





Compartilhar o seu link com: [luciorocha @ professores.utfpr.edu.br](mailto:luciorocha@professores.utfpr.edu.br)

Leitura recomendada: Apostila de Interfaces e Classes Abstratas:  Apostila_cap11_poo.tif


Leitura recomendada: Apostila de Classes Internas e Classes Anônimas:

 Apostila_cap13_poo.tif

João Pedro de Paula:<  Cópia de Aula 13 - POCO4A - Exercícios propostos >

Isabella Melo Almeida  Cópia de Aula 13 - POCO4A - Exercícios propostos

Fernando Rafael:  Exercício

Sefora<  Aula 13 - POCO4A >

Rafael Kendy Naramoto Lopes <  Cópia de Aula 13 - POCO4A - Exercícios propostos >

Pedro Reis e Thiago Tieghi  Cópia de Aula 13 - POCO4A - Exercícios propostos >

Filipe Augusto Parreira Almeida <  Cópia de Aula 13 - POCO4A - Exercícios propostos >

Gabriel Reis Macedo <  Cópia de Aula 13 - POCO4A - Exercícios propostos >

Deivi da Silva Galvao<  Cópia de Aula 13 - POCO4A - Exercícios propostos >


João Pedro S. Kawano <  Cópia de Aula 13 - POCO4A - Exercícios propostos >

Marcos Tadao Shoji<  Cópia de Aula 13 - POCO4A - Exercícios propostos >

mabyllly<  Cópia de Aula 13 - POCO4A - Exercícios propostos >

Thiago Cristovão de Souza <  Cópia de Aula 13 - POCO4A - Exercícios propostos >

Vitor Hugo L. A. de Oliveira <  Aula 13 - Exercícios propostos >

Victor Ramos Bernardes  Cópia de Aula 13 - POCO4A - Exercícios propostos

Lucas Viana <  Cópia de Aula 13 - POCO4A - Exercícios propostos >

Felipe Antonio Magro:  Cópia de Aula 13 - POCO4A - Exercícios propostos

Felipe Lorusso e Carlos Eduardo <[Aula 13](#)>

Rodrigo Leandro Benedito:  Cópia de Aula 13 - POCO4A - Exercícios propostos

Exemplo de Classe Interna

```
public class Externa {  
    private int var=111;  
  
    //public class Interna { //Estah disponivel para outras classes  
    //private class Interna { //Classes externas nao tem acesso  
    private static class Interna { //Classe isolada  
        private int var=222;
```

```

        public void imprimir(){
            System.out.println("INTERNA: " + var);
        }
    }

    public void imprimir(){
        System.out.println("EXTERNA");
    }

    public static void main( String [ ] args ){

        Externa externa = new Externa();
        externa.imprimir();
        //Externa.Interna interna = new Externa().new Interna();
        //interna.imprimir();

        Interna interna = new Interna();
        interna.imprimir();
    }
}

```

//Exemplos de sala

```

package javaapplication1;

public class Principal {
    private int var=111;

    public Principal(){
        ClasseInterna ci = new ClasseInterna();
    }
    public void imprimir(){
        System.out.println(var);
    }
    private static class ClasseInterna {
        private int var=222;
        public ClasseInterna(){
            System.out.println(var);
        }
        public void imprimir(){
            System.out.println(var);
        }
    }
}

```

```

public static void main(String[] args) {
    Principal principal =
        new Principal();
    principal.imprimir();

    ClasseInterna p =
        new ClasseInterna();
    p.imprimir();
}
}

-----

public class Principal {

    public abstract class SuperClasse {
        public abstract void imprimir();
    }

    public class SubClasse extends SuperClasse {
        public void imprimir(){
            System.out.println("SUBCLASSE");
        }
    }

    public static void main(String[] args) {
        new Principal();

        Principal.SuperClasse obj = new Principal().new SubClasse();
        Principal.SuperClasse obj2 = new Principal().new SuperClasse(){
            public void imprimir(){ System.out.println(); }
        };
    }
}
//////////
//Classe Interna Anonima

public class Principal {

    public abstract class SuperClasse {
        public abstract void imprimir();
    }
}

