



LANGAGE DE PROGRAMMATION JAVA SCRIPT

AVALON SECURE

06 SEPTEMBRE 2023

AVALON SECURE

LANGAGE JAVA SCRIPT

Réalisé par

Josias N. S. TITO

DEFINITION ET HISTORIQUE

- Le JavaScript est un langage de programmation qui permet de créer des pages interactives et vivante à l'aide de script. Ce langage est aussi un langage orienté objet et est aussi majoritairement Client-Side. Il nous permet de manipuler dynamiquement de HTML et du CSS.
- Le JavaScript, créé en 1995 par Brendan Eich (pour la Netscape Communication Corporation). Si le terme Java est commun au langage du même nom, le JavaScript est radicalement différent. La version ES5 date de 2009.

PREREQUIS A CONNAITRE

1. Syntaxe JavaScript

La syntaxe du Javascript n'est pas compliquée. De manière générale, les instructions doivent être séparées par un point-virgule que l'on place à la fin de chaque instruction :

```
<script>
    instruction_1;
    instruction_2;
    instruction_3;
</script>
```

Les fichiers JavaScript portent l'extension **.js**. Ce fichier est ensuite appelé depuis la page Web au moyen de l'élément **<script>** et de son attribut **src** qui contient l'URL du fichier **.js**. Par exemple dans le fichier hello.js, on écrit :

```
alert('Hello world!');
```

Et dans le body de la page html, on trouve :

```
<script src="hello.js"></script>
```

Pour éviter des problèmes de chargement sur les pages, il est conseillé de placer les éléments **<script>** juste avant la fermeture de **<body>**.

Les commentaires en JavaScript se traduisent par **/* */**

alert() est une instruction simple, appelée fonction, qui permet d'afficher une boîte de dialogue contenant un message.

La fonction **prompt()** s'utilise comme **alert()** mais avec une particularité. Elle renvoie ce que l'utilisateur a écrit sous forme d'une chaîne de caractères.

Exemple :

```
var text = prompt("Saisissez votre prénom") ;
```

Les variables en JavaScript

- La variable : qui permet de garder en mémoire tout type de donnée. Elle est utilisée dans les scripts. Elle ne peut pas commencer par un chiffre ni prendre le nom d'une fonction déjà existante en JavaScript. Il y a deux manières de créer une variable en JavaScript :

```
<script>  
    var maVariable = 2 ;  
</script>
```

ou

```
<script>  
    var maVariable ;  
    maVariable = 2 ;  
</script>
```

Lorsqu'une variable est déclarée pour le préciser que c'est une chaîne de caractère il faut utiliser les doubles quotes (".....").

Exemple :

```
<script>  
    var monText = " J'écris mon texte ici "  
</script>
```

Une variable peut enfin être de type booléen (boolean), avec deux états possibles : vrai ou faux (true ou false).

Les Conditions en JavaScript

➤ La Condition if else

La condition est composée :

- de la structure conditionnelle if ;
- de parenthèses qui contiennent la condition à analyser, ou plus précisément le booléen retourné par les opérateurs conditionnels ;
- d'accolades qui permettent de définir la portion de code qui sera exécutée si la condition se vérifie.

Syntaxe :

```
<script>
  If (condition) {
    Instruction 1 ;
    Instruction 2 ;
  } else {
    Instruction ;
  }
</script>
```

➤ La condition switch

Cette structure permet de gérer une courte liste de possibilités. Il présente une syntaxe particulière.

Syntaxe :

```
<script>
var maVariable = "Vvariable"
switch (maVariable) {
    case 1 :
        alert (".....") ;
        break ;
    case 2 :
        alert (".....") ;
        break ;
    default :
        alert (".....") ;
}
</script>
```

Break permet d'arrêter la fonction et de passer à un autre cas.

La fonction **parseInt** permet de préciser le type de la valeur

Exemple :

```
var message = parseInt ( prompt ("Choisissez le tiroir à ouvrir (1 à 4) :
" ) );
```

Quelques exercices sur les conditions

Exercice 1 : Condition if

Fournir un commentaire selon l'âge de la personne.

Vous devez fournir un commentaire sur 4 tranches d'âge qui sont les suivantes :

Tranche d'âge Exemple de commentaire

1 à 6 ans « Vous êtes un jeune enfant. »

7 à 11 ans « Vous êtes un enfant qui a atteint l'âge de raison. »

12 à 17 ans « Vous êtes un adolescent. »

18 à 120 ans « Vous êtes un adulte. »

Résultat

```
<script>
```

```
var age = parseInt(prompt('Quel est votre âge ?'));
```

```
if (1 <= age && age <= 6) {
```

```
    alert('Vous êtes un jeune enfant.');
```

```
} else if (7 <= age && age <= 11)
```

```
{
```

```
    alert ('Vous êtes un enfant qui a atteint l\'âge de raison.');
```

```
} else if (12 <= age && age <= 17)
```

```
{
```

```
    alert ('Vous êtes un adolescent.');
```

```
} else if (18 <= age && age <= 120)
```

```
{  
    alert ('Vous êtes un adulte.');
```

} else

```
{  
    alert ('Erreur !!');
```

}

```
</script>
```

Exercice 2 : Condition switch

Ecrire un script qui affiche le nom du jour en fonction de la valeur de jour sachant que les chiffres représentent un jour de la semaine par exemple (1 pour Lundi, 2 pour Mardi, 3 pour mercredi etc ...)

