Library Management System

Description:

- Project Title: Library Management System using MySQL server.
- Database: I get this database from free-open source.
- Project Objectives:

This project demonstrates the implementation of the "Library Management System using MySQL server." It includes creating and managing tables, performing crud operations and executing intermediate SQL queries. Exercise and practice skills in database design, manipulation and querying. Transform data into meaningful and actionable insights to help library management team to improve performance and overall management service.

Approach:

- In this project, I am using MySQL Workbench as tool to analyse Library management data and answer questions posed by the management team.
- Here, I use SQL queries to extract the required information from the database, also ensuring the accuracy and efficiency of the queries & the result-data solutions.
- The Final insights will help the product manager and the rest of the team to make informed decisions about the future direction of the Library Management System & service improvement.
- Approach used to complete this Project:

    1) CRUD operation: I perform different CRUD operations on database as per project task requirement. Here, I get hand on experience about different SQL commands, which help to manage and manipulate data as per task requirement. This will help me to maintain the Library Management System updated.

    2) Data Analysis and Insights: I perform analysis operations on the data from given database and solve given task by the management team. Here, I get a hand on experience to transform data into meaningful & actionable insights to help team to improve performance of library management and services, deliver to customers.

- Project Solution: [writing a query to find solution for given problem statement.]

  [A]. CRUD Operations:

  Task 1. Create a New Book Record:

  "978-1-60129-456-2', 'To Kill a Mockingbird', 'Classic', 6.00, 'yes', 'Harper Lee', 'J.B. Lippincott & Co.')"

```
-- Task 1. Create a New Book Record:
--                 "978-1-60129-456-2', 'To Kill a Mockingbird', 'Classic', 6.00, 'yes',
--                     'Harper Lee', 'J.B. Lippincott & Co.')"
select * from books;

insert into books( isbn, book_title, category, rental_price, status, author, publisher)
values ('978-1-60129-456-2', 'To Kill a Mockingbird', 'Classic', 6.00, 'yes',
        'Harper Lee', 'J.B. Lippincott & Co.');
```

Task 2: Update an Existing Member's Address:

```
--      >>> Task 2: Update an Existing Member's Address

select* from members;

update members
set member_address = '007 bond st'
where member_id = 'C101';
```

Task 3: Delete a Record from the Issued Status Table

Objective: Delete the record with issued_id = 'IS110' from the issued_status table.

```
--      >>> Task 3: Delete a Record from the Issued Status Table

-- Objective: Delete the record with issued_id = 'IS110' from the issued_status table.

select* from issued_status;

delete from issued_status
where issued_id = 'IS110';
```

Task 4: Retrieve All Books Issued by a Specific Employee

Objective: Select all books issued by the employee with emp_id = 'E101'.

```
--          >>> Task 4: Retrieve All Books Issued by a Specific Employee

--      Objective: Select all books issued by the employee with emp_id = 'E101'.

select* from issued_status;

select emp_id, emp_name, issued_book_name
from employees emp
join
issued_status ist on emp.emp_id = ist.issued_emp_id
where emp_id = 'E101';
```

Task 5: List Members Who Have Issued More Than One Book

Objective: Use GROUP BY to find members who have issued more than one book.

```
--          >>> Task 5: List Members Who Have Issued More Than One Book
--      Objective: Use GROUP BY to find members who have issued more than one book.

select ist.issued_member_id, mbrs.member_name, count(ist.issued_book_isbn) as book_count
from issued_status ist
join
members mbrs on ist.issued_member_id = mbrs.member_id
group by ist.issued_member_id, mbrs.member_name
having book_count > 1
order by book_count desc;
```

Task 6: Create Summary Tables:

Used CTA'S to generate new tables based on query results- "each book and total book issued cunt"

```
--          >>> Task 6: Create Summary Tables:
--      Used CTA'S to generate new tables
--      based on query results- "each book and total book_issued_count"

create table Book_Issue_count as
with base as(
        select ist.issued_book_name, count(ist.issued_id) as book_issue_count
        from books b
        join issued_status ist on b.isbn =  ist.issued_book_isbn
        group by ist.issued_book_name
)
select issued_book_name, book_issue_count
from base
order by book_issue_count desc;

select * from Book_issue_count;     -- call the our created CTC Table, by Select-command.
```

[B]. Data Analysis & Findings:

Task 7: Retrieve All Books in a Specific Category:

```
--          >>> Task 8 : Find Total Rental Income by Category:

select bk.category, sum(bk.rental_price) as Rental_Income
from books bk
join issued_status ist on bk.isbn = ist.issued_book_isbn
group by bk.category
order by category;
```

Task 8: Find Total Rental Income by Category:

```
--          >>> Task 8 : Find Total Rental Income by Category:

select bk.category, sum(bk.rental_price) as Rental_Income
from books bk
join issued_status ist on bk.isbn = ist.issued_book_isbn
group by bk.category
order by category;
```

