



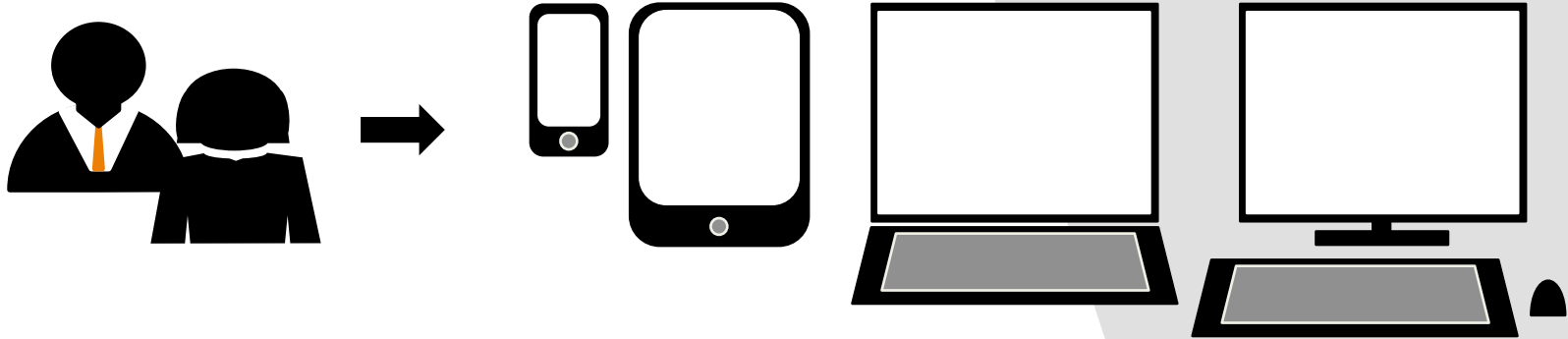
Responsive Web Design

An approach to create a responsive web site with menus and icons that is entirely responsive utilising JQuery

The Design Requirement...
A Simple Site Layout...
The Basic HTML...
The JQuery...
The Final Result...



The Design Requirement

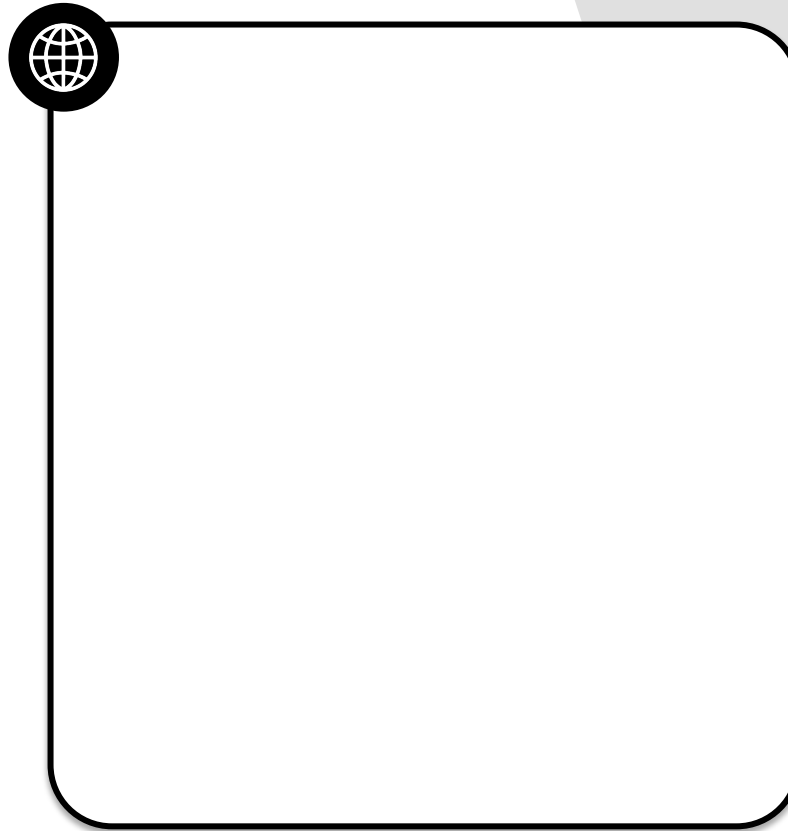


A significant challenge in web design whether for a web site or a web application is to account for the plethora of devices that people could use to view the site or application. In particular the ability to control the user experience making it usable and viewable no matter what size of screen the device they are using is set at.



