Desarrollo de Aplicaciones Web

Tema Nº4:JAVASCRIPT - I

Indicador de logro Nº4:Aplica los fundamentos de programación, a través de JavaScript para el agregado de características interactivas a los sitios web

**TEMA 01 Teoría de los**

Imagen que contiene Icono

Descripción generada automáticamente

**TEMA Nº4:**

JAVASCRIPT - I

**Subtema 1.1:**

¿Qué es JavaScript?

**MARCO TEÓRICO**

**JavaScript** (comúnmente abreviado **JS**)se usa para añadir características interactivas a tu sitio web, por ejemplo: Juegos, eventos que ocurren cuando un botón es presionado, efectos de estilo dinámicos, animaciones y mucho más.

Es un robusto lenguaje de programación que se puede aplicar a un documento HTML y usarse para crear interactividad dinámica en el sitio web. Fue inventado por **Brendan Eich**, cofundador del proyecto Mozilla, Mozilla Foundation y Corporación Mozilla.



**JavaScript** se usa principalmente del lado del cliente (es decir, se ejecuta en nuestro ordenador, no en el servidor) permitiendo crear efectos atractivos y dinámicos en las páginas web. Los navegadores web ya vienen con un intérprete de código **JavaScript.**

**JavaScript** no necesita de un compilador, sino se ejecuta en tiempo real mediante el intérprete integrado en el navegador web, que traduce o interpreta el código en tiempo real. También es un **lenguaje orientado a objetos**, abarca todo las características, conceptos y posibilidades que posee la **programación orientada a objetos**.

Es un lenguaje imperativo, es decir, primero se ejecuta la primera instrucción, después la segunda y así sucesivamente, no como otros lenguajes que necesitan que todo el código esté listo para poder recién ejecutarse.

La ventaja de este lenguaje es que al estar alojado en el ordenador del usuario los efectos son rápidos y dinámicos. Al ser un lenguaje de programación permite toda la potencia de la programación como el uso de variables, condicionales, bucles, etc.

**Subtema 1.2:**

Sintaxis

Este lenguaje está altamente inspirado en los lenguajes de programación **C y Java**, por lo cual la sintaxis es muy parecida a estos dos últimos lenguajes.

A diferencia de otros lenguajes no es necesario, pero es lo más recomendable ya que te evita solucionar muchos problemas, usar el **punto y coma** (**;**) al final de cada línea de código, pero si se debe usar cuando quieres escribir varias sentencias de código en una misma línea

La sintaxis de **JavaScript** define un conjunto de reglas que deben seguir para escribir código correctamente, y darle así instrucciones al programa en el que estamos trabajando, ejecutando la tarea especificada en dicho código. Si la **sintaxis** está mal escrita o si no se respetan las reglas establecidas por la **sintaxis**.

Las características de la sintaxis de **JS** son las siguientes:

* El **Intérprete** de **JavaScript** no toma en cuenta los espacios en blanco y las nuevas líneas, es decir, ignora cualquier espacio en blanco sobrante.



* Es **Case Sensitive**, es decir, es sensible a mayúsculas y minúsculas, no es lo mismo definir una variable **“numero” a “Numero”**, **JavaScript** lo tomará como dos variables diferente y se podrá usar sin ningún problema



* No se define el tipo de variable al crear una variable, es decir, no es necesario indicar el tipo de dato que se almacenara como en otros lenguajes que es necesario definir si es Integer o String. La misma variable puede almacenar varios tipos de datos durante la ejecución del script.



**Subtema 1.3:**

Variables

Una variable es un espacio que se guarda en memoria dentro de nuestro programa, para volver a usarlo después y, también, puede cambiar dinámicamente.

**Tipos de Datos**

Las variables pueden almacenar varios tipos de datos, que son:

* **String:** Usar comillas dobles o simples indica al JavaScript que este valor es de tipo de dato String.
* **Number:** En este caso no se usa comillas, porque es otro tipo de dato.
* **Bool:** Los valores booleanos es solo 0 y 1, que traduciéndolo sería False o True, respectivamente.

