

HTML

LES BASES

LA GENÈSE

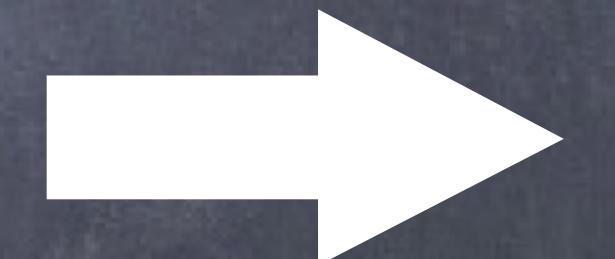
HTML et CSS sont des langages informatiques qui représentent la base des sites Internet ou logiciel full Web (ils servent essentiellement à mettre en forme du contenu et architecturer une page). Tous les sites ou logiciels full-web sont basés sur ces langages, ils sont incontournables et universels aujourd'hui. Ils sont la base même du Web.

Le langage HTML a été inventé par un certain Tim Berners-Lee en 1991...

INTERPRÉTATION

Quand vous regardez un site Internet il s'agit d'une interprétation de votre ordinateur (plus exactement du navigateur et du serveur selon les langages utilisés en plus du HTML et CSS), c'est d'ailleurs pour cela qu'il peut y avoir des petites différences visuelles entre les navigateurs que vous utilisez.

Le code interprété



Ce que me rend mon navigateur

The screenshot shows the homepage of JeuxVideo.com. The top navigation bar includes links for Applications, SYNTONY, JAVASCRIPT, JQUERY, MODELS, HTML, TWIG, CSS, POSTGRESOL, MYSQL, DOUR, CMS, JOOMLA, WORDPRESS, PRESTASHOP, VPS, GIT, SEO, GOOGLE, and RESEAUX SOCIAUX. The main menu features Jeux, Actu, TV, Vidéos, Forums, and Store. A search bar and a "Mon compte" link are also present. Below the menu, a banner highlights news about the Corsair K100 RGB keyboard, Steam's recent releases, the JeuxVideo.com Awards 2020, and Super Mario 3D World. The main content area displays a grid of news articles, including a special offer from Ubisoft's Lunar Sale, the awards ceremony, a documentary on playing video games in a cinema, and news about CD Projekt's Cyberpunk 2077 sources. A sidebar on the left shows a selection of video clips, and a TV section on the right displays a thumbnail for a news segment about CD Projekt's game development process.

Pour accéder à des outils spécifiques vous pouvez appuyer sur la touche F12 quand vous êtes sur votre navigateur

RÔLE

Chaque langage informatique à une utilisation spécifique ou permet de réaliser des actions d'une certaine manière

HTML (HyperText Markup Language) : il a fait son apparition dès 1991 lors du lancement du Web. Son rôle est de gérer et organiser le contenu. C'est donc en HTML que vous écrirez ce qui doit être affiché sur la page : du texte, des liens, des images... Vous direz par exemple : « Ceci est mon titre, ceci est mon menu, voici le texte principal de la page, voici une image à afficher, etc. » ;

CSS (Cascading Style Sheets, aussi appelées feuilles de style) : le rôle du CSS est de gérer l'apparence de la page web (agencement, positionnement, décoration, couleurs, taille du texte...). Ce langage est venu compléter le HTML en 1996.

Si vous voulez voir un «vieux» site Internet : <https://www.spacejam.com/>

RÔLE

Besoin d'un conseil ou d'un devis ? 07.67.69.28.20 | contact@pixel-online.fr



- Accueil
- Services
 - Site Internet & développement sur-mesure
 - Communication visuelle / Graphisme
 - Visibilité sur Internet / Référencement
 - Consulting / Audit technique / Bug
- Réalisations
- Blog
- Qui suis-je?
- Contact

Mobile Menu Toggle

- Accueil
- Services
 - Site Internet & développement sur-mesure
 - Communication visuelle / Graphisme
 - Visibilité sur Internet / Référencement
 - Consulting / Audit technique / Bug
- Réalisations
- Blog
- Qui suis-je?
- Contact

HTML brut



HTML +
CSS (il y
a aussi
du ja-
vascript
pour les
anima-
tions)

Le HTML définit le contenu.

