

2. UD2-PR.01: Mi primer programa.

La práctica por realizar para esta unidad consiste en resolver los siguiente ejercicios:

2.1	Ejercicio 1: Identificadores	1
2.2	Ejercicio 2: Utilizando variables	2
2.3	Ejercicio 3: Empezando a programar	3
2.4	Ejercicio 4: Localizando errores	6
2.5	Ejercicio 5: Uso de parámetros de entrada	8
2.6	Ejercicio 6: Enumerados	8
2.7	Ejercicio 7: Ecuaciones	9
2.8	Ejercicio 8: Operaciones matemáticas	10
2.9	Ejercicio 9: Manipulando parámetros de entrada	11
2.10	Ejercicio 10: Identificadores	12

Detalles de la tarea de esta unidad.

En esta unidad hemos tenido la oportunidad de crear nuestro primer programa en Java. Hemos realizado pequeños ejemplos sobre cada apartado tratado, y ahora se trata de ponerlos en práctica mediante la siguiente relación de ejercicios.

2.1 Ejercicio 1: Identificadores

Dados los siguientes identificadores, indica si son válidos o no. Justifica todas las respuestas.

- a) `main` **Correcto**, aunque no se reconoce como palabra reservada, no es recomendable usarla
- b) `mi edad` **Incorrecto**, contiene un espacio en medio
- c) `num_alumnos` **Correcto**, está formada por letras y el carácter de subrayado
- d) `____suma` **Correcto**, está formada por letras y el carácter de subrayado
- e) `$euros` **Correcto**, está formada por letras y el símbolo \$ que se considera también una letra
- f) `error?` **Incorrecto**, el carácter de interrogación no se considera una letra

- g) **9digitos** Incorrecto, empieza por un número
- h) **códigoCliente** Correcto, pues todos los caracteres son letras Unicode, aunque no es recomendable utilizar signos de puntuación a la hora de nombrar variables.
- i) **jose~** Incorrecto, pues el símbolo ~ no se considera una letra
- j) **año** Correcto, pues la letra ñ forma parte del código Unicode
- k) **PI** Correcto, pues está formado por letras. Además, se trata de una constante pues está escrito en mayúsculas.
- l) **int** Incorrecto, porque int es una palabra reservada y no se puede usar para nombrar variables.

2.2 Ejercicio 2: Utilizando variables

Dado el siguiente programa, modifícalo para incluir la declaración de las variables solicitadas a continuación.

El tipo de dato elegido debe ser el de menos bits posibles que puedan representar el valor. Justifica tu elección.

```
public class ejerciciovariables {  
    public static void main(String[] args) {  
    }  
}
```

- Si un empleado está casado o no.

Boolean. Sólo son posibles dos respuestas: Sí o No (Verdadero o Falso)

- Valor máximo no modificable: 999999.

Int. Es el tipo de menos bits necesario para representar esa cantidad.

- Día de la semana

Byte. Permite representar números enteros de -128 a 127, más que suficiente para guardar los valores del 1 al 7.

- Día del año.

Short. Tenemos que utilizar un tipo que permita representar números del 1 al 366. El tipo byte no nos vale, por lo tanto el tipo más pequeño después del byte es short.

- Sexo: con dos valores posibles 'V' o 'M'

Char. Con una sola letra podemos guardar los dos valores posibles.

También podemos representarlo como tipo enumerado, en ese caso la declaración de la variable iría fuera del main:

```
public enum sexo {V,M};
```

- Milisegundos transcurridos desde el 01/01/1970 hasta nuestros días.

Long. Los milisegundos transcurridos hasta, por ejemplo, el 01/01/2015 alcanzará la cantidad de 1,42007E+12, la cual puede representarse en una variable de tipo long.

- Almacenar el total de una factura

Float. Las cantidades en euros llevan decimales, por tanto, debemos utilizar un tipo de dato en coma flotante, y el más pequeño es float. No necesitamos el tipo double porque asumimos la posible pérdida de precisión en los decimales.

- Población mundial del planeta tierra.

El tipo de dato int no podría representar la población mundial del planeta, ya que el valor máximo que alcanza es de 2.147.483.647. Si se desea representar el valor correspondiente a la población mundial del planeta (cerca de 7 mil millones de habitantes), tendríamos que utilizar un tipo de dato long.

