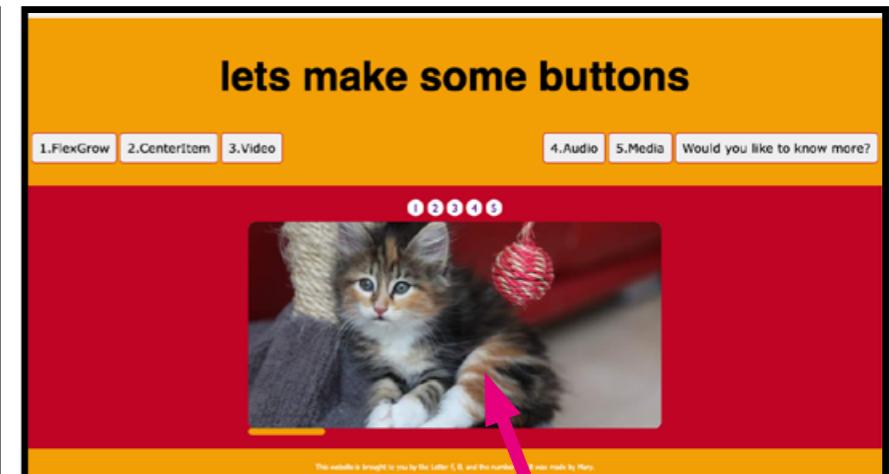
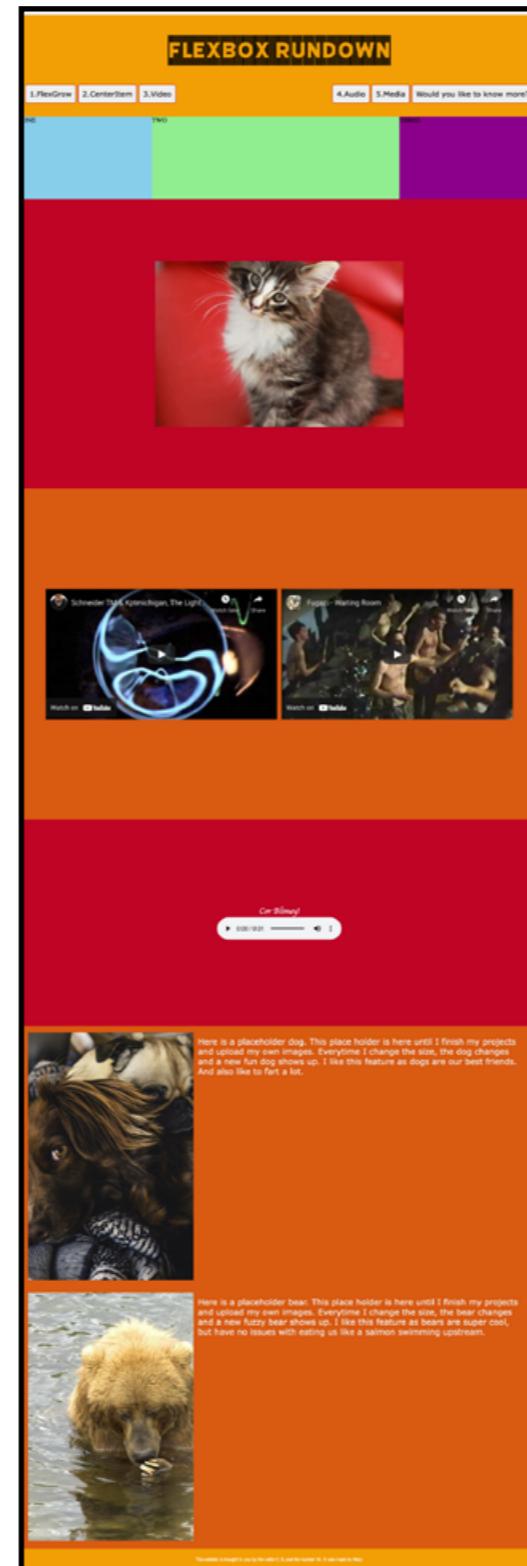


HTML / CSS Worksheet 5

This week's worksheet we will look at:
turning our `` into `<button>`
styling our buttons using css
adding a `<footer>`
making a linking button

duplicating our first page to a second page
centering a large horizontal image
(optional carousel worksheet linked)
change colourscheme with a plan
adding a google and adobe webfont

Visual Studio Code (VSC)
Adobe colorwheel
google fonts
adobe fonts



optional carousel!

