

The HyperText Markup Language



Jean-Claude AZIAHA



Objectif de la formation

Réaliser correctement une mise en page web
avec le langage HTML

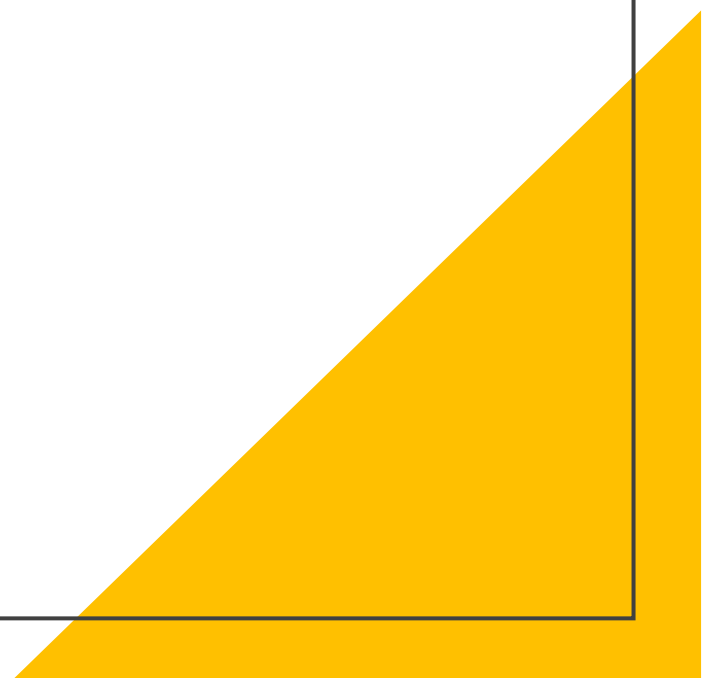
Programme

PARTIE A : LE CONCEPT DE SITE WEB

1. Qu'est-ce qu'un site web ?
2. Qu'est-ce qu'une page web ?
3. Comment visualiser une page web ?

Programme

PARTIE B : LE LANGAGE HTML

1. Qu'est-ce que le HTML ?
 2. Quels sont nos outils de développeur ?
 3. Comment structurer un document HTML ?
 4. Les liens
 5. Les images
 6. Les formulaires
 7. Les tableaux HTML5
 8. Les audios et vidéos
 9. L'arborescence d'un site web
- 
- A large yellow triangle is positioned in the bottom right corner of the slide, pointing towards the top right.

Programme

PARTIE C : EVALUATION ET CONCLUSION

1. Mini Projet
2. Qcm
3. Conclusion

PARTIE A

LE CONCEPT DE SITE WEB



A1- QU'EST-CE QU'UN SITE WEB ?

- ❖ Un **site web**, de manière standard, est un ensemble de pages web reliées entre elles.
- ❖ Cependant, il existe aussi des sites web qui sont constitués d'une seule page.
Exemples : One Page, Landing Page, Single Page Application (SPA).
- ❖ En résumé, un site web est une ressource en ligne constituée d'au moins une page web.

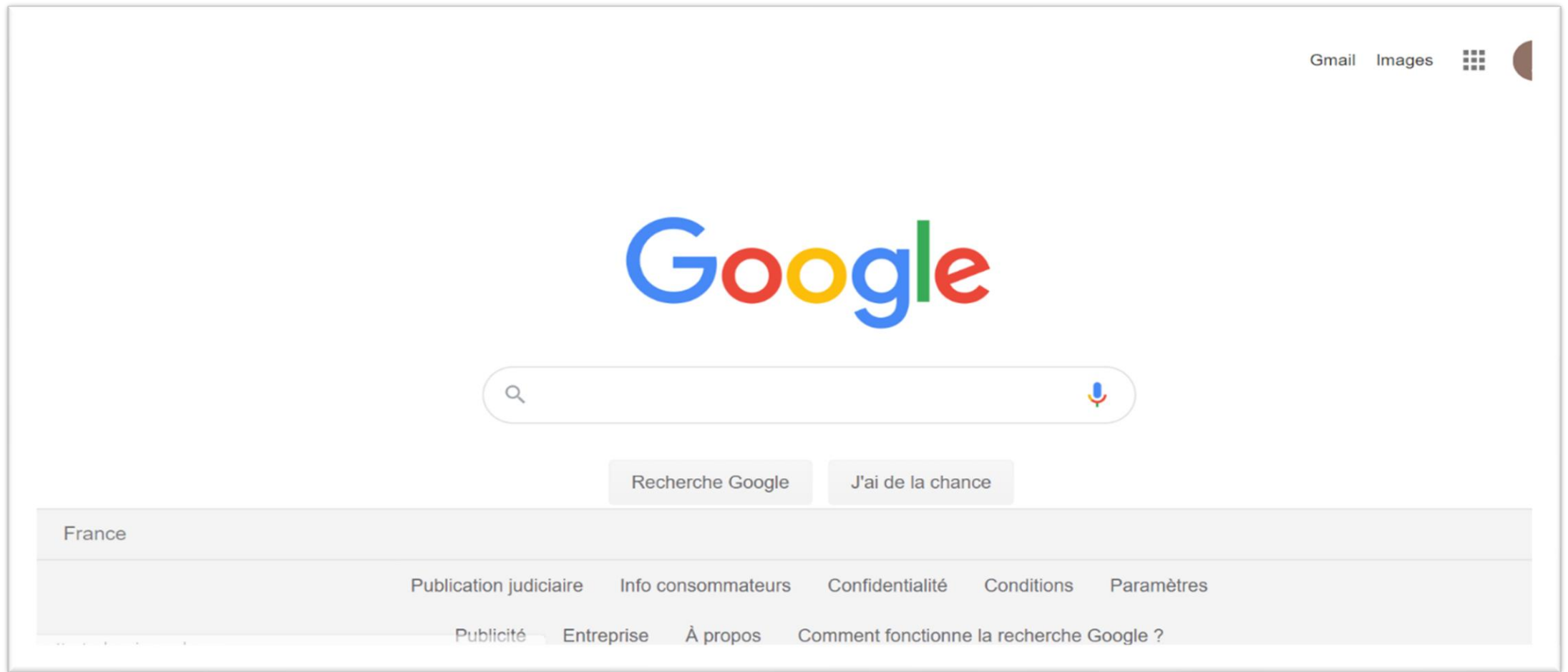
A2- QU'EST-CE DONC QU'UNE PAGE WEB ?

