

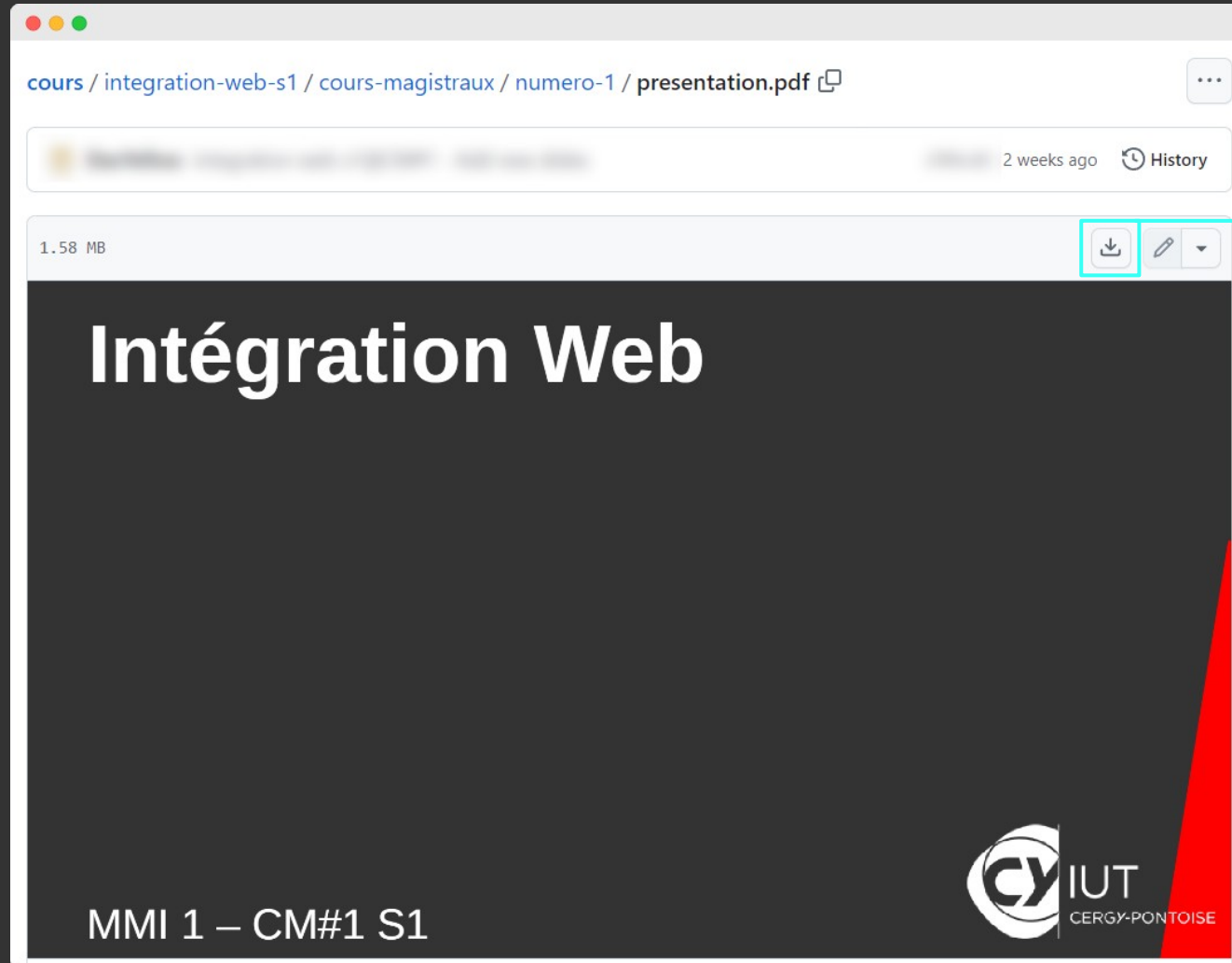
# Intégration Web

MMI 1 – TP#1 S1



Danielo **JEAN-LOUIS**  
Michele **LINARDI**

# Point sur les ressources de cours

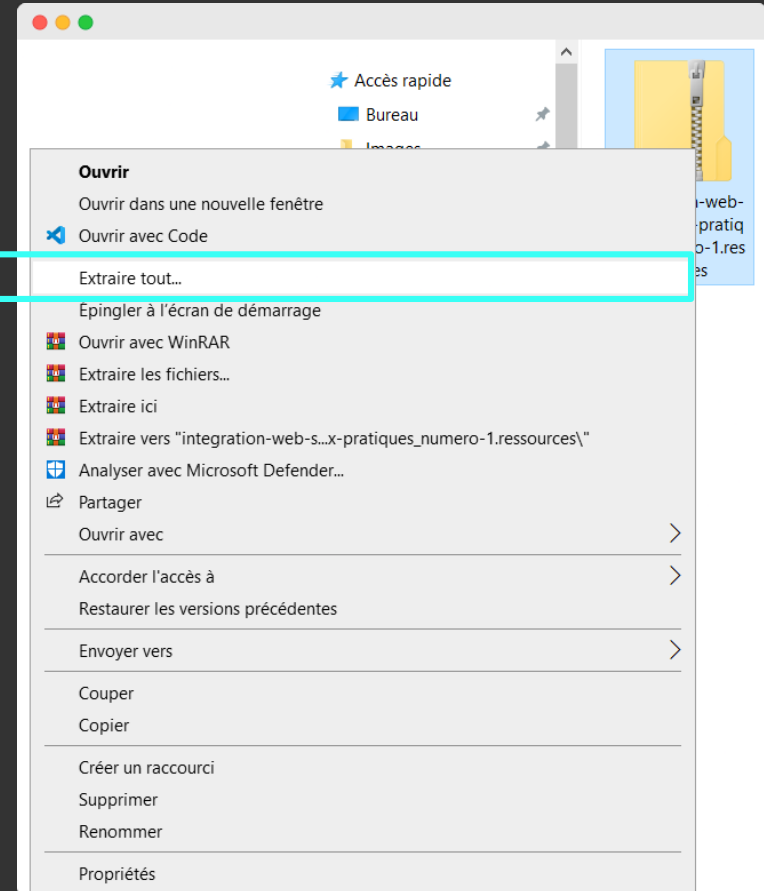


Cliquez dessus  
pour télécharger  
les pdf

# Point sur les ressources de cours

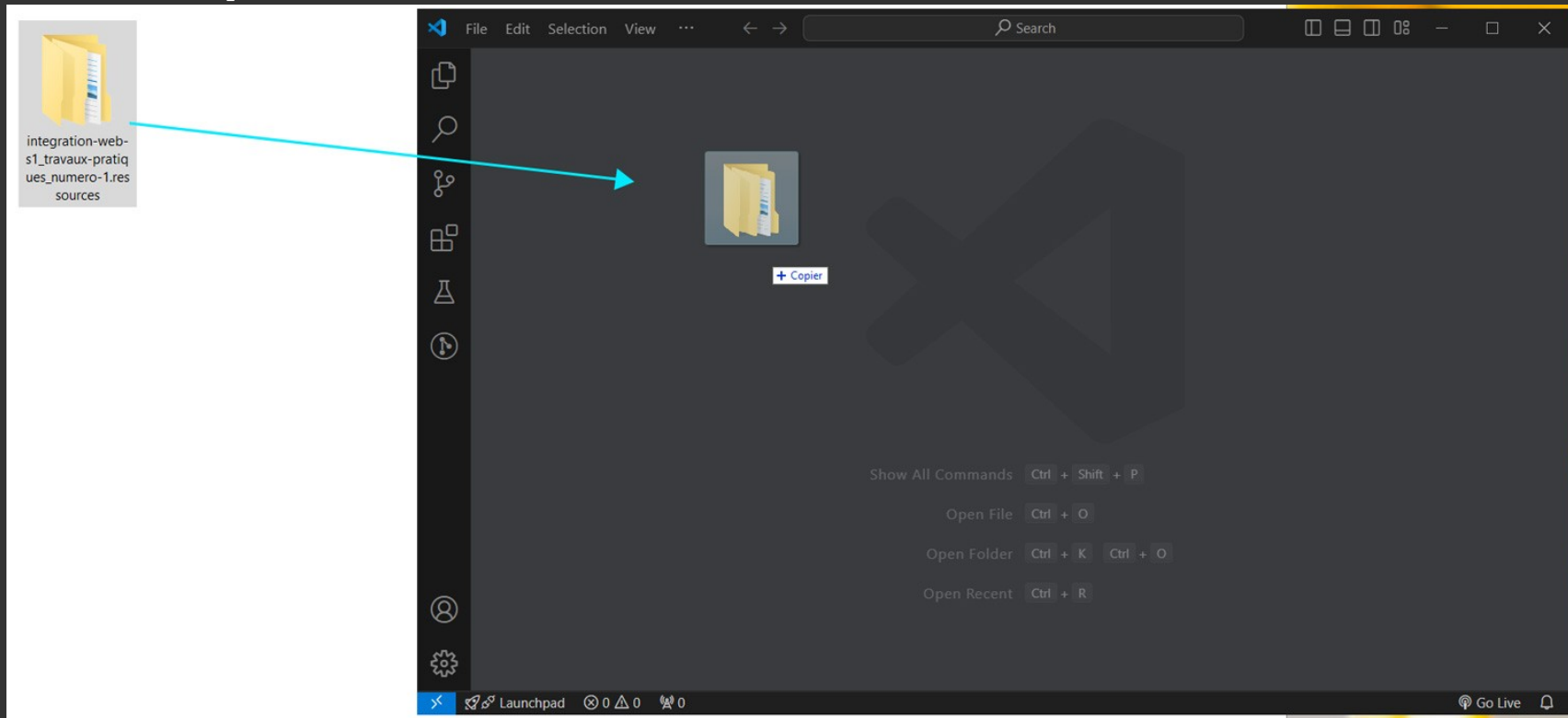
- Les pratiques sont contenues dans des archives (fichier .zip)
- Vous **devez** les extraire

Clic droit > Extraire tout



# Point sur les ressources de cours

- Glissez-déposez le dossier extrait dans VS Code pour avoir tous vos fichiers accessibles



# Pratiquons ! - Notre première page web (Partie 1)

Pré-requis :

- Avoir la ressource ressources/bases

A télécharger ici :

[https://github.com/DanYellow/cours/raw/refs/heads/main/integration-web-s1/travaux-pratiques/numero-1/integration-web-s1\\_travaux-pratiques\\_numero-1.ressources.zip](https://github.com/DanYellow/cours/raw/refs/heads/main/integration-web-s1/travaux-pratiques/numero-1/integration-web-s1_travaux-pratiques_numero-1.ressources.zip)

**Analysons tout ça ensemble**

# Squelette d'une page web – Balise <!DOCTYPE>

```
<!DOCTYPE html>
```

- Permet au navigateur de savoir quelle version de html est utilisée – ici html5
- Indispensable pour passer la validation W3C
- Balise pas vraiment auto-fermante et pas du tout classique
- Absence du <!doctype> active le mode quirks (appelé aussi mode compatibilité)
- **La balise est indispensable de nos jours**

## Sources :

- [https://developer.mozilla.org/fr/docs/Web/HTML/Quirks\\_Mode\\_and\\_Standards\\_Mode](https://developer.mozilla.org/fr/docs/Web/HTML/Quirks_Mode_and_Standards_Mode)
- <https://www.lesintegristes.net/2008/04/06/que-se-passe-t-il-en-mode-quirks-traduction/>
- <http://jkorpela.fi/quirks-mode.html>



# Squelette d'une page web – balise <html>


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

- Définit la racine structurelle d'un document html
- **Toutes les balises de la page** doivent se trouver dedans
- La balise doit contenir une balise <head> et une balise <body> (dans cet ordre)
- L'attribut lang permet de définir la langue de la page
- Cette balise est **unique**, il ne peut pas en avoir plusieurs par **page**

Source(s) :

- <https://www.w3.org/TR/2012/WD-html-markup-20121025/html.html#html>

# Squelette d'une page web – balise <head>



```
<head>
```

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

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

```
  <title>Ma première page web</title>
```

```
</head>
```

- Balise est **unique**, il ne doit pas en avoir plusieurs par page
