരോ വരിയിൽക്കൂടിയും കടന്നു പോവുകയും നിബന്ധനകൾ പാലിക്കുന്നുണ്ടോ എന്ന് പരിശോധിക്കുകയും ചെയ്യും. ഒരു വരി നിബന്ധന പാലിക്കുന്നുണ്ടെങ്കിൽ ആ വരി ഔട്ട്പുട്ടിൽ പ്രദർശിപ്പിക്കുന്നു. ലോജിക്കൽ ഓപ്പറേറ്ററുകൾ, റിലേഷണൽ ഓപ്പറേറ്ററുകൾ എന്നിവ ഉപയോഗിച്ച് നിബന്ധനകൾ നിർമ്മിക്കാവുന്നതാണ്. MySQL ൽ ഇതിന് വ്യത്യസ്ത ഓപ്പറേറ്ററുകൾ ഉപയോഗിക്കുന്നു. ഇവ പട്ടിക 9.6 ൽ കൊടുത്തിരിക്കുന്നു.

ഓപ്പറേറ്റർ		അർഥം/ ഫലം
റിലേഷണൽ ഓപ്പറേറ്റർ	=	തുല്യമായത്
	<> or !=	തുല്യമല്ലാത്തത്
	>	വലുത്
	<	ചെറുത്
	>=	വലുതോ തുല്യമായതോ
	<=	ചെറുതോ തുല്യമായതോ
ലോജിക്കൽ ഓപ്പറേറ്റർ	NOT	നിബന്ധന തെറ്റാണെങ്കിൽ ഔട്ട്പുട്ട് ശരി
	AND	രണ്ട് നിബന്ധനയും ശരിയാണെങ്കിൽ ഔട്ട്പുട്ട് ശരി
	OR	നിബന്ധനകളിൽ ഏതെങ്കിലും ശരിയാണെങ്കിൽ ഔട്ട്പുട്ട് ശരി

പട്ടിക 9.6: വ്യവസ്ഥകൾ ക്രമപ്പെടുത്തുന്നതിന് ഉപയോഗിക്കുന്ന ഓപ്പറേറ്ററുകൾ

ഒരു പട്ടികയിലെ പെൺകുട്ടികളുടെ വിശദാംശങ്ങൾ നൽകുന്നതിനുള്ള SQL പ്രസ്താവന ഇനി എഴുതാം. ക്വറി 9.12 ൽ നിബന്ധന ക്രമപ്പെടുത്തുന്നതിനുള്ള WHERE ഉപവാക്യവും, " = " ഓപ്പറേറ്ററും അടങ്ങിയിരിക്കുന്നു. അതിന്റെ ഔട്ട്പുട്ട് ചിത്രം 9.9 കാണിച്ചിരിക്കുന്നു.

ക്വറി 9.12

```
SELECT * FROM student
WHERE gender='F';
```

```
+--------+------------+--------+------------+------------+----------+
| adm_no | name       | gender | dob        | course     | f_income |
+--------+------------+--------+------------+------------+----------+
|   1005 | Meera      | F      | 1998-08-15 | Science    |     NULL |
|   1026 | Niveditha  | F      | 1999-03-04 | Humanities |    52000 |
|   1057 | Chaithanya | F      | 1999-06-03 | Science    |     NULL |
+--------+------------+--------+------------+------------+----------+
3 rows in set (0.06 sec)
```

ചിത്രം 9. 9: വിദ്യാർഥിനികളുടെ ലിസ്റ്റ് (ക്വറി 9.12 ന്റെ ഔട്ട്പുട്ട്)

നമുക്ക് വരുമാനം 25000 രൂപയിൽ താഴെയുള്ള സയൻസ് ഗ്രൂപ്പിലെ വിദ്യാർഥികളുടെ പേര്, കോഴ്സ്, മാസ വരുമാനം എന്നിവ കാണണം എന്നു വിചാരിക്കുക. ഇവിടെ നമുക്ക് രണ്ട് നിബന്ധനകൾ നിർമ്മിക്കേണ്ടതായി വരും. ഒന്ന് വരുമാനം പരിശോധിക്കുന്നതിനും, മറ്റൊന്ന് കോഴ്സ് പരിശോധിക്കുന്നതിനുമാണ്. ഇനി മുകളിൽ പറഞ്ഞ രണ്ടു നിബന്ധനകളും പട്ടികയിലെ റെക്കോർഡുകളുമായി യോജിക്കുന്നുണ്ടെങ്കിൽ name, course, f_income എന്നീ നിരകൾ പ്രദർശിപ്പിക്കുന്നതിന് വീണ്ടെടുക്കുന്നു (ക്വറി 9.13). ഇതിന്റെ ഔട്ട്പുട്ട് ചിത്രം 9.10 ൽ കാണിച്ചിരിക്കുന്നു. രണ്ടു നിബന്ധനകൾ യോജിപ്പിക്കുന്നതിനു വേണ്ടി AND ഓപ്പറേറ്റർ ഉപയോഗിക്കുന്നു.

ക്വറി 9.13

```
SELECT name, course, f_income FROM student
WHERE course='Science' AND f_income<25000;
```

സയൻസ് ഗ്രൂപ്പിൽ ഉൾപ്പെടാത്ത (കോമേഴ്സ്, ഹ്യൂമാനിറ്റീസ് ഗ്രൂപ്പുകൾ) വിദ്യാർഥികളുടെ name, course, f_income എന്നിവ പ്രദർശിപ്പിക്കാനാവശ്യമായ ക്വറി എഴുതുക.

```
+-----------+---------+----------+
| name      | course  | f_income |
+-----------+---------+----------+
| Alok      | Science |    24000 |
| Virat     | Science |    22000 |
| Sreekumar | Science |    15000 |
+-----------+---------+----------+
3 rows in set (0.00 sec)
```

ചിത്രം 9.10: AND ഓപ്പറേറ്ററിന്റെ ഉപയോഗം

ഈ ക്വറി എഴുതുമ്പോൾ നിബന്ധന തയാറാക്കുന്നതിനായി OR ഓപ്പറേറ്റർ നിങ്ങൾ ഉപയോഗിച്ചേക്കാം. ഇതേ വിവരങ്ങൾ NOT ഓപ്പറേറ്റർ ഉപയോഗിച്ചാലും ലഭിക്കും. ക്വറി 9.14 ഇത് വിവരിക്കുന്നു. ചിത്രം 9.11 ൽ അതിന്റെ ഔട്ട്പുട്ട് പ്രദർശിപ്പിച്ചിരിക്കുന്നു

ക്വറി 9.14

```
SELECT name, course, f_income FROM student
WHERE NOT course='Science';
```

stock എന്ന പട്ടികയുമായി ബന്ധപ്പെട്ട് നമുക്ക് ചില നിബന്ധനകൾ ഉണ്ടാക്കാം, അനുയോജ്യമായ ഓപ്പറേറ്ററുകൾ ഉപയോഗിച്ച് ഇതിനുള്ള ക്വറികൾ തയാറാക്കാൻ ശ്രമിക്കാം.

SQL ൽ നിബന്ധനകൾക്കായി പ്രത്യേകം ഓപ്പറേറ്ററുകൾ ഉണ്ട്. BETWEEN.... AND, IN, LIKE, IS മുതലായവ

```
+-----------+------------+----------+
| name      | course     | f_income |
+-----------+------------+----------+
| Bharath   | Commerce   |    45000 |
| Kaushi    | Commerce   |    17000 |
| Niveditha | Humanities |    52000 |
+-----------+------------+----------+
3 rows in set (0.00 sec)
```

ചിത്രം 9.11: NOT ഓപ്പറേറ്ററിന്റെ ഉപയോഗം

ഇതിൽ ഉൾപ്പെടുന്നു. നിബന്ധനകൾ തയാറാക്കുന്നതിന് ഈ ഓപ്പറേറ്ററുകൾ സഹായിക്കുന്നത് എങ്ങനെയെന്ന് നമുക്ക് ചർച്ച ചെയ്യാം.

a. വിലകളുടെ പരിധികൾ അടിസ്ഥാനമാക്കിയുള്ള നിബന്ധനകൾ

ഒരു പരിധിയിലുള്ള മൂല്യങ്ങളെ നിബന്ധനകളായി നൽകാം. പരിധി നിർദേശിക്കുന്നതിന് SQL ൽ BETWEEN..AND എന്ന ഓപ്പറേറ്റർ ഉപയോഗിക്കാം. കുടുംബ വരുമാനം Rs 25000 മുതൽ Rs 45000 വരെയുള്ള വിദ്യാർഥികളുടെ ലിസ്റ്റ് നമുക്ക് ആവശ്യമുണ്ടെന്ന് വിചാരിക്കുക. ക്വറി 9.15 ൽ കൊടുത്തിരിക്കുന്ന പ്രസ്താവന ഉപയോഗിച്ച് ഇത് ലഭ്യമാക്കാം. ചിത്രം 9.12 ൽ ഇതിന്റെ ഔട്ട്പുട്ട് കാണിച്ചിരിക്കുന്നു.

ക്വറി 9.15

```
SELECT name, f_income FROM student
WHERE f_income>=25000 AND f_income<=45000;
```

ഇതേ ഔട്ട്പുട്ട് ക്വറി 9.16 ൽ ഉള്ള പ്രസ്താവന ഉപയോഗിച്ച് ലഭിക്കുന്നതാണ്. ഈ പ്രസ്താവനയിൽ നിബന്ധന ഉണ്ടാക്കാൻ BETWEEN... AND ഓപ്പറേറ്റർ ഉപയോഗിച്ചിട്ടുണ്ട്. പരിധിയിൽ കൊടുത്തിരിക്കുന്ന കുറഞ്ഞ വിലയും കൂടിയ വിലയും ഔട്ട്പുട്ടിൽ ഉൾപ്പെട്ടിരിക്കുന്നു

ക്വറി 9.16

```
SELECT name, f_income FROM student
WHERE f_income BETWEEN 25000 AND 45000;
```

സംഖ്യാതരത്തിലുള്ള ഡാറ്റയുടെ പരിധി BETWEEN ... AND ഓപ്പറേറ്റർ അനുവദിക്കുന്നുവെന്ന് ക്വറി 9.16 ൽ നിന്നും ചിത്രം 9.12 ൽ നിന്നും നമുക്ക് അനുമാനിക്കാൻ കഴിയും.

```
+---------+----------+
| name    | f_income |
+---------+----------+
| Nike    |    35000 |
| Bharath |    45000 |
+---------+----------+
2 rows in set (0.00 sec)
```

ചിത്രം 9.12: BETWEEN .. AND ഓപ്പറേറ്ററിന്റെ ഉപയോഗം

b. വിലകളുടെ ലിസ്റ്റ് അടിസ്ഥാനമാക്കിയുള്ള നിബന്ധനകൾ

റെക്കോർഡുകൾ വീണ്ടെടുക്കുന്നതിനു വേണ്ടിയുള്ള നിബന്ധനകൾ തയാറാക്കുമ്പോൾ വിലകളുടെ ലിസ്റ്റ് ഉപയോഗിക്കാവുന്നതാണ്. വിലകൾ ഏതു ഡാറ്റ ഇനമായാലും നിർദിഷ്ട നിരയിലെ നിബന്ധനയുമായി യോജിക്കുന്നതായിരിക്കണം. ഈ അവസരത്തിൽ ലിസ്റ്റിനോടുകൂടി IN എന്ന ഓപ്പറേറ്റർ ഉപയോഗിക്കാം. സയൻസിലെയും ഹ്യൂമാനിറ്റീസിലെയും വിദ്യാർഥികളുടെ വിശദാംശങ്ങൾ നമുക്ക് വീണ്ടെടുക്കണമെന്നിരിക്കട്ടെ, ക്വറി 9.17 ലെ പ്രസ്താവനയ്ക്ക് ചിത്രം 9.13 ൽ കാണിച്ചിരിക്കുന്നതുപോലെ വിശദാംശങ്ങൾ നിർമ്മിക്കാൻ കഴിയും.

ക്വറി 9.17

```
SELECT * FROM student
WHERE course='Commerce' OR course='Humanities';
```

adm_no	name	gender	dob	course	f_income
1003	Bharath	M	1999-01-01	Commerce	45000
1025	Kaushi	M	1998-10-02	Commerce	17000
1026	Niveditha	F	1999-03-04	Humanities	52000

```
3 rows in set (0.00 sec)
```

ചിത്രം 9.13: ക്വറി 9.18 ന്റെ ഔട്ട്പുട്ട്

ഇതേ ഔട്ട്പുട്ട് ക്വറി 9.18 ൽ കാണുന്ന രീതിയിൽ IN ഓപ്പറേറ്റർ ഉപയോഗിച്ച് താഴെക്കാണുന്ന രീതിയിൽ ലഭ്യമാക്കാം.

ക്വറി 9.18

```
SELECT * FROM student
WHERE course IN('Commerce', 'Humanities');
```

ക്വറി 9.18 ൽ നാം കാണുന്നതുപോലെ IN ഓപ്പറേറ്റർ ഒരു റെക്കോർഡിലെ നിർദിഷ്ട നിരയിലെ (ഇവിടെ course) വില തന്നിരിക്കുന്ന ലിസ്റ്റിലെ ഏതെങ്കിലും വിലയോട് തുല്യത ഉണ്ടോ എന്ന് പരിശോധിക്കുന്നു. റെക്കോർഡുകളിലൂടെ കടന്നുപോകുമ്പോൾ തുല്യ തുക കണ്ടെത്തിയാൽ അവ പ്രദർശിപ്പിക്കുന്നു. ഈ പട്ടികയിൽ മൂന്ന് കോഴ്സ് മാത്രമേ ഉള്ളൂ. ക്വറി 9.19 ഉപയോഗിക്കുമ്പോൾ ചിത്രം 9.13 ൽ കാണിച്ചിരിക്കുന്ന അതേ ഫലം കാണിക്കുന്നതുമാണ്.

ക്വറി 9.19

```
SELECT * FROM student
WHERE course NOT IN('Science');
```

c. പാറ്റേൺ ചേർച്ച അടിസ്ഥാനമാക്കിയുള്ള നിബന്ധനകൾ

ചില മാതൃകകളെ അടിസ്ഥാനമാക്കി ഡാറ്റകൾ വീണ്ടെടുക്കേണ്ടതായി വരാം. അതായത് സ്ട്രിംഗ് ഡാറ്റയിലെ അക്ഷരങ്ങളിലുള്ള സാദൃശ്യമായിരിക്കും റെക്കോർഡുകൾ വീണ്ടെടുക്കുന്നതിനുള്ള മാനദണ്ഡം. SQL ൽ ഈ ആവശ്യത്തിനായി LIKE എന്ന പാറ്റേൺ മാച്ചിങ് ഓപ്പറേറ്റർ ഉപയോഗിക്കുന്നു. പാറ്റേണുകളെ രണ്ടു വൈൽഡ് കാർഡ് അക്ഷരങ്ങളായ % ഉം _ (അണ്ടർ സ്കോർ) ഉം ഉപയോഗിച്ച് നിർദേശിക്കാം. % (പെർസെന്റ്) സബ് സ്ട്രിങ്ങിന്റെ തുല്യത പരിശോധിക്കുന്നു. _ (അണ്ടർ സ്കോർ) ഒരു അക്ഷരത്തിന്റെ തുല്യത പരിശോധിക്കുന്നു. പാറ്റേണുകൾ കേസ് സെൻസിറ്റീവ് ആണ്, അതായത് വലിയ അക്ഷരങ്ങൾ ചെറിയ അക്ഷരങ്ങളുമായി തുല്യത പ്രാപിക്കില്ല. താഴെ പറയുന്ന ഉദാഹരണങ്ങൾ പരിഗണിക്കുക.

- "Ab%" എന്നത് Ab എന്ന അക്ഷരങ്ങളിൽ തുടങ്ങുന്ന ഏതു സ്ട്രിങ്ങിനും തുല്യമാണ്.
- "%cat%" എന്നത് "cat" എന്ന വാക്ക് ഉപസ്ട്രിങ്ങുമായി വരുന്ന ഏതു സ്ട്രിങ്ങുമായും തുല്യത പാലിക്കുന്നു. ഉദാഹരണത്തിന് "education", "indication", "catering" തുടങ്ങിയവ.
- "_ _ _ _" എന്നത് ഇടയിൽ ശൂന്യ സ്ഥലം ഇല്ലാത്ത നാല് അക്ഷരങ്ങൾ മാത്രമുള്ള ഏത് സ്ട്രിങ്ങിനും തുല്യമാണ്.

- "_ _ _%" എന്നത് ഏറ്റവും കുറഞ്ഞത് മൂന്ന് അക്ഷരങ്ങൾ ഉള്ള ഏത് സ്ട്രിങ്ങുമായും തുല്യത പാലിക്കും.

ക്വറി 9.20 LIKE ഓപ്പറേറ്ററിന്റെ ഉപയോഗം ക്വറി 9.20 വിവരിക്കുന്നു. പട്ടികയിലെ 'ar' എന്ന് അവസാനിക്കുന്ന പേരുകളുടെ ഒരു ലിസ്റ്റ് അത് നൽകുന്നു ചിത്രം 9.14 ൽ ഈ ക്വറിയുടെ ഔട്ട്പുട്ട് കാണിച്ചിരിക്കുന്നു

ക്വറി 9.20

```
SELECT name FROM student
WHERE name LIKE '%ar';
```

```
+------------+
| name       |
+------------+
| Divakar    |
| Sreekumar  |
+------------+
2 rows in set (0.00 sec)
```

ചിത്രം 9.14: പാറ്റേൺ ചേർച്ചയുടെ ഉപയോഗം

ക്വറി 9.21 'Divakar' എന്ന ഔട്ട്പുട്ട് തരുന്നു. കാരണം രണ്ട് _ (അണ്ടർ സ്കോർ) അക്ഷരങ്ങളാണ് പാറ്റേണിന്റെ തുല്യത പരിശോധിക്കുന്നത്.

ക്വറി 9.21

```
SELECT name FROM student
WHERE name LIKE 'Div__ar';
```

d. NULL വില പരിശോധന അടിസ്ഥാനമാക്കിയുള്ള നിബന്ധനകൾ

റെക്കോർഡുകളിൽ ചില ഫീൽഡുകളിൽ NULL വില ഉണ്ടായിരിക്കാമെന്നു നാം കണ്ടു കഴിഞ്ഞു. ഇങ്ങനെയുള്ള റെക്കോർഡുകളെ IS ഓപ്പറേറ്ററുകളുടെ സഹായത്തോടെ വീണ്ടെടുക്കാം. റെക്കോർഡിലെ നിർദിഷ്ട നിരയിൽ NULL മൂല്യങ്ങൾ വന്നാൽ നിബന്ധനകൾ ശരിയാകും. student എന്ന പട്ടികയിൽ f_income എന്ന നിരയിൽ NULL മൂല്യം മൂന്ന് വരികളിൽ ഉണ്ട്. (ചിത്രം 9.7 പരിശോധിക്കുക) ക്വറി 9.22 ഈ റെക്കോർഡുകൾ വീണ്ടെടുക്കുന്നു. ചിത്രം 9.15 ഇതിന്റെ ഔട്ട്പുട്ട് പ്രദർശിപ്പിക്കുന്നു.

```
+-------------+---------+
| name        | course  |
+-------------+---------+
| Meera       | Science |
| Divakar     | Science |
| Chaithanya  | Science |
+-------------+---------+
3 rows in set (0.00 sec)
```

ചിത്രം 9.15: NULL വില പരിശോധന

ക്വറി 9.22

```
SELECT name,course FROM student
WHERE f_income IS NULL;
```

f_income നിരയിൽ NULL മൂല്യങ്ങൾ ഇല്ലാത്ത റെക്കോർഡുകൾ വീണ്ടെടുക്കണമെന്നിരിക്കട്ടെ, ക്വറി 9.23 അത് നിർവഹിക്കുന്നു.

ക്വറി 9.23

```
SELECT name,course FROM student
WHERE f_income IS NOT NULL ;
```

നമുക്ക് ചെയ്യാം

stock എന്ന പട്ടികയെ അടിസ്ഥാനപ്പെടുത്തി SQL ഓപ്പറേറ്ററുകൾ ഉപയോഗിക്കാനുള്ള ക്വറികൾ നിർമ്മിക്കുക.

നിങ്ങളുടെ പുരോഗതി അറിയുക

1. ഒരു നിരയിൽ ആവർത്തനങ്ങൾ ഒഴിവാക്കാൻ SELECT കമാൻഡിന്റെ കൂടെ ഉപയോഗിക്കുന്ന കീവേർഡിന്റെ പേരെഴുതുക.
2. SELECT ക്വറിയുടെ കൂടെ ഉപയോഗിക്കുന്ന പ്രധാന ഉപവാക്യം ഏത്?
3. ഒരു നിരയിൽ NULL മൂല്യങ്ങൾ ഉണ്ടോ എന്ന് പരിശോധിക്കുന്നതിനുള്ള ഓപ്പറേറ്റർ ഏത്?

   a. IN b. LIKE c. IS d. NOT
4. പാറ്റേൺ ചേർച്ചയ്ക്ക് ഉപയോഗിക്കുന്ന ഓപ്പറേറ്ററാണ് ______________.
5. താഴെപ്പറയുന്ന പ്രസ്താവനയിലെ തെറ്റ് എന്ത്?

   SELECT * FROM emp WHERE grade = NULL;
6. ഒരു പട്ടികയിൽ നിന്ന് റെക്കോർഡുകൾ തിരഞ്ഞെടുക്കുന്നതിനുള്ള കമാൻഡാണ് __________________.

9.6.3 ORDER BY ഉപവാക്യം ഉപയോഗിച്ച് ഉത്തരങ്ങളുടെ ക്രമീകരണം (Sorting results using ORDER BY clause)

ഇതുവരെ ചർച്ച ചെയ്ത SELECT ക്വറിയുടെ ഫലങ്ങളിലെല്ലാം പട്ടികയിലുള്ള അതേ ക്രമത്തിലാണ് റെക്കോർഡുകൾ എപ്പോഴും ലഭിച്ചിട്ടുള്ളതെന്ന് നാം കണ്ടു. അവയെ ഏതെങ്കിലും രീതിയിൽ ക്രമീകരിക്കാൻ സാധിക്കുമോ? ഒരു ക്വറിയുടെ ഉത്തരത്തെ ആരോഹണക്രമത്തിലോ അവരോഹണക്രമത്തിലോ ക്രമീകരിക്കാൻ കഴിയും. ഇതിനായി ORDER BY എന്ന ഉപവവാക്യം ഉപയോഗിക്കാം. ക്രമീകരണത്തിന് ASC (ആരോഹണക്രമം) അല്ലെങ്കിൽ DESC (അവരോഹണക്രമം) എന്ന കീവേർഡ് ഉപയോഗിക്കാം. തനതു രീതിയിൽ പ്രദർശിപ്പിക്കുന്നത് ആരോഹണക്രമത്തിലായിരിക്കും. പട്ടികയിലുള്ള വരികളുടെ ക്രമത്തിൽ മാറ്റം വരുന്നില്ല. പകരം സ്ക്രീനിൽ പ്രദർശിപ്പിക്കുന്നവ മാത്രമാണ് ഈ രീതിയിൽ ക്രമീകരിക്കുന്നത്. ക്വറി 9.24 ഉപയോഗിച്ച് വിദ്യാർഥികളുടെ വിശദാംശങ്ങൾ അവരുടെ പേരിന്റെ അക്ഷരമാലാ ക്രമത്തിലാക്കാം. ചിത്രം 9.16 ഇതിന്റെ ഔട്ട്പുട്ട് കാണിക്കുന്നു.

ക്വറി 9.24 SELECT * FROM student ORDER BY name;

adm_no	name	gender	dob	course	f_income
1001	Alok	M	1998-10-02	Science	24000
1003	Bharath	M	1999-01-01	Commerce	45000
1057	Chaithanya	F	1999-06-03	Science	NULL
1006	Divakar	M	1998-02-21	Science	NULL
1025	Kaushi	M	1998-10-02	Commerce	17000
1005	Meera	F	1998-08-15	Science	NULL
1002	Nike	M	1998-11-26	Science	35000
1026	Niveditha	F	1999-03-04	Humanities	52000
1027	Sreekumar	M	1998-06-06	Science	15000
1004	Virat	M	1998-12-05	Science	22000

10 rows in set (0.00 sec)

ചിത്രം 9.16: വിദ്യാർഥികളുടെ വിശദാംശങ്ങൾ അവരുടെ പേരിന്റെ അക്ഷരമാലാ ക്രമത്തിൽ

ചിത്രം 9.16 വിദ്യാർത്ഥികളുടെ വിശദാംശങ്ങൾ അവരുടെ പേരിന്റെ അക്ഷരമാലാ ക്രമത്തിൽ

കുടുംബത്തിന്റെ മാസവരുമാനത്തിന്റെ ക്രമത്തിൽ അവരുടെ വിശദാംശങ്ങൾ നമുക്ക് കാണണമെന്നിരിക്കട്ടെ. ഏറ്റവും കൂടിയ വരുമാനം മുതൽ കുറഞ്ഞതു വരെയുള്ള ഫലം

കുടുംബത്തിന്റെ മാസവരുമാനത്തിനനുസരിച്ച് വിദ്യാർഥികളുടെ വിശദാംശങ്ങൾ നമുക്ക് ലഭിക്കണമെങ്കിൽ ക്വറി 9.25 ഉപയോഗിക്കാം. ചിത്രം 9.17 ൽ ഔട്ട്പുട്ട് കാണിച്ചിരിക്കുന്നു.

ക്വറി 9.25

```
SELECT * FROM student
ORDER BY f_income DESC;
```

ചിത്രം 9.17 ൽ കാണിച്ചിരിക്കുന്നതുപോലെ ORDER BY ഉപവാക്യത്തോടൊപ്പം ഉപയോഗിച്ചിരിക്കുന്ന നിരയിൽ NULL ഉണ്ടെങ്കിൽ അവ പട്ടികയുടെ അവസാനം പ്രദർശിപ്പിക്കും. ഒന്നിലധികം ക്രമീകരണങ്ങൾ ORDER BY ഉപവാക്യം ഉപയോഗിച്ച് നടത്താൻ സാധിക്കും. ഉദാഹരണത്തിന് വിദ്യാർഥികളെ അവരുടെ കോഴ്സ് അനുസരിച്ച്, പേരിന്റെ അക്ഷരമാലാ ക്രമത്തിലാക്കണം എന്ന് വിചാരിക്കുക. ക്വറി 9.26 ൽ കൊടുത്തിരിക്കുന്ന പ്രസ്താവന ഇതിനു വേണ്ടി ഉപയോഗിക്കാം. ചിത്രം 9.18 ൽ ഇതിന്റെ ഔട്ട്പുട്ട് കാണിച്ചിരിക്കുന്നു.

ക്വറി 9.26

```
SELECT name,course FROM student
ORDER BY course, name;
```

ചിത്രം 9.18 ലുള്ള ഔട്ട്പുട്ട് നോക്കുക. ആദ്യം course എന്ന നിരയിലെ വിലകളും, പിന്നീട് ഓരോ കോഴ്സിലെ പേരുകളും അക്ഷരമാലാ ക്രമത്തിലാക്കിയിരിക്കുന്നു.

```
+------------+------------+
| name       | course     |
+------------+------------+
| Bharath    | Commerce   |
| Kaushi     | Commerce   |
| Niveditha  | Humanities |
| Alok       | Science    |
| Chaithanya | Science    |
| Divakar    | Science    |
| Meera      | Science    |
| Nike       | Science    |
| Sreekumar  | Science    |
| Virat      | Science    |
+------------+------------+
10 rows in set (0.00 sec)
```

ചിത്രം 9.18: പല ഘടകങ്ങൾ ചേർന്ന ക്രമീകരണം

നേരത്തെ നാം WHERE ഉപവാക്യം ഉപയോഗിച്ച് നിബന്ധനകളുടെ അടിസ്ഥാനത്തിൽ വരികൾ വീണ്ടെടുക്കുന്നതെങ്ങനെ എന്ന് മനസ്സിലാക്കി. ഈ വരികൾ ഒരു പ്രത്യേക രീതിയിൽ ORDER BY ഉപവാക്യം ഉപയോഗിച്ച് ക്രമത്തിൽ പ്രദർശിപ്പിക്കാവുന്നതാണ്. ഇനി സയൻസ് ഗ്രൂപ്പിലെ കുട്ടികളുടെ പേര്, കുടുംബ വരുമാനം എന്നിവ വരുമാനത്തിന്റെ അവരോഹണ ക്രമത്തിൽ നമുക്ക് പ്രദർശിപ്പിക്കണമെന്നിരിക്കട്ടെ. ഈ ഔട്ട്പുട്ട് ലഭിക്കുന്നതിന് ക്വറി 9.27 ൽ കാണിച്ചിരിക്കുന്നതു പോലെ രണ്ട് ഉപവാക്യങ്ങൾ സംയോജിപ്പിക്കാവുന്നതാണ്. ഔട്ട്പുട്ട് ചിത്രം 9.19 കാണിച്ചിരിക്കുന്നു.

ക്വറി 9.27

```
SELECT name,course,f_income FROM student
WHERE course= 'Science'
ORDER BY f_income DESC;
```

ചിത്രം 9.19 കാണിച്ചിരിക്കുന്നതുപോലെ ക്രമീകരണനിരയിൽ NULL വിലയുള്ള റെക്കോർഡുകൾ അവസാനം കാണപ്പെടും. മറ്റൊരു പ്രധാനപ്പെട്ട കാര്യം ORDER BY ഉപവാക്യം WHERE ഉപവാക്യത്തിനു ശേഷമാണ് SELECT ൽ ഉപയോഗിക്കേണ്ടത്. കാരണം റെക്കോർഡുകൾ ആദ്യം തിരഞ്ഞെടുത്തതിനു ശേഷമാണ് അവ ക്രമീകരിക്കുന്നത്.

```
+------------+---------+----------+
| name       | course  | f_income |
+------------+---------+----------+
| Nike       | Science |    35000 |
| Alok       | Science |    24000 |
| Virat      | Science |    22000 |
| Sreekumar  | Science |    15000 |
| Meera      | Science |     NULL |
| Divakar    | Science |     NULL |
| Chaithanya | Science |     NULL |
+------------+---------+----------+
7 rows in set (0.00 sec)
```

ചിത്രം 9.19: നിബന്ധന അടിസ്ഥാനമാക്കിയുള്ള തിരഞ്ഞെടുക്കലും ക്രമീകരിക്കലും

നിങ്ങളുടെ പുരോഗതി അറിയുക

1. ORDER BY ഉപവാക്യത്തിന്റെ അർഥമെന്ത്?
2. ഡാറ്റ അവരോഹണക്രമത്തിൽ ക്രമീകരിക്കുന്നതിന് MySQL ൽ ഉപയോഗിക്കുന്ന കീവേർഡ് ഏത്?

   a. DESC   b. ASC   c. SORT   d. MODIFY
3. ORDER BY ഉപവാക്യത്തിന്റെ അഭാവത്തിൽ വരികൾ ഏതു ക്രമത്തിലാണ് പ്രദർശിപ്പിക്കപ്പെടുന്നത്?
4. SQLൽ ORDER BY ഉപവാക്യത്തിന്റെ തനതു ക്രമീകരണ രീതി എങ്ങനെയായിരിക്കും?

നമുക്ക് ചെയ്യാം

stock എന്ന പട്ടിക അടിസ്ഥാനമാക്കി വിവിധ നിബന്ധനകൾ ഉപയോഗിച്ച് റെക്കോർഡുകളെ ക്രമീകരിക്കാനുമുള്ള ക്വറികൾ നിർമ്മിക്കുക.

9.6.4 സംഗ്രഹ ഫങ്ഷനുകൾ (Aggregate functions)

ഒരു പട്ടികയിൽ മുഴുവനായോ WHERE ഉപയോഗിച്ച് പട്ടികയിൽ നിന്ന് വേർതിരിച്ചെടുത്ത ഒരു ഉപഗണത്തിനോ ഉപയോഗിക്കാവുന്ന ധാരാളം ബിൽട്ട് ഇൻ ഫങ്ഷനുകൾ MySQLൽ ഉണ്ട്. ഈ ഫങ്ഷനുകളെ സംഗ്രഹ ഫങ്ഷനുകൾ എന്നു പറയുന്നു. കാരണം ഇവ വരികളുടെ സംഗ്രഹത്തിനായി പ്രവർത്തിക്കുന്നു. ഈ ഫങ്ഷനുകളുടെ ഫലം ഒരൊറ്റ വിലയായിരിക്കും. സാധാരണയായി ഉപയോഗിക്കുന്ന സംഗ്രഹ ഫങ്ഷനുകൾ പട്ടിക 9.7 ൽ കൊടുത്തിരിക്കുന്നു.

ഫങ്ഷൻ	തിരികെത്തരുന്ന വില
SUM()	ആർഗ്യുമെന്റായി കൊടുത്തിരിക്കുന്ന നിരയിലുള്ള വിലകളുടെ തുക
AVG()	ആർഗ്യുമെന്റായി കൊടുത്തിരിക്കുന്ന നിരയിലുള്ള വിലകളുടെ ശരാശരി
MIN()	ആർഗ്യുമെന്റായി കൊടുത്തിരിക്കുന്ന നിരയിലെ ഏറ്റവും ചെറിയ വില
MAX()	ആർഗ്യുമെന്റായി കൊടുത്തിരിക്കുന്ന നിരയിലെ ഏറ്റവും വലിയ വില.
COUNT()	ആർഗ്യുമെന്റായി കൊടുത്തിരിക്കുന്ന നിരയിലെ NULL അല്ലാത്ത വിലകളുടെ എണ്ണം

പട്ടിക 9.7: MySQL ലെ ചില ബിൽട്ട് ഇൻ ഫങ്ഷനുകൾ

ക്വറി 9.28 വിദ്യാർഥിയുടെ കുടുംബ വരുമാനത്തിന്റെ ഏറ്റവും കൂടിയത്, കുറഞ്ഞത്, ശരാശരി എന്നിവ നൽകുന്നു. ചിത്രം 9.20 ഇതിന്റെ ഫലം കാണിച്ചിരിക്കുന്നു.

ക്വറി 9.28

```
SELECT MAX(f_income),MIN(f_income),AVG(f_income )
FROM student;
```

```
+---------------+---------------+---------------+
| MAX(f_income) | MIN(f_income) | AVG(f_income) |
+---------------+---------------+---------------+
|         52000 |         15000 |    30000.0000 |
+---------------+---------------+---------------+
1 row in set (0.03 sec)
```

ചിത്രം 9.20: സംഗ്രഹ ഫങ്ഷനുകളുടെ ഉപയോഗം

ഈ വിലകൾ കണക്കാക്കുമ്പോൾ സംഗ്രഹ നിരയിലെ NULL വിലകൾ പരിഗണിക്കുകയില്ല.

ക്വറി 9.29 ൽ കാണിച്ചിരിക്കുന്നതുപോലെ ചില തിരഞ്ഞെടുക്കൽ മാനദണ്ഡത്തിന്റെ അടിസ്ഥാനത്തിൽ രൂപീകൃതമായ പട്ടികയിലെ ഒരു ഉപഗണത്തിൽ ഈ ഫങ്ഷനുകൾ ഉപയോഗിക്കാവുന്നതാണ്. ഇത് സയൻസ് ഗ്രൂപ്പിലെ വിദ്യാർഥികളുടെ എണ്ണം തരുന്നു. ചിത്രം 9.19 ൽ അതിന്റെ ഔട്ട്പുട്ട് കാണിച്ചിരിക്കുന്നു.

ക്വറി 9.29

```
SELECT  COUNT(*), COUNT(f_income)
FROM student WHERE course = 'Science';
```

ചിത്രം 9.19 നാം പരിശോധിച്ചാൽ കൃത്യമായ ഉത്തരം 7 എന്ന് കാണാൻ കഴിയും. പിന്നെ എന്തുകൊണ്ടാണ് ചിത്രം 9.21 ലെ വിലകൾ തമ്മിൽ തുല്യമല്ലാത്തത്? ആദ്യത്തെ നിര COUNT (*) ആണ്. അത് course എന്ന നിരയിൽ സയൻസ് എന്ന വില ഉള്ള റെക്കോർഡു

കളുടെ എണ്ണം തരുന്നു. നക്ഷത്ര ചിഹ്നം (*) പട്ടികയിലെ എല്ലാ നിരകളെയും സൂചിപ്പിക്കുന്നു. അതുകൊണ്ട് ഒരു റെക്കോർഡിൽ ഏറ്റവും കുറഞ്ഞത് ഒരു ഫീൽഡെങ്കിലും ഉണ്ടെങ്കിൽ COUNT (*) ആ റെക്കോർഡ് പരിഗണിക്കും. പക്ഷെ ക്വറി 9.29ലെ COUNT (f_income), സയൻസിലെ f_income എന്ന നിരയിലെ NULL വിലകൾ കണക്കിലെടുക്കുന്നില്ല. അതുകൊണ്ടാണ് ചിത്രം 9.21 ലെ രണ്ടാമത്തെ നിരയിൽ 4 എന്ന വില കാണാൻ കഴിയുന്നത്. ചിത്രം 9.17 പരിശോധിക്കുക അതിൽ f_income എന്ന നിരയിലെ NULL ന്റെ എണ്ണം തിരിച്ചറിയുക.

```
+----------+-----------------+
| COUNT(*) | COUNT(f_income) |
+----------+-----------------+
|        7 |               4 |
+----------+-----------------+
1 row in set (0.03 sec)
```

ചിത്രം 9.21: COUNT() ഫങ്ഷന്റെ ഉപയോഗം

9.6.5 GROUP BY ഉപവാക്യം ഉപയോഗിച്ച് തരം തിരിക്കൽ (Grouping of records using GROUP BY clause)

ചിലപ്പോൾ മൂല്യങ്ങളുടെ അടിസ്ഥാനത്തിൽ പട്ടികയിലെ വരികൾ വിവിധ ഗണങ്ങളാക്കി നമുക്ക് വിഭജിക്കേണ്ടി വരും. പൊതുവിലയുടെ അടിസ്ഥാനത്തിൽ GROUP BY ഉപയോഗിച്ച് ഒരു പട്ടികയിലെ വരികൾ തരംതിരിക്കാം. GROUP BY ഉപവാക്യത്തിൽ കൊടുത്തിരിക്കുന്ന ആട്രിബ്യൂട്ട് ഉപയോഗിച്ച് ഗ്രൂപ്പുകൾ രൂപീകരിക്കാം. GROUP BY ഉപവാക്യത്തിൽ കൊടുത്തിരിക്കുന്ന ആട്രിബ്യൂട്ടിൽ ഒരേ വിലയുള്ള വരികൾ ഒരു ഗ്രൂപ്പിൽ കാണപ്പെടും. നിരയിലെ വ്യത്യസ്ത വിലകളുടെ അടിസ്ഥാനത്തിലാണ് ഗ്രൂപ്പുകൾ രൂപീകൃതമാകുന്നത്. അതിനാൽ ഈ പ്രക്രിയ റെക്കോർഡുകളുടെ തരംതിരിക്കലായി കണക്കാക്കാം. ക്വറി 9.29 ഉപയോഗിച്ച് സയൻസ് ഗ്രൂപ്പിലെ കുട്ടികളുടെ എണ്ണം ലഭിച്ചു. ഇനി ഓരോ ഗ്രൂപ്പിലെയും കുട്ടികളുടെ എണ്ണവും അതോടൊപ്പം കുടുംബത്തിന്റെ ശരാശരി മാസവരുമാനവും അറിയണമെന്ന് വിചാരിക്കുക. ക്വറി 9.30 ഉപയോഗിക്കാം. ഔട്ട്പുട്ട് ചിത്രം 9.22ൽ കൊടുത്തിരിക്കുന്നു.

ക്വറി 9.30

```
SELECT course, COUNT(*), AVG(f_income)
FROM student GROUP BY course ;
```

ഇവിടെ COURSE എന്ന നിരയിലെ വിവിധ വിലകളുടെ അടിസ്ഥാനത്തിൽ വിവിധ ഗ്രൂപ്പുകൾ രൂപീകരിച്ചിരിക്കുന്നു (കോമേഴ്സ്, ഹ്യൂമാനിറ്റീസ്, സയൻസ്). അതിനുശേഷം COUNT (*), AVG(f_income) എന്നീ ഫങ്ഷനുകൾ ഇതിലെ ഓരോ ഗ്രൂപ്പിലും ഉപയോഗിച്ചിരിക്കുന്നു.

```
+------------+----------+---------------+
| course     | COUNT(*) | AVG(f_income) |
+------------+----------+---------------+
| Commerce   |        2 |    31000.0000 |
| Humanities |        1 |    52000.0000 |
| Science    |        7 |    24000.0000 |
+------------+----------+---------------+
3 rows in set (0.00 sec)
```

ചിത്രം 9.22: GROUP BY ഉപയോഗിച്ചു കൊണ്ടുള്ള വരികളുടെ തരം തിരിക്കൽ

9.6.6 HAVING ഉപവാക്യത്തിലൂടെ നിബന്ധനകൾ പ്രയോഗിച്ച് ഗ്രൂപ്പുകൾ രൂപീകരിക്കുന്നു (Applying conditions to form groups using HAVING clause)

ഈ ഉപവാക്യം GROUP BY ഉപവാക്യത്തോടൊപ്പം ഉപയോഗിക്കുന്നു. ഓരോ വരിയിലും നിബന്ധനകൾ ഉപയോഗിക്കുന്ന WHERE ഉപവാക്യത്തെക്കുറിച്ചു ഇതിനോടകം പഠിച്ചു കഴിഞ്ഞു. HAVING ഉപവാക്യത്തിന്റെ സഹായത്തോടുകൂടി നിബന്ധനകൾ പ്രയോഗിച്ച് നമുക്ക് ഗ്രൂപ്പുകൾ രൂപീകരിക്കാവുന്നതാണ്. HAVING ഉപവാക്യത്തിലെ നിബന്ധന ഒരു കൂട്ടം റെക്കോർഡുകൾ രൂപീകരിക്കുന്നതിനാണ് ഉപയോഗിക്കുന്നത്, അല്ലാതെ വരികളിൽ ഒറ്റയ്ക്കല്ല. ക്വറി 9.31 ഗ്രൂപ്പുണ്ടാക്കുന്നതിനു വേണ്ട നിബന്ധനകൾ നൽകുന്നു. HAVING ഉപവാക്യത്തിന്റെ കൂടെ നിബന്ധനകൾ നൽകുമ്പോൾ മാത്രമാണ് ഇവിടെ ഗ്രൂപ്പുകൾ രൂപീകൃതമാകുന്നത്.

ക്വറി 9.31

```
SELECT course, COUNT(*)
GROUP BY course
HAVING COUNT(*)>3;
```

ചിത്രം 9.22 ഉം, 9.23 ഉം പരിശോധിച്ചാൽ നമുക്ക് സയൻസ് ഗ്രൂപ്പിൽ മാത്രമെ 3 വിദ്യാർഥികളിൽ കൂടുതൽ ഉള്ളൂ എന്ന് കാണാൻ കഴിയും. കൊമേഴ്സ്, ഹ്യുമാനിറ്റീസ് ഗ്രൂപ്പുകൾ ക്വറി 9.31 ന്റെ ഔട്ട്പുട്ടിൽ വന്നിട്ടില്ല. കാരണം, ഇവ ഓരോന്നിന്റെയും റെക്കോർഡുകളുടെ എണ്ണം യഥാക്രമം 2, 1 എന്നിങ്ങനെയാണ്. അതിനാലാണ് ഈ ഗ്രൂപ്പുകൾ ക്വറി 9.31 ന്റെ ഉത്തരത്തിൽ കാണാൻ സാധിക്കാത്തത്.

```
+----------+----------+
| course   | COUNT(*) |
+----------+----------+
| Science  |        7 |
+----------+----------+
1 row in set (0.00 sec)
```

ചിത്രം 9.23: നിബന്ധനകളുടെ അടിസ്ഥാനത്തിൽ ഗ്രൂപ്പുകൾ തിരിക്കുന്നു

നിങ്ങളുടെ പുരോഗതി അറിയുക

1. SQL ലെ സംഗ്രഹണത്തിനുപയോഗിക്കുന്ന ഫങ്ഷനുകൾ പട്ടികപ്പെടുത്തുക?
2. COUNT(*), COUNT(column_name) എന്നിവ എങ്ങനെ വ്യത്യാസപ്പെട്ടിരിക്കുന്നു?
3. WHERE, HAVING ഉപവാക്യങ്ങൾ തമ്മിലുള്ള വ്യത്യാസം എന്ത്?
4. താഴെ പറയുന്ന ക്വറിയുടെ ഉത്തരം എന്ത്?

```
SELECT COUNT (DISTINCT course) FROM student;
```

നമുക്ക് ചെയ്യാം

stock എന്ന പട്ടികയുടെ അടിസ്ഥാനത്തിൽ സംഗ്രഹ ഫങ്ഷനുകളും ORDER BY ഉപവാക്യവും ഉപയോഗിച്ച് ക്വറികൾ തയാറാക്കുക.

9.7 പട്ടികയിലെ ഡാറ്റയുടെ പരിഷ്കരണം (Modifying data in tables)

ചില സന്ദർഭങ്ങളിൽ ഒരു പട്ടികയിലെ നിരകളുടെ വിലകളിൽ മാറ്റം വരുത്തേണ്ടതായി വരാറുണ്ട്. ഉദാഹരണത്തിന് ശമ്പളം അല്ലെങ്കിൽ കൂലി പുതുക്കുമ്പോൾ വിദ്യാർഥിയുടെ കുടുംബത്തിന്റെ മാസവരുമാനത്തിൽ മാറ്റം വരാറുണ്ട്. അതുപോലെ ചില വിദ്യാർഥികളുടെ മാസവരുമാനം വിട്ടുപോയത് (NULL) പിന്നീട് സാധുവായ ഡാറ്റ നൽകി മാറ്റം വരുത്താവുന്നതാണ്. ഇങ്ങനെയുള്ള മാറ്റങ്ങൾ DML കമാൻഡായ UPDATE ഉപയോഗിച്ച് നിർവഹിക്കാം. അത് ഒന്നോ അതിലധികമോ നിർദിഷ്ട നിരകളിലെ വിലകളിൽ മാറ്റം വരുത്തുന്നതിന് ഉപയോഗിക്കാം. ഈ മാറ്റം പട്ടികയിലെ തിരഞ്ഞെടുത്ത വരികളിൽ മാത്രമെ പ്രതിഫലിക്കൂ. ഒരു വരിയിലെ നിരയിൽ പുതിയ ഡാറ്റ കൊണ്ടുവരുന്നതിന് SET കീവേർഡ് ഉപയോഗിക്കുന്നു. ഇത് UPDATE കമാൻഡിന്റെ കൂടെ നിർബന്ധമായും ഉപയോഗിക്കേണ്ട ഒരു ഉപവാക്യമാണ്. പുതിയ ഡാറ്റ ഒരു സ്ഥിര സംഖ്യയോ, പ്രയോഗമോ (Expression) മറ്റൊരു പട്ടികയിലെ ഡാറ്റയോ ആകാം. UPDATE കമാൻഡിന്റെ വാക്യഘടന താഴെ കൊടുക്കുന്നു.

```
UPDATE <table_name> SET <column_name> = <value>
[,<column_name> = <value>,. . . ]
[WHERE <condition>] ;
```

'Kaushi' എന്ന വിദ്യാർഥിയുടെ കുടുംബത്തിന്റെ മാസവരുമാനം Rs 27000 ലേക്ക് മാറ്റണമെന്ന് വിചാരിക്കുക. ക്വറി 9.32 ഇതിനായി ഉപയോഗിക്കുന്നു.

ക്വറി 9.32

```
UPDATE student
SET f_income = 27000
WHERE name = 'Kaushi' ;
```

ഈ ക്വറിയുടെ പ്രവർത്തനത്തിനു ശേഷം MySQL ൽ നിന്ന് താഴെക്കാണുന്ന രീതിയിലുള്ള പ്രതികരണം ലഭിക്കുന്നു.

```
Query O.K, 1 row affected (0.08 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

റെക്കോഡിൽ വന്ന മാറ്റം കാണുന്നതിനു വേണ്ടി താഴെ പറയുന്ന ക്വറി ഉപയോഗിക്കാം

```
SELECT * FROM student WHERE name = 'Kaushi'
```

ചിത്രം 9.24 ൽ ഈ ക്വറിയുടെ ഔട്ട്പുട്ട് കാണിച്ചിരിക്കുന്നു.

```
+--------+--------+--------+------------+----------+----------+
| adm_no | name   | gender | dob        | course   | f_income |
+--------+--------+--------+------------+----------+----------+
|   1025 | Kaushi | M      | 1998-10-02 | Commerce |    27000 |
+--------+--------+--------+------------+----------+----------+
1 row in set (0.00 sec)
```

ചിത്രം 9.24: f_income നിരയിൽ പരിഷ്കരിച്ച ഡാറ്റയോടുകൂടിയ Kaushi യുടെ വിശദാംശങ്ങൾ

ഈ ക്വറിയുടെ ഔട്ട്പുട്ട് ചിത്രം 9.17 മായി താരതമ്യം ചെയ്ത് മാറ്റങ്ങൾ മനസ്സിലാക്കുക. റെക്കോർഡിലെ നിരകൾക്ക് വില നൽകാൻ നമുക്ക് എക്സ്പ്രഷൻ ഉപയോഗിക്കാം. ക്വറി 9.33 ഈ ആശയം വിവരിക്കുന്നു.

ക്വറി 9.33

```
UPDATE student
SET f_income =  f_income + 1000
WHERE  f_income < 25000;
```

ഈ ക്വറിയോടുള്ള പ്രതികരണം താഴെ കൊടുക്കുന്നു.

```
Query O.K, 3 rows affected (0.06 sec)
Rows matched: 3 Changed: 3 Warnings: 0
```

3 വരികളാണ് തന്നിരിക്കുന്ന നിബന്ധനകൾ പാലിക്കുന്നത്. അതിനാൽ f_ income എന്ന നിരയിലെ ഈ വരികളെ 1000 വച്ച് വർധിപ്പിക്കുന്നു (ചിത്രം 9.17, 9.25 എന്നിവ പരിശോധിക്കുക)

NULL മൂല്യമുള്ള നിരകളിൽ മാറ്റം വരുത്തുന്നതിന് UPDATE കമാൻഡുകൾ ഉപയോഗിക്കാം. ഇപ്പോൾ NULL വിലയുള്ള നിരയിൽ മാസവരുമാനം 20000 രൂപ നമുക്ക് ചേർക്കണമെന്നിരിക്കട്ടെ. ക്വറി 9.34 അത് സാധ്യമാക്കുന്നു.

ക്വറി 9.34

```
UPDATE student
SET f_income =  20000
WHERE  f_income IS NULL;
```

ക്വറി 9.33 ലും 9.34 ലും വരുത്തിയ മാറ്റങ്ങൾ ചിത്രം 9.25 ൽ നമുക്ക് കാണാൻ കഴിയും. അതിന് SELECT *FROM student; എന്ന ക്വറി ഉപയോഗിക്കുക

```
+--------+------------+--------+------------+------------+----------+
| adm_no | name       | gender | dob        | course     | f_income |
+--------+------------+--------+------------+------------+----------+
|   1001 | Alok       | M      | 1998-10-02 | Science    |    25000 |
|   1002 | Nike       | M      | 1998-11-26 | Science    |    35000 |
|   1003 | Bharath    | M      | 1999-01-01 | Commerce   |    45000 |
|   1004 | Virat      | M      | 1998-12-05 | Science    |    23000 |
|   1005 | Meera      | F      | 1998-08-15 | Science    |    20000 |
|   1006 | Divakar    | M      | 1998-02-21 | Science    |    20000 |
|   1025 | Kaushi     | M      | 1998-10-02 | Commerce   |    27000 |
|   1026 | Niveditha  | F      | 1999-03-04 | Humanities |    52000 |
|   1027 | Sreekumar  | M      | 1998-06-06 | Science    |    16000 |
|   1057 | Chaithanya | F      | 1999-06-03 | Science    |    20000 |
+--------+------------+--------+------------+------------+----------+
10 rows in set (0.00 sec)
```

ചിത്രം 9.25: *f_income* നിരയിലെ തിരുത്തലുകൾ

നമുക്ക് ചെയ്യാം

stock എന്ന പട്ടിക ഉപയോഗിച്ച് ചില നിരകളിലെ വിലകളിൽ മാറ്റങ്ങൾ വരുത്താനാവശ്യമായ SQL പ്രസ്താവനകൾ എഴുതുക.

9.8 ഒരു പട്ടികയുടെ രൂപഘടന മാറ്റുന്നു (Changing the structure of a table

ചില സാഹചര്യങ്ങളിൽ നമുക്ക് പട്ടികയുടെ രൂപഘടനയിൽ മാറ്റങ്ങൾ വരുത്തേണ്ടതായി വരാം. ഈ പ്രവർത്തനം രൂപരേഖ (schema) യുമായി ബന്ധപ്പെട്ടതാണ്. അതിന് DDL കമാൻഡായ ALTER TABLE ഉപയോഗിക്കാം. നിരകൾ കൂട്ടിച്ചേർക്കൽ, ഒഴിവാക്കൽ, നിലവിലുള്ള നിരയുടെ ഡാറ്റ ഇനത്തിലും വലുപ്പത്തിലും മാറ്റം വരുത്തുക, പട്ടികയ്ക്ക് പുതിയ പേര് നൽകുക തുടങ്ങിയ പ്രവർത്തനങ്ങൾക്ക് ഈ നിർദേശം ഉപയോഗിക്കാം. ഇനി ഈ മാറ്റങ്ങൾ എങ്ങനെയാണ് വരുത്തുന്നത് എന്ന് നമുക്ക് നോക്കാം.

9.8.1 പുതിയ നിര കൂട്ടിച്ചേർക്കുന്നു (Adding a new column)

ഒന്നോ അതിലധികമോ നിരകൾ ഒരു പട്ടികയിൽ എവിടെ വേണമെങ്കിലും കൂട്ടിച്ചേർക്കാവുന്നതാണ്. പുതിയ നിര കൂട്ടിച്ചേർക്കുന്നതിനുള്ള (ALTER TABLEന്റെ) വാക്യഘടന താഴെ കൊടുക്കുന്നു.

```
ALTER TABLE <table_name>
ADD <column_name> <data_type> [<size>] [<constraint>]
[FIRST | AFTER <column_name> ] ;
```

ഇവിടെ <table_name> എന്നത് ഏത് പട്ടികയിലാണോ മാറ്റം വരുത്തേണ്ടത് അതിന്റെ പേരാണ്. ADD എന്നത് ഒരു പട്ടികയിൽ പുതിയ നിരകൾ കൂട്ടിച്ചേർക്കാനുള്ള കീവേഡാണ്. <column_name> <data_type> [<size>] എന്നിവ പുതിയ നിരയുടെ വിശദാംശങ്ങളാണ്. FIRST | AFTER എന്നത് പുതുതായി ചേർക്കുന്ന നിരയുടെ സ്ഥാനം സൂചിപ്പിക്കുന്നു.

പുതിയ നിര ആദ്യമാണ് ചേർക്കേണ്ടത് എങ്കിൽ നമുക്ക് FIRST എന്ന ഉപവാക്യം ഉപയോഗിക്കാം. പുതിയ നിര പ്രത്യേകസ്ഥാനത്താണ് ചേർക്കേണ്ടതെങ്കിൽ AFTER <column_name> എന്ന ഉപവാക്യം ഉപയോഗിക്കണം. പുതിയ നിരയുടെ സ്ഥാനം നാം വ്യക്തമാക്കിയിട്ടില്ലെങ്കിൽ അത് പട്ടികയുടെ അവസാനനിരയായി കൂട്ടിച്ചേർക്കപ്പെടും. പുതുതായി കൂട്ടിച്ചേർക്കുന്ന നിരകളിൽ NULL മൂല്യങ്ങളാണ് ഉണ്ടാവുക എന്നത് ശ്രദ്ധിക്കുക.

ഇനി നമുക്ക് gr_mks, reg_no എന്നീ രണ്ട് നിരകൾ student പട്ടികയിൽ കൂട്ടിച്ചേർക്കാം. ഇവ യഥാക്രമം ഗ്രേസ്മാർക്കും വിദ്യാർഥിയുടെ ഹയർ സെക്കന്ററി പരീക്ഷയിലെ രജിസ്റ്റർ നമ്പറും സംഭരിക്കുന്നതിനുള്ളതാണ്. ക്വറി 9.35 ന് പട്ടികയുടെ രൂപ ഘടനയിൽ ഈ മാറ്റങ്ങൾ വരുത്താൻ കഴിയും.

ക്വറി 9.35

```
ALTER TABLE student
ADD gr_mks INTEGER
AFTER dob, ADD reg_no INTEGER;
```

MySQL ന്റെ പ്രതികരണം താഴെ കൊടുക്കുന്നു.

```
Query O.K, 10 rows affected (0.25 sec)
Rows matched: 10 Changed: 10 Warnings: 0
```

9.8.2 ഒരു നിരയുടെ നിർവചനത്തിൽ മാറ്റം വരുത്തുന്നു (Changing the definition of a column)

ഒരു നിരയുടെ സവിശേഷതകളായ ഡാറ്റ ഇനം, വലുപ്പം അല്ലെങ്കിൽ കൺസ്ട്രയിന്റ് എന്നിവയ്ക്ക് മാറ്റം വരുത്തുന്നതിന് ALTER TABLE ന്റെ കൂടെ MODIFY എന്ന ഉപവാക്യം ഉപയോഗിക്കാം. ഇതിനുള്ള വാക്യഘടന താഴെ കൊടുക്കുന്നു.

```
ALTER TABLE <table_name>
MODIFY <column_name> <data_type> [<size>]
[<constraint>];
```

ഇനി പുതുതായി കൂട്ടിച്ചേർത്ത reg_no എന്ന നിരയിൽ UNIQUE എന്ന ഉപവാക്യം നൽകി രണ്ടു കുട്ടികൾക്ക് ഒരേ രജിസ്റ്റർ നമ്പർ ഉണ്ടായിരിക്കില്ല എന്ന് ഉറപ്പുവരുത്തുവാൻ സാധിക്കുന്നതാണ്. ക്വറി 9.36 ഈ മാറ്റം വരുത്തുന്നു.

ക്വറി 9.36

```
ALTER TABLE student
MODIFY reg_no INTEGER UNIQUE;
```

MySQL ന്റെ പ്രതികരണവും ക്വറി 9.35 ൽ നമുക്ക് ലഭിച്ച പ്രതികരണവും ഒന്നു തന്നെയായിരിക്കും. നമുക്ക് ഈ മാറ്റങ്ങൾ DESC student; എന്ന കമാൻഡ് ഉപയോഗിച്ച് മനസ്സിലാക്കാവുന്നതാണ്. ക്വറി 9.35 ലും 9.36 ലും വരുത്തിയിട്ടുള്ള മാറ്റങ്ങൾ നമുക്ക് വീക്ഷിക്കാവുന്നതാണ് ചിത്രം 9.26 ൽ ഔട്ട്പുട്ട് കാണിച്ചിട്ടുണ്ട്.

```
+-----------+-------------+------+-----+---------+----------------+
| Field     | Type        | Null | Key | Default | Extra          |
+-----------+-------------+------+-----+---------+----------------+
| adm_no    | int(11)     | NO   | PRI | NULL    | auto_increment |
| name      | varchar(20) | NO   |     | NULL    |                |
| gender    | char(1)     | YES  |     | M       |                |
| dob       | date        | YES  |     | NULL    |                |
| gr_mks    | int(11)     | YES  |     | NULL    |                |
| course    | varchar(15) | YES  |     | NULL    |                |
| f_income  | int(11)     | YES  |     | NULL    |                |
| reg_no    | int(11)     | YES  | UNI | NULL    |                |
+-----------+-------------+------+-----+---------+----------------+
8 rows in set (0.00 sec)
```

ചിത്രം 9.26: student പട്ടികയിൽ മാറ്റം വരുത്തിയതിനുശേഷമുള്ള രൂപഘടന

gr_mks എന്ന നിര dob ന് ശേഷവും reg_no എന്ന നിര അവസാന നിരയായും കൂട്ടിച്ചേർത്തതായി ചിത്രം 9.26 ൽ നമുക്ക് കാണാൻ സാധിക്കും. reg_no ന് നേരെ Key നിരയിൽ UNIQUE കൺസ്ട്രയിന്റ് ഉൾപ്പെട്ടിരിക്കുന്നതു കൂടി ശ്രദ്ധിക്കുക. SELECT *FROM student എന്ന കമാൻഡ് ഉപയോഗിച്ചാൽ ചിത്രം 9.27 ൽ കാണുന്നതുപോലെ പുതിയ നിരകളുടെ ഉള്ളടക്കത്തിൽ NULL വിലകൾ കാണാൻ കഴിയും. പുതുതായി നിർമ്മിച്ച നിരകളിൽ ഇതുവരെ വിലകൾ കൊടുക്കാത്തതു കൊണ്ടാണ് ഇങ്ങനെ സംഭവിച്ചത്.

adm_no	name	gender	dob	gr_mks	course	f_income	reg_no
1001	Alok	M	1998-10-02	NULL	Science	25000	NULL
1002	Nike	M	1998-11-26	NULL	Science	35000	NULL
1003	Bharath	M	1999-01-01	NULL	Commerce	45000	NULL
1004	Virat	M	1998-12-05	NULL	Science	23000	NULL
1005	Meera	F	1998-08-15	NULL	Science	20000	NULL
1006	Divakar	M	1998-02-21	NULL	Science	20000	NULL
1025	Kaushi	M	1998-10-02	NULL	Commerce	27000	NULL
1026	Niveditha	F	1999-03-04	NULL	Humanities	52000	NULL
1027	Sreekumar	M	1998-06-06	NULL	Science	16000	NULL
1057	Chaithanya	F	1999-06-03	NULL	Science	20000	NULL

10 rows in set (0.00 sec)

ചിത്രം 9.27:പട്ടികയുടെ രൂപഘടന പരിഷ്കരിച്ചതിനു ശേഷമുള്ള പുതിയ നിരകളുടെ ഉള്ളടക്കം

നമുക്ക് ചെയ്യാം

പുതുതായി കൂട്ടിച്ചേർത്ത നിരകളിൽ ഡാറ്റ ചേർക്കുന്നതിനും അത് പ്രദർശിപ്പിക്കുന്നതിനുമുള്ള SQL പ്രസ്താവനകൾ എഴുതുക.

stock എന്ന പട്ടികയ്ക്ക് ഘടനാപരമായ മാറ്റങ്ങൾ വരുത്തി അതിൽ പുതിയ വിലകൾ ചേർക്കുക.

9.8.3 പട്ടികയിൽ നിന്ന് ഒരു നിര ഒഴിവാക്കുന്നു (Removing column from a table)

പട്ടികയിൽ നിന്ന് ഒരു നിര ഒഴിവാക്കണമെങ്കിൽ ALTER TABLE കമാൻഡിനോടൊപ്പം DROP എന്ന ഉപവാക്യം ഉപയോഗിക്കണം ഇതിന്റെ വാക്യഘടന താഴെ കൊടുക്കുന്നു.

```
ALTER TABLE <table_name>
DROP <column_name>;
```

ഉദാഹരണമായി student എന്ന പട്ടികയിൽ നിന്ന് gr_mks എന്ന നിര ഒഴിവാക്കണമെങ്കിൽ ക്വറി 9.37 ഉപയോഗിച്ചാൽ മതിയാകും.

ക്വറി 9.37

```
ALTER TABLE student
DROP gr_mks;
```

മുൻപ് ALTER TABLE പ്രസ്താവനകളിൽ നിന്ന് നമുക്ക് ലഭിച്ച അതേ പ്രതികരണമായിരിക്കും MySQL ൽ നിന്നും ലഭിക്കുക. മാറ്റങ്ങൾ DESC student; എന്ന കമാൻഡ് ഉപയോഗിച്ച് നിരീക്ഷിക്കാവുന്നതാണ്.

9.8.4 പട്ടിക പുനർനാമകരണം ചെയ്യുന്നു (Renaming a table)

ഡാറ്റാബേസിലെ ഒരു പട്ടിക പുനർനാമകരണം ചെയ്യുന്നതിന് RENAME TO എന്ന ഉപവാക്യം ALTER TABLE കമാൻഡിനോടൊപ്പം ഉപയോഗിച്ചാൽ മതി. പട്ടികയിൽ വരികളുണ്ടെങ്കിലും അതിനെ നമുക്ക് പുനർനാമകരണം ചെയ്യാം. പട്ടിക പുനർനാമകരണം ചെയ്യുന്നതിനുള്ള കമാൻഡിന്റെ വാക്യ ഘടന താഴെ കൊടുക്കുന്നു

```
ALTER TABLE <table_name>
RENAME TO <new_table_name>;
```

ഉദാഹരണത്തിന് student എന്ന പട്ടികയ്ക്ക് student2015 എന്ന പേര് മാറ്റിനൽകണമെങ്കിൽ ക്വറി 9.38 ഉപയോഗിക്കാം.

ക്വറി 9.38

```
ALTER TABLE student
RENAME TO student2015;
```

```
+-------------------+
| Tables_in_school  |
+-------------------+
| student2015       |
+-------------------+
1 row in set (0.00 sec)
```

ചിത്രം 9.28: പുനർനാമകരണം ചെയ്ത പട്ടിക

ഈ ക്വറിയുടെ പ്രതികരണം താഴെ കൊടുക്കുന്നു

Query O.K, 0 rows affected (0.06 sec)

പട്ടികയുടെ പേര് മാറിയെങ്കിലും അതിലെ ഉള്ളടക്കത്തിൽ മാറ്റം വന്നിട്ടില്ല എന്ന് പ്രതികരണം സൂചിപ്പിക്കുന്നു.

മാറ്റങ്ങൾ കാണുന്നതിന് SHOW TABLES എന്ന ക്വറി ഉപയോഗിക്കാം. ചിത്രം 9.28 ൽ ഔട്ട്പുട്ട് കാണിച്ചിരിക്കുന്നു.

9.9 പട്ടികയിൽ നിന്ന് വരികൾ ഒഴിവാക്കുന്നു (Deleting rows from a table)

ചിലപ്പോൾ ഒരു പട്ടികയിൽ നിന്ന് ഒന്നോ അതിൽക്കൂടുതലോ റെക്കോർഡുകൾ നമുക്ക് ഒഴിവാക്കേണ്ടതായി വരും. ഇതിനായി DML കമാൻഡായ DELETE ഉപയോഗിക്കുന്നു. ഒഴിവാക്കേണ്ട വരികൾ WHERE ഉപവാക്യം ഉപയോഗിച്ച് തിരഞ്ഞെടുക്കുന്നു. WHERE ഉപയോഗിച്ചിട്ടില്ലെങ്കിൽ പട്ടികയിലെ എല്ലാ റെക്കോർഡുകളും നഷ്ടപ്പെടും. DELETE കമാൻഡ് ഒഴിവാക്കുന്നത് റെക്കോർഡുകളെയാണ്. ഇതിന്റെ വാക്യഘടന താഴെ കൊടുക്കുന്നു.

```
DELETE FROM <table_name> [WHERE <condition>] ;
```

ഉദാഹരണത്തിന് അഡ്മിഷൻ നമ്പർ 1027 ആയ Sreekumar എന്ന വിദ്യാർഥിയുടെ റെക്കോർഡ് പട്ടികയിൽ നിന്ന് ഒഴിവാക്കണമെങ്കിൽ ക്വറി 9.39 ഉപയോഗിക്കാം.

ക്വറി 9.39

```
DELETE FROM student2015 WHERE adm_no = 1027 ;
```

ഈ ക്വറിയുടെ ഔട്ട്പുട്ട് താഴെ കൊടുക്കുന്നു:

```
Query O.K, 1 row affected (0.08 sec)
```

ഇവിടെ പട്ടികയുടെ പേര് കൊടുത്തിരിക്കുന്നത് student2015 എന്നാണ്. പട്ടികയിലെ മാറ്റം ചിത്രം 9.29 ൽ നിരീക്ഷിക്കാവുന്നതാണ്. അത് ലഭിക്കുന്നതിന് താഴെ കൊടുക്കുന്ന ക്വറി ഉപയോഗിക്കുക.

```
SELECT * FROM student2015;
```

adm_no	name	gender	dob	course	f_income	reg_no
1001	Alok	M	1998-10-02	Science	25000	NULL
1002	Nike	M	1998-11-26	Science	35000	NULL
1003	Bharath	M	1999-01-01	Commerce	45000	NULL
1004	Virat	M	1998-12-05	Science	23000	NULL
1005	Meera	F	1998-08-15	Science	20000	NULL
1006	Divakar	M	1998-02-21	Science	20000	NULL
1025	Kaushi	M	1998-10-02	Commerce	27000	NULL
1026	Niveditha	F	1999-03-04	Humanities	52000	NULL
1057	Chaithanya	F	1999-06-03	Science	20000	NULL

9 rows in set (0.02 sec)

ചിത്രം 9.29: ഒരു റെക്കോർഡ് ഒഴിവാക്കിയതിനു ശേഷം പുനർനാമകരണം ചെയ്ത പട്ടികയിലെ ഉള്ളടക്കം

ചിത്രം 9.29 ൽ ഏഴു നിരകൾ മാത്രമെ കാണാൻ കഴിയൂ. കാരണം ക്വറി 9.37 ഉപയോഗിച്ച് gr_mks എന്ന നിര നേരത്തെ ഒഴിവാക്കിയിട്ടുണ്ട്.

ഒരു നിരയിൽ നിന്ന് മൂല്യങ്ങൾ ഒഴിവാക്കാൻ സാധിക്കുകയില്ല. ഇത് ഫലത്തിൽ കൊണ്ടുവരാൻ താഴെ കാണും വിധം UPDATE കമാൻഡ് ഉപയോഗിച്ച് നിരകളിൽ NULL മൂല്യങ്ങൾ ചേർക്കുകയാണ് വേണ്ടത്.

```
UPDATE <table_name>
       SET <column_name> = NULL
       [WHERE <condition>] ;
```

9.10 ഡാറ്റാബേസിൽ നിന്ന് പട്ടിക ഒഴിവാക്കുന്നു (Removing table from a database)

ഡാറ്റാബേസിൽ നിന്ന് ഒരു പട്ടിക ഒഴിവാക്കണമെങ്കിൽ DROP TABLE എന്ന കമാൻഡ് ഉപയോഗിച്ചാൽ മതി. ഈ DDL കമാൻഡ് പട്ടികയിൽ റെക്കോർഡുകൾ ഉണ്ടെങ്കിലും അതിനെ ഡാറ്റാബേസിൽ നിന്ന് സ്ഥിരമായി നീക്കം ചെയ്യുന്നു. ഈ കമാൻഡ് ഉപയോഗിക്കുമ്പോൾ വളരെ ശ്രദ്ധിക്കണം. കാരണം, ഒരിക്കൽ പട്ടിക നീക്കം ചെയ്താൽ അത് തിരിച്ചെടുക്കാൻ സാധിക്കുകയില്ല. ഈ കമാൻഡിന്റെ വാക്യഘടന താഴെ കൊടുക്കുന്നു.

```
DROP TABLE <table_name>;
```

ഉദാഹരണത്തിന് school ഡാറ്റാബേസിൽ നിന്ന് student2015 എന്ന പട്ടിക ഒഴിവാക്കണമെങ്കിൽ ക്വറി 9.40 ഉപയോഗിക്കാം.

ക്വറി 9.40

```
DROP TABLE student2015;
```

നിങ്ങളുടെ പുരോഗതി അറിയുക

1. ഒരു പട്ടികയുടെ ഘടനയിൽ മാറ്റം വരുത്താനുപയോഗിക്കുന്ന കമാൻഡാണ് _______.
2. ഒരു നിരയുടെ ഘടനയിൽ മാറ്റം വരുത്തിയാൽ അതിലെ വിലകളെ അത് ബാധിക്കുന്നു. ഈ പ്രസ്താവന ശരിയോ തെറ്റോ?
3. ഒരു പട്ടികയിൽ നിന്ന് എങ്ങനെയാണ് ഒരു നിര ഒഴിവാക്കുന്നത്?
4. ഒരു പട്ടികയിൽ നിന്ന് വരി ഒഴിവാക്കാനുള്ള കമാൻഡ് ഏത്?
5. DELETE FROM എന്ന കമാൻഡിന്റെ കൂടെ WHERE ഉപവാക്യം ഉപയോഗിക്കാതിരുന്നാൽ എന്ത് സംഭവിക്കും?

9.11 നെസ്റ്റഡ് ക്വറികൾ (Nested queries)

'നെസ്റ്റിങ്' എന്ന ആശയം നമുക്ക് സുപരിചിതമാണല്ലോ. 'നെസ്റ്റഡ് if', 'നെസ്റ്റഡ് ലൂപ്പ്' എന്നിവയിൽ ഇത് നാം ചർച്ച ചെയ്തിരുന്നു. ഇനി നമുക്ക് നെസ്റ്റഡ് ക്വറികളെക്കുറിച്ച് ചർച്ച ചെയ്യാം. നെസ്റ്റഡ് എന്നാൽ 'ഒന്നിനുള്ളിൽ മറ്റൊന്ന്' എന്നാണ് അർഥമാക്കുന്നത്. ഇവിടെ ഒരു ക്വറിയുടെ ഫലം ഓപ്പറേഷൻ സമയത്തു തന്നെ മറ്റൊരു ക്വറിയുടെ നിബന്ധനയിലേക്ക് കൈമാറ്റം ചെയ്യപ്പെടുന്നു. ഒരു MySQL ന്റെ ഉള്ളിലുള്ള ക്വറിയെ ഉപക്വറി (inner query, subquery) എന്നും, ആ ഉപക്വറി അടങ്ങിയിരിക്കുന്ന ക്വറിയെ ബാഹ്യക്വറി (outer query) എന്നും പറയുന്നു. SQL ആദ്യം WHERE ഉപവാക്യം അടങ്ങിയ ഉപക്വറി ആദ്യം പ്രവർത്തിപ്പിക്കുന്നു. അതിന്റെ ഫലം ബാഹ്യക്വറിയുടെ നിബന്ധനയിലേക്ക് കൈമാറുന്നു. റിലേഷണൽ ഓപ്പറേറ്റർ ഉപയോഗിക്കുമ്പോൾ ഉപക്വറി ഒറ്റവരി ഔട്ട്പുട്ട് തിരികെ നൽകുന്നുവെന്ന് ഉറപ്പുവരുത്തേണ്ടതാണ്.

ഏറ്റവും കൂടുതൽ കുടുംബവരുമാനമുള്ള വിദ്യാർഥിയുടെ പേരും കോഴ്സും നമുക്ക് പ്രദർശിപ്പിക്കണമെന്നിരിക്കട്ടെ. ക്വറി 9.41 ഉപയോഗിച്ച് നമുക്ക് ഈ കാര്യം ചെയ്യാവുന്നതാണ്. ചിത്രം 9.30 ൽ ഇതിന്റെ ഔട്ട്പുട്ട് കാണിച്ചിരിക്കുന്നു.

ക്വറി 9.41

```
SELECT name,course FROM student2015
WHERE f_income =
(SELECT MAX(f_income)
    FROM student2015);
```

```
+-----------+------------+
| name      | course     |
+-----------+------------+
| Niveditha | Humanities |
+-----------+------------+
1 row in set (0.03 sec)
```

ചിത്രം 9.30: നെസ്റ്റഡ് ക്വറിയുടെ ഫലം

ഈ ഉദാഹരണത്തിൽ ഉപക്വറി ആദ്യം വിലയിരുത്തി. അതിൽ നിന്ന് f_income എന്ന നിരയിലെ ഏറ്റവും വലിയ വില തിരികെ ലഭിക്കുന്നു (ഇവിടെ ഇത് Rs 52000). ഈ മൂല്യം പിന്നീട് പട്ടികയിലെ f_income എന്ന റെക്കോർഡിന്റെ മൂല്യവുമായി താരതമ്യം ചെയ്ത് തുല്യമാണെങ്കിൽ ബാഹ്യക്വറി അത് വീണ്ടെടുക്കുന്നു.

9.12 വ്യൂ എന്ന ആശയം (Concept of views)

MySQL വ്യൂ എന്ന ആശയത്തെ പിന്തുണയ്ക്കുന്നു. ഇത് RDBMS ലെ ഒരു സവിശേഷതയാണ്. വ്യൂ എന്നത് ഒരു സാങ്കൽപ്പിക പട്ടികയാണ് (Virtual table). യഥാർഥത്തിൽ ഇവ ഡാറ്റാബേസിൽ ഇല്ല. പക്ഷേ, ഒന്നോ അതിലധികമോ പട്ടികകളിൽ നിന്നും ഉരുത്തിരിച്ചെടുക്കാവുന്നതാണ്. ഏത് പട്ടികയിൽ നിന്നാണോ വ്യൂ നിർമ്മിക്കുന്നതിനാവശ്യമായ വരികൾ ശേഖരിക്കുന്നത് അതിനെ അടിസ്ഥാനപ്പട്ടിക (Base table) എന്നു പറയുന്നു. ഈ വരികൾ ഭൗതികമായി എവിടെയും സംഭരിക്കുന്നില്ല പകരം വ്യൂവിന്റെ നിർവചനം ഡാറ്റാബേസിനകത്ത് സംഭരിക്കുന്നു. അടിസ്ഥാന പട്ടികയിൽ സംഭരിച്ചിരിക്കുന്ന, നമുക്കാവശ്യമുള്ള വിവരങ്ങൾ കാണാൻ കഴിയുന്ന ഒരു ജാലകം പോലെയാണ് വ്യൂ. ഒരു ക്വറിയുടെ നിബന്ധന അടിസ്ഥാനമാക്കി മറ്റു പട്ടികകളിൽ നിന്നും വ്യൂവിലെ ഉള്ളടക്കം എടുക്കുന്നു. `CREATE VIEW` എന്ന DDL കമാൻഡ് ഉപയോഗിച്ച് വ്യൂ നിർമ്മിക്കാം. അതിന്റെ വാക്യഘടന താഴെ കൊടുക്കുന്നു.

```
CREATE VIEW <view_name>
AS SELECT <column_name 1> [, <column_name2>,...]
FROM <table_name>
[WHERE <condition> ];
```

ഇനി ജനുവരി 1,1999 ന് മുൻപ് ജനിച്ച കുട്ടികളുടെ വിവരങ്ങൾ ഉപയോഗിച്ച് ഒരു വ്യൂ ഉണ്ടാക്കാം. ക്വറി 9.42 student1998 എന്ന വ്യൂ നിർമിക്കുന്നു.

ക്വറി 9.42

```
CREATE VIEW student1998
AS SELECT * FROM student2015
WHERE dob < '1999-1-1';
```

ഈ ക്വറിയുടെ ഔട്ട്പുട്ട് താഴെക്കൊടുക്കുന്നു

```
Query O.K, 0 row affected (0.31 sec)
```

ചിത്രം 9.31 വ്യൂവിന്റെ ഘടന കാണിക്കുന്നു. ചിത്രം 9.32 വ്യൂവിന്റെ ഉള്ളടക്കം പ്രദർശിപ്പിക്കുന്നു. ഇത് ലഭിക്കുന്നതിന് യഥാക്രമം കമാൻഡുകളായ `DESC student1998;` , `SELECT * FROM student1998;` എന്നീ നിർദേശങ്ങൾ നൽകേണ്ടതാണ്.

```
+-----------+-------------+------+-----+---------+-------+
| Field     | Type        | Null | Key | Default | Extra |
+-----------+-------------+------+-----+---------+-------+
| adm_no    | int(11)     | NO   |     | 0       |       |
| name      | varchar(20) | NO   |     | NULL    |       |
| gender    | char(1)     | YES  |     | M       |       |
| dob       | date        | YES  |     | NULL    |       |
| course    | varchar(15) | YES  |     | NULL    |       |
| f_income  | int(11)     | YES  |     | NULL    |       |
| reg_no    | int(11)     | YES  |     | NULL    |       |
+-----------+-------------+------+-----+---------+-------+
7 rows in set (0.01 sec)
```

ചിത്രം 9.31: student1998 എന്ന വ്യൂവിന്റെ രൂപഘടന

```
+---------+---------+--------+------------+----------+---------+--------+
| adm_no  | name    | gender | dob        | course   | f_income| reg_no |
+---------+---------+--------+------------+----------+---------+--------+
|   1001  | Alok    | M      | 1998-10-02 | Science  |   25000 |   NULL |
|   1002  | Nike    | M      | 1998-11-26 | Science  |   35000 |   NULL |
|   1004  | Virat   | M      | 1998-12-05 | Science  |   23000 |   NULL |
|   1005  | Meera   | F      | 1998-08-15 | Science  |   20000 |   NULL |
|   1006  | Divakar | M      | 1998-02-21 | Science  |   20000 |   NULL |
|   1025  | Kaushi  | M      | 1998-10-02 | Commerce |   27000 |   NULL |
+---------+---------+--------+------------+----------+---------+--------+
6 rows in set (0.00 sec)
```

ചിത്രം 9.32: student1998 എന്ന വ്യൂവിന്റെ ഉള്ളടക്കം

ചിത്രം 9.31ൽ കീ എന്ന നിരയിൽ adm_no, reg_no എന്നിവയുടെ കൺസ്ട്രയിന്റ് ഉൾപ്പെട്ടിട്ടില്ല. ചിത്രം 9.32 കാണിക്കുന്നത് 1998 ൽ ജനിച്ച വിദ്യാർഥികളുടെ വിശദാംശങ്ങളാണ്.

പട്ടികകളോടൊപ്പം DML കമാൻഡ് ഉപയോഗിക്കുന്നതുപോലെ വ്യൂ വിനോടൊപ്പം അവ ഉപയോഗിക്കാം. അങ്ങനെയുള്ള ക്വറിയുടെ ഉത്തരം ചിത്രം 9.32 ൽ കൊടുത്തിരിക്കുന്നു. അടിസ്ഥാന പട്ടികകൾ നാം ഉപയോഗിക്കുന്നത് അതിനോട് ചേർന്നുള്ള വ്യൂവിൽ കൂടിയാണ്. അതിനാൽ വ്യൂവിൽ UPDATE, DELETE എന്നീ പ്രവർത്തനങ്ങൾ ചെയ്യുമ്പോൾ അടിസ്ഥാന പട്ടികയിലും ആ മാറ്റങ്ങൾ ഉണ്ടാകുന്നു. ക്വറി 9.43 ഈ ആശയം വിവരിക്കുന്നു.

ക്വറി 9.43

```
UPDATE student1998
SET reg_no = 2201020
WHERE adm_no=1001 ;
```

```
+--------+-------------+--------+------------+------------+----------+---------+
| adm_no | name        | gender | dob        | course     | f_income | reg_no  |
+--------+-------------+--------+------------+------------+----------+---------+
|   1001 | Alok        | M      | 1998-10-02 | Science    |    25000 | 2201020 |
|   1002 | Nike        | M      | 1998-11-26 | Science    |    35000 |    NULL |
|   1003 | Bharath     | M      | 1999-01-01 | Commerce   |    45000 |    NULL |
|   1004 | Virat       | M      | 1998-12-05 | Science    |    23000 |    NULL |
|   1005 | Meera       | F      | 1998-08-15 | Science    |    20000 |    NULL |
|   1006 | Divakar     | M      | 1998-02-21 | Science    |    20000 |    NULL |
|   1025 | Kaushi      | M      | 1998-10-02 | Commerce   |    27000 |    NULL |
|   1026 | Niveditha   | F      | 1999-03-04 | Humanities |    52000 |    NULL |
|   1057 | Chaithanya  | F      | 1999-06-03 | Science    |    20000 |    NULL |
+--------+-------------+--------+------------+------------+----------+---------+
9 rows in set (0.00 sec)
```

ചിത്രം 9.33: വ്യൂവിലൂടെയുള്ള അടിസ്ഥാന പട്ടികയുടെ പരിഷ്ക്കരണം.

ഇനി നമുക്ക് ഈ ക്വറി പ്രവർത്തിപ്പിച്ച ശേഷം student2015 എന്ന പട്ടികയിലെ ഡാറ്റ പരിശോധിക്കാം. SELECT * FROM student2015; എന്ന ക്വറിയുടെ ഔട്ട്പുട്ട് ചിത്രം 9.33ൽ കാണിച്ചിരിക്കുന്നു. സംഭരണസ്ഥലം പ്രത്യേകമായി ഉപയോഗിക്കാതെ ഒരേ പട്ടിക പല പട്ടികയായി ഉപയോഗിക്കാൻ സാധിക്കുന്നു എന്ന മേൻമ വ്യൂവിൽ അടങ്ങിയിരിക്കുന്നു. പങ്കുവയ്ക്കലിനോടൊപ്പം, സ്വകാര്യതയും വ്യൂ നടപ്പാക്കുന്നു എന്നുള്ളതാണ് മറ്റൊരു നേട്ടം. പട്ടികയിൽ നിന്ന് റെക്കോർഡുകൾ വീണ്ടെടുക്കുക, പരിഷ്ക്കരിക്കുക ഒഴിവാക്കുക എന്നിവയ്ക്കായി WHERE ഉപവാക്യത്തോടൊപ്പമുള്ള നിബന്ധനകളുടെ സങ്കീർണത കുറയ്ക്കുന്നതിന് ഇത് സഹായിക്കുന്നു.

ഒരു വ്യൂ ആവശ്യമില്ലെങ്കിൽ അത് ഡാറ്റാബേസിൽ നിന്ന് ഒഴിവാക്കുന്നതിന് DROP VIEW കമാൻഡ് ഉപയോഗിക്കാം. അത് അടിസ്ഥാന പട്ടികയെ ബാധിക്കുകയില്ല. ഇതിന്റെ വാക്യഘടന താഴെ കൊടുക്കുന്നു.