- ❖ Une **page web** est un document numérique conçu pour afficher ou transmettre des informations sous forme de données.
- ❖ Elle est principalement composée de **texte** et de **liens hypertextes**, qui permettent d'accéder à d'autres ressources disponibles sur le web.
- ❖ En complément, on peut également y intégrer des **images**, des **audios** et des **vidéos**, afin d'enrichir son contenu.

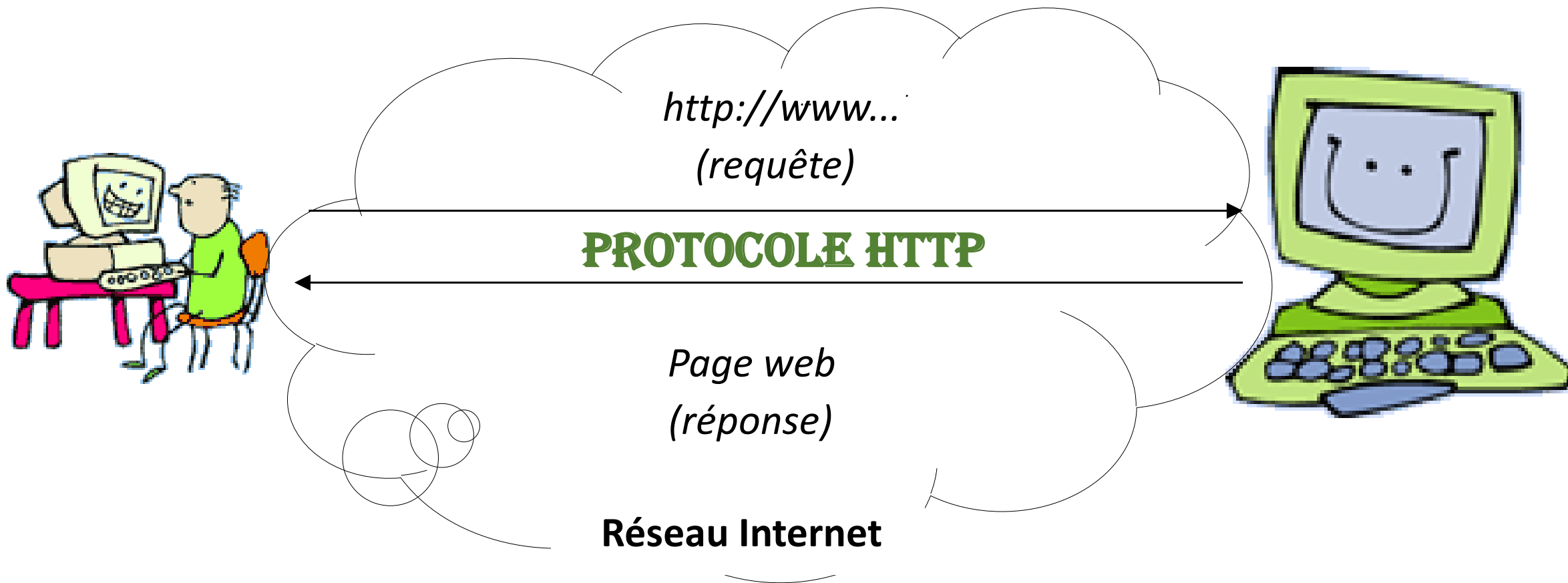
A3- COMMENT VISUALISER UNE PAGE WEB?

- ❖ Pour consulter un site web, nous avons besoin d'un **navigateur** (Firefox, Chrome, Safari...).
- ❖ Nous "ouvrons" la page du site en entrant son **nom de domaine** dans la barre d'URL du navigateur.
- ❖ **Exemple** : www.google.com



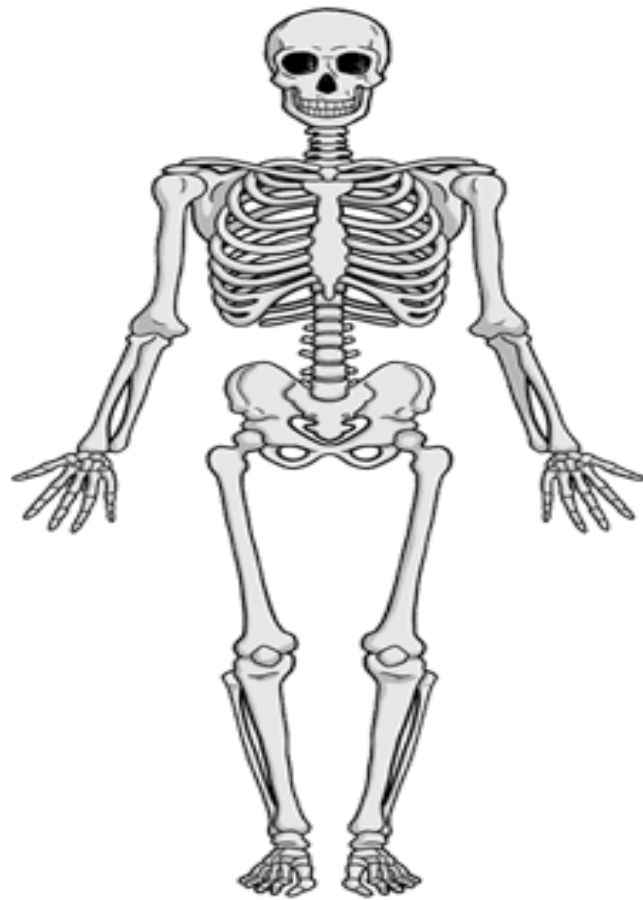


Quel processus se déclenche lorsque nous appuyons sur “Entrée” pour afficher la page d’accueil de Google ?



- ❖ Sur notre fameuse page d'accueil de Google, nous pouvons observer plusieurs éléments : du **texte coloré**, une **barre de recherche**, des **liens** qui permettent d'accéder à d'autres pages, etc.
- ❖ Mais comment a-t-on réussi à créer tout cela ?
Deux langages essentiels : **HTML** et **CSS**.
- ❖ **HTML** sert à structurer et ajouter le contenu de la page.
- ❖ **CSS** permet de styliser ce contenu en ajoutant des couleurs, positionnant les éléments, et personnalisant l'apparence selon nos besoins et préférences.

HTML



STRUCTURE

HTML + CSS



PRESENTATION

Vs.

Agence de voyage

Envie de partir en vacances?

Voici nos destinations:

- Bali
- Bora Bora
- Dubai
- Lomé
- Tunis

Contactez-nous

email: agende-de-voyage@gmail.com

Téléphone: 06 55 55 55 55

HTML+CSS

Agence de voyage

Envie de partir en vacances?

Voici nos destinations:

- A. Bali
- B. Bora Bora
- C. Dubai
- D. Lomé
- E. Tunis

Contactez-nous

email: agence-de-voyage@gmail.com

Téléphone: 06 55 55 55 55

email: agence-de-voyage@gmail.com

Téléphone: 06 22 22 22 22

HTML



PARTIE B

LE LANGUAGE HTML

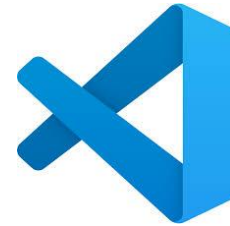
B1- QU'EST-CE QUE LE HTML ?

- ❖ **HTML** est l'acronyme de **H**yper**T**ext **M**arkup **L**anguage.
- ❖ C'est un langage utilisé pour ajouter du contenu sur une page web. Ce contenu peut inclure : du texte, des images, des audios, des vidéos, etc.
- ❖ Le code HTML est écrit à l'aide de balises.
 - Ces balises entourent les éléments avec les symboles < et >.
 - Exemple : <p>Hello World!</p>
- ❖ En plus d'ajouter du contenu, le HTML permet de structurer les éléments de la page afin de garantir une disposition claire et organisée.

B2- NOS OUTILS DE DÉVELOPPEMENT

- **Un éditeur de code**: Visual Studio Code

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



- **Un navigateur**: Chrome ou Safari...

<https://www.google.com/chrome>



B3-COMMENT STRUCTURER UN DOCUMENT HTML ?

a) Structure globale d'une page (Squelette)

```
<> cours.html > ...  
1  <!DOCTYPE html>  
2  <html lang="en">  
3      <head>  
4  
5      </head>  
6      <body>  
7  
8      </body>  
9  </html>
```

10

10

B3- COMMENT STRUCTURER UN DOCUMENT HTML ?

b) Principales balises et attributs HTML5

<!DOCTYPE html> : Indique que le document est de type HTML5. C'est une déclaration obligatoire pour tous les fichiers HTML.

<html></html> : La balise principale qui englobe tout le code HTML. C'est ici que l'ensemble des éléments HTML de la page sera contenu.

<head></head> : La section d'en-tête qui contient les informations permettant de configurer la page comme les métadonnées, et les fichiers CSS ou JavaScript liés et aussi d'afficher le titre de la page.

<body></body> : La section contenant toutes les données visibles sur la page

B3- COMMENT STRUCTURER UN DOCUMENT HTML ?

Il existe deux types d'éléments en HTML :

1. Les éléments standards
2. Les éléments vides

1. Les éléments standards

Les éléments standards sont les plus courants. Ils sont composés de deux balises

- ✓ Une balise ouvrante
- ✓ Une balise fermante

Exemples :

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

B3- COMMENT STRUCTURER UN DOCUMENT HTML ?

2. Les éléments vides

Les éléments vides ne contiennent pas de contenu. Contrairement aux éléments standards, ils ne nécessitent pas de balise fermante. Ces éléments servent généralement à inclure des ressources ou ajouter des espaces.

- ✓ Une balise auto fermante.

Exemples d'éléments vides :

<**img** src="image.jpg" alt="Description de l'image">

<**br**>

<**hr**>

<**meta** charset="UTF-8">

<**link** rel="stylesheet" href="styles.css">

B3- COMMENT STRUCTURER UN DOCUMENT HTML ?

<élément attribut1="valeur" attribut2="valeur"...> contenu </élément>

<élément attribut1="valeur" attribut2="valeur"... />

Remarquons que les attributs sont toujours placés dans la **balise ouvrante** de l'élément **HTML**.

B3- COMMENT STRUCTURER UN DOCUMENT HTML ?

c) structure du head

Le **head** contient des éléments qui donnent des informations sur le document (meta, title, link, style, script).

Exemple :

```
3 <head>
4   <meta charset="utf-8" />
5   <title> Un exemple </title>
6   <link rel="stylesheet" href="style.css"/>
7 </head>
```

B3- COMMENT STRUCTURER UN DOCUMENT HTML ?

c) Structure du head

meta: L'élément de metadonnées du document.

Il possède plusieurs attributs.

`<meta charset='utf-8' />` Pour définir l'encodage de la page afin que les caractères spéciaux soient bien pris en compte.

`<meta name='description' content='Site sympa...' />`

D'autres balises que nous verrons plus tard

title: Cet élément affiche le titre de votre page dans l'onglet.

link: Permet de relier un fichier CSS au fichier HTML

B3- COMMENT STRUCTURER UN DOCUMENT HTML ?

d) Structure du body

Le body est composé des éléments utiles pour bien structurer le document en grandes divisions/sections

body : la racine des divisions

header : introduction du document entier (en-tête de page), d'une section ou d'un article...

nav : division pour les liens de navigation (le menu)

main : représente le contenu majoritaire du body

section : division générique regroupant un même sujet

B3- COMMENT STRUCTURER UN DOCUMENT HTML ?

d) Structure body

article : division de contenu indépendant

aside : division dont le contenu est un complément par rapport à ce qui l'entoure

footer : conclusion du document entier (pied de page), d'une section ou d'un article...

div : joue le rôle de tous les éléments structurant du body

B3- COMMENT STRUCTURER UN DOCUMENT HTML ?

e) Les commentaires

Un commentaire HTML à 2 principales fonctions :

- Décrire le code pour le rendre plus compréhensible
- Empêcher le code d'être pris en compte par le navigateur

Voici la syntaxe : `<!-- code ou texte à commenter -->`

B3- COMMENT STRUCTURER UN DOCUMENT HTML ?

