

Blood Bank Management System

Submitted by
VARUNKARTHIK(RA2011003011159)
SAI HARISH (RA2011003011185)
CHANDRA SEKHAR (RA2011003011180)

Under the Guidance of

M. Karthikeyan

(Assistant Professor)
Department of Data Base Management System

In partial satisfaction of the requirements for the degree of

BACHELOR OF TECHNOLOGY
in
COMPUTER SCIENCE ENGINEERING



SCHOOL OF COMPUTING
COLLEGE OF ENGINEERING AND TECHNOLOGY
SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR - 603203
APRIL 2023



SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
Approved by the University of the South, 1988

SRM INSTITUTE OF SCIENCE & TECHNOLOGY
COLLEGE OF ENGINEERING & TECHNOLOGY
S.R.M. NAGAR, KATTANKULATHUR – 603203

BONAFIDE CERTIFICATE

Certified that this project report “**BLOOD BANK MANAGEMENT**” is the bonafide work of “**Varun Karthik(RA2011003011159), Chandra sekhar (RA2011003011180), Sai harish(RA2011003011185)**” of III Year/VI Sem B.tech(CSE) who carried out the mini projectwork under my supervision for the course 18CSC303J- Database Management systems in SRM Institute of Science and Technology during the academic year 2022-2023(Even sem).


SIGNATURE

Karthikeyan M
Assistant professor
Department of computing technologies
SRM institute of science and technology




SIGNATURE

DR.M. Pushpalatha
HEAD OF THE DEPARTMENT
Professor & Head
Department of computing technologies

ABSTRACT

With an increase in the population there is an increase in the need of blood. The growing population of the world results in a lot of potential blood donors. But in spite of this not more than 10% of the total world population participates in blood donation. With the growing population and the advancement in medical science the demand for blood has also increased.

Due to the lack of communication between the blood donors and the blood recipients, most of the patients in need of blood do not get blood on time and hence lose their lives. There is a dire need of synchronization between the blood donors and hospitals and the blood banks. This improper management of blood leads to wastage of the available blood inventory.

TABLE OF CONTENTS

Chapter No.	Title	Page No.
	<u>ABSTRACT</u>	3
	TABLE OF CONTENTS	4
	LIST OF FIGURES	5
	LIST OF TABLES	6
	<u>ABBREVIATIONS</u>	7
1	INTRODUCTION 1	
1.1	Introduction	9
1.2	Problem statement	10
1.3	Objectives	11
1.4	Scope and applications	
1.5	Software Requirements Specification	
2	LITERATURE SURVEY 14	
2.1	Existing system	14
2.2	Comparison of Existing vs Proposed system	14
3	SYSTEM ARCHITECTURE AND DESIGN 15	
3.1	Architecture Diagram	15
3.1.1	Front end (UI)design	16
3.1.2	Back end (Database) design	17
3.2	ER Diagram and Use case Diagram	18
4	Modules and Functionalities 20	
4.1	Connectivity used for database access	21
5	CODING AND TESTING RESULTS AND DISCUSSIONS 28	
6	CONCLUSION AND FUTURE ENHANCEMENT 29	
	REFERENCES	30

LIST OF FIGURES

Figure No.	Figure Name	Page No.
5.1	DATA BASE SCHEMA	13
5.2	ER DIAGRAM	13
5.3	THE DIFFERENT BLOOD TYPES	15
5.4	THE PATIENTS NAME WITH NAME	20

LIST OF TABLES

Table No.	Table Name	Page No.
6.1	CREATION OF BLOOD TABLE	12
6.2	CREATION OF BLOODBANK TABLE	13
6.3	CREATION OF PATIENT TABLE	14

ABBREVIATIONS

CRM	Customer Relationship Management
CSS	Cascading Style Sheet
CV	Computer Vision
DB	Data Base
CMS	Content Management System
SQL	Structured Query Language

CHAPTER1

INTRODUCTION

- A blood bank management system is a software application designed to manage the various aspects of a blood bank, including blood donation, blood screening, blood storage, and blood distribution. It is used by blood banks to maintain records of blood donors, blood products, and their inventory levels
- The main purpose of a blood bank management system is to ensure the safe and efficient use of blood products. It helps to track the inventory of blood products, maintain the quality of blood, and ensure the timely delivery of blood products to patients in need.
- Blood distribution: Enables blood bank staff to quickly and efficiently process requests for blood products and distribute them to hospitals and other healthcare facilities
- In a mini project, the focus can be on implementing some of these features and creating a functional prototype of the blood bank management system.

