

HTML

Hyper Text Mark-up Language

I. Qu'est-ce que le HTML ?

- A. Un bref historique...
- B. Présentation du langage
- C. Les outils
- D. Les sources

II. Comment s'écrit le HTML ?

- A. Syntaxe des balises
- B. Fonction des balises
- C. Les principales balises
- D. Un code propre

III. Structure d'une page HTML

- A. Structure globale
- B. La balise <head>
- C. Les balises <meta>
- D. Les balises <link>

IV. Analysons les balises

- A. Type block ou inline
- B. Attributs et valeurs
- C. class et id
- D. Les liens

V. Formulaire et tableaux

- A. Structure générale
- B. Les champs de saisie
- C. Les listes d'options
- D. Vérification des données
- E. Les tableaux

Qu'est-ce que le HTML ?

Un bref historique...



1991 - création du **Web** par **Tim Berners-Lee**

1991-1993 - création des bases du **HTML**

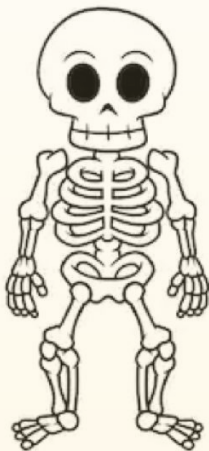
1994 - création du **W3C** (World Wide Web Consortium)

2008 - **version 5** du HTML

Présentation du langage

- ❑ Ce langage s'adresse au **navigateur** qui va le lire, l'analyser puis l'afficher sous la forme d'une **page web**
- ❑ Une **page web** standard est formée principalement de trois couches successives :
 - Le **HTML** [squelette de la page]
 - Le **CSS** [style et mise en forme]
 - Le **JavaScript** [interactivité et animations]
- ❑ Le langage **HTML** s'écrit dans un **éditeur de texte** (même dans un simple Bloc Notes)
- ❑ Le document doit être enregistré avec l'extension **.html** pour pouvoir être interprété correctement par le **navigateur**

Présentation du langage



HTML

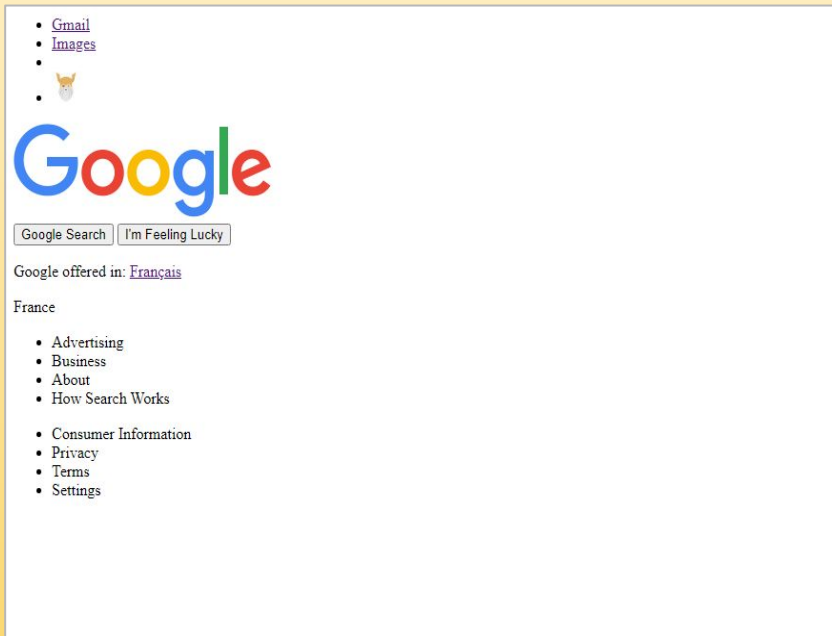


HTML + CSS

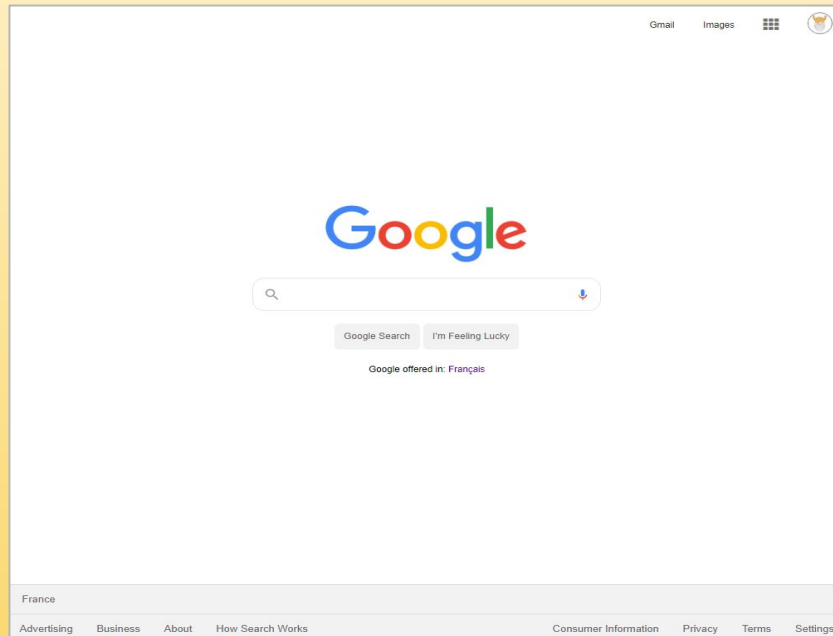


HTML+CSS+JavaScript

Présentation du langage



- HTML -



- HTML + CSS -

Présentation du langage

- ❑ Le **HTML** tout comme le **CSS** ne sont pas considérés comme des langages de **programmation** à proprement parler puisqu'aucun **script** n'est exécuté par le **navigateur**
- ❑ Le **HTML** est un simple langage de **balisage** utilisé pour définir la structure, le contenu et la sémantique d'une **page web**
- ❑ Ce langage est composé de **balises** contenant les différents éléments d'une **page web**. Ces **balises** indiquent au **navigateur** le type de contenu à afficher ainsi que sa valeur sémantique
- ❑ Le **HTML** est donc utilisé pour le référencement naturel (**Search Engine Optimization**) mais également pour les lecteurs de **page web** (différences d'intonation, précision du type de contenu affiché...)

Les outils

- ❑ Un **éditeur de texte** appelé aussi **IDE** (Integrated Development Environment) facilite l'écriture du **code** grâce à de nombreuses **fonctionnalités** comme la coloration syntaxique, des raccourcis clavier...

Sublime Text / Atom / Visual Studio Code ...

- ❑ De nombreux **éditeurs** de texte **en ligne** (sandbox) sont également disponibles pour pouvoir rapidement **tester** des bouts de code

JSFiddle / JsBin / CodeSandBox / CodePen

- ❑ L'extension **Emmet** présente par défaut dans l'éditeur de texte **VS Code** offre une **autocomplétion** à partir d'abréviations et raccourcis clavier

❑ <https://docs.emmet.io/abbreviations/syntax/>

Les outils

- ❑ L'**inspecteur** fourni par les navigateurs **Chrome** et **Firefox**, accessible avec la touche **f12** ou via le menu **Outils de développement**, permet d'avoir accès à la structure **HTML** d'une page
- ❑ Le validateur officiel de syntaxe **HTML** permet de vérifier la **conformité** de notre code aux normes **W3C**
- ❑ Des générateurs de faux texte [**Lorem Ipsum**] permettent d'avoir un rapide visuel de notre page **HTML**
- ❑ Des générateurs de **fausses images** offrent le même visuel rapide grâce des images de **toutes tailles**
- ❑ <https://validator.w3.org/> | <https://www.loremipsums.nl/> | <https://the-best-placeholder-generators/>

Les sources

- ❏ La documentation technique de **Mozilla Developer Network** pour les langages **HTML / CSS / JavaScript**
 - ❏ [MDN Web Docs](#)  
- ❏ Une documentation technique avec un **éditeur en ligne** pour rapidement y **tester** des fonctionnalités
 - ❏ [w3schools.com](#) 
- ❏ De nombreux **articles** aidant à la compréhension de certains **concepts** et **notions** de programmation
 - ❏ [alsacreations.com](#) 

Les sources

- ❏ Un **forum** de **développeurs** où trouver de nombreuses **réponses** à des questions **théoriques** ou **techniques**

- ❏ [stackOverFlow.com](https://stackoverflow.com) 

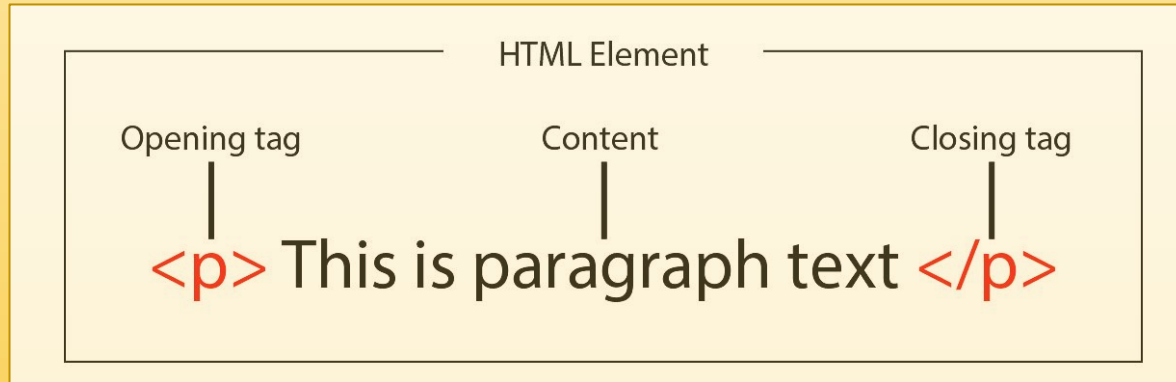
- ❏ Parce que la majorité des **problèmes** y ont probablement déjà été évoqués et des **solutions** apportées

- ❏ [Google](https://www.google.com)  

Comment s'écrit le HTML ?

