



4.A. Estructuras definidas por el usuario en JavaScript

2.- Creación de funciones.

2.1.- Parámetros.

Cuando se realiza una llamada a una función, muchas veces es necesario pasar parámetros (también conocidos como argumentos). Este mecanismo nos va a permitir enviar datos entre instrucciones.

Para pasar parámetros a una función, tendremos que escribir dichos parámetros entre paréntesis y separados por comas.

A la hora de definir una función que recibe parámetros, lo que haremos es, escribir los nombres de las variables que recibirán esos parámetros entre los paréntesis de la función.



Veamos el siguiente ejemplo:

```
function saludar(a,b) {  
  
    alert("Hola " + a + " y " + b + ".");  
  
}
```

Si llamamos a esa función desde el código:

```
saludar("Martin","Silvia");           // Mostraría una alerta con el texto: Hola Martin y Silvia.
```

Los parámetros que usamos en la definición de la función *a* y *b*, no usan la palabra reservada **var** para inicializar dichas variables. Esos *parámetros a y b serán variables locales a la función*, y se inicializarán automáticamente en el momento de llamar a la función, con los valores que le pasemos en la llamada. En el siguiente apartado entraremos más en profundidad en lo que son las variables locales y globales.

Otro ejemplo de función que devuelve un valor:

```
function devolverMayor(a,b) {  
  
    if (a > b) {  
  
        return a;  
  
    } else {  
  
        return b;  
  
    }  
  
}
```

Ejemplo de utilización de la función anterior:

```
document.write ("El número mayor entre 35 y 21 es el: " + devolverMayor(35,21) + ".");
```

Para saber más

[Más información sobre funciones.](#)

[Ejemplo de funciones con parámetros.](#)

Citas para pensar

"La inteligencia es la función que adapta los medios a los fines."

HARTMANN, N.

Créditos de la imagen.

Autoría: Matteo Piotto.

Licencia: CC BY-SA 2.0.

◀ Solución a la tarea para DWEC04

Ir a...



4.B. Objetos JavaScript en W3Schools ▶