

- 1) En JavaScript el **operador ===** se utiliza para:
- a) Comparar el valor.
  - b) Comparar el valor y el tipo de dato. ✓**
  - c) Compara el tipo de dato.
  - d) Ninguna de las respuestas es correcta.
- 2) La siguiente operación **`var x = 5 + "10"`**; da lugar a:
- a) Una suma y su resultado es 15.
  - b) Una suma y su valor es 105.
  - c) Una concatenación y se obtiene un string "105".
  - d) Ninguna de las respuestas es correcta. ✓**
- Concatena el 5 y el "10"  
Devuelve: 510
- 3) El método **`toFixed(2)`** de la variable **`var x = 9.656`**; es:
- a) `x.toFixed(2);` //aproxima a dos decimales.
  - b) `x.toFixed(2);` //devuelve 9.66. ✓**
  - c) `x.toFixed(2);` //devuelve un string.
  - d) Todas las respuestas son correctas.
- Redondea el valor al número de decimales indicado
- 4) ¿Cuál es la diferencia entre el **`for in`** y el **`for of`**?
- a) El `for in` recorre los valores de un array y el `for of` los índices.
  - b) El `for of` recorre los valores de un iterable y el `for in` los índices. ✓**
  - c) Ninguna de las opciones es correcta.
  - d) Son lo mismo.
- 5) ¿Qué instrucción es correcta para hacer **comentarios en JavaScript**?
- a) `ctrl + Ç`
  - b) `//` ✓**
  - c) `<!-- -->`
  - d) `\\`
- // :para comentarios de una sola línea  
/\* \*/ :Para comentarios de varias líneas
- 6) ¿Qué tipo de variables **permite hoisting**?
- a) `Var` ✓**
  - b) `let`
  - c) `const`
  - d) Todas son correctas.
- 7) ¿Cuál es la **diferencia** entre **`var`** y **`let`**?
- a) Sirven para lo mismo
  - b) `Let` se usa para limitar el ámbito al bloque actual ✓**
  - c) `Var` se usa para definir variables globales al archivo
  - d) Ninguna de las respuestas es correcta.

8) ¿Cuál será el resultado? `alert(1 + 1 + "1")`:

- a) 3
- b) 2

**c) 21 ✓**

d) Ninguna de las respuestas es correcta.

Suma los dos primeros valores y le concatena el segundo

9) ¿Qué lenguaje basado en javascript creó Microsoft?:

a) typescript

**b) jscript ✓**

c) newscript

d) Ninguna de las respuestas es correcta.

10) ¿Cuál de las siguientes sentencias NO devuelve 'object'?

**a) `var a = 1; typeof(a)` ✓**

b) `var a = null; typeof(a)`

c) `var a = ['pepe', true]; typeof(a)`

d) `var a = {1 : 'a', 2 : 'b'}; typeof(a)`

11) ¿Qué tipo de dato devuelve una entrada tipo prompt?

a) Depende de lo que indique el usuario.

**b) string. ✓**

c) undefined.

d) number.

12) El método "`trim()`" cuando se aplica a una cadena...

a) Elimina los espacios en blanco al principio de una cadena.

b) Ninguna respuesta es correcta.

c) Elimina los espacios al final de una cadena.

**d) Elimina espacios en blanco al principio y al final de una cadena. ✓**

13) La función "`parseInt()`" convierte que se le indica a...

a) Un número decimal.

b) Todas son correctas.

**c) Un número entero. ✓**

d) Ninguna es correcta.

14) ¿De qué tipo de lenguaje de programación es JavaScript según su forma de ejecución?

**a) Interpretado. ✓**

b) Dinámico.

c) Compilado.

d) Funcional.

15) ¿Qué formas tiene JavaScript para embeber el código?

- a) Las dos son correctas. ✓
- b) En la etiqueta head o body mediante la etiqueta <script>.
- c) Utilizando el atributo src de la etiqueta <script>.
- d) Ninguna son correctas

16) ¿Qué importancia tiene el punto y coma al final de cada instrucción?

- a) Es obligatorio después de cada instrucción.
- b) No es obligatorio, pero es buena práctica de programación. ✓
- c) No es obligatorio y tampoco es buena práctica.
- d) Ninguna opción es correcta

17) ¿Cómo se puede cargar asincrónicamente un script?

- a) <script src="noseque.js" defer> </scrip>
- b) <script src="noseque.js" async> </scrip> ✓
- c) <script src="noseque.js" async> </scrip>
- d) Ninguna es correcta.

18) La variable var...

