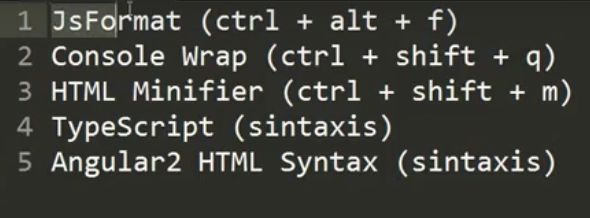
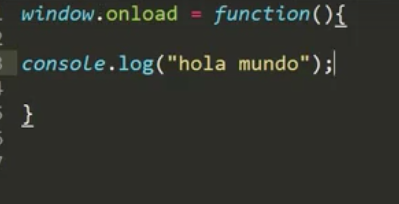
1. Instalar plugin en Sublime text



1. Organizar rápidamente el código
2. Genera resultados en la consola de una forma más rápida cuando queramos saber el valor de una variable con este atajo nos va a arrojar una consola
3. Permite comprimir en una versión mini el código html, css o js, útil para cuando estoy en producción cuando deseamos que las personas no tengan acceso de una forma clara y legible se comprime y así queda la versión .mini
4. Mostrar sintaxis más agradable de una mejor forma

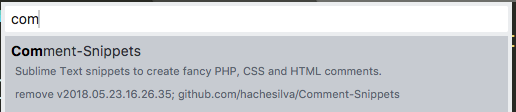
El plugin sigue est

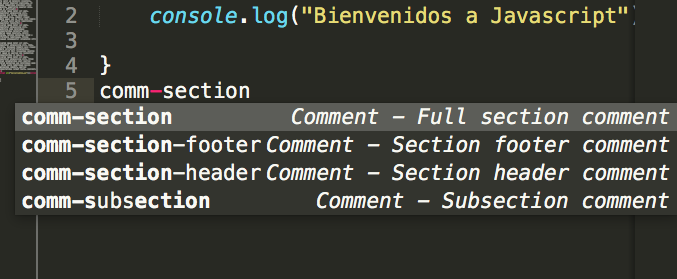


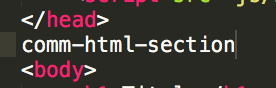


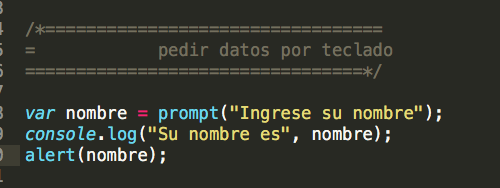
# Comentarios

Instalar el plugin

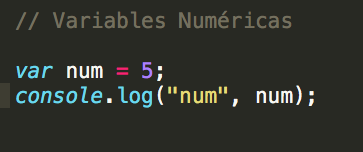


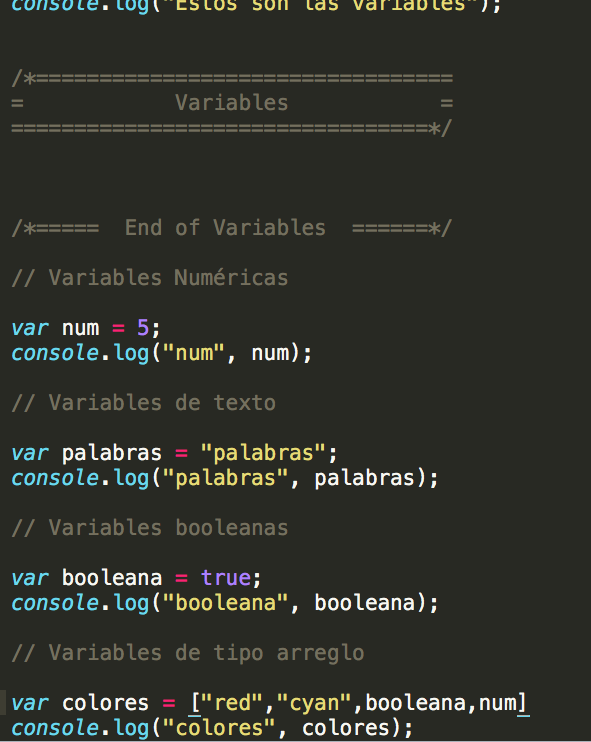




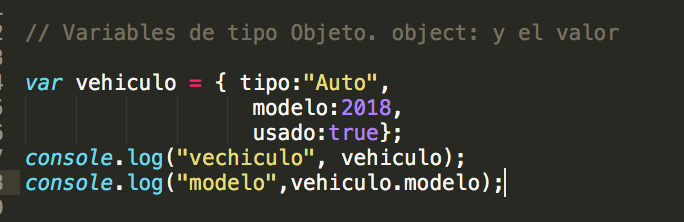


Seleccionamos la palabra y le damos ctrl+shift+q y sale esto



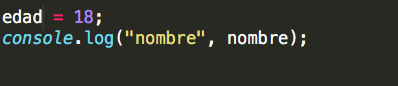




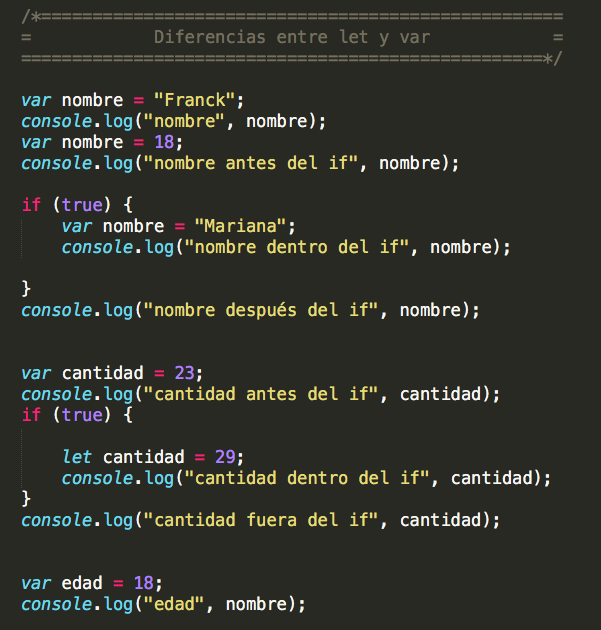


# Use strict

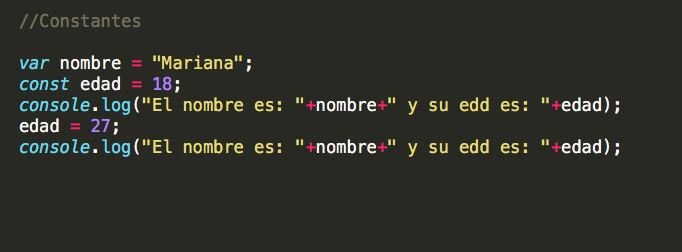




