**Name: Akhilesh Rawat**

**Roll No.- 7307**

**DBMS Lab**

**Assignment No. 2(MySQL Basic Queries)**

**Title:** Design at least 10 SQL queries for suitable database application using SQL DML statements: Insert, Select, Update, Delete with operators, functions, and set operator.

* **Create Employee table, Project table and add rows shown below**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Eid** | **EName** | **Address** | **Salary** | **Commision** |
| **1** | Amit | Pune | 35000 | 5000 |
| **2** | Sneha | Pune | 25000 |  |
| **3** | Savita | Nasik | 28000 | 2000 |
| **4** | Pooja | Mumbai | 19000 |  |
| **5** | Sagar | Mumbai | 25000 | 3000 |
| **6** | Rohit | Jaipur | 40000 |  |
| **7** | Poonam | Patana | 45000 | 2000 |
| **8** | Arjun | Delhi | 20000 | 900 |
| **9** | Rahul | Nagpur | 60000 | 5000 |
| **10** | Dulquer | Kochi | 30000 | 1000 |

|  |  |
| --- | --- |
| **PrNo** | **Addr** |
| 10 | Mumbai |
| 20 | Pune |
| 30 | Jalgoan |
| 40 | Nagpur |
| 50 | Delhi |
| 60 | Kochi |
| 70 | Pune |
| 80 | Nasik |

mysql> create table employee (Ename varchar(30),address varchar(30),salary int,commision int);

Query OK, 0 rows affected (0.38 sec)

mysql> insert into employee values('Sneha','Pune',25000,NULL);

Query OK, 1 row affected (0.09 sec)

(*Like this insert all values in both the tables and the result will be the following)*

mysql> select \* from employee;

+---------+----------+--------+-----------+

| Ename | address | salary | commision |

+---------+----------+--------+-----------+

| Amit | Pune | 35000 | 5000 |

| Sneha | Pune | 25000 | NULL |

| Savita | Nasik | 28000 | 2000 |

| Pooja | Mumbai | 19000 | NULL |

| Sagar | Mumbai | 25000 | 3000 |

| Rohit | Jaipur | 40000 | NULL |

| Poonam | Patna | 45000 | 2000 |

| Arjun | Delhi | 20000 | 900 |

| Rahul | Nagpur | 60000 | 5000 |

| Sheetal | Dehradun | 30000 | 1000 |

+---------+----------+--------+-----------+

*10 rows in set (0.00 sec)*

mysql> select \* from project;

+------+----------+

| Prno | Addr |

+------+----------+

| 10 | Mumbai |

| 20 | Pune |

| 30 | Jalagon |

| 40 | Nagpur |

| 50 | Delhi |

| 60 | Dehradun |

| 70 | Pune |

| 80 | Nasik |

+------+----------+

*8 rows in set (0.00 sec)*

* **Execute the following queries in MySQL:**

1. Find different locations from where employees belong to?

mysql> select address from employee;

+----------+

| address |

+----------+

| Pune |

| Pune |

| Nasik |

| Mumbai |

| Mumbai |

| Jaipur |

| Patna |

| Delhi |

| Nagpur |

| Dehradun |

+----------+

10 rows in set (0.00 sec)

1. What are maximum ,minimum salary, average salary and sum of all salaries?

mysql> select max(salary) from employee;

+-------------+

| max(salary) |

+-------------+

| 60000 |

+-------------+

mysql> select min(salary) as lowsal from employee;

+--------+

| lowsal |

+--------+

| 19000 |

+--------+

mysql> select avg(salary) from employee;

+-------------+

| avg(salary) |

+-------------+

| 32700.0000 |

+-------------+

mysql> select sum(salary) from employee;

+-------------+

| sum(salary) |

+-------------+

| 327000 |

+-------------+

1 row in set (0.00 sec)

1. Display the content of employee table according to the ascending order of salary amount.

mysql> select \* from employee order by salary asc;

+---------+----------+--------+-----------+

| Ename | address | salary | commision |

+---------+----------+--------+-----------+

| Pooja | Mumbai | 19000 | NULL |

| Arjun | Delhi | 20000 | 900 |

| Sneha | Pune | 25000 | NULL |

| Sagar | Mumbai | 25000 | 3000 |

| Savita | Nasik | 28000 | 2000 |

| Sheetal | Dehradun | 30000 | 1000 |

| Amit | Pune | 35000 | 5000 |

| Rohit | Jaipur | 40000 | NULL |

| Poonam | Patna | 45000 | 2000 |

| Rahul | Nagpur | 60000 | 5000 |

+---------+----------+--------+-----------+

10 rows in set (0.00 sec)

1. Find the name of employee who lived in Nasik or Pune city.

mysql> select ename from employee where address='Pune' or address='Nasik';