- a) declara una variable de scope global o local para la función sin importar el ámbito de bloque. Permite hoisting. ✓
- b) Declara una variable de scope global, local para la función o de bloque. Es reassignable y no permite hoisting.
- c) Declara una variable de scope global, local para la función o de bloque. No es reassignable, pero es mutable. No permite hoisting.
- d) Todas son correctas.

19) En Javascript, las variables que son null y undefined, ¿se consideran iguales?

- a) Verdadero. ✓
  - b) Falso.
- `null == undefined // true`  
`null === undefined // false`

20) ¿Cuáles son las estructuras de selección?

- a) If, switch y while.
  - b) While, for e if.
  - c) If, switch, while, for.
  - d) Todas las respuestas son incorrectas. ✓
- Selección: if y switch  
Iterativa: for y while

21) El primer carácter de una variable debe ser:

- a) Una letra.
  - b) Un símbolo de dólar.
  - c) Un guión bajo.
  - d) Todas son correctas. ✓
- Pero nunca un número ✗

22) ¿Cuál de estos es un tipo primitivo del JS?

a) int

**b) null ✓**

c) Ambas son incorrectas.

d) Ambas son correctas.

undefined

null

boolean

number

string



23) ¿Cómo se puede asignar un valor a una variable?

a) Asignación directa de un valor concreto.

b) Indirecta a través de un cálculo que se implica a otras variables o constantes.

c) A través de una solicitud del valor al usuario del programa (prompt)

**d) Todas son correctas. ✓**

24) Para declarar una variable local limitando su alcance al bloque, declaración o expresión donde se está usando, utilizamos:

a) var

**b) let ✓**

c) const

d) Ninguna es correcta.

25) Con la ventana "alert" se puede mostrar:

a) Texto

b) Variables

c) Texto + variables

**d) Todas son correctas ✓**

26) Cuando se produce una colisión entre las variables globales y locales, dentro de una función prevalecen:

**a) Las variables locales. ✓**

b) Las variables globales.

c) Depende.

d) Ninguna es correcta.

27) ¿Qué significa que JavaScript es de tipado débil y dinámico?

a) Se puede asignar las variables en la ejecución.

b) Las variables no tienen tipo hasta que son inicializadas

c) Siempre que pueda, el compilador intentará hacer la operación y devolver algo.

**d) Todas son correctas ✓**

28) ¿Cómo indicas una **operación ternaria**?

- a) Condición ¿ expr1 : expr2
- b) Condición ¿ expr1 : expr2
- c) Condición : expr1 ¿ expr2
- d) Condición ? expr1 : expr2 ✓**

29) ¿Qué **resultado** dará esta operación? `const x = 1; y = 3 ; z = 6; x = ( y + z ) * ( z - y );`

- a) 27
- b) 18
- c) 15

**d) Ninguna es correcta. ✓**

Ojo: es una constante y por ello no se le puede reasignar un valor ni alterar el que ya tiene

30) ¿Qué **identificador** es **correcto**?

- a) \$numero
- b) \_texto
- c) Variable

**d) Todas son correctas. ✓**

31) ¿Qué hace el **hoisting**?

- a) Mueve las inicializaciones al principio del ámbito.
- b) Mueve las declaraciones al principio del ámbito. ✓**
- c) Permite recorrer un array.
- d) Ninguna es correcta.

32) ¿Cómo se puede mirar la **longitud de una cadena**?

- a) cadena.size
- b) cadena.size()
- c) cadena.length ✓**
- d) cadena.length()

33) **Completa la frase. JavaScript...:**

- a) Está basado en programación funcional y no admite la herencia basada en prototipos.
- b) Admite la herencia basada en variables globales, siempre que se declare sin usar la palabra reservada var.
- c) Está basado en el concepto de objeto, pero no es un lenguaje orientado a objetos. Sus objetos utilizan herencia basada en prototipos. ✓**
- d) Está basado en el concepto de estructura de arrays prototipados de forma que todos los navegadores soportan el uso de estructuras asociativas para interpretar de forma más dinámica la web recibida.

