



Using Dreamweaver

8 – Dynamic HTML Menus

Dynamic HTML (DHTML) is a term that refers to websites that use a combination of **HTML**, scripting such as **JavaScript**, **CSS** and the **Document Object Model (DOM)**. HTML and CSS are used to create the page, the DOM allows different elements of the page to be identified as objects and scripting is used to make changes to the page after it has loaded. This allows pages to be designed that can change depending on what the user does. For example, part of the page format or content may change when the user clicks a button or moves their mouse over something. A common use for this is to create DHTML menus for site navigation as we will be doing in this section.

When DHTML first began to be used in the late 90s, there were many problems since the creators of the two main browsers, Microsoft and Netscape implemented it in different ways. This caused many developers to be cautious about using it since it was difficult to use it in a way that would work in both browsers. In more recent years, browsers have generally become better about following standards so DHTML is far more likely to work properly in different browsers.

Dreamweaver takes a lot of the pain out of DHTML by allowing you to avoid having to write JavaScript code. We actually used a bit of DHTML with the Francies Flowerpots site when we set up the image swapping in the navigation bar. We will use Dreamweaver to create menus on the page by using a series of **layers**. Layers use either the **DIV** or **SPAN** tag in HTML (usually DIV since this produces better results in older browsers). When sections of the page are enclosed in a SPAN or DIV tag, then CSS can be used to position them anywhere on the screen with a great deal of precision. This can be used to create rectangular shaped areas on the screen that when fit together, can form navigation menus. DHTML can then be used to hide or display these menus when needed.

Using layers for page layout

In the past, websites have often been designed using tables for page layout. Using layers with CSS enables you to position things on a page exactly without the need for tables. This technique will be used in the following exercise.

Note Positioning using styles won't be recognised in older web browsers that aren't compatible with CSS. Layers that are created using HTML's DIV tag will appear in older browsers as a series of paragraphs, one after the other so the order that they appear in the HTML code might be important if compatibility with older browsers is consideration for your site.

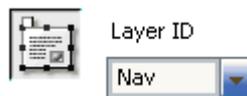
We will be doing a website for a small computer games directory. Before doing these exercises get the DHTML exercises from <http://oneil.com.au/pc/dreamweaver/dhtml.exe>. If you are doing these exercises as part of a class, your instructor may have made them available in another convenient location such as a shared network drive.

Exercise 1 – Creating a Page Layout with Layers

- 1) From the DHTML exercise files, open the index.html file. Currently it is a blank page with little more than a background image.
- 2) From the **Insert** menu, select **Layout Objects** and then select **Layer (AP Div** in some newer versions of Dreamweaver). A new layer will appear on the page.
- 3) Move your mouse over the edge of the layer and click to select it. When it is selected, small squares will appear around its edges. These can be used for resizing the layer with your mouse.



- 4) With the layer selected, you will see its properties in the **Properties** panel. In the properties, change the **Layer ID** to *Nav*.



- 5) Some of the other properties are explained below, along with what you need to change each property to. Make the changes indicated.

Property	Explanation	Change to:
L	Distance from the left of the page. This can be set as a number of pixels (E.g. 100px) or as a percentage of the screen width (E.g. 20%)	0
T	Distance from the top of the page.	0
W	Width of the layer	100px
H	Height of the layer	400px
Z-Index	Stacking order with a number from 1-10. If objects on the page overlap, the one with the higher number in the Z-Index property will be in front of the one with the lower number	1
Vis	Whether the object is hidden or visible when the page loads. DHTML can be used to change this property after the page is loaded. This is how we will make our menus appear and disappear on user actions.	Visible or Default
Bg image	Image to be displayed in the background of the layer	gd_bar.jpg
BG color	The background colour for the layer	Leave blank



Your page will now have a single layer, containing a background image which fits the dimensions of the layer. Our menus will sit on top of this layer when we create them later on. Now we have a layer for the navigation bar, we will create an additional layer for the page content to be placed in.

- 6) From the **Insert** menu, select **Layout Objects** and then select **Layer**. A new layer will appear on the page.
- 7) Click on the border of the new layer to select it.
- 8) Set the properties for the new layer as follows.

	Layer ID	L	105px	W	680px	Z-Index	1	Bg image	
	content	I	5px	H	600px	vis	default	Bg color	
Overflow	auto	Clip: L		R		T		B	

Note The borders around the edge are visual guides for Dreamweaver's design view. They won't be visible in the web browser. You can save and preview the page to check if you like.

