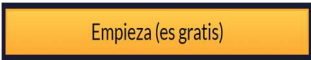

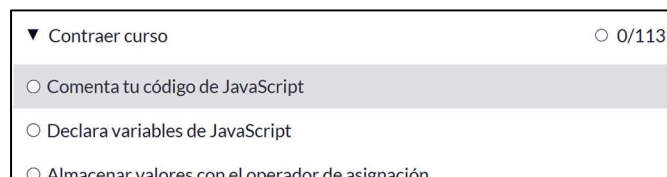


JavaScript es un lenguaje de scripting que puedes utilizar para hacer que las páginas web sean interactivas. Es una de las principales tecnologías de la web, junto con HTML y CSS, y es soportada por todos los navegadores modernos.

Para escribir tu código JavaScript (JS) requieres de un editor, como por ejemplo Visual Studio Code (VSC), donde puedes incrustar el código en un documento HTML y luego verás su ejecución en el navegador.

Otra aplicación donde puedes escribir tu código JS es en el Editor de código de FreeCodeCamp. Para ello debes:

- 1) Abrir: <https://www.freecodecamp.org/espanol/>
- 2) Pulsar  (la primera vez requiere de un mail para hacer el ingreso)
- 3) Selecciona la opción: 
- 4) Verás una lista de 113 temas del curso gratuito, selecciona cualquiera



- 5) Se habilitará, a la derecha, un panel donde: en la parte superior puedes escribir el código y en la parte inferior verá la salida por consola.

Veremos algunos ejemplos desde esta aplicación, para luego avanzar a la plataforma de VSC.

Las Variables y Tipo de Datos

Una variable en programación es un espacio en memoria que se utiliza para almacenar información. Las variables tienen un **nombre**, un **tipo** y un **valor**. Las variables son importantes en programación porque permiten a los programadores almacenar y manipular datos, escribir código más conciso y fácil de leer, y responder a los cambios en los datos.

En JS puedes declarar variables utilizando tres palabras clave: **var**, **let**, y **const**.

Debes tener en cuenta que los **nombres de variables** deben seguir ciertas reglas:

- Deben comenzar con una letra (a-z, A-Z) o un guion bajo (_).
- Pueden contener letras, números y guiones bajos.
- Son sensibles a mayúsculas y minúsculas (por ejemplo, miVariable y mivariable son diferentes).
- Deben evitar palabras reservadas de JavaScript, como var, let, const, function, etc.

Algunos **tipos** de datos de variables **primitivas** son:

- Números: Representan valores numéricos, como enteros o decimales.
- Cadenas (String): Representan texto.
- Booleanos: Representan valores verdaderos o falsos (true o false).
- Undefined: Indica que una variable no tiene un valor asignado.
- Null: Representa la ausencia de valor o un valor nulo.

Ejemplo # 1: Declarar variables primitivas

Código JS

```
1 //Variable global
2 var numero = 10;           //Tipo numérica
3 var nombre = "Mary";      //Tipo String
4
5 //Variable local
6 let cantidad = 5;
7 let mensaje = "Buen día..."; //Tipo String
8
9 //Constante
10 const pi = 3.1416;
11
12 var esMayor = true;       //Tipo Booleana
13 let x;                   //No definida
14 let y = null;             //Nula
15
16 //Imprimir valores
17 console.log(numero);
18 console.log(nombre);
19 console.log(cantidad);
20 console.log(mensaje);
21 console.log(pi);
22 console.log(esMayor);
23 console.log(x);
24 console.log(y);
```

Una **Variable local** es aquella que sólo es utilizable dentro del código de una función. Se eliminan cuando se completa la función

Una **Variable global** es aquella cuyo ámbito es todo el programa, incluso dentro de una función. Se eliminan cuando se cierra el navegador.

console.log es un método en JavaScript que se utiliza para imprimir mensajes o datos en la consola del navegador web o en la consola del entorno de desarrollo. Facilita la depuración y el registro de información durante el desarrollo de aplicaciones web.

Consola

```
10
Mary
5
Buen día...
3.1416
true
undefined
null
```

Con **console.log** puedes agrupar variables, concatenar texto, hacer saltos de línea, etc. Sustituye la línea 17 en adelante por lo siguiente:

```
console.log("Variable numero: ", numero, "\nVariable cantidad: ", cantidad);
console.log(mensaje, nombre, "¿Cómo estás?");
```

Existen otros Tipos de Datos para las variables, los datos **Compuestos**:

- Objetos (object): Colección de pares clave-valor.
- Arreglos (Arrays): Colección ordenada de elementos.
- Funciones (function): Son objetos que pueden ejecutarse como funciones.