A Simple Site Layout - 1

Lets begin with a simple layout – from here on in this presentation the following diagram denotes the browser window:



A Simple Site Layout - 2

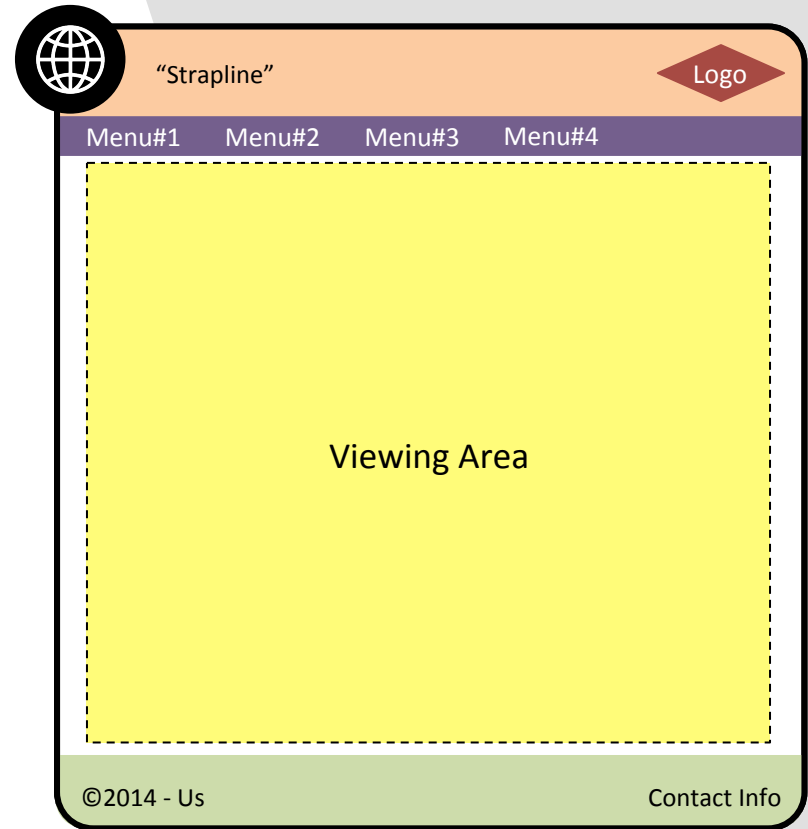
Within our browser window we will want to position the main aspects of a site.

This usually begins with:

- The placement of a banner bar;
- With the logo of your company on either the right or left of the screen;
- Perhaps a strapline welcoming visitors or with the mission statement for your company or organization;
- This is shortly followed by a menu of some kind either on a single line or as a dropdown;
- Then a footer at the bottom with copyright and contact information;
- Leaving you the center screen as viewing area ...

And there you have it – a simple web site set up!

Note: The awful colors are so we can see where every container is ...

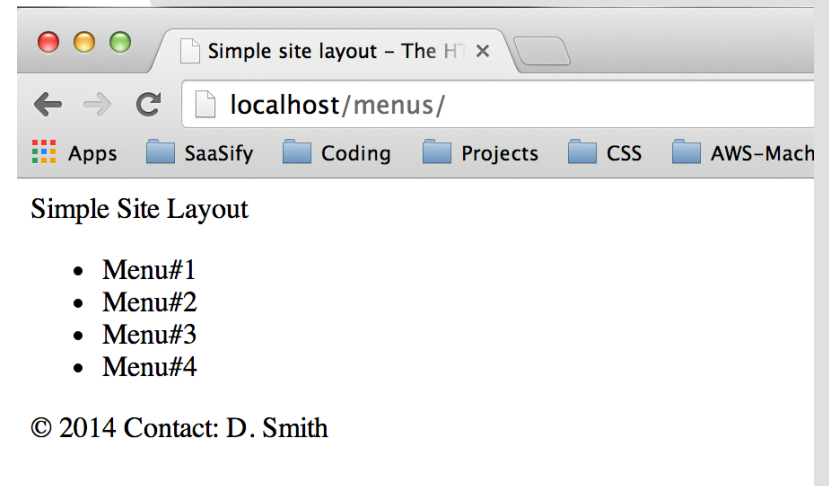


A Simple Slide Layout – 3

Lets have a look at the HTML that generates this simple site ...

