RDBMS ASSIGNMENTS LCA-2

Pranav Nigade

1272240673

SLIP 15

Consider the following Entities & Relationships

Doctor (dno, dname, address, city)

Hospital (hno, hname, street, hcity)

Hospital and Doctor are related with many-to-many.

Constraints : Primary key ,Address not null

Create a RDB in 3NF & construct the queries in SQL Server/Oracle

- Find the no. of hospitals every doctor is visiting.
- Find the doctors visiting those hospitals that are in the same city in which they live.
- Display the count of the doctors visiting hospitals on JM Road.
- List hospital wise doctors.

OUTPUT

CREATE TABLE Doctor (dno INT PRIMARY KEY, dname VARCHAR(50), address VARCHAR(100) NOT NULL, city VARCHAR(50));

CREATE TABLE Hospital (hno INT PRIMARY KEY, hname VARCHAR(50), street VARCHAR(50), hcity VARCHAR(50));

CREATE TABLE Doctor_Hospital (dno INT, hno INT, PRIMARY KEY (dno, hno), FOREIGN KEY (dno) REFERENCES Doctor(dno), FOREIGN KEY (hno) REFERENCES Hospital(hno));

INSERTING VALUES

INSERT INTO Doctor (dno, dname, address, city) VALUES (1, 'Dr. Smith', '123 Elm St', 'Pune'), (2, 'Dr. Johnson', '456 Oak St', 'Mumbai'), (3, 'Dr. Patel', '789 Pine St', 'Pune');

INSERT INTO Hospital (hno, hname, street, hcity) VALUES (1, 'City Hospital', 'JM Road', 'Pune'), (2, 'General Hospital', 'MG Road', 'Mumbai'), (3, 'Sunshine Hospital', 'JM Road', 'Pune');

INSERT INTO Doctor_Hospital (dno, hno) VALUES (1, 1), (1, 3), (2, 2), (3, 1);

TABLE Doctor

```
mysql> SELECT * FROM Doctor;
+-----+-------------+-------------+--------+
| dno | dname       | address     | city   |
+-----+-------------+-------------+--------+
|   1 | Dr. Smith   | 123 Elm St  | Pune   |
|   2 | Dr. Johnson | 456 Oak St  | Mumbai |
|   3 | Dr. Patel   | 789 Pine St | Pune   |
+-----+-------------+-------------+--------+
3 rows in set (0.01 sec)
```

TABLE Hospital

```
mysql> SELECT * FROM Hospital;
+-----+-------------------+---------+--------+
| hno | hname             | street  | hcity  |
+-----+-------------------+---------+--------+
|   1 | City Hospital     | JM Road | Pune   |
|   2 | General Hospital  | MG Road | Mumbai |
|   3 | Sunshine Hospital | JM Road | Pune   |
+-----+-------------------+---------+--------+
3 rows in set (0.00 sec)
```

TABLE Doctor_Hospital

```
mysql> SELECT * FROM Doctor_Hospital;
+-----+-----+
| dno | hno |
+-----+-----+
|   1 |   1 |
|   3 |   1 |
|   2 |   2 |
|   1 |   3 |
+-----+-----+
4 rows in set (0.00 sec)
```

QUERIES

Find the no. of hospitals every doctor is visiting.

```
mysql> SELECT d.dname, COUNT(dh.hno) AS Hospital_Count FROM Doctor d LEFT JOIN Doctor_Hospital dh ON d.dno = dh.dno GROUP BY d.dname;
+-------------+----------------+
| dname       | Hospital_Count |
+-------------+----------------+
| Dr. Smith   |              2 |
| Dr. Johnson |              1 |
| Dr. Patel   |              1 |
+-------------+----------------+
```

Find the doctors visiting those hospitals that are in the same city in which they live:

```
mysql> SELECT DISTINCT d.dname FROM Doctor d JOIN Doctor_Hospital dh ON d.dno = dh.dno JOIN Hospital h ON dh.hno = h.hno WHERE d.city = h.hcity;
+-------------+
| dname       |
+-------------+
| Dr. Smith   |
| Dr. Johnson |
| Dr. Patel   |
+-------------+
3 rows in set (0.01 sec)
```