Ejemplo # 2: Variables compuestas

```
1 // Objetos (object)
2 let persona = {
3   alumna: "Ana",
4   edad: 30,
5   ciudad: "Madrid"
6 };
7
8 // Arreglos (array)
9 let colores = ["rojo", "verde", "azul"];
10
11 // Funciones (function)
12 function saludar(nombre) {
13   console.log("¡Hola, " + nombre + "!");
14 }
15
16 //Llamando a la función saludar
17 saludar("Carlos"); // Imprimirá "¡Hola, Carlos!" en la consola
18
19 //Imprimiendo objeto y arreglo
20 console.log(persona);
21 console.log(colores);
```

```
¡Hola, Carlos!
{ alumna: 'Ana', edad: 30, ciudad: 'Madrid' }
[ 'rojo', 'verde', 'azul' ]
```

Dentro de la función, se utiliza la variable local **nombre**. Si intentas imprimir su valor desde afuera, te mostrará el error: **nombre is not defined**.

Podemos **cambiar el Tipo de dato** de una variable, sin que eso implique cambiar el contenido de la misma. Por ejemplo: una cadena de caracteres, o string, cambiarla a número o, viceversa.

Ejemplo # 3: Conversión de tipo de datos

```
1 let numero = 42;
2 let cadena = String(numero); //convierte un número a texto
3 //En ambas variables se almacena el mismo valor, pero el tipo de dato es diferente
4 console.log(typeof(numero)); //con typeof puedes averiguar el tipo de dato de cada variable.
5 console.log(typeof(cadena));
6
7 let texto = "24.5";
8 console.log(texto, "es de tipo:", typeof(texto));
9 let cantidad = Number(texto); //Convierte texto a número
10 console.log(cantidad, "es de tipo:", typeof(cantidad));
11 let entero = parseInt(texto); //Convierte a un entero
12 console.log("Ahora el texto es un entero", entero);
13 let decimal = parseFloat(texto);
14 console.log("Ahora el texto es un decimal", decimal);
```

```
number
string
24.5 es de tipo: string
24.5 es de tipo: number
Ahora el texto es un entero 24
Ahora el texto es un decimal 24.5
```

Veamos otro ejemplo con el tipo de dato objeto, específicamente relacionados con fecha, ya que es un tipo muy utilizado en el desarrollo web.

Ejemplo # 4: Calcular la edad

```
1 var fechaAct = new Date(); //Crea un objeto Date con la fecha y hora actual
2 console.log(fechaAct); //Imprime el objeto por consola
3 var fechaNac = new Date(1979,7,21); //Crea un objeto Date con el año, mes (0-11) y día señalado
4 console.log(fechaNac); //Imprime el objeto colocando 0 los parámetros no indicados (hora, minutos, Segundos)
5 console.log(fechaNac.getDate(), "/", fechaNac.getMonth()+1, "/", fechaNac.getFullYear()); //Imprime en el formato indicado
6
7 //Ahora si, calculemos la edad
8 var edad = fechaAct.getFullYear() - fechaNac.getFullYear(); //Extrae el año de ambas fechas, hace la resta y lo almacena en una variable
9 console.log("Tienes", edad, "años!"); //Imprime el resultado
```

Fri Oct 06 2023 11:00:08 GMT-0300 (hora estándar de Argentina)
Tue Aug 21 1979 00:00:00 GMT-0300 (hora estándar de Argentina)
21 / 8 / 1979
Tienes 44 años!

Ejemplo #5: ¿Cuántos días faltan?

Vamos a calcular cuántos días faltan para que finalice el 2023

```
1 var fechaActual = new Date();
2 var fechaFinal = new Date(2023,11,31); //Recuerda que el mes va desde 0 hasta 11
3 var diferencia = fechaFinal-fechaActual; //el resultado queda en milisegundos
4 //La función Math.ceil redondea a un número entero
5 var calcularDias = Math.ceil(diferencia / (1000 * 60 * 60 * 24)); //1000Ms*60s*60min*24h
6 console.log("Faltan " + calcularDias + " para que termine el 2023");
```

Faltan 91 para que termine el 2023