



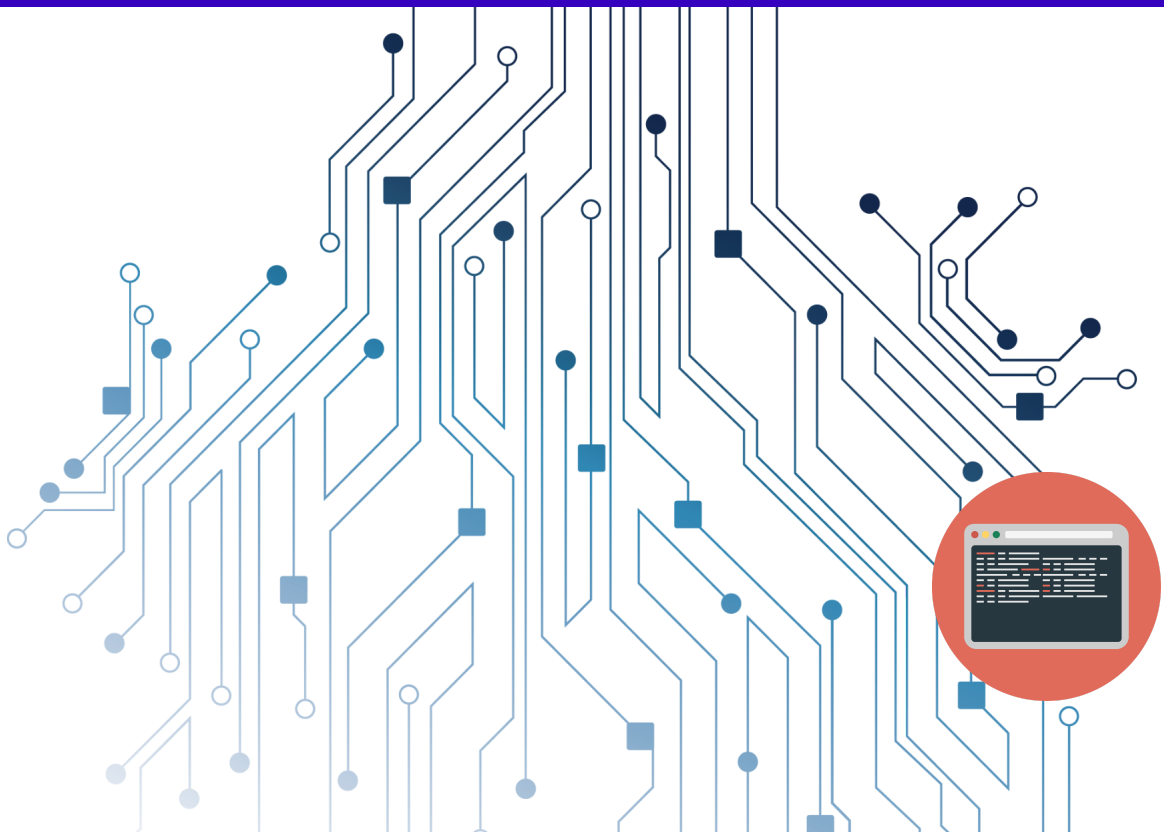
CFGS APLICACIONES MULTIPLATAFORMA - PROGRAMACIÓN



## UNIDAD 0

---

# PRESENTACIÓN



**DIEGO VALERO ARÉVALO**

BASADO EN APUNTES Y EJERCICIOS PROPORCIONADOS POR  
M<sup>ra</sup> CARMEN DÍAZ GONZÁLEZ - IES VIRGEN DE LA PALOMA

# U0 - PRESENTACIÓN

## CONTENIDO DEL CURSO

### UNIDAD 1 - INTRODUCCIÓN A LA PROGRAMACIÓN

Nuestros primeros pasos en programación empiezan conociendo los tipos de de datos con los que trabaja JAVA y su sintaxis básica. Esta unidad incluye:

- Introducción a las bases del sector de la programación.
- Introducción a las funcionalidades y herramientas de JAVA.
- Instalación y uso de JAVA.
- Explicación de la estructura de programas JAVA, los tipos de datos que maneja y su sintaxis.
- Cómo comunicarse con el usuario a través de métodos de entrada y salida.

