



1

Chapitre 2: HTML



Dorra DHAOU

Maître-assistant à l'ISIMM

Docteur et Ingénieur en Informatique

dorra.dhaou@fst.utm.tn / dorradhaou@gmail.com

AU. 2025 - 2026

1

Plan

2

- HTML
- Structure d'un document HTML
- Syntaxe
 - Texte
 - Listes
 - Tableaux
 - Formulaires
 - etc.

2

HTML

3

□ HTML (HyperText Markup Language)

- Hypertexte: texte + liens vers d'autres pages ou objets
- C'est un langage de balisage (<> </>) simple à utiliser
- Un document HTML définit:
 - La structure du texte (titres, paragraphes, sous titres, etc.)
 - Le contenu (images, vidéos, liens, etc.)
 - La mise en forme du texte (couleur, taille, alignement, etc.)

3

HTML

4

□ HTML (HyperText Markup Language)

- Le navigateur Web est chargé d'interpréter le code HTML et le présenter aux visiteurs du site
 - La présentation d'un document HTML diffère d'un navigateur à un autre
 - Elle diffère aussi d'une version à une autre du même navigateur
 - Il faut tester les pages Web sur plusieurs navigateurs

4

HTML

5

□ HTML (HyperText Markup Language)

- ▣ Ce n'est pas un langage proprement dit car on n'a pas de programmes créés
- ▣ C'est un langage multi-systèmes (Windows, Linux, Mac, etc.) et multi-plateformes (Explorer, Chrome, Firefox, etc.)
- ▣ Il permet de créer une page Web et assurer sa mise en forme
- ▣ À partir de sa version 4 (1997), la mise en forme est gérée par des feuilles de styles (fichiers CSS)

5

HTML

6

□ HTML (HyperText Markup Language)

- ▣ La version 5 (2014) a conservé des balises de la version 4 et apporté des nouvelles balises sémantiques:
 - Syntaxe: <DOCTYPE>, aucune différence entre minuscule et majuscule, etc.
 - Structuration: <section>, <header>, <aside>, etc.
 - Multimédia: <audio> et <video>
 - Types d'input pour les formulaires: datetime, date, etc.
- ▣ Suppression de balises: , <basefont>, <big>, etc.
 - Les feuilles de style (CSS) font l'affaire

6

7

Structure d'un document HTML

7

Structure d'un document HTML

8

```
<!DOCTYPE HTML>
<HTML>
  <HEAD>
    <meta charset="ISO-8859-1"/>
    <meta name="author" content="auteur"/>
    <TITLE>Hello world!!</TITLE>
  </HEAD>
  <BODY>
    Un premier exemple tout simple!!
  </BODY>
</HTML>
```

8

Structure d'un document HTML

9

- Le code HTML est écrit sous forme de balises (tags).
- Une balise doit être ouverte et fermée.
- La fermeture d'une balise est identique à son ouverture sauf qu'elle comprend un slash « / » avant le nom de la balise.
 - ▣ **<balise> texte affecté par la balise </balise>**
 - ▣ Exemple:
 - **<title>Cours Programmation WEB</title>**

9

Structure d'un document HTML

10

- Une balise peut être écrite en minuscule ou majuscule.
 - ▣ **<HTML>, <html> et <HtmL> sont équivalentes**
- Une balise peut comporter des attributs mis entre guillemets.
 - ▣ **<tag attribut1="arg1" attribut2="arg2"> texte </tag>**
 - ▣ Exemple: **<meta name="author" content="auteur"/>**
- Les sauts de lignes et espaces ne sont pas interprétés.