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Simple site layout - The HTML</title>
  </head>
  <body>
    <!-- Lets first define the total container -->
    <div class="mainContainer">
      <!-- First up in the container is the heading section -->
      <div class="mainHeader">
        <!-- The title for the site -->
        <span class="mainTitle">Simple Site Layout</span>
        <!-- The Logo for the simple site -->
        <img src="" />
      </div>
      <!-- Next up the menu area -->
      <div class="mainMenu">
        <nav>
          <ul>
            <li>Menu#1</li>
            <li>Menu#2</li>
            <li>Menu#3</li>
            <li>Menu#4</li>
          </ul>
        </nav>
      </div>
      <!-- Next the viewing area -->
      <div class="mainView"></div>
      <!-- Finally the footer section -->
      <div class="mainFooter">
        <!-- The copyright information -->
        <span class="mainCopyright">&#169; 2014</span>
        <!-- The contact information -->
        <span class="mainContactInfo">Contact: D. Smith</span>
      </div>
    </div>
  </body>
</html>
```

The resulting browser view of the HTML



Not very web like ... we need styles!



A Simple Site Layout - 4

Lets start building in the styles that will really generate this simple site ...

First lets add a link in the header of the original HTML file to include the styles ..

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <link rel='stylesheet' href='css/simplesite.css' />
    <title>Simple site layout - The HTML</title>
  </head>
```

Then we'll open a new file `simplesite.css` and start formatting the site ...

```
/* -----
   Set up a font for the Simple Site
   Check out www.google.com/fonts for
   examples of all kinds of fonts...
   ----- */

@import url(http://fonts.googleapis.com/css?family=Cabin);
```

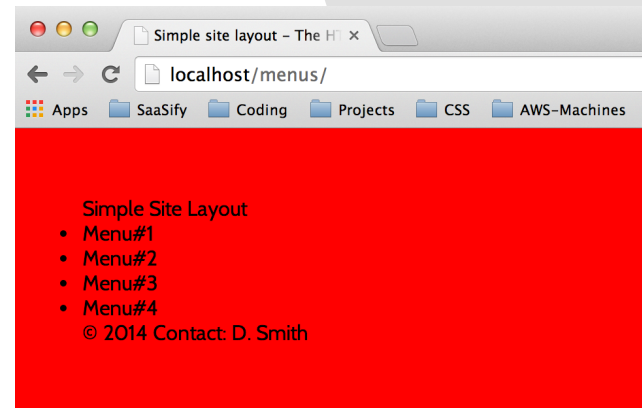
```
/* -----
   We will ensure there are no margins or padding set from the
   outset ...
   ----- */

* {
  margin:0;
  padding:0;
}

/* -----
   And to create a major contrast we'll use the font we loaded above
   set a background color for the whole page of red, a margin of 5% to
   create a border type effect, a font color black and no overflow
   (scrolling up and down, left or right) ...
   ----- */

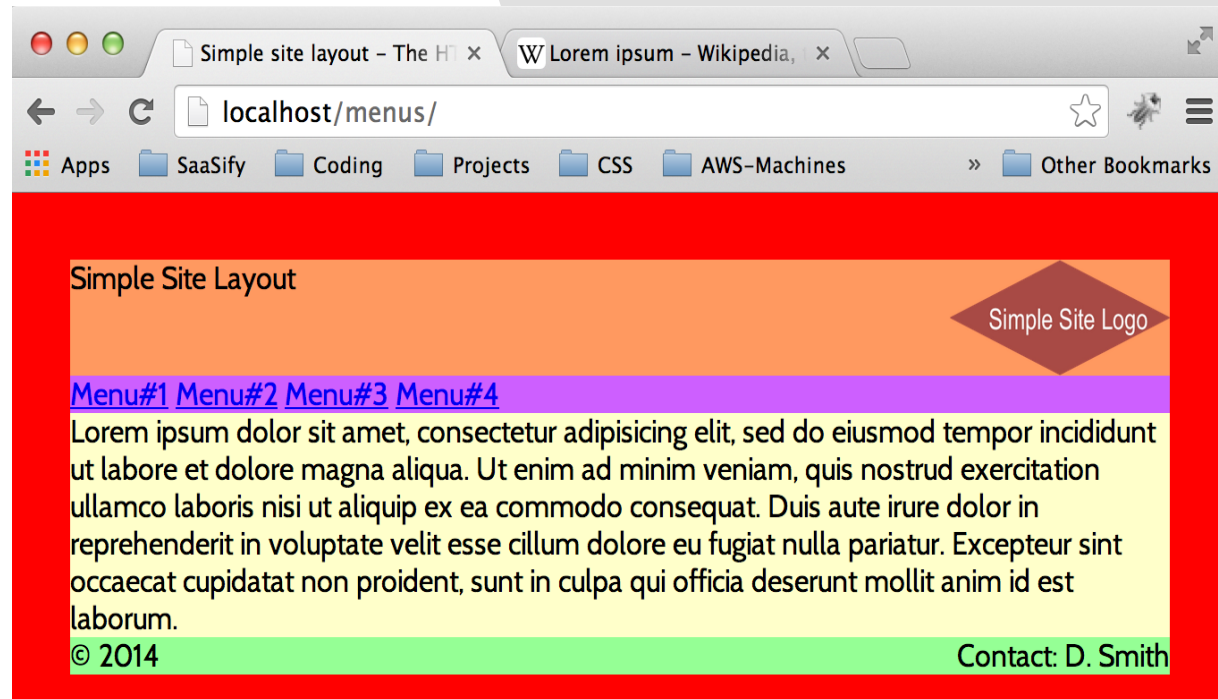
body {
  background-color: #ff0000;
  color:#000;
  font-family: 'Cabin', sans-serif;
  margin: 5%;
  overflow-x: hidden;
}
```

