INTRO TO FLEXBOX.

Poggie

02.23.2024

// TODO.

Hathon prizes

Test 1 advice

Lab 1 grades update

Intro to flexbox

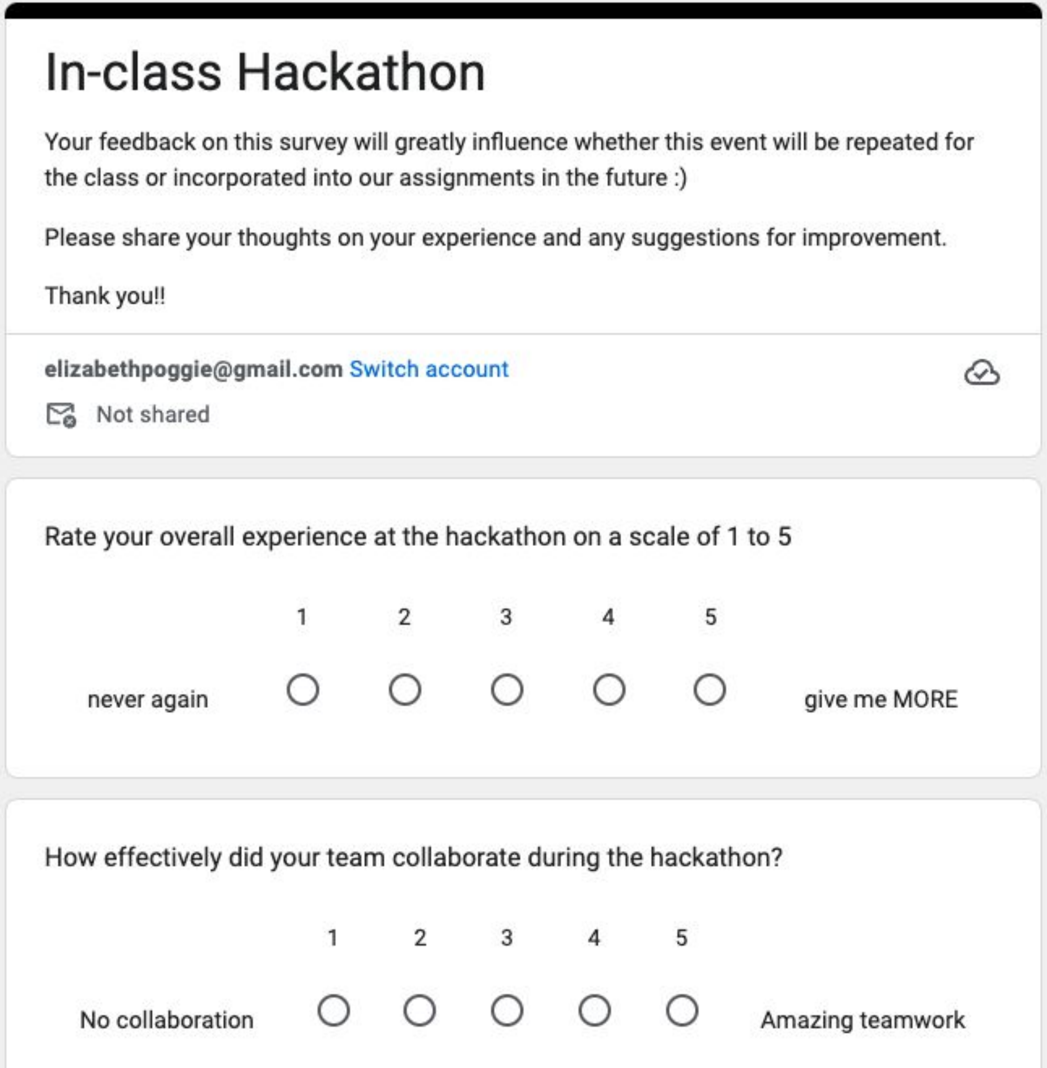
HACKATHON PRIZES.

1st

2nd

1st
2nd
3rd

FEEDBACK.



In-class Hackathon

Your feedback on this survey will greatly influence whether this event will be repeated for the class or incorporated into our assignments in the future :)

Please share your thoughts on your experience and any suggestions for improvement.

Thank you!!

elizabethpoggie@gmail.com Switch account

Not shared

Rate your overall experience at the hackathon on a scale of 1 to 5

	1	2	3	4	5	
never again	○	○	○	○	○	give me MORE

How effectively did your team collaborate during the hackathon?

	1	2	3	4	5	
No collaboration	○	○	○	○	○	Amazing teamwork

TEST 1.

What is the format.

This will be **a closed-book/computer** test with no access to notes permitted.

You will have approximately **2 hours** to complete the exam.

NO VSCODE.

Initially, I had intended to grant access to VsCode for the test, considering this is not a memorization-oriented course.

However, I realized that VsCode provides easy access to MDN Web Docs, which would compromise the integrity of the exam.

Regarding CSS properties, if any are required for the test, I will provide reminders of the property names as needed.

Types of short answer questions to expect:

Explain the difference between CSS vs HTML vs JavaScript

Understand the anatomy of a tag

Why use DevTools? What are DevTools manipulating?

Understand the relationship of different HTML elements with each other

Types of long answer questions to expect:

If presented with a code snippet, you should be capable of determining the visual output generated by the code.

Similarly, if given broken code, you should be able to tell me where the error is.

In summary.

It is not my intention to create any surprises.

