ASSIGNMENT 10

Q1:Convert Collage Management System Er diagram to table?

Rules to convert er to table

Rule 1: A strong entity set with only simple attribute will require only one table in relational model

Rule 2: A strong entity set with any number of composite attribute will require only one table in relation model, composite attribute will convert into simple attribute in table

Rule 3: A strong entity set with any number of multi-valued attribute will require two tables in relational model

i) One table will contain all the simple attributes with the primary key

ii)Other table will contain the primary key and all the multi valued attribute

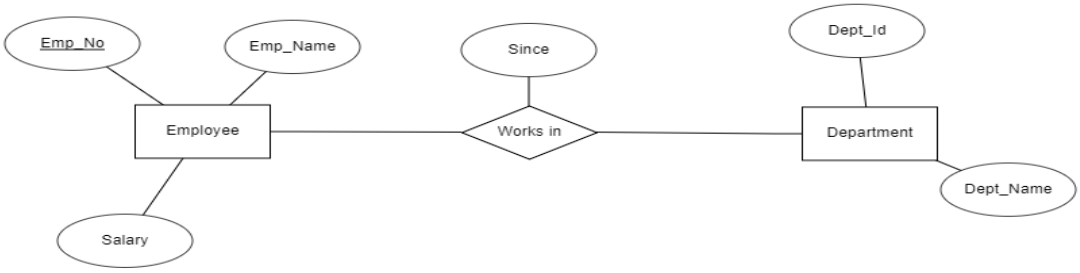
Rule 4: A relationship set will require one table in the relation model

i) attibute of the table are:

Primary key attributes of the participating entity set

Its own description attributes if any

ii) Set of non-descriptive attributes will be primary key



iii)For given er diagram,three table will be required in relational model

One table for entity set "Employee"

One table for entity set "Department"

One table for entity set "Works in"

Rule 5: For Binary Relationships with Cardinality Ratio

Case 1: Many-to-Many

In many to many relationship three tables will be required'

tabel1(a1,a2), Relationtable(a1,b1), table3(b1,b2)

Case 2: One-to-Many

In One-to-Many Two tables are required and second table and relation table make single table Table1(a1,a2), Relation-and-Table2(b1,b2,a1)

Case 3: Many-to-One

In Many-to-One relationship required Two Tables RelationTable-and-Table1(a1,a2,b1) , Table2(b1,b2)

Case 4:One-to-One

In One-to-One relationship Two tables will be required.Either combine 'R'

with A or B table

i) RelationshipTable-and-Table1(a1,a2,b1) and Table2(b1,b2)

ii)Table1(a1,a2) and RelationTable-and-Table2(b1,b2,a1)

Rule 6: for Binary Relationship with both Cardinality Constraints and Participation Constraints

Cardinality constraints will be discussed in Rule 5

Because of the total participation constraint,foreign key acquires NOT NULL constraint i.e now foreign key can not be null

Two Case

Case 1: For binary Relationship with cardinality Constraints and Total Participation Constraints from one Side

Case 2: For binary Relationship with cardinality Constraints and Total participation from Both Sides

```
mysql> show tables;
+-----------------------------------+
| Tables_in_collage_management_system |
+-----------------------------------+
| cources                           |
| department                        |
| enrols                            |
| exam                              |
| faculty                           |
| faculty_contact_merging           |
| hostel                            |
| student                           |
| student_contact_merging           |
| subject                           |
+-----------------------------------+
10 rows in set (0.00 sec)
```

All Tables of Collage_Management_System

i)Hostel Table

```
mysql> desc Hostel;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| Hostel_id   | int         | NO   | PRI | NULL    |       |
| No_Of_Seats | varchar(233) | YES  |     | NULL    |       |
| Hostel_Name | varchar(233) | YES  |     | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
3 rows in set (0.01 sec)
```