Display the count of the doctors visiting hospitals on JM Road:

```
mysql> SELECT COUNT(DISTINCT dh.dno) AS Doctor_Count FROM Doctor_Hospital dh JOIN Hospital h ON dh.hno = h.hno WHERE h.street = 'JM Road';
+--------------+
| Doctor_Count |
+--------------+
|            2 |
+--------------+
1 row in set (0.01 sec)
```

List hospital-wise doctors:

```
mysql> SELECT h.hname, d.dname FROM Hospital h JOIN Doctor_Hospital dh ON h.hno = dh.hno JOIN Doctor d ON dh.dno = d.dno ORDER BY h.hname, d.dname;
+-------------------+-------------+
| hname             | dname       |
+-------------------+-------------+
| City Hospital     | Dr. Patel   |
| City Hospital     | Dr. Smith   |
| General Hospital  | Dr. Johnson |
| Sunshine Hospital | Dr. Smith   |
+-------------------+-------------+
4 rows in set (0.01 sec)
```

SLIP 16

Consider the following Entities & Relationships

Car (cno,carmodel,ownername)

Driver (drvno, drvname, drvaddr, drvcity)

Accident (ac_id, place, year)

Car & Driver are related with many-to-many.

Car & Accident one-to-many.

Constraints : Primary key ,Foreign key, Accident year> 1900

Create a RDB in 3NF & construct the queries in SQL Server/Oracle

- List all accidents along with their place & cars involved in those accidents that took place in 2000.
- List the drivers involved in accidents that took place in “Pune”.
- Find the Place and year of accident that involved “Maruti-800”.

Display he driver names and the owner names of the cars which had accidents on Pune Nagar road.

OUTPUT

CREATE TABLE Car (cno INT PRIMARY KEY, carmodel VARCHAR(50), ownername VARCHAR(50));

CREATE TABLE Driver (drvno INT PRIMARY KEY, drvname VARCHAR(50), drvaddr VARCHAR(100), drvcity VARCHAR(50));

CREATE TABLE Accident (ac_id INT PRIMARY KEY, place VARCHAR(50), year INT CHECK (year > 1900));

CREATE TABLE Car_Driver (cno INT, drvno INT, PRIMARY KEY (cno, drvno), FOREIGN KEY (cno) REFERENCES Car(cno), FOREIGN KEY (drvno) REFERENCES Driver(drvno));

CREATE TABLE Car_Accident (cno INT, ac_id INT, PRIMARY KEY (cno, ac_id), FOREIGN KEY (cno) REFERENCES Car(cno), FOREIGN KEY (ac_id) REFERENCES Accident(ac_id));

INSERTING DATA

INSERT INTO Car (cno, carmodel, ownername) VALUES (1, 'Maruti-800', 'Rahul'), (2, 'Hyundai-i20', 'Sneha'), (3, 'Tata-Nexon', 'Amit');

INSERT INTO Driver (drvno, drvname, drvaddr, drvcity) VALUES (101, 'Pranav', 'MG Road', 'Pune'), (102, 'Saket', 'Laxmi Road', 'Mumbai'), (103, 'Vranda', 'Koregaon Park', 'Pune');

INSERT INTO Accident (ac_id, place, year) VALUES (201, 'Pune', 2000), (202, 'Pune Nagar road', 2001), (203, 'Mumbai', 2000);

INSERT INTO Car_Driver (cno, drvno) VALUES (1, 101), (1, 102), (2, 103), (3, 101);

INSERT INTO Car_Accident (cno, ac_id) VALUES (1, 201), (2, 202), (3, 201), (1, 203);

TABLE Car

```
mysql> SELECT * FROM Car;
+-----+------------+-----------+
| cno | carmodel   | ownername |
+-----+------------+-----------+
|   1 | Maruti-800 | Rahul     |
|   2 | Hyundai-i20| Sneha     |
|   3 | Tata-Nexon | Amit      |
+-----+------------+-----------+
3 rows in set (0.00 sec)
```

TABLE Driver

```
mysql> SELECT * FROM Driver;
+-------+---------+---------------+---------+
| drvno | drvname | drvaddr       | drvcity |
+-------+---------+---------------+---------+
|   101 | Pranav  | MG Road       | Pune    |
|   102 | Saket   | Laxmi Road    | Mumbai  |
|   103 | Vranda  | Koregaon Park | Pune    |
+-------+---------+---------------+---------+
3 rows in set (0.00 sec)
```

