

Práctica 0. Introducción a JAVA

DAW (CFGS Desarrollo de aplicaciones Web)

Módulo 3: Programación orientado a objetos

Curso: 2019-2020

Khalid Chalili Houlliche



Índice

Ejercicio 1: Imprimir en pantalla.....	3
Debug y resultado.....	3
Ejercicio 2: Bucle for, bucle if y condicionales. Pasar array por el método main.....	4
Debug y resultado.....	4
Ejercicio 3: Dar valor a las variables de tipo int. Bucle for y condicional if.....	5
Debug y resultado.....	5
Ejercicio 4: .length(), .charAt().....	6
Debug y resultado.....	6
Ejercicio 5: new StringBuilder(), .toString(), .reverse().....	7
Debug y resultado.....	7
Ejercicio 6: Utilización de arrays y ordenamiento de valores.....	8
Debug y resultado.....	8
Ejercicio 7: Utilización de arrays y ordenamiento de valores con el algoritmo de la burbuja.....	9
Debug y resultado.....	10
Ejercicio 8: Cálculo del factorial.....	11
Debug y resultado.....	11

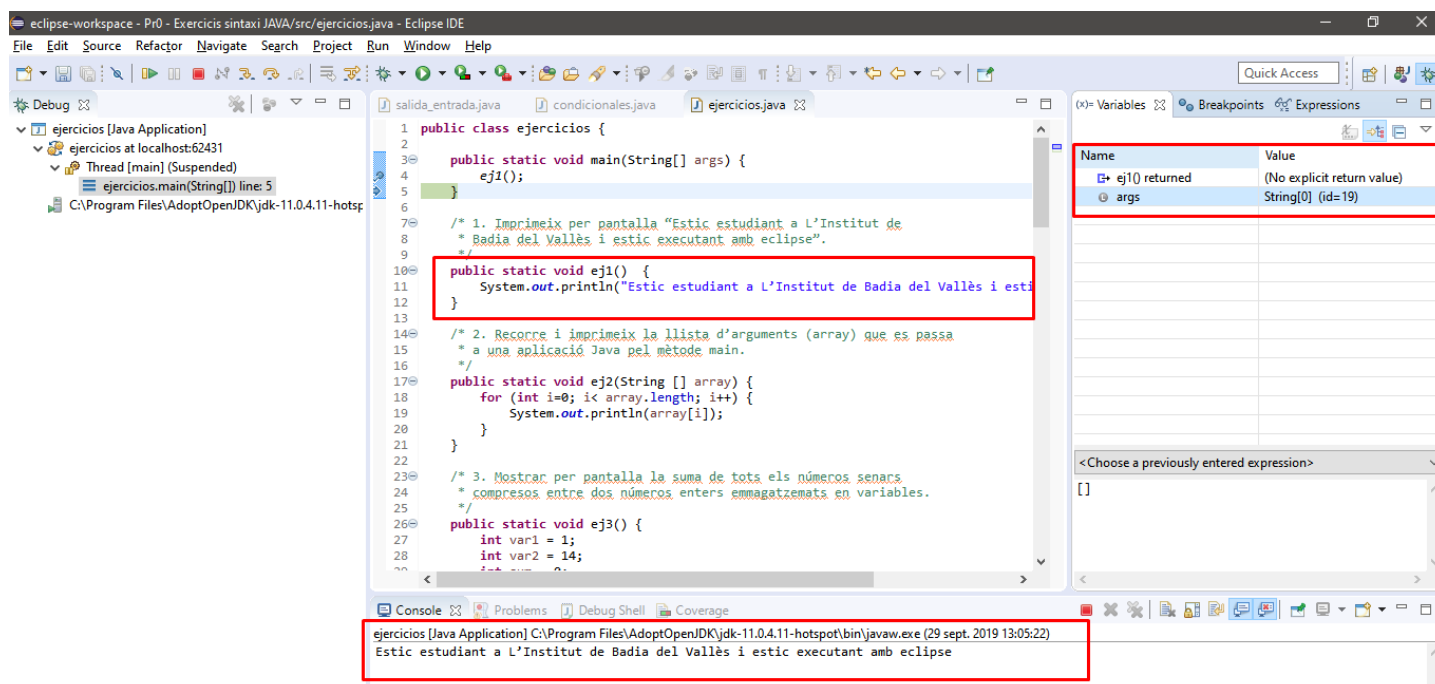
Ejercicio 1: Imprimir en pantalla

1. Imprimir por pantalla "Estoy estudiando en el Instituto de Badia del Vallès y estoy ejecutando con eclipse".

```
public class ejercicios {  
  
    public static void main(String[] args) {  
        ej1();  
    }  
  
    /* 1. Imprimeix per pantalla "Estic estudiant a L'Institut de  
     * Badia del Vallès i estic executant amb eclipse".  
     */  
    public static void ej1() {  
        System.out.println("Estic estudiant a L'Institut de Badia del Vallès i estic  
executant amb eclipse");  
    }  
}
```

Debug y resultado

El array "args" está vacío y no retorna ningún valor, solamente muestra por pantalla un String por consola.



Ejercicio 2: Bucle for, bucle if y condicionales. Pasar array por el método main

2. Recorre e imprime la lista de argumentos (array) que se pasa a una aplicación Java por el método main.

```
public class ejercicios {

    public static void main(String[] args) {
        ej2(args);
    }

    /* 2. Recorre i imprimeix la llista d'arguments (array) que es passa
     * a una aplicació Java pel mètode main.
     */
    // array son los parametros que recibe, en nuestro caso un array.
    // NO IMPORTA QUÉ NOMBRE PONER, LO QUE IMPORTA ES LA POSICIÓN DE LOS DATOS QUE RECIBE.
    public static void ej2(String [] array) {
        /* Bucle for donde i vale 0 inicialmente para después recorrer
         * la cantidad de valores que hay en el array. Dentro del bucle imprime el
