

The logo for Oracle Academy is centered on a light gray background. It features the word "ORACLE" in a bold, orange, sans-serif font. Below it, the word "Academy" is written in a smaller, dark gray, sans-serif font. The entire logo is framed by two horizontal dark gray bars, one at the top and one at the bottom.

# ORACLE

## Academy

# Java Foundations

2-3

## Introdução aos Conceitos da Programação Orientada a Objetos

**ORACLE**  
Academy



Copyright © 2022, Oracle e/ou suas empresas afiliadas. Oracle, Java e MySQL são marcas comerciais registradas da Oracle Corporation e/ou de suas empresas afiliadas. Outros nomes podem ser marcas comerciais de seus respectivos proprietários.

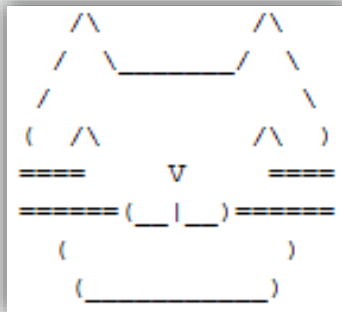
# Objetivos

- Esta lição abrange os seguintes objetivos:
  - Fazer a distinção entre programação procedural e programação orientada a objetos
  - Entender uma classe como um plano gráfico de um objeto
  - Entender como uma classe é usada para criar instâncias de objeto
  - Modelar objetos como uma combinação de...
    - Propriedades (campos de dados)
    - Comportamentos (métodos)



# Revisão

- Até aqui, abordamos...
  - Décadas de inovação da ciência da computação
  - Gigabytes de potência da computação moderna
- E muitos aspectos como a Internet...
  - Criamos um gato!

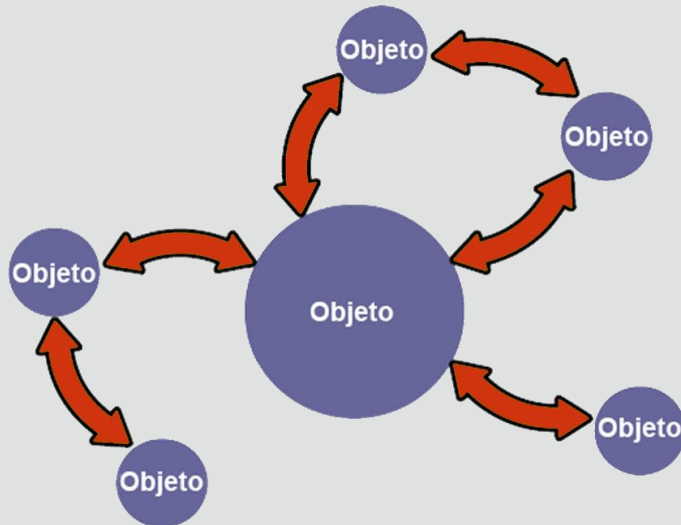


# O Java Pode Fazer Mais!

- Linguagens Procedurais...
  - Leem uma linha por vez
  - A linguagem C é procedural
- Linguagens orientadas a objetos...
  - Leem uma linha por vez
  - Modelam objetos por meio do código
  - Enfatizam a interação do objeto
  - Permitem uma interação sem uma ordem prescrita
  - Java e C++ são linguagens orientadas a objetos

# Programação Orientada a Objetos

- Interação de objetos
- Nenhuma sequência prescrita



O diagrama ilustra como a programação orientada a objetos foca na interação de objetos.

# Exercício 1



- Vá para <https://objectstorage.uk-london-1.oraclecloud.com/n/lrvrlgaqj8dd/b/Games/o/JavaPuzzleBall/index.html>
- Execute Basic Puzzles de 1 a 5
  - Sua Meta: projetar uma solução que desvie a bola para o Duke
- Considere o seguinte:
  - Quais objetos você encontra no campo de jogo?
  - O que acontece quando você insere um ícone de parede de triângulo ou parede simples no círculo azul?



