

Introduction Html / Css

Le langage Html

- Le HTML est un langage de description de document utilisé sur Internet pour créer des pages Web. Son sigle signifie « *HyperText Markup Language* » en anglais, littéralement « langage de marquage hypertexte » (liaisons directes entre les éléments).
- Le balisage HTML est incorporé dans le texte du document et est interprété par un navigateur Web.
- Le langage HTML a été développé par le scientifique britannique Tim Berners-Lee en 1991. Plus tard, des versions améliorées de HTML ont commencé à apparaître.

Le langage Html

- Le HTML et le CSS prennent une place importante et particulière parmi l'ensemble des langages informatiques
- Le HTML et le CSS sont deux véritables standards en informatique.
- les langages HTML et CSS vont se trouver à la base de tout projet web car ils ont un rôle qui les rend incontournables : les navigateurs (Google Chrome, Safari, etc.) sont des programmes qui ont été construits pour pouvoir lire du code HTML au départ et qui ne peuvent aujourd'hui lire que du code HTML, CSS et JavaScript et nous allons donc nous appuyer sur ces langages (et sur le HTML en particulier) pour pouvoir afficher nos pages.
- En bref : quel que soit notre projet web (blog, site e-commerce, application mobile, etc.), l'utilisation du HTML et du CSS est incontournable.
- Les dernières versions sont HTML5 et CSS3.

Le langage Html

- le HTML est un langage de structure : il permet d'indiquer au navigateur que tel contenu est un titre, que tel autre est un simple texte, que cet objet est une image, que celui-ci est une liste, etc... :
- Le navigateur, qui « comprend » le HTML, va se baser sur ces indications pour afficher les contenus.
- Le CSS est un langage de styles : il permet de modifier l'apparence ou le rendu visuel de nos contenus HTML et donc de nos pages web.

Historique des versions

Version	Année
Html	1991
Html+	1993
Html 2.0	1995
Html 3.2	1997
Html 4.01	1999
XHTML 1.0	2000
Html 5	2012
XHtml 5	2013

Historique du langage internet

Version	Année
Concept HyperText	1960
Code personnalisé	1990
Naissance du Php	1990
Naissance du Gif et du Flash	1995
Naissance du Css	1996
Retour du Javascript	2000
Responsive Design	2008
Html 5 Css3 : création de sites plus dynamiques	2010

Ensemble d'outils destinés au développement de pages web avec des interfaces riches.



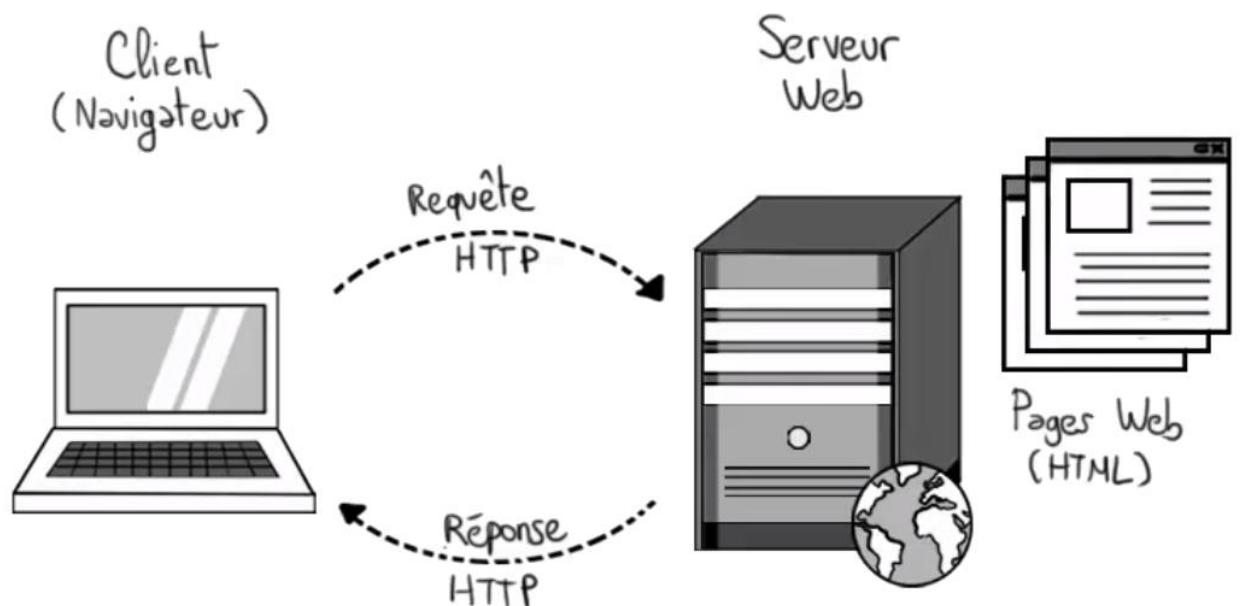
Le Html 5

- Beaucoup plus **orienté applicatif** que les versions précédentes
- Permet d'utiliser au maximum les **technologies web natives** pour construire une **page riche**
 - visionner des vidéos sans plugin (Silverlight, Flash...)
 - ajouter des effets visuels aux textes, images, vidéos
 - dessiner en SVG (vectoriel) et faire de la 3D
 - s'adapter aux différents supports d'affichage
 - ...



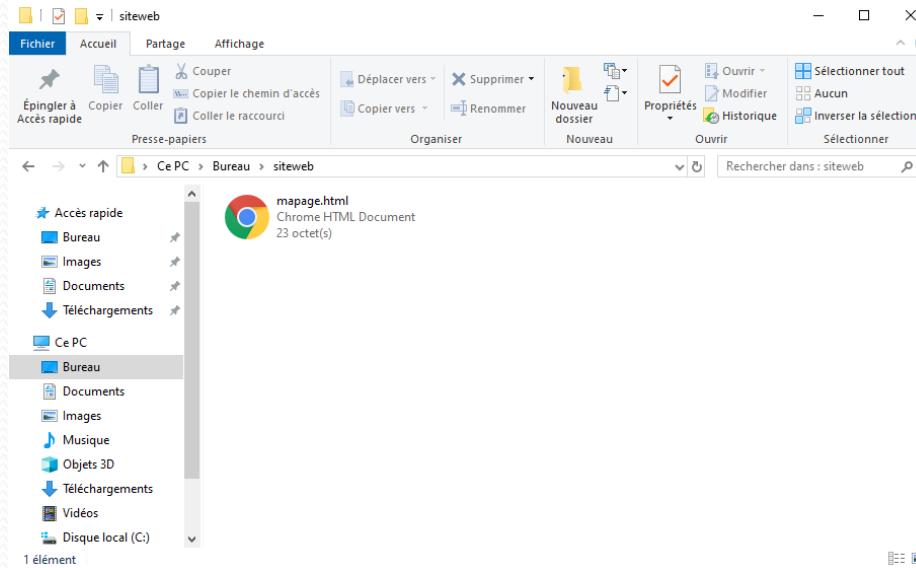
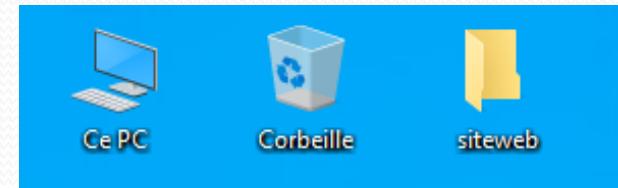
Comment ça marche ?

- On s'intéresse pour le moment aux **pages web statiques** :



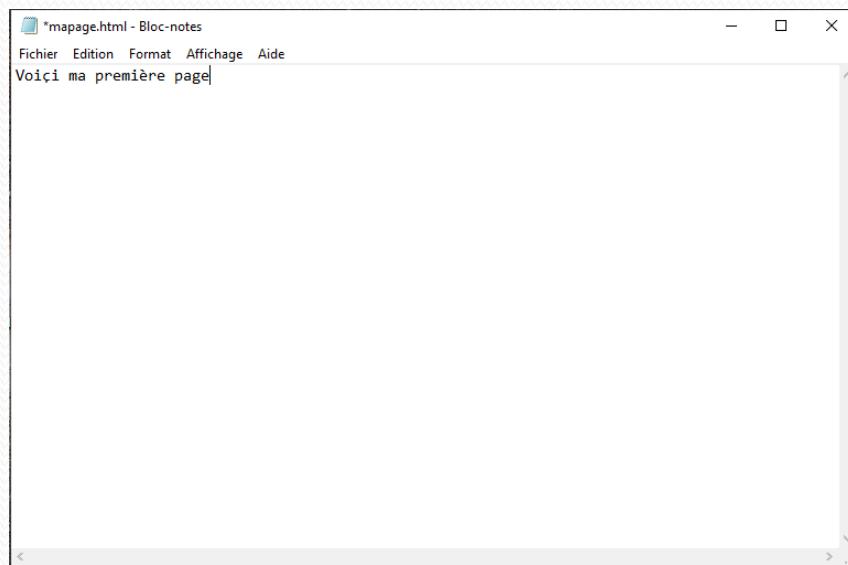
Le langage Html

- Nous avons simplement besoin d'un notepad,
- Créez un nouveau dossier « **siteweb** » sur le bureau,
- Avec un clic droit dans le dossier « Nouveau document texte »,
- Enregistrez le fichier à cet emplacement en le nommant « **mapage.html** »



Le langage Html

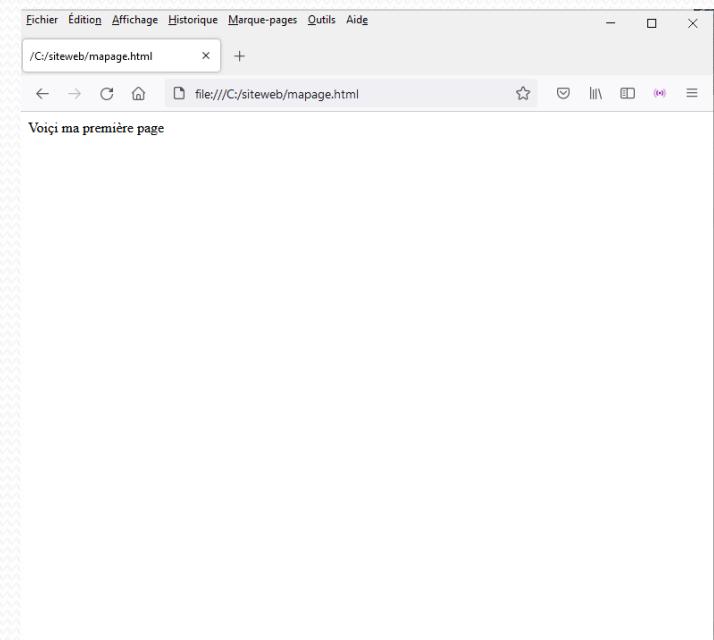
- Inscrivez une phrase dans le document comme ci-dessous,
- Ouvrez le fichier créé depuis un navigateur (Chrome, Edge, Firefox...),
- Le résultat dans le navigateur à droite



*mapage.html - Bloc-notes

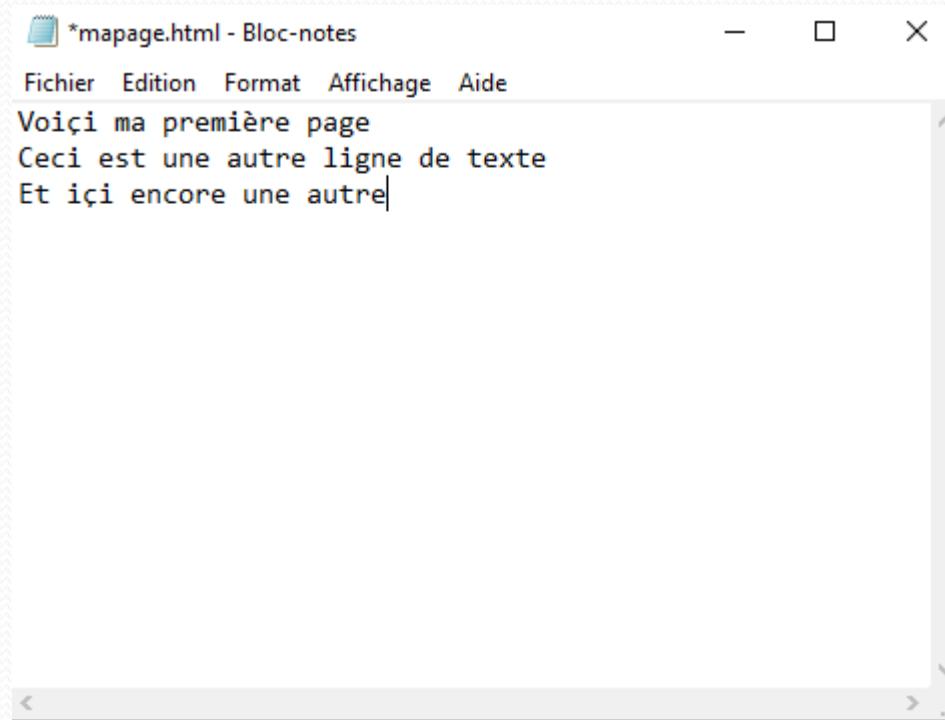
Fichier Édition Format Affichage Aide

Voiçi ma première page



Le langage Html

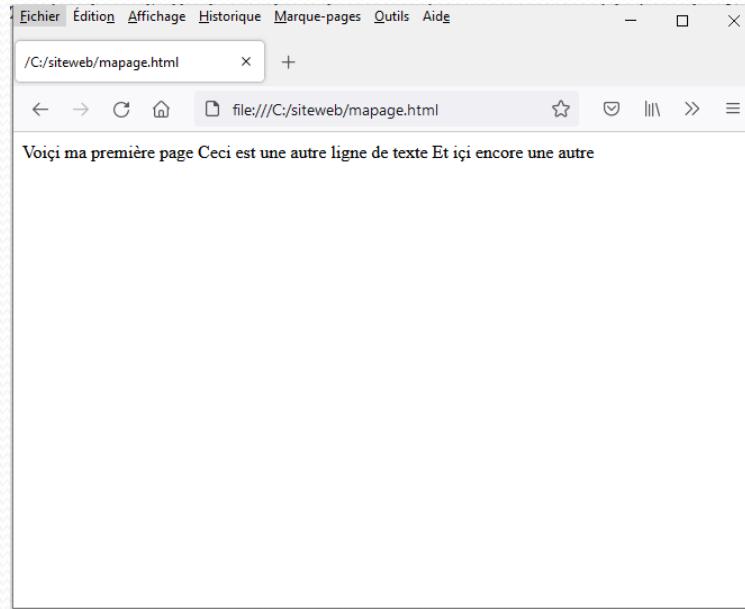
- Modifiez le fichier « **mapage.html** » comme ci-dessous :



```
*mapage.html - Bloc-notes
Fichier Edition Format Affichage Aide
Voiçi ma première page
Ceci est une autre ligne de texte
Et ici encore une autre|
```

Le langage Html

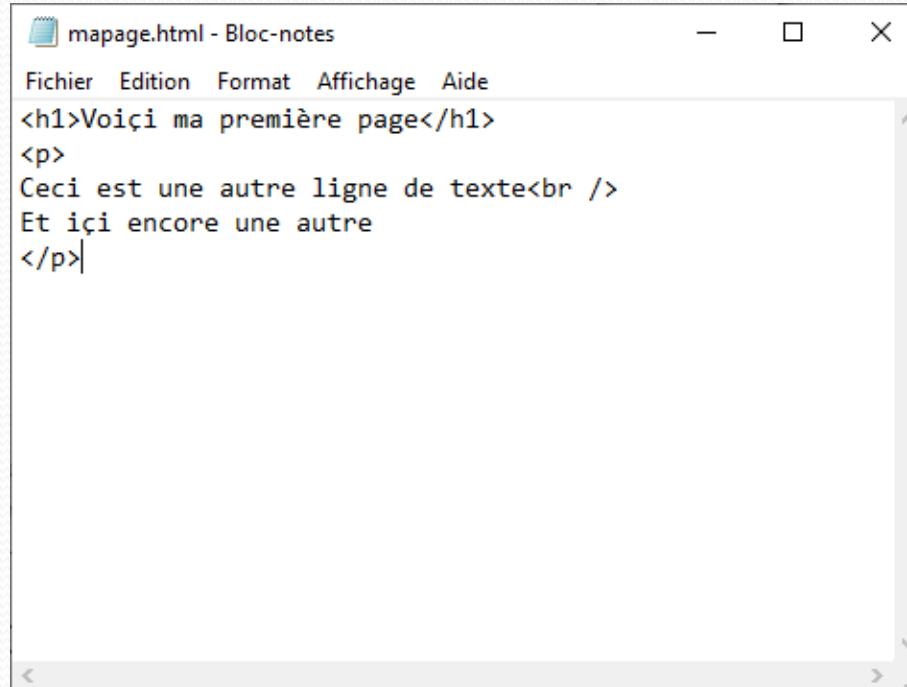
- Enregistrez, actualisez l'affichage dans le navigateur en pressant la touche F5 :



- On constate que le retour à la ligne présent dans le **fichier source** n'est pas conservé.

