ASSIGNMENT 5

Product Class

```
public class Product {
   private String productId;
   private String name;
   private double price;
   private int stockQuantity;

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

   public void updateStockQuantity(int quantity) {
      if (quantity > 0) {
         stockQuantity -= quantity;
      } else {
         System.out.println("Invalid quantity. Please enter a positive integer.");
      }
   }

   // Getters and setters
```

```
    public String getProductId() {
        return productId;
    }

    public String getName() {
        return name;
    }

    public double getPrice() {
        return price;
    }

    public int getStockQuantity() {
        return stockQuantity;
    }
}
```

Customer Class

```
import java.util.ArrayList;
import java.util.List;

public class Customer {
    private String customerId;
    private String name;
    private String email;
    private List<Product> cart;
```

```
public Customer(String customerId, String name, String email) {
    this.customerId = customerId;
    this.name = name;
    this.email = email;
    this.cart = new ArrayList<>();
}

public void addToCart(Product product) {
    cart.add(product);
}

public void removeFromCart(Product product) {
    cart.remove(product);
}

public void viewCart() {
    System.out.println("Cart contents:");
    for (Product product : cart) {
        System.out.println(product.getName() + " - " + product.getPrice());
    }
}

public void checkout() {
    // Create an order and calculate total amount
    Order order = new Order(customerId, cart);
    double totalAmount = order.calculateTotalAmount();
```

```
        System.out.println("Total amount: " + totalAmount);
        // Update stock quantities for each product in the cart
        for (Product product : cart) {
            product.updateStockQuantity(1);
        }
        cart.clear();
    }

    // Getters and setters
    public String getCustomerId() {
        return customerId;
    }

    public String getName() {
        return name;
    }

    public String getEmail() {
        return email;
    }
}
```

Order Class

```
import java.time.LocalDateTime;
import java.util.List;

public class Order {
```

```
private String orderId;
private Customer customer;
private List<Product> products;
private double totalAmount;
private LocalDateTime orderDate;

public Order(String orderId, Customer customer, List<Product> products) {
    this.orderId = orderId;
    this.customer = customer;
    this.products = products;
    this.orderDate = LocalDateTime.now();
}

public double calculateTotalAmount() {
    double total = 0;
    for (Product product : products) {
        total += product.getPrice();
    }
    return total;
}

// Getters and setters
public String getOrderId() {
    return orderId;
}
```

