Padrão Observer

Matheus Vianna Nunes
Gabriel De Albuquerque Basileu

Definição

É um padrão que define uma cadeia de dependência de um pra muitos de forma em que quando ocorre mudança com o objeto observado, que é chamado de “Subject”, os objetos dependentes são notificados, estes são chamados de “Observers”.

O “Subject” notifica seus dependentes normalmente chamando um de seusmétodos.

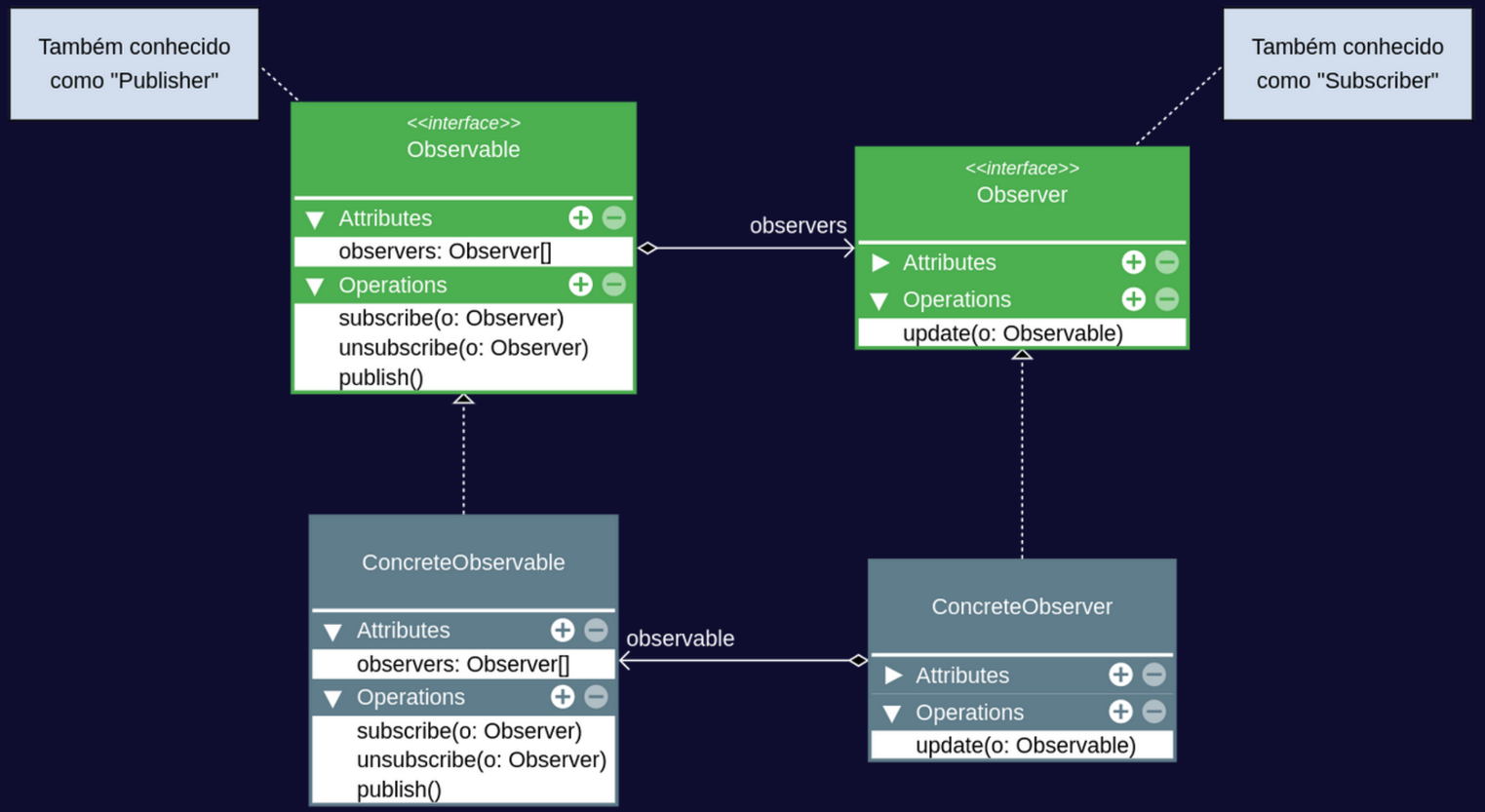
Subject

- Subject é o objeto observado, por padrão é responsável por manter uma lista de seus dependentes e notifica-los eventualmente sobre suas mudanças de estado.

Observer

- Os Observers são os objetos que recebem atualizações do Subject, podendo assim registrar-se no Subject para receber suas atualizações ou cancelar o registro assim deixando de receber atualizações.

Modelagem Conceitual



Relações com Outros Padrões

O Chain of Responsibility, Command, Mediator e Observer abrangem várias maneiras de se conectar remetentes e destinatários de pedidos:

- O Chain of Responsibility passa um pedido sequencialmente ao longo de um corrente dinâmica de potenciais destinatários até que um deles atua no pedido.

- O Command estabelece conexões unidirecionais entre remetentes e destinatários.

- O Mediator elimina as conexões diretas entre remetentes e destinatários, forçando-os a se comunicar indiretamente através de um objeto mediador.

