

# Using Dreamweaver CS6

## 4 – Creating a Template

Now that the main page of our website is complete, we need to create the rest of the pages. Each of them will have a layout that follows the plan shown below.

<i>Logo</i>	<i>Page Heading</i>
<i>Home</i>	<i>Page content in this area</i>
<i>About Us</i>	
<i>Gallery</i>	
<i>Ordering</i>	
<i>Contact Us</i>	
<i>Links</i>	

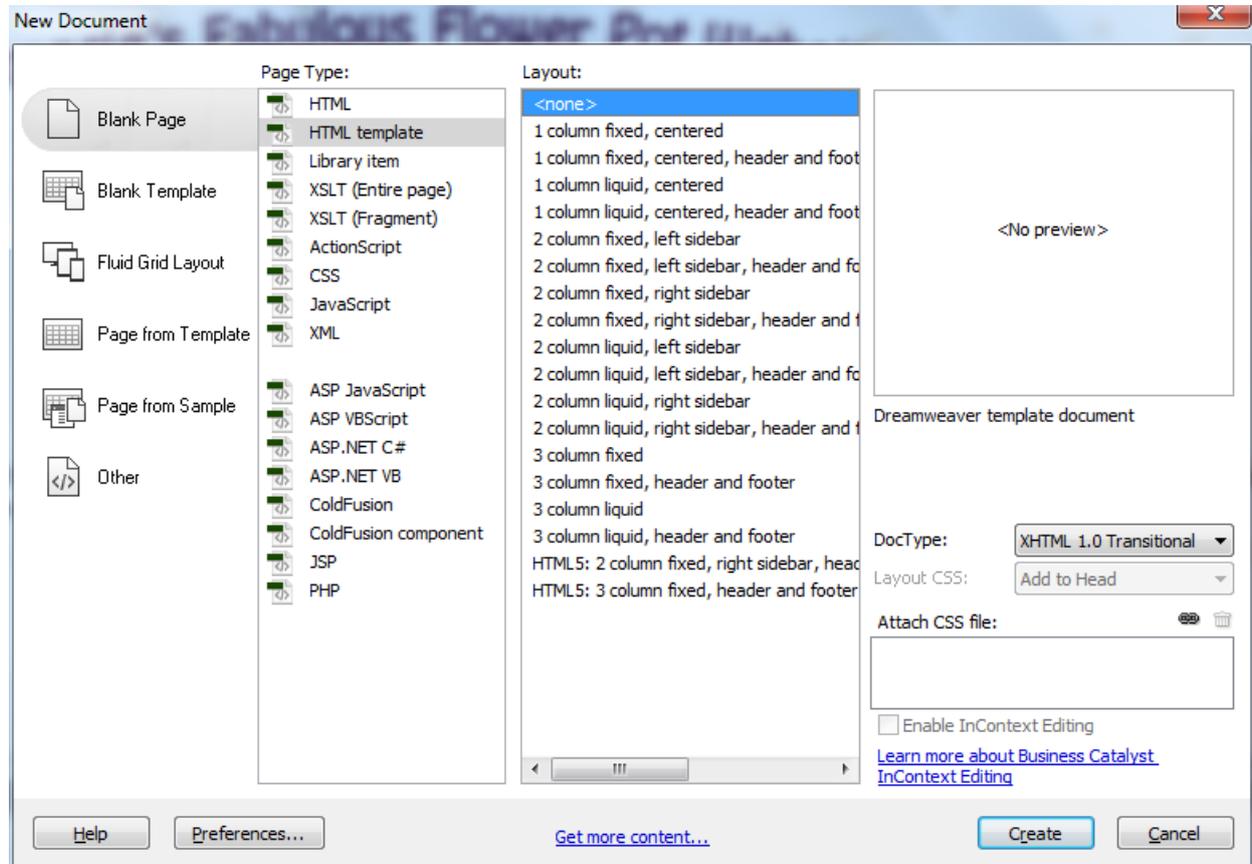
When we are creating several pages that will all be similar, we can use a template to create the basic structure and then use that template for a starting point as we create each individual page. While we create the template, we'll be using a few new techniques. Mainly, we will use CSS positioning to create the layout for the page. The CSS styles will be used to position the heading and the navigation links on the side. Where something like this would have been done with tables in older websites, CSS is a lot more common in modern website and allows you to design pages with more control and ease.

**Tip** An alternative to creating a template is to create one page with the right layout, then use **save as** to make copies of the page under different filenames and modify the content of each one.

## Exercise 1 – Starting the Template

In these exercises we will be continuing the *Francie's Flowerpot website* created in the last exercises.

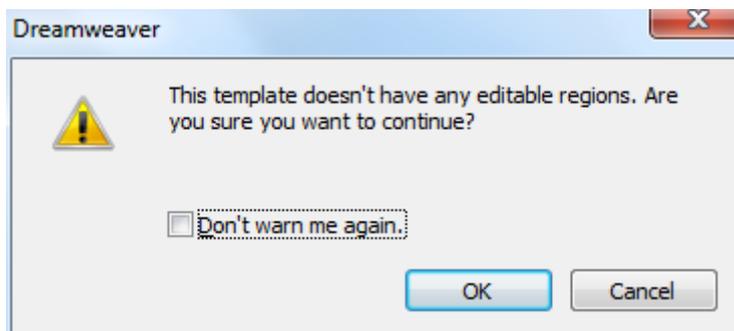
- 1) From the **File** menu select **New** ([Ctrl] [N]).



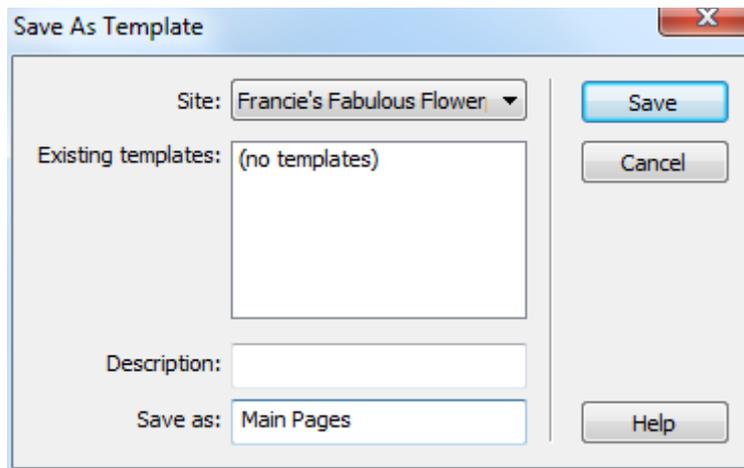
- 2) From the list of options, make sure **Blank Page** is selected in the left. Select **HTML template** and then click **Create**.

A new page will be created. We will use this page as the basis for our template. The first thing we will do is save the template.

- 3) From the **File** menu select **Save** or press [Ctrl] [S].



- 4) A message may appear telling us we don't have any editable regions yet. That's fine since we'll be doing that later. If you like, you can click the **Don't warn me again** option before clicking **OK**.



5) In the **Save as** area, enter *Main Pages* and click **Save**.

**Note** The template will be saved in a sub folder under the folder your site is in, so you can easily open the template for later modifications.

## Styles in Websites

Remember than an earlier exercise stated that HTML was originally only intended to define the structure of a website. E.g. headings and paragraphs. It was never originally intended to provide a lot of information about how these page elements looked. As the world wide web became more popular, website developers wanted more control over how their websites looked. This was accomplished with the addition of HTML tags like FONT that allowed for some basic formatting options. Tags like this tended to make HTML bloated and complex.



To address this issue, the **Worldwide Web Consortium** added **CSS**. CSS stands for **Cascading Style Sheets** and is the common way for modern websites to be positioned and formatted. The idea is that HTML still defines the structure of the page, while CSS describes how the elements on the page actually look. The ‘Cascading’ part of the name refers to the way styles work. Style rules can cascade through your website to make it easy to control formatting throughout your whole site from one location. CSS allows you to control things such as

- Where elements will be on the page - you can be very exact with positioning
- Formatting – such as font and colour

Using styles to format your website instead of using HTML formatting tags has the following advantages:

- 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.

A document in a website can be formatted by CSS in 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.

