

**Temario**

# **Unidad 2**

- a)El Entorno Java para la programación**
- b)Sentencias de control**
- c)Creando aplicaciones de consola en Java**

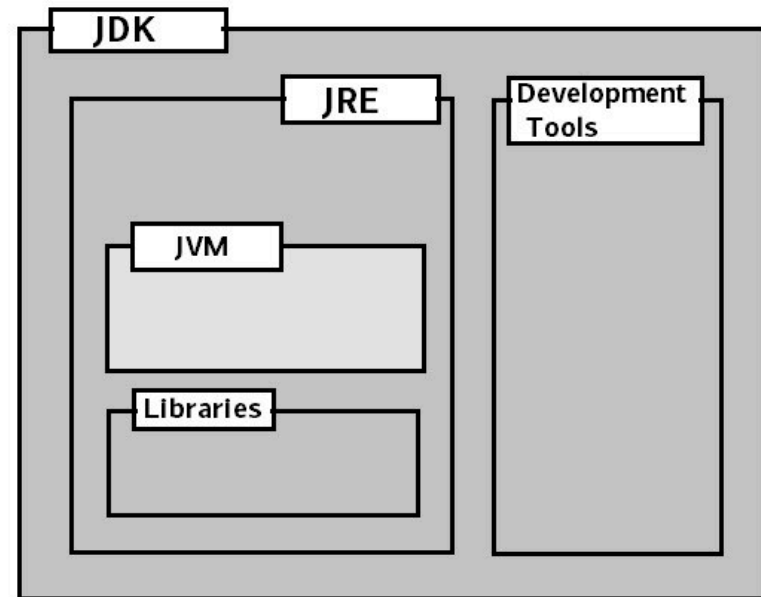
## Unidad 2

# El Entorno Java para la programación

Los entornos de desarrollo Java son aplicaciones que permiten al programador implementar las abstracciones del mundo real en un aplicación concreta mediante la introducción de secuencias de código con sus estructuras de programación.

## Unidad 2

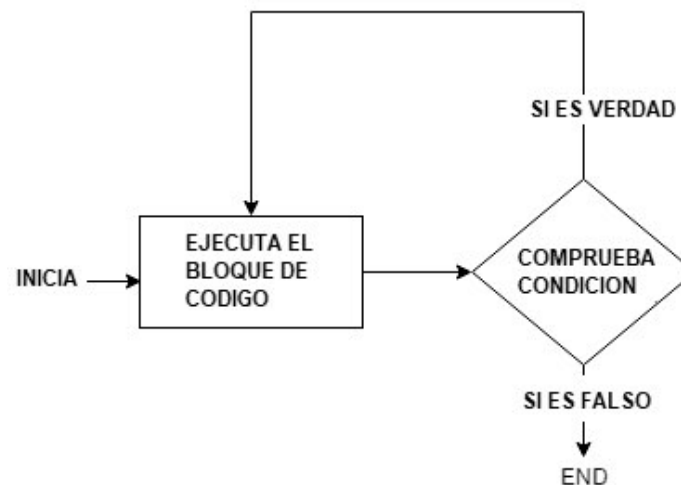
Los **entornos de desarrollo Java** (IDE) son herramientas que funcionan como un sistema integral que nos permite programar de forma mucho más sencilla que si solo empleásemos el JDK (*Java Development Kit*) de Java y un procesador de textos



## Unidad 2

# Sentencias de control

Un ciclo en Java o bucle en Java permite repetir una o varias instrucciones cuantas veces lo necesitemos o sea necesario



## Unidad 2

si quisiéramos escribir los números del uno al cien no tendría sentido escribir cien líneas de código mostrando un número en cada una de estas, para eso y para varias cosas más, es útil un ciclo

## Unidad 2

Existen diferentes tipos de ciclos o bucles en Java, cada uno tiene una utilidad para casos específicos y depende de nuestra habilidad y conocimientos poder determinar en qué momento o situación es bueno usar alguno de ellos. Tenemos entonces a nuestra disposición los siguientes tipos de ciclos en Java:

### FOR

un ciclo for es una estructura iterativa para ejecutar un mismo segmento de código una cantidad de veces deseada; conociendo previamente un valor de inicio, un tamaño de paso y un valor final para el ciclo realizando finalmente el incremento en su contador para la siguiente iteración.

```
for (int i = valor inicial; i <= valor final; i = i ++)  
{  
    ....  
    Instrucciones....  
    ....  
}
```

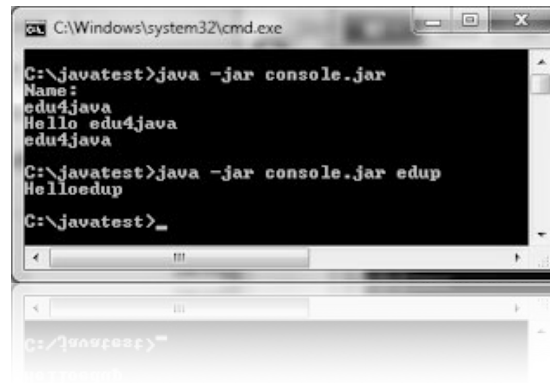
## Unidad 2

<p><b>WHILE</b></p> <p>El bucle while presenta ciertas similitudes y ciertas diferencias con el bucle for. La repetición en este caso se produce no un número predeterminado de veces, sino mientras se cumpla una condición. Conceptualmente el esquema más habitual es el siguiente:</p>	<pre>Int i = 0; <b>while</b> (i &lt;= valor <b>final</b>) {     i++; }</pre>
<p><b>DO-WHILE</b></p> <p>El bucle do ... while es muy similar al bucle while. La diferencia radica en cuándo se evalúa la condición de salida del ciclo. En el bucle while esta evaluación se realiza antes de entrar al ciclo, lo que significa que el bucle puede no llegar ejecutarse. En cambio, en un bucle do ... while, la evaluación se hace después de la primera ejecución del ciclo, lo que significa que el bucle obligatoriamente se ejecuta al menos en una ocasión. A modo de ejercicio, escribe este código y comprueba los resultados que se obtienen con él.</p>	<pre><b>do</b> {     contador += 1; } <b>while</b> (contador &lt; 10);</pre>

## Unidad 2

# Creando aplicaciones de consola en Java

Las aplicaciones de consola son aquellos programas en Java que se van a ejecutar en una ventana de comandos o de Shell, como vemos a continuación:



```
C:\Windows\system32\cmd.exe

C:\javatest>java -jar console.jar
Name:
edu4java
Hello edu4java
edu4java

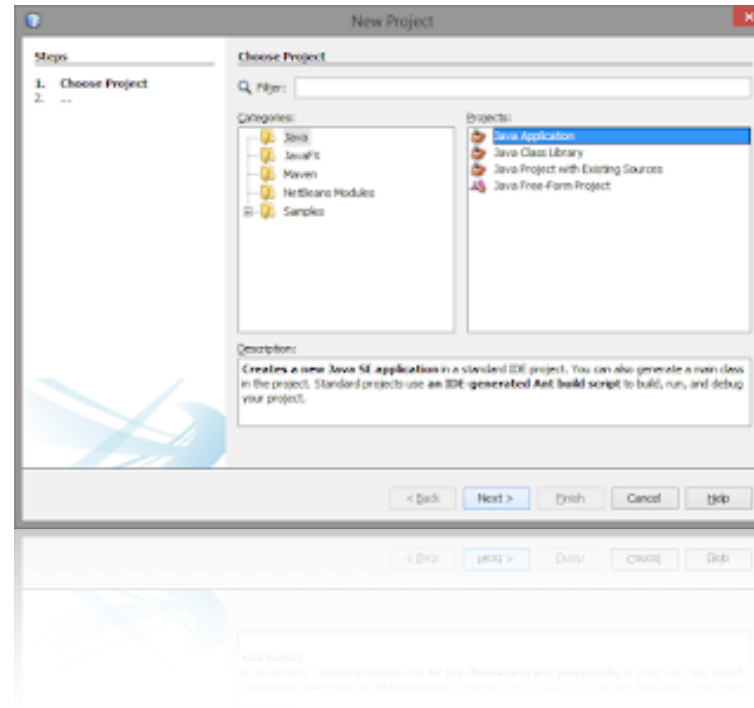
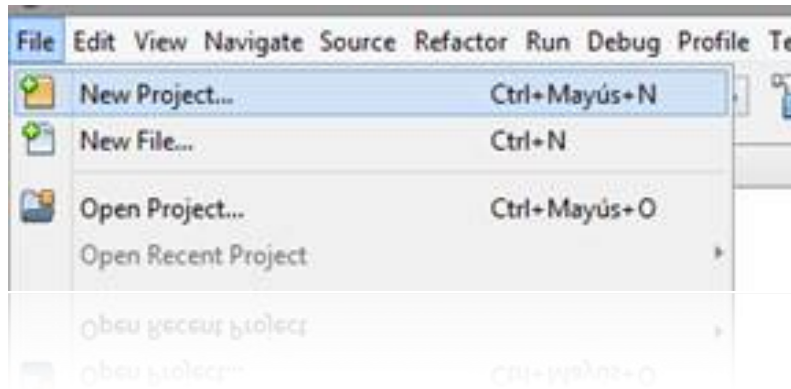
C:\javatest>java -jar console.jar edup
Helloedup

C:\javatest>
```