### UNIDAD 2 - ELEMENTOS LÓGICOS

Conoceremos los fundamentos de la lógica de programación para saber cómo fluye el código en un programa y alterarlo. Esta unidad incluye:

- Estructuras de control: condicionales, bucles y salto.
- Métodos o funciones

### UNIDAD 3 - PROGRAMACIÓN ORIENTADA A OBJETOS (POO)

Estudiaremos uno de los fundamentos básicos de la programación en JAVA: los objetos. Esta unidad incluye:

- Explicación de qué es un objeto en programación.
- Creación de clases y cómo desarrollarlas.
- Uso y creación de paquetes y librerías.

### UNIDAD 4 - ESTRUCTURAS DE ALMACENAMIENTO

Para poder trabajar con muchos datos del mismo tipo aprenderemos cómo poder almacenar todos en una misma variable, iniciándonos en las colecciones. Esta unidad incluye:

- Comprensión, funcionamiento, creación, desarrollo y manipulación de arrays unidimensionales y multidimensionales.
- La clase String para trabajar con cadenas de caracteres.

### UNIDAD 5 - COLECCIONES

Veremos clases avanzadas para ver más a fondo el almacenamiento de datos. Esta unidad incluye:

- Visión general de clases dedicadas al almacenamiento de datos y comprender la jerarquía de estas.
- Uso de ArrayList, HashMap y clases Set.
- Un diagrama para ayudarnos a elegir entre todas las clases de almacenamiento de datos.

### UNIDAD 6 - EXCEPCIONES

¿Qué pasa cuando un programa lanza un error? Aquí aprenderemos a poder trabajar con esos errores a nuestro favor. Esta unidad incluye:

- Comprensión de la jerarquía y tipos de la clase Throwable.
- Manejo y lanzamiento de excepciones.
- Creación de excepciones personalizadas.

### UNIDAD 7 - USO AVANZADO DE CLASES

Retomando la programación orientada a objetos veremos más usos y métodos de trabajo con clases. Esta unidad incluye:

- Explicación del concepto de composición y sus tipos.
- Herencia entre clases.
- Polimorfismo de clases.
- Clases abstractas.
- Interfaces.

## UNIDAD 8 - GESTIÓN DE FICHEROS

JAVA nos permite crear programas para manejar ficheros del sistema de archivos de nuestro sistema operativo. Esta unidad incluye:

- Explicación y uso de la clase File y la librería java.io.
- Explicación y uso de rutas absolutas y relativas.
- Acceso y manipulación de ficheros.
- Lectura y escritura de ficheros de texto y binarios a través de JAVA.

## UNIDAD 9 - TIPOS DE DATOS GENÉRICOS

Cuando trabajamos con muchos datos que son de diferentes tipos podemos hacer uso de tipos genéricos. Esta unidad incluye:

- Explicación de qué son los tipos genéricos.
- Uso de tipos genéricos en clases, interfaces y métodos.
- Aprender a limitar las clases de los tipos genéricos.
- Uso de comodines según el tipo.
- Restricciones al uso de tipos genéricos.

## ANEXO- IMPLEMENTACIÓN Y USO DE MÉTODOS Y CLASES RECOMENDADAS

Este anexo contiene métodos y clases que están recomendadas al trabajar con diferentes datos y otras clases en JAVA. Puedes echarle un vistazo en cualquier momento para resolver algunos casos de las baterías de ejercicios. El mismo contiene temario adicional a las unidades, por lo que, a menos que se puntualice en alguna unidad, no se recomienda consultarlo hasta haber completado la **unidad 7, Uso avanzado de clases**, ya que contiene terminología y sintaxis que probablemente aún no se hayan estudiado.

A lo largo de las unidades encontrarás diferentes **cajas y apartados** que te darán cierta información relacionada con el temario:



Esta es una **caja de consejos**. En ella podrás ver **información adicional** o **ayuda** para situaciones concretas. Algunas pueden ir acompañadas del texto



### TEN EN CUENTA

Las cajas con este título te proporcionan información sobre **ejemplos** o **datos importantes** que deberías tener en mente al programar.



Esta es una **caja de atención**. En ella se explican datos en detalle o ciertas cosas importantes que pueden ser **riesgos** o **errores a evitar** en el código. Es importante que no las ignores.

### EJEMPLO

MiPrograma.java

```
1 public class MiPrograma{}
```

Encontrarás estas cajas en muchos puntos de cada apartado de cada unidad. En ellas se ponen **ejemplos** de programas verificados que te pueden **servir de ayuda** para ver cómo se implementan ciertas cosas en un código real. Pueden acompañarse de varios **prompts de código** e incluso de **terminales de consola** para que veas el resultado de una ejecución.

