Chapter 9

Structured Query Language

SQL is a language designed for communicating with a database for implementing RDBMS.

RDBMS കൈകാര്യം ചെയ്യാനുപയോഗിക്കുന്ന കമ്പ്യൂട്ടർ ഭാഷയാണ് SQL

Features of SQL

- It is a relational database language not a programming language.
- It is simple and powerful.
- It allows the user to create, update, delete, and retrieve data from a database.

- ഇതൊരു പ്രോഗ്രാമിങ് ഭാഷയല്ല
- ഇത് ലഘുവും ശക്തവുമാണ്
- ഡേറ്റാബേസിൽ create, update, delete, retrieve data എന്നിവ സാധ്യമാണ്

Components of SQL

Data Definition Language(DDL)

DDL commands are used to create, modify and remove database objects. DDL commands are CREATE, ALTER and DROP.

ഡേറ്റാബേസ് നിർമ്മിക്കാനും മാറ്റം വരുത്താനും ഒഴിവാക്കാനുമുള്ള കമാന്റുകളാണ് DDL commands. CREATE, ALTER, DROP എന്നിവയാണ് കമാന്റുകൾ

Data Manipulation Language(DML)

DML is a component of SQL used to interact with database. DML commands are SELECT, INSERT, UPDATE and DELETE.

ഡേറ്റായെ കൈകാര്യം ചെയ്യാനുള്ള കമാന്റുകളാണ് DML. SELECT, INSERT, UPDATE, DELETE എന്നിവയാണ് കമാന്റുകൾ

Data Control Language(DCL)

DCL is used to control access to database. DCL commands are GRANT, COMMIT, ROLLBACK and REVOKE.

ഡേറ്റാബേസിനെ നിയന്ത്രിക്കാനുള്ള കമാന്റുകളാണ് DCL. GRANT, COMMIT, ROLLBACK, REVOKE എന്നിവയാണ് കമാന്റുകൾ

MySQL

MySQL is an is an open-source relational database management system (RDBMS). The main features of SQL are

- open source license.
- It is portable and works with many languages such as PHP, PERL, C, C++, JAVA etc.
- It is used for web applications.
- It is a central component of the widely used LAMP open source web application software.

Open-source അധിഷ്ഠിത relational database management system (RDBMS) ആണ് MySQL. ഇതിന്റെ സവിശേഷതകൾ

- open source license.
- PHP, PERL, C, C++, JAVA തുടങ്ങിയ ഭാഷകളോടൊപ്പം ഉപയോഗിക്കാം
- web applications നിർമ്മിക്കാം
- LAMP open source web application software ലെ പ്രധാന ഭാഗമാണ്.

MySQL Commands

Creating a database

A database in MYSQL is created using the CREATE DATABASE command. The syntax is
CREATE DATABASE <Database Name> ;

MYSQL ൽ ഡേറ്റാബേസ് നിർമ്മിക്കാൻ CREATE DATABASE command ഉപയോഗിക്കുന്നു.
CREATE DATABASE <Database Name>;

Opening a database

A database can be using the USE command. A database becomes active when we opens it. The syntax is
USE <Database_Name> ;

ഡേറ്റാബേസ് ഉപയോഗക്ഷമമാക്കുവാൻ USE കമാന്റ് ഉപയോഗിക്കുന്നു.
USE <Database_Name> ;

The SHOW command is used to check the existence of a database or table. The Syntax

Database, table എന്നിവയുടെ പട്ടിക കാണിക്കുവാൻ SHOW കമാന്റ് ഉപയോഗിക്കുന്നു.

is
SHOW DATABASES;
SHOW TABLES;

SHOW DATABASES;
SHOW TABLES;

Data types in SQL

Numeric data type

INT(INTEGERS) data type

Integers are represented in My SQL by INT data type. They are whole numbers ie, without fractional part.
Eg: INT(3) – Integer with 3 digits

INT ഉപയോഗിച്ച് ദശാംശമില്ലാത്ത പൂർണ്ണസംഖ്യകളെ ഉൾപ്പെടുത്താം
ഉദാ - INT(3) – 3 അക്കങ്ങളുള്ള സംഖ്യ

DEC (DECIMAL) data type

The DEC data type represents fractions. Numbers. The syntax is DEC(Size,Scale), where Size is the total number of digits and scale is the number of digits after the decimal point. For example DEC(5,2), where the total number of digits is 5 and the number of digits after the decimal point will be 2.

