Recibe una cálida:

¡Bienvenida!

Te estábamos esperando 😁

alkemy

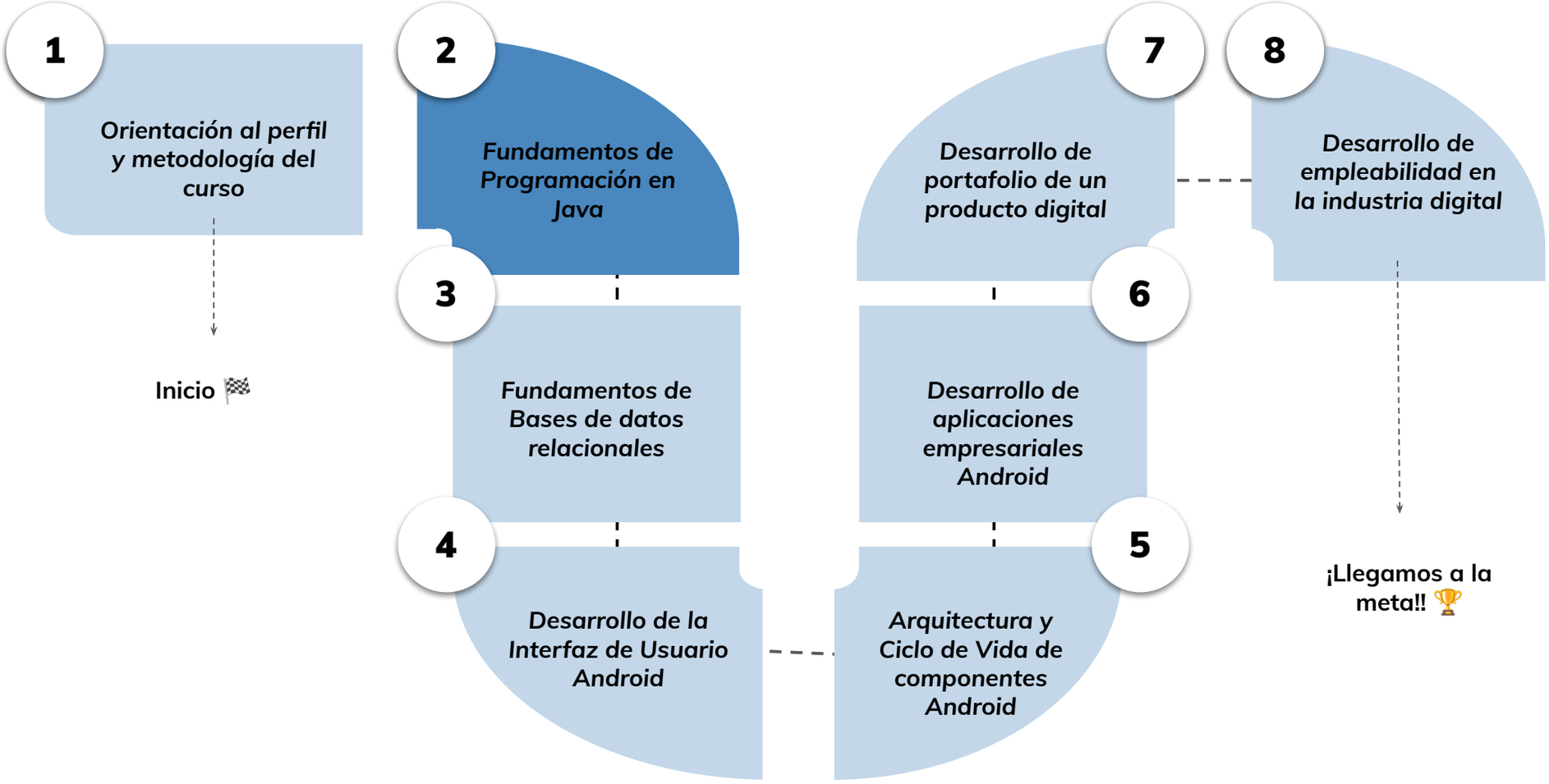
Clase grabada

El paradigma de orientación a objetos

AE5.1: Utilizar elementos de la programación orientada a objetos para la implementación de una pieza de software que da solución a un problema de baja complejidad.

