

Programas JavaScript

Autor: Juan Diego Vélez Sánchez

IS&C, Universidad Tecnológica de Pereira, Pereira, Colombia

Correo-e: juandiego.velez@utp.edu.co

Resume—Este documento presenta un resumen sobre las bases del lenguaje de programación JavaScript, uno de los lenguajes más utilizados y demandados, lo cual es indispensable aprenderlo, daremos a entender sobre las estructuras más importantes en la programación las cuales son el uso de variables, tipos de datos, bucles, mostrar mensajes y pedir datos al usuario.

Palabras clave— datos, bucles, variables, condición.

1. INTRODUCCION

JavaScript (JS) es un lenguaje de programación el cual está orientado a objetos, es implementado principalmente en las páginas web lo cual mejora la comunicación entre usuario y máquina, permitiendo que la página sea dinámica, todos los navegadores actuales soportan JavaScript.

Podemos importar el archivo.JS de maneras distintas:

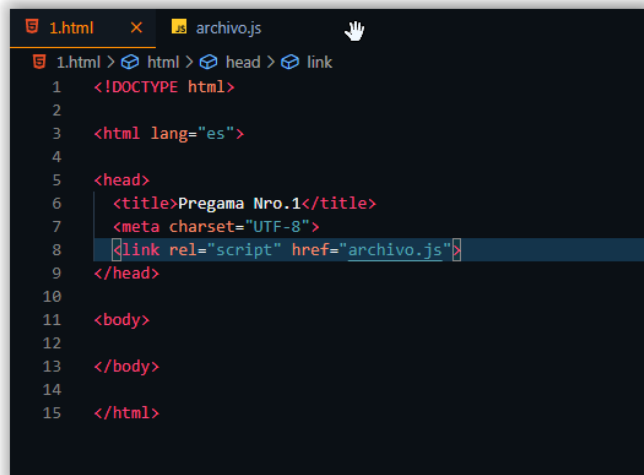


Figura 1. Introduciendo el archivo.js en el head de HTML.

Como podemos ver tenemos nuestra estructura HTML y en la parte del head introducimos con la etiqueta < link > el archivo correspondiente.

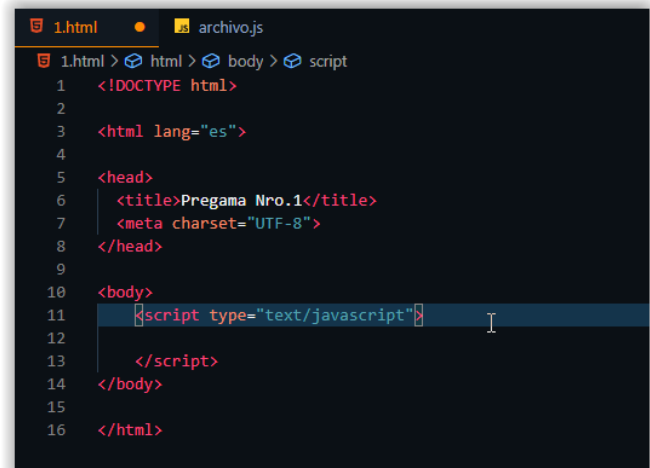


Figura 2. Introduciendo un script en el body de HTML.

En la figura 2 podemos apreciar que en el body introducimos una etiqueta < script > la cual dentro de esta se encuentra todo el código de JavaScript necesario.

PRAGRAMA Nro.1

En este programa nos permite mostrar un mensaje gracias a una función llamada alert, que recibe como parámetro el mensaje que se quiere imprimir en pantalla, este debe ir entre paréntesis para que el programa lo reconozca y el texto que queremos introducir debe ponerse entre comillas.

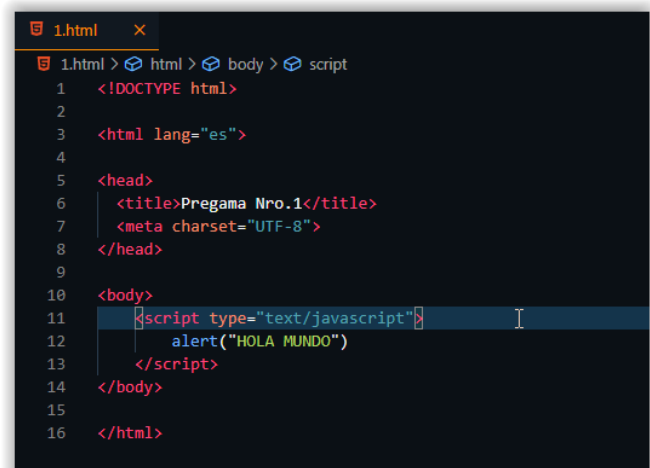


Figura 3. Código para mostrar un mensaje en pantalla.

