INTERNET PROGRAMMING AND MOBILE PROGRAMMING CEF 440
TASK 6

DATABASE IMPLEMENTATION

Submitted By:

NAMES	**MATRICULE**
NDIKINTUM CARL NFON	FE20A073
NYENTY EYONG ARREHQUETTE	FE20A094
OBASIARREY M’ONEKE MARY ARREY-NJOK	FE20A095
OROCKTAKANG AGBORBEJA NTANGTANG	FE20A097
SALLE-NJUME MERYL EPOTE	FE20A102

Course Instructor
Dr Nkemeni Valery

TABLE OF CONTENT

Introduction

Purpose:

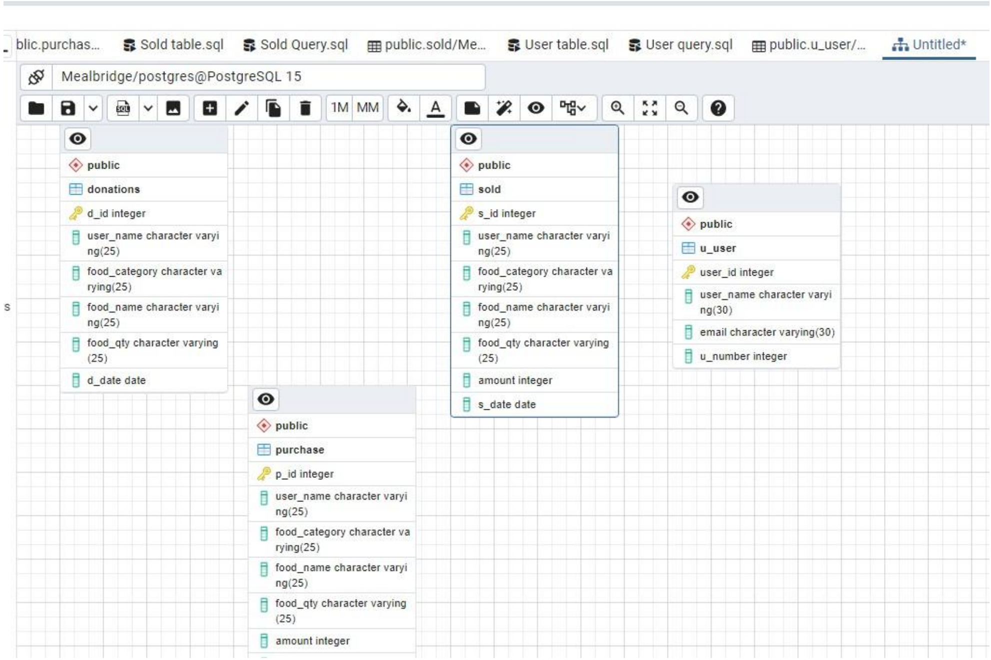
The purpose of this database is to support a food donation platform that allows users to donate food, sell food at discount prices, availability of food and request food.

Scope:

The database will store information related to users, food items and transactions.

Entity-Relationship Diagram (ERD)

The Entity-Relationship Diagram (ERD) is a visual representation of the entities (objects), relationships between entities, and attributes (properties) of entities in the food donation platform database.



public
donations
d_id integer
user_name character varying(25)
food_category character varying(25)
food_name character varying(25)
food_qty character varying (25)
d_date date

public
sold
s_id integer
user_name character varying(25)
food_category character varying(25)
food_name character varying(25)
food_qty character varying (25)
amount integer
s_date date

public
u_user
user_id integer
user_name character varying(30)
email character varying(30)
u_number integer

public
purchase
p_id integer
user_name character varying(25)
food_category character varying(25)
food_name character varying(25)
food_qty character varying (25)
amount integer

Physical Database Schema

a. User Table

```
CREATE TABLE U_user (
 User_id INT NOT NULL PRIMARY KEY,
  User_name VARCHAR(30) NOT NULL,
 Email VARCHAR  NOT NULL,
 U_number int NOT NULL
  );
```

b. Donations Table

```
CREATE TABLE Donations(
D_id INT NOT NULL, PRIMARY KEY
User_name VARCHAR(25),
food_category VARCHAR(25),
food_name VARCHAR(25) NOT NULL,
food_qty VARCHAR(25),
D_date date
);
```

