



# LA MANU

L'ÉCOLE DES MÉTIERS DU NUMÉRIQUE  
Manufacture de compétences



## SUPPORT APPRENANT

*JavaScript*



**CONFIDENTIEL**

*Ce document est strictement confidentiel et ne doit pas être diffusé  
sans accord préalable écrit*

# JavaScript

## Ce que l'on a vu...



Le langage JavaScript, aussi reconnu sous l'acronyme JS, a été inventé en 1995 par Brendan Eich de Netscape Communications Corporation. Initialement nommé LiveScript, il a été rebaptisé JavaScript en référence au programme Java de Sun Microsystems qui était très populaire à l'époque. Cependant JavaScript et Java ne doivent pas être confondus. La ressemblance entre les 2 langages se limite à leur nom.

JavaScript est un langage de programmation de scripts qui repose sur le standard ECMAScript. Ce standard, dont la 1<sup>ère</sup> édition date de 1997, a permis de réduire les problèmes de compatibilité entre les différentes implémentations concurrentes du JS.

Les langages de scripts, tels que le JavaScript, sont généralement des langages interprétés, c'est-à-dire que le code est exécuté instruction par instruction et le résultat obtenu est immédiatement renvoyé par le programme interpréteur au navigateur. A l'inverse, les langages compilés sont d'abord transformés dans un langage machine que l'ordinateur va pouvoir exécuter avant d'être renvoyé au navigateur.

Le JavaScript est un langage qui permet de dynamiser les pages web, il permet d'améliorer l'ergonomie et l'expérience utilisateur sur ces dernières. Son fonctionnement permet de dynamiser le contenu d'une page web sans être contraint de la recharger. JavaScript sert à faire réagir les pages aux actions de l'utilisateur et à ajouter des effets esthétiques à leurs contenus. Il permet ainsi, de manipuler dynamiquement le code d'un document HTML ou d'une feuille de style CSS.

Cependant, malgré l'importance et les avantages de l'utilisation de JavaScript, il faut garder en tête que ce langage comporte également quelques inconvénients. En effet, le JavaScript n'est pas pris en charge de la même façon par les différents navigateurs, ainsi il faudra penser à anticiper son comportement sur les différents navigateurs afin de prévoir des ajustements si nécessaire. De plus, les scripts sont vulnérables aux attaques informatiques. Ils peuvent être facilement désactivés, ou encore le JavaScript peut être utilisé pour exécuter du code malveillant sur l'appareil de l'utilisateur. Il faudra donc faire attention à sécuriser au maximum son code pour se prémunir d'attaques indésirables.

Du code JavaScript peut être ajouté dans une page web de 3 manières différentes :

- Directement dans une balise HTML en tant que valeur d'un attribut d'évènement.  
Ex : `<button type="button" onclick="alert('Merci d'avoir cliqué');">Cliquez ici !</button>`
- Entre des balises `<script></script>` généralement placées à la fin du `<body>`.  
Ex :  
`<body>`  
...  
`<script>alert('Bonjour !') ;</script>`  
`</body>`
- Dans un fichier externe, de la même manière que le fichier style.css, mais lié au document HTML grâce à une balise `<script></script>` généralement placée à la fin du `<body>` ou parfois dans la balise `<head></head>`.  
Ex :  
`<body>`  
...  
`<script src="script.js"></script>`

</body>

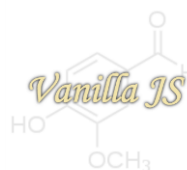
Tout comme pour le langage CSS, on préférera utiliser un fichier externe placé à la fin du <body> plutôt que d'insérer directement son code JavaScript dans le document HTML. En effet, placer le fichier script.js permet d'avoir un gain d'ergonomie car il n'y aura pas besoin d'attendre d'avoir fini de charger tout le fichier script.js avant que le navigateur ne commence à construire le reste de la page HTML.

Le langage JavaScript peut être complété par l'utilisation de frameworks (AngularJS...) et de bibliothèques (jQuery, ScrollReveal, Sweetalert...) qui permettent d'enrichir et de faciliter le développement d'applications.