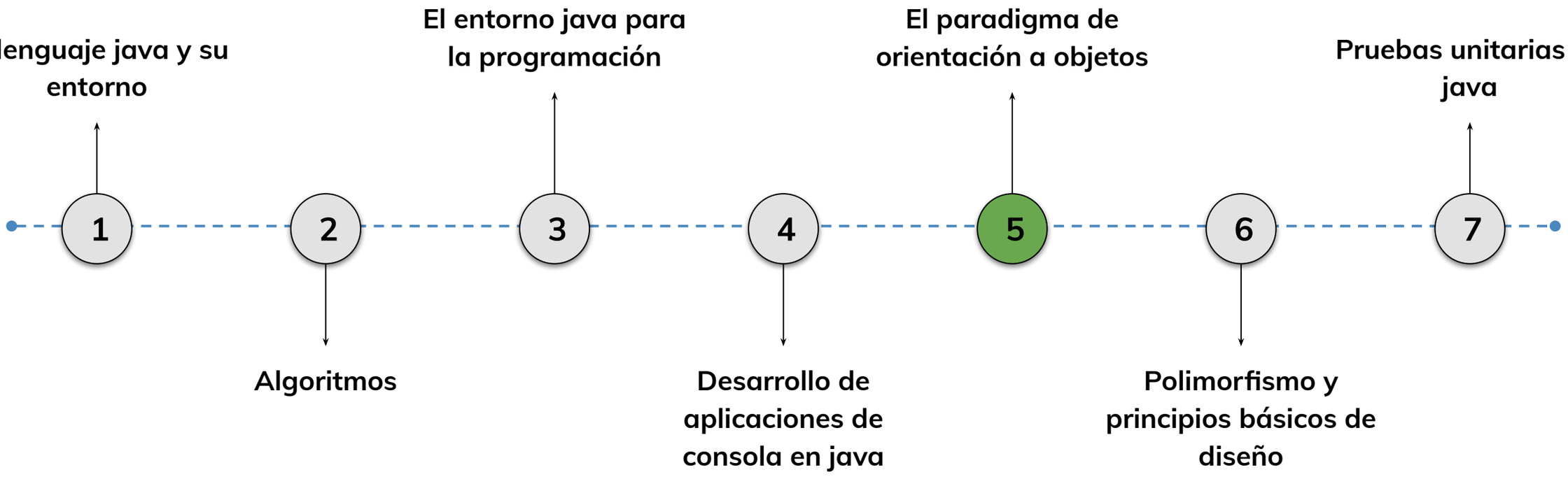
Hoja de ruta

*¿Cuáles **módulos** conforman el programa?*



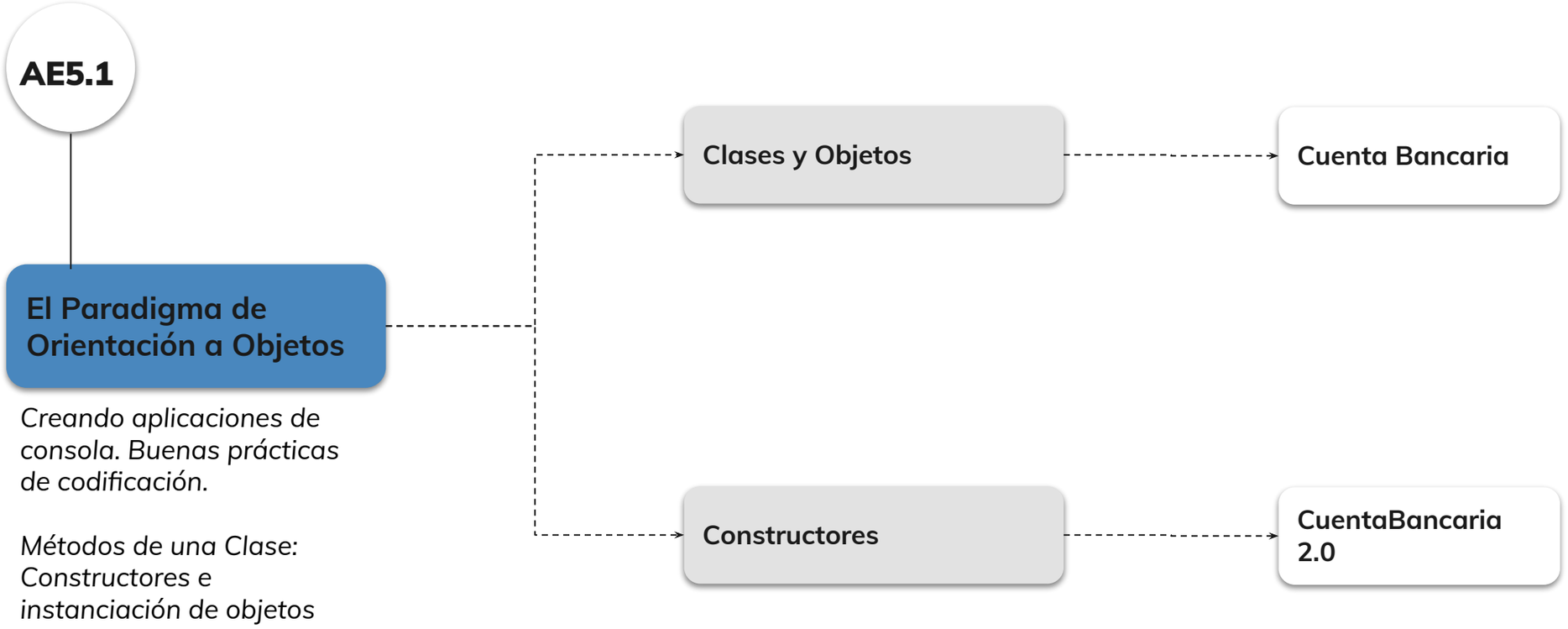
Roadmap de lecciones

*¿Cuáles **lecciones** estaremos estudiando en este módulo?*



Learning Path

¿Cuáles temas trabajaremos hoy?



alkemy

