Database Management System Lab

Assignment - 4

Name :- Harsh Singh

Roll No. :- 244CA018

1. Create the following tables with the following attributes and constraints on them.

a. **Employee** (Fname, mname, lname, Ssn, Bdate, address, gender, salary, Super_Ssn, Dept_num)
Lname, Ssn, Dept_num should be not null

```
=>  CREATE TABLE Employee (
    Fname VARCHAR(50),
    Mname VARCHAR(50),
    Lname VARCHAR(50) NOT NULL,
    Ssn CHAR(9) NOT NULL,
    Bdate DATE,
    Address VARCHAR(255),
    Gender CHAR(1),
    Salary DECIMAL(10, 2),
    Super_Ssn CHAR(9),
    Dept_num INT NOT NULL,
    PRIMARY KEY (Ssn),
    FOREIGN KEY (Dept_num) REFERENCES
    Department(Dept_num),
    FOREIGN KEY (Super_Ssn) REFERENCES Employee(Ssn)
    );
```

b. **Department** (Dept_num, Dept_name, Mgr_Ssn, Mgr_startdate)
Dept_name should be unique.

```
=>  CREATE TABLE Department (
    Dept_num INT PRIMARY KEY,
    Dept_name VARCHAR(100) UNIQUE NOT NULL,
    Mgr_Ssn CHAR(9),
    Mgr_startdate DATE,
    FOREIGN KEY (Mgr_Ssn) REFERENCES Employee(Ssn)
    );
```

c. **Department_locations** (Dept_num, location.
Dept_num and location both are primary key
Dept_num is foreign key

```
=>   CREATE TABLE Department_locations (
     Dept_num INT,
     Location VARCHAR(100),
     PRIMARY KEY (Dept_num, Location),
     FOREIGN KEY (Dept_num) REFERENCES
     Department(Dept_num)
     );
```

d. **Project** (Proj_num, Proj_name, Proj_location, Dept_num)

```
=>   CREATE TABLE Project (
     Proj_num INT PRIMARY KEY,
     Proj_name VARCHAR(100),
     Proj_location VARCHAR(100),
     Dept_num INT,
     FOREIGN KEY (Dept_num) REFERENCES
     Department(Dept_num)
     );
```

e. **Employee_Project** (Ssn, Proj_num, Hours)

```
=>   CREATE TABLE Employee_Project (
     Ssn CHAR(9),
     Proj_num INT,
     Hours DECIMAL(5, 2),
     PRIMARY KEY (Ssn, Proj_num),
     FOREIGN KEY (Ssn) REFERENCES Employee(Ssn),
     FOREIGN KEY (Proj_num) REFERENCES Project(Proj_num)
     );
```

f. **Dependent** (Ssn, Dept_name, gender, bdate, relationship)

```
=>   CREATE TABLE Dependent (
     Ssn CHAR(9),
     Dept_name VARCHAR(100),
     Gender CHAR(1),
     Bdate DATE,
     Relationship VARCHAR(50),
     PRIMARY KEY (Ssn, Dept_name),
     FOREIGN KEY (Ssn) REFERENCES Employee(Ssn),
     FOREIGN KEY (Dept_name) REFERENCES
     Department(Dept_name)
     );
```

2. Add two column blood group and hobbies to employee table.

```
=>   ALTER TABLE employee ADD (b_group char(2), hobbies
     varchar(100));
```

3. Increase the size of column blood group to 15 to the employee table.

```
=>   ALTER TABLE employee MODIFY (b_group char(15));
```

