



Using Dreamweaver

6 – Styles in Websites

So far these exercises have deliberately avoided using **HTML's** formatting options such as the **FONT** tag. This is because the basic formatting available in **HTML** has been made largely redundant with the addition of **Cascading Stylesheets (CSS)**.¹ The following points outline some of the advantages provided by using **CSS** in websites.



- More precise control than HTML alone can provide over layout, fonts, colours, backgrounds, and other typographical effects.
- A way to update the appearance and formatting of an unlimited number of pages by changing just one document.
- Increased compatibility across browsers and platforms.
- Less code, smaller pages, and faster downloads.

Recent versions of Dreamweaver contain extensive support for CSS (though this support can be limited by changing certain Dreamweaver settings). For instance, when you format text or paragraphs in Dreamweaver, in many cases, rather than use **HTML** for the formatting, Dreamweaver will use the more precise formatting made possible by **CSS**.

A document in a website can be formatted by **CSS** on several ways.

1. Linked or Imported Stylesheets

Using this method, an extra file with a **.css** extension will contain all of the formatting information for your website. Then each file within your website will either link to that file or import the file. Both methods have similar results, except that using the import technique, more than one **CSS** file may be used in a single document. For example, a **CSS** file might include a style rule that specifies that level one headings will be centre aligned with large, bold navy blue text. Every **HTML** file that links to or imports that **CSS** file will have that rule applied to its level one headings so that the formatting for an entire website can be controlled from a single **CSS** file.

2. Embedded Styles

Using this technique, an **HTML** file will have a section near the top of the document that will specify style rules that will only affect that document. If the document also links to an external **CSS** file, then the embedded style block will have priority. For example, if an **HTML** document links to a **CSS** file that says headings will be blue, but the document also has an embedded style block saying that headings will be green, then the headings for this document will be green.

3. Inline Styles

In cases where a certain style is only needed for one part of a document, then an inline style will be used. This adds some style information to a particular **HTML** tag which will only affect that tag. It will also overrule any embedded or imported styles. Dreamweaver will often use embedded styles when you format some selected text in an **HTML** document.

¹ For more information about **CSS**, refer to the W3C specification at <http://www.w3.org/TR/CSS21/> or work through the **CSS** exercises at <http://oneil.com.au/pc/html/HTML-8-Formatting-with-Style.pdf>

Style Rules

Each CSS file, embedded style block or inline style defines its styles by making use of **Style Rules**. A Style rule consists of 2 parts. The **Selector** determines what the style rule will actually be referring to while the **Declaration** contains the style information for that selector. Look at the examples below.

```
H1 {color:navy}
.news {color:red}
```

In the first example, the selector (H1) specifies that the style rule refers to any H1 tag in the document(s) that are affected. The declaration (color:navy) specifies that any part of the document matching the style selector will be formatted in navy blue.

In the second example, the selector uses a class called *news*. This means that any part of the documents using this style rule that refer to the *news* class will be formatted in the colour red.

When there are several selectors that will have the same declaration, they may be grouped together. For example, if your level 1, 2 and 3 headings will all be formatted with a navy blue colour, then instead of having a separate style rule for each one they can be grouped in to a single rule as shown below.

```
H1, H2, H3 {color: navy}
```

Declarations can also be grouped together. For example, if your headings are going to be bold, navy blue, 18 pixels high and formatted in Arial font, you can group the declarations in to a single rule as shown below.

```
H1 {color:#000080; font:bold 18px Arial}
```

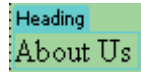
Using Dreamweaver, you can create CSS files, embedded style blocks and inline styles to take advantage of the CSS capabilities as we will do in the following exercises.

Note Although you can edit CSS files in Dreamweaver, you may find an application dedicated to this purpose is useful such as Topstyle - <http://www.topstyle4.com/>.

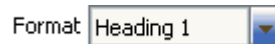
Exercise 1 – Creating a CSS File

In this exercise, we will be creating a CSS file to add some formatting to our website which is currently looking a bit plain.

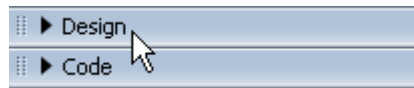
- 1) Make sure the *Francies Flowepots* site is open in Dreamweaver. Open any one of the HTML documents within the site such as *about.html*.
- 2) Click in the heading of the page.



