NAME: YADAV VISHAL RAMASHANKAR

SUBJECT : DATA BASE MANAGEMENT

*BATCH :* ***JUNE23-T305/DS/SQlEss/SW/4thJuly/3pm-5pm/R***

COURSE : MASTERS IN DATA SCIENCE AND ANALYTICS WITH A.I(ARTIFICIAL INTELLIGENCE)

CONTACT: 9322267388

COURSE MENTOR: MR. SAMEER WARSOLKAR SIR

# PROJECT ON : LIBRARY MANAGEMENT SYSTEM

LIBRARY\_MANAGEMENT\_SYSTEM

ABSTRACT:

Online Library Management System is a system which maintains the information about the

Books present in the library, their authors, the members of library to whom books are issued,

Library staff and all.

This is very difficult to organize manually. Maintenance of all this information manually is a

Very complex task. Owing to the advancement of technology, organization of an Online Library

Becomes much simple.

The Online Library Management has been designed to computerize and automate the operations

Performed over the information about the members, book issues and returns and all other

Operations.

This computerization of library helps in many instances of its maintenances.

It reduces the workload of management as most of the manual work done is reduced

THE PROJECTS AIMS AND OBJECTIVE:

The project aims and objectives that will be achieved after completion of this project are

Discussed in this subchapter. The aims and objectives are as follows:

 Online book reading.

 A search column to search availability of books.

 Facility to download required book.

 Video tutorial for students.

 An Admin login page where admin can add books, videos or page sources

 Open link for Learning Websites

# LIBRARY MANAGEMENT SYSTEM

***Project for SQL Module***

1. **Description:**

Following database schema is designed to function as a backend storage database for a web application built to manage a library.

By storing information in a relational database, all the tasks related to daily functioning of the library can be performed easily and much more efficiently. Some of the benefits of using this system to store data over traditional paper registers are as follows:

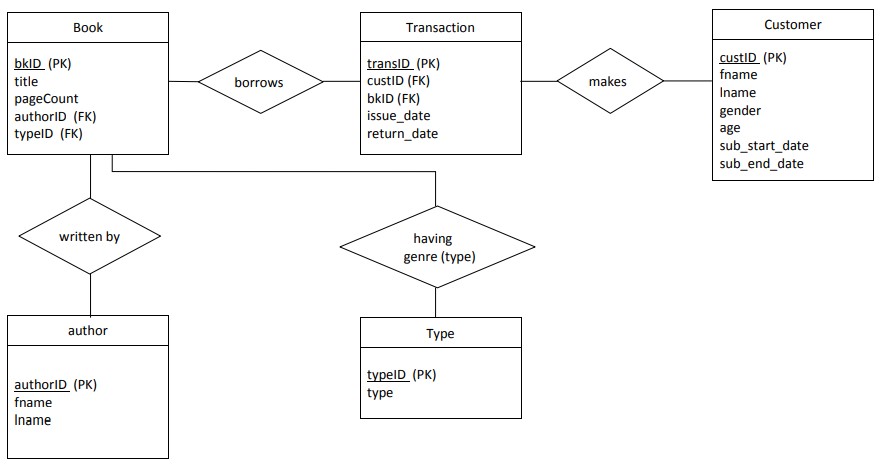
* + Updating and modifying booklist in the library is much easier and efficient
  + Maintaining customer details and their account is easy and efficient.
  + Calculations like due payments can be done automatically by DBMS, thus eliminating human error
  + RDBMS provides many ways to analyze available data, thus helping in making more informed decisions about inventory management and other aspects of library management

This database contains 5 tables:

* 1. Author
  2. Type (Genre of book)
  3. Book
  4. Customer
  5. Transaction

How these tables/entities are related to each other is shown pictorially on next page through ER diagram, i.e., Entity Relationship Diagram.

1. **ER-Diagram (Entity Relation – Diagram) for Library Management System:**



Page **2** of **14**

1. **Table Descriptions: 1. Author:**

+ + + + + + +

| Field | Type | Null | Key | Default | Extra |

+ + + + + + +

|  |  |  |
| --- | --- | --- |
| | authorID | int(11) | NO | PRI | NULL | | | | |
| | fname | varchar(50) | YES | | NULL | | | | |
| | lname | varchar(50) | YES | | NULL | | | | |
| + + + + + | + | + |

* + 1. **Type (Genre):**

+ + + + + + +

| Field | Type | Null | Key | Default | Extra |

+ + + + + + +

|  |  |  |
| --- | --- | --- |
| | typeID | int(11) | NO | PRI | NULL | | | | |
| | type | varchar(50) | YES | | NULL | | | | |
| + + + + + | + | + |

