AMERICAN INTERNATIONAL UNIVERSITY-BANGLADESH

DATABASE PROJECT

Submitted By:-

Group No: - 08

Course name: - Database

Section: - M

Name of the department: - CSE

Submitted To:-

Teacher's name: - Md. Sohidul Islam

Designation: - Assistant Professor, Computer Science

University name: - AIUB

Date of Submission:-

December 19, 2020

Group member list:-

08.	20-42191-1	Md. Tanvir Rahman Shafiullah
18.	20-42585-1	Saiful Islam
28.	20-42940-1	Md Ashraful Amin Sakib
38.	20-43365-1	Zarin Tasnim

Database Project

Introduction: - In this, project we represent our project by the help of ER diagram. Overall, this whole project displays 'Blood Bank Management System'. Blood Bank management system is a system that is designed to store, process and analyze information concerned with the works within a blood bank. This project aims at maintaining all the information pertaining to blood donors.

Scenario Description:- This is about to be a database management system about a blood bank. In this database the work will start with a table called receptionist which will include the information about the receptionist who will manage all the work first. After that we will work with a table called donor which will include the information about the donor. Then comes Blood table which will include the information that which type of bloods the groups and ids about the blood. Then comes the Blood Bank table which will include the information about the orders of the blood. Then there is Hospital table which will give the orders of blood. And the final table is Patient which is all about the information of patients.

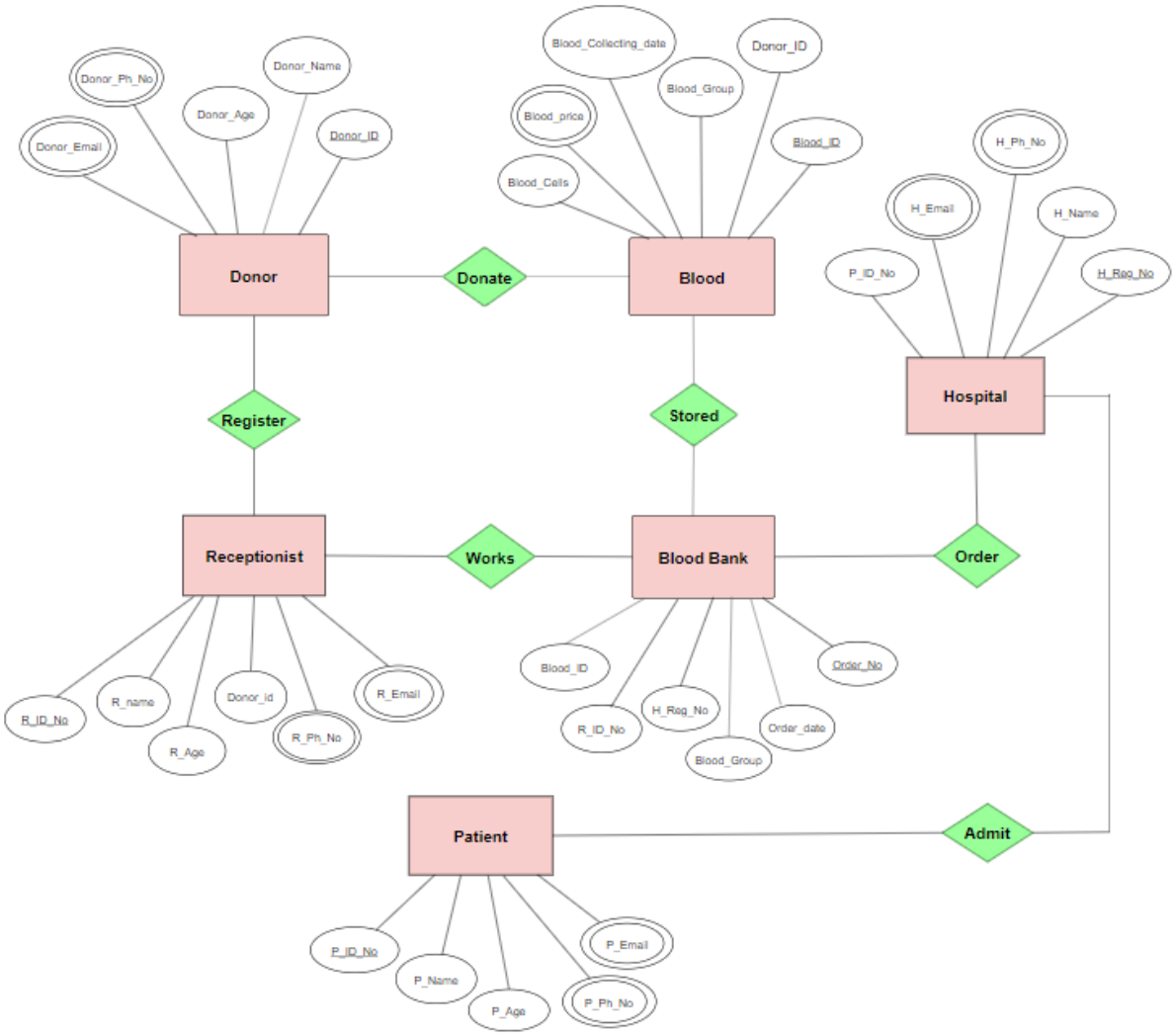
Features:

1. Donor management: donor registration, managing donor database, store and collecting blood.

2. Patient Register/Blood Sample Receiving Register, Donor Register.

3. List of the donor who is eligible for donation on a particular date with contact number.

ER DIAGRAM


Donor_Ph_No
Donor_Name
Donor_Age
Donor_Email
Donor_ID
Donor
Donate
Blood_Collecting_date
Donor_ID
Blood_Group
Blood_price
Blood_ID
Blood_Cells
Blood
H_Ph_No
H_Email
H_Name
P_ID_No
H_Reg_No
Hospital
Register
Stored
Receptionist
Works
Blood Bank
Order
R_ID_No
R_name
Donor_id
R_Email
R_Age
R_Ph_No
Blood_ID
Order_No
R_ID_No
H_Reg_No
Order_date
Blood_Group
Patient
Admit
P_ID_No
P_Name
P_Age
P_Email
P_Ph_No

Normalization:

Register:

UNF

Register (DONOR_ID, DONOR_NAME, DONOR_AGE, DONOR_PH_NO, DONOR_EMAIL, R_ID_NO, R_NAME, R_AGE, R_PH_NO, R_EMAIL, DONOR_ID)

1NF

DONOR_PH_NO, DONOR_EMAIL, R_PH_NO, R_EMAIL are multivalued attribute.

1. DONOR_ID, DONOR_NAME, DONOR_AGE, DONOR_PH_NO, DONOR_EMAIL, R_ID_NO, R_NAME, R_AGE, R_PH_NO, R_EMAIL, DONOR_ID

2NF

1. DONOR_ID, DONOR_NAME, DONOR_AGE, DONOR_PH_NO, DONOR_EMAIL

2. R_ID_NO, R_NAME, R_AGE, R_PH_NO, R_EMAIL, DONOR_ID

3NF

There is no transitive dependency. Relation already in 3NF

1. DONOR_ID, DONOR_NAME, DONOR_AGE, DONOR_PH_NO, DONOR_EMAIL

2. R_ID_NO, R_NAME, R_AGE, R_PH_NO, R_EMAIL, DONOR_ID

Table Creation:

1. DONOR_ID, DONOR_NAME, DONOR_AGE, DONOR_PH_NO, DONOR_EMAIL

2. R_ID_NO, R_NAME, R_AGE, R_PH_NO, R_EMAIL, **DONOR_ID**

Donate

UNF

Donate (DONOR_ID, DONOR_NAME, DONOR_AGE, DONOR_PH_NO, DONOR_EMAIL, BLOOD_ID, BLOOD_COLLECTING_DATE, BLOOD_GROUP, BLOOD_PRICE, BLOOD_CELLS, DONOR_ID)

1NF

DONOR_PH_NO, DONOR_EMAIL, are multivalued attribute.

