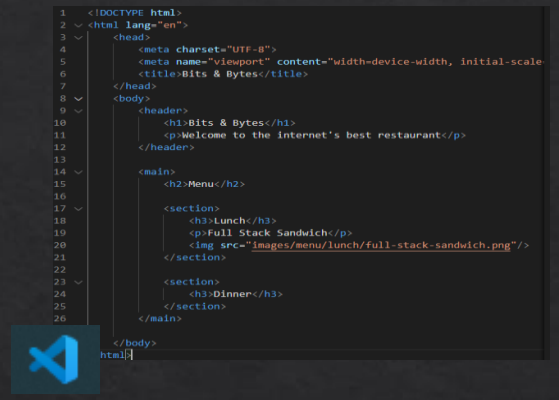
Module 3

Flexbox



Session Objectives:

- Lay out a series of elements using Flexbox
- Define a Flexbox container using a row or a column
- Apply normal flow to Flexbox items using flex-wrap
- Apply content alignment to flex items using justify-content and align-items
- Arrange items within a Flexbox container using order
- Size items within a Flexbox container using flex-basis, flex-grow, flex-shrink
- Understand how to add Flexbox layouts to existing Responsive CSS Grid layouts
- Understand when to use Flexbox or Grid or combine them both

Module 3
Flexbox

What is Flexbox?

Flexbox is a layout option in CSS that provides a flexible way to lay out items on a page. It's been around for several years now and is supported in about 98% of users' browsers.

Flexbox dynamically handles alignment and the space between the items without much calculation on the developer's part. It consists of a flex container that contains one or more flex items.

Typically, you don't use Flexbox to lay out an entire page, but to lay out a group of like elements on a part of the page.

Module 3
Flexbox

When to use Flexbox or Grid

Main considerations when trying to determine which technology to use. While both CSS Grid and Flexbox appear to compete with each other, the use cases they support differ. Here are some questions you should ask yourself:

1. Is the layout two-dimensional or one-dimensional?

CSS Grids help with two-dimensional layouts—meaning rows and columns—while Flexbox helps with one-dimensional layouts—that is, either rows or columns. But what does this mean?

For any layout that has symmetry in both the vertical and horizontal orientations at the same time, Grid is the best choice. One reason is that Grid is more performant for this type of structure. Another reason is that Grid provides a way of spacing rows and columns—called gap—that Flexbox doesn't have. It's difficult to provide this spacing manually and requires a lot of CSS to accomplish.

The strength of Flexbox is its ability to organize, align, and adjust related content. Specifically, when considering how information or components tell a story together as a component, Flexbox is generally the answer.

Module 3
Flexbox

When to use Flexbox or Grid

Main considerations when trying to determine which technology to use. While both CSS Grid and Flexbox appear to compete with each other, the use cases they support differ. Here are some questions you should ask yourself:

2. Is the design based upon content or structure?

Think of a house blueprint. If your task is to build the house, then structure is important. An architect is responsible for the overall task of laying out the house and how the spaces relate to one another. An interior designer is responsible for focusing on each room of that house blueprint and thinking about how to arrange the furniture.

Solving problems with CSS means interpreting a user interface design that is much like that blueprint. If the task involves applying CSS to solve a page structure or layout problem, then Grid is the go-to for solving this problem. This is often referred to as "outside in" or "layout first" design.

On the other hand, if the task involves placing, spacing, aligning, and adjusting related content together (furniture in a room) the go-to tool is likely to be Flexbox. This is often referred to as "inside out" or "content first" design.

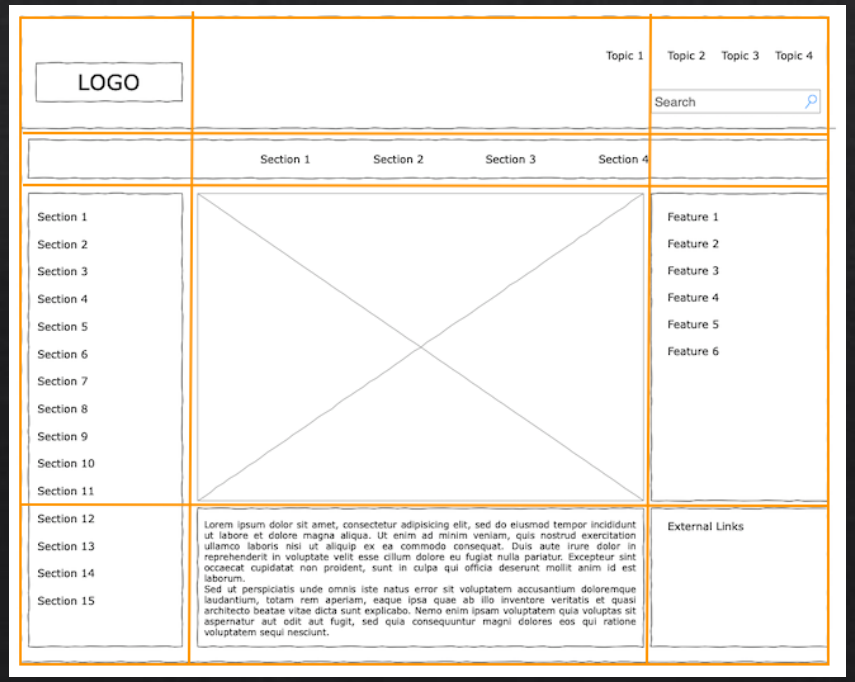
Module 3
Flexbox

When to use Flexbox or Grid

Grid and Flexbox don't have to be used separately. They're often used together to solve problems related to their individual strengths.

Layout symmetry in both rows and columns. There is spacing between the rows and columns. These are strengths of the grid layout.