```
DROP VIEW <view_name>;
```

ഉദാഹരണത്തിന് student1998 എന്ന വ്യൂ പട്ടിക ഒഴിവാക്കണമെങ്കിൽ ക്വറി 9.44 ഉപയോഗിക്കാം.

ക്വറി 9.44

```
DROP VIEW student1998;
```

നിങ്ങളുടെ പുരോഗതി അറിയുക

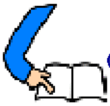
1. നെസ്റ്റഡ് ക്വറി എന്നാൽ എന്ത്?
2. SQL ലിലെ വ്യൂ എന്നാൽ എന്ത്?
3. വ്യൂ നിർമ്മിക്കുന്നതിന് SELECT കമാൻഡ് ആവശ്യമാണ്. ഈ പ്രസ്താവന ശരിയോ തെറ്റോ?
4. വ്യൂ ഒഴിവാക്കുമ്പോൾ ഡാറ്റാബേസിൽ നിന്ന് ഒരു പട്ടിക ഒഴിവാക്കുന്നു. ഇത് ശരിയോ തെറ്റോ എന്ന് പ്രസ്താവിക്കുക.
5. പട്ടിക പുതുക്കുവാൻ നമുക്ക് അതിന്റെ വ്യൂ ഉപയോഗിക്കാമോ?

നമുക്ക് ചെയ്യാം

MySQL ൽ നടപ്പിലാക്കിയിട്ടുള്ള എല്ലാ DDL, DML കമാൻഡുകളും ഉദാഹരണങ്ങളും ഔട്ട്പുട്ട് സഹിതം നാം ചർച്ച ചെയ്തു. ഇനി പട്ടിക 9.8ൽ നമുക്ക് അവ സംഗ്രഹിക്കാം. ആദ്യത്തെ രണ്ടു വരികൾ എഴുതി ചേർത്തിട്ടുണ്ട്. ബാക്കിയുള്ള വരികൾ നിങ്ങൾക്കായി മാറ്റിവച്ചിരിക്കുന്നു.

SQL	കമാൻഡ്	അനിവാര്യമായ കീവേഡ്	കമാൻഡിന്റെ ഉദ്ദേശ്യം
DDL	CREATE TABLE		ഒരു പട്ടിക നിർമിക്കാൻ
	ALTER TABLE	ADD/MODIFY	ഒരു പട്ടികയുടെ ഘടന മാറ്റുന്നതിന്
	DROP TABLE		
	CREATE VIEW		
	DROP VIEW		
DML	INSERT		
	SELECT		
	UPDATE		
	DELETE		

പട്ടിക 9.8: SQL കമാൻഡിന്റെ സംഗ്രഹം



നമുക്ക് സംഗ്രഹിക്കാം

റിലേഷണൽ ഡാറ്റാബേസിൽ പ്രവർത്തനങ്ങൾ ചെയ്യാനുപയോഗിക്കുന്ന ഭാഷയാണ് സ്ട്രക്ചേർഡ് ക്വറി ലാംഗേജ്. MySQL പ്രശസ്തമായ RDBMS പാക്കേജ് ആണ്. അത് ഉപയോഗിച്ച് നമുക്ക് നിർദേശങ്ങൾ നടപ്പിലാക്കാം. ഒരു പട്ടികയുടെ ഘടനയുമായി ബന്ധപ്പെട്ട പ്രവർത്തനങ്ങൾക്ക് വിവിധ DDL കമാൻഡുകൾ ഉണ്ട്. കൺസ്ട്രയിന്റുകൾ ഡാറ്റയുടെ സാധുതയും ഡാറ്റാബേസിന്റെ കെട്ടുറപ്പും ഉറപ്പു വരുത്തുന്നു. ഒരു പട്ടികയിലുള്ള ഡാറ്റയുമായി ബന്ധപ്പെട്ട വിവിധ പ്രവർത്തനങ്ങൾക്കുള്ള DML കമാൻഡുകൾ നാം ചർച്ച ചെയ്തു. ഡാറ്റ കൂട്ടിച്ചേർക്കൽ, വീണ്ടെടുക്കൽ, പുതുക്കൽ, ഒഴിവാക്കൽ തുടങ്ങിയ പ്രവർത്തനങ്ങൾ ഇതിൽ ഉൾപ്പെടും. നെസ്റ്റഡ് ക്വറി, വ്യൂ എന്നീ ആശയങ്ങളും എന്താണെന്ന് നാം പരിചയപ്പെട്ടു. ഈ പാഠഭാഗം നന്നായി മനസ്സിലാക്കേണ്ടത് കമ്പ്യൂട്ടർ മേഖലയിലെ നിങ്ങളുടെ ഉപരിപഠനത്തിന് ആവശ്യമാണ്.

നമുക്ക് പരിശീലിക്കാം

1. പരീക്ഷയിൽ വിദ്യാർഥികൾ നേടിയ മാർക്കും വിശദാംശങ്ങളും സംഭരിക്കുന്നതിനുള്ള പട്ടികയുടെ ഘടന താഴെ കൊടുത്തിരിക്കുന്നു.

ഡാറ്റ	ഡാറ്റ ഇനം	വിവരണം
Register number	Numeric	വിദ്യാർഥിയെ തിരിച്ചറിയാൻ ആവശ്യമുള്ളതും ഏകവുമായ ഡാറ്റ
Name	String	പരമാവധി 30 അക്ഷരങ്ങൾ
Course	String	സയൻസ്, കോമേഴ്സ് അല്ലെങ്കിൽ ഹ്യൂമാനിറ്റീസ്
Marks of six subjects	Numeric each	ആറു വ്യത്യസ്ത നിരകൾ ആവശ്യമാണ്

പട്ടിക നിർമ്മിക്കുന്നതിനും താഴെ പറയുന്ന ആവശ്യങ്ങൾക്കുമുള്ള SQL പ്രസ്താവനകൾ എഴുതുക

a) ഡാറ്റ ചേർക്കുക (10 റെക്കോർഡുകൾ).

b) എല്ലാ വിദ്യാർഥികളുടെയും വിശദാംശങ്ങൾ പ്രദർശിപ്പിക്കുക.

c) സയൻസ് ഗ്രൂപ്പിലെ എല്ലാ കുട്ടികളുടെയും വിശദാംശങ്ങൾ പട്ടികപ്പെടുത്തുക.

d) ഓരോ കോഴ്സിലെയും കുട്ടികളുടെ എണ്ണം കണക്കാക്കുക.

e) ആകെ മാർക്ക് കണ്ടുപിടിച്ച് Total എന്ന നിര പുതുക്കുക.

f) ഓരോ വിദ്യാർഥിയുടെയും ആറു മാർക്കുകളുടെ തുക ചേർത്ത് Total എന്ന നിര പുതുക്കുക.

g) ഓരോ ഗ്രൂപ്പിലെയും ഏറ്റവും ഉയർന്ന Total പ്രദർശിപ്പിക്കുക.

h) കോമേഴ്സ് ഗ്രൂപ്പിലെ ആറാമത്തെ വിഷയത്തിലെ ഏറ്റവും കുറഞ്ഞത്, ഏറ്റവും കൂടിയത്, ശരാശരി സ്കോർ എന്നിവ കണ്ടെത്തുക.

i) അക്ഷരമാലാക്രമത്തിൽ ഓരോ കോഴ്സിലെയും പേരുകൾ പ്രദർശിപ്പിക്കുക.

j) ഏറ്റവും കൂടുതൽ Total സ്കോറുള്ള കുട്ടിയുടെ പേര് പ്രദർശിപ്പിക്കുക.

2. ഒരു കമ്പ്യൂട്ടർ ഷോപ്പിലെ ഇനങ്ങളുടെ വിശദാംശങ്ങൾ സംഭരിക്കുന്നതിനുള്ള പട്ടികയുടെ ഘടന താഴെ കൊടുക്കുന്നു.

ഡാറ്റ	ഡാറ്റ ഇനം	വിവരണം
Item number	Numeric	ഒരു വസ്തു തിരിച്ചറിയാൻ തനതായതും ഏകവുമായ ഡാറ്റ
Item name	String	പരമാവധി 30അക്ഷരങ്ങൾ
Date of purchase	Date	ആവർത്തനം അനുവദിക്കുന്നു
Unit price	Fractional Number	ഒരു ഇനത്തിന്റെ വില
Quantity	Numeric	ഇനങ്ങളുടെ എണ്ണം
Manufacturer	String	വിതരണക്കാരന്റെ പേര് (ആവർത്തനം അനുവദിക്കുന്നു)

പട്ടിക നിർമ്മിക്കുന്നതിനും താഴെ പറയുന്ന ആവശ്യങ്ങൾക്കുമുള്ള SQL പ്രസ്താവനകൾ എഴുതുക.

a) ഡാറ്റ ചേർക്കുക (10 റെക്കോർഡുകൾ).

b) എല്ലാ ഇനങ്ങളുടെയും വിശദാംശങ്ങൾ പ്രദർശിപ്പിക്കുക.

c) ഓരോ ഇനത്തിന്റെ പേരും അവയുടെ വിലയും പ്രദർശിപ്പിക്കുക.

d) പട്ടികയിൽ ലഭ്യമായ ഒരു കമ്പനി നിർമ്മിച്ച (കമ്പനിയുടെ പേര് നൽകുക) ഇനങ്ങളുടെ പേര് പട്ടികപ്പെടുത്തുക.

e) ഓരോ നിർമ്മാതാവിന്റെയും ഇനങ്ങളുടെ എണ്ണം എഴുതുക.

f) ഏറ്റവും കൂടുതൽ വിലയുള്ള ഇനത്തിന്റെ വിശദാംശങ്ങൾ പ്രദർശിപ്പിക്കുക.

g) എല്ലാ ഇനങ്ങളുടെയും ശരാശരി വിലയെക്കാൾ കൂടുതൽ വിലയുള്ള ഇനങ്ങളുടെ പേര് പട്ടികപ്പെടുത്തുക.

h) 1-1-2015 നു ശേഷം വാങ്ങിയ ഇനങ്ങളുടെ പേര് പ്രദർശിപ്പിക്കുക.

i) പട്ടികയിലെ രണ്ടോ മൂന്നോ കമ്പനികൾ (കമ്പനിയുടെ പേര് നൽകുക) നിർമ്മിച്ച ഇനങ്ങളുടെ വിശദാംശങ്ങൾ ലഭ്യമാക്കുക.

j) ഒരു കമ്പനിയിലെ 20 എണ്ണത്തിൽ കൂടുതൽ സ്റ്റോക്കുള്ള ഇനങ്ങളുടെ വിശദാംശങ്ങൾ പ്രദർശിപ്പിക്കുക.

3. ഹയർ സെക്കന്ററി സ്കൂളിലെ അധ്യാപകരുടെ വിശദാംശങ്ങൾ സംഭരിക്കുന്നതിനുള്ള പട്ടികയുടെ ഘടന കൊടുക്കുന്നു.

ഡാറ്റ	ഡാറ്റ ഇനം	വിവരണം
Teacher ID	Numeric	ടീച്ചറെ തിരിച്ചറിയാൻ തനതായതും ഏകവുമായ ഡാറ്റ
Name	String	പരമാവധി 30അക്ഷരങ്ങൾ
Gender	Character	ആൺ അല്ലെങ്കിൽ പെൺ
Date of joining	Date	ആവർത്തനം അനുവദിക്കുന്നു
Department	String	സയൻസ്, കോമേഴ്സ്, ഹ്യൂമാനിറ്റീസ് അല്ലെങ്കിൽ ഭാഷ
Basic pay	Numeric	അധ്യാപകരുടെ അടിസ്ഥാന ശമ്പളം

പട്ടിക നിർമ്മിക്കുന്നതിനും താഴെപ്പറയുന്ന ആവശ്യങ്ങൾക്കുമുള്ള SQL പ്രസ്താവനകൾ എഴുതുക.

a) ഡാറ്റ ചേർക്കുക (10 റെക്കോർഡുകൾ).

b) എല്ലാ അധ്യാപികമാരുടെയും വിശദാംശങ്ങൾ പട്ടികപ്പെടുത്തുക.

c) സയൻസ് ഡിപ്പാർട്ട്മെന്റിലെ പുരുഷ അധ്യാപകരുടെ വിശദാംശങ്ങൾ പട്ടികപ്പെടുത്തുക.

d) അടിസ്ഥാന ശമ്പളം Rs 50000 രൂപയോ അതിൽ കൂടുതലോ ആയിട്ടുള്ള ഭാഷാധ്യാപകരുടെ പേരും അടിസ്ഥാന ശമ്പളവും പ്രദർശിപ്പിക്കുക.

e) അധ്യാപകരുടെ പേരും അടിസ്ഥാന ശമ്പളത്തിന്റെ 71% വും പ്രദർശിപ്പിക്കുക.

f) ഓരോ വിഭാഗത്തിലെയും അധ്യാപകരുടെ എണ്ണം കണ്ടുപിടിക്കുക.

g) ഒരു സ്കൂളിലെ അധ്യാപകരുടെ ശരാശരി അടിസ്ഥാന ശമ്പളത്തെക്കാൾ കുറവ് ശമ്പളമുള്ള അധ്യാപകരുടെ വിശദാംശങ്ങൾ പ്രദർശിപ്പിക്കുക.

h) 1-1-2010 ന് മുൻപ് ചേർന്ന പുരുഷ അധ്യാപകരുടെ പട്ടിക തയാറാക്കുക.

i) എല്ലാ അധ്യാപകരുടേയും അടിസ്ഥാന ശമ്പളം 1000 രൂപ വർധിപ്പിക്കുക.

j) ഭാഷാ വിഭാഗത്തിലെ അധ്യാപകരുടെ വിശദാംശങ്ങൾ ഒഴിവാക്കുക.

4. ഒരു ബാങ്കിലെ ഉപഭോക്താക്കളുടെ വിശദാംശങ്ങൾ സംഭരിക്കുന്നതിനുള്ള പട്ടികയുടെ ഘടന കൊടുക്കുന്നു.

ഡാറ്റ	ഡാറ്റ ഇനം	വിവരണം
Account number	Numeric	തിരിച്ചറിയാൻ തനതായതും ഏകവുമായ ഡാറ്റ
Name	String	പരമാവധി 30 അക്ഷരങ്ങൾ
Gender	Character	ആൺ അല്ലെങ്കിൽ പെൺ
Date of joining	Date	ആവർത്തനം അനുവദിക്കുന്നു
Type of account	String	SB അല്ലെങ്കിൽ Current അക്കൗണ്ട്
Balance amount	Numeric	ദശാംശ സംഖ്യകൾ അനുവദിക്കുന്നു

പട്ടിക നിർമ്മിക്കുന്നതിനും താഴെ പറയുന്ന ആവശ്യങ്ങൾക്കുമുള്ള SQL പ്രസ്താവനകൾ എഴുതുക.

a. ഡാറ്റ ചേർക്കുക (10 റെക്കോർഡുകൾ).

b. SB അക്കൗണ്ടുള്ള ഉപഭോക്താവിന്റെ വിശദാംശങ്ങൾ പ്രദർശിപ്പിക്കുക.

c. ബാലൻസ് തുക 5000 രൂപയിൽ കൂടുതൽ ഉള്ള വനിത ഉപഭോക്താക്കളുടെ പേര് പ്രദർശിപ്പിക്കുക.

d. ബാലൻസ് തുക Rs 10000 രൂപയെക്കാൾ കൂടുതൽ ഉള്ള സ്ത്രീകളായ ഉപഭോക്താക്കളുടെ വിശദാംശങ്ങൾ നൽകുക.

e. സ്ത്രീ, പുരുഷ ഉപഭോക്താക്കളുടെ എണ്ണം കണ്ടുപിടിക്കുക.

f. ഏറ്റവും കൂടുതൽ ബാലൻസ് തുക ഉള്ള ഉപഭോക്താക്കളുടെ പേര് പ്രദർശിപ്പിക്കുക.

g. Kumar എന്ന് അവസാനിക്കുന്ന ഉപഭോക്താക്കളുടെ പേര് പ്രദർശിപ്പിക്കുക.

h. 2000 രൂപ നീക്കിയിരിപ്പുള്ള ഉപഭോക്താക്കളുടെ ബാലൻസ് തുക പുതുക്കുക.

i. 20000 രൂപയോ അതിലധികമോ നീക്കിയിരിപ്പുള്ള ഉപഭോക്താക്കളുടെ അക്കൗണ്ടിൽ നിന്നും 2% നികുതി കുറച്ചതിനു ശേഷമുള്ള വിശദാംശങ്ങൾ പ്രദർശിപ്പിക്കുക.

j. കറന്റ് അക്കൗണ്ട് ഉള്ള ഉപഭോക്താക്കളുടെ വിശദാംശങ്ങൾ ഒഴിവാക്കുക.

നമുക്ക് വിലയിരുത്താം

1. CUSTOMER എന്ന പട്ടികയിലെ വരികൾ ഒഴിവാക്കാൻ ഉപയോഗിക്കുന്ന കമാൻഡ് ഏത്?

   a. `REMOVE FROM CUSTOMER`

   b. `DROP TABLE CUSTOMER`

   c. `DELETE FROM CUSTOMER`

   d. `UPDATE CUSTOMER`

2. ഒരു നിരയിലെ ചില വിലകൾ അറിയില്ലെങ്കിൽ ഇതിലേക്ക് ഒരു വരി എങ്ങനെ ചേർക്കാം.

3. `CHAR`, `VARCHAR` എന്നീ SQL ഡാറ്റ ഇനങ്ങൾ തമ്മിലുള്ള വ്യത്യാസം കണ്ടെത്തുക.

4. `PRIMARY KEY`, `UNIQUE` എന്നീ കൺസ്ട്രെയിന്റുകൾ തമ്മിലുള്ള വ്യത്യാസം എന്ത്?

5. SQL ലെ `NULL` മൂല്യം എന്നത്കൊണ്ട് അർഥമാക്കുന്നത് എന്ത്?

6. `SELECT` പ്രസ്താവനയുടെ ഉപവാക്യങ്ങളുടെ കൃത്യമായ ക്രമീകരണം ഏത് ?

   a. `SELECT, FROM, WHERE, ORDER BY`

d. SELECT, FROM, ORDER BY, WHERE

c. SELECT, WHERE, FROM, ORDER BY

d. SELECT, WHERE, ORDER BY, FROM

7. പാറ്റേൺ തുല്യതയ്ക്ക് ഉപയോഗിക്കുന്ന SQL ഓപ്പറേറ്ററാണ് ________.

8. താഴെ കൊടുത്തിരിക്കുന്ന സ്ട്രിംഗുകൾ വായിക്കുക.

(i) 'Sree Kumar' (ii) 'Kumaran' (iii) 'Kumar Shanu' (iv) 'Sreekumar'

SELECT പ്രസ്താവനയിൽ LIKE ഓപ്പറേറ്ററിന്റെ കൂടെ '%kumar' എന്ന പാറ്റേൺ കൃത്യമായി തുല്യത പ്രാപിക്കുന്നത് ഏതിനോടാണ് ?

a. സ്ട്രിംഗ് (i) ഉം സ്ട്രിംഗ് (ii) ഉം മാത്രം

b. സ്ട്രിംഗ് (i) ഉം സ്ട്രിംഗ് (iii) ഉം മാത്രം

c. സ്ട്രിംഗ് (i), (iii), (iv) മാത്രം

d. എല്ലാ സ്ട്രിംഗുകളും

9. SQL ലെ ഏതെങ്കിലും അഞ്ച് ബിൽട്ട് ഇൻ ഫങ്ഷനുകളും അവ ഓരോന്നും തരുന്ന മൂല്യങ്ങളും പട്ടികപ്പെടുത്തുക.

10. WHERE, HAVING തമ്മിലുള്ള ഏതെങ്കിലും നാല് വ്യത്യാസങ്ങൾ എഴുതുക.

11. SQL ലെ ഏതെങ്കിലും നാല് DML കമാൻഡുകൾ എഴുതുക

12. താഴെ പറയുന്ന ഓരോ SQL കമാൻഡിന്റെയും ഉപവാക്യങ്ങൾ എഴുതുക

a. INSERT INTO b. SELECT c. UPDATE

13. Customer എന്ന പട്ടിക പരിഗണിച്ച് താഴെ കൊടുത്തിരിക്കുന്ന SQL ക്വറികളുടെ ഔട്ട്പുട്ട് എഴുതുക

AccNo	Name	Branch	Amount
1001	Kumar	Calicut	10000
1002	Salim	Trivandrum	20000
1003	Fida	Kottayam	18000
1004	John	Kannur	30000
1005	Raju	Thrissur	5000

a. SELECT * FROM customer WHERE Amount>25000;

b. SELECT Name FROM customer WHERE Branch IN ('Calicut',' Kannur');

c. SELECT COUNT(*) FROM customer WHERE Amount< 20000;

d. SELECT Name FROM customer WHERE Name LIKE "%m%";

e. SELECT *FROM customer ORDER BY Amount DESC;

14. COUNT(*) ഉം COUNT(column_name) ഉം തമ്മിലുള്ള വ്യത്യാസം എഴുതുക.

15. ITEM എന്ന പട്ടിക പരിഗണിക്കുക.

Item Code	Name	Category	Unit Price	Sales Price
0001	Pencil	Stationery	5.00	8.00
0002	Pen	Stationery	8.00	10.00
0003	NoteBook	Stationery	10.00	20.00
0004	Chappal	Footwear	50.00	70.00
0005	Apple	Fruits	60.00	90.00
0006	Orange	Fruits	40.00	60.00
0007	Pen	Stationery	10.00	12.00

a. ഈ പട്ടികയ്ക്ക് അനുയോജ്യമായ ഒരു പ്രാഥമിക കീ നിർദേശിക്കുക. ഉത്തരം സാധൂകരിക്കുക.

b. താഴെ പറയുന്ന ആവശ്യങ്ങൾക്കുള്ള SQL പ്രസ്താവനകൾ എഴുതുക.

i) എല്ലാ സ്റ്റേഷനറി ഇനങ്ങളും പട്ടികപ്പെടുത്തുക.

ii) itemcode, name, profit എന്നിവ പട്ടികപ്പെടുത്തുക.

iii) ഓരോ വിഭാഗത്തിലുമുള്ള ഇനങ്ങളുടെ എണ്ണം കണ്ടുപിടിക്കുക.

iv) യൂണിറ്റ് വിലയുടെ അടിസ്ഥാനത്തിൽ എല്ലാ സ്റ്റേഷനറി ഇനങ്ങളും അവരോഹണക്രമത്തിൽ പട്ടികപ്പെടുത്തുക.

v) വിൽപ്പനവില ഏറ്റവും കൂടുതൽ ഉള്ള ഇനങ്ങൾ കണ്ടുപിടിക്കുക.

vi) എല്ലാ സ്റ്റേഷനറി ഇനങ്ങളുടെയും വിശദാംശങ്ങൾ അടങ്ങിയ വ്യൂ ഉണ്ടാക്കുക.

16. ഒരു പട്ടികയുടെ ഘടനയിൽ നമുക്ക് ചെയ്യാൻ സാധിക്കുന്ന വിവിധ പ്രവർത്തനങ്ങൾ എന്തൊക്കെ? ഇതിന് ഏത് കമാൻഡ് ഉപയോഗിക്കണം? ഓരോന്നിന്റെയും മാറ്റത്തിന് ഉപയോഗിക്കുന്ന ഉപവാക്യങ്ങൾ എഴുതുക.

17. SQLൽ 10 റെക്കോർഡുകളോടുകൂടി ഒരു പട്ടിക നിർമ്മിച്ചിട്ടുണ്ടെന്ന് വിചാരിക്കുക. നിർദിഷ്ട വരിയിലെ നിരകളിലുള്ള മൂല്യങ്ങൾക്ക് മാറ്റം വരുത്താൻ ഏത് SQL കമാൻഡാണ് ഉപയോഗിക്കുന്നത്? അതിന്റെ ഘടന എഴുതുക.

18. ഒരു നിരയിൽ ആവർത്തനം ഒഴിവാക്കുന്നതിന് വേണ്ടി SELECT കമാൻഡിന്റെ കൂടെ ഏത് കീ വേഡാണ് ഉപയോഗിക്കുന്നത്? കമാൻഡിന്റെ പേരെഴുതുക.

19. SQL ൽ DISTINCT ഉം UNIQUE ഉം തമ്മിലുള്ള വ്യത്യാസം എഴുതുക.

20. കൂട്ടത്തിൽപ്പെടാത്തത് ഏത്? കാരണം എഴുതുക.

a. CREATE   b. SELECT   c. UPDATE   d. INSERT

Q7I1N3

10 PHP ഉപയോഗിച്ചുള്ള സെർവർ ഭാഗത്തെ സ്ക്രിപ്റ്റിങ്

പ്രധാന പഠനനേട്ടങ്ങൾ

ഈ അധ്യായത്തിന്റെ പഠനം പൂർത്തിയാകുന്നതോടെ പഠിതാവ് ആർജിക്കേണ്ട പഠനനേട്ടങ്ങൾ

- PHP ഉപയോഗിക്കുന്നത്കൊണ്ടുള്ള ഗുണങ്ങൾ പട്ടികപ്പെടുത്തുന്നു
- PHP പ്രോഗ്രാമിന്റെ വാക്യഘടന (Syntax) വിശദീകരിക്കുന്നു
- ക്ലയന്റ് - സെർവർ (Client-Server) സാഹചര്യം വിവരിക്കുന്നു
- PHP - യിൽ ഉപയോഗിച്ചിരിക്കുന്ന വിവിധ ഡാറ്റ ഇനങ്ങളും (Datat ypes) ഓപ്പറേറ്ററുകളും വിവരിക്കുന്നു.
- PHP - യിൽ ലഭ്യമായ കൺട്രോൾ സ്ട്രക്ച്ചറുകളും, ലൂപ്പുകളും ഉപയോഗിച്ച് അൽഗോരിതത്തിൽനിന്നും കോഡുകൾ നിർമ്മിക്കുന്നു
- സാധാരണ ജോലികൾക്കുള്ള PHP ഫങ്ഷനുകൾ രൂപകൽപന ചെയ്യുന്നു
- തീയതി, സമയം, സ്ട്രിംഗ് തുടങ്ങിയവയുമായി ബന്ധപ്പെട്ട ഫങ്ഷനുകൾ തിരിച്ചറിയുന്നു
- സ്റ്റാറ്റിക്, ഡൈനാമിക് വെബ്പേജുകളെ വേർതിരിച്ചറിയുന്നു
- വിവിധ സാഹചര്യങ്ങൾക്ക് അനുയോജ്യമായ ഡാറ്റ സമർപ്പിക്കൽ രീതികൾ തിരഞ്ഞെടുക്കുന്നു
- ഡൈനാമിക് വെബ്പേജുകളുടെ നിർമ്മാണത്തിൽ ഡാറ്റാബേസ് ബന്ധിപ്പിക്കുന്നതിന്റെ പങ്ക് തിരിച്ചറിയുന്നു

മുൻ അധ്യായങ്ങളിൽ HTML-നെ കുറിച്ചും ജാവസ്ക്രിപ്റ്റ് ഉപയോഗിച്ചുള്ള ക്ലയന്റ് ഭാഗത്തെ സ്ക്രിപ്റ്റിംഗിനെകുറിച്ചും നാം പഠിച്ചു. HTML പ്രോഗ്രാമുകളിൽ ഉൾക്കൊള്ളിക്കുന്ന (Embed) ചെറു പ്രോഗ്രാമുകളാണ് സ്ക്രിപ്റ്റ് (Script) എന്ന് നമുക്ക് അറിയാം. ജാവാസ്ക്രിപ്റ്റ് HTML പ്രോഗ്രാമുകളിൽ ഉൾക്കൊള്ളിക്കുന്ന രീതിയും നാം മനസ്സിലാക്കിക്കഴിഞ്ഞു. ഒരു സെർവർ ഭാഗത്തെ സ്ക്രിപ്റ്റിംഗ് ഭാഷയായ PHP- യെക്കുറിച്ചും അത് ജാവസ്ക്രിപ്റ്റിനോടൊപ്പം HTML പ്രോഗ്രാമുകളിൽ ഉൾക്കൊള്ളിക്കുന്ന രീതിയുമാണ് ഈ അധ്യായത്തിൽ പ്രതിപാദിക്കുന്നത്. PHP ഭാഷയുടെ അടിസ്ഥാന വാക്യഘടന C++, ജാവാ സ്ക്രിപ്റ്റ് (Java Script) എന്നീ പ്രോഗ്രാമിങ് ഭാഷകൾക്ക് സമാനമാണ്. HTML-ൽ ഉൾപ്പെടുത്തുന്ന രീതിയും ജാവാസ്ക്രിപ്റ്റിലേതുപോലെയാണ്. ആയതിനാൽ PHP പഠനം സുഗമവും വളരെ രസകരവും ആയിരിക്കും.

10.1 PHP യെ കുറിച്ച് ഒരു അവലോകനം (Overview of PHP)

ലഭ്യമായ ഏറ്റവും പ്രചാരമുള്ള സെർവർ സൈഡ് സ്ക്രിപ്റ്റിംഗ് ഭാഷകളിലൊന്നാണ് PHP. ഡൈനാമിക് വെബ്പേജുകൾ നിർമിക്കുന്നതിന് ഇത് ഉപയോഗിക്കുന്നു. ഇതിന്റെ പൂർണ്ണരൂപം PHP ഹൈപ്പർ ടെക്സ്റ്റ് പ്രീപ്രൊസസർ (PHP Hyper-text Preprocessor) എന്നാണ്. സാധാരണയായി ഒരു വെബ് സെർവറിൽ PHP പ്രവർത്തിക്കുന്നു. ഇത് വ്യത്യസ്ത ഓപ്പറേറ്റിങ് സിസ്റ്റങ്ങളിലും പ്ലാറ്റ്ഫോമിലും ലഭ്യമാണ്. കൂടാതെ, PHP ഒരു ഇന്റർപ്രെറ്റർ മുഖേന പ്രവർത്തിക്കുന്ന ഭാഷയാണ്. ഓരോ തവണയും PHPഎൻജിൻ പ്രവർത്തിക്കുന്നതിനോടൊപ്പം PHP സ്ക്രിപ്റ്റും പ്രവർത്തിക്കുന്നു. ഒരു HTML പ്രോഗ്രാമിനുള്ളിൽ PHP

സ്ക്രിപ്റ്റ് എഴുതുവാൻ സാധിക്കും. അതിനാലാണ് PHP, HTML -ൽ ഉൾപ്പെടുത്താവുന്ന (embedded) ഭാഷയായി അറിയപ്പെടുന്നത്. ഒരു PHP പ്രോഗ്രാം പ്രവർത്തിക്കുമ്പോൾ നിർദ്ദേശങ്ങളുടെ പ്രോസസ്സിംഗ് നടന്നതിന് ശേഷം ഹൈപ്പർ ടെക്സ്റ്റ് രൂപത്തിലാണ് ഔട്ട്പുട്ട് ലഭിക്കുന്നത്.

റാസ്മസ് ലെർഡ്ഓഫ് (ജനനം 22 നവംബർ 1968) എന്ന ഗ്രീൻ ലാൻഡുകാരനായ പ്രോഗ്രാമറാണ് PHP നിർമിച്ചത്. ഇദ്ദേഹം PHP - യുടെ പിതാവായി അറിയപ്പെടുന്നു. 1995 - ൽ ഈ ഭാഷനിർമ്മാണം പൂർത്തിയായെങ്കിലും ഔദ്യോഗികമായി പ്രകാശനം ചെയ്തത് 1997 - ലാണ്. ഈ ഭാഷയുടെ ആദ്യ രണ്ടു പതിപ്പുകൾ ഇദ്ദേഹം എഴുതുകയും തുടർന്നുള്ള പതിപ്പുകളുടെ വികസനത്തിൽ പങ്കെടുക്കുകയും ചെയ്തു. ഈ പദ്ധതിയിൽ അദ്ദേഹം തുടർന്നും സഹകരിച്ച് കൊണ്ടിരിക്കുന്നു. www.php.net ആണ് PHP യുടെ ഔദ്യോഗിക വെബ്സൈറ്റ്.

റാസ്മസ് ലെർഡ് ഓഫ്

PHP ഒരു സ്വതന്ത്ര സോഫ്ട്വെയർ പദ്ധതിയാണ്. ഇതു സ്വന്തമാക്കാൻ വില നൽകേണ്ടതില്ല. ലോകമെമ്പാടുമുള്ള സ്വതന്ത്ര സോഫ്റ്റ്വെയർ നിർമ്മാതാക്കളും സന്നദ്ധ പ്രവർത്തകരും ഇതിനെ പിന്തുണയ്ക്കുന്നു. സർവർ ഭാഗത്തെ മറ്റ് ഭാഷകളേക്കാൾ കാര്യക്ഷമത ഉള്ളതും, തെറ്റുകൾ കടന്നുകൂടാൻ സാധ്യത കുറവുള്ളതും, MySQL അടക്കം ഇരുപതിൽ പരം ഡാറ്റാബേസ് സോഫ്ട്വെയറുകളുമായി ബന്ധിപ്പിക്കുവാൻ സാധിക്കുന്നതുമാണ് PHP. ഇമേജുകൾ, PDF ഫോർമാറ്റിൽ തയാറാക്കിയ ഫയലുകൾ, ഫ്ളാഷ് വീഡിയോകൾ തുടങ്ങി വിവിധതരം ഔട്ട്പുട്ടുകൾ കൈകാര്യം ചെയ്യുവാൻ PHP-യ്ക്ക് സാധിക്കും. ഈ സോഫ്ട്വെയർ സ്വതന്ത്രവും സൗജന്യമായി ലഭിക്കുന്നതുമായതിനാൽ ഇതുപയോഗിച്ച് വെബ് അപ്ലിക്കേഷനുകൾ തയ്യാറാക്കുന്ന സ്ഥാപനങ്ങൾക്ക് ലൈസൻസ് ഫീസിനത്തിലും ചെലവേറിയ സെർവർ ഹാർഡ്വെയർ വാങ്ങുന്ന ഇനത്തിലും ഉള്ള ലാഭം കൊണ്ട് നിർമാണച്ചിലവ് ഗണ്യമായി കുറക്കുവാൻ സാധിക്കുന്നു. അതേസമയം ഈ സോഫ്റ്റ്വെയർ ഉപയോഗിച്ച് ഗുണമേന്മയുള്ള ഉല്പന്നങ്ങൾ ചുരുങ്ങിയ സമയത്തിനുള്ളിൽ ചെയ്തു തീർക്കുവാനും സാധിക്കുന്നു. തത്ഫലമായി ഇന്ന് വൻകിട വെബ്സൈറ്റ് രൂപകല്പനകൾ കൂടുതലായി നടക്കുന്നത് PHP ഉപയോഗിച്ചാണ്.

ഏറ്റവും പുതിയ സ്ഥിതിവിവര കണക്ക് അനുസരിച്ച് 20 മില്യനിലധികം വെബ് സൈറ്റുകളിലും ഏകദേശം 1 മില്യൺ വെബ് സെർവറുകളിലും PHP ഉപയോഗിച്ച് വരുന്നു.

10.1.1 PHP ഉപയോഗിക്കുന്നതു കൊണ്ടുള്ള പ്രയോജനങ്ങൾ (Benefits of PHP)

ക്ലയന്റ് ഭാഗത്തെ സ്ക്രിപ്റ്റുകൾ ക്ലയന്റിൽ തന്നെ പ്രവർത്തിക്കുന്നതിനാൽ അത് വളരെ വേഗത്തിൽ പ്രവർത്തിക്കുന്നതായി നമുക്ക് അറിയാം. വളരെ വേഗത ഉള്ളതാണെങ്കിലും അതിന് ചില പരിമിതികൾ ഉണ്ട്. ക്ലയന്റ് ഭാഗത്തെ സ്ക്രിപ്റ്റിംഗ് ഭാഷ ബ്രൗസറിനെ ആശ്രയിച്ചാണ് പ്രവർത്തിക്കുന്നത്. തന്നെയുമല്ല ബ്രൗസറിലെ 'View Source' പോലുള്ള സങ്കേതങ്ങൾ ഉപയോഗിച്ച് ഉപയോക്താവിന് ജാവാ സ്ക്രിപ്റ്റ് കോഡ് കാണുന്നതിനും സാധിക്കും. ഇത് സുരക്ഷാ പ്രശ്നങ്ങൾ സൃഷ്ടിക്കുന്നു.

എന്നാൽ PHP സ്ക്രിപ്റ്റ് പ്രവർത്തിക്കുന്നത് ക്ലയന്റ് ബ്രൗസറിലല്ല, മറിച്ച് സെർവറിലാണ്. ബ്രൗസറിനെ ആശ്രയിക്കാതെ ഒരു കോഡ് എഴുതുന്നതിന് ഇത് നമ്മെ സഹായിക്കുന്നു. സെർവറിൽ പ്രവർത്തിച്ച ശേഷം ക്ലയൻറ് കമ്പ്യൂട്ടറിലേക്ക് പ്രോഗ്രാമിന്റെ ഔട്ട്പുട്ട് മാത്രമാണ് എത്തുന്നത്. അതു കൊണ്ട് PHP പ്രോഗ്രാമിന്റെ സോഴ്സ് കോഡ് കാണാൻ ഉപയോക്താവിന് കഴിയില്ല.

PHP യുടെ പ്രയോജനങ്ങൾ സംക്ഷിപ്തമായി ചുവടെ ചേർക്കുന്നു :

- PHP വളരെ എളുപ്പത്തിൽ HTML പ്രോഗ്രാമിൽ ഉൾക്കൊള്ളിക്കുവാൻ സാധിക്കുന്നതിനാൽ സ്റ്റാറ്റിക് വെബ് പേജുകളെ ഡയനാമിക് വെബ് പേജുകളാക്കിമാറ്റാൻ അനായാസം സാധിക്കുന്നു.
- ഇത് വിവിധതരം ഡാറ്റാബേസ് സോഫ്റ്റ്‌വെയറുകളുമായി യോജിച്ച് പ്രവർത്തിക്കുന്നു.
- ഇത് ഒരു സ്വതന്ത്ര ചട്ടക്കൂടിലധിഷ്ഠിതവും വിൻഡോസ്, ലിനക്സ്, മാക് തുടങ്ങിയ വെബ് സെർവറുകളിൽ പ്രവർത്തിക്കാൻ കഴിയുന്നതുമാണ്.
- ഇത് ഏറെ പ്രചാരത്തിലുള്ള വെബ് സെർവറുകളെ പിന്തുണക്കുന്നതും Apache-MySQL മായി അനായാസം യോജിച്ച് പ്രവർത്തിക്കുന്നതുമാണ്.
- ഇത് ലളിതവും പഠിക്കാൻ എളുപ്പവുമാണ്.
- മറ്റ് സ്ക്രിപ്റ്റ് ഭാഷകളെ അപേക്ഷിച്ച് വേഗത്തിൽ പ്രവർത്തിക്കുന്നു. കാരണം പ്രോഗ്രാം പ്രവർത്തിപ്പിക്കുന്നതിന് കംപ്യൂട്ടറിലെ അടിസ്ഥാന വിഭവങ്ങൾ കുറച്ചുമാത്രമേ ഉപയോഗിക്കേണ്ടതുള്ളൂ.
- വൈറസ്സുപോലെയുള്ള അപകടകാരികളായ സോഫ്ട്‌വെയറിൽ നിന്നുമുള്ള ആക്രമണങ്ങൾ തടയുന്നതിന് കൂടുതൽ സുരക്ഷ PHP ഉറപ്പ് വരുത്തിയിരിക്കുന്നു.
- സ്വതന്ത്ര സോഫ്ട്‌വെയർ നിർമ്മാണ കൂട്ടായ്മയിൽ നിന്നുള്ള വലിയ പിന്തുണ ഇതിനുണ്ട്.
- PHP മുഴുവനായും സൗജന്യമായ ഒരു സ്വതന്ത്ര സോഫ്ട്‌വെയർ ഉൽപ്പന്നമായതുകൊണ്ട് വ്യക്തിഗതമോ വ്യാവസായികമോ ആയ വെബ്സൈറ്റ് വികസനത്തിന് നിയമാനുസൃതമായി ഉപയോഗിക്കാവുന്നതാണ്.

> കേരള സംസ്ഥാന ഹയർസെക്കണ്ടറി വിദ്യാഭ്യാസ വകുപ്പിന്റെ ഏക ജാലക പ്രവേശന പോർട്ടലായ www.hscap.kerala.gov.in - ൽ PHP പ്രവർത്തിക്കുന്നു.

10.1.2 PHP യുടെ അടിസ്ഥാന പ്രമാണങ്ങൾ (Basics of PHP)

A. പ്രോഗ്രാമിങ്ങ് വികസനത്തിനുള്ള പരിസ്ഥിതി ഒരുക്കൽ (Setting up the development enviornment)

വെബ് പേജ് നിർമ്മാണത്തിന് ആവശ്യമായ വിഭവങ്ങൾ കമ്പ്യൂട്ടറിൽ ഒരുക്കേണ്ടത് വളരെ അത്യാവശ്യമാണ്. ആദ്യമായി PHP ക്ക് അനുയോജ്യമായ വെബ് സെർവറും ഇന്റെർപ്രെട്ടറും നമ്മുടെ കമ്പ്യൂട്ടറിൽ ഇൻസ്റ്റാൾ ചെയ്യേണ്ടതാണ്. ഇവ വെവ്വേറെയോ അല്ലെങ്കിൽ Apache വെബ്സെർവെർ, MySQL, PHP അടങ്ങിയ WAMP, LAMP, അല്ലെങ്കിൽ XAMPP പോലെയുള്ള സൗജന്യ സ്വതന്ത്ര സോഫ്ട്‌വെയർ പാക്കേജുകളുടെ കൂട്ടത്തിൽ നിന്നോ ഇൻസ്റ്റാൾ ചെയ്യാവുന്നതാണ്. WAMP വിൻഡോസിലും, LAMP ലിനക്സിലും XAMPP രണ്ടിലും പ്രവർത്തിക്കുന്നു.

വെബ് സെർവർ കമ്പ്യൂട്ടറിൽ പ്രവർത്തിക്കുന്നുണ്ടെന്ന് ഉറപ്പുവരുത്തിയതിനു ശേഷം മെനുവിൽ നിന്ന് വെബ് ബ്രൗസർ തുറക്കുക. അതിലെ അഡ്രസ്സ് ബാറിൽ http://localhost എന്നോ http://127.0.0.1 എന്നോ ടൈപ്പ് ചെയ്ത് ENTER അമർത്തുക. അപ്പോൾ വെബ്സെർവർ സോഫ്ട്‌വെയറിന്റെ തനതായ ഒരു അഡ്മിനിസ്ട്രേഷൻ ജാലകം ദൃശ്യമാകുന്നുവെങ്കിൽ സെർവർ ഇൻസ്റ്റലേഷൻ കൃത്യമായി എന്നുറപ്പിക്കാം.

B. സ്ക്രിപ്റ്റ് എഴുത്തും പ്രവർത്തനവും (Writing and running the script)

PHP-നിർദേശങ്ങൾ അടങ്ങുന്ന ടെക്സ്റ്റ് ഫയലുകളാണ് PHP സ്ക്രിപ്റ്റുകൾ. PHP കോഡ് എഴുതുവാൻ ഒരു സാധാരണ ടെക്സ്റ്റ് എഡിറ്റർ കൂടാതെ മറ്റൊരു പ്രത്യേക സോഫ്റ്റ് വെയറിന്റെയും ആവശ്യമില്ല. എല്ലാ PHP കോഡുകളും തുടങ്ങുന്നത് ‘<?php’ കൊണ്ടും അവസാനിക്കുന്നത് ‘?>’ കൊണ്ടുമാണ്. വെബ് സെർവർ ഈ ടാഗ് കാണുമ്പോൾ ഒരു PHP കോഡാണ് ഈ ടാഗുകൾക്കിടയിൽ എഴുതിയിരിക്കുന്നത് എന്ന് മനസ്സിലാക്കുകയും അതിനെ പ്രവർത്തിപ്പിക്കുകയും ചെയ്യുന്നു. PHP കോഡ് അടങ്ങുന്ന ടെക്സ്റ്റ് ഫയലുകളെ PHP എന്ന എക്സ്റ്റൻഷൻ ഉപയോഗിച്ച് വെബ് സെർവറിന്റെ റൂട്ട് ഡയറക്ടറിയിൽ (root direction) സേവ് (save) ചെയ്യേണ്ടതാണ്.

WAMP, XAMPP and LAMP

WAMP, LAMP, XAMPP എന്നിവയാണ് PHP വെബ് സൈറ്റുകൾക്ക് ആധാരമായിട്ടുള്ളത്. അതോടൊപ്പം ഇവ ലോക്കൽ സെർവറായും പ്രവർത്തിക്കുന്നു. ഇക്കാരണത്താൽ നാം നിർമ്മിക്കുന്ന വെബ് സൈറ്റ് നമ്മുടെ കമ്പ്യൂട്ടറിൽ തന്നെ പ്രവർത്തിച്ചു കാണുവാനും സാധിക്കും. വെബ് സൈറ്റ് നിർമ്മാണത്തിന്റെ സങ്കീർണത കുറയ്ക്കുവാനും ഡാറ്റാബേസ് ഉപയോഗം, പ്രോഗ്രാം നിർമ്മാണം മുതലായ സേവനങ്ങൾ എന്നിവ സുഗമമായി ചെയ്യുവാനും ഇത്തരം സോഫ്റ്റ്‌വെയറുകൾ ഉപകാരപ്രദമാണ്.

Apache, MySQL, PHP എന്നിവയെ സൂചിപ്പിക്കുന്നതാണ് WAMP. വിൻഡോസ് ഓപ്പറേറ്റിംഗ് സിസ്റ്റം ഉപയോഗിക്കുന്ന കമ്പ്യൂട്ടറുകളിൽ ഇൻസ്റ്റാൾ ചെയ്തിരിക്കുന്ന സ്വതന്ത്ര പാക്കേജുകളാണ് WAMP-കൾ. കമ്പ്യൂട്ടറിൽ വേഗം സജ്ജമാക്കാൻ കഴിയുന്നു എന്നതാണ് WAMP ഉപയോഗിക്കുന്നതിന്റെ ഗുണം. www.wampserver.com/en/ എന്ന സൈറ്റിൽ WAMP ലഭ്യമാണ്.

X-OS,Apache, MySQL, PHP, PERL എന്നിവയെ സൂചിപ്പിക്കുന്നതാണ്. XAMPP. X-OS എന്നതുകൊണ്ട് അർത്ഥമാക്കുന്നത് ഇത് എല്ലാത്തരം ഓപ്പറേറ്റിംഗ് സിസ്റ്റത്തിലും ഉപയോഗിക്കാം എന്നതാണ്. തുടക്കക്കാർക്ക് ഈ സോഫ്റ്റ്‌വെയർ വളരെ എളുപ്പം ഉപയോഗിക്കാം. http://www. apachefriends.org/en/xampp-windows.html എന്ന സൈറ്റിൽ ഇത് ലഭ്യമാണ്.

LINUX, Apache, MySQL, PHP എന്നിവയെ സൂചിപ്പിക്കുന്നതാണ് LAMP. ലിനക്സ് ഓപ്പറേറ്റിംഗ് സിസ്റ്റത്തെ ഉപയോഗപ്പെടുത്തുന്ന ഒരു സ്വതന്ത്ര വെബ് വികസന ചട്ടക്കൂടാണ് LAMP. ഈ ചട്ടക്കൂടിന് നാലു തലങ്ങൾ ഉള്ളതിനാൽ ചിലപ്പോൾ ഒരു LAMP ശേഖരം എന്നും ഇതിനെ സൂചിപ്പിക്കുന്നു.

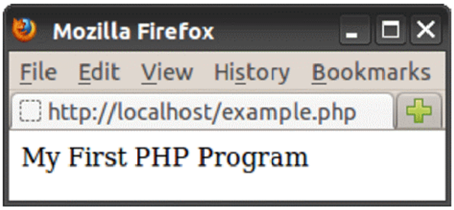
ഘട്ടം ഘട്ടമായി ഒരു ഉദാഹരണം നമുക്ക് ചെയ്തു നോക്കാം.

ഘട്ടം 1: ടെക്സ്റ്റ് എഡിറ്ററിൽ ഒരുപുതിയ ജാലകം തുറക്കുക . ഇതിനായി ടെക്സ്റ്റ് എഡിറ്ററുകളായ Notepad, GEdit, Geany തുടങ്ങിയവ ഉപയോഗിക്കാം. ചുവടെ നൽകിയിരിക്കുന്ന പ്രകാരം കോഡ് എഴുതുക

```
<?php
// this line of code displays a message
echo 'My First PHP Program';
?>
```

ഘട്ടം 2: ഈ ഫയൽ വെബ് സെർവറിന്റെ റൂട്ട് ഡയറക്ടറിയിൽ (WAMP സെർവറാണെങ്കിൽ 'C:\WAMP\WWW' ഡയറക്ടറിയിലും LAMP സെർവറാണെങ്കിൽ

'\VAR\WWW' എന്ന ഡയറക്ടറി യിലും) അനുയോജ്യമായ പേരിൽ .php എക്സ്റ്റൻഷനോടുകൂടി സേവ് ചെയ്യുക (ഉദാ: example.php)



ചിത്രം 10.1: ലേതു പോലുള്ള ഔട്ട്പുട്ട് ലഭിക്കുവാൻ

ഘട്ടം 3: വെബ് സെർവറിന്റെ പ്രവർത്തനം ആരംഭിച്ചു എന്ന് ഉറപ്പുവരുത്തുക.

ഘട്ടം 4: അതിനുശേഷം വെബ് ബ്രൗസർ തുറന്ന് ഫയൽ സംഭരണ സ്ഥലത്തിനു അനുസൃതമായ URL (ഉദാ: http://localhost/example.php) അഡ്രസ്സ് ബാറിൽ നൽകുക.

PHP -യിൽ ഓരോ വരി കോഡും അവസാനിക്കുന്നത് അർദ്ധവിരാമ ചിഹ്നത്തോടുകൂടിയാണ് (;). PHP കോഡിലെ അവസാന പ്രസ്താവനക്ക് അർദ്ധവിരാമം നിർബന്ധമില്ല.

WAMP-ന്റെ തനതായ റൂട്ട് ഡയറക്ടറി "C:\wamp\www" ഉം, XAMPP-ന്റേത് "C:\xampp\htdocs" -ഉം LAMP ന്റേത് Ubuntu-വിൽ "/var/www"-ഉം ആണ്.

വെബ് സെർവറിൽ നടക്കുന്ന PHP സ്ക്രിപ്റ്റിന്റെ പ്രവർത്തനം താഴെ പറയും വിധം ചുരുക്കി എഴുതാം.

1. ബ്രൗസറിന്റെ അഡ്രസ്സ് ബാറിൽ പേജിന്റെ URL (ഉദാ: http://localhost/first.php) ടൈപ്പ് ചെയ്തു കൊണ്ട് നാം ഒരു വെബ്പേജിനായി ആവശ്യപ്പെടുന്നു.
2. ആവശ്യപ്പെട്ട URL ഒരു PHP സ്ക്രിപ്റ്റാണെന്ന് മനസ്സിലാക്കുന്ന വെബ് സെർവർ അത് പ്രവർത്തിപ്പിക്കുവാൻ php എൻജിന് നിർദേശം നൽകുന്നു.
3. സ്ക്രിപ്റ്റ് പ്രവർത്തിക്കുകയും അനന്തരം പ്രവർത്തന ഫലം അടങ്ങുന്ന ഒരു HTML പേജ് നമ്മുടെ വെബ് ബ്രൗസറിലേക്ക് അത് അയക്കുകയും ചെയ്യുന്നു.

നമുക്ക് ചെയ്യാം

'letusdo' എന്ന പേരിൽ ഒരു ഫോൾഡർ വെബ് സെർവറിന്റെ റൂട്ട് ഡയറക്ടറിയിൽ നിർമ്മിക്കുക.കുറച്ച് ഫയലുകൾ ഈ ഫോൾഡറിലേക്കു കോപ്പി ചെയ്യുക. ചുവടെ നൽകിയിരിക്കുന്നത് ചെയ്തതിനു ശേഷം ഔട്ട്പുട്ട് നിരീക്ഷിക്കുക.

a. ബ്രൗസറിന്റെ അഡ്രസ് ബാറിൽ URL (http://localhost/letsdo) ടൈപ്പ് ചെയ്യുക.

b. 'index.html' എന്ന പേരിൽ ഒരു HTML ഫയൽ നിർമ്മിച്ച് 'letusdo' ഫോൾഡറിൽ സേവ് ചെയ്യുക. അതിനുശേഷം മുൻപ് പറഞ്ഞ URL പരീക്ഷിക്കുക.

c. 'letusdo.php' എന്ന പേരിൽ ഒരു PHP ഫയൽ നിർമ്മിച്ച് 'letusdo' ഫോൾഡറിൽ സേവ് ചെയ്യുക. അതിനു ശേഷം http://localhost/letusdo/letusdo.php എന്ന URL പരീക്ഷിക്കുക.

C. HTML-ഉം PHP-യും സംയോജിപ്പിക്കൽ (Combining HTML and PHP)

PHP-യും HTML-ഉം പരസ്പര ബന്ധമുള്ളതും അവ ഒരുമിച്ച് പ്രവർത്തിക്കുന്നതുമാണ്. PHP -ൽ നിന്നും HTML ഉൽപാദിപ്പിക്കപ്പെടുകയും, അത് PHP യിലേക്ക് വിവരങ്ങൾ കൈമാറും ചെയ്യുന്നു. PHP കോഡ് HTML പ്രോഗ്രാമിൽ ഉൾക്കൊള്ളിക്കുന്നത് "<?php", "?>" എന്നീ രണ്ട് പ്രത്യേക ടാഗുകൾക്കുള്ളിലാണ്. PHP കോഡ് വഴക്കമുള്ളതും ഡോക്യുമെന്റിൽ എവിടെ വേണമെങ്കിലും ഉൾപ്പെടുത്താവുന്നതുമാണ്. PHP ടാഗുകൾക്ക് പുറത്തുള്ളതെല്ലാം HTML ആയി പരിഗണിക്കപ്പെടും.

Program 10.1: HTML ഡോക്യുമെന്റിലുള്ള PHP സ്ക്രിപ്റ്റ്

```
<HTML>
<HEAD>
<TITLE> My first php page</TITLE>
</HEAD>
<BODY>
<?php
 echo "<H2> Welcome to php </H2>";
?>
</BODY>
</HTML>
```

ഉപയോക്താവ് ഈ HTML ഡോക്യുമെന്റ് ആവശ്യപ്പെടുമ്പോൾ വെബ് സെർവർ PHP കോഡ് തിരിച്ചറിയുകയും അത് പ്രവത്തിപ്പിച്ച് ലഭിക്കുന്ന ഔട്ട്പുട്ട് HTML ഡോക്യുമെന്റിൽ ഉൾപ്പെടുത്തി ഉപയോക്താവിന് തിരികെ അയക്കുകയും ചെയ്യുന്നു.

D. PHP-യിൽ കമന്റുകൾ (Comments in PHP) രേഖപ്പെടുത്തുന്നു

PHP കോഡിൽ പ്രോഗ്രാമിന്റെ ഭാഗമായി പ്രവർത്തിക്കാത്ത ഒരു വരിയാണ് കമന്റുകൾ. PHP-യിൽ ഒറ്റവരി കമന്റ് (//), പല വരി കമന്റ് (/* */) എന്നിങ്ങനെ രണ്ടു തരം കമന്റുകൾ ഉണ്ട്. പ്രോഗ്രാമിനെക്കുറിച്ച് മറ്റുള്ളവർക്ക് നൽകുന്ന കുറിപ്പുകളാണ് കമന്റുകൾ. ഇവ ഭാവിയിൽ പ്രോഗ്രാം വായിക്കുന്നവർക്ക് വളരെ പ്രയോജനം ചെയ്യും.

Program 10.2: HTML ഡോക്യുമെന്റിലെ PHP സ്ക്രിപ്റ്റ്

```
<?php
   /* Program to display the message
   Welcome to php */
   echo "Welcome to php"; //Shows a message
?>
```

E. PHP-യിലെ ഔട്ട്പുട്ട് പ്രസ്താവനകൾ (Output Statements in PHP)

ഔട്ട്പുട്ട് ലഭിക്കുവാനായി PHP യിൽ വിവിധ തരം വാക്യരചനകളും ഫങ്ഷനുകളും ലഭ്യമാണ്. ഇവക്ക് ഇന്റിജർ, ഫ്ളോട്ട്, സ്ട്രിംഗ്, അറെ തുടങ്ങിയ ഡാറ്റഇനങ്ങൾ കൈകാര്യം ചെയ്യുവാൻസാധിക്കും.

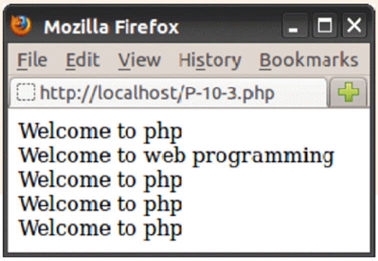
(i) echo-യും print-ഉം (echo and print)

PHP-യിൽ ഔട്ട്പുട്ട് വെബ്പേജിൽ പ്രദർശിപ്പിക്കുന്നതിന് ഉപയോഗിക്കുന്ന രണ്ട് പദങ്ങളാണ് echo, print എന്നിവ. എല്ലാത്തരം ഡാറ്റയും ഔട്ട്പുട്ട് ചെയ്യുവാൻ ഇവക്ക് സാധിക്കുമെങ്കിലും ഒന്നിലധികം ഔട്ട്പുട്ടുകൾ echo-ക്ക് മാത്രമേ കൈകാര്യം ചെയ്യാൻ കഴിയൂ.

echo, print എന്നീ പ്രസ്താവനകൾ ആവരണചിഹ്നത്തോട് കൂടിയോ അല്ലാതെയോ ഉപയോഗിക്കാം.

Program 10.3: echo, print എന്നിവ PHP സ്ക്രിപ്റ്റിൽ

```
<?php
   echo "Welcome to php<BR>", "Welcome to
   web programming<BR>";
   echo ("Welcome to php<BR>");
   print "Welcome to php<BR>";
   print ("Welcome to php<BR>");
?>
```



echo, print എന്നിവ തമ്മിലുള്ള വ്യത്യാസം താഴെ കാണുന്ന പട്ടികയിൽ വിവരിച്ചിരിക്കുന്നു.

echo	print
ആവരണ ചിഹ്നം ഇല്ലാതെ ഉപയോഗിക്കുമ്പോൾ ഒന്നിൽ കൂടുതൽ പരാമീറ്ററുകൾ നൽകാം.	ഒരു പരാമീറ്റർ മാത്രമേ നൽകാൻ കഴിയൂ.
വിലകൾ ഒന്നും തിരികെ തരുന്നില്ല.	ശരിയായ രീതിയിൽ പ്രവർത്തിച്ചാൽ TRUE ഉം സ്ക്രിങ് പ്രിന്റ് ചെയ്യാൻ സാധിക്കുന്നില്ലെങ്കിൽ FALSE എന്നും തിരികെ തരും.
print നേക്കാൾ അല്പം വേഗത കൂടുതൽ	echo യേക്കാൾ വേഗത അല്പം കുറവാണ്.

പട്ടിക 4.1: echo -യും print- ഉം തമ്മിലുള്ള താരതമ്യം

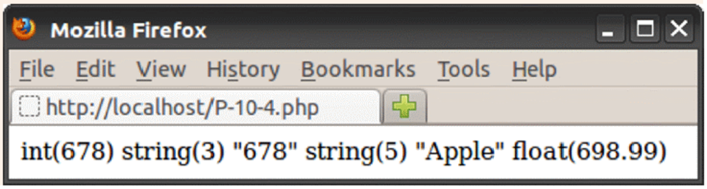
(ii) var_dump()

പ്രിന്റ് ചെയ്യുമ്പോൾ വേരിയബിളിന്റെ വിലക്കൊപ്പം ഡാറ്റഇനവും പ്രിന്റ് ചെയ്യുവാൻ var_dump( ) ഫങ്ഷൻ ഉപയോഗിക്കുന്നു. ഇന്റിജർ, ഫ്ളോട്ട്, സ്ട്രിംഗ്, അറെകൾ, ഒബ്ജക്ടുകൾ തുടങ്ങിയവ പ്രദർശിപ്പിക്കുവാൻ ഇത് ഉപയോഗിക്കാം. ഇതിന്റെ വാക്യഘടന താഴെ നൽകുന്നു.

```
var_dump(variable1, variable 2, ....variable n);
```

Program 10.4 : var_dump( ) വിശദീകരിക്കുന്നതിനുള്ള പ്രോഗ്രാം

```
<?php
    $a1=678;
    $a2="678";
    $a3="Apple";
    $a4=698.99;
    var_dump($a1);
    var_dump($a2);
    var_dump($a3);
    var_dump($a4);
?>
```



10.2 PHP-യുടെ അടിസ്ഥാന ഘടകങ്ങൾ (Fundamentals of PHP)

PHP-യിലെ വേരിയബിളുകൾ, ഡാറ്റ ഇനങ്ങൾ, ഓപ്പറേറ്ററുകൾ എന്നിവ ഈ ഭാഗത്ത് നാം ചർച്ച ചെയ്യും.

10.2.1 വേരിയബിളുകൾ (Variables)

PHP-യിൽ ഒരു വേരിയബിളിന്റെ പേര് '$' ചിഹ്നത്തോടു കൂടി ആരംഭിക്കുകയും തുടർന്ന് വേരിയബിളിനു നൽകാനുള്ള പേര് നൽകുകയും ചെയ്യുന്നു. മറ്റ് പ്രോഗ്രാമിങ് ഭാഷയിലേതുപോലെ വേരിയബിളുകൾ ഉപയോഗിക്കുന്നതിനു മുൻപ് PHP-യിൽ വേരിയബിളിനെ പ്രസ്താവിക്കേണ്ടതില്ല. ഒരു വേരിയബിളിലേക്ക് ആദ്യ വില നൽകുമ്പോൾ ആ വേരിയബിൾ നിർമ്മിക്കുന്നു. നൽകപ്പെടുന്ന വിലയ്ക്ക് അനുസരിച്ച് PHP സ്വയമേ വേരിയബിളിന്റെ ഡാറ്റ ഇനം നിശ്ചയിക്കുന്നു.

```
$Var_name = value;
```

```
<?php
    <?php
    $txt = "Hello world";
    $x = 25;
    $y = 23.5;
?>
```

മുകളിൽ കൊടുത്ത പ്രസ്താവനകളുടെ പ്രവർത്തനത്തിന് ശേഷം $txt എന്ന വേരിയബിളിൽ 'Hello world' എന്ന സ്ട്രിംഗ് വിലയും, $x എന്ന വേരിയബിളിൽ 25 എന്ന പൂർണ സംഖ്യാവിലയും ഇന്റിജർ വിലയും $y എന്ന വേരിയബിളിൽ 23.5 എന്ന ദശാംശ വിലയും ലഭിക്കുന്നു. വിലകൾക്ക് അനുസൃതമായി ഡാറ്റ ഇനങ്ങളും നിർണ്ണയിക്കപ്പെടുന്നു. വേരിയബിളിന് സ്ട്രിംഗ് വിലകൾ നൽകുമ്പോൾ ഉദ്ധരണി ചിഹ്നം വിലയുടെ ഇരുവശത്തും നൽകണം.

PHP-യിൽ വേരിയബിളുകളുകൾക്ക് പേര് നൽകുന്ന രീതി

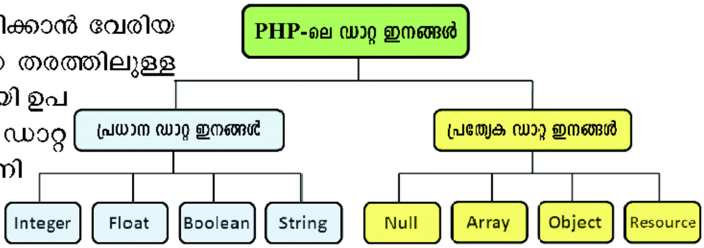
ഒരു വേരിയബിളിന് ചെറിയ പേരോ (m,n തുടങ്ങിയവ) വിവരണാത്മക പേരോ (Age, Reg_No, Amount തുടങ്ങിയവ) നൽകാവുന്നതാണ്.

PHP വേരിയബിളുകൾക്ക് പേര് നൽകുന്നതിനുള്ള നിയമങ്ങൾ താഴെ ചേർക്കുന്നു (Variable naming conventions in PHP)

- ഒരു വേരിയബിളിന്റെ പേര് $ ചിഹ്നത്തോടു കൂടി ആരംഭിക്കണം.
- വേരിയബിളിന്റെ പേര് തുടങ്ങുന്നത് അക്ഷരമോ അല്ലെങ്കിൽ അണ്ടർസ്കോറോ (_) ഉപയോഗിച്ചായിരിക്കണം.
- അക്ഷരങ്ങൾ, സംഖ്യകൾ, അണ്ടർസ്കോർ എന്നിവ മാത്രമേ പേരുകളിൽ അനുവദനീയമായിട്ടുള്ളൂ (a-z, A-Z, 0-9, and _).
- ചെറിയ അക്ഷരത്തിൽ എഴുതുന്ന പേരും വലിയ അക്ഷരത്തിൽ എഴുതുന്ന പേരും ഒരേ അക്ഷരമാണെങ്കിൽ കൂടി വ്യത്യസ്ത വേരിയബിളുകളായി പരിഗണിക്കും ($name, $NAME എന്നിവ രണ്ടു വ്യത്യസ്ത വേരിയബിളുകളാണ്).
- കോമ, ഉദ്ധരണി ചിഹ്നം, വിസർഗം, ചോദ്യ ചിഹ്നം, മറ്റ് വിശേഷ ചിഹ്നങ്ങൾ, ശൂന്യ സ്ഥലം തുടങ്ങിയവ വേരിയബിളിന്റെ പേരിൽ ഉൾപ്പെടാൻ പാടില്ല.

10.2.2 ഡാറ്റ ഇനങ്ങൾ (Data Types)

വിവിധ തരം ഡാറ്റകൾ സംഭരിക്കാൻ വേരിയബിളുകൾക്ക് കഴിയും. വിവിധ തരത്തിലുള്ള ഡാറ്റ ഇനങ്ങൾ വ്യത്യസ്തമായി ഉപയോഗിക്കുന്നു. PHP-യിലെ ഡാറ്റ ഇനങ്ങളെ ചിത്രം 10.2-ൽ കാണിച്ചിരിക്കുന്നവിധം തരംതിരിച്ചിരിക്കുന്നു.



ചിത്രം 10.2: PHP - യിലെ ഡാറ്റ ഇനങ്ങൾ

A. പ്രധാന ഡാറ്റ ഇനങ്ങൾ (Core data types)

ഇന്റിജർ, ഫ്ളോട്ട്, ഡബിൾ തുടങ്ങിയ പ്രധാന ഡാറ്റ ഇനങ്ങൾ C++-ൽ നാം പഠിച്ച ഡാറ്റ ഇനങ്ങൾക്ക് സമാനമാണ്. ഇന്റിജറിന്റേയും ഫ്ളോട്ടിന്റെയും വലിപ്പം കമ്പ്യൂട്ടറിന്റെ അടിസ്ഥാന സോഫ്ട്‌വെയറിനെ ആശ്രയിച്ചിരിക്കും.

ഇന്റിജർ (Integer)

ഇന്റിജറുകൾ പൂർണ്ണസംഖ്യകളാണ്. ഇതിൽ ദശാംശ സംഖ്യകൾ ഉൾപ്പെടില്ല. നെഗറ്റീവ് സംഖ്യകളും ഇന്റിജർ വിഭാഗത്തിൽ ഉൾപ്പെടും. ചിഹ്നമില്ലാത്ത ഇന്റിജർ ഡാറ്റ PHP ഇനം പിന്തുണക്കുന്നില്ല.

ഉദാഹരണം: 32, -32, 986 തുടങ്ങിയവ

ഫ്ളോട്ട്/ഡബിൾ (Float/Double)

ദശാംശസ്ഥാനമുള്ള സംഖ്യകളെ ഫ്ളോട്ട് അല്ലെങ്കിൽ ഡബിൾ എന്ന ഡാറ്റ ഇനത്തിൽ ഉൾപ്പെടുത്തിയിരിക്കുന്നു. ഫ്ളോട്ടിനേക്കാൾ ഡബിളിന് ദശാംശസ്ഥാനങ്ങളും കൃത്യതയും കൂടുതലാണ്.

ഉദാഹരണം: 123.56, 5.6, 1.2e3, 7E-10 etc.

സ്ട്രിംഗ് (String)

അക്ഷരങ്ങൾ/സംഖ്യകൾ/ചിഹ്നങ്ങൾ/ ഇവയുടെ കൂട്ടം, വാക്കുകൾ, വാചകങ്ങൾ തുടങ്ങിയവക്കാണ് ഈ ഡാറ്റ ഇനം ഉപയോഗിക്കുന്നത്. ഇത്തരത്തിലുള്ള ഡാറ്റ, ഉദ്ധരണി ചിഹ്നങ്ങൾക്ക് ഉള്ളിലാണ് നൽകേണ്ടത്. (" "), (' ') എന്നീ രണ്ടു തരം ഉദ്ധരണികളും ഉപയോഗിക്കാവുന്നതാണ്. ഉദ്ധരണി ചിഹ്നം ഉപയോഗിക്കുമ്പോൾ സ്ട്രിംഗിന്റെ തുടക്കവും ഒടുക്കവും ഒരേതരം ഉദ്ധരണികൾ ആയിരിക്കണം. അതായത് സ്ട്രിംഗ് ആരംഭിക്കുന്നത് ഏക ഉദ്ധരണി ചിഹ്നത്തിൽ ആണെങ്കിൽ അതേ ചിഹ്നത്തിൽ തന്നെ സ്ട്രിംഗ് അവസാനിപ്പിക്കണം.

ഉദാഹരണം: "Apple", 'PHP' etc.

ബൂളിയൻ (Boolean)

ബൂളിയൻ ഡാറ്റ ഇനം ശരി (TRUE) അല്ലെങ്കിൽ തെറ്റ് (FALSE) എന്നീ രണ്ട് അവസ്ഥകളെ പ്രതിനിധീകരിക്കുന്നു. പ്രോഗ്രാമിലെ ചില വാക്യങ്ങൾ ചില നിബന്ധനകൾക്ക് അനുസൃതമാണോ അല്ലയോ എന്ന് പരിശോധിക്കുമ്പോൾ ഇത്തരം ബൂളിയൻ വേരിയബിളുകൾ ഉപയോഗിക്കുന്നു. മുമ്പ് പറഞ്ഞതുപോലെ PHP ചെറിയക്ഷരങ്ങളെയും വലിയക്ഷരങ്ങളെയും വ്യത്യസ്തമായി പരിഗണിക്കുന്നതിനാൽ `yes/no`, `on/off`, `1/0` എന്നിങ്ങനെ മുൻനിർവചിത ബൂളിയൻ വിലകൾ മാത്രമേ നമുക്കുപയോഗിക്കാൻ സാധിക്കൂ.

```
< ?php
     $a = 1 ; // Assign the value TRUE TO $a
     $b = 0 ; // Assign the value FALSE to $b
? >
```

B. പ്രത്യേക ഡാറ്റ ഇനങ്ങൾ (Special datat ypes)

നൾ (NULL)

ഇത് ഒരു പ്രത്യേക തരം ഡാറ്റ ഇനമാണ്. ഇതിന്റെ ഏക വില NULL എന്നത് മാത്രമാണ്. ഒരു വേരിയബിൾ നിർമ്മിക്കുമ്പോൾ മറ്റു വിലകൾ ഒന്നും നൽകുന്നില്ലെങ്കിൽ PHP അതിനെ NULL ആയി പരിഗണിക്കുന്നു. വേരിയബിളുകൾ ശൂന്യമാക്കുന്നതിനും NULL ഡാറ്റ ഇനം ഉപയോഗിക്കുന്നു.

ഉദാഹരണം: `$x = null; //$x is assigned with NULL value`

അറെ (Array)

ഒരേ ഇനത്തിൽപെടുന്ന ഡാറ്റയുടെ കൂട്ടമാണ് അറെ. PHP-യിൽ ഓരോ അറെ അംഗത്തിനും ഒരു കീയും വിലയും ഉണ്ടായിരിക്കും. മൂന്നു തരം അറെകൾ PHP-യിൽ ലഭ്യമാണ്. അവയുടെ പേരുകൾ ഇൻഡക്സ്ഡ് അറെ, അസ്സോസിയേറ്റീവ് അറെ, മൾട്ടീഡയമെൻഷണൽ അറെ എന്നിങ്ങനെയാണ്. അറെകളെകുറിച്ച് കൂടുതൽ വിശദീകരണം ഭാഗം 10.4-ൽ നൽകിയിരിക്കുന്നു.

ഒബ്ജക്റ്റ് (Object)

C++ലെ ഒബ്ജക്റ്റുകൾക്കു സമാനമാണ് PHP യിലെ ഒബ്ജകറ്റ് ഡാറ്റ ഇനം. വേരിയബിളുകളും അത് പ്രോസസ്സ് ചെയ്യുന്ന ഫങ്ഷനുകളും ഒബ്ജെക്റ്റിൽ അടങ്ങിയിരിക്കുന്നു.

PHP-യിൽ ഒബ്ജക്റ്റുകൾ പ്രത്യേകമായിത്തന്നെ പ്രസ്താവിക്കണം. ആദ്യം ക്ലാസ് പ്രസ്താവിച്ചതിനുശേഷമാണ് അതിന്റെ ഇൻസ്റ്റൻസുകൾ അഥവാ ഒബ്ജക്റ്റുകൾ നിർമ്മിക്കുക. ഇതിനായി 'new' എന്ന കമാൻഡ് ഉപയോഗിക്കുന്നു.

റിസോഴ്സ് (Resource)

ഇവ യഥാർത്ഥ ഡാറ്റ ഇനങ്ങൾ അല്ല. ഇവ ചില പ്രത്യേക വേരിയബിളുകളാണ്. ഇവയ്ക്ക് ചില റഫറൻസുകൾ സൂക്ഷിക്കാൻ സാധിക്കും. ഫയൽ ഹാൻഡ്ലെർ, ഡാറ്റാബേസ് ഒബ്ജക്റ്റ് തുടങ്ങിയവ ഇതിൽ ഉൾപ്പെടും. റിസോഴ്സുകൾ നിർമിക്കുകയും ഉപയോഗിക്കുകയും ചെയ്യുന്നതിന് പ്രത്യേക ഫംഗ്ഷനുകൾ ഉപയോഗിക്കുന്നു.

10.2.3 PHP-യിലെ ഓപ്പറേറ്ററുകൾ (Operators)

PHP-യിൽ ഉപയോഗിക്കുന്ന ഓപ്പറേറ്ററുകൾ താഴെ കൊടുത്തിരിക്കുന്നവിധം തരംതിരിക്കാം:

a. വില നൽകൽ ഓപ്പറേറ്റർ

b. അരിത്മെറ്റിക് ഓപ്പറേറ്ററുകൾ

c. റിലേഷണൽ ഓപ്പറേറ്ററുകൾ

d. ലോജിക്കൽ ഓപ്പറേറ്ററുകൾ

e. സ്ട്രിംഗ് ഓപ്പറേറ്ററുകൾ

f. കൂട്ടിച്ചേർത്ത ഓപ്പറേറ്ററുകൾ

a. വില നൽകൽ ഓപ്പറേറ്റർ (Assignment Operator)

ഒരു വേരിയബിളിലേക്ക് നേരിട്ടോ അല്ലെങ്കിൽ മറ്റൊരു വേരിയബളിൽ നിന്നോ വില നൽകുന്നതിന് ഈ ഓപ്പറേറ്റർ ഉപയോഗിക്കുന്നു. '=' ചിഹ്നമാണ് വില നൽകൽ ഓപ്പറേറ്ററായി ഉപയോഗിക്കുന്നത്.

ഉദാ:

```
$my_var = 4; //$my_var is assigned a value 4
$new_var = $my_var; //$new_var is assigned with the value of $my_var
```

b. അരിത്മെറ്റിക് ഓപ്പറേറ്ററുകൾ (Arithmetic Operators)

ഓപ്പറേറ്റർ	അർഥമാക്കുന്നത്	ഉദാഹരണം	ഉത്തരം
+	സങ്കലനം	3 + 2	5
-	വ്യവകലനം	5 - 3	2
*	ഗുണനം	6 * 5	30
/	ഹരണം	9 / 3	3
%	ശിഷ്ടം കാണുക	10 % 3	1

c. റിലേഷണൽ ഓപ്പറേറ്ററുകൾ (Relational Operators)

ഓപ്പറേറ്റർ	അർഥമാക്കുന്നത്	ഉദാഹരണം (x = 3, y = 5)	ഉത്തരം
==	തുല്യമാണ്	`$x == $y`	തെറ്റ്
!=	തുല്യമല്ല	`$x != $y`	ശരി
<	ചെറുതാണ്	`$x < $y`	ശരി
>	വലുതാണ്	`$x > $y`	തെറ്റ്
<=	ചെറുതതോ തുല്യമോ	`$x <= $y`	ശരി
>=	വലുതോ തുല്യമോ	`$x >= $y`	തെറ്റ്

d. ലോജിക്കൽ ഓപ്പറേറ്ററുകൾ (Logical Operators)

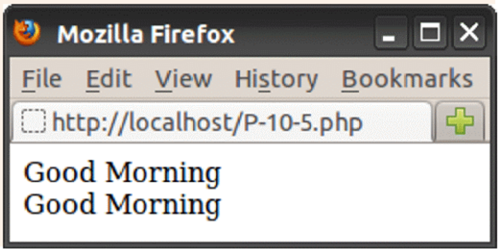
ഓപ്പറേറ്റർ	അർഥമാക്കുന്നത്	ഉദാഹരണം (a = 7, b = 2, c = 9)	ഉത്തരം
and അല്ലെങ്കിൽ &&	എല്ലാ റിലേഷണൽ പദപ്രയോഗങ്ങളും ശരിയാണെങ്കിൽ True തിരികെ തരുന്നു	(a>b) and (a>c)	തെറ്റ്
or അല്ലെങ്കിൽ ||	ഏതെങ്കിലും ഒരു റിലേഷണൽ പദപ്രയോഗ മെങ്കിലും ശരിയായാൽ True തിരികെ തരുന്നു	(a>b) or (a>c)	ശരി
xor	രണ്ട് റിലേഷണൽ പദപ്രയോഗങ്ങളിൽ ഏതെങ്കിലും ഒന്ന് മാത്രം ശരിയായാൽ True തിരികെ തരുന്നു	(a>b) xor (a>c)	ശരി
!	റിലേഷണൽ വാക്യം തെറ്റായാൽ True തിരികെ തരുന്നു	!(a>b)	തെറ്റ്

e. സ്ട്രിങ് ഓപ്പറേറ്ററുകൾ (String Operators)

രണ്ടു തരം സ്ട്രിങ് ഓപ്പറേറ്ററുകൾ ലഭ്യമാണ്. അവ കോൺകാറ്റിനേറ്റിങ് ഓപ്പറേറ്റർ ('.'), കോൺകാറ്റിനേറ്റിംഗ് അസൈൻമെന്റ് ഓപ്പറേറ്റർ ('.=') എന്നിവയാണ്. കോൺകാറ്റിനേറ്റിംഗ് ഓപ്പറേറ്റർ രണ്ടു സ്ട്രിങ്ങുകളെ ഒരുമിച്ച് ചേർത്ത് പുതിയ ഒരു സ്ട്രിങ് ആക്കി മാറ്റുന്നു. എന്നാൽ കോൺകാറ്റിനേറ്റിംഗ് അസൈൻമെന്റ് ഓപ്പറേറ്റർ '=' ചിഹ്നത്തിന് വലത് വശത്തുള്ള സ്ട്രിങ്ങിനെ ഇടത് വശത്തുള്ള സ്ട്രിങ്ങിനോട് ചേർക്കുന്നു.

Program 10.5: കോൺകാറ്റിനേഷൻ ഓപ്പറേഷൻ വിശദീകരിക്കുന്ന പ്രോഗ്രാം

```
<HTML>
<BODY>
  <?php
    $string1= 'Good';
    $string2= ' Morning';
    $newstring = $string1.$string2; //concatenation
    $string1 .= $string2; //concatenation assignment
    echo $newstring."<BR>";
    echo $string1;
  ?>
</BODY>
</HTML>
```



f. കൂട്ടിച്ചേർത്ത ഓപ്പറേറ്ററുകൾ (Combined Operators)

C++-ലേത് പോലെ രണ്ടു വ്യത്യസ്തതരം ഓപ്പറേറ്ററുകൾ കൂടിച്ചേർന്നാണ് ഇത്തരം ഓപ്പറേറ്ററുകൾ രൂപപ്പെടുന്നത്.

ഓപ്പറേറ്റർ	അർഥമാക്കുന്നത്	ഉദാഹരണം	സമാനപ്രവർത്തനം
+=	സങ്കലനത്തിന് ശേഷം വില നൽകുന്നു	`$x += 2;`	`$x = $x + 2;`
-=	വ്യവകലനത്തിന്ശേഷം വില നൽകുന്നു	`$x -= 4;`	`$x = $x - 4;`
*=	ഗുണനത്തിന്ശേഷം വില നൽകുന്നു	`$x *= 3;`	`$x = $x * 3;`
/=	ഹരണത്തിന്ശേഷം വില നൽകുന്നു	`$x /= 2;`	`$x = $x / 2;`
%=	ശിഷ്ടം കണ്ടുപിടിച്ചതിന്ശേഷം വില നൽകുന്നു	`$x %= 5;`	`$x = $x % 5;`
.=	കൂട്ടിച്ചേർത്തതിന്ശേഷം വില നൽകുന്നു	`$string1.="hello";`	`$string1=$string1."hello";`

g. വർദ്ധനവ്/ കുറവ് ഓപ്പറേറ്ററുകൾ

ഓപ്പറേറ്റർ	അർഥമാക്കുന്നത്	ഉദാഹരണം (if x = 2)	ഉത്തരം
++വേരിയബിൾ	മുൻപേയുള്ള വർദ്ധനവ്	`echo ++$x;`	3
--വേരിയബിൾ	മുൻപേയുള്ള കുറവ്	`echo --$x;`	1
വേരിയബിൾ++	ശേഷമുള്ള വർദ്ധനവ്	`echo $x++;`	2
വേരിയബിൾ--	ശേഷമുള്ള കുറവ്	`echo $x--;`	2

h. എസ്കേപ്പ് സീക്വൻസ്

ഇംഗ്ലീഷ് അക്ഷരമാലയിലെ ഒരു അക്ഷരവും അതിനു മുന്നിൽ ബാക്ക്സ്ലാഷ് (\) ചിഹ്നവും ചേരുമ്പോൾ എസ്കേപ്പ് ക്യാരക്ടർ ഉണ്ടാകുന്നു. HTML-ലെ <PRE> ടാഗ് ഉപയോഗിച്ചാണ് ഇവ വെബ് പേജിൽ പ്രദർശിപ്പിക്കുന്നത്.

ഓപ്പറേറ്റർ	അർത്ഥമാക്കുന്നത്	ഉദാഹരണം	ഉത്തരം
`\"`	" പ്രിന്റ് ചെയ്യുന്നു	`echo "\"Hello\"";`	"Hello"
`\'`	' പ്രിന്റ് ചെയ്യുന്നു	`echo "Raju\'s";`	Raju's
`\n`	പുതിയ വരിയിലേക്ക് പോകുന്നു	`echo "Raju\nHari";`	Raju Hari
`\t`	ഒരു ടാബ് സ്ഥലം നിർമിക്കുന്നു	`echo "Raju\tHari";`	Raju Hari
`\r`	Enter കീ ഉപയോഗിക്കുന്നതിന് തുല്യം		
`\$`	$ പ്രിന്റ് ചെയ്യുന്നു	`echo "50\$";`	50$
`\\`	\ പ്രിന്റ് ചെയ്യുന്നു	`echo "C:\\Windows";`	C:\Windows;

നിങ്ങളുടെ പുരോഗതി അറിയുക (Know your progress)

1. PHP-യുടെ പൂർണ്ണരൂപം ______ ആകുന്നു.
2. PHP-ഫയലുകളുടെ തനതായ ഫയൽ എക്സ്റ്റൻഷൻ ______ ആകുന്നു.
3. PHP-യുടെ പിതാവ് എന്നറിയപ്പെടുന്നത് ആരാണ്?
4. ഒറ്റവരി വിവരണത്തിന് PHP-യിൽ ______________ ഉപയോഗിക്കുന്നു.
5. WAMP-ന്റെ പൂർണ്ണ രൂപം ______________ ആകുന്നു.
6. LAMP-ന്റെ പൂർണ്ണ രൂപം ______________ ആകുന്നു.
7. ഒരു PHP സ്ക്രിപ്റ്റ് തുടങ്ങുന്നത് ______ ലും അവസാനിക്കുന്നത് _________ ലും ആകുന്നു.
8. PHP-യിൽ വേരിയബിളുകൾ എപ്പോഴും ആരംഭിക്കുന്നത്_________ ഉപയോഗിച്ചാണ്.
9. PHP-യുടെ വാക്യഘടന _____ പ്രോഗ്രാമിങ് ഭാഷയുടേതിന് സമാനമാണ്.
10. var_dump( ) ന്റെ ഉപയോഗം എന്ത്?
11. (.) ഓപ്പറേറ്റർ ____________ ന് ഉപയോഗിക്കുന്നു.

10.3 PHP-യിലെ നിയന്ത്രണ പ്രസ്താവനകൾ (Cotnrol Structures in PHP)

നിയന്ത്രണ പ്രസ്താവന PHP-യിലെ പ്രോഗ്രാമിന്റെ ഗതി നിയന്ത്രിക്കുന്നു. if.....else, switch തുടങ്ങിയ കണ്ടിഷണൽ പ്രസ്താവനകൾ ചില വ്യവസ്ഥകൾക്ക് അനുസരിച്ച് പ്രോഗ്രാമിലെ ഒരു ഭാഗം പ്രവർത്തിപ്പിക്കുന്നതിനോ അല്ലെങ്കിൽ പ്രവർത്തിപ്പിക്കാതിരിക്കുന്നതിനോ ഉപയോഗിക്കുന്നു. for, while, do…while എന്നീ ലൂപ്പുകൾ ഒരു പ്രത്യേക കോഡ് ആവർത്തിച്ച് പ്രവർത്തിപ്പിക്കുന്നതിന് ഉപയോഗിക്കുന്നു.

10.3.1 PHP-യിലെ കണ്ടിഷണൽ പ്രസ്താവനകൾ (Conditional Statements in PHP)

a. if പ്രസ്താവന

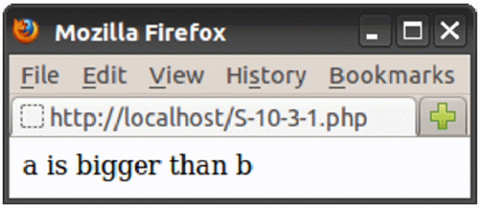
ഒരു പദപ്രയോഗത്തിന്റെ ഫലം ശരിയാണോ എന്ന് പരിശോധിച്ചതിനുശേഷം ശരിയെങ്കിൽ ഒന്നോ അതിലധികമോ പ്രസ്താവനകൾ പ്രവർത്തിപ്പിക്കുന്നതിന് ഇത് ഉപയോഗിക്കുന്നു.

വാക്യഘടന

```
if (expression)
{
  //code to execute if the
  //expression evaluates to true
}
```

ഉദാഹരണം 'a' വലുതാണോ എന്ന് പരിശോധിക്കുക

```
<?php
 $a = 10;
 $b = 5;
 if ($a > $b)
  echo "a is bigger than b";
?>
```



b. if.. else പ്രസ്താവന

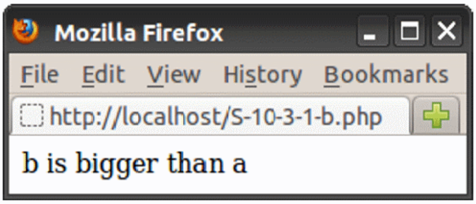
if ഉപയോഗിച്ച് പരിശോധിക്കുമ്പോൾ പദപ്രയോഗത്തിന്റെ ഫലം ശരിയാണെങ്കിൽ if-നോട് അനുബന്ധിച്ചുള്ള പ്രസ്താവനകൾ പ്രവർത്തിപ്പിക്കുകയും തെറ്റാണെങ്കിൽ else-നോട് അനുബന്ധിച്ചുള്ള പ്രസ്താവനകൾ പ്രവർത്തിപ്പിക്കുകയും ചെയ്യുന്നു.

വാക്യഘടന

```
if (expression)
{
//code to execute if the
//expression evaluates to true
}
else
{
//code to execute if the
//expression evaluates to false
}
```

ഉദാഹരണം
രണ്ടു സംഖ്യകളിൽ വലുത് കണ്ടുപിടിക്കുക

```
<?php
$a = 10;
$b = 65;
if ($a > $b)
   echo "a is bigger than b";
else
   echo "b is bigger than a";
?>
```



c. else if പ്രസ്താവന

വാക്യഘടന

```
if (expression)
{
}
else if (expression)
{
}
else if (expression)
{
}
else
{
//code to execute in all other cases.
}
```

ഉദാഹരണം

രണ്ടു സംഖ്യകൾ തുല്യമോ/വലുതോ എന്ന് കണ്ടുപിടിക്കുക.