1. DONOR_ID, DONOR_NAME, DONOR_AGE, DONOR_PH_NO, DONOR_EMAIL, BLOOD_ID, BLOOD_COLLECTING_DATE, BLOOD_GROUP, BLOOD_PRICE, BLOOD_CELLS, DONOR_ID

2NF

1. DONOR_ID, DONOR_NAME, DONOR_AGE, DONOR_PH_NO, DONOR_EMAIL

2. BLOOD_ID, BLOOD_COLLECTING_DATE, BLOOD_GROUP, BLOOD_PRICE, BLOOD_CELLS, DONOR_ID

3NF

There is no transitive dependency .Relation already in 3NF

1. DONOR_ID, DONOR_NAME, DONOR_AGE, DONOR_PH_NO, DONOR_EMAIL

2. BLOOD_ID, BLOOD_COLLECTING_DATE, BLOOD_GROUP, BLOOD_PRICE, BLOOD_CELLS, DONOR_ID

Table Creation:

1. DONOR_ID, DONOR_NAME, DONOR_AGE, DONOR_PH_NO, DONOR_EMAIL

2. BLOOD_ID, BLOOD_COLLECTING_DATE, BLOOD_GROUP, BLOOD_PRICE, BLOOD_CELLS, **DONOR_ID**

Works:

UNF

Works (R_ID_NO, R_NAME, R_AGE, R_PH_NO, R_EMAIL, ORDER_NO, ORDER_DATE, BLOOD_GROUP, BLOOD_ID, H_REG_NO, R_ID_NO)

1NF

R_PH_NO, R_EMAIL are multivalued attribute.

1. R_ID_NO, R_NAME, R_AGE, R_PH_NO, R_EMAIL, ORDER_NO, ORDER_DATE, BLOOD_GROUP, BLOOD_ID, H_REG_NO, R_ID_NO

2NF

1. R_ID_NO, R_NAME, R_AGE, R_PH_NO, R_EMAIL

2. ORDER_NO, ORDER_DATE, BLOOD_GROUP, BLOOD_ID, H_REG_NO, R_ID_NO

3NF

There is no transitive dependency .Relation already in 3NF

1. R_ID_NO, R_NAME, R_AGE, R_PH_NO, R_EMAIL

2. ORDER_NO, ORDER_DATE, BLOOD_GROUP, BLOOD_ID, H_REG_NO, R_ID_NO

Table Creation:

1. R_ID_NO, R_NAME, R_AGE, R_PH_NO, R_EMAIL

2. ORDER_NO, ORDER_DATE, BLOOD_GROUP, BLOOD_ID, H_REG_NO, **R_ID_NO**

Stored:

UNF

Stored (BLOOD_ID, BLOOD_COLLECTING_DATE, BLOOD_GROUP, BLOOD_PRICE, BLOOD_CELLS, ORDER_NO, ORDER_DATE, BLOOD_GROUP, BLOOD_ID, H_REG_NO, R_ID_NO)

1NF

1. BLOOD_ID, BLOOD_COLLECTING_DATE, BLOOD_GROUP, BLOOD_PRICE, BLOOD_CELLS, ORDER_NO, ORDER_DATE, BLOOD_GROUP, BLOOD_ID, H_REG_NO, R_ID_NO

2NF

1. BLOOD_ID, BLOOD_COLLECTING_DATE, BLOOD_GROUP, BLOOD_PRICE, BLOOD_CELLS

2. ORDER_NO, ORDER_DATE, BLOOD_GROUP, BLOOD_ID, H_REG_NO, R_ID_NO

3NF

There is no transitive dependency. Relation already in 3NF.

1. BLOOD_ID, BLOOD_COLLECTING_DATE, BLOOD_GROUP, BLOOD_PRICE, BLOOD_CELLS

2. ORDER_NO, ORDER_DATE, BLOOD_GROUP, BLOOD_ID, H_REG_NO, R_ID_NO

Table Creation:

1. BLOOD_ID, BLOOD_COLLECTING_DATE, BLOOD_GROUP, BLOOD_PRICE, BLOOD_CELLS

2. ORDER_NO, ORDER_DATE, BLOOD_GROUP, **BLOOD_ID**, H_REG_NO, R_ID_NO

Admit:

UNF:

Admit (P_ID_NO, P_NAME, P_AGE, P_PH_NO, P_EMAIL, H_REG_NO, H_NAME, H_PH_NO, H_EMAIL, P_ID_NO)

1NF

P_PH_NO, P_EMAIL, H_PH_NO, H_EMAIL are multivalued attribute.

1. P_ID_NO, P_NAME, P_AGE, P_PH_NO, P_EMAIL, H_REG_NO, H_NAME, H_PH_NO, H_EMAIL, P_ID_NO

2NF

1. P_ID_NO, P_NAME, P_AGE, P_PH_NO, P_EMAIL

2. H_REG_NO, H_NAME, H_PH_NO, H_EMAIL, P_ID_NO

3NF

There is no transitive dependency. Relation already in 3NF.

1. P_ID_NO, P_NAME, P_AGE, P_PH_NO, P_EMAIL

2. H_REG_NO, H_NAME, H_PH_NO, H_EMAIL, P_ID_NO

Table Creation:

1. P_ID_NO, P_NAME, P_AGE, P_PH_NO, P_EMAIL

2. H_REG_NO, H_NAME, H_PH_NO, H_EMAIL**, P_ID_NO**

Order:

UNF:

Order (H_REG_NO, H_NAME, H_PH_NO, H_EMAIL, ORDER_NO, ORDER_DATE, BLOOD_GROUP, BLOOD_ID, H_REG_NO)

1NF

H_PH_NO, H_EMAIL are multivalued attribute.

1. H_REG_NO, H_NAME, H_PH_NO, H_EMAIL, ORDER_NO, ORDER_DATE, BLOOD_GROUP, BLOOD_ID, H_REG_NO

2NF

1. H_REG_NO, H_NAME, H_PH_NO, H_EMAIL

2. ORDER_NO, ORDER_DATE, BLOOD_GROUP, BLOOD_ID, H_REG_NO

3NF

There is no transitive dependency. Relation already in 3NF.

1. H_REG_NO, H_NAME, H_PH_NO, H_EMAIL

2. ORDER_NO, ORDER_DATE, BLOOD_GROUP, BLOOD_ID, H_REG_NO

Table Creation:

1. H_REG_NO, H_NAME, H_PH_NO, H_EMAIL

2. ORDER_NO, ORDER_DATE, BLOOD_GROUP, BLOOD_ID, **H_REG_NO**

Temporary Tables:

1. DONOR_ID, DONOR_NAME, DONOR_AGE, DONOR_PH_NO, DONOR_EMAIL

2. R_ID_NO, R_NAME, R_AGE, R_PH_NO, R_EMAIL, **DONOR_ID**

3. BLOOD_ID, BLOOD_COLLECTING_DATE, BLOOD_GROUP, BLOOD_PRICE, BLOOD_CELLS, **DONOR_ID**

4. R_ID_NO, R_NAME, R_AGE, R_PH_NO, R_EMAIL

5. ORDER_NO, ORDER_DATE, BLOOD_GROUP, BLOOD_ID, H_REG_NO, **R_ID_NO**

6. BLOOD_ID, BLOOD_COLLECTING_DATE, BLOOD_GROUP, BLOOD_PRICE, BLOOD_CELLS

7. ORDER_NO, ORDER_DATE, BLOOD_GROUP, **BLOOD_ID**, H_REG_NO, R_ID_NO

8. P_ID_NO, P_NAME, P_AGE, P_PH_NO, P_EMAIL

9. H_REG_NO, H_NAME, H_PH_NO, H_EMAIL**, P_ID_NO**

10. H_REG_NO, H_NAME, H_PH_NO, H_EMAIL

11. ORDER_NO, ORDER_DATE, BLOOD_GROUP, BLOOD_ID, **H_REG_NO**

Final Tables:

1. DONOR_ID, DONOR_NAME, DONOR_AGE, DONOR_PH_NO, DONOR_EMAIL

2. R_ID_NO, R_NAME, R_AGE, R_PH_NO, R_EMAIL, **DONOR_ID**

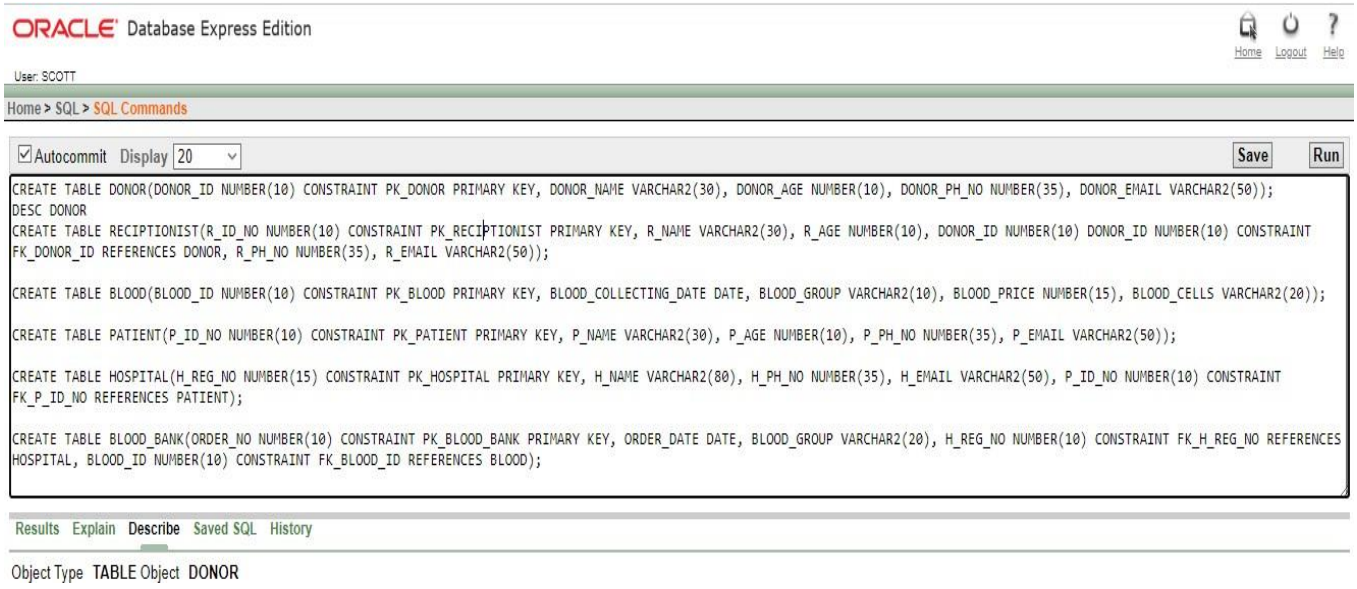
3. BLOOD_ID, BLOOD_COLLECTING_DATE, BLOOD_GROUP, BLOOD_PRICE, BLOOD_CELLS, **DONOR_ID**

4. P_ID_NO, P_NAME, P_AGE, P_PH_NO, P_EMAIL

5. H_REG_NO, H_NAME, H_PH_NO, H_EMAIL, **P_ID_NO**

6. ORDER_NO, ORDER_DATE, BLOOD_GROUP, **H_REG_NO, BLOOD_ID, R_ID_NO**

Table creation:-



1. CREATE TABLE DONOR(DONOR_ID NUMBER(10) CONSTRAINT PK_DONOR PRIMARY KEY, DONOR_NAME VARCHAR2(30), DONOR_AGE NUMBER(10), DONOR_PH_NO NUMBER(35), DONOR_EMAIL VARCHAR2(50));

Results Explain Describe Saved SQL History

Object Type TABLE Object DONOR

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
DONOR	DONOR_ID	Number	-	10	0	1	-	-	-
	DONOR_NAME	Varchar2	30	-	-	-	✓	-	-
	DONOR_AGE	Number	-	10	0	-	✓	-	-
	DONOR_PH_NO	Number	-	35	0	-	✓	-	-
	DONOR_EMAIL	Varchar2	50	-	-	-	✓	-	-

1 - 5

Language: en-us

2. CREATE TABLE RECIPTIONIST(R_ID_NO NUMBER(10) CONSTRAINT PK_RECIPTIONIST PRIMARY KEY, R_NAME VARCHAR2(30), R_AGE NUMBER(10), DONOR_ID NUMBER(10) CONSTRAINT FK_DONOR_ID REFERENCES DONOR, R_PH_NO NUMBER(35), R_EMAIL VARCHAR2(50));

Results Explain Describe Saved SQL History

Object Type TABLE Object RECIPTIONIST

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
RECIPTIONIST	R_ID_NO	Number	-	10	0	1	-	-	-
	R_NAME	Varchar2	30	-	-	-	✓	-	-
	R_AGE	Number	-	10	0	-	✓	-	-
	DONOR_ID	Number	-	10	0	-	✓	-	-
	R_PH_NO	Number	-	35	0	-	✓	-	-
	R_EMAIL	Varchar2	50	-	-	-	✓	-	-

1 - 6

Language: en-us

3. CREATE TABLE BLOOD(BLOOD_ID NUMBER(10) CONSTRAINT PK_BLOOD PRIMARY KEY, BLOOD_COLLECTING_DATE DATE, BLOOD_GROUP VARCHAR2(10), BLOOD_PRICE NUMBER(15), BLOOD_CELLS VARCHAR2(20), DONOR_ID NUMBER(10));

Results Explain Describe Saved SQL History

Object Type TABLE Object BLOOD

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
BLOOD	BLOOD_ID	Number	-	10	0	1	-	-	-
	BLOOD_COLLECTING_DATE	Date	7	-	-	-	✓	-	-
	BLOOD_GROUP	Varchar2	10	-	-	-	✓	-	-
	BLOOD_PRICE	Number	-	15	0	-	✓	-	-
	BLOOD_CELLS	Varchar2	20	-	-	-	✓	-	-
	DONOR_ID	Number	-	10	0	-	✓	-	-

1 - 6

4. CREATE TABLE PATIENT(P_ID_NO NUMBER(10) CONSTRAINT PK_PATIENT PRIMARY KEY, P_NAME VARCHAR2(30), P_AGE NUMBER(10), P_PH_NO NUMBER(35), P_EMAIL VARCHAR2(50));

Results Explain Describe Saved SQL History

Object Type TABLE Object PATIENT

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
PATIENT	P_ID_NO	Number	-	10	0	1	-	-	-
	P_NAME	Varchar2	30	-	-	-	✓	-	-
	P_AGE	Number	-	10	0	-	✓	-	-
	P_PH_NO	Number	-	35	0	-	✓	-	-
	P_EMAIL	Varchar2	50	-	-	-	✓	-	-

1 - 5

5. CREATE TABLE HOSPITAL(H_REG_NO NUMBER(15) CONSTRAINT PK_HOSPITAL PRIMARY KEY, H_NAME VARCHAR2(80), H_PH_NO NUMBER(35), H_EMAIL VARCHAR2(50), P_ID_NO NUMBER(10) CONSTRAINT FK_P_ID_NO REFERENCES PATIENT);

Results Explain Describe Saved SQL History

Object Type TABLE Object HOSPITAL

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
HOSPITAL	H_REG_NO	Number	-	15	0	1	-	-	-
	H_NAME	Varchar2	80	-	-	-	✓	-	-
	H_PH_NO	Number	-	35	0	-	✓	-	-
	H_EMAIL	Varchar2	50	-	-	-	✓	-	-
	P_ID_NO	Number	-	10	0	-	✓	-	-

1 - 5

6. CREATE TABLE BLOOD_BANK(ORDER_NO NUMBER(10) CONSTRAINT PK_BLOOD_BANK PRIMARY KEY, ORDER_DATE DATE, BLOOD_GROUP VARCHAR2(20), H_REG_NO NUMBER(10) CONSTRAINT FK_H_REG_NO REFERENCES HOSPITAL, BLOOD_ID NUMBER(10)

CONSTRAINT FK_BLOOD_ID REFERENCES BLOOD, R_ID_NO NUMBER(10) CONSTRAINT FK_R_ID_NO REFERENCES RECIPTIONIST);