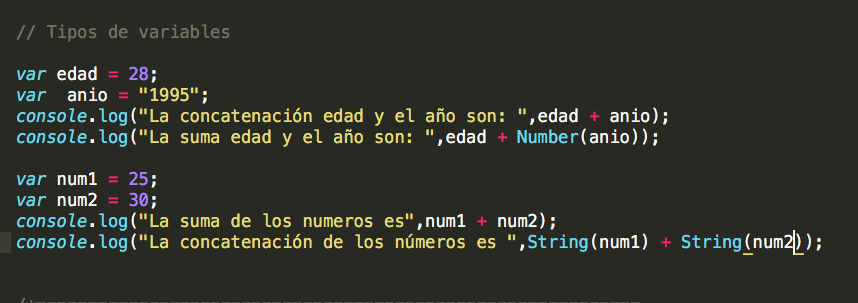
# Diferencia entre var y left



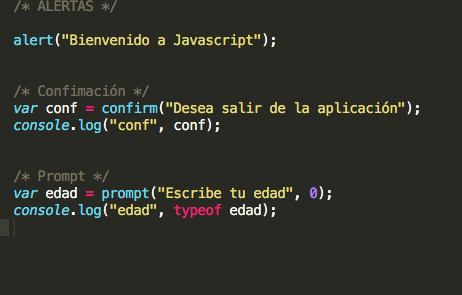
# Constantes



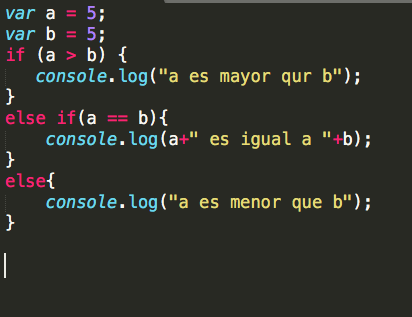
# Tipos de variables



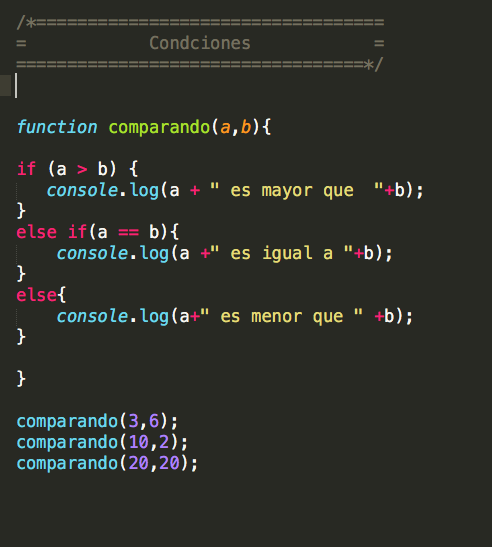
# Ventanas

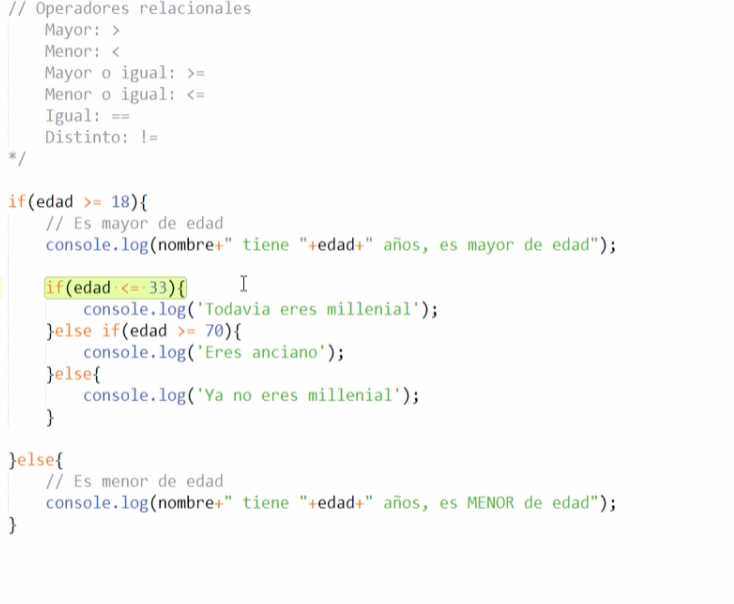


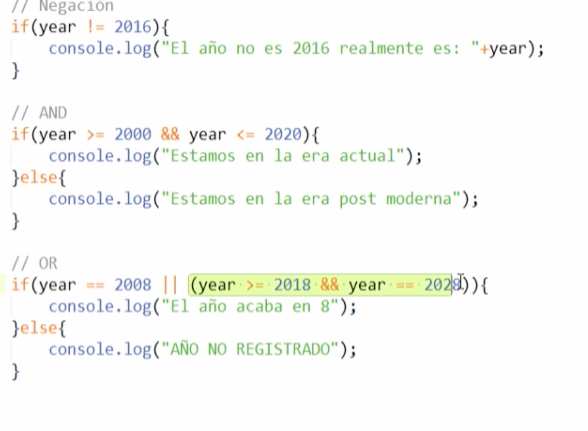
# Condiciones y Cambios

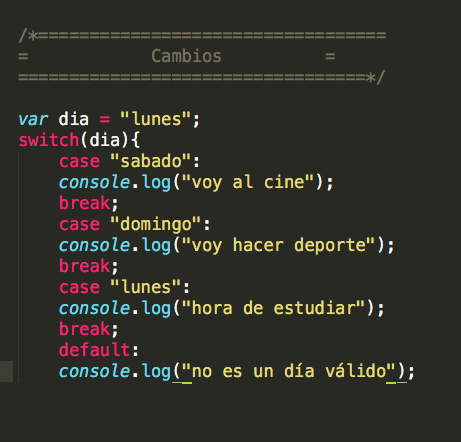


Ahora con funciones

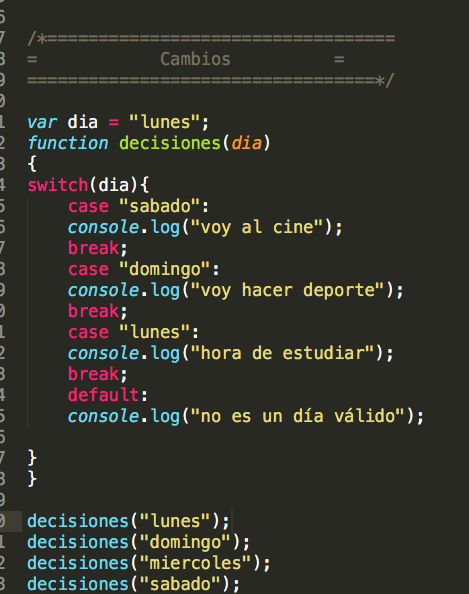




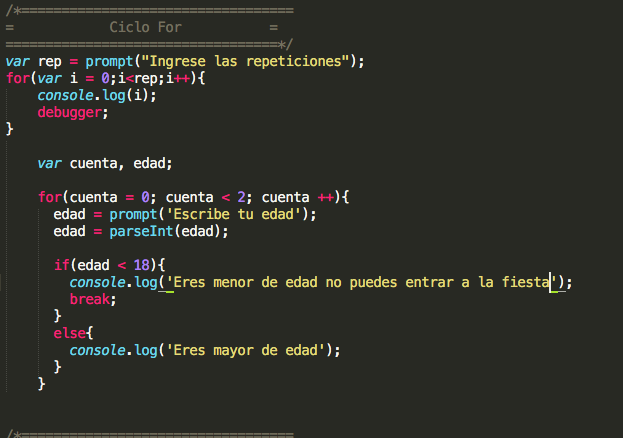


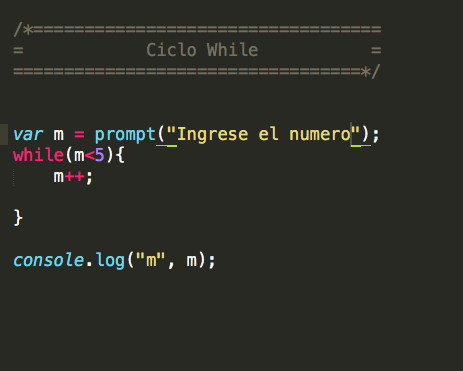


Con funciones

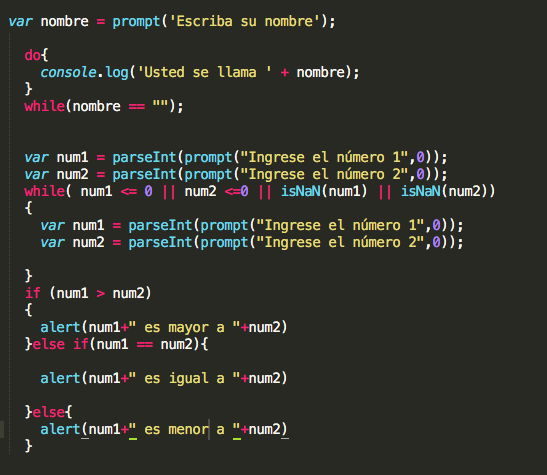