Le CSS permet de styliser le contenu

Dans ce cours nous utiliserons HTML5 et CSS3 qui sont actuellement les versions utilisées sur le web

LES OUTILS POUR COMMENCER

Bien qu'il soit possible de coder dans le bloc note directement nous utiliserons vscode car il comprend beaucoup d'outils et sera utile pour la suite de votre formation.

<https://code.visualstudio.com/>

Choisissez votre plateforme et installez-le!

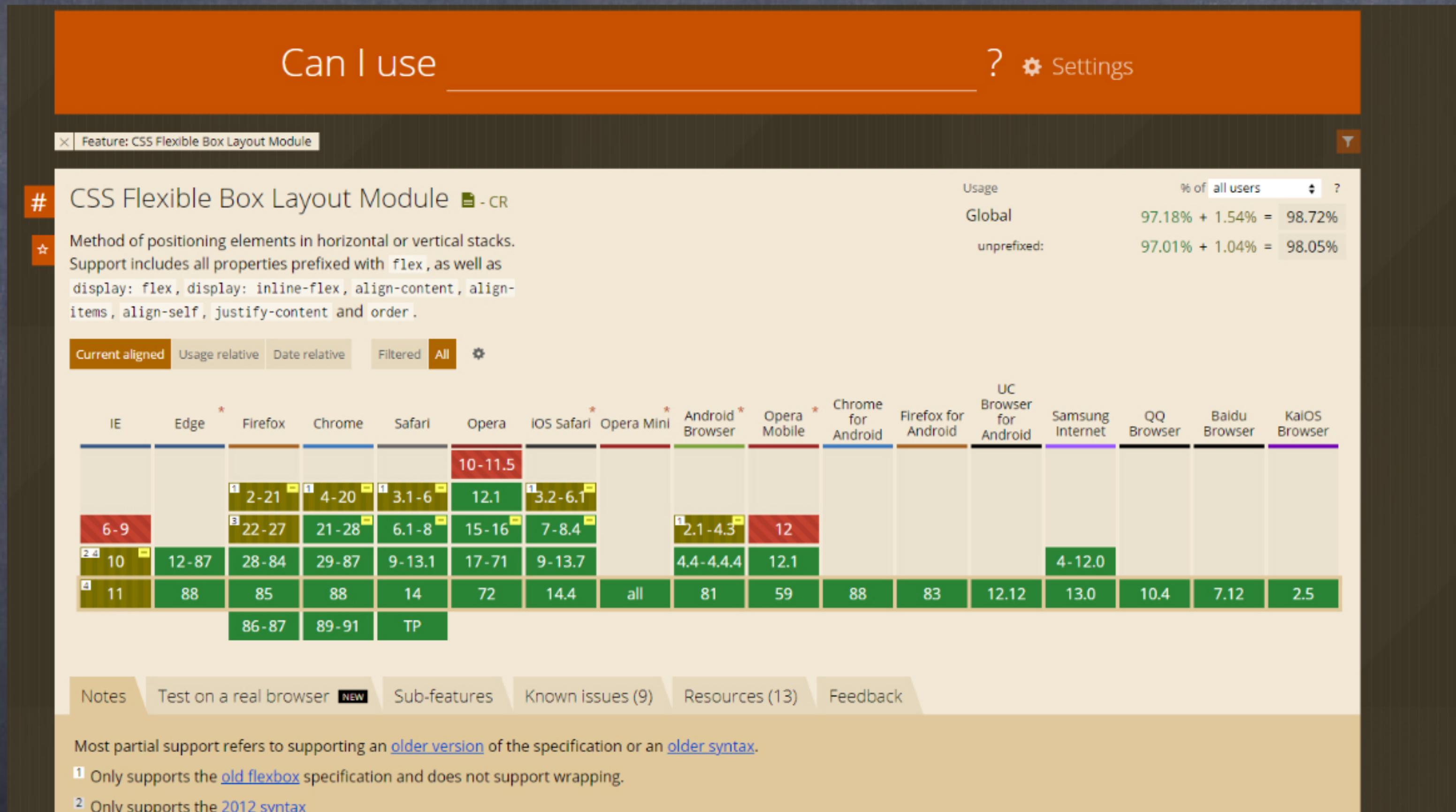
VSCode est IDE : Environnement de développement. Il s'agit d'un ensemble d'outils vous facilitant le développement informatique

Je vous invite également à installer les navigateurs les plus utilisés (Chrome, Firefox, Edge, Safari) en fonction de votre OS

Pourquoi? Car les navigateurs peuvent avoir des comportements différents en fonction d'un même code cela permet de tester son programme sur plusieurs supports.

