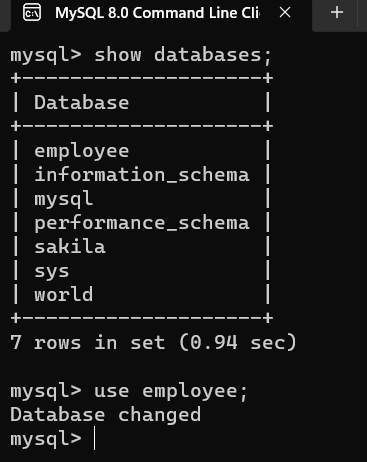
MySQL Assignment-1

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:-

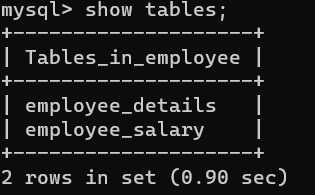
create table employee\_salary(

-> empId int,

-> project varchar(20),

-> salary int);

show tables;



INSERTION:-

/\* ADD RECORDS IN BOTH TABLES \*/

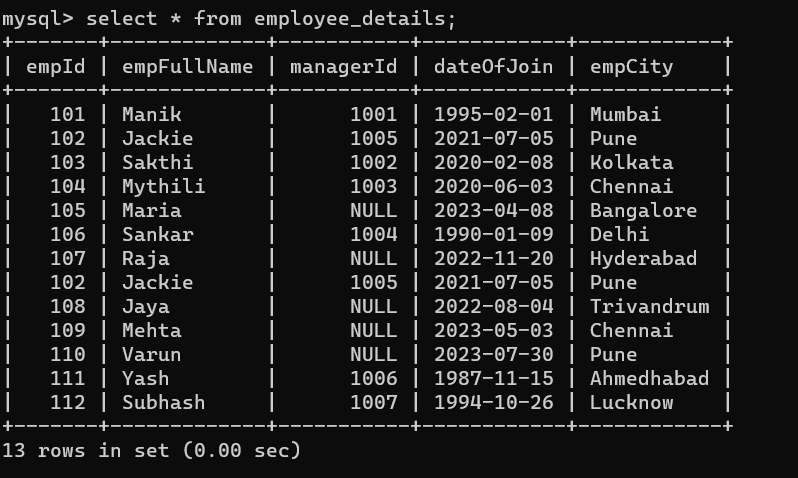
1. employee\_details TABLE values:

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");

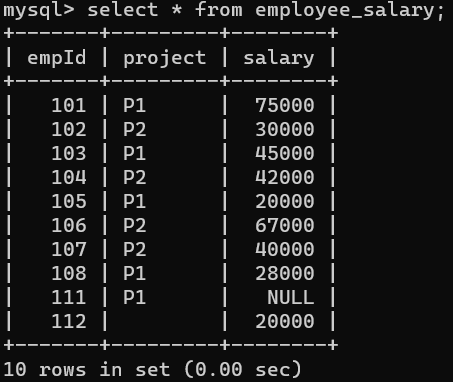
1. employee\_salary TABLE values:

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",40000.00),(108,"P1",28000.00);

select \* from employee\_details;



select \* from employee\_salary;



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

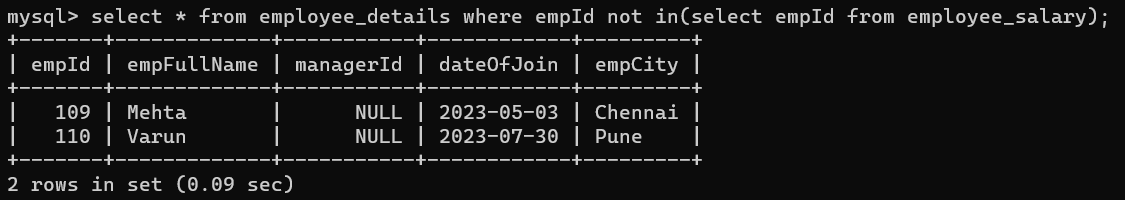
ANSWER:-

Using NOT IN:

select \* from employee\_details where empId not in(select empId from employee\_salary);

Using NOT EXISTS:

