

**SENA Servicio Nacional de Aprendizaje**

**DESARROLLO DE APLICACIONES PARA DISPOSITIVOS MÓVILES**

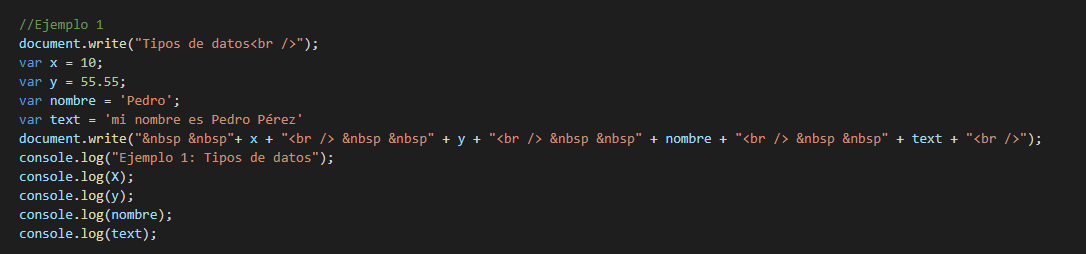
**Angélica Liliana Parada Velandia**

**Bogotá D.C.**

**Junio 2020**

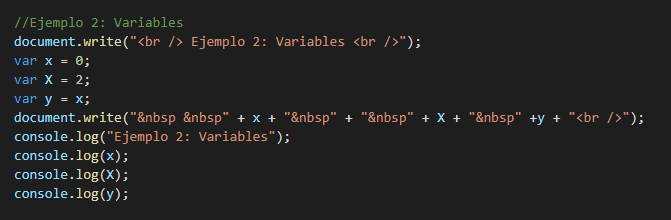
***Tipos de datos***

Valores con los que puede trabajar un programa.



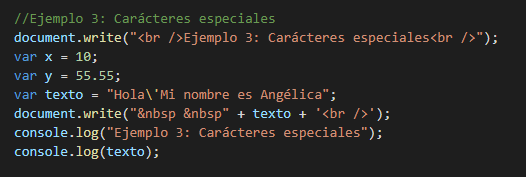
***Variables***

Contenedores para registrar datos.



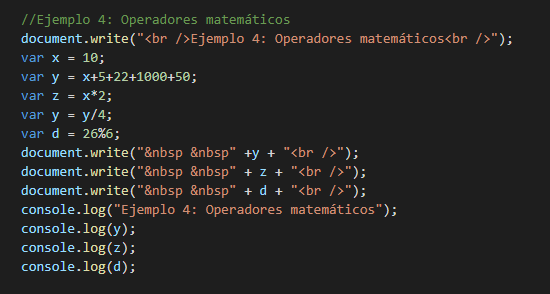
***Caracteres especiales***

Pueden ser añadidos a una cadena de texto usando \.



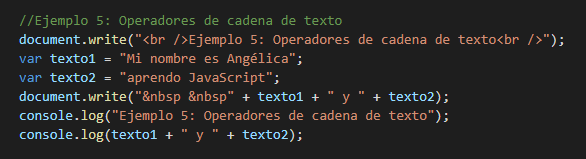
***Operadores matemáticos***

Realizan funciones aritméticas.



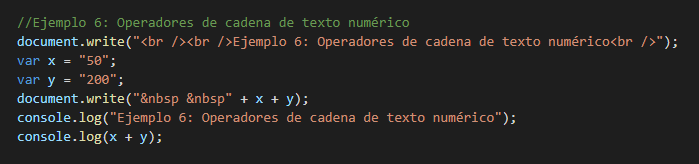
***Operadores de cadena de texto***

Concatenación para unir o juntar cadenas.



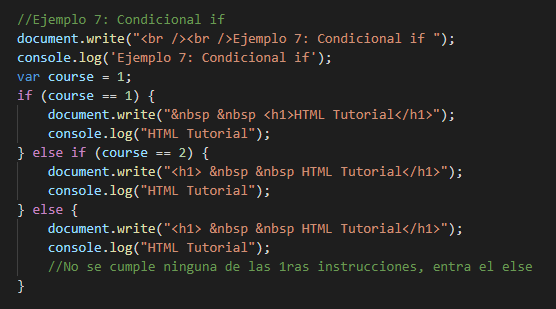
***Operadores de cadena de texto numérico***

Los números entre comillas son leídos como texto.



