Group 9 Project Proposal

Library Management System

Group Members:

Anish Shrestha, Brennan Tilyou, Ruxin Wang, Joshua Guillot, Kaden Wyble

Motivation

Libraries dole out a critical role in the Education industry. It is considered as the brain of any education institute, be it small or large schools, colleges, or universities. With the development of digital content, it becomes more important to manage the catalog of educational information with a scalable and reliable Library Management System that will support the general requirement of the library. Easy to use campus library management systems are now available for stress-free management of campus libraries of any size. A trustworthy web-based library management system provides a complete resolution for students, librarians, and faculty members. In this project, we want to design and create a Library Management System that can help librarians maintain the database of new books and the books that users borrow along with their due dates. This system completely automates all your library's activities.

Project Name: Library Management System
DB: MySQL
DB Name: db_library

Database Design

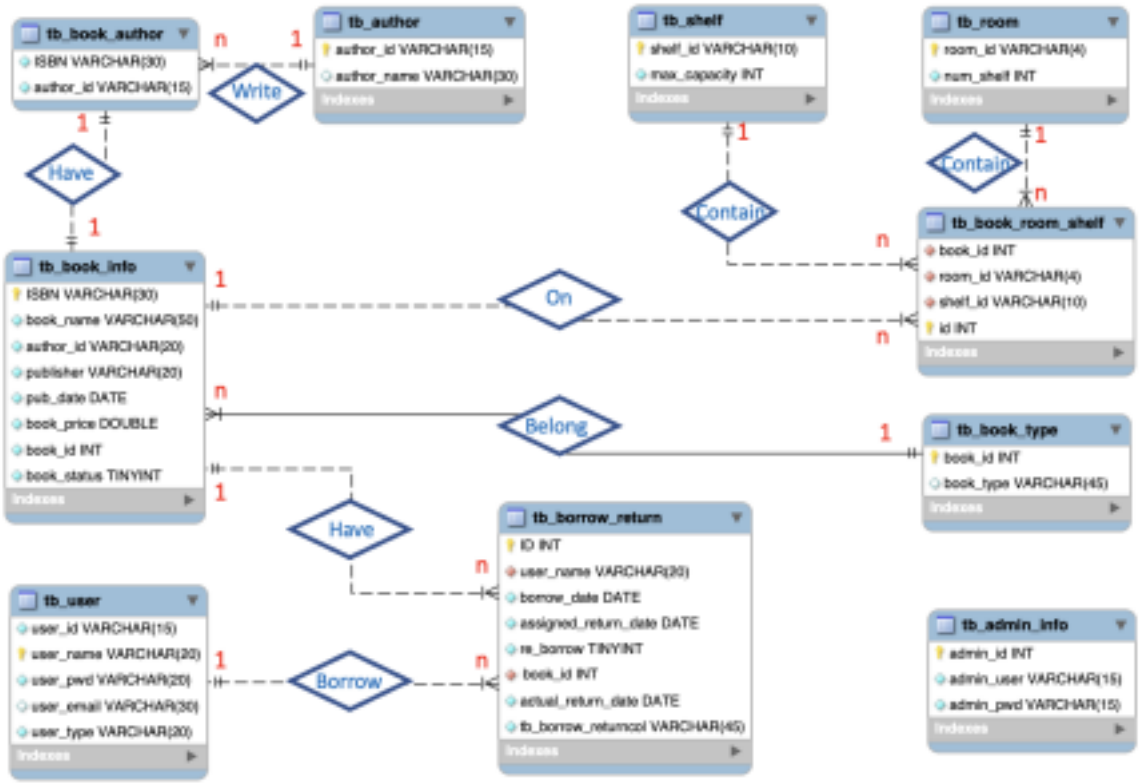
Tables:

1). tb_book_info
2). tb_user
3). tb_admin_info
4). tb_borrow_return
5). tb_book_type
6). tb_author
7). tb_book_author
8). tb_room
9). tb_shelf
10). tb_book_room_shelf