Las variables pueden ser declaradas, inicializadas y modificadas.

**Declarando una variable**

Para declarar una variable no le tenemos que asignar un valor, al asignarle un valor la estamos inicializando.

Una Variable puede ser declarado de 3 formas: **Var**, **Let** y **Const**

* Una variable **Const** o **Constante** no puede cambiar su valor, solo se define una vez y no se puede volver a definir nunca más. Además, este tipo de variable no puede ser declarada y después inicializada, tiene que ser declarada y a la vez inicializada
* Una variable **Let** a diferencia del **Const** si puede ser modificado (cambiar su valor) después de haber sido inicializado y, también, se le puede modificar el valor. Además, una **variable** declarada con **let** solo existe en el bloque de código en el que se encuentre, mas no en los otros
* Una variable **Var** a diferencia del **let** puede existir en cualquier parte del código, es decir, es una variable global y, además, se puede cambiar el valor, de la variable, cuando nosotros deseamos, cosa que no se puede hacer con el **const**

**Subtema 1.4:**

Operadores

Hay dos tipos de operadores, los **Aritméticos** y **Asignación**

**Operadores Aritméticos**

Un operador aritmético toma valores numéricos (ya sean literales o variables) como sus operandos y devuelve un solo valor numérico. Los más comunes son: **Suma (+), resta (-), multiplicación (\*) y división (/)**

**Operadores de Asignación**

Un operador de asignación asigna un valor al operando de la izquierda basado en el valor del operando de la derecha. Los más comunes son:

**Asignación (x = y)**

**Asignación de adición (x += y)**

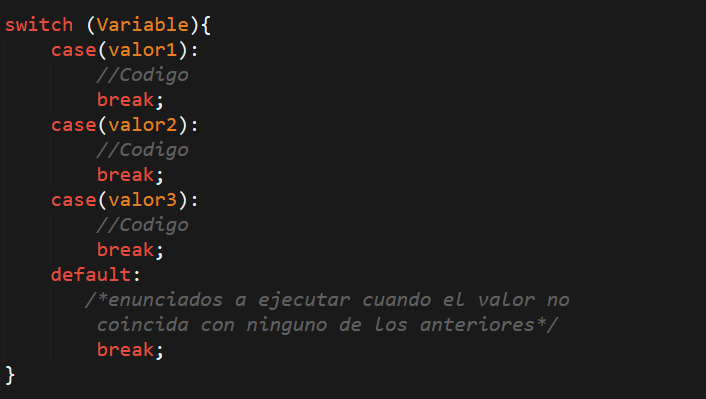
**Asignación de sustracción (x -= y)**

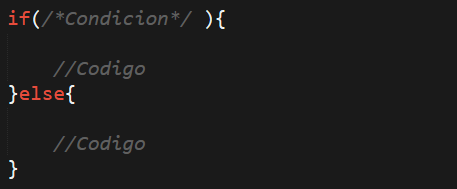
**Subtema 1.4:**

Condicionales y Estructuras

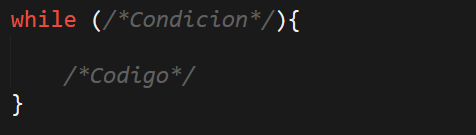
Una condicional es una sentencia que nos permite validar algo. Existen dos tipos de condicionales, que son: **if** y **switch**

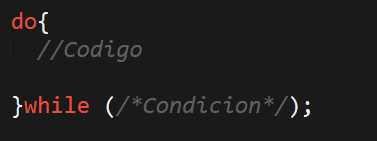
La estructura de las condicionales es la misma que en otros lenguajes de programación

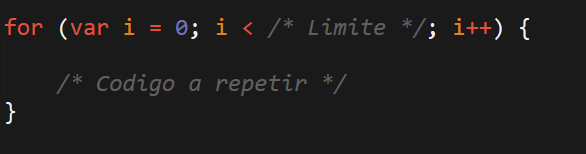




Las estructuras son **for**, **do while** y **while**, que nos permiten de forma rapida y sencilla hacer algo repetidamente.







