

Fundamentos de Programación con JavaScript

Sintaxis

02

JS



**código
Javascript**

JS

¿ Donde podemos colocar código JavaScript ?

1. En un archivo HTML, dentro de `<script>... ..</script>`
 - Justo antes de la etiqueta de cierre `</head>`
 - En cualquier lugar del `<body>`
 - En un evento de un elemento `html`
 - Justo antes de la etiqueta de cierre `</body>` ✓
2. En un archivo externo con extensión `.js`
enlazándolo así: `<script src="archivo.js"></script>`

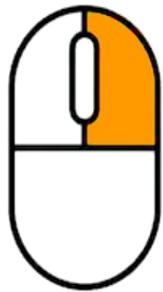
Depuración: corregir código JavaScript

Utilizando las herramientas para
desarrolladores de cualquier navegador:

SOLAPA **[consola]**

Ctrl+May+I

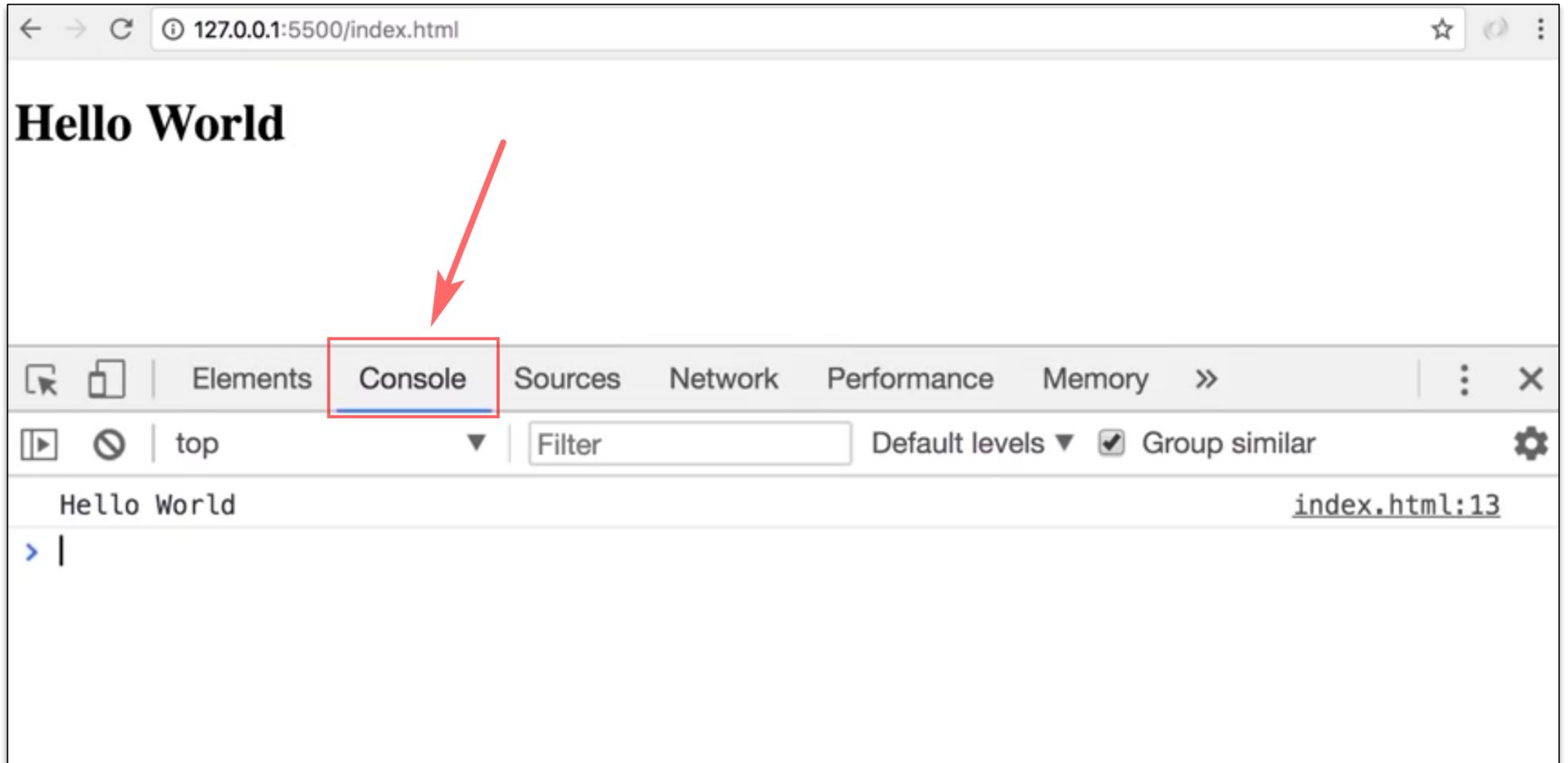
url: [about:blank](#)



