Design Principles and Patterns

Done by Ravi Theja Kolluru

Email ID: ravi.kolluru@accolitedigital.com

Assignment

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

2. Example(Java Classes) for Prototype(Creational) or Flyweight(Structural)

ANSWER

Chain of Responsibilities

- ✓ Chain of responsibility is behavioral pattern that creates a chain of receiver objects for a receiver.
- ✓ It decouples sender and receiver of a request.
- ✓ It is a loosely coupled design pattern.
- ✓ This possible where many objects can process the request.
- ✓ Handler is the one which dispathes the given request to other handlers.
- ✓ If the handler is unable to handle the request, then it is passed onto another.
- ✓ Reference to the main handler is known but not with others.
- ✓ Concrete handlers make sure that they are chained in a sequential order.
- ✓ Can be used to isolate request's sender and receiver.
- ✓ To avoid specifying handlers explicitly in code.

Advantages

- Increased request processing.
- Flexibility of assigning to Handlers.
- Reducing the coupling degree.
- Objects need not know chain structure.
- Chain structure can be hidden.

Disadvantages

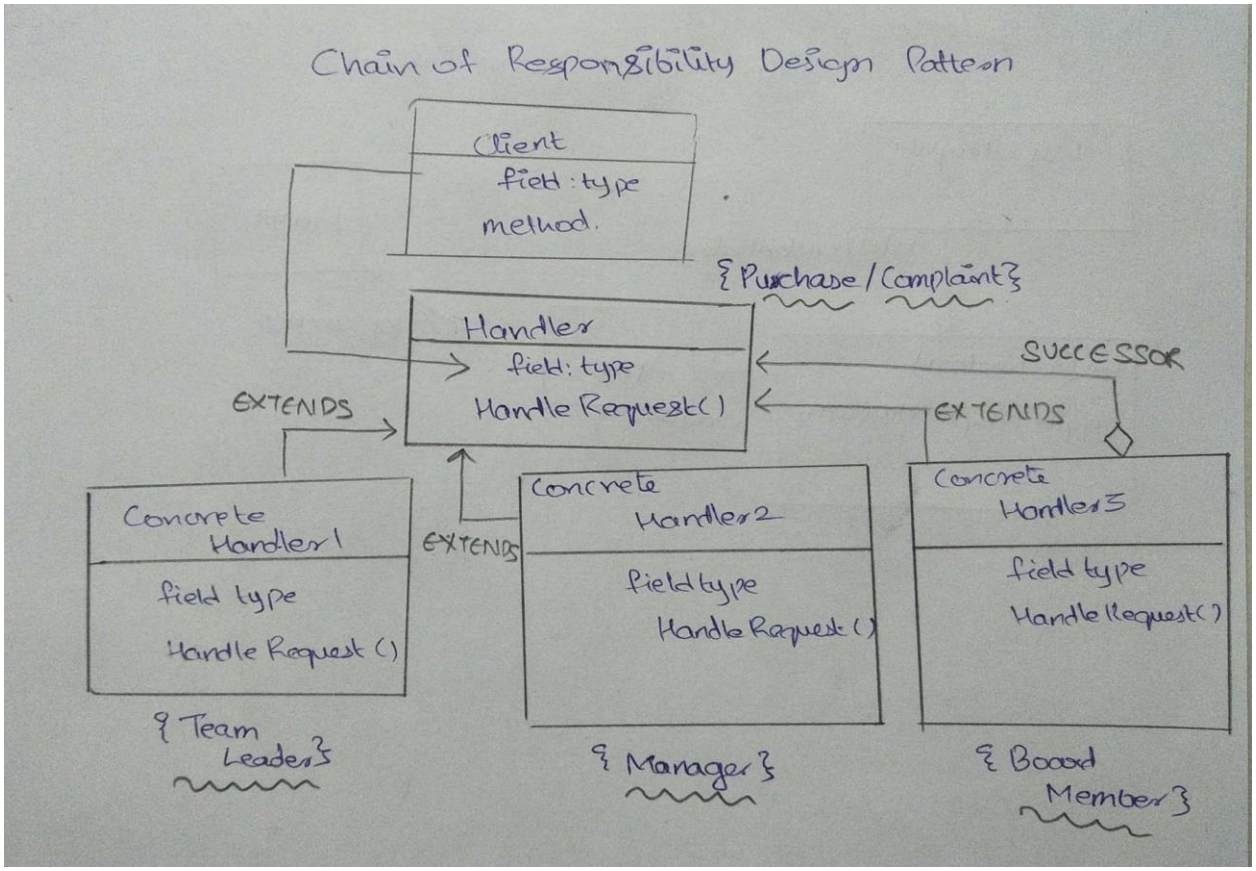
- Difficult to debug.
- Difficult to predict the direction of operation.