**1. MATERIALES**

Para la experiencia a realizar se requiere lo siguiente:

1. EQUIPO

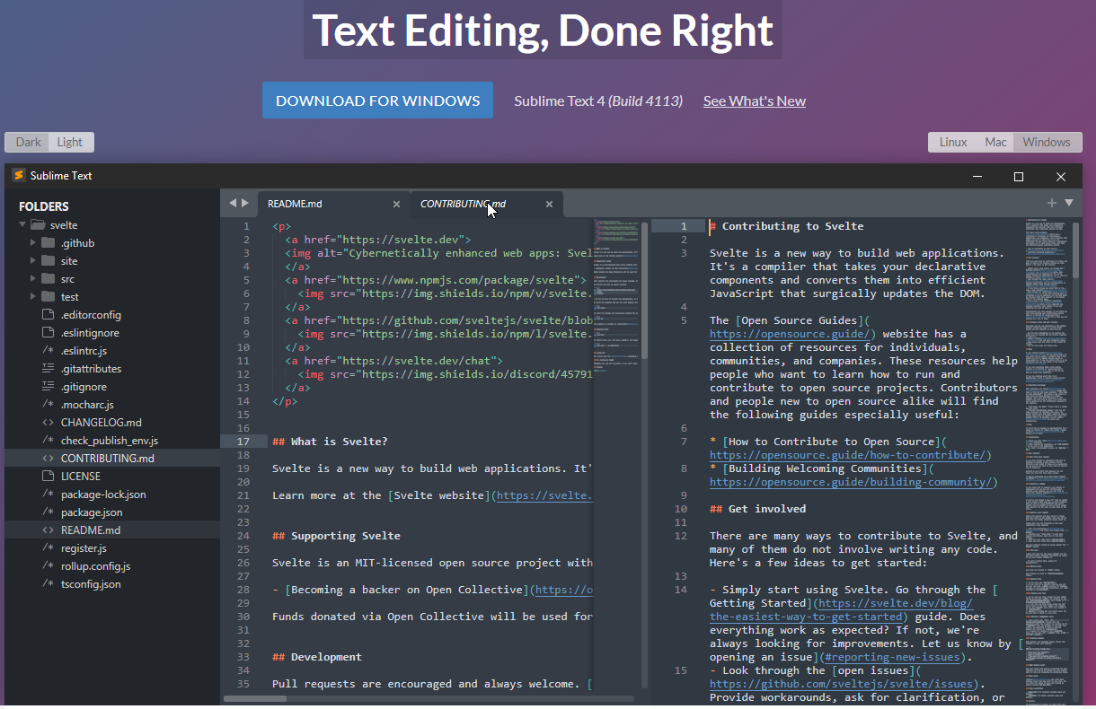
Se requiere tener un equipo de cómputo (PC o laptop).

1. HERRAMIENTAS:

* Sublime Text 3

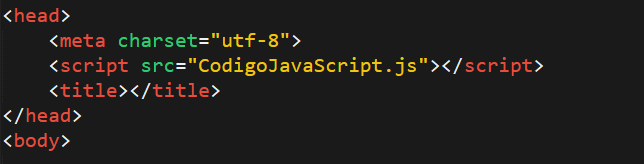
**2. PROCEDIMIENTO**

Se creará un nuevo archivo, en el **Sublime Text**, en donde se aplicará lo aprendido en esta guía.

Herramienta de trabajo:

Se creará dos archivos, un archivo **HTML** llamado **JavaScript\_Guia4.html** y un archivo **JAVASCRIPT** llamado **CodigoJavaScript.js**.

