

CSE 3421

Design Pattern

SUMMER 2021

MD. RAFI-UR-RASHID

LECTURER, DEPT. OF CSE, UIU

Structural Patterns

- How objects / classes can be combined
- Seven structural patterns
 - **Adapter**
 - Bridge
 - Composite
 - **Decorator**
 - Façade
 - Flyweight
 - Proxy



Adapter Pattern



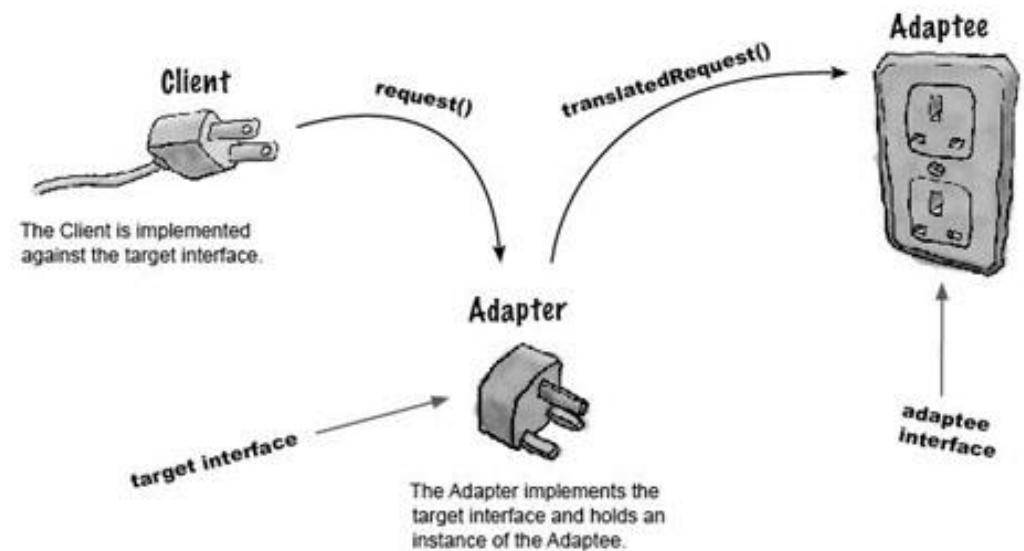
Motivation: Adapter

