CHARITY DONATION MANAGEMENT SYSTEM REPORT

CMT 302: ADVANCED DATABASE SYSTEMS

GROUP 13

QUESTION 40

SANDRA NYALE- 1049087

JOY KAMAU- 1049490

ELVIS MURAGA- 1049387

NELLY MURUGI- 1050304

JOSHUA ONCHIRI- 1049474

TABLE OF CONTENTS

TABLE OF FIGURES

INTRODUCTION

OVERVIEW

The goal of this project is to create a system for managing charitable donations that will improve and expedite the process for non-profit organizations. In order to build confidence with contributors, the system seeks to automate reporting, maintain donor profiles, centralize gift records, and offer transparency. The solution will guarantee that charities can effectively track donations, communicate with contributors, and produce insights to aid in decision-making by integrating cutting-edge database approaches. By addressing common contribution management issues, this solution will increase organizational effectiveness and provide effective support for philanthropic causes.

RATIONALE

The requirement for precise data tracking and management in non-profit organizations is the justification for creating a system for managing charitable donations. Maintaining well-organized donation records is a struggle for many charities, particularly when their donor base expands and becomes more diverse. Data discrepancies, redundant information, and time-consuming procedures are frequently the result of manual record-keeping or the use of simple tools like spreadsheets. These issues might turn off potential donors and erode their confidence in the organization's transparency.

These problems can be resolved with a specialized database-driven system that streamlines report preparation, manages donor profiles, and automates donation tracking. Donors can check their donation history, get acknowledgement, and remain in touch with the causes they support thanks to this system's simplified, user-friendly interface. A transparent and well-run donation system is

OBJECTIVES

The objectives of this project are:

1. **To design and develop a database system that enables efficient tracking and management of donations:** The system will store all donation data in an organized, secure, and accessible manner, allowing easy retrieval of information when needed.

2. **To create a centralized donor management feature:** The system will manage detailed donor profiles, track donation histories, and enable effective donor segmentation for targeted communication.

3. **To facilitate transparency and accountability:** By providing clear records and transaction histories, the system will build donor trust through transparency, ensuring that donations are tracked accurately and that fund usage is communicated effectively.

4. **To automate report generation and analytics:** The system will generate reports on donation patterns, donor engagement, and fundraising outcomes, providing insights that can help organizations optimize their strategies.

5. **To enhance the user experience for donors and charity staff:** The system will offer intuitive interfaces that simplify donation tracking for charity staff and give donors access to their history and acknowledgment letters, improving overall engagement and satisfaction.

SYSTEM DESIGN

ENTITIES

Entities in the system include:

- **Donor**: Stores information about each donor.
- **Donation**: Tracks each donation transaction.
- **Charity/Event**: Represents the cause or event, a donation is made toward.
- **Receipt**: Records details of the transaction and acknowledges donations.
- **Staff**: Manages the system and donation processing.

RELATIONSHIPS

- **Donor–Donation**: A donor can make multiple donations.
- **Donation–Charity/Event**: Each donation is linked to a specific charity/event.
- **Donation–Receipt**: Each donation has a receipt generated.
- **Staff–Charity/Event**: Staff manage different events or charity functions.

ER DIAGRAMS:

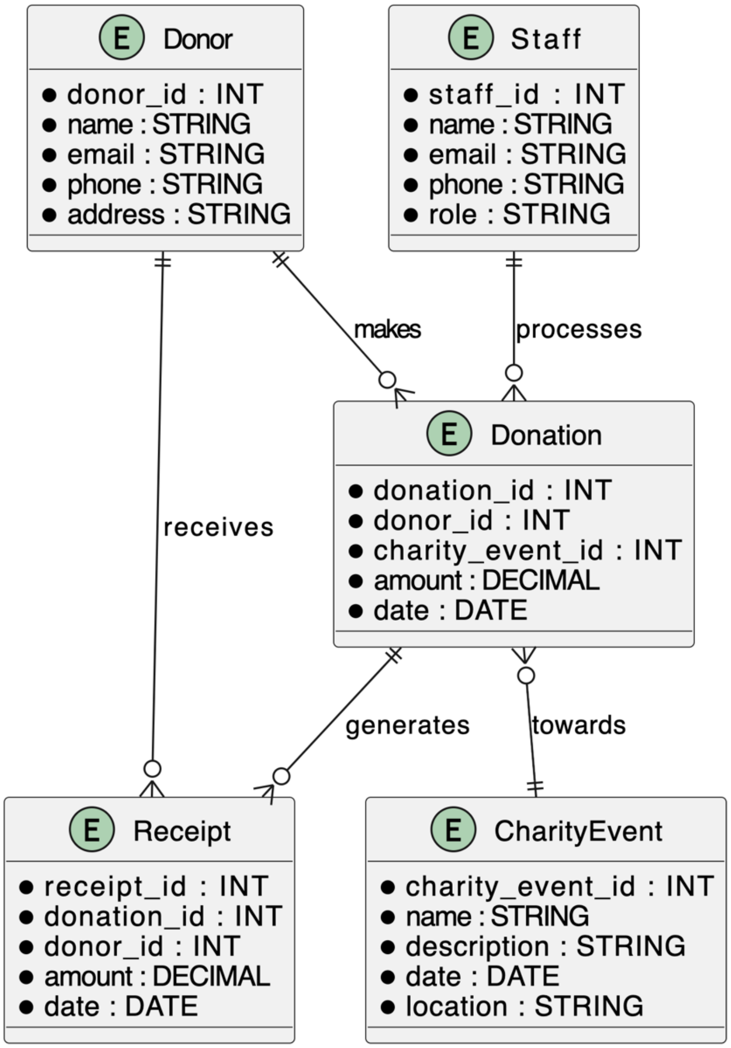


Figure 1

TABLE STRUCTURES

Define the structure for each table, including primary keys (PK), foreign keys (FK), and essential attributes:

1. Donor Table

- donor_id (PK, INT) — Unique identifier for each donor.
- name (VARCHAR) — Donor's name.
- email (VARCHAR) — Donor's email address.
- phone_number (VARCHAR) — Donor's contact number.
- address (VARCHAR) — Donor’s address.

