

DEPARTEMENT MATHÉMATIQUES ET INFORMATIQUE

Design Pattern

Filière :
« Génie du Logiciel et des Systèmes Informatiques Distribués »
GLSID

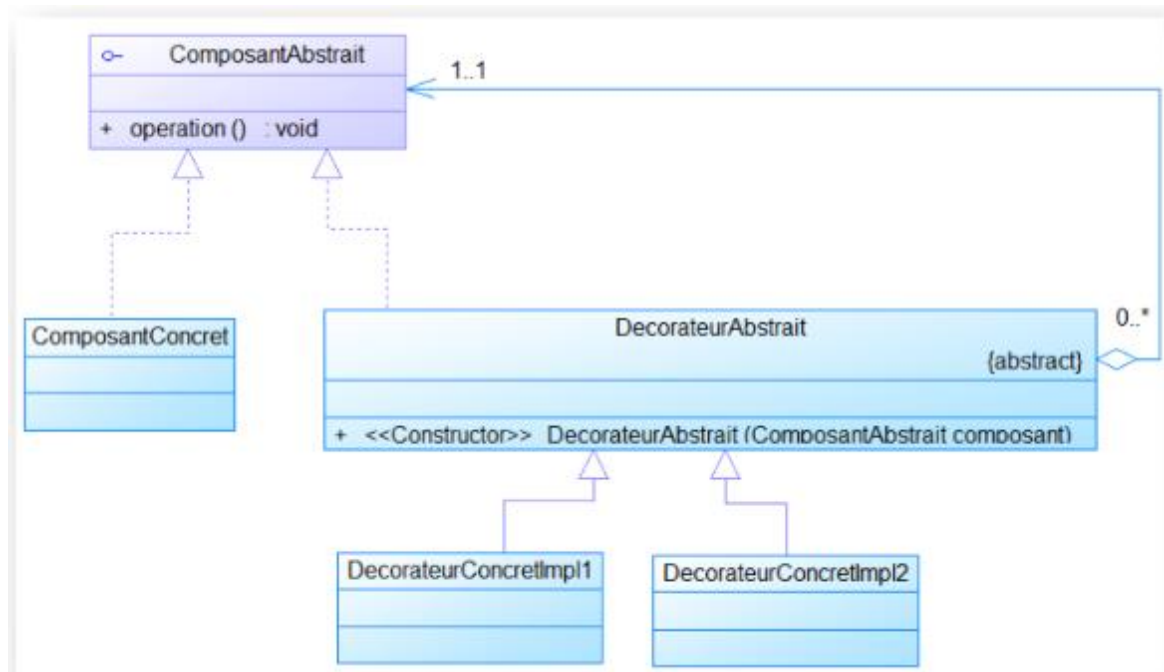
Pattern Decorator

Réalisé par :

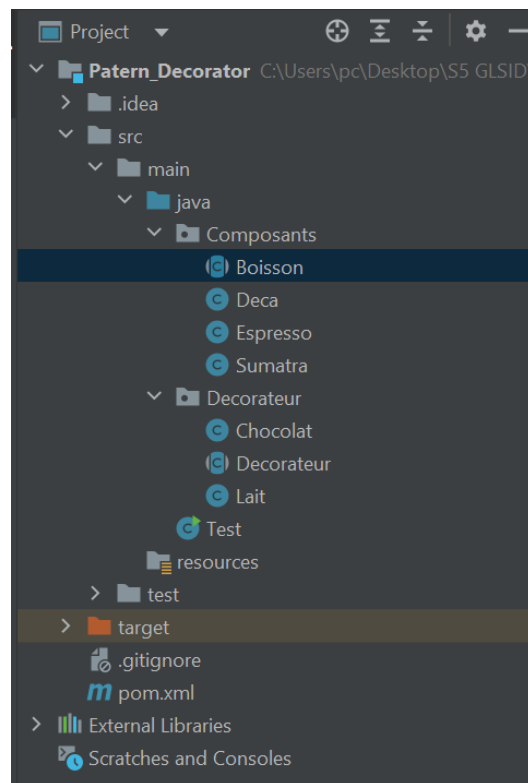
Najat ES-SAYYAD

Année Universitaire : 2023-2024

Pattern Decorator



Structure de projet :



