PROJECT REPORT

ON

SQL

(STRUCTURED QUERY LANGUAGE)

Submitted by:

**Ms. Shivani Kosare**

Submitted to:

ARC

TECHNOLOGIES &

INSTITUTIONS



2024

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Contents** | **Page No.** |
| **1** | **Introduction of Dataset** | **3** |
| **2** | **Attributes with their Datatype** | **6** |
| **3** | **Domain and Types** | **7** |
| **4** | **Count of elements** | **9** |
| **5** | **ER Diagram** | **12** |
| **6** | **Problem Statement** |  |
|  | * **Having, group by ,order by** | **13** |
|  | * **Join, having, group by** | **14** |
|  | * **Window function** | **15** |
|  | * **Subquery** | **21** |
|  | * **Join with where** | **22** |
|  | * **3 table join** | **23** |
|  | * **Create a new table using select query** | **23** |

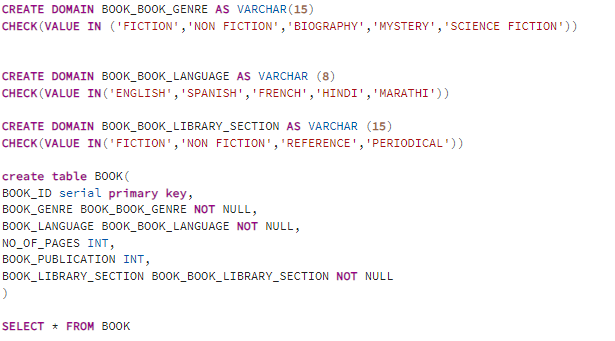
INTRODUCTION OF DATASET

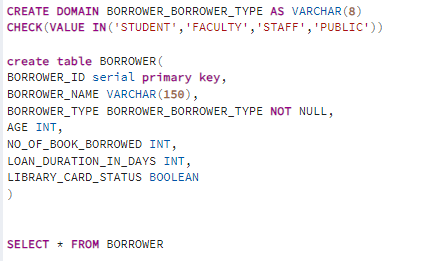
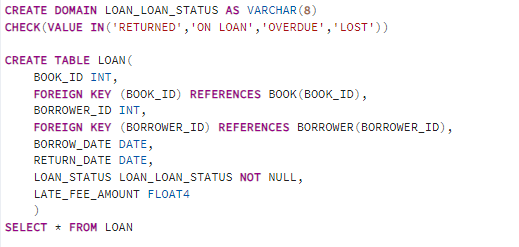
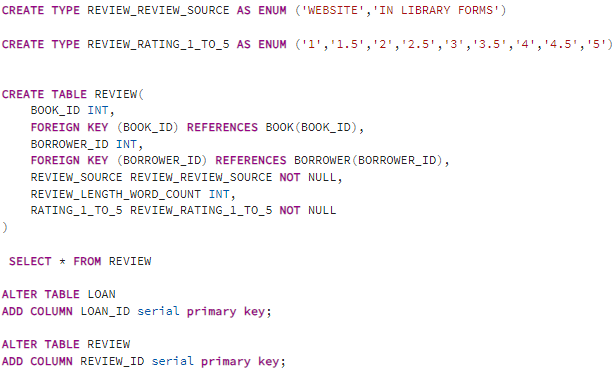
* The dataset is created taking the ideas from gemini and chatgpt.
* The dataset is representing the library management system -Books, Borrowers, loan(Borrowing), Reviewing Activity.
* The dataset majorly contain 4 tables which are as follows:-

**Tables:**

1. **Books:**
   * Book ID
   * **Categorical Data:**
     + Book Genre (Fiction, Non-Fiction, Biography, mystery, science fiction)
     + Book Language (English, Spanish, French, Hindi, Marathi)
     + Library section (Fiction, Non-Fiction, Reference, Periodicals)
   * **Numerical Data:**
     + Number of Pages (Integer)
     + Publication year (Integer)
2. **Borrowers:**
   * Borrower ID
   * Borrower name
   * **Categorical Data:**
     + Borrower Type (Student, Faculty, Staff, Public)
     + Library card status (Boolean)
   * **Numerical Data:**
     + Number of Books Borrowed (Integer, total across all borrows)
     + Loan Duration (Integer, days per borrow)
     + Age (Integer)
3. **Loan(Borrows):**
   * **Foreign Key :** Book ID (references Books table)
   * **Foreign Key :** Borrower ID (references Borrowers table)
   * **Categorical Data:**
     + Loan status (Returned, on-loan, overdue, lost)
   * **Numerical Data:**
     + Borrow Date (Date)
     + Return Date (Date)
     + Late fee amount
4. **Reviews:**
   * **Foreign Key:** Book ID (references Books table)
   * **Foreign Key:** Borrower ID (references Borrowers table)
   * **Categorical Data:**
     + Rating (1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5, 5)
     + Review source (websites, in-library form)
   * **Numerical Data:**
     + Review Length (Integer, word count)

