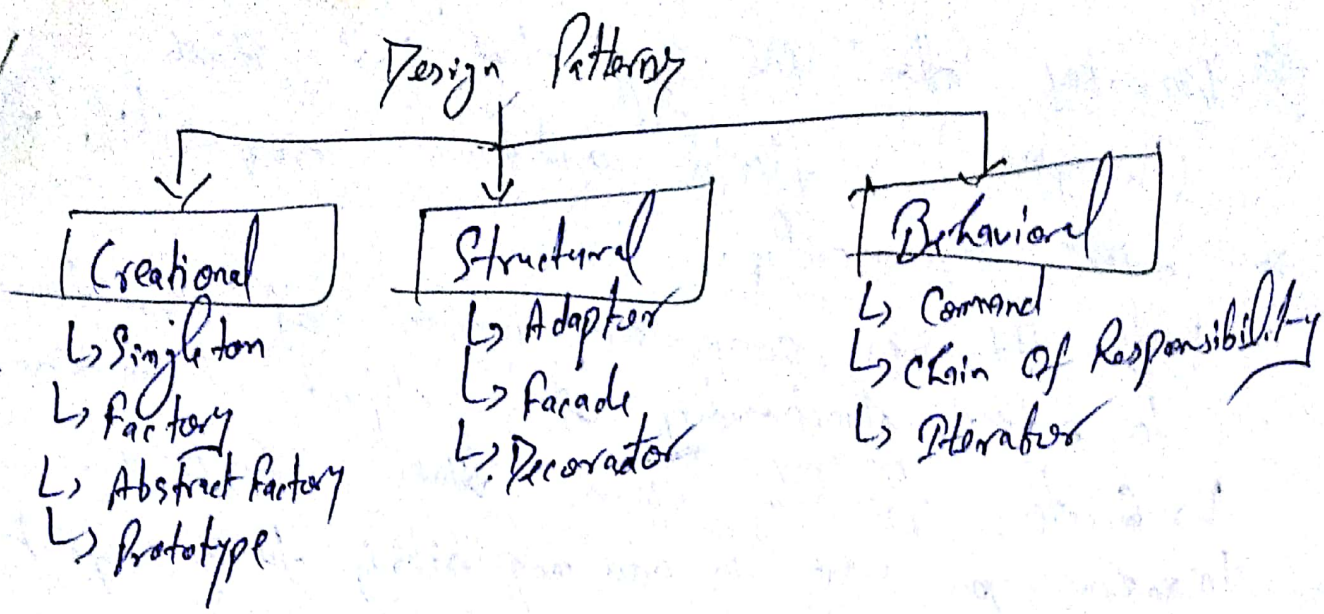
1



Creational DP: These are concerned with the way of creating objects. These dps are used when a decision must be made at the time of instantiation of a class.

Structural DP: These concern classes and object composition. Concept of inheritance is used to compose interfaces and define ways to compose objects to obtain new functionalities.

Behavioral DP: These concern with communication b/w objects.

J2EE DP: These are concerned with presentation tier.