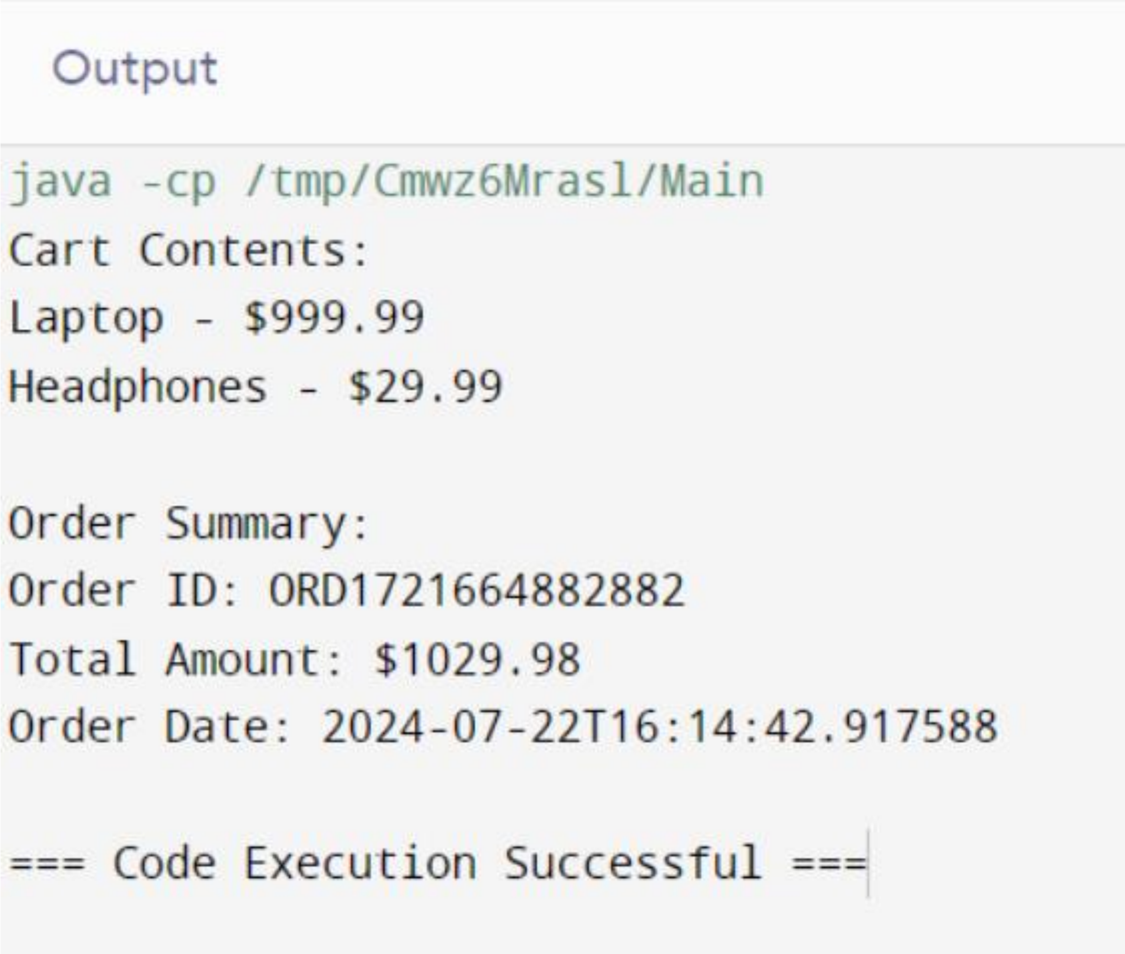
```
    public Customer getCustomer() {
        return customer;
    }

    public List<Product> getProducts() {
        return products;
    }

    public double getTotalAmount() {
        return totalAmount;
    }

    public LocalDateTime getOrderDate() {
        return orderDate;
    }
}
```

Inventory Class

```
import java.util.ArrayList;
import java.util.List;

public class Inventory {
    private List<Product> products;

    public Inventory() {
        this.products = new ArrayList<>();
    }
    public void addProduct(Product product) {
        products.add(product);
    }

    public Product getProductById(String productId) {
        for (Product product : products) {
            if (product.getProductId().equals(productId)) {
                return product;
            }
        }
        return null;
    }

    public void updateProductStock(String productId, int quantity) {
        Product product = getProductById(productId);
        if (product != null) {
            product.updateStockQuantity(quantity);
        }
    }

    // Getters and setters
    public List<Product> getProducts() {
        return products;
    }
}
```

Main Class

```
public class Main {
    public static void main(String[] args) {
        // Create products
        Product product1 = new Product("P001", "Product 1", 10.99, 10);
        Product product2 = new Product("P002", "Product 2", 5.99, 20);

        // Create inventory and add products
        Inventory inventory = new Inventory();
        inventory.addProduct(product1);
        inventory.addProduct(product2);

        // Create customer and add products to cart
        Customer customer = new Customer("C001", "John Doe",
"john.doe@example.com");
        customer.addToCart(product1);
        customer.addToCart(product2);
```



```
java -cp /tmp/Cmwz6Mrasl/Main
Cart Contents:
Laptop - $999.99
Headphones - $29.99

Order Summary:
Order ID: ORD1721664882882
Total Amount: $1029.98
Order Date: 2024-07-22T16:14:42.917588

=== Code Execution Successful ===
```