* The data consist of 7 categorical data and 9 numerical data.
* The data is put in the data set with help of insert queries from chatgpt.
* The tables are created with the help of following queries.



Attributes with their Datatype

* BOOK TABLE
* BOOK\_ID:- integer and is primary key
* BOOK\_GENRE:- character varying(15)
* BOOK\_LANGUAGE:- character varying(8)
* NO\_OF\_PAGES:- integer
* BOOK\_PUBLICATION:- integer
* BOOK\_LIBRARY\_SECTION:- character varying(15)
* BORROWER TABLE
* BORROWER\_ID:- integer and is primary key
* BORROWER\_NAME:- character varying(15)
* BORROWER\_TYPE:- character varying(8)
* AGE:- integer
* NO\_OF\_BOOK\_BORROWED:- integer
* LIBRARY\_CARD\_STATUS:- boolean
* LOAN TABLE
* BOOK\_ID:- integer and is a foreign key
* BORROWER\_ID:- integer and is a foreign key
* BORROW\_DATE:- date
* RETURN\_DATE:- date
* LOAN\_STATUS:- character varying(8)
* LATE\_FEE\_AMOUNT:- float4
* REVIEW TABLE
* BOOK\_ID:- integer and is a foreign key
* BORROWER\_ID:- integer and is a foreign key
* REVIEW\_LENGTH\_WORD\_COUNT:- integer

Domain and Types

* BOOK TABLE
* BOOK\_GENRE:- 1. FICTION

2. NON FICTION

3. BIOGRAPHY

4. MYSTERY

5. SCIENCE FICTION

* BOOK\_LANUAGE:- 1. ENGLISH

2. SPANISH

3. FRENCH

4. HINDI

5. MARATHI

* BOOK\_LIBRARY\_SECTION:- 1. FICTION

2. NON FICTION

3. REFERENCE

4. PERIODICAL

* BORROWER TABLE
* BORROWER\_TYPE:- 1. STUDENT

2. FACULTY

3. STAFF

4. PUBLIC

* LOAN TABLE
* LOAN STATUS:- 1. RETURNED

2. ON LOAN

3. OVERDUE

4. LOST

* REVIEW TABLE
* REVIEW\_SOURCE:- 1. WEBSITE

2. IN LIBRARY FORM

* REVIEW\_RATING\_1\_TO\_5:- 1. 1

2. 1.5

3. 2

4. 2.5

5. 3

6. 3.5

7. 4

8. 4.5

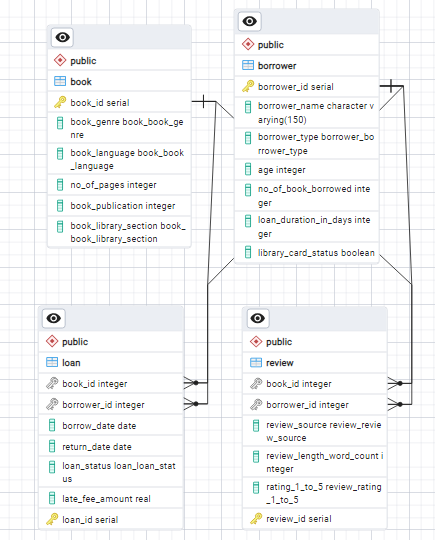
Count of Elements

* BOOK TABLE
* BOOK\_GENRE:-
* BOOK\_LANGUAGE:-
* **DISTINCT COUNT OF NO\_OF\_PAGES = 60**
* **DISTINCT COUNT OF BOOK\_PUBLICATION = 130**