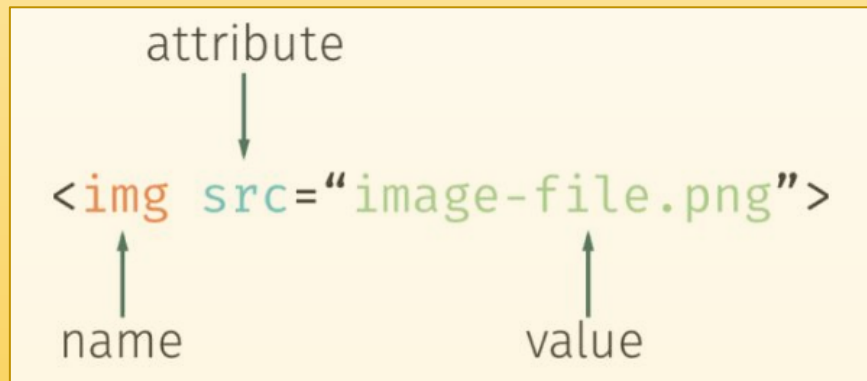
Syntaxe des balises

- ❑ Les balises **HTML** sont majoritairement composées d'une **balise** ouvrante `<p>` et d'une **balise** fermante `</p>` contenant chacune le **nom** de la balise
- ❑ Entre ces deux **balises** se trouve le **contenu** à afficher dans la **page web**. On appelle **élément** l'**ensemble** formé par les balises et leur contenu



Syntaxe des balises

- ❑ Les **balises** sont insensibles à la **casse** mais elles sont écrites en **minuscule** par convention
- ❑ Certaines **balises** appelées **orphelines** ne possèdent pas de **balise fermante** : ``
- ❑ Les **balises** peuvent posséder des **attributs** et leurs **valeurs** qui sont alors placés dans la **balise ouvrante**



Syntaxe des balises

- ❑ Une **balise** particulière permet d'écrire des **commentaires** dans son fichier **HTML**. Tout ce qui se trouve à l'intérieur de cette **balise** ne sera pas affiché par le **navigateur** dans la **page web**

<!-- ne s'affiche pas dans la page web -->

- ❑ Les **commentaires** seront cependant visibles dans le **code source**, il est donc recommandé de ne jamais y écrire de données sensibles !
- ❑ Les **commentaires** permettent de **renseigner** ou **expliquer** son **code** pour ses collaborateurs mais également pour soi-même
- ❑ Des portions de **code** peuvent aussi être **désactivées** en les plaçant entre ces **balises commentaires**

Fonction des balises

- ❑ Les **balises** définissent le rôle **sémantique** de chaque **élément** d'une **page web** selon sa catégorie :
titres, paragraphes, listes, liens, images, audios...
- ❑ Une **balise** est donc choisie en fonction du **contenu** qui lui est attribué. On ne peut cependant pas créer de **balise** soi-même : ~~<intro>~~
- ❑ Les **balises** peuvent aussi indiquer au **navigateur** le degré d'**importance** d'un mot ou d'un **ensemble** de mots :
`<mark> ≠ <i> <u>`
- ❑ <https://www.alsacreations.com/strong-b-em-i-quelle-balise-utiliser.html>

Fonction des balises

`<p>` Au sein d'un même paragraphe, certains mots peuvent être plus `` importants `` que d'autres. Certains vont demander `` une attention plus particulière ``.
Et certaines notions auront besoin d'une mise en lumière plus `<mark>` importante `</mark>`



Au sein d'un même paragraphe, certains mots peuvent être plus **importants** que d'autres. Certains vont demander *une attention plus particulière*. Et certaines notions auront besoin d'une mise en lumière plus **importante**

Fonction des balises

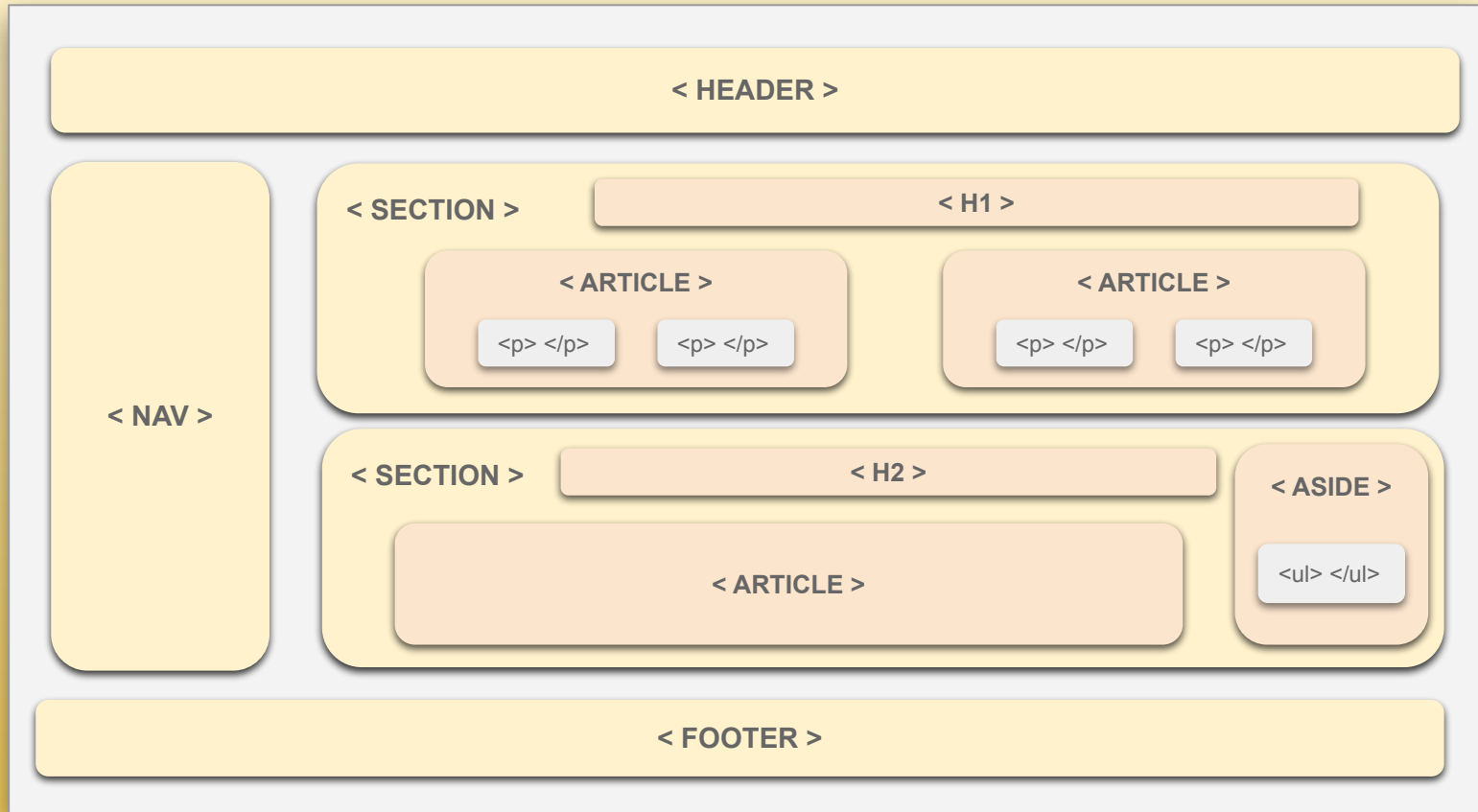
- ❑ Des **balises** peuvent être de type **container** et sont utilisées pour **organiser** son code. On parle ici d'organisation **sémantique** des éléments et non de leur **mise en page** qui sera elle réalisée avec le **CSS**
- ❑ Les **balises containers** sont nombreuses et permettent de **renseigner** au mieux le **navigateur** sur le **rôle** donné à chaque **contenu** au sein d'une **page web**

`<main> <section> <header> <footer> <article> <nav> <aside>`

- ❑ Certaines **balises containers** sont utilisées pour **sélectionner** et/ou **regrouper** des **éléments** auxquels on souhaite appliquer un **style** bien précis. Elles n'ont alors aucune valeur **sémantique** particulière

`<div> `

Fonction des balises



Fonction des balises

01 Smashing Magazine

nav

Articles Guides Books Workshops Membership Job Board Newsletter


Search articles...

header

Accessibility CSS **JavaScript** React Vue Guides UX Design Web Design Figma Wirepapers Business Career Privacy Jump

header

article



Ahmad Shadeed wrote

Using HSL Colors In CSS h2


JUL 5, 2023 in CSS, Colors, Tools

time

In this article, Ahmad Shadeed explains what HSL is, how to use it, and shares some of the useful use-cases and examples that you can use right away in your current projects.

[Continue reading »](#)


article



Andriy Zaslavskiy in Email, Tools, User Experience

Email Testing Flow As It Should Be


article



Ramona Schwering in UI, Tools, Testing

It's A (Front-End Testing) Trap! Six

main



Email Newsletter

Useful front-end & UX tips, delivered once a week.

With tools to help you get your work done better. Subscribe and get Vitaly's [Smart Interface Design Checklists PDF](#) — in your inbox.

On front-end & UX. Powered by spore.js.

- <header>
- <nav>
-
- <form>
- <article>
- <h2>
- <time>
- <main>

George Moller
_georgemoller

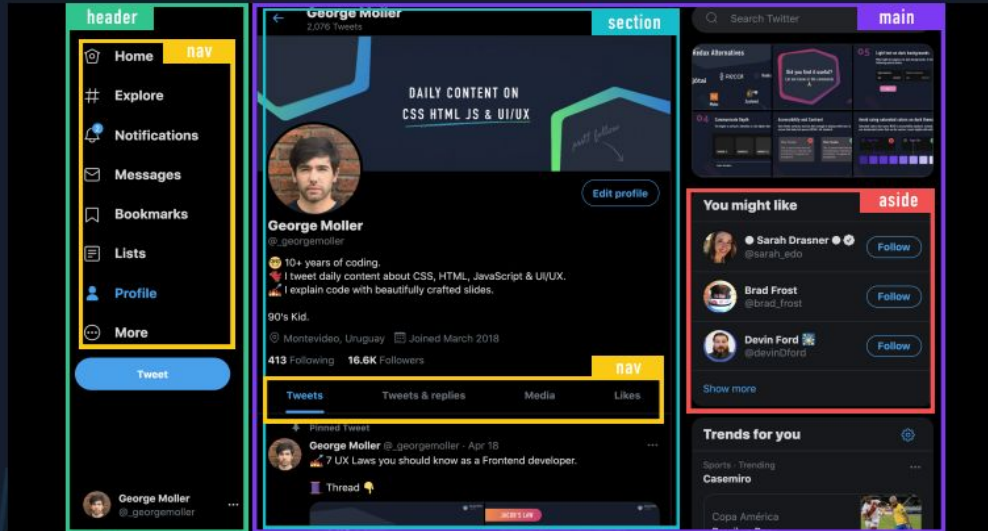
22

Fonction des balises

02 Twitter



George Moller
_georgemoller



- <header>
- <nav>
- <main>
- <aside>
- <section>

Fonction des balises



Celine Kafefrais

8 rue du fort de france

972 saint Pierre

06 23 78 65 12

Créative - Bon saucier

CHEF DE CUISINE

FORMATION

- ★ Diplôme de chef de cuisine - Académie du bon gout - Paris (75)
- ★ Diplôme de commis de cuisine - Lycée de la poivrière - Paris (75)
- ★ Diplôme baccalauréat professionnel de cuisinier - Lycée de la poivrière - Paris (75)
- ★ CAP cuisine - Lycée des dents rouges - Fort de france - Martinique (972)

