

CS-GY 6083 - B, Spring 2025

Principles of Database Systems

# PROJECT PART 2

PUNEETH KOTHA (**pk3058**)

JAYRAJ MANOJ PAMNANI (**jmp10051**)

ILKA JEAN (**ifj2007**)

## Table of Contents

<b>SUMMARY.....</b>	<b>2</b>
<b>Business Case.....</b>	<b>2</b>
<b>Approach .....</b>	<b>2</b>
<b>Business Value &amp; Performance Improvement.....</b>	<b>2</b>
<b>ENTITY ASSUMPTIONS .....</b>	<b>5</b>
<b>RELATION ASSUMPTIONS .....</b>	<b>7</b>
<b>Software &amp; Technologies Used:.....</b>	<b>8</b>
<b>DDL Code .....</b>	<b>8</b>
<b>List of tables, and total number of records of each table .....</b>	<b>45</b>
<b>Screenshots our Web Application .....</b>	<b>51</b>
Customer View.....	54
Employee View .....	61
<b>Security Features Implemented.....</b>	<b>64</b>
<b>Lesson Learned: Reflections on Project Work .....</b>	<b>64</b>
<b>Business analysis with 6 SQLs using our project data.....</b>	<b>65</b>

Github Repository Link: [PDS\\_PROJECT\\_GITHUB\\_REPO](#)

# Summary

## Business Case

Modern libraries face increasing challenges in managing their growing collections, diverse user base, and expanding range of services such as book rentals, study room reservations, and event management. Manual or outdated systems often lead to inefficiencies, poor user experience, and missed opportunities for engagement and revenue. The need for a comprehensive, user-friendly, and data-driven solution is critical for libraries to remain relevant and competitive in the digital age.

## Approach

The Chapter Library Management System was developed as a robust, web-based platform using Django, a leading Python web framework. The system is designed to automate and streamline all core library operations, including:

- Book Management: Enables librarians to add, edit, categorize, and track books, including details like authors, publishers, and available copies.
- User Management: Supports registration, authentication, and profile management for customers, employees, and authors, with role-based access.
- Rental System: Automates the process of borrowing and returning books, tracks rental history, and manages overdue items and invoices.
- Study Room Reservations: Allows users to view room availability and make reservations, improving space utilization.
- Event & Seminar Management: Facilitates the organization and registration for library events, workshops, and seminars, enhancing community engagement.
- Analytics & Dashboard: Provides real-time insights into key metrics such as book popularity, rental activity, revenue trends, and overdue items.

The system leverages a normalized relational database (MySQL or SQLite), ensuring data integrity and efficient querying. The user interface is modern, intuitive, and responsive, catering to both staff and patrons. Security best practices, such as password hashing and role-based permissions, are implemented to protect sensitive data.

## Business Value & Performance Improvement

Implementing the Chapter Library Management System delivers significant benefits to library operations and business performance:

- Operational Efficiency: Automation of routine tasks (book tracking, rentals, returns, reservations) reduces manual workload, minimizes errors, and speeds up service delivery.

- Enhanced User Experience: Self-service features (online search, reservations, event registration) improve patron satisfaction and engagement, leading to higher usage and retention.
- Data-Driven Decision Making: Built-in analytics and reporting empower management to make informed decisions on inventory, staffing, and programming, optimizing resource allocation.
- Revenue Growth: Streamlined rental and event management, along with automated invoicing, can increase revenue through better utilization of assets and timely fee collection.
- Scalability & Adaptability: The modular, web-based architecture allows the system to scale with the library's needs and adapt to new services or business models.
- Community Engagement: Event and seminar management features foster a vibrant library community, attracting new users and strengthening the library's role as a cultural hub.

In summary, the Chapter Library Management System transforms traditional library operations into a modern, efficient, and user-centric service, driving both operational excellence and strategic growth.

# ER Model

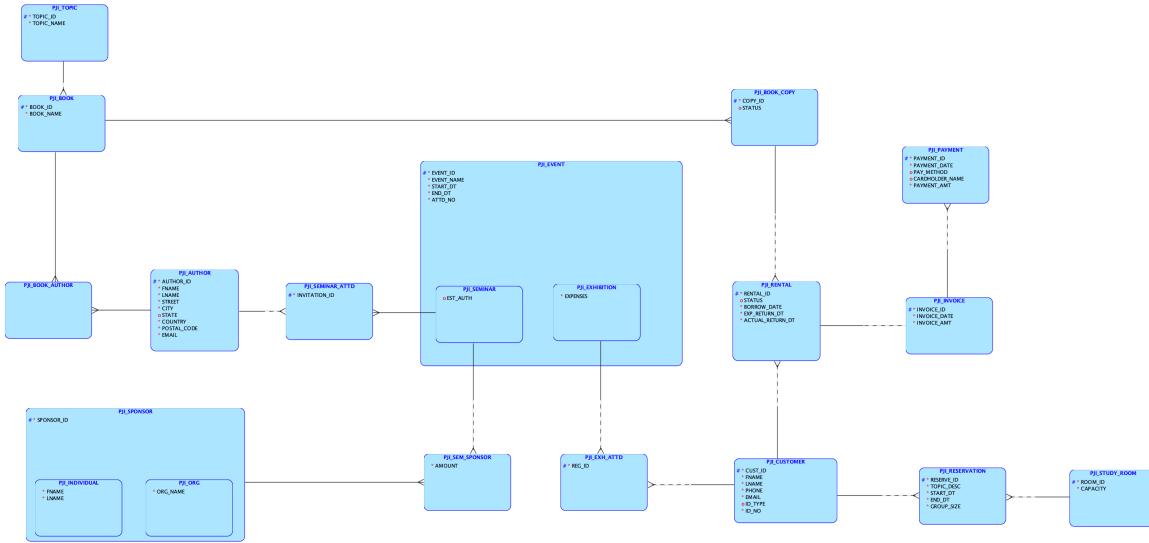


Figure: Logical Model

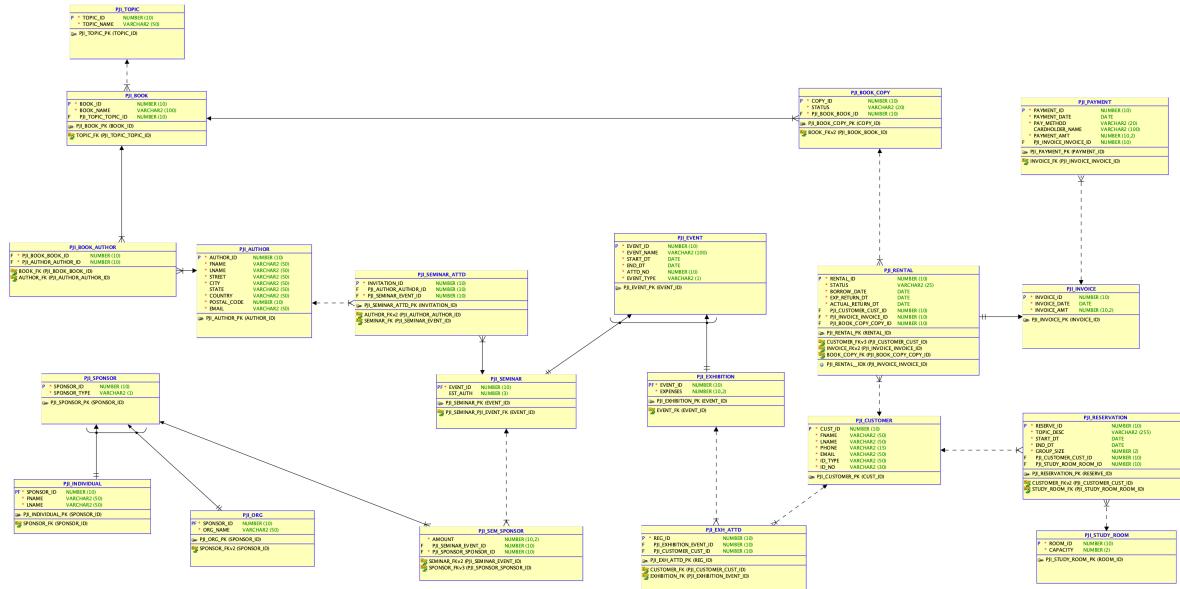


Figure: Relational Model

## ENTITY ASSUMPTIONS

### 1. Customer Identity

A customer must have a unique combination of ID type and ID number.

ID\_TYPE is restricted to only three valid values: 'Passport', 'SSN', or 'Driver License' using a check constraint.

Phone numbers and emails are treated as mandatory and unique identifiers for contact purposes.

### 2. Book Inventory & Classification

Each book is associated with only one topic, but multiple books can belong to the same topic.

The relationship between books and authors is many-to-many, captured via the associative entity PJI\_BOOK\_AUTHOR.

Multiple copies of the same book are tracked independently in PJI\_BOOK\_COPY, each with its own availability status.

### 3. Rental System

A rental record is created per book copy rented by a customer, ensuring each copy's return date is tracked independently.

Rental status includes 'Borrowed', 'Returned', 'Late', and 'Lost' to capture all realistic scenarios.

The invoice is generated per rental and is uniquely associated with one rental.

Rental costs and late fees are calculated automatically through a trigger based on return dates and fixed pricing rules.

It is assumed that the actual return date must be equal to or later than the borrow date.

### 4. Event Management

An EVENT can be of two types: Seminar ('S') or Exhibition ('E'), enforced by a check constraint.

Subtype tables (PJI\_SEMINAR, PJI\_EXHIBITION) are linked using the same primary key as PJI\_EVENT (inheritance modeling).

For seminars, sponsors are recorded, and each sponsor may contribute a monetary amount toward a seminar.

For exhibitions, the cost is tracked as a single EXPENSES value.

### 5. Sponsor Model

Sponsors can be either individuals or organizations, distinguished via the SPONSOR\_TYPE discriminator.

Subtype constraints are enforced with triggers to ensure that sponsors are properly categorized and inserted into their respective subtype tables (PJI\_ORG or PJI\_INDIVIDUAL).

A sponsor cannot simultaneously be recorded as both an organization and an individual.

## 6. Attendance Tracking

Only authors can attend seminars with a valid INVITATION\_ID.

Only customers can attend exhibitions with a valid REG\_ID.

The model does not allow authors to attend exhibitions or customers to attend seminars (this is enforced through table design).

## 7. Payment System

Each invoice can be paid using multiple payment methods.

The system enforces via a trigger that the total payments cannot exceed the invoice amount.

Cardholder name is required only for 'Credit' and 'Debit' payment methods.

If payment is via 'Cash' or 'PayPal', CARDHOLDER\_NAME must be NULL.

## 8. Room Reservations

Only one member of a group is required to register a study room reservation.

Each room reservation must have:

A valid topic description,

A start and end datetime (with END\_DT > START\_DT),

A group size  $\geq 1$ .

There is no overlap checking for reservations in the model—this may be handled at the application layer.

## 9. General Modeling Assumptions

All primary and foreign key relationships are strictly enforced.

All enumerated or categorical values (e.g., ID types, event types, sponsor types) are managed using check constraints rather than separate lookup/reference tables.

Inheritance is modeled using a combination of base and subtype tables, with triggers enforcing correct subtype insertions.

Triggers are used in place of procedural logic to automate and validate data updates (e.g., invoice generation, type enforcement).

## RELATION ASSUMPTIONS

1. PJI\_TOPIC to PJI\_BOOK is kept Target optional as a topic can exist even if there is no book yet of that topic in the library.
2. PJI\_BOOK to PJI\_BOOK\_AUTHOR to PJI\_AUTHOR relationship is kept mandatory as an author can't exist without having a book in our library and if a book is there, its author must be there. For public folklores, the author will be 'UNKNOWN'.
3. PJI\_AUTHOR to PJI\_SEMINAR\_ATTD is kept as target optional as author can exist without getting an invitation for any seminar, but if a seminar is held, then there should be an invitation sent to an author.
4. PJI\_SEMINAR to PJI\_SEMINAR\_ATTD is kept mandatory as seminar can't be organized without authors.
5. PJI\_BOOK to PJI\_BOOKCOPY relationship is kept mandatory as a book can't exist in a library without having a copy.
6. PJI\_CUSTOMER to PJI\_RENTAL is kept as Target optional as a customer can exist even he has not rented out any book yet. Why? Because generally libraries first issue a membership card for registering a customer and then and only then that customer could rent out a book from them.
7. PJI\_CUSTOMER to PJI\_EXH\_ATTD is kept target optional as a customer can exist without attending exhibition.
8. Similarly PJI\_CUSTOMER to PJI\_RESERVATION is also Target optional.
9. PJI\_STUDY\_ROOM to PJI\_RESERVATION is target optional as study rooms will exist without reservations.
10. PJI\_RENTAL to PJI\_INVOICE is mandatory on rental side and optional on invoice side as in libraries, invoices are generated when the customer returns a book after some time of reading it (minimum 1 day in our case), it could happen that he returns a book late due to which the invoice will also include a penalty amount. So the rental could exist without an invoice but an invoice couldn't exist without rental table.
11. PJI\_INVOICE to PJI\_PAYMENT is Target optional as first invoice is generated, which the customer verifies and then he is going to pay for that invoice.

12. PJI\_SEMINAR can be organized without any sponsor.

## Software & Technologies Used:

### Framework:

We have used **Django**, a high-level Python web framework, to build the core of the application. Django provides a robust structure for rapid development, security, and scalability, making it ideal for complex web applications like a library management system.

### Programming Language:

The primary programming language is **Python 3**. Python is known for its readability, simplicity, and strong community support, which accelerates development and maintenance.

### Frontend Technologies:

The user interface is built using **HTML5** and **CSS3** for structure and styling, along with **Bootstrap 5** for responsive design and modern UI components.

**JavaScript** is used to enhance interactivity and provide a seamless user experience.

### Database:

MySQL is supported for production environments, offering scalability, reliability, and advanced features for handling larger datasets and concurrent users.