- Contient toutes les balises <meta> de la page
  - Métadonnée : information générale de la page
- Contient la balise <title>

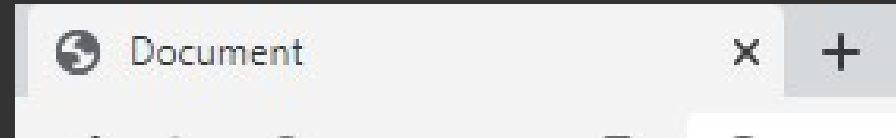
Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/head>
- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/meta>

# Squelette d'une page web – balise <title>

```
<title>Ma première page</title>
```

- Utile au référencement, le contenu est affiché sur la page de résultats des moteurs de recherche
- Contenu affiché dans l'onglet des navigateurs



- La balise est unique, il ne doit avoir qu'une seule balise <title> par page

Source(s) :

- <https://www.w3.org/TR/2012/WD-html-markup-20121025/title.html>
- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/title>

# Squelette d'une page web – balise <title>

- Permet à l'utilisateur de se repérer entre ses onglets / fenêtres
- Quelques règles :
  - Mettez un titre clair (une phrase)
    - Ex : « Menu – Mon super restaurant »
  - Titre unique pour chaque page d'un site

## Source(s) :

- <https://www.w3.org/TR/2012/WD-html-markup-20121025/title.html>
- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/title>

# Squelette d'une page web – balise <body>



```
<body>  
    <!-- Code de votre site -->  
</body>
```

- Cette balise est **unique**, il ne doit pas en avoir **plusieurs** par **fichier html**
- On y écrit les balises html qui affichent du contenu

Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/body>

# Balises HTML

- Deux types de fermeture de balises:
  - classiques
  - auto-fermantes

```
<p>Bonjour la promotion MMI</p>
```

Balise "classique"

```

```

Balise "auto-fermantes"

Source(s) :

- [https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises\\_HTML](https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises_HTML)

# Balises HTML

Balise  
ouvrante

Balise  
fermante

```
<p>Bonjour la promotion MMI</p>
```

Contenu

Source(s) :

- [https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises\\_HTML](https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises_HTML)
- [https://developer.mozilla.org/fr/docs/Learn/Getting\\_started\\_with\\_the\\_web/HTML\\_basics](https://developer.mozilla.org/fr/docs/Learn/Getting_started_with_the_web/HTML_basics) - anglais

# Balises HTML – balises "classique"

- S'ouvre avec une balise et se ferme avec la même balise avec un slash
- Le contenu se trouve entre les deux balises
- Exemple de balises "classique" :
  - `<div></div>` : `<div><p>Hello world</p></div>`
  - `<p></p>` : `<p>Bonjour la promotion MMI</p>`
  - `<a></a>` : `<a href="cyu.fr">Site université</a>`
  - [...]