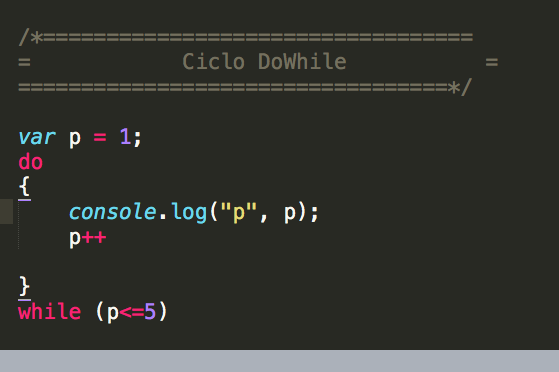
# Ciclos

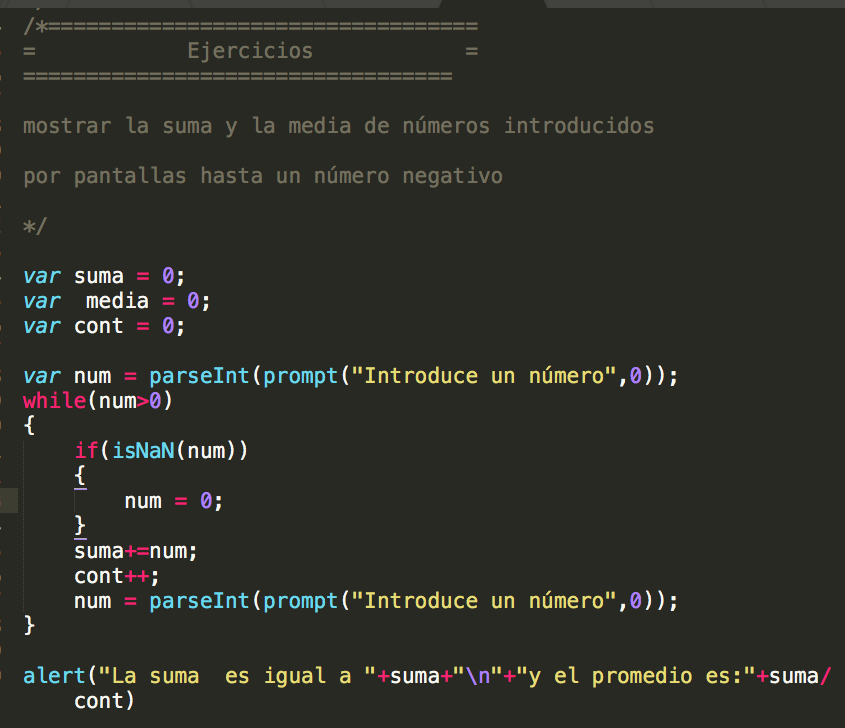


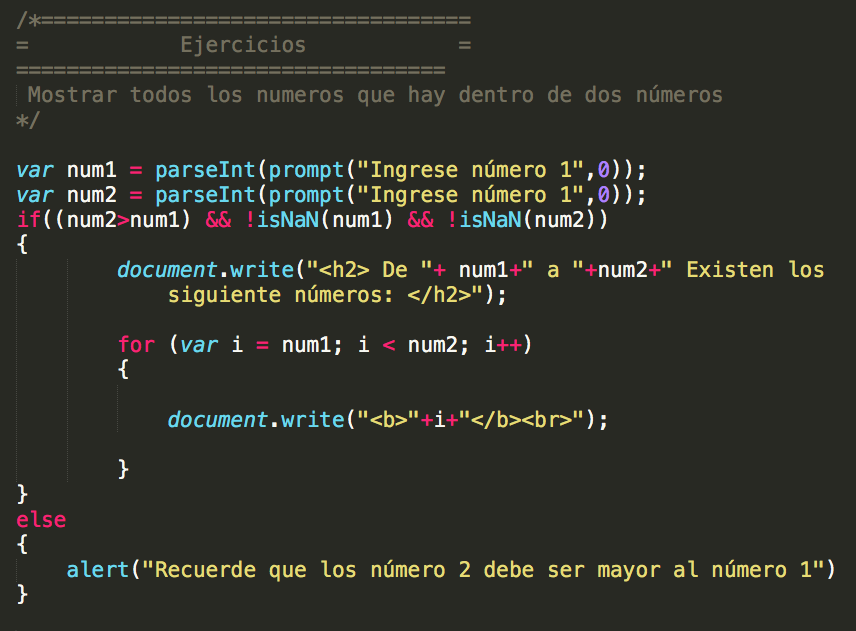




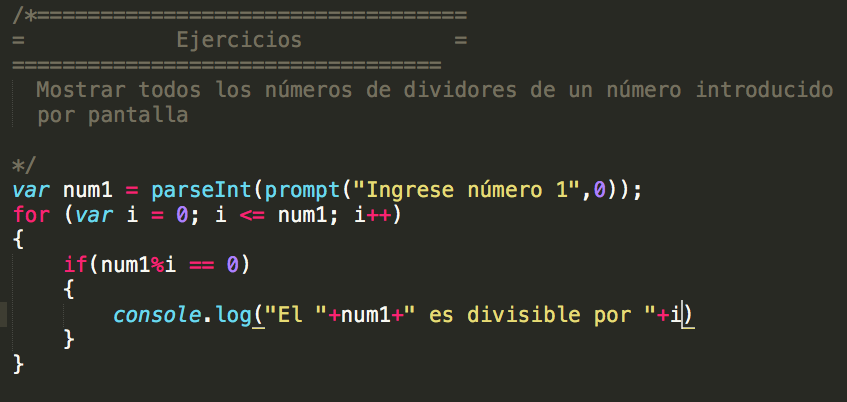


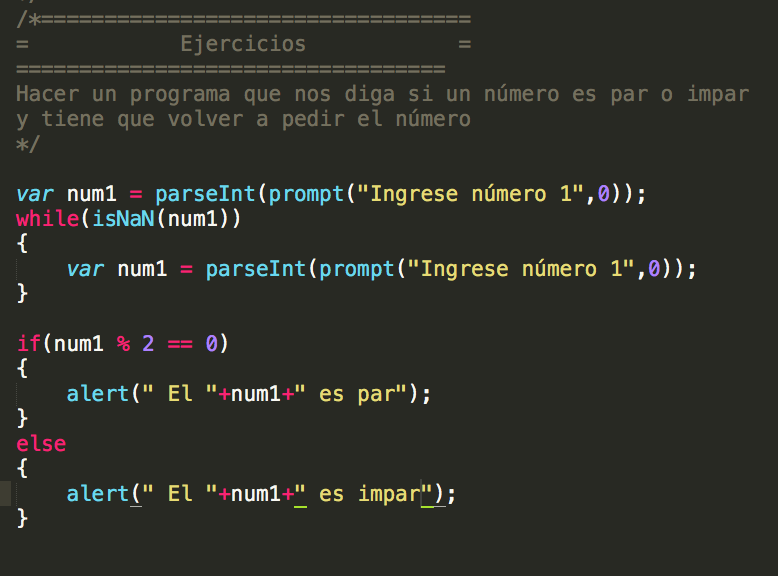


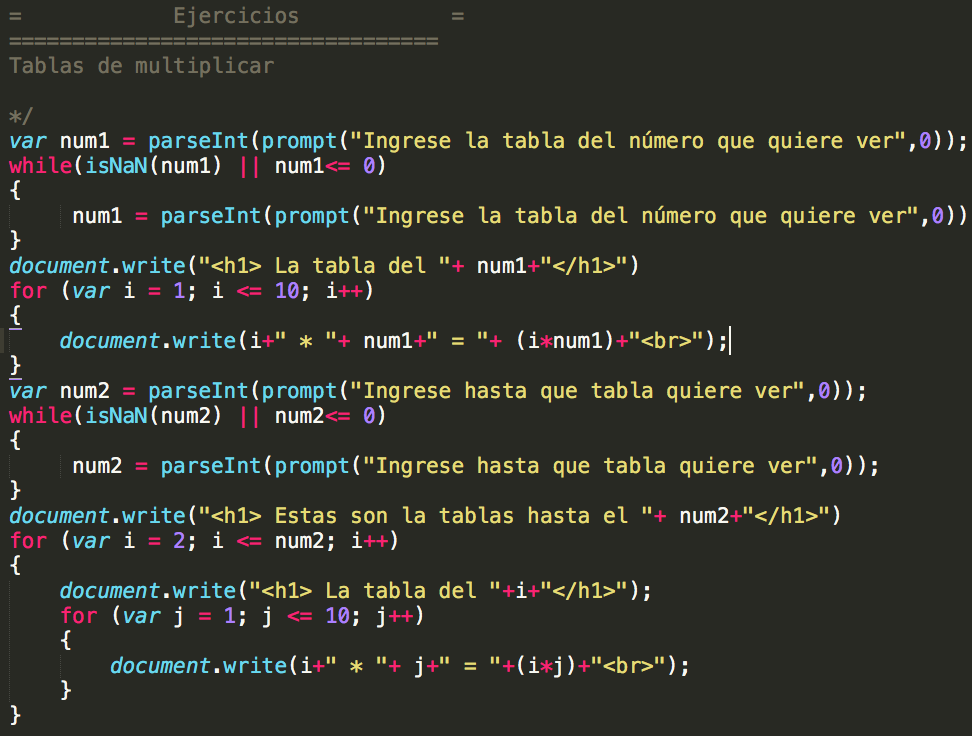


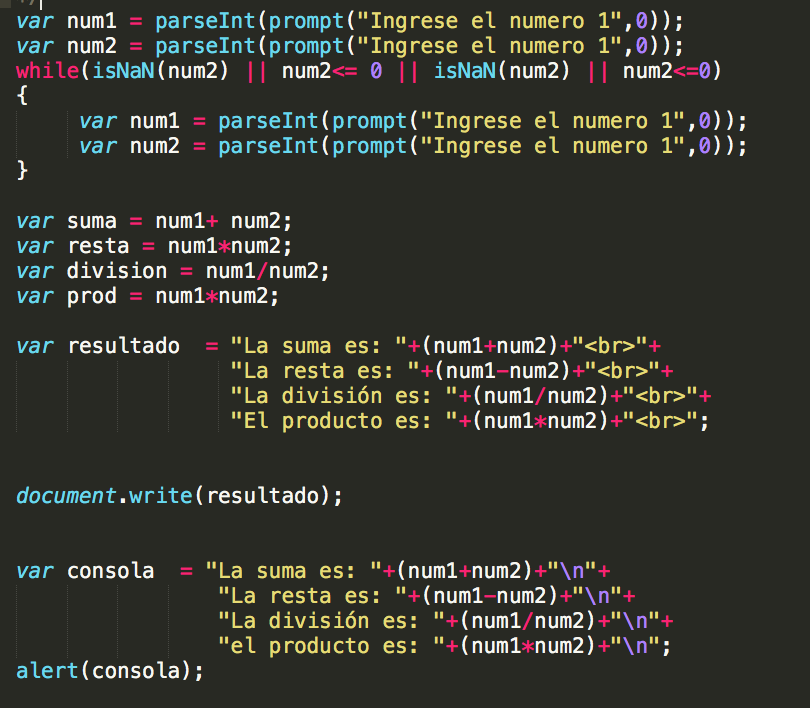




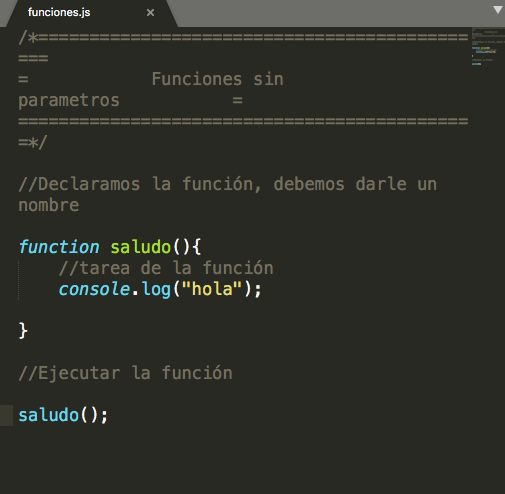




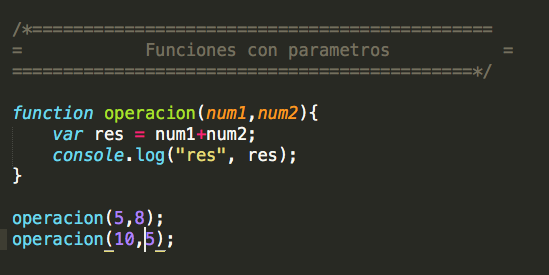


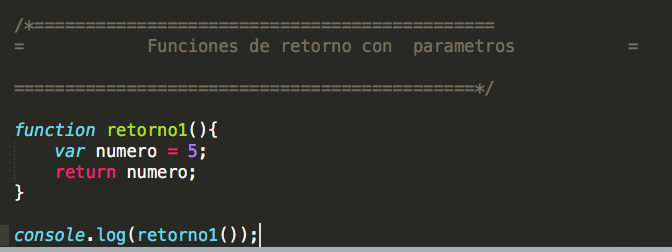


# 10. Funciones







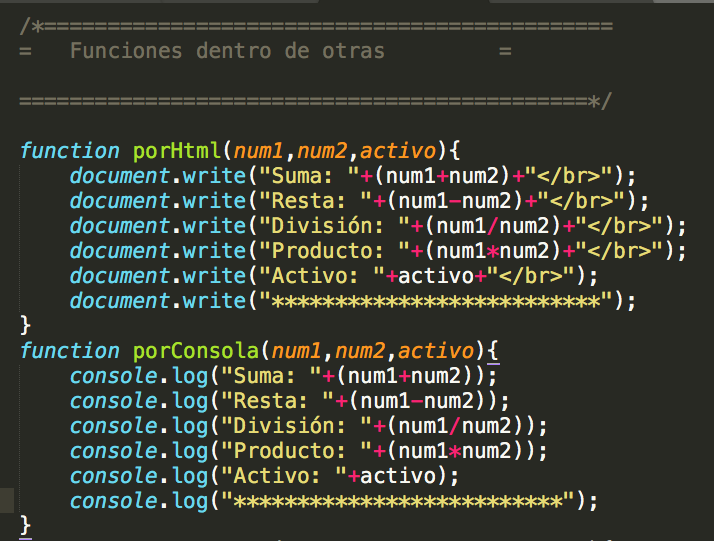


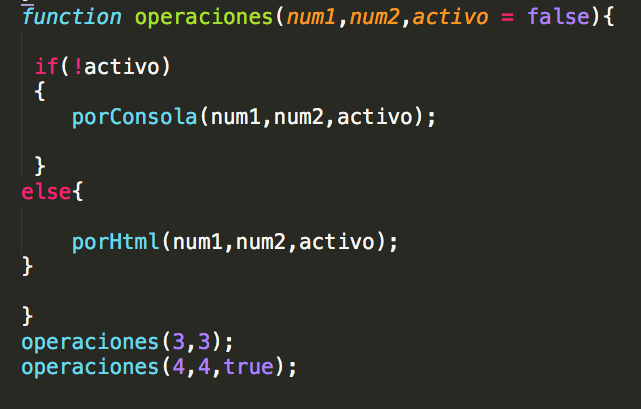






