DESIGN PATTERNS:

CREATIONAL DESIGN PATTERNS:

This design patterns are all about class instantiation. These patterns can be further divided into class-creation patterns and object-creational patterns.

1. SINGLETON – a class of which only single instance can exist
2. FACTORY – creates an instance of several derived classes.

STRUCTURAL DESIGN PATTERNS

This design patterns are all about class and object composition. Structural class-creation use inheritance to compose interfaces. Structural object-patterns define ways to compose objects to obtain new functionality.

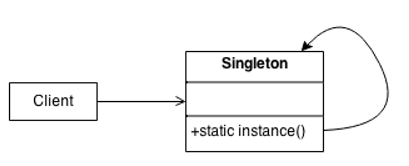
1. FAÇADE – a single class that represents an entire subsystem.
2. PROXY – an object representing another object.

SINGLETON DESIGN PATTERN

Intent:

Ensure a class has only one instance, and provide a global point of access to it.

STRUCTURE:



EXAMPLE IN JAVA:

class Singleton

{

private Singleton() { }

private static Singleton instance = new Singleton();

public static Singleton GetInstance()

{

return instance;

}

}

public class App {

public static void main(String[] args) {

Singleton s1 = Singleton.GetInstance();

Singleton s2 = Singleton.GetInstance();

System.out.println(s1==s2); //--> true

System.out.println(s1); //--> Singleton@6a6824be

System.out.println(s2); //--> Singleton@6a6824be

}

}

FAÇADE DESIGN PATTERN

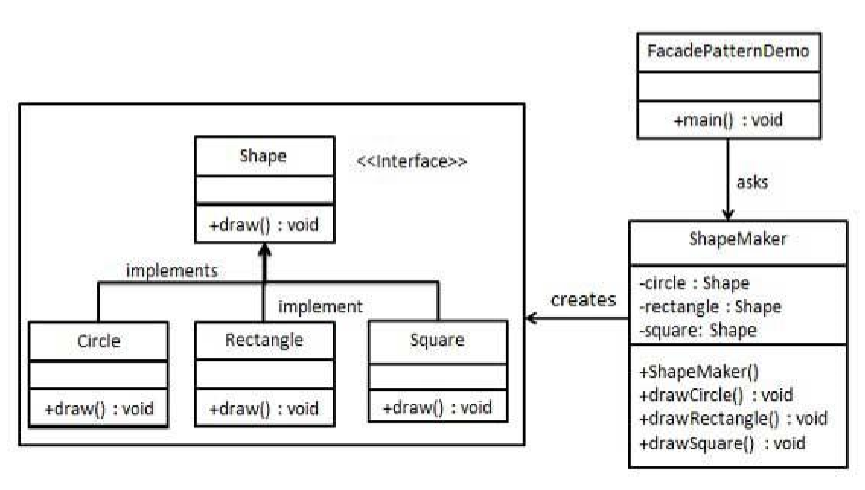
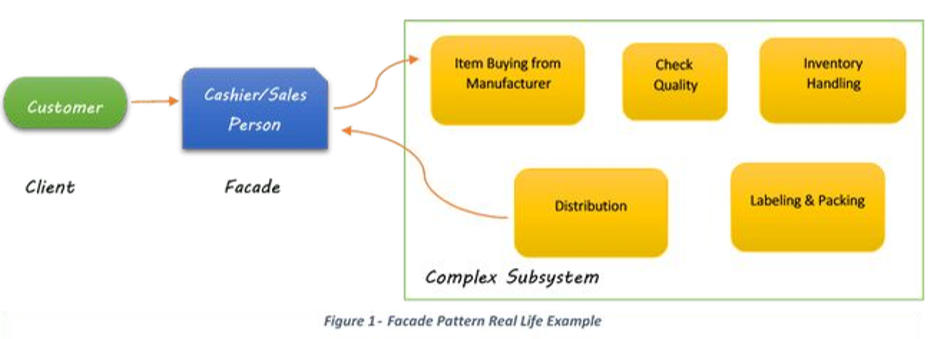
Intent:

Provide a unified interface to a set of interfaces in a subsystem. Façade defines a higher-level interface that makes the subsystem easier to use.

* Removes a complexity of code by introducing an additional layer over a set of complex components. It creates many other actions behind the scenes.

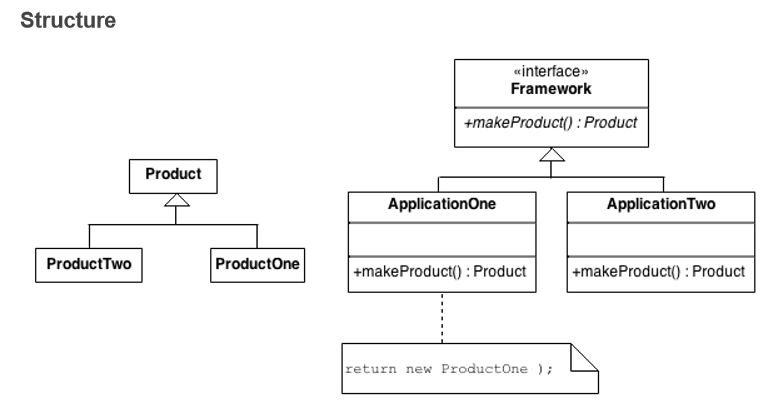
Example:

* ATM machine



FACTORY METHOD DESIGN PATTERN

Intent:

* Define an interface for creating an object, but let subclasses decide which class to instantiate. Factory method lets a class defer instantiation to subclasses.
* Defining a “virtual” constructor.
* The new operator considered harmful.

PROXY DESIGN PATTERN

Intent:

Proxy is a structural design pattern that lets you provide a substitute or placeholder for another object. A proxy controls access to the original object, allowing you to preform something either before or after the request gets through to the original object.