ഭിന്നസംഖ്യകളെ ഉൾപ്പെടുത്തുവാൻ DEC(Size,Scale), ഉപയോഗിക്കുന്നു. Size ആകെയുള്ള അക്കങ്ങളെയും ,Scale ദശാംശത്തിനുശേഷമുള്ള അക്കങ്ങളെയും സൂചിപ്പിക്കുന്നു.
ഉദാ - DEC(5,2) – ആകെ അക്കങ്ങൾ 5 ദശാംശം കഴിഞ്ഞ് 2.

String data types

A string is a group of characters.

ഒരുകൂട്ടം ക്യാരക്ടറുകളാണ് string

1)CHAR(CHARACTER) data type

Character includes letters, digits, special symbols etc. It is a fixed length data type. The syntax is CHAR(Size) where Size represent the maximum number of characters. The default size of CHAR data type is 1.

അക്ഷരങ്ങളും അക്കങ്ങളും ചിഹ്നങ്ങളും ഉൾപ്പെടുന്നതാണ് CHAR. ഇതിന്റെ വലിപ്പം സ്ഥിരമായിരിക്കും. CHAR(Size) എന്നാണ് ഇതുപയോഗിക്കുന്നത്. Size ഉപയോഗിക്കുന്ന ക്യാരക്ടറുകളുടെ എണ്ണത്തെ സൂചിപ്പിക്കുന്നു.

2) VARCHAR data type

The VARCHAR data type represent variable length strings. It is similar to CHAR data type but only allocates memory according to the size of the string. It saves memory space.

VARCHAR അക്ഷരങ്ങളും അക്കങ്ങളും ചിഹ്നങ്ങളും ഉൾപ്പെടുന്നു. VARCHAR(Size) എന്നാണ് ഇതുപയോഗിക്കുന്നത്. Size ആവശ്യമുള്ള ക്യാരക്ടറുകളുടെ എണ്ണത്തെ സൂചിപ്പിക്കുന്നു. പക്ഷേ മുഴുവൻ സ്ഥലവും ഉപയോഗിക്കുന്നില്ല.

Date and Time data type

1)Date data type:-The date data type is used for storing date. The date in MySQL is represented in YYYY-MM-DD format (Standard format).
2)Time data type:-The time data type is used for storing time.The format is HH:MM:SS.

1)Date data type:- YYYY-MM-DD format ൽ തീയതി സൂക്ഷിക്കുവാനുപയോഗിക്കുന്നു.
2)Time data type:- HH:MM:SS format ൽ സമയം സൂക്ഷിക്കുവാൻ ഉപയോഗിക്കുന്നു.

SQL commands

DDL Commands:-DDL command deals with structure of database. It consists of three commands CREATE, ALTER and DROP.

DDL Commands:-DDL command database ന്റെ ഘടനയുമായി ബന്ധപ്പെട്ട് നിൽക്കുന്നു. CREATE, ALTER and DROP എന്നിവയാണ് കമാന്റുകൾ

CREATE TABLE Command

The CREATE TABLE Command is used to create a table(relation). The syntax is
CREATE TABLE <TableName>
(<ColumnName1>
<DataType> [<Constraints>] , <ColumnName2>
<DataType> [<Constraints>] ,....................................);

The CREATE TABLE Command table (relation) നിർമ്മിക്കാൻ ഉപയോഗിക്കുന്നു. syntax
CREATE TABLE <TableName>
(<ColumnName1>
<DataType> [<Constraints>] , <ColumnName2>
<DataType> [<Constraints>] ,....................................);

TableName - the name of the table
ColumnName - name of the column
Constraint – Condition (check) applied to a column or group of columns.
Example:-CREATE TABLE STUDENT(ROLLNO INT,NAME VARCHAR(20),TOTAL_MARK INT,PERCENTAGE DEC(5,2));

TableName – table ന്റെ പേര്
ColumnName – കോളത്തിന്റെ പേര്.
Constraint – ഒന്നോ അതിലധികമോ കോളങ്ങളെ നിയന്ത്രിക്കുന്നു
Example:-CREATE TABLE STUDENT(ROLLNO INT,NAME VARCHAR(20),TOTAL_MARK INT,PERCENTAGE DEC(5,2));

Rules for naming a table

1. The name must not be an SQL keyword.
2. The name must begin with alplabets(A-Z or a-z).
3. The name may contain a character,under score(_) and dollar ($) sign.
4. The name should not be the name of an existing table(duplicate).
5. The name of the table must not contain white space,special symbols.

