```
#####################################################################
                Manjit Singh Duhan
                Addmission No: IITP001316
                roll No:  2303res134
                email: duhan.manjit@gmail.com
                       manjit_2303res134@iitp
#####################################################################


1. create table departments with attribute department_id(PK), department_name, manager_id, location_id.

    CREATE table department(
            department_id int NOT NULL PRIMARY KEY,
            department_name varchar(20),
            manager_id int,
            location_id int);

2. insert some values in departments table.

    INSERT into department values(101, "marketing", 110, 1001);
    INSERT into department values(102, "IT", 120, 1002);
    INSERT into department values(103, "sales", 130, 1003);

3. create table employee with attribute employee_id(PK), name, email, hire_year, salary, manage_id, manager_id,
department_id(FK).

    CREATE table employee(
                employee_id int NOT NULL,
                name varchar(20),
                email_id varchar(30),
                hire_year int,
                salary int,
                manager_id int,
                department_id int,
                PRIMARY KEY (employee_id),
                FOREIGN KEY (department_id) REFERENCES Department(department_id));

    INSERT into Employee values(110, 'mukesh', "mukesh@gmail.com", 2004, 25000, 110, 101);
    INSERT into Employee values(160, 'rahul', "rahul@gmail.com", 2006, 20000, 110, 101);
    INSERT into Employee values(120, 'ramesh', "ramesh@gmail.com", 2005, 19000, 120, 102);
    INSERT into Employee values(140, 'akash', "akash@gmail.com", 2010, 18000, 120, 102);
    INSERT into Employee values(150, 'kamlesh', "kamlesh@gmail.com", 2008, 25000, 130, 103);
    INSERT into Employee values(130, 'mohan', "mohan@gmail.com", 2002, 30000, 130, 103);
    INSERT into Employee values(170, 'rahul', "rahul1@gmail.com", 2012, 10000, 130, 103);

4. SQL query to find the name, department number, and department name for each employee.

    Select E.name, D.department_id, D.department_name FROM Employee as E natural join Department as D;
    Select E.Name, D.Department_Id, D.Department_Name from employee as E Inner Join Department as D on E.Department_Id =
 D.Department_ID;

```

5. SQL query to find all those employees who work in department ID 101 or 102. Return name, department id and department name.

```
Select E.name, D.department_id, D.department_name FROM Employee as E natural join Department as D Where (D.department_id = 101 or D.department_id = 102);
Select E.Name, D.Department_Id, D.Department_Name from employee as E natural Join Department as D Where D.department_id IN (101, 102);
Select E.Name, D.Department_Id, D.Department_Name from employee as E Inner Join Department as D on E.Department_Id = D.Department_ID where (D.department_id = 101 or D.department_id = 102);
Select E.Name, D.Department_Id, D.Department_Name from employee as E Inner Join Department as D on E.Department_Id = D.Department_ID where D.department_id IN (101, 102);
Select E.Name, D.Department_Id, D.Department_Name from employee as E Inner Join Department as D on E.Department_Id = D.Department_ID and D.department_id IN (101, 102);
```

6. SQL query to find the employees who earn less than the employee of ID 104.

```
Select employee.Name from employee where  employee.salary < (select employee.salary from employee where employee.employee_id = 140);
Select E.name FROM Employee as E join employee S on E.salary < S.salary and S.employee_id = 140;
Select E.name FROM Employee as E join employee S where E.salary < S.salary and S.employee_id = 140;
Select E.name, E.salary, E.employee_id FROM Employee as E join employee S where E.salary < S.salary and S.employee_id = 140;
```

7. create a table job_grade with attribute grade_level(PK), low_sal, high_sal. all attribute are INT.

```
CREATE table job_grade(
    grade_level int NOT NULL PRIMARY KEY,
    low_sal int,
    high_sal int);

INSERT into job_grade values (111, 1000, 9999);
INSERT into job_grade values (222, 10000, 19999);
INSERT into job_grade values (333, 20000, 40000);
```

8. SQL query to find the name, department_name, salary, and job grade for all employees.

```
Select
    E.Name, E.salary, D.Department_Name, J.grade_level from employee as E
    JOIN Department as D ON E.Department_Id = D.Department_ID
    JOIN Job_grade as J Where E.salary Between J.low_sal and J.high_sal;
```

9. find the employee with max salary.

```
select E.name from employee as E where E.salary = (select max(E1.salary) from employee as E1);
Select E.name from employee as E order by E.salary DESC LIMIT 1;
```

10. find 2nd highest salaried employee

```
SELECT E.name, E.salary FROM employee as E ORDER BY E.salary DESC LIMIT 1 OFFSET 1;
```