

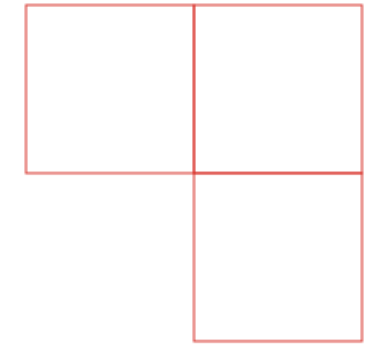
2

Sentencias y errores

Ve más allá

CONTENIDOS

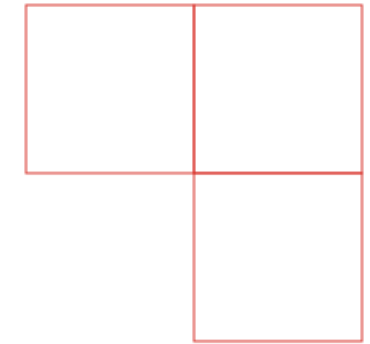
- **Unidad 1: Fundamentos del lenguaje Java**
 1. Nuestro primer programa
 - 2. Sentencias y errores**
 3. Variables y constantes
 4. Tipos de datos
 5. Operadores y expresiones
 6. Entrada y Salida





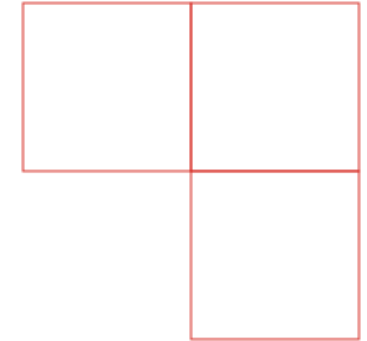
ÍNDICE

- Sentencias y errores
 1. Sentencias en Java
 2. Tipos de errores
 3. Consejos sobre los errores
 4. Comentarios
 5. Convenciones





Sentencias en Java



Las sentencias nos permiten realizar acciones en Java. Ejemplos de sentencias en Java son:

```
int entero = 1;
System.out.println("Hola " + nombre + ", ¿cómo estás? ");
import java.lang;
suma = entero + 5;
```

Fijaos que todas las sentencias deben finalizar con el símbolo punto y coma (;).

También es posible convertir un conjunto de sentencias como si se tratase de una sola, encerrándolas entre llaves ({ y }). Es útil, por ejemplo, para el uso de sentencias condicionales o bucles.



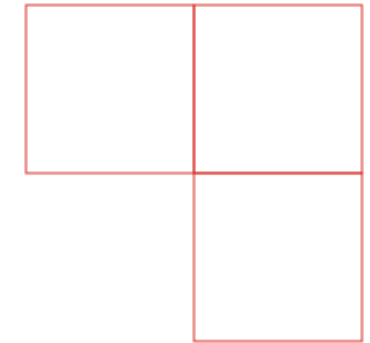
Tipos de errores

Como si se tratase de errores ortográficos, se suele hablar de los siguientes tipos de errores en programación:

1. **Léxico:** Escribir mal una sentencia del lenguaje, por ejemplo, escribir prit en lugar de print.
2. **Sintáctico:** Escribir una instrucción sin poner el ; al final de la misma
3. **Semántico:** Escribir un programa que se rompe o no hace lo que se esperaba

En programación a los dos primeros se les suele denominar errores sintácticos o de compilación y los detecta el propio entorno de desarrollo (eclipse en nuestro caso). Mientras que los últimos se suelen denominar errores de ejecución o excepciones.

Programación/Sentencias y errores



Errors



```
System.ouch.println("Hello, World!");
```

compile-time error
(syntax error)

```
System.out.println("Hello, Word!");
```

run-time error
(logic error)

```
System.out.println(1 / 0);
```

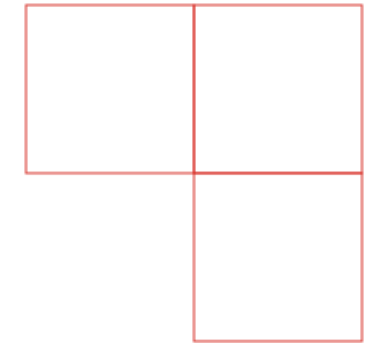
Doesn't compile? ☐
Compiles, runs, prints wrong value? ☐
Compiles, runs, prints error msg? ☐
Keyboard melts? ☐



<- solución



Tipos de errores



Text and Numbers

```
System.out.println(3 + 4 + 5);  
System.out.println("3 + 4 + 5");
```

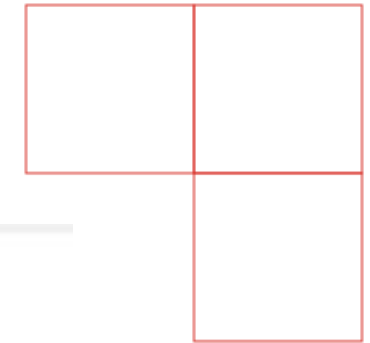


What will these lines print?