Results Explain Describe Saved SQL History

Object Type TABLE Object BLOOD_BANK

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
BLOOD_BANK	ORDER_NO	Number	-	10	0	1	-	-	-
	ORDER_DATE	Date	7	-	-	-	✓	-	-
	BLOOD_GROUP	Varchar2	20	-	-	-	✓	-	-
	H_REG_NO	Number	-	10	0	-	✓	-	-
	BLOOD_ID	Number	-	10	0	-	✓	-	-
	R_ID_NO	Number	-	10	0	-	✓	-	-

1 - 6

Language: en-us

Data Insertion:-

Insert value in DONOR table:

INSERT INTO DONOR VALUES (1000, 'SAKIB', 22, 01934567898,'sakib@gmail.com');

INSERT INTO DONOR VALUES (1001, 'RAKIB', 21, 01934147898,'rakib@gmail.com');

INSERT INTO DONOR VALUES (1002, 'ASIF', 23, 01567567898,'asif@gmail.com');

INSERT INTO DONOR VALUES (1003, 'AFIF', 29, 01934543268,'afif@gmail.com');

INSERT INTO DONOR VALUES (1004, 'RAYAN', 23, 01931234568,'rayan@gmail.com');

INSERT INTO DONOR VALUES (1005, 'IQRAM', 28, 01823457898,'iqram@gmail.com');

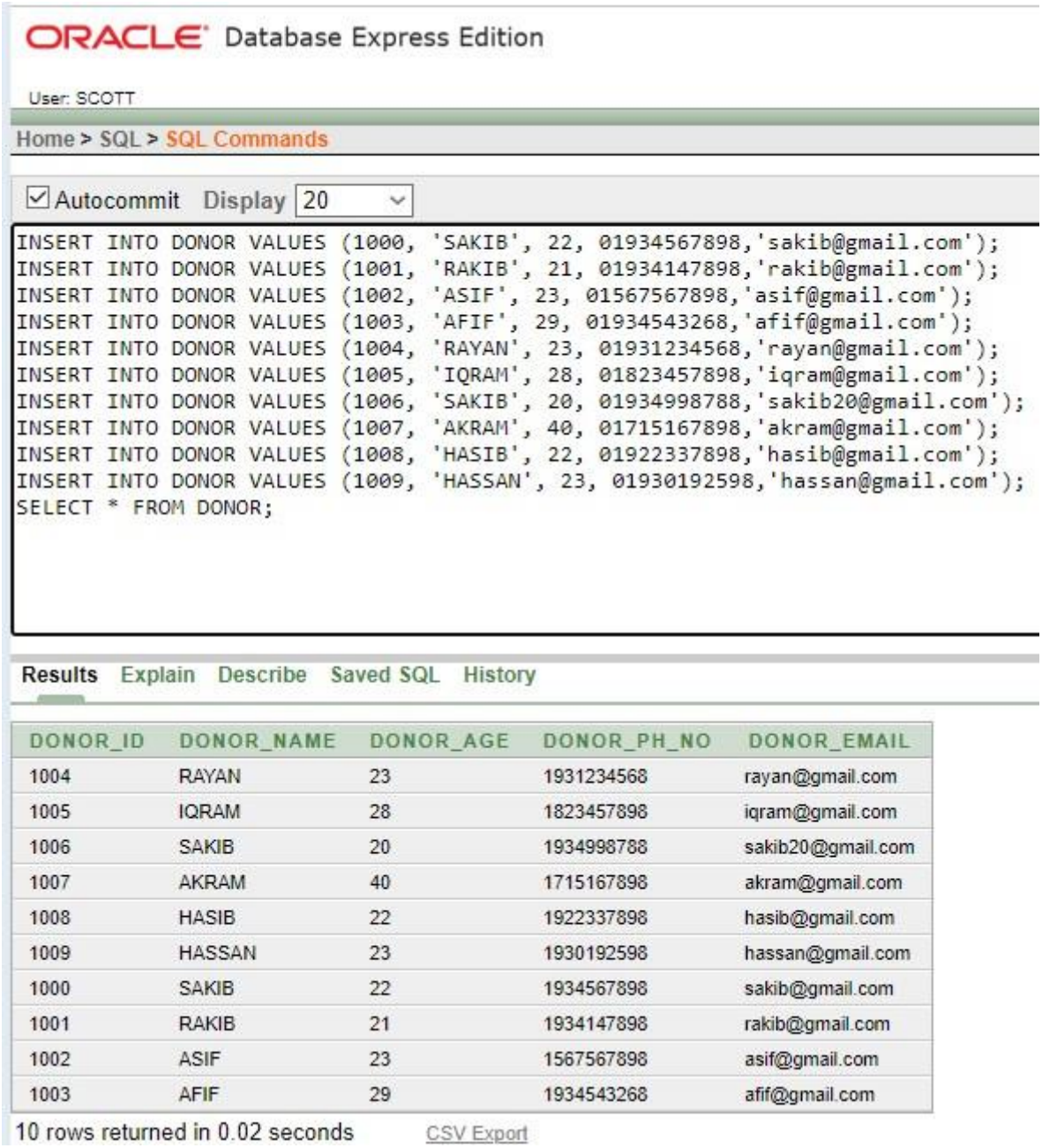
INSERT INTO DONOR VALUES (1006, 'SAKIB', 20, 01934998788,'sakib20@gmail.com');

INSERT INTO DONOR VALUES (1007, 'AKRAM', 40, 01715167898,'akram@gmail.com');

INSERT INTO DONOR VALUES (1008, 'HASIB', 22, 01922337898,'hasib@gmail.com');

INSERT INTO DONOR VALUES (1009, 'HASSAN', 23, 01930192598,'hassan@gmail.com');

SELECT * FROM DONOR;



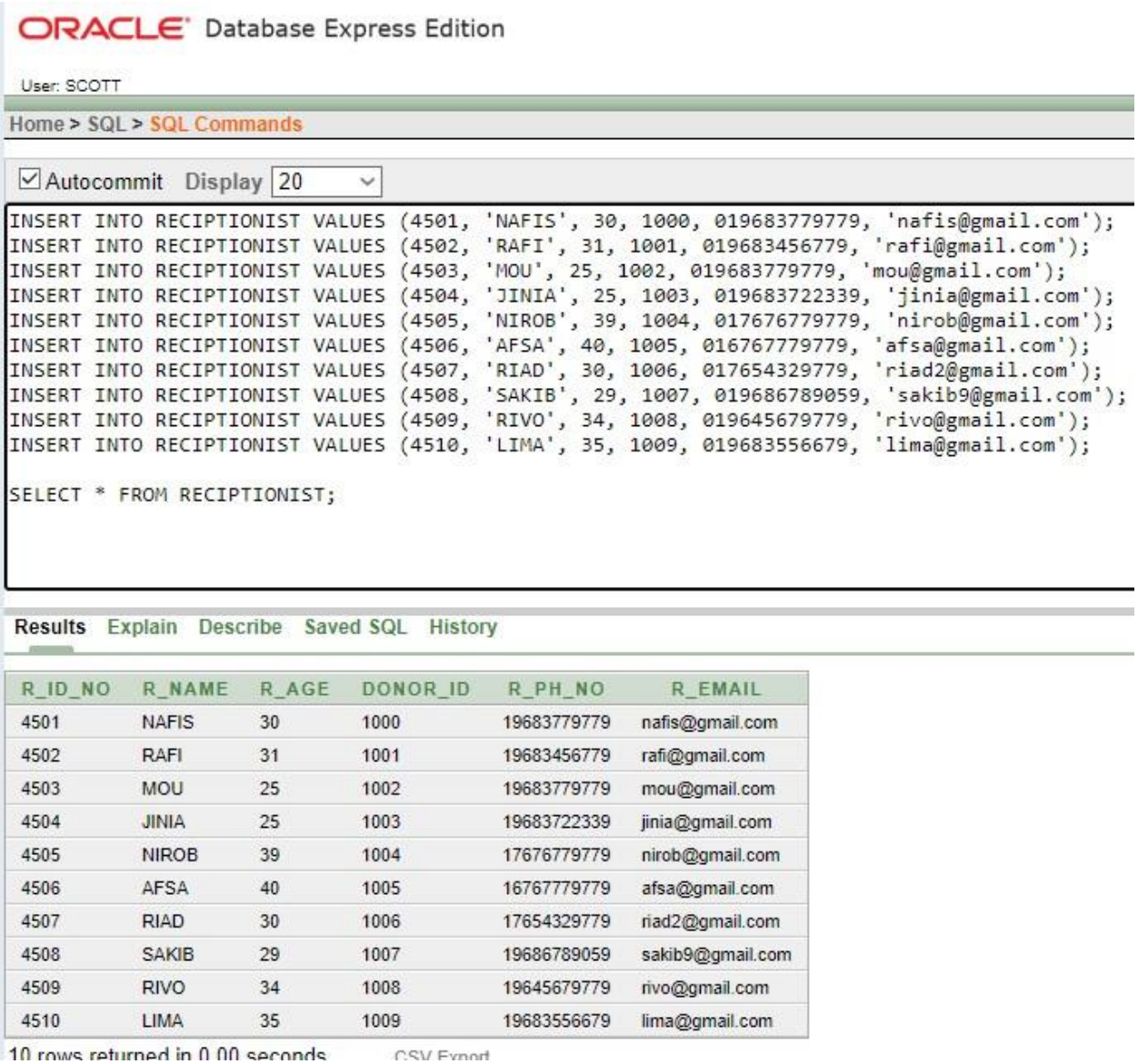
Insert value in RECIPTIONIST table:

INSERT INTO RECIPTIONIST VALUES (4501, 'NAFIS', 30, 1000, 019683779779, 'nafis@gmail.com');

INSERT INTO RECIPTIONIST VALUES (4502, 'RAFI', 31, 1001, 019683456779, 'rafi@gmail.com');

INSERT INTO RECIPTIONIST VALUES (4503, 'MOU', 25, 1002, 019683779779, 'mou@gmail.com');

```
INSERT INTO RECIPTIONIST VALUES (4504, 'JINIA', 25, 1003, 019683722339,
'jinia@gmail.com');

INSERT INTO RECIPTIONIST VALUES (4505, 'NIROB', 39, 1004, 017676779779,
'nirob@gmail.com');

INSERT INTO RECIPTIONIST VALUES (4506, 'AFSA', 40, 1005, 016767779779,
'afsa@gmail.com');

INSERT INTO RECIPTIONIST VALUES (4507, 'RIAD', 30, 1006, 017654329779,
'riad2@gmail.com');

INSERT INTO RECIPTIONIST VALUES (4508, 'SAKIB', 29, 1007, 019686789059,
'sakib9@gmail.com');

INSERT INTO RECIPTIONIST VALUES (4509, 'RIVO', 34, 1008, 019645679779,
'rivo@gmail.com');

INSERT INTO RECIPTIONIST VALUES (4510, 'LIMA', 35, 1009, 019683556679,
'lima@gmail.com');

SELECT * FROM RECIPTIONIST;
```



<u>Insert value in BLOOD table</u>:

INSERT INTO BLOOD VALUES (5980, to_date('15-8-2020','dd-mm-yyyy'), 'B POSITIVE', 1200, '3 cells counted ok', 1000);

INSERT INTO BLOOD VALUES (5981, to_date('18-8-2020','dd-mm-yyyy'), 'A POSITIVE', 1500, '3 cells counted ok', 1001);

INSERT INTO BLOOD VALUES (5982, to_date('19-8-2020','dd-mm-yyyy'), 'A NEGATIVE', 1500, '3 cells counted ok', 1002);

INSERT INTO BLOOD VALUES (5983, to_date('19-8-2020','dd-mm-yyyy'), 'B POSITIVE', 1200, '3 cells counted ok', 1003);

INSERT INTO BLOOD VALUES (5984, to_date('25-8-2020','dd-mm-yyyy'), 'B NEGATIVE', 1300, '3 cells counted ok', 1004);

INSERT INTO BLOOD VALUES (5985, to_date('28-8-2020','dd-mm-yyyy'), 'B POSITIVE', 1200, '3 cells counted ok', 1005);

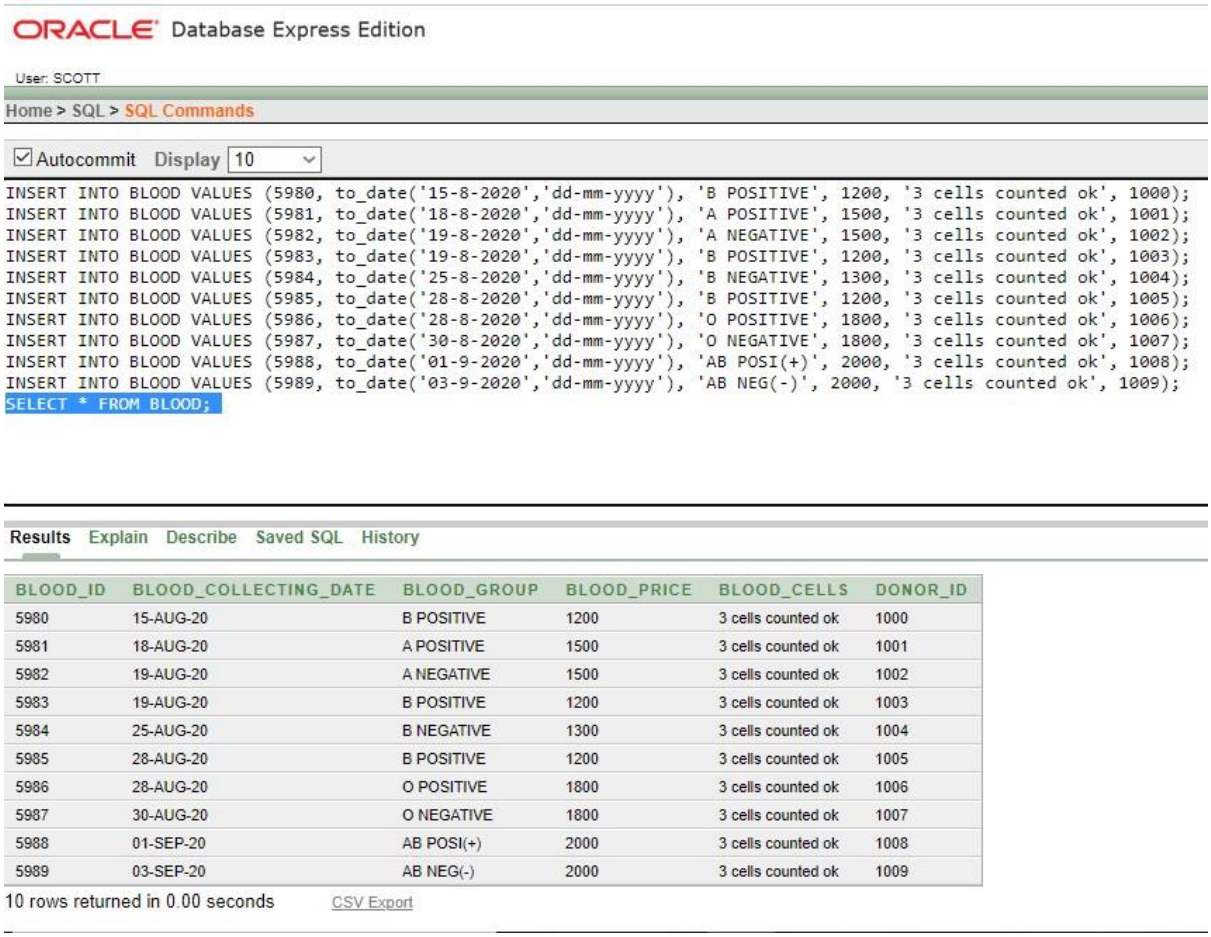
INSERT INTO BLOOD VALUES (5986, to_date('28-8-2020','dd-mm-yyyy'), 'O POSITIVE', 1800, '3 cells counted ok', 1006);