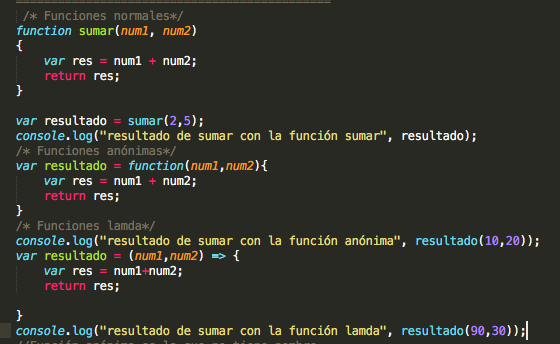






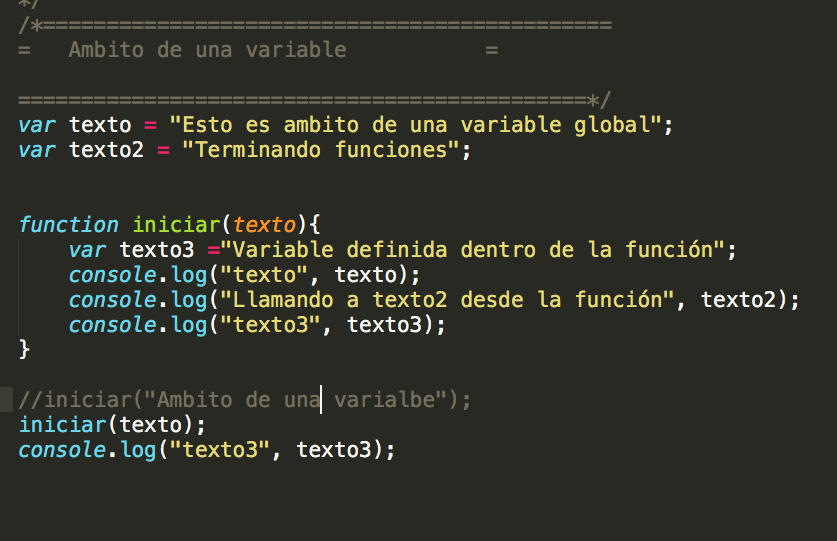


Una función de callback es una función anónima que se pasa como parámetro a otra función P y esa función P ejecuta la función anónima q se esta pasando





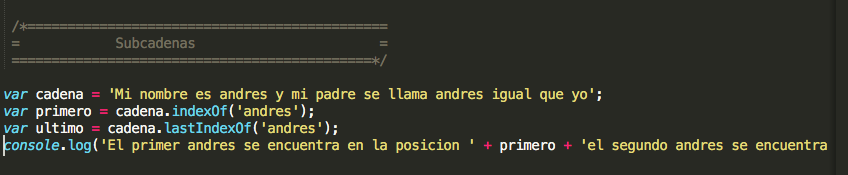


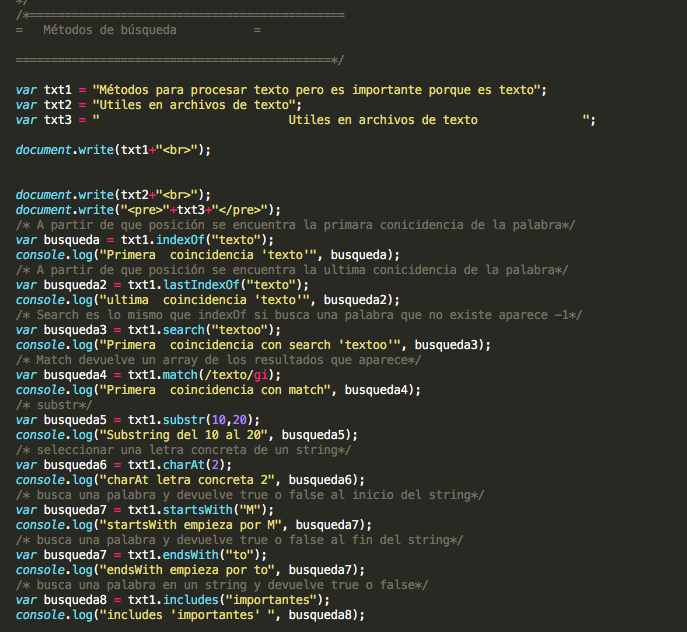


# 11.Métodos para procesar textos



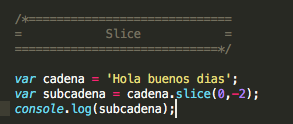
## Métodos de búsqueda



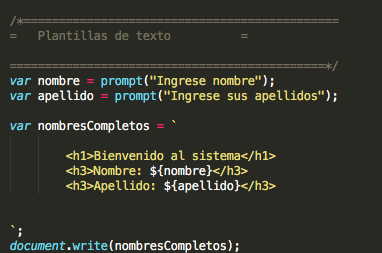


## Funciones de reemplazo





## Plantillas de Texto

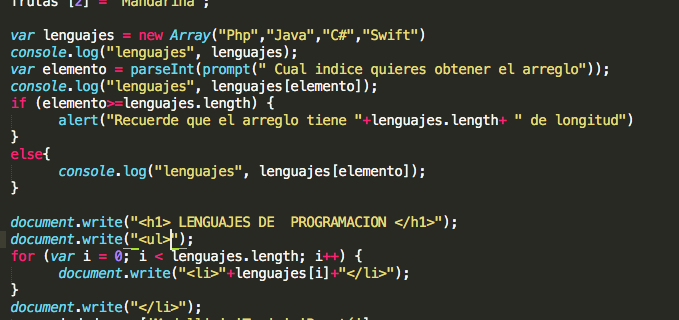




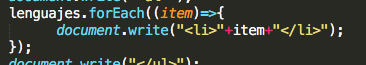


