

# TABLE DE MATIERE :

INTRODUCTION AU JAVASCRIPT …………………………………………………… page 2\_\_6

DATA STRUCTURS ………………………………………………………………………………………page 2\_\_\_\_3

VARIABLES…………………………………………………………………………………………………page\_\_\_\_\_\_3

CONDITIONS………………………………………………………………………………………………page4\_\_\_\_\_5

LOOPS………………………………………………………………………………………………………page 5\_\_\_\_\_6

FUNCTIONS…………………………………………………………………………………………....…page \_\_\_\_\_\_6

INTRODUCTION AU HTML…………………………………………………………………………..page7\_\_\_\_\_8

INTROUCTIONS AU CSS……………………………………………………………………………..page9\_\_\_\_10

INTRODUCTION AU GIT ET GITHUB…………………………………………………………….page11\_\_\_12

* **DEFINIITION :**

INTRODUCTION TO JAVASCRIPT

* [JavaScript](https://developer.mozilla.org/fr/docs/Glossary/JavaScript) est un langage de programmation qui permet de créer du contenu mis à jour de façon dynamique, de contrôler le contenu multimédia, d'animer des images, et tout ce à quoi on peut penser. Bon, peut-être pas tout, mais vous pouvez faire bien des choses avec quelques lignes de JavaScript.
* **DATA STRUCTURS :**
* Les langages de programmation disposent de structures de données natives. On listera les structures de données natives en JavaScript. On détaillera leurs propriétés et les façons de les utiliser voire de les combiner.

1. Le type String :

Ce type JavaScript est utilisé afin de représenter des données de texte.



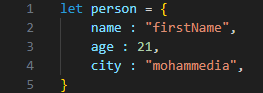
1. Le type booléen :

Un booléen représente le résultat d'une assertion logique et peut avoir deux valeurs : true (pour le vrai logique) et false (pour le faux logique) (voir Boolean pour plus de détails sur la représentation objet de ce type).



1. Le type objet

En JavaScript, les objets peuvent être considérés comme des collections de propriétés.



1. Le type Undefind :

Une variable à laquelle on n'a pas affecté de valeur vaudra undefined.



1. Le type Array(tableau) :

Un tableau est une collection d'éléments du même type de variable stockés dans des emplacements de mémoire contigus. C'est l'une des structures de données les plus populaires et les plus simples et elle est souvent utilisée pour implémenter d'autres structures de données. Chaque élément d'un tableau est indexé à partir de 0.



1. Le type null :

Le type nul ne possède qu'une valeur



1. Le type NaN :

En JavaScript, NaN (Not-a-Number) est une valeur spéciale qui représente un résultat qui n'est pas un nombre valide. NaN est un concept spécifique au langage qui indique qu'une opération mathématique ou une conversion a échoué.



* **VARIABLES :**

Une variable est un “stockage nommé” pour les données. Nous pouvons utiliser des variables pour stocker des goodies, des visiteurs et d’autres données.

1. LET :

Pour créer une variable en JavaScript, nous devons utiliser le mot-clé let.



1. VAR :

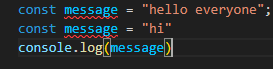
Le mot-clé var est presque identique à let. Il déclare également une variable, mais d’une manière légèrement différente, à la mode “old school”.



1. CONST :

pour déclarer une constante (non changeante), on peut utiliser const  plutôt que let,

Les variables déclarées à l’aide de const sont appelées “constantes”. Elles ne peuvent pas être réassignées. Une tentative de le faire provoquerait une erreur :



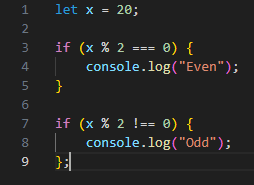
En effet, en utilise le plus de ces trois variables **LET** et **CONST.**

* **CONDITIONS :**

Les instructions conditionnelles JavaScript vous permettent d'exécuter des blocs de code spécifiques en fonction de conditions. Si la condition est remplie, un bloc de code particulier s'exécutera ; sinon, un autre bloc de code s'exécutera en fonction de la condition.

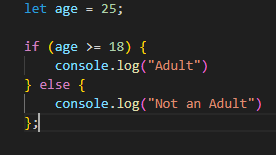
1. IF :

L'instruction if est utilisée pour évaluer une condition particulière. Si la condition est vraie, le bloc de code associé est exécuté.



