A
Project Report
on

Foundation Management System

Developed by

YASH SUHAGIYA(IT-115) – Department of IT, DD University
PARAS SUVAGIYA(IT-116) - Department of IT, DD University

Guided By
Internal Guide:
Prof. Roshni M. Raval
Department of Information Technology
Faculty of Technology
DD University

Department of Information Technology
Faculty of Technology, Dharmsinh Desai University
College Road, Nadiad-387001
October-2019

DHARMSINH DESAI UNIVERSITY
NADIAD-387001, GUJARAT

CERTIFICATE

This is to certify that the project entitled "**Foundation Management System**" is a bonafied report of the work carried out by

1) **Mr.YASH SUHAGIYA ,** Student ID No : **17ITUOS011**
2) **Mr.PARAS SUVAGIYA ,** Student ID No : **17ITUOS010**

of Department of Information Technology, semester V, under the guidance and supervision for the subject Database Management System. They were involved in Project training during academic year 2019-2020.

Prof. Roshni M. Raval
(Project Guide)
Department of Information Technology,
Faculty of Technology,
Dharmsinh Desai University, Nadiad
Date:

Prof. Vipul Dabhi
Head , Department of Information Technology,
Faculty of Technology,
Dharmsinh Desai University, Nadiad
Date:

ACKNOWLEDGEMENT

We would like to give our sincere acknowledgement to everybody responsible for the successful completion of our project “FOUNDATION MANAGEMENT SYSTEM”.

The success and final outcome of this project required a lot of guidance and assistance from many people and we are extremely privileged to have got this all along the completion of this project.

We owe our deep gratitude to our project guide Prof. Roshni M. Raval, who took been interest on our project work and guided us all along till the completion of our project work by providing all the necessary help for developing a good Database System.

We would also like to thank all our lecturers.

Finally we convoy our acknowledgement to all our friends and family members who directly or indirectly associated with us in the successful completion of the project.We thank one and all.

TABLE OF CONTENTS

1.SYSTEM OVERVIEW

1.1 CURRENT SYSTEM

Foundation Management System is engineered to effectively administer every aspect of an election from nominations through managing the Scholarship forms and Loan forms to processing final results.

Indians are becoming part of growing digital India. In India Foundation System involve manual scholarship using paper forms. Also counting are done through hands as well as registration of students are Manual. The Proposed system involve registration of users and Candidates through computers. Also contain computerize shopping of medicines & agriculture tools. The number of users obtained. It restricts duplication of students.

1.2 OBJECTIVES OF THE PROPOSED SYSTEM

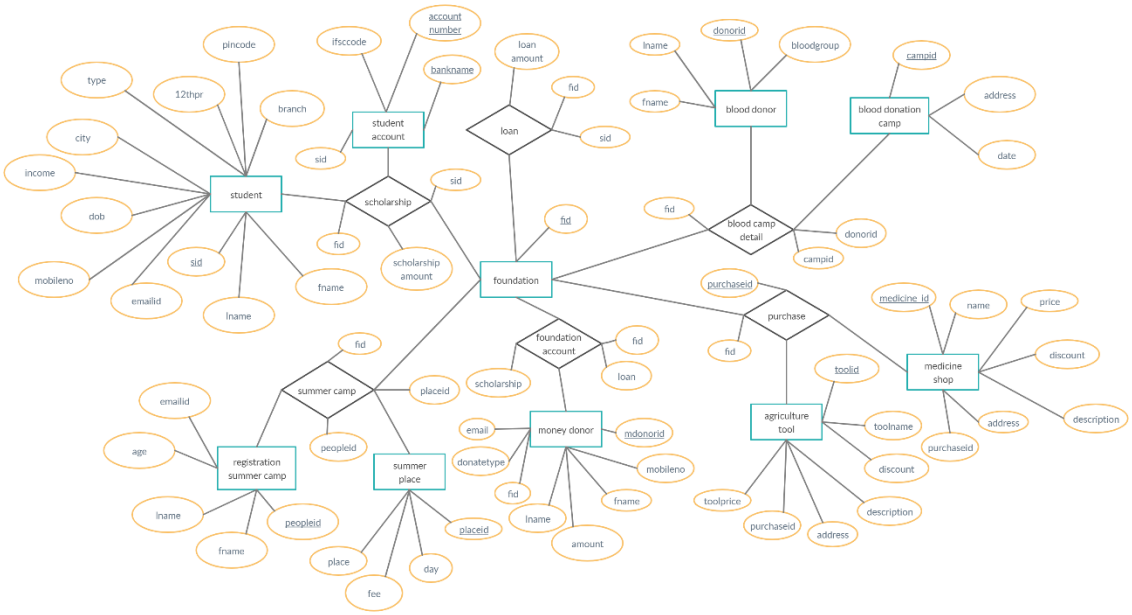
- Automate and facilitate process of managing the scholarship forms.
- Getting the whole data of students , Summer camp , Blood donation camp , Agriculture tool, People ,Money donor as well as Blood donor.
- Get Information like number of students getting their scholarship or loans, Summer Place Details, Address of blood donation camp, Discount on medicine and agriculture tool.

1.3 ADVANTAGES OF THE PROPOSED) SYSTEM

Through this system we try to achieve the below objectives

- To boost turnout of students, peoples and other donors.
- To inform users about their registration status.
- To allow users to change their vacation place.
- To counts scholarship or loan correctly.
- To stop the impersonation.
- To prevent users from geting Invalid scholarship.
- To Reduce cost, occur on management.
- Users Can fire the Complain to CEO of the Foundation to solve their isues directly.

2.E-R DIAGRAM


student
sid
fname
lname
emailid
mobileno
dob
income
city
type
12thpr
branch
plncode
student account
account number
ifsccode
bankname
scholarship
sid
fid
scholarship amount
loan
loan amount
fid
sid
foundation
fid
foundation account
fid
loan
scholarship
summer camp
fid
placeid
peopleid
registration summer camp
emailid
age
fname
lname
peopleid
summer place
placeid
place
day
fee
money donor
email
donatetype
mdonorid
mobileno
fid
fname
lname
amount
blood donor
donorid
bloodgroup
lname
fname
blood donation camp
campid
address
date
blood camp detail
fid
donorid
campid
purchase
purchaseid
fid
agriculture tool
toolid
toolname
toolprice
purchaseid
discount
description
address
medicine shop
medicine_id
name
price
discount
address
description
purchaseid

3.DATA DICTIONARY

3.1 Agriculture_Tool

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	tool_id	varchar(4)	latin1_swedish_ci		No	None		
2	toolname	varchar(20)	latin1_swedish_ci		No	None		
3	old_price	varchar(10)	latin1_swedish_ci		No	None		
4	discount	varchar(3)	latin1_swedish_ci		No	None		
5	new_price	varchar(10)	latin1_swedish_ci		No	None		
6	description	text	latin1_swedish_ci		No	None		
7	address	varchar(20)	latin1_swedish_ci		No	None		
8	purchare_id	varchar(4)	latin1_swedish_ci		No	None		
9	image	varchar(5000)	latin1_swedish_ci		No	None		

3.2 Blood_Camp_Detail

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	donorid	varchar(4)	latin1_swedish_ci		No	None		
2	campid	varchar(4)	latin1_swedish_ci		No	None		
3	fid	varchar(4)	latin1_swedish_ci		No	None		

3.3 Blood_Donation_Camp

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	campid	varchar(4)	latin1_swedish_ci		No	None		
2	address	varchar(40)	latin1_swedish_ci		No	None		
3	date	date			No	None		

3.4 Foundation

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	fid	varchar(4)	latin1_swedish_ci		No	None		

3.5 Blood_Donor

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	donorid	varchar(4)	latin1_swedish_ci		No	None		
2	fname	varchar(11)	latin1_swedish_ci		No	None		
3	lname	varchar(11)	latin1_swedish_ci		No	None		
4	bloodgroup	varchar(3)	latin1_swedish_ci		No	None		

3.6 Foundation_Account

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	fid	varchar(4)	latin1_swedish_ci		No	None		
2	scholarship	varchar(10)	latin1_swedish_ci		No	None		
3	loan	varchar(10)	latin1_swedish_ci		No	None		

3.7 Medicine_Shop

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	medicine_id	varchar(4)	latin1_swedish_ci		No	None		
2	name	varchar(50)	latin1_swedish_ci		No	None		
3	old_price	varchar(10)	latin1_swedish_ci		No	None		
4	discount	varchar(3)	latin1_swedish_ci		No	None		
5	new_price	varchar(10)	latin1_swedish_ci		No	None		
6	discription	text	latin1_swedish_ci		No	None		
7	address	varchar(30)	latin1_swedish_ci		No	None		
8	purchase_id	varchar(4)	latin1_swedish_ci		No	None		
9	image	varchar(5000)	latin1_swedish_ci		No	None		

3.8 Purchase

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	purchase_id	varchar(4)	latin1_swedish_ci		No	None		
2	fid	varchar(4)	latin1_swedish_ci		No	None		