- 9) Click inside the new layer (content).

If you can't click inside the layer, in some versions on Dreamweaver you might need to **right-click** on the layer and then select **Element View** and then **Full**.

- 10) Insert the image *gd_heading.png*.

- 11) Centre align the image and enter *Games Directory* for the image's **Alt** property.

- 12) On a new line under the heading, enter the following text.

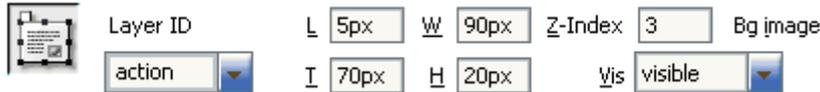
Welcome to my games directory. Here you will find a guide to some of the best current PC games in the market. Choose a genre from the menu on the left and then select a game from the menu to go to the website for that game.

- 13) Find or create some suitable images to add to the page (If you don't have any good ones, you can right-click and save the images from the sample version of these exercises at <http://oneil.com.au/pc/dreamweaver/dhtml/index.html>). The page should currently look similar to the example below.

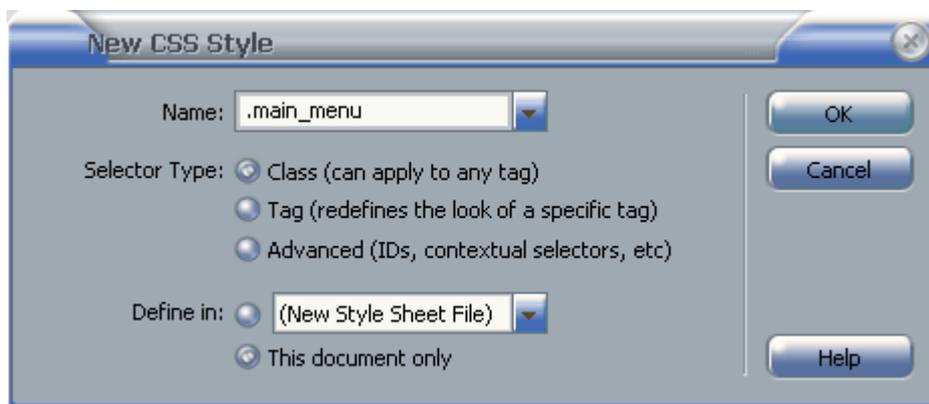


Exercise 2 – Creating the Menus

- 1) Click outside the existing layers.
- 2) From the **Insert** menu, select **Layout Objects** and then select **Layer**. A new layer will appear on the page.
- 3) Click the edge of the layer to select it.
- 4) Give the layer the following properties.



- 5) Click in the new layer and type the text *Action*.
- 6) Select the layer by clicking its border again.
- 7) In the **CSS Styles** panel click the **New CSS Rule** button at the bottom.
- 8) The style options box should appear.
- 9) Complete the new style options as follows and click **OK**.



- 10) Under the **Type** category, choose some type options that you think will look good.
- 11) In the **Block** category, set the **Text-align** property to *center*.
- 12) In the **Extensions** category, enter *Pointer* for the **Cursor** option.

Tip The hand option for the cursor property will mean that the mouse pointer will change to a hand when the mouse moves over the layer. This only works in Internet Explorer. If you want the same effect in Internet Explorer and Firefox (other browsers too), then enter pointer instead of hand. This option isn't in Dreamweaver's list but that doesn't stop you from entering it anyway.

- 13) Click **OK** and then **Done** to complete the creation of the new style.
- 14) In the layer's **Class** property select *main_menu* to apply the newly created style.
- 15) Create additional layers as shown on the following page.

Note Before you create a new layer, you may need to click outside the existing layers. Otherwise, you may be creating a layer within a layer (nested layer).