1

Hi. Let's get started. We made what could very well be a menu, but our list items could actually be buttons. Let's do that first. So Instead of `` we will have `<button>` If you did the anchor links, leave them. See if you can sort out how to replace the `` with `<button>`

Then Make one last button.

Save, commit, push.

.2

This button doesn't go anywhere, AND our `` styles are gone! Don't worry about making a new page yet, but let's look at styling our buttons using CSS.

The `<button>` in html5 provides a default button style, which is great because you now know you made a button, but it's not very pretty. We are going to style our buttons a few ways, but there are WAY more ways to style buttons using CSS and I encourage you to find your own style.

First, style your button like this: at the bottom of your css file.

```
<ul>
  <div class="container-navigation">
    <button>1.FlexGrow</button>
    <button>2.CenterItem</button>
    <button>3.Video</button>
    <button class="push">4.Audio</button>
    <button>5.Media</button>
  </div>
</ul>
```

```
<button>5.Media</button>
<a href="button.html" target="blank"><button>Would you
like to know more?</button></a>
</div>
```



Flexbox Rundown

1.FlexGrow | 2.CenterItem | 3.Video | 4.Audio | 5.Media | Would you like to know more?

```
/* button style */

button {
  border: 1px solid blueviolet;
  border-radius: 5px;
  font-family: Verdana, Geneva, Tahoma, sans-serif;
  font-size: large;
  padding: 10px;
  margin: 2px;
  transition-duration: .4s;
}
```

Everything we see here is pretty familiar except that

`transition-duration: .4s;`

What that does, is creates a visual transition between when the button is inactive, to when a user hovers over the button. Right now it won't do anything, so let's add that and see what happens.

Now when we hover over the button, it will transition from white to green and the text will go from black to white.

Now we will make a visual change when the user presses the button. So let's do that right now!

Again, everything looks familiar except for that `transform: translateY(4px);` So what happens? **Save, commit, and push.**

There is SO MUCH MORE you can do. You can search up buttons in css, and see what you can do. These are pretty vanilla buttons, but you could go bonkers if you want! [Starting here](#) is a good place.

Flexbox Rundown

1.FlexGrow 2.CenterItem 3.Video

4.Audio 5.Media Would you like to know more?

```
button:hover {  
    background-color: limegreen;  
    color: white;  
}
```

Flexbox Rundown

1.FlexGrow 2.CenterItem 3.Video

4.Audio 5.Media Would you like to know more?

```
button:active {  
    background-color: #3e8e41;  
    box-shadow: 0 5px darkred;  
    transform: translateY(4px);  
}
```

1.FlexGrow

2.CenterItem

3.Video

4.Au

ONE

TWO



3

Don't make a new page yet! We need to add a `<footer>` to our original page, then we are done with our index.html. We know that we have 5 anchor links to this index page and one button that will go to another page and that will share the same css file as the first page.

Footer time. (But put your name in there).
Check it, and you'll see it's there, but it's not
styles.

One more thing: we must remove the height requirement on the section above the footer. So head up to `.section-five` and remove the height. **Save, commit, push.**

```
</main>

<footer>
    <p>This website is brought to you by
    the Letter F, B, and the number 42. It
    was made by YOUR NAME HERE.</p>
</footer>
</body>
```

```
/* footer goes here */
```