3.9 Loan

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	sid	varchar(4)	latin1_swedish_ci		No	None		
2	fid	varchar(4)	latin1_swedish_ci		No	None		
3	loanamount	int(6)			No	None		

3.10 Scholarship

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	sid	varchar(4)	latin1_swedish_ci		No	None		
2	fid	varchar(4)	latin1_swedish_ci		No	None		
3	scholarshipamount	int(6)			No	None		

3.11 Money_Donor

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	mdonorid	varchar(4)	latin1_swedish_ci		No	None		
2	fname	varchar(11)	latin1_swedish_ci		No	None		
3	lname	varchar(11)	latin1_swedish_ci		No	None		
4	amount	varchar(10)	latin1_swedish_ci		No	None		
5	mobileno	varchar(10)	latin1_swedish_ci		No	None		
6	email	varchar(30)	latin1_swedish_ci		No	None		
7	donate_type	varchar(15)	latin1_swedish_ci		No	None		
8	fid	varchar(4)	latin1_swedish_ci		No	None		

3.12 Student_Account

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	sid	varchar(4)	latin1_swedish_ci		No	None		
2	accountno	varchar(11)	latin1_swedish_ci		No	None		
3	bankname	varchar(11)	latin1_swedish_ci		No	None		
4	ifsccode	varchar(11)	latin1_swedish_ci		No	None		

3.13 Student

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	**sid**	varchar(4)	latin1_swedish_ci		No	*None*		
2	**fname**	varchar(11)	latin1_swedish_ci		No	*None*		
3	**lname**	varchar(11)	latin1_swedish_ci		No	*None*		
4	**emailid**	varchar(30)	latin1_swedish_ci		No	*None*		
5	**mobileno**	varchar(10)	latin1_swedish_ci		No	*None*		
6	**dob**	date			No	*None*		
7	**income**	int(10)			No	*None*		
8	**city**	varchar(10)	latin1_swedish_ci		No	*None*		
9	**pincode**	int(6)			No	*None*		
10	**branch**	varchar(10)	latin1_swedish_ci		No	*None*		
11	**ptr**	float			No	*None*		
12	**type**	varchar(15)	latin1_swedish_ci		No	*None*		

3.14 Registration_SummerCamp

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	**peopleid**	varchar(4)	latin1_swedish_ci		No	*None*		
2	**fname**	varchar(11)	latin1_swedish_ci		No	*None*		
3	**lname**	varchar(11)	latin1_swedish_ci		No	*None*		
4	**age**	int(3)			No	*None*		
5	**emailid**	varchar(30)	latin1_swedish_ci		No	*None*		

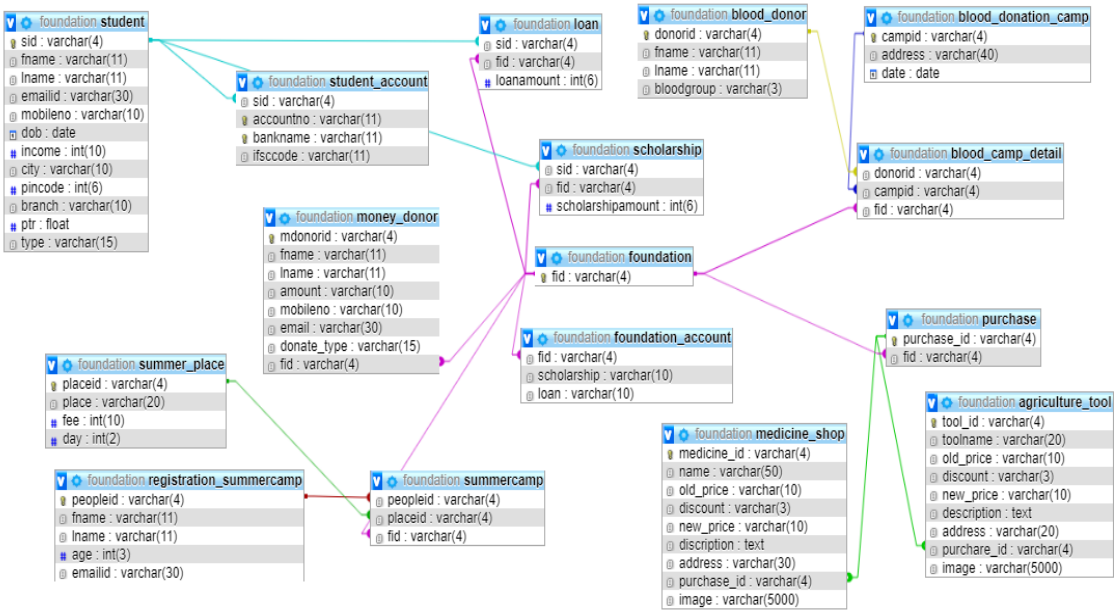
3.15 Summer_Place

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	**placeid**	varchar(4)	latin1_swedish_ci		No	*None*		
2	**place**	varchar(20)	latin1_swedish_ci		No	*None*		
3	**fee**	int(10)			No	*None*		
4	**day**	int(2)			No	*None*		

3.16 SummerCamp

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	**peopleid**	varchar(4)	latin1_swedish_ci		No	*None*		
2	**placeid**	varchar(4)	latin1_swedish_ci		No	*None*		
3	**fid**	varchar(4)	latin1_swedish_ci		No	*None*		

4.SCHEMA DIAGRAM



5.DATABASE IMPLEMENTATION

5.1 CREATE SCHEMA

5.1.1 Agriculture_Tool

```
CREATE TABLE `agriculture_tool` (
        `tool_id` varchar(4) PRIMARY KEY NOT NULL,
        `toolname` varchar(20) NOT NULL,
        `old_price` varchar(10) NOT NULL,
        `discount` varchar(3) NOT NULL,
        `new_price` varchar(10) NOT NULL,
        `description` text NOT NULL,
        `address` varchar(20) NOT NULL,
        `purchare_id` varchar(4) NOT NULL,
        `image` varchar(5000) NOT NULL,
        CONSTRAINT `agriculture_tool_ibfk_1` FOREIGN KEY (`purchare_id`)
        REFERENCES `purchase` (`purchase_id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5.1.2 Blood_Camp_Detail

```
CREATE TABLE `blood_camp_detail` (
        `donorid` varchar(4) NOT NULL,
        `campid` varchar(4) NOT NULL,
        `fid` varchar(4) NOT NULL,
        CONSTRAINT `blood_camp_detail_ibfk_1` FOREIGN KEY (`campid`)
        REFERENCES `blood_donation_camp` (`campid`),
        CONSTRAINT `blood_camp_detail_ibfk_2` FOREIGN KEY (`donorid`)
        REFERENCES `blood_donor` (`donorid`),
        CONSTRAINT `blood_camp_detail_ibfk_3` FOREIGN KEY (`fid`)
        REFERENCES `foundation` (`fid`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5.1.3 Blood_Donation_Camp

```
CREATE TABLE `blood_donation_camp` (
        `campid` varchar(4) PRIMARY KEY NOT NULL,
        `address` varchar(40) NOT NULL,
        `date` date NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5.1.4 Blood_Donor

```
CREATE TABLE `blood_donor` (
        `donorid` varchar (4) PRIMARY KEY NOT NULL,
        `fname` varchar (11) NOT NULL,
        `lname` varchar (11) NOT NULL,
        `bloodgroup` varchar (3) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5.1.5 Foundation

```
CREATE TABLE `foundation` (
        `fid` varchar (4) PRIMARY KEY NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5.1.6 Foundation_Account

```
CREATE TABLE `foundation_account` (
        `fid` varchar(4) NOT NULL,
        `scholarship` varchar(10) NOT NULL,
        `loan` varchar(10) NOT NULL,
        CONSTRAINT `foundation_account_ibfk_1` FOREIGN KEY (`fid`)
        REFERENCES `foundation` (`fid`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5.1.7 Loan

```
CREATE TABLE `loan` (
        `sid` varchar(4) NOT NULL,
        `fid` varchar(4) NOT NULL,
        `loanamount` int(6) NOT NULL,
        CONSTRAINT `loan_ibfk_1` FOREIGN KEY (`sid`) REFERENCES
        `student` (`sid`),
        CONSTRAINT `loan_ibfk_2` FOREIGN KEY (`fid`) REFERENCES
        `foundation` (`fid`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5.1.8 Purchase

```
CREATE TABLE `purchase` (
        `purchase_id` varchar(4) PRIMARY KEY NOT NULL,
        `fid` varchar(4) NOT NULL,
        CONSTRAINT `purchase_ibfk_1` FOREIGN KEY (`fid`)
        REFERENCES `foundation` (`fid`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5.1.9 Medicine_Shop

```
CREATE TABLE `medicine_shop` (
        `medicine_id` varchar(4) PRIMARY KEY NOT NULL,
        `name` varchar(50) NOT NULL,
        `old_price` varchar(10) NOT NULL,
        `discount` varchar(3) NOT NULL,
        `new_price` varchar(10) NOT NULL,
        `description` text NOT NULL,
        `address` varchar(30) NOT NULL,
        `purchase_id` varchar(4) NOT NULL,
        `image` varchar(5000) NOT NULL,
        CONSTRAINT `medicine_shop_ibfk_1` FOREIGN KEY (`purchase_id`)
        REFERENCES `purchase` (`purchase_id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5.1.10 Money_Donor

```
CREATE TABLE `money_donor` (
        `mdonorid` varchar(4) PRIMARY KEY NOT NULL,
        `fname` varchar(11) NOT NULL,
        `lname` varchar(11) NOT NULL,
        `amount` varchar(10) NOT NULL,
        `mobileno` varchar(10) NOT NULL,
        `email` varchar(30) NOT NULL,
        `donate_type` varchar(15) NOT NULL,
        `fid` varchar(4) NOT NULL,
        CONSTRAINT `money_donor_ibfk_1` FOREIGN KEY (`fid`)
        REFERENCES `foundation` (`fid`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5.1.11 Registration_SummerCamp

```
CREATE TABLE `registration_summercamp` (
        `peopleid` varchar(4) PRIMARY KEY NOT NULL,
        `fname` varchar(11) NOT NULL,
        `lname` varchar(11) NOT NULL,
        `age` int(3) NOT NULL,
        `emailid` varchar(30) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5.1.12 Scholarship

```
CREATE TABLE `scholarship` (
        `sid` varchar(4) NOT NULL,
        `fid` varchar(4) NOT NULL,
        `scholarshipamount` int(6) NOT NULL,
        CONSTRAINT `scholarship_ibfk_1` FOREIGN KEY (`sid`)
        REFERENCES `student` (`sid`),
        CONSTRAINT `scholarship_ibfk_2` FOREIGN KEY (`fid`)
        REFERENCES `foundation` (`fid`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5.1.13 Student

```
CREATE TABLE `student` (
        `sid` varchar(4) NOT NULL,
        `fname` varchar(11) NOT NULL,
        `lname` varchar(11) NOT NULL,
        `emailid` varchar(30) NOT NULL,
        `mobileno` varchar(10) NOT NULL,
        `dob` date NOT NULL,
        `income` int(10) NOT NULL,
        `city` varchar(10) NOT NULL,
        `pincode` int(6) NOT NULL,
        `branch` varchar(10) NOT NULL,
        `ptr` float NOT NULL,
        `type` varchar(15) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5.1.14 Student_Account

```
CREATE TABLE `student_account` (
        `sid` varchar(4) NOT NULL,
        `accountno` varchar(11) PRIMARY KEY NOT NULL,
        `bankname` varchar(11) PRIMARY KEY NOT NULL,
        `ifsccode` varchar(11) NOT NULL,
        CONSTRAINT `student_account_ibfk_1` FOREIGN KEY (`sid`)
        REFERENCES `student` (`sid`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5.1.15 SummerCamp

```
CREATE TABLE `summercamp` (
        `peopleid` varchar(4) NOT NULL,
        `placeid` varchar(4) NOT NULL,
        `fid` varchar(4) NOT NULL,
        CONSTRAINT `summercamp_ibfk_1` FOREIGN KEY (`peopleid`)
        REFERENCES `registration_summercamp` (`peopleid`),
        CONSTRAINT `summercamp_ibfk_2` FOREIGN KEY (`placeid`)
        REFERENCES `summer_place` (`placeid`),
        CONSTRAINT `summercamp_ibfk_3` FOREIGN KEY (`fid`)
        REFERENCES `foundation` (`fid`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5.1.16 Summer_Place

```
CREATE TABLE `summer_place` (
        `placeid` varchar(4) PRIMARY KEY NOT NULL,
        `place` varchar(20) NOT NULL,
        `fee` int(10) NOT NULL,
        `day` int(2) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5.2 INSERT DATA VALUE

5.2.1 Agriculture_Tool

INSERT INTO `agriculture_tool` (`tool_id`, `toolname`, `old_price`, `discount`, `new_price`, `description`, `address`, `purchare_id`, `image`) VALUES

('T01', 'Tractor', '220000', '10%', '200000', 'Tractors (one of which is pictured above) are available in sizes appropriate for farmers with 1 acre all the way up to those working 1,000 acres or more.', 'Krushi Bazar,Amreli', 'P01', '01.jpeg'),

('T02', 'Cultivator', '50000', '5%', '47500', 'Cultivators are used for—you probably already guessed this—soil cultivation. In particular, cultivators are used for weed control before planting into a bed, as well as incorporating crop or weed residues and preparing a seed bed.', 'Sardar Market,Surat', 'P03', '02.png'),

('T03', 'Cultipacker', '60000', '15%', '51000', 'Cultipackers are pulled behind tractors to firm seedbeds before seeding to set up your planting for good seed-to-soil contact. Following up broadcast seeding with a pass with the cultipacker will press the seeds into the soil.', 'Kheti Bazar Samiti,N', 'P09', '03.png'),

('T04', 'Plows', '30000', '10%', '270000', 'Moldboard plows: These are most often used on land that has not been in crop production before or has been fallow for a long time. The large wings of the plow are designed to cut into and turn over all of the soil in an area.', 'SK Road,Rajkot', 'P06', '04.png'),

('T05', 'Seed Drills', '55000', '5%', '50000', 'Seed drills are tractor attachments that insert seeds into the ground with minimal soil disturbance. They are most often used for row crops (such as grains), cover crops, and grasses or forage. There are no-till seed drills and traditional seed drills.', 'Krushi Bazar,Bhavana', 'P04', '05.png');

tool_id	toolname	old_price	discount	new_price	description	address	purchare_id	image
T01	Tractor	220000	10%	200000	Tractors (one of which is pictured above) are avai...	Krushi Bazar,Amreli	P01	01.jpeg
T02	Cultivator	50000	5%	47500	Cultivators are used for—you probably already gues...	Sardar Market,Surat	P03	02.png
T03	Cultipacker	60000	15%	51000	Cultipackers are pulled behind tractors to firm se...	Kheti Bazar Samiti,N	P09	03.png
T04	Plows	30000	10%	270000	Moldboard plows: These are most often used on land...	SK Road,Rajkot	P06	04.png
T05	Seed Drills	55000	5%	50000	Seed drills are tractor attachments that insert se...	Krushi Bazar,Bhavana	P04	05.png

5.2.2 Blood_Camp_Detail

INSERT INTO `blood_camp_detail` (`donorid`, `campid`, `fid`) VALUES

('D01', 'C02', 'F01'),('D02', 'C03', 'F01'),
('D03', 'C04', 'F01'),('D04', 'C05', 'F01'),
('D05', 'C06', 'F01'),('D06', 'C07', 'F01'),
('D07', 'C08', 'F01'),('D08', 'C09', 'F01'),
('D09', 'C10', 'F01'),('D10', 'C01', 'F01'),
('D10', 'C02', 'F01'),('D09', 'C03', 'F01'),
('D08', 'C04', 'F01'),('D07', 'C05', 'F01'),
('D06', 'C06', 'F01'),('D05', 'C07', 'F01'),
('D04', 'C08', 'F01'),('D03', 'C09', 'F01'),
('D02', 'C10', 'F01'),('D01', 'C01', 'F01');

donorid	campid	fid
D01	C02	F01
D02	C03	F01
D03	C04	F01
D04	C05	F01
D05	C06	F01
D06	C07	F01
D07	C08	F01
D08	C09	F01
D09	C10	F01
D10	C01	F01
D10	C02	F01
D09	C03	F01
D08	C04	F01
D07	C05	F01
D06	C06	F01
D05	C07	F01
D04	C08	F01
D03	C09	F01
D02	C10	F01
D01	C01	F01

5.2.3 Blood_Donation_Camp

INSERT INTO `blood_donation_camp` (`campid`, `address`, `date`) VALUES
('C01', 'Jakatnaka,Surat', '2019-11-01'),
('C02', 'College Road,Nadiad', '2019-11-04'),
('C03', 'Lalbaug,Vadodara', '2019-11-07'),
('C04', 'Umraj,Bharuch', '2019-11-11'),
('C05', 'SK Road,Rajkot', '2019-11-15'),

('C06', 'Kochrab Ashram,Ahmedabad', '2019-11-18'),
('C07', 'Banjara Party Plot,Godhara', '2019-11-20'),
('C08', 'Ghandhi Smruti Bhavan, Bhavanagar', '2019-11-22'),
('C09', 'Anand Nagar,Amreli', '2019-11-26'),
('C10', 'Raj Mahal Palace,Porbandar', '2019-11-30');

campid	address	date
C01	Jakatnaka,Surat	2019-11-01
C02	College Road,Nadiad	2019-11-04
C03	Lalbaug,Vadodara	2019-11-07
C04	Umraj,Bharuch	2019-11-11
C05	SK Road,Rajkot	2019-11-15
C06	Kochrab Ashram,Ahmedabad	2019-11-18
C07	Banjara Party Plot,Godhara	2019-11-20
C08	Ghandhi Smruti Bhavan, Bhavanagar	2019-11-22
C09	Anand Nagar,Amreli	2019-11-26
C10	Raj Mahal Palace,Porbandar	2019-11-30

5.2.4 Foundation_Account

INSERT INTO `foundation_account` (`fid`, `scholarship`, `loan`) VALUES
('F01', '100000', '100000');

fid	scholarship	loan
F01	100000	100000

5.2.5 Foundation

INSERT INTO `foundation` (`fid`) VALUES
('F01');

fid
F01

5.2.6 Blood_Donor

```
INSERT INTO `blood_donor` (`donorid`, `fname`, `lname`, `bloodgroup`) VALUES
('D01', 'Raj', 'Patel', 'A+'),('D02', 'Maitri', 'Maniya', 'B-'),
('D03', 'Paras', 'Suvagiya', 'A-'),('D04', 'Hardi', 'Patel', 'AB-'),
('D05', 'Pradip', 'Ghevariya', 'B-'),('D06', 'Vipul', 'Vyas', 'O-'),
('D07', 'Yashvi', 'Patel', 'A-'),('D08', 'Dhruvi', 'Korat', 'B+'),
('D09', 'Twiency', 'Korat', 'O+'),('D10', 'Raj', 'Kalathiya', 'AB+');
```

donorid	fname	lname	bloodgroup
D01	Raj	Patel	A+
D02	Maitri	Maniya	B-
D03	Paras	Suvagiya	A-
D04	Hardi	Patel	AB-
D05	Pradip	Ghevariya	B-
D06	Vipul	Vyas	O-
D07	Yashvi	Patel	A-
D08	Dhruvi	Korat	B+
D09	Twiency	Korat	O+
D10	Raj	Kalathiya	AB+

5.2.7 Loan

```
INSERT INTO `loan` (`sid`, `fid`, `loanamount`) VALUES
('S03', 'F01', 10000),
('S05', 'F01', 15000),
('S06', 'F01', 12500),
('S09', 'F01', 13000);
```

sid	fid	loanamount
S03	F01	10000
S05	F01	15000
S06	F01	12500
S09	F01	13000

5.2.8 Medicine_Shop

INSERT INTO `medicine_shop` (`medicine_id`, `name`, `old_price`, `discount`, `new_price`, `discription`, `address`, `purchase_id`, `image`) VALUES

('M01', 'Himalaya AyurSlim Capsule 60s', '220', '15%', '191.25', 'Obesity, Hyperlipidemia and craving for sugar.Two tablet twice daily after meals or as directed by your physician.', 'Kamrej,Surat', 'P05', 'm01.jpeg'),

('M02', 'Softovac Powder 100gm', '127', '0%', '127', 'SOFTOVAC is used to relieve constipation including constipation during pregnancy.It can also be used to help maintain normal bowel function if you suffer from haemorrhoids (piles).', 'Navrangpura,Ahmedabad', 'P09', 'm02.jpeg'),

('M03', 'SederOM Tablet', '330', '8%', '303.6', 'Helps in the formation of red blood and nerve cells.Facilitates aid in case of chronic blood loss or poor absorption', 'Vaniyavad,Nadiad', 'P07', 'm03.jpeg'),

('M04', 'Himalaya Liv.52 Tablet', '100', '10%', '90', 'Protects from alcohol-induced liver damage and arrests the progress of cirrhosis.Improves the appetite and promotes weight gain.', 'Hansot,Ankleshwar', 'P03', 'm04.jpeg'),

('M05', 'Dabur Pudin Hara Pearls', '30', '0%', '30', 'Dabur Pudin Hara Pearls contains herbal extracts known to naturally provide relief from various stomach related ailments like indigestion, stomachache, and gas.', 'Lalbaug,Vadodara', 'P06', 'm05.jpeg');

medicine_id	name	old_price	discount	new_price	discription	address	purchase_id	image
M01	Himalaya AyurSlim Capsule 60s	220	15%	191.25	Obesity, Hyperlipidemia and craving for sugar.Two ...	Kamrej,Surat	P05	m01.jpeg
M02	Softovac Powder 100gm	127	0%	127	SOFTOVAC is used to relieve constipation including...	Navrangpura,Ahmedabad	P09	m02.jpeg
M03	SederOM Tablet	330	8%	303.6	Helps in the formation of red blood and nerve cell...	Vaniyavad,Nadiad	P07	m03.jpeg
M04	Himalaya Liv.52 Tablet	100	10%	90	Protects from alcohol-induced liver damage and arr...	Hansot,Ankleshwar	P03	m04.jpeg
M05	Dabur Pudin Hara Pearls	30	0%	30	Dabur Pudin Hara Pearls contains herbal extracts k...	Lalbaug,Vadodara	P06	m05.jpeg

5.2.9 Money_Donor

INSERT INTO `money_donor` (`mdonorid`, `fname`, `lname`, `amount`, `mobileno`, `email`, `donate_type`, `fid`) VALUES
('D01', 'Rakesh', 'Singh', '15000', '9867543210', 'rakeshsingh@gmail.com', 'Scholarship', 'F01'),
('D02', 'Prakash', 'Joshi', '10000', '7896534210', 'prakashjoshi@gmail.com', 'Loan', 'F01'),

('D03', 'Rajesh', 'Patel', '18000', '6758493021', 'rajeshpatel@gmail.com', 'Scholarship', 'F01'),
('D04', 'Manan', 'Desai', '20000', '7890654321', 'manandesai@gmail.com', 'Scholarship', 'F01'),
('D05', 'Viraj', 'Ghelani', '5000', '8967452310', 'virajghelani@gmail.com', 'Loan', 'F01'),
('D06', 'Kabir', 'Singh', '35000', '7654321098', 'singhkabir@gmail.com', 'Scholarship', 'F01'),
('D07', 'Sapana', 'Kumar', '15500', '7098453218', 'sapanakumar@gmail.com', 'Loan', 'F01'),
('D08', 'Rakesh', 'Khunt', '25000', '6789054321', 'rakeshkhunt@gmail.com', 'Scholarship', 'F01'),
('D09', 'Paresh', 'Dhanani', '40000', '9867543211', 'pareshdhanani@gmail.com', 'Scholarship', 'F01'),
('D10', 'Roshan', 'Shah', '25500', '9867543214', 'roshanshah@gmail.com', 'Loan', 'F01');

mdonorid	fname	lname	amount	mobileno	email	donate_type	fid
D01	Rakesh	Singh	15000	9867543210	rakeshsingh@gmail.com	Scholarship	F01
D02	Prakash	Joshi	10000	7896534210	prakashjoshi@gmail.com	Loan	F01
D03	Rajesh	Patel	18000	6758493021	rajeshpatel@gmail.com	Scholarship	F01
D04	Manan	Desai	20000	7890654321	manandesai@gmail.com	Scholarship	F01
D05	Viraj	Ghelani	5000	8967452310	virajghelani@gmail.com	Loan	F01
D06	Kabir	Singh	35000	7654321098	singhkabir@gmail.com	Scholarship	F01
D07	Sapana	Kumar	15500	7098453218	sapanakumar@gmail.com	Loan	F01
D08	Rakesh	Khunt	25000	6789054321	rakeshkhunt@gmail.com	Scholarship	F01
D09	Paresh	Dhanani	40000	9867543211	pareshdhanani@gmail.com	Scholarship	F01
D10	Roshan	Shah	25500	9867543214	roshanshah@gmail.com	Loan	F01

5.2.10 Purchase

INSERT INTO `purchase` (`purchase_id`, `fid`) VALUES
('P01', 'F01'),
('P02', 'F01'),
('P03', 'F01'),
('P04', 'F01'),
('P05', 'F01'),
('P06', 'F01'),
('P07', 'F01'),
('P08', 'F01'),
('P09', 'F01'),
('P10', 'F01');

peopleid	placeid	fid
P10	P02	F01
P09	P03	F01
P07	P05	F01
P08	P06	F01
P05	P09	F01
P06	P01	F01
P04	P08	F01
P02	P04	F01
P03	P10	F01
P01	P07	F01

5.2.11 Registration_SummerCamp

INSERT INTO `registration_summercamp` (`peopleid`, `fname`, `lname`, `age`, `emailid`) VALUES
('P01', 'Rakesh', 'Roshan', 34, 'rakeshroshan@gmail.com'),
('P02', 'Kiran', 'Shah', 30, 'kiranshah@gmail.com'),
('P03', 'Keval', 'Joshi', 24, 'kevaljoshi@gmail.com'),
('P04', 'Tejas', 'Jaiswal', 29, 'tejasjaiswal@gmail.com'),
('P05', 'Palak', 'Patel', 45, 'palakpatel@gmail.com'),
('P06', 'Nandita', 'Raval', 50, 'nanditaraval@gmail.com'),
('P07', 'Yash', 'Patel', 39, 'yashpatel@gmail.com'),
('P08', 'Jaimin', 'Kumar', 18, 'jaiminkumar@gmail.com'),
('P09', 'Kartikey', 'Unagar', 20, 'kartikunagar@gmail.com'),
('P10', 'Dhruvi', 'Patel', 19, 'dhruvipatel@gmail.com');

peopleid	fname	lname	age	emailid
P01	Rakesh	Roshan	34	rakeshroshan@gmail.com
P02	Kiran	Shah	30	kiranshah@gmail.com
P03	Keval	Joshi	24	kevaljoshi@gmail.com
P04	Tejas	Jaiswal	29	tejasjaiswal@gmail.com
P05	Palak	Patel	45	palakpatel@gmail.com
P06	Nandita	Raval	50	nanditaraval@gmail.com
P07	Yash	Patel	39	yashpatel@gmail.com
P08	Jaimin	Kumar	18	jaiminkumar@gmail.com
P09	Kartikey	Unagar	20	kartikunagar@gmail.com
P10	Dhruvi	Patel	19	dhruvipatel@gmail.com

