



UNIVERSIDADE
TÉCNICA DO
ATLÂNTICO



CAMPUS
DO MAR

UNIVERSIDADE TÉCNICA DO ATLÂNTICO - UTA
LICENCIATURA ENGENHARIA INFORMÁTICA E TELECOMUNICAÇÕES

METODOS E CLASSES EM JAVA

Discente:
José Carlos Costa

Docente: Paulo Silva

MINDELO - SV
29 DE MARÇO DE 2023

SUMÁRIO

1 – Introdução	1
2 – Metodos e Classes	3
2.1 O que é uma classe	3
2.2 O que são variáveis e métodos de instância	3
2.3 O que é e para que serve método construtor	3
2.4 Encapsulamento	5
3 – Considerações finais	7

1 INTRODUÇÃO

Uma classe é um conceito fundamental da programação orientada a objetos, é um modelo para representar objetos do mundo real. Uma classe em JAVA contém variáveis e métodos, que podem ser usados para modelar um objeto ou entidade. As variáveis de instância são variáveis de objeto, cada objeto terá os seus próprios valores das variáveis de instância.

Os métodos de instância são funções associadas ao objeto, que podem ser usados para alterar o estado do objeto. Um método construtor é uma função especial que é invocada quando um novo objeto é criado. O construtor geralmente inicializa as variáveis de instância e executa outras tarefas necessárias para criar o objeto.

Atributos de acesso determinam quem pode acessar e alterar as variáveis de instância e os métodos de instância. O encapsulamento é o princípio que permite que as variáveis de instância e os métodos de instância sejam acessíveis apenas por outros membros da classe.

2 METODOS E CLASSES

2.1 O QUE É UMA CLASSE

Classe vem da taxonomia da biologia. Todos os seres vivos de uma mesma classe biológica têm uma série de atributos e comportamentos em comum, mas não são iguais, podem variar nos valores desses atributos e como realizam esses comportamentos.

Uma classe em Java é uma estrutura de objetos que define as características de um objeto, como variáveis de objeto (atributos) e comportamentos (métodos). Ela define a estrutura de dados usada para criar objetos que compartilham o mesmo tipo.

Por exemplo, a classe conta que definimos acima tem quatro atributos: número, dono, saldo e limite. Todos esses atributos são variáveis de objeto que definem as características de uma conta. Além disso, a classe conta também pode ter métodos, como depositar () e sacar (), que definem o comportamento da conta.

2.2 O QUE SÃO VARIÁVEIS E MÉTODOS DE INSTÂNCIA

Variáveis de instância são variáveis declaradas dentro de uma classe, mas fora de qualquer método. Essas variáveis são instanciadas quando um objeto é criado com a palavra-chave `new`, e são usadas para representar os estados internos do objeto. Métodos de instância são métodos que podem ser chamados em um objeto. Eles são declarados dentro da classe, mas fora de qualquer método. Esses métodos são usados para manipular o estado interno de um objeto. Exemplo:

```
public class Car // Variável de instância private String color; // Método de
instância public void setColor(String color) this.color = color;
```

Neste exemplo, a variável de instância `color` é usada para representar o estado interno do objeto `Car`. O método de instância `setColor` é usado para manipular o estado interno do objeto `Car`, alterando a cor do carro.

2.3 O QUE É E PARA QUE SERVE MÉTODO CONSTRUTOR

Sempre quando criamos um novo objeto em Java, utilizamos a sintaxe: `Classe nomeObjeto = new Classe();` Um método construtor, como o próprio nome já diz, é responsável pela criação do objeto daquela classe, iniciando com valores seus atributos ou realizando outras funções que possam vir a ser necessárias. Para que um método seja considerado construtor, ele deve possuir o mesmo nome da classe, inclusive com

correspondência entre letras maiúsculas e minúsculas e não deve ter retorno. Quando usamos a palavra-chave `new`, estamos passando para ela como um parâmetro, qual construtor deve ser executado para instanciar um objeto. Por padrão, todas as classes possuem um construtor com seu nome seguindo de parênteses `()`. Caso você não declare manualmente o construtor, o compilador do Java fará isso por você.

```
// Construtor public Pessoa(String n, int i, String e) nome = n; idade = i; endereco = e;
```