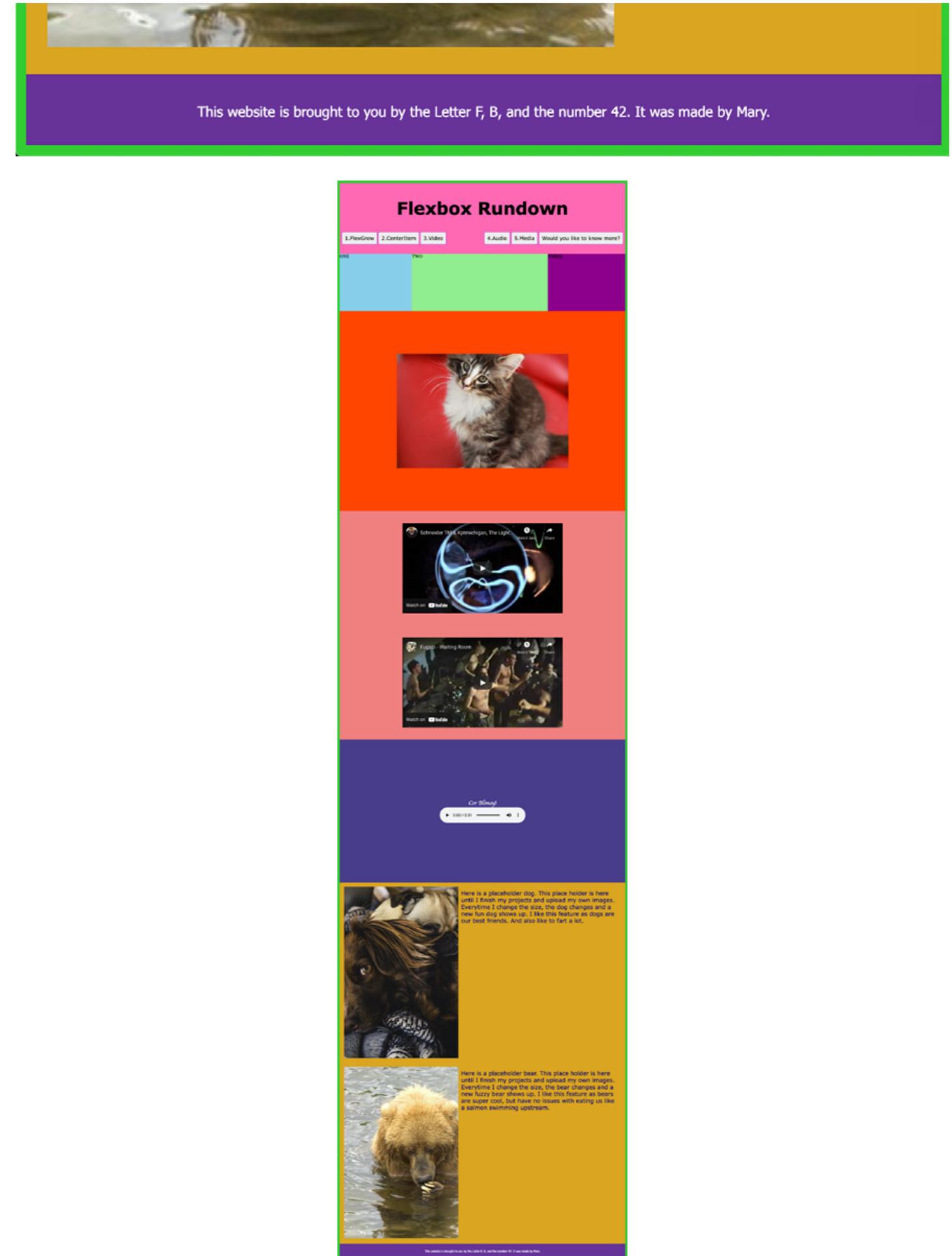
```
✓ .section-five {  
    background-color: #goldenrod;  
}
```

Ok. So, we have a footer, and a menu we are happy with, and let's have a look. If we were making a website we would know what our menu items would be, how many buttons we would need, and once we have all that content in, we can start duplicating pages and making new interior pages!

One final look.

Are we happy? Ok, now in VSC go to File > Save as and save a copy of your index.html file. I named mine **button.html** as that matches the button we created in step 1. You name it whatever you want, just make sure it matches in your html. Save it in the same folder as your index.html

Then **save, commit, and push**. Then click on the last button you made and see if a new window pops up. If you included the **target=" _blank"** it should.