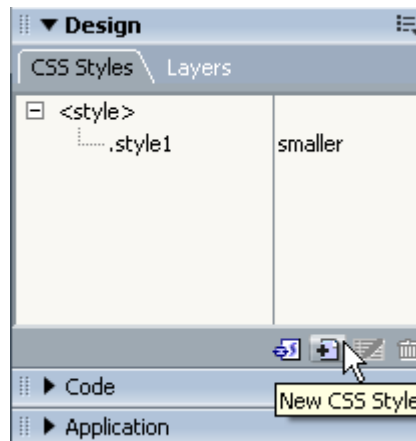
- 3) From the properties panel, set the **Format** to *Heading 1* (if you didn't already do this earlier)



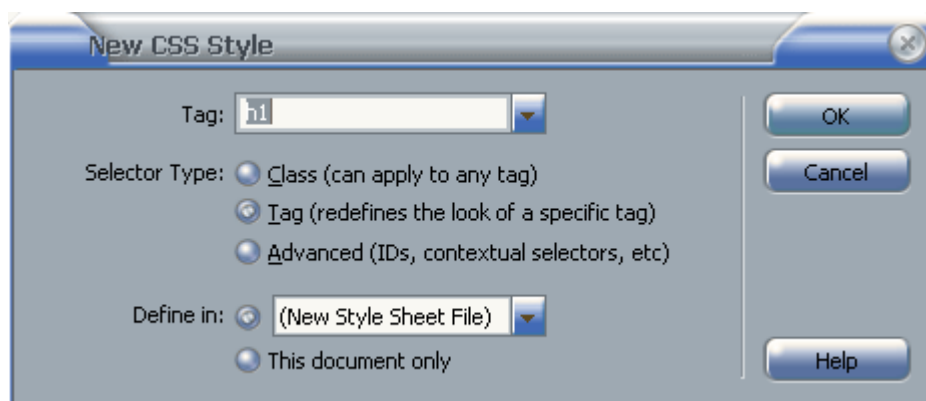
- 4) Display the **Design** panel on the right by clicking its heading or pressing **[Shift] [F11]**.



- 5) Click the **New CSS Style** icon  at the bottom of the **CSS Styles** panel.

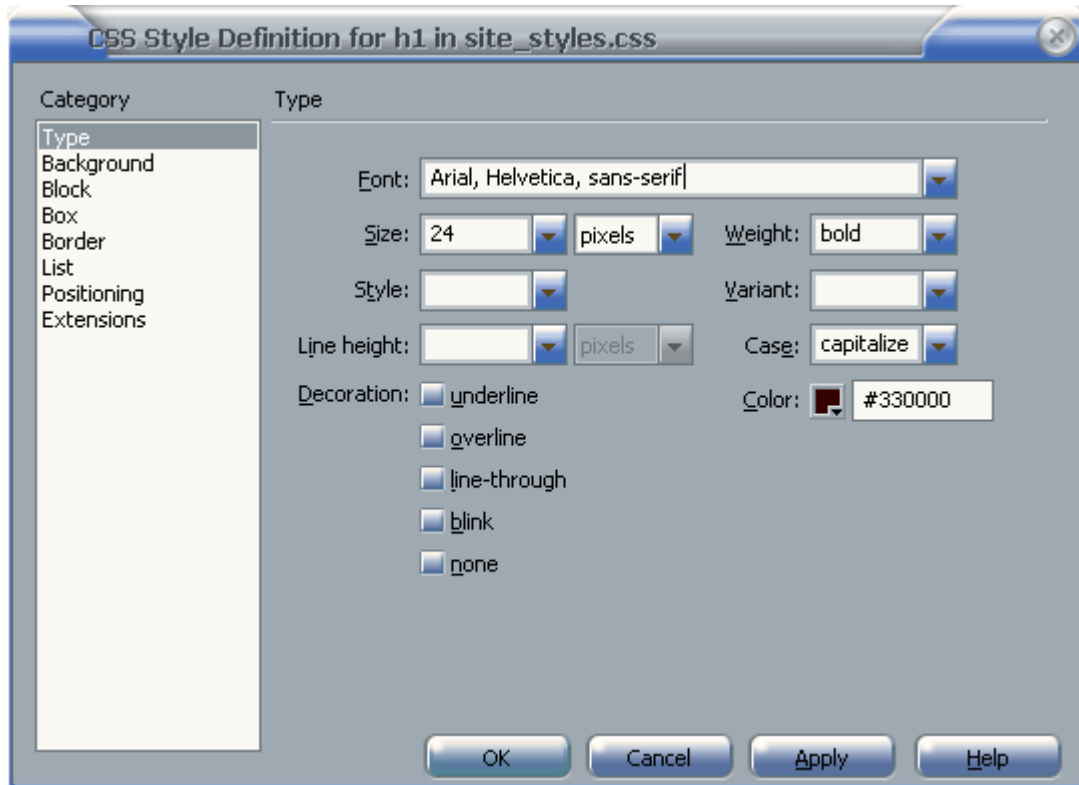


- 6) For the **Selector Type**, choose **Tag**.
- 7) Select *h1* from the list of tags.
- 8) Make sure **Define in** is set to *New Style Sheet File*.
- 9) Check that the options look like the example below and click **OK**.