## 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.

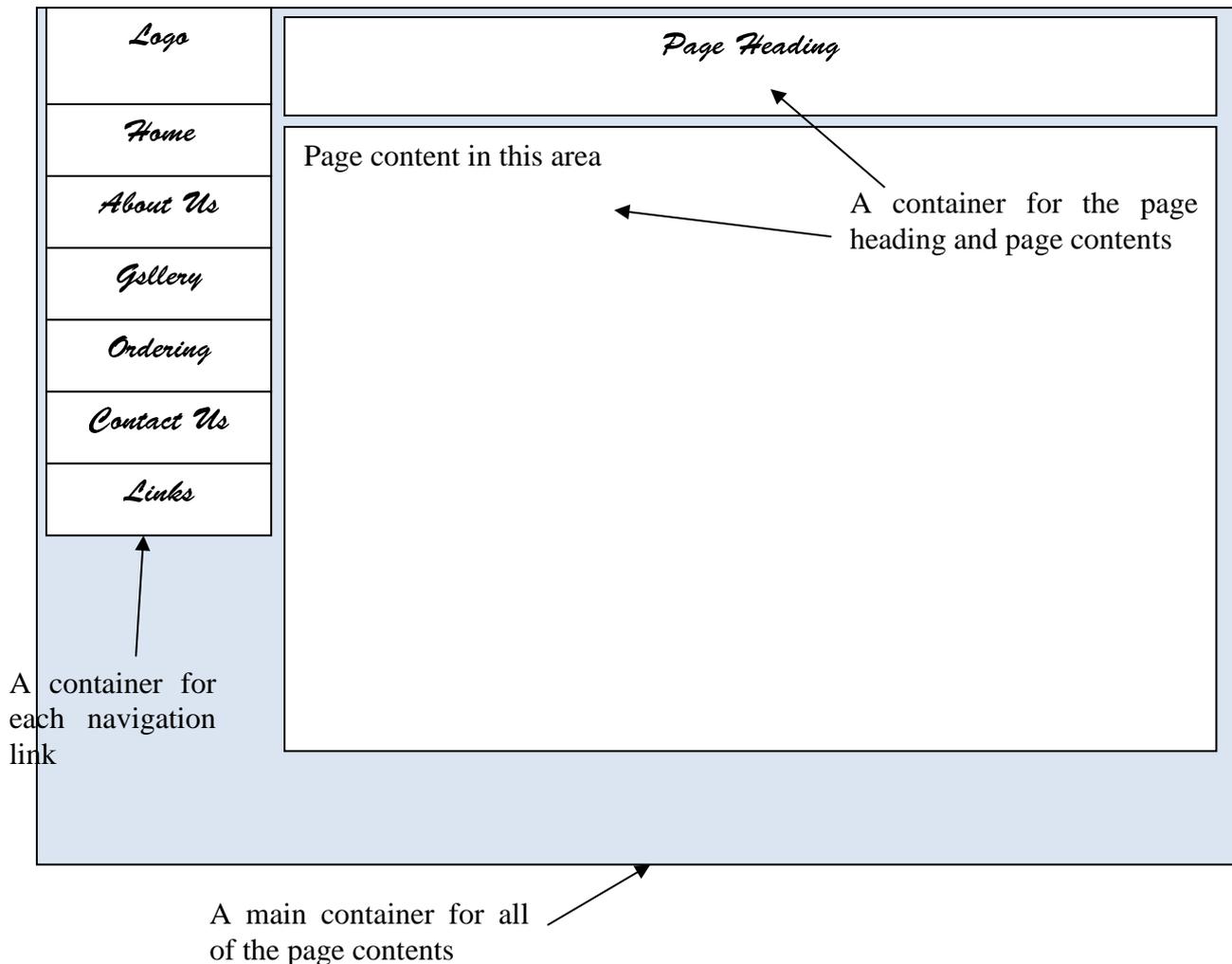
**Note** Dreamweaver allows you to control how CSS is written. If you want Dreamweaver to use abbreviated forms when writing CSS, you can open the Dreamweaver **Preferences** from the **Edit** menu. From the preferences you can go to the **CSS Styles** category and select the options to use shorthand.

When creating CSS rules:

Use shorthand for:  Font  
 Background  
 Margin and padding

## Exercise 2 – Creating a Page Layout with CSS

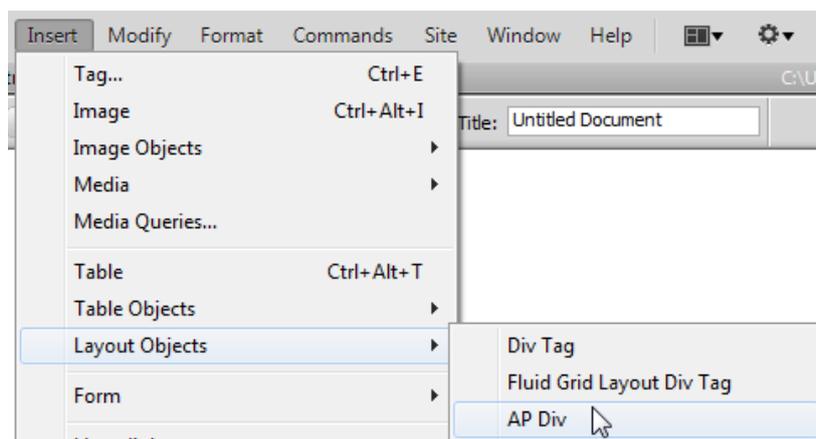
Our page will be structured using HTML's DIV (division) tags. A division tag creates a container on the page that will then be positioned and formatted using CSS. Our page will be created with a series of containers inside a main container similar to the following example.



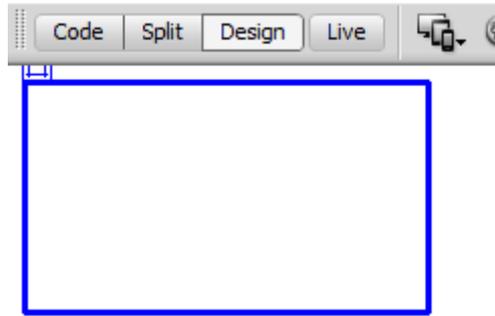
As you can see in the example, one container will be used to position the page contents on the screen. This will allow us to determine how much of the screen's width will be taken up with the page contents. Each of the other components on the page, I.e. navigation links, heading and contents will be contained inside the main container.

We'll start by creating the main container.

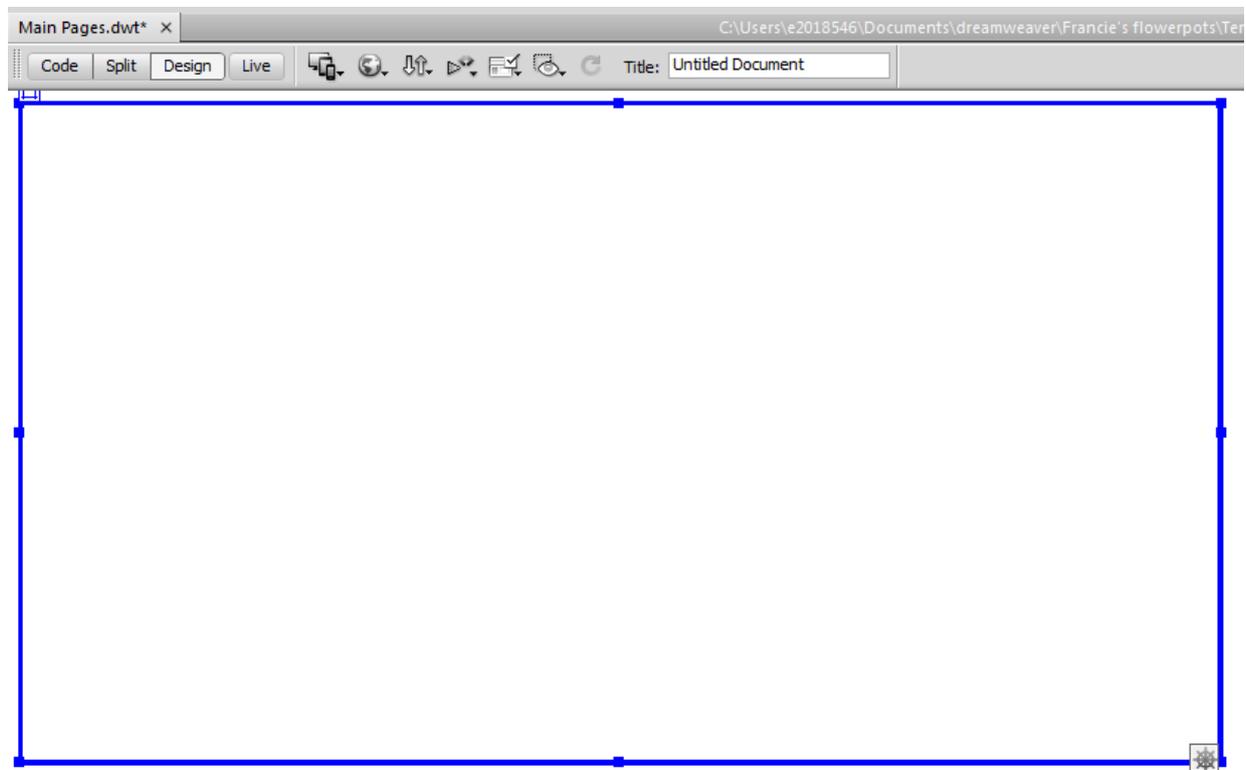
- 1) From the **Insert** menu select **Layout Objects** and then select **AP Div**.



A box (**DIV**) will appear on your page.



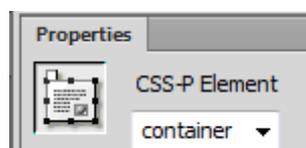
- 2) Click on the border of the **DIV** box to select it and then resize the box so that it takes up most of the space on the page.



