



Argentina
programa
4.0

.UBAfiuba 
FACULTAD DE INGENIERÍA

JavaScript

Clase 3 – Introducción a Declaración de Funciones

Bianchi – Marianetti

Febrero 2023

Contenido

- ¿Qué son las funciones?
- Objetivos y beneficios de usar funciones
- ¿Cómo hacemos para declararlas?
- Invocación de funciones
- Parámetros de las funciones
- Retorno de valores

¿Qué son las funciones?

Las funciones son **bloques de código que realizan alguna operación.**

- La idea principal detrás de las funciones es **evitar repetir código**, y en consecuencia **hacerlo reutilizable**. Además, hacer **que este sea más claro** para los desarrolladores.
- Las operaciones que realiza una función pueden ser variadas y tan simples como realizar una suma o tan complejo como obtener información de una base de datos.

Objetivos y beneficios de usar funciones

→ Facilita mantener el código

Hay que tomar en cuenta que el código **se lee más veces de las que se escribe**, por lo tanto hacer que el código sea más entendible hace que sea un mejor código. **Un código claro es más fácil de mantener.**

→ Evita repeticiones innecesarias de código

En todo código va a haber secciones que utilicen código creado en otra parte, las funciones evitan que haya mucho código duplicado y se vuelva complejo de entender.

→ Simplifican la construcción del código

Cuanto más crece un código más difícil es trabajar con este, las funciones hacen que sea más simple organizarlo en el archivo y en nuestro pensamiento.

¿Cómo hacemos para declararlas?

SINTAXIS

```
function nombre_funcion( parámetros ) {  
    // instrucciones  
    return valor_a_retornar;  
}
```

function

- Palabra reservada con la que comienza la declaración de una función

nombre_función

- Nombre dado a la función que luego se usará para invocarla

parámetros

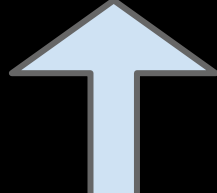
- La función puede recibir valores para utilizarlos
- Si es más de un valor, deben estar separados por “,”

return

- Indica el fin de la función (opcional)
- Puede devolver un valor o ninguno

EJEMPLO

```
<html>  
  <head>  
    <script>  
      function suma (sumando_1, sumando_2) {  
        const total = sumando_1 + sumando_2;  
        return total;  
      }  
    </script>  
  </head>  
  . . . . .
```



En este ejemplo, se declara una función llamada suma, que recibe 2 parámetros (2 valores), que serán sumados dentro de la función, y devolverá el total de la suma.

En este caso, la declaración de esta función se hace dentro de la misma página html y dentro del elemento <head>

¿Cómo hacemos para invocarlas?

Para utilizar una función que hayamos declarado, la tenemos que invocar mediante el nombre que le dimos.

También, si en la declaración se indicó que la función debe recibir parámetros, debemos pasarle valores para los parámetros

EJEMPLO

```
<html>
  <head>
    <script>
      function suma (sumando_1, sumando_2) {
        const total = sumando_1 + sumando_2;
        return total;
      }
    </script>
  </head>
  <body>
    <script>
      let total_suma = suma( 5, 10);
      alert("El total de la suma es: " + total_suma);
    </script>
  </body>
</html>
```

Invocamos a la función y le pasamos los valores que queremos que sume; y recibimos el total de la suma en la variable total_suma

EJERCICIO

Toma el código de este ejemplo y probalo.

Luego, modificalo para que en lugar de pasarle a la función suma, los valores 5 y 10, le pases dos valores ingresados por el internauta, solicitados mediante el uso del método prompt().

Recordá que lo que devuelve el prompt, siempre será de tipo string (cadena de caracteres); por lo tanto para convertirlo a número deberás usar parseInt.

Ejercicios

1. Tomá el ejercicio ejemplo y recortá la función suma y colocala en un archivo .js y vincula el archivo .js a tu página html, para que pueda seguir usando la función.
2. Agregá las funciones resta, multiplicación y división al archivo .js; y modificá el código JavaScript de la página html, para que en lugar de mostrar mediante un alert el total de la suma; muestre en el cuerpo de la página el resultado de la suma, la resta, la multiplicación y la división de los valores ingresados, con los respectivos mensajes aclaratorios.

Recordar: podemos usar `document.write` para mostrar lo que queramos, en la página

Resumen

- ✓ Una función es un bloque de código independiente que podemos invocar cuando la necesitamos
- ✓ Pueden recibir valores a través de parámetros.
- ✓ Puede devolver un valor o ninguno.

```
function nombre_funcion( parámetros ) {  
    // instrucciones  
    return valor_a_retornar;  
}
```

www.ingenieria.uba.ar

f    /ingenieriauba

 /FIUBAoficial