10

Structure d'un document HTML

11

Commentaires

- Pour écrire un commentaire dans le code source, il faut le placer entre « <!-- » et « --> ».
- Le commentaire n'est visible que dans le code source (invisible dans le navigateur).
 - ▣ Exemple1: <!-- ceci est un commentaire -->
 - ▣ Exemple2: <!--

```

      ceci est
      un commentaire
      sur plusieurs lignes
    -->

```

11

Structure d'un document HTML

12

Balise DOCTYPE

- C'est une balise exceptionnelle et indispensable écrite dans la première ligne du document.
 - ▣ Elle commence par un point d'exclamation
 - ▣ Elle informe sur le type de document (doctype) traité par le navigateur Web
- La version 5 du HTML a simplifié l'écriture de cette balise.
 - ▣ Exemple: <!DOCTYPE HTML>

12

Structure d'un document HTML

13

Balise HTML

- C'est l'élément racine du code d'une page Web.
- C'est la balise principale regroupant tout le code écrit.
 - ▣ Elle doit être fermée à la fin du code
 - ▣ Exemple:

```
<HTML>
  il faut mettre dedans tout le code
  ...
</HTML>
```

13

Structure d'un document HTML

14

Balise HEAD

- Il s'agit de l'entête du document.
- La balise comprend des informations générales du document (titre, auteur, encodage, etc.).
 - ▣ Ces informations ne seront pas affichées dans la page Web
 - ▣ Exemple:

```
<HEAD>
  définir le titre, auteur, etc.
</HEAD>
```

14

Structure d'un document HTML

15

Balise HEAD

- Elle fait appel aux scripts et feuilles de style.
 - ▣ Exemples: `<script>`, `<style>`, `<link>`
- Elle comprend aussi les métadonnées du document (informations descriptives sur l'auteur, l'encodage, etc.)
 - ▣ Exemple: `<meta>`

15

Structure d'un document HTML

16

Balise TITLE

- C'est l'élément le plus important de l'entête du document.
- Il définit le titre de la fenêtre ou de l'onglet dans le navigateur Web.
- Il n'apparaît pas dans le contenu de la page Web.
- Il apparaît dans les résultats de recherche.
 - ▣ Exemple:
`<TITLE> Programmation Web </TITLE>`

16

Structure d'un document HTML

17

Balise META

- Elle déclare les métadonnées du document.
- Ces informations ne seront pas affichées dans la page Web.
- Elles sont utiles pour:
 - ▣ Les navigateurs (encodage des caractères)
 - ▣ Les moteurs de recherche (mots clés)
- C'est une balise avec attributs mais sans corps.
 - ▣ Balise orpheline (<meta/>)

17

Structure d'un document HTML

18

Balise META

- Les attributs utilisés sont basés sur le principe de couple (*mot clé, valeur*).
 - ▣ Exemples:
 - <meta charset="utf-8"/>**: définir le type d'encodage
 - <meta name="description" content="la description de la page html"/>**: définir une description du document en utilisant des mots clés, utile pour les moteurs de recherche
 - <meta name="keywords" content="des mots clés..."/>**: spécifier des mots clés pour le document

18

Structure d'un document HTML

19

Balise META

`<meta name="author" content="auteur de la page" />`:
spécifier le nom de l'auteur du document

`<meta http-equiv="Content-Language" content="fr" />`:
spécifier la langue du site

- Bien évidemment, il y en a d'autres exemples de la balise `<meta>`.

19

Structure d'un document HTML

20

Balise BODY

- C'est le corps du document.
- Il est placé après l'entête.
- Il regroupe tout le code du contenu qui sera affiché dans la page.
- La balise est unique.

■ Exemple:

`<BODY>` tout le contenu à afficher `</BODY>`

20

21

Syntaxe - Texte

21

Texte

22

Texte préformaté

- La balise **<PRE>** est utilisée pour afficher un texte préformaté c'est-à-dire afficher un texte comme c'est écrit dans le document HTML.
 - ▣ Il doit respecter les espaces, tabulations, etc.
 - ▣ La balise n'est pas appliquée à des titres, paragraphes, etc.
 - ▣ Exemple:

`<PRE>` je voudrais bien que ce texte soit séparé
des espaces et affiché dans la page `</PRE>`

je voudrais bien que ce texte soit séparé des espaces et affiché dans la page

22

Texte

23

Texte préformaté

□ Avec/sans **<PRE>**

```

<body>
  <p>
    Les inscrits sont:
    Nom      Prénom   Groupe
    -----
    Youssef   Ahmed    1
    Tounsi    Ines     2
  </p>
  <pre>
    Les inscrits sont:
    Nom      Prénom   Groupe
    -----
    Youssef   Ahmed    1
    Tounsi    Ines     2
  </pre>
</body>

```

23

Texte

24

Mise en forme

```

<p>
  <b> b: affiche le texte en gras </b> <br>
  <i> i: affiche le texte en italique </i>
</p>
<p>
  <u> u: souligne le texte </u>
</p>
<p>
  <s> s: barre le texte </s>
</p>
<p>
  <small> small: police plus petite </small>
</p>
<p>
  <em> em: indique une mise en exergue (italique) </em> <br>
  <strong> strong: faire ressortir du texte (gras) </strong> <br>
  <var> var: variable </var> <br>
  <dfn> dfn: définition </dfn>
</p>

```

b: affiche le texte en gras

i: affiche le texte en italique

u: souligne le texte

~~s:~~ barre le texte

small: police plus petite

em: indique une mise en exergue (italique)

strong: faire ressortir du texte (gras)

var: variable

dfn: définition

24

Texte

25

Paragraphe

- Le contenu d'un paragraphe est délimité par les deux balises `<p>` et `</p>`.
- On ne peut pas imbriquer les balises `<p>`.
- Une ligne blanche est ajoutée par le navigateur avant et après le paragraphe.

▣ Exemple:

`<p>` contenu du paragraphe `</p>`

25

Texte

26

Paragraphe

- L'affichage du contenu d'un paragraphe dépend de la taille de fenêtre du navigateur.
- ▣ Le retour à la ligne est géré automatiquement

```
<p>
  La programmation web est la programmation
  informatique qui permet d'éditer des sites web.
  Elle permet la création d'applications destinées
  à être déployées sur Internet ou Intranet.
</p>
```

La programmation web est la programmation informatique qui permet d'éditer des sites web. Elle permet la création d'applications destinées à être déployées sur Internet ou Intranet.

26

Texte

27

Retour à la ligne

- La balise `
` (break) permet d'effectuer un retour à la ligne dans un paragraphe.
- C'est une balise ouvrante seulement

```
<p>
  La programmation web est la programmation <br>
  informatique qui permet d'éditer des sites web. <br>
  Elle permet la création d'applications destinées <br>
  à être déployées sur Internet ou Intranet.
</p>
```

La programmation web est la programmation
informatique qui permet d'éditer des sites web.
Elle permet la création d'applications destinées
à être déployées sur Internet ou Intranet.

27

Texte

28

Titre

- Il existe 6 niveaux de titres (écriture en gras).
- `<H1>`: le premier niveau ayant la plus grande taille
- ...
- `<H6>`: le dernier niveau ayant la plus petite taille

```
<h1> Titre de niveau 1 </h1>
<h2> Titre de niveau 2 </h2>
<h3> Titre de niveau 3 </h3>
<h4> Titre de niveau 4 </h4>
<h5> Titre de niveau 5 </h5>
<h6> Titre de niveau 6 </h6>
```

Titre de niveau 1

Titre de niveau 2

Titre de niveau 3

Titre de niveau 4

Titre de niveau 5

Titre de niveau 6

28

Texte

29

Citation

- Il existe trois balises dédiées aux citations:

- ▣ **<q>**: les citations courtes
- ▣ **<blockquote>**: les citations longues
- ▣ **<cite>**: source de citation

```
<p>
  Un exemple de citation: <br>
  <q> Ce qui se conçoit bien s'énonce clairement </q>
  <cite> Boileau </cite>
</p>
```

Un exemple de citation:
 “ Ce qui se conçoit bien s'énonce clairement ” *Boileau*