The course notes (Lectures 1-8) covered all the theory you could need :)

The practical experience from Lab 1, Lab 2, and the Hackathon have equipped you with amble in-class exposure to the material.

LAB 1 GRADES.

…you should be receiving them shortly!

FLEXBOX.

Intro.

Over the years, the evolution of web design has witnessed the emergence of innovative techniques while some older approaches have faded into obscurity.

Among these advancements, Flexbox stands out as a relatively recent addition to CSS. (~2013)

Intro.

Flexbox, short for "Flexible Box Layout," is a layout model in CSS designed for creating dynamic and responsive layouts.

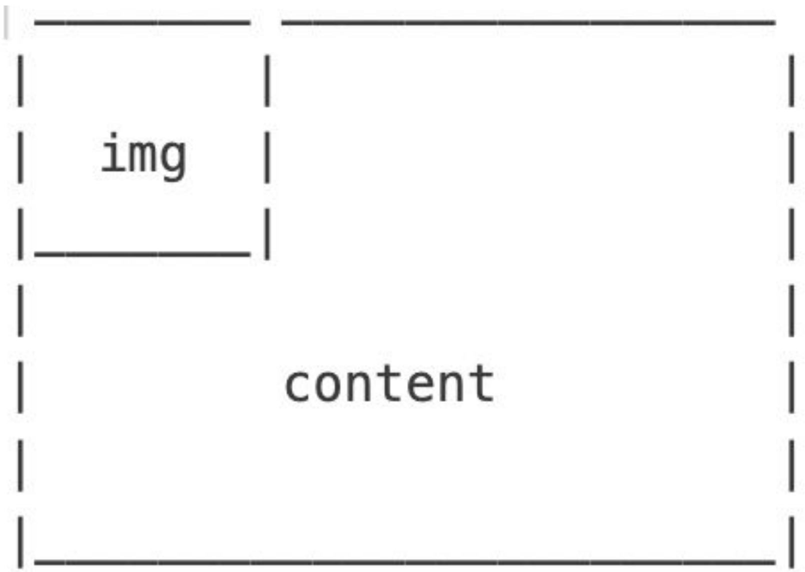
With Flexbox reduces the need for convoluted HTML structures and offers better support for various screen sizes and devices.

Intro.

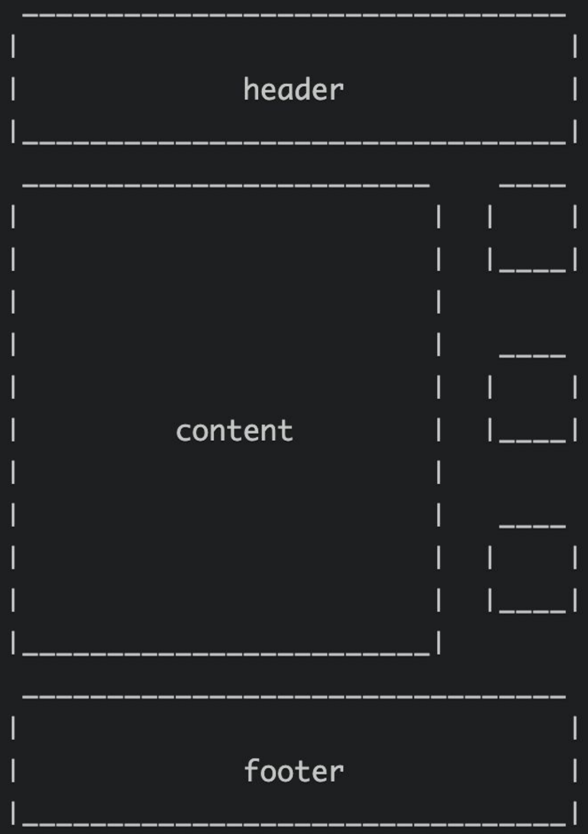
Whereas floats only let us horizontally position our boxes

Flexbox gives us complete control over the alignment, direction, order, and size of our boxes !!

When to use floats and when to use Flexbox?