Source(s) :

- [https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89%C3%A9ments\\_en\\_ligne](https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89%C3%A9ments_en_ligne)
- [https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89%C3%A9ments\\_en\\_bloc](https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89%C3%A9ments_en_bloc)



# Balises HTML – balises "classique"

- Les balises ne peuvent pas se chevaucher

```
<div>  
  <p>Hello world</div>  
</p>
```



**Interdit**

La balise `<p>` se ferme avant la balise `<div>`

```
<div>  
  <p>Hello world</p>  
</div>
```



**Correct**

Les balises s'ouvrent et se ferment dans le même ordre

Source(s) :

- [https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89l%C3%A9ments\\_en\\_ligne](https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89l%C3%A9ments_en_ligne)
- [https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89l%C3%A9ments\\_en\\_bloc](https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89l%C3%A9ments_en_bloc)

# Balises HTML – balises auto-fermantes

- Ne peut contenir du contenu (pas de balise fermante)
- Possède des attributs parfois nécessaires à son fonctionnement
- La barre oblique à la fin est optionnelle en HTML5
- Exemple de balises "auto-fermantes" :

- `<img />` : ``
- `<br>` : `<br>`
- `<hr>` : `<hr>`
- [...]

## Source(s) :

- [https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89%C3%A9ments\\_en\\_ligne](https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89%C3%A9ments_en_ligne)
- [https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89%C3%A9ments\\_en\\_bloc](https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89%C3%A9ments_en_bloc)

# Balises HTML

- Imbrication (infinie\*) possible

```
<ul>
  <li>Pikachu</li>
  <li>Pichu</li>
  <li>Posipi</li>
  <li>Pachirisu</li>
  <li>Emolga</li>
</ul>
```

```
<section>
  <ul>
    <li>
      <a href="">
        
      </a>
    </li>
  </ul>
</section>
```

Source :

- [https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises\\_HTML](https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises_HTML)

\* Certaines imbrications ne sont pas valides

# Pratiquons ! - Notre première page web (partie 2)

Pré-requis :

- Avoir la ressource ressources/bases

A télécharger ici :

[https://github.com/DanYellow/cours/raw/refs/heads/main/integration-web-s1/travaux-pratiques/numero-1/integration-web-s1\\_travaux-pratiques\\_numero-1.ressources.zip](https://github.com/DanYellow/cours/raw/refs/heads/main/integration-web-s1/travaux-pratiques/numero-1/integration-web-s1_travaux-pratiques_numero-1.ressources.zip)

# Balises HTML

- Deux types d'affichage pour les balises :
  - bloc
  - inline ("en ligne" en français)

## Sources :

- [https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89l%C3%A9ments\\_en\\_ligne](https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89l%C3%A9ments_en_ligne)
- [https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89l%C3%A9ments\\_en\\_bloc](https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89l%C3%A9ments_en_bloc)

# Pratiquons ! - Notre première page web (partie 3)

Pré-requis :

- Avoir la ressource ressources/bases

A télécharger ici :

[https://github.com/DanYellow/cours/raw/refs/heads/main/integration-web-s1/travaux-pratiques/numero-1/integration-web-s1\\_travaux-pratiques\\_numero-1.ressources.zip](https://github.com/DanYellow/cours/raw/refs/heads/main/integration-web-s1/travaux-pratiques/numero-1/integration-web-s1_travaux-pratiques_numero-1.ressources.zip)

# Balises type "bloc" - Caractéristiques

- Occupe tout l'espace disponible en largeur
- Force un retour à la ligne après elle (la pile d'assiette)
- **Peut** ne pas accepter une autre balise de type bloc
  - Ex : une balise `<p>` dans une autre balise `<p>`

## Sources :

- [https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89l%C3%A9ments\\_en\\_ligne](https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89l%C3%A9ments_en_ligne)
- [https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89l%C3%A9ments\\_en\\_bloc](https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89l%C3%A9ments_en_bloc)

# Balises type "bloc" - Caractéristiques

- Exemple de balises bloc
  - `<div>` : conteneur générique
  - `<p>` : paragraphe
  - `<ul>` : liste à puces non ordonnées
  - `<li>` : élément d'une liste à puces
  - [...]

Source(s) :

- [https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89l%C3%A9ments\\_en\\_ligne](https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89l%C3%A9ments_en_ligne)
- [https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89l%C3%A9ments\\_en\\_bloc](https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89l%C3%A9ments_en_bloc)



# Pratiquons ! - Notre première page web (partie 4)

Pré-requis :

- Avoir la ressource ressources/bases

A télécharger ici :

[https://github.com/DanYellow/cours/raw/refs/heads/main/integration-web-s1/travaux-pratiques/numero-1/integration-web-s1\\_travaux-pratiques\\_numero-1.ressources.zip](https://github.com/DanYellow/cours/raw/refs/heads/main/integration-web-s1/travaux-pratiques/numero-1/integration-web-s1_travaux-pratiques_numero-1.ressources.zip)

# Balises `<span>` et `<p>`

- Balises permettant d'afficher du texte (il en existe d'autres)
- `<span>` : balise inline
- `<p>` : balise bloc
- Possibilité d'imbriquer des `<span>` dans un `<p>` mais pas l'inverse

Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/p>
- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/span>

# Balises `<span>` et `<p>`

- Possibilité d'imbriquer un `<span>` dans un `<span>`
- **Impossible d'imbriquer un `<p>` dans un `<p>`**

Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/p>
- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/span>

# Balise <br>

- Permet de gérer des retours à la ligne **uniquement** quand ils sont importants :
  - poèmes
  - adresses
  - [...]
- Ne pas utiliser la balise dans d'autres cas

Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/HTML/Element/br>

# Balises type "en ligne" - Caractéristiques

- Peut contenir d'autres balises "inline"
  - Ne peut pas contenir de balises de type "bloc"
- Occupe la place de son contenu

Source(s) :

- [https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89l%C3%A9ments\\_en\\_ligne](https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89l%C3%A9ments_en_ligne)

# Balises type "en ligne" - Caractéristiques

- Exemple de balises inline
  - `<span>` : conteneur générique pour le texte
  - `<a>` : lien
  - `<select>` : liste d'options
  - [...]

Source(s) :

- [https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89l%C3%A9ments\\_en\\_ligne](https://developer.mozilla.org/fr/docs/Web/HTML/%C3%89l%C3%A9ments_en_ligne)

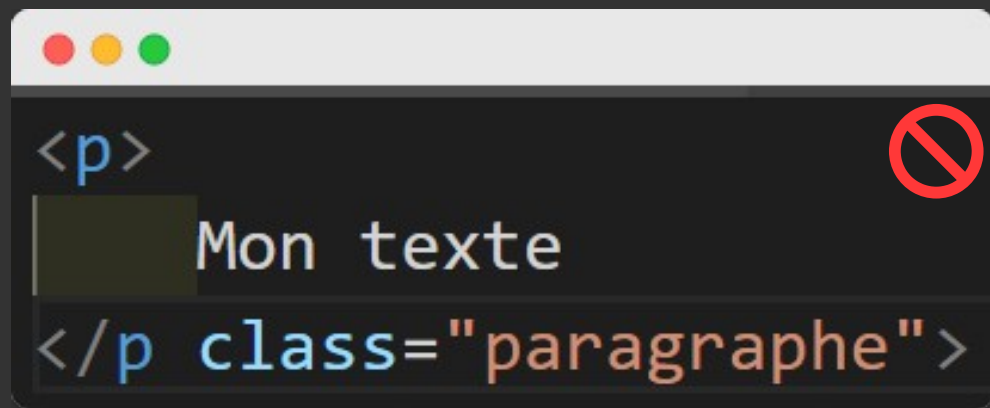
# Balises HTML - Attributs

- Représentent les informations complémentaires d'une balise
- Se placent **toujours dans la balise ouvrante** d'une balise
- Chaque attribut (+ valeur) doivent être espacés

## Source(s) :

- [https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises\\_HTML](https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises_HTML)
- <https://developer.mozilla.org/fr/docs/Web/HTML/Attributs>

# Balises HTML - Attributs

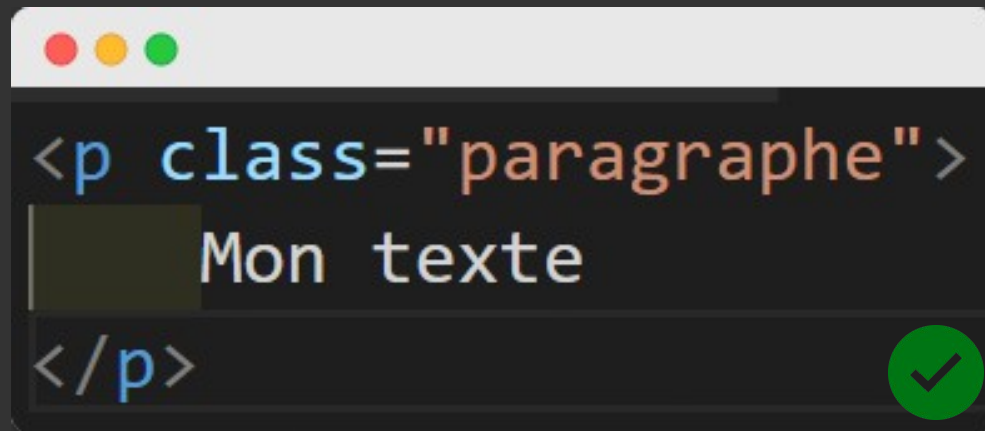


```
<p>  
Mon texte  
</p class="paragraphe">
```

A code editor window with a dark background and three colored window control buttons (red, yellow, green) at the top. The code inside shows an opening paragraph tag, the text "Mon texte", and a closing paragraph tag with an attribute. A red circle with a diagonal slash is positioned to the right of the code, indicating it is incorrect.

## Interdit

Mettre l'attribut dans la balise fermante



```
<p class="paragraphe">  
Mon texte  
</p>
```

A code editor window with a dark background and three colored window control buttons (red, yellow, green) at the top. The code inside shows an opening paragraph tag with an attribute, the text "Mon texte", and a closing paragraph tag. A green circle with a white checkmark is positioned to the right of the code, indicating it is correct.

## Correct

Mettre l'attribut dans la balise ouvrante

### Source(s) :

- [https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises\\_HTML](https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises_HTML)
- <https://developer.mozilla.org/fr/docs/Web/HTML/Attributs>



# Balises HTML - Attributs

- L'ordre des attributs n'a aucune importance
- Une balise peut contenir plusieurs attributs **uniques**
- La valeur d'un attribut doit être entre guillemets (simples ou doubles)
  - Si on ouvre avec des guillemets simples (') on ferme avec des guillemets simples (') idem pour les guillemets doubles
  - On ne peut donc pas combiner l'ouverture et la fermeture

Source(s) :

- [https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises\\_HTML](https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises_HTML)
- <https://developer.mozilla.org/fr/docs/Web/HTML/Attributs>

# Balises HTML - Attributs

- Certains attributs sont propres à certaines balises
- Ils peuvent être indispensables pour certaines balises
  - Exemple : Les attributs “alt” et “src” pour la balise “img”

## Source(s) :

- [https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises\\_HTML](https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises_HTML)
- <https://developer.mozilla.org/fr/docs/Web/HTML/Attributs>

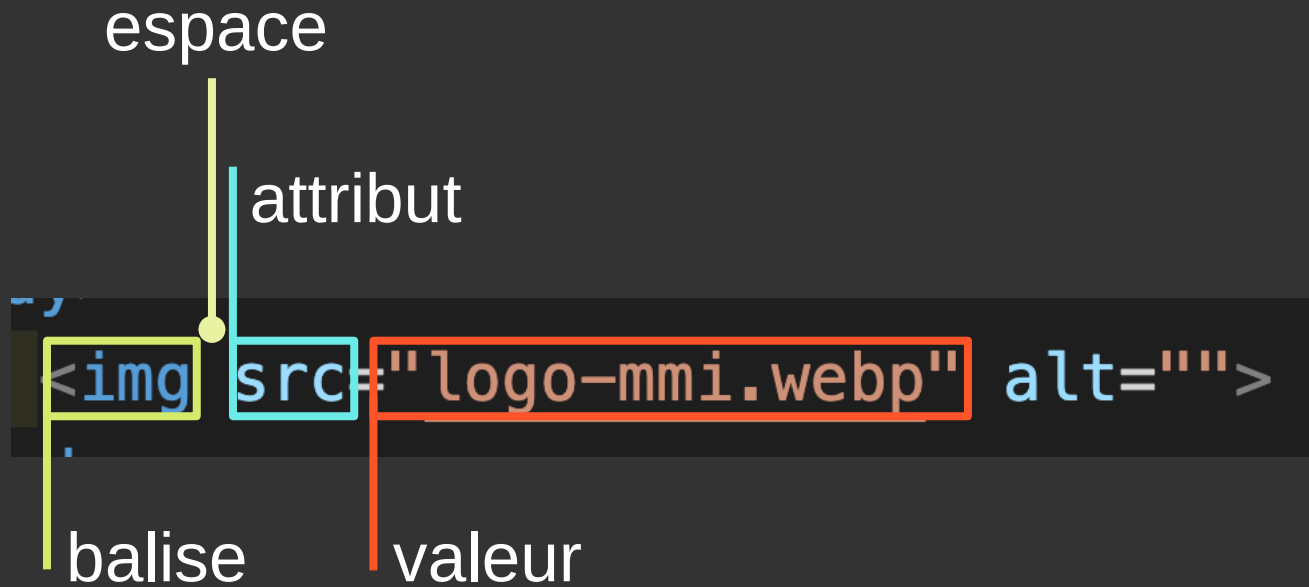
# Balises HTML - Attributs

- Si les attributs sont prédéfinis, ils peuvent complétés par les data-attributes
  - Attributs s'écrivant sous la forme data-{param}="valeur"  
Exemple : data-etudiants="63"
  - Utiles pour passer des informations vers d'autres langages

Source(s) :

- [https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises\\_HTML](https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises_HTML)
- <https://developer.mozilla.org/fr/docs/Web/HTML/Attributs>

# Balises HTML - Attributs



Source(s) :

- [https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises\\_HTML](https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises_HTML)
- <https://developer.mozilla.org/fr/docs/Web/HTML/Attributs>

# Pratiquons ! - Découvrons les attributs – Partie 1

Pré-requis :

- Avoir la ressource ressources/attributs

A télécharger ici :

[https://github.com/DanYellow/cours/raw/refs/heads/main/integration-web-s1/travaux-pratiques/numero-1/integration-web-s1\\_travaux-pratiques\\_numero-1.ressources.zip](https://github.com/DanYellow/cours/raw/refs/heads/main/integration-web-s1/travaux-pratiques/numero-1/integration-web-s1_travaux-pratiques_numero-1.ressources.zip)