Logical Schema (Primary key **Bold**; Foreign key underlined):

tb_book_info (**ISBN**, book_name, author_id, publisher, pub_date, book_price, book_id, book_status)
tb_user (user_id, **user_name**, user_pwd, user_email, user_type)
tb_admin_info (**admin_id**, admin_user, admin_pwd)
tb_borrow_return (ISBN, user_name, borrow_date, return_date, re_borrow_date)
tb_book_type (**book_id,** book_type)
tb_author (**author_id**, author_name)
tb_book_author (ISBN, author_id)
tb_shelf(**shelf_id**, max_capacity)
tb_room(**room_id**, num_shelf)
tb_book_room_shelf (book_id, room_id, shelf_id)

E-R Diagram



Functional Dependencies:

1. **ISBN** → book_name, author_id, publisher, pub_date, book_price, book_id, book_status, return_date, re_borrow_date, author_name
2. **user_name** → user_id, user_pwd, user_email, user_type

3. **book_id** → book_type, room_id, shelf_id, max_capacity, num_shelf
4. **admin_id** → admin_user, admin_pwd

Database Implementation

Data Definition Statements (Create Database)

```
DROP DATABASE IF EXISTS db_ library;
CREATE DATABASE db_library;
USE db_library;
```

Data Definition Statements (Create Tables)

```
CREATE TABLE IF NOT EXISTS `CSC4402 Group9`.`tb_book_info` (
  `ISBN` VARCHAR(30) NOT NULL,
  `book_name` VARCHAR(50) NOT NULL,
  `author_id` VARCHAR(20) NOT NULL,
  `publisher` VARCHAR(20) NOT NULL,
  `pub_date` DATE NOT NULL,
  `book_price` DOUBLE NOT NULL,
  `book_id` INT NOT NULL,
  `book_status` TINYINT NOT NULL,
  PRIMARY KEY (`ISBN`)
)
```

```
CREATE TABLE IF NOT EXISTS `CSC4402 Group9`.`tb_user` (
  `user_id` VARCHAR(15) NOT NULL,
  `user_name` VARCHAR(20) NOT NULL,
  `user_pwd` VARCHAR(20) NOT NULL,
  `user_email` VARCHAR(30) NOT NULL,
  `user_type` VARCHAR(20) NOT NULL, (1=admin, 2=user)
  PRIMARY KEY (`user_name`),
  UNIQUE INDEX `user_name_UNIQUE` (`user_name` ASC) VISIBLE
)
```

```
CREATE TABLE IF NOT EXISTS `CSC4402 Group9`.`tb_admin_info` (
  `admin_id` INT NOT NULL,
  `admin_user` VARCHAR(15) NOT NULL,
  `admin_pwd` VARCHAR(15) NOT NULL,
```

```
  PRIMARY KEY (`admin_id`)
)


CREATE TABLE IF NOT EXISTS `CSC4402 Group9`.`tb_borrow_return` (
  `ISBN` VARCHAR(30) NOT NULL,
  `user_name` VARCHAR(20) NOT NULL,
  `borrow_date` DATE NOT NULL,
  `return_date` DATE NOT NULL,
  `re_borrow` TINYINT NOT NULL, COMMENT 're_borrow wont stay as a boolean
',
  INDEX `ISBN_idx` (`ISBN` ASC) VISIBLE,
  INDEX `user_name_idx` (`user_name` ASC) VISIBLE
)


CREATE TABLE IF NOT EXISTS `CSC4402 Group9`.`tb_book_type` (
  `book_id` VARCHAR(20) NOT NULL,
  `book_type` VARCHAR(45) NOT NULL,
  PRIMARY KEY (`book_id`)
)


CREATE TABLE IF NOT EXISTS `CSC4402 Group9`.`tb_author` (
  `author_id` VARCHAR(15) NOT NULL,
  `author_name` VARCHAR(30) NOT NULL,
  PRIMARY KEY (`author_id`)
)


CREATE TABLE IF NOT EXISTS `CSC4402 Group9`.`tb_book_author` (
  `ISBN` VARCHAR(30) NOT NULL,
  `author_id` VARCHAR(15) NOT NULL
)


CREATE TABLE IF NOT EXISTS `CSC4402 Group9`.`tb_shelf` (
  `shelf_id` VARCHAR(10) NOT NULL,
  `max_capacity` INT NOT NULL,
  PRIMARY KEY (`shelf_id`)
)
```

```
CREATE TABLE IF NOT EXISTS `CSC4402 Group9`.`tb_room` (
  `room_id` VARCHAR(4) NOT NULL,
  `num_shelf` INT NOT NULL,
  PRIMARY KEY (`room_id`)
)
```

```
CREATE TABLE IF NOT EXISTS `CSC4402 Group9`.`tb_book_room_shelf` (
  `book_id` INT NOT NULL,
  `room_id` VARCHAR(4) NOT NULL,
  `shelf_id` VARCHAR(10) NOT NULL
)
```