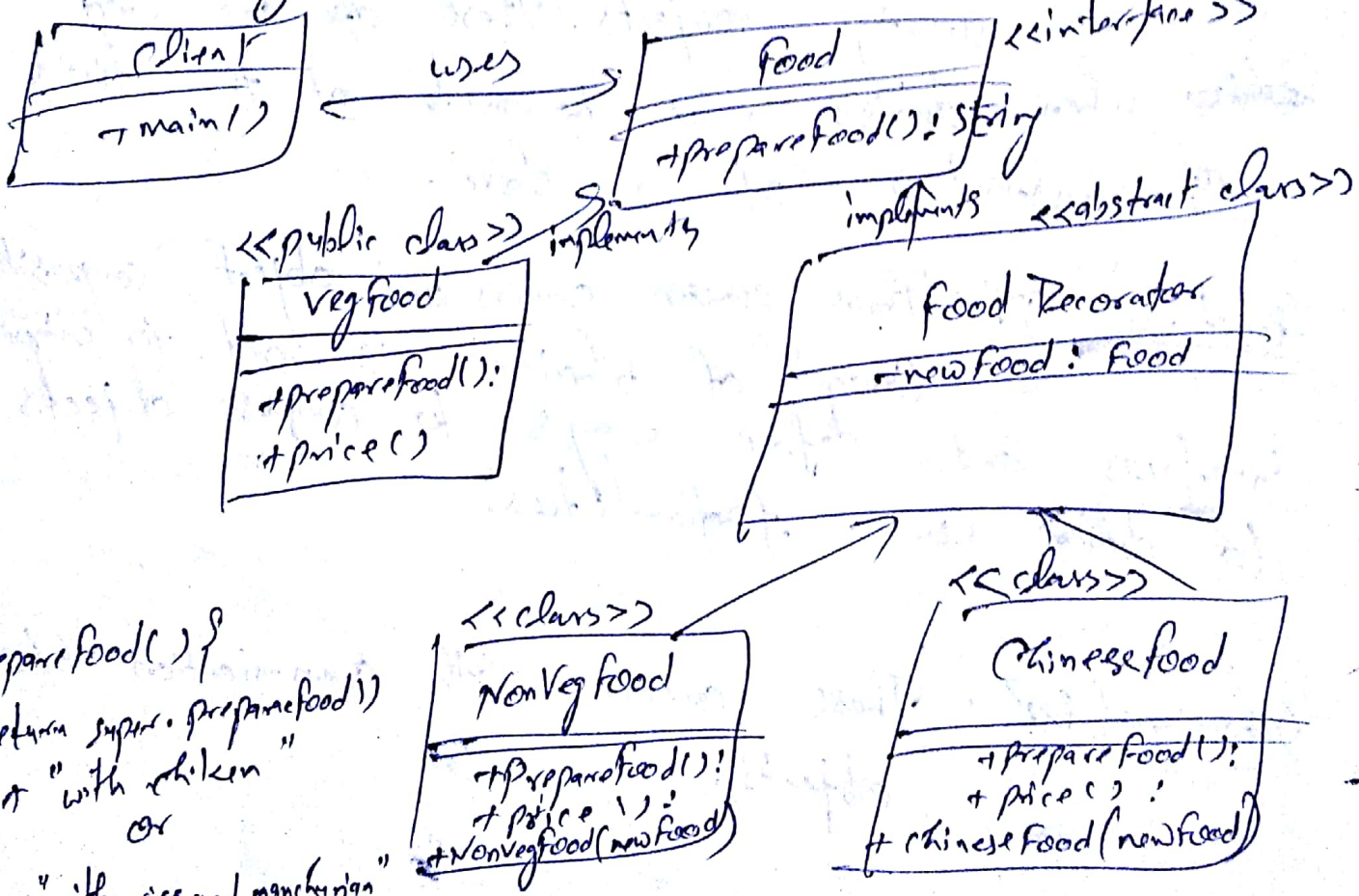
③ Decorator Pattern: It says that "just attach a flexible additional responsibility (Wrapper) to an object dynamically".

↳ Or It uses composition instead of inheritance to extend functionality of an object at run time.

↳ Greater flexibility than Static inheritance.

↳ When you want to add responsibility to an object that you may want to use in future.

↳ Extending functionality is not practical.




```
prepareFood(){
return super.prepareFood()
+ "with chicken"
      or
+ "with rice and manchurian"
}
```

```
Food f1 = new NonVegFood((Food)new VegFood());
Food f2 = new ChineseFood((Food)new VegFood());
```

