

SUBMIITED TO:

MA’AM ASMARA NISAR

SUBMITTED BY: MARKS OBTAINED:

Muhammad Zohaib SP24-BBD-108

Fawad Saqib SP24-BBD-047

Awais Manzoor SP24-BBD-026

Faiz ullah SP24-BBD-042

Khan Khattak SP24-BBD-100

DATABASES FOR BUSINESS

LIBRARY MANAGEMENT SYSTEM REPORT

 

**Library Management System – Oracle SQL Project Report**

**Project Title:**  
 Library Management System

**1. Objective:**  
The purpose of this project is to design a comprehensive and normalized database for a Library Management System that keeps track of students, books, borrowing activities, fines, suspensions, and librarian operations. The system ensures smooth operation and effective record management in a library environment.

**2. Technologies Used:**

* Oracle 11g
* SQL Developer
* SQL (DDL & DML)

**3. ERD Overview:**

|  |
| --- |
| Entities: |
| * Student |
| * Author |
| * Genre |
| * Librarian |
| * Book |
| * Book\_Info |
| * Librarian\_Info |
| * Borrow\_Date\_Details |
| * Borrowing |
| * Fine\_Reason |
| * Fine\_Status |
| * Fine |
| * Suspension |
| * Issue\_History |

Relationships:

* Book references Author and Genre
* Borrowing connects Student, Book, Librarian, and Borrow\_Date\_Details
* Fine connects to Borrowing, Fine\_Reason, and Fine\_Status
* Suspension is linked to Student and Librarian
* Issue\_History logs borrowing history linked to Student, Fine, Book, and Librarian

**4. Database Schema (DDL):**

**Student Table**

|  |
| --- |
| CREATE TABLE Student ( |
| Student\_ID VARCHAR2(20) PRIMARY KEY, |
| Student\_Name VARCHAR2(50) NOT NULL, |
| Reg\_Date DATE NOT NULL, |
| Contact\_No VARCHAR2(15) NOT NULL, |
| Email VARCHAR2(100) |
| ); |

**Author Table**

|  |
| --- |
| CREATE TABLE Author ( |
| Author\_ID NUMBER PRIMARY KEY, |
| Author\_Name VARCHAR2(100) NOT NULL |
| ); |

**Genre Table**

|  |
| --- |
| CREATE TABLE Genre ( |
| Genre\_ID NUMBER PRIMARY KEY, |
| Genre\_Name VARCHAR2(100) NOT NULL |
| ); |

**LIBRARIAN**

|  |
| --- |
| CREATE TABLE Librarian ( |
| Librarian\_ID VARCHAR2(10) PRIMARY KEY, |
| Librarian\_Name VARCHAR2(50) NOT NULL, |
| Contact\_Info VARCHAR2(50), |
| Gender VARCHAR2(10), |
| Working\_Hours VARCHAR2(30) |
| ); |

**Book Table**

|  |
| --- |
| CREATE TABLE Book ( |
| Book\_ID VARCHAR2(10) PRIMARY KEY, |
| Title VARCHAR2(100) NOT NULL, |
| Condition VARCHAR2(20) NOT NULL, |
| Rating NUMBER(3,1), |
| Author\_ID NUMBER NOT NULL, |
| Genre\_ID NUMBER, |
| Available VARCHAR2(3) CHECK (Available IN ('Yes', 'No')), |
| CONSTRAINT fk\_book\_author FOREIGN KEY (Author\_ID) REFERENCES Author(Author\_ID), |
| CONSTRAINT fk\_book\_genre FOREIGN KEY (Genre\_ID) REFERENCES Genre(Genre\_ID) |
| ); |

**Borrowing Table**

|  |
| --- |
| CREATE TABLE Borrowing ( |
| Borrow\_ID VARCHAR2(10) PRIMARY KEY, |
| Student\_ID VARCHAR2(20) NOT NULL, |
| Book\_ID VARCHAR2(10) NOT NULL, |
| Librarian\_ID VARCHAR2(10) NOT NULL, |
| Borrow\_Date DATE NOT NULL, |
| Due\_Date DATE NOT NULL, |
| Return\_Date DATE, |
| Status VARCHAR2(20) CHECK (Status IN ('Borrowed', 'Returned', 'Overdue')), |
| CONSTRAINT fk\_borrowing\_student FOREIGN KEY (Student\_ID) REFERENCES Student(Student\_ID), |
| CONSTRAINT fk\_borrowing\_book FOREIGN KEY (Book\_ID) REFERENCES Book(Book\_ID), |
| CONSTRAINT fk\_borrowing\_librarian FOREIGN KEY (Librarian\_ID) REFERENCES Librarian(Librarian\_ID), |
| CONSTRAINT fk\_borrowing\_date FOREIGN KEY (Borrow\_Date) REFERENCES Borrow\_Date\_Details(Borrow\_Date) |
| ); |