Layer ID	L	5px	W	90px	Z-Index	3	Bg image	<input type="text"/>	Class	main_menu
rpg	I	90px	H	20px	Vis	visible	Bg color	<input type="text"/>		

Layer ID	L	5px	W	90px	Z-Index	3	Bg image	<input type="text"/>	Class	main_menu
sim	I	110px	H	20px	Vis	visible	Bg color	<input type="text"/>		

Layer ID	L	5px	W	90px	Z-Index	3	Bg image	<input type="text"/>	Class	main_menu
sport	I	130px	H	20px	Vis	visible	Bg color	<input type="text"/>		

Layer ID	L	5px	W	90px	Z-Index	3	Bg image	<input type="text"/>	Class	main_menu
strategy	I	150px	H	20px	Vis	visible	Bg color	<input type="text"/>		

16) Add text to each of the layers so that they look like the example below.



17) Save and preview the file.

18) Take a look at the HTML and you will see that each layer has been created using a **DIV** tag. **CSS** (in the top section of the code) is then used to position each DIV tag using its **ID**.

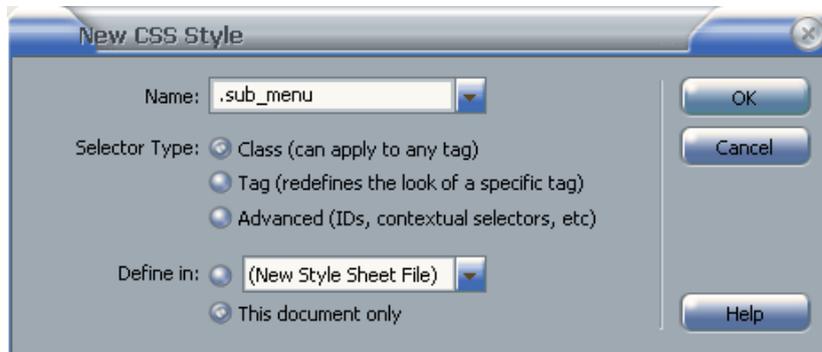
```
<p align="center"></p>
</div>
<div class="main_menu" id="action">Action</div>
<div class="main_menu" id="rpg">Role Playing</div>
<div class="main_menu" id="sim">Simulation</div>
<div class="main_menu" id="sport">Sport</div>
<div class="main_menu" id="strategy">Strategy</div>
</BODY>
```

```
#sim {
    position: absolute;
    width: 90px;
    height: 20px;
    z-index: 3;
    left: 5px;
    top: 110px;
    visibility: visible;
```

Exercise 3 – Adding a Sub Menu

Now that layers have been created for the main menu, we'll make some more layers for the first of our sub menus - Action. First we'll create a new style to be used by the sub menus.

- 1) Create a new style class called *sub_menu* (note that a class needs to have a . before its name)



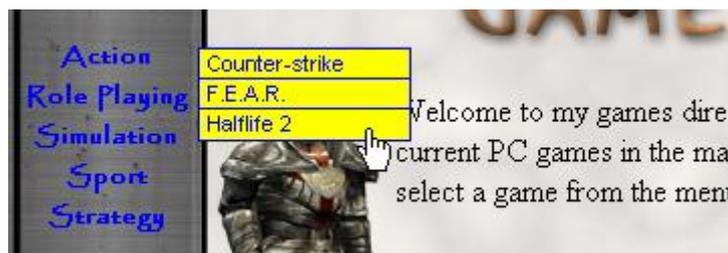
- 2) Give the new style the following attributes.

Category	Property	Attribute
Type	Font	Arial, Helvetica, sans-serif
	Size	11 pixels
	Color	Blue
Background	Background Color	Yellow
	Box	Padding-Left (not same for all)
Border	Style (same for all)	Solid
	Width (same for all)	1 pixel
	Color (same for all)	Blue
Extensions	Cursor	Pointer

- 3) When the new style class is created, add three new layers with the following properties.



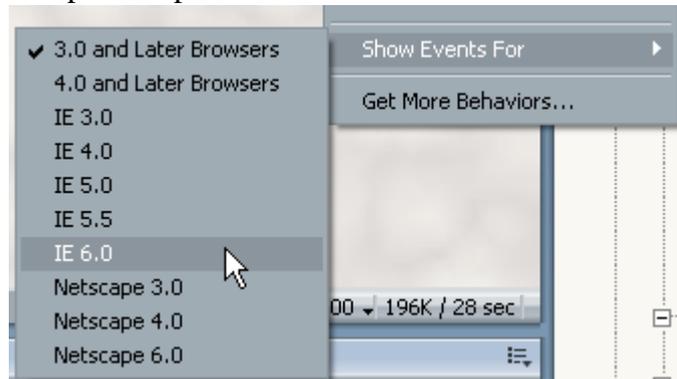
- 4) Add text to each layer so that they are looking like the example below.