- 10) Next we need to give a filename to the CSS file that is being created. Make sure the save location is the same as the HTML files in your website and enter *site_styles* as the filename (.css will be added to the filename). Click Save to create the CSS file. It will appear in your Files list.

- 11) Next, you will see the styles dialog. This allows us to create the style declaration for the *H1* tags. Make sure **Type** is selected from the **Categories** on the left and set the options as shown below.

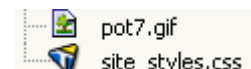


This will make level 1 headings use Arial font (if Arial isn't available on the computer being used to view your site, the alternate fonts will be used instead). The headings will be 24 pixels high, in bold text, the first letter of each word will be a capital and the colour will be a dark maroon.

- 12) Select **Block** from the list of categories on the left.
 13) Set Text-align to center.

Text align:

- 14) Click **OK** to complete the creation of the H1 style rule. You will see your stylesheet in the Files list.




The next step is to link our HTML files to the new CSS file. The HTML file you have open will already be linked so you will see the affect of the style rule on the heading.

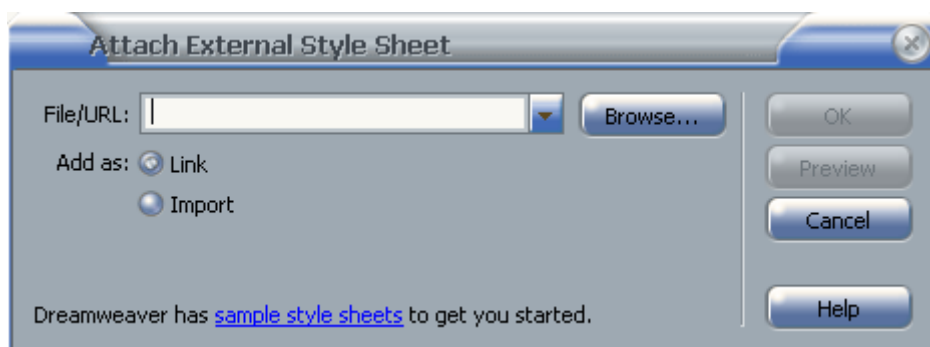
About Us

- 15) Open one of your other HTML files.

- 16) Make sure the heading is set to be formatted as Heading 1.

Format

- 17) In the CSS Styles panel, click the Attach Style Sheet icon  at the bottom of the panel.



- 18) Make sure **Link** is selected and click the **Browse** button.
- 19) Select the file *site_styles.css* and click **OK**.
- 20) Click **OK** again to close the **Attach External Style Sheet** dialog. The heading will now be updated with the *H1* rule from our *CSS* file.
- 21) Repeats steps 15-20 for each of your other HTML files so that each is linked to the new CSS file.
- 22) Make sure all the files are saved (**File** menu and then **Save All**).

Now that the files are all linked to *site_styles.css*, any changes we make to the style rules in there can be ‘cascaded’ to each of the HTML files.

- 23) With one of the HTML documents open, look at the **Code View [Ctrl] [V]**. Near the top will be a line like the following which links to your CSS file.

```
<LINK HREF="site_styles.css" REL="stylesheet" TYPE="text/css">
```

- 24) Return to **Design View**.

- 25) From your Files list on the side of the screen, double click on *site_styles.css* to open it. The file will contain one style rule with the H1 selector and several declarations grouped together as shown below.

```
hl {  
    font-family: Arial, Helvetica, sans-serif;  
    font-size: 24px;  
    font-weight: bold;  
    text-transform: capitalize;  
    color: #330000;  
    text-align: center;  
}
```

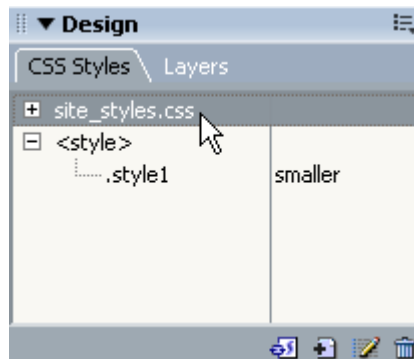
Note A style rule like the one shown above could be written in a more efficient abbreviated format like the example below. A CSS editor like Topstyle can do this task for you.

```
hl{COLOR:#330000;FONT:bold 24px Arial,Helvetica,sans-serif;TEXT-TRANSFORM:capitalize;TEXT-ALIGN:center}
```