|  |
| --- |
| * BOOK\_LIBRARY\_SECTION:- |
|  |
|  |

* BORROWER TABLE
* BORROWER\_TYPE:-
* **DISTINCT COUNT OF AGE = 33**
* **DISTINCT COUNT OF NO\_OF\_BOOK\_BORROWED = 6**
* **DISTINCT COUNT OF LOAN\_DURATION\_IN\_DAYS = 8**
* **DISTINCT COUNT OF LIBRARY\_CARD\_STATUS = 2**
* LOAN TABLE
* **LOAN STATUS:-**
* **DISTINCT COUNT OF BORROW\_DATE = 369**
* **DISTINCT COUNT OF RETURN\_DATE = 390**
* **DISTINCT COUNT OF LATE\_FEE\_AMOUNT = 174**
* REVIEW TABLE
* REVIEW\_SOURCE:-
* **DISTINCT COUNT OF REVIEW\_LENGTH\_WORD\_COUNT = 115**
* **DISTINCT COUNT OF DISTINCT RATING\_1\_TO\_5 = 8**

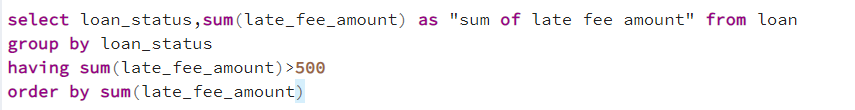