         * valor de la del array dependiendo de la posición que recorre el bucle. valor 0
         * es la posición 1 del array.
         */
        for (int i=0; i< array.length; i++) {
            System.out.println(array[i]);
        }
    }
}
```

Debug y resultado

Cuando entra en el bucle el valor i vale 0. Después de ejecutar el print de dentro del bucle la "i" valdrá 1 y así hasta el número 2 (0, 1, 2) en el bucle entra 3 veces.

The screenshot shows the Eclipse IDE interface. The main editor displays the source code of the 'ejercicios' class. The 'Variables' panel on the right shows the state of the program during a debug session. The 'array' variable is a String array with three elements: '5', '8', and '7'. The 'i' variable is highlighted in red and shows the value 0, indicating the first iteration of the loop. The 'Console' panel at the bottom shows the output of the program: 'Estic estudiant a L'Institut de Badia del Vallès i esti'.

Name	Value
no method return value	
array	String[3] (id=19)
> array[0]	"5" (id=22)
> array[1]	"8" (id=26)
> array[2]	"7" (id=27)
i	0

Console: ejercicios [Java Application] C:\Program Files\AdoptOpenJDK\jdk-11.0.4.11-hotspot\bin\javaw.exe (29 sept. 2019 13:18:51)

Ejercicio 3: Dar valor a las variables de tipo int. Bucle for y condicional if.

3. Mostrar por pantalla la suma de todos los números impares comprendidos entre dos números enteros almacenados en variables.

```
public class ejercicios {

    public static void main(String[] args) {
        ej3();
    }

    /* 3. Mostrar per pantalla la suma de tots els números senars
     * compresos entre dos números enters emmagatzemats en variables.
     */
    public static void ej3() {
        // Guardamos los valores de tipo Integer en variables de tipo Integer:
        int var1 = 1;
        int var2 = 14;
        int sum = 0;

        // Bucle for parecida a la práctica anterior
        for (int i = var1 ; i <= var2; i++) {
            // si el resultado es igual a 1 en el cálculo matematico, quiere decir que es
            // impar.
            if ((i%2) == 1) {
                //También podemos sumar de esta manera: sum += i;
                sum = sum + i;
            }
        }
        System.out.println("suma de números impares " + sum);
    }
}
```

Debug y resultado

En mi caso los números deben ser uno menor y el otro mayor para que el bucle funcione, si no habría que hacer otro bucle para saber qué numero es el menor y el mayor.