PROBLEM STATEMENT

- Due to the lack of communication between the blood donors and the blood recipients, most of the patients in need of blood do not get blood on time and hence lose their lives.
- There is a dire need of synchronization between the blood donors and hospitals and the blood banks.
- This improper management of blood leads to wastage of the available blood inventory.

OBJECTIVE:

Blood Bank Management System (BBMS) is a browser based system that is designed to store, process, retrieve and analyze information concerned with the administrative and inventory management within a blood bank. This project aims at maintaining all the information pertaining to blood donors, different blood groups available in each blood bank and help them manage in a better way. Aim is to provide transparency in this field, make the process of obtaining blood from a blood bank hassle free and corruption free and make the system of blood bank management effective. Our client is not interested in blood stocking instead we are stocking blood donors information. The donors who are interested in donating blood has to register in the database. There is no storage of blood so no complications in the project. The software is fully integrated with CRM (customer relationship management) as well as CMS (content management system) solution. It is developed in a manner that is easily manageable, time saving and relieving one from manual works. The requirement of the blood has to be requested and we supply the information of the donor. The donors can update their status whether they are available or not.

Scope and applications

- Donor management: Implement a donor management module that enables blood bank staff to manage donor information, such as their eligibility for blood donation, donation history, and contact details.
- Communication: Implement a communication module that enables blood bank staff to send reminders and notifications to donors about upcoming appointments, events, and blood donation drives
- Blood donation centers: A blood bank management system can be used by blood donation centers to manage their donor database, maintain donor communication, and ensure the availability of donors for blood donation.

Software Requirements Specification

This Software Requirements Specification document outlines the functional and non-functional requirements of the blood bank management system mini project.

The system will be designed to ensure the safe and efficient use of blood products, maintain the quality of blood, and ensure the timely delivery of blood products to patients in need.

The system will be user-friendly, reliable, secure, and scalable to meet the needs of blood banks and healthcare facilities

2 LITERATURE SURVEY

Existing system

In the existing system, the blood bank management system that only collects data but not blood is done manually. All the information related to donors, blood groups, and the inventory of blood units is recorded on paper. This manual process is time-consuming, and there is a high risk of errors in recording data. The process of searching for donors with specific blood groups is also cumbersome and can take a long time.

Proposed System

The proposed system is a computerized blood bank management system that only collects data but not blood. This system is designed to automate the manual processes of blood bank management. The proposed system will be a user-friendly interface that will enable donors to register themselves online, and their information will be stored in the database. The system will also store information about the inventory of blood units, blood group-wise donor list, and donor history.

Comparison of Existing vs Proposed system

The proposed system is an improvement over the existing system in several

ways. The proposed system is computerized, which means that the data can be stored and accessed more efficiently.

The proposed system will also reduce the time and effort required to search for donors with specific blood groups.

The proposed system will also reduce the risk of errors in recording data as it will be a digital system. Additionally, the proposed system will provide a more secure and centralized platform to store donor information, which will help to ensure the confidentiality and privacy of the donors.

3 SYSTEM ARCHITECTURE AND DESIGN

3.1 Architecture Diagram 3.11 Front end(UI)design

The image displays two screenshots of the 'Live SQL' web application interface.

Top Screenshot: SQL Worksheet

The interface shows a sidebar with navigation options: Home, SQL Worksheet (selected), My Session, Schema, Quick SQL, My Scripts, My Tutorials, and Code Library. The main area is titled 'SQL Worksheet' and contains a text editor with the following SQL code:

```
1 CREATE TABLE DONOR (
2   FIRST_NAME VARCHAR(256) NOT NULL,
3   MIDDLE_NAME VARCHAR(256) NOT NULL,
4   LAST_NAME VARCHAR(256) NOT NULL,
5   DOR DATE,
6   AGE VARCHAR(18),
7   WEIGHT NUMBER,
8   GENDER CHAR(1),
9   CITY VARCHAR(50),
10  PROFILE NUMBER
11 );
12
13 INSERT INTO DONOR VALUES ('TAGORE', 'PAVAN', 'KUNAR', TO_DATE('2002-12-17','YYYY-MM-DD'), '19 YEARS', 67, 'M', 'KIKATPALLY', 1234567891);
14 INSERT INTO DONOR VALUES ('A', 'RAH', 'REDDY', TO_DATE('2001-01-11','YYYY-MM-DD'), '20 YEARS', 71, 'M', 'KOPPALY', 2345678912);
15 INSERT INTO DONOR VALUES ('G', 'LOKESH', 'DON', TO_DATE('2002-04-08','YYYY-MM-DD'), '19 YEARS', 58, 'M', 'KARUPPAGHAT', 3456789123);
16 INSERT INTO DONOR VALUES ('A', 'BOHIDHA', 'ATLA', DATE '2003-04-26', '18 YEARS', 59, 'F', 'MIYAPUR', 5678912345);
```