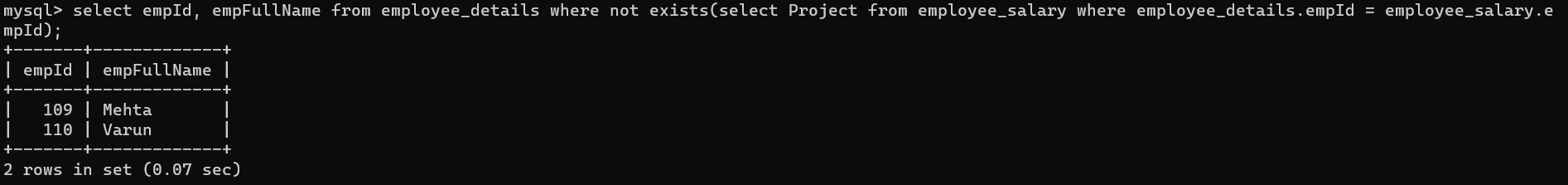
select \* from employee\_details where not exists(select empId from employee\_salary where employee\_details.empId = employee\_salary.empId);



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

ANSWER:-

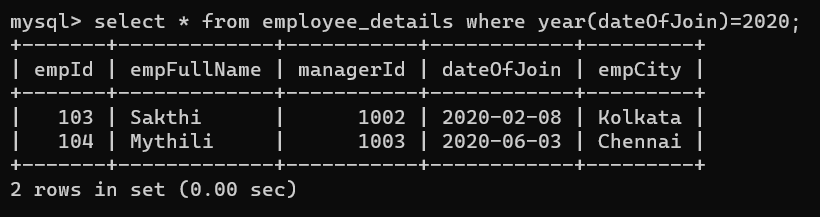
select empId, empFullName from employee\_details where not exists(select Project from employee\_salary where employee\_details.empId = employee\_salary.empId);



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;



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.