**ORACLE**  
Academy

JFo 2-3  
Introdução aos Conceitos da Programação  
Orientada a Objetos

Copyright © 2022, Oracle e/ou suas empresas afiliadas. Oracle, Java e MySQL são marcas comerciais registradas da Oracle Corporation e/ou de suas empresas afiliadas. Outros nomes podem ser marcas comerciais de seus respectivos proprietários.

7

Considere jogar além do Basic Puzzle 5. Os quebra-cabeças a partir do Basic.05 estão associados a lições posteriores.



## Sobre o Java Puzzle Ball

- Execute um conjunto de quebra-cabeças
- Familiarize-se com os mecanismos do jogo
- Considere as perguntas à medida que você joga
- Ouça a explicação das lições sobre o que você observou
- Aplique suas observações para compreender os conceitos de Java



**ORACLE**  
Academy

JFo 2-3  
Introdução aos Conceitos da Programação  
Orientada a Objetos

Copyright © 2022, Oracle e/ou suas empresas afiliadas. Oracle, Java e MySQL são marcas comerciais registradas da Oracle Corporation e/ou de suas empresas afiliadas. Outros nomes podem ser marcas comerciais de seus respectivos proprietários.

8

Este jogo reflete os conceitos Java por meio de mecanismos de jogo. É mais importante familiarizar-se com esses mecanismos do que solucionar cada quebra-cabeças. Não se preocupe se a relação entre os mecanismos do jogo e os conceitos Java não ficar clara de imediato. Explicaremos essa relação detalhadamente a você mais tarde. Durante essa explicação, as pessoas tendem a entender essa relação. À medida que desenvolver uma compreensão de como o jogo funciona, você conseguirá aplicar o que aprendeu como uma base para entender os conceitos difíceis de Java.





## Tipos de Objeto

- Quais objetos você encontra no campo de jogo?

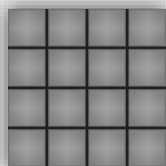
- Ball



- Duke



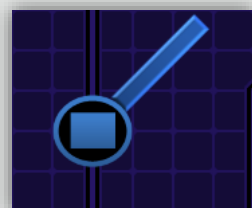
- LevelGeometry



- RedBumper



- BlueBumper



**ORACLE**  
Academy

JFo 2-3  
Introdução aos Conceitos da Programação  
Orientada a Objetos

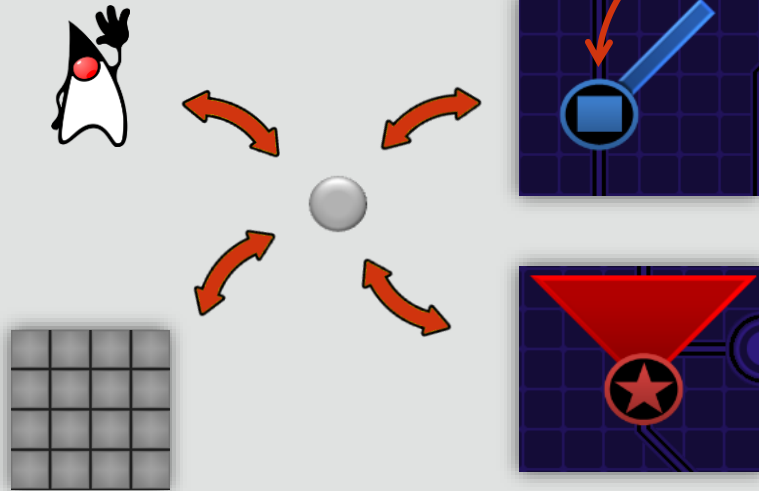
Copyright © 2022, Oracle e/ou suas empresas afiliadas. Oracle, Java e MySQL são marcas comerciais registradas da Oracle Corporation e/ou de suas empresas afiliadas. Outros nomes podem ser marcas comerciais de seus respectivos proprietários.

