Funciones con parámetros

Nombre: _____

Fecha: _____

Puntaje: _____

1.

¿Qué es un parámetro en una función?

1. Un valor que se pasa a la función al llamarla
2. Una variable definida en la declaración de la función que recibe un valor
3. El nombre de la función
4. El tipo de dato que retorna la función

2.

En la función function saludar(nombre), ¿qué es nombre?

1. Un argumento
2. Una variable global
3. Un parámetro
4. Un tipo de dato

3.

¿Cuál es la sintaxis correcta para definir una función con un parámetro en JavaScript?

1. function suma() { }
2. function suma[x] { }
3. function suma(x) { }

4.

El valor que se pasa a una función cuando se la invoca se denomina ____.

Respuesta: _____

5.

¿Qué imprime el siguiente código: function saludo(nombre) { console.log('Hola, ' + nombre); } saludo('María');?

1. Hola, nombre
2. Hola, saludo
3. Error
4. Hola, María

6.

¿Qué sucede si defines una función con un parámetro pero la llamas sin pasar ningún argumento?

1. Ocurre un error
2. El parámetro toma el valor undefined
3. El parámetro toma el valor null
4. La función no se ejecuta

7.

En la definición de una función, los nombres de los parámetros van entre ____.

Respuesta: _____

8.

¿Qué hace la siguiente función? function doble(x) { return x * 2; }

1. Devuelve el doble del argumento
2. Devuelve el argumento más 2
3. Devuelve el cuadrado del argumento
4. Imprime el doble

9.

¿Cómo se llama al acto de proporcionar un valor al parámetro durante la invocación?

1. Declarar
2. Definir parámetro
3. Pasar un argumento
4. Retornar

10.

¿Cuál de las siguientes NO es una ventaja de usar parámetros en funciones?

1. Permite reutilizar la función con diferentes valores
2. Hace que la función sea más flexible
3. Reduce la necesidad de variables globales
4. Aumenta la complejidad innecesariamente

11.

Si una función tiene dos parámetros, estos deben separarse por ____.

Respuesta: _____

12.

Observa el código: function suma(a, b) { return a + b; } let resultado = suma(3, 7); ¿Qué valor tiene resultado?

1. 37
2. 10
3. a+b
4. undefined

13.

¿Qué concepto corresponde a la variable que aparece en la definición de la función?

1. Parámetro formal
2. Argumento
3. Variable local
4. Parámetro real

14.

En la expresión miFuncion('rojo'), ¿qué es 'rojo'?

1. Un parámetro
2. Una variable
3. Un argumento
4. Un retorno

15.

¿Qué imprime el siguiente código? function mostrar(mensaje) { console.log(mensaje.toUpperCase()); } mostrar('hola');

1. hola
2. mensaje
3. toUpperCase
4. HOLA

16.

Completa la declaración de la función multiplicar que recibe un parámetro y retorna su cuadrado: function multiplicar(_____) { return x * x; }

Respuesta: _____

17.

¿Cuál de las siguientes funciones no utiliza correctamente un parámetro?

1. function f(z) { return; }
2. function f(x) { return x + 1; }
3. function f(y) { console.log(y); }
4. function f(w) { w = 5; return w; }

18.

¿Cuál es el resultado de ejecutar function foo(a, b) { return a - b; } foo(10, 4);?

1. 14
2. -6
3. 6
4. 7

19.

¿Qué diferencia hay entre parámetro y argumento?

1. Son sinónimos
2. Parámetro es el valor real; argumento es la variable
3. Parámetro es la variable en la definición; argumento es el valor en la llamada
4. No hay diferencia en programación

20.

En una función que saluda a una persona, ¿cómo se podría personalizar el saludo para diferentes nombres?

1. Definiendo una función sin parámetros y usando una variable global
2. Definiendo una función con un parámetro que recibe el nombre
3. Usando múltiples funciones idénticas para cada nombre
4. No es posible personalizarlo