2. **Donation Table**
   - donation_id (PK, INT) — Unique identifier for each donation.
   - donor_id (FK, INT) — References Donor table.
   - charity_id (FK, INT) — References Charity/Event table.
   - amount (DECIMAL) — Amount donated.
   - date (DATE) — Date of donation.
   - payment_method (VARCHAR) — Method used (e.g., credit card, bank transfer).
3. **Charity Table**
   - charity_id (PK, INT) — Unique identifier for each charity or event.
   - name (VARCHAR) — Name of the charity or event.
   - description (TEXT) — Description of the cause/event.
   - start_date (DATE) — Start date of the event.
   - end_date (DATE) — End date of the event.
4. **Receipt Table**
   - receipt_id (PK, INT) — Unique identifier for each receipt.
   - donation_id (FK, INT) — References Donation table.
   - issued_date (DATE) — Date the receipt was issued.
   - amount (DECIMAL) — Total donation amount on the receipt.
   - receipt_content (TEXT) — Receipt details or notes.

5. **Staff Table**
   - staff_id (PK, INT) — Unique identifier for each staff member.
   - name (VARCHAR) — Staff member's name.
   - role (VARCHAR) — Role or designation.
   - email (VARCHAR) — Staff member's email.
   - phone_number (VARCHAR) — Contact number.

6. Staff_Event Table

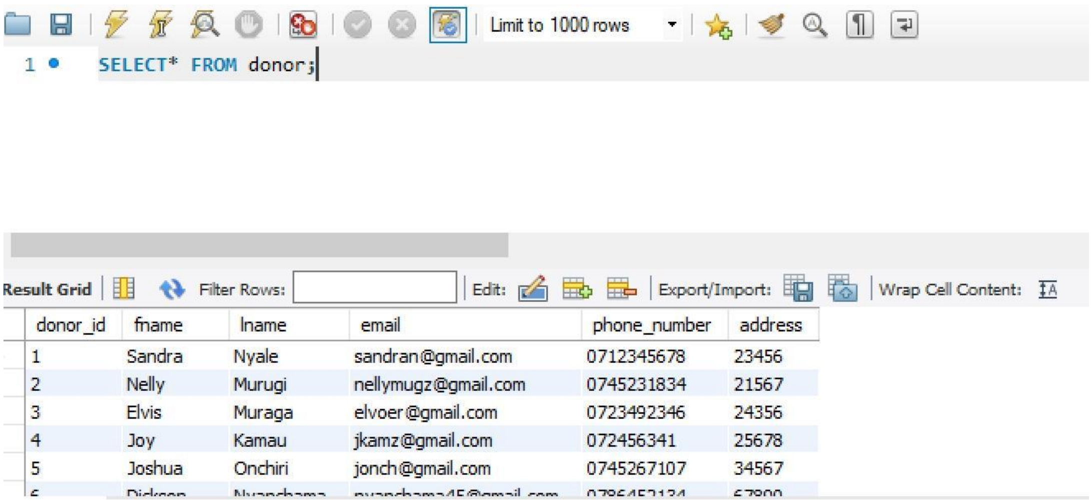
- staff_id (FK, INT) — References Staff table.
- charity_id (FK, INT) — References Charity/Event table.

SQL SCHEMA:

SQL schema to create these tables in a database:

a) Donor Table

```
CREATE TABLE Donor (
   donor_id INT PRIMARY KEY,
   name VARCHAR(100),
   email VARCHAR(100) UNIQUE,
   phone_number VARCHAR(15),
   address VARCHAR(255)
);
```



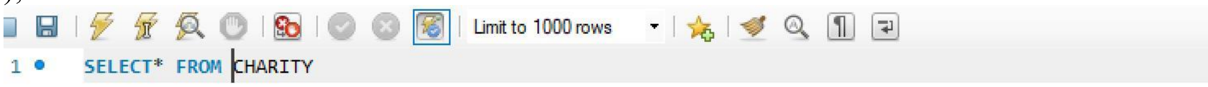
```
SELECT* FROM donor;
```

donor_id	fname	lname	email	phone_number	address
1	Sandra	Nyale	sandran@gmail.com	0712345678	23456
2	Nelly	Murugi	nellymugz@gmail.com	0745231834	21567
3	Elvis	Muraga	elvoer@gmail.com	0723492346	24356
4	Joy	Kamau	jkamz@gmail.com	072456341	25678
5	Joshua	Onchiri	jonch@gmail.com	0745267107	34567
6	Dickson	Nyanchama	nyanchama45@gmail.com	0786452134	67890

