

TITLE: PROJECT STAGE 2

NAME: KAREN ADJOA SELASI KOOMSON

Stage 2 – Database Implementation, Querying & Manipulation

1. Introduction

This stage implements the Public Library database from Stage 1 using **PostgreSQL**. I created the schema, loaded the provided CSV datasets, and executed analytical queries to validate functionality (borrowing, staff accountability, reporting).

Environment: PostgreSQL (psql + pgAdmin 4) on Windows.

2. Database Schema (DDL)

2.1 Genre

```
CREATE TABLE Genre (  
    Genre_ID SERIAL PRIMARY KEY,  
    Name VARCHAR(100) NOT NULL,  
    Description TEXT  
);
```

2.2 Catalog

```
CREATE TABLE Catalog (  
    Catalog_ID SERIAL PRIMARY KEY,  
    Name VARCHAR(100) NOT NULL,  
    Location VARCHAR(100)  
);
```

2.3 Material

```
CREATE TABLE Material (  
    Material_ID SERIAL PRIMARY KEY,  
    Title VARCHAR(200) NOT NULL,  
    Publication_Date DATE,  
    Catalog_ID INT REFERENCES Catalog(Catalog_ID) ON DELETE SET NULL,  
    Genre_ID INT REFERENCES Genre(Genre_ID) ON DELETE SET NULL );
```

2.4 Author

```
CREATE TABLE Author (  
    Author_ID SERIAL PRIMARY KEY,  
    Name VARCHAR(150) NOT NULL,  
    Nationality VARCHAR(100)  
);
```

2.5 Authorship

```
CREATE TABLE Authorship (  
    Authorship_ID SERIAL PRIMARY KEY,  
    Author_ID INT NOT NULL REFERENCES Author(Author_ID) ON DELETE CASCADE,  
    Material_ID INT NOT NULL REFERENCES Material(Material_ID) ON DELETE  
CASCADE );
```

2.6 Member

```
CREATE TABLE Member (  
    Member_ID SERIAL PRIMARY KEY,  
    Name VARCHAR(150) NOT NULL,  
    Contact_Info VARCHAR(150),  
    Join_Date DATE  
);
```

2.7 Staff

```
CREATE TABLE Staff (  
    Staff_ID SERIAL PRIMARY KEY,  
    Name VARCHAR(150) NOT NULL,  
    Contact_Info VARCHAR(150),  
    Job_Title VARCHAR(100),  
    Hire_Date DATE  
);
```

2.8 Borrow

```
CREATE TABLE Borrow (  
    Borrow_ID SERIAL PRIMARY KEY,  
    Material_ID INT NOT NULL REFERENCES Material(Material_ID) ON DELETE  
CASCADE,  
    Member_ID INT NOT NULL REFERENCES Member(Member_ID) ON DELETE CASCADE,  
    Staff_ID INT NOT NULL REFERENCES Staff(Staff_ID) ON DELETE SET NULL,  
    Borrow_Date DATE NOT NULL,  
    Due_Date DATE NOT NULL,  
    Return_Date DATE  
);
```