TABLE Accident

```
mysql> SELECT * FROM Accident;
+-------+----------------+------+
| ac_id | place          | year |
+-------+----------------+------+
|   201 | Pune           | 2000 |
|   202 | Pune Nagar road | 2001 |
|   203 | Mumbai         | 2000 |
+-------+----------------+------+
3 rows in set (0.00 sec)
```

TABLE Car_Driver

```
mysql> SELECT * FROM Car_Driver;
+-----+-------+
| cno | drvno |
+-----+-------+
|   1 |   101 |
|   3 |   101 |
|   1 |   102 |
|   2 |   103 |
+-----+-------+
4 rows in set (0.00 sec)
```

TABLE Car_Accident

```
mysql> SELECT * FROM Car_Accident;
+-----+-------+
| cno | ac_id |
+-----+-------+
|   1 |   201 |
|   3 |   201 |
|   2 |   202 |
|   1 |   203 |
+-----+-------+
4 rows in set (0.00 sec)
```

QUERIES

List all accidents along with their place & cars involved in those accidents that took place in 2000

```
mysql> SELECT A.ac_id, A.place, C.carmodel FROM Accident A JOIN Car_Accident CA ON A.ac_id = CA.ac_id JOIN Car C ON CA.cno = C.cno WHERE A.year = 20
00;
+-------+--------+-----------+
| ac_id | place  | carmodel  |
+-------+--------+-----------+
|   201 | Pune   | Maruti-800 |
|   201 | Pune   | Tata-Nexon |
|   203 | Mumbai | Maruti-800 |
+-------+--------+-----------+
3 rows in set (0.01 sec)
```

List the drivers involved in accidents that took place in "Pune".

```
mysql> SELECT DISTINCT D.drvname FROM Accident A JOIN Car_Accident CA ON A.ac_id = CA.ac_id JOIN Car_
Driver CD ON CA.cno = CD.cno JOIN Driver D ON CD.drvno = D.drvno WHERE A.place = 'Pune';
+----------+
| drvname |
+----------+
| Pranav  |
| Saket   |
+----------+
2 rows in set (0.00 sec)
```

Find the Place and year of accident that involved "Maruti-800".

```
mysql> SELECT A.place, A.year FROM Accident A JOIN Car_Accident CA ON A.ac_id = CA.ac_id JOIN Car C O
N CA.cno = C.cno WHERE C.carmodel = 'Maruti-800';
+--------+------+
| place  | year |
+--------+------+
| Pune   | 2000 |
| Mumbai | 2000 |
+--------+------+
2 rows in set (0.00 sec)
```

Display he driver names and the owner names of the cars which had accidents on Pune Nagar road.

```
mysql> SELECT D.drvname, C.ownername FROM Accident A JOIN Car_Accident CA ON A.ac_id = CA.ac_id JOIN
Car C ON CA.cno = C.cno JOIN Car_Driver CD ON C.cno = CD.cno JOIN Driver D ON CD.drvno = D.drvno WHER
E A.place = 'Pune Nagar road';
+---------+-----------+
| drvname | ownername |
+---------+-----------+
| Vranda  | Sneha     |
+---------+-----------+
1 row in set (0.00 sec)
```

SLIP 17

Consider the following Entities & Relationships

Doctor (dno, dname, address, city)

Patient (opdno, pat_name, addr, disease)

Patient and Doctor are related with many-to-many with descriptive attribute no_of_visits.

Constraints : Primary key , Address not null and no_of_visits <> 0

Create a RDB in 3NF & construct the queries in SQL Server/Oracle

- Find the no. of patients visited by “Dr.Apte”.
- Find the names of patients having “Cancer” and are visited by Mr Gandhi for more than 10 times.
- Find the no. of Patients suffering from “Asthama”.
  List doctor wise patients.

