

# Programação Orientada a Objetos

## Herança de Classes de Objetos

### Exercícios



Joinville Batista Junior

### Exemplo

```
package util;

import java.util.Calendar;
import java.util.GregorianCalendar;

public class Data {

    private int dia, mês, ano;

    public Data(int dia, int mês, int ano) {
        this.dia = dia;
        this.mês = mês;
        this.ano = ano;
    }

    public int getDia() {
        return dia;
    }
}
```

## Exemplo

```
public int getMês() {
    return mês;
}

public int getAno() {
    return ano;
}

public int calcularIdade () {
    GregorianCalendar data_atualGregorianCalendar = new GregorianCalendar();
    int dia_atual = data_atualGregorianCalendar.get(Calendar.DAY_OF_MONTH);
    int mês_atual = data_atualGregorianCalendar.get(Calendar.MONTH) + 1;
    // GregorianCalendar: mês varia de 0 a 11
    int ano_atual = data_atualGregorianCalendar.get(Calendar.YEAR);
    int idade = ano_atual - ano;
    if ((mês_atual < mês) || ((mês_atual == mês) && (dia_atual < dia)))
        idade--;
    return idade;
}
```

## Exemplo

```
public int compareTo(Data data) {
    if (ano > data.getAno()) return 1;
    if (ano < data.getAno()) return -1;
    if (mês > data.getMês()) return 1;
    if (mês < data.getMês()) return -1;
    if (dia > data.getDia()) return 1;
    if (dia < data.getDia()) return -1;
    return 0;
}

public String toString() {
    String data;
    if (dia < 10) data = "0" + dia; else data = "" + dia;
    if (mês < 10) data += "/0" + mês + "/"; else data += "/" + mês + "/";
    return data += ano;
}
}
```

## Exemplo

```
package entidade;

public class Obra {

    protected String título;
    protected String autor;

    public Obra(String título, String autor) {
        this.título = título;
        this.autor = autor;
    }

    public String getTítulo() {
        return título;
    }

    public String getAutor() {
        return autor;
    }

    public String toString () {
        return "Livro: " + título + "\n Autor: " + autor;
    }

}
```

## Exemplo

```
package entidade;

public class Livro extends Obra {

    private String editora;
    private int edição;
    private int ano_publicação;

    public Livro(String título, String autor, String editora, int edição, int ano_publicação) {
        super(título, autor);
        this.editora = editora;
        this.edição = edição;
        this.ano_publicação = ano_publicação;
    }

}
```

## Exemplo

```
public String getEditora() {
    return editora;
}

public int getEdição() {
    return edição;
}

public int getAnoPublicação() {
    return ano_publicação;
}

public String toString () {
    return "Livro: " + título + "\n Autor: " + autor + "\n Editora: " + editora
        + "\n Edição: " + edição + "\n Ano de Publicação: " + ano_publicação;
}
}
```

## Exemplo

```
package entidade;

import util.Data;

public class RelatórioTécnico extends Obra {

    public enum GrauSigilo {confidencial, reservado, público};

    String empresa;
    GrauSigilo grau_sigilo;
    Data relatórioData;

    public RelatórioTécnico(String título, String autor, String empresa,
        GrauSigilo grau_sigilo, Data relatórioData) {
        super(título, autor);
        this.empresa = empresa;
        this.grau_sigilo = grau_sigilo;
        this.relatórioData = relatórioData;
    }
}
```

## Exemplo

```
public String getEmpresa() {
    return empresa;
}

public GrauSigilo getGrauSigilo() {
    return grau_sigilo;
}

public Data getData() {
    return relatórioData;
}

public String toString () {
    return "Relatório Técnico: " + título + "\n Autor: " + autor
        + "\n Empresa: " + empresa + "\n Grau de Sigilo: " + grau_sigilo
        + "\n Data: " + relatórioData;
}
}
```

## Exemplo

```
package entidade;

import util.Data;

public class Tese extends Obra {

    private String universidade, orientador;
    private Data teseData;

    public Tese(String título, String autor, String universidade, String orientador,
        Data teseData) {
        super(título, autor);
        this.universidade = universidade;
        this.orientador = orientador;
        this.teseData = teseData;
    }
}
```

## Exemplo

```
public String getUniversidade() {
    return universidade;
}

public String getOrientador() {
    return orientador;
}

public Data getData() {
    return teseData;
}

public String toString () {
    return "Tese: " + título + "\n Autor: " + autor
        + "\n Universidade: " + universidade
        + "\n Orientador: " + orientador + "\n Data: " + teseData;
}
}
```

## Exemplo

```
package controle;

import entidade.Livro;
import entidade.Obra;
import entidade.RelatórioTécnico;
import entidade.RelatórioTécnico.GrauSigilo;
import entidade.Tese;
import java.util.ArrayList;
import util.Data;

public class Biblioteca {

    private static ArrayList<Obra> listaObra = new ArrayList();

    public static void main(String[] args) {
        criarObras ();
        imprimirObras ("Obras Cadastradas:", listaObra);
        imprimirObras ("Obras Selecionadas:", selecionarObras());
    }
}
```

## Exemplo

```
private static void criarObras () {
    listaObra.add(new Livro ("Core Java", "Horstman - Cornell",
        "Sun Microsystems Press", 1, 1999));
    listaObra.add(new Tese ("Desenvolvimento de Aplicações Críticas",
        "Paulo Oliveira", "Unicamp", "Roberto Sobral", new Data(29,3,2007)));
    listaObra.add(new RelatórioTécnico ("Java para Software Embarcado",
        "Adriana Messias", "Software Design", GrauSigilo.público, new Data(30,9,2012)));
}
```

## Exemplo

```
public static ArrayList<Obra> selecionarObras() {
    String string_inicial_titulo = null;
    int menor_edição = -1;
    int menor_ano_publicação = -1;
    GrauSigilo grau_sigilo = null;
    String string_inicial_orientador = null;
    Data menorData = null;
    System.out.println("Filtros de Obras:")
        + "\n - string inicial do título da obra: " + string_inicial_titulo
        + "\n - menor edição do livro: " + menor_edição
        + "\n - menor ano de publicação do livro: " + menor_ano_publicação
        + "\n - grau de sigilo do relatório técnico: " + grau_sigilo
        + "\n - string inicial do nome do orientador da tese: " + string_inicial_orientador
        + "\n - menor data da tese ou relatório técnico: " + menorData + "\n");
    return filtrarObras(string_inicial_titulo, menor_edição, menor_ano_publicação,
        grau_sigilo, string_inicial_orientador, menorData);
}
```

## Exemplo

```
private static ArrayList<Obra> filtrarObras(String string_inicial_titulo,
    int menor_edição, int menor_ano_publicação, GrauSigilo grau_sigilo,
    String string_inicial_orientador, Data menorData) {
    ArrayList<Obra> selecionadas_listaObra = new ArrayList();
    for (Obra itemObra : listaObra) {
        if ((string_inicial_titulo != null)
            && !itemObra.getTítulo().startsWith(string_inicial_titulo)) continue;
        if (itemObra instanceof Livro) {
            Livro itemLivro = (Livro) itemObra;
            if ((menor_edição != -1) && (itemLivro.getEdição() < menor_edição)) continue;
            if ((menor_ano_publicação != -1)
                && (itemLivro.getAnoPublicação() < menor_ano_publicação)) continue;
        } else if (itemObra instanceof RelatórioTécnico) {
            RelatórioTécnico itemRelatórioTécnico = (RelatórioTécnico) itemObra;
            if ((grau_sigilo != null)
                && (itemRelatórioTécnico.getGrauSigilo() != grau_sigilo)) continue;
            if ((menorData != null)
                && (itemRelatórioTécnico.getData().compareTo(menorData) < 0)) continue;
        }
    }
    selecionadas_listaObra.add(itemObra);
}
return selecionadas_listaObra;
}
```

## Exemplo

```
    } else if (itemObra instanceof Tese) {
        Tese itemTese = (Tese) itemObra;
        if ((string_inicial_orientador != null)
            && !itemTese.getOrientador().startsWith(string_inicial_orientador)) continue;
        if ((menorData != null)
            && (itemTese.getData().compareTo(menorData) < 0)) continue;
    }
    selecionadas_listaObra.add(itemObra);
}
return selecionadas_listaObra;
}
```



## Exemplo

```
private static void imprimirObras (String cabeçalho, ArrayList<Obra> listaObra) {  
    System.out.println(cabeçalho);  
    for (Obra itemObra : listaObra) {  
        System.out.println (itemObra. toString());  
    }  
    System.out.println();  
}  
  
}
```

## Exemplo

3.1 – Adapte o exemplo da apostila:

- a) substituindo as classes Obra, Livro, RelatórioTécnico e Tese por Veículo (modelo, potência, ano, zero[sim, não]), Carro (combustível (gasolina, alcool, flex]), quatro\_portas[sim, não]), Caminhão (n\_eixos, capacidade[normal, bitrem, treminhão]) e Moto(n\_cilindradas);
- b) substituindo os filtros de seleção por: ano\_mínimo; zero, combustível, n\_eixos\_máximo, n\_cilindradas\_mínimo;
- c) substituindo a classe Biblioteca por VendaVeículos e adaptando o nome dos métodos.

3.2 – Adapte o exercício 3.1:

- a) acrescentando a classe Venda (valor, data, cliente, veículo) e classe Cliente (nome, CPF, telefone) e adaptando os métodos de criação e impressão para vendas;
- b) adapte os nomes dos métodos de seleção e filtros para vendas, e acrescente os filtros adicionais: valor\_mínimo e valor\_máximo.

## Exemplo

3.3 – Adapte o o exercício 3.2:

- a) substituindo as entidades por classes de sua escolha, utilizando pelo menos: um dado do tipo enumerado e um dado booleano em uma das classes;
- b) e definindo os filtros que julgar necessário, incluído pelo menos um filtro booleano e um enumerado.