Package Composants

Classe abstraite Boisson :

```
1 package Composants;
2
3 14 usages 6 inheritors
4 public abstract class Boisson {
5     4 usages
6     protected String description ;
7     8 usages 3 overrides
8     public String getDescription(){
9         return description;
10    }
11
12    3 usages 5 implementations
13    public abstract double cout();
14 }
```

Classe Deca :

```
1 package Composants;
2
3 public class Deca extends Boisson{
4     public Deca() {
5         description="Deca";
6     }
7     @Override
8     public double cout() { return 8; }
9 }
10
11
```

Classe Espresso :

```
1 package Composants;
2
3 public class Espresso extends Boisson{
4     public Espresso() {
5         description="Espresso";
6     }
7     @Override
8     public double cout() { return 12; }
9 }
10
11
```

Classe Sumatra :

```
1 package Composants;
2
3 public class Sumatra extends Boisson{
4     public Sumatra() {
5         description="Sumatra";
6     }
7     @Override
8     public double cout() { return 6; }
9 }
10
11
```

Package Decorateur

Classe abstraite Decorateur :

```
1 package Decorateur;
2
3 import Composants.Boisson;
4
5 2 📄 2 inheritors
6 public abstract class Decorateur extends Boisson {
7     5 usages
8     protected Boisson boisson;
9     2 usages
10    public Decorateur(Boisson boisson) {
11        this.boisson=boisson ;
12    }
13    // pour forcer les autres classes qui hérite de cette
14    // classe de redéfinir la méthode getDescription
15    8 usages 2 implementations
16    public abstract String getDescription();
17 }
18
```

Classe Chocolat :

```
1 package Decorateur;
2
3 import Composants.Boisson;
4
5 4 usages
6 public class Chocolat extends Decorateur {
7     2 usages
8     public Chocolat(Boisson boisson) {
9         super(boisson);
10    }
11
12    8 usages
13    @Override
14    public String getDescription() {
15        return boisson.getDescription()+" Au chocolat";
16    }
17
18    8 usages
19    @Override
20    public double cout() {
21        return 1.2+ boisson.cout();
22    }
23 }
24
```

Classe Lait :

```
1 package Decorateur;  
2  
3 import Composants.Boisson;  
4  
5 public class Lait extends Decorateur {  
6  
7     2 usages  
8     public Lait(Boisson boisson) {  
9         super(boisson);  
10    }  
11  
12    8 usages  
13    @Override  
14    public String getDescription() {  
15        return boisson.getDescription()+" Au Lait";  
16    }  
17  
18    8 usages  
19    @Override  
20    public double cout() {  
21        return 2+ boisson.cout();  
22    }  
23 }
```

Test :

```
1 import ...  
2  
3 no usages  
4  
5 public class Test {  
6  
7     no usages  
8     public static void main(String[] args) {  
9         Boisson boisson;  
10        boisson=new Sumatra();  
11        System.out.println(boisson.getDescription());  
12        System.out.println(boisson.cout());  
13        System.out.println("-----");  
14        boisson=new Chocolat(boisson);  
15        System.out.println(boisson.getDescription());  
16        System.out.println(boisson.cout());  
17        System.out.println("-----");  
18        boisson=new Lait(boisson);  
19        System.out.println(boisson.getDescription());  
20        System.out.println(boisson.cout());  
21        System.out.println("\n*****\n");  
22  
23        Boisson boisson1;  
24        boisson1=new Espresso();  
25        System.out.println(boisson1.getDescription());  
26        System.out.println(boisson1.cout());  
27        System.out.println("-----");  
28    }  
29 }
```

```

28         boisson1=new Chocolat(boisson1);
29         System.out.println(boisson1.getDescription());
30         System.out.println(boisson1.cout());
31         System.out.println("-----");
32         boisson1=new Lait(boisson1);
33         System.out.println(boisson1.getDescription());
34         System.out.println(boisson1.cout());
35     }
36 }

```

Résultat :

```

"C:\Program Files\Java\jdk-17\bin\java.exe" "-javaagent:C:\Progr
Sumatra
6.0
-----
Sumatra Au chocolat
7.2
-----
Sumatra Au chocolat Au Lait
9.2

*****

Espresso
12.0
-----
Espresso Au chocolat
13.2
-----
Espresso Au chocolat Au Lait
15.2

Process finished with exit code 0

```