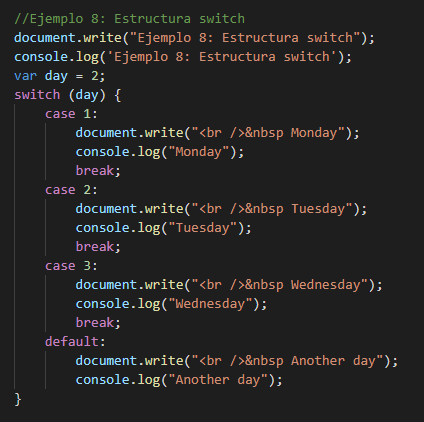
***Condicional if***

Un bloque de código se ejecuta si la condición en verdadera.

******

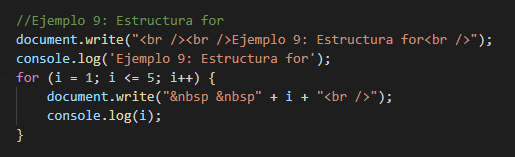
***Estructura switch***

Evalúa múltiples condiciones.

******

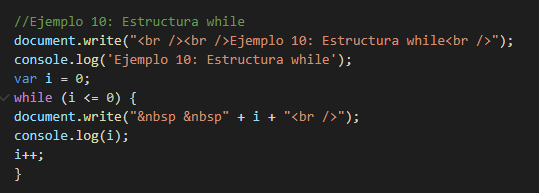
***Estructura for***

Bucle que se usa cuando se quiere ejecutar varias veces la misma línea de código.

******

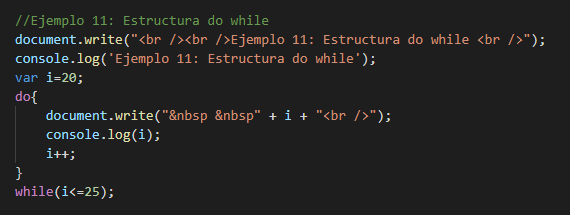
***Estructura while***

Mientras es verdadero se ejecuta el bloque de código.

******

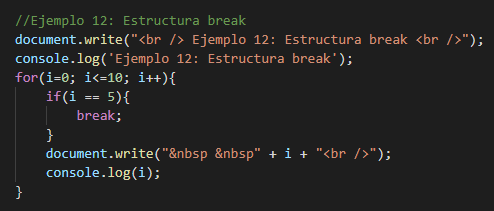
***Estructura do while***

El bucle ejecuta el bloque de código por lo menos una vez antes de evaluar la condición.



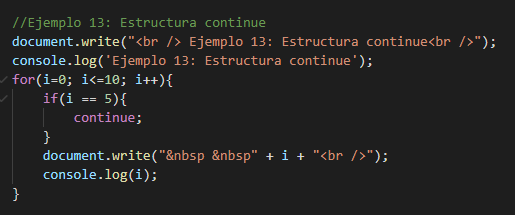
***Estructura break***

Finaliza el bucle.



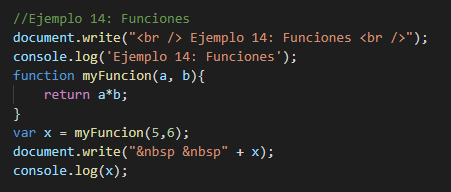
***Estructura continue***

Detiene una sola iteración del bucle y luego continúa ejecutándolo.



***Funciones***

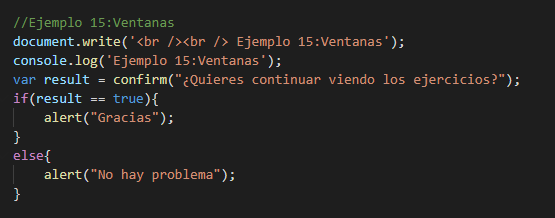
Código que ejecuta una tarea específica.



***Ventanas***

Mensaje emergente que se superpone en la ventana. Existen de tres tipos:

* Alert
* Prompt
* Confirm

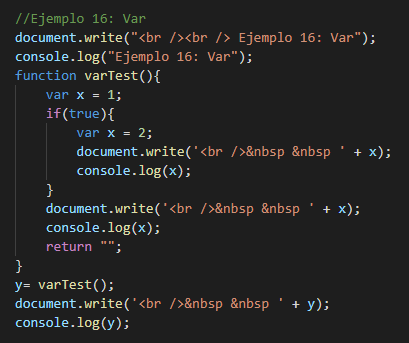


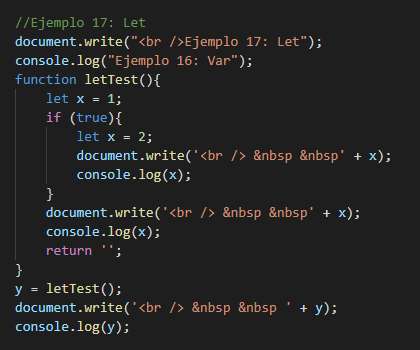
***EMACSCRIPT 6***

***Variables***

Existen tres maneras de definir una variable:

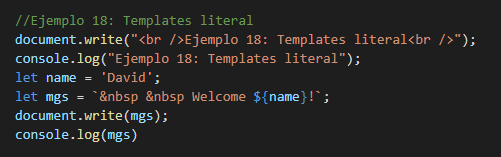
* Var
* Let
* Const

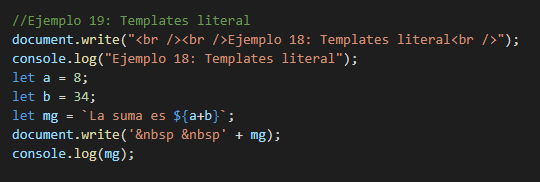




***Template literal***

Devuelven variables en una cadena.

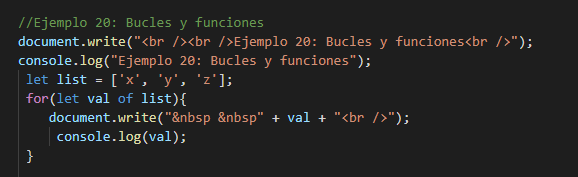




***Bucles y funciones***

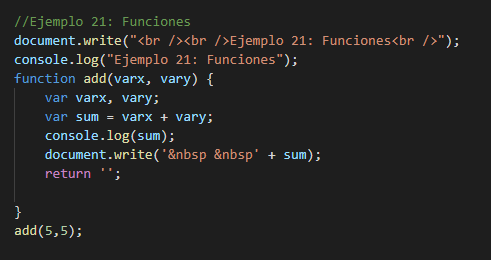
***For of***

Asigna un elemento.



***Funciones***

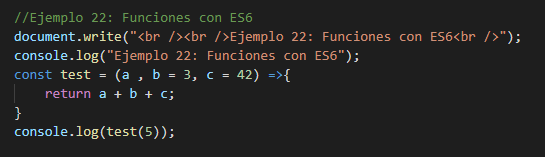
Cumple una tarea específica independiente al resto de código.



***Funciones***

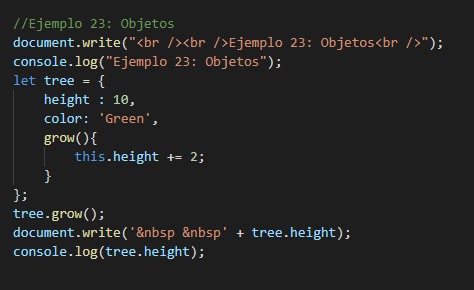
Cumple una tarea específica independiente al resto de código.

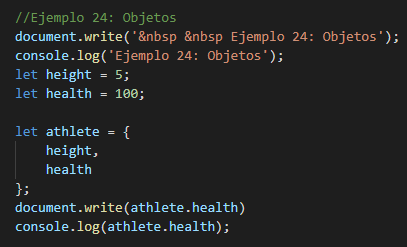
***Funciones con ES6***



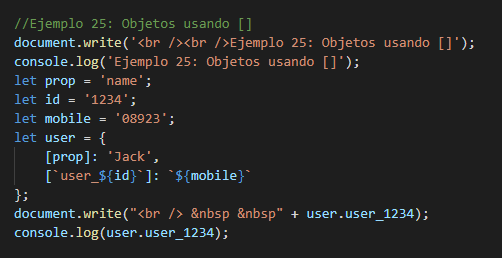
***Objeto***

Espacio de memoria que referencia o instancia una clase.



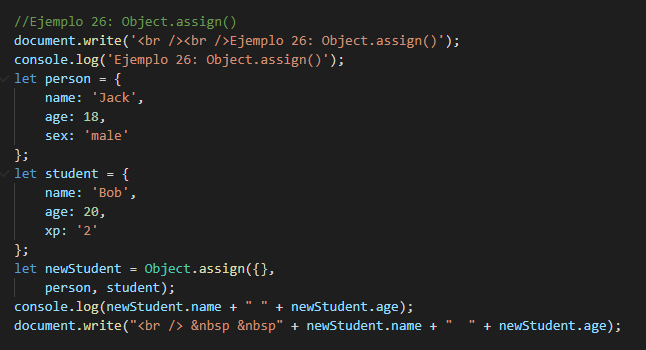


***Objeto usando []***



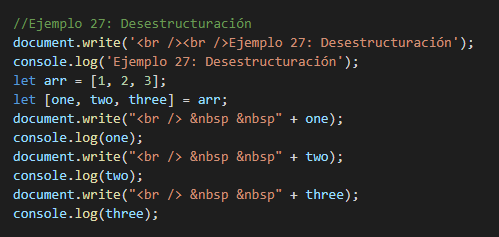
***Object.assign()***

Duplicado de objeto ya existente.