LES OUTILS POUR COMMENCER

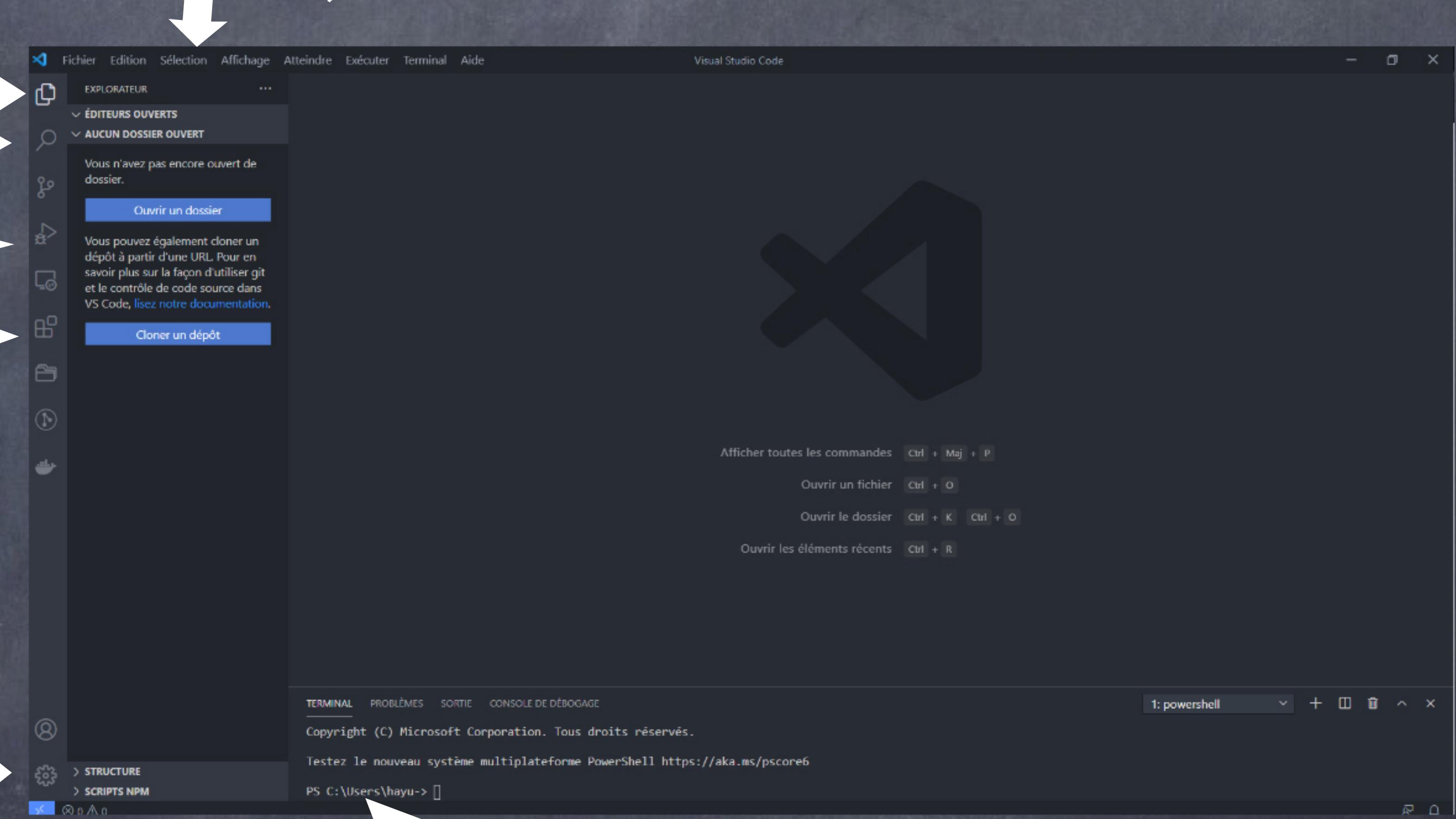
Pour vérifier si une fonctionnalité CSS/HTML/JS est disponible sur un navigateur vous pouvez utiliser :
<https://caniuse.com/>



