**DATABASE MANAGEMENT**

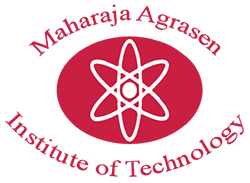
**SYSTEM**

**LAB MANUAL**

Submitted to: – Mrs.Neelam Sharma Name – Pankaj kumar

Roll No – 07414802718

Semester - 4th(4C4)



Maharaja Agrasen Institute of Technology, PSP Area,

Sector – 22, Rohini, New Delhi – 110085

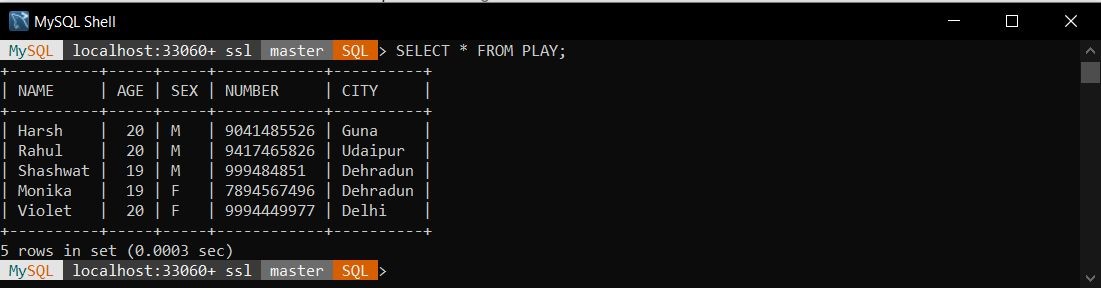
*PLEASE NOTE-  
  
Ma’am,I had already performed and checked 5 experiments before the lock down, which were written in my lab file.*

*Thanks!!*

## Experiment – 6

**Aim:**​ Write the queries for implementing the following functions: MAX (), MIN (), AVG (), COUNT () and other logical and pattern matching operations.

**Tool Used:**​ Maria DB **Database:**



### The SQL ANY and ALL Operators

The ANY and ALL operators are used with a WHERE or HAVING clause. The ANY operator returns true if any of the subquery values meet the condition. The ALL operator returns true if all of the subquery values meet the condition.

### Syntax: ALL

SELECT​ column\_name(s)

FROM​ table\_name

WHERE​ column\_name ​operator​ ALL

(​SELECT​ column\_name ​FROM​ table\_name ​WHERE​ condition);

### Syntax: ANY

SELECT​ column\_name(s)

FROM​ table\_name

WHERE​ column\_name ​operator​ ​ANY

(​SELECT​ column\_name ​FROM​ table\_name ​WHERE​ condition);

### The SQL AND, OR and NOT Operators

The WHERE clause can be combined with AND, OR, and NOT operators. The AND and OR operators are used to filter records based on more than one condition: The AND operator displays a record if all the conditions separated by AND is TRUE. The OR operator displays a record if any of the conditions separated by OR is TRUE. The NOT operator displays a record if the condition(s) is NOT TRUE.

### AND Syntax

SELECT​ column1, column2, ...

FROM​ table\_name

WHERE​ condition1 ​AND​ condition2 ​AND​ condition3 ...;

### NOT Syntax

SELECT​ column1, column2, ...

FROM​ table\_name

WHERE​ ​NOT​ condition;

### OR Syntax

SELECT​ column1, column2, ...

FROM​ table\_name

WHERE​ condition1 ​OR condition2 ​ ​OR​ condition3 ...;

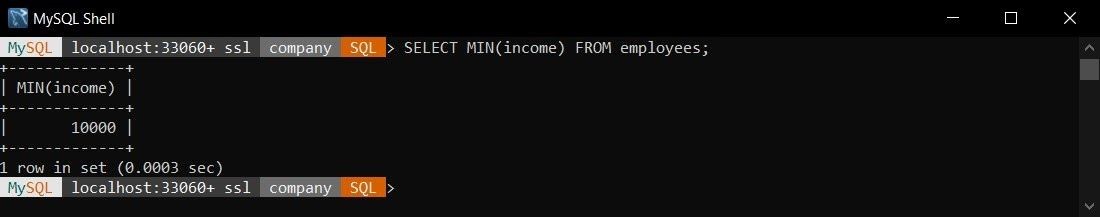
**MIN() Syntax:**

The MIN() function returns the smallest value of the selected column.

