

Programação Orientada a Objetos
SANTOS, Rafael

Capítulo 2 – Criando classes em Java

Sintaxe Básica

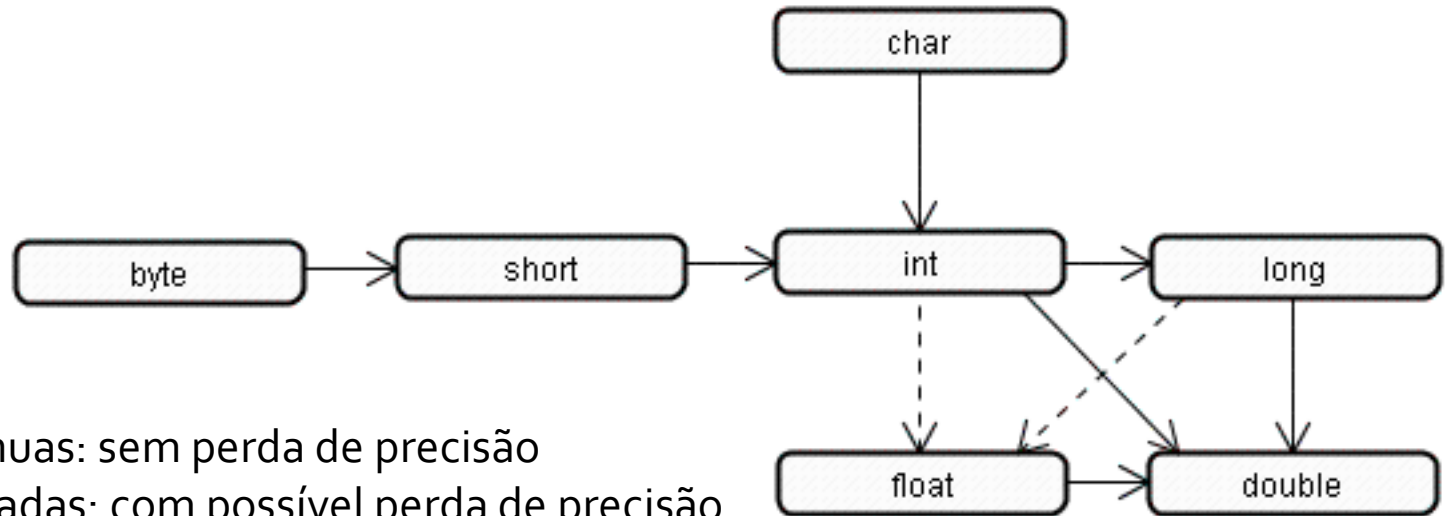
- Palavra-chave **class**
- Nome iniciando com **letra maiúscula**
- Corpo delimitado por **chaves**
- Ver classe:
 - Vazia (Fig. 2.1)

Campos em classe Java

- Implementados por **variáveis declaradas** na classe
- Possuem um **nome (identificador)**
- Possuem um **tipo** (primitivo ou classe)
- Ver classes:
 - RegistroAcademicoSimples (Fig. 2.2)
 - DataSemMetodos (Fig. 2.3)

Tipos primitivos

- byte, short, int, long
- float, double
- boolean
- char



Setas contínuas: sem perda de precisão

Setas tracejadas: com possível perda de precisão

Sentido contrário: exige-se casting explícito

Nomenclatura

- Identificadores formados por **letras**, **números**, **\$**, **_**
- **Não** pode iniciar com **números**
- Nome de **classe** iniciam com **letra maiúscula**
- Nomes de **variáveis** e de **métodos** iniciam com **letra minúscula**
- CamelCase

Métodos em classe Java

- Possuem um **nome (identificador)**
- **Tipo de retorno:** quando **void**, não há retorno, não sendo exigida a instrução **return** no corpo do método
- **Parâmetros** entre parênteses
- **Corpo**, onde são inseridas as instruções, é delimitado por **chaves**
- Ver classe:
 - DataSimples (Fig. 2.4)

Escopo de Variáveis

- Variável de **instância** (ou de **objeto**)
- Variável **local** ao método
- Variável **local** declaradas **como parâmetro**
- Ver classe:
 - Triangulo (Fig. 2.5)

Modificadores e níveis de acesso

- 3 modificadores, 4 níveis:
 - default (ou de pacote)
 - public
 - private
 - protected (default + subclasses)
- Ver classes:
 - DemoDataSimples (Fig. 2.6)
 - Data (Fig. 2.7)
 - DemoData (Fig. 2.8)

Padrão Javabeans

- Variáveis de instância privadas
- Métodos **capturadores** (get) públicos
- Métodos **configuradores** (set) públicos
- Apoio dado pelas **IDEs**

Exercício 7: Encontre os Erros

```
class EXDoisValores {  
    int valor1,valor2;  
  
    int maior() {  
        if (valor1 > valor2)  
            return true;  
        else return false;  
    }  
  
    void menor() {  
        if (valor1 < valor2)  
            return valor1;  
        else return valor2;  
    }  
} // fim da classe
```

Exercício 9: Encontre os Erros

```
public class ExNumeroComplexo {  
    float real, imaginário;  
  
    float valor() {  
        return real, imaginário;  
    }  
}
```

Exercícios sugeridos para sala

- 2.29
- 2.39
- 2.40
- 2.42
- 2.43
- 2.48
- 2.49

Trabalho de Pesquisa

- As linguagens de programação, como Java, usam posições de memória, denominadas variáveis, para armazenar dados que são tratados pelos programas. Pesquise, conceitue e dê exemplos de variáveis: locais aos métodos, definidas como parâmetros de métodos e de instância (ou de objeto)
- Discorra ainda sobre como os modificadores de acesso `private`, `public` e a ausência deles, afetam as variáveis, indicando em quais das categorias de variáveis acima podem ser usados e o porquê.