

Premiers pas avec HTML : construire la structure de base d'une page web



Premiers pas avec HTML

1. Introduction au développement web

- a. Quel est le rôle d'un développeur front-end ?
- b. Les technologies utilisées en développement front-end
- c. Qu'est-ce que le HTML et quel est son rôle dans le web ?

2. Les balises HTML : anatomie et utilisation

- a. La balise normale
- b. La balise auto-fermante

3. Les attributs HTML

4. L'indentation

5. Structure de base d'une page HTML5

6. Les balises essentielles : titres, paragraphes, listes, div & span, sémantiques, liens, images

7. Structurer une page avec les balises sémantiques (header, main, section...)

8. Les tableaux HTML

9. Exercice

Le développement web : kezaiko ?

Le développement web fait référence au processus d'écriture d'un site ou d'une page web dans un langage technique.

👉 On distingue deux catégories de développement :

LE DÉVELOPPEMENT FRONT-END

(front-office / côté client)

- Intégrateur
- Creative coder
- Développeur front-end

LE DÉVELOPPEMENT BACK-END

(back-office / côté serveur)

- Développeur back-end
- Développeur fullstack

Quel est le rôle du développeur front-end ?

La mission d'un développeur front-end consiste à participer à la création de l'interface utilisateur d'un site ou d'une application web.

1. Il intègre des maquettes graphiques sous forme de lignes de codes grâce aux langages HTML, CSS et JavaScript.
2. Il réalise des animations et des effets pour rendre n'importe quel site "époustouflant" 😎.
3. Il a le sens du détail et un bon œil pour créer une interface graphique au pixel près.
4. Il connaît les standards W3C, les règles d'accessibilité et les bonnes pratiques SEO.

Les technologies utilisées en front-end

Le développement front-end se concentre sur 3 principaux langages



HTML(5)



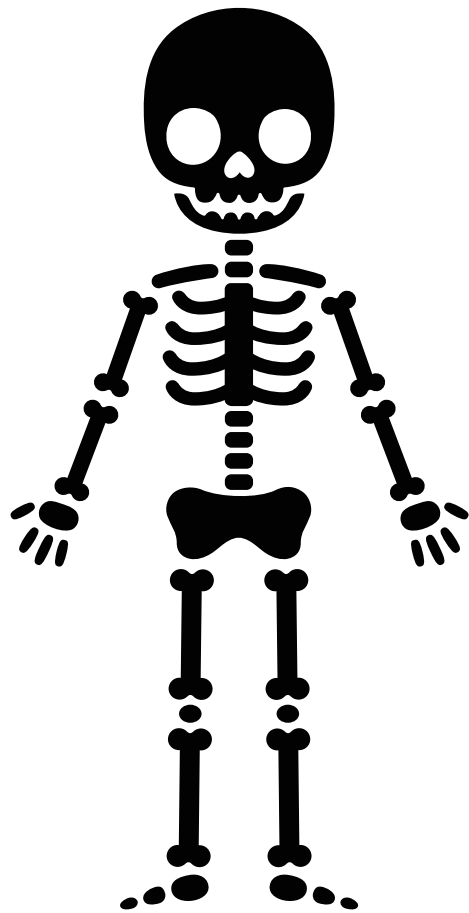
CSS(3)