Below the code editor, a message states: "One month, your SQL scripts could be an exciting object". The execution results show:

```
1 row(s) inserted.
1 row(s) inserted.
```

Bottom Screenshot: Schema View

The interface shows the 'Schema' view for a database named 'BLOOD'. The sidebar navigation is the same, but 'Schema' is selected. The main area shows the 'BLOOD' schema with tabs for Show All, Columns, Constraints, Related Constraints, Triggers, Indexes, and Code. The 'Columns' tab is active, displaying a table with the following data:

#	Column	Type	Length	Precision	Scale	Nullable	Semantics	Comment
1	CODE	NUMBER	22			No		
2	BLOOD_TYPE	VARCHAR2	4			No	Byte	
3	QUANTITY	NUMBER	22			No		

Below the columns table, there are sections for Constraints, Related Constraints, Triggers, and Indexes, all of which show "No constraints defined", "No related constraints defined", and "No triggers defined" respectively.

Live SQL

Schema: BLOODBANK

Show All Columns Constraints Related Constraints Triggers Indexes Code

Columns

#	Column	Type	Length	Precision	Scale	Nullable	Semantics	Comment
1	ORDERS	NUMBER	22			Yes		
2	BLOOD_TYPE	VARCHAR2	4			Yes	Byte	
3	QUANTITY	NUMBER	22			Yes		

Constraints

No constraints defined.

Related Constraints

No related constraints defined.

Triggers

No triggers defined.

Indexes

DESIGN:

The following Entities are used in our Database:

- 1). Donor
- 2). Blood.
- 3). Blood bank.
- 4). Patient.

Donor : Each donor Entity has a Name, Dob and Address associated with it.



One Donor can donate 1 kind of blood group. **Blood:**

Each blood Entity has associated with it a code ,blood type and quantity of blood.



n different blood groups will be sent to 1 blood bank.

Blood bank:

Every blood bag is sent to blood bank with a certain code Number and Total quantity.



n different Blood bags can be present in One blood bank.



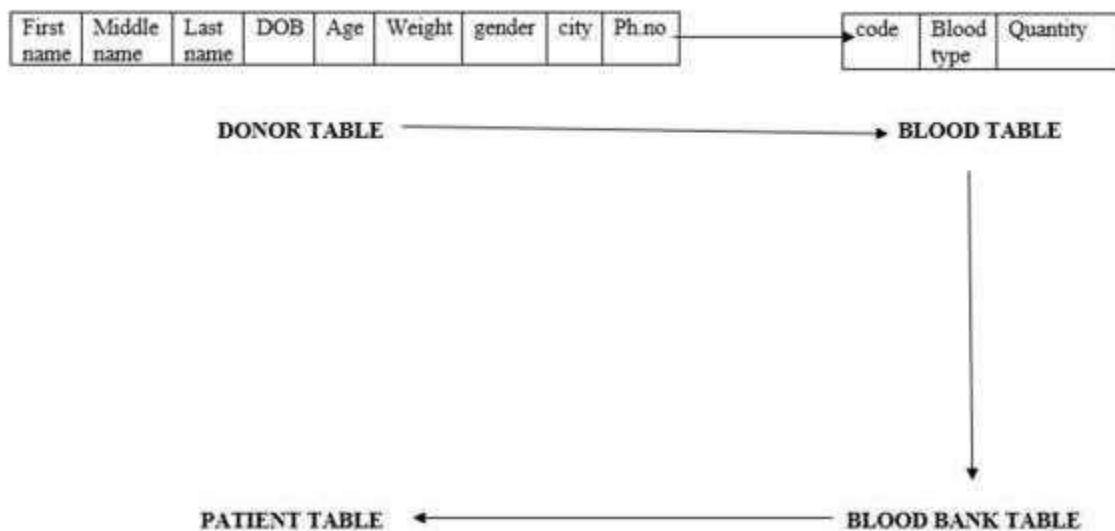
m blood banks can be associated with n patients.

Patient:

For every User entity, there exist Attributes Name, Address, id, Phone Number and quantity of blood required, where id also serves as the Primary Key for the entity.

- ☐ Every User can take blood from blood bank.

DATA BASE SCHEMA



This entity is used to associate a particular Book with a certain User. It is characterized by the Attributes –blood type, id, name, age. n Blood banks can be associated with One User’s Info.

- ☐ Id serves as Foreign Key between User Info and User.
- ☐ Blood type serves as Foreign Key between User Info and Product Info. ii).

