Blood Bank Donation System
Database

Group Members: Raul Cordero (leader), Purva Chandel, Divina Gorospe
Contents

Modified Project Summary (#4) ........................................................................................................................................... 2

Tables............................................................................................................................................................................. 3

SQL Statements and Screenshots (#1-2) .......................................................................................................................... 5

CSV Files (#3)..................................................................................................................................................................... 12

Java File (#5)..................................................................................................................................................................... 13

Group Contribution (#6) .................................................................................................................................................... 13
Modified Project Summary (#4)

Our project is based on a Blood Bank Donation System Database. In our project we created multiple tables with multiple attributes each. Our purpose was to be able to have a database that stored information about the Donor, the Recipient, Medical personnel, how much and when the blood was donated, and whenever there was a transaction made between the donor, the medical personnel, and the recipient. All tables are linked together using foreign keys to enforce relationships. We created three different views: PatientSeen, BloodStock, and a PatientList. This adds security to the individual tables by not showing irrelevant data to unauthorized viewers. The views used joins to combine columns from different tables. We have utilized the import functionality in MyPHP to add multiple rows of information in all of our tables. This process made it so much more efficient to populate the tables. Our java code is able to access our database, but will not be able to show the views.

Our Donor and Recipient tables are extremely similar to each other. Both of these tables primary key is the ID attribute. They both include the person's first name, last name, address, email, phone number, date of birth, and blood type. The MedicalPersonnel table’s primary key is also the ID attribute. This table includes the same attributes as the Donor and Recipient table with the exception of not including their blood type. Our BloodDonation tables primary key is the bloodID attribute with the foreign key being donorID which is linked to the Donor table. This table also includes the date when the blood was donated and the quantity or amount donated. Lastly, our BloodTransaction table has one primary key and three foreign keys. The primary key is the transacID and the three foreign keys are: emplID which is linked to the MedicalPersonnel Table, the recipientID which is linked to the Recipient table, and the bloodID which is linked to the BloodDonation table.

The first view that we created is the PatientSeen view. This view is for medical personnel who would like to see which patients they’ve seen. This view includes the medical personnel who either administered donated blood or took blood, the donor's or recipient's name, and the date it happened. The second view we created was the BloodStock view. This view is seen by the medical personnel. This is just a simple view that shows all the blood types we have and the quantity of each we have in stock. The final and third view we created is the PatientList view. This view was made to show certain information about all the patients that came through the blood bank whether they were donating or receiving blood. The view will include the patient's full name, blood type, age, address, email, and phone number.

The java code that we made will be able to access our database and have access to all five of our tables. This program will start at the main menu and will first ask the user which table he/she wishes to access or if he/she wishes to terminate the program. Once the user selects the table of his/her choosing the program will give him/her four options. The four options are: If they wish to add something to the table they are currently in, view something specifically based on an ID in the table they are currently in, if they wish to print out the whole table in which they are currently in, or if they wish to go back to the main menu. This program will run forever until the user is back in the main menu and selects to terminate the program.
# Tables

