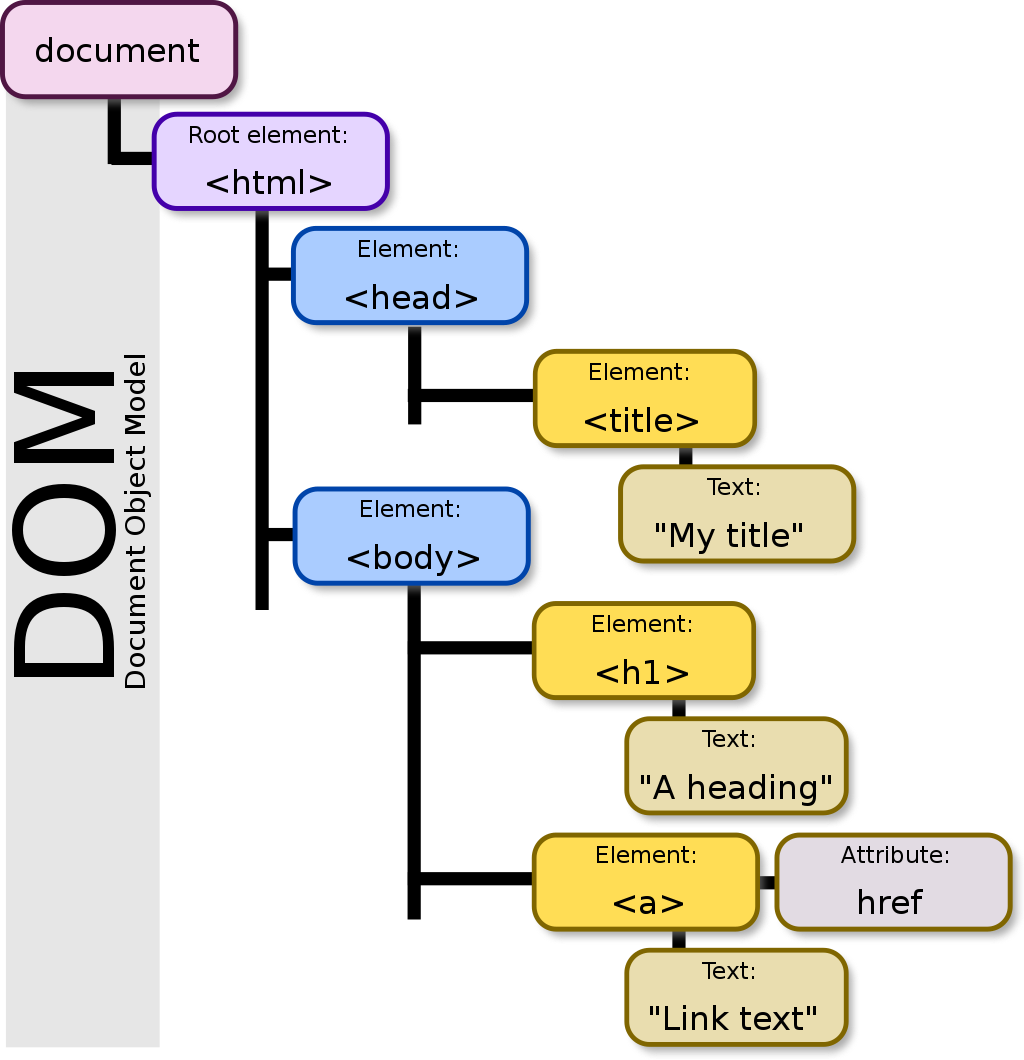
SOMMAIRE

1. Le DOM
2. Les bon outils (VSCode & extensions)
3. Les bases du JavaScript
4. Les sélecteurs
5. Le typage
6. Les fonctions
7. Les écouteurs d’évènements
8. Les boucles
9. La console c’est simplement magique
10. Interagir avec le DOM

**1) Le DOM**

Le **Document Object Model** (**DOM**) est une [interface de programmation](https://fr.wikipedia.org/wiki/Interface_de_programmation) normalisée par le [W3C](https://fr.wikipedia.org/wiki/W3C), qui permet à des scripts d'examiner et de modifier le contenu du [navigateur web](https://fr.wikipedia.org/wiki/Navigateur_web) Par le DOM, la composition d'un document [HTML](https://fr.wikipedia.org/wiki/HTML) ou [XML](https://fr.wikipedia.org/wiki/XML) est représentée sous forme d'un jeu d'objets – lesquels peuvent représenter une fenêtre, une phrase ou un style, par exemple – reliés selon une structure en [arbre](https://fr.wikipedia.org/wiki/Arbre_(informatique)). À l'aide du DOM, un script peut modifier le document présent dans le navigateur en ajoutant ou en supprimant des nœuds de l'arbre.