# 15. Arreglos

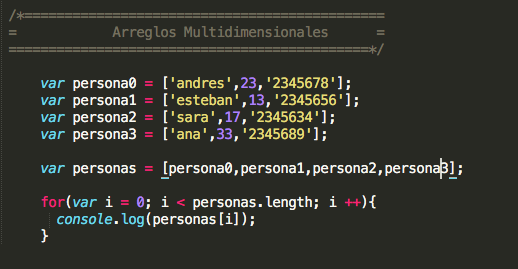


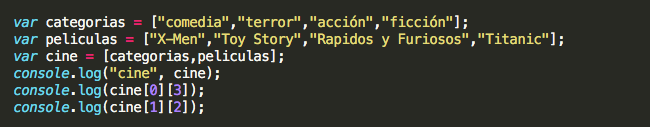


## Foreach



## Arreglos Multidimencionales

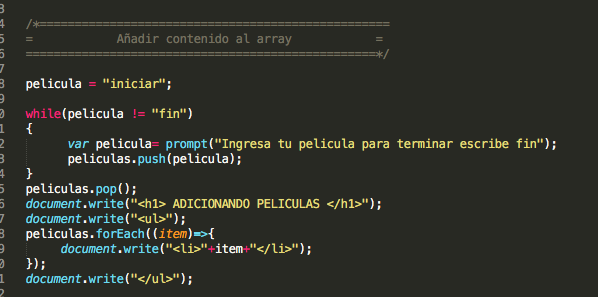




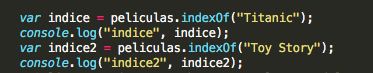
# 16. Operaciones con arreglos



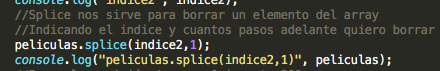
## Añadir Contenido a un Array

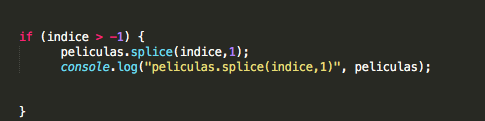


## Encontrar índice de un array



## Borrar elemento a partir de un Indice





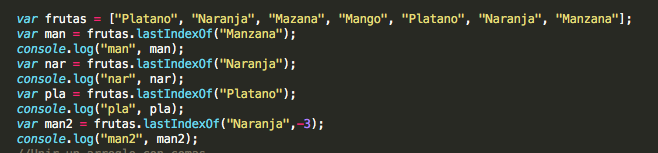
## Insertar un elemento del array en una posición dada