<table>
<thead>
<tr>
<th>Table Name</th>
<th>Attributes</th>
<th>Constraint</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Donor</td>
<td>donorID</td>
<td>pk</td>
<td>int(11)</td>
<td>auto_increment</td>
</tr>
<tr>
<td></td>
<td>firstName</td>
<td></td>
<td>varchar (50)</td>
<td>not null</td>
</tr>
<tr>
<td></td>
<td>lastName</td>
<td></td>
<td>varchar (50)</td>
<td>not null</td>
</tr>
<tr>
<td></td>
<td>address</td>
<td></td>
<td>varchar (60)</td>
<td>not null</td>
</tr>
<tr>
<td></td>
<td>email</td>
<td></td>
<td>varchar (100)</td>
<td>not null</td>
</tr>
<tr>
<td></td>
<td>phone</td>
<td></td>
<td>varchar (20)</td>
<td>not null</td>
</tr>
<tr>
<td></td>
<td>dob</td>
<td></td>
<td>date</td>
<td>not null</td>
</tr>
<tr>
<td></td>
<td>bloodType</td>
<td></td>
<td>varchar (3)</td>
<td>not null</td>
</tr>
<tr>
<td>Recipient</td>
<td>recipientID</td>
<td>pk</td>
<td>int(11)</td>
<td>auto_increment</td>
</tr>
<tr>
<td></td>
<td>firstName</td>
<td></td>
<td>varchar (50)</td>
<td>not null</td>
</tr>
<tr>
<td></td>
<td>lastName</td>
<td></td>
<td>varchar (50)</td>
<td>not null</td>
</tr>
<tr>
<td></td>
<td>address</td>
<td></td>
<td>varchar (60)</td>
<td>not null</td>
</tr>
<tr>
<td></td>
<td>email</td>
<td></td>
<td>varchar (100)</td>
<td>not null</td>
</tr>
<tr>
<td></td>
<td>phone</td>
<td></td>
<td>varchar (20)</td>
<td>not null</td>
</tr>
<tr>
<td></td>
<td>dob</td>
<td></td>
<td>date</td>
<td>not null</td>
</tr>
<tr>
<td></td>
<td>bloodType</td>
<td></td>
<td>varchar (3)</td>
<td>not null</td>
</tr>
<tr>
<td>BloodDonation</td>
<td>bloodID</td>
<td>PK</td>
<td>int(11)</td>
<td>auto_increment</td>
</tr>
<tr>
<td></td>
<td>donorID</td>
<td></td>
<td>int (11)</td>
<td>not null</td>
</tr>
<tr>
<td></td>
<td>dateDonated</td>
<td></td>
<td>datetime</td>
<td>not null</td>
</tr>
<tr>
<td></td>
<td>quantity</td>
<td></td>
<td>int</td>
<td>not null</td>
</tr>
<tr>
<td>MedicalPersonnel</td>
<td>empID</td>
<td>pk</td>
<td>int(11)</td>
<td>auto_increment</td>
</tr>
<tr>
<td></td>
<td>firstName</td>
<td></td>
<td>varchar (50)</td>
<td>not null</td>
</tr>
<tr>
<td></td>
<td>lastName</td>
<td></td>
<td>varchar (50)</td>
<td>not null</td>
</tr>
<tr>
<td></td>
<td>address</td>
<td></td>
<td>varchar (60)</td>
<td>not null</td>
</tr>
<tr>
<td>Field</td>
<td>Type</td>
<td>Nullable</td>
<td>Unique</td>
<td></td>
</tr>
<tr>
<td>---------------</td>
<td>------------</td>
<td>----------</td>
<td>--------</td>
<td></td>
</tr>
<tr>
<td>email</td>
<td>varchar(100)</td>
<td>not null</td>
<td></td>
<td></td>
</tr>
<tr>
<td>phone</td>
<td>varchar(20)</td>
<td>not null</td>
<td></td>
<td></td>
</tr>
<tr>
<td>dob</td>
<td>date</td>
<td>not null</td>
<td></td>
<td></td>
</tr>
<tr>
<td>transactID</td>
<td>int(11)</td>
<td>auto_increment</td>
<td>not null</td>
<td></td>
</tr>
<tr>
<td>empID</td>
<td>int(11)</td>
<td>not null</td>
<td></td>
<td></td>
</tr>
<tr>
<td>dateOut</td>
<td>datetime</td>
<td>not null</td>
<td></td>
<td></td>
</tr>
<tr>
<td>quantity</td>
<td>int</td>
<td>not null</td>
<td></td>
<td></td>
</tr>
<tr>
<td>recipientID</td>
<td>int(11)</td>
<td>not null</td>
<td></td>
<td></td>
</tr>
<tr>
<td>bloodType</td>
<td>varchar(3)</td>
<td>not null</td>
<td></td>
<td></td>
</tr>
<tr>
<td>bloodID</td>
<td>int(11)</td>
<td>not null</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
SQL Statements

