

Programas básicos en el html

Basic programs in html

Autor: Laura Quintero Montoya

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

Correo-e: l.quintero2@utp.edu.co

Resumen— Este documento presenta un resumen de los principales programas básicos que se maneja en el html, como agregar un texto, hacer sumas, y dos formas de hacer la operación factorial de un número, todo eso por medio de un lenguaje de programación llamado html.

Palabras clave— html, head, body, código, página, javascript, prompt, scrip, alert, factorial, var, str, i.

Abstract— this document presents a summary of the main basic programs that are handled in html, such as adding a text, doing sums, and two ways of doing the factorial operation of a number, all through a programming language called html.

Key Word— html, head, body, code, page, javascript, prompt, scrip, alert, factorial, var, str, i.

I. PROGRAMA DE TEXTO

En el siguiente programa vamos a leer una oración “HOLA MUNDO”, que mediante un alert, la página nos muestra la oración.

Presentamos a continuación, las interfaces de captura de datos y posterior ejecución de la página.

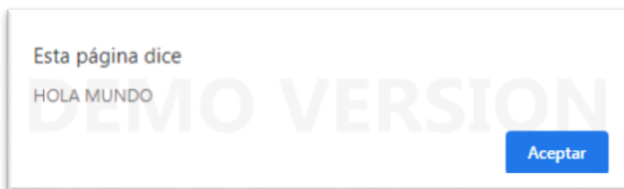


Figura 1. Captura del mensaje.

El código fuente que realiza el proceso, es el siguiente:

```

1  <!DOCTYPE html>
2
3  <html lang="es">
4
5  <head>
6    <title>Output</title>
7    <meta charset="UTF-8">
8  </head>
9
10 <body>
11
12   <script type="text/javascript">
13     alert('HOLA MUNDO');
14   </script>
15
16 </body>
17
18 </html>

```

Figura 2. Código fuente del programa de texto

En este código el **HTML** se le coloca “**lang**” que significa lenguaje y después un signo “=” y el “es” que significa que página está en español, se divide en dos partes: **head y body**; ya al finalizar estas se cierra el html

En el **HEAD**, que significa cabecera, son definiciones, este lleva dos etiquetas, el “**title**” que es el título de la página que va encabezado y el “**meta**” que significa va una serie de caracteres con el símbolo “**charset**”, que describe el “UTF” “Formato de Transformación Unicode” (en inglés) y “8” como el espacio en bits.

En el **BODY**, ahí se ve la página vacía, sin embargo, se encuentra una etiqueta llamada “**scrip**” nos determina el tipo de lenguaje que se va utilizar, en este caso es el “**javascript**”, este tiene la funcionalidad de manejar eventos, para responder desde que la pagina inicia, finaliza y reinicia, y es posible escribir dentro de javascript un programa completo; en este caso tiene una instrucción “**Alert**” que significa abrir una ventana y dentro esa la ventana escribir lo que se encuentra en comilla (en ese caso es HOLA MUNDO); eso se representa en la figura 1.

II. PROGRAMA DE DETECTOR DE NÚMERO

En el siguiente programa vamos a leer una oración “ingrese un número”, mediante un alert, después al ingresar el número elegido, la página carga rápidamente y nos muestra ese número diciendo “**El número es: n**”

Se presentamos a continuación, las interfaces de captura de datos y posterior ejecución de la página.

Figura 3. Captura del número.

Figura 4. Captura del resultado.

El código fuente que realiza el proceso, es el siguiente:

```

1  <!DOCTYPE html>
2
3  <html lang="es">
4
5  <head>
6      <meta charset="utf-8">
7      <title>Input</title>
8  </head>
9
10 <body>
11     <script type="text/javascript">
12
13         var num = prompt("Ingrese un número: ");
14         alert("El número es: " + num);
15
16     </script>
17 </body>
18
19 </html>

```

Figura 5. Código fuente del programa de detector de número.

En este código en el **HTML** no cambia nada al anterior, mientras que en el **HEAD** lo cambia su título por “**Input**”.

Lo que cambia es el **BODY**, tiene el mismo lenguaje que el anterior programa que es “**javascript**”, cambia sus instrucciones, ya sale uno nuevo que es “**var**” y “**num**” que

significa variable, una casilla donde es como una gaveta donde cabe algo y se le asigna un valor cualquiera. Su otra etiqueta que está a lado es “**prompt**” que indicará que está a la espera de órdenes, en ese caso su operación sería indicar el número que se colca en el sistema. En ese caso se colca “ingrese un número”, y al colocar algún número, se muestra el alert mostrando el número asignado. Por tal motivo se coloca la etiqueta “**alerte**” con la oración (“**El número es**” +**num**), que este es el número elegido por el usuario. Y estos tienen comillas porque estos son cadenas. Se cierra script, body y el html.

De esta manera se ejecuta el programa como se muestra en las capturas.

III. PROGRAMA DE SUMA

En el siguiente programa vamos a leer dos números por teclado, vamos a extraer su parte entera y seguidamente realizaremos la suma.

Una vez sumados los números, procederemos a mostrar mediante un alert, el valor de dicha suma.

Presentamos a continuación, las interfaces de captura de datos y posterior ejecución de la suma.

Figura 6. Captura del primer número.

Figura 7. Captura del segundo número.

Figura 8. Captura del resultado.

