GIMP® WEB 2.0 MENUS

Web 2.0 Menus: Horizontal Navigation Bar with Dynamic Background Image

WEB 2.0 MENUS: HORIZONTAL NAVIGATION BAR DYNAMIC BACKGROUND IMAGE

Before you begin this tutorial, you will need to copy the `menu_1_orange` folder in the `PortableApps/GIMP/buttons/images/ folder` and rename the folder `menu_1_blue`.

The folder contains three images in an .xcf format:

`btn_background.xcf`

`hover_button.xcf`

`nav_arrow_orange.xcf`

and the same three images in a .png format. The .xcf files contain the entire image and the .png files are saved as 1-pixel wide images.

You will need to change the color of the `hover_button` to a blue gradient. You can use the lighter and darker blues in the image for the foreground and background colors of the blend (gradient).

Change the `nav_arrow_orange` button to the lighter blue background and save both images with a .png extension.

You will also need to copy the banner image named `tech_services.png` that you saved in the `PortableApps/GIMP/banners/images` folder to the `menu_1_blue` folder.

Step 1: In a text editor, begin a new document and save in the `menu_1_blue` folder as `peavey_tech_services.html`.

Add the following html code to create a Web page with a banner image and an unordered list.

```
<html>

<head>

<title></title>

<style>


</style>

</head>

<body>

<p class = "center"><img src = "tech_services.png" alt = "banner" /></p>

<div id="menu">
```

```
<ul>

     <li><a href="index.htm">Home</a></li>

     <li><a href="about_us.htm">About Us></a></li>

     <li><a href="services.htm">Services</a></li>

     <li><a href="contact.htm">Contact U</a></li>

     <li><a href="blog.htm">Blog</a></li>

  </ul>

</div> <!-- ends the menu division -->

</body>

</html>
```

Step 2: Between the beginning and ending <style> tags in the head section, add a style to horizontally center the banner image.

```
p.center

    {

    text-align: center;

    }
```

FORMATTING THE NAVIGATION BAR

The navigation bar will be created by repeating the `tech_services_btn.png` image as many times as necessary to fit the width of the menu.

Step 1: In the html code, the code for the navigation bar is nested inside a division named `menu` so you will want to define a style for the menu division.

Below the `p.center` style in the style section add the following style to set the width of the navigation bar to the width of the banner image (800 pixels), the height to 60 pixels, the background image to the tech_services_btn.png image, and the margins to auto.

```
#menu

    {
    /* defines the width of the navigation menu */
    width: 800px;
    /* defines the height of the menu for Firefox */
    height: 60px;
    /* defines the image used for the menu bar */
    background: url(tech_services_btn.png) repeat-x;
    /* horizontally centers the navigation menu */
    margin: auto;
    }
```

Note: One of the questions you may be asking yourself is why is the selector `background` instead of `background-image`?

If you answered, "This is a shortcut style and repeat-x would not be a style for background-image," you would be correct.

Step 2: Below the #menu definition in the style section, add a style for the unordered list in the menu division that sets the width of the combined link text to 750 pixels, centers the link text, and removes the bullets.

#menu ul

{

/* defines the width of the combined link text */

width: 750px;

/* centers the text across the bar */

margin: 0 auto;

/* removes the bullets */

list-style:none;

}

The width of 750 pixels will be changed later depending on the number of characters in the text for each list item and the amount of padding defined

Step 3: Style each list item horizontally floating at the left, set the height to the height of the button image, define the font size and weight of the link text, define the line height, and define the padding.

