

Una función es una estructura donde guardaremos un fragmento de código para ejecutarlo cuando lo necesitemos.

Dentro de una función podemos ejecutar cualquier código que necesitemos siempre y cuando sea válido.

Para crear una función usaremos la siguiente sintaxis.

```
function nombreDeLaFunción(){  
  //Código a ejecutar  
}
```

Cuando queramos ejecutarla, pondremos su nombre y paréntesis, esto es **MUY IMPORTANTE**, si no ponemos los paréntesis, NO se ejecutará.

```
nombreDeLaFunción()
```

Las variables y las constantes tienen un ámbito de actuación, es decir, existen y pueden ser leídas, asignadas, reasignadas dentro de su ámbito.

```
let numberA = 3;
let numberB = 2;

function suma(){
  console.log(numberA + numberB)
}
```

En este caso numberA y numberB tienen un ámbito “**global**”, es decir se pueden leer desde cualquier parte del documento.

```
function suma() {
  let numberA = 3;
  let numberB = 2;
  console.log(numberA + numberB);
}

console.log(numberA, numberB);
```

En este caso numberA y numberB tienen un ámbito “**local**”, es decir sólo existen dentro de la función suma, por lo que NO existen fuera de esa función.

Las funciones tienen la opción de ejecutar código con información que le pasemos desde fuera.

No existe un número máximo de parámetros y deben colocarse separados por comas.

```
function suma(a, b){  
  console.log(a + b)  
}
```

A la hora de llamar a la función le debemos pasar los parámetros en el mismo orden que los hemos declarado en la función.

```
suma(3, 2)
```

En esta llamada “a” tomará el valor de 3 y “b” tomará el valor de 2

```
suma(2, 3)
```

En esta llamada “a” tomará el valor de 2 y “b” tomará el valor de 3

Las funciones **SIEMPRE** retornan un valor al ejecutarse, si no establecemos ese retorno será “undefined” y ese retorno lo podremos guardar una variable / constante para usarlo, para establecer el retorno usaremos la palabra “return”, esto **CORTA** la ejecución de la función, si ponemos algo después **NO SE EJECUTARÁ**

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

```
suma(3, 2);
```

Es **MUY IMPORTANTE** tener en cuenta que el valor de retorno se **PERDERÁ** si no lo guardamos.

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

```
const resultado = suma(3, 2);
```

En un programa se suelen crear bastantes funciones, por lo que se creó una forma distinta de escribir funciones, más corta y más simple.

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

```
const suma = (a, b) => {  
  return a + b;  
};
```

Cuando tenemos una función que sólo hace un return, podemos omitir las llaves y la palabra return, de esta forma conseguimos una sintaxis mucho más simple de escribir.

```
const suma = (a, b) => a + b;
```

Además si la función sólo recibe un parámetro, también podemos omitir los paréntesis.

```
const double = a => a * 2;
```