

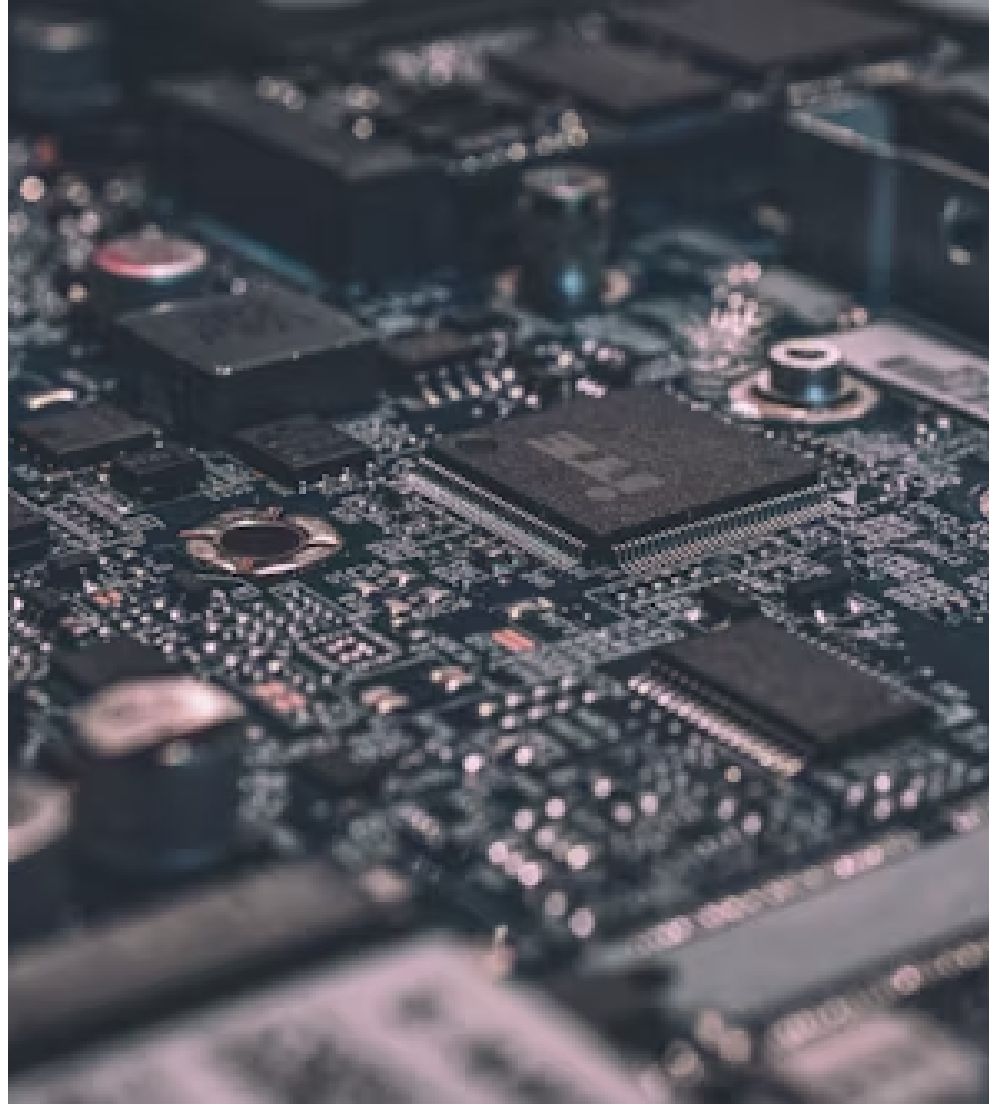
HNU 6058

Université de Montréal

Technologies du web

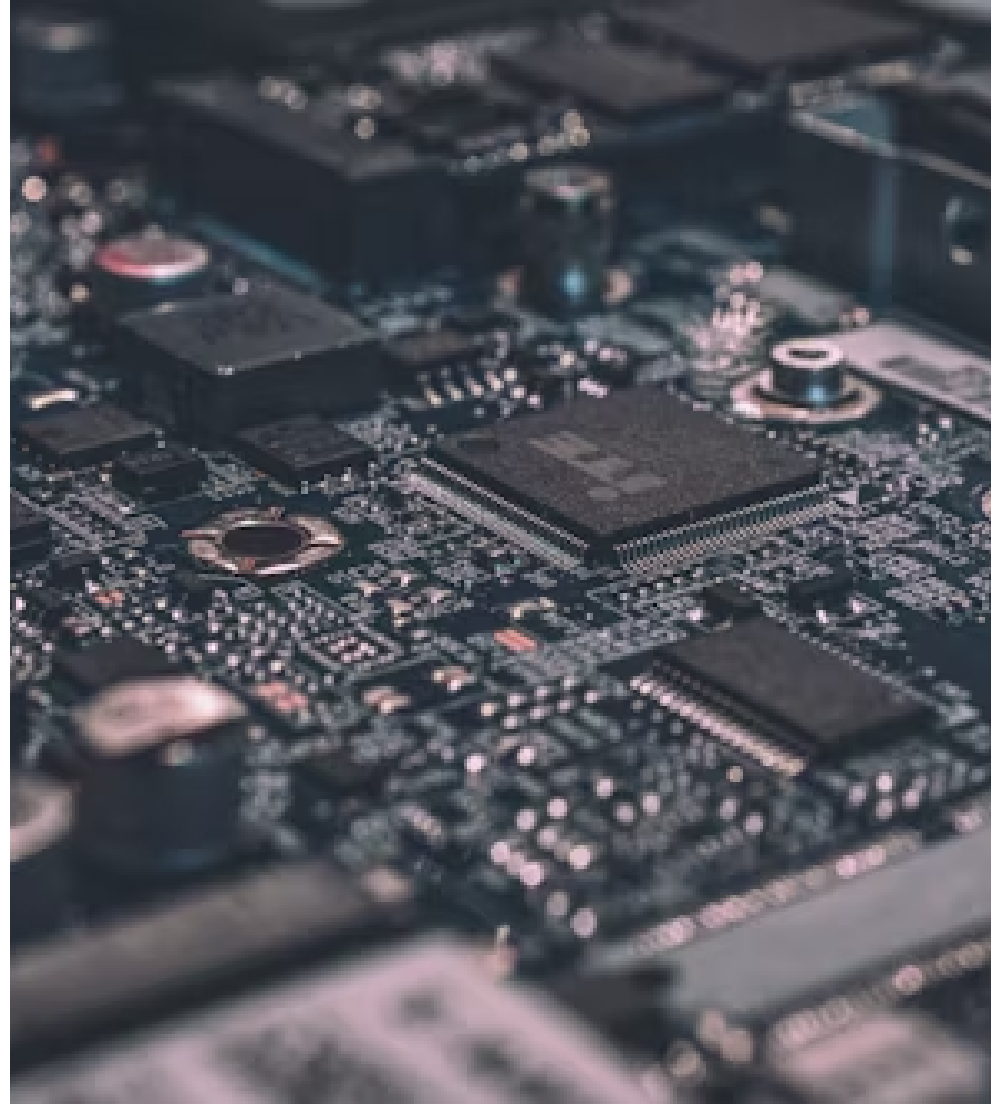
Séance du 14 février 2024

[Le site web du cours](#)



Rappel

1. Liens et images en HTML
2. Tableaux en HTML
3. Formulaires
4. Structure d'un futur CV en ligne



Objectifs de la séance

14 février

- Distinguer la forme du contenu avec CSS
- La logique des feuilles de style en cascade
- La syntaxe du CSS
- Atelier CSS : Mise en forme de votre site HTML (1re partie)
- Atelier CSS : Mise en forme de votre site HTML (2e partie)
- Maintenir le cap : les meilleures pratiques en matière de CSS

```
@keyframes ripple {  
  from {  
    width: .1%;  
    height: .1%;  
    opacity: 1;  
  }  
  to {  
    width: 100%;  
    height: 100%;  
    opacity: 0;  
  }  
}
```