### Other Tools:

**Git** is used for version control, enabling collaborative development, code tracking, and easy rollback of changes.

**pip** is the Python package manager used to install and manage project dependencies.

## DDL Code

### (MySQL)

```
-- Enable foreign key checks
SET FOREIGN_KEY_CHECKS=1;

-- Create tables
```

```
CREATE TABLE PJI_AUTHOR (
    AUTHOR_ID BIGINT NOT NULL AUTO_INCREMENT COMMENT 'UNIQUE IDENTIFIER FOR AUTHOR',
    FNAME VARCHAR(50) NOT NULL COMMENT 'FIRST NAME OF AUTHOR',
    LNAME VARCHAR(50) NOT NULL COMMENT 'LAST NAME OF AUTHOR',
    STREET VARCHAR(50) NOT NULL COMMENT 'STREET NAME',
    CITY VARCHAR(50) NOT NULL COMMENT 'CITY NAME',
    STATE VARCHAR(50),
    COUNTRY VARCHAR(50) NOT NULL,
    POSTAL_CODE BIGINT NOT NULL,
    EMAIL VARCHAR(50) NOT NULL COMMENT 'EMAIL ADDRESS',
    PRIMARY KEY (AUTHOR_ID)
);
```

```
CREATE TABLE PJI_TOPIC (
    TOPIC_ID BIGINT NOT NULL AUTO_INCREMENT COMMENT 'UNIQUE IDENTIFIER FOR TOPICS',
    TOPIC_NAME VARCHAR(50) NOT NULL COMMENT 'NAME OF THE TOPIC',
    PRIMARY KEY (TOPIC_ID)
);
```

```
CREATE TABLE PJI_BOOK (
    BOOK_ID BIGINT NOT NULL AUTO_INCREMENT COMMENT 'UNIQUE IDENTIFIER FOR BOOK',
    BOOK_NAME VARCHAR(100) NOT NULL COMMENT 'NAME OF THE BOOK',
    PJI_TOPIC_TOPIC_ID BIGINT,
    PRIMARY KEY (BOOK_ID),
    CONSTRAINT FK_PJI_BOOK_TOPIC FOREIGN KEY (PJI_TOPIC_TOPIC_ID) REFERENCES
    PJI_TOPIC(TOPIC_ID)
);
```

```
CREATE TABLE PJI_BOOK_COPY (
    COPY_ID BIGINT NOT NULL AUTO_INCREMENT COMMENT 'UNIQUE IDENTIFIER FOR COPY',
```

```
STATUS      ENUM('available', 'not available') NOT NULL COMMENT 'STATUS OF THE BOOK COPY',
PJI_BOOK_BOOK_ID BIGINT NOT NULL,
PRIMARY KEY (COPY_ID),
CONSTRAINT FK_PJI_BOOK_COPY_BOOK FOREIGN KEY (PJI_BOOK_BOOK_ID) REFERENCES
PJI_BOOK(BOOK_ID)
);
```

```
CREATE TABLE PJI_CUSTOMER (
    CUST_ID BIGINT NOT NULL AUTO_INCREMENT COMMENT 'UNIQUE IDENTIFIER FOR CUSTOMER',
    FNAME  VARCHAR(50) NOT NULL COMMENT 'FIRST NAME OF THE CUSTOMER',
    LNAME  VARCHAR(50) NOT NULL COMMENT 'LAST NAME OF THE CUSTOMER',
    PHONE  VARCHAR(15) NOT NULL COMMENT 'PHONE NUMBER',
    EMAIL  VARCHAR(50) NOT NULL COMMENT 'EMAIL ADDRESS',
    ID_TYPE ENUM('Passport', 'SSN', 'Driver License') NOT NULL COMMENT 'IDENTIFICATION TYPE',
    ID_NO  VARCHAR(30) NOT NULL COMMENT 'IDENTIFIER NUMBER',
    PRIMARY KEY (CUST_ID)
);
```

```
CREATE TABLE PJI_EVENT (
    EVENT_ID  BIGINT NOT NULL AUTO_INCREMENT COMMENT 'UNIQUE IDENTIFIER FOR EVENT',
    EVENT_NAME VARCHAR(100) NOT NULL COMMENT 'NAME OF THE EVENT',
    START_DT  DATETIME NOT NULL COMMENT 'START DATE TIME OF THE EVENT',
    END_DT    DATETIME NOT NULL COMMENT 'END DATE TIME OF THE EVENT',
    ATTD_NO   BIGINT NOT NULL COMMENT 'ESTIMATED NUMBER OF ATTENDEES',
    EVENT_TYPE ENUM('S', 'E') NOT NULL COMMENT 'S for Seminar, E for Exhibition',
    PRIMARY KEY (EVENT_ID)
);
```

```
CREATE TABLE PJI_EXHIBITION (
    EVENT_ID BIGINT NOT NULL COMMENT 'UNIQUE IDENTIFIER FOR EVENT',
```

```

EXPENSES DECIMAL(10,2) NOT NULL COMMENT 'ESTIMATED EXPENSES OF THE EXHIBITION',
PRIMARY KEY (EVENT_ID),
CONSTRAINT FK_PJI_EXHIBITION_EVENT FOREIGN KEY (EVENT_ID) REFERENCES PJI_EVENT(EVENT_ID)
);

CREATE TABLE PJI_SPONSOR (
    SPONSOR_ID BIGINT NOT NULL AUTO_INCREMENT COMMENT 'UNIQUE IDENTIFIER FOR SPONSOR',
    SPONSOR_TYPE ENUM('I', 'O') NOT NULL COMMENT 'I for Individual, O for Organization',
    PRIMARY KEY (SPONSOR_ID)
);

CREATE TABLE PJI_INDIVIDUAL (
    SPONSOR_ID BIGINT NOT NULL COMMENT 'UNIQUE IDENTIFIER FOR SPONSOR',
    FNAME VARCHAR(50) NOT NULL COMMENT 'FIRST NAME OF THE INDIVIDUAL',
    LNAME VARCHAR(50) NOT NULL COMMENT 'LAST NAME OF THE INDIVIDUAL',
    PRIMARY KEY (SPONSOR_ID),
    CONSTRAINT FK_PJI_INDIVIDUAL_SPONSOR FOREIGN KEY (SPONSOR_ID) REFERENCES
PJI_SPONSOR(SPONSOR_ID)
);

CREATE TABLE PJI_ORG (
    SPONSOR_ID BIGINT NOT NULL COMMENT 'UNIQUE IDENTIFIER FOR SPONSOR',
    ORG_NAME VARCHAR(50) NOT NULL COMMENT 'ORGANIZATION NAME',
    PRIMARY KEY (SPONSOR_ID),
    CONSTRAINT FK_PJI_ORG_SPONSOR FOREIGN KEY (SPONSOR_ID) REFERENCES
PJI_SPONSOR(SPONSOR_ID)
);

CREATE TABLE PJI_INVOICE (
    INVOICE_ID BIGINT NOT NULL AUTO_INCREMENT COMMENT 'UNIQUE IDENTIFIER FOR INVOICE',

```

```

INVOICE_DATE DATETIME NOT NULL COMMENT 'DATE OF THE INVOICE',
INVOICE_AMT DECIMAL(10,2) NOT NULL COMMENT 'INVOICE AMOUNT',
PRIMARY KEY (INVOICE_ID)
);

CREATE TABLE PJI_RENTAL (
    RENTAL_ID      BIGINT NOT NULL AUTO_INCREMENT COMMENT 'UNIQUE IDENTIFIER FOR RENTAL',
    STATUS         ENUM('Borrowed', 'Returned', 'Late', 'Lost') NOT NULL COMMENT 'STATUS OF THE RENTAL',
    BORROW_DATE    DATETIME NOT NULL COMMENT 'DATE OF THE BORROWING',
    EXP_RETURN_DT  DATETIME NOT NULL COMMENT 'EXPECTED RETURN DATE',
    ACTUAL_RETURN_DT DATETIME DEFAULT NULL COMMENT 'ACTUAL RETURN DATE',
    PJI_CUSTOMER_CUST_ID BIGINT,
    PJI_INVOICE_INVOICE_ID BIGINT,
    PJI_BOOK_COPY_COPY_ID BIGINT,
    PRIMARY KEY (RENTAL_ID),
    CONSTRAINT FK_PJI_RENTAL_CUSTOMER FOREIGN KEY (PJI_CUSTOMER_CUST_ID) REFERENCES
    PJI_CUSTOMER(CUST_ID),
    CONSTRAINT FK_PJI_RENTAL_INVOICE FOREIGN KEY (PJI_INVOICE_INVOICE_ID) REFERENCES
    PJI_INVOICE(INVOICE_ID),
    CONSTRAINT FK_PJI_RENTAL_BOOK_COPY FOREIGN KEY (PJI_BOOK_COPY_COPY_ID) REFERENCES
    PJI_BOOK_COPY(COPY_ID)
);

CREATE TABLE PJI_PAYMENT (
    PAYMENT_ID      BIGINT NOT NULL AUTO_INCREMENT COMMENT 'UNIQUE IDENTIFIER FOR PAYMENT',
    PAYMENT_DATE    DATETIME NOT NULL COMMENT 'DATE OF PAYMENT',
    PAY_METHOD      ENUM('Cash', 'Credit', 'Debit', 'PayPal') NOT NULL COMMENT 'PAYMENT METHOD',
    CARDHOLDER_NAME VARCHAR(100) COMMENT 'NULLABLE, REQUIRED IF METHOD IS CREDIT/DEBIT',
    PAYMENT_AMT     DECIMAL(10,2) NOT NULL COMMENT 'PAYMENT AMOUNT',
    PJI_INVOICE_INVOICE_ID BIGINT,

```

```

PRIMARY KEY (PAYMENT_ID),
CONSTRAINT FK_PJI_PAYMENT_INVOICE FOREIGN KEY (PJI_INVOICE_INVOICE_ID) REFERENCES
PJI_INVOICE(INVOICE_ID)
);

CREATE TABLE PJI_BOOK_AUTHOR (
    PJI_BOOK_BOOK_ID BIGINT NOT NULL,
    PJI_AUTHOR_AUTHOR_ID BIGINT NOT NULL,
    PRIMARY KEY (PJI_BOOK_BOOK_ID, PJI_AUTHOR_AUTHOR_ID),
    CONSTRAINT FK_PJI_BOOK_AUTHOR_BOOK FOREIGN KEY (PJI_BOOK_BOOK_ID) REFERENCES
PJI_BOOK(BOOK_ID),
    CONSTRAINT FK_PJI_BOOK_AUTHOR_AUTHOR FOREIGN KEY (PJI_AUTHOR_AUTHOR_ID) REFERENCES
PJI_AUTHOR(AUTHOR_ID)
);

CREATE TABLE PJI_STUDY_ROOM (
    ROOM_ID BIGINT NOT NULL AUTO_INCREMENT COMMENT 'UNIQUE IDENTIFIER FOR ROOM',
    CAPACITY TINYINT UNSIGNED NOT NULL COMMENT 'CAPACITY OF THE STUDY ROOM',
    PRIMARY KEY (ROOM_ID)
);

CREATE TABLE PJI_RESERVATION (
    RESERVE_ID BIGINT NOT NULL AUTO_INCREMENT COMMENT 'UNIQUE IDENTIFIER FOR
RESERVATION',
    TOPIC_DESC VARCHAR(255) NOT NULL COMMENT 'DESCRIPTION OF THE TOPIC',
    START_DT DATETIME NOT NULL COMMENT 'START DATE TIME',
    END_DT DATETIME NOT NULL COMMENT 'END DATE TIME',
    GROUP_SIZE TINYINT UNSIGNED NOT NULL COMMENT 'SIZE OF THE GROUP',
    PJI_CUSTOMER_CUST_ID BIGINT,
    PJI_STUDY_ROOM_ROOM_ID BIGINT,
    PRIMARY KEY (RESERVE_ID),

```

```
CONSTRAINT FK_PJI_RESERVATION_CUSTOMER FOREIGN KEY (PJI_CUSTOMER_CUST_ID) REFERENCES  
PJI_CUSTOMER(CUST_ID),
```

```
CONSTRAINT FK_PJI_RESERVATION_STUDY_ROOM FOREIGN KEY (PJI_STUDY_ROOM_ROOM_ID)  
REFERENCES PJI_STUDY_ROOM(ROOM_ID)
```

```
);
```

```
CREATE TABLE PJI_SEMINAR (
```

```
    EVENT_ID BIGINT NOT NULL COMMENT 'UNIQUE IDENTIFIER FOR EVENT',
```

```
    EST_AUTH SMALLINT UNSIGNED COMMENT 'ESTIMATED NUMBER OF AUTHORS PARTICIPATING IN A  
SEMINAR',
```

```
    PRIMARY KEY (EVENT_ID),
```

```
    CONSTRAINT FK_PJI_SEMINAR_EVENT FOREIGN KEY (EVENT_ID) REFERENCES PJI_EVENT(EVENT_ID)
```

```
);
```

```
CREATE TABLE PJI_SEMINAR_ATTD (
```

```
    INVITATION_ID BIGINT NOT NULL AUTO_INCREMENT COMMENT 'UNIQUE IDENTIFIER FOR INVITATION',
```

```
    PJI_AUTHOR_AUTHOR_ID BIGINT,
```

```
    PJI_SEMINAR_EVENT_ID BIGINT NOT NULL,
```

```
    PRIMARY KEY (INVITATION_ID),
```

```
    CONSTRAINT FK_PJI_SEMINAR_ATTD_AUTHOR FOREIGN KEY (PJI_AUTHOR_AUTHOR_ID) REFERENCES  
PJI_AUTHOR(AUTHOR_ID),
```

```
    CONSTRAINT FK_PJI_SEMINAR_ATTD_SEMINAR FOREIGN KEY (PJI_SEMINAR_EVENT_ID) REFERENCES  
PJI_SEMINAR(EVENT_ID)
```

```
);
```

```
CREATE TABLE PJI_SEM_SPONSOR (
```

```
    AMOUNT DECIMAL(10,2) NOT NULL COMMENT 'AMOUNT COST',
```

```
    PJI_SEMINAR_EVENT_ID BIGINT,
```

```
    PJI_SPONSOR_SPONSOR_ID BIGINT NOT NULL,
```

```
    PRIMARY KEY (PJI_SEMINAR_EVENT_ID, PJI_SPONSOR_SPONSOR_ID),
```

```
CONSTRAINT FK_PJI_SEM_SPONSOR_SEMINAR FOREIGN KEY (PJI_SEMINAR_EVENT_ID) REFERENCES  
PJI_SEMINAR(EVENT_ID),
```

```
CONSTRAINT FK_PJI_SEM_SPONSOR_SPONSOR FOREIGN KEY (PJI_SPONSOR_SPONSOR_ID)  
REFERENCES PJI_SPONSOR(SPONSOR_ID)
```

```
);
```

```
CREATE TABLE PJI_EXH_ATTD (
```

```
REG_ID BIGINT NOT NULL AUTO_INCREMENT COMMENT 'REGISTRATION ID',
```

```
PJI_EXHIBITION_EVENT_ID BIGINT,
```

```
PJI_CUSTOMER_CUST_ID BIGINT,
```

```
PRIMARY KEY (REG_ID),
```

```
CONSTRAINT FK_PJI_EXH_ATTD_EXHIBITION FOREIGN KEY (PJI_EXHIBITION_EVENT_ID) REFERENCES  
PJI_EXHIBITION(EVENT_ID),
```

```
CONSTRAINT FK_PJI_EXH_ATTD_CUSTOMER FOREIGN KEY (PJI_CUSTOMER_CUST_ID) REFERENCES  
PJI_CUSTOMER(CUST_ID)
```

```
);
```

```
-- Create triggers for inheritance constraints
```

```
DELIMITER //
```

```
CREATE TRIGGER TRG_PJI_ORG_SPONSOR_TYPE
```

```
BEFORE INSERT ON PJI_ORG
```

```
FOR EACH ROW
```

```
BEGIN
```

```
DECLARE sponsor_type CHAR(1);
```

```
SELECT SPONSOR_TYPE INTO sponsor_type
```

```
FROM PJI_SPONSOR
```

```
WHERE SPONSOR_ID = NEW.SPONSOR_ID;
```

```
IF sponsor_type != 'O' THEN
```

```

SIGNAL SQLSTATE '45000'

SET MESSAGE_TEXT = 'SPONSOR_TYPE must be O for PJI_ORG';

END IF;

END //


CREATE TRIGGER TRG_PJI_INDIVIDUAL_SPONSOR_TYPE
BEFORE INSERT ON PJI_INDIVIDUAL
FOR EACH ROW
BEGIN

DECLARE sponsor_type CHAR(1);

SELECT SPONSOR_TYPE INTO sponsor_type
FROM PJI_SPONSOR
WHERE SPONSOR_ID = NEW.SPONSOR_ID;

IF sponsor_type != 'I' THEN
    SIGNAL SQLSTATE '45000'
    SET MESSAGE_TEXT = 'SPONSOR_TYPE must be I for PJI_INDIVIDUAL';
END IF;

END //


CREATE TRIGGER TRG_PJI_EXHIBITION_EVENT_TYPE
BEFORE INSERT ON PJI_EXHIBITION
FOR EACH ROW
BEGIN

DECLARE event_type CHAR(1);

SELECT EVENT_TYPE INTO event_type
FROM PJI_EVENT

```

```

WHERE EVENT_ID = NEW.EVENT_ID;

IF event_type != 'E' THEN
    SIGNAL SQLSTATE '45000'
    SET MESSAGE_TEXT = 'EVENT_TYPE must be E for PJI_EXHIBITION';
END IF;

END //


CREATE TRIGGER TRG_PJI_SEMINAR_EVENT_TYPE
BEFORE INSERT ON PJI_SEMINAR
FOR EACH ROW
BEGIN
    DECLARE event_type CHAR(1);

    SELECT EVENT_TYPE INTO event_type
    FROM PJI_EVENT
    WHERE EVENT_ID = NEW.EVENT_ID;

    IF event_type != 'S' THEN
        SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'EVENT_TYPE must be S for PJI_SEMINAR';
    END IF;
END //

DELIMITER ;