Insert Data Statements

Insert Into Table (tb_author)

INSERT INTO `CSC4402 Group9`.`tb_author` (`author_id`, `author_name`) VALUES ('1', 'John Green');
INSERT INTO `CSC4402 Group9`.`tb_author` (`author_id`, `author_name`) VALUES ('2', 'Geroge R.R. Martin');

Insert Into Table (tb_book_type)

INSERT INTO `CSC4402 Group9`.`tb_book_type` (`book_id`, `book_type`) VALUES ('1', 'Fantasy');
INSERT INTO `CSC4402 Group9`.`tb_book_type` (`book_id`, `book_type`) VALUES ('2', 'Sci-Fi');

Insert Into Table (tb_shelf)

INSERT INTO `CSC4402 Group9`.`tb_shelf` (`shelf_id`, `max_capacity`) VALUES ('1', '50');
INSERT INTO `CSC4402 Group9`.`tb_shelf` (`shelf_id`, `max_capacity`) VALUES ('2', '50');

Insert Into Table (tb_room)

INSERT INTO `CSC4402 Group9`.`tb_room` (`room_id`, `num_shelf`) VALUES ('100', '12');
INSERT INTO `CSC4402 Group9`.`tb_room` (`room_id`, `num_shelf`) VALUES ('101', '10');

Insert Into Table (tb_book_author)

INSERT INTO `CSC4402 Group9`.`tb_book_author` (`ISBN`, `author_id`) VALUES('000-ISBN-017',7);
INSERT INTO `CSC4402 Group9`.`tb_book_author` (`ISBN`, `author_id`)
VALUES (' 000-ISBN-001', '3 ');

Insert Into Table (tb_user)

INSERT INTO `CSC4402 Group9`.`tb_user` (`user_id`, `user_name`, `user_pwd`, `user_email`, `user_type`) VALUES ('128364', 'anish', 'HereIAm!', 'ashre18@lsu.edu', '2');
INSERT INTO `CSC4402 Group9`.`tb_user` (`user_id`, `user_name`, `user_pwd`, `user_email`, `user_type`) VALUES ('164897', 'Donkey Kong', 'Bananas', 'DK@gmail.com', '1');

Insert Into Table (tb_book_info)

INSERT INTO `CSC4402 Group9`.`tb_book_info` (`ISBN`, `book_name`, `author_id`, `publisher`, `pub_date`, `book_price`, `book_id`, `book_status`) VALUES ("000-ISBN-001','Harry Potter and the Goblet of Fire','3','Bloomsbury Publishin','1998-07-02',21,1,0)
INSERT INTO `CSC4402 Group9`.`tb_book_info` (`ISBN`, `book_name`, `author_id`, `publisher`, `pub_date`, `book_price`, `book_id`, `book_status`) VALUES ("000-ISBN-002','Astrophyics for People in a Hurry','4','W. W. Norton & Compa','2017-05-02',10,2,0);

Insert Into Table (tb_book_room_shelf)