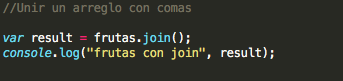
## Reemplazar índice dentro del arreglo



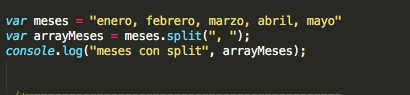
## Encontrar últimos índices de arreglos



## Unir arreglos



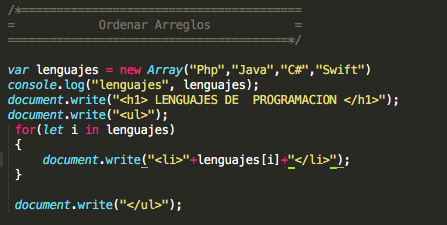
## Convertir cadena en arreglo



## Ordenar arreglos



## Recorrer un arreglo con for in

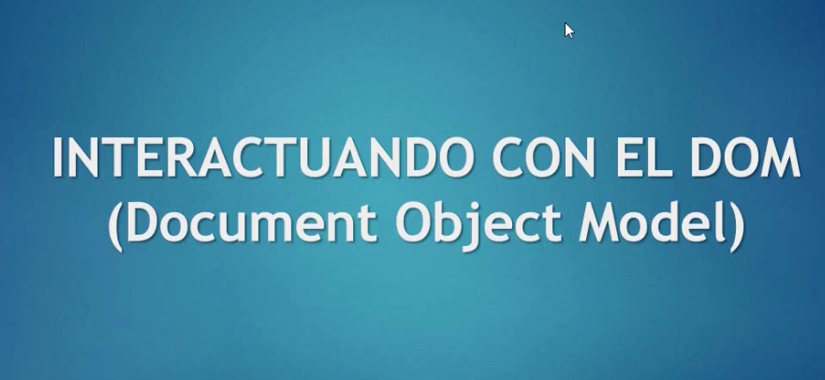


## Busquedas dentro de un array



## Ejercicio de array

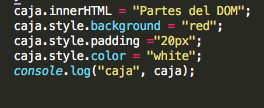
# Interactuando con el DOM



## Seleccionar elementos por su id o clase









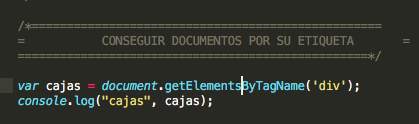
## Agregando clases



## QuerySelector

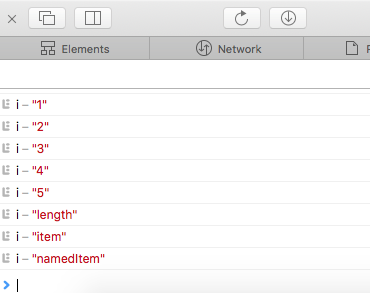


## Seleccionar elementos por su etiqueta

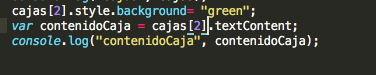


Me lista todos los elementos del DOM





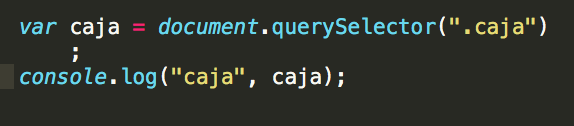
## Cambiar el contenido



Seleccionar algún elemento dentro del docuemnto HTML

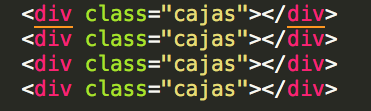
Cramos una caja

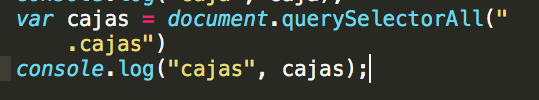




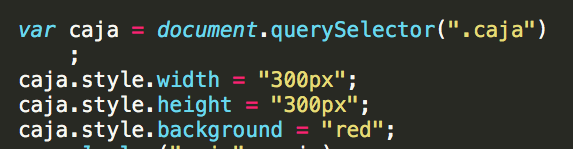


Ahora vamos a nombrar varias cajas









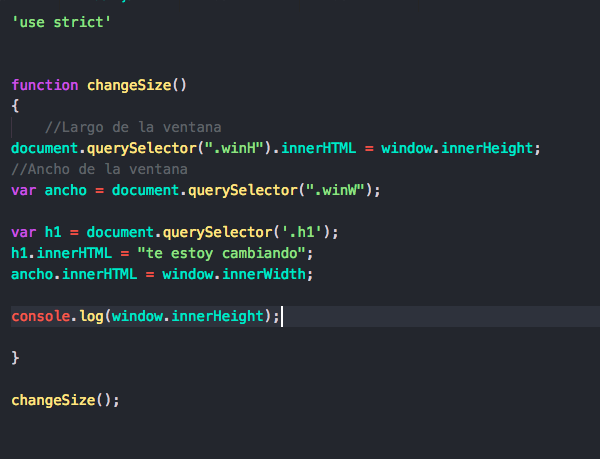


# Interactundo con BOM

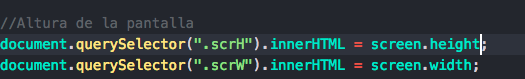




Ahora lo volvemos una función



## Altura de la pantalla



Para que el archivo de javascript que esta definido en la cabecera pueda acceder con el resto del documento se agrega esta función, esto es para incluirlo en el cuerpo y hacer que trabaje



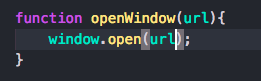
Ahora si creamos una función apra que me redireccione alguna parte