c. Sold table

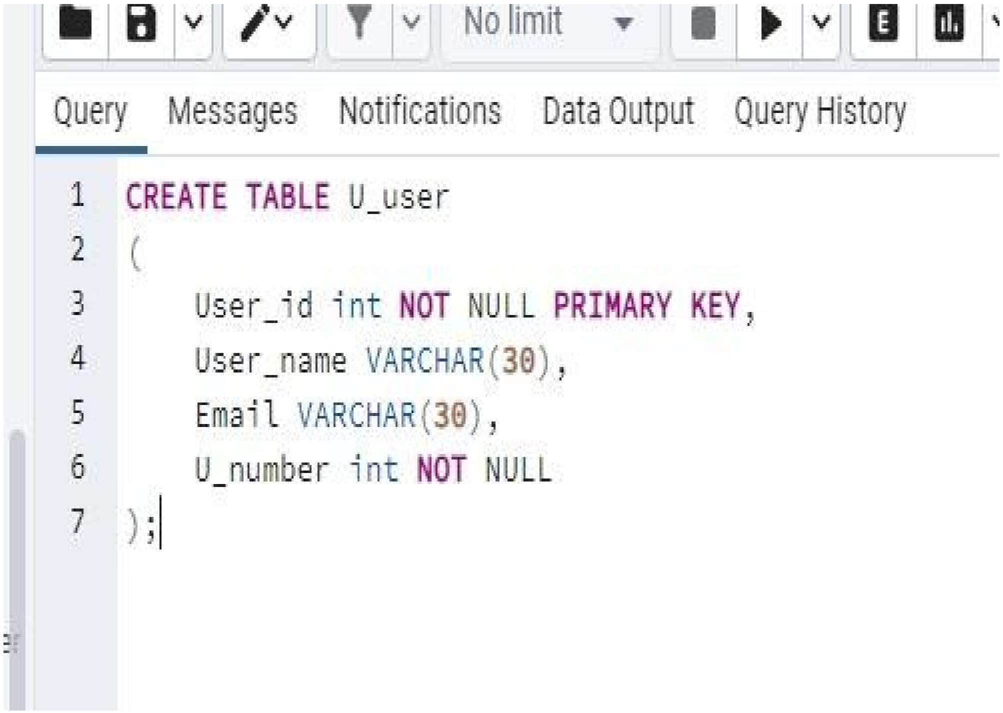
```
  CREATE TABLE Sold(
s_id INT NOT NULL PRIMARY KEY,
user_name VARCHAR(25),
food_category VARCHAR(25),
food_name VARCHAR(25),
food_qty VARCHAR(25),
Amount INT NOT NULL,
s_date DATE
);
```

Database Relationships

- One-to-Many:
  - Users to Food Items: One user can have multiple food items for donation or sale.
  - Users to Transactions: One user can have multiple transactions.

