

Escopo das Variáveis

ELABORATA
INFORMÁTICA



Variáveis

- Até agora, todas as variáveis que usamos foram declaradas no início do método main().
- Porém, Java permite que as variáveis sejam declaradas dentro de qualquer bloco.
- Um bloco começa com uma chave de abertura e termina com uma chave de fechamento.
- O bloco define um escopo. Logo, sempre que você iniciar um novo bloco criará um novo escopo. Um escopo determina o tempo de vida desses objetos.



Variáveis

- Os escopos mais importantes em Java são os definidos por uma classe e os definidos por um método.
- O escopo definido por um método começa com uma chave de abertura.
- No entanto, se esse método tiver parâmetros, eles também estarão incluídos dentro do escopo do método.



Variáveis

- Como regra, as variáveis declaradas dentro de um escopo não podem ser vistas (isto é, acessadas) por um código definido fora desse escopo.
- Logo, quando você declarar uma variável dentro de um escopo, localizará essa variável e protegendo-a contra modificação ou acesso não autorizado.
- Na verdade, as regras do escopo fornecem a base do encapsulamento.



Variáveis

- Os escopos podem ser aninhados.
- Quando isso ocorre, o escopo externo engloba o escopo interno.
- Ou seja, os objetos declarados no escopo externo poderão ser vistos por um código que estiver dentro do escopo interno.
- No entanto, o inverso não é verdadeiro. Objetos declarados dentro do escopo interno não podem ser vistos fora dele.



Variáveis

// Demonstra o escopo de bloco.

```
public class ScopoDemo {  
    public static void main(String[] args) {  
        int x; // conhecida pelo código dentro de main  
  
        x = 10;  
        if (x == 10) { //inicia novo escopo  
            int y = 20; // conhecida apenas nesse bloco  
            // tanto x quanto y são conhecidos aqui  
            System.out.println(" x e y: " + x + " " + y);  
            x = y * 2;  
        }  
        // y = 100; // Erro! y não é conhecida aqui (Aqui "y"  
        // está fora de seu escopo)  
  
        // x ainda é conhecida aqui  
        System.out.println("x = " + x);  
    }  
}
```



Variáveis

// Demonstra o tempo de vida de uma variável.

```
public class VarInitDemo {  
    public static void main(String[] args) {  
        int x;  
        for (x = ;x < 3; x++) {  
            int y = -1; // y será inicializada sempre que entrarmos no  
            bloco  
            System.out.println("y = " + y); // essa linha sempre exibe  
            y=100;  
            System.out.println("y agora é = " + y);  
        }  
    }  
}
```

A saída gerada é mostrada abaixo:

```
y = -1  
y agora é = 100  
y = -1  
y agora é = 100  
y = -1  
y agora é = 100
```



Variáveis

```
/*
 * Este programa tenta declarar uma variavel em um escopo
 * interno com o mesmo nome de uma definida em um escopo externo.
 *
 * *** o programa nao sera compilado
 */
public class NestVar {
    public static void main(String[] args) {
        int count;

        for (count = 0; count < 10; count = count + 1) {
            System.out.println("This is count: " + count); // essa
            linha sempre exibe -1
            int count; // invalido!!! Não pode declarar novamente por já
            ter sido declarada
            for (count = 0; count < 2; count++) {
                System.out.println("Este programa está com erro");
            }
        }
    }
}
```