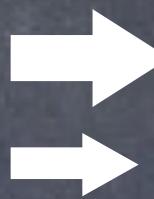
Vous aurez ainsi une présentation des supports et alternatives existantes
Il est important également de tester son code sur les navigateurs mobiles!!!

VSCODE

Menu complet



Gestion fichier en cours
Fonction de recherche



Debugger (utile pour php ou js par exemple)



Extension(module)



Paramétrage user



Terminal, très utile pour lancer des commandes

Vous avez forcément moins de menu à gauche.
C'est normal, ils apparaissent en fonction des extensions que vous installez!

INSTALLATION GIT

Git est un logiciel de versionning de code. Cela permet de sauvegarder son code, garder un historique de ses versions et facilite le travail entre développeurs.

Chacun travaille sur sa version de code.

Dans notre cas nous utiliserons GIT pour rechercher des éléments distants

Je vous invite à télécharger git ici
<https://git-scm.com/>

Pour vous aider dans l'installation
<https://astuces-informatique.com/comment-installer-utiliser-git-sous-windows/>

Vous retrouverez un guide d'utilisation de git en PDF dans ce cours (sur github)

VSCODE

Les plugins ou extensions vous aideront dans votre travail et vous faciliteront dans la réalisation de processus récurrent.

Les extensions de VScode utile pour le moment.

Apache conf

Auto Close Tag

Beautify

Bootstrap 4

color highlight

French Language

Gitlens

HTML boilerplate

HTML CSS Support

HTML Snippets

Intellisense for CSS

Live server

Live sass compiler

Project manager

LES BALISES ET ATTRIBUTS

L'HTML utilise des balises qui sont interprétées par le navigateur et en fonction de chaque le rendu à l'écran est différent.

Balises en paire

<h2>Titre secondaire</h2>

<h2> : « balise d'ouverture »

</h2> : « balise de fermeture ».

H2 permet d'afficher un titre 2 sur une page dans le même principe que Word.

Ici il n'y a pas d'attributs dans la balise, le texte contenu est un « noeud » de type texte

Exemple : <select>, <body>, <div>, , ...

Balise orpheline (auto fermantes)

Cette balise permet d'afficher l'image espagne3.jpg Si jamais la photo n'a pas été trouvée, elle sera remplacée par le texte « Photo d'espagne »

img est le nom de la balise

src et alt sont les attributs de cette balise

espagne3.jpg est la valeur de l'attribut src

Exemple :
,
, <input> ...

LES BALISES ET ATTRIBUTS

Pour connaitre les attributs disponibles sur les balises, il suffit de consulter la documentation.

Exemple balise img voir : https://www.w3schools.com/tags/tag_img.asp

Pensez à utiliser w3schools.com, celui-ci est le plus simple à comprendre pour le début.

Vous aurez accès à un éditeur de test.

Notez par ailleurs, que vous rencontrerez très vite des attributs «id» et «class» sur toute les balises existantes.

Ces attributs permettent d'apporter des customisations CSS ainsi que de jouer avec le DOM (utilisation de javascript).