El resultado en nuestra página web es el siguiente:

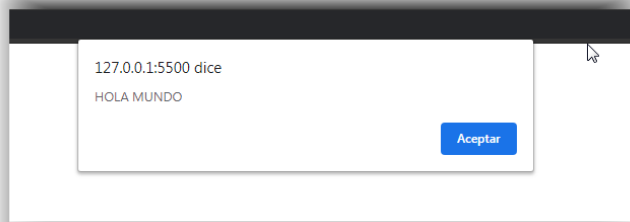


Figura 4. Resultado en el navegador tras ejecutar la función alert.

El resultado como podemos ver es una pequeña ventana, en el cual su contenido se observa que lo primero es la ruta de nuestra página con la palabra más la palabra dice, a continuación muestra nuestro mensaje que introducimos como parámetro más un botón que dice aceptar, el mensaje desaparecerá luego de dar click en el botón aceptar.

PRAGRAMA Nro.2

En el siguiente programa podemos observar que dentro de la etiqueta < Script > utilizamos una variable. Una variable es un tipo de dato almacenado en la memoria, el cual nos permite modificarlo durante la ejecución del programa, esta variable tiene asignada una función llamada prompt, la cual nos permite mostrar un mensaje y obtener un dato introducido por el usuario.

```

2.html > html > body > script
1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4    <meta charset="utf-8">
5    <title>Pregama Nro.1</title>
6  </head>
7  <body>
8    <script type="text/javascript">
9      var num = prompt("Ingrese un número: ");
10     alert("El número es: " + num);
11   </script>
12 </body>
13 </html>

```

Figura 5. Código el cual utiliza la función prompt.

Como podemos observar en el anterior código, utilizamos una variable la cual su sintaxis es, **var** más el nombre de esta, en este caso nuestra variable tiene como nombre num, esta contiene el dato introducido por el usuario gracias a la función prompt, dentro de esta hay un mensaje que se mostrara, en nuestro caso es el mensaje "Introduce un número" y esperara a que el usuario introduzca algo y le dé en aceptar.

Luego con la función alert mostramos el dato que introdujo el usuario, el signo + es para concatenar el mensaje y la variable que queremos que se muestre, la variable no está entre comillas pero si dentro de los paréntesis.

El resultado es el siguiente:

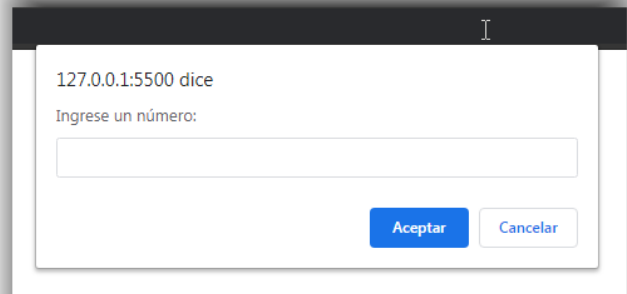


Figura 6. Ejecución de la función prompt.

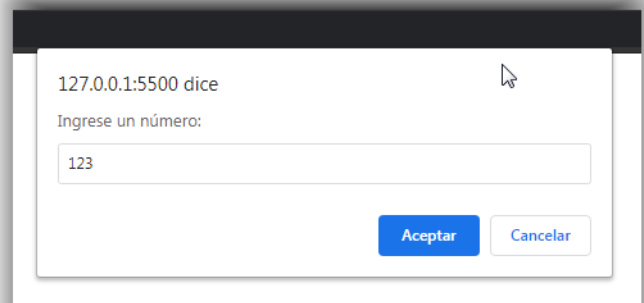


Figura 7. Entrada de dato por parte del usuario.

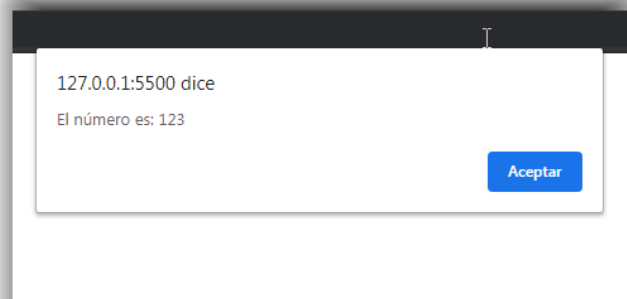


Figura 8. Función alert con el mensaje y el dato que introdujo el usuario.

