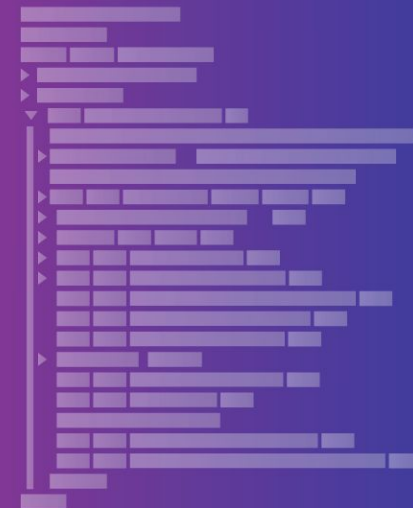


Encapsulamento e Sobrecarga



DEVinHouse

Parcerias para desenvolver a sua carreira

SENAI

<LAB365>

AGENDA

- Encapsulamento
- Sobrecarga

Boas práticas com Atributos

Nas boas práticas da Orientação a Objetos temos as boas práticas relacionadas aos atributos, que incluem tanto o nome dos atributos quanto a forma de acesso a eles.

Vimos que os nomes devem ser significativos e devem descrever a informação carregada por eles.

Em relação ao acesso a informação realizamos o controle da seguinte forma: todos os atributos de uma classe devem utilizar o modificador **private**, e para realizarmos o acesso a esse atributo devemos utilizar métodos chamados Getter e Setter.

Getters e Setters

Os Getters e Setters são métodos exclusivos para realizar o acesso aos atributos de uma classe. Os Getters são utilizados para pegar o valor de um atributo e os Setter para alterar o valor desse atributo.

Podemos ter atributos que são acessíveis apenas através do Getter e outro que são acessíveis através do Setter, assim há um controle do acesso de uma classe para o mundo externo.

Devemos apenas ter os Getters e Setter necessários para o funcionamento do programa, se não realizamos a leitura de um atributo, ele não deve ter um Getter.

Getter

Abaixo temos o exemplo do Getter, esse valores pode ser recuperados, mas não temos como adicionar valor aos atributos abaixo:

```
public class Pessoa {  
    private String nome;  
    private Integer idade;  
  
    public String getNome() {  
        return nome;  
    }  
  
    public Integer getIdade() {  
        return idade;  
    }  
}
```

Setter

Abaixo temos o exemplo do Setter, ele nos permite adicionar valor aos atributos, mas não nos permite visualizar os valores desses atributos.

```
public class Pessoa {  
    private String nome;  
    private Integer idade;  
  
    public void setName(String nome) {;  
        this.nome = nome;  
    }  
  
    public void setIdade(Integer idade) {  
        this.idade = idade;  
    }  
}
```

INTERVALO DE AULA

DEV!

Finalizamos o nosso primeiro período de hoje. Que tal descansar um pouco?!

Nos vemos em 20 minutos.

Início: 20:20

Retorno: 20:40



A Sobrecarga na Orientação a Objetos se refere a uma classe ter 2 ou mais métodos com o mesmo nome. Sendo assim podemos utilizar o mesmo nome, mas os parâmetros e o tipo de retorno de um método serão diferentes.

Esses componentes do método são o que chamamos de assinatura do método e contanto que a assinatura seja única o nome pode ser o mesmo. Vamos ver um exemplo de Sobrecarga.

Abaixo temos 2 métodos soma(), o primeiro tem 2 parâmetros e o segundo tem 3, assim já temos diferenças suficientes para o programa diferenciá-los.

```
public Double soma(Double numero1, Double numero2) {  
    return numero1+numero2;  
}
```

```
public Double soma(Double numero1, Double numero2, Double numero3) {  
    return numero1+numero2+numero3;  
}
```

Sobrecarga

Podemos também ter um tipo diferente de retorno entre os métodos em sobrecarga, abaixo temos 2 tipos de retorno para métodos com o mesmo nome:

```
List<Produto> produtos = new ArrayList<>();
```

```
public Produto removeProduto(int id) {  
    Produto resultado = produtos.remove(id);  
    return resultado;  
}
```

```
public Boolean removeProduto(Produto produto) {  
    Boolean resultado = produtos.remove(produto);  
    return resultado;  
}
```

AVALIAÇÃO DOCENTE

O que você está achando das minhas aulas neste conteúdo?

[Clique aqui](#) ou escaneie o QRCode ao lado para avaliar minha aula.

Sinta-se à vontade para fornecer uma avaliação sempre que achar necessário.





DEVinHouse

Parcerias para desenvolver a sua carreira

OBRIGADO!



<LAB365>