Objetivos de aprendizaje

¿Qué aprenderás?

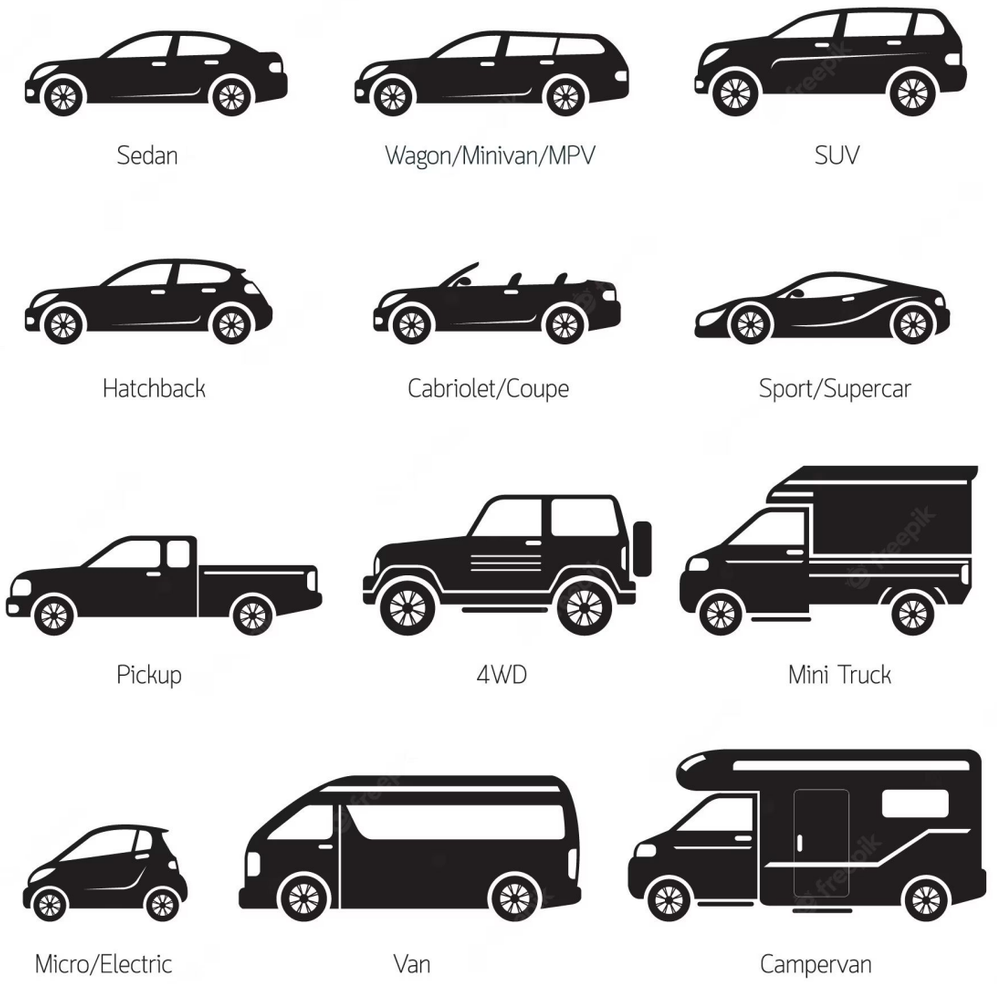
- Reconocer la importancia de la Orientación a Objetos.
- Comprender el concepto e implementación de las Clases y los Objetos.
- Comprender e implementar los métodos constructores
- Aprender a instanciar un objeto.