## Point accessibilité – l'attribut "alt"

- Permet aux lecteurs d'écran de "lire" l'image
  - Mettez la valeur la plus descriptive possible
- **Doit toujours être présent même vide**
- Sert au référencement du site

## Point accessibilité – l'attribut "alt"

- Mieux vaut un alt vide qu'un alt mal nommé
  - Ne pas mettre : “Image 1”
- S'affiche lorsque l'image est absente
  - Ne s'affichera pas si l'image prend du temps à charger
- Inutile d'écrire “Image de chat noir”, “Image” est redondant

# Attribut src – Chemins relatifs et absolus

- En HTML les fichiers ne sont pas embarqués, **ils sont liés**
  - Déplacer un fichier peut avoir des conséquences
  - La valeur de l'attribut "src" peut-être un chemin relatif ou absolu

## Sources :

- <https://www.alsacreations.com/astuce/lire/78-Quelle-est-la-difference-entre-les-chemins-relatifs-et-absolus-.html>
- [https://developer.mozilla.org/fr/docs/Apprendre/HTML/Introduction\\_%C3%A0\\_HTML/Creating\\_hyperlinks#URL\\_absolue\\_vs.\\_URL\\_relative](https://developer.mozilla.org/fr/docs/Apprendre/HTML/Introduction_%C3%A0_HTML/Creating_hyperlinks#URL_absolue_vs._URL_relative)



# Attribut src – Chemins relatifs et absolus

- Le chemin est le cheminement entre le fichier qui charge le fichier (ex : index.html) et le fichier qui va être chargé (ex : image.png)
- Caractères clés :
  - ./ : \*Chemin relatif au répertoire courant
  - / : séparateur entre les dossiers
  - ../ : \*permet de remonter d'un niveau dans la structure des dossiers. Se place au début du chemin

\* : Ces caractères s'utilisent **uniquement** pour les chemins relatifs

# Attribut src – Chemins relatifs et absolus

- Ne jamais, jamais utiliser l'anti-slash (\) comme séparateur de dossiers dans une url ou un chemin de ressource

```

```



Anti-slash dans l'url = interdit

## Source(s) :

- <https://www.alsacreations.com/astuce/lire/78-Quelle-est-la-difference-entre-les-chemins-relatifs-et-absolus-.html>
- [https://developer.mozilla.org/fr/docs/Apprendre/HTML/Introduction\\_%C3%A0\\_HTML/Creating\\_hyperlinks#URL\\_absolue\\_vs.\\_URL\\_relative](https://developer.mozilla.org/fr/docs/Apprendre/HTML/Introduction_%C3%A0_HTML/Creating_hyperlinks#URL_absolue_vs._URL_relative)

# Chemins absolus

- Chemin pointant vers une **ressource non dépendante à votre projet**
  - **N'est pas influencé par la structure de vos dossiers**
    - Ex : Si vous déplacez votre fichier html, ça ne changera rien
  - Exemple : lien vers une image sur instagram
- Ne pas utiliser les chemins absolus propres à votre ordinateur

## Source(s) :

- <https://www.alsacreations.com/astuce/lire/78-Quelle-est-la-difference-entre-les-chemins-relatifs-et-absolus-.html>
- [https://developer.mozilla.org/fr/docs/Apprendre/HTML/Introduction\\_%C3%A0\\_HTML/Creating\\_hyperlinks#URL\\_absolue\\_vs.\\_URL\\_relative](https://developer.mozilla.org/fr/docs/Apprendre/HTML/Introduction_%C3%A0_HTML/Creating_hyperlinks#URL_absolue_vs._URL_relative)

# Chemins absolus

- Exemple avec une ressource en ligne

```

```

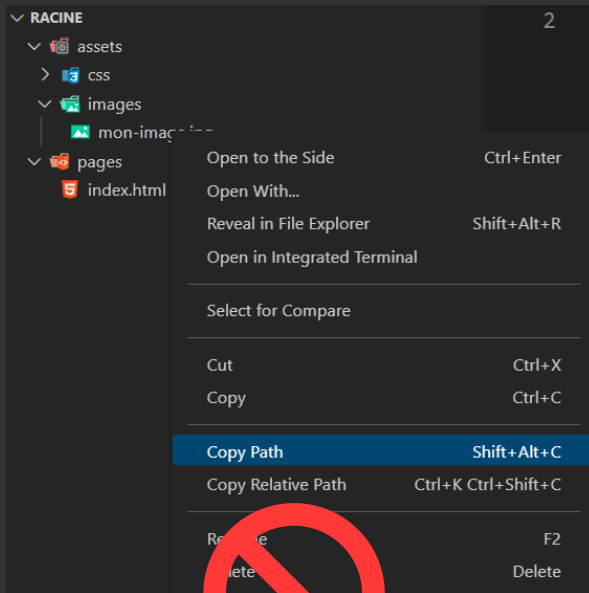
- Utiliser un chemin absolu local créera du travail supplémentaire sur les liens quand le site sera en ligne  
Un chemin comme "file:///C:/Users/PC/Desktop/mon-image.jpg" **est à bannir**

Source(s) :

- <https://www.alsacreations.com/astuce/lire/78-Quelle-est-la-difference-entre-les-chemins-relatifs-et-absolus-.html>
- [https://developer.mozilla.org/fr/docs/Learn/HTML/Introduction\\_to\\_HTML/Creating\\_hyperlinks#url\\_absolue\\_et\\_url\\_relative](https://developer.mozilla.org/fr/docs/Learn/HTML/Introduction_to_HTML/Creating_hyperlinks#url_absolue_et_url_relative)

# Chemins absolus

- Ne jamais, jamais utiliser le chemin absolu proposé par votre ordinateur
  - Ni la fonctionnalité de VS Code "Copy path"



- En utilisant la fonctionnalité "copy path" (clic droit sur le fichier), on obtient un chemin comme : "D:\MMI\dossier\_1\assets\images\mon-image.jpg"
- **Ce chemin ne va fonctionner QUE sur cet ordinateur.**

# Chemins absolus - Attention au hotlinking

- Lien vers une ressource externe sans demander l'autorisation au détenteur du serveur original
- Peut être illégal dans certains cas
- Augmente la charge d'un serveur tiers
- Pensez à télécharger l'image pour la charger de votre côté

Source(s) :

- <https://fr.wikipedia.org/wiki/Hotlinking>

# Chemins relatifs

- Chemin pointant vers une **ressource interne** à votre projet
  - **Est influencé par la structure de vos dossiers**
    - Ex : Si vous déplacez votre fichier html, ceci aura des conséquences
  - Exemple : lien vers une autre page html
- Le chemin est toujours construit par rapport au fichier qui charge la ressource

## Sources :

- <https://www.alsacreations.com/astuce/lire/78-Quelle-est-la-difference-entre-les-chemins-relatifs-et-absolus-.html>
- [https://developer.mozilla.org/fr/docs/Apprendre/HTML/Introduction\\_%C3%A0\\_HTML/Creating\\_hyperlinks#URL\\_absolue\\_vs.\\_URL\\_relative](https://developer.mozilla.org/fr/docs/Apprendre/HTML/Introduction_%C3%A0_HTML/Creating_hyperlinks#URL_absolue_vs._URL_relative)