The screenshot shows the Eclipse IDE with the following components:

- Editor:** Displaying the code for `ejercicios.java`. Lines 34-36 are highlighted: `int var1 = 1;`, `int var2 = 14;`, and `int sum = 0;`.
- Variables View:** Located on the right, it shows the state of variables during execution:

Name	Value
println() returned	(No explicit return value)
var1	1
var2	14
sum	49
- Console:** At the bottom, it shows the output of the program: `ejercicios [Java Application] C:\Program Files\AdoptOpenJDK\jdk-11.0.4.11-hotspot\bin\javaw.exe (29 sept. 2019 13:31:19) suma de números impares 49`.

Ejercicio 4: .length(), .charAt()

4. A partir de una cadena de la clase String, recorrerla caracter a caracter y devuelve o muestra por pantalla la cadena gira.

```
public class ejercicios {

    public static void main(String[] args) {
        ej4();
    }

    /* 4. A partir d'una cadena de la classe String, recorre-la
     * caracter a caracter i retorna o mostra per pantalla la cadena girada.
     */
    public static void ej4() {
        String cadena = "Hola";
        String invertida = "";

        // length() lee cuantas letras tiene el string
        System.out.println(cadena.length());

        /* Restamos -1 ya que la posicion H es el 0. y el 4 no existe.
         * Empezamos el bucle de mayor a menor para que pueda recorrer la palabra
         * de derecha a izquierda.
         */
        for (int longitud = cadena.length() - 1; longitud >= 0; longitud--) {
            // con -charAt(incremento del bucle) concatena los Strings.
            invertida += cadena.charAt(longitud);
        }

        System.out.println("Cadena original: " + cadena);
        System.out.println("Cadena invertida: " + invertida);
    }
}
```

Debug y resultado

Como vemos, entra con el 3, que es la posición última de la palabra.

The screenshot shows the Eclipse IDE interface. The main editor displays the source code of 'ejercicios.java'. The 'Variables' panel on the right shows the state of variables: 'cadena' is 'Hola' (id=22), 'invertida' is 'a' (id=35), and 'longitud' is 3. The 'Console' panel at the bottom shows the output: 'ejercicios [Java Application] C:\Program Files\AdoptOpenJDK\jdk-11.0.4.11-hotspot\bin\javaw.exe (29 sept. 2019 13:39:03) 4'.