ER Diagram



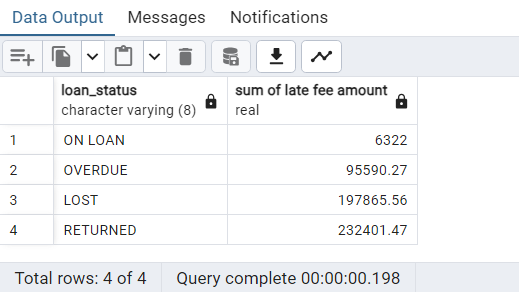
Problem Statement

* Having, group by, order by

Query:-

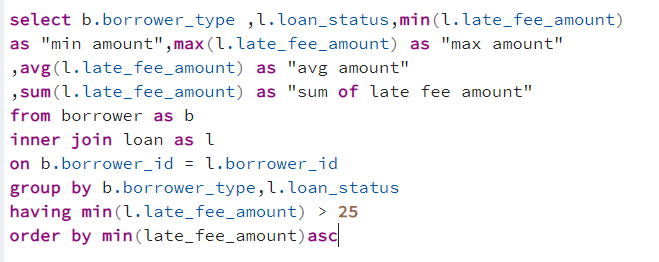


Output:-

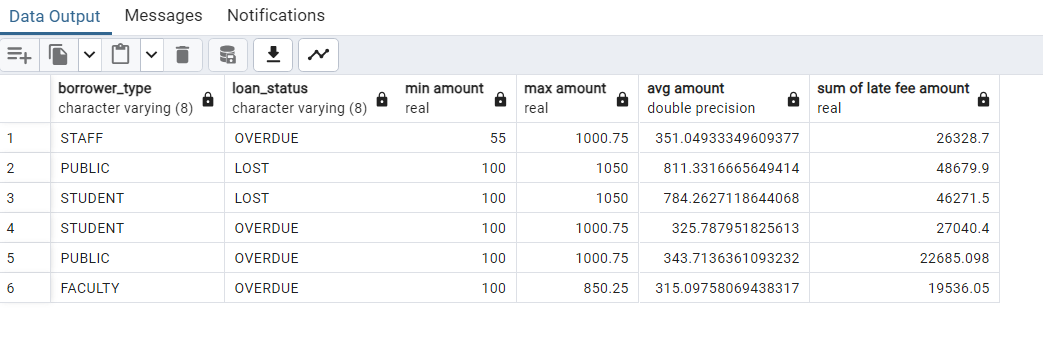


* Join, having, group by

Query:-



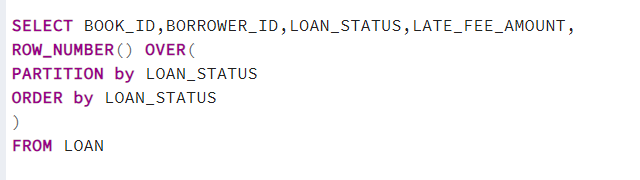
Output:-



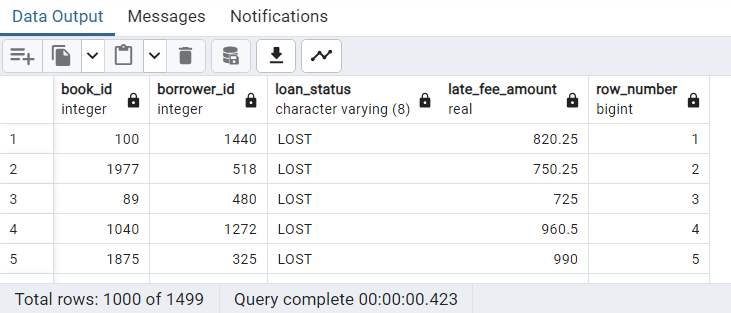
* **Window function**

**1.ROW NUMBER**

**Query:-**

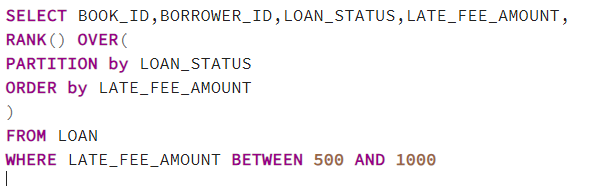
****

**Output:-**

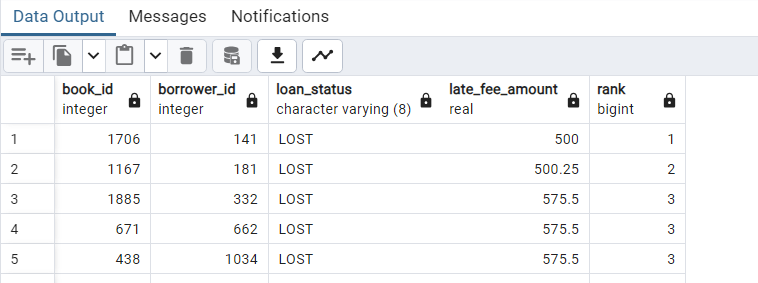
****

**2.RANK**

**Query:-**

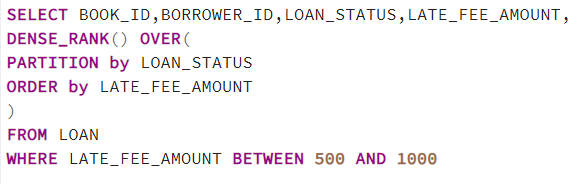
****

**Output:-**

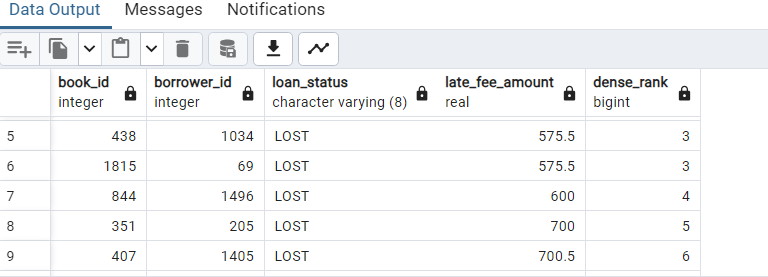
