**Database Management Systems (BCPR203)**

**Design Assignment**

**Sysmex Hospital Database**

The data given in the spreadsheet was segregated to form six different entities, which includes RefereeType, Referee, Patient, Department, Surgeonss and Reference. From the given spreadsheet, year and month was not included as an attribute since the referral date contains the year and month in it. The refereeTypeID, refereeID, patientID, deptID, empID, waitListPatientID is the externally added columns and they are the primary key it is auto incremented as well as NOT NULL. The Name of the entities such as (patientName, referedBy, SurgeonName) are divided into first name and last name. The date value should be in the format of yy-mm-dd.

All the above mentioned entities were obtained through Normalisation.

During the First normal form (1NF) all the given entities were together, which was then further segregated to two more forms and hence resulted in obtaining entities to create an ERD.

**Normalisation**

**Partial Dependency**: One of the primary key determines the other attribute.

**Transitive Dependency**: Non-key attribute determines another attribute.

**First Normal Form(1NF):**

In the first normal form, two entities are obtained which is the primary key.

* **ReferenceID** has the partial dependency with the refDate, refTypeID, refFrom, refereeID, refFirstname, refLastName, FSA, WLD. In which RefereeID is the Primary Key. Where the ReferenceID is unique and it can be chosen as the primary key so that other references related will form a partial dependency
* **PatientID** has the partial dependency with patientFirstName, patientLastName, DOB, gender, NHI, HTA. In which PatientID is the Primary Key. PatientID is unique and it can be used to link with patient details, department, surgeon. hence it is chosen as the primary key
* **DepartmentID** has the transitive dependency with depName.
* **SurgeonID** has the transitive dependency with Surgeon First Name, Surgeon Last Name, empID.
* RefTypeID and refFrom have transitive relation, RefereeID, refereeFirstname and refereeLastName have transitive relation.

**Second Normal Form(2NF)**

In the second normal the non key attribute is dependent on primary key. Where the refereeID, refTypeID, depID and empID are transitive

**Third Normal Form(3NF)**

In the third Normal form six entities are obtained to draw the ERD (ER diagram).

RefereeTypeID, refereeID, reference, patient, surgeon, department

The Primary key of one sided table will be the foreign key of the other sided table.

**Cardinality and Relationships**

From the seven entities, the relationships are obtained.

Patient

Referee

Reference

RefereeType

Surgeon

Department

From the above-mentioned ERD the relationships between each entity is understood clearly. The cross feet is used to indicate the many to one relation.

* RefereeType and the Referee has the one: many relations. The RefereeType can have many referee or one referee, whereas the referee has one and only one RefereeType.
* Referee and the Reference has the one: much relation. The Referee can have many reference or one reference, whereas the reference has one and only one referee.
* Reference and Patient has many: one relation. The Reference has one and only one patient, whereas the patient can have many or one reference.
* Reference and Surgeon has many: one relation. The reference can have one and only one Surgeon, whereas the Surgeon can have many or one reference.
* Surgeon and Department has many: one relation. The surgeon has one and only one Department, where as the department can have many or one surgeon.

Reference acts as the bridging table in the diagram.

**Database**

Six tables where created for the database “HospitalSysmexDB”

RefereeType, Referees, Patient, Department, Surgeonss and Reference2

|  |
| --- |
| RefereeType: refereeTypeID primary key (PK) + refereeFrom |
| Referees : refereeID (PK) + refereeTypeID foreign Key (FK) + refFirstName+ refLastName |
| Patient : patientID (PK) + patientFirstName+ patientLastName+ dob+ gender+ NHI+ HTA |
| Department: depID (PK) + depName |
| Surgeonss : empID (PK) + depID+ surgeonFirstName+ surgeonLastName+ depID (FK) |
| Reference2 : refID (PK)+ refDate+ FSA+ WLD+ empID+ refereeID+ patientID+ empID (FK)+ refereeID(FK)+ patientID(FK) |

**Insert** statement is used to insert the values into all the created tables and appropriate values are entered into each table. When the values are entered it can be displayed by using the **select** statement.

Once all the tables are created and values are entered into the table the queries are used to obtain the appropriate result.

Five query questions is answered in the Database.

1. How many people have been referred for surgery?

The question was to count the list of people referred for the surgery.

The query is derived with count to be Surgery Referees. It is used to find the total number of people for the surgery.

1. What is the average time taken to see a Surgeon by Department?

The average time to see a surgeon by a patient. Where the average time is taken through the difference between First specialist appointment date (FSA) and refDate from the reference table. EmpID and depID is joined by using a **JOIN** clause from surgeon and department. **GROUP** clause is used to as an aggregate function to group the result set.

1. Who has each Surgeon had on their list and how long have they been waiting or did they wait?

The query is to find the surgeons who have the list of patients on their waiting list, which also includes the refDate of waiting. The data also includes the name of the surgeons who has the list of patient. Concat is used to find the name of the patients. Datediff between FSA and WLD is used to find the waiting time of the patients. Join is used to join between patients and surgeons. Group is used to list the group of surgeons.

1. Assuming that all patients under 18 need to be seen by Paediatric Surgery, are there any patients who need to be reassigned?

The question is to check if there are any patients who are under 18 yet to be reassigned to paediatric surgery.

The dob of the patient is used to check the list of patients under 18 and those who are reassigned, the first and last name of the patient is concatenated by using CONCAT clause where the name of the patient can be identified. Join is then used to join between two columns reference and surgeon are joined. DOB > 2000 is used to see the patients who are under 18. The surgeonID for paediatric is 5 and not equal (<>) operator is used to find the list the list the of patients under paediatric.

1. What percentage of patient were seen within the target of 80 days by department?

The query is to find the percentage of patients for the target of 80 days in each department. The date difference between FSA and WLD is used. Which is multiplied with 80 and divided by 100. InnerJoin is used between surgeonss and department to find the department name which have the patients within the target of 80 days.

The table does not contain any subtype/supertype

**GitHub Version Control**

The GitHub version control contains the evidence of the work done throughout the assignment. The link is given below.

<https://github.com/SwathiSoman/DBMS_assignment1.git>

Invite Link : <https://github.com/SwathiSoman/DBMS_assignment1/invitations>