Exercise 2 – Additional Style Selectors

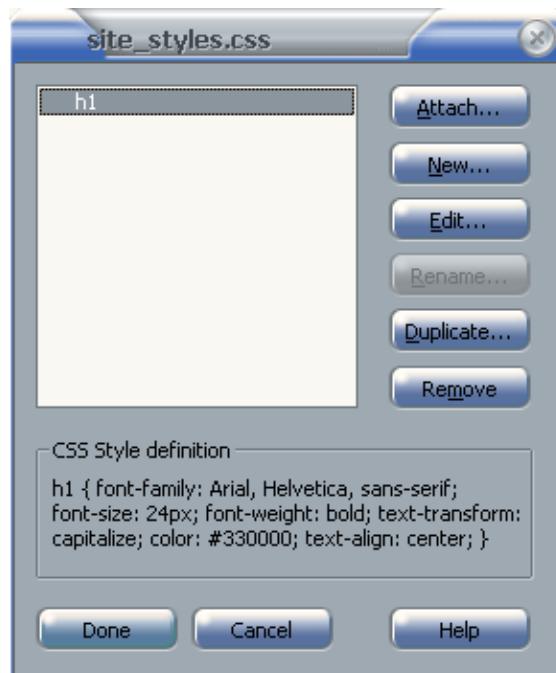
Make sure one of your HTML files is open. It doesn't matter which one, as long as it is linked to your CSS file.

- 1) Click on the *site_style.css* file within the **CSS Styles** panel to select it.

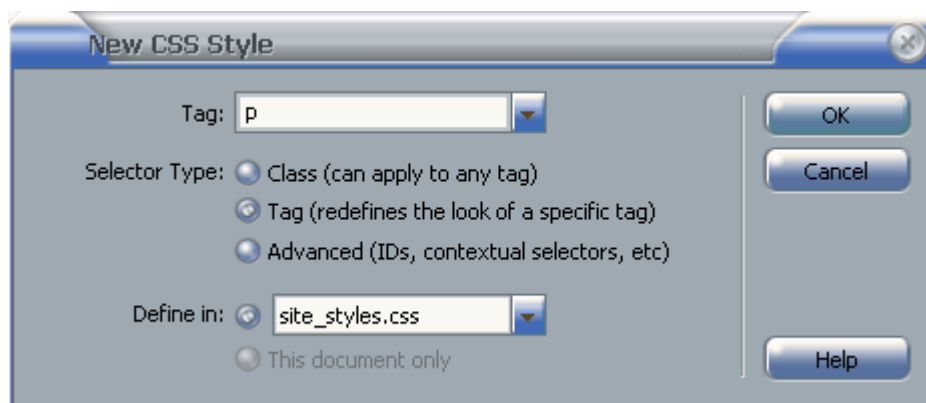


- 2) Click the Edit Style Sheet icon  at the bottom of the panel.

A list of all of the style rules currently contained in the CSS file will be shown.




- 3) Click the **New** button to create a new style rule within the CSS file.

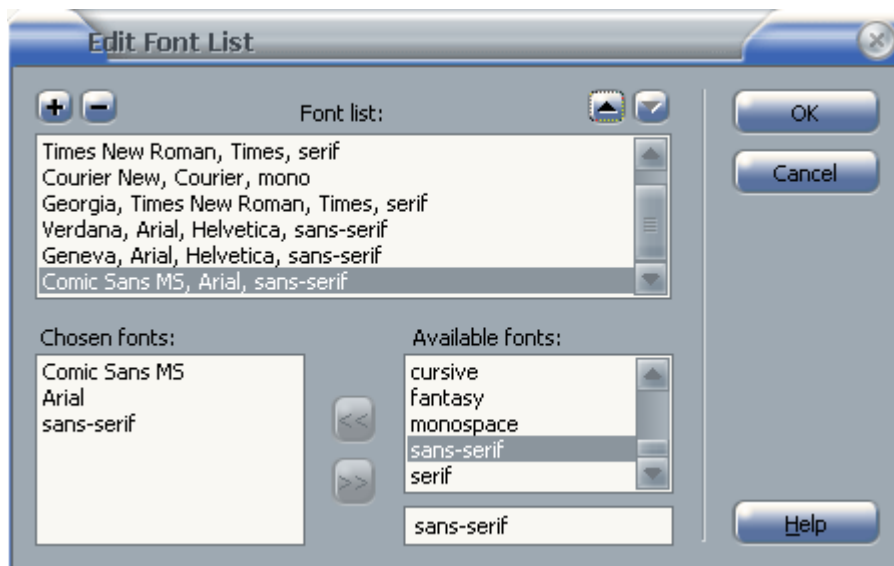


- 4) Set the Selector type to **P** as shown above and click **OK**. This will create a rule affecting all of the paragraph text in the website.

