

# **Tópicos Especiais em Sistemas para Internet I**

**Prof. Orlando Saraiva Júnior**  
**[orlando.nascimento@fatec.sp.gov.br](mailto:orlando.nascimento@fatec.sp.gov.br)**

"A Fallacy of Software: If it works, and we don't change anything, it will keep working."

Jessica Kerr

# **Tópicos Especiais em Sistemas para Internet I**

# Objetivo da aula

---

Novos conceitos:

Estrutura de Dados

Composição / Associação

# **Programa BancoFATEC v.3**

# O que vamos fazer ?

---

Criar uma classe `CarteiraClientes`, responsável em administrar diversas contas ( Poupança e Conta Corrente) de diversos clientes.

Uma classe `ContaCorrente`, responsável por uma conta corrente.

Uma classe `ContaPoupanca`, responsável por uma conta poupança.

# Conceitos

**Classe:** ~~É como a uma estrutura do C, mas mais completa.~~ Uma classe é uma estrutura que abstrai um conjunto de objetos com características similares.

**Objeto:** Uma instância da classe.

**Métodos:** ~~São funções da classe.~~ Os métodos determinam o comportamento dos objetos.

**Atributos:** ~~São as variáveis da classe.~~ Os atributos determinam as características dos objetos.



# Tipos Primitivos vs tipos por referência

---

Os tipos do Java são divididos em tipos primitivos, e tipo por referência. Os tipos primitivos são:

boolean, byte, char, short, int, long, float, double.  
Todos os tipos não primitivos são tipos por referência, portanto, as classes, que especificam os tipos de objeto, são tipos por referência.

# Tipos Primitivos vs tipos por referência

---

Ao utilizar um objeto de outra classe, uma referência ao objeto deve **invocar** (isto é, chamar) seus métodos.

Observe que as variáveis de um tipo primitivo não referenciam objetos, então essas variáveis não podem ser utilizadas para invocar métodos.

Cada classe que você declara pode fornecer um método especial chamado **construtor** que pode ser utilizado para inicializar um objeto de uma classe.

Um construtor deve ter o mesmo nome da classe.

Por padrão, o compilador fornece um **construtor padrão** sem parâmetros em qualquer classe que não inclui explicitamente um construtor.



Uma coleção é uma estrutura de dados (na realidade um objeto), que pode armazenar ou agrupar referências a outros objetos (um contêiner).

As classes e interfaces da estrutura de coleções são membros do pacote `java.util`

Quando uma instância da classe existente é usada como componente da outra classe;

Estamos lidando com um relacionamento tem-um.

No nosso exemplo, CarteiraClientes tem:

- *N* ContaPoupanca
- *N* ContaCorrente

# Dúvidas

**Prof. Orlando Saraiva Júnior**  
**[orlando.nascimento@fatec.sp.gov.br](mailto:orlando.nascimento@fatec.sp.gov.br)**

A classe `CarteiraCliente` não consegue gerenciar contas poupança.

A classe `ContaPoupanca` permite retiradas maiores que o saldo. Limite para saque deve ser o saldo.

A classe `ContaCorrente` permite retiradas maiores que o saldo. Limite para saque deve ser de até R\$ 500,00 negativo.