

Variáveis em Java

Maurício Linhares – mauricio.linhares@gmail.com

Variáveis em Java

```
Cachorro () :
```

```
new Cachorro ();  
outroCachorro.late ();
```

Não pode!

Pode!

Declaração de variáveis

- ~~As variáveis devem ter um tipo;~~
- As variáveis devem ter um nome;

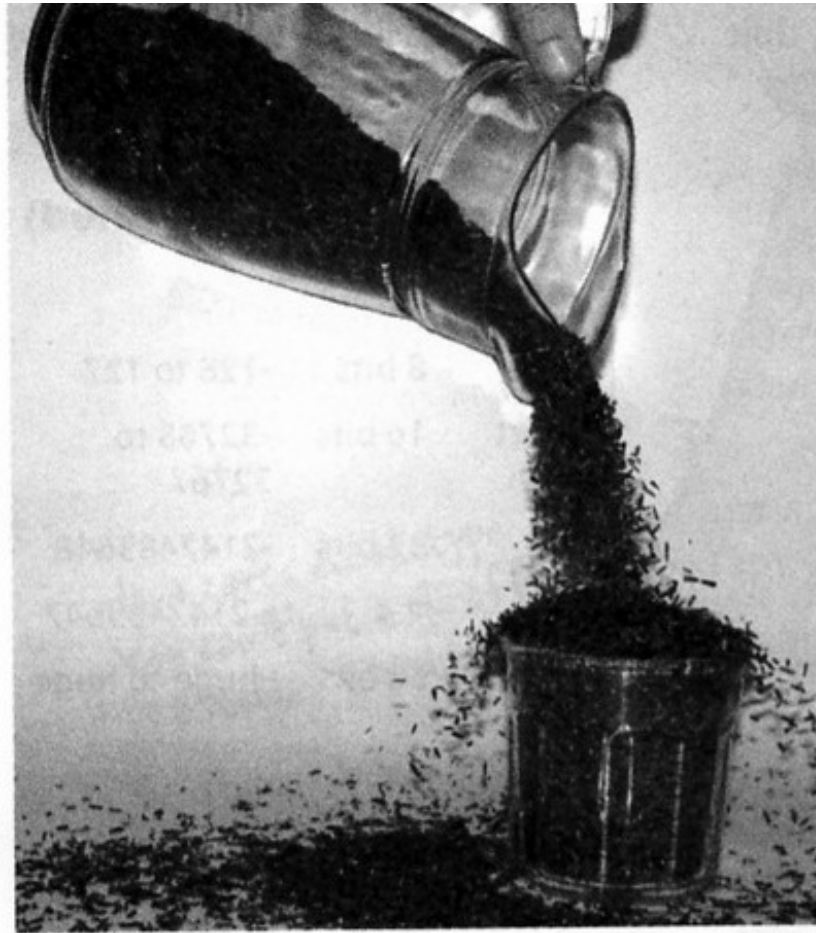
Nomes de identificadores do Java

- Devem começar com uma letra, ‘_’ ou ‘\$’ (mas nós não devemos usar \$)
- Não podem ser uma palavra reservada da linguagem Java;

reservadas da linguagem

package synchronized boolean do if private default goto
break double implements protected throw byte
import public throws case enum instanceof else
transient catch extends int short instanceof return
interface static void class finally long char final
volatile const float native super while strictfp
volatile const float native super while

O que é uma variável?



Atribuição de variáveis

```
public class TesteDeAtribuicoes {  
  
    public static void main(String[] args) {  
        int valor = 10;  
        boolean verdadeiro = false;  
        int soma = valor + 8;  
        int outroValor = valor;  
        double numeroEstranho = 234.987;  
        char caractere = 'P';  
    }  
}
```

Tamanhos dos tipos primitivos

boolean	1 bit	true ou false
character	16 bits	0 a 65535
byte	8 bits	-128 a 127
short	16 bits	-32768 a 32767
int	32 bits	-2147483648 a 2147483647
long	64 bits	Não cabe aqui 😊
float	32 bits	-
double	64 bits	-

Declarando inteiros

```
public class DeclarandoInteiros {  
  
    public static void main(String[] args)  
    {  
        int inteiro = 1;  
        short pequeno = -20;  
        long grande = 123455;  
        long outroLong = 123456663345L;  
    }  
}
```

Declarando números de ponto flutuante

```
public class DeclarandoFloats {  
  
    public static void main(String[] args)  
    {  
        float preco = 3.34F;  
        double medida = 12342.45566;  
    }  
}
```