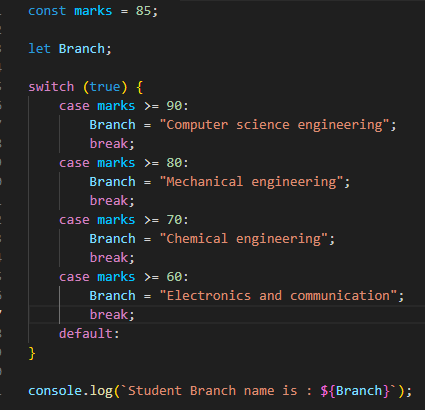
1. IF-ELSE :

L'instruction if-else exécutera une action pour une condition spécifique. Ici, nous utilisons l'instruction else dans laquelle l'instruction else est écrite après l'instruction if et ne contient aucune condition dans son bloc de code.



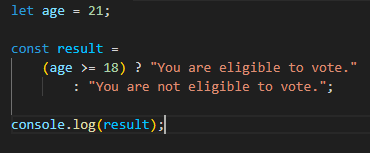
1. SWITCH CASE :

À mesure que le nombre de conditions augmente, vous pouvez utiliser plusieurs instructions else-if en JavaScript. Mais lorsque nous avons affaire à de nombreuses conditions, l'instruction switch peut être une option plus préférée.



1. TERNARY OPERATORS :

L'opérateur conditionnel, également appelé opérateur ternaire (?:), est un raccourci pour exprimer des instructions conditionnelles en JavaScript.



* **LOOPS (LES BOUCLES) :**

Les boucles JavaScript sont essentielles pour gérer efficacement les tâches répétitives. Elles exécutent un bloc de code de manière répétée tant qu'une condition spécifiée reste vraie. Ces boucles sont des outils puissants pour automatiser les tâches et rationaliser votre code.

1. FOR :

Par exemple, supposons que nous souhaitons imprimer « Hello World » 5 fois. Cela peut être fait facilement en utilisant JS Loop. Dans Loop, l'instruction ne doit être écrite qu'une seule fois et la boucle sera exécutée 5 fois comme indiqué ci-dessous :



**Syntaxe :**

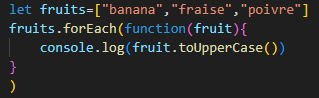
for (initialisation ; condition de test ; incrémentation/décrémentation) {

déclaration(s)

}

1. FOREACH :

Une fonction à exécuter pour chaque élément du tableau. Sa valeur de retour est ignorée.



* **FUNCTIONS :**

Une fonction JavaScript est définie avec le functionmot-clé, suivi d'un nom , suivi de parenthèses ().

1. **FONCTIONS CLASSIQUES :**

Les noms de fonction peuvent contenir des lettres, des chiffres, des traits de soulignement et des signes dollar (mêmes règles que les variables). Les parenthèses peuvent inclure des noms de paramètres séparés par des virgules :( paramètre1, paramètre2, ... ).

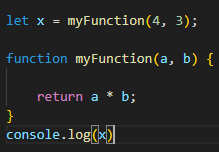
Le code à exécuter, par la fonction, est placé entre accolades : {}

**SYNTAX :**

function name(parameter1, parameter2, parameter3) {

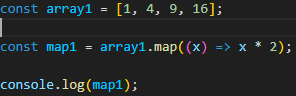
// code to be executed

}



1. **FONCTION .MAP :**

La map() méthode des Array. Instances crée un nouveau tableau rempli avec les résultats de l'appel d'une fonction fournie sur chaque élément du tableau appelant.



* **C’EST QUOI L’HTML :**

INTRODUCTION AU HTML

HTML signifie « HyperText Markup Language » qu'on peut traduire par « langage de balises pour l'hypertexte ». Il est utilisé afin de créer et de représenter le contenu d'une page web et sa structure. D'autres technologies sont utilisées avec HTML pour décrire la présentation d'une page (CSS) et/ou ses fonctionnalités interactives (JavaScript).

