

# 7 – Extra Techniques

# The Challenge

Now it's time to challenge the serious web developers among you. In this section we will create a website that will bring together skills learned in all of the previous exercises and adds a few new ones. In many sections, rather than tell you what to do, you will need to apply what you

have learned previously to accomplish a certain result. If you're not sure how to do anything in this section then you may have either skipped previous exercises or forgotten what you have learned in them. Either way, best to make sure your skills are up to scratch before you try to tackle what follows. For those who have easily done everything in the previous sections, here's a test of your skills.



# The Task

Your task in this section is to create a website for the Nuklear family. They want you to create a website to tell the world about them. Are you up to the task? Time to find out.

# Planning

Most good websites begin with some good planning and this one will be no different. The Nuklear family have thought long and hard about what they want in their website and have given you a rough **Site Map** as shown below.



Of course, Nick Nuklear knows a thing or two about computers so he whipped up a slightly fancier version as shown on the following page (A website's planning can be anything from rough sketches to formal illustrations but the important thing is that it needs to give you a clear idea of what the site will have.

**Note** Before you begin, make sure you have the images for these exercises saved in the location where you intend to create the Nuklear Family Website.



After deciding on the content of the site, the family decided on exactly how they wanted each page to look. A rough design plan for the home page is shown below. More detailed or neater plans for each page could be created and definitely would be for a professional site. Note the precision of the size and position for certain elements on the page. Standard old HTML isn't capable of that level of precision, but CSS is and since the majority of browsers now support CSS and its use is encouraged by the WC3, we will rely on it for our page formatting and layout. HTML for the structure of the page with CSS for the presentation (layout and formatting). Content and presentation will be handled separately as it should be.



**Note** The level of planning shown here should be considered to be a <u>minimum</u> amount for a good website. Most professional sites would require far more detailed plans.

Using Dreamweaver CC

### **Exercise 1 – Create the Site**

Assuming all the site planning is complete, we can now begin to create the site itself in dreamweaver.

 Create a new site, selecting a suitable location for the files in the site to be saved in. Make sure the exercise files for this website (a folder of images and one css file) are saved in the same location. Give the site a suitable name. If you can't remember how, the pictures below should give you some clues.

Files DOM Assets Snippets
Francie's Fabulous Flowerpots
Local Files 🕈
✓ ➡ Francie's Fabulous Flowerpots
Manage Sites
Import Site
Here you'll select the local folder and a name for your Dreamweaver site.
Site Name: Nuklear Family
Local Site Folder: C:\Users\Steve\Websites\Nuklear Family\
Files DOM Assets Snippets
🗎 Nuklear Family
Local Files 🕈
🗸 😑 🦳 Site - Nuklear Family (C: \Users \
> D pics
<pre>{} alternate styles.css</pre>

2) Once you have a site we will create a new HTML document that we will then use as the basis for our other pages (you could also create a template but we won't be creating many pages). Create a new document with the filename *index.html* and save it.

Local Files 🕈	
✓ ⊟ Site	e - Nuklear Family (C:\Users
> 🖿	pics
{}	alternate_styles.css
	index.html

3) Open the page and set the title the page to *Nuklear Family Website*.

Category	Title/Encoding		
Appearance (CSS)			
Appearance (HTML)	Title:	Nuklear Family Website	
Links (CSS)	Document Type (DTD):	HTML5	~
Headings (CSS)			
Title/Encoding	Encoding:	Unicode (UTF-8)	✓ (Reload)

## **Exercise 2 – Creating the page structure**

The layout of the page will be created by using HTML tags for each page section and then using CSS to position each section. First we'll create a container for the page.

1) From the Insert menu select Main and add *container* for the ID.

Insert: At insertio	n point 🗸	~	ОК
Class:	~		Help
ID: container	~		
New CSS	Rule		
	index.html* ×		

Content for id "container" Goes Here

We'll create a CSS rule to position this tag but first we need to create the CSS file.

2) Create a new CSS file called *nuklear\_styles.css*.

	CC Libraries Insert CSS Designer
	+ — Sources
	Create A New CSS File
	Attach Existing CSS File
	Define in Page
File/URL:	nuklear_styles.css Browse
	+ - Sources
	ALL SOURCES
	nuklear_styles.css

3) Create a new *#container* selector.

+	— Selectors
Q	Filter CSS Rules
#c	ontainer

Add the following CSS properties

	🖬 Layo	ut		
	width	:	<u>800</u> px	
	height		<u>100</u> %	
nar	gin	: 0px		
		<u>0</u> px		
<u>0</u> p:	×		<u>0</u> px	
		<u>0</u> px		

© Steve O'Neil 2017

http://www.oneil.com.au/pc/

### Advanced Techniques

position : absolute top: <u>0</u> px left: <u>100</u> px <u>bottom:</u> <u>0</u> px

Absolute positioning means that it will be positioned in a very specific way. I.e. 100 pixels from the left. 0 pixels gap on the top and bottom.

🖉 Backgrour	nd	
background- color	:	 #8596EE

4) Delete the placeholder text and you will be left with a blue box taking up the whole page with a 100 pixel wide gap on the left side. The HTML code and CSS code should look similar to the following examples

```
<!doctype html>
 1
 2 v <html>
 3 v <head>
4
    <meta charset="utf-8">
    <title>Nuklear Family Website</title>
 5
    k href="nuklear_styles.css" rel="stylesheet" type="text/css">
 6
 7
    </head>
8
9 v <body>
10 <main id="container"></main>
11 </body>
12
    </html>
13
                      @charset "utf-8";
                   1
                   2
                      #container {
                   3
                          margin: Opx;
                   4
                          width: 800px;
                   5
                          height: 100%;
                   6
                          background-color: #8596EE;
                   7
                          position: absolute;
                   8
                          top: 0px;
                   9
                          bottom: Opx;
                  10
                          left: 100px;
                  11
                      }
```

5) Insert a **Header** inside the container.

Insert Hea	der			×
Insert: Class:	After start of tag	<pre>~ <main id="conta&lt;/pre&gt;&lt;/th&gt;&lt;th&gt;iner"> ~</main></pre>	ОК Cancel	
ID:	header	~		Help

6) Insert a Navigation after the Header.

Using Dreamweaver CC

Insert Nav	igation			×
Insert:	After tag	✓ <header id="&lt;/p&gt;&lt;/th&gt;&lt;th&gt;'header"> ~</header>	ОК	
Class:		~		Help
ID:	navigation	~		

7) Insert a **Content** section after the navigation.

Insert Sect	ion			×
Insert:	After tag	∽ ⊂nav id="navig	gation"> 👻	ОК
Class:		~		Help
ID:	content	~		

8) Insert a Footer after the content

Insert Foot	ter		X
Insert:	After tag	<pre>&lt; <section <="" id="content" pre=""></section></pre>	> ~ OK Cancel
Class:	footer	 ~	Help

- 9) Delete the placeholder text in all except the Content section.
- 10) Create a *#header* selector and apply the following CSS rules using either the **CSS Designer** panel or by editing the CSS code.

```
#header {
    width: 800px;
    height: 40px;
    position: absolute;
    left: 0px;
    top: 0px;
    background-color: #8596EE;
}
```

11) Create a *#navigation* selector and apply the following CSS rules using either the **CSS Designer** panel or by editing the CSS code.

```
#navigation {
    width: 120px;
    position: absolute;
    top: 40px;
    bottom: 30px;
    left: 0px;
}
```

Using Dreamweaver CC

12) Create a *#content* selector and apply the following CSS rules using either the **CSS Designer** panel or by editing the CSS code.

```
#content {
    width: 680px;
    position: absolute;
    top: 40px;
    left: 120px;
    bottom: 30px;
    right: 0px;
    background-color: white;
}
```

13) Create a *#footer* selector and apply the following CSS rules using either the **CSS Designer** panel or by editing the CSS code.

```
#footer {
    width: 800px;
    height: 30px;
    position: absolute;
    bottom: 0px;
    left: 0px;
    text-align: center;
    font-size: smaller;
    background-color: #8596EE;
}
```

Lastly we'll add some smaller sections for the navigation items. Later we will add a sub-menu that will appear when a user moves their mouse over the family navigation button but for now we'll just add the main navigation menu items.

