|  |
| --- |
| Charity Data Base project |
|  |
| **Team Members:**   1. **محمود محمد طه أبوعياد** 2. **حسام الدين أنسى** 3. **طارق فهيم** 4. **محمد على يوسف الخياط** |

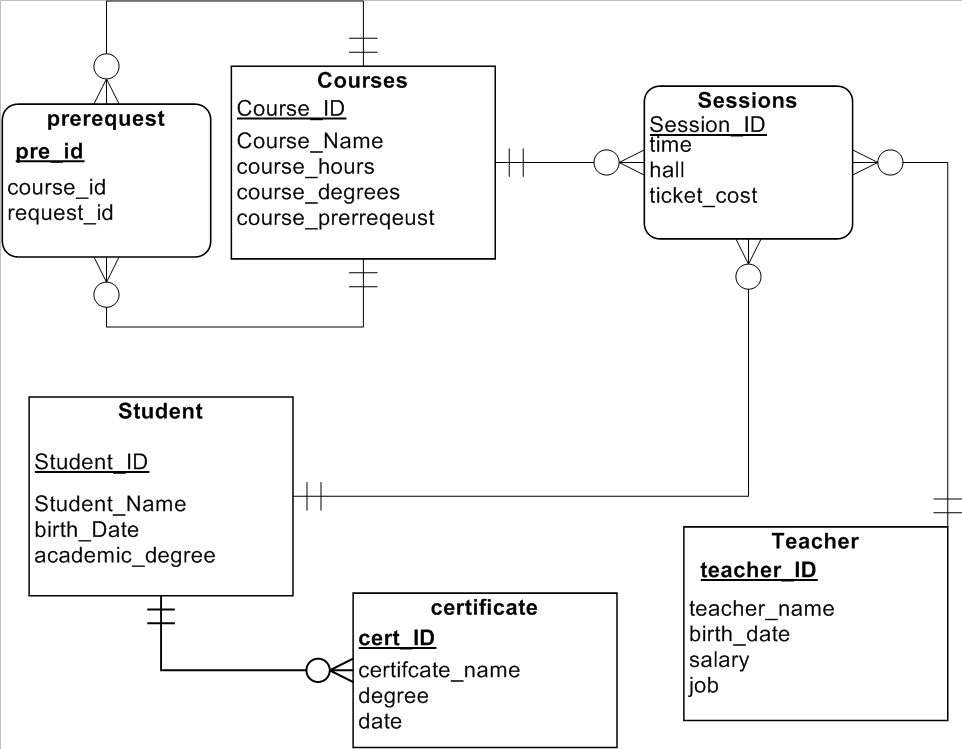
2016

Abstraction

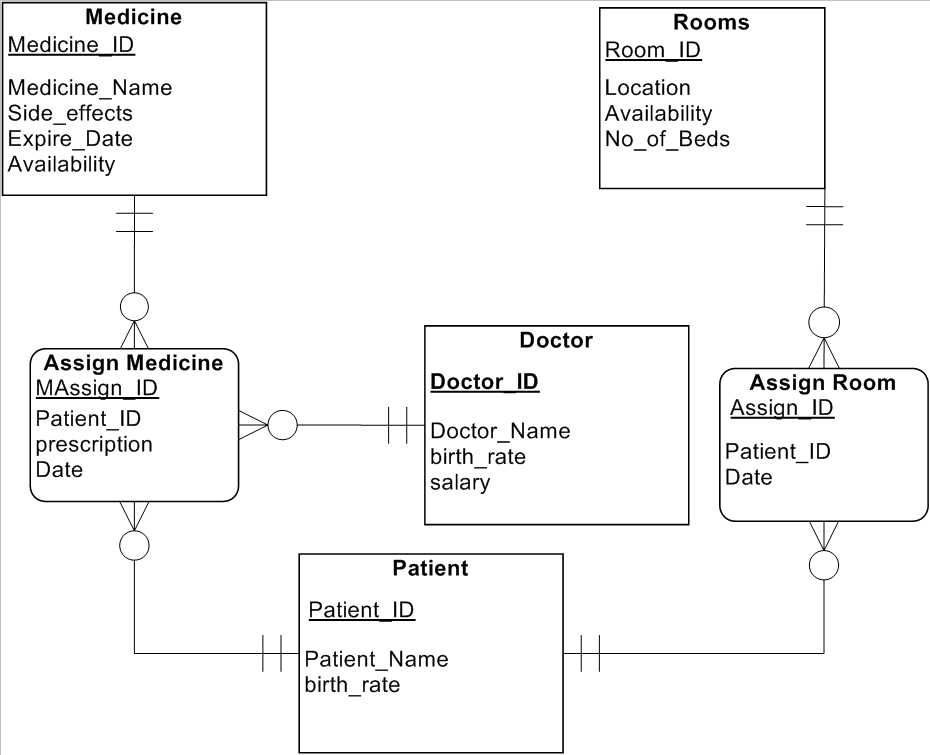
**Our project aims to store and restore all data needed for the charity system and safely update and delete from data**