Javascript



Les technologies utilisées en front-end



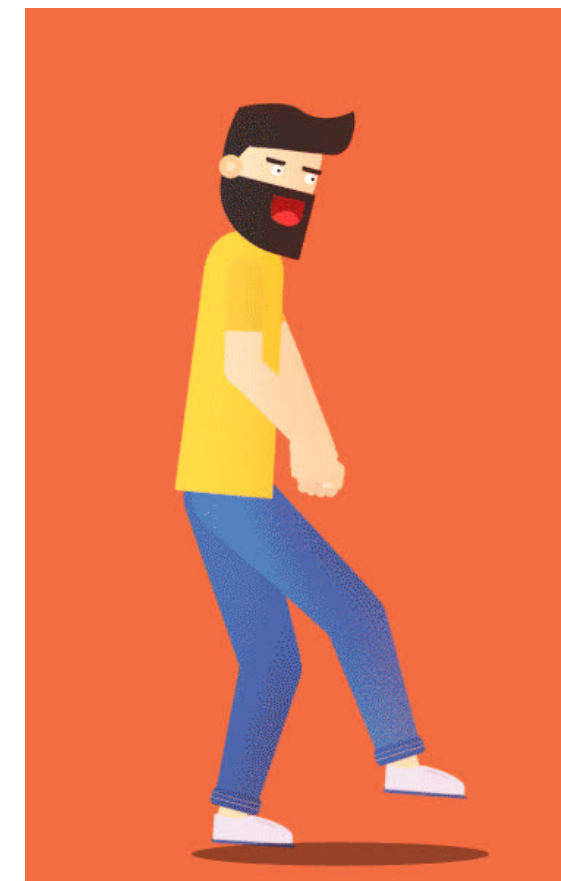
HTML

Le HTML est la structure
squelettique d'une page



HTML + CSS

Le CSS l'embellit



HTML + CSS + JAVASCRIPT

Le JavaScript la rend
dynamique

Les technologies utilisées en front-end

Acheter

HTML

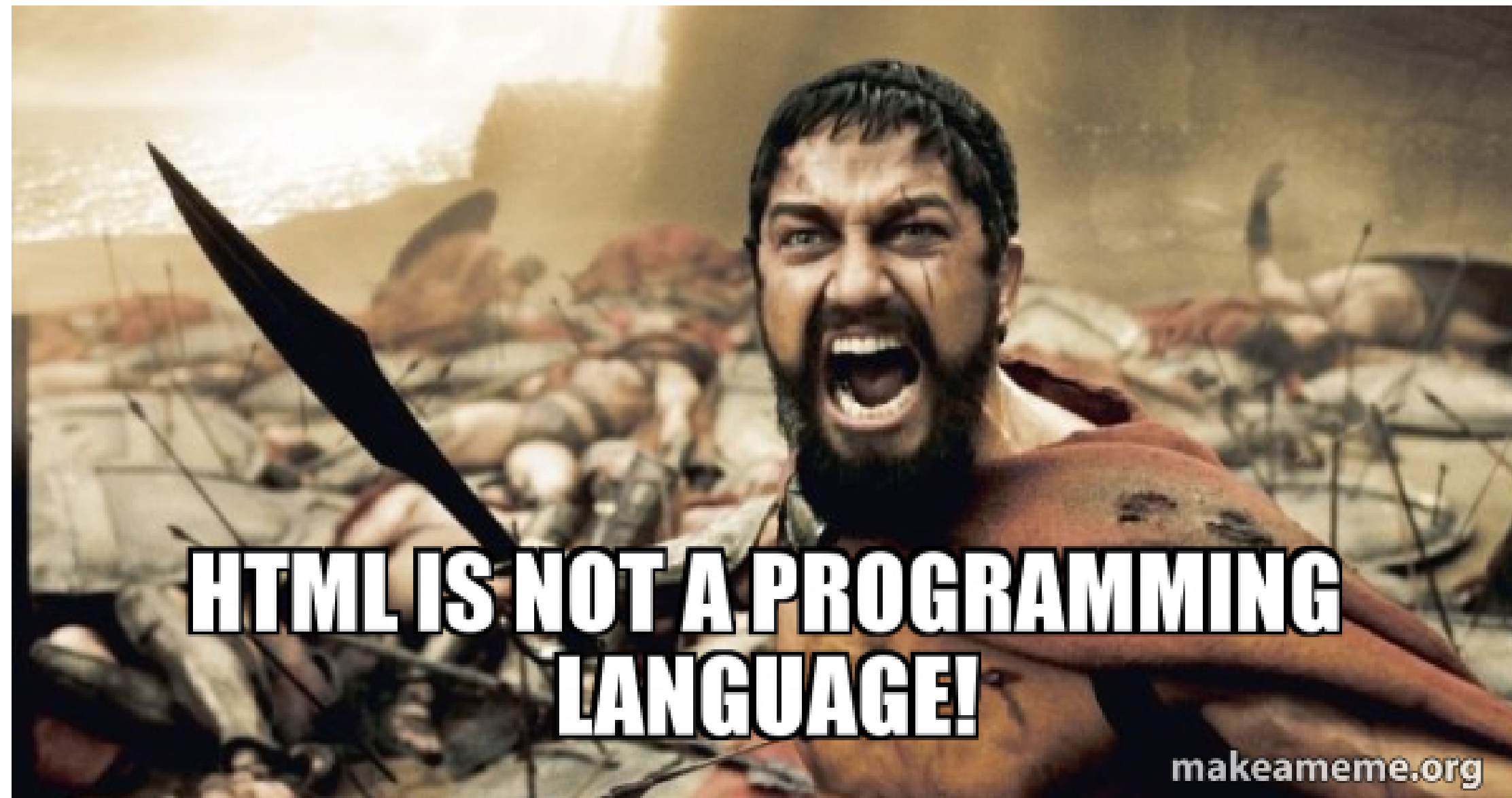
 Acheter

CSS

AJOUTER AU PANIER

JAVASCRIPT

Qu'est-ce que le HTML et quel est son rôle dans le web ?



Qu'est-ce que le HTML et quel est son rôle dans le web ?

👉 Le **HTML** (HyperText Markup Language) est le langage de base utilisé pour structurer une page web. Il définit la structure et le contenu d'un site grâce à un système de balises. Ces balises indiquent au navigateur comment organiser et afficher les éléments sur la page.



1. Il permet d'**afficher** du texte, des images, des vidéos, des liens, des tableaux, etc.
2. C'est un **langage facile à apprendre**, idéal pour débuter dans le développement web.
3. Une bonne structuration en HTML améliore le **référencement naturel** (SEO) d'un site web.

Les balises HTML : anatomie et utilisation

Les balises HTML sont des instructions entourées de chevrons < >

Elles s'écrivent toujours en minuscule et définissent les éléments de la page.

On peut tout à fait les imbriquer entre elles à condition de toujours les refermer.

👉 Il existe deux types de balises : **normales** ou **auto-fermantes**.

