



# Conception et création de pages web

Gérald Collaud

Centre NTE, Université de Fribourg



*Ecole Normale, 23 août 2000*





# Programme

- Quelques notions importantes
- Comment créer des pages web
- Aller plus loin
  
- Ateliers / exercices





# Objectifs

- Comprendre les mécanismes de base
- Etre à même de créer des pages web simples
- Concevoir et créer un mini site web
  
- Connaître ce qu'il est possible de faire en plus
- Avoir quelques points de départ

# Plan

- Partie 1 : Quelques notions de base
  - ◆ WWW
  - ◆ Types de fichiers habituels
  - ◆ Capacité du réseau
  - ◆ Composantes
- Partie 2 : Création de pages web
  - ◆ Outils
    - Texte
    - Image
    - Multimédia
  - ◆ HTML I

# Plan (suite)

- Partie 3 : Création d'un site web
  - ◆ Caractéristiques d'un bon site web
  - ◆ Votre travail en tant qu'auteur
  - ◆ Quelques conseils
- Partie 4 : Extensions
  - ◆ HTML II
    - Tables
    - *Image map*
    - Formulaires
  - ◆ HTML étendu
    - Bases des feuilles de style
    - Rudiments de Javascript



# Partie 1 : Notions de base



*Ecole Normale, 23 août 2000*



# World Wide Web (WWW)

- Comment ça fonctionne
- Outil fédérateur :
  - ◆ Type de données
  - ◆ Plateformes
  - ◆ Accès (URL)



# Types de fichiers habituels

- La plupart des sites web simples ont deux types de fichier :
  - ◆ HTML, pour les documents HTML
  - ◆ GIF (Graphics Interchange Format), pour les images, logos, diagrammes présents dans les pages HTML
- Il existe de nombreux autres formats :
  - ◆ JPEG, pour les photographies > 256 couleurs
  - ◆ D'innombrables formats spéciaux



# Capacité du réseau

- 1 KB = 1 kilobyte = 1024 bytes
- 1 image "standard" a une taille de 20kB
- Les modems actuels ont une capacité de 28.8 à 56 kilobits/sec
  - ◆ bits, pas des bytes!
  - ◆ Réfléchir avant de proposer de gros fichiers

Scripts CGI

Plugins

VRML

Java

ASP

PHP

SMIL

Video

XML

Audio

Feuilles de style

Javascript

HTML Etendu

Image Maps

Pages HTML statiques simples

# Avant de compliquer...

- Certaines extensions peuvent sembler naturelles mais :
  - ◆ Nécessite du temps et des compétences
  - ◆ Demande généralement beaucoup de bande passante
  - ◆ Demande des serveurs performants
  - ◆ Les utilisateurs peuvent parfois avoir besoin de compléments

# Video sur le web

- Nécessite beaucoup d'espace disque et de bande passante
- Difficile à éditer
- La qualité laisse souvent à désirer :
  - ◆ 1/4 d'écran ou même moins
  - ◆ 5 images / seconde (vidéo ou TV : 25-30)



# Quelques références

- [WebReference.com](http://WebReference.com)
- [WebDeveloper.com](http://WebDeveloper.com)
- [Serverwatch.com](http://Serverwatch.com)
- [www.zdnet.com](http://www.zdnet.com)
- [www.cnet.com](http://www.cnet.com)
- ... et beaucoup d'autres!





# Partie 2 : Création de pages web



*Ecole Normale, 23 août 2000*



# Code HTML

- Editeur texte
- Editeur de code
- WYSIWYG
- Convertisseur
- Couplage avec une base de données



# Images

- Acquisition
  - ◆ Caméra digitale
  - ◆ Scanner
  - ◆ Labo photo
  - ◆ Internet
- Traitement
  - ◆ Mac : Adobe Photoshop, Canvas, MacDraw, Graphic Converter, ...
  - ◆ PC : Paint Shop Pro, Adobe Photoshop, Corel PhotoPaint, ...
- Formats
  - ◆ GIF
  - ◆ JPEG



# Taille des images

- 100 KB est trop grand (max. 30KB)
- Réduire la taille
- Réduire le nombre de couleurs
- Utiliser des *thumbnails* (qui pointent vers la vraie image)



# Vidéo : acquisition

- Sources:
  - ◆ Bande analogique (VHS, Hi-8, etc)
  - ◆ Bande digitale -- format DV
  - ◆ Internet
- Cartes de capture Vidéo
  - ◆ Exemple : Targa Truevision, Radius MotoDV
  - ◆ Autre option : format DV  
IEEE 1394 "Firewire" qui permet un transfert direct sans conversion
- Caméras reliées à l'ordinateur (par exp. : Connectix Quickcam)

# Vidéo : formats

- QuickTime
  - ◆ Développé par Apple
  - ◆ Disponible sur Mac et PC
- AVI
  - ◆ Développé par Microsoft (Audio/Video Interleaved format)
- MPEG 2
  - ◆ Standard international
- Real Media
  - ◆ Développé par RealVideo (<http://www.realvideo.com>)
  - ◆ Standard 'de facto' pour le *streaming*

# Vidéo : traitement

- Nécessite généralement un logiciel commercial
  - ◆ Adobe Première
  - ◆ IMovie
- Et un ordinateur performant
  - ◆ G3, Mac; 400MHz, PC
  - ◆ 128 MB RAM
  - ◆ Plusieurs GB de disque dur (1 min. = 200MB)

# HTML I

- Langage de marquage ⇒ Balises
  - ◆ de document
  - ◆ de mise en page
  - ◆ de texte
  - ◆ de navigation
  - ◆ d'images
- Notes :
  - ◆ HTML évolue
  - ◆ HTML ne peut pas tout représenter
  - ↳ Extensions



# Balises de document

<html>

<head>

<title>Exemple de balises

</title>

</head>

<body>

Ceci est un document minimal.

</body>

</html>

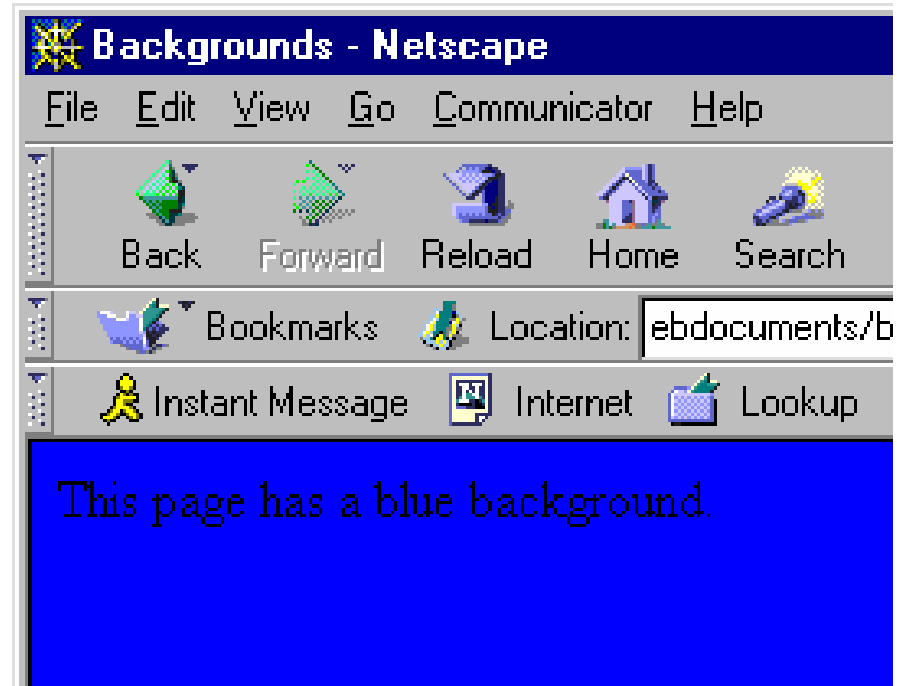
# Couleur de fond

- Attribut ajouté à la balise <body>
- Il y a 16 couleurs pré-définies que vous pouvez ajouter (red, blue yellow, etc.).
- Autrement, il faut connaître le code hexadécimal de la couleur



# Exemple

```
<html>
<head>
<title>Backgrounds</title>
</head>
<body bgcolor="#0000FF">
This page has a blue
background.
<p>
</body>
</html>
```



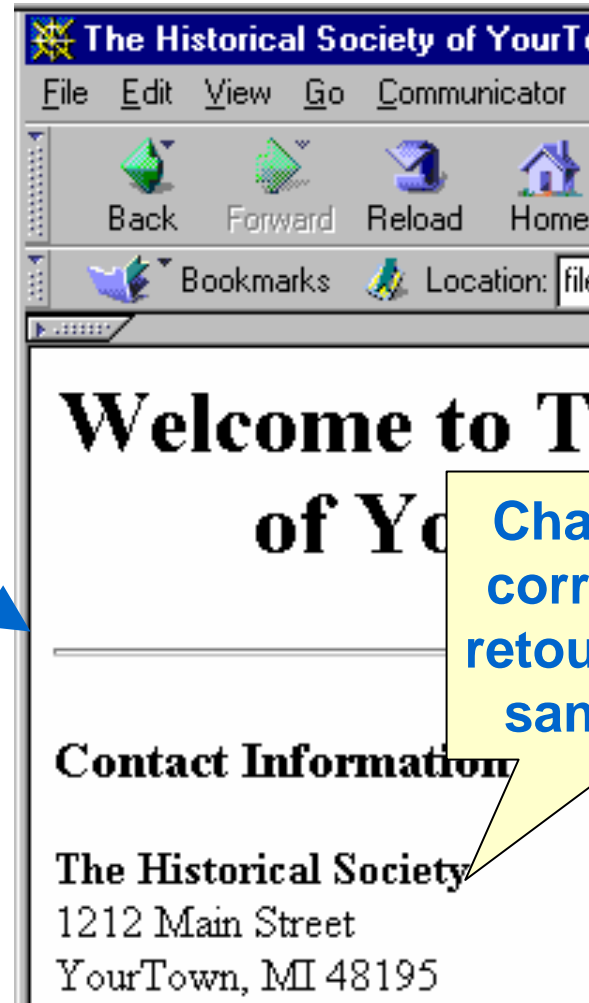


# Balise de mise en page

- Paragraphes et retour à la ligne
  - ◆ `<p></p>` entoure un paragraphe
    - Saute une ligne et démarre un nouveau paragraphe
  - ◆ `<br>` (sans fermeture)
    - Démarre le prochain mot sur la prochaine ligne
    - Pas d'espace entre les lignes
- Ligne horizontale
  - ◆ A utiliser pour séparer des portions de pages
  - ◆ `<hr>` (sans fermeture)
  - ◆ Peut inclure un attribut %
  - ◆ Exemple
    - `<hr>`
    - `<hr width="50%">`

# Exemple

<hr>  
<h3>Contact Information</h3>  
<b>The Historical Society</b>  
<br>  
1212 Main Street<br>  
YourTown, MI 48195<br>



Chaque <br> correspond à retour à la ligne sans espace

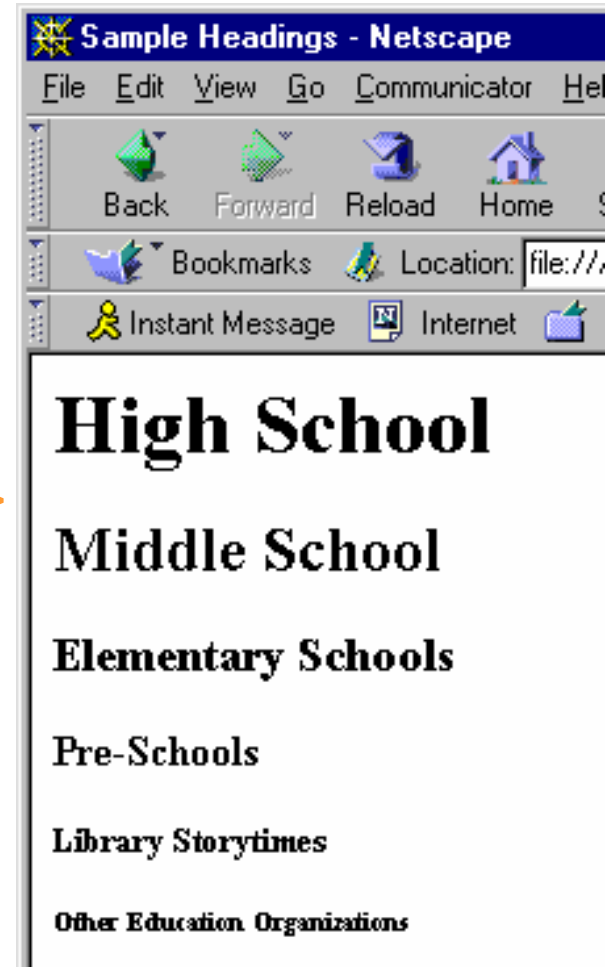
# Balises de texte : entêtes

- Dénote des titres et des sous-titres
- Du niveau 1 au niveau 6
- Montre l'importance relative de l'information



# Exemple

<h1>High School</h1>  
<h2>Middle School</h2>  
<h3>Elementary Schools</h3>  
<h4>Pre-Schools</h4>  
<h5>Library Storytimes</h5>  
<h6>Other Education Organizations</h6>



# Balises de texte : alignement

- Par défaut, le texte est aligné à gauche
- Pour aligner un paragraphe, il faut utiliser l'attribut align  
<p align=left> <p>
- Pour centrer des blocs de texte, utilisez :  
<center> </center>

# Balises de texte : couleur du texte

- Même principe que pour la couleur de fonds
- Attribut de la balise body aussi
- Par exemple :

```
<body bgcolor="#00FFFF" text="#800000">
```



# Balises de texte: gras et italique

- Pour mettre en évidence du texte
- Exemples :

`<b>Ceci est gras</b>`

**Ceci est gras**

`<i>Ceci est italic</i>`

*Ceci est italique*

# Balises de texte : listes non ordonnées

- Utile pour attirer l'attention sur des éléments qui n'ont pas d'ordre particulier

`<ul>`

`<li>Chiens`

- Chiens

`<li>Chats`

- Chats

`</ul>`





# Balises de texte : listes ordonnées

- Utile pour décrire des séquences ou des procédures par étape

<ol>

<li>Phase 1

1. Phase 1

<li>Phase 2

2. Phase 2

</ol>



# Balises de texte : listes de définition

- Affiche un élément suivi de sa définition

<dl>

<dt>Cirrus

Cirrus

<dd>high wispy

high wispy

<dt>Nimbus

Nimbus

<dd>dark

dark

</dl>



# Balise de navigation

- Liens relatifs

- ◆ Liens vers des documents sur le même serveur

- ↳ Permet de donner le nom du fichier en relation avec la page sur laquelle vous êtes

Par exemple, si index.html et page1.html sont dans le même dossier, le lien de la page index vers la page 1 sera :

```
<a href="page1.html">Aller à la page 1</a>
```

- Liens absolus

- ◆ Liens vers des documents situés sur un autre serveur

- ↳ Spécifier l'URL complet :

```
<a href="http://cnn.com/">Aller à CNN</a>
```



# Changer la couleur des liens

- Si vous changez la couleur de fonds, il peut être nécessaire de change la couleur des liens aussi.

```
<body bgcolor="#00FFFF" text="#000000" link="#FF0000"  
vlink="#000000">
```

- L'attribut **link** est la couleur avant que le lien soit suivi
- **Vlink** est la couleur d'un lien déjà visité

# Ajouter une image

```

```

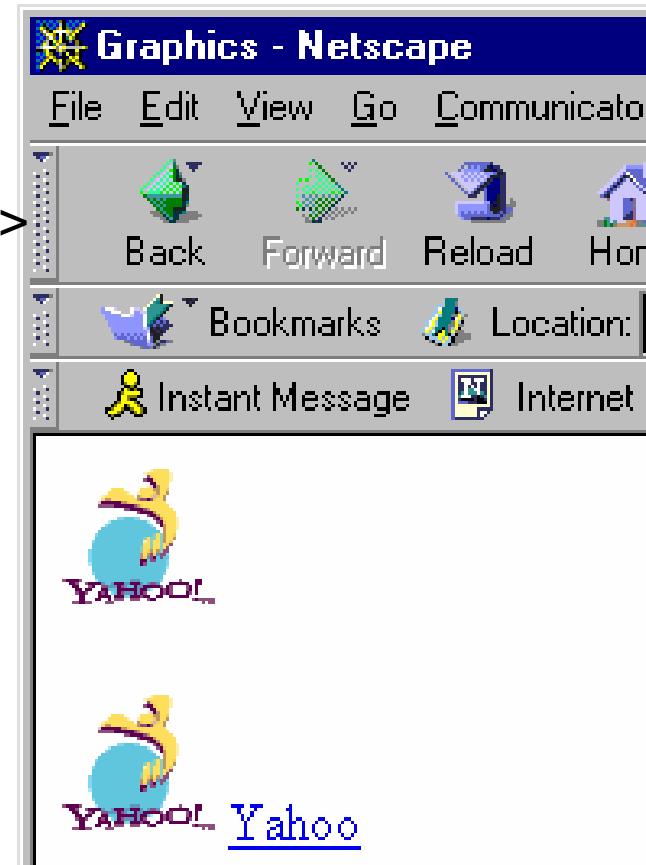
```
<p>
```

```

```

```
<a href="http://www.yahoo.com">Yahoo</a>
```

```
</p>
```



# Aligner des images

- Par défaut, le texte commence sur la ligne qui suit l'image
- L'attribut **align** peut modifier ce comportement
- Exemple :

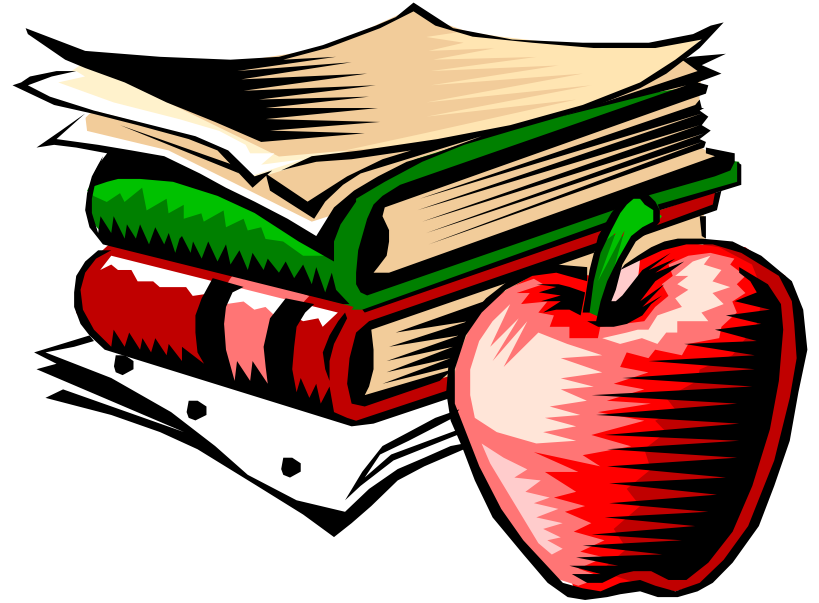
```

```



## 2 : Création de pages web : HTML

En utilisant l'attribut `align="right"`, un image peut se trouver sur la droite d'un texte.



# Liens et images

- Une image peut être un lien hypertexte

```
<a href="http://apple.com/"></A>
```

Cette ligne ajoute un cadre bleu autour d'une image de pomme.

- Pour ne pas afficher le cadre, utilisez l'attribut :

```
Border="0"
```







# Partie 3 : Site web



*Ecole Normale, 23 août 2000*



# Caractéristiques d'un bon site web

- Bien organisé
- Facile à naviguer
- Attrayant
- Utile
- A jour



# Votre travail

- Préparer le fichiers HTML
- Préparer les images
- Organiser les fichiers en un site cohérent
- Placer les fichiers sur un serveur
- Contrôler que tout fonctionne

# Conseils

- Emulez un site que vous aimez
- Choisissez un style pour le site
- Restez simple
- Offrez une fonction de recherche
- Testez votre site





# Partie 4 : Extensions



*Ecole Normale, 23 août 2000*

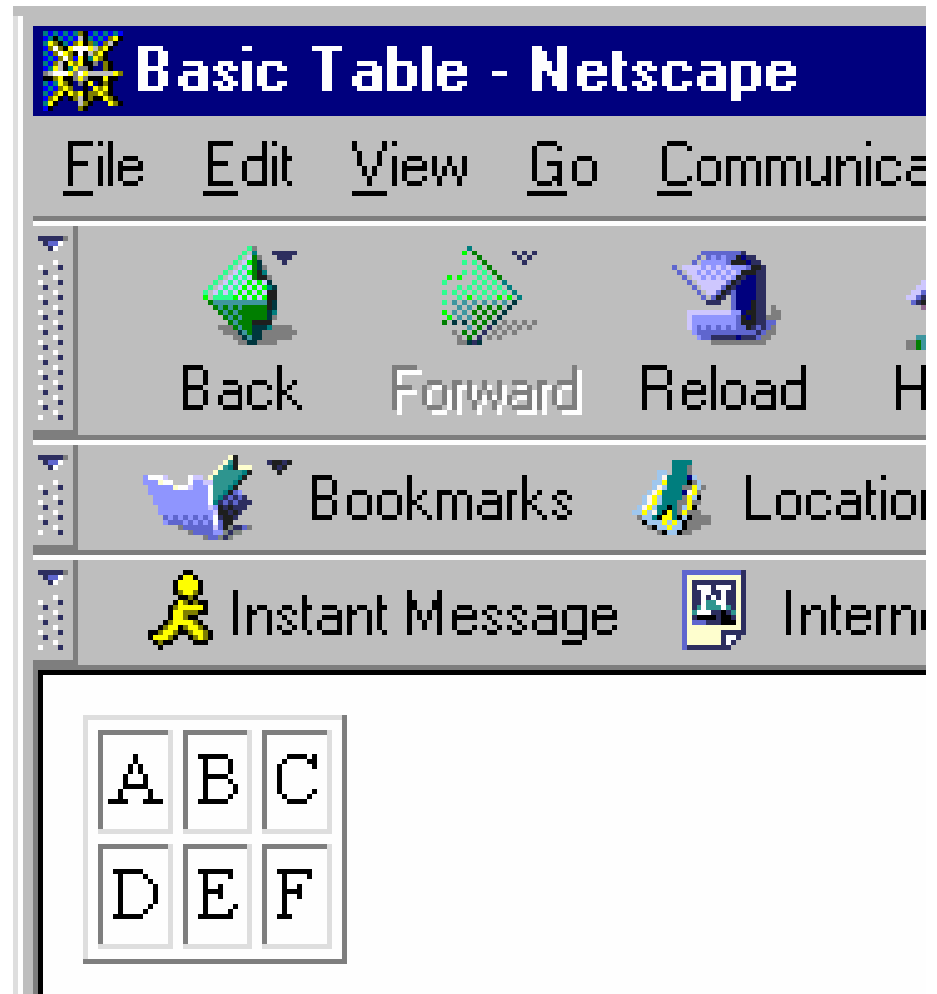


# 4.1 Les tables

```

<TABLE>
  <TR>
    <TD>A</TD>
    <TD>B</TD>
    <TD>C</TD>
  </TR>
  <TR>
    <TD>D</TD>
    <TD>E</TD>
    <TD>F</TD>
  </TR>
</TABLE>

```



# Ajouter un bord

- Un attribut de la balise table  
`<table border="2">`
- Donne par exemple un bord de 2 pixels



# Changer la taille d'une table

- Par défaut, les tables prennent autant de place que nécessaire pour afficher le texte
- Pour spécifier la taille, utilisez les attributs :
  - ◆ Height
  - ◆ Width
- `<table height="100" width="200">`
- Ces attributs peuvent aussi être utilisés dans la balise `<td>` pour changer la taille d'une cellule particulière



# Ajouter un légende

- Vient après la balise table  
`<caption> </caption>`
- Apparaît au-dessus de la table



# Introduire des espaces

- Attributs de la balise table

```
<table cellpadding="2" cellspacing="2">
```

- *Cellspacing* ajoute de l'espace entre les cellules sans changer la taille des cellules
- *Cellpadding* ajoute de l'espace autour du contenu des cellules



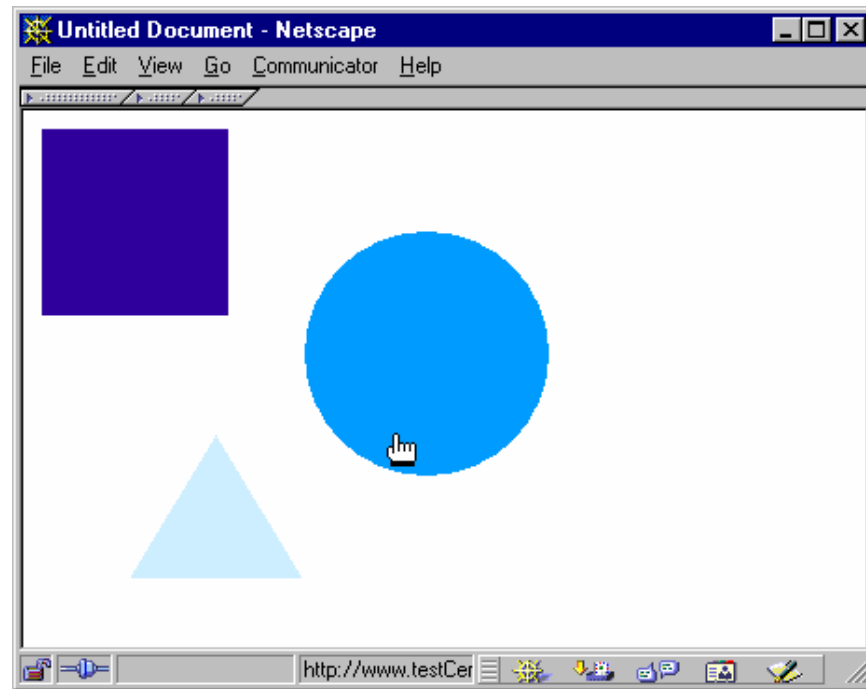
# Etendre des colonnes ou des lignes

- `<td colspan="2">` étend l'information sur 2 colonnes


- `<td rowspan="2">` étend l'information sur 2 lignes


## 4.2 Image map

- Utiliser certaines parties d'une image pour en faire des boutons
- Les coordonnées des zones sont mémorisées dans la page HTML



# Exemple

```

<map name="Map">
<area shape="rect" coords="-2,0,102,102"
href="http://www.testCarre.ch/">
<area shape="circle" coords="209,123,67"
href="http://www.testCercle.ch">
<area shape="poly" coords="94,168,139,241,51,243"
href="http://www.test.ch">
</map>
```

# Conseils

- Les zones clicables doivent être évidentes
- Incluez une alternative à l'image map, en général des liens textuels au bas de la page

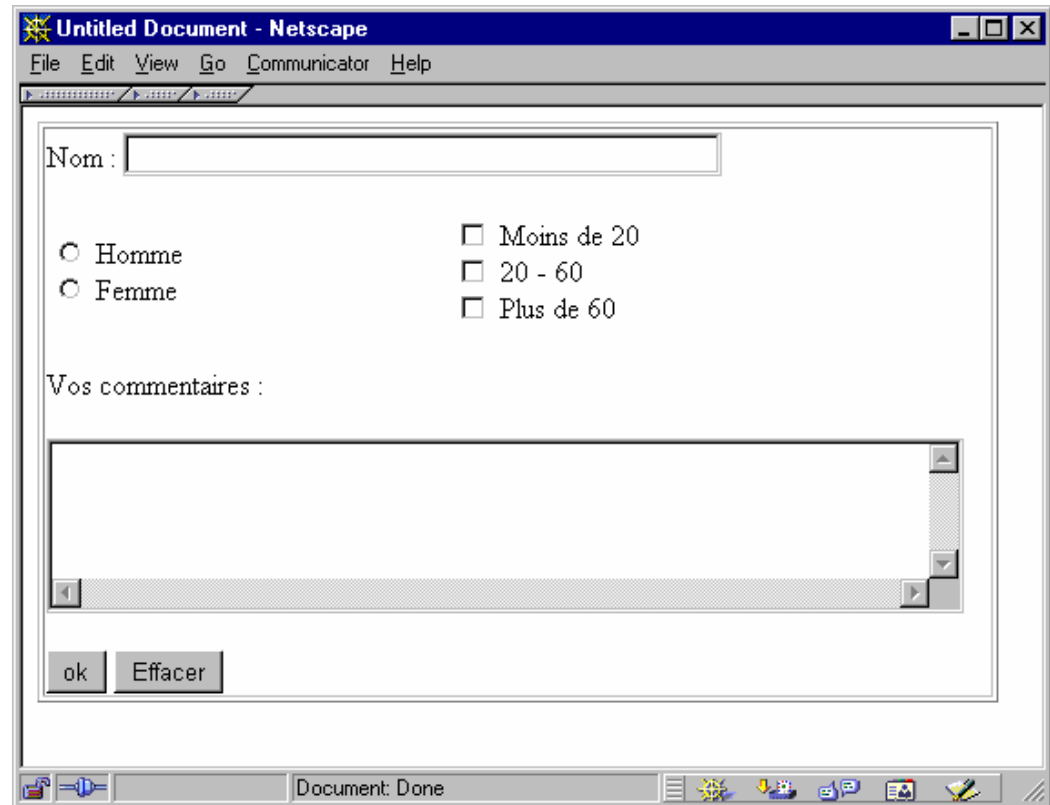


## 4.3 Les formulaires

`<form action="http://www.vive.com/cgi_program">`

...éléments d'entrée ...

`</form>`



# Les boutons

- Un formulaire est envoyé lorsque l'utilisateur clique sur le bouton submit.

```
<input type="submit" value="ok">
```

- Il est possible d'effacer tous les champs avec le bouton reset.

```
<input type="reset" value="Effacer">
```

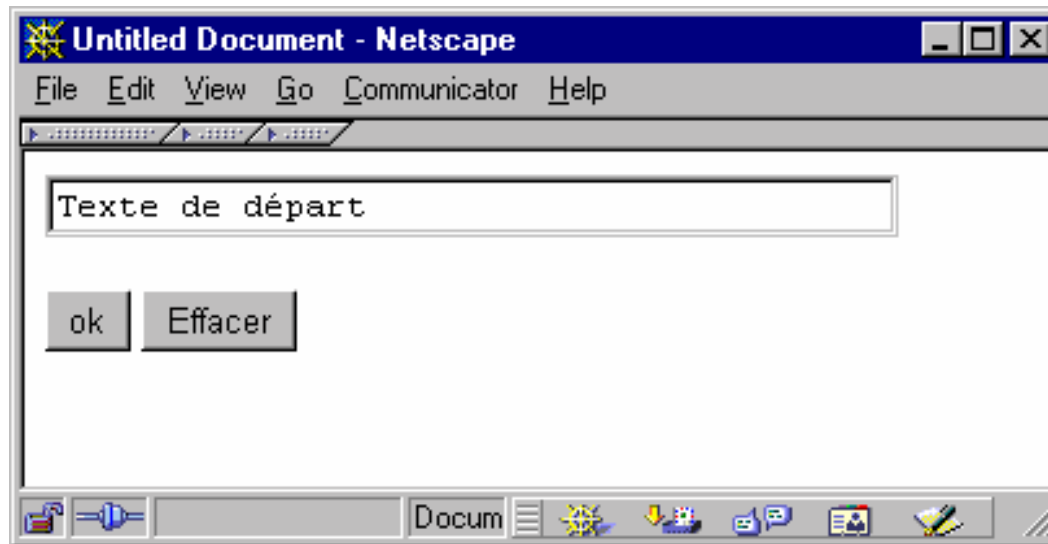




# Les champs d'entrée : texte

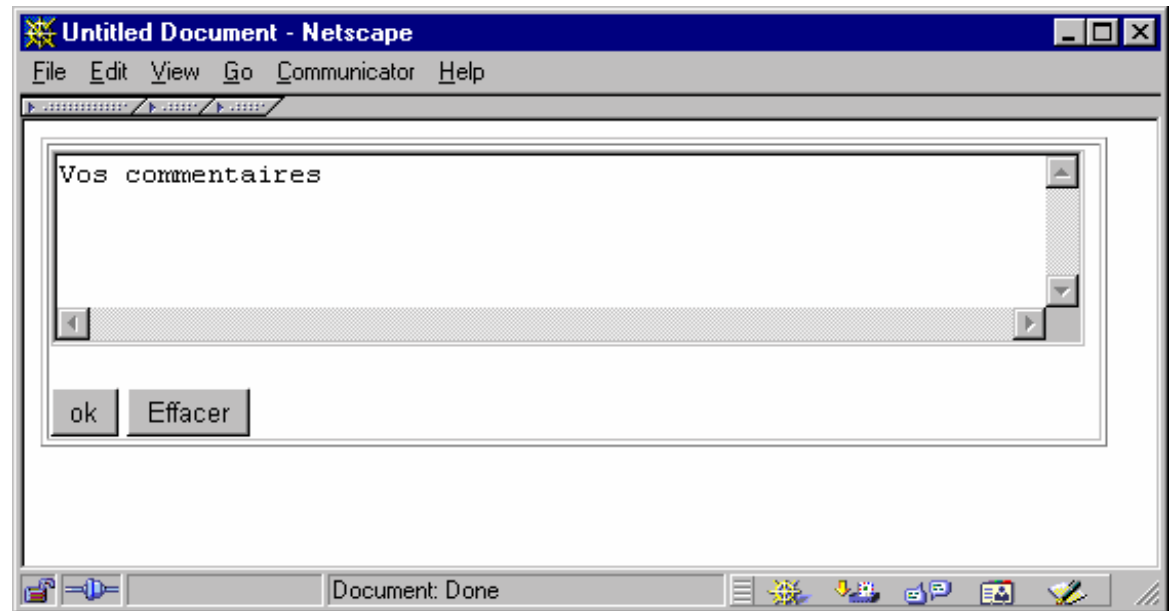
- Un champ de texte permet d'introduire une ligne  
`<input type="text" name="nom_du_champ" size="40">`
- L'attribut `size` détermine la taille du champ (en nombre de caractères)
- Il est possible de limiter le nombre de caractères avec l'attribut `maxlength`  
`<input type="text" name="nom" size="40" maxlength="20">`
- Il est possible de mettre une valeur par défaut avec l'attribut `value`  
`<input type="text" name="nom" size="40" value="texte de départ">`

# Exemple



# Les champs d'entrée : zone de texte

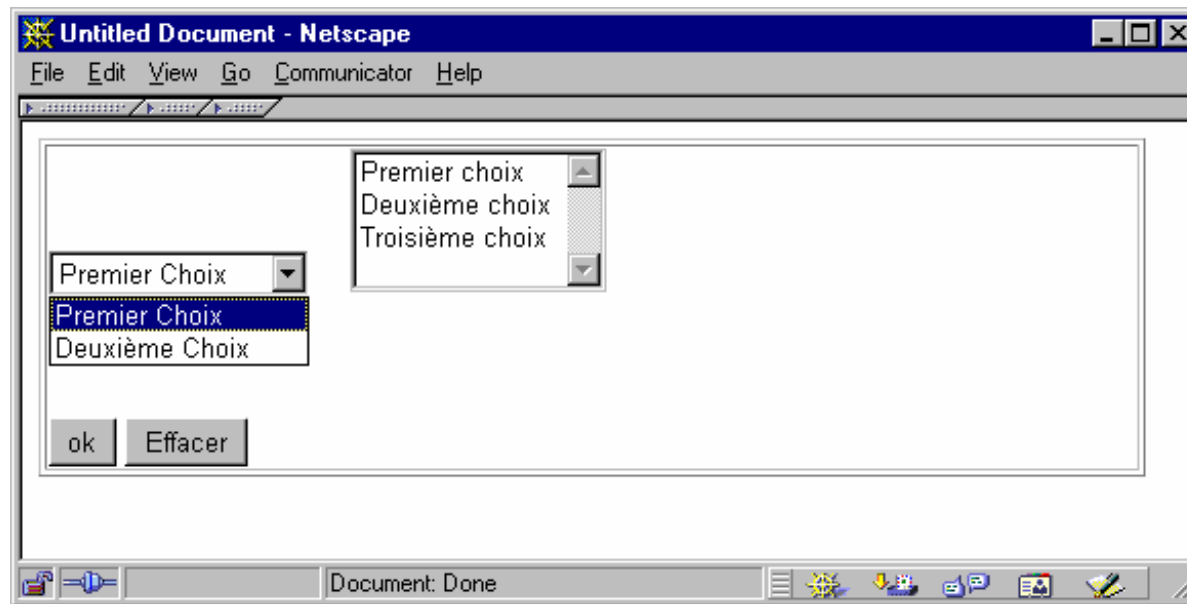
- Les champs de texte permettent d'introduire plusieurs lignes  
`<textarea name="nom_du_champ" rows="10" cols="50"> Vos commentaires</textarea>`



# Listes

```
<form method="post" action="">  
  <select name="select">  
    <option value="1">Premier Choix</option>  
    <option value="2">Deuxi&egrave;me Choix</option>  
  </select>  
  <select name="select2" size="4">  
    <option value="1">Premier choix</option>  
    <option value="2">Deuxi&egrave;me choix</option>  
    <option value="3">Troisi&egrave;me choix</option>  
  </select>  
  <input type="submit" name="Submit" value=" ok ">  
  <input type="submit" name="Submit2" value="Effacer">  
</form>
```

# Exemple

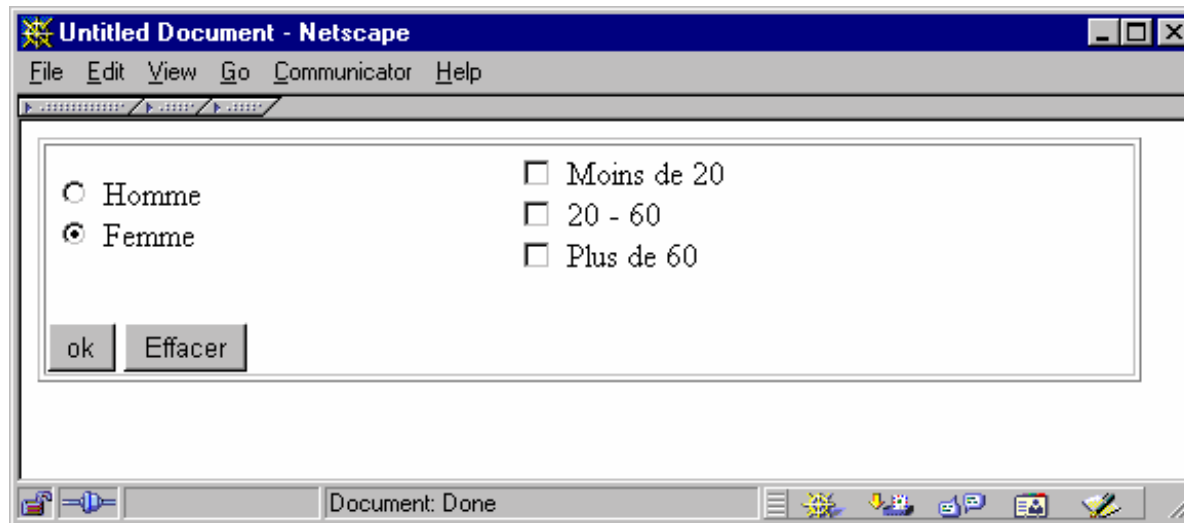


# Les boutons radio et les checkboxes

```
<form method="post" action="">  
<input type="radio" name="radiobutton">Homme <br>  
<input type="radio" name="radiobutton" checked>Femme<br>  
<input type="checkbox" name="checkbox">Moins de 20<br>  
<input type="checkbox" name="checkbox2">20 - 60<br>  
<input type="checkbox" name="checkbox3">Plus de 60<br>  
<input type="submit" name="Submit" value=" ok ">  
<input type="submit" name="Submit2" value="Effacer">  
</form>
```



# Exemple



## 4.4 Feuilles de style

- Sans feuilles de style, la balise <font> permet de définir un type, une taille et une couleur pour un texte
- Attributs :
  - ◆ **Size** : définit la taille de 1 à 7 (3 = normal) de manière absolue ou relative
  - ◆ **Color** : définit la couleur
  - ◆ **Face** : définit le type de caractère, par exemple : Arial, ...





# Désavantage du formatage classique

- Chaque zone doit contenir sa balise <font>. Par exemple, chaque ligne d'une table.
  - Chaque changement doit se faire à chaque balise <font>
- 
- ↳ Complicé
  - ↳ Risque d'erreur
  - ↳ Taille du fichier

# Déclaration de styles

- Les feuilles de style définissent l'apparence des balises
- Les modifications ne se font qu'à une seule place
- Un style est défini par un nom et une description  
`balise {propriété 1: valeur 1; propriété 2: valeur 2}`

- Exemple

`h3 {font-family:Arial; font-style: italic; color: green}`

- Une liste pour un attribut est aussi possible :

`Arial, Helvetica, sans-serif`



# Lieu de définition des styles

- Local : remplace la définition utilisant la balise <font>  
<h3 style="font-weight: bold">this will be bold</h3>
- Global : dans la balise <header>  
<style type="text/css">  
<!--  
balise {propriété 1: valeur 1; propriété 2: valeur: 2}  
-->  
</style>
- Extérieur : avec un fichier séparé :  
<link rel="stylesheet" href="style.css" type="text/css">

# Exemple simple

- Enlever les liens soulignés :

```
<style type="text/css">  
<!--  
a { text-decoration: none}  
-->  
</style>
```



# Exemple

```
<html><head>
<style type="text/css">
<!--
h4 {font: 17pt Arial,Helvetica"; font-weight: bold; color: maroon}
h2 {font: 15pt "Courier"; font-weight: bold; color: blue}
p {font: 12pt "Arial,Helvetica"; color: black}
-->
</style>
</head>
<body>
...
</body></html>
```

# Héritage (CSS)

- Une balise peut aussi hériter d'autres balises.
- On parle alors de *Cascading Style Sheet* (CSS)
- Note : en cas de conflit, la feuille de style la plus spécifique est celle qui est appliquée



# Plus de détails

- Définition des CSS :

<http://www.w3.org/TR/WD-CSS2/>

- Guides :

<http://msdn.microsoft.com/workshop/author/css/css.asp>

<http://msdn.microsoft.com/workshop/design/layout/css-des.asp>

- Exemples :

<http://www.hotwired.com/webmonkey/stylesheets/reference/examples.html>



# Quelques sites

- <http://www.w3.org/Style/css/>
- <http://www.htmlhelp.com/reference/css/>
- [http://www.canit.se/%7Egriffon/web/writing\\_stylesheets.html](http://www.canit.se/%7Egriffon/web/writing_stylesheets.html)
- <http://builder.cnet.com/Authoring/CSS/ss02.html>
- <http://www.htmlgoodies.com/beyond/classid.html>



## 4.5 Javascript : un survol

- Langage de programmation
- Interprété
- Généralement intégré au code HTML
- Ajoute une certaine interactivité



# Avantages et inconvénients

- Facile à apprendre
- Développement rapide
- Facile à débbugger
- Indépendant de la plateforme
- Petite période d'apprentissage
- Méthodes existantes limitées
- Difficile de cacher le code
- Peu d'outils de développement
- Pas toujours stable
- Pas toujours compatible

# Rudiments

- Habituellement, le code se place entre la fin de la balise de titre `</title>` et la fin de la balise d'entête `</head>`, mais peut se mettre aussi ailleurs
- Commence par :  
`<script language="JavaScript">`
- Se termine par :  
`</script>`
- Commentaires :  
`// et /* -- */`

# Exemple simple

```
<html>
<head>
<title>Premier script</title>
<script language="JavaScript">
document.write(" Hello world ! ")
</script>
</head>
<body>
...
</body>
</html>
```

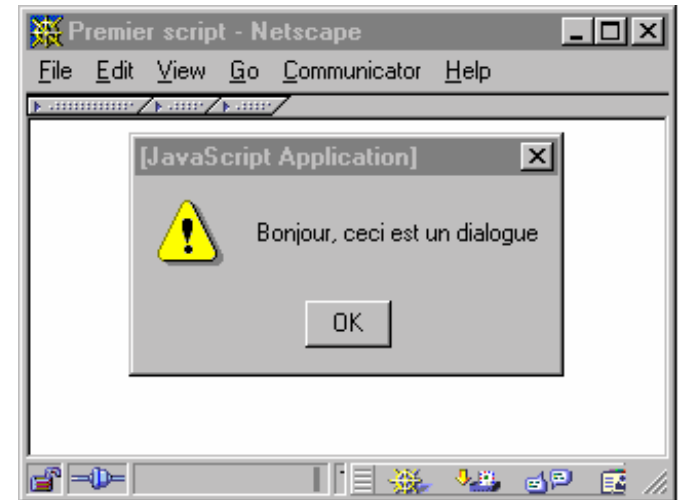
# Concepts principaux

- Evenements
- Fonctions et méthodes
- Variables
- Expressions
- Objets



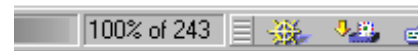
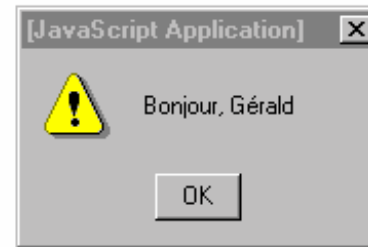
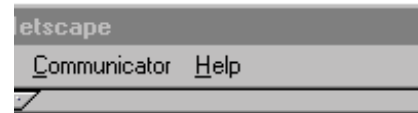
# Exemple : dialogue

```
<html>
<head>
<title>Premier script</title>
<script language="JavaScript">
<!-- Hide
alert("Bonjour, ceci est un dialogue");
// -->
</script>
</head>
<body>
</body>
</html>
```



# Exemple : variable

```
<script language="JavaScript">  
<!-- Hide  
var reponse = "Bonjour, Gérald"  
alert(reponse);  
// -->  
</script>
```