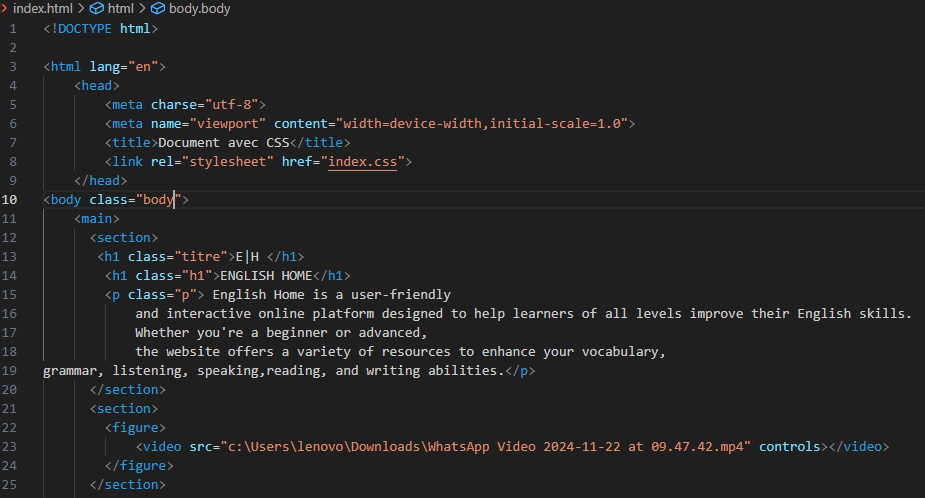
L'« hypertexte » désigne les liens qui relient les pages web entre elles, que ce soit au sein d'un même site web ou entre différents sites web. Les liens sont un aspect fondamental du Web. Ce sont eux qui forment cette « toile » (ce mot est traduit par web en anglais). En téléchargeant du contenu sur l'Internet et en le reliant à des pages créées par d'autres personnes, vous devenez un participant actif du World Wide Web.

Le langage HTML utilise des « balises » pour annoter du texte, des images et d'autres contenus afin de les afficher dans un navigateur web. Le balisage HTML comprend des « éléments » spéciaux tels que :

|  |  |
| --- | --- |
| Balise : | Définition : |
| <head> | Fournit des informations générales (métadonnées) sur le document, incluant son titre et des liens ou des définitions vers des scripts et feuilles de style. |
| <title> | Définit le titre du document (qui est affiché dans la barre de titre du navigateur ou dans l'onglet de la page). Cet élément ne peut contenir que du texte |
| <p> | Représente un paragraphe de texte. Les paragraphes sont généralement représentés comme des blocs et séparés par un espace vertical, leur première ligne est également parfois indentée. |
| <body> | Représente le contenu principal du document HTML. Il ne peut y avoir qu'un élément <body> par document. |
| <section> | Représente une section générique d'un document, par exemple un groupe de contenu thématique. Une section commence généralement avec un titre. |
| <nav> | Représente une section d'une page ayant des liens vers d'autres pages ou des fragments de cette page. Autrement dit, c'est une section destinée à la navigation dans un document (avec des menus, des tables des matières, des index, etc.). |
| <div> | Est le conteneur générique du contenu du flux. Il n'a aucun effet sur le contenu ou la mise en page tant qu'il n'est pas mis en forme d'une manière quelconque à l'aide de CSS. |
| <span> | Un conteneur générique en ligne pour les contenus phrasés. Il ne représente rien de particulier. |
| <ul> | Représente une liste d'éléments sans ordre particulier. Il est souvent représenté par une liste à puces. |
| <ol> | Représente une liste ordonnée. Les éléments d'une telle liste sont généralement affichés avec un indicateur ordinal pouvant prendre la forme de nombres, de lettres, de chiffres romains ou de points. |
| <li> | Utilisé pour représenter un élément dans une liste. Il doit être contenu dans un élément parent |
| <output> | représente un conteneur dans lequel un site ou une application peut injecter le résultat d'un calcul ou d'une action utilisateur. |

1. **Structure de base d'un document HTML :**

Pour écrire du HTML, il faut connaître les balises et la structure de base d'une page web. Un fichier HTML commence toujours par une déclaration de type de document, suivie des balises principales qui structurent le contenu.



1. **Explication des éléments :**

* <!DOCTYPE html> : Cette déclaration indique que le document est écrit en HTML5, la dernière version du langage HTML.
* <html lang="fr"> : Balise qui ouvre le document HTML. L'attribut lang="fr" indique que la langue principale du contenu est le français.
* <head> : Contient les informations sur le document, comme les métadonnées (caractères, viewport, etc.) et le titre de la page, qui apparaît dans l'onglet du navigateur.
* <meta charset="UTF-8"> : Définit l'encodage des caractères, ici UTF-8, ce qui permet d'afficher correctement les caractères spéciaux.
* <meta name="viewport" content="width=device-width, initial-scale=1.0"> : Permet d'optimiser la page pour les appareils mobiles.
* <title> : Le titre de la page qui apparaît dans l'onglet du navigateur.