* + 1. **Book:**

+ + + + + + +

| Field | Type | Null | Key | Default | Extra |

+ + + + + + +

|  |  |
| --- | --- |
| | bkID | int(11) | NO | PRI | NULL | | | |
| | title | varchar(50) | YES | | NULL | | | |
| | pageCount | int(11) | YES | | NULL | | | |
| | authorID | int(11) | YES | MUL | NULL | | | |
| | typeID | int(11) | YES | MUL | NULL | | | |
| + + + + + + | + |

* + 1. **Customer:**

+ + + + + + +

| Field | Type | Null | Key | Default | Extra |

+ + + + + + +

|  |  |  |  |
| --- | --- | --- | --- |
| | custID | int(11) | NO | PRI | NULL | | | | | |
| | fname | varchar(30) | |  YES | | NULL | | | | |
| | lname | varchar(30) | |  YES | | NULL | | | | |
| | gender | varchar(5) | |  YES | | NULL | | | | |
| | age | int(3) | YES | | | NULL | | | | |
| | sub\_start\_date | date | YES | | | NULL | | | | |
| | sub\_end\_date | date | YES | | | NULL | | | | |
| + + + + | + | + | + |

* + 1. **Transaction:**

+ + + + + + +

| Field | Type | Null | Key | Default | Extra |

+ + + + + + +

| transID | int(11) | | PRI | NULL | |

NO

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| | custID | | int(11) | YES | | MUL | NULL | | | | |
| | bkID | | int(11) | YES | | MUL | NULL | | | | |
| | issue\_date | | datetime | YES | | | NULL | | | | |
| | return\_date | | datetime | YES | | | NULL | | | | |

1. **Commands:** 
   * **Create Database: lms**
   * **Select Database: lms;**

* + **Create table named author:**
  + **Create table named type:**
  + **Create table named book:**

PRIMARY KEY (bkID),

CONSTRAINT fk\_bk\_author FOREIGN KEY (authorID) REFERENCES author(authorID), CONSTRAINT fk\_bk\_type FOREIGN KEY (typeID) REFERENCES type(typeID);

* + **Create table named customer: Create table named transaction:** CONSTRAINT fk\_trans\_cust

FOREIGN KEY (custID) REFERENCES

customer(custID),

CONSTRAINT fk\_trans\_bk

FOREIGN KEY (bkID)

REFERENCES book(bkID));

* + **Populate “author” table:**

INSERT INTO author

VALUES (1, 'William

Dean', 'Howells'), (2,

'Frederic', 'Brown'),

(3, 'Jack', 'London'),

(4, 'Albert', 'Blaisdell'),

(5, 'Ellis', 'Butler'),

(6, 'Arthur', 'Machen'),

(7, 'Titus', 'Lucretius'),

(8, 'Rabindranath', 'Tagore'),

(9, 'Isaac', 'Asimov'),

(10, 'Charles', 'Dickens'),

(11, 'Ralph Waldo', 'Emerson'),

(12, 'Dorothy', 'Canfield'),

(13, 'Givoanni', 'Boccaccio'),

(14, 'George', 'Orwell'),

(15, 'Publius', 'Ovid'),

(16, 'Robert Louis',

'Stevenson'), (17, 'Virginia',

'Woolf'),

(18, 'George', 'Eliot'),

(19, 'Amelia B.', 'Edwards'), (20, 'Fyodor', 'Dostoevsky'),

(21, 'Emily', 'Dickinson'),

(22, 'Edna', 'Ferber'),

(23, 'Joseph Sheridan',

'LeFanu'), (24, 'John',

'DosPassos'),

(25, 'Ruth', 'Stuart'),

(26, 'Vladimir', 'Nabokov'),

(27, 'Johanna', 'Spyri'),

(28, 'Ernest', 'Dowson'),

(29, 'Mary Hallock', 'Foote'),

(30, 'Zane', 'Grey'),

(31, 'H. P.', 'Lovecraft'),

(32, 'Samuel', 'Pepys'),

(33, 'Kate Dickinson', 'Sweetser'), (34, 'William', 'Lampton'), (35, 'Mother', 'Goose'), (36, 'Eleanor Hallowell', 'Abbott');

