

Composite Pattern



Juan Carlos Castro Guevara
Michael Steven Pinzón Solano
Daniel Alejandro Olarte Ávila
Shelsy Natalia Rodríguez Barajas

Ciencias de la computación e inteligencia artificial
Patrones de Diseño de Software
Juan Sebastián Leyva Casas
Universidad Sergio Arboleda
2021-03

Problema:

Se requiere agrupar objetos como un solo objeto y permitir componer objetos en estructuras de árbol y luego trabajar con estas estructuras como si fueran objetos individuales.

Por ejemplo, tenemos dos tipos de objetos: hojas y compuestos. Un compuesto puede contener varias hojas así como cierto número de compuestos más pequeñas. Estas compuestos pequeños también pueden contener algunas hojas o incluso compuestos más pequeños, y así sucesivamente.

Código:

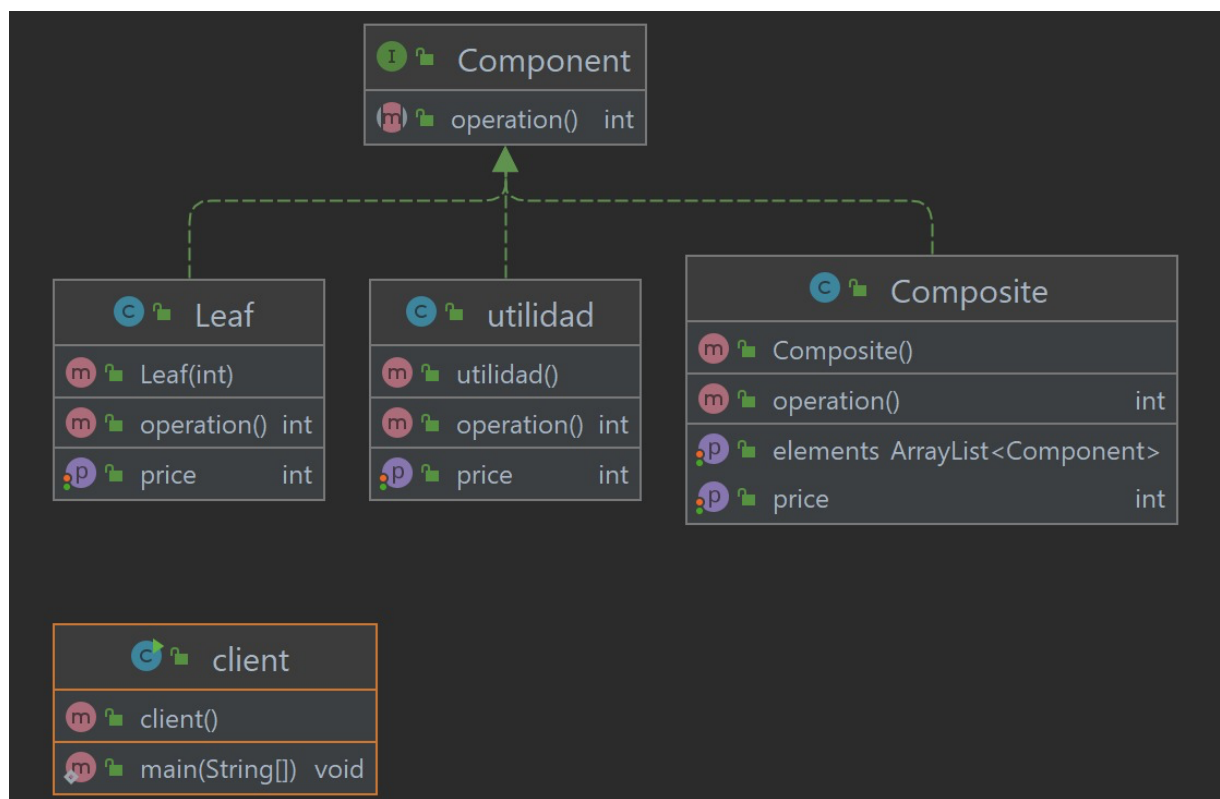
```
public class client {  
    public static void main(String[] args) {  
        Leaf camara = new Leaf( price: 300000);  
        Leaf pantalla = new Leaf( price: 150000);  
        Leaf bateria = new Leaf( price: 200000);  
        Leaf procesador = new Leaf( price: 450000);  
        Leaf botones = new Leaf( price: 10000);  
        Leaf microfono = new Leaf( price: 60000);  
  
        Composite celular = new Composite();  
        celular.getElements().add(camara);  
        celular.getElements().add(pantalla);  
        celular.getElements().add(bateria);  
        celular.getElements().add(procesador);  
        celular.getElements().add(botones);  
        celular.getElements().add(microfono);  
        System.out.println("el celular tiene un precio de "+ celular.operation());  
    }  
}
```

```
Leaf audifonos = new Leaf( price: 120000);
System.out.println("Los audifonos tienen un precio de "+ audifonos.operation());

Leaf cable = new Leaf( price: 80000);
Leaf marranito = new Leaf( price: 10000);
Composite cargador = new Composite();
cargador.getElements().add(cable);
cargador.getElements().add(marranito);
System.out.println("El cargador tiene un precio de "+ cargador.operation());

Composite empaque = new Composite();
empaquete.getElements().add(celular);
empaquete.getElements().add(audifonos);
empaquete.getElements().add(cargador);
empaquete.getElements().add(new Utilidad());
System.out.println("El empaque tiene un precio de "+ empaque.operation());
```

UML:





Salida:

```
"D:\Program Files\Java\jdk1.8.0_111\bin\java.exe" ...
```

```
el celular tiene un precio de 1170000
```

```
Los audifonos tienen un precio de 120000
```

```
El cargador tiene un precio de 90000
```

```
El empaque tiene un precio de 1480000
```

```
Process finished with exit code 0
```