The result in our browser ...



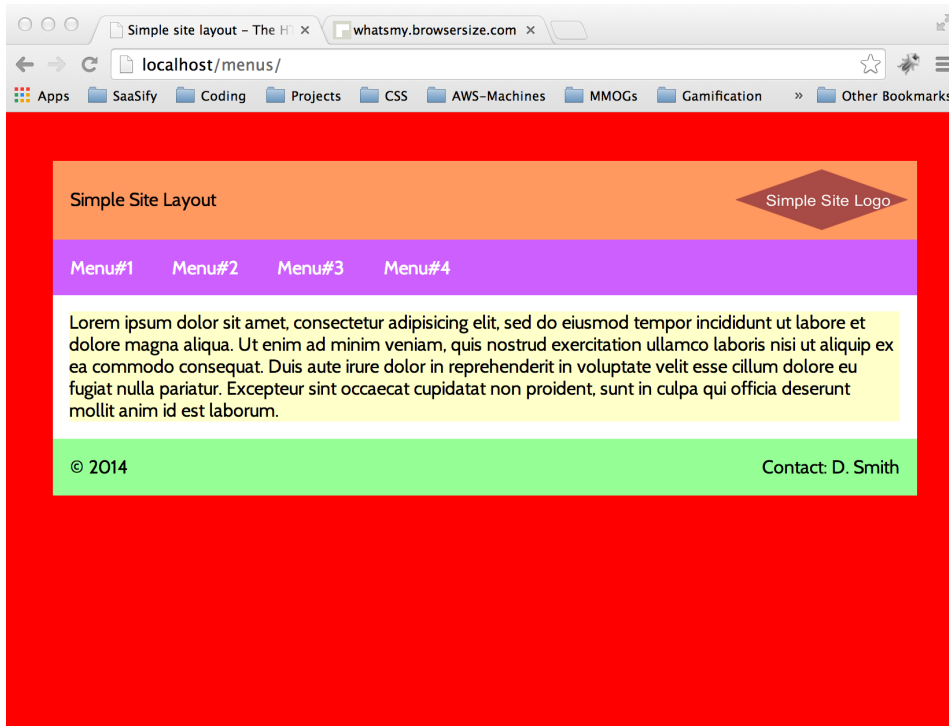
A Simple Site Layout - 5

We continue adding the main container, the heading section, the strapline, the image, the menu section, the main view, and the footer to get to this ...



A Simple Site Layout - 6

After a little bit more fiddling with CSS styles to pad items and space things it yields a visible site such as this ..