2.3 Ejercicio 3: Empezando a programar

Realiza las siguientes modificaciones en el programa anterior:

1. Añade comentarios, entre otros, debes incluir:
 - Nombre de clase, descripción y autor
 - Comentario para la clase main y para cada una de las llaves de cierre
2. Utiliza el operador de asignación para inicializar las variables a los valores que se indican en los mensajes.
3. Utiliza la secuencia de escape correspondiente para generar un tabulador al principio de cada línea salvo de la primera

4. Mostrar el siguiente resultado:

- Usando sólo la orden **println**:

```
----- EJERCICIO DE VARIABLES Y TIPOS DE DATOS -----  
El valor de la variable casado es true  
El valor de la variable MAXIMO es 999999  
El valor de la variable diasem es 1  
El valor de la variable diaanual es 300  
El valor de la variable miliseg es 1298332800000  
El valor de la variable totalfactura es 10350.678  
El valor de la variable poblacion es 6775235741  
El valor de la variable sexo es M
```

- Usando sólo la orden **print**:

```
----- EJERCICIO DE VARIABLES Y TIPOS DE DATOS -----  
El valor de la variable casado es true  
El valor de la variable MAXIMO es 999999  
El valor de la variable diasem es 1  
El valor de la variable diaanual es 300  
El valor de la variable miliseg es 1298332800000  
El valor de la variable totalfactura es 10350.678  
El valor de la variable poblacion es 6775235741  
El valor de la variable sexo es M
```

- Usando sólo la orden **printf**:

```
----- EJERCICIO DE VARIABLES Y TIPOS DE DATOS -----  
El valor de la variable casado es true  
El valor de la variable MAXIMO es 999999  
El valor de la variable diasem es 1  
El valor de la variable diaanual es 300  
El valor de la variable miliseg es 1298332800000  
El valor de la variable totalfactura es 10350,677734  
El valor de la variable totalfactura en notación científica es 1.035068E+04  
El valor de la variable poblacion es 6775235741  
El valor de la variable sexo es M
```

```

/*
 * Ejercicio03.java
 * Programa para practicar con los distintos tipos de datos
 * y las ordenes print, println y printf
 */

/**
 *
 * @author cris
 */
public class ejercicio03 {
    public enum sexo {V,M};

    // clase principal que inicia la aplicacion
    public static void main(String[] args) {
        boolean casado =true;
        final int MAXIMO = 999999;
        byte diasem = 1;
        short diaanual = 300;

        // segundos desde el 01/01/1970 hasta el 22/02/2001
        long miliseg = 1298332800000L;
        float totalfactura = 10350.678000f;
        long poblacion = 6775235741L;
        sexo s = sexo.M;

        // usando orden println
        System.out.println("----- EJERCICIO DE VARIABLES Y TIPOS DE DATOS -----");
        System.out.println("\tEl valor de la variable casado es " + casado);
        System.out.println("\tEl valor de la variable MAXIMO es " + MAXIMO);
        System.out.println("\tEl valor de la variable diasem es " + diasem);
        System.out.println("\tEl valor de la variable diaanual es " + diaanual);
        System.out.println("\tEl valor de la variable miliseg es " + miliseg);
        System.out.println("\tEl valor de la variable totalfactura es " +
totalfactura);
        System.out.println("\tEl valor de la variable poblacion es " + poblacion);
        System.out.println("\tEl valor de la variable sexo es " + s);

        // usando orden print
        System.out.print("\n\n--- EJERCICIO DE VARIABLES Y TIPOS DE DATOS ---\n");
        System.out.print("\tEl valor de la variable casado es " + casado + "\n");
        System.out.print("\tEl valor de la variable MAXIMO es " + MAXIMO + "\n");
        System.out.print("\tEl valor de la variable diasem es " + diasem + "\n");
        System.out.print("\tEl valor de la variable diaanual es " + diaanual +
"\n");
        System.out.print("\tEl valor de la variable miliseg es " + miliseg +
"\n");
        System.out.print("\tEl valor de la variable totalfactura es " +
totalfactura + "\n");
        System.out.print("\tEl valor de la variable poblacion es " + poblacion +
"\n");
        System.out.print("\tEl valor de la variable sexo es " + s + "\n");

        // usando orden printf
        System.out.printf("\n\n--- EJERCICIO DE VARIABLES Y TIPOS DE DATOS --\n");
        System.out.printf("\tEl valor de la variable casado es %b\n", casado);
        System.out.printf("\tEl valor de la variable MAXIMO es %d\n", MAXIMO);
        System.out.printf("\tEl valor de la variable diasem es %bd\n", diasem);
        System.out.printf("\tEl valor de la variable diaanual es %d\n", diaanual);
        System.out.printf("\tEl valor de la variable miliseg es %d\n", miliseg);
        System.out.printf("\tEl valor de la variable totalfactura es %f\n",
totalfactura);
        System.out.printf("\tEl valor de la variable totalfactura en notacion
cientifica es %E\n", totalfactura);
        System.out.printf("\tEl valor de la variable poblacion es %d\n",

