

Façade Design Pattern

by Ali Alkhafaji

Unified interface for a set of
interfaces to promote readability and
usability.

Motivation

- Dealing with multiple interfaces could make code harder to read or understand.
- Complexity of multiple systems makes it more difficult for coders to implement.

Mobile Phone Example Before Façade

/ Classes */*

```
class Network {
    public int findNetwork() {
        ...
    }
}
class Connection {
    public void connect(Network net, int number) {
        ...
    }
    public void disconnect(){
        ...
    }
}
class Keypad {
    public int getDialedNumber() {
        ...
    }
}
```

/ Client */*

```
class Call {
    public static void main(String[] args) {
        Network net = new Network ();
        Connection con = new Connection ();
        Keypad kp = new Keypad ();

        int availableNet = net.findNetwork();
        int number = kp.getDialedNumber();
        con.connect(availableNet, number);
        con.disconnect();
    }
}
```

Mobile Phone Example After Facade

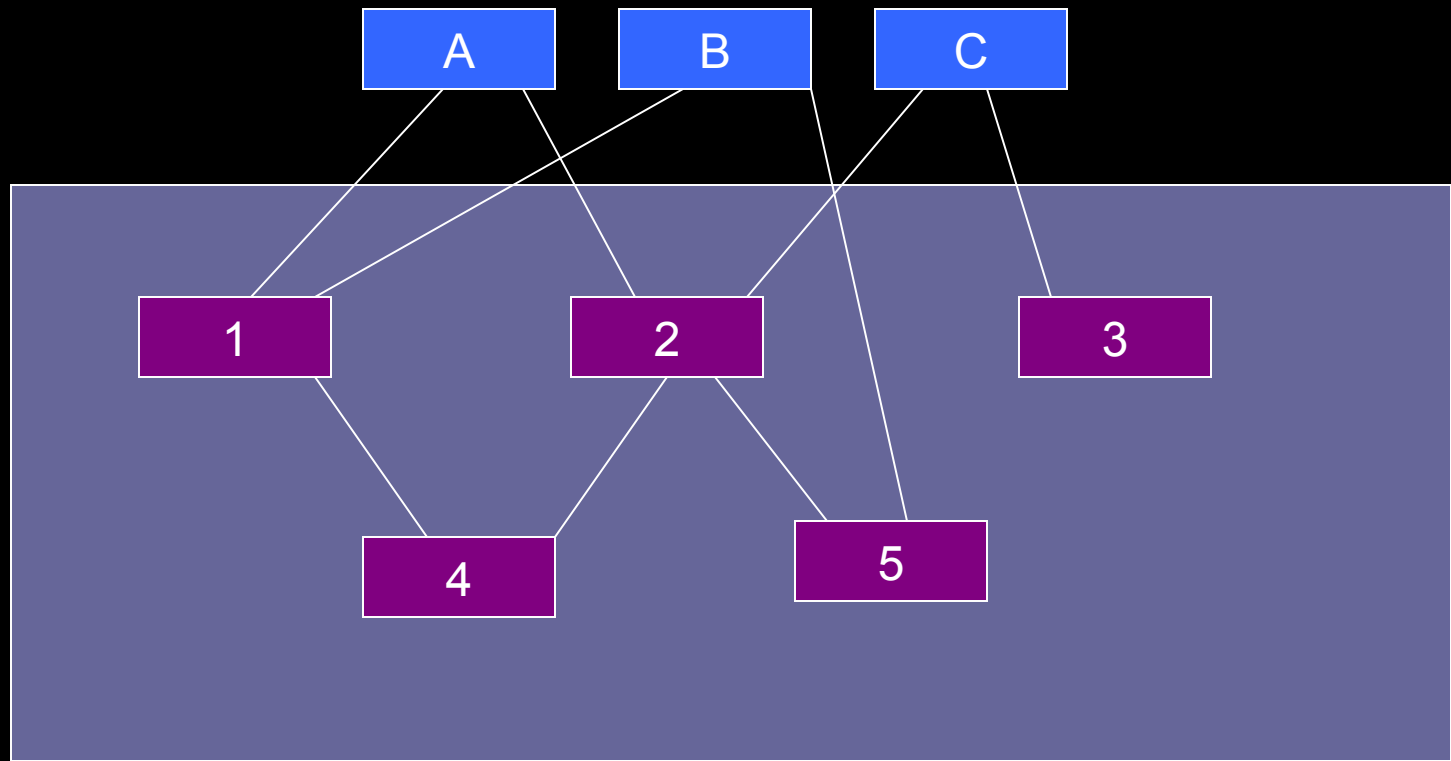
/ Facade */*

```
class Phone {  
    private Network net=null;  
    private Connection con=null;  
    private Keypad kp=null;  
  
    public Phone() {  
        this.net = new Network ();  
        this.con = new Connection ();  
        this.kp = new Keypad ();  
    }  
    public void call() {  
        int availableNet= net.findNetwork();  
        int number = kp.getDialedNumber();  
        con.connect(availableNet, number);  
        con.disconnect();  
    }  
}
```

