

UD1. IDENTIFICACIÓN DE LOS ELEMENTOS DE UN PROGRAMA INFORMÁTICO

BOLETÍN

1. Marca con una X la herramienta mínima necesaria para poder realizar las operaciones que se indican en la tabla:

<i>Operación</i>	<i>Herramientas</i>			
	JVM	JRE	JDB	JDK
Ejecutar una aplicación Java		X		
Desarrollar aplicaciones Java				X

2. Marca con una X la extensión correspondiente a cada tipo de fichero:

<i>Tipo de fichero</i>	<i>Extensión</i>	
	java	class
Fichero compilado de Java		X
Fichero fuente de Java	X	

3. Indica la afirmación correcta:
- a) Los ficheros compilados en Java están codificados en el lenguaje bytecode que puede ser interpretado por cualquier sistema operativo de forma directa.
 - b) Los ficheros compilados en Java están codificados en el lenguaje máquina para que puedan ser ejecutados por un sistema operativo concreto.
 - c) Los ficheros compilados en Java están codificados en el lenguaje máquina común a Linux y Windows para que puedan ser ejecutados cualquiera de estos sistemas operativos.
 - d) Los ficheros compilados en Java están codificados en el lenguaje bytecode para que puedan ser ejecutados por la JVM que debe estar instalada en el sistema.
4. Si suponemos que tenemos el JDK almacenado en la siguiente ruta C:\Program Files\Java\jdk-11.0.8, indica que ruta debemos de asignar a la variable JAVA_HOME.
- C:\Program Files\Java\jdk-11.0.8
5. Si tenemos un fichero mifichero.java y ejecutamos, desde la línea de comandos:

javac mifichero.java

- a. ¿Cómo se llama la operación que realiza? **Compilación del fichero mifichero.java**
- b. ¿Qué ficheros se generan en la operación? **Se crea un fichero mifichero.class en el caso de que tenga solo una clase. Si tuviera más de una clase, se crearía un fichero .class por cada clase.**

- c. De los ficheros que existen después de ejecutar `javac mifichero.java` indica cuál es el fichero fuente y el fichero objeto y en qué lenguaje está codificado el contenido de cada uno. El fichero fuente es el que está codificado en el lenguaje de programación Java (mifichero.java) y el fichero objeto (ejecutable por JVM) (mifichero.class) está codificado en *bytecode*.
- d. Indica cuál sería la orden que tendríamos que escribir en la línea de comandos para ejecutar el fichero resultante de la operación.

`java mifichero`

6. Indica cuales de las siguientes afirmaciones son verdaderas.

Operación	¿Verdadera?
Todo proyecto Java tiene que tener como mínimo una clase que debe contener un método <i>main</i>	SI
Los IDE para Java utilizan internamente las herramientas básicas de JVM para realizar las operaciones descritas	No, es el JDK
Un entorno de desarrollo integrado o IDE (Integrated Development Environment) es una aplicación que facilita el desarrollo de aplicaciones en algún lenguaje de programación	Si
El método main es private	No
El método main es estático	Si
El método main, como resultado de su ejecución devuelve o valor boolean	No, tiene como parámetro de salida <i>void</i> que indica que no devuelve nada

7. Indica las características de los comentarios que se indican en la tabla:

Símbolo de comienzo	Características de los comentarios
<code>//comentario</code>	Comentarios de una sola línea
<code>/*comentario*/</code>	Comentarios de varias líneas
<code>/**comentario*/</code>	Comentarios para crear documentación javadoc

8. Indica si el identificador es válido en cada caso, si no lo es, indica por qué.

Identificador	Válido/No válido	Explicación en el caso de que no sea válido
<code>MiVariable</code>	válido	
<code>MIVARIABLE</code>	válido	
<code>Mi Variable</code>	no válido	no se permiten espacios en blanco
<code>123Java</code>	no válido	no puede empezar por un número
<code>X</code>	válido	
<code>variable-2</code>	válido	
<code>X1</code>	válido	
<code>a+c</code>	no válido	no se permiten el carácter +
<code>class</code>	no válido	no se permiten palabras reservadas
<code>_mivariable</code>	válido	
<code>\$mivariable</code>	válido	
<code>suma_de_array</code>	válido	
<code>aprenderJava</code>	válido	
<code>suma&resta</code>	no válido	no se permite el símbolo &
<code>true</code>	no válido	no se permiten palabras reservadas

suma.de.array	válido	
---------------	--------	--

9. Indica de qué tipos son los literales siguientes y señala aquellos que no son correctos.

Literal	Tipo	Comentarios
982	int	
1234L	long	El tipo long lo indica la L final.
0b0100101	int	Literal en binario de tipo int (empieza en 0b y le sigue un número en binario)
'a'	char	
12.565	double	
"Tolosa"	String	
0.354	double	
154_.657	Incorrecto	No puede estar el _ antes de un punto
+1200	int	
'5'	char	
.006E4	double	Notación científica
" "	String	
0b1010000101000101101000	int	En binario
'\n'	char	Salto de línea
34.56L	long	
234_456_345L	long	El número se lee sin _
-25	int	
'\u0061'	char	En hexadecimal, el código Unicode
02344	int	Entero en base octal
1234	int	
.87	double	
2.34E3	double	Notación científica
+74.356E-3F	float	Notación científica

Operadores lógicos:

10. Si tenemos la expresión $a > b$ indica si es cierto o falso las siguientes afirmaciones. Probar en el ordenador.

Afirmación	Cierto
a y b pueden ser de tipo char	x
a y b pueden ser de tipo String	Para comparar usamos un método específico
a y b pueden ser de tipo float	x
a y b pueden ser de tipo boolean	No se pueden comparar valores boolean

11. Para hacer en el papel: Indica que muestra por pantalla este código:

```
int a = 20, b = 10, c = 0, d = 20, e = 40, f = 30;
boolean condition = true;
c = ++a;
System.out.println("Valor de c (++a) = " + c);
c = b++;
```

```

System.out.println("Valor de c (b++) = " + c);

c = --d;
System.out.println("Valor de c (--d) = " + c);

c = e--;
System.out.println("Valor de c (e--) = " + c);

System.out.println("Valor de !condition = " + !condition);

```

```

Valor de c (++a) = 21
Valor de c (b++) = 10
Valor de c (--d) = 19
Valor de c (e--) = 40
Valor de !condition = false

```

12. Implementar un proyecto que, dado un número de horas, calcule el número de semanas, días y horas equivalentes. Por ejemplo, para 1000 horas debe mostrar 5 semanas, 6 días y 16 horas.

```

public static void main(String[] args) {
    int totalH, semanas, dias, horas;
    Scanner miScanner=new Scanner(System.in);
    System.out.println("Introduce el número de horas para realizar
el cálculo");
    totalH = miScanner.nextInt();
    semanas = totalH / (24 * 7);
    dias = totalH % (24 * 7) / 24;
    horas = totalH % 24;
    System.out.println("El número total de horas es " + totalH);
    System.out.println("que es equivalente a:");
    System.out.println(semanas + " semanas");
    System.out.println(dias + " dias");
    System.out.println(dias + " horas");
}

```