

# Design Pattern Assignment

**Maithreyan K**

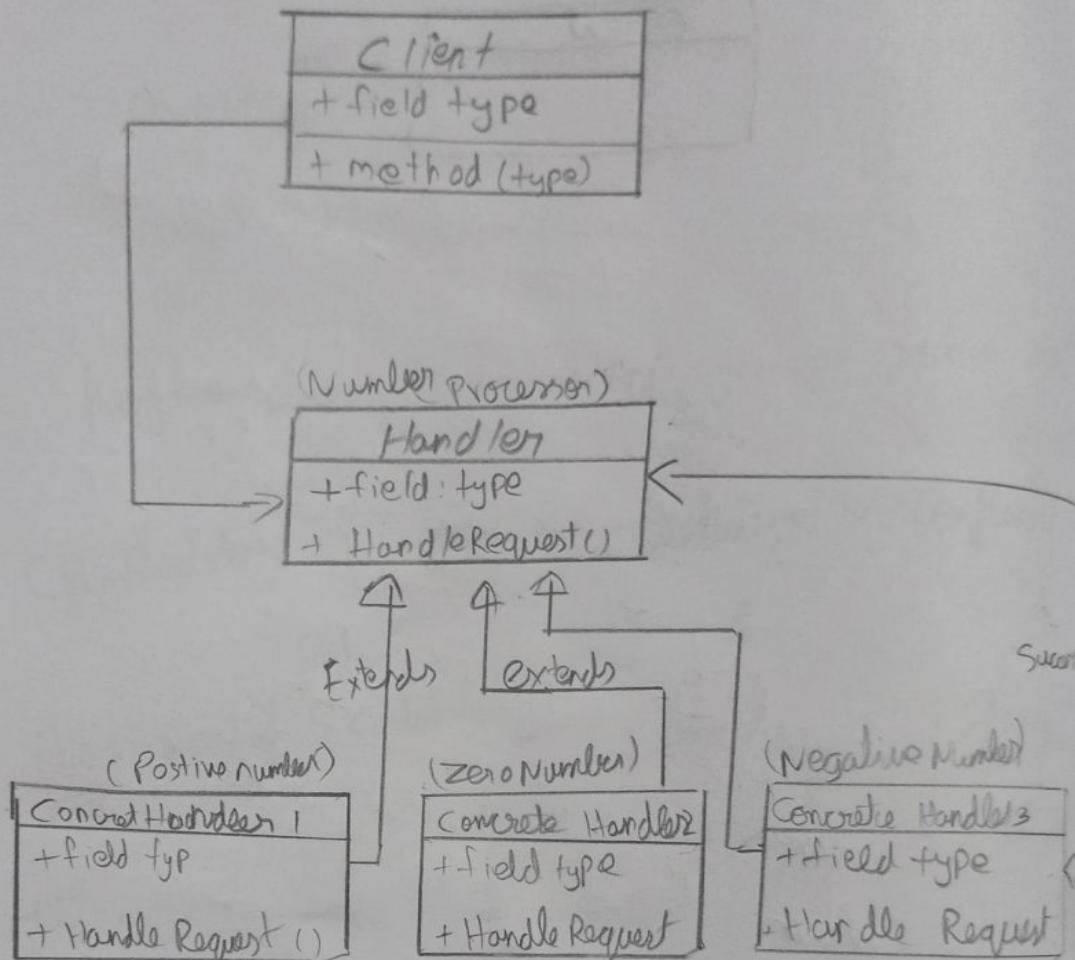
## **1. Class diagram and description of Chain of responsibilities design Pattern**

Chain of Responsibility is a behavioral design pattern that lets you pass requests along a chain of handlers. Upon receiving a request, each handler decides either to process the request or to pass it to the next handler in the chain. Chain of responsibility pattern is used to achieve loose coupling in software design.

Used in Scenarios where

- When you want to decouple a request's sender and receiver
- Multiple objects, determined at runtime, are candidates to handle a request
- When you don't want to specify handlers explicitly in your code
- When you want to issue a request to one of several objects without specifying the receiver explicitly.