14) Click inside the **Navigation** section. From the **Insert** menu select **DIV**. The DIV tag is a generic container for page content which will be suitable for our menu items. Enter home for the name of the div

Insert:	After start of tag	~	<nav id="navigation"></nav>	~	OK Cancel
Class:		~			Help
ID:	home	~			

15) Add 4 more **DIV** sections as shown below and using the names indicated. Each one should be <u>after</u> the previous one.



### Advanced Techniques

The HTML for the page layout so far should look similar to the following example.

```
<body>
<main id="container">
<header id="header"></header>
<nav id="navigation">
<div id="navigation">
<div id="navigation">
<div id="home">Content for id "home" Goes Here</div>
<div id="town">Content for id "town" Goes Here</div>
<div id="town">Content for id "town" Goes Here</div>
<div id="family">Content for id "family" Goes Here</div>
<div id="friends">Content for id "friends" Goes Here</div>
<div id="links">Content for id "links" Goes Here</div>
<div id="links">Content for id "links" Goes Here</div>
<div id="links">Content for id "links" Goes Here</div>
</main>
```

Delete the placeholder text and add new CSS selectors with the following properties.

```
#home {
    width: 120px;
    height: 25px;
    position: absolute;
    left: 0px;
    top: 0px;
}
#town {
    width: 120px;
    height: 25px;
    position: absolute;
    left: 0px;
    top: 25px;
}
#family {
    width: 120px;
    height: 25px;
    position: absolute;
    left: 0px;
    top: 50px;
3
#friends {
    width: 120px;
    height: 25px;
    position: absolute;
    left: 0px;
    top: 75px;
}
#links {
    width: 120px;
    height: 25px;
    position: absolute;
    left: 0px;
    top: 100px;
}
```

# Exercise 3 – Adding Some Content

Time to add some content to our page.

1) In the heading section and insert the image *heading\_home.png*, adding appropriate **ALT** text for the image.



2) Replace the text in the footer layer with the row of text shown below.

Home - Town - Family - Friends - Links

3) Each of the words will be a link to a page that we will create later based on this page. Select each 'link' and place the appropriate filename in the Link property to create the links.

Home	index.html
Town	town.html
Family	family.html
Friends	friends.html
Links	links.html

Home - Town - Family - Friends - Links

Remember to save the file regularly.

4) In each of the DIV layers within the navigation layer, add one of the navigation images as appropriate (E.g. *nav\_home.gif* for Home), replacing the text in each layer.

Remember to add appropriate **ALT** text for each image. When you save and preview the page it should look similar to the following example.

Using Dreamwe	Advanced Techniques	
1849 1	Nuklear Family Website	
Home	Content for id "content" Goes Here	
Town		
Family		
Friends		
Links		
	<u>Home – Iown – Family – Friends – Links</u>	

# Exercise 4 – CSS Formatting

1) Create a new CSS selector for the *body* tag.

#home
body

2) Set the **Background** properties as shown

💹 Backgroun	nd	
background- color	: 💽 #0A224E	
background-in	mage	
url pics,	;/background.jpg 📋	
gradient	🔀 none	
background- position	: left top	
background- size	:	
background- clip	:	
background- repeat	: == == = =	
background- origin	:	
background- attachment	: fixed	

Using Dreamweaver CC

3) Set the **Box** settings so that there is no **Padding** or **Margin**.

```
body {
    background-color: #0A224E;
    background-image: url(pics/background.jpg);
    background-position: left top;
    background-repeat: no-repeat;
    background-attachment: fixed;
    margin: 0px;
    padding: 0px;
}
```

Earlier we applied a white background colours to content area. Traditionally, HTML only allowed for the use of **JPEG** and **GIF** files but recent browsers also recognise the use of **PNG** files for images which have become increasingly common. One advantage of a **PNG** file is that it allows for partial transparency. We will take advantage of this by setting the background of the content section to use a **PNG** file that is partially transparent so that the background of the page will faintly show through. The image we will use is 1 pixel in height and width so it will be a small file size as it fills the area.

4) Change the background property of the **Content** section's CSS rule to remove the background colour you set earlier and set *content\_background.png* as the background image.