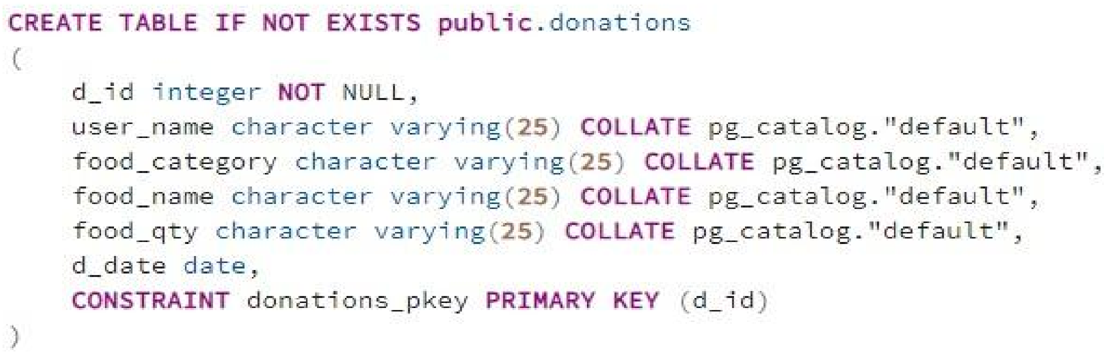
Queries

Query to create User Table



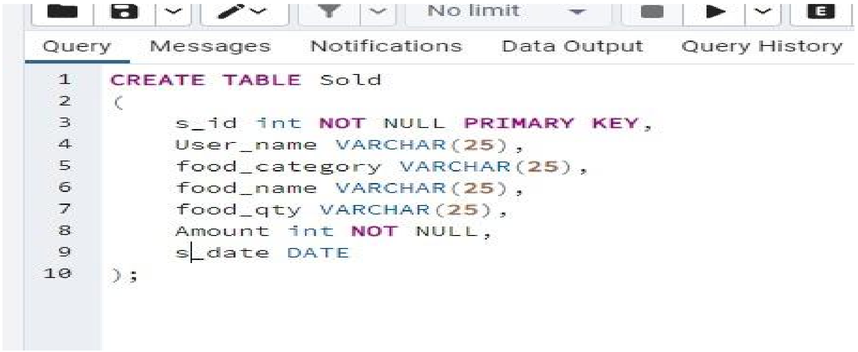
```
CREATE TABLE U_user
(
    User_id int NOT NULL PRIMARY KEY,
    User_name VARCHAR(30),
    Email VARCHAR(30),
    U_number int NOT NULL
);
```

Query to create Donations Table



```sql
CREATE TABLE IF NOT EXISTS public.donations
(
    d_id integer NOT NULL,
    user_name character varying(25) COLLATE pg_catalog."default",
    food_category character varying(25) COLLATE pg_catalog."default",
    food_name character varying(25) COLLATE pg_catalog."default",
    food_qty character varying(25) COLLATE pg_catalog."default",
    d_date date,
    CONSTRAINT donations_pkey PRIMARY KEY (d_id)
)
```

Query to create Sold table

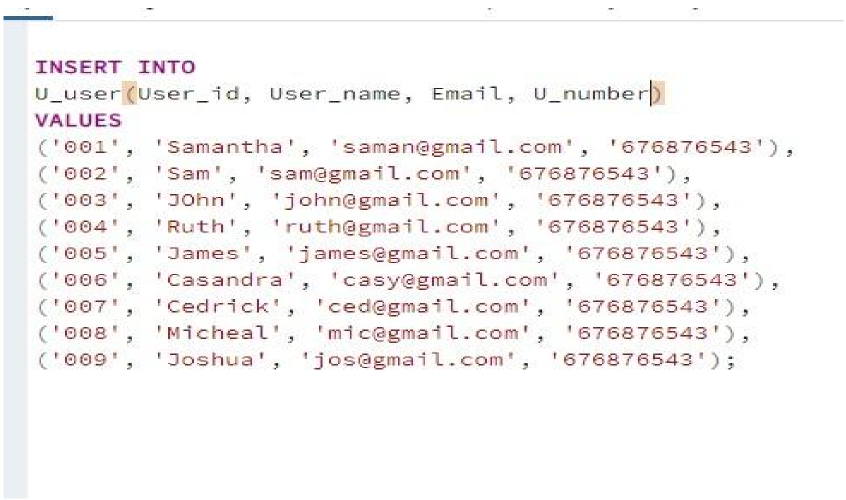


```sql
CREATE TABLE Sold
(
    s_id int NOT NULL PRIMARY KEY,
    User_name VARCHAR(25),
    food_category VARCHAR(25),
    food_name VARCHAR(25),
    food_qty VARCHAR(25),
    Amount int NOT NULL,
    s_date DATE
);
```

IMPORTATION OF DATA

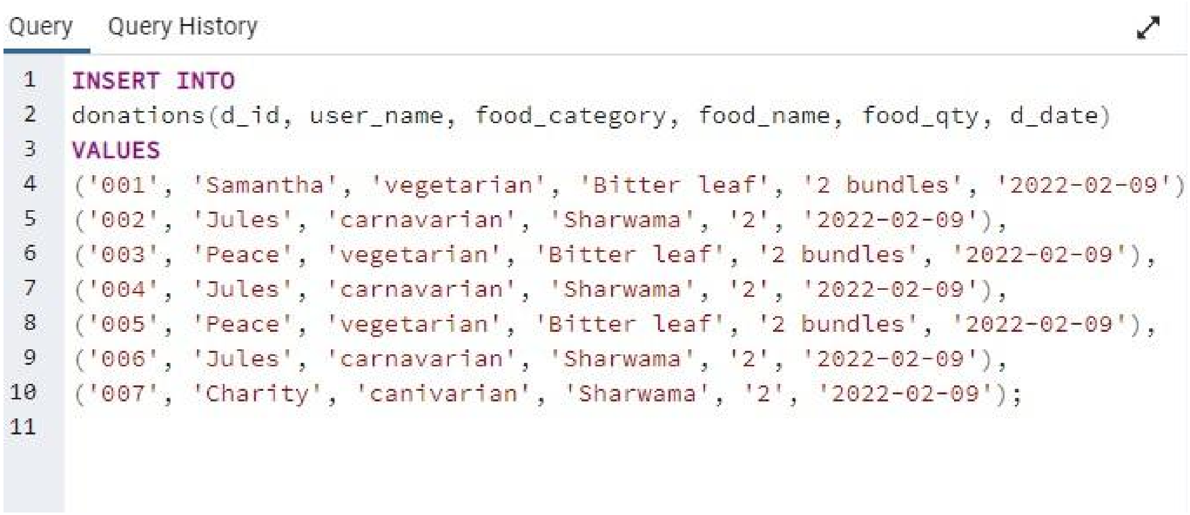
We used ‘MOCKAROO’ to import and generate data into the different tables

