

The logo for Oracle Academy. 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 centered on a light gray background, which is framed by dark gray horizontal bars at the top and bottom.

ORACLE

Academy

Java Foundations

2-3

Introducción a los Conceptos de Programación Orientada a Objetos

ORACLE
Academy



Copyright © 2022, Oracle y/o sus filiales. Oracle, Java y MySQL son marcas comerciales registradas de Oracle y/o sus filiales. Todos los demás nombres pueden ser marcas comerciales de sus respectivos propietarios.

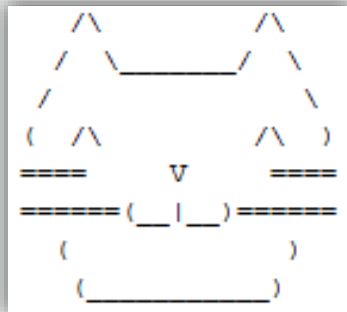
Objetivos

- En esta lección se abordan los siguientes objetivos:
 - Diferenciar entre la programación de procedimiento y la programación orientada a objetos
 - Considerar las clases como planos para objetos
 - Comprender que las clases se utilizan para crear instancias de objetos
 - Objetos de modelos como combinación de...
 - Propiedades (campos de datos)
 - Comportamientos (métodos)



Revisión

- Hasta ahora, hemos visto...
 - Décadas de innovación en las ciencias de la computación
 - Gigabytes de la potencia de computación actual
- Y nos gusta Internet...
 - ¡Hemos conseguido hacer un gato!

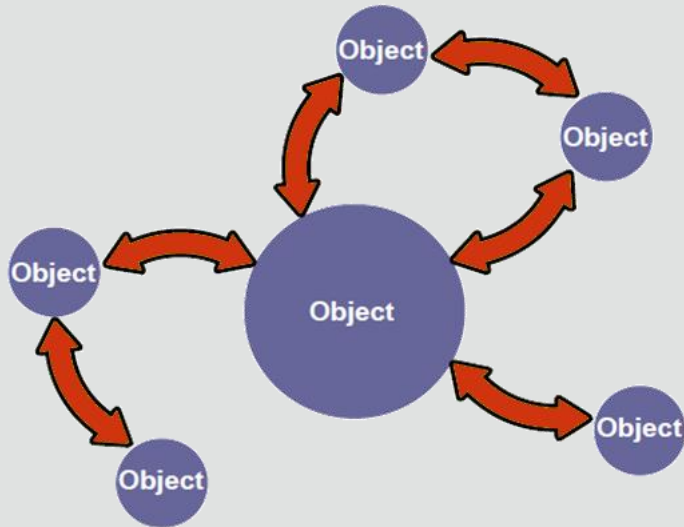


Java Pone Todo un Mundo de Posibilidades a Nuestro Alcance

- Lenguajes de procedimiento...
 - Lea las líneas de una en una
 - El lenguaje C es un lenguaje de procedimiento
- Lenguajes orientados a objetos...
 - Lea las líneas de una en una
 - Modele objetos utilizando código
 - Enfatique las interacciones de objetos
 - Permita las interacciones sin un orden prescrito
 - Los lenguajes Java y C++ son lenguajes orientados a objetos

Programación orientada a objetos

- Interacción de objetos
- No hay ninguna secuencia prescrita



En el diagrama se puede observar cómo la programación orientada a objetos hace hincapié en la interacción de objetos.



Ejercicio 1

- Vaya a <https://objectstorage.uk-london-1.oraclecloud.com/n/lrvrlgaqj8dd/b/Games/o/JavaPuzzleBall/index.html>
- Juegue los rompecabezas básicos del 1 al 5
 - Su objetivo: diseñar una solución que desvíe la bola a Duke
- Tenga en cuenta lo siguiente:
 - ¿Qué objetos ve en el área de juego?
 - ¿Qué ocurre si pone un icono de pared triangular o de pared sencilla en la rueda azul?