4. Drop column hobbies from the employee table.

```
=>   ALTER TABLE employee DROP COLUMN hobbies;
```

5. Rename Employee Table to Employee_details.

```
=>ALTER TABLE employee RENAME TO employee_details;
```

6. Insert at least five records in each table.

a. Employee

```
=>   INSERT INTO Employee VALUES ('John', 'A', 'Doe', '
     111223333', '15-MAR-1995', '123 Main St', 'M',
     3500.00, NULL, 1, 'A+');

     INSERT INTO Employee VALUES ('Jane', 'B', 'Smith',
     '222334555', '20-JUL-1990', '456 Oak Ave', 'F',
     4500.00, NULL, 2, 'B-');

     INSERT INTO Employee VALUES ('Alice', 'C', 'Johnson',
     '333445666', '11-JUN-1982', '789 Pine Blvd', 'F',
     6000.00, NULL, 3, 'O+');

     INSERT INTO Employee VALUES ('Bob', 'D', 'Williams',
     '444556777', '29-AUG-1975', '101 Maple Rd', 'M',
     5500.00, NULL, 1, 'B+');

     INSERT INTO Employee VALUES ('Charlie', 'E', 'Brown',
     '555667888', '17-FEB-1992', '202 Cedar St', 'M',
     3000.00, NULL, 2, 'A+');
```

b. Department

```
=>   INSERT INTO Department VALUES (1, 'Marketing',
     '111223333', '01-MAY-2010');

     INSERT INTO Department VALUES (2, 'Sales',
     '222334555', '14-AUG-2015');

     INSERT INTO Department VALUES (3, 'Engineering',
     '333445666', '21-SEP-2019');

     INSERT INTO Department VALUES (4, 'HR',
     '444556777', '19-MAR-2008');

     INSERT INTO Department VALUES (3, 'Finance',
     '555667888', 061-JUL-2005');
```

c. Department_Loacations

```
=>   INSERT INTO Department_locations VALUES (1, 'New
     York');

     INSERT INTO Department_locations VALUES (1, 'San
     Francisco');

     INSERT INTO Department_locations VALUES (2,
     'Chicago');

     INSERT INTO Department_locations VALUES (2, 'Los
     Angeles');

     INSERT INTO Department_locations VALUES (3,
     'Seattle');
```

d. Project

```
=>   INSERT INTO Project VALUES (1, 'Super', 'New York',
     1);

     INSERT INTO Project VALUES (2, 'TechUpgrade',
     'Seattle', 3);

     INSERT INTO Project VALUES (3, 'AdCampaign', 'San
     Francisco', 1);

     INSERT INTO Project VALUES (4, 'SalesBoost', 'Los
     Angeles', 2);

     INSERT INTO Project VALUES (5, 'HR', 'Paris', 2);
```

e. Employee_Project

```
=>   INSERT INTO Employee_Project VALUES ('111223333', 1,
     40);

     INSERT INTO Employee_Project VALUES ('222334555', 3,
     35);

     INSERT INTO Employee_Project VALUES ('333445666', 2,
     50);

     INSERT INTO Employee_Project VALUES ('444556777', 4,
     45);

     INSERT INTO Employee_Project VALUES ('555667888', 2,
     30);
```

f. Dependent

```
=>   INSERT INTO Dependent VALUES ('111223333',
     'Marketing', 'F', '10-MAR-2010', 'Wife');

     INSERT INTO Dependent VALUES ('222334555', 'Sales',
     'M', '19-AUG-1993', 'Son');

     INSERT INTO Dependent VALUES ('333445666',
     'Engineering', 'F', '22-APR-2017', 'Daughter');

     INSERT INTO Dependent VALUES ('444556777',
     'Marketing', 'F', '02-JAN-2005', 'Daughter');

     INSERT INTO Dependent VALUES ('555667888', 'Sales',
     'M', '15-JUL-2002', 'Son');
```

7. Give 1000 rupees bonus to each employee.

```
=>   UPDATE employee SET salary=salary+1000;
```

8. Increase the salary of the employees having salary <5000 by 500 rupees.

```
=>   UPDATE employee SET salary=salary+500 WHERE
     salary<5000;
```

9. Give 100 rupees bonus to employees having salary less than 10000 rupees and birth date before 1990.

```
=>   UPDATE employee SET salary=salary+100 WHERE
     salary<10000 and bdate<'01-JAN-1990';
```