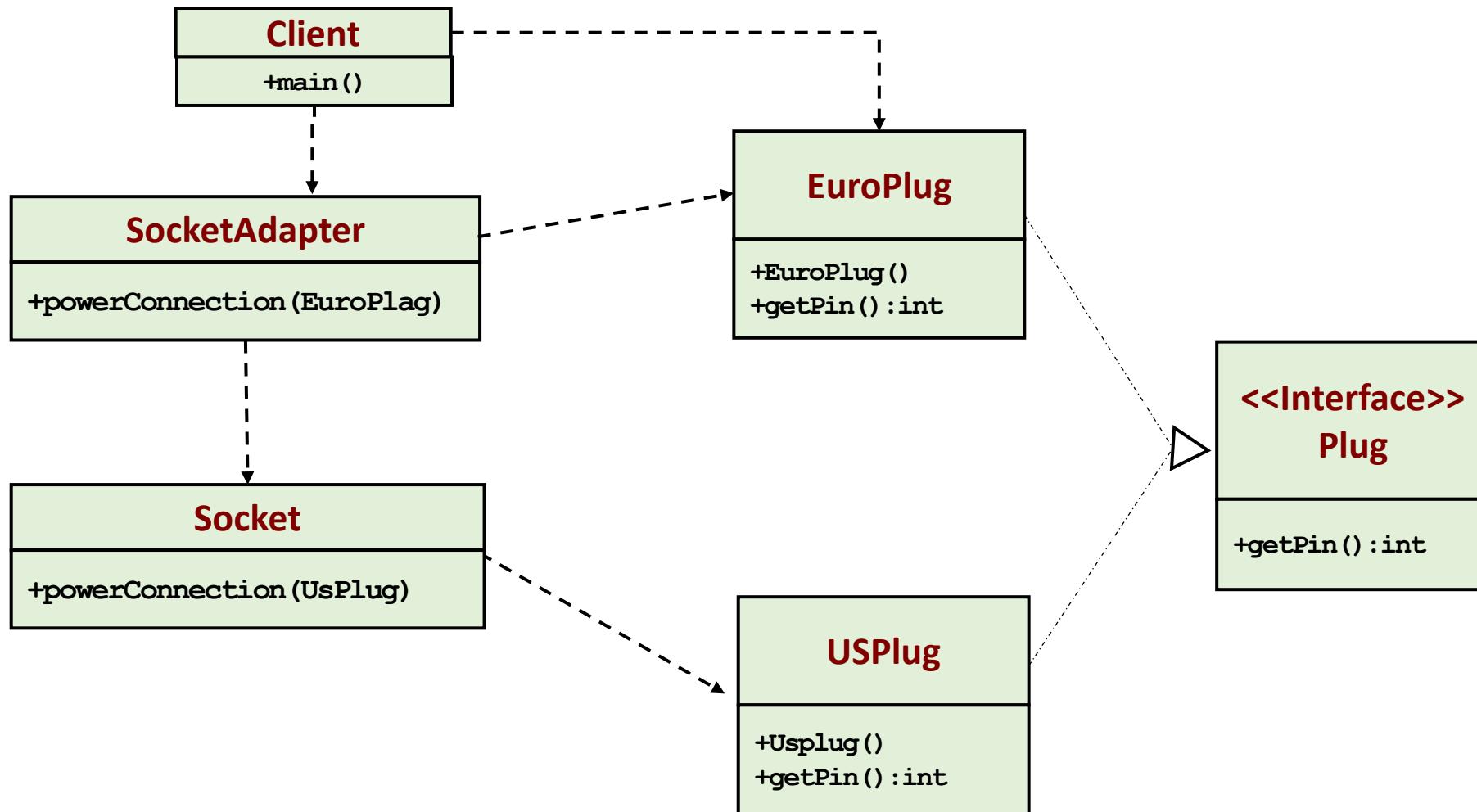


Adapter Method

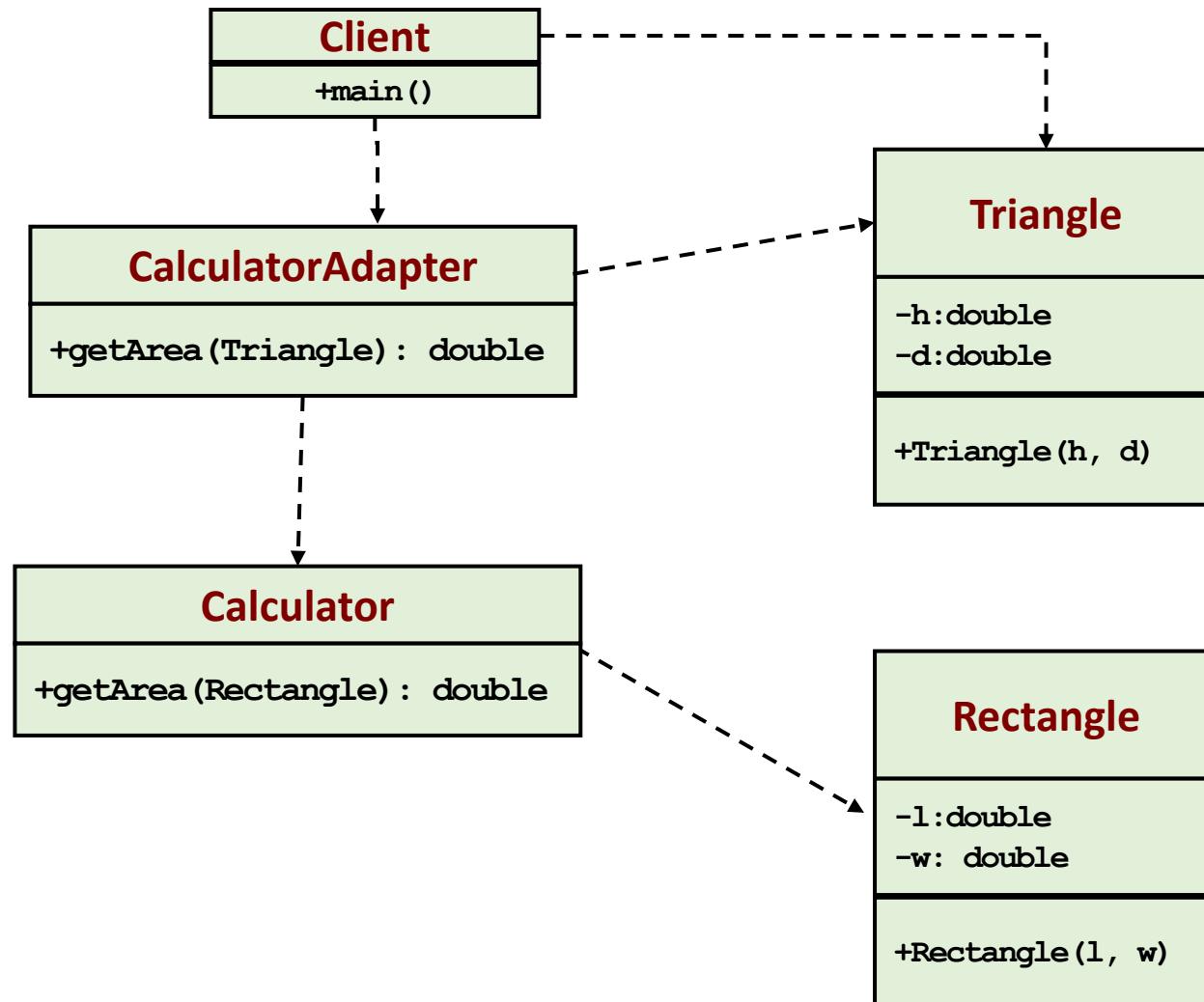
- Help communication between two **incompatible** classes/ interfaces
- An **Adapter class** takes responsibility of resolving such incompatibilities
- Clients are unaware of the complexity of the incompatibility issues
- Allows pre-existing classes to be used in your code.