Task 9: List Members Who Registered in the Last 2 years:

```
--          >>> Task 9 : List Members Who Registered in the Last 2 years:

select *
from members
where reg_date >= (select current_date - interval '2' year)
order by reg_date;
```

Task 10: List Employees with Their Branch Manager's Name and their branch details:

```
--          >>> Task 10 :
--      List Employees with Their Branch Manager's Name and their branch details**:

select
    emp.emp_id, emp.emp_name,
    br.manager_id, emp2.emp_name as manager_name,
    br.branch_id, br.branch_address, br.contact_no
from employees emp
join branch br
    on emp.branch_id = br.branch_id
join employees emp2
    on br.manager_id = emp2.emp_id
order by emp_id;
```

Task 11: Create a Table of Books with Rental Price Above a Certain Threshold:

i.e. [avg. of total Rental Price]

```
--          >>> Task 11 :
--      Create a Table of Books with Rental Price Above a Certain Threshold:
--          i.e. [avg. of total Rental Price]

select bk.category, sum(bk.rental_price) as Rental_Income
from books bk
join issued_status ist on bk.isbn = ist.issued_book_isbn
where rental_price > (select avg(bk.rental_price) from books bk)
group by bk.category
order by Rental_Income desc;
```

Tech-Stack Used:

Administration – Server Status:

MySQL [Sever] Workbench – local community version.

version – 8.0.41 (MySQL Community Server GPL)

Insights:

Insights & Result-Solution for given problem Statements:

[A]. CRUD Operations:

Task 1. Create a New Book Record:

"978-1-60129-456-2', 'To Kill a Mockingbird', 'Classic', 6.00, 'yes', 'Harper Lee', 'J.B. Lippincott & Co.')"

isbn	book_title	category	rental_price	status	author	publisher
978-1-60129-456-2	To Kill a Mockingbird	Classic	6.00	yes	Harper Lee	J.B. Lippincott & Co.
NULL	NULL	NULL	NULL	NULL	NULL	NULL

Task 2: Update an Existing Member_address ---> “123 Main St” -to- “007 bond st” – [@-Row C101]:

member_id	member_name	member_address	reg_date
C101	Alice Johnson	123 Main St	2021-05-15
C102	Bob Smith	456 Elm St	2021-06-20
C103	Carol Davis	789 Oak St	2021-07-10
C104	Dave Wilson	567 Pine St	2021-08-05
C105	Eve Brown	890 Maple St	2021-09-25
C106	Frank Thomas	234 Cedar St	2021-10-15
C107	Grace Taylor	345 Walnut St	2021-11-20

member_id	member_name	member_address	reg_date
C101	Alice Johnson	007 bond st	2021-05-15
C102	Bob Smith	456 Elm St	2021-06-20
C103	Carol Davis	789 Oak St	2021-07-10
C104	Dave Wilson	567 Pine St	2021-08-05
C105	Eve Brown	890 Maple St	2021-09-25
C106	Frank Thomas	234 Cedar St	2021-10-15

Task 3: Delete a Record from the Issued Status Table

Objective: Delete the record with issued_id = 'IS110' from the issued_status table.

Before Query [Result] →

issued_id	issued_member_id	issued_book_name	issued_date	issued_book_isbn	issued_emp_id
IS106	C106	Animal Farm	2024-03-10	978-0-330-25864-8	E104
IS107	C107	One Hundred Years of Solitude	2024-03-11	978-0-14-118776-1	E104
IS108	C108	The Great Gatsby	2024-03-12	978-0-525-47535-5	E104
IS109	C109	Jane Eyre	2024-03-13	978-0-141-44171-6	E105
IS110	C110	The Alchemist	2024-03-14	978-0-307-37840-1	E105
IS111	C109	Harry Potter and the Sorcerers Stone	2024-03-15	978-0-679-76489-8	E105
IS112	C109	A Game of Thrones	2024-03-16	978-0-09-957807-9	E106
IS113	C109	A Peoples History of the United States	2024-03-17	978-0-393-05081-8	E106
IS114	C109	The Guns of August	2024-03-18	978-0-19-280551-1	E106
IS115	C109	The Histories	2024-03-19	978-0-14-044930-3	E107

issued_id	issued_member_id	issued_book_name	issued_date	issued_book_isbn	issued_emp_id
IS106	C106	Animal Farm	2024-03-10	978-0-330-25864-8	E104
IS107	C107	One Hundred Years of Solitude	2024-03-11	978-0-14-118776-1	E104
IS108	C108	The Great Gatsby	2024-03-12	978-0-525-47535-5	E104
IS109	C109	Jane Eyre	2024-03-13	978-0-141-44171-6	E105
IS111	C109	Harry Potter and the Sorcerers Stone	2024-03-15	978-0-679-76489-8	E105
IS112	C109	A Game of Thrones	2024-03-16	978-0-09-957807-9	E106
IS113	C109	A Peoples History of the United States	2024-03-17	978-0-393-05081-8	E106
IS114	C109	The Guns of August	2024-03-18	978-0-19-280551-1	E106