- SQL keyword ആകാൻ പാടില്ല
- alplabet ൽ (A-Z or a-z) തുടങ്ങണം.
- അക്ഷരം, അക്കം, ,under score(_), dollar ($) എന്നിവ ഉൾപ്പെടുത്താം
- നിലവിലുള്ള table ന്റെ പേര് ആകാൻ പാടില്ല
- white space, special symbols എന്നിവ പാടില്ല

Constraints

A constraint is a condition applied to a column or group of columns. Constraints are classified into Table Constraints and Column Constraints. A table constraint is applied to a table where as a column constraint is applied to a column. Constraints ensure the database integrity hence they are also called database integrity constraints.

column constraints

- NOT NULL:- This constraint ensures that a column can never have NULL(empty) values.
- AUTO_INCREMENT:- it automatically assigns a series of number and insert it to column. The default starting value is 1. The auto increment column must be defined as primary key of the table. Only one auto_increment column is allowed in a table.
- UNIQUE:-This constraint ensures that no two rows have the same value in a specified column.
- DEFAULT:-This constraint is used to specify a default value for a column.

Table Constraints

- PRIMARY KEY:-It declares a column as the primary key of a table. The primary key cannot contain NULL value
- CHECK:-This constraint limits the values that can be inserted into a column of a table.

ഒന്നോ അതിലധികമോ കോളങ്ങളുടെ മൂല്യം നിയന്ത്രിക്കാനുള്ളതാണ് constraint. ഇത് Table Constraint, Column Constraint എന്ന് രണ്ടായി തിരിച്ചിരിക്കുന്നു. Table Constraint കൾ Table നെ മുഴുവനായും Column Constraint കൾ Column മാത്രമായും നിയന്ത്രിക്കുന്നു.

column constraints

- NOT NULL:- NULL value ഇല്ല എന്ന് ഉറപ്പ് വരുത്തുന്നു.
- AUTO_INCREMENT:- പ്രത്യേക കോളത്തിലെ മൂല്യം തനിയെ വർദ്ധിപ്പിക്കുന്നു. ഇത് primary key ആയിരിക്കണം. ഒരു കോളം മാത്രമേ ഇത്തരത്തിൽ അനുവദിക്കൂ. 1 മുതൽ ഇതിന്റെ മൂല്യങ്ങൾ ആരംഭിക്കുന്നു.
- UNIQUE:- പ്രത്യേക കോള്ത്തിലെ രണ്ട് വരികൾക്ക് ഒരേ മൂല്യം വരുന്നത് തടയുന്നു.
- DEFAULT:- ഒരു കോളത്തിൽ പ്രത്യേക മൂല്യം വരാൻ സഹായിക്കുന്നു.

Table Constraints

- PRIMARY KEY:- ഒരു കോളത്തെ PRIMARY KEY ആക്കുന്നു.
- CHECK:- ഒരു കോളത്തിൽ വരാവുന്ന മൂല്യങ്ങളെ നിയന്ത്രിക്കുന്നു.

DESC or DESCRIBE Command

The DESC or DESCRIBE Command is used view(display) the structure of a table.

DESC or DESCRIBE Command ഉപയോഗിച്ച് table ന്റെ ഘടന കാണാം

DML Commands

Data Manipulation Language (DML) commands are used for manipulating data in database.

ഡേറ്റാബേസിലെ ഡേറ്റായെ കൈകാര്യം ചെയ്യാൻ DML കമാന്റുകൾ ഉപയോഗിക്കുന്നു.

INSERT INTO Command

This command is used to insert a row(tuple) into a table. The syntax is:
INSERT INTO <TABLENAME> VALUES(<Value1>,<Value2>,....);
Example:-

- INSERT INTO STUDENT VALUES (1106,'ANISH',599);
- INSERT INTO STUDENT (ROLLNO,NAME,TOTALMARK) VALUES (1106,'ANISH',599);
- STUDENT (ROLLNO,NAME,TOTALMARK) VALUES (1106,'ANISH',599), (1109,'ANUJ',514);
- The above command creates two rows into the table STUDENT.