Example

If the company would like to make a purchase, if it is a smaller amount the team leader can be authorized to do so, but as the amount increases team leader does not perform the operation and chains it to manager, if it doesn't get handled over there it goes on to directors and throw an exception if not handled anywhere.

Another example of complain system can be taken for the same hierarchy.

ravi.kolluru@accolitedigital.com Accolite Software India Pvt. Ltd.



Creational Design Pattern

- Prototype design pattern is a creational design pattern.
- This hides the complexity of new instances created.
- It is used for replicating.
- It renders the components multiple times by changing the attributes and properties, like making use of template.
- Cloning operation takes place. Override clone interface.
- It has to avoid when class hierarchy is not independent and complex.
- The original object still remains intact.

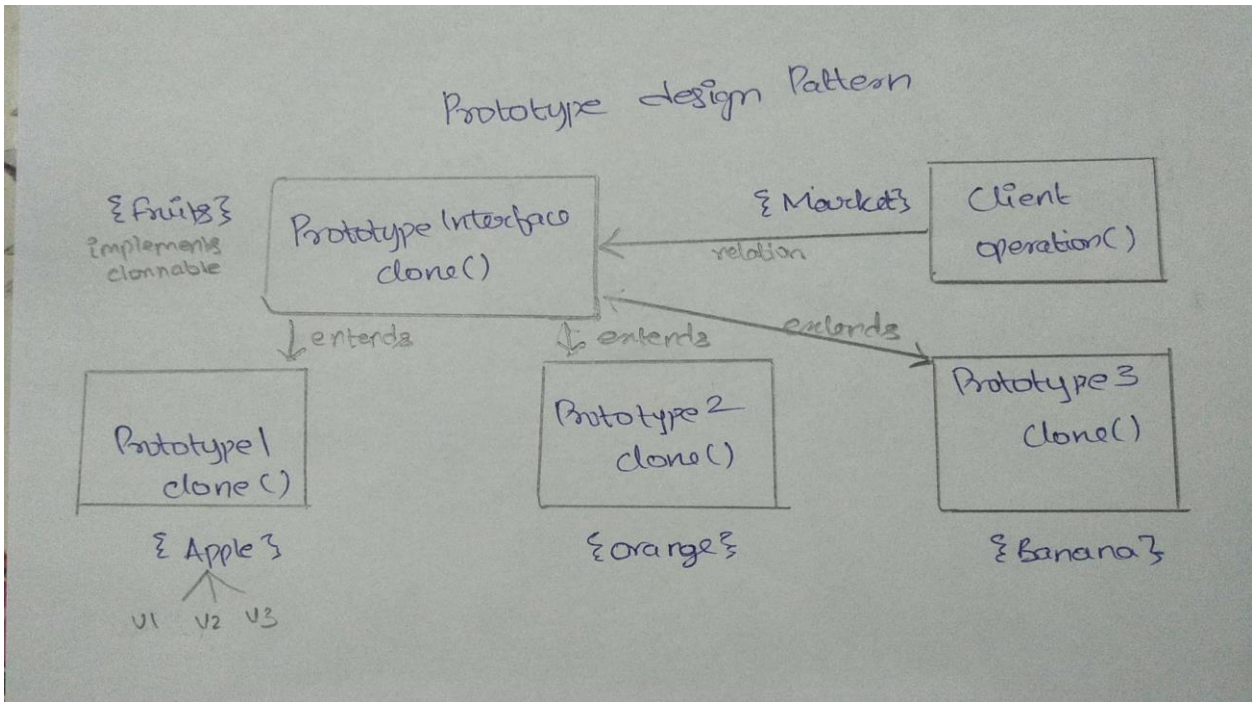
ravi.kolluru@accolitedigital.com Accolite Software India Pvt. Ltd.