Un peu de contexte

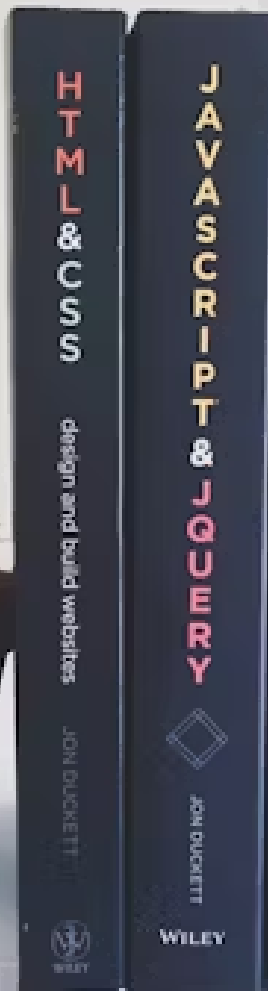
The only way to modify rendering of HTML elements in older browsers was through global settings of the browser. However, early HTML specifications attempted to put stakes in the ground toward rendering abstractions to be achieved in the future (e.g., the distinction between B and STRONG).



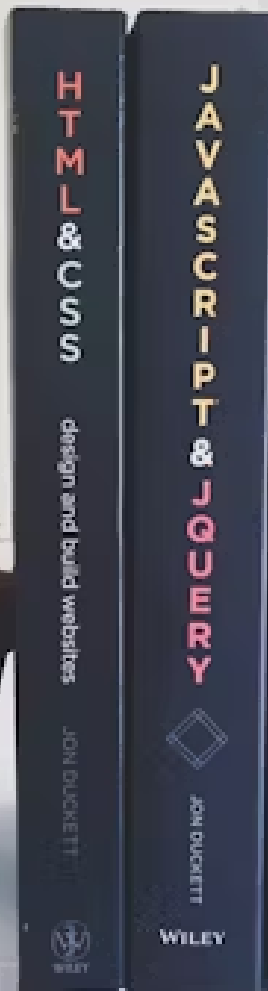
Historique

- 1994 : Håkon Wium Lie, au CERN, commence à travailler sur le CSS
- 1996 : Le W3C publie la première norme pour CSS Cascading Style Sheets
- 1999 CSS3 : couleurs , RGBA, HSLA , coins arrondis (`border-radius:`), ombres (hauteur, largeur, flou), dégradés, images d'arrière-plan multiples
- fonctionnement « à cascade » : Indique la priorité avec laquelle les informations sont traitées

Source

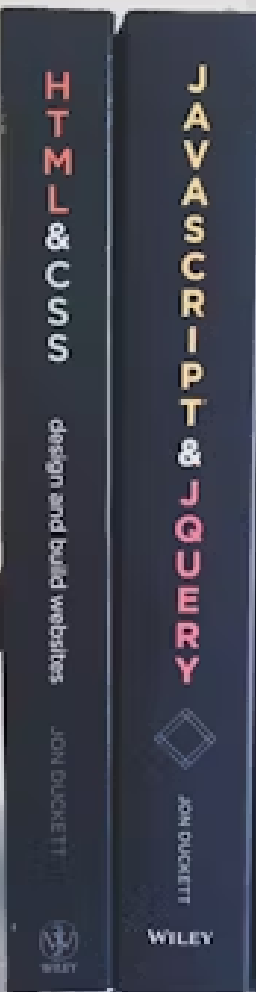


- La séparation de la structure du document de sa mise en page était un objectif du HTML depuis sa création en 1990 par Team Berners Lee
- Il existait déjà des technologies permettant de réguler le style d'une page (Word a été lancé en 1983, TeX en 1979), mais pas dans le web
- Selon l'idée originale, chaque navigateur aurait dû produire sa propre feuille de style
- Encouragé par Dave Ragget (le principal architecte de HTML3.0), Håkon a publié la première version de CSS 3 jours avant la présentation de Netscape Navigator



- 1995 : création de la liste de diffusion www-style et premier Workshop dédié à CSS
- 1996 : Internet Explorer et Netscape en concurrence pour prendre en charge CSS

Source



Points de discussion après première présentation

- Équilibre entre les préférences de l'auteur et celles de l'utilisateur
- CSS n'est pas un langage de programmation complet

Source

Qu'est-ce que le CSS ?

