RDBMS : Sql Server , MySql

SQL : Structured Query Language

It’s a language that is used to interact with RDBMS

Features are

1. Case insensitive
2. Its syntax is similar to english words
3. Easy to learn

Sql Statements

1.DDL : Data Definition Language : To define something

Create , alter , drop , truncate table

2. DML : Data Manipulation Language : Manipulate Records

Insert , delete , update , select

3.DQL : Data Query Language (select)

4.TCL : Transaction Control Language

Commit , rollback

5.DCL : Data Control Language

Grant , revoke

DDL Commands

Create : It is used to create <object>

Database : Collection of records

In terms of Sql Server , MySql > Database is a collection of objects

Where objects could be table, procedure , index , functions , views . etc

Syntax for making table

Create table <tablename> (column\_name column\_type)

-- Create database

--DDL

create database NewDb

-- Remove Database

--DDL

drop database NewDb

-- Switch to a particular database

use NewDb

-- Create table

--DDL

create table employee

(

id int,

name char(20),

dept char(20),

salary int

)

-- To see all records

-- DML / DQL

select \* from employee

insert into employee values(1, 'Ajay','HR',9000)

insert into employee values(2, 'Vijay','Accounts',23000)

insert into employee values(3, 'Sagar','HR',19000)

insert into employee values(4, 'Naveen','Accounts',33000)

insert into employee values(1, 'Preeti','Sales',29000)

insert into employee values(6, 'Deepak','Accts',23000)

insert into employee values(7, 'Nargis','HR',28000)

insert into employee values(8, 'Pawan','Sales',41000)

insert into employee values(9, 'Gagandeep','HR',45000)

insert into employee values(10, 'Hari','Accounts',63000)

select \* from employee

Operators

Which acts upon operands

Arithmatical -> + - \* / %

Comparison / Relational -> > , < , >= , <= <> , =

Logical (It is used when conditions are more than one) -> AND OR NOT

-- \* means all the columns

Select \* from employee

-- Projection

Select id , name from employee

select id , name , salary from employee

select id "Employee ID" from employee

select id As "Employee ID" from employee

select id AS "Employee ID" , name "Employee Name" from employee

-- Selection

-- done by usinh where clause

-- Give all records of HR dept

Select \* from employee where dept = 'HR'

-- Give all records of HR & Accounts dept

Select \* from employee where

dept = 'HR' Or dept ='Accts'

-- Give all records where dept is not HR

Select \* from employee where dept <> 'HR'

-- Give records where salary is more than 30000

select \* from employee where salary > 30000

-- Give records where salary is less than 30000

select \* from employee where salary < 30000

-- Give records where salary is more than 30000 and less than 45000

select \* from employee where salary > 30000 and salary < 45000

-- To replace AND , we can use BETWEEN operator

select \* from employee where salary BETWEEN 30000 AND 45000

-- To replace OR , we can use IN operator

-- Give records of HR and Accts Dept

select \* from employee where dept = 'HR'

OR dept='Accts'

select \* from employee where dept

IN ('HR','Accts')

select \* from employee

-- give all records where name starts with N

-- For pattern matching , we use LIKE operator

select \* from employee where Name like 'Deepak'

-- We have to wild card characters

-- % means any no. of characters

-- \_ means upto 1 character

select \* from employee where Name Like 'N%'

select \* from employee where Name Like '\_\_\_\_'

select \* from employee where Name Like '\_a%'

-- Modify , update record

Update table <tablename> set <columnname> = value

-- Insert multiple records with single insert starmenst

insert into employee values

(1, 'Ajay','HR',9000),

(2, 'Vijay','Accounts',23000),

(3, 'Sagar','HR',19000),

(4, 'Naveen','Accounts',33000),

(1, 'Preeti','Sales',29000),

(6, 'Deepak','Accts',23000),

(7, 'Nargis','HR',28000),

(8, 'Pawan','Sales',41000),

(9, 'Gagandeep','HR',45000),

(10, 'Hari','Accounts',63000)

select \* from employee

--- delete table

drop table employee

create table employee

(

id int,

name char(20),

dept char(20),

salary int

)

-- Modify , update record

-- Increase salary by 2000 for all where dept is HR

update Employee set salary = salary + 2000

select \* from employee

update Employee set salary = salary + 2000

where dept='HR'

-- Delete Record

-- delete {from} tablename

delete employee

delete employee where dept in ('HR', 'Accounts')

To modify table structure, the command that is used in alter table

select \* from employee

-- Add one more column

alter table employee add address char(20)

update employee set address='Delhi'

where dept='HR'

update employee set address='Calcutta'

where dept <> 'HR'

-- To remove column

alter table employee drop column dept

-- Change column width of address from 20 to 50

alter table employee alter column address

char(50)

-- Rename table

-- For this , we have to call a inbuilt

-- procedure sp\_rename

-- exec sp\_rename <table old\_name> , <table new\_name>

exec sp\_rename 'Employee' , 'Emp'

-- Rename a column name

exec sp\_rename 'Emp.name' ,'empname' ,'column'