Example: Plug





Example: Calculator



Practice Problems

Draw UML diagram for following scenarios using appropriate design patterns:

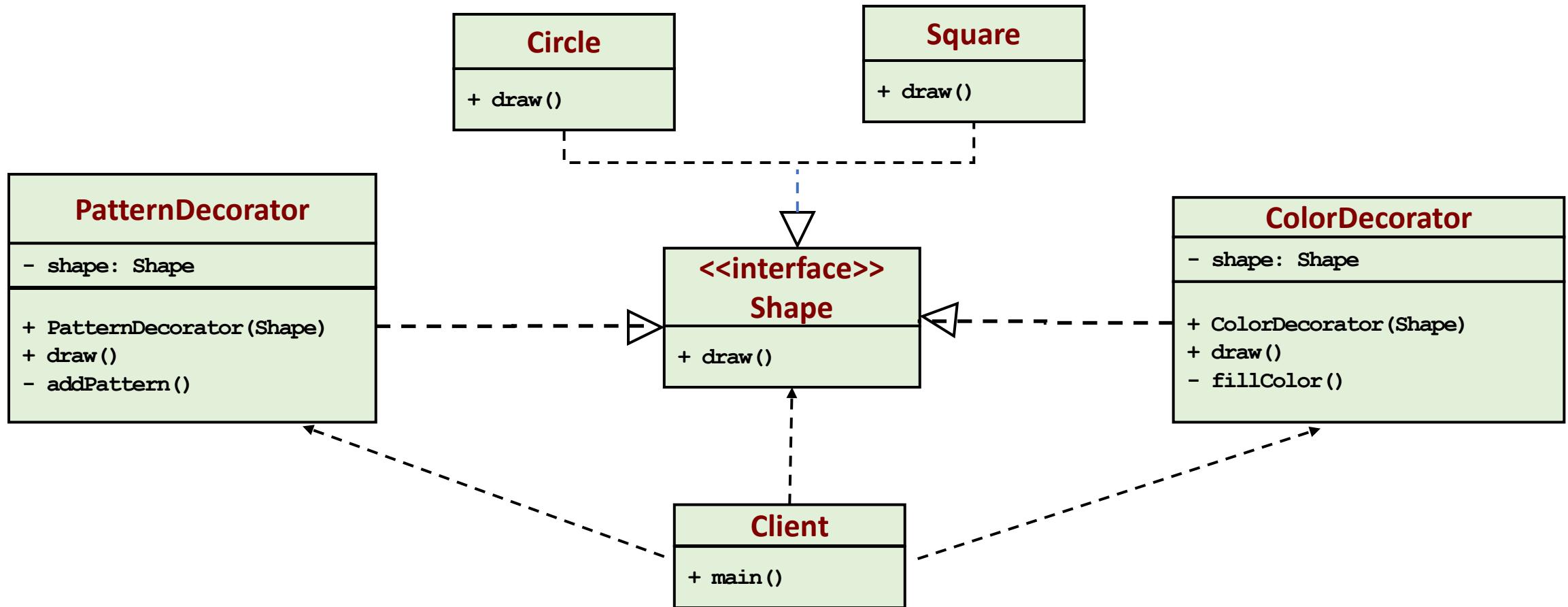
- In our system there is a video player which can play mp4, 3gp and vlc files. But client wants to play some audio files (e.g. mp3, wav) using our system.
- In our system there is a function which sorts an input array in ascending order. But client wants to find the second largest number using our system.