29

Texte

30

Citation

- La balise **<blockquote>** ajoute un retrait.

```
<p>
  Exemple sans blockquote: <br>
  La programmation web est la programmation <br>
  informatique qui permet d'éditer des sites web. <br>
  Elle permet la création d'applications destinées <br>
  à être déployées sur Internet ou Intranet. <br>
  Exemple avec blockquote: <br>
  <blockquote>
  La programmation web est la programmation <br>
  informatique qui permet d'éditer des sites web. <br>
  Elle permet la création d'applications destinées <br>
  à être déployées sur Internet ou Intranet.
  </blockquote>
</p>
```

30

Texte

31

Citation

- La balise **<blockquote>** ajoute un retrait.

Exemple sans blockquote:

La programmation web est la programmation informatique qui permet d'éditer des sites web. Elle permet la création d'applications destinées à être déployées sur Internet ou Intranet.

Exemple avec blockquote:

La programmation web est la programmation informatique qui permet d'éditer des sites web. Elle permet la création d'applications destinées à être déployées sur Internet ou Intranet.

31

Texte

32

Indices et exposants

- Les deux balises **<sub>** et **<sup>** permettent d'afficher un caractère respectivement comme indice et exposant.

```
<p>
Afficher a comme indice de X: X<sub>a</sub> <br>
Afficher b comme exposant de X: X<sup>b</sup> <br>
Afficher a comme indice et b comme exposant de X: X<sub>a</sub><sup>b</sup>
</p>
```

Afficher a comme indice de X: X_a

Afficher b comme exposant de X: X^b

Afficher a comme indice et b comme exposant de X: X_a^b

32

Texte

33

Caractères accentués

Caractère	Code HTML	Caractère	Code HTML
Á	Á	à	à
À	À	ä	ä
Ã	Ã	é	é
É	É	è	ê
È	È	ú	ú
Î	Î	ü	ü
Ö	Ö	ï	ï
Â	Â	ç	ç
Å	Å	â	â
Ä	Ä	æ	æ
Ê	Ê	è	è
Ë	Ë	ë	ë
Ô	Ô	û	û
Æ	Æ	ö	ö
Ç	Ç	î	î
		ù	ù

33

Texte

34

Caractères spéciaux

Caractère	Code HTML	Caractère	Code HTML
>	>	&	&
<	<	¶	¶
Espace insécable (non-breaking space)	 	œ	œ
®	®	"	"
™	™	©	©
»	»	±	±
«	«	±	¹
º	°	³	³
²	²	¼	¼
§	§	¾	¾
½	½	.	·
		‰	‰

34

35

Syntaxe - Listes

1. Introduction
2. Partie I
3. Partie II
4. Conclusion

- Introduction
- Partie I
- Partie II
- Conclusion

35

Listes

36

Liste ordonnée

- La balise **** (**ordered list**) permet de créer une liste ordonnée (numérotée).
 - ▣ La balise **** (**list item**) permet de créer un élément de la liste
 - ▣ Exemple:


```
<ol>
  <li> item 1 </li>
  <li> item 2 </li>
  ...
</ol>
```

36

Listes

37

Liste ordonnée

- L'attribut **type** permet de définir le type de numérotation utilisée (numérique ou alphabétique).
 - ▣ **type = "1"**: par défaut, utiliser des nombres (1, 2, ...)
 - ▣ **type = "A"**: utiliser des lettres en majuscule (A, B, C, ...)
 - ▣ **type = "a"**: utiliser des lettres en minuscule (a, b, c, ...)
 - ▣ **type = "I"**: utiliser des chiffres romains en majuscule (I, II, III, ...)
 - ▣ **type = "i"**: utiliser des chiffres romains en minuscule (i, ii, iii, ...)