#menu li

{

/* displays the list items horizontally rather than vertically */

display: inline;

```
/* allows you to center the button text later */

  float: left;

/* required by Firefox to display the background image */

  height: 61px;

/* defines the font size of the button text */

  font-size: 16px;

/* defines the weight of the button text */

  font-weight: bold;

/* set to height of button to vertically center the button text */

  line-height: 61px;

/* defines a 20 pixel on left and right of the combined link text*/

  margin: 0 20px;

  }
```

Home   About Us   Services   Contact Us   Blog

Step 4: Use the `a` selector to change the font color, remove the underline, and add a 25-pixel padding to the left and right of the link text.

```
#menu li a

  {

  /* defines the link text color */

  color: #485cd2;

  /* removes the underline from the link text */

  text-decoration: none;

  /* defines a padding of 0 pixels on top and bottom of link text */
```

/*defines a padding of 25 pixels on the left and right of link text */

padding: 0 25px;

/*you can only float block –level elements so you need to make the text block */

display: block;

/*This will allow the second list item to display to the right of the first, and so forth */

float: left;

/*required for Firefox */

height: 61px;

}

Step 5: use the `a:hover` selector to display a background image when the mouse is "hovered" over the link text and change the font color to white.

#menu li a:hover

{

/* repeat the tech_services_hover.png image **across** the link text and padding */

background: url(tech_services_hover.png) repeat-x;

/* define the link text color */

color: white;

}

Step 6: Because you cannot apply two background images (the button and the navigation arrow to the same selector, you will need to add a tag to the link text in the html code.

        \
**\Home**\\

        \
**\About Us**\\

        \
**\Services**\\

        \
**\Contact Us**\\

        \
**\Blog**\\

Step 7: Add a style to the `a:hover span` selector that displays a navigation arrow when the mouse is hovered over the link text.

Again, it is important that you check the display in both browsers as the display: block style is required by Firefox to move the arrow to the top of the button, and the float: left style is required by I.E. to display the image only behind the hovered button.

#menu li a:hover span

    {

    `/* displays the arrow image once at the top of the button */`

    background: url(tech_services_nav_arrow.png) no-repeat top;

    `/* move the arrow to the vertical top of button in Firefox */`

    display:block;

    `/* displays the background image behind one the hovered button */`

    float:left;

```
        }
```


Step 8: Often the Home button displays the background image without the navigation arrow even when it is not in the hover state.

To create this effect, you will need to add a class to the hyperlink code in the html file and add a style to the class.

    Home

Step 9: Add a style to the home class that is identical to the `#menu li a` style with the additional button background image.

#menu li .home

    {

    display: block;

    float: left;

    height: 61px;

    background: url(tech_services_hover.png) repeat-x;

    color: white;

    text-decoration: none;

    padding: 0 25px;

    }

The Home button displays the background image but only displays the navigation arrow when it is in the hover state.

Step 10: As an example, go to the #menu li a style and change the padding to

```
padding: 0 5px;
```

Save and test in the browser. You will see that the link text is not centered on the navigation bar. It is difficult sometimes because of font sizes and font weights to calculate the exact width for the `#menu ul` style, but you know that in this instance you need a smaller width, so let's try 650 pixels (5 buttons x 20 pixels = 100 and 750 – 100 = 650).

You will see that this is not quite perfect – there is slightly more space at the right of the navigation bar.

Experiment – I ended up with about 575 pixels.

Now you can UNDO until the padding is back to 25 pixels and the width is back to 750 pixels.

BE SURE TO CHECK THE DISPLAY IN BOTH FIREFOX AND I.E. TO BE SURE THAT ALL OF THE STYLES WORK PROPERLY IN BOTH BROWSERS.

CHANGING THE EMBEDDED STYLE TO A CASCADING STYLE SHEET

Step 1: Copy the styles between the beginning and ending `<style>` tags and paste them into a new blank document and save the document in the folder as `peavey_technology_services.css`.

Step 2: In the html file, add a link to the cascading stylesheet below the ending `<title>` tag:

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

Step 3: View the Web page in both browsers to check the formatting to be sure it displays correctly.

Step 4: Delete the beginning and ending style tags in the html document.

Step 5: Save both documents in the `menu_1_blue` folder.

At the Design tab, you will be coding the Home page for the Peavey Technology Web site.