## *Updated January 20, 2021*

## Objective

Your task is to use HTML5 **and CSS** to build and publish to your webhost account a semantically correct webpage listing the four 21st century US Presidents. The learning goals of this assignment are:

* To gain deeper understanding of semantic design principles, and exercise that understanding.
* To begin working with an external CSS file, text decoration properties, and the box model.
* To introduce the basics of Flexbox
* To begin practicing the use of em units.

## Task

Build a simple HTML file with a heading that states “21st Century U.S. Presidents. Then list the names of the four presidents each in his own paragraph. The structure for this is an article wrapper around all the content, with child elements of h1 and p. Do not include any other HTML markup. Give each element a class name that semantically represents what the element means.

Use CSS flexbox properties to organize the elements as follows.

|  |  |
| --- | --- |
| 21st Century U.S. Presidents | |
| George W Bush | Barack Obama |
| Donald Trump | Joseph Biden |

Decorate your page using CSS text decoration and box model properties as described in the specifications below.

**All sizing on this lab must use either em or percent units. Use of pixels and other units is prohibited.[[1]](#footnote-1)**

See the appendix to this document for more information about ems.

## Specifications

1. Do not rely on any browser default CSS. To ensure this is the case, include the following link element in your HTML <head> **ABOVE** your link to your own CSS file:

**<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/meyer-reset/2.0/reset.css">**

This “reset” file essentially turns off all the default CSS and leaves you to start with a blank presentation slate.

1. Use CSS box model properties to draw a border around each President. You have complete control over how you style that border.
2. Do NOT use the width property to set the width of your president boxes. Instead, use the third value of the flex: property. I recommend setting it to 34%.
3. Add at least 1em of padding and at least 1em of margin inside and outside each of the presidential borders. **IMPORTANT: horizontal padding + border + margin adds to the element’s width and must be considered when calculating how much horizontal space an element will take up.[[2]](#footnote-2) *[Do not use the declaration box-sizing: border-box; ]***
4. You may add height or additional vertical padding to your president boxes if you wish.
5. Set a font-family for all the text in the document other than the browser default font. Declare three families in your rule (with the latter two being fall back choices.) **Optional:** you may use different font-families for the heading and presidents. You may use different font-families for each political party of President.
6. Increase the font-size of the heading text to a size you find aesthetically pleasing. Increase the font-weight of the heading text.
7. Horizontally center the text in all the elements using CSS.
8. Use line-height to increase the line-height of text to 1.5.
9. Color the Democrat presidents (Obama/Biden) blue and the Republican presidents (Bush/Trump) red.
10. As Biden is current President, italicize him (using CSS).
11. **Optional:** You may (but are not required to) make political commentary by using CSS to underline presidents you like and strikethrough presidents you don’t like.
12. Resist the urge to add additional HTML markup or additional presentation complexities to your Lab 3 solution.

**Overarching Constraints**

1. As always: the webpage must be built using only HTML5. You are not permitted to use CSS nor JavaScript at all. You are not permitted to build in XHTML or HTML4.01. Any use of **<p>** or **<br>** or **<hr>** or **<b>** or **<i>** must be for semantic, not presentational, purposes. Beyond these constraints, you may use any other HTML5 element you deem semantically appropriate.
2. Your page layout MUST be achieved using Flexbox. You may NOT use float or position or grid or table or display: inline-block to organize the presentation of your content.[[3]](#footnote-3)
3. The <head> of your HTML5 file must contain:
   * A meta element naming the Unicode character set you want the browser to use (UTF-8).
   * a <title> element
   * a meta element naming YOU as the page author. E.G.   
     ***<meta name="author" content="Daniel Mittleman">***
   * Recommended: This element: ***<meta name="robots" content="noindex, nofollow">***that will prevent search engines from indexing your page.[[4]](#footnote-4)
4. **You must write a source code comment justifying every use of a <div>, <hr>, or <br> as being the best semantic option in the context it is used. Use of any <div>, <hr>, or <br> without an adjacent source code justification is considered wrong.**
5. All of your sizing must use either em or % units. You are not permitted to use pixel or any other sizing unit on this lab.
6. You may choose to discuss any other element choices you think merit discussion in source code comments.

## Deliverables

Publish this lab to your /lab3 folder as **index.html** with two subfolders as described above.

Submit to the Submissions Dropbox (as a plain text comment, please) your answers to:

1. Did your code pass HTML and CSS validation? If not, why?[[5]](#footnote-5) How did you address this issue?
2. What outstanding questions do you have for me to address in class this week? Or what new questions surfaced when doing this lab?
3. Anything else you want to reflect upon

## Deadline

As stated on D2L.

# Appendix: ems

The em unit is superior to the px unit for responsive design. But it is newer, and a bit trickier to use. In our course (now that we are into CSS), we will NEVER use pixel (px) units and rely primarily on em units so as to exercise skills and build confidence in using them.

**The em unit is equal to the current font-size.** That means for whatever element we are focused on, we first update the value of that element’s font-size property (in pixels) and we set the value of 1em to be whatever that element’s font-size is. Then, we can use the value of an em to calculate other property values for the element such as width, height, padding, border, margin (etc.) size.

To do this, it is important to know that elements inherit their font-size value from their parent element. And that begs the question: what then is the font-size of the html element (since it is the root element it has no place to inherit its value from)?

And that is a very good question. HTML and CSS standards do not define the default value of the root element. But every modern browser has defined the default value of the root element to be 16px. So, that is the universal value we can expect to start with. We, as designers, can change the font-size of the html element (and there are good arguments to do so), but for now lets leave it at 16px and work from there.

This table provides the value of an em for different pixel sizes based on 1em = 16px.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1px | .0625em | 8px | .5em | 18px | 1.125em |
| 2px | .125em | 9px | .5625em | 20px | 1.25em |
| 3px | .1875em | 10px | .625em | 24px | 1.5em |
| 4px | .25em | 11px | .6825em | 32px | 2em |
| 5px | .3125em | 12px | .75em | 48px | 3em |
| 6px | .375em | 13px | .8125em | 64px | 4em |
| 7px | .4375em | 14px | .875em | 80px | 5em |
| 8px | .5em | 15px | .9375em | 96px | 6em |

1. The one exception to this rule is the line-height property may be unitless. [↑](#footnote-ref-1)
2. I noticed some of you used box-sizing: border-box; on Lab 1. For this lab, do not add the declaration box-sizing: border-box. [↑](#footnote-ref-2)
3. These are all the other layout systems I can think of. [↑](#footnote-ref-3)
4. As you probably don't want this assignment to become part of your permanent portfolio of work on the web. If you do, then change noindex to index. [↑](#footnote-ref-4)
5. Validator is at <https://validator.w3.org/> [↑](#footnote-ref-5)