

ESCUELA DE FORMACIÓN DE TECNÓLOGOS

PROGRAMACION ORIENTADA A OBJETOS

ASIGNATURA:	POO
PROFESOR:	Ing. Yadira Franco R
PERÍODO ACADÉMICO:	2025-A

TAREA S2_1

TÍTULO:

CLASES , OBJETOS, ATRIBUTOS Y METODOS

INGRESO DE DATOS

Andrés Panchi



2025-A

REVISAR LOS VIDEOS

<https://www.youtube.com/watch?v=pXX3c8BloY0>

<https://www.youtube.com/watch?v=fdXkiuVHp4>

This en el Metodo constructor

https://www.youtube.com/watch?v=aUX_rQO1ac8

Método Constructor

<https://www.youtube.com/watch?v=4plcVlxh0lw>

Creación de Objetos

<https://www.youtube.com/watch?v=zhUTqZk3-ks>

NO OLVIDAR ¿Qué debe quedar muy claro?

- Primero va la clase.
- Dentro de la clase van los atributos y métodos.
- Después se crea el constructor.
- Luego en main() se instancia el objeto con new.
- Se usan los atributos y métodos a través del objeto.

INVESTIGE Y explique claramente PASO A PASO:

- Cómo se crea una clase.
- Cómo se crea un constructor con y sin parámetros
- Como se crear métodos
- Cómo se instancia un objeto y donde se lo ubica.
- Cómo se usan los atributos y métodos.
- Como se reutiliza los métodos si tengo tres o mas objetos

- Diferencia entre métodos constructor con parámetros y sin parámetros
- Diferencia entre métodos

INVESTIGACION

Una clase en Java se crea utilizando la palabra clave `class` seguida del nombre de la clase y contiene atributos y métodos que definen las características y comportamientos de los objetos.

Un constructor es un método especial que tiene el mismo nombre de la clase y sirve para inicializar los atributos; puede ser un constructor sin parámetros, que asigna valores predeterminados, o un constructor con parámetros, que recibe valores externos para asignarlos a los atributos. Los métodos se crean definiendo un tipo de retorno, un nombre, paréntesis y el cuerpo del método dentro de llaves, y permiten realizar acciones específicas dentro de la clase.

Un objeto se instancia usando la palabra clave `new` seguida del nombre de la clase y se suele ubicar dentro del método `main`. Los atributos y métodos se utilizan a través del objeto creado, accediendo mediante el operador punto. Cuando se tienen varios objetos, los métodos se reutilizan en cada uno, manteniendo la misma estructura de comportamiento pero trabajando con los valores propios de cada objeto.

La diferencia entre un constructor con parámetros y uno sin parámetros es que el primero permite personalizar cada objeto en su creación, mientras que el segundo asigna valores fijos. La diferencia entre métodos normales y constructores es que los métodos realizan acciones específicas y se llaman manualmente, mientras que los constructores se ejecutan automáticamente al crear un objeto para inicializar sus atributos.

Prácticas Progresivas para Entender el Flujo de POO en Java

La siguiente serie de **cuatro prácticas** te ayudará a comprender paso a paso cómo funciona la Programación Orientada a Objetos (POO) en Java. Iremos desde lo más básico (crear una clase y un objeto) hasta utilizar múltiples objetos y distinguir claramente entre **constructores** y **métodos** normales. Cada práctica incluye un objetivo, una actividad propuesta y un pequeño ejemplo o guía de referencia. ¡Manos a la obra!

Práctica 1: Mi Primera Clase y Objeto

Objetivo: Entender qué es una **clase** y qué es un **objeto**, y aprender a definir una clase simple con atributos y métodos. Se busca que comprendas que primero se crea una *clase* (el plano o molde), y luego podemos crear *objetos* (instancias concretas) a partir de esa clase.

Actividad: Crea una clase sencilla y utilízala para instanciar (crear) un objeto en Java. Sigue estos pasos básicos:

1. **Definir una clase:** Por ejemplo, una clase `Persona`. Dentro de la clase, declara uno o más **atributos** (datos que describen a la persona, por ejemplo nombre) y un **método** (una acción o comportamiento, por ejemplo un método que haga que la persona se presente).
2. **Instanciar la clase en main:** En otra clase con método `main` (por ejemplo `ProgramaPersona`), crea un objeto de tipo `Persona` usando la palabra clave `new`.
3. **Usar el objeto:** A través del objeto instanciado, asigna valor a sus atributos y llama a sus

métodos. Observa cómo el método puede usar los atributos del objeto.