Along the bottom of the editing window you will see that you are editing a **DIV HTML** tag which is inside the **BODY** tag. The div tag will have a name like apDiv1 but we can give it a name of our own choosing.

```
<body> <div #apDiv1>
```

- 3) With the **DIV** still selected, click in the box under **CSS-P Element** in the properties area. Change the name to *container*.



4) Take a look at **Code View**. A div tag has been created in the body area of the page.

```
<div id="container"></div>
```

5) Further up the page, a style section has been added. Later on, we will move the style information in to a separate file so that all of the pages in the site can use the same styles.

```
<style type="text/css">
#container {
    position: absolute;
    width: 811px;
    height: 449px;
    z-index: 1;
}
</style>
```

While we're in design view we'll make a quick change to the CSS code.

6) Adjust the CSS code so that it looks like the following.

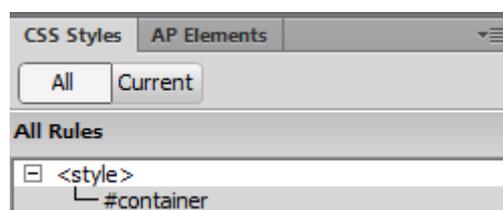
```
<style type="text/css">
#container {
    position: absolute;
    width: 900px;
    height: 100%;
    z-index: 1;
    background-color: #a2d39c;
    top: 0px;
    bottom: 0px;
}
</style>
```

<b>Width</b>	<b>900px</b>	The container will be 900 pixels wide. Since this will contain all of the elements in our web page, this allows us to set it to a size that should suit most computer screens.
<b>Height</b>	<b>100%</b>	The container should fill the screen from top to bottom
<b>Top</b>	<b>0</b>	There should be no gap between the top of the container and the top of the web browser window
<b>Bottom</b>	<b>0</b>	There should be no gap between the top of the container and the top of the web browser window
<b>Z-Index</b>	<b>1</b>	When you have several elements overlapping as we will, the Z-Index value determines which will be in front. Since we want the other elements to appear in front of this one, we will give them larger numbers for their Z-Index values.

7) Return to design view and save your work. You can preview it in your web browser if you wish.

8) Click the **CSS Styles** tab to display the panel if it is not already showing. If it is not on the right side of the screen where it usually is, you can select **CSS Styles** from the **Window** menu

The panel will list any styles being used by the current document. At the moment, the only one we have is the one we just created for the container. You may need to click the + sign next to <style> to make sure the styles are showing.

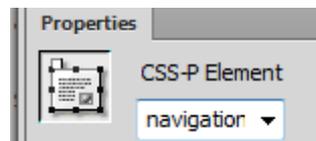


- 9) **Double-click** on #container in the **CSS Styles** panel. This will open the options for that style. There are different style categories down the left side which we will use later on.
- 10) Click **OK** or **Cancel** to close the options. You can also click the pencil icon at the bottom of the panel to edit any style you have selected in the list. 

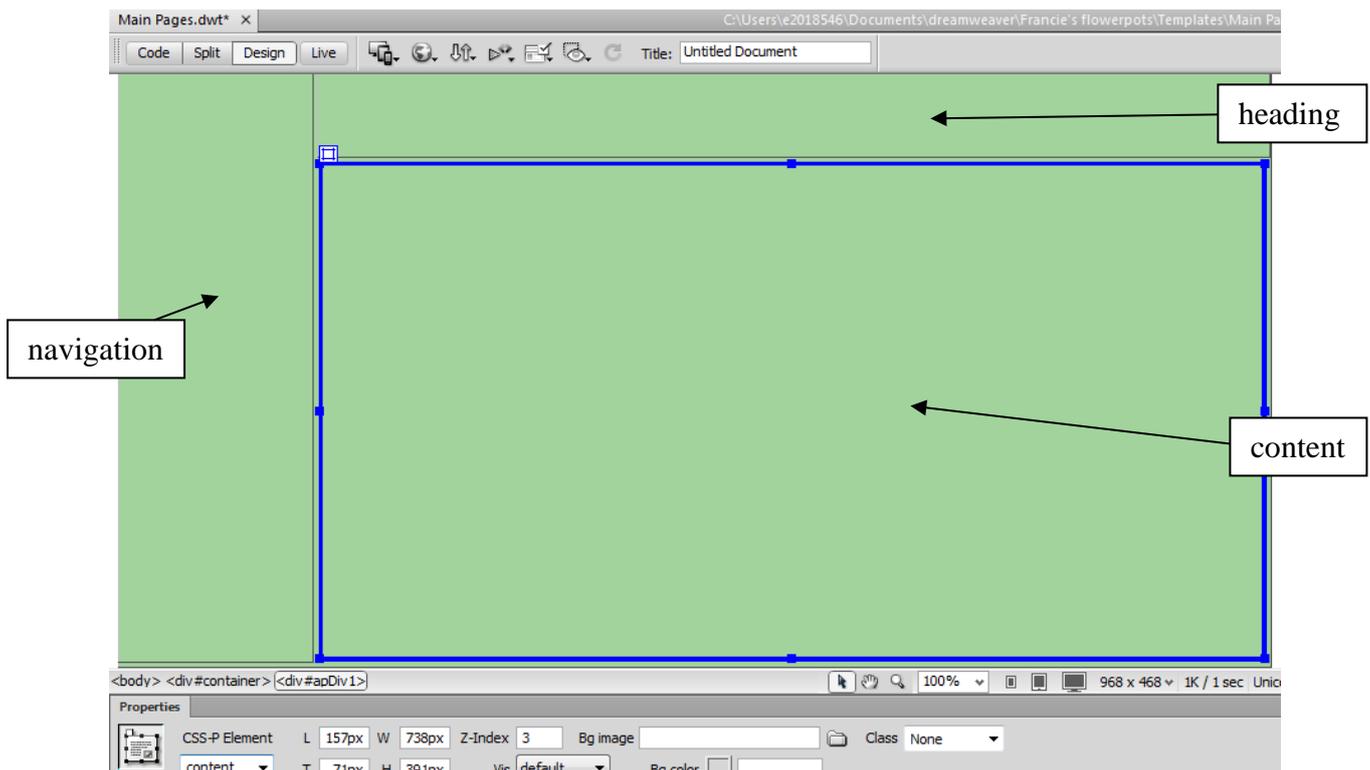
### Exercise 3 – Creating Nested DIV Tags

Now that we have created a Division that will hold our page content, we will create an additional division for the page navigation that will be inside that main one. Inside that will be a division for each of the options in our navigation bar. We will also add a division for the page heading and another one for the page content. This is referred to as ‘nesting’ as the new one will all be inside the one we just made.

- 1) Click inside the existing container **DIV** to make sure the new one will be inside it.
- 2) From the **Insert** menu select **Layout Objects** and then select **AP Div**.
- 3) Resize the new **DIV** so that it takes up the left part of the container (remember it will be used as a navigation bar).
- 4) Make sure the new **DIV** is still selected and change the name to *navigation*.



- 5) Click inside the *Container DIV* and outside the *Navigation DIV*. We want the next one to also be inside the container.
- 6) Insert another **DIV**. Name it *heading* and position it so that it goes along the top without overlapping the navigation.
- 7) Click inside a blank part of the *container DIV* (not inside one of the other **DIV**s).
- 8) Insert a third **DIV**. Name it *content* and position it so that it takes up the remaining space. The three **DIV**s should now be positioned similar to the following example. Don't worry about positioning them exactly. We will use CSS later to be more precise.



9) Check the HTML code in the **Code View**.

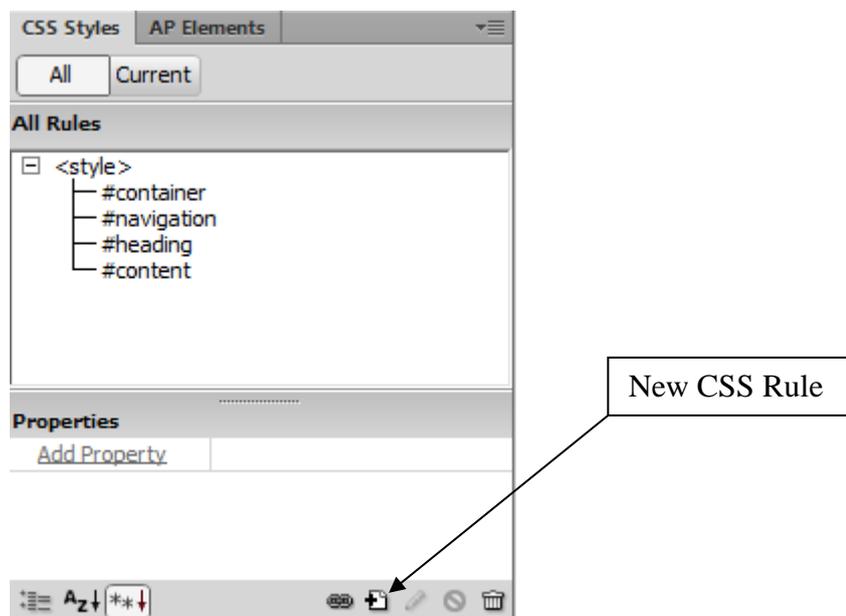
The HTML code should look similar to the following example. From the code you should be able to see that the three DIV tags you just created are nested within the container DIV. I.e. between the opening and closing tag of the container DIV. There will also be some new CSS code added to the top of the page which we will modify later on. Notice that Dreamweaver indents the nested tags. That makes no difference to how the website will appear in the browser but makes the HTML much easier to read.

```
<body>
<div id="container">
  <div id="navigation"></div>
  <div id="heading"></div>
  <div id="content"></div>
</div>
</body>
```

10) Return to **Design View**. It's time to start modifying the CSS. We'll start by creating some CSS code that tells the web browser how we want everything in the body section of our document to look.

