

# Techniques de Programmation Internet

## Année Spéciale ENSIMAG

ENSIMAG 2002 - 2003

James L. Crowley

Séance 4

11 février 2003

## Frames et JavaScript

### Plan :

Fenêtre de navigation.....	2
Codage Directe.....	2
Tables .....	2
Frames .....	6
TARGETS :.....	7
BASE.....	7
JavaScript.....	8
Qu'est que Java Script.....	8
Messages de Alert.....	10
Fonctions.....	11
Ouverture d'un fenêtre.....	13
Communication entre fenêtres.....	14
Panneau de Contrôle.....	16
Verification d'un mot de passe.....	18
Navigation par menu.....	19

Organisation :

Les notes et les exemples sont disponibles surs :

<http://www-prima.imag.fr/Prima/jlc/Courses/Courses.html>

**Fenêtre de navigation**

Une fenêtre de navigation aide la navigation sur un site.

Une telle fenêtre peut être réalisé par plusieurs techniques.

Les méthodes possibles comprennent :

- 1) Par codage direct, avec une table
- 2) Par une "frameset"
- 3) Par un scripte "javascript"
- 4) Par un scripte php.

**Codage Directe**

Une possibilité est d'inclure un bloc de code au début de chaque page html.

Ce bloc donne acces à des pages principales du site.

Le menu est placé dans une table.

**Tables**

Des tables sont définies par les balises "<TABLE>...</TABLE>"

Les attributs sont : ALIGN, BORDER, HSPACE, VSPACE, WIDTH.

D'autre attribut sont supportés uniquement par Netscape ou Explorer

Netscape seulement : CELLPADDING, CELLSPACING,

Explorer seulement :

BORDERCOLOR, BORDERCOLORLIGHT, BORDERCOLORDARK, VALIGN.

<TABLE ALIGN= {LEFT, RIGHT}>      Par défaut, les tables coupe la texte.

<TABLE >

Ici du texte.

TABLE
-------

Encore du texte.

L'attribut ALIGN spécifie que le texte droit passé à côté de la table.

```
<TABLE ALIGN= LEFT>
```

TABLE
-------

Ici du texte  
À droite de la table.

Voici un exemple d'une `<A href = "www/Seance4/menu1.html">page modèle</A>`

Les lignes d'une table sont définies avec `<TR>` et `<TD>`

```
<TABLE>
```

```
<TR> Table row <TH>Column Heading</TH><TH>Column Heading</TH></TR>
```

```
<TR> Table row <TD>Table Cell</TD><TD>Table Cell</TD></TR>
```

```
</TABLE>
```

Avec un menu dans une table alignée à gauche, si le texte dépasse le menu, il prend tout le page. Cela n'est pas tres joli.

On peut éviter ceci avec une table dans une table.

Le page est une table composée de deux colonne. La colonne de gauche contient une seconde table d'une colonne. La deuxième table contient le menu.

<table border="1"> <tr> <td>Menu</td> </tr> <tr> <td> </td> </tr> <tr> <td> </td> </tr> <tr> <td> </td> </tr> <tr> <td> </td> </tr> </table>	Menu					Le Page
Menu						

```
<TABLE><TR VALIGN=top>
```

```
<TABLE>
```

```
<TR><TD> MENU-1</TD><TR>
```

```
<TR><TD> MENU-2</TD><TR>
```

```
</TABLE>
```

```
<TR><TD>
```

Le contenu du page

```
</TD><TR></TABLE>
```

Une `<A href = "www/Seance4/menu2.html"> exemple </A>`

Sa source est :

```
<HTML>
<HEAD>
<TITLE>Exemple menu2.html</TITLE>
</HEAD>
<BODY BGCOLOR=#FFFFFF VLINK=#0000FF AVLINK=#0000FF
VLINK=#0000FF>
<H1 ALIGN="CENTER">A hard coded index table </H1>

<!-- Open a table for the entire page -->
<!-- This assures that the text can be beside the index
table -->

<TABLE>
<TR VALIGN=top>
<TD bgcolor="#FFFFFF">

<!-- Open a second table for the index -->
<!-- This assures that the text can be beside the index
table -->

<TABLE >
<TR>
<TD BGCOLOR="#00AAAA" WIDTH="160" CELLSPACING="2"
CELLPADDING="0">
<a href=" ../ ../ASI.TPI.S4.html">Seance 4</a>
</TD>
</TR>

<!-- An HREF is placed in each row. It has a blue background -->
<TR><TD BGCOLOR="#00AAAA">
<a href="menu1.html">menu1.html</a>
</TD></TR>

<!-- This is the current page. It is NOT an HREF and highlighted in RED -->
```

```
<TR><TD BGCOLOR="#FF0000">
<FONT COLOR="#FFFFFF" FACE="Arial, Helvetica">
menu2.html
</FONT></TD></TR>
```

```
<!-- An HREF is placed in each row. It has a blue background -->
<TR><TD BGCOLOR="#00AAAA">
<a href="menu3.html">menu3.html</a>
</TD></TR>
```

```
<!-- and empty row read for use -->
<TR><TD BGCOLOR="#00AAAA"></TD></TR>
</TABLE>
```

```
<!-- this is the main body of the page -->
<CENTER>
<TD>
<H1>An index table</H1>
</CENTER>
<HR>
```

This is an example of a hard coded index table.

Every page must be a table of tables. The outer table has two columns. The left column contains a table with one column containing the menu of HREF's to other pages.

```
</TD>
</TR>

</TABLE>
</BODY>
</HTML>
```

Un alternatif est de faire une "Frame Set".

## Frames

Les Frames était inventé par Netscape, il y a quelques années.  
Desormais, Ils font partie de la définition officielle de HTML 4.0.  
Ils sont supportés par les versions récentes d'Explorer

Un "frameset" est un maillage de la fenêtre.

On peut faire un menu de navigation avec une Frameset composé de deux cellules.  
À gauche de la page, on place une cellule composée d'une page de navigation avec les hrefs pour les autres pages. À droite, on affiche les pages.

```
<FRAMESET COLS="160,*">
```

Les attributs ROWS et COLS définit la taille et le nombre de lignes et colons.  
Les attributs sont entourés de "quotes".  
Le nombre d'argument détermine le nombre de colonnes ou lignes.  
Les valeurs déterminent leur taille.

Exemple :

ROWS ="100,150, \*" - Trois lignes de taille 100 pixels, 150 pixels, et la reste du page.

COLS = "20%, \*" - deux colonnes. Le premier a 20% du page. Le deuxième occupe le reste.

Le contenu des fenêtres est spécifié par une balise <FRAME>

```
<FRAMESET COLS="160,*">  
  <FRAME SRC="toc.html">  
  <FRAME NAME="main" SRC="bienvenu.html">  
</FRAMESET>
```

Les attributs comprennent :

SRC - L'URL de l'objet d'afficher. (HTML, image, filme, etc.)

NAME - Nom pour la référence de la fenêtre, for usage avec <A>

NORESIZE - Inhibition de la manipulation de la taille.

SCROLLING {YES, NO, AUTO} pour le défilement.

```
<NOFRAMES> </NOFRAMES>
```

Si le navigateur ne support pas les frames, le contenu de NOFRAMES est affichés.

**TARGETS :**

```
<HTML>
<HEAD></HEAD>
<BODY>
<H3>Menu</H3>
<A HREF="p1.html" TARGET="main">Page 1</A>
<A HREF="p2.html" TARGET="main">Page 2</A>
</BODY>
```

Un simple HREF dans un menu remplacerait le contenu de la fenêtre du menu. On voudrait plutôt remplacer le contenu de l'AUTRE fenêtre. Avec TARGET, on peut spécifier le fenêtre cible.

Les TARGETs prédéfinit :

\_blanc : Une nouvelle fenêtre vide.  
\_self : valeur par défaut - le fenêtre de la balise <A>  
\_parent : La fenêtre du parent. Pour un seul niveau \_parent=\_self  
\_top : le fenêtre du sommet de lahiérarchie.

**BASE**

On peut éviter de spécifier TARGET dans chaque <A> avec <BASE>

```
<HTML>
<HEAD><BASE target=main></HEAD>
<BODY>
<H3>Menu</H3>
<A HREF="p1.html">Page 1</A>
<A HREF="p2.html">Page 2</A>
</BODY>
```

## JavaScript

### Qu'est que Java Script

JavaScript est une langage "embedded" de script interprété par les navigateurs moderne. JavaScript a fut inventé par Netscape, mais il est également supporté par MS Explorer. JavaScript utilise le syntaxe du Java (inventé par SUN). Attention : JavaScript n'est pas Java!

JavaScript est encastré dans le code "html".  
Un script est facilement inclus dans un page html.

Le "wrapper" classique pour JavaScript est :

```
<SCRIPT LANGUAGE=JAVASCRIPT>  
<!-- // -->  
</SCRIPT>
```

Java est un langage a objet, mais faiblement typé.  
Dont les objets pre-définis sont :

"Navigator" a les attributs Navigator.appName et Navigator.appVersion.  
Screen avec les attributs Screen.width, Screen.height, Screen.colorDepth, Screen.pixelDepth

Les objet Java ont également les méthodes (procedures).  
Par exemple, l'objet "window" inclut :  
window.alert(),window.open(),window.close(),window.prompt()  
et bien d'autre



## Exemple S4.1.html

```

<HTML>
<HEAD>
<TITLE>Un Petit Script</TITLE>
</HEAD>
<BODY BGCOLOR=WHITE>
<H1>
<SCRIPT LANGUAGE=JAVASCRIPT TYPE="TEXT/JAVASCRIPT">
<!-- Projection des navigateurs avant Netscape 3.0

document.write(navigator.appName + " " +
    navigator.appVersion)

// Fin du zone protege -->
</SCRIPT>
</H1>
<NOSCRIPT>
    <H2>Cet page exige JavaScript</H2>
    <A HREF="http://home.netscape.com">Netscape</A> Home
</NOSCRIPT>
</BODY>
</HTML>

```

Les Scripts peuvent être placé entre <HEAD> et </HEAD>  
ou bien entre <BODY> et </BODY>.

Il sont placé dans une zone <SCRIPT> ... </SCRIPT>

<SCRIPT> peuvent prendre les attributs LANGUAGE et TYPE

Les commentaire en Java Script commence par "/\*" sur un ligne  
ou bien entre /\* et \*/ pour les ligne multiples.

On doit protéger les navigateurs non-equipé de Java Script avec un  
balise de commentaire <!-- fermé par -->

Un balise <NOSCRIPT> permet d'afficher un message d'erreur si  
le Browser n'est pas équipé de JavaScript.

**Messages de Alert**

On peut créer un boit avec un message d'alerte avec "alert".  
Java Permet les if () else; du style "C".

On peut rediriger l'utilisateur a une autre script avec  
l'attribut "location" de l'objet "window" : window.location

**Exemple S4.2.html**

```
<HTML>
<HEAD>
<TITLE>Exemple d'un ALERT</TITLE>
</HEAD>

<BODY BGCOLOR=WHITE>
<SCRIPT LANGUAGE=JAVASCRIPT TYPE="TEXT/JAVASCRIPT">
<!-- Projection des navigateurs avant Netscape 3.0

//Navigator indique le Navigateur courant.
if(navigator.appName == "Netscape")
{
    alert("Netscape Detected. Tres Bien!")

// window est le fenetre courant
// window.location est son address.
    window.location = "bienvenue.html"
}
else
{
    alert("Vous n'est pas sur Netscape!")
}
// Fin du zone protege -->
</SCRIPT>

<H1> Alert</H1>
<NOSCRIPT>
    <H2>Cet page exige JavaScript</H2>
    <A HREF="http://home.netscape.com">Netscape</A>
</NOSCRIPT>

</BODY>
</HTML>
```

avec le script bienvenu.html

```
<HTML>
<HEAD>
<TITLE>Bienvenu avec Netscape</TITLE>
</HEAD>
<BODY BGCOLOR=WHITE>
<H1> Netscape est bienvenu ici</H2>

<A HREF="S4.2.html">S4.2.html</A>
</BODY>
```

### **Fonctions**

On peut définir les fonctions JavaScript qui sont déclenché par les evenements.

Par exemple un fonction de changer le couleur du fond est

```
fonction setColor(colorstr)
{
    document.bgColor = colorstr;
}
```

## Exemple S4.3.html

```
<HTML>
<HEAD>
<TITLE>Exemple d'un fonction</TITLE>

<SCRIPT LANGUAGE=JAVASCRIPT>
<!-- Hide script from older browsers
function setColor(colorstr)
{
    document.bgColor = colorstr;
}
// End hiding script from older browsers -->

</SCRIPT>
</HEAD>

<BODY BGCOLOR=WHITE>
<H2>
Changer le couleur du fond
</H2>
<HR>
<FORM>
<INPUT TYPE="button" VALUE="blanc"
onClick="setColor('#ffffff')">
<INPUT TYPE="button" VALUE="gris"
onClick="setColor('#c0c0c0')">
<INPUT TYPE="button" VALUE="bleu ciel"
onClick="setColor('#64bae2')">
<INPUT TYPE="button" VALUE="citron"
onClick="setColor('#fff8b9')">
<INPUT TYPE="button" VALUE="mauve"
onClick="setColor('#c0c0f0')">
<INPUT TYPE="button" VALUE="peche"
onClick="setColor('#ffcb9f')">
<INPUT TYPE="button" VALUE="rouge"
onClick="setColor('#ff000')">
<INPUT TYPE="button" VALUE="vert"
onClick="setColor('#00ff00')">
<INPUT TYPE="button" VALUE="bleu"
onClick="setColor('#0000ff')">
</FORM>
</BODY>
</HTML>
```

**Ouverture d'un fenêtre**

## Exemple S4.4.html

On peut ouvrir un fenêtre avec la méthode `window.open()` de la classe `window`.

```
<HTML>
<HEAD>

<TITLE>Window Test</TITLE>
<SCRIPT LANGUAGE=JAVASCRIPT>
<!-- Hide script from older browsers
function newWindow()
{
    imgWindow = window.open('Skier.mov.gif', 'imgWin',
'width=175,height=150')
}
// End hiding script from old browsers -->
</SCRIPT>

</HEAD>
<BODY BGCOLOR=WHITE>
<CENTER><H1>Skiing in Grenoble</H1>
<H3>Its time for Spring Skiing</H3>

<A HREF="javascript:newWindow()">Click here</A> to see why.
</CENTER></BODY>
</HTML>
```

**Communication entre fenêtres**

Une fenetre fille peut communiquer avec une fenetre mere par "UpdateParent".

Exemple S4.5.html et S4.6.html

```
<HTML>
<HEAD>
<TITLE>Page Pere</TITLE>
</HEAD>
<BODY BGCOLOR=WHITE>
<SCRIPT LANGUAGE=JAVASCRIPT>
<!--
newWindow = window.open('S4.6.html', 'newWin')
// -->
</SCRIPT>
<CENTER><H1>Page Mere</H1>
<FORM NAME=outputForm>
<INPUT TYPE=TEXT SIZE=20 NAME=msgLine VALUE="">
</FORM></CENTER></BODY>
</HTML>
```

avec le script S4.6.html

```
<HTML>
<HEAD>
<TITLE>Page fils </TITLE>
<SCRIPT LANGUAGE=JAVASCRIPT>
<!--
function updateParent(textField)
{
    opener.document.outputForm.msgLine.value = "Bonjour " +
textField.value + "!";
    window.close()
}
// -->
</SCRIPT>
</HEAD>
<BODY BGCOLOR=WHITE>
<H1>Quel est votre nom?</H1>
<FORM>
<INPUT TYPE=TEXT ONBLUR="updateParent(this)" SIZE=20>
```

```
</FORM>
</BODY>
</HTML>
```

Exemple S4.7.html

```
<HTML>
<HEAD>
<TITLE>Main Window</TITLE>
</HEAD>
<BODY BGCOLOR=WHITE>
<CENTER><H1>Ceci est le Pere</H1>

<SCRIPT LANGUAGE=JAVASCRIPT>
<!--

newWindow = window.open('', 'newWin',
'toolbar=no,location=yes,scrollbars=yes,resizable=yes,width=
400,height=300')

newWindow.document.write("<HTML><HEAD><TITLE>Generated
Window</TITLE></HEAD><BODY BGCOLOR=WHITE><H2>Ceci est
nouveau fenetre</H2>")
newWindow.document.writeln("<BR>Ceci est un test<BR>")
newWindow.document.write("</BODY></HTML>")
newWindow.document.close()
// -->
</SCRIPT>
</BODY>
</HTML>
```

**Panneau de Contrôle**

## Exemple S4.8.html

```

<HTML>
<HEAD>
<TITLE>Creation d'un panneau de controle</TITLE>
<SCRIPT LANGUAGE=JAVASCRIPT>
<!--
newWindow = window.open('S4.9.html', 'newWin',
'width=300,height=300,left=600,top=0')
// -->
</SCRIPT>

</HEAD>
<BODY BGCOLOR=WHITE>
<CENTER><H1>Jim Crowley's Web Page</H1>
<H2></H2>
<IMG SRC="jim.GIF">
</CENTER></BODY>
</HTML>

```

avec le Panneau de Controle S4.9.html

```

<HTML>
<HEAD>
<TITLE>Panneau de Controle</TITLE>
<SCRIPT LANGUAGE=JAVASCRIPT>
<!--
function updateParent(newURL)
{
    opener.document.location = newURL
}
// -->
</SCRIPT>
</HEAD>
<BODY BGCOLOR=WHITE>
<CENTER><H1>Control Panel</H1>
<ul>
<H3><A HREF="javascript:updateParent('jlc.html')">Jim
Crowley</A><BR>
<A
HREF="javascript:updateParent('jlc.html#ResearchActivities')
">Research Activities</A>

```



```
<A
HREF="javascript:updateParent('jlc.html#Diplomas')">Diplomas
</A>
<A
HREF="javascript:updateParent('jlc.html#ProfessionalActivities')">Professional Activities</A>
<A
HREF="javascript:updateParent('jlc.html#Publications')">Publications</A>
<A
HREF="javascript:updateParent('http://volmontagne.decollage.org/english/')">Mountain Flying</A>
</ul>

</CENTER></BODY>
</HTML>
```

**Verification d'un mot de passe**

JavaScript peut être très utile pour la vérification des informations venant d'un formulaire.

Exemple S4.10.html

```

<HTML>
<HEAD>
<TITLE>Exemple d'une Forme</TITLE>
<SCRIPT LANGUAGE=JAVASCRIPT>
<!--
function validForm(passForm)
{
    if (passForm.passwd1.value == "")
    {
        alert("You must enter a password")
        passForm.passwd1.focus()
        return false
    }
    if (passForm.passwd1.value != passForm.passwd2.value)
    {
        alert("Passwords do not matched. Please try again.")
        passForm.passwd1.focus()
        passForm.passwd1.select()
        return false
    }
    return true;
}
// -->
</SCRIPT>
</HEAD>
<BODY BGCOLOR=WHITE>
<H2>Create a Password</H2>
<HR>
<FORM onSubmit="return validForm(this)"
action="thanks.html">
    <P>Your Name: <INPUT TYPE=TEXT size=30">
<P>Choose a Password: <INPUT TYPE=PASSWORD NAME="passwd1">
<P>Retype Password: <INPUT TYPE=PASSWORD NAME="passwd2">
<P><INPUT TYPE=SUBMIT Value="Submit">&nbsp;<input
TYPE=RESET>
</FORM>
</BODY>
</HTML>

```

**Navigation par menu**

En lieu d'exiger que l'utilisateur tape "submit", on peut permettre la dépacement par menu.

## Exemple S4.11.html

```
<HTML>
<HEAD>
<TITLE>Navitation par menu</TITLE>
<SCRIPT LANGUAGE=JAVASCRIPT>
<!--
function jumpPage(newLoc)
{
    newPage = newLoc.options[newLoc.selectedIndex].value
    if (newPage != "")
    {
        window.location.href = newPage
    }
}
// -->
</SCRIPT>
</HEAD>
<BODY BGCOLOR=WHITE>
<H2>Choix des Exemples</H2>
<HR>
<FORM ACTION="thanks.html" METHOD=GET>
<SELECT NAME="newLocation"
onChange="jumpPage(this.form.newLocation)">
<OPTION VALUE="" SELECTED>Select a topic
<OPTION VALUE="S4.1.html" >S4.1.html
<OPTION VALUE="S4.2.html" >S4.2.html
<OPTION VALUE="S4.3.html" >S4.3.html
<OPTION VALUE="S4.4.html" >S4.4.html
<OPTION VALUE="S4.5.html" >S4.5.html
<OPTION VALUE="S4.6.html" >S4.6.html
<OPTION VALUE="S4.7.html" >S4.7.html
<OPTION VALUE="S4.8.html" >S4.8.html
<OPTION VALUE="S4.9.html" >S4.9.html
<OPTION VALUE="S4.10.html" >S4.10.html
</SELECT>
</FORM>
</BODY>
</HTML>
```