🖉 Background	
background- color : 🔀 un	defined
background-image	
url pics/content_ba	ckgr 🛅
gradient 🔀 none	

This will create a background for the section that is easy to read text over but still shows the background of the page. This will become more obvious as we add content to that layer on each page that is created from this template.

- Tip Some older web browsers such as Internet Explorer 6 support PNG files but don't support their transparency properties. This problem may be solved by adding a simple script to your page. For details and instructions go to <a href="http://homepage.ntlworld.com/bobosola/">http://homepage.ntlworld.com/bobosola/</a>.
- 5) Change the background properties for the **Navigation** CSS rule so that it matches the following.

background-image				
url pics/nav_background 🛅 gradient 📿 none				
background- position background- size background-	:	left	top	
clip background- repeat	:	<b></b>	: ::::	

© Steve O'Neil 2017

### Advanced Techniques

6) Preview the page and it should look similar to the example below



**Tip** You could change the layout of the entire site now simply by editing the styles. For instance, if you wanted the menu to be across the top of the page, all you'd have to do would be to edit the positioning information for the styles. That will change the layout for the entire site. Some sites even have more than one CSS file so that the layout and format can be quickly changed simply by changing the CSS file with a script on the page. E.g. one set of styles for when the site is views on a desktop computer and another for when it is viewed on a phone.

### Exercise 5 – Modifying Formatting with CSS Rules

Time to establish some rules for formatting within our site.

1) Create some new style rules for the hyperlink compound classes as described below.

```
a:link {
    text-decoration: none;
    color: blue;
}
a:visited {
    text-decoration: none;
    color: purple;
}
a:active {
    text-decoration: none;
    color: red;
}
a:hover {
    text-decoration: none;
    color: blue;
    border-top: 1px solid red;
    border-bottom: 1px solid red;
}
```

- Edit the styles for the BODY tag so that the following style rules are added: font-family: "Comic Sans MS", Arial, sans-serif; font-size: 13px; color: navy;
- **Tip** Since the font families here won't be in the list of Dreamweaver's list of suggested fonts, you can either use the Manage Fonts option at the bottom of the font list or you could simply edit the code in the CSS file.

Advanced Techniques

### Exercise 6 – Adding a Menu

1) Select the Home image in the navigation area.



2) Set the **ID** property of the image to *home\_image*.

Properties			
Image, 1K	Src	pics/nav_home.gif	7
ID 10me_image	Link		Ī

- 3) Add appropriate IDs for each of the other images.
- 4) Select the home div which the image is inside of.
- **Tip** Remember that you can select the image and then use the bar along the bottom of the editing area to select elements of the page that are around that image.

body	main #container	nav #navigation		#home	img #home_image
Properties					
- R C	SS-P Element	0nv W 120nv	7-Index	<div< th=""><th>/id="home"&gt;</th></div<>	/id="home">

5) Display the **Behaviours** panel.



6) Add a **swap image** behaviour so that *nav\_home.gif* is swapped with *nav\_home\_act.gif* on mouse over.

Swap Image	×
_	ОК
Images:	image "home_image" in layer "home" in layer "navigati
	image "town_image" in layer "town" in layer "navigatio image "family_image" in layer "family" in layer "navigat image "friends_image" in layer "friends" in layer "navi
Set source to:	pics/nav_home_act.gif Browse
	Preload images

7) Add a Swap Image Restore behaviour.

Swap Image	
Swap Image Restore	

- 8) Add similar behaviours for the other navigation layers (remember to add a **Swap Image Restore** behaviour for each one as well. Also remember to make sure you have the DIV around the image selected and not the image itself when you add the behaviours).
- 9) Add a **Go to URL** behaviour for the home **div** tag so that it links to *index.html*. Make sure the behaviour is set to **onClick**.

Using Dream	Advanced Techniques		
Go To l	URL		×
			ОК
Oper	n in: Main wi	ndow*	Cancel
			Help
U	JRL: index.ht	ml Browse	
		Tag <div></div>	
		onMouseOver W Swap Image Restore	

- 10) Add similar behaviours for the other navigation DIVs. Remember we haven't saved the other pages yet but the file names will be **town.html**, **family.html**, **friends.html** and **links.html**.
- 11) Create new DIV sections within the <u>content</u> layer for *Winston*, *Elizabeth*, *Lizzy*, *Nick* and *Pets* as shown below. Give each layer a suitable ID name.

