

Variáveis Primitivas, Escopo de Variáveis e Operadores

Programação Orientada a Objetos

Prof. Tulio Alberton Ribeiro

Instituto Federal de Santa Catarina – IFSC
campus São José
`tulio.alberton@ifsc.edu.br`

11 de setembro de 2014



Aprenderemos a trabalhar com os seguintes recursos da linguagem Java

- declaração, atribuição de valores e casting;
- escopo de variáveis.



Dois modos de definir comentários.



Dois modos de definir comentários.

- Utilizando // você comenta até o final da linha



Dois modos de definir comentários.

- Utilizando `//` você comenta até o final da linha
- Utilizando `/* */` você comenta o que estiver dentro, útil para comentários longos.



Dois modos de definir comentários.

- Utilizando `//` você comenta até o final da linha
- Utilizando `/* */` você comenta o que estiver dentro, útil para comentários longos.

```
1 // declara a idade, isso é um comentário
2 int idade;
3 idade = 15;
4
5 /* isso também é um
6 comentário, porém com mais linhas,
7 como pode ser visto...
8 */
```



Dentro de um bloco podemos declarar e usar variáveis.

- Em java toda variável tem um tipo (String, int, Double, boolean, ...).



Dentro de um bloco podemos declarar e usar variáveis.

- Em java toda variável tem um tipo (String, int, Double, boolean, ...).
- Uma vez declarado o tipo da variável, não se pode mais alterar.



Dentro de um bloco podemos declarar e usar variáveis.

- Em java toda variável tem um tipo (String, int, Double, boolean, ...).
- Uma vez declarado o tipo da variável, não se pode mais alterar.
- Exemplo: tipoDaVariável nomeDaVariável.



Dentro de um bloco podemos declarar e usar variáveis.

- Em java toda variável tem um tipo (String, int, Double, boolean, ...).
- Uma vez declarado o tipo da variável, não se pode mais alterar.
- Exemplo: tipoDaVariável nomeDaVariável.
- Pode-se por praticidade no momento em que declara uma variável, já iniciá-la.



Declarando e Usando Variáveis

Dentro de um bloco podemos declarar e usar variáveis.

- Em java toda variável tem um tipo (String, int, Double, boolean, ...).
- Uma vez declarado o tipo da variável, não se pode mais alterar.
- Exemplo: tipoDaVariável nomeDaVariável.
- Pode-se por praticidade no momento em que declara uma variável, já iniciá-la.

```
1 int idade;  
2 idade = 15;  
3  
4 int idade = 15;
```



Operadores

Operadores: +, -, *, /, %

- + : adição
- - : subtração
- * : multiplicação
- / : divisão



Operadores

Operadores: +, -, *, /, %

- + : adição
- - : subtração
- * : multiplicação
- / : divisão
- % : módulo (retorna o resto de uma divisão inteira)

```
1 int quatro = 2 + 2;
2 int cinco = 7 - 2;
3 int oito = 4 * 2;
4 int dezesesseis = 64 / 4;
5
6 int um = 5 % 2;
7 //cinco dividido por 2 resulta em 2 e tem resto 1;
8 //o operador % retorna o resto da divisão inteira;
```



Alguns valores são incompatíveis.

```
1 double pi = 3.1415;
2 int x = pi; // não compila
3
4 // O mesmo ocorre no seguinte trecho:
5 int pi = 3.1415; // não compila
6
7 // Isso é possível:
8
9 double d = 5; // double pode conter um número inteiro
10
11 int x = d; // não compila, inteiro não pode conter um double sem
    fazer cast (moldado).
```



Utilizando cast é possível, porém perde a precisão.

```
1
2 double d3 = 3.14;
3
4 int i = (int) d3; // (int) estamos moldando o número 3.14 para ser
   um inteiro..
```



Onde a variável pode ser referenciada.

```
1
2 // aqui a variável i não existe
3 int i = 5;
4 // a partir daqui ela existe
5
6 public void calcSalario(int A, double B){
7     // as variáveis A e B só podem ser referenciadas aqui dentro,
8     // salvo se houver outras variáveis A e B no contexto global.
9 }
10
11 // as variáveis A e B não podem ser referenciadas
12 // fora do escopo da função, pois elas não existem fora.
```



Onde a variável pode ser referenciada.

```
1
2 // aqui a variável i não existe
3 int i = 5;
4 // a partir daqui ela existe
5
6 while (condicao) {
7     // a variável i ainda vale aqui
8     int j = 7;
9     // a variável j passa a existir
10 }
11
12 // aqui a variável j não existe mais,
13 // mas a variável i continua dentro do escopo
```



Onde a variável pode ser referenciada.

```
1 class Escopo {  
2     int x;  
3     double y;  
4     boolean z;  
5 }  
6 // escopo global, as variáveis x, y e z podem ser referenciadas  
7 // em toda classe.
```



Métodos com retorno

```
1 public int somaDoisValores(int a, int b){
2     return a + b;
3 }
4 // quando chamamos o método somaDoisValores(...) e o atribuímos a
   // uma variável
5 // por ex.:
6 soma = somaDoisValores(10, 20);
7
8 // o java irá armazenar o valor do retorno na variável soma.
9 // lembrando que a variável que armazenará o retorno
10 // deve ser do mesmo tipo, salvo se fizer cast.
```



Métodos sem retorno

```
1 int soma = 0;
2
3 public void somaDoisValores(int a, int b){
4     soma = a + b;
5 }
6 // quando chamamos o método somaDoisValores(...)
7 // por ex.:
8 somaDoisValores(10, 20);
9
10 // o java irá fazer a soma e nesse caso salvar em soma.
```

