



## **Semana 3: Declaración de variables y operadores aritméticos en Java**

### **Programación**

### **2018 - 2019**

### **Grado en Ingeniería Informática**

**Nota:** salvo que se diga lo contrario cree un programa nuevo (una clase) para cada ejercicio. Algunos ejercicios están diseñados para que aparezcan errores de compilación. En esos casos, no borre la sentencia que contiene el error, coméntela poniendo `//` al inicio de esa línea.

#### **Ejercicios a realizar**

##### **Ejercicio 1. Declaración de variables**

Cree una clase denominada `Ejercicio1`. Dentro de ella cree un método `main` y dentro de él en sucesivas líneas declare una variable distinta para cada uno de los 8 tipos básicos de java. Declare también una variable de tipo `String`. A continuación en líneas sucesivas asigne valores adecuados a cada una de estas variables. Muestre el valor de las variables por pantalla. Ejemplo para el caso de una variable de tipo `int`:

```
int varEntera;  
varEntera = 3456;  
System.out.println(varEntera);
```

##### **Ejercicio 2. Uso de variables no inicializadas**

Declare una variable de cualquier tipo sin inicializarla. Imprima esta variable por pantalla, ¿qué ocurre? ¿Por qué?

Cree otra variable del mismo tipo e iguálela a la primera variable, ¿qué ocurre? ¿Por qué?

##### **Ejercicio 3. Cambio del valor de una variable**

Cree una variable y asígnele un valor en la declaración, a continuación, en una nueva instrucción, asigne un nuevo valor a esa variable. ¿Es posible cambiar el valor de una variable? ¿Importa el tipo de la variable? Muestre el nuevo valor de la variable por pantalla.

##### **Ejercicio 4. Constantes**

Cree una constante de cualquier tipo (poniendo `final` antes del tipo) y asígnele un valor en la declaración. A continuación, en otra línea cambie el valor de la constante, ¿es posible? ¿Por qué?

Cree otra constante pero ahora no le asigne valor en la declaración sino en la línea siguiente ¿es posible? ¿Por qué?

En una línea posterior, cambie el valor de la segunda constante, ¿es posible? ¿Por qué?

##### **Ejercicio 5. Rangos de los tipos básicos numéricos**

Declare una variable para cada tipo numérico y pruebe a dar nuevos valores que sobrepasen los rangos de cada uno de los tipos, ¿qué ocurre?

**Ejercicio 6. Precisión de los distintos tipos numéricos**

Declare y dé valor a dos variables de tipo `float`, la primera valdrá `1234567890.0F` y la segunda `1234567899.0F`. Imprima por pantalla la resta de ambas. ¿Cuál es el resultado? Haga lo mismo pero con dos variables de tipo entero (sin decimales y sin utilizar el modificador `F`). ¿Cuál es el resultado? ¿Por qué?

**Ejercicio 7. Declaraciones simultáneas**

Declare de una sola vez (en una sola sentencia) 3 variables del mismo tipo básico (cualquiera de ellos). En líneas posteriores inicialice cada una de ellas.

Declare también, de una sola vez, tres variables de cualquier tipo numérico, inicializando la primera y la última en la misma declaración. A continuación en otra línea inicialice la segunda variable.

**Ejercicio 8. Declaraciones de variables repetidas**

Cree una clase y dentro de ella un método `main`. Dentro de él declare una variable de cualquier tipo, a continuación vuelva a declarar otra vez la misma variable con el mismo tipo. ¿Es posible? ¿Por qué? ¿Y si declaramos la variable con un tipo distinto?

**Ejercicio 9. Caracteres (tipo `char` y `String`)**

¿Es posible crear un `char` con más de un carácter (ejemplo `char b = 'hola'`)? ¿Y un `String`?

**Ejercicio 10. Copia de variables**

Cree dos variables de cualquiera de los tipos básicos, dé valor a la primera y a continuación iguale la segunda a la primera. Imprima por pantalla la segunda. En una línea posterior cambie el valor de la primera y vuelva a imprimir la segunda, ¿cambia la segunda de valor? ¿Por qué? Haga lo mismo para dos variables de la clase `String`.

**Ejercicio 11. Operaciones aritméticas con enteros (I)**

Cree un programa y dentro de él teclee lo siguiente:

```
int a,b;
int c,d;
a = 2;
b = 3 * 3;
c = 7 / 3;
d = a + b * c;
System.out.println(d);
```

¿Cuál es el resultado del código anterior? ¿Por qué?

**Ejercicio 12. Operaciones aritméticas con enteros (II)**

Declare tres variables para cada uno de los cuatro tipos enteros (doce variables en total) y otras tres de `char`. Para cada uno de los tipos, inicialice las dos primeras variables con un valor cualquiera y a la tercera dele consecutivamente en líneas distintas el valor de operar las dos anteriores con todos los operadores aritméticos de Java (suma, resta, etc.). Imprima los resultados por pantalla.

¿Se produce algún fallo? ¿Por qué? (Nota: una vez descubiertos los fallos, para que el programa pueda compilar, en lugar de borrar las líneas en las que hay fallos, conviértalas en comentarios usando //)

**Ejercicio 13. Operaciones aritméticas con enteros (III)**

Declare tres variables de tipo `int`. Dé valor 5 a la primera y valor 0 a la segunda. Haga que la tercera sea igual a la primera dividida por la segunda. Imprima el resultado por pantalla. ¿Se produce algún fallo? ¿Por qué? ¿Cambia el resultado si las variables en lugar de ser de tipo `int` son de cualquier otro tipo entero?

**Ejercicio 14. Operaciones fuera de rango**

Cree una variable de tipo `int` o `long` y asígnele un valor, usando operadores aritméticos, que esté fuera del rango de ese tipo de variables ¿Es posible? En caso afirmativo, imprima la variable por pantalla, ¿qué ocurre? Haga lo mismo para una variable `float` o `double`.

**Normas de entrega**

Los ejercicios se deben **subir a Aula Global** hasta las 9:00 del lunes 24 de Septiembre de 2018. Se deberá subir un fichero comprimido **zip** con los archivos `.java` de la carpeta **src** (se deben subir únicamente esos archivos, no el proyecto Eclipse completo). El nombre del fichero será “s3-iniciales-del-alumno1-iniciales-del-alumno2.zip” (por ejemplo Lucía Pérez Gómez y Juan García Jiménez subirán un archivo llamado s3-lpg-jgj.zip). **Ambos miembros de la pareja deberán subir el fichero.** Se debe usar el enlace de entrega del grupo de laboratorio al que se pertenezca.