Tip It helps to give your sub menus layer IDs that refer to the main menu item they relate to. In this case, the layers for the 'action' submenu all have 'act-' before their names.

Exercise 4 – Adding Menu Behaviours

- 1) Click on the border of the Action menu layer to select it.
- 2) Display the **Behaviours** panel from the **Window** menu or press **[Shift] [F4]** if it is not already showing.
- 3) Click the **Add Behaviour** button **+** along the top of the panel.
- 4) If a lot of the options in the menu are not available (in grey text) then it is probably because Dreamweaver is set to only show behaviours that will work in old browsers. If this is the case, then select **Show Events For** and then select a more recent web browser such as **IE6.0** or **Netscape 6.0**. More options will now be available in the Add Behaviour menu now.



First, we'll add a behaviour that will change the colour of the text when the mouse goes over the layer.

- 5) From the list of behaviours, choose **Change Property**.
- 6) Set the change property options to match the following and click **OK**.

Type of object:

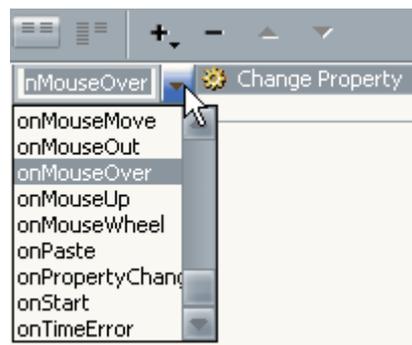
Named object:

Property: Select:

Enter:

New value:

- 7) Change the event to **onMouseOver**



Next, we'll create another behaviour that changes the text colour back to its original colour when the mouse moves off the layer.

8) With the *Action* layer still selected, add a new behaviour with the following settings.

Type of object: DIV

Named object: div "action"

Property: Select: style.color IE4

Enter:

New value: blue

9) Set the event to **onMouseOut**

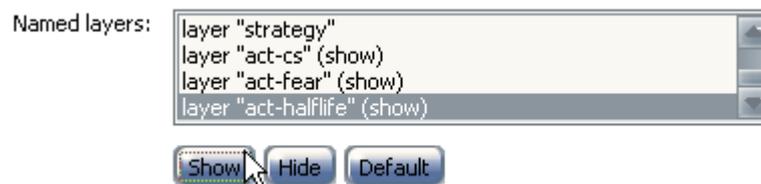


10) Save and preview the changes. Move your mouse over the layer. When the mouse goes on and off the layer, some JavaScript in the page changes the CSS colour property for the *Action* layer object. Next we'll add actions to display the sub menu layers when the mouse goes on the *Action* layer and hide the sub menu when the mouse goes off the *Action* layer.

11) With the *Action* layer still selected, click the **Add Behaviour** button.

12) This time, select **Show-Hide Layers** from the list of behaviours.

13) Scroll through the list of objects until you get to the ones for the *Action* sub menu. For each of the 'act' layers, select them and click the **Show** button so that they each have (**show**) next to them. Click **OK** when all three are set to show.



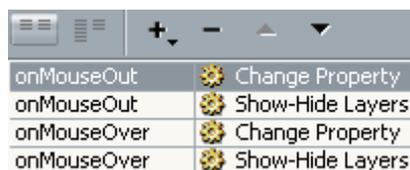
14) Set the event for the new action to **onMouseOver**.

15) Add a new **Show-Hide Layers** event.

16) This time, set each of the three sub menu layers to **Hide** and click **OK** when done.



17) Set the event to **onMouseOut**.



18) Save and preview the page so you can test the show-hide behaviours.

Late we'll change the options for the sub menu so that they are initially hidden when the page loads, but we need to set up the behaviours for those layers first. It's a pain to work with them when they're already hidden so we'll hide them last.

Exercise 5 – Setting Sub Menu Behaviours

Each of the sub menus will require quite a few behaviours. There will be two for changing the background colour when the mouse goes on and off. Another two for changing the text colour when the mouse goes on and off. Another two for showing and hiding the layers when the mouse goes on and off. Lastly, one which provides the link for when the layer is clicked. That gets pretty complex when you have to do it for each menu. In other words, pay close attention from now on!