```

### **(Oracle Data Modeler)**

-- Generated by Oracle SQL Developer Data Modeler 24.3.1.351.0831

```

-- at: 2025-03-30 20:37:54 EDT
-- site: Oracle Database 21c
-- type: Oracle Database 21c

-- predefined type, no DDL - MDSYS.SDO_GEOOMETRY
-- predefined type, no DDL - XMLTYPE

CREATE TABLE PJI_AUTHOR
(
    AUTHOR_ID NUMBER (10) NOT NULL ,
    FNAME VARCHAR2 (50) NOT NULL ,
    LNAME VARCHAR2 (50) NOT NULL ,
    STREET VARCHAR2 (50) NOT NULL ,
    CITY VARCHAR2 (50) NOT NULL ,
    STATE VARCHAR2 (50) ,
    COUNTRY VARCHAR2 (50) NOT NULL ,
    POSTAL_CODE NUMBER (10) NOT NULL ,
    EMAIL VARCHAR2 (50) NOT NULL
)
;

COMMENT ON COLUMN PJI_AUTHOR.AUTHOR_ID IS 'UNIQUE IDENTIFIER FOR AUTHOR'
;

COMMENT ON COLUMN PJI_AUTHOR.FNAME IS 'FIRST NAME OF AUTHOR'
;

COMMENT ON COLUMN PJI_AUTHOR.LNAME IS 'LAST NAME OF AUTHOR'
;

```

```
COMMENT ON COLUMN PJI_AUTHOR.STREET IS 'STREET NAME'
;

COMMENT ON COLUMN PJI_AUTHOR.CITY IS 'CITY NAME'
;

COMMENT ON COLUMN PJI_AUTHOR.EMAIL IS 'EMAIL ADDRESS'
;

ALTER TABLE PJI_AUTHOR
ADD CONSTRAINT PJI_AUTHOR_PK PRIMARY KEY ( AUTHOR_ID ) ;

CREATE TABLE PJI_BOOK
(
    BOOK_ID      NUMBER (10) NOT NULL ,
    BOOK_NAME    VARCHAR2 (100) NOT NULL ,
    PJI_TOPIC_TOPIC_ID NUMBER (10)
)
;

COMMENT ON COLUMN PJI_BOOK.BOOK_ID IS 'UNIQUE IDENTIFIER FOR BOOK'
;

COMMENT ON COLUMN PJI_BOOK.BOOK_NAME IS 'NAME OF THE BOOK'
;

ALTER TABLE PJI_BOOK
ADD CONSTRAINT PJI_BOOK_PK PRIMARY KEY ( BOOK_ID ) ;
```

```

CREATE TABLE PJI_BOOK_AUTHOR
(
  PJI_BOOK_BOOK_ID NUMBER (10) NOT NULL ,
  PJI_AUTHOR_AUTHOR_ID NUMBER (10) NOT NULL
)
;

CREATE TABLE PJI_BOOK_COPY
(
  COPY_ID      NUMBER (10) NOT NULL ,
  STATUS       VARCHAR2 (20) NOT NULL ,
  PJI_BOOK_BOOK_ID NUMBER (10) NOT NULL
)
;

COMMENT ON COLUMN PJI_BOOK_COPY.COPY_ID IS 'UNIQUE IDENTIFIER FOR COPY'
;

COMMENT ON COLUMN PJI_BOOK_COPY.STATUS IS 'STATUS OF THE BOOK COPY'
;

ALTER TABLE PJI_BOOK_COPY
  ADD CONSTRAINT PJI_BOOK_COPY_PK PRIMARY KEY ( COPY_ID );

CREATE TABLE PJI_CUSTOMER
(
  CUST_ID NUMBER (10) NOT NULL ,
  FNAME  VARCHAR2 (50) NOT NULL ,

```

```
LNAME VARCHAR2(50) NOT NULL,  
PHONE VARCHAR2(15) NOT NULL,  
EMAIL VARCHAR2(50) NOT NULL,  
ID_TYPE VARCHAR2(50) NOT NULL,  
ID_NO VARCHAR2(30) NOT NULL  
)  
;  
  
COMMENT ON COLUMN PJI_CUSTOMER.CUST_ID IS 'UNIQUE IDENTIFIER FOR CUSTOMER'  
;  
  
COMMENT ON COLUMN PJI_CUSTOMER.FNAME IS 'FIRST NAME OF THE CUSTOMER'  
;  
  
COMMENT ON COLUMN PJI_CUSTOMER.LNAME IS 'LAST NAME OF THE CUSTOMER'  
;  
  
COMMENT ON COLUMN PJI_CUSTOMER.PHONE IS 'PHONE NUMBER'  
;  
  
COMMENT ON COLUMN PJI_CUSTOMER.EMAIL IS 'EMAIL ADDRESS'  
;  
  
COMMENT ON COLUMN PJI_CUSTOMER.ID_TYPE IS 'IDENTIFICATION TYPE OF THE ID LIKE PASSPORT, SSN  
OR DRIVER'S LICENSE'  
;  
  
COMMENT ON COLUMN PJI_CUSTOMER.ID_NO IS 'IDENTIFIER NUMBER'  
;
```

```

ALTER TABLE PJI_CUSTOMER
ADD CONSTRAINT PJI_CUSTOMER_PK PRIMARY KEY ( CUST_ID ) ;

CREATE TABLE PJI_EVENT
(
    EVENT_ID NUMBER (10) NOT NULL ,
    EVENT_NAME VARCHAR2 (100) NOT NULL ,
    START_DT DATE NOT NULL ,
    END_DT DATE NOT NULL ,
    ATTD_NO NUMBER (10) NOT NULL ,
    EVENT_TYPE VARCHAR2 (1) NOT NULL
)
;

ALTER TABLE PJI_EVENT
ADD CONSTRAINT CH_INH_PJI_EVENT
CHECK (EVENT_TYPE IN ('E', 'S'))
;

COMMENT ON COLUMN PJI_EVENT.EVENT_ID IS 'UNIQUE IDENTIFIER FOR EVENT'
;

COMMENT ON COLUMN PJI_EVENT.EVENT_NAME IS 'NAME OF THE EVENT'
;

COMMENT ON COLUMN PJI_EVENT.START_DT IS 'START DATE TIME OF THE EVENT'
;

COMMENT ON COLUMN PJI_EVENT.END_DT IS 'END DATE TIME OF THE EVENT'
;

```

```
;
```

```
COMMENT ON COLUMN PJI_EVENT.ATTD_NO IS 'ESTIMATED NUMBER OF ATTENDEES'
```

```
;
```

```
ALTER TABLE PJI_EVENT
```

```
ADD CONSTRAINT PJI_EVENT_PK PRIMARY KEY ( EVENT_ID );
```

```
CREATE TABLE PJI_EXH_ATTD
```

```
(
```

```
REG_ID      NUMBER (10) NOT NULL ,
```

```
PJI_EXHIBITION_EVENT_ID NUMBER (10) ,
```

```
PJI_CUSTOMER_CUST_ID  NUMBER (10)
```

```
)
```

```
;
```

```
COMMENT ON COLUMN PJI_EXH_ATTD.REG_ID IS 'REGISTRATIION ID'
```

```
;
```

```
ALTER TABLE PJI_EXH_ATTD
```

```
ADD CONSTRAINT PJI_EXH_ATTD_PK PRIMARY KEY ( REG_ID );
```

```
CREATE TABLE PJI_EXHIBITION
```

```
(
```

```
EVENT_ID NUMBER (10) NOT NULL ,
```

```
EXPENSES NUMBER (10,2) NOT NULL
```

```
)
```

```
;
```

```
COMMENT ON COLUMN PJI_EXHIBITION.EVENT_ID IS 'UNIQUE IDENTIFIER FOR EVENT'
;

COMMENT ON COLUMN PJI_EXHIBITION.EXPENSES IS 'ESTIMATED EXPENSES OF THE EXHIBITION'
;

ALTER TABLE PJI_EXHIBITION
    ADD CONSTRAINT PJI_EXHIBITION_PK PRIMARY KEY ( EVENT_ID ) ;

CREATE TABLE PJI_INDIVIDUAL
(
    SPONSOR_ID NUMBER (10) NOT NULL ,
    FNAME    VARCHAR2 (50) NOT NULL ,
    LNAME    VARCHAR2 (50) NOT NULL
)
;

COMMENT ON COLUMN PJI_INDIVIDUAL.SPONSOR_ID IS 'UNIQUE IDENTIFIER FOR SPONSOR'
;

COMMENT ON COLUMN PJI_INDIVIDUAL.FNAME IS 'FIRST NAME OF THE INDIVIDUAL'
;

COMMENT ON COLUMN PJI_INDIVIDUAL.LNAME IS 'LAST NAME OF THE INDIVIDUAL'
;

ALTER TABLE PJI_INDIVIDUAL
    ADD CONSTRAINT PJI_INDIVIDUAL_PK PRIMARY KEY ( SPONSOR_ID ) ;
```

```

CREATE TABLE PJI_INVOICE
(
    INVOICE_ID NUMBER (10) NOT NULL ,
    INVOICE_DATE DATE NOT NULL ,
    INVOICE_AMT NUMBER (10,2) NOT NULL
)
;

COMMENT ON COLUMN PJI_INVOICE.INVOICE_ID IS 'UNIQUE IDENTIFIER FOR INVOICE'
;

COMMENT ON COLUMN PJI_INVOICE.INVOICE_DATE IS 'DATE OF THE INVOICE'
;

COMMENT ON COLUMN PJI_INVOICE.INVOICE_AMT IS 'INVOICE AMOUNT'
;

ALTER TABLE PJI_INVOICE
ADD CONSTRAINT PJI_INVOICE_PK PRIMARY KEY ( INVOICE_ID ) ;

CREATE TABLE PJI_ORG
(
    SPONSOR_ID NUMBER (10) NOT NULL ,
    ORG_NAME VARCHAR2 (50) NOT NULL
)
;

COMMENT ON COLUMN PJI_ORG.SPONSOR_ID IS 'UNIQUE IDENTIFIER FOR SPONSOR'
;

```

```
COMMENT ON COLUMN PJI_ORG.ORG_NAME IS 'ORGANIZATION NAME'
```

```
;
```

```
ALTER TABLE PJI_ORG
```

```
ADD CONSTRAINT PJI_ORG_PK PRIMARY KEY ( SPONSOR_ID );
```

```
CREATE TABLE PJI_PAYMENT
```

```
(
```

```
    PAYMENT_ID      NUMBER (10) NOT NULL ,
```

```
    PAYMENT_DATE    DATE NOT NULL ,
```

```
    PAY_METHOD      VARCHAR2 (20) NOT NULL ,
```

```
    CARDHOLDER_NAME VARCHAR2 (100) ,
```

```
    PAYMENT_AMT     NUMBER (10,2) NOT NULL ,
```

```
    PJI_INVOICE_INVOICE_ID NUMBER (10)
```

```
)
```

```
;
```

```
COMMENT ON COLUMN PJI_PAYMENT.PAYMENT_ID IS 'UNIQUE IDENTIFIER FOR PAYMENT'
```

```
;
```

```
COMMENT ON COLUMN PJI_PAYMENT.PAYMENT_DATE IS 'DATE OF PAYMENT'
```

```
;
```

```
COMMENT ON COLUMN PJI_PAYMENT.PAY_METHOD IS 'PAYMENT METHOD EG. DEBIT CARD, CREDIT CARD,  
PAYPAL, CASH, ETC.'
```

```
;
```

```
COMMENT ON COLUMN PJI_PAYMENT.CARDHOLDER_NAME IS 'NULLABLE, REQUIRED IF METHOD IS  
CREDIT/DEBIT'
```

```
;
```

```
COMMENT ON COLUMN PJI_PAYMENT.PAYMENT_AMT IS 'PAYMENT AMOUNT'
```

```
;
```

```
ALTER TABLE PJI_PAYMENT
```

```
ADD CONSTRAINT PJI_PAYMENT_PK PRIMARY KEY ( PAYMENT_ID );
```

```
CREATE TABLE PJI_RENTAL
```

```
(
```

```
RENTAL_ID      NUMBER (10) NOT NULL ,
```

```
STATUS         VARCHAR2 (25) NOT NULL ,
```

```
BORROW_DATE    DATE NOT NULL ,
```

```
EXP_RETURN_DT   DATE NOT NULL ,
```

```
ACTUAL_RETURN_DT DATE NOT NULL ,
```

```
PJI_CUSTOMER_CUST_ID NUMBER (10) ,
```

```
PJI_INVOICE_INVOICE_ID NUMBER (10) NOT NULL ,
```

```
PJI_BOOK_COPY_COPY_ID NUMBER (10)
```

```
)
```

```
;
```

```
COMMENT ON COLUMN PJI_RENTAL.RENTAL_ID IS 'UNIQUE IDENTIFIER FOR RENTAL'
```

```
;
```

```
COMMENT ON COLUMN PJI_RENTAL.STATUS IS 'STATUS OF THE RENTAL OF THE BOOK COPY AS AVAILABLE  
AND NOT AVAILABLE'
```

```
;
```

```
COMMENT ON COLUMN PJI_RENTAL.BORROW_DATE IS 'DATE OF THE BORROWING'
```

```
;
```

```
COMMENT ON COLUMN PJI_RENTAL.EXP_RETURN_DT IS 'EXPECTED RETURN DATE'
;

COMMENT ON COLUMN PJI_RENTAL.ACTUAL_RETURN_DT IS 'ACTUAL RETURN DATE'
;

CREATE UNIQUE INDEX PJI_RENTAL__IDX ON PJI_RENTAL
(
    PJI_INVOICE_INVOICE_ID ASC
)
;

ALTER TABLE PJI_RENTAL
    ADD CONSTRAINT PJI_RENTAL_PK PRIMARY KEY ( RENTAL_ID ) ;

CREATE TABLE PJI_RESERVATION
(
    RESERVE_ID      NUMBER (10) NOT NULL ,
    TOPIC_DESC      VARCHAR2 (255) NOT NULL ,
    START_DT        DATE NOT NULL ,
    END_DT          DATE NOT NULL ,
    GROUP_SIZE      NUMBER (2) NOT NULL ,
    PJI_CUSTOMER_CUST_ID  NUMBER (10) ,
    PJI_STUDY_ROOM_ROOM_ID NUMBER (10)
)
;

COMMENT ON COLUMN PJI_RESERVATION.RESERVE_ID IS 'UNIQUE IDENTIFIER FOR RESERVATION'
;
```

```
COMMENT ON COLUMN PJI_RESERVATION.TOPIC_DESC IS 'DESCRIPTION OF THE TOPIC'
;

COMMENT ON COLUMN PJI_RESERVATION.START_DT IS 'START DATE TIME'
;

COMMENT ON COLUMN PJI_RESERVATION.END_DT IS 'END DATE TIME'
;

COMMENT ON COLUMN PJI_RESERVATION.GROUP_SIZE IS 'SIZE OF THE GROUP'
;

ALTER TABLE PJI_RESERVATION
ADD CONSTRAINT PJI_RESERVATION_PK PRIMARY KEY ( RESERVE_ID );

CREATE TABLE PJI_SEM_SPONSOR
(
    AMOUNT      NUMBER (10,2) NOT NULL ,
    PJI_SEMINAR_EVENT_ID  NUMBER (10) ,
    PJI_SPONSOR_SPONSOR_ID NUMBER (10) NOT NULL
)
;

COMMENT ON COLUMN PJI_SEM_SPONSOR.AMOUNT IS 'AMOUNT COST'
;

CREATE TABLE PJI_SEMINAR
(

