

Bootcamp Java Developer

Fase 2 - Java Web Developer
Módulo 18

Funciones

Funciones

Las funciones son bloques de código que tienen la característica **de ser llamados o invocados**. La siguiente sería la sintaxis de una **función** que, **por ejemplo, contiene una alerta**:

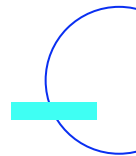
```
1  function miFuncion()  
2  {  
3      console.log('Hola estoy dentro de una función');  
4  }  
5
```



Si ese código se levanta en el navegador, básicamente no se obtiene nada, porque la función no ha sido invocada.

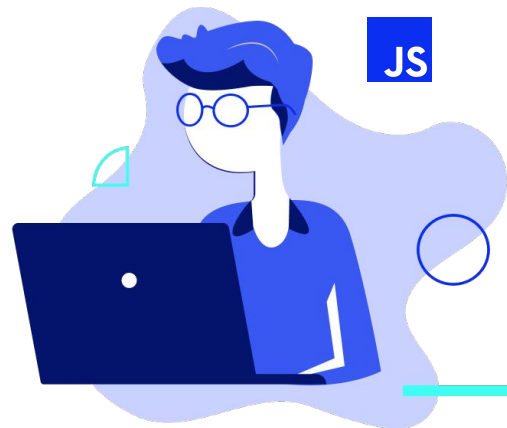
Para invocar una función debemos hacer lo siguiente:

```
1  function miFuncion()  
2  {  
3  |    console.log('Hola estoy dentro de una función');  
4  }  
5  
6  miFuncion();  
7
```



Para **realizar un llamado dos veces**, haremos:

```
1  ✓ function miFuncion()  
2    {  
3      console.log('Hola estoy dentro de una función');  
4    }  
5  
6    miFuncion();  
7    miFuncion();  
8
```



Funciones, return y parámetros

Se generó una función que tomará parámetros, es decir, se pasan determinados datos a esta función:

```
<p id="caja"></p>

<script>
function miOperacion(p1, p2) {
  return p1 * p2;
}
document.getElementById("caja").innerHTML = miOperacion(4, 3);
</script>
```



Se observa que **return** permite retornar un valor, y de esta forma también se pueden variar los datos que se pasan en la misma función. Es decir, **una misma función sirve para todo**.

Por ejemplo, se puede volver a utilizar de esta forma:

```
<p id="cajaA"></p>
<p id="cajaB"></p>

<script>
function miOperacion(p1, p2) {
  return p1 * p2;
}
document.getElementById("cajaA").innerHTML = miOperacion(4, 3);
document.getElementById("cajaB").innerHTML = miOperacion(2, 18);
</script>
```

La misma función dará dos resultados diferentes,
sin necesidad de crear una función nueva:

12

36

Es decir:

```
function ejemplo(parametro1, parametro2, parametro3)
{
  //lo que hace a función
}
```



Ámbito de la variable

Concepto

Algo importante a tener en cuenta es que **las variables pueden ser locales o globales**.
Por ejemplo:

```
1  function miOperacion() {  
2      |    var resultado = 30;  
3      |  
4      |  
5      |    console.log(resultado);  
6      |  
7  }
```

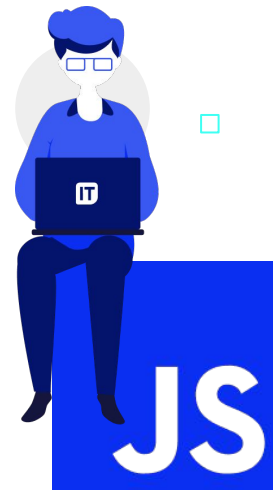
El código dará el siguiente **error**, porque la variable resultado está dentro del ámbito de **miOperación()** y no es posible utilizarla por fuera.

```
✖ 18:05:33.604 Uncaught ReferenceError: resultado is not defined  
    at ejemplo.html:17
```



Revisión

- Repasar el concepto de **función**.
- Aprender cómo funcionan los **módulos desde el código**.
- Estudiar cómo **trabajar con parámetros** desde una **función**.
- Entender cómo definir correctamente el **ámbito de una variable**.
- Trabajar en el **Proyecto Integrador**.
- Realizar las preguntas necesarias al/la docente antes de continuar.



**¡Sigamos
trabajando!**