

Approfondissement en HTML

Développement Web - HTML / CSS



Sommaire

1. Navigation.
2. Formulaires.
3. Médias avancés.
4. Caractères spéciaux

1. Navigation.

1. Navigation.

Rappel :

- Balise `<nav>` :
 - Section contenant des liens de navigation (Menu horizontal ou vertical)
 - Peut être à plusieurs endroits dans une page (Header, aside, footer, etc.)
- Balises `<a>` :
 - Pour « ancre »
 - Lien hypertexte permettant la navigation interne au site (menu) ou externe
 - Contient du texte, URL spécifié par l'attribut « href »

1. Navigation.

Navigation dans une page :

- Il est possible avec une balise `<a>` de naviguer au sein d'une même page.
- Deux prérequis sont nécessaires :
 - Un conteneur (balise de section ou balise de structure) avec un attribut « `id` »
 - Utiliser l'attribut « `href` » dans la balise `<a>` avec un fragment suivi de l'identifiant du conteneur

1. Navigation.

Navigation dans une page — Exemple :

```
<body>
  <header>
    <nav>
      <a href="#about">A propos</a>
      <a href="#contact">Me contacter</a>
    </nav>
  </header>
  <section id="about" style="background-color:lightcoral; height:600px;">
    <p>about</p>
  </section>
  <section id="contact" style="background-color:lightblue;height:600px;">
    <p>contact</p>
  </section>
  <footer>
    <a href="#">Retour en haut</a>
  </footer>
</body>
```

[A propos](#) [Me contacter](#)

about

contact

[Retour en haut](#)

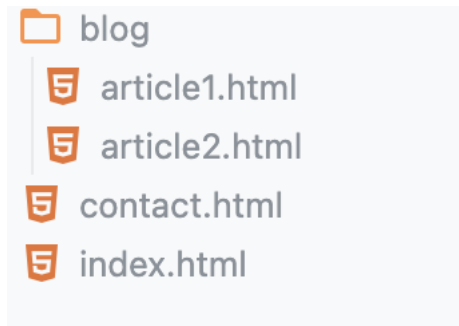
1. Navigation.

Navigation multi-pages :

- URL **absolues** et **relatives**
- Lorsqu'une balise `<a>` avec un attribut « href » est utilisée avec une URL relative, l'utilisateur du site est déplacé vers la page locale spécifiée
- Lorsqu'un permet de se déplacer dans un site, il n'est pas souhaité d'ouvrir un nouvel onglet.

1. Navigation.

Navigation multi-pages — Exemple :



Arborescence des fichiers

```
<body>
  <header>
    <nav>
      <a href="#about">A propos</a>
      <a href="blog/article1.html">article 1</a>
      <a href="blog/article2.html">article 2</a>
      <a href="contact.html">Me contacter</a>
    </nav>
  </header>
  <section id="about" style="background-color:lightcoral; height:600px;">
    <p>about</p>
  </section>
  <section id="contact" style="background-color:lightblue;height:600px;">
    <p>contact</p>
  </section>
  <footer>
    <a href="#">Retour en haut</a>
  </footer>
</body>
```

| URL relatives

1. Navigation.

Navigation externe :

- Lorsqu'une balise `<a>` avec un attribut « href » est utilisée avec une URL absolue, l'utilisateur du site est déplacé vers la page spécifiée
- Il est possible d'ouvrir la page dans un nouvel onglet en ajoutant l'attribut `target=« _blank »`.
- Lorsqu'un lien fait sortir d'un site, il est souhaité d'ouvrir un nouvel onglet.

1. Navigation.

Navigation externe — Exemple :

```
<body>
  <footer>
    <div id="socials">
      <!-- nouvel onglet -->
      <a href="https://facebook.com" target="_blank">Facebook</a>
      <a href="https://twitter.com" target="_blank">Twitter</a>
      <!-- redirection -->
      <a href="https://instagram.com">Instagram</a>
    </div>
    <a href="#">Retour en haut</a>
  </footer>
</body>
```

- URL absolues
- S'ouvriront dans un nouvel onglet
- Redirection : la page actuelle sera quittée

1. Navigation.



2. Formulaires

2. Formulaires.

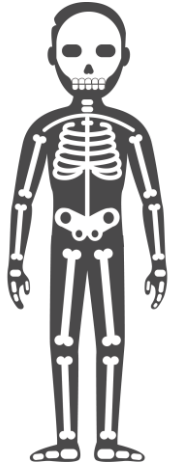
Rappel :

- Balise `<form>` :
 - Attribut « `method` » qui définit la méthode HTTP utilisée pour envoyer les données au serveur (GET ou POST)
- Balise `<label>` :
 - Nom du champ associé
 - Lié au champ par l'attribut « `for` »
- Balise `<input>` :
 - Attribut « `type` » qui définit le type d'entrée du champ
 - Lié au label par l'attribut « `name` »

2. Formulaires.

Fonctionnement d'un formulaire :

- Possède des champs pouvant être différents types
- Le formulaire HTML ne permet que de récolter les informations et ne les exploite pas. Une autre partie d'en s'en occuper à sa place.
Cela peut être :
 - Un langage de script (Javascript par exemple) relié au formulaire
 - Une API



HTML



Javascript

2. Formulaire.

Une API, qu'est-ce que c'est ?

