

Programação Orientada a Objetos

ITE - 003

profº Mauricio Conceição Mario

Conceitos de Orientação a Objetos

- **Herança:** ocorre quando uma classe passa a herdar características (atributos e métodos) definidas em outra classe, especificada como *ancestral* ou *superclasse*. A classe receptora de recursos é denominada subclasse.

Encapsulamento: disponibilização de uma interface pública para manipular os estados e executar as operações de um objeto. Os atributos de um objeto podem ser escondidos de outros objetos por uma interface pública de métodos, de modo a impedir acessos indevidos.

No encapsulamento os métodos podem ser definidos, através dos modificadores de acesso, como:

- `private;`
- `protected;`
- `public;`
- `static.`
- atributo disponível para consulta: método `getnomeatributo()` [accessor method].
- alteração de atributo: `setnomeatributo()` [mutator method].
- Métodos que executam operações ou funções: [workes methods].

Referência `this`: é a referência do próprio objeto, representa a localização em memória na qual o objeto recorrente foi instanciado.

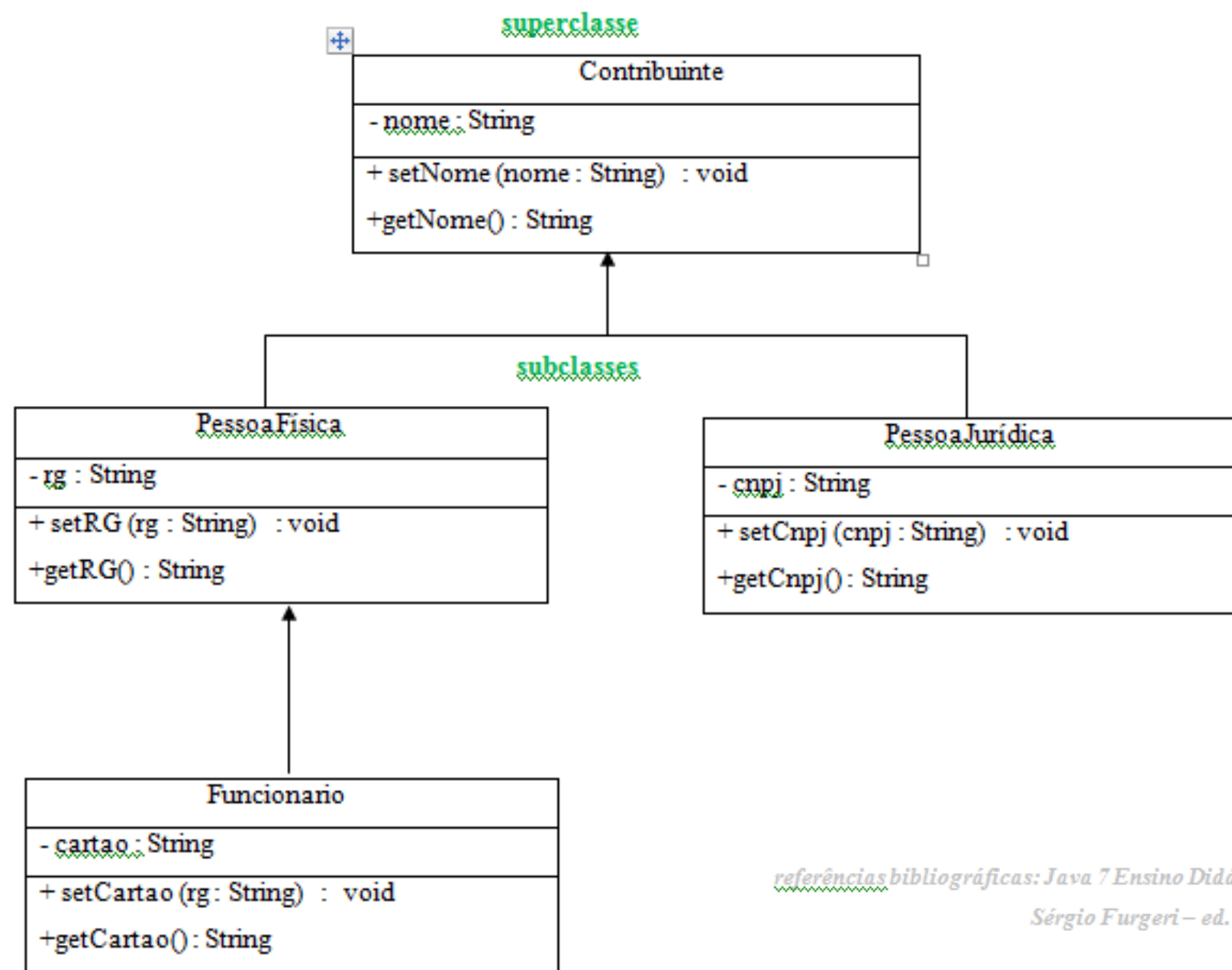
exemplo: `this.objeto = objeto;`

atributo de instância parâmetro

Requisitos da aplicação

- 1. Um funcionário da Receita pode inserir o nome de um contribuinte.
- 2. Um funcionário da Receita pode inserir o RG ou o CNPJ de um contribuinte.
- 3. Um funcionário da Receita pode inserir a identificação do cartão de um contribuinte.
- 4. Um funcionário da Receita pode consultar o nome de um contribuinte.
- 5. Um funcionário da Receita pode consultar o RG ou o CNPJ de um contribuinte.
- 6. Um funcionário da Receita pode consultar a identificação do cartão de um contribuinte.