COMPETENCES

- Elaborer et créer des cartes et des menus
- Préparer, confectionner et dresser des plats
- Réaliser des recettes à partir de fiches techniques
- Vérifier les préparations et les plats (goût, qualité, présentation)
- Evaluer la qualité des produits de base
- Encadrer une équipe de cuisine
- Organiser et entretenir le poste de travail et les équipements



CELINE KAFEFRAIS

8 rue du fort de france

972 saint Pierre

06 23 78 65 12

Créative - Bon saucier

FORMATION

- ★ Diplôme de chef de cuisine - Académie du bon gout - Paris (75)
- ★ Diplôme de commis de cuisine - Lycée de la poivrière - Paris (75)
- ★ Diplôme baccalauréat professionnel de cuisinier - Lycée de la poivrière - Paris (75)
- ★ CAP cuisine - Lycée des dents rouges - Fort de france - Martinique (972)

COMPETENCES

- Elaborer et créer des cartes et des menus
- Préparer, confectionner et dresser des plats
- Réaliser des recettes à partir de fiches techniques
- Vérifier les préparations et les plats (goût, qualité, présentation)
- Evaluer la qualité des produits de base
- Encadrer une équipe de cuisine
- Organiser et entretenir le poste de travail et les équipements
- Gérer les commande (achats, contrôles, relances) auprès des fournisseurs
- Contrôle comptable et bilan alimentaire

EXPERIENCES

CHEF DE CU

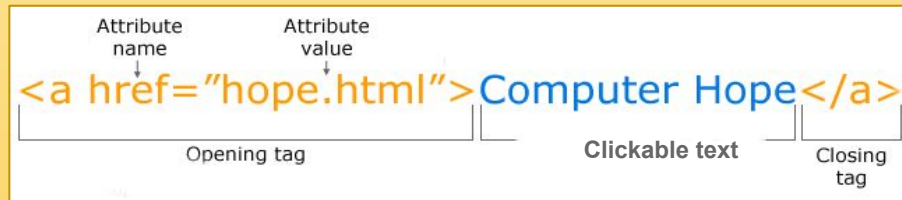
Les principales balises

- ❑ Des **titres** de plus ou moins grande **importance** sont représentés par les balises **heading**

`<h1> </h1>` → `<h6> </h6>`

- ❑ Les **paragraphes** sont eux représentés par la balise `<p> ... </p>`

- ❑ La balise `<a> ... ` permet d'intégrer des **liens** internes ou externes au sein d'une **page web**



- ❑ Cette **balise** dispose d'un **attribut** précisant au **navigateur** où chercher le document

Les principales balises

- ❑ Certaines **balises** ont un contenu dont le **rôle** est principalement **sémantique** comme `<hr/>` et `
`
- ❑ La balise `<hr/>` - horizontal rule - affiche donc une **ligne horizontale** et sa fonction est d'indiquer au **navigateur** une **séparation sémantique** entre deux contenus
- ❑ La balise `
` - break - va créer un **saut de ligne**. Sa fonction est d'indiquer au **navigateur** une **rupture sémantique** au sein d'un bloc de texte
- ❑ Le **rendu** affiché par le **navigateur** de ces deux **balises** est entièrement **modifiable** à l'aide du **CSS**
- ❑ Ces **balises** ne doivent donc être en aucun cas **utilisées** pour un **effet de style**. Le même **visuel** peut d'ailleurs être obtenu avec le **CSS**

Les principales balises

- ❑ Plusieurs types de **listes** sont disponibles en **HTML** : les listes **ordonnées**, les listes **non-ordonnées** et les listes **descriptives**

` ... `

` ... `

`<dl> ... </dl>`

- ❑ Les listes **ordonnées** et **non-ordonnées** contiennent des **list items** contenant les différents éléments énumérés ou listés

`` élément de liste ``

- ❑ Quant aux listes **descriptives**, elles contiennent des **description terms** et des **description details**

`<dt>` terme à décrire `</dt>`

`<dd>` définition du terme `</dt>`

Les principales balises

```
<ul>
```

```
<li> Avocats </li>
```

```
<li> Fruits secs </li>
```

```
<li> Lait d'amande </li>
```

```
</ul>
```



- Avocats
- Fruits secs
- Lait d'amande

```
<ol>
```

```
<li> Epluchez les avocats </li>
```

```
<li> Concassez les fruits secs </li>
```

```
<li> Ajouter le lait d'amande </li>
```

```
<li> Passez le tout au mixer </li>
```

```
</ol>
```



1. Epluchez les avocats
2. Concassez les fruits secs
3. Ajouter le lait d'amande
4. Passez le tout au mixer

Les principales balises

- ❑ Les **listes** peuvent être **imbriquées** les unes dans les autres qu'elles soient **ordonnées** ou **non-ordonnées**

```
<ol>
  <li> Les ingrédients
    <ul>
      <li> Avocats </li>
      <li> Fruits secs </li>
      <li> Lait d'amande </li>
    </ul>
  </li>
  <li> La recette
    <ol>
      <li> Epluchez les avocats </li>
      <li> Concassez les fruits secs </li>
      <li> Ajouter le lait d'amande </li>
      <li> Mixez le tout au mixer </li>
    </ol>
  </li>
</ol>
```



1. Les ingrédients
 - Avocats
 - Fruits secs
 - Lait d'amande
2. La recette
 1. Epluchez les avocats
 2. Concassez les fruits secs
 3. Ajouter le lait d'amande
 4. Mixez le tout au mixer

Les principales balises

- ❑ Le **navigateur** affiche un **alinéa** devant la **définition** contenue dans une **liste descriptive**

```
<dl>  
  <dt> HTML </dt>  
  <dd> HyperText Markup Language </dd>  
</dl>
```



HTML
HyperText Markup Language

- ❑ Un autre type de **description** est disponible avec la balise **<abbr>** qui possède un **attribut** dont la valeur apparaît dans un **encart** lors du survol de la **souris**

```
<p> Découvrons le  
  <abbr title="World Wide Web"> WWW </abbr>  
</p>
```



Découvrons le WWW



World Wide Web

Les principales balises

- ❑ Une **balise orpheline** permet d'afficher une **image** au sein de notre **page web** : la balise ``
Elle possède un **attribut** permettant d'indiquer au **navigateur** où trouver cette image

```

```



- ❑ Des **attributs** sont également disponibles pour préciser **largeur** et **hauteur** de l'**image**. Mais il sera préférable de la **redimensionner** avant pour **optimiser** le temps de **chargement** de la page

```
<img height="100px" width="200px" />
```

Les principales balises

- ❑ La balise `` dispose d'un **attribut** permettant de **décrire** l'image à afficher dans la **page web**. Cet attribut doit être **renseigné** pour plusieurs raisons :
- ❑ La **fonction** première de cet **attribut** est de proposer une **alternative** à afficher si, pour une raison ou une autre, l'**image** n'a pu être **chargée** par le navigateur
- ❑ Cet **attribut** permet également aux **lecteurs d'écran** de pouvoir **décrire** à l'utilisateur l'**image** qui est affichée à l'écran
- ❑ Cet **attribut** joue aussi un **rôle** important dans le **référencement** d'une page web, si l'**image** affichée a une **importance** particulière

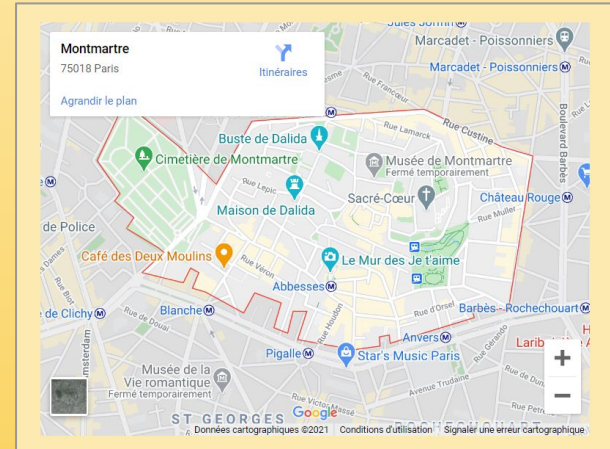
```

```


Les principales balises

- ❑ La balise `<iframe> ... </iframe>` permet d'afficher en miniature un élément de **site web** comme une vidéo **youtube** ou une carte de **google maps**
- ❑ La **balise** complète est généralement **disponible** sur le **site** concerné en cliquant sur **Partager** puis **Intégrer**

```
<iframe  
src="https://www.google.com/maps/embed?pb=!1m18!1m  
886215171986976!2m3!1f0!2f0!3f0!3m2!1i1024!2i768!4f1  
12!1m3!1d5247.2sfr" width="600" height="450">  
</iframe>
```



Les principales balises

- ❑ De nombreux **attributs** sont présents **par défaut** dans cette **balise**. Leur rôle concerne majoritairement les **règles** de fonctionnalités de l'**API** utilisée, comme celle de **youtube**

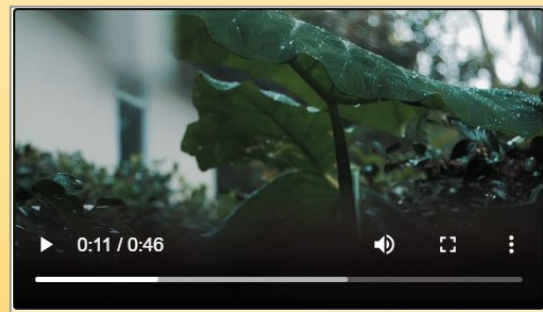
```
<iframe width="560" height="315"
src="https://www.youtube.com/embed/RNxnk
NP-2ho" title="YouTube video player"
frameborder="0" allow="accelerometer;
autoplay; clipboard-write; encrypted-media;
gyroscope; picture-in-picture" allowfullscreen>
</iframe>
```



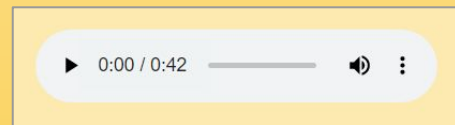
Les principales balises

- ❏ Des **balises** sont également disponibles pour intégrer un contenu **vidéo** ou **audio**. Elles possèdent un **attribut** indispensable pour l'**affichage** et la **lecture** du **media** : **controls**

```
<video src="raining.mp4" controls />
```



```
<audio src="rain-and-storm.mp3" controls />
```



Un code propre

- ❑ De nombreuses **balises** pouvant être imbriquées les unes dans les autres, il est **impératif** de garder un **code** le plus clair possible :
- ❑ L'ordre d'**ouverture** et de **fermeture** des **balises** doit donc être rigoureusement respecté : toujours fermer les **balises** contenues avant de fermer les **balises** contenant