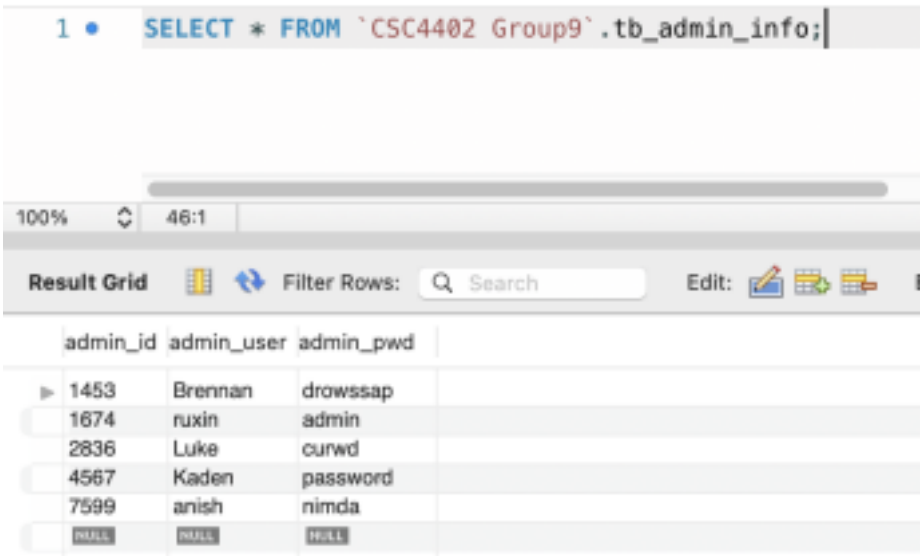
INSERT INTO `CSC4402 Group9`.`tb_book_room_shelf` (`book_id`, `room_id`, `shelf_id`, `id`) VALUES (1, '101', '5', 1);
INSERT INTO `CSC4402 Group9`.`tb_book_room_shelf` (`book_id`, `room_id`, `shelf_id`, `id`) VALUES (2, '100', '1', 2);

Insert Into Table (tb_borrow_return)

INSERT INTO `CSC4402 Group9`.`tb_borrow_return` (`id`, `user_name`, `borrow_date`, `assigned_return_date`, `re_borrow`, `book_id`, `actual_return_date`, `tb_borrow_returncol`) VALUES (1, 'anish', '2022-03-01', '2022-05-01', 0, 1, NULL, ' ');
INSERT INTO `CSC4402 Group9`.`tb_borrow_return` (`id`, `user_name`, `borrow_date`, `assigned_return_date`, `re_borrow`, `book_id`, `actual_return_date`, `tb_borrow_returncol`) VALUES (2, 'anish', '2022-03-02', '2022-05-05', 0, 3, '2022-05-01', ' ');

Insert Into Table (tb_admin_info)

INSERT INTO `CSC4402 Group9`.`tb_admin_info` (`admin_id`, `admin_user`, `admin_pwd`) VALUES (1453, 'Brennan', 'drowssap');
INSERT INTO `CSC4402 Group9`.`tb_admin_info` (`admin_id`, `admin_user`, `admin_pwd`) VALUES (1674, ruxin, 'admin');



Show Table Information (Select statements)

1. `SELECT * FROM `CSC4402 Group9`.tb_admin_info;`

2. `SELECT * FROM `CSC4402 Group9`.tb_author;`

```
1 •  SELECT * FROM `CSC4402 Group9`.tb_book_info;
```

00% 1:1

Result Grid Filter Rows: Search Edit: Export/Import:

ISBN	book_name	author_id	publisher	pub_date	book_price	book_id	book_status
000-ISBN-001	Harry Potter and the Goblet of Fire	3	Bloomsbury Publishin	1998-07-02	21	1	1
000-ISBN-002	Astrophyics for People in a Hurry	4	W. W. Norton & Compa	2017-05-02	10	2	0
000-ISBN-003	Harry Potter and the Deathly Hallows	3	Bloomsbury Publishin	2007-07-14	18	3	0
000-ISBN-004	Harry Potter and the Order of Phoenix	3	Bloomsbury Publishin	2003-06-21	18	4	0
000-ISBN-005	Harry Potter and the Half-blood Prince	3	Bloomsbury Publishin	2005-07-16	20	5	1
000-ISBN-006	Harry Potter and the Sorcerer's Stone	3	Bloomsbury Publishin	1997-06-26	18	6	0
000-ISBN-007	Harry Potter and the Chamber of Secrets	3	Bloomsbury Publishin	1998-07-02	20	7	0
000-ISBN-008	Harry Potter and the Prisoner of Azkaban	3	Bloomsbury Publishin	2000-07-08	20	8	0
000-ISBN-009	No Longer Human	5	Chikuma Shobo	1948-00-00	14	9	0
000-ISBN-010	The Hunger Games	8	Scholastic	2008-09-00	10	10	0
000-ISBN-011	Catching Fire	8	Scholastic	2009-09-01	14	11	0
000-ISBN-012	Mockingjay	8	Scholastic	2010-08-24	14	12	0
000-ISBN-013	The Ballad of Songbirds and Snakes	8	Scholastic	2020-05-19	20	13	0
000-ISBN-014	Diary of a Wimpy Kid	9	ABRAMS	2007-04-01	10	14	0
000-ISBN-015	Diary of a Wimpy Kid: Rodrick Rules	9	ABRAMS	2008-02-01	10	15	0
000-ISBN-016	The Return	6	Grand Central Publis	2020-09-29	14	16	0
000-ISBN-017	Untamed	7	The Dial Press	2020-03-10	15	17	0
000-ISBN-018	A Random Book	7	A Random Publication	2020-01-01	15	18	0

```
1 •  SELECT * FROM `CSC4402 Group9`.tb_book_author;
```