Representação em diagrama de classes UML (Unified Modeling Language):



referências bibliográficas: Java 7 Ensino Didático

Sérgio Furgeri – ed. Erica

Classes

```
package heranca;  
public class Contribuinte {  
    private String nome;  
    public void setNome (String nome) {  
        this.nome = nome;  
    }  
    public String getNome() {  
        return (nome);  
    }  
}
```

```
package heranca;  
public class PessoaFisica extends Contribuinte {  
    private String rg;  
    public void setRG (String rg) {  
        this.rg = rg;  
    }  
    public String getRG() {  
        return (rg);  
    }  
}
```

```
package heranca;  
public class PessoaJuridica extends Contribuinte {  
    private String cnpj;  
    public void setCnpj (String cnpj) {  
        this.cnpj = cnpj;  
    }  
    public String getCnpj() {  
        return (cnpj);  
    }  
}
```

```
package heranca;  
public class Funcionario extends PessoaFisica {  
    private String cartao;  
    public void setCartao (String cartao) {  
        this.cartao = cartao;  
    }  
    public String getCartao() {  
        return (cartao);  
    }  
}
```

Classes

```
package heranca;

import javax.swing.JOptionPane;

public class UsaFuncionario {

    public static void main (String args[]){
        //OBJETOS
        Funcionario funcionario = new Funcionario();
        PessoaJuridica func = new PessoaJuridica();

        String u,v,t,z;
        int x;

        z = JOptionPane.showInputDialog ("digite 1 se
contribuinte é pessoa fisica \n digite 2 se é pessoa jurídica");
        //conversão de string para flutuante:
        x = Integer.parseInt( z );
        if (z.length()==1) {
            switch (x) {
                case 1: {
                    u = JOptionPane.showInputDialog ("escrever o nome do
contribuinte: ");
                    v = JOptionPane.showInputDialog ("escrever o RG do
contribuinte: ");
                    t = JOptionPane.showInputDialog ("escrever a
identificação do cartão do contribuinte: ");

                    funcionario.setNome(u);
                    funcionario.setRG(v);
                    funcionario.setCartao(t);
                    System.out.println("NOME: " + funcionario.getNome());
                    System.out.println("RG: " + funcionario.getRG());
                    System.out.println("CARTÃO: " + funcionario.getCartao());
                }
            }
        }
    }
}
```

Classes

```
        .....  
        break;  
    }  
    case 2: {  
        u = JOptionPane.showInputDialog ("escrever o nome do  
contribuinte: ");  
        v = JOptionPane.showInputDialog ("escrever o CNPJ do  
contribuinte: ");  
        t = JOptionPane.showInputDialog ("escrever a identificação do  
cartão do contribuinte: ");  
  
        funcionario.setNome(u);  
        func.setCnpj(v);  
        funcionario.setCartao(t);  
        System.out.println("NOME: " + funcionario.getNome());  
        System.out.println("CNPJ: " + func.getCnpj());  
        System.out.println("CARTÃO: " + funcionario.getCartao());  
        break;  
    }  
    } //switch  
} //if  
} //void main()  
}
```

profº Mauricio Conceição Mario

referências bibliográficas: Java 7 Ensino Didático

Sérgio Furgeri – ed. Érica

Classes

```
package heranca;

public class UsaFuncionario {

    public static void main (String args[]){

        Funcionario funcionario = new Funcionario();
        funcionario.setNome("Juliana Matheus");
        funcionario.setRG("25.444.778-x");
        funcionario.setCartao("MGH329");
        System.out.println(funcionario.getNome());
        System.out.println(funcionario.getRG());
        System.out.println(funcionario.getCartao());
    }
}
```

package = pacote: em Java é um diretório em que está armazenado o conjunto de classes com o mesmo propósito ou afinidades.

prof *Mauricio Conceição Mario*

referências bibliográficas: *Java 7 Ensino Didático*

Sérgio Furgeri – ed. Érica

Resultados

Entrada [X]

? digite 1 se contribuinte é pessoa fisica
digite 2 se é pessoa jurídica

1

OK Cancelar

Entrada [X]

? escrever o RG do contribuinte:

12987765-0

OK Cancelar

Entrada [X]

? escrever o nome do contribuinte:

Juliana

OK Cancelar

Entrada [X]

? escrever a identificação do cartão do contribuinte:

NHG34

OK Cancelar

```
Problems Javadoc Declaration Console X
<terminated> UsaFuncionario [Java Application] C:\Program Files\Java\jre1.8.0_20\bin\javaw.exe (19/02/2015 21:43:37)
NOME: Juliana
RG: 12987765-0
CARTÃO: NHG34
```

Roteiro para trabalho - Herança:

1. Criar pasta para abrigar o trabalho;
2. Criar dentro da pasta, projeto java (IDE Eclipse);
3. Criar as classes Contribuinte, PessoaFisica, PessoaJurídica, Funcionario e UsaFuncionario.

package = pacote: em Java é um diretório em que está armazenado o conjunto de classes com o mesmo propósito ou a finalidades.
4. Compilar e executar (as classes executáveis) o projeto. Verificar os resultados;
5. Identificar a superclasse, as subclasses e a classe que se utiliza da generalização através dos atributos e métodos;
6. Quais os tipos de encapsulamento (qualificador) dos atributos?
7. Quais os tipos de qualificadores dos métodos?
8. Descrever os formatos dos métodos get..() e set..() (se retomam valores, tipos de argumentos, funcionalidades, etc);
9. Para que está sendo utilizado o operador this?
10. Qual o significado e funcionalidade de package em java?

11. Qual o significado de UML?

12. Na representação UML, o que representa o nome do primeiro campo?

13. O que estão contidos no segundo campo? Qual o formato?

14. O que estão contidos no segundo campo? Qual o formato?

15. Qual o significado dos sinais + e – no terceiro campo?

Campos do formato UML para as questões 12, 13 e 14:

12.
13.
14.