

Construtores

O (pseudo-)método construtor determina que ações devem ser executadas quando da criação de um objeto. Em Java, o construtor é definido como um método cujo nome deve ser o mesmo nome da classe e sem indicação do tipo de retorno -- nem mesmo void. O construtor é unicamente invocado no momento da criação do objeto através do operador new.

O retorno do operador new é uma referência para o objeto recém-criado. O construtor pode receber argumentos, como qualquer método. Usando o mecanismo de sobrecarga, mais de um construtor pode ser definido para uma classe.

Toda classe tem pelo menos um construtor sempre definido. Se nenhum construtor for explicitamente definido pelo programador da classe, um construtor padrão, que não recebe argumentos, é incluído para a classe pelo compilador Java.

No momento em que um construtor é invocado, a seguinte seqüência de ações é executada para a criação de um objeto:

1. O espaço para o objeto é alocado e seu conteúdo é inicializado (bitwise) com zeros.
2. O construtor da classe base é invocado. Se a classe não tem uma superclasse definida explicitamente, a classe Object é a classe base.
3. Os membros da classe são inicializados para o objeto, seguindo a ordem em que foram declarados na classe.
4. O restante do corpo do construtor é executado.

```
public class DataFabricacao {  
  
    private int dia;  
    private int mes;  
    private int ano;  
  
    public int getDia() {  
        return dia;  
    }  
    public void setDia(int dia) {  
        this.dia = dia;  
    }  
    public int getMes() {  
        return mes;  
    }  
    public void setMes(int mes) {  
        this.mes = mes;  
    }  
    public int getAno() {
```

```

        return ano;
    }
    public void setAno(int ano) {
        this.ano = ano;
    }

    public String dataCompleta() {
        return this.dia + "/" + this.mes + "/" + this.ano;
    }

    public DataFabricacao() {

    }

    public DataFabricacao(int dia, int mes, int ano) {
        this.dia = dia;
        this.mes = mes;
        this.ano = ano;
    }
}

```

1. Crie a classe **ExemploData** com os atributos privados e inteiros **dia**, **mes** e **ano**. Então faça o construtor padrão e um construtor sobrecarregado que inicializa apenas **mes** e **ano**.

```

public class ExemploData {
    private int dia;
    private int mes;
    private int ano;

    public int getDia() {
        return dia;
    }
    public void setDia(int dia) {
        this.dia = dia;
    }
    public int getMes() {
        return mes;
    }
    public void setMes(int mes) {
        this.mes = mes;
    }
    public int getAno() {
        return ano;
    }
}

```

```

    public void setAno(int ano) {
        this.ano = ano;
    }

    //Construtores
    public ExemploData() {

    }

    public ExemploData(int mes, int ano) {
        this.mes = mes;
        this.ano = ano;
    }
}

```

2. Faça uma classe **TesteExemploData.java** que peça para o usuário o digitar o mês e o ano do seu nascimento. Depois criar um objeto do tipo **ExemploData** e ao mesmo tempo já o inicialize com as variáveis que contêm mês e ano. Ao final, exiba a data em um **JOptionPane.showMessageDialog()**

```

import javax.swing.JOptionPane;

public class TesteExemploData {

    public static void main(String[] args) {

        int mes = Integer.parseInt(JOptionPane.showInputDialog("Digite o
mês: "));
        int ano = Integer.parseInt(JOptionPane.showInputDialog("Digite o
ano: "));

        ExemploData data = new ExemploData(mes, ano);

        String mensagem = "Mês e ano: " + data.getMes() + "/" +
data.getAno();

        JOptionPane.showMessageDialog(null, mensagem);

    }
}

```