**Fine\_Reason Table**

|  |
| --- |
| CREATE TABLE Fine\_Reason ( |
| Reason\_ID NUMBER PRIMARY KEY, |
| Description VARCHAR2(100) NOT NULL |
| ); |

**Fine\_Status Table**

|  |
| --- |
| CREATE TABLE Fine\_Status ( |
| Status\_ID NUMBER PRIMARY KEY, |
| Description VARCHAR2(20) NOT NULL |
| ); |

**Fine Table**

|  |
| --- |
| CREATE TABLE Fine ( |
| Fine\_ID VARCHAR2(10) PRIMARY KEY, |
| Borrow\_ID VARCHAR2(10) NOT NULL, |
| Amount NUMBER(10, 2) NOT NULL, |
| Reason\_ID NUMBER NOT NULL, |
| Status\_ID NUMBER NOT NULL, |
| Issue\_Date DATE NOT NULL, |
| Payment\_Date DATE, |
| CONSTRAINT fk\_fine\_borrow FOREIGN KEY (Borrow\_ID) REFERENCES Borrowing(Borrow\_ID), |
| CONSTRAINT fk\_fine\_reason FOREIGN KEY (Reason\_ID) REFERENCES Fine\_Reason(Reason\_ID), |
| CONSTRAINT fk\_fine\_status FOREIGN KEY (Status\_ID) REFERENCES Fine\_Status(Status\_ID) |
| ); |

|  |
| --- |
| CREATE TABLE Suspension ( |
| Suspension\_ID NUMBER PRIMARY KEY, |
| Student\_ID VARCHAR2(20) NOT NULL, |
| Reason VARCHAR2(255) NOT NULL, |
| Dues\_Amount NUMBER(10,2), |
| Issued\_By VARCHAR2(10) NOT NULL, |
| Issue\_Date DATE NOT NULL, |
| End\_Date DATE, |
| CONSTRAINT fk\_suspension\_student FOREIGN KEY (Student\_ID) REFERENCES Student(Student\_ID), |
| CONSTRAINT fk\_suspension\_librarian FOREIGN KEY (Issued\_By) REFERENCES Librarian(Librarian\_ID) |
| ); |

**Suspension Table**

**Issue\_History Table**

|  |
| --- |
| CREATE TABLE ISSUE\_HISTORY ( |
| ish\_id VARCHAR2(10) PRIMARY KEY, |
| STUDENT\_ID VARCHAR2(20), |
| FINE\_ID VARCHAR2(10), |
| BOOK\_ID VARCHAR2(10), |
| Librarian\_ID VARCHAR2(10), |
| Issue\_Date DATE, |
| Return\_Date DATE, |
| CONSTRAINT issueh\_STUDENT\_ID\_fk FOREIGN KEY (STUDENT\_ID) REFERENCES Student(Student\_ID), |
| CONSTRAINT issueh\_FINE\_ID\_fk FOREIGN KEY (FINE\_ID) REFERENCES Fine(Fine\_ID), |
| CONSTRAINT issueh\_BOOK\_ID\_fk FOREIGN KEY (BOOK\_ID) REFERENCES Book(Book\_ID), |
| CONSTRAINT issueh\_Librarian\_ID\_fk FOREIGN KEY (Librarian\_ID) REFERENCES Librarian(Librarian\_ID) |

**Book\_Info Table**

|  |
| --- |
| CREATE TABLE BOOK\_INFO ( |
| book\_info\_id NUMBER PRIMARY KEY, |
| BOOK\_ID VARCHAR2(10), |
| Title VARCHAR2(50), |
| CONSTRAINT book\_info\_BOOK\_ID\_fk FOREIGN KEY (BOOK\_ID) REFERENCES Book(Book\_ID) |
| ); |

**Librarian\_Info Table**

|  |
| --- |
| CREATE TABLE LIBRARIAN\_INFO ( |
| librarian\_info\_id NUMBER PRIMARY KEY, |
| Librarian\_ID VARCHAR2(10), |
| Librarian\_Name VARCHAR2(30), |
| CONSTRAINT librarian\_info\_Librarian\_ID\_fk FOREIGN KEY (Librarian\_ID) REFERENCES Librarian(Librarian\_ID) |
| ); |

**5. Features Covered:**

* Tracking student registrations
* Book availability and metadata management
* Borrowing transactions and dates
* Fine calculation and status handling
* Suspension mechanism based on dues
* Complete issue history for auditing