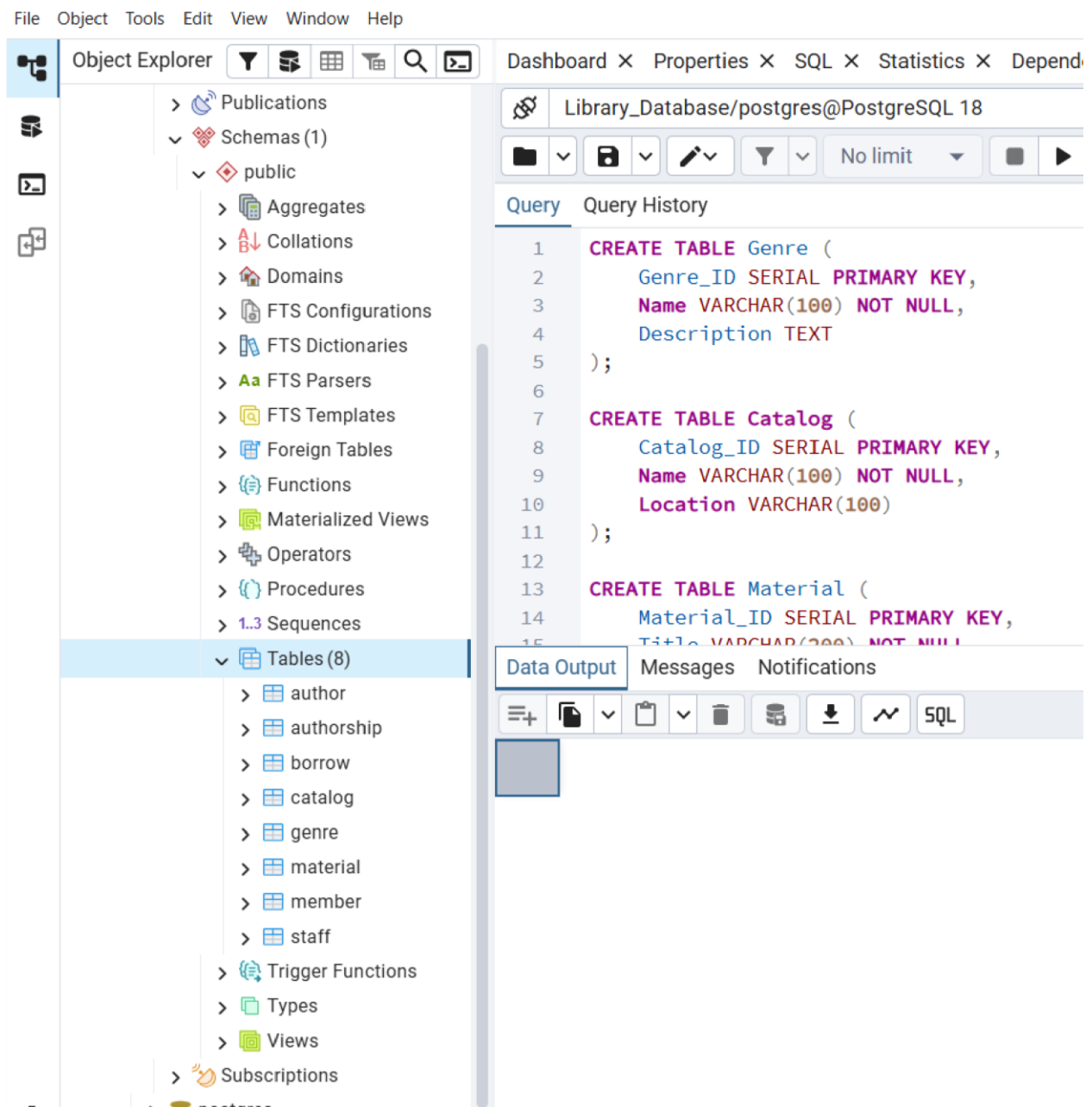


Figure 1 Successful creation of the all table

3. Data Loading

I imported the eight CSVs in a FK-safe order: **Genre** → **Catalog** → **Material** → **Author** →

Authorship → **Member** → **Staff** → **Borrow**. I used psql \COPY with UTF-8-cleaned files to avoid permission/encoding issues since I was struggling loading files on pgAdmin4.

Command used:

```
postgres=# CREATE DATABASE Library_Database;
CREATE DATABASE
postgres=# \c Library_Database;
You are now connected to database "Library_Database" as user "postgres".

Library_Database=# \COPY Genre FROM 'C:/data/Genre_UTF8.csv' DELIMITER ',' CSV
HEADER;
COPY 8

Library_Database=# \COPY Catalog FROM 'C:/data/Catalog.csv' DELIMITER ',' CSV
HEADER;
COPY 10

Library_Database=# \COPY Material FROM 'C:/data/Material.csv' DELIMITER ','
CSV HEADER;
ERROR:  invalid input syntax for type date: "1922"
CONTEXT:  COPY material, line 2, column publication_date: "1922"

Library_Database=# \COPY Material FROM 'C:/data/Material.csv' DELIMITER ','
CSV HEADER;
COPY 94

Library_Database=# \COPY Author FROM 'C:/data/Author.csv' DELIMITER ',' CSV
HEADER;
COPY 36

Library_Database=# \COPY Authorship FROM 'C:/data/Authorship.csv' DELIMITER
',' CSV HEADER;
COPY 94

Library_Database=# \COPY Member FROM 'C:/data/Member.csv' DELIMITER ',' CSV
HEADER;
COPY 40

Library_Database=# \COPY Staff FROM 'C:/data/Staff.csv' DELIMITER ',' CSV
HEADER;
COPY 14

Library_Database=# \COPY Borrow FROM 'C:/data/Borrow.csv' DELIMITER ',' CSV
HEADER;
COPY 90
```