< > Rompehielo🥶

¿Y tú que piensas?: 🙌

Vean con atención la imagen y respondan en el chat y levantando la mano:

- ¿Son los mismos objetos?
- ¿Qué características tienen en común?
- ¿En qué se diferencian?
- ¿Cómo los clasificarías/agruparías?



El Paradigma de Orientación a Objetos

El Paradigma de Orientación a Objetos

¿Qué es un Paradigma de programación?

Es una manera o estilo de programación. Existen diferentes formas de diseñar un programa y varios modos de trabajar para obtener los resultados que necesitan los programadores.

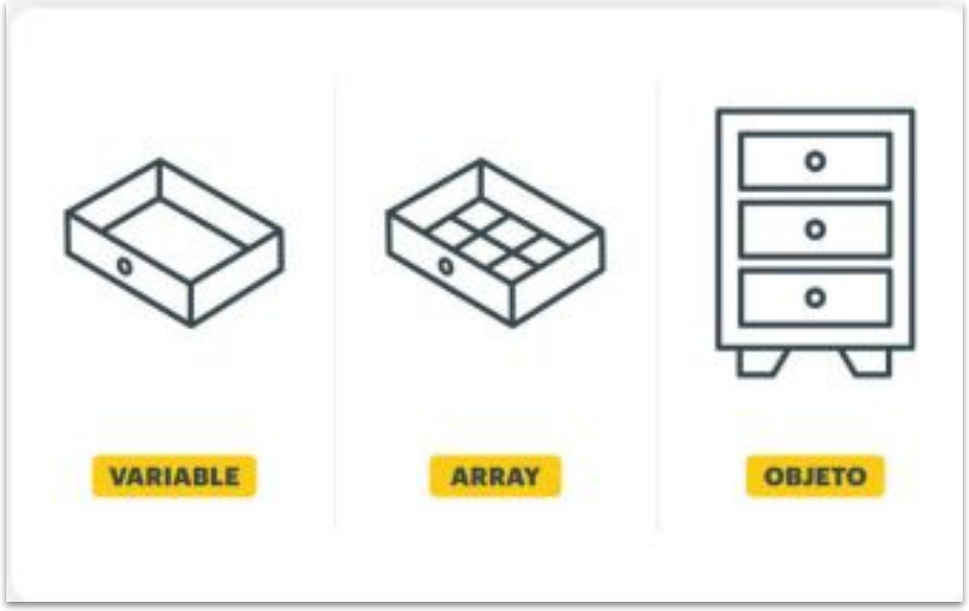
La Programación Orientada a Objetos (POO) es un paradigma que modela elementos del mundo real como objetos que tienen propiedades y comportamientos. Esto permite representar entidades y sus características para crear programas más intuitivos.