Le langage Html

- Pour **décrire** le document, il est donc nécessaire d'adopter un **langage** particulier : le **HTML**.
- Utilisez pour cela les **balises** de paragraphe **<p>...</p>**, de titre principal **<h1>...</h1>** et de retour à la ligne **
** :



The screenshot shows a Microsoft Notepad window with the title bar 'mapage.html - Bloc-notes'. The menu bar includes 'Fichier', 'Edition', 'Format', 'Affichage', and 'Aide'. The main content area contains the following HTML code:

```
<h1>Voiçi ma première page</h1>
<p>
Ceci est une autre ligne de texte<br />
Et ici encore une autre
</p>
```

Le langage Html

- Le langage HTML utilise des **balises** que le **navigateur** sait **interpréter**.
- **Les éléments peuvent se succéder** : il peut y avoir plusieurs paragraphes à la suite par exemple
- **Les éléments peuvent aussi s'imbriquer** pour conférer une hiérarchie au document il peut y avoir plusieurs paragraphes dans une section par exemple



STRUCTURE DU DOCUMENT

<!DOCTYPE>	Version du (X)HTML
<html>	Document HTML
<head>	Informations de la page
<body>	Contenu de la page

INFORMATIONS DE LA PAGE

<title>	Titre de la page
<meta />	Méta-données
<base />	Adresse de la base
<link />	Ressource externe
<style>	Définition de styles
<script>	Définition de scripts

BALISES DE STRUCTURE

<div>	Enveloppe en bloc
	Enveloppe en chaîne
<section>	Section
<nav>	Navigation
<header>	Header
<footer>	Footer
<article>	Article
<aside>	Contenu associé / secondaire

LIENS

	Lien vers un fichier
	Lien vers un mail
	Ancre
	Lien vers une ancre

MISE EN FORME DU TEXTE

<h[1-6]>	Niveaux de titres
<p>	Paragraphe
 	Saut de ligne
	Mise en gras (SEO)
	Mise en gras
	Mise en italique (SEO)
<i>	Mise en italique
<blockquote>	Citation externe
<mark>	Surlignage
<pre>	Texte pré-formaté
<code>	Code
<address>	Adresse
<cite>	Citation

TABLEAUX

<table>	Tableau
<caption>	Titre du tableau
<thead>	En-tête du tableau
<tfoot>	Pied du tableau
<tbody>	Corps du tableau
<th>	Cellule d'en-tête
<tr>	Ligne
<td>	Cellule

LISTES

	Liste
	Liste ordonnée
	Elément d'une liste
<dl>	Liste de définitions
<dt>	Terme
<dd>	Définition

FORMULAIRES

<form>	Zone d'un formulaire
<input />	Champ d'un formulaire
<textarea>	Champ texte multi-lignes
<button>	Bouton
<select>	Liste déroulante
<optgroup>	Groupe d'options
<option>	Option d'une liste
<label>	Titre d'un élément
<fieldset>	Groupe de champs
<legend>	Titre d'un fieldset
<datalist>	Liste de choix
<output>	Résultat d'un calcul

MÉDIAS

	Image
<map>	Zoning sur une image
<area>	Zone du zoning
<canvas>	Bloc graphique
<figure>	Conteneur de média
<figcaption>	Titre d'un conteneur
<audio>	Son
<source>	Ressources d'un média
<track>	Pistes d'un média
<video>	Vidéo
<object>	Object externe
<param>	Paramètre d'un objet

STRUCTURE D'UN DOCUMENT HTML 5

```

<html>
  <head>
    <title>Titre de la page</title>
    <meta charset="UTF-8" />
  </head>
  <body>
    <header>
      <nav></nav>
    </header>
    <section>
      <article></article>
      <article></article>
      <article></article>
      ...
    </section>
    <aside></aside>
    <footer></footer>
  </body>
</html>

```

HEADER

NAV

SECTION

ARTICLE

ARTICLE

ARTICLE

ASIDE

FOOTER

DOCUMENTATION

www.w3schools.com
Documentation de référence sur le HTML 5, le CSS 3, et d'autres langages de programmation ...

www.openclassrooms.com
Excellent tutoriels proposés gratuitement, avec beaucoup d'explications et d'exemples

www.alsacreations.com
Communauté dédiée à l'apprentissage des standards web (W3C, HTML, CSS, JS)

www.html5doctor.com
Tutoriels en anglais et documentation html

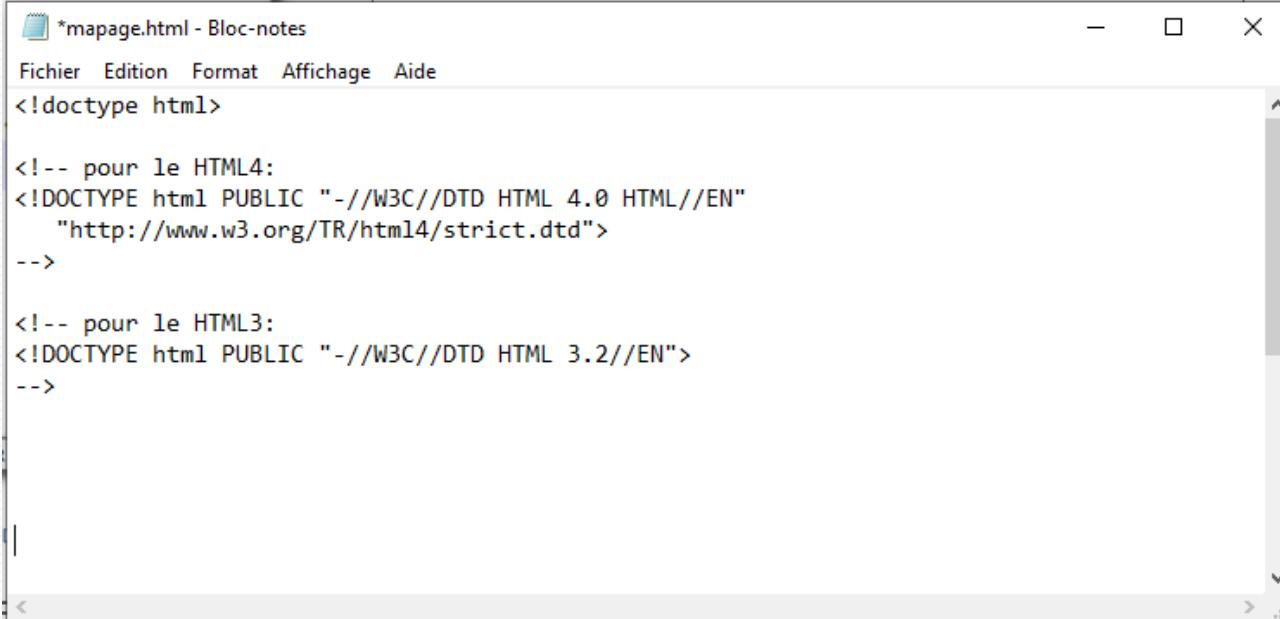
html5.validator.nu
Validation des pages HTML

RESSOURCES ET INSPIRATION

www.webcreme.com
www.cssmania.com
www.themeforest.net
www.wordpress.com
wwwprestashop.com
www.webdesign-inspiration.com
www.siteinspire.com
www.thebestdesigns.com
www.art-spire.com/category/webdesign/

Structure d'un document HTML5

- Déclaration du **type** de document et de la **version** de HTML utilisée, ici HTML5 :



The screenshot shows a Microsoft Notepad window titled "*mapage.html - Bloc-notes". The window contains the following code:

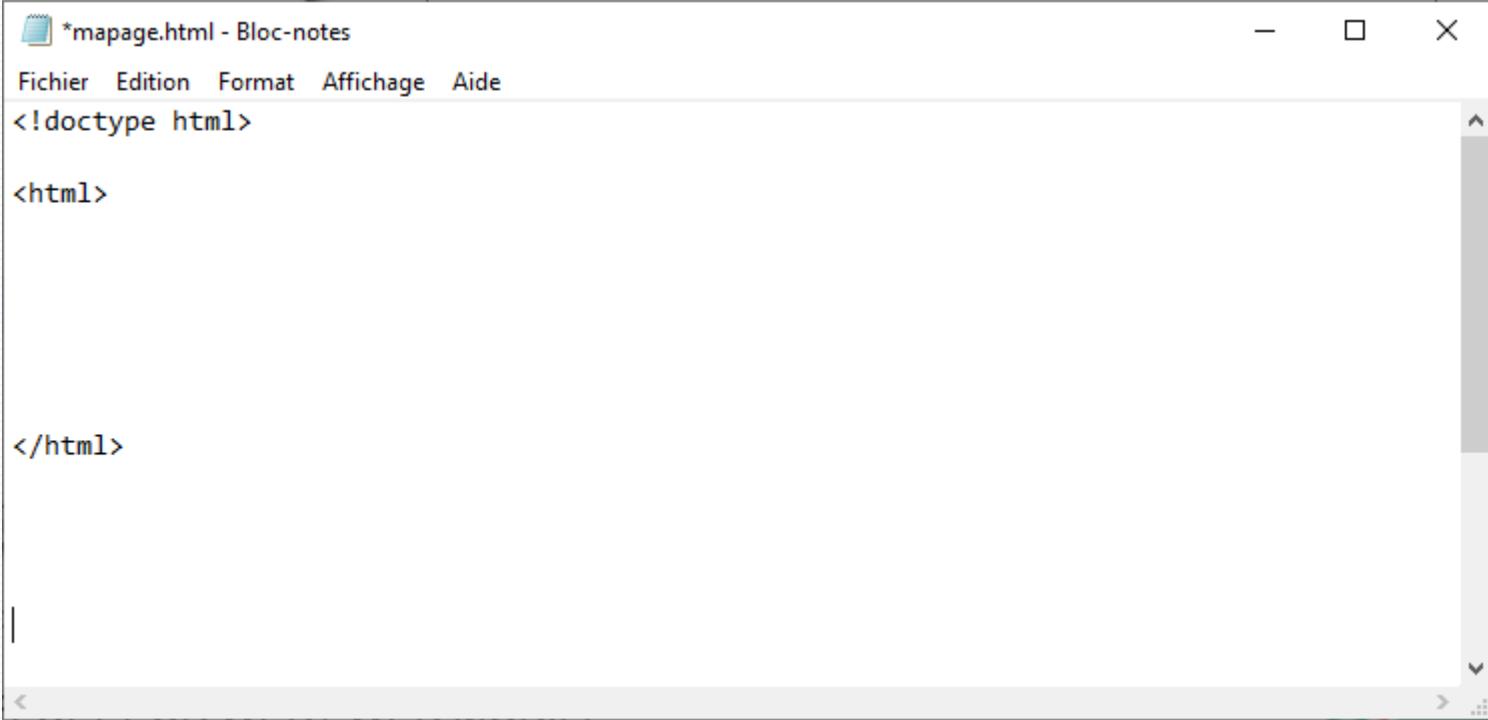
```
<!doctype html>

<!-- pour le HTML4:
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.0 HTML//EN"
    "http://www.w3.org/TR/html4/strict.dtd">
-->

<!-- pour le HTML3:
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 3.2//EN">
-->
```

Le langage Html

- On précise où commence et où se termine le contenu HTML avec les balises `<html>...</html>` :

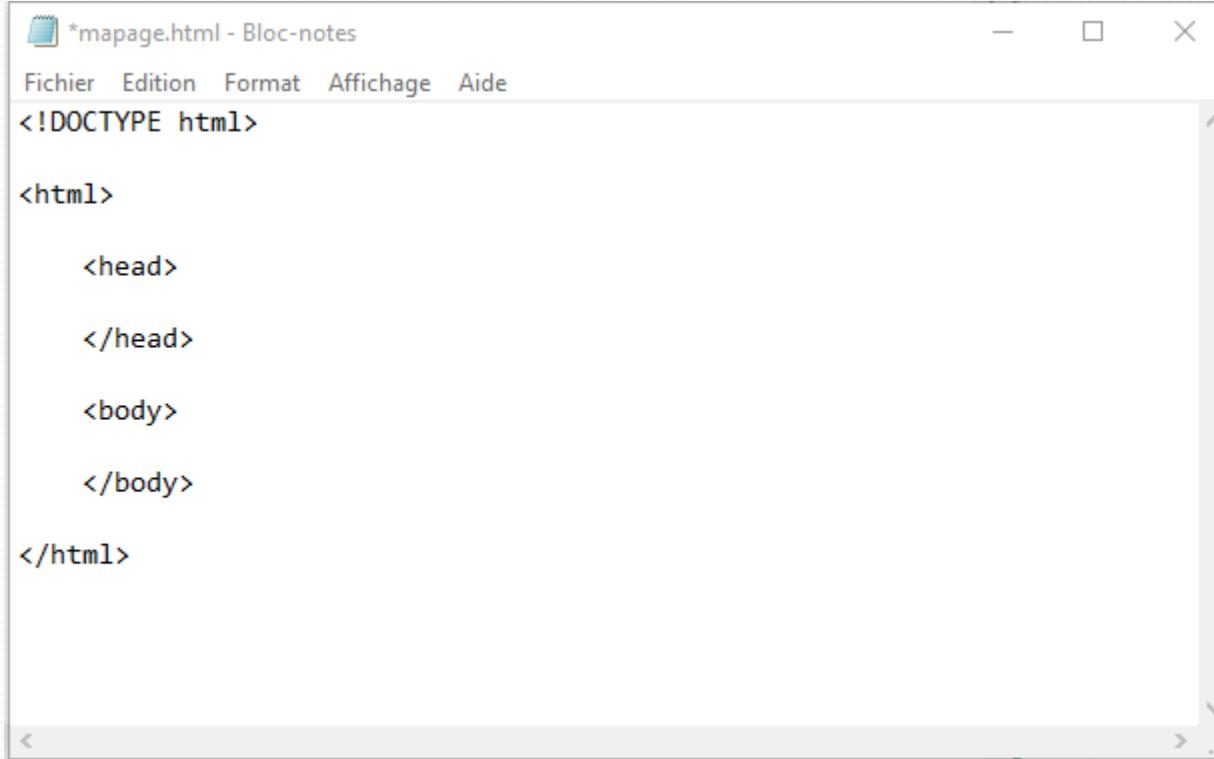


The screenshot shows a Microsoft Notepad window titled "mapage.html - Bloc-notes". The window contains the following code:

```
<!doctype html>
<html>
</html>
```

Le langage Html

- On définit les zones d'informations `<head>...</head>` et de contenu `<body>...</body>` :



The image shows a screenshot of a Microsoft Notepad window titled '*mapage.html - Bloc-notes'. The window contains the following HTML code:

```
<!DOCTYPE html>

<html>
  <head>
  </head>
  <body>
  </body>
</html>
```

Le langage Html

- L'attribut lang
- Permet de spécifier la langue de traitement du document global
- Indexation dans la bonne langue par les moteurs de recherche
- Synthèse vocale
- Vérification orthographique des formulaires

```
<html lang="fr">
```

Le langage Html

- La zone d'informations contient le **titre de la page**, le **codage des caractères...**
- La balise <title> sert au référencement (repris dans les résultats de recherche)

The image displays two windows side-by-side. On the left is a 'Bloc-notes' window titled '*mapage.html - Bloc-notes' containing the following HTML code:

```
<!doctype html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Ma page !</title>
  </head>
  <body>
  </body>
</html>
```

On the right is a web browser window titled 'file:///C:/siteweb/mapage.html'. The browser's title bar is highlighted with a yellow box, and an annotation with a blue arrow points to it, labeled 'Le texte de l'onget a été modifié' (The text of the tab has been modified). The browser interface includes standard navigation buttons (back, forward, search, etc.) and a status bar at the bottom.