37

Listes

38

Liste ordonnée

Exemples:

```
<ol>
  <li> Rouge </li>
  <li> Vert </li>
  <li> Bleu </li>
</ol>
```

1. Rouge
2. Vert
3. Bleu

```
<ol type="I">
  <li> Rouge </li>
  <li> Vert </li>
  <li> Bleu </li>
</ol>
```

I. Rouge
II. Vert
III. Bleu

```
<ol type="A">
  <li> Rouge </li>
  <li> Vert </li>
  <li> Bleu </li>
</ol>
```

A. Rouge
B. Vert
C. Bleu

```
<ol type="i">
  <li> Rouge </li>
  <li> Vert </li>
  <li> Bleu </li>
</ol>
```

i. Rouge
ii. Vert
iii. Bleu

```
<ol type="a">
  <li> Rouge </li>
  <li> Vert </li>
  <li> Bleu </li>
</ol>
```

a. Rouge
b. Vert
c. Bleu

38

Listes

39

Liste ordonnée

- L'attribut **start** permet de spécifier la valeur de départ d'une liste.

- Exemples:

```
<ol start="3">
  <li> Rouge </li>
  <li> Vert </li>
  <li> Bleu </li>
</ol>
```

```
3. Rouge
4. Vert
5. Bleu
```

```
<ol start="3" type="i">
  <li> Rouge </li>
  <li> Vert </li>
  <li> Bleu </li>
</ol>
```

```
iii. Rouge
iv. Vert
v. Bleu
```

39

Listes

40

Liste non ordonnée

- La balise **** (**unordered list**) permet de créer une liste non ordonnée (non numérotée utilisant des puces).
- L'attribut **type** permet de définir le type de puce utilisée.
 - ▣ **type = "disc"**: par défaut, un disque plein (rond noir)
 - ▣ **type = "circle"**: un cercle noir, intérieur transparent
 - ▣ **type = "square"**: un carré noir

40

Listes

41

Liste non ordonnée

□ Exemples:

```
<ul>
  <li> Rouge </li>
  <li> Vert </li>
  <li> Bleu </li>
</ul>
```

- Rouge
- Vert
- Bleu

```
<ul type="circle">
  <li> Rouge </li>
  <li> Vert </li>
  <li> Bleu </li>
</ul>
```

- Rouge
- Vert
- Bleu

```
<ul type="square">
  <li> Rouge </li>
  <li> Vert </li>
  <li> Bleu </li>
</ul>
```

- Rouge
- Vert
- Bleu

41

Listes

42

Liste de définitions

- La balise **<DL> (Description List)** permet de créer une liste de termes avec leurs définitions.

- **<DT> (Definition Term)**: l'élément à définir

- **<DD> (Definition Description)**: définition de l'élément

- Exemple:

```
<DL>
  <DT> Terme 1 </DT>
  <DD> Définition 1 </DD>
  <DD> Définition 2 </DD> ...
</DL>
```

42

Listes

43

Liste de définitions

□ Exemple:

```
<dl>
  <dt> Étoile </dt>
  <dd> Objet céleste étudié par les astrophysiciens.</dd>
  <dd> Danseur confirmé d'un ballet. </dd>
  <dt> Liste </dt>
  <dd> Ensemble d'éléments. </dd>
</dl>
```

Étoile	Objet céleste étudié par les astrophysiciens. Danseur confirmé d'un ballet.
Liste	Ensemble d'éléments.

43

Listes

44

Listes imbriquées

□ On peut imbriquer les listes ordonnées et non ordonnées.

□ Exemple:

```
<ol type="A">
  <li> Premier élément </li>
  <ul type="circle">
    <li> Début du premier élément </li>
    <li> Suite du premier élément </li>
  </ul>
  <li> Deuxième élément </li>
</ol>
```

A. Premier élément
◦ Début du premier élément
◦ Suite du premier élément
B. Deuxième élément

44

Exercice:

45

- Écrire le code HTML permettant d'obtenir le résultat ci-dessus.

Les développeurs Web utilisent ces langages lorsqu'ils créent un site:

HTML

HyperText Markup Language décrit la structure de la page et son contenu.

