MYSQL Management

Name of student

Name of professor

University

Course

Date

**INTRODUCTION:**

The first bit of this project is to define the data requirements of the database.

Dataset:

We shall implement a fictitious database within the hospital called the ***hospital\_db****:*

Which shall have the following related table-object classes:

1. Patients table
2. Doctors table
3. Locations table
4. Roles table
5. Prescriptions table
6. Suppliers table
7. Wards table
8. Appointments table
9. Payments table
10. Diagnosis table
11. Ambulances table
12. Inventories table

Now for the next step, we shall try to come upon with some factitious data on Microsoft Excel so that we can try to see how we shall be able to describe our dataset column variables. A quick summary of the dataset looks as below:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Patient\_date | P\_name | p\_location | p\_phone | p\_email | P\_address | P\_visit\_date | Is\_new |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

The sample dataset table above describes the patient table from which we can draw certain variables as we shall see later:

The flowing below is the table for the appointment table class:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| App\_id | Appoint\_date | Patient\_id | Doctor\_id | Is\_repeat |
|  |  |  |  |  |
|  |  |  |  |  |

**BUSINESS RULES**

The next step is to define our business rules for these database objects:

The database schema and rules dictate what can be done by each and every database user based on the intended business requirement. They are more of role based permissions:

1. ONE patient can book MANY appointments
2. ONE patient can be treated by MANY doctors
3. MANY doctors can treat MANY patients
4. ONE appointment can only have ONE patient
5. MANY payments can have MANY patients
6. ONE supplier can supply MANY drugs

The business rules should also be sensible and make sense as seen above.

**3RD NORMAL FORM NOTATION**

In order to set the two tables above in 3RD Normal form, we shall have a primary key and secondary key (foreign key) on the second column. We shall remove transitive depended from any of the tables and as such also remove the functional deaneries that might occur across the table son that we achieve THIRD NORMAL FORM. We should also have only unique rows among the primary keys on the table. The final results with sample data should looks as below for the two tables:

-***Patients table***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| P\_id | P\_name | p\_location | p\_phone | p\_email | P\_dob | P\_visit\_date | Is\_new |
| 001 | Alex | Rivers | +123456 | a@dummy.com | 1974.4.1 | 2022-04-03 | Y |
| 002 | Boni | Aknston | +12344 | a@dummy.com |  | 2022-03-01 | Y |
| 003 | Milca | Liman | +13455 | a@dummy.com |  | 2022-01-24 | N |

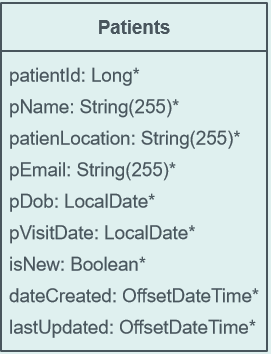
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| App\_id | Appoint\_date | Patient\_id | Doctor\_id | Is\_repeat |
| 2001 | 2022-05-04 | 001 | 9004 | Y |
| 2002 | 2022-06-03 | 003 | 9008 | Y |

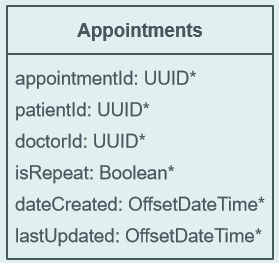
So as can be seen from the two tables above, we have eliminated any transitive dependencies and now we are left with values that are ready for relational database insertions.

**ERD DIAGRAMS**

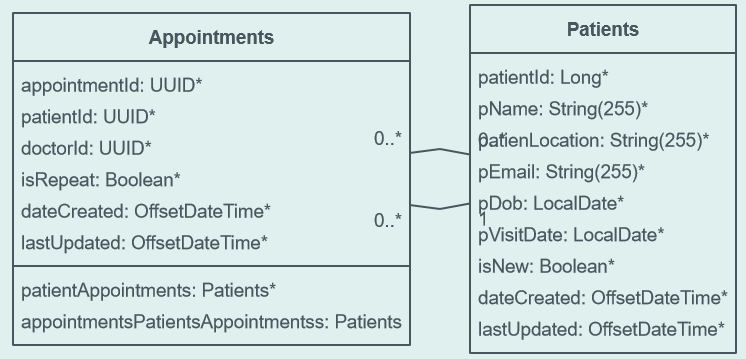
The next step now is to come up with the associated ERD diagrams for the two tables. The two tables above shall be translated into ERD diagrams as below:

The logic Diagrams for the two tables look as below:





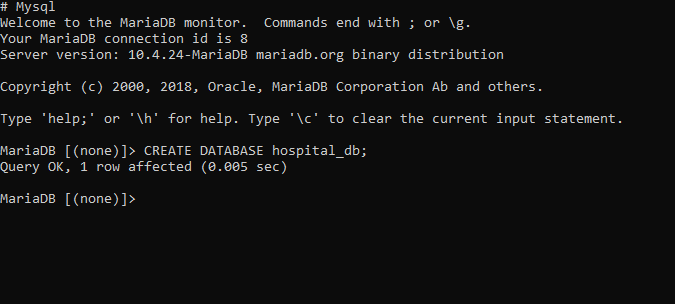
Then the relationship comes as below:



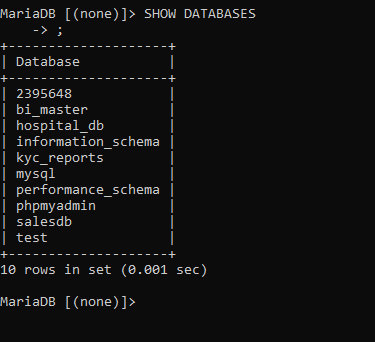
**SQL EXECUTIONS AND COMMANDS**

**The first process is to create the two tables:**

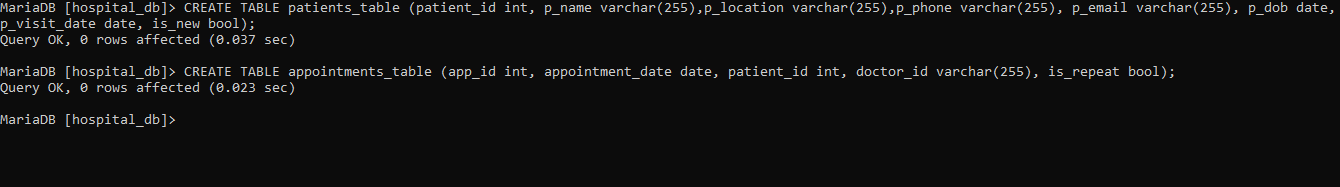
CREATE DATABASE hospital\_db;

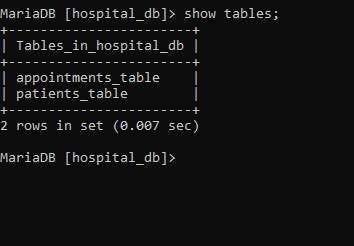


Show the database that we have created:

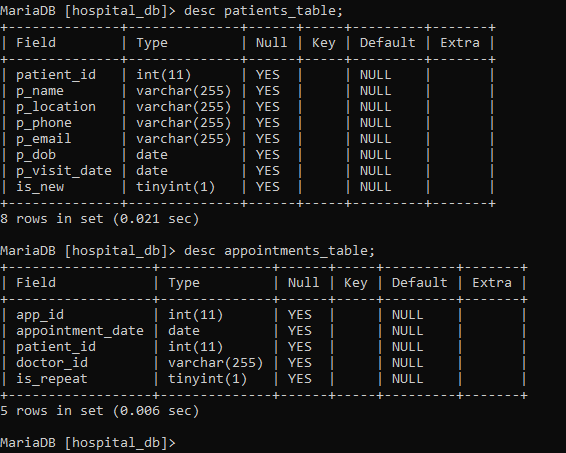


Create tables;





Show the structure of the tables;



Insert records;

 --INSERT RECORDS

 insert into patients\_table (patient\_id, p\_name,p\_location,p\_phone,p\_email,p\_dob,p\_visit\_date,is\_new) values('001','alex','Rivers','+12345678','alexis@dummy.com','1974-04-01','2022-04-03','Y')