



3. Variables y tipos de datos

 Date

Empty

 Status

Not started

 Type

Empty

Units



2. Estructura de un programa informático

Variables

Constantes

Tipos de datos

Literales

Variables

Se define variable como cualquier elemento que guarde datos que serán utilizados en código algún momento. Para su uso se define el tipo, su nombre y su valor

```
tipo nombre = valor String nombre = "variable"; int edad = 2
0; boolean experiencia = false;
```

Hay que tener en cuenta las siguientes cosas a la hora de trabajar con variables:

- Nombres descriptivos
- No utilizar palabras reservadas
- Comienzan en minúsculas o `_`, nunca con números
- Cada palabra con mayúsculas excepto la primera

Según su lugar de declaración pueden existir:

- Variables globales: aquellas declaradas al comienzo de la clase. Su utilización puede ser en toda la clase
- Variables de método: aquellas declaradas dentro de un bloque (como por ejemplo un método). Su utilización se restringe al método donde han sido declaradas
- Variables estáticas: aquellas que son declaradas con la palabra reservada `static` y pueden ser accedidas sin necesidad de crear un objeto de la clase.
- Variables finales o constantes: aquellas que no pueden cambiar su valor. Su nombre debe ir en mayúsculas.

```
public class Introduccion { // variables globales String nombre = "Develop"; int edad = 19; boolean experiencia = false; // variable estática static String nombreAccesible = "Elemento"; // variable final final static double PI = Math.PI; protected void metodoInicial(){ // variables de método String nombre = "Luis"; char letra = 'a'; System.out.println(nombre); System.out.println(this.nombre); } protected void metodoInicialDos(){ // variable de método no accesible System.out.println(letra); } }
```

Constantes

Una constante es un valor de un tipo con un nombre que no se puede modificar durante la ejecución del programa. En su definición se acompañan con la palabra reservada `final`. Existen multitud de constantes creadas en el lenguaje. Un claro ejemplo son los valores máximos y mínimos de cada uno de los tipos de datos primitivos numéricos y de carácter se almacenan en las siguientes constantes

globales:

1. Byte.MIN_VALUE || Byte.MAX_VALUE
2. Short.MIN_VALUE || Short.MAX_VALUE
3. Integer.MIN_VALUE || Integer.MAX_VALUE
4. Long.MIN_VALUE || Long.MAX_VALUE
5. Float.MIN_NORMAL || Float.MAX_VALUE
6. Double.MIN_NORMAL || Double.MAX_VALUE
7. (int) Character.MIN_VALUE || (int) Character.MAX_VALUE

Como se puede ver, todas las constantes por definición se definen con nombres en mayúsculas. Del mismo modo, para que estas sean accesibles desde fuera de la clase de forma directa, suelen ir acompañadas del modificador static

Tipos de datos

Los tipos de datos utilizados en programación son:

Tipo	Tamaño y formato	Rango
	Enteros	
byte	8 bits - complemento a 2	\$\$ -2^7 al $2^7 - 1$ \$\$
short	16 bits - complemento a 2	\$\$ -2^{15} a $2^{15} - 1$ \$\$
int	32 bits - complemento a 2	\$\$ -2^{31} al $2^{31} - 1$ \$\$
long	64 bits - complemento a 2	\$\$ -2^{63} al $2^{63} - 1$ \$\$
	Reales	
float	32 bits - IEEE 754	
double	64 bits - IEEE 754	
	Resto	
char	16 bits - caracteres UNICODE	
boolean	1bit	

Literales

Literales

Los valores literales son aquellos que podemos asignar a las variables. Dependiendo del tipo de variable podremos asignar unos valores u otros, tal y como se ha visto en los puntos anteriores. Además de todos los tipos literales vistos en la tabla anterior, también se pueden utilizar elementos especiales, siempre y cuando comiencen con la barra invertida :

carácter	acción
b	retroceso
t	tabular la cadena
n	salto de línea
f	form feed
r	retorno de carro
'	comilla simple
"	comilla doble

1. Declara una variable entera llamada "edad" con un valor inicial de 25.
2. Declara una variable booleana llamada "esVerdadero" con un valor inicial de true.
3. Declara una variable de cadena llamada "nombre" con un valor inicial de "Juan".
4. Declara una variable de doble precisión llamada "pi" con un valor inicial de 3.1416.
5. Declara una constante de entero llamada "MAX_EDAD" con un valor de 100.