# Chemins relatifs

**Le chemin est toujours calculé par rapport au fichier qui veut charger la ressource**

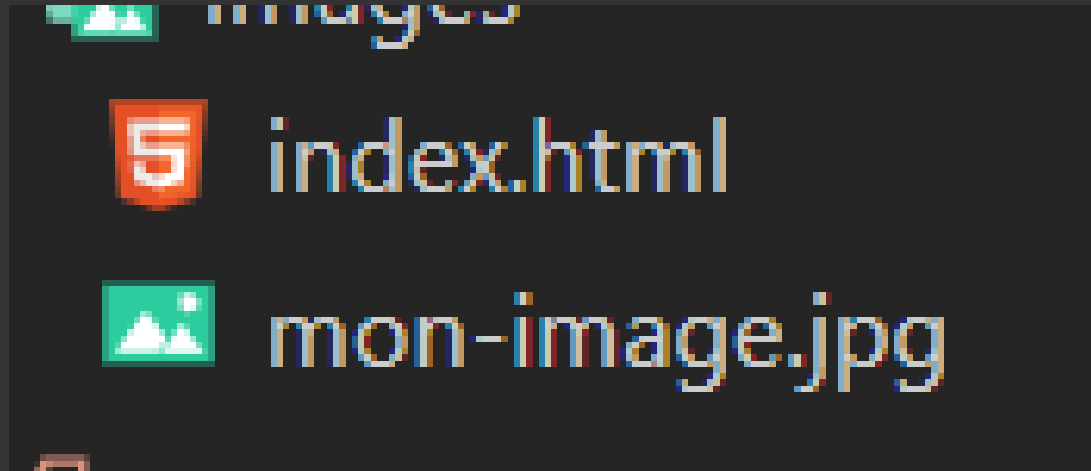


# Chemins relatifs

**Dans le cadre du cours, nous chargerons nos fichiers depuis les fichiers HTML (et parfois CSS)**

# Chemins relatifs – Exemple 1

- Si les deux fichiers sont au même niveau



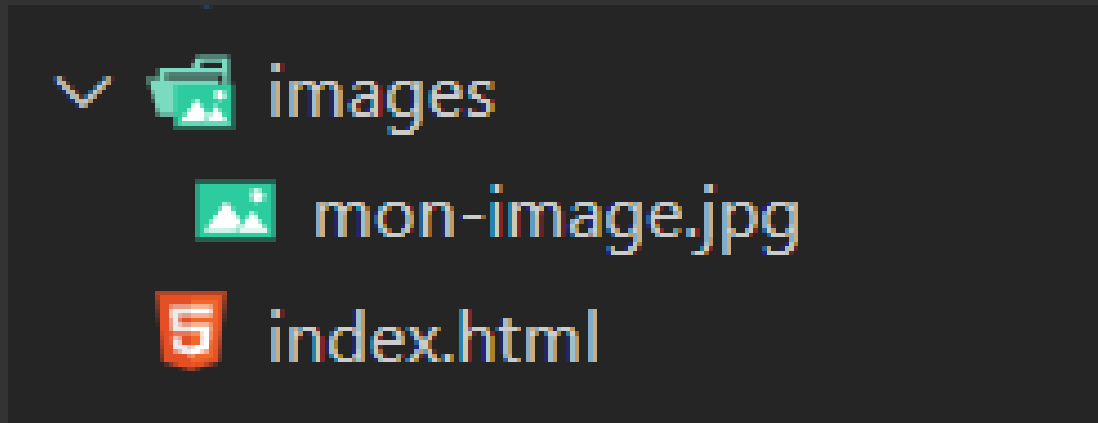
Chemin relatif = ./mon-image.jpg

Source(s) :

- <https://www.alsacreations.com/astuce/lire/78-Quelle-est-la-difference-entre-les-chemins-relatifs-et-absolus-.html>
- [https://developer.mozilla.org/fr/docs/Apprendre/HTML/Introduction\\_%C3%A0\\_HTML/Creating\\_hyperlinks#URL\\_absolue\\_vs.\\_URL\\_relative](https://developer.mozilla.org/fr/docs/Apprendre/HTML/Introduction_%C3%A0_HTML/Creating_hyperlinks#URL_absolue_vs._URL_relative)

# Chemins relatifs – Exemple 2

- Si les fichiers ne sont pas dans le même dossier, le caractère slash (/) sert de séparateurs entre les dossiers et fichiers du dossier



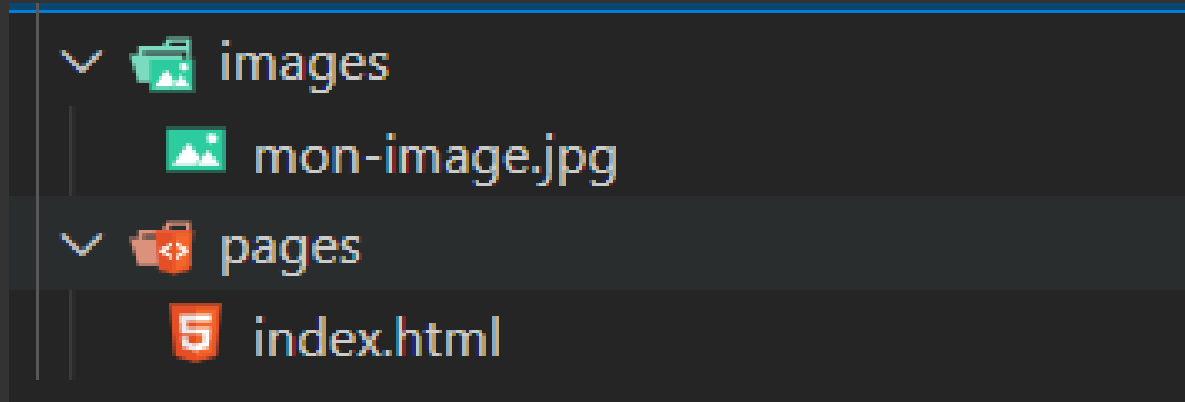
Chemin relatif = ../images/mon-image.jpg

Source(s) :

- <https://www.alsacreations.com/astuce/lire/78-Quelle-est-la-difference-entre-les-chemins-relatifs-et-absolus-.html>
- [https://developer.mozilla.org/fr/docs/Apprendre/HTML/Introduction\\_%C3%A0\\_HTML/Creating\\_hyperlinks#URL\\_absolue\\_vs.\\_URL\\_relative](https://developer.mozilla.org/fr/docs/Apprendre/HTML/Introduction_%C3%A0_HTML/Creating_hyperlinks#URL_absolue_vs._URL_relative)

# Chemins relatifs – Exemple 3

- Si le fichier que je veux charger est dans un répertoire parent (un niveau au-dessus), on doit écrire "../". Une fois au bon niveau, je mets le nom du dossier où est contenu mon image