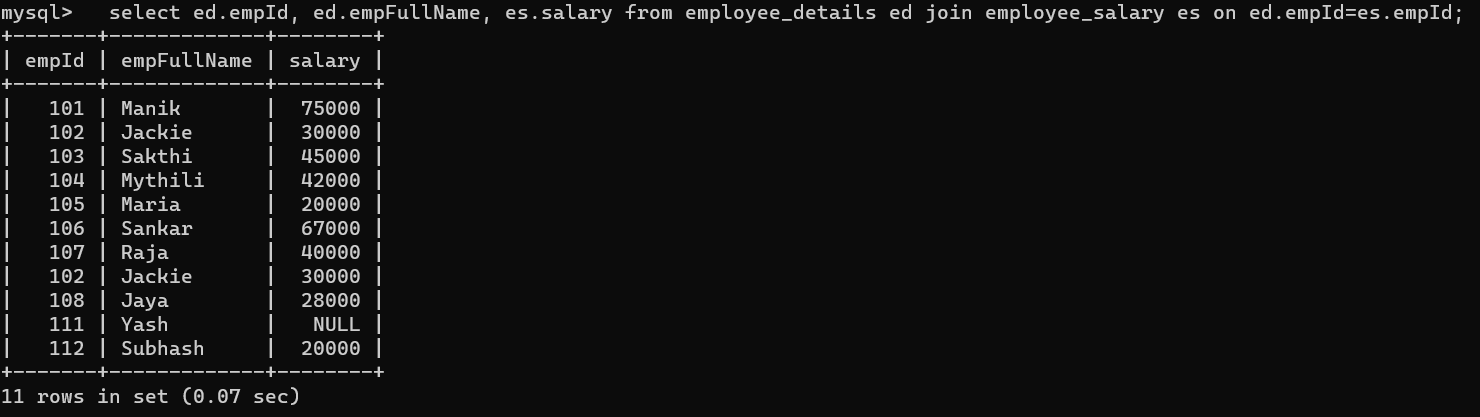
ANSWER:-

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:-

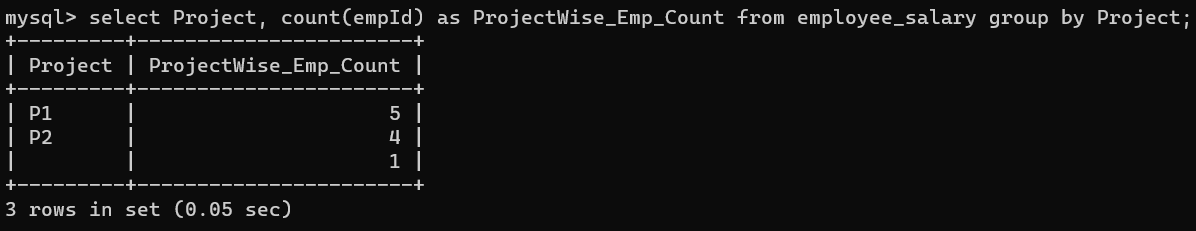
select \* from employee\_details ed where exists(select \* from employee\_salary es where ed.empId=es.empId);



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

ANSWER:-

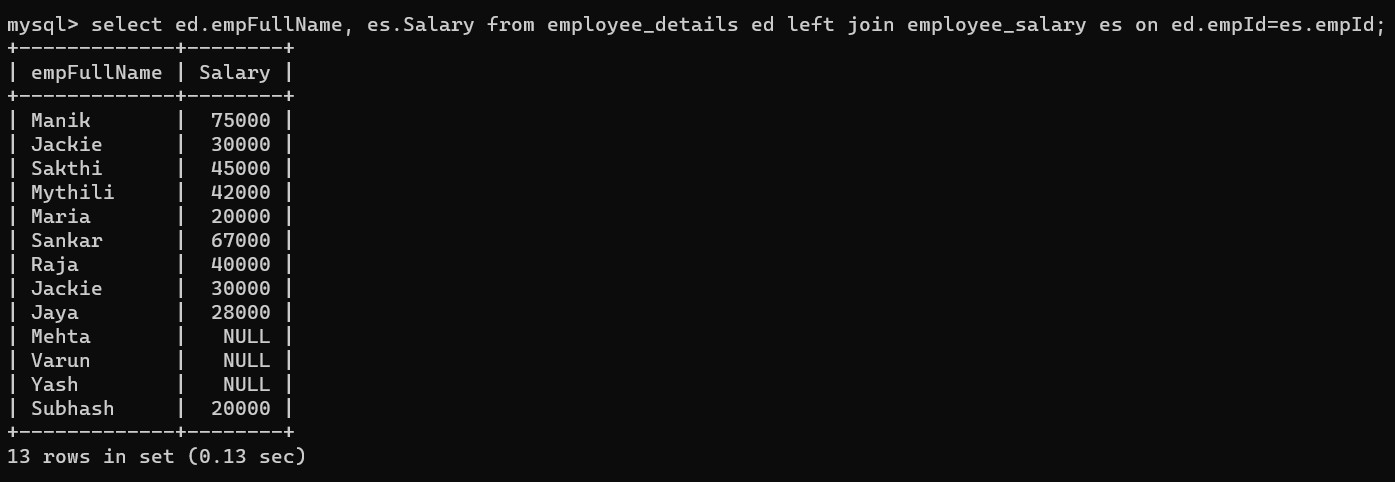
select Project, count(empId) as ProjectWise\_Emp\_Count from employee\_salary group by Project;



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

ANSWER:-

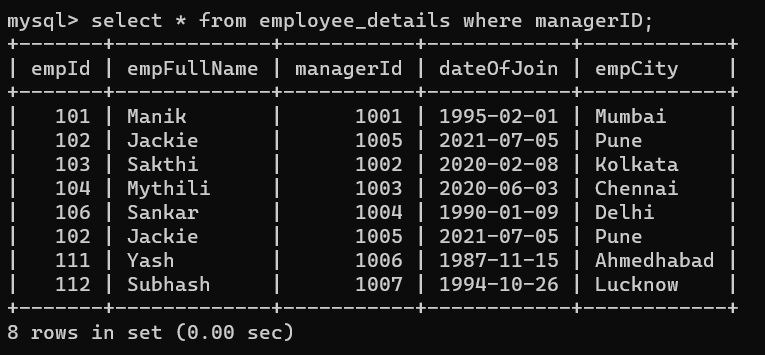
select ed.empFullName, es.Salary from employee\_details ed left join employee\_salary es on ed.empId=es.empId;