SELECT​ ​MIN​(column\_name)

FROM​ table\_name

WHERE​ condition;



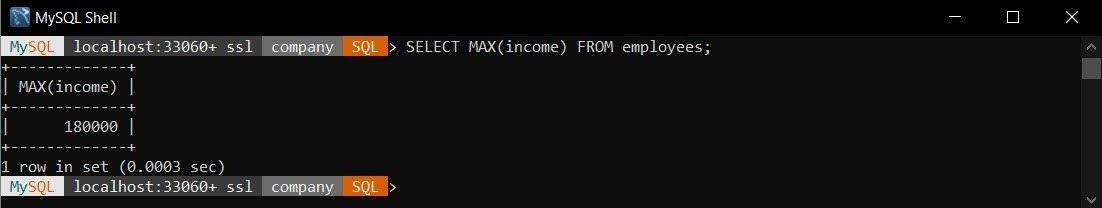
**MAX() Syntax:**

The MAX() function returns the largest value of the selected column.

SELECT​ ​MIN​(column\_name)

FROM​ table\_name

WHERE​ condition;



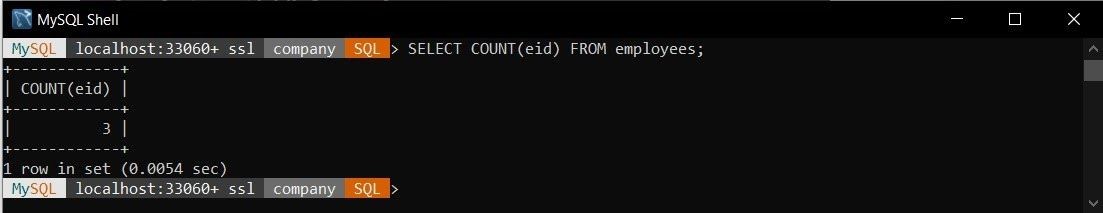
**COUNT() Syntax:**

The COUNT() function returns the number of rows that matches a specified criteria.

SELECT​ ​COUNT​(column\_name)

FROM​ table\_name

WHERE​ condition;



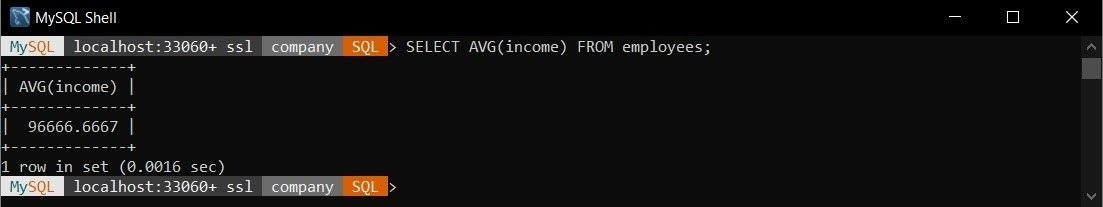
**AVG() Syntax:**

The AVG() function returns the average value of a numeric column.

SELECT​ ​AVG​(column\_name)

FROM​ table\_name

WHERE​ condition;



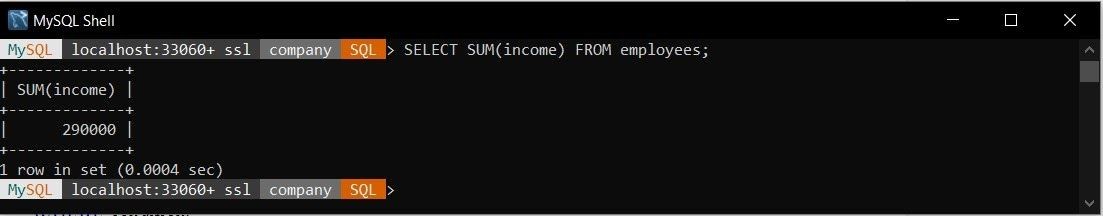
**SUM() Syntax:**

The SUM() function returns the total sum of a numeric column.

SELECT​ ​SUM​(column\_name)

F​ROM​ table\_name

WHERE​ condition;



## Experiment – 7

**Aim:** Write the queries to implement the month and date command.​

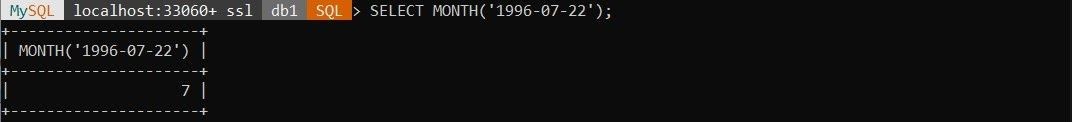
### Tool Used: Maria DB​

**Procedure:** In this experiment, we had focused on commands to implement the month​ and date commands. We will be applying these commands to the records we already have in previous experiments.