```

```

EVENT_ID NUMBER (10) NOT NULL ,
EST_AUTH NUMBER (3)

)

;

COMMENT ON COLUMN PJI_SEMINAR.EVENT_ID IS 'UNIQUE IDENTIFIER FOR EVENT'
;

COMMENT ON COLUMN PJI_SEMINAR.EST_AUTH IS 'ESTIMATED NUMBER OF AUTHORS PARTICIPATING IN A SEMINAR'
;

ALTER TABLE PJI_SEMINAR
ADD CONSTRAINT PJI_SEMINAR_PK PRIMARY KEY ( EVENT_ID ) ;

CREATE TABLE PJI_SEMINAR_ATTD
(
INVITATION_ID      NUMBER (10) NOT NULL ,
PJI_AUTHOR_AUTHOR_ID NUMBER (10) ,
PJI_SEMINAR_EVENT_ID NUMBER (10) NOT NULL
)
;

COMMENT ON COLUMN PJI_SEMINAR_ATTD.INVITATION_ID IS 'UNIQUE IDENTIFIER FOR INVITATION'
;

ALTER TABLE PJI_SEMINAR_ATTD
ADD CONSTRAINT PJI_SEMINAR_ATTD_PK PRIMARY KEY ( INVITATION_ID ) ;

CREATE TABLE PJI_SPONSOR
;
```

```

(
    SPONSOR_ID NUMBER (10) NOT NULL ,
    SPONSOR_TYPE VARCHAR2 (1) NOT NULL
)
;

ALTER TABLE PJI_SPONSOR
    ADD CONSTRAINT CH_INH_PJI_SPONSOR
        CHECK (SPONSOR_TYPE IN ('I', 'O'))
;

COMMENT ON COLUMN PJI_SPONSOR.SPONSOR_ID IS 'UNIQUE IDENTIFIER FOR SPONSOR'
;

ALTER TABLE PJI_SPONSOR
    ADD CONSTRAINT PJI_SPONSOR_PK PRIMARY KEY ( SPONSOR_ID ) ;

CREATE TABLE PJI_STUDY_ROOM
(
    ROOM_ID NUMBER (10) NOT NULL ,
    CAPACITY NUMBER (2) NOT NULL
)
;

COMMENT ON COLUMN PJI_STUDY_ROOM.ROOM_ID IS 'UNIQUE IDENTIFIER FOR ROOM'
;

COMMENT ON COLUMN PJI_STUDY_ROOM.CAPACITY IS 'CAPACITY OF THE STUDY ROOM'
;

```

```
ALTER TABLE PJI_STUDY_ROOM
    ADD CONSTRAINT PJI_STUDY_ROOM_PK PRIMARY KEY ( ROOM_ID ) ;

CREATE TABLE PJI_TOPIC
(
    TOPIC_ID NUMBER (10) NOT NULL ,
    TOPIC_NAME VARCHAR2 (50) NOT NULL
)
;

COMMENT ON COLUMN PJI_TOPIC.TOPIC_ID IS 'UNIQUE IDENTIFIER FOR TOPICS'
;

COMMENT ON COLUMN PJI_TOPIC.TOPIC_NAME IS 'NAME OF THE TOPIC'
;

ALTER TABLE PJI_TOPIC
    ADD CONSTRAINT PJI_TOPIC_PK PRIMARY KEY ( TOPIC_ID ) ;

ALTER TABLE PJI_BOOK_AUTHOR
    ADD CONSTRAINT AUTHOR_FK FOREIGN KEY
(
    PJI_AUTHOR_AUTHOR_ID
)
REFERENCES PJI_AUTHOR
(
    AUTHOR_ID
)
```

```
;
```

```
ALTER TABLE PJI_SEMINAR_ATTD  
ADD CONSTRAINT AUTHOR_FKv2 FOREIGN KEY  
(  
PJI_AUTHOR_AUTHOR_ID  
)  
REFERENCES PJI_AUTHOR  
(  
AUTHOR_ID  
)  
;
```

```
ALTER TABLE PJI_RENTAL  
ADD CONSTRAINT BOOK_COPY_FK FOREIGN KEY  
(  
PJI_BOOK_COPY_COPY_ID  
)  
REFERENCES PJI_BOOK_COPY  
(  
COPY_ID  
)  
;
```

```
ALTER TABLE PJI_BOOK_AUTHOR  
ADD CONSTRAINT BOOK_FK FOREIGN KEY  
(  
PJI_BOOK_BOOK_ID  
)
```

```
REFERENCES PJI_BOOK
(
BOOK_ID
)
;

ALTER TABLE PJI_BOOK_COPY
ADD CONSTRAINT BOOK_FKv2 FOREIGN KEY
(
PJI_BOOK_BOOK_ID
)
REFERENCES PJI_BOOK
(
BOOK_ID
)
;

ALTER TABLE PJI_EXH_ATTD
ADD CONSTRAINT CUSTOMER_FK FOREIGN KEY
(
PJI_CUSTOMER_CUST_ID
)
REFERENCES PJI_CUSTOMER
(
CUST_ID
)
;

ALTER TABLE PJI_RESERVATION
```

```
ADD CONSTRAINT CUSTOMER_FKv2 FOREIGN KEY
(
    PJI_CUSTOMER_CUST_ID
)
REFERENCES PJI_CUSTOMER
(
    CUST_ID
)
;
```

```
ALTER TABLE PJI_RENTAL
ADD CONSTRAINT CUSTOMER_FKv3 FOREIGN KEY
(
    PJI_CUSTOMER_CUST_ID
)
REFERENCES PJI_CUSTOMER
(
    CUST_ID
)
;
```

```
ALTER TABLE PJI_EXHIBITION
ADD CONSTRAINT EVENT_FK FOREIGN KEY
(
    EVENT_ID
)
REFERENCES PJI_EVENT
(
    EVENT_ID
)
```

```
)  
;  
  
ALTER TABLE PJI_EXH_ATTD  
    ADD CONSTRAINT EXHIBITION_FK FOREIGN KEY  
    (  
        PJI_EXHIBITION_EVENT_ID  
    )  
    REFERENCES PJI_EXHIBITION  
    (  
        EVENT_ID  
    )  
;  
  
ALTER TABLE PJI_PAYMENT  
    ADD CONSTRAINT INVOICE_FK FOREIGN KEY  
    (  
        PJI_INVOICE_INVOICE_ID  
    )  
    REFERENCES PJI_INVOICE  
    (  
        INVOICE_ID  
    )  
;  
  
ALTER TABLE PJI_RENTAL  
    ADD CONSTRAINT INVOICE_FKv2 FOREIGN KEY  
    (  
        PJI_INVOICE_INVOICE_ID
```

```
)  
REFERENCES PJI_INVOICE  
(  
INVOICE_ID  
)  
;  
  
ALTER TABLE PJI_SEMINAR  
ADD CONSTRAINT PJI_SEMINAR_PJI_EVENT_FK FOREIGN KEY  
(  
EVENT_ID  
)  
REFERENCES PJI_EVENT  
(  
EVENT_ID  
)  
;  
  
ALTER TABLE PJI_SEMINAR_ATTD  
ADD CONSTRAINT SEMINAR_FK FOREIGN KEY  
(  
PJI_SEMINAR_EVENT_ID  
)  
REFERENCES PJI_SEMINAR  
(  
EVENT_ID  
)  
;
```

```
ALTER TABLE PJI_SEM_SPONSOR  
ADD CONSTRAINT SEMINAR_FKv2 FOREIGN KEY  
(  
PJI_SEMINAR_EVENT_ID  
)  
REFERENCES PJI_SEMINAR  
(  
EVENT_ID  
)  
;
```

```
ALTER TABLE PJI_INDIVIDUAL  
ADD CONSTRAINT SPONSOR_FK FOREIGN KEY  
(  
SPONSOR_ID  
)  
REFERENCES PJI_SPONSOR  
(  
SPONSOR_ID  
)  
;
```

```
ALTER TABLE PJI_ORG  
ADD CONSTRAINT SPONSOR_FKv2 FOREIGN KEY  
(  
SPONSOR_ID  
)  
REFERENCES PJI_SPONSOR  
(
```

```
SPONSOR_ID
)
;

ALTER TABLE PJI_SEM_SPONSOR
ADD CONSTRAINT SPONSOR_FKv3 FOREIGN KEY
(
PJI_SPONSOR_SPONSOR_ID
)
REFERENCES PJI_SPONSOR
(
SPONSOR_ID
)
;

ALTER TABLE PJI_RESERVATION
ADD CONSTRAINT STUDY_ROOM_FK FOREIGN KEY
(
PJI_STUDY_ROOM_ROOM_ID
)
REFERENCES PJI_STUDY_ROOM
(
ROOM_ID
)
;

ALTER TABLE PJI_BOOK
ADD CONSTRAINT TOPIC_FK FOREIGN KEY
(
```

```

PJI_TOPIC_TOPIC_ID
)
REFERENCES PJI_TOPIC
(
TOPIC_ID
)
;

CREATE OR REPLACE TRIGGER ARC_FKArc_1_PJI_EXHIBITION
BEFORE INSERT OR UPDATE OF EVENT_ID
ON PJI_EXHIBITION
FOR EACH ROW
DECLARE
d VARCHAR2 (1);
BEGIN
SELECT A.EVENT_TYPE INTO d
FROM PJI_EVENT A
WHERE A.EVENT_ID = :new.EVENT_ID;
IF (d IS NULL OR d <> 'E') THEN
    raise_application_error(-20223,'FK EVENT_FK in Table PJI_EXHIBITION violates Arc constraint on Table PJI_EVENT - discriminator column EVENT_TYPE doesn''t have value "E"');
END IF;
EXCEPTION
WHEN NO_DATA_FOUND THEN
NULL;
WHEN OTHERS THEN
RAISE;
END;
/

```

```

CREATE OR REPLACE TRIGGER ARC_FKArc_1_PJI_SEMINAR
BEFORE INSERT OR UPDATE OF EVENT_ID
ON PJI_SEMINAR
FOR EACH ROW
DECLARE
    d VARCHAR2 (1);
BEGIN
    SELECT A.EVENT_TYPE INTO d
    FROM PJI_EVENT A
    WHERE A.EVENT_ID = :new.EVENT_ID;
    IF (d IS NULL OR d <> 'S') THEN
        raise_application_error(-20223,'FK PJI_SEMINAR_PJI_EVENT_FK in Table PJI_SEMINAR violates Arc constraint on Table PJI_EVENT - discriminator column EVENT_TYPE doesn''t have value "S"');
    END IF;
    EXCEPTION
    WHEN NO_DATA_FOUND THEN
        NULL;
    WHEN OTHERS THEN
        RAISE;
END;
/

```

```

CREATE OR REPLACE TRIGGER ARC_FKArc_2_PJI_ORG
BEFORE INSERT OR UPDATE OF SPONSOR_ID
ON PJI_ORG
FOR EACH ROW
DECLARE
    d VARCHAR2 (1);
BEGIN
    SELECT A.SPONSOR_TYPE INTO d

```

```

FROM PJI_SPONSOR A

WHERE A.SPONSOR_ID = :new.SPONSOR_ID;

IF (d IS NULL OR d <> 'O') THEN

    raise_application_error(-20223,'FK SPONSOR_FKv2 in Table PJI_ORG violates Arc constraint on Table
PJI_SPONSOR - discriminator column SPONSOR_TYPE doesn't have value "O"');

END IF;

EXCEPTION

WHEN NO_DATA_FOUND THEN

NULL;

WHEN OTHERS THEN

RAISE;

END;

/

```

```

CREATE OR REPLACE TRIGGER ARC_FKArc_2_PJI_INDIVIDUAL

BEFORE INSERT OR UPDATE OF SPONSOR_ID

ON PJI_INDIVIDUAL

FOR EACH ROW

DECLARE

d VARCHAR2 (1);

BEGIN

SELECT A.SPONSOR_TYPE INTO d

FROM PJI_SPONSOR A

WHERE A.SPONSOR_ID = :new.SPONSOR_ID;

IF (d IS NULL OR d <> 'I') THEN

    raise_application_error(-20223,'FK SPONSOR_FK in Table PJI_INDIVIDUAL violates Arc constraint on Table
PJI_SPONSOR - discriminator column SPONSOR_TYPE doesn't have value "I"');

END IF;

EXCEPTION

WHEN NO_DATA_FOUND THEN

```

```

NULL;

WHEN OTHERS THEN

    RAISE;

END;

/


-- Oracle SQL Developer Data Modeler Summary Report:

--

-- CREATE TABLE          20
-- CREATE INDEX          1
-- ALTER TABLE          40
-- CREATE VIEW           0
-- ALTER VIEW            0
-- CREATE PACKAGE        0
-- CREATE PACKAGE BODY   0
-- CREATE PROCEDURE      0
-- CREATE FUNCTION       0
-- CREATE TRIGGER        4
-- ALTER TRIGGER         0
-- CREATE COLLECTION TYPE 0
-- CREATE STRUCTURED TYPE 0
-- CREATE STRUCTURED TYPE BODY 0
-- CREATE CLUSTER        0
-- CREATE CONTEXT         0
-- CREATE DATABASE        0
-- CREATE DIMENSION       0
-- CREATE DIRECTORY        0
-- CREATE DISK GROUP      0
-- CREATE ROLE             0

```

```
-- CREATE ROLLBACK SEGMENT          0
-- CREATE SEQUENCE                  0
-- CREATE MATERIALIZED VIEW        0
-- CREATE MATERIALIZED VIEW LOG    0
-- CREATE SYNONYM                  0
-- CREATE TABLESPACE                0
-- CREATE USER                      0
--
-- DROP TABLESPACE                 0
-- DROP DATABASE                   0
--
-- REDACTION POLICY               0
--
-- ORDS DROP SCHEMA                0
-- ORDS ENABLE SCHEMA              0
-- ORDS ENABLE OBJECT              0
--
-- ERRORS                          0
-- WARNINGS                        0
```

## List of tables, and total number of records of each table

TABLE_NAME	TABLE_DESCRIPTION
PJI_AUTHOR	Stores information about book authors including...
PJI_BOOK	Contains book information and their topic associ...
PJI_BOOK_AUTHOR	Junction table linking books to their authors (ma...
PJI_BOOK_COPY	Tracks individual copies of books and their curre...
PJI_CUSTOMER	Stores customer information and identification d...
PJI_EVENT	Base table for all library events (exhibitions and...
PJI_EXH_ATTD	Tracks attendance records for exhibition events
PJI_EXHIBITION	Specific information for exhibition events includi...
PJI_INDIVIDUAL	Stores information about individual sponsors
PJI_INVOICE	Tracks rental invoices and their amounts
PJI_ORG	Stores information about organizational sponsors
PJI_PAYMENT	Records payments made for invoices with paym...
PJI_RENTAL	Tracks book rentals, return dates, and rental sta...
PJI_RESERVATION	Manages study room reservations and group siz...
PJI_SEM_SPONSOR	Links seminars to their sponsors with sponsorsh...
PJI_SEMINAR	Specific information for seminar events includin...
PJI_SEMINAR_ATTD	Tracks attendance records for seminar events
PJI_SPONSOR	Base table for all event sponsors (individual or o...
PJI_STUDY_ROOM	Information about available study rooms and the...
PJI_TOPIC	Categories or topics for books

*Image: List of the tables*

Below are the images of the total count of records in each of the tables.

```

1   -- Count Authors
2 *  SELECT 'PJI_AUTHOR' as TABLE_NAME, COUNT(*) as RECORD_COUNT FROM PJI_AUTHOR;
3

100%  1:3

Result Grid  Filter Rows: Search Export:

| TABLE_NAME | RECORD_COUNT |
| PJI_AUTHOR | 20           |

1
2   -- Count Books
3 *  SELECT 'PJI_BOOK' as TABLE_NAME, COUNT(*) as RECORD_COUNT FROM PJI_BOOK;
4

100%  1:4

Result Grid  Filter Rows: Search Export:

| TABLE_NAME | RECORD_COUNT |
| PJI_BOOK   | 20           |

1
2   -- Count Book-Author Relationships
3 *  SELECT 'PJI_BOOK_AUTHOR' as TABLE_NAME, COUNT(*) as RECORD_COUNT FROM PJI_BOOK_AUTHOR;
4

100%  1:4

Result Grid  Filter Rows: Search Export:

| TABLE_NAME | RECORD_COUNT |
| PJI_BOOK_AUTHOR | 20           |

1
2   -- Count Book Copies
3 *  SELECT 'PJI_BOOK_COPY' as TABLE_NAME, COUNT(*) as RECORD_COUNT FROM PJI_BOOK_COPY;
4