- Mise en forme VS structuration du contenu
- Individuation de plusieurs niveaux d'abstraction (tag, class, id)
- Gestion des fonts
- Gestion des couleurs (noms des couleurs, RGB, RGBA, HSL, notation hexadécimale)
- Gestion de propriétés du texte
- Propriétés des « boîtes » qui organisent le texte
- Propriétés de classification, par exemple : `p {display : list-item};`

```
n-reader-text:hover,  
n-reader-text:active,  
n-reader-text:focus {  
background-color: #f1f1f1;  
border-radius: 3px;  
box-shadow: 0 0 2px 2px rgba(0, 0, 0, 0.6);  
display: auto !important;  
color: #21759b;  
display: block;  
font-size: 14px;  
font-size: 0.875rem;  
font-weight: bold;  
height: auto;  
margin: 5px;  
line-height: normal;  
padding: 15px 23px 14px;  
text-decoration: none;  
width: 5px;  
width: auto;  
z-index: 100000; /* Above WP toolbar. */
```

CSS : mettre en forme (mais pas que)

- appliquer une mise en forme à partir d'une sémantique
- des règles imbriquées
- langage spécifique à un domaine (DSL)
- langage déclaratif (VS langages impératifs)
- langage de Turing incomplet

Une question se pose alors : les langages CSS et HTML sont-ils des langages de programmation ?

```
n-reader-text:hover,  
n-reader-text:active,  
n-reader-text:focus {  
background-color: #f1f1f1;  
border-radius: 3px;  
box-shadow: 0 0 2px 2px rgba(0, 0, 0, 0.6);  
display: auto !important;  
color: #21759b;  
display: block;  
font-size: 14px;  
font-size: 0.875rem;  
font-weight: bold;  
height: auto;  
margin: 5px;  
text-align: normal;  
width: 15px 23px 14px;  
text-decoration: none;  
text-align: center;  
font-size: 5px;  
text-align: auto;  
z-index: 100000; /* Above WP toolbar. */
```

Les trois critères de traitement pour un document CSS

- Traitement à cascade
- Priorité à la spécificité
- L'héritage des propriétés

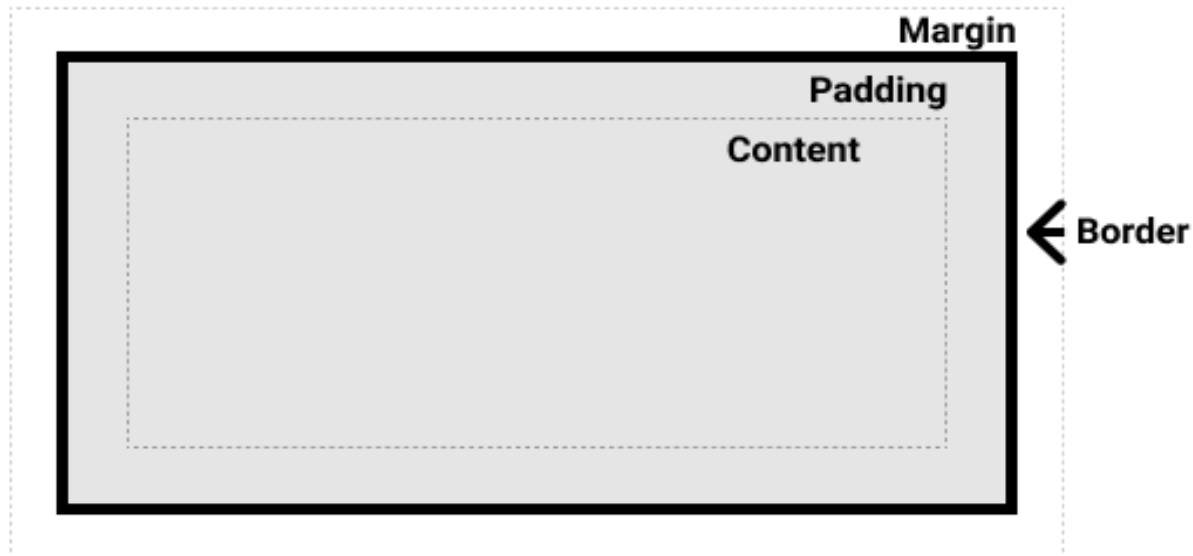
N.B.: seules quelques propriétés sont héritées, principalement celles liées à la mise en forme du texte – et pas celles de position ou de mise en page.

