DBMS Project

1. Donor function
   create function insert_donor_details1(dname varchar(255), phno dec(10),city
   varchar(255),colony varchar(255),uname varchar(255),pwd varchar(255))
   -> returns int
   -> deterministic
   -> begin
   -> declare success int default 0;
   -> insert into donor(name,phno,colony,city,username,password)
   values(dname,phno,colony,city,uname,pwd);
   -> set success=1;
   -> return success;
   -> end//
   Query OK, 0 rows affected (0.13 sec)

2. NGO

   create function insert_ngo_details(nname varchar(255), phno dec(10),status
   varchar(255),task varchar(255),uname varchar(255),pwd varchar(255))
   returns int
   deterministic
   begin
   declare success int default 0;
   insert into
   ngo(name,phno,status,task,username,password)values(nname,phno,status,task,uname,pwd);
   set success=1;
   return success;
   end//

3. Donate function
   Create function donate(name varchar(255),qun int,did int) returns int deterministic
   Begin
   Declare foodid int;
   declare success int default 0;
   Insert into fooditem(foodtype,quantity) values(name,qun);
   Select food_id into foodid from fooditem order by food_id desc limit 1;
   Insert into donation(food_id,donor_id) values(foodid,did);
   Set success=1;
   Return success;
   End//

4. Donation History Procedure

```
Create procedure donation_history(did int)
Begin
Declare success int default 0;
select foodtype,foodname,quantity from fooditem where food_id in(select food_id from
donation where donor_id=did);
set success=1;
end//
```

5. Ngo_Donor Procedue
```
Create procedure view_donor(nid int)
Begin
Declare success int default 0;
select name,phno,colony,city from donor where donor_id in(select donor_id from donation
where food_id in(select food_id from receive where ngo_id =nid));
set success =1;
end//
```
6. NGO Receive food
```
Create function receive_food(fid int,nid int)
Returns int
Deterministic
Begin
Declare success int default 0;
Insert into receive(food_id,ngo_id) values(fid,nid);
Set success=1;
Return success;
End//
```

7. Creating view
```
create view donor_ngo_view as select ngo_id,name,phno,status,task from ngo;
```

8. Subquery:
9. Cursor

10. Procedure
   ***Admin Donor Details
```
Create procedure donor_details()
Begin
Select donor_id,name,phno,colony,city,username,password from donor;
End/
```

   ***Admin FoodItem details

Create procedure food_details()
Begin

```sql
Select food_id,foodtype,foodname,quantity from fooditem;
```

    End//

***** Admin NGO Details
Create procedure ngo_details()
Begin

```sql
select ngo_id,name,phno,status,task,username,password from NGO;
```

end//

****Admin Needy Details
Create procedure needy_details1()
Begin

```sql
select needy_id,name,phno,income,colony,city from Needy_People;
```

end//

11. Trigger

```sql
CREATE TRIGGER before_receive_insert
BEFORE INSERT ON Receive
FOR EACH ROW
BEGIN
  DECLARE food_exists INT;
  SELECT COUNT(*) INTO food_exists
  FROM FoodItem
  WHERE food_id = NEW.food_id;
  IF food_exists = 0 THEN
    SIGNAL SQLSTATE '45000'
    SET MESSAGE_TEXT = 'Cannot insert into Receive table: food_id does not exist in FoodItem table';
  END IF;
END
```

```sql
DELIMITER //

CREATE PROCEDURE read_data_using_cursor()
BEGIN
   DECLARE done INT DEFAULT FALSE;
Declare id int;
   DECLARE name_var VARCHAR(255);
   DECLARE phno_var VARCHAR(20);
```

```
            DECLARE city_var VARCHAR(100);
            DECLARE colony_var VARCHAR(100);

            -- Declare a cursor for the select statement
            DECLARE cur CURSOR FOR
                SELECT donor_id,name, phno, city, colony FROM donor;

            -- Declare continue handler to exit loop when no more rows found
            DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = TRUE;

            -- Open the cursor
            OPEN cur;

            -- Loop through the records and fetch data into variables
            read_loop: LOOP
                -- Fetch the next row into variables
                FETCH cur INTO id,name_var, phno_var, city_var, colony_var;

                -- Exit the loop if no more rows
                IF done THEN
                    LEAVE read_loop;
                END IF;

                -- Output the fetched data
                SELECT id,name_var, phno_var, city_var, colony_var;
            END LOOP;

            -- Close the cursor
            CLOSE cur;
        END //

        DELIMITER ;
```

DBMS Project Table Details

->create database food_db;

- Admin Table

```
mysql> CREATE TABLE Admin (
    ->    admin_id INT AUTO_INCREMENT PRIMARY KEY,
    -> name varchar(30),
```

```
   -> username varchar(30),
   -> password varchar(10));
Query OK, 0 rows affected (0.18 sec)

mysql> desc Admin;
+----------+-------------+------+-----+---------+----------------+
| Field    | Type        | Null | Key | Default | Extra          |
+----------+-------------+------+-----+---------+----------------+
| admin_id | int         | NO   | PRI | NULL    | auto_increment |
| name     | varchar(30) | YES  |     | NULL    |                |
| username | varchar(30) | YES  |     | NULL    |                |
| password | varchar(10) | YES  |     | NULL    |                |
+----------+-------------+------+-----+---------+----------------+
4 rows in set (0.10 sec)
```

➔   insert into Admin(name,username,password) values('Akshata','akshata','123');