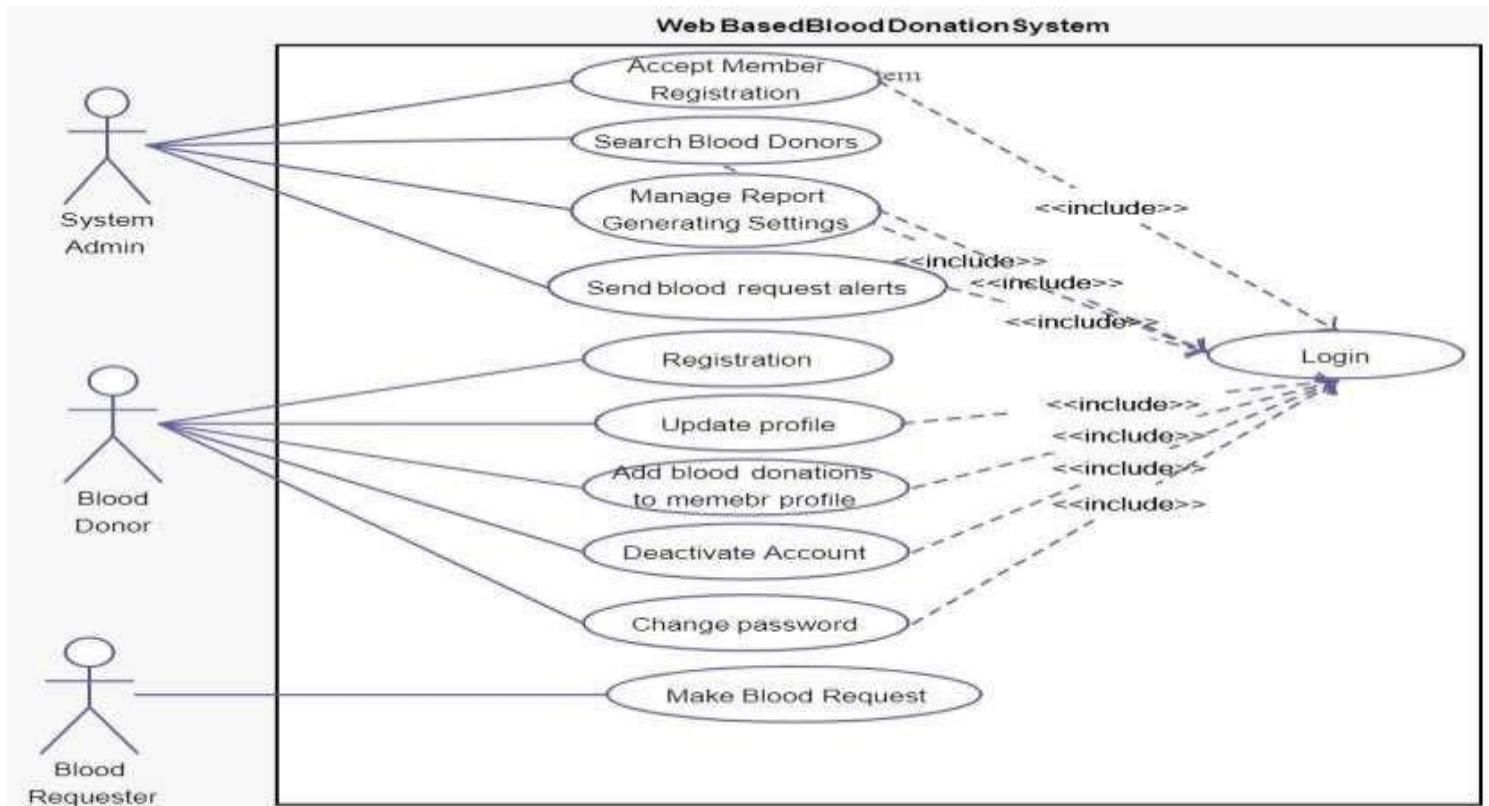
Product Info:

This entity is used to identify the type of blood needed. It comprises of the attributes code, blood type and quantity, code serves as the Primary Key for the entity. One Blood bag has only One Product Info.