4

Cool. Now you should have a duplicate of your index.html page. In your new, second page:

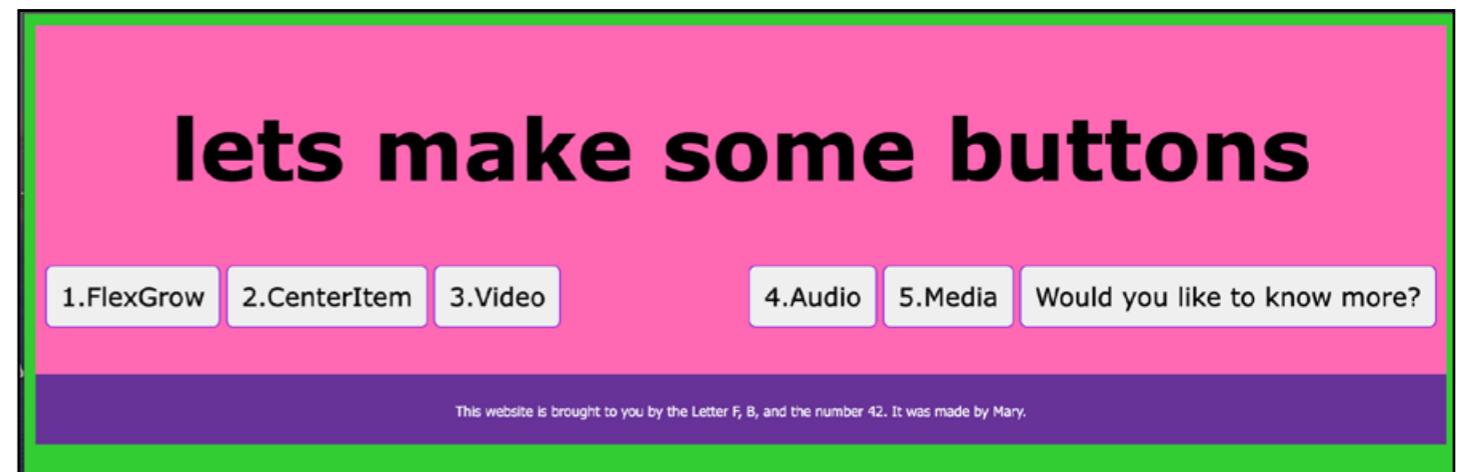
Change the `<h1>` in the `<header>` to
let's make some buttons.

Now, I want you to
remove everything in your second page
between `<main>` and `</main>` you got
that correctly, dear reader. So now, you have
a consistent template.

There's our header, there's our footer. There's
nothing in between.

Save, commit, push if you haven't done
so already.

```
<button>Would you like to know more?</button>
</div>
</ul>
</header>
<main>
</main>
<footer>
  <p>This website is brought to you by the Letter F, B, and the number 42. It was made by Mary.</p>
</footer>
```



5

First, we are going to place a horizontal image inside a new section. We will do it in the same way we did our sections on our index.html page. You can also reuse sections if you want, but we don't want to do that right now. Just so we can learn.

I have already set up some class selectors, because I already have something in mind. After practicing you should be able to think and plan on what you might want to name things to get organised.

I have added rounded corners on the image and a **box-shadow**. the 10px 10px is the size of the drop and the 20px is the blur. Give it a go by changing the size blur and color.

I put the img width at 70% which a little bit of a cheat, but now I know the placeholder image will scale as I change the width of the browser window, and everything stacks. My image will scale its size too. **Save, commit, push.**

```
</header>

<main>
  <section class="button-one-section">
    
  </section>
</main>

<footer>
```

```
.button-one-section {
  background-color: cornflowerblue;
  padding: 20px 0px;
  display: flex;
  justify-content: center;
  align-items: center;
}
```

```
.button-top-img {
  border: 3px solid white;
  border-radius: 10px;
  box-shadow: 10px 10px 20px black;
  width: 70%;}
```



Dear reader this is where you make a decision.

If you are happy with this great, but want to know how to place a slider in from an online tutorial using code snippets, **FINISH THIS WORKSHEET AND COME BACK AND DO THIS WORKSHEET.**

It is not required for the worksheets but it is encouraged as it will help you understand how to navigate the MANY issues that involve finding “cool code snippets” on the internet and then actually having to put them in to your html file with varying success. Either way, continue down and then decide.