• **Populate “type” table:** INSERT into

type VALUES (1,

'Science fiction'),

(2, 'Satire'),

(3, 'Drama'),

(4, 'Action and

Adventure'), (5,

'Romance'),

(6, 'Mystery'),

(7, 'Horror'),

(8, 'Health'),

(9, 'Guide'),

(10, 'Diaries'),

(11, 'Comics'),

(12, 'Diaries'),

(13, 'Journals'),

(14, 'Biographies'),

(15, 'Fantasy'),

(16, 'History'),

(17, 'Science'),

(18, 'Art');

• **Populate “book” table:**

INSERT INTO book VALUES

(1, 'A Daughter of the Snows', 199, 3, 9),

(2, 'The Near East: 10,000 Years of History', 298, 9, 13),

(3, 'The Cocoon: A Rest-Cure Comedy', 90, 25, 12),

(4, 'The Freakshow Murders', 321, 2, 3),

(5, 'Pharaohs, Fellahs and Explorers', 367, 19, 9),

(6, 'Hard Times', 293, 10, 1),

(7, 'A Modern Instance', 222, 1, 12),

(8, 'The Real Mother Goose', 88, 35, 1),

(9, 'A Thousand Miles Up the Nile', 110, 19, 18),

(10, 'Children of Blood and Bone', 137, 7, 13),

(11, 'A pushcart at the curb', 95, 24, 1),

(12, 'The Desert and the Sown', 126, 29, 10),

(13, 'Three Soldiers', 223, 24, 16),

(14, 'The End of Eternity', 168, 9, 1),

(15, 'Annie Kilburn', 291, 1, 12),

(16, 'A Touch of Sun and Other Stories', 141, 29, 9),

(17, 'Show Boat', 151, 22, 18),

(18, 'The Call of the Wild', 362, 3, 18),

(19, 'My Mark Twain', 339, 1, 9),

(20, 'Broken Ties', 134, 8, 5),

(21, 'Short Stories From American History', 305, 4, 8),

(22, 'Mrs Rosie and the Priest', 104, 13, 14),

(23, 'So Big', 209, 22, 3),

(24, 'Monsieur Maurice ', 92, 19, 8),

(25, 'The Master of Ballantrae', 236, 16, 13),

(26, 'The Unlived Life of Little Mary Ellen', 99, 25, 6),

(27, 'Mouse - The Last Train', 184, 2, 14),

(28, 'Edith Bonham', 122, 29, 7),

(29, 'Maybe Mother Goose', 161, 35, 14),

(30, 'The Noble Gases', 321, 9, 5),

(31, 'Rainy Week', 98, 36, 7),

(32, 'A Hazard of New Fortunes', 144, 1, 4),

(33, 'A Plot for Murder', 108, 2, 13),

(34, 'Nature', 357, 11, 8),

(35, 'Hickory Dickory Dock', 326, 35, 6),

(36, 'Big Fat Hen', 232, 35, 12),

(37, 'The Lone Star Ranger', 328, 30, 9),

(38, 'The Great Fire of London', 252, 32, 9),

(39, 'White Nights', 265, 20, 14),

(40, 'In a Glass Darkly', 173, 23, 9),

(41, 'Fanny herself', 248, 22, 10),

(42, 'Loaded', 273, 2, 4),

(43, 'A Foregone Conclusion', 121, 1, 15),

(44, 'The Hill of Dreams', 175, 6, 6),

(45, 'Adventure', 217, 3, 8),

(46, 'A Tagore Reader', 285, 8, 4),

(47, 'A Tale of Two Cities', 331, 10, 13),

(48, 'Dombey and Son', 344, 10, 14),

(49, 'Famous Women', 273, 13, 1),

(50, 'Rejection, The Ruling Spirit', 358, 5, 2);

• **Populate “customer” table:**

INSERT INTO customer VALUES

(1, 'Hazel', 'Green', 'F', 28, '2020-05-15', '2023-05-15'),

(2, 'Ashley', 'Marshall', 'F', 20, '2020-10-28', '2023-10-28'),

(3, 'Ansley', 'Green', 'F', 22, '2021-07-12', '2024-07-12'),

(4, 'Alcock', 'Chapman', 'F', 22, '2020-01-25', '2023-01-25'),

(5, 'Meadow', 'Taylor', 'F', 23, '2020-12-30', '2023-12-30'),

(6, 'Green', 'Wright', 'M', 32, '2020-03-09', '2023-03-09'),

(7, 'Paxton', 'Foster', 'M', 34, '2020-03-03', '2023-03-03'), (8, 'Gray', 'King', 'M', 62, '2020-06-26', '2023-06-26'),