Le langage Html

La balise <head>

Détermine l'entête du document et contient les méta-information

Les balises <meta>

- On y ajoutera également les appels pour les feuilles de style CSS et certains JavaScripts
- On peut y trouver des balises « meta » parmi lesquelles

<meta charset="UTF-8"/> : permet de définir l'encodage de caractère de la page (à mettre directement sous le <head>)

<meta name="description" content="description pour le référencement"/> affiché dans les résultats de recherche

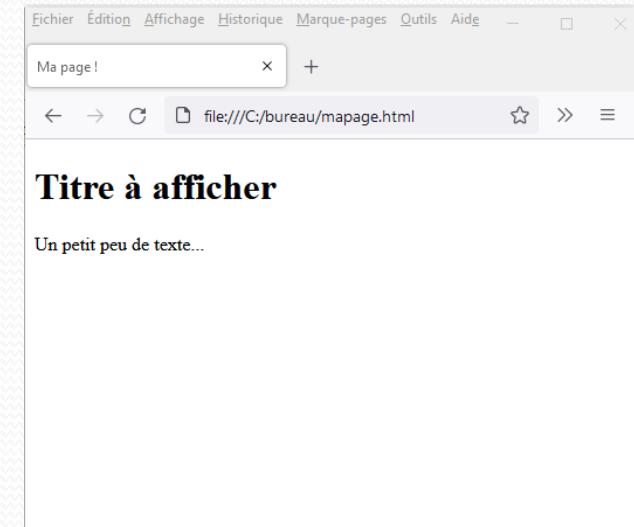
Le langage Html

- La partie visible de la page est à déclarer entre les balises `<body>...</body>` :

```
mapage.html - Bloc-notes
Fichier Edition Format Affichage Aide
<!DOCTYPE html>

<html>
    <head>
        <meta charset="UTF-8">
        <title>Ma page !</title>
    </head>

    <body>
        <h1>Titre à afficher</h1>
        <p>Un petit peu de texte...</p>
    </body>
</html>
```



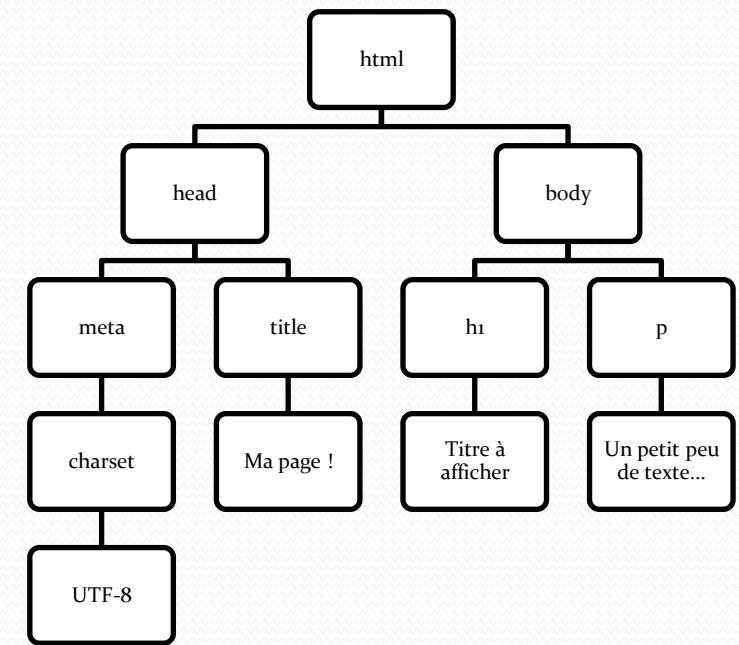
Le langage Html

- Un document HTML est structuré comme un arbre. **Cette structure sera très importante ensuite !**



A screenshot of a Microsoft Notepad window titled "mapage.html - Bloc-notes". The window shows the following HTML code:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Ma page !</title>
  </head>
  <body>
    <h1>Titre à afficher</h1>
    <p>Un petit peu de texte...</p>
  </body>
</html>
```



Le langage Html

- Pour **structurer** un document HTML5, tant sur le **positionnement** des éléments que la **hiérarchisation** des informations, il est important d'utiliser des balises de structure :

STRUCTURE D'UN DOCUMENT HTML 5

```
<html>
  <head>
    <title>Titre de la page</title>
    <meta charset="UTF-8" />
  </head>
  <body>
    <header>
      <nav></nav>
    </header>
    <section>
      <article></article>
      <article></article>
      <article></article>
      ...
    </section>
    <aside></aside>
    <footer></footer>
  </body>
</html>
```

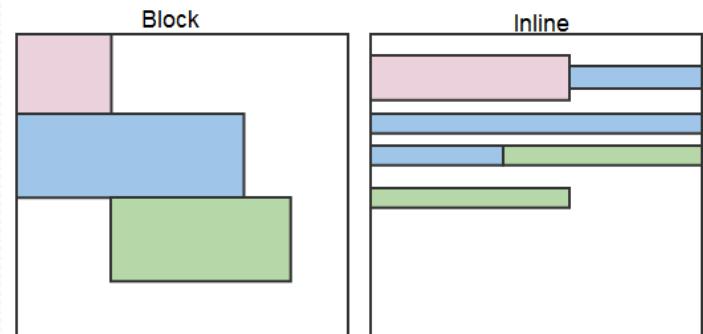
The diagram illustrates the structure of an HTML5 document. It consists of several nested sections:

- HEADER**: Contains a **NAV** element.
- SECTION**: Contains three **ARTICLE** elements.
- ASIDE**: Contains a single element with three dots (...).
- FOOTER**: Contains a single element with three dots (...).

Le langage Html

Une page web est en fait un empilement de blocs, de balises qui sont principalement de 2 types :

- « **inline** » (<a>, , ...)
 - Se trouve obligatoirement dans une balise de type « **block** »
 - Ne crée pas de retour à la ligne
- « **block** » (<p>, <h1>, <div>, <footer>...)
 - Crée obligatoirement un retour à la ligne
 - Par défaut, un bloc occupe 100% de la largeur disponible



Le langage Html

La balise « division » <div>

- <div> est un conteneur
- Cette balise n'a aucune valeur sémantique (aucun « sens »)
- Élément de type bloc qui peut inclure n'importe quel autre balise HTML
- Suivi d'un passage à la ligne
- Permet de regrouper d'autres blocs de HTML (paragraphes, tableaux, etc.) et de leur donner un style CSS commun

La balise paragraphe <p>

- Les paragraphes sont des éléments de type « bloc », suivis d'un passage à la ligne et d'une marge par défaut
- On peut imbriquer un paragraphe dans un div
- On ne doit PAS mettre un paragraphe dans un paragraphe

Le langage Html

Les titres h1 – h6

- Il existe six niveaux de titre <h1> <h2> <h3> <h4><h5> et <h6>
- (pas un de plus !)
- Un nouveau niveau provoque un passage à la ligne, les éléments de titre étant des éléments de bloc.
- La hiérarchisation visuelle est automatique (peut être modifiée via CSS)
- Il est possible d'avoir plusieurs titres de même niveau
- On ne peut pas mettre un Hn dans un p, ou un p dans un Hn

Le langage Html

- Le code HTML est interprété par le navigateur pour être lisible par l'internaute.
- Attention : ce code est **accessible à tous** : clic droit / Afficher le code source de la page :

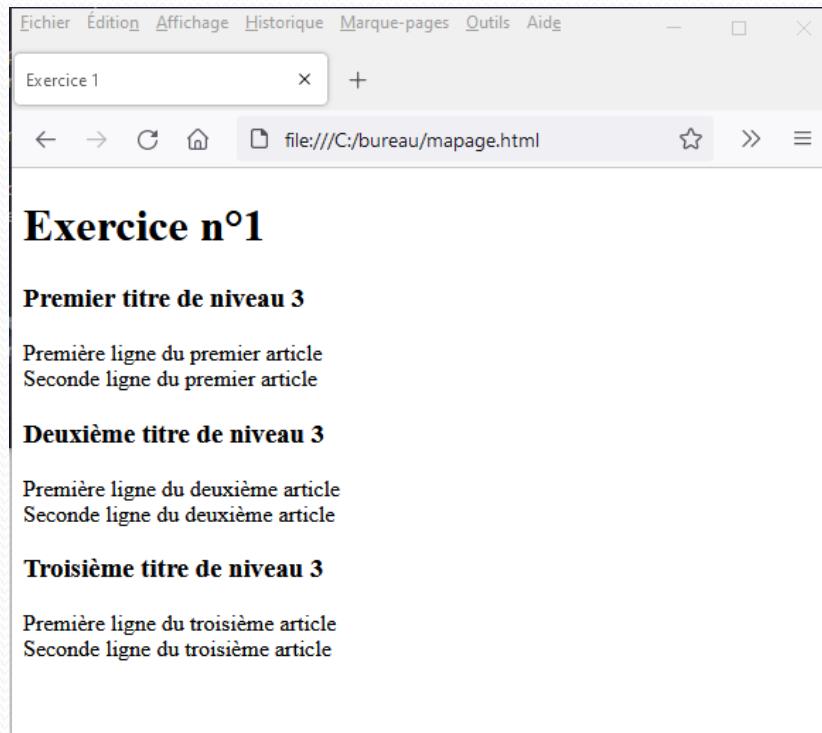


The screenshot shows a web browser window titled "Ma page !". The address bar shows "view-source:file:///C:/bureau/mapage.html". The page content displays the raw HTML code of the page:

```
1 <!DOCTYPE html>
2
3 <html>
4
5   <head>
6     <meta charset="UTF-8">
7     <title>Ma page !</title>
8   </head>
9
10  <body>
11    <h1>Titre à afficher</h1>
12    <p>Un petit peu de texte...</p>
13  </body>
14
15 </html>
16
17
```

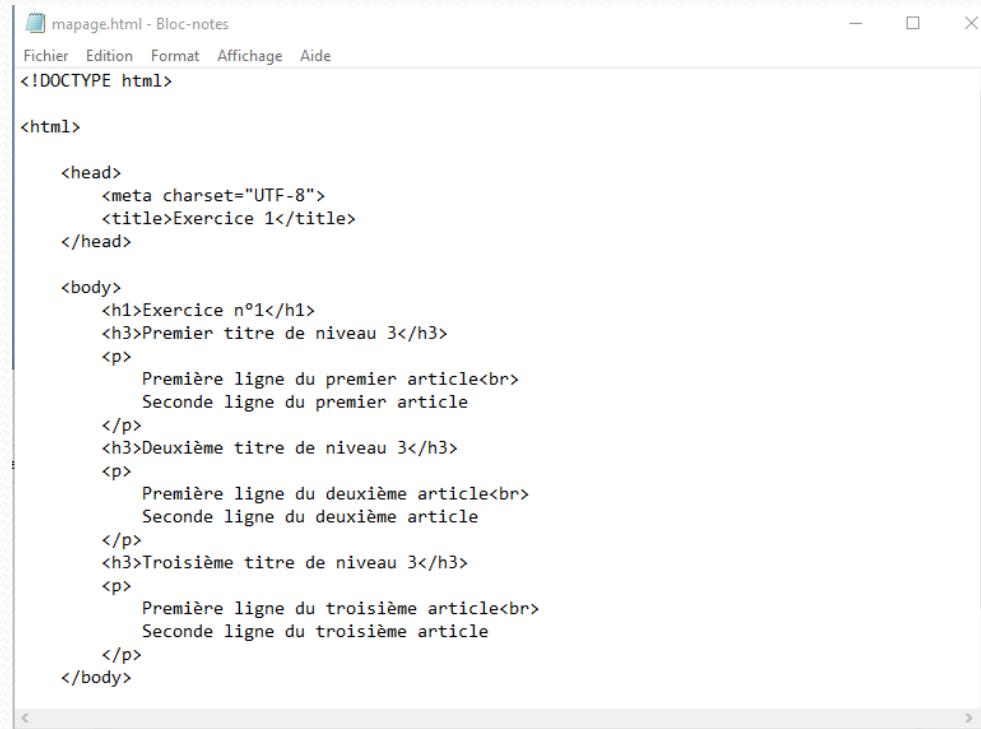
Le langage Html

- A l'aide des ressources à votre disposition, modifiez le fichier « **mapage.html** » pour obtenir le résultat suivant :



Le langage Html

- Solution proposée :



```
mapage.html - Bloc-notes
Fichier Edition Format Affichage Aide
<!DOCTYPE html>

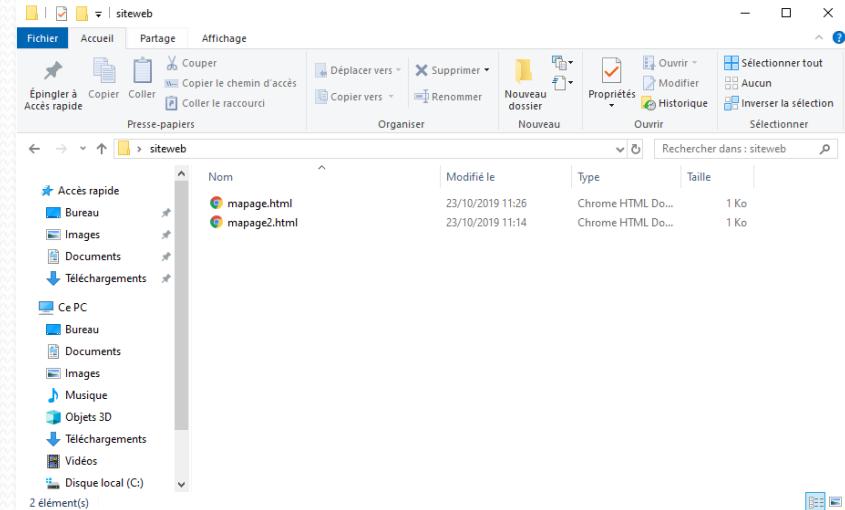
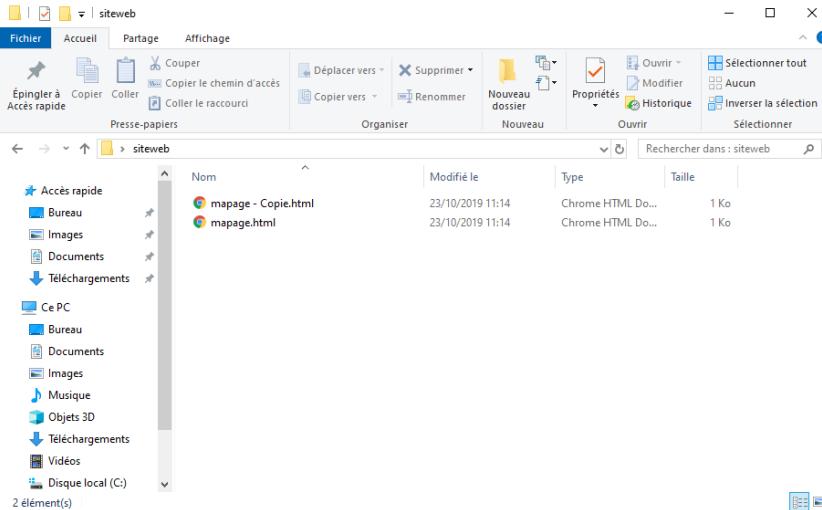
<html>
    <head>
        <meta charset="UTF-8">
        <title>Exercice 1</title>
    </head>

    <body>
        <h1>Exercice n°1</h1>
        <h3>Premier titre de niveau 3</h3>
        <p>
            Première ligne du premier article<br>
            Seconde ligne du premier article
        </p>
        <h3>Deuxième titre de niveau 3</h3>
        <p>
            Première ligne du deuxième article<br>
            Seconde ligne du deuxième article
        </p>
        <h3>Troisième titre de niveau 3</h3>
        <p>
            Première ligne du troisième article<br>
            Seconde ligne du troisième article
        </p>
    </body>

```

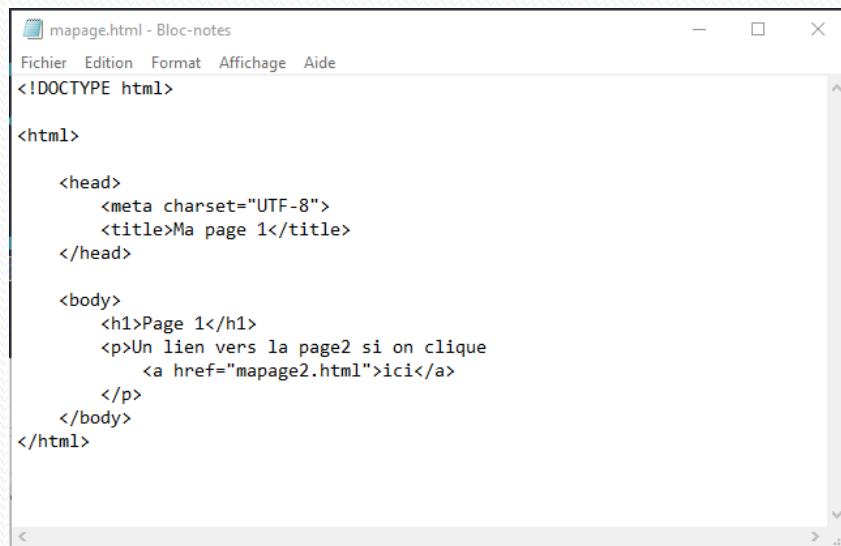
Le langage Html

- Dans le dossier de travail « **siteweb** », faites une copie de la page HTML existante.
- Renommez ensuite la copie obtenue en « **mapage2.html** ».



Le langage Html

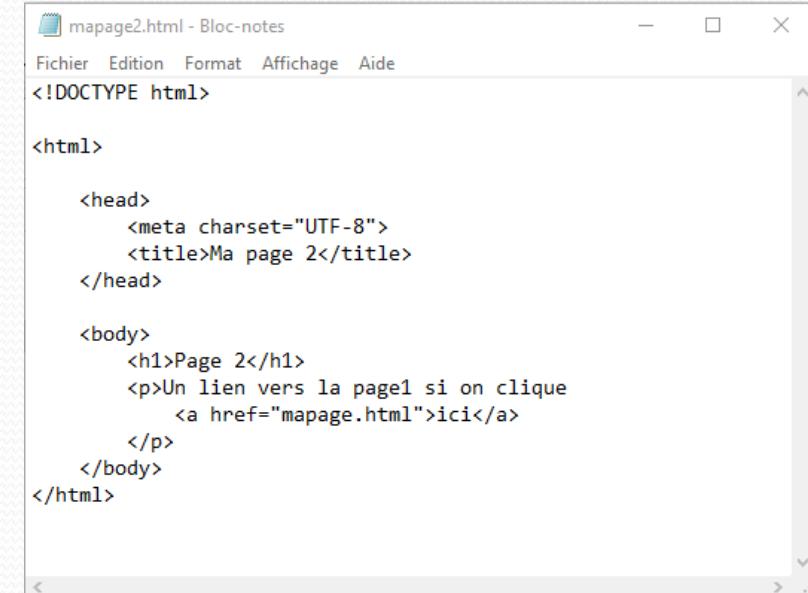
- Modifiez ces deux fichiers :



```
<!DOCTYPE html>

<html>
    <head>
        <meta charset="UTF-8">
        <title>Ma page 1</title>
    </head>