```

```
poblacion);
    System.out.printf("\tEl valor de la variable sexo es %s\n", s);
} // fin main
} // fin Ejercicio03
```

2.4 Ejercicio 4: Localizando errores

Indica los errores presentes en el siguiente código:

```
1. /*
2. calculadora.java
3. Programa que muestra las operaciones básicas de una calculadora
4. */
5. public class calculadora {
6. public static main(String[] args)
7. short x = 7;
8. int y = 5;
9. float f1 = 13.5;
10.     float f2 = 8f;
11.     //Valores de los parámetros de entrada a utilizar en la
    calculadora
12.     System.out.println("Siendo el valor de x ", x, " y el valor de
    y ", y);
13.     System.out.println("El resultado de x + y es " + (x + y));
14.     System.out.println("El resultado de x - y es " + (x - y));
15.     /* división
16.     System.out.printf("\n%s%s\n","La división entera ","x / y =
    ", (x/y));
17.     System.out.println("El resto de la división entera x % y = " +
    (x % y));
18.     /*Visualizando datos de tipo float*/
19.     System.out.printf("El valor de f1 es %f y el de f2 es
    %f\n",f1,f2);
20.     System.out.println("El resultado de f1 / f2 es " + (f1 / f2))
21. }
```

```
1. /*
2. operadoresaritmeticos.java
3. Programa que muestra el uso de los operadores aritméticos
4. */
```

5. public class operadoresaritmeticos {

```
6. public static void main(String[] args) {
```

7. short x = 7;

8. int y = 5;

```
9. float f1 = 13.5f;
```

10. float f2 = 8f;

```
11. System.out.println("El valor de x es " + x + " y el valor de y es " + y);
```

12. System.out.println("El resultado de x + y es " + (x + y));

13. System.out.println("El resultado de x - y es " + (x - y));

```
14. System.out.printf("\n%s%s\n", "División entera:", "x / y = ", (x/y));
```

15. System.out.println("Resto de la división entera: x % y = " + (x % y));

16. System.out.printf("El valor de f1 es %f y el de f2 es %f\n", f1, f2);

```
17. System.out.println("El resultado de f1 / f2 es " + (f1 / f2));
```

```
18. } }
```

ERRORES:

- Instrucciones 1 y 4 falta el asterisco del comentario.
- Instrucción 6 falta el void del main y la llave de comienzo del bloque.
- Instrucción 9 falta colocar la "f" al final del valor, para indicar que es de tipo float.
- Instrucción 11 las cadenas se concatenan con +.
- Instrucción 14 no se muestra el resultado de x/y porque falta un %s en el primer argumento y la concatenación debe usarse el símbolo + en lugar de la coma.
- Instrucción 17: falta punto y coma al final.
- Instrucción 18: falta la llave que cierra la clase operadoresaritmeticos.

SOLUCIÓN:

```
/*
operadoresaritmeticos.java
Programa que muestra el uso de los operadores aritméticos
*/
public class operadoresaritmeticos {

    /*
operadoresaritmeticos.java
Programa que muestra el uso de los operadores aritméticos
*/

    public static void main(String[] args) {
        short x = 7;
        int y = 5;
        float f1 = 13.5f;
        float f2 = 8f;

        System.out.println("El valor de x es " + x + " y el valor de y es " + y);
        System.out.println("El resultado de x + y es " + (x + y));
        System.out.println("El resultado de x - y es " + (x - y));
        System.out.printf("\n%s%s\n", "División entera:", "x / y = ", (x/y));
        System.out.println("Resto de la división entera: x % y = " + (x % y));
        System.out.printf("El valor de f1 es %f y el de f2 es %f\n", f1, f2);
        System.out.println("El resultado de f1 / f2 es " + (f1 / f2));
    }
}
```

