IST 263

# Lab 09

## This lab covers:

1. Centering
2. Advanced Flexbox Layout

## SETUP

Create a folder in your GitHub repository called lab09. Place a copy of your latest skills page and stylesheet in the new folder. Please make sure to keep the same file names in the new folder or your navigation will not work on later labs.

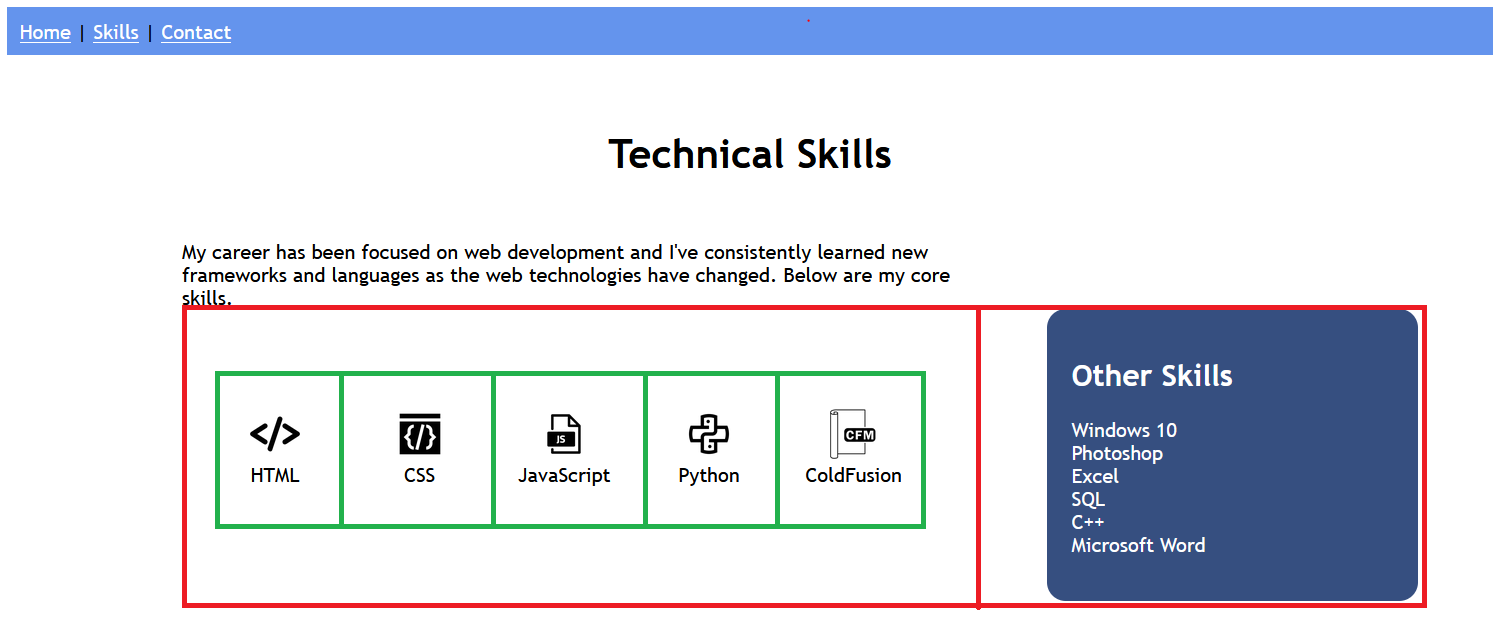
## Overview

This week in lab we are going to change our layout on the skills page. Here’s what our page is going to look like when we are done.

Graphical user interface, application

Description automatically generated

Spend some time thinking through where the rows and columns are in this design. Scroll down when you want to look at the solution I came up with.



The red is a flexbox with two columns. Inside that flexbox in green is another flexbox with five columns. Let’s go through the steps to create this entire page.

## Centering

Table

Description automatically generatedLet’s start with our existing skills page.   
  
We talked about three ways to center horizontally based on the item you are trying to center.

**Inline** - Is your item inline? Then you use the text-align property.

**Block** - Is your item block? Then you can center with margins or flexbox.

1. Add a main heading to the skills page with appropriate text. I called mine Technical Skills. Here’s what my page looks like after I add the heading.
2. A picture containing chart

   Description automatically generatedCenter that heading over the page using one of the techniques we discussed. Note: I want you to practice the techniques we learned in lecture so points will only be allotted for this section if you use one of the techniques. Here’s what my heading looks like after the header is centered.
3. I also added intro text to this page so write a couple of sentences describing your skills page and put them below the heading.