    <body>
        <h1>Page 1</h1>
        <p>Un lien vers la page2 si on clique
            <a href="mapage2.html">ici</a>
        </p>
    </body>
</html>
```



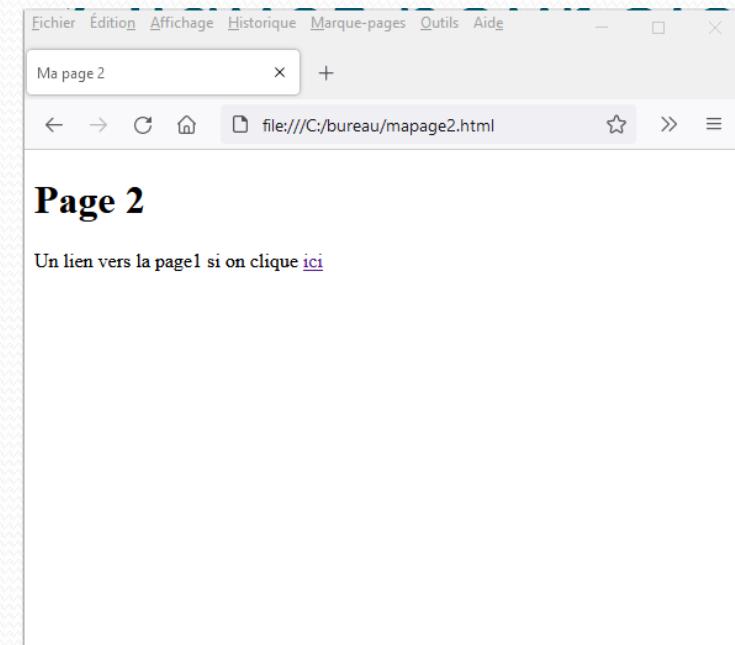
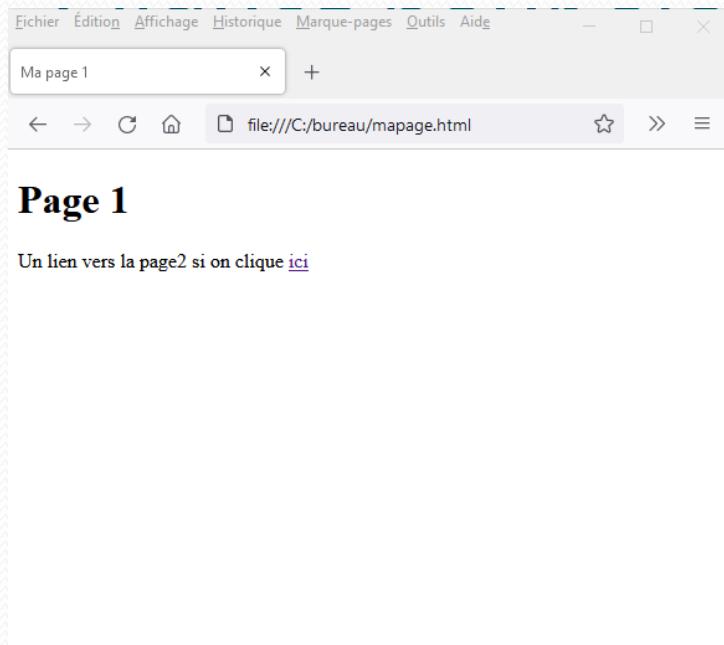
```
<!DOCTYPE html>

<html>
    <head>
        <meta charset="UTF-8">
        <title>Ma page 2</title>
    </head>

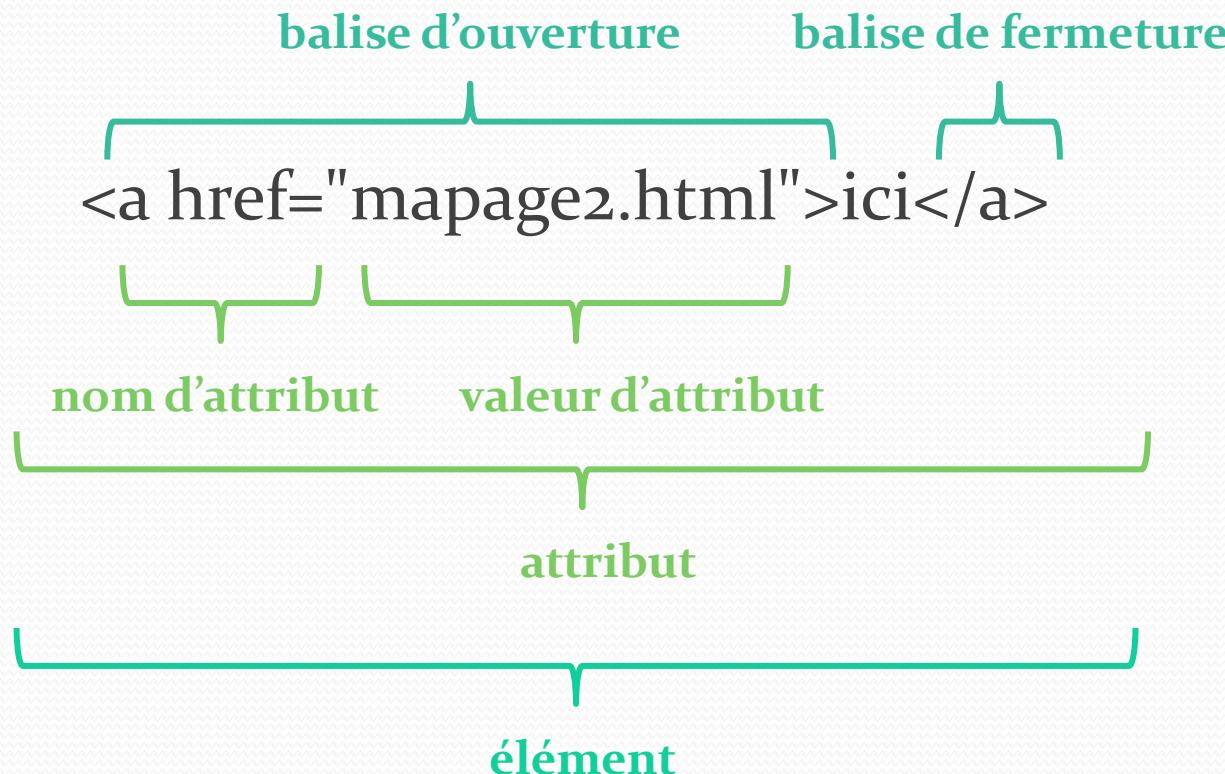
    <body>
        <h1>Page 2</h1>
        <p>Un lien vers la page1 si on clique
            <a href="mapage.html">ici</a>
        </p>
    </body>
</html>
```

Le langage Html

- Vous venez de réaliser un lien entre deux pages à l'aide de la balise `<a>...`

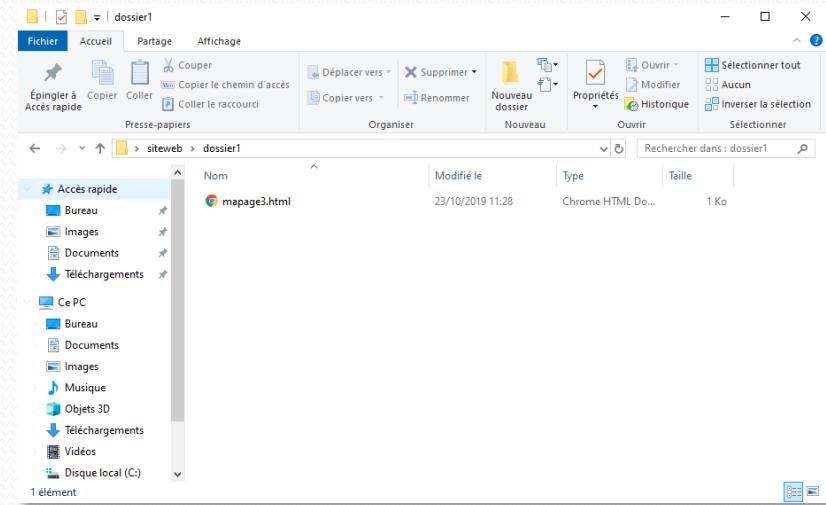
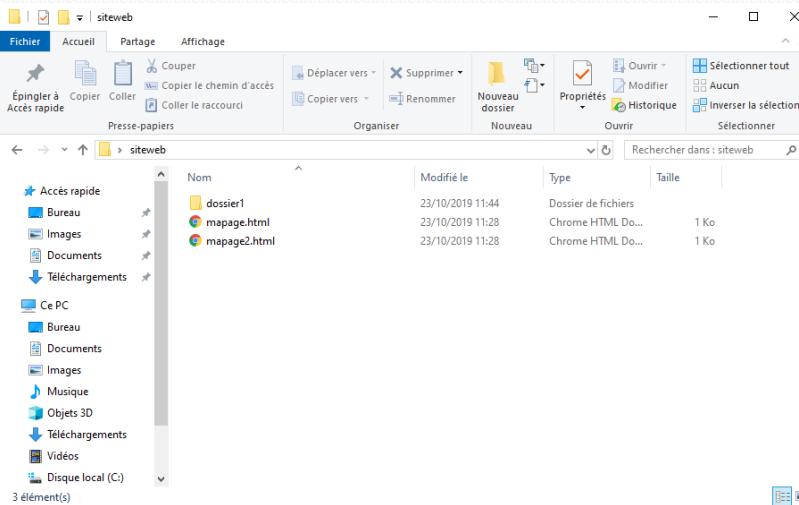


Le langage Html



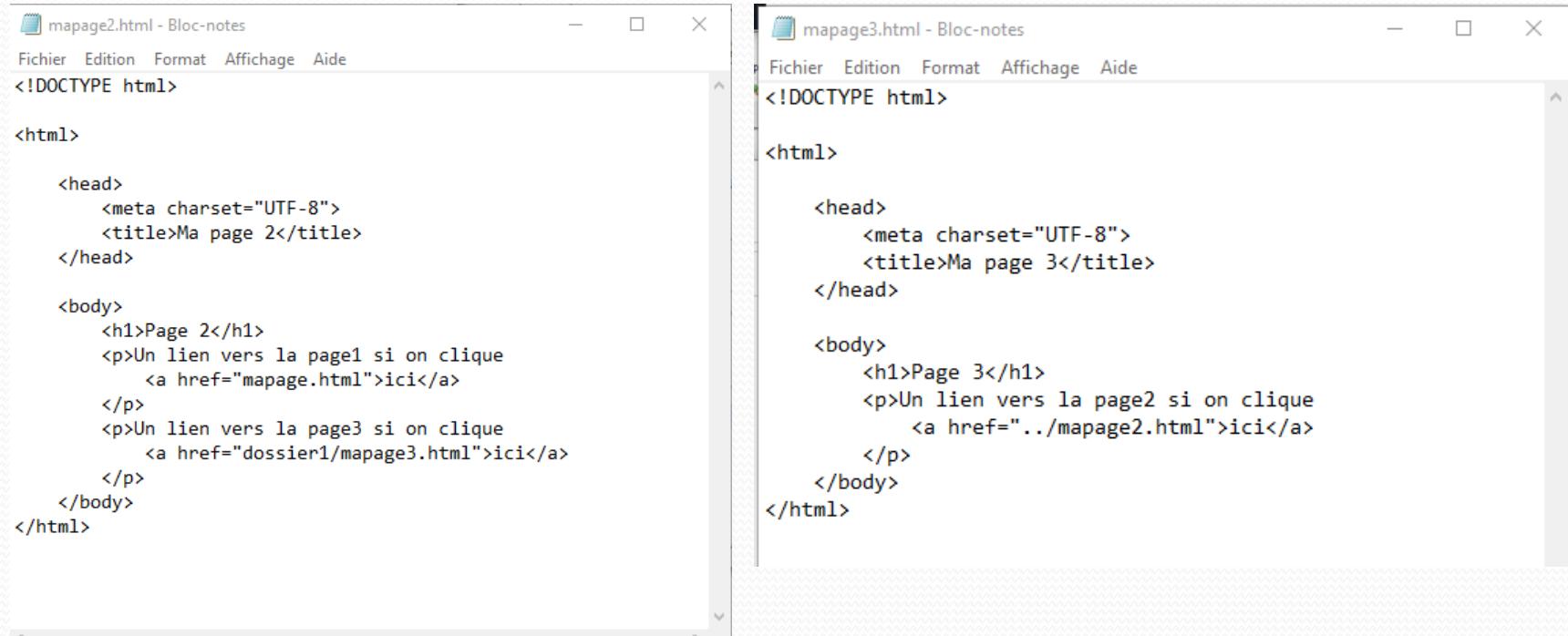
Le langage Html

- Dans le dossier de travail « **siteweb** », créez un dossier « **dossier1** ».
- Faites un copie du fichier « **mapage.html** » dans ce dossier et appelez le « **mapage3.html** ».



Le langage Html

- Modifiez les fichiers « mapage2.html » et « mapage3.html » :



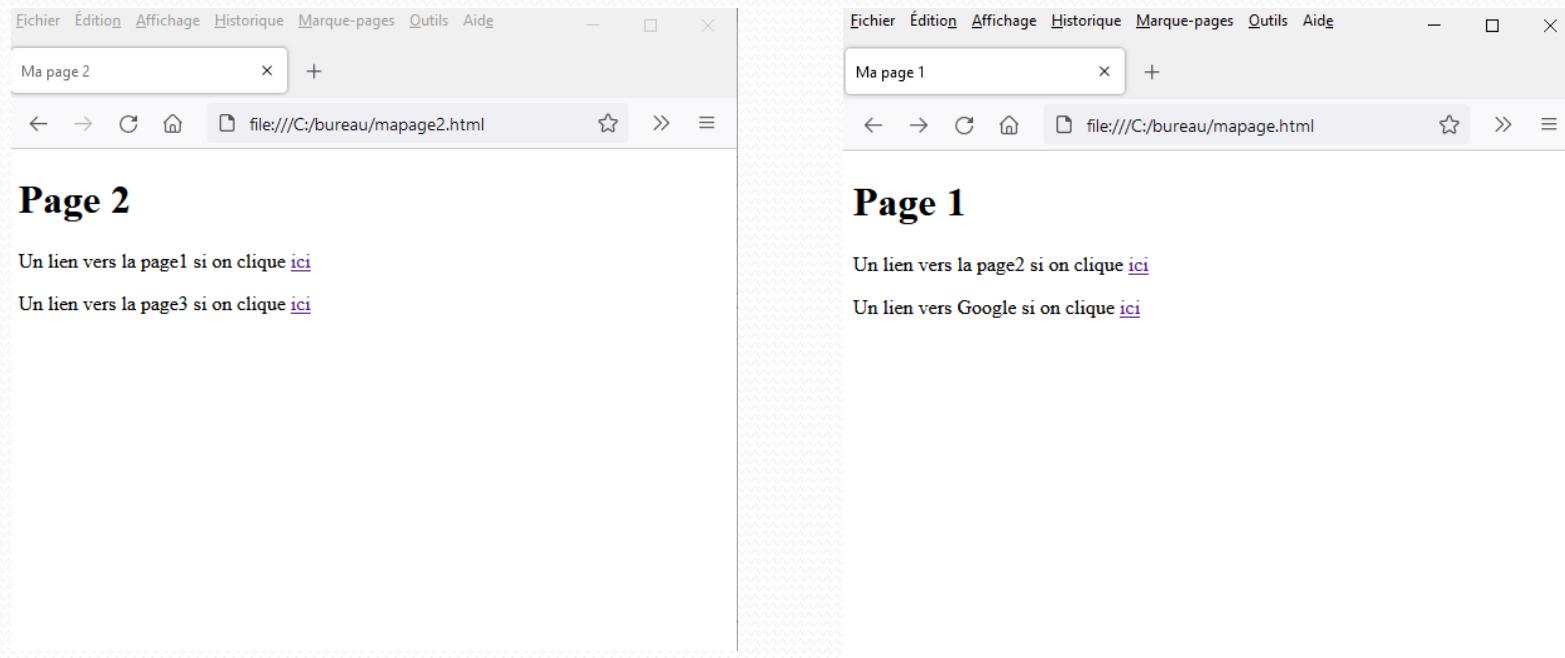
The image shows two windows of the Windows Notepad application side-by-side. Both windows have a title bar 'mapage2.html - Bloc-notes' and 'mapage3.html - Bloc-notes' respectively. The menu bar includes Fichier, Edition, Format, Affichage, and Aide. The content of each file is as follows:

```
mapage2.html content:
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Ma page 2</title>
  </head>
  <body>
    <h1>Page 2</h1>
    <p>Un lien vers la page1 si on clique
       <a href="mapage.html">ici</a>
    </p>
    <p>Un lien vers la page3 si on clique
       <a href="dossier1/mapage3.html">ici</a>
    </p>
  </body>
</html>
```

