



ORACLE

Academy



Java Foundations

3-1

O que É uma Variável?

ORACLE
Academy



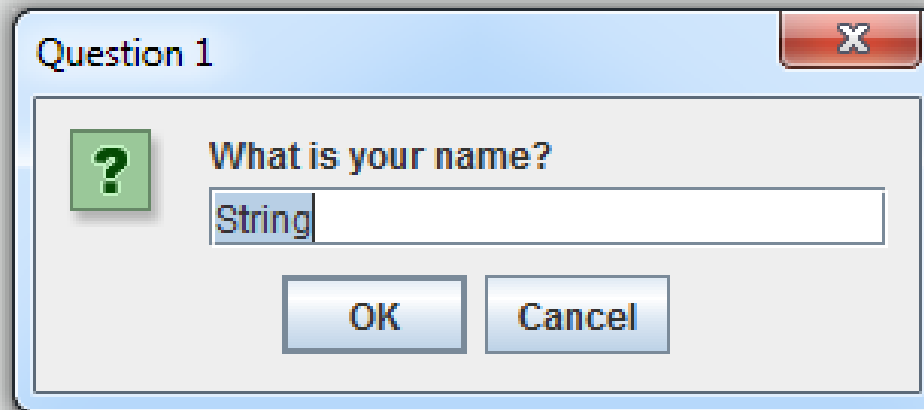
Objetivos

- Esta lição abrange os seguintes objetivos:
 - Entender as vantagens das variáveis
 - Identificar quatro tipos principais de variáveis:
 - (boolean, int, double, String)
 - Declarar e atribuir valores a variáveis
 - Nomear variáveis de acordo com as convenções



Exercício 1

- Execute o `JavaLibs.jar`
- Considere os tipos de dados que esse programa solicita



O Conjunto de Problemas 3 tem o objetivo de recriar esse programa com sua própria história. Esta seção ensina tudo o que você precisará para criá-lo

O que é uma Variável?

- Considere a variável x em uma equação
- Podemos atribuir qualquer valor a ela x

$$y = -2x + 5$$

$$x = 0$$

$$y = -2 \times 0 + 5$$

$$y = 0 + 5$$

$$y = 5$$

$$x = 2$$

$$y = -2 \times 2 + 5$$

$$y = -4 + 5$$

$$y = 1$$

O que É uma Variável em Java?

- Da mesma forma, podemos atribuir valores a variáveis Java
 - Escrever a linha `String x = "Alex";` é o mesmo que atribuir um valor "Alex" a uma variável x
 - Escrever `"Meu nome é " + x` é equivalente a escrever "Meu nome é Alex"

```
String x = "Alex";  
System.out.println("Meu nome é " + x);
```

"Meu nome é Alex"

Desvantagem de Não Usar Variáveis

- O código não é flexível
- Para substituir o nome "Alex", você precisa fazer muitas alterações em muitos locais:
 - Edição monótona
 - Risco de deixar passar um "Alex" despercebido

```
System.out.println("Meu nome é Alex");  
System.out.println("Alex é muito bacana");  
System.out.println("Viva o Alex!");  
System.out.println("Aproveite a Avaliação de Alex "  
    + "Dia! Meu nome é Alex. Sei como todos estão "  
    + " interessados em "  
    + "começar a apreciar o Alex no Dia da"  
    + "Avaliação do Alex! Alex, Alex, Alex! Sim "  
    + "Alex!!! Este sou eu! Alex é o melhor encontro de "  
    + "todos          os tempos!");
```




Vantagens de Usar Variáveis

- O código torna-se flexível
 - Lembrar e manipular variáveis
- Para substituir o nome "Alex", você faz uma alteração:
 - Edição eficiente
 - Sem risco de deixar passar um "Alex" despercebido

```
String x = "Sam";
System.out.println("Meu nome é " + x);
System.out.println(x + " é muito bacana!");
System.out.println("Viva " + x + "!");
System.out.println("Aproveite a Avaliação de " + x + " "
    + "Dia! Meu nome é " + x + ". Sei como todos estão "
    + "interessados em "
    + "começar a apreciar " + x + " no Dia da Avaliação de "
    + x + "! " + x + ", " + x + ", " + x + "! Sim "
    + x + "!!! Este sou eu! " + x + " é o melhor pretendente "
    + " de todos os tempos!");
```


Mais Vantagens com Variáveis

- Manipule valores muitas vezes de várias maneiras:
 - Mude você mesmo os valores diretamente (mostrado a seguir)
 - Altere valores calculados de forma programada
 - Mude com base na entrada do usuário

```
5    String x = "Alex";
6    x = "Sam";
7    x = "Nicky";
8    x = "Encontro Secreto";
9
10   "voltar" = x;    //Can't do this
```