Résultat

var jour = 7; // Vous pouvez changer cette valeur pour tester différents cas

```
switch (jour) {  
    case 1:  
        alert ("Lundi") ;  
        break;  
    case 2:  
        alert ("Mardi") ;  
        break;  
    case 3:  
        alert ("Mercredi") ;
```



```
        break;
    case 4:
        alert ("Jeudi") ;
        break;
    case 5:
        alert ("Vendredi") ;
        break;
    case 6:
        alert ("Samedi") ;
        break;
    case 7:
        alert ("Dimanche") ;
        break;
    default:
        alert ("Valeur non valide. Veuillez entrer un nombre entre 1 et
7.");
}
```

Les boucles en JavaScript

➤ **La boucle while**

Une boucle sert à répéter une série d'instructions. La répétition (ou itération) se fait jusqu'à ce qu'on dise à la boucle de s'arrêter. Pour une boucle, on pose une condition, et la boucle se répète tant que la condition est vérifiée

Syntaxe :

```
<script>
    while (condition) {
        instruction 1; instruction 2; instruction 3;
    }
</script>
```

➤ La boucle do while

Dans ce cas, la boucle est exécutée au moins une fois, après quoi on teste la condition

Syntaxe :

```
<script>
    do {
        instruction 1; instruction 2; instruction 3;
    } while (condition);
</script>
```

L'incréméntation permet d'ajouter une unité à un nombre au moyen d'une syntaxe courte. À l'inverse, la décréméntation permet de soustraire une unité.

Incréméntation : ++

Décréméntation : --

➤ La boucle for

Cette boucle est très utile pour l'incréméntation automatique.

Syntaxe :

```
<script>
    for (initialisation; condition; incrémentation) {
        instruction 1;
        instruction 2;
        instruction 3;
    }
</script>
```

Les fonctions en JavaScript

Les fonctions correspondent à un bloc de code dont le but est d'effectuer une tâche précise. Il y a deux types de fonctions en JavaScript à savoir :

- Les Fonctions Natives : Les fonctions natives sont des fonctions créées par des développeurs et qui ont été intégré au JavaScript lui-même.
- Les Fonctions Créées : Ce sont des fonctions créées par le codeur lui-même.

Pour créer une fonction en JavaScript il faut utiliser le mot clé **function**

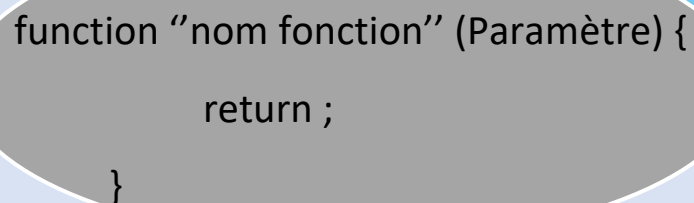
Syntaxe :

```
function "nom fonction" (Paramètre) {
    Instruction ;
}
```

Exemple :

```
function multiplication (x ; y) {  
    alert (x*y) ;  
}  
multiplication (10*20) ;
```

Syntaxe pour le retour du résultat



```
function "nom fonction" (Paramètre) {  
    return ;  
}
```

Exemple :

```
function multiplication (x ; y) {  
    return (x*y) ;  
}  
var resultat = multiplication (10*20) ;  
alert (resultat) ;
```

La création d'une variable resultat permet de stocker le resultat de la multiplication pour ensuite l'afficher à l'écran à l'aide de la fonction alert () .

Remerciement

Je remercie mon équipe Avalon Secure (Romaric S. Z. AGONHOU, Devon da-Silva, Obed COMBETTO, Hopeful O. DOSSOU, Farouk AKANHO, Tristan VIKOU, Firdaos LIAMIDI, Marius AKOWANOU) qui m'a permis de connaître des gens avec qui je peux travailler et collaborer sur des projets futurs, que cela soit en Cybersécurité, en Développement, en Réseaux & Cloud. Un spécial coucou à ma chère amie Nafal SEFOU d'être là pour nous. Infiniment merci à vous pour m'avoir aidé à écrire ce guide d'apprentissage qui va nous permettre d'avoir une idée sur le langage JavaScript.

Je tiens à vous remercier pour avoir choisi ce document pour votre apprentissage en JavaScript.

Nous aurons pour commencer besoin d'un outil : Visual Studio Code que vous pouvez télécharger via le lien

(<https://code.visualstudio.com/download>)

Pour toutes vos suggestions ou appréciations prier nous écrire par email via avalonsecure7@gmail.com .