

E-COMMERCE WEBSITE PROJECT

```
import java.util.*;

class Product {
    private int id;
    private String name;
    private double price;
    private int stock;

    public Product(int id, String name, double price, int stock) {
        this.id = id;
        this.name = name;
        this.price = price;
        this.stock = stock;
    }

    public int getId() {
        return id;
    }

    public String getName() {
        return name;
    }

    public double getPrice() {
        return price;
    }

    public int getStock() {
        return stock;
    }
}
```

```
public void reduceStock(int quantity) {  
    if (quantity <= stock) {  
        stock -= quantity;  
    }  
}
```

```
public void increaseStock(int quantity) {  
    stock += quantity;  
}
```

```
@Override  
public String toString() {  
    return "ID: " + id + ", Name: " + name + ", Price: $" + price + ", Stock: " + stock;  
}  
}
```

```
class ECommerce {  
    private Map<Integer, Product> products = new HashMap<>();  
    private List<String> orderHistory = new ArrayList<>();  
  
    public void addProduct(Product product) {  
        products.put(product.getId(), product);  
    }  
  
    public void viewProducts() {  
        if (products.isEmpty()) {  
            System.out.println("No products available.");  
            return;  
        }  
        for (Product product : products.values()) {
```

```
        System.out.println(product);
    }
}
```

```
public void placeOrder(int productId, int quantity) {
    Product product = products.get(productId);
    if (product != null) {
        if (product.getStock() >= quantity) {
            product.reduceStock(quantity);
            orderHistory.add("Product: " + product.getName() + ", Quantity: " + quantity + ", Total: $" +
(quantity * product.getPrice()));
            System.out.println("Order placed successfully for " + quantity + " unit(s) of " +
product.getName());
        } else {
            System.out.println("Insufficient stock for the product.");
        }
    } else {
        System.out.println("Product not found.");
    }
}
```

```
public void viewOrders() {
    if (orderHistory.isEmpty()) {
        System.out.println("No orders placed yet.");
    } else {
        for (String order : orderHistory) {
            System.out.println(order);
        }
    }
}
```

```
public void addSampleProducts() {
```

```

        addProduct(new Product(101, "Laptop", 750.5, 10));
        addProduct(new Product(102, "Smartphone", 500.0, 20));
        addProduct(new Product(103, "Headphones", 50.0, 50));
        addProduct(new Product(104, "Smartwatch", 200.0, 15));
    }
}

```

```

class Main { // Use 'Main' as the class name for online compilers

```

```

    public static void main(String[] args) {

```

```

        Scanner scanner = new Scanner(System.in);

```

```

        ECommerce eCommerce = new ECommerce();

```

```

        // Adding some sample products

```

```

        eCommerce.addSampleProducts();

```

```

        while (true) {

```

```

            System.out.println("\nE-Commerce Management System");

```

```

            System.out.println("1. Add Product");

```

```

            System.out.println("2. View Products");

```

```

            System.out.println("3. Place Order");

```

```

            System.out.println("4. View Orders");

```

```

            System.out.println("5. Exit");

```

```

            System.out.print("Enter your choice: ");

```

```

            int choice = scanner.nextInt();

```

```

            scanner.nextLine(); // Consume newline

```

```

            switch (choice) {

```

```

                case 1:

```

```

                    System.out.print("Enter Product ID: ");

```

```

                    int id = scanner.nextInt();

```

```

                    scanner.nextLine(); // Consume newline

```

```
System.out.print("Enter Product Name: ");  
  
String name = scanner.nextLine();  
  
System.out.print("Enter Product Price: ");  
  
double price = scanner.nextDouble();  
  
System.out.print("Enter Product Stock: ");  
  
int stock = scanner.nextInt();  
  
eCommerce.addProduct(new Product(id, name, price, stock));  
  
System.out.println("Product added successfully.");  
  
break;
```

case 2:

```
eCommerce.viewProducts();  
  
break;
```

case 3:

```
System.out.print("Enter Product ID to order: ");  
  
int productId = scanner.nextInt();  
  
System.out.print("Enter Quantity: ");  
  
int quantity = scanner.nextInt();  
  
eCommerce.placeOrder(productId, quantity);  
  
break;
```

case 4:

```
eCommerce.viewOrders();  
  
break;
```

case 5:

```
System.out.println("Exiting...");  
  
scanner.close();  
  
return;
```

default:

```
System.out.println("Invalid choice. Please try again.");
```

```
}
```

```
}
```

```
}
```

```
}
```