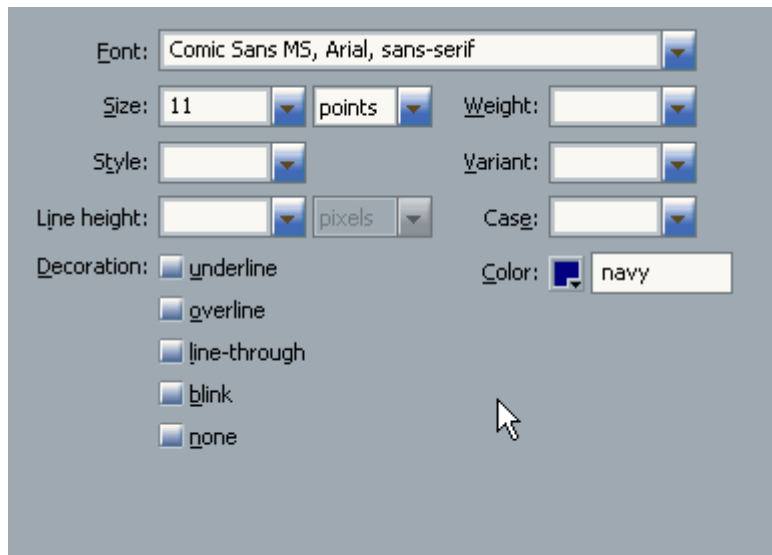
- 5) Make sure **Type** is selected from the list of categories on the left.



- 6) Click on the Font list. You will notice that your options are limited. For the sake of simplicity, Dreamweaver usually only shows a selection of the most commonly used fonts in websites. If you would like to use a font that doesn't appear in the list, choose the **Edit Font List** option.
- 7) Scroll through the list of **Available fonts** until you find *Comic Sans MS* (or another font if you don't have that one on your computer). Select *Comic Sans MS* and click the  button to add it to your list of **Chosen fonts**.
- 8) You can also specify alternate fonts in case someone looking at your website doesn't have the font you specify. Add *Arial* (a more common font) and then add *sans-serif* (a generic font name for any font like Arial that doesn't have serifs – in other words, no curly bits at the top and bottom of letters the way this text you're reading does).



- 9) When the options look like the example below, click **OK**.
- 10) Now choose your new font selection (*Comic Sans MS, Arial, sans-serif*) from the Fonts list.
- 11) For the other type options, set the font size to 11 with Points as the unit of measurement and enter navy for the colour. The completed options should look like the example on the following page.



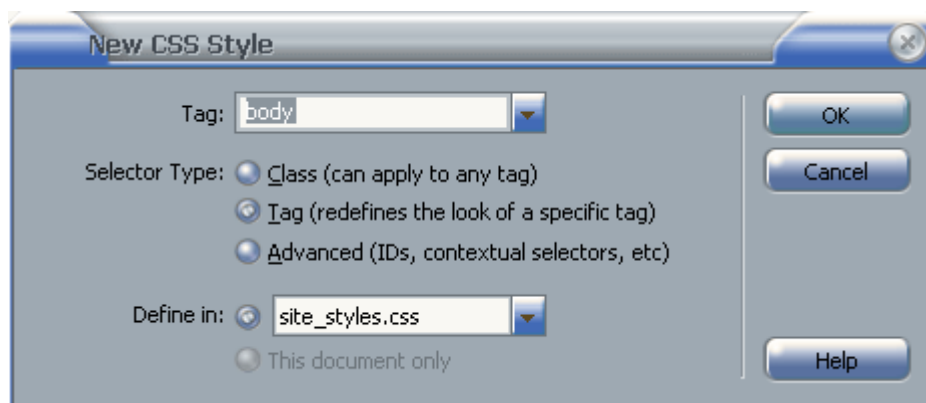
- 12) Click **OK** when the options have been selected.
- 13) Click on **Done** to close the style sheet options. You will see the changes in each of your documents. The changes only affect the paragraph text in the documents. Some other types of text such as text in table cells (in HTML's TD tags) and text in lists (in HTML's LI tags) will not have that formatting. We can easily modify the CSS file to group together a few selectors.
- 14) Open the site_styles.css file.
Change the P selector so that it also includes a selector for LI and TD tags with a comma between each one as shown in the example below.

```

p, td, li {
    font-family: "Comic Sans MS", Arial, sans-serif;
    font-size: 11pt;
    color: navy;
}

```

- 15) Save the changes to the CSS file. The text in tables, bulleted lists and numbered lists in the other documents will now be updated with information from the CSS file.
- 16) Repeat steps 1-2 to edit the stylesheet. Of course you could just open the CSS file itself but that requires a good knowledge of editing CSS files which you may get with enough practice.
- 17) Click **New** to add a new style selector.
- 18) The selector type for the new tag will be body as shown below. This will create a style rule which will affect the whole page of a document.