(9, 'Jones', 'Collins', 'M', 25, '2020-01-26', '2023-01-26'),

(10, 'Leslie', 'Young', 'F', 42, '2020-02-10', '2023-02-10'),

(11, 'Hailey', 'Hill', 'F', 27, '2021-01-16', '2024-01-16'),

(12, 'Washington', 'Martin', 'M', 24, '2021-05-23', '2024-05-23'),

(13, 'Ramirez', 'Green', 'M', 33, '2020-08-28', '2023-08-28'),

(14, 'Morgan', 'Smith', 'M', 29, '2021-05-15', '2024-05-15'),

(15, 'Perez', 'Baker', 'M', 53, '2020-10-27', '2023-10-27'),

(16, 'Alfie', 'Watson', 'M', 26, '2021-04-05', '2024-04-05'),

(17, 'Madison', 'Jackson', 'F', 35, '2021-02-15', '2024-02-15'),

(18, 'Beverly', 'Moore', 'F', 22, '2020-11-19', '2023-11-19'),

(19, 'Kaden', 'Anderson', 'M', 26, '2021-11-11', '2024-11-11'),

(20, 'Carter', 'Adams', 'M', 27, '2021-07-08', '2024-07-08'),

(21, 'Ackland', 'Adams', 'F', 28, '2021-05-19', '2024-05-19'),

(22, 'Hadley', 'Parker', 'F', 37, '2020-11-21', '2023-11-21'),

(23, 'Henderson', 'Cook', 'M', 38, '2021-08-06', '2024-08-06'),

(24, 'Wright', 'Young', 'M', 35, '2021-05-14', '2024-05-14'),

(25, 'Lindsey', 'Wilkinson', 'F', 28, '2021-10-08', '2024-10-08'),

(26, 'Wren', 'Chapman', 'F', 25, '2021-11-06', '2024-11-06'),

(27, 'Love', 'Chapman', 'F', 19, '2021-10-15', '2024-10-15'),

(28, 'Wes', 'Bell', 'M', 23, '2021-01-25', '2024-01-25'),

(29, 'Tyler', 'Carter', 'M', 27, '2021-08-24', '2024-08-24'),

(30, 'Edwin', 'Cook', 'M', 39, '2021-08-17', '2024-08-17'),

(31, 'Sally', 'Wood', 'F', 45, '2020-06-07', '2023-06-07'),

(32, 'Nelson', 'Smith', 'M', 27, '2020-11-15', '2023-11-15'),

(33, 'Arabelle', 'Brown', 'F', 52, '2021-03-11', '2024-03-11'),

(34, 'Carter', 'Harrison', 'M', 21, '2021-11-30', '2024-11-30'),

(35, 'Baker', 'Bell', 'M', 20, '2021-03-24', '2024-03-24'),

(36, 'Cherish', 'Cooper', 'F', 34, '2020-01-31', '2023-01-31'),

(37, 'Remington', 'Wright', 'M', 29, '2020-08-14', '2023-08-14'),

(38, 'Edwards', 'King', 'M', 24, '2020-02-01', '2023-02-01'),

(39, 'Harleigh', 'Taylor', 'F', 26, '2020-05-05', '2023-05-05'),

(40, 'Ryleigh', 'Clark', 'F', 45, '2020-10-29', '2023-10-29'),

(41, 'Albert', 'Richardson', 'F', 57, '2021-03-19', '2024-03-19'),

(42, 'Lee', 'Ward', 'M', 36, '2020-05-13', '2023-05-13'),

(43, 'Kaylin', 'Bell', 'F', 28, '2020-09-12', '2023-09-12'),

(44, 'Agard', 'Young', 'F', 23, '2021-04-03', '2024-04-03'), (45, 'Turner', 'Hall', 'M', 24, '2021-01-07', '2024-01-07'),

(46, 'Akes', 'White', 'F', 20, '2021-03-08', '2024-03-08'),

(47, 'Lewis', 'Lee', 'M', 35, '2021-01-30', '2024-01-30'),

(48, 'Karson', 'Cook', 'M', 30, '2021-06-26', '2024-06-26'),

(49, 'Hayden', 'Young', 'M', 31, '2020-12-15', '2023-12-15'),

(50, 'Sterling', 'Shaw', 'M', 26, '2021-06-24', '2024-06-24');