User table



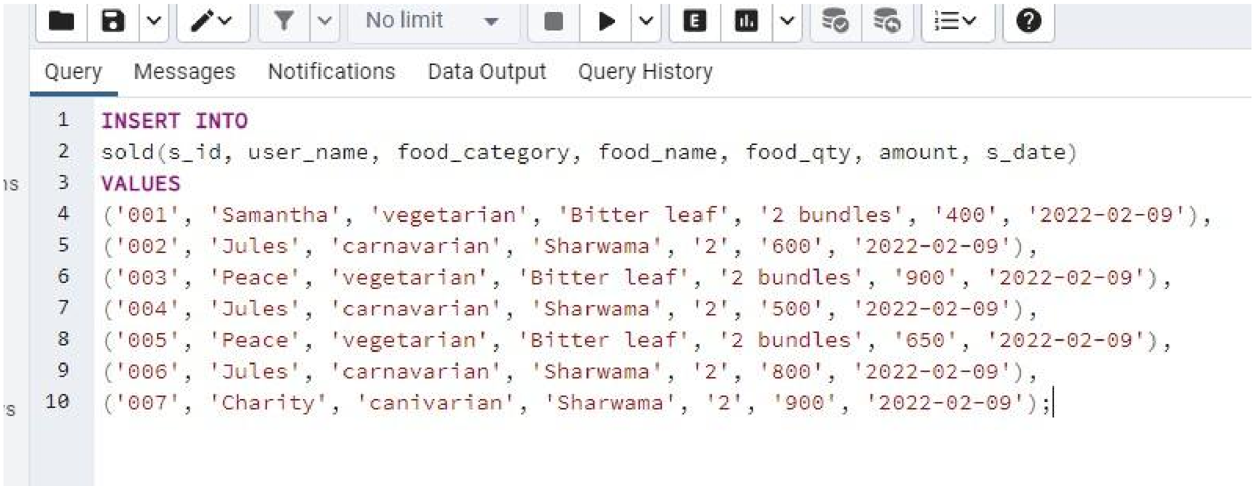
```
INSERT INTO
U_user(User_id, User_name, Email, U_number)
VALUES
('001', 'Samantha', 'saman@gmail.com', '676876543'),
('002', 'Sam', 'sam@gmail.com', '676876543'),
('003', 'JOhn', 'john@gmail.com', '676876543'),
('004', 'Ruth', 'ruth@gmail.com', '676876543'),
('005', 'James', 'james@gmail.com', '676876543'),
('006', 'Casandra', 'casy@gmail.com', '676876543'),
('007', 'Cedrick', 'ced@gmail.com', '676876543'),
('008', 'Micheal', 'mic@gmail.com', '676876543'),
('009', 'Joshua', 'jos@gmail.com', '676876543');
```

Donations Table



```
INSERT INTO
donations(d_id, user_name, food_category, food_name, food_qty, d_date)
VALUES
('001', 'Samantha', 'vegetarian', 'Bitter leaf', '2 bundles', '2022-02-09')
('002', 'Jules', 'carnavarian', 'Sharwama', '2', '2022-02-09'),
('003', 'Peace', 'vegetarian', 'Bitter leaf', '2 bundles', '2022-02-09'),
('004', 'Jules', 'carnavarian', 'Sharwama', '2', '2022-02-09'),
('005', 'Peace', 'vegetarian', 'Bitter leaf', '2 bundles', '2022-02-09'),
('006', 'Jules', 'carnavarian', 'Sharwama', '2', '2022-02-09'),
('007', 'Charity', 'canivarian', 'Sharwama', '2', '2022-02-09');
```

