



Funciones





**Activen las cámaras los que puedan y
pasemos asistencia**

***Utiliza
Javascript
para obtener y
modificar
elementos del
DOM a partir
de un
requerimiento***

- Unidad 1: Introducción a JavaScript.
- Unidad 2: Condiciones.
- Unidad 3: Funciones.
- Unidad 4: Arreglos y objetos.
- Unidad 5: Métodos de arreglos.
- Unidad 6: APIs



Te encuentras aquí





Inicio

{desafío}
latam_



/ Crear y llamar funciones */*

/ Crear funciones que reciban parámetros */*

/ Llamar funciones desde el onclick de un elemento */*

/ Crear funciones que devuelvan valores */*

Objetivos

Activación de conceptos

Contesta la pregunta correctamente y gana un punto

Instrucciones:

- Se realizará una pregunta, el primero en escribir “YO” por el chat, dará su respuesta al resto de la clase.
- El docente validará la respuesta.
- En caso de que no sea correcta, dará la oportunidad a la segunda persona que dijo “Yo”.
- Cada estudiante podrá participar un máximo de 2 veces.
- Al final, el/la docente indicará el 1º, 2º y 3º lugar.



¿Qué se obtiene al
comparar `2 === '2'` ?

¿Qué se obtiene al
comparar `true || false`?

¿Qué se obtiene al
comparar `true && false`?

¿Qué se obtiene al
comparar `true && false ==
false`? ¿En qué orden se
resuelve?

Activación de conceptos



Primer lugar:



Segundo lugar:



Tercer lugar:



Desarrollo

{desafío}
latam_



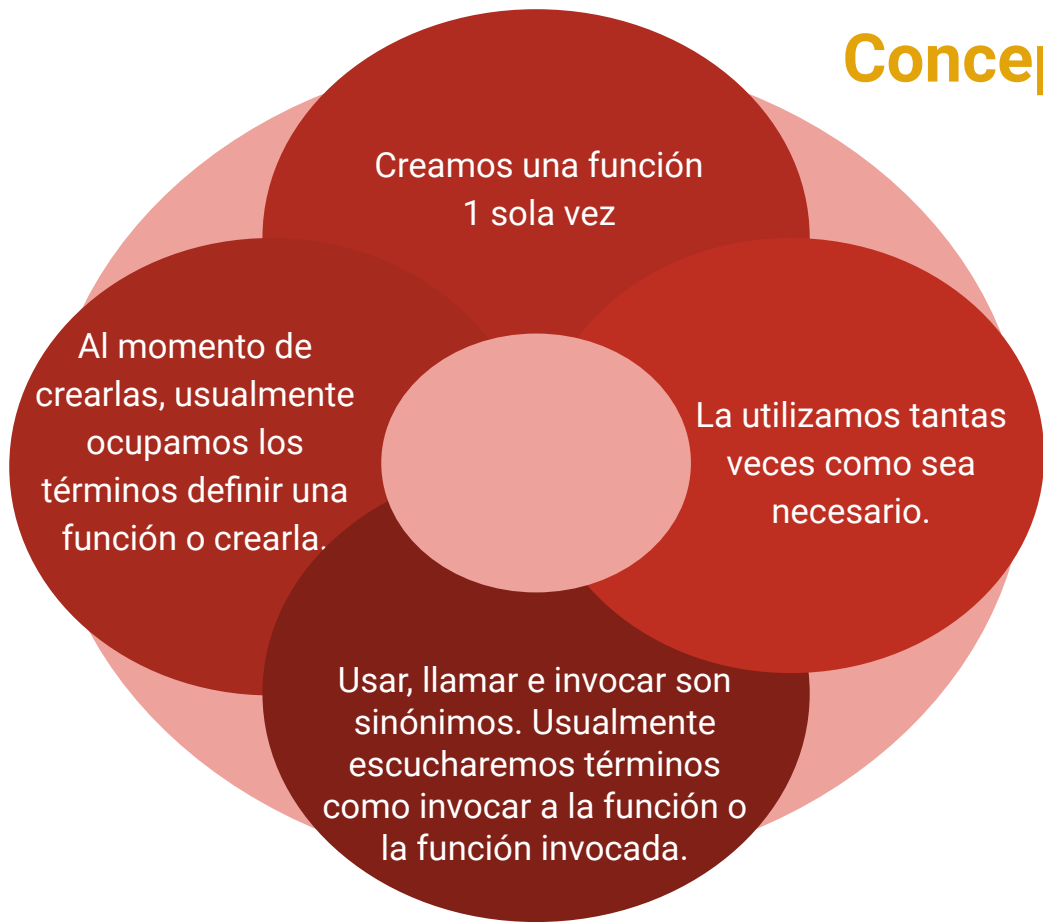
/* Introducción a funciones */

¿Qué es una función?