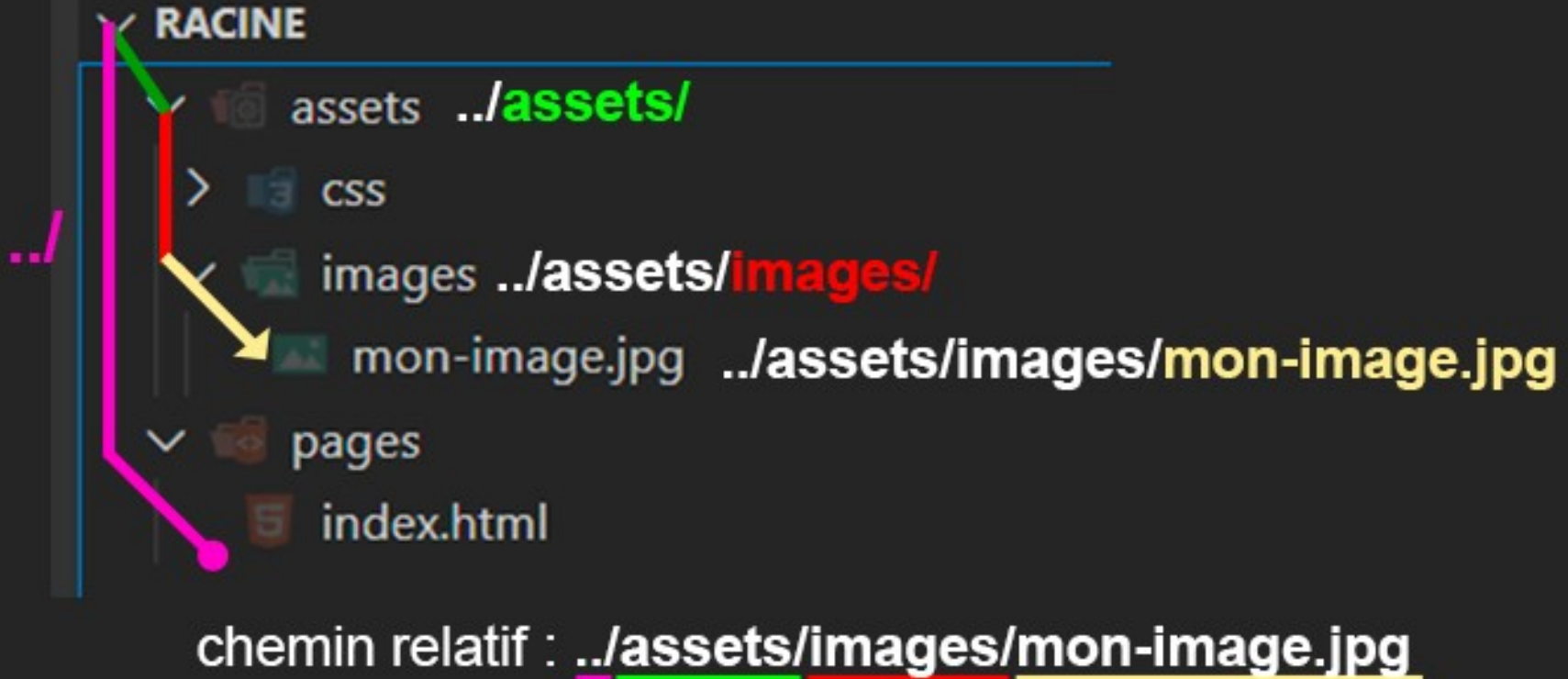


Chemin relatif = ../images/mon-image.jpg

## Source(s) :

- <https://www.alsacreations.com/astuce/lire/78-Quelle-est-la-difference-entre-les-chemins-relatifs-et-absolus-.html>
- [https://developer.mozilla.org/fr/docs/Apprendre/HTML/Introduction\\_%C3%A0\\_HTML/Creating\\_hyperlinks#URL\\_absolue\\_vs.\\_URL\\_relative](https://developer.mozilla.org/fr/docs/Apprendre/HTML/Introduction_%C3%A0_HTML/Creating_hyperlinks#URL_absolue_vs._URL_relative)

# Chemins relatifs – En détails

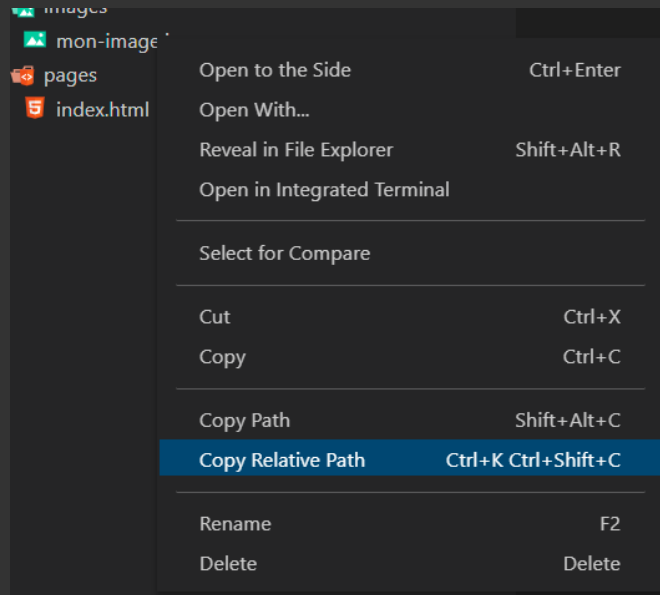


## Sources :

- <https://www.alsacreations.com/astuce/lire/78-Quelle-est-la-difference-entre-les-chemins-relatifs-et-absolus-.html>
- [https://developer.mozilla.org/fr/docs/Apprendre/HTML/Introduction\\_%C3%A0\\_HTML/Creating\\_hyperlinks#URL\\_absolue\\_vs.\\_URL\\_relative](https://developer.mozilla.org/fr/docs/Apprendre/HTML/Introduction_%C3%A0_HTML/Creating_hyperlinks#URL_absolue_vs._URL_relative)

# Chemins relatifs – Astuce VS Code

- Le logiciel permet de récupérer le chemin relatif d'un fichier **par rapport au projet**. Pour ce faire :



Menu quand vous faites un clic droit sur un fichier

- Faites un clic droit sur le nom du fichier (dans le panneau de gauche OU les onglets, il faut que le fichier soit ouvert)
- Sélectionnez l'option "copy relative path" (copier le chemin relatif)
- Collez-le où vous en avez besoin
  - N'oubliez pas de faire un ajustement si nécessaire

Note : VSCode propose un système d'auto-complétion quand vous écrivez vos chemins

# Chemins caractères de chemins – En résumé

- ./ : Répertoire courant
- ../ : Répertoire **parent** du répertoire courant
  - Possibilité d'enchaîner “../” pour remonter de plusieurs niveaux
- / (en début de chemin) : désigne la racine du serveur
- / (dans un chemin) : désigne un séparateur de dossiers

## Source(s) :

- <https://www.alsacreations.com/astuce/lire/78-quelle-est-la-difference-entre-les-chemins-relatifs-et-absolus.html>
- <https://stackoverflow.com/questions/7591240/what-does-dot-slash-refer-to-in-terms-of-an-html-file-path-location#answer-33774151>

# Règles de nom de fichier

- Préférez les noms en minuscules (de même pour l'extension du fichier)
  - Windows n'est pas sensible à la casse mais UNIX, oui. Donc : mon-image.jpg  $\neq$  MON-IMAGE.JPG
  - +90 % des serveurs sont sous système UNIX
- Évitez les espaces
  - Remplacer par un tiret (-) ou un underscore (\_)



# Règles de nom de fichier

- Évitez les caractères spéciaux (accents, ...)
  - Remplacez les lettres accentuées par la version sans accent
- Préférez des noms clairs et représentatifs
  - Pour savoir le contenu de vos fichiers sans les ouvrir

# Règles de nom de fichier

**Renommez vos fichiers s'ils ne correspondent pas à ces critères**

# Règles de nom de fichier

Ma photo d'été.JPG → ma-photo-d-ete.jpg



**Déconseillé**

- Espaces
- Accents
- Majuscules
- Caractères spéciaux



**Conseillé**

- Tiret/underscore à la place d'espaces
- Minuscules
- Pas de lettres accentuées
- Pas de caractères spéciaux

# Pratiquons ! - Découvrons les attributs – Partie 2/3

Pré-requis :