Declarando caracteres

```
public class DeclarandoCaracteres {  
  
    public static void main(String[] args)  
    {  
        char character = 'Y';  
        char unicode = '\\u0059';  
    }  
  
}
```

Perda de precisão

- Ao tentar colocar um número grande em uma variável que não é grande o suficiente, perdemos precisão;
- O compilador não aceita a perda de precisão a não ser que o programador se responsabilize por ela;
- Você não vai querer isso em um sistema que lide com dinheiro;

Perda de precisão

```
public class PrecisaoTest {  
  
    public static void main(String[] args) {  
        int grande = 40000;  
        short menor = (short) grande;  
        //correto  
  
        short pequeno = grande;  
        //errado  
    }  
}
```

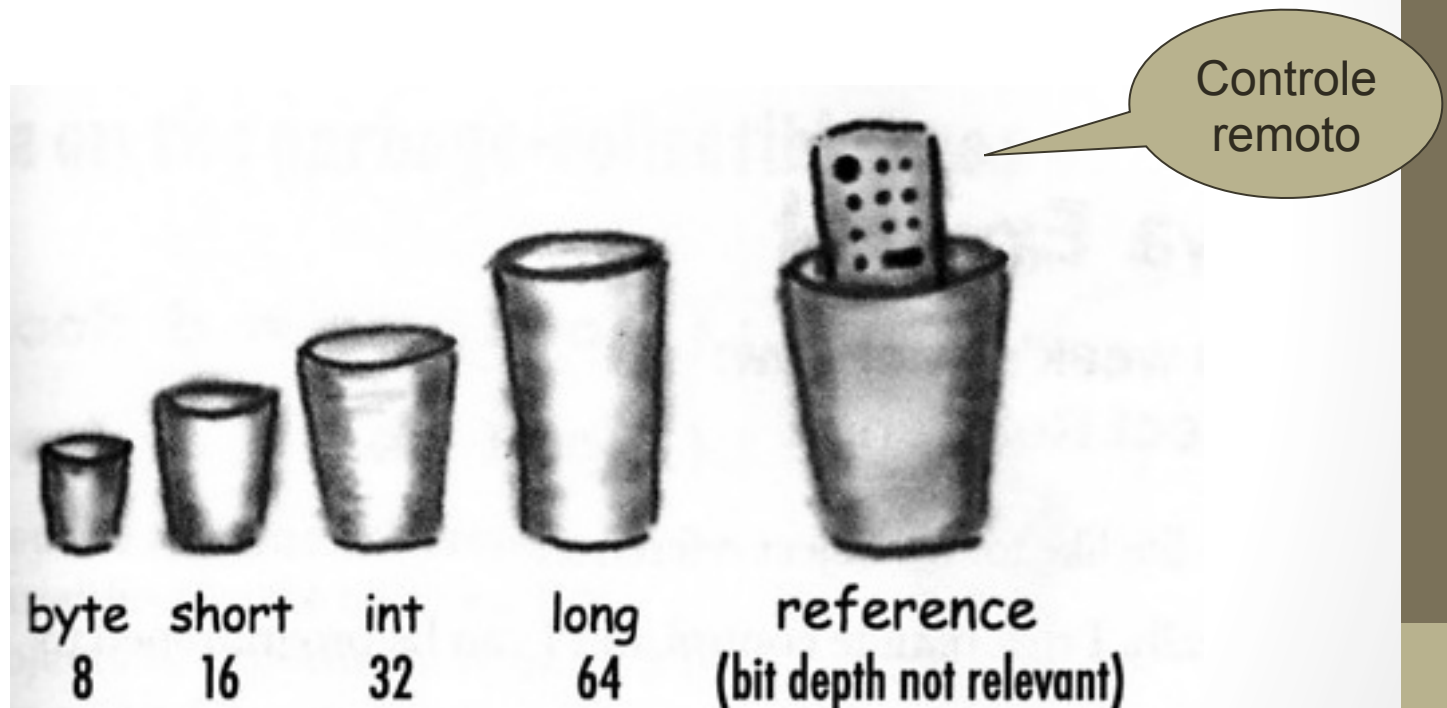
Matemática de ponto flutuante no Java

- Operações matemáticas com ponto flutuante no Java não são precisas;
- $1.01 + 1.17$ não é $2.18 \rightarrow 2.179999999999999997$;
- Não use números flutuantes para matemática exata no Java (dinheiro? dos outros? Nem pensar!);

BigDecimal e matemática precisa no Java

```
public class MatematicaComBigDecimal {  
  
    public static void main(String[] args) {  
        BigDecimal numero = new BigDecimal("10.7");  
        BigDecimal outroNumero = new BigDecimal("9.5");  
  
        System.out.println( numero.add( outroNumero ) );  
    }  
  
}
```