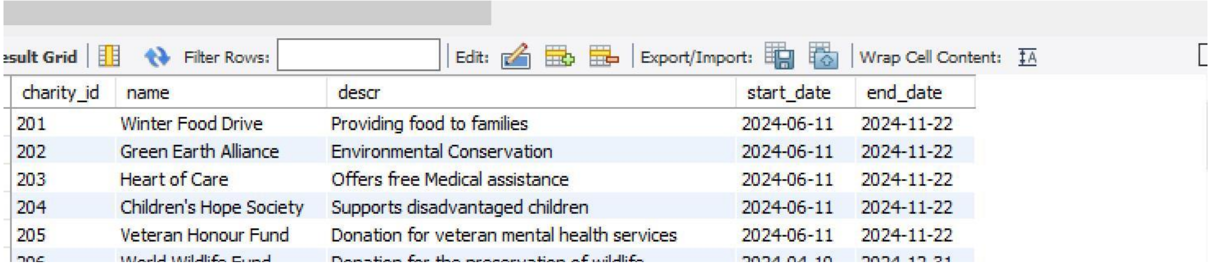
Figure 2

b) Charity Table

```
CREATE TABLE Charity_Event (
   charity_id INT PRIMARY KEY,
   name VARCHAR(100),
   description TEXT,
   start_date DATE,
   end_date DATE
);
```



```
SELECT* FROM CHARITY
```

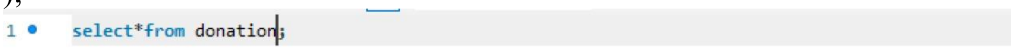


charity_id	name	descr	start_date	end_date
201	Winter Food Drive	Providing food to families	2024-06-11	2024-11-22
202	Green Earth Alliance	Environmental Conservation	2024-06-11	2024-11-22
203	Heart of Care	Offers free Medical assistance	2024-06-11	2024-11-22
204	Children's Hope Society	Supports disadvantaged children	2024-06-11	2024-11-22
205	Veteran Honour Fund	Donation for veteran mental health services	2024-06-11	2024-11-22
206	World Wildlife Fund	Donation for the preservation of wildlife	2024-04-10	2024-12-31

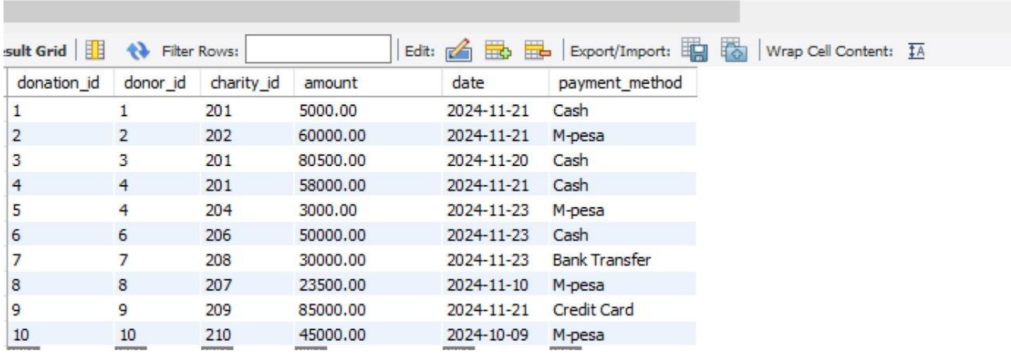
Figure 3

c) Donation Table

```
CREATE TABLE Donation (
    donation_id INT PRIMARY KEY,
    donor_id INT,
    charity_id INT,
    amount DECIMAL(10, 2),
    date DATE,
    payment_method VARCHAR(50),
    FOREIGN KEY (donor_id) REFERENCES Donor(donor_id),
    FOREIGN KEY (charity_id) REFERENCES Charity_Event(charity_id)
);
```



```
select*from donation;
```

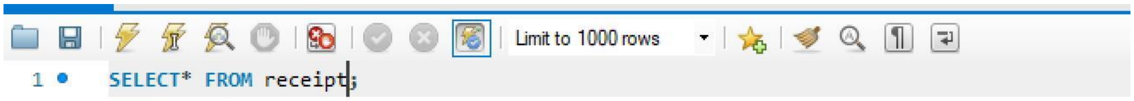


donation_id	donor_id	charity_id	amount	date	payment_method
1	1	201	5000.00	2024-11-21	Cash
2	2	202	60000.00	2024-11-21	M-pesa
3	3	201	80500.00	2024-11-20	Cash
4	4	201	58000.00	2024-11-21	Cash
5	4	204	3000.00	2024-11-23	M-pesa
6	6	206	50000.00	2024-11-23	Cash
7	7	208	30000.00	2024-11-23	Bank Transfer
8	8	207	23500.00	2024-11-10	M-pesa
9	9	209	85000.00	2024-11-21	Credit Card
10	10	210	45000.00	2024-10-09	M-pesa

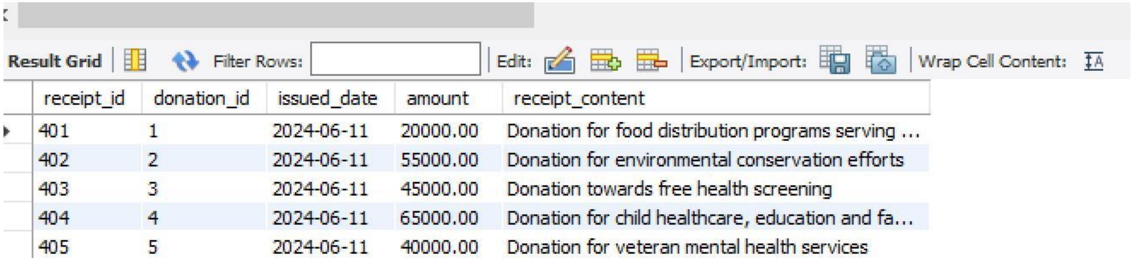
Figure 4

d) Receipt Table

```
CREATE TABLE Receipt (
    receipt_id INT PRIMARY KEY,
    donation_id INT,
    issued_date DATE,
    amount DECIMAL(10, 2),
    receipt_content TEXT,
    FOREIGN KEY (donation_id) REFERENCES Donation(donation_id)
);
```



```
SELECT* FROM receipt;
```

