



TALLER DE PARADIGMAS

Clase N°1

Magni Gastón

Massiatti Alexander

Fermani Julián

INTRODUCCION
INTRODUCCION
INTRODUCCION
INTRODUCCION
INTRODUCCION
INTRODUCCION

¿QUE ES UNA CLASE?

Las clases son modelos del mundo real que capturan la estructura y comportamiento compartidos por una colección de objetos de un mismo tipo (Seidl et al., 2015)

```
3 public class Ejemplo { no usages
4 }
```

ACTIVIDAD

- Detección de clases en un enunciado.
- Detección de atributos, su visibilidad y su tipo.
- Detección de métodos, parámetros de entrada, return salida.
- Uso de constructores para la creación de objetos.
- Utilización del método de representación de un objeto (toString()).



Imagina que estás desarrollando una aplicación para gestionar las cuentas bancarias de los clientes de una entidad financiera. Se requiere la creación de cuentas bancarias, donde los datos personales del cliente, el saldo disponible y otras operaciones estén controlados de manera adecuada.

El sistema debe permitir la creación de cuentas bancarias, considerando que inicialmente la información asociada a la cuenta puede estar incompleta o ser indefinida. Esta deberá contar con una serie de funcionalidades para interactuar con la cuenta de manera segura, permitiendo realizar operaciones como la modificación del saldo únicamente a través de depósitos o retiros, pero nunca mediante un acceso directo.

Además, la aplicación debe ofrecer la capacidad de visualizar de forma clara la información clave de cada cuenta. El sistema también deben manejar situaciones en las que se realicen intentos de ingresar cantidades inválidas, o cuando la cuenta quede con saldo negativo tras una retirada (Las cuentas pueden tener números rojos). El diseño debe permitir una gestión robusta de estas eventualidades.

¿QUE ES UN ATRIBUTO?

Las clases contienen una colección de atributos, que representan propiedades de los objetos. Dichos atributos tienen un tipo, el cual puede ser un tipo primitivo de dato de Java u objeto.

9

```
String atributoEjemplo = "valor";
```

TIPOS DE DATOS EN JAVA

Java cuenta con una variedad de tipos de datos o datos primitivos, los podemos diferenciar para su uso.

Datos relacionados con numeros enteros: byte, short, int, long

Datos relacionados con numeros con coma: float, double

Datos relacionados con cadenas de caracteres: char

Datos relacionados con valores booleanos: boolean

Los datos primitivos que utilizaremos son:

- *int*
- *float*
- *boolean*

(También utilizaremos String que en sí no es un dato primitivo)

¿QUE ES UN MÉTODO?

Los métodos de los objetos en el Paradigma Orientado a Objetos son las acciones o comportamientos que los objetos de una clase pueden realizar. Son funciones que están asociadas a una clase y permiten interactuar con las propiedades de sus objetos o modificar su estado.

5	<code>public void <u>metodoEjemplo()</u>{</code>
6	<code><u>//comportamiento</u></code>
7	<code>}</code>

¿COMO SE COMPONE UN MÉTODO?

TIPO DE RETORNO

Representa el tipo de retorno del método, el cual puede ser un tipo de dato primitivo, un objeto, o si no devuelve nada debe colocarse la palabra reservada void.

NOMBRE DEL MÉTODO

Representa la palabra clave con la que el método será convocado posteriormente.

PARAMETROS

Es un listado de parámetros separados por comas, que se utilizan en el cuerpo del método para cumplir la finalidad del método. Para cada parámetro se debe colocar previamente su tipo de dato. Un método puede no tener parámetros.

MÉTODOS ESPECIALES

CONSTRUCTOR

Un constructor sirve para inicializar el objeto y establecer sus propiedades y valores predeterminados. Tiene el mismo nombre que la clase y no cuenta con ningún valor de retorno, ya que su función principal es inicializar el objeto y no devolver ningún valor

```
public Persona() {  
}
```

MAIN

*El método main es el punto de entrada de un programa.
Es la función que se ejecuta primero cuando inicias un programa. Es el punto inicial desde donde se ejecutan todas las demás funciones, clases, o métodos.*

```
public class Main {  
    public static void main(String [] args){
```

TOSTRING

Es un método heredado de Object pero es una práctica común sobre- escribir este método en tus propias clases para proporcionar una representación más significativa y legible de los objetos.

```
@Override  
public String toString() {  
    return "Cuenta{" +  
        "Número de Cuenta=" + numeroCuenta +  
        ", Titular='" + titular + '\'' +  
        ", Saldo=" + saldo +  
        '}';  
}
```



MÉTODOS DE ACCESO

Java posee modificadores de acceso para restringir el alcance de una clase, constructores, atributos, métodos y variables.

POR DEFECTO

Cuando no se especifica el modificador de acceso se dice que está predeterminado. Los elementos que no se declaran utilizando modificadores de acceso son accesibles solo dentro del mismo paquete.

PROTEGIDO

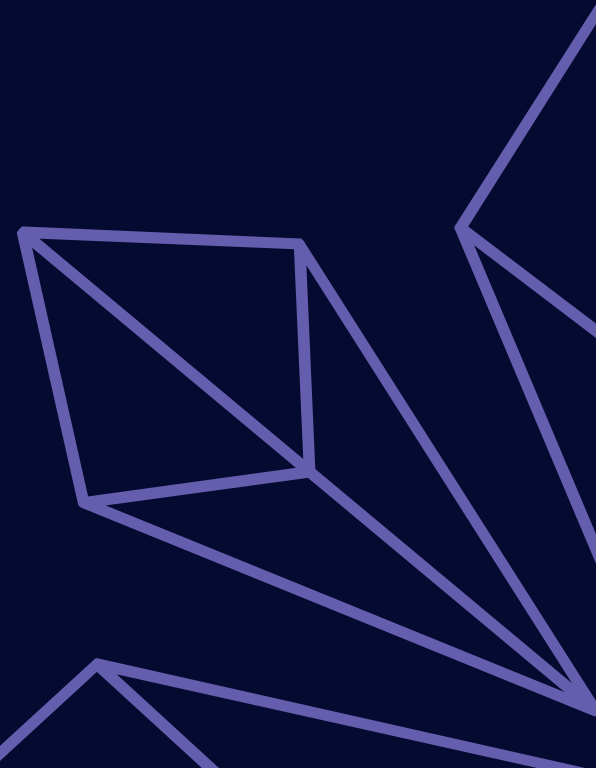
*El modificador de acceso protegido se especifica mediante la palabra clave **protected**. Los elementos declarados como protegidos son accesibles dentro del mismo paquete o subclases en paquetes diferentes.*

PÚBLICO

*El modificador de acceso público se especifica mediante la palabra clave **public**. El modificador de acceso público tiene el alcance más amplio entre todos los demás modificadores de acceso. Los elementos que se declaran públicos son accesibles desde cualquier lugar del programa.*

PRIVADO

*El modificador de acceso privado se especifica usando la palabra clave **private**. Los elementos declarados como privados son accesibles solo dentro de la clase en la que se declaran. Cualquier otra clase del mismo paquete no podrá acceder a estos elementos.*





GRACIAS