Grid is most appropriate for this design

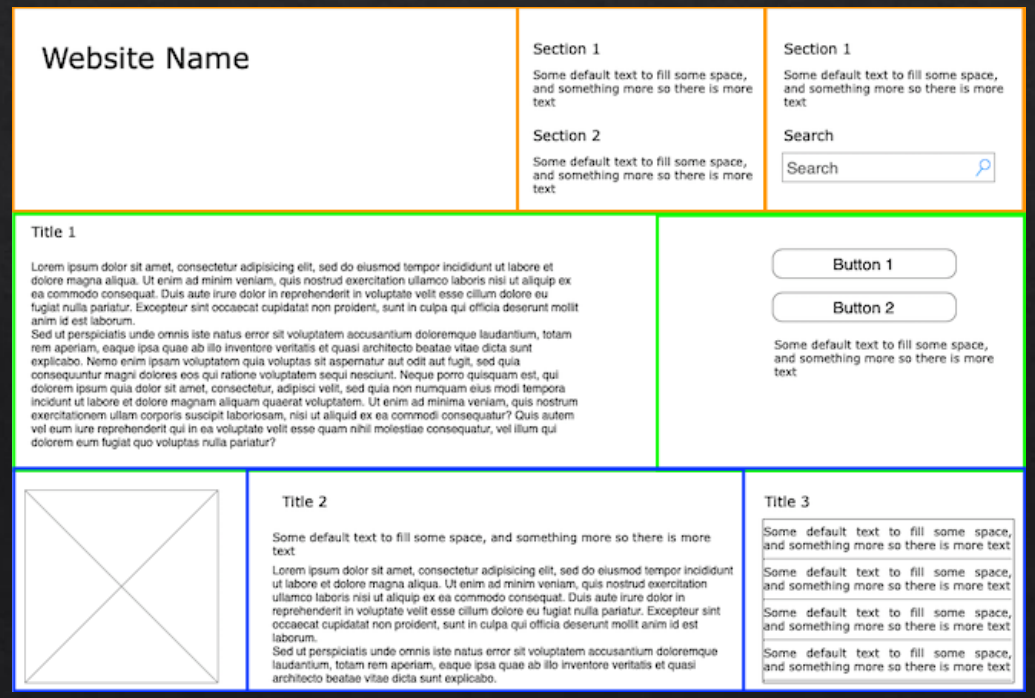


Module 3
Flexbox

When to use Flexbox or Grid

The first thing to notice in this image is the lack of symmetry. This page is seemingly made up of three rows that each have their own individual layout problems to solve. Arranging related content is the strength of Flexbox.

Flexbox is most appropriate for the rows in this design



Module 3
Flexbox

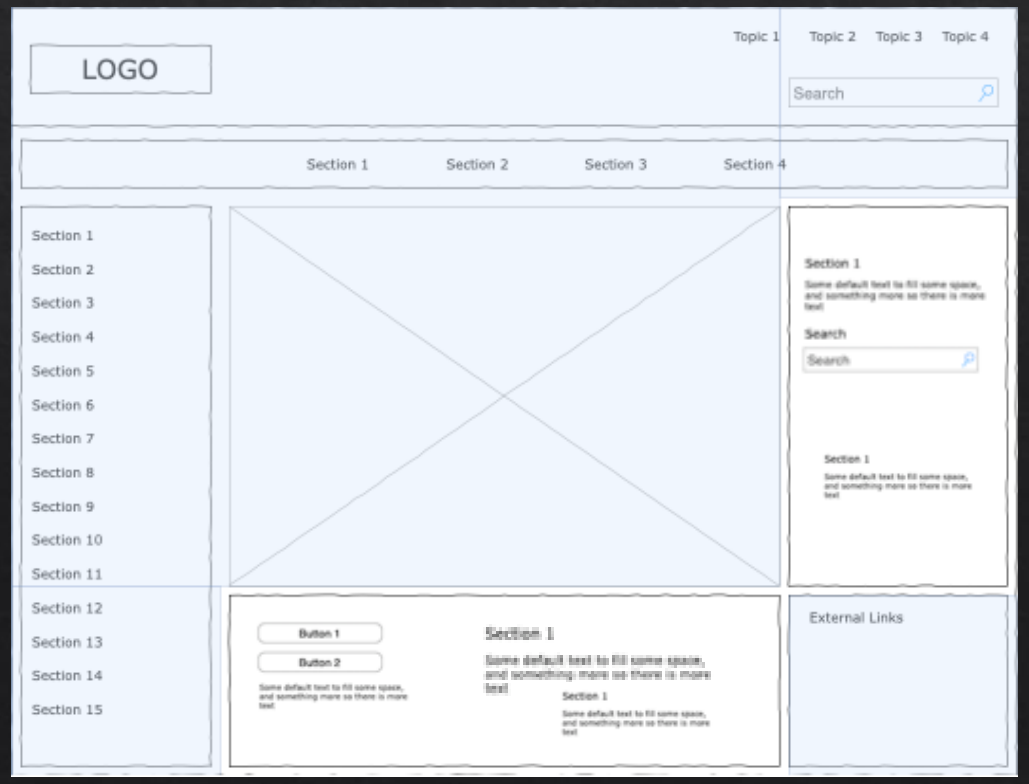
When to combine Flexbox and Grid

Grid and Flexbox don't have to be used separately. They're often used together to solve problems related to their individual strengths.

Grid is for building the layout of the page. Flexbox is for positioning elements relative to each other.

Grid is used for two-dimensional layouts, and Flexbox is used for one-dimensional layouts.

Using them together



Module 3
CSS Grid

New Tools!

Dev Pack