STRUCTURE DE LA PAGE HTML

Schéma structurel

Vue en code

Explication



```
<!doctype html>
<html lang="fr">
  <head>
    <!-- En-tête de la page -->
    <meta charset="utf-8" />
    <title>Titre</title>
  </head>
  <body>
    <!-- Corps de la page -->
  </body>
</html>
```

balises	Fonctions
<!doctype html>	La toute première ligne s'appelle le doctype. Elle est indispensable. Le doctype indique qu'il s'agit bien d'une page web HTML.
<html lang="fr" /> et </html>	Les balises <html> et </html> englobent tout le contenu de la page WEB. L'attribut « lang » associé à sa valeur « fr » précisent la langue utilisée pour le contenu de la page.
<head> et </head>	Cette section donne quelques informations générales sur la page.
<!-- En-tête de la page -->	C'est un commentaire qui n'a aucun impact sur la page WEB . Les commentaires sont indispensables pour obtenir une bonne lisibilité du code.
<meta charset="utf-8" />	indique le codage des caractères (ici utf8 qui permet d'afficher sans aucun problème pratiquement tous les symboles de toutes les langues de notre planète), notamment les caractères spéciaux.
<title>Titre</title>	C'est le titre de la page web.
<body> et </body>	Cette section est le corps de la page C'est dans celle-ci que sera écrit la plus grande partie du code.

Ceci est un squelette standard d'une page HTML pour une bonne compréhension du navigateur. (voir 1-doctype.html)

PRÉSENTATION BALISE

(voir 2-balises-base.html)

Balise de titre :

<h1>Titre principal</h1> <h2>Titre secondaire</h2>
<h3>Titre trois</h3> <h4>Titre quatre</h4>
<h5>Titre cinq</h5> <h6>Titre six</h6>

Sans CSS le titre 1 est plus gros que le 2, cependant il faut utiliser les titres par ordre un seul H1 par page et ensuite utilisation de H2>H3>H4>Hn

Paragraphe :

Les balises <p> et </p> délimitent un paragraphe.

<p>Ceci est un paragraphe.</p>
<p>Ceci est un paragraphe.</p>
<p>Ceci est un paragraphe.</p>

Saut de ligne :

<p>Ceci est un paragraphe.

Salut Jean</p>

PRÉSENTATION BALISE

Les balises `` et `` (`em` : emphase)

Ces balises mettent en évidence le texte qu'elle encadre.

`texte mis en valeur`

Les balises `` et `` (`strong` : important, fort)

Pour mettre plus en valeur un texte que précédemment , on utilise les balises `` et ``.

`texte mis plus en valeur`

Les balises `<mark>` et `</mark>`

Les balises `<mark>` et `</mark>` permettent de faire ressortir visuellement une portion de texte.

`<mark>texte mis plus en valeur visuellement</mark>`

PRÉSENTATION BALISE

Les balises <div></div>

Permet de créer un «bloc» dans une page

<div>contenu</div>

Les balises

Sans spécifier du CSS ne font rien à l'écran... Mais reste très utile

BLOC ET INLINE

Certaines balises prennent la largeur entière de l'écran

Ce sont des balises de type «bloc»

`<p>,<h1>,<div>`, etc...

Certaines balises ne prennent que l'équivalent du contenu

Ce sont des balises de type «inline»

`,,`, etc...

BLOC



Titre principal

Titre secondaire

Titre trois

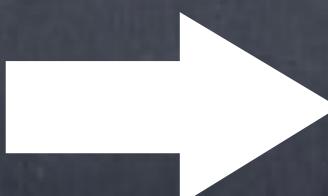
Titre quatre

Titre cinq

Titre six

ceci est un paragraphe

BLOC



Ceci
est
un
texte

INLINE

INLINE



Texte en em **Texte en strong** *Texte en mark*

Ceci est un/une div

Ceci est un/une div

ceci est un span ceci est un span

BLOC ET INLINE

Un élément de groupe %block peut contenir un (ou plusieurs) autres éléments %block et/ou %inline, sauf exceptions.

Un élément du groupe %inline ne peut contenir QUE un (ou plusieurs autres) éléments %inline.

Exemples d'imbrications :

```
<div><p>Paragraphe 1</p><p>Paragraphe 2</p></div>
```

La balise `<div>` (structure %block) englobe les deux paragraphes (structure %block). Ceci est autorisé.