100% 1:1

Result Grid Filter Rows: Search Edit:

ISBN	author_id	id
000-ISBN-001	3	1
000-ISBN-002	4	2
000-ISBN-003	3	3
000-ISBN-004	3	4
000-ISBN-005	3	5
000-ISBN-006	3	6
000-ISBN-007	3	7
000-ISBN-008	3	8
000-ISBN-009	5	9
000-ISBN-010	8	10
000-ISBN-011	8	11

```
SELECT * FROM `CSC4402 Group9`.tb_book_author;
```

4.

```
SELECT * FROM `CSC4402 Group9`.tb_book_info;
```

5.

```
SELECT * FROM `CSC4402 Group9`.tb_room_shelf;
```

```
1 •  SELECT * FROM `CSC4402 Group9`.tb_borrow_return;
```

Result Grid | Filter Rows: Search | Edit: | Export/Import:

id	user_name	borrow_date	assigned_return_da...	re_borrow	book_id	actual_return_da...	tb_borrow_return
1	anish	2022-03-01	2022-05-01	0	1	NULL	
2	anish	2022-03-02	2022-05-05	0	3	2022-05-01	
3	ruxin	2021-11-19	2022-04-23	0	17	2022-02-19	
4	Luffy	2022-10-07	2023-01-01	2	5	2022-03-10	
5	MattPat	2020-05-21	2021-07-16	1	8	2021-07-20	
6	anish	2020-05-01	2021-05-01	0	18	NULL	

```
1 •  SELECT * FROM `CSC4402 Group9`.tb_book_type;
```

Result Grid | Filter Rows: Search | Edit:

book_id	book_type
1	Fantasy
10	Short Stories
11	Suspense and Thrillers
12	Women's Fiction
2	Action
3	Action and Adventure
4	Comic Book or Graphic Novel
5	Detective and Mystery
6	Historical Fiction
7	Horror
8	Romance
9	Science Fiction

6.

```
SELECT * FROM `CSC4402 Group9`.tb_book_type;
```
7.
```
SELECT * FROM `CSC4402 Group9`.tb_borrow_return;
```

8. `SELECT * FROM `CSC4402 Group9`.tb_room;`

Data Manipulation Statements (Select statements)

Me as a user, the use cases in a Library System Management I will likely want that can be solved by Select statements

1. Select isbn, name and author of all book ordered by book name alphabetically:

```
SELECT a.ISBN, a.book_name, b.author_name FROM `CSC4402 Group9`.tb_book_info as a inner join `CSC4402 Group9`.tb_author as b on a.author_id = b.author_id order by book_name;
```

ISBN	book_name	author_name
2	Astrophyics for People in a Hurry	Neil de Grasse Tyson
7	Harry Potter and the Chamber of Secrets	JK Rowling
3	Harry Potter and the Deathly Hallows	JK Rowling
1	Harry Potter and the Goblet of Fire	JK Rowling
5	Harry Potter and the Half-blood Prince	JK Rowling
4	Harry Potter and the Order of Phoenix	JK Rowling
8	Harry Potter and the Prisoner of Azkaban	JK Rowling
6	Harry Potter and the Sorcerer's Stone	JK Rowling