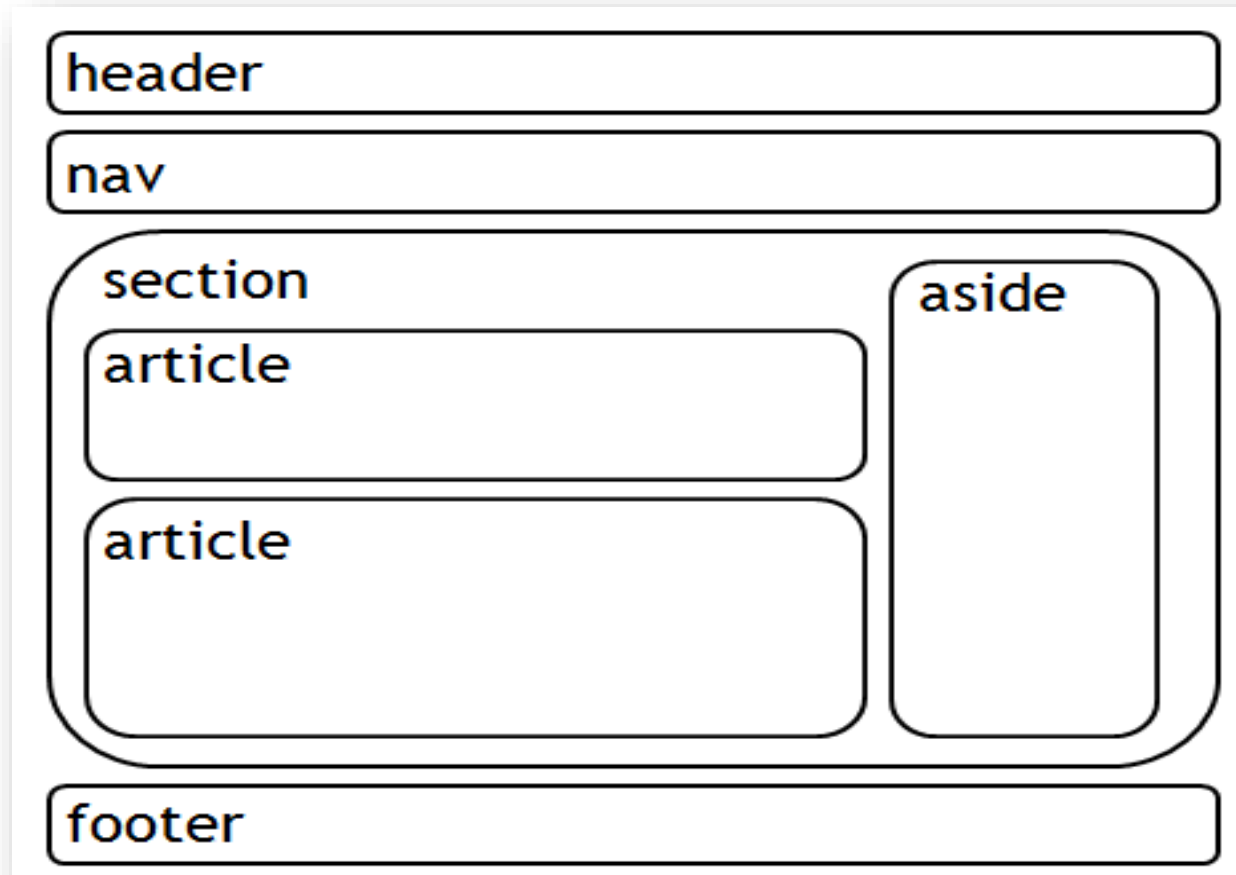
PRATIQUONS!

Exercice 1:

- 1) Créer la structure standard d'une page html
 - header, nav, main, section, article, footer, div
- 2) Y rajouter des éléments structurants
- 3) Ajouter des commentaires pour décrire le code

B3- COMMENT STRUCTURER UN DOCUMENT HTML ?

Exercice 2: Réaliser la mise en page correspondante à cette maquette



B3- COMMENT STRUCTURER UN DOCUMENT HTML ?

f) Les principales balises de contenu

Les titres

h1, h2, h3, h4, h5, h6

`<h1>`Ceci est un titre H1`</h1>`
`<h2>`Ceci est un titre H2`</h2>`
`<h3>`Ceci est un titre H3`</h3>`
`<h4>`Ceci est un titre H4`</h4>`
`<h5>`Ceci est un titre H5`</h5>`
`<h6>`Ceci est un titre H6`</h6>`

Ceci est un titre H1

Ceci est un titre H2

Ceci est un titre H3

Ceci est un titre H4

Ceci est un titre H5

Ceci est un titre H6

B3- COMMENT STRUCTURER UN DOCUMENT HTML ?

f) Les principales balises de contenu

Paragraphe, retour à la ligne, ligne horizontale

`<p>`Hello ! Ceci est mon premier paragraphe`</p>`

``Ceci permet aussi d'écrire du texte``

`
` : pour faire un retour à la ligne

`<hr>` : pour tracer une ligne horizontale

B3- COMMENT STRUCTURER UN DOCUMENT HTML ?

f) Les principales balises de contenu

Emphase, emphase forte

Pour mettre en exergue un morceau de texte, il existe deux possibilités :

- L'emphase (balise)
- L'emphase forte (balise)

Exemple : **Lorem ipsum** : *Lorem ipsum*

<p>

Lorem ipsum :

Lorem ipsum

</p>

B3- COMMENT STRUCTURER UN DOCUMENT HTML ?

<div></div> : Élément de type bloc utilisé pour regrouper des éléments.

- **Caractéristique** : Il occupe toute la largeur de son élément parent.
- **Exemple** :

```
<div>
```

```
<p>Les galaxies spirales sont fascinantes !</p>
```

```
<p>Elles contiennent des milliards d'étoiles et une structure majestueuse.</p>
```

```
</div>
```

**** : Élément de type inline.

- **Caractéristique** : Il occupe uniquement l'espace nécessaire à son contenu et s'adapte en fonction de celui-ci.
- **Exemple** :

```
<span>Les aurores boréales sont des phénomènes naturels spectaculaires.</span>
```

B3- COMMENT STRUCTURER UN DOCUMENT HTML ?

f) Les principales balises de contenu

Les listes non-ordonnées, ordonnées et les listes de description

* Les listes non-ordonnées

 chien

 chat

 poisson

 poule

 oiseau

• chien

• chat

• poisson

• poule

• oiseau

B3- COMMENT STRUCTURER UN DOCUMENT HTML ?

f) Les principales balises de contenu

Les listes non-ordonnées, ordonnées et les listes de description

* Les listes ordonnées

 chien

 chat

 poisson

 poule

 oiseau

1. chien

2. chat

3. poisson

4. poule

5. oiseau

B3- COMMENT STRUCTURER UN DOCUMENT HTML ?

f) Les principales balises de contenu

Les listes non-ordonnées, ordonnées et les listes de description

* Les listes de description

<dl>

<dt>AP1</dt>

<dd>Algorithmique</dd>

<dt>AP5</dt>

<dd>SGBD</dd>

<dt>ASR1</dt>

<dd>SE</dd>

</dl>

AP1

Initiation à l'algorithmique

AP5

intégration des SGBD dans les environnements Web

ASR1

Comprendre et utiliser un système d'exploitation et un réseau

B4- LES LIENS

a) Créer un lien

Pour créer un lien, il nous faut utiliser l'élément html a.

`<a>`

Un lien créé va toujours de pair avec son attribut **href**.

Ceci parce que c'est cet attribut qui permet de définir la destination du lien.

Exemple :

`Cliquez moi`

B4- LES LIENS

b) Différents types de liens

Il existe 3 types de liens

Les liens externes: En cliquant sur ce type de lien, l'on est automatiquement redirigé vers une page d'un autre site web.

Exemple :

`Lien externe vers le site de wikipedia`

B4- LES LIENS

b) Différents types de liens

Les liens internes: Il en existe 2 types

*** Les liens internes locaux (permettent d'accéder à une autre page du même site)**

Exemple :

`Accéder à la page 4 de mon propre site`

B4- LES LIENS

b) Différents types de liens

* Les liens internes ancrés (permettent d'accéder à un élément html présent sur la même page)

Cette dernière est un peu particulière car il faut donner un identifiant à l'élément qu'on veut cibler puis mettre le code permettant de rejoindre directement l'élément.

Exemple :

```
<a href= "#ok" >Accédez directement au paragraphe présent plus bas</a>  
<br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/>  
<p id="ok">Hello World !</p>
```

B5- LES IMAGES

a) Insérer une image

Pour insérer une image, il faut utiliser l'élément `img`.

Puis son attribut `src=""` afin de spécifier la source de l'image.

Suivi ensuite de son second attribut `alt=""`. Alt est également très important parce qu'il représente ce qui devra être affiché au cas où l'image ne se serait pas chargée pour x raison.

Exemple : ``

Il est également possible de redéfinir la taille d'une image dans le code HTML avec les attributs `height` et `width`

Exemple : ``

B5- LES IMAGES

a) Insérer une image

Les principaux formats

- **JPEG** : Ce format est bien adapté aux photos (image comportant beaucoup de couleur).
- **PNG** : Plus récent, libre de droit, compression sans détérioration de la qualité
- **GIF** : Ancien format limité à 256 couleurs mais qui possède la capacité d'animation.
- **SVG** : compression sans détérioration et bien adapté aux photos de profil et logo par exemple.
- **WEBP** : Format moderne développé par Google, offrant une meilleure compression que JPEG et PNG tout en supportant la transparence et l'animation.
- **AVIF** : Format basé sur la technologie AV1, offrant une excellente compression et qualité d'image. Supporte la transparence.

B5- LES IMAGES

b) Définir une image comme lien

Principe : La balise doit être incluse à l'intérieur de la balise <a>

Exemple:

```
<a href="profil.html">  
      
</a>
```


B6- LES FORMULAIRES

a) les principaux champs

Un formulaire représente un élément fondamental en HTML, car il permet d'envoyer des informations à un serveur.

Les formulaires sont souvent utilisés pour collecter des données auprès des utilisateurs, comme le nom, prénom, e-mail, mot de passe...

L'élément HTML pour créer un formulaire est : **<form></form>**

B6- LES FORMULAIRES

```
body>
  <form action="traitement.php" method="post">
    <div>
      <label for="firstName">Prénom</label>
      <input type="text" name="firstName" id="firstName">
    </div>
    <div>
      <label for="lastName">Nom</label>
      <input type="text" name="lastName" id="lastName">
    </div>
    <div>
      <label for="age">Age</label>
      <input type="number" name="age" id="age">
    </div>
    <div>
      <input type="submit">
    </div>
  </form>
</body>
```

```
<poql>
<\toiw>
```

B6- LES FORMULAIRES

Expliquons un peu tout ça

<label> : Définit l'étiquette d'un champ de formulaire.

Attribut **for** : Relie le label au champ via son id.

<input> : Crée un champ interactif dans un formulaire.

Attributs principaux :

type : Définit le type de champ (texte, nombre, etc.).

name : Identifie le champ pour transmettre ses données au serveur.

B6- LES FORMULAIRES

Les principaux types d'input

`<input type="text" />` Pour créer un champ de chaînes de caractères

`<input type="number" />` Pour créer un champ de nombre entier

`<input type="email" />` Pour créer un champ de type email

`<input type="tel" />` Pour créer un champ de type numéro de téléphone

`<input type="password" />` Pour créer un champ avec sa valeur cachée visuellement. Les caractères sont remplacés par des étoiles.

B6- LES FORMULAIRES

Les principaux types d'input

`<input type="date" />` Pour créer un champ de type date

`<input type="time" />` Pour créer un champ de type temps

`<input type="datetime-local" />` Pour créer un champ de date et temps

`<input type="month" />` Pour créer un champ de type mois

`<input type="week" />` Pour créer un champ de type semaine

B6- LES FORMULAIRES

`<input type="radio" />` Pour créer un bouton. Généralement, nous en créons au moins 2 pour donner la possibilité à l'utilisateur de n'en choisir qu'un.

