**Lesson 4 - Here comes CSS**

**Lecture Notes**

**Activity**

Please construct a CSS and a HTML file that will take the following HTML and transform it to the picture below

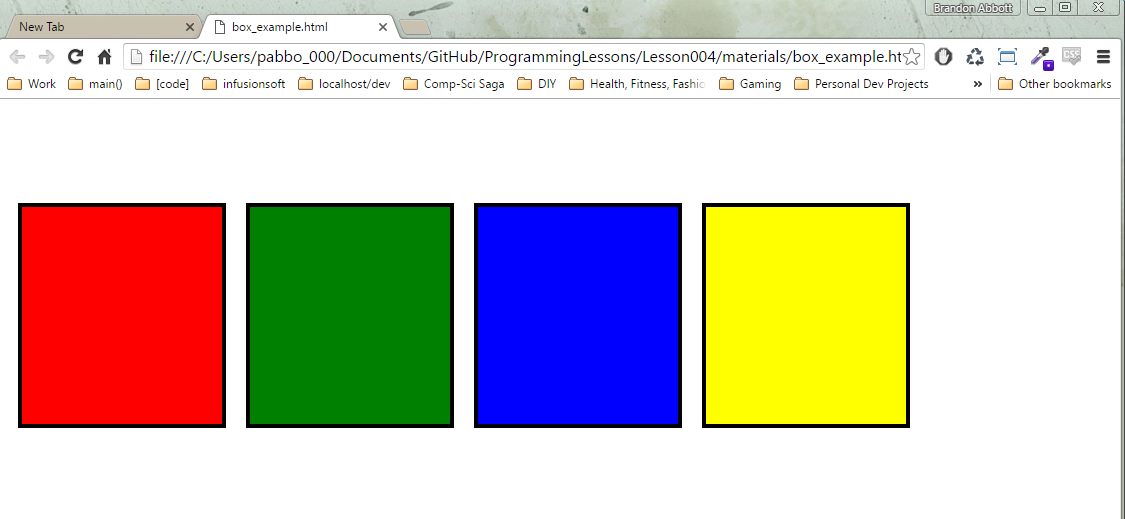
<body>

<div class="box red"></div>

<div class="box green"></div>

<div class="box blue"></div>

<div class="box yellow"></div>

</body>

**Lecture**

1. **Html Tables**
   1. **Basic HTML structure**
      1. <table> = defines the table
      2. <thead> = Optional, but great for legibility (required for some libraries)
      3. <tr> = table row, definitely want these
      4. <th> = table header - could use <td> instead, but <th> is convenient for css
      5. <td> = table data - always a direct child of <tr>

<table>

<thead>

<tr>

<th>Column Header</th>

</tr>

</thead>

<tbody>

<tr>

<td>Cell</td>

<tr>

</tbody>

</table>

* 1. **Pro Tips!**
     1. You can put stuff into a <td></td>
     2. But don’t put stuff in between say… <table> and <tr>
     3. <tr> must be a direct child to either <table>, <thead>, or <tbody>
     4. <td> must be a direct child to <tr>
     5. And so on…
  2. **The border attribute**
     1. Don’t use this. You can already set border using CSS
     2. This is on its way out of the html standard
     3. Thankfully it defaults to 0 and you can forget about it
     4. Separation of concerns
  3. **Styles**
     1. Border-collapse
        1. I didn’t know about this one!
        2. Collapses table borders and th/td borders down into one
        3. nice!
     2. Padding
        1. Applies to <th> and <td>
  4. **Chapter Summary**

Use the HTML **<table>** element to define a table  
Use the HTML **<tr>** element to define a table row  
Use the HTML **<td>** element to define a table data  
Use the HTML **<th>** element to define a table heading  
Use the HTML **<caption>** element to define a table caption  
Use the CSS **border** property to define a border  
Use the CSS **border-collapse** property to collapse cell borders  
Use the CSS **padding property** to add padding to cells  
Use the CSS **text-align** property to align cell text  
Use the CSS **border-spacing** property to set the spacing between cells  
Use the **colspan** attribute to make a cell span many columns  
Use the **rowspan** attribute to make a cell span many rows  
Use the **id** attribute to uniquely define one table

1. **HTML Lists**
   1. **Two types of lists**
      1. <ul> = un ordered list
      2. <ol> = ordered list (has numbers)
   2. **Don’t want bullet points?**
      1. style=”list-style-type: none”
   3. **<li> must be a direct child of <ul> or <ol>** 
      1. No funny business with things appearing between
      2. <ul> <a> <li> text</li> </a> </ul>
      3. This may be tempting to do, but don’t do it!
   4. **Nesting is ok (and encouraged)**
      1. You can put another <ul> inside of an already existing <li>
      2. You can actually put any element inside of an <li>
   5. **<dl> <dt> <dd>** 
      1. I have never seen these ever
      2. I would say - don’t use ‘em.
   6. **Horizontal lists**

ul#menu li {

display:inline;

}

* + 1. Very useful stuff! Especially when building a navigation

1. **HTML Block and Inline Elements** (<http://www.w3schools.com/html/html_blocks.asp>)
   1. **Block-level elements**
      1. <div>, <h1>, <h2>, <p>, etc….
      2. Always start on a new line and take up the full width available
      3. They stretch as far left-and right as possible
   2. **Inline elements**
      1. <span>, <a>, <img>
      2. They do-not take up the full width,
      3. and do not start on a new line
   3. **<div>**
      1. This is just a block-level element that comes with no padding, no margin, and no style
      2. These are useful for housing other, more meaningful elements such as <p> <h1> <img> <button> etc…
      3. This element has no inherent meaning
         1. It’s not a paragraph
         2. It’s not a header
         3. It’s just a plain-jane, run-of-the-mill, basic html element.
   4. **<span>**
      1. This is just a inline-level element that comes with no padding, no margin, and no style
      2. This is very useful for when you’re inside of an already meaningful element such as <p> <h1> <h2> <button> etc…
2. **HTML Entities** (<http://www.w3schools.com/html/html_entities.asp>)
   1. Sometimes you need special characters
   2. Special characters start with an ampersand (&)
   3. The most common one is the **non-breaking space**
      1. &nbsp;
      2. If you use &nbsp; the line will not break. That is - the words will not wrap down to the next line
      3. This is very useful in html headers such as <h1> <h2>
   4. A few other important ones (thankfully most are self-explanatory)
      1. &amp;
      2. &lt;
      3. &gt;
      4. &copy;
3. **CSS Layout - : The Display and Visibility Property**
   1. <http://www.w3schools.com/css/css_display_visibility.asp>
   2. <http://learnlayout.com/display.html>
   3. **Most HTML elements default either to** 
      1. display:block
      2. Display:inline
      3. If you’re curious - here’s all of them <http://www.w3schools.com/cssref/pr_class_display.asp>
   4. **Display:none or visiblity:hidden**
      1. Display:none - Stops the whole element from being rendered all-together
      2. Visibility:hidden - the element will not be shown, but the space it would normally occupy will remain.
      3. Used with javascript to show/hide thing
   5. **Block-Level Elements**
      1. Stretches to the left and right as far as it can
      2. <div>, <h1>, <p>, <form>, <header>, <footer>, <section>
   6. **Inline-elements**
      1. Does not start on a new line and only takes up as much width as necessary