9



# Interação de Objetos

- Interação de objetos
- Nenhuma sequência prescrita



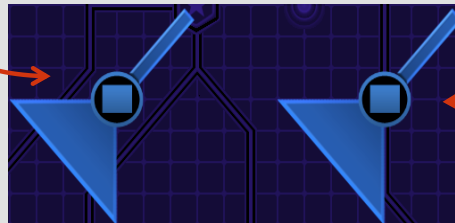
Vamos olhar  
este objeto de  
perto



# Objetos BlueBumper

- O que acontece quando você insere um ícone de parede de triângulo ou parede simples em um círculo azul?
- Uma parede aparece em cada instância de um objeto blue bumper (para-choque azul)
- As paredes dão aos para-choques comportamentos que desviam e interagem com a bola
- Todas as instâncias blue bumper compartilham esses mesmos comportamentos

Instância



Instância



ORACLE  
Academy

JFo 2-3  
Introdução aos Conceitos da Programação  
Orientada a Objetos

Copyright © 2022, Oracle e/ou suas empresas afiliadas. Oracle, Java e MySQL são marcas comerciais registradas da Oracle Corporation e/ou de suas empresas afiliadas. Outros nomes podem ser marcas comerciais de seus respectivos proprietários.

11

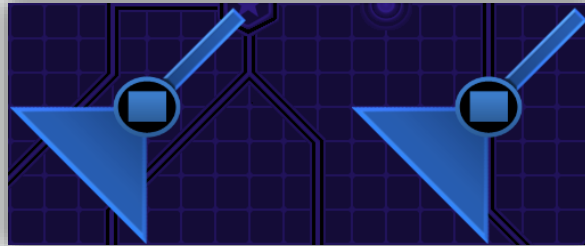
Um BlueBumper é um objeto, e toda instância desses objetos compartilharão o mesmo comportamento ao interagir com a Bola. Esses comportamentos podem incluir um desvio através do triângulo ou da parede simples.



# Descrevendo um BlueBumper

- Propriedades:

- Cor
- Forma
- Posição de x
- Posição de y



- Comportamentos:

- Fazer um som de ping
- Piscar
- Desviar a bola (através da Parede Simples)
- Desviar a bola (através da Parede de Triângulo)



# Descrevendo uma Bola

- Propriedades:

- Direção
- Posição de x
- Posição de y



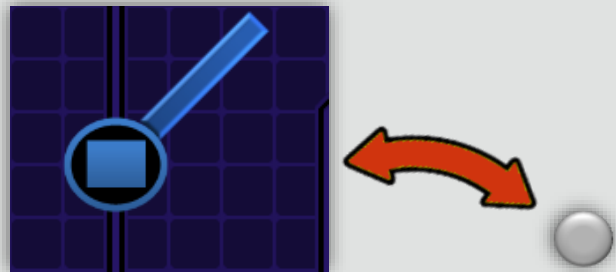
- Comportamentos:

- Fazer um som de ping
- Mudar a direção
- Mudar a posição de x
- Mudar a posição de y

## Interação entre o BlueBumper e a Bola



- A interação ocorre quando BlueBumper desvia a Bola Quando isso acontece...
- As propriedades da Bola mudam:
  - A Bola percorre outra direção
  - As posições x e y futuras da Bola mudam
- O BlueBumper executa comportamentos:
  - Faz um som de ping
  - Pisca



Todos os BlueBumpers compartilham a capacidade de fazer ping, piscar e interagir com a bola.

## Por que Isso Importa?



- Observamos aspectos importantes da programação orientada a objetos
- Lembre-se dessas observações à medida que as lições e os exercícios tornarem-se cada vez mais técnicos
  - Os objetos podem ser descritos como uma combinação de propriedades e comportamentos
  - Pode haver muitas instâncias do mesmo tipo de objeto
  - Todas as instâncias de um objeto compartilham os mesmos comportamentos
  - Os objetos podem interagir entre si, possivelmente afetando as propriedades de cada um e acionando outros comportamentos

# Um Exemplo Diferente

- Propriedades:

- Nome
- Idade
- Raça
- Comida Favorita



- Comportamentos:

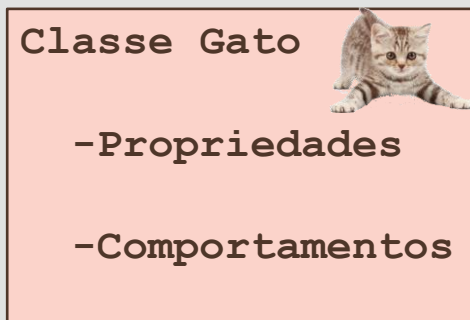
- Miar
- Brincar
- Lavar
- Comer
- Caçar

Note a diferença sutil entre Comida Favorita (que é uma descrição) e Comer (que é um verbo).

