



instantiva

C O N • S E N T I D O

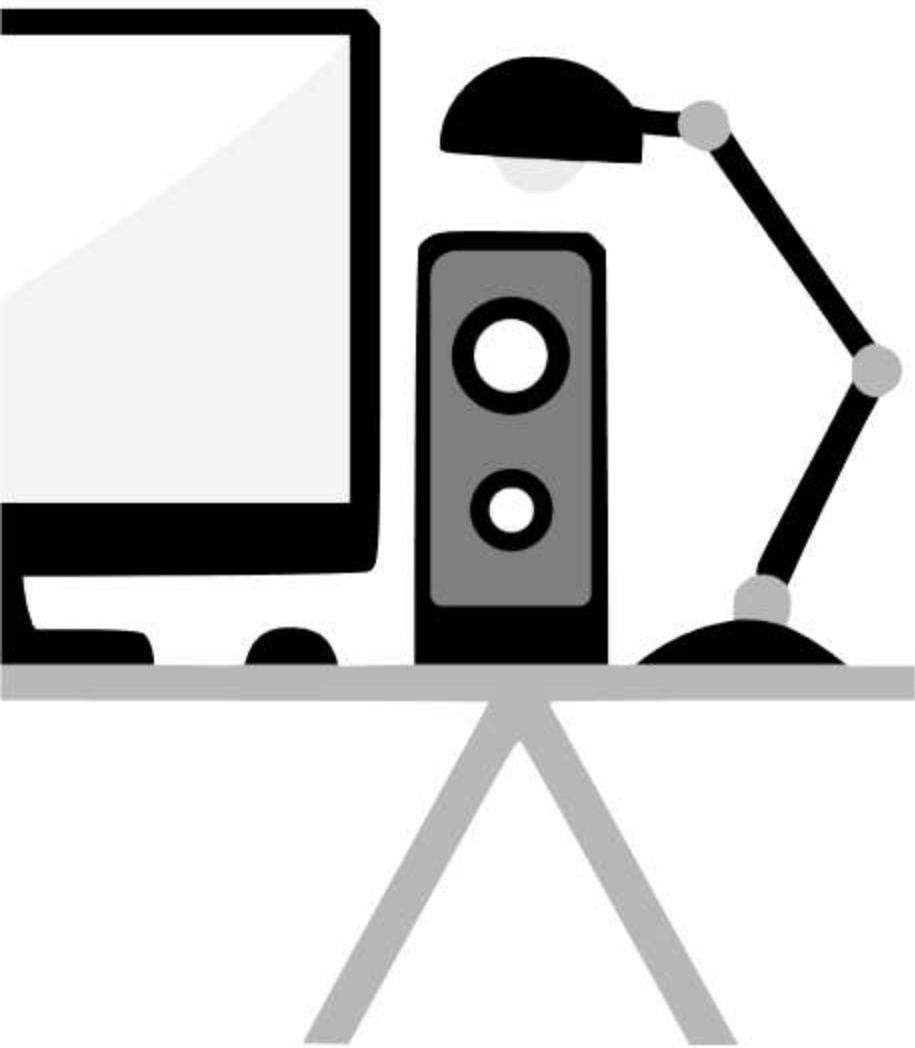
FUNCIONES

Lección 04

Competencia General

Codificar un programa utilizando funciones para la reutilización de código acorde al lenguaje JavaScript.






Funciones en JS

1. ¿Qué es una función? Y para qué sirve
2. Definir funciones
3. Paso de parámetros en una función
4. Retorno de una función
5. Variables locales y globales en funciones
6. Invocación de una función
7. Alcance de variables en funciones
8. El problema de las variables globales
9. Funciones anidadas

¿Qué es una función?

Una función es una secuencia de declaraciones. Esta se conforma por el cuerpo de la función, los parámetros de entrada y un retorno que puede devolver un valor.

La finalidad de una función es evitar repetir una secuencia en el código, resumiendo dicha secuencia a una invocación.



```
function nombre_funcion(parametros){  
  instrucciones;  
}
```

Definir una función

Para definir una función debemos:

1. Recurrir a la palabra reservada ***function***
2. Declarar el nombre de dicha función
3. Incluir los parámetros de entrada
4. Establecer las instrucciones o algoritmos que desarrolla dicha función.
5. Retornar el valor usando la palabra reservada ***return***

```
// Ejemplo de una función
function suma (a,b){
    return a + b;
}

console.log(suma(1,2)); // devuelve 3
```

Paso de parámetros a una función

Una función opcionalmente puede recibir una lista de parámetros. Estos pueden ser definidos desde su llamada o estableciendo un valor por defecto.

Si se establece un valor por defecto, este se asignará solo si no es introducido en los parámetros una vez llamada la función.

```
// Ejemplo de una función
function sumaTres (a, b = 3){
    return a + b;
}

console.log(sumaTres(1)); // devuelve 4
```

Retorno de una función

Para retornar un valor desde una función debemos utilizar la palabra reservada ***return***. Este valor podrá ser almacenado en una nueva variable mediante una asignación.

```
// Ejemplo de una función
function suma (a,b = 3){
    return a + b;
}

let resultado = suma(1,2);
console.log(resultado); // devuelve 3
```


Variables locales y globales

Dentro de una función podemos declarar variables, pero estas solo podrían ser utilizadas en el interior de la función, por el contrario una variable declarada fuera, es de ámbito global.

```
let num1 = 10

function multiplicaPorDiez (a){
  return a * num1;
}

let resultado = multiplicaPorDiez(3);
console.log(resultado); // devuelve 30
```


Invocación de una función

El solo hecho de definir una función no la ejecuta. Definir simplemente nombre y especifica que hace una función.

Para invocar una función debemos escribir el nombre de la función que creamos y ingresar los parámetros requeridos

```
function saludar (nombre){  
    return "Hola " + nombre;  
}  
  
console.log(saludar("Raul"));
```

Alcance de variable

Las variables definidas al interior de una función no pueden ser llamadas desde fuera dado que su ámbito está definido en la función.

```
function puntuacion(nombre) {  
  let num1 = 2;  
  let num2 = 3;  
  return nombre + ' anotó ' + (num1 + num2);  
}  
  
console.log(puntuacion("Patricio"));  
// devuelve Patricio anotó 5
```

El problema de las variables globales

El problema con las variables globales es que, dado que cada función tiene acceso a ellas, se hace difícil determinar qué función modifica o lee el valor y cuál será este valor.

Es por ello que si necesita mantener un orden y control de los valores en variables no se usen variables globales

```
var a = 100;  
function test(){  
    console.log(a);  
    var a = 10;  
    console.log(a);  
}
```

```
test();  
console.log(a);
```

```
//la ejecución de este código  
resulta en undefined 10 100
```

Funciones anidadas

Una función puede contener otra función en su declaración, esta función interna “hereda” los argumentos y variables de la función contenedora.

Una función interna incluye sus propias instrucciones y las declaraciones que se realicen en dicha función solo se pueden acceder en la función interna.

```
function puntuacion(nombre) {  
  let num1 = 2;  
  let num2 = 3;  
  function imprimir(nombre){  
    return nombre + ' anotó ' + (num1 + num2);  
  }  
  return imprimir(nombre);  
}  
  
console.log(puntuacion("Patricio"));  
// devuelve Patricio anotó 5
```


Ejercicio práctico

Realiza una función que reciba por parámetro un monto y un descuento. Realiza los cálculos para agregar IVA y aplicar el descuento al monto y retorna el valor resultante.





substantiva

C O N • S E N T I D O