Project components

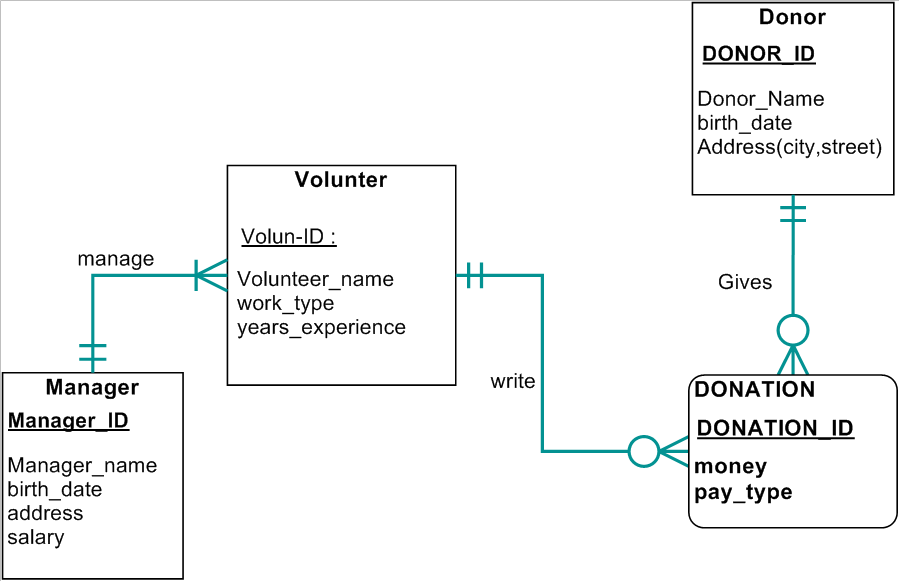
1. **ER diagrams**
2. **Relational diagrams**
3. **Documentation for each table**
4. **SQL scripts of the implementation of the ER diagrams**
5. **GUI to connect to the database with the user**
6. ER Diagrams
7. **For the educational sub-system**

****

1. **For the medical sub-system**

****

1. **For the Learning sub-system**

****

1. Relational diagrams

Learning sub-system

Prerequest

|  |  |  |
| --- | --- | --- |
| Prerequest\_ID | Course\_id | Prequest\_id |

Courses

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Course\_ID | Course\_Name | Course\_hours | Course\_degrees | Course\_prerequest |

Sessions

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Session\_ID­ | Course\_ID | time | hall | Ticket\_cost | Student\_id | Teacher\_id |

Student

|  |  |  |  |
| --- | --- | --- | --- |
| Student\_ID | Student\_Name | Birth\_date | Academic\_degree |

Certificate

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| certificate\_ID | certificate \_Name | Date | Degree | Student\_id |

Teacher

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| teacher\_ID | teacher \_Name | Birth\_date | salary | job |

Medical sub-system

Medicine

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Medicine\_ID | Medicine\_Name | Side\_effects | Expire\_Date | Availability |

Assign Medicine

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| MAssign\_ID | Patient\_ID | prescription | Date | Medicine\_id | Doctor\_id |

Patient

|  |  |  |
| --- | --- | --- |
| Patient\_ID | Patient\_Name | Patient\_Age |

Assign Room

|  |  |  |  |
| --- | --- | --- | --- |
| Assign\_ID | Patient\_ID | Date | Room\_id |

Rooms

|  |  |  |  |
| --- | --- | --- | --- |
| Room\_ID | Location | Availability | No\_of\_Beds |