11) Make sure the **CSS Styles** panel is showing (usually on the right of the screen). If it is not showing, select **CSS Styles** from the **Window** menu or press **Shift F11**.

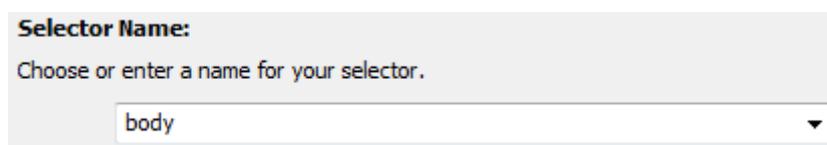
12) Click the **New CSS Rule** button at the bottom of the panel.



13) In the **Selector Type** option, select **Tag** since we want our new rule to apply to an existing HTML tag (the BODY tag).

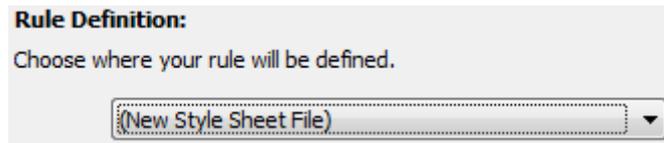


14) For **Selector Name** select **body** from the list (it may already be selected).

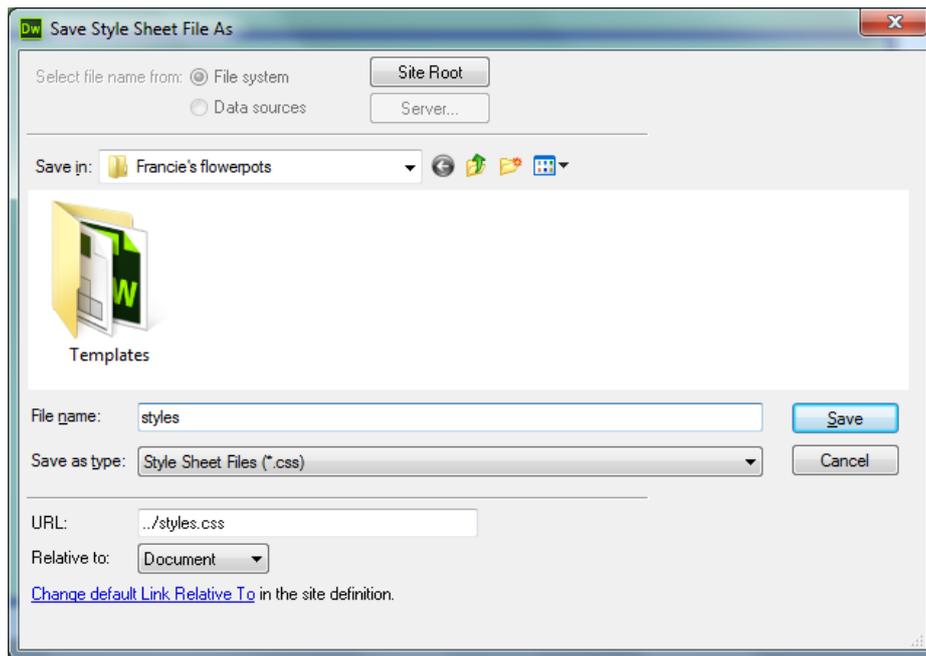


So far all the CSS used in our page is within the page itself. We want our CSS rules to be available to every page within our site. To make this possible, we will be placing all of our CSS rules into a separate CSS file. We will also move our existing CSS to the new file.

For the **Rule Definition** select **New Style Sheet File**.



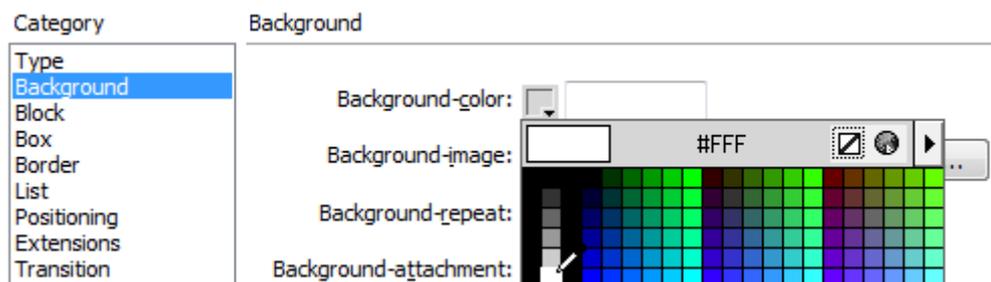
- 15) With the three options above selected, click **OK** to create the new style rule. Since we specified that we would create the new rule in a new CSS file, Dreamweaver will now ask you for a file name and location for the new file.
- 16) Make sure the location is set to save the file within the *Francie's Flowerpots* folder.
- 17) For the filename enter *styles*. When you save the file, Dreamweaver will add a *.css* extension so the filename will become *styles.css*.



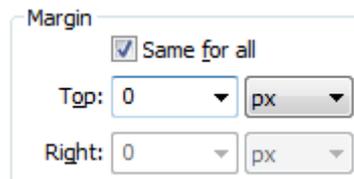
- 18) When the filename and location have been set, click **Save** to create the new CSS file. The new style rule we create will be part of that new file instead of being inside the document we are editing. The document we are editing will refer to that new file.

When the new CSS files is saved, the **CSS Rule Definition** options will appear. Instead of editing the CSS code directly, we can use these options to specify our CSS rules for the **BODY** tag.

- 19) Choose the **Background** category from the left side of the options.
- 20) Set the **Background-colour** to white. A Hexidecimal code #FFF will be selected to represent the colour white.

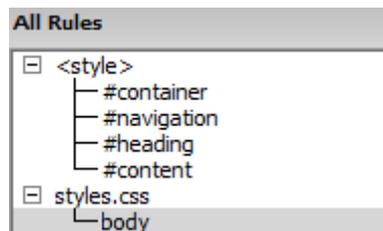


- 21) Select the **Box** category.
- 22) Enter 0 for the **Margin** option. If you leave the **Same for all** option ticked you won't need to set it for all margins. This option will mean that there won't be the usual blank space around the edges of the page.



23) Click **OK** to complete the Style rule options and close the options window.

In your **CSS Styles** panel a new rule for **BODY** will be added beneath a styles.css heading. This lets us know that the new rule is part of that external file instead of being inside our template document that we are editing.

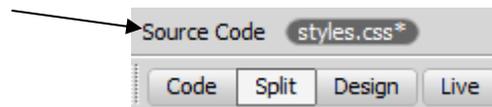


24) At the top of our editing window you will see that styles.css has been added beneath our template file. Click on *styles.css* to see the source code for our CSS file.



```
@charset "utf-8";
body {
    background-color: #FFF;
    margin: 0px;
}
```

25) Click the regular **Source Code** button to see the source code for our Main Pages template.



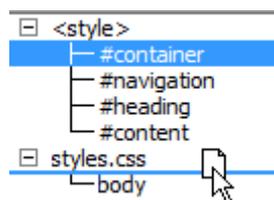
Beneath the existing style rules and right before the closing HEAD tag, you will see the following line.

```
<link href=" ../styles.css" rel="stylesheet" type="text/css" />
```

This tells the document that it will be using style rules from that file. In this way we can have every page within a site referring to a single file so that they can all use the same style rules. We're going to take all of our existing styles and move them to this new file.

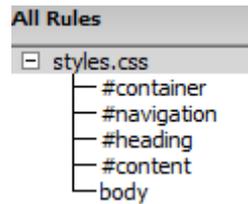
Generally it is best to keep your style rules in one file unless you have any rules that should only apply to particular pages.

In the CSS Rules panel, drag the #container style rule so that it is below the styles.css heading



26) Do the same for the #navigation, #heading and #content style rules (the # symbol in front of each one specifies that those rules are referring to an element on the page with that name).

27) Right click on <style> and select **Delete**.



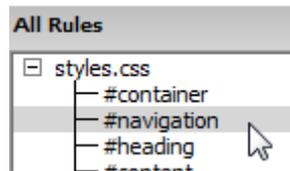
28) Your **CSS Styles** panel should now look like the example above.

29) In your HTML code view, the only style part remaining will be the link tag referring to the styles.css file. Check the code for *styles.css* and you will see that the style rules are all now in there.

#### Exercise 4 – Changing Existing Style Rules

We already edited the style rule for the container DIV to position it precisely. Now we will edit the style rules for the other DIVs to position them exactly where we want them.

