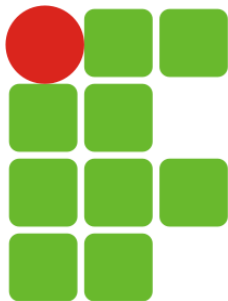


Sobrecarga de métodos e palavras reservadas: final, this e static

Tulio A.R.

IFSC
Instituto Federal de Santa Catarina (IFSC)
IFSC (UFSC)
`tulio.alberton@ifsc.edu.br`

Outubro, 2014





Sumário

Sobrecarga de métodos

Palavra reservada final

Palavra reservada static

Membros de classe estáticos: `static`

Membros de classe estáticos: `static`

Membros de classe estáticos: `static`

Palavra Reservada `this`

Resumo

1 Sobrecarga de métodos

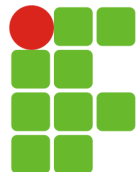
2 Palavra reservada final

3 Palavra reservada static

- Membros de classe estáticos: `static`
- Membros de classe estáticos: `static`
- Membros de classe estáticos: `static`

4 Palavra Reservada `this`

5 Resumo





Sobrecarga de métodos

Palavra reservada final

Palavra reservada static

Membros de classe estáticos: `static`

Membros de classe estáticos: `static`

Membros de classe estáticos: `static`

Palavra Reservada `this`

Resumo

■ Métodos sobrecarregados devem possuir assinaturas diferentes

- Neste caso, a assinatura de método é representada pelo número de parâmetros e pelo tipo dos parâmetros



Sobrecarga de métodos

Palavra reservada
final

Palavra reservada
static

Membros de classe
estáticos: `static`

Membros de classe
estáticos: `static`

Membros de classe
estáticos: `static`

Palavra
Reservada `this`

Resumo

Exemplo de sobrecarga de métodos

```
0 public class Data{
1     private int dia, mes, ano;
2
3     public void alterarData(int d){
4         this.dia = d;
5     }
6     public void alterarData(int d, int m){
7         this.dia = d; this.mes = m;
8     }
9     public void alterarData(int d, int m, int a){
10        this.dia = d; this.mes = m; this.ano = a;
11    }
12 }
```



Sobrecarga de métodos

Palavra reservada final

Palavra reservada static

Membros de classe estáticos: `static`

Membros de classe estáticos: `static`

Membros de classe estáticos: `static`

Palavra Reservada `this`

Resumo

Exemplo de como acessar os métodos

```
0 //Instanciando e chamando métodos sobrecarregados:
1 Data d = new Data();
2 d.alterarData(31);
3 d.alterarData(31,12);
4 d.alterarData(31,12,1969);
```



Sobrecarga de métodos

Palavra reservada final

Palavra reservada static

Membros de classe estáticos: `static`

Membros de classe estáticos: `static`

Membros de classe estáticos: `static`

Palavra Reservada `this`

Resumo

Constantes com o modificador `final`

- O modificador `final` pode ser usado em atributos ou métodos de uma classe, bem como em variáveis locais.
 - Uma vez que atribuiu valores para variáveis ou atributos, estes não poderão ser alterados
 - Por convenção, constantes deverão ser escritas em letras maiúsculas
 - Métodos não poderão ser sobrescritos (conceito de herança)

Membros de classe
estáticos: s t a t i cMembros de classe
estáticos: s t a t i cMembros de classe
estáticos: s t a t i c

Constantes com o modificador final, exemplo de uso

```
4 public class Celular{
5
6     private final int FREQUENCIA = 1800;
7     private final int SERIAL;
8
9     public Celular(int s){
10         this.SERIAL = s;
11     }
12
13     public final void iniciarChamada(){
14         // ....
15     }
16 }
```



Sobrecarga de métodos

Palavra reservada final

Palavra reservada static

Membros de classe estáticos: static

Membros de classe estáticos: static

Membros de classe estáticos: static

Palavra Reservada this

Resumo

Atributos não estáticos – cada instância da classe terá uma cópia distinta deste atributo.

```
16 public class Celular{
17     private int total;
18
19     public Celular(){
20         this.total = this.total + 1;
21     }
22     public void incrementar(){
23         this.total = this.total + 1;
24     }
25     public int getTotal(){
26         return this.total;
27     }
28 }
```




Sobrecarga de métodos

Palavra reservada final

Palavra reservada static

Membros de classe estáticos: `static`

Membros de classe estáticos: `static`

Membros de classe estáticos: `static`

Palavra Reservada this

Resumo

Verifique você mesmo

```
28     Celular a = new Celular();
29     Celular b = new Celular();
30     a.incrementar(); b.incrementar();
31
32     System.out.println(a.getTotal());
33     // o que será impresso?
34     System.out.println(b.getTotal());
35     // o que será impresso?
```