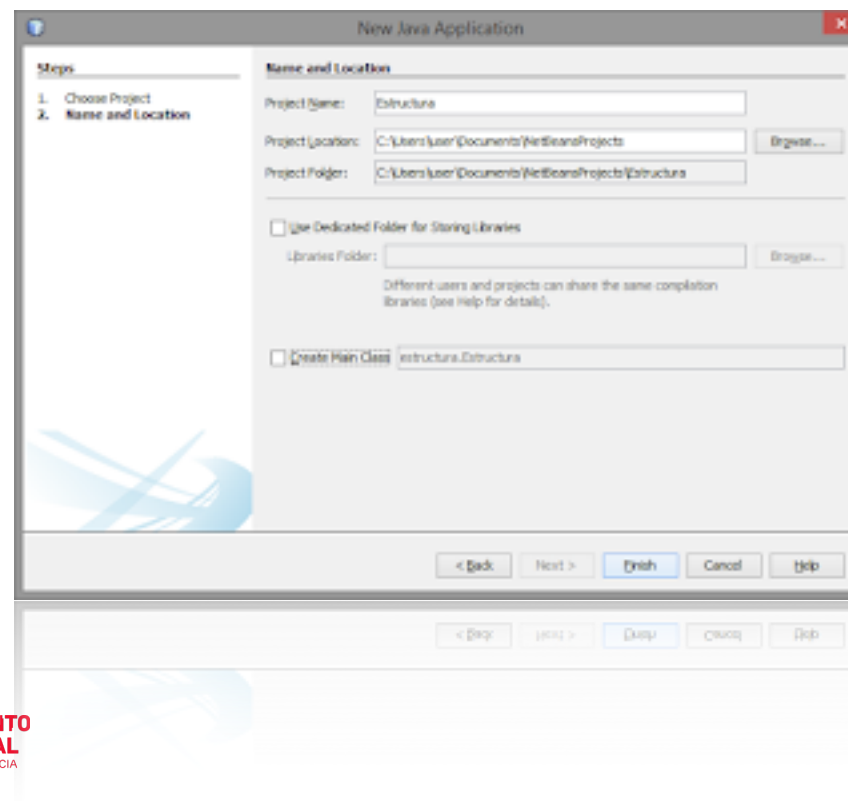
## Unidad 2

Creamos un nuevo proyecto al que vamos a llamar Estructura:



## Unidad 2

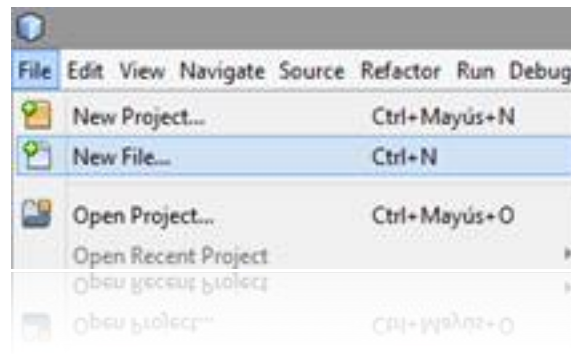
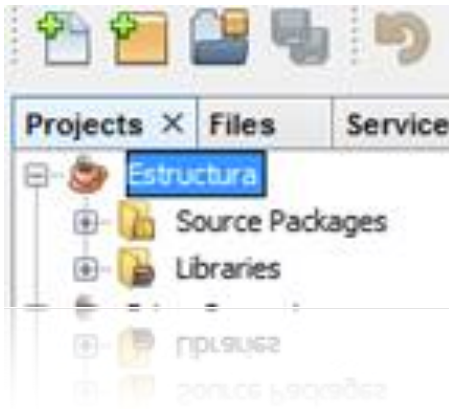
Desactivaremos Create Main Class:



## Unidad 2

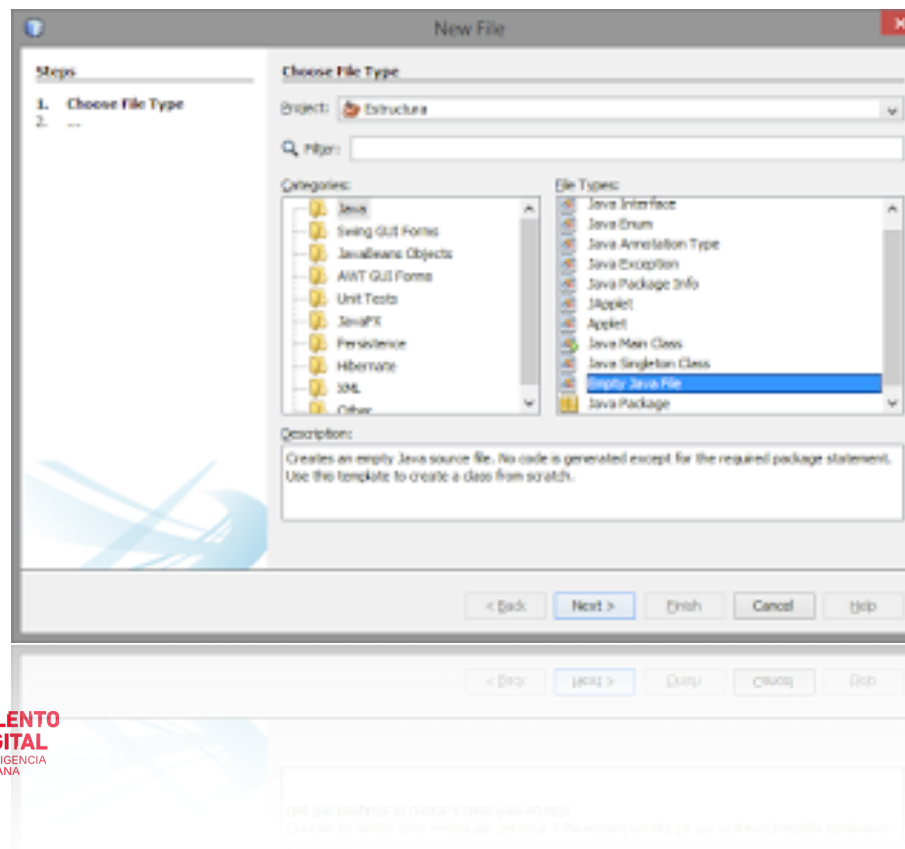
Hacemos clic en *Finish* y se nos crea un nuevo proyecto, pero sin ningún archivo:

nuevo fichero lo podemos hacer desde el menú File > New File... o haciendo clic en este icono