Advantages

- ✓ Saves time and resources
- ✓ Runtime allocations
- ✓ Dynamic
- ✓ Hides inner classes (similar to encapsulation)

Disadvantages

Implementing the subclasses while cloning could turn out complex as per the requirements.




```
Orange added to basket
Apple added to basket
Apple added to basket
```

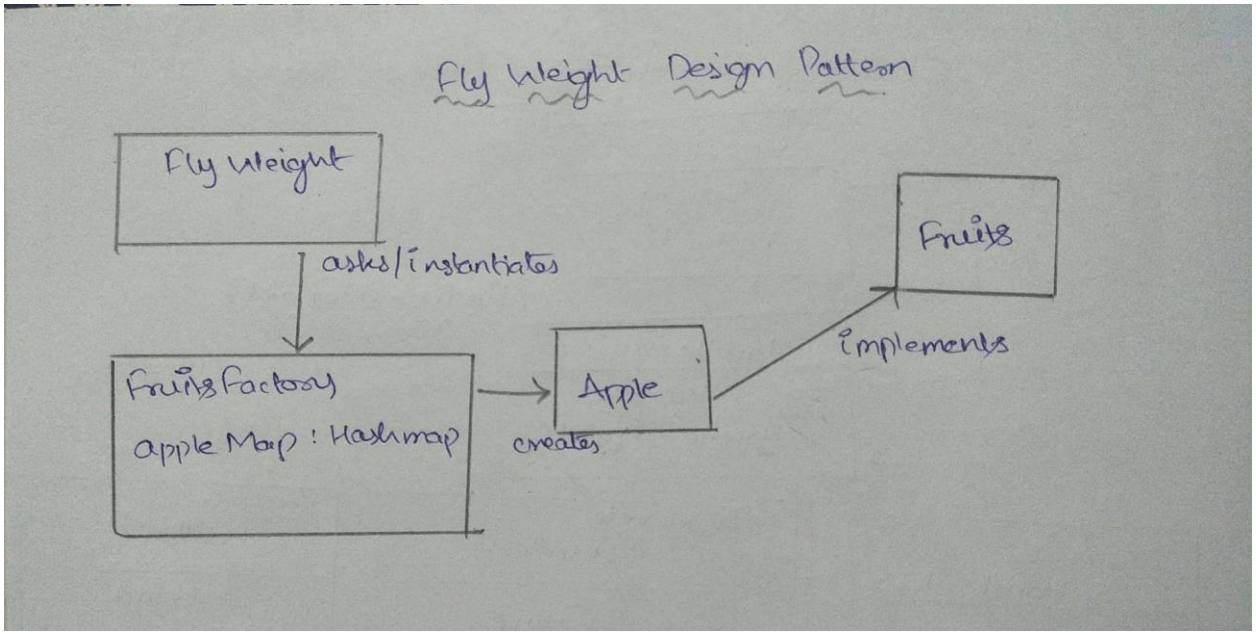

ravi.kolluru@accolitedigital.com Accolite Software India Pvt. Ltd.

Flyweight Design Pattern

- Flyweight design pattern are structural design patterns that reduce the creation of number of objects which provide the same functionality.
- They are immutable.
- An object is shared where ever required.
- It uses a hash map to store reference to object.
- An object is created for the first time when such functionality is occuring for first time but next time it recalls the object that was created in the past.
- Intrinsic state are the things that are constant and are stored in memory.
- Extrinsic states are the things that are not constant and are viewed as replicas (dynamic) and not stored in memory.

Advantage

Reduces memory usage




```
Adding Shimla apple to map
Adding Chile apple to map
Adding Green-mael apple to map
```

***

ravi.kolluru@accolitedigital.com Accolite Software India Pvt. Ltd.