Para crear los archivos se puede ir a la pestaña **File** **>> New File** o podemos usar las teclas **CRTL + N**. Para guardar los archivos hacemos uso de las teclas **CRTL + S**. En el archivo **.html** se hará el enlace entre dicho archivo y el archivo **.js**, de esta manera:



**Ejemplo 1:** Se crearán las etiquetas a usar dentro del HTML.

<body>

<h1>Pagina web de una Heladería que vende helados</h1>

<h2>Lista de helados</h2>

<label>

Opción 1: Helado de crema marca Heladovich

</label>

S/<input type="number" Id = "Opcion1" readonly="true"

value = "1.50"><br><br>

<label>

Opción 2: Helado tricolor marca Helardo

</label>

S/<input type="number" Id = "Opcion2" readonly="true"

value = "2.50"><br><br>

<label>

Opción 3: Helado de vainilla marca Heladosky

</label>

S/<input type="number" Id = "Opcion3" readonly="true"

value = "1.00"><br><br>

--------------------------------------------------------------------<br><br>

<label>

Ingresa la cantidad de dinero que tengas:

</label>

S/<input type="number" Id="DineroID"><br><br>

<label>

Escoge una de las 3 opciones:

</label>

<input type="number" Id="OpcionID">

<button type="button" onclick="Calcular()" >Calcular</button> <br><br>

--------------------------------------------------------------------<br><br>

<label>Puedes comprar: </label>

<input type="text" Id="ResultadoID" readonly="true"><br><br>

<label>Tienes un vuelto de: S/</label>

<input type="text" Id = "VueltoID" readonly="true">

</body>

Para poder ejecutar este programa haremos uso de **function** o **funciones**, si en otros lenguajes se usan **métodos** en **JavaScript** se usa **functions**. El **Math.trunc()** sirve para obtener la parte entera de un decimal y, también, se usa el **document.getElementByID('').value** para atrapar el value o valor de una etiqueta html que tenga dicho ID.

function Calcular(){

//Se obtiene el dinero que ingresaste

var Dinero = document.getElementById('DineroID').value;

//Se obtiene la opción que ingresaste

var Opcion = document.getElementById('OpcionID').value;

//Se obtiene el precio de la opción 1

var Opcion1 = document.getElementById('Opcion1').value;

//Se obtiene el precio de la opción 2

var Opcion2 = document.getElementById('Opcion2').value;

//Se obtiene el precio de la opción 3

var Opcion3 = document.getElementById('Opcion3').value;

var Resultado = 0;

var Vuelto = 0;

if( (Dinero < 1.50) && (Dinero > 0) ){

document.getElementById('ResultadoID').value = "No te alcanza para nada";

}

else if( (Dinero == 0) || (Dinero == null) ) {

document.getElementById('ResultadoID').value = "Ingresa una cantidad mayor a 0";

}

else

{

switch(Opcion){

case('1'):

Resultado = Math.trunc(Dinero / Opcion1);

Vuelto = Dinero % Opcion1;

//Se plasma el valor del Resultado al input con el Id “ResultadoID”

document.getElementById('ResultadoID').value =

"Puedes comprar "+Resultado+" helados";

document.getElementById('VueltoID').value = Vuelto;

break;

case('2'):

Resultado = Math.trunc(Dinero / Opcion2);

Vuelto = Dinero % Opcion2;

document.getElementById('ResultadoID').value =

"Puedes comprar "+Resultado+" helados";

document.getElementById('VueltoID').value = Vuelto;

break;

case('3'):

Resultado = Math.trunc(Dinero / Opcion3);

Vuelto = Dinero % Opcion3;

document.getElementById('ResultadoID').value =

"Puedes comprar "+Resultado+" helados";

document.getElementById('VueltoID').value = Vuelto;