También permite organizar el código en módulos o clases reutilizables, y sus respectivas instancias y objetos, lo que facilita la creación y mantenimiento de programas complejos.

El Paradigma de Orientación a Objetos

La Programación Orientada a objetos permite que el código sea reutilizable, organizado y fácil de mantener.

Sigue el principio de desarrollo de software utilizado por muchos programadores DRY (Don’t Repeat Yourself), para evitar duplicar el código y crear de esta manera programas eficientes.



Clases

Son las plantillas o moldes que utilizaremos para crear objetos. Definen las propiedades y comportamientos que los objetos de esa clase tendrán. Entonces, una clase es una especie prototipo de objetos: define los atributos que componen ese tipo de objetos y los métodos que pueden emplearse para trabajar con esos objetos.

En su forma más simple, una clase se define por la palabra reservada class seguida del nombre de la clase. El nombre de la clase debe empezar por mayúscula. Si el nombre es compuesto, cada palabra debe empezar por mayúscula.

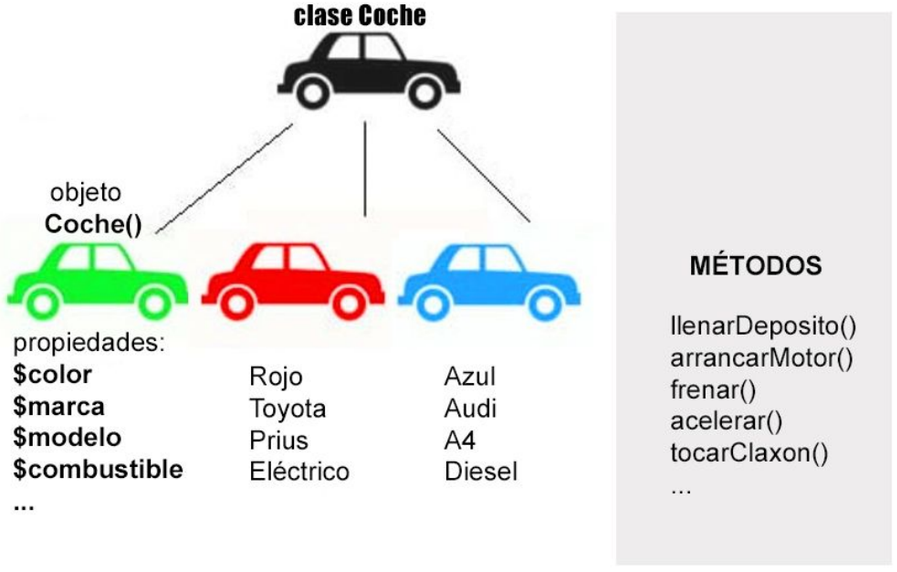
La definición de la clase se pone entre las llaves de apertura y cierre.

```
public class NombreClase {
	// atributos ;
	// constructores ;
	// métodos propios ;
}
```

Objetos

Son instancias específicas de una clase. Se crean a partir de la clase y tienen su propio estado y comportamiento.

- Estado: viene dado por sus atributos, que almacenan los datos del objeto. Por ejemplo, el color y marca de un coche.
- Comportamiento: viene dado por sus métodos que son las operaciones que puede realizar el objeto. Por ejemplo, el coche puede frenar y acelerar.



Estado de un objeto: Atributos

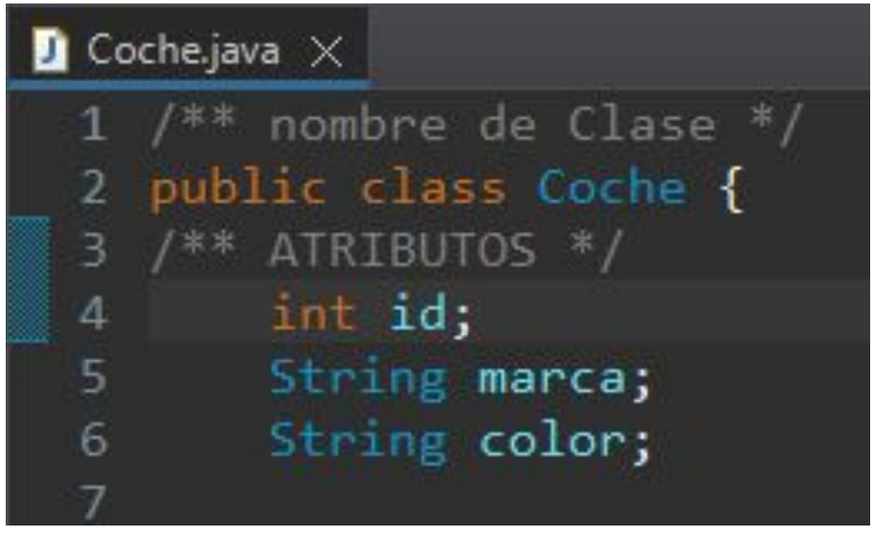
Los atributos son características comunes a todos los objetos.