2.5 Ejercicio 5: Uso de parámetros de entrada

Diseña un programa Java que pida un número por teclado, determine si el número es par y muestre el resultado por pantalla.

```
import java.util.Scanner;

/*
 * ejercicio05.java
 * Determina si un numero es multiplo de otro
 */

/**
 *
 * @author cris
 */
public class ejercicio05 {

    public static void main(String[] args) {

        Scanner teclado = new Scanner(System.in);
        int x , y;
        String resultado;

        System.out.print( "Introducir numero: " );
        x = teclado.nextInt(); // pedimos el numero al usuario

        resultado = ((x%2)==0)?"es par":" es impar";
        System.out.println("\tx = " + x + " " + resultado);
    } // fin main

} // fin ejercicio05
```

2.6 Ejercicio 6: Enumerados

Diseña un programa Java que cree un tipo de dato enumerado para los días de la semana. El programa debe realizar las siguientes operaciones:

- Crear una variable `diaSemana` del tipo enumerado y asignarle el valor `viernes`. Además, se mostrará por pantalla su valor.
- Asignar a la variable `diaSemana`, la cadena de texto `"Viernes"`. A continuación, mostrar por pantalla el valor de la variable de tipo enumerado.


```
/*
 * ejercicio06.java
 * Tipos enumerados
 */

/**
 *
 * @author cris
 */
public class ejercicio06 {

    // la lista
    public enum diaSemana {
        LUNS, MARTES, MERCORES, XOVES, VENRES, SABADO, DOMINGO
    }

    public static void main(String[] args) {

        diaSemana D = diaSemana.VENRES;
        System.out.println("El valor de la variable de tipo enumerado d es: " +
d);

        String diaActual="VENRES";
        d = diaSemana.valueOf(diaActual);
        System.out.println("El valor de la variable de tipo enumerado d es: " +
d);
    }

}
```

2.7 Ejercicio 7: Ecuaciones

Diseña un programa Java para resolver una ecuación de primer grado con una incógnita (**x**), suponiendo que los coeficientes de la ecuación (**a** y **b**) se introducen desde teclado.

$$ax + b = 10$$

```
import java.util.Scanner;

/*
 * ejercicio07.java
 * Programa que resuelve una ecuación de 1er. grado
 */

/**
 *
 * @author cris
 */
public class ejercicio07 {
```

```
// clase principal que inicia la aplicacion
public static void main(String[] args) {

    // clase Scanner para petición de datos
    Scanner teclado = new Scanner(System.in);
    double a, b, x;
    String resultado;

    System.out.print( "Introducir primer numero a: " );
    c1 = teclado.nextDouble(); // pedimos el primer numero al usuario
    System.out.print( "Introducir segundo numero b: " );
    c2 = teclado.nextDouble(); // pedimos el segundo numero al usuario

    // si el coeficiente de x es cero la ecuacion no se puede calcular
    resultado = (c1==0)?"Error, la ecuacion no se puede
    calcular":String.valueOf((10-b)/a);

    System.out.println("El valor de x en la ecuacion ax + b = 10 es " +
    resultado);

    } // fin main

} // fin ejercicio07
```

2.8 Ejercicio 8: Operaciones matemáticas

Diseña un programa Java que calcule las siguientes operaciones aritméticas y algebraicas para dos números introducidos desde teclado:

- suma
- resta
- multiplicación
- división
- potencia al cuadrado del primer número
- raíz cuadrada del segundo número

Si la entrada por teclado fuesen los valores de 3 y 4, la salida del programa sería algo similar a:

```
Introducir primer numero: 6
Introducir segundo numero: 4
x = 6.0 y = 4.0
x + y = 10.0
x - y = 2.0
```

```
x * y = 24.0
x / y = 1.5
x ^ 2 = 36.0
√ x = 2.0
```

```
import java.util.Scanner;

/*
 * ejercicio8.java
 * Programa que realiza calculos con los operadores matematicos
 * y las funciones de la clase Math
 */
/**
 *
 * @author cris
 */
public class ejercicio8 {

    //clase principal que inicia la aplicacion
    public static void main(String[] args) {
        // clase Scanner para petición de datos
        Scanner teclado = new Scanner(System.in);
        double x , y;

        System.out.print( "Introducir primer numero: " );
        x = teclado.nextDouble(); // pedimos el primer numero al usuario
        System.out.print( "Introducir segundo numero: " );
        y = teclado.nextDouble(); // pedimos el segundo numero al usuario

        System.out.println("\tx = " + x + "\ty = " + y);

        // operaciones aritmeticas
        System.out.println("\tx + y = " + String.valueOf(x+y));
        System.out.println("\tx - y = " + String.valueOf(x-y));
        System.out.println("\tx * y = " + String.valueOf(x*y));
        System.out.println("\tx / y = " + String.valueOf(x/y));

        // operaciones algebraicas
        System.out.println("\tx ^ 2 = " + String.valueOf(Math.pow(x, 2)));
        System.out.println("\t√x = " + String.valueOf(Math.sqrt(x)));

    } // fin main
} // fin ejercicio8
```