• **Populate “transaction” table:**

INSERT INTO transaction VALUES

(1, 37, 24, '2021-08-31 04:30:00', '2021-09-17 00:15:00'),

(2, 32, 17, '2021-10-01 14:28:00', '2021-10-07 11:22:00'),

(3, 40, 29, '2021-10-02 10:45:00', '2021-10-10 17:16:00'),

(4, 36, 44, '2021-10-03 15:57:00', '2021-10-09 10:43:00'),

(5, 11, 16, '2021-10-07 13:19:00', '2021-10-15 18:55:00'),

(6, 29, 34, '2021-10-09 08:32:00', '2021-10-14 08:17:00'), (7, 23, 26, '2021-10-09 15:48:00', '2021-10-21 10:51:00'),

(8, 3, 34, '2021-10-10 19:55:00', '2021-10-15 09:20:00'),

(9, 37, 50, '2021-10-14 10:25:00', '2021-10-14 12:52:00'),

(10, 36, 39, '2021-10-16 08:53:00', '2021-10-29 18:47:00'),

(11, 47, 14, '2021-10-24 10:50:00', '2021-11-05 18:19:00'),

(12, 32, 30, '2021-10-24 13:48:00', '2021-10-28 19:33:00'),

(13, 7, 33, '2021-10-27 08:10:00', '2021-11-10 14:34:00'),

(14, 32, 12, '2021-10-27 09:27:00', '2021-11-13 18:23:00'),

(15, 44, 19, '2021-10-29 14:39:00', '2021-11-07 18:34:00'),

(16, 11, 1, '2021-11-03 15:11:00', '2021-11-04 19:42:00'),

(17, 15, 3, '2021-11-06 11:51:00', '2021-11-12 11:49:00'),

(18, 4, 36, '2021-11-06 16:24:00', '2021-11-11 14:00:00'),

(19, 8, 44, '2021-11-12 14:02:00', '2021-11-13 15:21:00'),

(20, 33, 21, '2021-11-14 20:35:00', '2021-11-30 07:56:00'),

(21, 23, 18, '2021-11-16 07:40:00', '2021-11-24 11:57:00'),

(22, 13, 29, '2021-11-17 18:29:00', '2021-11-18 11:27:00'),

(23, 42, 31, '2021-11-19 18:54:00', '2021-12-05 14:59:00'),

(24, 11, 48, '2021-11-20 17:26:00', '2021-11-26 08:16:00'), (25, 29, 42, '2021-11-23 12:20:00', '2021-11-24 10:13:00'), (26, 37, 48, '2021-11-24 12:45:00', '2021-11-27 10:13:00'),

(27, 7, 20, '2021-11-26 19:55:00', '2021-12-03 15:55:00'),

(28, 39, 31, '2021-12-02 12:51:00', '2021-12-17 08:13:00'),

(29, 49, 17, '2021-12-14 10:11:00', '2021-12-21 09:41:00'),

(30, 37, 3, '2021-12-14 12:11:00', '2021-12-14 12:13:00'),

(31, 1, 49, '2021-12-15 15:21:00', '2021-12-17 09:45:00'),

(32, 49, 10, '2021-12-16 11:16:00', '2022-01-02 13:42:00'),

(33, 34, 26, '2021-12-19 17:38:00', '2021-12-30 09:59:00'),

(34, 21, 28, '2021-12-21 11:05:00', '2022-01-06 17:13:00'),

(35, 31, 14, '2021-12-22 09:48:00', '2022-01-06 16:37:00'),

(36, 10, 20, '2021-12-22 14:12:00', '2021-12-22 16:44:00'), (37, 25, 50, '2021-12-22 15:27:00', '2021-12-29 08:56:00'),

(38, 25, 32, '2021-12-22 19:01:00', '2022-01-04 11:16:00'),

(39, 11, 24, '2021-12-26 19:09:00', '2022-01-11 19:14:00'),

(40, 45, 31, '2021-12-28 19:35:00', '2022-01-07 09:41:00'),

(41, 36, 17, '2022-01-05 12:30:00', '2022-01-08 17:51:00'),

(42, 21, 16, '2022-01-07 12:04:00', '2022-01-12 16:31:00'),

(43, 27, 47, '2022-01-11 16:47:00', '2022-01-11 17:20:00'),

(44, 1, 23, '2022-01-12 08:13:00', '2022-01-22 13:34:00'),

(45, 11, 1, '2022-01-14 10:22:00', '2022-01-26 13:43:00'),

(46, 5, 20, '2022-01-14 13:19:00', '2022-01-22 10:37:00'),

(47, 22, 14, '2022-01-16 18:40:00', '2022-02-02 12:10:00'),

(48, 14, 14, '2022-01-19 13:28:00', '2022-01-23 19:36:00'), (49, 45, 9, '2022-01-27 12:06:00', '2022-02-02 15:06:00'),