2. Select ISBN, name of the book, author and count in the library ordered alphabetically

```
SELECT a.ISBN, a.book_name, b.author_name, COUNT(a.ISBN) as count FROM `CSC4402 Group9`.tb_book_info as a inner join `CSC4402 Group9`.tb_author as b on a.author_id = b.author_id GROUP BY a.book_name order by book_name;
```

ISBN	book_name	author_name	count
2	Astrophyics for People in a Hurry	Neil de Grasse Tyson	1
7	Harry Potter and the Chamber of Secrets	JK Rowling	1
3	Harry Potter and the Deathly Hallows	JK Rowling	1
1	Harry Potter and the Goblet of Fire	JK Rowling	1
5	Harry Potter and the Half-blood Prince	JK Rowling	1
4	Harry Potter and the Order of Phoenix	JK Rowling	1
8	Harry Potter and the Prisoner of Azkaban	JK Rowling	1
6	Harry Potter and the Sorcerer's Stone	JK Rowling	1

3. Select all the books of a title available for borrow:

```
SELECT ISBN, BO.book_name, AU.author_name, room_id, shelf_id FROM (
```

```
            select * from `CSC4402 Group9`.tb_book_info
            where ISBN not in (
                  Select ISBN from `CSC4402 Group9`.tb_borrow_return
                  where `CSC4402
Group9`.tb_borrow_return.assigned_return_Date IS NOT NULL AND `CSC4402
Group9`.tb_borrow_return.actual_return_date IS NULL)
            And book_name LIKE "Harry potter%"
      ) as BO
inner join `CSC4402 Group9`.tb_author as AU
ON BO.author_id = AU.author_id
left outer join `CSC4402 Group9`.tb_book_room_shelf as C
ON C.book_id = BO.book_id;
```

4. Filter by certain genre:

```
SELECT DISTINCT ISBN, book_name, C.author_name, publisher, pub_date FROM
`CSC4402 Group9`.tb_book_info A
INNER JOIN (
      SELECT book_id from `CSC4402 Group9`.tb_book_type AS A
    WHERE book_type = "Action"
    ) B
ON B.book_id = A.book_id
INNER JOIN `CSC4402 Group9`.tb_author C
ON A.author_id = C.author_id;
```

	ISBN	book_name	author_name	publisher	pub_date
▶	2	Astrophyics for People in a Hurry	Neil de Grasse Tyson	W. W. Norton & Compa	2017-05-02

5. Select all the books in a shelf:

```
SELECT A.book_id, A.ISBN, A.book_name, B.author_name, A.publisher,
A.pub_date FROM `CSC4402 Group9`.tb_book_info AS A
inner join `CSC4402 Group9`.tb_author as B
ON A.author_id = B.author_id
left outer join `CSC4402 Group9`.tb_book_room_shelf as C
ON C.book_id = A.book_id;
```

book_id	ISBN	book_name	author_name	publisher	pub_date
1	1	Harry Potter and the Goblet of Fire	JK Rowling	Bloomsbury Publishin	1998-07-02
2	2	Astrophyics for People in a Hurry	Neil de Grasse Tyson	W. W. Norton & Compa	2017-05-02
3	3	Harry Potter and the Deathly Hallows	JK Rowling	Bloomsbury Publishin	2007-07-14
4	4	Harry Potter and the Order of Phoenix	JK Rowling	Bloomsbury Publishin	2003-06-21
5	5	Harry Potter and the Half-blood Prince	JK Rowling	Bloomsbury Publishin	2005-07-16
6	6	Harry Potter and the Sorcerer's Stone	JK Rowling	Bloomsbury Publishin	1997-06-26
7	7	Harry Potter and the Chamber of Secrets	JK Rowling	Bloomsbury Publishin	1998-07-02
8	8	Harry Potter and the Prisoner of Azkaban	JK Rowling	Bloomsbury Publishin	2000-07-08

6. Select all the books of one author:

```
SELECT A.book_id, A.ISBN, A.book_name, B.author_name, A.publisher,
A.pub_date, C.room_id, C.shelf_id, CASE WHEN (actual_return_date IS NULL
AND borrow_date IS NOT NULL) THEN "Not Available" ELSE "Available" END as
available FROM `CSC4402 Group9`.tb_book_info A
inner join (SELECT * FROM `CSC4402 Group9`.tb_author WHERE author_name LIKE
"JK_ROWLING") B
ON A.author_id = B.author_id
left outer join `CSC4402 Group9`.tb_book_room_shelf as C
ON C.book_id = A.book_id
left outer join `CSC4402 Group9`.tb_borrow_return D
ON A.book_id = D.book_id;
```