2.9 Ejercicio 9: Manipulando parámetros de entrada

Diseña un programa Java que solicite un número de 6 dígitos del teclado, separe el número en sus dígitos individuales y los muestre por pantalla. Por ejemplo, si el número es 975863 que muestre:

9 7 5 8 6 3

```
import java.util.Scanner;

/*
 * ejercicio9.java
 * Programa que solicita un numero de teclado
 * e imprime sus digitos separados
 */

/**
 *
 * @author cris
 */
public class ejercicio9 {

    // clase principal que inicia la aplicacion
    public static void main(String[] args) {

        Scanner teclado = new Scanner (System.in);

        int num, d1, d2, d3, d4, d5;
        String cadena, resultado;

        System.out.print( "Introducir numero entero de 5 digitos: " );
        num = teclado.nextInt();

        d5 = num % 10;
        num = num /10;

        d4 = num % 10;
        num = num /10;

        d3 = num % 10;
        num = num /10;

        d2 = num % 10;
        num = num /10;

        d1 = num % 10;

        System.out.printf("\n\t%d\t%d\t%d\t%d\t%d\n", d1, d2, d3, d4, d5);

    } // fin main
} // fin ejercicio9
```

2.10 Ejercicio 10: Identificadores

Diseña un programa Java que realice las siguientes asignaciones y operaciones, en el orden que se muestran. Se indica la variable y el tipo de dato que recibe el valor o resultado de la operación

indicada:

Tipo_Variable resultado operación/asignación	Variable	Valores a asignar a la variable	Operación a realizar
float	x	4.5	
float	y	3.0	
int	i	2	
int	j		$i * x$
double	dx	2.0	
double	dz		$dx * y$
byte	bx	5	
byte	by	2	
byte	bz		$bx - by$
byte	bx	-128	
byte	by	1	
byte	bz		$bx - by$ (resultado tipo byte)
byte	bz		$bx - by$ (resultado tipo int)
short	sx	5	
short	sy	2	
short	sz		$sx - sy$
short	sx	32767	
short	sy	1	
short	sz		$sx + sy$

char	CX	\u000F	
char	CY	\u0001	
int	Z		CX - CY
int	Z		CX - 1
char	CX	\uFFFF	
int	Z	CX	
short	SX	CX	
short	SX	-32768	
char	CX	SX	
int	Z	SX	
short	SX	-1	
char	CX	SX	
int	Z	CX	

El resultado del programa debe ser el siguiente:

```

----- Conversiones entre enteros y coma flotante -----
Producto de int por float: j= i*x = 9
Producto de float por double: dz=dx * y = 6.0
----- Operaciones con byte -----
byte: 5 - 2 = 3
byte -128 - 1 = 127
(int)(-128 - 1) = -129
----- Operaciones con short -----
short: 10 - 1 = 3
short 32767 + 1 = -32768

----- Operaciones con char -----
char: - = 14
char(0x000F) - 1 = 14
(int)( ) = 65535
(short)( ) = -1