The diagram illustrates the correct nesting of HTML tags. The code snippet is: `<div> <p> some <mark> text here </mark> and here </p> </div>`. The closing tags are color-coded: `</mark>` is green, `</p>` is blue, and `</div>` is red. Brackets below the code show the nesting: a green bracket connects `<mark>` and `</mark>`, a blue bracket connects `<p>` and `</p>`, and a red bracket connects `<div>` and `</div>`. The brackets are nested, with the green one being the innermost and the red one being the outermost, demonstrating that inner tags are closed before outer tags.

```
<div> <p> some <mark> text here </mark> and here </p> </div>
```

Les principales balises

- ❏ Une **balise** permet d'afficher un **bouton** auquel attribuer une **action** précise qui sera souvent gérée avec le langage **JavaScript**

`<button> Cliquez ici </button>`



Cliquez ici

- ❏ Cependant il est possible de placer une **balise lien** à l'intérieur même de cette **balise**, dans la zone où **cliquer**, ou bien placer le **bouton** dans un **lien**

```
<button>
  <a href="https://recette-jus-avocat-fruits-secs">Accédez à la recette </a>
</button>
```

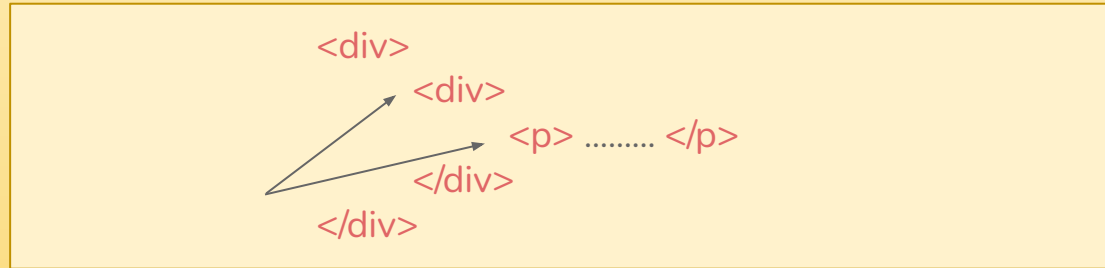
```
<a href="https://recette-jus-avocat-fruits-secs">
  <button> Accédez à la recette </button>
</a>
```

Un code propre



Un code propre

- ❑ Une bonne **indentation** permet également de produire un **code** bien aéré et lisible : cela consiste à opérer un **retrait** lors de l'**imbrication** de deux **balises**



- ❑ Pensez également à **espacer** les différentes parties de votre **code** pour le rendre moins **compact**
- ❑ Le choix d'une **balise** est un choix purement **sémantique**, quel que soit le visuel rendu par le **navigateur** de certaines **balises**. Le **style** est généré uniquement avec le **CSS**

Un code propre

```
<section class="zone_centrale">
```

```
  <aside class="aside">
    <p>CHEF DE CUISINE</p>
  </aside>
```

```
  <div class="cv">
```

```
    <h2>FORMATION</h2>
```

```
    <ul class="diplome">
```

- ★ Diplôme de chef de cuisine - Académie du bon gout - Paris (75)
- ★ Diplôme de commis de cuisine - Lycée de la poivrière - Paris (75)
- ★ Diplôme baccalauréat professionnel de cuisinier - Lycée de la poivrière - Paris (75)
- ★ CAP cuisine - Lycée des dents rouges - Fort de france - Martinique (972)

```
  </ul>
```

```
  <h2>COMPETENCES</h2>
```

```
  <ul>
```

- ➡ Elaborer et créer des cartes et des menus
- ➡ Préparer, confectionner et dresser des plats
- ➡ Réaliser des recettes à partir de fiches techniques
- ➡ Vérifier les préparations et les plats (goût, qualité, présentation)
- ➡ Evaluer la qualité des produits de base
- ➡ Encadrer une équipe de cuisine
- ➡ Organiser et entretenir le poste de travail et les équipements
- ➡ Gérer les commande (achats, contrôles, relances) auprès des fournisseurs
- ➡ Contrôle comptable et bilan alimentaire

```
  </ul>
```

```
  <h2>EXPERIENCES</h2>
```

```
  <ul>
```

- ★ 2018 - 2020 : chef de cuisine - restaurant A L'ACCRA (Paris)
- ★ 2015 - 2018 : chef de cuisine - restaurant AU POISSON FRAIS (Saint Ouen)
- ★ 2010 - 2015 : sous-chef de cuisine - restaurant O PAYS (Paris)
- ★ 2008 - 2010 : sous-chef de cuisine - restaurant LA CHALETTE CREOLE (Paris)

```
  </ul>
```

```
  <h2>LOISIRS</h2>
```

```
  <div class="loisirs">
```

```
    <div>
```

```
      <h4>➡ realisation de blogs culinaire</h4>
```

```
      <ul>
```

- tatie maryse
- je cuisine créole

```
      </ul>
```

```
    </div>
```

```
    <div>
```

```
      <h4>➡ randonnees et trek</h4>
```

```
      <ul>
```

- Île de La réunion - La tour des trois cirques (2019)
- Etats-Unis - Les zions narrows (2018)
- Guyane et sa forêt vierge (2016)
- GR20 - Corse (2015)

```
      </ul>
```

```
    </div>
```

```
</div>
```

```
</section>
```



```
<section class="zone_centrale">
```

```
  <aside class="aside">
    <p>CHEF DE CUISINE</p>
  </aside>
```

```
  <div class="cv">
```

```
    <h2>FORMATION</h2>
```

```
    <ul class="diplome">
```

- ★ Diplôme de chef de cuisine - Académie du bon gout - Paris (75)
- ★ Diplôme de commis de cuisine - Lycée de la poivrière - Paris (75)
- ★ Diplôme baccalauréat professionnel de cuisinier - Lycée de la poivrière - Paris (75)
- ★ CAP cuisine - Lycée des dents rouges - Fort de france - Martinique (972)

```
  </ul>
```

```
  <h2>COMPETENCES</h2>
```

```
  <ul>
```

- ➡ Elaborer et créer des cartes et des menus
- ➡ Préparer, confectionner et dresser des plats
- ➡ Réaliser des recettes à partir de fiches techniques
- ➡ Vérifier les préparations et les plats (goût, qualité, présentation)
- ➡ Evaluer la qualité des produits de base
- ➡ Encadrer une équipe de cuisine
- ➡ Organiser et entretenir le poste de travail et les équipements
- ➡ Gérer les commande (achats, contrôles, relances) auprès des fournisseurs
- ➡ Contrôle comptable et bilan alimentaire

```
  </ul>
```

```
  <h2>EXPERIENCES</h2>
```

```
  <ul>
```

- ★ 2018 - 2020 : chef de cuisine - restaurant A L'ACCRA (Paris)
- ★ 2015 - 2018 : chef de cuisine - restaurant AU POISSON FRAIS (Saint Ouen)
- ★ 2010 - 2015 : sous-chef de cuisine - restaurant O PAYS (Paris)
- ★ 2008 - 2010 : sous-chef de cuisine - restaurant LA CHALETTE CREOLE (Paris)

```
  </ul>
```

```
  <h2>LOISIRS</h2>
```

```
  <div class="loisirs">
```

```
    <div>
```

```
      <h4>➡ realisation de blogs culinaire</h4>
```

```
      <ul>
```

- tatie maryse
- je cuisine créole

```
      </ul>
```

```
    </div>
```

```
    <div>
```

```
      <h4>➡ randonnees et trek</h4>
```

```
      <ul>
```

- Île de La réunion - La tour des trois cirques (2019)
- Etats-Unis - Les zions narrows (2018)
- Guyane et sa forêt vierge (2016)
- GR20 - Corse (2015)

```
      </ul>
```

```
    </div>
```

```
</div>
```

```
</section>
```



Structure d'une page HTML

Structure globale

- ❑ La **structure** d'une page **HTML** repose sur quatre **balises** principales :
- ❑ La **première** indique au **navigateur** qu'il s'agit d'un **document** de type **HTML**
`<!DOCTYPE html>`
- ❑ La **deuxième** représente la **racine** du **document HTML** et englobe les **deux balises** suivantes
`<html> </html>`
- ❑ La **troisième** contient les **métadonnées** générales du **document** utiles à son **fonctionnement**
`<head> </head>`
- ❑ La **dernière** représente le **corps** de notre **page web** et contient donc son **contenu principal**
`<body> </body>`

Structure globale

```
<!DOCTYPE html>

<html lang="fr">

<head>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title> Nom de la page </title>
  <link rel="stylesheet" href="css/style.css" />
</head>

<body>

</body>

</html>
```

La balise <head>

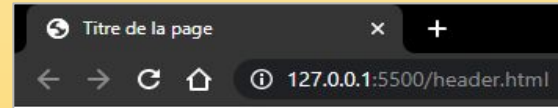
- ❑ Dans la balise <head> se trouvent toutes les données adressées au navigateur et nécessaires à l'affichage et au bon fonctionnement et de la page web
- ❑ Les balises <link> sont notamment présentes pour établir des liens avec les ressources externes permettant de gérer le style de la page web
- ❑ La balise <style> peut être présente pour déclarer des règles de style CSS devant être appliquées à des éléments de la page HTML
- ❑ Les balises <script> permettent d'assurer l'exécution d'un script, généralement JavaScript, à l'intérieur de notre page web

< script src="mon-script.js" ></script>

La balise <head>

- ❑ Les balises <meta> contiennent toutes les autres **métadonnées** ne pouvant être communiquées par le biais de l'une des autres **balises**
- ❑ La balise <title> contient le **titre** de la **page** qui apparaîtra dans l'**onglet** ou les **favoris** du **navigateur**, et également dans les **résultats** de recherche **Google**

<title> titre de la page </title>



- ❑ La balise <base> permet d'établir une **base** aux différentes **URL** utilisées au sein de la même **page HTML**
`<base href="http://www.base-url.com/">`

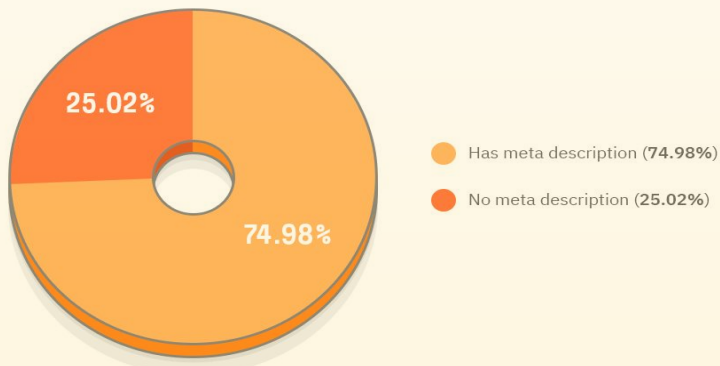
Les balises <meta>

- ❑ Seuls les robots de recherche ou les navigateurs peuvent lire les balises <meta>. Elles contiennent des données servant à l'optimisation du rendu ou du fonctionnement d'une page web
- ❑ Une balise <meta> peut servir à indiquer le système d'encodage des caractères utilisés dans la page web même si certains navigateurs comme Chrome corrigent automatiquement les encodages incorrects
`<meta charset="utf-8" />`
- ❑ Une balise <meta> possède généralement un attribut name et un attribut content dont les valeurs diffèrent selon l'utilisation prévue de ses données
- ❑ Cette balise <meta> contient des données permettant d'optimiser l'affichage sur les écrans de smartphones :
`<meta name="viewport" content="width=device-width, initial-scale=1.0" />`

Les balises <meta>

25.02% of Top-ranking Pages Don't Have a Meta Description

Based on a study of 192,656 pages ranking in the top 10.