```
mapage3.html content:
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Ma page 3</title>
  </head>
  <body>
    <h1>Page 3</h1>
    <p>Un lien vers la page2 si on clique
       <a href="../../mapage2.html">ici</a>
    </p>
  </body>
</html>
```

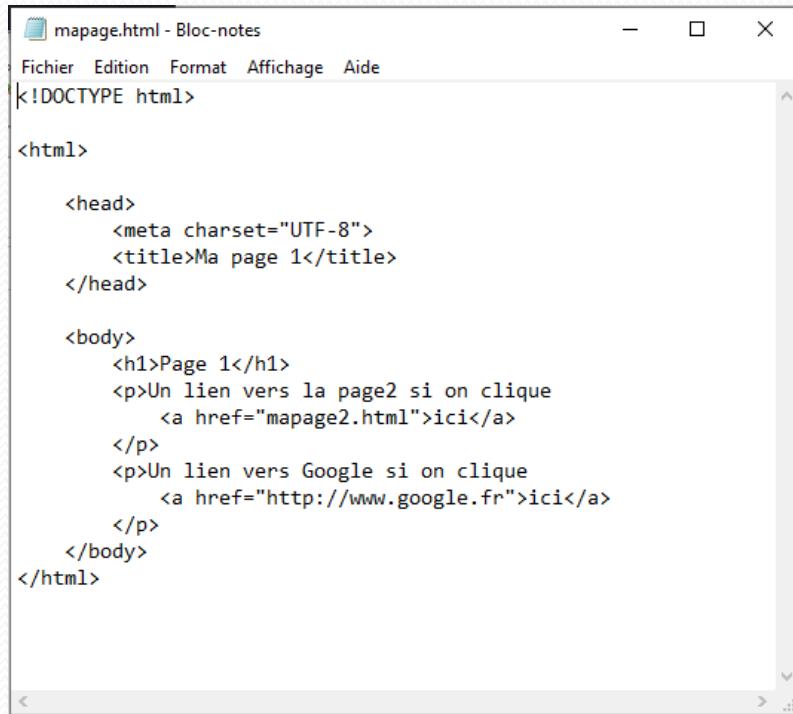
Le langage Html

- Vous venez de réaliser un lien entre deux pages situées à des endroits différents sur le même site en utilisant des **liens relatifs**.



Le langage Html

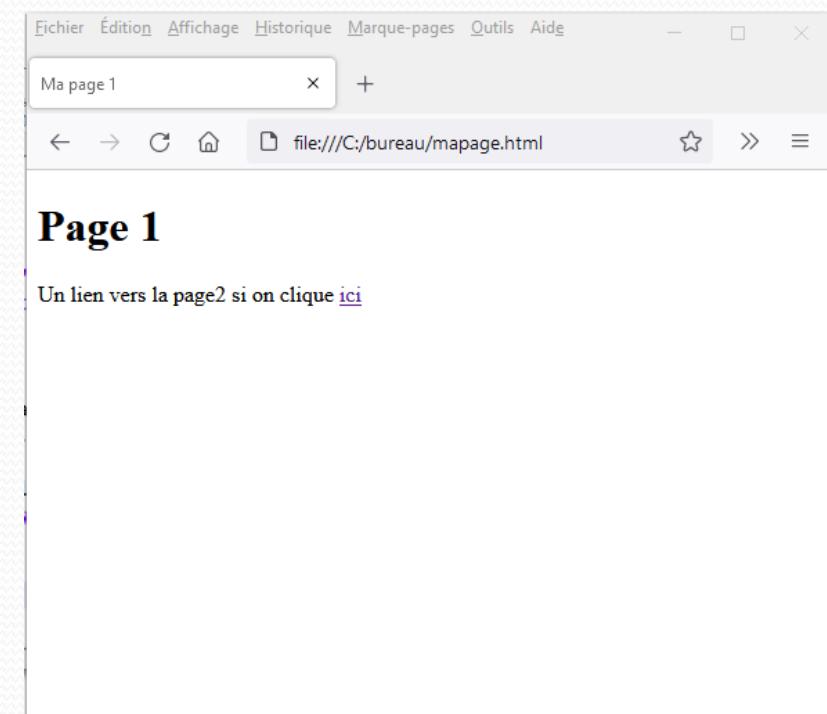
- Modifiez le fichier « **mapage.html** » :



```
mapage.html - Bloc-notes
Fichier Edition Format Affichage Aide
<!DOCTYPE html>

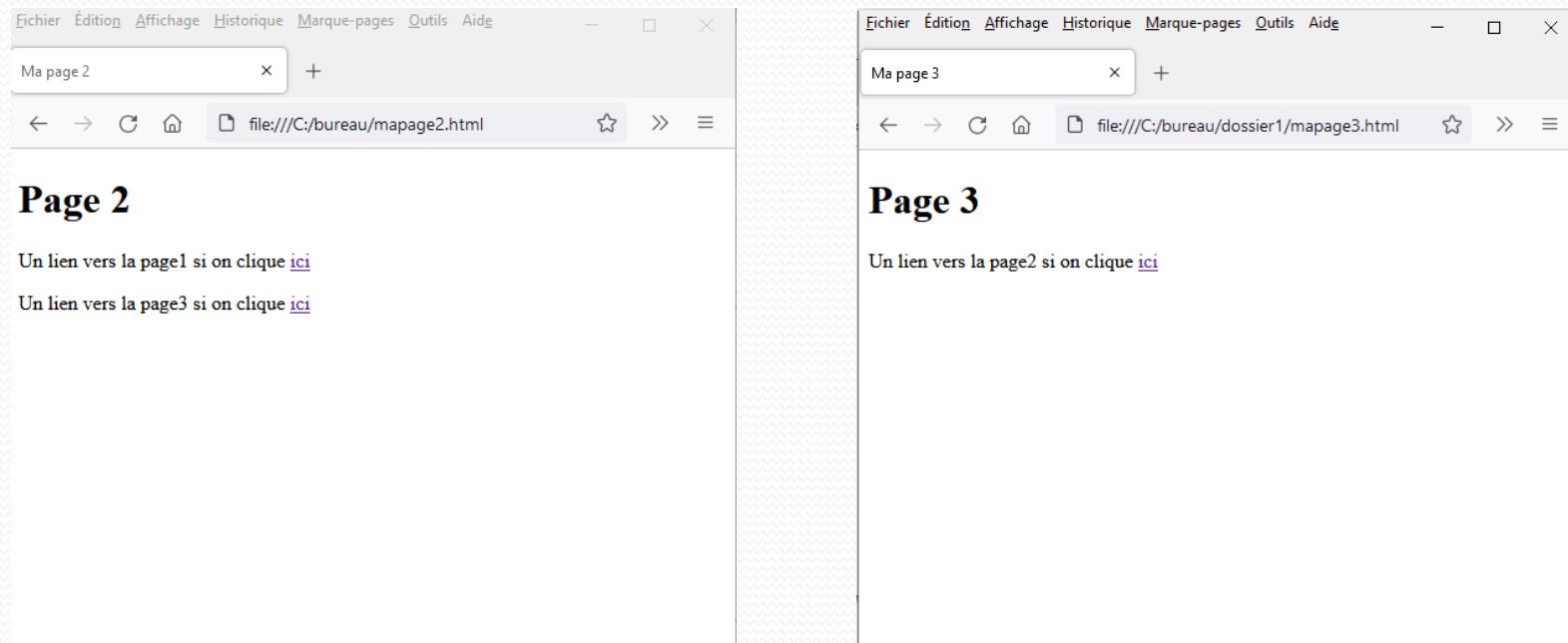
<html>
    <head>
        <meta charset="UTF-8">
        <title>Ma page 1</title>
    </head>

    <body>
        <h1>Page 1</h1>
        <p>Un lien vers la page2 si on clique
            <a href="mapage2.html">ici</a>
        </p>
        <p>Un lien vers Google si on clique
            <a href="http://www.google.fr">ici</a>
        </p>
    </body>
</html>
```



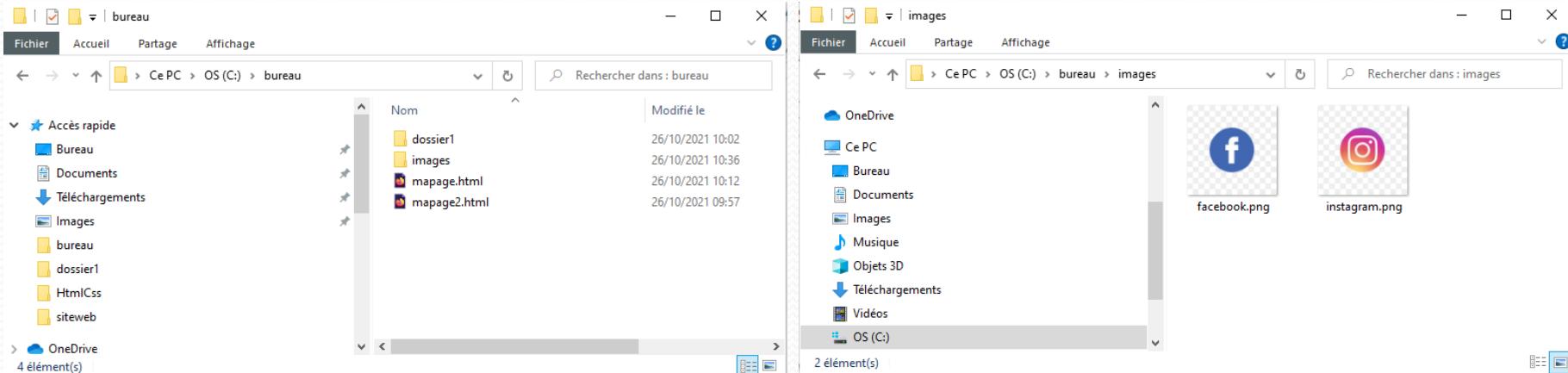
Le langage Html

- Vous venez de réaliser un lien vers une page située sur un autre site en utilisant un **lien absolu**.



Le langage Html

- Dans le dossier de travail « **siteweb** », créez un dossier « **images** ».
- Placez les fichiers « **facebook.png** » et « **instagram.png** » dans ce dossier.



Le langage Html

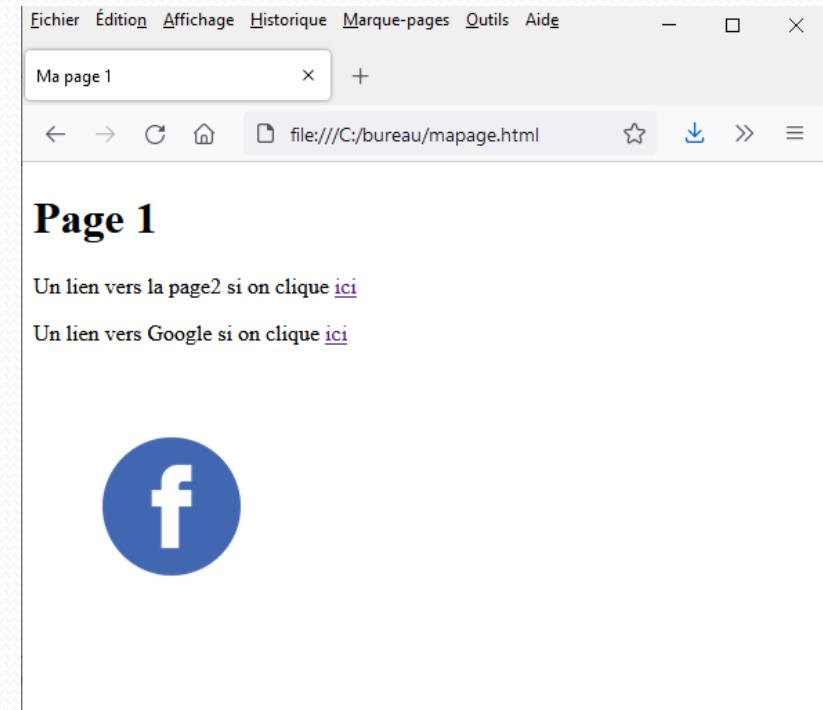
- Modifiez le fichier « **mapage.html** » pour insérer une image avec la balise **** :

```
mapage.html - Bloc-notes
Fichier Edition Format Affichage Aide
<!DOCTYPE html>

<html>

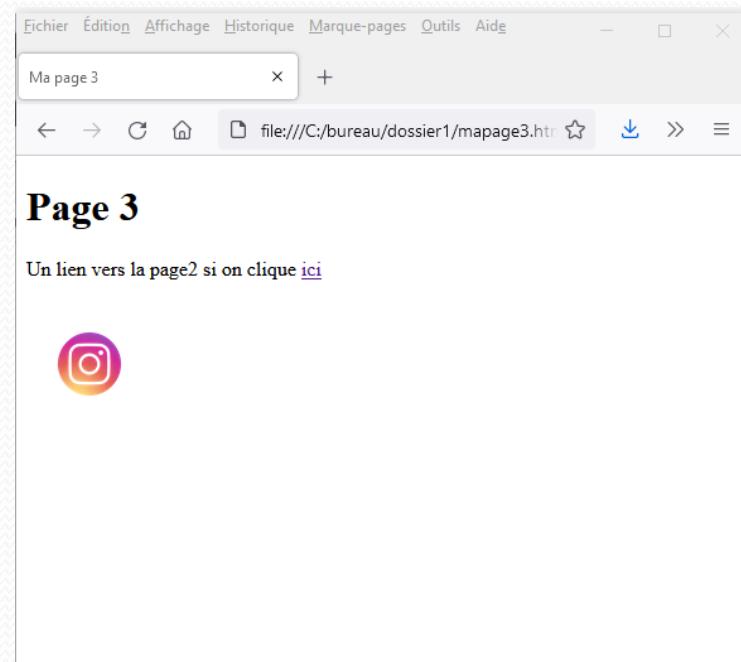
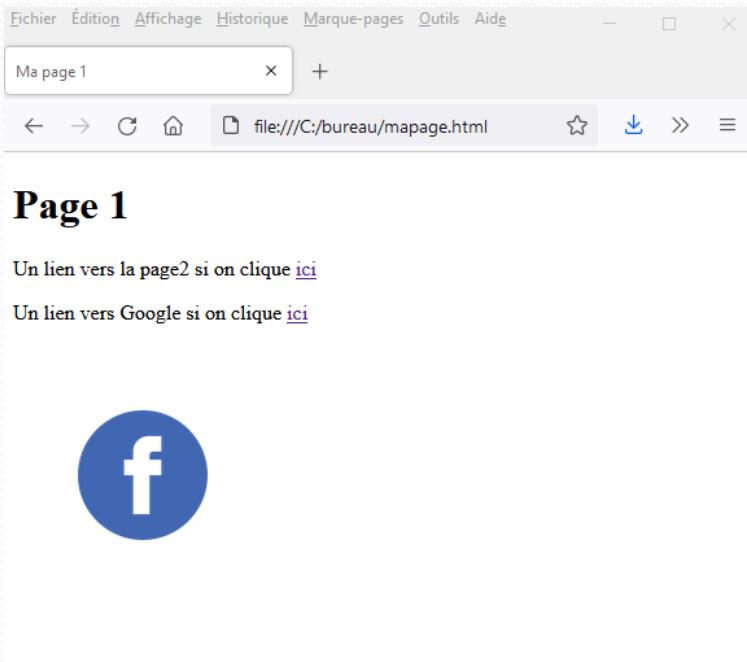
    <head>
        <meta charset="UTF-8">
        <title>Ma page 1</title>
    </head>

    <body>
        <h1>Page 1</h1>
        <p>Un lien vers la page 2 si on clique
            <a href="mapage2.html">ici</a>
        </p>
        <p>Un lien vers Google si on clique
            <a href="http://www.google.fr">ici</a>
        </p>
        
    </body>
</html>
```



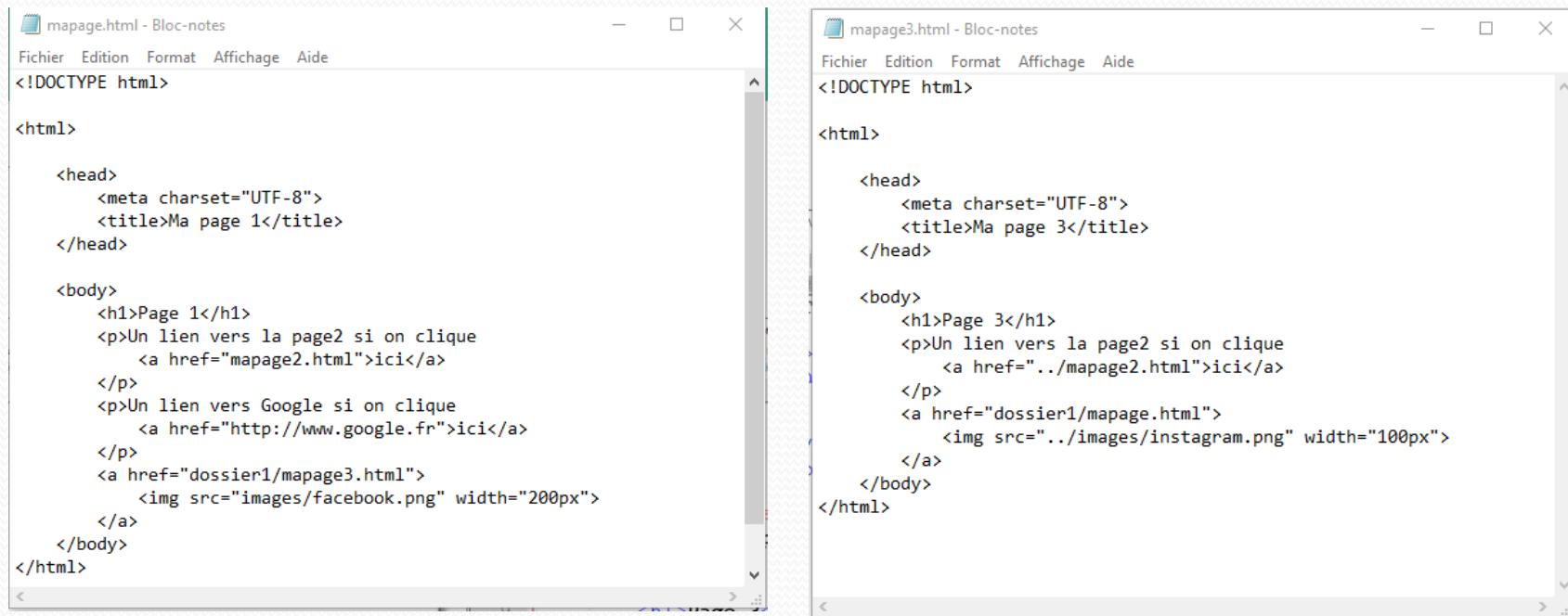
Le langage Html

- A l'aide des ressources à votre disposition, modifiez le fichier « **mapage.html** » pour ouvrir la page « **mapage3.html** » lorsque l'on clique sur l'image.
- Modifiez « **mapage3.html** » pour afficher l'image « **logo-html5.png** » avec une largeur de 100px.
- Modifiez « **mapage3.html** » pour ouvrir la page « **mapage.html** » lorsque l'on clique sur l'image.



Le langage Html

- Solution proposée :



The image shows two windows of Microsoft Notepad side-by-side, each displaying an HTML file.

mapage.html - Bloc-notes

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset="UTF-8">
        <title>Ma page 1</title>
    </head>
    <body>
        <h1>Page 1</h1>
        <p>Un lien vers la page2 si on clique
            <a href="mapage2.html">ici</a>
        </p>
        <p>Un lien vers Google si on clique
            <a href="http://www.google.fr">ici</a>
        </p>
        <a href="dossier1/mapage3.html">
            
        </a>
    </body>
</html>
```

mapage3.html - Bloc-notes

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset="UTF-8">
        <title>Ma page 3</title>
    </head>
    <body>
        <h1>Page 3</h1>
        <p>Un lien vers la page2 si on clique
            <a href="../mapage2.html">ici</a>
        </p>
        <a href="dossier1/mapage.html">
            