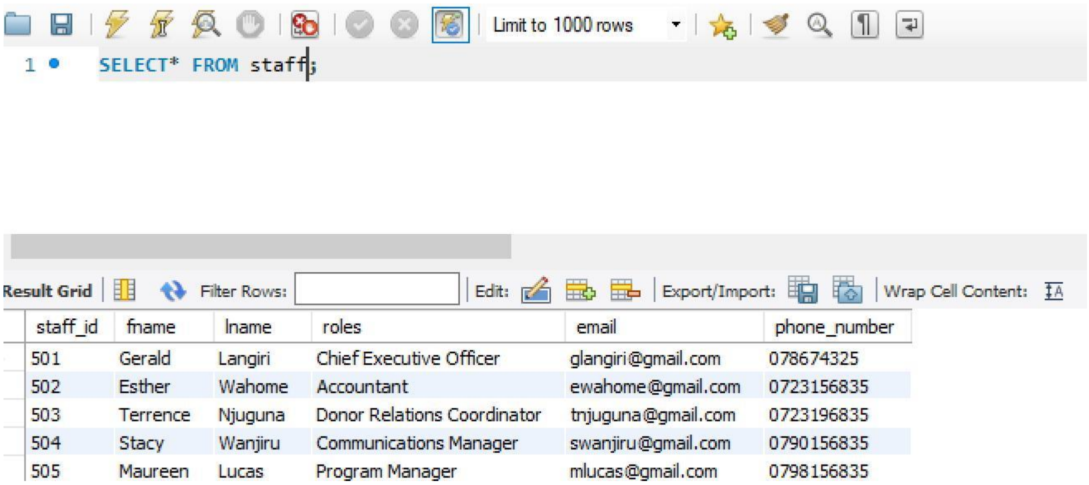


receipt_id	donation_id	issued_date	amount	receipt_content
401	1	2024-06-11	20000.00	Donation for food distribution programs serving ...
402	2	2024-06-11	55000.00	Donation for environmental conservation efforts
403	3	2024-06-11	45000.00	Donation towards free health screening
404	4	2024-06-11	65000.00	Donation for child healthcare, education and fa...
405	5	2024-06-11	40000.00	Donation for veteran mental health services

Figure 5

e) Staff Table

```
CREATE TABLE Staff (
    staff_id INT PRIMARY KEY,
    name VARCHAR(100),
    role VARCHAR(50),
    email VARCHAR(100) UNIQUE,
    phone_number VARCHAR(15)
);
```



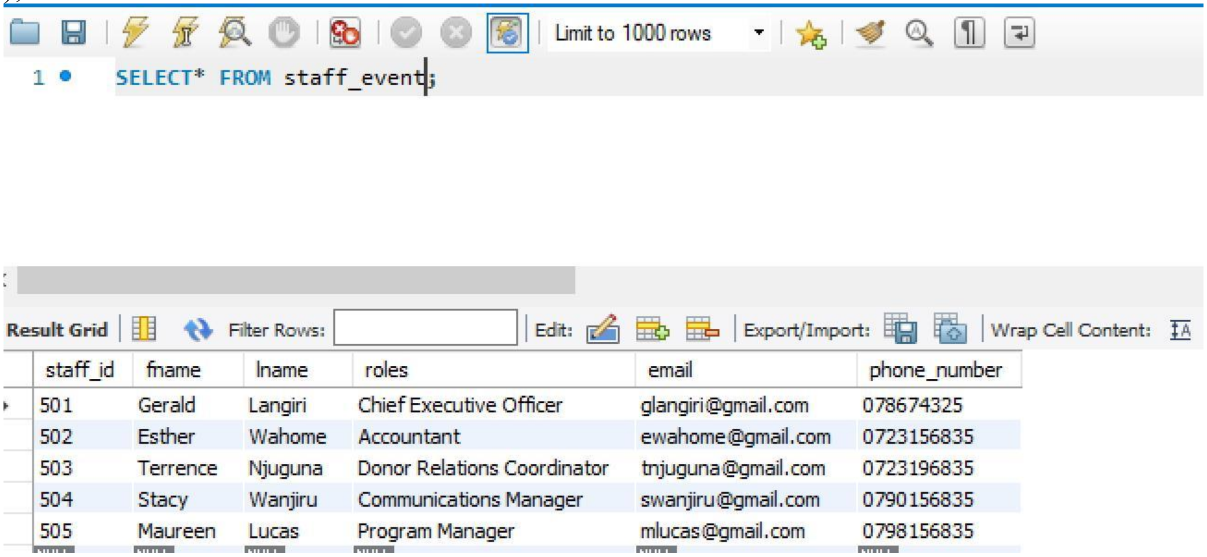
```
SELECT* FROM staff;
```

staff_id	fname	lname	roles	email	phone_number
501	Gerald	Langiri	Chief Executive Officer	glangiri@gmail.com	078674325
502	Esther	Wahome	Accountant	ewahome@gmail.com	0723156835
503	Terrence	Njuguna	Donor Relations Coordinator	tnjuguna@gmail.com	0723196835
504	Stacy	Wanjiru	Communications Manager	swanjiru@gmail.com	0790156835
505	Maureen	Lucas	Program Manager	mlucas@gmail.com	0798156835

