Flexbox

Contents

Flexbox

This is a brief introduction to Flexbox and its many layout functions.

Flexbox is a grid-based layout module used in CSS3. Its goal is to fill up space within a container. Flexbox has two-dimensional and three-dimensional layout capabilities. It can be used for anything from creating an entire page layout to centering an image on the page.

Flexbox provides an efficient way to arrange elements on a page. With Flexbox, items are directed, aligned, ordered, and responsive. There are several Flexbox specific properties that help accomplish these functions.

Create navigation bar with Flexbox in Codepen.io

This task describes how to create a responsive navigation bar with Flexbox. Navigation items will be equally spaced.

Ensure you have an operational understanding of HTML and CSS.

Ensure you have basic understanding of *Flexbox*

Also, before you begin, gather the following information:

- Website page titles
- Page URLs

1. Open Codepen.io
2. Click in HTML text editing window
3. Create an unordered list class tag
4. Define class as navigation
5. Write list items equal to number of pages
6. Assign each list item a page in desired order
7. Input a correlating page link for each list item
8. Insert page title inbetween <a> tags of URL
9. Write a closed unordered list tag
10. Click in CSS text editing window
11. Create class selector titled "navigation"
12. Define list-style as none
13. Define margin as 0
14. Define width as 100%
15. Define *display* as flex
16. Define *justify-content* as "space-around"
17. Define *flex-flow* as "row wrap"
18. Create class selector titled "navigation a"
19. Define text-decoration as none
20. Style navigation bar as desired

Reorder flex-items using Flexbox in Codepen.io

Flexbox's order property allows you to change the visual order of content but keep source order intact. This task shows how to reorder flex-items. One use case is repositioning a sidebar above main content.

Ensure you have an operational understanding of HTML and CSS.

Ensure you have basic understanding of *Flexbox*

Also, before you begin, gather the following information:

- Type of items (navigation bar elements, header, aside, footer)
- Number of items
- Desired styling of items

1. Open Codepen.io
2. Click in HTML text editing window
3. Create an unordered list class tag
4. Define unordered list class as "flex-container"
5. Create desired number of list class items
6. Define each list class as "flex-item"
7. Assign each list item a number inbetween open and close tag
8. Write a closed unordered list tag
9. Click in CSS text editing window
10. Create class selector titled "flex-container"
11. Define list-style as none
12. Define margin as 0
13. Define *display* as flex
14. Define *justify-content* as "space-around"
15. Define *flex-flow* as "row wrap"
16. Create class selector titled "flex-item:nth-of-type" for each flex item
17. Define each nth-of-type as corresponding flex-item number
18. Define order for each item as desired numerical order
19. Create class selector titled "flex-item"
20. Style each flex item as desired within "flex-item" class

Using Flexbox's flex-grow property in Codepen.io

This task describes how to use Flexbox's flex-grow property. Use it to control how left over space is distributed between items.

Ensure you have an operational understanding of HTML and CSS.

Ensure you have basic understanding of *Flexbox*

Also, before you begin, gather the following information:

- Type of items (columns, buttons, table elements, navigation bar elements)
- Number of items
- Desired styling of items

1. Open Codepen.io
2. Click in HTML text editing window
3. Create an unordered list class tag
4. Define unordered list class as "flex-container"
5. Create desired number of list class items
6. Define each list class as "flex-item"
7. Assign each list item a number inbetween open and close tag
8. Write a closed unordered list tag
9. Click in CSS text editing window
10. Create class selector titled "flex-container"
11. Define list-style as none
12. Define margin as 0
13. Define *display* as flex
14. Define *justify-content* as "space-around"
15. Define *flex-flow* as "row wrap"
16. Define *align-content* as "stretch"
17. Create class selector titled "flex-item:nth-of-type" for each flex item
18. Define each nth-of-type as corresponding flex-item number
19. Define *flex-grow* for each flex-item
    a) Define flex-grow as 1 for all normal sized flex-items
    b) Define flex-grow as number greater than 1 for desired flex-item
    This flex-item will be larger than the rest
20. Create class selector titled "flex-item"
21. Style each flex-item as desired within "flex-item" class

Basic Flexbox Properties

This is a reference table defining different Flexbox properties. Each property has an associated attribute, which creates a unique function in Flexbox.

Property	Attribute	Description
display	flex	Defines flex containter
justify-content	space-around	Evenly distributes flex items with equal space around them
flex-wrap	row wrap	Allows flex items to wrap as needed
order	integer	Controls order in which flex items appear in flex container
align-content	stretch	(default) Content stretches to take up remaining space on page
flex-grow	integer	Defines ability of flex item to grow if necessary