1) In the **CSS Styles** panel double-click on #navigation to edit this style rule.



2) Change following options. The background-color option is under the **Background** category while the rest are under the **Positioning** category.

<b>Width</b>	150px
<b>Height</b>	605px
<b>Top</b>	0
<b>Left</b>	0
<b>Z-Index</b>	2
<b>background-color</b>	#a2d39c

3) Click **OK** once you have entered all those options

4) In the **CSS Styles** panel double-click on #heading to edit this style rule.

5) Change following options.

<b>Width</b>	750px
<b>Height</b>	60px
<b>Top</b>	0
<b>Left</b>	150px
<b>Z-Index</b>	2
<b>background-color</b>	#a2d39c
<b>Padding</b>	5px – same for all (in the <b>Box</b> category)

6) Click **OK** once you have entered all those options

7) In the **CSS Styles** panel double-click on #content to edit this style rule.

<b>Width</b>	750px
<b>Height</b>	100%
<b>Top</b>	60px
<b>Bottom</b>	0
<b>Left</b>	150px
<b>Z-Index</b>	2
<b>background-color</b>	white
<b>Padding</b>	5px – same for all (in the <b>Box</b> category)

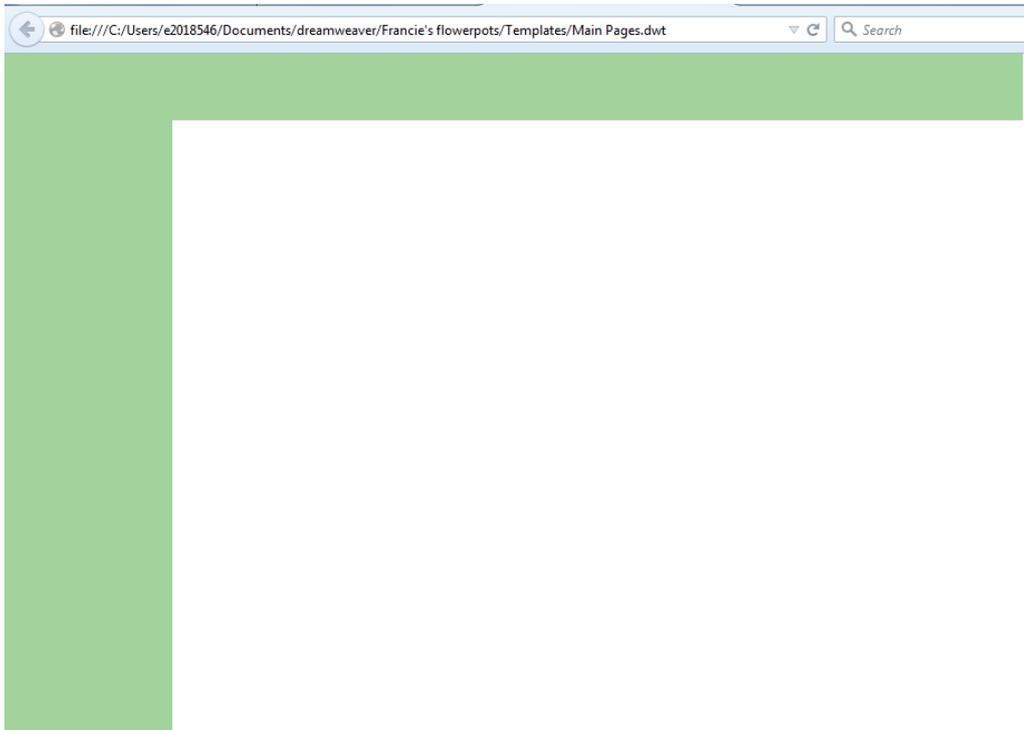
8) Click **OK** once you have entered all those options



You will notice that the Main Pages.dwt (the template) and the styles.css file both have a \* next to them which tells us that both files have unsaved changes.

9) From the **File** menu select **Save All**.

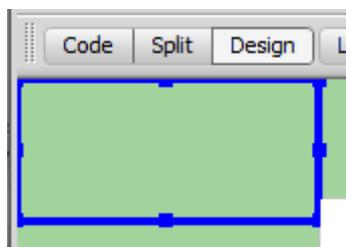
10) Preview your page to in a web browser view the changes.



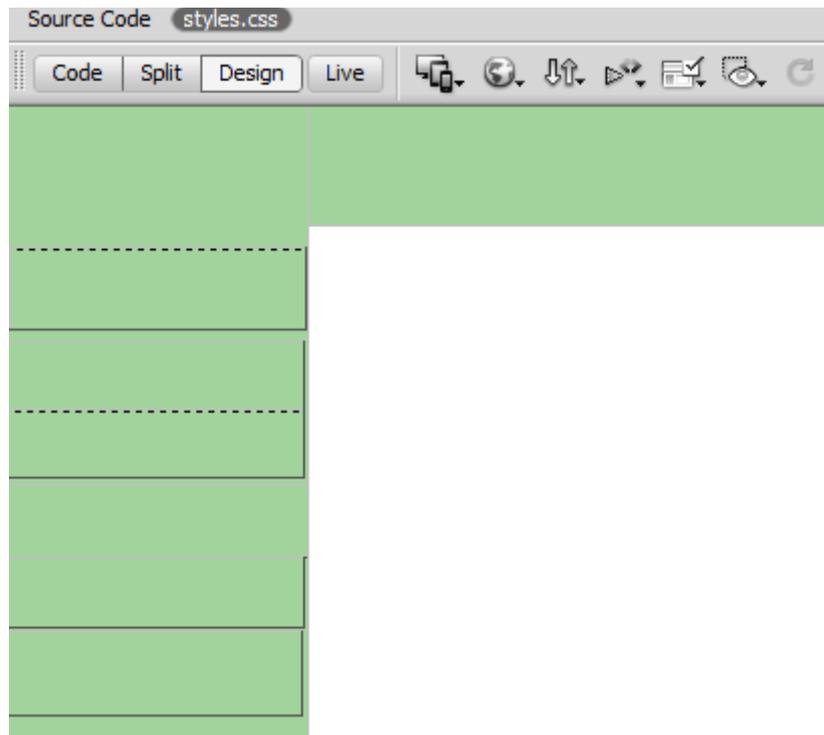
### Exercise 5 – Completing the Navigation

Now we will add the individual components of the navigation bar. This will consist of 7 **DIV** tags inside the navigation **DIV**. One will be a logo for the site and the other 6 will be links within the site navigation bar.

- 1) Click inside the navigation **DIV** to ensure that the new **DIV** we create will be inside it.
- 2) Insert a new **DIV**. Call it **nav\_logo** and position it at the top of the navigation section similar to the example below (remember we will use CSS later to position it precisely).

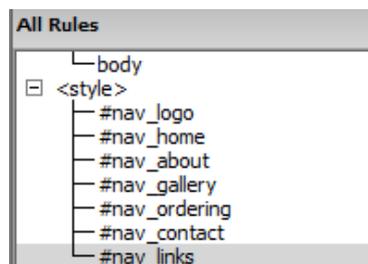


- 3) Click below the one you just added to make sure that you are inside the navigation **DIV** (the bar along the bottom of the design window should show `<body> <div#container> <div#navigation>`)
- 4) Insert a new **DIV**. Call it **nav\_home** and position it at the top of the navigation section similar to the example below (remember we will use CSS later to position it precisely).
- 5) Create 5 additional **DIVs** called **nav\_about**, **nav\_gallery**, **nav\_ordering**, **nav\_contact** and **nav\_links**. Position each one approximately under the previous one similar to the example below.



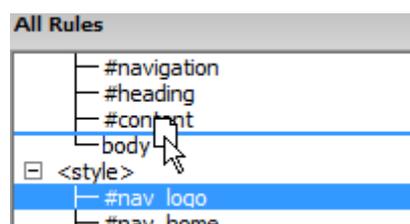
Now it's time to create the CSS rules for each of these new DIVs.

When we created the new DIVs, each one would have had a style rule created inside the page. For each one, we will edit the properties of the style rule and then move the style rule in to the CSS file.



- 6) Double click on #nav\_logo in the **CSS Styles** panel.
- 7) Edit the style properties to match the example below.
  - position: absolute;
  - width: 150px;
  - height: 125px;
  - z-index: 3;
  - left: 0px;
  - top: 0px;
  - background-color: #a2d39c;

Once you have finished editing the style rule, drag it in to the styles.css file.



