dbms lab 3

```
--q1
CREATE TABLE EMPLOYEE (
    Fname VARCHAR(15) NOT NULL,
    Minit CHAR(1),
    Lname VARCHAR(15) NOT NULL,
    Ssn CHAR(9) NOT NULL,
    Bdate DATE,
    Address VARCHAR(30),
    Sex CHAR(1),
    Salary DECIMAL(10, 2),
    Super_ssn CHAR(9),
    Dno INT,
    PRIMARY KEY (Ssn)
);

CREATE TABLE DEPARTMENT (
    Dname VARCHAR(15) NOT NULL UNIQUE,
    Dnumber INT NOT NULL,
    Mgr_ssn CHAR(9) NOT NULL,
    Mgr_start_date DATE,
    PRIMARY KEY (Dnumber),
    FOREIGN KEY (Mgr_ssn) REFERENCES EMPLOYEE(Ssn)
);

CREATE TABLE DEPT_LOCATIONS (
    Dnumber INT NOT NULL,
    Dlocation VARCHAR(15) NOT NULL,
    PRIMARY KEY (Dnumber, Dlocation),
    FOREIGN KEY (Dnumber) REFERENCES DEPARTMENT(Dnumber)
);

CREATE TABLE PROJECT (
    Pname VARCHAR(15) NOT NULL UNIQUE,
    Pnumber INT NOT NULL,
    Plocation VARCHAR(15),
    Dnum INT NOT NULL,
    PRIMARY KEY (Pnumber),
    FOREIGN KEY (Dnum) REFERENCES DEPARTMENT(Dnumber)
);
```

```
CREATE TABLE WORKS_ON (
   Essn CHAR(9) NOT NULL,
   Pno INT NOT NULL,
   Hours DECIMAL(3, 1) NOT NULL,
   PRIMARY KEY (Essn, Pno),
   FOREIGN KEY (Essn) REFERENCES EMPLOYEE(Ssn),
   FOREIGN KEY (Pno) REFERENCES PROJECT(Pnumber)
);

CREATE TABLE DEPENDENT (
   Essn CHAR(9) NOT NULL,
   Dependent_name VARCHAR(15) NOT NULL,
   Sex CHAR(1),
   Bdate DATE,
   Relationship VARCHAR(8),
   PRIMARY KEY (Essn, Dependent_name),
   FOREIGN KEY (Essn) REFERENCES EMPLOYEE(Ssn)
);

--q2
-- 1. CREATE: Creating the EMPLOYEE and DEPARTMENT tables
CREATE TABLE EMPLOYEE (
   Fname VARCHAR(15) NOT NULL,
   Minit CHAR(1),
   Lname VARCHAR(15) NOT NULL,
   Ssn CHAR(9) NOT NULL PRIMARY KEY,
   Bdate DATE,
   Address VARCHAR(30),
   Sex CHAR(1),
   Salary DECIMAL(10, 2),
   Super_ssn CHAR(9),
   Dno INT,
   FOREIGN KEY (Super_ssn) REFERENCES EMPLOYEE(Ssn),
   FOREIGN KEY (Dno) REFERENCES DEPARTMENT(Dnumber)
);

CREATE TABLE DEPARTMENT (
   Dname VARCHAR(15) NOT NULL UNIQUE,
   Dnumber INT NOT NULL PRIMARY KEY,
   Mgr_ssn CHAR(9) NOT NULL,
   Mgr_start_date DATE,
   FOREIGN KEY (Mgr_ssn) REFERENCES EMPLOYEE(Ssn)
);
```

```
-- 2. INSERT: Inserting data into the tables
INSERT INTO DEPARTMENT (Dname, Dnumber, Mgr_ssn, Mgr_start_date)
VALUES ('Research', 1, '123456789', '2020-01-01');

INSERT INTO EMPLOYEE (Fname, Minit, Lname, Ssn, Bdate, Address, Sex, Salary,
Super_ssn, Dno)
VALUES ('John', 'B', 'Doe', '987654321', '1980-05-15', '123 Elm St', 'M', 60000, NULL, 1);

-- 3. DELETE: Deleting records from the tables
-- Deleting an employee
DELETE FROM EMPLOYEE WHERE Ssn = '987654321';

-- Deleting a department
DELETE FROM DEPARTMENT WHERE Dnumber = 1;

-- 4. UPDATE: Updating records in the tables
-- Updating an employee's salary
UPDATE EMPLOYEE
SET Salary = 65000
WHERE Ssn = '987654321';

-- Updating a department's name
UPDATE DEPARTMENT
SET Dname = 'Development'
WHERE Dnumber = 1;

-- 5. ALTER: Altering the table structure
-- Adding a new column to the EMPLOYEE table
ALTER TABLE EMPLOYEE
ADD Email VARCHAR(50);

-- Dropping a column from the DEPARTMENT table
ALTER TABLE DEPARTMENT
DROP COLUMN Mgr_start_date;

-- 6. Entity Integrity Constraints
-- Primary Key Constraints ensure unique and non-null values
-- Attempt to insert a duplicate primary key (This will fail if '987654321' already exists)
INSERT INTO EMPLOYEE (Fname, Minit, Lname, Ssn, Bdate, Address, Sex, Salary,
Super_ssn, Dno)
VALUES ('Jane', 'A', 'Smith', '987654321', '1990-07-22', '456 Oak St', 'F', 70000, NULL, 1);

-- 7. Referential Integrity Constraints
-- Foreign Key Constraints ensure valid references between tables
```

```
-- Attempting to insert an employee with a non-existent department (This will fail if there is no department with Dnumber = 2)
INSERT INTO EMPLOYEE (Fname, Minit, Lname, Ssn, Bdate, Address, Sex, Salary, Super_ssn, Dno)
VALUES ('Alice', 'C', 'Johnson', '123123123', '1985-03-30', '789 Maple St', 'F', 80000, NULL, 2);



	- -q3
-- Query 0: Retrieve the birth date and address of the employee(s) whose name is ‘John B. Smith’.