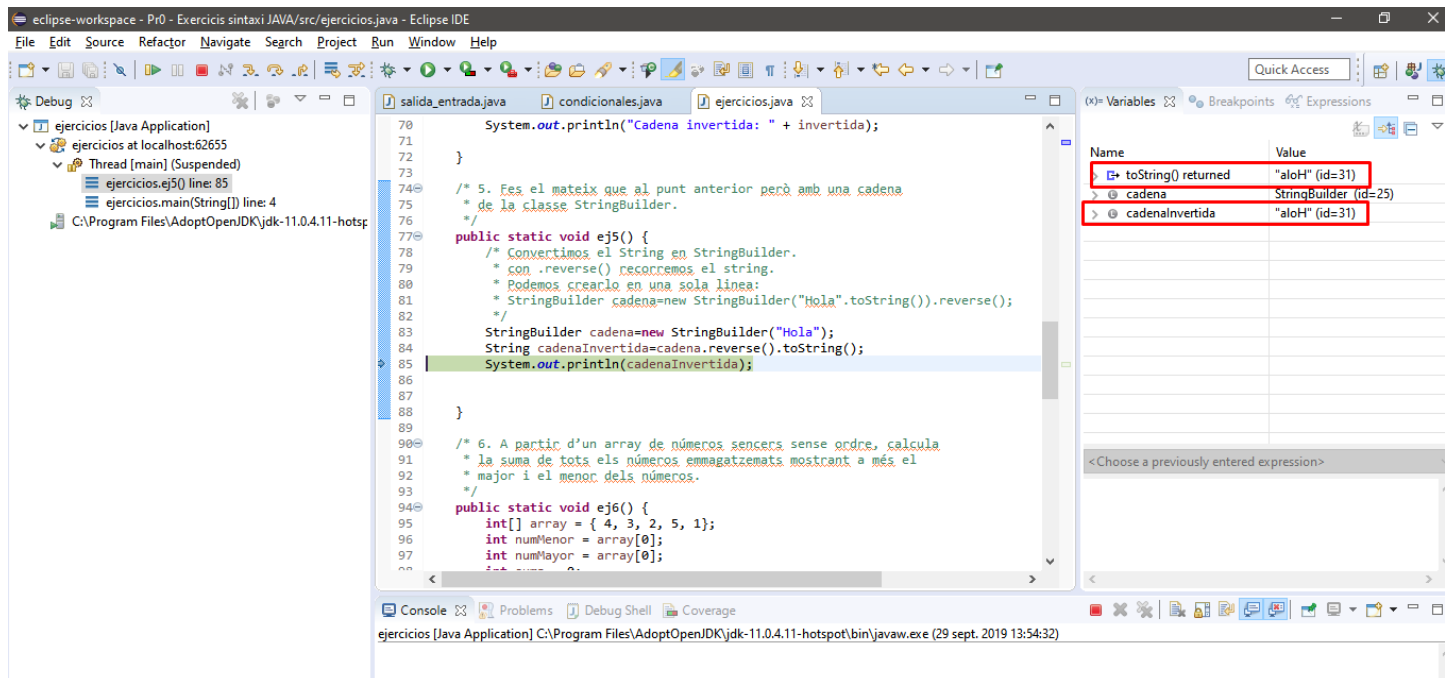
Ejercicio 5: new StringBuilder(), .toString(), .reverse()

5. Haz lo mismo que en el punto anterior pero con una cadena de la clase StringBuilder.

```
public class ejercicios {  
  
    public static void main(String[] args) {  
        ej5();  
    }  
  
    /* 5. Fes el mateix que al punt anterior però amb una cadena  
     * de la classe StringBuilder.  
     */  
    public static void ej5() {  
        /* Convertimos el String en StringBuilder.  
         * con .reverse() recorremos el string.  
         * Podemos crearlo en una sola línea:  
         * StringBuilder cadena=new StringBuilder("Hola".toString()).reverse();  
         */  
        StringBuilder cadena=new StringBuilder("Hola");  
        String cadenaInvertida=cadena.reverse().toString();  
        System.out.println(cadenaInvertida);  
  
    }  
}
```

Debug y resultado

ToString() retorna el resultado.



Ejercicio 6: Utilización de arrays y ordenamiento de valores

6. A partir de un array de números enteros sin orden, calcula la suma de todos los números almacenados mostrando además el mayor y el menor de los números.

```
public class ejercicios {

    public static void main(String[] args) {
        ej6();
    }
    /* 6. A partir d'un array de números sencers sense ordre, calcula
    * la suma de tots els números emmagatzemats mostrant a més el
    * major i el menor dels números.
    */
    public static void ej6() {
        int[] array = { 4, 3, 2, 5, 1};
        int numMenor = array[0];
        int numMayor = array[0];
        int suma = 0;

        for(int posicion = array.length -1; posicion >=0; posicion-- ) {
            int numero = array[posicion];
            suma += numero;
            if (numero > numMayor) {
                numMayor = numero;
            } if (numero < numMenor) {
                numMenor = numero;
            }
        }

        System.out.println("numero Mayor " + numMayor);
        System.out.println("numero Menor " + numMenor);
        System.out.println("suma " + suma);
    }
}
```

Debug y resultado

Empezamos con guardar una posición del array en dos variables, no importa cuales son. Recorre el bucle for de la cantidad de valores del array hasta el "0". Se aplica la lógica de los if.

The screenshot shows the Eclipse IDE interface. The main editor displays the source code for 'ejercicios.java'. A red box highlights the loop and conditional logic for finding the maximum and minimum values. The 'Variables' panel on the right shows the state of variables: 'array' (No explicit return value), 'numMenor' (1), 'numMayor' (5), and 'suma' (15). The 'Console' panel at the bottom shows the output: 'numero Mayor 5', 'numero Menor 1', and 'suma 15'.