Identifica si se creó un método constructor, y que sucede si no se lo crea??, Si no definimos ningún constructor en una clase, Java proporciona un *constructor por defecto* (sin parámetros) automáticamente, SI O NO??

```
Main.java x Persona.java x
1 public class Main {
2     public static void main(String[] args) {
3
4         //Instancio el objeto
5         Persona persona = new Persona( nombreApellido: "Ivan Andres Panchi Chávez", edad: 21, numeroCedula: "175476899-0");
6         Persona persona1 = new Persona( nombreApellido: "Camila Perez", edad: 20, numeroCedula: "125678941-9");
7
8         //Llamo al método
9         persona.mostrarDatos();
10        persona1.mostrarDatos();
11    }
12 }
13
```

```
Main.java x Persona.java x
1 //Creo la clase Persona con sus atributos
2 public class Persona {
3     String nombreApellido;
4     int edad;
5     String numeroCedula;
6
7     // Creo el constructor
8     Persona(String nombreApellido, int edad, String numeroCedula) {
9         this.nombreApellido = nombreApellido;
10        this.edad = edad;
11        this.numeroCedula = numeroCedula;
12    }
13
14    //Metodo
15    public void mostrarDatos(){
16        System.out.println("Hola, me llamo " + nombreApellido + " tengo "+edad+" años.\n"+"Mi número de cédula es: "+numeroCedula);
17    }
18 }
19
```

```
C:\Users\LAPTO\.jdk\openjdk-20.0.2\bin\java.exe "-javaa
Hola, me llamo Ivan Andres Panchi Chávez tengo 21 años.
Mi número de cédula es: 175476899-0
Hola, me llamo Camila Perez tengo 20 años.
Mi número de cédula es: 125678941-9

Process finished with exit code 0
```

¿Si no definimos ningún constructor en una clase, Java proporciona un constructor por defecto automáticamente?

Sí.

Práctica 2: Introducción a los Constructores (sin y con Parámetros)

Objetivo: Aprender qué es un **método constructor** en Java, para qué sirve y cómo definir dos tipos de constructores: uno *sin parámetros* (constructor por defecto) y otro *con parámetros*. Comprenderás cómo los constructores inicializan los atributos al crear un objeto y cómo se diferencian de los métodos normales en su definición y uso.



```
Main.java x Coche.java x
1 public class Main {
2     public static void main(String[] args) {
3
4         //Instancias de la clase Coche sin y con parámetros
5         Coche coche1 = new Coche();
6
7         Coche coche2 = new Coche( marca: "Toyota", modelo: "Fortuner", año: 2016);
8
9         //Llamo al método
10        coche1.mostrarinfo();
11        coche2.mostrarinfo();
12    }
13 }
14 |
```

```
1  public class Coche {
2      String marca;
3      String modelo;
4      int año;
5
6      Coche() { //Constructor sin parámetros
7          this.marca = "Hyundai";
8          this.modelo = "Sonata";
9          this.año = 2017;
10     }
11
12     Coche(String marca, String modelo, int año) { //Constructor con parámetros
13         this.marca = marca;
14         this.modelo = modelo;
15         this.año = año;
16     }
17
18
19     //Método
20
21     public void mostrarinfo() {
22         System.out.println("Resumen:\n" + marca + " " + modelo + " " + año + "\n");
23     }
24
25 }
26
```

Actividad: Define una nueva clase que incluya **constructores**. Por ejemplo, crea una clase Coche con algunos atributos, y proporciona dos constructores:

- Un constructor **sin parámetros** que asigne valores predeterminados a los atributos.
- Un constructor **con parámetros** que reciba valores para inicializar los atributos.

Además, incluye en la clase al menos un método normal (por ejemplo, mostrarInfo) para mostrar el estado del objeto. Luego, en el método main, instancia **dos objetos**: uno usando el constructor sin parámetros y otro usando el constructor con parámetros, y utiliza el método para verificar que los atributos fueron inicializados correctamente en cada caso.

Práctica 3: Usando Múltiples Objetos y Reutilización de Métodos (Ejemplo Mejorado)

Objetivo:

Comprobar cómo una misma **clase** puede servir de plantilla para **múltiples objetos** y cómo los **métodos definidos** en la clase se reutilizan en cada objeto.

Aquí afianzarás la idea de que cada objeto tiene **sus propios valores** en los atributos, pero **comparte la estructura y comportamientos** de la clase.

Actividad:

- Crea una clase que modele algo sencillo.
- Instancia **tres o más objetos** de esa clase y usa sus métodos.
- Por ejemplo, modela una **Pelota**:
 - Atributos: color y tamaño.
 - Método: mostrarDetalles(), que imprime el color y tamaño de la pelota.
- En el método main, crea al menos tres pelotas con diferentes características.
- Llama al método de cada pelota para **mostrar sus detalles**.