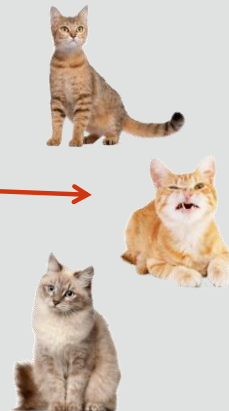


# Classes e Instâncias

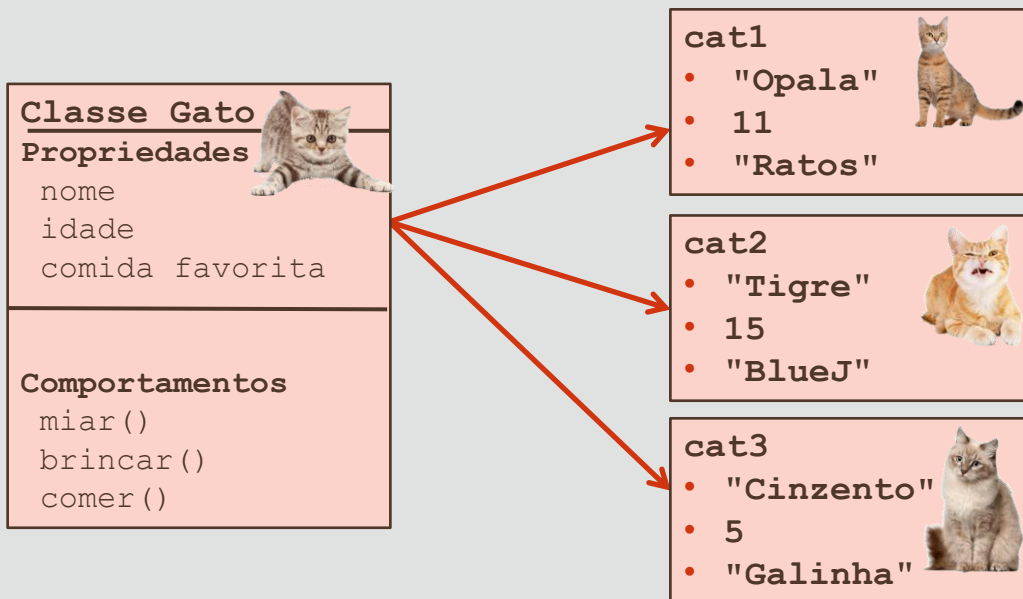
- A combinação de propriedades e comportamentos é...
  - Denominada classe
  - Um plano gráfico ou uma receita para um objeto
  - Usada para criar instâncias do objeto