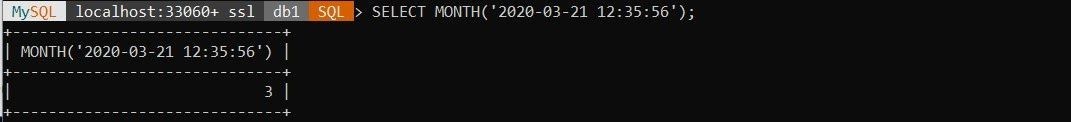
**Syntax:**

|  |
| --- |
| MONTH :  ​SELECT​ ​MONTH​(date​ ​ ​or​ timestamp​ ​); DATE :  ​SELECT​ ​DATE​(​date​ or​ ​ ​timestamp​); CURTIME :  ​SELECT​ ​CURTIME​(); CURDATE :  ​SELECT​ ​CURDATE​(); NOW :  ​SELECT​ ​NOW​(); DATE FORMAT :  ​SELECT​ ​DATE\_FORMAT​(​timestamp​, ​format​); |

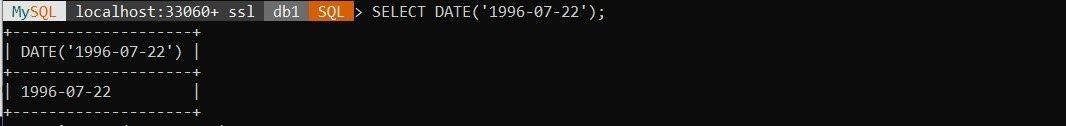
1. Get the month from date “1996-07-22”;



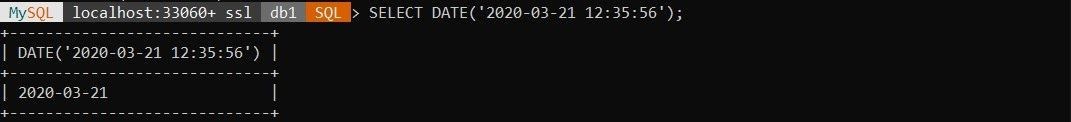
1. Get the month from date “2020-03-21 12:35:56”;



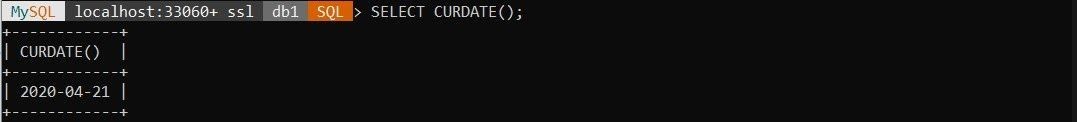
1. Create the column sub\_id as Primary Key.



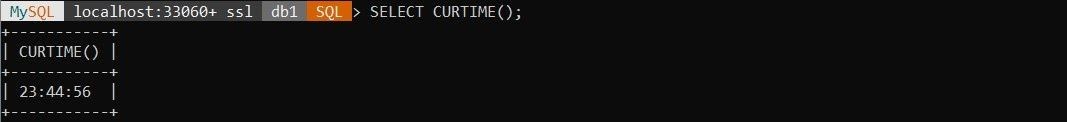
1. Change the constraint of dept\_id to make it a Foreign Key.



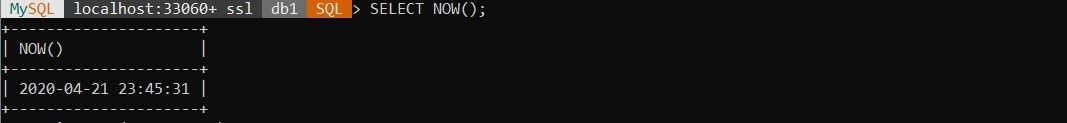
1. Get the current Date.



1. Get the current Time.



1. Get the current timestamp.



## Experiment – 8

**Aim:** Write the queries to implement Having and Group by clause on​ the table.

**Tool Used:** Maria DB​

**Procedure:** In this experiment, we had focused on commands to​ implement the clauses like Having and Group By. We will be applying these commands to the records we already have in previous experiments.