## Redireccionar una pagina





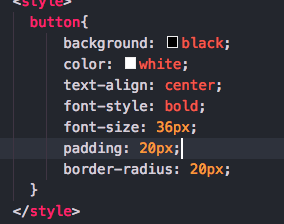
## Abrir una nueva pestaña

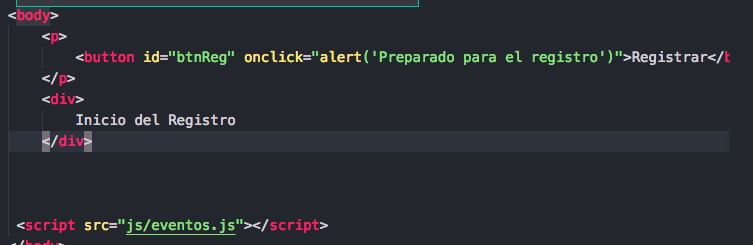


# Eventos

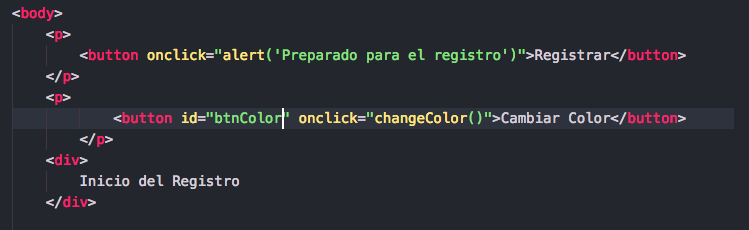
| **Evento** | **Descripción** | **Elementos para los que está definido** |
| --- | --- | --- |
| onblur | Deseleccionar el elemento | <button>, <input>, <label>, <select>, <textarea>, <body> |
| onchange | Deseleccionar un elemento que se ha modificado | <input>, <select>, <textarea> |
| onclick | Pinchar y soltar el ratón | Todos los elementos |
| ondblclick | Pinchar dos veces seguidas con el ratón | Todos los elementos |
| onfocus | Seleccionar un elemento | <button>, <input>, <label>, <select>, <textarea>, <body> |
| onkeydown | Pulsar una tecla (sin soltar) | Elementos de formulario y <body> |
| onkeypress | Pulsar una tecla | Elementos de formulario y <body> |
| onkeyup | Soltar una tecla pulsada | Elementos de formulario y <body> |
| onload | La página se ha cargado completamente | <body> |
| onmousedown | Pulsar (sin soltar) un botón del ratón | Todos los elementos |
| onmousemove | Mover el ratón | Todos los elementos |
| onmouseout | El ratón *"sale"* del elemento (pasa por encima de otro elemento) | Todos los elementos |
| onmouseover | El ratón *"entra"* en el elemento (pasa por encima del elemento) | Todos los elementos |
| onmouseup | Soltar el botón que estaba pulsado en el ratón | Todos los elementos |
| onreset | Inicializar el formulario (borrar todos sus datos) | <form> |
| onresize | Se ha modificado el tamaño de la ventana del navegador | <body> |
| onselect | Seleccionar un texto | <input>, <textarea> |
| onsubmit | Enviar el formulario | <form> |
| onunload | Se abandona la página (por ejemplo al cerrar el navegador) | <body> |

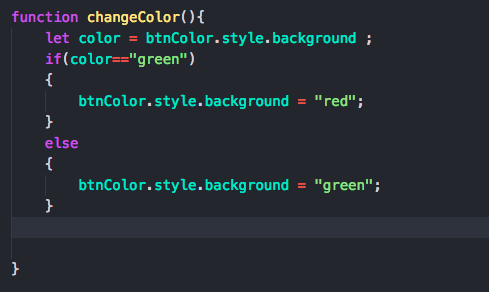
## Desde Html



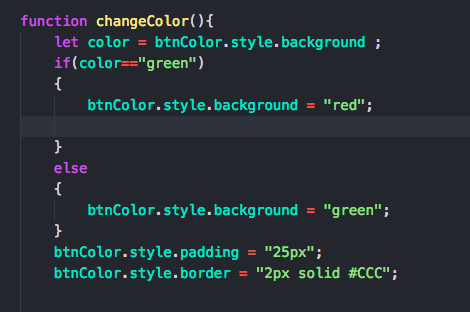


## Con funciones

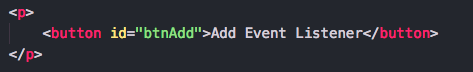


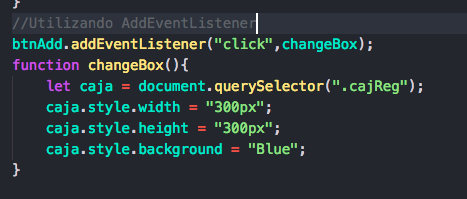


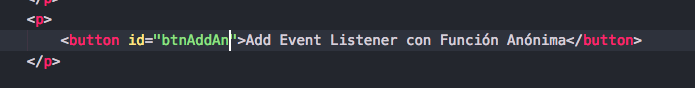
## Agregando padding y margin



## addEventListener

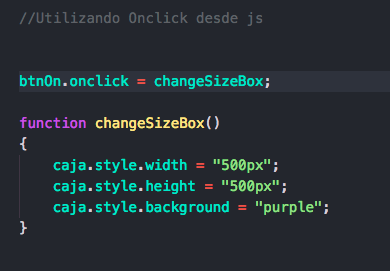




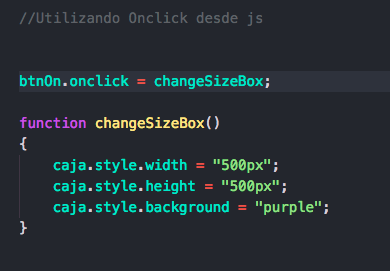




## .onclick desde js con función

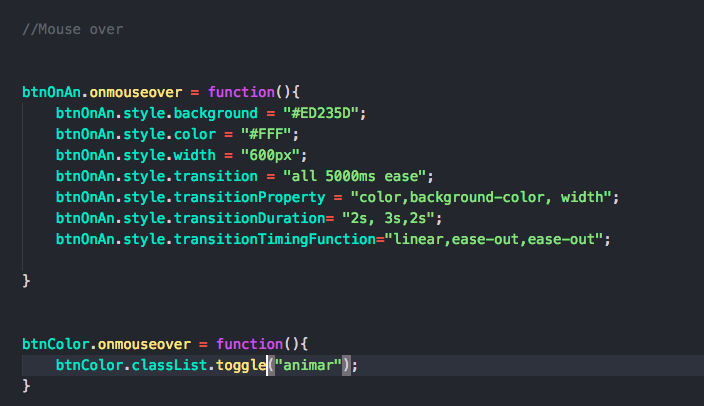


## Onclick desde js con anónima



## Mouse over

Nota: Para lograr la transition siempre primero debemos especificar un ancho





## Onclick ondblclick





## Cuadrado de un número