- Interface de Programmation Applicative (ou « Application Programming Interface »)
- Permet de faire le lien entre deux applications distinctes
 - Peut recevoir et/ou envoyer des données
- Sert de porte d'entrée à des applications privées
- Souvent normalisée par le style d'architecture « REST »
- Sécurisée par une clef d'API, permettant au serveur de vous identifier

2. Formulaire.

Protocole HTTP :

- Qu'est-ce que le protocole HTTP ?
 - Méthode de communication client-serveur créée par Tim Berners-Lee (encore) pour le World Wide Web
 - Fonctionne par le biais de requêtes :
 - GET : Le navigateur (client) demande au serveur des informations (Par exemple une page HTML)
 - POST : Le navigateur envoie des informations au serveur pour qu'il puisse les traiter
 - HEAD : Permet au navigateur de récupérer les données d'en-tête uniquement d'un fichier HTML auprès du serveur

2. Formulaires.

OK mais le formulaire ?

```
<form action="https://api.url.com" method="POST">

  <input type="hidden" name="access_key" value="YOUR_ACCESS_KEY_HERE" />

  <input type="text" name="name" required/>
  <input type="email" name="email" required/>
  <textarea name="message" required></textarea>

  <button type="submit">Submit Form</button>
</form>
```

Requête HTTP vers une API

Clef d'authentification auprès de l'API

2. Formulaires.

Attributs supplémentaires :

- « **action** » : URL de l'API qui va recevoir les données du formulaire lors de sa soumission
- « **target** » : Comme pour la balise `<a>` certains formulaires peuvent faire des redirections
- « **type** » : Déjà évoqué précédemment, l'attribut type permet à un champ de prendre de nombreuses formes

2. Formulaires.

Attribut de champ « type » :

- date : Permet de sélectionner une date
- email : Champ n'acceptant qu'une adresse mail syntaxiquement valide
- number : Champ n'acceptant que des nombres
- password : Masque automatiquement le texte écrit dans le champ
- tel : Champ dédié aux numéros de téléphones et qui affiche le clavier téléphonique sur mobile.

2. Formulaires.

Attribut de champ « type » :

- **checkbox** : Permet de faire une case à cocher dans le formulaire (Par exemple : « Être recontacté »)
- **radio** : Permet de sélectionner une seule valeur parmi un groupe commun (portant le même attribut « name ») de boutons radio
- **file** : Permet de sélectionner un fichier (Les types de fichiers acceptés sont définis par l'attribut « accept »)
- **hidden** : Masque le champ (pour passer des données en supplément de celles données par l'utilisateur, une clef d'API par exemple)

2. Formulaires.



3. Médias avancés.

3. Médias avancés.

Rappel :

- Balises `` :
 - Pour « image »
 - Permet d'afficher une image, URL spécifié par attribut « `src` »
- Balise `<figcaption>` :
 - Ajoute une légende à la suite d'une image

```
  
<figcaption style="margin-left:10px;">Logo de Supinfo</figcaption>
```



Logo de Supinfo

3. Médias avancés.

Rappel :

- Balises `<video>` & `<audio>` :
 - Contenu vidéo ou audio
 - Lecteur affiché par l'attribut « `controls` »
- Balise `<source/>` :
 - Encapsulé dans la balise video ou audio
 - Permet d'ajouter une ou plusieurs sources

```
<video controls>
  <source src="video.mp4" type="video/mp4" />
  <source src="video.webm" type="video/webm" />
  <p>
    Votre navigateur ne prend pas en charge les vidéos HTML5. Voici
    <a href="video.mp4">un lien pour télécharger la vidéo</a>.
  </p>
</video>
```

```
<audio controls>
  <source src="video.mp4" type="video/mp4" />
  <source src="video.webm" type="video/webm" />
  <p>
    Votre navigateur ne prend pas en charge les vidéos HTML5. Voici
    <a href="video.mp4">un lien pour télécharger la vidéo</a>.
  </p>
</audio>
```


3. Médias avancés.

Objets intégrés — iframe :

- « Inline frame »
- L'élément `<iframe>` permet d'imbriquer une page HTML au sein de la page actuelle
- Chaque **contexte de navigation** (« onglet » imbriqué dans une page) possède son propre historique, comme un onglet lambda d'un navigateur.
- Charger un `<iframe>` dans une page augmente considérablement la consommation de ressource de votre page. Il ne faut pas en abuser.

