Decorator Pattern

The Decorator pattern is a structural design pattern that allows behavior to be added to individual objects, either statically or dynamically, without affecting the behavior of other objects from the same class. It is particularly useful in scenarios where you want to add responsibilities to objects in a flexible and reusable way.

In this example, we've implemented the Decorator pattern to create customizable coffee drinks in a coffee shop system. The Coffee interface represents the base component, while the SimpleCoffee class is a basic implementation of a coffee drink.

The CoffeeDecorator is an abstract class that implements the Coffee interface and serves as a base for concrete decorators like MilkDecorator, MapleSyrupDecorator, and WhippedCreamDecorator. These decorators allow us to dynamically add different ingredients to the coffee, modifying its description and cost.