ORACLE
Academy

JFo 2-3
Introducción a los Conceptos de Programación
Orientada a Objetos

Copyright © 2022, Oracle y/o sus filiales. Oracle, Java y MySQL son marcas comerciales registradas de Oracle y/o sus filiales. Todos los demás nombres pueden ser marcas comerciales de sus respectivos propietarios.

7

Le animamos a jugar a más rompecabezas que los 5 básicos; estos otros rompecabezas los veremos en lecciones posteriores.

Información sobre Java Puzzle Ball



- Juega a varios rompecabezas
- Familiarícese con la mecánica del juego
- Plantéese preguntas conforme va jugando
- Escuche la información de la lección relativa a lo que ha observado
- Intente asimilar los conceptos de Java sirviéndose de sus observaciones



ORACLE
Academy

JFo 2-3
Introducción a los Conceptos de Programación
Orientada a Objetos

Copyright © 2022, Oracle y/o sus filiales. Oracle, Java y MySQL son marcas comerciales registradas de Oracle y/o sus filiales. Todos los demás nombres pueden ser marcas comerciales de sus respectivos propietarios.

8

La mecánica de este juego sirve para entender los conceptos de Java. Lo que realmente importa es familiarizarse con la mecánica del juego, más que resolver todos los rompecabezas. No se preocupe si la relación entre la mecánica del juego y los conceptos de Java no es obvia de inmediato. Le proporcionaremos más información al respecto más adelante; es ahí cuando la mayoría de la gente empieza a descubrir la conexión. Conforme vaya entendiendo cómo funciona el juego, podrá empezar a aplicar lo aprendido como base para comprender conceptos difíciles de Java.



Tipos de Objeto

- ¿Qué objetos ha encontrado en el área de juego?

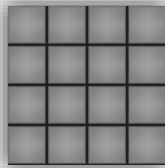
- Bola



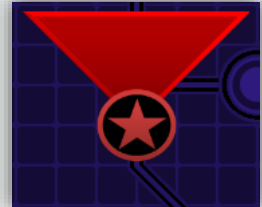
- Duke



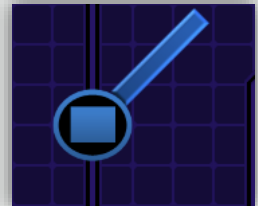
- Geometría de nivel



- Deflector rojo



- Deflector azul

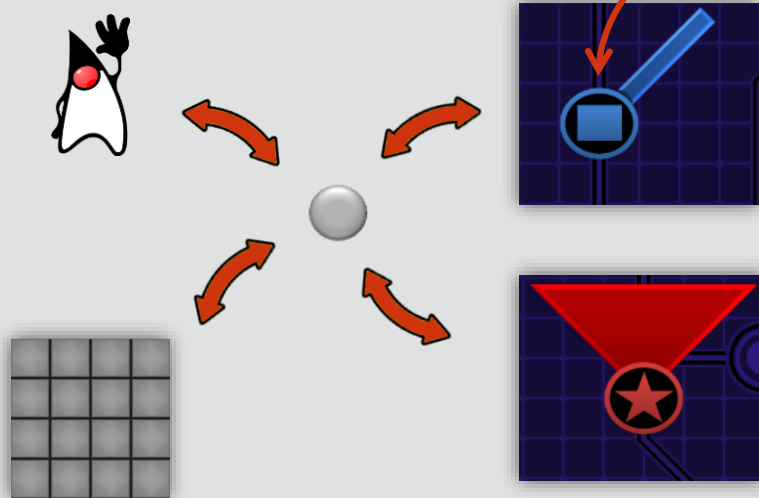




Interacción de Objetos

- Interacción de objetos
- No hay ninguna secuencia prescrita

Veamos más detenidamente este objeto



ORACLE
Academy

JFo 2-3
Introducción a los Conceptos de Programación
Orientada a Objetos

Copyright © 2022, Oracle y/o sus filiales. Oracle, Java y MySQL son marcas comerciales registradas de Oracle y/o sus filiales. Todos los demás nombres pueden ser marcas comerciales de sus respectivos propietarios.