Tipos de errores



Print and Println

```
System.out.print(3);  
System.out.println(4+5);
```



What do these lines print?

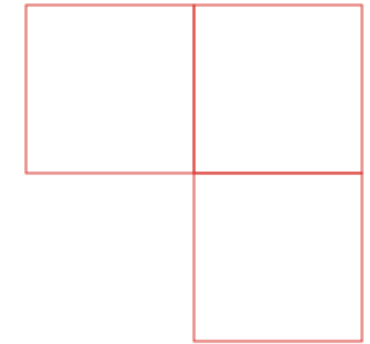




Consejos sobre los errores

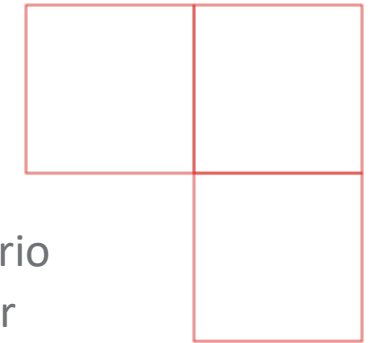
- Cuando empiezas a programar es muy habitual tener muchos errores. Cuando te pase a ti, trata de que no te agobien demasiado, recuerda que sólo se aprende cometiendo errores y que todos los programadores han pasado por una situación parecida.
- Muchos errores suelen ser pequeños detalles, que al principio suelen pasar desapercibidos. Trata de mirar no solamente donde está el error sino también un poco antes, por ejemplo, si has declarado una variable con otro nombre. Revisa tu código línea a línea y no esperes a tener el código completo para hacerlo.

Programación/Sentencias y errores





Comentarios



- En Java, usamos los comentarios para dar explicaciones sobre el código. Todo lo que pongamos como comentario será ignorado por el compilador, es decir, no se ejecutará. Sólo está ahí para que los humanos puedan entender mejor el código.
- Java permite tres tipos de comentarios, utilizando los símbolos:

// (doble barra). Permite comentar una sola línea.

Ejemplo: // este es un comentario de una línea

/* y */. Se usan para comentarios de más de una línea.

Ejemplo: /* Este comentario ocupa

un par de líneas */

/** y */. Utilizados para la documentación de javadoc.

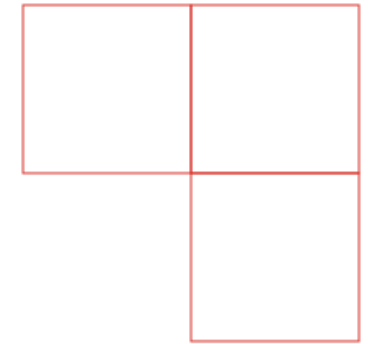
- Este sistema permite documentar a partir del código fuente.

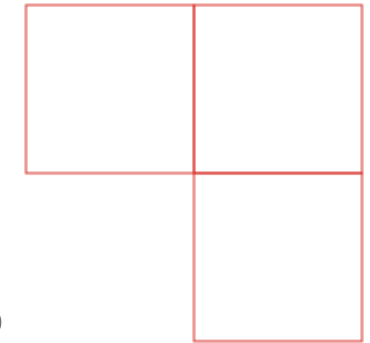


Comentarios

Comments

Programación/Sentencias y errores



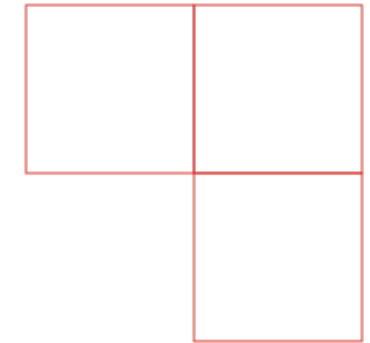


Convenciones

Convenciones de programación son un conjunto de directrices para un lenguaje de programación concreto que recomienda estilo, prácticas, y métodos de programación para cada aspecto de un programa escrito en cada lenguaje. Estas convenciones normalmente comprenden gestión de archivos, sangría, comentarios, declaraciones, sentencias, espaciado, convenciones de nombres, buenas prácticas de programación, principios de programación, buenas prácticas de arquitectura, etc.

Las convenciones de codificación son importantes para los programadores por varias razones:

- El 80% del coste del tiempo de vida de una pieza de software se va en mantenimiento.
- Casi nunca ningún software es mantenido durante toda su vida por su autor original.
- Las convenciones de nombrado mejoran la lectura del software, permitiendo a los ingenieros entender el nuevo código más rápidamente y mejor.
- Si lanzamos nuestro código fuente como un producto, necesitamos asegurarnos de que está tan bien empaquetado y limpio como cualquier otro producto que creemos.
- Para que las convenciones funcionen, todos aquellos que escriban software debe adherirse a ella.