Atributos de acesso e como isso afeta o acesso a variáveis de instância e métodos. Palavras-chave específicas de acesso que definem o nível de acesso que tanto variáveis quanto métodos de instância tem são chamados de atributos de acesso. A finalidade desses atributos é controlar o acesso aos membros da classe, ou seja, quem pode usar determinadas variáveis ou métodos.

Os atributos de acesso mais usuais são: `“public”`, `“private”` e `“protected”`. O `“public”` permite que qualquer pessoa tenha uso e acesse o método ou variável. Já o `“private”` restringe ao membro da classe, sendo que somente o código interno da classe tem acesso ao método ou variável. Por fim, o `“protected”` restringe ao membro da classe e às classes filhas.

As variáveis de instância são declaradas no âmbito da classe e ficam acessíveis aos seus métodos e aos métodos de outras classes. As diretrizes de acesso definidas para elas determinam como se pode ter acesso. Caso sejam declaradas como `“public”`, sua utilização é possível por qualquer classe. Se forem definidas como `“private”`, somente os métodos da mesma classe tem acesso.

Funcionalidades são definidas no âmbito da classe. Os atributos de acesso atribuídos a estes métodos estabelecem a maneira como podem ser chamados. Se um método for assinalado como `“public”`, ele pode ser usado por qualquer classe. Se for declarado como `“private”`, somente os métodos da mesma classe podem acessá-lo.

Como resultado, os atributos de acesso influenciam quais membros da classe podem ser expostos a outras classes. Se eles estiverem definidos como `“public”`, todas as variáveis e métodos da classe estarão disponíveis para uso fora da classe. Se eles forem definidos como `“private”` ou `“protected”`, somente alguns membros da classe estarão disponíveis. Os atributos de acesso controlam o nível de acesso que variáveis de instância e métodos têm e, como consequência, afetam a exposição de membros da classe a outras classes. Se definidos como `“public”`, todos os membros da classe estarão disponíveis, mas se forem declarados como `“private”` ou `“protected”`, somente alguns estarão acessíveis.

2.4 ENCAPSULAMENTO

Encapsulamento vem de encapsular, que em programação orientada a objetos significa separar o programa em partes, o mais isolado possível. A idéia é tornar o software mais flexível, fácil de modificar e de criar novas implementações. O Encapsulamento serve para controlar o acesso aos atributos e métodos de uma classe.

É uma forma eficiente de proteger os dados manipulados dentro da classe, além de determinar onde esta classe poderá ser manipulada. Usamos o nível de acesso mais restritivo, `private`, que faça sentido para um membro particular. Sempre usamos `private`, a menos que tenhamos um bom motivo para deixá-lo com outro nível de acesso. Não devemos permitir o acesso público aos membros, exceto em caso de ser constantes.

Isso porque membros públicos tendem a nos ligar a uma implementação em particular e limita a nossa flexibilidade em mudar o código.

3 CONSIDERAÇÕES FINAIS

Podemos concluir que a classe é uma estrutura de objetos que define as características de um objeto, como variáveis de objeto (atributos) e comportamentos (métodos). As variáveis de instância são usadas para representar os estados internos do objeto e os métodos de instância são usados para manipular o estado interno do objeto. O método construtor é uma função especial que é invocada quando um novo objeto é criado.

Os atributos de acesso determinam quem pode acessar e alterar as variáveis de instância e os métodos de instância. O encapsulamento é o princípio que permite que as variáveis de instância e os métodos de instância sejam acessíveis apenas por outros membros da classe.