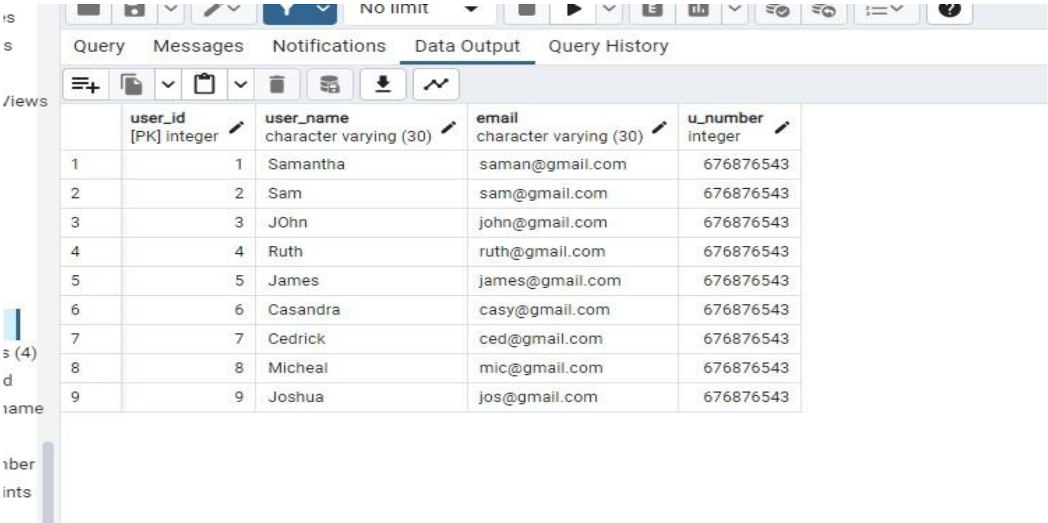
Sold Table



```sql
INSERT INTO
sold(s_id, user_name, food_category, food_name, food_qty, amount, s_date)
VALUES
('001', 'Samantha', 'vegetarian', 'Bitter leaf', '2 bundles', '400', '2022-02-09'),
('002', 'Jules', 'carnavarian', 'Sharwama', '2', '600', '2022-02-09'),
('003', 'Peace', 'vegetarian', 'Bitter leaf', '2 bundles', '900', '2022-02-09'),
('004', 'Jules', 'carnavarian', 'Sharwama', '2', '500', '2022-02-09'),
('005', 'Peace', 'vegetarian', 'Bitter leaf', '2 bundles', '650', '2022-02-09'),
('006', 'Jules', 'carnavarian', 'Sharwama', '2', '800', '2022-02-09'),
('007', 'Charity', 'canivarian', 'Sharwama', '2', '900', '2022-02-09');
```