```
<?php
$a = 55;
$b = 55;
if ($a > $b)
  echo "a is bigger than b";
else if ($b > $a)
  echo "b is bigger than a";
else
  echo "a and b are equal";
?>
```

d. switch പ്രസ്താവന

ഒരേ പദപ്രയോഗത്തിനുമേൽ നിരനിരയായി ഉപയോഗിക്കുന്ന if() പ്രസ്താവനകൾക്ക് സമാനമാണ് switch. ഇതിന്റെ വാക്യഘടനയും പ്രവർത്തനവും C++-ലെ Switch പ്രസ്താവനക്ക് സമാനമാണ്. എന്നാൽ PHP-യിലെ switch-ന് കേസ് ഐഡന്റിഫയറുകളായി സ്ട്രിംഗുകൾ ഉപയോഗിക്കാൻ സാധിക്കും.

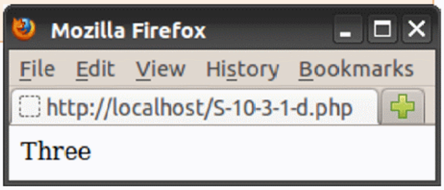
വാക്യഘടന

```
switch (expression)
{
 case result1:
   //execute this if result of
   //expression is result1
   break;
 case result2:
   //execute this if result of
   //expression is result2
   break;
 default:
   //execute this if result of
   //expression does not match
   //any case results
}
```

ഉദാഹരണം

1 മുതൽ 3 വരെയുള്ള സംഖ്യകൾ അക്ഷരത്തിൽ പ്രിന്റ് ചെയ്യുന്ന പ്രോഗ്രാം എഴുതുക

```
<?php
$n = 3;
switch ($n)
  { case 1:
       echo "One";
       break;
    case 2:
       echo "Two";
       break;
    case 3:
       echo "Three";
       break;
    default:
      echo "Enter correct no."
}?>
```



10.3.2 PHP- യിലെ ലൂപ്പുകൾ (Loops in PHP)

a. while ലൂപ്പ്

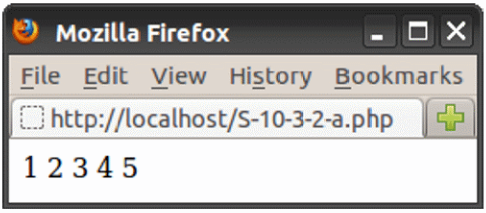
'while' നിയന്ത്രണഘടന ലൂപ്പിനുള്ളിലുള്ള പ്രസ്താവനകളെ പദപ്രയോഗത്തിന്റെ ഫലം False ആകുന്നതു വരെ ആവർത്തിച്ചു പ്രവർത്തിപ്പിക്കുന്നു.

വാക്യഘടന

```
while (expression)
{
 //code to execute if the
 //expression evaluates to true
}
```

ഉദാഹരണം
1 മുതൽ 5 വരെയുള്ള സംഖ്യകൾ പ്രിന്റ് ചെയ്യുന്ന പ്രോഗ്രാം

```
<?php
$i = 1;
while ($i <= 5)
 { echo "$i ";
   $i++; }
?>
```



b. do.....while ലൂപ്പ്

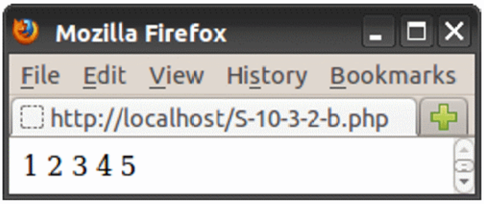
'do-while' പ്രസ്താവന ഒരു തവണയെങ്കിലും കോഡ് ശകലം പ്രവർത്തിപ്പിക്കുകയും ശേഷം വ്യവസ്ഥ True ആയിരിക്കുന്നുതു വരെ ആവർത്തിക്കുകയും ചെയ്യുന്നു.

വാക്യഘടന

```
do
{
 //code to execute if the
 //expression evaluates to true
} while (expression);
```

ഉദാഹരണം
1 മുതൽ 5 വരെയുള്ള സംഖ്യകൾ പ്രിന്റ് ചെയ്യുന്ന പ്രോഗ്രാം

```
<?php
$i = 1;
do
 { echo "$i ";
   $i++; }
while ($i <= 5);
?>
```

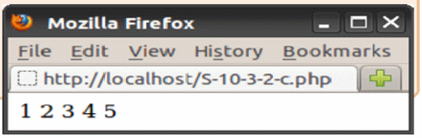


c. for ലൂപ്പ്

ഒന്നോ അതിലധികമോ പ്രസ്താവനകളെ ഒരു നിബന്ധനക്ക് വിധേയമായി എത്ര തവണ പ്രവർത്തിപ്പിക്കണമെന്ന് അറിയാമെങ്കിൽ `for` ലൂപ്പ് ഉപയോഗിക്കാം. നിശ്ചിത തവണ പ്രവർത്തിച്ച് കഴിയുമ്പോൾ ലൂപ്പ് പ്രവർത്തനം അവസാനിക്കുന്നു.

വാക്യഘടന	ഉദാഹരണം 1 മുതൽ 5 വരെയുള്ള സംഖ്യകൾ പ്രിന്റ് ചെയ്യുക

<pre>for(initialization;
      condition; increment)
{
  //code to be executed;
}</pre>
	<pre><?php
	  for ($i = 1;$i <= 5; $i++)
	    { echo "$i "; }
	?></pre>



d. ലൂപ്പിലെ continue, break എന്നീ പ്രസ്താവനകൾ

ലൂപ്പിനുള്ളിലെ പ്രസ്താവനകൾക്കിടയിൽ 'continue' വരികയാണെങ്കിൽ ലൂപ്പിലെ തുടർന്നുള്ള പ്രസ്താവനകൾ ഒഴിവാക്കുകയും ലൂപ്പിന്റെ ആരംഭത്തിലെ നിബന്ധന വില യിരുത്തിയതിനുശേഷം അടുത്ത ആവർത്തന പ്രക്രിയ നടത്തുകയും ചെയ്യുന്നു.

എന്നാൽ 'break' പ്രസ്താവന for, foreach, while, do...while, switch എന്നിവയുടെ പ്രവർത്തനം അവസാനിപ്പിക്കുന്നു. break നോടൊപ്പം വേണമെങ്കിൽ ഒരു സംഖ്യ ചേർക്കാ വുന്നതാണ്. ഒന്നിനുള്ളിൽ മറ്റൊന്ന് എന്ന രീതിയിൽ ഉപയോഗിച്ചിരിക്കുന്ന ലൂപ്പ് നിർമ്മി തികളിൽ എത്ര ലൂപ്പുകൾക്കുള്ളിൽ നിന്നും പുറത്തേക്കു വരണം എന്നതിനെ ഇത് സൂചി പ്പിക്കുന്നു. (ഉദാ: break 1;, break 2; മുതലായവ)

10.4 PHP യിലെ അറെകൾ (Arrays in PHP)

ഒരേ തരം ഡാറ്റകളുടെ കൂട്ടമാണ് അറെ.

ഓരോ അറെ അംഗത്തിനും (അറെ വേരിയബിൾ) ഒരു കീയും വിലയും ഉണ്ടായിരിക്കും. കീ ഉപയോഗിച്ചാണ് അറെ എലിമെന്റിലെ വില കണ്ടെത്തുന്നത്. C++-ൽ ഈ കീ ഇൻഡക്സ് എന്നാണ് അറിയപ്പെടുന്നത്.

ഉദാഹരണമായി a[1]=10 എന്ന് നാം എഴുതിയാൽ ഇൻഡക്സ് 1 എന്നത് കീയും 10 വില യുമാണ്. അറെയിൽ ഓരോ വേരിയബിളിനും കീയും വിലയും നൽകി അവയെ അറെ അംഗങ്ങൾ എന്നു വിളിക്കുന്നു. C++-ൽ കീ അഥവാ ഇൻഡക്സ് ആയി പോസിറ്റീവ് പൂർണ്ണ സംഖ്യകൾ മാത്രമാണ് നാം ഉപയോഗിക്കുന്നത്. എന്നാൽ PHP-യിൽ പോസിറ്റീവ് പൂർണ്ണ സംഖ്യകളോ സ്ട്രിംഗുകളോ കീകളായി നമുക്ക് ഉപയോഗിക്കാം. മൂന്നു തരം അറെകളാണ് PHP-യിൽ ഉള്ളത്.

- ഇൻഡെക്സ്ഡ് അറെ
- അസ്സോസിയേറ്റീവ് അറെ
- മൾട്ടി ഡയമെൻഷണൽ അറെ

ഇവയിൽ ഇൻഡെക്സ്ഡ് അറെയും, അസ്സോസിയേറ്റീവ് അറെയുമാണ് നാം ഇവിടെ ചർച്ച ചെയ്യുന്നത്. മൾട്ടി ഡയമെൻഷൻ അറെകളെക്കുറിച്ചുള്ള പഠനം ഈ പുസ്തകത്തിന്റെ പരിധിക്ക് പുറത്താണ്

10.4.1 ഇൻഡെക്സ്ഡ് അറെകൾ (Indexed Arrays)

ഇവ C++-ലെ അറെകൾക്ക് സമാനമാണ്. സംഖ്യകൾ കീ/ഇൻഡക്സ് ആയി ഉപയോഗി ക്കുന്ന അറെകളെയാണ് ഇൻഡക്സ്ഡ് അറെ എന്നു വിളിക്കുന്നത്. ഇത്തരം അറെകൾക്ക് സംഖ്യകൾ, അക്ഷരങ്ങൾ, മറ്റ് ഒബ്ജക്റ്റുകൾ എന്നിവയെല്ലാം സംഭരിക്കാൻ സാധിക്കുമെ

ങ്കിലും അവയുടെ ഇൻഡക്സ് എപ്പോഴും സംഖ്യകളായിരിക്കും. ഇൻഡക്സ് സംഖ്യ തുടങ്ങുന്നത് പൂജ്യത്തിൽ നിന്നാണ്. അറെ നിർമ്മിക്കുന്നതിന് PHP യിൽ array() എന്ന ഫങ്ഷൻ ഉപയോഗിക്കുന്നു.

ഇൻഡെക്സ്ഡ് അറെ നിർമ്മിക്കുന്ന വാക്യഘടന;

```
$array_name = array(value1,value2,value3,etc.);
```

അഥവാ

```
$array[key] = value;
```

ഇൻഡെക്സ്ഡ് അറെക്ക് ഉദാഹരണം

```
$regno = array(679867, 658367, 673590);
$colors = array("red", "brown", "yellow");
                    Or
$regno[0] = 679867;
$regno1] = 658367;
$regno[2] = 673590;
$colors[0] = "red";
$colors[1] = "brown";
$colors[2] = "yellow";
```

മുകളിൽ നൽകിയിരിക്കുന്ന ഉദാഹരണത്തിൽ രജിസ്റ്റർ നമ്പറുകൾ (679867, 658367,673590), കളറുകൾ ("red", "brown", "yellow") എന്നിവ ഈ അറെകളിലെ വിലകളും, 0,1,2 എന്നീ സംഖ്യകൾ കീ അഥവാ ഇൻഡെക്സും ആകുന്നു.

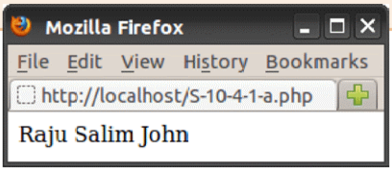
PHP-യിലുള്ള print_r( ) എന്ന ഫങ്ഷൻ ഉപയോഗിച്ച് ഒരു അറെയിലെ എല്ലാ വിലകളും പ്രിന്റ് ചെയ്യുവാൻ സാധിക്കും.

print_r( )ന്റെ വാക്യഘടന

```
print_r(array name)
```

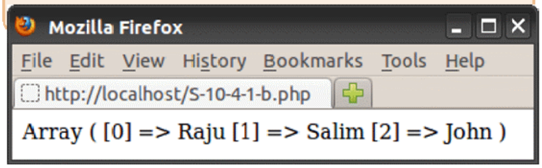
ഉദാഹരണം: for ലൂപ് ഉപയോഗിച്ച് അറേ പ്രിന്റ് ചെയ്യുന്നു

```
<?php
$Name = array("Raju",
          "Salim", "John");
for ($i = 0; $i < 3; $i++)
     echo $Name[$i]." ";
?>
```



ഉദാഹരണം: print_r( ) ഉപയോഗിച്ച് അറേ പ്രിന്റ് ചെയ്യുന്നു

```
<?php
  $Name = array("Raju",
            "Salim", "John");
  print_r($Name);
?>
```



10.4.2 അസ്സോസിയേറ്റീവ് അറെകൾ (Associative Arrays)

ഇത്തരം അറെകളിൽ കീകൾക്ക് സൗകര്യപ്രദമായ പേരുകൾ നൽകാം. അറെയുടെ പ്രവർത്തന രീതി ഇൻഡെക്സ്ഡ് അറെയിലേതുപോലെയാണ് എന്നാൽ കീ അഥവാ ഇൻഡക്സ് ഒരു സ്ട്രിംഗ് ആയിരിക്കും.

അസ്സോസിയേറ്റീവ് അറെയുടെ വാക്യ ഘടന

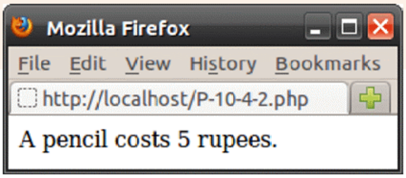
```
array(key=>value, key=>value, key=>value, etc.);
```

വിദ്യാർഥികളുടെ മാർക്കുകൾ സൂക്ഷിക്കുവാൻ സംഖ്യകൾ കീ ആയി വരുന്ന ഇൻഡെക്സ്ഡ് അറെയെക്കാൾ അഭികാമ്യം അസ്സോസിയേറ്റീവ് അറെകളാണ്. എന്തെന്നാൽ വിദ്യാർത്ഥികളുടെ പേരുകൾതന്നെ കീ രൂപത്തിൽ നൽകുവാൻ കഴിയും. അങ്ങിനെ വന്നാൽ ഓരോ പേരിന് നേരെയും സൂചിപ്പിച്ചിരിക്കുന്നത് അവരവരുടെ മാർക്ക് തന്നെയായിരിക്കും.

```
$result = array("Hari" => "35", "John" => "37", "Mini" => "43");
```

അസ്സോസിയേറ്റീവ് അറെക്ക് ഉദാഹരണം

```
<?php
  $cost=array("Pen"=>"35","Pencil"=>"5","Brush"=>"20");
  echo "A pencil costs " . $cost['Pencil'] . " rupees.";
?>
```



10.4.3 foreach ലൂപ്പുകൾ (foreach loops)

PHP-യിലെ for ലൂപ്പുകളുടെ തുടർച്ചയാണ് foreach ലൂപ്പ്. അറെയിലെ അംഗങ്ങളുടെ എണ്ണം കൃത്യമായി അറിയില്ലെങ്കിൽ ഈ ലൂപ്പ് ഉപയോഗിക്കുന്നത് നന്നായിരിക്കും. അറെയുടെ അവസാനം വരെയുള്ള എല്ലാ കീ/മൂല്യം ജോഡികളിലൂടെയും ഇത് കടന്നു പോകുന്നു. ഈ ലൂപ്പ് അറെകളിൽ മാത്രമാണ് പ്രവർത്തിക്കുന്നത് എന്നൊരു പ്രത്യേകതയുണ്ട്. ഇത് രണ്ടു രീതിയിൽ ഉപയോഗിക്കാം.

a. foreach ($array as $value)

ഓരോ ലൂപ്പ് ആവർത്തനത്തിലും തൽസ്ഥാനത്തുള്ള അറെ അംഗത്തിന്റെ വില $value വിലേക്ക് നൽകുകയും അറെ പോയിന്റർ ഒരു സ്ഥാനം മുന്നോട്ട് നീങ്ങുകയും ചെയ്യുന്നു. ഈ പ്രവർത്തനം അറെയിലെ അവസാന അംഗം എത്തുന്നതു വരെ ആവർത്തിക്കുന്നു.

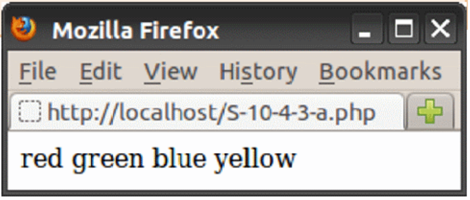
വാക്യഘടന

```
foreach ($array as $value)
  {
    //code to be executed;
  }
```

ഉദാഹരണം

Foreach ലൂപ്പിന്റെ പ്രവർത്തനം വിശദീകരിക്കുന്നു.

```
<?php
  $colors = array("red",
   "green","blue","yellow");
  foreach ($colors as $value)
      {echo "$value "; }
?>
```



b. foreach ($array as $key=>$value)

ഇത്തരം **Foreach** ലൂപ്പിൽ ഇവിടെ ഓരോ ലൂപ്പ് ആവർത്തനത്തിലും തൽസ്ഥാനത്തുള്ള അറെ അംഗത്തിന്റെ വില $value വിലേക്ക് നൽകുന്നതിനോടൊപ്പം അതിന്റെ കീ $key ലേക്ക് നൽകപ്പെടുകയും അറെ പോയിന്റർ ഒരു സ്ഥാനം മുന്നിലേക്ക് നീങ്ങുകയും ചെയ്യുന്നു. ഈ പ്രവർത്തനം അറെയിലെ അവസാന അംഗം എത്തുന്നത് വരെ ആവർത്തിക്കുന്നു.

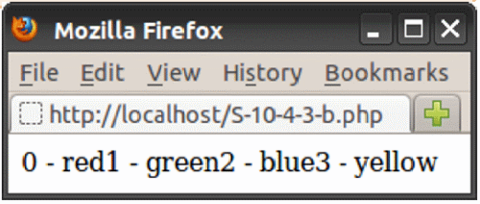
വാക്യഘടന

```
foreach ($array as $key=>
                    $value)
  {
    //code to be executed;
  }
```

ഉദാഹരണം

For each ലൂപ്പിന്റെ പ്രവർത്തനം വിശദീകരിക്കുന്നു.

```
<?php
$colors = array("red",
   "green","blue", "yellow");
foreach($colors as $key => $item)
   { echo "$key - $item"; }
?>
```



10.5 PHP-യിലെ ഫങ്ഷനുകൾ (Functions in PHP)

മറ്റു കമ്പ്യൂട്ടർ പ്രോഗ്രാമിങ് ഭാഷകളിലേതുപോലെ PHP-യിലും ഫങ്ഷനുകൾ രണ്ടുതരത്തിലുണ്ട്. ഉപയോക്തൃ നിർവചിത ഫങ്ഷനുകളും (user-defined) അന്തർ നിർമിത ഫങ്ഷനുകളും (built-in).

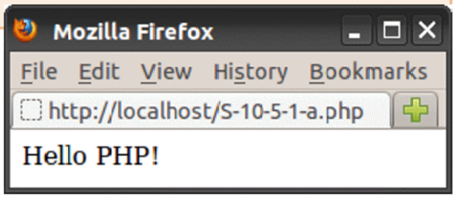
10.5.1 PHP-യിലെ ഉപയോക്തൃ നിർവചിത ഫങ്ഷനുകൾ (User-defined Functions in PHP)

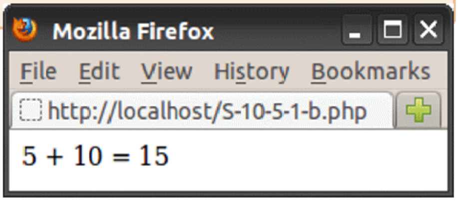
ഇത്തരം ഫങ്ഷനുകളുടെ നിർവചനം ഫങ്ഷൻ (function) എന്ന കീവേർഡോടു കൂടി ആരംഭിക്കുന്നു. PHP വേരിയബിളുകൾക്ക് പേരു നൽകുമ്പോൾ പാലിക്കേണ്ട നിയമങ്ങൾ തന്നെയാണ് ഫങ്ഷനുകൾക്ക് പേരു നൽകുമ്പോഴും പിന്തുടരേണ്ടത്. ഫങ്ഷൻ നിർവചനത്തിന്റെ വാക്യ ഘടന ചുവടെ നൽകിയിരിക്കുന്നു.

```
function functionName()
          {
           //code to be executed;
          }
```

ലളിതമായ ഫങ്ഷൻ	ആർഗ്യുമെന്റും റിട്ടേൺ വിലയും ഉള്ള ഫങ്ഷൻ

`<?php`
`function hello()`
` {`
`   echo "Hello PHP!";`
` }`
` hello(); //ഫങ്ഷനെ വിളിക്കുന്നു`
`?>`
	`<?php`
	` function add($a, $b)`
	`   {$c = $a + $b;`
	` return $c; }`
	` echo "5+10 =".add(5,10);`
	`?>`





നമുക്ക് ചെയ്യാം

അഞ്ച് വിദ്യാർഥികളുടെ പേരുകളും മാർക്കുകളും സംഭരിക്കുന്നതിനുള്ള ഒരു അസോസിയേറ്റീവ് അറെ നിർമിക്കുക. ഈ വിവരങ്ങൾ അറെയിൽ നിന്നും എടുത്ത് മോണിറ്ററിൽ പ്രദർശിപ്പിക്കുക. (ഇവിടെ കീ 'name' ഉം വാല്യൂ 'mark' ഉം ആകുന്നു)

10.5.2 ബിൽട്ട്-ഇൻ- ഫങ്ഷനുകൾ (Built-in Functions)

PHP യിൽ ധാരാളം ബിൽട്ട്-ഇൻ ഫങ്ഷനുകൾ ലഭ്യമാണ്. അവയിൽ ചിലത് ചുവടെ നൽകിയിരിക്കുന്നു.

a. date( ) ഫങ്ഷൻ (date( ) function)

ഈ ഫങ്ഷൻ ഉപയോഗിച്ച് തീയതി, സമയം, എന്നിവയുമായി ബന്ധപ്പെട്ട ഡാറ്റയെ അനുയോജ്യമായ ഘടനയിൽ (format) പ്രദർശിപ്പിക്കാം. ഈ ഫങ്ഷന്റെ വാക്യ ഘടന ചുവടെ നൽകുന്നു.

```
date(format, timestamp);
```

ഇതിൽ 'format' എന്നത് തീയതി പ്രദർശിപ്പിക്കുന്ന രീതിയും 'timestamp' എന്നത് ഒരു ദിവസത്തിന്റെ തീയതിയും സമയവും ആയിരിക്കും. ഇതിൽ 'timestamp' എന്ന ഘടകം date() ഫങ്ഷൻ ഉപയോഗിക്കുമ്പോൾ നിർബന്ധമില്ല. തനത് (default) ടൈം സ്റ്റാമ്പ് കംപ്യൂട്ടറിലെ ഇന്നത്തെ തീയതിയും സമയവും ആയിരിക്കും. തീയതി പ്രദർശിപ്പിക്കുന്ന രീതിക്ക് സാധാരണയായി ഉപയോഗിക്കുന്ന ചില അക്ഷരങ്ങൾ ചുവടെ നൽകിയിരിക്കുന്നു.

d - മാസത്തിലെ ദിവസം (01 to 31)

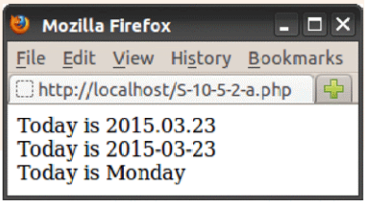
m - മാസം (01 to 12)

y - വർഷം (നാല് അക്കങ്ങൾ ഉള്ളത്)

l - ആഴ്ചയിലെ ദിവസം

date( ) ഫങ്ഷന് ഉദാഹരണം

```
<HTML>
<BODY>
    <?php
    echo "Today is ".date("Y.m.d")."<BR>";
    echo "Today is ".date("Y-m-d")."<BR>";
    echo "Today is " . date("l");
    ?>
</BODY>
</HTML>
```



b. സ്ട്രിംഗ് ഫങ്ഷനുകൾ (String functions)

chr() -തന്നിരിക്കുന്ന ആസ്കി (ASCII) കോഡിന് തത്തുല്യമായ ഒരു അക്ഷരം ഈ ഫങ്ഷൻ തിരികെ തരുന്നു.

വാക്യഘടന : chr(ascii code);

strlen() തന്നിരിക്കുന്ന സ്ട്രിങ്ങിന്റെ നീളം കണ്ടുപിടിക്കാൻ ഉപയോഗിക്കുന്നു

വാക്യഘടന : strlen(string);

strpos() ഒരു സ്ട്രിങ്ങിനുള്ളിൽ മറ്റൊരു സ്ട്രിങ്ങിന്റെ ആദ്യത്തെ സ്ഥാനം കണ്ടെത്തുന്നു.

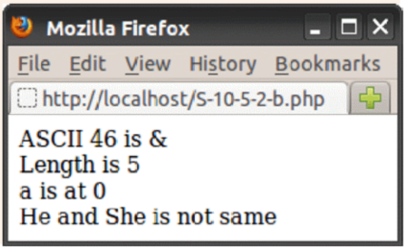
വാക്യഘടന : strpos(string, find, start);

strcmp() ആവരണ ചിഹ്നത്തിനുള്ളിൽ നൽകിയിട്ടുള്ള രണ്ടു സ്ട്രിങ്ങുകൾ തമ്മിൽ താരതമ്യം ചെയ്യുന്നു

വാക്യഘടന : strcmp(string1, string2);

സ്ട്രിങ് ഫങ്ഷനുകൾക്ക് ഉദാഹരണം

```
<HTML>
<BODY>
    <?php
    echo "ASCII 46 is ".chr(046)."<BR>";
    echo "Length is ".strlen("Apple")."<BR>";
    echo "a is at ".strpos("apple","a")."<BR>";
    if (strcmp("He", "She"))
    echo "He and She is not same";
    else
    echo  "He and She is same";
    ?>
</BODY>
</HTML>
```



നിങ്ങളുടെ പുരോഗതി അറിയുക

1. PHP-യിൽ array( ) ഫങ്ഷന്റെ ഉപയോഗം എന്ത്?
2. PHP-യിലെ വിവിധതരം ലൂപ്പുകൾ ഏതെല്ലാം?
3. PHP-യിലെ വിവിധതരം അറെകൾ ഏതെല്ലാം?
4. ചുവടെ ചേർത്തിരിക്കുന്നവയിൽ അറെ നിർമ്മിക്കുന്നതിനുള്ള ശരിയായ പ്രസ്താവനകൾ എടുത്തെഴുതുക:

```
i.    state[0]    = "karnataka";
ii.   $state[]    = array("karnataka");
iii.  $state[0]   = "karnataka";
iv.   $state      = array("karnataka");
```

a. iii ഉം iv ഉം b. ii ഉം iii ഉം c. i മാത്രം d. ii, iii , i എന്നിവ

5. സ്ട്രിംഗ് കീകൾ ഉപയോഗിക്കുന്ന അറെകളെ ______ എന്ന് വിളിക്കുന്നു.
6. ചുവടെ ചേർത്തിരിക്കുന്ന കോഡിന്റെ ഔട്ട്പുട്ട് എന്താണ്?

```
$Rent = 250;
  function Expenses($Other)
  {
     $Rent = 250 + $Other;
     return $Rent;
  }
  Expenses(50);
  echo $Rent;
```

a. 300 b. 250 c. 200 d. പ്രോഗ്രാം കംപയിൽ ചെയ്യില്ല

7. ഒരു സ്ട്രിങ്ങിന്റെ നീളം കണ്ടുപിടിക്കുന്നതിനുള്ള PHP ഫങ്ഷൻ ______ ആകുന്നു.
8. ഇപ്പോഴത്തെ സമയവും തീയതിയും PHP യിൽ നമുക്ക് എങ്ങിനെ കണ്ടുപിടിക്കാം?

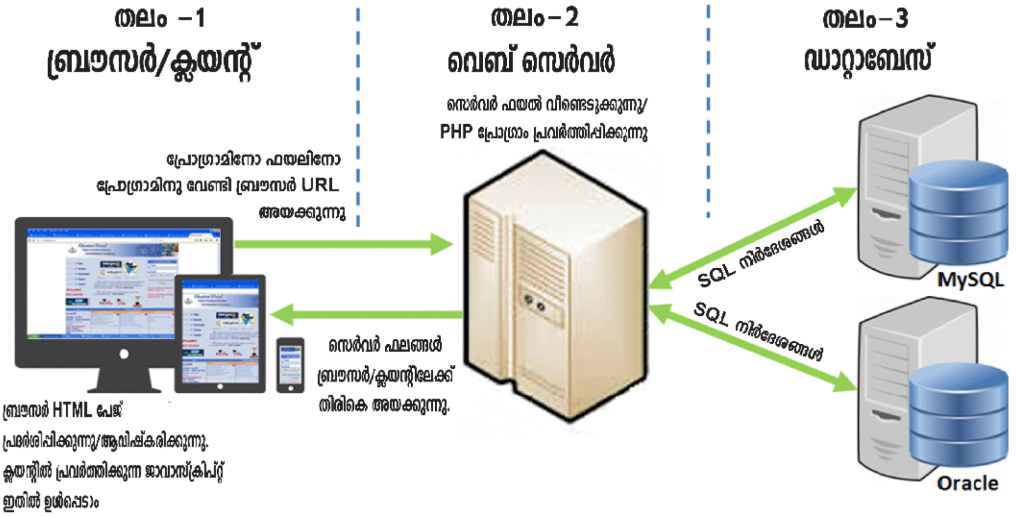
നമുക്ക് ചെയ്യാം

1. ഒരു സംഖ്യയുടെ ഫാക്ടോറിയൽ കാണുന്നതിനുള്ള ഫങ്ഷൻ PHP ഉപയോഗിച്ച് എഴുതുക. സംഖ്യ ഫങ്ഷനിലേക്ക് പാരാമീറ്റർ ആയി നൽകണം.
2. നടപ്പ് വർഷം പ്രദർശിപ്പിക്കുന്ന ഒരു PHP പ്രോഗ്രാം എഴുതുക.

10.6 PHP-യിലെ ത്രിതല ഘടന (Three Tier Architecture in PHP)

PHP-യിലെ ത്രിതല ഘടന ഒരു ക്ലയന്റ് സെർവർ ഘടനയാണ്. ഉപയോക്തൃ സമ്പർക്ക മുഖം (user interface), ആപ്ലിക്കേഷൻ പ്രോഗ്രാമുകൾ, ഡാറ്റ സംഭരണം തുടങ്ങി വിവിധ പ്രവർത്തനങ്ങൾക്കനുസരിച്ച് തട്ടുകളായി തരം തിരിച്ചിരിക്കുന്നു. ഈ തട്ടുകളുടെ തല ങ്ങൾ (Tier) എന്നും വിളിക്കുന്നു. ഈ മൂന്നു തലങ്ങളും ഒരു കമ്പ്യൂട്ടറിൽ തന്നെ സജ്ജ മാക്കാം. എന്നാൽ ത്രിതല ഘടനയിൽ സാധാരണയായി ഈ തലങ്ങളെ വ്യത്യസ്ത

കമ്പ്യൂട്ടറികളിലായിരിക്കും ക്രമീകരിക്കുന്നത് (Fig. 10.3). ത്രിതല ഘടനയിൽ ഒരു തട്ടിൽ വരുത്തുന്ന മാറ്റങ്ങൾ മറ്റുള്ള തട്ടുകളെ ബാധിക്കുകയില്ല. വെബ് അധിഷ്ഠിത ആപ്ലിക്കേഷൻ വികസനത്തിലെ മൂന്ന് തട്ടുകൾ അഥവാ തലങ്ങൾ താഴെ നൽകിയിരിക്കുന്നു.



ചിത്രം10.3: ത്രിതല ഘടന

തലം 1: ഉള്ളടക്കം അടങ്ങിയ വെബ്പേജിന്റെ Front end ബ്രൗസർ പ്രദർശിപ്പിക്കുന്നു (ഉദാ: PHP ക്ക് Apache). വെബ് സെർവറിൽ നിന്ന് ബ്രൗസറിന് ലഭിക്കുന്ന ഉള്ളടക്കത്തിനനുസരിച്ച് (HTML അല്ലെങ്കിൽ ജാവാ സ്ക്രിപ്റ്റ് ഉൾക്കൊള്ളിച്ച HTML) ബ്രൗസർ സ്റ്റാറ്റിക് അല്ലെങ്കിൽ ഡയനാമിക് പേജ് നിർമ്മിക്കുന്നു.

തലം 2: അപ്ലിക്കേഷൻ സർവർ അല്ലെങ്കിൽ വെബ് സർവർ അടങ്ങിയതും ഡൈനാമിക് പ്രോസസ്സിംഗിനു വേണ്ടിയുള്ളതുമായ തലമാണിത് (മധ്യതലം). ഇത് ബ്രൗസറിൽ/ക്ലയന്റിൽ നിന്നുള്ള അഭ്യർത്ഥനകൾ സ്വീകരിച്ച്, ആവശ്യപ്പെട്ട സ്ക്രിപ്റ്റ് പ്രവർത്തിപ്പിക്കുകയും ഔട്ട്പുട്ട് HTML ആയി ബ്രൗസറിലേക്ക്/ക്ലയന്റിലേക്ക് തിരിച്ച് അയക്കുകയും ചെയ്യുന്നു.

തലം 3: Back end ലെ ഈ തലത്തിൽ പ്രധാനമായി ഉൾക്കൊള്ളുന്നത് ഒരു ഡാറ്റാബേസ് അഥവാ ഡാറ്റാസ്റ്റോർ ആയിരിക്കും (ഉദാ: MySQL Server). ഇവിടെ ഡാറ്റയുടെ കൂട്ടവും അത് കൈകാര്യം ചെയ്യുന്ന DBMS സോഫ്ട്‌വെയറും ഉണ്ടായിരിക്കും. DBMS സോഫ്ട്‌വെയർ വെബ് സെർവറിൽ നിന്നും അയച്ച SQL കമാന്റുകൾ മനസ്സിലാക്കുകയും അവ പ്രവർത്തിപ്പിക്കുകയും വെബ് സെർവറിലേക്ക് ഫലങ്ങളെ തിരിച്ച് അയക്കുകയും ചെയ്യുന്നു.

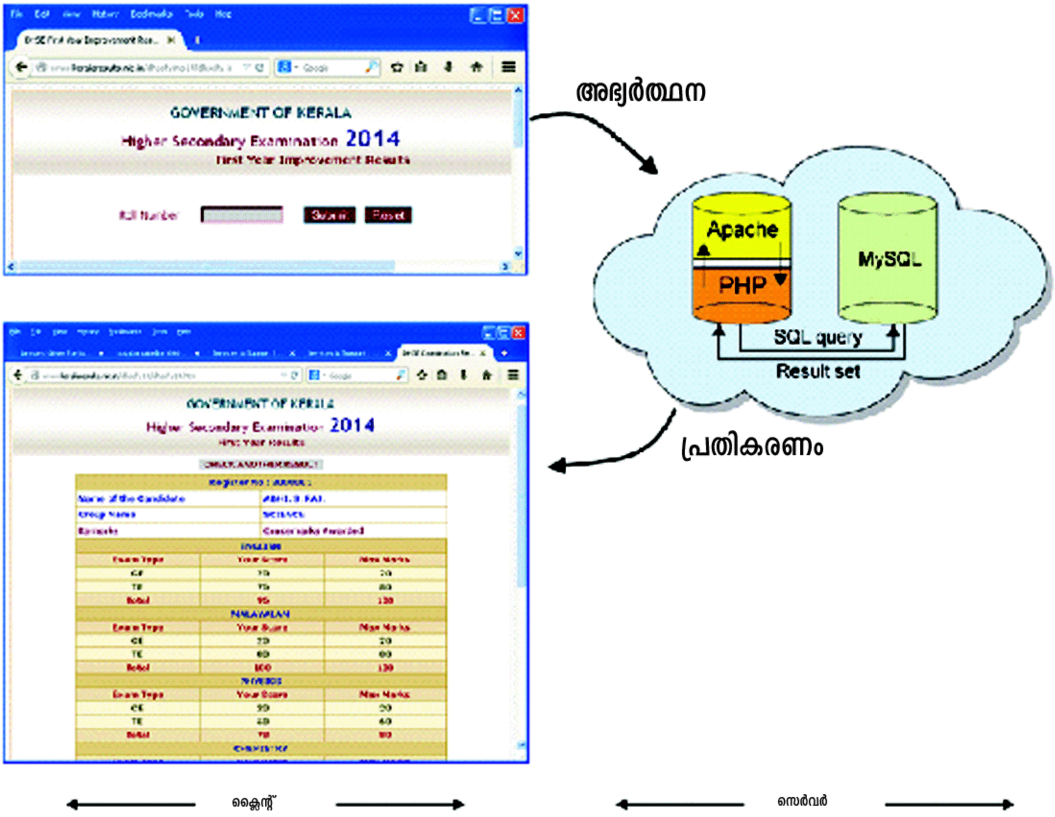
HTML, MySQL, ജാവാസ്ക്രിപ്റ്റ് ഉപയോഗിച്ചുള്ള ക്ലയന്റ് സൈഡ് സ്ക്രിപ്റ്റിംഗ് എന്നിവ നമ്മൾ ഇതിനോടകം പഠിച്ച് കഴിഞ്ഞു. PHP-യെ എങ്ങിനെ ക്ലയന്റ്, ഡാറ്റാബേസ് എന്നിവയുമായി ബന്ധിപ്പിച്ച് ത്രിതല ഘടന സജ്ജീകരിക്കാമെന്ന് അടുത്ത ഭാഗങ്ങളിൽ നാം പഠിക്കും.

10.7 PHP ഫോമുകൾ (PHP Forms)

ക്ലയന്റിൽ നിന്നും വിവരങ്ങൾ ശേഖരിച്ച് സെർവറിന് നൽകുന്നതിനായി PHP ഫോമുകൾ ഉപയോഗിക്കുന്നു. ടെക്സ്റ്റ് ബോക്സ്, കോംബോ ബോക്സ്, ബട്ടൺ മുതലായ എല്ലാ ഫോം ഘടകങ്ങളിൽ നിന്നും ഇൻപുട്ട് സ്വീകരിക്കാനും അവ വേണ്ടുന്ന രീതിയിൽ കൈകാര്യം ചെയ്യാനും PHP-ക്ക് കഴിയും. സെർവറിലേക്ക് ക്ലയന്റ് എങ്ങനെയാണ് വിവരങ്ങൾ അയയ്ക്കുന്നത്, സെർവറുമൊത്ത് എങ്ങനെയാണ് വിവരങ്ങൾ ശേഖരിക്കുന്നത്, എങ്ങനെ സെർവർ ക്ലയന്റിലേക്ക് ഉത്തരങ്ങൾ തിരികെ അയയ്ക്കുന്നു എന്നിവയെക്കുറിച്ച് ഈ ഭാഗത്ത് നാം പഠിക്കും.

ഫോമുകളുടെ പ്രവർത്തനത്തെ വിശദീകരിക്കുന്ന ഒരു ഉദാഹരണം നമുക്ക് നോക്കാം.

താഴെ നൽകിയിരിക്കുന്ന ഉദാഹരണത്തിൽ, ക്ലയന്റ് വശത്തുള്ള ബ്രൗസറിൽ പ്രദർശിപ്പിച്ചിരിക്കുന്ന ഫോമിലുള്ള ടെക്സ്റ്റ് ബോക്സിൽ ഒരു വിദ്യാർത്ഥിയുടെ രജിസ്റ്റർ നമ്പർ ശേഖരിക്കും (ചിത്രം 10.4). Submit ബട്ടൺ അമർത്തുമ്പോൾ, ക്ലയന്റ് ഈ രജിസ്റ്റർ നമ്പർ സെർവറിലേക്ക് അയയ്ക്കുന്നു. സെർവർ ഈ രജിസ്റ്റർ നമ്പർ സ്വീകരിച്ച് ഡാറ്റാബേസിൽ (ഇവിടെ MySQL) തിരയുന്നു. ഇത് കണ്ടെത്തിയാൽ, സെർവർ ഡാറ്റാബേസിൽ നിന്ന് ലഭിച്ച ഡാറ്റ അടങ്ങിയിരിക്കുന്ന വെബ് പേജ് നിർമിക്കുന്നു. കണ്ടെത്തിയില്ലെങ്കിൽ 'Wrong Register No' എന്ന സന്ദേശം അടങ്ങുന്ന ഒരു വെബ് പേജ് നിർമിക്കുന്നു. ഈ വെബ് പേജ് പിന്നീട് ക്ലയന്റിലേക്ക് തിരിച്ച് അയയ്ക്കുന്നു.



ചിത്രം10.4: ക്ലയന്റിനും സെർവറിനും ഇടക്കുള്ള വിവരത്തിന്റെ പ്രവാഹം

ഉപയോക്താക്കളിൽ നിന്ന് ഡാറ്റ സ്വീകരിക്കുന്നതിനുള്ള ഒരു ഫോം എങ്ങനെ സൃഷ്ടിക്കാമെന്ന് ചർച്ചചെയ്യുന്നതിന് മുമ്പ് ഗ്ലോബൽ വേരിയബിളുകളെക്കുറിച്ച് നാം മനസ്സിലാക്കേണ്ടതുണ്ട്.

10.7.1 PHP ഗ്ലോബൽ വേരിയബിളുകൾ (PHP Global Variables)

കോഡിനുള്ളിൽ ഏതു ഭാഗത്തു (ഫങ്ഷൻ, ക്ലാസ് അല്ലെങ്കിൽ ഫയലിനുള്ളിൽ) നിന്നും ഉപയോഗിക്കാവുന്നവയാണ് ഗ്ലോബൽ വേരിയബിളുകൾ സാധാരണയായി ഗ്ലോബൽ വേരിയബിളുകൾ സ്ക്രിപ്റ്റിന്റെ ആദ്യ ഭാഗത്ത് ഒരു ഫങ്ഷന് പുറത്താണ് പ്രസ്താവിക്കുന്നത്.

സൂപ്പർ ഗ്ലോബൽ അറെകൾ (Super global arrays)

"സൂപ്പർ ഗ്ലോബൽ" എന്ന് വിളിക്കുന്ന ചില മുൻനിർവചിത ഗ്ലോബൽ അറെകൾ PHP-യിൽ ഉണ്ട്. സ്ക്രിപ്റ്റിന്റെ നിലവിലെ പരിധി പരിഗണിക്കാതെ അതിൽ എപ്പോഴും ലഭ്യമായിട്ടുള്ള ഒരു പ്രത്യേക വേരിയബിളാണ് സൂപ്പർഗ്ലോബൽ അറെ. ഓരോ സൂപ്പർഗ്ലോബലിനും ഉപയോഗപ്രദമായ വ്യത്യസ്ത ലക്ഷ്യമുണ്ട്.

സൂപ്പർഗ്ലോബൽ	വിവരണം
$GLOBALS	ഗ്ലോബൽ പരിധിയിൽ ലഭ്യമായ വേരിയബിളുകൾ
$_SERVER	സെർവർ സംബന്ധിച്ച വിവരം
$_GET	HTTP GET രീതി ഉപയോഗിച്ച് കൈമാറിയ ഡാറ്റ
$_POST	HTTP POST രീതി ഉപയോഗിച്ച് കൈമാറിയ ഡാറ്റ
$_REQUEST	ഒരു HTTP അഭ്യർത്ഥന വഴി കൈമാറിയ ഡാറ്റ
$_FILES	ഒരു HTML ഫയൽ ഇൻപുട്ട് വഴി കൈമാറിയ ഡാറ്റ
$_SESSION	ഉപയോക്താവിന് ലഭ്യമായ ഇപ്പോഴുള്ള സെഷനിലെ ഡാറ്റ
$_COOKIE	ഉപയോക്താവിന്റെ ബ്രൗസറിൽ ഒരു കുക്കി ആയി സൂക്ഷിച്ചിരിക്കുന്ന ഡാറ്റ

ഇവയിൽ ചില സുപ്രധാന സൂപ്പർ ഗ്ലോബലുകൾ ഇവിടെ നമുക്ക് ചർച്ച ചെയ്യാം.

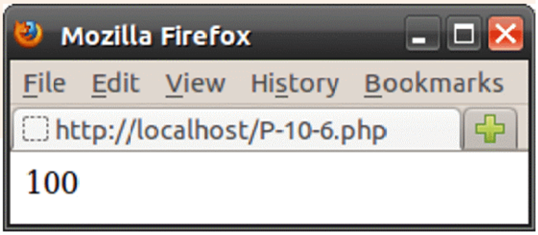
A. PHP $GLOBALS

PHP സ്ക്രിപ്റ്റിൽ എവിടെനിന്നു വേണമെങ്കിലും ഉപയോഗിക്കാവുന്ന സൂപ്പർ ഗ്ലോബൽ വേരിയബിളാണ് $GLOBALS. എല്ലാ ഗ്ലോബൽ വേരിയബിളുകളും $GLOBALS [ഇൻഡക്സ്] എന്ന അറെയിലാണ് PHP സൂക്ഷിക്കുന്നത്. ഇൻഡക്സ് സൂചിപ്പിക്കുന്നത് വേരിയബിളിന്റെ പേരാണ്.

$GLOBALS എന്ന സൂപ്പർ ഗ്ലോബൽ വേരിയബിൾ എങ്ങനെ ഉപയോഗിക്കുന്നു എന്ന് ചുവടെയുള്ള ഉദാഹരണത്തിൽ കാണിച്ചിരിക്കുന്നു.

Program 10.6 ഗ്ലോബൽ അറെ പരിശോധിക്കുന്നതിന്

```
<?php
  $x = 75;
  $y = 25;
  function addition() {
   $GLOBALS['z'] = $GLOBALS['x'] + $GLOBALS['y'];
  }
  addition();
  echo $z;
?>
```



മുകളിൽ ഉള്ള ഉദാഹരണത്തിൽ, 'z' എന്നത് $GLOBALS അറെയിലെ ഒരു വേരിയബിൾ ആയതിനാൽ, ഇത് ഫങ്ഷന് പുറമെയും ഉപയോഗിക്കാവുന്നതാണ്.

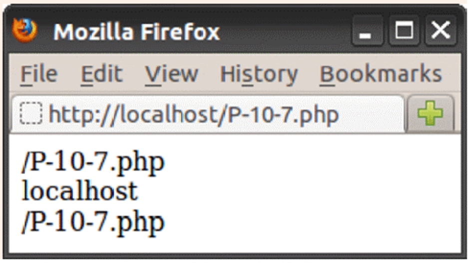
B. PHP $_SERVER

തലക്കെട്ടുകൾ, ഫയലിന്റെ സ്ഥാനം സൂചിപ്പിക്കുന്ന വഴികൾ, സ്ക്രിപ്റ്റിന്റെ സ്ഥാനങ്ങൾ എന്നിവയെക്കുറിച്ചുള്ള വിവരം സൂക്ഷിക്കുന്ന ഒരു PHP സൂപ്പർ ഗ്ലോബൽ വേരിയബിളാണ് $_SERVER.

$_SERVER-ലെ ചില ഘടകങ്ങൾ എങ്ങനെയാണ് ഉപയോഗിക്കേണ്ടതെന്ന് ചുവടെയുള്ള ഉദാഹരണം കാണിക്കുന്നു.

Program 10.7 $_SERVER വേരിയബിൾ പരിശോധിക്കുന്നതിന്

```
 <?php
echo $_SERVER['PHP_SELF'];
echo "<BR>";
echo $_SERVER['SERVER_NAME'];
echo "<BR>";
echo $_SERVER['SCRIPT_NAME'];
 ?>
```



മുകളിൽ ഉപയോഗിച്ച $ _SERVER ഘടകങ്ങളെ ഈ പട്ടിക വിശദീകരിക്കുന്നു:

ഘടകം / കോഡ്	വിവരണം
`$_SERVER['PHP_SELF']`	ഇപ്പോൾ പ്രവർത്തിക്കുന്ന സ്ക്രിപ്റ്റിന്റെ ഫയൽ നാമം തിരികെ നൽകുന്നു
`$_SERVER['SERVER_NAME']`	ഹോസ്റ്റ് സെർവറിന്റെ പേര് തിരികെ നൽകുന്നു
`$_SERVER['SCRIPT_NAME']`	$_SERVER['PHP_SELF'] പോലെ തന്നെയാണ് ഇതും

C. PHP $_REQUEST

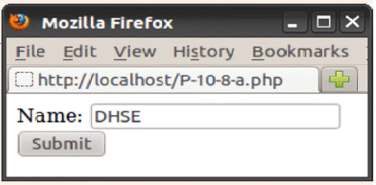
$_GET, $_POST, $_COOKIE എന്നീ സൂപ്പർ ഗ്ലോബലുകൾ ഉൾക്കൊള്ളുന്ന ഒരു അറെയാണ് $_REQUEST സൂപ്പർ ഗ്ലോബൽ. ഒരു HTML ഫോം സമർപ്പിച്ചതിനുശേഷം ഡാറ്റ ശേഖരിക്കാനാണ് ഇത് ഉപയോഗിക്കുന്നത്.

പ്രോഗ്രാം 10.8(a)-യിലെ കോഡ് ഇൻപുട്ട് ഫീൽഡും "Submit" ബട്ടണുമുള്ള ഒരു ഫോം കാണിക്കുന്നു. ഒരു ഉപയോക്താവ് Submit" -ൽ ക്ലിക്കുചെയ്ത് ഡാറ്റ സമർപ്പിക്കുമ്പോൾ, ഫോമിലെ ഡാറ്റ <FORM> ടാഗിലെ ആക്ഷൻ ആട്രിബ്യൂട്ടിൽ സൂചിപ്പിക്കുന്ന ഫയലിലേക്ക് അയയ്ക്കുന്നു. അപ്പോൾ പ്രോഗ്രാം 10.8(b) -യിലെ കോഡിൽ ഉപയോഗിച്ചിരിക്കുന്ന സൂപ്പർ ഗ്ലോബൽ വേരിയബിളായ $_REQUEST, ഇൻപുട് ഫീൽഡിലെ വില ശേഖരിക്കുകയും ഈ വിലയുടെ മുൻപിൽ 'ഹലോ' എന്ന വാചകം ചേർത്ത് പ്രദർശിപ്പിക്കുകയും ചെയ്യുന്നു.

ബാക്കിയുള്ള ആശയങ്ങൾ വിശദീകരിക്കുന്നതിന് ഇനി നമുക്ക് രണ്ടു പ്രോഗ്രാം കോഡുകൾ നോക്കാം. ഒരു HTML കോഡും ഒരു PHP കോഡും.

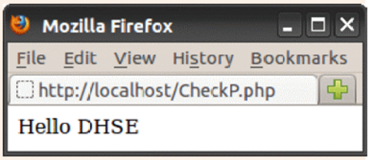
Program 10.8(a) $_REQUEST വേരിയബിൾ പരിശോധിക്കുന്നതിനുള്ള HTML

```
<HTML>
<BODY>
  <FORM Method="post" Action="CheckR.php">
  Name:
  <INPUT Type="text" Name="fname">
  <INPUT Type="submit" Value="Submit">
  </FORM>
</BODY>
</HTML>
```



Program 10.8(b) $_REQUEST വേരിയബിൾ പരിശോധിക്കുന്നതിനുള്ള CheckR.php

```
<?php
   $name = $_REQUEST['fname'];
   echo "Hello " . $name;
?>
```



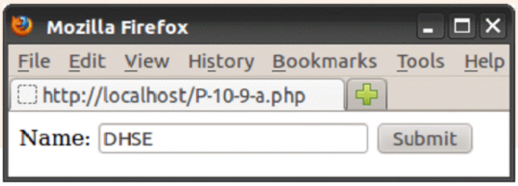
D. PHP $_POST

Method = "post" എന്ന രീതിയിൽ ഒരു HTML ഫോം സമർപ്പിച്ചതിനു ശേഷം ഫോമിലെ ഡാറ്റ ശേഖരിക്കാൻ PHP $_POST വ്യാപകമായി ഉപയോഗിച്ചു വരുന്നു.

ചുവടെയുള്ള ഉദാഹരണത്തിൽ ഒരു ഇൻപുട്ട് ഫീൽഡും ഒരു "Submit" ബട്ടണും ഉള്ള ഫോം കാണിച്ചിരിക്കുന്നു. ഒരു ഉപയോക്താവ് "Submit" ക്ലിക്കു ചെയ്യുമ്പോൾ <FORM> ടാഗിലെ ആക്ഷൻ ആട്രിബ്യൂട്ടിൽ കൊടുത്തിരിക്കുന്ന ഫയലിലേക്ക് ഫോമിലെ ഡാറ്റ അയയ്ക്കുന്നു. ഇപ്പോൾ ഇൻപുട്ട് ഫീൽഡിലെ വില ശേഖരിക്കാൻ $_POST എന്ന സൂപ്പർഗ്ലോബൽ വേരിയബിൾ നമുക്ക് ഉപയോഗിക്കാം.

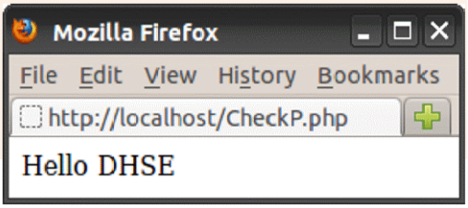
Program 10.9(a) $_POST വേരിയബിൾ പരിശോധിക്കുന്നതിനുള്ള HTML

```
<HTML>
<BODY>
  <FORM Method="post" Action="CheckP.php">
    Name:
    <INPUT Type="text" Name="fname">
    <INPUT Type="submit" Value = "Submit">
  </FORM>
</BODY>
</HTML>
```



Program 10.9(b) $_POST വേരിയബിൾ പരിശോധിക്കുന്നതിനുള്ള CheckP.php

```
<?php
  $name = $_POST['fname'];
  echo "Hello " . $name;
?>
```



E. PHP $_GET

Method = "get" എന്ന രീതിയിൽ ഒരു HTML ഫോം സമർപ്പിച്ചതിനു ശേഷം ഫോമിലെ ഡാറ്റ ശേഖരിക്കാൻ PHP $_GET ഉപയോഗിക്കുന്നു. $_GET-ന് URL-ൽ അയച്ച ഡാറ്റയും ശേഖരിക്കാനാകും

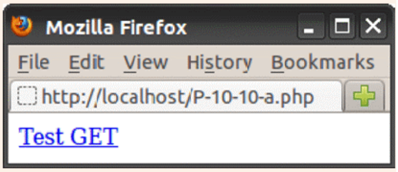
നമുക്ക് ചെയ്യാം

മുകളിൽ തന്നിരിക്കുന്ന ഉദാഹരണത്തിൽ ഫോമിലും സ്ക്രിപ്റ്റിലും "POST" നു പകരം "GET" ഉപയോഗിക്കുക. (Method="post" നു പകരം Method="get"), ($name=$_POST['fname']; നു പകരം $name=$_GET-['fname'];)

താഴെ കൊടുത്തിരിക്കുന്ന ഉദാഹരണത്തിലെ പോലെ GET ൽ ഹൈപ്പർ ലിങ്കിനോടൊപ്പം പരാമീറ്ററുകളും അയയ്ക്കാവുന്നതാണ്.

Program 10.10(a): $_GET ഉപയോഗിച്ച് പരാമീറ്ററോട് കൂടിയ ഹൈപ്പർ ലിങ്ക് പരിശോധിക്കുന്നതിനുള്ള HTML

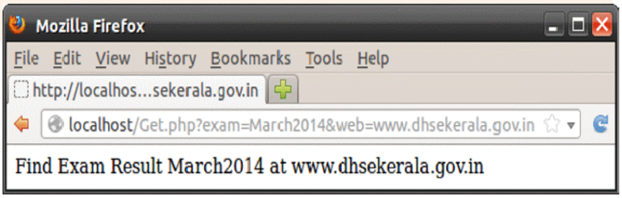
```
<HTML>
<BODY>
 <A Href="get.php?exam=March2014&web=
   www.dhsekerala.gov.in"> Test GET</A>
</BODY>
</HTML>
```



"Test GET" എന്ന ലിങ്കിൽ ഉപയോക്താവ് ക്ലിക്ക് ചെയ്യുമ്പോൾ, "exam", "web" എന്നീ പരാമീറ്ററുകൾ "get.php" യിലേക്ക് അയക്കുന്നു. അതിനു ശേഷം "get.php"-യിൽ അവയുടെ വിലകൾ $_GET കൊണ്ട് നമുക്ക് ഉപയോഗിക്കാം.

Program 10.10(b) $_GET വേരിയബിൾ പരിശോധിക്കുന്നതിനുള്ള Get.php

```
<?php
   echo "Find Exam Result ".$_GET['exam']
   echo " at ". $_GET['web'];
?>
```



10.7.2 get, post എന്നിവയുടെ താരതമ്യം (get and post comparison)

get, post എന്നിവ രണ്ടും ഫോമിൽ നിന്നും വെബ് സെർവറിലേക്ക് വിവരങ്ങൾ അയയ്ക്കുന്നതിനുള്ള രീതികളാണ്. രണ്ടും കീ/വില ജോഡികൾ അടങ്ങുന്ന ഒരു അറെ ഉണ്ടാക്കുന്നു (ഉദാഹരണം (കീ1 =>വില1, കീ2 =>വില2, കീ2> = വില3 ...)). ഇവിടെ കീകൾ ഫോം കൺട്രോളുകളുടെ പേരും വിലകൾ ഉപയോക്താവിൽ നിന്നുള്ള ഇൻപുട്ട് ഡാറ്റയുമാണ്.

get, post എന്നീ രീതികളുടെ അനുബന്ധ അറെയുടെ പേരുകൾ യഥാക്രമം $_GET, $_POST എന്നിവയാണ്. ഇവ സൂപ്പർ ഗ്ലോബലുകളാണ്. അതായത് പരിധിയില്ലാതെ എല്ലായ്പ്പോഴും ഉപയോഗിക്കാൻ കഴിയും, പ്രത്യേകിച്ച് ഒന്നും ചെയ്യാതെതന്നെ ഏതെങ്കിലും ഫങ്ഷൻ, ക്ലാസ് അല്ലെങ്കിൽ ഫയൽ എന്നിവയിൽ നിന്ന് നിങ്ങൾക്ക് അവ ഉപയോഗിക്കാം.

$_GET എന്നത് URL പാരാമീറ്ററുകൾ വഴി നിലവിലെ സ്ക്രിപ്റ്റിലേക്ക് കൈമാറുന്ന വേരിയബിളുകളുടെ ഒരു അറെ ആണ്.

$_POST എന്നത് HTTP POST രീതി വഴി നിലവിലെ സ്ക്രിപ്റ്റിലേക്ക് കൈമാറുന്ന വേരിയബിളുകളുടെ ഒരു അറെയാണ്.

A. get എപ്പോൾ ഉപയോഗിക്കാം? (When to use get?)

get രീതി ഉപയോഗിച്ച് ഒരു ഫോമിൽ നിന്നും അയച്ച വിവരങ്ങൾ എല്ലാവർക്കും ദൃശ്യമാകും (എല്ലാ വേരിയബിളിന്റെ പേരുകളും വിലകളും URL-ൽ പ്രദർശിപ്പിക്കും). get ന് 2000 അക്ഷരങ്ങൾ മാത്രമേ അയക്കാൻ കഴിയൂ.എന്നിരുന്നാലും, URL-ൽ വേരിയബിളുകൾ പ്രദർശിപ്പിച്ചിരിക്കുന്നതിനാൽ, പേജ് ബുക്ക്മാർക്ക് (bookmark) ചെയ്യാൻ കഴിയും. പ്രാധാന്യം കുറവായ ഡാറ്റ അയയ്ക്കാൻ get ഉപയോഗിക്കാവുന്നതാണ്.

പാസ്‌വേഡുകളോ അല്ലെങ്കിൽ മറ്റ് പ്രധാനപ്പെട്ട വിവരമോ അയയ്ക്കാൻ GET ഒരിക്കലും ഉപയോഗിക്കരുത്.

B. post എപ്പോൾ ഉപയോഗിക്കാം? (When to post?)

post രീതി ഉപയോഗിച്ച് ഒരു ഫോമിൽ നിന്ന് അയയ്ക്കുന്ന വിവരം മറ്റുള്ളവർക്ക് ദൃശ്യമാകുകയില്ല. (എല്ലാ പേരുകളും/വിലകളും HTTP അഭ്യർത്ഥനയുടെ ഭാഗമായി ഉൾക്കൊള്ളിക്കുന്നു.) കൂടാതെ അയക്കുന്ന ഡാറ്റയുടെ അളവിന് പരിധി ഇല്ല. ഈ രീതി സുരക്ഷിതമായതിനാൽ, ഫോമിൽ നിന്ന് ഡാറ്റ അയക്കുന്നതിന് മിക്ക ഡെവലപ്പർമാരും post ഉപയോഗിക്കുന്നതിന് മുൻഗണന നൽകുന്നു. എന്നിരുന്നാലും URL ൽ വേരിയബിളുകൾ പ്രദർശിപ്പിക്കാത്തതിനാൽ, പേജ് ബുക്ക്മാർക്ക് ചെയ്യാൻ സാധ്യമല്ല.

Method = "get"	Method = "post"
URL-ന്റെ ഭാഗമായി ഉള്ളടക്കങ്ങൾ കൈമാറ്റം ചെയ്യപ്പെടുന്നതുകൊണ്ട് ഡാറ്റ അഡ്രസ് ബാറിൽ ദൃശ്യമാകും.	ഉള്ളടക്കം ഇൻപുട്ട് ഫയലായി സ്ക്രിപ്റ്റുകളിലേക്ക് കൈമാറുന്നതിനാൽ ഡാറ്റ ദൃശ്യമാകില്ല
പേജ് ലിങ്ക് ബുക്ക്മാർക്ക് ചെയ്യാൻ കഴിയും	പേജ് ലിങ്ക് ബുക്ക്മാർക്ക് ചെയ്യാൻ കഴിയില്ല
URL-ന്റെ ഭാഗമായി ഡാറ്റ സമർപ്പിക്കുന്നു.	HTTP അഭ്യർത്ഥനയുടെ ഭാഗമായി ഡാറ്റ സമർപ്പിക്കുന്നു.
വേഗത്തിലാണ് ഡാറ്റ അയക്കുന്നതെങ്കിലും സുരക്ഷിതമല്ല.	ഡാറ്റ അയയ്ക്കൽ സുരക്ഷിതമാണ്, എന്നാൽ get-നെ അപേക്ഷിച്ച് വേഗത കുറവാണ്.
get-ന് 2000 അക്ഷരങ്ങൾ മാത്രമേ അയക്കാൻ കഴിയൂ.	ഡാറ്റ അയക്കുന്നതിന് പരിധിയില്ല

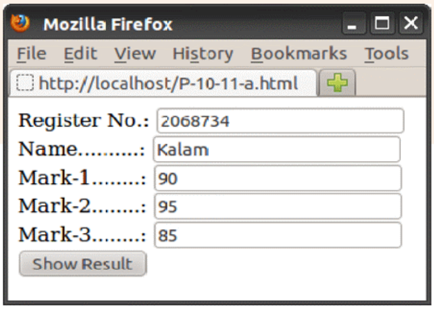
പ്രോജക്റ്റ് (Project)

ഒരു ഫോം എങ്ങനെ ഉപയോക്താവിൽ ക്ലയന്റിൽ നിന്നും ഡാറ്റ ശേഖരിക്കുന്നു, അത് എങ്ങനെ സെർവറിലേക്ക് അയയ്ക്കുന്നു (സമർപ്പിക്കുന്നു), എങ്ങനെ സെർവർ ഉപയോക്താവിന്/ക്ലയന്റിന് ഫലങ്ങൾ തിരികെ അയയ്ക്കുന്നു എന്നൊക്കെ നമുക്ക് ഇപ്പോൾ അറിയാം. വെബ് പ്രോഗ്രാമിങ്ങിനെക്കുറിച്ച് നാം പഠിച്ച കാര്യങ്ങൾ വിശദീകരിക്കുന്ന ഒരു പ്രോജക്റ്റ് പ്രവർത്തനക്ഷമമാക്കി കൊണ്ട് നമുക്ക് ഇത് ക്രോഡീകരിക്കാം.

രജിസ്റ്റർ നമ്പർ, പേര്, മൂന്ന് വിഷയത്തിന്റെ മാർക്കുകൾ (100-ൽ) എന്നിവ സ്വീകരിക്കുന്നതിനുള്ള ഒരു ഫോം സൃഷ്ടിക്കുകയും, അത് സെർവറിലേക്ക് സമർപ്പിക്കുകയും ചെയ്യുക. സെർവറിൽ പ്രവർത്തിക്കുന്ന സ്ക്രിപ്റ്റ് ഈ ഡാറ്റ സ്വീകരിച്ച് ഫലം കണ്ടെത്തി തിരികെ ക്ലയന്റിലേക്ക് അയക്കുന്നു. പരീക്ഷ ജയിക്കുന്നതിന് ഒരു വിദ്യാർഥിക്ക് ഓരോ വിഷയത്തിലും കുറഞ്ഞത് 40 മാർക്ക് ലഭിക്കണം. ആകെ മാർക്ക് <150 ആണ് എങ്കിൽ 'വിജയിച്ചു', ആകെ മാർക്ക്> = 150-ഉം <180-ഉം ആയാൽ 'സെക്കൻഡ് ക്ലാസ്', ആകെ മാർക്ക്> = 180-ഉം <225-ഉം ആയാൽ 'ഫസ്റ്റ് ക്ലാസ്സ്', ആകെ മാർക്ക്> = 225 ആണ് എങ്കിൽ 'ഡിസ്റ്റിങ്ഷൻ'.

Program 10.11(a) വിദ്യാർഥിയുടെ ഡാറ്റ സ്വീകരിക്കുന്ന ഫോം HTML

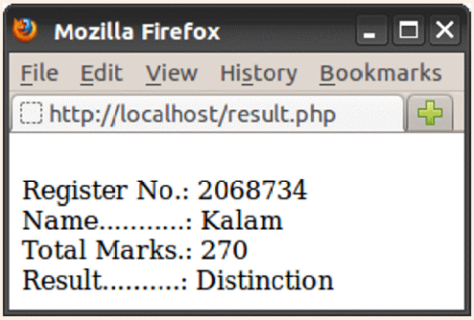
```
</HTML>
</BODY>
  <FORM Method="post" Action="Result.php">
     Register No:
     <INPUT Type="text" Name="regno"> <BR>
     Name..............:
     <INPUT Type="text" Name="sname"> <BR>
     Mark-1..............:
     <INPUT Type="text" Name="mark1"> <BR>
     Mark-2..............:
     <INPUT Type="text" Name="mark2"> <BR>
     Mark-3..............:
     <INPUT Type="text" Name="mark3"> <BR>
     <INPUT Type="submit" Value="Show Result">
  </FORM>
</BODY>
</HTML>
```



Program 10.11(b) ഫലം പ്രദർശിപ്പിക്കുന്നതിനുള്ള സ്ക്രിപ്റ്റ് Result.PHP

```
<?php
  $m1 = $_POST ['mark1'];
  $m2 = $_POST ['mark2'];
  $m3 = $_POST ['mark3'];
  $tmarks = $m1 + $m2 + $m3;
  if (($m1<40) || ($m1<40) || ($m1<40))
     $result = "Failed";
```

```
  elseif ($tmarks < 150)
     $result = "Passed";
  elseif ($tmarks < 180)
     $result = "Second Class";
  elseif ($tmarks < 225)
     $result = "First Class";
  else
     $result = "Distinction";
  echo "<BR> Register No.: " . $_POST ['regno'];
  echo "<BR> Name..................: " . $_POST ['sname'];
  echo "<BR> Total Marks.: " . $tmarks;
  echo "<BR> Result..............: " . $result;
?>
```



നമുക്ക് ചെയ്യാം

രജിസ്റ്റർ നമ്പർ സ്വീകരിക്കുന്നതിനും "display.php"-യിലേക്ക് സമർപ്പിക്കുന്നതിനുള്ള ഒരു HTML ഫോം നിർമിക്കുക. display.php ലഭിച്ച രജിസ്റ്റർ നമ്പർ ബ്രൗസറിൽ പ്രദർശിപ്പിക്കണം.

മേൽ പറഞ്ഞ പ്രവർത്തനങ്ങൾ get രീതി ഉപയോഗിച്ച് ചെയ്യുക

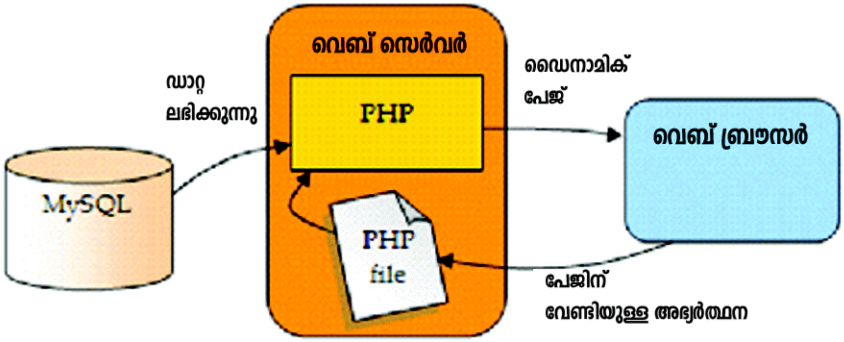
a. ഫോം സമർപ്പിച്ചതിന് ശേഷം ബ്രൗസറിലെ അഡ്രസ് ബാറിലെ URL എടുത്തെഴുതുക

b. ഫോം സമർപ്പിച്ചതിനുശേഷം ബുക്മാർക് ചെയ്യുക. ബുക്മാർക് ഉപയോഗിച്ച് ഫോം തിരികെ എടുക്കുക.

get -നു പകരം post ഉപയോഗിച്ച് മേല്പറഞ്ഞ പ്രവർത്തനങ്ങൾ ആവർത്തിക്കുക

10.8 PHP-യെ ഡാറ്റാബേസുമായി (MySQL) ബന്ധിപ്പിക്കുക (Connecting PHP to Database)

സ്റ്റാറ്റിക് വെബ് പേജുകൾ (നമ്മൾ നിർമ്മിക്കുന്ന സാധാരണ പേജുകൾ) എപ്പോഴും ഒരുപോലെ ആയിരിക്കും. നാം ഒരു പുതിയ പേജ് ലോഡ് ചെയ്യുന്നതു വരെ അതിന്റെ ഉള്ളടക്കം മാറുന്നില്ല. ഡൈനാമിക്ക് പേജുകൾ നേരെ വിപരീതമാണ്, അവ ലോഡ് ചെയ്യുമ്പോഴെല്ലാം ഉള്ളടക്കം മാറുന്നു. കൂടാതെ ഉപയോക്താക്കൾ എന്തു ചെയ്യുന്നുവെന്നതിനെ അടിസ്ഥാനമാക്കി (ചില ബട്ടണിൽ, വാക്കുകളിൽ അല്ലെങ്കിൽ ഇമേജിൽ ക്ലിക്ക് ചെയ്യുക വഴി) അവയുടെ ഉള്ളടക്കം മാറ്റാൻ കഴിയും. ആരെങ്കിലും വിവരങ്ങൾ ആവശ്യപ്പെടുമ്പോഴെല്ലാം ഡൈനാമിക് വെബ് സൈറ്റ് ഓരോ തവണയും ഒരു പുതിയ വെബ് പേജ് സൃഷ്ടിക്കുന്നു. അഭ്യർത്ഥനയുടെ അടിസ്ഥാനത്തിൽ ഒരു വെബ് പേജ് ടെംപ്ലേറ്റിലേക്ക് ഡാറ്റാബേസിൽ നിന്ന് വിവരങ്ങൾ എടുത്തുകൊണ്ട് വെബ് സെർവറിൽ ഒരു വെബ് പേജ് നിർമ്മിക്കപ്പെടുന്നു. ഇത് ഒരു HTML പേജായി ബ്രൗസറിലേക്ക് അയക്കുന്നു. പ്രോഗ്രാമർ ഒരു സെർവർ സൈഡ് സ്ക്രിപ്റ്റിംഗ് ഭാഷ ഉപയോഗിച്ച് എഴുതിയ നിർദ്ദേശങ്ങൾ പാലിച്ചുകൊണ്ടാണ് വെബ് സെർവർ ഈ വെബ് പേജ് സൃഷ്ടിക്കുന്നത്.



ചിത്രം 10.5: ബ്രൗസർ,PHP, ഡാറ്റാബേസ് ഇവ തമ്മിലുള്ള സമ്പർക്കം

ഒരു ഡാറ്റാബേസ് ഉപയോഗപ്പെടുത്തിയ "kerala results.nic.in" വെബ്സൈറ്റിന്റെ ഒരു ഉദാഹരണം മുൻപ് ചർച്ച ചെയ്തിട്ടുണ്ട് (ചിത്രം 10.5 കാണുക). ഹയർ സെക്കണ്ടറി പരീക്ഷ ഫലങ്ങൾ പ്രസിദ്ധീകരിക്കുന്ന പല വെബ് സൈറ്റുകളിൽ ഒന്നാണ് ഇത്. ഈ സൈറ്റിന്റെ സെർവറിൽ ഒരു അക്കാദമിക വർഷം പരീക്ഷയിൽ പങ്കെടുത്ത എല്ലാ വിദ്യാർഥികളുടെയും പരീക്ഷ ഫലങ്ങൾ ശേഖരിക്കുന്ന ഒരു ഡാറ്റാബേസ് അടങ്ങിയിരിക്കുന്നു. ഒരു ഫോം (ക്ലയന്റ് ഭാഗം) വഴി റജിസ്റ്റർ നമ്പർ സമർപ്പിക്കുമ്പോൾ, അത് വെബ് സെർവറിൽ എത്തുന്നു. വെബ് സെർവർ ഇത് PHP പ്രോഗ്രാമിന് കൈമാറുന്നു. PHP പ്രോഗ്രാം SQL പ്രസ്താവന വഴി രജിസ്ടർ നമ്പരുമായി ബന്ധപ്പെട്ട ഡാറ്റക്കുവേണ്ടി ഡാറ്റാബേസ് സെർവറിനോട് അഭ്യർഥിക്കുന്നു (SELECT * FROM marks WHERE regno= 2056789 എന്നത് പോലെ). ഡാറ്റാബേസിൽ നിന്നും ഈ രജിസ്ടർ നമ്പറിനായി ഡാറ്റാബേസ് സെർവർ തിരയുകയും അനുബന്ധ ഡാറ്റ (പരീക്ഷാ ഫലം) തിരികെ PHP-ലേക്ക് നൽകുകയും ചെയ്യുന്നു. PHP പ്രോഗ്രാം, HTML ഉപയോഗിച്ച് ഉചിതമായി ഫോർമാറ്റ് ചെയ്ത വെബ് പേജ് തയ്യാറാക്കുകയും അതിൽ തിരിച്ചുകിട്ടിയ വിവരങ്ങൾ ഉൾക്കൊള്ളിക്കുകയും ചെയ്യുന്നു. പുതുതായി സൃഷ്ടിച്ച വെബ് പേജ് വെബ് സെർവർ ക്ലയന്റിനു തിരിച്ചു നൽകുന്നു.

മുൻ അധ്യായങ്ങളിലെ (ഡാറ്റാബേസ് നിയന്ത്രണ സംവിധാനം, സ്ട്രക്ച്ചേർഡ് ക്വറി ലാൻഗ്വേജ്) ചർച്ചകളിൽ നിന്നും, SQL പ്രസ്താവനകൾ ഉപയോഗിച്ച് ഡാറ്റാബേസുമായി എങ്ങനെ ഇടപെടണമെന്ന് നമുക്ക് അറിയാം. ഈ ഭാഗത്ത്, ഒരു PHP പ്രോഗ്രാം ഉപയോഗിച്ച് അതെല്ലാം ചെയ്യാനും, കൂടാതെ വീണ്ടെടുക്കപ്പെട്ട ഡാറ്റ HTML-ലുമായി കൂട്ടിച്ചേർത്ത് ഡൈനാമിക് വെബ് പേജുകൾ സൃഷ്ടിക്കാനും നമ്മൾ പഠിക്കും.

10.8.1 MySQL ഡാറ്റാബേസുമായി ബന്ധം സ്ഥാപിക്കുന്നു (Establishing connection to MySQL database)

ഒരു PHP പ്രോഗ്രാമിന് ഡാറ്റാബേസിൽ നിന്നും ഡാറ്റ എടുക്കുന്നതിന്, ആദ്യം പ്രോഗ്രാമും ഡാറ്റാബേസ് സർവറുമായി (ഇവിടെ MySQL) ബന്ധം സ്ഥാപിക്കുകയും ശേഷം ഉപയോഗിക്കേണ്ട ഡാറ്റാബേസ് ഏതാണെന്ന് വ്യക്തമാക്കുകയും വേണം. ഇതിനുശേഷം മാത്രമേ PHP പ്രോഗ്രാമിന് SQL ക്വറികൾ (Queries) ഡാറ്റാബേസിന് നൽകാൻ സാധിക്കുകയുള്ളൂ.

MySQL ഡാറ്റാബേസുമായി ബന്ധം സ്ഥാപിക്കുന്നത് മൂന്ന് ഘട്ടങ്ങളിലൂടെയാണ്:

1. MySQL-മായി ബന്ധം സ്ഥാപിക്കുക

2. നമ്മൾ ഉപയോഗിക്കാൻ ആഗ്രഹിക്കുന്ന ഡാറ്റാബേസ് ഏതെന്ന് വ്യക്തമാക്കുക.
3. ഡാറ്റ വീണ്ടെടുക്കുക അല്ലെങ്കിൽ ഡാറ്റാബേസിലേക്ക് ഡാറ്റ ചേർക്കുക

ഘട്ടം 1 – MySQL-മായി ബന്ധം സ്ഥാപിക്കുക (Open a connection to MySQL)

PHP-യും MySQL ഡാറ്റാബേസ് സെർവറും തമ്മിൽ ഒരു ബന്ധം സ്ഥാപിക്കാൻ MySQL _connect () ഫംഗ്ഷൻ ഉപയോഗിക്കുന്നു. ഇത് മൂന്ന് സ്ട്രിംഗ് ആർഗ്യുമെന്റുകൾ, സ്വീകരിക്കുന്നു. ഹോസ്റ്റ് / സെർവർ പേര്, ഉപയോക്തൃനാമം (User name), പാസ്സ്‌വേർഡ്. നിർദ്ദിഷ്ട MySQL സെർവറിലേക്ക് വിജയകരമായി ബന്ധം സ്ഥാപിച്ച് കഴിയുമ്പോൾ ഫംഗ്ഷൻ ഒരു ലിങ്ക് തിരികെ നൽകുന്നു. എന്തെങ്കിലും പിഴവ് സംഭവിച്ചാൽ NULL എന്ന് ലഭിക്കുന്നു.

```
$link_id = mysql_connect("local host", "User ID", "Password")
```

Program 10.12 MySQL-ലേക്കുള്ള ബന്ധം വിശദീകരിക്കുന്ന സ്ക്രിപ്റ്റ് Connect.PHP

```
<?php
 $user_name = "root"; //Initialising variables
 $password = "root";
 $server = "localhost";
 //Connecting to MySQL database server
 $db_handle = mysql_connect($server,$user_name,$password);
 if (!$db_handle) {
     die('Could not connect: ' . mysql_error());
 }
 echo 'Connected successfully';
 mysql_close($db_handle);
?>
```

പ്രോഗ്രാം 10.12 ൽ, ആദ്യത്തെ മൂന്നു വരികൾ വേരിയബിളുകൾ സജ്ജമാക്കുന്നതിനുള്ള താണ്. ഈ ഉദാഹരണത്തിൽ ഉപയോക്തൃ നാമവും പാസ്സ്‌വേർഡും "root" എന്നാകുന്നു.

```
$user_name="root";
$password="root";
$server="localhost";
```

MySQL മായി ബന്ധം സ്ഥാപിക്കുന്ന നിർദേശമാണ് ചുവടെ ചേർക്കുന്നത്

```
mysql_connect($server, $user_name, $password);
```

mysql_connect ലേക്കുള്ള പരാമീറ്ററുകൾ നേരിട്ട് നൽകാനും സാധിക്കും. പ്രസ്താവന ശ്രദ്ധിക്കുക

```
mysql_connect( 'localhost', 'root', 'root' );
```

നമുക്ക് ചെയ്യാം

തെറ്റായ പാസ്സ്‌വേർഡ് ഉപയോഗിച്ച് മുകളിലുള്ള ഉദാഹരണം പ്രവർത്തിപ്പിച്ച് ഔട്ട്പുട്ട് പരിശോധിക്കുക

ഘട്ടം 2 – നിങ്ങൾക്ക് ഉപയോഗിക്കേണ്ട ഡാറ്റാബേസിന്റെ പേര് വ്യക്തമാക്കുക (Specify the database you want to open)

MySQL ഡാറ്റാബേസ് സെർവറിൽ ധാരാളം ഡാറ്റാബേസുകൾ ഉണ്ടായിരിക്കാം. mysql_select_db( ) ഫങ്ഷൻ ഉപയോഗിച്ച് അവയിൽ നിന്നും ഒരു പ്രത്യേക MySQL ഡാറ്റാബേസ് തിരഞ്ഞെടുക്കാം. ഡാറ്റാബേസിന്റെ പേര്, ഒരു ലിങ്ക് ഐഡന്റിഫയർ എന്നിവ സ്ട്രിംഗ് രൂപത്തിലുള്ള ആർഗ്യുമെന്റായി ഇത് സ്വീകരിക്കുന്നു. ഡാറ്റാബേസ് കണ്ടെത്തുന്നുവെങ്കിൽ, True ഉം അല്ലാത്തപക്ഷം False ഉം അത് തിരികെ നൽകുന്നു.

```
mysql_select_db($database);
```

Program 10.13 ഡാറ്റാബേസിലുള്ള ബന്ധം വിശദീകരിക്കുന്ന സ്ക്രിപ്റ്റ് (DBConnect.PHP)

```
<?php
 $user_name = "root";
 $password = "root";
 $server = "localhost";
 $database = "mysql"; //initializing variable
 $db_handle = mysql_connect($server,$user_name,$password);
 if (!$db_handle) {
     die('Could not connect: ' . mysql_error());
 }
 echo 'Connected successfully';
 //Opening the database mysql
 $db_found = mysql_select_db($database);
 if ($db_found) {
     echo "Database Found";
 }
 else {
     echo "Database NOT Found";
 }
 mysql_close($db_handle);
?>
```

ഇവിടെ ഒരു പ്രസ്താവന ഉപയോഗിച്ച് $database എന്ന വേരിയബിളിലേക്ക് നമ്മുടെ ഡാറ്റാബേസിന്റെ പേര് നൽകിയിരിക്കുന്നു.

```
$database = "mysql";
```

ചുവടെ ചേർത്തിരിക്കുന്ന പ്രസ്താവന ഉപയോഗിച്ച് ഡാറ്റാബേസ് തുറക്കുന്നു

```
$db_found = mysql_select_db($database);
```

ഇപ്പോൾ സജീവമായ ബന്ധമാണ് ഉപയോഗിക്കുന്നുവെന്ന് ഇവിടെ നാം അനുമാനിക്കുന്നു. മറ്റൊരു ബന്ധം ഉപയോഗിക്കേണ്ടി വരുമ്പോൾ ഡാറ്റാബേസ് തുറക്കുന്നതിനു താഴെ പറയുന്ന പ്രസ്താവനയും ഉപയോഗിക്കാം. ഇവിടെ രണ്ടാമത്തെ പരാമീറ്ററായി ഉപയോഗിക്കുന്നത് ബന്ധത്തിന്റെ റിസോഴ്സ് ലിങ്ക് ഐഡന്റിഫയർ ആയിരിക്കും.

```
$db_found = mysql_select_db($database, $db_handle);
```

ഇപ്പോൾ നമ്മൾ സെർവറിലേക്ക് ബന്ധം സ്ഥാപിക്കുകയും ആവശ്യമായ ഡാറ്റാബേസ് തിരഞ്ഞെടുക്കുകയും ചെയ്തു. ഇനി ഡാറ്റാബേസിനുള്ളിൽ ഡാറ്റ തിരയൽ ആരംഭിക്കുകയും ഡൈനാമിക് വെബ് പേജുകൾ സൃഷ്ടിക്കുകയും വേണം.

ഘട്ടം 3 - ഡാറ്റാബേസിൽ നിന്ന് ഡാറ്റ വീണ്ടെടുക്കുക (Reading data from database)

ഡാറ്റാബേസിൽ നിന്ന് ഡാറ്റ വീണ്ടെടുക്കുന്നത് രണ്ട് ഘട്ടങ്ങളിലാണ്

a. mysql_query( ) ഫങ്ഷൻ ഉപയോഗിച്ച് ഡാറ്റാബേസിൽ SQL ക്വറി പ്രവർത്തിപ്പിക്കുക.

b. mysql_fetch_array( ) ഫങ്ഷൻ ഉപയോഗിച്ച് നിരകളെ (Rows) ഒരു അറെയിലേക്ക് മാറ്റുക.

ഈ അറെ ഉപയോഗിച്ചാണ് PHP പ്രോഗ്രാം ഡാറ്റ കൈകാര്യം ചെയ്യുന്നത്.

a. mysql_query()

ഡാറ്റാബേസിൽ SQL ക്വറി പ്രവർത്തിപ്പിക്കുന്നത് mysql_query() ഫങ്ഷൻ ഉപയോഗിച്ചാണ്. ഇത് ഡാറ്റാബേസിലേക്ക് ക്വറി അയക്കുകയും പ്രവർത്തനം ഫലപ്രദമായാൽ ഒരു റിസോർസ് ഇനം ക്വറി ഹാൻഡിൽ തിരികെ ലഭിക്കുകയും ചെയ്യുന്നു.

```
$result_set = mysql_query(query, connection)
```

- SELECT, SHOW, DESCRIBE അല്ലെങ്കിൽ EXPLAIN എന്നീ SQL പ്രസ്താവനകൾക്ക്, mysql_query () വിജയകരമായാൽ ഒരു റിസോഴ്സ് ഇനം ക്വറി ഹാൻഡിൽ തിരികെ നൽകുന്നു. തെറ്റ് സംഭവിച്ചാൽ FALSE ലഭിക്കുന്നു.

വിദ്യാർഥിയുടെ രജിസ്റ്റർ നമ്പർ സ്വീകരിക്കുന്നതിനുള്ള ഫോ

- മറ്റ് തരം SQL പ്രസ്താവനകളായ, UPDATE, DELETE, DROP തുടങ്ങിയവയ്ക്കായി, mysql_query( ) വിജയകരമായാൽ TRUE ഉം അല്ലാത്തപക്ഷം FALSE ഉം തരുന്നു.
- ക്വറിയിൽ സൂചിപ്പിച്ച ടേബിളുകൾ ഉപയോഗിക്കുന്നതിനു ഉപയോക്താവിന് അനുമതിയില്ലെങ്കിൽ mysql_query( ) പരാജയപ്പെടുകയും FALSE തിരികെ ലഭിക്കുകയും ചെയ്യും
- ഡാറ്റ വീണ്ടെടുക്കുന്നതിന് തിരികെ ലഭിച്ച ഫലം mysql_ fetch_ array() യിലേക്ക് കൈമാറുന്നു.

ഒരു SELECT പ്രസ്താവന എത്ര വരികൾ തിരികെ നൽകിയെന്ന് കണ്ടുപിടിക്കാൻ mysql_num_rows() ഉപയോഗിക്കുന്നു. DELETE, INSERT, REPLACE അല്ലെങ്കിൽ UPDATE പ്രസ്താവന എത്ര വരികളെ ബാധിച്ചു എന്ന് കണ്ടുപിടിക്കാൻ mysql_affected_rows() ഉപയോഗിക്കുന്നു.

b. mysql_fetch_array()

MySQL റിസൾട്ട് ഹാൻഡിലിൽ നിന്നും mysql_fetch_array() ഉപയോഗിച്ച് ഡാറ്റ വരികൾ അറെയായി ലഭ്യമാക്കുന്നു. mysql_query() ഫങ്ഷൻ നൽകിയ റിസൾട്ട് ഹാൻഡിൽ ഒരു

പരാമീറ്ററായി സ്വീകരിക്കുകയും വീണ്ടെടുത്ത നിരകളെ അറെ രൂപത്തിൽ തിരികെ തരുകയും ചെയ്യുന്നു. ഈ അറെ ഒരു അസ്സോസിയേറ്റീവ് അറെയോ അഥവാ ന്യൂമെറിക് അറെയോ അതുമല്ലെങ്കിൽ ഇവ രണ്ടുമോ ആകാം. ഇനി ഡാറ്റ നിരകൾ ഒന്നുംതന്നെ വീണ്ടെടുക്കാൻ അവശേഷിക്കുന്നില്ലെങ്കിൽ False തിരികെ നൽകുന്നു.

