

HTML/CSS/JavaScript

M. Tellene

HTML/CSS/JavaScript

Un peu de terminologie

Internet	•	• Application (web) permettant de chercher des pages
Navigateur web	•	• Ensemble de réseaux
Fichier HTML	•	• Chaîne de caractère permettant de trouver et d'identifier une ressource
URL	•	• Logiciel permettant de consulter des pages web
Base de données	•	• Fichier (texte) qui peut être lu et affiché par un navigateur
Moteur de recherche	•	• Ensemble exploitable de données

HTML/CSS/JavaScript

Un peu de terminologie

Internet : Ensemble de réseaux

Navigateur web : Logiciel permettant de consulter des pages web

Fichier HTML : Fichier (texte) qui peut être lu et affiché par un navigateur

URL : Chaîne de caractère permettant de trouver et d'identifier une ressource

Base de données : Ensemble exploitable de données

Moteur de recherche : Application (web) permettant de chercher des pages

HTML/CSS/JavaScript

Les langages HTML et CSS sont à la base du fonctionnement de tous les sites web. Quand vous consultez un site avec votre navigateur, il faut savoir que, en coulisses, des rouages s'activent pour permettre au site web de s'afficher.

HTML/CSS/JavaScript

Les langages HTML et CSS sont à la base du fonctionnement de tous les sites web. Quand vous consultez un site avec votre navigateur, il faut savoir que, en coulisses, des rouages s'activent pour permettre au site web de s'afficher.

L'ordinateur se base sur ce qu'on lui a expliqué en HTML et CSS pour savoir ce qu'il doit afficher, comme le montre le dessin suivante.

HTML/CSS/JavaScript

HTML et CSS sont deux « langues » qu'il faut savoir parler pour créer des sites web. C'est le navigateur web qui fera la traduction entre ces langages et ce que vous verrez s'afficher à l'écran.

Mais pourquoi deux langages pour créer des sites web ?

HTML/CSS/JavaScript

HTML et CSS sont deux « langues » qu'il faut savoir parler pour créer des sites web. C'est le navigateur web qui fera la traduction entre ces langages et ce que vous verrez s'afficher à l'écran.

Mais pourquoi deux langages pour créer des sites web ?

Un pour la structure et un pour le style

HTML/CSS/JavaScript

Pour créer un site web, on doit donner des **instructions** à l'ordinateur. Il ne suffit de taper le texte qui devra figurer dans le site (comme on le ferait dans un traitement de texte Word, par exemple), il faut aussi indiquer où placer ce texte, insérer des images, faire des liens entre les pages, etc.

HTML - Généralité

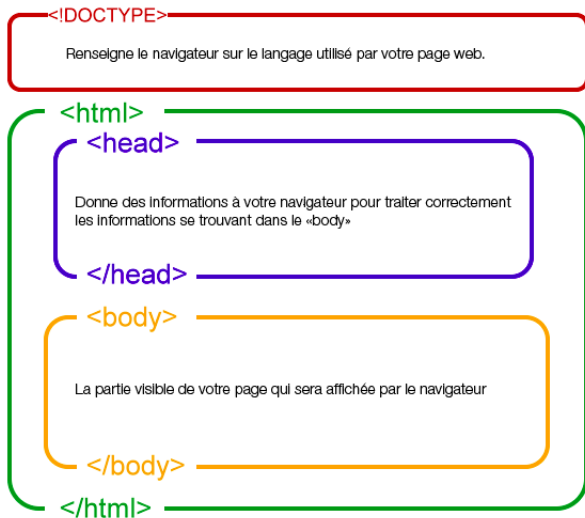
Quelques généralités sur le langage CSS :

- HTML = HyperText Markup Language
- Langage de balisage



HTML - Structure

Structure d'un fichier HTML :



HTML - Les balises

Le HTML est un langage permettant de structurer les pages web. Il est catégorisé comme étant un langage de balisage

HTML - Les balises

Le HTML est un langage permettant de structurer les pages web. Il est catégorisé comme étant un langage de balisage

Balise (ou tag) : suite de caractères permettant de définir un "bloc" ainsi que son type.

HTML - Les balises

Qu'est-ce qu'une balise ?

- Une balise est un mot-clé entre < et >
- Il existe des balises pour tout
- Il existe deux types de balises :
 - les balises ouvrantes et fermantes
 - les balises orphelines

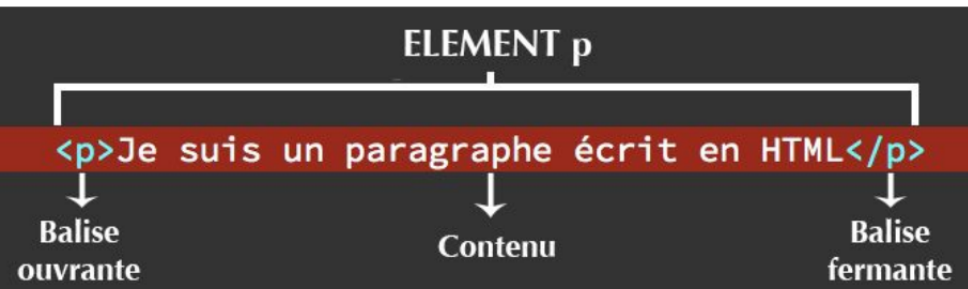
HTML - Les balises

Balises ouvrantes et fermantes :

- Balises qui vont **par paire**
- La première de la forme `<mot_clé>` est dite ouvrante
- La seconde de la forme `</mot_clé>` est dite fermante

HTML - Les balises

Exemple de balise ouvrante et fermante : l'élément p



HTML - Les balises

Balises ouvrantes et fermantes :

- Balises qui vont **par paire**
- La première de la forme `<mot_clé>` est dite ouvrante
- La seconde de la forme `</mot_clé>` est dite fermante

ex : `<body>`, `<title>`, `<p>`, ...

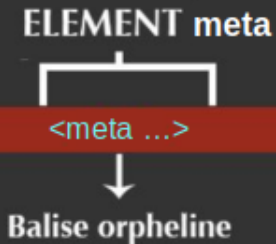
HTML - Les balises

Balises orphelines :

- Balises **seules**
- À la fois balise ouvrante et fermante
- De la forme `<mot_cle>`

HTML - Les balises

Exemple de balise orpheline : l'élément meta



HTML - Les balises

Balises orphelines :

- Balises **seules**
- À la fois balise ouvrante et fermante
- De la forme `<mot_cle>`

ex : `<link>`, `<meta>`, ...

Programmer en HTML

Quelques balises utiles

CSS - Généralité

Quelques généralités sur le langage CSS :

- CSS = Cascading Style Sheets
- Présent dès l'origine du World Wide Web
- Devient couramment utilisé dans les années 2000



Le CSS sert à mettre en forme et à styliser les pages web

Le CSS sert à mettre en forme et à styliser les pages web

Il peut se mettre à 3 endroits possibles

- Dans la balise dont on veut modifier le style
- Dans la partie HEAD du fichier HTML
- Dans un fichier à part

CSS

Où mettre du CSS ?

1ère façon : dans la balise dont on veut modifier le style

```
<p>Un texte en couleur par défaut</p>
```

Un texte en couleur par défaut

```
<p style="color: green;">Un texte en vert</p>
```

Un texte en vert

```
<p style="color: green; text-decoration: underline;">Un texte en vert</p>
```

Un texte en vert

CSS

Où mettre du CSS ?

1ère façon : dans la balise dont on veut modifier le style

Une méthode parfaite ?

CSS

Où mettre du CSS ?

1ère façon : dans la balise dont on veut modifier le style

Une méthode parfaite ?

W3Schools lists 228 of them. HTML Dog lists 125. [Our almanac](#) has 176, and I know we omit stuff on purpose (e.g. we file `margin-left` under `margin` instead of making its own entry).

[The horse's mouth?](#)

// *520 distinct property names from 66 technical reports and 66 editors' drafts.*

Source : <https://css-tricks.com/how-many-css-properties-are-there/>

CSS

Où mettre du CSS ?

2ème façon : dans la partie HEAD du fichier HTML

```
<style>
  selecteur {
    code css;
    un autre code css;
  }
</style>
```

```
<style>
  p {
    color: green;
    text-decoration: underline;
  }
</style>
```

```
<p>Un texte</p>
```

Un texte

Où mettre du CSS ?

2ème façon : dans la partie HEAD du fichier HTML

Une méthode parfaite ?

Où mettre du CSS ?

2ème façon : dans la partie HEAD du fichier HTML

Une méthode parfaite ?

Pas forcément, cela peut rendre code HTML peu lisible

Démonstration pour la troisième méthode

Programmer en CSS

Quelques propriétés utiles

Les sélecteurs

Le CSS permet de choisir un élément précis à styliser

Nous allons voir 2 sélecteurs :

- les identifiants
- les classes

Les sélecteurs

Les identifiants doivent être **uniques**

Ils sont définis dans le HTML avec l'attribut `id`

Ils sont récupérés dans le CSS avec un `#`

Les sélecteurs

Les classes sont utilisées lorsque l'on veut appliquer le même style à plusieurs éléments

Ils sont définis dans le HTML avec l'attribut `class`

Ils sont récupérés dans le CSS avec un `.`

Les formulaires

Un formulaire est constitué de différentes zones que va pouvoir renseigner l'utilisateur

Les formulaires

Un formulaire est constitué de différentes zones que va pouvoir renseigner l'utilisateur

- texte
- valeur
- choisir dans une liste
- cocher des éléments

Les formulaires

Un formulaire est constitué de différentes zones que va pouvoir renseigner l'utilisateur

- texte
- valeur
- choisir dans une liste
- cocher des éléments

Les données peuvent ensuite être envoyées à un serveur afin qu'il les traite et stocke dans des bases de données

Les formulaires

L'attribut principal de la balise `<form>` est `action`

Les formulaires

L'attribut principal de la balise `<form>` est `action`

Il définit l'action qui devra être exécutée lorsque l'utilisateur aura validé le formulaire

Les formulaires

L'attribut principal de la balise `<form>` est `action`

Il définit l'action qui devra être exécutée lorsque l'utilisateur aura validé le formulaire

Valeurs possibles :

- exécution d'un script sur le serveur (PHP ou CGI)
- envoi par mail (`mailto`)

Les formulaires

La validation de formulaire se fait grâce à un bouton `submit`
C'est un bouton qui a pour unique fonction d'envoyer le formulaire

Les formulaires

Démonstration de formulaire

Les formulaires

Lorsqu'un formulaire est soumis, les données de ce dernier sont envoyées

Il existe 2 méthodes d'envoi :

- POST
- GET

Les formulaires

Lorsqu'un formulaire est soumis, les données de ce dernier sont envoyées

Il existe 2 méthodes d'envois :

- POST
- GET

Vous **devez** utiliser la méthode POST

Les formulaires

GET écrit tous les noms des champs et toutes les valeurs saisies dans l'URL

Les formulaires

GET écrit tous les noms des champs et toutes les valeurs saisies dans l'URL

- problème de sécurité
- limitation de l'URL

Les formulaires

GET écrit tous les noms des champs et toutes les valeurs saisies dans l'URL

- problème de sécurité
- limitation de l'URL

La méthode POST n'affiche rien dans l'URL

Les formulaires

Quelques éléments de base d'un formulaire

Les flexbox

Le module des boîtes flexibles (flexbox), a été conçu comme

- un modèle de disposition unidimensionnel (une seule dimension à la fois : une ligne ou une colonne)
- et comme une méthode permettant de distribuer l'espace entre des objets d'une interface ainsi que de les aligner

Les flexbox

Lorsqu'on travaille avec les boîtes flexibles, deux axes interviennent :

- l'axe principal (*main axis*)
- l'axe secondaire (*cross axis*)

Les flexbox

Lorsqu'on travaille avec les boîtes flexibles, deux axes interviennent :

- l'axe principal (*main axis*)
- l'axe secondaire (*cross axis*)

L'axe principal est défini par la propriété `flex-direction` et l'axe secondaire est l'axe qui lui est perpendiculaire.