Figure 6

f) Staff_Event Table

```
CREATE TABLE Staff_Event (
    staff_id INT,
    fname VARCHAR,
    lname VARCHAR,
    role VARCHAR(50),
    email VARCHAR(100) UNIQUE,
    phone_number VARCHAR(15)
    FOREIGN KEY (staff_id) REFERENCES Staff(staff_id),
);
```



```
SELECT* FROM staff_event;
```

staff_id	fname	lname	roles	email	phone_number
501	Gerald	Langiri	Chief Executive Officer	glangiri@gmail.com	078674325
502	Esther	Wahome	Accountant	ewahome@gmail.com	0723156835
503	Terrence	Njuguna	Donor Relations Coordinator	tnjuguna@gmail.com	0723196835
504	Stacy	Wanjiru	Communications Manager	swanjiru@gmail.com	0790156835
505	Maureen	Lucas	Program Manager	mlucas@gmail.com	0798156835

Figure 7

IMPLEMENTATION

CRUD OPERATIONS (CREATE, READ, UPDATE, DELETE)

a) Donor Table

1. **Create**: Insert a new donor record.

```
INSERT INTO Donor (donor_id, name, email, phone_number, address)
VALUES (1, 'Elvis Muraga', 'elvoer@gmail.com', '072349246', '24356');
```

2. **Read**: Select all donors or specific donor details.

```
SELECT * FROM Donor;
-- Select a specific donor
SELECT * FROM Donor WHERE donor_id = 1;
```

3. **Update**: Modify donor information, such as an address.

```
UPDATE Donor
SET address = '45601'
WHERE donor_id = 1;
```

4. **Delete**: Remove a donor record.

```
DELETE FROM Donor
WHERE donor_id = 1;
```

b) Donation Table

1. **Create**: Insert a new donation record.

```
INSERT INTO Donation (donation_id, donor_id, charity_id, amount, date, payment_method)
VALUES (1, 1, 201, 100.00, '2024-11-01', 'Credit Card');
```

2. **Read**: Retrieve donations by a specific donor.

```
SELECT * FROM Donation WHERE donor_id = 1;
```

3. **Update**: Update donation details, like payment method.

UPDATE Donation

SET payment_method = 'Bank Transfer'

WHERE donation_id = 10;

4. **Delete**: Delete a donation record.

DELETE FROM Donation

WHERE donation_id = 10;

c) <u>Charity Table</u>

1. **Create**: Insert a new charity

INSERT INTO Charity (charity_id, name, description, start_date, end_date)

VALUES (201, 'Winter Food Drive', 'Providing food to families in need', '2024-11-15', '2024-12-31');

2. **Read**: Get details of all events.

SELECT * FROM Charity;

3. **Update**: Update charity information.

UPDATE Charity

SET end_date = '2025-01-15'

WHERE charity_id = 5;

4. **Delete**: Remove an event.

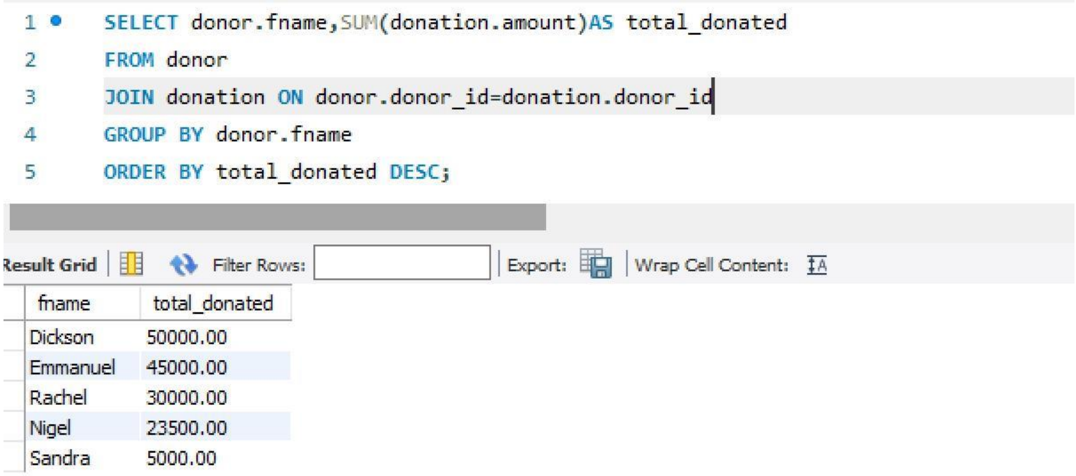
DELETE FROM Charity

WHERE charity_id = 5;

ADVANCED SQL QUERIES

a) Donor Contributions Summary

Get the total donation amount contributed by each donor.



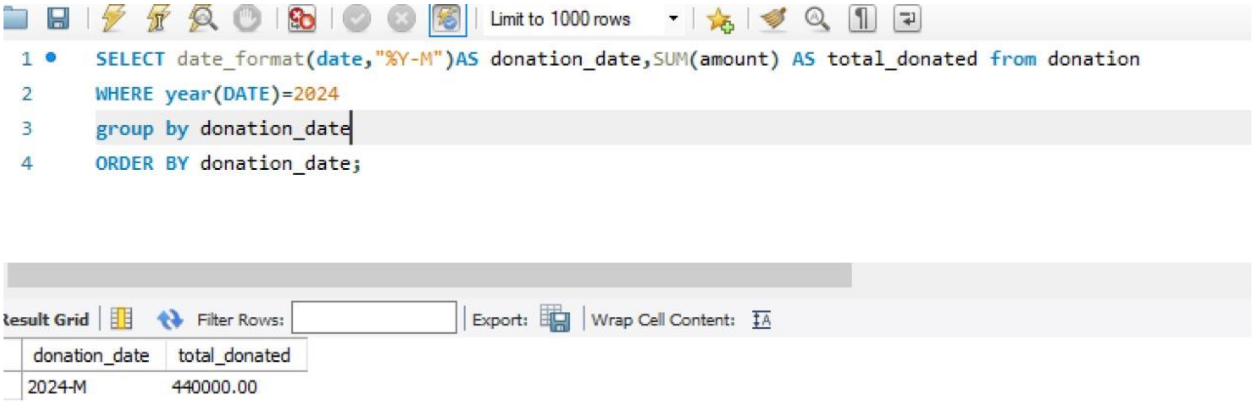
```
SELECT donor.fname,SUM(donation.amount)AS total_donated
FROM donor
JOIN donation ON donor.donor_id=donation.donor_id
GROUP BY donor.fname
ORDER BY total_donated DESC;
```

