

E-commerce Billing System with Dynamic Pricing Strategy

Implement an E-commerce billing system that allows dynamic pricing adjustments through the use of the Decorator Pattern. This system will enable the application of discounts, taxes, and promotions to products in a flexible and extendable manner.

Features:

1. **Dynamic Product Pricing:** Adjust product prices dynamically by applying discounts, taxes, and promotions.
2. **User Interaction:** Allow users to enter multiple products with their prices and quantities, and apply various pricing adjustments interactively.
3. **Detailed Billing:** Print a detailed bill that includes the base price, applied discounts, taxes, final price, and total amount.

Usage of Decorator Pattern: The Decorator Pattern is used to add additional functionalities (discounts, taxes, promotions) to the products dynamically at runtime. Each decorator wraps a product and adds its specific behavior to the price calculation, ensuring that the core product class remains unchanged. This approach adheres to the Open/Closed Principle from SOLID principles, allowing the system to be extended without modifying existing code.

Implementation Files

1. **Product.java** - Base Product class
2. **ProductDecorator.java** - Abstract Decorator class
3. **DiscountDecorator.java** - Concrete Decorator for applying discounts
4. **TaxDecorator.java** - Concrete Decorator for applying taxes
5. **PromotionDecorator.java** - Concrete Decorator for applying promotions
6. **Printer.java** - Utility class for printing the bill
7. **ECommerceBillingSystem.java** - Main application class