- ¿Qué es Javascript?

- ¿Qué es ECMAScript?

Ecma es un organismo que estandarizó el uso de de javascript para que fuera independiente de cualquier empresa. Javascript es la implementación de ese estándar.

- Ejecutando código JavaScript en el navegador

- Sintaxis

\*\*Algoritmo\*\*

- Método para resolver un problema que consiste en la realización de un conjunto de pasos lógicamente ordenados que nos permiten resolver un problema o tomar una decisión.

Entrada => Procesos => Salida

\*\*1 - Definición del problema :\*\*

Es donde se determina que es lo que debe resolverse con el computador, el cual requiere una definición clara y precisa

\*\*2 - Análisis del problema:\*\*

los datos de entrada

los datos de salida

los métodos y formulas que se necesitan para procesar los datos

\*\*3 - Diseño del algoritmo: \*\*

Este proceso consiste en definir la secuencia de pasaos que se deben llevar a cabo para conseguir la salida identificada en el paso anterior

\*\*4 - Codificación:\*\*

Escribir la solución del problema en una serie de instrucciones detalladas en un código reconocible por la computadora. La serie de instrucciones detalladas se conoce como código fuente, el cual se escribe en un lenguaje de programación o lenguaje de alto nivel

\*\*5- Prueba y depuración:\*\*

Se denomina prueba de escritorio a la comprobación que se hace de un algoritmo para saber si esta bien realizado. Esta prueba consiste en tomar datos específicos como entrada y seguir la secuencia indicada en el algoritmo hasta obtener un resultado, el análisis de estos resultados indicara si el algoritmo esta correcto o si por el contrario hay necesidad de corregirlo o hacerle ajustes

\*\*Que es JavaScript:\*\*

--------------------

Es un lenguaje de programación que sirve para crear paginas web dinámicas que surge en los años 90'.

- Es un lenguaje interpretado los navegadores son los encargados de leer e interpretar el código para su funcionamiento

Formas de Ejecutar codigo JS

--------------------------------

\*\*En el navegador\*\* A traves de la consola (presionando f12 en el navegador) y en el apartado console ejecutar lo que seria codigo js

\*\*En el HTML\*\* ya sea en la ejecución a través de un boton o dentro de la estiqueta <script>

\*\*En un Archivo separado JS:\*\* Es un archivo externo con extencion JS que luego se linkea al html con la etiqueta <script>

\*\*Variables:\*\* pueden definirse de tres formas

--------------------------------------

Es un espacio de memoria donde guardo datos:

`let nombre\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_=\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"Pablo"`

`definir(let nombre)\_\_\_\_\_asignar(=)\_\_\_\_\_\_\_\_\_\_\_valor "Pablo" `

\*let:\* Crea variables de entorno, solo existen donde están definidas y no por fuera (el scope)

{

let variable = "Prueba"

si llamo a variable si me devuelve el valor Prueba

}

si llamo a variable no me devuelve el valor Prueba

\*var:\* Crea variables globales que existen dentro y fuera de donde estan definidas Ejemp

{

var variable = "Prueba"

variable me devuelve el valor Prueba

}

variable me devuelve el valor Prueba

\*const:\* Crea variables en las que su contenido no puede modificarse pero si se le puede incluir mas información. su scope es igual al de let

\*\*Tipos de Datos\*\*

----------------------

String: son las cadenas de texto se definen entre "" comillas "hola" o "32" ese numero es un string porque esta dentro de " "

Number: son los valores numericos

Boolean: devuelve un valor verdadero o falso

Array: los arreglos es una colección de de objetos o valores que esta definido dentro de corchetes [ ] Ejem let arreglo = [1, 2, "auto", true]

Object: Son un grupo de propiedades guardadas en una variable por ejemplo let persona = {nombre: "lucas", edad: 31}

Empty: devuelve algo de valor null es decir nulo

\*\*Operadores\*\*

-----------------

\*\*Asignación:\*\* se utiliza para asignar un valor a una variable o a una constante este se utiliza con el signo igual " = " .

\*\*Aritméticos:\*\* son operadores binarios es decir que necesita dos partes son operadores matemáticos + , - , \* , / , %.

\*\*Unitarios:\*\* requieren solo un operando que realiza operaciones de sirve para ( ++ )incrementar ( -- ) decrecer un valor ( ! ) sirve para negar un valor.

\*\*Operador de acumulación:\*\* es con un += e indica que una variable mantiene su valor mas algo mas que se le quiera sumar

\*\*Operador de disminución:\*\* es con un -= e indica que una variable mantiene su valor menos algo que se le quiera restar

\*\*Condicionales:\*\*Sirve para ejecutar un bloque de código siempre que cumpla una condición dada

\*\*Relacionales:\*\* Sirve para comparar en relación de un operador con otro para saber si son == iguales, si son != distintos, si son > mayor que, si son < menor que

\*\*Lógicos:\*\* sirve para evaluar operaciones complejas. Sirve para construir operaciones lógicas AND que se escribe && y significa y...

OR que se escribe || y significa o....

Funciones Globales

-------------------------

Son funciones que ya están definidas y solo basta con llamarlas y pasar los parámetros (si lo necesita)

\*\*alert( )\*\*: muestra un mensaje en un popup.

\*\*console.log( )\*\*: muestra un mensaje en la consola también puede recibir datos.

\*\*console.error()\*\* muestra un mensaje en la consola también puede recibir datos pero con un formato de error.

\*\*typeOf():\*\* Sirve para saber el tipo al que pertenece el parámetro que le estoy pasando

let letras = "texto"

typeOf(letras) ----> Me devuelve "String"

\*\*prompt():\*\* Ejecuta un Pop Up del navegador donde te permite introducir texto en un input y tiene un botón de cancela y de aceptar