- O Observer permite que destinatários inscrevam-se ou cancelem sua inscrição dinamicamente para receber pedidos.

Exemplo Observer

```java
package br.edu.femass;

public interface Subscriber {
    public void update(String s, String a);
}
```

Exemplo Subject

```java
package br.edu.femass;

public interface Subject {
    public void registerObserver(Subscriber observer);
    public void unregisterObserver(Subscriber subscriber);
    public void notifyAllObserver(String notify);
}
```

```java
package br.edu.femass;
import java.util.ArrayList;

public class Account implements Subject {
    private String accountName;
    private ArrayList<Subscriber> followers;
    public Account(String accountName) {
        this.accountName = accountName;
        followers = new ArrayList<Subscriber>();
    }
    @Override
    public void registerObserver(Subscriber subscriber) {
        if(subscriber != null){
            this.followers.add(subscriber);
        }
        System.out.println(subscriber + " Started following " + accountName);
    }

    @Override
    public void unregisterObserver(Subscriber subscriber) {
        if(subscriber != null){
            this.followers.remove(subscriber);
        }
        System.out.println(subscriber + " Stopped following " + accountName);
    }

    @Override
    public void notifyAllObserver(String tweet) {
        for(Subscriber follower : followers)
            follower.update(accountName, tweet);
    }
    public void tweet(String tweet) {
        System.out.println(accountName + " has tweeted: " + tweet + "\n");
        notifyAllObserver(tweet);
    }
}
```

Exemplo Classe Concreta do Subject

Exemplo Classe Concreta do Observer

```java
package br.edu.femass;

...
public class Follower implements Subscriber {
    private String followerName;

    public Follower(String followerName){
        this.followerName = followerName;
    }

    @Override
    public void update(String accountName, String tweet) {
        System.out.println(followerName + " has received " + accountName + "'s tweet: " + tweet);
    }

    @java.lang.Override
    public java.lang.String toString() {
        return "Follower{" + followerName + "}";
    }

}
```

Exemplo de uso classe Main

```java
package br.edu.femass;

...
public class Main {
    Run | Debug
    public static void main(String[] args) {
        //Criando as contas
        Account matheus = new Account(accountName: "Matheus");
        Account gabriel = new Account(accountName: "Gabriel");

        //Criando os seguidores de Matheus
        Follower Gustin = new Follower(followerName: "@Gustin");
        Follower Mefiu = new Follower(followerName: "@Baby");
        Follower Tozin = new Follower(followerName: "@Tozin");

        //Criando os seguidores de Gabriel
        Follower Baby = new Follower(followerName: "@Mefiu");
        Follower Adryan = new Follower(followerName: "@Adryan");
        Follower Dudu = new Follower(followerName: "@Pedro");

        //Adicionando os seguidores a lista de 'seguidos' de Matheus
        matheus.registerObserver(Gustin);
        matheus.registerObserver(Mefiu);
        matheus.registerObserver(Tozin);

        //Adicionando os seguidores a lista de 'seguidos' de Gabriel
        gabriel.registerObserver(Baby);
        gabriel.registerObserver(Adryan);
        gabriel.registerObserver(Dudu);

        //Gerando um publicação no Twitter
        matheus.tweet(tweet: "guys dont play league of legends, once you start you wont be able to do anything else... help");
        gabriel.tweet(tweet: "Twitter is purging a lot of spam/scam accounts right now, so you may see your follower count drop");

        //Retirando um seguidor da lista de 'seguidos' de Matheus
        matheus.unregisterObserver(Gustin);
    }
}
```

Saída no Terminal

```
Follower{@Gustin} Started following Matheus
Follower{@Baby} Started following Matheus
Follower{@Tozin} Started following Matheus
Follower{@Mefiu} Started following Gabriel
Follower{@Adryan} Started following Gabriel
Follower{@Pedro} Started following Gabriel
Matheus has tweeted: guys dont play league of legends, once you start you wont be able to do anything else... help

@Gustin has received Matheus's tweet: guys dont play league of legends, once you start you wont be able to do anything else... help
@Baby has received Matheus's tweet: guys dont play league of legends, once you start you wont be able to do anything else... help
@Tozin has received Matheus's tweet: guys dont play league of legends, once you start you wont be able to do anything else... help
Gabriel has tweeted: Twitter is purging a lot of spam/scam accounts right now, so you may see your follower count drop

@Mefiu has received Gabriel's tweet: Twitter is purging a lot of spam/scam accounts right now, so you may see your follower count drop
@Adryan has received Gabriel's tweet: Twitter is purging a lot of spam/scam accounts right now, so you may see your follower count drop
@Pedro has received Gabriel's tweet: Twitter is purging a lot of spam/scam accounts right now, so you may see your follower count drop
Follower{@Gustin} Stopped following Matheus

Process finished with exit code 0
```

Bibliografia

- Design Patterns: Elements of Reusable Object-Oriented Software
- https://www.devmedia.com.br/padrao-de-projeto-observer-em-java/26163
- https://refactoring.guru/pt-br/design-patterns/observer

Repositório:

https://github.com/GabrielAL4/observer-pattern