****

**3.DENSE RANK**

**Query:-**

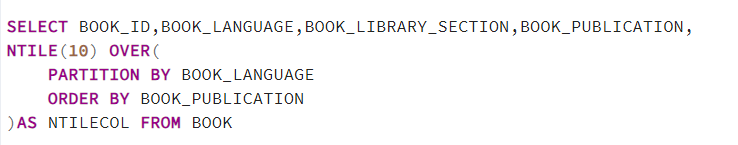
****

**Output:-**

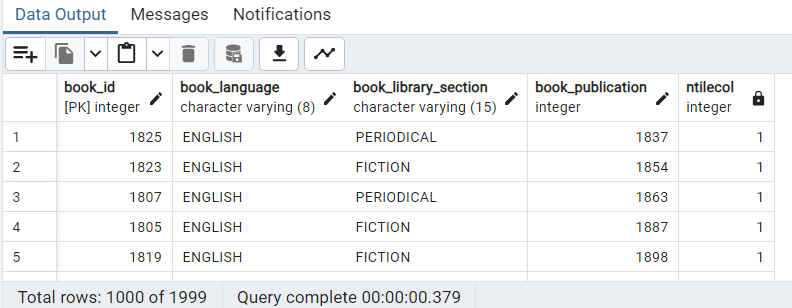
****

**4.NTILE**

**Query:-**

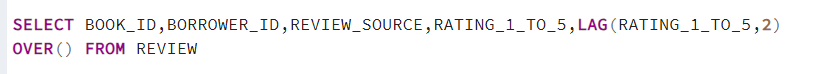
****

**Output:-**

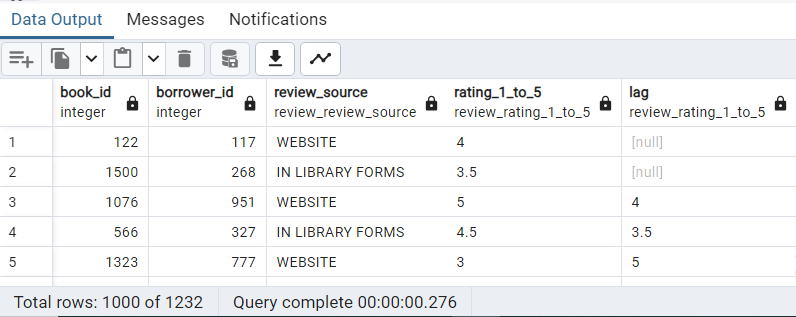
****

**5.LAG**

**Query:-**

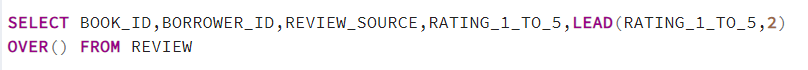
****

**Output:-**

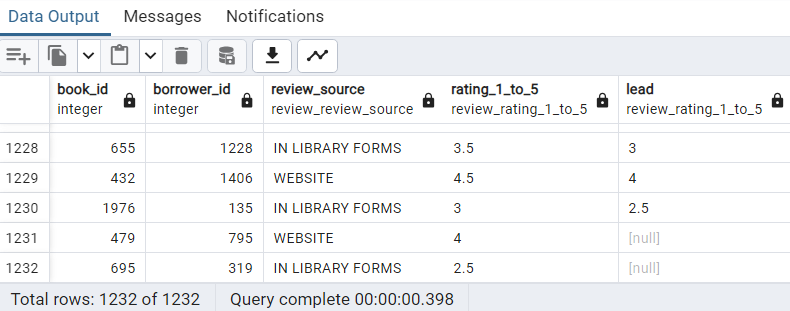
****

**6.LEAD**

**Query:-**

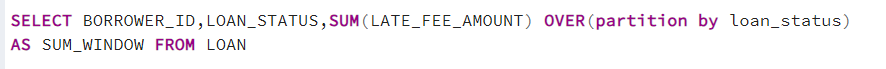
****

**Output:-**

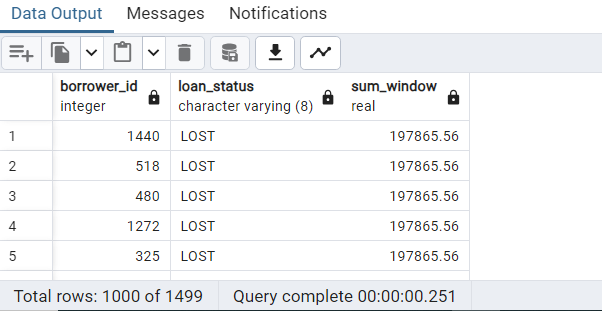
****

**7.SUM**

**Query:-**

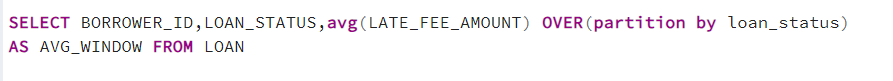
****

**Output:-**

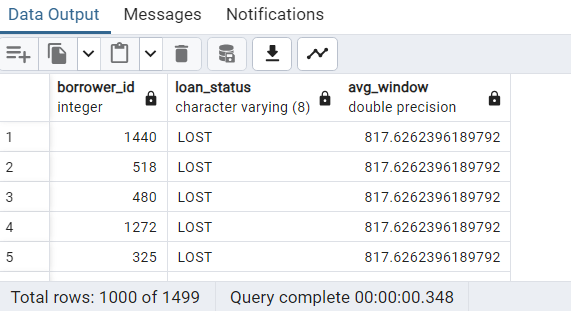