Sobrecarga de métodos

Palavra reservada final

Palavra reservada static

Membros de classe estáticos: s t a t i c

Membros de classe estáticos: s t a t i c

Membros de classe estáticos: s t a t i c

Palavra Reservada this

Resumo

- Atributos **estáticos** ficam **comuns para todos** os objetos que foram instanciados para esta classe, sendo assim chamados de “atributos da classe”
 - Não se pode usar o **this** para acessar um membro estático. Deve-se usar o nome da Classe
- Classes podem possuir métodos estáticos e estes podem ser invocados sem que necessite criar uma instância da classe
 - Métodos estáticos geralmente são usados para acessar atributos estáticos

```
35 public class Celular{
36     private static int total = 0;
37     public static int getTotal(){
38         return Celular.total;
39     }
40 }
```



Sobrecarga de métodos

Palavra reservada final

Palavra reservada static

Membros de classe estáticos: `static`

Membros de classe estáticos: `static`

Membros de classe estáticos: `static`

Palavra Reservada `this`

Resumo

```
40 public class Celular{
41     private static int total = 0;
42     private int serial;
43
44     public Celular(int s){
45         this.serial = s;
46         Celular.total = Celular.total + 1;
47     }
48     public static int getTotal(){
49         return Celular.total;
50     }
51     public int getSerial(){
52         return this.serial;
53     }
54 }
```



Sobrecarga de métodos

Palavra reservada final

Palavra reservada static

Membros de classe estáticos: `static`

Membros de classe estáticos: `static`

Membros de classe estáticos: `static`

Palavra Reservada `this`

Resumo

```
54 System.out.println(Celular.getTotal());
55 // o que será impresso?
56 Celular c = new Celular(123);
57 Celular d = new Celular(456);
58 System.out.println(Celular.getTotal());
59 // o que será impresso?
60 System.out.println(d.getSerial());
61 // o que será impresso?
```



Sobrecarga de métodos

Palavra reservada
final

Palavra reservada
static

Membros de classe
estáticos: `static`

Membros de classe
estáticos: `static`

Membros de classe
estáticos: `static`

Palavra
Reservada `this`

Resumo

```
62 public class Complexo{
63     private int real;
64     private int imaginario;
65
66     public Complexo(int a, int b){
67         this.real = a; this.imaginario = b;
68     }
69     public void soma(Complexo c){
70         this.real = this.real + c.real;
71         this.imaginario = c.imaginario;
72     }
73     public void imprimir(){
74         System.out.println(this.real + "," + this.
75             imaginario);
76     }
77 }
```



Sobrecarga de métodos

Palavra reservada final

Palavra reservada static

Membros de classe estáticos: `static`

Membros de classe estáticos: `static`

Membros de classe estáticos: `static`

Palavra Reservada this

Resumo

```
76 Complexo a = new Complexo(1,2);
77 Complexo b = new Complexo(3,4);
78 a.soma(b);
79 a.imprimir();
80 b.imprimir();
```

Classe Bicicleta

```
0     public class Bicicleta{
1         private double valor;
2
3         public void imprimirValor(){
4             System.out.println(this.valor);
5         }
6     }
```



Sobrecarga de métodos

Palavra reservada final

Palavra reservada static

Membros de classe estáticos: `static`

Membros de classe estáticos: `static`

Membros de classe estáticos: `static`

Palavra Reservada `this`

Resumo

```
45 public class Principal {
46     private double valor;
47     public void imprimirVelocidade() {
48         System.out.println(this.valor);
49     }
50     public void teste() {
51         Bicicleta b = new Bicicleta();
52         b.acelerar(10);
53         b.imprimirVelocidade();
54     }
55     public static void main(String[] args) {
56         Principal p = new Principal();
57         p.imprimirVelocidade();
58         p.teste();
59     }
60 }
61 //Na linha 48 o this é referência para o objeto de
    qual classe? Principal ou Bicicleta?
```



Sobrecarga de métodos

Palavra reservada final

Palavra reservada static

Membros de classe estáticos: `st at i c`

Membros de classe estáticos: `st at i c`

Membros de classe estáticos: `st at i c`

Palavra Reservada `this`

Resumo

■ Sobrecarga de métodos

- Uma classe pode ter mais de um método com o mesmo nome, porém com assinaturas diferentes

■ A palavra `this` é uma referência para o objeto atual

- Apesar de não obrigatório na maioria dos casos, seu uso é desejado para facilitar a leitura do código

■ Atributos estáticos ficam comum para todos objetos instanciados da classe

- Imagine que é uma variável compartilhada entre todos os objetos da classe

■ Modificador final é usado para definir constantes

- Atributo final não poderá ter seu valor alterado