3. Médias avancés.

Attributs d'une balise `<iframe>` :

- « `src` » : URL de la source à intégrer
- « `title` » : Titre et description de la page encapsulée
- « `width` » et « `height` » : Largeur et hauteur (en pixel) de l'élément
- « `allowfullscreen` » : Permet à l'iframe de pouvoir être mis en plein écran

3. Médias avancés.

Utilisation courante :

- De nombreux sites utilisent des iframes pour intégrer des applications externes :
 - Intégration de carte (Google Maps, OpenStreetMap)
 - Lecteurs vidéo (Youtube, Dailymotion, etc.)
 - Processus de paiement des banques (Confirmation mobile « 3D Secure »)
 - Feed de réseaux sociaux (Timeline twitter, etc.)
 - Contenus audio (Radios, playlists Spotify, etc.)
 - ...

3. Médias avancés.

Mauvais usages :

- Les iframes sont pratiques mais ont des répercussions sur un site (Possible baisse de référencement naturel, alourdissement de la page)
- Il ne faut donc pas substituer les autres balises médias par des iframes si ce n'est pas nécessaire
- Utiliser un iframe sur un site revient à faire confiance à la source affichée :
 - Il y existe des risques, une source non-fiable pourra injecter du code aux utilisateurs de votre site

3. Médias avancés.

Iframe — Exemple :

```
<iframe
  src="https://www.google.com/maps/embed?pb=supinfo"
  width="600"
  height="450"
  style="border:0;"
  allowfullscreen=""
  loading="lazy"
  referrerpolicy="no-referrer-when-downgrade">
</iframe>
```



3. Médias avancés.

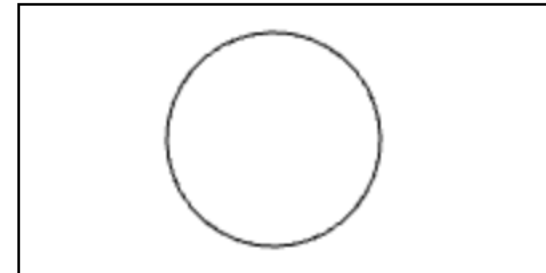
Objets intégrés — canvas :

- L'élément `<canvas>` est un conteneur utilisé pour dessiner facilement des graphiques en Javascript
- Comme pour la balise `<iframe>`, `<canvas>` utilise les attributs « `width` » et « `height` » pour définir la taille du conteneur
- Usage situationnel mais bon à connaître

3. Médias avancés.

Canvas — Exemple :

```
<canvas id="myCanvas"
  width="200"
  height="100"
  style="border:1px solid #000000;">
  <script>
    <!-- Dessine un cercle -->
    var c = document.getElementById("myCanvas");
    var ctx = c.getContext("2d");
    ctx.beginPath();
    ctx.arc(95, 50, 40, 0, 2 * Math.PI);
    ctx.stroke();
  </script>
</canvas>
```



3. Médias avancés.

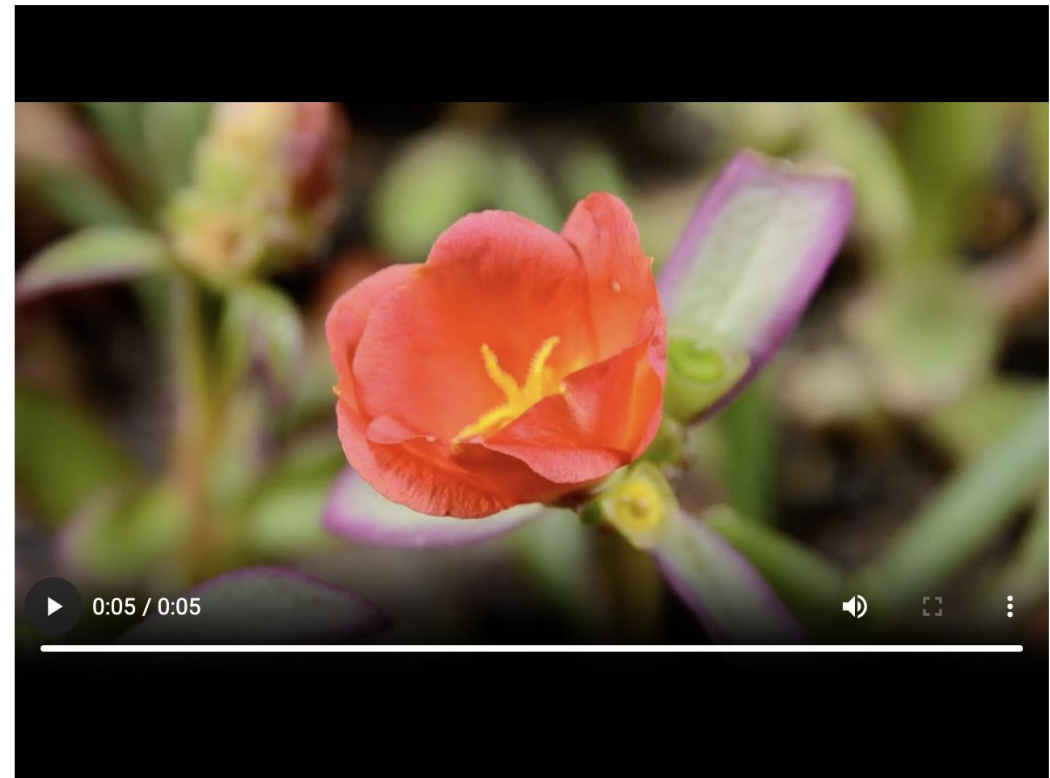
Objets intégrés — embed :

