Design patterns:

* Creational
  + Builder
  + Factories
    - Abstract Method
    - Factory method
  + Prototype
  + Singleton
* Structural
  + Adapter
  + Bridge
  + Composite
  + Decorator
  + Façade
  + Flyweight
  + Proxy
* Behavioral
  + Chain of Responsibility
  + Command
  + Interpreter
  + Iterator
  + Mediator
  + Memento
  + Observer
  + State
  + Strategy
  + Template Method
  + Visitor

SOLID principles:

Single Responsibility Principle – a class should have a single reason to change.

Open close- your app should be open for inheritance, but closed for modification

Liskov Substitution – any function that takes as argument a Base class object, should work perfectly fine with a derived class object.

Interface segregation :