- Las funciones son conjuntos de instrucciones que podemos programar una vez y utilizarlas cada vez que necesitemos.
- Las funciones tenemos que crearlas y luego usarlas (llamarlas)

```
function aprenderHaciendo() {  
  // ¡Manos al código!  
}
```

Conceptos básicos



Creando nuestra primera función

```
/* Creamos la función */  
function pintar_negro(){  
    elemento = document.querySelector("body")  
    elemento.style.backgroundColor = "black"  
}  
  
/* Llamamos a la función */  
pintar_negro();
```

Creando nuestra primera función

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8" />
5     <meta http-equiv="X-UA-Compatible" content="IE=edge" />
6     <meta name="viewport" content="width=device-width, initial-scale=1.0" />
7     <title>Document</title>
8   </head>
9   <body>
10    <script>
11      /* Creamos la función */
12      function pintar_negro() {
13        elemento = document.querySelector("body");
14        elemento.style.backgroundColor = "black";
15      }
16
17      /* Llamamos a la función */
18      pintar_negro();
19    </script>
20  </body>
21 </html>
22
```

Ejercicio

Dentro de la página web, crear la función `pintar_rojo` y la función `pintar_amarillo` y luego llamarlas.

Ejercicio

¡Manos al teclado!



*/** Crear y llamar funciones **/* ✓

*/** Crear funciones que reciban parámetros **/*

*/** Llamar funciones desde el onclick de un elemento **/*

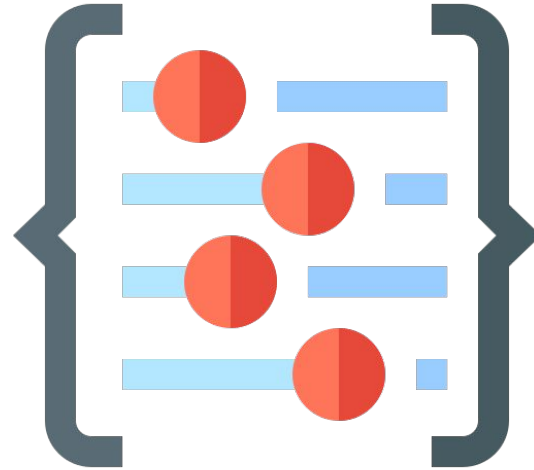
*/** Crear funciones que devuelvan valores **/*

Objetivos

/* Funciones y parámetros */

¿Qué son los parámetros?

- Los parámetros son valores que puede recibir una función que nos permite hacerla mucho más flexible.
- Por ejemplo podemos modificar la función pintar que creamos previamente para que reciba un color y luego utilizar este color para pintar la página.



Ejemplo de uso

```
pintar = function(color){  
  elemento = document.querySelector("body")  
  elemento.style.backgroundColor = color  
}  
  
pintar("black");  
pintar("red");  
pintar("yellow");
```

Parámetros y argumentos

```
pintar = function(color){  
  elemento = document.querySelector("body")  
  elemento.style.backgroundColor = color  
}  
  
pintar("black")
```

2 Al ejecutar la función color toma el valor de "black"

1. Llamamos a la función con el argumento "black"

Funciones y parámetros

Funciones con múltiples parámetros

Las funciones pueden tener más de un parámetro, para lograrlo simplemente tenemos que separarlos con coma al momento de definir la función.

```
funcionMultiplesParametros(par1, par2, par3)
```

Al llamar a la función tenemos que pasar los valores en el mismo orden que están definidos los parámetros.

```
funcionMultiplesParametros("azul", "#id-2", 5)
```

¿Qué se muestra en la consola del inspector de elementos?

```
funcionMultiplesParametros(par1, par2,  
par3) {  
  console.log(par2)  
}
```

```
funcionMultiplesParametros(3, 2, 1)
```

¡Manos a la obra!



Funciones y parámetros

Parámetros con valores por defecto

```
pintar = function(color = "black"){  
  elemento = document.querySelector("body")  
  elemento.style.backgroundColor = color  
}  
  
pintar();  
pintar("red");  
pintar("yellow");
```


¿Qué se muestra en la consola del inspector de elementos?

```
funcionMultiplesParametros(par1, par2 = 2, par3 = 2 )  
{  
  console.log(par2 + par3)  
}  
  
funcionMultiplesParametros(3)
```

¡Manos a la obra!



*/** Crear y llamar funciones **/* 

*/** Crear funciones que reciban parámetros **/* 

*/** Llamar funciones desde el onclick de un elemento **/*

*/** Crear funciones que devuelvan valores **/*

Objetivos

/* Funciones y DOM */

Funciones

Dentro de onclick llamaremos a la función, la función debe estar definida dentro de script. El script puede estar en el mismo archivo o en otro.