****

**8.AVG**

**Query:-**

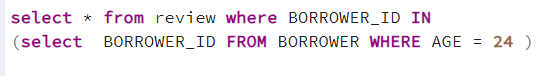
****

**Output:-**

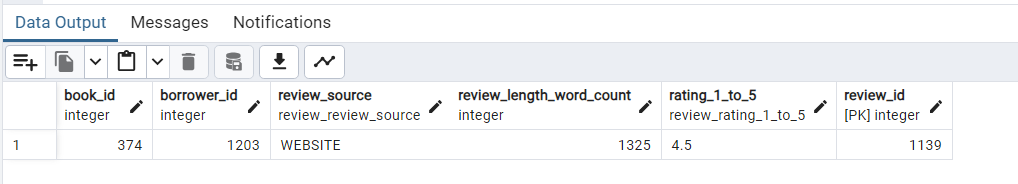
****

* **Subquery**

**Query:-**

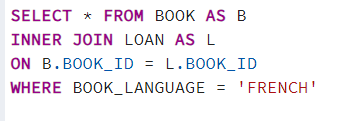
****

**Output:-**

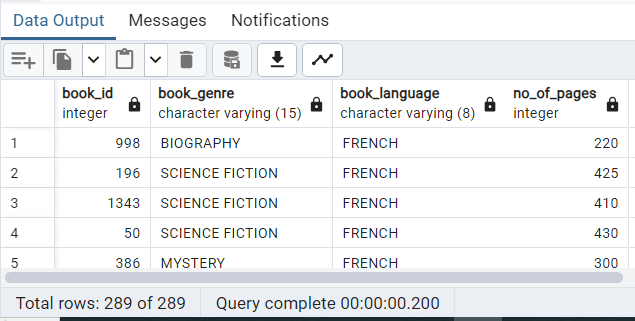
****

* **Join with where**

**Query:-**

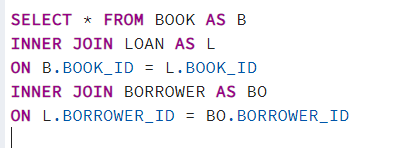
****

**Output:-**

****

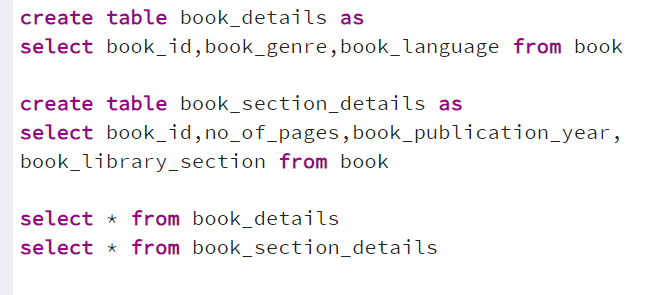
* **3 Table Join**

**Query:-**

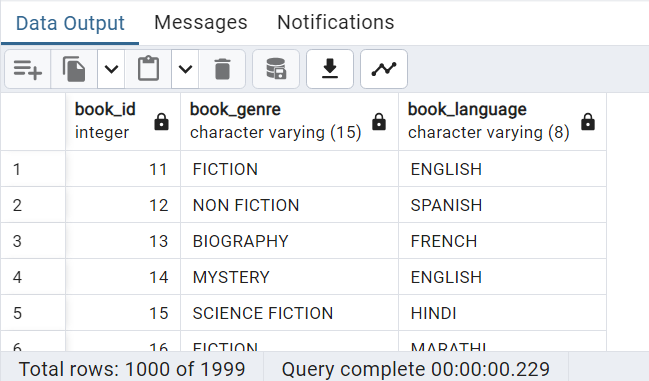
****

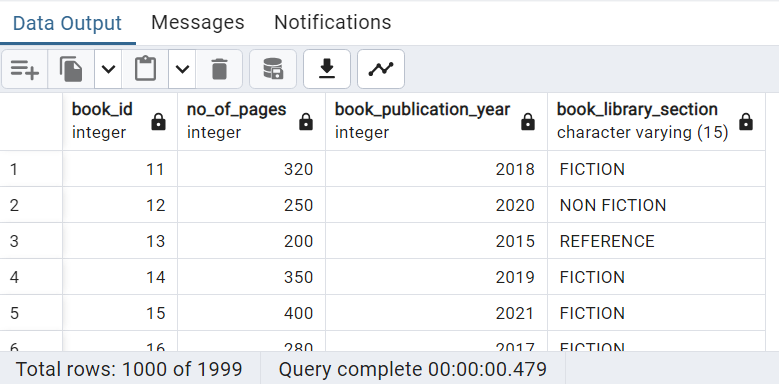
* **Create a new table using select query**

**Query:-**

****

**Output:-**

****

****