



101

Sobrecarga de Métodos

- Métodos com nomes iguais, mas assinaturas diferentes
- Métodos com o mesmo nome podem ser declarados dentro de uma mesma classe
 - Desde que possuam um conjunto diferente de parâmetros;
 - Sobrecarga de métodos.

Professor Ariel da Silva Dias - www.arieldias.com - Obra Gratuita, proibida reprodução e venda

102

Sobrecarga de Métodos

- Quando um método sobrecarregado é invocado, o compilador seleciona o método apropriado
 - De acordo com o número, tipo e ordem dos argumentos passados para o método.
- Desta forma, podemos ter um conjunto de métodos com o mesmo nome que realizam o mesmo tipo de operação sobre argumentos diferentes.

Professor Ariel da Silva Dias - www.arieldias.com - Obra Gratuita, proibida reprodução e venda

103

Sobrecarga de Métodos

```
1 public class Calcular
2 {
3     int quadrado(int num)
4     {
5         return num*num;
6     }
7
8     double quadrado(double num)
9     {
10        return num*num;
11    }
12
13    public void print()
14    {
15        System.out.println("Quadrado de 7.5 é: " + quadrado(7.5));
16        System.out.println("Quadrado de 7 é: " + quadrado(7));
17    }
18 }
```

Professor Ariel da Silva Dias - www.arieldias.com - Obra Gratuita, proibida reprodução e venda

104

Sobrecarga de Métodos

- Por exemplo, o método *quadrado()* da classe *Calcular* são sobrecarregados, cada um com duas versões:
 - Uma que calcula o quadrado de um número int;
 - Uma que calcula o quadrado de um número double.

Professor Ariel da Silva Dias - www.arieldias.com - Obra Gratuita, proibida reprodução e venda

105

Sobrecarga de Construtores

- Em POO é possível que objetos de uma mesma classe sejam inicializados de formas diferentes
 - Através da sobrecarga de construtores;
 - Basta definir múltiplos construtores com assinaturas diferentes
 - Número e tipo de argumentos.

```

1 public class Tempo
2 {
3     private int h, m, s;
4
5     public Tempo()
6     {
7         h = m = s = 0;
8     }
9
10    public Tempo(int hora)
11    {
12        h = hora;
13        m = s = 0;
14    }
15
16    public Tempo(int hora, int minuto)
17    {
18        h = hora;
19        m = minuto;
20        s = 0;
21    }
22 }

```

Professor Ariel da Silva Dias - www.arieldias.com - Obra Gratuita, proibida reprodução e venda

106

Sobrecarga de Métodos

- Outros métodos podem possuir o mesmo nome que a classe
 - Embora não se tratem de construtores;
 - Não são chamados quando um objeto da classe é criado;
 - Possuem tipo de retorno.
- Um erro comum é colocar um tipo de retorno em um método com o mesmo nome da classe e confundir-lo com um construtor.

Professor Ariel da Silva Dias - www.arieldias.com - Obra Gratuita, proibida reprodução e venda

107

Sobrecarga de Métodos

```
1 public class ConstrutorFalso
2 {
3     public int ConstrutorFalso()
4     {
5         System.out.println("Um objeto foi criado?");
6         return 1;
7     }
8
9     public ConstrutorFalso()
10    {
11        System.out.println("Um objeto foi criado!");
12    }
13
14    public static void main(String args[])
15    {
16        ConstrutorFalso obj = new ConstrutorFalso();
17    }
18 }
```

Professor Ariel da Silva Dias - www.arieldias.com - Obra Gratuita, proibida reprodução e venda

108

Composição

- Uma classe pode ter referências a objetos de outras classes como membros
 - Composição, ou relacionamento *tem-um*.
- Por exemplo, um despertador precisa saber o horário atual
 - É razoável embutir duas referências a objetos de uma classe *Hora* como membros da classe *Despertador*.

Professor Ariel da Silva Dias - www.arieldias.com - Obra Gratuita, proibida reprodução e venda

109

Composição

```

1 public class Hora
2 {
3     private int h, m;
4
5     public int getH()
6     {
7         return h;
8     }
9
10    public int getM()
11    {
12        return m;
13    }
14
15    public void setH(int valor)
16    {
17        h = valor;
18    }
19
20    public void setM(int valor)
21    {
22        m = valor;
23    }
24 }

```

```

1 public class Despertador
2 {
3     boolean ligado;
4     Hora despertador, horarioAtual;
5
6     public void setDespertador (int h, int m, int s, boolean valor)
7     {
8         despertador.setH(h);
9         despertador.setM(m);
10        despertador.setS(s);
11        ligado = valor;
12    }
13
14    public void setHorarioAtual (int h, int m, int s)
15    {
16        horarioAtual.setH(h);
17        horarioAtual.setM(m);
18        horarioAtual.setS(s);
19    }
20 }

```

Professor Ariel da Silva Dias - www.arieldias.com - Obra Gratuita, proibida reprodução e venda

110

Referência

- Notas de aula Professor José Romildo Malaquias - UFOP

Professor Ariel da Silva Dias - www.arieldias.com - Obra Gratuita, proibida reprodução e venda

111

Por hoje é só!

Orientações

- Pratique os exercícios desta aula
- Somente com a prática constante você terá fluência no assunto!

Qualquer dúvida:

- www.arieldias.com
- contato@arieldias.com

Bons estudos! Até a próxima!

Professor Ariel Dias - www.arieldias.com - Obra licenciada, proibida reprodução e uso sem autorização

112