```
$result_set = mysql_query(query, connection);
$fetched_row = mysql_fetch_array($result_set);
```

നിലവിലില്ലാത്ത ഡാറ്റാബേസ് നാമം ഉപയോഗിച്ച് പ്രോഗ്രാം 10.13 പ്രവർത്തിപ്പിച്ച് ലഭിക്കുന്ന ഔട്ട്പുട്ട് ശ്രദ്ധിക്കുക

നമുക്ക് ചെയ്യാം

താഴെ കൊടുത്തിരിക്കുന്ന രണ്ട് ഉദാഹരണങ്ങളുടെ സഹായത്തോടെ ഈ രണ്ട് ഫങ്ഷനുകളുടെ പ്രവർത്തനത്തെ വിശദീകരിക്കാനാകും.

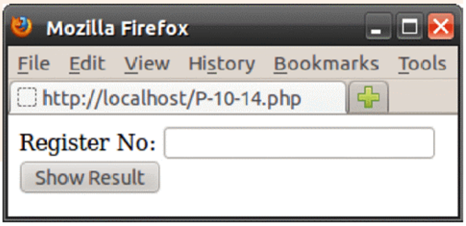
താഴെ പറയുന്ന പ്രോഗ്രാം (പ്രോഗ്രാം 10.14) ഫോം ഉപയോഗിച്ച് ഒരു രജിസ്റ്റർ നമ്പർ സ്വീകരിച്ച് showmarks.phpലേക്ക് സമർപ്പിക്കുന്നു. MySQL ഡാറ്റാബേസിലെ മാർക്കുകളുടെ പട്ടിക PHP പ്രോഗ്രാം തിരയുകയും, അവ ലഭിച്ചാൽ ഈ ഡാറ്റ അടങ്ങിയ വെബ് പേജ് സൃഷ്ടിക്കുകയും ചെയ്യുന്നു.

കുറിപ്പ്: പ്രോഗ്രാമുകൾ 10.14-ഉം 10.15-ഉം ചെയ്യുന്നതിനു മുൻപ്, "mysql" എന്നുള്ള ഡാറ്റാബേസിൽ താഴെ കൊടുത്തിരിക്കുന്ന ഘടനയിൽ "Marks" എന്നു പേരുള്ള ഒരു പട്ടിക (Table) ഉണ്ടാക്കുക. അതിൽ കുറച്ച് ഡാറ്റ നൽകുക (ഗ്രൂപ്പ് നാമം കൊമേഴ്സ്/സയൻസ് ആയിരിക്കണം).

ഫീൽഡ് നാമം	ഡാറ്റ ഇനം
regno	Number
sname	Char(20)
sgroup	Char(20)
mark1	Number
mark2	Number
mark3	Number

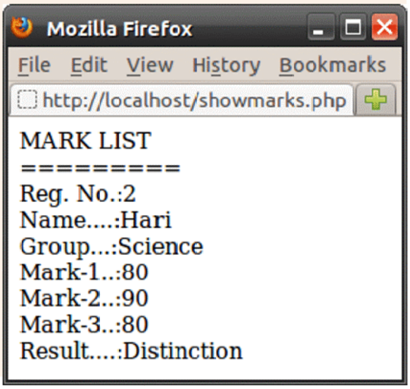
Program 10.14(a) വിദ്യാർഥിയുടെ രജിസ്റ്റർ നമ്പർ സ്വീകരിക്കുന്നതിനുള്ള ഫോം HTML

```
<HTML>
<BODY>
  <FORM Method="post" Action="showmarks.php">
    Register No:
    <INPUT Type="text" name="regno"> <BR>
    <INPUT Type="submit" Value="Show Result">
  </FORM>
</BODY>
</HTML>
```



Program 10.14(b) ഡാറ്റാബേസ് തിരയുന്നതിനും ഫലം പ്രദർശിപ്പിക്കുന്നതിനുമുള്ള സ്ക്രിപ്റ്റ് showmarks.php

```
<?php
$user_name = "root";
$password = "root";
$server = "localhost";
$database = "mysql";
$regno = $_POST['regno'];
$db_handle = mysql_connect($server,$user_name,$password);
if (!$db_handle) {
   die("Could not connect: " . mysql_error());}
$db_found = mysql_select_db($database, $db_handle);
if ($db_found) { //Retrieving data from database
   $SQL = "SELECT * FROM marks WHERE regno = '".$regno."'";
   $result_set = mysql_query($SQL);
   $record = mysql_fetch_array($result_set);
   echo "<BR>MARK LIST";
   echo "<BR>=========";
   echo "<BR>Reg. No.:".$record['regno'];
   echo "<BR>Name....:".$record['sname'];
   echo "<BR>Group...:".$record['sgroup'];
   echo "<BR>Mark-1..:".$record['mark1'];
   echo "<BR>Mark-2..:".$record['mark2'];
   echo "<BR>Mark-3..:".$record['mark3'];
   echo "<BR>Result..:". compute_result($record['mark1'],
                        $record['mark2'], $record['mark3']);
 }
else
 {echo "<BR>Database NOT Found";}
mysql_close($db_handle);
//Function to calculate result
function compute_result($m1, $m2, $m3)
{  $tmarks = $m1 + $m2 + $m3;
   if (($m1<40) || ($m1<40) || ($m1<40))
        $result = "Failed";
   elseif ($tmarks < 150)
        $result = "Passed";
   elseif ($tmarks < 180)
        $result = "Second Class";
   elseif ($tmarks < 225)
          $result = "First Class";
   else
        $result = "Distinction";
return $result; }
?>
```

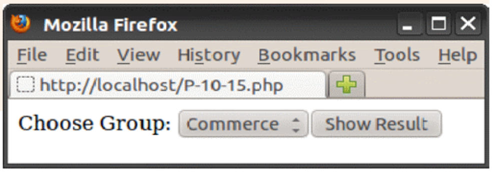


ദീർഘചതുരത്തിൽ ഉൾപ്പെടുത്തിയ കോഡ് നിരീക്ഷിക്കുക. ഡാറ്റാബേസിൽ നിന്ന് ഡാറ്റ ലഭിക്കുന്നതിന് ഈ ഭാഗം ഉപയോഗിക്കുന്നു. നമുക്ക് പ്രവർത്തിപ്പിക്കേണ്ട ക്വറി പ്രസ്താവന $SQL എന്ന വേരിയബിളിലേക്ക് പ്രാരംഭവിലയായി ഇവിടെ കൊടുത്തിരിക്കുന്നു. mysql_query ($SQL) എന്ന അടുത്ത പ്രസ്താവന പ്രവർത്തിപ്പിക്കുകയും $result_set എന്ന വേരിയബിളിൽ സൂക്ഷിച്ചിരിക്കുന്ന ഫലം തിരികെ നൽകുകയും ചെയ്യുന്നു. $result_set, mysql_fetch_array ($ result_set) ലേക്ക് കൈമാറുന്നു. അത് $result_set നൽകിയ ഡാറ്റ ഉപയോഗിച്ച് ഒരു അസോസിയേറ്റ് അറെ ഉണ്ടാക്കി തിരികെ നൽകുന്നു. ഫങ്ഷൻ തിരിച്ചു നൽകിയ അറെ $record എന്ന വേരിയബിളിൽ സൂക്ഷിക്കുന്നു. ഇതിൽ നിന്നും വിവരങ്ങൾ എടുക്കുവാൻ ഫീൽഡ് നാമം ഇൻഡക്സ് ആയി നൽകണം, അതായത് $record['sname'] എന്ന രീതിയിൽ.

നമുക്ക് മറ്റൊരു പ്രോഗ്രാം നോക്കാം. കോംബോ ബോക്സ് ഉപയോഗിച്ച് ഒരു ഗ്രൂപ്പ് (സയൻസ്/കോമേഴ്സ്) തിരഞ്ഞെടുക്കുന്നതിനുള്ള പ്രോഗ്രാം (പ്രോഗ്രാം 10.15). ആ ഗ്രൂപ്പിൽ പഠിക്കുന്ന വിദ്യാർത്ഥികളുടെ ഫലം ഈ പ്രോഗ്രാം ഒരു പട്ടികരൂപത്തിൽ പ്രദർശിപ്പിക്കുകയും ചെയ്യുന്നു.

Program 10.15(a) ഗ്രൂപ്പ് സ്വീകരിക്കുന്നതിനുള്ള ഫോം HTML

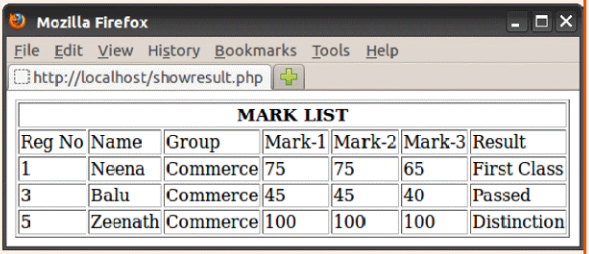
```
<HTML>
<BODY>
   <FORM Method="post" Action="showresult.php">
   Choose Group:
   <SELECT Name = "cgroup">
      <OPTION>Science</OPTION>
      <OPTION>Commerce</OPTION>
   <INPUT Type="submit" Value= "Show Result">
   </FORM>
</BODY>
</HTML>
```



Program 10.15(b) ഡാറ്റാബേസ് തിരയുന്നതിനും ഫലം പ്രദർശിപ്പിക്കുന്നതിനുമുള്ള സ്ക്രിപ്റ്റ് showresult.PHP

```
<?php
   $user_name = "root";
   $password = "root";
   $server = "localhost";
   $database = "mysql";
   $db_handle = mysql_connect($server,$user_name,$password);
   if (!$db_handle) {
       die('Could not connect: ' . mysql_error());
   }
   $db_found = mysql_select_db($database, $db_handle);
   if ($db_found)
   {
```

```
    $SQL = "SELECT * FROM marks WHERE sgroup =
                                       '".$_POST[cgroup]."'";
    $select_query = mysql_query($SQL);
    echo "<TABLE Border = 1">;
    echo "<TR>";
    echo "<TH Colspan="7"><CENTER> MARK LIST </CENTER> </TH>";
    echo "</TR>";
    echo "<TR>";
    echo "<TD>Reg No</TD>";
    echo "<TD>Name</TD>";
    echo "<TD>Group</TD>";
    echo "<TD>Mark-1</TD>";
    echo "<TD>Mark-2</TD>";
    echo "<TD>Mark-3</TD>";
    echo "<TD>Result</TD>";
    echo "</TR>";
    while($record = mysql_fetch_array($select_query)){
          echo "<TR>";
          echo "<TD>" . $record['regno'] . "</TD>";
          echo "<TD>" . $record['sname'] . "</TD>";
          echo "<TD>" . $record['sgroup'] . "</TD>";
          echo "<TD>" . $record['mark1'] . "</TD>";
          echo "<TD>" . $record['mark2'] . "</TD>";
          echo "<TD>" . $record['mark3'] . "</TD>";
          echo "<TD>" . compute_result($record['mark1'],
                $record['mark2'], $record['mark3']). "</TD>";
          echo "</TR>";
    }
    echo "</TABLE>";
}
else {
    echo '<BR>Database NOT Found';
}
mysql_close($db_handle);
//Function to calculate result
function compute_result($m1, $m2, $m3)
{  $tmarks = $m1 + $m2 + $m3;
   if (($m1<40) || ($m1<40) || ($m1<40))
          $result ="Failed";
   elseif ($tmarks < 150)
          $result ="Passed";
   elseif ($tmarks < 180)
          $result ="Second Class";
   elseif ($tmarks < 225)
          $result ="First Class";
   else
          $result = "Distinction";
return $result;
} ?>
```



ദീർഘചതുരത്തിൽ ഉൾപ്പെടുത്തിയ കോഡ് നിരീക്ഷിക്കുക. ഡാറ്റാബേസിൽ നിന്നും ഒരു പട്ടിക രൂപത്തിൽ ഡാറ്റ പ്രദർശിപ്പിക്കുന്നത് ഈ ഭാഗമാണ്. ഡാറ്റ ഒരു പട്ടിക രൂപത്തിൽ പ്രദർശിപ്പിക്കാൻ, mysql_fetch_array($result_set) മുഖേന ലഭിച്ച ഡാറ്റ <TABLE> ടാഗ് ഉപയോഗിച്ച് ക്രമീകരിച്ചിരിക്കുന്നു. while ലൂപിലെ ഓരോ ആവർത്തനത്തിലും, <TR></TR> ടാഗ് ഉപയോഗിച്ച് പട്ടികയിൽ ഒരു പുതിയ വരി ചേർക്കപ്പെടുകയും, അസോസിയേറ്റ് അറെയിലുള്ള ഡാറ്റ <TD></TD> ടാഗ് ഉപയോഗിച്ച് ശരിയായ കോളത്തിൽ സ്ഥാപിക്കുകയും ചെയ്യും. ലൂപ്പിന്റെ എല്ലാ ആവർത്തനങ്ങളിലും mysql_fetch_array($result_set) പുതിയ ഡാറ്റ $result-ൽ ലോഡ് ചെയ്യുന്നു. ഈ പ്രവർത്തനം പൂർത്തിയാക്കിയാൽ ഫങ്ഷൻ false തിരികെ നൽകി ലൂപ്പ് പ്രവർത്തനം അവസാനിപ്പിക്കും.

ഘട്ടം 4 - ബന്ധം വിച്ഛേദിക്കുക (Close the connection)

ഡാറ്റാബേസുകളുടെ എല്ലാ പ്രവർത്തനങ്ങളും അവസാനിച്ചതിന് ശേഷം mysql_close() ഫങ്ഷൻ ഉപയോഗിച്ച് ഡാറ്റാബേസ് സെർവറുമായുള്ള ബന്ധം വിച്ഛേദിക്കുന്നു. PHP സ്ക്രിപ്റ്റ്, പ്രവർത്തനം അവസാനിപ്പിക്കുമ്പോൾ MySQL സെർവറിലേക്കുള്ള ലിങ്ക് സ്വയം വിച്ഛേദിച്ചിരിക്കും. PHP സ്ക്രിപ്റ്റ് അവസാനിക്കുന്നതിനു മുമ്പ് നമുക്ക് ബന്ധം വിച്ഛേദിക്കണമെങ്കിൽ, ഈ ഫങ്ഷനോടൊപ്പം പരാമീറ്ററായി ലിങ്ക് ഐഡന്റിഫയർ ഉപയോഗിക്കണം.

```
mysql_close($db_handle);
```

വിജയകരമായാൽ True-ഉം അല്ലെങ്കിൽ False-ഉം ഈ ഫങ്ഷൻ തിരികെ തരുന്നു.

മുൻ ഉദാഹരണങ്ങളിലെ mysql_close($db_handle); പ്രസ്താവന നോക്കുക.

ഒരേസമയം ഒന്നിൽ കൂടുതൽ MySQL ബന്ധങ്ങൾ പ്രശ്നങ്ങൾ സൃഷ്ടിക്കാം. ആയതിനാൽ ക്വറി പ്രവർത്തിപ്പിച്ചു കഴിഞ്ഞാൽ ഡാറ്റാബേസ് ബന്ധം വിച്ഛേദിക്കുന്നതാണ് അഭികാമ്യം.

10.8.2 PHP ഉപയോഗിച്ച് ഒരു പുതിയ ടേബിൾ നിർമ്മിക്കുന്നു (Creating a new table using PHP)

ഒരു പുതിയ പട്ടിക സൃഷ്ടിക്കുന്നതിനായി, mysql_query() ഫങ്ഷൻ ഉപയോഗിച്ച് ഒരു SQL പ്രസ്താവന തയ്യാറാക്കി പ്രവർത്തിപ്പിക്കുക. അതിനുള്ള ഉദാഹരണം ഇതാണ്.

```
$SQL = "CREATE TABLE marks (regno number,tname char(20),
     m1 number,m2 number,m3 number,distinction char(20))";
$select_query = mysql_query($SQL);
```

10.8.3 PHP ഉപയോഗിച്ച് ഒരു ടേബിളിലേക്കു ഡാറ്റ ചേർക്കുന്നു (Inserting data into a table using PHP)

ടേബിളിലേക്കു ഡാറ്റ ചേർക്കാൻ നാം ഒരു SQL പ്രസ്താവന തയാറാക്കുകയും mysql_query() ഫങ്ഷൻ ഉപയോഗിച്ച് അത് പ്രവർത്തിപ്പിക്കുകയും വേണം. $sregno,

$sname, $sm1, $sm2, $sm3, $sresult എന്നീ വേരിയബിളുകളിൽ ടേബിളിലേക്ക് ചേർക്കേണ്ട ഡാറ്റ ഉണ്ടെന്നിരിക്കട്ടെ. ഇതിനാവശ്യമായ SQL പ്രസ്താവന ചുവടെ ചേർക്കുന്നു.

```
$SQL = "INSERT INTO marks VALUES (".$sregno.",'".$sname.
       "',".$sm1.",".$sm2.",".$sm3.",'".$sresult."')";
$select_query = mysql_query($SQL);
```

10.8.4 PHP ഉപയോഗിച്ച് ഒരു ടേബിളിലെ ഡാറ്റയിൽ മാറ്റം വരുത്തുന്നു (Updating data in a table using PHP)

ടേബിളിലെ ഡാറ്റ പുതുക്കുന്നതിനായി, നാം ഒരു SQL സ്റ്റേറ്റ്മെന്റ് തയ്യാറാക്കുകയും അത് mysql_query () ഉപയോഗിച്ച് പ്രവർത്തിപ്പിക്കുകയും വേണം. $sregno എന്ന വേരിയബിളിലെ രജിസ്റ്റർ നമ്പർ സൂചിപ്പിക്കുന്ന വിദ്യാർത്ഥിയുടെ മാർക്ക് '70' ആക്കി മാറ്റണം എന്ന് വിചാരിക്കുക. ഇതിനായി വേണ്ടുന്ന പ്രസ്താവന ചുവടെ നൽകുന്നു:

```
$SQL = "UPDATE marks SET m1=70 WHERE regno='".$sregno."'";
$select_query = mysql_query($SQL);
```

നമുക്ക് സംഗ്രഹിക്കാം

സെർവർ ഭാഗത്തെ സ്ക്രിപ്റ്റിംഗ് ഭാഷകളായ PHP, PERL, ASP.NET തുടങ്ങിയവയുടെ സഹായത്തോടെയാണ് ഡൈനാമിക് വെബ് പേജുകൾ നിർമ്മിക്കുന്നത്. സെർവർ ഭാഗത്തെ സ്ക്രിപ്റ്റിംഗ് ഭാഷകളിൽ ഏറ്റവും പ്രചാരമേറിയത് PHP-യാണ്. വിവിധ വെബ് സെർവറുകളിലും ഓപ്പറേറ്റിംഗ് സിസ്റ്റങ്ങളിലും PHP പ്രവർത്തിക്കും. HTML-ന്റെയും MySQL പോലെയുള്ളള ഡാറ്റാബേസ്കളുടേയും ന്റെയും സഹായത്തോടെ PHP-ക്ക് മികച്ച ഡൈനാമിക് വെബ് പേജുകൾ നിർമ്മിക്കുവാൻ സാധിക്കും. C++ പ്രോഗ്രാമിങ് ഭാഷയുടെ വാക്യഘടന അനുവർത്തിക്കുന്നതിനാൽ PHP പഠനം വളരെ എളുപ്പമാണ്.

നമുക്ക് പരിശീലിക്കാം

1. സംഖ്യ 5-ന്റെ ഗുണനപ്പട്ടിക പ്രദർശിപ്പിക്കുന്നതിനുള്ള ഒരു PHP പ്രോഗ്രാം എഴുതുക.
2. 100-നും 200-നും ഇടയിൽ വരുന്ന എല്ലാ പാലിൻഡ്രോം സഖ്യകളും പ്രിന്റ് ചെയ്യുന്നതിനുള്ള ഒരു PHP പ്രോഗ്രാം എഴുതുക.
3. 1000-നു താഴെ വരുന്ന എല്ലാ ആംസ്ട്രോങ് സംഖ്യകളും പ്രിന്റ് ചെയ്യുന്നതിനുള്ള ഒരു PHP പ്രോഗ്രാം എഴുതുക.
4. PHP-യും HTML-ഉം ഉപയോഗിച്ച് ചുവടെ നൽകിയിരിക്കുന്ന പട്ടിക പ്രദർശിപ്പിക്കുക. ഓരോ സെല്ലും അതാതു വരിയുടെയും നിരയുടെയും നമ്പർ പ്രദർശിപ്പിക്കണം. പട്ടിക നിർമ്മാണത്തിന് PHP-യിലെ ലൂപ്പ് പ്രസ്താവന ഉപയോഗിക്കേണ്ടതാണ്.

Column 1	Column 2	Column 3	Column 4	Column 5
row 1, col 1	row 1, col 2	row 1, col 3	row 1, col 4	row 1, col 5
row 2, col 1	row 2, col 2	row 2, col 3	row 2, col 4	row 2, col 5
row 3, col 1	row 3, col 2	row 3, col 3	row 3, col 4	row 3, col 5
row 4, col 1	row 4, col 2	row 4, col 3	row 4, col 4	row 4, col 5
row 5, col 1	row 5, col 2	row 5, col 3	row 5, col 4	row 5, col 5
row 6, col 1	row 6, col 2	row 6, col 3	row 6, col 4	row 6, col 5
row 7, col 1	row 7, col 2	row 7, col 3	row 7, col 4	row 7, col 5
row 8, col 1	row 8, col 2	row 8, col 3	row 8, col 4	row 8, col 5
row 9, col 1	row 9, col 2	row 9, col 3	row 9, col 4	row 9, col 5
row 10, col 1	row 10, col 2	row 10, col 3	row 10, col 4	row 10, col 5

5. മൂന്നു സംഖ്യകളിൽ ഏറ്റവും വലുത് കണ്ടുപിടിക്കുന്നതിനുള്ള ഒരു പ്രോഗ്രാം എഴുതുക.
6. Name, RollNo, Gender എന്നീ ഫീൽഡുകളും Submit Button-ഉം ഉള്ള ഒരു രജിസ്ട്രേഷൻ ഫോം നിർമ്മിക്കുക. ഈ വിശദാംശങ്ങൾ സ്വീകരിക്കുന്ന ഒരു PHP സ്ക്രിപ്റ്റ് എഴുതുകയും ഉപയോക്താവ് Submit ബട്ടൺ ക്ലിക്കുചെയ്യുമ്പോൾ ബ്രൗസറിൽ അത് പ്രദർശിപ്പിക്കുകയും ചെയ്യുക.
7. നൽകിയിരിക്കുന്ന പ്രശ്നങ്ങൾക്ക് അനുയോജ്യമായ PHP സ്ക്രിപ്റ്റ് എഴുതുക: ഒരു സ്ട്രിംഗ് സ്വീകരിക്കുന്നതിനുള്ള ഫോം രൂപകൽപന ചെയ്യുക. സ്വീകരിച്ച സ്ട്രിങ്ങിൽ എത്ര സ്വരാക്ഷരങ്ങൾ (a,e,i,o,u) ഉണ്ടെന്നു കണ്ടുപിടിക്കുന്ന ഫങ്ഷൻ എഴുതുക.
8. MySQL-ൽ ഒരു പട്ടികനിർമ്മിച്ച് കുറച്ച് ഡാറ്റ റെക്കോർഡുകൾ ചേർക്കുക. ഈ റെക്കോർഡുകൾ ബ്രൗസറിൽ പ്രദർശിപ്പിക്കുന്നതിനുള്ള ഒരു PHP സ്ക്രിപ്റ്റ് എഴുതുക.
9. MySQL-ൽ ഒരു പട്ടികനിർമ്മിച്ച് കുറച്ച് ഡാറ്റ റെക്കോർഡുകൾ ചേർക്കുക. ഫോം ഉപയോഗിച്ച് ഒരു വില സ്വീകരിക്കുന്നതിന് PHP സ്ക്രിപ്റ്റ് എഴുതുക, ഡാറ്റാബേസിൽ വില തിരയുകയും ബ്രൗസറിൽ റെക്കോർഡ് പ്രദർശിപ്പിക്കുകയും ചെയ്യുക.

നമുക്ക് വിലയിരുത്താം

1. ഔട്ട്പുട്ട് വിജയകരമായി ലഭിക്കുമ്പോൾ 1 തിരികെ തരുന്ന ഫങ്ഷൻ ഏതാണ്?

   a. echo() b. print() c. രണ്ടും d. ഇവയൊന്നുമല്ല

2. ഒന്നിൽ കൂടുതൽ പരാമീറ്ററുകൾ സ്വീകരിക്കുന്ന പ്രിന്റ് ഫങ്ഷൻ ഏതാണ്?

   a. echo() b. print() c. printf d. ഇവയെല്ലാം

3. പാസ്‌വേഡുകളോ മറ്റു തന്ത്രപ്രധാനമായ വിവരങ്ങളോ അയക്കുമ്പോൾ താഴെ നൽകിയിരിക്കുന്നവയിൽ ഉപയോഗിക്കരുതാത്തത് ഏത്?

   a. get b. post c. REQUEST d. NEXT

4. ഒരു സ്ട്രിംഗ് വേരിയബിളിലെ അക്ഷരങ്ങളുടെ എണ്ണം കണ്ടെത്തുന്ന ഫങ്ഷൻ ഏതാണ് ?

   a. count($variable) b. len($variable) c. strcount($variable) d. strlen($variable)

5. താഴെ കൊടുത്തിരിക്കുന്നതിൽ ഏതാണ് പ്രീ-ഡിഫൈൻഡ് (pre-defined) വേരിയബിൾ അല്ലാത്തത് ?

   a. $_GET b. $_ASK c. $_REQUEST d. $_POST

6. $_GET വേരിയബിൾ ഉപയോഗിച്ച് ഡാറ്റ സ്വീകരിക്കുമ്പോൾ ഡാറ്റ കാണാൻ കഴിയുന്നത്

   a. ആർക്കുമില്ല b. നിങ്ങൾക്ക് മാത്രം c. എല്ലാവർക്കും d. ചിലർക്കുമാത്രം

7. $_POST വേരിയബിൾ ഉപയോഗിച്ച് ഡാറ്റ സ്വീകരിക്കുമ്പോൾ ഡാറ്റ കാണാൻ കഴിയുന്നത്

   a. ആർക്കുമില്ല b. നിങ്ങൾക്ക് മാത്രം c. എല്ലാവർക്കും d. ചിലർക്കുമാത്രം

8. get, post എന്നീ രണ്ട് രീതികൾ ഉപയോഗിച്ച് ഫോമിലെ ഡാറ്റ അയക്കുമ്പോൾ താഴെ നൽകിയിക്കുന്നതിൽ ഏത് വേരിയബിൾ ഉപയോഗിച്ചാണ് ഡാറ്റ സ്വീകരിക്കുന്നത്?

   a. $BOTH b.$_BOTH c. $REQUEST d. $_REQUEST

9. താഴെ നൽകിയിരിക്കുന്ന പരാമീറ്ററുകളിൽ mysql_connect()-ൽ ഉപയോഗിക്കാത്ത പാരാമീറ്റർ ഏത് ?

   a. ഡാറ്റാബേസ് ഹോസ്റ്റ് b. യൂസർ ഐഡി c. പാസ്‌വേർഡ് d. ഡാറ്റാബേസ് നാമം

10. താഴെ നൽകിയിരിക്കുന്ന വിവിധ ഘട്ടങ്ങളെ ശരിയായ രീതിയിൽ ക്രമപ്പെടുത്തുക:

    i. MySQL സെർവറിലേക്കുള്ള കണക്ഷൻ തുറക്കുന്നു.

    ii. SQL ക്വറി പ്രവർത്തിപ്പിക്കുന്നു

    iii. ക്വറി ഉപയോഗിച്ച് ഡാറ്റ എടുക്കുന്നു

    iv. ഡാറ്റാബേസ് തിരഞ്ഞെടുക്കുന്നു

    v. കണക്ഷൻ വിച്ഛേദിക്കുന്നു

    a. 1,4,2,3,5 b. 4,1,2,3,5 c. 1,5,4,2,1 d. 4,1,3,2,5

11. എന്താണ് PHP?
12. PHP-യും ജാവാസ്ക്രിപ്റ്റും തമ്മിലുള്ള വ്യത്യാസം എന്ത്?
13. ഒരു PHP പ്രോഗ്രാം പ്രവർത്തിപ്പിക്കുന്നതിനുള്ള വിവിധ ഘട്ടങ്ങൾ ഏതെല്ലാം?
14. PHP-യിൽ echo-യുടെ ഉപയോഗം എന്ത്?
15. PHP-യിൽ print-ന്റെ ഉപയോഗം എന്ത്?
16. 'echo', 'print'എന്നീ പ്രസ്താവനകൾ തമ്മിലുള്ള വ്യത്യാസങ്ങൾ എന്തെല്ലാം?
17. ക്ലയന്റ് ഭാഗത്തെ സ്ക്രിപ്റ്റിംഗ് ഭാഷയായ ജാവാസ്ക്രിപ്റ്റിനെക്കാൾ PHP മികച്ചതായത് എന്തുകൊണ്ട്?
18. ഒരു PHP പ്രസ്താവന അവസാനിപ്പിക്കുവാൻ ഏത് അക്ഷരമാണ് ഉപയോഗിക്കുന്നത്? ഈ അക്ഷരം ഒഴിവാക്കിയാലും തെറ്റാണെന്ന സന്ദേശം വരാത്തത് ഏത് അവസരത്തിലാണ്?

19. എസ്കേപ് സീക്വൻസ് എന്നാൽ എന്ത്? സാധാരണയായി ഉപയോഗിച്ച് വരുന്ന മൂന്ന് എസ്കേപ് സീക്വൻസുകൾ എഴുതുക.

20. ചുവടെ നൽകിയിരിക്കുന്ന PHP സ്ക്രിപ്റ്റുകളുടെ ഔട്ട്പുട്ട് പ്രവചിക്കുക:

```
a. <?php
   echo "Today looks\nbright and sunny";
   ?>

b. <?php
    echo "Ours not to question why;";
    echo "Ours but to do and die";
    ?>

c. <?php
   $x = 8;
   $y = 8;
   echo($x = = $y);
   ?>
```

21. ചുവടെ നൽകിയിരിക്കുന്ന PHP സ്ക്രിപ്റ്റുകളിലെ തെറ്റ് കണ്ടെത്തുക

```
a. <?php
    /* print a line of output /
    echo 'Hello'
    ?>

b. <?php
    echo '<p align="right">Copyright Me, 2008</p>';
    ?>

c. <?php
    echo 'Line 1;
    ?>
```

22. PHP-യിലെ വിവിധ തരം ലൂപ്പുകൾ ഏതെല്ലാം?

23. PHP-യിൽ ഒരു അറെ നിർമ്മിക്കുന്നത് എങ്ങിനെ?

24. HTML ഫോമിൽ action ആട്രിബ്യൂട്ടിന്റെ പ്രാധാന്യം എന്ത്?

25. "scert@hotmail.com" എന്ന സ്ട്രിങ്ങിൽ നിന്നും "hotmail.com" എന്ന സ്ട്രിങ്ങ് മാത്രം എടുക്കുന്നത് എങ്ങിനെ?

26. ചുവടെ നൽകിയിരിക്കുന്ന കോഡിന്റെ ഔട്ട്പുട്ട് പ്രവചിക്കുക:

```
$a = 10;
if ($a > 5 or < 15)
        echo 'true';
else
        echo 'false';
```

27. ചുവടെ നൽകിയിരിക്കുന്ന കോഡിന്റെ ഔട്ട്പുട്ട് പ്രവചിക്കുക:

```
function fn ($var)
{
$var = $var - ($var/10*5);
return $var;
}
echo fn(100);
```

28. സൂപ്പർ ഗ്ലോബൽ അറെകൾ ഏത് രീതിയിൽ PHP ഉപയോഗിച്ചുള്ള പ്രോഗ്രാമിങ്ങിനെ സഹായിക്കുന്നു?

29. $_SERVER-ന്റെ ഉപയോഗം എന്താണ്?

30. ഒരു ഫോം സമർപ്പിക്കുമ്പോൾ ഉപയോഗിക്കുന്ന get, post എന്നീ രീതികളുടെ വ്യത്യാസങ്ങൾ എഴുതുക.

31. ഒരു ഫോം സമർപ്പിക്കുമ്പോൾ അഡ്രസ്സ്ബാറിൽ കാണപ്പെട്ട URL http://localhost:8080/check2.php?exam=March2018&web=www.dhsekerala.gov.in എന്നതാണ്. ഇവിടെ ഉപയോഗിച്ചിരിക്കുന്ന ഫോം സമർപ്പണ രീതി ഏതാണെന്ന് തിരിച്ചറിയുക.

32. മുൻപ് സേവ് ചെയ്തിരുന്ന ബുക്മാർക്ക് ഉപയോഗിച്ച് ഒരു ഡയനാമിക് വെബ് പേജ് തുറക്കാൻ ശ്രമിച്ചപ്പോൾ ശരിയായ രീതിയിൽ വരുന്നില്ല. ഇതിന് കാരണം എന്തായിരിക്കും?

33. MySQL-മായുള്ള ബന്ധം നമുക്ക് ഏത് രീതിയിൽ സജ്ജമാക്കാം?

34. PHP-യിൽ എങ്ങിനെ ഒരു MySQL ഡാറ്റാബേസ് തിരഞ്ഞെടുക്കാം?

35. SQL ക്വറി എങ്ങിനെ പ്രവർത്തിപ്പിക്കാം? ഈ ക്വറിയുടെ ഔട്ട്പുട്ട് എങ്ങിനെ ലഭിക്കും?

36. PHP-യും JavaScript-ഉം തമ്മിലുള്ള വ്യത്യാസം എഴുതുക.

37. PHP-യെ, MySQL മായി ബന്ധിപ്പിക്കുന്നതിനുള്ള നടപടിക്രമങ്ങൾ വിവരിക്കുക.

M7K6C6

11 കമ്പ്യൂട്ടിങ്ങിലെ മുന്നേറ്റങ്ങൾ

പ്രധാന പഠന നേട്ടങ്ങൾ

ഈ അധ്യായത്തിന്റെ പഠനം പൂർത്തിയാകുന്നതോടെ പഠിതാവ് ആർജിക്കേണ്ട പഠനനേട്ടങ്ങൾ

- സീരിയൽ കമ്പ്യൂട്ടിങ്ങ് (Serial Computing) രീതികൾ മനസിലാക്കുന്നു
- ഡിസ്ട്രിബ്യൂറ്റഡ് കമ്പ്യൂട്ടിങ്ങ് (Distributed Computing) രീതികൾ തിരിച്ചറിയുന്നു
- സീരിയൽ കമ്പ്യൂട്ടിങ്ങും (Serial Computing) പാരലൽ കമ്പ്യൂട്ടിങ്ങും (Parallel Computing) താരതമ്യം ചെയ്യുന്നു
- ക്ലസ്റ്റർ കമ്പ്യൂട്ടിങ്ങും (Cluster Computing) ഗ്രിഡ് കമ്പ്യൂട്ടിങ്ങും (Grid Computing) തമ്മിലുള്ള വ്യത്യാസം കണ്ടെത്തുന്നു
- ക്ലസ്റ്റർ കമ്പ്യൂട്ടിങ്ങിന്റെ സാങ്കേതികതകൾ തിരിച്ചറിയുന്നു
- ഗ്രിഡ് കമ്പ്യൂട്ടിങ്ങിന്റെ ആവശ്യകത തിരിച്ചറിയുന്നു
- ക്ലൗഡ് കമ്പ്യൂട്ടിങ് മാതൃകയുടെ ആവശ്യകത പരിശോധിക്കുന്നു
- ആധുനിക യുഗത്തിലെ കമ്പ്യൂട്ടിങ്ങിൽ നിർമ്മിത ബുദ്ധി (Artifical Intelligence) യുടെ പങ്ക് വിശദമാക്കുന്നു
- സേവനമേഖലയിൽ ബയോമെട്രിക് ഉപകരണങ്ങളുടെ ഉപയോഗം തിരിച്ചറിയുന്നു
- കമ്പ്യൂട്ടർ ദൃഷ്ടിയുടെ (Computer Vision) പ്രസക്തി ഉദാഹരണസഹിതം വിശദീകരിക്കുന്നു
- റോബോട്ടിക്സിന്റെ (Robotics) സാധ്യതകൾ തിരിച്ചറിയുന്നു
- ബയോഇൻഫോർമാറ്റിക്സും അതിന്റെ വൈദ്യശാസ്ത്ര രംഗത്തെ പ്രാധാന്യവും വിശദീകരിക്കുന്നു
- വിദൂര സംവേദനത്തെ (Remote Sensing) കുറിച്ചും അതിന്റെ പ്രായോഗിക മേഖലകളെകുറിച്ചും വിശദമാക്കുന്നു
- ഭൂവിവരവ്യവസ്ഥയുടെ (Geographical Information System) ആവശ്യകത നിർണയിക്കുന്നു.

ഇന്റർനെറ്റിന്റെ വർദ്ധിച്ച ജനപ്രീതിയും, ശക്തിയേറിയ കമ്പ്യൂട്ടറുകളുടെ ലഭ്യതയും, ചെലവ് കുറഞ്ഞ ഉപകരണങ്ങളോടു കൂടിയ അതിവേഗ ശൃംഖലകളും കമ്പ്യൂട്ടിങ്ങ് രീതിയിൽ പല മാറ്റങ്ങളും ഉണ്ടാക്കിയിട്ടുണ്ട്. കമ്പ്യൂട്ടിങ്ങ് മേഖലയിൽ ക്രമാനുഗതമായി മാറ്റങ്ങൾ വന്നു കൊണ്ടിരിക്കുകയാണ്. ആ മേഖലയിൽ ഉയർന്നു വരുന്ന പ്രശ്നങ്ങളെ പുതിയ സാങ്കേതികവിദ്യ ഉപയോഗിച്ച് പരിഹരിക്കേണ്ടതുണ്ടെന്നും നമുക്ക് അറിയാം. മാറിക്കൊണ്ടിരിക്കുന്ന വിവിധ മേഖലകളിലെ കമ്പ്യൂട്ടിങ്ങ് പ്രതിസന്ധികൾ ആധുനിക കമ്പ്യൂട്ടിങ് മാതൃകകൾ ഉപയോഗിച്ച് പരിഹരിക്കുന്നതിനെക്കുറിച്ചാണ് ഈ അദ്ധ്യായത്തിൽ നാം ചർച്ച ചെയ്യുന്നത്. ഡിസ്ട്രിബ്യൂട്ടെഡ് കമ്പ്യൂട്ടർ വ്യവസ്ഥ ഇന്റർനെറ്റിന്റെ സഹായത്താൽ ഒന്നിലധികം കമ്പ്യൂട്ടറുകളെ കോർത്തിണക്കി സങ്കീർണ്ണങ്ങളായ പ്രശ്നങ്ങൾ പരിഹരിക്കുന്നു. അത് ഡാറ്റയ്ക്ക് കൂടുതൽ പ്രാധാന്യം നൽകയും ശൃംഖലാ കേന്ദ്രീകൃതമാവുകയും ചെയ്യുന്നു. ഇപ്പോൾ ഡിസ്ട്രിബ്യൂറ്റഡ് കമ്പ്യൂട്ടിങ്ങിന്റെ ഉപയോഗം കൂടുതൽ വ്യാപകമായിക്കൊണ്ടിരിക്കുകയാണ്. വലിയ തോതിൽ വിഭവങ്ങൾ പങ്കുവെച്ചുകൊണ്ട് മികച്ച പ്രകടനം കാഴ്ചവെയ്ക്കുക എന്നതിനാണ് ഇന്ന് ഡിസ്ട്രിബ്യൂറ്റഡ് കമ്പ്യൂട്ടിങ്ങ് (DISTRIBUTED COMPUTING) പ്രാധാന്യം നൽകുന്നത്. സോഷ്യൽ മീഡിയ സൈറ്റുകളായ ഫേസ്ബുക്ക്, ടിറ്റെർ തുടങ്ങിയവയും സെർച്ച് എൻജിൻ വെബ്സൈറ്റുകളായ ഗൂഗിൾ (GOOGLE), ബിങ് (BING) തുടങ്ങിയവയും ഡിസ്ട്രിബ്യൂറ്റഡ് കമ്പ്യൂ

ട്ടിങ്ങ് ആശയങ്ങൾ പ്രാവർത്തികമാക്കിയിട്ടുണ്ട്. ഇവയെല്ലാം കേന്ദ്രീകൃത ഡാറ്റ നിയന്ത്രണ സംവിധാനത്തിലാണ് പ്രവർത്തിക്കുന്നത്. എന്നിരുന്നാലും വലിയ തോതിൽ വിപുലീകരിക്കാവുന്ന ഈ സംവിധാനങ്ങളിൽ മെച്ചപ്പെട്ട കമ്പ്യൂട്ടിങ്ങ് മാതൃകകളായ ക്ലസ്റ്റർ കമ്പ്യൂട്ടിങ്ങ് (CLUSTER COMPUTING), ഗ്രിഡ് കമ്പ്യൂട്ടിങ്ങ് (GRID COMPUTING), ക്ളൗഡ് കമ്പ്യൂട്ടിങ്ങ് (CLOUD COMPUTING) തുടങ്ങിയവ വ്യാപകമായി ഉപയോഗിക്കുന്ന പ്രവണതയാണുള്ളത്. ഈ മേഖലകളെക്കുറിച്ചുള്ള സംക്ഷിപ്തമായ ചർച്ചകൾ ഈ അദ്ധ്യായത്തിൽ ഉണ്ടാകും.

ധാരാളം ഗവേഷണങ്ങൾ നടന്നുകൊണ്ടിരിക്കുന്ന ഒരു മേഖലയാണ് കമ്പ്യൂട്ടേഷണൽ ബുദ്ധി (COMPUTATIONAL INTELLIGENCE). ബുദ്ധിവൈഭവമുള്ള കമ്പ്യൂട്ടറിന് മനുഷ്യരുമായി ആശയവിനിമയം നടത്തുവാൻ സാധിക്കും. മനുഷ്യനും യന്ത്രവും തമ്മിലുള്ള പരസ്പര വിനിമയത്തെക്കുറിച്ചുള്ള പഠനമാണ് സൈബർനെറ്റിക്സ് (CYBERNETICS). എക്കാലവും ആകാംക്ഷയും കൗതുകവും ഉണർത്തുന്ന ഒരു മേഖലയാണ് നിർമ്മിതബുദ്ധി (ARTIFICIAL INTELLIGENCE). ബയോമെട്രിക്, നാച്ചുറൽ ലാംഗ്വേജ് പ്രോസസ്സിംഗ് (NLP), റോബോട്ടിക്സ് തുടങ്ങിയ സാങ്കേതിക വിദ്യകളിൽ നിർമിതബുദ്ധി പലതരത്തിൽ ഉപയോഗിച്ചിരിക്കുന്നു. ഇൻപുട്ട് സ്വീകരിക്കുന്നതിലും ഔട്ട്പുട്ട് നൽകുന്നതിലും ഉള്ള കൃത്യതയ്ക്ക് കമ്പ്യൂട്ടറുകൾ പേരുകേട്ടതാണ്. അനിശ്ചിതത്വവും ഭാഗിക സത്യങ്ങളും നിറഞ്ഞ മറ്റൊരു കമ്പ്യൂട്ടിങ് മേഖലയായ സോഫ്റ്റ് കമ്പ്യൂട്ടിങ്ങിനെ (SOFT COMPUTING) കുറിച്ചും ഈ അദ്ധ്യായത്തിൽ ചർച്ച ചെയ്യുന്നു. കമ്പ്യൂട്ടർ രംഗത്തെ വികാസം മറ്റേതൊരു മേഖലയ്ക്കും എന്ന പോലെ ജീവശാസ്ത്രമേഖലയ്ക്കും ഗണ്യമായ സംഭാവനകൾ നൽകിയിട്ടുണ്ട്. കമ്പ്യൂട്ടർ സാങ്കേതികവിദ്യ ഉപയോഗിച്ച് ജീവശാസ്ത്രപരമായ വിവരങ്ങൾ കൈകാര്യം ചെയ്യുന്ന മേഖലയാണ് ബയോഇൻഫോർമാറ്റിക്സ് (BIOINFORMATICS). ശാസ്ത്രജ്ഞന്മാർ ലോക നന്മയ്ക്കായി ഡി എൻ എ കമ്പ്യൂട്ടിങ് വികസിപ്പിച്ചെടുക്കുവാൻ ശ്രദ്ധ ചെലുത്തുന്നുണ്ട്. ഈ വിഷയങ്ങളും ഇവിടെ ചർച്ച ചെയ്യുന്നുണ്ട്. വിദൂര സംവേദനത്തെക്കുറിച്ചും (REMOTE SENSING) ഭൂവിവരവ്യവസ്ഥയെ (GIS) കുറിച്ചും ചർച്ച ചെയ്തുകൊണ്ട് അദ്ധ്യായം അവസാനിക്കുന്നു.

11.1 ഡിസ്ട്രിബ്യൂറ്റഡ് കമ്പ്യൂട്ടിങ്ങ് മാതൃകകൾ (DISTRIBUTED COMPUTING PARADIGMS)

ചരിത്രപരമായി ഹാർഡ് വെയർ, ഓപ്പറേറ്റിംഗ് സിസ്റ്റം, ശൃംഖല മുതലായവയിലെ വിവിധ മാറ്റങ്ങളിലൂടെ ആണ് കമ്പ്യൂട്ടിങ്ങ്മേഖല കടന്നുവന്നത്. കമ്പ്യൂട്ടർ നമ്മുടെ ദൈനംദിന ജീവിതത്തിന്റെ ഭാഗമായി മാറിക്കഴിഞ്ഞു. കമ്പ്യൂട്ടർ സംവിധാനങ്ങൾ ഫലപ്രദമായ ഉപയോഗിക്കുന്നുവെന്ന് ഡിസ്ട്രിബ്യൂറ്റഡ് കമ്പ്യൂട്ടിങ്ങ് ഉറപ്പു വരുത്തുന്നു. ഒരു കമ്പ്യൂട്ടർ ശൃംഖലയിലൂടെ ആശയവിനിമയം നടത്തികൊണ്ടിരിക്കുന്ന രണ്ടോ അതിലധികമോ കമ്പ്യൂട്ടറുകൾ, ഒരു പ്രോഗ്രാമിന്റെ തന്നെ വ്യത്യസ്ത ഭാഗങ്ങൾ ഒരേസമയം പ്രവർത്തിപ്പിക്കുന്ന രീതിയാണ് ഡിസ്ട്രിബ്യൂറ്റഡ് കമ്പ്യൂട്ടിങ്ങ്.

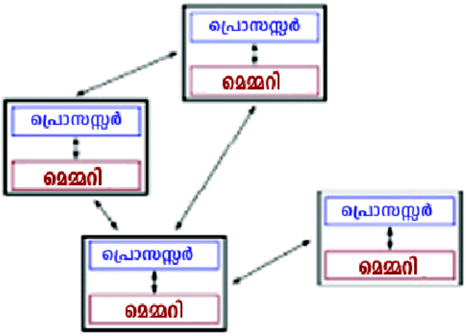
ഏതൊരു സങ്കീർണമായ വിഷയത്തിന്റെയും രൂപകൽപ്പനയെക്കുറിച്ച് അല്ലെങ്കിൽ സ്വരൂപത്തെകുറിച്ച് പഠിക്കുന്നതാണ് മാതൃക (PARADIGM) എന്ന പദം കൊണ്ട് ഉദ്ദേശിക്കുന്നത്. ഈ മാതൃക അനുസരിച്ചു വിഷയങ്ങളുടെ വിശദാംശങ്ങൾ തിരിച്ചറിയുന്നതിനും തരംതിരിക്കുന്നതിനും നമുക്ക് കഴിയും. സമൂഹത്തിലെ പല മേഖലകളിലേയും വിവരങ്ങൾ കൈകാര്യം ചെയ്യാനുള്ള കഴിവ് പരമാവധി വർധിപ്പിക്കുവാൻ ആധുനിക കമ്പ്യൂട്ടിങ്ങ് മാതൃകകൾ (ADVANCED

COMPUTING PARADIGMS) അത്യന്താപേക്ഷിതമാണ്. ആധുനിക കമ്പ്യൂട്ടിങ്ങ് മാതൃകകളിൽ, പാരലൽ കമ്പ്യൂട്ടിങ്ങ് (PARALLEL COMPUTING), ക്ലസ്റ്റർ കമ്പ്യൂട്ടിങ്ങ് (CLUSTER COMPUTING), ഗ്രിഡ് കമ്പ്യൂട്ടിങ്ങ് (GRID COMPUTING), ക്ലൗഡ് കമ്പ്യൂട്ടിങ്ങ് (CLOUD COMPUTING) തുടങ്ങിയവ ഉൾപ്പെടുന്നു. പാരലൽ കമ്പ്യൂട്ടിങ്ങിന്റെയും ഡിസ്ട്രിബ്യൂറ്റഡ് കമ്പ്യൂട്ടിങ്ങിന്റെയും വിവിധ രൂപങ്ങളാണ് ഇവയെല്ലാം ഇവ ചുവടെ ചർച്ച ചെയ്യുന്നു.

11.2 ഡിസ്ട്രിബ്യൂറ്റഡ് കമ്പ്യൂട്ടിങ്ങ് (DISTRIBUTED COMPUTING)

നമ്മുടെ ദൈനംദിന ജീവിതത്തിൽ ഡിസ്ട്രിബ്യൂറ്റഡ് കമ്പ്യൂട്ടിങ്ങ് ഉപയോഗിക്കുന്ന വിവിധ മേഖലകൾ പരിഗണിക്കാം. കോർബാങ്കിങ് സംവിധാനമാണ് ഇപ്പോൾ ബാങ്കുകളിൽ ഉപയോഗിക്കുന്നതെന്ന് നമുക്കറിയാം. ഇതിന്റെ ഫലമായി ഏതു എ ടി എം കൗണ്ടറിൽ നിന്നും നമുക്ക് പണം പിൻവലിക്കുവാൻ സാധിക്കുന്നു. ഇതുപോലെ ട്രെയിൻ, വിമാന ടിക്കറ്റുകളും ഓൺലൈൻ ആയി നമുക്ക് ബുക്ക് ചെയ്യാം. ഇത് എല്ലാം സാധ്യമാകുന്നത് ഡിസ്ട്രിബ്യൂറ്റഡ് സാഹചര്യം ഉപയോഗിച്ചാണ്. ഏറ്റവും വലിയ ഡിസ്ട്രിബ്യൂറ്റഡ് കമ്പ്യൂട്ടിങ്ങ് സംവിധാനത്തിനുള്ള നല്ല ഒരു ഉദാഹരണമാണ് വേൾഡ് വൈഡ് വെബ് (WWW). ഇക്കാലത്തു ഡിസ്ട്രിബ്യൂറ്റഡ് വ്യവസ്ഥയാണ് കമ്പ്യൂട്ടിങ്ങ് മേഖലയിൽ മേൽക്കോയ്മ പുലർത്തുന്നത്. ഒരു വലിയ കൃത്യത്തെ ചെറിയ കൃത്യങ്ങളായി വിഭജിച് വിവിധ കമ്പ്യൂട്ടറുകളിലായി വിന്യസിച്ച് പരിഹരിക്കുന്ന രീതിയാണ് ഡിസ്ട്രിബ്യൂറ്റഡ് കമ്പ്യൂട്ടിങ്ങ് എന്നത്. പിന്നീട് ഓരോ കമ്പ്യൂട്ടറിലെയും ഫലങ്ങളെ സംയോജിപ്പിച്ച് അന്തിമ ഫലത്തിൽ എത്തിച്ചേരുന്നു. ഒരു വലിയ കൃത്യം ചെയ്യുന്നതിനാവശ്യമായ സമയം കുറയ്ക്കുവാൻ ഡിസ്ട്രിബ്യൂറ്റഡ് കമ്പ്യൂട്ടിങ്ങ് വഴിയൊരുക്കുന്നു.

ശൃംഖല വഴി പരസ്പരം ആശയവിനിമയം നടത്തുന്ന രണ്ടോ അതിലധികമോ കമ്പ്യൂട്ടറുകളിൽ ഒരു പ്രോഗ്രാമിന്റെ പല ഭാഗങ്ങൾ സമാന്തരമായി പ്രവർത്തിപ്പിച്ചാണ് ഇത് സാധ്യമാക്കുന്നത്.



ചിത്രം 11.1: ഡിസ്ട്രിബ്യൂറ്റഡ് കമ്പ്യൂട്ടിങ്

ഒറ്റ കമ്പ്യൂട്ടർ ജോലി ചെയ്ത തീർക്കുവാൻ ആവശ്യമായ സമയത്തേക്കാൾ കുറഞ്ഞ സമയത്തിൽ ഡിസ്ട്രിബ്യൂറ്റഡ് കമ്പ്യൂട്ടിങ്ങ് ജോലി ചെയ്ത് തീർക്കുന്നു. ഡിസ്ട്രിബ്യൂറ്റഡ് കമ്പ്യൂട്ടിങ്ങിൽ, എല്ലാ പ്രൊസസ്സറുകൾക്കും പരസ്പരം പങ്കിടാൻ കഴിയാത്ത സ്വകാര്യ മെമ്മറികളാണ് ഉള്ളത്. ആയതിനാൽ ആശയങ്ങൾ സന്ദേശ രൂപത്തിൽ മാത്രമായിരിക്കും. കൈമാറ്റം ചെയ്യപ്പെടുന്നത്. ഡിസ്ട്രിബ്യൂറ്റഡ് കമ്പ്യൂട്ടിങ്ങ് രീതിയുടെ ഘടന ചിത്രം 11.1 – ൽ കാണിച്ചിരിക്കുന്നു.

a. ഡിസ്ട്രിബ്യൂറ്റഡ് കമ്പ്യൂട്ടിങ്ങിന്റെ മേന്മകൾ

ഡിസ്ട്രിബ്യൂറ്റഡ് കമ്പ്യൂട്ടിങ്ങിന്റെ മേന്മകൾ ചുവടെ ചേർക്കുന്നു.

ചെലവ് കുറയ്ക്കുന്നു: ഉപകരണങ്ങളുടെ പരസ്പര സഹകരണത്തിലൂടെ കമ്പ്യൂട്ടിങ്ങ് പ്രക്രിയയുടെ മൊത്തത്തിലുള്ള ചെലവ് ഡിസ്ട്രിബ്യൂറ്റഡ് കമ്പ്യൂട്ടിങ്ങ് കുറയ്ക്കുന്നു.

വേഗത: കമ്പ്യൂട്ടേഷണൽ ഭാരം വ്യത്യസ്ത നോഡുകളിലായി വിന്യസിക്കുന്നതിനാൽ ഓരോന്നിനുമുള്ള ജോലി ഭാരം കുറയുകയും തത്ഫലമായി വേഗത വർദ്ധിക്കുകയും ചെയ്യുന്നു.

വിശ്വാസ്യത : ഏതെങ്കിലും ഒരു കമ്പ്യൂട്ടറിന്റെ പ്രവർത്തനം നിലച്ചാലും ഡിസ്ട്രിബ്യൂട്ടഡ് കമ്പ്യൂട്ടിങ്ങ് സംവിധാനത്തിന്റെ പ്രവർത്തനം തടസപ്പെടുന്നില്ല.

വിപുലീകരണ ശേഷി: ഡിസ്ട്രിബ്യൂട്ടഡ് കമ്പ്യൂട്ടിങ്ങിൽ ആവശ്യത്തിനനുസരിച്ചു നോഡുകളുടെ എണ്ണം വ്യത്യാസപ്പെടുത്താം .

b. ഡിസ്ട്രിബ്യൂട്ടഡ് കമ്പ്യൂട്ടിങ്ങിന്റെ പോരായ്മകൾ

ഡിസ്ട്രിബ്യൂട്ടഡ് കമ്പ്യൂട്ടിങ്ങിന്റെ പരിമിതികൾ ചുവടെ ചേർക്കുന്നു.

സങ്കീർണതകൾ : ഡിസ്ട്രിബ്യൂട്ടഡ് കമ്പ്യൂട്ടിങ്ങ് സംവിധാനം ഒരുക്കുന്നതിനായി ധാരാളം പ്രത്യേക പ്രോഗ്രാമുകൾ ആവശ്യമാണ്.

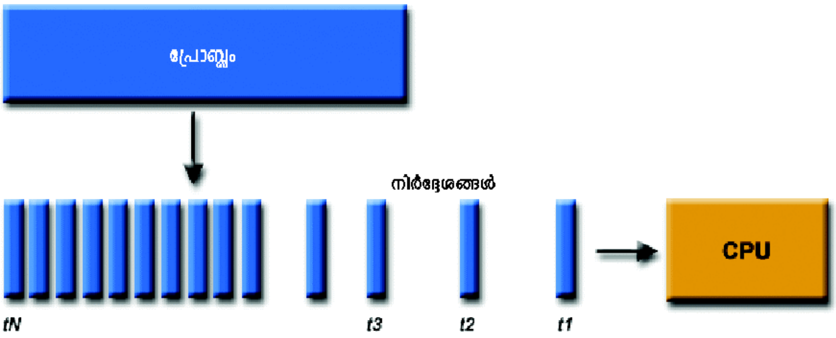
സുരക്ഷിതത്വം : ഈ സംവിധാനത്തിൽ ശൃംഖലയിലൂടെ കമ്പ്യൂട്ടറുകളിലേക്കു വിവരങ്ങൾ കൈമാറ്റം ചെയ്യേണ്ടി വരുന്നു. ആയതിനാൽ അനധികൃത ഉപയോക്താക്കൾ വിവരങ്ങൾ കൈക്കലാക്കാനും അവയെ ദുരുപയോഗം ചെയ്യാനും സാധ്യത കൂടുതലാണ്

ശൃംഖലയോടുള്ള ആശ്രയത്വം: ഡിസ്ട്രിബ്യൂട്ടഡ് സംവിധാനം ശൃംഖലയുമായി ബന്ധിപ്പിച്ചിരിക്കുന്നതിനാൽ, ശൃംഖലയ്ക്കു തകരാറുണ്ടായാൽ മുഴുവൻ സംവിധാനവും അസ്ഥിരമാകും.

വിവിധ തരം ഡിസ്ട്രിബ്യൂട്ടഡ് കമ്പ്യൂട്ടിങ്ങ് രീതികൾ നമുക്ക് ഇപ്പോൾ ചർച്ച ചെയ്യാം.

11.2.1 പാരലൽ കമ്പ്യൂട്ടിങ്ങ് (parallel computing)

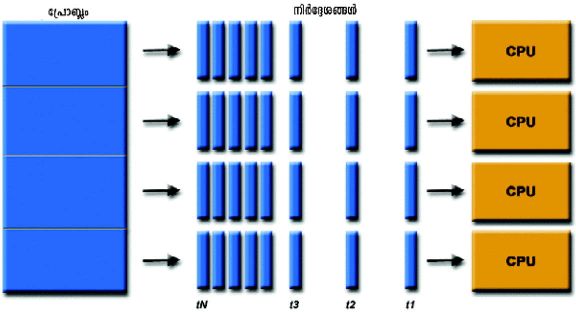
സാധാരണയായി സോഫ്റ്റ്‌വെയറുകൾ തയ്യാറാക്കുന്നത് നിർദ്ദേശങ്ങൾ ഒന്നിനു പുറകേ മറ്റൊന്നായി പ്രവർത്തിക്കത്തക്ക വിധത്തിലുള്ള കമ്പ്യൂട്ടിങ് രീതിയ്ക്ക് വേണ്ടിയാണ്. അതായത് ഒരു പ്രോസസ്സർ മാത്രം ഉള്ള കമ്പ്യൂട്ടറിൽ ആയിരിക്കും ഈ സോഫ്റ്റ്‌വെയർ പ്രവർത്തിക്കുക. ഇവിടെ ഒരു കൃത്യത്തെ തുടർച്ചയായുള്ള നിർദേശങ്ങൾ ആയി വിഭജിക്കുന്നു. ഈ നിർദ്ദേശങ്ങൾ ക്രമാനുഗതമായി നടപ്പിലാക്കുന്നു. ഒരു സമയത്ത് ഒരു നിർദേശം ആയിരിക്കും നടപ്പിലാക്കുന്നത്. ഇതിനെ സീരിയൽ കമ്പ്യൂട്ടിങ്ങ് എന്ന് പറയുന്നു. സീരിയൽ പ്രോസസ്സിങ്ങ് ചിത്രം 11.2 ൽ ചിത്രീകരിച്ചിരിക്കുന്നു.



ചിത്രം 11.2: സീരിയൽ പ്രോസസ്സിങ്ങ്

അഞ്ചുപേർ ചേർന്ന് ഒരു കയർ ഉപയോഗിച്ച് ഒരു പാറയെ ഉയർന്നു എന്ന് സങ്കൽപ്പിക്കുക. സമാന്തരമായി ആണ് ഇവരെല്ലാം പാറ ഉയർത്താൻ ശ്രമിക്കുന്നത്. ഇതുപോലെയാണ് പാരലൽ കമ്പ്യൂട്ടിങ്ങിൽ പ്രശ്നങ്ങളെ വിവിധ പ്രോസസ്സിംഗ് യൂണിറ്റുകളിൽ പരിഹരിക്കുന്നത്. ചെയ്യേണ്ട പ്രവർത്തിയെ വിഭജിച്ച് വിവിധ പ്രോസസ്സറുകൾക്ക് നൽകുന്നു. വിഭജിക്കപ്പെട്ട ഇത്തരം ചെറിയ

പ്രവർത്തികൾ വിവിധ പ്രോസസ്സറുകൾ സമാന്തരമായി ചെയ്യുന്നു. പാരലൽ കമ്പ്യൂട്ടിങ്ങിൽ വ്യത്യസ്ത പ്രൊസസ്സറുകൾ പൊതുവായി ഉപയോഗിക്കുന്ന മെമ്മറികൾ ലഭ്യമാണ്.



ചിത്രം 11.3: പാരൽ പ്രോസസ്സിംഗ്

ചിത്രം 11.4: പാരൽ പ്രോസസ്സിംങ്ങിലെ മെമ്മറി പങ്കു വയ്ക്കൽ

സന്ദേശങ്ങൾ മാധ്യമം വഴി കൈമാറുന്നതിന് പകരം പൊതുവായ മെമ്മറികളിൽ സൂക്ഷിച്ച് ആശയവിനിമയം നടത്താം. ചിത്രം 11.3 ൽ പാരലൽ പ്രോസസ്സിംഗ് ചിത്രീകരിച്ചിരിക്കുന്നു. പൊതുവായ മെമ്മറി വിവിധ പ്രോസസ്സിങ്ങ് യൂണിറ്റുകൾ എങ്ങനെ ഉപയോഗിക്കുന്നു എന്ന് ചിത്രം 11.4 വിശദീകരിക്കുന്നു. പട്ടിക 11.1 ൽ പാരലൽ കമ്പ്യൂട്ടിങ്ങ്, സീരിയൽ കമ്പ്യൂട്ടിങ്ങ് എന്നിവ തമ്മിൽ താരതമ്യം ചെയ്യാതിരിക്കുന്നു. കാലാവസ്ഥ പ്രവചനം, ആണവ ശാസ്ത്രം, ബഹിരാകാശ ശാസ്ത്രം തുടങ്ങിയവ പാരലൽ കമ്പ്യൂട്ടിങ്ങ് ഉപയോഗിക്കുന്ന ചില മേഖലകളാണ്.

സീരിയൽ കമ്പ്യൂട്ടിങ്	പാരലൽ കമ്പ്യൂട്ടിങ്
ഒരു പ്രോസസ്സർ ഉപയോഗിക്കുന്നു	ഒന്നിലധികം പ്രോസസ്സറുകൾ പൊതുവായ മെമ്മറി ഉപയോഗിക്കുന്നു
കൃത്യത്തെ പല നിർദേശങ്ങളായി വിഭജിക്കുന്നു.	കൃത്യത്തെ സമാന്തരമായി നിർദ്ധാരണം ചെയ്യുവാൻ കഴിയുന്ന വിധത്തിൽ പല ഭാഗങ്ങളാക്കി വിഭജിക്കുന്നു
നിർദ്ദേശ്ശങ്ങൾ ഒന്നിന് പുറകെ മറ്റൊന്നായി പ്രവർത്തിക്കുന്നു	നിർദേശങ്ങളുടെ ഓരോ ഭാഗവും വ്യത്യസ്ത പ്രോസസ്സറുകളിൽ ഒന്നിച്ച് പ്രവർത്തിക്കുന്നു
ഒരു സമയം ഒരു നിർദേശം മാത്രമേ പ്രോസെസ്സറിൽ പ്രവർത്തിക്കൂ	ഒന്നിലധികം നിർദ്ദേശങ്ങൾ വ്യത്യസ്ത പ്രൊസസ്സറുകളിൽ ഒരേസമയം സമാന്തരമായി പ്രവർത്തിക്കുന്നു

പട്ടിക 11.1: സീരിയൽ കമ്പ്യൂട്ടിങ്ങ്, പാരലൽ കമ്പ്യൂട്ടിങ്ങ് എന്നിവ തമ്മിലുള്ള താരതമ്യം

അനേകം CPU കളുള്ള വേഗതയേറിയ കമ്പ്യൂട്ടറാണ് ഒരു സൂപ്പർ കമ്പ്യൂട്ടർ. സാധാരണ കമ്പ്യൂട്ടർ സീരിയൽ പ്രക്രിയ ഉപയോഗിക്കുന്നു. എന്നാൽ ഒരു സൂപ്പർ കമ്പ്യൂട്ടർ തികച്ചും വിഭിന്നമായ രീതിയിലാണ് പ്രവർത്തിക്കുന്നത്. സീരിയൽ പ്രക്രിയയ്ക്കു പകരം സവിശേഷമായ സമാന്തര പ്രക്രിയയാണ് ഉപയോഗിക്കുന്നത്. അതായതു ഒരു കാര്യം ഒരു സമയത്തു ചെയ്യുന്നതിന് പകരം ഒന്നിലധികം കാര്യങ്ങൾ ഒരേ സമയത്തു ചെയ്യുക എന്ന രീതിയാണ് ഇത്. ഇന്ത്യയിലെ ഏറ്റവും വേഗതയേറിയ സൂപ്പർ കമ്പ്യൂട്ടർ ആയി കരുതുന്നത് ഇൻസ്റ്റിറ്റ്യൂട്ട് ഓഫ് ട്രോപ്പിക്കൽ മീറ്റിയോറോളജിയുടെ ആദിത്യ എന്ന സൂപ്പർ കമ്പ്യൂട്ടർ ആണ്. 790 + TERA FLOPS ഉയർന്ന പ്രവർത്തനക്ഷമതയുള്ള ആദിത്യ എന്ന കമ്പ്യൂട്ടർ IBM iDataPlex cluster ആണ് ഉപയോഗിക്കുന്നത്. ഇപ്പോൾ ഇന്ത്യയിലെ ഏറ്റവും വേഗതയേറിയ സൂപ്പർ കമ്പ്യൂട്ടറായ ഇൻസ്റ്റിറ്റ്യൂട്ട് ഓഫ് ട്രോപ്പിക്കൽ മീറ്റിയോറോളജിയുടെ പ്രത്യുഷ് 6.8 Peta Flops വേഗതയിൽ ആണ് പ്രവർത്തിക്കുന്നത്. ലോകത്തിലെ തന്നെ നാലാമത്തെ വേഗതയേറിയ കമ്പ്യൂട്ടറാണിത്.

a. പാരലൽ കമ്പ്യൂട്ടിങ്ങിന്റെ മേന്മകൾ

പാരലൽ കമ്പ്യൂട്ടിങ്ങിന് ധാരാളം മേന്മകൾ ഉണ്ട്. അവയിൽ ചിലതു ചുവടെ ചേർക്കുന്നു.

- ഒന്നോ അതിലധികമോ നോഡുകൾ തകരാറിലായാലും താഴ്ന്ന പ്രവർത്തനക്ഷമതയോടു കൂടി പാരലൽ കമ്പ്യൂട്ടിങ്ങ് സംവിധാനം പ്രവർത്തിക്കുന്നു. ഇതിനെ ഫോൾട്ട് ടോളറൻസ് (Fault Tolerance) എന്ന് പറയുന്നു.
- ഓരോ ഉപയോക്താവിനും ഈ സംവിധാനത്തിലെ കമ്പ്യൂട്ടിങ്ങ് ശക്തിയും സംഭരണ ഉപകരണങ്ങളും മറ്റുള്ളവരുമായി പങ്കിടാം
- പ്രക്രിയകളെ വിവിധ നോഡുകളിലേക്കു വികേന്ദ്രീകരിക്കുന്നതിനാൽ, ജോലിഭാരം സംവിധാനം മൊത്തത്തിൽ പങ്കിടുന്നു. ഇതിനെ ജോലി ഭാരം പങ്കുവെക്കൽ (ലോഡ് ഷെയറിങ്) എന്ന് വിളിക്കുന്നു.
- ഈ സംമ്പ്രദായത്തെ വളരെ എളുപ്പത്തിൽ വിപുലീകരിക്കാനും വ്യാപിപ്പിക്കാനും സാധിക്കും.

b. പാരലൽ കമ്പ്യൂട്ടിങ്ങിന്റെ പോരായ്മകൾ

പാരലൽ കമ്പ്യൂട്ടിങ്ങിന്റെ ചില പോരായ്മകൾ ഇവയാണ്:

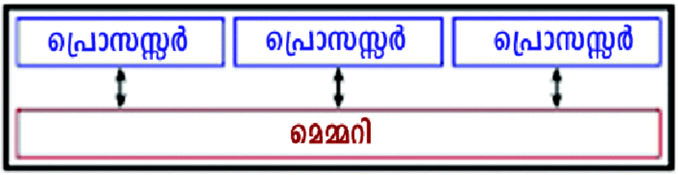
- സീരിയൽ ആപ്ലിക്കേഷനെ അപേക്ഷിച്ചു തത്തുല്യമായ പാരലൽ ആപ്ലിക്കേഷന്റെ പ്രവർത്തനം അതിസങ്കീർണമാണ്
- ഒരു കമ്പ്യൂട്ടറിൽ പ്രവർത്തിക്കുന്ന പ്രോഗ്രാം മറ്റൊന്നിലേക്കു മാറ്റുമ്പോൾ ആവശ്യമായ മാറ്റങ്ങൾ വരുത്തിയാൽ മാത്രമേ ഈ പ്രോഗ്രാമുകൾ കൃത്യമായി പ്രവർത്തിക്കുകയുള്ളൂ.

11.2.2 ഗ്രിഡ് കമ്പ്യൂട്ടിങ്ങ്

കോടിക്കണക്കിനു സ്വകാര്യ കമ്പ്യൂട്ടറുകളും (സർവറുകൾ, ഡെസ്ക്ടോപ്പുകൾ ലാപ്ടോപ്പുകൾ തുടങ്ങിയവയും) ആയിരക്കണക്കിന് മൊബൈൽ ഫോണുകൾ, ടെലിസ്കോപ്പുകൾ, സെൻസറുകൾ തുടങ്ങിയ ഉപകരണങ്ങളും പരസ്പരം ബന്ധിപ്പിച്ച് ഒരു ശക്തമായ കമ്പ്യൂട്ടർ രൂപകൽപ്പന ചെയ്യുന്നത് ഒന്ന് സങ്കൽപ്പിച്ചു നോക്കുക. ഇതാണ് ഗ്രിഡ് കമ്പ്യൂട്ടിങ്ങ്. വൈദ്യുതോർജം ലഭ്യമാകുന്നതുപോലെ കമ്പ്യൂട്ടേഷണൽ ഊർജം (വിഭവങ്ങൾ, സേവനങ്ങൾ, ഡാറ്റ) ലഭിക്കുന്നതിനെ ഗ്രിഡ് കമ്പ്യൂട്ടിങ്ങ് എന്ന് പറയുന്നു. ഒരു പൊതുലക്ഷ്യത്തിനായി പ്രവർത്തിക്കുന്നതും ലോകത്തിന്റെ വിവിധ ഭാഗങ്ങളിലായി വിന്യസിച്ചിരിക്കുന്നതുമായ വിഭവങ്ങളുടെ ശേഖരമാണ് ഗ്രിഡ്. വൈദ്യുതോർജം എവിടെ, എങ്ങനെ ഉണ്ടാക്കുന്നു എന്നത് പരിഗണിക്കാതെ ഒരു വൈദ്യുത സോക്കറ്റ് വഴി അത് ലഭിക്കുന്നതുപോലെ, കമ്പ്യൂട്ടേഷണൽ ഊർജവും ഉപയോക്താക്കൾക്ക് ലഭ്യമാകുന്നു. ഇവിടെ ഗ്രിഡിലുള്ള കമ്പ്യൂട്ടറുകളിലെ പ്രോഗ്രാമുകളും, ഉപയോഗിക്കാത്ത അനുബന്ധ ഉപകരണങ്ങളും (സാധാരണയായി പ്രവർത്തന സമയവും മെമ്മറിയും) ഗ്രിഡിലെ മറ്റു കമ്പ്യൂട്ടറുകൾക്ക് ഉപയോഗിക്കാൻ സാധിക്കുന്നു.

ഗ്രിഡിലെ കമ്പ്യൂട്ടറുകൾ തമ്മിലുള്ള വിനിമയ വേഗത കുറവായതിനാൽ, കൃത്യം വിഭജിച്ച് സ്വതന്ത്ര ഭാഗങ്ങളാക്കി ഗ്രിഡിലെ വിവിധ കമ്പ്യൂട്ടറുകളിലേക്ക് അയയ്ക്കുന്നു. ഓരോ

കമ്പ്യൂട്ടറിൽ നിന്നും ലഭിക്കുന്ന ഫലം സർവറിലേക്കു തിരികെ അയയ്ക്കുന്നു. ഗ്രിഡിൽ പൂർത്തിയാകുന്ന സേവനങ്ങൾ വിവിധതരത്തിലുള്ളതാണ്. വിജ്ഞാനം ലഭ്യമാക്കുന്നവ (knowledge grid), ഡാറ്റ ലഭ്യമാക്കുന്നവ (data grid), കമ്പ്യൂട്ടേഷൻ സാധ്യമാക്കുന്നവ (computational grid) മുതലായവ. വളരെ വേഗം പരസ്പര സഹകരണം സാധ്യമാകുന്നതിനാൽ നിലവിലുള്ള ഹാർഡ്‌വെയർ ഫലപ്രദമായി ഉപയോഗിച്ചു കൊണ്ട് വലുതും സങ്കീർണവുമായ പ്രശ്നങ്ങളെ ചുരുങ്ങിയ സമയത്തിനുള്ളിൽ ഗ്രിഡ് കമ്പ്യൂട്ടിങ്ങ് പരിഹരിക്കുന്നു. ദുരന്ത നിവാരണം, കാലാവസ്ഥാ പ്രവചനം, ഓഹരി കമ്പോള പ്രവചനം, ബയോ ഇൻഫോർമാറ്റിക്സ്, തുടങ്ങിയ മേഖലകളിൽ ഗ്രിഡ് കമ്പ്യൂട്ടിങ്ങ് ഉപയോഗിക്കുന്നു. ഗ്രിഡ് കമ്പ്യൂട്ടിങ്ങ് രീതിയുടെ ഘടന ചിത്രം 11.5 ൽ ചിത്രീകരിച്ചിരിക്കുന്നു.



ചിത്രം 11.5: ഗ്രിഡ് കമ്പ്യൂട്ടിങ്ങ് സംവിധാനം

a. ഗ്രിഡ് കമ്പ്യൂട്ടിങ്ങിന്റെ മേന്മകൾ

ഗ്രിഡ് കമ്പ്യൂട്ടിങ്ങ് ഉപയോഗിക്കുന്നതിന്റെ ഗുണങ്ങൾ ഇവയാണ്

- അതിസങ്കീർണമായ വലിയ കൃത്യങ്ങൾ കുറഞ്ഞ സമയം കൊണ്ട് ചെയ്തു തീർക്കുവാൻ ഗ്രിഡുകൾക്കു കഴിയുന്നു
- നിലവിൽ ലഭ്യമായ ഉപകരണങ്ങളെ ഗ്രിഡുകൾ കൂടുതൽ ഫലപ്രദമായി ഉപയോഗിക്കുന്നു.
- ഡെസ്ക്ടോപ്പ് കമ്പ്യൂട്ടറുകൾ അല്ലെങ്കിൽ സർവർ കമ്പ്യൂട്ടറുകൾ കൂട്ടിച്ചേർത്ത് വളരെ അനായാസമായി കമ്പ്യൂട്ടിങ്ങ് ശക്തി വർദ്ധിപ്പിക്കാം

b. ഗ്രിഡ് കമ്പ്യൂട്ടിങ്ങിന്റെ പോരായ്മകൾ

ഗ്രിഡ് കമ്പ്യൂട്ടിങ്ങിന്റെ ന്യൂനതകൾ ഇവയാണ്:

- കമ്പ്യൂട്ടറുകൾ തമ്മിലുള്ള ആശയ വിനിമയ വേഗത കുറവായതിനാൽ ഗ്രിഡിന്റെ പ്രവർത്തന വേഗതയെ അത് ബാധിക്കുന്നു
- വിവിധ കമ്പ്യൂട്ടറുകളും സർവറുകളും തമ്മിലുള്ള ലൈസൻസിങ് പ്രശ്നങ്ങൾ ചില ആപ്ലിക്കേഷനുകളുടെ പ്രവർത്തനത്തെ ബാധിക്കുന്നു.

11.2.3 ക്ലസ്റ്റർ കമ്പ്യൂട്ടിങ്ങ്

ക്ലസ്റ്റർ കമ്പ്യൂട്ടിങ്ങ് എന്നത് കമ്പ്യൂട്ടിങ്ങ് മേഖലയിലെ ഒരു സംവിധാനമാണ്. ഇവിടെ ഒരു കൂട്ടം സ്വകാര്യ കമ്പ്യൂട്ടറുകൾ, സംഭരണ ഉപകരണങ്ങൾ എന്നിവയെ പരസ്പരം ബന്ധിപ്പിച്ച് ഒറ്റ കമ്പ്യൂട്ടർ എന്ന നിലയിൽ പ്രവർത്തിപ്പിക്കുന്നു. ക്ലസ്റ്റർ കമ്പ്യൂട്ടറിന്റെ ഭാഗങ്ങൾ വേഗതയേറിയ ഒരു ലാൻ (LAN) ശൃംഖല വഴി ബന്ധിപ്പിച്ചിരിക്കുന്നു. ക്ലസ്റ്ററിൽ സമാന്തര പ്രക്രിയ വഴിയാണ് ഗണന ശക്തി കിട്ടുന്നത്. ശാസ്ത്ര മേഖലയിലും, മറ്റു ആവശ്യങ്ങൾക്കും ഉപയോഗിക്കുന്ന സമാന്തര പ്രക്രിയയുടെ ഏറ്റവും ചിലവു കുറഞ്ഞ മാതൃകയാണ് ഇത്. ക്ലസ്റ്റർ കമ്പ്യൂട്ടിങ് ഉപയോഗിക്കുന്നതിനുള്ള പ്രധാന കാരണം തകരാറു പരിഹരിക്കുവാനുള്ള അവയുടെ കഴിവും, അതുവഴി കമ്പ്യൂട്ടേഷണൽ ഊർജ്ജ ലഭ്യത ഏതു സമയവും ഉറപ്പാക്കാമെന്നതുമാണ്.

ചിത്രം 11.6 ൽ ഒരു കമ്പ്യൂട്ടിങ് ക്ലസ്റ്റർ ചിത്രീകരിച്ചിരിക്കുന്നു.

ഇലക്ട്രോണിക് വാണിജ്യം (ഇ-കോമേഴ്സ്) മുതൽ ഉയർന്ന കാര്യക്ഷമത ആവശ്യമുള്ള ഡാറ്റാബേസ് ആപ്ലിക്കേഷനുകൾ വരെ ക്ലസ്റ്റർ കമ്പ്യൂട്ടറുകൾ പിന്തുണയ്ക്കുന്നു.

ക്ലസ്റ്റർ കമ്പ്യൂട്ടറുകളിൽ ഏറ്റവും വിപുലമായി ഉപയോഗിക്കുന്ന ഓപ്പറേറ്റിംഗ് സിസ്റ്റമാണ് ലിനക്സ്.

ചിത്രം 11.6 ഒരു കമ്പ്യൂട്ടിങ് ക്ലസ്റ്റർ

a. ക്ലസ്റ്റർ കമ്പ്യൂട്ടിങ്ങിന്റെ മേന്മകൾ

ക്ലസ്റ്റർ കമ്പ്യൂട്ടിങ് ഉപയോഗിക്കുന്നതിന്റെ മേന്മകൾ ഇവയാണ്.

വില നിർവഹണ അനുപാതം (Price-perfomrance Ratio): ക്ലസ്റ്റർ കമ്പ്യൂട്ടിങ് പ്രവർത്തന ചെലവ് ഗണ്യമായി കുറയ്ക്കുന്നു.

ലഭ്യത : ഒരു ഉപകരണം തകരാറായാലും സംവിധാനം മൊത്തമായി അപ്പോഴും പ്രവർത്തനക്ഷമമായിരിക്കും

വികസിപ്പിക്കൽ: ആവശ്യകതയനുസരിച്ചു പ്രോസസ്സറും നോഡും ക്ലസ്റ്ററുകളിലേക്ക് കൂട്ടിച്ചേർക്കാം.

b. ക്ലസ്റ്റർ കമ്പ്യൂട്ടിങ്ങിന്റെ പോരായ്മകൾ

ക്ലസ്റ്റർ കമ്പ്യൂട്ടിങ് ഉപയോഗിക്കുന്നത് മൂലം ഉള്ള പോരായ്മകൾ ഇവയാണ്.

പ്രോഗ്രാം എഴുതുമ്പോൾ ഉണ്ടാകുന്ന പ്രശ്നങ്ങൾ : വ്യത്യസ്ത കമ്പ്യൂട്ടറുകളിൽ വിവിധതരം സോഫ്റ്റ്‌വെയർ ഉപയോഗിക്കുകയാണെങ്കിൽ, അവ കൂട്ടിയോജിപ്പിച്ചു ഒന്നാക്കുമ്പോൾ ബുദ്ധിമുട്ടുകൾ ഉണ്ടാകാം.

തകരാറുകൾ കണ്ടെത്തുവാനുള്ള ബുദ്ധിമുട്ട്: ഒറ്റ അസ്തിത്വമായി കൈകാര്യം ചെയ്യുന്നതിനാൽ തകരാറുള്ള ഘടകങ്ങൾ കണ്ടെത്തുവാൻ ബുദ്ധിമുട്ടാണ്.

Beowulf Cluster എന്നത്, പേഴ്സണൽ കമ്പ്യൂട്ടറുകൾ ഉപയോഗിച്ച് നിർമ്മിച്ച ചെലവ് കുറഞ്ഞ ഒരു ക്ലസ്റ്റർ ആണ്. ഒരു ലാൻ(LAN) കണക്ഷനും കുറച്ചു പഴയ കമ്പ്യൂട്ടറുകളും ഉപയോഗിച്ച് ഇത് നിർമിക്കാം. ലിനക്സ്, സൊളാരിസ് തുടങ്ങിയ സ്വതന്ത്ര ഓപ്പറേറ്റിംഗ് സിസ്റ്റങ്ങളിലാണ് Beowulf Cluster ലെ കമ്പ്യൂട്ടറുകൾ പ്രവർത്തിക്കുന്നത്. ഉയർന്ന നിലവാരത്തിലുള്ള പ്രവർത്തനം കാഴ്ചവെച്ചില്ലെങ്കിലും ഇവയുടെ പ്രവർത്തനം ഒരു സാധാരണ കമ്പ്യൂട്ടറിന്റെ പ്രവർത്തനത്തെക്കാൾ പല മടങ്ങ് മികച്ചതാണ്.

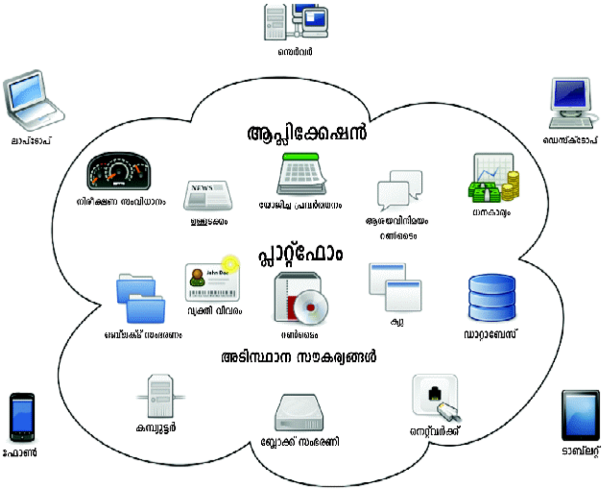
11.2.4 ക്ലൗഡ്കമ്പ്യൂട്ടിങ്

ക്ലൗഡ് കമ്പ്യൂട്ടിങ് പ്രചാരമേറി വരുന്ന ഒരു കമ്പ്യൂട്ടിങ് സാങ്കേതിക വിദ്യയാണ്. ഉപയോക്താവിന് ആവശ്യമുള്ള വിഭവങ്ങൾ, അടിസ്ഥാന സൗകര്യങ്ങൾ, സോഫ്റ്റ്‌വെയറുകൾ എന്നിവ പങ്കുവെക്കുക എന്നതാണ് ഇതിന്റെ പരമ പ്രധാനമായ ലക്ഷ്യം. ഇന്റർനെറ്റും കേന്ദ്രീകൃത വിദൂര സർവറുകളും ഉപയോഗിച്ച് ഡാറ്റയും ആപ്ലിക്കേഷനുകളും കാത്ത്സൂക്ഷിക്കുന്നു.

വിദൂര സ്ഥലത്ത് സംരക്ഷിച്ചിരിക്കുന്ന കമ്പ്യൂട്ടിങ് വിഭവങ്ങളെ ഒരു സേവനമായി ഉപയോക്താവിന് നൽകുക എന്നതാണ് ക്ലൗഡ് കമ്പ്യൂട്ടിങ് കൊണ്ട് ഉദ്ദേശിക്കുന്നത്. ഇ-മെയിൽ സേവനം ഒരു ക്ലൗഡ് കമ്പ്യൂട്ടിങ്ങിന് ഉദാഹരണമാണ്. വെബ് അധിഷ്ഠിത ഇ-മെയിൽ സേവനദാതാ

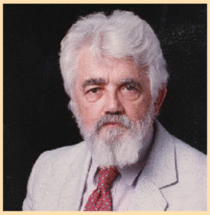
ക്കളായ യാഹൂ, ജിമെയിൽ, ഔട്ട് ലുക്ക് തുടങ്ങിയവ സ്വകാര്യ ഇമെയിൽ അക്കൗണ്ടുകൾ പരിപാലിക്കുന്നതിനാവശ്യമായ ഹാർഡ്‌വെയറും സോഫ്റ്റ്‌വെയറും സൂക്ഷിക്കുന്നു. ഇ-മെയിൽ അക്കൗണ്ട് ഉപയോഗിക്കേണ്ടി വരുമ്പോൾ വെബ് ബ്രൗസറിൽ ഇ-മെയിൽ ക്ലൈന്റ് തുറന്നു ലോഗിൻ ചെയ്ത് ഉപയോഗിക്കുന്നു. ഈ പ്രക്രിയ്ക്ക് ഇന്റർനെറ്റ് എപ്പോഴും ആവശ്യമാണ്. നമ്മുടെ ഇ-മെയിൽ സന്ദേശങ്ങളും, സന്ദേശങ്ങൾ കൈകാര്യം ചെയ്യുന്ന സോഫ്റ്റ്‌വെയറുകളും നമ്മുടെ കമ്പ്യൂട്ടറിൽ അല്ല സൂക്ഷിച്ചിരിക്കുന്നത്. ഇ-മെയിൽ സേവനദാതാക്കൾ ഇവയെ ക്ലൗഡിലാണ് സൂക്ഷിച്ചിരിക്കുന്നത്. എവിടെ നിന്നും ഏതു സമയത്തും ക്ലൗഡിൽ നിന്ന് ഇന്റർനെറ്റ് വഴി നമുക്ക് ഇവ ഉപയോഗിക്കാം.

ഇമെയിൽ സേവനം പോലെ, ഓഫീസ് സോഫ്റ്റ്‌വെയർ (വേർഡ് പ്രൊസസ്സർ , സ്പ്രെഡ് ഷീറ്റ് എന്നിവ) ഗ്രാഫിക് സോഫ്റ്റ്‌വെയർ തുടങ്ങിയവ ഇപ്പോൾ ക്ലൗഡ് വഴി ലഭ്യമാണ്. വ്യക്തികൾക്ക് ഈ സോഫ്ട്‌വെയറുകൾ സ്വകാര്യ ആവശ്യങ്ങൾക്കായി ഉപയോഗിക്കുവാനും, തനതു ഫയലുകൾ (ഡോക്യൂമെന്റുകൾ, സ്പ്രെഡ് ഷീറ്റുകൾ, ചിത്രങ്ങൾ തുടങ്ങിയവ) ക്ലൗഡിലെ സ്വകാര്യ സംഭരണ സ്ഥലത്തു സൂക്ഷിക്കുവാനും കഴിയും. ക്ലൗഡ് കമ്പ്യൂട്ടിങ്ങ് രീതി എന്നത് ഒരു കമ്പ്യൂട്ടിങ്ങ് മാതൃകയാണ്. ഇവിടെ കമ്പ്യൂട്ടിങ്ങ് ശക്തി, സംഭരണശേഷി, ശൃംഖല, സോഫ്റ്റ്‌വെയർ തുടങ്ങിയവയെ സംയോജിപ്പിച്ച്, വിദൂരതയിൽ ലഭ്യമാകുന്ന വിധത്തിൽ, ഇന്റർനെറ്റിലെ ഒരു സേവനമായി മാറ്റിയിരിക്കുന്നു. ലഭ്യമാകേണ്ട വിവരങ്ങൾ ആകാശത്ത് മേഘങ്ങൾ സ്ഥിതി ചെയ്യുന്നത് പോലുള്ള ഒരു ക്ലൗഡ് സംവിധാനത്തിൽ നിന്നോ വിദൂര സ്ഥലത്ത് നിന്നോ ലഭിക്കുന്നതിനാലാണ് ക്ലൗഡ് എന്ന് ഈ സംവിധാനത്തെ വിശേഷിപ്പിക്കുന്നത്. ക്ലൗഡ്കമ്പ്യൂട്ടിങ്ങ് ഉപയോഗിക്കണമെങ്കിൽ ഇന്റർനെറ്റ് സേവനവും ക്ലൗഡ് സേവനദാതാവ് നൽകുന്ന അക്കൗണ്ടും



ചിത്രം 11.7: ക്ലൗഡ് കമ്പ്യൂട്ടിങ്ങ്

1961ൽ, നിർമ്മിത ബുദ്ധി (artificial intelligence) വന്ന് പത്തു വർഷത്തിന് ശേഷം, പ്രൊഫ. ജോൺ മക്കാർത്തിയാണ് (John Mccarthy) ഹാർഡ്‌വെയറും ആപ്പിക്കേഷനുകളും പങ്കിടാം എന്ന ആശയം കൊണ്ടുവന്നത്. ജലമോ, പാചക വാതകമോ എങ്ങനെ വിതരണം ചെയ്യുന്നുവോ, അതുപോലെ സൗജന്യമായോ കുറഞ്ഞ ചിലവിലോ കമ്പ്യൂട്ടിങ്ങ് ശക്തിയും നൽകുവാൻ കഴിയുമെന്ന് അദ്ദേഹം ഭാവനയിൽ കണ്ടു. salesforce.com എന്ന ലളിതമായ വെബ് സൈറ്റ് വഴി 1999ൽ ഉപയോക്താക്കൾക്ക് എന്റർപ്രൈസ് ആപ്ലിക്കേഷൻ നൽകിയതാണ് ക്ലൗഡ്കമ്പ്യൂട്ടിങ്ങ് ചരിത്രത്തിലെ ആദ്യ നാഴികക്കല്ല്. ഇത് വിദഗ്ധർക്കും മുൻനിര സോഫ്റ്റ്‌വെയർ സ്ഥാപനങ്ങൾക്കും അവരുടെ ഉൽപ്പന്നങ്ങൾ ഇന്റർനെറ്റിലൂടെ നൽകുവാൻ വഴിയൊരുക്കി. ഇപ്പോൾ ക്ലൗഡ് കമ്പ്യൂട്ടിങ്ങ് എന്നത് വിവര സാങ്കേതിക മേഖലയിൽ ഉയർന്നു വന്നുകൊണ്ടിരിക്കുന്ന പ്രധാന വിഷയങ്ങളിൽ ഒന്നാണ്.



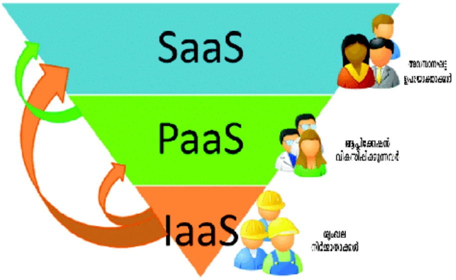
ജോൺ മക്കാർത്തി (19272011)

ആവശ്യമാണ്. ക്ലൗഡ് കമ്പ്യൂട്ടിങ് ഒരു വരിസംഖ്യ അധിഷ്ഠിത സേവനമാണ്, അവിടെ കമ്പ്യൂട്ടിങ്ങ് ഉപകരണങ്ങളും, ശൃംഖലയുമായി ബന്ധിപ്പിച്ചിട്ടുള്ള സംഭരണ സ്ഥലവും നമുക്ക് ലഭ്യമാകുന്നു. ചിത്രം 11.7 ൽ ക്ലൗഡ് കമ്പ്യൂട്ടിങ്ങിന്റെ രേഖാചിത്രം കൊടുത്തിരിക്കുന്നു.

a. ക്ലൗഡ് സേവന മാതൃകകൾ

ക്ലൗഡ് സേവനദാതാക്കൾ തരുന്ന സേവനങ്ങളെ പ്രധാനമായും മൂന്ന് വിഭാഗങ്ങളായി തരം തിരിക്കാം

സോഫ്റ്റ്‌വെയർ ഒരു സേവനമായി (SaaS), പ്ലാറ്റ്ഫോം ഒരു സേവനമായി (PaaS), അടിസ്ഥാനസൗകര്യം ഒരു സേവനമായി (IaaS) എന്നിവയാണവ. ഇവയെ കുറിച്ച് നമുക്ക് വിശദമായി ചർച്ച ചെയ്യാം (ചിത്രം 11.8 നോക്കുക)



ചിത്രം 11.8: അടിസ്ഥാന ക്ലൗഡ് കമ്പ്യൂട്ടിങ്ങ് മാതൃകകൾ

1 സോഫ്റ്റ്‌വെയർ ഒരു സേവനമെന്ന നിലയിൽ (Software as a Service)

SaaS സേവനദാതാക്കൾ വരിക്കാർക്ക് വിഭവങ്ങളും ആപ്ലിക്കേഷനുകളും സേവനമായി നൽകുന്നു. ഉപയോക്താവിന്റെ ആവശ്യാനുസരണം സമ്പൂർണരൂപത്തിലുള്ള ആപ്ലിക്കേഷനുകൾ ഒരു സേവനമായി ലഭ്യമാകുന്നു. ക്ലൗഡിൽ സൂക്ഷിച്ചിട്ടുള്ള ആപ്ലിക്കേഷനുകളും സേവനങ്ങളും ഉപയോഗിക്കുവാൻ ഉള്ള അവകാശം ഉപയോക്താക്കൾ വാങ്ങുന്നു. എല്ലാ ജിമെയിൽ അക്കൗണ്ട് ഉപയോക്താക്കൾക്കും ഗൂഗിൾ ഡോക് എന്ന ഗൂഗിളിന്റെ സേവനം സൗജന്യമായി ഉപയോഗിക്കുവാൻ കഴിയുന്നു. ഇവിടെ ആപ്ലിക്കേഷന്റെ ഒരു പതിപ്പ് ക്ലൗഡിൽ പ്രവർത്തിക്കുകയും ഒന്നിലധികം ഉപയോക്താക്കൾക്ക് ഇതിന്റെ സേവനം ലഭ്യമാവുകയും ചെയ്യുന്നു. നമ്മുടെ കമ്പ്യൂട്ടറുകൾ, ലാപ്ടോപ്പുകൾ, മൊബൈലുകൾ എന്നിവകളിൽ ആപ്ലിക്കക്കേഷനുകളുടെ പതിപ്പുകൾ ഇൻസ്റ്റാൾ ചെയ്യേണ്ട ആവശ്യം ഈ ക്ലൗഡ് സേവനം കൊണ്ട് ഇല്ലാതാകുന്നു. ഗൂഗിൾ ഡോക്സിന്റെ (Google Docs) കാര്യത്തിൽ, വേർഡ് പ്രോസസ്സറിന്റെ ഒരു പകർപ്പ് മാത്രം ക്ലൗഡിൽ പ്രവർത്തിക്കുന്നു. ഇതു Saas വഴി, ഉപയോക്താവിന് സ്വന്തം കമ്പ്യൂട്ടറിൽ ഈ സോഫ്റ്റ്‌വെയർ ഇൻസ്റ്റാൾ ചെയ്യാതെ തന്നെ അതു ഉപയോഗിക്കാം ക്ലൗഡിൽ നിന്ന് ഉപയോഗിക്കുന്നത് കാരണം ഈ സോഫ്റ്റ്‌വെയറിന്റെ ഒരേ പതിപ്പ് തന്നെ നിങ്ങളുടെ എല്ലാ കമ്പ്യൂട്ടറുകളിലും വളരെ എളുപ്പത്തിൽ ലഭ്യമാകുന്നു. പണം നൽകിയുള്ള സേവനങ്ങൾ, വില നൽകിയതിനുശേഷമേ ഉപയോക്താക്കൾക്ക് ലഭ്യമാകൂ.

സോഫ്റ്റ്‌വെയർ ഒരു സേവനമായി കണക്കാക്കുന്ന കരാറിൽ, ഉപയോക്താവിന് വളരെ കുറഞ്ഞ തോതിലുള്ള നിയന്ത്രണമേ ക്ലൗഡിൽ ഉള്ളൂ. സർവറുകൾക്കു വേണ്ടിയോ സോഫ്റ്റ്‌വെയർ ലൈസൻസിനു വേണ്ടിയോ വലിയ തോതിലുള്ള നിക്ഷേപം ഉപഭോക്താവ് നടത്തേണ്ടതില്ല. അതുപോലെ സോഫ്റ്റ്‌വെയറിന്റെ ഒരു പതിപ്പ് മാത്രം സൂക്ഷിക്കുകയും പരിപാലിക്കുകയും ചെയ്യേണ്ടതുള്ളൂ എന്നതിനാൽ സേവനദാതാക്കൾക്കും ചെലവ് വളരെ കുറവാണ്. ഇക്കാലത്തു ഇത്തരം സേവനങ്ങൾ പല സ്ഥാപനങ്ങളും നൽകുന്നുണ്ട്. അഡോബിന്റെ (Adobe) ചിത്ര പ്രവർത്തന സോഫ്റ്റ്‌വെയറായ (Image Processing Software) ക്രിയേറ്റീവ് ക്ലൗഡ് (Creative Cloud) മൈക്രോസോഫ്റ്റിന്റെ ഓഫീസ് സോഫ്റ്റ്‌വെയറായ ഓഫീസ് 365, facebook.com തുടങ്ങിയവ ഉദാഹരണങ്ങളാണ്.

2. പ്ലാറ്റ്ഫോം ഒരു സേവനമെന്ന നിലയിൽ (Platform as a Service) (PaaS)

സോഫ്റ്റ്‌വെയർ ഒരു സേവനമായി ഉപയോഗിക്കുന്ന സംവിധാനത്തിൽ നിന്നും ഒരുപടി ഉയർന്ന തലമാണ് പ്ലാറ്റ്ഫോം ഒരു സേവനമായി നൽകുന്നത്. പ്ലാറ്റ്ഫോം (PaaS) ഒരു സേവനമായി നൽകുന്ന സേവനദാദാക്കൾ, വരിക്കാർക്ക് ഇന്റർനെറ്റ് വഴി ആപ്ലിക്കേഷനുകൾ വികസിപ്പിക്കുവാനും പ്രവർത്തിപ്പിക്കുവാനുമുള്ള അടിസ്ഥാന ഘടകങ്ങൾ നൽകുന്നു. മറ്റൊരുതരത്തിൽ പറഞ്ഞാൽ, ഒരു ആപ്ലിക്കേഷൻ രൂപകൽപ്പന ചെയ്യുവാനും, വികസിപ്പിക്കുവാനും, നിർമ്മിക്കുവാനും, പരിശോധിക്കുവാനുമുള്ള അടിസ്ഥാന സൗകര്യം ഇത്തരം ക്ലൗഡ് നൽകുന്നു. സേവന ദാതാവ് നൽകുന്ന അടിസ്ഥാന സൗകര്യങ്ങൾ ഉപയോഗിച്ച് ഉപയോക്താവിനു തനതായ ആപ്ലിക്കേഷൻ നിർമ്മിക്കുവാനുള്ള സ്വാതന്ത്ര്യം ഉണ്ട്. ആപ്ലിക്കേഷനുകളുടെ പരിപാലനത്തിനും വികാസത്തിനുമാവശ്യമായ ഘടകങ്ങൾ നൽകുവാനായി, പ്ലാറ്റ്ഫോം സേവനദാതാക്കൾ ലാംപ് (LAMP) ലിനക്സ് (LINUX), അപ്പാച്ചെ (APPACHE), മൈ എസ്ക്യൂഎൽ (MySQL), പി എച് പി (PHP) എ എസ് പി ഡോട്ട് നെറ്റ് (ASP.NET), പൈതൺ (PYTHON) തുടങ്ങിയവ ഒഎസ് (OS) ന്റെയും ആപ്ലിക്കേഷൻ സർവറുകളുടെയും സംയോജിത രൂപം വാഗ്ദാനം ചെയ്യുന്നു. ഗൂഗിളിന്റെ ആപ് എൻജിൻ (App Engine), മൈക്രോസോഫ്ട് ആഷർ (Azure), ഫോഴ്സ് ഡോട്ട് കോം (Force.com) തുടങ്ങിയവ പ്ലാറ്റ്ഫോം സേവനങ്ങൾക്ക് ഉദാഹരങ്ങളാണ്.

3. അടിസ്ഥാന സൗകര്യങ്ങൾ ഒരു സേവനമെന്ന നിലയിൽ (Infrastructure as a Service(IaaS))

പേര് സൂചിപ്പിക്കുന്നത് പോലെ, കമ്പ്യൂട്ടേഷണൽ അടിസ്ഥാന സൗകര്യങ്ങൾ നൽകുവാനാണ് ഐഎഎസ് (IaaS) ഉടമ്പടി മുൻഗണ നൽകുന്നത്. അടിസ്ഥാന സംഭരണശേഷി, കമ്പ്യൂട്ടിങ് ശേഷി എന്നിവ നിലവാരമുള്ള സർവീസുകളായി ഐഎഎസ് (IaaS) ശൃംഖല വഴി നൽകുന്നു. സർവർ, സംഭരണികൾ, ശൃംഖല ഉപകരണങ്ങൾ, കേന്ദ്രീകൃത ഡാറ്റ സംഭരണികൾ തുടങ്ങിയവ ജോലിഭാരം കൈകാര്യം ചെയ്യുവാനായി സജ്ജമാക്കിയിരിക്കുന്നു. ഈ അടിസ്ഥാന സൗകര്യത്തിൽ ഉപഭോക്താക്കൾ അവരവരുടെ സോഫ്റ്റ് വെയറുകൾ സജ്ജീകരിച്ചിരിക്കണം. ആമസോൺ വെബ് സർവീസ് ജോയന്റ് (Joyent), എ ടി ആൻഡ് ടി (AT & T), ഗോ ഗ്രിഡ് (GoGrid) എന്നിവ ചില ഉദാഹരങ്ങളാണ്.

b. ക്ലൗഡ് കമ്പ്യൂട്ടിങ്ങിന്റെ മേന്മകൾ

ക്ലൗഡ് കമ്പ്യൂട്ടിങ്ങ് ഉപയോഗിക്കുന്നത് കൊണ്ടുള്ള ചില മേന്മകൾ ചുവടെ ചേർക്കുന്നു.

ചെലവ് ലാഭിക്കാം : സ്ഥാപനങ്ങൾക്ക് അവരുടെ മൂലധനചിലവ് കുറയ്ക്കുകയും പ്രവർത്തന ചിലവ് ഉപയോഗപ്പെടുത്തി കമ്പ്യൂട്ടിങ്ങ് ശേഷി വർദ്ധിപ്പിക്കുകയും ചെയ്യാം.

വികസന സാധ്യതയും/വഴക്കവും: സ്ഥാപനങ്ങൾക്ക് അവരുടെ പ്രവർത്തനങ്ങൾ ചെറിയ അളവിൽ തുടങ്ങി വളരെ വേഗത്തിൽ വലിയ അളവിലേക്കു വികസിപ്പിക്കുവാനും ആവശ്യം വന്നാൽ പഴയ അവസ്ഥയിലേക്ക് മടങ്ങുവാനും കഴിയുന്നു. അതോടൊപ്പം തന്നെ കമ്പനികൾക്കു വളരെ തിരക്കുള്ള സമയത്ത് അധിക വിഭവങ്ങൾ ഉപയോഗിച്ച് ഉപഭോക്താക്കളെ തൃപ്തിപ്പെടുത്തുവാനും ക്ലൗഡ് കമ്പ്യൂട്ടിങ്ങ് രീതി സഹായിക്കുന്നു.

വിശ്വാസ്യത : ഒന്നിൽ കൂടുതൽ മേഖലകളിലെ സർവീസുകൾ ഉപയോഗിക്കുന്നതിനാൽ തകരാറുകളിൽ നിന്ന് തിരിച്ചുവരുവാൻ സഹായിക്കുന്നു.

പരിപാലനം : ക്ലൗഡ് സേവനദാതാക്കൾ തന്നെ ഈ സംവിധാനത്തിന്റെ പരിപാലനം നടത്തുന്നതിനാൽ ഉപഭോക്താവ് ഇതിന്റെ പരിപാലനം ഏറ്റെടുക്കേണ്ടതില്ല.

യാത്രയിലും ലഭ്യമാകുന്നു : ഈ സംവിധാനം എവിടെ നിന്നും ഉപയോഗിക്കുവാൻ കഴിയുന്നതിനാൽ ജോലി സംബന്ധമായി യാത്ര ചെയ്യുന്ന ജീവനക്കാർക്ക് അവരുടെ സ്ഥാപനങ്ങളിലെ ജോലി തടസ്സം കൂടാതെ നിർവഹിക്കുവാൻ കഴിയുന്നു.

c. ക്ലൗഡ് കമ്പ്യൂട്ടിങ്ങിന്റെ പോരായ്മകൾ

ക്ലൗഡ് കമ്പ്യൂട്ടിങ്ങ് രീതി ഉപയോഗിക്കുന്നവർ അഭിമുഖീകരിക്കേണ്ടിവരുന്ന ചില വെല്ലുവിളികൾ ഇവയാണ്.

സുരക്ഷയും സ്വകാര്യതയും: വിവരങ്ങൾ പൊതു ശൃംഖലകൾവഴി കൈമാറ്റം ചെയ്യുമ്പോഴും, പൊതു സംഭരണ വ്യവസ്ഥയിൽ സൂക്ഷിക്കുമ്പോഴും, ഇവ മോഷ്ടിക്കുവാനോ, തകരാർ വരുത്തുവാനോ സാധ്യതയുണ്ട്.

മാനദണ്ഡങ്ങളുടെ കുറവ്: പൊതുവായ മാനദണ്ഡങ്ങൾ ഒന്നും ക്ലൗഡിന് ഇല്ലാത്തതിനാൽ, മിക്ക ക്ലൗഡുകളും പരസ്പരം സഹകരിച്ച് പ്രവർത്തിക്കാത്തവയാണ്.

നമുക്ക് വിലയിരുത്താം

1. ക്ലസ്റ്റർ കമ്പ്യൂട്ടിങ്ങിന്റെ മേന്മകൾ എഴുതുക?
2. എന്താണ് ക്ലൗഡ് കമ്പ്യൂട്ടിങ്ങ്?
3. വിവിധ ക്ലൗഡ് സേവന മാതൃകളുടെ പേരെഴുതുക?
4. എന്താണ് സോഫ്റ്റ്‌വെയർ ഒരു സേവനം എന്നാൽ എന്ത്?
5. ഏതെങ്കിലും രണ്ട് എസ് എ എ എസ് (SaaS) സേവനദാതാക്കളുടെ പേരുകൾ എഴുതുക ?
6. ഏതെങ്കിലും രണ്ട് ഐ എ എ എസ് (IaaS) സേവനദാതാക്കളുടെ പേരുകൾ എഴുതുക?

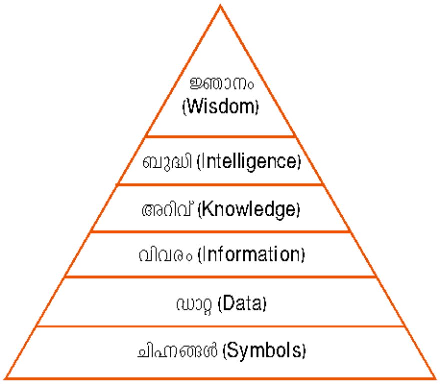
11.3 നിർമിത ബുദ്ധി (Artificial Intelligence)

നിർമിത ബുദ്ധി (AI) എന്നത് കമ്പ്യൂട്ടർ ശാസ്ത്രത്തിൽ ഏറ്റവും നൂതനമായ മേഖലയാണ്. 1950 ലാണ് അലൻ ടൂറിംഗ് ഇതിന് ആദ്യ നിർവചനം നൽകിയത്. മനുഷ്യ മസ്തിഷ്കത്തിന്റെ പ്രവർത്തനത്തെ എങ്ങനെ ഒരു യന്ത്രത്തിന് അനുകരിക്കാമെന്ന് അലൻ ട്യൂറിംഗ് പഠിച്ചു. ഇദ്ദേഹത്തിന്റെ പഠന ഫലമായി AI മേഖലയിലെ ആദ്യ പ്രസിദ്ധീകരണം, ഇന്റലിജന്റ് മെഷിനറി, എന്ന പേരിൽ പ്രസിദ്ധീകരിച്ചു. 1956 ൽ ജോൺ മക്കാർത്തി സംഘടിപ്പിച്ച ഡാർട് മൗത് കോൺഫറൻസിൽ ആണ് ആദ്യമായി നിർമിത ബുദ്ധി എന്ന പദം ഉപയോഗിച്ചത്. AI ഇപ്പോൾ ധാരാളം ഉപമേഖലകളിലും ശ്രദ്ധ കേന്ദ്രീകരിച്ചിരിക്കുന്നു. പൊതു ഉദ്ദേശ്യ മേഖലകളായ ഗ്രഹണശക്തി (ഉൾകാഴ്ച) [Perception], യുക്തിപരമായ ന്യായവാദങ്ങൾ [Logical reasoning] തുടങ്ങിയവയിലും നിർദിഷ്ട കൃത്യങ്ങളായ ചതുരംഗക്കളി, ഗണിത സിദ്ധാന്തങ്ങൾ തെളിയിക്കുക, കമ്പ്യൂട്ടറിന്റെ വിഷൻ (കാഴ്ച ശക്തി) [Computer Vision], സ്വാഭാവിക ഭാഷാ പ്രക്രിയ (Natural Language Processing), വൈദ്യ പരിശോധന തുടങ്ങിയവയിലും നിർമിത ബുദ്ധി ശ്രദ്ധ കേന്ദ്രീകരിച്ചിരിക്കുന്നു.

പലപ്പോഴും ഈ മേഖലയെ കണ്ടിരിക്കുന്നത് ഗവേഷണത്തിന്റെ പരസ്പര ബന്ധിത മേഖലയായിട്ടാണ്. ഇവിടേയ്ക്ക് മറ്റു ഗവേഷണ മേഖലയിൽ നിന്നുള്ള ഗവേഷകരും കടന്നുവരുന്നുണ്ട്, അവിടെ അവർ പ്രവർത്തിക്കുന്ന മേഖലയ്ക്ക് ആവശ്യമായ ഉപകരണങ്ങൾ കണ്ടെത്തുകയും പദസംഗ്രഹങ്ങളെ ക്രമപ്പെടുത്തി പ്രവർത്തനങ്ങളെ യന്ത്രവൽക്കരിക്കുകയും ചെയ്യുന്നു .

അതുപോലെ മനുഷ്യബുദ്ധി അധികമായി പ്രയോജനപ്പെടുത്തേണ്ട ഏതൊരു മേഖലയിലും AI ഗവേഷകർക്ക് അവരുടെതായ രീതി തെരഞ്ഞെടുക്കാവുന്നതാണ്. ഒരർഥത്തിൽ ശരിക്കും ഇത് ഒരു സാർവത്രിക മേഖലയാണ്.

ചിത്രം 11.9 ൽ അറിവിന്റെ വ്യത്യസ്തതലങ്ങളെ പിരമിഡ് രൂപത്തിൽ ചിത്രീകരിച്ചിരിക്കുന്നു. ഇവയിൽ അറിവും ബുദ്ധിയും ആണ് എ ഐ യുടെ പ്രധാന പഠന മേഖലകൾ. ഇവയെ കമ്പ്യൂട്ടറിലേക്ക് സന്നിവേശിപ്പിച്ച് സിസ്റ്റത്തെ അറിവുള്ളതാക്കി മാറ്റുന്നു. ജീവശാസ്ത്ര വ്യവസ്ഥയുടെ വിവിധ കമ്പ്യൂട്ടിങ്ങ് മാതൃകകൾ വികസിപ്പിക്കുന്നതിൽ വിജയം കണ്ടെത്തിയിട്ടുണ്ട്. ഉദാ: ജീവനാഡീകോശത്തിന്റെ മാതൃകകൾ. എന്നാൽ മനുഷ്യ ബുദ്ധിയുടെ ഭാഗങ്ങളായ സ്വയമറിവ്, പ്രബുദ്ധത, മനോവികാരം തുടങ്ങിയ അതിസങ്കീർണങ്ങളായ പ്രശ്ന മാതൃകകൾ സൃഷ്ടിക്കുവാൻ ഇന്നും കഴിഞ്ഞിട്ടില്ല.



ചിത്രം 11.9: അറിവിന്റെ പിരമിഡ്

മനുഷ്യ മസ്തിഷ്കത്തെ അനുകരിക്കുവാൻ കഴിയുന്ന രീതിയിൽ ഒരു യന്ത്രം നിർമ്മിക്കുവാൻ കഴിയുമെന്ന് അലൻ ട്യൂറിംഗ് വിശ്വസിച്ചു. മനുഷ്യമസ്തിഷ്കത്തിന് കഴിയുന്ന എല്ലാ കാര്യങ്ങളും മികച്ച രീതിയിൽ രൂപകൽപ്പന ചെയ്ത ഒരു കമ്പ്യൂട്ടറിന് ഇത് സാധിക്കുമെന്ന് അലൻ ട്യൂറിംഗ് ശക്തമായി വിശ്വസിച്ചു. ഒരു ദീർഘ വീക്ഷണമായി തന്നെ അദ്ദേഹത്തിന്റെ വാക്കുകൾ 50 വർഷത്തിന് ശേഷവും നില നിൽക്കുന്നു.

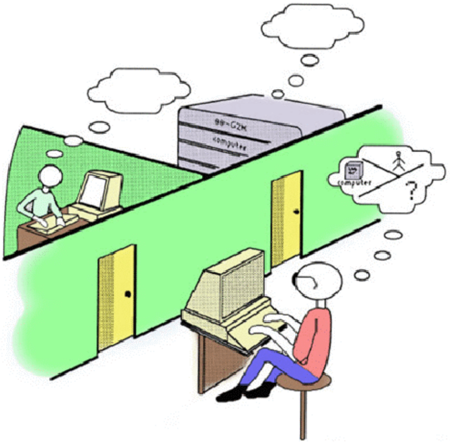
എല്ലാ ബുദ്ധിപരമായ ഘടനയ്ക്കും പിന്നിൽ പ്രവർത്തിക്കുന്ന ശക്തി അറിവാണ്. അറിവിന്റെ പിരമിഡിൽ അടിസ്ഥാന ഘടകമായി കൊടുത്തിരിക്കുന്നത് ചിഹ്നങ്ങൾ ആണ്. ചിഹ്നങ്ങളുടെ ഒരു ശേഖരമാണ് ഡാറ്റ. ഡാറ്റയെ പ്രോസസ്സ് ചെയ്യുമ്പോൾ വിവരങ്ങൾ കിട്ടുന്നു, വിവരങ്ങളെ ചിട്ടപ്പെടുത്തി വെയ്ക്കുമ്പോൾ അത് അറിവായി മാറുന്നു. തീരുമാനം എടുക്കുവാൻ വിവരം സഹായിക്കുന്നു. നേടിയ അറിവിൽ നിന്നും പ്രയോജനകരമായ അനുമാനത്തിൽ എത്തുവാനുള്ള കഴിവിനെ ബുദ്ധി എന്ന് പറയുന്നു. ആഗ്രഹിക്കുന്ന ലക്ഷ്യം നേടിയെടുക്കുന്നതിന് ബുദ്ധിയെ നയിക്കുന്ന പക്വമായ മനസ്സാണ് ജ്ഞാനം.

അറിവിനെ എങ്ങനെ സിസ്റ്റത്തിൽ ഉൾക്കൊള്ളിക്കാമെന്നും സാധാരണ പ്രോഗ്രാമിങ് ഭാഷാ ഉപയോഗിച്ചു പരിഹരിക്കുവാൻ സാധിക്കാത്ത യഥാർത്ഥ ജീവിത പ്രശ്നങ്ങളെ ബുദ്ധിപരമായി എങ്ങനെ പരിഹരിക്കാമെന്നും ഇവിടെ നാം പഠിക്കുന്നു. മനുഷ്യന്റെ യുക്തിക്കു സമാനമായ പ്രവർത്തനങ്ങളെ പ്രയോജനപ്പെടുത്തി അതിസങ്കീർണമായ പ്രശ്നപരിഹാരത്തിനുള്ള കമ്പ്യൂട്ടർ പ്രോഗ്രാമുകൾ വികസിപ്പിക്കുക എന്നതാണ് AI യുടെ അടിസ്ഥാനം.

നിർമിത ബുദ്ധിയോടുള്ള ടൂറിംഗ് പരീക്ഷണ സമീപനം (Turing Test approach to AI)

അലൻ ട്യൂറിംഗ് നിർദേശിച്ച ട്യൂറിംഗ് പരീക്ഷണം ബുദ്ധിശക്തിയുടെ തൃപ്തികരമായ പ്രവർത്തന രീതി നിർവചിക്കുവാനുള്ള രൂപകല്പനയാണ്. ട്യൂറിംഗ് ടെസ്റ്റ്ന്റെ വിശദാംശങ്ങൾ പതിനൊന്നാം ക്ലാസ്സിലെ കമ്പ്യൂട്ടർ സയൻസ് പാഠ പുസ്തകത്തിൽ (അദ്ധ്യായം ഒന്ന്) വിശദീകരിച്ചിട്ടുണ്ട്. ഒരു ലക്ഷ്യം നേടിയെടുക്കുന്നതിനുവേണ്ടിയുള്ള എല്ലാ പ്രവർത്തനങ്ങളും മനുഷ്യ സാധ്യമായ തലത്തിൽ നിർവഹിക്കാനുള്ള ശേഷിയെയാണ് ടൂറിങ്ങ് കമ്പ്യൂട്ടറുകളുടെ ബൗദ്ധിക ചേഷ്ട (Intelligence behaviour) എന്നു നിർവഹിക്കുന്നത്.

ഈയൊരു കഴിവ് ആർജിക്കുന്നതു വഴി ചോദ്യ കർത്താവിനെ വിഡ്ഢിയാക്കാനും കമ്പ്യൂട്ടറിനു കഴിയും. ഇത് ഒരു ടൈപ്പ് റൈറ്ററിന്റെ സഹായ ത്താൽ മനുഷ്യൻ കമ്പ്യൂട്ടറിനെ ചോദ്യം ചെയ്യുന്ന ഒരു പരീക്ഷണമാണ് എന്ന് പറയാം. കമ്പ്യൂട്ടർ ഈ പരീക്ഷ വിജയിക്കണമെങ്കിൽ, തിരികെ ലഭിക്കുന്ന മറുപടിയിൽ നിന്ന് ചോദ്യകർത്താവിനു കമ്പ്യൂട്ട റിനെയും മനുഷ്യനെയും തമ്മിൽ വേർതിരിച്ചറിയു വാൻ പറ്റാത്ത അവസ്ഥ ഉണ്ടാകണം. ട്യൂറിംഗ് ടെസ്റ്റിന്റെ ചിത്രീകരണം ചിത്രം 11.10ൽ നൽകിയിരി ക്കുന്നു. ഇന്നും, ട്യൂറിംഗ് ടെസ്റ്റ് എന്ന് പറയുന്നത് ഒരു യന്ത്രം ബുദ്ധിയുള്ളതാണ് എന്ന് തെളിയിക്കു വാനുള്ള അന്തിമ പരീക്ഷണമായി കരുതുകയും, ഈ പരീക്ഷണം വിജയിക്കുന്നതിനായി കമ്പ്യൂട്ടറിന് പോഗ്രാം വഴി ധാരാളം കൃത്യങ്ങൾ നൽകുകയും ചെയ്യുന്നു.



ചിത്രം 11.10: ടൂറിംഗ് പരീക്ഷണം

അക്കാര്യത്തിൽ കമ്പ്യൂട്ടറിനു താഴെപറയുന്ന കഴിവുകൾ കൂടി ഉണ്ടായിരിക്കണം.

- **സ്വാഭാവിക ഭാഷ പ്രക്രിയ (Natural Language Processing (NLP)):** ഇംഗ്ലീഷ് ഭാഷയിൽ (അല്ലങ്കിൽ മറ്റു ഭാഷകളിൽ) വിജയകരമായി ആശയവിനിമയം നടത്തുവാനുള്ള കഴിവ് ആർജിക്കുക. സ്വയം സംഭാഷണം തിരിച്ചറിയൽ (Automatic Speech Recognition), ശബ്ദ സമന്വയം (Speech Synthesis), യന്ത്ര പരിഭാഷ (Machine Translation), കയ്യക്ഷരം തിരിച്ച റിയൽ (Handwritten Character Recognition) തുടങ്ങിയവയൊക്കെ NLP യുമായി ബന്ധപ്പെട്ട പ്രായോഗിക ഉപയോഗങ്ങളാണ്. ഒരാളിന്റെ സ്വാഭാവിക ഭാക്ഷയിൽ കമ്പ്യൂട്ടറുമായി ആശയവിനിമയം നടത്തുക എന്നതാണ് ഇത്കൊണ്ട് ഉദ്ദേശിക്കുന്നത്.

- **അറിവിന്റെ പ്രതിനിധീകരണം (Knowledge Representation)**: ചോദ്യം ചെയ്യലിന് മുൻപോ ഇടയിലോ മനുഷ്യ പരിജ്ഞാനം ഉൾപ്പെടുത്തുന്നതിന്

- **സ്വയം വിചിന്തനം (Automated Reasoning):** ആർജിച്ച അറിവുകളിൽ നിന്ന് ചോദ്യങ്ങൾക്ക് ഉത്തരം നൽകുവാനും പുതിയ നിഗമനങ്ങളിൽ എത്തുവാനുമുള്ള കഴിവ്

- **യന്ത്രപഠനം (Machine Learning) :** പുതിയ സാഹചര്യങ്ങളുമായി പൊരുത്തപ്പെടുവാനും പുതിയ മാതൃകകൾ കണ്ടെത്തുവാനും അവലംബിക്കുവാനുമുള്ള കഴിവ്.

എങ്ങനെ ഒരു ഉപയോക്താവിനോട് പ്രാദേശിക ഭാഷയിലൂടെ സംവദിക്കുന്നുവെന്നും എങ്ങനെ ഭാഷ പ്രക്രിയ പ്രവർത്തിക്കുന്നുവെന്നും ഒരു സാമർഥ്യമുള്ള സംവിധാനം വിശദീകരിക്കേണ്ടി വരും. യന്ത്രങ്ങൾ മനുഷ്യനെപോലെ പെരുമാറുമ്പോൾ, നിർമിത ബുദ്ധിയുടെ പ്രോഗ്രാമുകൾ പ്രാഥമികമായി നേരിടുന്ന വെല്ലുവിളി, ഈ പ്രോഗ്രാമുകൾ മനുഷ്യന്റെ ആശയ വിനിമയത്തിലെ സ്വാഭാവിക രീതികൾ അനുസരിച്ചു പ്രവർത്തിക്കുകയും അവയെ സ്വയം മനസ്സിലാക്കുവാൻ പര്യാപ്തമാക്കുകയും ചെയ്യുക എന്നതാണ്. മനുഷ്യ മാതൃകകളെ അനുകരിച്ചോ അല്ലാതെയോ ആകാം ഇത്തരം സിസ്റ്റങ്ങളുടെ അടിസ്ഥാനപരമായ നിർമ്മിതിയും ന്യായവാദങ്ങളും. കമ്പ്യൂട്ടറിന്റെ സമ്പൂർണമായ ട്യൂറിംഗ് ടെസ്റ്റിന്റെ വിജയത്തിന്, ചുവടെ ചേർക്കുന്ന സവിശേഷതകളും കൂടി ആവശ്യമാണ്

- **കമ്പ്യൂട്ടർ ദൃഷ്ടി (Computer Vision):** വസ്തുക്കളെ നിരീക്ഷിക്കുവാനുള്ള കഴിവാണിത്. ഒരു യന്ത്രത്തിന് കാണുവാനുള്ള കഴിവ് ഉണ്ടാകണമെങ്കിൽ, ചിത്രങ്ങൾ ഗ്രഹിക്കുക, പരിവർത്തനം നടത്തുക, പ്രക്രിയയ്ക്ക് വിധേയമാക്കുക, പരിശോധിക്കുക, മനസ്സിലാക്കുക, നിർവചിക്കുക തുടങ്ങിയ പ്രവർത്തനങ്ങൾ കൂടി നടത്തേണ്ടി വരും.

- **റോബോട്ടിക് പ്രവർത്തനങ്ങൾ (Robotic Activities):** യന്ത്രത്തെ ചലിപ്പിക്കുക എന്നതാണ് ഇത്കൊണ്ട് ഉദ്ദേശിക്കുന്നത്. റോബോട്ടിനെ സാമർഥ്യം ഉള്ളതാക്കി മാറ്റുവാൻ, അതിനു ബുദ്ധി നൽകുക എന്നതാണ് പ്രഥമ കർത്തവ്യം. മാറിക്കൊണ്ടിരിക്കുന്ന പരിസ്ഥിതിയെ തരണം ചെയ്യുവാനായി റോബോട്ടുകളിൽ ചുറ്റുപാടുകളെ നിരീക്ഷിക്കുവാനുള്ള സെൻസറുകൾ ഉൾപ്പെടുത്തുകയും, ഗ്രഹിക്കുന്ന തരംഗങ്ങളെ അതിന്റെ ബുദ്ധിപരമായി നിയന്ത്രണ ഭാഗത്തു എത്തിക്കുകയും വേണം.

ജീവശാസ്ത്ര മാതൃകകളെ അനുകരിച്ചും, വിവിധ പ്രക്രിയകകൾ ഉപയോഗപ്പെടുത്തിയും, സ്വാഭാവികമായ ബുദ്ധിയെ ഉത്തേജിപ്പിച്ചും ധാരാളം വിജയങ്ങൾ ഈ മേഖലയിൽ നേടുവാൻ കഴിഞ്ഞിട്ടുണ്ട്. ആർട്ടിഫിഷൽ ന്യൂറൽ നെറ്റ്‌വർക്ക് (Artificial Intelligent Neural Network), ഇവല്യൂഷനറി കമ്പ്യൂട്ടേഷൻ (Evolutionary Computation), സ്വാം ഇന്റലിജൻസ് (Swarm Intelligence), ഫസീ ലോജിക് (Fuzzy Logic) തുടങ്ങിയവ ബുദ്ധിപരമായ അൽഗോരിതങ്ങളാണ്. ഈ അൽഗോരിതങ്ങൾ നിർമ്മിത ബുദ്ധിയുടെ ഭാഗമായി മാറിയിരിക്കുന്നു.

നിർമിത ബുദ്ധിയും മത്സരകളികളും

കമ്പ്യൂട്ടറിനെ ഗെയിം കളിക്കുവാൻ പ്രാപ്തമാക്കുക എന്നതാണ് കമ്പ്യൂട്ടർ ശാസ്ത്ര ഗവേഷണങ്ങളുടെ ഒരു പ്രധാന ലക്ഷ്യം. പ്രത്യേകിച്ച് നിർമിത ബുദ്ധിയിൽ. ഷാന്നോൻ എന്ന വ്യക്തിയുടെ ചതുരംഗ കളിയുടെ പ്രോഗ്രാമുമായി ബന്ധപ്പെട്ട പ്രബന്ധവും സാമുവൽ എന്ന വ്യക്തിയുടെ 'ചെക്കേർസ് (Checkers)' എന്ന പ്രോഗ്രാംമും ആണ് ഈ മേഖലയിലെ നാഴികക്കല്ലായി കണക്കാക്കുന്നത്. നിർമിത ബുദ്ധിയുടെ മേഖലയിൽ മത്സരകളികൾ പ്രധാന പങ്കുവഹിക്കുന്നതിന്റെ കാരണങ്ങൾ ചുവടെ ചേർക്കുന്നു.

- മത്സരക്കളികളുടെ നിയമാവലികൾ പരിമിതമായതുകാരണം ഈ മേഖലയിൽ ഉള്ള വിശാലമായ അറിവിന്റെ ആവശ്യമില്ല.
- പ്രോഗ്രാമുകളുടെ വികസനത്തിന് ഈ മേഖലയിലെ ധാരാളം വിദഗ്ധർ സഹായിക്കുന്നു

- വളരെ കുറഞ്ഞ പരിശ്രമം കൊണ്ട് ജയപരാജയം അറിയുവാൻ കഴിയുന്ന വിധമാണ് മത്സരക്കളികളുടെ ഘടന രൂപകൽപന ചെയ്തിരിക്കുന്നത്
- മറ്റു മേഖലകളിൽ നിന്ന് വ്യത്യസ്തമായി ഈ മേഖലയിലെ വിദഗ്ധർക്ക് ഒരു നീക്കത്തിന്റെ കാരണം നിൽകാൻ കഴിയുന്നു.
- മത്സരകളികളിൽ ഒരാൾക്ക് യഥാർഥ ജീവിത സാഹചര്യങ്ങളിലൂടെ കടന്നു പോകുവാനുള്ള അവസരം അനുവദിക്കുന്നു
- 1997 ൽ പ്രശസ്ത ചതുരംഗ കളിക്കാരനായ കാസ്പറോവിനെ ഐ ബി എം ന്റെ ഡീപ് ബ്ലൂ സൂപ്പർ കമ്പ്യൂട്ടർ (Deep Blue) ചതുരംഗ കളിയിൽ (Chess Game) പരാജയപ്പെടുത്തി. അദ്ദേഹത്തെ പരാജയപ്പെടുത്തിയ സിസ്റ്റം, ചതുരംഗ കളിയുടെ വിശാല അറിവുകൾ ഉപയോഗിച്ച് രൂപകൽപ്പന ചെയ്തതായിരുന്നു.

11.4 കമ്പ്യൂട്ടേഷണൽ ബുദ്ധിവൈഭവം (Computational Intelligence)

പുരാതന കാലം മുതൽക്കുതന്നെ, വിവിധതരം ജോലികൾ ചെയ്യുവാൻ യന്ത്രങ്ങൾ നാം ഉപയോഗിച്ചിരുന്നു. ചെയ്യേണ്ട ജോലിക്കനുസരിച്ചിട്ടുള്ള യന്ത്രങ്ങളാണ് നമ്മൾ തിരഞ്ഞെടുത്തിരുന്നത്. കമ്പ്യൂട്ടറുകളുടെ കണ്ടുപിടുത്തത്തോടുകൂടി കൂടുതൽ സമയം എടുക്കുന്നതും തെറ്റുകൾ വരുവാൻ സാധ്യതയുള്ളതുമായ ഗണനം ലാളിത്യവും കൃത്യതയും കൈവരിച്ചു. അതിനുശേഷം യുക്തിസഹജമായ പ്രക്രിയയിൽ കമ്പ്യൂട്ടറുകളുടെ സഹായം തേടാനാകുമോ എന്ന് നമ്മൾ ചിന്തിച്ചു തുടങ്ങി. അലൻ ട്യൂറിംഗ് എന്ന ശാസ്ത്രജ്ഞനാണ്, ഈ ആശയത്തിന്റെ വിത്തുകൾ ശാസ്ത്രമേഖലയിൽ പാകിയത്. യഥാർത്ഥ ജീവിത പ്രശ്നങ്ങളെ കമ്പ്യൂട്ടറുകൾ ഉപയോഗിച്ച് പരിഹരിക്കുവാൻ ഉതകുന്ന, പ്രക്രിയാ മാതൃകകൾ രൂപകൽപ്പന ചെയ്യുക എന്നത് ഇന്ന് ഒരു പ്രധാന വെല്ലുവിളിയായി കരുതപ്പെടുന്നു.