Como podemos observar la función prompt nos arroja una ventana la cual contiene el mensaje y un cuadro de texto para que el usuario pueda introducir un dato, aparece dos botones, el botón de aceptar y cancelar, si dejamos el cuadro de texto vacío no mostrara nada, en cambio si le damos cancelar la variable retorna un valor null o sea que no contiene nada.

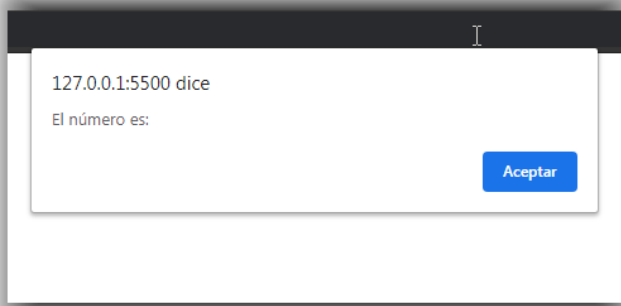


Figura 9. Resultado tras no poner nada en el cuadro de texto y darle en aceptar.

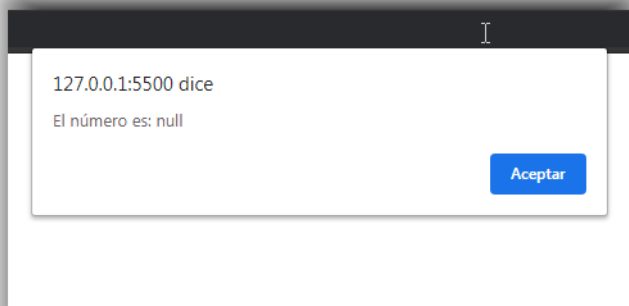


Figura 10. Resultado tras darle en la opción cancelar.

PRAGRAMA Nro.3

El siguiente programa permitirá que el usuario pueda introducir dos números, los cuales posteriormente serán sumados en otra variable y mostrados en pantalla por medio de la función alert.

```

1 <!DOCTYPE html>
2 <html lang="es">
3 <head>
4   <meta charset="utf-8">
5   <title>Suma</title>
6 </head>
7 <body>
8   <script type="text/javascript">
9     var str1 = prompt("Ingrese un número: ");
10    var str2 = prompt("Ingrese otro número: ");
11
12    var num1 = parseFloat(str1, 10);
13    var num2 = parseFloat(str2, 10);
14
15    var suma = num1 + num2;
16
17    alert("Suma = " + suma);
18  </script>
19 </body>
20 </html>

```

Figura 11. Código para sumar dos números.

En el anterior código podemos observar lo siguiente:

Inicializamos dos variables las cuales permitirán almacenar el dato introducido por el usuario con la función prompt.

Continuando podemos ver que hay dos variables con la función parseFloat, esta permitirá convertir números de tipo cadena de texto a números utilizables este puede tener parte decimal, esta función tiene como parámetros dentro de los paréntesis la variable a convertir.

Los números que vamos a introducir para el ejemplo es 5 y 7, la salida de este código es:

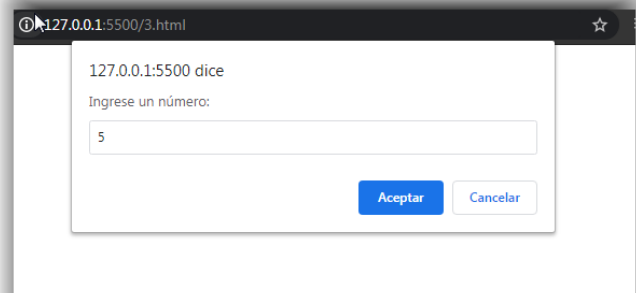


Figura 12. Primer prompt asignado a la variable str1.

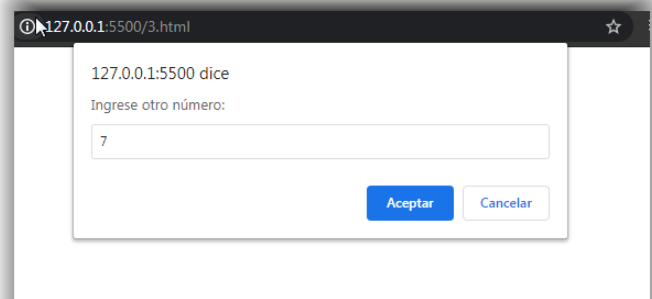


Figura 13. Segundo prompt asignado a la variable str2.

