Logotipo, Icono

Descripción generada automáticamente

**ECOLE TECHNIQUE  
ECOLE DES MÉTIERS LAUSANNE**

07 novembre 2023

Adrian toledo

Enseignant : Aurélie Curchod

« Snake »

Support de clase : P\_Bulle

Faire un réplica du célèbre jeu Snake

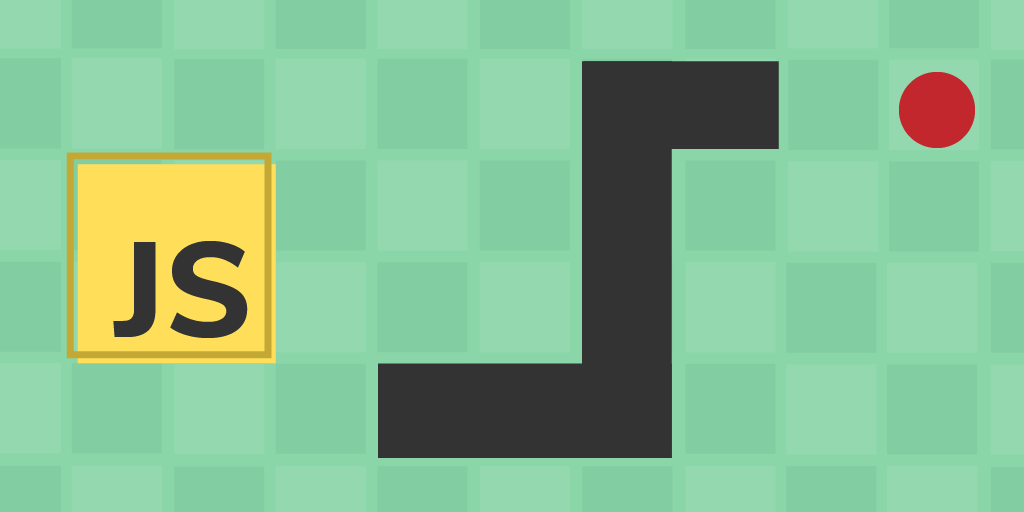


Table des matières

[1. P\_Bulle : Snake 1](#_Toc150256903)

[1.1. Introduction 1](#_Toc150256904)

[2. Conclusions 1](#_Toc150256905)

[3. Sources 1](#_Toc150256906)

# P\_Bulle : Snake

## Introductio

## Objectif

# JavaScript

JavaScript es un lenguaje de programacion interpretados que permite inetrectuar y trabajar sobre los codigos HTML de las paginas web. Su origen viene del dialecto ECMAScripti qeu se s define como codigo orientado a obketos, basados en prototipos, débilmente tipado y dinámico.

Actualmente, los navegadores modernos soportan el dialecto ECMAScript5.1 para poder soportar los archivos javascirpt.

## IDE

Un IDE (Intergrated Development Enviroment) es un editor que soporta los proyectos completos y permite navegar entre archivos, provee autompletados, ambiente de pruebas y mas caracteristicas.

Para el uso de JavaScipt se requiere editores que puedan controlarel archivo y su contenido en el proyecto. Para este caso, usaremos un Microsoft Visual Code como IDE par construir el codigo Snake a medidas.

## Introduction « Hello World »

Declarar en el HTML

<script></script>

En caso de un archivo externo :

<script src="/path/to/script.js"></script>

Para mostrar un mensaje :

Alert(``Hello’’)

## Fundamentos de JavaScript

### Variables

* + - * + Let : se usa la declaracion LET para guardar variables. Este puede registrar tanto numeros como texto. Ex « let name = ‘adri’ »
        + Const : se usa la declaracion constante CONST para crear variables que su valor no vaya ser cambiado. Ejemplo :

Const COLRO\_RED= ‘#F00’

* + - * + Var : las declaraciones con VAR se utilizave recurrentemente para la declaracion de variables, debido a su falta de definicion y visibilidad exacta, fue remplazado por el LET. VAR tiene la desventaja que su uso dentro de bloques o methodos seran leido de forma global, por lo que no existe variables temporales. Su uso actual es para crear variables que formen parte del inicio del codigo. Ejemplo : « var name = ‘adri’ »

### Tipos de datos

* Numeros « let n=123 ; »
* Numero decimales « let n=12.3 ; »
* Numeros mayores a (253-1) « let number=254n ;»
* String, cadena de caracteres « let name = ‘adri’ » « let nameAge = ‘name adrian age ${age}’ //Se muestra name adrian age 24 »
* Boolean « let eating = false »
* Nul et undefined « let age = null ; let age2 = undefined »
* Typeof para conocer el tipo de una variable ex : «typeof age  // ‘number’ »