- 1) Click on the border of the *Counter-strike* menu to select it.
- 2) Add a **Change Property** behaviour with the following options.

Type of object: DIV

Named object: div "act-cs"

Property: Select: style.backgroundColor IE4

Enter:

New value: navy

Note Don't forget to change the name of the object that the action refers to – it's a common mistake.

- 3) Set the event for the new behaviour to **onMouseOver**.
- 4) Add another **Change Property** behaviour with the following options.

Type of object: DIV

Named object: div "act-cs"

Property: Select: style.backgroundColor IE4

Enter:

New value: yellow

- 5) Set the event for the new behaviour to **onMouseOut**.
- 6) Add a **Change Property** behaviour with the following options.

Type of object: DIV

Named object: div "act-cs"

Property: Select: style.color IE4

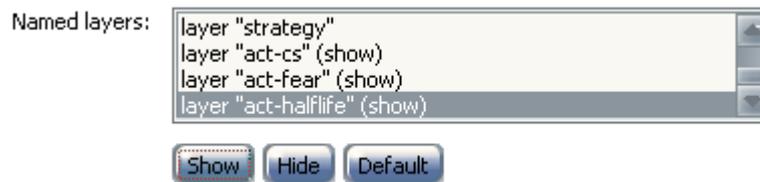
Enter:

New value: yellow

- 7) Set the event for the new behaviour to **onMouseOver**.
- 8) Add another similar behaviour that sets the **style.color** property for the act-cs object to *blue* (its original colour).
- 9) Set the event for this behaviour to **onMouseOut**.

If you preview the page now, you'll see the formatting for that layer change when you move your mouse over it. That just leaves the layer show-hide behaviours and of course the link.

- 10) Add a **Show-Hide Layers** behaviour with the *act-cs* layer still selected.
- 11) Set it to show each of the three sub menu layers and click **OK**.



- 12) Set the event for this behaviour to **onMouseOver**.
- 13) Add a new behaviour that hides the three sub menu layers and set its event to **onMouseOut**.

Now when you preview the page, the three sub menu layers will remain visible if the mouse goes over the *act-cs* layer, but will hide when the mouse goes off that layer (unless it goes on to the main *Action* layer which is also set to show the sub menu).

- 14) Add a new behaviour. This time it will be a **Go To URL** behaviour.
- 15) Set the URL for the link to go to as shown below and click **OK**.



- 16) Leave the event for the behaviour as **onClick**.
- 17) Now for the bad news, you will need to go through all of that again for both of the other sub menu layers (F.E.A.R. and Halflife 2). For each one, repeat the steps from step one, making sure you are selecting the appropriate layer in each step.

Note If you feel comfortable with Code View, you can do this a lot quicker by copying certain information from the layer you have already done and pasting it in the right place in the other layers, simply changing the layer name where appropriate. This isn't for the faint hearted though and you really need to be comfortable with editing the code to do it properly this way. With a bit of practice, you may find this is the best way to work rather than repeating the whole process for every single menu item.

```
<DIV ID="strategy" STYLE="position:absolute; width:90px; height:20px; z-index:3; left: 5px; top: 150px;
visibility: visible;" CLASS="main_menu">Strategy</DIV>
<DIV ID="act-cs" STYLE="position:absolute; width:100px; height:15px; z-index:5; left: 95px; top: 75px;" CLASS=
"sub_menu" onClick="MM_goToURL('parent','http://www.counter-strike.com/');return document.MM_returnValue"
onMouseOver=
"MM_changeProp('act-cs','','style.backgroundColor','navy','DIV');MM_changeProp('act-cs','','style.color','yello
w','DIV');MM_showHideLayers('act-cs','','show','act-fear','','show','act-halflife','','show')" onMouseOut=
"MM_changeProp('act-cs','','style.backgroundColor','yellow','DIV');MM_changeProp('act-cs','','style.color','blu
e','DIV');MM_showHideLayers('act-cs','','hide','act-fear','','hide','act-halflife','','hide')"> Counter-strike
</DIV>
<DIV ID="act-fear" STYLE="position:absolute; width:100px; height:15px; z-index:5; left: 95px; top: 90px;" CLASS
="sub_menu">F.E.A.R.</DIV>
<DIV ID="act-halflife" STYLE="position:absolute; width:100px; height:15px; z-index:5; top: 105px; left: 95px;"
CLASS="sub_menu">Halflife
2 </DIV>
```

- 18) For the *F.E.A.R.* layer, the URL it needs to go to when clicked is <http://whatisfear.com/>.
- 19) For the *Halflife 2* layer, the URL it needs to go to when clicked is <http://half-life2.com/>.