4. Querying & Manipulation

Q1. Members registered in 2018

```
SELECT name, join_date  
  
FROM member  
WHERE EXTRACT(YEAR FROM join_date) = 2018;
```

	name character varying (150) 🔒	join_date date 🔒
1	Alice Johnson	2018-01-10
2	Bob Smith	2018-03-15
3	Carol Brown	2018-06-20
4	David Williams	2018-09-18

Figure 2Query 1 – Members registered in 2018

Q2. Total books borrowed per member

```
SELECT m.name, COUNT(b.borrow_id) AS total_borrowed  
FROM member AS m  
JOIN borrow AS b ON m.member_id = b.member_id  
GROUP BY m.name  
ORDER BY total_borrowed DESC, m.name;
```



	name character varying (150) 	total_borrowed bigint 
1	Emily Miller	7
2	Harry Garcia	6
3	Grace Wilson	6
4	David Williams	5
5	Alice Johnson	5
6	Frank Davis	4
7	Isla Thomas	4
8	Michael Harris	4
9	Kate Anderson	4
10	Bob Smith	4
11	Luke Jackson	3
12	Carol Brown	3
13	Matthew Smith	2
14	Tiffany Allen	2
15	Kelly Miller	2
16	Mary Nelson	2
17	Rachel Young	2
18	Benjamin Costa	2
19	Peter Lewis	2
20	Olivia Clark	2
21	Mia White	2

Figure 3Query 2- Total books borrowed per member

22	Lisa Daniels	2
23	Quinn Hall	2
24	Noah Harris	2
25	Jack Martinez	2
26	Caroline Miller	1
27	Anthony Herrera Jr.	1
28	Casey Davis	1
29	Jennifer Johnson	1
30	Anita Clark DVM	1
31	Sam Walker	1
32	Robin Key	1
33	Gregory Carr	1
34	Robert Kaufman	1

Q3. Average borrowing duration (days)

```
SELECT ROUND(AVG(b.return_date - b.borrow_date), 2) AS  
avg_borrow_duration_days  
FROM borrow AS b  
WHERE b.return_date IS NOT NULL;
```

	avg_borrow_duration numeric
1	16.10

Figure 4 Query 3. Average borrowing duration (days)

Q4. Author with the most books

```
SELECT a.name AS author_name, COUNT(DISTINCT au.material_id) AS total_books  
FROM author AS a  
JOIN authorship AS au ON a.author_id = au.author_id  
GROUP BY a.name  
ORDER BY total_books DESC, a.name  
LIMIT 1;
```

	author_name character varying (150)	total_books bigint
1	Oscar Wilde	8

Figure 5 Query 4. Author with the most books

Q5. Authors with their book titles

```
SELECT a.name AS author_name, m.title AS book_title FROM  
author AS a  
JOIN authorship AS au ON a.author_id = au.author_id  
JOIN material AS m ON au.material_id = m.material_id  
ORDER BY a.name, m.title;
```

	author_name character varying (150) 🔒	book_title character varying (200) 🔒
1	Albert Camus	The Stranger
2	Aldous Huxley	Brave New World
3	Alexandre Dumas	The Count of Monte Cristo
4	Alexandre Dumas	The Man in the Iron Mask
5	Alexandre Dumas	The Black Tulip
6	Alexandre Dumas	The Three Musketeers
7	Arthur Koestler	Darkness at Noon
8	Bram Stoker	Dracula
9	Charles Dickens	Oliver Twist
10	Charles Dickens	Great Expectations
11	Charles Dickens	David Copperfield
12	Charles Dickens	Little Dorrit
13	Charles Dickens	A Tale of Two Cities
14	Charles Dickens	Bleak House
15	Charles Dickens	Hard Times
16	Charlotte Bronte	Jane Eyre
17	D.H. Lawrence	Sons and Lovers
18	Emily Bronte	Wuthering Heights
19	Ernest Hemingway	For Whom the Bell Tolls
20	Ernest Hemingway	The Old Man and the Sea
21	Ernest Hemingway	The Sun Also Rises