യഥാർഥ ജീവിത പ്രശ്നങ്ങൾ പരിഹരിക്കുന്നതിന് യന്ത്രങ്ങൾ ഉപയോഗിക്കുവാൻ തുടങ്ങിയതോടുകൂടി, മനുഷ്യനും യന്ത്രവും പ്രത്യേകിച്ച് കമ്പ്യൂട്ടറുകളും തമ്മിലുള്ള ഇടപെടലുകൾ വളരെ രസകരമായ പഠനമായി മാറി. അത്തരം പഠനങ്ങൾ സൈബർനെറ്റിക്സ് എന്നറിയപ്പെടുന്ന പഠന ശാഖയുടെ വളർച്ചയ്ക്ക് കാരണമാകുകയും ചെയ്തു. മനുഷ്യനും യന്ത്രവും തമ്മിലുള്ള ആശയവിനിമയത്തിന്റെയും നിയന്ത്രണത്തിന്റെയും പഠനമാണ് സൈബർനെറ്റിക്സ്. ഉദാഹരണത്തിന്, ഈ പഠനങ്ങൾകൊണ്ട് വളർന്ന മേഖലകളാണ്, മനുഷ്യ ഇടപെടൽ ഇല്ലാത്ത ബഹിരാകാശ വാഹനത്തിന്റെ നിയന്ത്രണ സംവിധാനം, സ്വാഭാവിക ഭാഷ പ്രയോഗത്തിലൂടെ കമ്പ്യൂട്ടറുകൾക്ക് നിർദേശങ്ങൾ നൽകൽ, കയ്യെഴുത്തിലൂടെ ഇൻപുട് നൽകൽ, കാര്യക്ഷമതയുള്ള ആയുധങ്ങൾ (Smart Weapons), നിയന്ത്രിത ആണവ പോർമുനകൾ തുടങ്ങിയവ. കമ്പ്യൂട്ടേഷണൽ ബുദ്ധിവൈഭവം (Computational Intelligence) എന്നത് പഠന ഘടകങ്ങൾ, അനുകൂലനം, പരിണാമം, ഫസ്സീ ഭാക്ഷാശാസ്ത്രം (Fuzzy Linquistic) തുടങ്ങിയവ സംയോജിപ്പിച്ച് ബുദ്ധിവൈഭവമുള്ള പ്രോഗ്രാമുകൾ നിർമ്മിക്കുകയും, അവ ഉപയോഗിച്ച് യന്ത്രത്തെ വിവേകമുള്ളതാക്കുകയും ചെയ്യുന്നു. കമ്പ്യൂട്ടിങ്ങ് ബുദ്ധിവൈഭവ വിദഗ്ധർ നിർമ്മിത ബുദ്ധി ഉപയോഗിച്ച് പരിഹരിക്കുവാൻ കഴിയാത്ത പ്രശ്നങ്ങളിലേക്കാണ് ശ്രദ്ധകേന്ദ്രീകരിച്ചിരിക്കുന്നത്. എന്നാൽ മനുഷ്യനും, ചില ജീവജാലങ്ങളും അവയുടെ സ്വാഭാവിക ബുദ്ധി ഉപയോഗിച്ച് ഈ പ്രശ്നങ്ങൾ പരിഹരിക്കുന്നുണ്ട്.

ആളുകൾ ഭംഗിയായി ചെയ്യുന്ന ജോലികൾ കമ്പ്യൂട്ടർ ഉപയോഗിച്ച് എങ്ങനെ ചെയ്യുക്കുവാൻ കഴിയും എന്ന പഠനമാണത് നിർമിത ബുദ്ധി. ഇത് വിവിധ ഗവേഷണ മേഖലകളായ കമ്പ്യൂട്ടർ സയൻസ്, ജീവശാസ്ത്രം, വൈദ്യശാസ്ത്രം, റോബോട്ടിക്സ് തുടങ്ങിയ പഠനശാഖകളോട് കൂടി ചേർന്ന് നിൽക്കുന്നു.

11.4.1 കമ്പ്യൂട്ടേഷണൽ ബുദ്ധിവൈഭവത്തിന്റെ മാതൃകകൾ (Computational Intelligence Paradigms)

സങ്കീർണ്ണവും, മാറികൊണ്ടിരിക്കുന്നതുമായ സാഹചര്യങ്ങളെ ബുദ്ധിപൂർവമായ പ്രവർത്തന രീതികൊണ്ട് സുഗമമാക്കുവാനും, അനുയോജ്യമായ രീതികൾ കണ്ടെത്തുവാനും, യഥാർഥ ജീവിത പ്രശ്നങ്ങൾ പരിഹരിക്കുവാനുള്ള പഠനമാണ് കമ്പ്യൂട്ടേഷണൽ ബുദ്ധിവൈഭവം. ഈ പ്രക്രിയയിൽ നിർമിത ബുദ്ധിയുടെ മേഖലകളായ പുതിയ സാഹചര്യങ്ങൾ മനസിലാക്കുക അല്ലങ്കിൽ പൊരുത്തപ്പെടുക തുടങ്ങിയവയും കമ്പ്യൂട്ടേഷണൽ ബുദ്ധിയുടെ മേഖലകളായ, കമ്പ്യൂട്ടിങ്ങ് മാതൃക, കൃത്രിമ നാഡീശൃംഖല (Artificial Intelligent Neural Network), ഇവല്യൂഷനറി കമ്പൂട്ടേഷൻ (Evolutionary Computation), സ്വാം ഇന്റലിജൻസ് (Swarm Intelligence), ഫസീ ലോജിക് (Fuzzy Logic) എന്നിവയും ഉൾകൊള്ളുന്നു.

ഇതിൽ കമ്പ്യൂട്ടേഷണൽ ബുദ്ധിവൈഭവ മാതൃക ഒറ്റപ്പെട്ട പ്രവർത്തന രീതികളിലൂടെ യഥാർഥ ലോക പ്രശ്നങ്ങളെ വിജയകരമായി പരിഹരിക്കുവാൻ സഹായിക്കുന്നു. എന്നാൽ പുതിയ സാഹചര്യത്തിൽ സങ്കരയിനം മാതൃകകൾ വികസിപ്പിച്ചെടുക്കുകയും, അവ ഒരു സാഹചര്യത്തിലും ഒന്ന് മറ്റൊന്നിനേക്കാൾ മേൽക്കോയ്മ നേടാത്ത അവസ്ഥ ഉണ്ടാക്കുകയും ചെയ്തു. വ്യക്തിഗത ഘടകങ്ങളുടെ പോരായ്മകൾ ഒഴിവാക്കികൊണ്ടു, സങ്കരയിനം കമ്പ്യൂട്ടേഷണൽ ബുദ്ധിവൈഭവ മാതൃകകളുടെ സംവിധാനത്തിൽ തനതു ഘടകങ്ങളുടെ ശക്തി പരമാവധി മുതലെടുക്കുക എന്നതാണ് ഇതിന്റെ ഉദ്ദേശ്യം. ഇവിടെ കമ്പ്യൂട്ടേഷണൽ ബുദ്ധിവൈഭവ മാതൃകയിൽ നാലു പ്രധാന മാതൃകകൾ ആണ് പരിഗണിക്കുന്നത് ഇവ കൃത്രിമ നാഡീ ശൃംഖല (Artificial Intelligent Neural Network), ഇവല്യൂഷനറി കമ്പ്യൂട്ടേഷൻ (Evolutionary Computation), സ്വാം ഇന്റലിജൻസ് (Swarm Intelligence), ഫസീ ലോജിക് (Fuzzy Logic) എന്നിവയാണ്.

സോഫ്റ്റ് കമ്പ്യൂട്ടിങ്ങ് (Soft Computing)

കാലിഫോർണിയ യൂണിവേഴ്സിറ്റി യിലെ കമ്പ്യൂട്ടർ സയൻസ് അദ്ധ്യാപകനായ Lotfi.A.Zadeh ന്റെ അഭിപ്രായത്തിൽ, സോഫ്റ്റ് കമ്പ്യൂട്ടിങ്ങ് രീതി എന്നത് കമ്പ്യൂട്ടിങ്ങ് മേഖലയിലെ വികസിച്ചു കൊണ്ടിരിക്കുന്ന കമ്പൂട്ടിങ്ങ് മേഖലയാണ്. ഇത് അനിശ്ചിതവും അവ്യക്തവുമായ പരിസ്ഥിതിയെ മനസ്സിലാക്കുവാനുള്ള മനുഷ്യ മനസ്സിന്റെ ശ്രദ്ധേയമായ കഴിവുകൾക്ക് സമാനമായവയെ സമാന്തരമായി അവതരിപ്പിക്കുന്നു. വ്യത്യസ്ത ഗ്രൂപ്പുകളുടെ ഒരു മാതൃകയായിട്ടും, കമ്പ്യൂട്ടേഷണൽ ബുദ്ധി മാതൃകകളുടെയും അനുമാനരീതികളുടെ കൂട്ടായ സംവിധാനമായിട്ടാണ് ഇതിനെ കരുതിപോരുന്നത്. വ്യത്യസ്ത മാതൃകകളിൽ നിന്നുള്ള സാങ്കേതികതകളെ കൂട്ടിച്ചേർത്ത് സങ്കരയിനം സംവിധാനം ഉണ്ടാക്കുവാനും സാധിക്കും. അർദ്ധസത്യം, അവ്യക്തത, അനിശ്ചിതത്വം, ഏകദേശരൂപം തുടങ്ങിയ വിവിധതരത്തിലുള്ള സാഹചര്യങ്ങളോട് സോഫ്റ്റ് കമ്പ്യൂട്ടിങ്ങ് രീതി സഹിഷ്ണുത പുലർത്തുന്നു. സോഫ്റ്റ് കമ്പ്യൂട്ടിങ്ങിന്റെ അനുകരണീയ മാതൃക മനുഷ്യമസ്തിഷ്കമാണ്. ഫലത്തിൽ യഥാർഥ ജീവിതത്തിലെ അവ്യക്തമായ സാഹചര്യങ്ങളെ കൈകാര്യം ചെയ്യുവാനുതകുന്ന ഒരു കൂട്ടം സമ്പ്രദായങ്ങളെ സമന്വയിപ്പിച്ച് പ്രവർത്തിപ്പിക്കുവാനും, സൗകര്യപ്രദമായ രീതിയിൽ ആശയങ്ങളെ കൈകാര്യം ചെയ്യിക്കുവാനും സോഫ്റ്റ് കമ്പ്യൂട്ടിങ്ങ് രീതിയ്ക്ക് കഴിയുന്നു.

മുകളിൽ സൂചിപ്പിച്ചിരിക്കുന്ന എല്ലാ കമ്പ്യൂട്ടേഷണൽ ബുദ്ധി മാതൃകകളുടെയും ഉത്ഭവം ജൈവ വ്യവസ്ഥിതിയിൽ നിന്നാണ്. കൃത്രിമ നാഡീ ശൃംഖല (Artificial Neural Network) മാതൃകയാക്കുന്നത് ജൈവ വ്യവസ്ഥിതിയിലെ നാഡീവ്യൂഹത്തെയാണ്. ഇവല്യൂഷനറി കമ്പ്യൂട്ടിങ് (Evolutionary Computing) മാതൃകയാക്കുന്നതു പ്രകൃതി പരിണാമത്തെയാണ് (ജനിതക പരിണാമവും, സ്വഭാവ പരിണാമവും ഉൾപ്പടുന്നു). സ്വാം ഇന്റലിജൻസ് (Swarm Intelligence) മാതൃകയാക്കുന്നത് കോളനികളായി ജീവിക്കുന്ന ജീവികളുടെ അല്ലെങ്കിൽ തേനീച്ചക്കൂട്ടത്തിന്റെ സാമൂഹിക സ്വഭാവ സവിശേഷതയാണ്, ഫസി വ്യവസ്ഥിതി (Fuzzy System) ഉരുത്തിരിഞ്ഞിരിക്കുന്നത്, ജീവികൾ അവയുടെ പരിസ്ഥിതിയുമായി എങ്ങനെ സംവദിക്കുന്നുവെന്നതും, അനിശ്ചിതമായ, അല്ലെങ്കിൽ അവ്യക്തമായ ആവശ്യങ്ങളോട് എങ്ങനെ പ്രതികരിക്കുന്നു വെന്നതും ഗ്രഹിച്ചാണ്.

A. കൃത്രിമ നാഡീ ശൃംഖല (Artificial Neural Network)

മനുഷ്യ മസ്തിഷ്കത്തെ സങ്കീർണ്ണവും ക്രമീകരിക്കാത്ത രീതിയിൽ സമാന്തരമായി പ്രവർത്തിക്കുന്നതുമായ ഒരു കമ്പ്യൂട്ടറിനോട് ഉപമിക്കാം. രൂപരേഖകളെ തിരിച്ചറിയുക, കാര്യങ്ങൾ ഗ്രഹിക്കുക, ചലിക്കുക തുടങ്ങിയ പ്രവർത്തികൾ ഒരു കമ്പ്യൂട്ടറിനേക്കാൾ വേഗത്തിൽ മസ്തിഷ്കത്തിന് ചെയ്യുവാൻ കഴിയുന്നു. എന്നിരുന്നാലും സിലിക്കൺ ഗേറ്റ്സ് നാനോ സെക്കന്റുകളിലും നാഡി വ്യവസ്ഥ മില്ലി സെക്കന്റുകളിലുമാണ് കാര്യങ്ങൾ നടപ്പിലാക്കുന്നത്. മസ്തിഷ്കത്തിന് ഈ സവിശേഷതകൾ കൂടാതെ ഗ്രഹിക്കുവാനുള്ള കഴിവ്, ഓർമിക്കുവാൻ ഉള്ള കഴിവ്, എപ്പോഴും അനുമാനിക്കുവാനുള്ള കഴിവ് തുടങ്ങിയവയും ഉണ്ട്. ജീവശാസ്ത്രപരമായ നാഡീ വ്യവസ്ഥകളുടെ, ഗവേഷണമേഖലകളിലെ പ്രക്രിയ മാതൃകകളാണ് കൃത്രിമ നാഡീ ശൃംഖല.

മനുഷ്യന്റെ മസ്തിഷ്കത്തിലുള്ള കോർട്ടെക്സിൽ [Cortex] 10 - 500 ലക്ഷം കോടി നാഡീകോശങ്ങളും 60 ട്രില്യൺ സിനാപ്സുകളും ഉണ്ട് എന്നാണ് കണക്കാക്കിയിരിക്കുന്നത്, സിനാപ്സ് (Synapses) എന്നത് ഒരു നാഡീകോശത്തിൽ നിന്ന് മറ്റൊരു നാഡീകോശത്തിലേക്കു ആവേഗങ്ങളെ സംയോജിപ്പിക്കുന്ന പ്രധാന ഇടമാണിത്. നാഡീകോശങ്ങളെ ഏകദേശം 1000 ത്തോളം പ്രധാന ഘടകങ്ങളായി ക്രമീകരിച്ചിരിക്കുന്നു. ഓരോന്നിനും 500 ഓളം നാഡീവ്യൂഹങ്ങൾ ഉണ്ട്. ചെറിയ കാര്യങ്ങൾ കൃത്രിമ നാഡീ ശൃംഖലകൾ (ANN) പരിഹരിക്കുന്നു എന്നതാണ് നാഡീ മാതൃകകളുടെ (neural modelling) ഇപ്പോഴത്തെ വിജയം

B. ഇവല്യൂഷനറി കമ്പ്യൂട്ടേഷൻ (Evolutionary Computation)

സ്വാഭാവിക പരിണാമ പ്രക്രിയയെ അനുകരിക്കുക എന്നതാണ് പരിണാമ കമ്പ്യൂട്ടിങ് രീതിയുടെ പ്രധാന ലക്ഷ്യം. ഉത്തമഗുണങ്ങൾ ഉള്ളവ അതിജീവിക്കു എന്നതാണ് ഇവിടത്തെ പ്രധാന ആശയം. അതായതു ദുർബലർ നശിപ്പിക്കപ്പെടും. പ്രകൃതി പരിണാമ പ്രക്രിയയിൽ, ഒരു ജീവിയുടെ നിലനിൽപ്പ് പ്രത്യുൽപ്പാദനം വഴിയാണ് നടക്കുന്നത്. എല്ലാ സന്തതികളിലും മാതാപിതാക്കളുടെ രണ്ടുപേരുടെയും ജനിതക ഘടകങ്ങൾ ഉണ്ടായിരിക്കും. മാതാപിതാക്കളുടെ മികച്ച സ്വഭാവ സവിശേഷതകൾ സന്തതികൾക്ക് കിട്ടും എന്ന് പ്രതീക്ഷിക്കുന്നു. അതിനു വിരുദ്ധമായി മോശപ്പെട്ട സ്വഭാവ സവിശേഷതകൾ പാരമ്പര്യമായി ലഭിക്കുന്നവർക്ക് അതിജീവനം സാധ്യമല്ലാതെ വരുന്നു. എല്ലാ ജീവിവർഗങ്ങളും നിലനിൽക്കാൻ സാധ്യത ഉള്ളതിലും കൂടുതൽ കുഞ്ഞുങ്ങളെ ഉൽപ്പാദിപ്പിക്കുന്നുണ്ട്. അതിനാൽ അവർ തമ്മിൽ ആഹാരത്തിനു വേണ്ടിയും മറ്റും സമരം

നടക്കുകയും അവയിൽ ഏറ്റവും നന്നായി പ്രകൃതിയോട് ഇണങ്ങി ജീവിക്കാൻ കഴിയുന്നവ മാത്രം നിലനിൽക്കുകയും ചെയ്യുന്നു. ഇത് ചില പക്ഷി വർഗ്ഗങ്ങളിൽ നമുക്കു കാണുവാൻ സാധിക്കും. അവിടെ കൂടുതൽ ഭക്ഷണം ലഭിക്കുവാനും ശക്തിയാർജിക്കുവാനുമായി പക്ഷി കുഞ്ഞുങ്ങളിൽ ദുർബലന്മാരെ കൂട്ടിൽ നിന്ന് പുറംതള്ളി ഇല്ലാതാക്കുന്നു.

പരിണാമ അൽഗൊരിതങ്ങൾ വ്യക്തികളുടെ സാമൂഹിക ജീവിതത്തിൽ ഉപയോഗിക്കുന്നു. ഇവിടെ വ്യക്തികളെ ക്രോമോസോം ആയി കരുതാം. ഒരു സമൂഹത്തിലെ വ്യക്തികളുടെ സ്വഭാവ സവിശേഷതകളെ ക്രോമോസോം നിർവചിക്കുന്നു. ഓരോ സ്വഭാവ സവിശേഷതയും ഓരോ ജീനുമായി (Gene) ബന്ധപ്പെട്ടിരിക്കുന്നു. ഒരു സ്വഭാവത്തെ നിർണ്ണയിക്കുന്ന ജീനിന് വ്യത്യസ്ത തരങ്ങൾ ഉണ്ടാകും. ഒരു ജീനിന്റെ വ്യത്യസ്ത തരങ്ങളെ അലീലുകൾ [Allele] എന്ന് വിളിക്കുന്നു. ഓരോ തലമുറയിലെയും ജീവികൾ അവയുടെ സന്തതിപരമ്പരകളെ സൃഷ്ടിക്കുവാൻ മത്സരിക്കുന്നു.

അതിജീവന ശേഷി കൂടുതലുള്ള വ്യക്തികൾക്ക് പ്രത്യുൽപ്പാദന ശേഷിക്കുള്ള സാധ്യത മികച്ചതാണ്. സന്താനപരമ്പരകൾ സൃഷ്ടിക്കപ്പെടുന്നത് മാതാപിതാക്കളുടെ സ്വഭാവസവിശേഷതകൾ സമന്വയിപ്പിച്ചു കൊണ്ടാകും. ഇതിനെ ക്രോസ്സോവർ എന്ന് പറയുന്നു. ഒരു ജീവിയുടെ ജനിതകഘടനയിൽ പെട്ടെന്ന് ഉണ്ടാകുന്നതും അടുത്ത തലമുറയിലേക്കു കൈമാറ്റം ചെയ്യപ്പെടുന്നതുമായ മാറ്റങ്ങളാണ് ഉല്പരിവർത്തനം (mutation). ഒരു ജീവി വർഗ്ഗത്തിലെ ഓരോ ജീവിയിലും ഉല്പരിവർത്തനം വഴി ക്രോമോസോമുകളിലെ അലീലുകളിൽ മാറ്റങ്ങൾ സംഭവിക്കുന്നു. ഒരു ജീവിയുടെ അതിജീവനശേഷി സാധ്യമാവുന്നത് അവയുടെ ആവശ്യങ്ങൾ നേടുവാനും മുന്നിലുള്ള പ്രതിബന്ധങ്ങളെ തരണം ചെയ്യുവാനുമുള്ള ക്ഷമതയെ ആധാരമാക്കിയാണ്.

യഥാർഥ ലോകത്തിലെ പ്രവർത്തനങ്ങളായ ഡാറ്റ ഖനനം (ഡാറ്റ മൈനിംഗ്) (പുതിയ വിവരങ്ങൾ ഡാറ്റാബേസിൽ നിന്ന് കണ്ടെത്തുക), തകരാറുകൾ നിർണ്ണയിക്കുക (Fault Diagnosis), വർഗീകരണം (classification), പദ്ധതി തയ്യാറാക്കുക (scheduling) തുടങ്ങിയവയിൽ വളരെ വിജയകരമായി തന്നെ ഇവല്യൂഷനറി കമ്പ്യൂട്ടേഷൻ (Evolutionary Computation) നടപ്പിലാക്കിയിരിക്കുന്നു.

C. സ്വാം ഇന്റലിജൻസ് (Swarm Intelligence)

കോളനിയായി താമസിക്കുന്ന ജീവികളുടെ ജീവിത രീതികളെക്കുറിച്ചുള്ള പഠനത്തിൽ നിന്നാണ് സ്വാം ഇന്റലിജൻസ് എന്ന ആശയം ഉരുത്തിരിഞ്ഞത് (ചിത്രം 11.11). ജീവികളുടെ സാമൂഹിക സ്വഭാവത്തെക്കുറിച്ചുള്ള പഠനങ്ങൾ വളരെ കാര്യക്ഷമമായ അൽഗോരിതങ്ങൾ രൂപകൽപ്പന ചെയ്യുവാൻ പ്രേരണയായി. ഉദാഹരണത്തിന്, ആകാശത്തു കൂട്ടമായി പറക്കുന്ന പക്ഷികളുടെ പ്രവചനാതീതമായ ചലന ക്രമീകരണങ്ങളുടെ മനോഹരമായ ആവിഷ്കാരപഠനങ്ങൾ നയിച്ചത് പാർട്ടിക്കിൾ സ്വാം ഓപ്ടിമൈസേഷൻ അൽഗോരിതത്തിലേക്കാണ് (Particle Swarm Optimization Algorithms). ഭക്ഷണസാധനങ്ങൾ തേടിയുള്ള ഉറുമ്പുകളുടെ സഞ്ചാരത്തെക്കുറിച്ചുള്ള പഠനം ആന്റ് കോളനി ഓപ്റ്റിമൈസേഷൻ അൽഗോരിത (Ant colony optimization algrorithm) മായി പരിണമിച്ചു.

ചിത്രം 11.11: ഉറുമ്പുകളുടെ സമൂഹം

യന്ത്രപഠനം (Machine Learning)

സങ്കീർണ്ണ രൂപങ്ങളിൽനിന്നും പ്രയോഗസിദ്ധമായ ഡാറ്റയിൽ നിന്നും ഒരു യന്ത്രത്തിന് അനുമാനങ്ങളിൽ എത്തിച്ചേരുവാനുള്ള കഴിവിനെ യന്ത്രപഠനം എന്ന് പറയുന്നു. ചിത്രങ്ങളുടെയും ശബ്ദത്തിന്റെയും വിശകലന മേഖലകളിലാണ് യന്ത്രപഠനം കൂടുതലായി ഉപയോഗിച്ചിരിക്കുന്നത്. അതോടൊപ്പം ഡാറ്റ ഖനന പ്രക്രിയകളിലും ഇത് പ്രയോജനപ്പെടുത്തുന്നു. കമ്പ്യൂട്ടേഷണൽ ബുദ്ധിവൈഭവ (Computational Intelligence) സമ്പ്രദായങ്ങൾ എല്ലാം ഒരുപോലെയാണെങ്കിലും പ്രധാന അറിവുകളുടെ പ്രതിനിധീകരണം യന്ത്രപഠനത്തിൽ സ്ഥിരമാണ്. എന്നാൽ കമ്പ്യൂട്ടേഷണൽ ബുദ്ധിവൈഭവത്തിൽ പ്രധാന അറിവുകളുടെ പ്രതിനിധീകരണം ഏകദേശ രൂപത്തിലുള്ള അവതരണമാണ് (Fuzzy).

D. ഫസി സിസ്റ്റം (Fuzzy System)

പരമ്പരാഗത സെറ്റ് തിയറിയിൽ അംഗങ്ങൾ ഒരു ഗണത്തിൽ ഉൾപ്പെടുന്നുണ്ടോ ഇല്ലയോ എന്നുള്ള നിഗമനമാണ് നടക്കുന്നത്. അതുപോലെ ബൈനറി വിലയുടെ (binary value) യുക്തിയിൽ മൂല്യങ്ങൾ 0 അല്ലെങ്കിൽ 1 ആകാം. എന്നിരുന്നാലും മനുഷ്യന്റെ യുക്തി എല്ലായിപ്പോഴും ഇത്രയും കൃത്യമായിക്കൊള്ളണമെന്നില്ല. നമ്മുടെ നിരീക്ഷണത്തിലും യുക്തിയിലും പലപ്പോഴും അനിശ്ചിതത്വം ഉൾപ്പെട്ടിട്ടുണ്ടാകും. ഉദാഹരണത്തിന് 'ചില കമ്പ്യൂട്ടർ സയൻസ് വിദ്യാർഥികൾക്ക് എല്ലാ ഭാഷയിലും പ്രോഗ്രാം ചെയ്യാൻ സാധിക്കും; എന്ന വാചകം എല്ലാവർക്കും മനസ്സിലാക്കുവാൻ കഴിയും. പക്ഷെ കമ്പ്യൂട്ടറിനു എങ്ങനെ ഈ യാഥാർഥ്യത്തെ അവതരിപ്പിക്കുവാനും യുക്തിവിചാരം നടത്തുവാനും കഴിയും? ഫസി സെറ്റുകളും ഫസി ലോജിക്കും ഏകദേശ യുക്തിവിചാരത്തെ അനുവദിക്കുന്നു. അംഗങ്ങൾ ഒരു വിഭാഗത്തിൽ എത്രത്തോളം ഉൾക്കൊണ്ടിട്ടുണ്ട് എന്ന് ഫസി സെറ്റുകൾ കൊണ്ട് പരാമർശിക്കുന്നു. ഫസി ലോജിക് അനിശ്ചിതമായ വസ്തുതകളിൽ നിന്ന്, ഓരോ വസ്തുതകളിലും അടങ്ങിയിരിക്കുന്ന നിശ്ചിതത്വത്തിന്റെ അംശം ഉറപ്പാക്കിക്കൊണ്ട്, പുതിയ വസ്തുതകളുടെ യുക്തിവിചാരത്തിലെത്തിച്ചേരുന്നു. ഒരർഥത്തിൽ ഫസി സെറ്റുകളും ഫസി ലോജിക്കും സാമാന്യ ബോധത്തിന്റെ മാതൃകകൾ അനുവദിക്കുന്നു.

വാഹനങ്ങളിലെ ഗിയർ ട്രാൻസ്മിഷൻ, റാക്കിങ് സംവിധാനങ്ങളുൾ തുടങ്ങിയവയുടെ നിയന്ത്രണം, ലിഫ്റ്റ് നിയന്ത്രണം, ഗൃഹോപകരണങ്ങളുടെ നിയന്ത്രണം, ട്രാഫിക് സിഗ്നൽ നിയന്ത്രണം തുടങ്ങിയ മേഖലകളിൽ ഫസി സംവിധാനം (Fuzzy System) വിജയകരമായി നടപ്പിലാക്കിയിരിക്കുന്നു.

11.4.2 കമ്പ്യൂട്ടേഷണൽ ബുദ്ധിയുടെ പ്രയോഗം

കമ്പ്യൂട്ടേഷണൽ ബുദ്ധിവൈഭവം (പൊതുവായ് നിർമിതബുദ്ധി എന്ന് അറിയപ്പെടുന്നു) എന്നത് വിശാലമായ ഇടപെടൽ ഉള്ളതും മറ്റു പല പ്രവർത്തന മേഖലകളിലും സ്വാധീനമുള്ളതുമാണ്. കമ്പ്യൂട്ടിംഗ് മേഖലയിൽ മാത്രമല്ല, ഗണിതം, ഭാഷാശാസ്ത്രം, മനശാസ്ത്രം, നാഡീ ശാസ്ത്രം, മെക്കാനിക്കൽ എഞ്ചിനീയറിംഗ്, സ്റ്റാറ്റിസ്റ്റിക്സ്,സാമ്പത്തിക ശാസ്ത്രം, നിയന്ത്രണ സിദ്ധാന്തവും സൈബർനെറ്റിക്സും, തത്വചിന്ത തുടങ്ങി മറ്റു പലമേഖലകളിലും ഇത് ഉപയോഗിക്കുന്നു. കമ്പ്യൂട്ടേഷണൽ ബുദ്ധിവൈഭവത്തിന്റെ പ്രയോഗത്തെക്കുറിച്ച് ഇപ്പോൾ സമഗ്ര പഠനം നടത്തിക്കൊണ്ടിരിക്കുന്നു. കമ്പ്യൂട്ടേഷണൽ ബുദ്ധിയുടെ പ്രചുര പ്രചാരം നേടിയ ചില പ്രയോഗങ്ങളെക്കുറിച്ച് നമുക്ക് ഇവിടെ ചർച്ച ചെയ്യാം.

A. ബയോമെട്രിക്സ്

ബയോമെട്രിക്സ് എന്നത് മനുഷ്യന്റെ സവിശേഷതകളുമായും വിശേഷണ ഗുണങ്ങളുടെ അളവുകളുമായും ബന്ധപ്പെട്ടിരിക്കുന്നു.വ്യക്തികളെ തിരിച്ചറിയുവാൻ ബയോമെട്രിക് തിരിച്ചറിയൽ സംവിധാനം ഉപയോഗിക്കുന്നു. തിരിച്ചറിയുവാൻ കഴിയുന്നതും അളക്കുവാൻ കഴിയുന്നതുമായ സ്വഭാവ സവിശേഷതകൾ ഉപയോഗിച്ച് ഒരു വ്യക്തിയെ അടയാളപ്പെടുത്തുവാനും വിവരങ്ങൾ തയാറാക്കുവാനും ബയോമെട്രിക് തിരിച്ചറിയലുകൾ ഉപയോഗിക്കുന്നു. ശാരീരിക പ്രവർത്തനങ്ങൾ, ശരീര ഘടനയുമായും ശരീരത്തിന്റെ സവിശേഷതകളുമായും ബന്ധപ്പെട്ടിരിക്കുന്നു. വിരലടയാളം, ഹസ്തരേഖകൾ, മുഖം, കയ്യുടെ അളവുകൾ, ഐറിസ്, റെറ്റിന, ഗന്ധം, മണം തുടങ്ങിയവ ഇതിൽ ഉൾപ്പെടുന്നു. സ്വഭാവ സവിശേഷതകൾ വ്യക്തിയുടെ സ്വഭാവമാതൃകയുമായി ബന്ധപ്പെട്ടിരിക്കുന്നു. ഇതിൽ ശബ്ദം, വേഗത, എന്നിവ ഉൾപ്പെട്ടിരിക്കുന്നു.

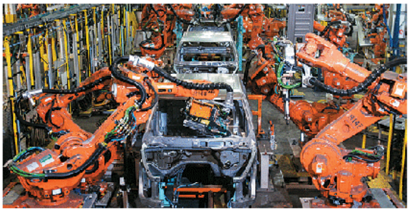
ഹാജർ രേഖപ്പെടുത്തുവാനും, കമ്പ്യൂട്ടറും മറ്റ് ഉപകരണങ്ങളും ഉപയോഗിക്കുന്നതിന്റെ ആധികാരികത ഉറപ്പാക്കുവാനും, ആധാർകാർഡ് പോലുള്ള പ്രധാനപ്പെട്ട മേഖലകളിലും ബയോമെട്രിക് തിരിച്ചറിയൽ സാധാരണയായി ഉപയോഗിക്കുന്നു. ചിത്രം 11 .12 ബയോമെട്രിക്കിന്റെ ഉപയോഗം കാണിച്ചിരിക്കുന്നു.

ചിത്രം. 11.12: വിരലടയാളവും റെറ്റിനയും തിരിച്ചറിയൽ

B. റോബോട്ടിക്സ്

റോബോട്ട് ഒരു ഇലക്ട്രോമെക്കാനിക്കൽ ഉപകരണമാണ്. ഇവയ്ക്കു സാഹചര്യത്തിന് അനുസരിച്ചു പ്രതികരിക്കാനും സ്വതന്ത്ര രീതിയിൽ തീരുമാനം എടുത്ത് പ്രവർത്തിച്ച് അതിലൂടെ പ്രത്യേക കർത്തവ്യം നിർവഹിക്കുവാനുമുള്ള കഴിവുണ്ട്.

റോബോട്ടുകളുടെ രൂപകൽപ്പന, നിർമാണം, സിദ്ധാന്തം, ഉപയോഗങ്ങൾ എന്നിവയുടെ ശാസ്ത്രീയ പഠനത്തേ റോബോട്ടിക്സ് എന്ന് നിർവചിച്ചിരിക്കുന്നു. റോബോട്ടിക്സ് എന്നത് ബഹുമുഖ ശാസ്ത്രശാഖയാണ്. മെക്കാനിക്കൽ എഞ്ചിനീയറിങ്, ഇലക്ട്രിക്കൽ എഞ്ചിനീയറിംഗ്, കമ്പ്യൂട്ടർ സയൻസ് തുടങ്ങിയ മേഖലകളിൽ വ്യാപിച്ചു കിടക്കുന്ന പഠന ശാഖയാണ് ഇത്. റോബോട്ടിക്കിന്റെ ശാസ്ത്രീയ പഠനത്തിൽ, റോബോട്ടുകളുടെ രൂപകൽപ്പന, നിർമ്മാണം, പ്രവർത്തനം, ഉപയോഗം തുടങ്ങിയവയും, കമ്പ്യൂട്ടർ സംവിധാനമുപയോഗിച്ച് ഇവയുടെ നിയന്ത്രണം, സംവേദ ഉപകരണങ്ങൾ വഴിയുള്ള വിവര ശേഖരണം, ഗ്രഹിക്കുന്ന വിവരങ്ങളെ പ്രക്രിയയ്ക്ക് വിധേയമാക്കുക എന്നിവയും ഉൾപ്പെടുന്നു. ചിത്രം 11.13 ൽ റോബോട്ടുകളുടെ ചില ഉപയോഗങ്ങൾ കാണിച്ചിരിക്കുന്നു.


(a)

(b)

(c)

ചിത്രം. 11.13: (a) കാർ നിർമാണത്തിൽ, (b) കാർഷിക മേഖലയിൽ (c) സൈനിക ഡ്രോൺ എന്നിവയിൽ റോബോട്ടുകൾ

ഇന്നു റോബോട്ടുകൾ വാഹന വ്യവസായം, ബഹിരാകാശമേഖല, സൈനികമേഖല തുടങ്ങിയ നിരവധി മേഖലകളിൽ ഉപയോഗിക്കുന്നു. ഇവയിൽ ചിലതു നമുക്കു ചർച്ച ചെയ്യാം.

വാഹന നിർമാണ മേഖലയിലെ ഉപയോഗങ്ങൾ: റോബോട്ടിക് കൈകൾ വാഹന നിർമാണ പ്രക്രിയയിൽ ഉപയോഗിക്കുന്നു. ഇവ വെൽഡിങ്, കട്ടിങ്, ഉയർത്തൽ, ക്രമപ്പെടുത്തൽ, വളയ്ക്കൽ തുടങ്ങിയ ഒന്നിലധികം ജോലികൾ ചെയ്യുവാൻ കഴിവുള്ളവയാണ്.

ബഹിരാകാശ പര്യവേഷണം: ബഹിരാകാശ നിലയങ്ങൾ സ്ഥാപിക്കുക, ബഹിരാകാശ വാഹനത്തിൽനിന്നും ഉപഗ്രഹങ്ങൾ വിക്ഷേപിക്കുക എന്നീ പ്രവർത്തികൾ മനുഷ്യനിയന്ത്രണത്തിലൂടെ യന്ത്രക്കൈകൾ നിർവഹിക്കുന്നു.

സ്മാർട്ട് ഹോംസ്: വീടുകളുടെ സുരക്ഷയ്ക്കും ഊർജ്ജ ഉപയോഗ നിയന്ത്രണത്തിനും പരിസര അവലോകനത്തിനും സ്വയം പ്രവർത്തന സംവിധാനം ഉപയോഗിക്കുന്നു. കതകുകളും ജനലുകളും സ്വയം തുറക്കുവാനും, ഗൃഹോപകരണങ്ങളും ലൈറ്റും ശീതീകരണികളും മുൻകൂട്ടി തയ്യാറാക്കി വെച്ച പ്രോഗ്രാമുകൾ അനുസരിച്ചു സ്വയം പ്രവർത്തന ക്ഷമമാക്കുവാനും കഴിയുന്നു.

ബുദ്ധിമുട്ടുള്ള സാഹചര്യങ്ങളിലെ പര്യവേക്ഷണം. മനുഷ്യന് പ്രവേശിക്കുവാൻ കഴിയാത്ത അപകട സാധ്യത കൂടിയ മേഖലകളിൽ എത്തിച്ചേരുവാൻ റോബോട്ടുകൾക്കു കഴിയുന്നു. അഗ്നിപർവ്വതങ്ങളുടെ ഉൾഭാഗത്തും ആഴക്കടൽ നിരീക്ഷണത്തിനും ഇവ ഉപയോഗിക്കുന്നു. 1960 കളുടെ തുടക്കം മുതൽ തന്നെ നാസ (NASA) ഗ്രഹങ്ങളുടെ പര്യവേക്ഷണത്തിനായി റോബോട്ടുകളെ ഉപയോഗിച്ച് തുടങ്ങിയിരുന്നു

സൈനികപരമായ ഉപയോഗങ്ങൾ: ഇക്കാലത്തു ആധുനിക സൈന്യങ്ങൾ പല ആവശ്യങ്ങൾക്കായി എയർ ബോൺ റോബോട്ടുകൾ (ഡ്രോൺസ്) ഉപയോഗിക്കുന്നു. നിരീക്ഷണ മേഖലയിലാണ് പ്രധാനമായും ഡ്രോണുകൾ ഇന്ന് കൂടുതലായും ഉപയോഗിക്കുന്നത്. ആകാശ മാർഗത്തിലൂടെ ശത്രുക്കളുടെ ആവാസ കേന്ദ്രങ്ങൾ ആക്രമിക്കുവാൻ ഡ്രോണുകൾ ഉപയോഗിക്കുന്നു. ഭാവിയിൽ, സ്വയം നിയന്ത്രിത എയർക്രാഫ്റ്റുകളും വാഹനങ്ങളും ഉപയോഗിച്ച് ഇന്ധനവും സ്ഫോടകവസ്തുക്കളും യുദ്ധമേഖലകളിൽ എത്തിക്കുവാനും മൈനുകൾ നീക്കം ചെയ്യുവാനും ഉപയോഗിച്ചേക്കാം.

കാർഷിക മേഖലയിലെ ഉപയോഗങ്ങൾ: വികസിത രാജ്യങ്ങളിലെ വലിയ കൃഷിമേഖലകളിൽ വിളവെടുപ്പ് നടത്തുവാനും വിളകൾ സ്വയം ശേഖരിക്കുവാനും യന്ത്രങ്ങൾ ഉപയോഗിക്കുന്നു. റോബോട്ടിക് ക്ഷീരശാലകളിൽ മൃഗങ്ങൾക്കു ഭക്ഷണം നൽകുവാനും പാൽ ശേഖരിക്കുവാനും വിദൂര സ്ഥലങ്ങളിൽ ഇരുന്ന് സാധിക്കുന്നു.

1942 ൽ പ്രസിദ്ധീകരിച്ച റൺറൌണ്ട് (RUNROUND) എന്ന ചെറുകഥയിൽ അമേരിക്കൻ കഥാകൃത്തും ബയോ കെമിസ്ട്രി പ്രൊഫസറുമായ ഐസക് അസിമോസ് (ISSAC ASIMOU) ആണ് റോബോട്ടിക് എന്ന പദം ആദ്യമായി ഉപയോഗിച്ചത്. റോബോട്ടാ എന്ന ചെക്ക് പദത്തിൽ നിന്നാണ് ഈ വാക്കിന്റെ ഉദ്ഭവം. ഇതിന്റെ അർത്ഥം 'പ്രതിഫലം വാങ്ങാതെ നിർബന്ധമായി കായികാധ്വാനം ചെയ്യുക' എന്നാണ്.

C. കമ്പ്യൂട്ടർ ദൃഷ്ടി (COMPUTER VISION)

ദ്വിമാന ദൃശ്യങ്ങളിൽ നിന്ന്, വ്യക്തമായതും, രൂപഘടനയുടെ അർഥപൂർണമായ വിവരങ്ങ ളടങ്ങിയതുമായ ത്രിമാന ദൃശ്യങ്ങളുടെ നിർമാണമാണ് കമ്പ്യൂട്ടർ ദൃഷ്ടി. കമ്പ്യൂട്ടറുകളുടെയും ക്യാമറകളുടെയും സഹായത്തോടുകൂടി ശേഖരിക്കുന്ന ദ്വിമാന ദൃശ്യങ്ങളെ അടിസ്ഥാനമാക്കി കമ്പ്യൂട്ടർ വിഷൻ ത്രിമാന ദൃശ്യങ്ങളുടെ ആകൃതിയും ഗുണങ്ങളും കണക്കാക്കുന്നു. ഇതിനെ ചിത്രങ്ങളെ ഗ്രഹിക്കുക (IMAGE UNDERSTANDING) എന്നും വിളിക്കുന്നു.

ദൃശ്യ മാധ്യമങ്ങളുടെ സഹായത്താൽ ബൗദ്ധിക ലോകത്തെ മനസ്സിലാക്കാൻ കമ്പ്യൂട്ടറുകളെ പ്രാപ്തമാക്കുക എന്നതാണ് ഈ ശാഖയുടെ ഉദ്ദേശം. ചിത്രങ്ങൾ ഏതു രൂപത്തിലും എടുക്കാവു ന്നതാണ്. അതായത്, ക്രമമായ വീഡിയോ ആയിട്ടോ, ഒന്നിൽ കൂടുതൽ ക്യാമറകൾ ഉപയോഗിച്ച് വിവിധ തലങ്ങളിലെ ചിത്രങ്ങളായിട്ടോ, മെഡിക്കൽ സ്കാനറിൽ നിന്നുള്ള ബഹുമുഖ ഡാറ്റകളായിട്ടോ ആകാം ഇത്.

കമ്പ്യൂട്ടർ വിഷൻ ആദ്യമായി വികസിപ്പിച്ചത് സൈനിക ആവശ്യങ്ങൾക്കാണ്. ഇത് നിർമിത ബുദ്ധിയുടെയും റോബോട്ടിക്സി ന്റെയും ഒരു പ്രധാന ഭാഗമാണ്. ദൃശ്യ പുന രാവിഷ്കരണം, സ്ഥിതി മനസ്സിലാക്കൽ, വീഡിയോ ട്രാക്കിംഗ്, ഗ്രഹിക്കൽ, ഇൻഡ ക്സിങ്, വസ്തുവിനെ മനസ്സിലാക്കൽ, ചിത്ര പുനരാവിഷ്കരണം തുടങ്ങിയവ കമ്പ്യൂട്ടർ വിഷന്റെ ഉപമേഖലകൾ ആണ്. ചിത്രം 11.14ൽ ചൊവ്വയിൽ ക്യൂരിയോസിറ്റി എന്ന മാർസ് റോവർ കമ്പ്യൂട്ടർ വിഷൻ ഉപയോ ഗിച്ച് ഗവേഷണം നടത്തുന്നതിന്റെ ചിത്രം പ്രദർശിപ്പിച്ചിരിക്കുന്നു.

ചിത്രം 11.14: ചൊവ്വ പര്യവേഷണ വാഹനം ക്യൂര്യോസിറ്റി.

D. സ്വാഭാവിക ഭാഷ പ്രക്രിയ (നാച്ചുറൽ ലാംഗ്വേജ് പ്രോസസ്സിംഗ് (NLP))

ജനങ്ങളുടെ സംസാര ഭാഷയെ സ്വാഭാവിക ഭാഷകൾ (Natural Language) എന്ന് പറയുന്നു. കമ്പ്യൂട്ടർ ശാസ്ത്ര മേഖലയിലെ ഒരു ശാഖയായ സ്വാഭാവിക ഭാഷ പ്രക്രിയ (നാച്ചുറൽ ലാംഗ്വേജ് പ്രോസസ്സിംഗ്) പ്രധാനമായും ലക്ഷ്യമിടുന്നത് ഇംഗ്ലീഷ്, മലയാളം തുടങ്ങിയ ഭാഷകളിലൂടെ കമ്പ്യൂട്ടറുമായി ആശയവിനിമയം നടത്തുവാനുള്ള സംവിധാനം ഉണ്ടാക്കുക എന്നതാണ്. നിർമ്മിത ബുദ്ധി (AI) യുടെ ഉപമേഖലയാണ് സ്വാഭാവിക ഭാഷ പ്രക്രിയ (Natural Language Processing). ഏതെങ്കിലും സവിശേഷ അറിവ് ഇല്ലാതെ തന്നെ കമ്പ്യൂട്ടറുമായി ആളുകൾക്ക് ഇടപെടാൻ ഇത് സഹായിക്കുന്നു. ഏതൊരു വ്യക്തിയ്ക്കും അവരുടെ സ്വന്തം ഭാഷ ഉപയോഗിച്ച് കമ്പ്യൂട്ടറുമായി സംസാരിക്കാം എന്നതാണ് ഇതുകൊണ്ടു ഉദ്ദേശിക്കുന്നത്. അതിനായി ഏതെ

ങ്കിലും പ്രോഗ്രാമിങ് ഭാഷ പഠിക്കേണ്ട ആവശ്യം ഇല്ല. ചിഹ്നങ്ങളുടെയും ചട്ടങ്ങളുടെയും ഒരു ഗണമാണ് ഭാഷ എന്ന് നമുക്ക് അറിയാം. പുതിയ ആശയങ്ങൾ വിനിമയം ചെയ്യുവാൻ ചിഹ്നങ്ങളെ കൂട്ടിയോജിപ്പിക്കുന്നു. ഭാഷ നിയമങ്ങൾ ചിഹ്നങ്ങളെ ചിട്ടപ്പെടുത്തുവാനുള്ള മാർഗനിർദേശങ്ങൾ തരുന്നു. ഒരു ഭാഷയുടെ വ്യാകരണമെന്നാണ് ഈ നിയമങ്ങളെ നമ്മൾ വിളിക്കുന്നത്. സ്വാഭാവിക ഭാഷകൾ മനസ്സിലാകുന്ന തരത്തിൽ കമ്പ്യൂട്ടർ പ്രോഗ്രാമുകൾ വികസിപ്പിക്കുക എന്നത് ബുദ്ധി മുട്ടുള്ള പ്രവർത്തിയാണ്. NLP കമ്പ്യൂട്ടർ രണ്ടു കാര്യങ്ങൾ ചെയ്യുന്നു സ്വാഭാവിക ഭാഷ മനസ്സി ലാക്കുന്നു (Natural Language Understanding (NLU)), സ്വാഭാവിക ഭാഷ ഉൽപ്പാദിപ്പിക്കുന്നു (Natural Language Generation(NLG)). ഇംഗ്ലീഷ്, മലയാളം തുടങ്ങിയ ഏതൊരു സ്വാഭാവിക ഭാഷയും മനസ്സിലാക്കുകയും കിട്ടുന്ന ഇൻപുട്ടിനെ യുക്തിഭദ്രമായി സമീപിക്കുകയും ചെയ്യുന്ന രീതിയാണ് സ്വാഭാവിക ഭാഷ മനസിലാക്കുക (NLU) എന്ന പ്രക്രിയ. സ്വാഭാവിക ഭാഷ ഉൽപ്പാദനം (NLG) ഔട്ട്പുട്ട് നിർമാണം കൈകാര്യം ചെയ്യുന്നു. ഇതു കൂടാതെ സംഭാഷണ ശകലങ്ങളെ വാക്കു കളായി എഴുതുന്ന ശബ്ദം തിരിച്ചറിയൽ (voice recognition) സംവിധാനവും നിലവിലുണ്ട്.

അന്താരാഷ്ട്ര സാഹചര്യങ്ങളുമായി താരതമ്യം ചെയ്യുമ്പോൾ ഇന്ത്യയിലെ സ്വാഭാവിക ഭാഷ പ്രക്രിയ ശൈശവാവസ്ഥയിലാണ്. ഇന്ത്യ ഒരു ബഹുഭാഷ രാഷ്ട്രമായതിനാൽ ഇതിന്റെ വിദൂര സാധ്യതകൾ തിരിച്ചറിഞ്ഞ ഭാരത സർക്കാരിന്റെ ഇലക്ട്രോണിക്സ് വിഭാഗത്തിന്റെ കീഴിലുള്ള DOE (Department of Eletcronics), TDIL തുടങ്ങിയ സ്ഥാപനങ്ങൾ സ്വാഭാവികഭാഷ പ്രക്രി യയെ ആധാരമാക്കിയുള്ള പ്രോഗ്രാമുകൾ വികസിപ്പിക്കുവാൻ തുടങ്ങി.

E. സ്വയം സംഭാഷണം ഗ്രഹിക്കുക (Automatic Speech Recognition (ASR)):

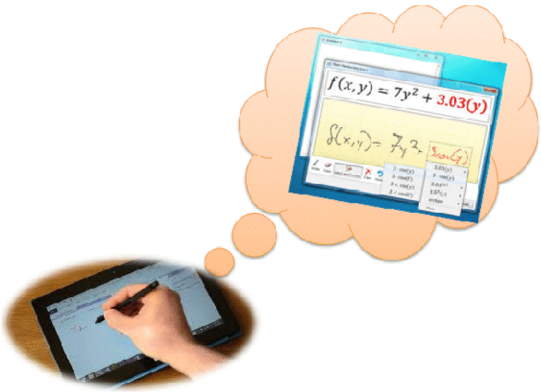
മലയാളം പോലുള്ള സ്വാഭാവിക ഭാഷകൾ ഉപയോഗിച്ച് കമ്പ്യൂട്ടറുമായി ആശയവിനിമയം നടത്തുവാനുള്ള നിർമ്മിത ബുദ്ധിയിലെ രീതികളെ സ്വാഭാവിക ഭാഷ പ്രക്രിയ എന്ന് പറയുന്നു. സ്വാഭാവിക ഭാഷാ പ്രക്രിയയിലെ അതിവേഗം വികാസം പ്രാപിച്ചു കൊണ്ടിരിക്കുന്ന സാങ്കേ തികവിദ്യയാണ് സംഭാഷണം ഗ്രഹിക്കുക (Automatic speech recognition) എന്നത്. വാണിജ്യപര മായി ഏറെ സാധ്യതയുള്ള ഒരു സംവിധാനമാണിത്. മനുഷ്യർക്കിടയിലുള്ള ആശയവിനിമയ മേഖലയിൽ സംസാരഭാഷ മേധാവിത്വം പുലർത്തുന്നു. അതുകൊണ്ട് ജനങ്ങൾ കമ്പ്യൂട്ടറുമായി ആശയവിനിമയം നടത്തുവാൻ സ്വാഭാവികമായും പ്രാദേശിക ഭാഷയിൽ ഉള്ള ഒരു മാധ്യമം പ്രതീക്ഷിക്കും. മൈക്രോഫോണിലോ ടെലിഫോണിലോ ഉപയോക്താക്കൾ നടത്തുന്ന സംഭാഷ ണങ്ങൾ കമ്പ്യൂട്ടർ മനസ്സിലാക്കുകയും അവയെ വാക്കുകളായി എഴുതുവാനുള്ള സ്വയം സംഭാഷണ ഗ്രഹിക്കൽ (Automatic Speech Recognition (ASR)) സംവിധാനം വികസിപ്പിച്ചാണ് ഈ ലക്ഷ്യം കൈവരിക്കുന്നത്. തൽഫലമായി മനുഷ്യനും കമ്പ്യൂട്ടറും തമ്മിലുള്ള പ്രധാന ആശയവിനിമയ രീതിയാക്കുവാൻ ഇതിന് കഴിയും. കമ്പ്യൂട്ടറുമായി വിനിമയം നടത്തേണ്ട ഏതു പ്രക്രിയയ്ക്കും ASR ഉപയോഗപ്പെടുത്താവുന്നതാണ്. കേട്ടെഴുത്ത് (Dictation), നിർദ്ദേശങ്ങളും നിയന്ത്രണങ്ങളും, ശബ്ദ ഡയലിംഗ്, ശബ്ദം ഉപയോഗിച്ച് ഡാറ്റാബേസിലെ ഡാറ്റയുടെ തിരഞ്ഞെ ടുക്കൽ, ഓഫീസ് ആവശ്യങ്ങൾക്കായുള്ള കേട്ടെഴുത്തു ഉപകരണങ്ങൾ, വിദേശ ഭാഷയിലേ ക്കുള്ള സ്വയം പരിഭാഷ തുടങ്ങിയവ ASR സംവിധാനം പിന്തുണയ്ക്കുന്ന ചില വിലപ്പെട്ട ആപ്ലി ക്കേഷനുകൾ ആണ്. ചിത്രം 11.15 ൽ മനുഷ്യനും കമ്പ്യൂട്ടറും തമ്മിൽ ASR വഴിയുള്ള സമ്പർക്ക ത്തിന്റെ ഉദാഹരണം കാണിച്ചിരിക്കുന്നു.

ചിത്രം 11.15: ASR വഴിയുള്ള സമ്പർക്കം

Apple iOS ന്റെ ഭാഗമായ 'Siri' എന്ന മൊബൈൽ അപ്ലിക്കേഷൻ, മൈക്രോസോഫ്ട് ഫോണിന്റെ Cortana, ആൻഡ്രോയിഡിന്റെ ഗൂഗിൾ നൗ എന്ന അപ്ലിക്കേഷൻ തുടങ്ങിയവ മികച്ച വ്യക്തിഗത സഹായികൾ ആണ്. ഇവ വാക്കാലുള്ള ചോദ്യങ്ങൾക്കു ഉത്തരം പറയുവാനും, അഭിപ്രായങ്ങൾ നൽകുവാനും, പ്രവർത്തനങ്ങൾ നടത്തുവാനും സഹായിക്കുന്നു

F ഒപ്റ്റിക്കൽ ക്യാരക്റ്റർ റെകഗ്നിഷൻ (Optical Character Recognition - OCR) / ഹാൻഡ്റിട്ടേൻ ക്യാരക്റ്റർ റെകഗ്നിഷൻ (Handwritten Character Recognition - HCR)

വളരെ വേഗത്തിലും കൃത്യമായും മനസ്സിൽ പതിയുന്ന രീതിയിൽ ചിത്രങ്ങളുടെ രൂപരേഖകളെ തിരിച്ചറിയുവാൻ കഴിവുള്ളവയാണ് നമ്മുടെ ദൃശ്യ ഘടന. നമ്മുടെ ചുറ്റുപാടുമുള്ള പലതരത്തിലുള്ള രൂപരേഖകളെയും വസ്തുക്കളെയും വലിയ ബുദ്ധിമുട്ടില്ലാതെ തന്നെ നമ്മൾ ഗ്രഹിക്കുന്നു. OCR ഉം HCR ഉം സാധാരണ സന്ദർഭങ്ങളിൽ രൂപരേഖകളെ കൃത്യമായി ഗ്രഹിക്കുവാൻ ഉപയോഗിക്കുന്ന പ്രക്രിയയുടെ അവിഭാജ്യഘടകങ്ങൾ ആയി മാറിയിരിക്കുകയാണ്. രൂപരേഖകൾ തിരിച്ചറിയുവാനുള്ള ഗവേഷണത്തിന്റെയും വികാസത്തിന്റെയും തുടക്കത്തിൽ OCR ഉം ആയി ബന്ധപ്പെട്ട ഗവേഷണങ്ങൾക്കാണ് കൂടുതൽ മുൻതൂക്കം നൽകിയിരിക്കുന്നത്. ഇത്തരം സോഫ്റ്റ് വെയറുകൾ, പ്രിന്റ് ചെയ്ത വാചകങ്ങളെ സ്കാൻ ചെയ്ത് ചിത്രങ്ങളാക്കി (അക്കങ്ങൾ, അക്ഷരങ്ങൾ, ചിഹ്നങ്ങൾ) കമ്പ്യൂട്ടർ പ്രക്രിയയ്ക്ക് വിധേയമാക്കുവാൻ കഴിയുന്ന രൂപത്തിലേക്ക് (ASCII പോലെ) മാറ്റുന്നു. നിലവിൽ മിക്കഭാഷകളിലും സാമാന്യം ഭേദപ്പെട്ട നല്ല OCR പാക്കേജുകൾ ലഭ്യമാണ്. ആധുനിക കമ്പ്യൂട്ടറുകൾ കണ്ടെത്തുന്നതിന് മുൻപ് തന്നെ, കയ്യെഴുത്തു വാചകങ്ങളുടെ സ്വയം തിരിച്ചറിയൽ എന്നത് ശാസ്ത്രകാരന്മാർക്കും എഞ്ചിനീയർമാർക്കും വെല്ലുവിളി നിറഞ്ഞ ഗവേഷണ മേഖലയായിരുന്നു. കയ്യെഴുത്തു രേഖകളെ സൃഷ്ടിക്കുന്നതിനെ ആധാരമാക്കി HCR-ൽ ഓൺലൈൻ എന്നും ഓഫ്‌ലൈൻ എന്നും രണ്ടു തരത്തിലുള്ള സമീപനരീതികൾ ഉണ്ട്. ഇലക്ട്രോണിക് പ്രതലത്തിൽ പ്രത്യേകതരം പേന ഉപയോഗിച്ച് എഴുതുന്ന അവസരത്തിൽ തന്നെ ഡാറ്റക്ക് രൂപം കൊടുക്കുന്ന രീതിയാണ് ആദ്യത്തേത്. ചിത്രം 11.16 ൽ ഇത് ചിത്രീകരിച്ചിരിക്കുന്നു. എഴുതിയതിന് ശേഷം സ്കാനർ ഡാറ്റയെ ആർജിക്കുന്ന രീതിയാണ് രണ്ടാമത്തേത്.



ചിത്രം 11.16: കൈയ്യെഴുത്ത് തിരിച്ചറിയൽ സംവിധാനം

വിദേശഭാഷകളിലും ഇന്ത്യൻ ഭക്ഷകളിലും ധാരാളം ആശാവഹമായ ഗവേഷണ ഫലങ്ങൾ HCR മേഖലയിൽ ഉണ്ടായിട്ടുണ്ട്. മൊബൈലിൽ ഉപയോഗിക്കുന്ന ഗൂഗിളിന്റെ ഓൺലൈൻ കയ്യെഴുത്തു തിരിച്ചറിയൽ അപ്ലിക്കേഷൻ ആയ 'ഗൂഗിൾ ഹാൻഡ് റൈറ്റിംഗ് ഇൻപുട്ട്' 82 ഓളം ഭാഷകളിലെ കൈയ്യെഴുത്തുകൾ തിരിച്ചറിയുവാൻ കഴിവുള്ളതാണ്.

G. ബയോഇൻഫോർമാറ്റിക്സ് (Bioinformatics)

കമ്പ്യൂട്ടറിന്റെ സഹായത്താൽ ജീവശാസ്ത്രപരമായ വിവരങ്ങളെ ബയോഇൻഫോർമാറ്റിക്സ് കൈകാര്യം ചെയ്യുന്നു. ജീവശാസ്ത്രപരമായ വിവരങ്ങളും ജനിതക വിവരങ്ങളും കമ്പ്യൂട്ടറിന്റെ സഹായത്താൽ ശേഖരിക്കുകയും, സംഭരിക്കുകയും, വിശകലനം ചെയ്യുകയും, ബന്ധിപ്പിക്കുകയും ചെയ്യുന്നു ഇവയെ ജീനുകളെ അടിസ്ഥാനമാക്കിയ ഔഷധ നിർമാണത്തിനും വികസനത്തിനും ഉപയോഗിക്കുന്നു. മനുഷ്യ ജനിതക പദ്ധതിയുടെ ഫലമായി ഉരുത്തിരിഞ്ഞ കണ്ടുപിടുത്തങ്ങളെ വിശകലനം ചെയ്യുന്നതിലും പൊതുജനങ്ങളിൽ എത്തിക്കുന്നതിലും ബയോ ഇൻഫോർമാറ്റിക്സ് എന്ന ശാസ്ത്രശാഖ ശ്രദ്ധേയമായ പങ്കുവഹിക്കുന്നു.

ബയോഇൻഫോർമാറ്റിക്സിന് പ്രധാനമായും മൂന്നു ലക്ഷ്യങ്ങൾ ഉണ്ട്. ഗവേഷകർക്ക് നിലവിലുള്ള വിവരങ്ങൾ ഉപയോഗിക്കുവാൻ കഴിയുന്നവിധത്തിൽ വിവരങ്ങൾ, ക്രമപ്പെടുത്തി സൂക്ഷിക്കുകയും അവർ കണ്ടെത്തുന്ന പുതിയ വിവരങ്ങൾ സമർപ്പിക്കുവാനുമുള്ള സംവിധാനം ഒരുക്കുകയും ചെയ്യുക എന്നതാണ് ബയോഇൻഫോർമാറ്റിക്സിന്റെ ആദ്യ ലക്ഷ്യം. രണ്ടാമത്തെ ലക്ഷ്യം ഈ ഡാറ്റയെ വിശകലനം ചെയ്യുന്നതിനുള്ള ഉപകരണങ്ങളും ഉപാധികളും വികസിപ്പിക്കുക എന്നതാണ്. ഈ ഉപകരണങ്ങൾ ഉപയോഗിച്ച് ജീവശാസ്ത്രപരമായി അർഥവത്തായ രീതിയിൽ ഡാറ്റയെ വിശകലനവും പരിവർത്തനവും ചെയ്യുക എന്നതാണ് മൂന്നാമത്തെ ലക്ഷ്യം.

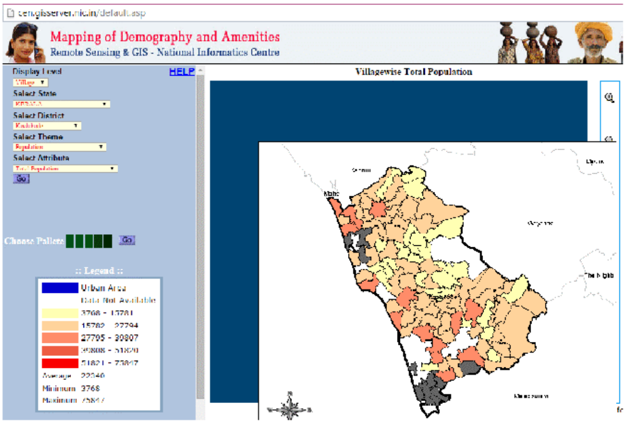
മനുഷ്യ ജനിതക പദ്ധതി (ഹ്യൂമൻ ജീനോം പ്രൊജക്റ്റ്)

മനുഷ്യ ജനിതകഘടനയെയും അവയിൽ ഉൾപ്പെട്ടിട്ടുള്ള ജീനുകളെയും തിരിച്ചറിയുക എന്ന പ്രയത്നമാണ് ഇത് ലക്ഷ്യമാക്കുന്നത്. എല്ലാ ജീനുകളെയും ചേർത്ത് നമ്മുടെ ജനിതകഘടന (Genome) എന്ന് അറിയപ്പെടുന്നു. ഈ പദ്ധതിയെ ഏകോപിപ്പിച്ചിരുന്നത് യുണൈറ്റഡ് സ്റ്റേറ്റ്സ് ഡിപ്പാർട്മെന്റ് ഓഫ് എനർജിയും നാഷണൽ ഇൻസ്റ്റിറ്റ്യൂട്ട് ഓഫ് ഹെൽത്തുമാണ്. ഒരു വ്യക്തിയുടെ പ്രാഥമിക രേഖാരൂപം ഗവേഷകർക്ക് കിട്ടുവാൻ മനുഷ്യ ജനിതക പദ്ധതി ഗവേഷകരെ സഹായിച്ചു. ഈ ഗവേഷണത്തിന് ജീനുകളുടെ പ്രവർത്തനങ്ങളെയും പ്രോട്ടീനുകളെയും കുറിച്ചുള്ള കൂടുതൽ വിവരങ്ങൾ കമ്പ്യൂട്ടറിന്റെ വിപുലമായ ഉപയോഗത്താൽ കണ്ടെത്തുവാൻ കഴിഞ്ഞു. ഈ അറിവ് വൈദ്യശാസ്ത്രം, ജൈവ വസാങ്കേതികവിദ്യയിലും, ജീവശാസ്ത്രരംഗം തുടങ്ങിയ മേഖലകളിൽ വളരെ വലിയ സ്വാധീനം സൃഷ്ടിക്കുവാൻ കഴിയും.

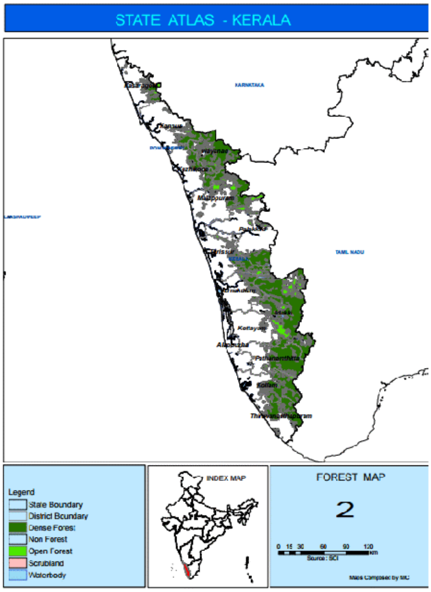
H ഭൂവിവരവ്യവസ്ഥ (Geographic Information System)

ഭൂമിയെ ഭ്രമണം ചെയ്യുന്ന ധാരാളം ഉപഗ്രഹങ്ങൾ ഉണ്ടെന്ന് നമുക്ക് അറിയാം. ഇവയുടെ രൂപകൽപ്പന, നിർമ്മാണം, വിക്ഷേപണം എന്നിവ ഭൂമിയുടെ വിവിധതരം ചിത്രങ്ങൾ തുടർച്ചയായി ഭൂമിയിലേക്ക് കൈമാറുക എന്ന ഉദ്ദേശത്തോടുകൂടിയാണ്. ഇങ്ങനെ കിട്ടുന്ന വിദൂര സംവേദന വിവരങ്ങളാണ്, നഗരാസൂത്രണത്തിനും, കാർഷിക ആവശ്യങ്ങൾക്കും, ദുരന്ത നിവാരണത്തിനും, ക്രൂഡോയിൽ സാന്നിദ്ധ്യം കണ്ടെത്തുവാനുമുള്ള സംവിധാനങ്ങളിൽ ഉപയോഗിക്കുന്നത്. ഡിജിറ്റൽ കാർട്ടോഗ്രാഫി (DIGITAL CARTOGRAPHY), കമ്പ്യൂട്ടർ എയ്ഡഡ് ഡിസൈൻ (CAD) എന്നിവയിൽ നിന്നാണ് ജിസ് (Geographic Information System) സാങ്കേതികവിദ്യ വികസിപ്പിച്ചത്.

ഭൂമിയുടെ ഉപരിതല സവിശേഷതകളും സ്ഥാനീയ വിവരങ്ങളും ഒരു സെർവറിൽ ഭൂപടങ്ങളിലെ വിവരങ്ങളായി ഉൾക്കൊള്ളിച്ച് സംഭരിച്ചു വെയ്ക്കുന്നു. ഈ ഡാറ്റബേസിൽ ആവശ്യാനുസരണം മാറ്റങ്ങൾ വരുത്തുക, വിശകലനം ചെയ്യുക, ക്രോഡീകരിക്കുക തുടങ്ങിയ ജോലികൾ ചെയ്യുന്നതിനായി തയ്യാറാക്കിയ സാങ്കേതിക വിദ്യയാണ് ഭൂവിവര വ്യവസ്ഥ എന്ന് പറയുന്നത്. ഭൂമിയുടെ ഉപരിതലത്തിലെ വിവിധ ഭാഗങ്ങളുടെ വിവരങ്ങൾ ശേഖരിക്കുവാനും സംഭരിക്കുവാനും, പരിശോധിക്കുവാനും പ്രദർശിപ്പിക്കുവാനുമുള്ള കമ്പ്യൂട്ടർ സംവിധാനമാണ് GIS. പല തരത്തിലുള്ള ഡേറ്റ ഒരു ഭൂപടത്തിൽ GIS-ന് പ്രദർശിപ്പിക്കാൻ കഴിയും. ചുരുക്കത്തിൽ, ഭൂമിയെ സംബന്ധിക്കുന്ന വിവരങ്ങൾ കമ്പ്യൂട്ടറിൽ ശേഖരിച്ചുവെച്ച് ആവശ്യാനുസരണം ഉപയോഗപ്പെടുത്തുന്നു. പലതരത്തിലുള്ള വിവരങ്ങളെ ഒരു ഭൂപടത്തിൽ പ്രദർശിപ്പിക്കുവാൻ ഭൂവിവരവ്യവസ്ഥ ഉപയോഗിക്കാം. ജനങ്ങൾക്ക് മനസ്സിലാകുന്ന വിധത്തിൽ രേഖാചിത്രങ്ങളും, അവ തമ്മിലുള്ള ബന്ധങ്ങളും എളുപ്പത്തിൽ ചിത്രീകരിക്കുവാനും പരിശോധിക്കുവാനും ഈ ആപ്ലിക്കേഷൻ ഉപയോഗിക്കുന്നു. ചിത്രം 11.17 (a) യിൽ കോഴിക്കോട് ജില്ലയിലെ ഓരോ വില്ലേജിലെയും ജനസംഖ്യയുടെ വിന്യാസം ചിത്രീകരിച്ചിരിക്കുന്നു.



ചിത്രം 11.17(a): GIS ഉപയോഗിച്ചുള്ള കോഴിക്കോട് ജില്ലയുടെ ജനസംഖ്യ വിന്യാസം



ചിത്രം 11.17(b): GIS ഉപയോഗിച്ചുള്ള കേരളത്തിലെ വനങ്ങളുടെ രേഖാചിത്രം

മണ്ണിന്റെ വിന്യാസഘടനയുടെ ചിത്രീകരണം, വനങ്ങളുടെ വിന്യാസ ചിത്രീകരണം (ചിത്രം 11.17 (b)), കാർഷികമേഖലയുടെ ചിത്രീകരണം, ഇ-ഗവർണനൻസ്, ജലവിഭവ പരിപാലനം, പ്രകൃതി ദുരന്ത പരിപാലനം തുടങ്ങിയ വിവിധ മേഖലകളിൽ ഭൂവിവരവ്യവസ്ഥ ഉപയോഗിക്കുന്നു. 'അങ്ങനെ സംഭവിച്ചാൽ' എന്ന അവസ്ഥയെ കൈകാര്യം ചെയ്യുവാനുള്ള കഴിവ്, പദ്ധതികളുടെ ശേഷി വർധിപ്പിക്കുക, മേന്മയേറിയ ഗുണനിലവാര വിശകലം എന്നിവയാണ് ഭൂവിവരവ്യവസ്ഥയുടെ പ്രയോജനങ്ങൾ. സുസ്ഥിര വികസന പദ്ധതികളായ നഗരവികസന പദ്ധതികൾ, അടിസ്ഥാന സൗകര്യ ആസൂത്രണം, കൃത്യമായ കാർഷിക പദ്ധതികളുടെ ആസൂത്രണം എന്നിവയ്ക്ക് ഭൂവിവര വ്യവസ്ഥ ആപ്ലിക്കേഷനുകൾ ഉപയോഗിക്കുന്നു.

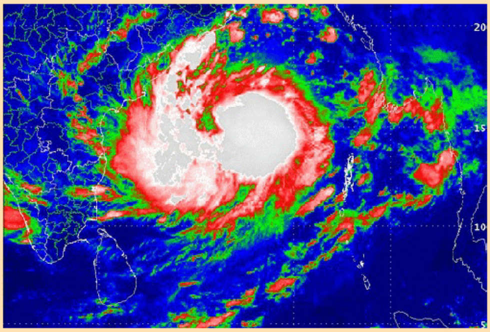
വിദൂര സംവേദനം (റിമോട്ട് സെൻസിംഗ്)

അകലെ നിന്ന് കാര്യങ്ങൾ മനസ്സിലാക്കുക എന്നതാണ് വിദൂര സംവേദനം കൊണ്ട് ഉദ്ദേശിക്കുന്നത്. വസ്തുക്കളുമായി നേരിട്ടു ബന്ധപ്പെടാതെ തന്നെ, ഒരു വസ്തുവിന്റെ വിവരങ്ങൾ കണ്ടെത്തുന്ന ശാസ്ത്രവും കലയുമാണ് വിദൂര സംവേദനം.

ഉപഗ്രഹ വിദൂര സംവേദനത്തിൽ സംവേദന ഉപകരണങ്ങൾ ഉപയോഗിച്ച് മനുഷ്യന് ദൃശ്യമാകുന്ന വൈദ്യുത കാന്തിക വർണ്ണരാജിയുടെ വിവിധ തലത്തിലെ പ്രസരണങ്ങൾ പിടിച്ചെടുക്കുന്നു. ഭൂമിയുടെ ഉപരിതലം, സമുദ്രം, അന്തരീക്ഷം എന്നിവയുടെ വസ്തുതകളും സവിശേഷതകളും ലഭിക്കുന്നതിനു വേണ്ടിയാണ് ഈ സാങ്കേതികവിദ്യ ഉപയോഗിക്കുന്നത്.

നമുക്ക് കാലാവസ്ഥ പ്രവചനത്തിന്റെ കാര്യം പരിഗണിക്കാം. കാലാവസ്ഥ ഉപഗ്രഹങ്ങൾ വഹിക്കുന്ന ഉപകരണമായ റേഡിയോ മീറ്ററർസ് എന്ന ഉപകരണം, ഭൂമിയെ സ്കാൻ ചെയ്തു ചിത്രങ്ങൾ ഉണ്ടാക്കുന്നു. ഈ ഡാറ്റ വിശകലനം ചെയ്ത് നമ്മുടെ കാലാവസ്ഥ ശാസ്ത്രജ്ഞന്മാർ കാലാവസ്ഥ പ്രവചിക്കുന്നു

കാലാവസ്ഥ ശാസ്ത്രജ്ഞന്മാർ മഴയും താപനിലയും മാത്രമല്ല ഭൂകമ്പ സാധ്യത, ചുഴലിക്കാറ്റ്, കൊടുംകാറ്റ്, എന്നിവയും പ്രവചിക്കുന്നു. ഇന്ത്യ മെറ്റീറോളജിക്കൽ ഡിപ്പാർട്മെന്റ് (India Meteorological Department) ഇന്ത്യയിൽ സ്ഥാപിച്ചിരിക്കുന്നത് ഈ ഉദ്ദേശ്യത്തോടു കൂടിയാണ്. ഭാരത സർക്കാരിന്റെ കീഴിലുള്ള ഭൗമശാസ്ത്ര മന്ത്രാലയത്തിന്റെ ശാഖയാണിത്. കാലാവസ്ഥ നിരീക്ഷണം, കാലാവസ്ഥ പ്രവചനം, ഭൂകമ്പ ശാസ്ത്രം തുടങ്ങിയവയുടെ ഉത്തരവാദിത്വം ഉള്ള പ്രധാന ഏജൻസിയാണിത്. 2014 ഒക്ടോബറിൽ ബംഗാൾ ഉൾക്കടലിൽ രൂപം കൊണ്ട ഹുഡ് ഹുഡ് ചുഴലികൊടുംകാറ്റിന്റെ ഉപഗ്രഹചിത്രം ശ്രദ്ധിക്കുക (ചിത്രം 11.18)



ചിത്രം 11.18: ബംഗാൾ ഉൾക്കടലിന്റെ മുകളിലായി രൂപം കൊണ്ട ഹുഡ് ഹുഡ് എന്ന ചുഴലിക്കാറ്റിന്റെ ഉപഗ്രഹ ചിത്രം

നിങ്ങളുടെ പുരോഗതി അറിയുക

1 OCR, HCR എന്നീ സംവിധാനങ്ങൾ തമ്മിലുള്ള വ്യത്യാസമെന്ത്?

2 ജിസ് കൊണ്ട് എന്താണ് ഉദ്ദേശിക്കുന്നത്?

3 റോബോട്ടുകളുടെ രൂപകൽപ്പന, നിർമ്മാണം, സിദ്ധാന്തം, ഉപയോഗം എന്നിവയുമായി ബന്ധപ്പെട്ട ശാസ്ത്രപഠനത്തെ________ എന്ന് വിളിക്കുന്നു.

4 എന്താണ് NLP (Natural Language Processing)?

5 കമ്പ്യൂട്ടർ ദൃഷ്ടി (VISION) എവിടെ ഉപയോഗിക്കുന്നു എന്നതിനു ഒരു ഉദാഹരണം എഴുതുക?

6 ANN ന്റെ പൂർണരൂപം ____________ ആണ്

നമുക്ക് സംഗ്രഹിക്കാം

ഡിസ്ട്രിബ്യൂറ്റഡ് കമ്പ്യൂട്ടിങ്ങിലേക്കുള്ള സമീപകാലത്തെ മാറ്റം ശൃംഖലാടിസ്ഥാനത്തിലുള്ള കമ്പ്യൂട്ടിങ്ങിനെ ശരിയായി മനസ്സിലാക്കുന്നതിന്റെ ആവശ്യകത വർധിപ്പിച്ചു. ഡിസ്ട്രിബ്യൂറ്റഡ് പ്ലാറ്റ്ഫോം, ഭൂമിശാസ്ത്രപരമായി വിന്യസിച്ച ശൃംഖലകൾ മുതൽ പരസ്പരം ബന്ധിപ്പിച്ച പ്രത്യേകമായ ഘടനാരൂപകൽപ്പനകളെ വരെ തമ്മിൽ ബന്ധിപ്പിച്ചിരിക്കുന്നു. പാരലൽ കമ്പ്യൂട്ടിങ്ങിൽ ഒരു പ്രശ്നത്തെ വിവിധ CPU കളിലായി വിഭജിച്ച് കൊടുക്കുന്നു. ക്ലസ്റ്ററുകൾ കുറെ കമ്പ്യൂട്ടറുകളെ ചേർത്ത് വലിയ കേന്ദ്രീയമായി നിയന്ത്രിക്കുന്ന ഒരൊറ്റ അസ്തിത്വമായി നിലനിർത്തുന്നു. ഗ്രിഡ് കമ്പ്യൂട്ടിങ്ങിൽ ഓരോ നോഡിനും മറ്റു നോഡുകളിലെ സംവിധാനങ്ങളെ അവ പ്രാദേശികമാണ് എന്ന പോലെ ഉപയോഗിക്കാം. ഇന്റർനെറ്റ് വഴി ആപ്ലിക്കേഷനും, സേവനങ്ങളും ലഭ്യമാക്കുന്നതിനെയാണ് ക്ലൗഡ് കമ്പ്യൂട്ടിങ് എന്ന് പറയുന്നത്. ഭാവിയിൽ കുറഞ്ഞ വിലയിൽ അതിവേഗ ശൃംഖലകൾ വഴി പ്രോസസ്സറിന്റെ ശക്തി കമ്പ്യൂട്ടിങ്ങിന് ലഭ്യമാകും എന്നത് വ്യക്തമാണ്. യന്ത്രമോ സോഫ്ട്വെയറോ പ്രകടമാക്കുന്ന ബുദ്ധിയാണ് നിർമ്മിത ബുദ്ധി എന്നത് ബുദ്ധി സൃഷ്ടിക്കുക എന്നത് ഈ പഠനമേഖലയുടെ ലക്ഷ്യമാണ്. ജിസ് സോഫ്ട്വെയർ പ്രധാനമായും ഉപയോഗിക്കുന്നത് ഭൗമശാസ്ത്രപരമായ ഡാറ്റ ആസൂത്രണം ചെയ്യുവാനും ക്രമീകരിക്കുവാനുമാണ്. വിദൂരസംവേദനം വസ്തുക്കളുടെ വിവരങ്ങൾ ശേഖരിക്കുന്ന ഒരു രീതിയാണ്. ഇത് വസ്തുക്കളുമായി നേരിട്ടു ബന്ധപ്പെടാതെയും, നേരിട്ടു സ്ഥല നിരീക്ഷണം നടത്താതെയുമുള്ള രീതിയാണ്. കാർഷിക മേഖല, ഭൂഗർഭ ശാസ്ത്രം, കാലാവസ്ഥ, വാനശാസ്ത്രം തുടങ്ങിയ വിവിധ മേഖലകളിൽ വിദൂര സംവിധാനം ഉപയോഗിക്കുന്നു.

നമുക്ക് വിലയിരുത്താം

1 ഡിസ്ട്രിബ്യൂറ്റഡ് കമ്പ്യൂട്ടിങ്ങിന്റെ സവിശേഷതകൾ വിവരിക്കുക

2 പാരലൽ കമ്പ്യൂട്ടിങ്ങും സീരിയൽകമ്പ്യൂട്ടിങ്ങും തമ്മിൽ താരതമ്യം ചെയ്യുക?

3 ഡിസ്ട്രിബ്യൂറ്റഡ് കമ്പ്യൂട്ടിങ്ങിന്റെ മാതൃക ഉപയോഗിക്കുന്ന നിലവിലുള്ള ഹാർഡ്വെയർ ... ആണ്.

4 ക്ലസ്റ്റർ കമ്പ്യൂട്ടിങ്ങിന്റെ മേന്മകൾ എഴുതുക

5 നാം കമ്പ്യൂട്ടറുകൾ ഉപയോഗിക്കുന്ന രീതിയെ എങ്ങനെയാണു ക്ലൗഡ് സേവന മാതൃകൾ മാറ്റുന്നത് വിശദമാക്കുക.

6 SaaS ന്റെ സേവനങ്ങൾ നൽകുന്ന സ്ഥാപനങ്ങൾക്ക് ഉദാഹരണങ്ങൾ എഴുതുക

7 ക്ലൗഡ് കമ്പ്യൂട്ടിങ്ങിന്റെ മേന്മകളും പരിമിതികളും എഴുതുക.

8 അറിവിന്റെ പിരമിഡ് ചിത്രീകരിക്കുകയും വിശദമാക്കുകയും ചെയ്യുക?

9 എന്താണ് കമ്പ്യൂട്ടേഷണൽ ബുദ്ധിവൈഭവം ?

10 സൈബർനെറ്റിക്സ് നിർവചിക്കുക?

11 ഒരു കമ്പ്യൂട്ടറിന് ബുദ്ധിയുണ്ടോ ഇല്ലയോ എന്ന് തീരുമാനിക്കുന്ന പരീക്ഷണത്തെ .... എന്ന് വിളിക്കുന്നു

12 എന്താണ് ANN ?

13 സ്വാം ഇന്റലിജൻസ് വിശദമാക്കുക

14 ഫസ്സി സിസ്റ്റം വിശദമാക്കുക

15 ബയോമെട്രിക്സിന്റെ ചില ഉപയോഗങ്ങൾ എഴുതുക

16 റോബോട്ടിക്കിന്റെ ഉപയോഗങ്ങൾ എഴുതുക?

U1G2V2

12 വിവര ആശയവിനിമയ സാങ്കേതികവിദ്യയും സമൂഹവും

പ്രധാന പഠന നേട്ടങ്ങൾ

ഈ അധ്യായത്തിന്റെ പഠനം പൂർത്തിയാകുന്നതോടെ പഠിതാവ് ആർജിക്കേണ്ട പഠനനേട്ടങ്ങൾ

- ഇ-ഭരണം ഇ-വ്യാപാരം ഇ-പഠനം എന്നിവയുടെ സൗകര്യങ്ങൾ പട്ടികപ്പെടുത്തുന്നു.
- ഇ-പഠനത്തിനുപയോഗിക്കുന്ന ഉപകരണങ്ങൾ വിശദീകരിക്കുന്നു.
- ഇ-ഭരണം ഇ-വ്യാപാരം ഇ-പഠനം എന്നീ ആപ്ലിക്കേഷനുകൾ നടപ്പാക്കുമ്പോൾ അഭിമുഖീകരിക്കുന്ന വിവിധ മേൻമകളും വെല്ലുവിളികളും പട്ടികപ്പെടുത്തുന്നു.
- ഓരോ മേഖലയിലെയും ഉപകാരപ്രദമായ ചില വെബ് സൈറ്റുകൾ ഉപയോഗിക്കുന്നു.
- സൈബർ ഇടത്തെകുറിച്ച് വിശദീകരിക്കുന്നു.
- വിവിധ തരത്തിലുള്ള സൈബർ കുറ്റകൃത്യങ്ങൾ വേർതിരിക്കുന്നു.
- വിവിധ ബൗദ്ധിക സ്വത്തവകാശങ്ങൾ വിശദീകരിക്കുകയും പട്ടികപ്പെടുത്തുകയും ചെയ്യുന്നു.
- സൈബർ നീതിശാസ്ത്രവും (ethics) നിയമവും വിശദീകരിക്കുന്നു.
- ഐടി നിയമത്തിന്റെ പ്രാധാന്യം തിരിച്ചറിയുന്നു.
- ഇൻഫോമാനിയയെക്കുറിച്ച് മനസിലാക്കുന്നു.

കഴിഞ്ഞ കുറച്ച് ദശാബ്ദങ്ങളായി ഇന്റർനെറ്റിന്റെ ഉപയോഗം അതിവേഗം വർദ്ധിച്ചിട്ടുണ്ട്. ഇന്റർനെറ്റും അതിന്റെ സേവനങ്ങളും ഇല്ലാത്ത ഒരു ദിവസത്തെക്കുറിച്ച് ആധുനിക സമൂഹത്തിന് ചിന്തിക്കുവാൻ സാധ്യമല്ല. ATM ഇല്ലാത്ത ബാങ്കിങ്ങ്, ഇവാണിജ്യം ഉപയോഗിക്കാത്ത വ്യപാരം, കമ്പ്യൂട്ടർ ഉപയോഗിച്ച് രോഗനിർണ്ണയം നടത്താത്ത ആശുപത്രികൾ വിവര സാങ്കേതിക വിദ്യാധിഷ്ഠിതമായ ഉപകരണങ്ങൾ ഇല്ലാത്ത വിദ്യാഭ്യാസം മുതലായവയെകുറിച്ച് നമുക്ക് ചിന്തിക്കാനാകുമോ? തീർച്ചയായും വിവര ആശയ വിനിമയ സാങ്കേതിക വിദ്യ നമ്മുടെ ജീവിതത്തെ വളരെ എളുപ്പമാക്കി. ഇന്റർനെറ്റ് ഏറ്റവും വലിയ ആശയ വിനിമ മാധ്യമമാണെന്ന് നാം പഠിച്ചു. സാമൂഹിക മാധ്യമങ്ങൾ സമൂഹത്തിൽ വലിയ സ്വാധീനം ചെലുത്തിയിട്ടുണ്ട്. സർക്കാർ സേവനങ്ങൾ ജനങ്ങളിലെക്കെത്തിക്കാൻ ഇഭരണം പുതിയ വഴികൾ തുറന്നു. ഇ-വാണിജ്യം വ്യാപാര ഇടപാടുകളെ വളരെ എളുപ്പത്തിലാക്കുകയും ഇ-പഠനം പഠനത്തിനും അധ്യാപന മേഖലക്കും പുതിയ വിജ്ഞാന മണ്ഡലം തുറക്കുകയും ചെയ്തു. വിവര സാങ്കേതികവിദ്യ (IT) യുടെ വിപുലീകരിച്ച രൂപമായി വിവര ആശയ വിനിമയ സാങ്കേതിക വിദ്യ (ICT) പലപ്പോഴും ഉപയോഗിക്കുന്നു. ടെലി കമ്യൂണിക്കേഷനെ സംയോജിപ്പിക്കുവാനും വിവരങ്ങൾ ച്ചടിക്കുവാനും വ്യാഖ്യനിക്കുവാനും പ്രക്രിയ ചെയ്യുവാനും, സംഭരിക്കുവാനും, ഡാറ്റ പിടിച്ചെടുക്കുവാനും കമ്പ്യൂട്ടർ സാങ്കേതിക വിദ്യ ഉപയോഗിച്ച് സാധിക്കുന്നു. ഡാറ്റ പിടിച്ചെടുക്കുക, സംഭരിക്കു

ക, പ്രക്രിയ ചെയ്യുക, വ്യാഖ്യാനിക്കുക, അച്ചടിക്കുക എന്നിവയ്ക്കുള്ള സാങ്കേതിക വിദ്യകൾ അടങ്ങിയ കമ്പ്യൂട്ടറുകളും ടെലികമ്യൂണിക്കേഷനും തമ്മിലുള്ള സംയോജനമാണ് ICT യുടെ പ്രത്യേകത.

ഭൂമി, വീട് തുടങ്ങിയവയിലുള്ള അവകാശം പോലെ തന്നെ സംഗീതം സിനിമ, സോഫ്റ്റ് വെയർ, രൂപകൽപ്പനകൾ എന്നിവക്കും ഉടമസ്ഥാവകാശങ്ങളുണ്ട്. ഈ അവകാശങ്ങളെ ബൗദ്ധിക സ്വത്തവകാശം എന്നു വിളിക്കുന്നു. ഇതുമായി ബന്ധപ്പെട്ട കാര്യങ്ങൾ ഈ അധ്യായത്തിൽ ചർച്ച ചെയ്തിരിക്കുന്നു. എല്ലാ സാങ്കേതിക വിദ്യകൾക്കും ഒരു ഇരുണ്ട വശമുണ്ട്. ഇന്റർനെറ്റിനും അതിന്റേതായ പ്രശ്നങ്ങളും ഭീഷണികളുമുണ്ട്. ധാരാളം ആളുകൾ ഈ ഉപാധിയെ നിയമവിരുദ്ധമായ പ്രവർത്തനങ്ങൾക്ക് ഉപയോഗിക്കുന്നു. അതുപോലെയുള്ള പ്രവർത്തനങ്ങളെ കുറിച്ചും ഇന്റർനെറ്റിൽ എങ്ങനെ സുരക്ഷാ സംവിധാനം ഒരുക്കാമെന്നും നമുക്ക് ചർച്ച ചെയ്യാം.

12.1 ICT സേവനങ്ങൾ (ICT Services)

വിവര സാങ്കേതിക വിദ്യയുടെ പ്രചാരം കമ്പ്യൂട്ടറുകളുടെയും ആശയ വിനിമയ സാങ്കേതിക വിദ്യയുടെയും സഹായത്തോടു കൂടി ലഭ്യമായ സേവനങ്ങളുടെ എണ്ണം വർധിപ്പിക്കുന്നതിന് കാരണമായി. ഉപയോഗിക്കുന്ന നടപടി ക്രമങ്ങളിൽ മാറ്റം വന്ന സേവനങ്ങളിൽ ചിലത് നാം ഇവിടെ ചർച്ച ചെയ്യുന്നു.

12.1 ഇ-ഭരണം (e-Governance)

സർക്കാരിന്റെ പ്രവർത്തനങ്ങൾ വളരെ അനായാസമാക്കുന്നതിന് ഐ ടി വലിയ പങ്ക് വഹിക്കുന്നുണ്ട്. വിവരങ്ങൾ ശേഖരിക്കുക, സംസ്കരിക്കുക, വിശദീകരിക്കുക, സംഭരിക്കുക, കൈമാറ്റം ചെയ്യുക എന്നിവയ്ക്ക് വേണ്ടിയുള്ള പല സാങ്കേതിക വിദ്യകൾ ഇതിൽ ഉൾപ്പെടുന്നു. വളരെ പെട്ടെന്ന് ന്യായമായ തീരുമാനങ്ങൾ എടുക്കുവാൻ ഇത് സർക്കാരുകളെ സഹായിക്കുന്നു. എല്ലാ സർക്കാർ സേവനങ്ങളിലും സുതാര്യതയും ഉത്തരവാദിത്തവും വർധിപ്പിക്കുന്നു. ഐ സി ടി ഉപയോഗിച്ച് കൊണ്ടുള്ള ഭരണ നിർവ്വഹണം ആണ് ഇഭരണം എന്ന പേരിൽ പൊതുവെ അറിയപ്പെടുന്നു.

സർക്കാർ സേവനങ്ങൾ സൗകര്യപ്രദവും, കാര്യക്ഷമവും, സുതാര്യവുമായ രീതിയിൽ പൗരന്മാരിലേക്ക് എത്തിക്കുവാൻ ഐ സി ടി ഉപയോഗിച്ച് കൊണ്ടുള്ള ഇ-ഭരണത്തിന് സാധിക്കുന്നു. സർക്കാർ സേവനങ്ങൾ ഫലപ്രദമായ രീതിയിൽ തൽക്ഷണം ജനങ്ങളിലേക്ക് എത്തുന്നു എന്ന് ഉറപ്പു വരുത്തുകയാണ് ഇതിന്റെ പ്രധാന ലക്ഷ്യം. കേരളത്തിൽ മോട്ടോർ വാഹന വകുപ്പ്, വിദ്യാഭ്യാസ വകുപ്പ്, റവന്യൂ വകുപ്പ് തുടങ്ങിയവയിൽ ഇ-ഭരണം വളരെ വിജയകരമായ രീതിയിൽ നടപ്പിലാക്കിയിട്ടുണ്ട്. ഇനി നമുക്ക് ഇഭരണത്തിലെ വിവിധ വ്യവഹാരങ്ങളെ കുറിച്ച് ഇപ്പോൾ ചർച്ച ചെയ്യാം.

A. ഇ-ഭരണത്തിലെ വിവിധതരം വ്യവഹാരങ്ങൾ (Types of Interactions in e-Governance)

ഭരണ നിർവ്വഹണത്തിൽ വിവിധതരം പങ്കാളികളുമായുള്ള പരസ്പര വ്യവഹാരങ്ങൾ ഇ ഭരണം വഴി സുഗമമാകുന്നു. ഇത്തരത്തിലുള്ള വിവിധതരം വ്യവഹാരങ്ങളെ കുറിച്ച് ചുവടെ പ്രതിപാദിക്കുന്നു.

സർക്കാരിൽ നിന്നും സർക്കാരിലേക്ക്/Government to Government (G2G)

സർക്കാർ സംവിധാനത്തിലുള്ള ഏജൻസികൾ, വകുപ്പുകൾ, സംഘടനകൾ എന്നിവയുടെ ഇടയിൽ നടക്കുന്ന ഇലക്ട്രോണിക് രൂപത്തിലുള്ള ഡാറ്റയുടെയും വിവരങ്ങളുടെയും പങ്കുവയ്ക്കലാണിത്. ആശയ വിനിമയം, ഡാറ്റയുടെ ഉപയോഗം, ഡാറ്റ പങ്കുവയ്ക്കൽ എന്നിവ മെച്ചപ്പെടുത്തി ഇ-ഭരണ സംരംഭങ്ങളെ പിൻതുണക്കുക എന്നതാണ് G2G യുടെ ലക്ഷ്യം.

സർക്കാരിൽ നിന്നും പൗരന്മാരിലേക്ക്/Government to Citizens (G2C)

ഇത് സർക്കാരിനും പൗരന്മാർക്കുമിടയിൽ സമ്പർക്കം സാധ്യമാക്കുന്നു. ഇവിടെ പൊതുജനങ്ങൾ ധാരാളം പൊതുസേവനങ്ങൾ ഉപയോഗപ്പെടുത്തുന്നു. പൊതുസേവനങ്ങളുടെ ലഭ്യതയും ഉപയോഗക്ഷമതയും ഇത് വർധിപ്പിക്കുന്നു. സേവനങ്ങളുടെ ഗുണനിലവാരവും ഇത് ഉയർത്തുന്നു. സർക്കാരിനും പൗരന്മാർക്കും ഇടയിൽ ഒരു സൗഹൃദ വലയം ഒരുക്കുക എന്നതാണ് ഇതിന്റെ പ്രധാന ഉദ്ദേശം.

സർക്കാരിൽ നിന്നും വ്യാപാരത്തിലേക്ക്/Government to Business (G2B)

ഇവിടെ ഇഭരണ സംവിധാനം വ്യാപാര സമൂഹത്തിന് സർക്കാരുമായി സംവദിക്കുവാനുള്ള അവസരം ഒരുക്കുന്നു. ചുവപ്പുനാട പ്രവണത ഒഴിവാക്കുക, സമയനഷ്ടം പരിഹരിക്കുക, പ്രവർത്തന ചെലവ് കുറയ്ക്കുക, സർക്കാരുമായുള്ള വ്യാപാര ഇടപാടുകളിൽ കൂടുതൽ സുതാര്യത വരുത്തുക എന്നിവയാണ് ഇതിന്റെ ലക്ഷ്യം.

സർക്കാരിൽ നിന്നും ജീവനക്കാരിലേക്ക് / Government to Employees (G2E)

സർക്കാരിന്റെ നയങ്ങൾ ജീവനക്കാരിലൂടെയാണ് നടപ്പിലാക്കുന്നത്. ഇതിനായി സർക്കാരും ജീവനക്കാരും പരസ്പരം പതിവായി സമ്പർക്കം പുലർത്തേണ്ടതുണ്ട്. ഗവൺമെന്റും ജീവനക്കാരും തമ്മിൽ തമ്മിൽ സമ്പർക്കം പുലർത്തുന്നു. സർക്കാരിന്റെ വിവിധ പരിപാടികൾ നടപ്പിലാക്കുന്നതിനുള്ള നയങ്ങളും മാർഗ്ഗ നിർദ്ദേശങ്ങളും സർക്കാർ ഉത്തരവുകളുടെയും വിജ്ഞാപനങ്ങളുടെയും രൂപത്തിൽ ഇ-ഭരണ പോർട്ടലുകൾ വഴി ജീവനക്കാർക്ക് ലഭ്യമാക്കിയിട്ടുണ്ട്. ഇ-ഭരണ സേവനങ്ങൾ ഉപയോഗപ്പെടുത്തിക്കൊണ്ടാണ് സർക്കാർ ജീവനക്കാരുടെ ശമ്പളം, വ്യക്തിഗത വിവരങ്ങൾ എന്നിവ കൈകാര്യം ചെയ്യുന്നത്. (ഉദാ: സ്പാർക്ക് വെബ് സൈറ്റ് - www.spark.gov.in) ഐ സി ടിയുടെ സൗകര്യങ്ങൾ ഉപയോഗപ്പെടുത്തി ഇത്തരം പ്രവർത്തനങ്ങൾ വളരെ വേഗത്തിലും കാര്യക്ഷമതയോടെയും കൈകാര്യം ചെയ്യുവാൻ സാധിക്കുന്നു.

B. ഇ-ഭരണത്തിന്റെ അടിസ്ഥാന സൗകര്യം (e-Governance Infrastructure)

ഇഭരണത്തിന്റെ അടിസ്ഥാന സൗകര്യവും സംഭരണവും നൽകുന്നതിനായുള്ള സ്റ്റേറ്റ് ഡാറ്റ സെന്ററുകൾ (SDC), അവയെ ബന്ധിപ്പിക്കുന്നതിനായുള്ള സ്റ്റേറ്റ് വൈഡ് ഏരിയ നെറ്റ്വർക്ക് (SWAN), പൊതുജനങ്ങൾക്ക് സേവനങ്ങൾ എത്തിക്കുന്നതിനുള്ള പൊതുസേവന കേന്ദ്രങ്ങൾ (കോമൺ സർവീസ് സെന്ററുകൾ) എന്നീ ഘടകങ്ങളാണ് ഇന്ത്യയിലെ ഇഭരണത്തിന്റെ ഘടനയിൽ ഉൾപ്പെടുന്നത്. ഇ-ഭരണത്തിന്റെ ഈ മൂന്നു ഘടകങ്ങളുടേയും സംയോജനം ചിത്രം 10.1 ൽ കാണിച്ചിരിക്കുന്നു.

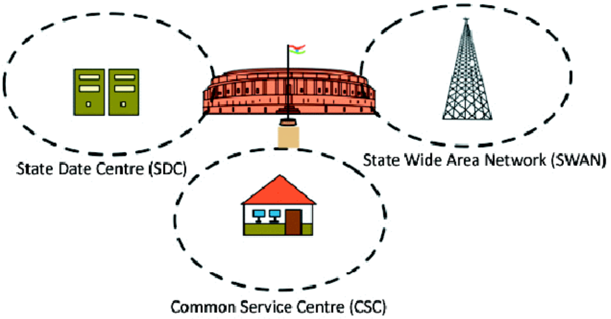


Fig. 12.1:ഇ-ഭരണത്തിന്റെ അടിസ്ഥാന സൗകര്യം/e-Governance infrastructure

i. സ്റ്റേറ്റ് ഡാറ്റ സെന്റർ / State Data Cetnre (SDC)

