

Métodos, atributos e classes no Java

Veja neste artigo quais são e como utilizar os modificadores de acesso a métodos, atributos e classes da linguagem Java. Serão apresentados aqui os modificadores `private`, `public`, `protected`, `default`, `final`, `abstract` e `static`.

Os **modificadores de acesso** são padrões de visibilidade de acessos às **classes, atributos e métodos**. Esses modificadores são palavras-chaves reservadas pelo [Java](#), ou seja, palavras reservadas não podem ser usadas como nome de métodos, classes ou atributos.

Como boas práticas (best practices) do **Java**, na maioria das declarações de variáveis de instância são definidos os seus atributos com a palavra-chave **private**, para garantir a segurança de alterações acidentais, sendo somente acessíveis através dos métodos. Essa ação tem como efeito ajudar no encapsulamento dos dados, preservando ainda mais a segurança e a aplicação de programação orientada a objetos do Java.

 Documentação: [Modificadores de acesso no PHP](#)

Por exemplo, quando um programa cria (instancia) um objeto da classe Banco, a variável senha é encapsulada (ocultada) no objeto onde pode ser acessada apenas por métodos da classe do objeto, os métodos **getters** e **setters**, que manipulam a variável de instância. Vejamos abaixo a explicação sobre cada um.

 Saiba mais: [Como criar minha primeira classe em Java](#)

public

Uma declaração com o modificador `public` pode ser acessada de qualquer lugar e por qualquer entidade que possa visualizar a classe a que ela pertence.

private

Os membros da classe definidos como não podem ser acessados ou usados por nenhuma outra classe. Esse modificador não se aplica às classes, somente para seus métodos e atributos. Esses atributos e métodos também não podem ser visualizados pelas classes herdadas.

protected

O modificador `protected` torna o membro acessível às classes do mesmo pacote ou através de herança, seus membros herdados não são acessíveis a outras classes fora do pacote em que foram declarados.

default (padrão):

A classe e/ou seus membros são acessíveis somente por classes do mesmo pacote, na sua declaração não é definido nenhum tipo de modificador, sendo este identificado pelo compilador.

Listagem 1: Exemplo de declaração do modificador default

```
public class Modificador_Default {  
  
    public static void main(String[] args) {  
  
        String nome = "Flávia Bernandes";  
  
        System.out.printf("Nome.: %s", nome);  
  
    }  
  
}
```

final

Quando é aplicado na classe, não permite estende-la, nos métodos impede que o mesmo seja sobrescrito (overriding) na subclasse, e nos valores de variáveis não pode ser alterado depois que já tenha sido atribuído um valor.

abstract

Esse modificador não é aplicado nas variáveis, apenas nas classes. Uma classe abstrata não pode ser instanciada, ou seja, não pode ser chamada pelos seus construtores. Se houver alguma declaração de um método como abstract (abstrato), a classe também deve ser marcada como abstract.

static

É usado para a criação de uma variável que poderá ser acessada por todas as instâncias de objetos desta classe como uma variável comum, ou seja, a variável criada será a mesma em todas as instâncias e quando seu conteúdo é modificado numa das instâncias, a modificação ocorre em todas as demais. E nas declarações

de métodos ajudam no acesso direto à classe, portanto não é necessário instanciar um objeto para acessar o método.

No exemplo da Listagem 2, é gerada a saída da contagem de 4 valores (1,2,3,4), se comentarmos a linha que tem a variável static e descomentar da qual não tem, veremos que a sua saída será de 4 saídas com o mesmo valor (1,1,1,1).

Nas variáveis estáticas isso acontece porque todas as instâncias da mesma classe compartilham a mesma cópia das variáveis estáticas, sendo inicializadas quando a classe é carregada (instanciada).

Listagem 2: Exemplo do modificador static

```
public class Conta_Instancias {

    private int tamanho;

    private static int conta = 0; //IMPRIME A CONTAGEM DE 4 VALORES

    //private int conta = 0; //IMPRIME A CONTAGEM DE 1 VALOR

    public Conta_Instancias(){

        conta++;

        System.out.println("Valor = "+conta);

    }

    public static void main(String[] args) {

        Conta_Instancias c = new Conta_Instancias();

        Conta_Instancias dois = new Conta_Instancias();

        Conta_Instancias tres = new Conta_Instancias();

        Conta_Instancias quatro = new Conta_Instancias();

    }
```

}

Tabela dos modificadores de acesso

 Artigo relacionado: [Modificadores de acesso do Java](#)

	private	default	protected	public
mesma classe	sim	sim	sim	sim
mesmo pacote	não	sim	sim	sim
pacotes diferentes (subclasses)	não	não	sim	sim
pacotes diferentes (sem subclasses)	não	não	não	sim

Ainda no Java existem alguns outros tipos que não são muito usados, os que apresentei são os mais usados e os principais.

Enviem-me opiniões do que acharam desse post, espero que tenham gostado até a próxima!