Assignment-4

Submitted by: Tanmay Vig

Roll Number: 19BCS061

Batch: 2nd year CSE

Create the table employee and solve the following queries.

```
+--------+----------+--------+-----------+
| emp_id | emp_name | salary | dept_name |
+--------+----------+--------+-----------+
|    101 | Amit     |  25000 | IT        |
|    102 | Sunil    |  20000 | Sales     |
|    103 | Rakesh   |  18000 | MKTG      |
|    104 | Ajay     |  16000 | IT        |
|    105 | Suhail   |  20000 | Sales     |
|    106 | Arif     |  18000 | HR        |
|    107 | Suresh   |  24000 | Sales     |
|    108 | Vijay    |  22000 | MKTG      |
+--------+----------+--------+-----------+
```

Query for creating the above table:

```
mysql> CREATE TABLE employee(
    -> emp_id int,
    -> emp_name varchar(20),
    -> salary int,
    -> dept_name varchar(20));
Query OK, 0 rows affected (0.11 sec)

mysql> DESC employee;
+-----------+-------------+------+-----+---------+-------+
| Field     | Type        | Null | Key | Default | Extra |
+-----------+-------------+------+-----+---------+-------+
| emp_id    | int         | YES  |     | NULL    |       |
| emp_name  | varchar(20) | YES  |     | NULL    |       |
| salary    | int         | YES  |     | NULL    |       |
| dept_name | varchar(20) | YES  |     | NULL    |       |
+-----------+-------------+------+-----+---------+-------+
4 rows in set (0.03 sec)

mysql> INSERT INTO employee VALUES
    -> (101, 'Amit',25000,'IT'),
    -> (102,'Sunil',20000,'Sales'),
    -> (103,'Rakesh',18000,'MKTG'),
    -> (104,'Ajay',16000,'IT'),
```

```
    -> (105,'Suhail',20000,'Sales'),
    -> (106,'Arif',18000,'HR'),
    -> (107,'Suresh',24000,'Sales'),
    -> (108,'Vijay',22000,'MKTG');
Query OK, 8 rows affected (0.02 sec)
Records: 8  Duplicates: 0  Warnings: 0

mysql> SELECT * FROM employee;
+--------+----------+--------+-----------+
| emp_id | emp_name | salary | dept_name |
+--------+----------+--------+-----------+
|    101 | Amit     |  25000 | IT        |
|    102 | Sunil    |  20000 | Sales     |
|    103 | Rakesh   |  18000 | MKTG      |
|    104 | Ajay     |  16000 | IT        |
|    105 | Suhail   |  20000 | Sales     |
|    106 | Arif     |  18000 | HR        |
|    107 | Suresh   |  24000 | Sales     |
|    108 | Vijay    |  22000 | MKTG      |
+--------+----------+--------+-----------+
8 rows in set (0.01 sec)
```

1. Display total sum required to pay the salary of all employees.

Answer:

```
mysql> SELECT SUM(salary) AS Total_Salary
    -> FROM employee;
+--------------+
| Total_Salary |
+--------------+
|       163000 |
+--------------+
1 row in set (0.01 sec)
```

2. Display the average salary, minimum salary, and maximum salary of the company.

Answer:

```
mysql> SELECT AVG(salary), MAX(salary), MIN(salary)
```

```
    -> FROM employee;
+-------------+-------------+-------------+
| AVG(salary) | MAX(salary) | MIN(salary) |
+-------------+-------------+-------------+
|  20375.0000 |       25000 |       16000 |
+-------------+-------------+-------------+
1 row in set (0.00 sec)
```

3. Display the sum of salary department-wise.

Answer:

```
mysql> SELECT dept_name, SUM(salary)
    -> FROM employee
    -> GROUP BY dept_name;
+-----------+-------------+
| dept_name | SUM(salary) |
+-----------+-------------+
| IT        |       41000 |
| Sales     |       64000 |
| MKTG      |       40000 |
| HR        |       18000 |
+-----------+-------------+
4 rows in set (0.00 sec)
```

4. Display the maximum salary department-wise.

Answer:

```
mysql> SELECT dept_name, MAX(salary)
    -> FROM employee
    -> GROUP BY dept_name;
+-----------+-------------+
| dept_name | MAX(salary) |
+-----------+-------------+
| IT        |       25000 |
| Sales     |       24000 |
| MKTG      |       22000 |
| HR        |       18000 |
+-----------+-------------+
```

```
4 rows in set (0.00 sec)
```

5.a. Display the details of the employee who earns the maximum salary.

Answer:

```
mysql> SELECT * FROM employee
    -> HAVING MAX(salary);
+--------+----------+--------+-----------+
| emp_id | emp_name | salary | dept_name |
+--------+----------+--------+-----------+
|    101 | Amit     |  25000 | IT        |
+--------+----------+--------+-----------+
1 row in set (0.00 sec)
```

5.b. Display details of every employee having maximum salary in his department.

Answer:

```
mysql> SELECT * FROM employee a
    -> WHERE salary = ( SELECT MAX(salary) FROM employee b
    -> GROUP BY dept_name
    -> HAVING b.dept_name=a.dept_name);
+--------+----------+--------+-----------+
| emp_id | emp_name | salary | dept_name |
+--------+----------+--------+-----------+
|    101 | Amit     |  25000 | IT        |
|    106 | Arif     |  18000 | HR        |
|    107 | Suresh   |  24000 | Sales     |
|    108 | Vijay    |  22000 | MKTG      |
+--------+----------+--------+-----------+
4 rows in set (0.00 sec)
```

6. Display the details of the employee who earns more salary than the average salary of his department.

Answer:

```
mysql> SELECT * FROM employee a
    -> WHERE salary > (SELECT AVG(salary) FROM employee b
    -> GROUP BY dept_name
    -> HAVING b.dept_name=a.dept_name);
```

```
+--------+----------+--------+-----------+
| emp_id | emp_name | salary | dept_name |
+--------+----------+--------+-----------+
|    101 | Amit     |  25000 | IT        |
|    107 | Suresh   |  24000 | Sales     |
|    108 | Vijay    |  22000 | MKTG      |
+--------+----------+--------+-----------+
3 rows in set (0.00 sec)
```

7. Display total number of employees.

Answer:

```
mysql> SELECT COUNT(emp_id) FROM employee;
+---------------+
| COUNT(emp_id) |
+---------------+
|             8 |
+---------------+
1 row in set (0.00 sec)
```