# ECLIPSE IDE

El entorno de desarrollo que usaremos para programar en JAVA es **Eclipse**, en el cual podemos **escribir**, **editar**, **compilar** y **ejecutar un programa**, e incluso ejecutar **debugging**.

Para empezar, puedes descargarlo desde la página oficial de Eclipse Foundation:

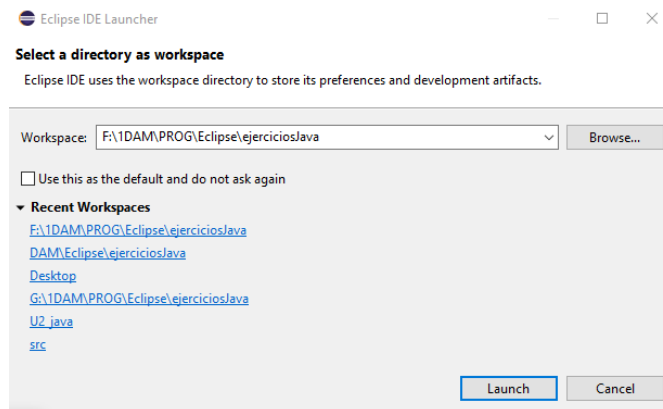
ECLIPSE FOUNDATION DOWNLOADS >

Selecciona la versión **más reciente de Eclipse IDE** y descarga el archivo acorde a tu sistema operativo.

Lo bueno de Eclipse es que es un **programa autoejecutable**, lo que significa que **no tenemos que instalarlo**, sólo ejecutarlo para su uso.

## >> INICIAR ECLIPSE IDE: CONFIGURACIÓN DEL WORKSPACE

Al iniciar Eclipse, este nos pedirá una ruta en la que guardar el **workspace**, que es donde se guardarán todos los archivos relacionados a la configuración del editor.



## >> CONFIGURAR UN PROYECTO JAVA

Para comenzar a programar en JAVA, debemos crear un proyecto. Para ello, una vez abierto Eclipse:



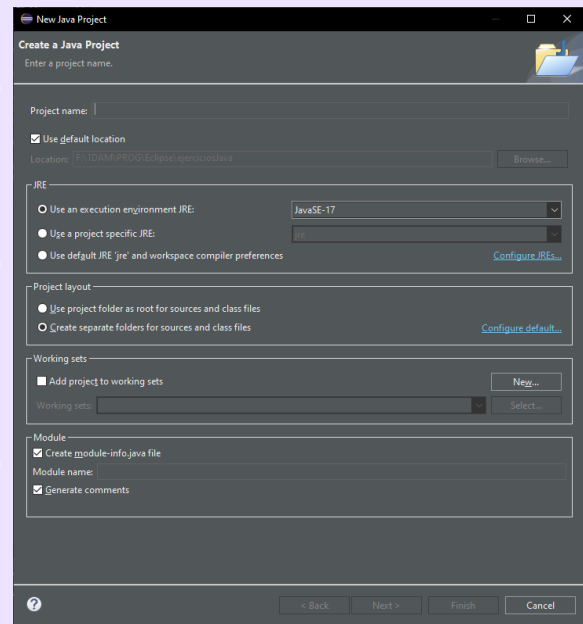
A continuación aparecerá una ventana en la que configuraremos nuestro proyecto. En esta ventana debes añadir los siguientes datos:

Nombre del proyecto.

Marca la casilla de “use default location” para añadir tu proyecto a donde has guardado el **workspace**, o si no, selecciona **otra ruta**.

La **versión del JRE** que usará el proyecto (Puedes saber más sobre el JRE y el JDK en la **unidad 1, apartado 1.2.2 - Herramientas de JAVA e instalación**).

Asegúrate de **desmarcar** la casilla “**Create module-info.java file**”. Este es un recurso que no se suele utilizar ya que suele generar problemas o errores.



Una vez creado el proyecto, podremos ver en la pestaña Package Explorer una carpeta con el nombre src. Esta es la carpeta que contendrá todos los paquetes y clases de nuestro proyecto.



Puedes cambiar el **tema de color** de Eclipse en:

Window

>

Preferences

>

General

>

Appearance

>

Theme:

## >> EMPEZAR A CREAR UN PROYECTO JAVA

Para añadir elementos al proyecto, simplemente haz **click derecho del ratón** en el **Package Explorer**, selecciona **New** y podrás ver el **listado de lo que puedes crear**, incluyendo **proyectos, paquetes, clases y archivos**.

**¿Listo para la aventura? ¡Vamos a programar!**