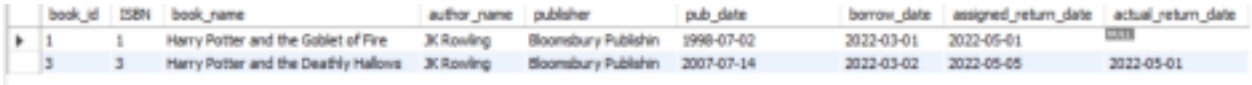
book_id	ISBN	book_name	author_name	publisher	pub_date	room_id	shelf_id	available
1	1	Harry Potter and the Goblet of Fire	JK Rowling	Bloomsbury Publishin	1998-07-02	101	6	Not Available
3	3	Harry Potter and the Deathly Hallows	JK Rowling	Bloomsbury Publishin	2007-07-14	100	2	Available
4	4	Harry Potter and the Order of Phoenix	JK Rowling	Bloomsbury Publishin	2003-06-21	101	2	Available
5	5	Harry Potter and the Half-blood Prince	JK Rowling	Bloomsbury Publishin	2005-07-16	NULL	NULL	Available
6	6	Harry Potter and the Sorcerer's Stone	JK Rowling	Bloomsbury Publishin	1997-06-26	NULL	NULL	Available
7	7	Harry Potter and the Chamber of Secrets	JK Rowling	Bloomsbury Publishin	1998-07-02	NULL	NULL	Available
8	8	Harry Potter and the Prisoner of Azkaban	JK Rowling	Bloomsbury Publishin	2000-07-08	NULL	NULL	Available

7. Select all the books overdue by me (given a user_name) ordered by issued date

// TODO - Do not think this use case needs to be a SQL statement

8. Select all the books that I borrowed ordered by issued date

```
SELECT A.book_id, A.ISBN, A.book_name, B.author_name, A.publisher,
A.pub_date, C.borrow_date, C.assigned_return_date, C.actual_return_date
FROM `CSC4402 Group9`.tb_book_info A
inner join `CSC4402 Group9`.tb_author as B
ON A.author_id = B.author_id
inner join (SELECT * FROM `CSC4402 Group9`.tb_borrow_return as C
      WHERE C.user_name LIKE "anish"
    ) C
ON A.book_id = C.book_id;
```



book_id	ISBN	book_name	author_name	publisher	pub_date	borrow_date	assigned_return_date	actual_return_date
1	1	Harry Potter and the Goblet of Fire	JK Rowling	Bloomsbury Publishin	1998-07-02	2022-03-01	2022-05-01	NULL
3	3	Harry Potter and the Deathly Hallows	JK Rowling	Bloomsbury Publishin	2007-07-14	2022-03-02	2022-05-05	2022-05-01

9. Select all the books by a publisher

```
SELECT A.ISBN, A.book_name, B.author_name, Count(A.book_id) as count FROM
`CSC4402 Group9`.tb_book_info A
inner join `CSC4402 Group9`.tb_author as B
ON A.author_id = B.author_id
GROUP BY ISBN;
```

Me as a librarian, the use cases in a Library System Management I will likely want that can be solved by Select statements:

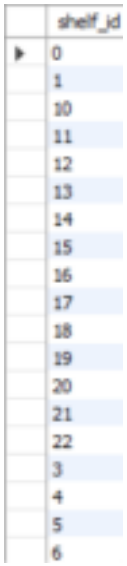
10. Select all the books currently issued

```
SELECT A.book_id, A.ISBN, A.book_name, B.author_name, A.publisher,
A.pub_date, C.borrow_date, C.user_name, C.assigned_return_date,
C.actual_return_date FROM `CSC4402 Group9`.tb_book_info A inner
join `CSC4402 Group9`.tb_author as B
ON A.author_id = B.author_id
inner join `CSC4402 Group9`.tb_borrow_return as C
ON A.book_id = C.book_id;
```

book_id	ISBN	book_name	author_name	publisher	pub_date	borrow_date	user_name	assigned_return_date	actual_return_date
1	1	Harry Potter and the Goblet of Fire	JK Rowling	Bloomsbury Publishin	1998-07-02	2022-03-01	anish	2022-05-01	NULL
3	3	Harry Potter and the Deathly Hallows	JK Rowling	Bloomsbury Publishin	2007-07-14	2022-03-02	anish	2022-05-05	2022-05-01