INSERT INTO BLOOD VALUES (5987, to_date('30-8-2020','dd-mm-yyyy'), 'O NEGATIVE', 1800, '3 cells counted ok', 1007);

INSERT INTO BLOOD VALUES (5988, to_date('01-9-2020','dd-mm-yyyy'), 'AB POSI(+)', 2000, '3 cells counted ok', 1008);

INSERT INTO BLOOD VALUES (5989, to_date('03-9-2020','dd-mm-yyyy'), 'AB NEG(-)', 2000, '3 cells counted ok', 1009);

SELECT * FROM BLOOD;



BLOOD_ID	BLOOD_COLLECTING_DATE	BLOOD_GROUP	BLOOD_PRICE	BLOOD_CELLS	DONOR_ID
5980	15-AUG-20	B POSITIVE	1200	3 cells counted ok	1000
5981	18-AUG-20	A POSITIVE	1500	3 cells counted ok	1001
5982	19-AUG-20	A NEGATIVE	1500	3 cells counted ok	1002
5983	19-AUG-20	B POSITIVE	1200	3 cells counted ok	1003
5984	25-AUG-20	B NEGATIVE	1300	3 cells counted ok	1004
5985	28-AUG-20	B POSITIVE	1200	3 cells counted ok	1005
5986	28-AUG-20	O POSITIVE	1800	3 cells counted ok	1006
5987	30-AUG-20	O NEGATIVE	1800	3 cells counted ok	1007
5988	01-SEP-20	AB POSI(+)	2000	3 cells counted ok	1008
5989	03-SEP-20	AB NEG(-)	2000	3 cells counted ok	1009

Insert value in PATIENT table:

INSERT INTO PATIENT VALUES (2020, 'IMRAN', 22, 01919995552, 'imranali@gmail.com');

INSERT INTO PATIENT VALUES (2021, 'PROMY', 21, 01918153612, 'promy1@gmail.com');

INSERT INTO PATIENT VALUES (2022, 'MARIAM', 44, 01918357612, 'mariam5@gmail.com');

INSERT INTO PATIENT VALUES (2023, 'KAYSER', 51, 01852171612, 'kayser@gmail.com');

INSERT INTO PATIENT VALUES (2024, 'RIA', 45, 01318171612, 'riaislam@gmail.com');

INSERT INTO PATIENT VALUES (2025, 'RAJU', 35, 01543126612, 'raju99@gmail.com');

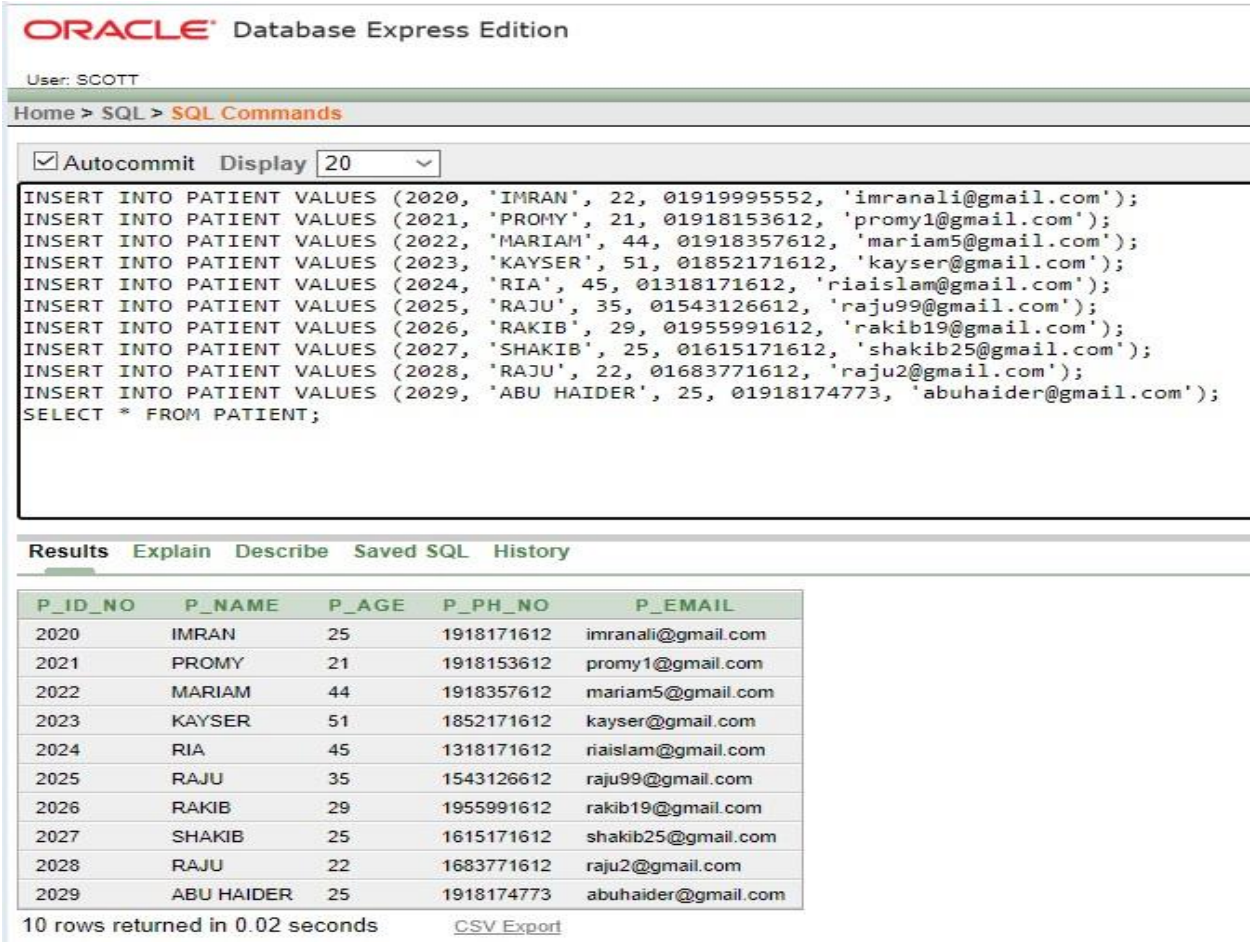
INSERT INTO PATIENT VALUES (2026, 'RAKIB', 29, 01955991612, 'rakib19@gmail.com');

INSERT INTO PATIENT VALUES (2027, 'SHAKIB', 25, 01615171612, 'shakib25@gmail.com');

INSERT INTO PATIENT VALUES (2028, 'RAJU', 22, 01683771612, 'raju2@gmail.com');

