

Laboratorio 2

Objetivos

- Aprender a programar en Java utilizando el entorno Eclipse
- Conocer los tipos de datos y las instrucciones básicas de Java

Aspectos a considerar

- Documentar la funcionalidad de cada método una vez escrita su cabecera y antes de iniciar su implementación. Para ello, colócate en la línea sobre la cabecera del método y, una vez escrita, teclea `/**`; a continuación completa la información del método.
- Si se ha seleccionado la opción Project → Build Automatically, a medida que se vaya escribiendo el código del método, Eclipse irá informando de los errores de compilación que encuentre.

Entregables

Al final del laboratorio hay que entregar el proyecto Java exportado desde Eclipse.

- El nombre del fichero que hay que subir a eGela debe ser: **apellido_nombre_L02.zip**
- La entrega será **INDIVIDUAL** y se realizará a través de eGela.

Tareas a realizar

1. En eGela encontrarás el manual *"Primeros pasos en eclipse"*, bajo la etiqueta *software*, que explica cómo crear tu primer proyecto. El programa principal de este proyecto escribe "Hola mundo" en la pantalla.
2. Crea un nuevo proyecto **numbers** y una nueva clase, asegurándote de tener seleccionado JavaSE-1.8 en la parte de *Use an execution environment JRE*.
3. Crear un paquete llamado `packNumbers` en tu nuevo proyecto.
4. Crea en tu paquete una clase llamada `WorkingWithNumbers`, que NO contenga el método `main` y en la que tendrás que incluir los siguientes métodos:
 - a. `isEven`: método que dado un número indica si es par o no.
 - b. `evenValues`: método que muestra en pantalla los números pares comprendidos entre el 1 y el número que recibe como parámetro desde la entrada.
5. Hacer otra clase llamada **DEMO** dentro del paquete `packNumbers` que tenga un programa principal (método `main`) que llame a los métodos de la clase `WorkingWithNumbers`. Recuerda que para ello tendrás que crear una instancia de clase `WorkingWithNumbers`. En esta primera versión, utiliza valores constantes introducidos como parámetros para llamar a los métodos.
6. Modifica el programa principal (método `main`) para leer desde el teclado el valor que se pasará como parámetro al invocar al método `evenValues`:
 - Abrir una vía de comunicación con el teclado:

```
Scanner consola = new Scanner (System.in);
```

NOTA: *System.in* es la entrada estándar de teclado.

- Para poder utilizar Scanner es necesario escribir `import java.util.Scanner`, antes del comienzo de la definición de la clase.
 - Para leer un número entero desde la línea de comunicación abierta:

```
n = consola.nextInt ();
```

 teniendo que ser la variable `n` de tipo `int`.
 - Cuando terminen todas las lecturas hay que cerrar la comunicación con el teclado:

```
consola.close ();
```
7. Guarda el trabajo realizado. Para ello, sigue los pasos indicados en el manual de Eclipse de eGela (bajo etiqueta **Software**), sección “exportación del proyecto”. El resultado de la exportación debe ser un fichero .zip que puedes copiar para llevarte a casa el trabajo realizado en el laboratorio.
 8. Cuando quieras abrir un proyecto previamente Exportado, tendrás que Importarlo (por ejemplo, si quieres trabajar en casa con el proyecto empezado y exportado en el laboratorio). En la sección “Importación del proyecto” del manual de Eclipse (está en eGela), se describen los pasos a seguir para importar un proyecto previamente exportado.

Tareas Adicionales:

9. Escribir un método en la clase `WorkingWithNumbers` que recoja dos números enteros e indique si el resultado de su multiplicación es positivo, negativo o nulo. Después, probarlo desde el programa principal (`main`) de la clase `DEMO`.
10. Escribir un método en la clase `WorkingWithNumbers` que recoja un número entero y devuelva la suma de sus divisores propios.
11. Calcular en el programa principal de la clase `DEMO` (utilizando el método implementado en 10):
 - Si un número es perfecto o no. Un número es perfecto cuando la suma de sus divisores propios es igual al número. Por ejemplo: 6, 28 y 496 son números perfectos.
 - Decir si dos números son amigos o no. Diremos que los números `N` y `M` son amigos, cuando la suma de los divisores propios de `N` es `M`, y la suma de los divisores propios de `M` es `N`. Por ejemplo: 220 y 284 son números amigos; 1184 y 1210 también son números amigos.