Insert Div			×
Insert:	After start of tag	<pre> <section id="content">  </section></pre>	ОК
Class:	winston		Cancel
ID:	New CSS Rule		Help
Insert Div			X
Insert:	After tag	<pre> <div id="winston"> </div></pre>	OK
Class:	elizabeth		Cancel
ID:	New CSS Rule		Help

12) Change the placeholder text to become each family member's name.

1809	Nuklear Family Website
Home	Winston
Town	Elizabeth
	Lizzy
Family	Nick
Friends	Pets Contact for id Verstant Cons Lland
Links	content for to content Goes Here

### Advanced Techniques

13) Create new CSS rules for each of the new DIVs as shown below.

<pre>#winston background-color: #8798F0; border: 1px solid #000033; position: absolute; visibility: visible; height: 18px; width: 90px; left: 0px; top: 55px; text-align: center;</pre>	<pre>#elizabeth background-color: #8798F0; border: 1px solid #000033; position: absolute; visibility: visible; height: 18px; width: 90px; left: 0px; top: 73px; text-align: center;</pre>
<pre>#nick background-color: #8798F0; border: 1px solid #000033; position: absolute; visibility: visible; height: 18px; width: 90px; left: 0px; top: 109px; text-align: center;</pre>	<pre>#pets background-color: #8798F0; border: 1px solid #000033; position: absolute; visibility: visible; height: 18px; width: 90px; left: 0px; top: 127px; text-align: center;</pre>

#lizzy
background-color: #8798F0;
border: 1px solid #000033;
position: absolute;
visibility: visible;
height: 18px;
width: 90px;
left: 0px;
top: 91px;
text-align: center;

14) Preview the page in a web browsers. Each of the **DIV** items should be positioned and formatted so that it looks like a sub-menu under the family button.

1800	Nuklear Family Website
Home	Content for id "content" Goes Here
Town	
Family	Winston
Friends	Elizabeth
Links	Lizzy Nick
a eA	Pets

a) For each of the **DIV**s in the menu, create a **Go to URL** action so that when they are clicked, they will go to pages that will be created later. Make sure the actions is set to onClick. The files that each should go to are *winston.html*, *elizabeth.html*, *lizzy.html*, *nick.html* and *pets.html*.

Tag <div></div>	
🔳 🗄 +,	
onClick	Go To URL

### Advanced Techniques

15) For each of the menu layers, create a new Change Property behaviour so that the background colour changes to yellow. It should be set to change in the Behaviours panel onMouseOver. Make sure the Element ID is set to the layer you want to change before you click OK.

Change Property					X
Type of element: Element ID: Property:	DIV div "winston O Select:	" background	~ ) Color	~	OK Cancel Help
New value:	O Enter:				
	0	nClick	Go To URL		

16) Create another **Change Property** behaviour for each layer so that the background colour changes back to its original colour of **#8798F0**. This one will activate **onMouseOut**.

Chang	ge Property						×
Тур	be of element: Element ID: Property: New value:	DIV div "winst O Select: Enter: #8798F0	on" backgrour		~	~	ОК Cancel Help
			onClick onMouseOut onMouseOver	<ul> <li>Go To URL</li> <li>Change Property</li> <li>Change Property</li> </ul>			

If you need to edit any of these behaviours (E.g. if you find you had the wrong **Element ID** or **Property** selected for one of them), you can double click a behaviour in the behaviours panel to edit it.

**Tip** You may want to add a style rule for each of the menu and navigation **DIV**s so that the mouse will change to a hand shape when it goes over the layer. In the CSS Designer panel, you can select the More category and then set the **Cursor** option to *pointer*.

+	<ul> <li>Propertie</li> </ul>	:5	
	T 🗆 🖾	Ţ.	Show Set
	Layout	More	
	More		
	cursor	: pointer	

17) Preview the page in a browser. When you move your mouse over one of the family menu options the background should change to yellow. When you move your mouse off it will change back to the original colour.

Now to finish the menu, we need to add some behaviours that will make the sub-menu appear and disappear at the right time which can be a bit tricky. We create behaviours so that the menu will show when the mouse is over the Family layer or any of the menu layers. We will also need to add some behaviours so that the menu layers are hidden when the mouse goes off them.