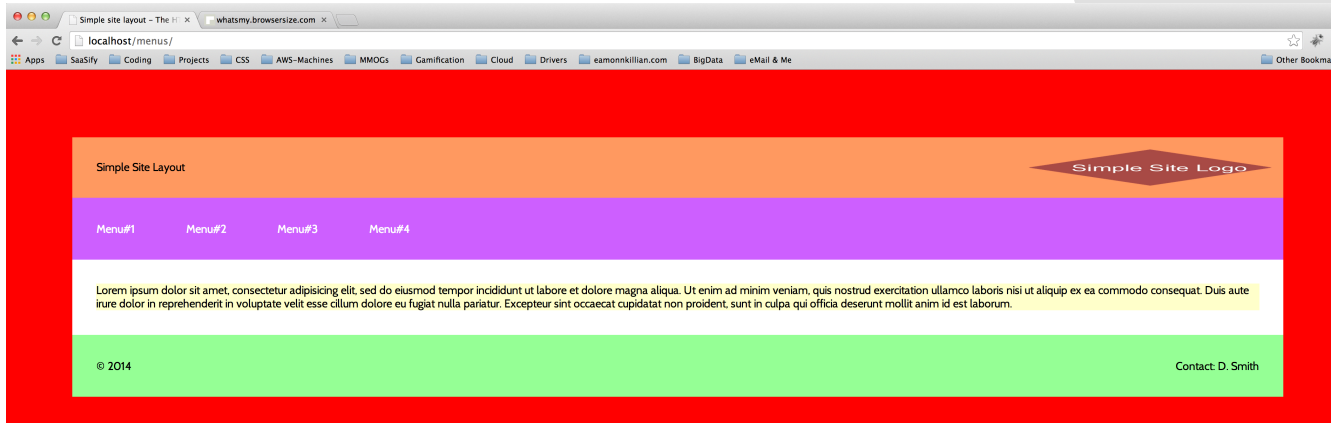
```
/* -----  
The main container  
----- */  
  
.mainContainer {  
  position: relative;  
  float: left;  
  top: 0;  
  width: 100%;  
  height: 100%;  
  background-color: #fff;  
}  
  
/* -----  
The heading section  
----- */  
  
.mainHeader {  
  position: relative;  
  float: left;  
  top: 0;  
  width: 100%;  
  height: 12%;  
  background-color: #ff9966;  
}  
  
.mainTitle {  
  position: absolute;  
  top: 50%;  
  margin-top: -10px;  
  padding-left: 2%;  
}  
  
.logoImage {  
  position: relative;  
  float: right;  
  width: 20%;  
  height: 10%;  
  padding: 1%;  
}  
  
/* -----  
The menu section  
----- */  
  
.mainMenu {  
  position: relative;  
  float: left;  
  width: 100%;  
  height: 12%;  
  background-color: #cc66ff;  
}  
  
nav ul li {  
  display: inline-block;  
  padding: 2%;  
}  
  
/* -----  
The menu section  
----- */  
  
.mainView {  
  position: relative;  
  float: left;  
  width: 96%; // make this the 100% minus the margin either side  
  height: 50%;  
  margin: 2%;  
  background-color: #ffffcc;  
}  
  
/* -----  
The footer section  
----- */  
  
.mainFooter {  
  position: relative;  
  float: left;  
  width: 100%;  
  height: 12%;  
  background-color: #99ff99;  
}  
  
.mainCopyright {  
  position: relative;  
  float: left;  
  padding: 2%;  
}  
  
.mainContactInfo {  
  position: relative;  
  float: right;  
  padding: 2%;  
}
```



A Simple Site Layout - 7

Not bad actually (in terms of what we were seeking) but the problems begin when we start to re-size the browser window ..

Wide and tall thin browser windows:



Problems:

- 1) Logo is responding but ugly;
- 2) Fonts are the same size rather than scaling to the view size;
- 3) My middle section even when set to a % is only the height necessary for the text rather than a set height ...

Bottom Line “I want more control!”



Introducing JQuery

JQuery is a JavaScript library that helps designers manipulate everything in the DOM.

Whats the DOM?

DOM stands for Document Object Model which is the logical structure of everything that goes into building up your browsers view of the web page you have asked it to display. You can get a much better explanation of it here:

<http://www.w3.org/TR/DOM-Level-2-Core/introduction.html>



Using JQuery

First we need to tell the browser we are going to be using JQuery. We do this in the main header. There are two ways to link in the JQuery library:

1) Download it and include it;

You can download JQuery library straight from these sites:

- <http://jquery.com/download/>
- <https://developers.google.com/speed/libraries/devguide#jquery>

2) Link to it on the web;

You can link to the JQuery library online by adding this line into the header:

- `<script src=http://ajax.googleapis.com/ajax/libs/jquery/1.11.1/jquery.min.js></script>`

Lets link to it like this:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <link rel='stylesheet' href='css/simplesite.css' />
    <script src=http://ajax.googleapis.com/ajax/libs/jquery/1.11.1/jquery.min.js></script>
    <title>Simple site layout - The HTML</title>
  </head>
```



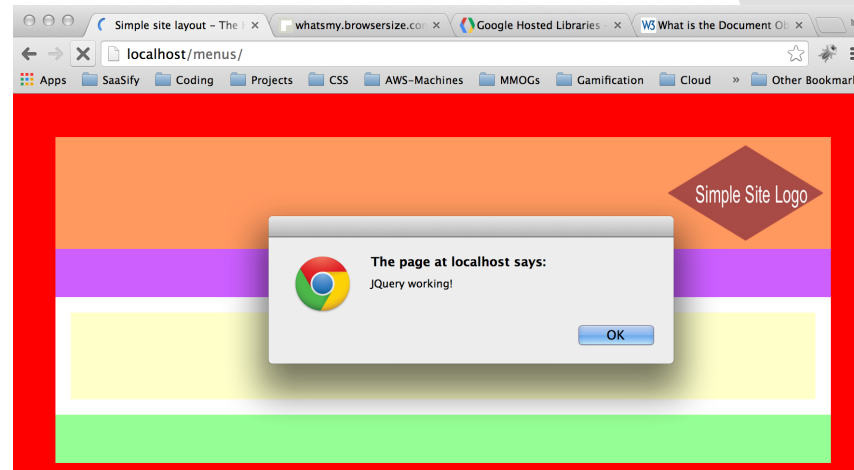
How do I know its working?

So we've added a javascript library but how do we know whether its working?
Simple – lets add a little test script in the body of our HTML ... like this:

```
<body>
  <script type="text/javascript">
    $(document).ready(function(){
      alert("jQuery working!");
    });
  </script>

  <!-- Lets first define the total container -->
  <div class="mainContainer">
```

... then reload the page and “voila!” you should have a message from JQuery!



Now for our responsive script

Clearly we could write all of the code straight in between the `<script>` tags, but that would probably make the HTML document look a bit messy and its better to extract it out to a separate javascript file.

So I create a `simplesite.js` file and link to it from the header ... so the HTML document now looks like this ...

```
<html>
  <head>
    <meta charset="UTF-8">
    <link rel='stylesheet' href='css/simplesite.css' />
    <script src=http://ajax.googleapis.com/ajax/libs/jquery/1.11.1/jquery.min.js></script>
    <script type="text/javascript" src="js/simplesite.js"></script>
    <title>Simple site layout - The HTML</title>
  </head>
```



First up in the script is ..

Lets wait until the full document is loaded and then find out what size we are dealing with ...

```
$(document).ready(function() {  
  
    //  
    // Lets get the window height and width attributes  
    //  
  
    var windowWidth = $(window).width();  
    var windowHeight = $(window).height();  
    var windowArea = windowWidth*windowHeight;
```

Lets also remove all mentions of Heights, Widths, Margins, Padding and Font-Sizes from the CSS file ... we will be dealing with all of them from within JQuery ...



Dealing with Fonts

On the basis of the window area we can define a specific font-size for the body which will flow to all of the components of the site ...

```
var fontSize = '10px';
if (windowArea < 99999) {
    $('body').css('font-size', fontSize);
}
if (windowArea > 100000 && windowArea < 249999) {
    fontSize = '12px';
    $('body').css('font-size', fontSize);
}
if (windowArea > 250000 && windowArea < 383999) {
    fontSize = '14px';
    $('body').css('font-size', fontSize);
}
if (windowArea > 384000 && windowArea < 467999) {
    fontSize = '16px';
    $('body').css('font-size', fontSize);
}
if (windowArea > 468000 && windowArea < 537599) {
    fontSize = '18px';
    $('body').css('font-size', fontSize);
}
if (windowArea > 537600 && windowArea < 614399) {
    fontSize = '20px';
    $('body').css('font-size', fontSize);
}
if (windowArea > 614400 && windowArea < 691199) {
    fontSize = '22px';
    $('body').css('font-size', fontSize);
}
if (windowArea > 691200 && windowArea < 786431) {
    fontSize = '24px';
    $('body').css('font-size', fontSize);
}
if (windowArea > 786432) {
    fontSize = '24px';
    $('body').css('font-size', '24px');
}
```

Now our fonts will change as the window size changes!



Centering the height of items ...

For items like the strapline, the logo, the menu items and the footer information we want to have these centered height-wise in their respective boxes ... clearly this means a margin or padding from the top of their container ... now we can figure this out precisely ...

Get the height of the item – in this case the strapline ...