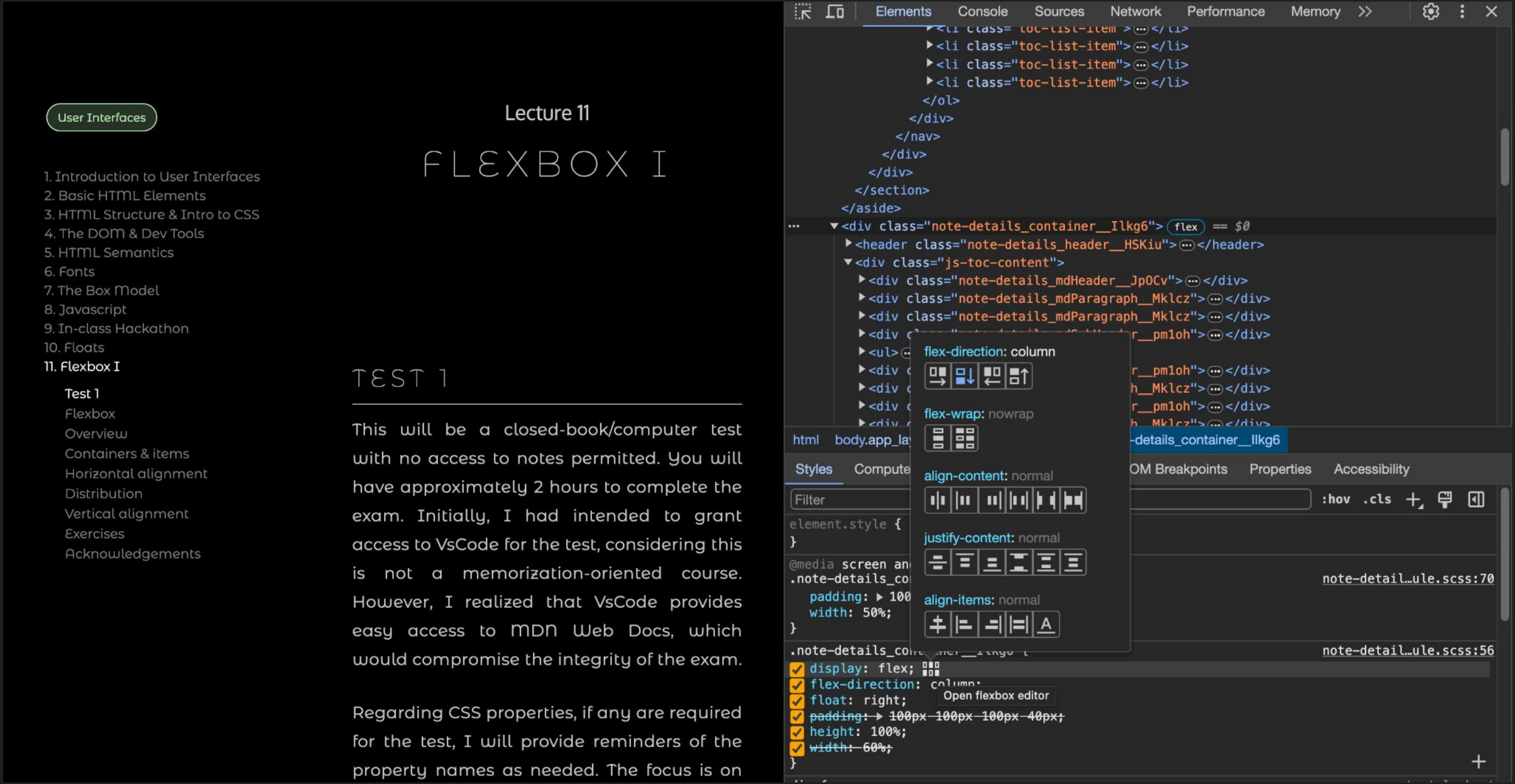
```
 _________  ____________________
|         |                     |
|   img   |                     |
|_________|                     |
|                               |
|            content            |
|                               |
|_______________________________|
```



```
 ______________________________
|                              |
|            header            |
|______________________________|
 ______________________     ____
|                      |   |    |
|                      |   |____|
|                      |
|                      |    ____
|                      |   |    |
|        content       |   |____|
|                      |
|                      |    ____
|                      |   |    |
|                      |   |____|
|______________________|
 ______________________________
|                              |
|            footer            |
|______________________________|
```

In 2 weeks, you should be able to create pretty much any layout.

GETTING STARTED.



User Interfaces

Lecture 11

FLEXBOX I

1. Introduction to User Interfaces
2. Basic HTML Elements
3. HTML Structure & Intro to CSS
4. The DOM & Dev Tools
5. HTML Semantics
6. Fonts
7. The Box Model
8. Javascript
9. In-class Hackathon
10. Floats
11. Flexbox I
    - Test 1
    - Flexbox
    - Overview
    - Containers & items
    - Horizontal alignment
    - Distribution
    - Vertical alignment
    - Exercises
    - Acknowledgements

TEST 1

This will be a closed-book/computer test with no access to notes permitted. You will have approximately 2 hours to complete the exam. Initially, I had intended to grant access to VsCode for the test, considering this is not a memorization-oriented course. However, I realized that VsCode provides easy access to MDN Web Docs, which would compromise the integrity of the exam.

Regarding CSS properties, if any are required for the test, I will provide reminders of the property names as needed. The focus is on

```
            ▶<li class="toc-list-item">⋯</li>
            ▶<li class="toc-list-item">⋯</li>
            ▶<li class="toc-list-item">⋯</li>
            ▶<li class="toc-list-item">⋯</li>
           </ol>
          </div>
         </nav>
        </div>
       </div>
      </section>
     </aside>
…  ▼<div class="note-details_container__Ilkg6"> flex == $0
      ▶<header class="note-details_header__HSKiu">⋯</header>
      ▼<div class="js-toc-content">
        ▶<div class="note-details_mdHeader__JpOCv">⋯</div>
        ▶<div class="note-details_mdParagraph__Mklcz">⋯</div>
        ▶<div class="note-details_mdParagraph__Mklcz">⋯</div>
        ▶<div c                        r__pm1oh">⋯</div>
        ▶<ul>⋯
        ▶<div c                        r__pm1oh">⋯</div>
        ▶<div c                        h__Mklcz">⋯</div>
        ▶<div c                        r__pm1oh">⋯</div>
        ▶<div                          h__Mklcz">⋯</div>
```

```
element.style {
}
@media screen an
.note-details_co                          note-detail…ule.scss:70
  padding: ▶ 100
  width: 50%;
}
.note-details_cont                        note-detail…ule.scss:56
☑ display: flex;
☑ flex-direction: column;
☑ float: right;
☑ ~~padding: ▶ 100px 100px 100px 40px;~~
☑ height: 100%;
☑ ~~width: 60%;~~
}
```

