# Formation Javascript

**Formateur** LESUEUR Yohann: ylesueur@dawan.fr

**Projet**: https://github.com/ylesueur-dawan/javascript-22-03-2021.git

Extension VScode:
https://marketplace.visualstudio.com/items?itemName=CoenraadS.bracket-pair-colorizer

Javascript est un langage dynamiquement typé (on ne precise pas les types lors de la creation).

Types primitif:
Number: nombre entier, ou à virgule
String: chaine de caractères (lettre, un mot, une phrase)
Boolean: vrai ou faux (true ou false);
undefined: la valeur n'est pas définie
null
---
object,
BigInt
Symbol

Regles de nommage d'une variable:
1- Le nom d'une variable ne peut contenir que des lettres, des chiffres, des _ (underscore) et des $
2- Le nom d'une variable ne peut pas commencer par un chiffre
3- Les nom d'une variable ne doit pas être un mot clef
https://www.w3schools.com/js/js_reserved.asp
4- Le nom d'une variable est sensible aux majuscules: Age, age, AgE sont 3 variables différentes

Convention de nommage:
1- camelCase + minuscule: nomDeLaVariable
2- snake_case + majuscule: NOM_DE_CONSTANTE
3- Toujours choisir des noms explicites pour nos variables

Les types primitifs (number, string, boolean) ne sont pas modifiables
Toutes les operations qu'on effectuent, creer une nouvelle valeur a partir de l'ancienne
Il faut donc soit la stockée soit l'utiliser directement
Mais la variable "originelle" ne sera pas modifiée.

Litteraux de Gabarit:
https://developer.mozilla.org/fr/docs/Web/JavaScript/Reference/Template_literals

Les fonctions:
https://www.dropbox.com/s/2qvvv4anskgnz0c/fonction.png?dl=0

Les méthodes sur les string:
https://developer.mozilla.org/fr/docs/Web/JavaScript/Reference/Global_Objects/String

Les événements HTML:
https://www.w3schools.com/tags/ref_eventattributes.asp

Position un element:
https://developer.mozilla.org/fr/docs/Web/API/Element/insertAdjacentElement

Css Diner
https://flukeout.github.io/

Array:
https://developer.mozilla.org/fr/docs/Web/JavaScript/Reference/Global_Objects/Array

On ne peut pas copier un objet en faisant:
let obj2 = obj1

Pour copier un objet ou tableau, il faut utiliser:
let obj2 = {...obj1 }
ou
let tab2 = [...tab1]

Type primitif:
https://www.dropbox.com/s/d3wqeotmzpbj1ps/stack_heap.png?dl=0

Objets, Tableau:
https://www.dropbox.com/s/d3wqeotmzpbj1ps/stack_heap.png?dl=0

Spread Operateur:
https://developer.mozilla.org/fr/docs/Web/JavaScript/Reference/Operators/Spread_syntax

Les propriétés & méthodes associés aux éléments du DOM:
https://developer.mozilla.org/fr/docs/Web/API/HTMLElement
https://developer.mozilla.org/fr/docs/Web/API/Element

Les users
https://jsonplaceholder.typicode.com/users

Les datasets:
https://developer.mozilla.org/fr/docs/Web/HTML/Global_attributes/data-*

Faille XSS:
https://zestedesavoir.com/articles/232/les-failles-xss/

https://www.root-me.org/
https://xss-game.appspot.com/level2

Les tableaux:
foreach : https://developer.mozilla.org/fr/docs/Web/JavaScript/Reference/Global_Objects/Array/forEach

FontAwesome:
https://fontawesome.com/

XHR:
https://developer.mozilla.org/fr/docs/Web/API/XMLHttpRequest

Ready State: (entre l'etape 3 et 4 on pourrait afficher un loader https://i.stack.imgur.com/kQ7r0.gif)

https://developer.mozilla.org/fr/docs/Web/API/XMLHttpRequest/readyState

API REST
https://www.redhat.com/fr/topics/api/what-is-a-rest-api

API SOAP vs REST:
https://www.redhat.com/fr/topics/integration/whats-the-difference-between-soap-rest

JSON Placeholder:
https://jsonplaceholder.typicode.com/users

Requete HTTP verbes:
https://developer.mozilla.org/fr/docs/Web/HTTP/Methods

Requete HTTP status:
https://developer.mozilla.org/fr/docs/Web/HTTP/Status

Fetch:
https://developer.mozilla.org/fr/docs/Web/API/Fetch_API

AOS
https://michalsnik.github.io/aos/

APIS Public:
https://github.com/public-apis/public-apis

// https://pi.delivery/#apipi_get

// https://api.pi.delivery/v1/pi?start=0&numberOfDigits=10
// https://jsonplaceholder.typicode.com/users

Fake Store:
https://fakestoreapi.com/products
https://via.placeholder.com/250x150

Aliments:
https://fr.openfoodfacts.org/data

Exercices:
ex1: https://github.com/ylesueur-dawan/javascript-22-03-2021/blob/master/99-exercices/js/ex1.js
ex2: https://github.com/ylesueur-dawan/javascript-22-03-2021/blob/master/99-exercices/js/ex2.js

ex: https://github.com/ylesueur-dawan/javascript-22-03-2021/tree/master/07-DOM/03-exercices

ex: s'entrainer avec fetch / xhr
https://github.com/ylesueur-dawan/javascript-22-03-2021/blob/master/99-exercices/js/ex3.js

ex : mini ecommerce
https://github.com/ylesueur-dawan/javascript-22-03-2021/blob/master/07-DOM/09-e-commerce/index.html

Sharemycode:
https://sharemycode.fr

ex: https://sharemycode.fr/cxk