`<input type="checkbox" />` Pour créer une case à cocher. Généralement, nous l'utilisons pour donner la possibilité à l'utilisateur de choisir plusieurs options par exemple.

`<input type="hidden" />` Pour créer un champ invisible

`<input type="search" />` Pour créer un champ de barre de recherche

B6- LES FORMULAIRES

`<input type="file" />` Pour permettre de charger un fichier depuis son ordinateur.

`<input type="url" />` Pour créer un champ qui accepte une url

`<input type="range" />` Pour créer un champ de d'encadrement de valeurs

`<input type="submit" />` Pour créer le bouton d'envoi du formulaire

`<input type="reset" />` Pour réinitialiser les données du formulaire

B6- LES FORMULAIRES

Les autres éléments d'un formulaire

Dans un formulaire, il n'y a pas que les inputs et les labels.

`<textarea></textarea>` : Permet de définir un gros champs de type text.

`<select></select>` : Pour créer une liste d'options déroulantes

`<optgroup></optgroup>` : inclus dans le select, il permet d'ordonner notre liste en regroupant les options

`<option></option>` : inclus dans le optgroup du select, il permet de choisir un élément de notre liste.

`<fieldset></fieldset>` : Permet également de structurer la sémantique de notre formulaire.

`<legend></legend>` : inclus dans le fieldset, grâce à cet élément, on va pouvoir donner un mini titre.

B6- LES FORMULAIRES

b) les autres attributs

`placeholder=""` : Cet attribut peut être appliqué aux inputs. La valeur qui lui sera attribuée s'affichera dans le champ. Ce champ est surtout utilisé à titre indicatif.

`value=""` : Cet attribut permet de donner une valeur propre à l'élément du formulaire.

`id=""` : Permet de cibler l'élément du formulaire en lui attribuant un identifiant.

B6- LES FORMULAIRES

b) les autres attributs

`class=""` : Permet de cibler l'élément du formulaire en lui attribuant une classe.

`min-length=""` : Pour attribuer une longueur minimale de caractères.

`max-length=""` : Pour attribuer une longueur maximale de caractères.

`min=""` : Pour attribuer la valeur numérique minimale.

`max=""` : Pour attribuer la valeur numérique maximale

B6- LES FORMULAIRES

b) Les autres attributs

`readonly=""` Pour indiquer que la valeur d'un input n'est pas modifiable.

`pattern=""` : Pour mettre en place une règle de validation d'un input

`autocomplete=""` : Pour activer le remplissage automatique d'un input

`disabled=""` : Permet d'indiquer si le contrôle est désactivé

B6- LES FORMULAIRES

c) Sécuriser un champ de formulaire

Il est possible de sécuriser un champ de formulaire en lui donnant l'attribut **required**.

Cependant, il faudrait savoir que cette barrière ne sera pas suffisante pour affirmer qu'un formulaire a été sécurisé correctement.

Des sécurités supplémentaires devront être faites avec du **JAVASCRIPT** et surtout avec un langage coté serveur comme du **PHP** par exemple.

B6- LES FORMULAIRES

Informations principales

Algérie ▼

Nom :

Prenom :

Age :

Email :

Hobbies

Qu'aimez-vous faire pour passer le temps ?

- ☐ Sport
- ☐ Lecture
- ☐ Voyage

Valider

Nous allons nous amuser en écrivant le code HTML permettant d'avoir le même rendu sur une page web.

PRATIQUONS !

B6- LES FORMULAIRES

d) Les méthodes 'get' et 'post'

Un formulaire possède principalement comme attributs: `method` et `action`.

`method=""` peut avoir la valeur **get** ou **post**.

get veut tout simplement dire que toutes les informations envoyées par le formulaire seront visibles dans la barre d'url.

post aura la même fonction à la différence qu'il cachera toutes les informations envoyées par le formulaire.

`action=""` pour indiquer la page sur laquelle les informations devront être envoyées pour être traitées.

B7- LES TABLEAUX

Les tableaux HTML sont utilisés pour structurer des données.

`<table></table>` (caption, tr, th, td, thead, tbody, tfoot)

```
8  <body>
9  <table>
10 <thead>
11 <tr>
12 <th>Nom</th>
13 <th>Prénom</th>
14 <th>Age</th>
15 </tr>
16 </thead>
17
18 <tbody>
19 <tr>
20 <td>Dupont</td>
21 <td>Jean</td>
22 <td>33 ans</td>
23 </tr>
24 </tbody>
25 </table>
26 </body>
```

```
30 <\poql>
32 <\tspje>
```

B7- LES TABLEAUX

Il est possible d'ajouter plusieurs lignes (tr) dans le tbody afin d'insérer plus de valeur.

Il est tout à fait possible de fusionner des cellules du tableau entre elles que cela soit en longueur ou en largeur.

Pour cela il faut utiliser les attributs `rowspan` et les `colspan`

`rowspan` pour définir le nombre de lignes que prendra un `<td>` ou `<th>` dans un `<tr>`

`colspan` pour définir le nombre de colonnes que prendra un `<td>` ou `<th>` dans un `<tr>`

B7- LES TABLEAUX

Attaquants		Milieux de terrain		Défenseurs		Gardiens
Pointes	Ailliers	M. offensifs	M. défensifs	Def. centraux	def. latéraux	---
Olivier Giroud	Kylian Mbappe	Antoine Griezmann	N'Golo Kanté	Rafael Varane	Benjamin Pavard	Hugo Lloris
Ben Yedder	Ousmane Dembélé	Nabil Fékir	Paul Pogba	Smuel Umtiti	Lucas Hernandez	Steve Mandanda

Nous allons nous exercer en réécrivant le code correspondant au tableau ci-dessus.

PRATIQUONS !