(50, 20, 13, '2022-01-27 13:47:00', '2022-02-10 11:31:00'), (51, 24, 18, '2022-01-29 15:54:00', '2022-02-02 08:59:00'),

(52, 6, 4, '2022-01-30 15:25:00', '2022-02-04 09:19:00'),

(53, 39, 2, '2022-01-30 17:01:00', '2022-02-13 11:13:00'),

(54, 47, 26, '2022-02-02 11:02:00', '2022-02-12 11:36:00'), (55, 39, 47, '2022-02-11 08:55:00', '2022-02-24 12:40:00'), (56, 23, 35, '2022-02-12 17:59:00', '2022-02-24 08:56:00'),

(57, 35, 1, '2022-02-13 19:07:00', '2022-02-17 11:01:00'),

(58, 35, 1, '2022-02-16 07:08:00', '2022-02-17 11:14:00'),

(59, 50, 3, '2022-02-26 17:54:00', '2022-03-16 11:53:00'),

(60, 15, 12, '2022-02-27 15:38:00', '2022-03-11 14:44:00');

/\* to view all above created tables\*/

select\*from transaction; select\*from author; select\*from type; select\*from book;

select\*from customer;

/\*1. Select customers who have not borrowed any book from library\*/

select\*from customer; select\*from transaction; select customer.fname, customer.custID, transaction.bkID from customer left join transaction

on customer.custID=transaction.cust\_id;

select customer.custID, concat(customer.fname," " ,customer.lname) as customer\_full\_name from customer left join transaction

on customer.custID=transaction.cust\_id where bkID is null;

/\*2. Select book with highest page count\*/

select\*from book;

select max(page\_count) from book;

select bkid, title, page\_count from book where page\_count = (select max(page\_count) from book);

/\*3. Select customer who has borrowed some book for longest period\*/

select\*from customer;

select\*from transaction;

select customer.custID,

concat(customer.fname," " ,customer.lname) as customer\_full\_name,

MAX(DATEDIFF(Transaction .return\_date, Transaction .issue\_date)) AS MaxBorrowedDays from customer join transaction

on customer.custID=transaction.cust\_id group by customer.custID order by MaxBorrowedDays DESC

LIMIT 1;

/\*4.display list of all books along with their authors\*/

of

select\*from author; select\*from book;

select book.bkid, book.title, concat(author.author\_first\_name," ", author.author\_last\_name) as author from book join author on author.author\_id= book.author\_id;

/\*5.Display list of top 20 most borrowed books\*/

select\*from book;

select\*from transaction;

select transaction.bkID, book.title, count(transaction.trans\_id) as times\_borrowed from book join transaction on transaction.bkID=book.bkid GROUP BY book.title

ORDER BY times\_borrowed desc

limit 20 ;

/\*6.Display list of top 10 customers who have borrowed most book\*/

select\*from transaction;

select\*from customer;

select customer.custID, concat( customer.fname," ",customer.lname) as name, count(transaction.trans\_id) as book\_borrowed from customer join transaction on transaction.cust\_id=customer.custID GROUP BY customer.custID, name

ORDER BY book\_borrowed DESC LIMIT 10;

of

**5. Sub-queries: 1. Select customers who have not borrowed any book from library**

**Ans=** select customer.custID, concat(customer.fname," " ,customer.lname) as customer\_full\_name from customer left join transaction

on customer.custID=transaction.cust\_id

• where bkID is null;

**RESULT: I GOT RESULT AS:**

|  |  |  |  |
| --- | --- | --- | --- |
| **custID** | **name** | custID | customer\_full\_name |
| 2 | Ashley Marshall | 2 | Ashley Marshall |
| 9 | Jones Collins | 9 | Jones Collins |
| 12 | Washington Martin | 12 | Washington Martin |
| 16 | Alfie Watson | 16 | Alfie Watson |
| 17 | Madison Jackson | 17 | Madison Jackson |
| 18 | Beverly Moore | 18 | Beverly Moore |
| 19 | Kaden Anderson | 19 | Kaden Anderson |
| 26 | Wren Chapman | 26 | Wren Chapman |
| 28 | Wes Bell | 28 | Wes Bell |
| 30 | Edwin Cook | 30 | Edwin Cook |
| 38 | Edwards King | 38 | Edwards King |
| 41 | Albert Richardson | 41 | Albert Richardson |
| 43 | Kaylin Bell | 43 | Kaylin Bell |
| 46 | Akes White | 46 | Akes White |
| 48 | Karson Cook | 48 | Karson Cook |