## advanced flexbox

1. We are going to replace the table you previously. You could delete the code for the table, or put HTML comment tags (<!-- -->) around the table to remove it from displaying in your browser screen.
2. Pick your five most important skills and find icons to illustrate those skills. Put them on your page. You could choose icons that show how well you know each skill or icons that illustrate the skill itself. You can find icons with creative commons licenses here:   
   <https://fontawesome.com/icons>   
   or here  
   <https://thenounproject.com>.
   1. Put a line break between your icon and skill name.
   2. Size the icons with a width property. You don’t have to resize in Photoshop for this lab.

Here’s what my page looks like now.  
  
Graphical user interface, text, application, email

Description automatically generated

1. I like to use the technique I showed you in class to create all my flexbox containers. Create a set of empty HTML div tags on the page that we will put our five skills into. This is one row with five columns. That means one div parent tag with 5 div tags nested inside it. Put these empty div tags into your page.
2. Copy and paste each icon with its skill into its corresponding div tag. Nothing is going to change after you do this because we haven’t styled anything yet.
3. Add the style to create a flexbox to the row container. You will probably have to add a class attribute to the div tag that is your flexbox row. Hint: You should only be adding one style declaration (selector, property and value) to your stylesheet.

Here’s what mine looks like when it’s done.

Graphical user interface, application

Description automatically generated

1. Next let’s add the box I labeled “Other Skills”. Underneath the flexbox with your icons add an aside container tag. Aside is one of the semantic HTML container elements.
2. Add the other five skills from your table to the sidebar.
   1. Add the header using an h2.
   2. Add a background color to your sidebar and change the text color if desired.
   3. Recall from the previous lab how to do a curved corners and apply the style for that to your sidebar.

When you are done it should look like this.  
  
Graphical user interface, application

Description automatically generated

1. The aside skills area is meant to show on the right-hand side of the page. To do that we are going to create another flexbox. I’ll refer to this container as flexbox 2 for the rest of this lab. This one will have one row and two columns. That means one div parent tag with 2 div tags nested inside it. Put these empty div tags into your page.
2. Be careful here! Move the skills with the five icons in flexbox 2. Make sure to get all the tags used to create flexbox 1. Then move the aside into the correct container in flexbox. Nothing will change in your page because we haven’t styled it yet.
3. Add a style to specify flexbox 2 as a flexbox. When you are done, your page should look like this.  
     
   Graphical user interface, application

   Description automatically generated
4. Flexbox 2 is all squished to the left side of the screen. What I really want is for the icons to take up ¾ or the page and the aside to get the remaining ¼. Using the property flex or the property flex-basis, make your columns in flexbox 2 the correct size. When you are finished it should look like this.  
     
   Graphical user interface, text, application

   Description automatically generated
5. The icons with skills are all squished together so let’s fix that. Use the justify-content property and play around with the values until you get something that looks like the image below. Be careful which container you apply this style to!  
     
   Graphical user interface, application

   Description automatically generated
6. Spacing is the only thing that separates you from a finished lab. Add margins and padding to finish the design. Here’s a list of the places you should do this:
   1. Minus the navigation, the entire content area has space that separates it from the edge of the browser window.
   2. There’s extra space inside the aside so that the words don’t squish against the side of that box.
   3. There’s space between the sentences you wrote as an intro and the icons.

Graphical user interface, application

Description automatically generated

## Extra credit- 3pts – completely optional

1. In my starting image I had my intro sentences aligned with my icons and not extending to the area over my aside. If you make this happen on your page, then it will earn you 3 extra credit points.   
     
   If you take on this challenge - you must do this by yourself. If you get help from another human being, then you forfeit the extra credit points. Another human being includes me, the TAs, your classmates, other students, a person on the internet, etc.

## What will You hand IN?

Create a word document, pdf or use the "write submission" option in blackboard to provide the following:

1. Could this design have been completed using only one flexbox? Describe why or whynot.

This layout cannot be created with only one flexbox. It consists of a parent flexbox and smaller flexboxes that can be manipulated and moved around. One flexbox would not have the flexibility to move things around into this layout.

1. Submit the url for your skills page.

<https://mbpaninski.github.io/ist263/lab09/week7skills.html>

1. Validate the skills page and submit the validation link.

<https://html5.validator.nu/?doc=https%3A%2F%2Fmbpaninski.github.io%2Fist263%2Flab09%2Fweek7skills.html>

1. Submit answers to the following:
   1. What questions did you have about the lab? What didn't you fully understand?

I am still figuring out how flexboxes work and how to manipulate items with them. I attempted the extra credit but could not figure out how to manipulate the flexboxes for it.

* 1. What was the hardest part of the lab?

Using flexboxes and the sizing properties

* 1. Rate your comfort level with this week's topics.  
     1 ==> I can do this on my own and explain how to do it.  
     2 ==> I can do this on my own without any help.  
     3 ==> I can do this with help or guidance from others.   
      If you choose this level, please indicate HOW this person helped you.
     + - My friend looked over my code to help me figure out why my flexbox was so small at one point.   
         4 ==> I don't understand this at all yet and need extra help.