**INTRODUCTION AU CSS :**

* **C’EST QUOI LE CSS :**

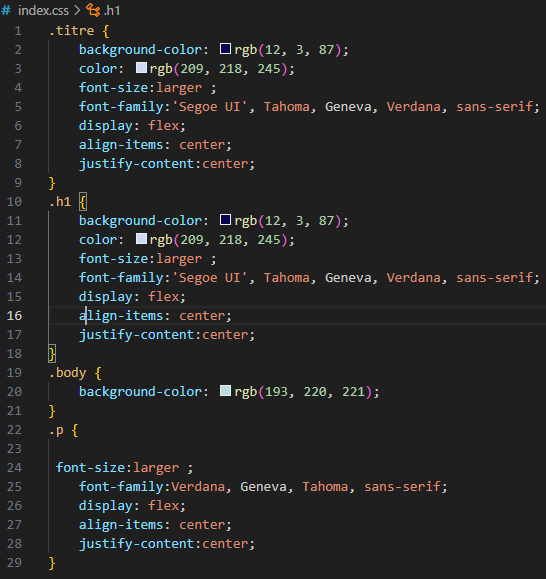
CSS (Cascading Style Sheets) est un langage utilisé pour définir la présentation et la mise en forme d'un document HTML. Tandis que HTML structure le contenu de la page, CSS permet de le rendre esthétiquement attrayant en contrôlant des aspects tels que les couleurs, les polices, les espacements, les alignements, et les dispositions. CSS offre une grande flexibilité pour concevoir des interfaces utilisateur, que ce soit pour des sites web simples ou complexes. Grâce à des propriétés comme **color**, **font-size**, **margin**, **padding**, et **background,** il est possible de personnaliser chaque élément HTML avec des styles visuels.

En outre, CSS permet de rendre les sites web adaptatifs, grâce à des fonctionnalités comme les media queries, qui ajustent l'apparence du site en fonction de la taille de l'écran.

En combinant HTML pour le contenu et CSS pour la présentation, les développeurs peuvent créer des pages web esthétiques et fonctionnelles, tout en gardant une séparation claire entre le contenu et le design.

1. **Structure de base d'un fichier CSS :**

Le CSS se compose de règles de style qui sont appliquées aux éléments HTML. Une règle de style contient un sélecteur et des déclarations.



1. **Explication des éléments** **:**

* Sélecteur : C’est l'élément HTML auquel vous appliquez le style. Par exemple, body sélectionne l'élément <body> de la page.
* Déclaration : Une déclaration se compose de deux parties : une propriété (par exemple, background-color, color) et une valeur .

1. **Ajouter des règles de style dans un fichier CSS :**

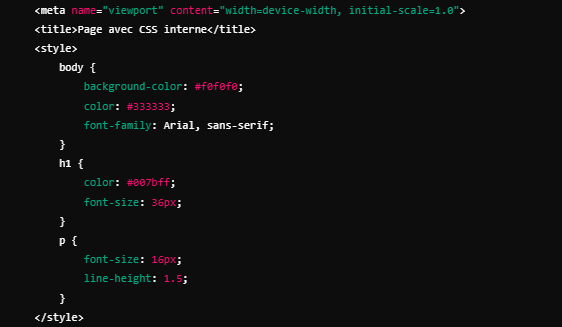
Il y a trois façons d'ajouter du CSS à une page HTML :

a. CSS externe (le plus courant)

Le CSS peut être écrit dans un fichier séparé, par exemple styles.css. Ce fichier est ensuite lié à la page HTML. ( comme l’exemple au-dessus)

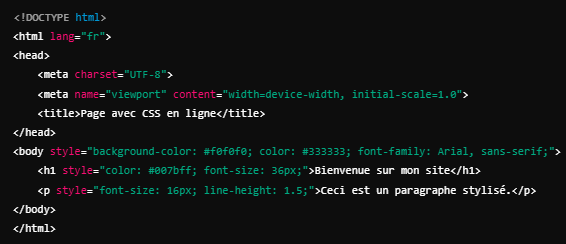
b. CSS interne

Vous pouvez également inclure des styles directement dans la section <head> de votre fichier HTML en utilisant la balise <style>.



c. CSS en ligne

Il est également possible d'ajouter du CSS directement sur un élément HTML en utilisant l'attribut style. Cependant, cette méthode est moins recommandée car elle rend le code plus difficile à maintenir.