inspeccionar



Depuración: corregir código JavaScript



Sintaxis JavaScript

- **JS distingue entre Mayúsculas y minúsculas (case sensitive)**
- No se tienen en cuenta espacios en blanco o saltos de línea
- Los nombres de variables y otros objetos javascript
 - No deben comenzar por números
 - No deben tener ningún signo de puntuación (excepto `_` o `$`)
 - No deben ir acentuados
 - No deben tener espacios en blanco
 - No se pueden utilizar palabras reservadas del lenguaje
 - Los bloques finalizan con `}`
- Es conveniente, pero no necesario
 - Terminar cada sentencia con `;`
 - Poner cada sentencia en una línea
 - Sangrar (tabular) el código

Palabras reservadas

No se pueden utilizar palabras reservadas del lenguaje

```
A: abstract
B: boolean, break, byte
C: case, catch, char, class, const, continue
D: debugger, default, delete, do, double
E: else, enum, export, extends
F: false, final, finally, float, for, function
G: goto
I: if, implements, import, in, instanceof, int, interface
L: long
N: native, new, null
P: package, private, protected, public
R: return
S: short, static, super, switch, synchronized
T: this, throw, throws, transient, true, try, typeof
V: var, volatile, void
W: while, with
```

Comentarios

Los comentarios en el código sirven para ayudar al programador u a otra persona a comprender mejor el funcionamiento del código, o simplemente para dejar anotaciones en él.

El texto que se escribe como comentario no se ejecuta.

- **Línea simple:** colocando delante **dos barras inclinadas**
// esto es un comentario de una sola línea
- **Multilínea:** encerrado entre **barra-asterisco ... asterisco-barra**
/
esto es un comentario que ocupa
varias líneas de texto dentro del código .js
/

Estilos de escritura de código

Se utilizan cuando necesitamos escribir dos o más palabras juntas:

Camel case (cajaDeCamello) comienza en minúsculas y la primera letra del resto de palabras en mayúsculas.

Pascal case (CajaDePascal) es similar a camel case, excepto que la primera letra siempre está en mayúscula.

Snake case (caja_de_serpiente) las palabras se separan con un carácter de subrayado (_) sin espacios y todas en minúsculas.

Tipos de datos (dinámicos = no es necesario declarar el tipo)

Primitivos (*typeof ...*)

Se accede directamente al valor

- **string**
- **number**
- **boolean**
- **null**
- **NaN**
- **undefined**

Compuestos

Se accede a ellos por **referencia**

- **Function ()**
 - **Array = []**
 - **Object = { }**
 - **Class { }**
- etc...

Tipos de datos (dinámicos = no es necesario declarar el tipo)

Primitivos (*typeof ...*)

Se accede directamente al valor

- String (cadenas) >> `let palabra1 = "hola"` o `pablabra2 = 'Mundo'`
- number >> no distingue entre enteros y reales (separador decimal = el punto)
- boolean >> solo admite **true** o **false**
- undefined >> valor dado por javascript
- null >> valor dado por el usuario

Orden del código

- 1. Importación de módulos**
- 2. Declaración de variables**
- 3. Declaración de funciones**
- 4. Código JS**

Fundamentos de Programación con JavaScript



JS