← After Query__ [Result table]

Task 4: Retrieve All Books Issued by a Specific Employee

Objective: Select all books issued by the employee with emp_id = 'E101'.

emp_id	emp_name	issued_book_name
E101	John Doe	Moby Dick
E101	John Doe	To Kill a Mockingbird

Task 5: List Members Who Have Issued More Than One Book

Objective: Use GROUP BY to find members who have issued more than one book.

issued_member_id	member_name	book_count
C109	Ivy Martinez	7
C107	Grace Taylor	6
C105	Eve Brown	5
C110	Jack Wilson	5
C106	Frank Thomas	4
C102	Bob Smith	2
C108	Henry Anderson	2

Task 6: Create Summary Tables:

Used CTA'S to generate new tables based on query results- "each book and total book issued cunt"

issued_book_name	book_issue_count
Animal Farm	2
The Great Gatsby	2
Harry Potter and the Sorcerers Stone	2
One Hundred Years of Solitude	1
Jane Eyre	1
A Game of Thrones	1
A Peoples History of the United States	1
The Guns of August	1
The Histories	1
Guns, Germs, and Steel: The Fates of Human Societies	1
1984	1
Pride and Prejudice	1
Brave New World	1
The Road	1
The Shining	1
Fahrenheit 451	1

issued_book_name	book_issue_count
Dune	1
Where the Wild Things Are	1
The Kite Runner	1
Charlotte's Web	1
Beloved	1
A Tale of Two Cities	1
The Stand	1
Moby Dick	1
To Kill a Mockingbird	1
The Hobbit	1
Angels & Demons	1
The Diary of a Young Girl	1
Sapiens: A Brief History of Humankind	1
1491: New Revelations of the Americas Before Colum...	1
The Catcher in the Rye	1

[B]. Data Analysis & Findings:

Task 7: Retrieve All Books in a Specific Category:

category	book_title	author
History	The Histories	Herodotus
History	The Guns of August	Barbara W. Tuchman
History	Sapiens: A Brief History of Humankind	Yuval Noah Harari
History	The Diary of a Young Girl	Anne Frank
History	A Peoples History of the United States	Howard Zinn
History	Guns, Germs, and Steel: The Fates of Hu...	Jared Diamond
History	1491: New Revelations of the Americas B...	Charles C. Mann
Fiction	The Kite Runner	Khaled Hosseini
Fiction	The Alchemist	Paulo Coelho
Fiction	Beloved	Toni Morrison

Task 8: Find Total Rental Income by Category:

category	Rental_Income
Children	7.50
Classic	59.00
Dystopian	25.50
Fantasy	28.50
Fiction	12.00
History	49.50
Horror	13.00
Literary Fiction	6.50
Mystery	7.50
Science Fiction	8.50

Task 9: List Members Who Registered in the Last 2 years:

member_id	member_name	member_address	reg_date
C119	John	143 Main St	2024-05-01
C118	Sam	133 Pine St	2024-06-01
NULL	NULL	NULL	NULL

Task 10: List Employees with Their Branch Manager's Name and their branch details:

emp_id	emp_name	manager_id	manager_name	branch_id	branch_address	contact_no
E101	John Doe	E109	Daniel Anderson	B001	123 Main St	+919099988676
E102	Jane Smith	E109	Daniel Anderson	B002	456 Elm St	+919099988677
E103	Mike Johnson	E109	Daniel Anderson	B001	123 Main St	+919099988676
E104	Emily Davis	E109	Daniel Anderson	B001	123 Main St	+919099988676
E105	Sarah Brown	E109	Daniel Anderson	B001	123 Main St	+919099988676
E106	Michelle Ramirez	E109	Daniel Anderson	B001	123 Main St	+919099988676
E107	Michael Thompson	E110	Laura Martinez	B005	890 Maple St	+919099988680
E108	Jessica Taylor	E110	Laura Martinez	B004	567 Pine St	+919099988679
E109	Daniel Anderson	E109	Daniel Anderson	B003	789 Oak St	+919099988678
E110	Laura Martinez	E110	Laura Martinez	B005	890 Maple St	+919099988680
E111	Christopher Lee	E110	Laura Martinez	B005	890 Maple St	+919099988680

Task 11: Create a Table of Books with Rental Price Above a Certain Threshold:

i.e. [avg. of total Rental Price]

category	Rental_Income
History	44.00
Classic	29.50
Fantasy	28.50
Dystopian	20.00
Science Fiction	8.50
Mystery	7.50
Horror	7.00
Literary Fiction	6.50
Fiction	6.50

Results:

- In this Project, I get hand on experience about different SQL commands, which help to manage and manipulate data as per task requirement. This will help me to maintain the Library Management System updated.
- Also, here I get a hand on experience to transform data into meaningful & actionable insights to help team to improve performance of library management and services, deliver to customers.