**Instâncias do objeto**



# Criando Novas Instâncias de um Plano Gráfico



**Todas as instâncias do gato compartilham a capacidade de miar, brincar e comer**

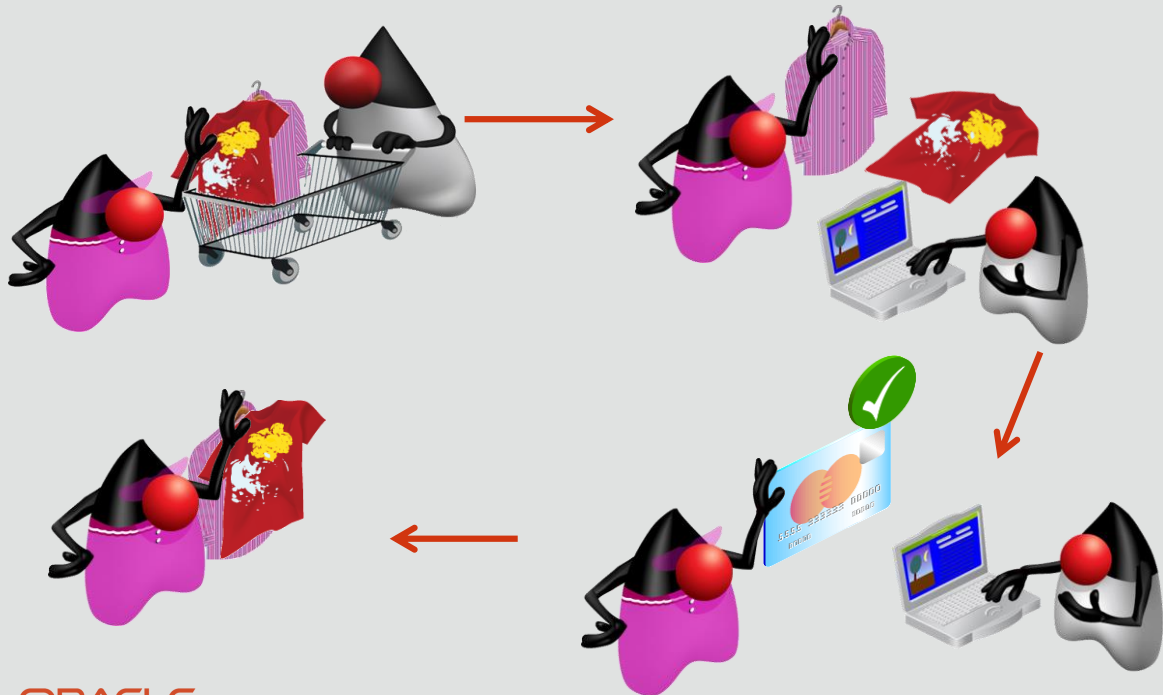
# Estratégia Orientada a Objetos

- Como você escreve programas para obter esse nível de flexibilidade?
- Quando você tem uma ideia ou um requisito para um programa...
  - Considere que tipo de objetos pode existir nesse programa
  - Considere as propriedades e os comportamentos desses tipos de objetos
  - Considere o modo como os objetos interagem

Ao projetar um programa Java, primeiro identifique os objetos. Em seguida, determine as características ou as propriedades dos objetos, bem como os comportamentos ou as operações deles. Também é útil considerar como os objetos interagirão ou afetarão as propriedades uns dos outros.

A última etapa é traduzir essa análise em um código Java para criar seu aplicativo. Mostraremos essa última etapa no fim desta lição para mostrar como isso é possível. Você não terá de criar seu próprio código de uma classe Java até seções mais adiante.

# Loja de Compras On-line Duke's Choice



**ORACLE**  
Academy

JFo 2-3  
Introdução aos Conceitos da Programação  
Orientada a Objetos

Copyright © 2022, Oracle e/ou suas empresas afiliadas. Oracle, Java e MySQL são marcas comerciais registradas da Oracle Corporation e/ou de suas empresas afiliadas. Outros nomes podem ser marcas comerciais de seus respectivos proprietários.

20

Vamos dar uma olhada no cenário de um carrinho de compras on-line. Imagine uma loja de compras on-line denominada Duke's Choice. A principal compradora é a mãe dele, Sra. Duke. Quando faz compras, ela coloca os itens em um carrinho de compras. Como gosta de camisas, ela coloca as camisas no carrinho. Depois que enche o carrinho, ela encaminha-se para fechar a conta. O processo de fechamento da conta submete a compra a um cartão de crédito, que é verificado. A Sra. Duke recebe um número de pedido para que possa rastrear o pedido ou devolvê-lo

Como desenvolvedor de software, quando for apresentado a um cenário como a Duke's Choice de um aplicativo que precisa ser desenvolvido, você poderá analisar o cenário dividindo-o em etapas e definindo os objetos do cenário.

# Características dos Objetos

- Os objetos são físicos ou conceituais
- Os objetos têm propriedades:
  - Tamanho
  - Preço
  - Cor
- Os objetos têm comportamentos:
  - Comprar
  - Colocar o item no carrinho
  - Pagar



Físico:  
camisa



Conceitual:  
conta on-line



O valor da propriedade  
Cor é vermelho



Sra. Duke

Para validar objetos em um domínio de problema, como o processo do pedido da loja Duke's Choice, você identifica as propriedades de todos os objetos.

