**SEHEJ BAKSHI DBMS Lab 24/07/2020**

**Assignment No. 2 (My SQL 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 | Patna | 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

-> ( eid int primary key,

-> ename varchar(20) not null,

-> address char(20) not null,

-> salary int not null,

-> commision int

-> );

Query OK, 0 rows affected (1.17 sec)

mysql> create table Project

-> ( prNo int,

-> addr char(20) not null,

-> primary key(prNo)

-> );

Query OK, 0 rows affected (0.38 sec)

mysql> alter table Employee add prNo int not null

-> ;

Query OK, 0 rows affected (1.20 sec)

Records: 0 Duplicates: 0 Warnings: 0

mysql> alter table Employee

-> add foreign key(prNo) references Project(prNo);

Query OK, 0 rows affected (1.97 sec)

Records: 0 Duplicates: 0 Warnings: 0

mysql> insert into Project(prNo, addr) values

-> (10, 'Mumbai'), (20, 'Pune'), (30, 'Jalgaon'), (40, 'Nagpur'),

-> (50, 'Delhi'), (60, 'Kochi'), (70, 'Pune'), (80, 'Nasik');

Query OK, 8 rows affected (0.29 sec)

Records: 8 Duplicates: 0 Warnings: 0

mysql> insert into Employee(eid, ename, address, salary, commision, prNo) values

-> (1, 'Amit', 'Pune', 35000, 5000, 10),

-> (2, 'Sneha', 'Pune', 25000, NULL, 30),

-> (3, 'Savita', 'Nasik', 28000, 2000, 50),

-> (4, 'Pooja', 'Mumbai', 19000, NULL, 80),

-> (5, 'Sagar', 'Mumbai', 25000, 3000, 20),

-> (6, 'Rohit', 'Jaipur', 40000, NULL, 40),

-> (7, 'Poonam', 'Patna', 45000, 2000, 70),

-> (8, 'Arjun', 'Delhi', 20000, 900, 10),

-> (9, 'Rahul', 'Nagpur', 60000, 5000, 60),

-> (10, 'Dulquer', 'Kochi', 30000, 1000, 50);

Query OK, 10 rows affected (0.22 sec)

Records: 10 Duplicates: 0 Warnings: 0

**Execute the following queries in MySQL:**

1. **Find different locations where employees belong to?**

mysql> select distinct address

-> from Employee;

+---------+

| address |

+---------+

| Pune |

| Nasik |

| Mumbai |

| Jaipur |

| Patna |

| Delhi |

| Nagpur |

| Kochi |

+---------+

8 rows in set (0.05 sec)

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

mysql> select

-> max(salary) as max,

-> min(salary) as min,

-> avg(salary) as avg,

-> sum(salary) as total\_sum

-> from Employee;

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

| max | min | avg | total\_sum |

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

| 60000 | 19000 | 32700.0000 | 327000 |

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

1 row in set (0.04 sec)

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

mysql> select \*

-> from Employee

-> order by salary asc;

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

| eid | ename | address | salary | commision | prNo |

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

| 4 | Pooja | Mumbai | 19000 | NULL | 80 |

| 8 | Arjun | Delhi | 20000 | 900 | 10 |

| 2 | Sneha | Pune | 25000 | NULL | 30 |

| 5 | Sagar | Mumbai | 25000 | 3000 | 20 |

| 3 | Savita | Nasik | 28000 | 2000 | 50 |

| 10 | Dulquer | Kochi | 30000 | 1000 | 50 |

| 1 | Amit | Pune | 35000 | 5000 | 10 |

| 6 | Rohit | Jaipur | 40000 | NULL | 40 |

| 7 | Poonam | Patna | 45000 | 2000 | 70 |

| 9 | Rahul | Nagpur | 60000 | 5000 | 60 |

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

10 rows in set (0.00 sec)

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

mysql> select ename, address

-> from Employee

-> where address in('Nasik', 'Pune');

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

| ename | address |

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

| Amit | Pune |

| Sneha | Pune |

| Savita | Nasik |

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

3 rows in set (0.00 sec)

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

mysql> select ename, commision

-> from Employee

-> where commision IS NULL;

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

| ename | commision |

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

| Sneha | NULL |

| Pooja | NULL |

| Rohit | NULL |

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

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 (0.13 sec)

Rows matched: 1 Changed: 1 Warnings: 0

mysql> select ename, address

-> from Employee;