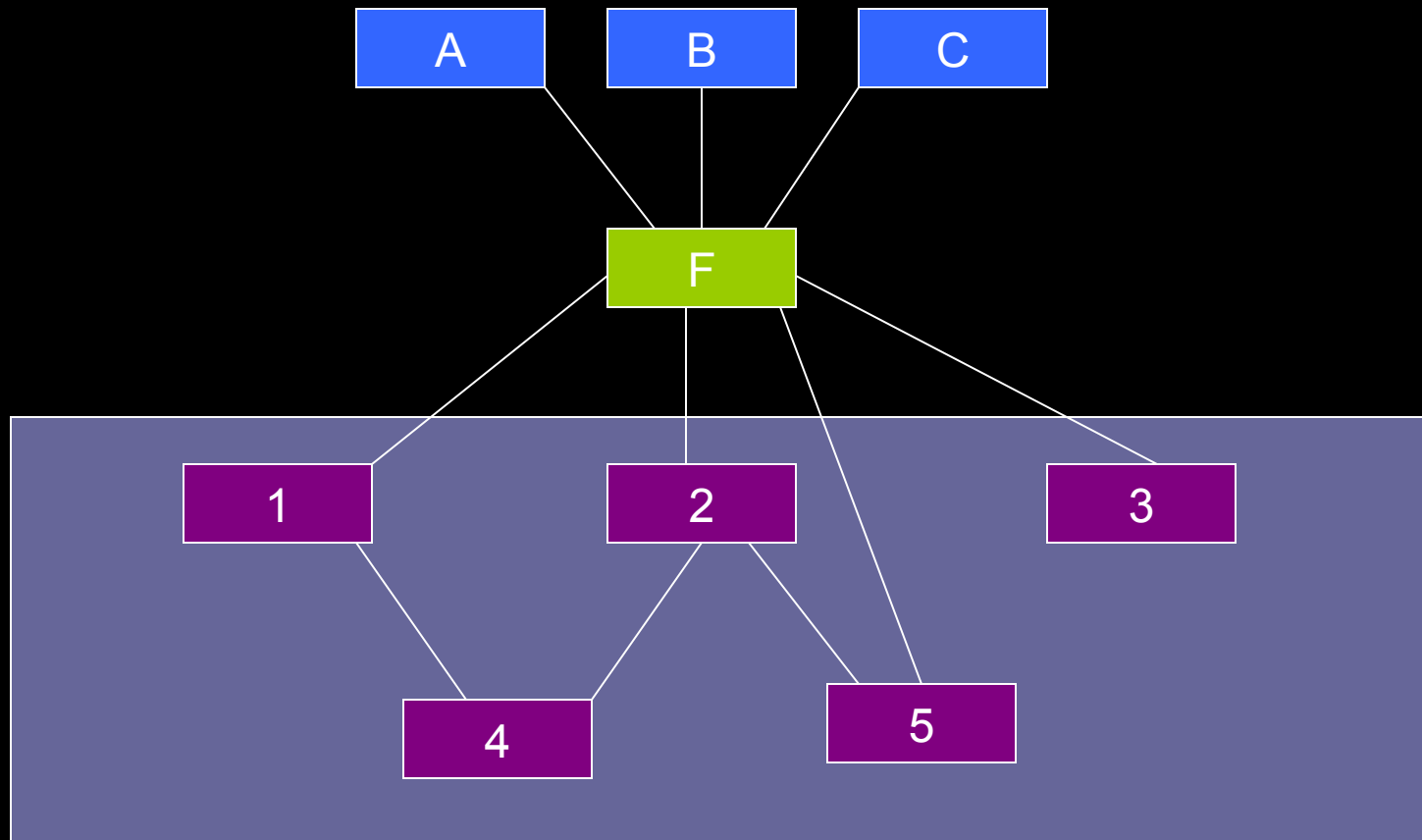
/ Client */*

```
class Call {  
    public static void main(String[] args) {  
        Phone phone = new Phone();  
  
        phone.call();  
    }  
}
```

Before Façade Pattern



After Façade Pattern



Similar Patterns

- **Adapter Pattern:**

- However, the Façade Pattern defines a new interface where the Adapter Pattern use an old one.

- **Flyweight Pattern:**

- In the Façade Pattern we are introducing one object to represent a subsystem where in the in the Flyweight Pattern we are introducing many small objects.

- **Mediator Pattern:**

- Just like the Façade Pattern, the Mediator Pattern abstracts functionality but unlike the Façade Pattern it adds new value instead of making the subsystem simpler.

Questions...

© Original Artist
Reproduction rights obtainable from
www.CartoonStock.com



MARK AND GWYNETH HAD A DARK SECRET. AT NIGHT,
ONCE THE CURTAINS WERE CLOSED TO THE SUBURBAN WORLD
OUTSIDE, THE FACADE THEY PUT UP TO THE NEIGHBOURS WAS
REVEALED TO BE 100% GENUINE... THEY WERE TOTALLY BORING.

search ID: rjo1140