```

```

public class SubClasse extends SuperClasse {
    public void imprimir(){
        System.out.println("SUBCLASSE");
    }
}

public static void main(String[] args) {
    //new Principal();
    int var=333;
    //Principal.SuperClasse obj = new Principal().new SubClasse();
    Principal.SuperClasse obj2 = new Principal().new SuperClasse(){
        public void imprimir(){ System.out.println(var); }
    };
    obj2.imprimir();
}
}

```

Exercícios

- 1) Observe o diagrama a seguir:

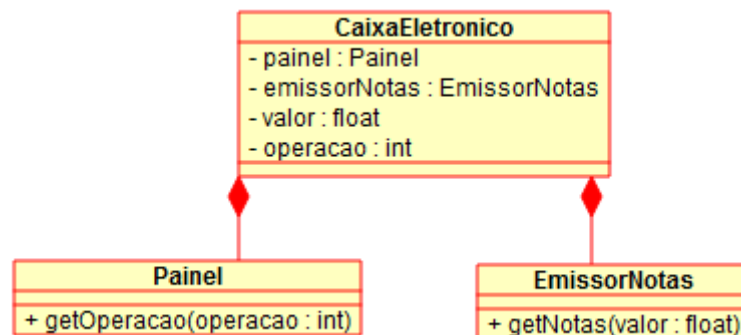


Figura 1 - Diagrama UML de Classes.

- a) Implemente um programa orientado a objetos que utilize classes internas para implementar o diagrama da Figura 1.

```

public class CaixaEletronico {
    private Painei painel;
    private EmissorNotas emissorNotas;
    private float valor;
    private int operacao;
    public class Painei {

        public void getOperacao(int operacao) {

```

```

        //TODO
    }
}
public class EmissorNotas {
    public void getNotas(float valor){
        //TODO
    }
}
}

```

b) Utilize classes internas com o modificador de acesso 'static'.

```

public class CaixaEletronico {
    private Painei painel;
    private EmissorNotas emissorNotas;
    private float valor;
    private int operacao;
    private static class Painei {

        public void getOperacao(int operacao) {
            //TODO
        }
    }
    private static class EmissorNotas {
        public void getNotas(float valor){
            //TODO
        }
    }
}

```

Exemplo de Classe Interna Anônima com Classe Concreta

```
public class Principal {  
  
    public class CaixaEletronico {  
  
        public void imprimir() {  
        }  
        public void imprimir2() {  
        }  
  
    }  
    public class CaixaSubclasse extends CaixaEletronico {  
  
    }  
  
    public static void main(String [ ] args){  
  
        //Classe interna anonima: classe que eh subclasse da classe  
        Principal.CaixaEletronico caixa =  
            new Principal().new CaixaEletronico(){  
                public void imprimir(){ System.out.println("CLASSE ANONIMA"); }  
                public void imprimir2(){ System.out.println("CLASSE ANONIMA"); }  
            };  
        caixa.imprimir();  
        caixa.imprimir2();  
    }  
}
```

Exemplo de Classe Interna Anônima com Classe Abstrata

```
public class Principal {  
  
    public abstract class CaixaEletronico {  
  
        public abstract void imprimir();  
  
        public abstract void imprimir2();  
  
    }  
    public class CaixaSubclasse extends CaixaEletronico {  
  
        public void imprimir(){}  
        public void imprimir2(){}  
  
    }  
  
    public static void main(String [ ] args){  
  
        //Classe interna anonima: classe que eh subclasse da classe  
        Principal.CaixaEletronico caixa =  
            new Principal().new CaixaEletronico(){  
                public void imprimir(){ System.out.println("CLASSE ANONIMA"); }  
                public void imprimir2(){ System.out.println("CLASSE ANONIMA"); }  
            };  
        caixa.imprimir();  
        caixa.imprimir2();  
    }  
}
```