11. Select all the shelves that can hold X number of books

```
SELECT A.shelf_id FROM `CSC4402 Group9`.tb_shelf as A
LEFT OUTER JOIN (SELECT shelf_id, count(shelf_id) as occupied_capacity FROM
`CSC4402 Group9`.tb_book_room_shelf GROUP BY shelf_id) as C
ON A.shelf_id = C.shelf_id
WHERE A.shelf_id NOT IN (SELECT DISTINCT shelf_id from `CSC4402
Group9`.tb_book_room_shelf) OR A.max_capacity - C.occupied_capacity >= 49;
```



shelf_id
0
1
10
11
12
13
14
15
16
17
18
19
20
21
22
3
4
5
6

12. Select all the overdue books.

```
DELIMITER $$
DROP PROCEDURE overdue$$
CREATE PROCEDURE overdue()
BEGIN
      DECLARE today_date DATE;
      SET today_date='2023-02-05';
      SELECT A.book_id, A.ISBN, A.book_name, B.borrow_date,
   B.assigned_return_date FROM `CSC4402 Group9`.tb_book_info A inner
         join (SELECT * FROM `CSC4402 Group9`.tb_borrow_return WHERE
                  borrow_date IS NOT NULL AND assigned_return_date <
today_date
            ) B
      ON A.book_id = B.book_id;
END$$
DELIMITER ;

CALL overdue ();
```

book_id	ISBN	book_name	borrow_date	assigned_return_date
1	1	Harry Potter and the Goblet of Fire	2022-03-01	2022-05-01
3	3	Harry Potter and the Deathly Hallows	2022-03-02	2022-05-05

13. Select for viewing user information given a user_name

```
SELECT A.user_id, A.user_name, A.user_email, A.user_type from tb_user A
WHERE A.user_name LIKE "anish";
```

14. Select for viewing all the overdue books given a user_name

```
DELIMITER $$
DROP PROCEDURE overdue$$
CREATE PROCEDURE overdue(
      IN required_user_name VARCHAR(20)
)
BEGIN
      DECLARE today_date DATE;
      SET today_date='2023-02-05';
      SELECT A.book_id, A.ISBN, A.book_name, B.borrow_date,
   B.assigned_return_date FROM `CSC4402 Group9`.tb_book_info A inner
         join (SELECT * FROM `CSC4402 Group9`.tb_borrow_return WHERE
                  borrow_date IS NOT NULL AND assigned_return_date <
today_date
            ) B
      ON A.book_id = B.book_id
    inner join (SELECT * FROM tb_user WHERE user_name LIKE
required_user_name) C
    ON B.user_name = C.user_name;
END$$
DELIMITER ;

CALL overdue ("JoeyB");
```

Data Manipulation Statements (Update statements)

Update for tb_book_info

```
UPDATE `CSC4402 Group9`.`tb_book_info` SET `pub_date` = '1998-07-02' WHERE
(`ISBN` = '1');
```

Update for when user returns a borrowed book with certain id

```
UPDATE `CSC4402 Group9`.`tb_borrow_return` SET `actual_return_date` =
'2022-05-02' WHERE (`id` = '1');
```

Update for when a book's location is changed

```
UPDATE `CSC4402 Group9`.`tb_book_room_shelf` SET `shelf_id` = '6' WHERE
(`id` = '1');
```

Update the name of an author whose id is 1

```
UPDATE 'CSC4402 Group9' . 'tb_author' SET 'author_name' = 'Christopher
Paolini' WHERE ('author_id' = '1');
```

Update the capacity of a shelf if it is replaced with a larger one

```
UPDATE 'CSC4402 Group9' . 'tb_shelf' SET 'max_capacity' = '150' WHERE ('shelf_id' = '32')
```

Show all Tables information