Introduction

Analysis of the Problem

The project's main issues focus on the need for a centralized, effective system for managing library operations. Key problem points include:

- **Manual Processes**: Existing systems require significant manual intervention, leading to inefficiencies and inaccuracies.
- **Poor Data Tracking**: Inadequate tracking of which students have borrowed which books and the absence of organized student preferences.

By establishing a relational database system with clearly defined tables and relationships, this project aims to address these problems.

Discussion of Data

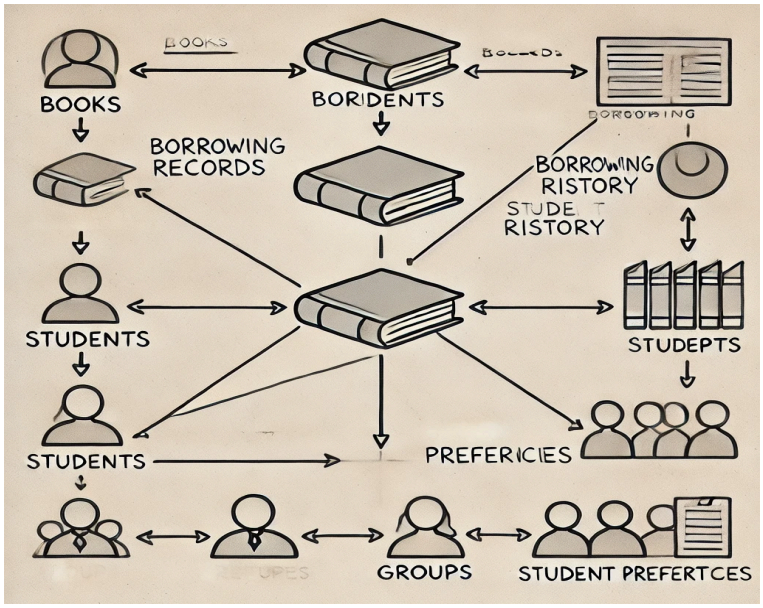
The database will primarily manage four categories of data:

1. **Books**: Details such as title, author, publication year, genre, and availability status.
2. **Students**: Information like student ID, name, and associated borrowing records.
3. **Groups**: Organizational units that group students based on shared preferences or academic needs.
4. **Preferences**: Records of students' preferred genres or book types, which inform group assignments and library resource planning.

Database Design

The database design is based on an Entity-Relationship diagram, which illustrates the key entities and their relationships:

- **Books**: The central entity, linked to borrowing records and student preferences.
- **Students**: Associated with borrowing history and preferences.
- **Groups**: Connected to students and preferences to reflect collective reading interests.



Explanation of Tables and Relationships

1. **Books Table**: Contains fields like BookID, Title, Author, Genre, PublicationYear, and AvailabilityStatus.
2. **Students Table**: Includes StudentID, Name, Email, and GroupID.
3. **Groups Table**: Contains GroupID and GroupName, serving as a link to students.
4. **Preferences Table**: Maps student preferences to genres, using StudentID and Genre.
5. **Borrowing Table**: Tracks BorrowID, StudentID, BookID, BorrowDate, and ReturnDate.

These tables' connections ensure data integrity and enable efficient querying. The Students table and the Borrowing table, for example, have a one-to-many link, however the Books table and Borrowing have a similar relationship.

```
-- Books table

CREATE TABLE books (

    id INTEGER PRIMARY KEY,

    isbn TEXT NOT NULL,

    title TEXT NOT NULL,

    rating INTEGER CHECK(rating BETWEEN 1 AND 10),
```

```
    format TEXT CHECK(format IN ('paperback', 'hardcover')),

    pages INTEGER,

    publication_date DATE,

    year INTEGER,

    is_checked_out BOOLEAN DEFAULT 0

);



-- Students table

CREATE TABLE students (

    id INTEGER PRIMARY KEY,

    first_name TEXT NOT NULL,

    last_name TEXT NOT NULL,

    email TEXT,

    group_id CHAR(1),

    FOREIGN KEY (group_id) REFERENCES groups(id)

);



-- Groups table

CREATE TABLE groups (

    id CHAR(1) PRIMARY KEY,

    group_name TEXT

);
```