```

100% 1:4

Result Grid Filter Rows: Search Export:

TABLE_NAME	RECORD_COUNT
PJI_BOOK_COPY	40

```
1
2    -- Count Customers
3    SELECT 'PJI_CUSTOMER' as TABLE_NAME, COUNT(*) as RECORD_COUNT FROM PJI_CUSTOMER;
4
```

100% 1:4

Result Grid Filter Rows: Search Export:

TABLE_NAME	RECORD_COUNT
PJI_CUSTOMER	20

```
1
2    -- Count Events
3    SELECT 'PJI_EVENT' as TABLE_NAME, COUNT(*) as RECORD_COUNT FROM PJI_EVENT;
4
```

100% 1:4

Result Grid Filter Rows: Search Export:

TABLE_NAME	RECORD_COUNT
PJI_EVENT	20

```
1
2    -- Count Exhibition Attendance
3    SELECT 'PJI_EXH_ATTD' as TABLE_NAME, COUNT(*) as RECORD_COUNT FROM PJI_EXH_ATTD;
4
```

100% 1:4

Result Grid Filter Rows: Search Export:

TABLE_NAME	RECORD_COUNT
PJI_EXH_ATTD	20

```
1
2    -- Count Exhibitions
3    SELECT 'PJI_EXHIBITION' as TABLE_NAME, COUNT(*) as RECORD_COUNT FROM PJI_EXHIBITION;
4
```

100% 1:4

Result Grid Filter Rows: Search Export:

TABLE_NAME	RECORD_COUNT
PJI_EXHIBITION	10

```
1
2    -- Count Individual Sponsors
3    SELECT 'PJI_INDIVIDUAL' as TABLE_NAME, COUNT(*) as RECORD_COUNT FROM PJI_INDIVIDUAL;
4
```

100% 1:4

Result Grid Filter Rows: Search Export:

TABLE_NAME	RECORD_COUNT
PJI_INDIVIDUAL	10

```
1
2    -- Count Invoices
3    SELECT 'PJI_INVOICE' as TABLE_NAME, COUNT(*) as RECORD_COUNT FROM PJI_INVOICE;
4
```

100% 1:4

Result Grid Filter Rows: Search Export:

TABLE_NAME	RECORD_COUNT
PJI_INVOICE	10

```
1
2    -- Count Organization Sponsors
3    SELECT 'PJI_ORG' as TABLE_NAME, COUNT(*) as RECORD_COUNT FROM PJI_ORG;
4
```

100% 1:4

Result Grid Filter Rows: Search Export:

TABLE_NAME	RECORD_COUNT
PJI_ORG	10

```
1
2    -- Count Payments
3    SELECT 'PJI_PAYMENT' as TABLE_NAME, COUNT(*) as RECORD_COUNT FROM PJI_PAYMENT;
4
```

100% 1:4

Result Grid Filter Rows: Search Export:

TABLE_NAME	RECORD_COUNT
PJI_PAYMENT	10

```

1
2      -- Count Rentals
3      SELECT 'PJI_RENTAL' as TABLE_NAME, COUNT(*) as RECORD_COUNT FROM PJI_RENTAL;
4

```

100% 1:4

**Result Grid** Filter Rows: Search Export:

TABLE_NAME	RECORD_COUNT
PJI_RENTAL	20

```

1
2      -- Count Reservations
3      SELECT 'PJI_RESERVATION' as TABLE_NAME, COUNT(*) as RECORD_COUNT FROM PJI_RESERVATION;
4

```

100% 1:4

**Result Grid** Filter Rows: Search Export:

TABLE_NAME	RECORD_COUNT
PJI_RESERVATION	20

```

1
2      -- Count Seminar Sponsors
3      SELECT 'PJI_SEM_SPONSOR' as TABLE_NAME, COUNT(*) as RECORD_COUNT FROM PJI_SEM_SPONSOR;
4

```

100% 1:4

**Result Grid** Filter Rows: Search Export:

TABLE_NAME	RECORD_COUNT
PJI_SEM_SPONSOR	10

```

1
2      -- Count Seminars
3      SELECT 'PJI_SEMINAR' as TABLE_NAME, COUNT(*) as RECORD_COUNT FROM PJI_SEMINAR;
4

```

100% 1:4

**Result Grid** Filter Rows: Search Export:

TABLE_NAME	RECORD_COUNT
PJI_SEMINAR	10

```

1
2      -- Count Seminar Attendance
3      SELECT 'PJI_SEMINAR_ATTD' as TABLE_NAME, COUNT(*) as RECORD_COUNT FROM PJI_SEMINAR_ATTD;
4

```

100% 1:4

**Result Grid** Filter Rows: Search Export:

TABLE_NAME	RECORD_COUNT
PJI_SEMINAR_ATTD	20

```
1
2    -- Count Sponsors
3    SELECT 'PJI_SPONSOR' as TABLE_NAME, COUNT(*) as RECORD_COUNT FROM PJI_SPONSOR;
4
```

100% 1:4

Result Grid Filter Rows: Search Export:

TABLE_NAME	RECORD_COUNT
PJI_SPONSOR	20

```
1
2    -- Count Study Rooms
3    SELECT 'PJI_STUDY_ROOM' as TABLE_NAME, COUNT(*) as RECORD_COUNT FROM PJI_STUDY_ROOM;
4
```

100% 1:4

Result Grid Filter Rows: Search Export:

TABLE_NAME	RECORD_COUNT
PJI_STUDY_ROOM	20

```
1
2    -- Count Topics
3    SELECT 'PJI_TOPIC' as TABLE_NAME, COUNT(*) as RECORD_COUNT FROM PJI_TOPIC;
4
```

100% 1:4

Result Grid Filter Rows: Search Export:

TABLE_NAME	RECORD_COUNT
PJI_TOPIC	20

# Screenshots our Web Application

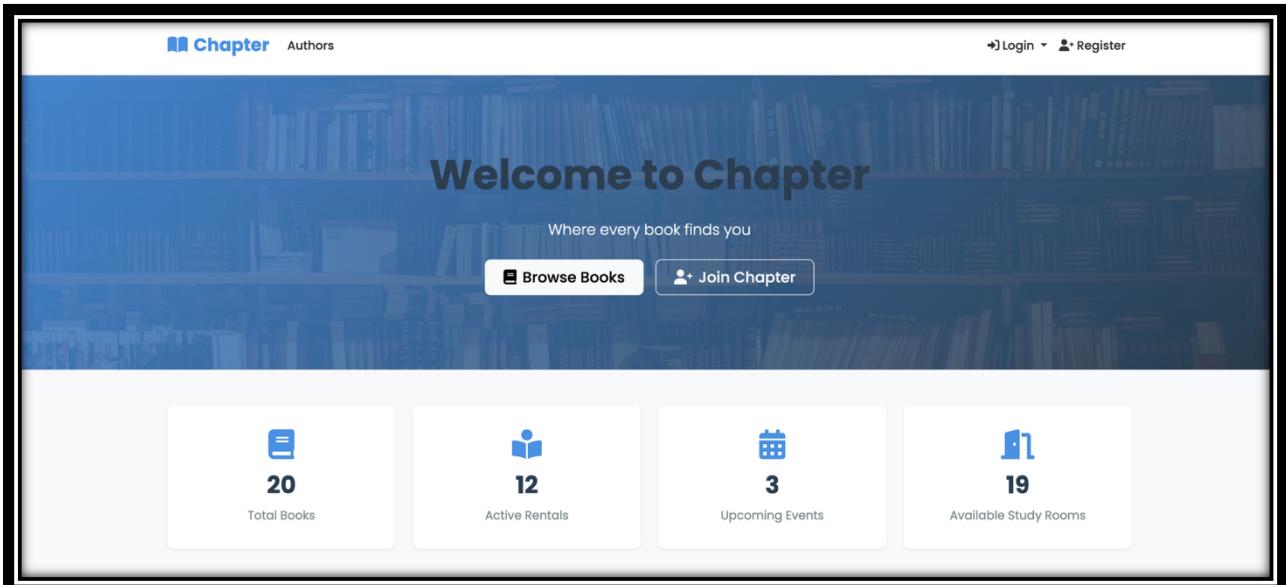


Image: Homepage

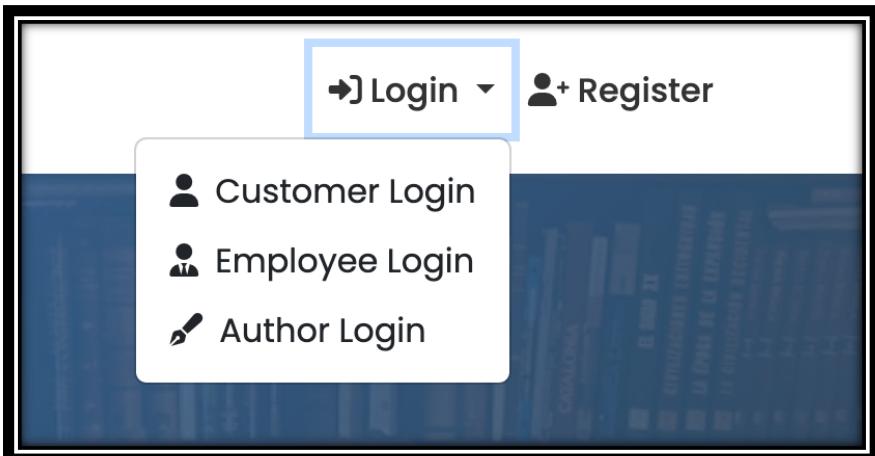
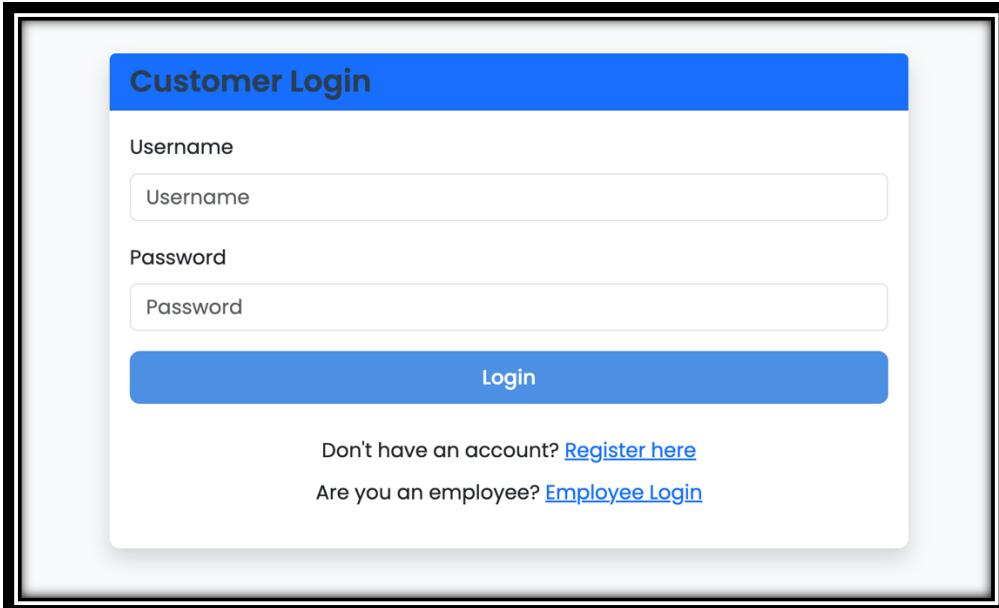


Image: Account Login Choice



**Customer Login**

Username

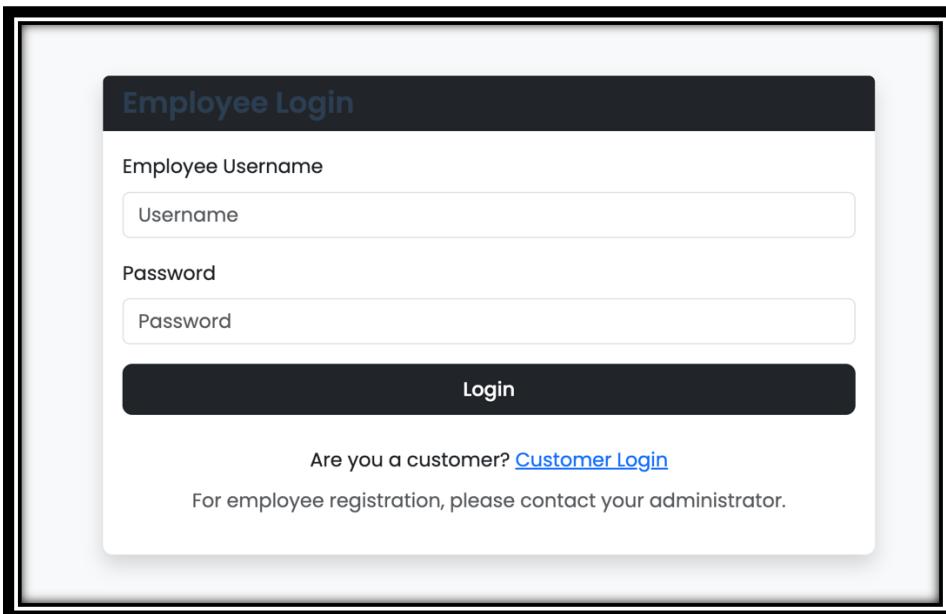
Password

**Login**

Don't have an account? [Register here](#)

Are you an employee? [Employee Login](#)

*Image: Customer Login*



**Employee Login**

Employee Username

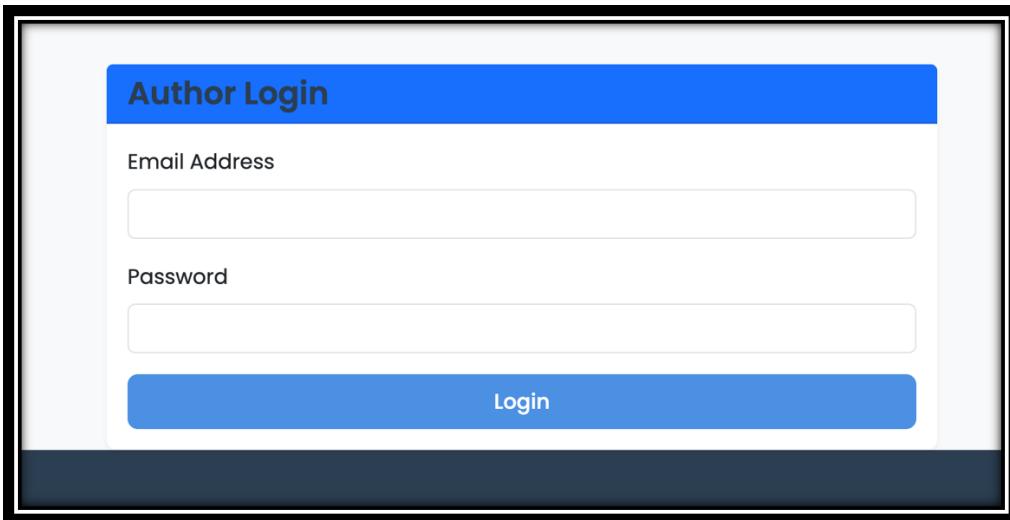
Password

**Login**

Are you a customer? [Customer Login](#)

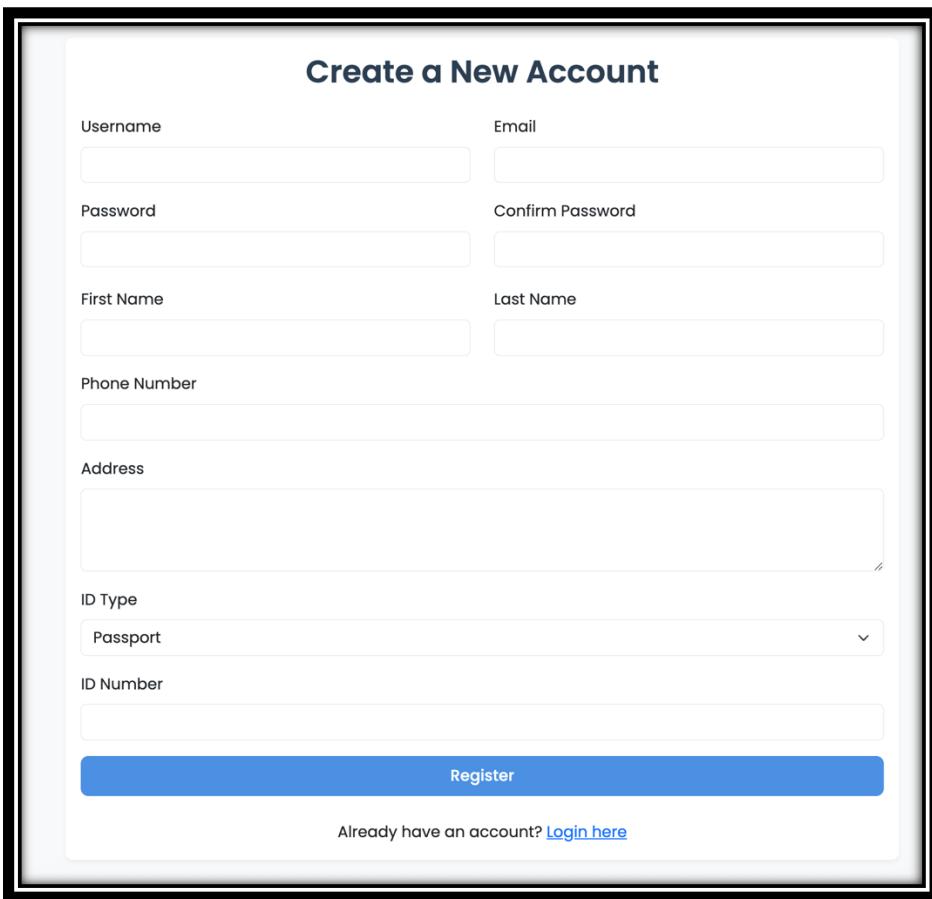
For employee registration, please contact your administrator.