5.2.12 Scholarship

INSERT INTO `scholarship` (`sid`, `fid`, `scholarshipamount`) VALUES
('S01', 'F01', 7500),
('S02', 'F01', 5500),
('S04', 'F01', 8500),
('S07', 'F01', 8000),
('S08', 'F01', 9000),
('S10', 'F01', 6000);

sid	fid	scholarshipamount
S01	F01	7500
S02	F01	5500
S04	F01	8500
S07	F01	8000
S08	F01	9000
S10	F01	6000

5.2.13 SummerCamp

INSERT INTO `summercamp` (`peopleid`, `placeid`, `fid`) VALUES
('P10', 'P02', 'F01'),('P09', 'P03', 'F01'),
('P07', 'P05', 'F01'),('P08', 'P06', 'F01'),
('P05', 'P09', 'F01'),('P06', 'P01', 'F01'),
('P04', 'P08', 'F01'),('P02', 'P04', 'F01'),
('P03', 'P10', 'F01'),('P01', 'P07', 'F01');

peopleid	placeid	fid
P10	P02	F01
P09	P03	F01
P07	P05	F01
P08	P06	F01
P05	P09	F01
P06	P01	F01
P04	P08	F01
P02	P04	F01
P03	P10	F01
P01	P07	F01

5.2.14 Summer_Place

INSERT INTO `summer_place` (`placeid`, `place`, `fee`, `day`) VALUES
('P01', 'Mahabaleshwar', 7000, 3),('P02', 'Saurashtra', 6000, 5),

('P03', 'Meghalaya', 20000, 5),('P04', 'Aasam', 18000, 4),
('P05', 'Mumbai', 10000, 3),('P06', 'Delhi', 15000, 5),
('P07', 'Kedarnath', 10000, 4),('P08', 'Kerala', 25000, 7),
('P09', 'Shimla', 25000, 6),('P10', 'Ooty', 10000, 4);

placeid	place	fee	day
P01	Mahabaleshwar	7000	3
P02	Saurashtra	6000	5
P03	Meghalaya	20000	5
P04	Aasam	18000	4
P05	Mumbai	10000	3
P06	Delhi	15000	5
P07	Kedarnath	10000	4
P08	Kerala	25000	7
P09	Shimla	25000	6
P10	Ooty	10000	4

5.2.15 Student

INSERT INTO `student` (`sid`, `fname`, `lname`, `emailid`, `mobileno`, `dob`, `income`, `city`, `pincode`, `branch`, `ptr`, `type`) VALUES
('S01', 'Jenish', 'Vekariya', 'jenishvekr0031@gmail.com', '9365788787', '2001-08-13', 200000, 'Surat', 395006, 'Science', 94, 'Scholarship'),
('S02', 'Yash', 'Suhagiya', 'yashsuhagiya@gmail.com', '7878906543', '2000-07-13', 150000, 'Surat', 395006, 'Science', 93, 'Scholarship'),
('S03', 'Rishil', 'Bhalani', 'rishilbhalani98@gmail.com', '7689054321', '2000-07-14', 257000, 'Surat', 395006, 'Commerce', 84, 'Loan'),
('S04', 'Keval', 'Talaviya', 'kevaltalaviya67@gmail.com', '8980764532', '2000-07-01', 234000, 'Surat', 395006, 'Science', 93, 'Scholarship'),
('S05', 'Raj', 'Patel', 'raj5674@gmail.com', '9078653412', '2001-09-13', 150000, 'Vadodara', 396005, 'Commerce', 79, 'Loan'),
('S06', 'Dhruvang', 'Patel', 'dpc292299@gmail.com', '7890654321', '2000-02-29', 140000, 'Bharuch', 348002, 'Science', 80, 'Loan'),
('S07', 'Abhi', 'Thummar', 'abhithummar45@gmail.com', '6543217890', '2000-07-13', 200000, 'Surat', 395006, 'Commerce', 90, 'Scholarship'),
('S08', 'Pradip', 'Ghevariya', 'pradipghevariya41@gmail.com', '9786564320', '1999-11-05', 160000, 'Surat', 395006, 'Science', 93, 'Scholarship'),
('S09', 'Manan', 'Solanki', 'mnnsolnki@gmail.com', '9876543210', '1999-05-19', 190000, 'Nadiad', 387001, 'Commerce', 82, 'Loan'),

('S10', 'Milan', 'Soriya', 'milansoriya89@gmail.com', '8998985656', '2000-05-05', 300000, 'Rajkot', 345016, 'Commerce', 88, 'Scholarship');

sid	fname	lname	emailid	mobileno	dob	income	city	pincode	branch	ptr	type
S01	Jenish	Vekariya	jenishvekr0031@gmail.com	9365788787	2001-08-13	200000	Surat	395006	Science	94	Scholarship
S02	Yash	Suhagiya	yashsuhagiya@gmail.com	7878906543	2000-07-13	150000	Surat	395006	Science	93	Scholarship
S03	Rishil	Bhalani	rishilbhalani98@gmail.com	7689054321	2000-07-14	257000	Surat	395006	Commerce	84	Loan
S04	Keval	Talaviya	kevaltalaviya67@gmail.com	8980764532	2000-07-01	234000	Surat	395006	Science	93	Scholarship
S05	Raj	Patel	raj5674@gmail.com	9078653412	2001-09-13	150000	Vadodara	396005	Commerce	79	Loan
S06	Dhruvang	Patel	dpc292299@gmail.com	7890654321	2000-02-29	140000	Bharuch	348002	Science	80	Loan
S07	Abhi	Thummar	abhithummar45@gmail.com	6543217890	2000-07-13	200000	Surat	395006	Commerce	90	Scholarship
S08	Pradip	Ghevariya	pradipghevariya41@gmail.com	9786564320	1999-11-05	160000	Surat	395006	Science	93	Scholarship
S09	Manan	Solanki	mnnsolnki@gmail.com	9876543210	1999-05-19	190000	Nadiad	387001	Commerce	82	Loan
S10	Milan	Soriya	milansoriya89@gmail.com	8998985656	2000-05-05	300000	Rajkot	345016	Commerce	88	Scholarship

5.2.16 Student_Account

INSERT INTO `student_account` (`sid`, `accountno`, `bankname`, `ifsccode`) VALUES
('S03', '34781378541', 'KOTAK', 'KTKE013550'),
('S07', '43912764902', 'BOB', 'BOBN450345'),
('S10', '67582356719', 'SBI', 'SBIN011053'),
('S01', '67584930212', 'SBI', 'SBIN011050'),
('S04', '67584970982', 'SBI', 'SBIN014093'),
('S08', '74530915636', 'ICICI', 'ICICI9960'),
('S05', '76450914587', 'PNB', 'PNBN908742'),
('S02', '93710573829', 'CBI', 'CBIN054800'),
('S06', '94786234501', 'CBI', 'CBIN041060'),
('S09', '95674210567', 'HDFC', 'HDFC013534');

sid	accountno	bankname	ifsccode
S03	34781378541	KOTAK	KTKE013550
S07	43912764902	BOB	BOBN450345
S10	67582356719	SBI	SBIN011053
S01	67584930212	SBI	SBIN011050
S04	67584970982	SBI	SBIN014093
S08	74530915636	ICICI	ICICI9960
S05	76450914587	PNB	PNBN908742
S02	93710573829	CBI	CBIN054800
S06	94786234501	CBI	CBIN041060
S09	95674210567	HDFC	HDFC013534

5.3 QUERIES

5.3.1 Display All The Information Of Student Who Lives In Surat.

SELECT * FROM `student` WHERE `city`='Surat';

sid	fname	lname	emailid	mobileno	dob	income	city	pincode	branch	12thpr	type
S01	Jenish	Vekariya	jenishvekr0031@gmail.com	9365788787	2001-08-13	200000	Surat	395006	Science	94	Scholarship
S02	Yash	Suhagiya	yashsuhagiya@gmail.com	7878906543	2000-07-13	150000	Surat	395006	Science	93	Scholarship
S03	Rishil	Bhalani	rishilbhalani98@gmail.com	7689054321	2000-07-14	257000	Surat	395006	Commerce	87	Loan
S04	Keval	Talaviya	kevaltalaviya67@gmail.com	8980764532	2000-07-01	234000	Surat	395006	Science	93	Scholarship
S07	Abhi	Thummar	abhithummar45@gmail.com	6543217890	2000-07-13	200000	Surat	395006	Commerce	90	Scholarship
S08	Pradip	Ghevariya	pradipghevariya41@gmail.com	9786564320	1999-11-05	160000	Surat	395006	Science	93	Scholarship

5.3.2 Count Number Of Students In Each Bank And Display Along With Bank Name In Decreasing Order.

SELECT COUNT(`sid`) AS 'No. of Students',`bankname` FROM `student_account` GROUP BY `bankname` ORDER BY 'No. of Students' DESC;

No. of Students	bankname
3	SBI
2	CBI
1	KOTAK
1	PNB
1	BOB
1	ICICI
1	HDFC

5.3.3 Display Full Name , Age And Email ID Of Person Who Register For Summer Camp And Whose Summer Place Has More Than 5 Days.