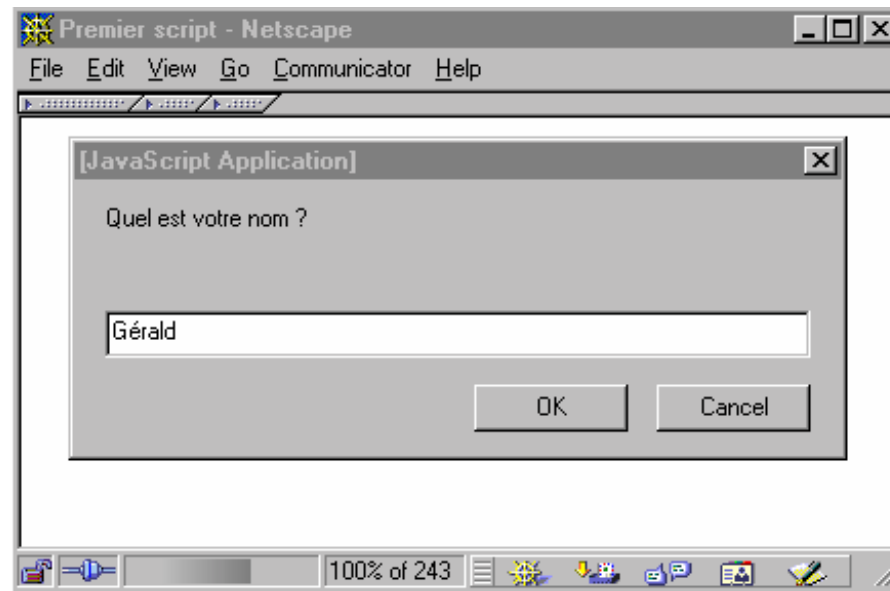


# Exemple : dialogue

```

<script language="JavaScript">
<!-- Hide
var reponse = prompt("Quel est votre nom ?", "");
alert("Bonjour, " + reponse);
// -->
</script>

```



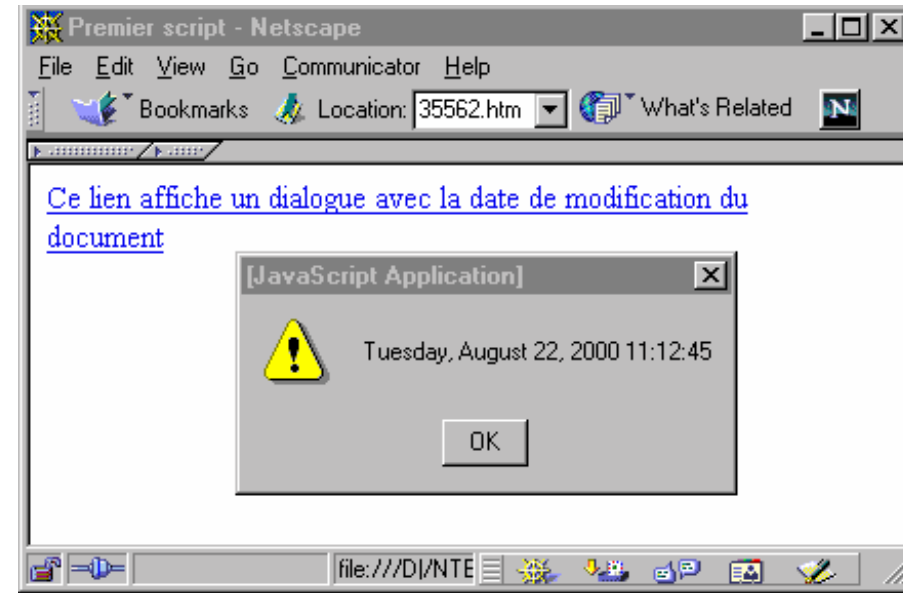


# Exemple : événements et fonctions

```

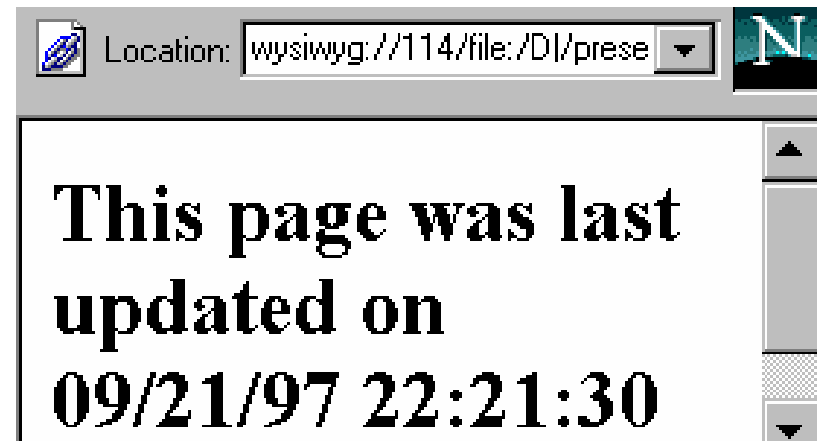
<html>
<head><title>Premier script</title>
<script language="JavaScript">
function giveDate() {
    alert(document.lastModified);
}
</script>
</head>
<body bgcolor=ffffff>
<a href="#" onClick="giveDate();" >
Ce lien affiche un dialogue avec la date de modification du document</a>
</body>
</html>

```



# Exemple : objets et fonctions

```
<script language="JavaScript">
<!-- Hide
document.write("<h2>This page was last updated on " +
document.lastModified + "</h2>")
// -->
</script>
```



## Exemple : if-then

```
<html><head><title>Premier script</title>
<script language="JavaScript">
var reponse = prompt("Aimez-vous ce cours ?", "répondez par oui ou
    non")
if (monkey_love == "oui") {
    alert("J'en suis très content");
}
</script>
</head>
<body bgcolor=ffffff>
</body> </html>
```

# Conseils

- Javascript vs. Java
- Javascript vs. Javascript
- Livre de référence
- *View source*
- Expérimentez



# Quelques exemples

- Touchez les carrés

[http://www.hotwired.com/webmonkey/98/03/image\\_swap.html](http://www.hotwired.com/webmonkey/98/03/image_swap.html)

- Calculs simples

<http://www.hotwired.com/webmonkey/98/03/calculations.html>

- Validation d'un formulaire

<http://mis.huji.ac.il/ttt/validate.html>

<http://mis.huji.ac.il/ttt/hase.validate.html> h



# Collections d'exemples

- Calculatrices, Calendriers, Menus, Table de multiplication, Utilisation des cookies, Encryption, ...

<http://gsb.haifa.ac.il/~sheizaf/ecommerce/javascript.examples.html>

- Divers

<http://mis.huji.ac.il/ttt/javascript.examples.html>

<http://tanega.com/java/java.html>