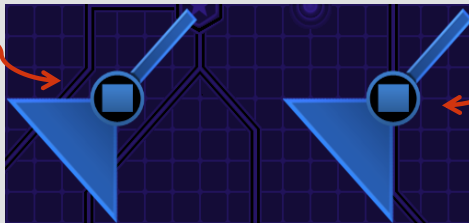
10



Objetos Deflectores Azules

- ¿Qué ocurre si pone un icono de pared triangular o de pared sencilla en la rueda azul?
- Todas las instancias de los objetos deflectores azules tienen una pared
- Las paredes proporcionan comportamientos que desvían e interactúan con la bola
- Todas las instancias de los deflectores azules tienen estos mismos comportamientos

WebCenter
Sites



WebCenter
Sites

ORACLE
Academy

JFo 2-3
Introducción a los Conceptos de Programación
Orientada a Objetos

Copyright © 2022, Oracle y/o sus filiales. Oracle, Java y MySQL son marcas comerciales registradas de Oracle y/o sus filiales. Todos los demás nombres pueden ser marcas comerciales de sus respectivos propietarios.

11

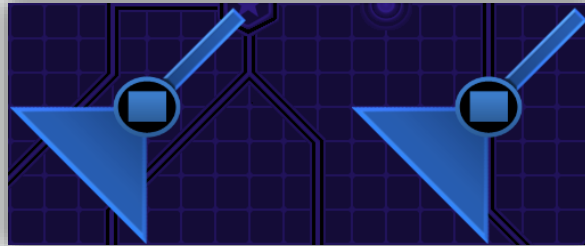
Un deflector azul es un objeto, y todas las instancias de estos objetos compartirán el mismo comportamiento al interactuar con la bola. Estos comportamientos incluyen los desvíos realizados mediante paredes triangulares o sencillas.



Descripción de los Deflectores Azules

- Propiedades:

- Color
- Forma
- Posición x
- Posición y



- Comportamientos:

- Hacer sonido de ping
- Parpadear
- Desviar bola (mediante pared sencilla)
- Desviar bola (mediante pared triangular)



Descripción de una Bola

- Propiedades:

- Direction
- Posición x
- Posición y

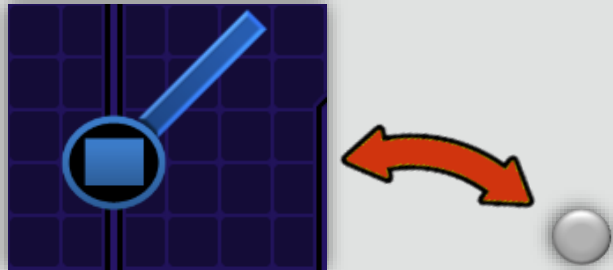


- Comportamientos:

- Emitir sonido ping
- Cambiar dirección
- Cambiar posición x
- Cambiar posición y

Deflector Azul e Interacción con la Bola

- La interacción tiene lugar cuando el deflector azul desvía la bola Cuando esto sucede...
- Cambian las propiedades de la bola:
 - La bola cambia de dirección
 - Las posiciones x e y de la bola van a cambiar
- El deflector azul tiene los siguientes comportamientos:
 - Emite un sonido ping
 - Parpadea



Todos los deflectores azules tienen la capacidad de emitir sonidos ping, parpadear e interactuar con la bola.



¿Por Qué Es Importante?

- Hemos observado algunos aspectos importantes relativos a la programación orientada a objetos
- Recuerde estas observaciones a medida que las lecciones y los ejercicios se vuelvan más técnicas
 - Los objetos se puede describir como una combinación de propiedades y comportamientos
 - Puede haber varias instancias del mismo tipo de objeto
 - Todas las instancias de un objeto comparten los mismos comportamientos
 - Puede que los objetos interactúen entre sí, y que ello afecte a sus propiedades y desencadenen otros comportamientos

Otro Ejemplo

- Propiedades:

- Nombre
- Edad
- Raza
- Comida favorita



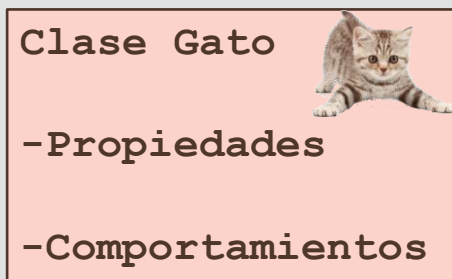
- Comportamientos:

- Maullar
- Jugar
- Lavarse
- Comer
- Cazar

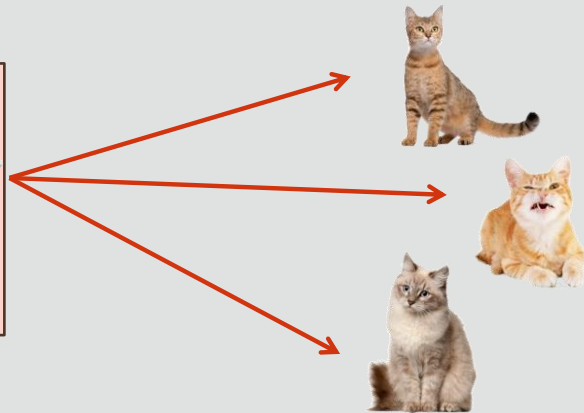
Observe la sutil diferencia entre Comida favorita (que corresponde a una descripción) y Comer (que es un verbo).

Clases e Instancias

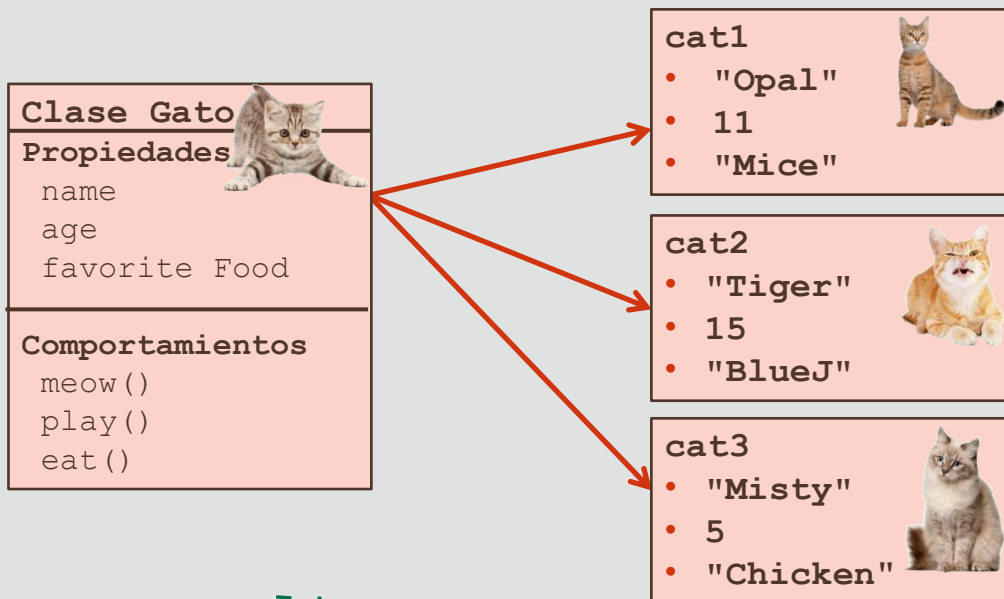
- La combinación de propiedades y comportamientos...
 - Se denomina clase
 - Es el plano o la receta de un objeto
 - Se utiliza para crear instancias de objetos