Par contre, nous n'aurions pas pu écrire :

```
<span><p>Paragraphe 1</p><p>Paragraphe 2</p></span>
```

car la balise `` (dont la structure est %inline) n'est pas autorisée à contenir des éléments de structure %block comme les paragraphes `<p>`.

exceptions : <https://www.alsacreations.com/astuce/lire/55-balises-bloc-et-en-inline-les-exceptions.html>

LES LISTES ET TABLEAUX

```
<ul>  
  <li>element 1</li>  
  <li>element 2</li>  
  <li>element 3</li>  
  <li>element 4</li>  
  <li>element 5</li>  
  <li>element 6</li>  
  <li>element 7</li>  
</ul>  
  
<ol>  
  <li>element 1</li>  
  <li>element 2</li>  
  <li>element 3</li>  
  <li>element 4</li>  
  <li>element 5</li>  
  <li>element 6</li>  
  <li>element 7</li>  
</ol>
```

Les listes permettent d'ordonner des éléments et sont également utiles pour le référencement naturel

Utilisez des listes à la place de (tiret +
)

 est utilisé pour des listes sans numération, il est possible avec CSS de changer l'icône avant les éléments.

 est utilisé pour des listes ordonnées, par exemple pour une recette de cuisine!

LES LISTES ET TABLEAUX

```
<table border=1>
  <thead>
    <tr>
      <th>Entete 1</th>
      <th>Entete 2</th>
      <th>Entete 3</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Element 1</td>
      <td>Element 2</td>
      <td>Element 3</td>
    </tr>
    <tr>
      <td>Element 3</td>
      <td colspan="2">Element 4</td>
    </tr>
    <tr>
      <td rowspan="2">Element 5</td>
      <td>Element 6</td>
      <td>Element 7</td>
    </tr>
    <tr>
      <td>Element 8</td>
      <td>Element 9</td>
    </tr>
  </tbody>
</table>
```

Les tableaux sont moins utilisés aujourd'hui car ils ne s'adaptent pas bien au mobile (responsive design)

<table> : Défini un tableau

<thead> : Entête

<tr> : Ligne

<th> : cellule entête

<td> : cellule du tableau

<tbody> : contenu du tableau

<colspan> : permet de fusionner une cellule sur la ligne

<rowspan> : permet de fusionner une cellule sur la colonne

Avec du CSS on peut rendre un tableau plus « joli »

Entete 1	Entete 2	Entete 3
Element 1	Element 2	Element 3
Element 3	Element 4	
	Element 6	Element 7
Element 5	Element 8	Element 9

LES LIENS ET IMAGES

voir 4-liens-images.html

Les liens permettent de rediriger les internautes vers une autre page ou ouvrir un document.

Lien vers google
href=source
title=information du lien qui apparaît en infobulle (utile référencement)

Pour les images :

src=chemin
alt=information de l'image (utile référencement)

LES LIENS ET IMAGES

Chemin absolu :

Il s'agit du chemin complet :

<https://www.google.fr>

https : protocole sécurisé

www : sous domaine

google : domaine

fr : extension

Il existe aussi une variante que vous rencontrerez en production ou en utilisant des virtualHost (PHP)

/contact.html /info/produit-1.html

Le / devant revient à dire : <https://monsite.fr/>(page)

Le / renvoie à la racine du serveur (utile quand grosse imbrication de fichier et inclusion de contenu)

LES LIENS ET IMAGES

Chemin relatif :

Url à partir du dossier courant

Si image.jpg et index.html au même niveau alors le lien
src='image.jpg' ou src='./image.jpg'