Tout ce que nous manipulons avec les boîtes flexibles fera référence à ces axes.

Les flexbox

L'axe principal est défini par la propriété `flex-direction` qui peut prendre quatre valeurs :

- `row`
- `row-reverse`
- `column`
- `column-reverse`

Les flexbox

La zone d'un document sujette au modèle de disposition flexbox est appelée un **conteneur flexible**.

Pour créer un conteneur flexible, il faut que la valeur de la propriété `display` de cet élément soit `flex` ou `inline-flex`. Dès que c'est le cas, les éléments « enfants » directs deviennent des éléments flexibles (*flex items*).

Les flexbox

Comme pour les autres propriétés CSS, certaines valeurs initiales sont définies, aussi, lorsqu'on crée un conteneur flexible, tous les éléments flexibles se comporteront de la façon suivante :

- les éléments s'afficheront en lignes horizontales (`flex-direction = row`)

Les flexbox

Comme pour les autres propriétés CSS, certaines valeurs initiales sont définies, aussi, lorsqu'on crée un conteneur flexible, tous les éléments flexibles se comporteront de la façon suivante :

- les éléments s'afficheront en lignes horizontales (`flex-direction = row`)
- les éléments seront placés à partir de la ligne de début de l'axe principal

Les flexbox

Comme pour les autres propriétés CSS, certaines valeurs initiales sont définies, aussi, lorsqu'on crée un conteneur flexible, tous les éléments flexibles se comporteront de la façon suivante :

- les éléments s'afficheront en lignes horizontales (`flex-direction = row`)
- les éléments seront placés à partir de la ligne de début de l'axe principal
- les éléments ne s'étireront pas le long de l'axe principal mais pourront se rétrécir si nécessaire

Les flexbox

Comme pour les autres propriétés CSS, certaines valeurs initiales sont définies, aussi, lorsqu'on crée un conteneur flexible, tous les éléments flexibles se comporteront de la façon suivante :

- les éléments s'afficheront en lignes horizontales (`flex-direction = row`)
- les éléments seront placés à partir de la ligne de début de l'axe principal
- les éléments ne s'étireront pas le long de l'axe principal mais pourront se rétrécir si nécessaire
- les éléments seront étirés le long de l'axe secondaire afin d'occuper l'espace sur cet axe

Les flexbox

Comme pour les autres propriétés CSS, certaines valeurs initiales sont définies, aussi, lorsqu'on crée un conteneur flexible, tous les éléments flexibles se comporteront de la façon suivante :

- les éléments s'afficheront en lignes horizontales (`flex-direction = row`)
- les éléments seront placés à partir de la ligne de début de l'axe principal
- les éléments ne s'étireront pas le long de l'axe principal mais pourront se rétrécir si nécessaire
- les éléments seront étirés le long de l'axe secondaire afin d'occuper l'espace sur cet axe
- la propriété `flex-basis` vaut `auto`

Les flexbox

Comme pour les autres propriétés CSS, certaines valeurs initiales sont définies, aussi, lorsqu'on crée un conteneur flexible, tous les éléments flexibles se comporteront de la façon suivante :

- les éléments s'afficheront en lignes horizontales (`flex-direction = row`)
- les éléments seront placés à partir de la ligne de début de l'axe principal
- les éléments ne s'étireront pas le long de l'axe principal mais pourront se rétrécir si nécessaire
- les éléments seront étirés le long de l'axe secondaire afin d'occuper l'espace sur cet axe
- la propriété `flex-basis` vaut `auto`
- la propriété `flex-wrap` vaut `nowrap`

Les flexbox

Une fonctionnalité majeure des boîtes flexibles est de permettre l'alignement et la justification des éléments le long des axes principal et secondaire tout en distribuant l'espace entre les éléments flexibles

Les flexbox

Une fonctionnalité majeure des boîtes flexibles est de permettre l'alignement et la justification des éléments le long des axes principal et secondaire tout en distribuant l'espace entre les éléments flexibles

La propriété `align-items` permet d'aligner les éléments le long de l'axe secondaire. Il existe 4 valeurs possibles :