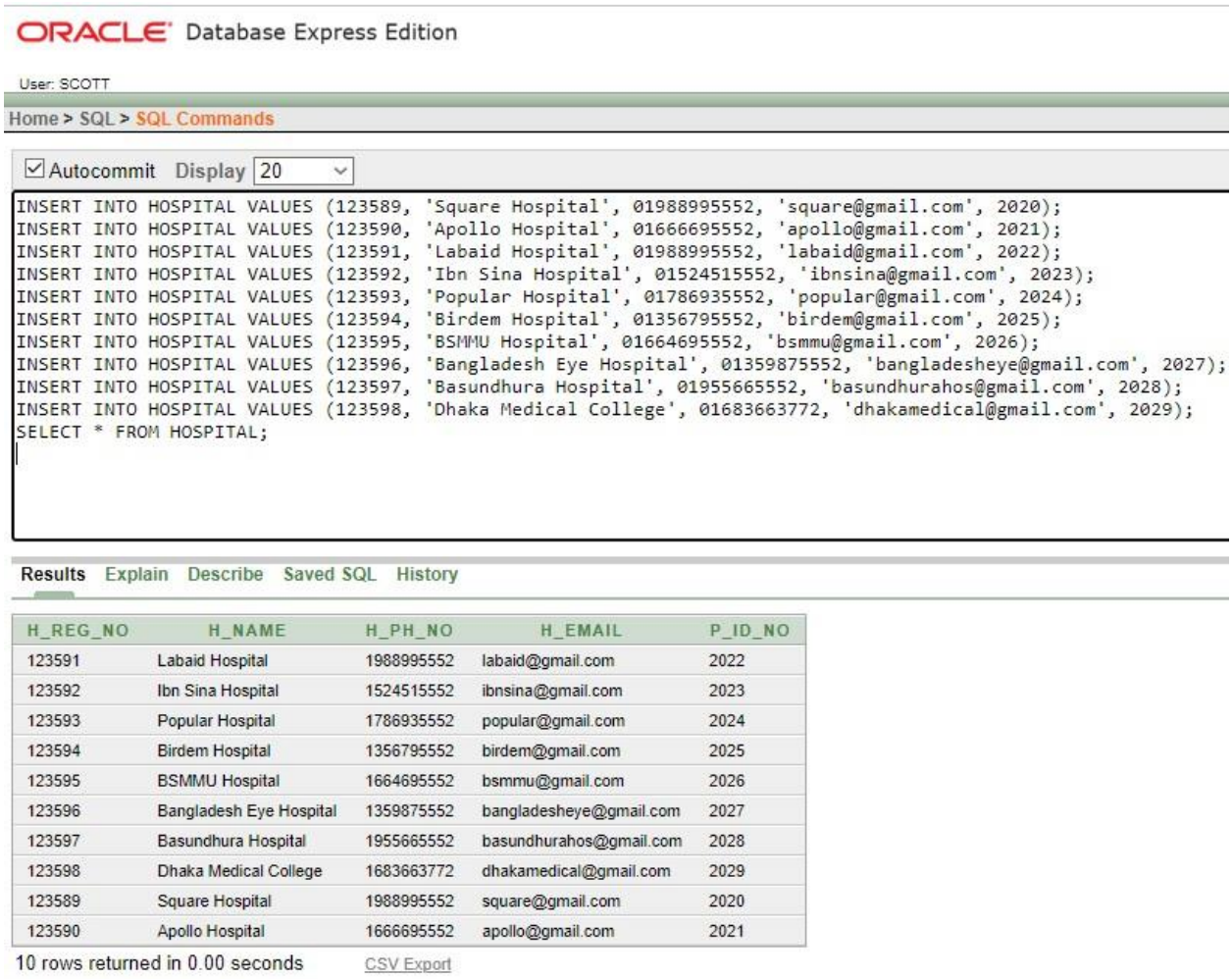
INSERT INTO PATIENT VALUES (2029, 'ABU HAIDER', 25, 01918174773, 'abuhaider@gmail.com');

SELECT * FROM PATIENT;



Insert value in HOSPITAL table:

```
INSERT INTO HOSPITAL VALUES (123589, 'Square Hospital', 01988995552,
'square@gmail.com', 2020);

INSERT INTO HOSPITAL VALUES (123590, 'Apollo Hospital', 01666695552,
'apollo@gmail.com', 2021);

INSERT INTO HOSPITAL VALUES (123591, 'Labaid Hospital', 01988995552,
'labaid@gmail.com', 2022);

INSERT INTO HOSPITAL VALUES (123592, 'Ibn Sina Hospital', 01524515552,
'ibnsina@gmail.com', 2023);

INSERT INTO HOSPITAL VALUES (123593, 'Popular Hospital', 01786935552,
'popular@gmail.com', 2024);

INSERT INTO HOSPITAL VALUES (123594, 'Birdem Hospital', 01356795552,
'birdem@gmail.com', 2025);

INSERT INTO HOSPITAL VALUES (123595, 'BSMMU Hospital', 01664695552,
'bsmmu@gmail.com', 2026);

INSERT INTO HOSPITAL VALUES (123596, 'Bangladesh Eye Hospital', 01359875552,
'bangladesheye@gmail.com', 2027);

INSERT INTO HOSPITAL VALUES (123597, 'Basundhura Hospital', 01955665552,
'basundhurahos@gmail.com', 2028);

INSERT INTO HOSPITAL VALUES (123598, 'Dhaka Medical College', 01683663772,
'dhakamedical@gmail.com', 2029);

SELECT * FROM HOSPITAL;
```



Insert value in BLOOD_BANK table:

INSERT INTO BLOOD_BANK VALUES (101, to_date('05-09-2020','dd-mm-yyyy'), 'B POSITIVE', 123589, 5980, 4501);

INSERT INTO BLOOD_BANK VALUES (102, to_date('06-09-2020','dd-mm-yyyy'), 'A POSITIVE', 123590, 5981, 4502);

INSERT INTO BLOOD_BANK VALUES (103, to_date('07-09-2020','dd-mm-yyyy'), 'O POSITIVE', 123591, 5986, 4503);

INSERT INTO BLOOD_BANK VALUES (104, to_date('08-09-2020','dd-mm-yyyy'), 'B NEGATIVE', 123592, 5984, 4504);

INSERT INTO BLOOD_BANK VALUES (105, to_date('09-09-2020','dd-mm-yyyy'), 'A NEGATIVE', 123593, 5982, 4505);

INSERT INTO BLOOD_BANK VALUES (106, to_date('09-09-2020','dd-mm-yyyy'), 'O NEGATIVE', 123594, 5987, 4506);

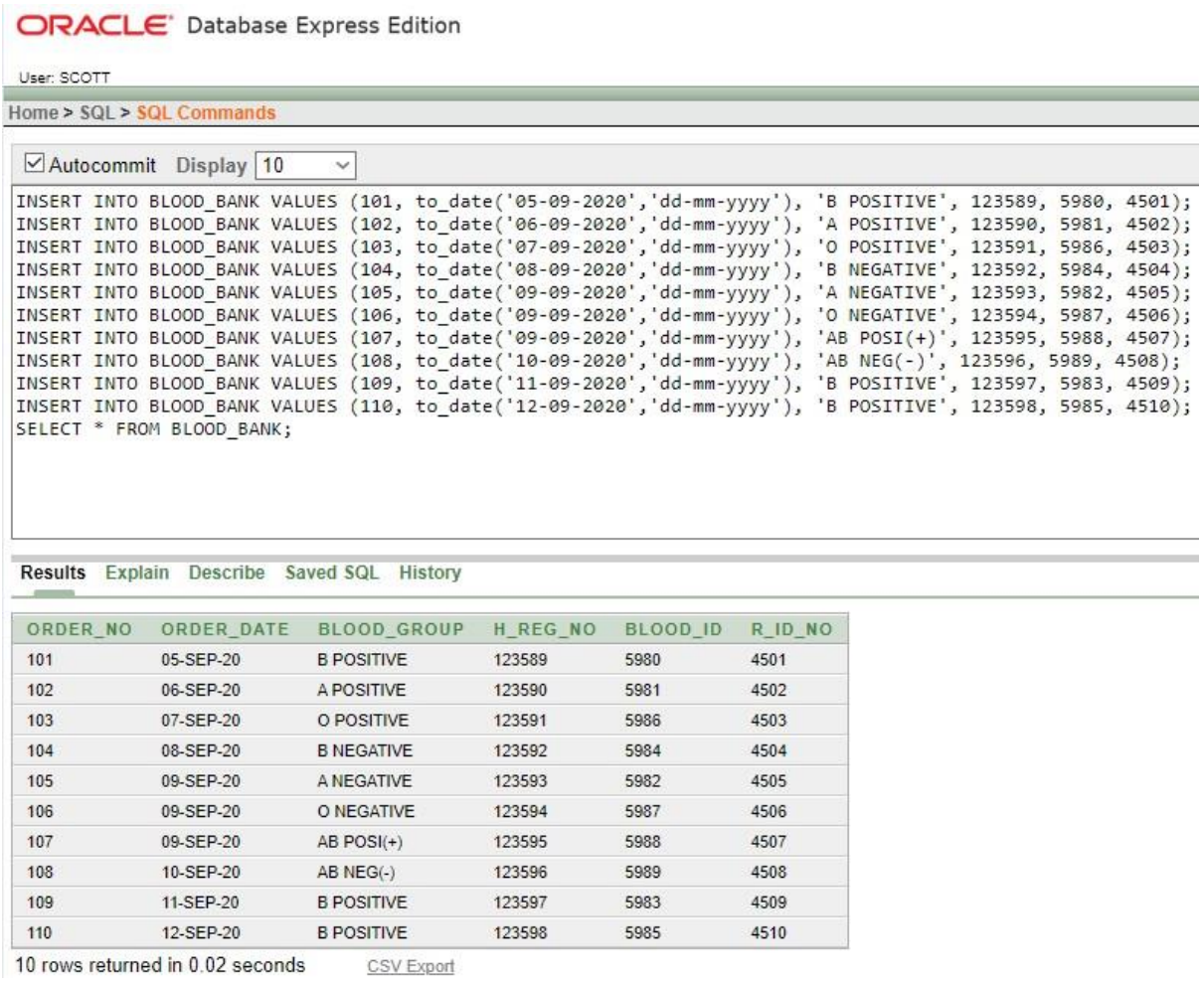
INSERT INTO BLOOD_BANK VALUES (107, to_date('09-09-2020','dd-mm-yyyy'), 'AB POSI(+)', 123595, 5988, 4507);