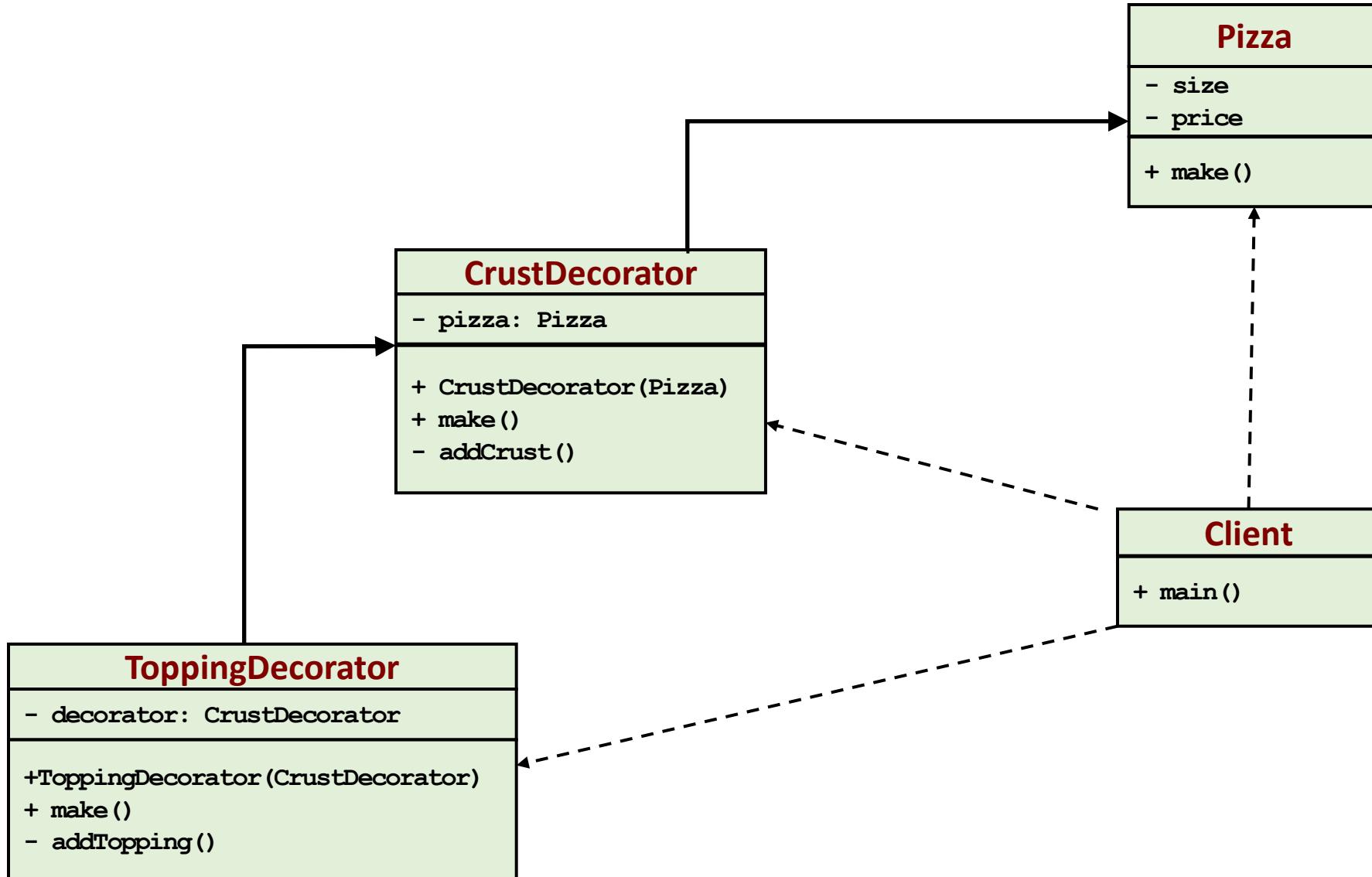
Decorator Pattern



Decorator Method

- We **cannot modify** existing functionalities but we can **extend** them.
- In other words, this pattern is open for extension but closed for modification
- Client-specified embellishment of a core object by recursively wrapping it. Like-wrapping a gift, putting it in a box, and wrapping the box.
- Pay as you go approach





Practice Problems

Draw UML diagram for following scenarios using appropriate design patterns:

- Think of the power-point software. You can make a very basic presentation. Then you can add different features like annotations, transitions, animations etc.
- Think of a website builder (e.g. WordPress). You can start with a very basic template. Then you can gradually add fonts, colors and widgets.



Thank You