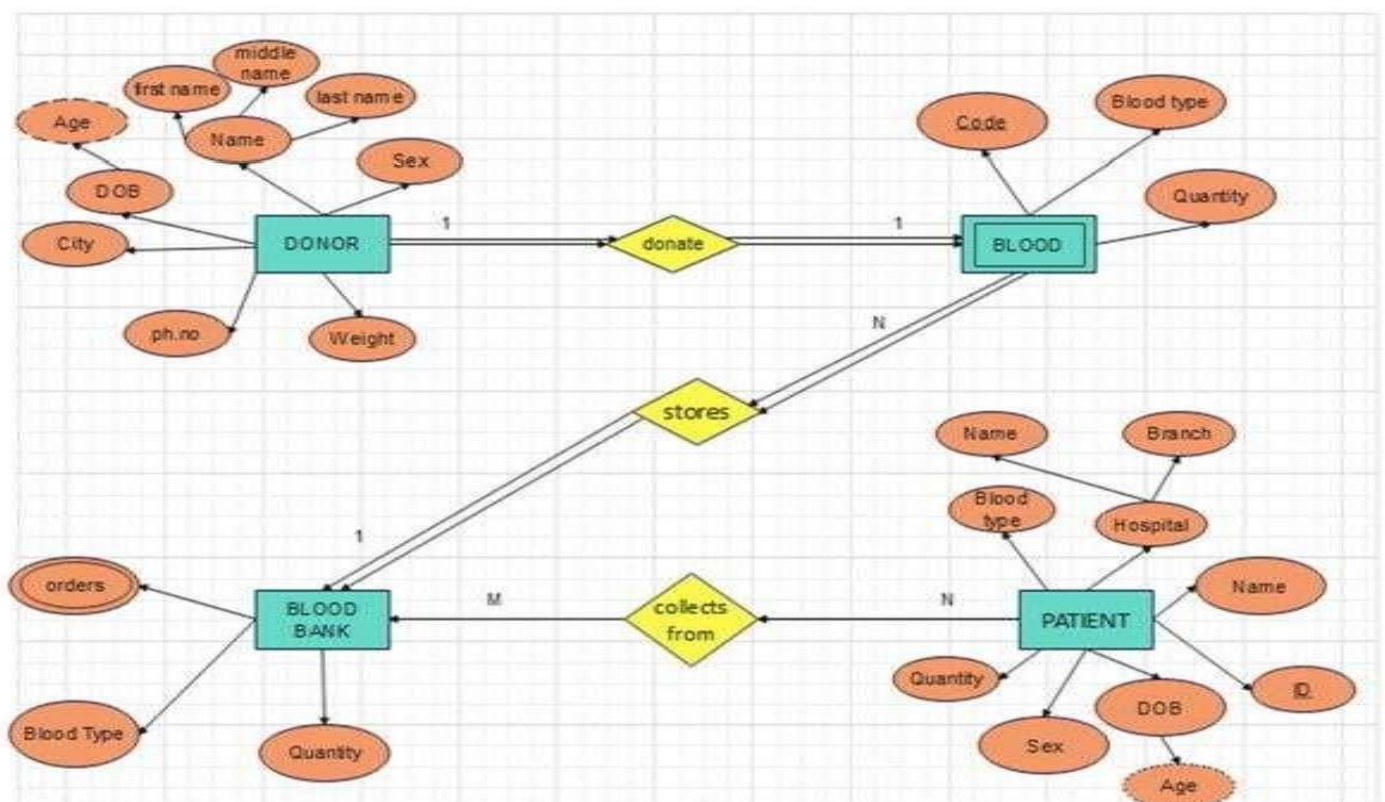
Overall, the blood bank entity has the following defined Relationships:

- ☐ n Blood bags can be present in One blood bank.
- ☐ Any number of patients can take blood from any number of blood banks.

3.2 ER Diagram and Use case Diagram



ER DIAGRAM



4. Modules and Functionalities

4.1 Donor Management Module

- Registration of donors
- Search donors by blood group, location, or name
- Update and delete donor information

4.1.1 User Management Module

- Admin login
- Add, update, and delete user accounts
- Manage user permissions

4.1.2 Backup and Recovery Module

- Regular backup of the database
- Restore data in case of data loss

4.2 Connectivity used for database access

JDBC (Java Database Connectivity): This is a Java-based connectivity option that provides a standard interface for accessing relational databases. It enables Java programs to interact with databases, and it is supported by most major database vendors.

DATABASE USED

WHY WE USED SQL DATABASE ?

Faster Query Processing

Large amount of data is retrieved quickly and efficiently. Operations like Insertion, deletion, manipulation of data is also done in almost no time.

No Coding Skills

For data retrieval, large number of lines of code is not required. All basic keywords such as SELECT, INSERT INTO, UPDATE, etc are used and also the syntactical rules are not complex in SQL, which makes it a userfriendly language.

Standardized Language

Due to documentation and long establishment over years, it provides a uniform platform worldwide to all its users.

Portable

It can be used in programs in PCs, server, laptops independent of any platform (Operating System, etc). Also, it can be embedded with other applications as per need/requirement/use.

Interactive Language

Easy to learn and understand, answers to complex queries can be received in seconds.