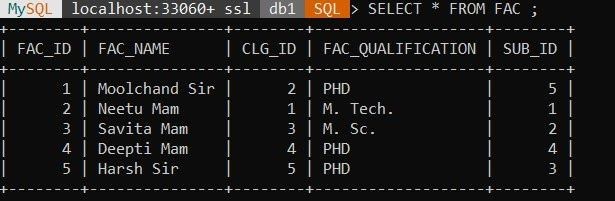
**Syntax:**

SELECT column\_name

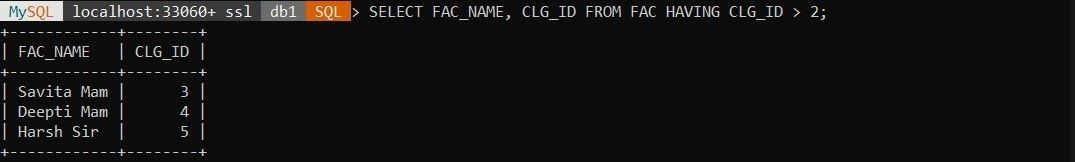
FROM table\_name

GROUP BY column\_name

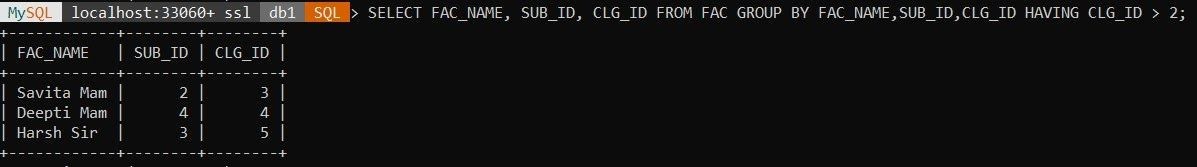
HAVING condition; **Table:**



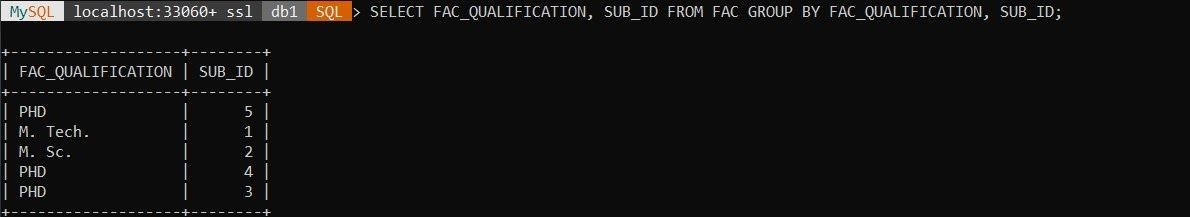
1. Get names of all the teachers having college id > 2.



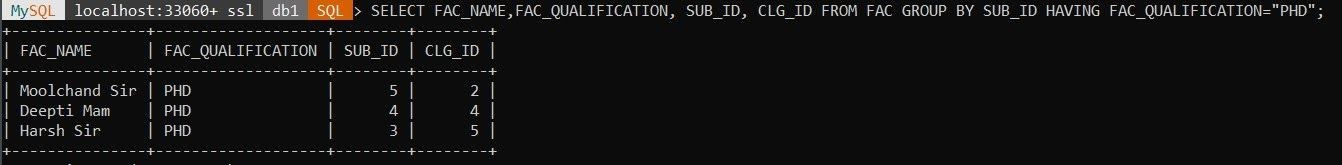
1. Get names of all the teachers having college id > 2 and group them by subject id.



1. Get the qualification of all the teachers and group them by their subject id.



1. Get the details of teachers having qualification = “PHD.” and group by subject id.



**Experiment – 9**

**SQL Commands:**

CREATE TABLE Stud (

PersonID int,

Name varchar(255),

Marks int

);

INSERT INTO Stud VALUES(1,"Yashika",91);

INSERT INTO Stud VALUES(2,"Rishabh",81);

INSERT INTO Stud VALUES(3,"Reena",67);

INSERT INTO Stud VALUES(4,"Mohit",94);

INSERT INTO Stud VALUES(5,"Nikita",50);

CREATE VIEW Stud\_View AS

SELECT Name, Marks

FROM Stud;

SELECT \* FROM Stud\_View;

