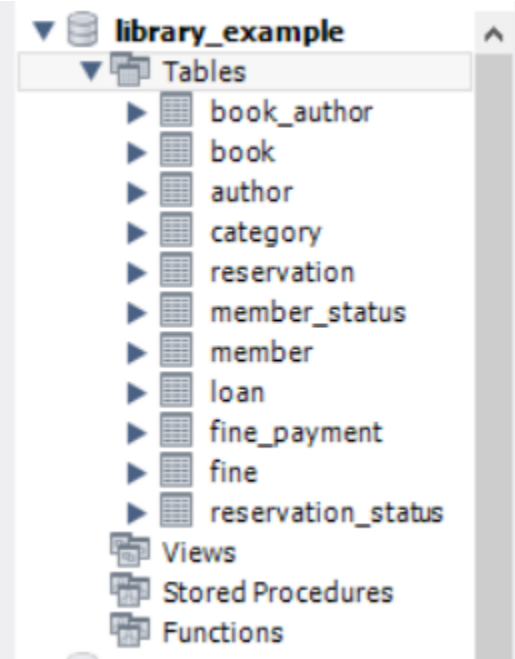
# Library Management System Example

David Asare

I created the schema called “library_example” and imported a file to create the tables.




Below is the created statements file I imported.

```sql
/*
Create tables for the library management database
This works for MySQL. To transform it to other databases such as SQL Server, Oracle, or Postgres,
you should be able to just change the data types to types recognised by the database.
*/

CREATE TABLE reservation_status (
  id INT,
  status_value VARCHAR(50),
  CONSTRAINT pk_res_status PRIMARY KEY (id)
);

CREATE TABLE category (
  id INT,
  category_name VARCHAR(100),
  CONSTRAINT pk_category PRIMARY KEY (id)
);


CREATE TABLE book (
  id INT,
  title VARCHAR(500),
  category_id INT,
  publication_date DATE,
  copies_owned INT,
  CONSTRAINT pk_book PRIMARY KEY (id),
  CONSTRAINT fk_book_category FOREIGN KEY (category_id) REFERENCES category(id)
);

CREATE TABLE author (
  id INT,
  first_name VARCHAR(300),
  last_name VARCHAR(300),
  CONSTRAINT pk_author PRIMARY KEY (id)
);
```

```sql
CREATE TABLE book_author (
  book_id INT,
  author_id INT,
  CONSTRAINT fk_bookauthor_book FOREIGN KEY (book_id) REFERENCES book(id),
  CONSTRAINT fk_bookauthor_author FOREIGN KEY (author_id) REFERENCES author(id)
);

CREATE TABLE member_status (
  id INT,
  status_value VARCHAR(50),
  CONSTRAINT pk_memberstatus PRIMARY KEY (id)
);

CREATE TABLE member (
  id INT,
  first_name VARCHAR(300),
  last_name VARCHAR(300),
  joined_date DATE,
  active_status_id INT,
  CONSTRAINT pk_member PRIMARY KEY (id),
  CONSTRAINT fk_member_status FOREIGN KEY (active_status_id) REFERENCES member_status(id)
);

CREATE TABLE reservation (
  id INT,
  book_id INT,
  member_id INT,
  reservation_date DATE,
  reservation_status_id INT,
  CONSTRAINT pk_reservation PRIMARY KEY (id),
  CONSTRAINT fk_res_book FOREIGN KEY (book_id) REFERENCES book(id),
  CONSTRAINT fk_res_member FOREIGN KEY (member_id) REFERENCES member(id)
);
```

```sql
CREATE TABLE fine_payment (
    id INT,
    member_id INT,
    payment_date DATE,
    payment_amount INT,
    CONSTRAINT pk_fine_payment PRIMARY KEY (id),
    CONSTRAINT fk_finepay_member FOREIGN KEY (member_id) REFERENCES member(id)
);
```

```sql
CREATE TABLE loan (
    id INT,
    book_id INT,
    member_id INT,
    loan_date DATE,
    returned_date DATE,
    CONSTRAINT pk_loan PRIMARY KEY (id),
    CONSTRAINT fk_loan_book FOREIGN KEY (book_id) REFERENCES book(id),
    CONSTRAINT fk_loan_member FOREIGN KEY (member_id) REFERENCES member(id)
);
```

```sql
CREATE TABLE fine (
    id INT,
    book_id INT,
    loan_id INT,
    fine_date DATE,
    fine_amount INT,
    CONSTRAINT pk_fine PRIMARY KEY (id),
    CONSTRAINT fk_fine_book FOREIGN KEY (book_id) REFERENCES book(id),
    CONSTRAINT fk_fine_loan FOREIGN KEY (loan_id) REFERENCES loan(id)
);
```

Below is the mysql file I made called “library_example_script” where I made up examples for the queries.

```
-- Query examples I made based on the library database

-- Inserting names of authours into the author table.

INSERT INTO `library_example`.`author` (`id`, `first_name`, `last_name`) VALUES ('1', 'Tony', 'Stark');
INSERT INTO `library_example`.`author` (`id`, `first_name`, `last_name`) VALUES ('2', 'Bruce', 'Wayne');
INSERT INTO `library_example`.`author` (`id`, `first_name`, `last_name`) VALUES ('3', 'Oliver', 'Queen');

select * from library_example.author;
```

