Name: Prince Chaudhary

USN: 1NT22CS143

Consider a relational database schema for a Company database below. Employee (F\_name: string, L\_name: string, Emp\_id:integer, Bdate: date, Address:string, Gender:string,Salary: integer, Super\_Emp\_id: integer, D\_no: integer) Department (D\_name: string, D\_no: integer, D\_Mgr\_id:integer, Mgr\_start\_date: date) Dept\_Location (D\_no: integer, D\_location:string) Project (P\_name: string, P\_number:integer, P\_location:string, D\_no:integer ) Works\_on (Emp\_id: integer, P\_no: integer, Hours: in) Dependent(Emp\_id:integer,Dependent\_name:string,Gender:string,Bdate:date,Relationship:String) For the above schema, perform the following a) Create the above tables by specifying primary keys and foreign keys. b) Insert around 10 records in each of the tables. c) Find the names and address of all employees who work on same department. d) 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.

Code:

use andrewdb;

CREATE TABLE Employee (

Emp\_id INTEGER PRIMARY KEY,

F\_name VARCHAR(255),

L\_name VARCHAR(255),

Bdate DATE,

Address VARCHAR(255),

Gender VARCHAR(10),

Salary INTEGER,

Super\_Emp\_id INTEGER,

D\_no INTEGER,

FOREIGN KEY (Super\_Emp\_id) REFERENCES Employee(Emp\_id)

);

CREATE TABLE Department (

D\_no INTEGER PRIMARY KEY,

D\_name VARCHAR(255),

D\_Mgr\_id INTEGER,

Mgr\_start\_date DATE,

FOREIGN KEY (D\_Mgr\_id) REFERENCES Employee(Emp\_id)

);

CREATE TABLE Dept\_Location (

D\_no INTEGER,

D\_location VARCHAR(255),

PRIMARY KEY (D\_no, D\_location),

FOREIGN KEY (D\_no) REFERENCES Department(D\_no)

);

CREATE TABLE Project (

P\_number INTEGER PRIMARY KEY,

P\_name VARCHAR(255),

P\_location VARCHAR(255),

D\_no INTEGER,

FOREIGN KEY (D\_no) REFERENCES Department(D\_no)

);

CREATE TABLE Works\_on (

Emp\_id INTEGER,

P\_no INTEGER,

Hours INTEGER,

PRIMARY KEY (Emp\_id, P\_no),

FOREIGN KEY (Emp\_id) REFERENCES Employee(Emp\_id),

FOREIGN KEY (P\_no) REFERENCES Project(P\_number)

);

CREATE TABLE Dependent (

Emp\_id INTEGER,

Dependent\_name VARCHAR(255),

Gender VARCHAR(10),

Bdate DATE,

Relationship VARCHAR(255),

PRIMARY KEY (Emp\_id, Dependent\_name),

FOREIGN KEY (Emp\_id) REFERENCES Employee(Emp\_id)

);

ALTER TABLE Employee

ADD CONSTRAINT FK\_Department\_D\_no

FOREIGN KEY (D\_no) REFERENCES Department(D\_no);

-- Inserting into Employee table

INSERT INTO Employee VALUES

(1, 'John', 'Doe', '1990-01-15', '123 Main St', 'Male', 60000, NULL, null),

(2, 'Jane', 'Smith', '1988-05-20', '456 Elm St', 'Female', 65000, 1, null),

(3, 'Michael', 'Johnson', '1992-03-12', '789 Oak St', 'Male', 55000, NULL, null),

(4, 'Emily', 'Brown', '1995-07-25', '101 Pine St', 'Female', 70000, 1, null),

(5, 'David', 'Williams', '1987-11-03', '222 Maple St', 'Male', 62000, 4, null);

INSERT INTO Department VALUES

(1, 'Engineering', 1, '2020-01-01'),

(2, 'Marketing', 2, '2019-06-01'),

(3, 'Finance', 3, '2020-03-15'),

(4, 'Human Resources', 4, '2018-11-10'),

(5, 'Research and Development', 5, '2017-09-05');

SET SQL\_SAFE\_UPDATES = 0;

UPDATE Employee

JOIN Department ON Employee.D\_no = Department.D\_no

SET Employee.D\_no = Department.D\_no;

INSERT INTO Dept\_Location VALUES

(1, 'New York'),

(2, 'Los Angeles'),

(3, 'Chicago'),

(4, 'Houston'),

(5, 'San Francisco');

INSERT INTO Project VALUES

(1, 'Project A', 'New York', 1),

(2, 'Project B', 'Los Angeles', 2),

(3, 'Project C', 'Chicago', 3),

(4, 'Project D', 'Houston', 4),

(5, 'Project E', 'San Francisco', 5);

INSERT INTO Works\_on VALUES

(1, 1, 40),

(2, 2, 35),

(3, 3, 30),

(4, 4, 25),

(5, 5, 20);

INSERT INTO Dependent VALUES

(1, 'Emily Doe', 'Female', '2015-03-10', 'Daughter'),

(2, 'Michael Smith', 'Male', '2018-07-20', 'Son'),

(3, 'Sophia Johnson', 'Female', '2020-01-05', 'Daughter'),

(4, 'Jacob Brown', 'Male', '2019-04-15', 'Son'),

(5, 'Olivia Williams', 'Female', '2017-08-12', 'Daughter');

show tables;

SELECT \* FROM Employee;

c.Finding the names and addresses of all employees who work in the same department:

SELECT E.F\_name, E.L\_name, E.Address FROM Employee E JOIN Department D ON E.D\_no = D.D\_no JOIN Employee E2 ON D.D\_no = E2.D\_no AND E.Emp\_id != E2.Emp\_id

D. d) Retrieving 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:

Code:

SELECT E.F\_name, E.L\_name, P.P\_name FROM Employee E JOIN Works\_on W ON E.Emp\_id = W.Emp\_id JOIN Project P ON W.P\_no = P.P\_number ORDER BY E.D\_no, E.L\_name, E.F\_name

Screenshots:



