

**Completion: Required**

**Submission: Required**

---

Last Name: \_\_\_\_\_

First Name: \_\_\_\_\_

---

This *tutorial+assignment* is to help you to learn basic HTML and CSS code for creating a website to present/visualize data etc.

**1) How and what to submit?**

***Submit the following (upload in Blackboard to the available container) in "one" PDF document (not in docx or any other format):***

- i) The certification page (see next page) should be the first page, followed by***
- ii) your solution to the problems given on this assignment.***

***One way is to copy the certification page and your solution to the problems into a Word document and then save the Word document as PDF, and upload the PDF version (not docx version). Only the PDF version will be graded.***

**2) Only ONE upload attempt is allowed: Before submitting a document through Blackboard, you should review the document being uploaded to make sure that you are uploading the correct document (e.g. do not upload the assignment belonging to another course). To help you prevent uploading wrong documents, notes (titled "[HelpOnSubmissionThroughBlackboard](#)" on how to save & review drafts before final submission have been uploaded under [Reference Material](#) folder.**

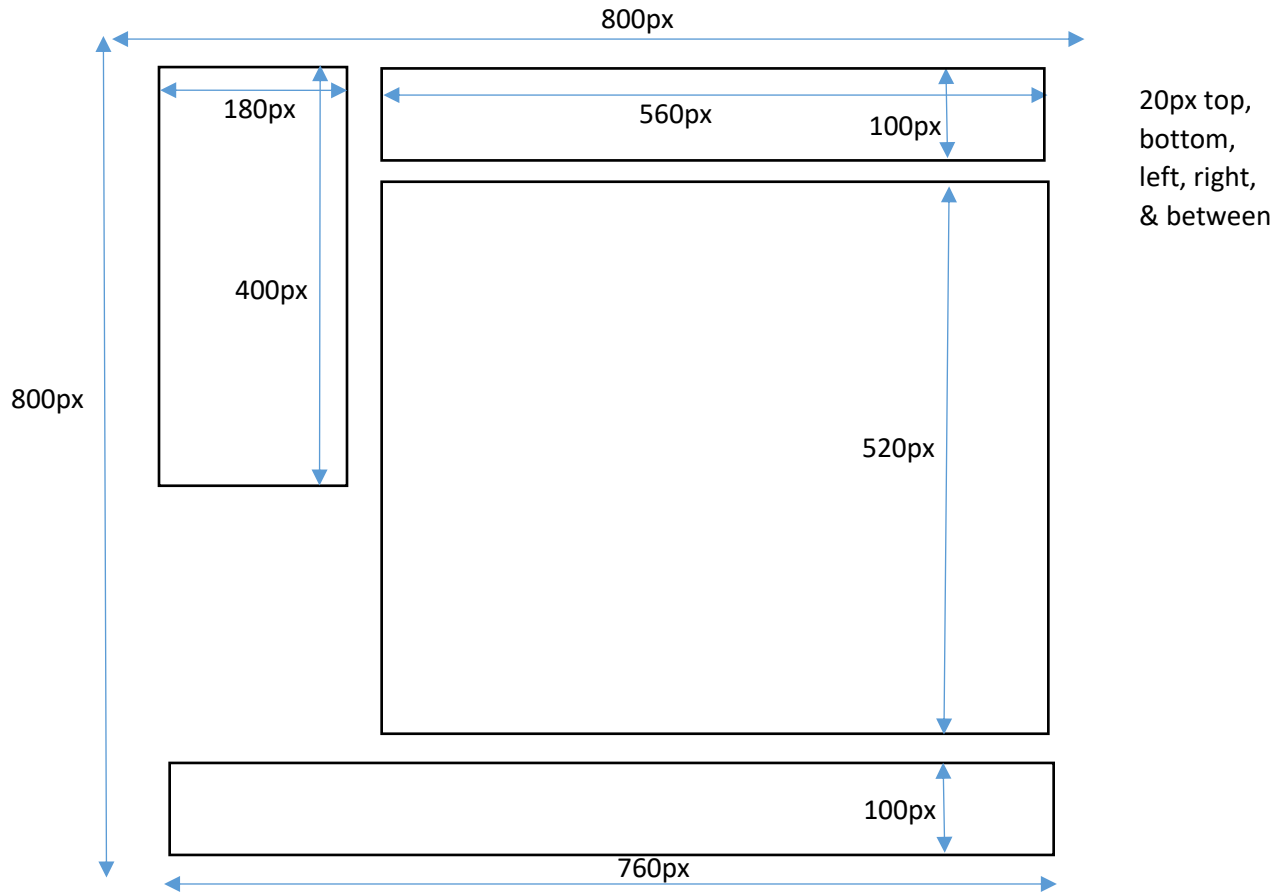
# **Certification Page**

*This page must be the first page of your uploaded document.*

*Your assignment will not be graded without this page (completed with your full name in the area provided) as the first page of your uploaded document.*

I, \_\_\_\_\_, certify that the work I am uploading represents my own efforts, and is not copied from anyone else or any other resource (such as Internet). *Furthermore, I certify that I have not let anyone copy from my work.*

## Tutorial Portion



A webpage is seen as different areas referred to as DIVs. In the above divided structure of the webpage being designed, there are four DIV areas in addition to the canvas (background).