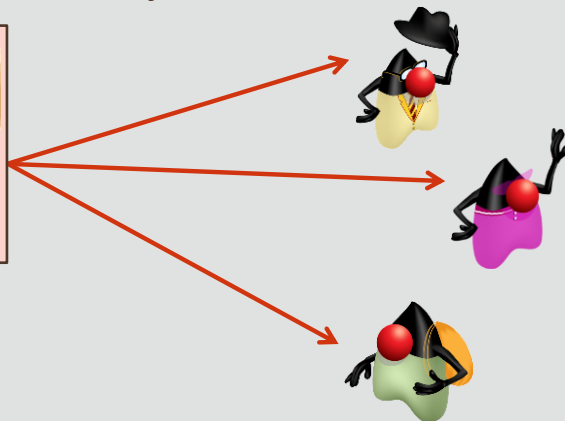
- Os objetos são físicos ou conceituais. Uma camisa é um exemplo de um objeto físico. A conta do cartão de crédito de um cliente é um exemplo de um objeto conceitual porque não é algo que você possa tocar fisicamente.
- Os objetos têm propriedades (atributos), como tamanho, nome e forma, que representam o estado do objeto. Por exemplo, uma pessoa tem um nome (Sra. Duke), e um objeto poderia ter uma propriedade cor. O valor de todas as propriedades de um objeto geralmente é denominado *estado atual* do objeto. Um objeto poderia ter uma propriedade color com um valor vermelho e uma propriedade size com um valor grande.
- Os objetos também têm comportamentos (ações que podem realizar) como, no nosso exemplo, comprar, colocar um item no carrinho e pagar.

# Classes e Instâncias

- Lembre-se de que uma classe...
  - É um plano gráfico ou uma receita para um objeto
  - Descreve as propriedades e os comportamentos de um objeto
  - É usada para criar instâncias do Objeto



Instâncias do objeto



Acabamos de abordar algumas das características e dos comportamentos dos objetos no cenário Duke's Choice. Veja a seguir um exemplo de um dos objetos da Duke's Choice, Customer (Cliente), e a função dele na loja Customer é a classe, e uma classe é um plano gráfico ou uma receita de um objeto. A classe descreve as propriedades e os comportamentos de um objeto.

Você usa as classes para criar instâncias do objeto, como as três instâncias do objeto Customer, conforme ilustrado pelas três imagens.

## Exercício 2, Parte 1

- Dado o cenário a seguir, quais objetos você poderia modelar para concluir seu programa?
  - Crie um programa para uma máquina de contagem de moedas. Essa máquina deve medir, contar e classificar moedas com base no respectivo tamanho ou valor.
  - Ela também deve imprimir um recibo.
- Liste pelo menos 3 objetos que você precisaria modelar para este exemplo:

- 
- 
- 



## Exercício 2, Parte 2

- |   |  |  |
|---|--|--|
| <ul style="list-style-type: none"><li>• Escolha um objeto da Parte 1</li><li>• Quais propriedades e comportamentos desse objeto você poderia incluir no seu programa?</li></ul> | <ul style="list-style-type: none"><li>• Propriedades:<ul style="list-style-type: none"><li>—</li><li>—</li><li>—</li></ul></li></ul> | <ul style="list-style-type: none"><li>• Comportamentos:<ul style="list-style-type: none"><li>—</li><li>—</li><li>—</li></ul></li></ul> |
|---|--|--|



# Propriedades e Comportamentos da Classe Customer

- Propriedades:

- Nome
- Endereço
- Idade
- Número do pedido
- Número do cliente



- Comportamentos:

- Comprar
- Definir o endereço
- Adicionar um item ao carrinho
- Pedir um desconto
- Exibir detalhes do cliente

Pense em algumas propriedades e comportamentos que fazem parte da classe Customer da loja Duke's Choice. Agora considere como você escreveria essas informações como uma classe Java.

# Convertendo para uma Sintaxe Java

```
1 public class Customer {  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11 }
```

Propriedades

Comportamentos

Estamos começando a ver a transição para a sintaxe Java.