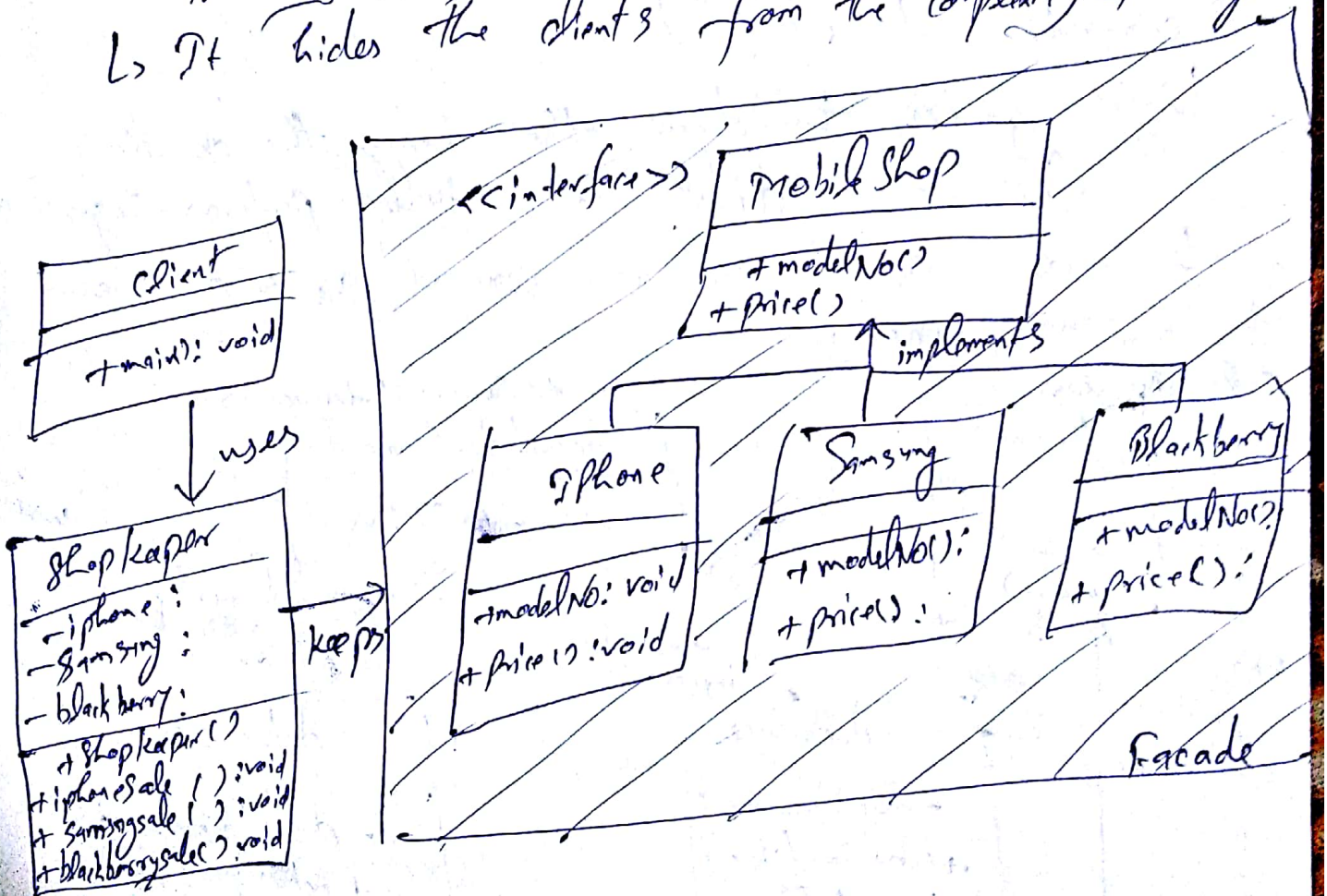
O/P: Veg food with chicken
Veg food with rice and manchurian

Facade Pattern: It says that "just provide an unified and simplified interface to a set of interface in a subsystem, therefore, it hides the complexity of the subsystem from the client."

↳ Or It describes a higher level interface that makes the subsystem easier to use.

↳ Practically, every Abstract factory is a type of Facade.

↳ It hides the clients from the complexity of subsystem.



Design Pattern :

(1) Command Pattern.

It says that "encapsulate a request under an object as a command and pass it to the invoker object. Invoker object looks for the appropriate object which can handle this command and pass the command to the corresponding object and that object execute the command.

- It separates the object that invokes the operation from the object that actually performs the operation.
- It made easy to add new command as existing classes remain unchanged.