A behaviour will be added to the **Family DIV** that will set all of the menu layers to Show when the mouse is over the Family layer. Another behaviour will be added that will set the menu to hide when the mouse goes off the Family layer. We will then add behaviours to each of the menu layers to ensure they are still set to show when the mouse is over one of them.

18) Select the Family DIV (if you click on the border of the rectangle it should select it).

Town		
+++ Family	Winston	
Friends	Elizabeth	
Linke	LIZZY	
body main #container nav #navigat	tion div #family	img #family_image
	47	

19) Add a **Show-hide Elements** behaviour that will show each of the layers in the menu – **Winston**, **Elizabeth**, **Lizzy**, **Nick** and **Pets** (later we will change their properties so that they are hidden to start with). Make sure the behaviour is set to **onMouseOver**.

Show-Hide E	lements	×
Elements:	div "winston" (show) div "elizabeth" (show) div "lizzv" (show) Show Hide Default	OK Cancel Help

20) Add another behaviour for the Family DIV that will all of the menu layers to *hide* **onMouseOut**.

Show-Hide E	lements			X
Elements: div "winston" (hide) div "elizabeth" (hide) div "lizzv" (hide) Show Hide Default			OK Cancel Help	
		onClick	🛱 Go To URL	
		onMouseOut	Show-Hide Elements	
		onMouseOut	🗘 Swap Image Restore	
		onMouseOver	🌣 Swap Image	
		onMouseOver	Show-Hide Elements	

- 21) Select the **Winston** DIV. Add a **Show-hide Elements** behaviour that sets each of the family menu layers to **Show** when the mouse is over the Winston layer. This will prevent those layers from vanishing when the user moves their mouse off the Family DIV and on to the menu.
- 22) Add another Show-hide Elements behaviour that will hide the menu layers onMouseOut.

- **Tip** While you are adding these behaviours, check that none of the ones you had already created get changed back to **onFocus**.
- 23) Repeat steps 21 and 22 for each of the other four menu DIVs. This will make sure that they stay visible as long as the mouse is over one of them.
- 24) The last thing you need to do for the menu is to change the style properties for each of the menu layers so that they will be hidden when the page loads. Now they will only appear when the mouse goes over the Family navigation button.

+ - Selectors					
₽ Filter CSS Rules					
a:hover					
#winston					
#elizabeth					
visibility	: hidden				

25) Save and preview your page to test the menu (remember that the links go to files that haven't been created yet but we're about to change that).



By now the Code for your page will be looking a lot more complex as the **HTML** is being combined with **javascript** for the behaviours.

```
<body
onLoad="MM_preloadImages('pics/nav_home_act.gif','pics/nav_town_act.gif','pics/nav_famil
y_act.gif','pics/nav_friends_act.gif','pics/nav_links_act.gif')">
 <main id="container">
  <header id="header"><img src="pics/heading_home.png" width="460" height="40"</pre>
  alt="Nuklear Family Website"/></header>
  <nav id="navigation">
    <div id="home" onClick="MM_goToURL('parent','index.html');return</pre>
    document.MM_returnValue"
    onMouseOver="MM_swapImage('home_image','','pics/nav_home_act.gif',1)"
    onMouseOut="MM_swapImgRestore()"><img src="pics/nav_home.gif" alt="Home" width="120"
    height="25" id="home_image"/></div>
    <div id="town" onClick="MM_goToURL('parent','town.html');return</pre>
    document.MM_returnValue"
    onMouseOver="MM_swapImage('town_image','','pics/nav_town_act.gif',1)"
    onMouseOut="MM_swapImgRestore()"><img src="pics/nav_town.gif" alt="About our town"
    width="120" height="25" id="town_image"/></div>
```

### **Exercise 7 – Adding more Pages**

- 1) Use the **Save as** option from the file menu (**Ctrl Shift S**) to save a copy of this page as *town.html*.
- 2) Change the **Document Title** of the page to *Nuklear Family's Town*.

