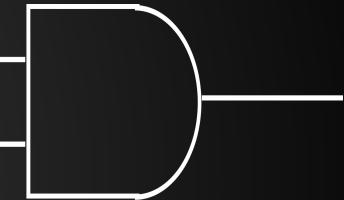


Développement Web, niveau 1

Nicolas Triau
Mail : nicolas.triau@yahoo.fr
Discord : gillesleveau

Objectif du cours

DOM



Comprendre et créer une architecture de page web

Manipuler HTML/CSS/JS
JS ne sera utilisé que dans un cadre “basique”, sans trop de logique

Premiers pas dans les logiques algorithmiques.
Compréhension des bases d'un langage de programmation

1. La page Web

Nicolas Triau
Mail : nicolas.triau@yahoo.fr
Discord : gillesleveau

La page Web

- **Le DOM**

- Le **Document Object Model** est ce qui est compris par le navigateur pour la page web
- Il fait appelle à des **objets** qui ont des **propriétés** et des **méthodes**
 - Les propriétés sont des informations, des indicateurs sur **l'instance** de l'objet. Une instance est émanation concrète de l'objet qui sert de **modèle**
 - Les méthodes sont des fonctions pouvant être utilisés par une instance d'un objet.

La page Web

- **La structure d'une page web**

- A la racine du document, il se trouve la balise **HTML** qui définit la structure de la page
- A l'intérieur, on trouve la balise **HEAD** et **BODY**
- **HEAD** va contenir des informations / indications sur la page. Le titre, liaison de fichiers CSS et/ou JS etc..
- **BODY** va contenir les éléments visibles de la page. A noter qu'il est aussi possible d'appeler des scripts et du faire du CSS directement dans le HTML

La page Web

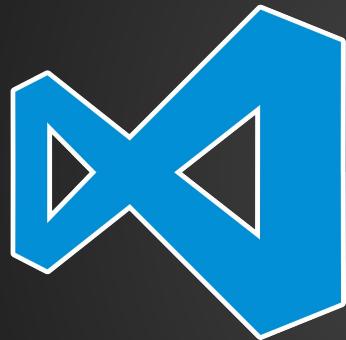
- L'inspecteur

- Afin d'analyser une page web, de regarder la structure ou de la modifier, il est possible d'utiliser **l'inspecteur**. Ce dernier possède plusieurs outils :
 - **Inspecter** qui permet de voir ce que contient la page web.
Que ce soit dans la balise head ou dans le body
 - **Console** permettant d'exécuter du code JS, de tester votre code, d'accéder à des logs d'erreurs, de warnings etc..
 - **Network**, ce dernier montrera toutes les requêtes réseau (JS, images, API etc...)

2. Faire sa page web HTML/CSS

Nicolas Triau
Mail : nicolas.triau@yahoo.fr
Discord : gillesleveau

Créer sa page web

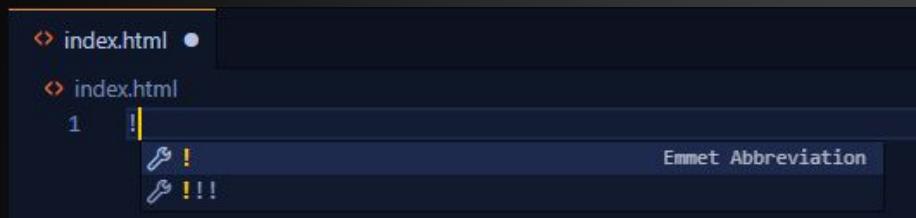


Visual Studio Code
(VSCode)



Extension Live
Server

Créer sa page web



```
↳ index.html > ⚒ html
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Document</title>
7  </head>
8  <body>
9
10 </body>
11 </html>
```

Créer sa page web

- Que voit-on ?

- **DOCTYPE** permet simplement d'indiquer au navigateur que nous utiliser HTML5
- On aperçoit la balise **HTML** qui va contenir l'ensemble de notre page
- Dans le **HEAD**, on aperçoit des balises **meta** servant à décrire au navigateur le comportement de la page web qui est déclarée
- Dans le **BODY** il n'y a actuellement rien. Mais nous pouvons construire nos pages avec des balises : <p>, <h1>, <div> etc...
- Pour lier votre document CSS à votre HTML :

```
<link rel="stylesheet" href="style.css">
```

Créer sa page web

- Utiliser le CSS

- Afin de pouvoir utiliser du CSS, **il faut cibler l'élément voulu** dans le fichier. Via un id, une class, un chemin, un nom de balise etc..
- Un sélecteur peut sélectionner plusieurs éléments à la fois
- Si vous sélectionnez un **objet parent**, les enfants **hériteront** de **certaines** propriétés CSS (color, font..)
- Chaque sélecteur se déclare différemment en CSS (#, ., chemin avec espaces)

```
<body>
|   <h1 class="my_title">Ceci est un titre</h1>
|</body>
```

```
.my_title {
    color: red
}
```

```
body h1 {
    color: blue
}
```

