SQL ACE ACADAMY CERTIFICATION

**Submitted to :**Ace Academy

Submitted by:

**Nagulapalli Deepika**

**Malla Reddy University powered by sunstone,**

**Hyderabad**

**PROJECT-1**

**LIBRARY MANAGEMENT SYSTEM**

**Overview:**

An online library management system offers a user-friendly way of issuing books and viewing different books and titles available under a category. This type of Management Information System (MIS) can be developed in Asp.Net using C#, with SQL queries enabling quick retrieval of the required information.

In a typical college library, both teachers and students can issue books. The return period varies for both groups, and each book has a unique ID, even if they are copies of the same book by the same author. The system captures details such as who has issued the book, the issue duration, and any applicable fines.

**TASK 1:**

To create an online library management system, we need to define the database schema using an ER diagram and implement it in MySQL. The entities and relationships must capture all the necessary details for both students and teachers, including book issuance, return periods, and fines.

**Entities and Attributes:**

1. Students

student id (Primary Key)

first name

last name

date of birth

email

2. Teachers

teacher id (Primary Key)

-first name

- last name

- department

- email

3. Books

- book id (Primary Key, unique for each copy)

- title

- author

- publication year

- isbn

- category

4. issued Books

- issue id (Primary Key)

- Book id (Foreign Key from Books) Teachers)

- User type (Indicates whether the user is a student or teacher)

- Issue date

- Due date

- Return date

- fine

**SQL Statements**

SQL

**Creating the Students table**

CREATE TABLE Students (

Student id INT AUTO\_INCREMENT PRIMARY KEY,

First name VARCHAR(50) NOT NULL,

Last name VARCHAR(50) NOT NULL,

Date of birth DATE NOT NULL,

Email VARCHAR(100) NOT NULL UNIQUE);

**Creating the Teachers table**

CREATE TABLE Teachers (

Teacher id INT AUTO\_INCREMENT PRIMARY KEY,

First name VARCHAR(50) NOT NULL,

Last name VARCHAR(50) NOT NULL,

department VARCHAR(50) NOT NULL,

email VARCHAR(100) NOT NULL UNIQUE);

**Creating the Books table**

CREATE TABLE Books (

Book d INT AUTO\_INCREMENT PRIMARY KEY,

Title VARCHAR(100) NOT NULL,

Author VARCHAR(100) NOT NULL,

Publication year YEAR NOT NULL,

Isbn VARCHAR(20) NOT NULL UNIQUE,

Category VARCHAR(50) NOT NULL);

**Creating the Issued Books table to manage book issuances**

CREATE TABLE Issued Books (

Issue id INT AUTO\_INCREMENT PRIMARY KEY,

Book id INT NOT NULL,

User id INT NOT NULL,

User type ENUM('Student', 'Teacher') NOT NULL,

Issue date DATE NOT NULL,

Due date DATE NOT NULL,

Return date DATE,

Fine DECIMAL(10, 2),

FOREIGN KEY (book id) REFERENCES Books(book id),

A check constraint to ensure user id references either Students or Teachers based on user type

FOREIGN KEY (user id) REFERENCES Students(student id) ON DELETE CASCADE,

FOREIGN KEY (user id) REFERENCES Teachers(teacher id) ON DELETE CASCADE);

**Explanation:**

1. Students Table: Stores details about students.

2. Teachers Table: Stores details about teachers.

3. Books Table: Stores details about books, each book copy has a unique ID.

4. Issued Books Table: Manages book issuances, includes user type to differentiate between students and teachers, and captures the issue date, due date, return date, and fine if applicable.

**Steps to Create the ER Model File**

1. Open MySQL Workbench.

2. Create a new model:

- Go to File > New Model.

3. Create the tables:

- Go to the Model menu and select Add Diagram.

- In the new diagram, right-click and select Create Table... for each of the entities (Students, Teachers, Books, Issued Books).

- Define the columns and set the primary keys and foreign keys according to the SQL definitions provided.

4. Define Relationships:

- Use the Place a Relationship tools from the toolbar to connect the tables according to the described relationships.

5. Export the Diagram:

- Once the diagram is complete, go to File > Export > Export as PNG... or Export as PDF... and save the file to your computer.

**Uploading to GitHub**

1. Create a GitHub Account:

- If you don’t already have one, sign up at [GitHub](https://github.com).

2. Create a New Repository:

- Go to your GitHub profile and click on the Repositories tab.

- Click the New button to create a new repository.

- Fill in the repository name, description, and set

the repository to either public or private.

- Click Create repository.

3. Clone the Repository to Your Local Machine:

- Open a terminal or command prompt.

- Navigate to the directory where you want to clone the repository.

- Use the command:

sh

git clone https://github.com/your-username/your-repository.git

4. Add the ER Model File:

- Copy the ER model file (e.g., ER\_Diagram.png or ER\_Diagram.pdf) to the cloned repository directory.

5. Commit and Push the Changes:

- In the terminal, navigate to your repository directory:

sh

cd your-repository

- Add the file to the staging area:

sh

git add ER\_Diagram.png

- Commit the changes:

sh

git commit -m "Add ER diagram file"

- Push the changes to GitHub:

sh

git push origin main

Following these steps, your ER model file will be uploaded to your GitHub repository.

GitHub link:

https://github.com/davidjaimes/sql-employee-database