Exercise 6 – Hiding the Sub Menus

- 1) Select the *Counter-strike* layer by clicking its border.
- 2) Change the **Vis** property to *Hidden*. 
- 3) Do the same for the other two sub menu layers.
- 4) Preview the page again and test the menu.

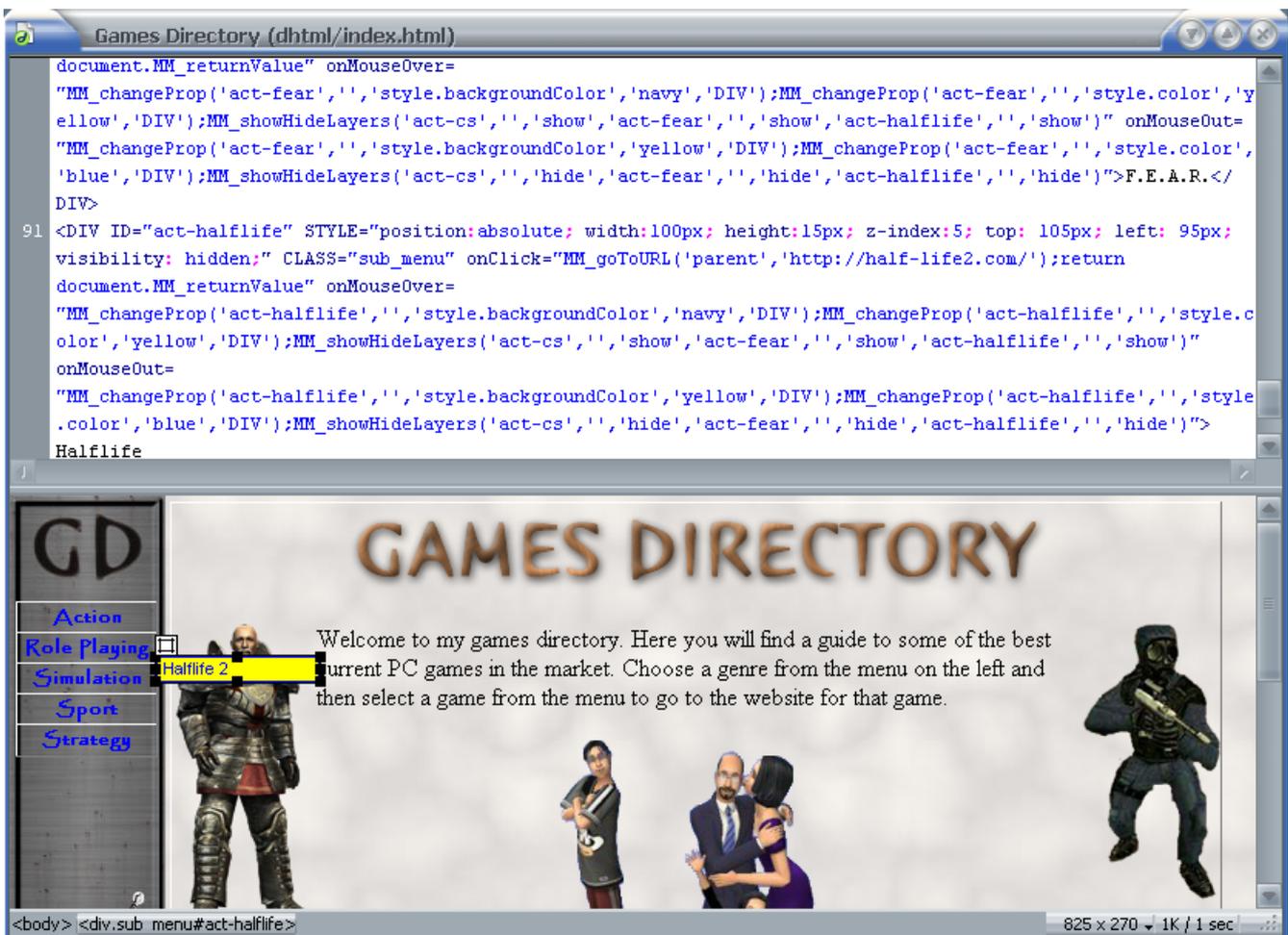
Should work fine now. Problem is, what do you do when you want to edit one of those layers that are now hidden. One solution is to select the layer in **Code View**. Then it will be visible when you return to **Design View**.