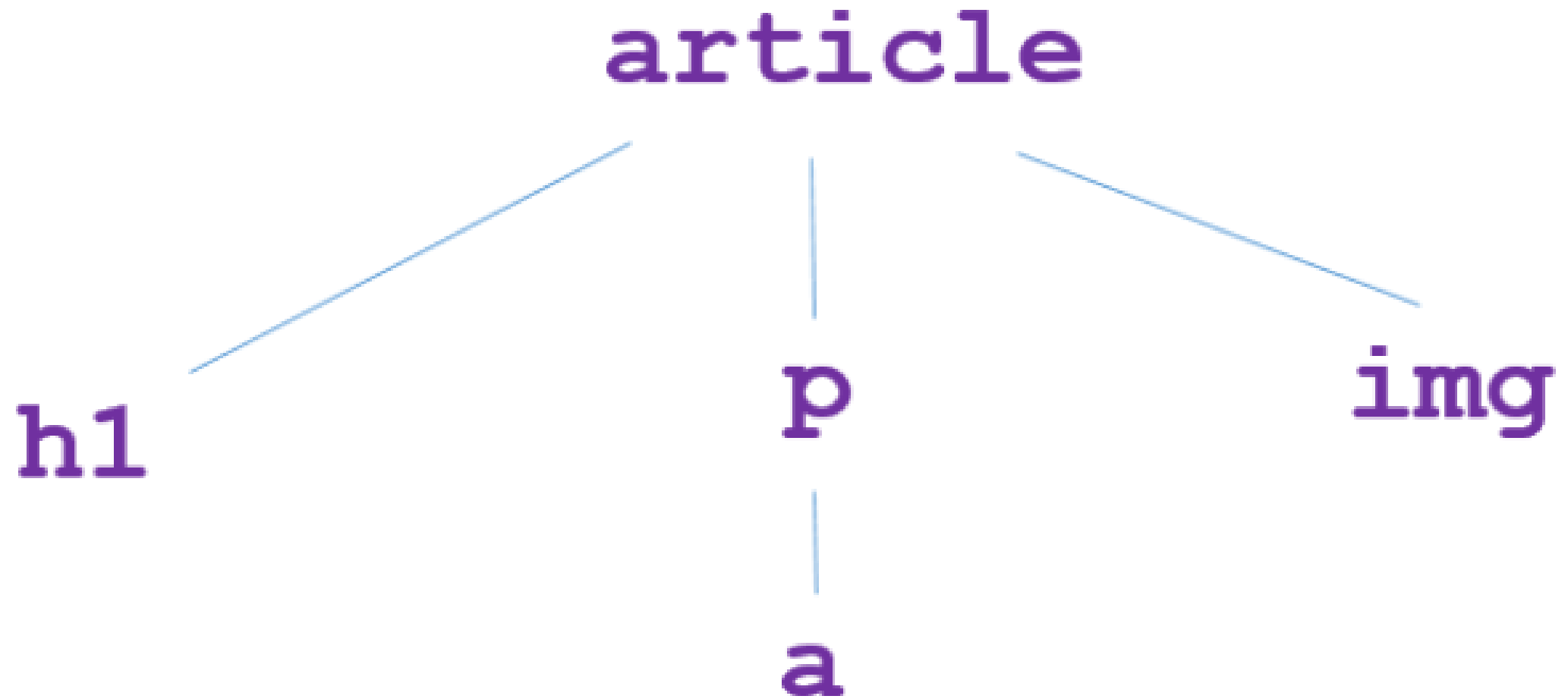
B8- LES AUDIOS ET VIDÉOS

Attention: les navigateurs ne supportent pas tous les mêmes formats audio (Wav, Ogg) et/ou vidéo (MP4, WebM, Ogg)

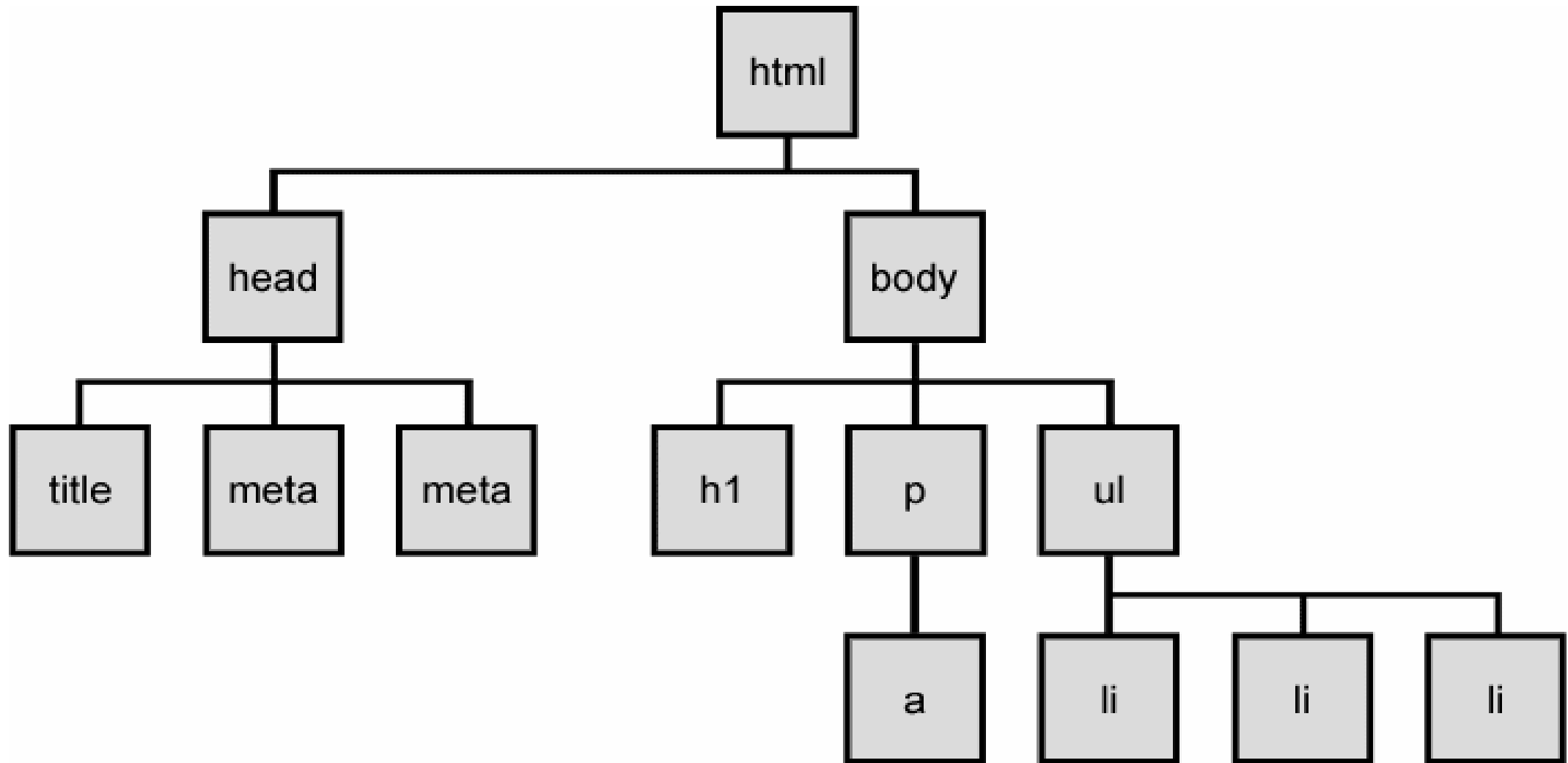
```
<audio autoplay controls loop muted >  
  <source src="maChanson.mp3" type="audio/mp3" />  
  <source src="maChanson.ogg" type="audio/ogg" />  
</audio>
```

```
<video width="400" height="222" controls="controls" >  
  <source src="maVideo.mp4" type="video/mp4" />  
  <source src="maVideo.ogg" type="video/ogg" />  
</video>
```