*Image: Employee Login*



A screenshot of an 'Author Login' interface. At the top, a blue header bar contains the text 'Author Login'. Below it is a light gray input field labeled 'Email Address' with a corresponding text input box. Underneath is another light gray input field labeled 'Password' with a corresponding text input box. At the bottom center is a blue rectangular button labeled 'Login'.

Image: Author Login



A screenshot of an 'Account Creation Page'. The title 'Create a New Account' is centered at the top. The page contains several input fields arranged in pairs: 'Username' and 'Email', 'Password' and 'Confirm Password', 'First Name' and 'Last Name', 'Phone Number' and an empty field, 'Address' and an empty field, and 'ID Type' (with a dropdown menu showing 'Passport') and 'ID Number'. At the bottom is a blue rectangular button labeled 'Register'. Below the button, a link says 'Already have an account? [Login here](#)'.

Image: Account Creation Page

## Customer View

### Books

<b>The History of Time</b> Authors: Sarah Davis Category: Physics Available Copies: 1  <a href="#">View</a>	<b>Modern Physics</b> Authors: Michael Brown Category: Science Available Copies: 2  <a href="#">View</a>	<b>Art in Renaissance</b> Authors: Emma Johnson Category: Art History Available Copies: 2  <a href="#">View</a>
<b>Classic Literature</b> Authors: John Smith Category: Literature Available Copies: 1  <a href="#">View</a>	<b>Digital Revolution</b> Authors: David Wilson Category: Technology Available Copies: 2  <a href="#">View</a>	<b>Philosophy of Mind</b> Authors: Sarah Davis Category: Philosophy Available Copies: 2  <a href="#">View</a>
<b>Understanding Psychology</b> Authors: Michael Brown Category: Psychology Available Copies: 2  <a href="#">View</a>	<b>Business Strategy</b> Authors: Emma Johnson Category: Business Available Copies: 2  <a href="#">View</a>	<b>Advanced Mathematics</b> Authors: John Smith Category: Mathematics Available Copies: 1  <a href="#">View</a>
<b>Human Biology</b>	<b>Quantum Mechanics</b>	<b>Organic Chemistry</b>

Image: Available Books List

54

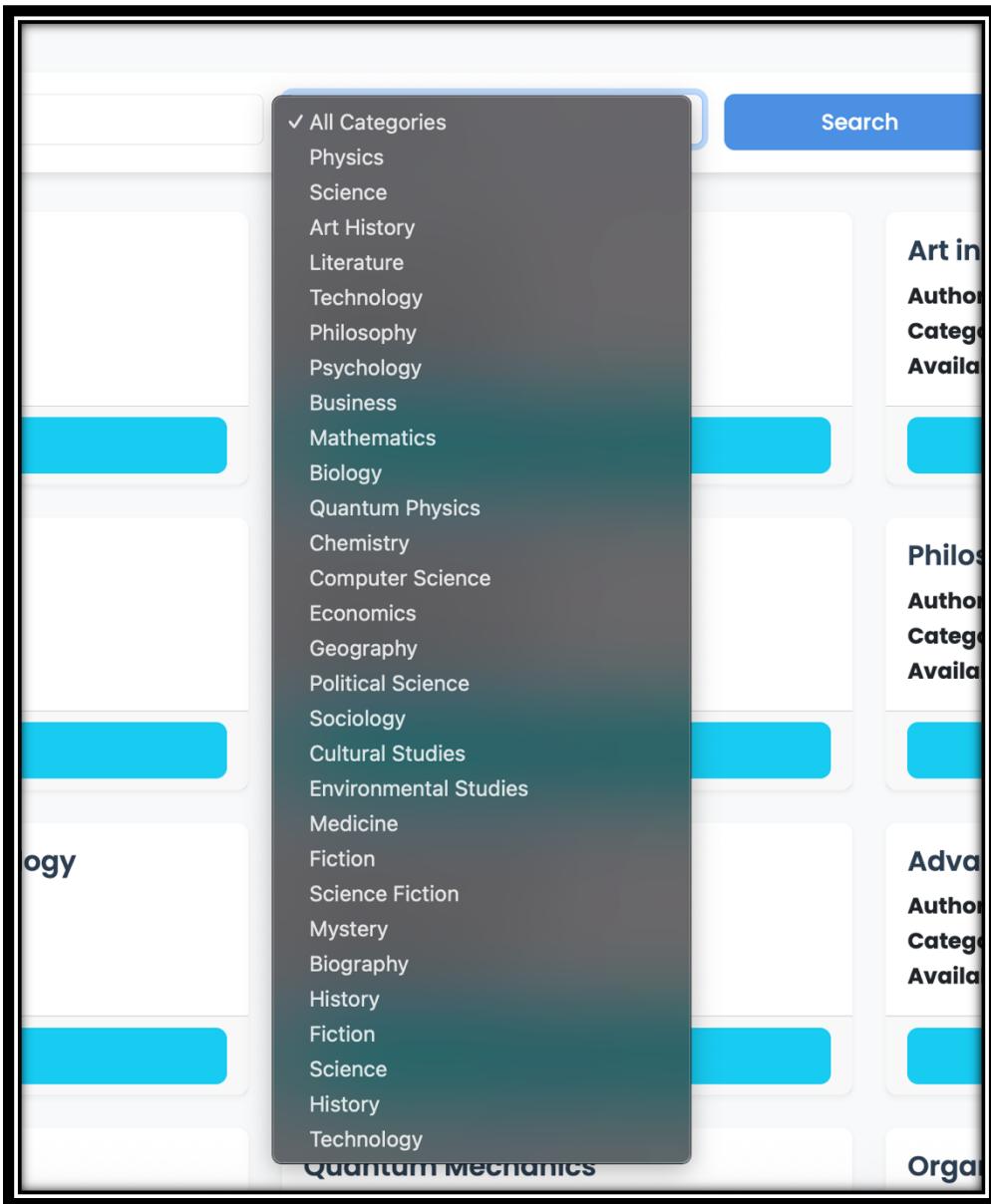


Image: Category of Books List

**Authors**

Search authors...

**John Smith**  
Email: john.smith@email.com  
Country: USA  
Location: New York, NY

**Emma Johnson**  
Email: emma.j@email.com  
Country: USA  
Location: Los Angeles, CA

**Michael Brown**  
Email: mbrown@email.com  
Country: USA  
Location: Chicago, IL

**Sarah Davis**  
Email: sarah.d@email.com  
Country: USA  
Location: Houston, TX

**David Wilson**  
Email: dwilson@email.com  
Country: USA  
Location: Phoenix, AZ

**Lisa Anderson**  
Email: lisa.a@email.com  
Country: USA  
Location: Philadelphia, PA

**James Taylor**  
Email: jtaylor@email.com  
Country: USA  
Location: San Antonio, TX

**Jennifer Thomas**  
Email: jthomas@email.com  
Country: USA  
Location: San Diego, CA

**Robert Martinez**  
Email: rmartinez@email.com  
Country: USA  
Location: Dallas, TX

[View Details](#) [View Details](#)

Image: Authors List

**Events at Chapter**

### Exhibitions

Event Name	Start Date	End Date	Expected Attendance	Action
Science Fiction Book Showcase	May. 20, 2025, 6:01 p.m.	May. 21, 2025, 6:01 p.m.	120 people	<a href="#">Register</a>
Photography Showcase	May. 19, 2025, 11:00 a.m.	May. 19, 2025, 7:00 p.m.	75 people	<a href="#">Register</a>
Children's Literature Exhibition	May. 17, 2025, 6:01 p.m.	May. 18, 2025, 6:01 p.m.	75 people	<a href="#">Register</a>
Summer Book Fair 2024	May. 14, 2025, 6:01 p.m.	May. 15, 2025, 6:01 p.m.	100 people	<a href="#">Register</a>
Art Exhibition: Modern Masters	May. 12, 2025, 10:00 a.m.	May. 12, 2025, 6:00 p.m.	50 people	Event Ended
Architecture Models Display	Aug. 15, 2024, 9:00 a.m.	Aug. 25, 2024, 5:00 p.m.	160 people	Event Ended
Digital Art Show	Aug. 01, 2024, 10:00 a.m.	Aug. 10, 2024, 6:00 p.m.	140 people	Event Ended
Cultural Heritage Exhibition	Jul. 15, 2024, 9:00 a.m.	Jul. 25, 2024, 6:00 p.m.	130 people	Event Ended
Environmental Awareness Display	Jul. 01, 2024, 10:00 a.m.	Jul. 10, 2024, 6:00 p.m.	110 people	Event Ended
Book Art Exhibition	Jun. 15, 2024, 9:00 a.m.	Jun. 25, 2024, 5:00 p.m.	90 people	Event Ended
Technology Showcase	Jun. 01, 2024, 10:00 a.m.	Jun. 10, 2024, 6:00 p.m.	200 people	Event Ended
Photography Exhibition	May. 15, 2024, 9:00 a.m.	May. 25, 2024, 5:00 p.m.	120 people	Event Ended
Historical Documents Display	May. 01, 2024, 10:00 a.m.	May. 10, 2024, 6:00 p.m.	80 people	Event Ended
Science Fair 2024	Apr. 15, 2024, 9:00 a.m.	Apr. 20, 2024, 5:00 p.m.	150 people	Event Ended

Image: Events List (Exhibitions)

## Seminars

Event Name	Start Date	End Date	Expected Attendance	Action
Digital Publishing Trends	May. 19, 2025, 6:01 p.m.	May. 19, 2025, 9:01 p.m.	40 people	Author Registration Only
Writing Workshop with Best-Selling Authors	May. 15, 2025, 6:01 p.m.	May. 15, 2025, 10:01 p.m.	50 people	Author Registration Only
Author Workshop: Writing for Digital Age	May. 13, 2025, 10:00 a.m.	May. 13, 2025, 12:00 p.m.	50 people	Author Registration Only
Innovation Workshop	Aug. 20, 2024, 1:00 p.m.	Aug. 20, 2024, 4:00 p.m.	40 people	Event Ended
Leadership Skills	Aug. 05, 2024, 2:00 p.m.	Aug. 05, 2024, 5:00 p.m.	35 people	Event Ended
Business Strategy	Jul. 20, 2024, 1:00 p.m.	Jul. 20, 2024, 4:00 p.m.	45 people	Event Ended
Creative Writing	Jul. 05, 2024, 2:00 p.m.	Jul. 05, 2024, 5:00 p.m.	25 people	Event Ended
Project Management	Jun. 20, 2024, 1:00 p.m.	Jun. 20, 2024, 4:00 p.m.	40 people	Event Ended
Public Speaking	Jun. 05, 2024, 2:00 p.m.	Jun. 05, 2024, 5:00 p.m.	30 people	Event Ended
Data Analysis	May. 20, 2024, 1:00 p.m.	May. 20, 2024, 4:00 p.m.	35 people	Event Ended
Digital Marketing	May. 05, 2024, 2:00 p.m.	May. 05, 2024, 5:00 p.m.	45 people	Event Ended
Research Methodology	Apr. 20, 2024, 1:00 p.m.	Apr. 20, 2024, 4:00 p.m.	40 people	Event Ended
Writing Workshop	Apr. 05, 2024, 2:00 p.m.	Apr. 05, 2024, 5:00 p.m.	50 people	Event Ended

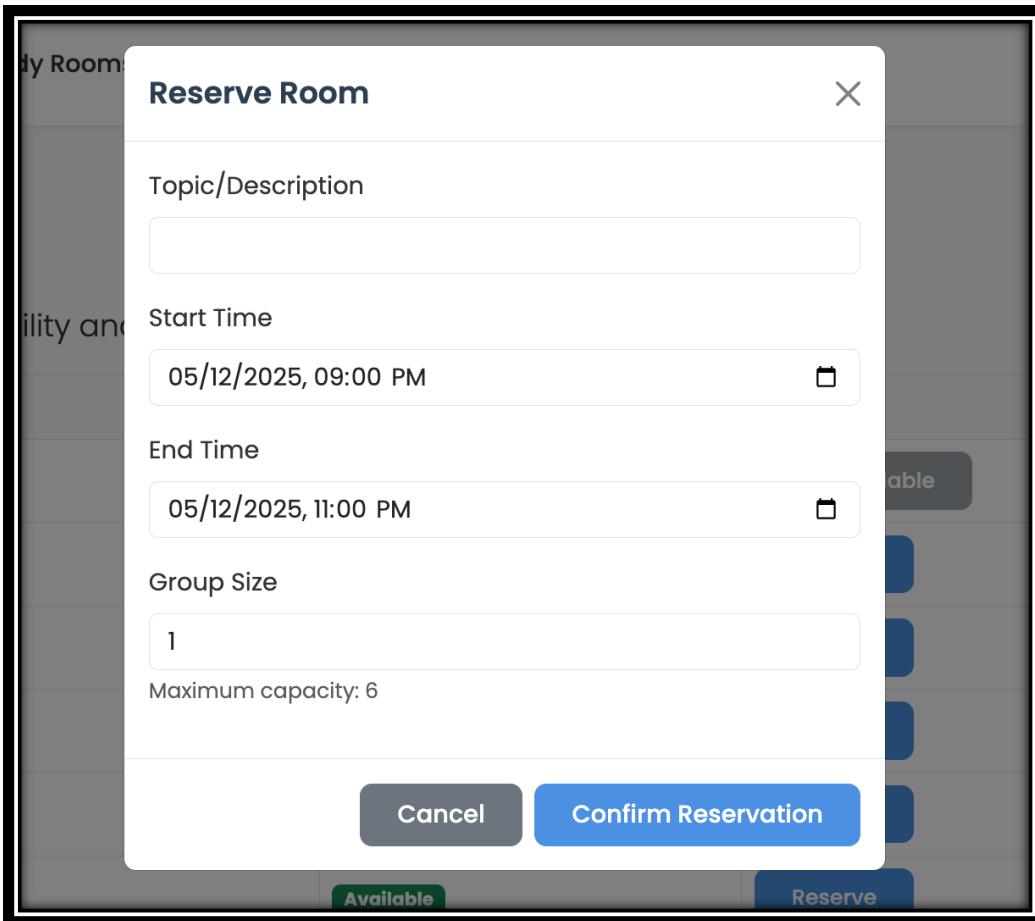
Image: Events List (Seminars)

## Study Rooms

Reserve a study room. See availability and details below.

