



// Codice Java che implementa il design pattern Decorator

```
// Messaggio e' un Component
public interface Messaggio {
    public String getDestin();
    public String getTesto();
}
```

// Testo e' un ConcreteComponent

```
public class Testo implements Messaggio {
    private String dest;
    private String testo;
    public Testo(String d, String t) {
        dest = d;
        testo = t;
    }
    public String getDestin() {
```

```
        System.out.print("[Testo.getDestin] ");
        return dest;
    }
    public String getTesto() {
        System.out.print("[Testo.getTesto] ");
        return testo;
    }
}

// Modificatore e' un Decorator

public class Modificatore implements Messaggio {
    private Messaggio p;
    public Modificatore(Messaggio c) {
        p = c;
    }
    public String getDestin() {
        System.out.print("[Modificatore.getDestin] ");
        return p.getDestin();
    }
    public String getTesto() {
        System.out.print("[Modificatore.getTesto] ");
        return p.getTesto();
    }
}

// Riduttore e' un ConcreteDecorator

public class Riduttore extends Modificatore {
    public Riduttore(Messaggio c) {
        super(c);
    }
    // trasforma il testo fornito dal componente piu' interno
    public String getTesto() {
        System.out.print("[Riduttore.getTesto] ");
        String s = super.getTesto();
        s = s.replaceAll("!", "");
        s = s.replaceAll("four", "4");
        s = s.replaceAll("going to", "gonna");
        s = s.replaceAll("would you", "wonna");
        return s;
    }
}

// Indirizzi e' un ConcreteDecorator

public class Indirizzi extends Modificatore {
    public Indirizzi(Messaggio c) {
        super(c);
    }
}
```

```
// fa corrispondere un numero al destinatario
public String getDestin() {
    System.out.print("[Indirizzi.getDestin] ");
    String s = super.getDestin();
    s = s.replaceFirst("Taro", "3336543211");
    s = s.replaceFirst("Saro", "3939685732");
    return s;
}

// Client istanzia opportunamente e invoca i metodi

public class Client {
    private Messaggio m = new Riduttore(new Testo("Taro", "There are four of us here"));

    public static void main(String[] args) {
        System.out.println("Istanziamento ... ");
        Client c = new Client();
        c.test();
        c.test2();
    }

    private void test() {
        System.out.println("Invocazione metodo getTesto ... ");
        System.out.println("\ntesto: " + m.getTesto()+"\n");
        System.out.println("Invocazione metodo getDestin ... ");
        System.out.println("\ndestin: " + m.getDestin()+"\n");
    }

    private void test2() {
        System.out.println("Istanziamento m2 ... ");
        Messaggio m2 = new Riduttore(new Indirizzi(new Testo("Saro", "Saro, are you there?")));

        System.out.println("Invocazione metodo getTesto ... ");
        System.out.println("\ntesto: " + m2.getTesto()+"\n");
        System.out.println("Invocazione metodo getDestin ... ");
        System.out.println("\ndestin: " + m2.getDestin()+"\n");
    }
}
```

### Output dell'esecuzione

```
Istanziamento ...
Invocazione metodo getTesto ...
[Riduttore.getTesto] [Modificatore.getTesto] [Testo.getTesto]
testo: There are 4 of us here

Invocazione metodo getDestin ...
[Modificatore.getDestin] [Testo.getDestin]
```

destin: Taro

Istanziamento m2 ...

Invocazione metodo getTesto ...

[Riduttore.getTesto] [Modificatore.getTesto] [Modificatore.getTesto] [Testo.get1

testo: Saro, are you gonna go home now?

Invocazione metodo getDestin ...

[Modificatore.getDestin] [Indirizzi.getDestin] [Modificatore.getDestin] [Testo.g

destin: 3939685732