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

ANSWER:-

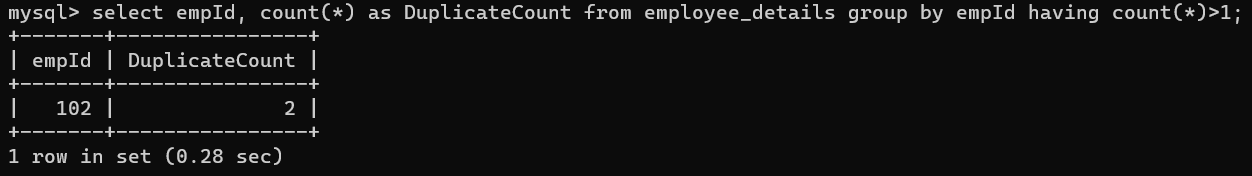
select \* from employee\_details where managerID;



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;

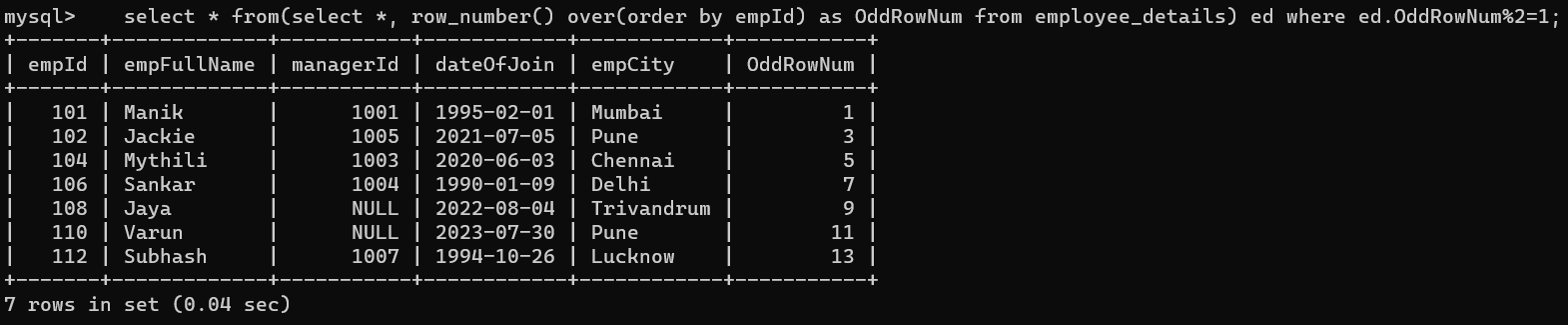


QUESTION 10: Write an SQL query to fetch only odd rows from the table.

ANSWER:-

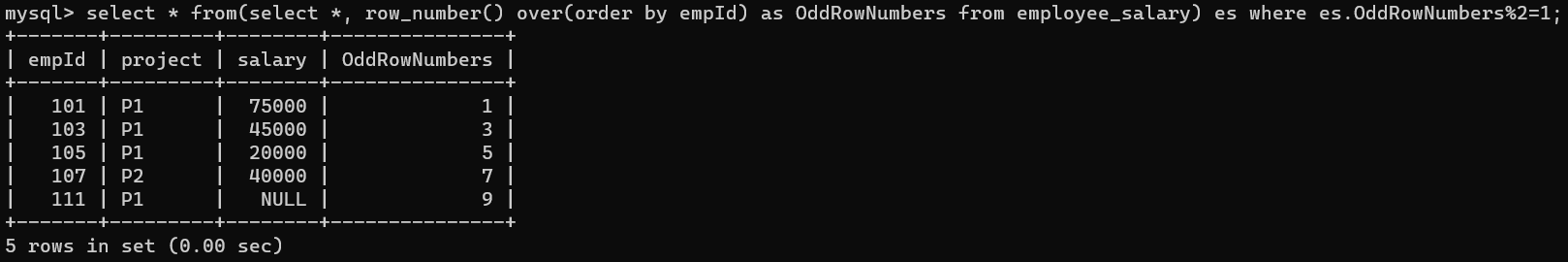
Query for employee\_details table:-

select \* from(select \*, row\_number() over(order by empId) as OddRowNum from employee\_details) ed where ed.OddRowNum%2=1;



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;



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)));