	author_name character varying (150) 🔒	book_title character varying (200) 🔒
43	Herman Melville	Bartleby, the Scrivener
44	Herman Melville	Typee
45	Herman Melville	Moby-Dick
46	J.D. Salinger	The Catcher in the Rye
47	J.R.R. Tolkien	The Lord of the Rings
48	Jack Kerouac	On the Road
49	Jack London	Martin Eden
50	Jack London	White Fang
51	Jack London	The Sea-Wolf
52	Jack London	The Call of the Wild
53	James Joyce	Ulysses
54	James Joyce	A Portrait of the Artist as a Young Man
55	Jane Austen	Pride and Prejudice
56	Jane Austen	Sense and Sensibility
57	Jane Austen	Emma
58	Jane Austen	Persuasion
59	Jane Austen	Northanger Abbey
60	Jane Austen	Mansfield Park
61	John Steinbeck	East of Eden
62	John Steinbeck	The Grapes of Wrath
63	John Steinbeck	Of Mice and Men

	author_name character varying (150) 🔒	book_title character varying (200) 🔒
22	Ernest Hemingway	A Farewell to Arms
23	F. Scott Fitzgerald	Tender Is the Night
24	F. Scott Fitzgerald	This Side of Paradise
25	F. Scott Fitzgerald	The Great Gatsby
26	F. Scott Fitzgerald	The Beautiful and Damned
27	Franz Kafka	Amerika
28	Franz Kafka	The Metamorphosis
29	Franz Kafka	The Castle
30	Franz Kafka	The Trial
31	Fyodor Dostoevsky	Demons
32	Fyodor Dostoevsky	Crime and Punishment
33	Fyodor Dostoevsky	The Brothers Karamazov
34	Fyodor Dostoevsky	The Idiot
35	Gabriel Garcia Marquez	One Hundred Years of Solitude
36	George Orwell	Nineteen Eighty-Four
37	H.G. Wells	The Island of Doctor Moreau
38	H.G. Wells	The Invisible Man
39	H.G. Wells	The Time Machine
40	H.G. Wells	The War of the Worlds
41	Harper Lee	To Kill a Mockingbird
42	Herman Melville	Billy Budd, Sailor

	author_name character varying (150) 🔒	book_title character varying (200) 🔒
64	Joseph Conrad	The Secret Agent
65	Joseph Conrad	Heart of Darkness
66	Joseph Conrad	Lord Jim
67	Joseph Conrad	Nostromo
68	Joseph Heller	Catch-22
69	Leo Tolstoy	The Death of Ivan Ilyich
70	Leo Tolstoy	War and Peace
71	Leo Tolstoy	Anna Karenina
72	Leo Tolstoy	Resurrection
73	Mark Twain	The Prince and the Pauper
74	Mark Twain	The Adventures of Huckleberry Finn
75	Mark Twain	The Adventures of Tom Sawyer
76	Mark Twain	A Connecticut Yankee in King Arthur's Co...
77	Mary Shelley	Frankenstein
78	Nathaniel Hawthorne	The Scarlet Letter
79	Oscar Wilde	An Ideal Husband
80	Oscar Wilde	An Ideal Husband
81	Oscar Wilde	Lady Windermere's Fan
82	Oscar Wilde	The Importance of Being Earnest
83	Oscar Wilde	The Picture of Dorian Gray
84	Oscar Wilde	The Picture of Dorian Gray

85	Oscar Wilde	The Importance of Being Earnest
86	Oscar Wilde	Lady Windermere's Fan
87	Toni Morrison	Beloved
88	Victor Hugo	The Last Day of a Condemned Man
89	Victor Hugo	The Man Who Laughs
90	Victor Hugo	The Hunchback of Notre-Dame
91	Victor Hugo	Les Misérables
92	Victor Hugo	The Toilers of the Sea
93	Vladimir Nabokov	Lolita
94	William Faulkner	The Sound and the Fury

Figure 6 Query 5. Authors with their book titles

Q6. Most borrowed book

```

SELECT m.title, COUNT(b.borrow_id) AS times_borrowed
FROM material AS m
JOIN borrow AS b ON m.material_id = b.material_id
GROUP BY m.title
ORDER BY times_borrowed DESC, m.title
LIMIT 1;

