

## **POJO Classes:**

Product.java

```
package com.example.ecommerce;
```

```
public class Product {  
    private String productId;  
    private String name;  
    private double price;  
  
    public void addProduct(String productId, String name, double price) {  
        this.productId = productId;  
        this.name = name;  
        this.price = price;  
    }  
  
    public String listProducts() {  
        return "Product: " + name + " (ID: " + productId + "), Price: $" + price;  
    }  
  
    // Getters and setters  
}
```

java

```
// Order.java
```

```
package com.example.ecommerce;
```

```
public class Order {  
    private String orderId;  
    private Product product;
```

```
private int quantity;
```

```
public void createOrder(String orderId, Product product, int quantity) {  
    this.orderId = orderId;  
    this.product = product;  
    this.quantity = quantity;  
}
```

```
public void cancelOrder() {  
    this.orderId = null;  
    this.product = null;  
    this.quantity = 0;  
}
```

```
// Getters and setters  
}
```

```
java
```

```
// Payment.java
```

```
package com.example.ecommerce;
```

```
public class Payment {
```

```
    private Order order;
```

```
    private String paymentMethod;
```

```
    public String processPayment() {
```

```
        return "Processing payment of $" + (order.getProduct().getPrice() * order.getQuantity())  
+  
        " via " + paymentMethod;
```

```
}
```

```
public void refundPayment() {  
    System.out.println("Refunding payment for order " + order.getId());  
}
```

```
// Getters and setters  
}
```

### **Configuration Class:**

```
// 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() {  
        Product product = new Product();  
        product.addProduct("P100", "Smartphone", 599.99);  
        return product;  
    }
```

```
    @Bean
```

```
    public Order order() {  
        Order order = new Order();
```

```
        order.createOrder("ORD100", product(), 2);  
        return order;  
    }  
}
```

@Bean

```
public Payment payment() {  
    Payment payment = new Payment();  
    payment.setOrder(order());  
    payment.setPaymentMethod("Credit Card");  
    return payment;  
}
```

@Bean

```
public EcommerceService ecommerceService() {  
    EcommerceService service = new EcommerceService();  
    service.setProduct(product());  
    service.setOrder(order());  
    service.setPayment(payment());  
    return service;  
}  
}
```

### **Main Application:**

java

```
package com.example.ecommerce;
```

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

```
public class MainApp {
```

```

public static void main(String[] args) {

    AnnotationConfigApplicationContext context =
        new AnnotationConfigApplicationContext(AppConfig.class);

    EcommerceService ecommerceService = context.getBean(EcommerceService.class);

    // Use the ecommerce service

    System.out.println(ecommerceService.getProduct().listProducts());

    System.out.println(ecommerceService.getPayment().processPayment());

    context.close();
}
}

```

### Case Study 3: Annotation-Based Configuration - Library Management System

#### Implementation Details:

##### 1. POJO Classes with Annotations:

// Book.java

```
package com.example.library;
```

```
import org.springframework.stereotype.Component;
```

```
@Component
```

```
public class Book {
```

```
    private String isbn;
```

```
    private String title;
```

```
    private String author;
```

```
    public void addBook(String isbn, String title, String author) {
```

```
    this.isbn = isbn;

    this.title = title;

    this.author = author;
}
```

```
public String searchBook() {

    return "Book: " + title + " by " + author + " (ISBN: " + isbn + ")";

}
```

```
    // Getters and setters
}
```

```
// Member.java
```

```
package com.example.library;
```

```
import org.springframework.stereotype.Component;
```

```
@Component
```

```
public class Member {
```

```
    private String memberId;
```

```
    private String name;
```

```
    public void registerMember(String memberId, String name) {
```

```
        this.memberId = memberId;
```

```
        this.name = name;
```

```
    }
```

```
    public String viewMembers() {
```

```
        return "Member: " + name + " (ID: " + memberId + ")";
```

```
}
```

```
// Getters and setters
```

```
}
```

```
// Loan.java
```

```
package com.example.library;
```

```
import org.springframework.beans.factory.annotation.Autowired;
```

```
import org.springframework.stereotype.Component;
```

```
@Component
```

```
public class Loan {
```

```
    @Autowired
```

```
    private Book book;
```

```
    @Autowired
```

```
    private Member member;
```

```
    private String loanDate;
```

```
    private String dueDate;
```

```
    public void issueBook(String loanDate, String dueDate) {
```

```
        this.loanDate = loanDate;
```

```
        this.dueDate = dueDate;
```

```
    }
```

```
    public void returnBook() {
```

```
        this.loanDate = null;
```

```
        this.dueDate = null;
    }
}
```

```
// Getters and setters
}
```

## 2. Service Class:

```
// LibraryService.java
```

```
package com.example.library;
```

```
import org.springframework.beans.factory.annotation.Autowired;
```

```
import org.springframework.stereotype.Service;
```

```
@Service
```

```
public class LibraryService {
```

```
    @Autowired
```

```
    private Book book;
```

```
    @Autowired
```

```
    private Member member;
```

```
    @Autowired
```

```
    private Loan loan;
```

```
// Methods to manage library operations
```

```
// Getters (optional)
```

```
}
```

**Main Application with Component Scan:**



```
// MainApp.java

package com.example.library;

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

@ComponentScan(basePackages = "com.example.library")
public class MainApp {
    public static void main(String[] args) {
        AnnotationConfigApplicationContext context =
            new AnnotationConfigApplicationContext(MainApp.class);

        LibraryService libraryService = context.getBean(LibraryService.class);

        // Configure some sample data
        libraryService.getBook().addBook("978-3-16-148410-0", "Spring in Action", "Craig Walls");
        libraryService.getMember().registerMember("M100", "Alice Johnson");
        libraryService.getLoan().issueBook("2023-05-15", "2023-06-15");

        // Use the library service
        System.out.println(libraryService.getBook().searchBook());
        System.out.println(libraryService.getMember().viewMembers());

        context.close();
    }
}
```