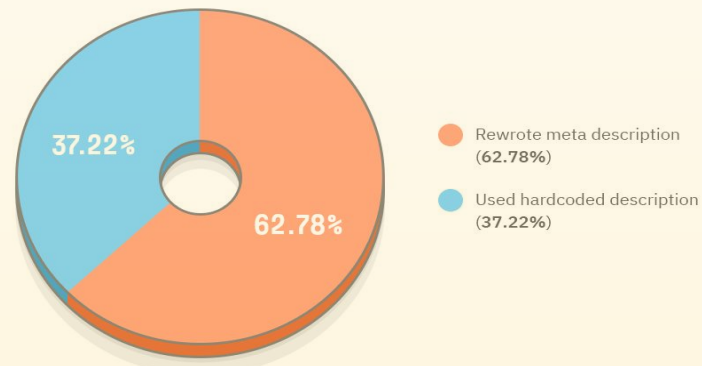


© <https://ahrefs.com/blog/meta-description-study/>

ahrefs

Google Rewrites Meta Descriptions 62.78% of the Time

Based on a study of 192,656 pages ranking in the top 10.



© <https://ahrefs.com/blog/meta-description-study/>

ahrefs

Les balises <meta>

- ❑ Les **données** de certaines balises <meta> concernent le **SEO** ou **référencement naturel**, c'est à dire la façon dont une **page web** sera **référéncée** par **les moteurs de recherche**

- ❑ Une **balise** fréquemment **utilisée** dans ce domaine indique la **description** que l'on retrouve sous le **résultat** dans les **moteurs de recherche**

`<meta name="description" content="court descriptif de la page web" />`

- ❑ Parmi les balises <méta> essentielles en matière de **SEO** se trouvent les **métadonnées de robots** utilisées pour donner des **directives** aux **moteurs de recherche** sur la façon d'**étudier** les **liens** sur une **page web**

`<meta name="robots" content="..." />`

- ❑ Après les récentes mises à jour des **algorithmes de Google**, l'utilisation de certaines balises <meta> en matière de **référencement naturel** est devenue presque inutile, comme celle-ci par exemple :

`<meta name="keywords" content="..." />`

Les balises <meta>

- ❑ Un **article** intéressant abordant la **notion** du **viewport** aujourd'hui **indispensable** dans le **responsive design**
 - ❑ <https://www.alsacreations.com/Comprendre-le-Viewport-dans-le-Web-mobile.html>
- ❑ Plus de **précisions** concernant le vaste sujet des balises <meta> pour vous aider un peu à **choisir...**
 - ❑ <https://www.alsacreations.com/À-quoi-servent-les-balises-meta.html>
 - ❑ <https://www.ionos.fr/les-principales-balises-meta-et-leurs-fonctions>
- ❑ Une **présentation** des principales balises **meta-robots** et de leur **utilisation** par les **moteurs de recherche**
 - ❑ <http://robots-txt.com/meta-robots/>

Les balises <link>

- ❑ Les balises <link> assurent le lien avec les ressources externes utilisées pour la gestion du style
- ❑ Ces balises possèdent un attribut précisant au navigateur l'URL de la ressource. Cette URL peut être absolue ou relative

`<link href="nom-fichier.css" />` [URL relative]

`<link href="https://fonts.googleapis.com/family=Bellota" />` [URL absolue]

- ❑ Un attribut permet d'indiquer le type de relation existant entre la ressource et le fichier HTML

`<link rel="stylesheet" href="nom-fichier.css" />`

- ❑ Un attribut permet de simplement préciser le type de la ressource liée au fichier HTML

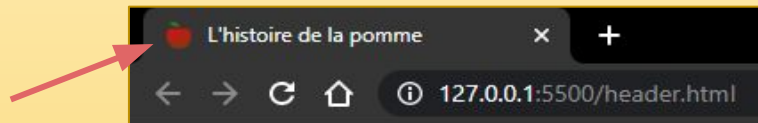
`<link rel="stylesheet" href="nom-fichier.css" type="text/CSS" />`

- ❑ <https://www.seoh.fr/outils-tutos-astuces-seo/structure-url-absolue-relative.html>

Les balises <link>

- ❑ La **ressource** avec laquelle un **lien** est **établi** peut-être une **favicon** : une **petite icône** qui apparaît sur la **gauche** de la **barre d'adresse** et qui participe à l'**image** du **site web**

```
<link rel="shortcut icon" href="favicon.ico" type="image/x-icon" />
```



- ❑ Un **lien** peut également être **établi** avec une **bibliothèque d'icônes** accessibles **en ligne** via un **CDN**

```
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/font-awesome/all.min.css" />
```



Les balises <link>

- ❑ La librairie d'icônes la plus largement utilisée est FontAwesome. Le lien à inclure dans la balise <link> est disponible sur <https://cdnjs.com/>
- ❑ Les icônes sont intégrées via une balise <i> ou possédant un attribut class pour pouvoir l'identifier, il suffit alors simplement de copier cette balise depuis la librairie

<i class="fas fa-hamburger"></i> →



- ❑ D'autres librairies comme material.io proposent des fichiers png ou svg à télécharger et il ne reste plus qu'à l'utiliser dans une simple balise

 →



Les balises <link>

- ❑ La **ressource externe** au fichier **HTML** peut être une **bibliothèque de polices accessibles en ligne**

```
<link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Bangers" />
```

ATTENTION : ÉDITION SPÉCIALE !!

- ❑ Une **librairie externe** comme **Bootstrap** peut également être **appelée** depuis le fichier **HTML**

```
<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap.min.css"/>
```

- ❑ Un **attribut** permet aussi de préciser le **support** sur lequel la **ressource** doit être exclusivement **utilisée**, comme des **tailles d'écran** spécifiques ou une **utilisation** particulière comme **l'impression**

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

Analysons les balises

Les éléments de type **block** ou **inline**

- ❑ Les éléments **HTML** sont de type **block** ou **inline**. Selon le **type** auquel ils appartiennent, les **éléments** n'occuperont pas le même **espace** au sein d'une **page web**
- ❑ Les **éléments** de type **block** occupent toute la **largeur** disponible de l'**écran** ou de leur **conteneur** et imposent un **retour à la ligne** aux éléments suivants. Leur **taille** peut être **modifiée**
- ❑ Les **éléments** de type **inline** n'occupent que la **largeur** de leur **contenu**. D'autres **éléments** de type **inline** peuvent donc se retrouver **à côté**. Leur **taille** ne peut pas être **modifiée**

Elément de type **block**

Elément de type **inline**

Elément de type **inline**

Les éléments de type **block** ou **inline**

- ❑ Excepté les balises **containers**, les **éléments** de type **block** ne peuvent pas contenir d'autres **éléments** de type **block** mais ils peuvent contenir des **éléments** de type **inline**

un paragraphe **block** contenant

un autre paragraphe **block**

paragraphe **block** contenant une balise **inline** à l'intérieur

Les éléments de type **block** ou **inline**

- ❑ Tous les **titres** du `<h1>` au `<h6>`, les **paragraphes**, comme tous les **éléments** de **liste**, sont de type **block**
- ❑ L'ensemble des **balises-containers** sont également de type **block**, excepté la balise `` qui est de type **inline**
- ❑ Les **liens** `<a>` et les balises sémantiques `<mark>` `` `` sont des éléments de type **inline**
- ❑ Les balises multimédia `` `<iframe>` `<audio>` `<video>` `<map>` et les **boutons** sont de type **inline-block** : elles n'occupent que la **largeur** de leur **contenu** mais leurs **dimensions** peuvent être **modifiées**
- ❑ Le **CSS** nous donnera la possibilité de **changer** le **type** des **éléments** selon les besoins de la **mise en page**

Les attributs et leurs valeurs

- ❑ Chaque élément **HTML** peut disposer d'un ou **plusieurs attributs** auxquels sont données des **valeurs**. Ces **attributs** vont permettre de **configurer** les éléments ou bien d'adapter leur **comportement**
- ❑ Avec l'**évolution** progressive du **CSS**, beaucoup d'**attributs** qui étaient disponibles en **HTML** sont aujourd'hui **obsolètes** et sont maintenant des **propriétés** utilisées en **CSS**
- ❑ Des **attributs** appelés attributs **universels** peuvent être utilisés pour n'importe quel **élément HTML**
- ❑ D'autres **attributs** sont quant à eux **spécifiques** et sont **réservés** à certaines **balises HTML**

```
<p style=" " class=" " id=" " > ... </p>
```

```
<video poster=" "> </video> <ol type=" "> ... </ol>
```

Les attributs et leurs valeurs

- ❑ La **valeur** de certains **attributs** peut être un **booléen** [une **valeur** ne pouvant être que **true** ou **false**]
- ❑ Le simple fait que cet **attribut** soit **présent** suffit à lui donner la **valeur true**, tout comme son **absence** suffit à lui donner la **valeur false**

`<audio controls > ... </audio>`
`<audio controls="controls" > ... </audio>`
`<audio controls=" " > ... </audio>` } \neq `<audio> ... </audio>`

- ❑ <https://meiert.com/en/blog/boolean-attributes-of-html/>

Les attributs et leurs valeurs

- ❑ Parmi les **attributs universels** le plus souvent utilisés, celui permettant d'**afficher** une **infobulle** lors du **passage** de la **souris** sur l'élément :

```

```



- ❑ Un **attribut universel** permet aussi de **masquer** un **élément** pour pouvoir le faire **apparaître** au moment voulu, avec **JavaScript** par exemple

```
<p hidden > Bienvenue cher client </p>
```

Les attributs et leurs valeurs

- ❑ Parmi les nouveautés de **HTML 5**, l'attribut universel **data-*** permet de **stocker** des **données** au sein d'un **élément** afin de pouvoir les **manipuler** ultérieurement avec le **CSS** ou **JavaScript**
- ❑ Le **nom** donné à cet **attribut-data** est choisi en fonction de type de **donnée** qui y est **stockée**
`<div data-name="John Doe" data-subject="John Doe Biography" > ... </div>`
- ❑ Les **robots de recherche** n'ont cependant pas **accès** à ces **données**, il faudra donc être **prudent** dans l'utilisation de cet **attribut** pour ne pas fragiliser le **SEO** du site web
- ❑ <https://www.alsacreations.com/article/lire/1397-attributs-data-et-api.html>
- ❑ <https://css-tricks.com/a-complete-guide-to-data-attributes/>