Si le fichier est dans un dossier parent

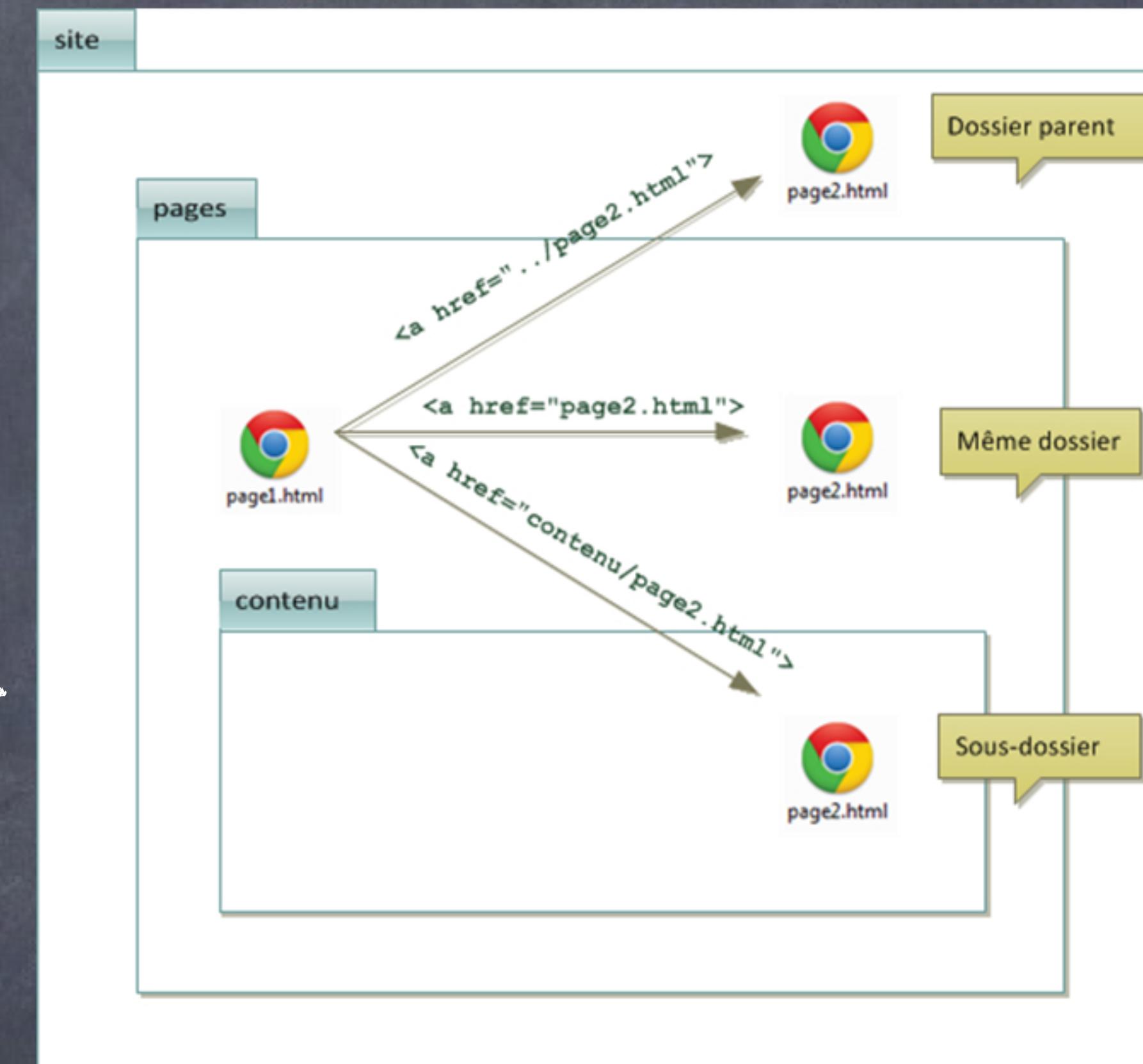
Si image.jpg est dans asset et index dans content
dans index.html pour faire référence à l'image on doit
sortir de content pour revenir un niveau au dessus et na-
vigner dans asset, le chemin sera

src='..../asset/image.jpg'

On peut aller autant de niveau au dessus en chainant le
. /

Exemple

../../../../image.jpg (3 niveaux au dessus)



LES FORMULAIRES

Les formulaires sont un peu complexe à mettre en place, il faut le construire de A à Z.

```
<form method='GET|POST' action='PAGE DE DESTINATION'>  
    <LES CHAMPS DU FORMULAIRE>  
    <input type='submit' name='validation_form_1'>  
    <input type='reset'>  
</form>
```

Le formulaire est compris entre les balises `<form>`

La balise form contient des attributs :

METHOD : GET OU POST (méthode de récupération des données)

- GET données en clairs passage URL

- POST données non visibles dans L'URL (Vous comprendrez l'intérêt pendant votre cours PHP ou NODE)

ACTION : Page de destination une fois le formulaire envoyé)

Dans tous les cas il y aura un bouton de soumission de formulaire (`type='submit'`) pour valider votre formulaire, voir un champs de `type='reset'` pour le remettre à 0

LES FORMULAIRES

Les formulaires sont des éléments très utiles pour récupérer des informations utilisateurs : Crédit de compte, newsletter, commande en ligne, blog,... C'est utilisé partout et c'est du HTML.