Balises normale

```
<balise>Bonjour</balise>
```

Balise auto-fermante

```
<balise>  
<balise />
```

La balise normale

👉 Une balise normale **s'ouvre et se ferme**. On a donc une balise ouvrante et une balise fermante. Elle indique au navigateur les délimitations de l'impact de la balise :

```
<balise>Bonjour</balise>
```

Imbrication de balises (ce qu'on appelle les noeuds) :

```
<p>Bonjour <strong>l'AFEC</strong> !</p>
```

Exemples de balises normales :

```
<p>Bonjour !</p>
```

```
<section>Bonjour !</section>
```

```
<nav>Bonjour !</nav>
```

```
<div>Bonjour !</div>
```

La balise auto-fermante

👉 Ces balises n'ont **pas de balise ouvrante ni fermante**. Elle servent le plus souvent à insérer un élément autonome à un emplacement précis. Il n'est pas nécessaire de délimiter le début et la fin de la balise.

`<balise>`

`<balise />`

Le slash (/) et l'espace de fin ne sont pas obligatoires. On peut écrire une balise auto-fermante de deux façons. Cependant, la bonne pratique recommande de ne pas ajouter un slash.

Exemples de balises normales :

`<br>`

`<hr>`

`<img src="" alt="">`

Les attributs universels en HTML

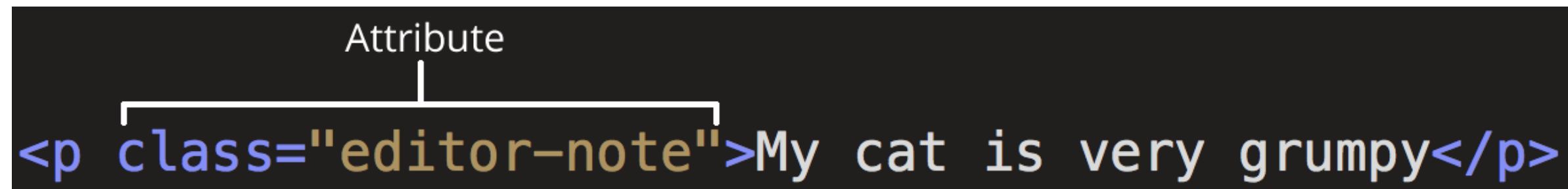
Les attributs universels HTML sont des **options supplémentaires** qui permettent d'ajouter des informations aux balises. Ils modifient leur comportement ou leur apparence.