Pour en savoir plus : voir [ICI](#)

```
n-reader-text:hover,  
n-reader-text:active,  
n-reader-text:focus {  
background-color: #f1f1f1;  
border-radius: 3px;  
box-shadow: 0 0 2px 2px rgba(0, 0, 0, 0.6);  
color: auto !important;  
color: #21759b;  
display: block;  
font-size: 14px;  
font-size: 0.875rem;  
font-weight: bold;  
height: auto;  
margin: 5px;  
padding: 15px 23px 14px;  
text-decoration: none;  
width: 5px;  
width: auto;  
z-index: 100000; /* Above WP toolbar. */
```

CSS : Concepts fondamentaux

- Sélecteurs = élément, `.class`, `#id`
- Le modèle de boîte = padding, margin, border
- Display = inline (`span`, `i`), block (`div`, `p`), flexbox, grid



Dans la pratique...

Lier une feuille de style à une page CSS

CSS externe

```
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet" href="chemin-vers-le-fichier-style.css">
  </head>
  <body>

    </body>
</html>
```

Autres approches :

- CSS interne : balise `<style></style>` dans l'entête du document HTML
- CSS en ligne : attribut `style` dans les balises HTML. Par exemple : `<h1 style="color:blue;">Un titre bleu</h1>`



Atelier CSS : mettons en forme notre CV !

Nous nous concentrons en particulier sur la balise `grid`

pour structurer une liste avec une correspondance **clé-valeur**

et pour **adapter** le contenu de notre liste à différents types d'écrans.

D'autres éléments HTML pour la structuration du CV

- `section`
- `span`
- `header`
- `section`
- `dl - dd - dt` : Liste de définitions
- `hr` : règle horizontale
- `footer`
- `rel` : spécifie la relation entre le document actuel et le document lié.

Des éléments CSS pour la structuration du CV

Body

- `margin: 1rem auto` : valeur absolu pour centrer le contenu sur la page et limiter sa largeur
- `max-width: 44rem` : valeur absolu pour centrer le contenu
- `padding: 1rem` : Pour éviter que le texte soit collé sur le bord de la fenêtre

Des éléments CSS pour la structuration du CV

Head

- `text-align: center`
- `border-top: 6px double currentColor;` : Bordure décorative

dl - dt - dd

- description list – description term – description definition
- `display: grid` : Vs. table et vs. flexbox
- `repeat` : Function CSS pour attribuer une valeur à toutes les 12 colonnes du grid
- `1fr` : L'espace disponible est divisé uniformément pour chaque colonne du grid
- `baseline` : La ligne de laquelle découle la hauteur totale des éléments

grid

- `grid-template-rows` : `repeat(3, auto);`
- `auto` : L'élément occupe l'espace suffisant pour exister
- `1fr` : L'élément prend tout l'espace disponible, à moins que quelque chose d'autre n'apparaisse
- `row-gap: 0.25rem;`
- `gap: .25rem .5rem;`
- `column-gap: 0.5rem;`

Sélecteurs

- `.entree + .entree` : une classe qui suit une autre classe
- `.entree .nom` : un élément qui a une classe (nom) qui descend d'une autre classe (entree)
- `.entree.nom` : un élément qui a la classe entree et la classe nom
- `<` : section qui est child direct de l'élément article
- `#entree`

Padding VS Margin

- `padding: 1.5rem 0;` : padding-top et padding-bottom 1.5, padding-left et padding-right 0

Fonts formats

TTF True Type Font (.ttf)

