

Assignment - 6

WAP to implement Prototype Design Pattern

What is Prototype Design Pattern?

- Prototype Pattern says that **cloning of an existing object instead of creating new one and can also be customized as per the requirement.**
- This pattern should be followed, if the cost of creating a new object is expensive and resource intensive.

Advantages of Prototype Pattern

1. It reduces the need of sub-classing.
2. It hides complexities of creating objects.

Prototype Pattern use for

1. When the classes are instantiated at runtime.
2. When the cost of creating an object is expensive or complicated.

Code for the same: -

```
package prototype;  
import java.util.ArrayList;  
import java.util.List;  
  
class Vehicle implements Cloneable {  
    private List<String> vehicleList;
```

//Implementing the clonable interface for the cloning

```
public Vehicle() {
    this.vehicleList = new ArrayList<String>();
}

public Vehicle(List<String> list) {
    this.vehicleList = list;
}

public void insertData() {
    vehicleList.add("Honda amaze");
    vehicleList.add("Audi A4");
    vehicleList.add("Hyundai Creta");
    vehicleList.add("Maruti Baleno");
    vehicleList.add("Renault Duster");
}

public List<String> getVehicleList() {
    return this.vehicleList;
}

@Override
public Object clone() throws CloneNotSupportedException {
    List<String> tempList = new ArrayList<String>();

    for(String s : this.getVehicleList()) {
        tempList.add(s);
    }

    return new Vehicle(tempList);
}

public class PrototypePatternExample {

    public static void main(String[] args) throws CloneNotSupportedException {
        Vehicle a = new Vehicle();
        a.insertData();

        Vehicle b = (Vehicle) a.clone();
        //Getting the vehicle list
        List<String>list=b.getVehicleList();
        //Adding the vehicle
        list.add("Honda new Amaze");
    }
}
```

```
System.out.println(a.getVehicleList());  
System.out.println(list);
```

//Removing the vehicle

```
a.getVehicleList().remove("Audi A4");  
System.out.println(list);  
System.out.println(a.getVehicleList());  
}  
}
```

Output: -

```
PS D:\Design-And-Pattern> cd "d:\Design-And-Pattern\" ; if ($?) { javac PrototypePatternExample.java } ;  
[Honda amaze, Audi A4, Hyundai Creta, Maruti Baleno, Renault Duster]  
[Honda amaze, Audi A4, Hyundai Creta, Maruti Baleno, Renault Duster, Honda new Amaze]  
[Honda amaze, Hyundai Creta, Maruti Baleno, Renault Duster, Honda new Amaze]  
[Honda amaze, Audi A4, Hyundai Creta, Maruti Baleno, Renault Duster]  
PS D:\Design-And-Pattern>
```