👉 Règles d'utilisation des attributs :

- Ils sont placés dans la balise ouvrante.
- Une balise peut contenir plusieurs attributs.
- Chaque attribut a une valeur entre guillemets ("") sauf exceptions : required, disabled...
- Une balise ne peut pas avoir plusieurs attributs du même nom.

👉 Une balise auto-fermante peut également avoir des attributs.
Dans ce cas, ils sont contenus au niveau de la balise auto-fermante.

Exemple d'une balise normale Lien avec plusieurs attributs



The diagram shows the HTML tag `<p class="editor-note">My cat is very grumpy</p>` on a dark background. A white bracket is positioned above the tag, spanning from the opening angle bracket to the closing angle bracket. A vertical line descends from the center of this bracket to the word "Attribute", which is centered above the space between the opening and closing tags. The text "class='editor-note'" is highlighted in a light blue color.

```
<p class="editor-note">My cat is very grumpy</p>
```

Les attributs personnalisés en HTML

👉 Il existe deux catégories d'attributs :

- Les **attributs génériques** : Utilisables sur plusieurs types de balises (slide précédente)
- Les **attributs personnalisés** : Ces attributs sont inventés par le développeur et commencent toujours par data-. Ils servent à stocker des informations supplémentaires pour JavaScript.

Exemple d'une balise bouton avec plusieurs attributs personnalisés

```
<button data-id="123456" data-prix="99">Acheter</button>
```

Comment récupérer ces attributs avec JavaScript ?

```
let bouton = document.querySelector("button");  
console.log(bouton.dataset.id);  
console.log(bouton.dataset.prix);
```

L'indentation

👉 Il y a dans l'organisation de l'ouverture et de la fermeture des balises des décalages étranges. Les balises ne sont pas toutes au même niveau. En développement, on appelle ça **l'indentation**.

👉 Elle consiste à des **tabulations** au début de chaque ligne pour en faire ressortir certaines portions de code. Pour le HTML, l'indentation a un intérêt ergonomique et esthétique mais dans d'autres langages, elle s'avère obligatoire pour le bon fonctionnement.

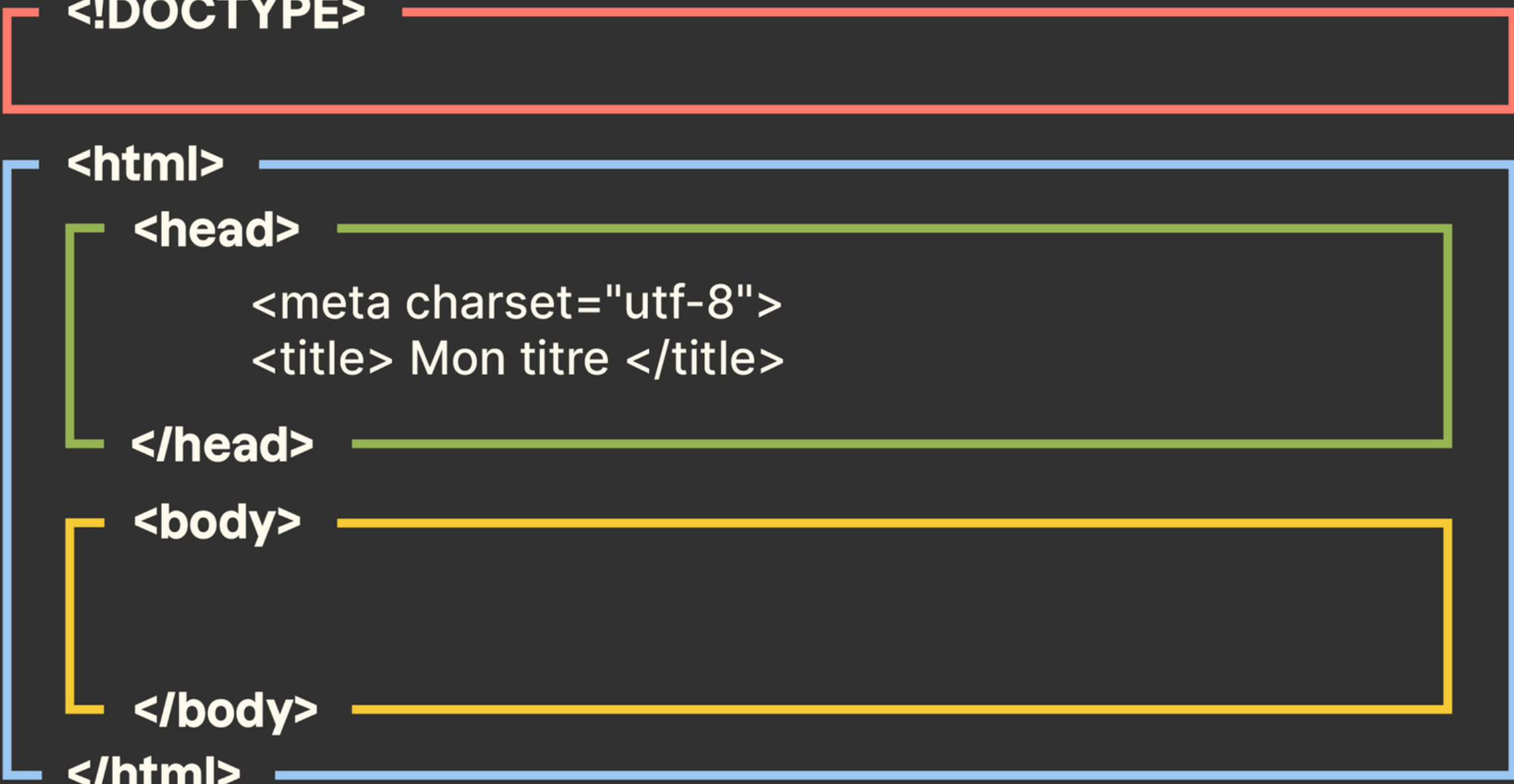
👉 Le code est également **plus lisible et maintenable**

