

TP#4

Intégration web



Un site web ne fonctionne pas sans...

- Développeur
- Code HTML/CSS
- Navigateur
- ...design



Source de design

- Graphistes
- Gabarits
- Frameworks (CSS)



Tailwind CSS

- Framework CSS
- Projet gratuit et Open Source
- Propose de multiples classes CSS permettant d'appliquer un style cohérent
- Une classe CSS = un besoin

Source(s) :

- <https://github.com/tailwindlabs/tailwindcss>
- <https://tailwindcss.com/>



Framework

- Ensemble de composants logiciels
- Sert de fondation pour un projet
- Peut être composé de bibliothèques / *libraries*
 - Bibliothèque : collection d'éléments prêts à être utilisés
- Existe pour de multiples langages

Source(s) :

- <https://fr.wikipedia.org/wiki/Framework>



**Autrement dit un framework est
une boîte à outils.**



Tailwind CSS

Avec Tailwind CSS (ou un autre framework CSS), les développeurs n'ont plus à s'occuper du design, ils ne se contentent que du code HTML et CSS. Ainsi, ils peuvent obtenir rapidement un beau site web opérationnel.

Source(s) :

- <https://github.com/tailwindlabs/tailwindcss>
- <https://tailwindcss.com/>



Tailwind CSS - Concurrents

- Bootstrap par Twitter
 - Premier framework CSS populaire
- Foundation
- Windicss
- ...



Tailwind CSS

- Gère le responsive design
 - Appliquer un style différent en fonction de la taille de l'écran
- Permet d'étendre les fonctionnalités avec son propre code
 - Nous ne verrons pas ceci en cours
- Dernière version : 4 (01/2025)
- Documentation en anglais : **Pensez à la consulter**
- Utilise le rem comme unité de mesure
- Propose un système de grille adaptable

Source(s) :

- <https://github.com/tailwindlabs/tailwindcss>
- <https://tailwindcss.com/>



Tailwind CSS - Installation



```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Tailwindcss exemple</title>
  <script src="https://unpkg.com/@tailwindcss/browser@4"></script>
</head>
<body>
  <main></main>
</body>
</html>
```



Tailwind CSS - Installation

```
<script src="https://unpkg.com/@tailwindcss/browser@4"></script>
```

Le CSS de Tailwind CSS est chargé via une balise `<script>`. C'est différent de ce que nous avons vu jusqu'à présent en cours. Mais le principe le reste le même, on charge des classes CSS pour les utiliser dans le HTML.



**Et voilà !
Tailwindcss est prêt à être utilisé.
Il suffit d'utiliser ses classes.**



Une classe = un besoin

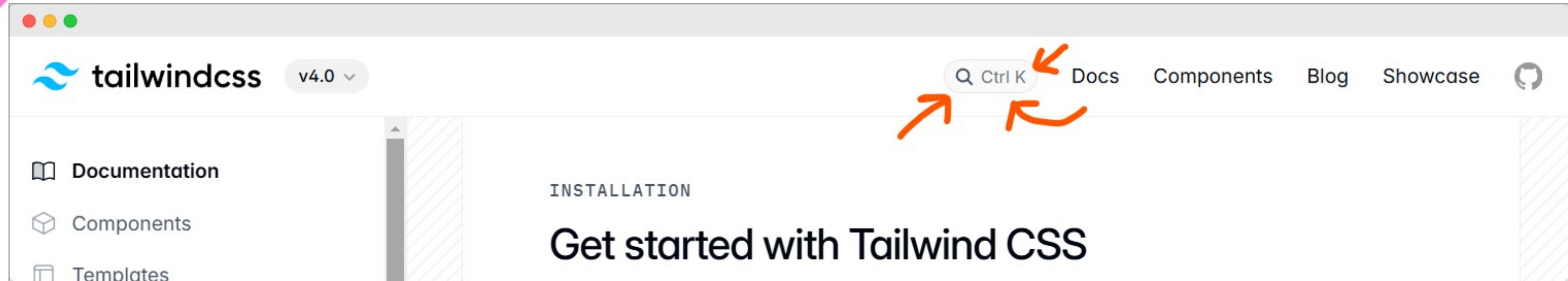
- Tailwindcss contient des milliers de classes CSS
- Chaque classe à un rôle
- Les classes suivent une nomenclature précise
- On ne perd plus son temps à nommer des classes
- Rend le code HTML très verbeux
- Vos balises auront plein de classes CSS
- La documentation, meilleure alliée pour utiliser le framework
- N'hésitez pas à utiliser la recherche

Source(s) :

- <https://github.com/tailwindlabs/tailwindcss>
- <https://tailwindcss.com/>



Une classe = un besoin



Il y a beaucoup de classes avec Tailwind CSS.
N'hésitez pas à utiliser le moteur de recherche pour trouver les classes.

Source(s) :

- <https://github.com/tailwindlabs/tailwindcss>
- <https://tailwindcss.com/>



Une classe = un besoin



```
<h1 class="text-3xl font-bold tracking-tight text-gray-900">  
    Mon super texte  
</h1>
```

Avec Tailwind CSS, on associe beaucoup de classes CSS.
Et les éléments HTML ont souvent beaucoup de classes CSS.