### Array

* Crear : « Let array = [1,2] »; « let array = new Array(item1, item2)
* Insertar : « array[1] = 3 » ;
* Longitud de la cadena : « array.length »
* Conocer el ultimo valor : « array.at(-1) »
* Pop – elimina el ultimo valor : « array.pop()»
* Push – anade un valor al final de la cadena : « array.push(7) »
* Shift – elimina el primer valor de la cadena : « array.shift() »
* Unishift – aande un valor al principio de la cadena : « array.unshift(5) »
* Matriz - let matrix = [[1, 2, 3], [4, 5, 6],[7, 8, 9]]; alert(matrix[1][1])
* ForEach : arr.forEach(function(item, index, array) { // ... do something with item });

### Conversiones

* ToString cambia un valor a tipo texto ejemplo : « let value = true ; value = String(value) »
* ToNumber cambia un valor a tipo numero ejemplo : « let value = ‘123’; value = Number(value) »
* ToNumber cambia un valor a tipo bolean ; « let value = 1 ; value=Boolean(value)

### Mateaticas - Operadores clasicos

* « let suma = 2+1 »
* « let resta = 2-1 »
* « let multi = 2\*2 »
* « let division = 2/1 »
* « let resto = 2%1 »
* « let exponencias = 2\*\*2 »
* «  let n +=5 »
* « let n++ ; »
* Random
* Math(). floor

### Comparadores , conditionales y operadores logicos

* Estructura de « if (){}else{} »
* Mayor que : « a>b » O mayor o igual que « a>=b »
* Menor que « b<a » o menor que « a <=b »
* Iugal « a==b » o no es iugal « a+=b »
* Operador OR « a||b »
* Operador AND « a %%b »
* Operador NOT « !a »
* Operador Nullish Coalescing « a ?? b », si el primer valor es nullo o desconocido, volvera el seugndo valor.

### Funciones

* Function showMessage(){}
* Let sum = (a,b) => a+b ;

### Condiciones

* If(condition){}
* Switch(xxxx){ case ‘xxxx’ : \_\_\_ [break] default : \_\_\_[break]}

### Bucles

* « While (condition) {} » Mientras la condicion sea cierta, se seguira procesando las iteraciones.
* « Do {} While (condition) » Primero ejecutara el contendio de la funcion y luego verificara si la condicion es verdadera para ejecutar de nuevo la funcion
* « For (let i=0 ; i >3 ; i++ ){} » en este bucle, se trabaja en funcion de la variables creada « i », donde hay que crearla « let i=0 », poner su condicion « i>3 » y la modificacion para el siguiente bucle « i++ »

## Interacciones

* Alert : se muestra un mensaje al usuario hasta que acepte

« alert(« ‘hello world’) »

* Prompt : acepta la introducciond de dos parametros donde el primero srivira para mostrar un mensaje y el seguno para registarar lo que el usuario quiera escribir.

« let test = prompt(‘Cuantos anos tienes ?’,’[respuesta]’)

* Confirm : muestra un mensaje y espera que el usuario elgia entre confirmar o cancelar.

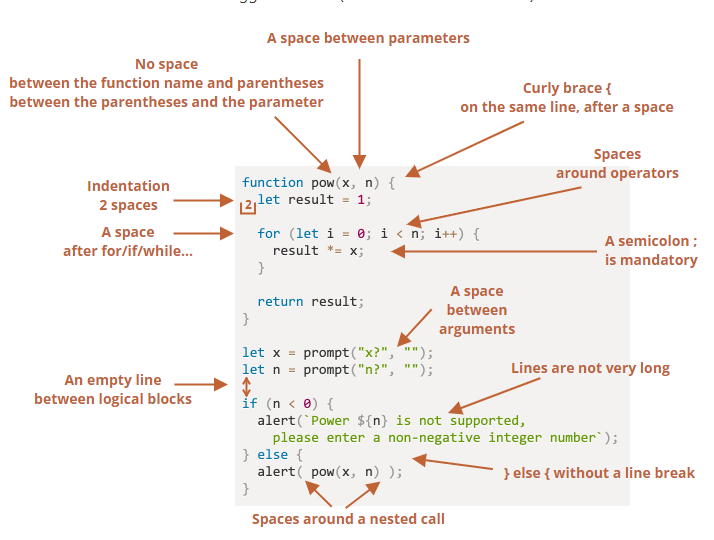
## Objects

* Declarar : Let user = new Object() ; Let user = {}
* Propiedades : let user = {name : ‘Fede’, age : 30}
* Set propiedad : user[‘celibataire’] = true ;
* Get propiedad : user[‘age’] ;
* Delate propiedad : delate user [‘age’];

## Inputs

## Import/export

## Coding style



## Tableau

## Canvas

# Snake Game

# Conclusions

# Sources

* Chat GPT n'a pas été utilisé pour la réalisation de ce projet