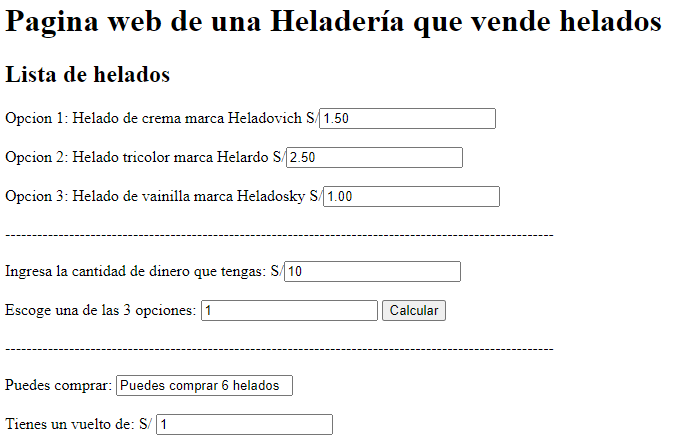
break;

}

}

}

**Resultados:**



**Ejemplo 2:** Se hará una pirámide usando los valores que están en el rango de 1 hasta un valor ingresado a un input, en nuestro caso será del 1 al 10.

<body>

<h2>Mostrar una pirámide desde el numero 1 hasta el numero ingresado</h2>

<input type="number" Id = "NumeroID"><br><br>

<button type="button" onclick="Mostrar()" >Mostrar</button> <br><br>

</body>

function Mostrar(){

var Numero = document.getElementById('NumeroID').value; //Se obtiene el numero ingresado

if (Numero > 0 && Numero <= 20) {

var rep; //Declaramos la variable rep

var i; //Declaramos la variable i

for (i = 0; i <= Numero; i++) { //En este for ira del numero 1 hasta el numero ingresado

//También aquí se crearán filas en la que estarán los datos

//del segundo for

for (rep = 1; rep <= i; rep++) {

document.write(rep); //Mostrara una fila de todos los números hasta el máximo

//numero, ósea, i. Si i es 2 se creará dos filas, en la

//primera se mostrará el 1 y en la segunda se mostrará: 1 y 2

}

document.write("<br>");

}

}

else {

alert("El número tiene que ser menor a 20");

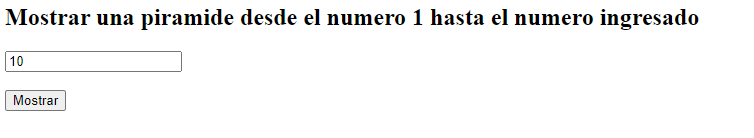
}

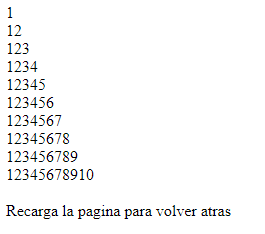
document.write("<br>");

document.write("Recarga la pagina para volver atras");

}

**Resultado:**





Todos los resultados deben mostrar el código de estado HTTP de modo satisfactorio.

**ACTIVIDAD VIRTUAL:**

1. **CUESTIONARIO TÉCNICO**

Revisa y analiza el tema desarrollado en la presente sesión, luego responde las siguientes preguntas propuestas:

* ¿Qué es JavaScript?
* ¿Para qué usar JavaScript?
* ¿Cuántos y cuáles son los tipos de operadores?
* ¿En qué se diferencia el Switch con el if?
* Realice una App similar aplicando todo lo aprendido de esta guía.

1. **ENLACES DE REFERENCIA**

* https://www.aprenderaprogramar.com/index.php?option=com\_content&view=article&id=590:ique-es-y-para-que-sirve-javascript-embeber-javascript-en-html-ejercicio-ejemplo-basico-cu00731b&catid=69&Itemid=192
* https://devcode.la/tutoriales/sintaxis-en-javascript/
* https://developer.mozilla.org/es/docs/Learn/Getting\_started\_with\_the\_web/JavaScript\_basicshttps://www.nextu.com/blog/que-es-json/
* http://www.codexexempla.org/curso/curso\_4\_2\_c.php
* https://developer.mozilla.org/es/docs/Web/JavaScript/Guide/Loops\_and\_iteration

1. **CONCLUSIONES DE LA EXPERIENCIA**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_