- 5) Switch to code view by clicking the button or by pressing **[Ctrl] [F]**.

Each of the layers uses a **DIV** tag and the first attribute in the tag will be the ID which shows the name of the tag. So if we want to find the layer named *act-fear*, that makes it easy enough to find.

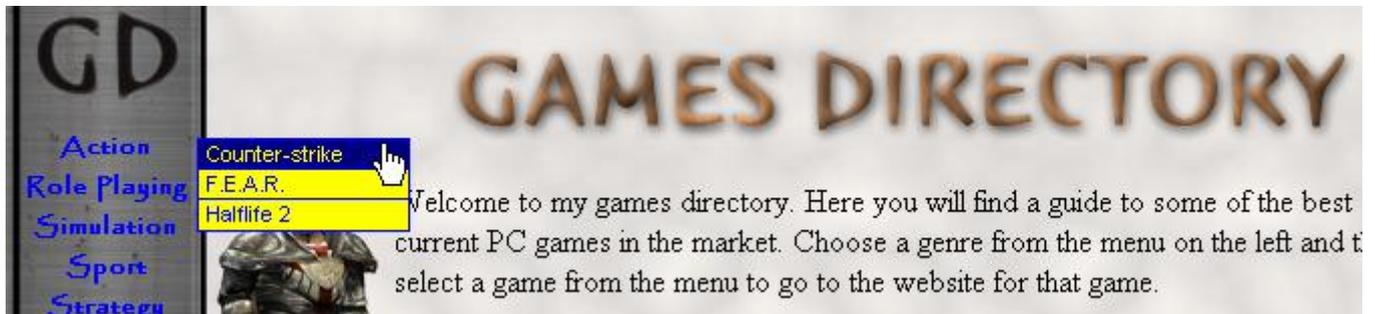
- 6) Look through the code until you get to a line beginning with **<DIV ID="act-fear"**.
- 7) Click anywhere in that line. You won't see any change just yet but...
- 8) Return to Design view by clicking the button or by pressing **[Ctrl] [F]**. The layer you just selected by clicking inside its HTML code will now be visible until you click elsewhere. While it's selected and visible, you could make any necessary changes.

Tip If you are switching back and forth between **Code View** and **Design View** a lot, you might be better off using the Split view so you can see them both at the same time.