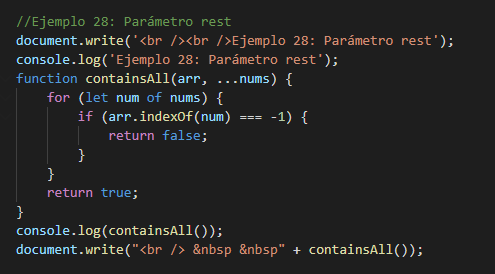
***Desestructuración***

Descompone un array en diferentes variables.



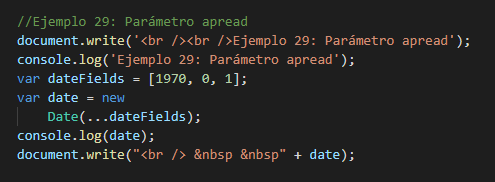
***Parámetro rest***

Permite pasar una cantidad de parámetros variables en una función.



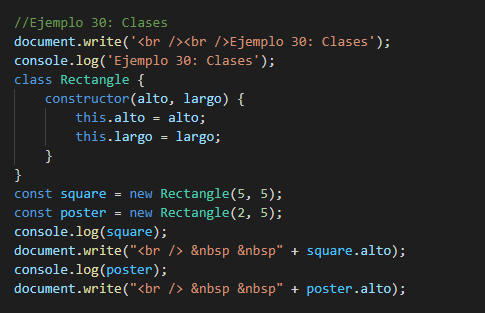
***Parámetro apread***

Usado para pasar una cantidad de variables en una función, pero en objetos, arrays o argumentos de funciones.



***Clases***

Tipo de dato abstracto usado para crear múltiples objetos.

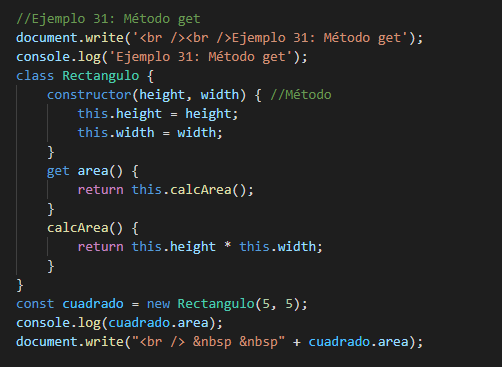


***Métodos de clase***

Bloque de código que pertenece a una clase y contiene una serie de instrucciones.

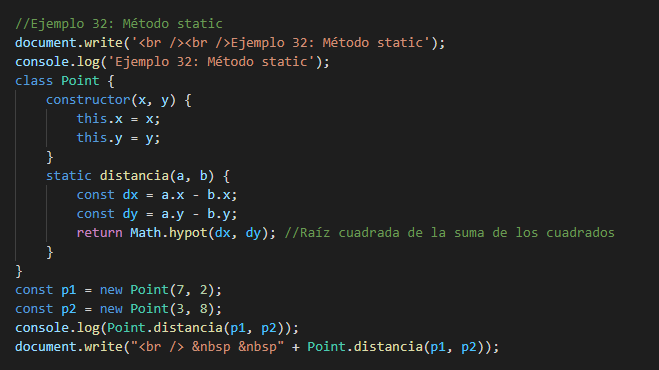
***Método get***

Obtiene un dato.