```

```
-32768 short-char-int = 32768
-1 short-char-int = 65535
```

```
/*
 * ejercicio10.java
 * Operaciones de conversion con numeros y caracteres
 */

/**
 *
 * @author cris
 */
public class ejercicio10 {

    public static void main(String[] args) {
        // al inicializar las variables debemos indicar que son float
        // ya que si no el compilador da error porque los literales
        // en coma flotante son double por defecto
        float x=4.5f;
        float y=(float)3.0;

        System.out.println("----- Conversiones entre enteros y coma flotante -----");
        int i=2;
        int j= (int)(i*x);
        System.out.println
            (" Producto de int por float: j= i*x = " + j);

        double dx=2.0;
        double dz=dx * y;
        System.out.println
            (" Producto de float por double: dz=dx * y = " + dz);

        System.out.println
            ("\n----- Operaciones con byte -----");
        byte bx = 5; // literal int convertido sin cast a byte
                    // si el valor está dentro de los permitidos para byte -128 to
127
        byte by = 2;

        // Aunque usemos operandos byte, el cálculo en la JVM se hace en int
        // debemos hacer un cast a byte de nuevo
        byte bz = (byte)(bx - by);
        System.out.println(" byte: 5 - 2 = " + bz);

        bx = -128; by = 1;
        bz = (byte)(bx - by);
        System.out.println(" byte -128 - 1 = " + bz );

        j = (int)(bx - by);
        System.out.println(" (int)(-128 - 1) = " + j );

        System.out.println
            ("\n----- Operaciones con short -----");
        short sx = 5; // literal int convertido sin cast a
                    // byte si está dentro del rango de short
        short sy = 2;
```

```
// Aunque usemos operandos short, el cálculo en la JVM se hace en int
// debemos hacer de nuevo el casting a short
short sz = (short)(sx - sy);
System.out.println(" short: 10 - 1 = " + sz );

sx = 32767;
sy = 1;
sz = (short)(sx + sy);
System.out.println(" short 32767 + 1 = " + sz );

System.out.println
    ("\n----- Operaciones con char -----");

/*char cx = '\u007D';// el código Unicode en hexadecimal
char cy = '\u0063';*/

char cx = '\u000F';// el código Unicode en hexadecimal
char cy = '\u0001';

// Podemos hacer operaciones con tipos char
// porque lo que realmente se almacena es el código Unicode
int z = cx - cy; // literales char 'convertidos' a int sin cast
System.out.println(" char: \u000F - \u0001 = " + z );

z = cx - 1;
System.out.println(" char(0x000F) - 1 = " + z );

cx = '\uFFFF';
z = cx; // literal char convertido a int sin cast
System.out.println(" (int)(\uFFFF) = " + z );

sx = (short)cx;
System.out.println(" (short)(\uFFFF) = " + sx );

sx = -32768; // literal short
cx = (char)sx; // convertido a char
z = cx; // convertido a int
System.out.println(" -32768 short-char-int = " + z );

sx = -1; // literal short
cx = (char)sx; // convertido a char
z = cx; // convertido a int
System.out.println(" -1 short-char-int = " + z );
}
}
```


3. Consejos y recomendaciones

La descarga de los archivos de instalación es recomendable que se realice a través de los enlaces propuestos dentro de los contenidos de la unidad de trabajo.

Para la creación de archivos “.java” es conveniente crear una carpeta donde ir alojándolos.

Puedes tomar como referencia el ejercicio resuelto de los contenidos de la unidad de trabajo.

4. Formato de entrega

Para la entrega, los ejercicios que incluyan la realización de un programa java, deberán ser entregados en un fichero/proyecto cuyo nombre sea similar a UD2_EjerXX.java, donde la XX se sustituirá por el número del ejercicio. Además, deberás incluir un pdf con las soluciones y explicaciones/justificaciones solicitadas para el resto de los ejercicios.



Para la entrega de esta tarea debes generar un único fichero .zip/.rar que contenga todos los ficheros anteriormente indicados y que debes nombrar de la siguiente forma:

- Ciclo-módulo: DAM-PROG-
- número de unidad (UD2)
- identificación de la práctica (PR.01)
- apellidos, seguidos de tu nombre, separados por guión bajo.

“DAM-PROG-UD2-PR.01-apellido1_apellido2_nombre.xxx”

Ejemplo: DAM-PROG-UD2-PR.01-Fernandez_Lopez_Maria.pdf



Nota: Asegúrate que el nombre no contenga la letra ñ, tildes ni caracteres especiales extraños. Así por ejemplo la alumna Begoña Sánchez Mañas sería:

DAM-PROG-UD2-PR.01-Sanchez_Manas_Begona

5. Recursos necesarios para realizar la tarea

- Ordenador personal
- Sistema operativo Windows o Linux
- Conexión a Internet
- JDK y JRE de Java
- Entorno NetBeans

6. Consejos y recomendaciones

Antes de afrontar la tarea debes haber leído y comprendido el contenido de la unidad.

7. Evaluación

La tarea se puntúa numéricamente según los siguientes criterios por ejercicio:

1. 0,55 ptos
2. 0,50 ptos
3. 0,60 ptos
4. 0,35 ptos
5. 1,00 ptos
6. 1,00 ptos
7. 1,00 ptos
8. 1,50 ptos
9. 1,50 ptos
10. 2,00 ptos

TOTAL: 10 ptos

8. Materiales

8.1.1 Recursos didácticos

- Apuntes en el aula virtual.
- Ordenador personal y conexión a internet
- Manual Java IES San Clemente: <https://manuais.iessanclemente.net/index.php/Java>