Instancias de objetos



Creación de Nuevas Instancias a partir de un Plano



Todas las instancias de gato tienen la capacidad de maullar, jugar y comer

Estrategia Orientada a Objetos

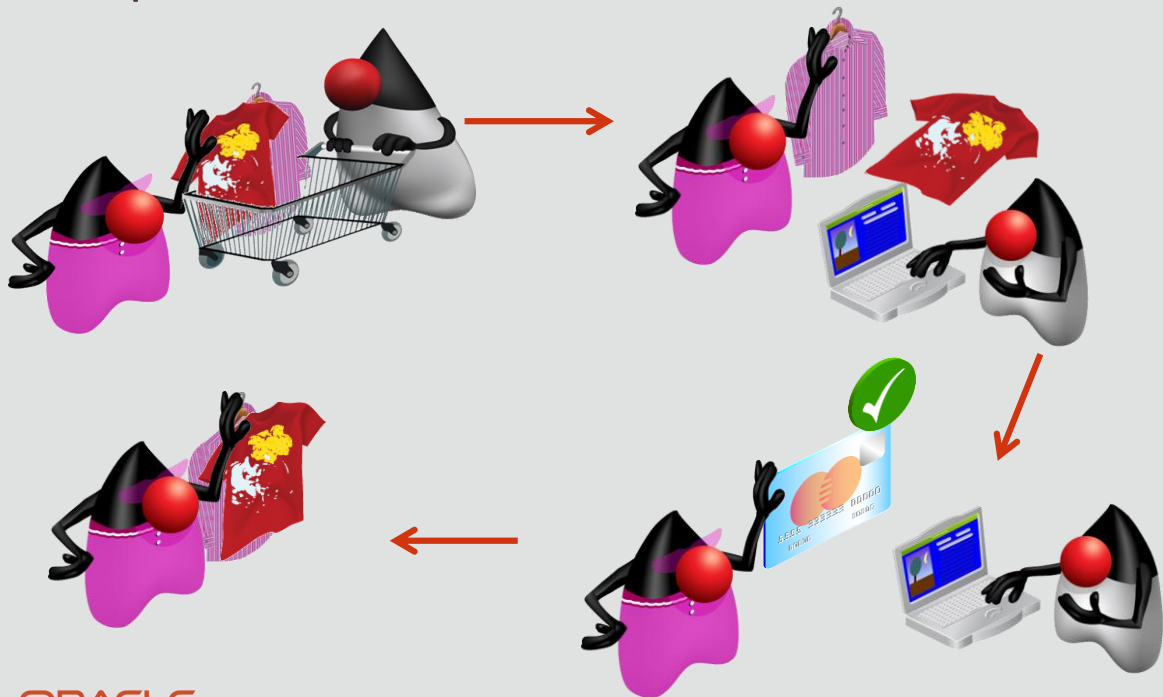
- ¿Cómo se pueden escribir programas que tengan este grado de flexibilidad?
- Cuando se le ocurra una idea para un programa o tenga un requisito para este...
 - Considere qué tipo de objetos puede haber en este programa
 - Tenga en cuenta las propiedades y los comportamientos de estos tipos de objetos
 - Plántese cómo interactúan los objetos

Cuando se diseña un programa Java, en primer lugar hay que identificar los objetos; después, determinar las características o propiedades del objeto; y, a continuación, determinar los comportamientos o las operaciones de este. También resulta útil tener en cuenta cómo interactúan los objetos o como su interacción afecta a las propiedades de estos.

El último paso consiste en convertir dicho análisis en código Java para crear la aplicación.

Enseñaremos este último paso casi al final de la lección para demostrar que es posible. No se le pedirá escribir su propio código para una clase Java hasta más avanzado el curso.

Compra Online en Duke's Choice



ORACLE
Academy

JFo 2-3
Introducción a los Conceptos de Programación
Orientada a Objetos

Copyright © 2022, Oracle y/o sus filiales. Oracle, Java y MySQL son marcas comerciales registradas de Oracle y/o sus filiales. Todos los demás nombres pueden ser marcas comerciales de sus respectivos propietarios.

20

Observemos un ejemplo de carro de compra online. Imagine una tienda online llamada Duke's Choice. Su cliente número uno es su madre, la Sra. Duke. Conforme la Sra. Duke va viendo productos que le gustan, los va colocando en un carro de la compra. A la Sra. Duke le gustan las camisas, así que va metiendo camisas en el carro de la compra. Una vez lleno el carro, querrá finalizar la compra. Durante el proceso de finalización de compra, se asigna la compra a una tarjeta de crédito, cuyos datos se verifican y, a continuación, la Sra. Duke recibe un número de pedido para que pueda realizar un seguimiento de este o devolverlo.