- NGO

```
mysql> create table NGO(ngo_id int auto_increment primary key,name varchar(30),phno dec(10),status
varchar(255),task varchar(255),people int(5),username varchar(255),password varchar(10));
Query OK, 0 rows affected, 1 warning (0.06 sec)

mysql> desc NGO;
+----------+---------------+------+-----+---------+----------------+
| Field    | Type          | Null | Key | Default | Extra          |
+----------+---------------+------+-----+---------+----------------+
| ngo_id   | int           | NO   | PRI | NULL    | auto_increment |
| name     | varchar(30)   | YES  |     | NULL    |                |
| phno     | decimal(10,0) | YES  |     | NULL    |                |
| status   | varchar(255)  | YES  |     | NULL    |                |
| task     | varchar(255)  | YES  |     | NULL    |                |
| people   | int           | YES  |     | NULL    |                |
| username | varchar(255)  | YES  |     | NULL    |                |
| password | varchar(10)   | YES  |     | NULL    |                |
+----------+---------------+------+-----+---------+----------------+
8 rows in set (0.00 sec)
```

- insert into NGO(name,phno,status,task,people) values('Baya Karvel',9099234576,'Good','Providing education',100);

- Donor
  ```
  mysql> Create table Donor (donor_id  int NOT NULL AUTO_INCREMENT,name varchar(255),phno
  dec(10),password varchar(255),colony varchar(255),city varchar(255),username varchar(255),PRIMARY
  KEY (donor_id));
  Query OK, 0 rows affected (0.31 sec)
  ```

mysql> desc Donor;

```
+----------+--------------+------+-----+---------+----------------+
| Field    | Type         | Null | Key | Default | Extra          |
+----------+--------------+------+-----+---------+----------------+
| donor_id | int          | NO   | PRI | NULL    | auto_increment |
| name     | varchar(255) | YES  |     | NULL    |                |
| phno     | decimal(10,0)| YES  |     | NULL    |                |
| password | varchar(255) | YES  |     | NULL    |                |
| colony   | varchar(255) | YES  |     | NULL    |                |
| city     | varchar(255) | YES  |     | NULL    |                |
| username | varchar(255) | YES  |     | NULL    |                |
+----------+--------------+------+-----+---------+----------------+
```
7 rows in set (0.04 sec)

- Needy_People
  mysql> Create table Needy_People(needy_id int NOT NULL AUTO_INCREMENT,name varchar(255),username varchar(255), password varchar(255), phno dec(10) , colony varchar(255),city varchar(255),income float, PRIMARY KEY (needy_id));
  Query OK, 0 rows affected (0.06 sec)

  mysql> desc Needy_People;
```
+----------+--------------+------+-----+---------+----------------+
| Field    | Type         | Null | Key | Default | Extra          |
+----------+--------------+------+-----+---------+----------------+
| needy_id | int          | NO   | PRI | NULL    | auto_increment |
| name     | varchar(255) | YES  |     | NULL    |                |
| username | varchar(255) | YES  |     | NULL    |                |
| password | varchar(255) | YES  |     | NULL    |                |
| phno     | decimal(10,0)| YES  |     | NULL    |                |
| colony   | varchar(255) | YES  |     | NULL    |                |
| city     | varchar(255) | YES  |     | NULL    |                |
| income   | float        | YES  |     | NULL    |                |
+----------+--------------+------+-----+---------+----------------+
```
  8 rows in set (0.01 sec)

- Food Item

  mysql> Create table FoodItem(food_id int NOT NULL AUTO_INCREMENT,foodtype varchar(255),foodname varchar(255),quantity int, PRIMARY KEY (food_id));
  Query OK, 0 rows affected (0.05 sec)

  mysql> desc FoodItem;
```
+----------+--------------+------+-----+---------+----------------+
| Field    | Type         | Null | Key | Default | Extra          |
+----------+--------------+------+-----+---------+----------------+
| food_id  | int          | NO   | PRI | NULL    | auto_increment |
| foodtype | varchar(255) | YES  |     | NULL    |                |
```

```
| quantity | int         | YES |    | NULL    |                |
+----------+--------------+------+-----+---------+----------------+
3 rows in set (0.05 sec)
```

Relationship set

- Receive
  ```
  mysql> CREATE TABLE Receive (
      ->    receive_ID INT AUTO_INCREMENT PRIMARY KEY,
      ->    ngo_id INT,
      ->    food_id INT,
      ->    FOREIGN KEY (ngo_id) REFERENCES NGO(ngo_id),
      ->    FOREIGN KEY (food_id) REFERENCES FoodItem(food_id)
      -> );
  Query OK, 0 rows affected (0.09 sec)
  ```

- Donation
  ```
  mysql> CREATE TABLE Donation (
      ->    donation_id INT AUTO_INCREMENT PRIMARY KEY,
      ->    donor_id INT,
      ->    food_id INT,
      ->    FOREIGN KEY (donor_id) REFERENCES Donor(donor_ID),
      ->    FOREIGN KEY (food_id) REFERENCES FoodItem(food_ID)
      -> );
  Query OK, 0 rows affected (0.09 sec)
  ```

- Account

  ```
  CREATE TABLE Account (
     account_id INT AUTO_INCREMENT PRIMARY KEY,
     admin_id INT,
     ngo_id INT,
     donor_id INT,
     needy_id INT,
     FOREIGN KEY (admin_id) REFERENCES Admin(admin_id),
     FOREIGN KEY (ngo_id) REFERENCES NGO(ngo_id),
     FOREIGN KEY (donor_id) REFERENCES Donor(donor_id),
     FOREIGN KEY (needy_id) REFERENCES Needy_People(needy_id)
  );
  ```