JavaScript

Langage interprété par le navigateur, donc orienté user. Permet de dynamiser les sites internet.  
Langage orienté prototype -> contrairement aux langages fortement typés, dans lesquels les objets ne peuvent pas être modifiés, les prototypes peuvent l'être en cours de route.  
Langage très polyvalent, avec plus de 2000 fonctions !

Implémenter JS :  
Dans le dossier de mon site internet :  
MonSiteInternet – index.html

css

-Scripts ou Stag

Balise script : permet de faire interpréter JS dans le navigateur. Faire de préférence les différents scripts dans des fichiers à parts. Il est possible de définir le script directement dans les balises <script> si le script est court (une ligne max).   
  
Par défaut, le script se place au bas du body, après le html. A un niveau pro, le script sera placé dans le head. Possibilité de mettre plusieurs scripts.

Balise noscript : permet d'afficher un message d'erreur en cas de dysfonctionnement de JavaScript. Cette balise peut aussi contenir tout le contenu du site ne dépendant pas de JS.

Fonction alert : permet de faire afficher un pop-up d'information à l'écran. (Équivalent ConsoleWriteLine)

Cmd prompt : permet d'envoyer un message à l'écran, et de récupérer une information en retour (réponse de l'utilisateur) (équivalent ConsoleReadLine)

Cmd confirm : permet de récupérer un booleen en réponse. ("ok" ou "annuler")

Let : permet de déclarer une variable.

Typeof() : permet de récupérer le type d'une variable. Le type d'une variable en JS est similaire à Python, il est dynamique et défini par la valeur associée à la variable.

Math.random() : permet d'obtenir un chiffre aléatoire entre 0 et 1  
Pour obtenir un nombre plus grand il faut utiliser la formule :   
Math.random() \* (max - min) + min ;(on multiplie le nombre aléatoire par la différence entre le nombre max et le nombre min, puis on ajoute le min pour rester dans le range)  
ex : pour un nombre aléatoire entre 30 et 100   
math.random()\*(100-30) => 0.1235\*70 => 8.645. Ce nombre est en dessous de 30, il faut donc rajouter le minimum pour rentrer dans le range.

Math.floor : tronque un nombre à l'entier atteint (ex : 38.99999 sera tronqué à 38)  
Math.ceil : arrondis tjrs au supérieur (8.10 deviendra 9)  
Math.round : arrondis au plus proche (8.666 deviendra 9)

# Les conditions:

Elles fonctionnent exactement pareil que pour C#.  
  
If  
If… else if …. else  
switch case   
ternaire

# Le DOM (Document Object Model)

## Methodes

Document = code HTML.   
document.body = permet de récupérer la partie body du HTML  
document.getElementById("identifiant") = permet de récupérer un élément particulier parmi le code HTML. Très utile si l'on veut manipuler les différents éléments et les stocker dans des variables. Ex : let main\_h1= document.getElementById("MainH1")  
  
Pour récupérer le contenu d'un élément :   
document.getElementById("MainH1").innerText OU main\_h1.innerText (si l'élément à été mis dans une variable.)

Pour changer le contenu d'un élément :

main\_h1.innerText = "Coucou les Wad" /!\ Ce changement est dynamique! Le document sur le server n'est JAMAIS modifié. Le changement disparaitra lorsque la page sera rechargée par exemple.

Si je ne connais pas l'id:  
document.getElementByTagName("tag") = va récupérer tous les éléments contenant la balise indiquée en paramètre.   
Peut permettre de mettre tous les éléments dans une collection d'un coup ! Ex. :  
let h1\_collection = document.getElementByTagName("h1").

Document.write(" mon texte") = permet d'écrire du contenu html. Peu utilisé.

Attribut d'évènements : seuls attributs dans lesquels nous pouvons directement insérer du javascript dans du HTML.

# Les fonctions

onclick ="nomDeFonction" -> permet d'appeler une fonction qui sera exécutée lorsque le bouton sera click

Pour sécuriser nos fonctions il est plus judicieux de les écrire au sein d'une constante.

Throw permet d'envoyer un message d'erreur visible uniquement dans la console. Cette erreur arrête la fonction.

## Let vs. Var

Var permet d'associer 2 variables ayant le même nom. A éviter d'utiliser. Privilégier le let.

Innerhtml : interprète les balises html en tant que telles

Innertext : interprète TOUT comme texte, balises incluses.

# Les tableaux

Possible de dépasser la longueur du tableau pour rajouter une donnée, mais les indices passés seront "vides" (mais existant).

Btn.onclick = function () {fonctionAappliquer(param1, param2)}   
En JS, pour activer une fonction onclick, on la met dans une autre fonction anonyme. Si on laisse la fonction telle qu'elle, on obtiendra que le résultat de la fonction appliquée.

# Les Objets

En js, pour créer un objet, on utilise une paire d'accolades. Plus facile de gérer des tableaux d'objets

Boucle forIn:  
  
for (const key in nomDuTableau) {

Const elem = nomDuTableau[key]

} Cette boucle permet de manipuler aussi les clé qui ne sont pas de type number

for of similaire à une boucle foreach

getTime = nbrs de milisecondes depuis le 1 janvier 1970 à aujourd'hui.