19) Select **Background** from the **Category** list.

20) For **Background image** enter *back.jpg*.

Background image:

21) Select **Box** from the **Category** list.

22) For the **Margin** option, make sure **Same for all** is ticked and enter 0 for the margin size. This will remove the blank space around the edge of the page.

Margin

Same for all

Top:


Right:

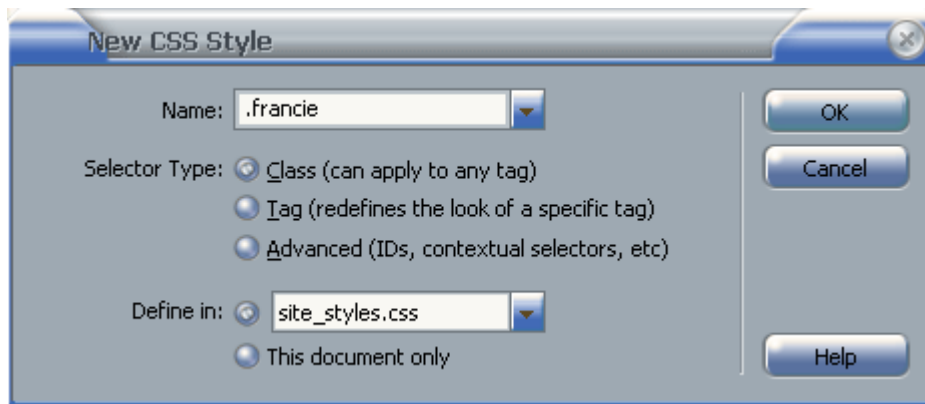
Bottom:

Left:

23) Click **OK** and then **Done** to apply the changes.

Exercise 3 – Creating CSS Classes


- 1) Click on the **New CSS Style** icon  in the **CSS Styles** panel.
- 2) For Selector Type choose class and make sure Define in is set to site_styles.css.
- 3) Enter *.francie* for the class name (class names must begin with a full stop).



- 4) Click **OK** to create the new style rule.
- 5) In the **Type** category, select a dark green colour for the text.

Color:  #003333

- 6) In the Background category, choose a light green colour for the Background Colour.

Background color:  #B7FFC9

- 7) Click **OK** to finish adding the new style rule.
- 8) Open *about.html*.
- 9) Select the text *Francie's Flowerpots*.
- 10) From the Properties Panel, select *.francie* from the **Style** list.

Style **francie** 

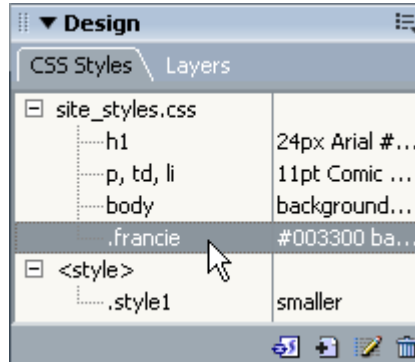
The selected text will be formatted according to the new style class you created.


- 11) Apply the new style class to other pages of the website where the name of the business appears.

Exercise 4 – Editing a Style

Any style can be edited and any changes made to a style will be seen in parts of the website that use the changed style. We'll change the new style class to see how the changes update through the site.

- 1) Select the *.francie* class in the **CSS Styles** panel.



- 2) Click the **Edit Style** icon  at the bottom of the palette.
- 3) Change the text colour to a dark brown and click **OK**.


Color:  #660000

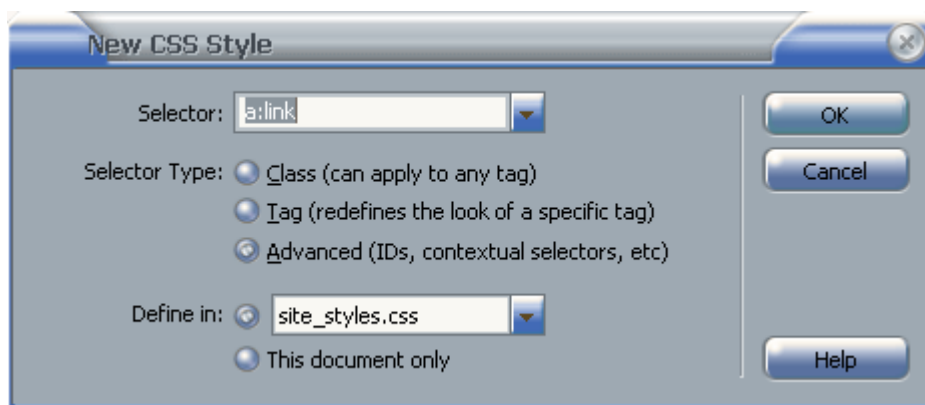
Any parts of your website referring to that class will now change.