INTRODUCTION AU GIT ET GITHUB

* C’EST QUOI GIT :

Git est un système de contrôle de version décentralisé utilisé pour gérer et suivre les modifications apportées au code source tout au long du développement d'un projet. Il permet aux développeurs de travailler sur différentes parties d'un projet de manière simultanée sans interférer les uns avec les autres. Chaque utilisateur peut avoir une copie locale complète du projet, effectuer des modifications, puis les synchroniser avec d'autres utilisateurs via des opérations comme les commits (enregistrement des changements) et les branches (version parallèle du projet). Git facilite ainsi la collaboration, la gestion des versions, et la récupération d'anciennes versions du code en cas de besoin.

* C’EST QUOI GITHUB :

GitHub, quant à lui, est une plateforme en ligne qui héberge des dépôts Git et permet aux développeurs de collaborer plus facilement sur des projets de manière centralisée. En plus de stocker le code, GitHub offre des outils de gestion de projet, comme les pull requests (propositions de modifications), les issues (suivi des problèmes) et les actions automatiques pour tester et déployer le code. GitHub permet de partager le code avec d'autres développeurs à travers des dépôts publics ou privés, et il est devenu un outil essentiel dans le développement open-source, où des millions de projets sont hébergés et maintenus par des communautés de développeurs du monde entier. En résumé, Git et GitHub travaillent ensemble pour offrir un environnement puissant de gestion de code, de collaboration et de partage de projets.

* COMMENT UTILISER GIT :

Pour utiliser Git, il faut d'abord installer Git sur votre ordinateur, puis apprendre à utiliser les commandes de base pour créer et gérer des dépôts (repositories), suivre les changements, et collaborer avec d'autres développeurs.

* + 1. Installer Git :

Vous pouvez télécharger Git à partir du site officiel. Suivez les instructions d'installation pour votre système d'exploitation (Windows, macOS, Linux).

* + 1. Créer un dépôt Git local :

Un dépôt Git est un dossier contenant votre projet et son historique de modifications. Voici les étapes pour créer un dépôt Git local :

a. Initialiser un dépôt Git :

Naviguez dans le dossier de votre projet à l'aide du terminal, puis exécutez la commande suivante pour initialiser un dépôt Git, en utilise **GIINIT**

b. Ajouter des fichiers au dépôt :

Une fois le dépôt initialisé, vous pouvez ajouter des fichiers à votre dépôt pour qu'ils soient suivis par Git. En utilise GIT ADD .

c. Faire un commit :

Un commit enregistre les modifications dans l'historique de votre dépôt. Après avoir ajouté les fichiers, vous pouvez effectuer un commit.

* + 1. Commandes utiles de GitEn utilise : git commit -m "Message décrivant les

modifications"

* Afficher l'historique des commits : **git log**
* Afficher l'état des fichiers modifiés **: git status**
* Supprimer un fichier du suivi Git **: git rm nom\_du\_fichier**
  + 1. Utiliser GitHub :

a. Créer un dépôt sur GitHub :

Allez sur GitHub, connectez-vous et créez un nouveau dépôt en ligne.

b. Connecter votre dépôt local à GitHub :

Une fois votre dépôt GitHub créé, vous pouvez le connecter à votre dépôt local :

**git remote add origin** [**https://github.com/votre\_utilisateur/nom\_du\_depot.git**](https://github.com/votre_utilisateur/nom_du_depot.git)

c. Pousser (push) vos modifications vers GitHub :

Pour envoyer vos commits locaux vers GitHub, utilisez la commande git push :

**git push -u origin main**

Conclusion :

Git est un outil puissant qui permet de suivre les versions de votre code, de collaborer avec d'autres développeurs, et de gérer les modifications de manière efficace. En combinant Git avec des plateformes comme GitHub, vous pouvez travailler en équipe, partager votre code, et profiter des fonctionnalités avancées comme les branches, les pull requests, et la gestion des versions.