Table ലേയ്ക്ക് ഒന്നോ അതിലധകമോ വരികൾ ചേർക്കുവാൻ INSERT INTO കമാന്റ് ഉപയോഗിക്കുന്നു.syntax:
INSERT INTO <TABLENAME> VALUES(<Value1>,<Value2>,....);
Example:-

- INSERT INTO STUDENT VALUES (1106,'ANISH',599);
- INSERT INTO STUDENT (ROLLNO,NAME,TOTALMARK) VALUES (1106,'ANISH',599);
- STUDENT (ROLLNO,NAME,TOTALMARK) VALUES (1106,'ANISH',599), (1109,'ANUJ',514);
- മുകളിൽ തന്നിരിക്കുന്ന command രണ്ട് rows STUDENT table ൽ ചേർക്കുന്നു.

SELECT Command

The SELECT Command is used to select rows (tuples or record) from a table. The syntax is
SELECT <ColumnName1>,[<ColumnName2>,....] FROM <TableName> ;
Example:-SELECT ROLLNO,NAME,TOTALMARK from STUDENT ;
The output is

ROLLNO	NAME	TOTALMARK
1106	ANISH	599
1109	ANUJ	514

Note:-To select all columns instead of column name asterisk(*) can be used .
Example :-SELECT * FROM STUDENT ;

Table ൽ ഉൾപ്പെട്ടിരിക്കുന്ന record കളെ എടുക്കാനുള്ള കമാന്റാണ് SELECT.
SELECT <ColumnName1>,[<ColumnName2>,....] FROM <TableName> ; എന്ന വിധത്തിലാണ് ഈ കമാന്റ് ഉപയോഗിക്കുന്നത്.
SELECT * FROM <TableName> ; എന്ന കമാന്റ് എല്ലാ കോളങ്ങളെയും എടുത്തുകാണിക്കുന്നു.

The DISTINCT Keyword

The Keyword DISTINCT is used to avoid duplicate rows from the result of a select command.
The keyword ALL is used to display duplicate rows in a select command.

SELECT കമാന്റിന്റെ ഫലത്തിലെ ആവർത്തനം ഒഴിവാക്കാൻ DISTINCT എന്ന വാക്ക് ഉപയോഗിക്കുന്നു.
SELECT കമാന്റിന്റെ ഫലത്തിലെ ആവർത്തനം ഉൾപ്പെടുത്താൻ ALL എന്ന വാക്ക് ഉപയോഗിക്കുന്നു.

WHERE Clause

The WHERE clause is used to select rows or columns from a table which satisfy a specific condition. The Syntax is
SELECT <Columnname>,[<ColumnName>,.....] FROM <TableName> WHERE <Condition> ;
The conditions can be expressed with the help of relational operators. The list of operators used in SQL are >, >=, <,<=,=,<>

പ്രത്യേക കണ്ടീഷൻ അനുസരിക്കുന്ന റിക്കോർഡുകളെ കണ്ടെത്താൻ WHERE ഉപയോഗിക്കുന്നു. കണ്ടീഷനുകൾ രൂപികരിക്കാൻ റിലേഷനൽ ഓപ്പറേറ്ററുകൾ ഉപയോഗിക്കുന്നു.

The Like keyword

The Like keyword is used to search for a specified pattern in a column.
Examples:-

- Select * from STUDENT where place='kannur';
  - Selects all from student table whose place is equal to 'kannur'

- Select * from STUDENT where name like 'A%';
  - Selects all from student table whose Name start with A, (% replace for any number of characters).
- Select * from STUDENT where name like '%m';
  - Selects all from student table whose name end with m
- Select * from STUDENT where name like '%la%';
  - Selects all from student table whose name contains la.

പ്രത്യേക പാറ്റേൺ കണ്ടെത്താൻ LIKE ഉപയോഗിക്കുന്നു.

BETWEEN ………… AND Operator

The BETWEEN …………. AND operator is used to specify a range .

