

Basic Web Page Construction Assignment

HTML

1. Open a new html file – name it myPage.htm
2. Insert the basic required html elements:

```
<DOCTYPE html>
<html>
    <head>
        <title>My Page</title>
    </head>
    <body>
    </body>
</html>
```
3. Go back and insert four elements into the body - <header>, <sidebar>, <main>, <footer>
4. Within the <header> element, add a <div> to hold the logo and the tagline. Find/download an image to use as a logo. Make the logo clickable by inserting in an <a> tag.
5. Also within the <header> element, add a <nav> element to hold the navigation panel.
6. Within the <nav> element, add an un-ordered list with several items. Make each item clickable by inserting it within an <a> tag. Have at least one of the list items contain another un-ordered list with several items to act as a sub-menu.
7. Within the <sidebar> element, add an un-ordered list with a few items in it. You can make the items clickable by using the <a> tag if you like.
8. Add an <article> element to the <main> element and fill it with “Lorem Ipsum” text (do a search to find text to copy/paste). Add some <h1>, <h2>, <p> tags to format the text.
9. Add a copyright message into the <footer> element.
10. View the completed HTML in a browser to make sure that all of the elements are visible.

CSS

1. Open a new css file and name it myPage.css.
2. Edit the html to add a link to read the css file:

```
<link href='myPage.css' type='text/css' rel='stylesheet'>
```
3. Edit the css to create style rules to layout the header and navigation. Make the header scale to the full width of the browser window. Make the logo scale to display a reasonable size even when the window is resized. Add a background color, image or gradient to the header.
4. Position the tagline text at the right of the browser window using absolute positioning (remember that absolute positioning requires a parent within the html hierarchy to be positioned using a method other than static. You can use relative positioning on the <header> element with a top and left equal to 0 to satisfy this rule).
5. Scale the <nav> element to fit the width of the header in which it is contained. Add a gradient to the background to style the panel and make it look attractive.

6. Style the un-ordered list to remove the bullets. Remove the underscores (text-decorations) from the links. Use float left to have the buttons stack up against one another. You will need to set a height for the nav panel to keep it from collapsing because all of its contents are floating elements. Add line dividers between nav buttons.
7. Style the sub-menus to add lines between the elements and background colors.
8. Set the visibility of all sub-menus to invisible.
9. Add a rule to the sidebar to float it right and size it to fit. Add borders and background colors to separate it from the rest of the page.
10. Anchor the footer to the bottom of the page using fixed position. Size it to fit the full width and add background color to make it stand out.
11. Add links to google fonts if you want to add any downloadable fonts to your page.
12. Display a scrollbar in the main window if the content overflows the provided space.

Javascript/JQuery

1. Add a link to the html to include the jquery source:
`<script type='text/javascript'
src='http://ajax.googleapis.com/ajax/libs/jquery/1.11.0/jquery.min.js'></script>`
2. Create a new js file and name it myPage.js
3. Add a link to the html to include the myPage.js source:
`<script type='text/javascript' src='myPage.js'></script>`
4. Add a jquery document ready handler to the js file.
5. Write mouseenter handlers to animate the nav panel buttons by changing their background and color styles. (Don't forget to include mouseleave handlers to restore the normal appearance).
6. Add some jQuery calls into the mouseenter handler to toggle the sub menu visibility.
7. If you have time, add some jQuery to make the logo interesting when moused over.
8. Add more animations to make the page interesting.