INSERT INTO BLOOD_BANK VALUES (108, to_date('10-09-2020','dd-mm-yyyy'), 'AB NEG(-)', 123596, 5989, 4508);

INSERT INTO BLOOD_BANK VALUES (109, to_date('11-09-2020','dd-mm-yyyy'), 'B POSITIVE', 123597, 5983, 4509);

INSERT INTO BLOOD_BANK VALUES (110, to_date('12-09-2020','dd-mm-yyyy'), 'B POSITIVE', 123598, 5985, 4510);

SELECT * FROM BLOOD_BANK;

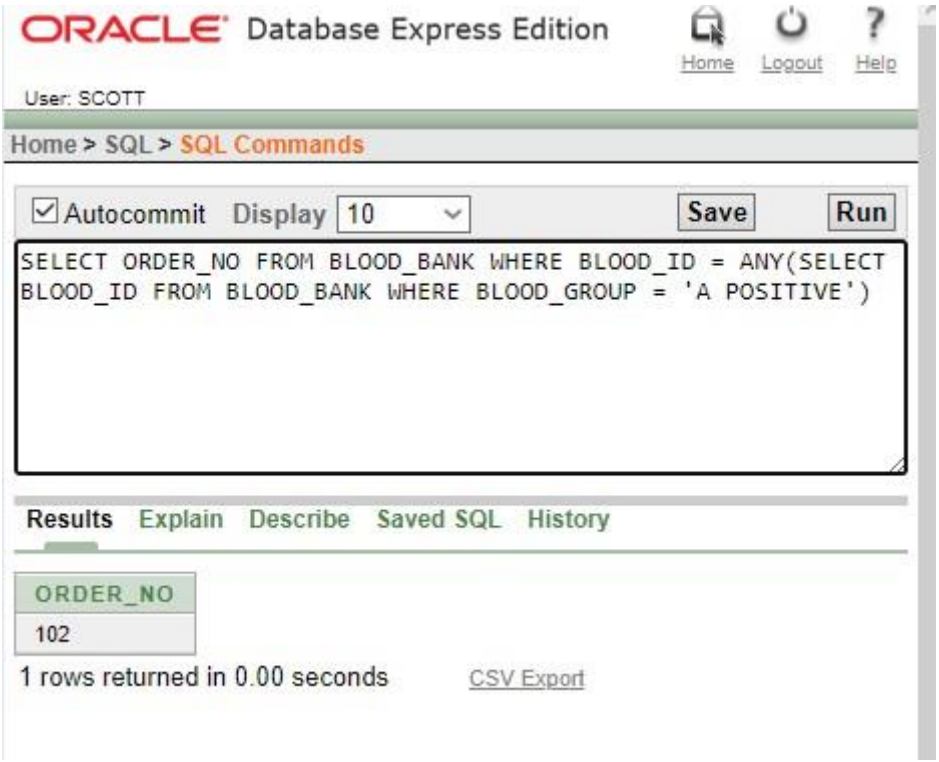


Writing Query:-

Subquery:-

Show the ORDER_NO whose BLOOD_GROUP is A POPSITIVE using subquery.

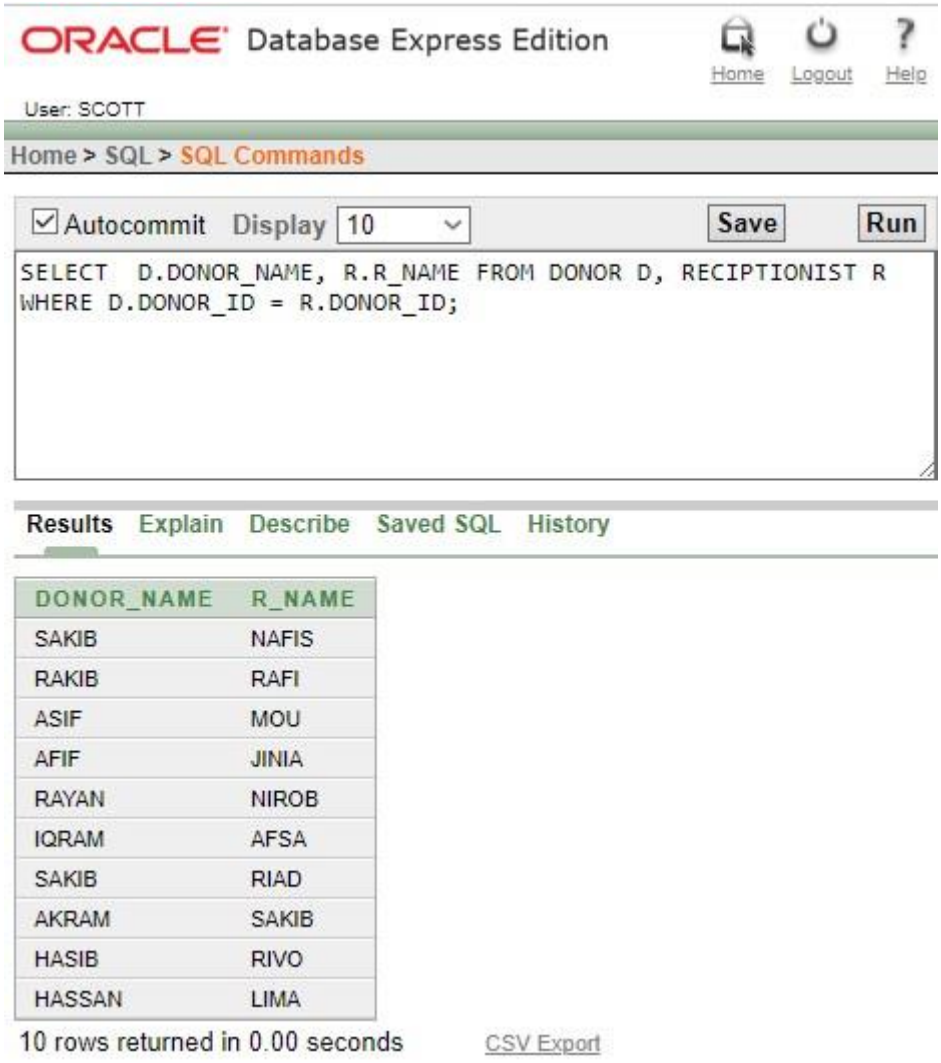
- SELECT ORDER_NO FROM BLOOD_BANK WHERE BLOOD_ID = ANY(SELECT BLOOD_ID FROM BLOOD_BANK WHERE BLOOD_GROUP = 'A POSITIVE')



Joining:-

Write a query the DONOR_NAME and R_NAME for all DONOR and RECIPTIONIST

- SELECT D.DONOR_NAME, R.R_NAME FROM DONOR D, RECIPTIONIST R WHERE D.DONOR_ID = R.DONOR_ID;



Conclusion:-

This project gives us more than enough opportunity to enhance our knowledge about how to design, code, diagram, creating table inserting value measure and execute. This has helped to implement the different Database management principles Concepts such as data integrity and continuity.

*Future Enhancement
Since there were a small amount of contact information, it may be difficult for some people to get blood quickly. I would like to gather more information about contacts in other cities and villages and will provide people with much more support to connect all of us with morality.