Result Grid | Filter Rows:

| id | first_name | last_name |
|---|---|---|
| 1 | Tony | Stark |
| 2 | Bruce | Wayne |
| 3 | Oliver | Queen |
| NULL | NULL | NULL |

```
-- Inserting info into category table
INSERT INTO `library_example`.`category` (`id`, `category_name`) VALUES ('100', 'Cat One');
INSERT INTO `library_example`.`category` (`id`, `category_name`) VALUES ('200', 'Cat Two');
INSERT INTO `library_example`.`category` (`id`, `category_name`) VALUES ('300', 'Cat three');

select * from library_example.category;
```

Result Grid | Filter Rows:

| id | category_name |
|---|---|
| 100 | Cat One |
| 200 | Cat Two |
| 300 | Cat three |
| NULL | NULL |

```
-- Inserting info into book table.
INSERT INTO `library_example`.`book` (`id`, `title`, `category_id`, `publication_date`, `copies_owned`) VALUES ('10', 'My Story One', '100', '2021-03-19', '38');
INSERT INTO `library_example`.`book` (`id`, `title`, `category_id`, `publication_date`, `copies_owned`) VALUES ('20', 'My Story Two', '200', '2020-01-2', '45');
INSERT INTO `library_example`.`book` (`id`, `title`, `category_id`, `publication_date`, `copies_owned`) VALUES ('30', 'My  Story Three', '300', '2019-02-12', '34');

select * from library_example.book;
```

Result Grid | Filter Rows: | Edit: | Export/I

| | id | title | category_id | publication_date | copies_owned |
|---|---|---|---|---|---|
| ▶ | 10 | My Story One | 100 | 2021-03-19 | 38 |
| | 20 | My Story Two | 200 | 2020-01-02 | 45 |
| | 30 | My  Story Three | 300 | 2019-02-12 | 34 |
| * | NULL | NULL | NULL | NULL | NULL |

```
-- Inserting info into member_status tabe.
INSERT INTO `library_example`.`member_status` (`id`, `status_value`) VALUES ('451', 'value One');
INSERT INTO `library_example`.`member_status` (`id`, `status_value`) VALUES ('452', 'Value Two');
INSERT INTO `library_example`.`member_status` (`id`, `status_value`) VALUES ('453', 'Value Three');

select * from library_example.member_status;
```

Result Grid | Filter Rows:

| | id | status_value |
|---|---|---|
| ▶ | 451 | value One |
| | 452 | Value Two |
| | 453 | Value Three |
| * | NULL | NULL |

```
-- Inserting info into the member table
INSERT INTO `library_example`.`member` (`id`, `first_name`, `last_name`, `joined_date`, `active_status_id`) VALUES ('351', 'Jason', 'Todd', '2018-02-28', '451');
INSERT INTO `library_example`.`member` (`id`, `first_name`, `last_name`, `joined_date`, `active_status_id`) VALUES ('352', 'Barbra ', 'Gordon', '2019-01-27', '452');
INSERT INTO `library_example`.`member` (`id`, `first_name`, `last_name`, `joined_date`, `active_status_id`) VALUES ('353', 'Felicity', 'Smoke', '2020-03-26', '453');

select * from library_example.member;
```

Result Grid | Filter Rows: | Edit:

| id | first_name | last_name | joined_date | active_status_id |
|---|---|---|---|---|
| 351 | Jason | Todd | 2018-02-28 | 451 |
| 352 | Barbra | Gordon | 2019-01-27 | 452 |
| 353 | Felicity | Smoke | 2020-03-26 | 453 |
| NULL | NULL | NULL | NULL | NULL |

```
-- Inserting info into loan table.

INSERT INTO `library_example`.`loan` (`id`, `book_id`, `member_id`, `loan_date`, `returned_date`) VALUES ('251', '10', '351', '2020-02-17', '2020-02-19');
INSERT INTO `library_example`.`loan` (`id`, `book_id`, `member_id`, `loan_date`, `returned_date`) VALUES ('252', '20', '352', '2021-03-18', '2021-03-19');
INSERT INTO `library_example`.`loan` (`id`, `book_id`, `member_id`, `loan_date`, `returned_date`) VALUES ('253', '30', '353', '2022-04-19', '2022-04-20');
```

```
select * from library_example.loan
```

Result Grid | Filter Rows: | Edit:

| id | book_id | member_id | loan_date | returned_date |
|---|---|---|---|---|
| 251 | 10 | 351 | 2020-02-17 | 2020-02-19 |
| 252 | 20 | 352 | 2021-03-18 | 2021-03-19 |
| 253 | 30 | 353 | 2022-04-19 | 2022-04-20 |
| NULL | NULL | NULL | NULL | NULL |

```
-- Inserting info the fine table
INSERT INTO `library_example`.`fine` (`id`, `book_id`, `loan_id`, `fine_date`, `fine_amount`) VALUES ('151', '10', '251', '2020-02-20', '1000');
INSERT INTO `library_example`.`fine` (`id`, `book_id`, `loan_id`, `fine_date`, `fine_amount`) VALUES ('152', '20', '252', '2019-05-12', '2000');
INSERT INTO `library_example`.`fine` (`id`, `book_id`, `loan_id`, `fine_date`, `fine_amount`) VALUES ('153', '30', '253', '2021-05-13', '3000');
```

```
select * from library_example.fine
```