<input type='text' name='nom' required pattern='...' placeholder='texte'>
Input : Balise auto fermante

Type : text, password, number, mail, date, checkbox, radio selon votre besoin

Attention certains types ne sont pas encore bien reconnus par les navigateurs anciens (date, search, tel, -mail,... il faut vérifier)

Name : unique à la page, permet de récupérer le contenu ensuite pour le traiter avec un langage serveur (comme PHP par exemple ou NODE)

Required : Si champ obligatoire, si facultatif ne rien mettre

Pattern : Il s'agit d'une REGEX (expression régulière que vous verrez en détails par la suite)

Placeholder : texte par défaut utile que sur certains types : text, password, number, ...

LES FORMULAIRES

Il est possible de porter le focus :

```
<p>  
    <label for="age">Votre âge</label>  
    <input id="age" type="number" min="25" max="50" name="age"/>  
</p>
```

En utilisant la balise «label» et l'attribut for faisant référence à l'id du champ, si vous cliquez sur le label, le champ deviendra brillant.

Vous verrez également l'attribut «Value» sur certains éléments, cela permet de définir une valeur à l'élément (checkbox, radio, select, mais également sur des «input type» pour de la récupération de données)

Quand vous avez la possibilité d'un choix multiple checkbox ou select multiple dans notre cas il est impératif de rajouter [] au name cela transforme la variable de reception en tableau et sera utile pour récupérer toutes les infos.

Même si il existe des vérifications HTML de formulaire, il faudra toujours vérifier les données utilisateurs côté serveur! Car elles peuvent être altérées côté client (voir consol

TEMPLATE HTMLS

Vous avez vu les bases de l'HTML, voici un template conventionnel HTMLS

```
<!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">
  <meta name="description" content="Ceci est un template de base">
  <title>Le titre de ma page</title>
</head>
<body>
  <!--CONTENU HAUT DU SITE-->
  <header>
    <nav><!--MENU--></nav>
  </header>
  <!--SECTION PRINCIPALE DE LA PAGE-->
  <main>
    <!--CERTAINES BALISES DE STRUCTURE-->
    <article></article>
    <section></section>
    <div></div>
  </main>
  <!--BARRE LATERALE-->
  <aside>
  </aside>
  <!--CONTENU BAS DU SITE-->
  <footer></footer>
</body>
</html>
```

The screenshot shows a website template with the following structure:

- header**: Contains the title "header".
- NAVIGATION**: A navigation menu with links to "WHO WE ARE", "WHAT WE DO", and "CONTACT".
- Search query**: A search bar with a "Go!" button.
- heading**: A heading section containing placeholder text: "Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec a diam lectus. Sed sit amet ipsum mauris. Maecenas congue ligula ac quam viverra nec consectetur ante hendrerit. Donec et mollis dolor. Praesent et diam eget libero egestas mattis sit amet vitae augue. Nam tincidunt congue enim, ut porta lorem lacinia consectetur."
- subsection**: A subsection section containing placeholder text: "Donec ut libero sed arcu vehicula ultricies a non tortor. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean ut gravida lorem. Ut turpis felis, pulvinar a semper sed, adipiscing id dolor."
- another subsection**: Another subsection section containing placeholder text: "Pellentesque auctor nisi id magna consequat sagittis. Curabitur dapibus enim sit amet elit pharetra tincidunt feugiat nisl imperdiet. Ut convallis libero in urna ultrices accumsan. Donec sed odio eros."
- Footer**: Contains the copyright notice: "© Copyright 2050 by example.com. All rights reserved."
- Sponsors**: A sidebar on the right with the text "Be sure to check out our sponsors too!" and a list of five items: "• item", "• item", "• item", "• item", "• item".

Un peu de lecture très utile :

https://developer.mozilla.org/fr/docs/Apprendre/HTML/Comment/D%C3%A9couper_une_page_web_en_sections_logiques