

CSE443

HW2

PART 1

- a) No it doesn't. Because Object's clone method checks that the interface is implemented otherwise it will throw an exception. A call to clone() method will result in CloneNotSupportedException, if that class doesn't implement Cloneable interface.
- b) To prevent the cloning of a Singleton object we need override clone() method and throw an exception CloneNotSupportedException or return the same instance from clone method. So whenever user will try to create clone of singleton object, it will throw exception and hence the class remains singleton.
- c) For Question 1:

A class implements the Cloneable interface to indicate to the Object.clone() method that it is legal for that method to make a field-for-field copy of instances of that class. It will create clone of a singleton object, then it will create a copy that is there are two instances of a singleton class, hence the class is no more singleton.

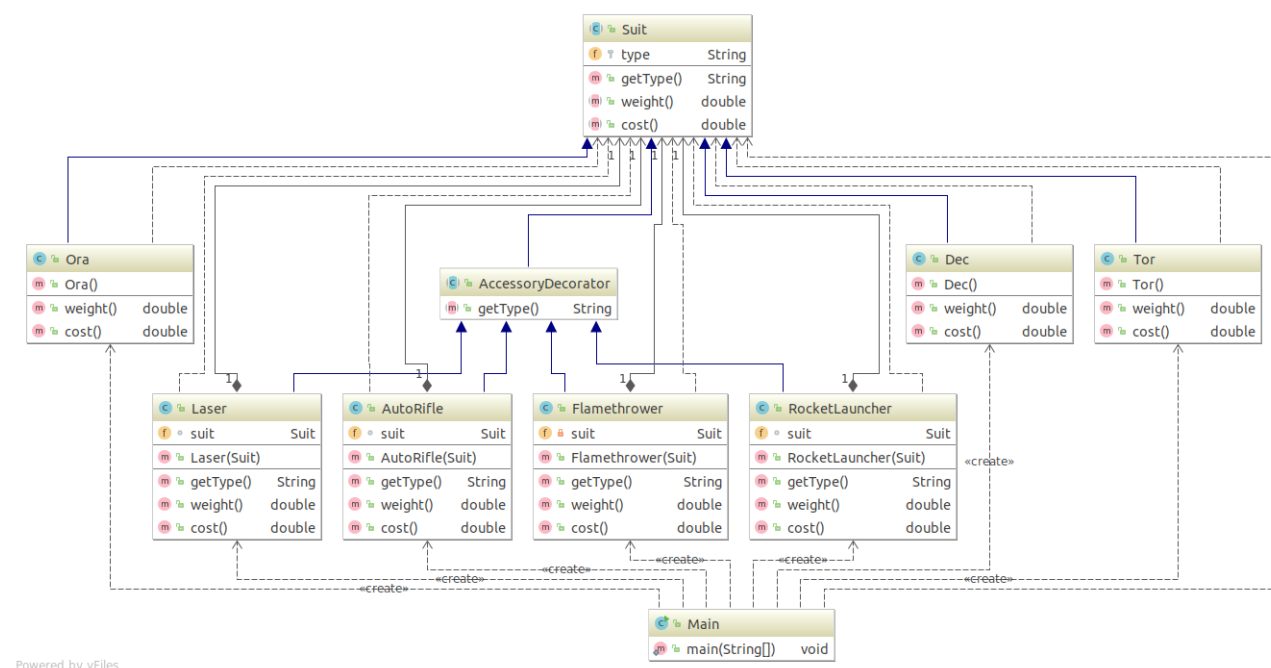
PART 2

In this part, I implemented program using Decorator Design pattern. Suit class is the main class for decorator. All suits and all accessories extended from Suit abstract class.

Running Example

I use Dec, Ora, Tor suits for testing. For each suits, I added some additional accessories.

