



**Attribution-NonCommercial-
NoDerivatives 4.0 International
(CC BY-NC-ND 4.0)**



Este trabalho está licenciado com uma Licença [Creative Commons -
Atribuição-NãoComercial-SemDerivações 4.0 Internacional](#).

Programação Orientada a Objetos

2

BACHARELADO EM ENGENHARIA DE COMPUTAÇÃO

PROF. LUCIO AGOSTINHO ROCHA

AULA 13: CLASSE INTERNA E CLASSE ANÔNIMA

1º.SEMESTRE 2023

Classe Interna

Classe Interna

4

- **Classe Interna:**
 - É uma definição de Classe dentro de uma Classe.
 - Propósito:
 - ✦ Agregar várias Classes, sem necessidade de criar novos arquivos.
 - ✦ Proteger a visibilidade de uma Classe de outras Classes.
 - Comporta-se como uma parte da Classe.
 - Possui todos os membros da Classe convencional:
Variáveis de instância e métodos.
 - Classes Internas possuem acesso aos membros privados da Classe Externa.
 - Nota: Classe Interna comum não pode definir membros 'static'.
Para isso, a Classe Interna deve ser 'static'.

Classe Interna

5

- Classe Interna:
 - Deve ser instanciada com uma referência ao objeto da ClasseExterna.

```
public class ClasseExterna { //1)
    private int var;

    public void executar(){
        ClasseInterna classeInterna = new ClasseInterna();
    }
    private class ClasseInterna {
        //TEM acesso aos membros da ClasseExterna
        var = 111;
    }
} //fim classe Externa

...
ClasseExterna classeExterna = new ClasseExterna();
classeExterna.executar();
```

Classe Interna

6

- Classe Interna:
 - Deve ser instanciada com uma referência ao objeto da ClasseExterna.

```
public class ClasseExterna { //2)
    private int var;

    ...
    private class ClasseInterna {
        //TEM acesso aos membros da ClasseExterna
        var = 111;
    }
    ...
    ClasseExterna.ClasseInterna classeInterna =
        new ClasseExterna().new ClasseInterna();
}
```

Classe Interna

7

- Classe Interna 'static':
 - Motivação: Classe Interna com uso exclusivo dentro da Classe.
 - Comporta-se como uma Classe isolada dentro da Classe Externa.
 - É uma Classe interna, mas sem uma referência para a Classe Externa.
 - Não possui acesso imediato aos membros da Classe Externa.
 - Outras Classes não têm acesso à Classe Interna static (deveria ser declarada 'private').

```
public class ClasseExterna{  
    private int var;  
    private static class ClasseInterna {  
        //NÃO TEM acesso aos  
        // membros da ClasseExterna  
        //var = 111;  
    }  
    ...  
    ClasseExterna.ClasseInterna classeInterna =  
        new ClasseExterna().new ClasseInterna();  
}
```

Resumo

8

- **Classe Interna:**
 - Gera um arquivo .class separado
 - Modificadores de acesso permitidos:
 - ✦ public, protected, private, ou acesso de package
 - Classe externa é responsável por criar objetos da Classe Interna
 - Classe interna também pode ser 'static'

Classe Anônima

Classe Anônima

10

- **Classe Anônima:**
 - Uma classe anônima é uma subclasse sem nome de uma superclasse OU
 - É uma classe que implementa uma interface.
 - Em ambos os casos, É uma Classe que não tem um nome.
 - ✦ Não há a palavra reservada 'class', mas não é só isso: a classe é criada, mas não se tem a referência para criar uma instância da classe anônima.
 - Não tem Construtor:
 - ✦ Utiliza o construtor da superclasse
 - Classe interna anônima pode acessar os membros da sua Classe de primeiro nível.

```
public class GUI3 {  
    private int var;  
  
    botao.addActionListener(  
        new ActionListener() { //chamada da  
            //classe interna anônima para  
            //implementar a interface ActionListener  
  
            //Classe interna anonima TEM acesso aos  
            //membros da classe superior de primeiro nivel  
            public void actionPerformed(ActionEvent e) {  
                var=1;  
                System.out.println(var);  
            }  
        });//fim classe interna  
    }  
}//fim classe
```



Revisão

Revisão

13

- Classe Interna
- Classe Anônima

Exercícios

14

<Ver conteúdo na plataforma de ensino>



Referências

15

- Referências bibliográficas da disciplina.