ഇഭരണം സാധ്യമാക്കുന്നതിനു ആവശ്യമായ ഒരു പ്രധാന ഘടകമാണ് സ്റ്റേറ്റ് ഡാറ്റാ സെന്റർ. ദേശീയ ഇഭരണ പദ്ധതി/National e-Governance Plan (NeGP) മുൻകൈ എടുത്താണ് ഇത് രൂപപ്പെടുത്തിയത്. സേവനങ്ങൾ, ഉപയോഗങ്ങൾ, അടിസ്ഥാന സൗകര്യം എന്നിവയെ സംയോജിപ്പിക്കുകയും, G2G, G2C, G2B എന്നീ സേവനങ്ങൾക്ക് ഇലക്ട്രോണിക് ഡാറ്റാ വിതരണം കാര്യക്ഷമമായ രീതിയിൽ നടത്തുന്നതിനും വേണ്ടിയാണ് NeGP യുടെ കീഴിൽ സ്റ്റേറ്റ് ഡാറ്റ സെന്റർ നിർമ്മിക്കുവാൻ തീരുമാനിച്ചത്. ഒരു പൊതുവിതരണ സംവിധാനത്തിലൂടെ ഓരോ സംസ്ഥാന സർക്കാരിനും ഈ സേവനങ്ങൾ നിർവഹിക്കാവുന്നതാണ്. സ്റ്റേറ്റ് വൈഡ് ഏരിയ നെറ്റ്‌വർക്ക് (SWAN), ഗ്രാമങ്ങളിൽ വരെ വ്യാപിച്ചു കിടക്കുന്ന പൊതു സേവന കേന്ദ്രങ്ങൾ (CSC) മുതലായവ ഈ പൊതുവിതരണ സംവിധാനങ്ങൾക്ക് പിന്തുണ നൽകുന്നു. സ്റ്റേറ്റ് ഡാറ്റാ സെന്റർ നിരവധി സേവനങ്ങൾ നൽകുന്നു. സംസ്ഥാനങ്ങളുടെ പരമപ്രധാനമായ ഡാറ്റ സംരക്ഷിക്കുക, വിവിധ സേവനങ്ങൾക്കു വേണ്ടിയുള്ള ഓൺലൈൻ സംവിധാനം ഒരുക്കുക, സംസ്ഥാന ഇൻട്രാനെറ്റ് സംവിധാനത്തിന്റെ കവാടമാകുക, പൗരൻമാർക്കു സേവനങ്ങളുടെ കവാടമാകുക, ദുരന്ത നിവാരണ സംവിധാനമൊരുക്കുക മുതലായ പ്രവർത്തനങ്ങൾ സ്റ്റേറ്റ് ഡാറ്റാ സെന്റർ എന്നിവ അവയിൽ ഉൾപ്പെടുന്നു. മാനേജ്മെന്റ് നിയന്ത്രണങ്ങൾ, ഡാറ്റ മാനേജ്മെന്റിന്റെ മൊത്തം ചെലവ് കുറയ്ക്കൽ, വിഭവ നിർവ്വഹണം, വിന്യാസം, തുടങ്ങിയവയെല്ലാം സ്റ്റേറ്റ് ഡാറ്റ സെന്ററുകൾ നൽകുന്ന സൗകര്യങ്ങളാണ്.

ii. കേരള സ്റ്റേറ്റ് വൈഡ് ഏരിയ നെറ്റ്‌വർക്ക് /Kerala State Wide Area Network (KSWAN)

KSWAN, സ്റ്റേറ്റ് ഇൻഫർമേഷൻ ഇൻഫ്രാസ്ട്രക്ച്ചറിന്റെ/(State Information Infrastructure (SII)) നട്ടെല്ലായി പ്രവർത്തിക്കുന്നു. ഇത് തിരുവനന്തപുരം, കൊച്ചി, കോഴിക്കോട് എന്നീ പ്രധാന നഗരങ്ങളെ താൽപര്യകേന്ദ്രങ്ങളാക്കി (Hubs) 14 ജില്ലകളിലെ 152 ബ്ലോക്ക് പഞ്ചായത്തുകളെ തമ്മിൽ ബന്ധിപ്പിക്കുന്നു. ഈ ശൃംഖല വഴി അനേകം സർക്കാർ കാര്യാലയങ്ങളേയും ബന്ധിപ്പിക്കുന്നു. സ്റ്റേറ്റ് ഡാറ്റ സെന്ററിൽ നിന്നുള്ള അപേക്ഷകൾക്ക് അനുസരിച്ചു G2G, G2C സേവനങ്ങളെ ഏകോപിപ്പിച്ചു പ്രവർത്തിക്കുക എന്നത് കേരള സ്റ്റേറ്റ് വൈഡ് ഏരിയ നെറ്റ്‌വർക്ക് (KSWAN) ന്റെ പ്രധാന ചുമതലയാണ്.

iii. പൊതുസേവന കേന്ദ്രങ്ങൾ / Common Service Cetnre (CSC)

സർക്കാർ, സ്വകാര്യ, സാമൂഹ്യ വകുപ്പുകളുടെ സേവനങ്ങൾ രാജ്യത്തെ ഗ്രാമീണർക്ക് എത്തിക്കുന്നതിനുള്ള സേവന കേന്ദ്രങ്ങളാണ് പൊതു സേവന കേന്ദ്രങ്ങൾ. വെബ് അധിഷ്ഠിത ഇ-ഭരണ സേവനങ്ങൾ ഗ്രാമ പ്രദേശങ്ങളിൽ എത്തിക്കുന്നു എന്നതാണ് **CSC** യുടെ പ്രധാന സവിശേഷത. ഈ സംവിധാനം ഉപയോഗപ്പെടുത്തിക്കൊണ്ട് നമുക്ക് വൈദ്യുതി ബിൽ, ടെലിഫോൺ ബിൽ, ജലഉപയോഗ ബിൽ എന്നിവ അടയ്ക്കുവാൻ സാധിക്കുന്നു. കൂടാതെ അപേക്ഷകൾ ഓൺലൈൻ ആയി സമർപ്പിക്കുവാനും ആവശ്യക്കാർക്ക് യോഗ്യതാപത്രം (certificate) തയ്യാറാക്കി വിതരണം ചെയ്യുവാനും ഇത് വഴി സാധിക്കുന്നു. പൊതു സേവനകേന്ദ്രങ്ങൾ വഴി ലഭ്യമാകുന്ന മറ്റു സേവനങ്ങൾ ചുവടെ ചേർക്കുന്നു.

- കാർഷിക സേവനങ്ങൾ
- വിദ്യാഭ്യാസവും പരിശീലനവുമായി ബന്ധപ്പെട്ട സേവനങ്ങൾ
- ആരോഗ്യ സേവനങ്ങൾ
- ഗ്രാമീണ ബാങ്കിന്റെയും ഇൻഷുറൻസിന്റെയും സേവനങ്ങൾ

- വിനോദ സംബന്ധമായ സേവനങ്ങൾ
- വാണിജ്യപരമായ സേവനങ്ങൾ

കേരളത്തിൽ അക്ഷയ സെന്ററുകൾ പൊതു സേവന കേന്ദ്രങ്ങളായി പ്രവർത്തിക്കുന്നു.

അക്ഷയ സെന്ററുകൾ

കേരളത്തിലെ മലപ്പുറം ജില്ലയിൽ 2002 ൽ അക്ഷയ സെന്ററുകൾക്ക് തുടക്കം കുറിച്ചു. സാധാരണക്കാരായ പൗരന്മാരിൽ ഇ-സാക്ഷരത (e-literacy) അല്ലെങ്കിൽ കമ്പ്യൂട്ടർ സാക്ഷരത ഉണ്ടാക്കിയെടുക്കുക എന്നതാണ് ഇതിന്റെ ഉദ്ദേശം. കേരള സംസ്ഥാന വിവര സാങ്കേതിക വിദ്യ മിഷൻ /Kerala State Information Technology Mission (KSITM) ന്റെ ഒരു അതിരടയാളം (landmark) ആയാണ് അക്ഷയ സെന്ററുകൾ രൂപകൽപന ചെയ്തിരിക്കുന്നത്. ഐ.സി.ടി യുടെ പ്രയോജനങ്ങൾ ഡിജിറ്റൽ വേർതിരിവില്ലാതെ സംസ്ഥാനത്തെ എല്ലാ ജനവിഭാഗങ്ങളിലേക്കും എത്തിക്കുക എന്നതാണ് ഇതുകൊണ്ടു ഉദ്ദേശിക്കുന്നത്. ഇഗ്രാന്റ്സ് (e-gratnz), ഇഫയലിംഗ് (e-filing), ഇ-ഡിസ്ട്രിക്റ്റ് (e-district), ഇടിക്കറ്റിങ് (e-ticketing) എന്നിവ ഇതിന്റെ സേവനങ്ങളിൽ ചിലതാണ്. അതുപോലെ റേഷൻ കാർഡ്, തെരഞ്ഞെടുപ്പ് ഐ ഡി എന്നിവയുടെ അപേക്ഷ സമർപ്പിക്കുവാനും, ആധാർ കാർഡിൽ പേര് ചേർക്കുവാനും, മറ്റ് ആധാർ സേവനങ്ങൾക്കും, ഇൻഷുറൻസ് ബാങ്കിങ് സേവനങ്ങൾക്കുമെല്ലാം നമുക്ക് അക്ഷയ സെന്ററുകളെ ഉപയോഗപ്പെടുത്താം.

2002 നവംബർ 18ന് മുൻ രാഷ്ട്രപതി
ഡോ. എ പി ജെ അബ്ദുൽകലാം അക്ഷയ പദ്ധതി ഉദ്ഘാടനം ചെയ്തു.

സ്വകാര്യ പൊതുപങ്കാളിത്ത / Private Public Partnership (PPP Model) മാതൃകയിൽ ആണ് അക്ഷയ കേന്ദ്രങ്ങൾ പ്രവർത്തിക്കുന്നത്. അക്ഷയ സെന്ററിന്റെ ഉടമസ്ഥാവകാശം ഒരു സ്വകാര്യ വ്യക്തിക്കാണെങ്കിൽ പോലും തദ്ദേശ സ്വയംഭരണ വകുപ്പാണ് ഈ വ്യക്തിയെ തെരഞ്ഞെടുക്കുന്നത്. അക്ഷയ സെന്റർ പ്രവർത്തിക്കുന്ന പഞ്ചായത്തിലെ വ്യക്തികൾക്കാണ് മുൻഗണന. ഉടമസ്ഥന്റെ വരുമാനം എന്നത് ഓരോ ഇടപാടിനും പൗരന്മാരിൽ നിന്നും ഈടാക്കുന്നതോ അല്ലെങ്കിൽ സർക്കാരിൽ നിന്നും ലഭിക്കുന്നതോ ആയ നാമമാത്രമായ സേവന വേതനം മാത്രമാണ്.

അക്ഷയ
അവസരങ്ങളുടെ ജാലകം
www.akshaya.kerala.gov.in

ഭാരത സർക്കാരിന്റെ കീഴിലുള്ള യുണീക് ഐഡന്റിഫിക്കേഷൻ അതോറിറ്റി ഓഫ് ഇന്ത്യ / Unique Identification Authortiy of India (UIDAI), ഇ-ഭരണ സംവിധാനത്തിന്റെ ഭാഗമായി പൗരന്മാർക്ക് നൽകുന്ന 12 അക്ക വ്യക്തിഗത തിരിച്ചറിയൽ സംഖ്യയാണ് ആധാർ. ഇന്ത്യയിൽ എവിടെയും ഒരു തിരിച്ചറിയൽ രേഖയായും, മേൽ വിലാസത്തിന്റെ തെളിവ് ആയും ഇത് ഉപയോഗിക്കാം. ഇന്ത്യയിലെ ഏതൊരു പൗരനും പ്രായ ലിംഗഭേദം ഇല്ലാതെ ആധാറിൽ പേര് രേഖപ്പെടുത്താവുന്നതാണ്. ഓരോ വ്യക്തിക്കും സൗജന്യമായി ഒരു പ്രാവശ്യം പട്ടികയിൽ ചേർക്കാവുന്നതാണ്. പേര് ചേർക്കുന്ന സമയത്ത് പൗരന്റെ അടിസ്ഥാന വിവരങ്ങൾ (പേര്, വയസ്സ്, ലിംഗം, വിലാസം), ബയോമെട്രിക് വിശദാംശങ്ങൾ (ഫോട്ടോ, 10 വിരലിന്റെയും 2 കൃഷ്ണമണികളുടെയും അടയാളം) എന്നിവ കൃത്യമായി രേഖപ്പെടുത്തുന്നതാണ്. ഓരോ ആധാർ നമ്പറും അനന്യവും വ്യക്തിയുടെ ജീവിതാവസാനം വരെ നിയമ സാധുതയുള്ളതുമായിരിക്കും. ബാങ്കിങ് സേവനങ്ങൾ പ്രയോജനപ്പെടുത്തുവാനും , മൊബൈൽഫോൺ കണക്ഷൻ എടുക്കുന്നതിനും മറ്റു സർക്കാർ, സർക്കാർ ഇതര സേവനങ്ങൾക്കും ആധാർ നമ്പർ ഉപയോഗപ്പെടുത്താം.

C. ഇ-ഭരണത്തിന്റെ പ്രയോജനങ്ങൾ

ഇ-ഭരണം പ്രയോഗത്തിൽ വരുത്തുന്നത് കൊണ്ട് ധാരാളം മേന്മകൾ ഉണ്ട്. സർക്കാരിന്റെ പ്രവർത്തനങ്ങളെ പറ്റിയും, സർക്കാർ നടപ്പിൽ വരുത്താൻ ഉദ്ദേശിക്കുന്ന നയപരിപാടികളെ കുറിച്ചുമുള്ള അവബോധം ജനങ്ങളിൽ സൃഷ്ടിച്ചെടുക്കുവാൻ ഇ-ഭരണത്തിന് കഴിയുന്നു. ഇഭരണത്തിന്റെ പ്രധാന പ്രയോജനങ്ങൾ ചുവടെ ചേർക്കുന്നു.

- ഇ-ഭരണം സർക്കാർ സർവീസുകളിൽ കമ്പ്യൂട്ടർ വത്കരണം സാധ്യമാക്കുന്നു. ജനക്ഷേമ പ്രവർത്തനങ്ങളുമായി ബന്ധപ്പെട്ട വിവരങ്ങൾ എളുപ്പത്തിൽ ജനങ്ങളിൽ എത്തിച്ചേരുന്നു എന്ന് ഉറപ്പു വരുത്തുന്നു.
- ഇ-ഭരണം സർക്കാരിന്റെ എല്ലാ തലങ്ങളിലും പൗരന്മാരുടെ ഇടപെടലുകൾ ഉറപ്പു വരുത്തിക്കൊണ്ട് ജനാധിപത്യം ശക്തിപ്പെടുത്തുന്നു.
- ഇത് സർക്കാർ പ്രവർത്തനങ്ങൾ കൂടുതൽ സുതാര്യമാക്കുകയും അഴിമതി തടയാൻ സഹായിക്കുകയും ചെയ്യുന്നു.
- സൂക്ഷ്മമായി നിരീക്ഷിക്കപെടുന്നു എന്ന ബോധ്യം ഉള്ളതിനാൽ എല്ലാ സർക്കാർ വകുപ്പുകളുടെയും പ്രവർത്തനങ്ങൾ കാര്യക്ഷമവും ഉത്തരവാദിത്തപൂർണവും ആകുന്നു.
- ഇ-ഭരണം കാര്യക്ഷമമായി പ്രയോഗത്തിൽ വരുത്തുന്നതിലൂടെ പൊതുജനത്തിന് നിരന്തരമായി കാര്യാലയങ്ങൾ സന്ദർശിക്കേണ്ട അവസ്ഥ ഒഴിവാകുന്നു. ഇത് സമയവും സാമ്പത്തും ലാഭിക്കുന്നു.

D. ഇ-ഭരണത്തിന്റെ വെല്ലുവിളികൾ (Challenges to e-Governance)

ഇ-ഭരണത്തിന് ഒരുപാട് മേന്മകൾ ഉണ്ടെങ്കിലും, ചില വെല്ലുവിളികളും ഇതിന് അഭിമുഖീകരിക്കേണ്ടി വരുന്നു. ഇ-ഭരണം പ്രയോഗത്തിൽ വരുത്തുമ്പോൾ അഭിമുഖീകരിക്കേണ്ടി വരുന്ന ചില വെല്ലുവിളികൾ എന്തൊക്കെ എന്ന് നോക്കാം.

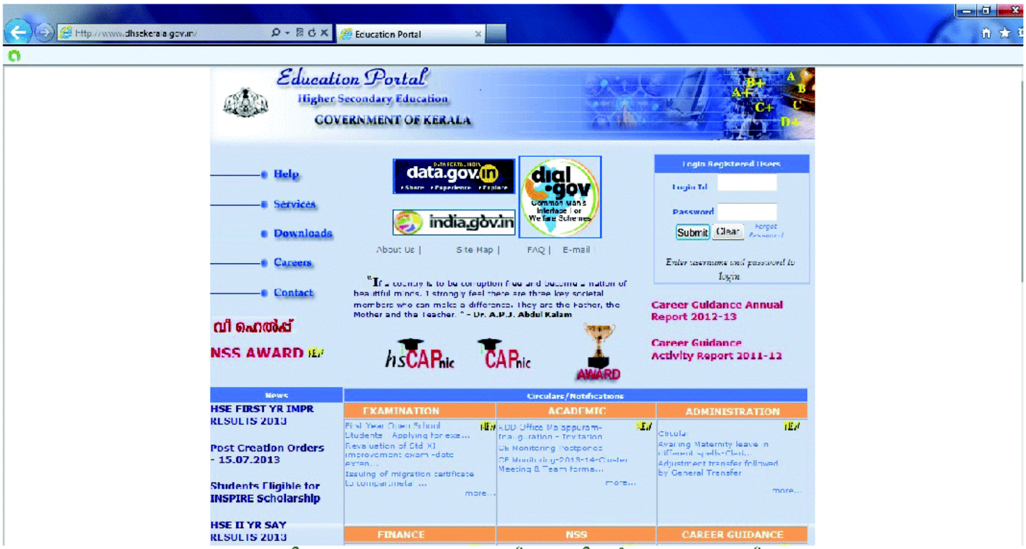
- ഡിജിറ്റൽ വിഭജനം മൂലം ഇ-ഭരണ സേവനം കൃത്യമായി ഉപയോഗപ്പെടുത്തുന്നതിന് ചില തടസ്സങ്ങൾ നേരിടുന്നു. വിദൂര സ്ഥലങ്ങളിൽ ജീവിക്കുന്ന കമ്പ്യൂട്ടർ സാക്ഷരത കുറഞ്ഞ ആളുകൾക്ക് ഇഭരണ സേവനങ്ങൾ പ്രയോജനപ്പെടുത്തുന്നതിന് ബുദ്ധിമുട്ടുകൾ നേരിടുന്നു.
- ഇ-ഭരണ സംവിധാനം കമ്പ്യൂട്ടർ/വെബ് അധിഷ്ഠിതമായതിനാൽ മികച്ച രീതിയിലുള്ള സുരക്ഷാ സംവിധാനങ്ങൾ ആവശ്യമാണ്. അല്ലാത്ത പക്ഷം സൈബർ ആക്രമണങ്ങൾക്കുള്ള സാധ്യത വളരെ കൂടുതലാണ്.
- ഇ-ഭരണ സേവനങ്ങൾ പ്രയോഗത്തിൽ വരുത്തുന്നതിനും അതിന്റെ പരിപാലനത്തിനും വലിയ തോതിലുള്ള മുതൽമുടക്കും ആസൂത്രണവും ആവശ്യമാണ്.
- വ്യക്തിഗത വിവരങ്ങളും, ബയോമെട്രിക് ഡാറ്റയും നിർദ്ദിഷ്ട ഏജൻസികളുമായി പങ്കുവയ്ക്കുന്നതിൽ ഭൂരിഭാഗം ആളുകളും ഉത്കണ്ഠാകുലരാണ്.
- ഇ-ഭരണം കാര്യശേഷി ഉള്ളതും ഫലപ്രാപ്തവുമാകണമെങ്കിൽ വിവിധ വകുപ്പുകളുടെ ഏകോപനം വളരെ അത്യാവശ്യമാണ്.

E. ഇഭരണവുമായി ബന്ധപ്പെട്ട വെബ്സൈറ്റുകൾ (Useful e-Governance websites)

കേരള സർക്കാരിന്റെ വിവിധ സേവനങ്ങൾ ജനങ്ങളിലേക്ക് എത്തിക്കുന്നതിന് വേണ്ടിയുള്ള ചില വെബ്സൈറ്റ് വിലാസങ്ങൾ പട്ടിക 10.1 ൽ കാണിച്ചിരിക്കുന്നു. ചിത്രം 10.2 ൽ DHSE. ചിത്രം 10.3ൽ ഇ-ഡിസ്ട്രിക്ട് പ്രധാന പേജും കാണിച്ചിരിക്കുന്നു.

വെബ്സൈറ്റുകളും മേൽവിലാസവും	ഉപയോഗം / സേവനം
www.dhsekerala.gov.in	കേരള സർക്കാരിന്റെ കീഴിലുള്ള ഹയർ സെക്കണ്ടറി വിദ്യാഭ്യാസ വകുപ്പിന്റെ ഔദ്യോഗിക വെബ്സൈറ്റ്. വിദ്യാർത്ഥികൾക്കും അധ്യാപകർക്കും സ്കൂൾ ഭരണാധികാരികൾക്കും ആവശ്യമായ വിവിധ സൗകര്യങ്ങളും സേവനങ്ങളും ഇത് ലഭ്യമാക്കുന്നു.
www.edistrict.kerala.gov.in	ഇത് സർക്കാർ സേവനങ്ങൾ പൊതുജനങ്ങളിലേക്ക് എത്തിക്കുന്നു.
www.incometaxindia.gov.in	ഇത് ആദായനികുതി വകുപ്പിന്റെ സേവനങ്ങൾ ലഭ്യമാക്കുന്നു.
www.keralamvd.gov.in	ഇതാണ് കേരള മോട്ടോർ വാഹന വകുപ്പിന്റെ ഔദ്യോഗിക വെബ്സൈറ്റ്.
www.rti.gov.in	സർക്കാരിൽ നിന്നോ അധികാരികളിൽ നിന്നോ വിവരങ്ങൾക്കായി അപേക്ഷ നൽകിയ പൗരന് വിവരാവകാശ നിയമം 2005 ഉത്തരവ് അനുസരിച്ച് സമയബന്ധിതമായി മറുപടി നൽകുന്നു.
www.itmission.kerala.gov.in	കേരള സർക്കാരിന്റെ ഈ വെബ്സൈറ്റ് വിവിധ ഐ ടി സംരംഭങ്ങൾക്ക് ഭരണപരമായ പിന്തുണ നൽകുന്നു.
www.spark.gov.in	ജീവനക്കാരന്റെ സേവന വേതന കാര്യങ്ങൾ കൈകാര്യം ചെയ്യുന്ന G2E വിഭാഗത്തിലുള്ള വെബ്സൈറ്റ്.

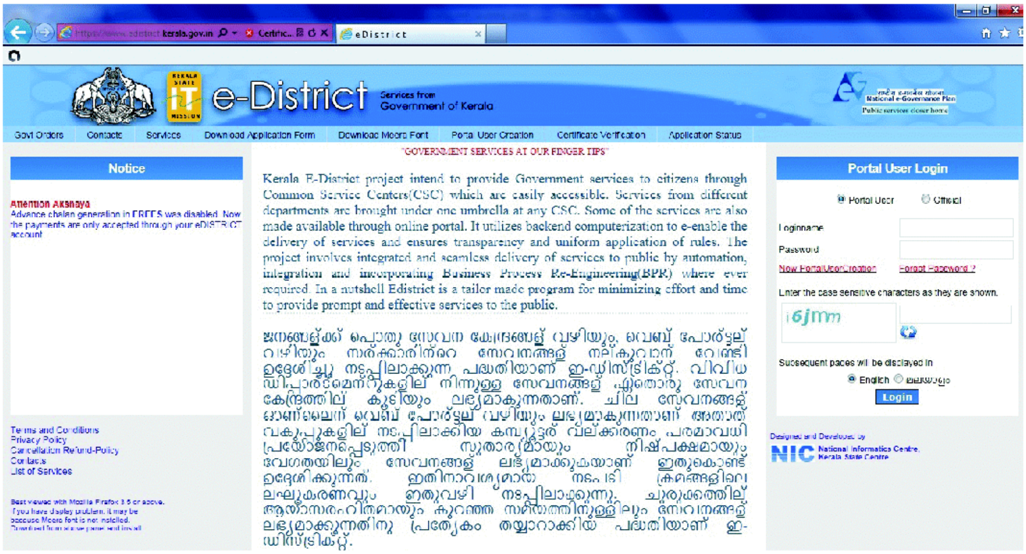
പട്ടിക 121 ഇ-ഭരണവുമായി ബന്ധപ്പെട്ട വെബ്സൈറ്റുകൾ



ചിത്രം 12.2: DHSE വെബ്സൈറ്റിന്റെ പ്രധാന പേജ്

www.dhsekerala.gov.in

ഹയർ സെക്കന്ററി പരീക്ഷകൾക്ക് രജിസ്റ്റർ ചെയ്യൽ, ഹാൾ ടിക്കറ്റ് വിതരണം, പരീക്ഷാഫല പ്രസിദ്ധീകരണം, വിവിധ ഉത്തരവുകളുടെയും വിജ്ഞാപനങ്ങളുടെയും വിതരണം മുതലായവ ഇതിന്റെ സേവന പരിധിയിൽ ഉൾപ്പെടുന്നു. വിവിധ ഫോറങ്ങളും, മാതൃക ചോദ്യ കടലാസുകൾ ഉൾപ്പെടെയുള്ള പഠന സാമഗ്രികളും ഡൗൺലോഡ് ചെയ്യാനുള്ള സൗകര്യം ഈ വെബ്സൈറ്റിൽ ലഭ്യമാണ്.



ചിത്രം 12.3: ഇ - ഡിസ്ട്രിക്ട് (e-District) പോർട്ടലിന്റെ പ്രധാന പേജ്

ദേശീയ ഇഭരണ പദ്ധതിയ്ക്ക് കീഴിൽ സംസ്ഥാനം ഏറ്റെടുത്തിരിക്കുന്ന ദൗത്യ നിർവ്വഹണ പദ്ധതി (State Mission Mode Project) ആണ് ഇഡിസ്ട്രിക്ട് (e-District). കമ്പ്യൂട്ടർവത്കൃത സേവനങ്ങൾ പൊതുജനങ്ങളിലെത്തിക്കുക എന്നതാണ് ഈ പദ്ധതിയുടെ ലക്ഷ്യം.

കേരളത്തിലെ റവന്യൂ വകുപ്പുമായി ബന്ധപ്പെട്ട വിവിധ യോഗ്യതാ പത്രങ്ങളുടെ (Certificates) വിതരണം ഇഡിസ്ട്രിക്ട് പോർട്ടലുമായി സമന്വയിപ്പിച്ചിരിക്കുന്നു ഇത് സർക്കാർ കാര്യാലയങ്ങളിലെ പല സങ്കീർണമായ വ്യവസ്ഥകളെയും ലഘൂകരിക്കുകയും പൗരന്റെ ജീവിത രീതിയെ എളുപ്പമാക്കുകയും ചെയ്യുന്നു. പൗരനെ സംബന്ധിച്ചിടത്തോളം പൊതുരേഖകളുടെ പകർപ്പുകൾ നേടുന്നതോ, അപേക്ഷകൾ സമർപ്പിക്കുന്നതോ, അധികാരികളെ കാണുന്നതോ, മറ്റു ദൈനം ദിനാവശ്യങ്ങൾക്കുള്ള വിവരങ്ങൾ ആരായുന്നതോ ബുദ്ധിമുട്ടേറിയ കാര്യമാണ്. ഇത് അവന്റെ/അവളുടെ യാത്രാ ചെലവ് വർധിപ്പിക്കുകയും, ദിവസവരുമാനവും സമയവും നഷ്ടപ്പെടുത്തുകയും ചെയ്യുന്നു. സർക്കാർ സ്ഥാപനങ്ങളിൽ ചിലപ്പോൾ ബന്ധപ്പെട്ട രേഖകളോ, വിവരങ്ങളോ അല്ലെങ്കിൽ അധികാരികളോ യഥാസമയം ലഭ്യമാകണമെന്നില്ല. ആയതിനാൽ സ്ഥാപനത്തിലേക്കുള്ള സന്ദർശനം ആവർത്തിക്കേണ്ടി വരികയും ചെലവ് വർധിക്കാനിടയാകുകയും ചെയ്യുന്നു. ആവശ്യമായ രേഖകൾ സഹിതം ഒരാൾക്ക് ഇഡിസ്ട്രിക്ട് വെബ് പോർട്ടലിൽ നേരിട്ടോ, അക്ഷയ കേന്ദ്രങ്ങൾ മുഖാന്തരമോ അപേക്ഷകൾ സമർപ്പിച്ച് സേവനങ്ങൾ തേടാവുന്നതാണ് . അക്ഷയ കേന്ദ്രങ്ങൾ പൊതു പൗരനുവേണ്ടി ഓൺലൈൻ അപേക്ഷകൾ സമർപ്പിക്കുന്നു. ഈ അപേക്ഷകൾ വിവിധ ഘട്ടങ്ങളിലൂടെ വില്ലേജ് ഓഫീസറുടെ അരികിൽ എത്തിച്ചേരുന്നു. ഈ അപേക്ഷകളിലുളള വില്ലേജ് നടപടിക്ക് ശേഷം ഓഫീസറുടെ തൊട്ടടുത്ത ഉന്നത അധികാരിക്ക് സമർപ്പിക്കുകയോ, അല്ലെങ്കിൽ അംഗീകരിച്ചതിനു ശേഷം ഡിജിറ്റൽ ഒപ്പ് സഹിതം ആവശ്യക്കാരന് വിതരണം ചെയ്യുകയോ ചെയ്യുന്നു. യോഗ്യതാ പത്രം ലഭ്യമായാൽ അപേക്ഷകനെ എസ് എം എസ് വഴി അറിയിക്കുകയും ചെയ്യുന്നു.

സ്വയം വിലയിരുത്താം

1. ഐ സി ടി ഉപയോഗിച്ച് സർക്കാർ സേവനങ്ങൾ സൗകര്യപ്രദവും, കാര്യക്ഷമവും, സുതാര്യവുമായ രീതിയിൽ പൗരന് എത്തിക്കുവാനുള്ള പ്രയോഗത്തിന്റെ പേരെന്ത്?
2. ഇ-ഭരണം എന്ന പദം നിർവ്വചിക്കുക.
3. ഭരണ നിർവ്വഹണത്തിൽ വിവിധതരം പങ്കാളികളുമായുള്ള പരസ്പര വ്യവഹാരങ്ങൾ ഇ-ഭരണം സുഗമമാക്കുന്നു. ഈ പ്രസ്താവന ശരിയോ തെറ്റോ എന്ന് പറയുക.
4. ഒരു ഇ-ഭരണ വെബ്സൈറ്റിന് ഉദാഹരണം നൽകുക.
5. എന്താണ് KSWAN?

12.1. 2 ഇ-വ്യാപാരം (e-Business)

നമ്മൾ പലപ്പോഴും സാധനങ്ങൾ വാങ്ങാൻ കടകളിൽ പോവുകയും അതിനായി ഒരുപാട് സമയവും അധ്വാനവും നീക്കി വെക്കുകയും ചെയ്യാറുണ്ട്. വീട് വിട്ട് പുറത്തിറങ്ങാതെ സാധനങ്ങൾ വാങ്ങുക എന്ന ആശയം എങ്ങനെയുണ്ട്? എന്നാൽ ഇന്ന് ഇതൊരു യാഥാർത്ഥ്യമാണ്. ഒരു വിധം എല്ലാ സേവനങ്ങളും വ്യാപാരവും ഇന്ന് ഓൺലൈനിൽ ലഭ്യമാണ്. അത് ഏതാനും മൗസ് ക്ലിക്കുകൾ മാത്രം അകലെയാണ്.

ഇ-വ്യാപാരം (e-Business) എന്നത് കൊണ്ട് ഉദ്ദേശിക്കുന്നത്, വ്യാപാര വിവരങ്ങൾ പങ്കുവയ്ക്കുക, വ്യാപാര ബന്ധങ്ങൾ നിലനിർത്തുക, ഐ സി ടി ഉപയോഗിച്ച് വ്യാപാര ഇടപാടുകൾ നടത്തുക എന്നിവയാണ്. കമ്പനികകൾ (വ്യാപാരം), പൊതു സ്ഥാപനങ്ങൾ (ഭരണസമിതി), വ്യക്തികൾ (ഉപഭോക്താവ്) മുതലായവർ ഇ-വ്യാപാരത്തിന്റെ ഭാഗമായി വരുന്നു. ഇ-വാണിജ്യ സേവനങ്ങൾ കാര്യക്ഷമമായി ഉപയോഗിച്ചുകൊണ്ട് മികച്ച രീതിയിലുള്ള വിപണനത്തിനും, വില്പന വർധനവിനും ഇ-വ്യാപാരം സഹായിക്കുന്നു. ഓൺലൈൻ കൂടിക്കാഴ്ചകൾ സംഘടിപ്പിക്കുക, പ്രവർത്തനതലം പങ്കുവെക്കുക മുതലായ കാര്യങ്ങൾ വഴി നിരന്തര സമ്പർക്കവും, യാത്രാച്ചെലവും കുറയ്ക്കാൻ സാധ്യമാകുന്നു.

A. ഇ-വാണിജ്യവും ഇ-വ്യാപാരവും (e-Commerce and e-Business)

ഇ-വാണിജ്യവും ഇ-വ്യാപാരവും പലപ്പോഴും ഇടകലർത്തി ഉപയോഗിക്കുന്ന പദങ്ങളാണെങ്കിലും ഇവ വ്യത്യാസപ്പെട്ടിരിക്കുന്നു. പണം കൈമാറ്റം ചെയ്യുന്ന വ്യാപാര ഇടപാടുകളാണ് ഇവാണിജ്യത്തിൽ ഉൾപ്പെടുന്നതെങ്കിൽ വ്യാപാരം നടത്തിക്കൊണ്ടുപോകുവാൻ ആവശ്യമായ എല്ലാ ഘടകങ്ങളും ഇ-വ്യാപരത്തിൽ ഉൾപ്പെടുന്നു. വിപണനം, അസംസ്കൃത വസ്തുക്കളുടെയും ചരക്കുകളുടെയും വാങ്ങൽ, ഉപഭോക്താക്കൾക്കുള്ള പരിശീലനം, വിതരണക്കാരെ കണ്ടെത്തൽ എന്നിവയെല്ലാം ഇ-വ്യാപാരത്തിൽ ഉൾപ്പെടുന്നു. അതിനാൽ ഇ-വാണിജ്യത്തിന്റെ വിപുലീകരണമാണ് ഇ-വ്യാപാരം.

B. ഇലക്ട്രോണിക് രൂപത്തിൽ പണം നൽകുന്ന സമ്പ്രദായം (Eletcronic Payment System (EPS))

കീശയിൽ പണമില്ലാതെയുള്ള ജീവിതം നിങ്ങൾക്കു ചിന്തിക്കുവാൻ കഴിയുമോ? പണം കൈകൊണ്ടു തൊടാതെ ഒരാഴ്ചയോളം തള്ളി നീക്കി എന്ന് ആരെങ്കിലും പറഞ്ഞാൽ നിങ്ങൾ വിശ്വസിക്കുമോ? ഇന്ന് നാം ജീവിക്കുന്ന ലോകത്തു ഒരു വിധം എല്ലാ കച്ചവട

സാധനങ്ങളിലും അതിന്റെ വില അനുബന്ധമായി ചേർത്തിരിക്കും (Price tag). അപ്പോൾ പിന്നെ കാശ് നൽകാതെ ഒരു സാധനം വാങ്ങുക എന്നത് എങ്ങനെ സാധ്യമാകും?

ഇലക്ട്രോണിക് രൂപത്തിൽ പണം നൽകുന്ന സമ്പ്രദായം Eletcronic Payment System (EPS) ഇതിനു പരിഹാരമാകുന്നു. നമ്മൾ ഇലക്ട്രോണിക് വ്യാപാരത്തിലേക്ക് കടന്നു ചെല്ലുമ്പോൾ പണമിടപാടുകളും ഇലക്ട്രോണിക് രൂപത്തിലാകേണ്ടതുണ്ട്. സമുചിതമായ രീതിയിൽ പ്രയോഗത്തിൽ വരുത്തിയാൽ ഇത് സൗകര്യപ്രദവും സുരക്ഷിതവുമായിരിക്കും. ഇ-വ്യാപാരത്തിൽ ഇ പി എസ് ഒരു പ്രധാന പങ്ക് വഹിക്കുന്നു. വളരെ വലിയ പ്രാധാന്യമുണ്ട്. ഓൺലൈൻ സാഹചര്യത്തിൽ ഉപഭോക്താവും വിൽപ്പനക്കാരനും തമ്മിൽ നടത്തുന്ന സാമ്പത്തിക ഇടപാടുകളെ ആണ് ഇലക്ട്രോണിക് രൂപത്തിലുള്ള പണമിടപാട് സമ്പ്രദായം (Eletcronic Payment System) എന്ന് പറയുന്നത്. ഒരു ബാങ്കിന്റെയോ ഇടനിലക്കാരന്റെയോ പിന്തുണയോടെയുള്ള ഡിജിറ്റൽ സാമ്പത്തിക ഉപകരണം ഉപയോഗപ്പെടുത്തിയാണ് ഇവിടെ സാമ്പത്തിക ഇടപാടുകൾ സുഗമമാക്കുന്നത് (അതായത് ക്രെഡിറ്റ്/ ഡെബിറ്റ് കാർഡ്, ഇലക്ട്രോണിക് ചെക്ക് അല്ലെങ്കിൽ ഡിജിറ്റൽ പണം)

C. ഇ – ബാങ്കിങ്

വിവരസാങ്കേതിക വിദ്യയുടെ പുരോഗതിക്കനുസരിച്ചു ബാങ്കിങ് മേഖലയിലും വലിയ തോതിലുള്ള മാറ്റങ്ങൾ സംഭവിച്ചിട്ടുണ്ട്. ബാങ്കിങ് മേഖലയിലെ നമ്മുടെ പരമ്പരാഗത സങ്കൽപ്പങ്ങൾ ഒരുപാട് മാറിക്കഴിഞ്ഞു. ഉദാഹരണമായി, ഇന്ന് സാമ്പത്തിക ഇടപാടുകൾക്ക് ബാങ്ക് സന്ദർശിക്കേണ്ട ആവശ്യം തന്നെ ഇല്ല. യാത്രകളിൽ പോലും ഓൺലൈൻ ബാങ്കിങ് സൗകര്യങ്ങൾ ഉപയോഗപ്പെടുത്തി നമുക്ക് ഇടപാടുകൾ നടത്താം. എ ടി എം, ഡെബിറ്റ് കാർഡ്, ക്രെഡിറ്റ് കാർഡ്, ഇന്റർനെറ്റ് ബാങ്കിങ്, കോർ ബാങ്കിങ് എന്നിങ്ങനെയുള്ള സൗകര്യങ്ങൾ പരമ്പരാഗത ബാങ്കിങ് സമ്പ്രദായത്തിൽ നിന്നും ഇ-ബാങ്കിങ് സമ്പ്രദായത്തിലേക്കു മാറുവാൻ സഹായിച്ചിട്ടുണ്ട്. ബാങ്കിങ് സേവനങ്ങൾ ഉപഭോക്താക്കൾക്ക് നേരിട്ട് ഇലക്ട്രോണിക് മാർഗ്ഗത്തിലൂടെ നൽകുന്നതിനെയാണ് ഇ-ബാങ്കിങ് അല്ലെങ്കിൽ ഇലക്ട്രോണിക് ബാങ്കിങ് എന്ന് പറയുന്നത്. ഇതിൽ ഭൂമിശാസ്ത്രപരമായ പരിമിതികൾ ഇല്ലാതെ ഡാറ്റ ഉപയോഗിക്കുവാൻ കഴിയും.

D. ഇ – വ്യാപാരത്തിന്റെ മേൻമകൾ

ഇ-വ്യാപാരം ഉപഭോക്താക്കൾക്കും സ്ഥാപനത്തിനും ധാരാളം മേന്മകൾ പ്രദാനം ചെയ്യുന്നു. ഇ-വ്യാപാരത്തിന്റെ ഉപയോഗം കൊണ്ട് ലഭ്യമാകുന്ന ചില പ്രധാന ഗുണങ്ങളെ പറ്റി നമുക്ക് ചർച്ച ചെയ്യാം.

- ***ഇത് ഭൂമിശാസ്ത്രപരമായ പരിമിതികൾ മറികടക്കുന്നു.*** സാധാരണ രീതിയിലുള്ള കച്ചവടത്തിൽ നിങ്ങൾക്ക് സേവനങ്ങൾ ലഭിക്കുന്നത് പരിമിതമായ സ്ഥലങ്ങളിൽ നിന്ന് മാത്രമായിരിക്കും. പക്ഷെ ഈ പരിമിതി ഇ-വാണിജ്യം വഴി പരിഹരിക്കപ്പെടുന്നു.
- ***ഇ-വ്യാപാരം പ്രവർത്തന ചെലവ് കുറയ്ക്കുന്നു.*** വ്യാപാരിക്ക് ഇ-വാണിജ്യത്തിൽ ഒരു പ്രധാന സ്ഥലത്തെ ഭൗതിക സാഹചര്യങ്ങൾ ആവശ്യമില്ല; അതിനാൽ പ്രവർത്തന ചെലവ് കുറയുന്നു. ഇതുവഴി ലഭ്യമാകുന്ന ലാഭത്തിന്റെ ഒരു വിഹിതം ഇളവ് (Discount) രൂപത്തിൽ ഉപഭോക്താവിന് തന്നെ കൊടുക്കുവാൻ സാധിക്കുന്നു.

- ***ഇത് യാത്രാ സമയവും ചെലവും കുറയ്ക്കുന്നു.*** ചിലപ്പോൾ ഉപഭോക്താക്കൾക്ക് അവരുടെ പ്രിയപ്പെട്ട വിപണനശാലകളിൽ എത്തിച്ചേരുവാൻ ദീർഘദൂരം യാത്ര ചെയ്യേണ്ടതായി വന്നേക്കാം. ഇ-വ്യാപാരം വഴി ഇതേ വിപണനശാലസാങ്കൽപ്പിക മായി സന്ദർശിക്കുവാൻ അനുവദിക്കുന്നു.

- ***ഇത് എല്ലായ്പ്പോഴും തുറന്നു പ്രവർത്തിക്കുന്നു.*** ഇ-വ്യാപാര സേവനം എല്ലായ്പ്പോഴും ലഭ്യമാണ് (24x7). വ്യാപാരിയെ സംബന്ധിച്ചിടത്തോളം, അവർക്ക് വിൽപന വർദ്ധിക്കുന്നു. ഉപഭോക്താവിനാണെങ്കിൽ "എല്ലായ്പ്പോഴും" തുറന്നു പ്രവർത്തിക്കുന്ന വിപണനശാലകൾ കൂടുതൽ സൗകര്യപ്രദമാണ്.

- ***ഒരുപാട് ഉൽപന്നങ്ങൾക്കിടയിൽ നിന്ന് നമുക്കാവശ്യമായ ഉൽപന്നം വളരെ പെട്ടെന്ന് കണ്ടെത്തുവാൻ കഴിയും.*** ഇവ്യാപാര വെബ്സൈറ്റ് വഴി ഉപഭോക്താക്കൾക്ക് വ്യത്യസ്ത വ്യാപാരികളുടെ അനേകം ഉൽപന്നങ്ങൾക്കിടയിൽ നിന്ന് ഇഷ്ടമുള്ളത് തെരഞ്ഞെടുക്കുവാൻ കഴിയുന്നു. ചില വെബ്സൈറ്റുകൾ ഉപഭോക്താവിന്റെ മുൻഗണനയും വാങ്ങിച്ച വസ്തുക്കളുടെ പട്ടികയും ഓർമിപ്പിക്കുന്നതിനാൽ ഉൽപന്നങ്ങൾ ആവർത്തിച്ചു വാങ്ങുന്നവർക്ക് ഇത് സൗകര്യപ്രദമാണ്. ഉൽപന്നങ്ങളുടെ സ്വഭാവം, വിലയുടെ താരതമ്യം മുതലായ സവിശേഷതകൾ ഇ-വ്യാപാരത്തിന്റെ മറ്റൊരു ആകർഷണമാണ്.

മൊബൈൽ ഉപകരണങ്ങളായ മൊബൈൽ ഫോൺ, പി ഡി എ (PDA) മുതലായവ ഉപയോഗിച്ച്കൊണ്ടുള്ള വ്യാപാര പ്രവർത്തനങ്ങളെയും, പ്രക്രിയകളെയും, ഉപയോഗങ്ങളെയും ആണ് എം വ്യാപാരം (m-Business) അല്ലെങ്കിൽ മൊബൈൽ വ്യാപാരം (mobile business) എന്ന പദം കൊണ്ട് അർത്ഥമാക്കുന്നത്. എം വ്യാപാരം ഇ-വ്യാപാരത്തിന്റെ ഒരു ശാഖയായി കണക്കാക്കാം.

ഇന്റർനെറ്റ് ഉപയോഗിച്ച് ലഘു വ്യാപാരം നടത്തുന്നതിനെയാണ് ഇടൈലിങ് e-tailing (അല്ലെങ്കിൽ ഇലക്ട്രോണിക് റീടൈലിങ് (eletcronic retailing) എന്ന് പറയുന്നത്. ഇത് B2C (ബിസിനസ് ടു കൺസ്യൂമർ) യുടെ സർവ്വ സാധാരണമായ ഒരു രൂപമാണ്.

E. ഇ-വ്യാപാരത്തിന്റെ വെല്ലുവിളികൾ (Chalanges to e-Business)

വ്യാപാരലോകത്ത് ഇ-വ്യാപാരത്തിന് അതിശക്തമായ സാധ്യതകൾ ഉണ്ടെങ്കിലും ഇന്ത്യ പോലുള്ള വികസ്വര രാഷ്ട്രങ്ങളിൽ ഇതിന് പല വിധ വെല്ലുവിളികളെയും അഭിമുഖീകരിക്കേണ്ടി വരുന്നു. പ്രധാനപ്പെട്ട വെല്ലുവിളികൾ താഴെ ചർച്ച ചെയ്യുന്നു.

- ജനസംഖ്യയുടെ നല്ലൊരു ശതമാനവും ഐ ടിയുടെ പ്രയോഗത്തെക്കുറിച്ചും ഉപയോഗത്തെ കുറിച്ചും ബോധവാന്മാരല്ല. സ്ഥിരമായി ഇന്റർനെറ്റ് ഉപയോഗിക്കുന്ന ആളുകൾക്ക് പോലും ഓൺലൈൻ വ്യാപാരത്തെ കുറിച്ചും അതിന്റെ സാധ്യതകളെ കുറിച്ചും അറിവില്ല എന്നത് അത്ഭുതകരമാണ്.

- ഇ-വ്യാപാരത്തിന് വളരെ അത്യാവശ്യമായ പ്ലാസ്റ്റിക് പണം എന്നറിയപ്പെടുന്ന ക്രെഡിറ്റ് കാർഡ്, ഡെബിറ്റ് കാർഡ് എന്നിവയും ഇന്റർനെറ്റ് ബാങ്കിങ് സംവിധാനവും ഭൂരിഭാഗം ആളുകളും പ്രത്യേകിച്ച് ഗ്രാമീണ മേഖലയിലുള്ളവർക്ക് പ്രാപ്യമല്ല.

- വേണ്ട മുൻകരുതലുകൾ ഇല്ലാതെ ഉപയോഗിക്കുന്നതുകൊണ്ട് ഉപഭോക്താക്കൾക്ക് അവരുടെ വിലപ്പെട്ട വിവരങ്ങളായ ക്രെഡിറ്റ്/ഡെബിറ്റ് കാർഡ് നമ്പർ, രഹസ്യകോഡ് (Password) എന്നിവ നഷ്ടപ്പെടാൻ സാധ്യതയുണ്ട്.

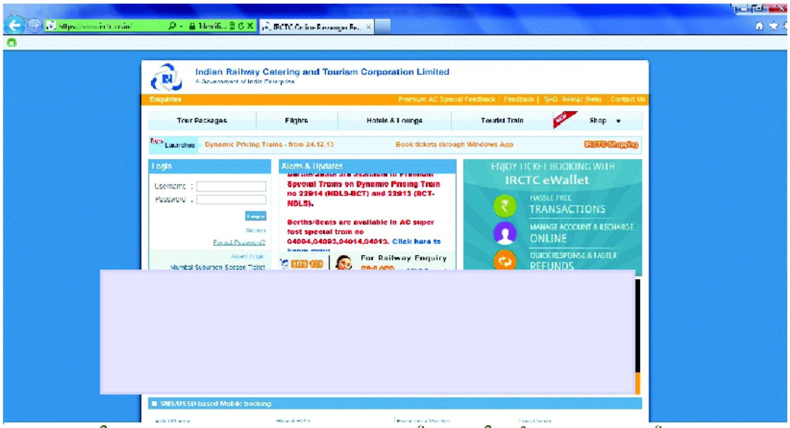
- വസ്ത്രങ്ങൾ, കരകൗശല വസ്തുക്കൾ, ആഭരണങ്ങൾ തുടങ്ങിയ ഉൽപന്നങ്ങൾ ഭൗതിക സാഹചര്യത്തിൽ പരിശോധിച്ച് വാങ്ങാവുന്നതാണ്. എന്നാൽ ഓൺലൈൻ വ്യാപാരത്തിൽ ഉപഭോക്താവിന് 'സ്പർശിച്ചും അനുഭവിച്ചറിഞ്ഞും' ഉൽപന്നങ്ങൾ വാങ്ങുവാൻ സാധ്യമല്ല.
- ഇ-വ്യാപാരത്തിന്റെ വിജയത്തിനായി വ്യാപാര സ്ഥാപനങ്ങൾ കുറ്റമറ്റതും, കാര്യക്ഷമവുമായ രീതിയിൽ ഉൽപന്നങ്ങൾ വിതരണം ചെയ്യേണ്ടത് അത്യാവശ്യമാണ്.

F. ഇ-വ്യാപാരത്തിന് പ്രയോജനകരമായ വെബ്സൈറ്റുകൾ (Useful e-Business websites)

ജനപ്രീതിയാർജ്ജിച്ച ചില ഇ-വ്യാപാര വെബ്സൈറ്റുകൾ പട്ടിക 12.2 ൽ കൊടുത്തിരിക്കുന്നു. കൂടാതെ ചില ഇ-വ്യാപാര വെബ്സൈറ്റുകളുടെ പ്രധാന താളുകൾ ചിത്രം 12.4 ലും 12.5 ലും കാണിച്ചിരിക്കുന്നു.

വെബ്സൈറ്റുകളും മേൽവിലാസവും	ഉപയോഗം / സേവനം
www.irctc.co.in	തീവണ്ടി, വിമാന ടിക്കറ്റുകൾ ബുക്ക് ചെയ്യുവാനും, റദ്ദ് ചെയ്യുവാനും, ഹോട്ടലുകളിൽ മുറികൾ ബുക്ക് ചെയ്യുവാനും വേണ്ടിയുള്ള ഇന്ത്യൻ റെയിൽവേ കാറ്ററിംഗ് ആൻഡ് ടൂറിസം ലിമിറ്റഡ് (IRCTC) ന്റെ ഈ വെബ്സൈറ്റ് ഉപയോഗിക്കാം.
www.amazon.com	വാഷിംഗ്ടൺ ആസ്ഥാനമായി പ്രവർത്തിക്കുന്ന അമേരിക്കൻ ഓൺലൈൻ വ്യാപാരി.
www.ebay.in	ഇന്ത്യയിലെ ഏറ്റവും വലിയ ഓൺലൈൻ വ്യാപാര വെബ്സൈറ്റുകളിൽ ഒന്ന്.
www.licindia.in	ഇൻഷുറൻസ് (Insurance) സ്ഥാപനത്തിന്റെ വെബ്സൈറ്റ്.
www.airindia.com	എയർ ഇന്ത്യ വിമാന ടിക്കറ്റുകൾ ബുക്ക് ചെയ്യുന്നതിനുള്ള ഓൺലൈൻ വെബ്സൈറ്റ്.
www.keralartc.com	കെഎസ്ആർടിസി ബസ് ടിക്കറ്റുകൾ ബുക്ക് ചെയ്യുന്നതിനുള്ള വെബ്സൈറ്റ്.
www.bookmyshow.com	സിനിമാശാലകളിലേക്കുള്ള ടിക്കറ്റ് ബുക്ക് ചെയ്യുന്നതിനുള്ള വെബ്സൈറ്റ്.

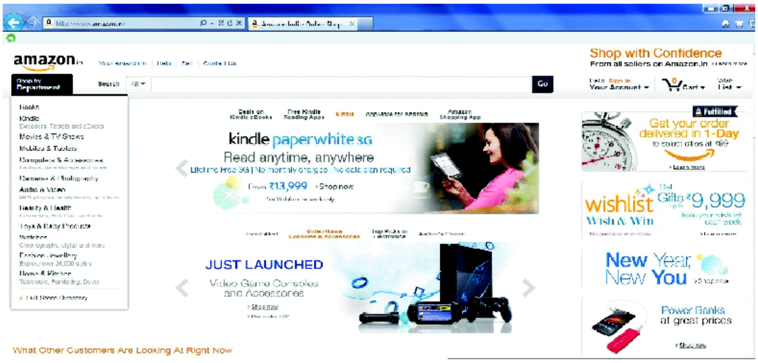
പട്ടിക 12.2: ചില ഇ-വ്യാപാര വെബ്സൈറ്റുകൾ



ചിത്രം 12.4: www.irctc.co.in വെബ്സൈറ്റിന്റെ പ്രധാന പേജ്

www.irctc.co.in

IRCTC ഒരു ഭാരത സർക്കാർ സംരംഭമാണ്. ഭക്ഷണം തയ്യാറാക്കി വിതരണം ചെയ്യുക (Catering), വിനോദസഞ്ചാരം (Tourism), തീവണ്ടി, വിമാന ടിക്കറ്റ് ബന്ധപ്പെട്ട പ്രവർത്തനങ്ങൾ ഇത് കൈകാര്യം ചെയ്യുന്നു. ഓൺലൈൻ വഴി ടിക്കറ്റ് ബുക്ക് ചെയ്യാനുള്ള സൗകര്യം ഒരുക്കുന്നതോടൊപ്പം റിസർവേഷൻ നില, തീവണ്ടി സമയം, ഹോട്ടൽ ബുക്കിങ്ങ് എന്നിവയ്ക്കും ഈ വെബ്സൈറ്റ് ഉപയോഗിക്കാം.



ചിത്രം 12.5: www.Amazon.in വെബ്സൈറ്റിന്റെ പ്രധാന പേജ്

www.amazon.com

ആമസോൺ ലോകത്തിലെ ഏറ്റവും വലിയ ചില്ലറ വ്യാപാരിയാണ്. ആമസോൺ തുടക്കത്തിൽ ഒരു ഓൺലൈൻ പുസ്തകശേഖരമായി ആരംഭിച്ചു. ഇത് പിന്നീട് സോഫ്റ്റ്‌വെയർ, വീഡിയോ ഗെയിംസ്, ഇലക്ട്രോണിക് ഉപകരണങ്ങൾ, ഗൃഹോപകരണങ്ങൾ, ഭക്ഷണ സാധനങ്ങൾ, കളിപ്പാട്ടങ്ങൾ, ആഭരണങ്ങൾ എന്നിവയുടെ വില്പനക്കാരായി മാറി. ഈ കമ്പനി ഉപഭോക്തൃ ഇലക്ട്രോണിക്സും (consumer eletcronics) നിർമ്മിക്കുന്നു. വിവിധ രാജ്യങ്ങളിൽ വ്യാപാരം നടത്തുന്നതിനായി ഇവർക്ക് വ്യത്യസ്ത വ്യാപാര വെബ്സൈറ്റുകൾ നിലവിലുണ്ട്. 2013 ൽ www.amazon.in എന്ന വെബ്സൈറ്റ് ഇന്ത്യയിൽ തുടക്കം കുറിച്ചു.

സ്വയം വിലയിരുത്താം

1. ഓൺലൈൻ ചുറ്റുപാടിൽ വില്പനക്കാരനും വാങ്ങുന്നവനും തമ്മിൽ നടത്തുന്ന സാമ്പത്തിക ഇടപാട് സംവിധാനത്തെ .... എന്ന് അറിയപ്പെടുന്നു.
2. ഇ-വ്യാപാരം നിർവ്വചിക്കുക.
3. ഇ-ബാങ്കിങ് നിർവ്വചിക്കുക.
4. താഴെ കൊടുത്തിരിക്കുന്ന പ്രസ്താവന ശരിയോ തെറ്റോ എന്ന് പരിശോധിക്കുക.

   'ഇ-വ്യാപാരം എന്നത് ഇ-വാണിജ്യത്തിന്റെ വിപുലീകരണമാണ്.

12.1. 3 ഇ-പഠനം (e-Learning)

അധ്യാപകർ, വിദ്യാർത്ഥികൾ, പഠന ഉപകരണങ്ങൾ മുതലായവ അടങ്ങുന്ന ഒരു ക്ലാസ് മുറി നമുക്കെല്ലാവർക്കും സുപരിചിതമാണ്. എന്നാൽ ഒരു സാങ്കൽപ്പിക ക്ലാസ് മുറിയെ

പറ്റി ചിന്തിച്ചുനോക്കൂ, അവിടെ അധ്യാപകരും വിദ്യാർത്ഥികളും വ്യത്യസ്ഥമായ സ്ഥലങ്ങളിലോ ഒരു പക്ഷേ പല രാജ്യങ്ങളിലോ ആയിരിക്കും ഉണ്ടാകുക. ഇത് സാധ്യമാകുമോ? അതെ അത് സാധ്യമാണ്. ഇ-പഠനം വഴി സാമ്പ്രദായിക പഠന രീതിയിലുള്ള പല പരിമിതികളെയും മറികടക്കുവാൻ കഴിയുന്നു. ഇലക്ട്രോണിക് മാധ്യമങ്ങളെയും, ഐ സി ടിയെയും ഉൾക്കൊള്ളിച്ചുകൊണ്ടുള്ള പഠനരീതിയെയാണ് ഇ-പഠനം എന്ന് വിശേഷിപ്പിക്കുന്നത്.

A. ഇ – പഠനത്തിന് ഉപയോഗിക്കുന്ന ഉപകരണങ്ങൾ (e-Learning tools)

ഇ-പഠനം മെച്ചപ്പെടുത്തുന്നതിനായി നിരവധി ഉപകരണങ്ങൾ ഉപയോഗിക്കുന്നുണ്ട്. ഇ-പഠന ഉപകരണങ്ങൾ താഴെ ചർച്ച ചെയ്യുന്നു.

i. ഇലക്ട്രോണിക് ബുക്ക് റീഡർ (e-Book Reader)

ഒരു വലിയ ബാഗും അതിൽ നിറയെ പുസ്തകങ്ങളുമായി സ്കൂളിൽ പോകുന്നതിന് പകരം നമുക്ക് ആവശ്യമായ എല്ലാ വിവരങ്ങളും ഒരു ചെറിയ ഉപകരണത്തിൽ സൂക്ഷിച്ച് സ്കൂളിൽ പോകുന്നത് ചിന്തിച്ചു നോക്കൂ? അത്! രസകരമാണ് അല്ലേ? ഒരുപാട് പുസ്തകങ്ങളിലായി അടങ്ങിയിട്ടുള്ള ആശയങ്ങളെ ഡിജിറ്റൽ രൂപത്തിലാക്കി സൂക്ഷിച്ചുവെക്കുവാനും കയ്യിൽ കൊണ്ട് നടക്കുവാനും കഴിയുന്ന ഉപകരണമാണ് ഇ-ബുക്ക് റീഡർ. ഇ-ബുക്ക് റീഡറിൽ ഏത് പുസ്തകവും അതിന്റെ ഏതു പേജും അനായാസേന നമുക്ക് തുറക്കാം. ഈ ഉപകരണം ഉപയോഗിച്ച് ഇന്റർനെറ്റിൽ നിന്ന് വെബ് പ്രമാണങ്ങൾ ഡൗൺലോഡ് ചെയ്ത് എപ്പോൾ വേണമെങ്കിലും വായിക്കുവാനും ഓൺലൈനായി പണമടയ്ക്കുവാനും സാധിക്കുന്നു. ഇ-ബുക്ക് റീഡറിലെ ഓഡിയോ ഔട്ട്പുട്ട് സംവിധാനം ഉപയോഗപ്പെടുത്തി പാഠഭാഗങ്ങൾ വായിച്ച് കേൾക്കുവാനും കഴിയുന്നു.

ii. ഇ – ടെക്സ്റ്റ്/e-Text (ഇ – പുസ്തകം)

പുസ്തകത്തിലുള്ള വിവരങ്ങൾ ഇലക്ട്രോണിക് രൂപത്തിൽ ലഭ്യമാക്കുന്നതിനെയാണ് ഇ-ടെക്സ്റ്റ് (ഇ-പുസ്തകം) എന്നു പറയുന്നത്. ഇ-ബുക്ക് റീഡറോ, കമ്പ്യൂട്ടറോ, മറ്റേതെങ്കിലും ഇലക്ട്രോണിക് ഉപകരണമോ ഉപയോഗിച്ച് ഇ-ടെക്സ്റ്റ് വായിക്കുവാനാകും. നമുക്ക് താൽപര്യമുള്ള രീതിയിൽ ടെക്സ്റ്റിന്റെ രൂപഘടനയിൽ സോഫ്റ്റ്വെയറുകൾ ഉപയോഗിച്ച് മാറ്റം വരുത്താവുന്നതാണ്. ഒരു കമ്പ്യൂട്ടറിന്റെയോ ഇ-ടെക്സ്റ്റ് റീഡറിന്റെയോ സഹായത്തോടെ ഇ-ടെക്സ്റ്റ് വായിച്ച് കേൾക്കുവാനാകും. ഇത് കാഴ്ച ശക്തി കുറഞ്ഞവർക്ക് വളരെയധികം സഹായകരമാണ്.

iii. ഓൺലൈൻ ചാറ്റ് (Online chat)

ഇന്റർനെറ്റ് ഉപയോഗിച്ച് രണ്ടോ അതിലധികമോ വ്യക്തികൾ തമ്മിൽ ടെക്സ്റ്റ്, രൂപത്തിൽ സന്ദേശങ്ങൾ കൈമാറുന്ന രീതിയാണ് ഓൺലൈൻ ചാറ്റ്. സാങ്കൽപ്പിക ക്ലാസ് മുറികളിൽ അധ്യാപകരും വിദ്യാർത്ഥികളും തമ്മിൽ വിവിധ വിഷയങ്ങളെകുറിച്ച് ചർച്ച നടത്തുന്നത് ഓൺലൈൻ ചാറ്റിങ്ങിലൂടെയാണ്. ഇന്റർനെറ്റിന്റെ വേഗത കുറവാണെങ്കിലും ടെക്സ്റ്റ് രൂപത്തിലുള്ള ഓൺലൈൻ ചാറ്റിങ്ങ് നടത്താവുന്നതാണ്. പരസ്പരം കണ്ട് സംസാരിക്കുന്നതിനും ഓൺലൈൻ ചാറ്റിങ്ങിൽ സാധ്യമാണ്. ഇതിന് വേഗതയേറിയ ഇന്റർനെറ്റ് ബന്ധം, വെബ് ക്യാമറ, സ്പീക്കർ തുടങ്ങിയ ഉപകരണങ്ങളും ആവശ്യമാണ്. ഓൺലൈൻ ചാറ്റ് സാമൂഹിക ഇടപെടലുകൾക്കുള്ള സാധ്യതയും ഒരുക്കുന്നു. രണ്ട് വ്യക്തികൾ തമ്മിൽ വിവിധ സ്ഥലങ്ങളിൽ ഇരുന്ന് ആശയവിനിമയം നടത്തുന്നതിന് ഇത് സഹായിക്കുന്നു. ഈ സംവിധാനം ഉപയോഗിച്ചാണ് അധ്യാപകരുടെ ഓൺലൈനിലൂടെ

യുള്ള തത്സമയക്ലാസുകൾ ലഭ്യമാകുന്നത്. വിദ്യാർത്ഥികൾക്ക് വിദൂര സ്ഥലങ്ങളിൽ ഇരുന്ന് ലോഗിൻ ചെയ്ത് ക്ലാസുകളിൽ പങ്കെടുക്കാം.

iv. ഇ-കണ്ടന്റ് (e-Content)

വീഡിയോകൾ, അവതരണങ്ങൾ, ഗ്രാഫിക്സ്, ആനിമേഷൻ എന്നിവ ഉൾപ്പെടുത്തി ഇ-കണ്ടന്റ് നിർമ്മിക്കുകയും അതുവഴി ഇ-പഠനം സാധ്യമാക്കുകയും ചെയ്യാം. ഇവ ടെലിവിഷനിലൂടെയോ, വെബ്സൈറ്റുകൾ മുഖേനയോ പ്രക്ഷേപണം ചെയ്യുവാനും സാധിക്കുന്നു. അപ്പ്ലോഡ് ചെയ്ത ഉള്ളടക്കങ്ങൾ പിന്നീട് വെബ്സൈറ്റുകളിൽ നിന്ന് ഡൗൺലോഡ് ചെയ്യുവാനും എപ്പോൾ വേണമെങ്കിലും ഉപയോഗിക്കുവാനും സാധിക്കുന്നു. ഉള്ളടക്കങ്ങൾ എളുപ്പത്തിൽ ജനങ്ങളിലേക്ക് എത്തിക്കുവാൻ മൾട്ടി മീഡിയ ഉള്ളടക്കങ്ങൾ വഴി സാദ്ധ്യമാക്കുന്നു. സാമ്പ്രദായിക ക്ലാസ് മുറികളിൽ അസാധ്യമായത് കൃത്യമായി നിർമിച്ച് കാണിക്കുവാനും ഇത് വഴി സാധ്യമാകുന്നു. കേരള സർക്കാരിന്റെ സമഗ്ര പോർട്ടൽ ഇ-കണ്ടന്റിന് ഒരു ഉദാഹരണമാണ്. (www. samagra. kite. kerala.gov.in)

v. വിദ്യാഭ്യാസ ടി വി ചാനലുകൾ (Educational TV Channels)

ഇ-പഠനം സാധ്യമാക്കുന്നതിന് നിരവധി ടി വി ചാനലുകളും വെബ്ചാനലുകളും ലഭ്യമാണ്. ഈ ചാനലുകൾ ഉപയോഗിച്ച് വിവിധ വിഷയങ്ങളിലെ ക്ലാസുകൾ, വിദഗ്ധരുമായി നടത്തിയ അഭിമുഖങ്ങൾ, ലാബുകളിൽ നടത്തിയ പരീക്ഷണങ്ങൾ എന്നിവ സംപ്രേഷണം ചെയ്യുവാൻ സാധിക്കുന്നു. ഇവയിൽ ചില ചാനലുകൾ ഇന്റർനെറ്റിലും ലഭ്യമാണ്. ദൂരദർശന്റെ വ്യാസ്, കേരളാ ഗവൺമെന്റിന്റെ വിക്ടേഴ്സ് തുടങ്ങിയവ വിദ്യാഭ്യാസ ടെലിവിഷൻ ചാനലുകൾക്ക് ഉദാഹരണങ്ങളാണ്.

B. ഇ-പഠനത്തിന്റെ മേൻമകൾ (Advantages of e-Learning)

ഇ-പഠനത്തിന് ധാരാളം മേൻമകളുണ്ട്. അവ താഴെ കൊടുത്തിരിക്കുന്നു.

- വിദൂര സ്ഥലങ്ങളിലുള്ള വിദ്യാർത്ഥികൾക്ക് വൈവിധ്യമാർന്നതും ഗുണമേൻമയുള്ളതുമായ വിവിധ വിഷയങ്ങളിലുള്ള കോഴ്സുകൾ ഇ-പഠനം വഴി പഠിക്കുവാൻ കഴിയുന്നു.
- ഇ-പഠനത്തിന് ചെലവ് കുറവാണ്. ഇത് വഴി യാത്രാസമയവും, അധ്യാപക വേതനവും ലാഭിക്കാം.
- പരിമിതമായ സാമ്പത്തിക സ്ഥിതിയുള്ളവർക്ക് പോലും കുറഞ്ഞ ചിലവുള്ള ഇ-പഠനം സാധ്യമാണ്.
- ഈ സംവിധാനം ഉപയോഗിച്ച് തദ്ദേശീയവും, അന്തർദേശീയവുമായ സ്ഥാപനങ്ങളിൽ നിന്നും ഓൺലൈൻ കോഴ്സുകൾ പഠിക്കുന്നതിന് അവസരം ലഭിക്കുന്നു.
- സ്ഥലവും, സമയവും ഇ-പഠനത്തിന് തടസമാകുന്നില്ല.

C. ഇ-പഠനത്തിന്റെ വെല്ലുവിളികൾ (Challanges to e-Learning)

ഇ-പഠനം നേരിടുന്ന ചില വെല്ലുവിളികൾ താഴെ പറയുന്നവയാണ്.

- അധ്യാപകരും വിദ്യാർത്ഥികളും തമ്മിൽ മുഖാമുഖമുള്ള പഠനം സാധ്യമല്ല.
- അടിസ്ഥാന സൗകര്യങ്ങളുടെ പരിമിതി കാരണം അധ്യാപകരും വിദ്യാർത്ഥികളും തമ്മിലുള്ള ഇടപെടലുകൾ പരിമിതപ്പെടുത്തുന്നു.
- ഉപകരണങ്ങളുടെയും സാങ്കേതികവിദ്യയുടെയും (കമ്പ്യൂട്ടറും വേഗതയേറിയ ഇന്റർനെറ്റും) ആവശ്യകത ഇ-പഠനത്തിന് തടസമാകുന്നു.

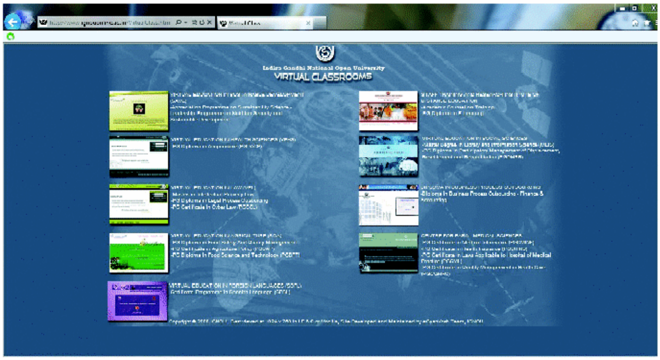
- നിരന്തരമായ പ്രോത്സാഹനം ലഭിക്കേണ്ട വിദ്യാർത്ഥികൾക്ക് അത് വേണ്ട രീതിയിൽ ലഭിക്കാതെ പോകുന്നു.
- ലബോറട്ടറികളുടെ അഭാവം കാരണം പ്രായോഗിക പഠനത്തെ പരിമിതപ്പെടുത്തുന്നു.

D. ഇ-പഠനത്തിന് പ്രയോജനകരമായ വെബ് സൈറ്റുകൾ (Useful e-Learning Website)

ഇ-പഠനത്തിന് ഉപകരിക്കുന്ന വെബ്സൈറ്റുകളുടെ ഒരു പട്ടിക 10.3 ൽ സംഗ്രഹിച്ചിരിക്കുന്ന ചിത്രം 10.6 ൽ ഇഗ്നോയുടെ (ഇന്ദിരാഗാന്ധി നാഷണൽ ഓപ്പൺ യൂണിവേഴ്സിറ്റി) പ്രധാന പേജാണ് കാണിച്ചിരിക്കുന്നത്.

വെബ്സൈറ്റുകളും മേൽവിലാസവും	ഉപയോഗം / സേവനം
www.ignouonline.ac.in	വിവിധ കോഴ്സുകളിൽ വിദൂര വിദ്യാഭ്യാസം നൽകുന്ന ഓപ്പൺ യൂണിവേഴ്സിറ്റികളിൽ ഒന്നായ ഇന്ദിരാഗാന്ധി ഓപ്പൺ യൂണിവേഴ്സിറ്റിയുടെ ഔദ്യോഗിക വെബ്സൈറ്റാണിത്.
www.nptel.iitm.ac.in	എഞ്ചിനിയറിംഗ്, സയൻസ്, ഹ്യുമാനിറ്റീസ് വിഷയങ്ങളുടെ ഇ-കണ്ടെന്റുകളും വീഡിയോകളും തയ്യാറാക്കി ഓൺലൈനായി കോഴ്സുകൾ നൽകുന്ന നാഷണൽ പ്രോഗ്രാം ഓൺ ടെക്നോളജി എൻഹാൻസ്ഡ് ലേണിങ്ങിന്റെ വെബ്സൈറ്റാണിത്.
www.ncert.nic.in	നാഷണൽ കൗൺസിൽ ഓഫ് എഡ്യുക്കേഷണൽ റിസേർച്ച് ആൻഡ് ട്രെയിനിങ്ങിന്റെ വെബ്സൈറ്റാണിത്. വിവിധ പഠനസഹായികളും, സംശയനിവാരണ സാമഗ്രികളും ഹിന്ദി, ഇംഗ്ലീഷ് എന്നീ ഭാഷകളിൽ ഈ വെബ്സൈറ്റിൽ ഉൾപ്പെടുത്തിയിരിക്കുന്നു.
www.spoken_tutorial.org	മുംബൈ ഐ.ഐ.ടിയുടെ ഒരു സംരഭമാണിത്. ഇവ സ്പോക്കൺ ട്യൂട്ടോറിയൽസ് വഴി സോഫ്റ്റ്വെയർ പരിശീലനം സാധ്യമാക്കുന്നു. എല്ലാ കോഴ്സുകളും കേന്ദ്ര സർക്കാരിന്റെ പൂർണ്ണസഹായത്തോടുകൂടി സൗജന്യമായി നടത്തപ്പെടുന്നു.
www.w3schools.com	വെബ്സൈറ്റുകൾ നിർമ്മിക്കുന്നതിനാവശ്യമായ HTML, CSS, JavaScript, PHP, SQL, JQuery തുടങ്ങിയവ പഠിക്കുന്നതിന് ആവശ്യമായ പഠനസഹായികളും സംശയനിവാരണ സാമഗ്രികളും w3schools ൽ ലഭ്യമാണ്.
www.gcflearnfree.org	ഗുഡ്‌വിൽ കമ്മ്യൂണിറ്റി ഫൗണ്ടേഷൻ സഹായത്തോടുകൂടി പ്രവർത്തിക്കുന്ന ഒരു വെബ്സൈറ്റാണ് gcflearnfree.org. ഗണിതം, ശാസ്ത്രം, സാങ്കേതികവിദ്യ, ഇംഗ്ലീഷ് മുതലായ വിഷയങ്ങളിലെ പാഠഭാഗങ്ങൾ വളരെ ലളിതമായി ഇതിൽ കൊടുത്തിരിക്കുന്നു. ആർക്കും എവിടെ വൈദഗ്ധ്യവും അറിവുനേടുന്നതിന് ഇത് സഹായിക്കുന്നു.

പട്ടിക 12.3 ചില ഇ-പഠന വെബ്സൈറ്റുകൾ



ചിത്രം. 12.6 IGNOU വെബ്സൈറ്റിന്റെ പ്രധാനപേജ്.

www.ignouonline.ac.in

ഭാരത സർക്കാരിന്റെ കീഴിൽ പ്രവർത്തിക്കുന്ന ഒരു ഓപ്പൺ സർവ്വകലാശാലയാണ് ഇഗ്നോ (ഇന്ദിരാഗാന്ധി നാഷണൽ ഓപ്പൺ യൂണിവേഴ്സിറ്റി). ഈ വെബ്സൈറ്റിലൂടെ **eGyankosh** പോലുള്ള മികച്ച സേവനങ്ങൾ നൽകിവരുന്നു. പഠന കുറിപ്പുകളുടെയും വീഡിയോകളുടെയും ഒരു ഡിജിറ്റൽ കലവറയാണ് eGyankosh. ഗ്യാൻദർശൻ (Gyandarshan), ഗ്യാൻവാണി (Gyanvani), എഡ്യുസാറ്റ് (EDUSAT) മുതലായ വിദ്യാഭ്യാസ ചാനലുകളും ഈ വെബ്സൈറ്റ് ഉപയോഗിച്ച് സംപ്രേഷണം ചെയ്യുന്നു. ഇത് കൂടാതെ സർവ്വകലാശാല നടത്തിവരുന്ന എല്ലാ ഓൺലൈൻ പ്രോഗ്രാമുകളും സാങ്കൽപ്പിക ക്ലാസ് മുറികളിലൂടെ ലഭ്യമാണ്.

ചിത്രം. 12.7 NPTEL വെബ്സൈറ്റിന്റെ പ്രധാന പേജ്

www.nptel.iitm.ac.in

നാഷണൽ പ്രോഗ്രാം ഓൺ ടെക്നോളജി എൻഹാൻസ്ഡ് ലേണിങ് എന്നതിന്റെ ചുരുക്കപേരാണ് NPTEL. ഇത് ഏഴ് ഐ.ഐ.ടികളുടെയും ഐ.ഐ.എസ്.സിയുടെയും സംയുക്ത സംരംഭമാണ്. എഞ്ചിനിയറിങ്, സയൻസ് വിഷയങ്ങൾക്ക് ആവശ്യമായ പഠന സാമഗ്രികൾ നിർമ്മിക്കുക എന്നതാണ് ഇതിന്റെ ലക്ഷ്യം. ഇത്തരം വെബ്സൈറ്റുകൾ വെബ് അധിഷ്ഠിത പഠന സാമഗ്രികൾ നിർമ്മിക്കുകയും അവ സൗജന്യമായി ലഭ്യമാക്കുകയും ചെയ്യുന്നു. വീഡിയോ ഉൾപ്പെടുത്തികൊണ്ടുള്ള അധ്യാപനവും ഇത് വഴി സാധ്യമാണ്.

സ്വയം വിലയിരുത്താം

1. ഇന്റർനെറ്റ് ഉപയോഗിച്ച് രണ്ടോ അതിലധികമോ വ്യക്തികൾ തമ്മിൽ ടെക്സ്റ്റ് രൂപത്തിലുള്ള സന്ദേശങ്ങൾ കൈമാറുന്ന രീതിയാണ്................
2. വേറിട്ടു നിൽക്കുന്നത് ഏത്?
   a. ഇ-ബുക്ക് റീഡർ b. ഇ-ടെക്സ്റ്റ്
   c. ടെലിവിഷൻ ചാനൽ d. ഇ-ബിസിനസ്
3. ഇ-ടെക്സ്റ്റ് നിർവ്വചിക്കുക.
4. ഇ-ലേണിങ് ഉപകരണത്തിന് ഒരു ഉദാഹരണം നൽകുക?

12.2 വിവരങ്ങളുടെ സുരക്ഷിതത്വം (Information security)

ഇന്നത്തെ കാലത്ത് ആശയവിനിമയം, സാധനങ്ങൾ വാങ്ങിക്കൽ, ബാങ്കിങ് എന്നിങ്ങനെയുള്ള എല്ലാവിധ പ്രവർത്തനങ്ങളും ഇന്റർനെറ്റിലൂടെ നിർവഹിക്കപ്പെടുന്നു. ഇങ്ങനെയുള്ള പ്രവർത്തനങ്ങൾ നിർവഹിക്കപ്പെടുമ്പോൾ കമ്പ്യൂട്ടറുകൾ തമ്മിൽ വിവരങ്ങൾ കൈമാറ്റം ചെയ്യപ്പെടുന്നു. കമ്പ്യൂട്ടറുകൾ തമ്മിൽ കൈമാറ്റം ചെയ്യപ്പെടുന്ന വിവരങ്ങളുടെ സുരക്ഷിതത്വം ആശങ്ക ഉണ്ടാക്കുന്നതാണ്. ഈ ഭാഗത്ത് സൈബർ സുരക്ഷ, പകർപ്പവകാശം, വ്യാപാരമുദ്ര ലംഘനങ്ങൾ, സൈബർ കുറ്റകൃത്യങ്ങൾ എന്നിങ്ങനെ ഇന്റർനെറ്റുപയോഗിക്കുമ്പോൾ ശ്രദ്ധിക്കേണ്ട കാര്യങ്ങളെക്കുറിച്ച് വിശദമായി ചർച്ച ചെയ്യുന്നു.

12.2.1 ബൗദ്ധിക സ്വത്തവകാശം (Intellectual Property Right)

സംഗീതം, സാഹിത്യം, ചിത്രരചന, നൂതനമായ കണ്ടുപിടുത്തങ്ങൾ, ഡിസൈനിംഗ്, സോഫ്റ്റ്‌വെയർ വികസനം എന്നീ മേഖലകളിൽ ധാരാളം പേർ സൃഷ്ടിപരമായ പ്രവർത്തനങ്ങൾ നിർവഹിച്ചു പോരുന്നു. ധാരാളം സമയവും കഠിനാധ്വാനവും ഇവർ അതിനായി ഉപയോഗിക്കുന്നു. ഇങ്ങിനെ ഉണ്ടാകുന്ന അറിവുകളെയും കണ്ടെത്തലുകളെയുമാണ് ബൗദ്ധിക സ്വത്ത് എന്ന് പറയുന്നത്. അനുവാദമില്ലാതെ ഒരാളുടെ ആശയങ്ങൾ മറ്റൊരാൾ എടുക്കുന്നത് നീതിക്കു നിരക്കാത്തതാണ്. ഇങ്ങനെയുള്ള ബൗദ്ധിക അറിവുകൾ വികസിപ്പിക്കുന്നവർക്ക് അതിന്റെ ആനുകൂല്യം ലഭിക്കണം. അതുകൊണ്ട് ഇവ സംരക്ഷിക്കേണ്ടതുണ്ട്. 1883ലെ പാരീസ് കൺവെൻഷനിലാണ് ബൗദ്ധിക സ്വത്തവകാശത്തിന്റെ പ്രാധാന്യം തിരിച്ചറിഞ്ഞത്.

1886ൽ ഉണ്ടായ ബേൺ (BERNE) കൺവെൻഷനിലും അനുബന്ധമായ നിയമ നിർമ്മാണം നടത്തുകയുണ്ടായി. രണ്ട് ഉടമ്പടികളുടെയും കാര്യനിർവഹണം നടത്തുന്നത് വേൾഡ് ഇന്റലെക്ചൽ പ്രോപ്പർട്ടി ഓർഗനൈസേഷൻ (WIPO) ആണ്. WIPO, 1960 ൽ യുനൈറ്റഡ് നേഷന്റെ (UN) ആഭിമുഖ്യത്തിൽ സ്ഥാപിക്കപ്പെട്ടു. ലോകത്താകമാനമുള്ള ബൗദ്ധിക സ്വത്തുക്കളെയും അതിന്റെ അവകാശികളെയും കണ്ടെത്തി സംരക്ഷിക്കുക എന്നതാണ് WIPO യുടെ ദൗത്യം. WIPOയുടെ അടയാളചിഹ്നം ചിത്രം 10.5 ൽ കൊടുത്തിരിക്കുന്നു.

ബൗദ്ധിക സ്വത്തവകാശം എന്നത് ഭൂമിക്ക് മേലുള്ള അവകാശം, വീടിന് മേലുള്ള അവകാശം എന്നിങ്ങിനെ മറ്റേതൊരു അവകാശവുംപോലെ തന്നെയാണ്. ഒരു പ്രത്യേക കാലയളവിലേക്ക് ഒരാൾക്ക് അയാളുടെ കണ്ടെത്തലിന്റെ മുകളിൽ പൂർണമായ അവകാശം നൽകുന്നു. ഒരാൾക്ക് തന്റെ കണ്ടുപിടുത്തത്തിലൂടെയോ സൃഷ്ടിയിലൂടെയോ സാമ്പത്തിക നേട്ടം ഉണ്ടാകുന്നതിനും, അംഗീകാരം ലഭിക്കുന്നതിനും IPR ആളുകളെ പ്രാപ്തരാക്കുന്നു. IPR ഉടമകൾക്ക് തങ്ങളുടെ സൃഷ്ടികൾ പണത്തിനു പകരമായി നൽകാനും സാധ്യമാക്കുന്നു. കമ്പനി അവർക്ക് ലഭിക്കുന്ന അവകാശം വിപണിയിലെത്തിക്കുകയും

ചെയ്യുകയും ഈ നവ സൃഷ്ടിയെ സമൂഹത്തിൽ വിൽക്കുകയും ചെയ്യുന്നു. ഇത്തരത്തിൽ ബൗദ്ധിക സ്വത്തവകാശിയും കമ്പനിയും സമൂഹവും സൃഷ്ടിയിൽ നിന്ന് ആനുകൂല്യം കൈപ്പറ്റുന്നു. യുണൈറ്റഡ് നേഷൻസും (UN) മിക്കവാറും എല്ലാ രാജ്യങ്ങളും നവസൃഷ്ടികളെ പ്രോത്സാഹിപ്പിക്കുന്നതിനായി ബൗദ്ധിക സ്വത്തവകാശത്തെയും പ്രോത്സാഹിപ്പിക്കുന്നു.

ചിത്രം. 12.8 : WIPO യുടെ ലോഗോ

ഓരോ രാജ്യത്തിനും അതിന്റേതായ രീതിയിലുള്ള ബൗദ്ധിക സ്വത്തവകാശ രജിസ്ട്രേഷൻ സംവിധാനങ്ങളുണ്ട്. വ്യാപാരമുദ്ര, വ്യാവസായികരൂപരേഖ മുതലായവക്ക് WIPO ഒരു അന്താരാഷ്ട്ര രജിസ്ട്രേഷൻ സംവിധാനമായി പ്രവർത്തിക്കുന്നു. WIPO യുടെ നിയമം എല്ലാ അംഗ രാജ്യങ്ങൾക്കും ബാധകമാണ്. ബൗദ്ധിക സ്വത്തവകാശത്തെ രണ്ടായി തരംതിരിക്കുന്നു- വ്യവസായിക സ്വത്തവകാശം, പകർപ്പവകാശം.

A) വ്യാവസായിക സ്വത്ത് (Industrial property)

ചിത്രം. 12.9 : IP India യുടെ ലോഗോ

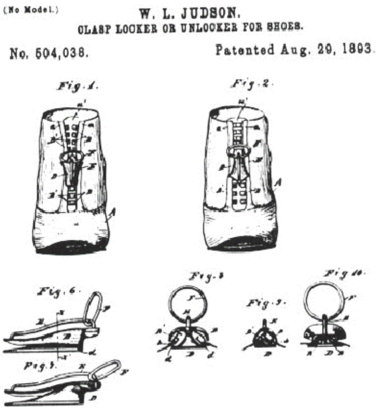
വ്യവസായം, വാണിജ്യം, കാർഷികോൽപ്പന്നങ്ങൾ എന്നിവയെ സംബന്ധിക്കുന്നതാണ് വ്യാവസായിക സ്വത്ത്. കണ്ടെത്തലുകൾ, വ്യാപാരമുദ്രകൾ, വ്യവസായരൂപരേഖകൾ, ഭൗമശാസ്ത്രപരമായ സൂചകങ്ങൾ എന്നിവ ഈ നിയമ പ്രകാരം സംരക്ഷിക്കപ്പെടുന്നു. ഇന്ത്യയിൽ ബൗദ്ധികസ്വത്തവകാശം കേന്ദ്ര വ്യവസായ വകുപ്പിന്റെ കീഴിലുള്ള കൺട്രോളർ ജനറൽ ഓഫ് പേറ്റന്റ് ഡിസൈൻസ് ആൻഡ് ട്രേഡ്മാർക്സിൽ രജിസ്റ്റർ ചെയ്യാവുന്നതാണ്. ബൗദ്ധിക സ്വത്തവകാശത്തിന്റെ അടയാള ചിഹ്നം ചിത്രം 10.6 ൽ കാണിച്ചിരിക്കുന്നു.

കുത്തകാവകാശം (Patents): ഒരു കണ്ടുപിടുത്തത്തിന്റെ മേൽ അത് കണ്ടുപിടിക്കുന്ന വ്യക്തിക്ക് നൽകുന്ന പരിപൂർണമായ അവകാശമാണ് കുത്തകാവകാശം. കണ്ടുപിടുത്തമെന്നാൽ ഒരു പുതിയ ഉൽപ്പന്നമോ, ഒരു കണ്ടെത്തലിനു സഹായിക്കുന്ന പ്രക്രിയയോ ആകാം. ഒരു നിശ്ചിത കാലയളവിലേക്കാണ് കണ്ടുപിടിക്കുന്ന ആൾക്ക് അയാളുടെ ഉൽപ്പന്നത്തിന്റെ മേൽ നിയമാധികാരമുള്ളത്. കുത്തകാവകാശം ലഭിക്കാൻ താഴെ പറയുന്ന നിബന്ധനകൾ പാലിക്കണം.

- ഒരു പ്രക്രിയയുമായോ ഉൽപ്പന്നവുമായോ ബന്ധപ്പെട്ടിരിക്കണം
- പുതിയതായിരിക്കണം
- കണ്ടെത്തലിനായുള്ള ഒരു ഘട്ടം അടങ്ങിയിരിക്കണം
- വ്യാവസായികോപയോഗത്തിന് അനുയോജ്യമാകണം
- മറ്റുള്ളവർക്ക് ഉപദ്രവകരമായിത്തീരുന്നതൊന്നും വികസിപ്പിക്കരുത്.

കുത്തകാവകാശ സംരക്ഷണമെന്നാൽ ഒരു ഉല്പന്നവും ഉടമസ്ഥന്റെ/യുടെ അനുവാദം കൂടാതെ ഉണ്ടാക്കാനോ ഉപയോഗിക്കാനോ വിതരണം ചെയ്യാനോ വില്പന നടത്താനോ പാടില്ലെന്ന നിയമ വ്യവസ്ഥയാണ്. ഒരു കണ്ടുപിടുത്തം മറ്റുള്ളവർ എങ്ങനെ ഉപയോഗിക്കണമെന്ന് തീരുമാനിക്കാനുള്ള അവകാശം കുത്തകാവകാശിക്ക് ലഭിക്കുന്നു. ഉടമ, തന്റെ കണ്ടു

പിടുത്തത്തിന്റെ അവകാശം മറ്റൊരാൾക്ക് നൽകിയാൽ പിന്നീട് അയാൾക്കായിരിക്കും അതിന്റെ കുത്തകാവകാശം. കുത്തകാവകാശത്തിനുള്ള അപേക്ഷ ലഭിക്കുന്നത് മുതൽ 20 വർഷത്തേക്കാണ് ഇന്ത്യയിൽ ഇതിന്റെ കാലാവധി. കുത്തകാവകാശ കാലാവധി കഴിഞ്ഞാൽ സുരക്ഷിതത്വം അവസാനിക്കുകയും എല്ലാവർക്കും സൗജന്യമായി കണ്ടുപിടുത്തം ഉപയോഗിക്കുകയും ചെയ്യാം. ചിത്രം 10.7ൽ കുടുക്കിനുള്ള (zipper) കുത്തകാവകാശം വിശദമായി ചേർത്തിരിക്കുന്നു.

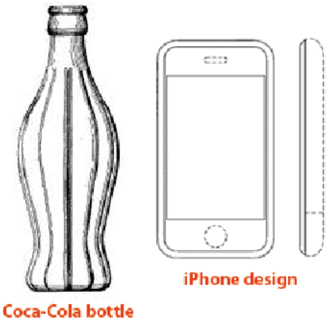


ചിത്രം. 12.10 : സിഷറിന്റെ ആദ്യ പേറ്റന്റ്

വ്യാപാരമുദ്ര (Trademark): ചില സാധനങ്ങളെ തിരിച്ചറിയുന്നതിനോ, ചില കമ്പനികളോ വ്യക്തികളോ നൽകുന്ന സേവനങ്ങളെ പ്രത്യേകമായി തിരിച്ചറിയുന്നതിനോ ഉപയോഗിക്കുന്ന പ്രത്യേക മുദ്രയാണിത്. ഇത് ഒരു ഉൽപ്പന്നത്തെയോ സേവനത്തെയോ തിരിച്ചറിയാൻ സഹായിക്കുന്ന പേരോ, അടയാള ചിഹ്നമോ, പ്രതീകമോ ആകാം. ഉൽപ്പന്നങ്ങളെ തിരിച്ചറിയുവാൻ വ്യാപാരമുദ്രയുടെ ഉടമയ്ക്ക് അത് ഉപയോഗിക്കാൻ അധികാരമുണ്ട്. ഉപയോക്താക്കൾക്ക് അത് തിരിച്ചറിയുവാനും ഉൽപ്പന്നത്തെയോ സേവനത്തെയോ വിലയ്ക്ക് വാങ്ങുവാനും സാധിക്കുന്നു. വ്യാപാരമുദ്ര രജിസ്റ്റർ ചെയ്തിരിക്കണം. 10 വർഷത്തേക്കാണ് ആദ്യ കാലാവധി. തുടർന്ന് ഇവ പുതുക്കാം. ഒരു വ്യക്തിയോ കമ്പനിയോ വ്യാപാരമുദ്ര ഉപയോഗിക്കുന്നുവോ എന്ന് കൺട്രോളർ ജനറൽ ഓഫ് പേറ്റന്റ് ഡിസൈൻസ് ആൻഡ് ട്രേഡ് മാർക്കിലൂടെ (http://ipindia-online.gov.in) അന്വേഷണം നടത്തി തിരിച്ചറിയാൻ സാധിക്കും. വ്യാപാരമുദ്ര രജിസ്ട്രേഷൻ അതാത് രാജ്യങ്ങൾക്ക് മാത്രം ബാധകമാണ്. ചിത്രം 10.8ൽ ഇന്ത്യയിലെ ചില പ്രശസ്തമായ വ്യാപാരമുദ്രകൾ കാണിച്ചിരിക്കുന്നു.

ചിത്രം. 12.11: പ്രശസ്തമായ വ്യാപാര മുദ്രകൾ

വ്യാവസായിക രൂപരേഖ (Industrial designs): ഒരു വസ്തുവിന്റെ ആലങ്കാരികമോ സൗന്ദര്യപരമോ ആയ സ്വഭാവ സവിശേഷതകൾ പ്രതിപാദിക്കുന്നതാണ് വ്യാവസായിക രൂപരേഖ. ഒരു വസ്തുവിന്റെ ത്രിമാന സവിശേഷതകളായ ആകൃതി, ഉപരിതലം എന്നിവയും ദ്വിമാന സവിശേഷതകളായ മാതൃക, വരകൾ, നിറങ്ങൾ എന്നിവയുമൊക്കെ ഈ രൂപകല്പനയിൽ ഉൾപ്പെട്ടേക്കാം. പ്രവർത്തനപരമായ സവിശേഷതകൾക്കപ്പുറം വസ്തുക്കളുടെ ദൃശ്യപരമായ രൂപഘടനയെ വ്യാവസായിക രൂപരേഖ അവകാശം സംരക്ഷിക്കുന്നു. വൈവിധ്യമാർന്ന വ്യവസായ ഉൽപ്പന്നങ്ങളിലും കരകൗശല വസ്തുക്കളിലും, മെഡിക്കൽ ഉപകരണങ്ങൾ, ആഭര



ചിത്രം. 12.12: പ്രശസ്തമായ വ്യാവസായിക രൂപരേഖകൾ

ണങ്ങൾ, വാച്ചുകൾ, വാഹനങ്ങൾ, വസ്ത്ര രൂപകല്പനകൾ എന്നിവയിലും വ്യാവസായിക രൂപരേഖ ഉപയോഗിക്കപ്പെടുന്നു. കൊക്കക്കോള കുപ്പിയുടെയും, ഐ ഫോണിന്റെയും രജിസ്റ്റർ ചെയ്ത രൂപകല്പന ചിത്രം 10.9 ൽ കാണിച്ചിരിക്കുന്നു.

ഭൗമശാസ്ത്രപരമായ സൂചകങ്ങൾ (Geographical indications)

ഭൂമിശാസ്ത്രപരമായ ഉത്ഭവം, ഗുണങ്ങൾ അല്ലെങ്കിൽ ആ പ്രദേശത്തിന്റേതായ ഒരു സൽപ്പേര് ഉൽപ്പന്നങ്ങളിൽ ഉപയോഗിക്കുന്നതിനെയാണ് ഭൗമശാസ്ത്രപരമായ സൂചകങ്ങൾ എന്ന് പറയുന്നത്. കാർഷികോൽപ്പന്നങ്ങൾക്ക് അവയുടെ ഉൽപ്പാദന സ്ഥലത്തിന്റെ ഗുണങ്ങളോടൊപ്പം അവിടുത്തെ മണ്ണിന്റെയും കാലാവസ്ഥയുടെയും സ്വാധീനവും ഉണ്ടാകും. ഉത്ഭവ സ്ഥലം ഗ്രാമമോ നഗരമോ ചില മേഖലകളോ രാജ്യമോ ആകാം. ആറന്മുളക്കണ്ണാടിയും പാലക്കാടൻ മട്ട അരിയും കേരളവുമായി ബന്ധപ്പെട്ട ഭൗമശാസ്ത്രപരമായ സൂചകങ്ങൾക്കുള്ള ചില പ്രശസ്തമായ ഉൽപ്പന്നങ്ങൾ ആണ്.

ചിത്രം. 12.13: കേരളവുമായ ബന്ധപ്പെട്ട ഭൗമശാസ്ത്രപരമായ സൂചകങ്ങൾ

b) പകർപ്പവകാശം (Copyright)

ഒരു നിശ്ചിത കാലത്തേക്ക് ഒരു സ്രഷ്ടാവിന് തന്റെ സൃഷ്ടിയിന്മേലുള്ള അവകാശമാണ് പകർപ്പവകാശം. പുസ്തകങ്ങൾ, സംഗീതം, ചിത്രങ്ങൾ, ശില്പങ്ങൾ, സിനിമ, പരസ്യം, കമ്പ്യൂട്ടർ സോഫ്റ്റ്‌വെയർ തുടങ്ങിയവയിലുള്ള സൃഷ്ടിപരമോ ബൗദ്ധികമോ കലാപരമോ ആയ പ്രവർത്തനങ്ങളുടെ വൈവിധ്യത്തെ ബാധകമാക്കിയുള്ള അവകാശമാണിത്. സൃഷ്ടികളുടെ പുനരുൽപ്പാദനം, പൊതുജനവുമായുള്ള ആശയവിനിമയം, അനുരൂപീകരണം, പരിഭാഷ എന്നിവയൊക്കെ ഈ അവകാശത്തിന്റെ പരിധിയിൽപ്പെടുന്നു. ഇന്ത്യയിൽ 1958 ജനുവരിയിലാണ് പകർപ്പവകാശനിയമം – 1957 നിലവിൽ വന്നത്. ഈ നിയമത്തിൽ അഞ്ച് പ്രാവശ്യം ഭേദഗതി വരുത്തി. 2012 ൽ പകർപ്പവകാശ സംരക്ഷണ നിയമത്തിൽ ചില സുപ്രധാന ഭേദഗതികൾ വരുത്തി. ഡിജിറ്റൽ പരിസ്ഥിതിയിലേക്കുള്ള ഈ നിയമത്തിന്റെ വ്യാപനം, ഇന്റർനെറ്റ് സേവന ദാതാക്കളുടെ ബാധ്യതകൾ, സംഗീത സംവിധായകർക്ക് റോയൽറ്റി ലഭിക്കുന്നതിനുള്ള അവകാശം ഉറപ്പാക്കൽ, അംഗപരിമിതർ ഏതെങ്കിലും തരത്തിലുള്ള ഉല്പന്നമോ സേവനമോ ഉപയോഗിക്കുന്നത് പകർപ്പവകാശ സംരക്ഷണ നിയമത്തിന്റെ പേരിൽ തടയാതിരിക്കൽ എന്നിവയാണ് ഭേദഗതികൾ.

ഇന്ത്യൻ പകർപ്പവകാശനിയമപ്രകാരം, ഒരു പ്രവൃത്തി അതിന്റെ സൃഷ്ടികർമത്തിൽത്തന്നെ പകർപ്പവകാശത്തിൽ സ്വമേധയാ സംരക്ഷിക്കപ്പെടുന്നു. പകർപ്പവകാശത്തിന് എഴുത്തുകാരന്റെ മരണശേഷം 60 വർഷം വരെ നിയമ സാധുത ഉണ്ടായിരിക്കും. പകർപ്പവകാശ രജിസ്ട്രേഷൻ സൃഷ്ടിപരമായ പ്രവർത്തനങ്ങൾക്ക് നിയമപരമായ പരിരക്ഷ നൽകുന്നു. ഇത് സ്രഷ്ടാവിന് സൃഷ്ടിക്കുമേൽ നിയമപരമായ അവകാശം നൽകിക്കൊണ്ട് അത് ഒരു ബൗദ്ധികസ്വത്താക്കി മാറ്റുന്നു.

Copyright Office
Government of India

ചിത്രം. 12.14: കോപ്പിറൈറ്റ് ഓഫീസ്, ഇന്ത്യയുടെ ലോഗോ

പകർപ്പവകാശം ലഭിക്കുന്നതിന് രജിസ്റ്റർ ചെയ്യേണ്ട ആവശ്യമില്ല എന്ന വസ്തുത ശ്രദ്ധിക്കേണ്ടതാണ്. ഇന്ത്യയിലെ പകർപ്പവകാശ രജിസ്ട്രേഷൻ മാനവവിഭവവികസന മന്ത്രാലയത്തിന് കീഴിലുള്ള പകർപ്പവകാശ കാര്യാലയമാണ് കൈകാര്യം ചെയ്യുന്നത്. ഇന്ത്യൻ കാര്യാലയത്തിന്റെ ഓഫീസിന്റെ ലോഗോ ചിത്രം 10.11 ൽ നൽകിയിരിക്കുന്നു.

പട്ടിക 10.2 ൽ ബൗദ്ധികസ്വത്തിന്റെ രജിസ്ട്രേഷൻസ്ഥിതി വ്യക്തമാക്കാൻ ഉപയോഗിക്കുന്ന വിവിധ ചിഹ്നങ്ങൾ കാണിച്ചിരിക്കുന്നു

ഒരു സൃഷ്ടിയുടെ പകർപ്പവകാശ ഉടമകൾക്ക് ചുവടെ കൊടുത്തിരിക്കുന്ന പ്രവർത്തനങ്ങൾക്ക് അംഗീകാരം നൽകാനോ നിഷേധിക്കാനോ കഴിയും.

- അച്ചടിച്ച രൂപവും ശബ്ദ റെക്കോർഡിങ്ങും ഉൾപ്പടെ എല്ലാ രൂപത്തിലുമുള്ള അതിന്റെ പുനഃസൃഷ്ടി;
- പൊതുവായ കാര്യനിർവഹണവും പൊതുജനങ്ങൾക്കുള്ള ആശയവിനിമയവും;
- അതിന്റെ സംപ്രേഷണം;
- മറ്റു ഭാഷകളിലേക്കുള്ള വിവർത്തനം;
- നോവലിനെ സിനിമയുടെ തിരക്കഥയാക്കി മാറ്റുന്നതു പോലെയുള്ള അനുരൂപീകരണങ്ങൾ.

സ്രഷ്ടാക്കൾ തങ്ങളുടെ സൃഷ്ടികൾ സാമ്പത്തികനേട്ടത്തിനു വേണ്ടി വ്യക്തികൾക്കോ, കമ്പനികൾക്കോ വിൽക്കാറുണ്ട്.

ബൗദ്ധിക സ്വത്തവകാശം	
രജിസ്റ്റർ ചെയ്ത വ്യാപാരമുദ്ര	®
രജിസ്റ്റർ ചെയ്യാത്ത വ്യാപാരമുദ്ര	TM
പകർപ്പവകാശം	©
പകർപ്പവകാശ റെക്കോർഡിങ് ശബ്ദം	!

പട്ടിക 12.4: ബൗദ്ധിക സ്വത്തവകാശത്തിൽ ഉപയോഗിക്കുന്ന അടയാളങ്ങൾ