### **DIV areas:**

- Canvas → 800x800
- Guide → 180x400
- Header → 560x100
- Body → 560x560
- Footer → 760x100

### **Contents of DIVs:**

- Canvas → Gray texture & color
- Guide → Guide (links) for additional webpages
- Header → "CSCI 3342 Assignment #1"
- Body → Description of some sort
- Footer → "Edinburg – Harlingen – Brownsville"

Create index.html in your RPi's /home/pi/ directory with the following specifying all the DIVs with some initial content:

```
<!DOCTYPE html>
<html>
  <head>
    <title> CSCI 3342 Assignment #1 </title>
  </head>
  <body>
    <div id="canvas">
      <div id="guide">
        <h3>Guide</h3>
        <hr>
        <ul>
          <li>Intro</li>
          <li>Exams</li>
          <li>Quizzes</li>
          <li>Assignments</li>
        </ul>
      </div>
      <div id="header">
        <h1>CSCI 3342 Assignment #1</h1>
      </div>
      <div id="body">
        <h2>What will we try to cover?</h2>
        <p>Each student will need a Raspberry &pi. <br/>Seems like a nice
          summer to learn Internet Programming. </p>
        <p>Topics</p>
        <p>    <ol>
              <li>HTML</li>
              <li>CSS</li>
              <li>Python</li>
              <li>PHP</li>
              <li>PHP</li>
              <li>Node JS</li>
              <li>XML</li>
              <li>JSON</li>
            </ol>
          </p>
      </div>
      <div id="footer">
        <h6>Edinburg-Harlingen-Brownsville</h6>
      </div>
    </div>
  </body>
</html>
```

Run the above by typing <file:///home/pi/index.html> in RPi's browser.

To give **style** to the webpage, we can use CSS (**Cascading Style Sheet**) file. Modify the above file to add the reference to the CSS file that we will create shortly:

```
<head>
  <title> CSCI 3342 Assignment #1 </title>
  <link rel="stylesheet" type="text/css" href="e1.css" />
</head>
```

Create /home/pi/e1.css file as follows which gives background color and text color to each DIV:

```
#canvas {
  background-color: grey;
  color: white;
}
#guide {
  background-color: blue;
  color: yellow;
}
#header {
  background-color: yellow;
  color: blue;
}
#body {
  background-color: cyan;
  color: black;
}
#footer {
  background-color: black;
  color: white;
}
```

Run the above by typing <file:///home/pi/index.html> in RPi's browser.

Modify #canvas DIV in e1.css file to setup 800 pixels of width with default left and right margins:

```
#canvas {
  background-color: grey;
  color: white;
  width: 800px;
  margin-left: auto;
  margin-right: auto;
}
```

Modify #header DIV in e1.css file to position header per design, and center the text within it:

```
#header {  
    background-color: yellow;  
    color: blue;  
    width 560px;  
    margin-left: 220px;  
    margin-right: 20px;  
    text-align: center;  
}
```

Refresh the browser to verify the effect of the above change.

Similarly, position the other DIVs according to the design template. The modified CSS file should look as follows:

```
#canvas {
    background-color: grey;
    color: white;
    width: 800px;
    margin-left: auto;
    margin-right: auto;
}
#guide {
    background-color: blue;
    color: yellow;
    width: 180px;
    margin-left: 20px;
}
#header {
    background-color: yellow;
    color: blue;
    width 560px;
    margin-left: 220px;
    margin-right: 20px;
    text-align: center;
}
#body {
    background-color: cyan;
    color: black;
    width 560px;
    margin-left: 220px;
    margin-right: 20px;
}
#footer {
    background-color: black;
    color: white;
    margin-left: 20px;
    margin-right: 20px;
    text-align: center;
}
```

Refresh the browser to verify the effect of the above changes.

As you may note, the DIVs are pushing each other in vertical direction. We need the #header and #body DIVs to be side by side with #guide DIV. So, we need to “float” #guide DIV in order for it not to delimit the vertical space. Modify #guide as follows, and refresh the browser to note the effect of the change:

```
#guide {  
    background-color: blue;  
    color: yellow;  
    width: 180px;  
    margin-left: 20px;  
    float: left;  
}
```

Refresh the browser to verify the effect of the above changes.

Convert the items in #guide DIV to links by modifying #guide section in index.html as follows:

```
<div id="guide">  
    <h3>Guide</h3>  
    <ul>  
        <li><a href=" " >Intro</a></li>  
        <li><a href=" " >Exams</a></li>  
        <li><a href=" " >Quizzes</a></li>  
        <li><a href=" " >Assignments</a></li>  
    </ul>  
</div>
```

Refresh the browser to verify the effect of the above changes.

