

Módulo: Programación.

Tema 1: Java Básico. Noviembre 2017.

APELLIDOS: _Barbara_____ NOMBRE: _Angelo_____
CURSO: _1 DAW_____ FECHA: _20/11/2017_____

PREGUNTAS TIPO TEST (penalizan)

- Indica la afirmación falsa:
 - La JVM interpreta el código Java
 - El código fuente de Java ha de precompilarse generando bytecodes
 - Los archivos .java contienen el código bytecode **FALSO**
 - Los bytecodes pueden ejecutarse en cualquier plataforma con la JVM adecuada.
- Indica la respuesta falsa. En Java:
 - Un literal de tipo entero se considera como un byte **FALSO**
 - Un literal de tipo decimal se considera como un double
 - Un literal de tipo entero se considera como un int
 - Se utiliza el sufijo L para indicar que el literal es un long (200L)

PREGUNTAS CORTAS

- Comando que genera la documentación javadoc (incluyendo las etiquetas del autor y de la versión) para la clase Anacleto.java __javadoc -private -author -version Anacleto.java__
- Los ficheros con extensión .java se denominan código: __código fuente__
- Indica todos los tipos de comentarios válidos que conozcas y explícalos brevemente

/** */ comentario javadoc. Presente en la documentación javadoc
/* */ comentario de bloque. No está presente en la documentación javadoc
// comentario de línea. No está presente en la documentación javadoc

- Indica los ocho tipos de datos primitivos soportados por Java: __byte, short, int, long, float, double, char, boolean,____

ANALIZANDO CÓDIGO (8 PUNTOS)

- Comenta a la derecha lo que mostraría cada System.out.println()...

```
int i = 256;  
byte b = (byte)i;  
System.out.println(i); 256  
System.out.println(b); 0
```

- Considera el siguiente código. Tras su ejecución indica los valores de la variable i : __y n: __

```
int i = 10; 11  
int n = ++i%5; 1
```

- Añade a la derecha de las sentencias System... un comentario con el valor que muestra:

```
boolean b1, b2;  
int entero = 21;  
b1 = entero >= 18;  
b2 = !b1;  
System.out.println("" + b1); //muestra: TRUE  
System.out.println("" + b2); //muestra: FALSE
```

- Explica detalladamente cada una de las siguientes líneas de código:

```
int num1 = 16; declaración de una variable de tipo entero 16  
int num2 = 020; declaración de una variable entero expresada en octal  
int num3 = 0x14; declaración de una variable entero expresada en hexadecimal  
System.out.println(num1 < num2); //muestra: FALSE  
System.out.println(num3 < num2); //muestra: FALSE  
System.out.println(num1 == num2); //muestra: TRUE
```

- Explica si hubiera algún tipo de error en estas sentencias:

```
int i = 12;  
byte b = i; Es necesario poner el casting (byte)i o dará error "possibly lost of conversion"  
char letra;  
letra = "C"; Es necesario poner comillas simples, 'C', o dará error ya que las comillas  
simple se utilizan para el String  
letra = 67;
```

12. Considera el siguiente código y contesta:

```
public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);
    do {
        System.out.println("Dame un entero: ");
        int a = scanner.nextInt();
        System.out.println("Dame otro entero: ");
        int b = scanner.nextInt();
        System.out.println("Mayor o igual: "+devuelveMayorOIgual(a, b));
    } while (deseaContinuar());
}
```

- En el siguiente instante de la ejecución selecciona de las tres cuál es la pila de llamadas:

Dame un entero:

11

Dame otro entero:

22

Mayor o igual: 22

¿Desea continuar?(s/n)

d

¿Desea continuar?(s/n)

S

Dame un entero:

MAIN

deseaContinuar
devuelveMayorOIgual
main

main

deseaContinuar
main

- Implementa el método devuelveMayorOIgual mediante el operador ternario

```
static int devuelveMayorOIgual(int a, int b){
    return (a<b)?b:a;
}
```

- Implementa el método deseaContinuar. Fíjate que sólo admite los cuatro caracteres válidos “n”, “s”, “N” ó “S”. Utiliza la clase Teclado.leerCaracter(), pero no utilices bucles infinitos.

```
static boolean deseaContinuar(){
    char continuar;
    do{
        continuar=Teclado.leerCaracter("Desea continuar?(s/n)");
    } while(continuar!='s' && continuar!='S' && continuar!='n' && continuar!='N');
    return (continuar=='s' || continuar=='S');
}
```