Exercise 5 – Creating Styles for Hyperlinks

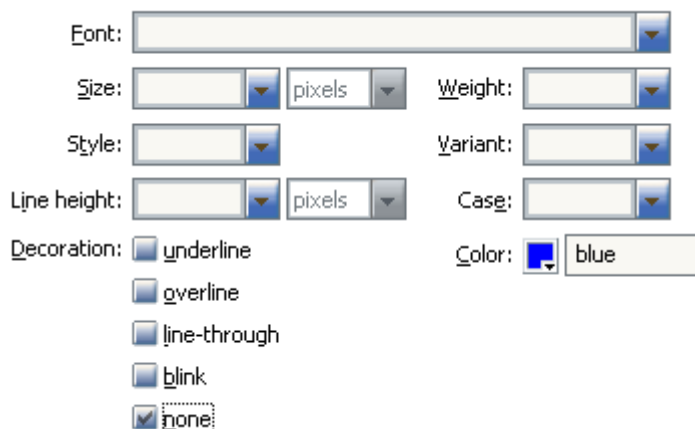
The CSS specification includes a few style classes that are reserved especially for hyperlinks (the **A** or **anchor** tag).


- The `a:link` class allows you to create style rules for regular hyperlinks (usually blue).
- The `a:visited` class allows you to create style rules for hyperlinks that point to a location the user has already visited (usually purple by default).
- The `a:active` class allows you to create rules for hyperlinks that are active, or links that have been clicked (usually red while the browser waits for the next page to start appearing).
- The `a:hover` class allows you to create rules for hyperlinks that have a user's mouse 'hovering' over them.

- 1) Open any of the HTML documents in your site.
- 2) From the CSS Styles panel, click the **New CSS Style** icon .
- 3) Choose **Advanced** for **Selector Type** and then choose `a:link` for the selector. Make sure **Define in** is set to `site_style.css`.

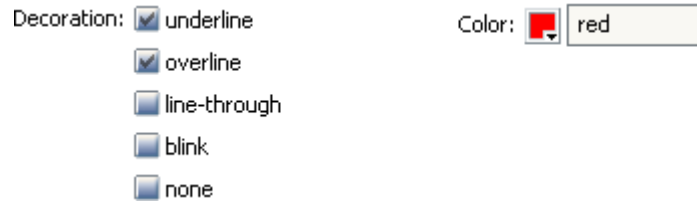


- 4) Click **OK** to create the new style rule.
- 5) Set font **Decoration** to *none* (so the links won't have the usual underline) and the **Color** to *blue*.

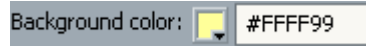


- 6) Click **OK** to create the new style rule.
- 7) From the CSS Styles panel, click the **New CSS Style** icon .
- 8) Create similar classes for `a:visited` (with the colour changed to purple) and `a:active` (with the colour changed to red).
- 9) Create a new class and choose **Advanced** for **Selector Type**. Choose `a:hover` for the selector. Make sure **Define in** is set to `site_style.css`.

10) In the **Type** category, set the **Color** to *red* with the **Decoration** set to *underline* and *overline*.



11) In the **Background** category, set a light yellow for the background colour.



12) Click **OK** to finish creation of the new style rule.

13) From the **File** menu select **Save All** to make sure all documents including the CSS file are saved.

14) Preview one of the HTML documents in your web browser. Move your mouse over the links to test them.

15) Open the *site_styles.css* file to see the style rules that have been created.

Dynamic HTML

DHTML or Dynamic HTML is a combination of HTML, CSS and scripting such as Javascript, that are used to make web pages more interactive. Later exercises will make use of Dreamweaver's DHTML features to create some advanced web site navigation tools but for now we'll use it for a more simple effect. We're going to use DHTML to change the images in the navigation bar when the mouse moves over them. In each link on the navigation bar, we will set things up so that when a mouse goes over the image, it is replaced with a modified version of the image. When the mouse goes away from the link, the original image will be restored. This will result in a simple mouse over effect.

Exercise 6 – Creating a DHTML Mouse Over Effect

- 1) Since we're going to be modifying the navigation bar, we'll need to open the template for editing so open *Main Pages.dwt*.
- 2) Click on the first linked image in the navigation bar (Home).



- 3) Before we assign any behaviours to this image, it would be useful to give it a name, otherwise it will be referred to as something generic like *image1*. In the **Properties Panel**, enter *nav_home* as the name of the selected image.