Room ID	Capacity	Status	Action
1	4	Occupied	Not Available
2	6	Available	Reserve
3	8	Available	Reserve
4	4	Available	Reserve
5	6	Available	Reserve
6	8	Available	Reserve
7	4	Available	Reserve
8	6	Available	Reserve
9	8	Available	Reserve
10	4	Available	Reserve
11	6	Available	Reserve
12	8	Available	Reserve

Image: Study Rooms List



A screenshot of a room reservation form titled "Reserve Room". The form includes fields for Topic/Description, Start Time (05/12/2025, 09:00 PM), End Time (05/12/2025, 11:00 PM), Group Size (1), and Maximum capacity: 6. There are "Cancel" and "Confirm Reservation" buttons at the bottom.

Available Reserve

Topic/Description

Start Time  
05/12/2025, 09:00 PM

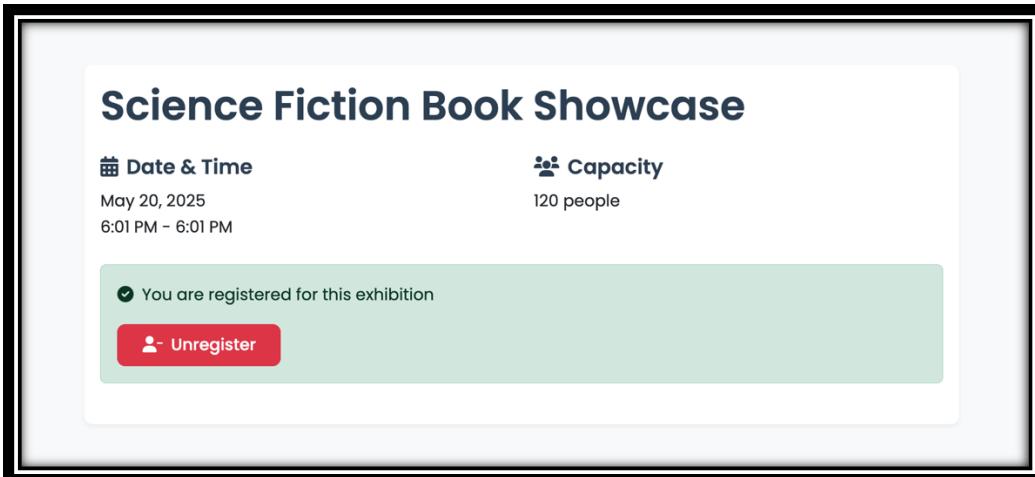
End Time  
05/12/2025, 11:00 PM

Group Size  
1

Maximum capacity: 6

Cancel Confirm Reservation

Image: Room Reservation Form



A screenshot of an event registration confirmation page for "Science Fiction Book Showcase". It shows Date & Time (May 20, 2025, 6:01 PM - 6:01 PM) and Capacity (120 people). A message indicates "You are registered for this exhibition" with an "Unregister" button.

Date & Time  
May 20, 2025  
6:01 PM - 6:01 PM

Capacity  
120 people

You are registered for this exhibition

Unregister

Image: Event Registration Confirmation Page

My Rentals				
Book	Borrow Date	Expected Return	Status	Actions
Classic Literature	May 12, 2025	May 26, 2025	Borrowed	<a href="#">Details</a>
Advanced Mathematics	May 12, 2025	May 26, 2025	Borrowed	<a href="#">Details</a>
Art in Renaissance	May 12, 2025	May 26, 2025	Returned	<a href="#">Details</a>
Modern Physics	May 12, 2025	May 26, 2025	Returned	<a href="#">Details</a>
Modern Physics	May 12, 2025	May 26, 2025	Returned	<a href="#">Details</a>
The History of Time	May 12, 2025	May 26, 2025	Borrowed	<a href="#">Details</a>
The History of Time	May 12, 2025	May 26, 2025	Returned	<a href="#">Details</a>
The History of Time	May 12, 2025	May 26, 2025	Returned	<a href="#">Details</a>
The History of Time	May 10, 2025	May 17, 2025	Borrowed	<a href="#">Details</a>

Image: List of Books which the Customer has Rented out

### Rental Details

Book	The History of Time
Copy ID	#1
Borrow Date	May 12, 2025
Expected Return	May 26, 2025
Status	Returned
Actual Return	May 12, 2025

### Invoice Information

Invoice #	3
Amount	\$5.00
Status	Paid

[← Back to Rentals](#)

Image: Invoice

**My Reservations**

Upcoming Reservations Past Reservations

Room	Topic	Start Time	End Time	Group Size	Actions
Room #1	Upcoming Study Session	May 12, 2025 - 7:31 p.m.	May 12, 2025 - 8:31 p.m.	3 people	<button>Cancel</button>

[Reserve a Room](#)

Image: Reservations List (Upcoming)

**My Reservations**

Upcoming Reservations Past Reservations

Room	Topic	Start Time	End Time	Group Size
Room #1	Past Study Session	May 12, 2025 - 4:31 p.m.	May 12, 2025 - 5:31 p.m.	2 people

[Reserve a Room](#)

Image: Reservations List (Past)

**My Profile**

**AliceJ**  
 alice.j@email.com  
 Member since: May 12, 2025

---

<b>Account Information</b>	<b>Customer Information</b>		
Username	AliceJ	Name	Alice Johnson
Email	alice.j@email.com	Phone	555-0101
Last Login	May 12, 2025, 7:54 p.m.	Address	None

**Quick Links**

[My Rentals](#)
[My Reservations](#)
[My Invoices](#)

Image: Profile Page

## Employee View

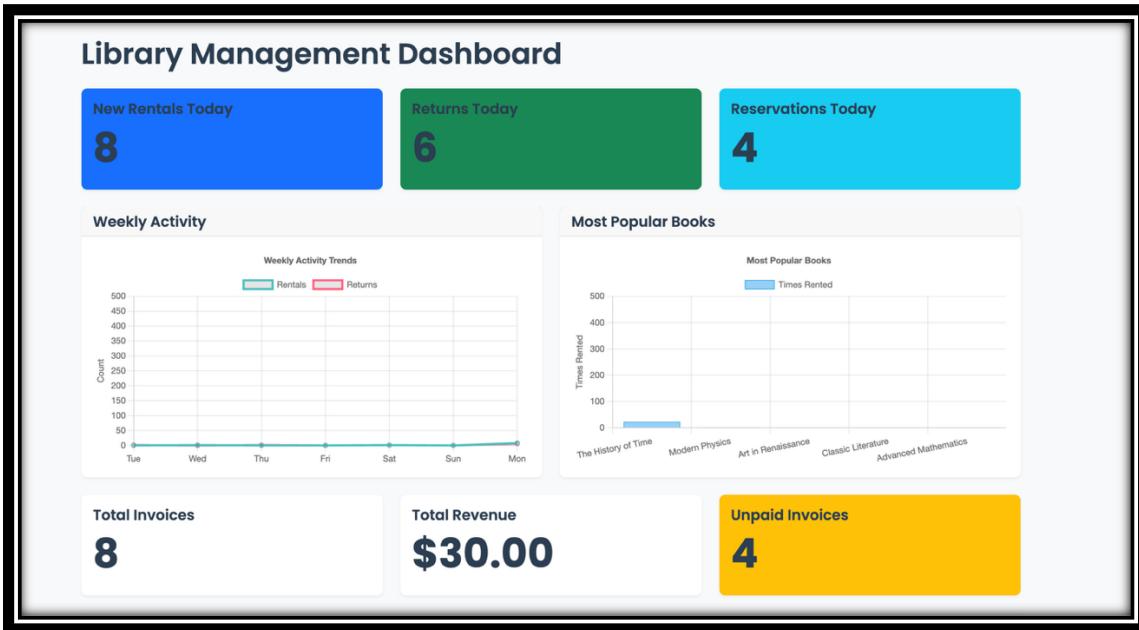


Image: Employee View Dashboard

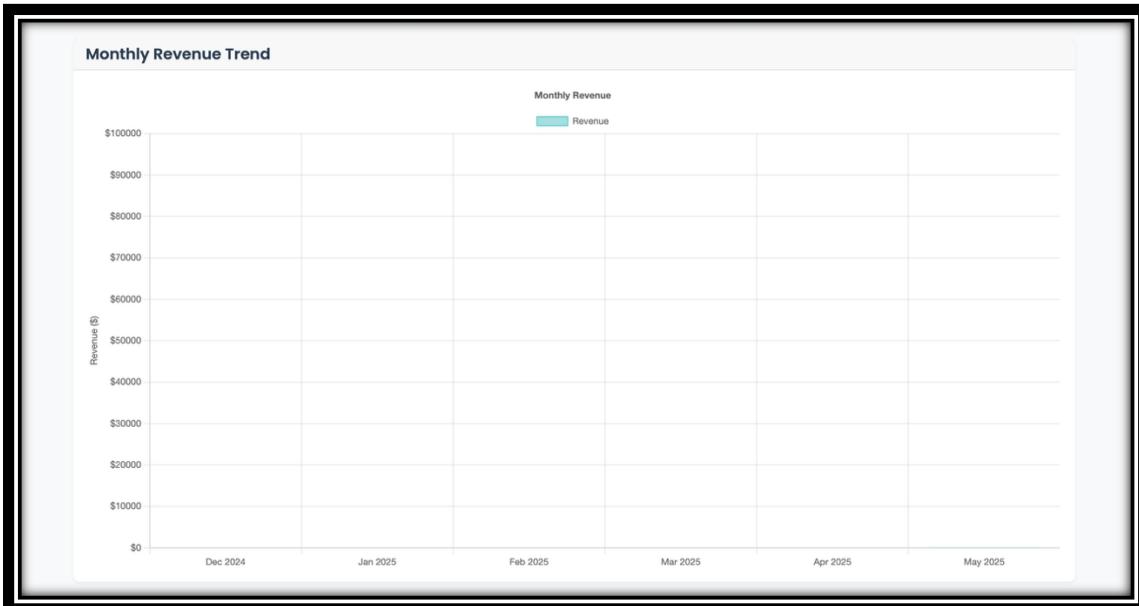


Image: Graph of Monthly Revenue

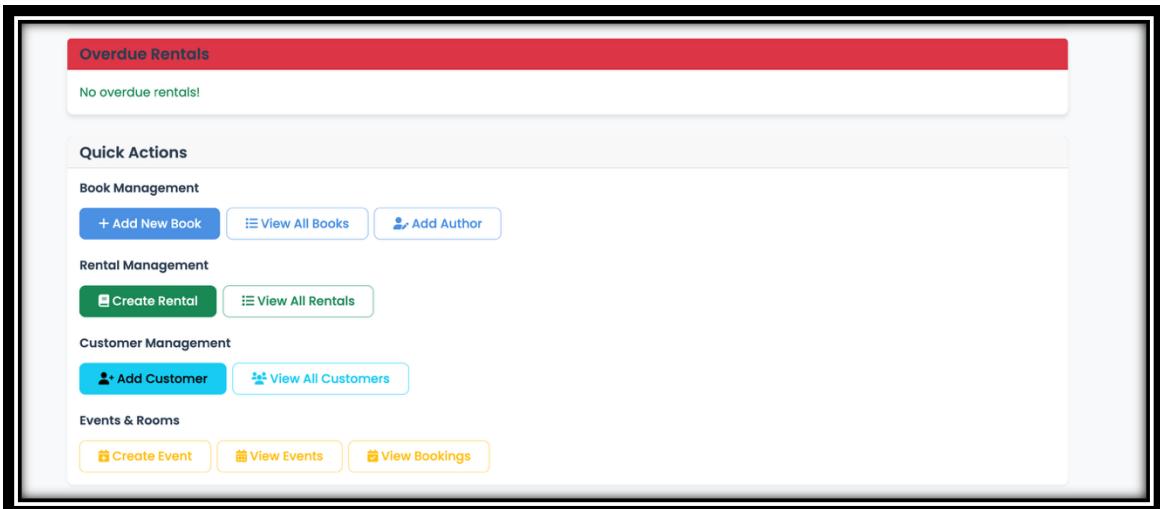


Image: List of Overdue Rentals and Some Quick Actions

The screenshot shows a "Create Book" form. The title "Create Book" is at the top. The form fields include: "Book name" (input field), "Topic" (input field with a dropdown arrow), "Authors" (list box containing "John Smith", "Emma Johnson", "Michael Brown", "Sarah Davis", and "David Wilson"), "Initial copies" (input field with value "1"), and "Number of copies to create (only for new books)" (input field). At the bottom are two buttons: a blue "Save Book" button with a save icon and a grey "Cancel" button with a cancel icon.

Image: Create Book Form

Books		
<input type="text" value="Search books..."/> <input type="button" value="All Categories"/> <input type="button" value="Search"/> <input type="button" value="Clear"/> <input type="button" value="+ Add New Book"/>		
<b>The History of Time</b> Authors: Sarah Davis Category: Physics Available Copies: 1	<b>Modern Physics</b> Authors: Michael Brown Category: Science Available Copies: 2	<b>Art in Renaissance</b> Authors: Emma Johnson Category: Art History Available Copies: 2
<input type="button" value="View"/> <input type="button" value="Edit"/> <input type="button" value="+ Add Copies"/>	<input type="button" value="View"/> <input type="button" value="Edit"/> <input type="button" value="+ Add Copies"/>	<input type="button" value="View"/> <input type="button" value="Edit"/> <input type="button" value="+ Add Copies"/>
<b>Classic Literature</b> Authors: John Smith Category: Literature Available Copies: 1	<b>Digital Revolution</b> Authors: David Wilson Category: Technology Available Copies: 2	<b>Philosophy of Mind</b> Authors: Sarah Davis Category: Philosophy Available Copies: 2
<input type="button" value="View"/> <input type="button" value="Edit"/> <input type="button" value="+ Add Copies"/>	<input type="button" value="View"/> <input type="button" value="Edit"/> <input type="button" value="+ Add Copies"/>	<input type="button" value="View"/> <input type="button" value="Edit"/> <input type="button" value="+ Add Copies"/>
<b>Understanding Psychology</b> Authors: Michael Brown Category: Psychology Available Copies: 2	<b>Business Strategy</b> Authors: Emma Johnson Category: Business Available Copies: 2	<b>Advanced Mathematics</b> Authors: John Smith Category: Mathematics Available Copies: 1
<input type="button" value="View"/> <input type="button" value="Edit"/> <input type="button" value="+ Add Copies"/>	<input type="button" value="View"/> <input type="button" value="Edit"/> <input type="button" value="+ Add Copies"/>	<input type="button" value="View"/> <input type="button" value="Edit"/> <input type="button" value="+ Add Copies"/>

Image: Total Books Available with Copies Count and Authors

Author Information		Books by John Smith			
<b>John Smith</b>					
<b>Email:</b> john.smith@email.com <b>Address:</b> 123 Oak St New York, NY 10001 USA		<b>Title</b> <b>Topic</b> <b>Available Copies</b> <b>Actions</b>			
		Classic Literature	Literature	1	<input type="button" value="View"/>
		Advanced Mathematics	Mathematics	1	<input type="button" value="View"/>
		Microeconomics	Economics	2	<input type="button" value="View"/>
		Environmental Science	Environmental Studies	2	<input type="button" value="View"/>
Seminar Participation					
This author has not participated in any seminars.					