BETWEEN …………. AND ഉപയോഗിച്ച് തുടർച്ചയായുള്ള ചില സെറ്റ് കണ്ടുപിടിക്കാം

Example:-

- Select * from STUDENT where mark1 between 40 and 50;
  - Selects all from student table whose mark in between 40 and 50
- Select * from STUDENT where mark1 not between 20 and 35;
  - Selects all from student table whose mark is not in between 20 and 35

The IN Operator

The IN operator is used for setting a condition satisfying any list of values.

IN ഉപയോഗിച്ച് തെരഞ്ഞെടുക്കപ്പെട്ട് സെറ്റ് കണ്ടുപിടിക്കാം

Example:-

- SELECT Name,Total_Mark FROM STUDENT WHERE COURSE IN ('COMMERCE','HUMANITIES') ;
  - Displays name,totalmark of all students studying in either Commerce or Humanities.

IS NULL Operator

The SQL NULL is the term used to represent a missing value. The IS NULL operator in a WHERE clause to find rows containing a null value in a particular column.

Example:-

SELECT * FROM STUDENT WHERE Course IS NULL ;

Display details of students whose course is not specified.

ശൂന്യമായിക്കിടക്കുന്ന ചില കോളങ്ങൾ കണ്ടുപിടിക്കാൻ IS NULL ഉപയേഗിക്കുന്നു.

ORDER BY Clause

The ORDER BY clause is used to sort the result of a select statement in ascending or descending order. The default order is ascending. The ascending order is specified by the ASC keyword and descending order by DESC keyword.

ഒന്നോ രണ്ടോ കോളങ്ങളെ അടിസ്ഥാനപ്പെടുത്തി ascending അല്ലെങ്കിൽ descending order ൽ ലഭിക്കുവാൻ ORDER BY ഉപയോഗിക്കുന്നു. സാധാരണയായി ascending order ലാണ് കാണുന്നത്. descending order ൽ കാണാൻ DESC ഉപയോഗിക്കുന്നു. ASC ഉപയോഗിച്ചാൽ ascending order ൽ കാണാം

COUNT ( ) Function

The COUNT() function returns the number of rows that matches a specified condition. The COUNT(*) function returns the number of records in a table.

Example:

SELECT COUNT(*) from Student ;

Display the total number of Students in the Students table.

പ്രത്യേക കണ്ടീഷൻ അനുസരിച്ചുള്ള മൂല്യങ്ങളുടെ എണ്ണം കണ്ടുപിടിക്കാൻ COUNT() ഉപയോഗിക്കുന്നു. പ്രത്യേക കണ്ടീഷൻ അനുസരിച്ചുള്ള റിക്കോർഡുകളുടെ എണ്ണം കണ്ടുപിടിക്കാൻ COUNT(*) ഉപയോഗിക്കുന്നു.

GROUP BY Clause

The GROUP BY clause is used to group the rows of a table based on a common value.

Example:-

SELECT Course,COUNT(*) FROM STUDENT GROUP BY Course

Display each Course and Number of Students in each Course.

പൊതുമൂല്യം അനുസരിച്ച് കോളങ്ങളെ ഒന്നിപ്പിക്കുവാൻ GROUP BY ഉപയോഗിക്കുന്നു.

Aggregate Functions

The aggregate functions acts on a group of data and returns a single data. They are also called summary functions.

Commonly used aggregate functions

AVG() - Returns the average value

COUNT() - Returns the number of rows
FIRST() - Returns the first value
LAST() - Returns the last value
MAX() - Returns the largest value
MIN() - Returns the smallest value
SUM() - Returns the sum
ഒരു കോളത്തിലെ മൂല്യങ്ങളെ ഒന്നായി പരിഗണിച്ച് പൊതുവായ മൂല്യം കാണാൻ aggregate function ഉപയോഗിക്കുന്നു.

Changing the structure of a table

The structure of a table can be changed by using DDL Commands. It includes adding or removing columns, Changing data type and size of existing columns, renaming table etc.
ഒരു table ന്റെ ഘടന മാറ്റാൻ സാധിക്കുന്നതാണ്. പുതിയ കോളങ്ങൾ ചേർക്കാനും ഉള്ളത് ഒഴിവാക്കാനും പേര് മാറ്റാനും ഡേറ്റായുടെ തരവും വലിപ്പവും മാറ്റാനും കഴിയും

ALTER TABLE Command

The ALTER TABLE Command is used to change table structure.
Adding New Column

- ALTER TABLE <TableName> ADD (<ColumnName> <NewDataType> [(<Size>)] <ColumnConstraint>);
  - Example:- ALTER TABLE STUDENT ADD (Grade CHAR(2));
  - To add a new column Grade to the table STUDENT .