        </a>
    </body>
</html>
```

Le langage Html

Éléments en ligne « inline »

- Ces éléments sont des éléments dit « inline »
- On les place autour du contenu (et du texte)
- Ne crée pas de retour à la ligne
- Servent à donner du sens aux parties du texte/contenu

L' élément ``

- Texte en gras

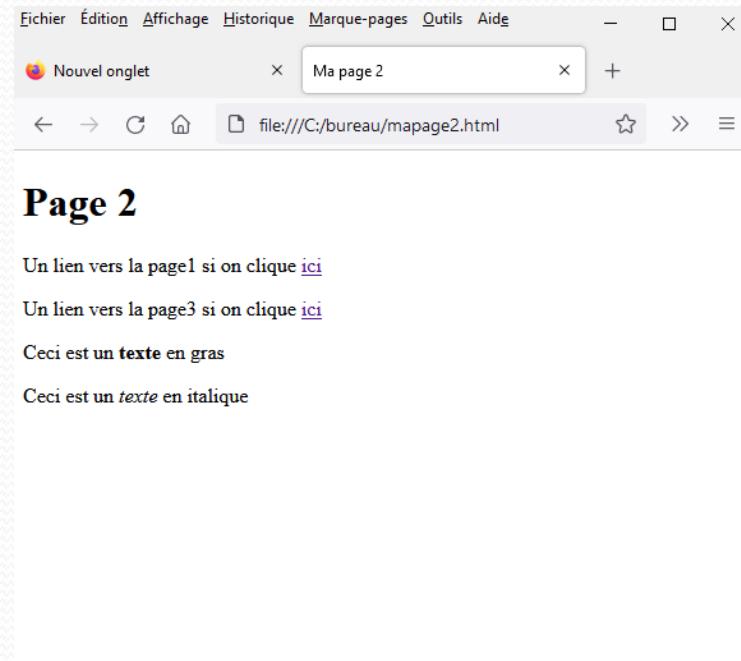
L' élément ``

- Texte en italique

Ces deux balises sont utiles au référencement.

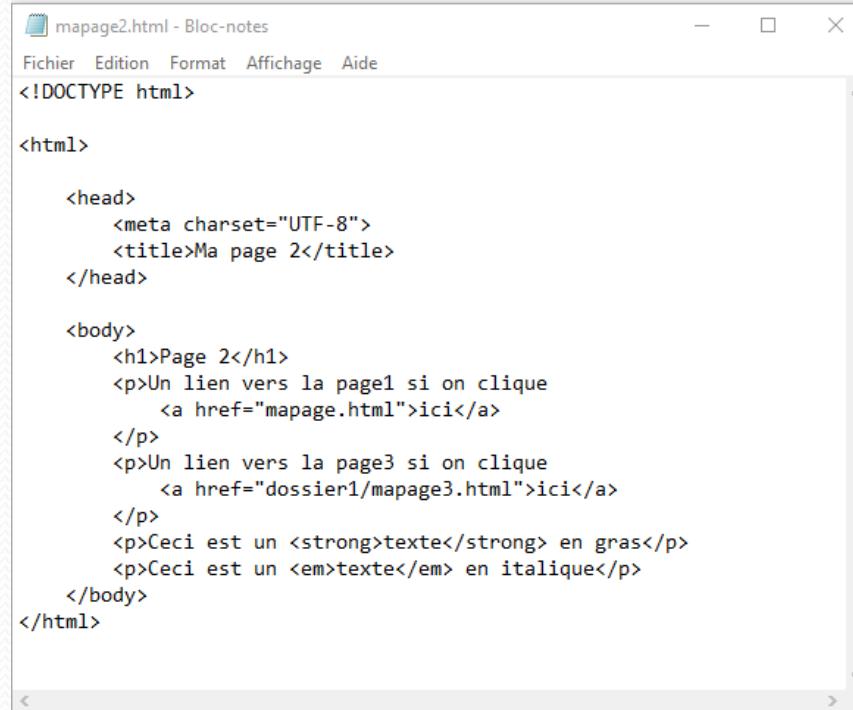
Le langage Html

- En utilisant ces deux dernières balises, modifiez le fichier « **mapage2.html** » pour obtenir le résultat suivant :



Le langage Html

- La solution proposée



The screenshot shows a Microsoft Notepad window titled "mapage2.html - Bloc-notes". The window contains the following HTML code:

```
<!DOCTYPE html>

<html>

    <head>
        <meta charset="UTF-8">
        <title>Ma page 2</title>
    </head>

    <body>
        <h1>Page 2</h1>
        <p>Un lien vers la page1 si on clique
            <a href="mapage.html">ici</a>
        </p>
        <p>Un lien vers la page3 si on clique
            <a href="dossier1/mapage3.html">ici</a>
        </p>
        <p>Ceci est un <strong>texte</strong> en gras</p>
        <p>Ceci est un <em>texte</em> en italique</p>
    </body>
</html>
```

Le langage Html

L' élément

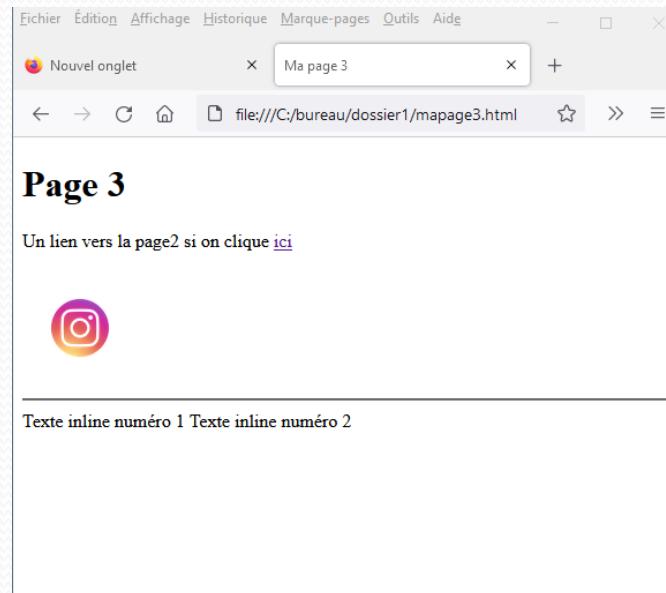
- Un élément « générique » structurant de type « inline »
- Permet de styler facilement des morceaux de texte en CSS
- Par défaut affiche son contenu sans le modifier
- l'équivalent d'une div, en ligne

On va utiliser également l'élément <hr>

- Cette balise crée un trait horizontal
- Peut être employée pour séparer le contenu dans un document HTML

Le langage Html

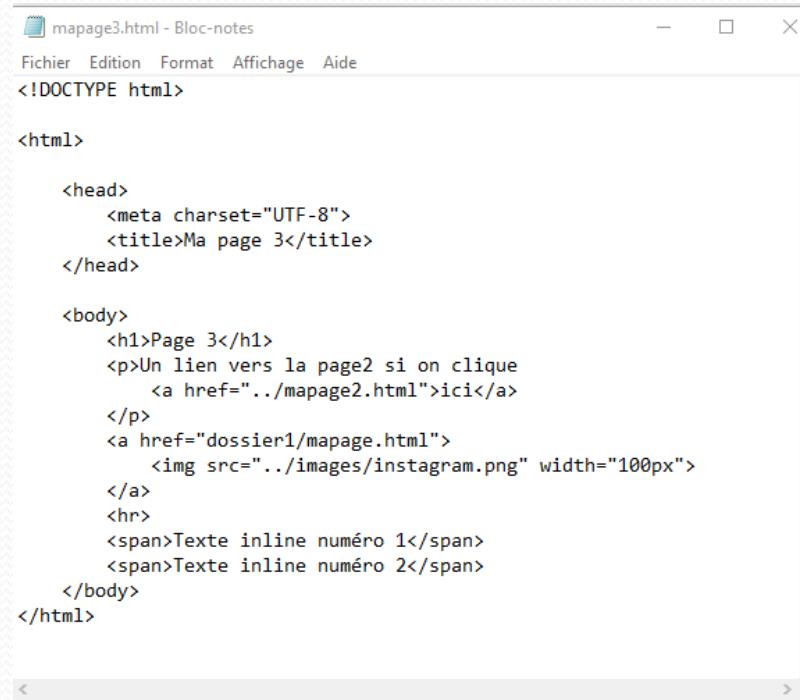
- En utilisant ces deux dernières balises, modifiez le fichier « **mapage3.html** » pour obtenir le résultat suivant :



Le langage Html

- Solution proposée.

Contrairement à la balise `<p>`,
les deux textes contenus dans la balise `` s'affichent sur la même ligne

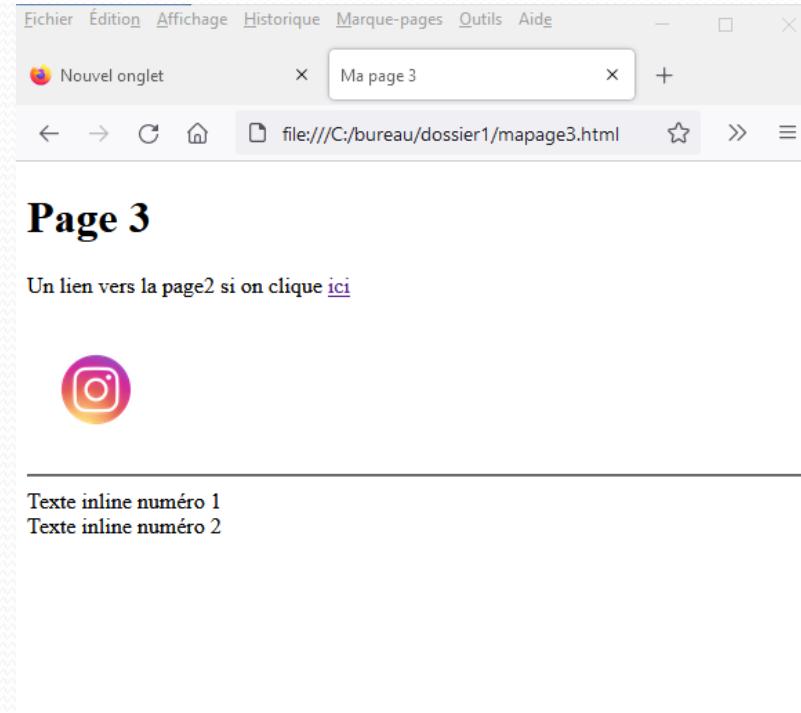


```
mapage3.html - Bloc-notes
Fichier Edition Format Affichage Aide
<!DOCTYPE html>

<html>
    <head>
        <meta charset="UTF-8">
        <title>Ma page 3</title>
    </head>
    <body>
        <h1>Page 3</h1>
        <p>Un lien vers la page2 si on clique
            <a href="../mapage2.html">ici</a>
        </p>
        <a href="dossier1/mapage.html">
            
        </a>
        <hr>
        <span>Texte inline numéro 1</span>
        <span>Texte inline numéro 2</span>
    </body>
</html>
```

Le langage Html

- Remplacez les balises par les balises <div>



Le langage Html

A ce stade, vous êtes capables de :

- ✓ Créer un fichier HTML
- ✓ Mettre en place la structure de base d'un fichier HTML
- ✓ Changer le titre d'une page
- ✓ Editer du contenu textuel (hiérarchie des titres, paragraphes, retour à la ligne)
- ✓ Faire des liens entre pages (relatifs et absolus)
- ✓ Insérer une image et la redimensionner
- ✓ Ajouter des balises de type inline (gras, italique)

Le langage Css

Pour manipuler tous ces éléments et obtenir le rendu attendu de la page sur l'écran, il va falloir à présent s'intéresser au **CSS** (Cascading Style Sheets).

Nous souhaitons par exemple modifier les caractéristiques de présentation du style **<h3>** en mettant le texte en italique et de couleur bleu.

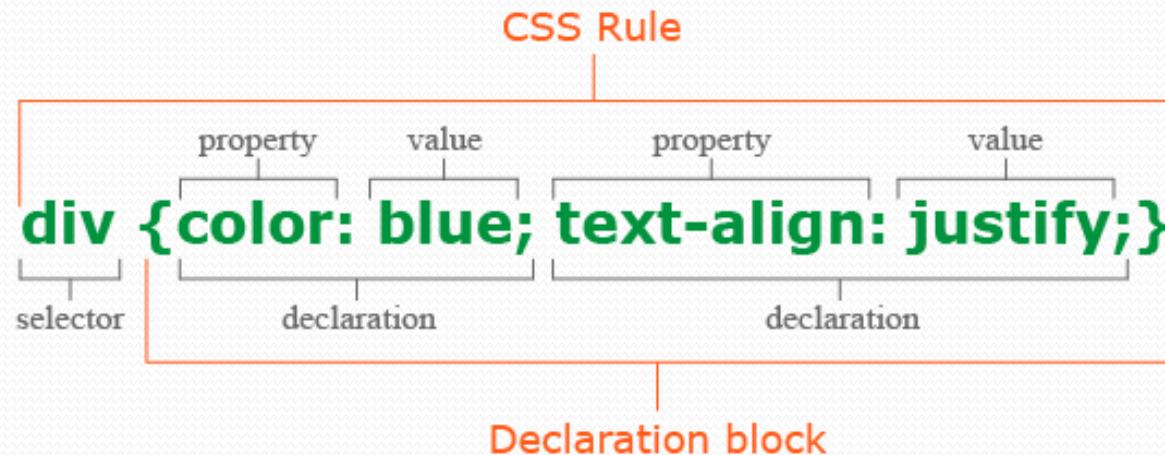
Voici un exemple de règle CSS le permettant :

```
h3 {  
    font-style : italic;  
    color : blue;  
}
```

Le langage Css

Une règle est composée de deux parties :

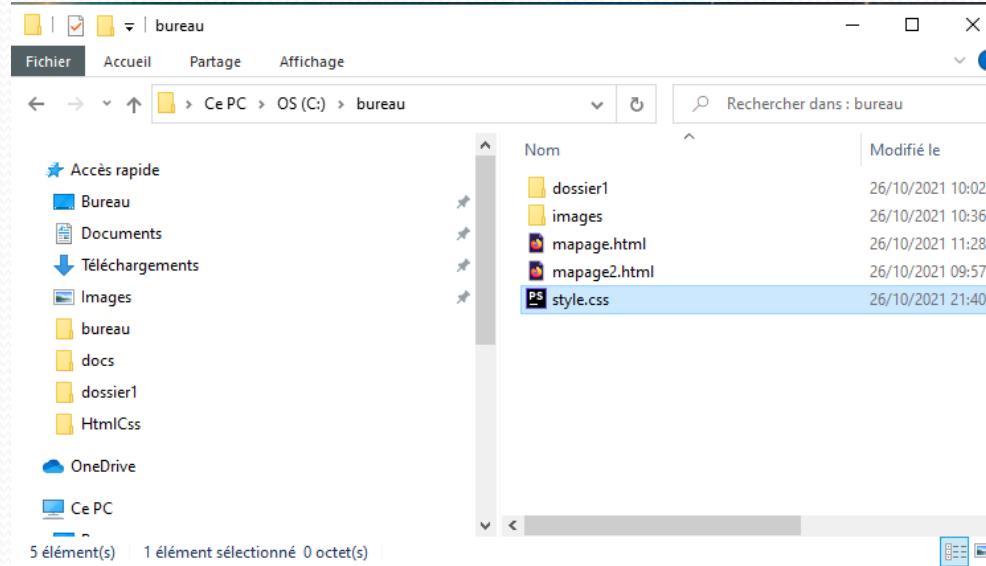
- un **sélecteur CSS** qui indique à quels éléments HTML s'applique le style
- un **bloc de déclaration** dans lequel sont présents des paires **propriété/valeur** séparées par des points-virgules.



Le langage Css

Ces règles CSS seront décrites dans un fichier qui portera l'extension .CSS, **style.css** par exemple.

Dans notre projet, créer un nouveau fichier et nommez le « **style.css** » qui sera notre feuille de style externe.



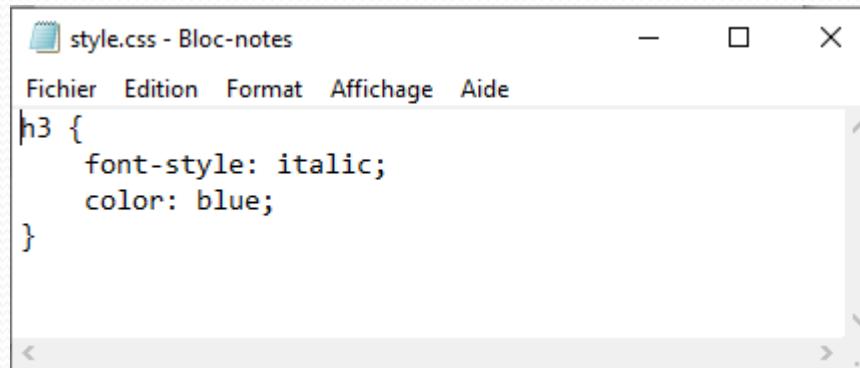
Le langage Css

On peut sélectionner n'importe quel élément HTML et lui appliquer un style.

Une propriété appliquée sur un élément HTML, s'applique par défaut à tous les éléments présents dans le document HTML.

Ajoutons la déclaration de style pour la balise `<h3>` dans ce fichier. Comme pour le fichier HTML, on peut observer la structure du document CSS.

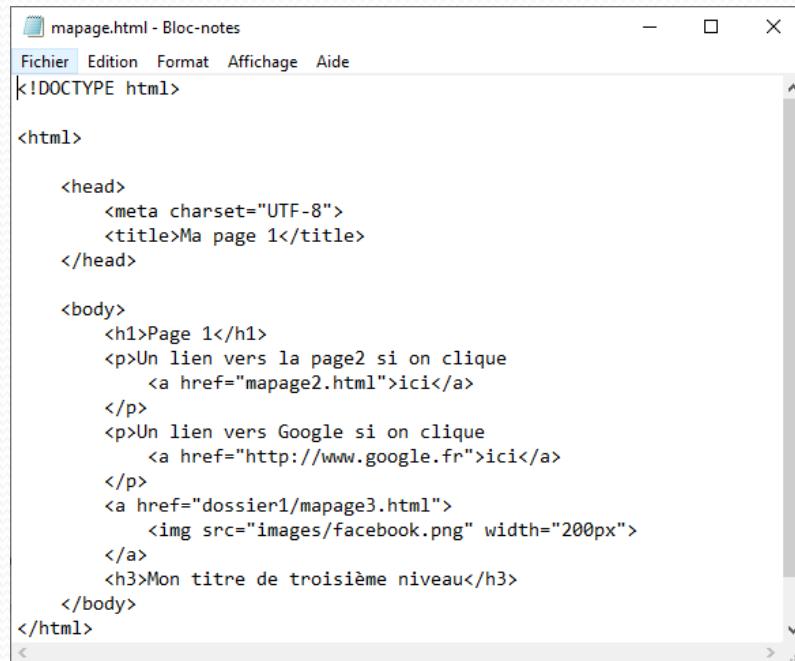
Tous les titres h3 auront le même format : texte en italique et bleue.