BASE PRICES



ORA WITH ACCESSORIES

Ora,Flamethrower: 1550.0k TL, 32.0 kg
Ora,Flamethrower,AutoRifle: 1580.0k TL, 33.5 kg
Ora,Flamethrower,AutoRifle,RocketLauncher: 1730.0k TL, 41.0 kg
Ora,Flamethrower,AutoRifle,RocketLauncher,Laser: 1930.0k TL, 46.5 kg

UML Diagram

PART 3

I use Factory Method to design this program.

Test Code

```
TEST TPX100
-----
TPX100
Skeleton: Aluminum alloy
Engine: Single jet engine
Seats: 50
-----

TEST TPX200
-----
TPX200
Skeleton: Nickel alloy
Engine: Twin jet engine
Seats: 100
-----

TEST TPX300
-----
TPX300
Skeleton: Titanium alloy
Engine: Quadro jet engine
Seats: 250
-----
```

Test Output

```
PlaneFactory factory = new PlaneFactory();

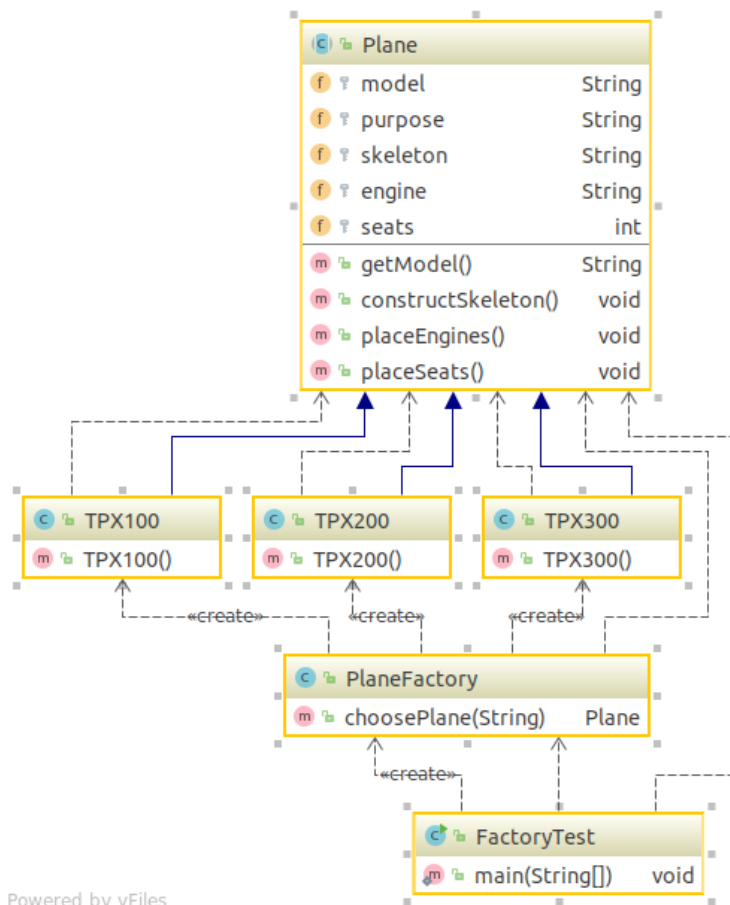
Plane plane1 = factory.choosePlane( model: "TPX100");
Plane plane2 = factory.choosePlane( model: "TPX200");
Plane plane3 = factory.choosePlane( model: "TPX300");

System.out.println("TEST TPX100");
System.out.println("-----");
System.out.println(plane1.getModel());
plane1.constructSkeleton();
plane1.placeEngines();
plane1.placeSeats();
System.out.println("-----\n");

System.out.println("TEST TPX200");
System.out.println("-----");
System.out.println(plane2.getModel());
plane2.constructSkeleton();
plane2.placeEngines();
plane2.placeSeats();
System.out.println("-----\n");

System.out.println("TEST TPX300");
System.out.println("-----");
System.out.println(plane3.getModel());
plane3.constructSkeleton();
plane3.placeEngines();
plane3.placeSeats();
System.out.println("-----");
```

UML Diagram



Powered by yFiles

PART 4

UML Diagram

In zip file, [uml_part4.png](#).