SELECT `fname`, `lname`, `age`, `emailid` FROM registration_summercamp R INNER JOIN summercamp S INNER JOIN summer_place P ON P.placeid=S.placeid AND R.peopleid=S.peopleid WHERE P.day>5;

fname	lname	age	emailid
Kiran	Shah	30	kiranshah@gmail.com
Keval	Joshi	24	kevaljoshi@gmail.com
Palak	Patel	45	palakpatel@gmail.com
Dhruvi	Patel	19	dhruvipatel@gmail.com

5.3.4 Display Tool Name , New Price , Description And Address Of Agriculture Tool Whose New Price Is In Between 40000 And 60000.

SELECT `toolname`, `new_price`, `description`, `address` FROM `agriculture_tool` WHERE `new_price` BETWEEN 40000 AND 60000;

toolname	new_price	description	address
Cultivator	47500	Cultivators are used for—you probably already gues...	Sardar Market,Surat
Cultipacker	51000	Cultipackers are pulled behind tractors to firm se...	Kheti Bazar Samiti,N
Seed Drills	50000	Seed drills are tractor attachments that insert se...	Krushi Bazar,Bhavana

5.3.5 Display Full Name , Mobile No And Income Of Students Who Are Getting Scholarship Amount More Than 6000.

SELECT `fname`, `lname`,`mobileno`,`income` FROM `student` S WHERE EXISTS (SELECT P.sid FROM scholarship P WHERE P.sid=S.sid AND P.scholarshipamount>6000);

fname	lname	mobileno	income
Jenish	Vekariya	9365788787	200000
Keval	Talaviya	8980764532	234000
Abhi	Thummar	6543217890	200000
Pradip	Ghevariya	9786564320	160000

5.3.6 Display All The Information Of Money Donor Whose Name Starts With R.

SELECT * FROM `money_donor` WHERE `fname` LIKE 'R%';

mdonorid	fname	lname	amount	mobileno	email	donate_type	fid
D01	Rakesh	Singh	15000	9867543210	rakeshsingh@gmail.com	Scholarship	F01
D03	Rajesh	Patel	18000	6758493021	rajeshpatel@gmail.com	Scholarship	F01
D08	Rakesh	Khunt	25000	6789054321	rakeshkhunt@gmail.com	Scholarship	F01
D10	Roshan	Shah	25500	9867543214	roshanshah@gmail.com	Loan	F01

5.3.7 Count Number Of Blood Donor Who Have Same Blood Group And Display It Where Count Is Greater Than Or Equal To 2.

SELECT COUNT(donorid) AS 'No. Of Donors',bloodgroup FROM `blood_donor` GROUP BY bloodgroup HAVING COUNT(donorid)>=2;

No. Of Donors	bloodgroup
2	A-
2	B-

5.3.8 Display Student ID , Nane And Scholarship Amount Of Three Students Who Are Getting Maximum Scholarship.

SELECT student.sid , student.fname , student.lname , scholarship.scholarshipamount FROM `scholarship` LEFT JOIN student ON scholarship.sid=student.sid ORDER BY scholarshipamount DESC LIMIT 3;

sid	fname	lname	scholarshipamount ▼ 1
S08	Pradip	Ghevariya	9000
S04	Keval	Talaviya	8500
S07	Abhi	Thummar	8000

5.3.9 Display Information About Student Whose Family Income Is Greater Than 150000.

SELECT `fname`,`lname`,`mobileno`,`income`,`12thpr` FROM student WHERE income IN (SELECT income FROM student WHERE income>150000);

fname	lname	mobileno	income	12thpr
Jenish	Vekariya	9365788787	200000	94
Rishil	Bhalani	7689054321	257000	87
Keval	Talaviya	8980764532	234000	93
Abhi	Thummar	6543217890	200000	90
Pradip	Ghevariya	9786564320	160000	93
Manan	Solanki	9876543210	190000	91
Milan	Soriya	8998985656	300000	88

5.4 PROCEDURES AND EXCEPTION HANDLING

5.4.1 This "New_Student" Procedure Works Like Insert Query But If Ptr Is Greater Than 85 Then It Adds To Scholarship Otherwise To Loan And Insert Into Student Account Table Also.

```
DELIMITER $$
CREATE DEFINER=`root`@`localhost` PROCEDURE `New_Student`(IN `Fname`
VARCHAR(11), IN `Lname` VARCHAR(11), IN `Emailid` VARCHAR(30), IN
`Mobileno` VARCHAR(10), IN `Dob` DATE, IN `Income` INT(10), IN `City`
VARCHAR(10), IN `Pincode` INT(6), IN `Branch` VARCHAR(10), IN `Ptr` FLOAT,
IN `AccountNo` VARCHAR(11), IN `BankName` VARCHAR(11), IN `IFSC`
VARCHAR(12))
BEGIN
        DECLARE CNT varchar(4);

        SELECT COUNT(sid) INTO CNT FROM student;
        SET CNT=CNT+1;
        IF Ptr>85 THEN
                INSERT INTO student(sid ,
                fname,lname,emailid,mobileno,dob,income,city,pincode,branch,ptr,type)
                VALUES(concat('S',CNT),Fname,Lname,Emailid,Mobileno,Dob,Income,
                City,Pincode,Branch,Ptr,'Scholarship');
        ELSE
                INSERT INTO student(sid ,
                fname,lname,emailid,mobileno,dob,income,city,pincode,branch,ptr,type)
                VALUES(concat('S',CNT),Fname,Lname,Emailid,Mobileno,Dob,Income,
                City,Pincode,Branch,Ptr,'Loan');

        END IF;
        INSERT INTO student_account(sid,accountno,bankname,ifsccode)
        VALUES(concat('S',CNT),AccountNo,BankName,IFSC);
        SELECT 'Your Data Added Successfully To The Table.' AS MESSAGE;
END$$
DELIMITER ;
```

- Output Message

MESSAGE
Your Data Added Successfully To The Table.

- Student Table

S07	Abhi	Thummar	abhithummar45@gmail.com	6543217890	2000-07-13	200000	Surat	395006	Commerce	90	Scholarship
S08	Pradip	Ghevariya	pradipghevariya41@gmail.com	9786564320	1999-11-05	160000	Surat	395006	Science	93	Scholarship
S09	Manan	Solanki	mnnsolnki@gmail.com	9876543210	1999-05-19	190000	Nadiad	387001	Commerce	82	Loan
S10	Milan	Soriya	milansoriya89@gmail.com	8998985656	2000-05-05	300000	Rajkot	345016	Commerce	88	Scholarship
S11	Keyur	Patel	keyurpatel@gmail.com	8749256471	2000-03-12	140000	Mahesana	354008	Science	86	Scholarship

- Student Account Table

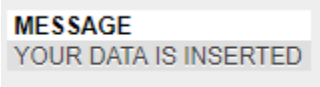
sid	accountno	bankname	ifsccode
S03	34781378541	KOTAK	KTKE013550
S11	38629516390	SBI	SBIN0034598
S07	43912764902	BOB	BOBN450345
S10	67592256719	SBI	SBIN011052

5.4.2 This "New_Money_Donor" Procedure Insert The Value Into Money Donor Table With Checking Email Id Is Valid Or Not.

```
DELIMITER $$
CREATE DEFINER=`root`@`localhost` PROCEDURE `New_Money_Donor`(IN
`Fname` VARCHAR(11), IN `Lname` VARCHAR(11), IN `Amount` VARCHAR(10),
IN `Mobileno` VARCHAR(10), IN `Emailid` VARCHAR(30), IN `DonateType`
VARCHAR(15))
BEGIN
        DECLARE CNT varchar(4);

        SELECT COUNT(mdonorid) INTO CNT FROM money_donor;
        SET CNT=CNT+1;
        IF Emailid LIKE '%@gmail.com' THEN
                INSERT INTO money_donor(mdonorid ,
                fname,lname,amount,mobileno,email,donate_type,fid)
                VALUES(concat('D',CNT),
                Fname,Lname,Amount,Mobileno,Emailid,DonateType,'F01');
        SELECT 'YOUR DATA IS INSERTED' AS MESSAGE;
        ELSE
                SELECT 'YOUR EMAIL ID IS NOT VALID' AS ERROR;
        END IF;
END$$
DELIMITER ;
```

Case 1 : If Email ID Is Valid.