Exercício 2

- Crie um novo projeto e adicione o arquivo `Variables02.java` a ele
- Siga as etapas no exercício
- Execute o programa entre cada etapa e observe a saída
- Seu programa deve gerar as seguintes saídas:

– Depois da Etapa 1)

```
puppy  
puppy
```

– Depois da Etapa 2)

```
kitty  
kitty
```

– Depois da Etapa 3)

```
kitty  
bunny
```

A Natureza dos Programas Linha por Linha

- A partir da linha 8, x sempre é igual a “kitty” até...
- A partir da linha 14 onde x sempre é igual a “bunny”

```
7 public static void main(String[] args) {  
8     String x = "kitty";  
9     System.out.println(x);           //prints "kitty"  
10    //...  
11    //...  
12    System.out.println(x);           //prints "kitty"  
13  
14    x = "bunny";  
15    //...  
16    //...  
17    System.out.println(x);           //prints "bunny"  
18    //...  
19  
20 }  
21 }
```



Muitos Tipos de Variáveis

- As variáveis podem existir para muitos tipos diferentes no Java
- Aqui estão as variáveis que você viu:

Tipo	Palavra-chave	Exemplo de Valores
Booleano	<code>boolean</code>	<code>true</code> , <code>false</code>
Inteiro	<code>int</code>	1, -10, 20000, 123_456_789
Duplo	<code>double</code>	1.0, -10.0005, 3.141
String	<code>String</code>	"Alex", "Jantei demais."

Declarando uma Variável

- Java é uma “linguagem que envolve muita digitação”
 - Você precisa declarar que tipo de dados sua variável tratará usando palavras-chave

tipo *nome*




```
boolean bool;  
int x;  
double y;  
String z;
```

- Depois de você declarar uma variável...
 - Essa variável passa a existir
 - Não é preciso declará-la novamente

Opções para Declarar e Atribuir Valores

- Declare e atribua uma variável em uma única linha

tipo *nome* *valor*

`boolean bool = true;`

- Declare uma variável em uma linha e atribua um valor mais tarde

```
boolean bool;
```

```
bool = true;
```

Atribuindo Valores Inadequados

- Os valores atribuídos devem ser apropriados para o tipo de dados que você declarou



```
int x = 3;
```



```
int z = "Puppies!";
```


Valores Matemáticos Inapropriados

- Podemos atribuir qualquer valor a x
- Não podemos atribuir um valor String a x
 - Isso não faz sentido!

$$y = -2x + 5$$

$x = \text{"Puppies!"}$

$y = -2(\text{"Puppies!"}) + 5$

$y = ???$



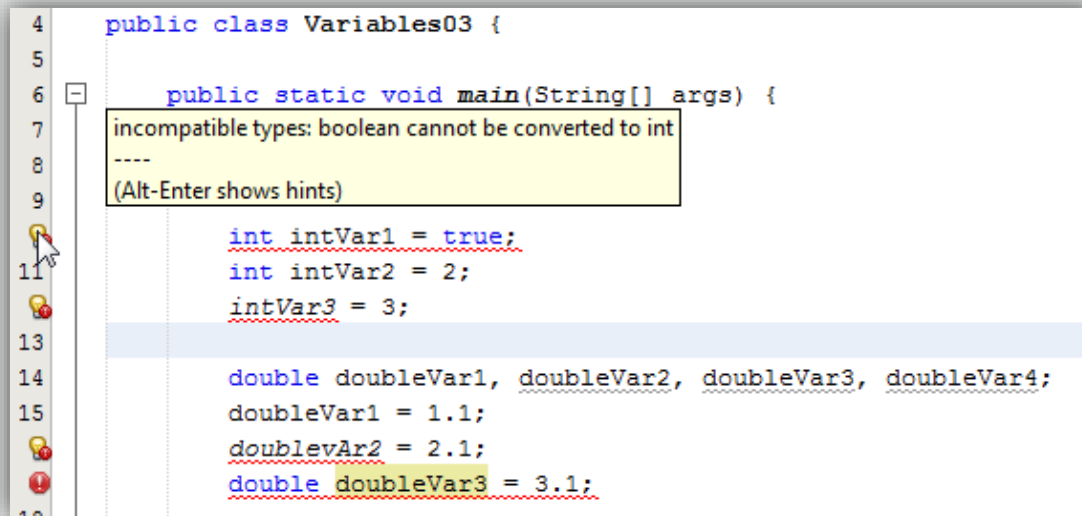
Exercício 3, Parte 1

- Crie um novo projeto e adicione o arquivo `Variables03.java` a ele
 - Existem seis erros neste programa
 - Você consegue corrigir esses erros para que o programa produza a saída a seguir?

```
bool = true
intVar1 = 1
intVar2 = 2
intVar3 = 3
doubleVar1 = 1.1
doubleVar2 = 2.1
doubleVar3 = 3.1
doubleVar4 = 4.1
stringVar1 = 11
stringVar2 = 22
```

