Day 8 Spring Day-wise Assignment

Case Study 2: Java-Based Configuration - E-Commerce System

Scenario

An e-commerce platform requires product, order, and payment management using Java configuration.

AppConfig.java

```
package com.example.ecommerce;

import org.springframework.context.annotation.Bean;
import org.springframework.context.annotation.Configuration;

@Configuration
public class AppConfig {

    @Bean
    public Product product() {
        return new Product("Laptop", 999.99);
    }

    @Bean
    public Order order() {
        return new Order(product());
    }
}
```

Product.java

```
package com.example.ecommerce;

public class Product {
    private String name;
    private double price;

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

    // Getters and business methods
}
```

Order.java

```
package com.example.ecommerce;

public class Order {
    private Product product;
    private int quantity;
    private String orderId;

    public void createOrder(String orderId, int quantity) {
        this.orderId = orderId;
        this.quantity = quantity;
        System.out.println("Order created: " + orderId + " for " + product.getName() + " x" +
quantity);
```

```
    }

    public void cancelOrder() {
        System.out.println("Order cancelled: " + orderId);
    }

    // Getters and setters
    public Product getProduct() { return product; }
    public void setProduct(Product product) { this.product = product; }
    public int getQuantity() { return quantity; }
    public void setQuantity(int quantity) { this.quantity = quantity; }
    public String getOrderId() { return orderId; }
    public void setOrderId(String orderId) { this.orderId = orderId; }
}
```

Payment.java

```
package com.example.ecommerce;

public class Payment {
    private Order order;
    private double amount;
    private String paymentId;

    public void processPayment() {
        this.amount = order.getProduct().getPrice() * order.getQuantity();
        this.paymentId = "PAY-" + System.currentTimeMillis();
        System.out.println("Payment processed: " + paymentId + " Amount: $" + amount);
    }
```

```
    public void refundPayment() {
        System.out.println("Payment refunded: " + paymentId + " Amount: $" + amount);
    }

    // Getters and setters
    public Order getOrder() { return order; }
    public void setOrder(Order order) { this.order = order; }
    public double getAmount() { return amount; }
    public void setAmount(double amount) { this.amount = amount; }
    public String getPaymentId() { return paymentId; }
    public void setPaymentId(String paymentId) { this.paymentId = paymentId; }
}
```

EcommerceService.java

```
package com.example.ecommerce;

public class EcommerceService {
    private Product product;
    private Order order;
    private Payment payment;

    public void processOrder() {
        product.addProduct("P1001", "Smartphone", 599.99);
        product.listProducts();
        order.createOrder("ORD123", 2);
        payment.processPayment();
    }
```

```
    // Getters and setters
    public Product getProduct() { return product; }
    public void setProduct(Product product) { this.product = product; }
    public Order getOrder() { return order; }
    public void setOrder(Order order) { this.order = order; }
    public Payment getPayment() { return payment; }
    public void setPayment(Payment payment) { this.payment = payment; }
}
```

Main Application

```
import org.springframework.context.annotation.AnnotationConfigApplicationContext;

public class Main {
    public static void main(String[] args) {
        var context = new AnnotationConfigApplicationContext(AppConfig.class);
        Product product = context.getBean(Product.class);
        System.out.println(product.getName());
    }
}
```