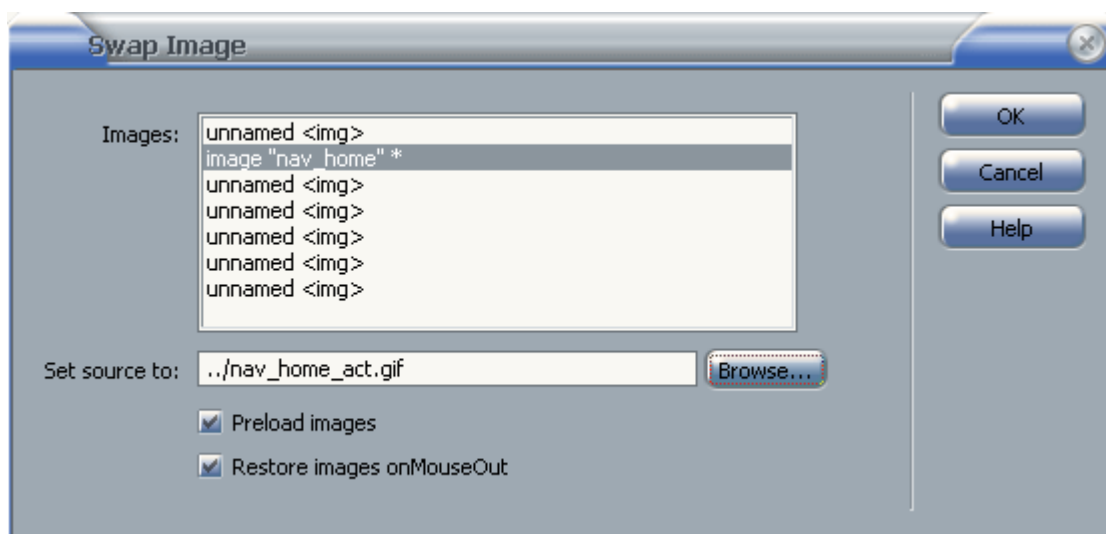
Note The object name for the image is not the same as the filename. The Object name identifies the part of the page where the image appears so that it can be referred to from elsewhere. The filename identifies which image file is to appear in that position.

- 4) Click on the heading for the **Tag Inspector Panel** and then click the **Behaviours** tab. If that panel is not visible, you can select **Behaviours** from the **Window** menu or press **[Shift] [F4]**.



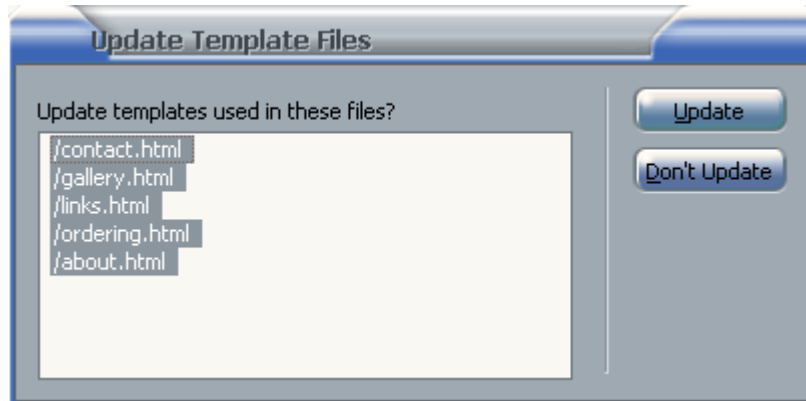
- 5) With the Home image still selected, click the **Add Behaviour** icon **+** in the **Behaviours Panel**.
- 6) A list of available behaviours will appear. From the list select *Swap Image*.
- 7) From the list of images select *nav_home*. You will notice that the other images don't have names. We'll give them each a name before assigning a behaviour to them.
- 8) Make sure the **Preload Images** and **Restore Images onMouseOut** options are ticked.
- 9) Click the Browse button and select the image *nav_home_act.gif*.

Note Giving each of your files clear names like this makes it easy to tell what's what. In this case, images in the navbar all start with *nav*. Images used for actions have the same name as the normal images with *act* added to the end of the names.



- 10) Click **OK** to add the behaviour. A second behaviour restoring the original image will also be added for when the mouse goes off the image.

- 11) Create similar behaviours for each of the other links in the navbar. Make sure you assign a name for each image before you create a behaviour.
- 12) Once you have created MouseOver behaviours for each of the link images, save the changes to the template. Make sure each of the files based on the template is updated.



- 13) Close the template.
- 14) Open and preview one of the site's HTML documents in your web browser. Test the links.