👉 Il existe plusieurs **styles d'indentation**. Renseignez-vous sur celui qui vous correspond le mieux.



Structure de base d'une page HTML5

<!DOCTYPE>

The diagram illustrates the structure of an HTML5 document. It features a dark gray background with three nested rectangular boxes. The outermost box is red and contains the text '<!DOCTYPE>'. Inside this, a blue box contains '<html>' at the top and '</html>' at the bottom. Within the blue box, there is a green box containing '<head>' at the top, '</head>' at the bottom, and two lines of text in the middle: '<meta charset="utf-8">' and '<title> Mon titre </title>'. Below the green box, within the blue box, is a yellow box containing '<body>' at the top and '</body>' at the bottom.

<html>

<head>

<meta charset="utf-8">

<title> Mon titre </title>

</head>

<body>

</body>

</html>

Structure de base d'une page HTML5

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Titre de la page affiché dans la barre du navigateur</title>
  </head>
  <body>

    </body>
</html>
```

Le doctype <!DOCTYPE html>

👉 La toute première ligne d'un fichier HTML est appelée **Doctype**. Elle est indispensable pour assurer un affichage correct du fichier dans les navigateurs.

```
<!DOCTYPE html>
```

- Elle indique au navigateur **quelle version de HTML** est utilisée.
- Sans elle, les navigateurs peuvent interpréter le code de manière imprévisible.
- En HTML5, cette déclaration est simplifiée par rapport aux anciennes versions.
- Elle ne s'ouvre ni ne se ferme (pas de </DOCTYPE>).
- Elle n'est pas sensible à la casse (<!Doctype HTML> fonctionne aussi, mais on l'écrit toujours en majuscule pour les bonnes pratiques).
- Elle ne fait pas partie du corps de la page, elle est uniquement une instruction pour le navigateur.

La balise <html>

👉 Cette balise est **la principale de votre fichier HTML**.

Elle va englober toutes les prochaines. La balise ouvrante est tout au début de votre fichier et la balise fermante est tout à la fin.

- Un fichier HTML comporte automatiquement cette balise.
- On spécifie sur cette balise la langue utilisée pour le contenu de page grâce à l'attribut **lang=""**

```
<html></html>
```

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

La balise <head>

👉 L'élément **<head>** correspond à l'**en-tête d'un fichier HTML**. Il contient des **informations générales** (ou métadonnées) sur la page, qui sont essentielles pour les navigateurs et les moteurs de recherche, mais **ne sont pas affichées directement** aux utilisateurs.