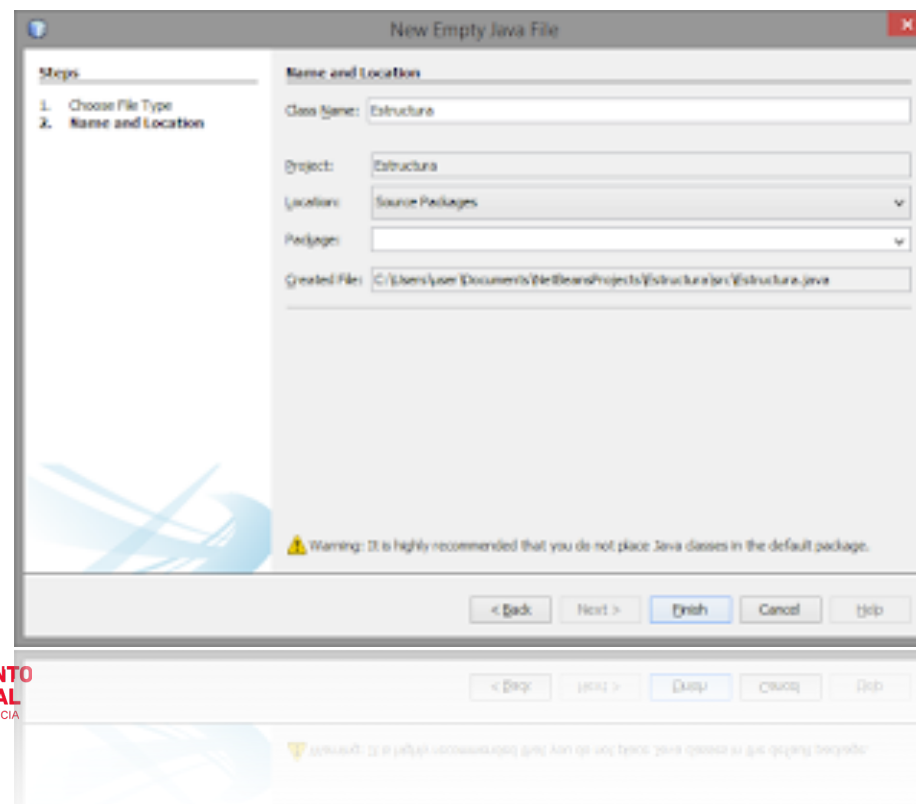
## Unidad 2

Vamos a seleccionar como tipo de archivo, un archivo vacío Empty Java File:



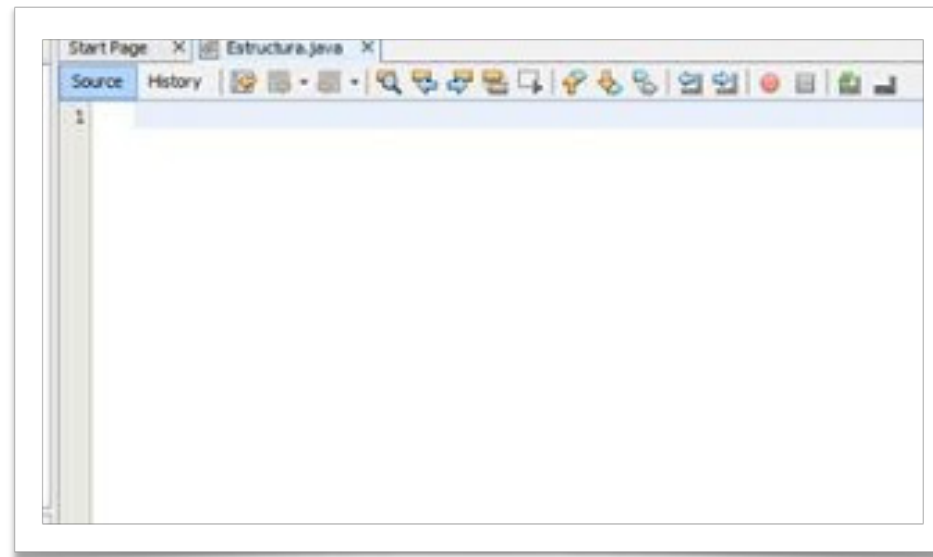
## Unidad 2

Al fichero le vamos a llamar Estructura:



## Unidad 2

Y se nos ha creado un archivo vacío:

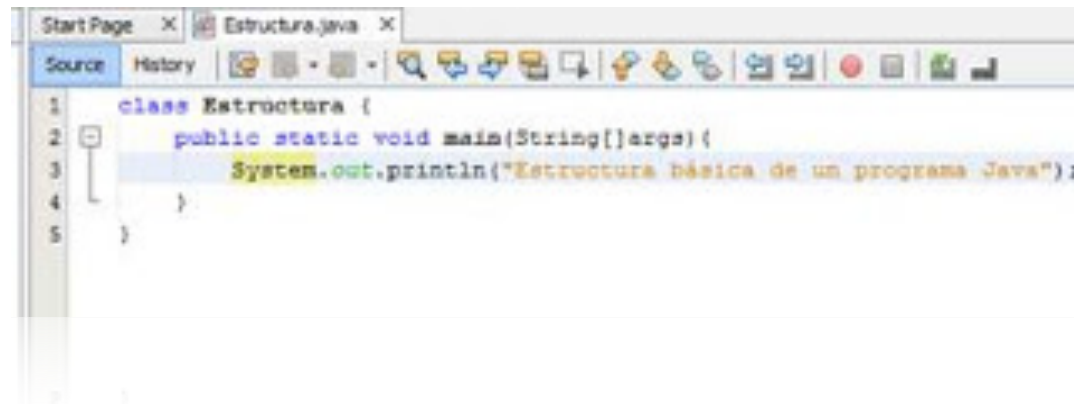


## Unidad 2



## Unidad 2

Vamos a empezar a escribir código en este archivo:



```
1 class Estructura {
2     public static void main(String[] args) {
3         System.out.println("Estructura básica de un programa Java");
4     }
5 }
```

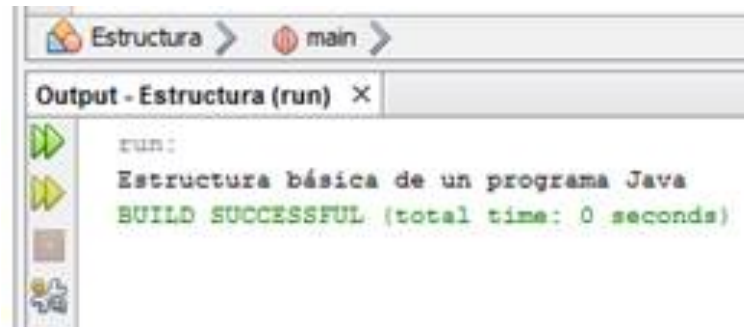


## Unidad 2

1. La declaración class, lo que hacemos es declarar una clase, el nombre de la clase debe coincidir con el nombre del archivo, en este caso Estructura.
2. La segunda línea es la declaración main, estamos indicando que la parte principal del programa empieza aquí, main es el punto de entrada de la mayoría de los programas, hay excepciones como por ejemplo los applets que son programas que se ejecutan como parte de una página Web, y los servlets que son programas que se ejecutan en un servidor.
3. Las llaves también forman parte de la estructura del programa, gracias a ellas creamos los bloques. En este ejemplo vemos que la llave de apertura de la línea uno tiene que ver con la llave de cierre de la línea cinco, forma un grupo, dentro de este grupo hay una llave de apertura y otra de cierre que forma un bloque dentro de ese grupo.
4. La función System.out.println nos va a imprimir en pantalla lo que le pasemos dentro de los paréntesis entre comillas.

## Unidad 2

Si este programa lo ejecutamos nos aparece en pantalla lo que hay entre paréntesis en la función `System.out.println`:



## Material complementario de la unidad

### Link a video relacionado

Java y el entorno de desarrollo: <https://youtu.be/6mSBnIEzSms>

### Link a lectura complementaria

<https://www.campusmvp.es/recursos/post/Los-mejores-entornos-de-desarrollo-para-Java.aspx>

<https://jarroba.com/instalar-bien-eclipse-un-ide-de-muchos/>