Hostel Table Has Three attribute(Hostel_id,No_of_Seats,Hostel_Name) and Hostel Table has One-to-Many Relationship with Student so Rule-5 Case-1 is followed

ii)Student Table

```
mysql> create table Student
    -> ( S_id int primary key,
    -> Address varchar(233),
    -> DOB varchar(233),
    -> First_Name Varchar(233),
    -> Last_Name varchar(233),
    -> Fk_Hostel_id  int,
    -> FOREIGN KEY(Fk_Hostel_id) REFERENCES Hostel(Hostel_id));
Query OK, 0 rows affected (0.23 sec)
```

```
mysql> desc Student;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| S_id        | int         | NO   | PRI | NULL    |       |
| Address     | varchar(233) | YES  |     | NULL    |       |
| DOB         | varchar(233) | YES  |     | NULL    |       |
| First_Name  | varchar(233) | YES  |     | NULL    |       |
| Last_Name   | varchar(233) | YES  |     | NULL    |       |
| Fk_Hostel_id | int        | YES  | MUL | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
6 rows in set (0.01 sec)
```

Student Table has 6 attributes(S_id,Address,Dob,First_Name,Last_Name,Fk_Hostel_id) and Student table has Many-to-One Relationship with Hostel and Student table followed Rule-3 and Rule-5 Case-3

iii)Faculty Table

```
mysql> desc faculty;
+---------------+-------------+------+-----+---------+-------+
| Field         | Type        | Null | Key | Default | Extra |
+---------------+-------------+------+-----+---------+-------+
| fid           | int         | NO   | PRI | NULL    |       |
| Name          | varchar(233) | YES  |     | NULL    |       |
| Department    | varchar(233) | YES  |     | NULL    |       |
| Salary        | int         | YES  |     | NULL    |       |
| department_id | int         | YES  | MUL | NULL    |       |
+---------------+-------------+------+-----+---------+-------+
5 rows in set (0.00 sec)
```

Faculty Table has 5 attribute(fid,Name,Department,salary,Department_id) and Faculty table has One-to-Many relationship with Student and Many-to-One relationship with Department and Faculty table followed Rule-2 and Rule 5 Case-2 and Case-3

iv)Faculty_Contact_Merging Table

```
mysql> desc Faculty_Contact_Merging
    -> ;
+-----------+------+------+-----+---------+-------+
| Field     | Type | Null | Key | Default | Extra |
+-----------+------+------+-----+---------+-------+
| Mobile_No | int  | NO   | PRI | NULL    |       |
| fid       | int  | YES  | MUL | NULL    |       |
+-----------+------+------+-----+---------+-------+
2 rows in set (0.01 sec)
```

Faculty_Contact_merging has 2 attribute(Mobile_No,fid) and This Table is created because Faculty Table has Multi-Valued Attribute Phone_Number So Faculty table followed Rule-3

v)Cources Table

```
mysql> desc cources;
+---------------+-------------+------+-----+---------+-------+
| Field         | Type        | Null | Key | Default | Extra |
+---------------+-------------+------+-----+---------+-------+
| cid           | int         | NO   | PRI | NULL    |       |
| Duration      | varchar(233)| YES  |     | NULL    |       |
| Cource_Name   | varchar(233)| YES  |     | NULL    |       |
| department_id | int         | YES  | MUL | NULL    |       |
+---------------+-------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
```

Cources Table has 4 attribute(cid,Duration,Cource_Name,Department_id) and Cource table has Many-to-Many relationship with Student and Many-to-One Relationship with Department and Cource table followed Rule-5 Case 1 and Case 3

vi)Enrols Relationship table

```
mysql> create table enrols
    -> ( S_id int primary key,
    -> cid int unique,
    -> foreign key(S_id) references Student(S_id),
    -> foreign key(cid) references Cources(cid));
Query OK, 0 rows affected (0.21 sec)

mysql> desc enrols;
+-------+------+------+-----+---------+-------+
| Field | Type | Null | Key | Default | Extra |
+-------+------+------+-----+---------+-------+
| S_id  | int  | NO   | PRI | NULL    |       |
| cid   | int  | YES  | UNI | NULL    |       |
+-------+------+------+-----+---------+-------+
2 rows in set (0.01 sec)
```