Open flexbox editor

flex-direction: column

flex-wrap: nowrap

align-content: normal

justify-content: normal

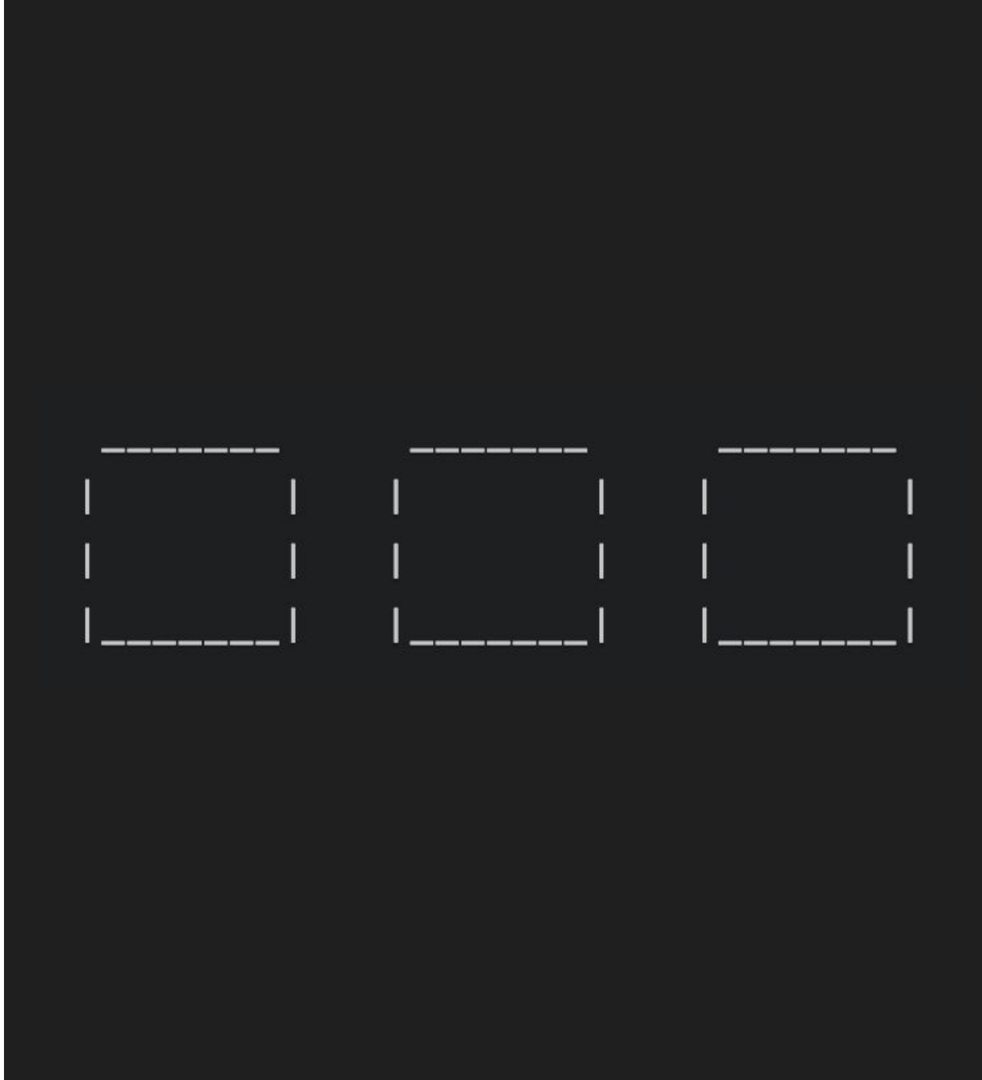
align-items: normal

Overview.

Flexbox is a way to arrange items into rows or columns.

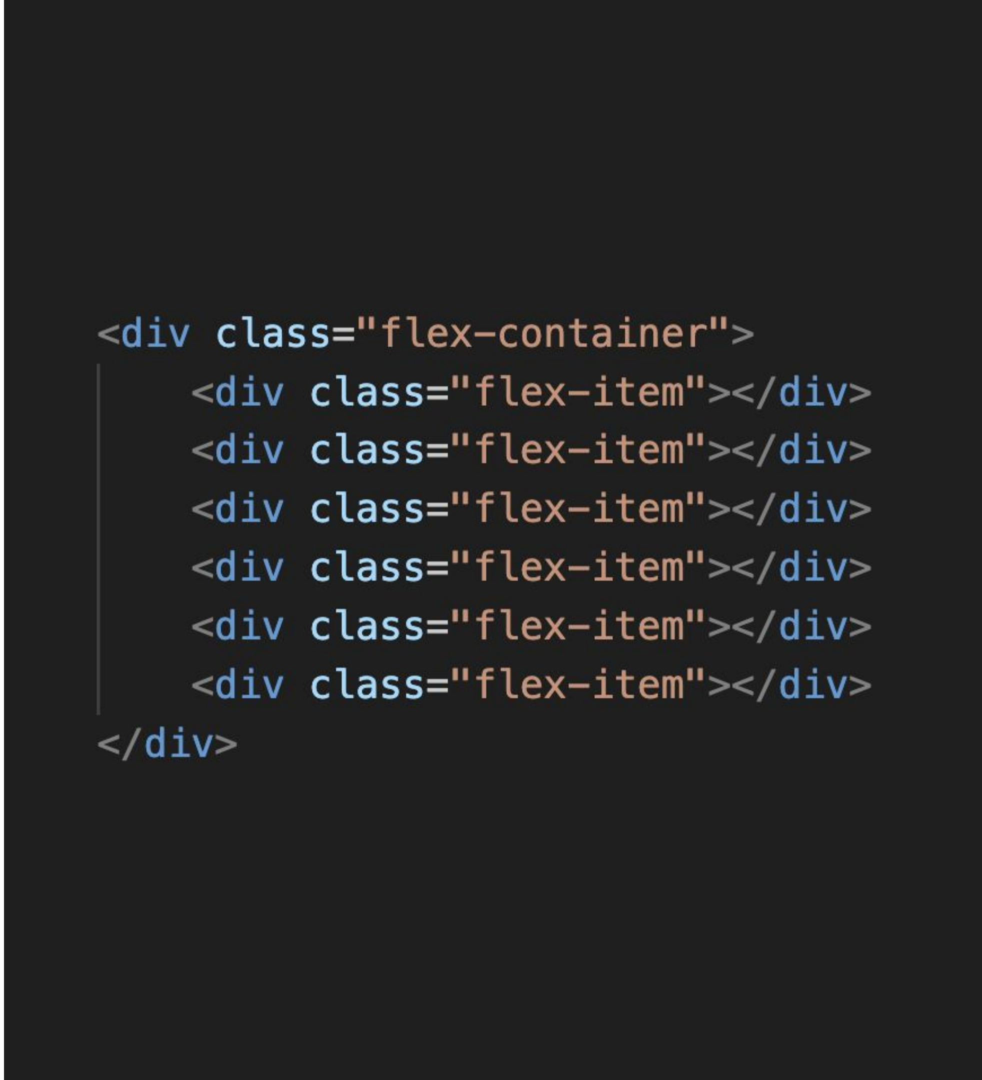
These items will move based on some rules that you can define in the parent element.