Exemplo de Classe Interna Anônima com Interface

```
public class Principal {  
  
    public interface CaixaEletronico {  
  
        public abstract void imprimir();  
  
        public abstract void imprimir2();  
  
    }  
    public class CaixaSubclasse implements CaixaEletronico {  
  
        public void imprimir(){}  
        public void imprimir2(){}  
  
    }  
  
    public static void main(String [ ] args){  
  
        //Classe interna anonima: classe que eh subclasse da classe  
        Principal.CaixaEletronico caixa =  
            new Principal.CaixaEletronico(){  
                public void imprimir(){ System.out.println("CLASSE ANONIMA"); }  
                public void imprimir2(){ System.out.println("CLASSE ANONIMA"); }  
            };  
        caixa.imprimir();  
        caixa.imprimir2();  
    }  
}
```

Exercícios

- 1) Observe o diagrama a seguir:

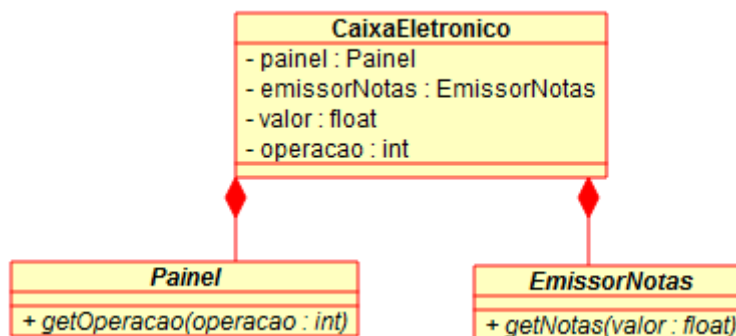


Figura 2 - Diagrama UML de Classes.

- a) No diagrama da Figura 2, as classes Painei e EmissorNotas são classes abstratas com todos os métodos abstratos. Implemente um programa orientado a objetos que utilize classes internas anônimas para implementar o diagrama da Figura 2.

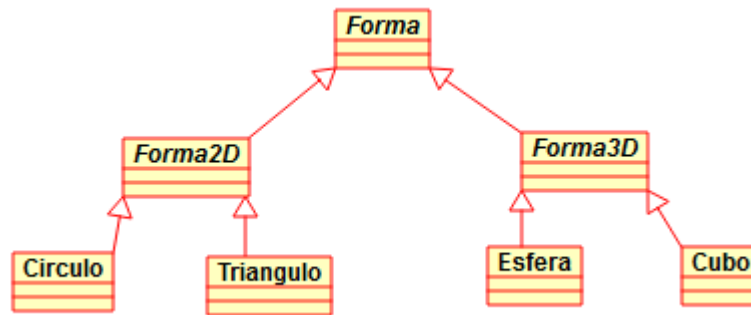
- b) As classes internas anônimas, da mesma forma que as outras classes, invocam o construtor da superclasse. Ilustre um exemplo.
- c) A Classe interna anônima pode acessar os membros da sua Classe de primeiro nível. Ilustre um exemplo.
- d) Modifique o exemplo anterior para que Painei e EmissorNotas sejam interfaces. Implemente um programa orientado a objetos que utilize classes internas anônimas para implementar o diagrama da Figura 2.

2) Faça a implementação Orientada a Objetos do problema anunciado a seguir:

- Crie 3 (três) classes não relacionadas por herança: Construção, Carro e Bicicleta.
- Dê a cada Classe atributos e comportamentos únicos que não estão presentes em outras classes.
- Crie a Interface EmissaoCarbono com um método getEmissaoCarbono.

- d) Cada Classe deve implementar a Interface EmissaoCarbono.
- e) Invoque o método getEmissaoCarbono de cada objeto.

3) Observe a Figura 2 a seguir:



- a) Implemente a hierarquia de Classes mostrada na Figura. Apenas as Classes folha são Classes concretas, as demais são classes abstratas.
- b) A Classe Forma2D deve conter o método getArea.
- c) A Classe Forma3D deve conter os métodos getArea e getVolume.
- d) Crie uma Classe Principal que tenha um vetor de Formas com objetos de cada Classe concreta.
- e) O programa deve imprimir o tipo de cada objeto instanciado.