```

	title character varying (200) 🔒	times_borrowed bigint 🔒
1	Catch-22	5

Figure 7 Query 6. Most borrowed book

Q7. Members with unreturned books

```

SELECT DISTINCT m.name
FROM member AS m
JOIN borrow AS b ON m.member_id = b.member_id
WHERE b.return_date IS NULL
ORDER BY m.name;

```

	name character varying (150) 🔒
1	Mia White
2	Quinn Hall
3	Harry Garcia
4	Olivia Clark
5	Frank Davis
6	Noah Harris
7	Luke Jackson
8	Kelly Miller
9	Tiffany Allen
10	Alice Johnson
11	Casey Davis
12	Grace Wilson
13	Caroline Miller
14	Isla Thomas
15	Kate Anderson
16	Michael Harris

Figure 8 Query 7. Members with unreturned books

Q8. Staff who processed the most transactions

```

SELECT s.name AS staff_name, COUNT(b.borrow_id) AS transactions_handled
FROM staff AS s
JOIN borrow AS b ON s.staff_id = b.staff_id
GROUP BY s.name
ORDER BY transactions_handled DESC, s.name
LIMIT 1;

```

	staff_name character varying (150) 🔒	transactions_handled bigint 🔒
1	Amy Green	18

Figure 9 Query 8. Staff who processed the most transactions

Q9. Most borrowed genre

```
SELECT g.name AS genre_name, COUNT(b.borrow_id) AS times_borrowed
FROM genre     AS g
JOIN material AS m ON g.genre_id = m.genre_id
JOIN borrow   AS b ON m.material_id = b.material_id
GROUP BY g.name
ORDER BY times_borrowed DESC, g.name
LIMIT 1;
```

	genre_name character varying (100) 🔒	times_borrowed bigint 🔒
1	General Fiction	41

Figure 10 Query 9. Most borrowed genre

Q10. Overdue books (due date passed, not returned)

```
SELECT m.title, b.member_id, b.due_date
FROM borrow AS b
JOIN material AS m ON b.material_id = m.material_id
WHERE b.return_date IS NULL
      AND b.due_date < CURRENT_DATE
ORDER BY b.due_date, m.title;
```

	title character varying (200) 🔒	due_date date 🔒	member_id integer 🔒
1	The Sun Also Rises	2021-11-11	7
2	The Old Man and the Sea	2021-12-20	1
3	Ulysses	2023-01-18	9
4	The Great Gatsby	2023-02-13	1
5	Lolita	2023-03-22	11
6	Brave New World	2023-03-31	12
7	The Sound and the Fury	2023-04-05	13
8	Catch-22	2023-04-15	17
9	Darkness at Noon	2023-04-20	8
10	Sons and Lovers	2023-04-16	9
11	The Grapes of Wrath	2023-04-18	20
12	Les Misérables	2022-06-26	36
13	Oliver Twist	2023-10-29	38
14	The Stranger	2022-07-24	14
15	The Scarlet Letter	2024-08-01	34
16	The Count of Monte Cri...	2023-05-24	37
17	The Brothers Karamazov	2023-09-10	6
18	Of Mice and Men	2024-06-10	14
19	Emma	2022-06-14	15
20	Frankenstein	2023-07-14	8
21	The Lord of the Rings	2022-10-17	11
22	Anna Karenina	2024-05-30	37

Figure 11 Query 10. Overdue books (due date passed, not returned)

5. Notes on Data Quality & Assumptions

- The Publication_Date field was cleaned and standardized to use full date values like 1922 to 1922-01-01) so that PostgreSQL would accept them as valid DATE types.
- Curly quotation marks and other symbols that caused import errors were fixed.
- Each record in the Borrow table represents one borrowing transaction per material, which matches the format and logic used in the provided data files.

6. Conclusion

This stage successfully implemented the library database in PostgreSQL and confirmed that the design from Stage 1 works as intended. All tables were created, data imported, and queries

executed correctly. The system now supports essential library functions—tracking materials, members, borrowing, and staff activities and provides useful insights such as popular genres and overdue items.

7. References (IEEE)

- [1] R. Elmasri and S. B. Navathe, *Fundamentals of Database Systems*, 7th ed. Pearson, 2016.
- [2] CS 504 Course Slides, “Database Implementation & SQL,” 2025