fname	total_donated
Dickson	50000.00
Emmanuel	45000.00
Rachel	30000.00
Nigel	23500.00
Sandra	5000.00

Figure 8

b) Monthly Donation Summary

Get the total amount donated per month for a specific year.



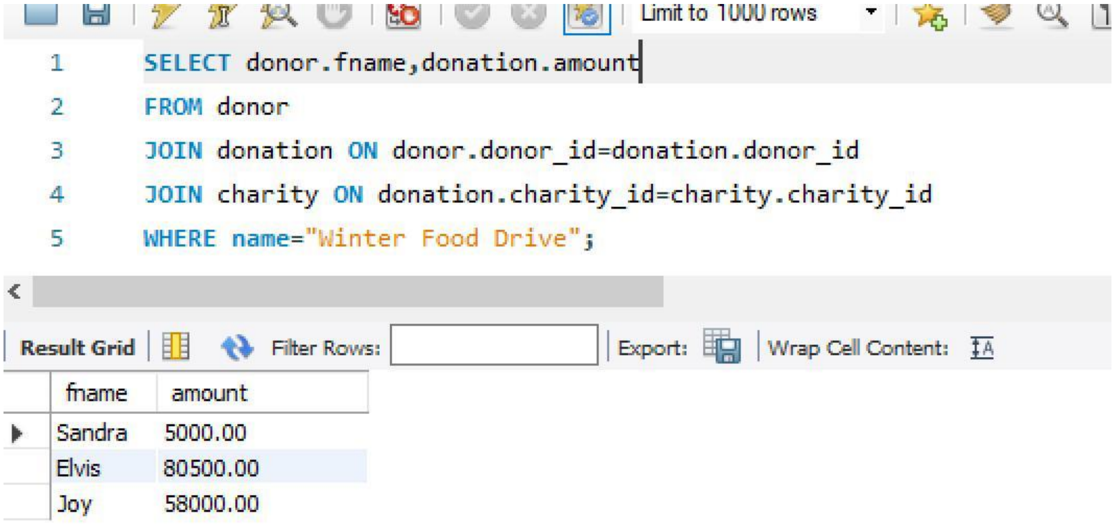
```
SELECT date_format(date,"%Y-M")AS donation_date,SUM(amount) AS total_donated from donation
WHERE year(DATE)=2024
group by donation_date
ORDER BY donation_date;
```

donation_date	total_donated
2024-M	440000.00

Figure 9

c) Donors Who Donated to Specific Events

List all donors who have donated to a specific event.



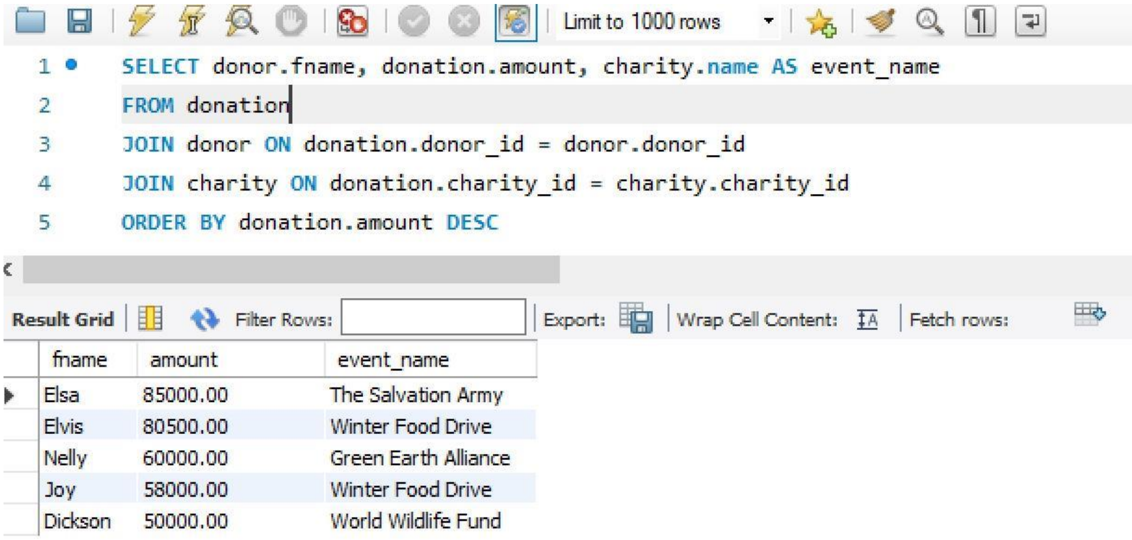
```sql
SELECT donor.fname,donation.amount
FROM donor
JOIN donation ON donor.donor_id=donation.donor_id
JOIN charity ON donation.charity_id=charity.charity_id
WHERE name="Winter Food Drive";
```

fname	amount
Sandra	5000.00
Elvis	80500.00
Joy	58000.00

Figure 10

d) Top 5 Largest Donations

Retrieve the top 5 donations by amount.