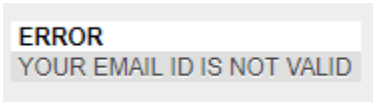
- Output Message



- Money Donor Table

D09	Paresh	Dhanani	40000	9867543211	pareshdhanani@gmail.com	Scholarship	F01
D10	Roshan	Shah	25500	9867543214	roshanshah@gmail.com	Loan	F01
D11	Prem	Chopra	20000	7592057249	chopraprem@gmail.com	Scholarship	F01

Case 2 : If Email ID is Not Valid.

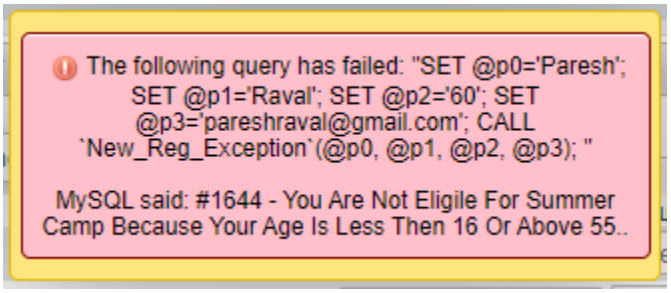


5.4.3 This "New_Reg_Exception" Generates An Exception When Registered People's Age Is Not Between 16 And 55 Otherwise Enter The Data Into Table.

```
DELIMITER $$
CREATE DEFINER=`root`@`localhost` PROCEDURE `New_Reg_Exception`(IN `Fname` VARCHAR(11), IN `Lname` VARCHAR(11), IN `Age` INT(3), IN `Emailid` VARCHAR(30))
BEGIN
	DECLARE CNT varchar(4);
	DECLARE age_exception CONDITION FOR SQLSTATE '22012';
	DECLARE EXIT HANDLER FOR age_exception
	BEGIN
		RESIGNAL SET MESSAGE_TEXT='You Are Not Eligile For Summer Camp Because Your Age Is Less Then 16 Or Above 55..';
	END;

	IF Age>16 AND Age<55 THEN
		IF Emailid LIKE '%@gmail.com' THEN
			SELECT 'YOU ARE ELIGIBLE' AS MESSAGE;
			SELECT COUNT(peopleid) INTO CNT FROM registration_summercamp;
			SET CNT=CNT+1;
```

```
                INSERT INTO
                registration_summercamp(peopleid,fname,lname,age,emailid)
                VALUES( concat('P',CNT),Fname,Lname,Age,Emailid);
        ELSE
                SELECT 'YOUR EMAIL ID IS NOT VALID' AS ERROR;
        END IF;
    ELSE
        SIGNAL age_exception;
    END IF;
END$$
DELIMITER ;
```

Case 1 : If Age Is Not Between 16 And 55.



Case 2 : If Email ID Is Not Valid.

ERROR
YOUR EMAIL ID IS NOT VALID

Case 3 : If Data Is Correct.

1. Output Message

MESSAGE
YOU ARE ELIGIBLE

2. Registration Table

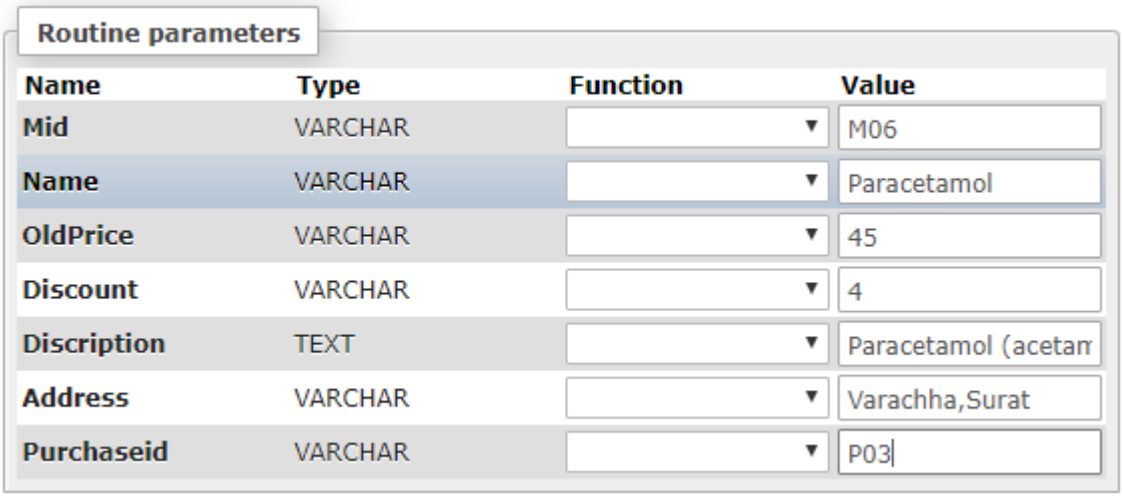
P09	Kartikey	Unagar	20	kartikunagar@gmail.com
P10	Dhruvi	Patel	19	dhruvipatel@gmail.com
P11	Paresh	Raval	54	pareshraval@gmail.com

5.5 FUNCTIONS

5.5.1 This "New_Medicine_Price" Function Is Calculate The New Price Of Entered Medicine With The Help Of Discount And Old Price.

```
DELIMITER $$
CREATE DEFINER=`root`@`localhost` FUNCTION `New_Medicine_Price`(`Mid`
VARCHAR(4), `Name` VARCHAR(50), `OldPrice` VARCHAR(10), `Discount`
VARCHAR(3), `Discription` TEXT, `Address` VARCHAR(30), `Purchaseid`
VARCHAR(4)) RETURNS varchar(10) CHARSET latin1
BEGIN
        DECLARE NewPrice varchar(10);
        DECLARE Price varchar(10);

        SET Price=((OldPrice*Discount)/100);
        SET NewPrice=OldPrice-Price;

        INSERT INTO medicine_shop
        (medicine_id,name,old_price,discount,new_price,discription,address,purchase_id)
        VALUES(Mid ,
        Name,OldPrice,Discount,NewPrice,Discription,Address,Purchaseid);

        RETURN NewPrice;
END$$
DELIMITER ;
```

- Input



- Output

New_Medicine_Price
43.2

- Medicine Shop Table

M04	Himalaya Liv.52 Tablet	100	10%	90	Protects from alcohol-induced liver damage and arr...	Hansot,Ankleshwar	P03
M05	Dabur Pudin Hara Pearls	30	0%	30	Dabur Pudin Hara Pearls contains herbal extracts k...	Lalbaug,Vadodara	P06
M06	Paracetamol	45	4	43.2	Paracetamol (acetaminophen) is a pain reliever and...	Varachha,Surat	P03

5.5.2 This "Loan_Amout" Function Gives The Loan Amount Information Of Given Student ID.

```
DELIMITER $$
CREATE DEFINER=`root`@`localhost` FUNCTION `Loan_Amount`(`Sid`
VARCHAR(4)) RETURNS int(6)
BEGIN

    DECLARE Amount int(6);

    SELECT L.loanamount INTO Amount FROM loan L INNER JOIN student S ON
    L.sid=S.sid WHERE S.sid=Sid;

    RETURN Amount;
END$$
DELIMITER ;
```

- Input

Routine parameters

Name	Type	Function	Value
Sid	VARCHAR		S06

- Output

Loan_Amount
12500

5.6 TRIGGERS

5.6.1 This "New_Student" Trigger Called After New Data Is Inserted Into Student Table.This Trigger Insert The Data Into Scholarship Or Loan Table And Update The Value In Foundation Account.

```
CREATE TRIGGER `New_Student` AFTER INSERT ON `student`
 FOR EACH ROW BEGIN
        DECLARE Amount int(10);
        DECLARE SAmount int(10);

        IF new.ptr > 85 THEN
                INSERT INTO scholarship(sid,fid,scholarshipamount)
                VALUES(new.sid,'F01',10000);
                SELECT scholarship INTO SAmount FROM foundation_account;
                SET Amount=SAmount-10000;
                UPDATE foundation_account SET scholarship=Amount;
        ELSE
                INSERT INTO loan(sid,fid,loanamount) VALUES(new.sid,'F01',15000);
                SELECT loan INTO SAmount FROM foundation_account;
                SET Amount=SAmount-15000;
                UPDATE foundation_account SET loan=Amount;
        END IF;
END
```

- Before Insert The Value

    1. Scholarship Table

sid	fid	scholarshipamount
S01	F01	7500
S02	F01	5500
S04	F01	8500
S07	F01	8000
S08	F01	9000
S10	F01	6000

    2. Foundation Table

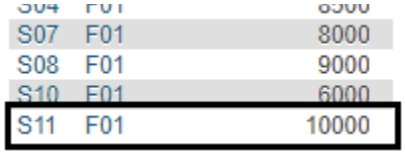
fid	scholarship	loan
F01	100000	100000

- Insert The Value In Student Table

```
INSERT INTO `student` (`sid`, `fname`, `lname`, `emailid`, `mobileno`, `dob`, `income`, `city`, `pincode`, `branch`, `ptr`, `type`) VALUES ('S11', 'Keyur', 'Patel', 'keyurpatel@gmail.com', '8749256471', '2000-03-12', 140000, 'Mahesana', 354008, 'Science', 86, 'Scholarship');
```

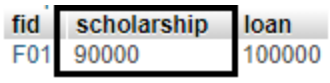
- After Insert The Value

1. Scholarship Table



S07	F01	8000
S08	F01	9000
S10	F01	6000
S11	F01	10000

2. Foundation Account Table



fid	scholarship	loan
F01	90000	100000

5.6.2 This "New_Donor" Trigger Called After Insert The Value Into Money Donor Table. This Trigger Update The Value Of Foundation Account As The Donor Donate The Money For Scholarship Or Loan.

```
CREATE TRIGGER `New_Donor` AFTER INSERT ON `money_donor`
 FOR EACH ROW BEGIN
        DECLARE NewAmount varchar(10);
        DECLARE Amount varchar(10);

        IF new.donate_type='Scholarship' THEN
                SELECT scholarship INTO Amount FROM foundation_account;
                SET NewAmount=Amount+new.amount;
                UPDATE foundation_account SET scholarship=NewAmount;
        ELSEIF new.donate_type='Loan' THEN
                SELECT loan INTO Amount FROM foundation_account;
                SET NewAmount=Amount+new.amount;
                UPDATE foundation_account SET loan=NewAmount;
        END IF;
END
```

- Before Insert The Value

fid	scholarship	loan
F01	100000	100000

- Insert The Value In Money Donor Table

```
INSERT INTO `money_donor`(`mdonorid`, `fname`, `lname`, `amount`, `mobileno`, `email`, `donate_type`, `fid`) VALUES ('D12','Milan','Soriya',30000,8794635218,'milansoriya@gmail.com','Loan','F01');
```

- After Insert The Value

fid	scholarship	loan
F01	90000	130000

5.7 CURSORS

5.7.1 This "SummerPlace" Cursor Gives The Information About People Full Name And Where They Are Going For Summer Vacation.

```
DELIMITER $$
CREATE DEFINER=`root`@`localhost` PROCEDURE `SummerPlace`()
BEGIN
	DECLARE Fname varchar(11);
	DECLARE Lname varchar(11);
	DECLARE Place varchar(20);
	DECLARE finished int(2);

	DECLARE c2 CURSOR FOR SELECT R.fname,R.lname,SP.place FROM
	registration_summercamp R INNER JOIN summercamp SC ON
	R.peopleid=SC.peopleid INNER JOIN summer_place SP ON
	SC.placeid=SP.placeid;

	DECLARE CONTINUE HANDLER FOR NOT FOUND SET finished=1;

	OPEN c2;

	gets: LOOP

		FETCH c2 INTO Fname,Lname,Place;
		IF finished=1 THEN
			LEAVE gets;

		END IF;

		SELECT Fname,Lname,Place AS 'Summer Camp Place';

	END LOOP gets;

	CLOSE c2;

END$$
DELIMITER ;
```

- Output

Fname	Lname	Summer Camp Place
Dhruvi	Patel	Saurashtra

Fname	Lname	Summer Camp Place
Kartikey	Unagar	Meghalaya

Fname	Lname	Summer Camp Place
Yash	Patel	Mumbai

Fname	Lname	Summer Camp Place
Jaimin	Kumar	Delhi

Fname	Lname	Summer Camp Place
Palak	Patel	Shimla

Fname	Lname	Summer Camp Place
Nandita	Raval	Mahabaleshwar

Fname	Lname	Summer Camp Place
Tejas	Jaiswal	Kerala

Fname	Lname	Summer Camp Place
Kiran	Shah	Aasam

Fname	Lname	Summer Camp Place
Keval	Joshi	Ooty

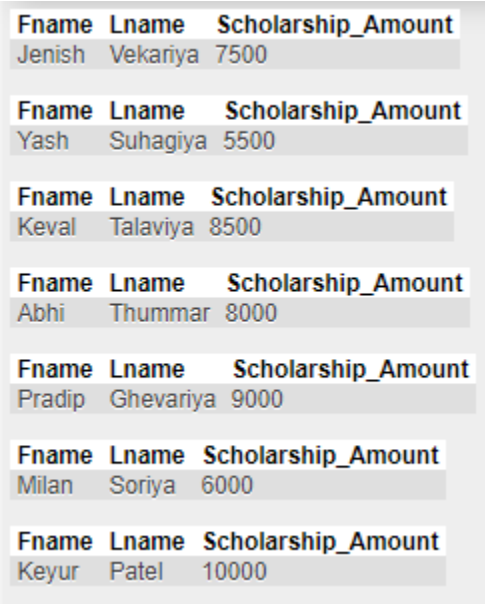
Fname	Lname	Summer Camp Place
Rakesh	Roshan	Kedarnath

5.7.2 This "Student_Scholarship" Cursor Gives The Information About Student Full Name And Their Allocated Scholarship Amount.

```
DELIMITER $$
CREATE DEFINER=`root`@`localhost` PROCEDURE `Student_Scholarship_Cursor`()
BEGIN
        DECLARE Fname varchar(11);
        DECLARE Lname varchar(11);
        DECLARE Amount int(6);
        DECLARE finished int(2);

        DECLARE c1 CURSOR FOR SELECT
        S.fname,S.lname,SP.scholarshipamount FROM student S INNER JOIN
        scholarship SP ON S.sid=SP.sid;
        DECLARE CONTINUE HANDLER FOR NOT FOUND SET finished=1;
```

```
        OPEN c1;
         get_detail: LOOP

         FETCH c1 INTO Fname,Lname,Amount;

         IF finished=1 THEN
               LEAVE get_detail;
         END IF;

         SELECT Fname,Lname,Amount AS Scholarship_Amount;

         END LOOP get_detail;
         CLOSE c1;
END$$
DELIMITER ;
```

- Output



Fname	Lname	Scholarship_Amount
Jenish	Vekariya	7500

Fname	Lname	Scholarship_Amount
Yash	Suhagiya	5500

Fname	Lname	Scholarship_Amount
Keval	Talaviya	8500

Fname	Lname	Scholarship_Amount
Abhi	Thummar	8000

Fname	Lname	Scholarship_Amount
Pradip	Ghevariya	9000

Fname	Lname	Scholarship_Amount
Milan	Soriya	6000

Fname	Lname	Scholarship_Amount
Keyur	Patel	10000

5.8 VIEWS

5.8.1 Information Of All The Account Whose Bank Name Is SBI.

CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`localhost` SQL SECURITY DEFINER VIEW `bank_details` AS select `student_account`.`sid` AS `sid`,`student_account`.`accountno` AS `accountno`,`student_account`.`bankname` AS `bankname`,`student_account`.`ifsccode` AS `ifsccode` from `student_account` where `student_account`.`bankname` = 'SBI' ;

sid	accountno	bankname	ifsccode
S11	38629516390	SBI	SBIN0034598
S10	67582356719	SBI	SBIN011053
S01	67584930212	SBI	SBIN011050
S04	67584970982	SBI	SBIN014093

5.8.2 Give The Information Of Blood Donation Camp Which Is Between Dates '2000-11-05' And '2000-11-20'.

CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`localhost` SQL SECURITY DEFINER VIEW `camp_details` AS select `blood_donation_camp`.`campid` AS `campid`,`blood_donation_camp`.`address` AS `address`,`blood_donation_camp`.`date` AS `date` from `blood_donation_camp` where `blood_donation_camp`.`date` between '2019-11-05' and '2019-11-20' ;

campid	address	date
C03	Lalbaug,Vadodara	2019-11-07
C04	Umraj,Bharuch	2019-11-11
C05	SK Road,Rajkot	2019-11-15
C06	Kochrab Ashram,Ahmedabad	2019-11-18
C07	Banjara Party Plot,Godhara	2019-11-20

6.FUTURE ENHANCEMENTS OF THE SYSTEM

- We will design Front-end Design in HTML , CSS , JavaScript and Develop Bank-end in Python.
- For security purpose New Registration is done using OTP.
- We will make database more consistent and We are making this database efficient and easy to implement with huge data capacity.
- Methods and user data input will be lot easy after the implement of GUI.
- We will also add some extra features so that the users can get answer for their complaints as fast as possible.

7.BIBLIOGRAPHY

- For the successful implementation of this project we referred to many websites and books.
- The schema was designed by taking ideas from website of election commission of india.
- We created the ER Diagram and Schema Diagram on "Creatly.com".
- Mostly we referred the online material for syntax of procedures, triggers, Exception and cursors.

Reference book:
Data Base System Concepts
-Henry F. Korth & A. Silberschatz 2nd Ed. McGraw-Hill 1991

Reference Websites:

- https://www.stackoverflow.com/
- https://www.w3school.com/
- https://www.tutorialspoint.com/
- http://www.mysqltutorial.org/