OBJECT ORIENTED ANALYSIS & DESIGN
DATA STRUCTURES & ALGORITHMS

Creational Patterns

Factory Method

Definition

Defines an interface for creating an object, but let subclasses decide which class to instantiate. Factory Method lets a class defer instantiation to subclasses

Problem & Context

If an object needs to know the selection criteria to instantiate an appropriate class, this results in a high degree of coupling. Whenever the selection criteria change, every object that uses the class must be changed correspondingly

Solution

Encapsulate the functionality required to select and instantiate the appropriate class. One way to do this is to create an abstract class or an interface that declares the factory method. Different subclasses can then implement the method in its entirety

Abstract Factory

Definition

Provides an interface for creating families of related or dependent objects without specifying their concrete classes

Problem & Context

Useful when an object wants to create an instance of a suite of related and dependent classes without having to know which specific concrete class is instantiated. In its absence, the required implementation needs to be present wherever such an instance is created

Solution

Provide the necessary interface for creating instances. Different concrete factories implement this interface. In Java, it is an abstract class with its concrete subclasses as factories. Each factory is responsible for creating and providing access to the objects

Builder

Definition

Separates the construction of a complex object from its representation so that the same construction process can create different representations

Problem & Context

Object construction details are kept within the object as part of its constructor. This may not be effective when the object being created is complex and the object creation process produces different representations of the object. The object can become bulky (construction bloat) and less modular

Solution

Move the construction logic out of the object class to separate classes referred to as builder classes. A dedicated object referred to as a Director, is responsible for invoking different builder methods required for the construction of the final object. Different client objects can make use of the Director object to create the required object

Prototype

Definition

Specifies the kind of objects to create using a prototypical instance and creates new objects by copying this prototype

Problem & Context

When clients need to create a set of objects that are cost prohibitive and alike or differ only in terms of their state, create one object upfront and designate it as a prototype object or simply make a copy of the prototype object

Solution

Provide a way for clients to create a copy of the prototype object. By default, all Java objects inherit the built in clone() method that creates a clone of the original object

Singleton

Definition

Ensures a class has only one instance and provide a global point of access to it

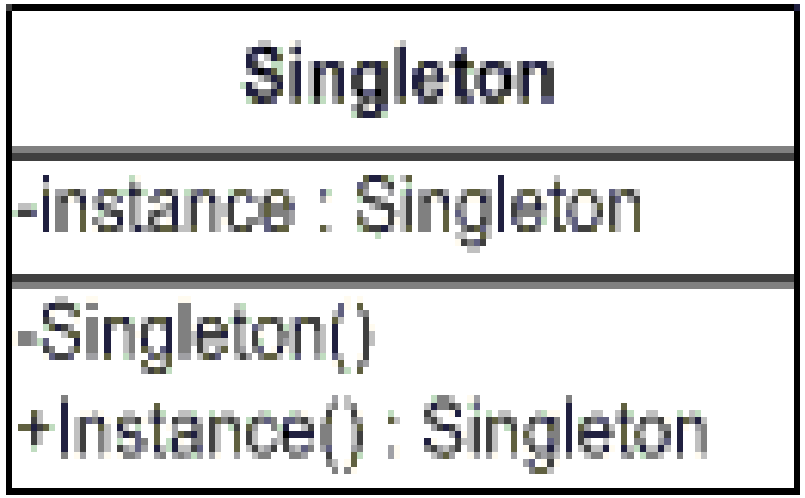
Problem & Context

Sometimes there may be a need to have one and only one instance of a given class during the lifetime of an application. This may be due to necessity or, more often, because only a single instance of the class is sufficient

Solution

Create a class with a method that creates a new instance of the object if one does not exist. If one does exist, it returns a reference to the object that already exists. To make sure that the object cannot be instantiated any other way, the constructor is made either private or protected

Singleton (Diagram)



Thank You!