

PROGRAMMATION WEB 1

Hatim KHARRAZ AROUSSI

2024-2025

Plan

- Chapitre 1 : Généralités : Internet et Web
- Chapitre 2 : Langage HTML
- Chapitre 3 : Langage CSS
- Chapitre 4 : Langage JQuery

CHAPITRE1 : Généralités Internet et Web

**Notions de base sur le réseau Internet et le
service du Web**

Généralités : Internet et Web

4

1. Notions de bases : Internet
2. Modèle TCP/IP
3. Adresse IP
4. DNS
5. Services Internet
6. Mécanisme Client / Serveur sur Internet
7. L'HyperText Transfert Protocol HTTP
8. Fonctionnement du Web
9. L'Uniform Resource Locator URL
10. Page Web
11. Site Web
12. Présentation sous le Web

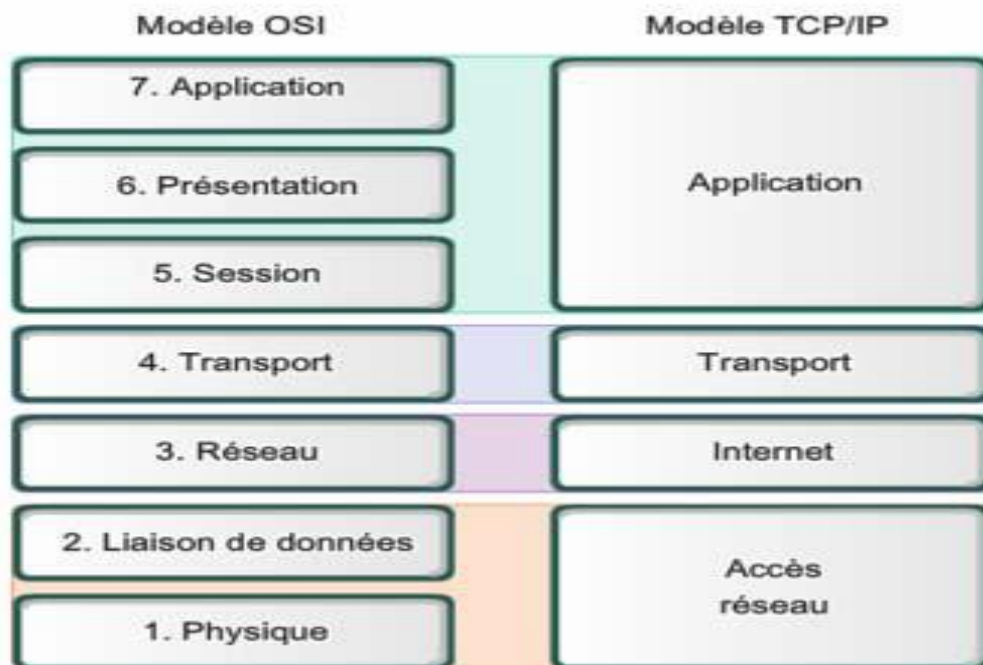
1. Notions de bases : Internet

- **Internet** = *Interconnected Networks*.
- Première utilisation remonte à **octobre 1972**.
- **Réseau informatique mondial** constitué d'un ensemble de réseaux nationaux, publiques et privés (**Réseau de réseaux**).
- Ces réseaux sont reliés et gérés par le **modèle de communication TCP/IP**.
- TCP/IP est un ensemble de protocoles qui coopèrent dans le but d'offrir une **interface unique à leurs utilisateurs (Services réseaux)**.

2. Modèle TCP/IP

6

- Modèle **TCP/IP** (**T**ransmission **C**ontrol **P**rotocol/**I**nternet **P**rotocol) = Pile de 4 Couches.
- Chaque couche est composée d'un ensemble de protocoles de
- Communication.
- La communication sur Internet entre machines se base sur la notion **d'adressage IP**.



3. Adresse IP

- Sur Internet, les ordinateurs communiquent entre eux grâce au **protocole IP (Internet Protocol)**
- Une adresse IP est notée @IPv4 est formée de **4 octets : 4 décimaux séparés par des points.**
- Un octet correspond à 8 bits de l'@ ce qui est fait **une adresse IP est constituée de 32 bits.**
- Exemple @IPv4 :
 - ▣ Format binaire : 10101100.00010000.11111110.00000001
 - ▣ Format décimal : 172 . 16 . 254 . 1

4. DNS

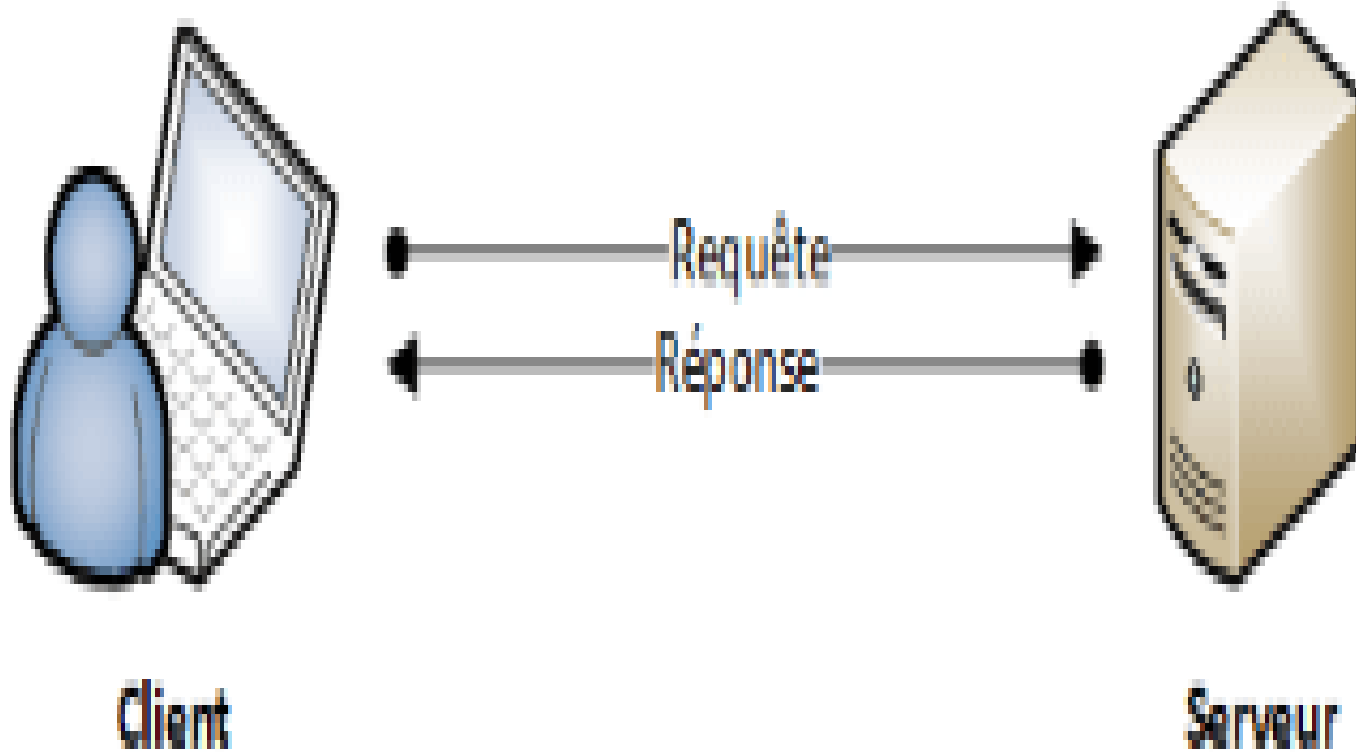
- Il est pratiquement impossible aux humains de connaître les adresses (IP) des machines auxquelles ils veulent accéder.
- Afin de résoudre ce problème, le système des noms de domaine ou **DNS (Domain Name System)** a été introduit.
- Le **DNS** est le **système d'adressage des serveurs** mis au point pour établir une **relation** entre **l'adresse numérique** d'un serveur (**adresse IP**) et une **adresse logique**.
- Le nom de domaine est plus facilement lisible : **www.google.com** est le nom de domaine correspondant **216.58.208.238**
- Le mécanisme consistant à trouver l'adresse IP correspondant au nom d'un hôte est appelé « **résolution de nom de domaine** ».

5. Services Internet

- L'utilisateur d'Internet, "l'internaute", a accès à de très **nombreux services** tels que :
 - Courrier électronique (protocole **SMTP**),
 - Transfert de fichiers (protocole **FTP**),
 - Connexion au serveur à distance (protocole **Telnet**),
 - la recherche d'informations (protocole **HTTP**) ..., etc.
- La **recherche d'informations** ou bien le **Web**, noté aussi **WWW** (**World Wide Web**) est un moyen d'accès et de partage d'information.
- Le Web utilise le protocole **HTTP** (HyperText Transfer Protocol) pour l'échange d'information entre le logiciel client et le serveur.

6. Mécanismes Client/ Serveur sur Internet

10



7. L'HyperText Transfert Protocol HTTP

- **Protocole de transfert hypertexte** est un **protocole de communication client-serveur** développé pour le *World Wide Web*.
- **HTTPS** (avec **S** pour *secured*, soit « **sécurisé** ») est la variante du HTTP *sécurisée*.
- **HTTP** est un protocole de la couche application.
- Un serveur **HTTP** utilise alors par défaut **le port 80** (**443 pour HTTPS**).
- **HTTP/HTTPS** est le protocole de navigation sur les pages Web.

8. Fonctionnement du Web

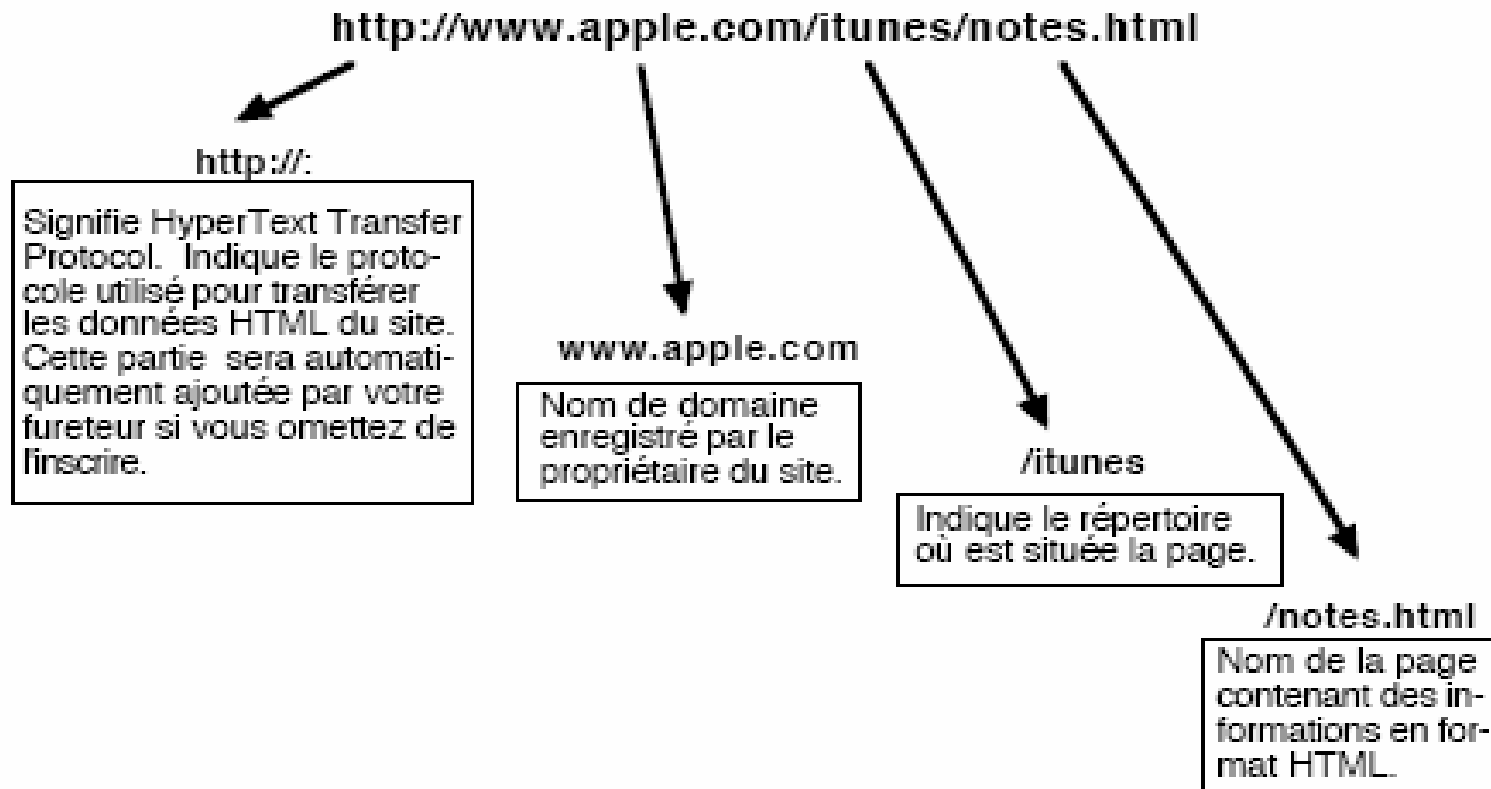
12



9. L'Uniform Resource Locator URL

13

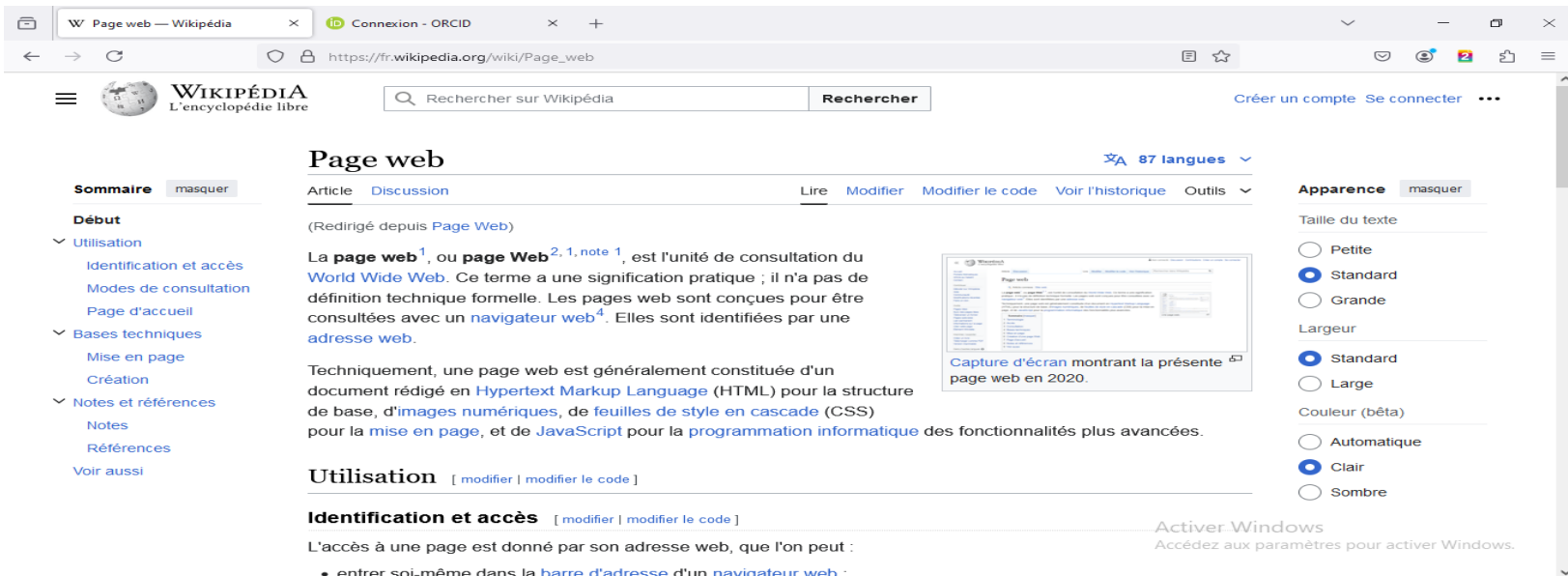
Une adresse URL (Uniform Resource Locator) peut-être composée de plusieurs parties. En voici les principales.



10. Page Web

14

- **Une ressource** du World Wide Web
- Créée par des webmasters à l'aide des **langages HTML/et CSS**
- Possède **une adresse Web (URL)**.
- Peut contenir du **texte**, des **images**, des **tableaux**, des **formulaires** et **autres éléments multimédias**.
- Visualisée par les **internauts (Clients)** grâce à des **navigateurs Web**.



The screenshot shows the French Wikipedia page for "Page web". The browser address bar displays "https://fr.wikipedia.org/wiki/Page_web". The page content includes a search bar, a table of contents, and a main text block. The main text explains that a "page web" is the unit of consultation on the World Wide Web, created using HTML and CSS, and accessed via a web browser. It also mentions technical details like Hypertext Markup Language (HTML) and JavaScript. A small inset image shows a screenshot of the page as it appeared in 2020. The right sidebar contains appearance settings like "Taille du texte" (Text size) and "Apparence" (Appearance).

Page web 87 langues

Article Discussion Lire Modifier Modifier le code Voir l'historique Outils

(Redirigé depuis [Page Web](#))

La **page web**¹, ou **page Web**^{2, 1, note 1}, est l'unité de consultation du **World Wide Web**. Ce terme a une signification pratique ; il n'a pas de définition technique formelle. Les pages web sont conçues pour être consultées avec un **navigateur web**⁴. Elles sont identifiées par une **adresse web**.

Techniquement, une page web est généralement constituée d'un document rédigé en **Hypertext Markup Language** (HTML) pour la structure de base, d'**images numériques**, de **feuilles de style en cascade** (CSS) pour la **mise en page**, et de **JavaScript** pour la **programmation informatique** des fonctionnalités plus avancées.

Utilisation [modifier | modifier le code]

Identification et accès [modifier | modifier le code]

L'accès à une page est donné par son adresse web, que l'on peut :

- entrer soi-même dans la **barre d'adresse** d'un **navigateur web** :

Activer Windows
Accédez aux paramètres pour activer Windows.

11. Site Web

15

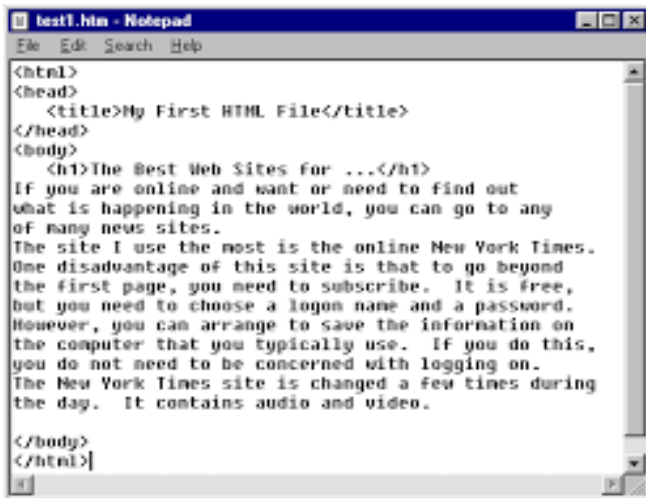
- **Un site web** (aussi appelé **site internet**) est un ensemble de pages Web hébergées sur un serveur Web connecté à Internet.
- **Un site web** est habituellement **architecturé autour d'une page centrale, appelée «page d'accueil»**.
- **Un site Web** propose des **liens** entre les différents **pages Web**.



12. Présentation sous le Web

16

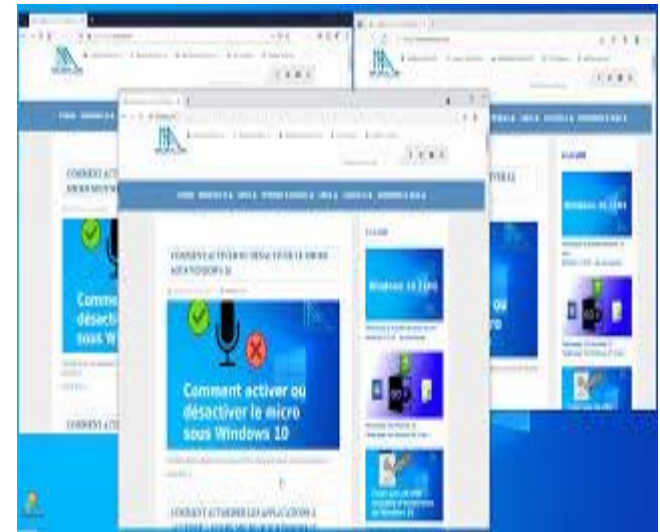
- Une page Web est le résultat de l'interprétation des documents Web (sous HTML/CSS, XML/XSLT, ...) sous un navigateur Web (Mozilla Firefox, Google Chrome, Opera, ...).
- HTML et CSS (Chapitres suivants) sont 2 langages de présentations permettant la présentation d'un contenu utilisateur sur Internet.



```
test1.htm - Notepad
File Edit Search Help
<html>
<head>
  <title>My First HTML File</title>
</head>
<body>
  <h1>The Best Web Sites for ...</h1>
  If you are online and want or need to find out
  what is happening in the world, you can go to any
  of many news sites.
  The site I use the most is the online New York Times.
  One disadvantage of this site is that to go beyond
  the first page, you need to subscribe. It is free,
  but you need to choose a logon name and a password.
  However, you can arrange to save the information on
  the computer that you typically use. If you do this,
  you do not need to be concerned with logging on.
  The New York Times site is changed a few times during
  the day. It contains audio and video.
</body>
</html>
```

Document HTML

Interprétation sous
navigateur



Page Web

CHAPITRE 2: HTML

Langage de présentation Web côté client

HTML

18

1. Généralités
2. Balises de structure
3. Balises de formatage et mise en page
4. Balises de listes
5. Images
6. Balises d'hyperliens
7. Tableaux
8. Cadres – Frames
9. Formulaires

1. Généralités

19

- **HTML = HyperText Markup Language**
- Langage de balisage Hyper-Texte
- Langage de présentation de documents sur le Web
- Un document HTML est un fichier ASCII (texte) composé de:
 - ▣ Contenu utilisateur
 - ▣ Balises
- Un navigateur (browser) permet d'interpréter les balises et d'afficher le document HTML : **Page Web.**

1. Généralités – Balises HTML

20

- Balise = Texte entouré par < et > (chevrons)
 - Une balise marque une action pour le navigateur : **Ce qu'il doit faire ?**
 - Balises insensibles à la casse :
 - <BALISE>
 - <balise>
 - <Balise>
 - <bAlIsE> } Équivalent
mais XHTML → minuscules
 - Ouvrante : <balise>
 - Fermante : </balise>
- } Encadrement d'un bloc

□ Non visibles dans le rendu du navigateur

□ Attributs/options : <balise attr="val" opt>

1. Généralités – Utilisation des balises

21

□ Utilisation CORRECTE

□ `<i>TexteTexte Texte</i>`



□ Utilisation INCORRECTE

□ `<i>TexteTexte Texte</i>`



□ Jamais de chevauchement

1. Généralités – Document HTML

22

- Structure d'un document HTML :
 1. Information sur la version de HTML utilisée
 2. En-tête déclaratif
 3. Corps, contenu du document

1.

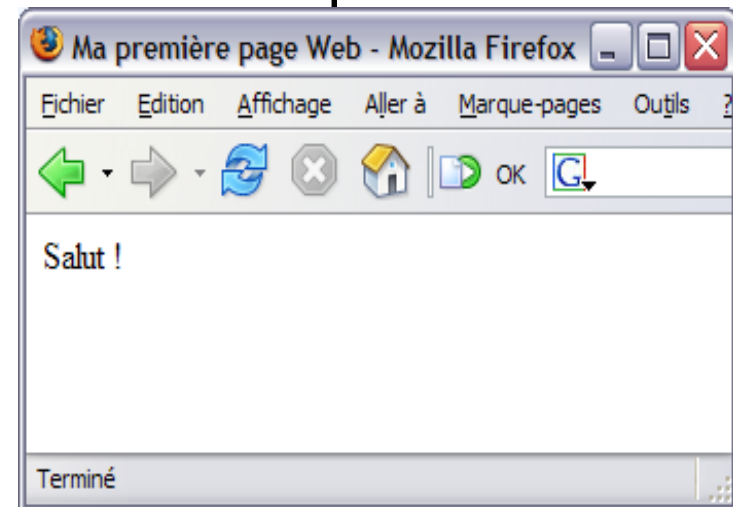
```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"  
"http://www.w3.org/TR/html4/strict.dtd">
```

2.

```
<html>  
<head>  
<title>Ma première page Web</title>  
</head>
```

3.

```
<body>  
Salut !  
</body>  
</html>
```



2. Balises de structure

23

- Racine du document :
 - ▣ `<html> ... </html>`
- En-tête du document :
 - ▣ `<head> ... </head>`
- Titre descriptif du document :
 - ▣ `<title> ... </title>`
- Corps du document :
 - ▣ `<body> ... </body>`

2. Balises de structure - Body

24

□ Syntaxe :

```
<body      Bgcolor="couleur d'arrière-plan"  
          Background="URL de l'image d'arrière-plan"  
          Link="couleur des liens non visités"  
          VLink="couleur des liens visités"  
          Text="couleur du texte" >  
    ...Corps du document  
</body>
```

- Chaque attribut de couleur peut prendre pour valeur l'un des 16 noms de couleurs réservés: **Black, White, Aqua, Silver, Gray, Maroon, Red, Purple, Fuschia, Green, Lime, Olive, Yellow, Navy, Blue et Teal**, ou un nombre de six chiffres hexadécimaux codant les intensités de RVB (exemple : **Blue=#0000FF**).

3. Balises de formatage et mise en page

25

- Gras : `` ... ``
- Italique : `<I>` ... `</I>`
- Texte souligné : `<U>` ... `</U>`
- Texte barré : `<S>` ... `</S>`
- Texte en indice : `_{` ... `}`
- Texte en exposant: `^{` ... `}`
- Saut de ligne : `
`
- Texte en centre : `<Center>` ... `</Center>`

3. Balises de formatage et mise en page

26

□ Titres (6 niveaux de hiérarchie) :

□ `<h1>` ... `</h1>`

□ `<h2>` ... `</h2>`

□ `<h3>` ... `</h3>`

□ `<h4>` ... `</h4>`

□ `<h5>` ... `</h5>`

□ `<h6>` ... `</h6>`

□ Syntaxe: `<Hn ALIGN = "LEFT/RIGHT/CENTER/JUSTIFY">` texte `</Hn>`

Où n=1,2,3,4,5 ou 6.

3. Balises de formatage et mise en page

27

□ Ligne horizontale : **<hr>**

□ Syntaxe :

**<hr ALIGN="alignement" SIZE= épaisseur
WIDTH=longueur >**

□ Paragraphe : **<p>** ... **</p>**

□ Syntaxe :

<p ALIGN="alignement" >
.... **Paragraphe** ...
</p>

3. Balises de formatage et mise en page

28

- Texte préformaté (brut) :
 - ▣ `<pre> ... </pre>`
 - Groupe « mode en ligne » : Applique un formatage spécial à un ensemble de mots.
 - ▣ ` ... `
 - Groupe « mode bloc » : applique un alignement spécial à un bloc de lignes.
 - ▣ `<div ALIGN="alignement" > ... </div>`
 - Mise en forme :
 - ``
... Texte ...
``
- Size : prend une valeur entre 1 et 7 (3 la valeur par défaut).

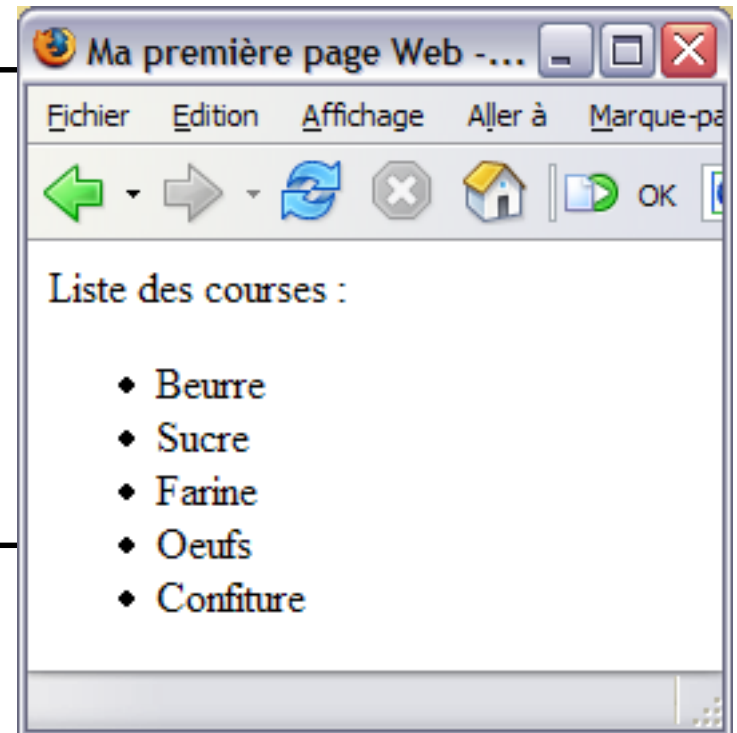
4. Balises de listes – liste non numérotée

29

- Liste : `<ul TYPE="DISC / SQUARE / CIRCLE" > ... `
- Élément : ` ... []`

Liste des courses :

```
<ul>  
<li>Beurre</li>  
<li>Sucre</li>  
<li>Farine</li>  
<li>Oeufs  
<li>Confiture  
</ul>
```



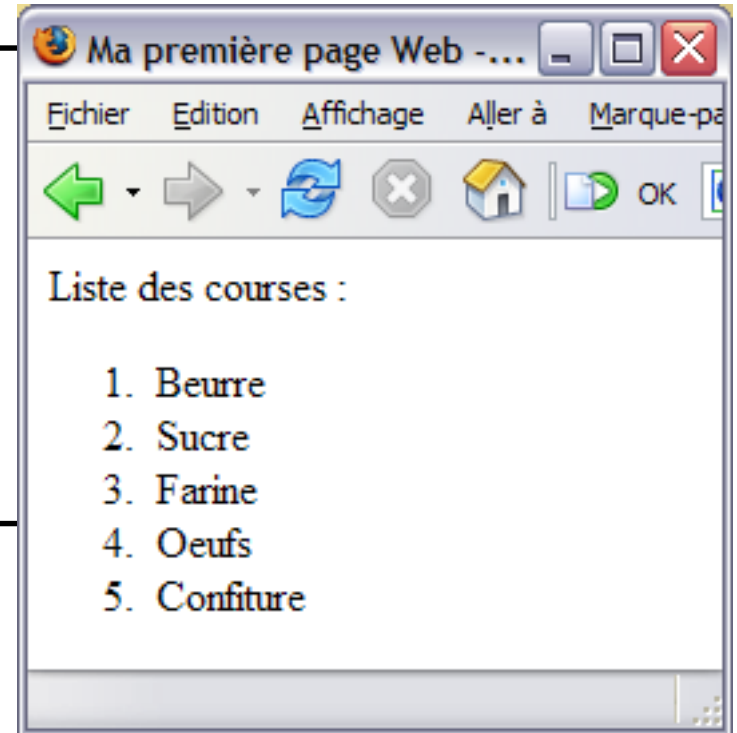
4. Balises de listes – liste numérotée

30

- Liste : `<ol TYPE="1 / A / a / I / i" START="valeur initiale" > ... `
- Élément : ` ... []`

Liste des courses :

```
<ol>  
<li>Beurre</li>  
<li>Sucre</li>  
<li>Farine</li>  
<li>Oeufs  
<li>Confiture  
</ol>
```



5. Les images

31

- `` : Ajouter une image au sein du document

Syntaxe :

```
<img SRC = "URL du fichier image"  
width = largeur height = hauteur  
alt = "texte alternatif"  
BORDER = épaisseur du cadre  
ALIGN = "TOP/MIDDLE/BOTTOM/LEFT/  
RIGHT"  
HSPACE = espacement horizontal  
VSPACE = espacement vertical
```

>

5. Les images

32

<i>Attribut</i>	<i>Fonction</i>
SRC	Spécifie l'URL (chemin) du fichier qui contient l'image.
WIDTH	Spécifie la largeur de l'image en pixels.
HEIGHT	Spécifie la hauteur de l'image en pixels.
BORDER	Définit l'épaisseur du cadre de l'image
ALT	Spécifie le texte de remplacement de l'image destiné pour les navigateurs Non graphiques ou dans lesquels l'utilisateur a désactivé le chargement des images.
ALIGN	Gère l'alignement du texte adjacent à l'image.
HSPACE	Définit l'espace qui sera réservé à droite et à gauche de l'image.
VSPACE	Définit l'espace qui sera réservé en haut et en bas de l'image.

5. Les images

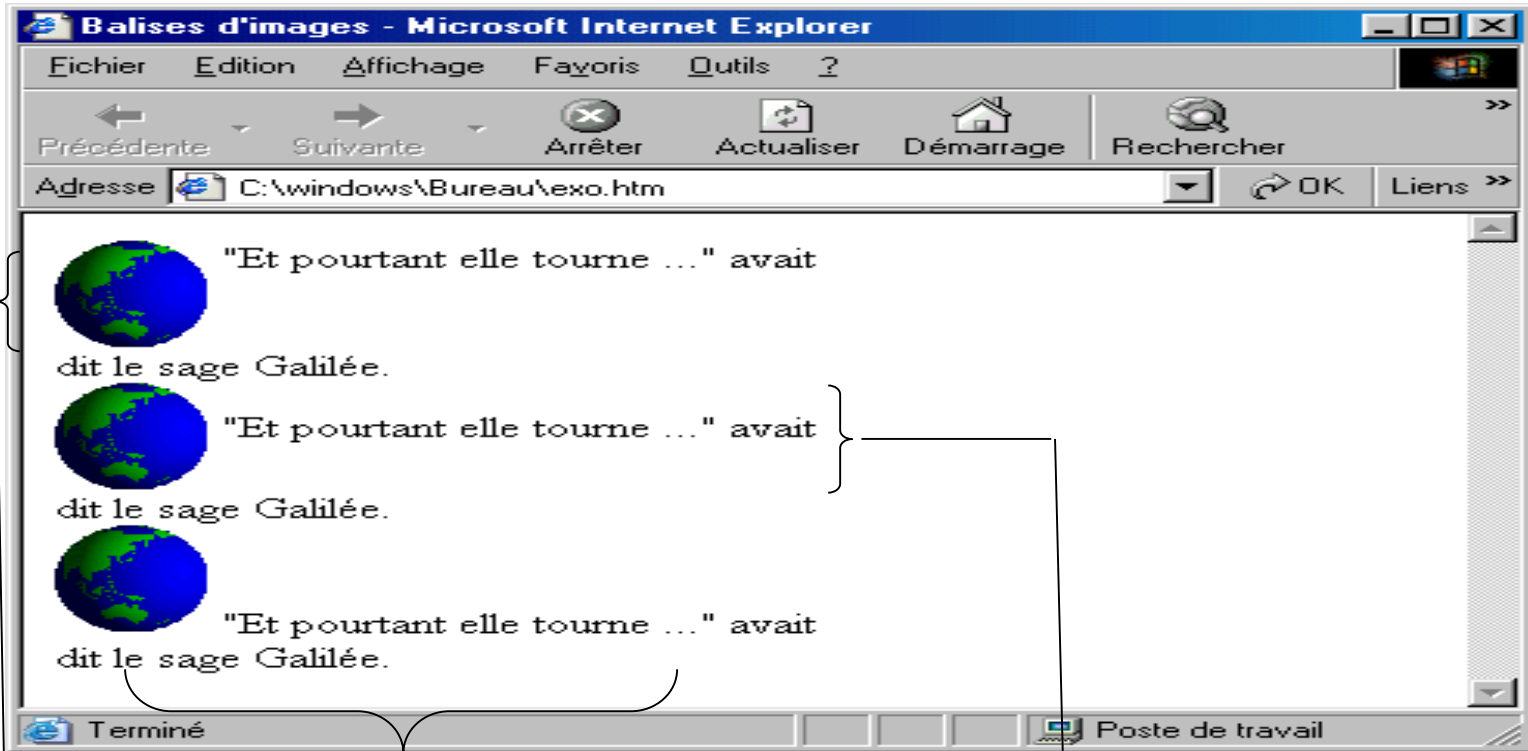
33

- Attribut **ALIGN** : Gère l'habillage de l'image par le texte et prend les valeurs indiquées dans le tableau ci-dessous

<i>Valeur</i>	<i>Fonction</i>
TOP	Aligne le haut du texte adjacent sur le sommet de l'image.
MIDDLE	Aligne la ligne de base du texte adjacent sur le milieu de l'image.
BOTTOM	Aligne la ligne de base du texte adjacent sur la base de l'image.(valeur par défaut)
LEFT	Aligne l'image sur la marge gauche et permet son habillage par le texte sur sa droite.
RIGHT	Aligne l'image sur la marge droite et permet son habillage par le texte sur sa gauche.

5. Les images

34



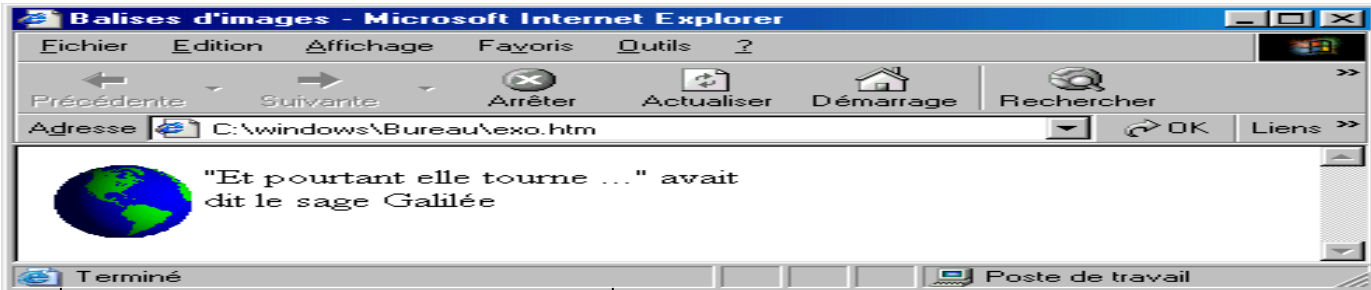
ALIGN= TOP

ALIGN= BOTTOM

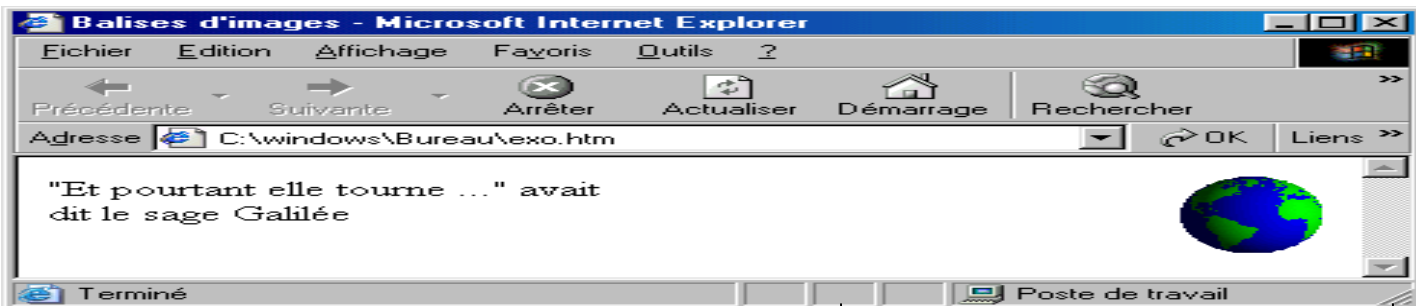
ALIGN= MIDDLE

5. Les images

35



ALIGN = LEFT



ALIGN = RIGHT

6. Balise d'hyperliens - Liens

36

- Base de la navigation hypertexte
- Lien : zone active cliquable
 - `support du lien`
 - Support : texte, image, contenus entre `<a>` et ``
 - **URL** : *Universal Resource Locator* (adresse externe de la destination)
 - **Adresse** : *adresse locale de la destination*

6. Balise d'hyperliens - Liens

37

- 3 types de liens :
 - ▣ **Lien externe** : destination dans un autre ordinateur accessible à travers une adresse URL.

Exemples d'URL :

- ✓ ***http://serveur/chemin.../fichier***
Pour accéder à des pages Web
- ✓ ***ftp://serveur/chemin.../fichier***
Pour faire un transfert de fichiers
- ✓ ***mailto:utilisateur@hôte***
Pour envoyer un mail

6. Balise d'hyperliens - Liens

38

- ▣ **Lien locale** : la destination se trouve dans le même dossier avec le document HTML.

Exemple d'appel de lien :

```
<a href="info.html" > Pour des  
Informations cliquer ici </a>
```

6. Balise d'hyperliens - Liens

39

- **Lien mixte** : destination se trouve dans un autre dossier de votre ordinateur.

Exemples d'adresses :

✓ **lecteur://chemin.../fichier**

Adressage absolu

✓ **../.../.../fichier**

Adressage relatif

6. Balise d'hyperliens - Ancres

40

- Ancre (nommée) : point cible
 - ▣ `point d'ancrage`
 - ▣ `name=` nom unique (à la charge du concepteur)
- Cibler l'ancre dans un lien : URL/adresse se terminant par `#nom` de l'ancre.

Exemple :

- ✓ `Cibler ancre`
- ✓ `Cibler ancre`
- ✓ `Cibler ancre`

7. Tableaux

41

- Tableau : `<table> ... </table>`
- Ligne : `<tr> ... [</tr>]`
- Cellule : `<td> ... [</td>]`
- Cellule d'en-tête : `<th> ... [</th>]`
- Ligne `<tr>` = Ensemble de colonnes (cellules `<td>`)

7. Tableaux

42

□ Structure d'un tableau :

```
<TABLE>
  <TR>
    <TH> En-tête 1 </TH>
    ...
    <TH> En-tête n </TH>
  </TR>
  <TR>
    <TD> Élément de colonne 1 </TD>
    ...
    <TD> Élément de colonne n </TD>
  </TR>
  ...
  <TR>
    <TD> Élément de colonne 1 </TD>
    ...
    <TD> Élément de colonne n </TD>
  </TR>
</TABLE>
```

En-tête 1	En-tête n
Cellule 1	Cellule n
Cellule 1	Cellule n

Ligne 1

Ligne 2

Ligne m

7. Tableaux

43

□ Exemple de tableau :

```
<TABLE>
  <TR>
    <TH> Matière </TH>
    <TH> Trimestre </TH>
    <TH> Volume Horaire </TH>
  </TR>
  <TR>
    <TD> HTML </TD>
    <TD> 1 </TD>
    <TD> 20 </TD>
  </TR>
  <TR>
    <TD> JavaScript </TD>
    <TD> 2 </TD>
    <TD> 30 </TD>
  </TR>
  <TR>
    <TD> ASP </TD>
    <TD> 2 </TD>
    <TD> 40 </TD>
  </TR>
</TABLE>
```

Matière	Trimestre	Volume Horaire
HTML	1	20
JavaScript	2	30
ASP	2	40

7. Tableaux

44

- Légende d'un tableau :

```
<TABLE>
```

```
  <CAPTION> texte de légende
```

```
  </CAPTION>
```

```
  ...
```

```
</TABLE>
```

- Exemple :

```
<TABLE WIDTH= 100% BORDER= 2>
```

```
  <CAPTION >
```

```
    <FONT SIZE=4 COLOR="BLUE">
```

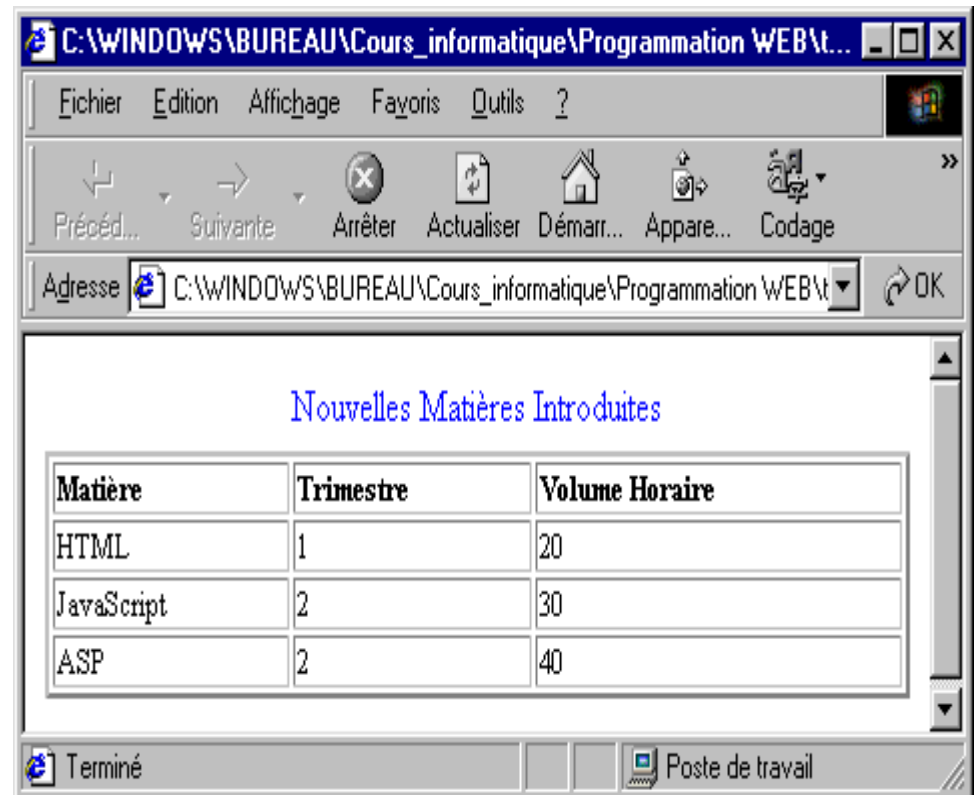
```
      Nouvelles Matières Introduites
```

```
    </FONT>
```

```
  </CAPTION>
```

```
  ....
```

```
</TABLE>
```



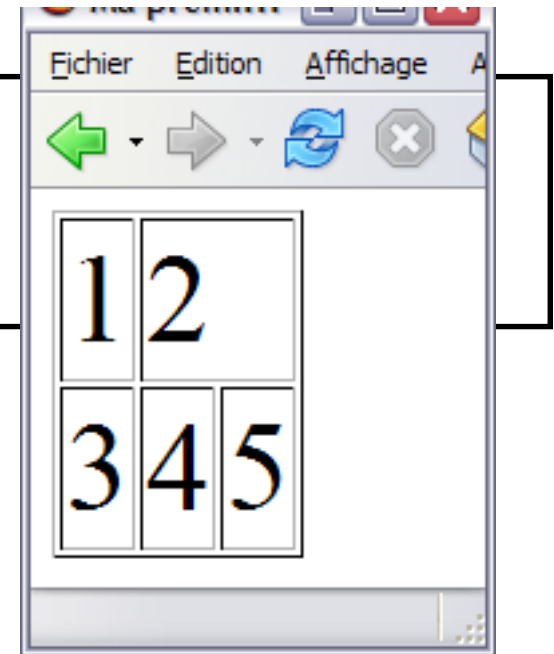
7. Tableaux – Fusion de cellules

45

- Fusion Horizontale : Extension de cellule sur C colonnes :

- `<td colspan="C">`

```
<table>  
<tr> <td>1 <td colspan=2>2 <!-- suite -->  
<tr> <td>3 <td>4 <td>5  
</table>
```



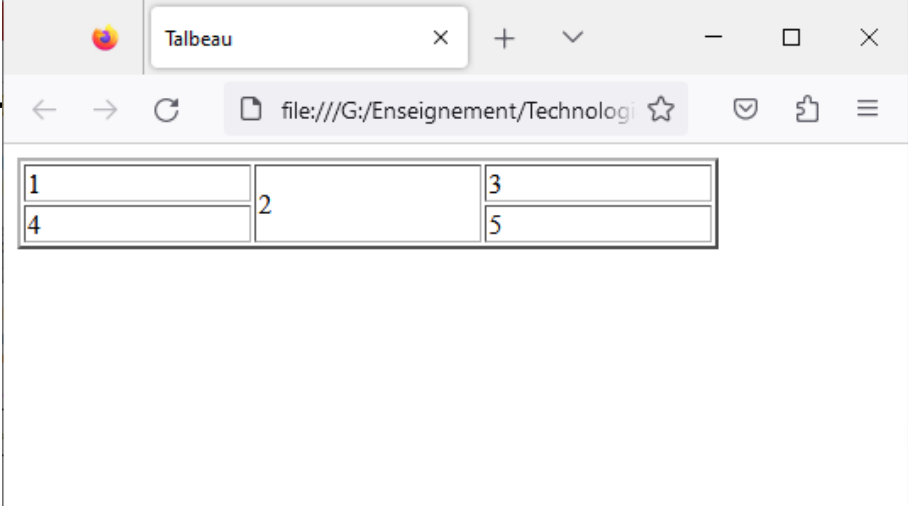
7. Tableaux – Fusion de cellules

46

- Fusion Verticale : Extension de cellule sur L lignes :

- `<td rowspan="L">`

```
<table>
<tr> <td>1 <td rowspan=2>2 <td>3
<tr> <td>4 <td>5
</table>
```



The screenshot shows a web browser window titled "Tableau" with a file path in the address bar: `file:///G:/Enseignement/Technologi`. The browser displays a table with the following content:

1	2	3
4		5

8. Cadres - Frames

47

- Diviser la fenêtre du navigateur en cadres :
 - Division horizontale :
`<frameset rows="liste"> ... </frameset>`
 - Division verticale :
`<frameset cols="liste"> ... </frameset>`
 - *Liste* : espacements des colonnes ou des lignes
 - en pixels : 30 , en % : 20% , ce qu'il reste : *

- `<frameset>` peut contenir :
 - `<frameset>` : Cadres imbriqués
 - `<frame>` : définit ce qui doit être affiché dans le cadre.

8. Cadres

48

- Syntaxe de `<frame>`:

```
<frame name="nom" src="URL">
```

- `name=`

- Nom du cadre / de la fenêtre
- Utilisé pour qu'il soit la cible de liens :
- ` ... `

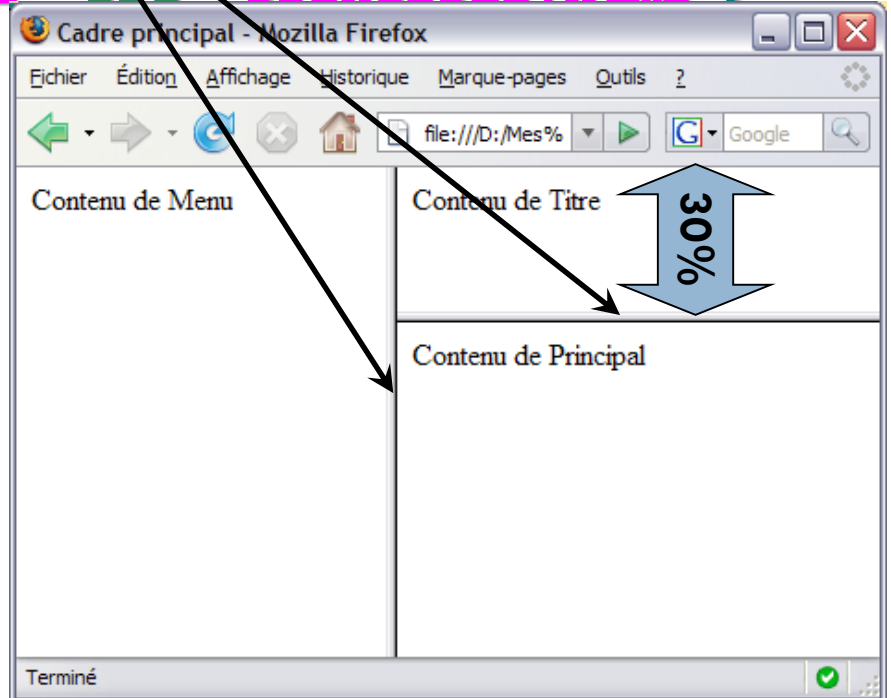
- `src=`

- URL de la page à afficher dans le cadre

8. Cadres - Exemple

49

```
<frameset cols="200,*">  
  <frame name="menu" src="menu.html">  
  <frameset rows="30%,*">  
    <frame name="titre" src="titre.html">  
    <frame name="principal" src="principal.html">  
  </frameset>  
</frameset>
```



9. Formulaire

- Ajoute une interactivité entre la page Web et l'utilisateur.
- Un formulaire est composé d'un ensemble de champs de contrôles.
- L'utilisateur entre ses données par le biais des champs de contrôles.
- Le navigateur rassemble ces données, ouvre une connexion HTTP et les transmet au serveur.
- Le serveur traitera, ensuite, les données reçues et envoie la réponse à l'utilisateur sous forme d'une page HTML.
- Les données transmises par un formulaire sont traitées par un script ASP ou PHP.

9. Formulaire

51

- Création de formulaire :
 - `<form> ... </form>`
- Attributs :

<i>Attribut</i>	<i>Fonction</i>
ACTION	Spécifie l'URL du script qui doit traiter les données du formulaire.
METHOD	Indique la méthode HTTP employée pour transmettre les données au serveur. Les valeurs possibles sont: GET (valeur par défaut) ou POST .
TARGET	Indique le cadre vers lequel la réponse du script doit être chargée.
NAME	Spécifie le nom du formulaire.
ENCTYPE	Spécifie le type du fichier envoyé.
ACCEPT	Spécifie la liste des contenus acceptables par le serveur de traitement.
ACCEPT-CHARSET	Spécifie la liste des jeux de caractères acceptables par le script de traitement.

9. Formulaires

52

- Composants du formulaire :
 - Zones de texte.
 - Cases à cocher.
 - Boutons d'options.
 - Zones de liste.
 - Boutons de commandes.

9. Formulaires

53

- Balises permettant la création des composants du formulaire :

<i>Type de Champ</i>	<i>Balise HTML</i>
Champ de saisie	<code><INPUT TYPE = "TEXT"></code>
Champ de mot de passe	<code><INPUT TYPE = "PASSWORD"></code>
Zone de texte multilignes	<code><TEXTAREA></TEXTAREA></code>
Fichier	<code><INPUT TYPE = "FILE"></code>
Case à cocher	<code><INPUT TYPE = "CHECKBOX"></code>
Bouton d'option	<code><INPUT TYPE = "RADIO"></code>
Zone de liste	<code><SELECT> ...<OPTION> ...</SELECT></code>
Bouton de commande de validation	<code><INPUT TYPE = "SUBMIT"></code>
Bouton de commande d'initialisation	<code><INPUT TYPE = "RESET"></code>
Bouton de commande "scripté"	<code><INPUT TYPE = "BUTTON"></code>

9. Formulaire – Zone de texte

54

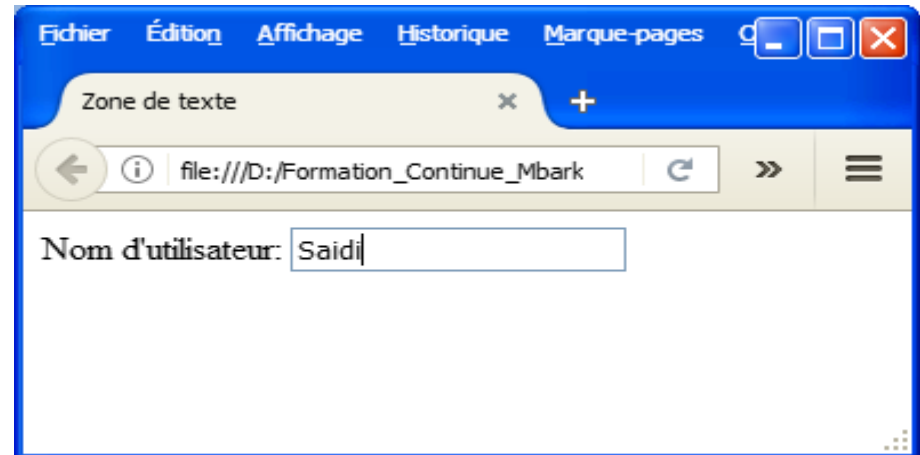
□ Syntaxe :

```
<INPUT type="text" name="name"  
value="text_par_defaut"  
size=longueur  
maxlength=longueur_maximale >
```

□ Exemple :

Nom d'utilisateur :

```
<INPUT TYPE = TEXT  
NAME = "nom_login"  
SIZE = 20 MAXLENGTH= 15>
```



9. Formulaire – Zone mot de passe

55

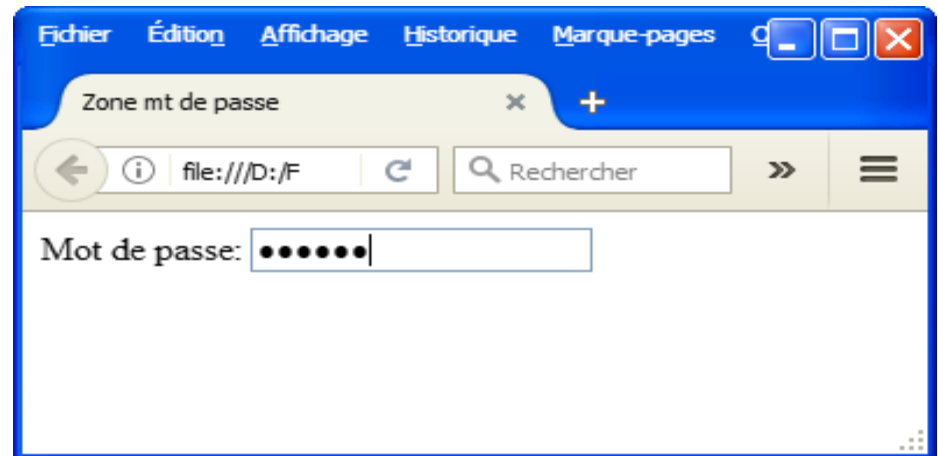
□ Syntaxe :

```
<INPUT type=« password" name="name"  
size=longueur  
maxlength=longueur_maximale >
```

□ Exemple :

Mot de passe :

```
<INPUT TYPE = PASSWORD  
NAME = "login"  
SIZE = 20 MAXLENGTH= 15>
```



9. Formulaire – Zone -multilignes

56

□ Syntaxe :

```
<TEXTAREA name="name"  
    rows=Nb_lig cols=Nb_col >
```

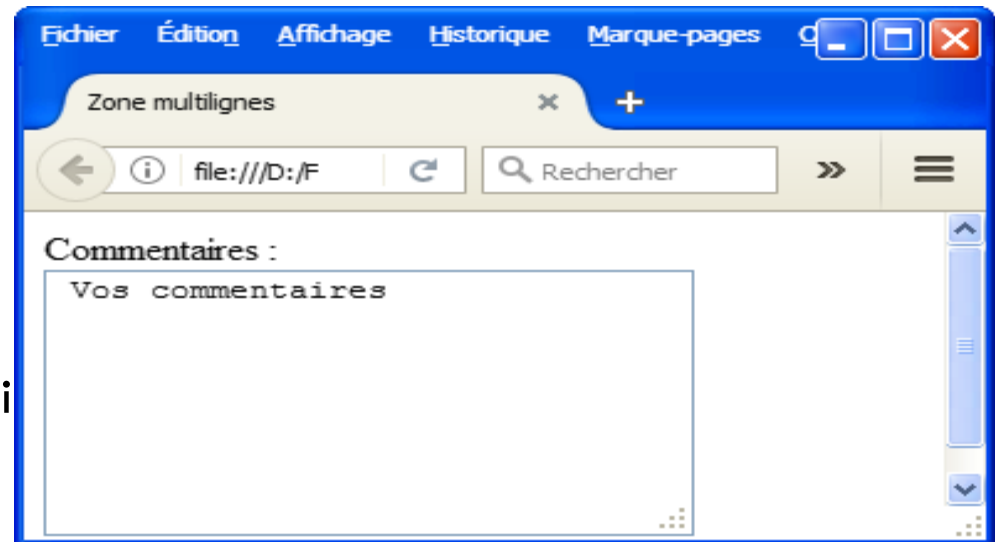
.... Texte par défaut

```
</TEXTAREA>
```

□ Exemple :

Commentaires :


```
<Textarea NAME = "login"  
rows = 7 cols= 30> Vos commentai  
</Textarea>
```



9. Formulaire – Fichier

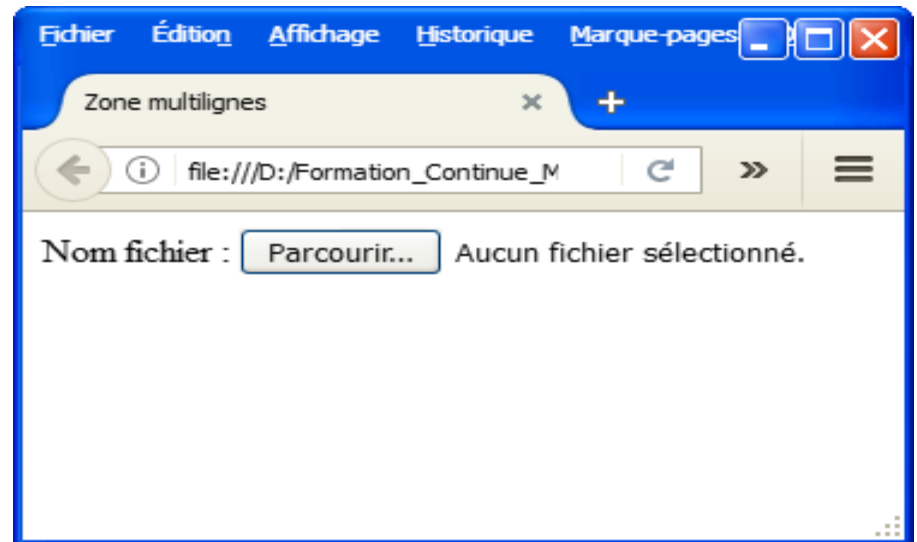
57

- Envoyer un fichier via un formulaire
- Syntaxe :
`<INPUT type= "FILE" name="name">`
- Navigateur offre un bouton *Parcourir...*

- Exemple :

Nom fichier :

```
<INPUT TYPE = FILE  
      NAME = "Fich" >
```



9. Formulaire – Cases à cocher

58

□ Syntaxe :

```
<INPUT type= "CHECKBOX" name="name"  
value=valeur [Checked]>
```

□ Exemple :

```
<INPUT TYPE = CHECKBOX NAME = "BD"
```

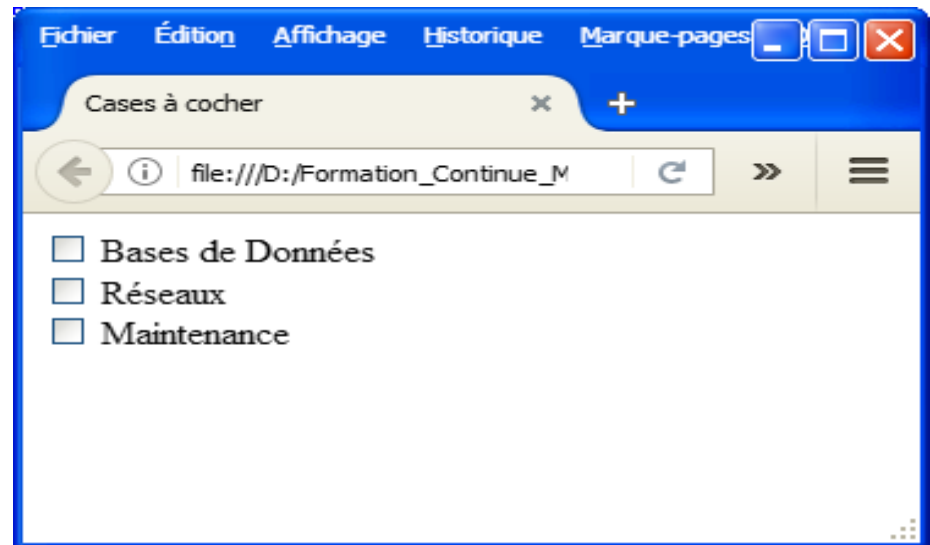
```
VALUE = "1"> Bases de Données <BR>
```

```
<INPUT TYPE = CHECKBOX NAME = "RES"
```

```
VALUE = "2"> Réseaux <BR>
```

```
<INPUT TYPE = CHECKBOX NAME = "MT"
```

```
VALUE = "3"> Maintenance <BR>
```



9. Formulaire – Boutons d'options

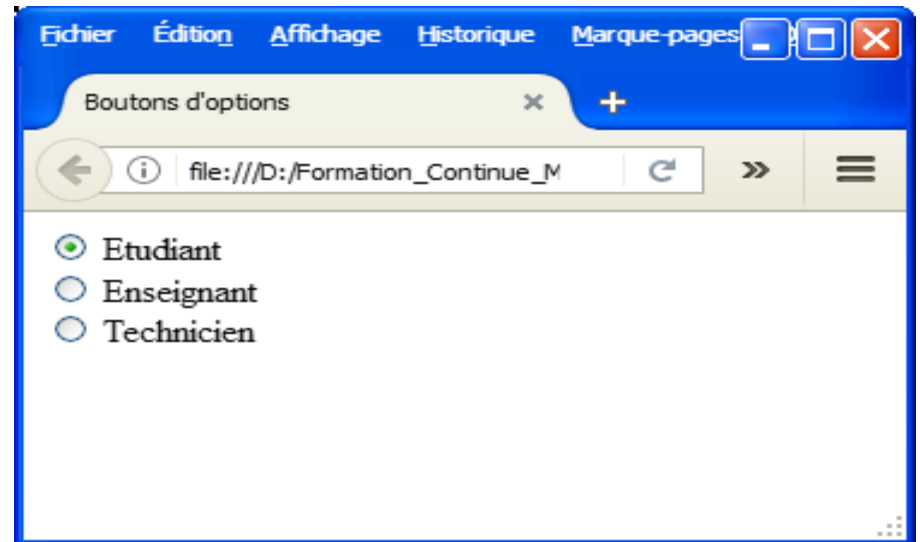
59

□ Syntaxe :

```
<INPUT type= "OPTION" name="name"  
value=valeur [Checked]>
```

□ Exemple :

```
<INPUT TYPE = RADIO NAME = "ACT"  
VALUE = "etudiant" CHECKED > Etudiant <BR>  
<INPUT TYPE = RADIO NAME = "ACT"  
VALUE = "prof" > Enseignant <BR>  
<INPUT TYPE = RADIO NAME = "ACT"  
VALUE = "tech" > Technicien <BR>
```



9. Formulaire – Liste d'options

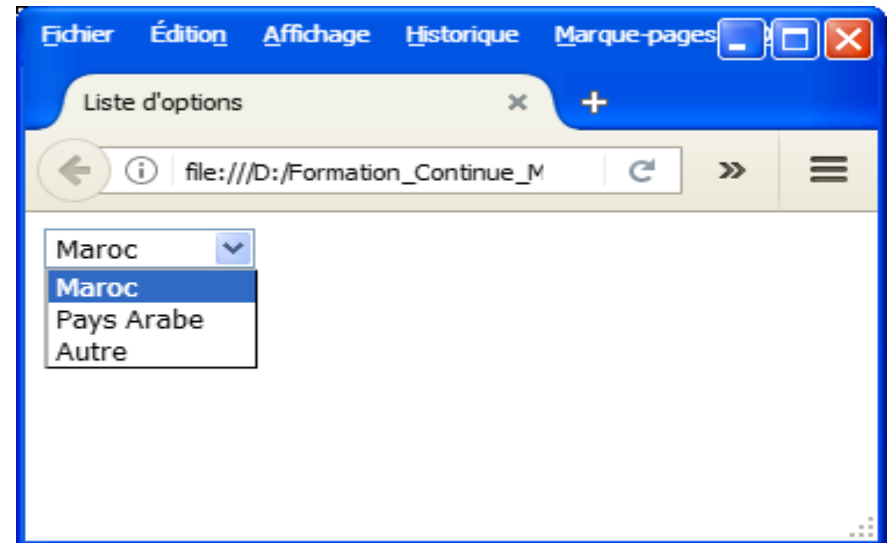
60

□ Syntaxe :

```
<Select name="name">  
  <Option value=valeur1> Option 1 </Option>  
  .....  
  <Option value=valeurn> Option n </Option>  
</Select>
```

□ Exemple :

```
<SELECT NAME = "pays" >  
  <OPTION VALUE = "ma" SELECTED > Maroc </OPTION>  
  <OPTION VALUE = "arabe" > Pays Arabe </OPTION>  
  <OPTION VALUE ="autre" > Autre </OPTION>  
</SELECT>
```



9. Formulaires – Boutons de commande

61

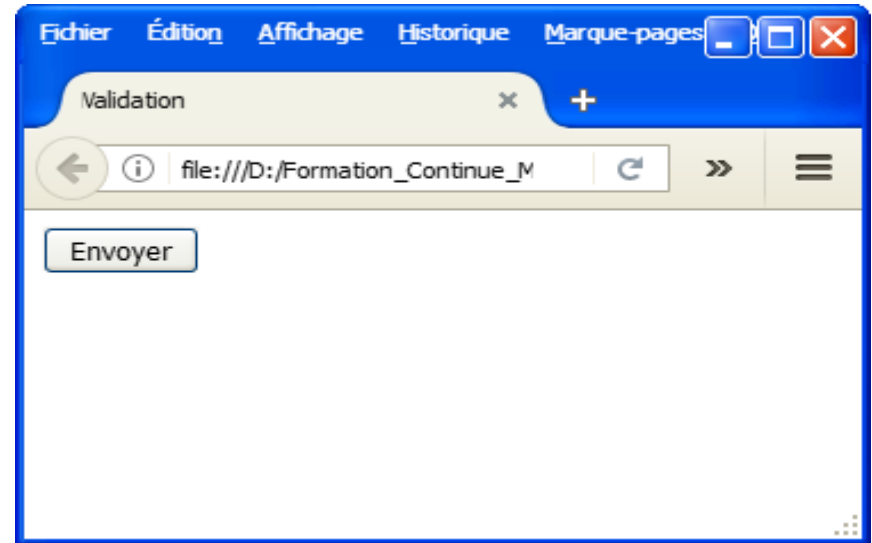
- **Bouton de validation** : Permet l'envoi des données du formulaire au serveur.

- Syntaxe :

```
<Input type="Submit" name="nom"  
value=valeur >
```

- Exemple :

```
<INPUT TYPE = "SUBMIT" VALUE = "Envoyer">
```



9. Formulaire – Boutons de commande

62

- **Bouton d'initialisation** : Les champs reprennent leurs valeurs par défaut.

- Syntaxe :

```
<Input type=« Reset" name="nom"  
value=valeur >
```

- Exemple :

```
<INPUT TYPE = "Reset" VALUE = "Initialiser">
```



9. Formulaire – Boutons de commande

63

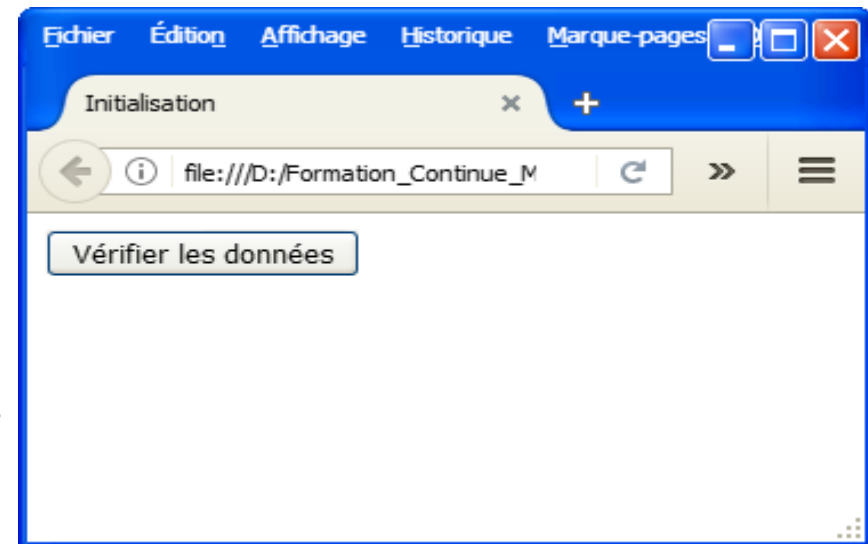
- **Bouton scripté** : déclencher un traitement en utilisant un script côté client (en JavaScript, VBScript...).

- Syntaxe :

```
<Input type= "Button" name="nom"  
value=valeur  
Onclick= "fonction" >
```

- Exemple :

```
<INPUT TYPE =BUTTON NAME ="verifier"  
VALUE ="Vérifier les données" OnCilck="verif()">
```



Séparation du contenu de la mise en forme

CSS

65

1. Introduction
2. Définition d'un style
3. Style interne/externe
4. Classe et ID
5. SPAN et DIV
6. Positionner avec CSS
7. Liste des propriétés CSS

1. Introduction

66

- HTML des débuts :
 - Mise en forme figée, très lourde à modifier
 - Mise en forme peu portable
 - Pages complexes
 - Nécessité d'une méthode alternative

1. Introduction

67

- Avec **CSS = Cascading Style Sheets**
 - Positionnement flexible des styles:
 - Fichier séparé
 - Au début du document
 - Dans les balises à mettre en forme
 - Styles alternatifs
 - Modification de style très aisée

2. Définition d'un style

68

□ Définition de base d'un style :

balise { propriété de style: valeur; propriété de style: valeur; ... }

□ Exemple :

H3 { font-family: Arial; font-style: italic }

□ Quelques règles :

- Les feuilles de style portent sur des balises ou attributs,
- Les propriétés de style sont entourées par des accolades "{ }",
- Le couple "propriété de style/valeur" est séparé par un double-point (:),
- Chaque couple "propriété de style/valeur" est séparé par un point-virgule (;),
- Il n'y a pas de limite pour le nombre de couples "propriétés de style/valeur",

2. Définition de style

- L'espace entre propriétés de style et valeur n'est pas obligatoire mais aide fortement à la lisibilité du code source,
- Pour la lisibilité toujours, vous pouvez écrire vos styles sur plusieurs lignes :

```
H3 { font-family: Arial;  
      font-style: italic;  
      color: green  
    }
```

- On peut attribuer plusieurs valeurs à une même propriété. Dans ce cas, on séparera les différentes valeurs par des virgules :

```
H3 {font-family: Arial, Helvetica, sans-serif}
```

- On peut attribuer un même style à plusieurs balises (séparées par des virgules) :

```
H1, H2, H3 {font-family: Arial; font-style: italic}
```

3. Style interne / externe

70

- Deux façons différentes pour la définition du style CSS :

- ▣ **Style interne :**

- A l'intérieur des balises <HEAD> </HEAD> :

- Syntaxe :

```
<HTML>
  <HEAD>
    <STYLE type="text/css">
      <!--
        La ou les feuille(s) de style
      -->
    </STYLE>
  </HEAD>
  <BODY> .... </BODY>
</HTML>
```

3. Style interne / externe

- La balise `<STYLE>` avertit le navigateur que l'on va utiliser des feuilles de style.
- L'attribut `type="text/css"` informe que ce qui suit est du texte et qu'il s'agit de cascading style sheets (css).
- La balise Html de commentaires `<!-- ... -->` empêche que les browsers qui ne connaissent pas les feuilles de style, tentent d'interpréter ces instructions. Les informations à l'intérieur des tags de commentaires seront ignorées par ces browsers.
- Pour vos propres commentaires à propos des feuilles de style, on utilisera la convention désormais classique (C, C++, Javascript...) de `/* commentaires */`.

3. Style interne / externe

72

- A l'intérieur des balises `<BODY> </BODY>` :

- Le style est défini à l'intérieur de chaque balise,

- Syntaxe :

```
<BODY>
```

```
  <balise style="propriété: valeur; propriété: valeur"> ...
```

```
  </balise>
```

```
</BODY>
```

- Exemple :

```
<BODY>
```

```
<H1 style="font-family: Arial; font-style: italic"> blabla </H1>
```

```
</BODY>
```


3. Style interne / externe

73

▣ Style externe :

- Définir une page externe d'extension .css regroupant toutes les feuilles de styles,
- Reliant les pages Web d'un site à cette page de style externe, sous la syntaxe :

```
<HTML>  
  <HEAD>  
    <LINK rel=stylesheet type="text/css" href="styles.css">  
  </HEAD>  
  <BODY>  
  </BODY>  
</HTML>
```

3. Style interne / externe

- La balise `<LINK>` avertit le browser qu'il faudra réaliser un lien.
- L'attribut `rel=stylesheet` (sans s et sans guillemets) précise qu'il y trouvera une feuille de style externe.
- L'attribut `type="text/css"` précise que l'information est du texte et du genre cascading style sheets (css).
- L'attribut classique de lien `href=" ... "` donne le chemin d'accès et le nom du fichier à lier.

4. Classes et ID

75

- Notion de classes :
 - Définir des styles différents pour la même balise
 - Création d'une classe de style :
 - *balise.nom_de_classe { propriété de style: valeur } OU*
 - *.nom_de_classe { propriété de style: valeur }*
 - Appel de la classe de style dans la balise :
 - *<balise class="nom_de-classe"> ... </balise>*
 - Exemple :
 - .essentiel { font-weight: bold; color: #000080 }*
 - <P class=essentiel> ... blabla ... </P>*
 - <H1 class=essentiel>Titre 1</H1>*
 - <TABLE>...<TD class=essentiel>cellule</TD>....</TABLE>*

4. Classes et ID

76

□ Notion des ID :

- Les ID fonctionnent exactement comme les classes,

- Syntaxe :

#nom_de_ID { propriété de style: valeur }

Pour l'appeler : <balise id="nom_de_ID"> ... </balise>

- Exemple :

essentiel { font-weight: bold; color: #000080 }

<P id="essentiel"> ... blabla ... </P>

5. SPAN et DIV

77

- ** ... ** :
 - ▣ permet d'appliquer des styles à des éléments de texte d'un paragraphe.
 - ▣ Exemple : Je voudrais écrire :

Un monde de **géants**.

```
<HTML>
  <HEAD>
    <STYLE type="text/css">
      .element {font-size: x-large; color: red}
    </STYLE>
  </HEAD>
  <BODY>
    <P>Un monde de <SPAN class=element>géants</SPAN>.</P>
  </BODY>
</HTML>
```

5. SPAN et DIV

78

- **<DIV> ... </DIV> :**
 - Permet d'appliquer un style à un groupe de paragraphes.
 - Exemple : Pour écrire :

La balise <DIV>

Commentaire :

N'oubliez pas l'attribut class!

```
<HTML>
  <HEAD>
    <STYLE type="text/css">
      .zone {font-size: x-small}
    </STYLE>
  </HEAD>
  <BODY>
    La balise
    <DIV class=zone>
      <P>Commentaire :</P>
      <P>N'oubliez pas l'attribut class!</P>
    </DIV>
  </BODY>
</HTML>
```

6. Positionner avec CSS

79

- **Position absolue** : $\{position: absolute\}$
 - se détermine par rapport au coin supérieur gauche de la fenêtre du browser,
 - Les coordonnées de ce point sont $top = 0$ et $left = 0$,



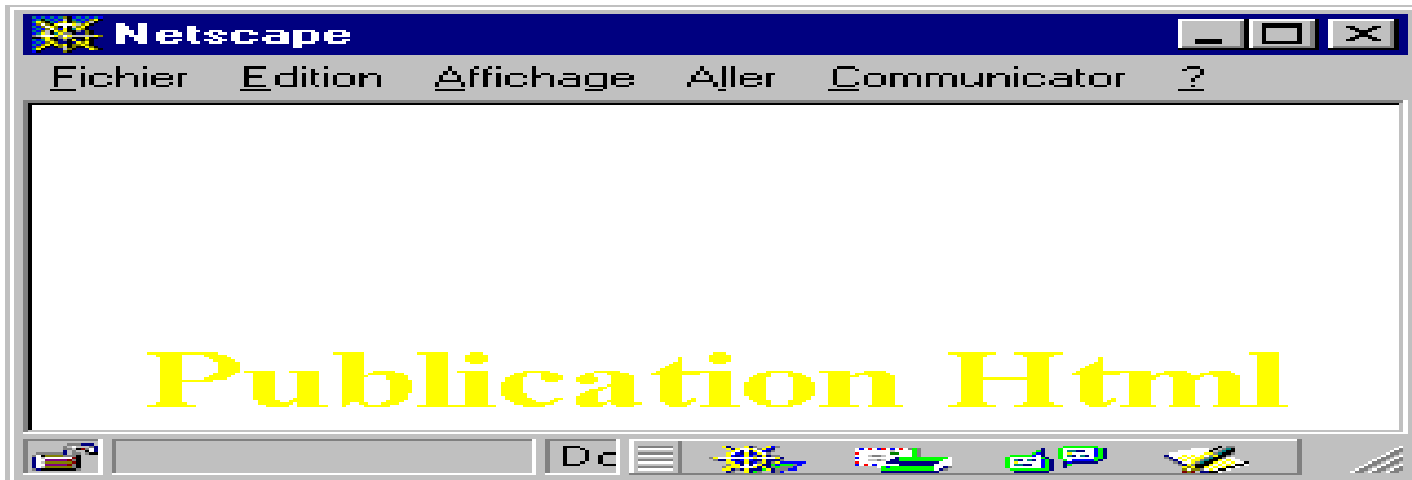
- **Position relative** : $\{position: relative\}$
 - se détermine par rapport à d'autres éléments de la page, par exemple un élément du code Html (image).

6. Positionner avec CSS

80

□ Positionner du texte :

- Exemple : Plaçons en position absolue le texte "Publication Html" à 100 pixels du haut de la fenêtre (*top*) et à 25 pixels à gauche (*left*).



6. Positionner avec CSS

81

```
<HTML>
  <HEAD>
    <STYLE type="text/css">
      .pub{position: absolute; top: 100px; left: 25px;
        color: red; font-size: 48; font-weight: bold;}
    </STYLE>
  </HEAD>
  <BODY>
    <DIV class=pub> Publication Html </DIV>
  </BODY>
</HTML>
```

6. Positionner avec CSS

82

□ Positionner une image :

- Exemple : Plaçons l'image **htmlplus.gif** en position absolue à 50 pixels de haut de la fenêtre (*top*) et à 100 pixels à gauche (*left*). Les dimensions de l'image sont *width=242* et *height=84*.



6. Positionner avec CSS

83

```
<HTML>
```

```
<BODY>
```

```
<span style="position: absolute; top: 50px; left: 100px; width:  
242px; height: 84px;">
```

```
<IMG src="htmlplus.gif">
```

```
</span>
```

```
</BODY>
```

```
</HTML>
```

6. Positionner avec CSS

84

- **Superposer du texte sur une image :**
 - Exemple : Reprenons l'image **htmlplus.gif** et on y superposera le nom de l'auteur de ce tutorial, au pixel près dans la barre qui souligne le terme Html.



6. Positionner avec CSS

85

```
<HTML>
```

```
<BODY>
```

```
<span style="position: absolute; top: 50px; left: 100px; width:  
242px; height: 84px;">
```

```
<IMG src="htmlplus.gif">
```

```
</span>
```

```
<span style="position: absolute; top: 96px; left: 145px;  
color: yellow; font-size: x-small; font-weight: bold;">
```

```
Van Lancker Luc
```

```
</span>
```

```
</BODY>
```

```
</HTML>
```

7. Liste des propriétés CSS

86

□ **Styles de police :**

□ **Font-family :**

- Définit un nom de police ou une famille de police précise : *Arial, Times, Helvetica, ...*
- Exemple : *H3 { font-family : Arial }*

□ **Font-style :**

- Définit le style de l'écriture *normal* ou *italique* ou *oblique*
- Exemple : *H2 { font-style : italic }*

□ **Font-weight :**

- Définit l'épaisseur de la police *normal* ou *bold* ou *bolder* ou *lighter* ou valeur numérique soit *(100 | 200 | 300 | 400 | 500 | 600 | 700 | 800 | 900)*
- Exemple : *P { font-weight : bold }*

7. Liste des propriétés CSS

87

□ Font-size :

- Définit la taille de la police *xx-small* ou *x-small* ou *small* ou *medium* ou *large* ou *x-large* ou *xx-large* ou taille précise en points (*pt*) ou en *inches* (*in*) ou en *centimètres* (*cm*) ou en *pixels* (*px*).
- Exemple : `P { font-size : 12pt }`

□ **Styles du texte** :

□ text-align :

- Définit l'alignement du texte *left* ou *center* ou *right*
- Exemple : `H1 { text-align : center }`

7. Liste des propriétés CSS

88

□ text-indent :

- Définit un retrait dans la 1^{ère} ligne d'un bloc de texte, souvent utilisé avec `<P></P>` spécifié en **inches (in)** ou en **centimètres (cm)** ou en **pixels (px)**
- Exemple : `P{ text-indent : 1 cm }`

□ text-decoration :

- Définit une décoration du texte : **overline**, **underline**, ...
- Exemple : `P{ text-decoration : overline }`

□ text-transform :

- Définit la casse du texte (majuscule, minuscule), **uppercase** ou **lowercase** ou **capitalize**.
- Exemple : `P{ text-transform : uppercase }`

7. Liste des propriétés CSS

89

□ Color :

- Définit la couleur du texte en **hexadécimale** ou couleur (**red, blue, green, yellow, white, black, ...**)
- Exemple : `H1 {color : blue}`

□ word-spacing :

- Définit l'espace entre les mots en **points (pt)** ou **inches (in)** ou **centimètres (cm)** ou **pixels (px)** ou **pourcentage (%)**
- Exemple : `P { word-spacing : 5pt}`

□ letter-spacing :

- Définit l'espace entre les lettres en **points (pt)** ou **inches (in)** ou **centimètres (cm)** ou **pixels (px)** ou **pourcentage (%)**
- Exemple : `P { letter-spacing : 2pt}`

7. Liste des propriétés CSS

□ line-height :

- Définit l'interligne en **points (pt)** ou **inches (in)** ou **centimètres (cm)** ou **pixels (px)** ou **pourcentage (%)**
- Exemple : `P{ line-height : 12pt }`

□ Width :

- Définit la largeur d'un élément du texte ou d'une image en **points (pt)** ou **inches (in)** ou **centimètres (cm)** ou **pixels (px)** ou **pourcentage (%)**
- Exemple : `IMG{width : 250px }`

□ Height :

- Définit la hauteur d'un élément du texte ou d'une image en **points (pt)** ou **inches (in)** ou **centimètres (cm)** ou **pixels (px)** ou **pourcentage (%)**
- Exemple : `H2{ height : 150px }`

7. Liste des propriétés CSS

□ **Arrière plans :**

□ **background-color :**

- Définit la couleur de l'arrière plan en **hexadécimale** ou couleur (**red, blue, ...**) ou **transparent**
- Exemple : `H3{background-color : green}`

□ **background-image :**

- Définit l'image d'arrière plan en donnant l'**URL** de l'image
- Exemple : `BODY{background-image :image.gif}`

□ **background-attachment :**

- Spécifie si l'image d'arrière plan reste fixe ou suit les déplacements de l'écran **scroll** ou **fixed**.
- Exemple : `BODY{background-image :image.gif ;
background-attachment : fixed}`

7. Liste des propriétés CSS

92

▣ background-position :

- Spécifie la position de l'image d'arrière plan par rapport au coin supérieure gauche de la fenêtre en **points** (pt) ou **inches** (in) ou **centimètres** (cm) ou **pixels** (px) ou **pourcentage** (%) ou {**top** ou **center** ou **bottom**, **left** ou **center** ou **right**}.
- Exemple :

```
BODY{background-image :image.gif ;  
background-position :left, top}
```

▣ **Marges** :

▣ margin-top :

- Détermine la valeur de la marge supérieure en **pixels** (px) ou **automatique** (auto).
- Exemple :

```
P {margin-top :10px}
```

7. Liste des propriétés CSS

▣ margin-right :

- Détermine la valeur de la marge droite en **pixels (px)** ou **automatique (auto)**.
- Exemple : P {margin-right :10px}

▣ margin-bottom :

- Détermine la valeur de la marge inférieure en **pixels (px)** ou **automatique (auto)**.
- Exemple : P {margin-bottom :10px}

▣ margin-left :

- Détermine la valeur de la marge gauche en **pixels (px)** ou **automatique (auto)**.
- Exemple : P {margin-left :10px}

7. Liste des propriétés CSS

94

□ **Bords :**

□ **border-top-width :**

- Définit l'épaisseur du bord supérieur **thin** ou **medium** ou **thick** ou **points (pt)** ou **inches (in)** ou **centimètres (cm)** ou **pixels (px)** ou **pourcentage (%)**.
- Exemple : `H2 {border-top-width :thick}`

□ **border-right-width :**

- Définit l'épaisseur du bord droit **thin** ou **medium** ou **thick** ou **points (pt)** ou **inches (in)** ou **centimètres (cm)** ou **pixels (px)** ou **pourcentage (%)**.
- Exemple : `H2 {border-right-width :thick}`

7. Liste des propriétés CSS

95

□ border-bottom-width :

- Définit l'épaisseur du bord inférieur **thin** ou **medium** ou **thick** ou **points (pt)** ou **inches (in)** ou **centimètres (cm)** ou **pixels (px)** ou **pourcentage (%)**.
- Exemple : `H2 {border-bottom-width :thick}`

□ border-left-width :

- Définit l'épaisseur du bord gauche **thin** ou **medium** ou **thick** ou **points (pt)** ou **inches (in)** ou **centimètres (cm)** ou **pixels (px)** ou **pourcentage (%)**.
- Exemple : `H2 {border-left-width :thick}`

7. Liste des propriétés CSS

96

□ border-color :

- Définit la couleur de la bordure.
- Exemple : `H2 {border-color :blue}`

□ border-style :

- Définit le style de trait de la bordure *none* ou *solid* ou *dotted* ou *dashed* ou *double* ou *groove* ou *ridge* ou *inset* ou *outset*.
- Exemple : `H2 {border-style :solid}`

7. Liste des propriétés CSS

97

□ Remplissage :

□ padding-top :

- Définit la valeur de remplissage haut entre l'élément et le bord en **point (pt)** ou **inches (in)** ou **centimètres (cm)** ou **pixels (px)** ou **pourcentage (%)**.
- Exemple : `H2 {padding-top :5px}`

□ padding-right :

- Définit la valeur de remplissage droit entre l'élément et le bord en **point (pt)** ou **inches (in)** ou **centimètres (cm)** ou **pixels (px)** ou **pourcentage (%)**.
- Exemple : `H2 {padding-right :5px}`

7. Liste des propriétés CSS

□ padding-bottom :

- Définit la valeur de remplissage bas entre l'élément et le bord en **point (pt)** ou **inches (in)** ou **centimètres (cm)** ou **pixels (px)** ou **pourcentage (%)**.
- Exemple : `H2 {padding-bottom :5px}`

□ padding-left :

- Définit la valeur de remplissage gauche entre l'élément et le bord en **point (pt)** ou **inches (in)** ou **centimètres (cm)** ou **pixels (px)** ou **pourcentage (%)**.
- Exemple : `H2 {padding-left :5px}`

7. Liste des propriétés CSS

□ **Listes :**

□ List-style-type :

- Définit le type de puces ou de numérotation **disc** ou **circle** ou **square**.
- Exemple : `UL {List-style-type :disc}`

□ List-style-image :

- Permet de remplacer les **puces** par une **image**.
- Exemple :

```
UL {
    List-style-image : url('image.gif');
}
```

Une bibliothèque pour JavaScript

JQuery – Write less Do more

101

1. [Introduction](#)
2. [Syntaxe JQuery](#)
3. [Sélecteurs JQuery](#)
4. [Actions JQuery](#)
5. [Evénements JQuery](#)
6. [Animations JQuery](#)

1. Introduction – Qu'est ce que JQuery?

102

- JQuery est une bibliothèque de fonctions en JavaScript.
- C'est une librairie légère : "écrire moins, faire plus" .
- Cette bibliothèque contient les fonctionnalités suivantes :
 - ▣ Sélection des éléments HTML
 - ▣ Manipulation des éléments HTML
 - ▣ Manipulation du CSS
 - ▣ Fonctions d'événements HTML
 - ▣ Effets JavaScript et animations
 - ▣ Parcours et modification du DOM (Model Object Document)
 - ▣ Ajax (Asynchronous JavaScript and XML)

1. Introduction – Ajouter JQuery à vos pages

103

- La bibliothèque JQuery est stockée dans un fichier unique en JavaScript, contenant toutes les fonctions jQuery.

```
<head>
```

```
  <script type="text/javascript" src="jquery.js"></script>
```

```
</head>
```

1. Introduction – Télécharger JQuery

104

- Deux versions de JQuery sont disponibles : une version minimale et une autre non compressée (pour débbuger ou lire).
- Les deux versions sont disponibles à l'adresse : **jquery.com** Si l'on ne veut pas stocker localement la librairie, elle est disponible en ligne:

```
<head>  
  <script type="text/javascript"  
    src="http://ajax.googleapis.com/ajax/libs/jquery/1.4.2/jquery.min.js">  
  </script>  
</head>
```

```
<head>  
  <script type="text/javascript"  
    src="http://ajax.microsoft.com/ajax/jquery/jquery-1.4.2.min.js">  
  </script>  
</head>
```


1. Introduction – Exemple de base en JQuery

105

```
<html>
  <head>
    <script type="text/javascript" src="jquery.js"></script>
    <script type="text/javascript">
      $(document).ready(function(){
        $("button").click(function(){
          $("p").hide();
        });
      });
    </script>
  </head>
  <body>
    <h2>Ceci est un titre</h2>
    <p>Ceci est un paragraphe.</p>
    <p>Un autre paragraphe.</p>
    <button>Cliquer ici</button>
  </body>
</html>
```

2. Syntaxe JQuery

106

- La syntaxe de JQuery est faite sur mesure pour la sélection des éléments HTML et effectuer certaines actions.

- La syntaxe de base est la suivante :

\$ (sélecteur). action ()

- un signe \$ pour définir JQuery.
 - un (sélecteur) afin de trouver les éléments HTML.
 - une action() JQuery à effectuer.
- JQuery utilise une combinaison de XPATH et CSS pour la sélection des éléments.

2. Syntaxe JQuery - Appels

107

- Quelques appels JQuery:
 - ▣ `$(this).hide()` : pour cacher l'élément HTML courant.
 - ▣ `$("#test").hide()` : pour cacher l'élément HTML dont l'ID est *test*.
 - ▣ `$("p").hide()` : pour cacher tous les éléments HTML de type p.
 - ▣ `$(".test").hide()` : pour cacher tous les éléments possédant la classe "test".

2. Syntaxe JQuery – Fonction Ready

108

- Toutes les fonctions JQuery se situent dans la fonction *ready* de l'objet document :

- Syntaxe :

- `$(document).ready(function () { code });`

- ou en raccourci :

- `$(function () { code });`

- Il s'agit d'éviter d'exécuter du code JQuery avant le chargement complet de la page (erreurs pour cacher des éléments ou récupérer la taille d'une image avant son chargement).

2. Syntaxe JQuery – Exemples

109

□ Exemple 1 :

□ CARACTÉRISTIQUES

■ DOUBLE SCRIPT

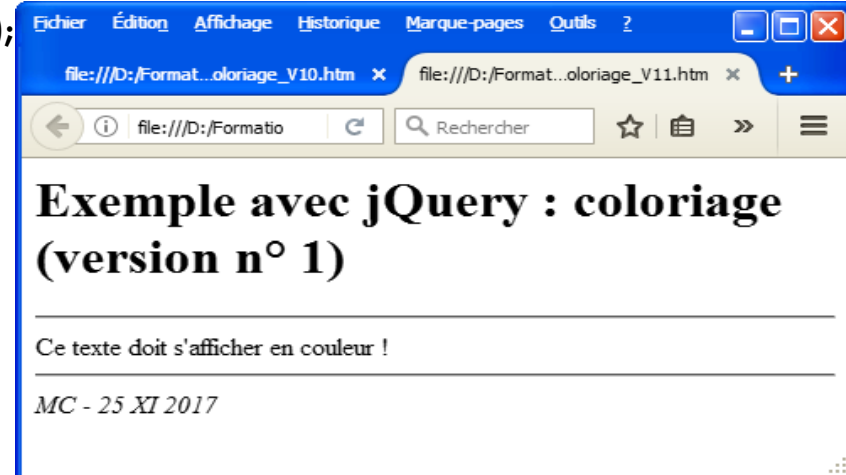
- Bibliothèque JQuery + script particulier à la page : application d'un style de couleur à la division identifiée par contenu.

■ IDENTIFICATION D'UNE SÉLECTION D'ÉLÉMENT(S) DE LA PAGE

- notation : `jQuery(sélection)` ou `$(sélection)` avec *sélection 1* exprimée a priori comme en CSS, dans une chaîne de caractères.
- exemple : `jQuery("div#contenu")` ou `$("#div#contenu")`

■ MANIPULATION DE LA SÉLECTION PAR APPLICATION D'UNE MÉTHODE

- définition d'un style via `.css(propriété, valeur)`
- exemple : `$("#div#contenu").css("color", "red");`



2. Syntaxe JQuery – Exemples

110

▣ Code en HTML du contenu de la page

```
<head>
  <title>Exemple avec jQuery : coloriage (version n° 1)</title>
  <script type="text/javascript" src="jquery.js"></script>
  <script type="text/javascript" src="jq-colorier-v1.js"></script>
</head>
<body>
  <h1>Exemple avec jQuery : coloriage (version n° 1)</h1>
  <hr />
  <div id="contenu">Ce texte doit s'afficher en couleur !</div>
  <hr />
  <address>MC - 25 XI 2017</address>
</body>
```

▣ Code en JavaScript du script : **jq-colorier-v1.js**

```
$("div#contenu").css("color", "red");
```

2. Syntaxe JQuery – Exemples

111

□ Exemple2 :

□ Amélioration de l'exemple 1

- Existence d'un cas de mauvaise exécution
 - Script inopérant si chargement avant le contenu de la page
- Garantir la bonne exécution dans tous les cas
 - Exécution du code seulement après le chargement de la page
- Mécanisme particulier dans JQuery
 - indication d'une fonction anonyme ,exécutée après le chargement :

`$(document).ready(function () { code });`

ou en raccourci : `$(function () { code });`

2. Syntaxe JQuery – Exemples

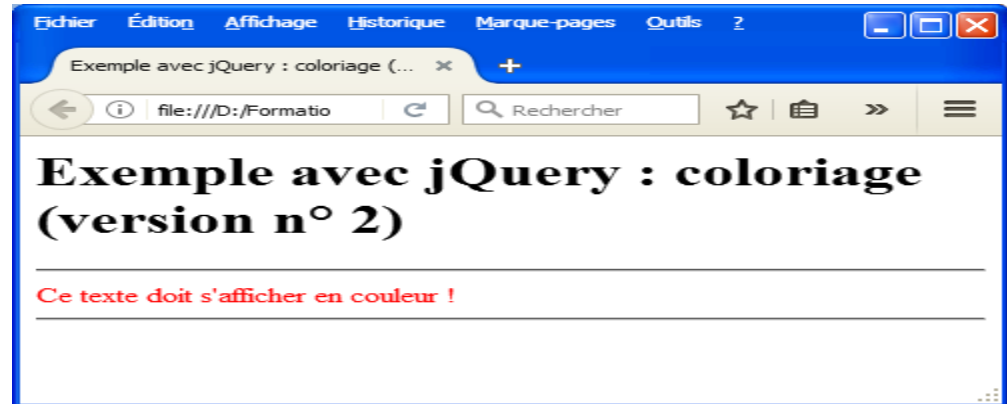
112

- Code en HTML du contenu de l'entête de la page

```
<head>  
  <title>Exemple avec jQuery : coloriage (version n° 2)</title>  
  <script type="text/javascript" src="jquery.js"></script>  
  <script type="text/javascript" src="jq-colorier-v2.js"></script>  
</head>
```

- Code en JavaScript du script : *jq-colorier-v2.js*

```
$( function () {  
    $("div#contenu").css("color", "red");  
});
```



3. Sélecteurs JQuery

113

- Sélecteurs d'éléments :
 - ▣ JQuery utilise la syntaxe CSS pour sélectionner des éléments :
 - \$("p") sélectionne tous les éléments **p**
 - \$("p.intro") sélectionne les éléments p possédant la classe **intro**
 - \$("p#demo") sélectionne l'unique élément p ayant pour ID **demo**

3. Sélecteurs JQuery

□ Sélecteurs d'attributs :

▣ JQuery utilise la syntaxe XPATH pour sélectionner des attributs :

- `$("[href]")` sélectionne tous les éléments avec un attribut **href**
- `$("[href='#']")` sélectionne tous les éléments possédant un attribut **href** ayant pour valeur **'#'**
- `$("[href!='#']")` sélectionne tous les éléments possédant un attribut **href** différent de **'#'**
- `$("[href$='.jpg']")` sélectionne tous les éléments possédant un attribut **href** dont la valeur se termine par **'jpg'**

3. Sélecteurs JQuery – Récapitulatif des sélecteurs

115

Sélecteur	Exemple	Description
*	<code>\$("*")</code>	tous les éléments
#id	<code>\$("#lastname")</code>	l'élément avec l'id lastname
.class	<code>\$(".intro")</code>	tous les éléments avec la classe "intro"
element	<code>\$("p")</code>	tous les éléments p
:first	<code>\$("p:first")</code>	le premier élément p
:last	<code>\$("p:last")</code>	le dernier élément p
:even	<code>\$("tr:even")</code>	les lignes tr paires
:odd	<code>\$("tr:odd")</code>	les lignes tr impaires
:eq(index)	<code>\$("ul li:eq(3)")</code>	le 4ème élément d'une liste (l'index commence à 0)
:gt(no)	<code>\$("ul li:gt(3)")</code>	les éléments d'une liste dont l'index est supérieur (strict) à 3
:lt(no)	<code>\$("ul li:lt(3)")</code>	les éléments d'une liste dont l'index est inférieur (strict) à 3
:header	<code>\$(":header")</code>	tous les éléments h1, h2 ...

3. Sélecteurs JQuery – Récapitulatif des sélecteurs

116

Sélecteur	Exemple	Description
[attribute]	<code>\$("[href]")</code>	tous les éléments ayant un attribut href
[attribute=value]	<code>\$("[href='default.htm']")</code>	tous les éléments possédant un attribut href dont la valeur est "default.htm"
[attribute\$=value]	<code>\$("[href\$='.jpg']")</code>	tous les éléments possédant un attribut href dont la valeur se termine par ".jpg"
:input	<code>\$(":input")</code>	tous les éléments de type input (formulaires)
:text	<code>\$(":text")</code>	tous les éléments input de type 'text'
:password	<code>\$(":password")</code>	tous les éléments input de type 'password'
:radio	<code>\$(":radio")</code>	tous les éléments input de type 'radio'
:checkbox	<code>\$(":checkbox")</code>	tous les éléments input de type 'checkbox'
:submit	<code>\$(":submit")</code>	tous les éléments input de type 'submit'
:reset	<code>\$(":reset")</code>	tous les éléments input de type 'reset'

3. Sélecteurs JQuery – Récapitulatif des sélecteurs

117

Sélecteur	Exemple	Description
:button	<code>\$(":button")</code>	tous les éléments input de type 'button'
:image	<code>\$(":image")</code>	tous les éléments input de type 'image'
:file	<code>\$(":file")</code>	tous les éléments input de type 'fichier'
:enabled	<code>\$(":enabled")</code>	tous les éléments input activés
:disabled	<code>\$(":disabled")</code>	tous les éléments input désactivés
:selected	<code>\$(":selected")</code>	tous les éléments input sélectionnés
:checked	<code>\$(":checked")</code>	tous les éléments input cochés

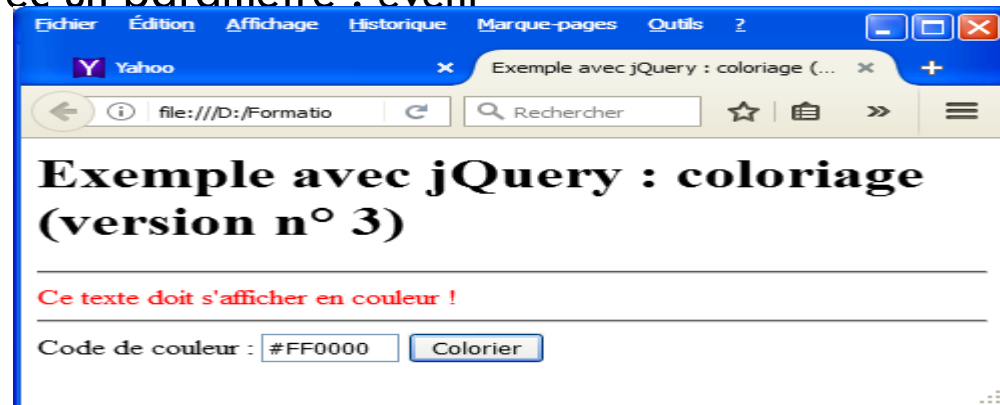
3. Sélecteurs JQuery - Exemple

118

□ Exemple3 : Coloriage avec une couleur saisie dans un formulaire

□ Caractéristiques :

- IDENTIFICATION DE LA ZONE DE SAISIE ET DU BOUTON D'ACTIVATION
 - attribut id dans les balises input en XHTML récupération de la valeur en jQuery : `$(sélection).val()`
- INDICATION DU TRAITEMENT ASSOCIÉ AU CLIC SUR LE BOUTON
 - méthode en JQuery portant le nom de l'événement : `.click(fonction)` fonction de traitement de l'événement avec un paramètre : event



3. Sélecteurs JQuery - Exemple

119

▣ Code en HTML du formulaire

```
<form action="">
```

```
Code de couleur : <input type="text" name="zcouleur"  
id="zcouleur" value="#FF0000" size="7" />
```

```
<input type="button" value="Colorier" id="bouton" />
```

```
</form>
```

▣ Code en JavaScript du script : *jq-colorier-v3.js*

```
$( function(){
```

```
  $('#input#bouton').click( function () {
```

```
    // application, apres le clic sur le bouton, de la couleur
```

```
    // definie dans la zone identifiee par #zcouleur,
```

```
    // au texte de la division identifiee par #contenu
```

```
    $("div#contenu").css("color", $("input#zcouleur").val() );
```

```
  } );
```

```
};
```

4. Actions JQuery

120

- Types d'actions :
 - Modifier le contenu HTML
 - Modifier valeur d'objet
 - Manipulation du CSS
 - Ajouter / Supprimer des classes
 - Manipulation de la forme des éléments
 - Insérer / Remplacer / Supprimer des éléments
 - Insérer / Remplacer / Supprimer des attributs

4. Actions JQuery – Récapitulatif des actions

121

Action	Exemple	Description
.html	<code>\$("#p").html("<div>Bonjour</div>");</code>	modifier le contenu de tous les p du document
.val()	<code>\$("#input:checkbox:checked").val();</code>	obtenir la valeur de la première checkbox cochée
.val(valeur)	<code>\$(":text[name='txt']").val("Hello");</code>	modifier la valeur d'un champ text de nom txt
.addClass	<code>\$("#p").addClass("red");</code>	Ajouter la classe red aux éléments p
.removeClass	<code>\$("#p").removeClass("blue");</code>	Supprimer la classe blue aux éléments p
.toggleClass	<code>\$("#div").toggleClass("main");</code>	Ajouter la classe main au div si absent ou l'inverse

4. Actions JQuery – Récapitulatif des actions

122

Action	Exemple	Description
.css	<code>\$("#div").css("color", "blue");</code>	Modifie la couleur du texte de div en bleu
.append	<code>\$("#ul").append("Test");</code>	sélection des ul et ajout à la fin
.replaceWith	<code>\$("#h1").replaceWith("<div>Test</div>");</code>	remplace tous les <code><h1></code> par des <code><div>Test</div></code>
.remove	<code>\$("#span.names").remove();</code>	supprimer tous les span de classe names
.attr	<code>\$("#a").attr("href", "home.htm");</code>	ajoute l'attribut href=home.htm à l'élément a
.removeAttr	<code>\$("#button").removeAttr("disabled")</code>	supprime l'attribut disabled de tous les boutons

5. Événements JQuery

123

- ❑ Les événements JQuery sont liés aux actions de l'utilisateur sur la page au moment de la navigation.
- ❑ Événements de la souris :

Événement	Exemple	Description
click()	<code>\$('#bouton').click()</code>	Au clic de la souris sur bouton
dblclick()	<code>\$('#bouton').dblclick()</code>	Au double clic de la souris sur bouton
mouseenter() Ou mouseover()	<code>\$('#texte').mouseenter()</code>	A l'entrée de la souris au champs texte
mouseleave() Ou mouseout()	<code>\$('#texte').mouseleave()</code>	Lorsque la souris quitte le champs texte
mousedown()	<code>\$('#texte').mousedown()</code>	Lorsqu'on clic et on maintien le clic sur texte
mouseup()	<code>\$('#texte').mouseup()</code>	Lorsqu'on relâche le clic de la souris sur texte
mousemove	<code>\$('#texte').mousemove()</code>	lorsqu'on déplace la souris sur le champ texte
scroll()	<code>\$(window).scroll()</code>	Utilisation de la roulette dans le document

5. Événements JQuery

124

□ Événements du clavier :

Événement	Exemple	Description
keydown()	<code>\$('#input#texte').keydown()</code>	Appui sur une touche du clavier
keyup()	<code>\$('#input#texte').keyup()</code>	<i>Relâchement d'une touche du clavier</i>
keypress()	<code>\$('#input#texte').keypress()</code>	<i>Maintien d'une touche du clavier enfoncée</i>

□ Événements d'Eléments :

Événement	Exemple	Description
focus()	<code>\$('#input#texte').focus()</code>	Réception de focus par l'élément texte
blur()	<code>\$('#input#texte').blur()</code>	Perte de focus par l'élément texte
Change()	<code>\$('#input#texte').change()</code>	<i>Modification d'un élément texte</i>
load()	<code>\$('#body').load()</code>	Chargement de la page

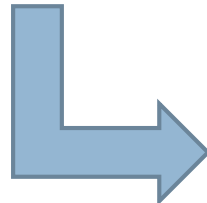
5. Événements JQuery - Exemple

125

- Exemple4 : Au double clic sur le bouton, on change le texte du titre H1



Après l'événement



5. Événements JQuery - Exemple

126

▣ Code en HTML du corps de la page Web

```
<body>
  <h1>Mon Titre</h1>
  <button>Double clic sur moi</button>
</body>
```

▣ Code en JavaScript du script : *jq-Even-jq.js*

```
$(function() {
  $("button").dblclick( function(event) {
    $("h1").text("Nouveau titre");
  }); // au double clic sur le bouton, je change mon titre H1.
});
```

6. Animations JQuery

127

- JQuery permet d'ajouter de l'animation à la page Web de type :
 - Afficher / Masquer l'élément,
 - Apparition / Disparition progressive avec un effet d'opacité,
 - Apparition / Disparition progressive avec un effet de translation,
 - Animation personnalisée
 - Etc...
- NOTATION D'UNE VITESSE : '*slow*' ou '*normal*' ou '*fast*' ou **nombre** (le nombre correspond à la durée totale exprimée en millisecondes).

6. Animations JQuery – Apparition et Disparition

128

- Afficher un élément :

```
$("#div").show();
```

- Afficher un élément lentement (slow=600ms) :

```
$("#div").show("slow");
```

- Cacher un élément rapidement (fast=200ms) :

```
$("#div").hide("fast");
```

- Inverser (afficher ou cacher) en une durée fixée :

```
$("#div").toggle(100);
```


6. Animations JQuery – Translation et Opacité

129

□ Translation :

```
$("#div").slideUp();
```

```
$("#div").slideDown("fast");
```

```
$("#div").slideToggle(1000);
```

□ Opacité :

```
$("#div").fadeIn("fast");
```

```
$("#div").fadeOut("normal");
```

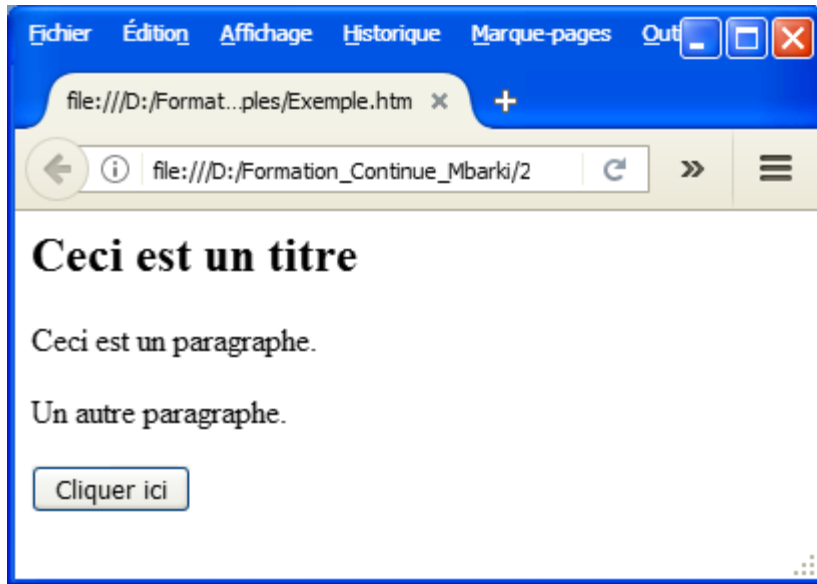
□ opacité fixée :

```
$("#div").fadeTo("fast", 0.5);
```

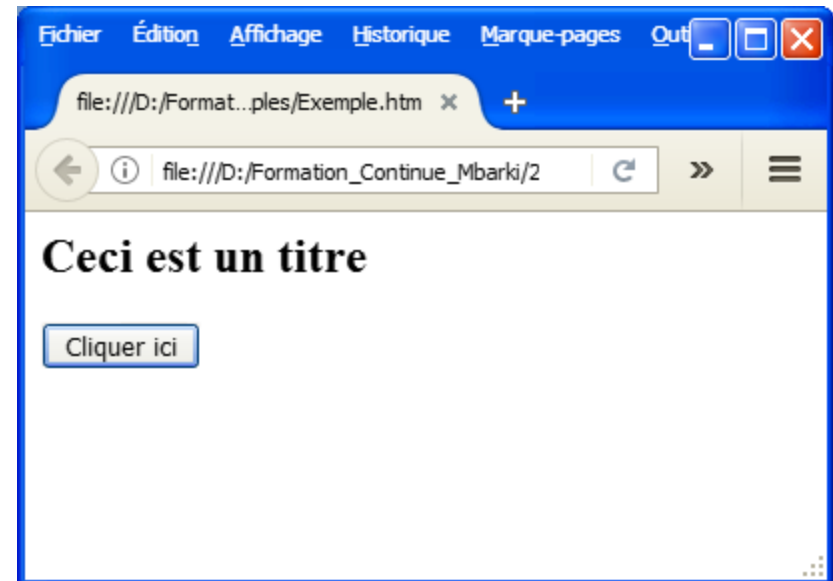
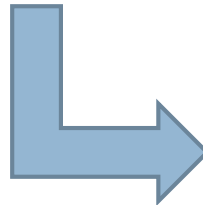
6. Animations JQuery - Exemple

130

- Exemple 5 : Au clic sur le bouton, le texte du paragraphe disparaît et apparaît avec une vitesse lente.



Après l'événement



6. Animations JQuery - Exemple

131

▣ Code en HTML du corps de la page Web

```
<body>
  <h2>Ceci est un titre</h2>
  <p>Ceci est un paragraphe.</p>
  <p>Un autre paragraphe.</p>
  <button>Cliquer ici</button>
</body>
```

▣ Code en JavaScript du script : *jq-Anim-jq.js*

```
$(document).ready(function() {
  $("button").click(function() {
    $("p").toggle('slow');
  });
});
```

6. Animations JQuery – Animation personnalisée

132

- Modification des propriétés CSS des éléments en fonction du temps:

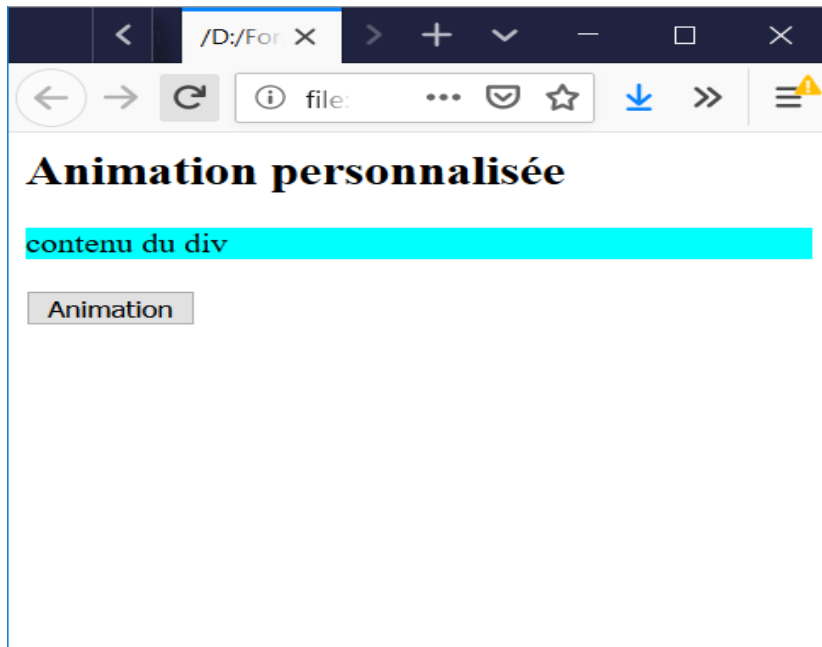
.animate (options, durée)

- ▣ Options : ensemble de propriétés CSS :
 - { 'propriété1 : valeur1', ..., 'propriété n : valeur n' }
- ▣ Durée : Temps d'exécution de l'animation.

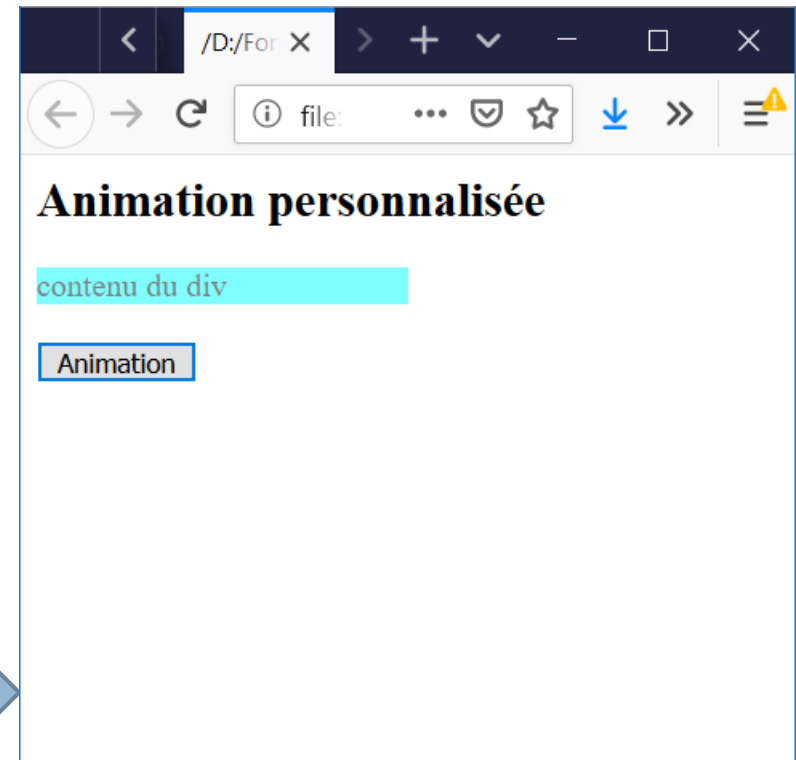
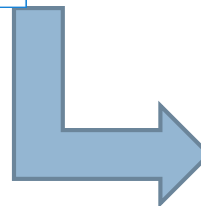
6. Animations JQuery - Exemple

133

- Exemple 6 : Au clic sur le bouton, Réduction de la largeur à 50% d'une section div et changement d'opacité. Le tout en 1s.



Après l'événement



6. Animations JQuery - Exemple

134

▣ Code en HTML du corps de la page Web

```
<body>
  <h2>Animation personnalisée</h2>
  <div style="background-color : aqua">contenu du div</div><br>
  <button> Animation </button>
</body>
```

▣ Code en JavaScript du script : ***jq-Anim-jq2.js***

```
$(document).ready(function() {
  $("button").click(function() {
    $("div") .animate( {width: "50%", opacity: 0.5}, 1000);
  });
});
```

6. Animations JQuery – Enchaînement d'animations

135

- Plusieurs appels à `.animate()`:

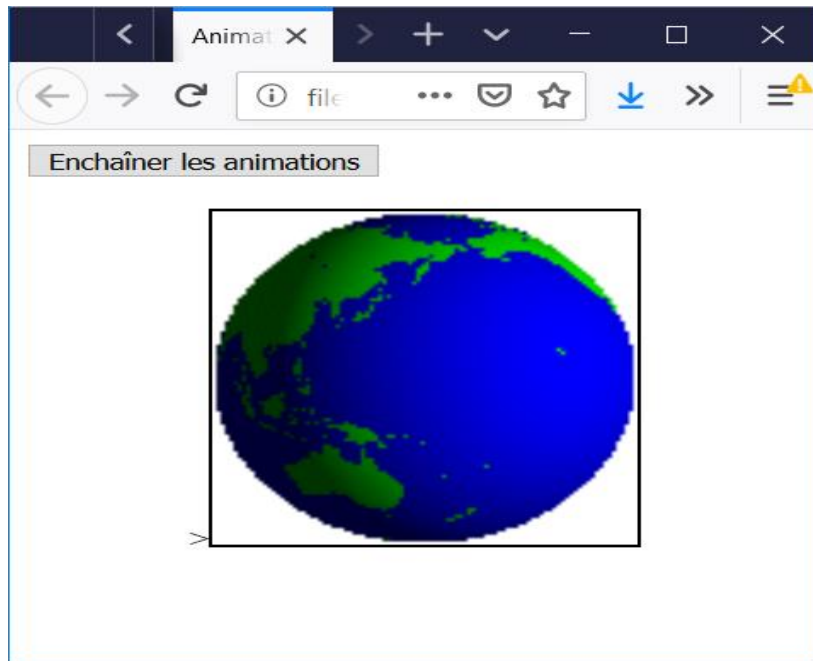
`$(sélecteur).animate (...).animate(...)...;`

- ▣ Par défaut les animations sont effectuées l'une à la suite de l'autre.

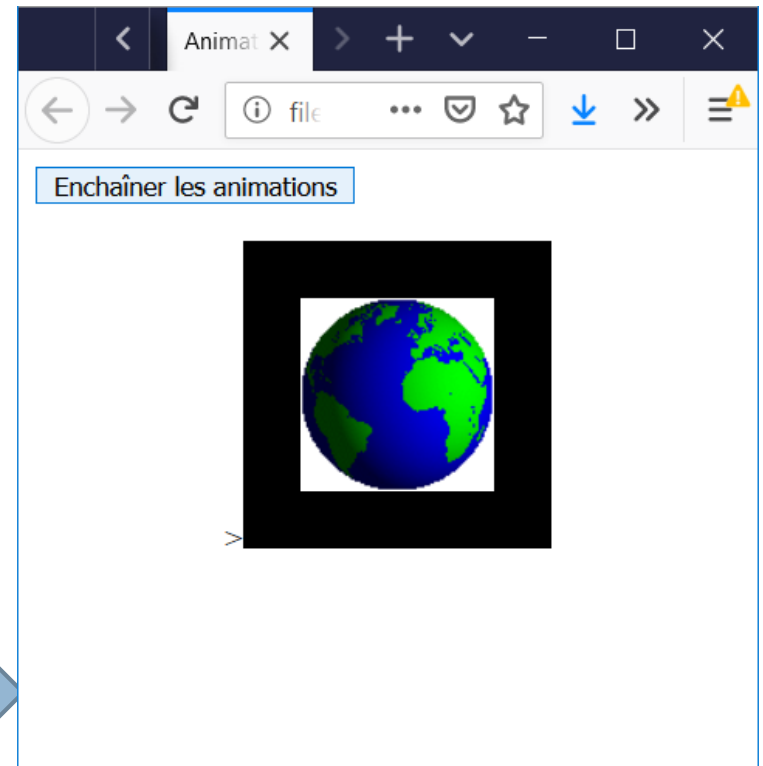
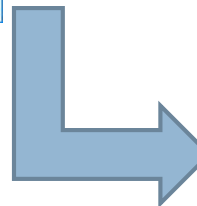
6. Animations JQuery - Exemple

136

- Exemple 7 : Au clic sur le bouton, la largeur de la bordure est augmenté à 30px, et la largeur de l'image est diminuée de 100px. Le tout en 1s.



Après l'événement



6. Animations JQuery - Exemple

137

▣ Code en HTML du corps de la page Web

```
<body>
  <button id="enchainer">Enchaîner les animations</button><br/>
  <br />
  <center>
    
  </center>
</body>
```

▣ Code en JavaScript du script : **jq-Anim-jq3.js**

```
$(function() {
  $('#enchainer').click( function() {
    $('img').animate({ 'border-width': '+=30'}, 1500 )
    .animate({ 'width': '-=100'}, 1500);
  });
});
```

6. Animations JQuery – Enchaînement d'animations

138

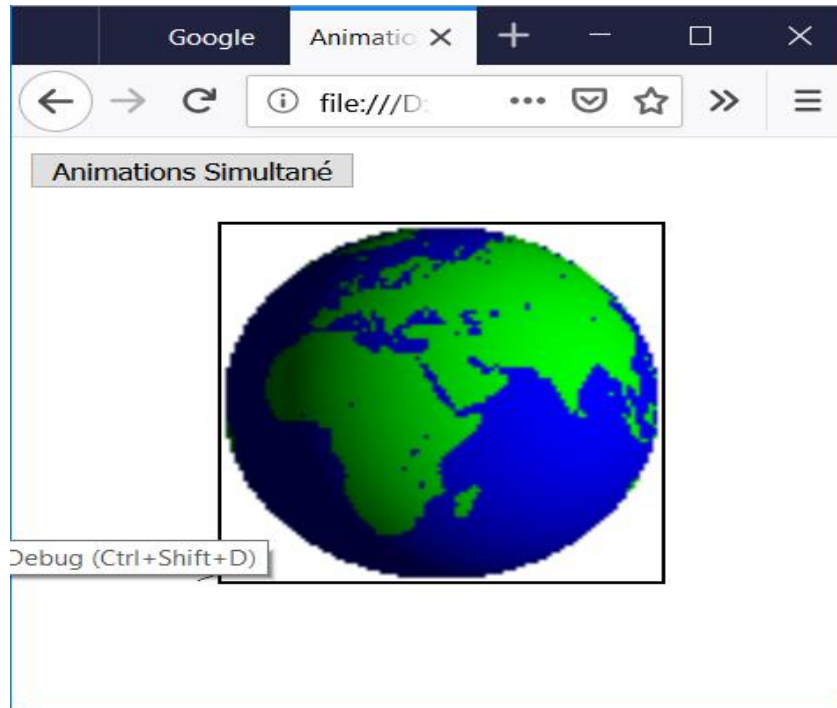
- ▣ Les animations enchainées peuvent être exécutées au même temps en utilisant : *"queue:false"*

```
$(sélecteur).animate({propriété : valeur, ...},  
                    {queue : false, durée})  
    .animate(...)...;
```

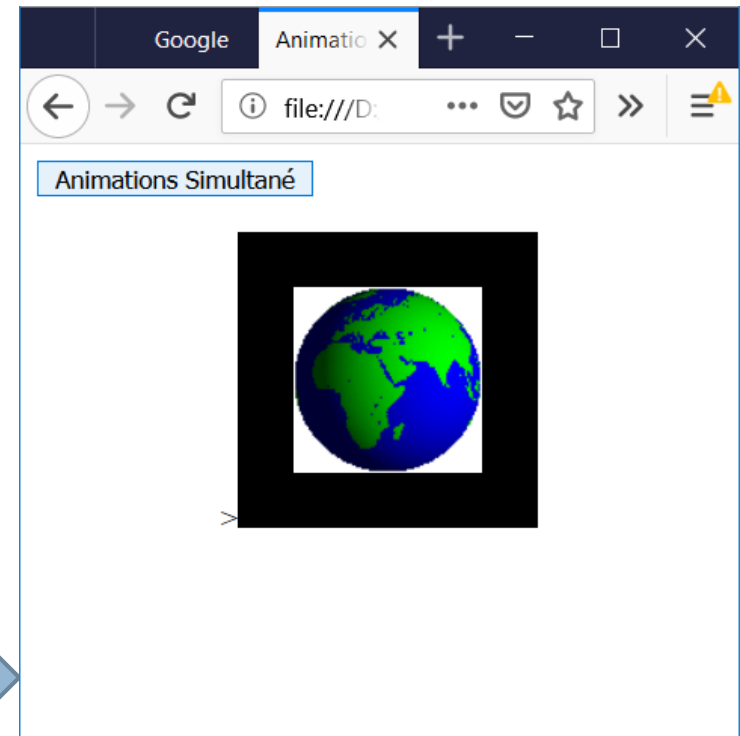
6. Animations JQuery - Exemple

139

- **Exemple 8** : Au clic sur le bouton, la largeur de la bordure est augmenté à 30px, et la largeur de l'image est diminuée de 100px simultanément. Le tout en 1s.



Après l'événement



6. Animations JQuery - Exemple

140

▣ Code en HTML du corps de la page Web

```
<body>
  <button id="enchainer">Enchaîner les animations</button><br/>
  <br />
  <center>
    
  </center>
</body>
```

▣ Code en JavaScript du script : **jq-Anim-jq4.js**

```
$(function() {
  $('#Simultané').click( function() {
    $('img').animate({ 'border-width': '+=30' },{queue:false, 1500} )
    .animate({ 'width': '-=100'}, 1500);
  });
});
```