

CSS, SCSS et les langages de programmation

Introduction

Si l'on compare CSS avec d'autres langages de programmation plus classiques, on remarque assez vite des distinctions flagrantes (surtout dans les plus anciennes versions de CSS).

Pour des raisons évidentes : le but de CSS n'est pas de proposer un **langage turing complet**, mais un langage permettant de **décrire tout type d'UI possible** en combinaison avec le HTML. Ainsi, il n'a pas été pensé de la même façon que les autres langages : le but n'est pas de récupérer des inputs de l'utilisateur, d'effectuer des opérations mathématiques sur ces inputs, ou exécuter des instructions de façon conditionnelle en fonction de ceux-ci : le but est simplement, pour une page HTML donnée, de proposer un langage permettant la mise en place d'une UI dans le navigateur web.

Des besoins en évolution

Au fur et à mesure de l'avancée du web, les limitations de CSS ont commencé à se faire sentir.

En effet, plus les sites web ont commencé à gagner en volume, plus les fichiers CSS sont devenus massifs, et plus la maintenance de ces fichiers est devenue problématique

CSS, par exemple, ne permettait pas avant **2012** d'utiliser des variables, et ces variables ne sont devenues compatibles avec tous les navigateurs qu'en **2017**.

Si l'on possède un site gigantesque servant une grande quantité de page, et que du jour au lendemain, les designers décident de changer la charte graphique de la marque, il faut alors trouver à la main la totalité des références aux couleurs dans le CSS et trouver un moyen de changer ces couleurs.

Le CSS est également connu pour sa tendance à ne pas du tout respecter les principes **DRY** (Don't Repeat Yourself)

La réponse de SCSS

Les technologies utilisées dans nos navigateurs ont du mal à s'adapter aux changements rapides de l'écosystème web. Chaque fois que l'on souhaite sortir un nouveau standard en CSS, Javascript ou HTML, il faut pouvoir assurer une certaine forme de compatibilité avec les navigateurs récents. Et généralement, ces standards ne sont utilisés par les développeurs que lorsque tous les navigateurs encore maintenus ont décidé d'adopter ce standard.

L'idée de SCSS était donc la suivante : créer un langage de programmation reprenant la totalité de la syntaxe de CSS, y ajouter toutes les fonctionnalités manquantes, et ensuite transpiler le SCSS en CSS classique.