Les attributs et leurs valeurs

- ❑ Les **éléments** de type **block** ou **inline-block** dont les **dimensions** peuvent être **modifiées** disposent d'un **attribut** permettant de le faire

```
<img width="12rem" height="7rem" > ... </img>
```

- ❑ Les **éléments** nécessitant d'indiquer au **navigateur** une **URL** pour accéder à la **ressource** à utiliser disposent d'**attributs spécifiques**

```

```

```
<a href="http://www.site-a-visiter.com" > visitez leur site ici </a>
```

- ❑ Les **éléments** dont le rôle est d'**afficher** un contenu provenant d'une **ressource externe** possèdent un **attribut** permettant de proposer une **alternative** à afficher

```

```

Les attributs et leurs valeurs

- ❑ Un **attribut** permet également de **retarder** le **chargement** du contenu à **afficher** jusqu'au **moment** où le l'utilisateur **scrolle** jusqu'à cet élément

```

```

- ❑ Les **éléments** proposant l'accès à un **lien hypertexte** disposent d'**attributs spécifiques** permettant d'**ouvrir** ce **lien** dans un **nouvel onglet** et cela en toute **sécurité**

```
<a href="..." target="_blank" rel="noopener noreferrer" >visitez leur site ici </a>
```

- ❑ <https://blog.dareboost.com/fr/2017/03/target-blank-rel-noopener-securite-performance/>
- ❑ <https://developer.mozilla.org/fr/docs/Web/HTML/Attributes>

Les attributs et leurs valeurs

- ❑ L'attribut **media** permet de préciser des **media queries** au navigateur pour le **chargement** d'images
- ❑ Selon la **taille de l'écran** sur laquelle se trouve l'utilisateur, l'**image** ou le **format** de l'image pourront ne pas être les mêmes
- ❑ Les élément **HTML** utilisés pour cela sont les balises **<picture>** et également **<source>** auquel on ajoutera l'attribut **srcset** pour préciser quelle image **charger**

```
<picture>  
  <source media="(max-width:500px)" srcset="pink-flower.jpg">  
  <source media="(min-width:800px)" srcset="white-flower.jpg">  
    
</picture>
```


Attributs **id** et **class**

- ❑ Un **attribut** permet de **cibler** un **élément HTML** précis en lui donnant un **identifiant** personnalisé afin de pouvoir **interagir** avec lui

```
<h2 id="titre-cv"> ... </h2>
```

- ❑ Plusieurs **id différents** peuvent être **déclarés** dans une **page HTML**, mais un **même id** doit par contre être **unique** au sein d'une **page HTML**
- ❑ Cet **id** permet de **manipuler** de plusieurs façons l'**élément** ciblé : pour lui appliquer un **style** particulier à l'aide du **CSS** ou bien pour **modifier** son comportement avec **javaScript**
- ❑ Un **id** permet également de pouvoir **cibler** un **élément** à l'aide d'une **balise lien** au sein d'une **même page HTML** ou dans une **page différente**

```
<a href="#titre-cv"> Accédez au lien ici </a>
```

Attributs **id** et **class**

- ❑ L'attribut **class** permet de cibler un groupe d'éléments dans un document HTML en leur donnant le même nom de classe

```
<p class="nom-chapitre"> ... </p>
```

- ❑ Cet attribut permet d'interagir de la même manière avec tous les éléments possédant une même classe

```
<p class="emplois-cv"> ... </p>
```

```
<p class="emplois-cv"> ... </p>
```

- ❑ On pourra alors appliquer le même style à tous les éléments d'une même classe avec du CSS ou encore modifier leur comportement de la même façon avec du JavaScript
- ❑ Des éléments peuvent posséder la même classe dans une ou plusieurs pages HTML permettant ainsi de créer une identité visuelle commune entre les différentes pages d'un site web

Attributs id et class

- ❑ Une **classe** pour donner la même couleur **bleue** à deux **paragraphes**, une autre **classe** pour donner la même couleur **rouge** à deux autres **paragraphes** et un **id** unique pour donner un **style** au **titre** seulement

Les attributs class et id

Lorem ipsum dolor sit amet consectetur adipisicing elit. Laudantium eum earum nihil magni deserunt, ab sequi! Beatae ratione cum aut enim, temporibus minus numquam laudantium aliquam ea dolore veritatis dolores sed rem, molestias nihil voluptas adipisci nemo quia illum maxime!



Autem natus et dicta vel esse quasi sit corporis expedita nam accusantium. Sed repudiandae, odit soluta quo magnam aliquam obcaecati ipsam, at quisquam adipisci ut non assumenda corrupti laudantium molestias, tenetur fugit repellendus harum consectetur eligendi nesciunt maiores nemo. Harum?

Veniam nostrum consectetur vero? Ea sint animi repellendus dignissimos temporibus facere beatae eaque iste porro quae. Facere, pariatur natus voluptatum repellat aperiam necessitatibus soluta nesciunt delectus deserunt earum exercitationem quisquam a vel quis excepturi omnis quae ratione eligendi minima sapiente?

Iste rem voluptates commodi, dolorum autem quae, dolor ea tempore consequuntur harum aperiam cum, nulla ipsa! Libero animi, quae, placeat aperiam praesentium, atque architecto quod laboriosam nemo magnam non. Sequi eaque fuga voluptates est. Tempore voluptatem aperiam repellat quibusdam sequi!



Les liens hypertextes

- ❑ Les balises `<a>` contiennent des **liens hypertextes** dirigeant vers un autre **endroit** de la **même page** ou bien vers une **autre page** du même site ou d'un autre site
- ❑ Le **contenu** de cette **balise** est l'endroit où **cliquer** pour être **dirigé** vers la **page** ou le **lieu** de la **page** indiqué
` Accéder au sommaire `

- ❑ Ce **contenu** doit être le plus **précis** et **évocateur** possible, notamment pour les **lecteurs** de **page web**
` Cliquez ici ` 
- ❑ Un **lien** peut être **contenu** dans un autre **élément HTML** ou lui-même **contenir** un autre **élément HTML**, il suffit alors de **cliquer** sur cet **élément** pour être **dirigé** vers l'**endroit** décidé

` `

Les liens hypertextes

- ❑ Les liens peuvent par exemple renvoyer aux différentes pages d'un site web et constituer ainsi son menu de navigation

```
<ul>
  <li> <a href="home.html" > Home </a> </li>
  <li> <a href="about.html" > About </a> </li>
  <li> <a href="blog.html" > Blog </a> </li>
  <li> <a href="contact.html" > Contact </a> </li>
</ul>
```



- [Home](#)
- [About](#)
- [Blog](#)
- [Contact](#)

- ❑ Le style et la disposition des éléments peuvent être facilement modifiés avec quelques règles de CSS

Home About Blog Contact

Les liens hypertextes

- ❑ La **page HTML** qui est **ciblée** dans un **lien** doit être **située** dans le même **dossier** que la **page HTML** contenant ce **lien**. Sinon il faudra **se déplacer** dans l'**arborescence** des **fichiers**
- ❑ Pour pouvoir **remonter** d'un **niveau**, il faut faire **précéder** le fichier du symbole **.. /**

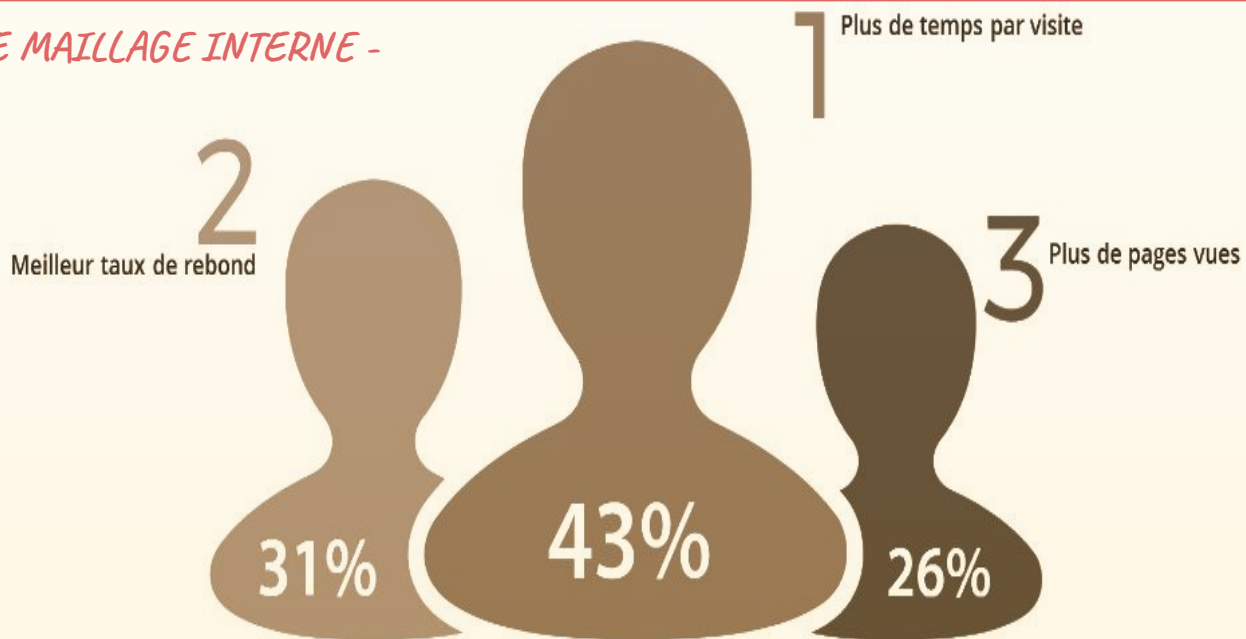
` ... `

- ❑ Les balises **<a>** sont également beaucoup **utilisées** lors du travail sur le **référencement naturel** d'un **site web**
- ❑ Une pratique **SEO** consiste en effet à **multiplier** au sein d'un **site web** les **liens** entre **différentes pages** dont le **contenu** appartient au même **champ sémantique**

❑ <https://fr.semrush.com/blog/bonnes-pratiques-maillage-interne/>

Les liens hypertextes

- LE MAILLAGE INTERNE -



Les liens hypertextes

- ❑ Un **lien** peut également **diriger** l'utilisateur vers un **endroit précis** à l'intérieur d'une **autre page** en précisant un **id**

```
<a href="autre-page.html#menu" > Accéder au Menu </a>
```

- ❑ L'utilisation de l'**attribut** permettant d'ouvrir le **lien** dans un **autre onglet** peut rendre le site **vulnérable** à certaines formes d'**attaques**. Pour **éviter** cela, il suffit d'ajouter l'attribut **rel** avec les valeurs suivantes :

```
<a target="_blank" rel="noopener noreferrer" > ... </a>
```