CSS

Cascading Style Sheets décrit à quoi ressemble un site et même dans une certaine mesure comment il répond à certains événements.

JavaScript

JavaScript est le langage de programmation utilisé pour définir la logique et la fonction d'un site au-delà de la simple apparence. Tout calcul ou "réflexion" se fait en JavaScript.

PHP

HyperText Preprocessor le meilleur langage de création de page dynamique

SQL

Structured Query Language le langage interrogation des bases de données

45

Correction

46

```
<p>Les développeurs Web utilisent ces langages lorsqu'ils créent un site:</p>
<dl>
  <dt>HTML</dt>
  <dd><em>HyperText Markup Language</em> décrit la structure de la page et
son contenu.</dd>
  <dt>CSS</dt>
  <dd><em>Cascading Style Sheets</em> décrit à quoi ressemble un site et
même dans une certaine mesure <br> comment il répond à certains
événements.</dd>
  <dt>JavaScript</dt>
  <dd>JavaScript est le langage de programmation utilisé pour définir la
logique et la fonction <br> d'un site au-delà de la simple apparence. Tout
calcul ou "réflexion" se fait en JavaScript.</dd>
  <dt>PHP</dt>
  <dd><em>HyperText Preprocessor</em> le meilleur langage de création de
page dynamique</dd>
  <dt>SQL</dt>
  <dd><em>Structured Query Language</em> le langage interrogation des bases
de données</dd>
</dl>
```

46

47

A SUIVRE...

TO BE CONTINUED...

47

48

Syntaxe - Liens

48

Liens

49

- La balise `<a>` permet de créer un lien hypertexte.
- Un lien est un outil de navigation entre pages ou objets situés dans le même dossier ou dans des dossiers différents.

▣ Exemple:

` Si on clique sur ce lien, on passera à la page pageCible `

Si on clique sur ce lien, on passera à la page pageCible

49

Liens

50

- Un lien non visité est souligné et coloré en bleu.
- Un lien visité est souligné et coloré en violet.
- Un lien actif est souligné et coloré en rouge.
- Les attributs de la balise `<a>` sont principalement:
 - **href**: spécifier la source du document cible
 - **target**: spécifier le type d'affichage du document
 - **id**: attribuer un identifiant unique à une ancre dans un document
 - **download**: télécharger un document
 - **title**: décrire un lien en affichant une infobulle

50

Liens

51

Lien externe

- Il permet d'accéder à un autre site Web.
- L'attribut **href** comprend l'adresse complète du site ou du fichier ainsi que le protocole utilisé.

▣ Exemples:

- ` Accéder à Google `
- ` Accéder au serveur ftp `

51

Liens

52

Lien interne au site

- Il pointe vers un autre document (page Web) situé dans le même répertoire ou dans un répertoire différent.

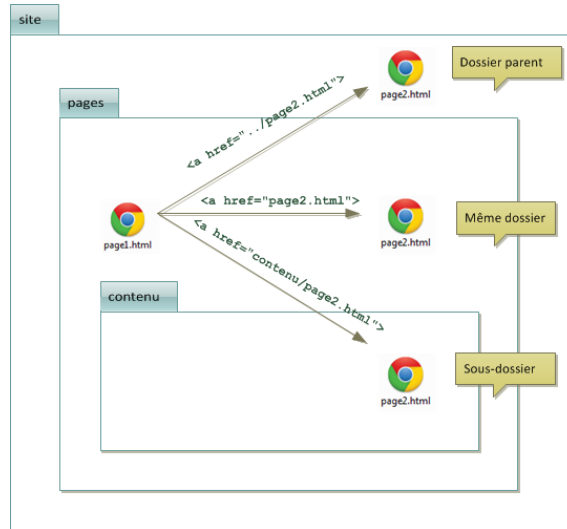
▣ Exemples:

- ` Accéder à la page (page2) du même répertoire courant `
- ` Accéder à la page (page2) du répertoire (contenu) `
- ` Accéder à la page (page2) du répertoire parent `