Result Grid | Filter Rows: | Edit:

| id | book_id | loan_id | fine_date | fine_amount |
|---|---|---|---|---|
| 151 | 10 | 251 | 2020-02-20 | 1000 |
| 152 | 20 | 252 | 2019-05-12 | 2000 |
| 153 | 30 | 253 | 2021-05-13 | 3000 |
| NULL | NULL | NULL | NULL | NULL |

fine 8

```
-- Inserting info into fine_payment table
INSERT INTO `library_example`.`fine_payment` (`id`, `member_id`, `payment_date`, `payment_amount`) VALUES ('141', '351', '2021-08-19', '2000');
INSERT INTO `library_example`.`fine_payment` (`id`, `member_id`, `payment_date`, `payment_amount`) VALUES ('142', '352', '2021-09-12', '4000');
INSERT INTO `library_example`.`fine_payment` (`id`, `member_id`, `payment_date`, `payment_amount`) VALUES ('143', '353', '2021-10-13', '6000');

select * from library_example.fine_payment
```

Result Grid | Filter Rows: | Edit:

| id | member_id | payment_date | payment_amount |
|---|---|---|---|
| 141 | 351 | 2021-08-19 | 2000 |
| 142 | 352 | 2021-09-12 | 4000 |
| 143 | 353 | 2021-10-13 | 6000 |
| NULL | NULL | NULL | NULL |

```
-- Inserting info into reservation_status table
INSERT INTO `library_example`.`reservation_status` (`id`, `status_value`) VALUES ('700', '421');
INSERT INTO `library_example`.`reservation_status` (`id`, `status_value`) VALUES ('800', '422');
INSERT INTO `library_example`.`reservation_status` (`id`, `status_value`) VALUES ('900', '423');

select * from library_example.reservation_status;
```

Result Grid | Filter Rows:

| id | status_value |
|---|---|
| 700 | 421 |
| 800 | 422 |
| 900 | 423 |
| NULL | NULL |

```
INSERT INTO `library_example`.`reservation` (`id`, `book_id`, `member_id`, `reservation_date`, `reservation_status_id`) VALUES ('651', '10', '351', '2020-06-18', '700');
INSERT INTO `library_example`.`reservation` (`id`, `book_id`, `member_id`, `reservation_date`, `reservation_status_id`) VALUES ('652', '20', '352', '2021-03-19', '800');
INSERT INTO `library_example`.`reservation` (`id`, `book_id`, `member_id`, `reservation_date`, `reservation_status_id`) VALUES ('653', '30', '353', '2022-04-20', '900');
```

```
select * from library_example.reservation;
```

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

| id | book_id | member_id | reservation_date | reservation_status_id |
|---|---|---|---|---|
| 651 | 10 | 351 | 2020-06-18 | 700 |
| 652 | 20 | 352 | 2021-03-19 | 800 |
| 653 | 30 | 353 | 2022-04-20 | 900 |
| NULL | NULL | NULL | NULL | NULL |

```
-- Displaying records with fine amount larger than 1000
select *from library_example.fine
where fine_amount > 1000;
```

Result Grid | Filter Rows: | Edit: | Ex

| id | book_id | loan_id | fine_date | fine_amount |
|---|---|---|---|---|
| 152 | 20 | 252 | 2019-05-12 | 2000 |
| 153 | 30 | 253 | 2021-05-13 | 3000 |
| NULL | NULL | NULL | NULL | NULL |

```
-- Counting number of rows in the fine payment table
select count(payment_amount)
from library_example.fine_payment;
```

Result Grid | Filter Rov

| | count(payment_amount) |
|---|---|
| ▶ | 3 |

```
-- Selecting the rows with first names that start with letter b from the author table.
select * from library_example.author
where first_name like 'b%';
```

Result Grid | Filter Rows:

| | id | first_name | last_name |
|---|---|---|---|
| ▶ | 2 | Bruce | Wayne |
| * | NULL | NULL | NULL |

```
-- Selecting rows from member table and member_status joining the two tables
select member.first_name, member.last_name, member.joined_date, member_status.status_value
from library_example.member, library_example.member_status
where member.id = member_status.id;
```

Result Grid | Filter Rows: | Export:

| | first_name | last_name | joined_date | status_value |
|---|---|---|---|---|

```
-- Selecting the largest payment value from the fine_payment table
select max(payment_amount) as LargestPayment
from library_example.fine_payment;
```

Result Grid | Filt

| | LargestPayment |
|---|---|
| ▶ | 6000 |

References

https://www.youtube.com/watch?v=yldrIdoXiYk