Observa:

El **mismo método** se ejecuta en cada objeto, pero usando **los valores propios** de cada instancia.



```
1 public class Main {
2     public static void main(String[] args) {
3
4         //Cre las instancias
5         Pelota pelota = new Pelota( color: "rojo", tamaño: "grande");
6         Pelota pelota1 = new Pelota( color: "azul", tamaño: "mediano");
7         Pelota pelota2 = new Pelota( color: "amarillo", tamaño: "pequeño");
8
9
10        //Llamo a los métodos
11        pelota.mostrarDetalles();
12        pelota1.mostrarDetalles();
13        pelota2.mostrarDetalles();
14    }
15 }
```

```
Main.java x Pelota.java x
1 public class Pelota {
2     //Atributos
3     String color;
4     String tamaño;
5
6     //Creo al constructor
7     Pelota(String color, String tamaño) {
8         this.color = color;
9         this.tamaño = tamaño;
10    }
11    //Creo el método
12    public void mostrarDetalles(){
13        System.out.println("Pelota "+color+" y "+tamaño);
14    }
15
16 }
17
```

```
Main x
C:\Users\LAPT0\.jdk\openjdk-20.0.2\bin\java.exe "-javaage
Pelota rojo y grande
Pelota azul y mediano
Pelota amarillo y pequeño

Process finished with exit code 0
```


Práctica 4: Consolidando Conceptos - Constructores vs. Métodos Normales

Objetivo: Diferenciar claramente entre **métodos constructores** (con o sin parámetros) y **métodos normales** de una clase, consolidando todo el flujo POO aprendido. Tras esta práctica, sabrás identificar qué hace cada tipo de método, cuándo se ejecuta cada uno y cómo utilizarlos correctamente al crear y manipular objetos.

```
Main.java x Libro.java x
1 public class Main {
2     public static void main(String[] args) {
3
4         // Instanciando un objeto usando el constructor sin parámetros
5         Libro libro1 = new Libro();
6
7         // Instanciando un objeto usando el constructor con parámetros
8         Libro libro2 = new Libro(titulo: "Cien Años de Soledad", autor: "Gabriel García Márquez", paginas: 471);
9
10        // Llamo al método
11        libro1.imprimirInfo();
12        libro2.imprimirInfo();
13    }
14 }
15
```

```
Main.java x Libro.java x
4 usages
1 class Libro {
2     // Atributos
3     String titulo;
4     String autor;
5     int paginas;
6     // Constructor sin parámetros
7     Libro() {
8         this.titulo = "Desconocido";
9         this.autor = "Desconocido";
10        this.paginas = 0;
11    }
12    // Constructor con parámetros
13    Libro(String titulo, String autor, int paginas) {
14        this.titulo = titulo;
15        this.autor = autor;
16        this.paginas = paginas;
17    }
18    // Creo el método
19    public void imprimirInfo() {
20        System.out.println("Título: " + titulo);
21        System.out.println("Autor: " + autor);
22        System.out.println("Páginas: " + paginas);
23        System.out.println("-----");
24    }
25 }
26
```

Actividad: Diseña una clase que integre todo lo aprendido. Por ejemplo, crea una clase Libro con varios atributos (titulo, autor, paginas). Provee **dos constructores**: uno **sin parámetros** (que inicialice los atributos con valores por defecto, p.ej. "Desconocido") y otro **con parámetros** (que reciba valores para cada atributo). Agrega también al menos un **método normal** - por ejemplo, imprimirInfo() que muestre los datos del libro formateados. En el método main, instancia objetos de tipo Libro usando ambos constructores y prueba el método normal. Esto te permitirá comparar: ¿cómo se usan y cuándo se ejecutan el constructor vs. el método normal?

El constructor se ejecuta automáticamente en el momento en que se crea un objeto de una clase utilizando la palabra clave new, y su función principal es inicializar los atributos del objeto, ya sea con valores por defecto o con valores proporcionados. En cambio, un método normal se usa manualmente llamándolo desde el objeto creado, mediante el operador punto (.), y se puede ejecutar tantas veces como sea necesario para realizar una acción o mostrar información. Mientras que el constructor sólo se ejecuta una vez al crear el objeto, los métodos normales se ejecutan cada vez que son llamados.

```

Main x
C:\Users\LAPTO\jdk\openjdk-20.0.2\bin\java.exe
Título: Desconocido
Autor: Desconocido
Páginas: 0
-----
Título: Cien Años de Soledad
Autor: Gabriel García Márquez
Páginas: 471
-----|
Process finished with exit code 0
```

Subir la práctica de manera individual al git, tanto los programas como la investigación, enviar el enlace TAREA2_Apellido_Nombre