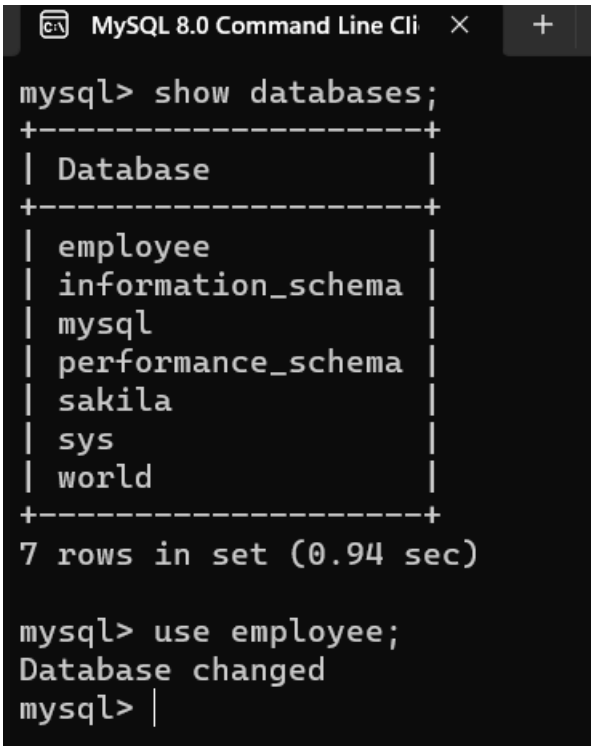
MySQL Assignment-1

CREATION:-

create database employee;

use employee;



QUESTION 1:

Assignment-1 --- Create two tables: EmployeeDetails and EmployeeSalary. AND

Columns for EmployeeDetails: EmpId FullName ManagerId DateOfJoining City && Columns for EmployeeSalary: : EmpId Project Salary Variable.

ANSWER:

Create EmployeeDetails table:-

```
create table employee_details(
   -> empId int,
   -> empFullName varchar(20),
   -> managerId int,
   -> dataOfJoin date,
   -> empCity varchar(20));
```

Create EmployeeSalary table:-

```
create table employee_salary(
  -> empId int,
  -> project varchar(20),
  -> salary int);
```

show tables;

```
mysql> show tables;
+--------------------+
| Tables_in_employee |
+--------------------+
| employee_details   |
| employee_salary    |
+--------------------+
2 rows in set (0.90 sec)
```

INSERTION:-

/* ADD RECORDS IN BOTH TABLES */

1) employee_details TABLE values:

```
insert into employee_details(empId,empFullName,managerId,dataOfJoin,empCity)
values(101,"Manik",1001,"1995-02-01","Mumbai"),(102,"Jackie",NULL,"2021-07-
05","Pune"),(103,"Sakthi",1002,"2020-02-08","Kolkata"),(104,"Mythili",1003,"2020-06-
03","Chennai"),(105,"Maria",NULL,"2023-04-08","Bangalore"),(106,"Sankar",1004,"1990-01-
09","Delhi"),(107,"Raja",NULL,"2022-11-20","Hyderabad"),(102,"Jackie",1005,"2021-07-
05","Pune"),(108,"Jaya",NULL,"2022-08-04","Trivandrum"),(109,"Mehta",NULL,"2023-05-
03","Chennai"),(110,"Varun",NULL,"2023-07-30","Pune");
```

2) employee_salary TABLE values:

```
insert into
employee_salary(empId,Project,Salary)values(101,"P1",75000.00),(102,"P2",30000.00),(103,
"P1",45000.00),(104,"P2",42000.00),(105,"P1",20000.00),(106,"P2",67000.00),(107,"P2",400
00.00),(108,"P1",28000.00);
```

select * from employee_details;

```
mysql> select * from employee_details;
+-------+-------------+-----------+------------+------------+
| empId | empFullName | managerId | dateOfJoin | empCity    |
+-------+-------------+-----------+------------+------------+
|   101 | Manik       |      1001 | 1995-02-01 | Mumbai     |
|   102 | Jackie      |      1005 | 2021-07-05 | Pune       |
|   103 | Sakthi      |      1002 | 2020-02-08 | Kolkata    |
|   104 | Mythili     |      1003 | 2020-06-03 | Chennai    |
|   105 | Maria       |      NULL | 2023-04-08 | Bangalore  |
|   106 | Sankar      |      1004 | 1990-01-09 | Delhi      |
|   107 | Raja        |      NULL | 2022-11-20 | Hyderabad  |
|   102 | Jackie      |      1005 | 2021-07-05 | Pune       |
|   108 | Jaya        |      NULL | 2022-08-04 | Trivandrum |
|   109 | Mehta       |      NULL | 2023-05-03 | Chennai    |
|   110 | Varun       |      NULL | 2023-07-30 | Pune       |
|   111 | Yash        |      1006 | 1987-11-15 | Ahmedhabad |
|   112 | Subhash     |      1007 | 1994-10-26 | Lucknow    |
+-------+-------------+-----------+------------+------------+
13 rows in set (0.00 sec)
```

select * from employee_salary;

```
mysql> select * from employee_salary;
+-------+---------+--------+
| empId | project | salary |
+-------+---------+--------+
|   101 | P1      |  75000 |
|   102 | P2      |  30000 |
|   103 | P1      |  45000 |
|   104 | P2      |  42000 |
|   105 | P1      |  20000 |
|   106 | P2      |  67000 |
|   107 | P2      |  40000 |
|   108 | P1      |  28000 |
|   111 | P1      |   NULL |
|   112 |         |  20000 |
+-------+---------+--------+
10 rows in set (0.00 sec)
```

QUESTION 2: SQL Query to fetch records that are present in one table but not in another table.

ANSWER:-

Using NOT IN:

```
select * from employee_details where empId not in(select empId from employee_salary);
```

Using NOT EXISTS:

select * from employee_details where not exists(select empId from employee_salary where employee_details.empId = employee_salary.empId);

```
mysql> select * from employee_details where empId not in(select empId from employee_salary);
+-------+-------------+-----------+------------+---------+
| empId | empFullName | managerId | dateOfJoin | empCity |
+-------+-------------+-----------+------------+---------+
|   109 | Mehta       |      NULL | 2023-05-03 | Chennai |
|   110 | Varun       |      NULL | 2023-07-30 | Pune    |
+-------+-------------+-----------+------------+---------+
2 rows in set (0.09 sec)
```

QUESTION 3: SQL query to fetch all the employees who are not working on any project.

ANSWER:-

select empId, empFullName from employee_details where not exists(select Project from employee_salary where employee_details.empId = employee_salary.empId);

```
mysql> select empId, empFullName from employee_details where not exists(select Project from employee_salary where employee_details.empId = employee_salary.e
mpId);
+------+------------+
| empId | empFullName |
+------+------------+
|  109 | Mehta      |
|  110 | Varun      |
+------+------------+
2 rows in set (0.07 sec)
```

QUESTION 4: SQL query to fetch all the Employees from EmployeeDetails who joined in the Year 2020.

ANSWER:-

select * from employee_details where year(dateOfJoin)=2020;

```
mysql> select * from employee_details where year(dateOfJoin)=2020;
+-------+-------------+-----------+------------+---------+
| empId | empFullName | managerId | dateOfJoin | empCity |
+-------+-------------+-----------+------------+---------+
|   103 | Sakthi      |      1002 | 2020-02-08 | Kolkata |
|   104 | Mythili     |      1003 | 2020-06-03 | Chennai |
+-------+-------------+-----------+------------+---------+
2 rows in set (0.00 sec)
```

select * from employee_details where dateOfJoin between date '2020-01-01' and date '2020-12-31';

QUESTION 5: Fetch all employees from EmployeeDetails who have a salary record in EmployeeSalary.

ANSWER:-

Using INNER JOIN:-

select ed.empId, ed.empFullName, es.salary from employee_details ed join employee_salary es on ed.empId=es.empId;

Using SUB-QUERY:-

select * from employee_details ed where exists(select * from employee_salary es where ed.empId=es.empId);

```
mysql>   select ed.empId, ed.empFullName, es.salary from employee_details ed join employee_salary es on ed.empId=es.empId;
+-------+-------------+--------+
| empId | empFullName | salary |
+-------+-------------+--------+
|   101 | Manik       |  75000 |
|   102 | Jackie      |  30000 |
|   103 | Sakthi      |  45000 |
|   104 | Mythili     |  42000 |
|   105 | Maria       |  20000 |
|   106 | Sankar      |  67000 |
|   107 | Raja        |  40000 |
|   102 | Jackie      |  30000 |
|   108 | Jaya        |  28000 |
|   111 | Yash        |   NULL |
|   112 | Subhash     |  20000 |
+-------+-------------+--------+
11 rows in set (0.07 sec)
```

QUESTION 6: Write an SQL query to fetch a project-wise count of employees.

ANSWER:-

select Project, count(empId) as ProjectWise_Emp_Count from employee_salary group by Project;

```
mysql> select Project, count(empId) as ProjectWise_Emp_Count from employee_salary group by Project;
+---------+-----------------------+
| Project | ProjectWise_Emp_Count |
+---------+-----------------------+
| P1      |                     5 |
| P2      |                     4 |
|         |                     1 |
+---------+-----------------------+
3 rows in set (0.05 sec)
```

QUESTION 7: Fetch employee names and salaries even if the salary value is not present for the employee.