Doctor

|  |  |  |  |
| --- | --- | --- | --- |
| Doctor\_ID | Doctor\_Name | Birth\_date | Salary |

Fund raising sub-system

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Manager\_ID | Manager\_Name | Birth\_date | Address | Salary |

Manager

Volunteer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Volunteer\_ID | Volunteer\_name | Work\_type | Years\_experience | Manager\_id |

Donation

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Donation\_ID | Money | Pay\_type | Donor\_id | Volunteer\_id |

Donor

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Donor\_ID | Donor\_name | Address | Brith\_date | Address |

1. Documentation of the system
2. Learning sub-system

Courses   
Here we get the course required for the case the system store all its data and courses

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Course | | | | | |
| Field | Type | Primary Key | Secondary Key | Null | Comments |
| Course\_ID | Int (11) | Yes |  | No | Identifier Key |
| Coures\_Name | Varchar2(30) |  |  | No | Name of this course |
| Course\_degrees | Varchar2(30) |  |  | No | Degrees of the course |

Student   
Here we get the students required for the case the system store all its data and students

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| student | | | | | |
| Field | Type | Primary Key | Secondary Key | Null | Comments |
| student\_ID | Int (11) | Yes |  | No | Identifier Key |
| studnet\_Name | Varchar2(30) |  |  | No | Get the name |
| academic\_degrees | Varchar2(30) |  |  | No | Degrees of the course |
| Birth\_date | Date |  |  | No | Identified the birth date |

Teacher  
Here we get the teachers required for the case the system store all its data and teachers

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| teacher | | | | | |
| Field | Type | Primary Key | Secondary Key | Null | Comments |
| teacher\_ID | Int (11) | Yes |  | No | Identifier Key |
| teacher\_Name | Varchar2(30) |  |  | No | Get the name |
| Salary | Int(11) |  |  | No | Identifies the salary |
| Birth\_date | Date |  |  | No | Identified the birth date |

Certificate   
Here we get the Certificate s required for the case the system store all its data and Certificates

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Certificate | | | | | |
| Field | Type | Primary Key | Secondary Key | Null | Comments |
| Certificate \_ID | Int (11) | Yes |  | No | Identifier Key |
| Certificate \_Name | Varchar2(30) |  |  | No | Get the name |
| academic\_degrees | Varchar2(30) |  |  | No | Degrees of the course |
| Date\_get | Date |  |  | No | Identified the date |
| Student\_id | Int(11) |  | Yes | No | Students owns the certificate |

sessions   
Here we get the sessions required for the case the system store all its data and sessions

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| session | | | | | |
| Field | Type | Primary Key | Secondary Key | Null | Comments |
| session \_ID | Int (11) | Yes |  | No | Identifier Key |
| Hall | Varchar2(30) |  |  | No | Get the hall |
| Ticket\_cost | int(11) |  |  | No | Cost of session |
| Time | Date |  |  | No | Identified the date |
| Student\_id | Int(11) |  | Yes | No | Students owns the certificate |
| Teacher\_id | Int(11) |  | Yes | No | Teacher teaches the session |
| Course\_id | Int(11) |  | Yes | No | Course taught in the session |

1. Medical sub-system

Medicine   
Here we get the Medicine required for the case the doctor is up to and check whether it has side effects or not , depending on the case we assign the medicine if available and this medicine goes to the assigned medicine table for that particular patient

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Medicine | | | | | |
| Field | Type | Primary Key | Secondary Key | Null | Comments |
| Medicine\_ID | Int (11) | Yes |  | No | Identifier Key |
| Medicine\_Name | Varchar2(30) |  |  | No | Name of this Medicine |
| Side\_Effects | Varchar2(30) |  |  | No | Side effects of this Medicine |
| Expire\_Date | Int(8) |  |  | No | Expiration date of this Medicine |
| Availability | Text |  |  | No | Whether it’s available or not |

Assigned Medicine   
here we receive this assigned medicine by a prescription from the doctor and add it to that particular patient with the unique id he/she gets and this assigned medicine id is written in that patients medical history for further recommendations