- 8) Edit each of the remaining style rules as follows. After you have edited each one, move it in to the styles.css file.

```

#nav_home {
    position: absolute;
    width: 150px;
    height: 80px;
    z-index: 3;
    top: 125px;
    left: 0px;
}
#nav_about {
    position: absolute;
    width: 150px;
    height: 80px;
    z-index: 3;
    top: 205px;
    left: 0px;
}
#nav_gallery {
    position: absolute;
    width: 150px;
    height: 80px;
    z-index: 3;
    top: 285px;
    left: 0px;
}
#nav_ordering {
    position: absolute;
    width: 150px;
    height: 80px;
    z-index: 3;
    top: 365px;
    left: 0px;
}
#nav_contact {
    position: absolute;
    width: 150px;
    height: 80px;
    z-index: 6;
    top: 445px;
    left: 0px;
}
#nav_links {
    position: absolute;
    width: 150px;
    height: 80px;
    z-index: 3;
    top: 525px;
    left: 0px;
}

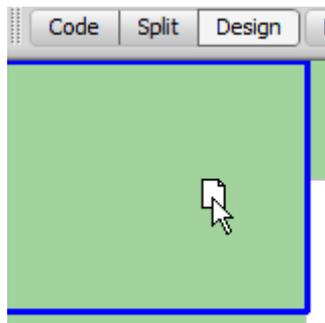
```

9) Remember to save changes as you go.

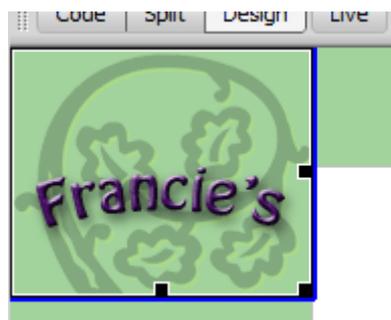
We are going to add an image inside each of the navigation DIV tags that we have placed.

10) Click inside the top one which should be nav\_logo.

11) In your files list there should be an image called nav\_logo.gif. Drag it to place it inside the nav\_logo DIV.



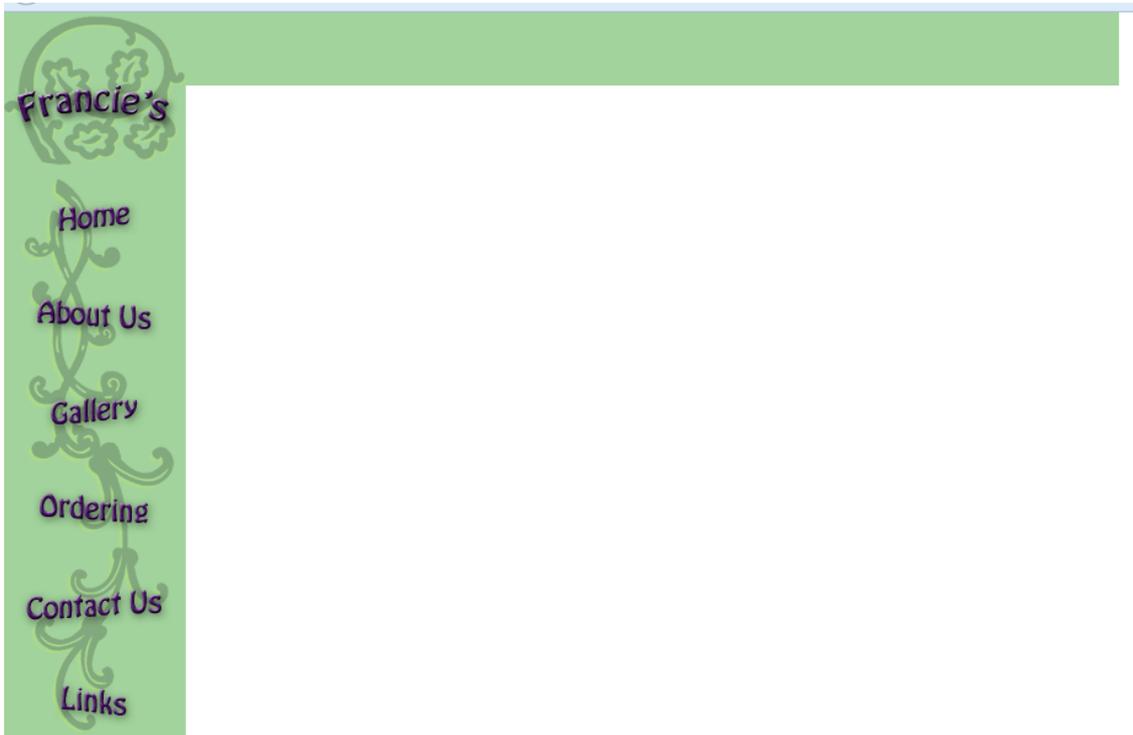
12) When you are prompted to provide alternate text for the image, type *Francie's Flowerpots Logo*.



13) Add images for each of the remaining DIVs in the navigation as follows. You may need to scroll down the page to see the last couple when you are adding the images.

Image name	DIV to put it in	Alternate text
nav_home.gif	nav_home	Francie's Flowerpots Homepage
nav_about.gif	nav_about	About Francie's Flowerpots
nav_gallery.gif	nav_gallery	Francie's Flowerpots Gallery
nav_ordering.gif	nav_ordering	Order from Francie's Flowerpots
nav_contact.gif	nav_contact	Contact Francie's Flowerpots
nav_links.gif	nav_links	Links to Francie's favourite websites

14) Preview the page in a web browser. It should look like the following example.



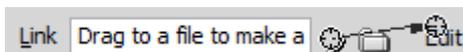
The last thing we will do with our navigation is to turn each of the navigation images in to a link that points to the relevant page.

15) Select the Home image in the navigation bar.



Remember what you have learned in previous exercises and create a link to the file index.html

**Hint** you can use the point to file button next to the Link property in the properties panel and then point to the index.html file in the File List.



16) Create links for each of the remaining images as shown below.

**Image name**

nav\_about.gif  
 nav\_gallery.gif  
 nav\_ordering.gif  
 nav\_contact.gif  
 nav\_links.gif

**Links to**

about.html  
 gallery.html  
 ordering.html  
 contact.html  
 links.html

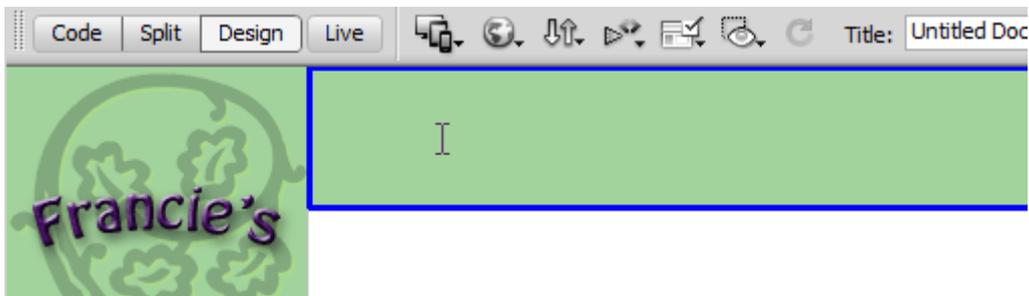
**Note** Each filename will have ../ before it in the properties. That is because our template file is stored in a folder below the other files. The ../ Tells Dreamweaver to refer to the folder one level up for those files.

17) Save changes to the file.

## Exercise 6 – Editable Regions

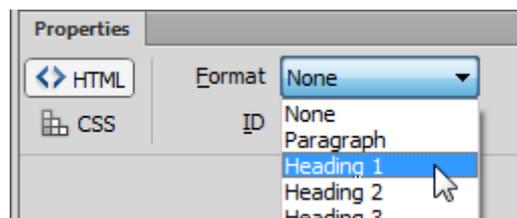
We have almost finished setting up our template. All that remains is to specify areas of the template that can be edited on pages that we create using this template. All remaining areas on the page (mostly the page structure and navigation bar) will not be able to be edited without opening the template itself.

1) Click inside the Heading area of the page.



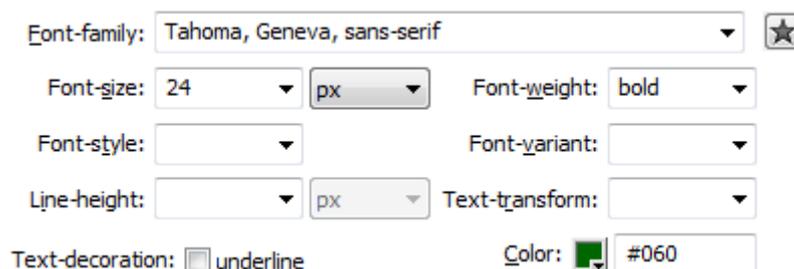
We want this part of the page to be formatted using the **H1** (heading 1) HTML tag.

2) In the **Properties Panel** change the format to **Heading 1**.

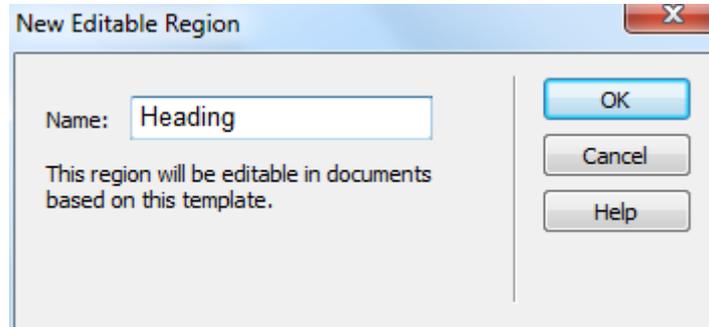


We'll create a CSS Style rule for our Heading 1.