Name	Value
println() returned	(No explicit return value)
array	(id=22)
numMenor	1
numMayor	5
suma	15

Console Output:

```
numero Mayor 5
numero Menor 1
suma 15
```


Ejercicio 7: Utilización de arrays y ordenamiento de valores con el algoritmo de la burbuja.

7. A partir de un array formado por 5 números enteros, ordena de menor a mayor los números dentro del array y posteriormente muestra el array por pantalla. Usar el algoritmo de la burbuja.

```
public class ejercicios {

    public static void main(String[] args) {
        ej7();
    }

    /* 7. A partir d'un array format per 5 números sencers, ordena de menor
    * a major els números dins de l'array i posteriorment mostra l'array
    * per pantalla. Utilitza l'algoritme de la bombolla.
    */
    public static void ej7() {
        int array[] = {5,3,2,7,6};

        for (int i = 0; i <= array.length -1; i++) {
            for (int j = 0; j <= array.length -2; j++) {
                /*se compara la posición j con la siguiente, si es mayor entonces
                * se guarda una variable temporal de la posición siguiente (j+1)
                * para luego escribirla en la posición inicial.
                */
                if (array[j] > array[j + 1]) {
                    int temp = array[j + 1];
                    array[j +1] = array[j];
                    array[j] = temp;
                }
            }
        }

        // Resultado
        for (int x = 0; x <= array.length -1; x++) {
            System.out.println(array[x]);
        }
    }
}
```

Debug y resultado

The screenshot shows the Eclipse IDE with the following components:

- Editor:** Displays the source code of `ejercicios.java`. A breakpoint is set at line 131.
- Variables Panel:** Shows the state of variables at the current execution point.

Name	Value
no method return value	
array	(id=22)
i	0
j	0
temp	3

Si el valor es menor: Pasa esto en los últimos bucles del for

The three screenshots show the state of variables at different points in the execution:

- First Screenshot:** `i` is 2, `j` is 3.
- Second Screenshot:** `i` is 3, `j` is 3.
- Third Screenshot:** Shows the return value of `println()` as "(No explicit return value)", `array` as (id=22), and `x` as 4.

Recorre el for para mostrar el array:

The screenshot shows the Eclipse IDE with the following components:

- Editor:** Displays the source code of `ejercicios.java`.
- Console:** Shows the output of the program:

```
ejercicios [Java Application] C:\Program Files\AdoptOpenJDK\jdk-11.0.4.11-hotspot\bin\javaw.exe (29 sept. 2019 14:03:56)
2
3
5
6
7
```

Ejercicio 8: Cálculo del factorial.

8. Muestra por pantalla el resultado de calcular el factorial de un número pasado como argumento de la aplicación.

```
public class ejercicios {

    public static void main(String[] args) {
        ej8(args);
    }
    /*
     * 8. Mostra per pantalla el resultat de calcular el factorial d'un número
     * passat com a argument de l'aplicació.
     */
    public static void ej8(String[] args8) {
        //pasamos la variable de String a Integer
        int numero = Integer.parseInt(args8[0]);
        //El valor factorial es 1 para que se pueda multiplicar
        int factorial = 1;
        System.out.println("integer: " + numero);
        /* Un bucle while donde compara si el numero es diferente a 0
         * por que hay un contador que va restando menos uno cada vez
         * acaba cuando está dentro del bucle, restamos numero,
         * que es el argumento que pasamos.
         */
        while (numero != 0) {
            factorial *= numero;
            numero--;
        }
        System.out.println("factorial: " + factorial);
    }
}
```

Debug y resultado

Como vemos, el numero que pasamos por argumento es un String, por eso lo pasamos a Integer. La variable numero irá variando a lo largo del bucle, y que hay un decremento de 1 cada vez que pasa por el bucle, así cuando esté calculado sale del bucle y muestra la suma de factoriales. Cada vez que pasa, multiplicamos el numero factorial por el numero del contador.