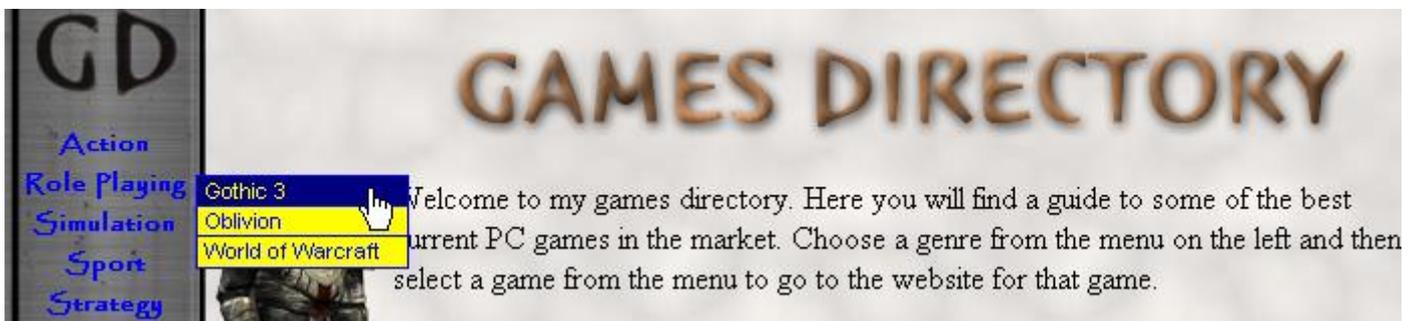


Exercise 7 – Finishing the Other Menus

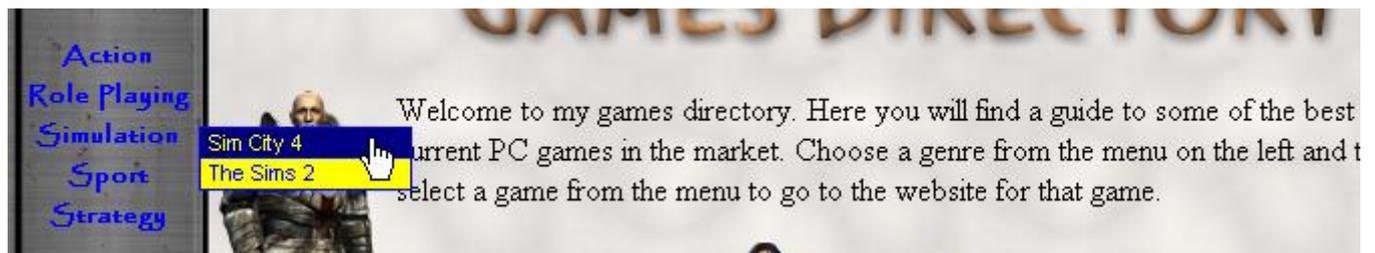
Your final task (and not a small one) is to complete the rest of the menus. The examples below show how each menu should look. Below each example is a list of the layer names and URLs for each sub menu. You'll need to use your own judgement to decide what other properties will be needed to position each layer in the right spot. Remember the show-hide actions and remember to hide each layer when you're done.



You should already have the first menu done.



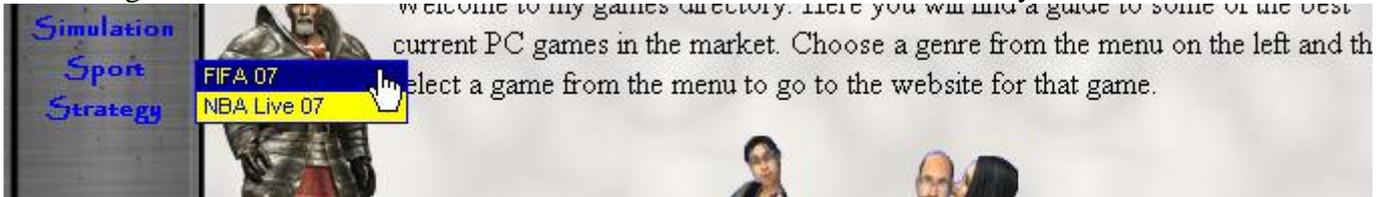
Layer Name	URL to go to
rpg-gothic3	http://www.gothic3.com/
rpg-oblivion	http://www.elderscrolls.com/
rpg-wow	http://www.worldofwarcraft.com/



Layer Name	URL to go to
sim-city	http://simcity.ea.com/
sim-sims2	http://www.thesims2.com/

Using Dreamweaver

Dynamic HTML Menus

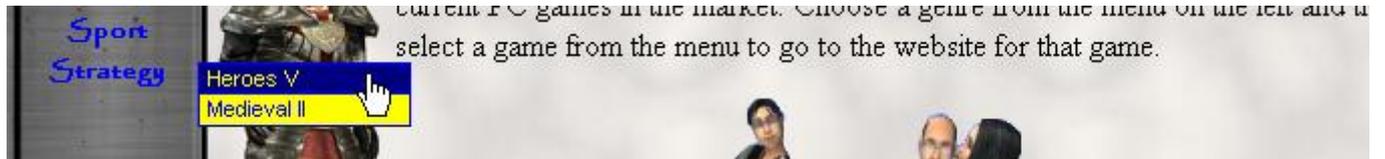


Layer Name

sport-fifa
sport-nba

URL to go to

<http://www.fifa07.ea.com/>
<http://www.easports.com/nbalive07/>



Layer Name

strat-heroes
strat-medieval

URL to go to

<http://www.mightandmagic.com/>
<http://www.totalwar.com/>

The completed site should look similar to the example below. Not the fanciest or most elaborate site, but it's served to show you how to put together DHTML menus. Not the easiest way to set up a website's navigation but it can be pretty effective. Is it worth the effort? That's for you to decide. Of course, normally, you'd use menus for users to navigate through your own site, not just links to other sites. This would mean that you'd need to create a template with the menus or copy them to each page in your site.