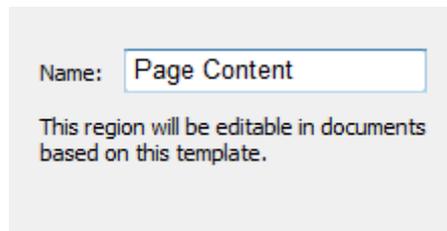
- 3) Click the **New CSS Rule** button  at the bottom of the **CSS Rules** panel.
- 4) Set the **Selector Type** option to **Tag** and set the **Selector Name** to **h1**.
- 5) In the **Rule Definition** option, make sure **styles.css** is selected so that the new rule will be in that file.
- 6) Click **OK** to continue to the style properties.
- 7) Set the following options in the **Type** category.



- 8) In the **Block** category change the **Text-align** property to **center**.
- 9) Click **OK** to complete the style rule. We won't see much change yet until we put some text in that space. Instead of typing text now, we will add an Editable Region. Then when we create a page based on this template, any text we type in that editable area will be formatted using that heading 1 style rule.
- 10) From the **Insert** menu select **Template Objects** and then select **Editable Region**.
- 11) Enter **Heading** for the name of the editable region and click **OK**.



- 12) Click inside the Content area (the white area).
- 13) Insert another Editable Region called **Page Content** and click **OK**.

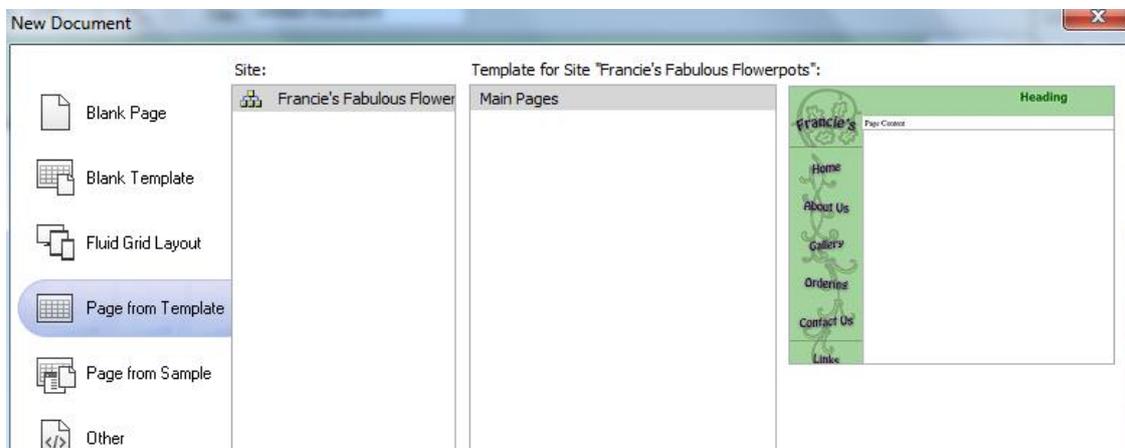


- 14) From the **File** menu select **Save All**.

## Exercise 7 – Using the Template

We have already created blank documents for the other pages to test our links, but we won't keep those ones. Instead, we'll create new pages using the template and then save the new pages over the top of the old ones.

- 1) From the **File** menu select **New**. **[Ctrl] [N]**
- 2) When the new file dialog appears, select the **Page From Template** option on the side.



- 15) Any Dreamweaver sites you have created will be listed in the middle section. Make sure your Francie's Flowerpots site is selected and the Main Pages template is selected.
- 16) Click **Create** to make a new page based on the template.

17) Move your mouse outside the editable areas and a cross will appear on your mouse.

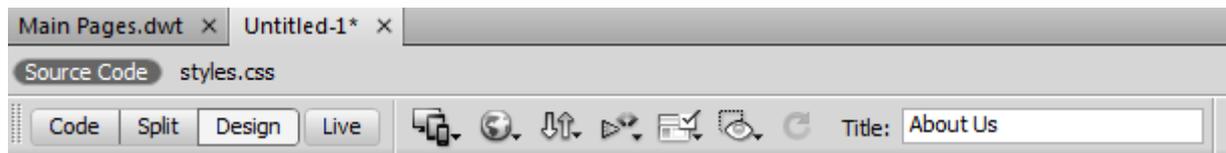


- 3) Click in the editable area for the heading. Delete the text that is there and type *About Us*. Format it as a level 1 heading.
- 4) Click in the *Main* editable area. Replace that text with the following.

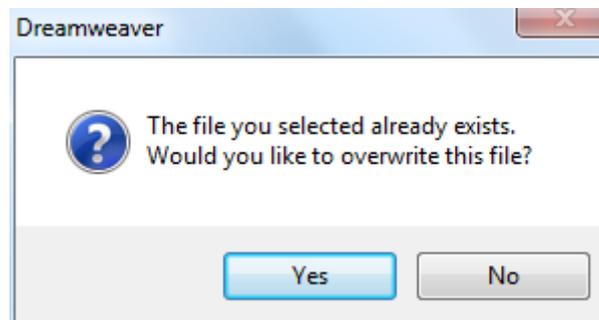
*Francie's Flowerpots is a business located in Western Australia. All of our pots are handmade using quality materials and techniques. We deliver to anywhere in the world. A sample of our range of pots can be viewed from the Gallery page. Other types of pots or custom made pots are available on request. Visit our Contact page to request a complete catalogue or more information about what's in the gallery.*

*Francie began making her own pots in the early 90s and began selling them to interested friends as a hobby. Since then the business has grown and we now have a team of skilled potters working to please our customers. We're certain that we have just the right pot for you.*

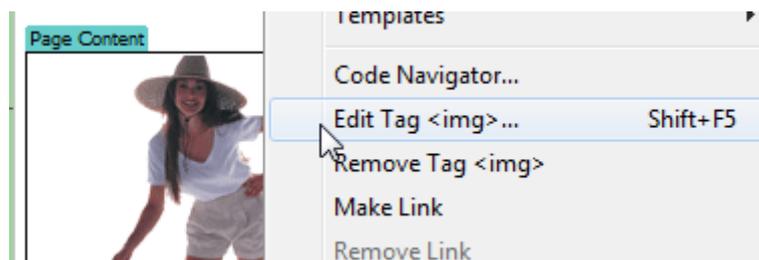
- 5) Change the **Title** of the page to *About Us*.



- 6) Save the page as *about.html*. When you are prompted to replace the existing file, click **Yes**.



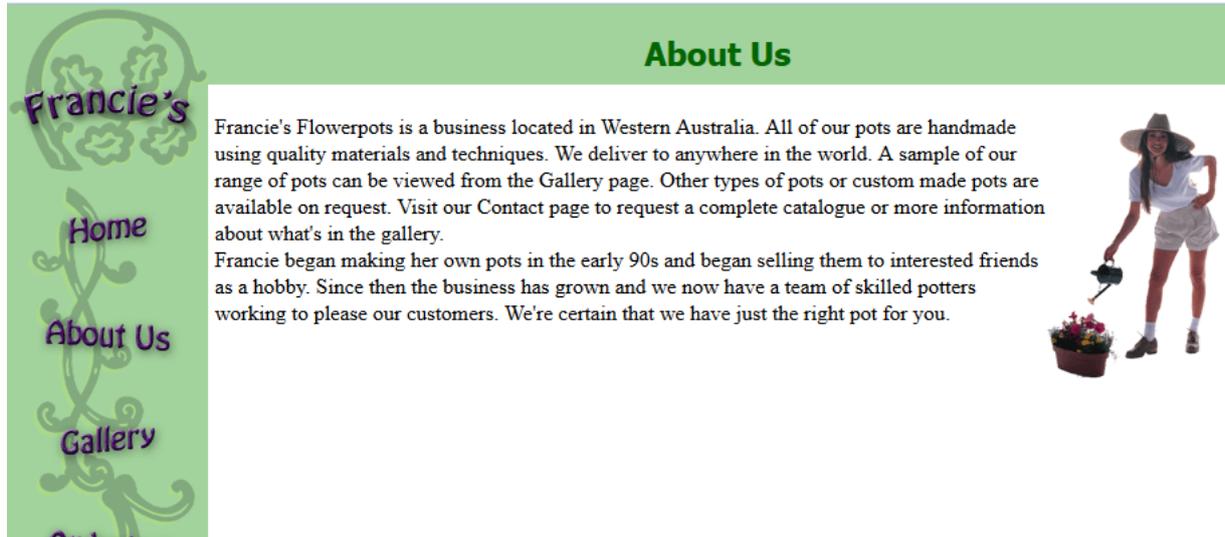
- 7) Click at the beginning of the first line (before the text *Francie's Flowerpots is a business*) and insert the image *about\_francie.gif*.
  - 8) For the alternate text type *About Francie*.
  - 9) Make sure the picture is selected. We will need to edit the Image tag to make it align to the right of the page. We can do it by editing the HTML directly or we can do it by using the tag editor.
- 18) Right click on the image and select Edit Tag <img>



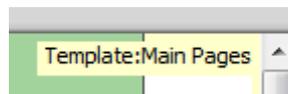
- 19) Change the **Alignment** option to **Right** and click **OK**.