El estado o información de un objeto se almacena en atributos. Los atributos pueden ser de tipos primitivos de Java o del tipo de otros objetos.

La declaración de un atributo de un objeto tiene la siguiente forma:

<tipo de dato> <nombre del atributo> ;

<tipo>: indica la clase a la que pertenece el atributo definido.
<nombre>: puede ser cualquier identificador válido y denomina el atributo que está siendo declarado.



LIVE CODING

Billetera Virtual:

Vamos a crear una Wallet desde el comienzo! Para esto, crearemos una clase llamada Cuenta que manipulara los siguientes datos:

1. *Atributos de la clase Cuenta:*
   -Número de cuenta (int)
   -Titular (String)
   -Saldo (double)
1. *Crear un objeto billetera1 y asignarle valores a estos atributos.*

Tiempo: 25 Minutos

alkemy

LIVE CODING

https://gist.github.com/Mariocanedo/3719eb9cdaef3efba361af4bcda26f5a

Ejercicio N° 1

Cuenta Bancaria

Cuenta Bancaria

Hora de codear: 🙌

La POO nos ayuda a programar pensando y modelando la realidad. Vamos a poner en práctica lo aprendido.

Consigna: ✍️

- Crea un proyecto llamado Banco.
- Crea una Clase llamada CuentaBancaria
- Agrega los siguientes atributos y dales un valor:
    - entero número de cuenta
    - doble saldo actual
    - cadena titular

Tiempo🕐: *25 minutos*

Momento:

Time-out!

⏰5 -10 min.

alkemy

Métodos de Clase: Constructores

Métodos de una Clase

¿Qué son?:

Son funciones o procedimientos que representan el comportamiento o las acciones que los objetos de esa clase pueden realizar.

Los métodos pueden aceptar parámetros como entrada, realizar cálculos, modificar el estado de los objetos y devolver o no resultados como salida.

Ayudan a organizar y modularizar el código, ya que permiten dividir la lógica del programa en tareas más pequeñas y manejables.

```
[<modificador>] <nombre de método> ( <argumento>* ) {
    <sentencia>;
}
```

Métodos Constructores

¿Qué son?:

Los constructores son métodos propios del objeto que permiten la creación de dicho objeto. A la creación de un objeto se le denomina instanciación.

Los constructores se encargan de dar un estado inicial a nuestro objeto. Estos métodos tienen como característica principal tener el mismo nombre que la propia clase. Por lo tanto, debemos tener un método dentro de la clase “Coche” que se llame igual que ella.

Métodos Constructores

El método constructor se ejecuta cada vez que se instancia un objeto de la clase. Este método se utiliza para inicializar los atributos del objeto que se crea.

Para diferenciar entre los atributos del objeto y parámetros del método constructor, se utiliza la palabra this.

La instancia de un objeto consiste en asignar un espacio de memoria al que se hace referencia con el nombre del objeto.

```java
/**
 * constructor vacío o por defecto
 */
    public Coche () {

    }
/**
 * constructor parametrizado
 */
    public Coche(int id, String marca, String color) {
        this.id = id;
        this.marca = marca;
        this.color = color;
    }

}