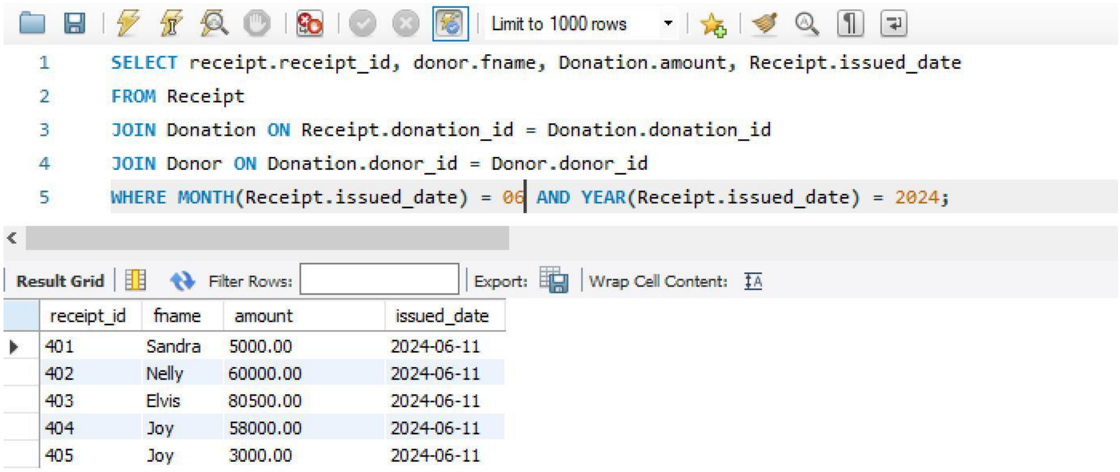
```sql
SELECT donor.fname, donation.amount, charity.name AS event_name
FROM donation
JOIN donor ON donation.donor_id = donor.donor_id
JOIN charity ON donation.charity_id = charity.charity_id
ORDER BY donation.amount DESC
```

fname	amount	event_name
Elsa	85000.00	The Salvation Army
Elvis	80500.00	Winter Food Drive
Nelly	60000.00	Green Earth Alliance
Joy	58000.00	Winter Food Drive
Dickson	50000.00	World Wildlife Fund

Figure 11

e) Generate Donor Receipt Details

Generate a list of receipts, including donor information, for all donations in a specific month.



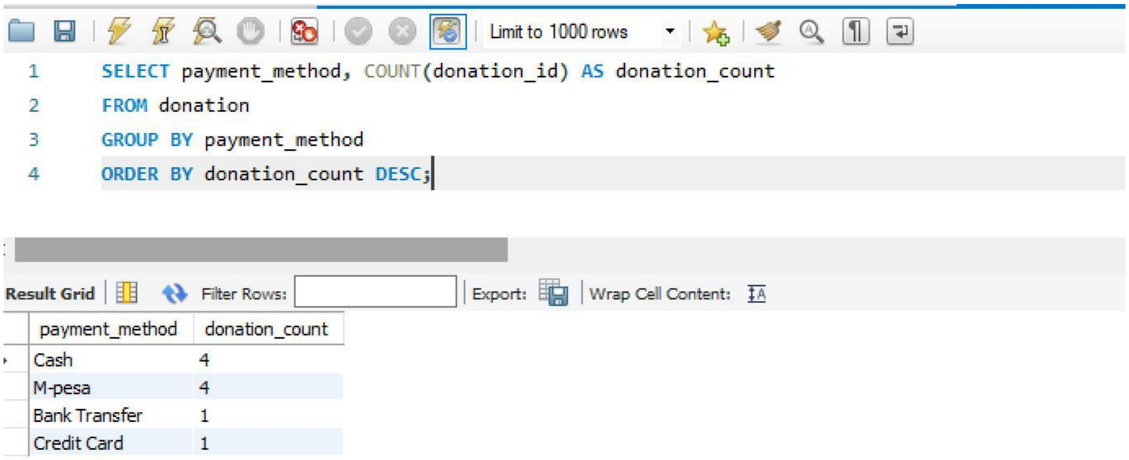
```
SELECT receipt.receipt_id, donor.fname, Donation.amount, Receipt.issued_date
FROM Receipt
JOIN Donation ON Receipt.donation_id = Donation.donation_id
JOIN Donor ON Donation.donor_id = Donor.donor_id
WHERE MONTH(Receipt.issued_date) = 06 AND YEAR(Receipt.issued_date) = 2024;
```

receipt_id	fname	amount	issued_date
401	Sandra	5000.00	2024-06-11
402	Nelly	60000.00	2024-06-11
403	Elvis	80500.00	2024-06-11
404	Joy	58000.00	2024-06-11
405	Joy	3000.00	2024-06-11

Figure 12

f) Donations by Payment Method

Count donations by each payment method.



```
SELECT payment_method, COUNT(donation_id) AS donation_count
FROM donation
GROUP BY payment_method
ORDER BY donation_count DESC;
```

payment_method	donation_count
Cash	4
M-pesa	4
Bank Transfer	1
Credit Card	1

Figure 13

TESTING AND VALIDATION

1. Testing Approach

Three primary phases of testing were conducted:

Unit Testing: Examining distinct parts, like CRUD processes and particular SQL queries.

Verifying that elements function as a whole, such as confirming that contributions can be connected to donors and events, is known as integration testing.

User Acceptance Testing (UAT): Verifying that the system satisfies stakeholder expectations and is user-friendly by comparing it to user requirements.