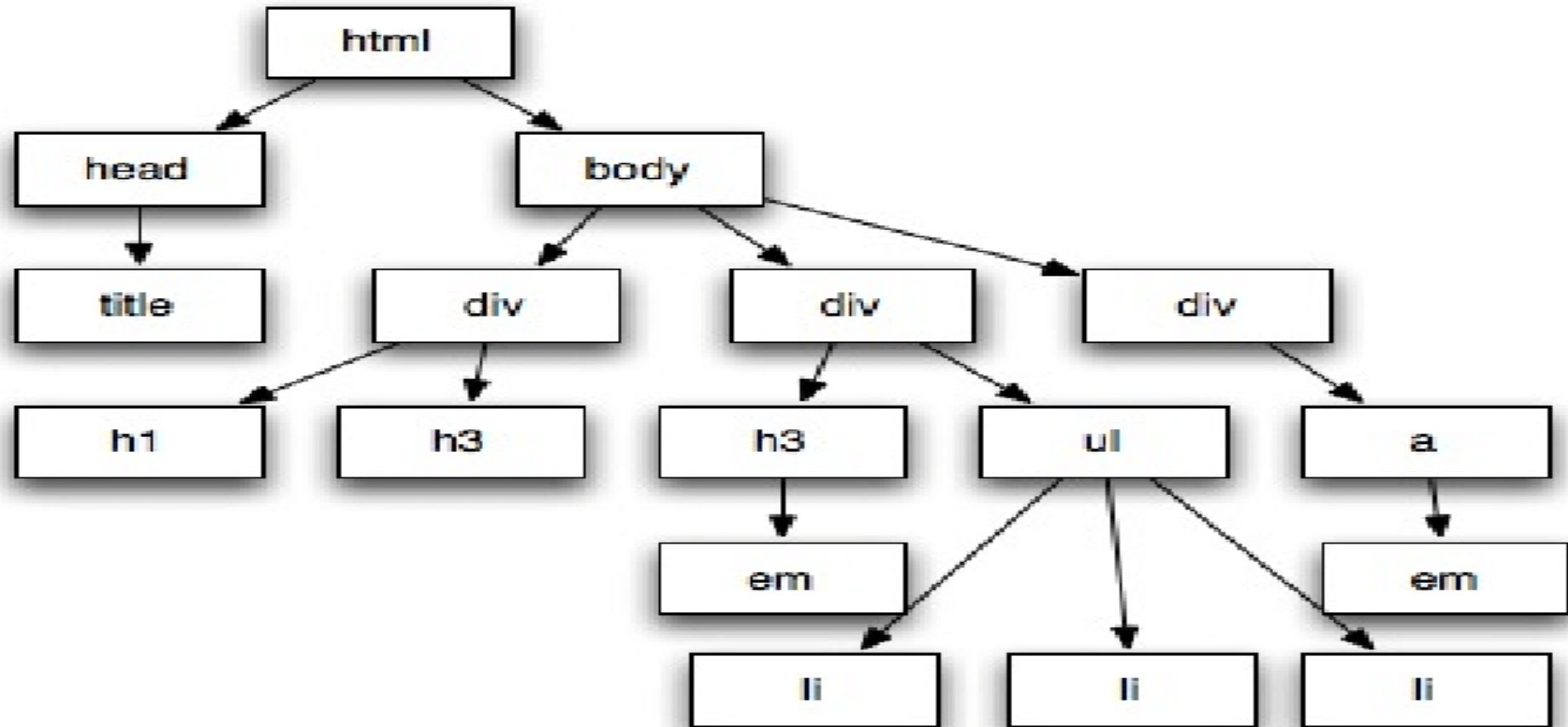
B9- ARBORESCENCE



B9- ARBORESCENCE



B9- ARBORESCENCE



PARTIE C

EVALUATION ET CONCLUSION

C1- MINI-PROJET

Afin de s'assurer que les notions ont bien été assimilées, nous allons créer un simple CV en HTML en nous basant sur une maquette.

C2- QCM

En évaluation, vous devrez répondre aux questions posées dans le document qcm-html

The background of the slide is a complex, abstract graphic. It features a network of white lines connecting various points, some of which are highlighted with blue and white circular nodes. Overlaid on this network are several geometric shapes, including triangles and polygons, in shades of blue and purple. Binary code (0s and 1s) is scattered throughout the design, appearing in different orientations and colors (white, blue, and purple).

C3- CONCLUSION

Au final, les éléments du langage HTML5 servent principalement à :

- ❑ **Structurer du contenu** : divisions, titres, paragraphes, tableaux, listes...
- ❑ **Inclure des objets externes** : images, sons, vidéos
- ❑ **Réaliser des liens hypertextes**