```
var straplineHeight = $('.mainTitle').height();
```

Then get the height of its container ...

```
var mainheadingHeight = mainContainerHeight * .2;
```

Then calculate its margin-top and set it ...

```
$('.mainTitle').css('margin-top', (mainheadingHeight - straplineHeight) / 2);  
$('.mainTitle').css('margin-left', mainContainerWidth * 0.05);
```

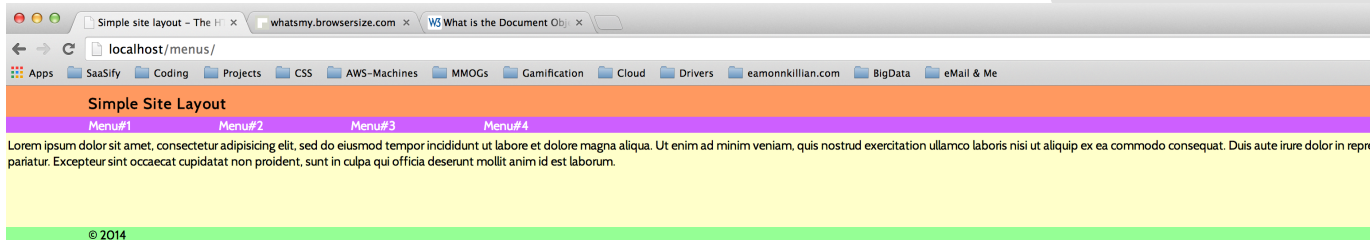
Repeat this process for other items that need to be centered from a height perspective!



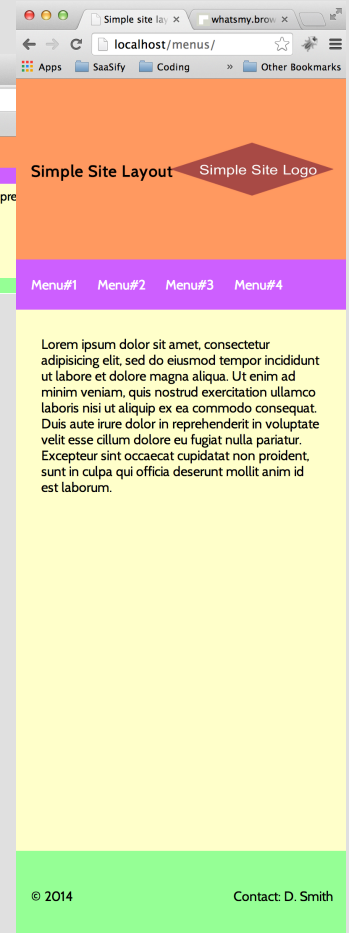
Keep going and test it!

Keep working through all the items on the site until you have the finished javascript ... then resize the window and hit reload to see the results ...

Wide and short



Tall and Narrow



Now for the real magic of JQuery ..

The final set is the one that really adds the magic ... we've tested by reloading the page which is okay but too static for our dynamic web world ... what we need is to run this re-sizing if the user ever changes the size of the window ... fortunately JQuery comes with an event handler that recognises if the page is re-sized! Fantastic! So lets make all of our code into a function and call it (a) At load time and (b) any time the page is re-sized!

```
function resizeMySite() {  
    "put all our code in here surrounded by the brackets!"  
}
```

Add this code to the javascript ...

```
//  
// Call the sizing function once the document is loaded ..  
//  
  
resizeMySite();  
  
//  
// Call the sizing function if the window is ever re-sized!!! ..  
//  
  
$(window).resize(function() {  
    resizeMySite();  
});
```



Hope this helps with RWD ... EJK

