**Case Study: Online Shopping System**

**Problem Statement:**

Design and implement an Online Shopping System using Oracle SQL and PL/SQL. The system will be used to manage products, customers, and orders. Your task is to create the necessary database schema, populate the database with sample data, and develop PL/SQL procedures to handle order placement, order status updates, and generating order summaries.

**Requirements:**

1. **Product Management**:
   * Implement the functionality to add, update, delete, and search for products.
   * Ensure that each product has attributes such as PRODUCT\_ID, PRODUCT\_NAME, CATEGORY, PRICE, STOCK\_QUANTITY, and DESCRIPTION.
2. **Customer Management**:
   * Implement the functionality to add, update, delete, and search for customers.
   * Ensure that each customer has attributes such as CUSTOMER\_ID, FIRST\_NAME, LAST\_NAME, EMAIL, PHONE\_NUMBER, and ADDRESS.
3. **Order Management**:
   * Implement the functionality to place orders, update order status, and track order details.
   * Track orders with attributes such as ORDER\_ID, CUSTOMER\_ID, ORDER\_DATE, ORDER\_STATUS, and TOTAL\_AMOUNT.

**Tasks:**

1. **Design the Database Schema**:
   * Create the Products, Customers, and Orders 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 Products, Customers, and Orders tables to facilitate testing of the system.
3. **Develop PL/SQL Procedures**:
   * Create a procedure to handle order placement. The procedure should check product availability, insert a new order record, and update product stock quantity.
   * Create a procedure to handle order status updates. The procedure should update the status of the order based on its progress.
   * Create a procedure to generate order summaries, including details such as order ID, customer information, order date, and total amount.

**Expected Outcomes:**

1. **Products Table**:
   * Contains all information about the products available for purchase.
2. **Customers Table**:
   * Contains details of all registered customers.
3. **Orders Table**:
   * Tracks the orders placed by customers, including order dates and statuses.
4. **PL/SQL Procedures**:
   * Efficiently manage order placement, status updates, and generating order summaries, maintaining accurate records in the database.

**Deliverables:**

1. SQL scripts to create the Products, Customers, and Orders tables.
2. SQL scripts to insert sample data into the tables.
3. PL/SQL scripts for the procedures to place orders, update order status, and generate order summaries.
4. Documentation explaining how to set up and use the system, including how to run the PL/SQL procedures.

**Database Schema:**

1. **Products Table**:
   * **PRODUCT\_ID**: Number, Primary Key
   * **PRODUCT\_NAME**: Varchar2(100)
   * **CATEGORY**: Varchar2(50)
   * **PRICE**: Number
   * **STOCK\_QUANTITY**: Number
   * **DESCRIPTION**: Varchar2(200)
2. **Customers Table**:
   * **CUSTOMER\_ID**: Number, Primary Key
   * **FIRST\_NAME**: Varchar2(50)
   * **LAST\_NAME**: Varchar2(50)
   * **EMAIL**: Varchar2(100)
   * **PHONE\_NUMBER**: Varchar2(15)
   * **ADDRESS**: Varchar2(200)
3. **Orders Table**:
   * **ORDER\_ID**: Number, Primary Key
   * **CUSTOMER\_ID**: Number, Foreign Key References Customers(CUSTOMER\_ID)
   * **ORDER\_DATE**: Date
   * **ORDER\_STATUS**: Varchar2(20)
   * **TOTAL\_AMOUNT**: Number

**Case Study Task:**

* **Design**: Create the database schema as provided.
* **Implement**: Insert sample data into the tables.
* **Develop**: Write PL/SQL procedures for order placement, updating order status, and generating order summaries.
* **Test**: Test the procedures with various scenarios (e.g., placing an order, updating order status, generating summaries, ensuring proper updates).