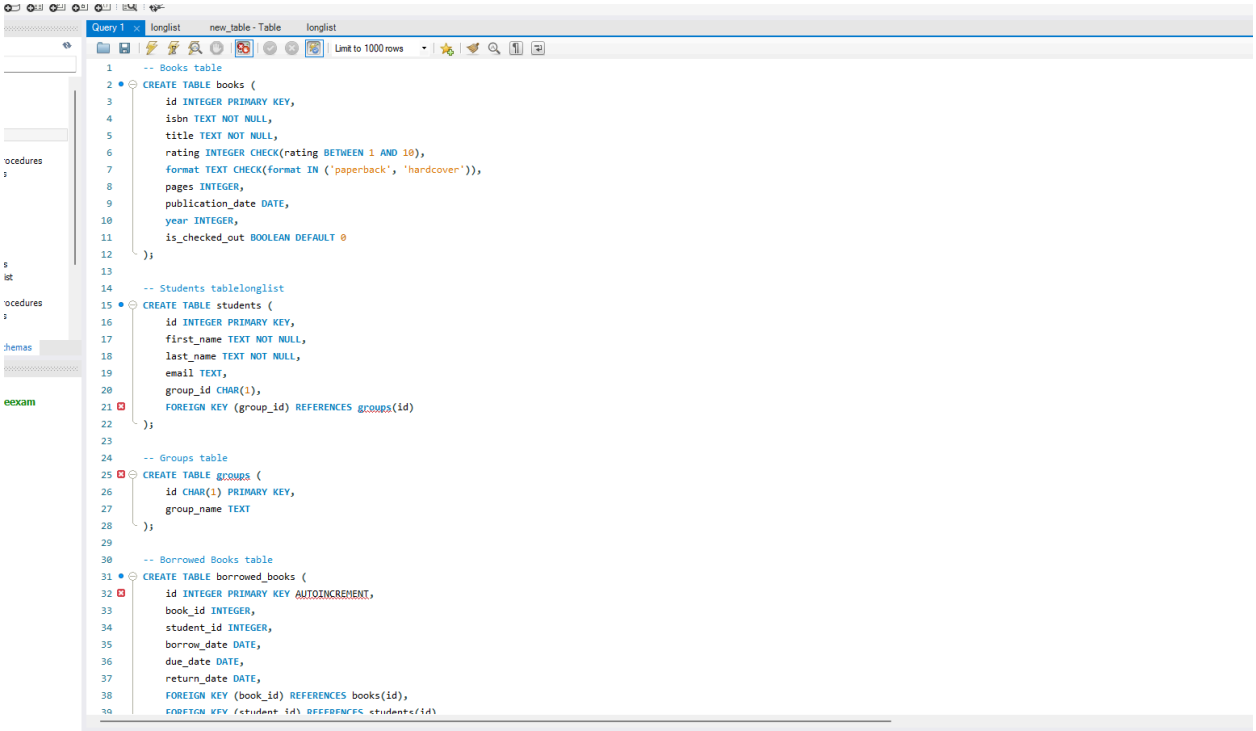
```sql
-- Borrowed Books table

CREATE TABLE borrowed_books (

    id INTEGER PRIMARY KEY AUTOINCREMENT,

    book_id INTEGER,

    student_id INTEGER,

    borrow_date DATE,

    due_date DATE,

    return_date DATE,

    FOREIGN KEY (book_id) REFERENCES books(id),

    FOREIGN KEY (student_id) REFERENCES students(id)

);

CREATE TABLE preferences (

    student_id INTEGER NOT NULL,

    isbn TEXT NOT NULL,

    PRIMARY KEY (student_id, isbn),

    FOREIGN KEY (student_id) REFERENCES students(student_id),

    FOREIGN KEY (isbn) REFERENCES books(isbn)

);
```

Implementation

Tools Used

1. **Node.js**: A runtime environment for building fast, scalable, and event-driven server-side applications. It was selected for its asynchronous capabilities, which ensure efficient handling of database queries and interactions.