3. Interactions en Javascript *(Javascript != Java)*

Nicolas Triau
Mail : nicolas.triau@yahoo.fr
Discord : gillesleveau

Interagir avec Javascript

- **Les bases d'un langage de programmation**
 - **Types de données.** String, Int, Float, Array/List, Objet/Dict/Struct, None/Null
 - *String* -> Une chaîne de caractère
 - *Int/Float/Number* -> Nombre entier, nombre décimal
 - *Array/List* -> Succession de données
 - *Object/Dict/Struct* -> Associations de données avec une clé avec une valeur associée
 - *None/Null* -> Absence de donnée

Interagir avec Javascript

- **Les bases d'un langage de programmation**

- **Variables.** Principe de stocker une donnée sous un alias
- **Fonctions.** Suite d'instructions ré-utilisable (ex: attribuer un text à une balise, changer une couleur etc..)
- **Boucles (Loop).** Permet d'exécuter un bout de code X fois selon les conditions que nous déterminons

```
JS script.js > ...
1  let my_name = "Nicolas"
2
3  console.log(my_name)
4  const changeName = () => {
5      my_name = "Guido"
6  }
7  changeName()
8  console.log(my_name)
```

Exemple de code déclarant une variable ainsi qu'une fonction changeant le contenu de la variable

Interagir avec Javascript

- Interagir avec le DOM

- Testons de sélectionner notre <h1>. Dans mon exemple il possède la **class** “my_title”

```
1 const title = document.querySelector('.my_title')
2 console.log(title)
```

```
▶ <h1 class="my_title"> ⏎
```

```
script.js:2:9
```

- La balise du titre apparaît dans les logs. Vous pouvez y consulter ses propriétés, méthodes, ainsi que la ligne dans le script où le console.log a été fait

Interagir avec Javascript

- ***Interagir avec le DOM***

- Exécutons désormais une action au click de l'utilisiteur
- Il existe une méthode pour les balises HTML qui s'appelle **addEventListener** permettant de trigger une fonction dès qu'un évènement est détecté.

```
document.addEventListener("click", () => {
    title.style.color = "#0F0"
})
```

- On détecte l'événement au click sur la page web, sur le document.
- Maintenant, essayez de changer la couleur au click du text, et non pas du document

4. Projet

Nicolas Triau
Mail : nicolas.triau@yahoo.fr
Discord : gillesleveau

<https://www.fandom.com/>

- Reproduire la page Fandom (un article) de votre choix
- Intégrer en JS le bouton expand

Nicolas Triau
Mail : nicolas.triau@yahoo.fr
Discord : gillesleveau

5. Ressources dev & web

Nicolas Triau
Mail : nicolas.triau@yahoo.fr
Discord : gillesleveau

Ressources

- ***Difference Front / Back***

Front / Client :

- Gère logique & comportement utilisateurs
- HTML/CSS/JS/PHP
- VueJS, React, Next, Angular...

API :

- Porte d'entrée pour interroger le back
- Intermédiaire pour rediriger, sécuriser, ordonner

Back / Serveur :

- Gère logique business
- Calls et gestion DB
- Python / Go / JS / PHP / SQL..

Ressources

- **RGB / Hexadécimal**

- L'affichage dans un PC se fait via des pixels qui ont 3 couleurs primaires. Rouge/Red (R), Vert/Green (V/G), Bleu/Blue (B)
- L'intensité de fait via un octet chacun. Résultat d'un maximum de 255, pour 256 possibilités. Le RGB s'exprime donc de 0 à 255 pour chaque valeur.
- L'Hexadécimal est une autre forme de notation, plus compacte. Elle va de 0 à 9, puis de A à F. Spectre de 16 possibilités. Chaque couleur à 2 caractères associés. Ainsi 16x16 donnant 256 possibilités pour une valeur max de 255 par couleur.

Exemple : #5F6834 -> rgb(95, 104, 52)

Ressources

- **Naming Variables**

- Dans un langage de programmation, une variable doit avoir un nom explicite, clair, décrivant son contenu, et correspondant à la convention de nommage du projet
- Exemple de convention de nommage pour des variables :
 - camelCase
 - PascalCase
 - snake_case
 - kebab-case
 - SCRAMING_SNAKE_CASE

- **Librairies CSS & JS**
- **No-code**
- **Site ressources web (doc, couleurs, animations etc...)**
- **Codewars**
- **Hacksplaining**

Nicolas Triau
Mail : nicolas.triau@yahoo.fr
Discord : gillesleveau