CREATE TABLE Donor(
donorID INT AUTO_INCREMENT NOT NULL,
firstName VARCHAR(50) NOT NULL,
lastname VARCHAR(50) not null,
address VARCHAR(60) not null,
email VARCHAR(100) not null,
phone VARCHAR(20) not null,
dob DATE not null,
bloodType varchar(3) NOT NULL,
PRIMARY KEY (donorID));
CREATE TABLE Recipient(
  recipientID INT AUTO_INCREMENT NOT NULL,
  firstName VARCHAR( 50 ) NOT NULL,
  lastname VARCHAR( 50 ) not null,
  address VARCHAR( 60 ) not null,
  email VARCHAR( 100 ) not null,
  phone VARCHAR( 20 ) not null,
  dob DATE not null,
  bloodType varchar (3)NOT NULL,
  PRIMARY KEY ( recipientID )
);
CREATE TABLE MedicalPersonnel(
empID INT AUTO_INCREMENT NOT NULL,
firstName VARCHAR(50) NOT NULL,
lastname VARCHAR(50) NOT NULL,
address VARCHAR(60) NOT NULL,
email VARCHAR(100) NOT NULL,
phone VARCHAR(20) NOT NULL,
dob DATE NOT NULL,
PRIMARY KEY (empID)
);
CREATE TABLE BloodDonation(
  bloodID INT(11) AUTO_INCREMENT,
  donorID INT(11) NOT NULL,
  dateDonated DATETIME NOT NULL,
  quantity INT NOT NULL,
  PRIMARY KEY (bloodID),
  FOREIGN KEY (donorID) REFERENCES Donor(donorID)
);
CREATE TABLE BloodTransaction(
    transactID INT(11) AUTO_INCREMENT,
    empID INT(11) NOT NULL,
    dateOut DATETIME NOT NULL,
    quantity INT NOT NULL,
    recipientID INT(11) NOT NULL,
    bloodID INT(11) NOT NULL,
    PRIMARY KEY (transactID),
    FOREIGN KEY (empID) REFERENCES MedicalPersonnel(empID),
    FOREIGN KEY (recipientID) REFERENCES Recipient(recipientID),
    FOREIGN KEY (bloodID) REFERENCES BloodDonation(bloodID)
);
create view PatientSeen as select 
concat_ws(' ', m.firstName, m.lastName) as 'Medical Personnel',
concat_ws(' ', r.firstname, r.lastName) as 'Patient Name',
dateOut as 'Date Seen'
From MedicalPersonnel m, BloodTransaction b, Recipient r
where m.empID = b.empID AND r.recipientID = b.recipientID
order by m.lastName ASC;