OUTPUT

CREATE TABLE Doctor (dno INT PRIMARY KEY, dname VARCHAR(50), address VARCHAR(100) NOT NULL, city VARCHAR(50));

CREATE TABLE Patient (opdno INT PRIMARY KEY, pat_name VARCHAR(50), addr VARCHAR(100) NOT NULL, disease VARCHAR(50));

CREATE TABLE Doctor_Patient (dno INT, opdno INT, no_of_visits INT CHECK (no_of_visits <> 0), PRIMARY KEY (dno, opdno), FOREIGN KEY (dno) REFERENCES Doctor(dno), FOREIGN KEY (opdno) REFERENCES Patient(opdno));

INSERT DATA

INSERT INTO Doctor (dno, dname, address, city) VALUES (1, 'Dr. Apte', '123 Elm St', 'Pune'), (2, 'Mr. Gandhi', '456 Oak St', 'Mumbai');

INSERT INTO Patient (opdno, pat_name, addr, disease) VALUES (101, 'John Doe', '789 Pine St', 'Cancer'), (102, 'Jane Smith', '456 Cedar St', 'Asthama'), (103, 'Alice Brown', '321 Maple St', 'Cancer');

INSERT INTO Doctor_Patient (dno, opdno, no_of_visits) VALUES (1, 101, 5), (1, 102, 3), (2, 103, 15), (2, 101, 12);

TABLE Doctor

```
mysql> SELECT * FROM Doctor;
+-----+------------+-----------+--------+
| dno | dname      | address   | city   |
+-----+------------+-----------+--------+
|   1 | Dr. Apte   | 123 Elm St | Pune   |
|   2 | Mr. Gandhi | 456 Oak St | Mumbai |
+-----+------------+-----------+--------+
2 rows in set (0.00 sec)
```

TABLE Patient

```
mysql> SELECT * FROM Patient;
+-------+-------------+--------------+---------+
| opdno | pat_name    | addr         | disease |
+-------+-------------+--------------+---------+
|   101 | John Doe    | 789 Pine St  | Cancer  |
|   102 | Jane Smith  | 456 Cedar St | Asthama |
|   103 | Alice Brown | 321 Maple St | Cancer  |
+-------+-------------+--------------+---------+
3 rows in set (0.00 sec)
```

TABLE Doctor_Patient

```
mysql> SELECT * FROM Doctor_Patient;
+-----+-------+--------------+
| dno | opdno | no_of_visits |
+-----+-------+--------------+
|   1 |   101 |            5 |
|   1 |   102 |            3 |
|   2 |   101 |           12 |
|   2 |   103 |           15 |
+-----+-------+--------------+
4 rows in set (0.00 sec)
```

QUERIES

Find the number of patients visited by “Dr. Apte”:

```
mysql> SELECT COUNT(DISTINCT dp.opdno) AS Patient_Count FROM Doctor d JOIN Doctor_Patient dp ON d.dno = dp.dno WHERE d.dname = 'Dr. Apte';
+---------------+
| Patient_Count |
+---------------+
|             2 |
+---------------+
1 row in set (0.00 sec)
```

Find the names of patients having “Cancer” and visited by Mr. Gandhi more than 10 times:

```
mysql> SELECT p.pat_name FROM Patient p JOIN Doctor_Patient dp ON p.opdno = dp.opdno JOIN Doctor d ON dp.dno = d.dno WHERE p.disease = 'Cancer' AND d.dname
= 'Mr. Gandhi' AND dp.no_of_visits > 10;
+-------------+
| pat_name    |
+-------------+
| John Doe    |
| Alice Brown |
+-------------+
2 rows in set (0.01 sec)
```

Find the number of patients suffering from “Asthama”:

```
mysql> SELECT COUNT(*) AS Patient_Count FROM Patient WHERE disease = 'Asthama';
+----------------+
| Patient_Count |
+----------------+
|              1 |
+----------------+
1 row in set (0.01 sec)
```

List doctor-wise patients:

```
mysql> SELECT d.dname AS Doctor_Name, p.pat_name AS Patient_Name FROM Doctor d JOIN Doctor_Patient dp ON d.dno = dp.dno JOIN Patient p ON dp.opdno = p.opdno
 ORDER BY d.dname, p.pat_name;
+-------------+--------------+
| Doctor_Name | Patient_Name |
+-------------+--------------+
| Dr. Apte    | Jane Smith   |
| Dr. Apte    | John Doe     |
| Mr. Gandhi  | Alice Brown  |
| Mr. Gandhi  | John Doe     |
+-------------+--------------+
4 rows in set (0.00 sec)
```