**--LIKE**

--LIKE operator is used to search for a specified pattern in a column.

--Mostly like operator is used in where clause.

--Like operator used wildcards for searching a pattern

--1. % - Represents zero, one or multiple characters or numbers./ A substitute for Zero or more characters

--2. \_ - Represents one or single characters./A A substitute for exactly one character.

--3.[Charlist] - Any single character in charlist ex: [ABC]

--4.[^Charlist] -any character not in charlist

--ex: Seeta,meeta,geeta sena, sona siya

--'S%' - start with 'S' character and it will display all the names which starts with S.

--'%S' - End with 'S' character and it will display all the names which END with S.

--'%S%' -Anywhere inside record/column if 'S' character and it will display all the names which starts or ends or anywhere inside into a column.

select \* from employee where FirstName like 's%' ---at the start of name s

select \* from employee where FirstName like '%A' -- at the end of name a

select \* from employee where FirstName like '%A%' --anywhere inside or start or end.

--Display the name whose third letter starts with r

select \* from employee where FirstName like '\_\_r%' --Kirti,Virat

--Display the name which starts with s and ends with A

select \* from employee where FirstName like 's%a'

select \* from employee where FirstName like '[ARV]%' -- it will display the names which start with A,R and V.

select \* from employee where FirstName like '[^ARV]%' -- it will display the name which not start with A,R and V.

select \* from employee where FirstName like '%[ARV]'

select \* from employee where firstname like '[A-E]%' -- it will diplay all the names which is in range of A to E

select \* from employee where firstname like '[A-O]%'

**--Q.How will you display the names which ends with r and t?**

**--Q.How will you display the name whose second last letter is T?**

--Arithmatic operator

--these operators used to perform mathematical operation like +,-,\*,/ and %

select \* from employee

select \*,MonthlyIcrement =salary+1000 from employee

**Q. Find out loss of pay for 2 days .**

**--2.order by**

--This clause is used to sort the result in ascending (ASC) or Descending(DESC) order.

--If the column contains NULL value in it and if we are performing order by operation then NULL value should be first in ASC and Last in DESC.

select \* from employee order by salary -- this is by default ascending order

select \* from employee order by salary ASC

select \* from employee order by salary desc

select \* from employee order by LastName desc

select \* from employee order by LastName

insert into employee values(11,'Meena','','Hyderabad','HR','')

insert into employee (EID,FirstName,LOc) values(12,'Roshan','Yavatmal')

select \* from employee order by FirstName asc

**--NULL Values**

--A column with a NULL value is column with NO value

--NULL valu is diffrent from 0(zero) and blank/empty sapce.

select \* from employee where salary = NULL

--Q.How to test the NULL values from column?

--There are two ways to check the NULL values from column

--1.IS NULL

--2.IS NOT NULL

select \* from employee where salary = NULL -- Blank /not possible to check by using comaprision/logical/arithmatic opeartor

select \* from employee where salary is NULL

select \* from employee where salary is not NULL

**--2.DML(Data Manipulation Language)**

**--UPDATE**

--Update statement is used to update complete column data or specific record if condition is provided.

-- By using update statement you can only play with table data.

--synatx:

--UPDATE TABLE\_NAME SET COLUMN\_NAME ='VALUE' where COLUMN\_NAME ='CONDITION'

create table UPDATE\_DELETE (U\_ID int, UNAME varchar(20) ,ULOC varchar(20))

insert into UPDATE\_DELETE values (1,'Sagar','PUNE')

insert into UPDATE\_DELETE values (2,'Amit','Sangli')

insert into UPDATE\_DELETE values (3,'Sarika','Bijapur')

insert into UPDATE\_DELETE values (4,'Rohan','Mumbai')

insert into UPDATE\_DELETE values (5,'Amrita','Palampur')

select \* from UPDATE\_DELETE

update UPDATE\_DELETE SET ULOC ='Pune' where U\_ID >=2

update UPDATE\_DELETE SET ULOC ='Jaipur' where U\_ID =5

update UPDATE\_DELETE SET UNAME ='Sohan' where U\_ID =4

**--DELETE**

--Delete statement is used to delete the data from table row by row.

--By using DELETE statement it is not possible to delete the structure.

--We can delete the table data at one time or row by row by specifying an condition.

--syntax:

--DELETE TABLE\_NAME where COULMN\_NAME ='CONDITION'

select \* from UPDATE\_DELETE

delete UPDATE\_DELETE -- it will delete the complete data from table.

delete UPDATE\_DELETE where U\_ID =5

delete UPDATE\_DELETE where U\_ID <=2

**--1.Data Defination Language(DDL) - DR.CAT**

--Along with DDL statements "TABLE" Keyword is mandotory.

**--DROP**

--DROP statement will delete the table structure as well as table data.

--Drop statement we can drop or delete the database.

--syntax:

--DROP TABLE TABLE\_NAME

--DROP DATABASE DATABASE\_NAME

DROP table UPDATE\_DELETE -- it will delete table data as well as table structutre.

--Q.Diffrence between Delete and Drop?

select \* from employee where firstname like '%[\_]%'

select \* from employee where FIRSTname like '%#\_%'escape'#'

--'%[rt]'

**--Truncate**

--Truncate statement allow you to delete the records from a table at once.

--It wont delete the structure of the table

--In Truncate you can't delete the data Row-By-Row by specifying a condtion.

--syntax : truncate table table\_name

create table Truncate1 (U\_ID int, UNAME varchar(20) ,ULOC varchar(20))

insert into Truncate1 values (1,'Sagar','PUNE')

insert into Truncate1 values (2,'Amit','Sangli')

insert into Truncate1 values (3,'Sarika','Bijapur')

insert into Truncate1 values (4,'Rohan','Mumbai')

insert into Truncate1 values (5,'Amrita','Palampur')

select \* from Truncate1

truncate table truncate1

--Q. What is the diffrence between Delete,Drop and Truncate?

--Q. What is the diffrence between DML,and DDL statements?

--Q.How will you delete the data from a table at once?