**Case Study: Library Management System**

**Problem Statement:**

Design and implement a Library Management System using Oracle SQL and PL/SQL. The system will be used to manage books, members, and borrowing transactions. Your task is to create the necessary database schema, populate the database with sample data, and develop PL/SQL procedures to handle book borrowing, returning, and fine calculation.

**Requirements:**

1. **Book Management**:
   * Implement the functionality to add, update, delete, and search for books.
   * Ensure that each book has attributes such as BOOK\_ID, TITLE, AUTHOR, PUBLISHER, CATEGORY, and AVAILABLE\_COPIES.
2. **Member Management**:
   * Implement the functionality to add, update, delete, and search for members.
   * Ensure that each member has attributes such as MEMBER\_ID, FIRST\_NAME, LAST\_NAME, EMAIL, PHONE\_NUMBER, and ADDRESS.
3. **Borrowing Transactions**:
   * Implement the functionality to create, update, and manage borrowing transactions.
   * Track transactions with attributes such as TRANSACTION\_ID, MEMBER\_ID, BOOK\_ID, BORROW\_DATE, RETURN\_DATE, and STATUS.

**Tasks:**

1. **Design the Database Schema**:
   * Create the Books, Members, and Borrowing\_Transactions tables with the appropriate fields and constraints.
   * Define primary keys and foreign keys to maintain data integrity.
2. **Populate the Database with Sample Data**:
   * Insert sample records into the Books, Members, and Borrowing\_Transactions tables to facilitate testing of the system.
3. **Develop PL/SQL Procedures**:
   * Create a procedure to handle borrowing a book. The procedure should check book availability, insert a new borrowing transaction record, and update the book's available copies.
   * Create a procedure to handle returning a book. The procedure should update the borrowing transaction record with the return date, calculate any applicable fines, and update the book's available copies.
   * Create a procedure to calculate fines for overdue books based on the borrowing duration.

**Expected Outcomes:**

1. **Books Table**:
   * Contains all information about the books available in the library.
2. **Members Table**:
   * Contains details of all registered members of the library.
3. **Borrowing\_Transactions Table**:
   * Tracks the borrowing transactions, including borrow dates, return dates, and statuses.
4. **PL/SQL Procedures**:
   * Efficiently manage book borrowing, returning, and fine calculation, maintaining accurate records in the database.

**Deliverables:**

1. SQL scripts to create the Books, Members, and Borrowing\_Transactions tables.
2. SQL scripts to insert sample data into the tables.
3. PL/SQL scripts for the procedures to borrow books, return books, and calculate fines.
4. Documentation explaining how to set up and use the system, including how to run the PL/SQL procedures.

**Database Schema:**

1. **Books Table**:
   * **BOOK\_ID**: Number, Primary Key
   * **TITLE**: Varchar2(100)
   * **AUTHOR**: Varchar2(50)
   * **PUBLISHER**: Varchar2(50)
   * **CATEGORY**: Varchar2(50)
   * **AVAILABLE\_COPIES**: Number
2. **Members Table**:
   * **MEMBER\_ID**: Number, Primary Key
   * **FIRST\_NAME**: Varchar2(50)
   * **LAST\_NAME**: Varchar2(50)
   * **EMAIL**: Varchar2(100)
   * **PHONE\_NUMBER**: Varchar2(15)
   * **ADDRESS**: Varchar2(200)
3. **Borrowing\_Transactions Table**:
   * **TRANSACTION\_ID**: Number, Primary Key
   * **MEMBER\_ID**: Number, Foreign Key References Members(MEMBER\_ID)
   * **BOOK\_ID**: Number, Foreign Key References Books(BOOK\_ID)
   * **BORROW\_DATE**: Date
   * **RETURN\_DATE**: Date
   * **STATUS**: Varchar2(20)

**Case Study Task:**

* **Design**: Create the database schema as provided.
* **Implement**: Insert sample data into the tables.
* **Develop**: Write PL/SQL procedures for borrowing books, returning books, and calculating fines.
* **Test**: Test the procedures with various scenarios (e.g., borrowing a book, returning a book, calculating fines, ensuring proper updates).