

## Assignment - 04

- \* Date of completion: 31<sup>st</sup> / August / 2020
- \* Date of submission: 14<sup>th</sup> / September / 2020
- \* Title: Design at least 10 SQL queries for suitable database, application using SQL DML statements: All types of join, sub-query & view.
- \* Problem Statement:  
Design at least 10 SQL queries for suitable database application using SQL DML statements; All types of joins, sub-query & view.
- \* Objective: To understand:
  - Types of joins
  - subquery & its types
  - complex views & simple view.
- \* Outcome: Students will be able to
  - identify & implement types of joins, subquery & view.
  - implementation & updation of simple view.
- \* SW And HW Requirements:-  
MySQL, Windows 10 (64-bit), i5 processor.



## \* THEORY:

### - JOIN:

SQL join is used to fetch data from two or more tables, which is joined to appear as single set of data. SQL join is used for combining column from two or more tables by using values common to both tables.

### TYPES OF JOIN:

#### ° CROSS JOIN:

- this join returns the cartesian product of rows from tables in join. It will return a table which consists of recorder which combines each row from first table to each row of second table.

Syntax:

```
Select column list from table1
CROSS JOIN
table2 ;
```

#### ° INNER JOIN: This is a simple JOIN in which the result is based on matched data as per equality condition specified in the query.

Syntax:

```
Select columnlist from table1 inner join table2 where
table1.colname = table2.colname;
```

#### ° NATURAL JOIN: It is a type of inner join which is based on column having same name & same datatype



present in both tables to be joined.

Syntax:

Select \* from table1 NATURAL JOIN table2;

- OUTER JOIN: It is based on both matched and unmatched data. Outer join subdivides further into.

i> Left Outer join: returns a result table with matched data of two variables table then remaining rows of the left table & null for the right table's column.

Syntax:

Select columnlist from table1 left outer join table2 on table1.col = table2.col;

ii> Right Outer join: returns a result table with matched data columns of two tables & then remaining rows of right table and null for the left table's column.

Syntax:

Select columnlist from table1 right outer join table2 on table1.col = table2.col;

iii> Full Outer join: returns a result table with the matched data of 2 tables & then remaining rows of both left table & right table.

Syntax: (Only supported by ORACLE)

Select columnlist from table1 full outer join table2 on table1.col = table2.col;



### SELF JOIN:

- The table is joined with itself.

Syntax:

select columnlist from table1 T1, table2 T2  
where condition;

### COUNT JOIN:

- GROUP BY

- it groups rows that have same values
- it often used with aggregate functions like count, max, min, sum, avg, etc.

Syntax:

select columnlist, count(col) from table1 join table2  
where condition group by columns;

### SUB QUERY:

- an SQL query nested inside a larger query
- A subquery may occur in
  - a select clause
  - a where clause
  - a from clause

Syntax:

select columnlist from tablelist where  
condition is (select columnlist from tablelist);

### COMPLEX VIEW:

- view created using more than one table.

Syntax:

create view view name as select columnlist from  
tableview1 join table2 using (column);



## ◦ SIMPLE VIEW :

Syntax:

create view view-name as select col1, col2 ...  
from table-name where condition;

## - Inserting into a view (simple):

Syntax:

insert into view-name (cols) values (val1, val2, val3);

## - Updating a view (simple)

Syntax:

update view-name set col-name = value  
where condition;

## - deleting from a view (simple):

Syntax:

delete from view-name where condition;

## - dropping a view:

drop view view-name;

## ◦ MULTITABLE JOIN :

Syntax:

select columnlist from table1 inner join table2  
using / on inner join table3 using (col) where  
condition;

Simple views can be updated / modified but  
complex views cannot be updated / modified

### o HAVING clause:

- added because where keyword cannot be used with aggregate functions

Syntax:

select col from table1 where condition group by col(s)  
having condition group by column(s)

### \* Conclusion:-

- Thus we implemented various types of joins in SQL
- we used subquery in SQL.
- we implemented complex view in SQL
- we modified a simple view in SQL