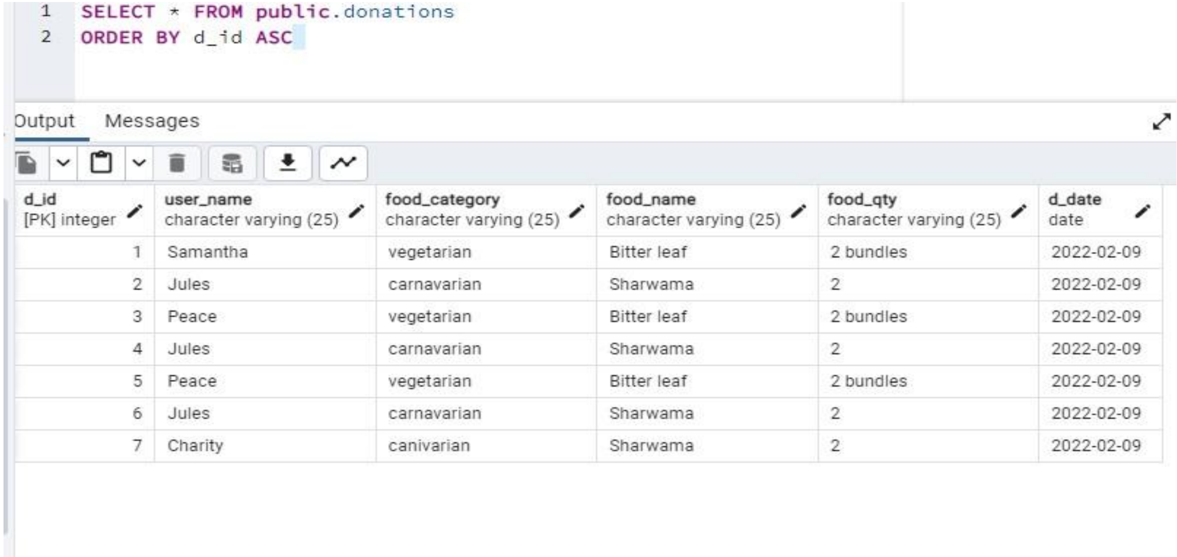
TABLES

User Table



	user_id [PK] integer	user_name character varying (30)	email character varying (30)	u_number integer
1	1	Samantha	saman@gmail.com	676876543
2	2	Sam	sam@gmail.com	676876543
3	3	JOhn	john@gmail.com	676876543
4	4	Ruth	ruth@gmail.com	676876543
5	5	James	james@gmail.com	676876543
6	6	Casandra	casy@gmail.com	676876543
7	7	Cedrick	ced@gmail.com	676876543
8	8	Micheal	mic@gmail.com	676876543
9	9	Joshua	jos@gmail.com	676876543

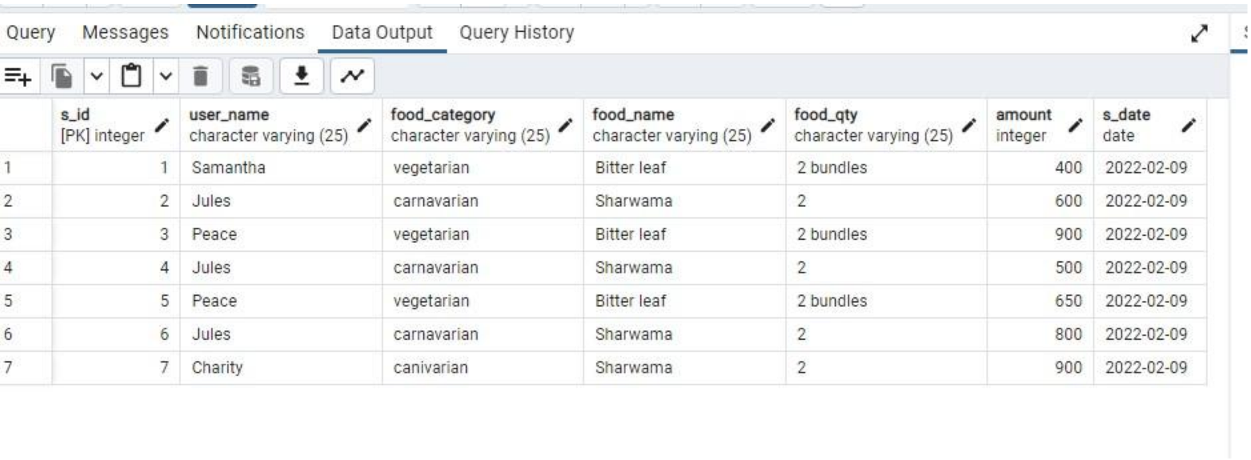
Donations Table



```
SELECT * FROM public.donations
ORDER BY d_id ASC
```

d_id [PK] integer	user_name character varying (25)	food_category character varying (25)	food_name character varying (25)	food_qty character varying (25)	d_date date
1	Samantha	vegetarian	Bitter leaf	2 bundles	2022-02-09
2	Jules	carnavarian	Sharwama	2	2022-02-09
3	Peace	vegetarian	Bitter leaf	2 bundles	2022-02-09
4	Jules	carnavarian	Sharwama	2	2022-02-09
5	Peace	vegetarian	Bitter leaf	2 bundles	2022-02-09
6	Jules	carnavarian	Sharwama	2	2022-02-09
7	Charity	canivarian	Sharwama	2	2022-02-09

Sold Table



	s_id [PK] integer	user_name character varying (25)	food_category character varying (25)	food_name character varying (25)	food_qty character varying (25)	amount integer	s_date date
1	1	Samantha	vegetarian	Bitter leaf	2 bundles	400	2022-02-09
2	2	Jules	carnavarian	Sharwama	2	600	2022-02-09
3	3	Peace	vegetarian	Bitter leaf	2 bundles	900	2022-02-09
4	4	Jules	carnavarian	Sharwama	2	500	2022-02-09
5	5	Peace	vegetarian	Bitter leaf	2 bundles	650	2022-02-09
6	6	Jules	carnavarian	Sharwama	2	800	2022-02-09
7	7	Charity	canivarian	Sharwama	2	900	2022-02-09

Conclusion

- The database design outlined above provides a foundation for building a food donation platform with essential features and functionalities. It allows users to interact with the system, donate or sell food items conveniently.