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

| ename | address |

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

| Amit | Nasik |

| Sneha | Pune |

| Savita | Nasik |

| Pooja | Mumbai |

| Sagar | Mumbai |

| Rohit | Jaipur |

| Poonam | Patna |

| Arjun | Delhi |

| Rahul | Nagpur |

| Dulquer | Kochi |

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

10 rows in set (0.00 sec)

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

mysql> select \*

-> from Employee

-> where ename like 'A%';

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

| eid | ename | address | salary | commision | prNo |

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

| 1 | Amit | Nasik | 35000 | 5000 | 10 |

| 8 | Arjun | Delhi | 20000 | 900 | 10 |

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

2 rows in set (0.07 sec)

1. **Find the count of staff from Mumbai.**

mysql> select count(\*) as staff, address

-> from Employee

-> where address = 'Mumbai';

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

| staff | address |

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

| 2 | Mumbai |

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

1 row in set (0.03 sec)

1. **Find the count of staff from each city**

mysql> select count(\*) as staff, address

-> from Employee

-> group by address;

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

| staff | address |

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

| 2 | Nasik |

| 1 | Pune |

| 2 | Mumbai |

| 1 | Jaipur |

| 1 | Patna |

| 1 | Delhi |

| 1 | Nagpur |

| 1 | Kochi |

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

8 rows in set (0.00 sec)

1. **Find city wise minimum salary.**

mysql> select min(salary), address

-> from Employee

-> group by address;

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

| min(salary) | address |

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

| 28000 | Nasik |

| 25000 | Pune |

| 19000 | Mumbai |

| 40000 | Jaipur |

| 45000 | Patna |

| 20000 | Delhi |

| 60000 | Nagpur |

| 30000 | Kochi |

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

8 rows in set (0.00 sec)

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

mysql> select max(salary), address

-> from Employee

-> where salary > 26000

-> group by address;

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

| max(salary) | address |

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

| 35000 | Nasik |

| 40000 | Jaipur |

| 45000 | Patna |

| 60000 | Nagpur |

| 30000 | Kochi |

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

5 rows in set (0.00 sec)

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

mysql> select \*

-> from Employee;

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

| eid | ename | address | salary | commision | prNo |

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

| 1 | Amit | Nasik | 35000 | 5000 | 10 |

| 2 | Sneha | Pune | 25000 | NULL | 30 |

| 3 | Savita | Nasik | 28000 | 2000 | 50 |

| 4 | Pooja | Mumbai | 19000 | NULL | 80 |

| 5 | Sagar | Mumbai | 25000 | 3000 | 20 |

| 6 | Rohit | Jaipur | 40000 | NULL | 40 |

| 7 | Poonam | Patna | 45000 | 2000 | 70 |

| 8 | Arjun | Delhi | 20000 | 900 | 10 |

| 9 | Rahul | Nagpur | 60000 | 5000 | 60 |

| 10 | Dulquer | Kochi | 30000 | 1000 | 50 |

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

10 rows in set (0.00 sec)

mysql> delete from Employee where salary > 30000;

Query OK, 4 rows affected (0.28 sec)

mysql> select \*

-> from Employee;

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

| eid | ename | address | salary | commision | prNo |

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

| 2 | Sneha | Pune | 25000 | NULL | 30 |

| 3 | Savita | Nasik | 28000 | 2000 | 50 |

| 4 | Pooja | Mumbai | 19000 | NULL | 80 |

| 5 | Sagar | Mumbai | 25000 | 3000 | 20 |

| 8 | Arjun | Delhi | 20000 | 900 | 10 |

| 10 | Dulquer | Kochi | 30000 | 1000 | 50 |

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

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 (0.19 sec)

mysql> select \* from Employee;

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

| eid | ename | address | salary | commision | prNo |

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

| 4 | Pooja | Mumbai | 19000 | NULL | 80 |

| 8 | Arjun | Delhi | 20000 | 900 | 10 |

| 10 | Dulquer | Kochi | 30000 | 1000 | 50 |

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

3 rows in set (0.00 sec)

1. **Display all the employee from Pune alphabetically.**

mysql> insert into Employee(eid, ename, address, salary, commision, prNo) values

-> (11, 'Sehej', 'Pune', 60000, 5000, 60);

Query OK, 1 row affected (0.11 sec)

mysql> select \*

-> from Employee

-> where address = 'Pune'

-> order by ename;

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

| eid | ename | address | salary | commision | prNo |

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

| 11 | Sehej | Pune | 60000 | 5000 | 60 |

| 2 | Sneha | Pune | 25000 | NULL | 30 |

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

2 rows in set (0.01 sec)