- `stretch` (par défaut)
- `flex-start` ou `start`
- `flex-end` ou `end`
- `center`

Les flexbox

La propriété `justify-content` est utilisée afin d'aligner les éléments le long de l'axe principal dans la direction définie par `flex-direction`

- `start`
- `center`
- `space-between`
- `space-around`

CSS responsive

Ce qui est appelé *responsive design* est un design qui va s'adapter à la taille de l'écran et de la position

Le design va pouvoir s'adapter **si** le développeur crée les adaptations

CSS responsive

Ce qui est appelé *responsive design* est un design qui va s'adapter à la taille de l'écran et de la position

Le design va pouvoir s'adapter **si** le développeur crée les adaptations

Afin de détecter la résolution de l'écran, on utilise des **media queries**

CSS responsive

Media queries : module CSS permettant d'adapter le contenu d'une page web aux caractéristiques de l'appareil de l'utilisateur

CSS responsive

Media queries : module CSS permettant d'adapter le contenu d'une page web aux caractéristiques de l'appareil de l'utilisateur

Les media queries interrogent l'écran pour avoir des informations et savoir dans quelle résolution doit s'afficher la page HTML

CSS responsive

Media queries : module CSS permettant d'adapter le contenu d'une page web aux caractéristiques de l'appareil de l'utilisateur

Les media queries interrogent l'écran pour avoir des informations et savoir dans quelle résolution doit s'afficher la page HTML

« Si la zone d'affichage de la page a une dimension comprise entre 800 et 1200px, utiliser telle feuille de style. Si c'est une autre fourchette, utiliser d'autres styles »

CSS responsive

En plus de la résolution, les media queries peuvent détecter l'orientation du média

Cela se fait dans l'attribut `media` avec la propriété `orientation`

2 valeurs possibles :

- `landscape`
- `portrait`

JavaScript

JavaScript est un langage de programmation utilisé par les développeurs pour concevoir des sites web interactifs

JavaScript

JavaScript est un langage de programmation utilisé par les développeurs pour concevoir des sites web interactifs

Les fonctions JavaScript peuvent permettre d'améliorer l'expérience utilisateur d'un site web, de la mise à jour des flux de médias sociaux à l'affichage d'animations et de cartes interactives

JavaScript

JavaScript est un langage de programmation utilisé par les développeurs pour concevoir des sites web interactifs

Les fonctions JavaScript peuvent permettre d'améliorer l'expérience utilisateur d'un site web, de la mise à jour des flux de médias sociaux à l'affichage d'animations et de cartes interactives

En tant que **langage de script côté client**, c'est l'une des principales technologies du web

JavaScript

Le JavaScript permet de rendre les pages web localement dynamique (indépendamment du serveur) en manipulant le DOM et en créant des événements sur le DOM

JavaScript

Le JavaScript permet de rendre les pages web localement dynamique (indépendamment du serveur) en manipulant le DOM et en créant des événements sur le DOM

DOM : Document Object Model. Représentation (sous forme d'arbre) d'une page HTML manipulant le JavaScript

JavaScript

Lors du travail sur une page, il faut utiliser dans le code HTML une balise pour préciser que le code qui suit est du JavaScript et non plus du HTML

C'est le rôle de la balise `<script>`

JavaScript

Les éléments de bases du JavaScript :

- les variables
- les conditions
- les boucles
- les fonctions

JavaScript

Le JavaScript est très utilisé pour faire de la programmation événementielle

JavaScript

Le JavaScript est très utilisé pour faire de la programmation événementielle

La programmation événementielle est un paradigme de programmation dans lequel l'exécution d'actions est déclenchée automatiquement lorsqu'un événement survient

JavaScript

Afin de faire de la programmation événementielle en JavaScript, on utilise ce que l'on appelle des **écouteurs** (listeners en anglais)

JavaScript

Afin de faire de la programmation événementielle en JavaScript, on utilise ce que l'on appelle des **écouteurs** (listeners en anglais)

Ces écouteurs sont mis en place par l'instruction `Element.addEventListener()`

JavaScript

Démonstration