```
<button onclick="pintar('black')"> Pintar negro</button>
<script>
  function pintar(color) {
    elemento = document.querySelector("body")
    elemento.style.backgroundColor = color
  }
</script>
```

Reutilizando funciones

Ahora que onclick llama a la función, se hace sencillo reutilizar la lógica.

```
<button onclick="pintar('black')"> Pintar negro</button>
<button onclick="pintar('red')"> Pintar rojo</button>
<button onclick="pintar('green')"> Pintar verde</button>
<script>
  function pintar(color) {
    elemento = document.querySelector("body");
    elemento.style.backgroundColor = color;
  }
</script>
```

Ejercicio guiado

En una página nueva pondremos 3 imágenes, al hacer click en una de ellas le agregaremos bordes.

Demostración



Solución Ejercicio Guiado

```




<script>
  /* Creamos la función */
  function agregarBordes(elementId) {
    elemento = document.querySelector('#' + elementId);
    elemento.style.border="dashed 3px brown"
  }
</script>
```

Ejercicio

- Modifica la página anterior para que la función adicionalmente al elemento pueda recibir un color.
- Especificar distintos colores al llamar a la función en el onclick.

Ejercicio

¡Manos al teclado!



*/** Crear y llamar funciones **/* ✓

*/** Crear funciones que reciban parámetros **/* ✓

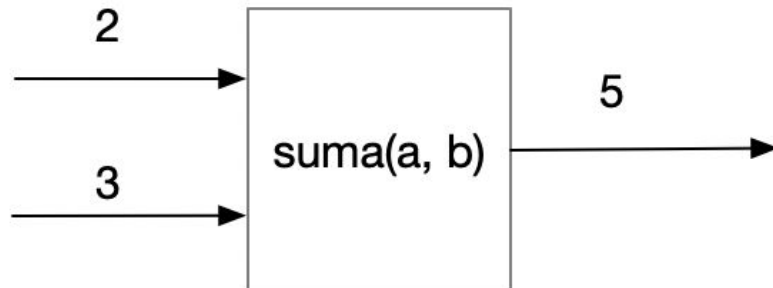
*/** Llamar funciones desde el onclick de un elemento **/* ✓

*/** Crear funciones que devuelvan valores **/*

Objetivos

/* El retorno de las funciones */

Valor y funciones



Las funciones reciben valores y pueden además devolver valores

```
function suma(a, b){  
  return a + b  
}
```

```
alert(suma(2,3)) /* Opción 1 */  
resultado = suma(2,3) /* Opción 2 */  
console.log(resultado)
```

Devolver el valor de funciones

```
» function suma (a, b){  
  alert (a + b)  
}  
suma(2, 3)  
← undefined
```

```
function suma (a, b) {  
  return a + b  
}  
suma(2, 2)
```

No todas las funciones devuelven un valor, sino solo aquellas que tienen return

Ejercicio

Nos piden crear una función llamada `getBkgColor` para obtener el color de fondo de un elemento web. O sea debe buscar un elemento a partir de un selector (`#id` o `.class`) entregado y debe devolver el color de fondo del elemento.

- ¿Cuál es/son los parámetros de entrada de la función?
- ¿Qué valor deberíamos devolver?
- Ahora creamos la función

Ejercicio

¡Manos al teclado!



Solución Ejercicio

```
<div id="el-1" class="element" style="background-color: red"></div>
<div id="el-2" class="element" style="background-color: yellow"></div>
<script>
  function getBkgColor(selector){
    ele = document.querySelector(selector)
    return ele.style.backgroundColor
  }
  getBkgColor("#el-1") /* Probamos nuestra función */
</script>
```

Ejercicio

Nos piden crear una función `getValue` que obtenga el valor de un input a partir de un selector y devuelva un texto que diga "mucho" si el valor es mayor que un parámetro, "exacto" si el valor es igual al parámetro o "muy poco" si el valor es menor al parámetro.

Antes de hacer el ejercicio discutir:

- ¿Cuál/es es/son los parámetros de entrada de la función?
- ¿Qué valor deberíamos devolver?

Ejercicio

¡Manos al teclado!



`/* Crear y llamar funciones */` ✓

`/* Crear funciones que reciban parámetros */` ✓

`/* Llamar funciones desde el onclick de un elemento */` ✓

`/* Crear funciones que devuelvan valores */` ✓

Objetivos



Cierre

{desafío}
latam_



¿Existe algún concepto que no
hayas comprendido?
¡Revisémoslo antes de
terminar la clase!

Reflexionemos



- Revisar la guía que trabajarán de forma autónoma.
- Revisar en conjunto el desafío.

¿Qué sigue?



*Academia de
talentos digitales*

www.desafiolatam.com



/DesafioLatam



/DesafioLatam



/DesafioLatam



/DesafioLatam