10. Give 100 rupees bonus to employees having salary less than 10000 rupees or birth date before 1990.

```
=>   UPDATE employee SET salary=salary+100 WHERE
     salary<10000 or bdate<'01-JAN-1990';
```

11. Give 100 rupees bonus to employees having salary between 1000 to 5000 rupees and birth date before 1990.

```
=>   UPDATE employee SET salary=salary+100 WHERE salary
BETWEEN 1000 AND 5000 AND bdate<'01-JAN-1990';
```

12. Give 100 rupees bonus to employees having salary between 1000, 3000 and 5000 rupees.

```
=>UPDATE employee SET salary=salary+100 WHERE salary in
(1000, 3000, 5000);
```

13. Update phone number with 0000 where NULL.

```
=>   UPDATE employee SET phone_no='0000' WHERE phone_no
     IS NULL;
```

14. Give 100 rupees bonus to employees having salary not between 1000 to 5000 rupees and birth date before 1990.

```
=>   UPDATE employee SET salary=salary+100 WHERE salary
     NOT BETWEEN 1000 AND 5000 AND bdate < TO_DATE('01-
     JAN-1990', 'DD-MON-YYYY');
```

15. Give 100 rupees bonus to employees having salary between 1000, 3000 and 5000 rupees.

```
=>   UPDATE employee SET salary=salary+100 WHERE salary
     in (1000, 3000, 5000);
```

16. Delete from employee the rows having bdate less than 1970.

```
=>   DELETE FROM employee WHERE bdate < '01-JAN-1970';
```

17. List the name and age of all employees.

```
=>   SELECT fname, mname, lname,(SYSDATE - BDATE)/365.25
     as AGE from employee;
```

18. Display the salaries offered to the employees.

```
=>   SELECT salary FROM employee;
```

19. List the Bdate and Salary of Employee 'Smith'.

```
=>   SELECT bdate, salary FROM employee WHERE fname LIKE
     'Smith';
```

20. Find the location of Project 'SUPER'.

```
=>   SELECT proj_location FROM project WHERE
     proj_name='Super';
```

21. Find the dependent details of Employee with Ssn number 482928.

```
=>   SELECT * FROM dependent WHERE ssn='482928';
```

22. List the employees having salary > 2000 and bdate before 1/1/1990.

```
=>   SELECT * FROM employee WHERE salary > 2000 AND bdate
     < TO_DATE('01-JAN-1990', 'DD-MON-YYYY');
```

23. List the employees belonging to dept_num 1.

```
=>   SELECT * FROM employee WHERE dept_num = 1;
```

24. List the project details of dept_num 5.

```
=>   SELECT * FROM project WHERE dept_num = 5;
```

25. List the employee details with their department name.

```
=>   SELECT * FROM employee JOIN department ON
     employee.dept_num = department.dept_num;
```

26. List the employee details with their project names.

```
=>   SELECT * FROM employee JOIN project ON
     employee.dept_num = project.dept_num;
```

27. List the employees belonging to Marketing department.

```
=>   SELECT employee.* FROM employee JOIN department ON
     employee.dept_num = department.dept_num WHERE
     department.dept_name='Marketing';
```

28. List the project details belonging of Sales department.

```
=>   SELECT * FROM project WHERE dept_name='Sales';
```

29. List the dependent details of employee 'Smith'.

```
=>   SELECT dependent.* FROM employee JOIN dependent ON
     employee.ssn=dependent.ssn AND
     employee.fname='Smith';
```

30. List the various locations of 'Marketing' department.

```
=>   SELECT location FROM department_locations;
```

31. List the employees going to 'Surathkal' branch.

```
=>   SELECT employee.* FROM employee JOIN
     department_locations ON
     employee.dept_num=department_locations.dept_num AND
     department_locations.location='Surathkal';
```

32. List the employees in the descending order of their salary.

```
=>   SELECT * FROM employee ORDER BY salary DESC;
```

33. List the dependents in the descending order of their names.

```
=>   SELECT dependent.* FROM dependent JOIN employee ON
     dependent.ssn=employee.ssn ORDER BY employee.fname
     DESC;
```