

Decorator

Design Patterns



see what you can learn

Intent

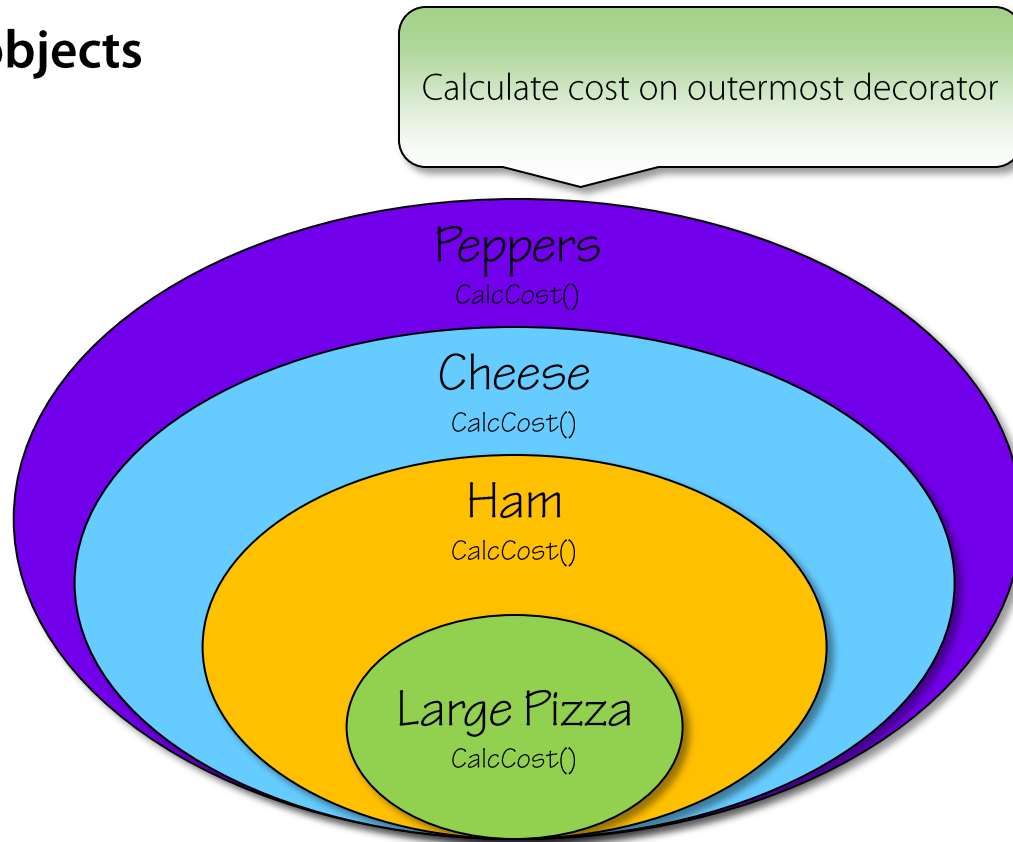
- Add functionality to existing objects dynamically.
- Alternative to sub classing
- Flexible design
- Support Open Closed Principle

Applicability

- Legacy Systems
- Add functionality to controls
- Sealed classes

How Does It Work

- It wrap objects



Consequences

- The original object is unaware of any decorations.
- There is no one big feature-laden class with all the options in it.
- The decorations can be composed together in a mix-and-match fashion.
- Can increase complexity of code

Related Patterns

- Adapter
- Bridge
- Composite
- Façade
- Flyweight
- Proxy

Summary

- Add behavior to objects dynamically
- Wraps the objects
- Support Open Closed Principle
- Implementation