# Terminologia Java

## Declaração de classe

```
1 public class Customer {
2     public String name = "Junior Duke";
3     public int custID = 1205;
4     public String address;
5     public int orderNum;
6     public int age;
7
8     public void displayCustomer() {
9         System.out.println("Cliente: "+name);
10    } //método final displayCustomer
11 } //classe final Customer
```

**Campos**  
(Propriedades)  
(Atributos)

**Métodos**  
(Comportamentos)

**ORACLE**  
Academy

JFo 2-3  
Introdução aos Conceitos da Programação  
Orientada a Objetos

Copyright © 2022, Oracle e/ou suas empresas afiliadas. Oracle, Java e MySQL são marcas comerciais registradas da Oracle Corporation e/ou de suas empresas afiliadas. Outros nomes podem ser marcas comerciais de seus respectivos proprietários.

27

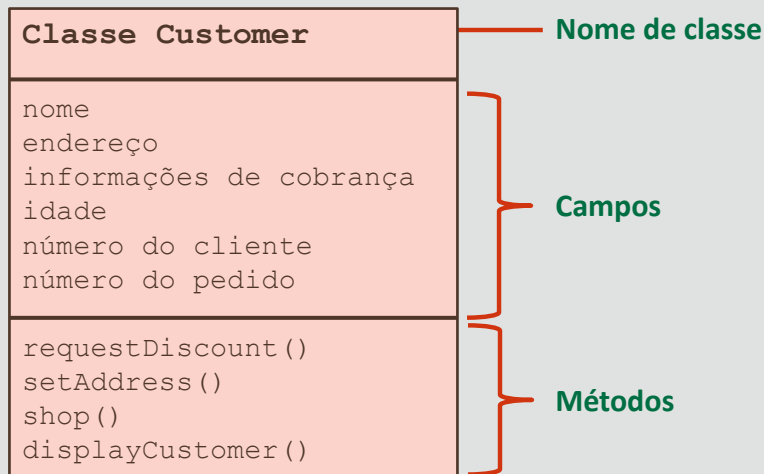
No slide anterior, você identificou algumas propriedades e comportamentos que poderiam estar na classe `Customer`. Este exemplo de código demonstra como as propriedades e os métodos são criados no Java.

Aqui estão os componentes básicos de uma classe Java:

- Declaração de classe. Note que a classe inteira está entre chaves.
- Campos da classe. Esses campos representam as propriedades ou os atributos da classe.
- Métodos da classe. Esses métodos representam os comportamentos ou as operações. Aqui você vê um único método, `displayCustomer`.

**Observação:** no exemplo do código, a palavra “public” é um modificador. Você aprenderá sobre modificadores mais adiante neste curso.

# Modelando Propriedades e Comportamentos



Quando você estiver criando um aplicativo, geralmente será útil criar um modelo simples que descreva os componentes de uma classe. Na tabela, o nome da classe é listado no início. As propriedades ou os campos são listados na segunda linha; os comportamentos, ou métodos, são listados na terceira linha. Fazendo uma analogia em termos gramaticais, considere a classe como um substantivo, as propriedades ou os campos como adjetivos e os comportamentos ou métodos como verbos.

# Campos de Dados

- Os Campos ou Campos de Dados são a terminologia Java oficial
- Eles também são denominados:
  - Propriedades
  - Atributos
  - Membros de Dados
- O Java tem maneiras particulares de representar os dados
  - A Seção 3 fará uma análise mais detalhada dos dados
  - Usaremos o método main para essa investigação
  - Por enquanto, é adequado incluir um grande volume de código no método main
  - MAS não recomendamos em hipótese alguma o uso de um método main muito grande
  - A Seção 4 mostrará como evitar esse cenário

# Resumo

- Nesta lição, você deverá ter aprendido a:
  - Fazer a distinção entre programação procedural e programação orientada a objetos
  - Entender uma classe como um plano gráfico de um objeto
  - Entender como uma classe é usada para criar instâncias de objeto
  - Modelar objetos como uma combinação de...
    - Propriedades (campos de dados)
    - Comportamentos (métodos)



