J1

body {

font: 100%/1.4 Verdana, Arial, Helvetica, sans-serif;

background-color: #42413C;

margin: 0;

padding: 0;

color: #000;

}

/\* ~~ Element/tag selectors ~~ \*/

ul, ol, dl { /\* Due to variations between browsers, it's best practices to zero padding and margin on lists. For consistency, you can either specify the amounts you want here, or on the list items (LI, DT, DD) they contain. Remember that what you do here will cascade to the .nav list unless you write a more specific selector. \*/

padding: 0;

margin: 0;

}

h1, h2, h3, h4, h5, h6, p {

margin-top: 0; /\* removing the top margin gets around an issue where margins can escape from their containing div. The remaining bottom margin will hold it away from any elements that follow. \*/

padding-right: 15px;

padding-left: 15px; /\* adding the padding to the sides of the elements within the divs, instead of the divs themselves, gets rid of any box model math. A nested div with side padding can also be used as an alternate method. \*/

}

a img { /\* this selector removes the default blue border displayed in some browsers around an image when it is surrounded by a link \*/

border: none;

}

/\* ~~ Styling for your site's links must remain in this order - including the group of selectors that create the hover effect. ~~ \*/

a:link {

color: #42413C;

text-decoration: underline; /\* unless you style your links to look extremely unique, it's best to provide underlines for quick visual identification \*/

}

a:visited {

color: #6E6C64;

text-decoration: underline;

}

a:hover, a:active, a:focus { /\* this group of selectors will give a keyboard navigator the same hover experience as the person using a mouse. \*/

text-decoration: none;

}

/\* ~~this fixed width container surrounds the other divs~~ \*/

.container {

width: 960px;

background-color: #FFF;

margin: 0 auto; /\* the auto value on the sides, coupled with the width, centers the layout \*/

}

/\* ~~ the header is not given a width. It will extend the full width of your layout. It contains an image placeholder that should be replaced with your own linked logo ~~ \*/

.header {

background-color:#00F;

}

/\* ~~ These are the columns for the layout. ~~

1) Padding is only placed on the top and/or bottom of the divs. The elements within these divs have padding on their sides. This saves you from any "box model math". Keep in mind, if you add any side padding or border to the div itself, it will be added to the width you define to create the \*total\* width. You may also choose to remove the padding on the element in the div and place a second div within it with no width and the padding necessary for your design. You may also choose to remove the padding on the element in the div and place a second div within it with no width and the padding necessary for your design.

2) No margin has been given to the columns since they are all floated. If you must add margin, avoid placing it on the side you're floating toward (for example: a right margin on a div set to float right). Many times, padding can be used instead. For divs where this rule must be broken, you should add a "display:inline" declaration to the div's rule to tame a bug where some versions of Internet Explorer double the margin.

3) Since classes can be used multiple times in a document (and an element can also have multiple classes applied), the columns have been assigned class names instead of IDs. For example, two sidebar divs could be stacked if necessary. These can very easily be changed to IDs if that's your preference, as long as you'll only be using them once per document.

4) If you prefer your nav on the right instead of the left, simply float these columns the opposite direction (all right instead of all left) and they'll render in reverse order. There's no need to move the divs around in the HTML source.

\*/

.sidebar1 {

float: left;

width: 180px;

background-color:#0CF;

padding-bottom: 10px;

}

.content {

padding: 10px 0;

width: 780px;

float: left;

}

/\* ~~ This grouped selector gives the lists in the .content area space ~~ \*/

.content ul, .content ol {

padding: 0 15px 15px 40px; /\* this padding mirrors the right padding in the headings and paragraph rule above. Padding was placed on the bottom for space between other elements on the lists and on the left to create the indention. These may be adjusted as you wish. \*/

}

/\* ~~ The navigation list styles (can be removed if you choose to use a premade flyout menu like Spry) ~~ \*/

ul.nav {

list-style: none; /\* this removes the list marker \*/

border-top: 1px solid #666; /\* this creates the top border for the links - all others are placed using a bottom border on the LI \*/

margin-bottom: 15px; /\* this creates the space between the navigation on the content below \*/

}

ul.nav li {

border-bottom: 1px solid #666; /\* this creates the button separation \*/

}

ul.nav a, ul.nav a:visited { /\* grouping these selectors makes sure that your links retain their button look even after being visited \*/

padding: 5px 5px 5px 15px;

display: block; /\* this gives the link block properties causing it to fill the whole LI containing it. This causes the entire area to react to a mouse click. \*/

width: 160px; /\*this width makes the entire button clickable for IE6. If you don't need to support IE6, it can be removed. Calculate the proper width by subtracting the padding on this link from the width of your sidebar container. \*/

text-decoration: none;

background-color:#00C;

color:#000;

}

ul.nav a:hover, ul.nav a:active, ul.nav a:focus { /\* this changes the background and text color for both mouse and keyboard navigators \*/

background-color:#0FF;

color:#00C;

}

/\* ~~ The footer ~~ \*/

.footer {

padding: 10px 0;

background-color: #CCC49F;

position: relative;/\* this gives IE6 hasLayout to properly clear \*/

clear: both; /\* this clear property forces the .container to understand where the columns end and contain them \*/

}

/\* ~~ miscellaneous float/clear classes ~~ \*/

.fltrt { /\* this class can be used to float an element right in your page. The floated element must precede the element it should be next to on the page. \*/

float: right;

margin-left: 8px;

}

.fltlft { /\* this class can be used to float an element left in your page. The floated element must precede the element it should be next to on the page. \*/

float: left;

margin-right: 8px;

}

.clearfloat { /\* this class can be placed on a <br /> or empty div as the final element following the last floated div (within the #container) if the #footer is removed or taken out of the #container \*/

clear:both;

height:0;

font-size: 1px;

line-height: 0px;

}

.button {

background-color:#F00;

color: white;

padding: 14px 20px;

margin: 5px 0;

border: none;

cursor: pointer;

width: 80%;

border-radius: 12px;

alignment-adjust:middle;

-webkit-transition-duration: 0.4s; /\* Safari \*/

transition-duration: 0.4s;

}

.button:hover {

border: none;

box-shadow: 0 12px 16px 0 rgba(0,0,0,03),0 17px 50px 0 rgba(0,0,0,0.5);

}

#input type{

background-color:#F09;

color: white;

padding: 14px 20px;

margin: 8px 0;

border: none;

cursor: pointer;

width: 80%;

border-radius: 12px;

alignment-adjust:middle;

}

input[type=text], input[type=password], input[type=number],input[type=email]

{

width: 100%;

padding: 12px 20px;

margin: 8px 0;

display: inline-block;

border: 1px solid #ccc;

box-sizing: border-box;

font-size:15px;

}

#PreviewPicture {

width: 250px;

height: 250px;

padding-left:20px;

background-position: center center;

background-size: cover;

-webkit-box-shadow: 0 0 1px 1px rgba(0, 0, 0, .4);

display: inline-block;

}

-->