

Tema 2 Introducción al JavaScript.

Practica 02-02 JS Tipos de Datos

Nombre	Ismael	Curso	2 DAW
Apellidos	González Tempa	Fecha	

1. ¿Qué etiqueta se utiliza para insertar código JavaScript en HTML?

Se usa la etiqueta `<script> ... </script>`.

2. ¿Qué ventajas tiene incrustar el código JavaScript en un archivo externo?

Permite reutilizar el código en varias páginas, facilita el mantenimiento, mejora la separación entre HTML, CSS y JS, y el navegador puede almacenarlo en caché para optimizar el rendimiento.

3. ¿Qué extensión suelen tener los archivos que contienen código JavaScript?

La extensión es `.js`.

4. ¿Cómo se referencia un archivo `.js` en HTML?

`<script src="archivo.js"></script>`

5. ¿Qué debe hacer si el archivo JavaScript se encuentra en otro directorio?

`<script src="js/archivo.js"></script>`

6. ¿Qué etiqueta se usa para mostrar un mensaje si el navegador no soporta JavaScript?

Se usa la etiqueta `<noscript> ... </noscript>`.

7. ¿Qué diferencia hay al insertar JavaScript en XHTML?

`<![CDATA[...]]>`.

8. ¿Qué es la ofuscación de código y por qué se utiliza?

Es la práctica de transformar el código para hacerlo difícil de leer o entender, con el fin de protegerlo contra copias o manipulaciones no autorizadas.

9. ¿Cómo se llama el lenguaje que se menciona en el texto?

El lenguaje se llama JavaScript.

10. ¿Qué nombre le dio Microsoft a JavaScript?

Microsoft lo llamó JScript.

11. ¿Por qué Microsoft usó el nombre Jscript?

Porque el nombre JavaScript era una marca registrada de Netscape.

12. ¿Qué problema surgió con la versión de JScript de Microsoft?

No era totalmente compatible con JavaScript estándar, lo que ocasionó problemas de compatibilidad entre navegadores.

13. ¿Qué es el DOM y por qué fue diseñado?

El DOM (Document Object Model) es una interfaz que representa un documento HTML o XML como un árbol de objetos. Fue diseñado para que los lenguajes de script pudieran acceder, modificar y actualizar dinámicamente el contenido, estructura y estilo de las páginas web.

14. ¿Qué navegadores incorporaron el DOM desde su primera versión?

Internet Explorer, Netscape, Firefox, Chrome, Safari y Opera, aunque inicialmente con diferencias de implementación.

15. ¿Qué estándar regula el lenguaje JavaScript desde 1997?

El estándar es ECMAScript.

16. ¿Qué documento detalla las especificaciones del estándar ECMA?

El documento es ECMA-262.

17. ¿Qué relación tiene JavaScript con Java?

Son lenguajes diferentes. La relación se limitó a una estrategia de marketing, ya que se eligió el nombre JavaScript para aprovechar la popularidad de Java en ese momento.

18. ¿Quién desarrolló originalmente JavaScript y con qué nombre?

Fue desarrollado por Brendan Eich en Netscape. Primero se llamó Mocha, luego LiveScript y finalmente JavaScript.

19. Usa la función alert para enseñar el mensaje holamundo

```
<script>
alert("holamundo");
</script>
```

20. En el un archivo y declara varias variables en una sola linea

```
let nombre = "Ana", edad = 25, ciudad = "Madrid";
```

21. que caracteres y nombres están prohibidos como nombre de variables ?

Una variable no puede comenzar con un número, no puede contener espacios ni símbolos como - + * / . , excepto _ y \$ que sí se permiten.

22. Crea un ejemplo para cada tipos de datos (Crea una tabla en html5 en el documento htm)

Cadena	var cadena = "Hola Mundo";
Número	var numero = 123;
Boolean	var booleano = true;
Null	var nulo = null;
Object	var objeto = { nombre: "Ana", edad: 30 };
arrays	var array = [1,2,3,4];
Function	var funcion = function() { return "Hola"; };

23. Usa la función typeof para mostrar el tipo de los códigos

```
alert(cadena + " → " + typeof cadena);
alert(numero + " → " + typeof numero);
alert(booleano + " → " + typeof booleano);
alert(nulo + " → " + typeof nulo);
alert("{} + "nombre: Ana, edad:30}" + " → " + typeof objeto);
alert("[ " + array + " ] → " + typeof array);
alert("function(){} → " + typeof funcion);
```

24. Ejecuta estos datos y muestralos en un documento htm

```
var a=true;
var b=hola;
var c=1;
```

4+4	alert("4+4 = " + (4+4));
"4"+ 4	alert("'"4'+4 = " + ("4"+4));
a+c	alert("a+c = " + (a+c));
b+c	alert("b+c = " + (b+c));

2.9e3+1	<code>alert("2.9e3 + 1 = " + (2.9e3 + 1));</code>
2.9e3 + b	<code>alert("2.9e3 + b = " + (2.9e3 + b));</code>
2.9e3 + a	<code>alert("2.9e3 + a = " + (2.9e3 + a));</code>

25. Usa la función parseFloat y parseInt para convertir explícitamente los valores 2.9e3, 30.1e-3, 0,000000004 , 0,6, 3000000 , 301231000

```

alert("parseFloat(2.9e3) = " + parseFloat(2.9e3));
alert("parseFloat(30.1e-3) = " + parseFloat(30.1e-3));
alert("parseFloat(0.000000004) = " + parseFloat(0.000000004));
alert("parseFloat(0.6) = " + parseFloat(0.6));
alert("parseFloat(3000000) = " + parseFloat(3000000));
alert("parseFloat(301231000) = " + parseFloat(301231000));

```

```

alert("parseInt(2.9e3) = " + parseInt(2.9e3));
alert("parseInt(30.1e-3) = " + parseInt(30.1e-3));
alert("parseInt(0.000000004) = " + parseInt(0.000000004));
alert("parseInt(0.6) = " + parseInt(0.6));
alert("parseInt(3000000) = " + parseInt(3000000));
alert("parseInt(301231000) = " + parseInt(301231000));

```

26. Usa la función length para hallar el tamaño de las cadenas "123456" ,

```

var cadena = "123456";
alert("Longitud de '123456' = " + cadena.length);

```

27. Declara varias variables con const e intenta cambiarlas y visualizalas con un alert que pasa?

```

const pi = 3.14;
const nombre = "Ana";

```

```

alert("pi = " + pi);
alert("nombre = " + nombre);

```

```

// Intentos de cambio generan error
// pi = 3.1416;
// nombre = "Juan";
// Si descomentas, el navegador mostrará un error

```

28. Declara varias variables con var dentro de un bloque if al principio e intenta acceder a ellas para mostrarlas dentro del bloque y fuera del bloque con un alert que pasa?

```

if(true){
  var x = 10;

```



```
var y = 20;  
alert("Dentro del if: x = " + x + ", y = " + y);  
}
```

```
alert("Fuera del if: x = " + x + ", y = " + y);  
// Se pueden ver, porque var tiene alcance global o de función
```

29. Realiza el mismo ejercicio anterior pero con varias variables con let dentro de un bloque if . Se pueden ver el contenido de las variables.

```
if(true){  
  let a = 5;  
  let b = 15;  
  alert("Dentro del if: a = " + a + ", b = " + b);  
}
```

```
// alert("Fuera del if: a = " + a + ", b = " + b);  
// Error: a y b no están definidas fuera del bloque
```