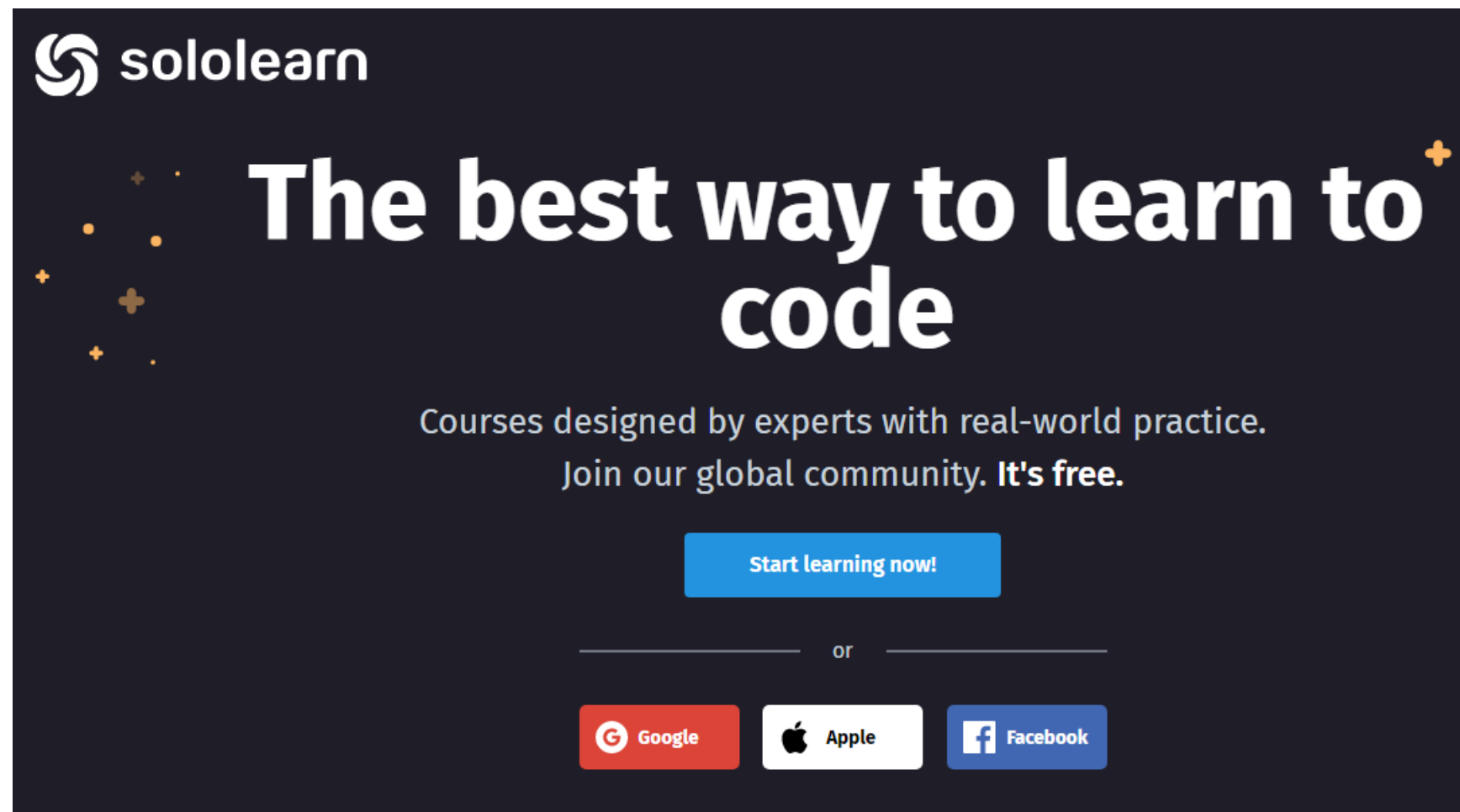
Convenciones

Lo habitual es que cuando te incorpores a una empresa, sea allí donde te den las convenciones. Pero si no fuera así, siempre puedes recurrir al documento original realizado por Sun Microsystems (empresa creadora del lenguaje Java). Puedes encontrar dicho documento en el siguiente enlace:





APUNTE...



sololearn

The best way to learn to code

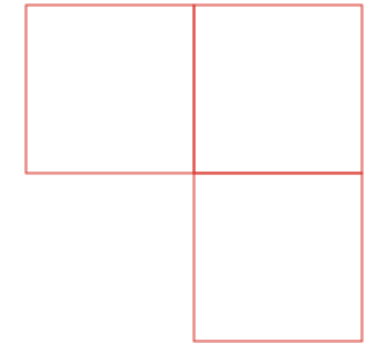
Courses designed by experts with real-world practice.
Join our global community. **It's free.**

Start learning now!

or

Google Apple Facebook

Programación/Unidad 1





**Universidad
Europea**

GRACIAS

Pedro J. Camacho

Universidadeuropea.com

Ve más allá