

AW
Desarrollo de Aplicaciones Web
2º Curso

DWES
Desarrollo Web Entorno Servidor

UD 4
Iniciación a Java

IES BALMIS
Dpto Informática
Curso 2025-2026
Versión 1 (10/2025)

1. EJERCICIOS

UD4Ejer01

Crea un programa que muestre en pantalla el texto "Dime tu nombre: " y después, en la misma línea, lee el nombre del usuario, que puede contener espacios.

Luego escribirá 5 veces "Hola, " seguido por el nombre del usuario, en 5 líneas distintas.

```
run:  
Dime tu nombre: Juan  
Hola, Juan  
Hola, Juan  
Hola, Juan  
Hola, Juan  
Hola, Juan  
BUILD SUCCESSFUL (total time: 5 seconds)
```

UD4Ejer02

Pide al usuario un número entero y muéstra su factorial.

El factorial de un número se define en principio como el producto de todos los números enteros positivos desde 1 (es decir, los números naturales) hasta n.

Ejemplo: Factorial de 4 = 1 * 2 * 3 * 4 = 24

```
run:  
Introduce un numero entero: 4  
El factorial de 4 es 24  
BUILD SUCCESSFUL (total time: 9 seconds)
```

UD4Ejer03

Pide al usuario un número del 1 al 12 y muestra el nombre del correspondiente mes, usando una estructura de control de tipo "switch".

```
run:  
Introduce un numero entero: 7  
El mes es Julio  
BUILD SUCCESSFUL (total time: 7 seconds)
```

```
run:  
Introduce un numero entero: 11  
El mes es Noviembre  
BUILD SUCCESSFUL (total time: 2 seconds)
```

UD4Ejer04

Pide al usuario un número del 1 al 12 y muestra el nombre del correspondiente mes, usando un **array de cadenas** de texto en el que estarán almacenados dichos nombres.

```
run:  
Introduce un numero entero: 02  
El mes es Febrero  
BUILD SUCCESSFUL (total time: 2 seconds)
```

UD4Ejer05

Pide al usuario un número entero y di si es (o no) un palíndromo primo, con la ayuda de dos funciones booleanas "esPrimo" y "esPalindromo".

Algunos ejemplos son:

**2, 3, 5, 7, 11, 101, 131,
151, 181, 191, 313, 353,
373, 383, 727, 757, 787,
797, 919, 929, 10301,
10501, 10601, 11311,
11411, 12421, 12721...**

El código de las funciones se ofrecen a continuación, pero el alumno debe comprender su estructura y probar el "**Debugger**" para comprobar su correcto funcionamiento.

```
// FUNCIÓN - esPrimo
public static boolean esPrimo(int numero) {
    int contador = 2;
    boolean primo=true;

    while ((primo) && (contador!=numero)){
        if ( (numero % contador) == 0) {
            primo = false;
        }
        contador++;
    }
    return primo;
}

// FUNCIÓN - esPalindromo
public static boolean esPalindromo(int numero) {
    String sPalabra=String.valueOf(numero);
    int inc = 0;
    int des = sPalabra.length()-1;
    boolean palindromo = true;

    while ((inc<des) && (palindromo)){
        if (sPalabra.charAt(inc)==sPalabra.charAt(des)){
            inc++;
            des--;
        } else {
            palindromo = false;
        }
    }
    return palindromo;
}
```

```
run:  
Introduce un numero entero: 181  
El numero 181 es primo  
El numero 181 es palindromo  
El numero 181 es primo y palindromo  
BUILD SUCCESSFUL (total time: 3 seconds)
```



```
run:  
Introduce un numero entero: 1526  
El numero 1526 NO es primo  
El numero 1526 NO es palindromo  
BUILD SUCCESSFUL (total time: 3 seconds)
```



```
run:  
Introduce un numero entero: 13  
El numero 13 es primo  
El numero 13 NO es palindromo  
BUILD SUCCESSFUL (total time: 2 seconds)
```

UD4Ejer06

Crea un programa que tendrá tres fases:

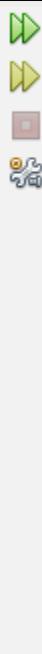
- 1) En la primera fase, el usuario introducirá números enteros positivos, que se guardarán en un ArrayList (esta fase terminará cuando introduzca un número negativo, que no se guardará).
- 2) En la segunda fase, el usuario introducirá números enteros positivos de uno en uno, a lo que se le contestará si el correspondiente número está o no en la lista de datos que se habían introducido inicialmente. Cuando introduzca nuevamente un número negativo, terminará la fase de búsqueda.
- 3) Por último, se mostrarán ordenados los datos que contiene el ArrayList.



```
run:  
-----  
Introduce un numero entero (negativo para terminar): 7  
Introduce otro numero entero (negativo para terminar): 5  
Introduce otro numero entero (negativo para terminar): 12  
Introduce otro numero entero (negativo para terminar): 4  
Introduce otro numero entero (negativo para terminar): -1  
-----  
Introduce numero a buscar (negativo para terminar): 5  
5 aparece  
Introduce otro numero a buscar (negativo para terminar): 12  
12 aparece  
Introduce otro numero a buscar (negativo para terminar): 8  
8 no existe  
Introduce otro numero a buscar (negativo para terminar): -1  
-----  
Valores introducidos:  
4  
5  
7  
12  
BUILD SUCCESSFUL (total time: 23 seconds)
```

UD4Ejer07

Crea una variante del ejercicio **UD4Ejer06** en la que los datos no serán numéricos, sino cadenas de texto, de modo que cada fase terminará cuando se introduzca la palabra "**fin**".

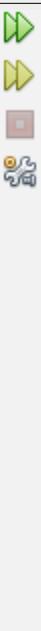


```
run:
-----
Introduce una palabra ('fin' para terminar): manzana
Introduce otra palabra ('fin' para terminar): pera
Introduce otra palabra ('fin' para terminar): naranja
Introduce otra palabra ('fin' para terminar): fin
-----
Introduce una palabra a buscar ('fin' para terminar): pera
pera aparece
Introduce otra palabra a buscar ('fin' para terminar): uva
uva no existe
Introduce otra palabra a buscar ('fin' para terminar): fin
-----
Valores introducidos:
manzana
naranja
pera
BUILD SUCCESSFUL (total time: 23 seconds)
```

UD4Ejer08

Crea una variante del mejorada ejercicio **UD4Ejer07**, en la que la lista de datos será una clase **ListaDeDatos**, con métodos públicos

- "void incluir(texto)",
- "boolean contiene(texto)" y
- "void mostrarDatosOrdenados()".



```
run:
-----
Introduce una palabra ('fin' para terminar): manzana
Introduce otra palabra ('fin' para terminar): uva
Introduce otra palabra ('fin' para terminar): pera
Introduce otra palabra ('fin' para terminar): fin
-----
Introduce una palabra a buscar ('fin' para terminar): naranja
naranja no existe
Introduce otra palabra a buscar ('fin' para terminar): uva
uva aparece
Introduce otra palabra a buscar ('fin' para terminar): fin
-----
Valores introducidos:
manzana
pera
uva
BUILD SUCCESSFUL (total time: 33 seconds)
```