Remove the unsightly underlines from the links by modifying the CSS file to inhibit “text-decoration” on any “a” type links:

```
a {  
    text-decoration: none;  
}
```

Refresh the browser to verify the effect of the above changes.

Change the color of links to yellow to stand-out on the blue background by specifying “color” to “a” type in CSS file:

```
a {  
    text-decoration: none;  
    color: yellow;  
}
```

Refresh the browser to verify the effect of the above changes.

Remove the unsightly bullets from the unordered list under #guide DIV by adding the following to the CSS file:



```
#guide ul {  
    list-style-type: none;  
}
```

Refresh the browser to verify the effect of the above changes.

Set the padding to a reasonable 10 pixels from the left for the links:

```
#guide ul {  
    list-style-type: none;  
    padding: 10px;  
}
```

Refresh the browser to verify the effect of the above changes.

Now, we want to add 10 pixels to all of the contents within #guide, and reduce the width to 160 pixels so that #guide does not encroach upon #header and #body DIVs:

```
#guide {  
    background-color: blue;  
    color: yellow;  
    width: 160px;  
    margin-left: 20px;  
    float: left;  
    padding: 10px;  
}
```

Refresh the browser to verify the effect of the above changes.

As one can note, #header is not at the same height as #guide. This because of the default margin for h1, h2, h3, etc. Modify CSS to remove margins from all types of headings:

```
h1, h2, h3, h4, h5, h6 {  
    margin: 0px;  
}
```

Refresh the browser to verify the effect of the above changes.

One can add 20 pixels of padding to top and bottom of canvas to make it look even on all sides:

```
#canvas {  
    background-color: grey;  
    color: white;  
    width: 800px;  
    margin-left: auto;  
    margin-right: auto;  
    padding-top: 20px;
```

```
padding-bottom: 20px;  
}
```

Refresh the browser to verify the effect of the above changes.

Also, add 20 pixels of top margin to #body DIV to separate it from #header DIV:

```
#body {  
    background-color: cyan;  
    color: black;  
    width 560px;  
    margin-left: 220px;  
    margin-right: 20px;  
    margin-top: 20px;  
}
```

Refresh the browser to verify the effect of the above changes.

One can define classes in HTML and target these classes in CSS. Modify index.html as follows to define a class “thispage” within “Intro” link:

```
<div id="guide">  
    <h3>Guide</h3>  
    <ul>  
        <li><a class="thispage" href=" ">Intro</a></li>  
        <li><a href=" ">Exams</a></li>  
        <li><a href=" ">Quizzes</a></li>  
        <li><a href=" ">Assignments</a></li>  
    </ul>  
</div>
```

Modify CSS to have this link boldfaced:

```
#guide .thispage {  
    font-weight: bold;  
}
```

Refresh the browser to verify the effect of the above changes.

Get a snippet of an image and save it as mascot.png or mascot.bmp in the home directory, and reference it accordingly as part of #guide DIV:

```
<div id="guide">  
    <h3>Guide</h3>  
    <ul>  
        <li><a class="thispage" href=" ">Intro</a></li>  
        <li><a href=" ">Exams</a></li>  
        <li><a href=" ">Quizzes</a></li>  
        <li><a href=" ">Assignments</a></li>  
    </ul>  
      
</div>
```

DIV is block level delimiter. One can have inline delimiters as well. Let's modify index.html to define a class for the word "try" in #body to be targeted in CSS:

```
<div id="body">
  <h2>What will we <span class="style1">try</span> to learn?</h2>
  <p>Raspberry &pi;; <br/> Seems like a good summer to learn
    Internet Programming.</p>
  <p>Topics</p>
  <p>    <ol>
      <li>HTML</li>
      <li>CSS</li>
      <li>Python</li>
      <li>PHP</li>
      <li>PHP</li>
      <li>Node JS</li>
      <li>XML</li>
      <li>JSON</li>
    </ol>
  </p>
</div>
```

Modify CSS to target the class "style" as follows:

```
#body .style1 {
  color: purple;
  font-weight: bold;
}
```

Refresh the browser to verify the effect of the above changes.

One can add animation as well. Let's make the word "try" have different background colors and different font weights with 1 second to animate, 1 second of delay between each animation and let it animate forever. Modify CSS as follows (other animation attributes are added for you to play with):

```
#body .style1 {
    color: purple;
    font-weight: bold;
    animation-name: an1;
    animation-duration: 1s;
    animation-delay: 1s;
    animation-iteration-count: infinite;
    animation-direction: alternate;
    animation-timing-function: linear;
    animation-fill-mode: backwards;
}

@keyframes an1 {
    from {background-color: grey; font-weight:1;}
    to {background-color: cyan; font-weight:bold;}
}
```

Refresh the browser to verify the effect of the above changes.