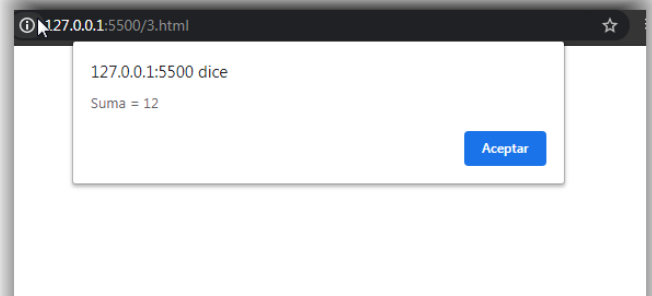


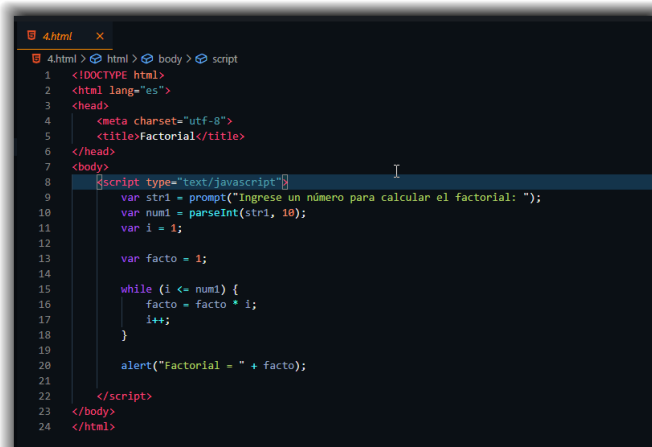
Figura 13. Ejecución del alert tras convertir los datos y calcular la suma de estos, mostrándolo en pantalla.

PRAGRAMA Nro.4

En este programa vamos a ver como se utiliza un pilar de la programación, el cual es el uso de bucles. Los bucles que podemos encontrar en la mayoría de lenguajes de programación son: while, do while, for y en algunos for each.

En el siguiente código podemos observar que utilizamos conceptos ya antes vistos, variables, funciones para mostrar y pedir datos al usuario, pero también entra algo nuevo llamado while, el cual es uno de los bucles que podemos utilizar. La estructura de este es:

Llamamos al bucle y entre los paréntesis pondremos una condición, en este de que el valor en la variable “i” sea menor o igual al número introducido por el usuario, y entre llaves estará almacenado el código que queremos ejecutar un número determinado de veces.



```

1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4    <meta charset="utf-8">
5    <title>Factorial</title>
6  </head>
7  <body>
8    <script type="text/javascript">
9      var str1 = prompt("Ingrese un número para calcular el factorial: ");
10     var num1 = parseInt(str1, 10);
11     var i = 1;
12
13     var facto = 1;
14     while (i <= num1) {
15       facto = facto * i;
16       i++;
17     }
18
19     alert("Factorial = " + facto);
20   </script>
21 </body>
22 </html>

```

Figura 14. Código del bucle while para calcular el factorial de un número.

En el código podemos ver que le pide un número al usuario para calcular su factorial, luego de pedir el número lo aplicamos en la función parseInt, la cual es parecida a la función parseFloat solo que esta convierte el número de texto a número utilizable sin parte decimal. Inicializamos la variable i que nos permitirá contar las veces que se repetirá el bucle y la variable facto que contendrá el resultado para luego mostrarla en la pantalla.

Para el ejemplo introducimos el 5, el resultado es el siguiente:

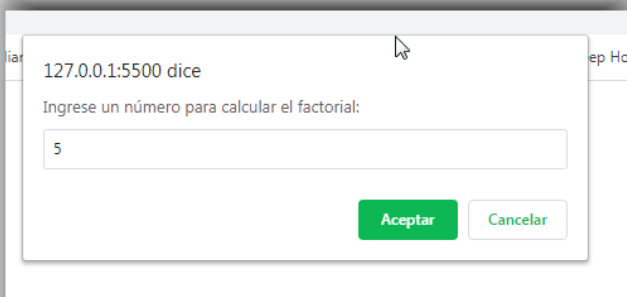


Figura 15. Introducción de datos.