Image: Author Information

## Security Features Implemented

- We have implemented user authentication and role-based access control to ensure that only authorized users can access sensitive features and data.
- We have used secure password hashing and storage mechanisms to protect user credentials.
- We have enabled CSRF (Cross-Site Request Forgery) protection on all forms to prevent unauthorized actions from malicious sites.
- We have applied input validation and sanitization throughout the application to prevent SQL injection and cross-site scripting (XSS) attacks.
- We have enforced HTTPS (when deployed in production) to encrypt data transmitted between the client and server.
- We have restricted access to administrative and sensitive endpoints to only users with appropriate permissions.
- We have used Django's built-in security middleware to provide additional layers of protection, such as clickjacking prevention and secure session management.
- We have ensured that user-uploaded files are handled securely to prevent the execution of malicious code.

## Lesson Learned: Reflections on Project Work

Working on this project has been a valuable learning experience for us, both technically and personally. We have gained a deeper understanding of full-stack web application development, particularly using Django and relational databases. We learned how to design and implement a scalable, modular system that addresses real-world business needs, and how to integrate features such as authentication, role-based access, and analytics.

What went well was our ability to break down the project into manageable components and assign clear responsibilities within the team. The use of version control (Git) and regular communication helped us stay organized and track progress. We were especially pleased with how the user interface turned out modern, intuitive, and responsive which we believe will enhance the user experience.

However, not everything went smoothly. One of the main challenges we faced was time management, especially when balancing this project with other academic and personal commitments. Coordinating with team members remotely also posed some difficulties, such as scheduling meetings across different time zones and ensuring everyone was on the same page regarding requirements and deadlines. There were moments when technical issues, like database migrations or deployment bugs, took longer to resolve than anticipated.

Despite these constraints, we learned the importance of adaptability, clear communication, and proactive problem-solving. We also realized the value of thorough documentation and testing, which helped us

catch issues early and made collaboration easier. Overall, this project has strengthened our technical skills and our ability to work effectively as part of a team, and we are proud of what we accomplished together.

## Business analysis with 6 SQLs using our project data

### Q1: Table joins with at least 3 tables in join

Select query:

```
SELECT
    r.RENTAL_ID,
    CONCAT(c.FNAME, ' ', c.LNAME) AS Customer_Name,
    b.BOOK_NAME,
    CONCAT(a.FNAME, ' ', a.LNAME) AS Author_Name,
    t.TOPIC_NAME,
    r.BORROW_DATE,
    r.EXP_RETURN_DT AS Expected_Return_Date,
    r.STATUS
FROM
    PJI_RENTAL r
JOIN
    PJI_CUSTOMER c ON r.PJI_CUSTOMER_CUST_ID = c.CUST_ID
JOIN
    PJI_BOOK_COPY bc ON r.PJI_BOOK_COPY_COPY_ID = bc.COPY_ID
JOIN
    PJI_BOOK b ON bc.PJI_BOOK_BOOK_ID = b.BOOK_ID
JOIN
    PJI_BOOK_AUTHOR ba ON b.BOOK_ID = ba.PJI_BOOK_BOOK_ID
JOIN
    PJI_AUTHOR a ON ba.PJI_AUTHOR_AUTHOR_ID = a.AUTHOR_ID
JOIN
    PJI_TOPIC t ON b.PJI_TOPIC_TOPIC_ID = t.TOPIC_ID
ORDER BY
    r.BORROW_DATE DESC;
```

Result of the query:

RENTAL_ID	Customer_Name	BOOK_NAME	Author_Name	TOPIC_NAME	BORROW_DATE	Expected_Return_Da...	STATUS
10	Kelly Thomas	The History of Time	Sarah Davis	Physics	2025-05-08 22:46:37	2025-05-12 22:46:37	Returned
14	Grace Wilson	The History of Time	Sarah Davis	Physics	2025-05-07 22:46:37	2025-05-20 22:46:37	Borrowed
7	Noah Harris	The History of Time	Sarah Davis	Physics	2025-05-06 22:46:37	2025-05-17 22:46:37	Returned
12	Ivy Taylor	The History of Time	Sarah Davis	Physics	2025-05-05 22:46:37	2025-05-11 22:46:37	Borrowed
13	Henry Moore	The History of Time	Sarah Davis	Physics	2025-05-04 22:46:37	2025-05-17 22:46:37	Borrowed
5	Peter Thompson	The History of Time	Sarah Davis	Physics	2025-05-01 22:46:37	2025-05-14 22:46:37	Returned
16	Eve Davis	The History of Time	Sarah Davis	Physics	2025-04-30 22:46:37	2025-05-13 22:46:37	Borrowed
8	Mia White	The History of Time	Sarah Davis	Physics	2025-04-29 22:46:37	2025-05-23 22:46:37	Returned
2	Sofia Robinson	The History of Time	Sarah Davis	Physics	2025-04-28 22:46:37	2025-05-17 22:46:37	Returned
19	Bob Smith	The History of Time	Sarah Davis	Physics	2025-04-25 22:46:37	2025-05-21 22:46:37	Borrowed
17	David Brown	The History of Time	Sarah Davis	Physics	2025-04-24 22:46:37	2025-05-18 22:46:37	Borrowed
3	Ryan Martinez	The History of Time	Sarah Davis	Physics	2025-04-24 22:46:37	2025-05-22 22:46:37	Returned
18	Carol Williams	The History of Time	Sarah Davis	Physics	2025-04-23 22:46:37	2025-05-21 22:46:37	Borrowed
15	Frank Miller	The History of Time	Sarah Davis	Physics	2025-04-23 22:46:37	2025-05-15 22:46:37	Borrowed
1	Tyler Clark	The History of Time	Sarah Davis	Physics	2025-04-21 22:46:37	2025-05-15 22:46:37	Returned
11	Jack Anderson	The History of Time	Sarah Davis	Physics	2025-04-19 22:46:37	2025-05-21 22:46:37	Borrowed
20	Alice Johnson	The History of Time	Sarah Davis	Physics	2025-04-18 22:46:37	2025-05-11 22:46:37	Borrowed
6	Olivia Martin	The History of Time	Sarah Davis	Physics	2025-04-17 22:46:37	2025-05-19 22:46:37	Returned
4	Quinn Garcia	The History of Time	Sarah Davis	Physics	2025-04-11 22:46:37	2025-05-10 22:46:37	Returned
9	Liam Jackson	The History of Time	Sarah Davis	Physics	2025-04-09 22:46:37	2025-05-23 22:46:37	Returned

Business Purpose:

This query provides a comprehensive view of book rentals, showing which customers have borrowed which books, who wrote them, what topics they cover, when they were borrowed, and their current status. This helps library staff track borrowing patterns, understand customer preferences, and monitor the status of rentals.

## Q2: Multi-row subquery

Select query:

```

SELECT
    c.CUST_ID,
    CONCAT(c.FNAME, ' ', c.LNAME) AS Customer_Name,
    e.EVENT_NAME,
    ex.EXPENSES
FROM
    PJI_CUSTOMER c
JOIN
    PJI_EXH_ATTD ea ON c.CUST_ID = ea.PJI_CUSTOMER_CUST_ID
JOIN
    PJI_EXHIBITION ex ON ea.PJI_EXHIBITION_EVENT_ID = ex.EVENT_ID
JOIN
    PJI_EVENT e ON ex.EVENT_ID = e.EVENT_ID
WHERE
    ex.EXPENSES > (
        SELECT AVG(EXPENSES)
    )

```

```

        FROM PJI_EXHIBITION
    )
ORDER BY
    ex.EXPENSES DESC;

```

Result of the query:

CUST_ID	Customer_Name	EVENT_NAME	EXPENSES
1	Alice Johnson	Science Fair 2024	7500.00
19	Sofia Robinson	Science Fair 2024	7500.00
4	David Brown	Science Fair 2024	7500.00
10	Jack Anderson	Science Fair 2024	7500.00
16	Peter Thompson	Science Fair 2024	7500.00
7	Grace Wilson	Science Fair 2024	7500.00
13	Mia White	Science Fair 2024	7500.00

Business Purpose:

This query identifies customers who have attended premium exhibitions (those with expenses above the average). This information can be used for targeted marketing of future premium events, analyzing customer preferences for higher-budget events, and evaluating the return on investment for more expensive exhibitions.

### Q3: Correlated subquery

Select query:

```

SELECT
    a.AUTHOR_ID,
    CONCAT(a.FNAME, ' ', a.LNAME) AS Author_Name,
    a.EMAIL,
    a.CITY,
    a.COUNTRY
FROM
    PJI_AUTHOR a
WHERE
    NOT EXISTS (
        SELECT 1

```

```

        FROM PJI_SEMINAR_ATTD sa
        WHERE sa.PJI_AUTHOR_AUTHOR_ID = a.AUTHOR_ID
    )
ORDER BY
    a.LNAME, a.FNAME;

```

Result of the query:

AUTHOR_ID	Author_Name	EMAIL	CITY	COUNTRY
21	Chetan Bhagat	chetanbhagat@hotmail.com	Mumbai	India

Business Purpose:

This query identifies authors who have not yet participated in any seminars. This information helps the library's outreach team target these authors for future seminar invitations, potentially increasing author engagement and diversifying seminar participants. It can also help analyze if there are geographic patterns to non-participation.

#### **Q4: SET operator query**

Select query:

```

SELECT
    a.FNAME,
    a.LNAME,
    'Author and Customer' AS Role
FROM
    PJI_AUTHOR a
JOIN
    PJI_CUSTOMER c ON a.FNAME = c.FNAME AND a.LNAME = c.LNAME
UNION
SELECT
    a.FNAME,
    a.LNAME,
    'Author Only' AS Role
FROM

```

```
PJI_AUTHOR a
WHERE
NOT EXISTS (
    SELECT 1
    FROM PJI_CUSTOMER c
    WHERE a.FNAME = c.FNAME AND a.LNAME = c.LNAME
)
UNION

SELECT
    c.FNAME,
    c.LNAME,
    'Customer Only' AS Role
FROM
    PJI_CUSTOMER c
WHERE
NOT EXISTS (
    SELECT 1
    FROM PJI_AUTHOR a
    WHERE c.FNAME = a.FNAME AND c.LNAME = a.LNAME
)
ORDER BY
    LNAME, FNAME;
```

Result of the query:

FNAME	LNAME	Role
Kevin	Allen	Author Only
Jack	Anderson	Customer Only
Lisa	Anderson	Author Only
Chetan	Bhagat	Author Only
David	Brown	Customer Only
Michael	Brown	Author Only
Michelle	Clark	Author Only
Tyler	Clark	Customer Only
Eve	Davis	Customer Only
Sarah	Davis	Author Only
Patricia	Garcia	Author Only
Quinn	Garcia	Customer Only
Margaret	Hall	Author Only
Noah	Harris	Customer Only
Liam	Jackson	Customer Only
Alice	Johnson	Customer Only
Emma	Johnson	Author Only
Steven	King	Author Only
Daniel	Lee	Author Only
Elizabeth	Lewis	Author Only
Olivia	Martin	Customer Only
Robert	Martinez	Author Only
Ryan	Martinez	Customer Only
Frank	Miller	Customer Only
Henry	Moore	Customer Only
Sofia	Robinson	Customer Only
Christo...	Rodriguez	Author Only
Bob	Smith	Customer Only
John	Smith	Author Only
Ivy	Taylor	Customer Only
James	Taylor	Author Only
Jennifer	Thomas	Author Only
Kelly	Thomas	Customer Only
Peter	Thompson	Customer Only
Joseph	Walker	Author Only
Mia	White	Customer Only
Carol	Williams	Customer Only
David	Wilson	Author Only
Grace	Wilson	Customer Only
Barbara	Wright	Author Only
Nancy	Young	Author Only

Business Purpose:

This query creates a comprehensive categorization of all individuals in the library system. Identifying who serves dual roles as both authors and customers can help with community engagement strategies, personalized marketing, and creating more meaningful connections within the library community. It also helps identify potential authors who are already customers for future book events.

#### **Q5: Query with in-line view or WITH clause**

Select query:

```

WITH SeminarSponsorship AS (
    SELECT
        ss.PJI_SEMINAR_EVENT_ID,
        SUM(ss.AMOUNT) AS Total_Sponsorship
    FROM
        PJI_SEM_SPONSOR ss
    GROUP BY
        ss.PJI_SEMINAR_EVENT_ID
)
SELECT
    e.EVENT_ID,
    e.EVENT_NAME,
    e.START_DT,
    e.END_DT,
    e.ATTND_NO AS Expected_Attendees,
    s.EST_AUTH AS Estimated_Authors,
    COALESCE(ss.Total_Sponsorship, 0) AS Total_Sponsorship
FROM
    PJI_EVENT e
JOIN
    PJI_SEMINAR s ON e.EVENT_ID = s.EVENT_ID
LEFT JOIN
    SeminarSponsorship ss ON s.EVENT_ID = ss.PJI_SEMINAR_EVENT_ID
ORDER BY
    e.START_DT;

```

Result of the query:

	EVENT_ID	EVENT_NAME	START_DT	END_DT	Expected_Attende...	Estimated_Auth...	Total_Sponsors...	
	11	Writing Workshop	2024-04-05 14:00:00	2024-04-05 17:00:00	50	3	1000.00	
	12	Research Methodology	2024-04-20 13:00:00	2024-04-20 16:00:00	40	2	1500.00	
	13	Digital Marketing	2024-05-05 14:00:00	2024-05-05 17:00:00	45	2	1200.00	
	14	Data Analysis	2024-05-20 13:00:00	2024-05-20 16:00:00	35	3	1800.00	
	15	Public Speaking	2024-06-05 14:00:00	2024-06-05 17:00:00	30	1	800.00	
	16	Project Management	2024-06-20 13:00:00	2024-06-20 16:00:00	40	2	2000.00	
	17	Creative Writing	2024-07-05 14:00:00	2024-07-05 17:00:00	25	2	900.00	
	18	Business Strategy	2024-07-20 13:00:00	2024-07-20 16:00:00	45	3	1600.00	
	19	Leadership Skills	2024-08-05 14:00:00	2024-08-05 17:00:00	35	2	1100.00	
	20	Innovation Workshop	2024-08-20 13:00:00	2024-08-20 16:00:00	40	2	1700.00	

Business Purpose:

This query provides a financial overview of seminar events, showing the total sponsorship amount for each seminar along with event details. This information is essential for budget planning, evaluating the financial success of different seminar types, and identifying seminars that may need additional

sponsorship. It also helps with planning future seminars by analyzing the relationship between author participation, attendee numbers, and sponsorship amounts.

### Q6: TOP-N/BOTTOM-N query

Select query:

```
SELECT
    b.BOOK_ID,
    b.BOOK_NAME,
    t.TOPIC_NAME,
    COUNT(r.RENTAL_ID) AS Rental_Count
FROM
    PJI_BOOK b
JOIN
    PJI_TOPIC t ON b.PJI_TOPIC_TOPIC_ID = t.TOPIC_ID
JOIN
    PJI_BOOK_COPY bc ON b.BOOK_ID = bc.PJI_BOOK_BOOK_ID
LEFT JOIN
    PJI_RENTAL r ON bc.COPY_ID = r.PJI_BOOK_COPY_COPY_ID
GROUP BY
    b.BOOK_ID, b.BOOK_NAME, t.TOPIC_NAME
ORDER BY
    Rental_Count DESC
LIMIT 5;
```

Result of the query:

BOOK_ID	BOOK_NAME	TOPIC_NAME	Rental_Count
1	The History of Time	Physics	20
2	Modern Physics	Science	0
3	Art in Renaissance	Art History	0
4	Classic Literature	Literature	0
5	Digital Revolution	Technology	0

\*\*Our Records are empty that's why zero is shown in Rental\_Count

Business Purpose:

This query identifies the most popular books in the library based on rental frequency. This information is valuable for inventory management, helping librarians make informed decisions about which books to acquire more copies of, which topics are trending among patrons, and for understanding reading preferences in the community. It can also guide future book acquisitions and promotional activities.