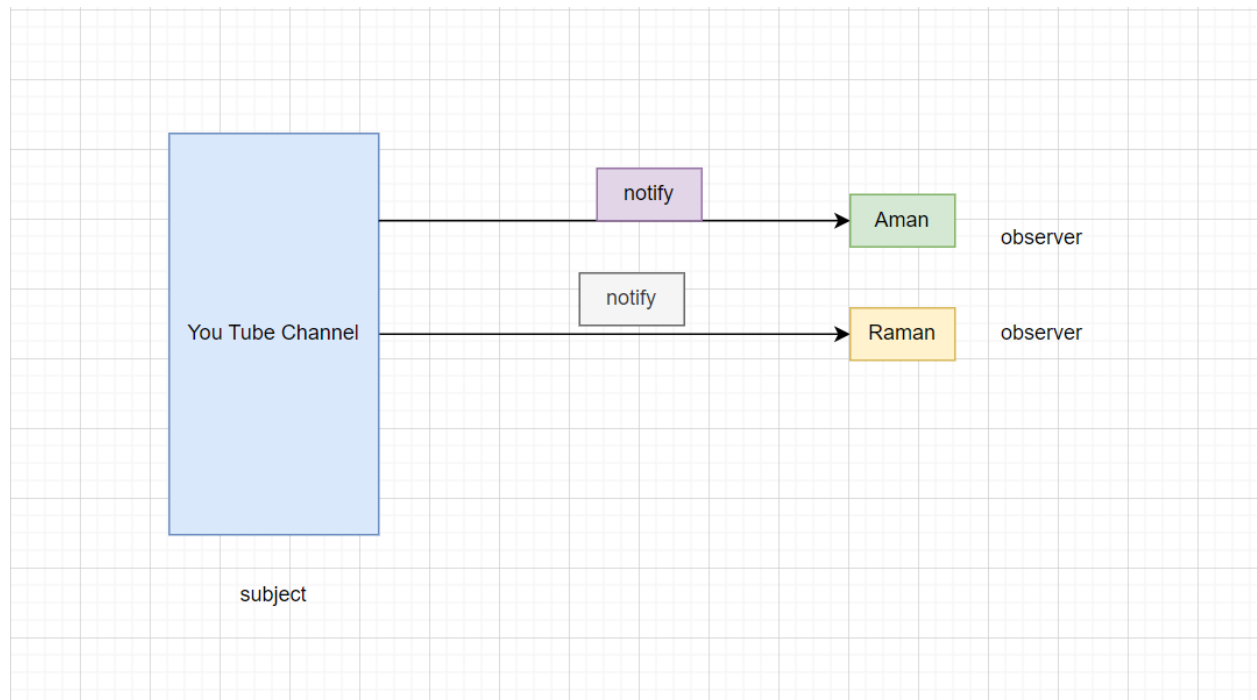


Observer Design Pattern

1. It is a behavioral design pattern
It doesn't care about how object is creating but it take cares about like how 2 objects interact with each other.
2. In this when subject changes its state all its dependent objects notified the changes



Let us suppose we have one you tube channel. If we upload the any new video then how Aman/Raman comes to know that new video is uploaded.

One way is they regularly comes to channel and check. But this not good way.

Second way is if we have subscribe button and if Aman/Raman click on subscribe button, then during upload any video in you tube, a notification will send to Aman/Raman.

Here the youtube is subject. Since Aman/Raman is observing the channel so they are observer.

Steps to implement Behavior DP

1. Create subject interface

```
public interface Subject {  
  
    void subscribe(Observer ob);  
  
    void unsubscribe(Observer ob);  
  
    void notifyChanges();  
}
```

2. Create Observer interface

```
public interface Observer {  
  
    void notified();  
}
```

3. Create YouTubeChannel class implements the Subject

```
public class YouTubeChannel implements Subject {  
  
    List<Observer> subscriber = new ArrayList<>();  
  
    @Override  
    public void subscribe(Observer ob) {  
        this.subscriber.add(ob);  
    }  
  
    @Override  
    public void unsubscribe(Observer ob) {  
        this.subscriber.remove(ob);  
    }  
  
    @Override  
    public void videoUploaded() {  
        for (Observer ob : this.subscriber) {  
            ob.notified();  
        }  
    }  
}
```

4. Create Subscriber class implements the Observer

```
public class Subscriber implements Observer{

    String name;

    Subscriber(String name) {
        this.name=name;
    }

    @Override
    public void notified() {
        System.out.println("Hello " + this.name + " The new video is uploaded");
    }
}
```

5. Create Main class

```
public class Demo {

    public static void main(String[] args) {

        YouTubeChannel channel = new YouTubeChannel();

        Subscriber subscriber = new Subscriber("Anuj");
        channel.subscribe(subscriber);
        channel.videoUploaded();
    }
}
```