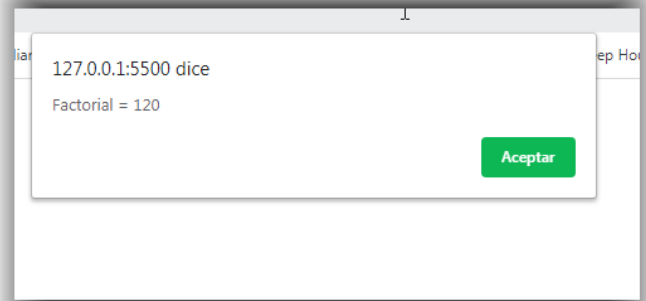


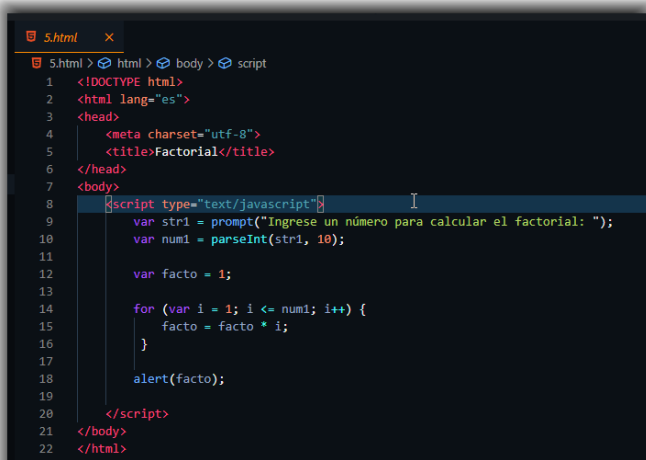
Figura 16. Resultado tras ejecutar el código.

Internamente el código funciona así:

Al introducir como numero el 5 este se le asignara al variable str1 y convertido a número entero en la variable num1, inicializamos i y facto, eventualmente entraremos al bucle while, en el cual el código hará que facto se le asigne el resultado tras multiplicar facto por i en este caso 1 por 1, en la línea 17 de nuestro código podemos observar que delante de la variable i hay dos signos + los cuales permiten aumentar a esta en uno en uno, lo cual valdrá 2 en el próximo ciclo, después de aumentar la variable i volvemos al principio de nuestro bucle para evaluar la condición, en este caso si se cumple ya que 2 es menor e igual a 5, por lo cual facto será el resultado entre facto que es 1 e i que es 2 y posteriormente i se aumentara en uno lo cual valdrá 3, y así sucesivamente hasta que i sea 6 ya que en ese punto no cumplirá con la condición y saldrá del bucle para ejecutar el resto del código, que en nuestro caso es la salida de pantalla de la variable facto.

PRAGRAMA Nro.5

El programa 5 es igual al 4 solo que en este utilizamos el bucle for.



```

1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4    <meta charset="utf-8">
5    <title>Factorial</title>
6  </head>
7  <body>
8    <script type="text/javascript">
9      var str1 = prompt("Ingrese un número para calcular el factorial: ");
10     var num1 = parseInt(str1, 10);
11
12     var facto = 1;
13
14     for (var i = 1; i <= num1; i++) {
15       facto = facto * i;
16     }
17
18     alert(facto);
19   </script>
20 </body>
21 </html>

```

Figura 17. Código del bucle for.

Resultado será el mismo que en la figura 15 y 16, e internamente similar, utilizaremos como entrada el número 5 de nuevo para el ejemplo, el bucle for funciona de la siguiente manera.

Llamamos al bucle y dentro de los paréntesis la variable que vamos utilizar para contar los ciclos en nuestro caso `i` y la inicializamos en 1, la condición para que este se ejecute la cual es que `i` sea menor o igual a 5, el aumento de la variable que ente caso es `++` o mejor dicho en uno en uno, o sea que `i` que vale 1 en el próximo ciclo valdrá 2 y así sucesivamente hasta que valga 6.

Dentro del bucle está el código para calcular el factorial, el cual se le asigna a `facto` el resultado de la multiplicación entre `facto` e `i` que es 1 por 1, después `i` será 2 y `facto` será 1 por 2, después `i` será 3 por lo cual `facto` será 2 por 3 y así sucesivamente hasta que la condición no se cumpla, para luego que salga del ciclo for muestre el resultado por medio del `alert`.