- 1989
- Windows et Macintosh
- Les données pour l'impression et les données de l'écran sont dans un seul fichier
- Facilité d'installation et d'utilisation

Fonts formats

OTF OpenType Font (.oft)

- 1997
- Prend en charge des fonctions de typographie plus avancées.
- Par Microsoft et Adobe en tant qu'extension du format de police TrueType d'Apple

Fonts formats

WOFF Web Open Font Format (.woff)

- 2009
- Mozilla
- Les données des polices sont compressées, ce qui permet aux sites utilisant WOFF d'utiliser moins de bande
- Chargement plus rapide des pages Web
- Plus riche à niveau des métadonnées : par exemple, des informations sur la licence

Fonts formats

Font-weight

- 300 : light
- 400 : normal
- 500 : medium
- 600 : semi-bold
- 700 : bold
- 900 : heavy
- lighter/bolder : poids relatifs, par rapport à la police du parent

Fonts formats

Importation des fonts dans un document CSS

```
@font-face {  
  font-family: 'Raleway-Light';  
  src: url('raleway-light.ttf') format('truetype');  
}
```

Box shadow

```
box-shadow: 0 0 10px rgba(0, 0, 0, .15);
```

Pseudo-éléments

`dt::after`

- Les pseudo-éléments CSS : pratiques pour ajouter de la décoration supplémentaire.
- Le contenu d'un pseudo-élément ne sera **pas lu par des lecteurs d'écrans ou copié dans le presse-papier** si vous sélectionnez le texte.
- Dans notre cas, nous structurons la définition comme suit: `Term: Definition`

Variante

- `@import`
- `grid-template-columns: [colonne-secondaire] 4fr [colonne-principale] 8fr;`

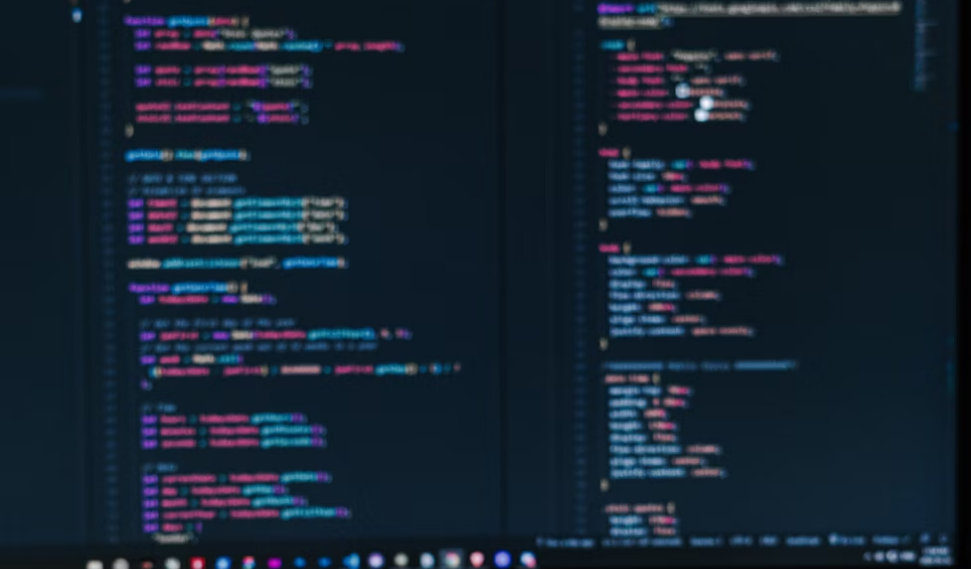
Les bonnes pratiques pour structurer un document en CSS



