

ASSIGNMENT-01

Name : K. Gayathri

Reg No : 192311448

Course code : CSA0593

course Name : Database Management
system for transaction
Management

Date of submission : 14/11/24.

Academic Library system with digital Resources
create a database for a library that manages both
physical books and digital resources.

Requirements:

Design tables for books, e-books, journals, authors, and
user checkout history.

Write stored procedures to handle checkouts, returns
and overdue fines for physical and digital items. Give
write queries to track popular titles, user borrowing
history, and utilization rates of digital vs physical
resources. Add triggers to update resource availability
in real-time and notify users when reserved items
are available.

Here three types of ERD Models are there.

1. Conceptual ERD Model
2. Logical ERD Model
3. Physical ERD Model.

Conceptual ERD Model:

- correctly identified entities and relationships
- consider adding an entity for "category" to
further normalize the data.

Logical ERD Model:

well defined attributes for each entity

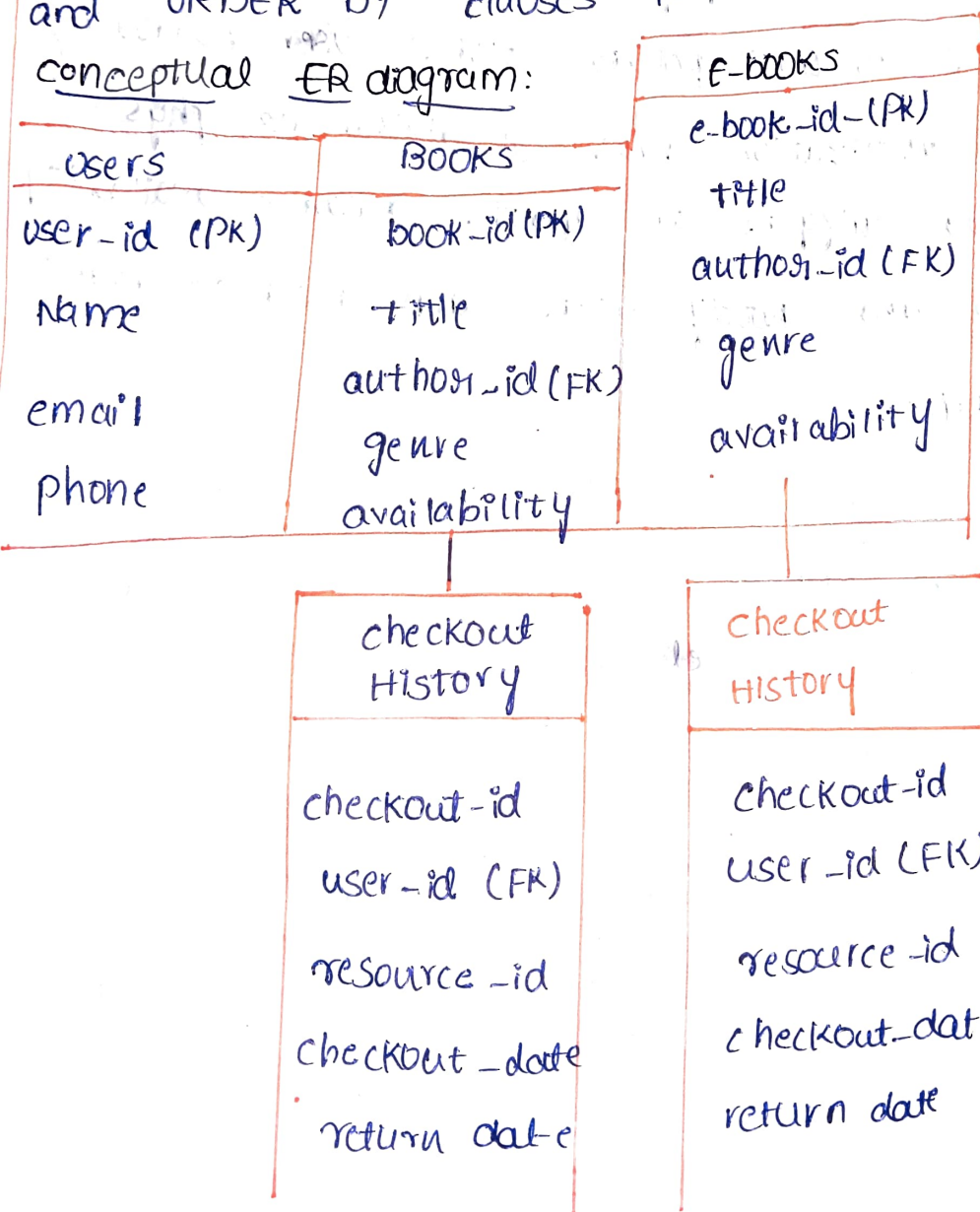
consider adding a "status" attribute to the Enrollment entity (e.g., "enrolled", "completed", "inprogress").

Physical ERD Model:

proper use of primary keys, foreign keys, and data types

consider indexing columns used in WHERE, JOIN and ORDER BY clauses for improved performance.