**2. VSelect book with highest page count**

**RESULT**

|  |  |
| --- | --- |
| **bkID** | **title** |
| 5 | Pharaohs, Fellahs and Explorers |

ANS:

select max(page\_count) from book;

select bkid, title, page\_count

from book where

|  |  |  |  |
| --- | --- | --- | --- |
| bkid |  | title | page\_count |
|  | 5 | Pharaohs, Fellahs and Explorers | 367 |

page\_count = (select max(page\_count) from book);

I GOT RESULT AS:

select max(page\_count) from book;

|  |
| --- |
| max(page\_count) |
| 367 |

select bkid, title, page\_count

from book where

page\_count = (select max(page\_count) from book);

**3. Select customer who has borrowed some book for longest period**

**RESULT:**

|  |  |
| --- | --- |
| **custID** | **name** |
| 50 | Sterling Shaw |

ANS =

select customer.custID,

concat(customer.fname," " ,customer.lname) as customer\_full\_name,

MAX(DATEDIFF(Transaction .return\_date, Transaction

.issue\_date))

AS MaxBorrowedDays from customer join transaction

on customer.custID=transaction.cust\_id group by customer.custID order by MaxBorrowedDays DESC LIMIT 1;

I GOT RESULT AS:

custID customer\_full\_name MaxBorrowedDays

|  |
| --- |
| Sterling Shaw |

50 18

**6. JOINS:**

**1. Display list of all books along with their authors**

**RESULT:**

|  |  |  |
| --- | --- | --- |
| **bkID** | **title** | **author** |
| 1 | A Daughter of the Snows | Jack London |
| 2 | The Near East: 10,000 Years of History | Isaac Asimov |
| 3 | The Cocoon: A Rest-Cure Comedy | Ruth Stuart |
| 4 | The Freakshow Murders | Frederic Brown |
| 5 | Pharaohs, Fellahs and Explorers | Amelia B. Edwards |
| 6 | Hard Times | Charles Dickens |
| 7 | A Modern Instance | William Dean Howells |
| 8 | The Real Mother Goose | Mother Goose |
| 9 | A Thousand Miles Up the Nile | Amelia B. Edwards |
| 10 | Children of Blood and Bone | Titus Lucretius |
| 11 | A pushcart at the curb | John DosPassos |
| 12 | The Desert and the Sown | Mary Hallock Foote |
| 13 | Three Soldiers | John DosPassos |
| 14 | The End of Eternity | Isaac Asimov |
| 15 | Annie Kilburn | William Dean Howells |
| 16 | A Touch of Sun and Other Stories | Mary Hallock Foote |
| 17 | Show Boat | Edna Ferber |
| 18 | The Call of the Wild | Jack London |
| 19 | My Mark Twain | William Dean Howells |
| 20 | Broken Ties | Rabindranath Tagore |
| 21 | Short Stories From American History | Albert Blaisdell |
| 22 | Mrs Rosie and the Priest | Givoanni Boccaccio |
| 23 | So Big | Edna Ferber |
| 24 | Monsieur Maurice | Amelia B. Edwards |
| 25 | The Master of Ballantrae | Robert Louis Stevenson |
| 26 | The Unlived Life of Little Mary Ellen | Ruth Stuart |
| 27 | Mouse - The Last Train | Frederic Brown |
| 28 | Edith Bonham | Mary Hallock Foote |
| 29 | Maybe Mother Goose | Mother Goose |
| 30 | The Noble Gases | Isaac Asimov |
| 31 | Rainy Week | Eleanor Hallowell Abbott |

|  |  |  |
| --- | --- | --- |
| 32 | A Hazard of New Fortunes | William Dean Howells |
| 33 | A Plot for Murder | Frederic Brown |
| 34 | Nature | Ralph Waldo Emerson |
| 35 | Hickory Dickory Dock | Mother Goose |
| 36 | Big Fat Hen | Mother Goose |
| 37 | The Lone Star Ranger | Zane Grey |
| 38 | The Great Fire of London | Samuel Pepys |
| 39 | White Nights | Fyodor Dostoevsky |
| 40 | In a Glass Darkly | Joseph Sheridan LeFanu |
| 41 | Fanny herself | Edna Ferber |
| 42 | Loaded | Frederic Brown |
| 43 | A Foregone Conclusion | William Dean Howells |
| 44 | The Hill of Dreams | Arthur Machen |
| 45 | Adventure | Jack London |
| 46 | A Tagore Reader | Rabindranath Tagore |
| 47 | A Tale of Two Cities | Charles Dickens |
| 48 | Dombey and Son | Charles Dickens |
| 49 | Famous Women | Givoanni Boccaccio |
| 50 | Rejection, The Ruling Spirit | Ellis Butler |