2. **Mysql**: A lightweight, self-contained SQL database engine, ideal for small-to-medium-sized projects. It was chosen for its simplicity and ability to handle structured data effectively.



```sql
-- Books table
CREATE TABLE books (
    id INTEGER PRIMARY KEY,
    isbn TEXT NOT NULL,
    title TEXT NOT NULL,
    rating INTEGER CHECK(rating BETWEEN 1 AND 10),
    format TEXT CHECK(format IN ('paperback', 'hardcover')),
    pages INTEGER,
    publication_date DATE,
    year INTEGER,
    is_checked_out BOOLEAN DEFAULT 0
);

-- Students tablelonglist
CREATE TABLE students (
    id INTEGER PRIMARY KEY,
    first_name TEXT NOT NULL,
    last_name TEXT NOT NULL,
    email TEXT,
    group_id CHAR(1),
    FOREIGN KEY (group_id) REFERENCES groups(id)
);

-- Groups table
CREATE TABLE groups (
    id CHAR(1) PRIMARY KEY,
    group_name TEXT
);

-- Borrowed Books table
CREATE TABLE borrowed_books (
    id INTEGER PRIMARY KEY AUTOINCREMENT,
    book_id INTEGER,
    student_id INTEGER,
    borrow_date DATE,
    due_date DATE,
    return_date DATE,
    FOREIGN KEY (book_id) REFERENCES books(id),
    FOREIGN KEY (student_id) REFERENCES students(id)
```

3. **Visual Studio Code**: A powerful integrated development environment (IDE) used for coding, debugging, and testing the application. It supports extensions for this project which made things less complicated for the development process.
4. **GitHub**: For version control and collaboration, ensuring that the project remains well-organized and easily accessible for future enhancements.

library.db | app.js | index.html

C: > Users > steph_ > OneDrive > Documents > DATABASE PROJECT > app.js > ...

```javascript
const mysql = require('mysql2');

// Create a connection to the database
const connection = mysql.createConnection({
  host: 'localhost',
  user: 'root',
  password: 'Ogechi1561!',
  database: 'library.db'
});

// Connect to MySQL
connection.connect((err) => {
  if (err) {
    console.error('Error connecting to the database:', err.stack);
    return;
  }
  console.log('Connected to MySQL as id ' + connection.threadId);
});

// Insert books into the database
const insertBooks = () => {
  const books = [
    ['9788439736967', 'Boulder', 10, 'paperback', 112, '2022-08-02', 2023, 1],
    ['9781628971538', 'Whale', 3, 'paperback', 368, '2023-01-19', 2023, 0],
    ['9781642861181', 'The Gospel According to the New World', 32, 'paperback', 184, '2023-03-07', 2023, 1],
    ['9781787300453', 'Mac and His Problem', 7, 'paperback', 224, '2019-06-06', 2020, 0],
    ['9781474623239', 'Time Shelter', 30, 'hardcover', 304, '2022-04-21', 2023, 1],
    ['9781846277493', 'Is Mother Dead', 29, 'paperback', 330, '2022-09-27', 2023, 0],
    ['9781538758255', 'Jimi Hendrix Live in Lviv', 12, 'hardcover', 416, '2023-04-27', 2023, 0],
    ['9781951142179', 'The Birthday Party', 27, 'paperback', 454, '2023-01-24', 2023, 0],
    ['9781800611522', 'While We Were Dreaming', 5, 'paperback', 528, '2023-03-30', 2023, 1],
    ['9781782275630', 'Pyre', 17, 'paperback', 224, '2022-04-08', 2023, 1],
```