Lets try to re-create the following 👉



```
 -------     -------     -------
|       |   |       |   |       |
|       |   |       |   |       |
|_______|   |_______|   |_______|
```

What happens if i add more flex-items?



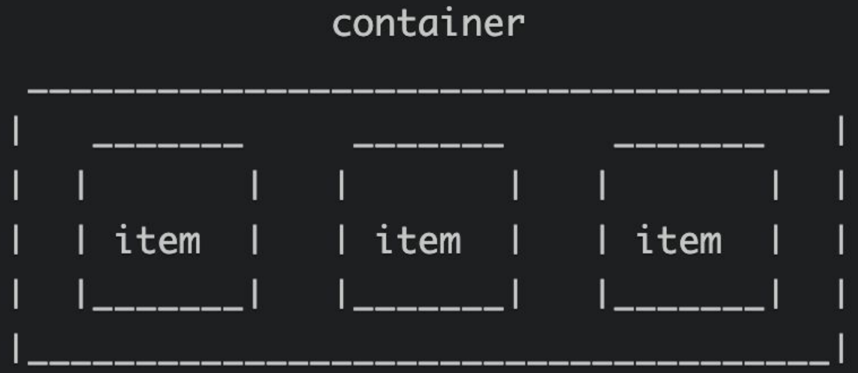
```html
<div class="flex-container">
    <div class="flex-item"></div>
    <div class="flex-item"></div>
    <div class="flex-item"></div>
    <div class="flex-item"></div>
    <div class="flex-item"></div>
    <div class="flex-item"></div>
</div>
```

What’s the difference between containers & items?

Containers & items.

A “flex container” is any element that has “display: flex”

A "flex item" is any element that lives directly inside of a flex container.

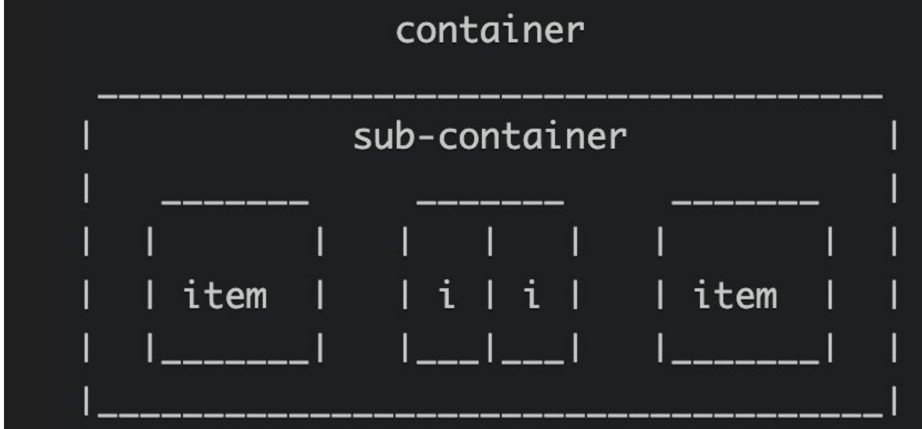


```
                  container
 _________________________________________
|    _______      _______      _______    |
|   |       |    |       |    |       |   |
|   | item  |    | item  |    | item  |   |
|   |_______|    |_______|    |_______|   |
|_________________________________________|
```

…and yes they can be nested!

Any element can be both a flex-container and a flex-item.

In other words, for any flex-item, you can also put “display: flex” on it too !!