**ANS:**

select book.bkid, book.title, concat(author.author\_first\_name," ", author.author\_last\_name)

as author from book join author

on author.author\_id= book.author\_id;

I GOT RESULT AS:



**2. Display list of top 20 most borrowed books**

**RESULT:**

|  |  |  |
| --- | --- | --- |
| **bkID** | **title** | **times\_borrowed** |
| 1 | A Daughter of the Snows | 4 |
| 14 | The End of Eternity | 4 |
| 3 | The Cocoon: A Rest-Cure Comedy | 3 |
| 17 | Show Boat | 3 |
| 20 | Broken Ties | 3 |
| 26 | The Unlived Life of Little Mary Ellen | 3 |
| 31 | Rainy Week | 3 |
| 12 | The Desert and the Sown | 2 |
| 16 | A Touch of Sun and Other Stories | 2 |
| 18 | The Call of the Wild | 2 |
| 24 | Monsieur Maurice | 2 |
| 29 | Maybe Mother Goose | 2 |
| 34 | Nature | 2 |
| 44 | The Hill of Dreams | 2 |
| 47 | A Tale of Two Cities | 2 |
| 48 | Dombey and Son | 2 |
| 50 | Rejection, The Ruling Spirit | 2 |
| 2 | The Near East: 10,000 Years of History | 1 |
| 4 | The Freakshow Murders | 1 |
| 9 | A Thousand Miles Up the Nile | 1 |

ANS:

select transaction.bkID, book.title, count(transaction.trans\_id) as times\_borrowed from book join transaction on transaction.bkID=book.bkid GROUP BY book.title

ORDER BY times\_borrowed desc

limit 20 ;

I GOT RESULT AS:

|  |  |  |
| --- | --- | --- |
| bkID | title | times\_borrowed |
| 14 | The End of Eternity | 4 |
| 1 | A Daughter of the Snows | 4 |
| 20 | Broken Ties | 3 |
| 3 | The Cocoon: A Rest-Cure Comedy | 3 |
| 17 | Show Boat | 3 |
| 31 | Rainy Week | 3 |
| 26 | The Unlived Life of Little Mary Ellen | 3 |
| 44 | The Hill of Dreams | 2 |
| 29 | Maybe Mother Goose | 2 |
| 47 | A Tale of Two Cities | 2 |
| 48 | Dombey and Son | 2 |
| 50 | Rejection, The Ruling Spirit | 2 |
| 24 | Monsieur Maurice | 2 |
| 12 | The Desert and the Sown | 2 |
| 18 | The Call of the Wild | 2 |
| 34 | Nature | 2 |
| 16 | A Touch of Sun and Other Stories | 2 |
| 49 | Famous Women | 1 |
| 30 | The Noble Gases | 1 |
| 9 | A Thousand Miles Up the Nile | 1 |

**3. Display list of top 10 customers who have borrowed most books**

**RESULT:**

|  |  |  |
| --- | --- | --- |
| **custID** | **name** | **books\_borrowed** |
| 11 | Hailey Hill | 5 |
| 37 | Remington Wright | 4 |
| 23 | Henderson Cook | 3 |
| 32 | Nelson Smith | 3 |
| 36 | Cherish Cooper | 3 |
| 39 | Harleigh Taylor | 3 |
| 49 | Hayden Young | 2 |
| 7 | Paxton Foster | 2 |
| 15 | Perez Baker | 2 |
| 21 | Ackland Adams | 2 |

ANS :

select customer.custID, concat( customer.fname," ",customer.lname) as name, count(transaction.trans\_id) as book\_borrowed from customer join transaction on transaction.cust\_id=customer.custID GROUP BY customer.custID, name

ORDER BY book\_borrowed DESC

LIMIT 10;

I GOT RESULT AS:

|  |  |  |
| --- | --- | --- |
| custID | name | book\_borrowed |
| 11 | Hailey Hill | 5 |
| 37 | Remington Wright | 4 |
| 39 | Harleigh Taylor | 3 |
| 32 | Nelson Smith | 3 |
| 36 | Cherish Cooper | 3 |
| 23 | Henderson Cook | 3 |
| 49 | Hayden Young | 2 |
| 1 | Hazel Green | 2 |
| 21 | Ackland Adams | 2 |
| 25 | Lindsey Wilkinson | 2 |

# ꧁乂✰ THANK YOU ✰乂꧂