5. CODING AND TESTING

SQL CODE:

CREATION OF DONOR TABLE

```
CREATE TABLE DONOR ( FIRST_NAME VARCHAR(256) NOT NULL, MIDDLE_NAME VARCHAR(256) NOT NULL,
LAST_NAME VARCHAR(256) NOT NULL,DOB DATE,AGE NUMBER,WEIGHT NUMBER,GENDER CHAR(1),CITY
VARCHAR(50),MOBILE NUMBER);
```

```
INSERT INTO DONOR VALUES('TAGORE','PAVAN','KUMAR','17-DEC-
2002','19YEARS',67,'M','KUKATPALLY','1234567891');
```

```
INSERT INTO DONOR VALUES('A','RAM','REDDY','11-JAN-2001','20 YEARS' ,71,'M','KOMPALLY','2345678912');
```

```
INSERT INTO DONOR VALUES('G','LOKESH','DON','08-APR-2002','19
YEARS',58,'M','KHARMANGHAT','3456789123');
```

```
INSERT INTO DONOR VALUES('A','ROHITHA','ATLA','26-APR-2003','18 YEARS',59,'F','MIYAPUR','5678912345');
```

```
INSERT INTO DONOR VALUES('G','MANIDEEP','REDDY','14-FEB-2002','19
YEARS',71,'M','AMEERPET','6789123456');
```

```
INSERT INTO DONOR VALUES('G','RITEESH','DON','11-NOV-2002','19 YEARS',74,'M','MAREDPALLY','7891234567');
```

CREATION OF BLOOD TABLE

```
CREATE TABLE BLOOD (CODE NUMBER,BLOOD_TYPE VARCHAR2(4),QUANTITY NUMBER );
```

```
INSERT INTO BLOOD VALUES(1,'A+',350);
```

```
INSERT INTO BLOOD VALUES(2,'A-',245);
```

```
INSERT INTO BLOOD VALUES(3,'B+',370);
```

```
INSERT INTO BLOOD VALUES(4,'B-',310);
```

```
INSERT INTO BLOOD VALUES(5,'AB+',330);
```

```
INSERT INTO BLOOD VALUES(6,'AB-',215);
```

```
INSERT INTO BLOOD VALUES(7,'O+',250);
```

```
INSERT INTO BLOOD VALUES(8,'O-',285);
```

```
SELECT * FROM BLOOD;
```

CREATION OF BLOODBANK TABLE:

```
CREATE TABLE BLOODBANK(ORDERS NUMBER,BLOOD_TYPE VARCHAR2(4),QUANTITY NUMBER);
```

```
INSERT INTO BLOODBANK VALUES(3,'A+',13);
```

```
INSERT INTO BLOODBANK VALUES(4,'A-',8);
```

```
INSERT INTO BLOODBANK VALUES(5,'B+',9); INSERT
INTO BLOODBANK VALUES(1,'B-',5);
INSERT INTO BLOODBANK VALUES(2,'AB-',15);
INSERT INTO BLOODBANK VALUES(6,'AB+',3);
INSERT INTO BLOODBANK VALUES(4,'O-',11);
INSERT INTO BLOODBANK VALUES(7,'O+',8);
SELECT * FROM BLOODBANK;
```

CREATION OF PATIENT TABLE

```
CREATE TABLE PATIENT(ID NUMBER PRIMARY KEY,NAME VARCHAR2(40),GENDER CHAR(1),DOB DATE
,BLOOD_GROUP
VARCHAR2(4),QUANTITY NUMBER,HOSPITAL_NAME VARCHAR2(40),PLACE VARCHAR(20));
INSERT INTO PATIENT VALUES(1204,'RAGHU','M','07-OCT-2002','O+',2,'KRISHNA
HOSPITALS','DILSUKHNAGAR');
INSERT INTO PATIENT VALUES(1205,'SOUMYA','F','03-MAR-2001','A+',1,'RAINBOW HOSPITALS','CHANDA
NAGAR');
INSERT INTO PATIENT VALUES(1206,'SHYAM','M','01-SEP-2002','A-',2,'KRISHNA HOSPITALS','KOMPALLY');
INSERT INTO PATIENT VALUES(1207,'MEGHANA','F','05-MAR-2002','AB-',2,'KIMS HOSPITAL','PATANCHERU');
INSERT INTO PATIENT VALUES(1208,'SHIVA','M','07-FEB-2001','AB+',1,'KK HEALTH CARE','BHEL');
INSERT INTO PATIENT VALUES(1209,'RANGA RAO','M','01-NOV-2002','AB-',1,'KIMS HOSPITALS','KONDAPUR');
INSERT INTO PATIENT VALUES(1210,'SHIVANI','F','21-JAN-2003','B+',1,'GANDHI HOSPITALS','MIYAPUR');
INSERT INTO PATIENT VALUES(1211,'KIRAN','M','06-DEC-2002','B-',4,'SWAPNA HOSPITAL','DILSUKHNAGAR');
INSERT INTO PATIENT VALUES(1212,'ADITYA','M','14-FEB-2001','AB-',1,'ANKURA HOSPITAL','KPHB');
INSERT INTO PATIENT VALUES(1213,'CHARANI','F','11-JAN-2003','O-',2,'MIDWAY CENTRAL
HOSPITAL','LINGAMPALLY'); SELECT * FROM PATIENT;
```