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Assigned Medicine | | | | | |
| Field | Type | Primary Key | Secondary Key | Null | Comments |
| MAssign\_ID | Int (11) | Yes |  | No | Identifier Key |
| Patient\_ID | Int (11) |  | Yes | No | Foreign Key for Patient Table |
| Prescription | Varchar2(30) |  |  | No | Prescription of this Medicine to be included in the history of the patient |
| Date | Int(8) |  |  | No | Date of Assigning this medicine |
| Medicine\_id | Int(11) |  | Yes | No | Identified the medicine to be assigned |
| Doctor\_id | Int(11) |  | Yes | No | Identifies the doctor gave you the medicine |

Patient

This is one of the most important tables as without it there would be no medical services given to patients as there will be no patients , we reserve the patient’s name ,age for identification and assign each patient and id for further services and to ease the process as for the medical history is really important to recommend the best way to treat this patient whether it’s by a surgery , medicine … etc

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Patient | | | | | |
| Field | Type | Primary Key | Secondary Key | Null | Comments |
| Patient\_ID | Int (11) | Yes |  | No | Identifier Key |
| Patient\_Name | text |  |  | No | Patient Name |
| Birth\_date | date |  |  | No | Patient Age |

Assign Room

Here we check the assigned rooms for the patient with 2 keys one from the room and the other from the patient to make sure this room is taken or available and for history of this room as occupied

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Assign Room | | | | | |
| Field | Type | Primary Key | Secondary Key | Null | Comments |
| Assign\_ID | Int (11) | Yes |  | No | Identifier Key |
| Patient\_ID | Int (11) |  | Yes | No | Patient Foreign ID |
| Date | Int (8) |  |  | No | Data of Room Assignment |
| Room\_id | Int(11) |  | yes | no | Identifies the room to be assign |

Rooms  
In this table we check if there are any available rooms , how many beds in a room and the location of each room in the premises

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rooms | | | | | |
| Field | Type | Primary Key | Secondary Key | Null | Comments |
| Room\_ID | Int (11) | Yes |  | No | Identifier Key |
| Location | Text |  |  | No | Location of the room in the Hospital or caring center |
| Availability | Varchar2(30) |  |  | No | To Check if there are any available rooms |
| No\_of\_Beds | Int(5) |  |  | No | To determine how many beds in a room for variations |

Doctors  
in this table we get the required for the identification of each doctor for the medical history of each patient along with the financial matter of the financial department and the services given within a period of time

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Doctors | | | | | |
| Field | Type | Primary Key | Secondary Key | Null | Comments |
| Doctor\_ID | Int (11) | Yes |  | No | Identifier Key |
| Doctor\_Name | text |  |  | No | Name of Doctor for Medical History |
| Birth\_date | Int(8) |  |  | No | To Check for validation of employment |
| Salary | Int(5) |  |  | No | For Financial Purposes |

1. Fund raising sub-system

Donor  
Here we get the Donor s required for the case the system store all its data and Donors

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| teacher | | | | | |
| Field | Type | Primary Key | Secondary Key | Null | Comments |
| Donor r\_ID | Int (11) | Yes |  | No | Identifier Key |
| Donor\_Name | Varchar2(30) |  |  | No | Get the name |
| address | Varchar2(30) |  |  | No | Here we get the address |
| Birth\_date | Date |  |  | No | Identified the birth date |

volunteer  
Here we get the volunteers required for the case the system store all its data and volunteers

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| teacher | | | | | |
| Field | Type | Primary Key | Secondary Key | Null | Comments |
| volunteers \_ID | Int (11) | Yes |  | No | Identifier Key |
| volunteers \_Name | Varchar2(30) |  |  | No | Get the name |
| Work\_type | Varchar2(30) |  |  | No | Type of work done |
| Years\_experience | Int(11) |  |  | No | Year of experience in work |
| Manager\_id | Int(11) |  | Yes | No | Specifies the manger |

Manager  
Here we get the Managers required for the case the system store all its data and Managers

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| teacher | | | | | |
| Field | Type | Primary Key | Secondary Key | Null | Comments |
| Manager\_ID | Int (11) | Yes |  | No | Identifier Key |
| Manager\_Name | Varchar2(30) |  |  | No | Get the name |
| address | Varchar2(30) |  |  | No | Here we get the address |
| Birth\_date | Date |  |  | No | Identified the birth date |

Donation   
Here we get the Donations required for the case the system store all its data and Donations