En calidad de desarrollador de software, cuando se le presente un caso como el de Duke's Choice para una aplicación que tiene que desarrollar, puede analizar la situación desglosándola en pasos y definiendo los objetos que participan.

Características de los Objetos

- Los objetos pueden ser físicos o conceptuales
- Los objetos tienen propiedades:
 - Precio
 - Lista
 - Color
- Los objetos tienen comportamientos:
 - Comprar
 - Colocar producto en el carro
 - Pagar



Físicas:
camisa



Conceptuales:
cuenta online



El valor de la
propiedad de color es
rojo



Sra. Duke

Para validar objetos en un dominio de problemas como el proceso de pedidos de Duke's Choice, es necesario identificar las propiedades de todos los objetos.

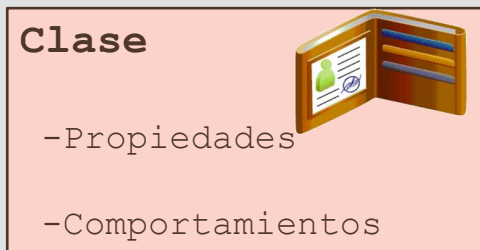
Los objetos pueden ser físicos o conceptuales. Una camisa es un ejemplo de objeto físico. La cuenta de la tarjeta de crédito del cliente es un ejemplo de un objeto conceptual, porque no es algo que se pueda tocar físicamente.

Los objetos tienen propiedades (atributos) como el tamaño, el nombre, la forma, etc., que representan el estado del objeto. Por ejemplo, las personas tienen nombres (Sra. Duke), y los objetos pueden tener una propiedad de color. El valor de todas las propiedades de un objeto se suele denominar *estado actual* del objeto. Un objeto puede tener una propiedad de color con el valor de rojo y una propiedad de tamaño con un valor de grande.

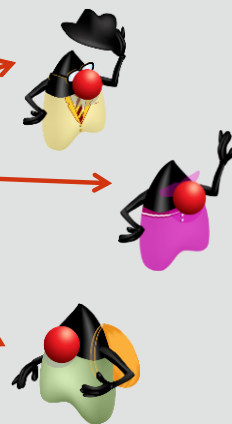
Los objetos también tienen comportamientos (cosas que pueden hacer) como, en nuestro ejemplo, comprar, colocar un producto en el carro y pagar.

Clases e Instancias

- Recuerde que una clase...
 - Es el plano o la receta de un objeto
 - Describe las propiedades y los comportamientos de un objeto
 - Se utiliza para crear instancias de objetos



Instancias de objetos



Acabamos de comentar algunos de los objetos, características y comportamientos del caso Duke's Choice. A continuación vamos a ver uno de los objetos de Duke's Choice, el cliente, y la función que desempeña en la tienda. El cliente es la clase, y una clase es el plano o la receta de un objeto. Las clases describen las propiedades y los comportamientos de los objetos. Las clases sirven para crear instancias de objetos, como las tres instancias de objetos del cliente, que se muestran en las tres imágenes.

Ejercicio 2, Parte 1

- Dada la siguiente situación, ¿qué objetos podría modelar para completar el programa?
 - Diseñe un programa para una máquina clasificadora de monedas
 - Esta máquina debe medir, contar y clasificar las monedas en función de su tamaño o valor
 - También debe imprimir un recibo
- Enumere al menos 3 objetos que tendría que modelar para este ejemplo:
 -
 -
 -



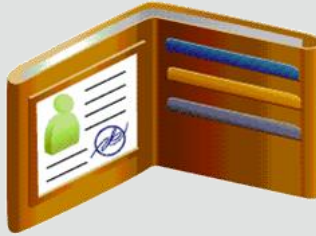
Ejercicio 2, Parte 2

- | | | |
|--|---|---|
| <ul style="list-style-type: none">• Elija un objeto de la Parte 1• ¿Qué propiedades y comportamientos de este objeto podría incluir en su programa? | <ul style="list-style-type: none">• Propiedades:<ul style="list-style-type: none">——— | <ul style="list-style-type: none">• Comportamientos:<ul style="list-style-type: none">——— |
|--|---|---|

Propiedades y Comportamientos de los Clientes

- Propiedades:

- Nombre
- Dirección
- Edad
- Número de pedido
- Número de cliente



- Comportamientos:

- Comprar
- Definir dirección
- Agregar un producto al carro
- Pedir un descuento
- Mostrar los datos del cliente

Piense en algunas propiedades y comportamientos que pertenezcan a la clase de cliente de Duke's Choice. Piense en cómo escribiría esta información como clase Java.