+--------+

| ename |

+--------+

| Amit |

| Sneha |

| Savita |

+--------+

3 rows in set (0.00 sec)

1. Find the name of employees who does not get commission.

mysql> select ename from employee where commision is NULL;

+-------+

| ename |

+-------+

| Sneha |

| Pooja |

| Rohit |

+-------+

3 rows in set (0.00 sec)

1. Change the city of Amit to Nashik.

mysql> update employee set address='Nasik' where Ename='Amit';

Query OK, 1 row affected (0.09 sec)

Rows matched: 1 Changed: 1 Warnings: 0

mysql> select \* from employee;

+---------+----------+--------+-----------+

| Ename | address | salary | commision |

+---------+----------+--------+-----------+

| Amit | Nasik | 35000 | 5000 |

| Sneha | Pune | 25000 | NULL |

| Savita | Nasik | 28000 | 2000 |

| Pooja | Mumbai | 19000 | NULL |

| Sagar | Mumbai | 25000 | 3000 |

| Rohit | Jaipur | 40000 | NULL |

| Poonam | Patna | 45000 | 2000 |

| Arjun | Delhi | 20000 | 900 |

| Rahul | Nagpur | 60000 | 5000 |

| Sheetal | Dehradun | 30000 | 1000 |

+---------+----------+--------+-----------+

10 rows in set (0.00 sec)

1. Find the information of employees whose name starts with ‘A’.

mysql> select ename from employee where ename like 'A%';

+-------+

| ename |

+-------+

| Amit |

| Arjun |

+-------+

2 rows in set (0.00 sec)

1. Find the count of staff from Mumbai.

mysql> select count(address) from employee where address='Mumbai';

+----------------+

| count(address) |

+----------------+

| 2 |

+----------------+

1 row in set (0.00 sec)

1. Find the count of staff from each city

mysql> select address,count(\*) from employee group by address;

+----------+----------+

| address | count(\*) |

+----------+----------+

| Nasik | 2 |

| Pune | 1 |

| Mumbai | 2 |

| Jaipur | 1 |

| Patna | 1 |

| Delhi | 1 |

| Nagpur | 1 |

| Dehradun | 1 |

+----------+----------+

8 rows in set (0.00 sec)

1. Find the address from where employees are belonging as well as where projects are going on.

(Use union operator)

mysql> select address from employee union select addr from project;

+----------+

| address |

+----------+

| Nasik |

| Pune |

| Mumbai |

| Jaipur |

| Patna |

| Delhi |

| Nagpur |

| Dehradun |

| Jalagon |

+----------+

9 rows in set (0.01 sec)

1. Find city wise minimum salary.

mysql> select address,min(salary) from employee group by address;

+----------+-------------+

| address | min(salary) |

+----------+-------------+

| Pune | 25000 |

| Nasik | 28000 |

| Mumbai | 19000 |

| Jaipur | 40000 |

| Patna | 45000 |

| Nagpur | 60000 |

| Dehradun | 30000 |

| Delhi | 20000 |

+----------+-------------+

8 rows in set (0.00 sec)

1. Find city wise maximum salary having maximum salary greater than 26000

mysql> select address,max(salary) from employee where salary>=26000 group by address;

+----------+-------------+

| address | max(salary) |

+----------+-------------+

| Nasik | 35000 |

| Jaipur | 40000 |

| Patna | 45000 |

| Nagpur | 60000 |

| Dehradun | 30000 |

+----------+-------------+

5 rows in set (0.00 sec)

1. Delete the employee who is having salary greater than 30,000.

mysql> delete from employee where salary>30000;

Query OK, 4 rows affected (0.06 sec)

mysql> select \* from employee;

+---------+----------+--------+-----------+

| Ename | address | salary | commision |

+---------+----------+--------+-----------+

| Sneha | Pune | 25000 | NULL |

| Savita | Nasik | 28000 | 2000 |

| Pooja | Mumbai | 19000 | NULL |

| Sagar | Mumbai | 25000 | 3000 |

| Arjun | Delhi | 20000 | 900 |

| Sheetal | Dehradun | 30000 | 1000 |

+---------+----------+--------+-----------+

6 rows in set (0.00 sec)

1. Delete the information of employees whose name starts with‘s’.

mysql> delete from employee where Ename like 'S%';

Query OK, 4 rows affected (0.07 sec)

mysql> select \* from employee;

+-------+---------+--------+-----------+

| Ename | address | salary | commision |

+-------+---------+--------+-----------+

| Pooja | Mumbai | 19000 | NULL |

| Arjun | Delhi | 20000 | 900 |

+-------+---------+--------+-----------+

2 rows in set (0.00 sec)

1. Display all the employee from Pune alphabetically.

mysql> select ename from employee where address='Pune' order by address;

+-------+

| ename |

+-------+

| Amit |

| Sneha |

+-------+

2 rows in set (0.00 sec)