- L'élément `<embed>` relativement similaire à l'élément `iframe` mais plus permissif. Il permet d'intégrer n'importe quel média et non juste de l'HTML
- Comme pour la balise `<iframe>`, `<embed>` utilise les attributs « `width` » et « `height` » pour définir la taille du conteneur
- L'élément est en voie de dépréciation mais reste utilisé sur de nombreux sites

3. Médias avancés.

Embed — Exemple :

```
<embed  
  type="video/mp4"  
  src="flower.mp4"  
  width="640"  
  height="480"  
  title="Fleur qui s'ouvre"  
>
```



3. Médias avancés.



4. Caractères spéciaux.

4. Caractères spéciaux.

Encodage des caractères spéciaux :

- Dans certains cas ou avec certains (vieux) navigateurs, il est parfois nécessaire d'encoder certains caractères spéciaux pour s'assurer de leur bon rendu sur une page HTML
- Un caractère spécial est n'importe quel caractère autre qu'une lettre (sans accent) ou un chiffre

4. Caractères spéciaux.

Encodage des caractères spéciaux :

- Deux façons d'encoder un caractère spécial :
 - **Entité HTML** : Commence toujours par le caractère « & » et finit toujours par le caractère « ; »
 - **Caractère Ascii** : Commence toujours par les caractères « &# » et finit toujours par le caractère « ; »

4. Caractères spéciaux.

Espaces en HTML :

- En HTML, les espaces multiples sont automatiquement rognés



Petit exemple

- L'encodage du caractère espace est « » (pour **N**on **B**reakable **S**Pace)



Petit exemple

4. Caractères spéciaux.

Quelques caractères spéciaux communs :

Caractère spécial	Entité HTML	Caractère ASCII	Description
©	©	©	Copyright
®	®	®	Marque déposée
™	™	™	Marque commerciale
&	&	&	Et commercial
@	@	@	Arobase
«	«	«	Guillemet français ouvrant
»	»	»	Guillemet français fermant
“	“	“	Guillemet double ouvrant (anglais)
”	”	”	Guillemet double fermant (anglais)
‘	‘	‘	Guillemet simple ouvrant (anglais)
’	’	’	Guillemet simple fermant (anglais)

4. Caractères spéciaux.

Accents :

- Vous avez déjà sûrement vu des pages ou mails mal encodées avec des accents remplacés par ce genre de caractère :

Caractère spécial	Entité HTML	Caractère ASCII	Description
à	à	à	a accent grave
é	é	é	e accent grave
ê	ê	ê	e accent circonflexe
œ	œ	œ	oe ligaturés
û	û	û	u accent circonflexe
ù	ù	ù	u accent grave
€	€	€	Euro
\$	\$	$	Dollar
£	£	£	Pound, Livre sterling
¥	¥	¥	Yen

4. Caractères spéciaux.

Comment éviter les soucis d'encodage :

- La balise `<meta />` est là pour ça :
 - En plus de permettre de spécifier les métadonnées d'une page (description, auteur, taille de la fenêtre, etc.), il est possible de spécifier l'encodage souhaité de la page avec l'attribut « `charset` »
- Différentes normes d'encodages existantes, mais la plus commune et celle couvrant le plus de caractères est la norme « UTF-8 »

4. Caractères spéciaux.

Comment éviter les soucis d'encodage — Exemple :



```
<!DOCTYPE html>
<html>
  <head>
    <title>titre de la page</title>
    <meta charset="UTF-8"/>
    <meta name="description" content="Description de la page"/>
  </head>
  <body>
  </body>
</html>
```

4.Caractères spéciaux.



