

01 - Ejercicios: Variables y operadores

Para realizar estos ejercicios crearemos un nuevo proyecto llamado Ejercicios01. Dentro de él crearemos un paquete para cada clase que creemos. Una clase por ejercicio.

***** Los nombres (identificadores) de las variables deben ser representativos, usando el formato "lowerCamelCase".**

1. Programa java que realice lo siguiente: declarar una variable de tipo int, una variable de tipo double y una variable de tipo char y asigna a cada una un valor. A continuación muestra por pantalla: El valor de cada variable, la suma de int + double, la diferencia de double - int, el valor numérico correspondiente al carácter que contiene la variable char.

2. Programa java que realice lo siguiente: declarar una variable de cada tipo de datos primitivo y asignar a cada una un valor literal (excepto boolean). A continuación, realiza una serie de operaciones entre ellas (una por cada variable) y muestra por pantalla su resultado. (suma, resta, multiplicación, división y resto).

3. Programa Java que declare una variable entera "num" y asígnele un valor. A continuación escribe las instrucciones que realicen lo siguiente: Incrementar num en 77, decrementarla en 3, Duplicar su valor. Después de cada operación el valor de la variable se debe haber modificado.

4. Programa que declare cuatro variables enteras variableA, variableB, variableC y variableD y asígnele un valor a cada una. A continuación, realiza las instrucciones necesarias para que: B tome el valor de C, C tome el valor de A, A tome el valor de D, D tome el valor de B. ¿Qué necesitamos si no queremos perder ningún valor?

5. Programa que declare una variableA de tipo entero y asígnele un valor. A continuación muestra un mensaje indicando si A es par o impar. Utiliza el operador condicional (? :) dentro del println para resolverlo. En otro mensaje indicar si A es a la vez múltiplo de 2 (par) y múltiplo de 3.

6. Programa que declare una variable `variableA` de tipo entero y asigne un valor. A continuación muestra un mensaje indicando si el valor de `A` es positivo o negativo. Consideraremos el 0 como positivo. Utiliza el operador condicional (`? :`) dentro del `println` para resolverlo.

7. Programa que declare una variable `variableA` de tipo entero y asigne un valor. A continuación muestra un mensaje indicando si el valor de `A` es positivo o negativo, si es par o impar, si es múltiplo de 5, si es múltiplo de 10 y si es mayor o menor que 100. Consideraremos el 0 como positivo. Utiliza el operador condicional (`? :`) dentro del `println` para resolverlo.

8. Dadas las variables de tipo `int` con valores `a = 5`, `b = 3`, `c = -12` indicar si la evaluación de estas expresiones daría como resultado verdadero o falso (mostrarlo por pantalla). Para evaluarlas asignamos la expresión a una variable de tipo `boolean`, y la mostramos por pantalla:

- | | | |
|--|--|--------------------------------|
| a) <code>a > 3</code> | b) <code>a > c</code> | c) <code>a < c</code> |
| d) <code>b < c</code> | e) <code>b != c</code> | f) <code>a == 3</code> |
| g) <code>a * b == 15</code> | h) <code>a * b == -30</code> | i) <code>c / b < a</code> |
| j) <code>c / b == -10</code> | k) <code>c / b == -4</code> | l) <code>a + b + c == 5</code> |
| m) <code>(a+b == 8) && (a-b == 2)</code> | n) <code>(a+b == 8) (a-b == 6)</code> | |
| o) <code>a > 3 && b > 3 && c < 3</code> | p) <code>a > 3 && b >= 3 && c < -3</code> | |

9. Crea 2 variables de tipo `short`, y 2 de tipo `long`. Dale a las variables de tipo `long` el máximo mayor positivo, y el mínimo negativo. Luego asigna cada una de esas variables `long` a las variables de tipo `short`. Utiliza la conversión **cast** y muestra las 4 variables por pantalla. Además muestra un mensaje indicando qué es lo que ocurre en ambos casos.

10. Crea una variable de tipo `float` y otra variable de tipo `double`. Asigna la variable `double` a la variable `float`. Explica qué has hecho.

11. Usar el operador ? para que me indique si un número es positivo y par. Si no, que me indique lo contrario.

12. Usar el operador ? para que me indique si un número es o múltiplo de 3 o múltiplo de 7, y además tiene que ser positivo. En caso afirmativo nos dirá que cumple las características nombradas, y sino, que no las cumple.

13. Crea una variable tipo String y dale valor. Muestra en un mensaje con println la longitud de la cadena de texto.

14. Crea 4 variables String y les introducimos una cadena que represente un número decimal, un número entero largo (long), un número tipo byte, y un número int. Ahora crea 4 variables de los tipos anteriores, y asigna el valor de las variables String a las variables del tipo concreto usando los métodos necesarios.

15. Crea una variable de tipo String que contenga una cadena de texto larga (más de 20 caracteres). Utiliza el método substring de la clase String, para obtener una cadena intermedia, dentro de la cadena principal. Además se deben crear 2 constantes tipo String. Una con el mensaje "Comienza el programa" y la otra con "Termina el programa". Mostrar cada una antes y después del resto del código.

16. Mediante el operador ? y creando solo una variable tipo String, la inicializamos con un texto que represente un número entero (Ej: "1234"). Debes comprobar si la cadena tiene 5 o cifras. Si se cumple la condición me debe mostrar el valor numérico de dicha cadena y si no se cumple, debe mostrar un mensaje diciendo la cadena tiene menos de 5 cifras.

17. Mediante el operador ? y creando dos variables tipo String, debemos inicializarlas con 2 cadenas de texto. El programa debe comparar ambas cadenas y mostrarme uno de los siguientes mensajes, dependiendo del valor de las cadenas:

- -"cadena1" es mayor que "cadena2"
- -"cadena2" es mayor que "cadena1"
- -"cadena1" es igual que "cadena2"

Donde “cadena1” y “cadena2” son los valores que he almacenado en ambas variables. Las cadenas se comparan atendiendo al valor de sus caracteres. Usar método de la clase String.

18. Leer desde teclado un valor de cada tipo de datos primitivos que conocemos y almacenarlo en una variable de cada tipo. Para ello mostraremos un mensaje indicándonos el tipo de variable que debemos de meter. Un mensaje para cada tipo. Posteriormente mostrar por pantalla cada uno de los datos. (Usaremos la clase Scanner)