- Avoir la ressource ressources/attributs

A télécharger ici :

[https://github.com/DanYellow/cours/raw/refs/heads/main/integration-web-s1/travaux-pratiques/numero-1/integration-web-s1\\_travaux-pratiques\\_numero-1.ressources.zip](https://github.com/DanYellow/cours/raw/refs/heads/main/integration-web-s1/travaux-pratiques/numero-1/integration-web-s1_travaux-pratiques_numero-1.ressources.zip)

# Point accessibilité – Images

- Rappel : Présence de l'attribut "alt" même s'il doit être vide
- Les mobiles utilisent souvent la data pour accéder à Internet. Mettre des images de 15 Mo (même très belles) consomme beaucoup de données
  - Limiter la taille des images. Préférer des images qui font 1 Mo ou moins

## Source(s) :

- <https://jakearchibald.com/2020/avif-has-landed/> - anglais
- <https://squoosh.app/> - pour compresser vos images
- [https://developer.mozilla.org/fr/docs/Web/Media/Formats/Image\\_types](https://developer.mozilla.org/fr/docs/Web/Media/Formats/Image_types)

# Point accessibilité – Images

- Formats à utiliser :
  - jp(e)g / png / gif
  - webp : format avec compression sans perte par Google (pas géré par Safari)
    - Peu populaire
  - AVIF : format compressé sans perte
    - Gère la transparence et les animations
    - Géré par tous les navigateurs modernes
  - svg : pour les logos / icônes – format vectoriel

## Source(s) :

- <https://jakearchibald.com/2020/avif-has-landed/> - anglais
- <https://squoosh.app/> - pour compresser vos images
- [https://developer.mozilla.org/fr/docs/Web/Media/Formats/Image\\_types](https://developer.mozilla.org/fr/docs/Web/Media/Formats/Image_types)

# Point accessibilité – Images

- jpeg : format d'image **compressé avec perte**
  - Extension : .jpg ou .jp(e)g
  - Format le plus courant sur le web
- png : format d'image **compressé sans perte**
  - Extension : .png
  - Gère la transparence
  - Format lourd. Utilisez-le si vous avez besoin de transparence ou de “précision” pour les textes

Source(s) :

- <https://graphilink.fr/jpg-png-gif-differences/>
- <https://lehollandaisvolant.net/tuto/html.php>

# Point accessibilité – Images

- gif : format d'image **compressé avec perte**
  - Extension : .gif
  - Gère les animations
  - Limité à 256 couleurs → Inadapté pour vos photos de vacances
  - Format lourd

## Source(s) :

- <https://graphilink.fr/jpg-png-gif-differences/>
- <https://lehollandaisvolant.net/tuto/html.php>



# Balises HTML - Attributs

- Permettent d'ajouter des fonctionnalités supplémentaires aux balises
- Certains attributs sont exclusifs à certaines balises
  - Exemple : l'attribut "src" n'aura aucun effet sur les balises `<p>` ou `<li>`

## Source(s) :

- [https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises\\_HTML](https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises_HTML)
- <https://developer.mozilla.org/fr/docs/Web/HTML/Attributs> - Liste attributs existants

# Balises HTML - Attributs

- L'ordre des attributs dans une balise (ouvrante) n'a aucune importance
- La même balise peut avoir plusieurs attributs **uniques**

## Source(s) :

- [https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises\\_HTML](https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises_HTML)
- <https://developer.mozilla.org/fr/docs/Web/HTML/Attributs> - Liste attributs existants

# Balises HTML - Attributs

Attribut

```
<a href="https://www.u-cergy.fr/">Site de l'université de Cergy-Pontoise</a>
```

Balise

Valeur d'attribut

Fermeture balise

## Sources :

- [https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises\\_HTML](https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises_HTML)
- <https://developer.mozilla.org/fr/docs/Web/HTML/Attributs> - Liste attributs existants

# Balises HTML – Exemple d'attributs

- `id="val"` : identifiant unique, utilisable dans le CSS
  - La valeur d'un id est unique dans une page
  - Un élément ne peut pas avoir plusieurs id

```
<div id="hello">World</div>
```

- `class="val"` : Utilisable dans le CSS. La valeur peut être multiple dans une page

```
<span class="hello">World</span>
```

Source(s) :

- [https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises\\_HTML](https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises_HTML)
- <https://developer.mozilla.org/fr/docs/Web/HTML/Attributs> - Liste attributs existants

# Balises HTML – Exemple d'attributs

- `src="val"` : Utilisé pour les balises permettant d'afficher un média (vidéo, image, son)

```
<audio  
  controls  
  src="/media/cc0-audio/t-rex-roar.mp3"  
>  
  Votre navigateur ne supporte pas cette balise.  
</audio>
```

- `title="val"` : Utilisable par toutes les balises pour afficher une information supplémentaire au survol avec la souris

```
<span title="hello">World</span>
```



Rendu HTML  
(au survol)

Source(s) :

- [https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises\\_HTML](https://developer.mozilla.org/fr/docs/Apprendre/HTML/Balises_HTML)
- <https://developer.mozilla.org/fr/docs/Web/HTML/Attributs> - Liste attributs existants

# Pratiquons ! - Découvrons les attributs – Partie 4

Pré-requis :

- Avoir la ressource ressources/attributs

A télécharger ici :

[https://github.com/DanYellow/cours/raw/refs/heads/main/integration-web-s1/travaux-pratiques/numero-1/integration-web-s1\\_travaux-pratiques\\_numero-1.ressources.zip](https://github.com/DanYellow/cours/raw/refs/heads/main/integration-web-s1/travaux-pratiques/numero-1/integration-web-s1_travaux-pratiques_numero-1.ressources.zip)

## Point accessibilité – l'attribut "title"

- Permet d'ajouter des informations supplémentaires à un élément
- Ne doit pas se substituer à du texte critique visible sans interaction (survol uniquement)
  - **Ne s'affiche pas pour les utilisateurs au clavier ni sur terminal tactile**

# html – Récapitulatif

- Langage de description
  - Les structures comme “if” ou “while” sont impossibles
- Utilisation de balises qui peuvent être
  - auto-fermantes `<br>`
  - “classique” `<p>Hello world</p>`
- Des attributs peuvent être ajoutés pour ajouter des informations complémentaires à une balise



# html – Récapitulatif

- La valeur de l'attribut "id" doit être unique sur une page
- Une balise HTML peut être :
  - inline : sa largeur est égale à son contenu
  - block : sa largeur est égale à l'espace disponible mais elle peut être définie via le CSS (notion abordée plus tard - TP#3)

# html – Récapitulatif

- Les balises peuvent s'imbriquer
  - Attention à leurs fermetures
- Certaines balises ont un rôle qui leur est propre
  - Gare au respect de la sémantique
- Extension de fichiers .html
- On développe souvent avec VS Code (gratuit)



**Pour s'entraîner :**  
"Test – Bases du HTML" sur ENT

## A propos des tests

- Tests obligatoires mais non notés
  - Vous êtes obligé de faire le test du TP1 pour accéder à celui du TP2, etc.
  - Indispensable pour accéder au test final qui sera noté
  - Tentatives illimitées avec délai entre chaque tentative

## A propos des tests

- Reprennent les notions des TP passés
  - Ex : test TP#3 = questions TP#1 + #2 + #3
  - Des questions nécessitent parfois des recherches en ligne
    - **Vous avez le droit à Internet**
- Plus de détails dans la description des tests

**Questions ?**