```

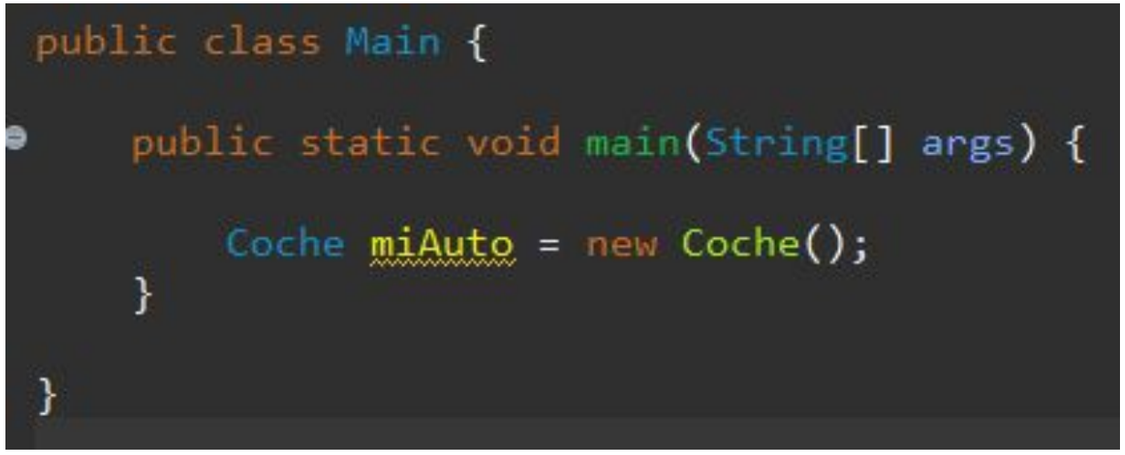
Métodos Constructores

Para crear objetos, basta con declarar una variable de alguno de los tipos de las clases definidas:

NombreClase nombreObjeto = new nombreClase();

Para crear el objeto y asignar un espacio de memoria es necesario utilizar el operador new. El operador new instancia el objeto y reserva espacio en memoria para los atributos y devuelve una referencia que se guarda en la variable.

A partir de este momento los objetos ya pueden ser referenciados por su nombre (miAuto).



Métodos Constructores

También podemos utilizar el método constructor parametrizado para instanciar un objeto, escribiendo los parámetros que le darán valor a cada atributo.

```
2  public class Main {
3
4      public static void main(String[] args) {
5
6          Coche miAuto = new Coche(123, "Nissan", "Negro");
7      }
8
9  }
10
```

LIVE CODING

Billetera Virtual:

Ahora ya podemos construir un objeto Cuenta! Veamoslo practicando:

1. *Agregar métodos constructores que inicialicen los atributos numeroCuenta, saldo y nombre(titular).*
2. *Instanciar una Cuenta.*
3. *Mostrar los valores de los atributos por pantalla.*

Tiempo: 25 Minutos

LIVE CODING

https://gist.github.com/Mariocanedo/52256915d9d37b0330cec72d5ec32019

Ejercicio N° 1

CuentaBancaria 2.0

CuentaBancaria 2.0

Contexto: 🙌

Vamos a mejorar la clase CuentaBancaria creada en el ejercicio de la clase anterior.

Consigna: ✍️

1. Agregar métodos constructores
2. Instanciar un objeto CuentaBancaria y darle valor a sus atributos.
3. Mostrar los datos por pantalla.

Tiempo🕐: *25 minutos*

Resumen

¿Qué logramos en esta clase?

- ✓ ***Comprender la importancia de la POO***
- ✓ ***Conocer la implementación de Clases y el estado y comportamiento de los Objetos***
- ✓ ***Comprender el concepto e implementación de los métodos constructores***
- ✓ ***Aprender cómo instanciar un objeto de Clase***

alkemy

¿Alguna consulta?

alkemy

¡Ponte a prueba!

Momento de ejercitación

Te invitamos a aprovechar esta última sección del espacio sincrónico para realizar de manera individual las ***actividades disponibles en la plataforma.*** *Estas propuestas son clave para afianzar lo trabajado y* ***forman parte obligatoria del recorrido de aprendizaje.***

💪 Análisis de caso ———— 💪 Selección Múltiple

💪 Comprensión lectora

Si al resolverlas surge alguna duda, compartela o tráela al próximo encuentro sincrónico.

alkemy

< ¡Muchas gracias! >