ANSWER:-

select ed.empFullName, es.Salary from employee_details ed left join employee_salary es on ed.empId=es.empId;

```
mysql> select ed.empFullName, es.Salary from employee_details ed left join employee_salary es on ed.empId=es.empId;
+-------------+--------+
| empFullName | Salary |
+-------------+--------+
| Manik       |  75000 |
| Jackie      |  30000 |
| Sakthi      |  45000 |
| Mythili     |  42000 |
| Maria       |  20000 |
| Sankar      |  67000 |
| Raja        |  40000 |
| Jackie      |  30000 |
| Jaya        |  28000 |
| Mehta       |   NULL |
| Varun       |   NULL |
| Yash        |   NULL |
| Subhash     |  20000 |
+-------------+--------+
13 rows in set (0.13 sec)
```

QUESTION 8: Write an SQL query to fetch all the Employees who are also managers.

ANSWER:-

select * from employee_details where managerID;

```
mysql> select * from employee_details where managerID;
+-------+-------------+-----------+------------+------------+
| empId | empFullName | managerId | dateOfJoin | empCity    |
+-------+-------------+-----------+------------+------------+
|   101 | Manik       |      1001 | 1995-02-01 | Mumbai     |
|   102 | Jackie      |      1005 | 2021-07-05 | Pune       |
|   103 | Sakthi      |      1002 | 2020-02-08 | Kolkata    |
|   104 | Mythili     |      1003 | 2020-06-03 | Chennai    |
|   106 | Sankar      |      1004 | 1990-01-09 | Delhi      |
|   102 | Jackie      |      1005 | 2021-07-05 | Pune       |
|   111 | Yash        |      1006 | 1987-11-15 | Ahmedhabad |
|   112 | Subhash     |      1007 | 1994-10-26 | Lucknow    |
+-------+-------------+-----------+------------+------------+
8 rows in set (0.00 sec)
```

QUESTION 9: Write an SQL query to fetch duplicate records from EmployeeDetails.

ANSWER:-

select empId, count(*) as DuplicateCount from employee_details group by empId having count(*)>1;

```
mysql> select empId, count(*) as DuplicateCount from employee_details group by empId having count(*)>1;
+-------+----------------+
| empId | DuplicateCount |
+-------+----------------+
|   102 |              2 |
+-------+----------------+
1 row in set (0.28 sec)
```

QUESTION 10: Write an SQL query to fetch only odd rows from the table.

ANSWER:-

Query for employee_details table:-

select * from(select *, row_number() over(order by empId) as OddRowNum from employee_details) ed where ed.OddRowNum%2=1;

```
mysql>    select * from(select *, row_number() over(order by empId) as OddRowNum from employee_details) ed where ed.OddRowNum%2=1;
+-------+-------------+-----------+------------+------------+-----------+
| empId | empFullName | managerId | dateOfJoin | empCity    | OddRowNum |
+-------+-------------+-----------+------------+------------+-----------+
|   101 | Manik       |      1001 | 1995-02-01 | Mumbai     |         1 |
|   102 | Jackie      |      1005 | 2021-07-05 | Pune       |         3 |
|   104 | Mythili     |      1003 | 2020-06-03 | Chennai    |         5 |
|   106 | Sankar      |      1004 | 1990-01-09 | Delhi      |         7 |
|   108 | Jaya        |      NULL | 2022-08-04 | Trivandrum |         9 |
|   110 | Varun       |      NULL | 2023-07-30 | Pune       |        11 |
|   112 | Subhash     |      1007 | 1994-10-26 | Lucknow    |        13 |
+-------+-------------+-----------+------------+------------+-----------+
7 rows in set (0.04 sec)
```

Query for employee_salary table:-

select * from(select *, row_number() over(order by empId) as OddRowNumbers from employee_salary) es where es.OddRowNumbers%2=1;

```
mysql> select * from(select *, row_number() over(order by empId) as OddRowNumbers from employee_salary) es where es.OddRowNumbers%2=1;
+-------+---------+--------+--------------+
| empId | project | salary | OddRowNumbers |
+-------+---------+--------+--------------+
|   101 | P1      |  75000 |            1 |
|   103 | P1      |  45000 |            3 |
|   105 | P1      |  20000 |            5 |
|   107 | P2      |  40000 |            7 |
|   111 | P1      |   NULL |            9 |
+-------+---------+--------+--------------+
5 rows in set (0.00 sec)
```

QUESTION 11: Write a query to find the 3rd highest salary from a table without top or limit keyword.

ANSWER:-

select ed.empId,ed.empFullName,es.Salary from employee_details ed join employee_salary es on ed.empId=es.empId where Salary=(select max(Salary) from employee_salary where Salary < (select max(Salary) from employee_salary where Salary < (select max(Salary) from employee_salary)));

```
mysql> select ed.empId,ed.empFullName,es.Salary from employee_details ed join employee_salary es on ed.empId=es.empId where Salary=(select max(Salary) from
employee_salary where Salary < (select max(Salary) from employee_salary where Salary < (select max(Salary) from employee_salary)));
+-------+-------------+--------+
| empId | empFullName | Salary |
+-------+-------------+--------+
|   103 | Sakthi      |  45000 |
+-------+-------------+--------+
1 row in set (0.09 sec)
```