Modifying Existing column

- ALTER TABLE <TableName> MODIFY (<ColumnName> <NewDataType> [(<NewSize>)]) ;

Removing column from a table

- ALTER TABLE <TableName> DROP <ColumnName> ;
  - Example:-ALTER TABLE STUDENT DROP Total_Mark ;
  - To remove the column Total_Mark from the table STUDENT.

Renaming a table

- ALTER TABLE <TableName> RENAME TO <New_Table_Name> ;
  - Example:- ALTER TABLE STUDENT RENAME TO STU;
  - To rename STUDENT table to STU.

Deleting rows from a table

The DELETE command can be used to remove individual rows from a table.
The Syntax is:
DELETE FROM <Table_Name> [WHERE <condition>] ;
Example:-DELETE FROM STUDENT Where RollNO=1109 ;
To remove the details of student with RollNo 1109.
DELETE command ഉപയോഗിച്ച് ഒരു table ലെ റിക്കോർഡുകളെ ഒഴിവാക്കാം

UPDATE Command

The UPDATE command is used to change the values in a column of specified rows. The rows are set to new values using the SET keyword. The syntax is:
UPDATE <TableName> SET <ColumnName> =<value>[,<ColumnName>=<value> ....] [Where <Condition>];
Example:-
UPDATE STUDENT SET ROLLNO=1114 WHERE Name='Manu' ;
To change the rollno of a student named Manu from the table student.
UPDATE command ഉപയോഗിച്ച് ഒരു table ലെ റിക്കോർഡുകൾക്ക് ആവശ്യമായ മാറ്റം വരുത്താം.

Removing table from a database

A table can be removed from a database by using the DROP TABLE Command. It is a DDL which permanently removes table from the database.The syntax is
DROP TABLE <TableName> ;
Example:-
DROP TABLE STUDENT;
To remove the table STUDENT from the table.
DROP TABLE Command ഉപയോഗിച്ച് ഡേറ്റാബേസിൽനിന്നും table നെ മുഴുവനായി ഒഴിവാക്കാം

Nested Queries

A Nested query is a query placed within another SQL query. The query that contains the subquery is called outer query and the inner query is called subquery. The inner query is

evaluated first and then the outer query is evaluated.
Example:-
UPDATE STUDENT SET Total_Mark=Total_Mark + 25 WHERE AGE IN (SELECT AGE FROM STUDENT WHERE AGE >=21) ;
Update Total_Mark by 25 marks in Students table for all students whose age is greater than or equal to 27.
ഒരു query യുടെ ഉള്ളിൽ മറ്റൊരു query വയ്ക്കുന്നതാണ് nested query. ഇവിടെ അകത്തുള്ള query ആദ്യം പ്രവർത്തിച്ച് അതിന്റെ ഫലത്തിനനുസരിച്ച് പുറമേയുള്ള query കൾ പ്രവർത്തിക്കുന്നു.

Concept of Views

A view is a virtual table which is derived from an existing table(Base table). The CREATE VIEW Command is used to create a VIEW. The Syntax is
CREATE VIEW <View_Name> AS SELECT <ColumnName1> [,<ColumnName2>,......]
FROM <TableName> [Where <Condition>] ;
Example:-
CREATE VIEW Student1 AS SELECT * FROM STUDENT Where Course='Science ' ;
The DROP VIEW Command can be used to remove view
Advantages of View are

- Views allows to setup different security levels for a table.
- Views allows to see the same data in a different way.
- It helps to hide complexity.

ഒന്നോ അതിലധികമോ table ൽ നിന്നും കണ്ടെത്തുന്ന സാങ്കൽപിക table ആണ് view.

Advantages of View are

- table ന്റെ സുരക്ഷ വർദ്ധിപ്പിക്കാൻ കഴിയുന്നു.
- ഒരു ഡേറ്റായെ പല വിധത്തിൽ കാണാൻ കഴിയുന്നു.
- സങ്കീർണ്ണത ഒളിപ്പിക്കാൻ കഴിയുന്നു.