(Ici le texte sera très gros, gras, couleur grise et les lettres proches)

Source(s) :

- <https://github.com/tailwindlabs/tailwindcss>
- <https://tailwindcss.com/>



Une classe = un besoin

Taille de police

Espace entre les lettres

Graisse du texte

Couleur de texte

```
<h1 class="text-3xl font-bold tracking-tight text-gray-900">  
  Mon super texte  
</h1>
```

Source(s) :

- <https://github.com/tailwindlabs/tailwindcss>
- <https://tailwindcss.com/>



Classes conditionnelles

- Permet d'appliquer une classe sous condition : survol, focus, thème sombre...
 - Fonctionne avec toutes les classes
- Utilise un système de modifier
- La syntaxe est la suivante :

modifier:classe
(tout collé)

Source(s) :

- <https://tailwindcss.com/docs/hover-focus-and-other-states>
- <https://tailwindcss.com/docs/hover-focus-and-other-states#appendix>



Classes conditionnelles - Syntaxe

Le modifier. Ici on applique la classe "bg-sky-700" lorsque notre bouton est survolé.

Il ne faut pas oublier les deux-points (:) après le *modifier*



```
<button class="bg-sky-500 hover:bg-sky-700">  
  Télécharger SAE  
</button>
```

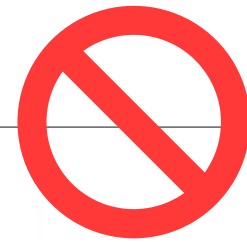
Source(s) :

- <https://tailwindcss.com/docs/hover-focus-and-other-states>
- <https://tailwindcss.com/docs/hover-focus-and-other-states#appendix>



Classes conditionnelles - Syntaxe

```
<button class="bg-sky-500 hover:bg-sky-700:font-bold">
  Télécharger SAE
</button>
```



Interdit

Mettre plusieurs classes sur le même *modifier*

```
<button
  class="bg-sky-500 hover:bg-sky-700 hover:font-bold">
  Télécharger SAE
</button>
```



Correct

Associer un *modifier* à chaque classe

Source(s) :

- <https://tailwindcss.com/docs/hover-focus-and-other-states>
- <https://tailwindcss.com/docs/hover-focus-and-other-states#appendix>



Classes conditionnelles

- Rôles possibles (liste non exhaustive) :
 - pseudo-classes (:hover, :focus...)
 - pseudo-éléments (::before, ::selection...)
 - media-queries (responsive, dark mode...)
 - attributs (required, inert...)
 - ...

Source(s) :

- <https://tailwindcss.com/docs/hover-focus-and-other-states>
- <https://tailwindcss.com/docs/hover-focus-and-other-states#appendix>



Pratiquons ! - Initiation Tailwind CSS

(Partie 1/2)

Pré-requis :

- Avoir la ressource `ressources/tailwind`

A télécharger ici : https://github.com/DanYellow/cours/raw/refs/heads/main/s2-integration-web/travaux-pratiques/numero-4/s2-integration-web_travaux-pratiques_numero-4.ressources.zip



Responsive design

- Adaptation de la mise en page en fonction de la taille de l'écran
- Utilisation de media-queries
- Un code HTML pour tous les terminaux

Sujet abordé plus en profondeur en S3

Source(s) :

- https://developer.mozilla.org/fr/docs/Web/CSS/Media_Queries/Using_media_queries



Responsive design avec Tailwind CSS

Modifieur	Largeur minimum
sm:	640px
md:	768px
lg:	1024px
xl:	1280px
2xl:	1536px

Source(s) :

- https://developer.mozilla.org/fr/docs/Web/CSS/Media_Queries/Using_media_queries
- <https://tailwindcss.com/docs/responsive-design>



Responsive design avec Tailwind CSS



```
<img class="w-16 md:w-32 lg:48" src="" alt="">
```

Ici notre balise `` aura une taille de `w-16` par défaut, de `w-32` pour un écran ayant une largeur minimale de 768px (md:) et de `w-48` pour un écran ayant une largeur minimale de 1024px (lg:)

Source(s) :

- https://developer.mozilla.org/fr/docs/Web/CSS/Media_Queries/Using_media_queries
- <https://tailwindcss.com/docs/responsive-design>



Tester son site mobile sur PC

- Le mode responsive (ou vue adaptative) des navigateurs permet de tester son site à des résolutions différentes
 - **Ce mode garde le rendu du navigateur**
- Sur macOS, il est possible de brancher son iPhone / iPad et tester avec l'inspecteur de Safari
 - Ou d'utiliser le simulateur iOS (nécessite Xcode)

Source(s) :

- <https://www.idownloadblog.com/2019/06/21/how-to-use-safari-web-inspector-ios-mac/> - anglais
- <https://developers.google.com/web/tools/chrome-devtools/remote-debugging> - anglais



Tester son site mobile sur PC

- Sur Windows / macOS, il est possible de brancher son smartphone Android et de tester avec l'inspecteur de Chrome
- Raccourci clavier : ctrl + shift/maj + m (Chrome / Brave / Firefox)