Category	Title/Encoding		
Appearance (CSS)			
Appearance (HTML)	Title:	Nuklear Family's Town	
Links (CSS)	Document Type (DTD):	HTML5	~
Headings (CSS)			
Title/Encoding	Encoding:	Unicode (UTF-8)	<ul> <li>Reload</li> </ul>

3) Change the image in the header to *heading\_town.png* and change the **ALT** text for the image.



**Tip** You can use the Point to file to change a file for an image just as you can use it to link to files.

<u>S</u> rc	pics/heading_town.png	G		Class
Link		€	<del>ريجـ_</del> Point to	File

- 4) Make copies of this page to create *family.html*, *friends.html* and *links.html*.
- 5) Change the document title and header image in each.
- 6) Open the *family.html* page and save a copy of it as *winston.html*.
- 7) Change the page title to Nuklear Family Winston.
- 8) In the content section, replace the existing text with *Winston* and make it a level 1 heading..
- 9) Make copies of this file for *elizabeth.html*, *lizzy.html*, *nick.html* and *pets.html*. Change the page title and content in each one.
- 10) Preview one of the pages in your browser. Check the links in the navigation menu and in the footer to make sure they load the right pages.
- 11) Add some content to the main page as shown below.

# Using Dreamweaver CC Advanced Techniques Home Welcome to the website of the Nuklear family. We hope so will enjoy browsing through our site and finding out about our family. Family Friends Links Iniks

Image filename - main\_family.png

12) Add the following content to the Town page.

	Nuklear ranniy 5 town
ne	Our house is situated in the scenic town of Pleasntview which is only a short drive from
vn 👘	Sim City. The family home is a 2 bedroom, 2 bathroom two story house with a study, 2
lly	car garage and pool. Take a look at the pictures below.
ıds	

- 13) Add the image *town1.jpg* below the text as shown below along with the text next to the image. You will need to set the image to align left so that the text can wrap around it to the right.
- **Tip** If you want a bit of space between the image and the text you can add some horizontal space by adding a hspace attribute to the img tag in the html code as shown.

```
<img src="pics/town1.jpg" alt="" width="300" height="225" align="left" hspace="5"/>Picture 1
```



14) Give the image the **ID** *town\_photo*.

Properties		
Image, 23K	<u>S</u> rc	pics/town1.jpg
ID town_photo	<u>L</u> ink	

For the next part we're going to add some HTML so switch to your code view (Ctrl ~).

15) Edit the HTML to add anchor tags to the text next to the picture. We won't add any filenames to link to though. The anchors will be used in a behaviour.

<img <br="" alt="Our&lt;br&gt;Town" name="town_photo" src="pics/town1.jpg" width="300"/> height="225" id="town_photo" align="left" hspace="5" /> Picture 1 Picture 2 Picture 3 Picture 4 Picture 5	→	<img <br="" alt="Our&lt;br&gt;Town" name="town_photo" src="pics/town1.jpg" width="300"/> height="225" id="town_photo" align="left" hspace="5" /> <a id="pic1">Picture 1</a> <a id="pic2">Picture 2</a> <a id="pic3">Picture 3</a> <a id="pic4">Picture 4</a>
Picture 5Picture 6		<a id="pic5">Picture 5</a> <a id="pic6">Picture 6</a>

Advanced Techniques

16) Select the last bit of text (Picture 6) and create a **Swap Image** behaviour that changes the picture to *town6.jpg*. Untick the **Restore images onMouseOut** option. Set it to **onclick** and instead of mouseOver since ee want it to stay on that image until another link is clicked.

Swap Image	×
Images:	image "town_image" in layer "town" in layer "navigatio image "family_image" in layer "family" in layer "navigat
	image "links_image" in layer "links" in layer "navigatio image "lown_photo" in layer "content" in layer "contain
Set source to:	pics/town6.jpg     Browse       Preload images     Restore images onMouseOut
	Tag <a> ■■ ■ + - ▲ ▼</a>

17) Add similar behaviours for each of the other links so that when each link is clicked, a different picture will be displayed. Save and preview to see the results.

onClick



🌣 Swap Image



To avoid the text and pictures appearing squashed up next to the navigation area, you may want to add a style rule for your **#content** selector so that there is a bit of padding. If you do that you will also need to adjust the width to allow for the extra padding.

18) Change the **#content** selector to add some padding.

