**JAVASCRIPT**

Le JavaScript est un langage de programmation créé en 1995. Le JavaScript est aujourd’hui l’un des langages de programmation les plus populaires et il fait partie des langages web dits « standards » avec le HTML et le CSS. Son évolution est gérée par le groupe ECMA International qui se charge de publier les standards de ce langage.

On dit que le HTML, le CSS et le JavaScript sont des standards du web car les principaux navigateurs web (Google Chrome, Safari, Firefox, etc.) savent tous « lire » (ou « comprendre » ou « interpréter ») ces langages et les interprètent généralement de la même façon ce qui signifie qu’un même code va généralement produire le même résultat dans chaque navigateur.

Pour définir ce qu’est le JavaScript et le situer par rapport aux autres langages, et donc pour comprendre les intérêts et usages du JavaScript il faut savoir que :

* Le JavaScript est un langage dynamique ;
* Le JavaScript est un langage (principalement) côté client ;
* Le JavaScript est un langage interprété ;
* Le JavaScript est un langage orienté objet.

Un langage dynamique

Le JavaScript est un langage dynamique, c’est-à-dire un langage qui va nous permettre de générer du contenu dynamique pour nos pages web.

Un contenu « dynamique » est un contenu qui va se mettre à jour dynamiquement, c’est-à-dire changé sans qu’on ait besoin de modifier le code manuellement mais plutôt en fonction de différents facteurs externes.

On oppose généralement les langages « dynamiques » aux langages « statiques » comme le HTML et le CSS. Illustrons les différences d’utilisation entre ces types de langage en discutant des possibilités du HTML, du CSS et du JavaScript.

Pour rappel, le HTML est un langage de balisage (langage qui utilise des balises) qui est utilisé pour structurer et donner du sens aux différents contenus d’une page. Le HTML nous permet de communiquer avec un navigateur en lui indiquant que tel contenu est un titre, tel contenu est un simple paragraphe, tel autre est une liste, une image, etc.

Le navigateur comprend les différentes balises HTML et va alors afficher notre page à nos visiteurs en tenant compte de celles-ci.

Le contenu HTML ne va jamais être affiché tel quel, brut, mais des règles de mises en forme vont lui être appliquées. Ces règles de styles vont être définies en CSS. Le CSS va ainsi nous permettre d’arranger les différents contenus HTML de la page en les positionnant les uns par rapport aux autres, en modifiant la couleur des textes, la couleur de fond des éléments HTML, etc.

Le CSS va ainsi se charger de l’aspect visuel de notre page tandis que le HTML se charge de la structure (définir les contenus) de celle-ci.

Le HTML et le CSS forment ainsi un premier couple très puissant. Cependant, nous allons être limités si nous n’utilisons que ces deux langages tout simplement car ce sont des langages qui ne permettent que de créer des pages « statiques ».

Une page statique est une page dont le contenu est le même pour tout le monde, à tout moment. En effet ni le HTML ni le CSS ne nous permettent de créer des contenus qui vont se mettre à jour par eux-mêmes. Le CSS, avec les animations, nous permet de créer des styles pseudo-dynamiques mais tout de même prédéfinis.

C’est là où le JavaScript entre en jeu : ce langage va nous permettre de manipuler des contenus HTML ou des styles CSS et de les modifier en fonction de divers évènements ou variables. Un évènement peut être par exemple un clic d’un utilisateur à un certain endroit de la page tandis qu’une variable peut être l’heure de la journée.

Un langage interprété

On peut encore séparer les langages selon qu’ils puissent être exécutés directement (on parlera alors de langages interprétés) ou qu’il faille les transformer en une autre forme pour pouvoir les exécuter (on parlera alors le langage compilés).

Le JavaScript est un langage interprété. Cela signifie qu’il va pouvoir être exécuté directement sous réserve qu’on possède le logiciel interpréteur. Pas de panique ici : tous les navigateurs connus possèdent leur interpréteur JavaScript.