Prog02: Basic Fixed Fluid Hybrid Layout   25 Points Possible
5/25/2024

| Attempt 1 ⌄ | ◐ In Progress **NEXT UP: Submit Assignment** | 🗨 Add Comment |

~~Unlimited Attempts Allowed~~

⌄ Details

PURPOSE: to expose the student to the power of fixed / fluid /web / hybrid design layout. Learning about Fixed, Fluid, and Hybrid layouts in web development offers several benefits to a web developer:
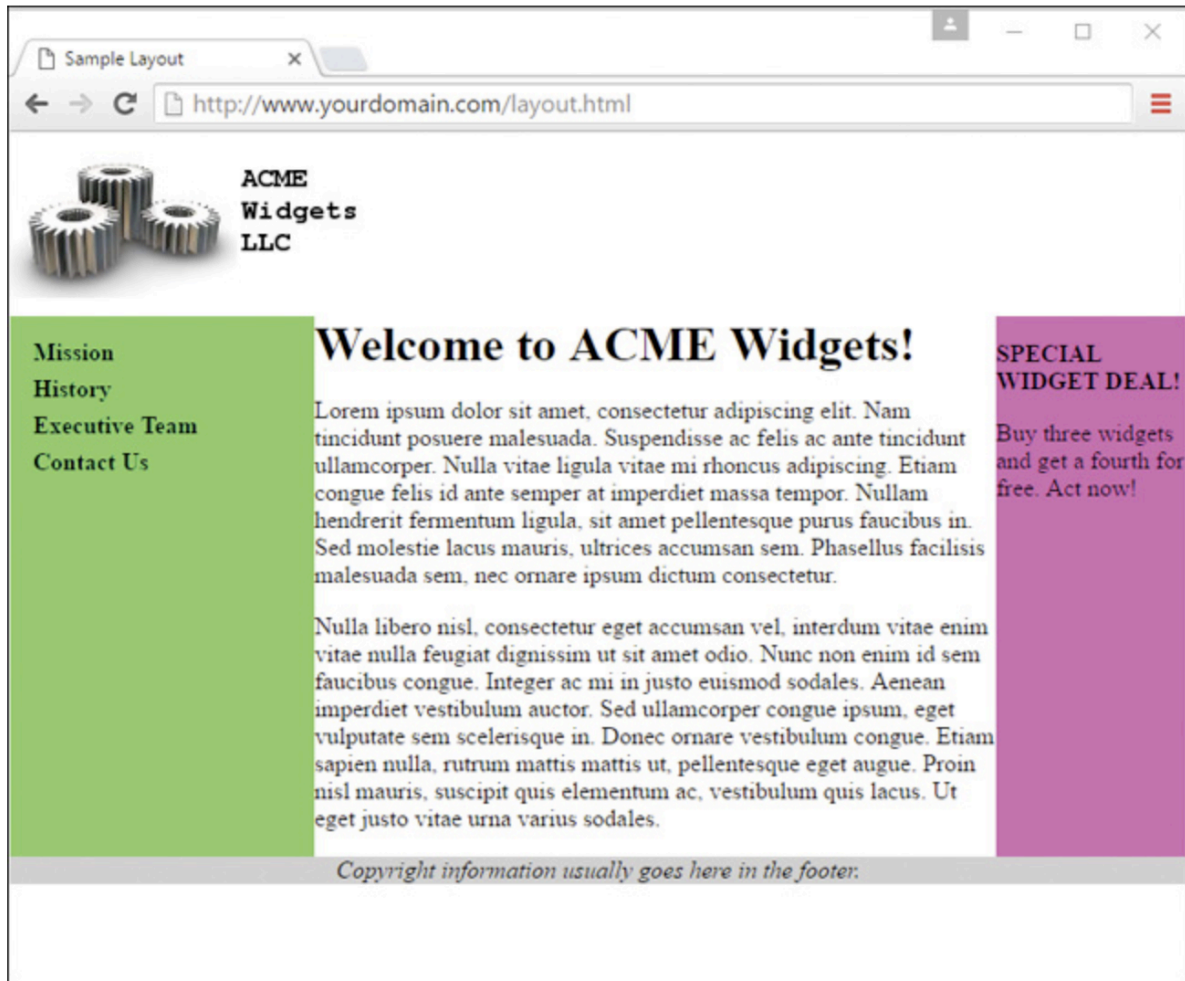
1. **Adaptability**: Understanding different layout types allows developers to create websites that adapt to various screen sizes and devices.

2. **User Experience**: Implementing responsive design techniques improves user experience by ensuring content is displayed optimally on different devices.

3. **Cross-Device Compatibility**: Knowledge of different layouts ensures that websites look good and function well across a range of devices, from desktops to smartphones.

4. **Marketability**: Proficiency in creating responsive layouts enhances a developer's marketability as responsive design is widely sought after in the industry.

5. **Future-Proofing**: With the increasing use of mobile devices, learning about layouts that cater to different screen sizes ensures that websites remain relevant and accessible in the future.

Overall, mastering Fixed, Fluid, and Hybrid layouts equips web developers with the skills needed to create modern, user-friendly, and adaptable websites.

Figure 3.17 (shown below) shows the finished fixed/fluid hybrid layout, but notice a few areas for improvement: There isn't any space around the text in the right-side column, there aren't any margins between the body text and either column, the footer strip is a little sparse, and so on. Fix these design elements. And change those colors to be "Clemson Tigers" compliant!

Using the code you fixed in the preceding exercise, try to make it responsive, using only the brief information you learned in this chapter. Just converting container elements to relative sizes should go a long way toward making the template viewable on your smartphone or other small device, but a media query and alternative style sheet certainly wouldn't hurt, either.

NOTE: the source code for this webpage is in this chapter's github repo from chapter 3

FIGURE 3.17

Congratulations! It's a fixed-width/fluid hybrid layout (although you'll want to do something about those colors!).

Add your navigation bar to this page. It should be the same on ALL of your course web pages.

Host the resulting web page on your server with the file name layout.html.

Submit a link to your webpage.

Save your HTML and CSS files in a folder named Prog02 in your GitHub Repository for this course.

Submit a link to this layout folder.

⌄ **View Rubric**

Some Rubric

Criteria	Ratings		Pts
html submitted	**2 pts** **Full Marks**	**0 pts** **No Marks**	/ 2 pts
link work	**2 pts** **Full Marks**	**0 pts** **No Marks**	/ 2 pts
css submitted	**2 pts** **Full Marks**	**0 pts** **No Marks**	/ 2 pts
design elements fixed	**6 pts** **Full Marks**	**0 pts** **No Marks**	/ 6 pts
colors fixed	**2 pts**	**0 pts**	/ 2 pts

Some Rubric

Criteria	Ratings		Pts
	Full Marks	**No Marks**	
randomized link used	**1 pts** **Full Marks**	**0 pts** **No Marks**	/ 1 pts
responsive	**5 pts** **Full Marks**	**0 pts** **No Marks**	/ 5 pts
common navigation bar	**5 pts** **Full Marks**	**0 pts** **No Marks**	/ 5 pts
			Total Points: 0

Enter Web URL

https://

<

(https://clemson.instructure.com/courses/222478/modules/items/4094337)

Assignment
>

(https://clemson.instructure.com/courses/222478/modules/item