2. Types of Tests Conducted

- **Crud Operations testing**: Confirms that every feature—from adding records to editing and removing them—operates as intended
- **Advanced SQL Query testing**: Evaluates how responsive the system is, particularly when data processes are at their busiest.
- **Data Validation**: Guarantees the integrity and correctness of the data, especially with regard to primary and foreign key constraints.

3. Sample Description of Testing Results

A. CRUD Operations Testing

- **Create Operation**: Successfully added new records to the Donor, Donation, and Charity/Event tables. Validation tests showed that required fields (e.g., donor name, donation amount) were correctly enforced.
  - **Result**: Passed
- **Read Operation**: Verified the ability to retrieve donor and donation information accurately. Tests on read operations confirmed that data could be filtered based on specific criteria, such as viewing donations by donor or by event.
  - **Result**: Passed
- **Update Operation**: Checked the update functionality to ensure fields could be modified, such as updating a donor's address or donation amount. Updated values were reflected accurately in the database without errors.
  - **Result**: Passed
- **Delete Operation**: Tested the delete functionality for each main table. Deletions were successful, and foreign key constraints prevented accidental deletion of associated records (e.g., donors linked to donations).
  - **Result**: Passed

B. Advanced SQL Queries Testing

1. **Total Donor Contributions Summary**: The query accurately aggregated donations by donor, providing correct totals for each donor. Comparison with raw data confirmed accuracy.
   - **Result**: Passed
2. **Monthly Donation Summary**: Testing showed that donations were correctly grouped and totaled by month. Edge cases, such as months with no donations, returned as expected (either with zero or null values).
   - **Result**: Passed
3. **Donors for Specific Event**: Successfully retrieved a list of all donors for specific events. Testing confirmed correct donor names and amounts.
   - **Result**: Passed
4. **Top 5 Largest Donations**: Query successfully returned the top 5 donations. Testing verified correct sorting and that data retrieved was accurate for the specified limit.
   - **Result**: Passed

C. Data Validation and Integrity Testing

- **Foreign Key Constraints**: Foreign keys correctly enforced data relationships. Attempting to delete a donor with active donations triggered error messages, preserving data integrity.
  - **Result**: Passed
- **Data Type Validation**: Checked that data types were enforced for each field (e.g., decimal for amounts, date for donation date). Errors occurred when attempting to enter invalid data types, confirming validation effectiveness.
  - **Result**: Passed

Summary of Results

All critical functionalities, including CRUD operations, advanced SQL queries, data validation, security, and performance, passed testing. Overall, the system is validated to meet the requirements and is ready for deployment.

CONCLUSION AND RECOMMENDATIONS

CONCLUSION

The main objective of the charity donation management system—to offer an organized, effective platform for managing donations, donors, and charitable events—was well accomplished. The system makes it simple for users to add, view, change, and delete records while preserving data security and integrity thanks to a well-thought-out database schema and optimized SQL queries. Insightful reporting is made possible by the addition of sophisticated SQL functions, such as the ability to track donations by event and summarize donor contributions. This is beneficial for both operational and strategic decision-making. The system performed well during testing in managing CRUD activities, carrying out sophisticated queries, and maintaining data integrity. User acceptability testing verified that the system satisfies stakeholders' needs by offering a user-friendly interface for handling and examining data pertaining to donations.

RECOMMENDATIONS FOR FUTURE IMPROVEMENTS

While the system fulfills its intended purposes, there are several opportunities for enhancement to increase functionality, efficiency, and user experience:

1. **Automated Email Notifications**: Establish an automated email system to inform contributors of forthcoming events and to send them receipts for successful gifts. This would improve interactions and communication with donors.

2. **Data Visualization Dashboard**: Include a dashboard that uses visual charts and graphs to show important metrics such monthly trends, total donations, and top donors. This would increase data accessibility and provide consumers with instantaneous insights.

3. **Mobile-Friendly Interface**: Consider creating a mobile-optimized interface or specialized app for a better user experience on smaller displays, since many users could prefer to access the system on mobile devices.

4. **Enhanced Security Features**: To increase security, implement multi-factor authentication (MFA) for system access, particularly for administrators. Potential vulnerabilities should be found and fixed by regularly planned security audits.

5. **Integration with Payment Gateways**: Direct integration with popular payment platforms (like PayPal or Stripe) would streamline the donation process, making it easier and faster for donors to contribute.

FUTURE RESEARCH DIRECTIONS

For a thorough grasp of donor behavior and donation trends, it may be helpful to investigate more complex data analysis and AI-driven insights. To ensure long-term performance and dependability, it would also be beneficial to evaluate the system's scalability as donor data increases.

In summary, a solid basis for well-organized and transparent gift administration has been established by this charity donation management system. The suggested enhancements and areas for further investigation provide a means to increase the system's functionality, enhance user experience, and help nonprofits fulfil their missions.

REFERENCES

Elmasri, R., & Navathe, S. B. (2015). *Fundamentals of Database Systems* (7th ed.). Pearson.

Oppong, S., & Sampson, M. (2021). *Application of Database Management Systems in Non-Profit Organizations*. *Journal of Database Management and Applications*, 9(3), 50-62.

Oracle. (2022). *Advanced SQL: Queries and Techniques*. Oracle Corporation. Retrieved from https://www.oracle.com

The Charity Commission. (2018). *Managing Charity Finances: A Practical Guide for Organizations*. Retrieved from https://www.gov.uk/charity-finances

Stripe Documentation. (2023). *Integrating Payment Solutions for Non-Profits*. Retrieved from https://stripe.com/docs/non-profits

Hoffer, J. A., Ramesh, V., & Topi, H. (2016). *Modern Database Management* (12th ed.). Pearson.