

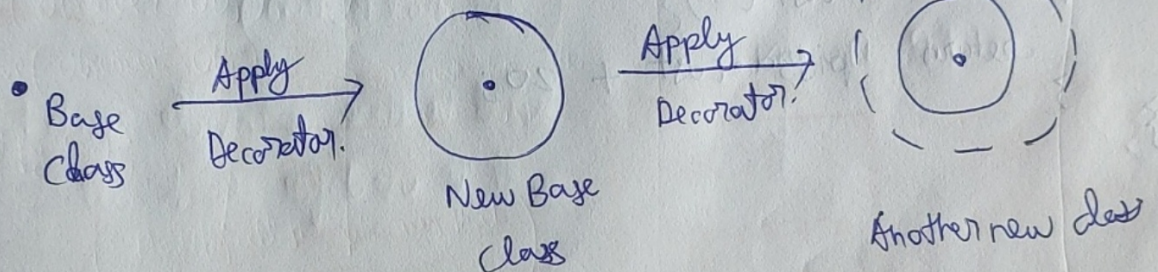
LEC 4

DECORATOR PATTERN [Structural Pattern]

This one is a very interesting one.

It is somewhat like a mask.

You have an entity and you decorate it with something that becomes a new entity.




```
public abstract class BasePizza {
    public abstract int cost();
}
```

```
public class Margarita extends BasePizza {
```

```
@Override
    public int cost() {
```

```
        return 100;
    }
```

```
public class VegDelight extends BasePizza {
```

```
@Override
```

```
    public int cost() {
        return 80;
    }
```

```
}
```

```
public abstract class Toppings extends BasePizza {
```

```
    BasePizza pizza;
```

```
    public abstract int cost();
}
```

```
public class ExtraCheese extends Toppings {
```

```
    ExtraCheese(BasePizza pizza) {
```

```
        this.pizza = pizza;
```

```
@Override
```

```
    public int cost() {
```

```
        return pizza.cost() + 20;
    }
```

```
}
```

Remember both
BasePizza and
Toppings are abstract
classes and
not interfaces

Used in ExtraCheese and olives
We reduced code
duplication
here

Learnings abstract
class can extend
another abstract class

Constructor
Injection


```
public class Olives extends Toppings {
```

```
Olives (BasePizza pizza) {
```

```
this.pizza = pizza
```

Constructor injection

```
@Override
```

```
public int cost () {
```

```
return pizza.cost + 10;
```

```
}
```

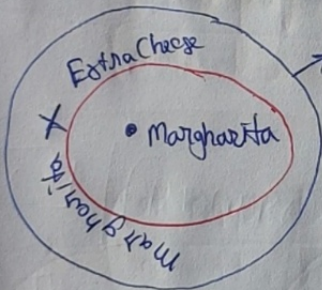
Pizza Shop

```
just Base Pizza just Margharita = new Margharita();
```

→ Just
margharita

just Margharita.cost() → 100.

```
Base Pizza addCheese = new ExtraCheese(justMargharita);
```



decorating
it.

```
addCheese.cost()
```

BasePizza = justMargharita

```
public int cost () {
```

```
BasePizza.cost + 20
```

120

Margharita

```
cost () {
```

```
return 100;
```

Now you can even add olives on
top of that

```
Base Pizza olives = new Olives (addCheese);
```

```
olives.cost
```

130

olives

```
cost () {
```

```
pizza.cost + 10
```

pizza = addCheese

ExtraCheese

```
cost () {
```

```
pizza.cost + 20
```

pizza = Margharita

Margharita

```
cost () {
```

```
return 100;
```