കമ്പ്യൂട്ടർ സോഫ്റ്റ്‌വെറുകൾ (സോഴ്സ് കോഡ്, ഡാറ്റബേസ്, വെബ് സൈറ്റുകൾ എന്നിവ) ഒരു സാഹിത്യ സൃഷ്ടി പോലെ പകർപ്പവകാശത്തിന് വിധേയമാക്കാം. കമ്പ്യൂട്ടർ സോഫ്റ്റ്‌വെയർ പകർപ്പവകാശത്തിനു കീഴിൽ സംരക്ഷിക്കപ്പെട്ടിട്ടുണ്ടെങ്കിലും ഇപ്പോൾ അവ കുത്തകാവകാശം നേടിയെടുക്കുന്നു. സോഫ്റ്റ്‌വെയർ വികസനം ഒരു വ്യവസായമായി കണക്കാക്കപ്പെട്ടതും പകർപ്പവകാശവുമായി താരതമ്യം ചെയ്യുമ്പോൾ കുത്തകാവകാശം മികച്ച സംരക്ഷണം നൽകുന്നതും ഇതിനൊരു കാരണമാണ്. കുത്തകാവകാശ സംരക്ഷണത്തിന് ആവശ്യമായ മാനദണ്ഡങ്ങൾ കർശനമാണ് എന്നത് ശ്രദ്ധിക്കേണ്ടതാണ്. പട്ടിക 10.3 ൽ അവകാശങ്ങളും അനുബന്ധ ബൗദ്ധിക സ്വത്തുക്കളും പ്രദർശിപ്പിച്ചിരിക്കുന്നു.

ബൗദ്ധിക സ്വത്തവകാശത്തിന്റെ രജിസ്ട്രേഷനിലെ വ്യത്യാസങ്ങൾ പട്ടിക 10.4 ൽ കൊടുത്തിരിക്കുന്നു.

ബുദ്ധി	വസ്തു	അവകാശം
ആശയം	കണ്ടുപിടുത്തം/പുതുമ	കുത്തകാവകാശം
ആശയം	മേന്മ + സവിശേഷത	വ്യാപാരമുദ്ര
ആശയം	രൂപം	രൂപരേഖ
സാധനം	ഭൂമിശാസ്ത്രപരമായ ഉത്ഭവം	ഭൂമിശാസ്ത്രപരമായ സൂചകങ്ങൾ
ആശയം	ഭാവനകൾ	പകർപ്പവകാശം

പട്ടിക 12.5: ബൗദ്ധിക സ്വത്തവകാശത്തിന്റെ അവകാശങ്ങൾ

	കുത്തകാവകാശം	വ്യാപാരമുദ്ര	പകർപ്പവകാശം
പരാമർശിക്കുന്നു	ഉൽപ്പന്നം, പ്രക്രിയ	പേര്, അടയാള ചിഹ്നം, അടയാളങ്ങൾ	സൃഷ്ടിപരമായ, ബുദ്ധിപരമായ അല്ലെങ്കിൽ കലാപരമായ പ്രവർത്തന രീതികൾ
രജിസ്ട്രേഷൻ	ആവശ്യമാണ്	ആവശ്യമാണ്	യാന്ത്രികമായി രജിസ്റ്റർ ചെയ്യാൻ കഴിയും
കാലാവധി	20 വർഷം	10 വർഷം	ഒടുവിലത്തെ എഴുത്തുകാരന്റെ മരണശേഷം 60 വർഷം വരെ
പുനരുൽപ്പാദനം	ഇല്ല	ഉണ്ട്	

പട്ടിക 12.6: ബൗദ്ധിക സ്വത്തവകാശത്തിന്റെ രജിസ്ട്രേഷനിലെ വ്യത്യാസങ്ങൾ

12.2.2 അവകാശ കയ്യേറ്റം (Infringement)

അനുമതി കൂടാതെ കുത്തകാവകാശമോ, പകർപ്പവകാശമോ, വ്യാപാരമുദ്രയോ ഉപയോഗിക്കുന്നത് ബൗദ്ധിക സ്വത്തിന്മേലുള്ള കയ്യേറ്റമാണ്. ബൗദ്ധിക സ്വത്തിന്റെ സ്വഭാവം, അധികാര പരിധി (രാജ്യങ്ങൾ തമ്മിൽ), പ്രവൃത്തിയുടെ സ്വഭാവം എന്നിവയുടെ അടിസ്ഥാനത്തിൽ ഇത് പൗരാവകാശ നിയമത്തിന്റെയോ ക്രിമിനൽ നിയമത്തിന്റെയോ ലംഘനമായിരിക്കാം.

കുത്തകാവകാശമുള്ള ആളുടെ (patent holder) അനുമതിയില്ലാതെ കുത്തകാവകാശമുള്ള കണ്ടുപിടുത്തം ഉപയോഗിക്കുകയോ വിൽക്കുകയോ ചെയ്യുന്നത് കുത്തകാവകാശ കയ്യേറ്റമാണ്. ഗവേഷണാവശ്യങ്ങൾക്ക് വേണ്ടി കുത്തകാവകാശമുള്ള കണ്ടുപിടുത്തങ്ങൾ ഉപയോഗിക്കുന്നതിന് ധാരാളം രാജ്യങ്ങൾ അനുമതി നൽകുന്നു. മൊബൈൽ ഫോൺ സാങ്കേതികവിദ്യയിൽ ആപ്പിളും സാംസങ്ങും തമ്മിലുള്ള നിയമപരമായ തർക്കം കുത്തകാവകാശ കയ്യേറ്റത്തിന് ഒരുദാഹരണമാണ്.

ഒരാളുടെ വ്യാപാരമുദ്രയോട് സാമ്യമുള്ള വ്യാപാരമുദ്ര മറ്റൊരാൾ ഉപയോഗിക്കലാണ് വ്യാപാരമുദ്ര കയ്യേറ്റം. ഇവിടെ രണ്ടു കക്ഷികളും ഉപയോഗിക്കുന്നത് ഒരേ ഉൽപ്പന്നം അഥവാ സേവനം തന്നെയായിരിക്കും. നിയമപരമായ ആനുകൂല്യം ലഭിക്കാൻ വ്യാപാരമുദ്ര രജിസ്റ്റർ ചെയ്യണം. 'HORLIKS' എന്ന വ്യാപാര നാമം ഉപയോഗിച്ച് ഒരു ചെറിയ കമ്പനി മിഠായികൾ നിർമിച്ചത് 'HORLICKS' എന്ന വ്യാപാര നാമം ഉപയോഗിച്ചിരുന്ന കമ്പനിയുടെ വ്യാപാര മുദ്രാവകാശത്തിൻമേലുള്ള കയ്യേറ്റത്തിനുദാഹരണമാണ്.

പകർപ്പവകാശ കയ്യേറ്റം എന്നത് സൃഷ്ടാവിന്റെ അനുമതിയില്ലാതെ അയാളുടെ കൃതി പുനഃസൃഷ്ടിക്കുകയോ വിതരണം ചെയ്യുകയോ പ്രദർശിപ്പിക്കുകയോ മാറ്റുകയോ പരിഭാഷപ്പെടുത്തുകയോ ആണ്. ഇതിനെ സാഹിത്യ മോഷണം എന്ന് വിളിക്കുന്നു.

സോഫ്റ്റ്‌വെയറുകൾ നിർമ്മാതാക്കളുടെ അനുവാദമോ, അംഗീകാരമോ കൂടാതെ മോഷ്ടിച്ച് മറ്റു കമ്പ്യൂട്ടറുകളിൽ അനധികൃതമായി കോപ്പി ചെയ്യുന്നതിനെ സോഫ്റ്റ്‌വെയർ പൈറസി (മോഷണം) എന്ന് പറയുന്നു. ഒരു സംഗീതശകലം ഗാനരചയിതാവിന്റെയോ, കലാകാരന്റെയോ, ഗാനത്തിന്റെ പകർപ്പവകാശമുള്ള കമ്പനിയുടെയോ അനുവാദമില്ലാതെ കോപ്പി ചെയ്യുകയോ വിതരണം ചെയ്യുകയോ ചെയ്യുന്ന പ്രവർത്തനത്തെ മ്യൂസിക് പൈറസി എന്ന് പറയുന്നു.

നിങ്ങളുടെ പുരോഗതി അറിയുക

1. മനസ്സിന്റെ സൃഷ്ടി ____________ എന്നറിയപ്പെടുന്നു.
2. W I P O എന്നതിന്റെ പൂർണരൂപം എഴുതുക.
3. ബൗദ്ധിക സ്വത്തവകാശത്തെ ____________ എന്നും, ____________ എന്നും രണ്ടായി തരംതിരിക്കുന്നു.
4. കുത്തകാവകാശം എന്നത് ____________ പ്രത്യേകമായി നൽകിയ അവകാശമാണ്.
5. ____________ എന്നത് ഒരു ഉൽപന്നമോ സേവനമോ തിരിച്ചറിയാനുള്ള അടയാളമാണ്
6. വ്യാവസായിക രൂപരേഖ എന്നാൽ എന്താണ്?
7. ഭൗമശാസ്ത്രപരമായ സൂചകങ്ങളുടെ പ്രാധാന്യമെന്ത്?

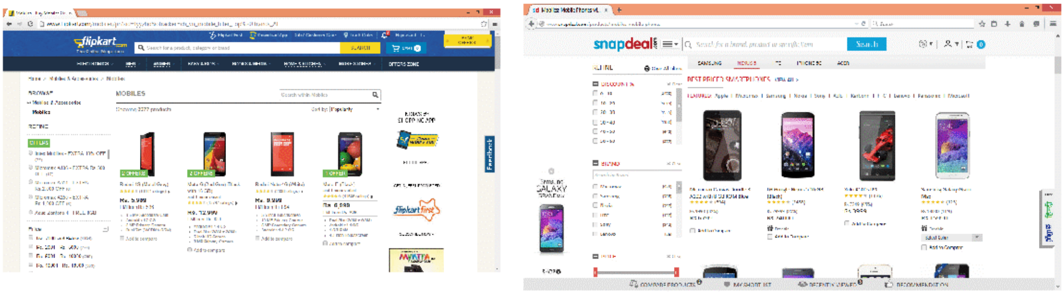
12.2.3 സൈബർ ഇടം (Cyber space)

ഇന്റർനെറ്റുമായി ബന്ധിപ്പിച്ചിട്ടുള്ള കംപ്യൂട്ടർ സംവിധാനങ്ങൾ സൃഷ്ടിച്ച ഒരു അയഥാർഥ പരിതസ്ഥിതിയാണ് സൈബർ ഇടം. ഒരു പക്ഷേ, ഒരു സ്ഥലത്ത് പല കാര്യങ്ങളും ചെയ്യാൻ കഴിയുന്ന ഒരു സമ്പ്രദായത്തെ സൂചിപ്പിക്കാനാണ് ഈ പദം ഉപയോഗിക്കുന്നത്. സൈബർ ഇടം നമ്മുടെ ജീവിതത്തിൽ സ്വാധീനം ചെലുത്തുന്ന വിവിധ അവസരങ്ങൾ നമുക്ക് ചർച്ച ചെയ്യാം. ആശയവിനിമയത്തിനായി മുൻകാലങ്ങളിൽ നിലവിലിരുന്ന പോസ്റ്റൽ സേവനത്തിന്റെ സ്ഥാനത്ത് ഇന്ന് ഇ-മെയിൽ ഏറെ സ്വീകാര്യതയും നിയമസാധുതയും നേടിയിരിക്കുന്നു. ഹയർ സെക്കന്ററി വിദ്യാഭ്യാസ വകുപ്പിൽ വിദ്യാർഥിപ്രവേശനം, പരീക്ഷ, നാഷണൽ സർവീസ് സ്കീം, ഭരണനിർവഹണം എന്നിവയെല്ലാം വെബ്സൈറ്റിന്റെയും ഇ-മെയിലുകളുടെയും സഹായത്തോടെയാണ് നടക്കുന്നത്. സാധാരണ ജനങ്ങളിൽ ഇന്റർനെറ്റ് പ്രശസ്തമായ ആശയവിനിമയ മാധ്യമമായി മാറിയിരിക്കുന്നു.

ഇന്ത്യയിലെ ഒരു മെട്രോ നഗരത്തിൽ 2012 ഡിസംബറിൽ ഓടിക്കൊണ്ടിരുന്ന ബസ്സിനുള്ളിൽ വെച്ച് ഒരു പെൺകുട്ടി ശാരീരിക പീഡനത്തിനിരയായി. കുറച്ചു ദിവസങ്ങൾക്കുശേഷം ആ കുട്ടി മരണപ്പെട്ടു. ജീവിതത്തിന്റെ വിവിധ മേഖലകളിലുള്ളവർ സാമൂഹിക മാധ്യമങ്ങൾ വഴി ഇതിനെതിരെ ശക്തമായി പ്രതികരിച്ചു. ഇതിന്റെ ഫലമായി ലോക്സഭ 2013ൽ ലൈംഗികാതിക്രമങ്ങൾക്ക് എതിരെ The Criminal Law (Amendent) Act 2013 എന്ന ശക്തമായ നിയമം കൊണ്ടുവന്നു. ഇന്റർനെറ്റ് ഉപയോഗിച്ച് സാമൂഹിക മാധ്യമങ്ങൾ വഴി സംഘടിപ്പിച്ച ഈ പ്രതിഷേധത്തിന് സമൂഹത്തെ സ്വാധീനിക്കാനായി.

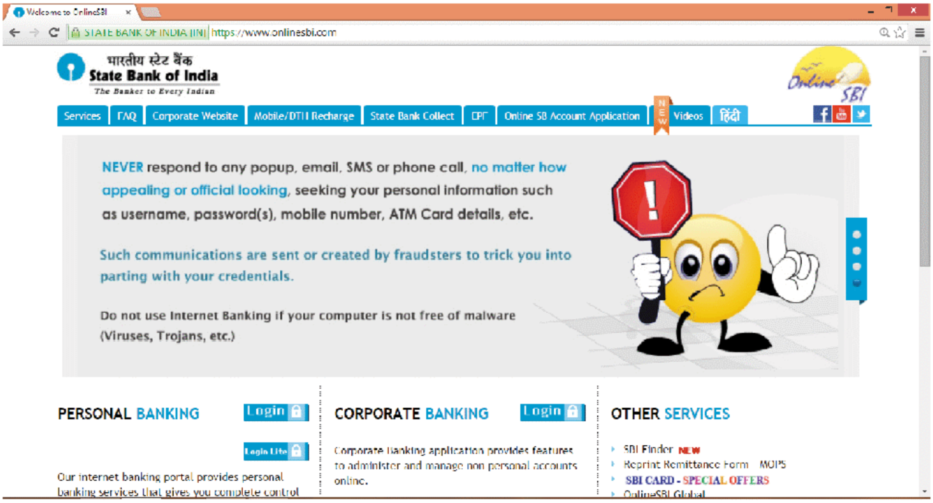
പണ്ട് കടയിൽ പോയി സാധനങ്ങൾ വാങ്ങിയിരുന്നതിന് പകരം ഇന്റർനെറ്റ് വ്യാപാര സൈറ്റുകൾ വഴി ഇന്ന് സാധനങ്ങൾ വാങ്ങാവുന്നതാണ്. ഉൽപ്പന്നങ്ങളെക്കുറിച്ചുള്ള വിവരങ്ങൾ ഇത്തരം ഇ-വാണിജ്യ സൈറ്റുകൾ നൽകുന്നു. അനായാസമായി സാധനങ്ങൾ

വാങ്ങുന്നതിനുള്ള സൗകര്യം ഇതുവഴി സാധ്യമാകുന്നു. ഇന്റർനെറ്റ് ബാങ്കിങ്, ക്രെഡിറ്റ് കാർഡ്, ഡെബിറ്റ് കാർഡ് എന്നിവ വഴി സാമ്പത്തിക വിനിമയങ്ങളും ലളിതമായി. എല്ലാ സേവനങ്ങളും ഓൺലൈനാക്കി മാറ്റുവാൻ ഇന്റർനെറ്റിന്റെ സഹായത്തോടെ ഇപ്പോൾ സാധിക്കുന്നു. ചിത്രം 12.12ൽ ചില ഇ-വാണിജ്യ വെബ് സൈറ്റുകൾ കാണിച്ചിരിക്കുന്നു.



ചിത്രം 12.12: ഇ - വാണിജ്യ വെബ്സൈറ്റിന് ഒരു ഉദാഹരണം

ഇന്ന് എല്ലാ ബാങ്കുകളും അവരുടെ ഉപയോക്താക്കൾക്ക് ഇന്റർനെറ്റ് ബാങ്കിങ് സൗകര്യം ഏർപ്പെടുത്തിയിരിക്കുന്നു. സാമ്പത്തിക വിനിമയം നടത്തുവാനും, ടെലിഫോൺ ബിൽ, വൈദ്യുത ബിൽ, ഓൺലൈനായി വ്യാപാരം നടത്തൽ, ട്രെയിൻ ടിക്കറ്റ് ബുക്കിംഗ്, സിനിമ ടിക്കറ്റ് ബുക്കിംഗ് എന്നിവയ്ക്കെല്ലാം ഇന്റർനെറ്റ് ബാങ്കിങ് സൗകര്യം ഉപയോഗപ്പെടുത്തുന്നു. ഈ പ്രവർത്തനങ്ങൾക്ക് ബന്ധപ്പെട്ട ഓഫീസുകൾ സന്ദർശിക്കുന്നതുമായി താരതമ്യം ചെയ്യുമ്പോൾ ഇന്റർനെറ്റ് ബാങ്കിങ്ങിലൂടെ സമയവും അധ്വാനവും ലാഭിക്കാൻ കഴിയുന്നു. അങ്ങനെ ബാങ്ക് ഇടപാടുകളും ഇന്റർനെറ്റിലൂടെ ലഭ്യമാകുന്ന ജനപ്രിയ സേവനമായി മാറി. ഒരു ഇന്റർനെറ്റ് ബാങ്കിങ് വെബ്സൈറ്റ് ചിത്രം 10.13 ൽ കാണിച്ചിരിക്കുന്നു. കാലക്രമത്തിൽ കൂടുതൽ സേവനങ്ങൾ ഓൺലൈനായി മാറിക്കൊണ്ടിരിക്കുന്നു. വെബ്ബിലെ വിവിധ സേവനങ്ങൾ ലഭിക്കുന്നതിനായി ധാരാളം ആളുകൾ ഗണ്യമായ സമയം ഇന്റർനെറ്റിനു മുമ്പിൽ വിനിയോഗിക്കുന്നു. ഇന്റർനെറ്റ് പലപ്പോഴും സൈബർ ഇടം എന്നും അറിയപ്പെടുന്നു. സൈബർ ഇടത്തെ പ്രതീകാത്മകമായി ചിത്രം 10.14 ൽ കാണിച്ചിരിക്കുന്നു.

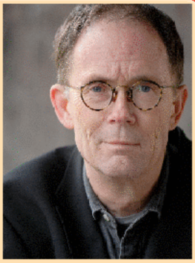


ചിത്രം 12.13: ഇന്റർനെറ്റ് ബാങ്കിംഗ് വെബ്സൈറ്റ്

കമ്പ്യൂട്ടർ നെറ്റ്‌വർക്കുകളിലൂടെ ആശയവിനിമയം നടത്തുന്ന ഒരു പ്രതീതി ലോകമാണ് സൈബർ ഇടം. വിവരങ്ങളുടെ പ്രധാന പാതയായ ഇവിടെ വ്യക്തിപരമായ വിവരങ്ങൾ ശേഖരിക്കുകയും, ആശയങ്ങൾ വിനിമയം നടത്തുകയും, പരസ്പരം സംവാദങ്ങളിൽ ഏർപ്പെടുകയും, സാമൂഹിക സഹായം നൽകുകയും, വ്യാപാരം നടത്തുകയും, വിനോദങ്ങളിൽ ഏർപ്പെടുകയും, ചർച്ചകളിൽ മുഴുകുകയും ചെയ്യുന്നു. സാമൂഹിക ഇടപെടലുകൾക്ക് ആധിപത്യമുള്ള ഒരു സ്ഥലമാണ് സൈബർ ഇടം. ഇഷ്ടാനുസരണം ഏതു പ്രവർത്തനത്തിൽ ഏർപ്പെടാനും എന്തും സ്വതന്ത്രമായി പ്രകടിപ്പിക്കാനുമുള്ള ഒരു അനിയന്ത്രിത ഇലക്ട്രോണിക് മാധ്യമായി ചിലർ സൈബർ ഇടത്തെ കാണുന്നു. അത്തരം പ്രവൃത്തികൾ മറ്റുള്ളവർക്ക് പലപ്പോഴും ബുദ്ധിമുട്ടുകൾ സൃഷ്ടിക്കുകയും തെറ്റിലേക്ക് നയിക്കുകയും ചെയ്യുന്നു. അതിനാൽ ഇന്റർനെറ്റ് ഉപയോഗിക്കുന്ന ഒരു വ്യക്തി മറ്റുള്ളവർക്ക് ഗുണകരമാകുന്ന വിധത്തിൽ നിയമങ്ങളും സദാചാരമൂല്യങ്ങളും പാലിക്കേണ്ടതാണ്. നിയന്ത്രണരഹിതമായ ഈ ഇടം കുറ്റവാളികൾക്കും ലഭ്യമാണ്. സൈബർ ഇടത്തിലെ ആശയവിനിമയം, സാമ്പത്തിക ഇടപാടുകൾ മുതലായവ ദിനംപ്രതി വർദ്ധിക്കുന്നു. അതിനാൽ കുറ്റകൃത്യങ്ങൾക്കുള്ള മാധ്യമമായും ഇത് മാറുന്നു. അതുകൊണ്ട് തന്നെ സൈബർ ഇടത്തിന്റെ സുരക്ഷ ഗൗരവമേറിയ പരിഗണനവിഷയമായി മാറിയിരിക്കുന്നു.

ചിത്രം 10.14: സൈബർ ഇടത്തിന്റെ പ്രതീകാത്മക അവതരണം

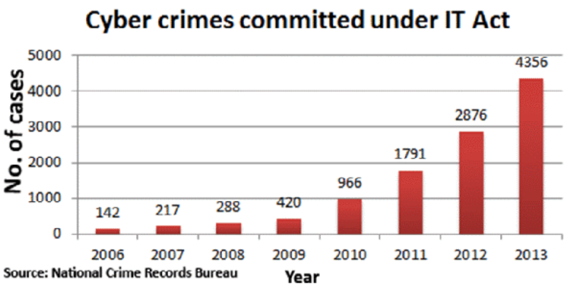
സൈബർ ഇടം എന്ന പദം സൃഷ്ടിച്ചത് കനേഡിയൻ ശാസ്ത്രനോവൽ എഴുത്തുകാരനായ വില്യം ഗിബ്സൺ ആണ്. അദ്ദേഹത്തിന്റെ ഹ്രസ്വ കഥയായ ബേണിങ് ക്രോമിൽ 1982ലാണ് ഇത് പരാമർശിച്ചിരിക്കുന്നത്. യഥാർഥ ലോകത്തിൽ ഇല്ലാത്ത സംഭവങ്ങൾ അല്ലെങ്കിൽ ഇടപാടുകൾ നടക്കുന്ന ഒരു ലോകത്തെ പ്രതിനിധാനം ചെയ്യാൻ അദ്ദേഹം അത് ഉപയോഗിച്ചു. സൈബർ ഇടം എന്നത് പൊതുവായ ഒരു മായാലോകമാണ് എന്ന് അദ്ദേഹം വിശ്വസിച്ചു. മായാദൃശ്യം എന്നത് സൈബർ ഇടത്തിന്റെ ഉള്ളിലുള്ള ജീവന്റെ സ്വഭാവമാണ് എന്ന് ഗിബ്സൺ പറയുന്നു.



12.2.4 സൈബർ കുറ്റകൃത്യങ്ങൾ (Cyber crimes)

വിനോദത്തിനും വ്യാപാരത്തിനും ആശയവിനിമയത്തിനും വിദ്യാഭ്യാസാവശ്യങ്ങൾക്കുമായി ഇന്റർനെറ്റ് പുതിയ വാതായനങ്ങൾ തുറക്കുന്നു എന്ന് നമുക്കറിയാം. അതേസമയം ചിലർ നിയമവിരുദ്ധമായ പ്രവർത്തനങ്ങൾക്കും ഇന്റർനെറ്റ് ഉപയോഗിക്കുന്നു.

ഓൺലൈൻ ബാങ്കിങ്ങിനും മറ്റ് സാമ്പത്തിക ഇടപാടുകൾക്കുമായി സ്മാർട്ട് ഫോണുകളിലൂടെയും ടാബ്‌ലറ്റുകളിലൂടെയുമുള്ളള വർധിച്ച ഇന്റർനെറ്റ് ഉപയോഗം അപകട സാധ്യതയും വർധിപ്പിക്കുന്നു. ഉയർന്ന തോതിലുള്ള ഇന്റർനെറ്റ് നുഴഞ്ഞു കയറ്റവും ഓൺലൈൻ ബാങ്കിങ് സംവിധാനങ്ങളും സൈബർ അക്രമികൾക്ക് ഉപദ്രവകാരികളായ



ചിത്രം 12.15: ഐ.ടി. നിയമത്തിന്റെ പരിധിയ്ക്കുള്ളിൽപ്പെട്ട സൈബർ കുറ്റകൃത്യങ്ങൾ

സോഫ്റ്റ്‌വെയർ (മാൽവെയർ) ഉപയോഗിച്ചോ നിയമവിരുദ്ധമായ ഹാക്കിങ് നടത്തിയോ ഓൺലൈൻ സാമ്പത്തിക ഇടപാടുകളെ ഉന്നം വയ്ക്കാൻ അവസരം ഒരുക്കുന്നു.

നാഷണൽ ക്രൈം റെക്കോർഡ്സ് ബ്യൂറോയുടെ കണക്കു പ്രകാരം കഴിഞ്ഞ വർഷങ്ങളിൽ നടന്ന സൈബർകുറ്റകൃത്യങ്ങളുടെ സ്ഥിതിവിവരക്കണക്ക് ചിത്രം 11.17 ലെ ഗ്രാഫിൽ കാണിച്ചിരിക്കുന്നു. ഇന്ത്യയിൽ സൈബർ കുറ്റകൃത്യങ്ങൾ ഗണ്യമായി വർധിക്കുന്നുവെന്ന് ഗ്രാഫിൽ കാണാൻ സാധിക്കും.

കമ്പ്യൂട്ടറുകളും കമ്പ്യൂട്ടർ നെറ്റ്‌വർക്കുകളും ഒരു ടൂളായോ അവയെ ഒരു ലക്ഷ്യമായോ അതുമല്ലെങ്കിൽ അവ ഒരു കുറ്റകൃത്യം നടത്തുന്നതിനുള്ള സ്ഥലമായോ ഉപയോഗിക്കുന്നതിനെയാണ് സൈബർ കുറ്റകൃത്യം എന്ന് നിർവചിച്ചിരിക്കുന്നത്. സൈബർ കുറ്റകൃത്യത്തിന്റെ ഒരു സുപ്രധാന ഘടകം അതിന്റെ പ്രാദേശികമല്ലാത്ത സ്വഭാവമാണ്. വളരെ അകലങ്ങളിലായി വേർതിരിക്കപ്പെട്ടിട്ടുള്ള നിയമവ്യവസ്ഥകളിൽ ഒരു കുറ്റകൃത്യം സംഭവിക്കാം. ഒരു ആക്രമണകാരിക്ക് ഒരു രാജ്യത്ത് നിന്നും മറ്റൊരു രാജ്യത്തെ ലക്ഷ്യമാക്കി ആക്രമണം നടത്തുവാൻ കഴിയും. അതിനാൽ വിവിധ രാജ്യങ്ങളിലെ അന്വേഷണ സംഘവും നീതിന്യായക്കോടതികളും കൈകോർക്കേണ്ടത് ആവശ്യമാണ്. ഇന്റർനെറ്റിന്റെ അജ്ഞാത സ്വഭാവം കാരണം ആളുകൾ വിവിധ കുറ്റകൃത്യങ്ങളിൽ ഏർപ്പെടാൻ സാധ്യതയുണ്ട്. ആളുകൾ അറിഞ്ഞോ അറിയാതെയോ സൈബർ കുറ്റകൃത്യങ്ങളിൽ ഏർപ്പെടുന്നു.

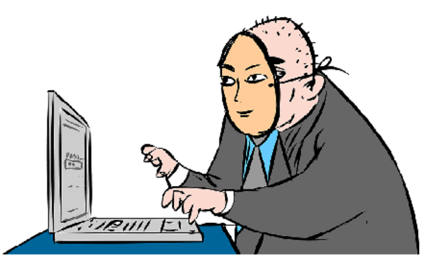
ഫിഷിങ്, ഹാക്കിങ്, സേവന ആക്രമണങ്ങളുടെ നിഷേധം എന്നിവ സൈബർ കുറ്റകൃത്യങ്ങളിൽ ഉൾപ്പെടുന്നു. ഇവ പതിനൊന്നാം ക്ലാസ്സിലെ 9-ാം അധ്യായത്തിൽ നാം പഠിച്ചിട്ടുണ്ട്. അനുവാദമില്ലാതെയുള്ള കമ്പ്യൂട്ടർ ഉപയോഗം, ക്രെഡിറ്റ് കാർഡ് വഞ്ചന, നിയമാനുസൃതമല്ലാതെയുള്ള ഡൗൺലോഡ്, അശ്ലീല ചിത്രങ്ങൾ, സൈബർ ഭീകരവാദം, വൈറസുകൾ, ഉപയോഗശൂന്യമായ ഇ-മെയിലുകൾ (സ്പാമുകൾ) എന്നിവയൊക്കെ കമ്പ്യൂട്ടർ കുറ്റകൃത്യങ്ങളിൽപ്പെടുന്ന പ്രവർത്തനങ്ങളാണ്.

സൈബർ കുറ്റകൃത്യങ്ങളെ പ്രധാനമായും മൂന്നായി തരം തിരിച്ചിരിക്കുന്നു- വ്യക്തികൾക്കെതിരെയുള്ള കുറ്റകൃത്യങ്ങൾ, സ്വത്തിനെതിരെയുള്ള കുറ്റകൃത്യങ്ങൾ, ഭരണകൂടത്തിനെതിരെയുള്ള കുറ്റകൃത്യങ്ങൾ.

a) വ്യക്തികൾക്കെതിരെയുള്ള സൈബർ കുറ്റകൃത്യങ്ങൾ (Cyber crimes against individuals)

ബ്രോഡ്ബാൻഡ് ഇന്റർനെറ്റിന്റെ പ്രചാരം ദൈനംദിന പ്രവത്തനങ്ങളിൽ ഇന്റർനെറ്റിനുള്ള സ്വാധീനം വർധിപ്പിച്ചു. ഇത് ഓൺലൈൻ കുറ്റകൃത്യങ്ങളുടെ അപകട സാധ്യതയും വർധിപ്പിച്ചു. ഒരു വ്യക്തിക്ക് ശാരീരികമോ മാനസികമോ ആയ ബുദ്ധിമുട്ടുകൾ ഉണ്ടാക്കുന്ന മറ്റൊരാളുടെ സൈബർ ഇടത്തിലെ പ്രവൃത്തി സൈബർ കുറ്റകൃത്യമായി കണക്കാക്കുന്നു. ഒരാളുടെ സ്വകാര്യതയിൽ ഇടപെടുക, ആൾമാറാട്ടം നടത്തുക, മറ്റൊരാളുടെ പേരിൽ ഇന്റർനെറ്റ് ഉപയോഗിക്കുക, മറ്റൊരാളെ ശല്യപ്പെടുത്തുക എന്നിവയാണ് വ്യക്തികൾക്കെതിരെയുള്ള സൈബർ കുറ്റകൃത്യങ്ങൾ.

(i) സ്വകാര്യ വിവരങ്ങളുടെ മോഷണം (Identity theft): ഒരു വ്യക്തിയെ തിരിച്ചറിയാനുള്ള പേര്, ക്രെഡിറ്റ് കാർഡ്, ആധാർ നമ്പർ തുടങ്ങിയവ അയാളുടെ സമ്മതം കൂടാതെ മറ്റൊരാൾ തട്ടിപ്പ് നടത്തുന്നതിനും കുറ്റകൃത്യങ്ങൾക്കുമായി ഉപയോഗിക്കുന്നതാണ് സ്വകാര്യ വിവരങ്ങളുടെ മോഷണം എന്നറിയപ്പെടുന്നത്. ഒരാളുടെ ബാങ്ക് അക്കൗണ്ടുകളും സാമൂഹിക മാധ്യമ അക്കൗണ്ടുകളും ഉപയോഗിക്കുന്നതിനായി അയാളുടെ വ്യക്തിപര



മായ തിരിച്ചറിയൽ വിവരങ്ങൾ മോഷ്ടിക്കുന്ന പ്രവൃത്തിയാണിത്. ഇത് അയാളെ അപകീർത്തിപ്പെടുത്തുന്നതിനോ അയാളുടെ അക്കൗണ്ടിൽ നിന്ന് പണം മോഷ്ടിക്കുന്നതിനോ സാധനങ്ങൾക്ക് വില നൽകുന്നതിനോ, ധനകാര്യ സ്ഥാപനങ്ങളിൽനിന്ന് പണം ലഭിക്കുന്നതിനോ മറ്റെന്തെങ്കിലും ആനുകൂല്യങ്ങൾ ലഭിക്കുന്നതിനോ ആകാം.

ബാങ്ക് അക്കൗണ്ടിൽനിന്ന് പണം പിൻവലിക്കുന്നതിനു മുമ്പ് മോഷ്ടാവ് യഥാർഥത്തിലുള്ള ഉടമ പണമിടപാട് വിവരങ്ങൾ അറിയാതിരിക്കുന്നതിനായി മെയിലിങ് മേൽവിലാസം മാറ്റും. അയാളുടെ വ്യക്തിവിവരങ്ങൾ പുതിയ അക്കൗണ്ടുകളോ ക്രെഡിറ്റ് കാർഡ് അക്കൗണ്ടുകൾ തുടങ്ങുന്നതിനോ മൊബൈൽ ഫോൺ കണക്ഷൻ എടുക്കുന്നതിനോ മറ്റോ ഉപയോഗിച്ചേക്കാം.

(ii) ശല്യപ്പെടുത്തൽ (Harassment): വിശിഷ്ട വ്യക്തികളെപ്പറ്റി ചാറ്റ് റൂം, സാമൂഹിക മാധ്യമങ്ങൾ, ഇ-മെയിൽ തുടങ്ങിയവയിൽ അയാളുടെ ലിംഗഭേദം, വംശം, മതം, ദേശീയത എന്നിവയെ ഉന്നം വച്ച് പരിഹാസ്യമായ അഭിപ്രായങ്ങൾ ഇടുന്നത് ഇത്തരം ശല്യപ്പെടുത്തലാണ്. അശ്ലീല പദങ്ങളുപയോഗിക്കുക, അസ്സാന്മാർഗികമോ നിയമപരമല്ലാത്തതോ ആയ പ്രവൃത്തികളിലൂടെ ഭീഷണിപ്പെടുത്തുക എന്നിവയും ശല്യപ്പെടുത്തലിൽ ഉൾപ്പെടുന്നു.

ഇന്റർനെറ്റുപയോഗിച്ച് ഒരാളെ ശല്യപ്പെടുത്തുന്നതിന് സൈബർ സ്റ്റോക്കിങ് (stalking) എന്ന് പറയുന്നു. പ്രശസ്തരായ ചിലർ തങ്ങളുടെ പേരിൽ വന്നിട്ടുള്ള ഫേസ് ബുക്ക് അക്കൗണ്ടുകൾ വ്യാജമാണെന്ന് പറയുന്നത് നാം വായിച്ചിട്ടുണ്ട്. ഇത്തരം അക്കൗണ്ടുകളിൽ അപകീർത്തികരമായ ചിത്രങ്ങളോ വിവരങ്ങളോ വന്നത് കൊണ്ടാണിങ്ങനെ പറയുന്നത്. കുറ്റവാസനയുള്ള ചിലർ ബഹുമാന്യരായ വ്യക്തികളുടെ ഫോട്ടോയും വ്യക്തിവിവരങ്ങളും ഉപയോഗിച്ച് പ്രൊഫൈൽ നിർമ്മിക്കും. ചിലർ അപഹാസ്യമായ ഇ-മെയിലുകൾ, ഫേസ് ബുക്ക് പോസ്റ്റിങ്ങുകൾ എന്നിവ അയച്ച് ആളുകളെ അപകീർത്തിപ്പെടുത്തും. ഇവയെല്ലാം ഇത്തരം ശല്യപ്പെടുത്തലുകളാണ്. ദുരാരോപണങ്ങൾ, ഭീഷണികൾ, നിരീക്ഷിക്കലുകൾ, സ്വകാര്യവിവരങ്ങളുടെ മോഷണം, വിവരങ്ങൾ നശിപ്പിക്കൽ ഇവയെല്ലാം സൈബർ സ്റ്റോക്കിങ്ങിന്റെ പ്രത്യേകതകളാണ്. പ്രായപൂർത്തിയാകാത്ത കുട്ടികളെ ലൈംഗികമായി ചൂഷണം ചെയ്യുന്നതും സൈബർ സ്റ്റോക്കിങ്ങിൽ ഉൾപ്പെടും. ഇത് സൗഹൃദങ്ങളും തൊഴിലുകളും പ്രതിച്ഛായയും ആത്മവിശ്വാസവും നഷ്ടപ്പെടുത്തും.

(iii) ആൾമാറാട്ടവും വഞ്ചനയും (Impersonation and cheating): ഒരു വ്യക്തി മറ്റൊരാളായി അഭിനയിച്ച് ആളുകളെ ദ്രോഹിക്കുന്നതിനെ ആൾമാറാട്ടം എന്ന് പറയുന്നു. ഇന്റർനെറ്റിന്റെ അജ്ഞാതസ്വഭാവം ഉപയോഗിച്ച് ചിലർ ഓൺലൈൻ ആൾമാറാട്ടം നടത്തുന്നു.

വിദൂര രാജ്യങ്ങളിൽ നിന്നും ഇന്ത്യയിലേക്ക് വൻതുകകൾ മാറ്റുന്നതിന് നമ്മുടെ സഹായം ആവശ്യപ്പെട്ടു കൊണ്ടുള്ള ഇ-മെയിലുകൾ നമുക്ക് ലഭിക്കാറുണ്ട്. ഈ പണം ഏതെങ്കിലും വസ്തു, സ്വർണം എന്നിവ വിൽക്കാനുള്ള ആസ്തിയാണെന്ന് അവർ പറയും. നിയമനടപടികൾ പൂർത്തിയാക്കി ഈ ആസ്തി വിൽക്കുന്നതിന് അവർക്ക് കുറച്ചു പണം ആവശ്യമുണ്ടെന്നും അതിൽ ഒരു പങ്ക് നാം വഹിക്കേണ്ടതുണ്ട് എന്നും, പകരം ആസ്തിയുടെ 50% വരെ തരാമെന്നും വാഗ്ദാനം നൽകുന്നു. ഇതിനു വേണ്ടി നാം അയയ്ക്കുന്ന പണം സ്വീകരിച്ച്, ചില പ്രശ്നങ്ങൾ കൂടി തീർക്കാനുണ്ടെന്നു പറഞ്ഞ് കൂടുതൽ പണം ആവശ്യ

പ്പെടുന്നു. ഇതിന് ഇരയാകുന്നവർക്ക് വൻ തുകകൾ നഷ്ടപ്പെടുന്നു. ഇതുപോലെ വ്യത്യസ്ത കഥകളുമായി ധാരാളം മെയിലുകൾ വന്നു കൊണ്ടിരിക്കുന്നു. ഇത്തരം മെയിലുകൾ ഇന്റർനെറ്റ് വഞ്ചനയ്ക്ക് ഉദാഹരണങ്ങളാണ്.

സമാനമായ കുറ്റകൃത്യങ്ങൾ ഓൺലൈൻ ലേലത്തിലും സംഭവിക്കുന്നു. ചിലപ്പോൾ ഇന്റർ നെറ്റ് ലേലസൈറ്റുകളിൽ വിൽപ്പനയ്ക്കുള്ള ഉൽപ്പന്നങ്ങൾ അവതരിപ്പിക്കാറുണ്ട്. അവർ സാധനം തരുന്നതിന് മുമ്പ് തന്നെ പണം വാങ്ങുകയും സാധനം തരാതിരിക്കുകയും ചെയ്യുന്നു.

(iv) സ്വകാര്യത ലംഘനം (Violation of privacy): വ്യക്തമായ കാരണം കൂടാതെ മറ്റൊരാളുടെ വ്യക്തിജീവിതത്തിൽ കടന്നു കൂടുകയോ നുഴഞ്ഞു കയറുകയോ ചെയ്യുന്നതാണ് സ്വകാര്യത ലംഘനം. സ്വകാര്യത നഷ്ടപ്പെട്ട ആളിനോ സ്ഥാപനത്തിനോ നിയമനടപടി എടുക്കുന്നതിന് അവകാശമുണ്ട്. വ്യക്തിഗതവിവരങ്ങൾ, ഫോട്ടോ, തൊഴിലിട ദൃശ്യങ്ങൾ എന്നിവ പ്രചരിപ്പിക്കുന്നത് ഇതിൽ ഉൾപ്പെടുന്നു. സ്ത്രീകളുടെ ചിത്രങ്ങൾ എടുക്കുന്നതിന് പൊതു സ്ഥലങ്ങളിൽ ഒളി ക്യാമറ, മൊബൈൽ ക്യാമറ എന്നിവ ഉപയോഗിക്കുന്നതിനെ ക്കുറിച്ച് നാം കേട്ടിട്ടുണ്ട്. അനുവാദമില്ലാതെ ഒരാളുടെ ഫോട്ടോ എടുക്കുന്നത് സ്വകാര്യതാ ലംഘനമാണ്. അനുവാദമില്ലാതെ മറ്റുള്ളവരുടെ ദൃശ്യങ്ങൾ പൊതു മാധ്യമങ്ങളിൽ പോസ്റ്റ് ചെയ്യുന്നതും അവ പകർത്തുന്നതും ഇ-മെയിലിൽ പ്രചരിപ്പിയ്ക്കുന്നതും സ്വകാര്യതാ ലംഘനമാണ്.

(v) അശ്ലീല വസ്തുക്കളുടെ വ്യാപനം (Dissemination of obscene material): അശ്ലീലത പോലുള്ള കുറ്റകൃത്യങ്ങൾ വ്യാപിക്കുന്നതിന് ഇന്റർനെറ്റ് ഒരു മാധ്യമമായിരിക്കുന്നു. അശ്ലീല വസ്തു ക്കൾ പോസ്റ്റ് ചെയ്യുകയോ വിതരണം ചെയ്യുകയോ ചെയ്യുന്നത് സൈബർ കുറ്റകൃത്യമാണ്. ഇന്റർനെറ്റിലെ അശ്ലീലത പല രൂപത്തിലുണ്ട്. നിരോധിച്ചിട്ടുള്ള സൈറ്റുകൾ എടുക്കുക, കമ്പ്യൂട്ടർ ഉപയോഗിച്ച് ഇവ നിർമ്മിക്കുക, ഇന്റർനെറ്റിലൂടെ ഇവ ഡൗൺലോഡ് ചെയ്യുക എന്നിവയെല്ലാം ഇതിൽ ഉൾപ്പെടുന്നു. ഇത് പലപ്പോഴും പ്രായപൂർത്തിയാകാത്ത കുട്ടികളെ തെറ്റിലേക്ക് നയിക്കുന്നു.

ഇരകളുടെ അറിവില്ലാതെയാണ് മിക്കവാറും സൈബർ കുറ്റങ്ങളും നടക്കുന്നത്. എന്നാൽ എളുപ്പത്തിൽ പണമുണ്ടാക്കുന്നതിന് പലപ്പോഴും ഇരകൾ അറിഞ്ഞുകൊണ്ടുതന്നെ ഇതിൽ പങ്കെടുക്കുന്നുമുണ്ട്. ഇത്തരത്തിലുള്ള കുറ്റകൃത്യങ്ങൾ തടയാൻ വിവിധ സംവിധാ നങ്ങൾ ഉണ്ടെന്ന് നാം അറിഞ്ഞിരിക്കണം. ഇ-മെയിൽ സേവനങ്ങൾ നൽകുന്നവർ 'സ്പാം ഫിൽറ്റർ' എന്ന ഒരു സംവിധാനം, ആവശ്യമില്ലാത്ത മെയിലുകൾ തടയാൻ വേണ്ടി ഉപയോ ഗിക്കുന്നു. ഒരു പരിധിവരെ തട്ടിപ്പുകൾ തടയാൻ ഓൺലൈൻ സാമ്പത്തിക ഇടപാടു കൾക്ക് പല തരത്തിലുള്ള ആധികാരികതാ സംവിധാനങ്ങളുണ്ട്. എന്നിരുന്നാലും ഇന്റർ നെറ്റ് ഉപയോഗിക്കുമ്പോൾ നാം അതീവ ശ്രദ്ധ പാലിക്കേണ്ടതുണ്ട്.

B. സ്വത്തിനെതിരായ സൈബർ കുറ്റകൃത്യങ്ങൾ (Cyber crimes against property)

ക്രെഡിറ്റ് കാർഡ്, ബൗദ്ധിക സ്വത്ത് എന്നിങ്ങനെയുള്ള സ്വത്തുക്കൾക്കെതിരെ നടത്തുന്ന സൈബർ കുറ്റകൃത്യങ്ങളെയാണ് സ്വത്തിനെതിരായ സൈബർ കുറ്റകൃത്യങ്ങൾ എന്ന് പറയുന്നത്. ഹാക്കിങ്, സോഫ്റ്റ്‌വെയർ മോഷണം, വസ്തുവകകൾ മനപ്പൂർവ്വം നശിപ്പി ക്കൽ, സൈബർ ഇടത്തിൽ നുഴഞ്ഞു കയറ്റം, കമ്പ്യൂട്ടറിൽ ശേഖരിച്ച് വച്ചിരിക്കുന്ന വിവര

ങ്ങൾ അനധികൃതമായി കൈവശപ്പെടുത്തൽ എന്നിവയെല്ലാം തന്നെ സ്വത്തിനെതിരായ സൈബർ കുറ്റകൃത്യങ്ങൾ ആണ്.

സ്വത്തിനെതിരായ സൈബർ കുറ്റകൃത്യങ്ങളിൽ ചിലത് താഴെ കൊടുത്തിരിക്കുന്നു.

(i) ക്രെഡിറ്റ് കാർഡ് തട്ടിപ്പ് (Credit card fraud): അനുവാദമില്ലാതെ മറ്റൊരാളുടെ ക്രെഡിറ്റ് കാർഡ് വിവരങ്ങളുപയോഗിച്ച് പണം എടുക്കുക, വാങ്ങിക്കുന്ന സാധനങ്ങൾക്ക് പണം നൽകുക എന്നിവയെല്ലാം ക്രെഡിറ്റ് കാർഡ് തട്ടിപ്പാണ്. ഇന്റർനെറ്റ് മോഷ്ടാക്കൾ വലിയ സ്ഥാപനങ്ങളുടെ വെബ്സൈറ്റ് ഹാക്ക് ചെയ്ത് വളരെയധികം ആളുകളുടെ ക്രെഡിറ്റ് കാർഡ് വിവരങ്ങൾ മോഷ്ടിച്ച സംഭവങ്ങളുണ്ട്. മോഷ്ടാക്കൾ ഈ വിവരങ്ങൾ ആർക്കെങ്കിലും പണം നൽകുന്നതിനുപയോഗിക്കുന്നു, ചിലപ്പോൾ മറ്റ് മോഷ്ടാക്കൾക്ക് കുറഞ്ഞ വിലയ്ക്ക് ഇവ വിൽക്കുകയും ചെയ്യുന്നു.

(ii) ബൗദ്ധിക സ്വത്ത് മോഷണം (Intellectual property theft): ബൗദ്ധിക സ്വത്തവകാശ ലംഘനം ഈ ഇനത്തിൽപ്പെട്ടതാണ്. പകർപ്പവകാശം, കുത്തകാവകാശം, വ്യാപാരമുദ്ര എന്നിവയുടെ ലംഘനങ്ങൾ സ്വത്തിനെതിരായ കടന്നു കയറ്റമാണ്. ഈയിടയ്ക്ക് ഒരു ഇന്ത്യൻ ഐ.ടി കമ്പനി, പ്രോഗ്രാം കോഡിലെ തെറ്റുകൾ തിരുത്താനുള്ള ഒരു സോഫ്റ്റ്വെയർ നിർമ്മിച്ചു. ഒരു തൊഴിലാളി ഇത് സി.ഡിയിൽ പകർത്തി കമ്പനിയുടെ എതിരാളികൾക്ക് വലിയ വിലയ്ക്ക് വിൽക്കാൻ ശ്രമിച്ചു. ഇത് കമ്പനിക്ക് വലിയ സാമ്പത്തിക നഷ്ടവും സ്വത്ത് നഷ്ടവും ഉണ്ടാക്കി. ഇതും ബൗദ്ധിക സ്വത്ത് മോഷണമായി പരിഗണിക്കപ്പെടുന്നു. സോഫ്റ്റ്വെയർ മോഷണവും സൈബർ നിയമപ്രകാരം കുറ്റകരമാണ്.

ഇക്കാലത്ത് ബൗദ്ധിക സ്വത്ത് മോഷണം സർവ്വസാധാരണമാണ്. ഏതു വിഷയത്തിലും നമുക്ക് സ്വതന്ത്രമായി വിവരങ്ങൾ ഇന്റർനെറ്റിൽ നിന്നും എടുക്കാം. മറ്റൊരാളുടെ ഭാഷ, ചിന്തകൾ, ആശയങ്ങൾ ഇവ നമ്മുടെ സ്വന്തം കൃതികളിൽ ഉപയോഗിക്കുന്നതിന് 'പ്ലേജറിസം' (Plagiarism) അഥവാ സാഹിത്യ ചോരണം എന്ന് പറയുന്നു. ഇന്ത്യയിൽ ബൗദ്ധിക സ്വത്തവകാശം സംരക്ഷിക്കുന്നതിന് ശക്തമായ പകർപ്പവകാശ നിയമങ്ങൾ ലഭ്യമാണ്. ഓൺലൈനിൽ ലഭ്യമായിട്ടുള്ള വിവിധ ടൂളുകൾ ഉപയോഗിച്ച് എളുപ്പത്തിൽ പ്ലേജറിസം കണ്ടുപിടിക്കാൻ സാധിക്കും .

(iii) ഇന്റർനെറ്റ് സമയമോഷണം (Internet time theft): ഇന്ന് ഒട്ടു മിക്ക മോഡത്തിനും റൗട്ടറിനും വയർലെസ് ഇന്റർനെറ്റ് സൗകര്യമുണ്ട്. ഇത് വീടുകൾ, സ്കൂളുകൾ, വ്യാപാര സ്ഥാപനങ്ങൾ എന്നിവിടങ്ങളിലൊക്കെ ഇന്റർനെറ്റ് പങ്കു വയ്ക്കൽ സൗകര്യമൊരുക്കുന്നു. ഇവയൊക്കെ ശരിയായ വിധത്തിൽ പാസ്വേഡ് ഉപയോഗിച്ച് സുരക്ഷിതമാക്കിയില്ലെങ്കിൽ മറ്റുള്ളവർ നമ്മുടെ ഇന്റർനെറ്റ് ഉപയോഗിക്കും. ഒരാളുടെ ഇന്റർനെറ്റ് സൗകര്യം മറ്റൊരാൾ അനധികൃതമായി ഉപയോഗിക്കുന്നതിനെയാണ് ഇന്റർനെറ്റ് സമയ മോഷണം എന്നു പറയുന്നത്. ഇത് യഥാർഥ ഇന്റർനെറ്റ് ഉപയോക്താവിന് പണവും ഇന്റർനെറ്റ്

സമയവും നഷ്ടപ്പെടുത്തുന്നു. ഇതിനും പുറമെ മറ്റുള്ളവർ നമ്മുടെ ഇന്റർനെറ്റ് അക്കൗണ്ടുപയോഗിച്ച് കുറ്റകൃത്യങ്ങൾ ചെയ്യുകയും നാം അതിന് ഉത്തരവാദികൾ ആകുകയും ചെയ്യുന്നു.

വിവിധ തരത്തിലുള്ള ആക്രമണങ്ങളായ വൈറസ്, വേംസ്, മാൻ ഇൻ ദി മിഡിൽ അറ്റാക്ക് എന്നിവ പതിനൊന്നാം ക്ളാസ്സിലെ 9-ാം അധ്യായത്തിൽ നാം പഠിച്ചു കഴിഞ്ഞു. ഇതെല്ലാം ഇത്തരത്തിലുള്ള മോഷണത്തിന് ഉദാഹരണങ്ങളാണ്.

C. ഭരണകൂടത്തിനെതിരായ സൈബർ കുറ്റകൃത്യങ്ങൾ (Cyber crimes against government)

ഇ-ഭരണത്തിന്റെ പ്രശസ്തി ഭരണകൂടങ്ങളെ സൈബർ കുറ്റകൃത്യങ്ങൾക്ക് എളുപ്പത്തിൽ വിധേയരാക്കി. ഇങ്ങനെ വിവിധതരത്തിലുള്ള സർക്കാർ കമ്പ്യൂട്ടർ നെറ്റ്‌വർക്കുകളും, വെബ് സൈറ്റുകളും സൈബർ കുറ്റകൃത്യങ്ങളുടെ അപകട ഭീഷണി നേരിടുന്നു. സൈബർ ഭീകരത, വെബ്സൈറ്റ് വികൃതമാക്കൽ, ഇ- ഭരണ വെബ്സൈറ്റുകൾക്കെതിരെയുള്ള ആക്രമണം എന്നിവ ഇത്തരം കുറ്റകൃത്യങ്ങളുടെ വിവിധ ഭാവങ്ങളാണ്.

(i) സൈബർ ഭീകരത (Cyber terrorism)

ന്യൂക്ലിയർ പവർ പ്ലാന്റുകൾ, വ്യോമഗതാഗതനിയന്ത്രണം, ഗ്യാസ് ലൈൻ നിയന്ത്രണം, ടെലികോം തുടങ്ങിയ സൂഷ്മസംവേദനക്ഷമതയുള്ള നെറ്റ്‌വർക്കുകൾ എന്നിവയിലെ സൈബർ ആക്രമണങ്ങളെ ഈ വിഭാഗത്തിൽപെടുത്താം. ലോക വ്യാപകമായി ഭരണകൂടങ്ങൾക്കെതിരായി ഈ വിധത്തിലുള്ള ആക്രമണങ്ങൾ വർധിച്ചു വരുന്നു. രാജ്യത്തിന്റെ സാമ്പത്തിക സാങ്കേതിക അടിത്തറയെ ബാധിക്കുന്ന വിധത്തിൽ ദേശവിരുദ്ധ ശക്തികൾ ഇന്റർനെറ്റ് ഉപയോഗിക്കുന്നതാണ് സൈബർ ഭീകരത.

2010 ൽ ഇറാന്റെ രഹസ്യ ന്യൂക്ലിയർ പദ്ധതിയെ 'ടക്സ്നെറ്റ്' (tuxnet) എന്ന് പേരുള്ള വൈറസ് ഉപയോഗിച്ച് ആക്രമിച്ചു. യുറേനിയം സമ്പുഷ്ടീകരണ പരിപാടിയെ പ്രവർത്തന രഹിതമാക്കാൻ ഉദ്ദേശിച്ചിട്ടുള്ളതായിരുന്നു ഈ വൈറസ്. ഈ വൈറസ് വളരെയധികം ന്യൂക്ലിയർ കൺട്രോളുകളെ ബാധിച്ച് അവയ്ക്ക് തെറ്റായ നിർദേശങ്ങൾ നൽകി. ന്യൂക്ലിയർ പ്രവർത്തനങ്ങൾ തെറ്റായ രീതിയിൽ നടത്തി സജ്ജീകരണങ്ങൾ കേടു വരുത്തി. സൈബർ ഭീകരതക്കെതിരെ ഒരു രാജ്യത്തിന് വലിയ വില നൽകേണ്ടി വരും. അതിനാൽ സെർവറുകൾക്ക് വളരെ ശക്തിയേറിയ സുരക്ഷാ സംവിധാനങ്ങൾ ഒരുക്കേണ്ടത് സർക്കാരിന്റെ ബാധ്യതയാണ്.

(ii) വെബ്സൈറ്റ് വികൃതമാക്കൽ (Website defacement)

ഇത് ഭരണകൂടത്തിനെതിരെയുള്ള സാധാരണ സൈബർ ആക്രമണമാണ്. ഇതിൽ സർക്കാർ വെബ് സൈറ്റുകൾ ഹാക്ക് ചെയ്യുകയും അവയിൽ സർക്കാരിനെതിരെ ദോഷകരമായ അഭിപ്രായങ്ങൾ ഇടുകയും ചെയ്യുന്നു.

(iii) ഇ-ഭരണ വെബ് സൈറ്റുകൾക്കെതിരെയുള്ള ആക്രമണങ്ങൾ (Attacks against e-governance websites)

ഇത്തരം ആക്രമണങ്ങൾ ഒരു പ്രത്യേകതരം ഓൺലൈൻ സർക്കാർ സേവനം ലഭ്യമല്ലാതാക്കുന്നു. പതിനൊന്നാം ക്ലാസ്സിലെ 9-ാം അധ്യായത്തിൽ നാം പഠിച്ച 'ഡിസ്ട്രിബ്യൂട്ടഡ് ഡിനയൽ ഓഫ് സർവീസ്' (DDOS) ഉപയോഗിച്ചാണ് ഇത് ചെയ്യുന്നത്. ഹാക്കർമാർ വെബ്സൈറ്റ് തട്ടിയെടുത്ത് ഉപയോഗിക്കുന്നത് ഇതിന്റെ മറ്റൊരു തലമാണ്. അവർക്ക് ഉള്ളടക്ക നിയന്ത്രണത്തിലൂടെ (Content Management System) വെബ്സൈറ്റ് നിയന്ത്രണം സാധിക്കുകയും വിവരങ്ങൾ നശിപ്പിക്കാൻ സാധിക്കുകയും ചെയ്യുന്നു. ഇത് മൂലം സർക്കാരിന് വലിയ നഷ്ടം സംഭവിക്കുന്നു.

സൈബർ ഇടത്തിന്റെ അജ്ഞാത സ്വഭാവം കാരണം ആധുനിക ലോകത്ത് സൈബർ ഭീകരത വർധിച്ചിട്ടുണ്ട്. വളരെയധികം ആളുകളെ ബാധിക്കുന്ന വിവിധ ലക്ഷ്യ സ്ഥാനങ്ങളുടെ ലഭ്യതയും, ദൂരെ നിന്ന് ആക്രമണങ്ങൾ നടത്താമെന്നുള്ളതും സൈബർ ഇടത്തിലൂടെയുള്ള ഭീകര പ്രവർത്തനങ്ങൾ വർധിപ്പിക്കുന്നു.

12.2.5 സൈബർ സദാചാരം (Cyber ethics)

ആഗോളതലത്തിൽ സൈബർ കുറ്റകൃത്യങ്ങൾ വർധിച്ചു വരുന്നതായി കണക്കുകൾ സൂചിപ്പിക്കുന്നു. ഇന്റർനെറ്റ് കുറ്റകൃത്യങ്ങൾ പലതരമാണ്. അവ വിവിധ രീതികളിൽ ചെയ്യുന്നു. ആദ്യഘട്ടം നമ്മുടെ കുടുംബത്തെയും വ്യാപാരത്തെയും ഇത്തരം കുറ്റകൃത്യങ്ങളിൽ നിന്ന് സംരക്ഷിക്കുക എന്നതാണ്. അതുപോലെ തന്നെ സൈബർ ഇടത്തിലെ നമ്മുടെ പ്രവർത്തനങ്ങൾ മറ്റുള്ളവർക്ക് ദോഷം വരുത്തില്ല എന്ന് ഉറപ്പു വരുത്തേണ്ടതുണ്ട്. നമ്മുടെ പ്രവർത്തനങ്ങൾ മറ്റു പലരും നിരീക്ഷിക്കുന്നുണ്ടെന്ന് നാം ഓർക്കണം.

ഇന്റർനെറ്റ് ഉപയോഗിക്കുന്നവർക്കുള്ള സൈബർ സദാചാര നിർദേശങ്ങൾ താഴെ കൊടുക്കുന്നു.

- നിങ്ങളുടെ കമ്പ്യൂട്ടറിൽ ആന്റിവൈറസുകൾ, ഫയർവാൾ, സ്പാം ബ്ലോക്കിങ് സോഫ്റ്റ്‌വെയർ എന്നിവ ഉപയോഗിക്കുക.
- ഓൺലൈൻ പണമിടപാടുകൾ നടത്തുമ്പോൾ സുരക്ഷാ വെബ്സൈറ്റുകൾ ഉണ്ടെന്ന് ഉറപ്പു വരുത്തുക
- അജ്ഞാതർ അയയ്ക്കുന്ന ഇ-മെയിലുകളോട് പ്രതികരിക്കാതിരിക്കുക
- അസാധാരണവും സങ്കീർണവുമായ പാസ്‌വേർഡുകൾ ഉപയോഗിക്കുക. നിശ്ചിത ഇടവേളകളിൽ പാസ്‌വേർഡുകൾ മാറ്റുക. പാസ്‌വേഡിൽ കുറഞ്ഞത് 8 അക്ഷരങ്ങൾ ഉണ്ടായിരിക്കണം. അക്ഷരങ്ങളും, സംഖ്യകളും, ചിഹ്നങ്ങളും ഉപയോഗിക്കണം.
- ഒരു കരാറിന്റെയോ സന്ദേശത്തിന്റെയോ നേരെയുള്ള ചെക്കുബോക്സുകളോ OK ബട്ടണുകളോ അവ വായിച്ചു തീരുന്നതിന് മുമ്പ് ക്ലിക്ക് ചെയ്യരുത്.
- ആധികാരികതയില്ലാത്ത സോഫ്റ്റ്‌വെയർ ഉപയോഗിക്കരുത്.
- നിങ്ങളുടെ തിരിച്ചറിയൽ രേഖ മറച്ചുവച്ച് മറ്റുള്ളവരെ കബളിപ്പിക്കരുത്.

- സാമൂഹിക മാധ്യമങ്ങളിലോ ഇ-മെയിലിലോ മോശമായതോ പരുക്കനായതോ ആയ ഭാഷ ഉപയോഗിക്കരുത്.
- നിങ്ങളുടെ സ്വന്തം കമ്പ്യൂട്ടറുകൾ അല്ലാത്തവ ഉപയോഗിക്കുമ്പോൾ അക്കൗണ്ടിൽ പ്രവേശിക്കുന്നതിന് മുമ്പ് (login), 'Remember me' എന്ന ചെക്ക് മാർക്ക് നീക്കം ചെയ്യുക.

നമുക്ക് ചെയ്യാം

- വിവിധ തരം സൈബർ കുറ്റകൃത്യങ്ങളെപ്പറ്റി ബോധവൽക്കരണ പരിപാടികൾ സംഘടിപ്പിക്കുക.
- സൈബർ ഇടത്തിൽ എങ്ങനെ നമ്മെ സംരക്ഷിക്കാം എന്നതിനെ കുറിച്ച് പട്ടിക തയാറാക്കുക.

നിങ്ങളുടെ പുരോഗതി അറിയുക

1. സൈബർ ഇടം എന്നാലെന്ത്?
2. കമ്പ്യൂട്ടർ, മൊബൈൽ ഫോൺ, ഇന്റർനെറ്റ് എന്നിവ ഉപയോഗിച്ചുള്ള കുറ്റകൃത്യങ്ങളെ ______________ എന്ന് വിളിക്കുന്നു.
3. വ്യക്തികൾക്കെതിരെയുള്ള സൈബർ കുറ്റകൃത്യം എന്നാൽ എന്താണ്?
4. സൈബർ ഭീകരത എന്നത് ______________ ന് എതിരായ ഒരു സൈബർ കുറ്റകൃത്യമാണ്.
5. ഒരാളുടെ യൂസർ നെയിം, പാസ്‌വേർഡ് തുടങ്ങിയവ മോഷ്ടിക്കുന്നത് ______________ ആണ്.

12.2.6 സൈബർ നിയമങ്ങൾ (Cyber laws)

ഇന്റർനെറ്റിന്റെ നിയമപരവും നിയന്ത്രണപരവുമായ കാര്യങ്ങളെ സൂചിപ്പിക്കുന്നതിനാണ് സൈബർ നിയമങ്ങൾ എന്ന പേര് ഉപയോഗിക്കുന്നത്. കമ്പ്യൂട്ടറുകളുടെയും ഇന്റർനെറ്റിന്റെയും നിയന്ത്രണത്തെ സംബന്ധിക്കുന്ന നിയമങ്ങൾ എന്ന് സൈബർ നിയമത്തെ നിർവചിക്കാം.

വിവിധ തരം സൈബർ കുറ്റകൃത്യങ്ങൾ ഉണ്ടെന്ന് നാം കണ്ടു. മോഷണം, വഞ്ചന, കൃത്രിമ രേഖയുണ്ടാക്കൽ, അപകീർത്തിപ്പെടുത്തൽ എന്നിവയെല്ലാം പരമ്പരാഗത സ്വഭാവമുള്ള കുറ്റകൃത്യങ്ങളും ഇന്ത്യൻ ശിക്ഷാ നിയമങ്ങൾക്ക് (Indian Penal Code) വിധേയവുമാണ്. വിവര സാങ്കേതികവിദ്യ നിയമം 2000 വും ഐ.ടി. ആക്ട് ഭേദഗതി ബിൽ 2008 ഉം ഇന്ത്യയിൽ വ്യാപകമായി വരുന്ന സൈബർ കുറ്റങ്ങളെ നേരിടാനുള്ളതാണ്. ഇന്റർനെറ്റ് ഉപയോഗിച്ചുള്ള എല്ലാ പ്രവർത്തനങ്ങളെയും ഇടപാടുകളെയും സ്പർശിക്കുന്നതാകയാൽ സൈബർ നിയമം പ്രാധാന്യമുള്ളതാണ്. നാം മനസ്സിലാക്കിയാലും ഇല്ലെങ്കിലും സൈബർ ഇടത്തിലുള്ള എല്ലാ പ്രവർത്തനത്തിനും പ്രതിപ്രവർത്തനത്തിനും നിയമപരമായ വശങ്ങൾ ഉണ്ട്.

12.2.7 വിവര സാങ്കേതികവിദ്യ നിയമം 2000 (2008 ൽ ഭേദഗതി വരുത്തിയത്) (Information Technology Act 2000 [Amended in 2008])

കമ്പ്യൂട്ടറുകൾ, സെർവറുകൾ, കമ്പ്യൂട്ടർ നെറ്റ്‌വർക്കുകൾ, ഇലക്ട്രോണിക് രൂപത്തിലുള്ള ഡാറ്റയും വിവരങ്ങളും എന്നിവയുടെ ഉപയോഗത്തെ നിയന്ത്രിക്കുന്ന ഭാരത സർക്കാരിന്റെ നിയമമാണ് വിവരസാങ്കേതികവിദ്യ നിയമം 2000. നെറ്റ്‌വർക്ക് സേവനദാതാവിന്റെ ബാധ്യതകൾ, ആധികാരികത വരുത്തൽ, ഡിജിറ്റൽ ഒപ്പ്, സൈബർ കുറ്റകൃത്യങ്ങൾ എന്നിവയുമായി ബന്ധപ്പെട്ട വിവിധ വശങ്ങളെ ഈ നിയമം സ്പർശിക്കുന്നു. ഇലക്ട്രോണിക് ഡാറ്റ കൈമാറ്റത്തിലൂടെയുള്ള ഇടപാടുകൾക്കും മറ്റ് ഇലക്ട്രോണിക് ആശയ വിനിമയ രീതികൾക്കും ഈ നിയമം അംഗീകാരം നൽകുന്നു.

പേപ്പർ ഉപയോഗിച്ചുള്ള ആശയ വിനിമയത്തിന് പകരമുള്ള മാർഗങ്ങൾ നൽകുകയും സർക്കാർ ഏജൻസികളുടെ രേഖകൾ ഇലക്ട്രോണിക് ആയി ഫയൽ ചെയ്യാനും ഐ.ടി നിയമം സഹായിക്കുന്നു. കൂടാതെ ഇലക്ട്രോണിക് ആശയ വിനിമയത്തിന് നിയമ പ്രാബല്യം നൽകുന്നു. ഇത് സൈബർ ഇടത്തിലുള്ള കുറ്റകൃത്യങ്ങൾ, തർക്കങ്ങൾ എന്നിവ നേരിടുന്നതിനും സൈബർ കുറ്റകൃത്യങ്ങളുടെ ഇരകൾക്ക് നീതി ലഭ്യമാക്കുവാൻ സഹായകമാവുകയും ചെയ്യുന്നു. ഇന്ത്യയിലെ ഇ-വാണിജ്യത്തിനുള്ള നിയമപരമായ അടിത്തറ ഒരുക്കുന്നതിന് ഐ.ടി. നിയമം ലക്ഷ്യമിടുന്നു. ഐ.ടി. വ്യവസായത്തെ പ്രോത്സാഹിപ്പിക്കുന്നതിനും ഇ-വാണിജ്യം നിയന്ത്രിക്കുന്നതിനും ഇ-ഭരണത്തിന് സൗകര്യമൊരുക്കുന്നതിനും സൈബർ കുറ്റങ്ങൾ തടയുന്നതിനുമാണ് ഈ നിയമം ഉണ്ടാക്കിയിട്ടുള്ളത്. ആഗോള തലത്തിൽ രാജ്യത്തെ സേവിക്കുന്നതിന് ഇന്ത്യക്കകത്തു തന്നെയുള്ള സുരക്ഷാ ക്രമീകരണ രീതികളെ ഇത് പ്രോത്സാഹിപ്പിക്കുന്നു. ഈ നിയമമനുസരിച്ച് ചില നിയമ ലംഘനങ്ങൾ ഗൗരവമുള്ള കുറ്റങ്ങളായി കണക്കാക്കി കുറ്റവാളികളെ നിയമ വിചാരണയ്ക്ക് വിധേയമാക്കുകയും ചെയ്യുന്നു. അതിനാൽ ഐ.ടി. നിയമം 2000 ന്റെ പ്രധാനപ്പെട്ട ആശയങ്ങൾ മനസ്സിലാക്കേണ്ടതാണ്.

2000 മെയ് മാസത്തിലാണ് ഇന്ത്യൻ പാർലമെന്റിൽ വിവര സാങ്കേതികവിദ്യ ബിൽ പാസ്സാക്കിയത്. ഇത് വിവര സാങ്കേതികവിദ്യ നിയമം 2000 എന്നറിയപ്പെടുന്നു. പിന്നീട് ഈ നിയമം 'ഐ.ടി. ഭേദഗതി ബിൽ 2008' എന്ന പേരിൽ ഭേദഗതി ചെയ്ത് 2008 ഡിസംബറിൽ പാസ്സാക്കി. വിവര സാങ്കേതികവിദ്യ നിയമം 2000 നു ശേഷമുള്ള ഐ.ടി പുരോഗതിയും സുരക്ഷാ കാര്യങ്ങളും ഇതിൽ ഉൾപ്പെടുത്തിയിട്ടുണ്ട്. വൈറസ് കൊണ്ടോ, സേവനം നിഷേധിക്കുന്നത് കൊണ്ടോ, കമ്പ്യൂട്ടറുകൾക്കോ, കമ്പ്യൂട്ടർ നെറ്റ്‌വർക്കുകൾക്കോ ഉണ്ടാകുന്ന കേടുപാടുകൾക്ക് നഷ്ടപരിഹാരം നൽകേണ്ടതാണ്. ഈ നിയമത്തിലെ 65-74 ഭാഗങ്ങൾ സൈബർ കുറ്റങ്ങളെപ്പറ്റി പ്രത്യേകം പ്രതിപാദിക്കുന്നു.

ഐ.ടി നിയമം 2000 ലും ഐ.ടി ഭേദഗതി 2008 ലും സൂചിപ്പിച്ചിട്ടുള്ള ചില സാധാരണ കുറ്റങ്ങളും അവയുടെ ശിക്ഷകളും

ഭാഗം	കുറ്റം	ശിക്ഷ
65	കമ്പ്യൂട്ടറിന്റെ സോഴ്സ് കോഡിൽ നാശം വരുത്തുക	2 ലക്ഷം രൂപ വരെ പിഴയോ, 3 വർഷം തടവോ രണ്ടും കൂടിയോ
66	കമ്പ്യൂട്ടറുമായി ബന്ധപ്പെട്ട കുറ്റങ്ങൾ	5 ലക്ഷം രൂപ പിഴയോ, 3 വർഷം വരെ തടവോ രണ്ടും കൂടിയോ
66B	മോഷ്ടിച്ച കമ്പ്യൂട്ടർ, ആശയവിനിമയ സംവിധാനം ഇവ വാങ്ങുക	1 ലക്ഷം രൂപ പിഴയോ, 3 വർഷം വരെ തടവോ രണ്ടും കൂടിയോ
66C	ഐഡന്റിറ്റി മോഷണം നടത്തുക	1 ലക്ഷം രൂപ പിഴയോ, 3 വർഷം വരെ തടവോ
66D	കമ്പ്യൂട്ടർ വിഭവങ്ങൾ ഉപയോഗിച്ച് ആൾ മാറാട്ടം നടത്തി വഞ്ചിക്കുക	1 ലക്ഷം രൂപ പിഴയോ, 3 വർഷം വരെ തടവോ രണ്ടും കൂടിയോ
66E	സ്വകാര്യത ലംഘനം	2 ലക്ഷം രൂപ പിഴയോ, 3 വർഷം വരെ തടവോ രണ്ടും കൂടിയോ
66F	സൈബർ ഭീകരത	ജീവിതാവസാനം വരെ തടവ്
67	ഇലക്ട്രോണിക് രീതിയിൽ അശ്ലീലം പ്രചരിപ്പിക്കൽ	2 മുതൽ 3 വർഷം വരെ തടവും 5 ലക്ഷം രൂപ വരെ പിഴയും. കുറ്റം ആവർത്തിച്ചാൽ 5 വർഷം വരെ തടവും 10 ലക്ഷം രൂപ വരെ പിഴയും
67A	ഇലക്ട്രോണിക് രീതിയിൽ ലൈംഗിക അതിക്രമം പ്രചരിപ്പിക്കൽ	10 ലക്ഷം രൂപ വരെ പിഴയും, 5 വർഷം വരെ തടവും
67B	ഇലക്ട്രോണിക് രീതിയിൽ കുട്ടികൾക്കെതിരെയുള്ള ലൈംഗിക അതിക്രമം പ്രചരിപ്പിക്കൽ	5 വർഷം വരെ തടവും 10 ലക്ഷം രൂപ വരെ പിഴയും. കുറ്റം ആവർത്തിച്ചാൽ 7 വർഷം വരെ തടവും 10 ലക്ഷം രൂപ വരെ പിഴയും

ഓൺലൈൻ ആശയവിനിമയ സംവിധാനങ്ങളായ ഇ-മെയിൽ, സാമൂഹിക മാധ്യമം, എസ്.എം.എസ്. മുതലായ വയിലൂടെ അശ്ലീല സന്ദേശങ്ങൾ അയച്ചാലുള്ള ശിക്ഷാനടപടികൾ ഇന്ത്യൻ പീനൽ കോഡിലെ (IPC) ബന്ധപ്പെട്ട വിഭാഗങ്ങൾക്കനുസരിച്ചായിരിക്കും.

12.2.8 സൈബർ കുറ്റാന്വേഷണം (Cyber forensics)

സൈബർ കുറ്റങ്ങളുടെ വർധന നിത്യജീവിതത്തെയും രാജ്യ സുരക്ഷയെയും ബാധിക്കുന്നു. വളരെയധികം സൗകര്യങ്ങൾ ഇന്റർനെറ്റ് വഴി ലഭ്യമാണെങ്കിലും അത് കുറ്റവാളികൾക്ക് കൂടുതൽ കുറ്റങ്ങൾ ചെയ്യുന്നതിന് സഹായകമാകുന്നു. സൈബർ കുറ്റങ്ങൾ കണ്ടുപിടിച്ച് അന്വേഷണം നടത്തി ശിക്ഷിക്കുവാൻ പരമ്പരാഗതമായി ഉണ്ടായിരുന്ന നിയമങ്ങൾ ഫലപ്രദമായിരുന്നില്ല.

ശാസ്ത്രീയമായ അറിവുപയോഗിച്ച് കുറ്റകൃത്യങ്ങൾ കണ്ടുപിടിക്കുകയും, അവ തെളിവുകൾ സഹിതം ശേഖരിക്കുകയും, വിശകലനം ചെയ്ത് കോടതിക്ക് സമർപ്പിക്കുകയും

ചെയ്യുന്ന പ്രക്രിയയാണ് ഫോറൻസിക്സ്. നിയമവും കമ്പ്യൂട്ടർ സയൻസും യോജിപ്പിച്ച് കമ്പ്യൂട്ടർ സിസ്റ്റങ്ങൾ, നെറ്റ്‌വർക്കുകൾ, ആശയവിനിമയ സംവിധാനങ്ങൾ ഇവയിൽ നിന്നുള്ള വിവരങ്ങൾ വിശകലനം നടത്തി കോടതിക്ക് തെളിവ് കൊടുക്കുന്ന ശാസ്ത്രശാഖയാണ് സൈബർ ഫോറൻസിക്സ്. ശേഖരിച്ച തെളിവുകളുടെ സത്യസന്ധതയെ സംരക്ഷിക്കുന്ന വിധത്തിൽ ഡാറ്റ വിശകലനം ചെയ്ത് നിയമാനുസരണം ഫലപ്രദമായി ഉപയോഗിക്കലാണ് കമ്പ്യൂട്ടർ ഫോറൻസിക്സിന്റെ ലക്ഷ്യം.

12.4.9 ഇൻഫോമാനിയ (Infomania)

വിവരങ്ങൾ വിജയത്തിന്റെ താക്കോലാണ്. അവ ശേഖരിച്ച് പ്രൊസസ്സ് ചെയ്തു വേണം ഉപയോഗിക്കേണ്ടത്. എന്നാൽ ഒരാൾക്ക് താങ്ങാവുന്നതിലധികം വിവരങ്ങൾ ലഭിക്കുകയാണെങ്കിൽ അയാൾക്ക് എന്ത് ചെയ്യാനാകും? ഇങ്ങനെ അധികവിവരങ്ങൾ ഉപയോഗിച്ച് പരിക്ഷീണിതനാകുന്ന അവസ്ഥയാണ് ഇൻഫോമാനിയ.

ഇന്റർനെറ്റ്, ഇമെയിൽ, സെൽഫോൺ തുടങ്ങി അനേകം സ്രോതസ്സുകളിൽ കുമിഞ്ഞുകൂടുന്ന വിവരങ്ങൾ പ്രോസസ്സ് ചെയ്യാൻ കഴിയാത്തതു മൂലമാണ് ഇൻഫോമാനിയ ഉണ്ടാകുന്നത്. വിവരങ്ങൾ ശേഖരിക്കുമ്പോൾ അവയുടെ ഗുണമേന്മയും പ്രസക്തിയും കണക്കിലെടുക്കണം. വിവരങ്ങൾ ശേഖരിക്കാനുള്ള അമിതാവേശമാണ് ഇൻഫോമാനിയ. ഇത് മൂലം കുടുംബം, കർത്തവ്യങ്ങൾ തുടങ്ങിയ കൂടുതൽ പ്രധാനപ്പെട്ട കാര്യങ്ങൾ അവഗണിക്കപ്പെടുന്നു. ഭക്ഷണ സമയത്തു പോലും വിവരങ്ങൾ ബ്രൗസ് ചെയ്ത് കൊണ്ടിരിക്കുന്നവരെ നാം കണ്ടിട്ടുണ്ട്. ഇത് ഒരു മാനസിക പ്രശ്നമാണ്. സ്ഥിരമായി ഇ-മെയിൽ, സാമൂഹിക നെറ്റ്‌വർക്കുകൾ, ഓൺലൈൻ വാർത്തകൾ എന്നിവ പരതികൊണ്ടിരിക്കുന്നത് ഇൻഫോമാനിയയുടെ ലക്ഷണങ്ങളാണ്. തങ്ങളെ തന്നെ മെച്ചപ്പെടുത്തി സ്വന്തം ഗ്രൂപ്പിൽ നിന്നും പുറത്താകാതിരിക്കുന്നതിനാണ് മിക്കവാറും ഇങ്ങനെ ചെയ്യുന്നത്.

ഇൻഫോമാനിയയ്ക്ക് അടിമപ്പെട്ടിട്ടുള്ളവർക്ക് ശ്രദ്ധയും ഉറക്കവും കുറയുന്നു. സാങ്കേതിക വിദ്യയുടെ അമിതോപയോഗം ബുദ്ധി കുറയ്ക്കുന്നു. ചിലർ ഇ-മെയിലിലും സാമൂഹിക മാധ്യമങ്ങളിലും വരുന്ന സന്ദേശങ്ങൾക്ക് മറുപടി അയയ്ക്കുന്നതിന് മുൻഗണന നൽകുന്നു. ഇതിനാൽ അവർ കുടുംബത്തിലെയും തൊഴിലിലെയും പ്രധാന കാര്യങ്ങൾ മറന്നു പോകുന്നു. ഇതിൽ നിന്ന് രക്ഷ നേടാൻ കുടുംബ കാര്യങ്ങളും തൊഴിലും ചെയ്യുമ്പോൾ സാമൂഹിക മാധ്യമങ്ങൾ ഉപയോഗിക്കുന്നതിൽ നിന്നും, ഓൺലൈൻ വാർത്തകൾ കാണുന്നതിൽ നിന്നും തൽക്കാലത്തേക്കെങ്കിലും മാറി നിൽക്കാൻ അഥവാ ലോഗ് ഓഫ് ചെയ്യാൻ പഠിക്കണം.

നിങ്ങളുടെ പുരോഗതി അറിയുക

1. ഇന്ത്യയിൽ ഏതു വർഷമാണ് ഐ.ടി നിയമം നിലവിൽ വന്നത്?
2. ഐ.ടി നിയമത്തിന്റെ ഉദ്ദേശ്യമെന്താണ്?
3. സൈബർ ഫോറൻസിക്സ് എന്നതുകൊണ്ട് ഉദ്ദേശിക്കുന്നതെന്താണ്?
4. അറിവ് സമ്പാദിക്കാനുള്ള അമിതാവേശമാണ് ________

നമുക്ക് ചുരുക്കാം

വിവര സാങ്കേതികവിദ്യ അതിവേഗം വളരുന്ന ഒരു മേഖലയാണ്. നമ്മുടെ ദൈനംദിന പ്രവർത്തനങ്ങളിൽ ഇത് വ്യാപകമായി ഉപയോഗിക്കപ്പെടുന്നു. വ്യാപാര ഇടപാടുകൾ, സർക്കാരിന്റെ വിവിധ പ്രവർത്തനങ്ങൾ, പഠന പ്രക്രിയകൾ തുടങ്ങിയവയാണ് വിവര സാങ്കേതികവിദ്യ ഗണ്യമായി ഉപയോഗിച്ചിരിക്കുന്ന പ്രധാന മേഖലകൾ. ആശയ വിനിമയത്തിന്റെ പുരോഗതി വിവര സാങ്കേതികവിദ്യയുടെ വളർച്ചക്ക് വളരെയേറെ കടപ്പെട്ടിരിക്കുന്നു. വിവര സാങ്കേതികവിദ്യയുടെ വേറൊരു പര്യായമായി വിവരവും ആശയവിനിമയ സാങ്കേതികവിദ്യയും ഉപയോഗിക്കുന്നു. അതിനാൽ വിവര സാങ്കേതികവിദ്യയും വിവര ആശയവിനിമയ സാങ്കേതിക വിദ്യയും ഉപയോഗിച്ചുള്ള സേവനങ്ങൾ ആധുനിക സമൂഹത്തിന്റെ അവിഭാജ്യ ഘടകമായി തീർന്നിട്ടുണ്ട്. ഇരുപത്തി ഒന്നാം നൂറ്റാണ്ടിലെ സാങ്കേതിക വൈദഗ്ധ്യമുള്ള ലോകത്ത് കൂടുതൽ ആളുകൾ ഇന്റർനെറ്റ് വഴി സാമൂഹിക നെറ്റ് വർക്കുകൾ, നെറ്റ് ബാങ്കിങ്, മൊബൈൽ അപ്ലിക്കേഷനുകൾ, ഇ-ഷോപ്പിംഗ് എന്നിവയിലൂടെ ഇന്റർനെറ്റ് ഉപയോഗിക്കുന്നു. ഒരു വ്യക്തിക്ക് സൈബർ ഇടം വഴി പലതരം ജോലികൾ ചെയ്യാൻ കഴിയും. അജ്ഞാത സ്വഭാവം കാരണം സൈബർ ഇടം ക്രിമിനലുകൾക്ക് വേദിയായി മാറി. കുറ്റങ്ങൾ ഒരു വ്യക്തിക്കോ, സ്വത്തിനോ സർക്കാരിനോ എതിരായിരിക്കാം. സുരക്ഷിതമായിരിക്കാനുള്ള എളുപ്പ മാർഗ്ഗം ഇന്റർനെറ്റിലെ കെണികളെക്കുറിച്ച് അറിഞ്ഞിരിക്കുക എന്നതാണ്. സൃഷ്ടിക്കുന്ന സ്രഷ്ടാവ് അല്ലെങ്കിൽ സംഗീതം, സോഫ്റ്റ്‌വെയർ, കലാസൃഷ്ടി, കൃതികൾ, കണ്ടുപിടുത്തങ്ങൾ തുടങ്ങിയവയുടെ ഉടമസ്ഥാവകാശം ബൗദ്ധിക സ്വത്തവകാശനിയമങ്ങളാൽ സംരക്ഷിക്കപ്പെട്ടിരിക്കുന്നു. സൈബർ സ്പേസിൽ സുരക്ഷിതമായ പരിസ്ഥിതിക്ക് ചട്ടങ്ങൾ ഏർപ്പെടുത്തുന്ന സൈബർ നിയമങ്ങളുണ്ട്. ഇന്ത്യക്ക് കരുത്തുറ്റ ഐ.ടി നിയമമുണ്ട്. കമ്പ്യൂട്ടറുകൾ, ഇന്റർനെറ്റ്, മൊബൈൽ ഫോൺ എന്നിവ ഉപയോഗിച്ച് നടത്തിയ വിവിധ കുറ്റകൃത്യങ്ങളെ ഇത് അഭിസംബോധന ചെയ്യുന്നു.

എല്ലാ സിവിൽ ക്രിമിനൽ കേസുകളിലും നിയമ നടപടികളുടെ ഭാഗമായി സൈബർ കുറ്റാന്വേഷണം ഉയർന്നു വന്നു. ഐ.ടി അടിസ്ഥാന സൗകര്യങ്ങൾ ഉപയോഗിച്ച് നടത്തിയ കുറ്റകൃത്യങ്ങൾ എളുപ്പത്തിൽ കണ്ടെത്താൻ കഴിയുന്ന ഡിജിറ്റൽ തെളിവുകൾ അവശേഷിക്കുന്നു. ഇന്റർനെറ്റ് ഉപയോഗിക്കുമ്പോൾ ഞങ്ങളുടെ പ്രവർത്തനങ്ങൾ മറ്റുള്ളവർക്ക് ദോഷം വരുത്തുന്നില്ലെന്ന് ഉറപ്പു വരുത്തുക. വിവരങ്ങൾ കൂടുതലാകുമ്പോൾ അവ സൃഷ്ടിച്ച പ്രശ്നങ്ങൾ പരിഹരിക്കുന്നതിനായി 'ഇൻഫോമാനിയ' എന്ന പദം ഉപയോഗിക്കുന്നു.

നമുക്ക് വിലയിരുത്താം

1. ഇടെക്സ്റ്റ് വായിക്കാൻ ഉപയോഗിക്കുന്ന ഒരു ഇലക്ട്രോണിക് ഉപകരണത്തിന് പേര് നൽകുക
2. ഏതെങ്കിലും രണ്ട് ഇ-പഠന ഉപകരണങ്ങൾക്ക് പേര് നൽകുക
3. ഇഭരണത്തിലെ വിവിധ തരം ആശയ വിനിമയം പരിചയപ്പെടുത്തുക
4. ഇഭരണത്തിന്റെ മേന്മകൾ എന്തെല്ലാമാണ്
5. അക്ഷയ സെന്ററുകളുടെ കടമകൾ എന്തെല്ലാമാണ്?
6. ഇ-പഠനം നടപ്പാക്കുന്നതിൽ അഭിമുഖീകരിക്കുന്ന പ്രധാന വെല്ലുവിളികൾ എന്തെല്ലാമാണ്?
7. ഇവ്യാപാരം നടപ്പാക്കുന്നതിന്റെ ഗുണങ്ങളും ദോഷങ്ങളും താരതമ്യം ചെയ്യുക
8. സൈബർ ഇടം എന്നാൽ എന്താണ്?
9. സൈബർ ഇടം ഒരു അയഥാർത്ഥ ലോകം എന്ന് പറയുന്നതെന്തുകൊണ്ടാണ്?
10. ഫിഷിംഗ് (phishing) എന്നത് ....ന് ഉദാഹരണമാണ്
11. വിവിധ തരത്തിലുള്ള സൈബർ കുറ്റകൃത്യങ്ങൾ വിശദമായി വിവരിക്കുക
12. സൈബർ ഇടത്തിൽ സ്വയം സംരക്ഷിക്കപ്പെടുന്നതിനുള്ള ഏറ്റവും നല്ല മാർഗ്ഗമാണ് അവബോധം. അഭിപ്രായം പറയുക
13. വ്യാപാരമുദ്രയും വ്യാവസായിക രൂപകൽപ്പനയും തമ്മിലുള്ള വ്യത്യാസം എന്താണ്?
14. പകർപ്പവകാശം എന്താണ്? കുത്തകാവകാശത്തിൽ നിന്ന് ഇത് എങ്ങിനെ വ്യത്യാസപ്പെട്ടിരിക്കുന്നു?
15. ബൗദ്ധിക സ്വത്തവകാശം മുഖേന ഉടമക്ക് നൽകിയ പ്രത്യേക അവകാശം വിവരിക്കുക
16. എന്താണ് പൈറസി അഥവാ സോഫ്റ്റ്‌വെയർ മോഷണം?
17. കയ്യേറ്റം ചെയ്യൽ (Infringement) എന്നത് കൊണ്ട് ഉദ്ദേശിക്കുന്നതെന്താണ്?
18. സൈബർ നിയമം പ്രധാനമായിരിക്കുന്നത് എന്തുകൊണ്ട്?
19. സൈബർ കുറ്റകൃത്യത്തിന്റെ പ്രാദേശികമല്ലാത്ത സ്വഭാവം അന്വേഷകർക്ക് പ്രശ്നങ്ങൾ സൃഷ്ടിക്കുന്നു. വിശദീകരിക്കുക
20. ഇൻഫോമാനിയ ഒരു മാനസിക പ്രശ്നമായി മാറിയിരിക്കുന്നു. നിങ്ങളുടെ അഭിപ്രായം എഴുതുക

പദാവലി

അംഗങ്ങൾ	:	Elements
അടിസ്ഥാന ക്ലാസ്	:	Base Class
അടിസ്ഥാന ഡാറ്റ ഇനം	:	Fundamental data type
അടിസ്ഥാന വിലാസം	:	Base address
അടിസ്ഥാനപരമായ	:	Fundamental
അന്തർനിർമിത ഫങ്ഷൻ	:	built in funcion
അവ്യക്തയുക്തി	:	fuzzy logic
അളക്കാവുന്ന അവസ്ഥ	:	scalabiltiy
അവകാശ കൈയ്യേറ്റം	:	infringement
അബ്സ്ട്രാക്ട് ഡാറ്റകൾ	:	abstract data types
അസ്ഥിര ദശാംശ സംഖ്യ	:	floating point number
ആധാരസംഖ്യ	:	base number
ആധികാരികത	:	authentication
ആഗോള സ്ഥാന നിർണ്ണയ വ്യവസ്ഥ	:	global positioning sysetm (GPS)
ആർഗ്യുമെൻറ്	:	argument
ഇ - വ്യാപാരം	:	e-business
ഉപവാക്യ ടാഗ്	:	phrase tag
IP മേൽവിലാസം	:	IP address
കമ്പ്യൂട്ടിങ് യന്ത്രം	:	computing machine
കമ്പ്യൂട്ടിങ് വിജ്ഞാനശാഖ	:	computing discipline
കമ്പ്യൂട്ടിങ് സിദ്ധാന്തം	:	theory of computing
ക്ലസ്റ്റർ കമ്പ്യൂട്ടിങ്	:	cluster computing
കുത്തകാവകാശം	:	patent
കുറവ് ഓപ്പറേറ്റർ	:	decrement operator()
ക്രമ ലിസ്റ്റ്	:	ordered list
ക്രമപ്പെടുത്തൽ	:	sorting
ക്രമപ്രകാരമുള്ള വർഗീകരണം	:	hierarchial classification
ക്രമീകരിക്കുക	:	set up
ഗണന ബുദ്ധി	:	computational intelligenece

ഘടന	:	template
R വില	:	R Value
ടൈപ്പ് മോഡിഫയർ	:	type modifier
ഡാറ്റ അംഗങ്ങൾ	:	data elements
ഡാറ്റ ഇനം	:	data type
ഡാറ്റ സ്ട്രക്ചർ	:	data structure
ഡാറ്റ സുരക്ഷ	:	data security
ഡാറ്റ സംഗ്രഹം	:	data abstraction
ഡാറ്റ നിർവചന ഭാഷ	:	data definion language
ഡാറ്റ കൈകാര്യം ചെയ്യുന്ന ഭാഷ	:	data manipulation language
ഡാറ്റ നിയന്ത്രണ ഭാഷ	:	data control language
ഡൊമൈൻ നാമകരണ സമ്പ്രദായം	:	domain naming system
തനതുവില	:	fefault value
തലക്കെട്ട്	:	heading
തിരയൽ	:	dearch
ത്രികോണമിതി	:	trigonometry
ത്രിതല ഘടന	:	three tier architecture
ദ്വിതീയ മെമ്മറി	:	secondary memory
ദൃശ്യസാങ്കേതിക വിവരവ്യവസ്ഥ	:	graphical information sysetm
നിർമിത ബുദ്ധി	:	artificial intelligence
നിർവചനം	:	definition
നീക്കം ചെയ്യൽ	:	deletion
നീക്കി വയ്ക്കുക	:	allocate
നിർമിത ബുദ്ധി	:	artifical intelligence
പകർപ്പവകാശം	:	copy right
പദപ്രയോഗം	:	expression
പരിപാലനം	:	maintainabiltiy
പരിവർത്തനാത്മകം	:	dynamic
പരിണാമിക ഗണം	:	evolutionary computation
പുനരുപയോഗം	:	reusabiltiy

പൊതു അവലോകനം	:	overview
പൊതു സേവന കേന്ദ്രങ്ങൾ	:	common service centre
പ്രഖ്യാപനം	:	declaration
പ്രഖ്യാപന പ്രസ്താവന	:	declaration statement
പ്രശ്നങ്ങൾ നിർധാരണം ചെയ്യുക	:	problem solving
പ്രസ്താവന	:	statement
പ്രാഥമിക മെമ്മറി	:	primary memory
പ്രാഥമികമായ ഡാറ്റാ സ്ട്രക്ച്ചർ	:	simple data structure
പ്രാദേശിക DNS	:	local DNS
പ്രാരം വില നൽകൽ	:	initialization
പ്രാവർത്തികമാക്കൽ	:	implementation
പ്രോഗ്രാമിങ് മാതൃക	:	programming paradigm
ബാഹ്യ ഇനം മാറ്റൽ	:	explicit type conversion
ബാഹ്യഫയൽ	:	external file
ബൗദ്ധിക സ്വത്താവകാശം	:	intellectual property right
മാതൃക	:	paradigm
മുഖം തിരിച്ചറിയൽ	:	face recognition
മെമ്മറി വലുപ്പം	:	memory size
മെമ്മറിസ്ഥാനം	:	memory location
മൊബൈൽ ആശയവിനിയമം	:	mobile communication
രൂപമാറ്റം വരുത്തുക	:	encrypt
രൂപീകൃത ക്ലാസ്	:	derived class
രൂപീകൃത ഡാറ്റ ഇനം	:	derived data type
രൂപീകൃത ഡാറ്റ ഇനം	:	derived data type
രേഖീയം	:	linear
രേഖീയമല്ലാത്തത്	:	non liner
ലയിക്കുക	:	merging
ലിഖിതം	:	document
വർദ്ധനവ് ഓപ്പറേറ്റർ	:	increment operator
വഹനീയത	:	portabiltiy

വാക്യ ഘടന	:	syntax
വിദൂര സെർവർ	:	remote server
വില നൽകൽ പ്രസ്താവന	:	assignment statement
വിലാസ സമ്പ്രദായം	:	addressing scheme
വ്യാപാര മുദ്ര	:	trade mark
വൃത്താകൃത ക്യൂ	:	circular queue
ശകലം	:	segment
ശീർഷകം	:	caption
സന്ദേശ ജാലകം	:	message window
സമഗ്രത	:	integrtiy
സമ്പർക്കമുഖം	:	interface
സംയോചിതം/ഏകോപിതം	:	grouped
സംയോജിത/ഏകോപിത ഡാറ്റ സ്ട്രക്ചർ	:	compound data structure
സാർഥക ചിഹ്നങ്ങൾ	:	reserved character
സിംഗിൾ ഇൻഹെറിറ്റൻസ്	:	single inheritance
സിംഗിൾ ലിങ്ക്ട് ലിസ്റ്റ്	:	singly linked lsti
സുരക്ഷിതം	:	secure
സെർവർ നവീകരണം	:	server update
സ്ക്രീൻ വലുപ്പം	:	screen size
സ്ട്രക്ച്ചർ	:	structure
സ്ഥാനീയ സംഖ്യാന സമ്പ്രദായം	:	positional value number system
സ്ഥിര ഡാറ്റ സ്ട്രക്ച്ചർ	:	static data structure
സ്ഥിരവില	:	constant
സ്വയം സൂചിത	:	self referential
സ്റ്റാക്	:	stack
സ്റ്റോക് അണ്ടർഫ്ളോ	:	stack underflow
സ്റ്റോക് ഓവർഫ്ളോ	:	stack overflow
സംഭാഷണം ഗ്രഹിക്കുക	:	speech recognition
സൈബർ ഇടം	:	cyber space
സൈബർ നിയമങ്ങൾ	:	cyber laws

സൈബർ കുറ്റാന്വേഷണം	:	cyber forensics
സ്വയം പ്രവർത്തിത ഭാഷണവേചനം	:	auomtatic speech recognition (ASR)