52

Liens

53



53

Liens

54

Lien interne à la page

- Il permet de pointer vers une partie d'un document (titre, section, définition, etc.).
 - ▣ Il s'agit d'un lien vers **une ancre**
 - ▣ C'est un texte ou image cliquables contenant un lien pointant vers un endroit précis de la même page ou d'une autre page d'un site Web.
 - ▣ C'est une sorte de point de repère que vous pouvez mettre dans vos pages HTML lorsqu'elles sont très longues.
- Deux étapes sont nécessaires:
 - ▣ **Attribuer un identifiant à une ancre avec l'attribut **id****
 - Exemple: ` texte ciblé `
 - ▣ **Créer un lien vers l'ancree avec l'attribut **href****
 - Exemple: ` texte du lien `

54

Liens

55

Lien interne à la page

```
<html>
  <body>
    <h1><a id="top">Titre de niveau 1</a></h1>
    <p>Paragraphe 1</p>
    <p>Paragraphe 2</p>
    <p>Paragraphe 3</p>
    ...
    <a href="#top">Aller vers le haut</a>
  </body>
</html>
```

55

Liens

56

Lien interne à la page

- Un lien peut aussi pointer vers une ancre dans une autre page du même répertoire ou d'un répertoire différent.
- L'idée, c'est de faire un lien qui ouvre une autre page ET qui amène directement à une ancre située plus bas sur cette page.
- En pratique, c'est assez simple à faire : il suffit de taper le nom de la page, suivi d'un dièse (#), suivi du nom de l'ancre.
 - ▣ Exemples:
 - Pointer vers une ancre dans la page (page2) du même répertoire
 - Pointer vers une ancre dans la page (page3) du répertoire dossier

56

Liens

57

Lien de téléchargement

- L'attribut **download** permet de télécharger un fichier (compressé, image, document pdf, vidéo, etc.)

□ Exemples:

- `` Cliquer pour télécharger le fichier compressé ``
- `` Cliquer pour télécharger le document Word ``
- `` Cliquer pour télécharger le fichier Excel ``

57

Liens

58

Lien de téléchargement

□ Exemples:

- `` Cliquer pour télécharger le fichier pdf ``
 - Le document sera affiché dans le navigateur et peut être ensuite téléchargé
- `` Cliquer pour télécharger le fichier mp3 ``
 - Le fichier sera joué dans le navigateur et peut être ensuite téléchargé
- `` Cliquer pour télécharger le fichier mp4 ``
 - Le fichier sera joué dans le navigateur et peut être ensuite téléchargé

58

Liens

59

Lien de messagerie

- Il est créé en utilisant une URI dont le schéma est **mailto** et l'information est l'adresse mail dans l'attribut **href** de la balise **<a>**.

■ Exemples:

- `` cliquer pour envoyer un e-mail ``: lancer une application de messagerie installée avec un nouveau message vide
- `` cliquer pour envoyer un e-mail ayant l'objet (message) ``

59

Liens

60

Lien de messagerie

- `` cliquer pour envoyer un e-mail ayant l'objet (message) et le corps (bonjour) ``
- `` cliquer pour envoyer un e-mail ayant l'objet (message) et le corps (bonjour) à deux destinataires ``

60

Liens

61

Lien ouvrant une nouvelle fenêtre

- L'attribut **target** permet d'ouvrir un lien dans une nouvelle fenêtre.

□ Exemples:

- ``
cliquer pour ouvrir le lien dans une autre fenêtre ``
- ``
cliquer pour ouvrir le lien dans la totalité de la fenêtre courante ``

61

Liens

62

Lien affichant une infobulle

- L'attribut **title** permettra d'afficher une bulle d'aide si on passe le curseur sur un lien.

□ Exemple:

- `` cliquer pour passer à Google ``

[Cliquer pour passer à Google](https://www.google.com)

Accéder à Google

62