****Any code snippets grabbed from websites must be referenced in your final project in a .txt file. The description and link of the code snippet must be included in a list and submitted with your hand-in****

lets make some buttons

1.FlexGrow 2.CenterItem 3.Video 4.Audio 5.Media Would you like to know more?

OR

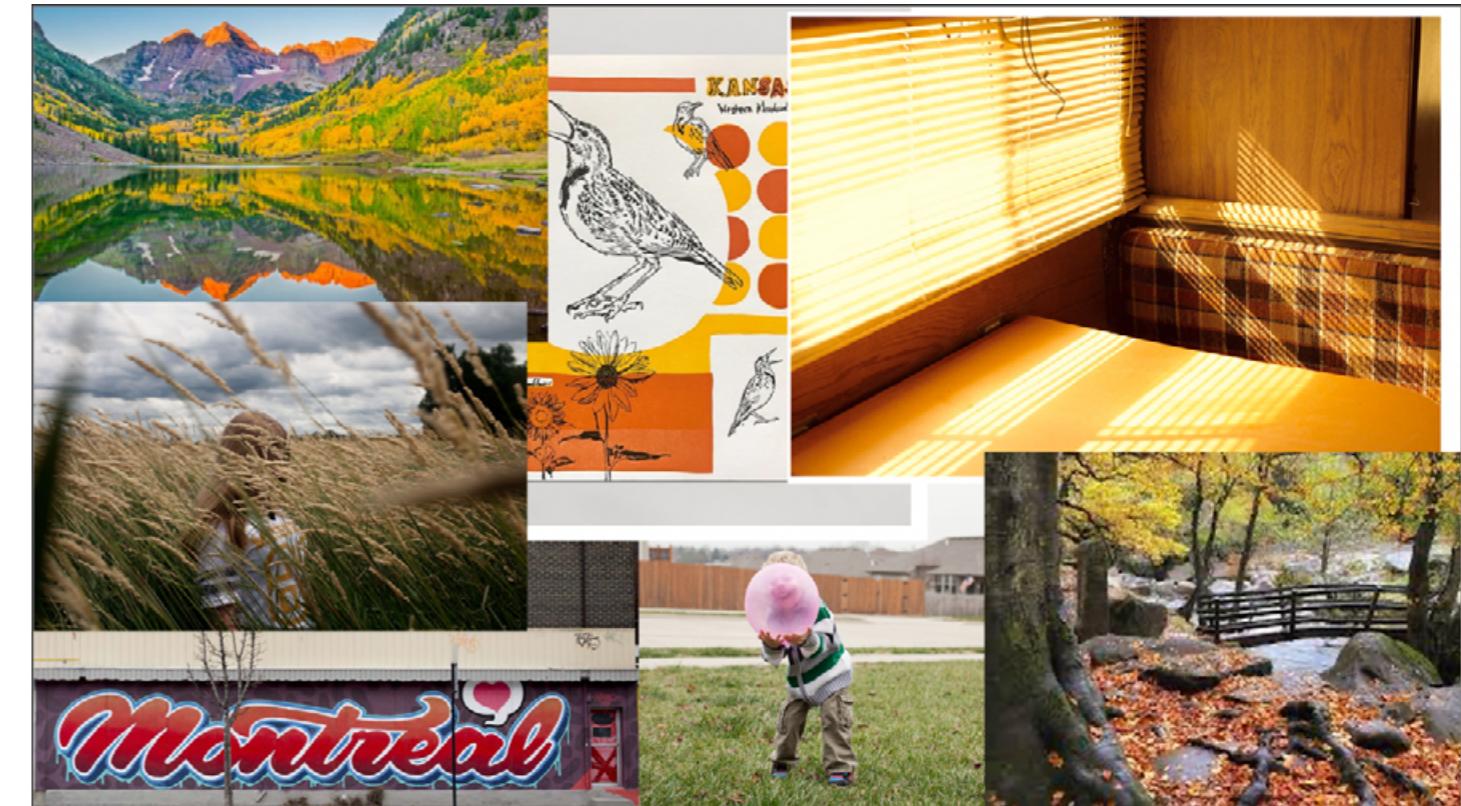
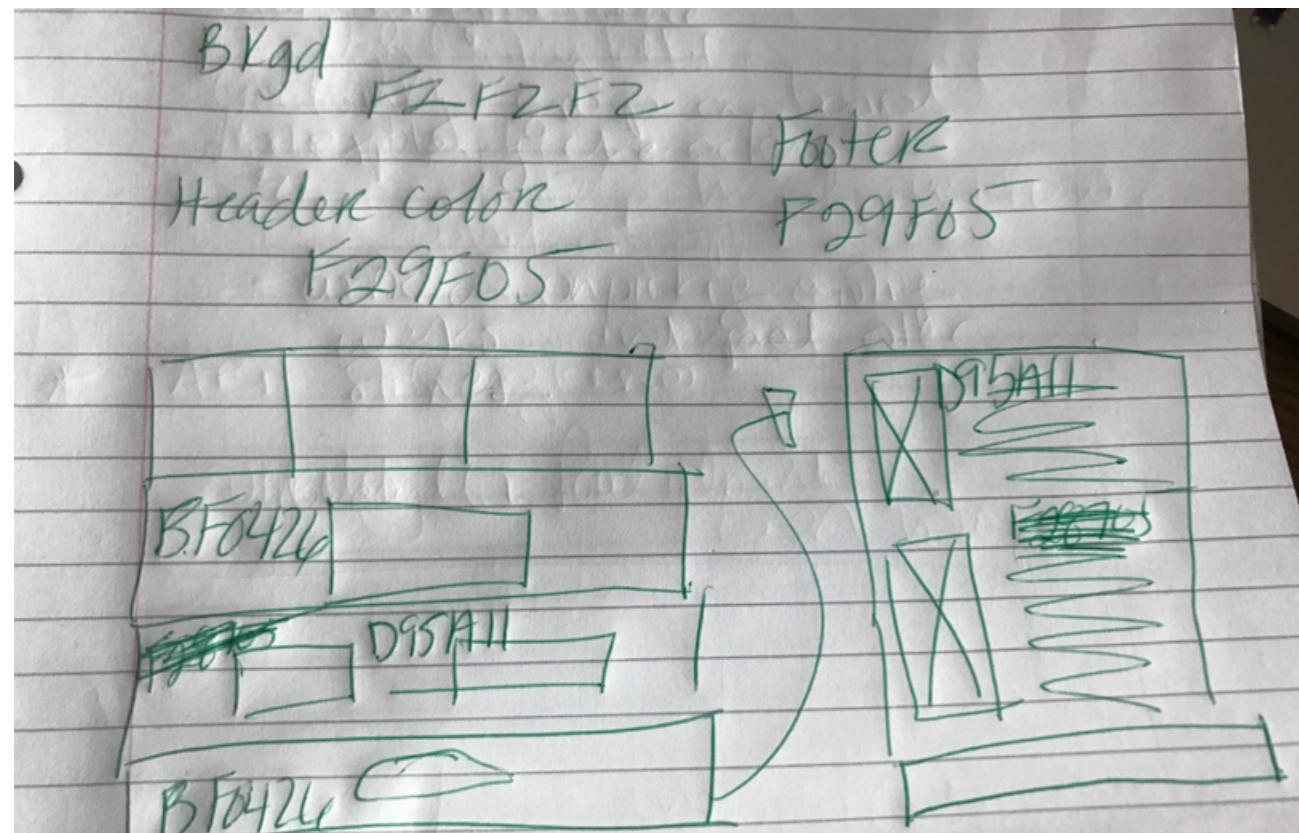


6

My colours are OK, but there are different ways to put in colours in your CSS file including tints and transparency. I'm only showing you a small snapshot, [to know more you can read about it here](#). I made a quick mood board, and then ran it through adobe colors. It gave me a quick color palette. This is a great way to get colors quick. Adobe colors provide [hex colors](#), they are a series of letters and numbers starting with a # tag. (alt3 on a mac keyboard).

<https://color.adobe.com/create/image>

I made a plan using these hex numbers, and changed the sections in my index.html page to look more like my color palette.



My **body** is now **#F2F2F2**

My **header** is now **#F29F05**

My **.section-two** is now **#BF0426**

My **.section-three** is now **#D95A11**

My **.section-four** is now **#BF0426**

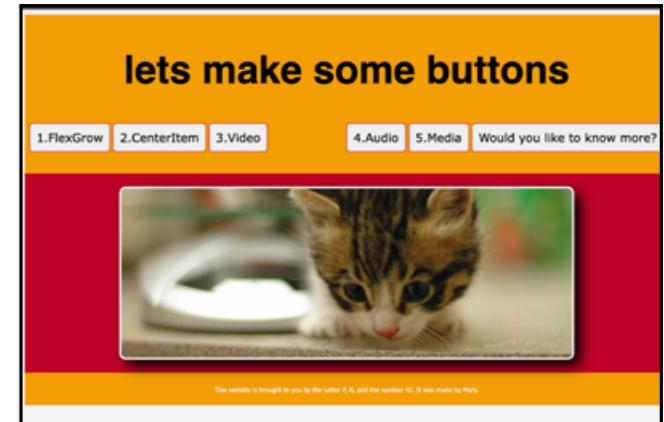
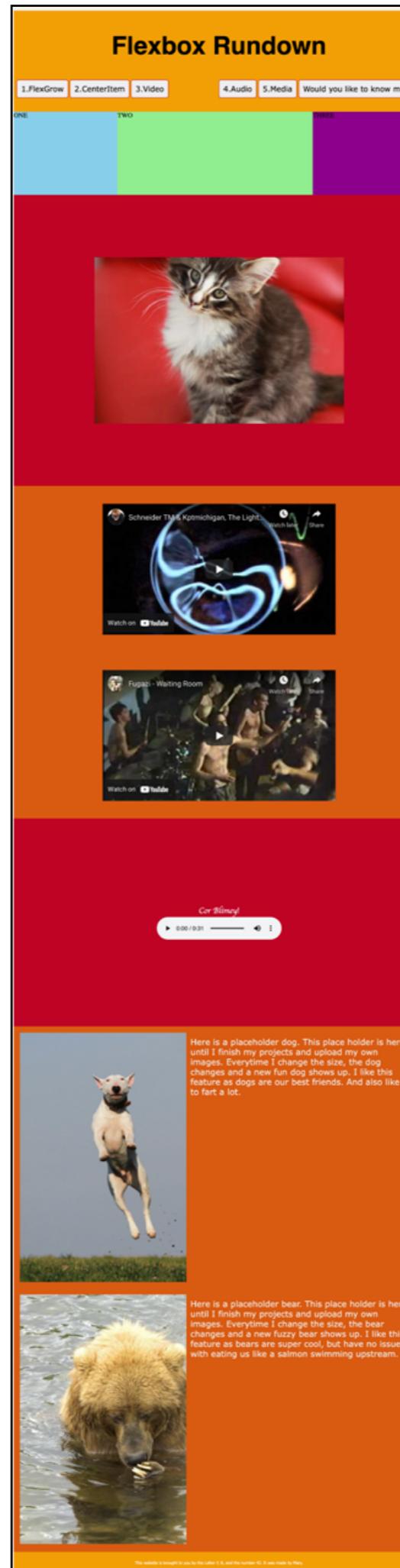
My **.section-five** is now **#D95A11**

My **footer** is now **#F29F05**

My **.button-one-section** is now **#BF0426**

And you can just keep going. I styled a bit more of the buttons, and changed the text of the media container. You can think about this as you make your own website. Also if you already have a color palette before you start building, this will make your styling go smoother and faster.

Save, commit, push



7

The last thing I'm going to do is add a google font, and an adobe font, one on each page. It is treated in the same way that we add our style sheets, in the `<head>` section of our html files. It's more about going and finding the font you want. Go to https://developers.google.com/fonts/docs/getting_started and you will see something like this. But I don't want tangerine. I want a different font. So I go to <https://fonts.google.com/> and get

Here's an example. Copy and paste the following HTML into a file:

```
<html>
  <head>
    <link rel="stylesheet"
          href="https://fonts.googleapis.com/css?family=Tangerine">
    <style>
```

down to business. You can choose whatever font you want, but I chose <https://fonts.google.com/specimen/Josefin+Sans#standard-styles>

There is no need to download it, you copy from "use on the web and grab the url

Put the link in your html like you would your style.css and style your `h1` in your css so the font-family has 'Josefin Sans' (or whatever you choose), as your first font.