```
style.css - Bloc-notes
Fichier Edition Format Affichage Aide
h3 {
    font-style: italic;
    color: blue;
}
```

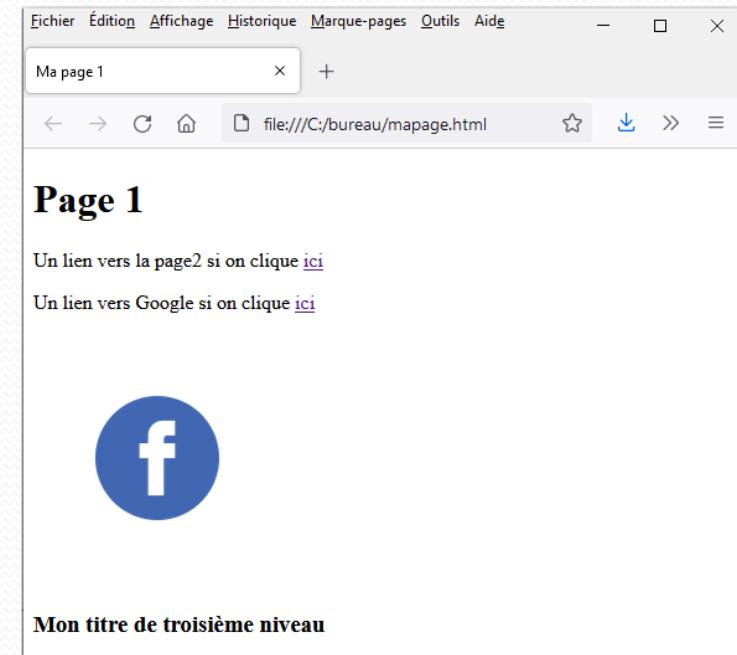
Le langage Css

Ajoutons un titre **<h3>** dans notre HTML et observons le résultat dans le navigateur.



```
mapage.html - Bloc-notes
Fichier Édition Format Affichage Aide
<!DOCTYPE html>

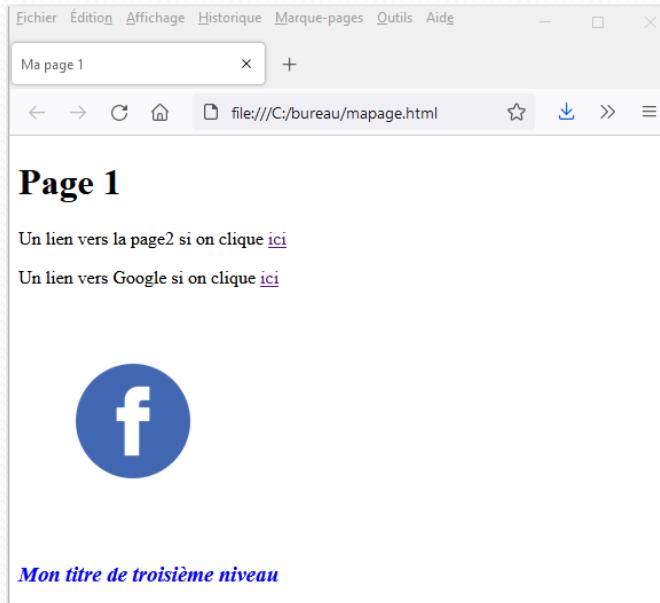
<html>
    <head>
        <meta charset="UTF-8">
        <title>Ma page 1</title>
    </head>
    <body>
        <h1>Page 1</h1>
        <p>Un lien vers la page2 si on clique
            <a href="mapage2.html">ici</a>
        </p>
        <p>Un lien vers Google si on clique
            <a href="http://www.google.fr">ici</a>
        </p>
        <a href="dossier1/mapage3.html">
            
        </a>
        <h3>Mon titre de troisième niveau</h3>
    </body>
</html>
```



Le langage Css

La mise en forme n'est pas prise en compte car nous n'avons pas associé le fichier « **style.css** » au document « **index.html** ». Pour cela, il faut ajouter dans l'entête HTML la ligne suivante :

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



Le langage Css

Jusqu'à présent nous avons utilisé un simple bloc note pour notre compréhension.
Mais l'écriture du code nécessite l'utilisation d'outils plus performants :

- Pas de coloration syntaxique
- Nécessite de connaître tout le language
- Pas de correction des erreurs
- Pas d'autocomplétion du code...

Le langage Css

Nous avons à notre disposition des IDE (Integrated Development Environment) prennants en compte ces fonctions :

- Notepad++ (gratuit)
- Brackets
- Sublime Text
- Atom
- VisualStudio de Microsoft (gratuit) Le plus populaire.
- JetBrains **PhpStorm** (payant)

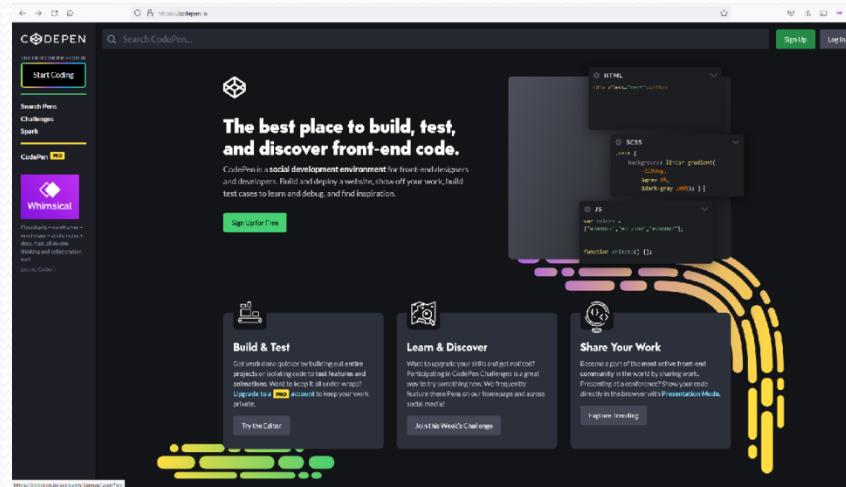


Le langage Css

Il existe aussi en ligne des outils qui nous permettent de tester du code :

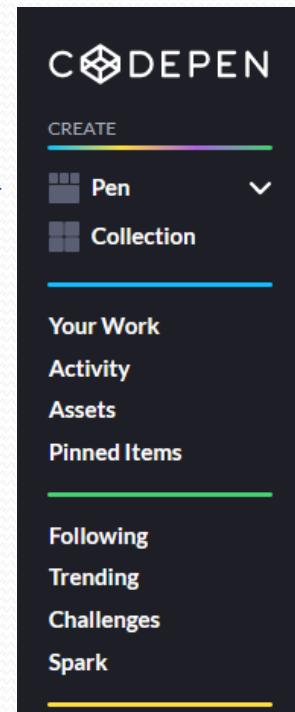
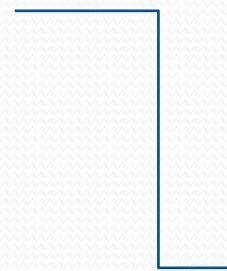
<https://codepen.io>

- Pour notre apprentissage, nous allons juste tester quelques possibilités du language Css et en voir les principes
- Inscrivez vous sur le site

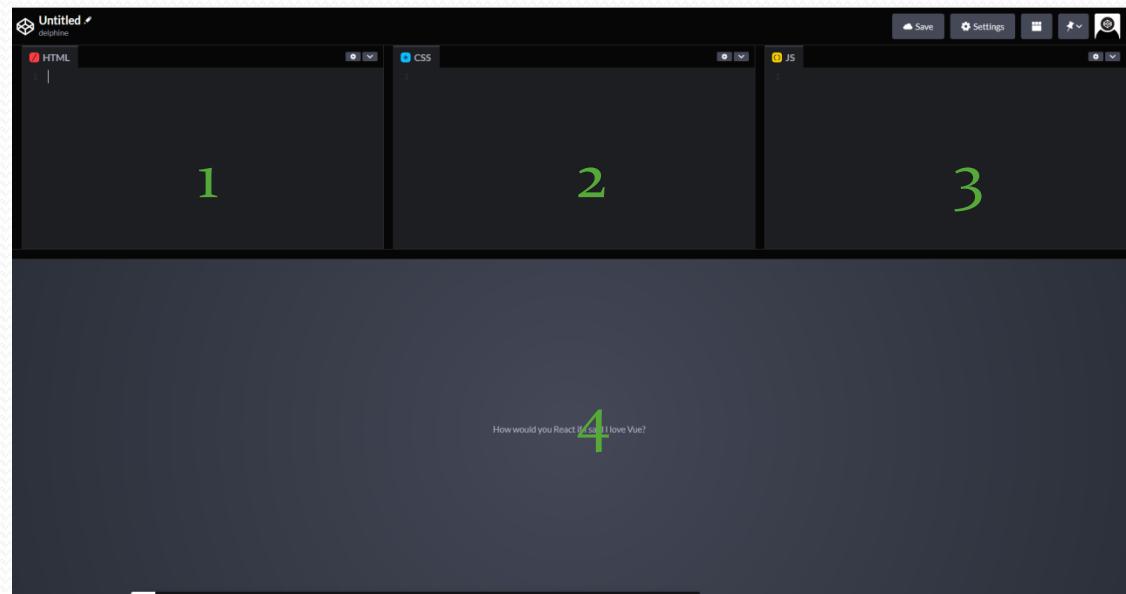


Le langage Css

- Cliquez sur « Pen » :



- 1) emplacement du Html
- 2) emplacement zone du Css
- 3) emplacement du Javascript
- 4) fenêtre du résultat



Le langage Css

- Les sélecteurs de classe
- class=" " est un attribut qui peut se mettre sur n'importe quelle balise Html
Il permet de cibler de manière plus « spécifique » certains éléments

<p class="important"> Un paragraphe spécifique qui est mis en avant </p>

Il est possible d'avoir plusieurs classes sur un élément, en les séparant d'un espace.

<p class="remarque important"> Une remarque mis en avant </p>

Une même classe peut être utilisée sur plusieurs balises dans le même document

<p class="remarque"> une remarque paragraphe</p>

<blockquote class="remarque "> une remarque citation</blockquote>

Le langage Css

- Cibler une classe indépendamment de la balise

Par exemple à l'intérieur du fichier Css .remarque { } va cibler

```
<p class="remarque"> ... </p>
<blockquote class="remarque"> .... </blockquote>
```

Une même classe peut être utilisée sur plusieurs balises HTML

- Cibler une balise HTML avec une classe

Pour cibler un élément doté d'une classe en CSS on inscrit : element.maclasse { }

Par exemple : p.monparagraphe { } va cibler tous les éléments comme celui-ci :

```
<p class="monparagraphe"> ... </p>
```

Le langage Css

- L'id (identifiant)
- Il a le même rôle qu'une classe, mais doit être unique sur la page

Un seul attribut id par balise est possible

On le note <element id="monid">

On le cible en CSS avec #monid { }

Le langage Css

- Convention de nommage

Un nom de classe ou d'ID ne doit pas avoir d'accent ou de caractères spéciaux

On a tendance à éviter le « tout majuscule »

Un nom d'ID ne prend jamais d'espace

Un nom de classe ne prend pas d'espace, sinon c'est une nouvelle classe

Le langage Css

- En résumé

Html ciblé	Css
<p> </p>	p{}
<element class=<< laclasse" > ... </element>	.laclasse { }
<element id="monid"> ... </element>	#monid { }
<p class=<< laclasse" > ... </p>	p.laclasse { }
<p> <element class=<< laclasse" > </element>	p .laclasse { }
<p> ... <p> <div> ... </div>	p, div { }

Le langage Css

Les pseudos-classes sur les liens

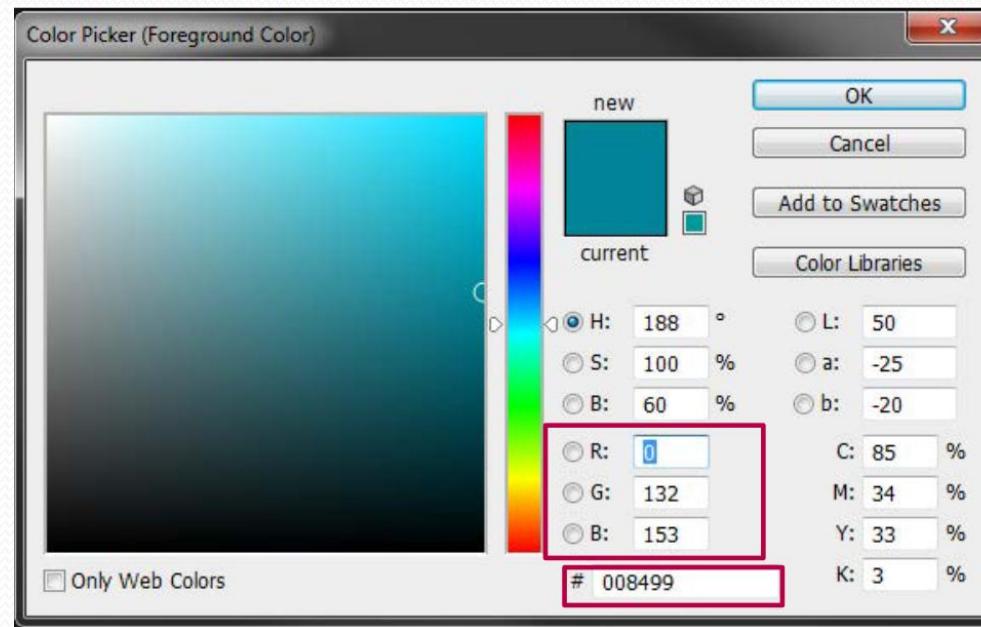
- Le sélecteur `a:link` désigne les liens hypertextes non visités.
- Le sélecteur `a:visited` désigne les liens hypertextes déjà visités
- Le sélecteur `a:hover` désigne un lien survolé.
- Le sélecteur `a:active` désigne un lien cliqué.

`:hover` peut être appliqué sur d'autres éléments que des liens (à partir de IE6).

On peut écrire `p:hover` pour créer un effet au survol d'un paragraphe par exemple

Le langage Css

- Les couleurs en CSS
-



Le langage Css

- Notation hexadécimale : #ffffff (blanc)
- Notation hexadécimale courte (qui est doublée pour obtenir la version longue) : #fff
- Notation RGB : rgb(255,255,255) / rgba(255,255,255,1)
- Mot clé : white
- La couleur est une propriété utilisable sur :
 - color
 - background
 - border
 - ...

Le langage Css

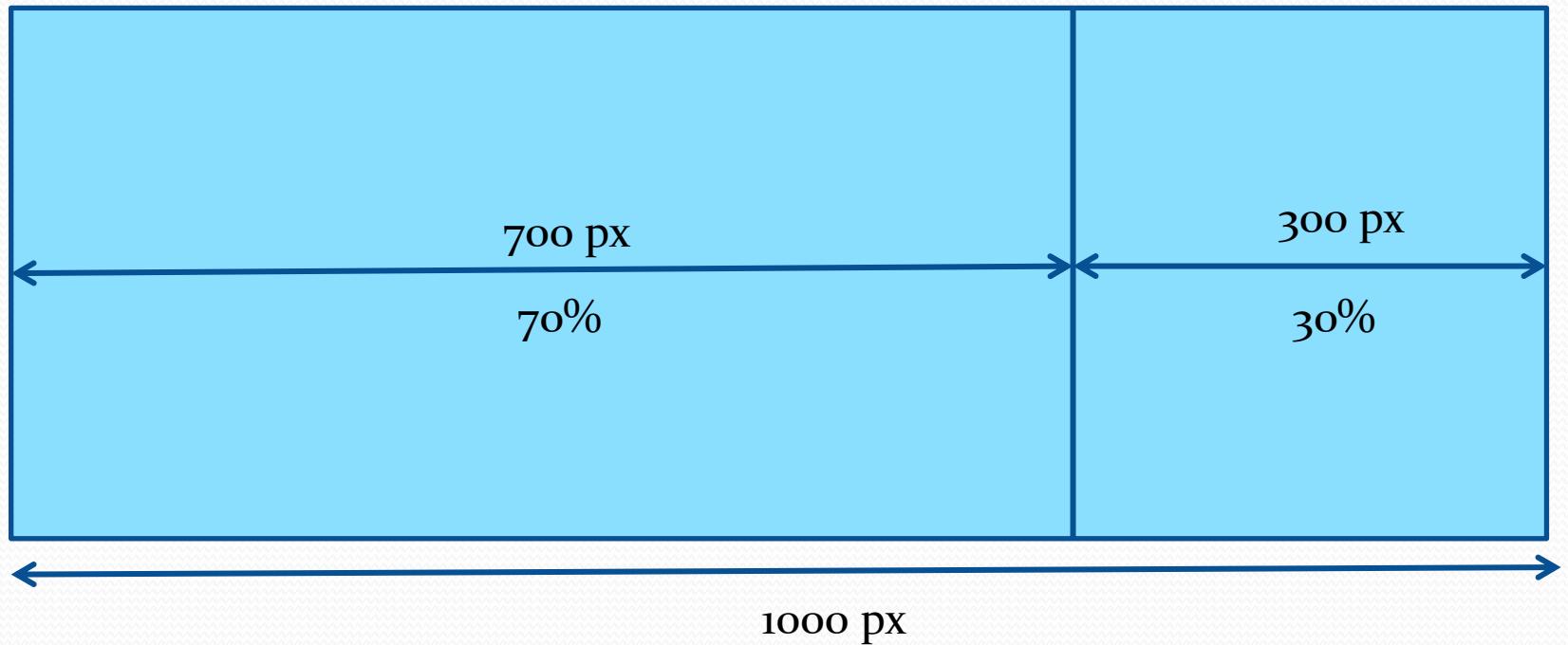
- Le Pixel – px
- Unité « fixe »
 - Dépendance à la résolution du périphérique
 - Valeur « compilée » finale en CSS

Le langage Css

- Le cadratin – em
- Unité fluide utilisée pour les polices d'écriture
- Sa taille est relative à la taille de la police de l'élément parent.
- Utiliser des polices en em permet de garder le rythme typographique et les proportions entre les différents éléments

Le langage Css

- Le pourcentage : %
- Unités fluides (relatives à la taille de leur parent) :
 - %: proportion relative à la dimension de l'élément parent ou la taille de la police selon la propriété.



Le langage Css

- Quelle unité utiliser ?
- "px" pour exprimer une dimension fixe (indépendante de la taille de la police)
- "em" pour attribuer des tailles de polices en fonction de la taille de la police de leur parent.
- "%" pour attribuer des dimensions proportionnées aux dimensions de l'élément parent.
- Une mise en page fluide s'appuie généralement sur un dimensionnement des éléments en pourcentage

Le langage Css

- Propriétés de typographie, polices
- font-family : indiquer la (ou les) polices à utiliser
La police doit être présente sur le poste de l'utilisateur
- font-size : valeur en unité permet de définir la taille d'une police (= la « hauteur »)
On utilise les unités vues précédemment (px, em, etc.)
- font-weight: valeur permet de déterminer l'épaisseur de la police.
Principales valeurs possibles : normal, bold (gras)
Permet de mettre en gras d'autres éléments
- font-style : valeur permet de préciser le style de police
Principales valeurs possibles : normal, italic, oblique.
Permet de mettre en italique des éléments

Le langage Css

- `line-height`: valeur change l'interlignage

Les valeurs possibles sont des valeurs numériques en unité de mesure (px, em, %) mais il vaut mieux ne pas mettre d'unité du tout pour garder plus de flexibilité.

- `color`: valeur permet de donner la couleur du texte

Les valeurs possibles sont toutes les couleurs présentées précédemment

Le langage Css

- **text-align** : valeur permet d'aligner le texte à gauche, droite ou milieu
- Les valeurs possibles sont : left, right, center, justify (comme sur Word)
- L'alignement se fait sur le contenu des balises de type bloc (paragraphes, titres, divs, etc.)
- **text-transform** permet de transformer le texte en majuscule, minuscule
Valeurs possibles :
 - none
 - capitalize
 - uppercase
 - lowercase
- **text-decoration** : valeur permet de changer différents valeurs de « décoration »
Valeurs possibles :
 - none: permet d'empêcher l'héritage de la propriété et plus particulièrement de supprimer le soulignement par défaut des liens
 - overline : surlignement
 - underline : soulignement
 - line-through: texte barré

Le langage Css

- Les bordures
- La propriété border permet d'ajouter une bordure à un élément

Elle a 3 « sous propriétés » possibles :

- style,
- color ,
- et width.

On regroupe souvent comme ceci : border: width style color;

Par exemple :

```
h2 {  
    border: 2px solid #009860 ;  
}
```

Le langage Css

- On peut décomposer les 4 bordures de la manière suivante :

border-top: ...;

border-left: ...;

border-right: ...;

border-bottom: ...;

- Si on ne précise pas à quelle bordure on applique la propriété, elle s'applique aux 4.
- Dans l'exemple ci-dessous :

margin:20px 10px 5px 0px

20px => marge haut

10px => marge droite

5px => marge bas

0px => marge gauche

si uniquement 2 chiffres : haut+bas / gauche+droite

Le langage Css

- Style de bordure :

none : pas de bordure (par défaut);

dotted : pointillés;

dashed : tirets;

solid : un trait simple;

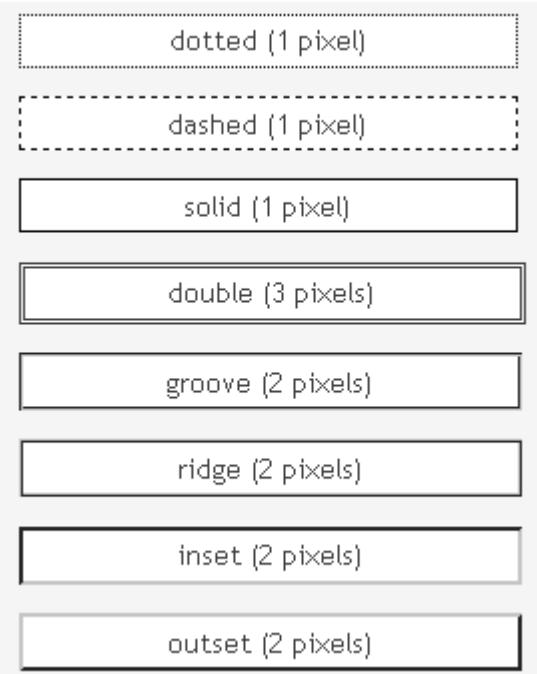
double : bordure double;

groove : en relief;

ridge : autre effet relief;

inset : effet 3D global enfoncé;

outset : effet 3D global surélevé.

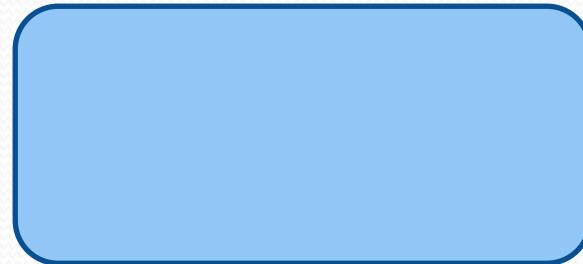


Le langage Css

- border-radius : valeur permet de faire des bords arrondis (supporté à partie de IE9, propriété CSS3)
Valeur : une valeur en unité de mesure (px, em ou même %)
 - Le même arrondi partout : border-radius : 10px;
 - Un arrondi par angle (en haut à gauche, en haut à droite, en bas à droite, en bas à gauche) : border-radius : 2px 20px 5px 10px;

Le langage Css

- background-color : valeur permet de donner une couleur de fond.
- Valeurs : n'importe quelle couleur CSS
- Sur un élément bloc (prend toute la largeur) ou inline (prend la largeur du contenu)



Le langage Css

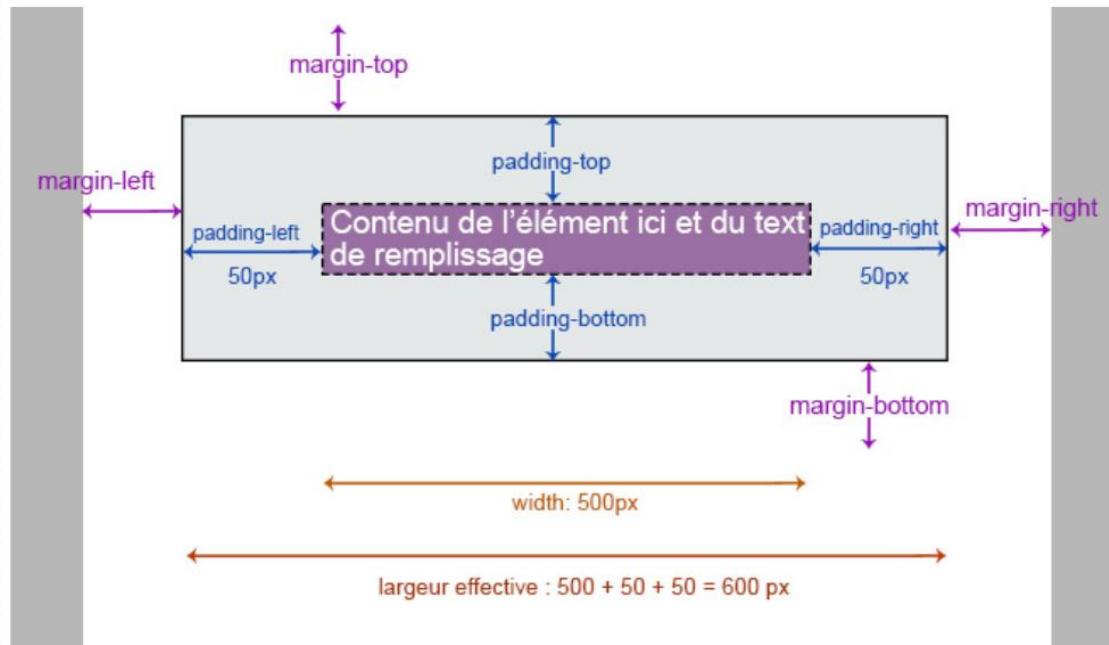
- background-image : valeur permet de définir une image de fond pour l'élément
Valeur : none ou url("monimage.png")
Par défaut : l'image se répète horizontalement et verticalement (comme une mosaïque)
- background-repeat permet de définir la façon dont l'image va se répéter.

Le langage Css

- Dimensions, margin et padding
- Largeur et hauteur d'un élément : width et height width (largeur) et height (hauteur) ne sont applicables QUE sur des éléments de bloc
Valeurs possibles : auto (par défaut, toute la taille du parent), valeur numérique en px, % ou même em. body{ width: 800px;}

Le langage Css

- Marges
- Il existe deux types de marges : intérieure appelée padding, et extérieure appelée margin



Le langage Css

- La propriété margin détermine l'espace entre le bord de l'élément, et l'élément suivant.
Par défaut margin: valeur applique la même valeur aux 4 cotés
- Valeur possible : auto, valeur en unité de mesure, positive ou négative
On peut la décomposer individuellement en :
 - margin-top : marge extérieure haute
 - margin-right : marge extérieure droite
 - margin-bottom : marge extérieure basse
 - margin-left : marge extérieure gaucheOu rassembler les 4 valeurs : margin: 10px 5px 8px 15px; (dans l'ordre : haut, droite, bas, gauche)

Le langage Css

- Padding : la marge « intérieure »
- La propriété padding détermine un espacement entre le bord de la boite et son contenu
Par défaut padding : valeur applique la même valeur aux 4 côtés
Valeur possible : valeur en unité de mesure
- On peut la décomposer individuellement en :
 - margin-top : marge extérieure haute
 - margin-right : marge extérieure droite
 - margin-bottom : marge extérieure basse
 - margin-left : marge extérieure gaucheOu rassembler les 4 valeurs : margin: 10px 5px 8px 15px; (dans l'ordre : haut, droite, bas, gauche)

Le langage Css

Aller plus loin...

- Flexbox pour créer des mises en page fluides :

<https://www.alsacreations.com/tuto/lire/1493-css3-flexbox-layout-module.html>

- Adapter son site au mobile : le responsive webdesign

<https://mediaqueri.es/>

```
<link rel="stylesheet" media="screen and (max-width: 640px)" href="smallscreen.css" />
```

```
@media screen and (max-width: 640px) {  
    .bloc {  
        display:block;  
    }  
}
```

SÉLECTEURS

*	Tous les éléments
div	Toutes les balises div
div, span	Toutes les balises span
div span	Tous les span dans les div
div > span	Tous les span enfants des div
#bloc	Identifiant unique (id="bloc")
.bloc	Classe (class="bloc")
ul#bloc	Une liste (ul) avec un id bloc
span.bloc	Span ayant la classe bloc
#bloc *	Tous les éléments dans #bloc
a:link	Lien
a:active	Lien actif
a:visited	Lien visité
a:hover	Lien survolé

TEXTE

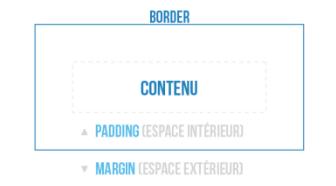
font-family	Famille de typos (arial, times)
font-size	Taille de la typo (px/em)
font-weight	Graisse (bold, normal)
font-style	Orientation (italic, normal)
color	Couleur (#F00)
text-transform	Transformation (uppercase)
text-decoration	Décoration (underline)
text-align	Alignement (left, right, center)
line-height	Hauteur de ligne (px/em)
letter-spacing	Espacement entre les lettres
text-indent	Décalage de la première ligne

ESPACEMENT & BORDURES

margin	Marge extérieure (px/em)
margin-top	Marge en haut
margin-right	Marge à droite
margin-bottom	Marge en bas
margin-left	Marge à gauche
padding	Espace intérieur
border	bordure (1px solid #F00)

ARRIÈRE-PLAN

background	Arrière-plan
background-color	Couleur de fond
background-image	Image de fond
background-repeat	Répétition du fond
background-attachment	Attachement du fond
background-position	Position du fond



LISTES

list-style-type	Type de puce
list-style-position	Position de la puce
list-style-image	Image de la puce

POSITIONNEMENT ET TAILLES

width	Largeur (px/em)
min-width	Largeur minimale
max-width	Largeur maximale
height	Hauteur (px/em)
min-height	Hauteur minimale
max-height	Hauteur maximale
position	Position (absolute, relative)
top	Référence au haut
right	Référence à la droite
bottom	Référence au bas
left	Référence à la gauche
float	Element flottant (left/right)
display	Mode d'affichage (block)
z-index	Profondeur/niveau
overflow	Dépassement hors conteneur

POSITIONS ET TAILLES

width	Largeur (px/em)
height	Hauteur (px/em)
position	Position (absolute, relative)
top	Référence au haut
right	Référence à la droite
bottom	Référence au bas
left	Référence à la gauche
float	Element flottant (left/right)
display	Mode d'affichage (block)
z-index	Profondeur/niveau
overflow	Dépassement hors conteneur

NOUVEAUTÉS CSS 3

font-face	Chargement d'une typo
border-radius	Coins arrondis
box-shadow	Ombre d'un bloc
gradient	Degradé
opacity	Transparence
text-shadow	Ombre sur un texte
transform	Transformation
transition	Transition

MEDIAS QUERIES

Ex : @media screen and (max-width: 640px) { ... }

Médias

screen	Écrans
handheld	Mobiles / Petits périphériques
print	Impression
braille	Plages braille
embossed	Imprimantes braille
projection	Projecteurs (ou slides)
tty	Terminal/police à pas fixe
tv	Téléviseur
all	Tous les précédents

Opérateurs

and	et
only	seulement
not	non

Critères

color	Support de la couleur
color-index	Couleurs indexées
device-aspect-ratio	Ratio du périphérique (16/9)
aspect-ratio	Ratio de la zone d'affichage
device-height	Hauteur du périphérique
device-width	Largeur du périphérique
grid	Périphérique bitmap (ex : lcd)
height	Hauteur de la zone d'affichage
monochrome	Monochrome / Noir et blanc
orientation	Orientation (portrait/landscape)
resolution	Résolution du périphérique
width	Largeur de la zone d'affichage

PSEUDO CLASSES

:after	Insertion de contenu après
::before	Insertion de contenu avant
::first-letter	Selection de la première lettre
::first-line	Selection de la première ligne
::selection	Selection

:active	Actif
::checked	Coché/checké
::disabled	Désactivé
::empty	Vide
::enable	Activé
::first-child	Premier enfant
::first-of-type	Premier du type
::focus	Curseur dedans
::hover	Au dessus
::in-range	Dans une plage de valeurs
::invalid	Invalidé
::lang(language)	Dans la langue (fr/en/it/...)
::last-child	Dernier enfant
::last-of-type	Dernier du type
::link	Lien
::not(selector)	N'est pas un sélecteur
::nth-child(n)	Chaque (n) enfant
::nth-last-child(n)	Dernier de chaque (n) enfant
::nth-last-of-type(n)	Dernier de chaque (n) du type
::nth-of-type(n)	Chaque (n) du type
::only-of-type	Seulement du type
::only-child	Seulement les enfants
::optional	Optionnel
::out-of-range	En dehors des valeurs
::read-only	Lecture seule
::read-write	Lecture et écriture
::required	Requis
::root	Racine du document
::target	Cible
::valid	Valide
::visited	Visité