+ - Selectors	KILLIIII <u>U</u> PX IIIIIII
₽ Filter CSS Rules	padding : 5px
#container	5 px 间
#header	
#navigation	<u>5 px</u> <u>6</u> px
#content	
#footer	<u>5</u> px

Advanced Techniques

19) Adding padding has made the whole content section wider so we'll have to adjust the width to compensate. Remember that padding is added on both sides so if you added 5px padding you'll need to reduce your width by 10px



20) Edit the *family.html* page so that it looks like the following. Each family member's name should be a link to their page and should be in an unordered (bulleted) list. The images to use are *icon\_winston.png*, *icon\_elizabeth.png*, *icon\_lizzy.png*, *icon\_nick.png* and *icon\_pets.png*.



21) Modify the file *winston.html* so it looks like the following. You will need to create a style rule for H1 to make it look like the heading in the example and you will also need to change the style rule for paragraphs (p selector) so that text alignment is justified. Of course changing the style rule to make paragraphs justified aligned might mean you'll need an inline style to make the paragraph with the picture centre aligned.



22) Edit the page *elizabeth.html* so that it appears like the following example.



23) Modify *lizzy.html* so that it looks like the following.



### Using Dreamweaver CC

24) Modify *nick.html* so that it looks like the following.



25) Modify *pets.html* so that it looks like the following.



### Using Dreamweaver CC

26) Edit the friends page to looks similar to the following example including the bold (strong) formatting on the names. Add names and descriptions for the images (a 12x3 table might be useful for layout). You could add a style rule for your table cells (**td** selector) that sets their vertical alignment to top.



A	few of o	ur family friends	*
a	Bianca	Bianca is an accountant in a large firm and is an old high school friend of Elizabeth.	
	Lara	Elizabeth's good friend Lara likes to travel the world. When she comes to visit she always finds time to challenge nick to a round of computer games.	m
	Clarice & Amy	Amy is one of Lizzy's school friends. Her mum Clarice often comes around to spend time with Elizabeth while the girls are at school.	
	Frank	Frank is Nick's pen pal from the distant country of Azeroth. His hobbies are travelling the countryside finding new forms of wildlife.	
*	Gary, Tom and Bill Simpson	Tom lives with his two sons Gary and Bill. Since the death of his wife he has worked hard in his job as an army officer to put the boys through school.	
	Harry Coulston	Self confessed sci-fi geek Harry is a good friend of Winston. They have worked together on several important science projects over the years.	-



27) Preview the page in a browser.

You may notice that since this is a long page, when you scroll down, the navigation and page headings scroll out of sight. We'll make a little change so that only the content section scrolls while the header, navigation and footer stay in place.

28) Select the #*content* selector in the CSS Designer panel.29) Find the overflow-y (vertical) property and change it to auto.

overflow-y : auto

Now when there is too much content to fit in the content section's available space there will be a scrollbar down the side. It won't appear on pages where everything already fits.

### Advanced Techniques

30) Edit the links page to look like the following and add some of your own favourite links in the bullet points.



### Exercise 8 – Changing Style Sheets

Now to really demonstrate the power of styles, let's try using a different version of our stylesheet that has different attributes for many of our style rules.

- 1) Open any one of the pages in your site (they all currently use the same stylesheet).
- 2) From the exercise files, make sure that the file alternate\_styles.css is saved in the same folder as your main website.
- 3) In your HTML code, locate the line near the top that links to the stylesheet.

k href="nuklear\_styles.css" rel="stylesheet" type="text/css">

4) Change it so that it refers to the alternate version of the stylesheet.

### <link href="alternate\_styles.css" rel="stylesheet" type="text/css">

- 5) Save and preview the page to see the differences. The only change we've made is referring to a different stylesheet but that alone has made significant changes to the formatting and layout of the page.
- Tip When you are creating / linking a CSS file to your Dreamweaver document, you can choose the conditional usage option. This allows you to have more than one CSS file that the browser uses based on certain conditions. E.g. you can specify a condition that tells a browser to use a different CSS file that is designed for smaller screens if the screen size is smaller than a certain amount.

<u>F</u> ile/URL:	[	Browse
Add as:	O⊥ink ○Import	
> Conditional Us	age (Optional)	