SNAPSHOTS:

SQL Worksheet

CWS

Mail

Actions

Save

Run

```
1 CREATE TABLE DONOR ( FIRST_NAME VARCHAR(100) NOT NULL, MIDDLE_NAME VARCHAR(250) NOT NULL, LAST_NAME VARCHAR(250) NOT NULL,DOB DATE,ADE NUMBER,HEIGHT NUMBER,GENDER CHAR(1)
2 CHECK2 INTO DONOR VALUES('JAGGER','RAYAN','KUMAR','27-DEC-2002','18YEARS','M','HUGHTALLY','1234567891')
3 INSERT INTO DONOR VALUES ('A','RAH','REJOY','12-JAN-2001','20 YEARS','F','U','NORMAL','1234567891')
4 -- Create view showing all rows where height >= 48 inches
5 CREATE VIEW V_DONORS AS SELECT * FROM DONOR WHERE HEIGHT >= 48;
```

CODE	BLOOD_TYPE	QUANTITY
1.	A+	150
2.	A-	245
3.	B+	170
4.	B-	110
5.	AB+	100
6.	AB-	225
7.	O+	210
8.	O-	205

Download CSV
0 rows selected.

Table created.

© 2022 Oracle - See SQL 22.2, running Oracle Database 19c Enterprise Edition - 19.14.0.0.0 - Database Documentation - Ask Tom - Site Map

Built with Visual Studio Code - Release Notes | Terms of Use

[Feedback](#)
[Help](#)
[mailto:nrd810@unimelb.edu.au](#)

SQL Worksheet

Clear
Find
Actions
Save
Run

```

1 CREATE TABLE DONOR ( FIRST_NAME VARCHAR(250) NOT NULL, MIDDLE_NAME VARCHAR(250) NOT NULL, LAST_NAME VARCHAR(250) NOT NULL, DOB DATE, AGE NUMBER, HEIGHT NUMBER, GENDER CHAR(1)
2 INSERT INTO DONOR VALUES ('SAGORE', 'NAGAN', 'KUMAR', '17-DEC-2000', '20 YEARS', '87', 'H', 'KUMATRALY', '234567891');
3 INSERT INTO DONOR VALUES ('A', 'BAN', 'REDDY', '33-JAN-2001', '28 YEARS', '71', 'H', 'REDDIPALLY', '2345678912');
4 TRUNCATE TABLE DONOR;
5
6

```

ORDERS	BLOOD_TYPE	QUANTITY
3	A+	13
4	A-	8
5	B+	0
1	B-	5
2	AB-	15
6	AB+	9
4	O-	11
7	O+	8

Download SQL
5 rows selected.
Table created.

© 2022 Oracle / Live SQL 22.2 - running Oracle Database 19c Freeform Edition - 19.14.0.0.0 - Database Configuration - Ask Tom - Day 504

[illegible]

SAMPLE QUERIES:

SELECT FIRST_NAME,MIDDLE_NAME,LAST_NAME,DOB FROM DONOR WHERE DOB LIKE "%2002";

OUTPUT:

TAGORE|PAVAN|KUMAR|17-DEC-2002

G|MANIDEEP|REDDY|14-FEB-2002

G|RITEESH|DON|11-NOV-2002

OUTPUT NUMBER OF DONORS HAVING CERTAIN AGE

INPUT:

SELECT AGE,COUNT(*) FROM DONOR GROUP BY AGE;

OUTPUT:

18 YEARS|1

19 YEARS|3

19 Years|1

20 years|1

OUTPUT THE DIFFERENT BLOOD TYPES INPUT: SELECT BLOOD_TYPE FROM BLOOD GROUP BY BLOOD_TYPE; OUTPUT:

A+

A- A

B+

AB-

B+

B- O

+

O-

OUTPUT THE PATIENTS NAME WITH SELECT

ID,NAME FROM PATIENT GROUP BY ID;

OUTPUT:

1204|RAGHU

1205|SOWMYA

1206|SHYAM

1207|MEGHANA

1208|SHIVA

1209|RANGA RAO

1210|SHIVANI

1211|KIRAN

1212|ADITYA

1213|CHARANI

SNAPSHOTS:

Live SQL

Feedback

Help

nm610@srmsl.edu.in

SQL Worksheet

Clear

Find

Actions

Save

Run

BLOOD_TYPE

O+

AB+

B+

A+

A+

B+

O+

AB+

Download CSV

8 rows selected

© 2022 Oracle - Live SQL 22.3.2, running Oracle Database 19c Enterprise Edition - 19.14.0.0.0 - Database Documentation - Ask Tom - DevGym

Built with using Oracle APEX - Privacy - Terms of Use

Live SQL

Feedback

Help

nm610@srmsl.edu.in

SQL Worksheet

Clear

Find

Actions

Save

Run

1 SELECT ID,NAME FROM PATIENT 1

2

3

4

ID	NAME
1204	RAGHU
1205	SOUNYA
1206	SHASHI
1207	MEGHANA
1208	SHIVA
1209	RAHSA RAO
1210	SHIVANI
1211	KIRAN
1212	ADITHYAN
1213	CHARAN

Download CSV

10 rows selected

© 2022 Oracle - Live SQL 22.3.2, running Oracle Database 19c Enterprise Edition - 19.14.0.0.0 - Database Documentation - Ask Tom - DevGym

Built with using Oracle APEX - Privacy - Terms of Use

6 RESULTS AND DISCUSSIONS

Database design and management: The success of any blood bank management system depends on its ability to effectively manage the database. Results can be presented on the database design, normalization, and indexing used in the system.

Discussions can focus on the performance and scalability of the database, as well as any challenges encountered in managing the database

User interface: The user interface is a critical aspect of any blood bank management system, as it directly impacts user experience. Results can be presented on the user interface design, including the layout, colors, and typography used. Discussions can focus on the ease of use of the system, user feedback, and any modifications made based on user feedback.

7 CONCLUSION:

To summarize and conclude, in this course based project, we had created on Blood bank database system. Furthermore, we have given information about the Donor basic details and their blood type, quantity and that donated blood is stored in the blood banks so, that the patients can collect the required blood easily from the blood bank. Throughout the project, we have made use of concepts taught to us, including DDL, DML commands, Primary and Foreign Keys e.t.c. Furthermore, to practically show the usage of the database, a large number of sample queries for each and every entity based tables have been provided. We believe that a model like ours can be implemented seamlessly in order to make the experience of reading pleasurable and comfortable to the end user.

FUTURE ENHANCEMENT

Integration with other healthcare systems: A blood bank management system could be integrated with other healthcare systems such as electronic health records (EHRs) to improve the flow of information and ensure continuity of care for patients

Donor recruitment and retention: A blood bank management system could be enhanced to include features that facilitate donor recruitment and retention, such as targeted messaging, reminders for donors, and rewards or incentives

Mobile application: A mobile application could be developed to allow donors to easily schedule appointments, view their donation history, and receive notifications about blood drives and shortages

Overall, there are many potential future enhancements that could be implemented in a blood bank management system to improve its functionality, efficiency, and effectiveness. The specific enhancements chosen will depend on the needs of the blood bank and the goals of the system.

8 REFERENCES

- [1] Hernández, E.; Sanchez-Anguix, V.; Julian, V.; Palanca, J.; Duque, N. Rainfall prediction: A deep learning approach. In International Conference on Hybrid Artificial Intelligence Systems; Springer: Cham, Switzerland, 2016; pp. 151–162.
- [2] Goswami, B.N. The challenge of weather prediction. *Resonance* **1996**, 1, 8–17.
- [3] Nayak, D.R.; Mahapatra, A.; Mishra, P. A survey on rainfall prediction using artificial neural network. *Int. J. Comput. Appl.* **2013**, 72, 16.
- [4] Kashiwao, T.; Nakayama, K.; Ando, S.; Ikeda, K.; Lee, M.; Bahadori, A. A neural network-based local rainfall prediction system using meteorological data on the internet: A case study using data from the Japan meteorological agency. *Appl. Soft Comput.* **2017**, 56, 317–330.
- [5] Mislan, H.; Hardwinarto, S.; Sumaryono, M.A. Rainfall monthly prediction based on artificial neural network: A case study in Tenggarong Station, East Kalimantan, Indonesia. *Procedia Comput. Sci.* **2015**, 59, 142–151.
- [6] Muka, Z.; Maraj, E.; Kuka, S. Rainfall prediction using fuzzy logic. *Int. J. Innov. Sci. Eng. Technol.* **2017**, 4, 1–5.