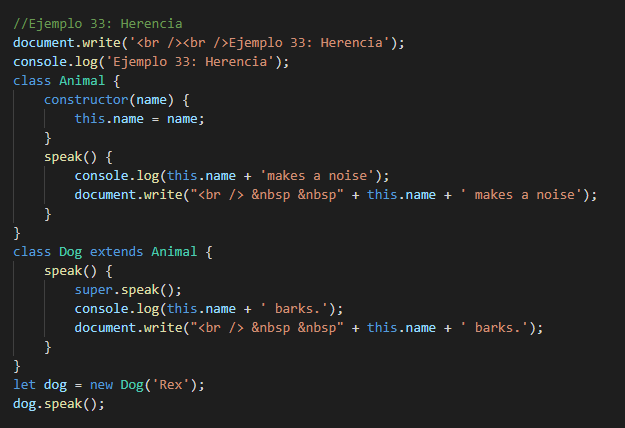
***Método static***

Crea una función para una aplicación



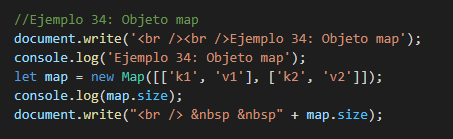
***Herencia***

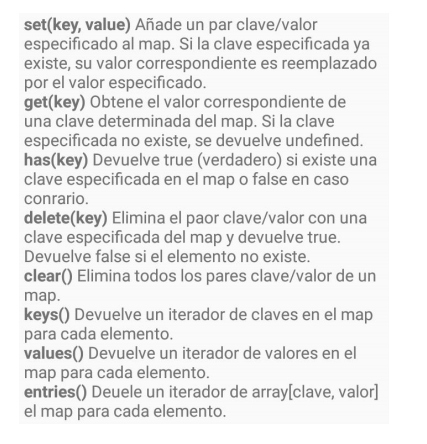
Extends crea una case secundaria que hereda todas las propiedades del método.



***Objeto map***

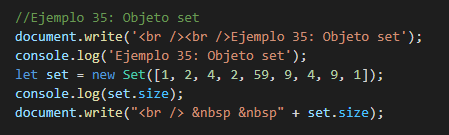
Usado para contener clave/valor que pueden ser usados en objetos.



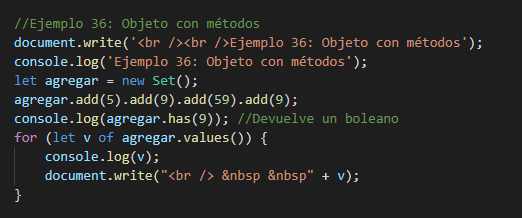


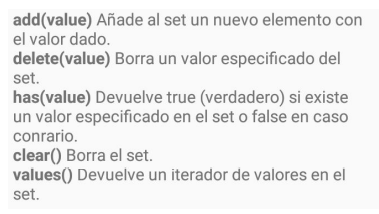
***Objeto set***

Utilizado para obtener la cantidad de valores distintos dentro de un array.



***Métodos***



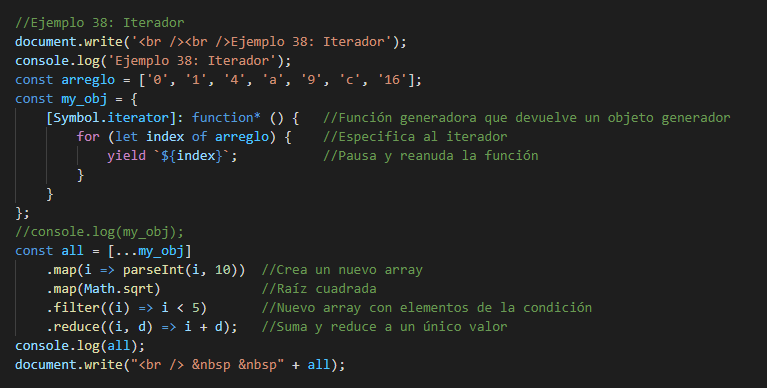


***Promises***

Mejora para la programación asíncrona.

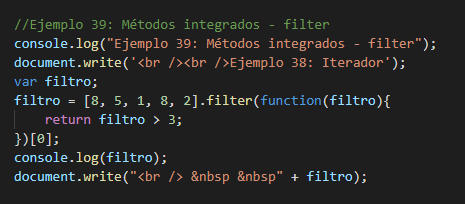


***Iterador***

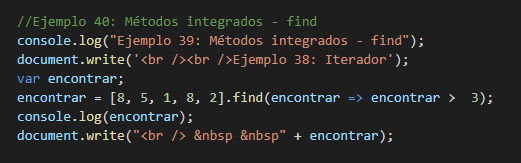


***Iterador***

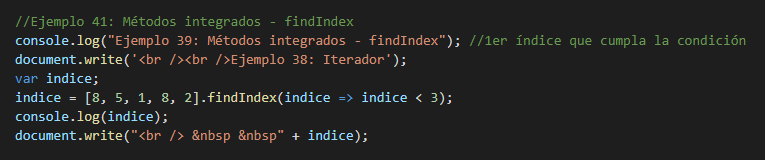
***Métodos integrados- filter***



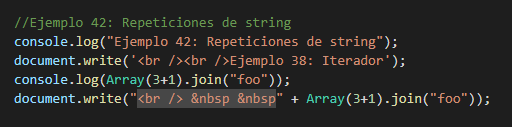
***Métodos integrados- find***



***Métodos integrados- findIndex***



***Repeticiones de Strings***



***Búsquedas de Strings***