- 34) Para mostrar una salida directamente dentro del documento HTML:
- a) Se utiliza la sentencia `document.write("texto");` y es la opción más recomendable.
  - b) Se utiliza la sentencia `document.getElementById("id").innerHTML` y es la opción más recomendable.
  - c) Se utiliza la sentencia `doctype.write("texto");` pero su uso se recomienda solo para pruebas.
  - d) Se utiliza la sentencia `document.getElementById("id").innerHTML` y es la opción más recomendable. ✓
- 35) ¿Qué tipos de salida por consola existen en Javascript?
- a) `console.log()`, `console.info()`, `console.warn()` y `console.error()`
  - b) Ninguna respuesta es correcta. ✓
  - c) `console.warning()`, `console.log()`, `console.info()` y `console.error()`
  - d) `console.log()`, `console.info()`, `console.warn()` y `console.error()`
- 36) ¿Cuál fue el primer nombre que tuvo JavaScript?
- a) LiveScript.
  - b) Siempre se llamó JavaScript.
  - c) Mocha. ✓
  - d) Ninguna de las respuestas es correcta.
- 37) ¿Qué valor daría de resultado "true" al convertirlo a booleano?
- a) "hola"
  - b) 1
  - c) [1, 2, 3]
  - d) Todas son correctas. ✓
- 38) ¿Qué tipo de operador es "&&"?
- a) Aritmético.
  - b) Lógico. ✓
  - c) Condicional.
  - d) De comparación.
- 39) ¿Cuál de las siguientes variables está mal declarada según las reglas de estilo?
- a) `var _aux;`
  - b) `var $aux;`
  - c) `var 2aux;` ✓
  - d) `var aux;`

40) ¿Cuál es la forma más correcta de definir constantes en javascript ?

a) Con const "nombre" ✓

b) con var "nombre"

c) sin var sólo el "nombre"

d) Cualquiera siempre y cuando esté definida en el ámbito global

Por regla de estilo el nombre de una constante debe ir en mayúsculas

41) ¿Qué pinta este script por pantalla?

```
<script>
  let numeros = 5
  for (let i = 0; i < numeros; i++) {
    document.write(i);
    i++;
    i++;
    document.write(i);
  }
</script>
```

Realiza dos saltos en la iteración

a) 0235 ✓

b) Ninguna de las anteriores

c) 035

d) 023undefined

42) Según las reglas de sintaxis los literales se:

a) Se escriben con comillas dobles.

b) Se escriben con comillas simples.

c) Se escriben con comillas simples o dobles. ✓

d) Las cadenas de un carácter van en comillas simples y las demás en comillas.

43) ¿Qué veremos en pantalla?

```
var resultado = !(true && false) && (false || true);
resultado ? document.write("Hola") : alert("Adios");
```

!(true && false) = true  
(false || true) = true

true && true = true

a) Se escribirá en la web "Hola" ✓

b) Veremos una alerta "Hol"

c) Se escribirá en la web "Adios"

d) Veremos una alerta de "Adios"

44) ¿Qué diferencia hay entre .indexOf( ) y .charAt( )?

a) Ninguna es correcta.

b) charat( ) devuelve un carácter mientras que indexOf( ) devuelve el índice. ✓

c) indexOf( ) devuelve un carácter mientras que charat( ) devuelve el índice.

d) Todas son correctas.

charAt() recibe un entero y devuelve el carácter del índice indicado  
indexOf() recibe una cadena y devuelve el índice del primer carácter de la cadena

45) ¿Qué muestra por consola la siguiente función?

```
var pi = 3.14;  
function numero_pi() {  
  console.log(pi[0]);  
}  
numero_pi()
```

a) 3.14

b) 3

c) undefined ✓

d) null

No se puede indexar en una variable de tipo number

46) ¿Qué muestra por consola la siguiente función?

```
var txt = "texto";  
function texto() {  
  console.log(`${texto[0]}`);  
}  
texto()
```

a) "texto"

b) "t"

c) undefined

d) Ninguna es correcta. ✓

En las plantillas de cadena se `` ✓

NO " " ✗

47) ¿Qué muestra por consola la siguiente función?

```
var diez = (10 === parseInt("10.0"));  
function estricto() {  
  console.log(diez);  
}  
estricto();
```

Realiza una conversión a entero, int

a) true ✓

b) false

c) Da error.

d) Ninguna es correcta.

48) ¿Cuál de estos identificadores es válido en JavaScript?

a) var Numero;

b) var \$numero;

c) Ninguna es correcta.

d) Ambas son correctas. ✓



49) JavaScript es:

- a) Orientado a objetos.
- b) Débilmente tipado. ✓**
- c) No tiene intérprete
- d) Su sintaxis se asemeja a otros lenguajes como Python o Ruby.

50) ¿Qué se mostrará por pantalla después de la ejecución de este código?

```
var numeros = [1, 2, 3, 4];

function muestraNumero() {
    var numeros = [1, 2, 3, 4, 5];

    for (var i = 0; i < 5; i++) {
        if ({numeros[i] % 2} == 0) {
            alert(numeros[i]);
        }
    }
}

alert(numeros);
```

La función no es llamada por lo que no se dispara el contenido de la misma

El alert que se dispara no entra dentro del ámbito de la función

- a) undefined
- b) Mostrará dos alerts con los números 2 y 4.
- c) Mostrará un alert con 1, 2, 3, 4. ✓**
- d) Dará un error y no mostrará nada.