Variáveis que apontam pra objetos





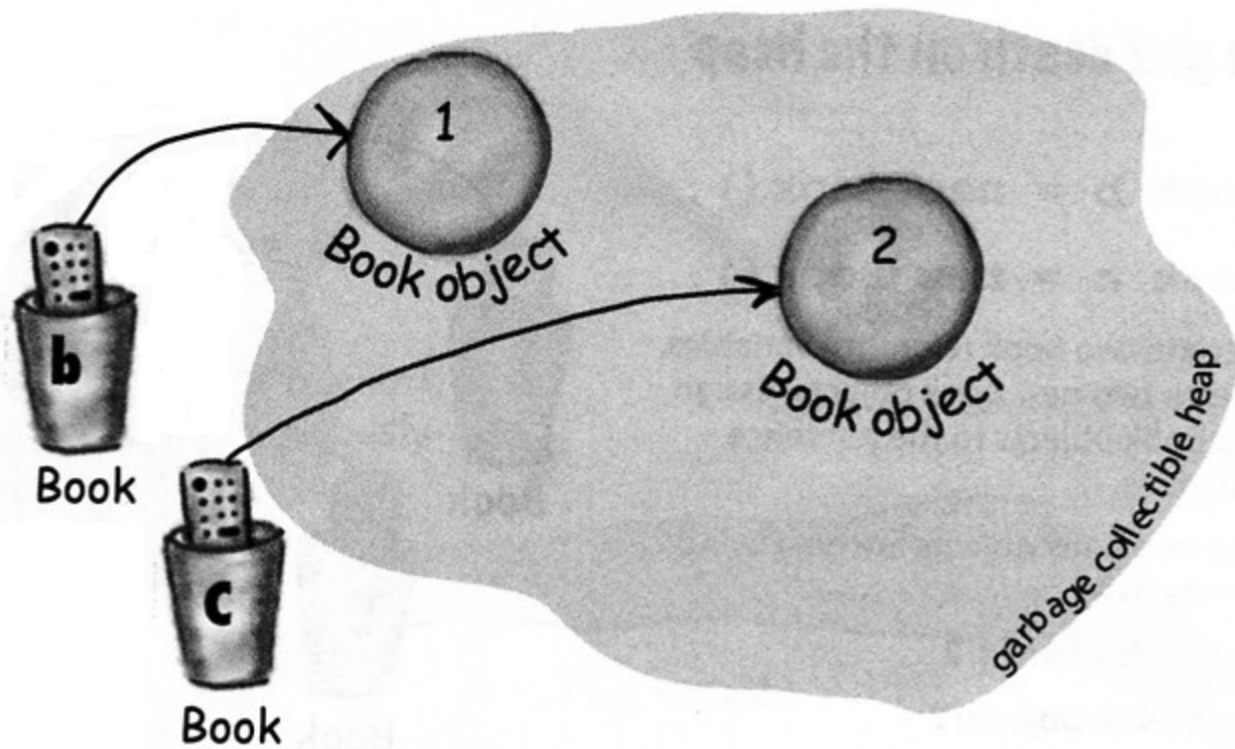
```
Dog d = new Dog();  
d.bark();
```

Primitivos e objetos

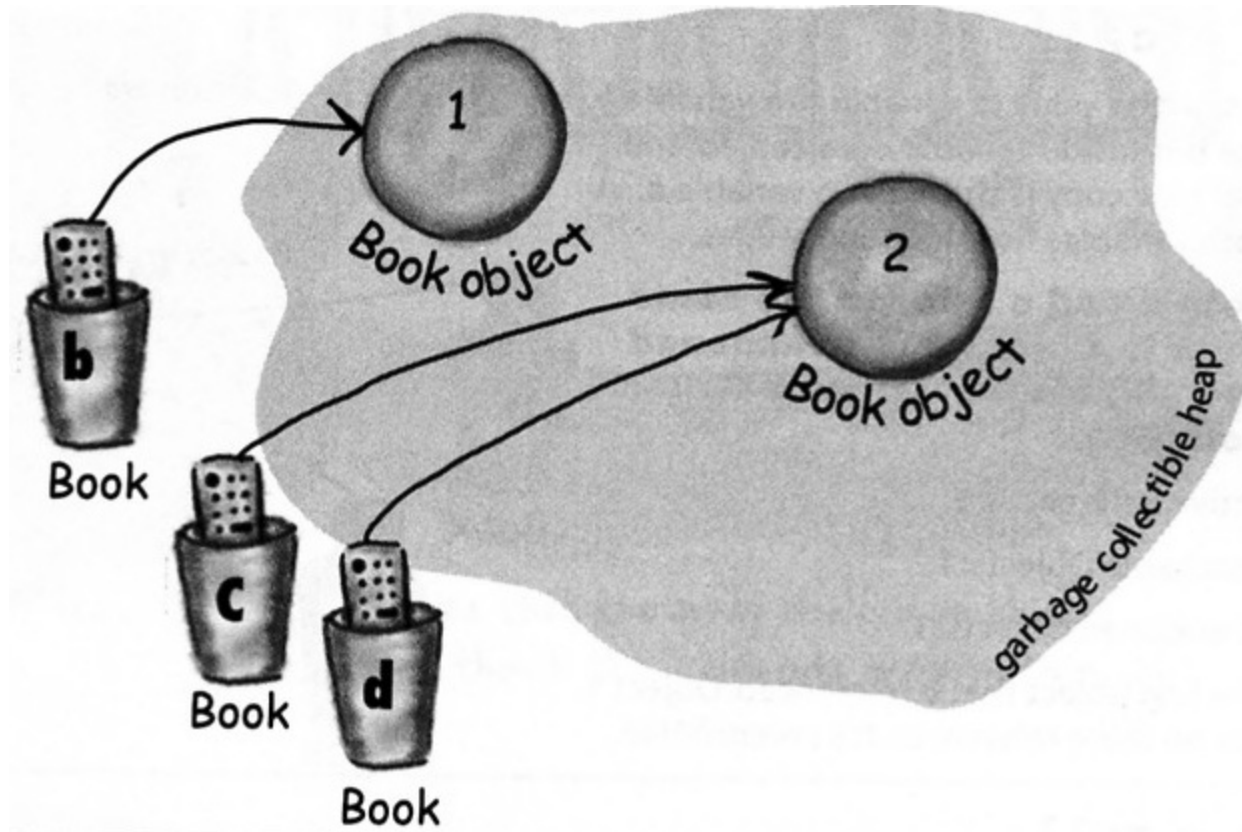
- Variáveis de tipos primitivos guardam o valor do tipo primitivo;

• Variáveis de tipos primitivos guardam o valor do tipo primitivo; variáveis de objetos guardam o caminho pra se chegar no objeto (o controle remoto);