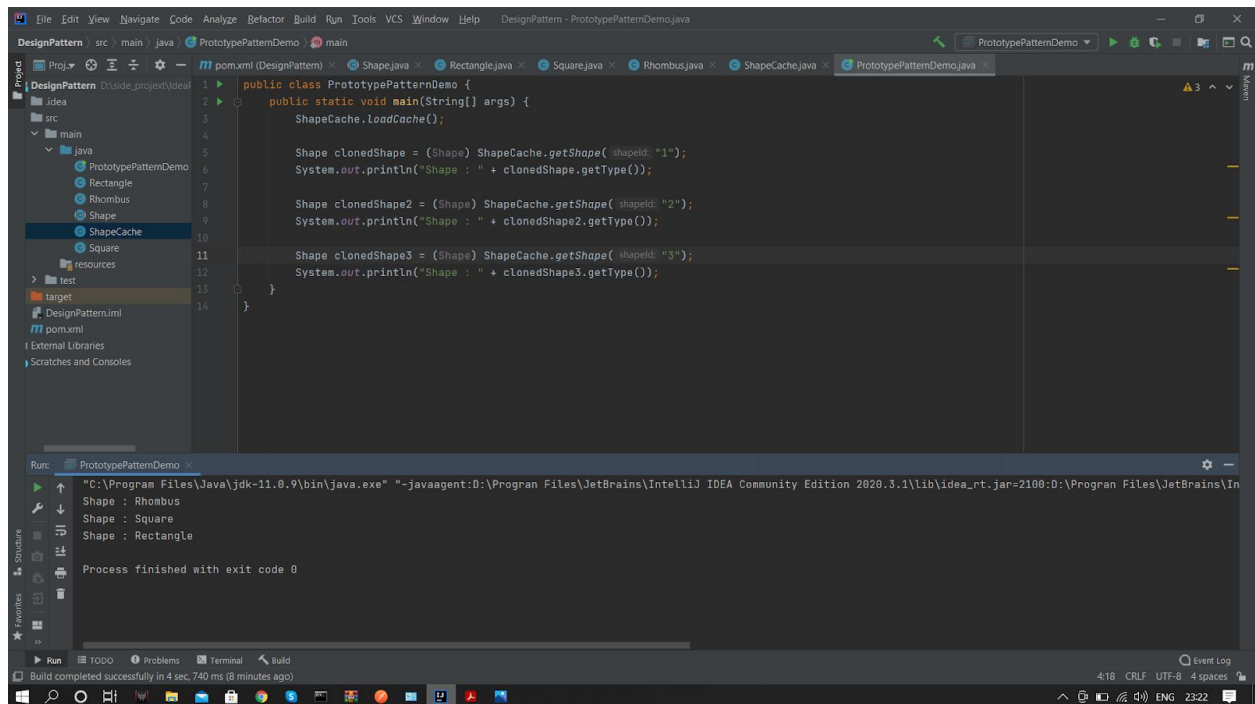
# Chain of Responsibility



Chain

Positive Numbers  
↓  
Zero Numbers  
↓  
Negative Numbers

## 2. Example(Java Classes) for Prototype(Creational)



The screenshot displays an IDE window titled "DesignPattern - PrototypePatternDemo.java". The main editor shows the following Java code:

```
1 public class PrototypePatternDemo {  
2     public static void main(String[] args) {  
3         ShapeCache.loadCache();  
4  
5         Shape clonedShape = (Shape) ShapeCache.getShape("1");  
6         System.out.println("Shape : " + clonedShape.getType());  
7  
8         Shape clonedShape2 = (Shape) ShapeCache.getShape("2");  
9         System.out.println("Shape : " + clonedShape2.getType());  
10  
11        Shape clonedShape3 = (Shape) ShapeCache.getShape("3");  
12        System.out.println("Shape : " + clonedShape3.getType());  
13    }  
14 }
```

The left sidebar shows the project structure with the following files:

- DesignPattern
  - src
    - main
      - java
        - PrototypePatternDemo
        - Rectangle
        - Rhombus
        - Shape
        - ShapeCache
        - Square
      - resources
    - test
    - target
  - DesignPattern.xml
  - pom.xml
  - External Libraries
  - Scratches and Consoles

The bottom panel shows the Run output for "PrototypePatternDemo":

```
"C:\Program Files\Java\jdk-11.0.9\bin\java.exe" "-javaagent:D:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2020.3.1\lib\idea_rt.jar=2100:D:\Program Files\JetBrains\In  
Shape : Rhombus  
Shape : Square  
Shape : Rectangle  
Process finished with exit code 0
```

The status bar at the bottom indicates "Build completed successfully in 4 sec, 740 ms (8 minutes ago)" and "4:18 CRLF UTF-8 4 spaces".