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| teacher | | | | | |
| Field | Type | Primary Key | Secondary Key | Null | Comments |
| Donation\_ID | Int (11) | Yes |  | No | Identifier Key |
| Pay\_type | Varchar2(30) |  |  | No | How to be paid |
| Money | Int(11) |  |  | No | Amount of money |
| Volunteer\_id | Int(11) |  | Yes | No | Specifies the volunteer |
| donor\_id | Int(11) |  | Yes | No | Specifies the donor |

1. SQL codes
2. Learning sub-system

create table student (

student\_id number(11) primary key,

student\_name varchar2(30) ,

birth\_date date not null ,

academic\_degree number(11) );

create table teacher (

teacher\_id number(11) primary key,

teacher\_name varchar2(30) ,

birth\_date date not null ,

job varchar2(30) );

create table certificate (

certificate\_id number(11) primary key,

certificate\_name varchar2(30) ,

date\_get date not null,

degree varchar2(30) ,

student\_id number(11) ,

FOREIGN KEY (student\_id) references student(student\_id)

);

create table course (

course\_id number(11) primary key,

course\_name varchar2(30) ,

course\_degree number(11) ,

course\_hours number(11) );

create table prerequest (

prerequest\_id number(11) primary key,

course\_id number(11) ,

prequest\_id number(11) ,

FOREIGN KEY (course\_id) references course(course\_id) ,

FOREIGN KEY (prequest\_id) references course(course\_id)

);

create table sessions (

certificate\_id number(11) primary key,

cost number(11) ,

date\_get date not null,

hall varchar2(30) ,

course\_id number(11) ,

student\_id number(11) ,

teacher\_id number(11) ,

FOREIGN KEY (student\_id) references student(student\_id) ,

FOREIGN KEY (course\_id) references course(course\_id) ,

FOREIGN KEY (teacher\_id) references teacher(teacher\_id)

);

1. Medical sub-system

create table medicine (

medicine\_id number(11) not null PRIMARY KEY , medicine\_name varchar2(30) not null ,

side\_effects varchar2(30) ,

expire\_date date not null,

availability number(11) not null);

create table room(

room\_id number(11) not null PRIMARY KEY,

location varchar2(30) not null,

availability number(11) not null, no\_of\_bed number(11) not null);

create table doctors(

doctor\_id number(11) not null PRIMARY KEY, doctor\_name varchar2 (30) not null,

birth\_date date not null,

salary number(11) not null);

create table patients(

patient\_id number(11) not null PRIMARY KEY, patient\_name varchar2(30) not null,

birth\_date date not null);

create table assign\_room(

assign\_id number(11) PRIMARY KEY,

patient\_id number(11),

room\_id number(11),

assign\_date date ,

FOREIGN KEY (patient\_id) REFERENCES patients(patient\_id) ,

FOREIGN KEY (room\_id) REFERENCES room(room\_id)

);

create table assign\_medicine(

assign\_id number(11) PRIMARY KEY,

patient\_id number(11),

medicine\_id number(11),

doctor\_id number(11),

assign\_date date ,

prescription varchar2(30),

FOREIGN KEY (patient\_id) REFERENCES patients(patient\_id) ,

FOREIGN KEY (medicine\_id) REFERENCES medicine(medicine\_id) ,

FOREIGN KEY (doctor\_id) REFERENCES doctors(doctor\_id)

);

3- fund raisig sub-system

create table Donor (

Donor\_ID number(11) primary key,

address varchar2(30) not null ,

birth\_date date

);

create table Manager (

Manager\_ID number(11) primary key,

Manager\_name varchar2(30) not null,

address varchar2(30) ,

Salary number(11)

);

create table volunteer (

volunteer\_ID number(11) primary key,

volunteer\_name varchar2(30) not null,

year\_experience varchar2(30) ,

manager\_id number(11),

FOREIGN KEY (manager\_id) REFERENCES manager(manager\_id)

);

create table Donation (

donation\_ID number(11) primary key,

money number(11) not null ,

pay\_type varchar2(30) ,

Donor\_id number(11) ,

volunteer\_id number(11) ,

FOREIGN KEY (Donor\_id) REFERENCES Donor(Donor\_id),

FOREIGN KEY (volunteer\_id) REFERENCES Volunteer(volunteer\_id)

);

Screen shots of the gui with java