```
<head></head>
```

👉 L'élément **<head>** permet de :

- Définir le titre de la page
- Spécifier l'encodage des caractères
- Lier des fichiers CSS pour le style de la page.
- Inclure des scripts JavaScript qui s'exécutent sur la page.
- Ajouter des métadonnées utiles pour le référencement.
- ⚠ Ne pas confondre les balises **<head>** et **<header>**

La contenu de la balise <head> : le minimum vital

(...)

<head>

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

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<meta name="description" content="Découvrez notre site.">
```

```
<meta name="author" content="Jean Dupont">
```

```
<title>Ma Première Page HTML</title>
```

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

```
<script src="script.js" defer></script> <!--peut aussi être placé dans le footer-->
```

</head>

(...)

La contenu de la balise <head> : le minimum vital

Balise	Fonction
<title>	Définit le titre affiché dans l'onglet du navigateur.
<meta charset="UTF-8">	Définit l'encodage des caractères pour afficher correctement les accents et symboles. ⚠ Ne pas confondre avec la langue
<meta name="viewport" content="width=device-width, initial-scale=1.0">	Permet un affichage responsive sur mobile.
<meta name="description" content="Texte descriptif">	Fournit une description de la page pour le référencement.
<meta name="author" content="Nom">	Indique le nom de l'auteur du site.
<link rel="stylesheet" href="style.css">	Lie un fichier CSS externe pour styliser la page.
<script src="script.js" defer></script>	Ajoute un fichier JavaScript, différé pour ne pas ralentir le chargement.

La contenu de la balise <head> en plus poussé

Balise	Fonction
<meta property="og:(...)" content="http://www.votresiteweb.com/image-nom.jpg"/>	Plusieurs balises opengraph existent à renseigner dans la partie <head> d'un document : lorsqu'un internaute partage la page sur un réseau social, l'aperçu s'affiche conformément aux instructions renseignées dans ces balises.
<link rel="icon" href="favicon.ico" />	Afficher le favicon d'un site.
<meta http-equiv="content-language" content="fr">	Par exemple, si "Content-Language: de-DE" est mis en place, cela signifie que la page est destinée à un public parlant l'allemand (par contre, cela n'indique pas que la page est écrite en allemand. Par exemple, elle pourrait être écrite en anglais dans le cadre d'un cours de langue destiné aux allemands).
<script></script>	L'élément <script> peut être utilisé dans la balise <head> pour intégrer ou faire référence à un script exécutable
<base>	L'élément <base> définit l'URL de base à utiliser pour recomposer toutes les URL relatives contenues dans un document. Il ne peut y avoir qu'un seul élément <base> au sein d'un document.

La balise <body>

👉 **Tout ce qui est affiché dans le navigateur est contenu dans <body> :**

<body></body>

- Titres, paragraphes, images, vidéos
- Liens, boutons, formulaires
- Tableaux, listes, menus
- ⚠ Il n'existe qu'un seul <body> par page HTML.
- ⚠ Il est placé après <head> dans la structure du document
- ⚠ <html> englobe toute la page, y compris <head> et <body>.
- ⚠ <body> ne contient que le contenu visible pour l'utilisateur.

La balise <body>

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Exemple de page</title>
</head>
<body>
  <h1>Bienvenue sur mon site !</h1>
  <p>Ceci est mon premier site en HTML.</p>
  
  <ul>
    <li>Article 1</li>
    <li>Article 2</li>
    <li>Article 3</li>
  </ul>
</body>
</html>
```

Exercice en binôme

1) Créer un dépôt GIT pour le binôme et le clôner

2) Préparer le squelette d'une page avec HTML en intégrant :

- Les balises indispensables pour structurer une page HTML avec du contenu français.
- Les balises indispensables pour le SEO
- La balise pour la rendre compatible pour du responsive design
- Toute autre balise que vous jugerez nécessaire

Aide : balises meta

<https://gist.github.com/whitingx/3840905>

Binôme 3
Sébastien +
Alvyn

Binôme 1
Alan +
Guillaume

Binôme 2
Nina +
Jean-François

Binôme 3
Angélique +
Nuno

Binôme 3
Ilona +
Adrien