Design Pattern:

There are three type of design pattern:

1. Creational Design Pattern

* Factory Method Pattern:

A Factory Pattern or Factory Method Pattern says that just **define an interface or abstract class for creating an object but let the subclasses decide which class to instantiate.** In other words, subclasses are responsible to create the instance of the class.

The Factory Method Pattern is also known as **Virtual Constructor.**

* **Abstract Factory Pattern:**

Abstract Factory Pattern says that just **define an interface or abstract class for creating families of related (or dependent) objects but without specifying their concrete sub-classes.**That means Abstract Factory lets a class returns a factory of classes. So, this is the reason that Abstract Factory Pattern is one level higher than the Factory Pattern.

An Abstract Factory Pattern is also known as **Kit.**

* **Singleton Design Pattern:**

Singleton Pattern says that just**"define a class that has only one instance and provides a global point of access to it".**

To create the singleton class, we need to have static member of class, private constructor and static factory method.

If singleton class is Serializable, you can serialize the singleton instance. Once it is serialized, you can deserialize it but it will not return the singleton object.

To resolve this issue, you need to override the **readResolve() method** that enforces the singleton. It is called just after the object is deserialized. It returns the singleton object.

* **Prototype Pattern:**

Prototype Pattern says that **cloning of an existing object instead of creating new one and can also be customized as per the requirement**.

* **Builder Design Pattern:**

Builder Pattern says that **"construct a complex object from simple objects using step-by-step approach"**

* **Object Pool Pattern:**

Object Pool Pattern says that **" to reuse the object that are expensive to create".**

Basically, an Object pool is a container which contains a specified amount of objects. When an object is taken from the pool, it is not available in the pool until it is put back.**Objects in the pool have a lifecycle: creation, validation and destroy.**

1. Structural Design Pattern:

**Structural design patterns** are concerned with how classes and objects can be composed, to form larger structures.

The structural design patterns **simplifies the structure by identifying the relationships**.

These patterns focus on, how the classes inherit from each other and how they are

 composed from other classes.

There are 7 types of Structural design pattern: