

Le langage HTML 5

cours réalisé par Anis Assas et Yessine Zekri

assas anis@yahoo.fr & yessine.zekri.isetjb@gmail.com

Le langage HTML : Plan

- Introduction
- Concept de balises
- Structure d'une page HTML 5
- Attributs d'une balise
- Organisation du texte et mise en forme
- Titres
- Listes
- Liens
- Tableaux
- Formulaires
- Les balises multimédia
- Les balises structurelles
- Conventions et bonnes pratiques



HTML : Introduction

- **HTML : Hyper Text Markup Language**
 - Ce n'est pas un langage de programmation
 - C'est plutôt un langage de description
 - Composé de plusieurs balises (tags)
 - Interprété par le navigateur
 - Il est créé en 1991 par *Tim Berners-Lee* et il est standardisé depuis 1994 par W3C



assas anis@yahoo.fr & yessine.zekri.isetjb@gmail.com

Standardisation

- **W3C** (World Wide Web Consortium)
 - organisme de standardisation fondé par Tim Berners-Lee
 - chargé de promouvoir la compatibilité des technologies web (HTML, XML, CSS, SOAP...)
 - **Historique**
 - Les langages de balisage sont issus du langage **SGML** (Standard Generalized Markup Language) créé en 1986 pour structurer des contenus très divers
 - En 1992, le CERN (Centre Européen de Recherche Nucléaire) rend public le projet World Wide Web définissant un langage de présentation et de structuration de documents hypertextes, dérivé de SGML
HTML = HyperText Markup Language
 - En 1998, le langage **XML** (eXtensible Markup Language) a vu le jour
 - Son succès dans de multiples domaines d'application a conduit le W3C (World Wide Web Consortium) à créer le langage **XHTML** (eXtensible HyperText Markup Language) en 2000, non plus comme une nouvelle version de HTML, mais comme une reformulation de HTML en tant qu'une application XML.
- Cours réalisée par Aïss Assassi et Yessine Zekri

HTML : évolution

- **HTML1** : première version créée par Tim Berners-Lee en 1991.
- **HTML2** : deuxième version, apparue en 1994. On commence à parler de W3C.
- **HTML3** : apparue en 1996 avec plusieurs nouveautés comme les tableaux, les scripts, le positionnement du texte autour des images, etc.
- **HTML4** : apparue en 1998 avec la possibilité :
 - d'utiliser de frames (découpage d'une page en plusieurs parties),
 - des améliorations sur les formulaires,
 - d'utiliser des feuilles de style (CSS).
- **HTML5** : finalisée en octobre 2014 et permet de :
 - simplifier les déclarations et intégrer de nouveaux médias
 - ajouter plus de précisions sur les champs d'un formulaire
 - améliorer la structuration du document (intégration de balises sémantiques)

Quel IDE pour HTML?

- **Un éditeur HTML** (ou éditeur Web) : logiciel conçu pour faciliter la préparation et la modification de documents écrits en HTML. Il existe deux catégories d'éditeurs :
 - Les éditeurs visuels : Visual Studio Code, GoLive, DreamWeaver, frontpage, Komodo ...
 - Les éditeurs textuels ou classiques : Notepad++, Sublime Text, Atom, Bluefish, HTMLEdit ...
- **Visual Studio Code:** <https://code.visualstudio.com/docs/Download>
 - Gratuit
 - Pouvant s'adapter selon le langage de programmation
 - Extensible via l'installation de quelques centaines d'extensions
- **NB :** Pour activer la sauvegarde automatique : aller dans **File > Auto save**

Création d'un projet

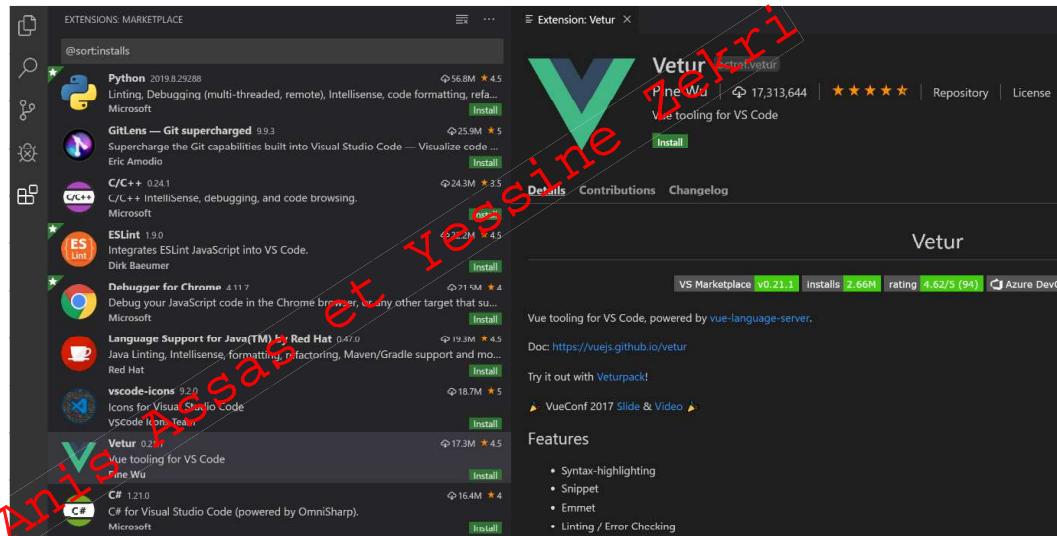
- Pour créer un projet sous **VSC**
 - Allez dans **Fichier > Ouvrir le dossier...**
 - Cliquez sur **Nouveau dossier** et saisissez **formation-html**
 - Cliquez sur le dossier **formation-html** puis sur le dossier **Sélectionner un dossier**
 - Créez un fichier **index.html** dans **formation-html**
 - Dans **index.html**, saisissez **html:5** puis cliquez sur **Entrée**
- **Code généré :**

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
</body>
</html>
```

Extension Live Server

■ Installez l'extension **Live Server**

- Cliquer sur l'icone  ou **Ctrl+Shift+X**
- Tapez par la suite **Live Server**
- Cliquez sur le bouton **Install**



■ Faites un clic droit sur **index.html**

■ Cliquez sur **Open with Live Server**

■ Vous pouvez aussi utiliser l'icone **Go Live**



Concepts de balise : Syntaxe

- Il existe deux types de balises.
 - balise vide** : balise sans contenu. Ses informations apparaissent dans des attributs. Chaque attribut définit une propriété de la balise.
 - Syntaxe** : <nom-balise attribut1 = "valeur1" attribut2 = "valeur2" ... />

- balise non vide** : balise avec contenu, qui apparaît entre <nom-balise> et </nom-balise>
- Syntaxe** : <nom-balise attribut1 = "valeur1" attribut2 = "valeur2" ... > Contenu </nom-balise>
<h1 align = "center">Le langage HTML</h1>
<p>Ceci est un paragraphe</p>
- Commentaire** : balise particulière
 <!-- ceci est un commentaire -->

Concepts de balise : Propriétés

- Les balises doivent être ouvertes puis fermées récursivement, comme des parenthèses
([...]{...})
- Tout ce qui est entouré par deux balises s'appelle : élément
- Un élément peut éventuellement contenir du texte, liens, balises...
- Un texte en clair (non-entouré par une balise) est affiché
- Les commentaires en HTML sont ignorés par le navigateur
- Les commentaires ne sont pas affichés mais restent visibles dans le code source de la page
(NB : ne jamais contenir une information confidentielle).

Structure d'une page HTML : Composition

- Le contenu d'une page HTML est compris entre deux balises **html**
- Une page HTML est composée de deux parties :
 - une entête : **head**
 - un corps : **body**
- DOCTYPE**
 - Ce n'est pas une balise
 - C'est facultatif
 - C'est une directive permettant de préciser qu'il s'agit d'un document HTML et indiquant sa version

```
<!DOCTYPE html>
<html>
  <head>
    </head>
  <body>
    </body>
</html>
```

```
<!-- HTML4.01 transitional -->
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<!-- HTML5 -->
<!DOCTYPE html>
```

HEAD

- Que peut contenir <HEAD> ?

```
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Cours HTML 5</title>
    <link rel="stylesheet" type="text/css" href="styles.css">
    <script src="script.js" type="text/javascript" render></script>
</head>
```

- <title>** titre du document (affiché par le navigateur en haut de la page)
- <link>** pour référencer un fichier (CSS par exemple)
- <style>** pour inclure du code CSS
- <meta>** peut contenir :
 - des informations sur le codage
 - des informations pour les navigateurs et les moteurs de recherche

BODY

- Que peut contenir <BODY> ?

```
<body>
```

Les informations qui seront affichées dans le navigateur

```
</body>
```

- Dans le <Body>

- texte
- tableau
- image/vidéo/document
- menu
- lien
- formulaire
- liste
- ...

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Cours HTML 5</title>
</head>
<body>
  <h1 align="center" >Le langage HTML</h1>
  
</body>
</html>
```

cours réalisé par

Assas et Yessine Zekri

Les attributs d'une balise

- Les balises, dans certains cas, ne suffisent pas.
- On leur associe donc des attributs

- Balise avec un seul attribut
- Balise avec plusieurs attributs
- Balise avec attribut ne prennent pas de valeur

```
<input type="text">
```

```
<input type="text" placeholder="saisir votre nom">
```

```
<input type="text" placeholder="saisir votre nom" readonly>
```

Remarque :

Avec HTML5, la valeur d'un attribut peut être entourée par des guillemets, des apostrophes comme elle peut ne pas être entourée si elle ne contient pas d'espace.

Quelques attributs standards

- **class** : nom de classe pour cibler plusieurs éléments (à voir dans la partie CSS)
`<p class="paragraphe">Paragraphe 1</p>`
- **id** : identifiant unique dans la page pour cibler un seul élément (à voir dans la partie CSS)
`<p class="paragraphe" id="para1">Paragraphe 1</p>
<p class="paragraphe" id="para2">Paragraphe 2</p>`
- **style** : style CSS de l'élément
`<p style="color: blue;">Paragraphe avec du texte en bleu.</p>`
- **width et height** peuvent être utilisés comme attributs pour les balises suivantes :
 - img , canvas, iframe, object, Video

Quelques attributs standards

- Exemples avec width et height :

```
<canvas width="350" height="350" style="background-color: yellow;"></canvas>
```

```
<object data="fichier.pdf" type="application/pdf" width="350" height="350"></object>
```

```
<video width="350" height="350" controls>
  <source src="video.mp4" type="video/mp4">
</video>
```

cours réalisé par Anis Assas et Yessine Zekri

Organisation du texte : Les paragraphes

< p > Un paragraphe </ p >

< i > Un texte en italic </ i >

< b > un texte en gras </ b >

< em > Un texte un peu en valeur </ em >

< strong > Un texte bien en valeur </ strong >

< small > Un texte à une taille inférieure </ small >

< del > Un texte barré </ del >

< ins > Un texte souligné </ ins >

< sub > Un texte en indice </ sub >

< sup > Un texte en exposant </ sup >

< mark > Un texte marqué </ mark >

< i > Italic </ i >	Italic
< b > Bold </ b >	Bold
< em > Emphasized </ em >	Emphasized
< strong > Strong </ strong >	Strong
< small > small </ small >	small
< del > Deleted </ del >	Deleted
< ins > Inserted </ ins >	Inserted
v_f	v _f
a²	a ²
< mark > Marked </ mark >	Marked

Yessine Zekri
Assas et Anis

Autres balises de formatage

- pour souligner un texte
- pour afficher une ligne horizontale
- abréviation

`<u>Un texte souligné</u>`

`<hr>`

- adresse formatée (en italique)

`<address><p>Adresse : 123, Rue de l'Exemple</p></address>`

- citation (en italique)

`<p>Mon livre préféré est <cite>Les misérables</cite> de Victor Hugo</p>`

- élément de code informatique

`<code> select * from client; </code>`

- `<pre>` : texte pré-formaté

`<pre>function Hello(name){ return "Bonjour " + name + " !"; }</pre>`

Exemples

- Avec **del** et **ins**

```
<p>La dernière version de HTML est <del>quatre</del> <ins>cinq</ins></p>
```

- Avec **mark**

```
<p>HTML est très <mark> facile </mark>!.</p>
```

- Avec **address**, **strong**, **br** et **abbr**

```
<address>
  <strong>Twitter, Inc.</strong><br>
  1355 Market Street, Suite 900<br>
  San Francisco, CA 94103<br>
  <abbr title="Phone">Ph:</abbr>(123)456-7890
```

- Avec **pre**

```
<pre>
  Le texte dans une balise "pre" est affiché.
  Les sauts de lignes et les espaces sont respectés.
</pre>
```

Les titres : les différents niveaux

- De plus grand au plus petit

1. `<h1>...</h1>`
2. `<h2>...</h2>`
3. `<h3>...</h3>`
4. `<h4>...</h4>`
5. `<h5>...</h5>`
6. `<h6>...</h6>`

```
<h1>Titre de niveau 1</h1>
<h2>Titre de niveau 2</h2>
<h3>Titre de niveau 3</h3>
<h4>Titre de niveau 4</h4>
<h5>Titre de niveau 5</h5>
<h6>Titre de niveau 6</h6>
```

Titre de niveau 1

Titre de niveau 2

Titre de niveau 3

Titre de niveau 4

Titre de niveau 5

Titre de niveau 6

Les règles à respecter

- Avoir **une seule balise h1** par page
- Rédiger des **h1 différents** pour chaque page : Chaque page devrait avoir son propre titre principal (balise <h1>) qui est spécifique au contenu de cette page. Le titre principal doit être unique pour chaque page.
- Utiliser les **niveaux inférieurs** (**du h2 au h6**) pour structurer le contenu d'une page.
- Les titres ne doivent pas être utilisés uniquement pour **définir** la taille de la police d'un texte : Leur objectif principal est de structurer le contenu de la page et de faciliter la navigation pour les utilisateurs et les moteurs de recherche.
- Les niveaux de titres peuvent être **imbriqués** afin de créer des sous-sections qui reflètent l'organisation. Leur objectif principal est de structurer le contenu de la page et de faciliter la navigation pour les utilisateurs et les moteurs de recherche.

Les listes : les différents niveaux

- **...** : une liste non-ordonnée (non-numérotée)
- **...** : une liste ordonnée
- **...** : un élément d'une liste
- **<dl>...</dl>** : une liste de description
- **<dt>...</dt>** : un terme d'une liste
- **<dd>...</dd>** : la description d'un terme

Exemples de listes

- Liste non-numérotée

```
<ul>
  <li>Barcelone</li>
  <li>Marseille</li>
  <li>Manchester</li>
</ul>
```

- Barcelone
- Marseille
- Manchester

- Liste numérotée

```
<ol>
  <li>Barcelone</li>
  <li>Marseille</li>
  <li>Manchester</li>
</ol>
```

1. Barcelone
2. Marseille
3. Manchester

cours réalisé par Anis Assas et Yessine Zekri

Exemples : Listes numérotées

- Si on veut remplacer les chiffres par des lettres

```
<ol type="A">
  <li>Barcelone</li>
  <li>Marseille</li>
  <li>Manchester</li>
</ol>
```

- A. Barcelone
- B. Marseille
- C. Manchester

- Pour commencer à partir d'un nombre autre que 1

```
<ol start="5">
  <li>Barcelone</li>
  <li>Marseille</li>
  <li>Manchester</li>
</ol>
```

- 5. Barcelone
- 6. Marseille
- 7. Manchester

cours réalisé par Anis Assas et Yessine Zekri

Exemples

- Il est possible d'imbriquer les listes

```
<ul>
  <li>France</li>
  <li>Italie
    <ul>
      <li>Milan</li>
      <li>Turin</li>
    </ul>
  </li>
</ul>
```

- France
- Italie
 - Milan
 - Turin

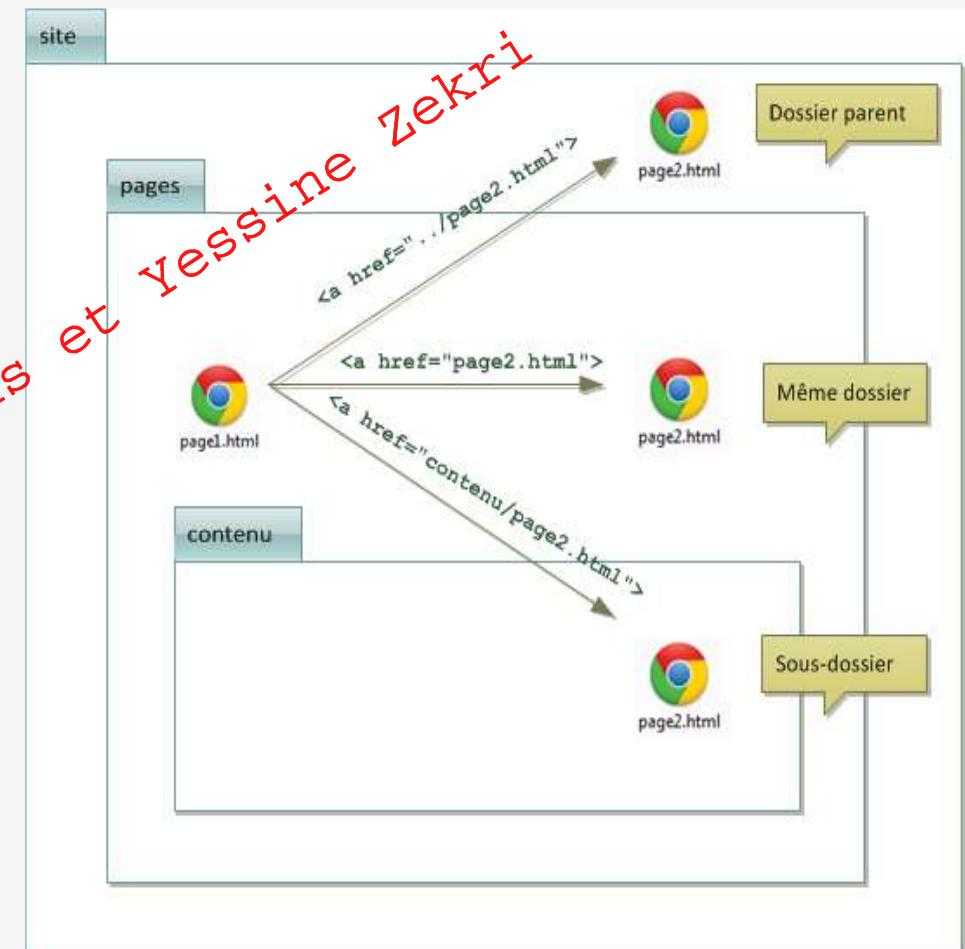
- Exemple d'une liste de description

```
<dl>
  <dt>France</dt>
  <dd>Bleu blanc rouge</dd>
  <dt>Allemagne</dt>
  <dd>Noir jaune rouge</dd>
</dl>
```

France
Bleu blanc rouge
Allemagne
Noir jaune rouge

Les liens

- `<a>...` : insérer un lien (interne ou externe)
- Pour créer un lien vers un autre document local ou distant
 - `monsite`
 - `Page 2`
- **href** est l'attribut de l'élément `<a>` le plus important puisqu'il indique la destination du lien.
- Utiliser un ancre pour pointer vers un signet ('name' ou 'id') dans un même document.
Cours réalisé par Anis Assas et Yessine Zekri



Exemples

- Lien pour télécharger un fichier

```
<a href="https://www.7-zip.org/a/7z1900-x64.exe">Download 7-zip</a>
```

- Le lien sera ouvert dans une nouvelle fenêtre

```
<a href="https://www.unicef.org/fr" target="_blank">UNICEF</a>
```

- Pour définir une base pour tous les liens de la page pour simplifier la création de liens

```
<base href="https://www.unicef.org/fr/">
<a href="agir">Agir & Donner</a>
```

- Créer une ancre dans une même page vers l'id d'un élément

```
<a href="#ancre">Ancre</a>
<p id="ancre">Les ancrées se définissent grâce à l'attribut NAME ou ID</p>
```

Les tableaux

- Les tables constituent un mode privilégié de présentation d'information structurée.
- Une table (balise **<table>**) est divisée en lignes et colonnes
- Une bordure délimite ou non les cellules de la table (Balise **<td>**), organisée en lignes (balise **<tr>**)
- La table préserve en permanence l'aspect visuel de colonnes et dont toutes les cellules ont la même largeur.
- En revanche, les lignes peuvent avoir des hauteurs différentes
- Les différentes balises d'un tableau :
 - **<table>** : tableau
 - **<caption>** : légende du tableau (texte associé)
 - **<th>** : cellule d'en-tête dans un tableau
 - **<tr>** : ligne dans un tableau
 - **<td>** : cellule dans un tableau

Exemple : tableau avec une bordure

- Tableau avec une bordure

```
<table border="1">
  <caption>Moyenne par matière</caption>
  <tr>
    <th>Matiere</th>
    <th>Moyenne</th>
  </tr>
  <tr>
    <td>PHP</td>
    <td>10</td>
  </tr>
  <tr>
    <td>Java</td>
    <td>8</td>
  </tr>
</table>
```

Matiere	Moyenne
PHP	10
Java	8

Exemple : Fusion des cellules

- Fusionner des cellules d'un tableau

```
<table border="2">
  <tr>
    <td rowspan="2">Sites</td>
    <td colspan="2">Performances</td>
  </tr>
  <tr>
    <td>Nombre visiteurs/jour </td>
    <td>PageRank</td>
  </tr>
  <tr>
    <td>www.monsite.com</td>
    <td>3000</td>
    <td>7/10</td>
  </tr>
</table>
```

cours réalisé par Anis Assas et Yessine Zekri

- **rowspan** : fusion verticale de 2 ou plusieurs lignes
- **colspan** : fusion horizontale de 2 ou plusieurs colonnes

Sites	Performances	
www.monsite.com	Nombre visiteurs/jour	PageRank
www.monsite.com	3000	7/10

Exemple : Regroupement des éléments

- Regrouper les éléments d'un tableau

Mois	Loyer
January	600\$
February	700\$
Total	1 300\$

cours réalisé par Anis Assas et Yessine Zekri

```
<table border="1">
  <thead>
    <tr>
      <th>Mois</th>
      <th>Loyer</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>January</td>
      <td>600$</td>
    </tr>
    <tr>
      <td>February</td>
      <td>700$</td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <td>Total</td>
      <td>1 300$</td>
    </tr>
  </tfoot>
</table>
```

Les formulaires : Déclaration

- Déclaration d'un formulaire

```
<form method="POST ou GET" action="page web destination"></form>
```

- Les attributs d'un formulaire

- **method** : concerne l'envoi de données et peut prendre deux valeurs :

- ❖ **GET** : (utilisé par défaut) non fréquemment utilisée car limitée à 255 caractères.

- En plus, les informations envoyées seront visibles dans la zone d'adresse (URL).

- ❖ **POST** : plus utilisée que GET car elle permet d'envoyer un grand nombre d'informations

- et les données saisies dans le formulaire ne transitent pas par la barre d'adresse (URL).

Les formulaires : Déclaration

- **Les attributs d'un formulaire**

- **action** : indique l'adresse (absolue ou relative) de la page ou du programme qui va traiter les informations du formulaire (généralement avec un langage autre que HTML).

```
<form action="traitement.php"></form>
<form action="http://www.monsite.com/traitement.php"></form>
<form action="mailto:yessine.zekri.isetjb@gmail.com"></form>
```

- **name** : indique le nom interne du formulaire
 - **onsubmit** : gère un événement quand l'utilisateur clique sur le bouton d'envoi des données (pour valider les données par exemple).
 - **onreset** : gère un événement quand l'utilisateur efface en bloc toutes les saisies réalisées (pour annuler la saisie des données par exemple).

- **Que peut-on avoir dans un formulaire?**

- Des zones de saisie, des zones de choix, des boutons...

Zones de saisie

- Une zone de saisie mono-ligne

```
<input type="text" name="nom-input">
```

- Une zone de saisie multi-lignes

```
<textarea name="nom-textarea" rows="5"></textarea>
```

- Un libellé associé à une zone de saisie. Généralement on associe la même valeur aux attributs **for**, **id** et **name**

```
<label for="nom">Nom :</label>
<input type="text" name="nom" id="nom" value="John Wick">
```

- Avec HTML 5, les 3 écritures suivantes sont équivalentes :

```
<input type="text" value="John Wick">
<input type="text" value="John Wick">
<input type='text' value='John Wick'>
```

- **email** : pour les adresses emails

```
<input type="email" name="user_email">
```

Zones de saisie

- **password** : pour les mots de passe
`<input type="password" id="pass" name="password">`
- **color** : pour les couleurs
`<input type="color" id="head" name="couleur" value="#00ff00">`
- **number** : pour les nombres (attributs min, max et step)
`<input type="number" name="age" min="10" max="100" step="5" >`
- **date** : pour les calendriers (on pourrait également utiliser le type : month, week, time, datetime-local, ...)
`<input type="date" id="date" value="2023-07-30">`
`<input type="month" id="month" value="2023-12">`
`<input type="week" id="week" value="2023-W01">`
`<input type="time" id="time" value="12:30">`
`<input type="datetime-local" id="time" value="2023-07-30T12:30">`
- **url** : pour les adresses URL
`<input type="url" name="url" placeholder="https://example.com" size="30">`

Zones de saisie

- **pattern** : attribut pour l'utilisation des expressions régulières (regex) pour le filtrage (validité de données)

```
<input type="text" pattern="[a-z]{3,5}">  
<style>  
    input:invalid {  
        border: red solid 3px;  
    }  
</style>
```

Texte : azert

Texte : azerty

Le champ de saisie doit contenir entre
3 et 5 lettres minuscules

- Autres exemples :

```
<input type="text" pattern="^[A-Z][0-9]{3,}">  
<style>  
    input:invalid {  
        border: red solid 3px;  
    }  
</style>
```

le champ de saisie doit commencer
par une lettre majuscule suivie d'au
moins 3 chiffres consécutifs.

Zones de saisie

- Autres exemples :

```
<input type="text" pattern="[a-zA-Z]+[0-9]{4}[A-Z]?">
<style>
    input:invalid {
        border: red solid 3px;
    }
</style>
```

```
<input type="email"
pattern="[a-zA-Z0-9.-_]+@[a-zA-Z]+\.[a-zA-Z]{2,3}">
<style>
    input:invalid {
        border: red solid 3px;
    }
</style>
```

cours réalisé par ANIS ASSAS et YESSINE ZEKRI

Le champ de saisie doit commencer par une ou plusieurs lettres (minuscules ou majuscules), suivies de 4 chiffres et éventuellement d'une lettre majuscule à la fin.

Le champ de saisie doit contenir une adresse e-mail valide, avec un nom d'utilisateur pouvant inclure des lettres, des chiffres, des points (.), des tirets (-) et des traits de soulignement (_), suivi du symbole "@" et d'un nom de domaine contenant des lettres, suivi d'un point (.) et d'une extension de domaine de 2 à 3 lettres.

Zones de saisie

- Autres exemples :

```
<input type="url" pattern="http://www\...+\...+"/> cours réalisé par Anis Assas
<style>
    input:invalid {
        border: red solid 3px;
    }
</style>
```

Yessine Zekri

Le champ de saisie doit contenir une URL commençant par "<http://www.>", suivie de n'importe quel contenu avant un autre point(.) et une extension de domaine.

(? \leftrightarrow {,1}) 0 ou 1 répétition
(* \leftrightarrow {0,}) 0 ou plusieurs répétitions
(+ \leftrightarrow {1,}) 1 ou plusieurs répétitions
(. \leftrightarrow absolument n'importe quel caractère)