The screenshot shows the Google Fonts website for the 'Josefin Sans' font family. At the top, there are tabs for 'Select styles', 'Glyphs', 'About', and 'License'. A 'Selected family' sidebar is open, showing 'Josefin Sans' is selected with 'Regular 400' chosen. Below the sidebar, a message says 'You can now share your selected families with others.' with a 'Dismiss' button. The main content area shows 'Variable axes' with sliders for 'Italic' (set to 0) and 'Weight' (set to 400). A preview window displays the text 'Almost before we we had left the gi' in the selected font. Below the preview, there are sections for 'Sample glyphs' (showing letters A through P) and 'Use on the web' (with options for <link> or @import and the corresponding CSS code). The CSS code for @import is highlighted in pink.

So now we have Josefin Sans as our h1, for index.html, and the regular verdana for our second page, because we haven't declared that stylesheet for our interior page.

Now it's time for Adobe fonts.

```
<link rel="stylesheet" href="styles.css">
<link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Josefin+Sans&display=swap">
```

```
h1 {
  font-family: 'Josefin Sans', Verdana,
  Geneva, Tahoma, sans-serif;
  font-size: 60px;
  text-align: center;
}
```

Flexbox Rundown

1.FlexGrow 2.CenterItem 3.Video 4.Audio 5.Media Would you like to know more?

lets make some buttons

1.FlexGrow 2.CenterItem 3.Video 4.Audio 5.Media Would you like to know more?



8

Adobe fonts work in a similar way, but require you to be signed in. **First thing I did was read the info in the link below.** Like all Adobe tutorials, they can explain it better than I ever could. **So do that first.**

<https://helpx.adobe.com/fonts/using/add-fonts-website.html> Follow these steps.

I chose filicudi-color-pride as my font.
I made a web project, which I named test
(I recommend you name it something more memorable) and then adobe provides the relative links. (images to follow)

Filicudi Color Pride



Add fonts to [test](#)

Enter a new project name or select from your existing projects:

test

Select the fonts you'd like to use on your website. Adding more beyond the default selection may affect website load time.

Filicudi Color Pride

[Pride](#)

[THE QUICK BROWN FOX JU...](#)

This project also includes Filicudi Color Pride.

Cancel

Save

There you go. Adobe gives us the same information that google fonts does. Let's put that stylesheet on our second page, just like we did for the google font, but using this information.

Now, in your **h1** in your css, add the style after 'Josefin-Sans' **save**, **commit and push** (do you need the reminder anymore?) so it looks like this and see what happens.

1 fonts added to test

To use these fonts on a web page, copy this code into the <head> tag of your HTML.

```
<link rel="stylesheet" href="https://use.typekit.net/voo6uzo.css">
```

If you'd like to use fonts in HTML email, use the [@import link](#).

Fonts added:

Filicudi Color Pride

```
font-family: filicudi-color-pride, sans-serif;  
font-weight: 700;  
font-style: normal;
```

[Edit Project](#)[Done](#)

```
h1 {  
    font-family: 'Josefin Sans',  
    'filicudi-color-pride', Verdana,  
    Geneva, Tahoma, sans-serif;  
    font-size: 60px;  
    text-align: center;  
}
```

Ok! **Congratulations.** If you, dear reader, have finished these worksheets, you now have the tools to get started on your own website! **Is there more to learn? Absolutely!** This is just the beginning of learning about what html and css can do, and what you can create with it. It takes practice, patience, and sometimes meditation, or a long walk, or banging your head on the desk (not recommended). But now, you can get started. Have a plan and use what you have learned. Well done. **Pat yourself on the back.**

Flexbox Rundown

1.FlexGrow

2.CenterItem

3.Video

4.Audio

5.Media

Would you like to know more?

LETS MAKE SOME BUTTONS

1.FlexGrow

2.CenterItem

3.Video

4.Audio

5.Media

Would you like to know more?