CSS RESET pour enlever les valeurs de default des navigateurs Web

Eric Meyer nous propose un exemple de CSS Reset

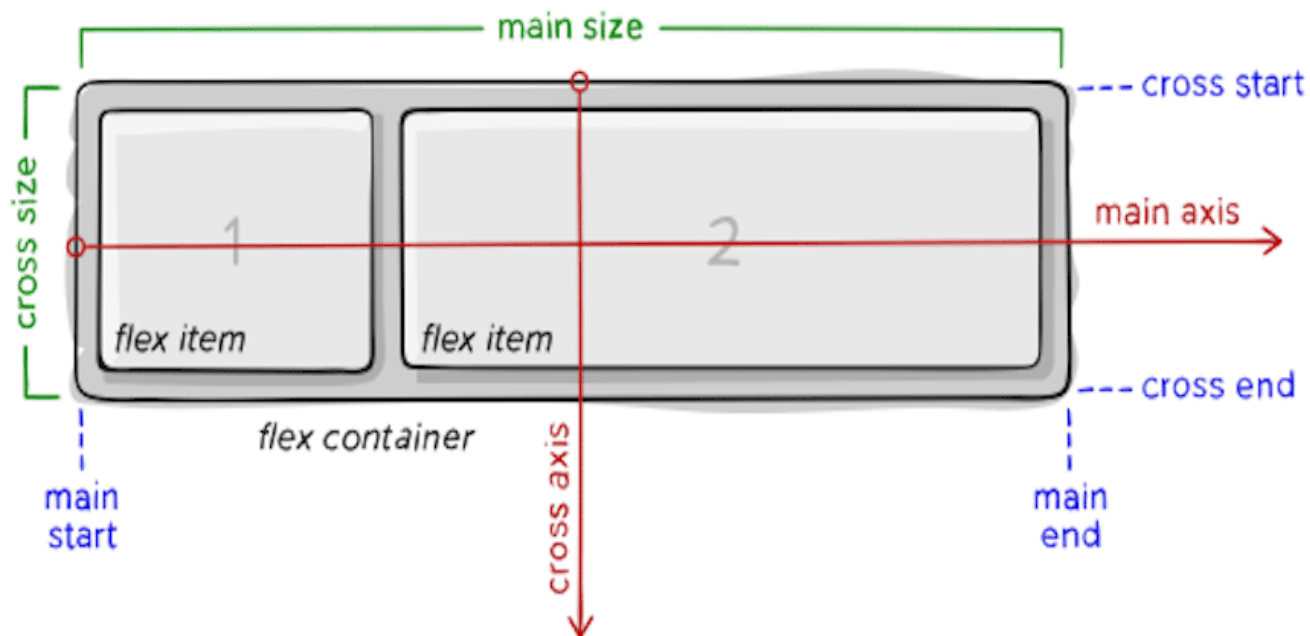




- Méthode DRY (Do not repeat yourself)
- Des sections pour chaque style dans le HTML (texte, liste, link, navigation, etc.)

Pour aller plus loin

Flexbox



Pour un design « responsive » :

@media

Automatiser la cohérence des couleurs grâce à des variables

Ajouter des animations à votre contenu

Une usage plus « moderne »
: générateurs de sites
statiques

Le plus simple possible : Pandoc et Bash

1. Installer Pandoc, un logiciel de conversion entre formats
2. Créer le dossier `mon-site`
3. Dans le dossier, créer le fichier `accueil.md` et deux autres fichiers pour le reste du contenu du site, par exemple `page-1.md` et `page-2.md`.
4. Dans le même dossier, créer un fichier `variables.yml`, pour saisir les métadonnées de votre site
5. Dans le même dossier, créer un fichier bash `site.sh`. Vous en aurez besoin pour saisir les commandes Pandoc pour convertir vos fichiers markdown en HTML.

6. Ouvrez le terminal, allez dans le dossier de votre site (commande `cd chemin/vers/mon-dossier`) et exécutez la commande `bash site.sh`
7. Pour visualiser le site localement, ouvrez via Live Server la page "accueil.html", générée par Pandoc.
8. Pour mettre à jour le site, il suffit d'éditer les fichiers au format Markdown, puis de reconstruire le site en utilisant la commande `bash site.sh` dans votre terminal

Vous trouverez sur la page StudiUM du cours une site très basique généré avec cette approche

Autres exemples de générateurs de sites statiques :

- 11ty (JavaScript)
- Hugo (Go)
- Nikola (Python)
- Jekyll (Ruby)
- Pelican (Python)