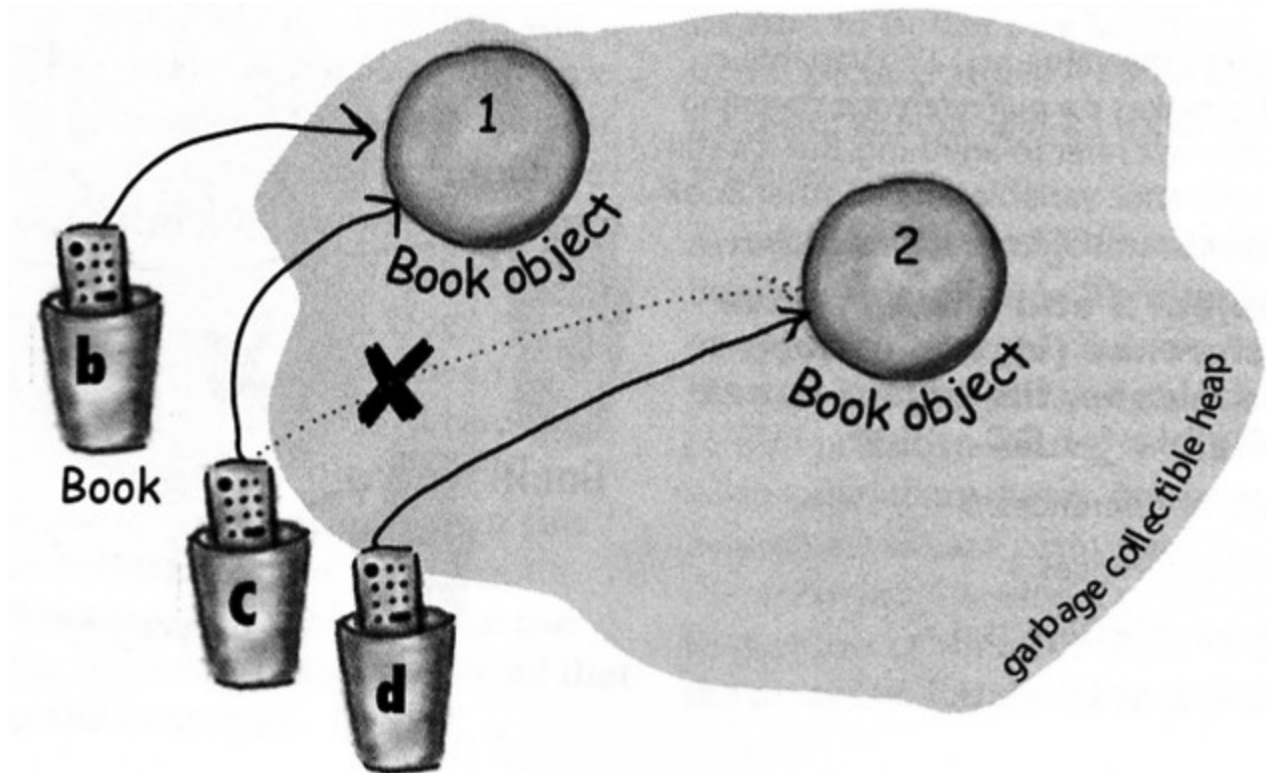
Referências



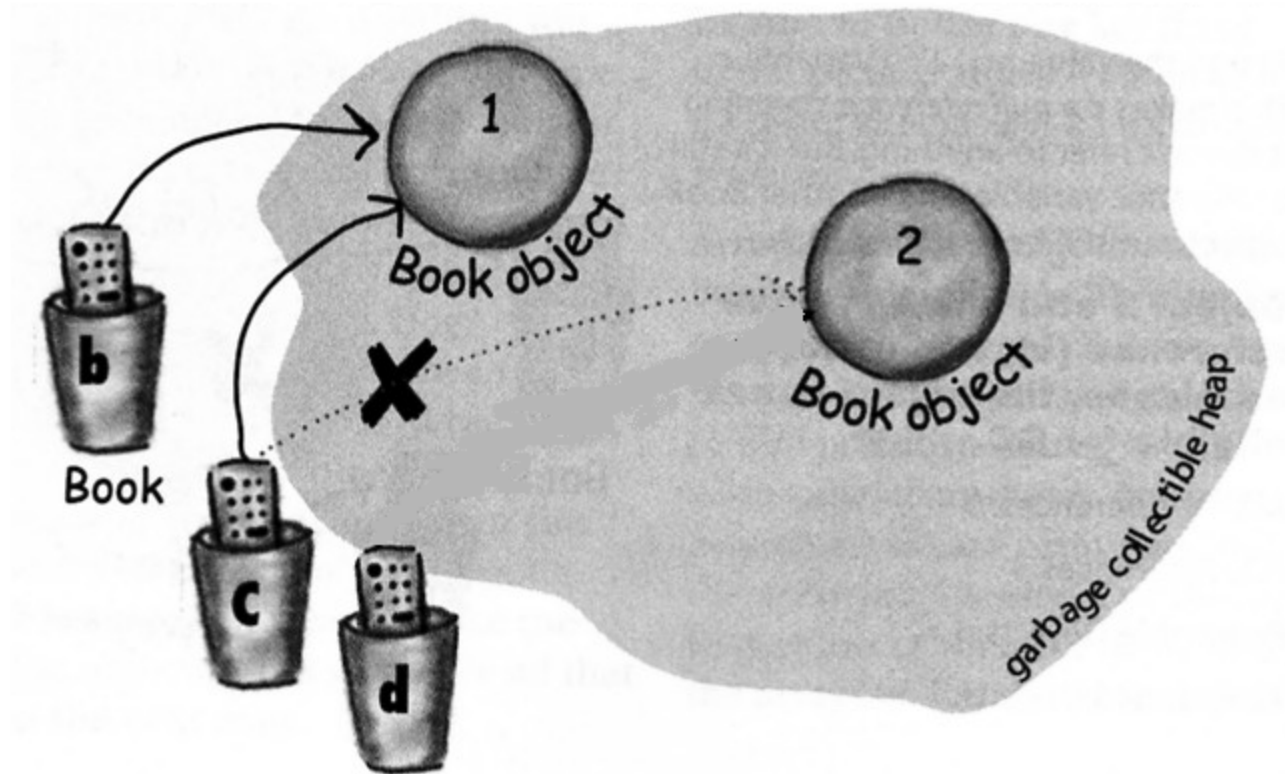
Uma nova referência



E uma troca de referências



E agora a televisão está fora do ar



Exercício - 1

- Defina a diferença entre objeto e variável
- Defina a diferença entre tipos primitivos e objetos dentro da linguagem Java

Exercício - 2

- Defina o que fazem as seguintes palavras reservadas do Java e demonstre o seu uso com exemplos em código:
 - strictfp
 - final
 - break
 - continue