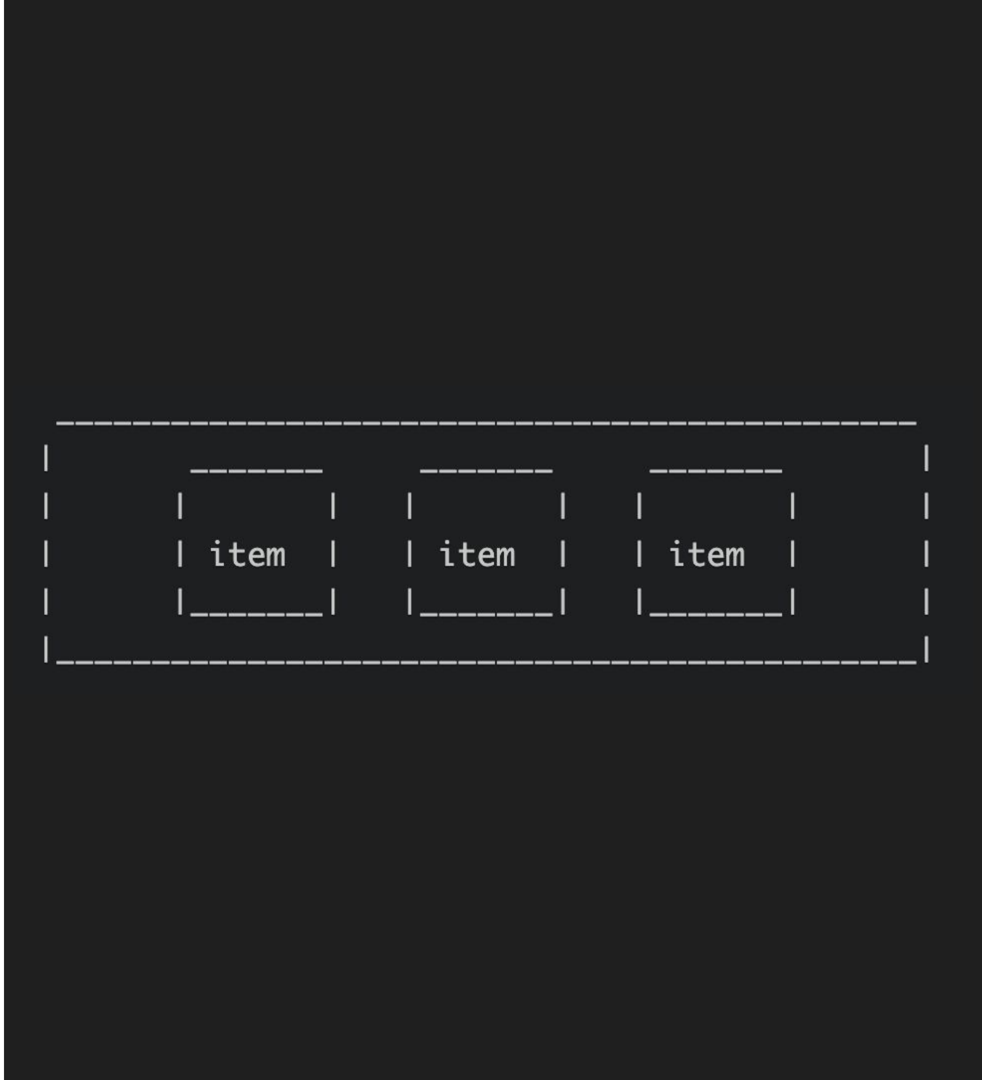


```
               container
 _____________________________________
|            sub-container            |
|   _______     _______     _______   |
|  |       |   |   |   |   |       |  |
|  | item  |   | i | i |   | item  |  |
|  |_______|   |___|___|   |_______|  |
|_____________________________________|
```

ALIGNING OUR ITEMS.

What happens if we want to change how these items are distributed on the page?

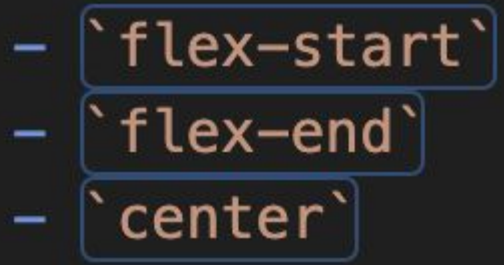
What happens if we want the below layout?



```
 _____________________________________________
|                                             |
|       _______     _______     _______       |
|      |       |   |       |   |       |      |
|      | item  |   | item  |   | item  |      |
|      |_______|   |_______|   |_______|      |
|_____________________________________________|
```

Properties.

The `justify-content` property accepts these values:

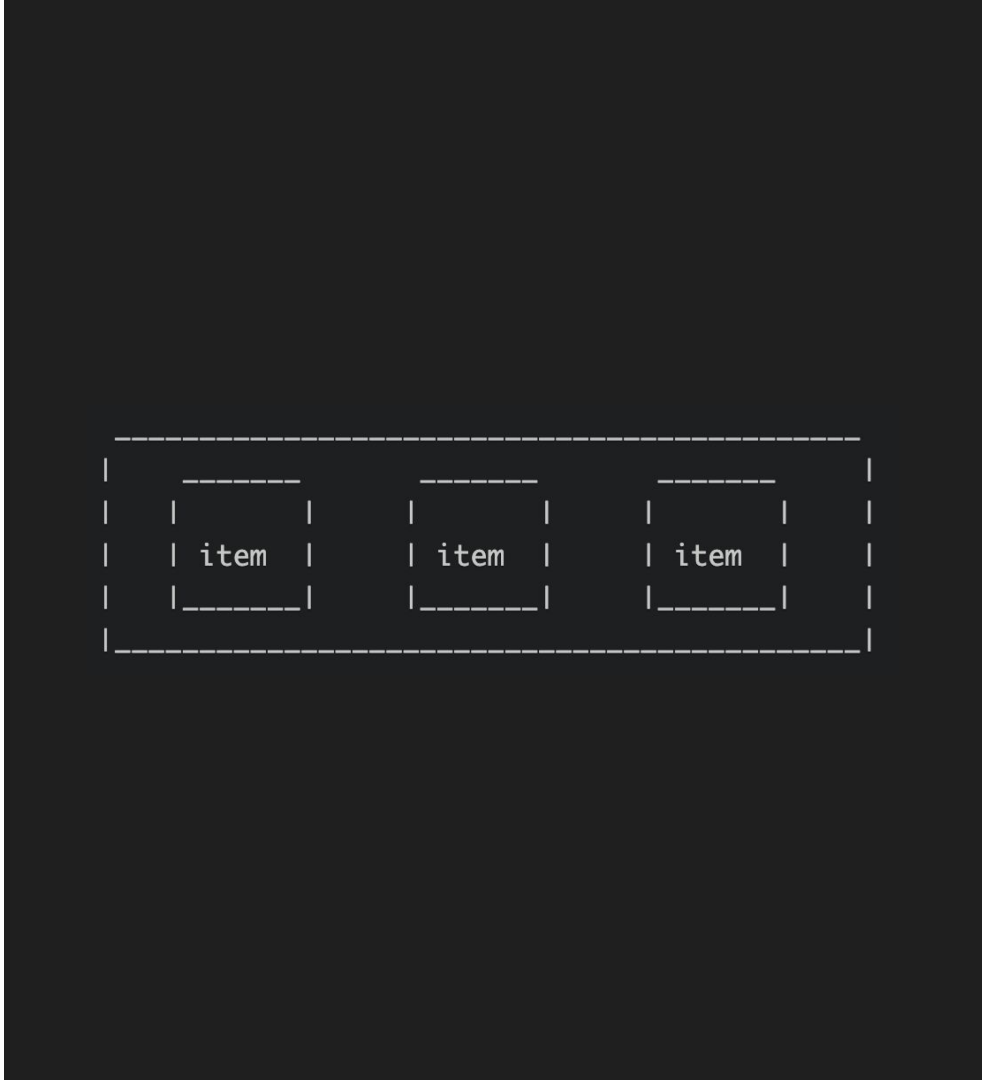


```
- `flex-start`
- `flex-end`
- `center`
```

DISTRIBUTING OUR ITEMS.

What happens if we want our items spread out evenly?

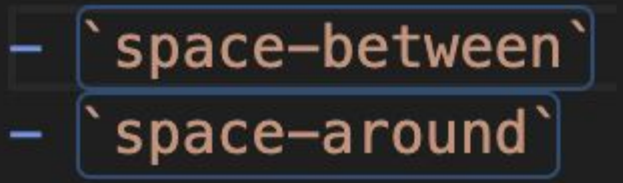
How do we make this layout?



```
 ____________________________________________
|    _______       _______       _______     |
|   |       |     |       |     |       |    |
|   | item  |     | item  |     | item  |    |
|   |_______|     |_______|     |_______|    |
|____________________________________________|
```

MORE Properties.

The `justify-content` property ALSO accepts these values:

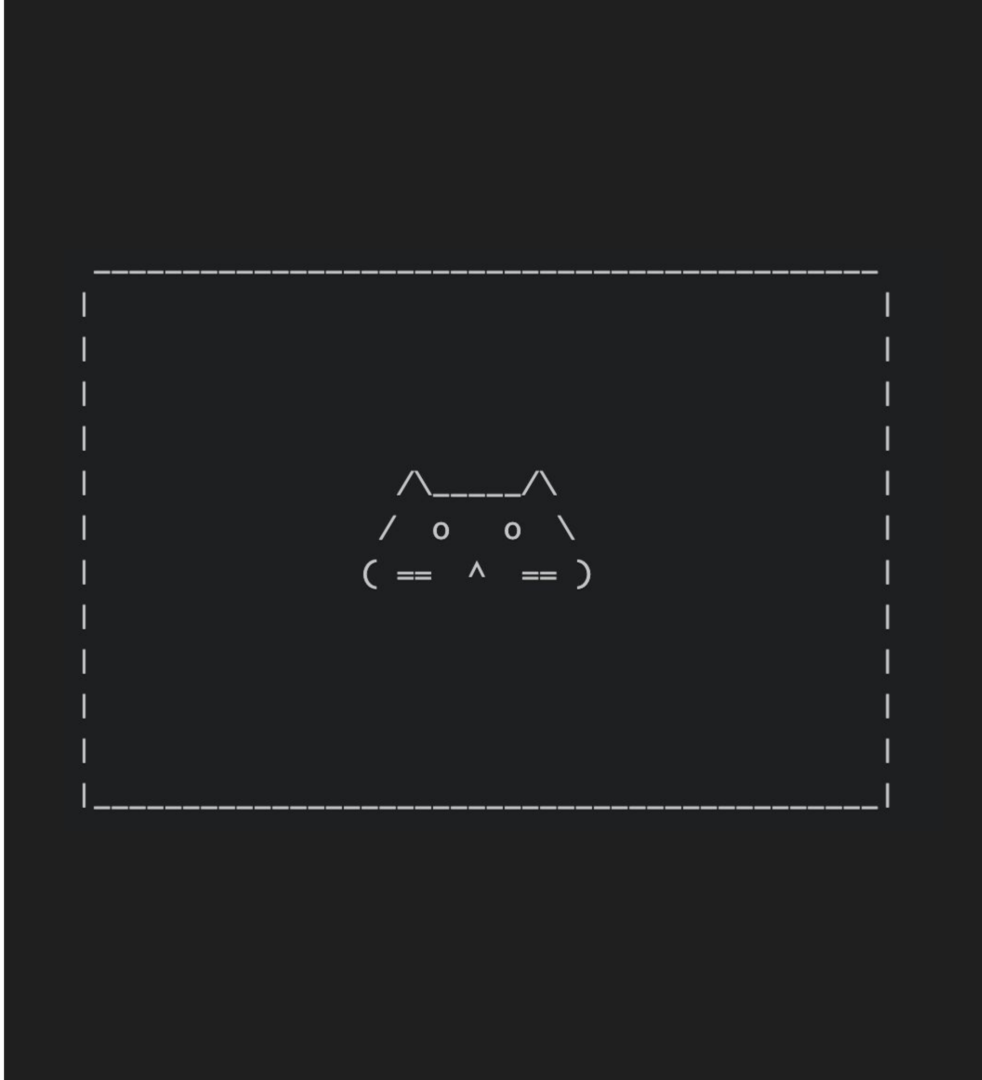


- `space-between`
- `space-around`

VERTICALLY ALIGNING OUR ITEMS.

```
 ____________________________________________
|                                            |
|                                            |
|                                            |
|                                            |
|                 /\_____/\                  |
|                /  o   o  \                 |
|               ( ==  ^  == )                |
|                                            |
|                                            |
|                                            |
|                                            |
|____________________________________________|
```

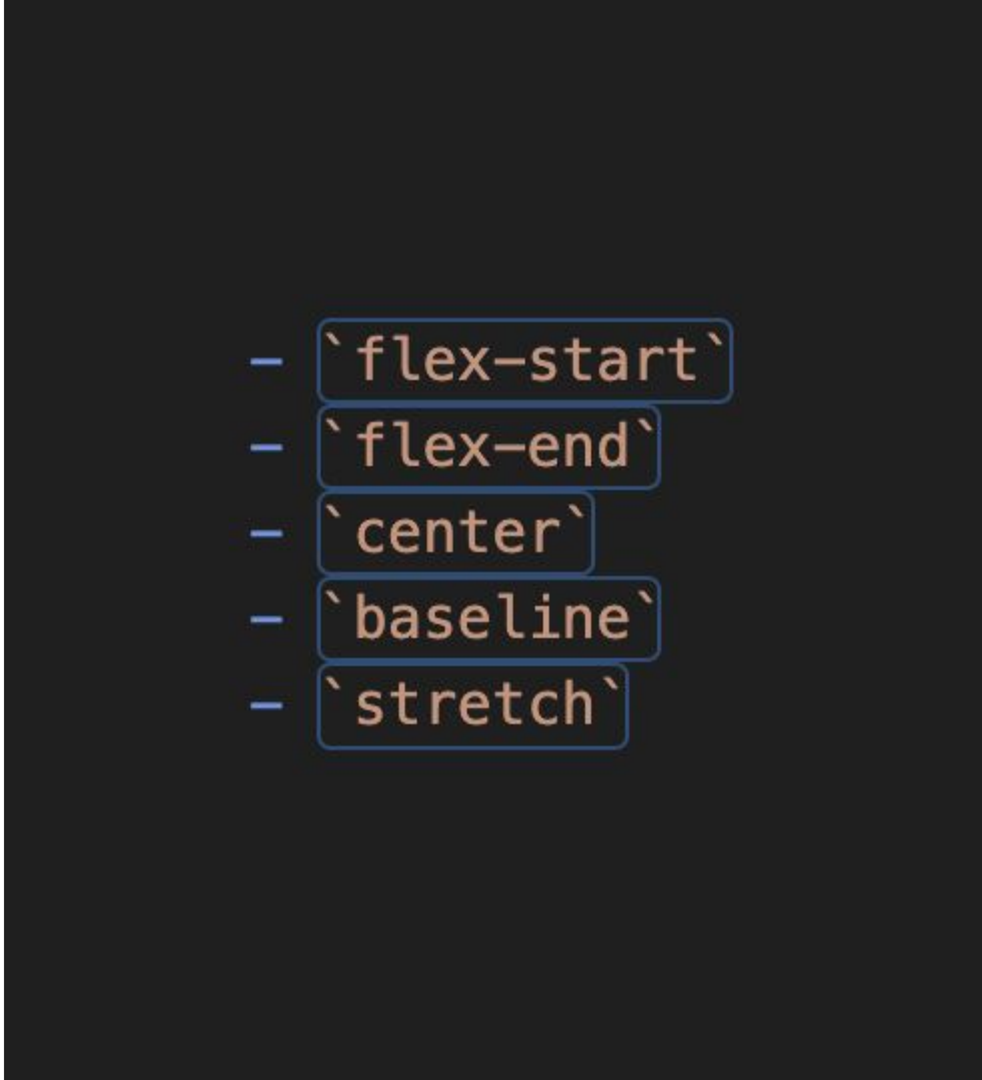
How can we make this?



```
 ____________________________________________
|                                            |
|                                            |
|                                            |
|                                            |
|                 /\_____/\                  |
|                /  o   o  \                 |
|               ( ==  ^  == )                |
|                                            |
|                                            |
|                                            |
|                                            |
|____________________________________________|
```

MOOOOEEEREEEEEEE Properties.

`align-items` accepts the following values:



```
- `flex-start`
- `flex-end`
- `center`
- `baseline`
- `stretch`
```

EXERCISES.