SELECT Bdate, Address
FROM EMPLOYEE
WHERE Fname = 'John' AND Minit = 'B' AND Lname = 'Smith';

-- Query 1: Retrieve the name and address of all employees who work for the ‘Research’ department.
SELECT Fname, Lname, Address
FROM EMPLOYEE, DEPARTMENT
WHERE Dname = 'Research' AND Dnumber = Dno;

-- Query 2: For every project located in ‘Stafford’, list the project number, the controlling department number, and the department manager’s last name, address, and birth date.
SELECT Pnumber, Dnum, Lname, Address, Bdate
FROM PROJECT, DEPARTMENT, EMPLOYEE
WHERE Dnum = Dnumber AND Mgr_ssn = Ssn AND Plocation = 'Stafford';

-- Query 1A
SELECT Fname, EMPLOYEE.Name, Address
FROM EMPLOYEE, DEPARTMENT
WHERE DEPARTMENT.Name = 'Research' AND DEPARTMENT.Dnumber = EMPLOYEE.Dnumber;

-- Query 1′
SELECT EMPLOYEE.Fname, EMPLOYEE.LName, EMPLOYEE.Address
FROM EMPLOYEE, DEPARTMENT
WHERE DEPARTMENT.DName = 'Research' AND DEPARTMENT.Dnumber = EMPLOYEE.Dno;

-- Query 8: For each employee, retrieve the employee’s first and last name and the first and last name of his or her immediate supervisor.
SELECT E.Fname, E.Lname, S.Fname, S.Lname
FROM EMPLOYEE AS E, EMPLOYEE AS S
WHERE E.Super_ssn = S.Ssn;
```

```
-- Query 1B
SELECT E.Fname, E.LName, E.Address
FROM EMPLOYEE AS E, DEPARTMENT AS D
WHERE D.DName = 'Research' AND D.Dnumber = E.Dno;

-- Query 9: Select all EMPLOYEE Ssns.
SELECT Ssn
FROM EMPLOYEE;

-- Query 10: Select all combinations of EMPLOYEE Ssn and DEPARTMENT Dname.
SELECT Ssn, Dname
FROM EMPLOYEE, DEPARTMENT;

-- Query 11: Retrieve the salary of every employee.
SELECT Salary
FROM EMPLOYEE;

-- Query 11A: Retrieve all distinct salary values.
SELECT DISTINCT Salary
FROM EMPLOYEE;

-- Query 4: List all project numbers for projects involving an employee named 'Smith', either as a
worker or as a department manager.
(SELECT DISTINCT Pnumber
FROM PROJECT, DEPARTMENT, EMPLOYEE
WHERE Dnum = Dnumber AND Mgr_ssn = Ssn AND Lname = 'Smith')
UNION
(SELECT DISTINCT Pnumber
FROM PROJECT, WORKS_ON, EMPLOYEE
WHERE Pnumber = Pno AND Essn = Ssn AND Lname = 'Smith');

-- Query 12: Retrieve all employees whose address is in Houston, Texas.
SELECT Fname, Lname
FROM EMPLOYEE
WHERE Address LIKE '%Houston, TX%';

-- Query 12A: Find all employees who were born during the 1950s.
SELECT Fname, Lname
FROM EMPLOYEE
WHERE Bdate LIKE '195__%';

-- Query 13: Show the resulting salaries if every employee working on the 'ProductX' project is
given a 10% raise.
```

```
SELECT E.Fname, E.Lname, 1.1 * E.Salary AS Increased_sal
FROM EMPLOYEE AS E, WORKS_ON AS W, PROJECT AS P
WHERE E.Ssn = W.Essn AND W.Pno = P.Pnumber AND P.Pname = 'ProductX';

-- Query 14: Retrieve all employees in department 5 whose salary is between $30,000 and
$40,000.
SELECT *
FROM EMPLOYEE
WHERE Salary BETWEEN 30000 AND 40000 AND Dno = 5;

-- Query 15: Retrieve a list of employees and the projects they are working on, ordered by
department and, within each department, ordered alphabetically by last name, then first name.
SELECT D.Dname, E.Lname, E.Fname, P.Pname
FROM DEPARTMENT AS D, EMPLOYEE AS E, WORKS_ON AS W, PROJECT AS P
WHERE D.Dnumber = E.Dno AND E.Ssn = W.Essn AND W.Pno = P.Pnumber
ORDER BY D.Dname, E.Lname, E.Fname;
```