Source(s) :

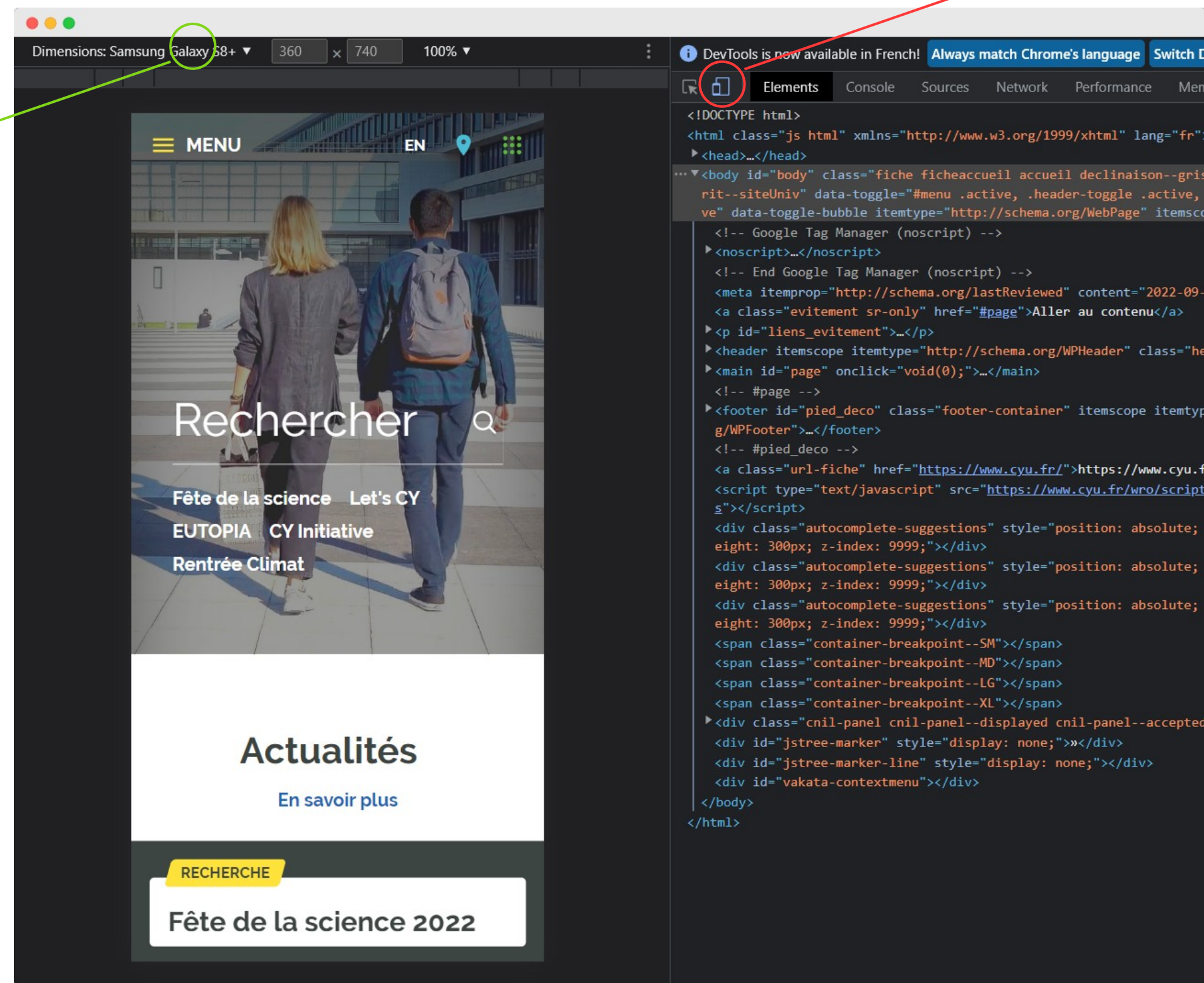
- <https://www.idownloadblog.com/2019/06/21/how-to-use-safari-web-inspector-ios-mac/> - anglais
- <https://developers.google.com/web/tools/chrome-devtools/remote-debugging> - anglais



Tester son site mobile sur PC

Activation du mode responsive

Changer d'appareil



Source(s) :

- <https://www.idownloadblog.com/2019/06/21/how-to-use-safari-web-inspector-ios-mac/> - anglais
- <https://developers.google.com/web/tools/chrome-devtools/remote-debugging> - anglais



Valeurs arbitraires

- Permettent d'étendre les fonctionnalités de base
- Créent des classes à la volée
- Utilisent la même syntaxe que Tailwind CSS

Source(s) :

- <https://tailwindcss.com/docs/adding-custom-styles#using-arbitrary-values>



Valeurs arbitraires



```
<div class="bg-[#365ffa] text-[1.15rem]">  
  <!-- [...] -->  
</div>
```

On crée deux nouvelles classes :

- Une classe background-color avec la couleur `#365ffa`
- Une classe font-size avec la taille 1.15rem

Source(s) :

- <https://tailwindcss.com/docs/adding-custom-styles#using-arbitrary-values>

Questions ?