El código fuente que realiza el proceso, es el siguiente:

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

Figura 9. Código fuente del programa de suma.

En este código en el **HTML** no cambia nada al anterior, mientras que en el **HEAD** lo cambia su título por “suma”.

Lo que cambia es el **BODY**, tiene el mismo lenguaje que el anterior programa que es “**javascript**”, cambia sus instrucciones, se abre la etiqueta de variable, e ingresar un número, este número se va guardar al “**str1**”. Pero esos número que ingresamos en el **str1** y **str2** son cadenas, por lo tanto, para que esas cadenas se vuelva números, debe recurrir una función de javascript, que se llama “**parseFloat**”, que significa número con decimales, este recibe el **str1** y **str2** en base 10, que es un sistema común. Todo eso se guarda en “**num1**” y “**num2**” respectivamente. Con eso se puede crear una nueva variable que es “**suma**”, para realizar esa dicha operación colocamos “**suma= num1 + num2;**”; después se coloca la etiqueta “**alert**” se vuelve cadena el resultado de la suma; se cierra script, body y el html. Termina y refresca la página.

De esta manera se ejecuta el programa como se muestra en las capturas.

IV. PROGRAMA FACTORIAL (PRIMERA FORMA)

En el siguiente programa vamos a leer un número por teclado, vamos a extraer su factorial y seguidamente realizar esa operación con su respuesta. Presentamos a continuación, las interfaces de captura de datos y posterior ejecución de la factorial.

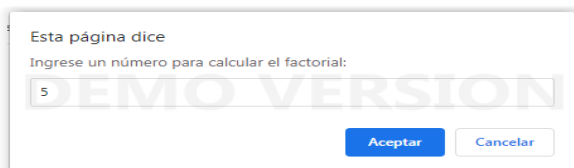


Figura 10. Captura del número

Y posterior al aceptar, se da su resultado.



Figura 11. Resultado de la factorial.

El código fuente que realiza el proceso, es el siguiente:

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="utf-8">
  <title>Factorial</title>
</head>
<body>
  <script type="text/javascript">

    var str1 = prompt("Ingrese un número para calcular el factorial: ");
    var maximo = parseInt(str1, 10);

    var i = 1;
    var factorial = 1;

    while (i <= maximo)
    {
      factorial = factorial * i;
      i = i + 1;
    }

    alert("Factorial de " + maximo + " es " + factorial);

  </script>
</body>
</html>
```

Figura 12. Código fuente del programa factorial primera forma.

En este código en el **HTML** no cambia nada al anterior, mientras que en el **HEAD** lo cambia su título por “factorial”.

En el **BODY** se hace todo igual como la suma, sin agregar “**str2**”, ya que es un solo valor lo que vamos preguntar al programa. El “**max**” es el límite superior a donde pueda llegar el programa, se le agrega “**parseInt (str1, 10);** para saca el valor numérico como un entero. Después se le añade “**Var i=1;**” y “**var factorial=1**” debido a que todos los factoriales comienza en 1.

Se abre una nueva etiqueta llamada “**while**” que significa mientras; a lado de este se le añade “**(i <= máximo)**” indicando al programa que es igual o mayor a 5; y abre llaves y colocamos “**{factorial = factorial * i; i = i + 1;}**” iniciando un ciclo y el mismo programa detecte ese ciclo. Se le completa con “**alert "Factorial de " + máximo + " es " + factorial;**” que indica el resultado de la operación. Se cierra body y el html. Termina y refresca la página.

V. PROGRAMA FACTORIAL (SEGUNDA FORMA)

En el siguiente programa es exactamente el mismo, lo único que cambia son sus códigos. Vamos a leer un número por teclado, vamos a extraer su factorial y seguidamente realizar esa operación con su respuesta.

Presentamos a continuación, las interfaces de captura de datos y posterior ejecución de la factorial.

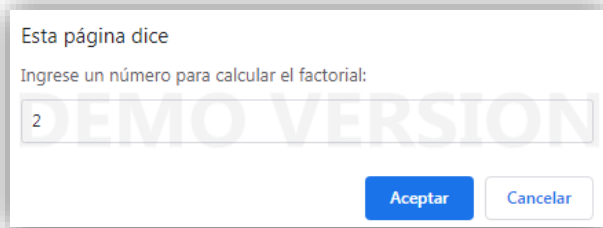


Figura 13. Captura del número para calcular el factorial.

Y posterior al aceptar, se da su resultado.



Figura 14. Resultado de la factorial.

El código fuente que realiza el proceso, es el siguiente:

```

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 maximo = parseInt(str1, 10);
11
12     var factorial = 1;
13
14     for (var i = 1; i <= maximo; i = i + 1)
15     {
16       factorial = factorial * i;
17     }
18
19     alert("El factorial de " + maximo + " es " + factorial);
20
21   </script>
22 </body>
23 </html>

```

Figura 15. Código fuente del programa factorial segunda forma.

En este código en el **HTML** no cambia nada al anterior, igual que en el **HEAD**.

En el **BODY**, casi todo es igual al anterior, lo único diferente es que se quita la ética **"White"** con **"for"**, que significa para para, y se le añade **"(var i = 1; i <= maximo; i = i + 1)"** y se abre las llaves **"{factorial = factorial * i;}"**, todo se hace igual al anterior programa y su operación es la misma.