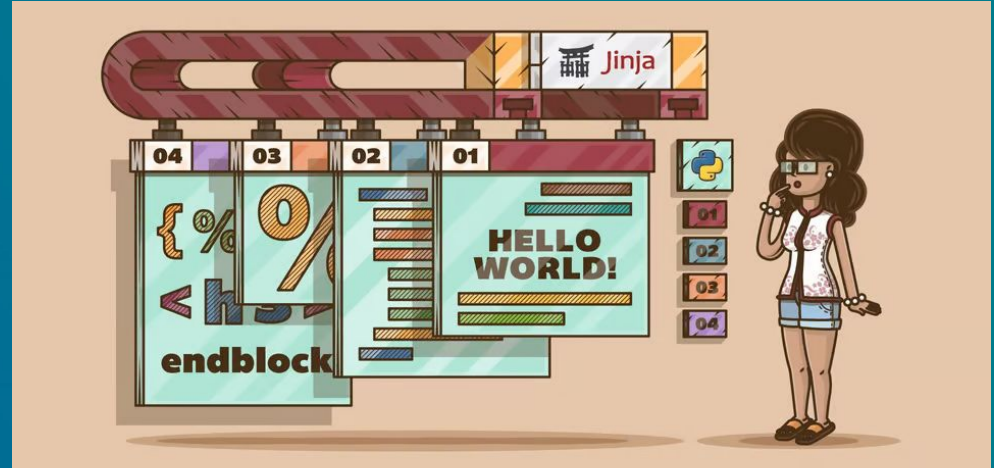


Paradigmas de programación



¿Qué son los paradigmas de programación?

Los paradigmas son las diferentes formas de usar la programación para resolver un problema.

Podría decirse que cada paradigma es un enfoque distinto o una forma de pensar distinta.

Los paradigmas no están ligados a un lenguaje de programación específico, son conceptos a aplicar.

Ejemplos:

- Programación estructurada
- Programación funcional
- Programación orientada a objetos (POO).
- Programación reactiva.

Programación estructurada

Programación secuencial con la que todos aprendemos a programar. Usa ciclos y condicionales.



Programación funcional

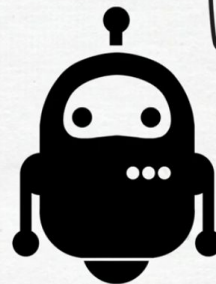
Divide el programa en tareas pequeñas que son ejecutadas por funciones.

FUNCIONES

Una función es un conjunto de líneas de código que realizan una tarea específica y pueden retornar algo.

```
function saludar ( ) {  
    =====  
}
```

```
function saludar (fernando NOMBRE) {  
    =====  
}
```



```
saludar ( );  
saludar (fernando);
```

Un valor que se le envía a la función

PARÁMETRO

Programación orientada a objetos

Agrupar las funciones en entidades llamadas objetos, los cuales tienen características y comportamientos específicos.

¿QUÉ ES LA PROGRAMACIÓN ORIENTADA A OBJETOS?

Es un paradigma de programación que organiza las funciones en entidades llamadas objetos.

Los objetos se crean a partir de una plantilla llamada **clase**. Cada objeto es una instancia de su clase.



CLASE

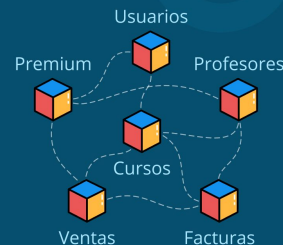
INSTANCIACIÓN

OBJETO



Los objetos tienen **datos (atributos)** y **funcionalidades (métodos)**.

En una aplicación los objetos están separados **pero se comunican entre ellos**.



ATRIBUTOS

Nombres
Apellidos
Correo
Contraseña
Premium

MÉTODOS

Editar perfil
Iniciar sesión
Cerrar sesión
Cambiar contraseña
Pasar a premium

Puedes programar con este paradigma **en la mayoría de lenguajes**.



Programación reactiva

Es un paradigma de programación orientado a flujos de datos y a la propagación de cambios. En este enfoque, los programas están diseñados para reaccionar automáticamente a eventos o cambios en los datos de forma asíncrona.

Según el manifiesto reactivo, los sistemas reactivos deben ser:

- **Responsivos:** aseguran la calidad del servicio cumpliendo tiempos de respuesta establecidos.
- **Resilientes:** Mantener responsivos incluso en situaciones de error.
- **Elásticos:** Mantener responsivos incluso en alta carga de trabajo.
- **Orientados a mensajes:** Minimizar el acoplamiento

Librerías para aplicar la programación reactiva

RxJava



Project Reactor



RxJS



Ventajas de Programación Reactiva



Escalabilidad:
Implementación poco acoplada, y con facilidad para aislar fallos.



Rendimiento:
Procesar cargas de trabajo altas con menos hilos

¿Qué empresas usan programación reactiva?



Netflix



GitHub



Microsoft