**2) Les bons outils**

* Les extensions qui vous aideront au quotidien :
  + Prettier
  + JavaScript (ES6) code snippets
  + Code spell checker
  + Beautify css/sass/scss
  + Jquery code snippets
  + CSS formater
  + CSS reset
  + CSS formater
  + Debugger for Chrome
  + GitHub
  + HTML snippetes
  + Live server

**3) Les bases du JavaScript**

Nous allons passer en revu notre premier module avec une intégration classique d’une interface Html, CSS et JavaScript. Ceci nous permettra dans un premier temps de comprendre comment : importer une librairie, l’utiliser et débuter ce cours avec une base qui nous permettra d’aller plus loin.

**Les trois types de déclarations :**

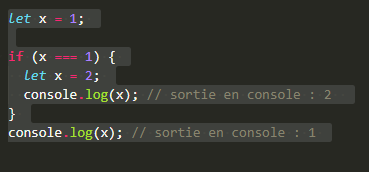
**var :**

On déclare une variable, éventuellement en initialisant sa valeur.

**let :**

On déclare une variable dont la portée est celle du bloc courant (une fonction par exemple), en initialisant éventuellement sa valeur.

Ex :



**const :**

On déclare une constante nommée, dont la portée est celle du bloc courant, accessible en lecture seule.

**Les variables :**

Les variables sont utilisées comme des noms symboliques désignant les valeurs utilisées dans l'application. Les noms des variables sont appelés identifiants. Ces identifiants doivent respecter certaines règles : elles peuvent commencer par des caractères [a-z] > [A-Z], par un \_ ou un $.

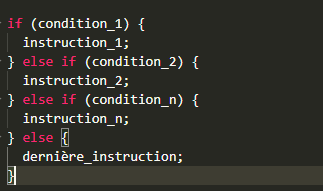
Voici quelques exemples de déclarations valides : ma\_variable, Ma\_Variable, \_maVar, $maVar

**Les conditions :**

Une instruction conditionnelle est un ensemble de commandes qui s'exécutent si une condition donnée est vérifiée. JavaScript possède deux instructions conditionnelles : if...else et switch.

**Instruction if...else**

On utilise l'instruction **if** lorsqu'on souhaite exécuter une instruction si une condition logique est vérifiée (vraie). La clause **else** est optionnelle et permet de préciser les instructions à exécuter si la condition logique n'est pas vérifiée (l'assertion est fausse). Voici un exemple qui illustre l'utilisation de l'instruction **if :**



On pourrait traduire ceci littéralement par :

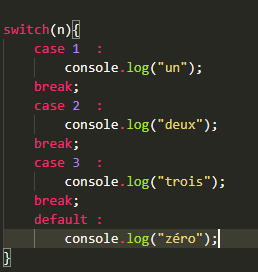
**Si** la condition 1 est remplis, l’instruction 1 est jouée, **sinon si** condition 2 est bien remplis, l’instruction 2 est jouée, **sinon** la dernière instruction sera jouée.

**Les falsy values (valeur considéré non true) :**

* false
* undefined
* null
* 0
* NaN
* la chaîne de caractères vide ("")

**Instruction switch**

L'instruction switch permet à un programme d'évaluer une expression et d'effectuer des instructions en fonction des différents cas de figures correspondants aux différentes valeurs. Si un cas correspond au résultat de l'évaluation, le programme exécute l'instruction associée.



Un exemple sympa de calculateur simplifié :

<https://www.programiz.com/javascript/switch-statement#example>

**Les boucles et itérations :**

1) Les sélecteurs

Vanilla :

L’élément initiale sera toujours “document” il est central lors de l’utilisation d’un sélecteur dans le DOM, voici nos différents sélecteurs :

querySelector

querySelectorAll

getElementById

getElementByClassName

JQuery :

Avec jQuery il est possible d’utiliser :

1°) le “$” puis en fonction de l’élément du DOM

* Un ID --> $(‘#monId)
* Une classe --> $(‘.maClasse’)
* Un élément du DOM --> $(‘button, label, h1, input’)
* Quelques exemples de sélecteurs

2°) le jQuery

* Nous remplaçons simplement “$” par “jQuery”
* Personnaliser votre sélecteur :