<table>
<thead>
<tr>
<th>Medical Personnel</th>
<th>Patient Name</th>
<th>Date Seen</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ashley Asher</td>
<td>Aspen Holland</td>
<td>2016-03-03</td>
</tr>
<tr>
<td>Eli Cole</td>
<td>Ashley Sky</td>
<td>2016-03-06</td>
</tr>
<tr>
<td>Laura Grove</td>
<td>Hal Kim</td>
<td>2016-03-17</td>
</tr>
<tr>
<td>Tim Hane</td>
<td>Moms Hane</td>
<td>2016-03-08</td>
</tr>
<tr>
<td>Roy Hot</td>
<td>Lina Krane</td>
<td>2016-03-18</td>
</tr>
<tr>
<td>Alice Holland</td>
<td>Kate Free</td>
<td>2016-03-07</td>
</tr>
<tr>
<td>Reno Jal</td>
<td>Tim Hane</td>
<td>2016-03-13</td>
</tr>
<tr>
<td>Sam George</td>
<td>Eli Mac</td>
<td>2016-03-09</td>
</tr>
<tr>
<td>Holly Mac</td>
<td>Eliza Car</td>
<td>2016-03-14</td>
</tr>
<tr>
<td>Deanna Mano</td>
<td>Holly Ashor</td>
<td>2016-03-16</td>
</tr>
<tr>
<td>Deanna Mano</td>
<td>Eliza Car</td>
<td>2016-03-19</td>
</tr>
<tr>
<td>Melissa Smith</td>
<td>Melissa Stine</td>
<td>2016-03-01</td>
</tr>
<tr>
<td>Melissa Smith</td>
<td>Chris Cole</td>
<td>2016-03-02</td>
</tr>
<tr>
<td>Melissa Smith</td>
<td>Deanna Riza</td>
<td>2016-03-05</td>
</tr>
<tr>
<td>Melissa Smith</td>
<td>Ron Ali</td>
<td>2016-03-10</td>
</tr>
<tr>
<td>Crean Snow</td>
<td>Eli Cole</td>
<td>2016-03-11</td>
</tr>
<tr>
<td>Maya Stain</td>
<td>Alice Holland</td>
<td>2016-03-12</td>
</tr>
</tbody>
</table>

create view BloodStock as select 
Donor.bloodType as 'Blood Type',
sum(BloodDonation.quantity) as 'In Stock'
from BloodDonation
join Donor on BloodDonation.donorID = Donor.donorID
where BloodDonation.bloodID not in (select bloodID from BloodTransaction)
group by bloodType

<table>
<thead>
<tr>
<th>Blood Type</th>
<th>In Stock</th>
</tr>
</thead>
<tbody>
<tr>
<td>A+</td>
<td>4</td>
</tr>
<tr>
<td>A</td>
<td>2</td>
</tr>
<tr>
<td>AB+</td>
<td>2</td>
</tr>
<tr>
<td>AB-</td>
<td>1</td>
</tr>
<tr>
<td>B</td>
<td>2</td>
</tr>
<tr>
<td>O+</td>
<td>3</td>
</tr>
<tr>
<td>O-</td>
<td>4</td>
</tr>
</tbody>
</table>
create view PatientList as select
concat_ws(', ', lastName, firstName) as Name,
bloodType as 'Blood Type',
date_format(from_days(to_days(now())-to_days(dob)), '%y')+0 as Age,
address as Address,
email as Email,
phone as Phone
from Donor union
select
concat_ws(', ', lastName, firstName) as 'PatientName',
bloodType as 'Blood Type',
date_format(from_days(to_days(now())-to_days(dob)), '%y')+0 as Age,
address as Address,
email as Email,
phone as Phone
from Recipient
Order by Name
## CSV Files (#3)

<table>
<thead>
<tr>
<th>Table</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>BloodDonation</td>
<td><img src="" alt="bloodDonation.csv" /></td>
</tr>
<tr>
<td></td>
<td><img src="" alt="additionalBloodDonation.csv" /></td>
</tr>
<tr>
<td>BloodTransaction</td>
<td><img src="" alt="bloodTransaction.csv" /></td>
</tr>
<tr>
<td>Donor</td>
<td><img src="" alt="donor.csv" /></td>
</tr>
<tr>
<td>MedicalPersonnel</td>
<td><img src="" alt="medicalPersonnel.csv" /></td>
</tr>
<tr>
<td>Recipient</td>
<td><img src="" alt="recipient.csv" /></td>
</tr>
</tbody>
</table>
Java File (#5)

projectDb.java - included in zip file

Group Contribution (#6)

Individual contributions:

Java- Created by Raul, revised by Divina and Purva. We all made the changes to make it user friendly by adding additional code to format the outputs.

Database tables - Created by Divina, rows populated by Purva using csv files. Raul ensured that tables were correctly implemented and accessible using Java.

CSV Files: Created by Purva.

Final Document - Created by Divina using google docs, revised by Raul and Purva. Everyone had their inputs on different sections.