Exercício 3, Dicas 1

- Seu IDE sublinha o código que está com problema
 - Para obter detalhes, mantenha o cursor sobre o código ou o ícone na margem esquerda
 - Seu IDE pode dar dicas de soluções possíveis
 - Clique no ícone na margem esquerda



```
4 public class Variables03 {
5
6     public static void main(String[] args) {
7         incompatible types: boolean cannot be converted to int
8         ----
9         (Alt-Enter shows hints)
10
11         int intVar1 = true;
12         int intVar2 = 2;
13         intVar3 = 3;
14
15         double doubleVar1, doubleVar2, doubleVar3, doubleVar4;
16         doubleVar1 = 1.1;
17         doubleVar2 = 2.1;
18         double doubleVar3 = 3.1;
19     }
20 }
```

Exercício 3, Dicas 2

- As soluções sugeridas por seu IDE às vezes são ruins
 - Não confie cegamente nessas soluções
- Sua própria capacidade de solucionar problemas pode ser um recurso excepcional



Erros com Variáveis

- Atribuindo valores inadequados a um tipo de variável

```
int intVar1 = true;
```

- Esquecendo de declarar um tipo de variável

```
intVar3 = 3;
```

- Escrevendo uma variável com a grafia incorreta

```
double doubleVar2;  
doublevAr2 = 2.1;    //O Java faz distinção  
                      //entre maiúsculas e  
                      //minúsculas
```

Erros com Variáveis

- Declarar a mesma variável duas vezes

```
double doubleVar3;  
double doubleVar3 = 3.1;
```

- Esquecer de atribuir um valor antes de usar uma variável

```
double doubleVar4;  
System.out.println(doubleVar4);
```

A atribuição de um valor inicial à variável denomina-se “inicialização”

Você Pode Ter Notado...

- É possível declarar muitas variáveis em uma única linha

```
double doubleVar1, doubleVar2, doubleVar3;
```

- É possível atribuir valores ao declarar muitas variáveis

```
double doubleVar1, doubleVar2, doubleVar3 = 3.1;
```

- É uma questão de preferência pessoal...
 - Declarar toda variável em linhas separadas
 - Declarar todas as variáveis de um tipo específico em uma única linha



Atribuição de Nomes Inadequados a Variáveis

- Você pode atribuir praticamente qualquer nome a uma variável

```
int dsfdsfspoop = 20;    //Ha ha!
```

- Isso seria divertido, mas...
- Será que, ao ler o código, você ou um amigo entenderia quais dados dsfdsfspoop representa?
- Em geral, não é recomendável o uso de nomes muito pequenos

```
int x = 20;
```

- Isso é muito útil para teste...
- E é uma prática comum em loops pequenos (assunto abordado mais adiante), mas...
- Será que, ao ler o código, você ou um amigo entenderia quais dados x representa?

Atribuição de Nomes Muito Inadequados a Variáveis



- As variáveis podem compartilhar o mesmo nome

```
int x = 20;  
double x = 22.0;  
System.out.println(x); //Que x?
```

- As variáveis não podem começar com números

```
boolean 1337Hacker = true;
```

- As palavras-chave não podem ser usadas para nomes de variáveis

```
int continue = 20;
```

- As palavras-chave ficam azuis no NetBeans
- As palavras-chave têm significados especiais no Java

Convenções de Nomenclatura de Variáveis



- Comece cada variável com uma letra minúscula
As palavras subsequentes devem ser capitalizadas:
 - myVariable
- Escolha nomes que sejam mnemônicos e que indiquem a intenção da variável para o observador casual
- Lembre-se de que...
 - Os nomes fazem distinção entre letras maiúsculas e minúsculas
 - Os nomes não podem incluir espaço em branco

```
int studentAge = 20;  
String myCatchPhrase = "Aproveite o Dia de Avaliação de Alex!";
```

Resumo

- Nesta lição, você deverá ter aprendido a:
 - Entender os benefícios de variáveis
 - Identificar quatro tipos principais de variáveis:
 - (boolean, int, double, String)
 - Declarar e atribuir valores a variáveis
 - Nomear variáveis de acordo com as convenções





ORACLE

Academy