### Quelques commandes...

Commande JavaScript	Description
alert('message');	Boîte de dialogue contenant une chaîne de caractères ou une variable.
alert(variable);	
prompt('message');	Boîte de dialogue permettant d'afficher un message avec un champ de saisi.
document.getElementById('nomId').value;	Récupération de la valeur d'un élément par son id.
Math.trunc(nombre);	Troncature entière d'un nombre en retirant sa partie décimale.
Math.round(nombre);	Arrondi à l'entier le plus proche.
isNaN(nombre)	Fonction permettant de déterminer si une valeur n'est pas un nombre.
function nomFonction() { ... }	Déclaration d'une fonction.
var nomVariable = valeur;	Déclaration d'une variable.
.src	Propriété permettant de renvoyer ou modifier le chemin d'une image.
.addEventListener();	Méthode mettant en place une fonction à l'appel d'un évènement donné.
.style.propriétéCSS	Modification des propriétés CSS d'un élément.
Evènements JavaScript	
click	Evènement au clic sur l'élément.
dblclick	Evènement au double-clic sur l'élément.
mouseover	Evènement à l'entrée du curseur sur l'élément.
mouseout	Evènement à la sortie du curseur sur l'élément.
focus	Evènement quand l'élément a le focus.
focusin	Evènement quand l'élément s'apprête à prendre le focus.
focusout	Evènement quand l'élément s'apprête à perdre le focus.
blur	Evènement à la perte du focus sur l'élément.
keydown	Evènement à l'appui (sans relâcher) sur une touche du clavier sur l'élément.
keyup	Evènement au relâchement d'une touche du clavier sur l'élément.
keypress	Evènement à l'appui (et relâche) d'une touche du clavier sur l'élément.

## VanillaJS



VanillaJS n'est pas un nouveau framework de JavaScript, il s'agit en réalité du langage JavaScript lui-même. Au fil du temps, le langage a été délaissé au profit de frameworks ou de bibliothèques. Pour pallier ce délaissement et réattirer l'attention sur le langage JavaScript, on a commencé à parler de VanillaJS. Les curieux attirés par cette nouvelle nomenclature retrouveront finalement le JavaScript. Malgré une syntaxe qui peut paraître plus lourde que celle de la bibliothèque jQuery, le langage JavaScript reste plus performant. En effet, les commandes JavaScript s'exécutent beaucoup plus rapidement que la plupart des framework ou bibliothèque.

## Liens utiles

<https://www.grafikart.fr/tutoriels/javascript>  
<https://openclassrooms.com/en/courses/1916641-dynamisez-vos-sites-web-avec-javascript>  
<https://openclassrooms.com/en/courses/2984401-apprenez-a-coder-avec-javascript>  
<https://openclassrooms.com/fr/courses/5543061-ecrivez-du-javascript-pour-le-web>  
<https://developer.mozilla.org/fr/docs/Web/JavaScript>  
<https://www.w3schools.com/js/>  
[https://www.w3schools.com/jsref/dom\\_obj\\_event.asp](https://www.w3schools.com/jsref/dom_obj_event.asp)  
<https://www.alsacreations.com/tuto/liste/5-javascript.html>  
<https://www.toutjavascript.com/main/index.php3>  
<http://adrienjoly.com/cours-javascript/>  
<https://sweetalert.js.org/docs/>  
<https://scrollrevealjs.org/>  
<https://greensock.com/>  
<https://fr.tuto.com/blog/2017/01/plugins-javascript-gratuits.htm>  
<https://blog.bitsrc.io/11-javascript-animation-libraries-for-2018-9d7ac93a2c59>  
<http://vanilla-js.com/>  
<http://jesuisunblog.com/vanillajs/>

## Bonnes pratiques

Rappel des principales bonnes pratiques	
<input type="checkbox"/>	Une instruction switch doit toujours contenir une étiquette default, et une instruction break dans chaque étiquette case.
<input type="checkbox"/>	Prendre en compte tous les cas possibles pour les conditions.
<input type="checkbox"/>	Utilisation des simples quotes.
<input type="checkbox"/>	Utilisation de l'anglais est obligatoire pour les noms de variables, des class, des id...
<input type="checkbox"/>	Mettre le script dans un fichier à part et l'inclure en bas de page HTML avant la balise fermante </body>.
<input type="checkbox"/>	Attention à l'ordre des balises script, mettre le fichier Script.js en dernier.
<input type="checkbox"/>	Utiliser les commentaires.
<input type="checkbox"/>	Vérifier que la saisie utilisateur correspond à ce qui est attendu.
<input type="checkbox"/>	Pour une meilleure compréhension du code, il faut indenter son code.
<input type="checkbox"/>	Vocabulaire spécifique :

- |  |  |
|--|--|
|  | <ul style="list-style-type: none"><li>• Opérateurs de comparaison</li><li>• Conditions</li><li>• Instructions</li><li>• Fonctions</li><li>• Eléments</li><li>• Evènements</li><li>• Concaténation</li><li>• Template strings</li><li>• Regex</li></ul> |
|--|--|