- ❑ Lorsque le **lien** propose un **ressource** à ouvrir comme une **image** ou un fichier **PDF**, il est possible de proposer **automatiquement** à l'utilisateur son **téléchargement**

```
<a href="menu-restaurant.pdf" download > Menu disponible ici </a>
```


Les liens hypertextes

- ❑ La balise `<a>` offre la possibilité de proposer à l'utilisateur l'envoi d'un email en ouvrant automatiquement sa boîte de messagerie par défaut

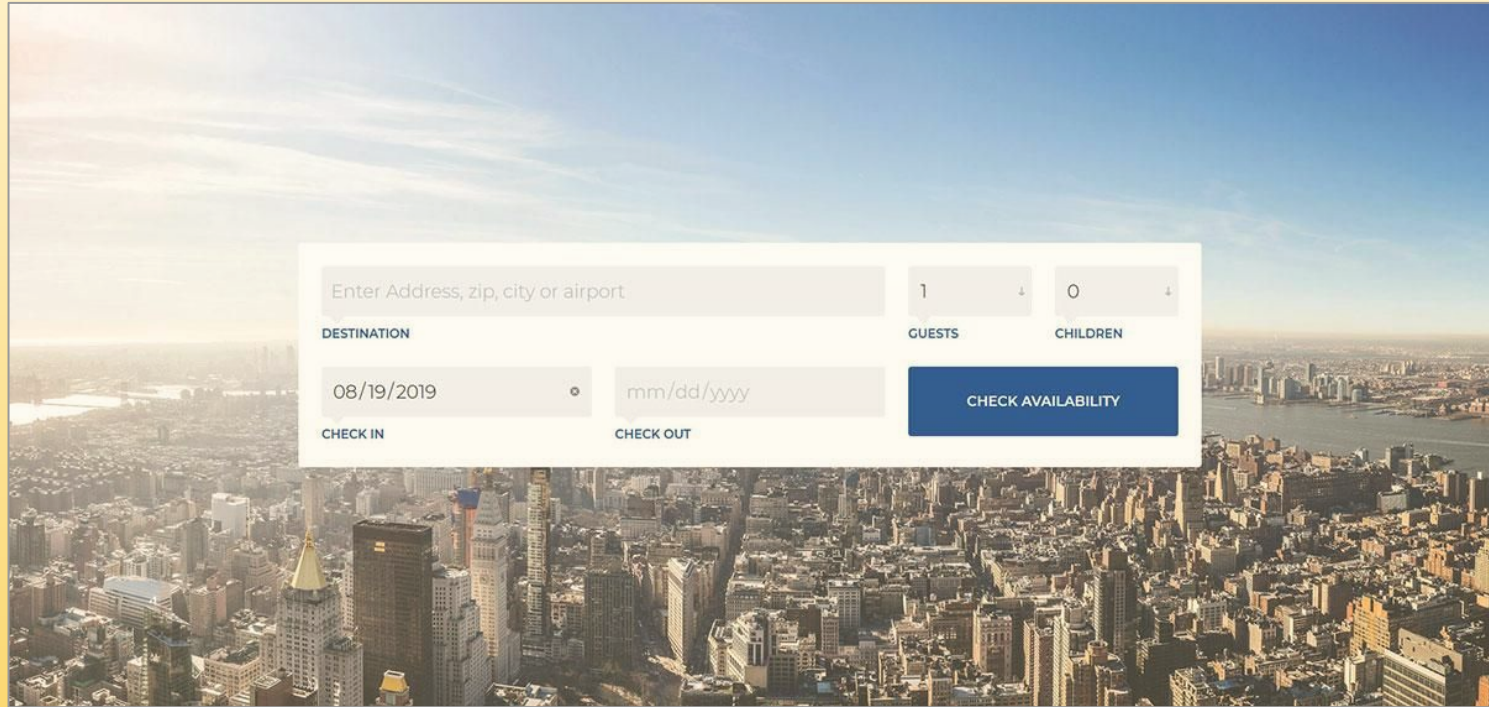
` Nous envoyer un mail `

- ❑ Des précisions peuvent alors être apportées quant au sujet du mail, l'objet du mail ou encore le nom des personnes à mettre en copie

` ... `

Cc: nobody;	
Cci:	
objet du mail	

Formulaires et tableaux



Enter Address, zip, city or airport

DESTINATION

08/19/2019

CHECK IN

mm/dd/yyyy

CHECK OUT

1

GUESTS

0

CHILDREN

CHECK AVAILABILITY

Structure générale

- ❑ Des **formulaires** sont disponibles en **HTML** avec la balise **<form>** permettant aux **visiteurs** d'un **site web** de **saisir** des **données** qui pourront être **stockées** et/ou **manipulées**
- ❑ Les **raisons** d'utiliser un **formulaire HTML** sont nombreuses : demande d'**inscription** ou de **connexion**, demande de **contact**, écriture de **commentaires**...



Structure générale

- ❑ La balise `<form>` possède deux **attributs** permettant de gérer le **fonctionnement** du **formulaire** et le **traitement** des **données** qui y seront saisies : **method** et **action**

- ❑ L'attribut **action** indique l'**URL** de la page qui va **traiter** les **données** grâce à un **langage serveur** comme le **PHP**

`<form action="form.php" > ... </form>`

- ❑ L'attribut **method** indique la **façon** dont seront **envoyées** les **données** du formulaire lors de la **requête HTTP** au serveur :

`<form method="get" > ... </form>` [directement ajoutées à l'URL]

`<form method="post" > ... </form>` [dans le corps de la requête HTTP]

- ❑ https://developer.mozilla.org/fr/docs/Learn/Forms/Sending_and_retrieving_form_data

Structure générale

- ❑ La **balise** principale d'un **formulaire** est la balise `<input>` qui permet à l'utilisateur de **saisir** des **données**. Son attribut **type** possède de nombreuses **valeurs** afin d'indiquer le **type** de données attendues

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



champ de saisie ...|

- ❑ Les **rendus** peuvent légèrement **différer** d'un **navigateur** à un autre selon le **type** de la **donnée** sélectionnée
- ❑ La balise `<label>` est généralement **associée** à la balise `<input>`. Elle lui sert alors d'**étiquette** et permet d'**informer** l'utilisateur sur le **type** de **donnée** qui est attendu

```
<label> Votre nom : </label>  
<input type="text" />
```



Votre nom :

Structure générale

- ❑ L'attribut **id** est utilisé pour **identifier** chaque balise `<input>` d'un formulaire. La même **valeur** est donnée à l'attribut **for** de la balise `<label>` permettant ainsi de **lier** les deux
- ❑ Un attribut **name** est également ajouté à la balise `<input>` pour pouvoir **identifier** et **manipuler** les **données** en **PHP** ou autre. La **valeur** qui lui est donnée est souvent la **même** que celle de l'**id**

```
<label for="age"> Votre âge : </label>  
<input type="text" id="age" name="age" />
```

- ❑ Le dernier **élément** nécessaire pour **envoyer** les **données** est un **bouton**. La balise `<input>` possède le **type** **submit** qui joue ce rôle et l'attribut **value** permet d'y ajouter du **texte**

```
<input type="submit" value="Envoyer" />
```



Envoyer

Les champs de saisie

- ❑ Les **champs de saisie** offrent la possibilité d'**afficher** un texte **par défaut** avec l'attribut **placeholder**. Ce texte **disparaît** dès que l'utilisateur commence à **écrire** dans le champ

```
<input type="text" placeholder="John Doe" />
```



- ❑ Le type **email** précise au **navigateur** que la **donnée** attendue est une **adresse mail**. Des **attributs** permettant de **guider** l'utilisateur lors de sa **saisie** sont disponibles.
- ❑ Le type **password** offre un **champ de saisie** dont le contenu apparaîtra **masqué** lors de la **saisie**

```
<input type="password" />
```



Les champs de saisie

- ❑ Les **champs de saisie** des **formulaire**s sont représentés avec la balise `<input>`. Grâce à l'attribut **type**, il est possible de préciser au navigateur le **type** des **données** attendues
- ❑ Le principal **type** est une zone de **texte monoligne**. Un champ de texte **multiligne** existe cependant avec la balise `<textarea>`
- ❑ Cette **balise** possède les attributs **cols** et **rows** pour modifier la **largeur** et la **hauteur** de la zone de saisie. Quelque soit sa taille, l'**utilisateur** peut la **redimensionner** à partir du coin inférieur droit

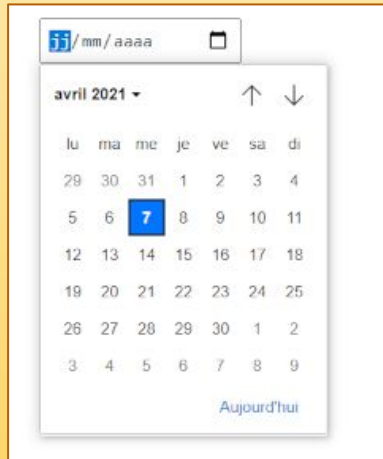


Votre commentaire :

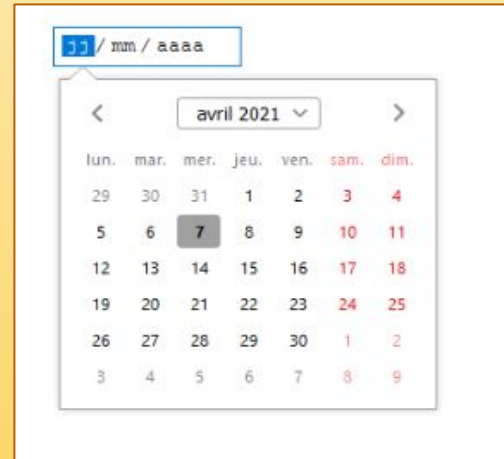
A red arrow points to the bottom-right corner of the text input field, indicating the handle for resizing.