Traducción a Sintaxis Java

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

Estamos comenzando a ver la traducción a sintaxis Java.

Terminología Java

Declaración de clase

```
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("Customer: "+name);  
10    }  
11 }
```

**Campos
(Propiedades)
(Atributos)**

**Métodos
(Comportamientos)**

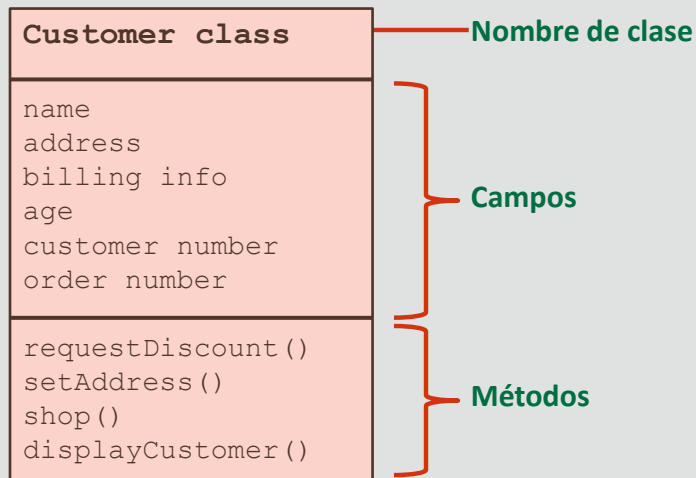
En la diapositiva anterior, ha especificado algunas propiedades y comportamientos que pueden estar en la clase de cliente. Este ejemplo de código muestra cómo se crean las propiedades y los métodos en Java.

Estos son los componentes básicos de una clase Java:

- Declaración de clase. Tenga en cuenta que toda la clase está comprendida entre corchetes angulares.
- Campos de la clase. Estos campos representan las propiedades o los atributos de la clase.
- Métodos de la clase. Estos métodos representan los comportamientos u operaciones. Aquí solo puede ver un método, `displayCustomer`.

Note: En el ejemplo de código, la palabra “public” es un modificador. Explicaremos los modificadores más en detalle más adelante.

Modelación de Propiedades y Comportamientos



Al diseñar una aplicación, suele resultar útil crear un modelo sencillo en el que se describan los componentes de una clase. En la tabla, el nombre de clase aparece en la parte superior. Las propiedades o los campos aparecen en la segunda fila; los comportamientos o métodos, en la tercera. Si considera esta modelación en términos lingüísticos, la clase correspondería a los sustantivos, las propiedades a los adjetivos, y los comportamientos o métodos a los verbos.

Campos de Datos

- Los Campos o Campos de datos son la terminología oficial de Java
- También se denominan:
 - Propiedades
 - Atributos
 - Miembros de datos
- Java tiene una forma muy particular de representar los datos
 - En la sección 3, se verá esto en más detalle
 - Utilizaremos el método principal para realizar esta investigación
 - Por ahora no pasa nada por incluir mucho código en el método principal
 - PERO no se recomienda en absoluto emplear un método principal grande.
 - En la sección 4 se explica más en detalle cómo evitar esta situación

Resumen

- En esta lección, debe haber aprendido lo siguiente:
 - Diferenciar entre la programación de procedimiento y la programación orientada a objetos
 - Considerar las clases como planos para objetos
 - Comprender que las clases se utilizan para crear instancias de objetos
 - Objetos de modelos como combinación de...
 - Propiedades (campos de datos)
 - Comportamientos (métodos)