Conceptual ER diagram:



Fines
fine-id (PK)
checkout-id (FK)
amount
fine-date

Reservations
reservation-id — (PK)
user-id — (FK)
resource-id — (FK)
reservation-date

Logical Design (ER diagram with Foreign Keys)

users
user-id — (PK)
name
email
phone

BOOKS
book-id — (PK)
title
author-id — (FK)
genre
availability(B)

e-books
E-book-id — (PK)
title
author-id — (FK)
genre
availability(B)

checkout History
checkout-id
user-id — (FK)
resource-id
resource-type (V)
checkout-date
return-date

checkout History
checkout-id
user-id (FK)
resource-id
resource-type (V)
checkout-date
return-date

Fines
Fine-id (PK)
checkout-id — (FK)
amount (D)
Fine-date (D)

Reservations
Reservation-id (PK)
user-id (FK)
resource-id (FK)
reservation-date (D)

journals

journal-id (pk)

title (c)

publisher (c)

Authors

author-id (pk)

name (c)

biography (c)

Physical ER-Diagram for the Academic Library System

Authors

author-id (int, pk)

name (varchar(100))

bio (text)

Books

book-id (int, pk)

title (varchar(255))

author-id (int, fk)

publisher (varchar(100))

published-date (date)

isbn (varchar(13))

E-books

ebook-id (int, pk)

title (varchar(255))

author-id (int, fk)

publisher (varchar(100))

file-format (varchar(50))

published-date (date)

journals

journal-id (int, pk)

title (varchar(255))

publisher (varchar(100))

issue-date (date)

volume (varchar(50))

users

user-id (int, pk)

name (varchar(100))

email (varchar(100))

phone (varchar(15))

registration-date (date)

checkout History

checkout-id (INT, PK)
user-id (INT, FK)
resource-id (INT, FK)
resource-type (ENUM)
checkout-date (DATE)
return-date (DATE)

Reservations

reservation-id (INT, PK)
user-id (INT, FK)
resource-type (ENUM)
resource-id (INT, FK)
reservation-date (DATE)

SQL Statements for the Academic Library system.

Table creation.

```
CREATE TABLE Authors (  
  Author_ID INT Primary Key,  
  Name VARCHAR(100),  
  Biography TEXT  
);
```

CREATE TABLE BOOKS (

BOOK-ID INT PRIMARY KEY,

Title VARCHAR (200),

Author-ID INT,

Publisher VARCHAR (100),

Publication-Date DATE,

ISBN VARCHAR (20)

Availability BOOLEAN,

FOREIGN KEY (Author-ID) REFERENCES Authors

(Author-ID)

);

CREATE TABLE E-BOOKS (

EBOOK-ID INT PRIMARY KEY,

Title VARCHAR (200),

Author-ID INT,

Publisher VARCHAR (100),

Publication-Date DATE,

ISBN VARCHAR (20)

File-path VARCHAR (255),

Availability BOOLEAN,

FOREIGN KEY (Author-ID) REFERENCES

Authors (Author-ID)

);

```
CREATE TABLE journals (
```

```
journal_ID INT PRIMARY KEY,
```

```
Title VARCHAR (200),
```

```
Publisher VARCHAR (100),
```

```
Publication - Date DATE,
```

```
ISSN VARCHAR (20),
```

```
Availability BOOLEAN);
```

```
CREATE TABLE users (
```

```
User-ID INT PRIMARY KEY,
```

```
Name VARCHAR (100),
```

```
Email VARCHAR (100),
```

```
Password VARCHAR (255)
```

```
);
```

```
CREATE TABLE CHECKOUT-HISTORY (
```

```
checkout-ID INT PRIMARY KEY,
```

```
User-ID INT,
```

```
Book-ID INT,
```


EBOOK_ID INT,

checkout - date DATE

Return - date DATE,

Overdue - fine DECIMAL(10,2),

FOREIGN KEY (user-ID) REFERENCES users (user-ID),

FOREIGN KEY (BOOK-ID) REFERENCES BOOKS (BOOK-ID),

FOREIGN KEY (EBOOK-ID) REFERENCES E-BOOKS (EBOOK-ID),

FOREIGN KEY (journal-ID) REFERENCES journals (journal-ID)

);

CREATE TABLE Reservations (

Reservation-ID INT PRIMARY KEY,

User-ID INT,

BOOK-ID INT,

EBOOK-ID INT,

Journal-ID, INT,

Reservation - date DATE,

FOREIGN KEY (user-ID) REFERENCES users (user-ID),

FOREIGN KEY (BOOK-ID) REFERENCES BOOKS (BOOK-ID),

FOREIGN KEY (EBOOK-ID) REFERENCES E-BOOKS (EBOOK-ID),

FOREIGN KEY (journal-ID) REFERENCES journals (journal-ID)

);

Conclusion :

The Academic library system with Digital Resources database design successfully integrates physical and digital resource Management, providing an efficient and user-friendly platform for library operations. Finally The Academic library system with Digital Resources database design efficiently and addresses the complex needs of modern libraries, integrating physical and digital resources.