Les champs de saisie

- ❑ Le type **date** permet à l'utilisateur de **sélectionner** une date dans un **calendrier** affiché par le navigateur. Vous pouvez **restreindre** l'intervalle proposé au visiteur. Le **rendu** visuel peut légèrement **différer** selon le **navigateur**



- Navigateur Chrome -



- Navigateur Firefox -

Les champs de saisie

- ❑ Le type **checkbox** affiche une **case** à **cocher**. Dans le cas où il y a **plusieurs** cases, l'**utilisateur** peut en cocher une ou **plusieurs** à la fois. L'attribut **checked** permet de cocher une case **par défaut**

```
<input type="checkbox" id="news" name="news" checked >  
<label for="news">Je souhaite recevoir votre newsletter </label>
```



☒ Je souhaite recevoir votre newsletter

- ❑ Le type **radio** affiche un **cercle** à **sélectionner**. Dans le cas où il y a **plusieurs** choix, l'**utilisateur** ne peut en choisir qu'**un seul** la fois. Les différents **cercles** doivent tous avoir l'attribut **name** avec la même **valeur**

```
<p>Sélectionnez votre niveau : </p>  
<input type="radio" name="level" id="debutant" checked >  
<label>Débutant </label>  
<input type="radio" name="level" id="expert" >  
<label>Expert </label>
```



Sélectionnez votre niveau :

- ☒ Débutant
☐ Expert

Les champs de saisie

- ❑ Comme le type **submit** affiche un **bouton** pour **soumettre** les **données** d'un formulaire, le type **reset** propose un **bouton** pour les **réinitialiser**. L'attribut **value** permet de lui choisir un **texte**

```
<input type="reset" value="Effacer les données" />
```



Effacer les données

- ❑ De **nombreux** autres **types d'input** existent permettant de **joindre** un **fichier**, afficher un **horaire**, saisir une **url**, choisir une **couleur** et bien d'autres choses...
- ❑ https://www.w3schools.com/html/html_form_input_types.asp
- ❑ <https://developer.mozilla.org/fr/docs/Web/HTML/Element/Input>

Listes d'options

- ❑ La balise `<select>` permet d'afficher une **liste d'options déroulante**. Comme pour un `<input>`, cette balise est liée à une balise `<label>` à l'aide d'un couple d'attributs **for-id**
- ❑ L'attribut **label** est lui toujours présent pour **identifier** la **donnée** qui sera envoyée au **serveur**. L'attribut **value** permet de **préciser** le **choix** qui sera fait par l'**utilisateur**

```
<label for="countries" >Choose your destination : </label>
<select name="countries" id="countries" >
  <option value="Maldives" > Maldives </option>
  <option value="Turkey" > Turkey </option>
  <option value="Canada" > Canada </option>
  <option value="Zambia" > Zambia </option>
</select>
```



Choose your destination :

Choose your destination :

Maldives ▼

Maldives
Turkey
Canada
Zambia

Listes d'options

- ❑ La balise `<optgroup>` permet de classer la liste d'options à l'aide de sous-catégories dont les titres seront renseignés par l'attribut `label`

```
<select name="countries" id="countries">
  <optgroup label="Asia">
    <option value="Malaysia"> Malaysia </option>
    <option value="Philippines"> Philippines </option>
  </optgroup>
  <optgroup label="Europe">
    <option value="Netherlands"> Netherlands </option>
    <option value="Spain"> Spain </option>
  </optgroup>
</select>
```



Choose your destination :

- Asia**
 - Malaysia
 - Philippines
 - Japan
- Europe**
 - Netherlands
 - Spain
 - Ireland
- Africa**
 - Egypt**
 - Cameroon
 - Mauritius

Les champs de saisie

- ❑ La balise `<datalist>` propose une **alternative** à la balise `<select>` et fonctionne avec les balises `<input>` et `<option>`
- ❑ Cette méthode permet de ne pas **restreindre** l'utilisateur dans son **choix** : il pourra alors saisir une **valeur** qui ne figure pas dans la liste proposée
- ❑ Le **champ de texte** offre également une **saisie semi-automatique** en fonction des lettres saisies
- ❑ Cependant, la création de **sous-catégories** grâce à la balise `<optgroup>` n'est pas disponible avec cette méthode
- ❑ <https://stackoverflow.com/questions/6865943/html-form-select-option-vs-datalist-option>

Listes d'options

- ❑ La balise `<input>` doit posséder l'attribut `list` dont la valeur est la même que l'`id` de la balise `<datalist>`
- ❑ Les balises `<option>` ne nécessitent pas de **balises fermantes**

```
<label for="country"> Choose your country : </label>
<input list="country-name" name="country" id="country">
<datalist id="country-name">
  <option value="UK">
  <option value="Germany">
  <option value="USA">
  <option value="Japan">
  <option value="India">
</datalist>
```


Listes d'options

- ❑ La balise `<fieldset>` est une **balise-container** permettant d'**organiser** les différents **éléments** d'un **formulaire** en les répartissant dans des **groupes** auxquels la balise `<legend>` donnera une **étiquette**

```
<fieldset>
  <legend> Account Informations </legend>
  <label for="nom" > Nom </label>
  <input type="text" id="nom" name="nom" >
</fieldset>
<fieldset>
  <legend> Shipping infos </legend>
  <label for="ship-nom" > Nom </label>
  <input type="text" id="ship-nom" name="ship-nom" >
</fieldset>
```



Account Informations

Nom

Prénom

email

Shipping infos

Nom

Adresse

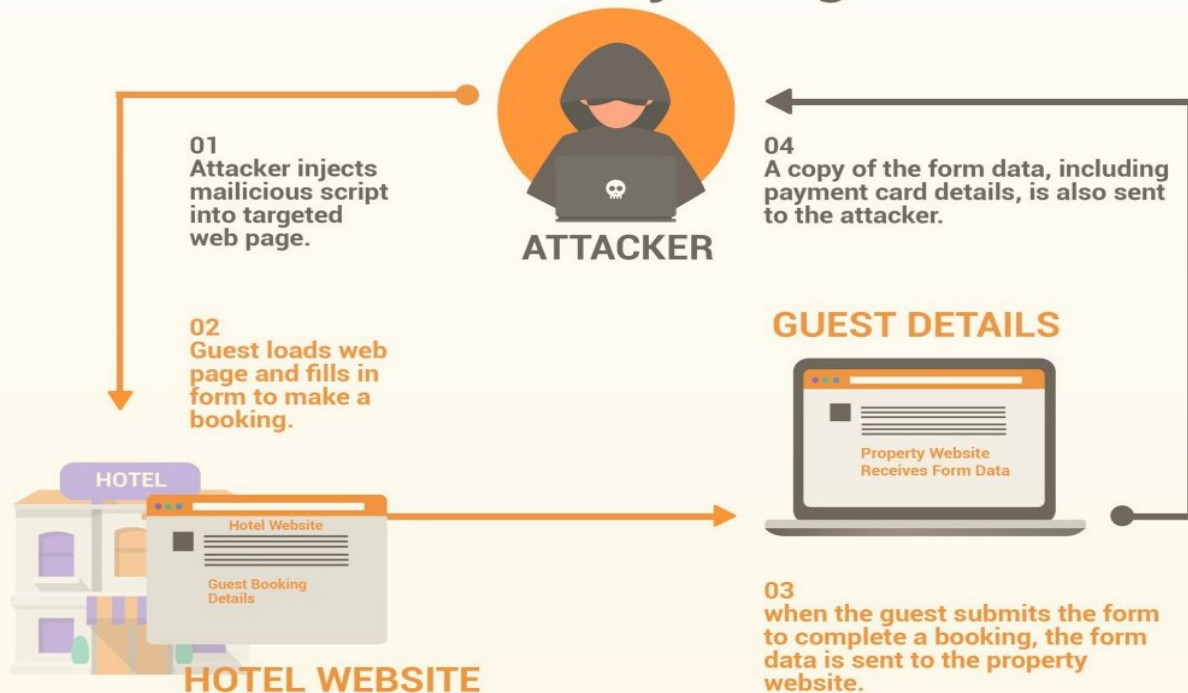
Zip Code

Vérification des données

- ❑ Les **formulaires** peuvent facilement être la **cible d'attaques**. Une des techniques utilisées est d'**injecter** du **code malveillant** dans les **champs de saisie**
- ❑ La plupart des **navigateurs** offrent aujourd'hui des systèmes de **contrôle** : ils peuvent **bloquer** l'envoi de **code JavaScript** ou du **formulaire** si certaines **données** n'ont pas la forme attendue
- ❑ Les **données** d'un formulaire peuvent aussi tout simplement contenir des **erreurs de saisie**. Pour ces raisons, il est important de pouvoir **vérifier** les **données** saisies avant de les **envoyer** sur un **serveur**
- ❑ De nombreux **attributs HTML** sont disponibles permettant de **limiter** au maximum ces **erreurs de saisie** en précisant certaines **conditions** requises

Vérification des données

How Does Formjacking Work?



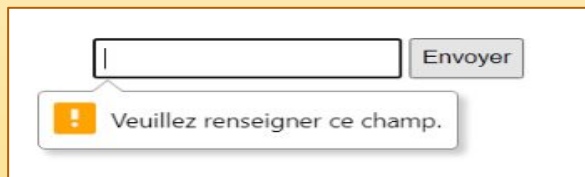
Vérification des données

- ❑ Les attributs **minlength** et **maxlength** permettent de préciser le **nombre** minimum ou maximum de **caractères** attendu dans un **champ de saisie**
- ❑ Dans les **champs** de saisie **numérique** comme **number** ou **date**, les attributs **min** et **max** permettent de préciser les valeurs **minimales** ou **maximales** attendues
- ❑ Pour ces mêmes champs **numériques**, l'attribut **step** permet de spécifier les **intervalles** lors de l'utilisation des **commandes** fléchées sur le côté



Vérification des données

- ❑ L'attribut **required** permet de **forcer** l'utilisateur à **remplir** un champ en **bloquant** l'**envoi** du formulaire si le champ de saisie est **vide**. Un **message** s'affiche alors pour **avertir** l'utilisateur



- ❑ L'attribut **autocomplete** propose à l'utilisateur une **auto-complétion** basée sur un choix de valeurs
- ❑ L'attribut **pattern** permet de préciser une succession de **caractères**, appelée **expression régulière** et la **valeur** saisie devra alors **respecter** le modèle de cette **expression** pour être **valide**
- ❑ <https://developer.mozilla.org/en-US/docs/Web/HTML/Attributes/autocomplete>
- ❑ <https://dridk.me/expression-reguliere.html>

Les tableaux

- ❑ Le moment est venu de tester votre capacité à **chercher** par vous-même et bien sûr à **trouver l'information** dont vous avez besoin
- ❑ Vous êtes libres du **choix** des **sources** où chercher les **informations** mais prenez le temps de les **recouper** pour confirmer leur **véracité**
- ❑ Le fait de **multiplier** les **sources** vous permettra également de **tester** votre **compréhension** du sujet abordé
- ❑ La meilleure façon d'**acquérir** correctement de nouvelles **données** est de les **tester** tout de suite dans votre **éditeur de texte** ou bien même **en ligne**

Les tableaux

- ❑ Des **tableaux** peuvent être réalisés en **HTML**, ils nécessitent l'utilisation de plusieurs **balises** différentes
- ❑ Un minimum de **CSS** sera également nécessaire pour que le **rendu visuel** soit **exploitable**

Commande de fournitures :

Article	Quantité	Prix	Total
Cahier	10	1.50	15€
Crayon	15	0.80	12€
Gomme	7	0.30	2.10€
Total général			29.10€



Commande de fournitures :

Article	Quantité	Prix	Total
Cahier	10	1.50	15€
Crayon	15	0.80	12€
Gomme	7	0.30	2.10€
Total général			29.10€