Enrols table has 2 attribute(S_id,cid) and it show Many-to-Many relationship with Student and Cources Table and it followed Rule 5 Case 1

vii)Department Table

```
mysql> create table department
    -> ( department_id int primary key,
    -> department_name varchar(233));
Query OK, 0 rows affected (0.16 sec)
```

```
mysql> desc department;
+-----------------+-------------+------+-----+---------+-------+
| Field           | Type        | Null | Key | Default | Extra |
+-----------------+-------------+------+-----+---------+-------+
| department_id   | int         | NO   | PRI | NULL    |       |
| department_name | varchar(233) | YES  |     | NULL    |       |
+-----------------+-------------+------+-----+---------+-------+
2 rows in set (0.01 sec)
```

Department table has 2 attribute(department_id,department_Name) and Department table has One-to-Many relationship with(Cources,Faculty,Exam)Tables and Followed Rule 5 Case 2

viii) Exam Table

```
mysql> create table Exam
    -> ( exam_code int primary key,
    -> datee date,
    -> time varchar(233),
    -> Room int);
Query OK, 0 rows affected (0.24 sec)

mysql> alter table exam add department_id int;
Query OK, 0 rows affected (0.11 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> alter table foreign key(department_id) references department(department_id);
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to y
tment_id) references department(department_id)' at line 1
mysql> alter table exam foreign key(department_id) references department(department_id);
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to y
tment_id) references department(department_id)' at line 1
mysql> alter table exam add foreign key(department_id) references department(department_id);
Query OK, 0 rows affected (0.69 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc exam;
+---------------+-------------+------+-----+---------+-------+
| Field         | Type        | Null | Key | Default | Extra |
+---------------+-------------+------+-----+---------+-------+
| exam_code     | int         | NO   | PRI | NULL    |       |
| datee         | date        | YES  |     | NULL    |       |
| time          | varchar(233)| YES  |     | NULL    |       |
| Room          | int         | YES  |     | NULL    |       |
| department_id | int         | YES  | MUL | NULL    |       |
+---------------+-------------+------+-----+---------+-------+
5 rows in set (0.01 sec)
```

Exam Table has 4 attribute(exam_code,datee,time,Room) and has Many-to-One relationship with Department and Followed Rule 5 Case 3

ix)Student_Contact_Merging Table

```
mysql> create table student_contact_merging
    -> ( S_id int,
    -> Phone_N int primary key);
Query OK, 0 rows affected (0.17 sec)

mysql> desc student_contact_merging;
+---------+------+------+-----+---------+-------+
| Field   | Type | Null | Key | Default | Extra |
+---------+------+------+-----+---------+-------+
| S_id    | int  | YES  |     | NULL    |       |
| Phone_N | int  | NO   | PRI | NULL    |       |
+---------+------+------+-----+---------+-------+
2 rows in set (0.01 sec)
```

Student_Contact_Merging table has 2 attribute(S_id,Phone_N) this table is created because Student table has Multi-Valued-Attribute and according to Rule 3 We have to make Third table for Multi-Valued-Attribute

x) Subject Table

```
mysql> create table subject
    -> (subject_id int primary key,
    -> subject_name varchar(233),
    -> fid int,
    -> foreign key(fid) references faculty(fid));
Query OK, 0 rows affected (0.20 sec)

mysql> desc subject;
+--------------+--------------+------+-----+---------+-------+
| Field        | Type         | Null | Key | Default | Extra |
+--------------+--------------+------+-----+---------+-------+
| subject_id   | int          | NO   | PRI | NULL    |       |
| subject_name | varchar(233) | YES  |     | NULL    |       |
| fid          | int          | YES  | MUL | NULL    |       |
+--------------+--------------+------+-----+---------+-------+
3 rows in set (0.01 sec)
```

Subject table has 3 attribute(subject_id,subject_name,fid) and Subject table has Many-to-One relationship with Faculty and Followed Rule-5 Case 3