- 10) You can also right-click the image and then select the alignment options from the shortcut menu.
- 11) Save and preview the document.



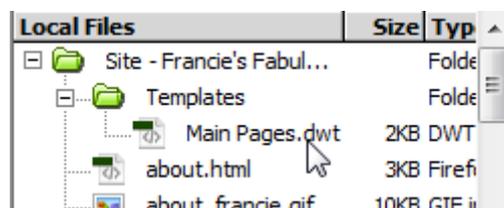
**Note** in the top right corner of the editing window, you might see a small tip indicating that this page is based on a template.



### Exercise 8 – Updating the Template

When we make changes to a template, it will also change any pages based on that template.

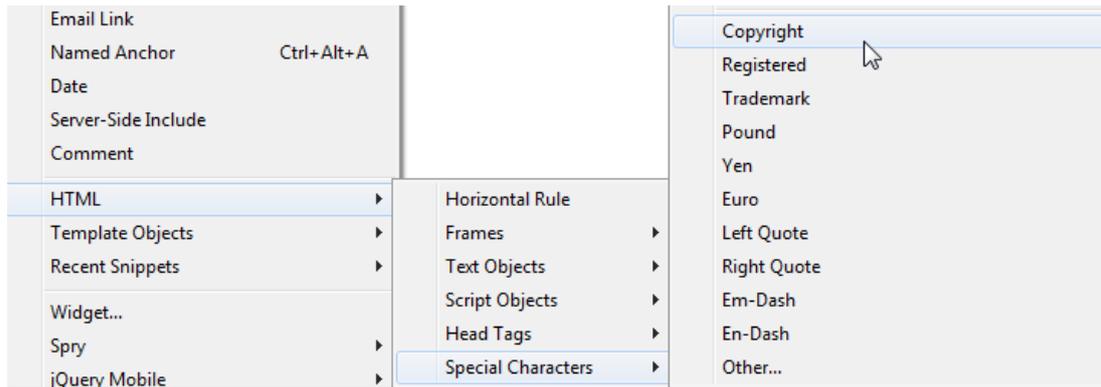
- 1) Close the about.html file by clicking on the close button in the top right corner of the editing window or by pressing **[Ctrl] [W]**.
- 2) If you don't still have the template open, open it. Remember that the template is saved in a folder underneath your other files. If you have it showing in your file list you can double click it to open it. You can also use the recent files list under the **File menu**.



- 3) Make a new blank line underneath the Page Content editable region.
- 4) We will add a small text navigation line similar to the one on the home page. Enter the text shown below.
- 5) About Us - Gallery - Ordering - Contact Us - Links
- 6) Create links to the appropriate pages so that it looks like the example below.
- 7) [About Us](#) - [Gallery](#) - [Ordering](#) - [Contact Us](#) - [Links](#)
- 8) Add another blank line under the links line.

We want the line to start with a copyright symbol.

9) From the **Insert** menu select **HTML, Special Characters** and **Copyright**.



10) After the copyright symbol type *Francie's Flowerpots* followed by the current year.

[About Us](#) - [Gallery](#) - [Ordering](#) - [Contact Us](#) - [Links](#)

© Francie's Flowerpots 2014

If you check the HTML, you will see that the code **&copy;** has been used to add the copyright symbol.

We'll create a new style rule which will be used for the page footer.

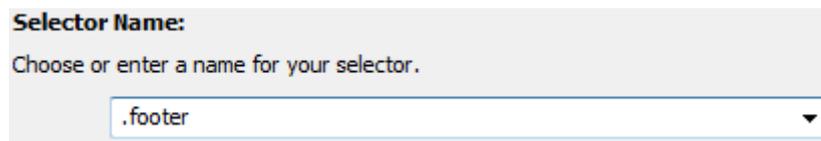
11) Click the **New CSS Rule** button at the bottom of the **CSS Styles** panel.



12) For the **Selector Type** choose **Class**.



13) In the selector name type **.footer** (including the full stop before it). A class is a special type of style rule that can be applied wherever we need it. Class names begin with a full stop.



14) Make sure the **Rule Definition** is set to save in **styles.css** and click **OK**.

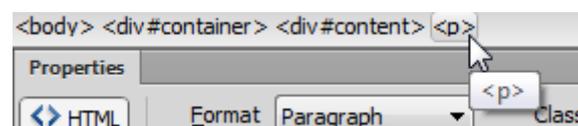
15) In the **Type** category set the **Font-size** to **smaller**.

16) In the **Block** category set the **Text-Align** to **center**.

17) Click **OK** to create the new style rule.

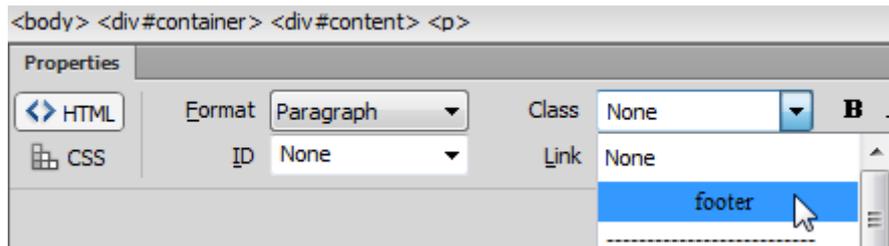
18) Click inside the links line.

19) At the bottom of the design window, make sure the **<p>** tag properties are showing. You can click the **<p>** to make sure.



We can now select a property that will tell this paragraph to use the footer style rule.

20) Change the **Class** property to **footer**.

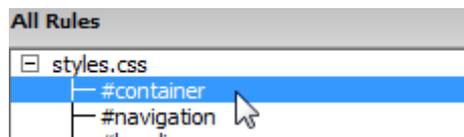


20) Do the same for the line with the copyright symbol.

In the **HTML** you will see a **class** attribute added to the **P** tag. `<div id="content">`

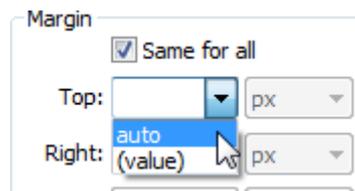
The last change we will make is a change to the container CSS rule. We will change it so that the container is centred on the page.

21) Double click the #container style rule in the CSS Styles panel to edit the style.



22) In the **Positioning** category change the **Position** from absolute to **relative**.

23) In the **Box** category set the **Margin** property to auto with the **Same for all** option ticked.

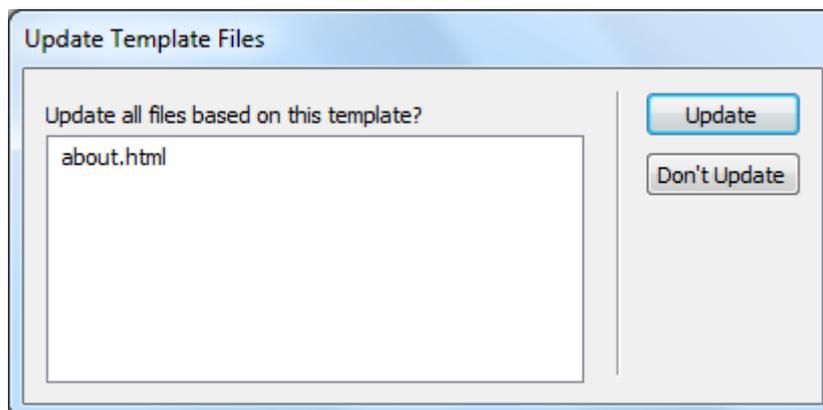


This will mean that any blank space around the edge of the container will be distributed evenly on either side, resulting in a centred position. The positioning needs to be set to relative to allow this to work as absolute positioning will fix it in place.

24) Click **OK** to complete the change.

25) From the File menu select Save All.

Since you have modified the template, you will be asked if you want to update pages based upon that template.



26) Click the **Update** button to confirm. Another message will appear once the update is done. You can close that message.

27) Open the about.html page. You will see that the changes to the template have been applied to this page.

28) Preview the page in your web browser.

## About Us

**Francie's**

Home

About Us

Gallery

Ordering

Contact Us

Links

Francie's Flowerpots is a business located in Western Australia. All of our pots are handmade using quality materials and techniques. We deliver to anywhere in the world. A sample of our range of pots can be viewed from the Gallery page. Other types of pots or custom made pots are available on request. Visit our Contact page to request a complete catalogue or more information about what's in the gallery.

Francie began making her own pots in the early 90s and began selling them to interested friends as a hobby. Since then the business has grown and we now have a team of skilled potters working to please our customers. We're certain that we have just the right pot for you.

[About Us](#) - [Gallery](#) - [Ordering](#) - [Contact Us](#) - [Links](#)

© Francie's Flowerpots 2014



### Shortcuts Learned

[Ctrl] [N]	Create a new document
[Ctrl] [Alt] [V]	Insert an editable region in to a template
[Ctrl] [W] or [Ctrl] [F4]	Close the current document
[Ctrl] [Q] or [Alt] [F4]	Exit Dreamweaver