

## Exercise

### Lesson 1 Exercises - HTML Basics

(create new files like “1a.html”, “1b.html”, etc. to try these exercises)

1a. Create a button with the text “Click” inside

Click

1b. Create 2 buttons. 1 with your name and  
1 with your favorite food

Simon chocolate

1c. Create a paragraph with the text “Hello, World!” inside      Hello, World!

1d. Create a paragraph below the previous paragraph and write  
something you did today

Hello, world!

Today I went to the grocery store to buy some eggs and vegetables.

Solutions + more exercises in description



1e. Create a link to google.com (or a similar website for your country)

(don't worry if your link is purple, it just  
means you visited the website before)

[Search with Google](#)



Google

1f. Make the link from the previous exercise (1e.) open in a new tab

### Challenge Exercise

1g. Try to copy this design  
using HTML (the link  
goes to amazon.com)

[Back to Amazon](#)

Nike Black Running Shoes

\$39 - in stock.

Free delivery tomorrow.

[Add to Cart](#) [Buy now](#)

Solutions + more exercises in description

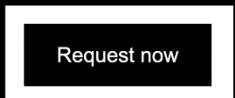


## **Lesson 2 Exercises - CSS Basics**

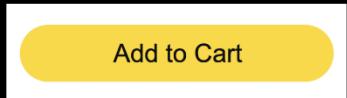
(create new files like “2a.html”, “2b.html”, etc. to try these exercises)

Use CSS to recreate these buttons from famous websites:

2a. Uber



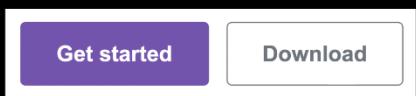
2b. Amazon



2c. GitHub

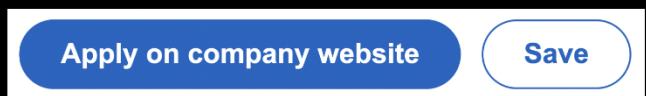


2d. Bootstrap



rgb(121, 82, 179)    rgb(108, 117, 125)

2e. LinkedIn



rgb(10, 102, 194)

**Solutions + more exercises in description**



## **Challenge Exercise**

2f. Continuing from exercise 1g. recreate this design using CSS:

(hint: you'll need the CSS properties: color, font-weight, and font-size)

rgb(0, 113, 133)

[Back to Amazon](#)

rgb(0, 118, 0)

**Nike Black Running Shoes**

\$39 - in stock.

rgb(255, 216, 20)

Free delivery tomorrow.

rgb(255, 164, 28)

Add to Cart

Buy now

**Solutions + more exercises in description**

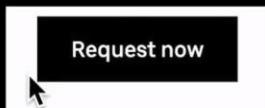


## **Lesson 3 Exercises - Hovers, Transitions, Shadows**

Use CSS to recreate these buttons from famous websites:

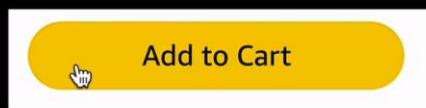
(colors don't have to be perfect, try to pick colors that are close enough)

3a. Uber



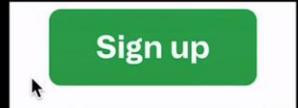
(opacity)

3b. Amazon



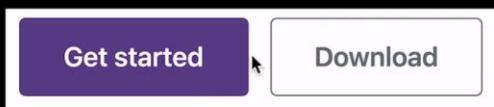
(background color)

3c. GitHub



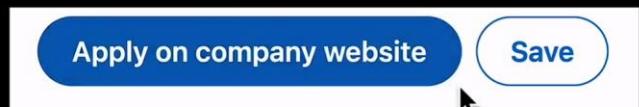
(shadow)

3d. Bootstrap

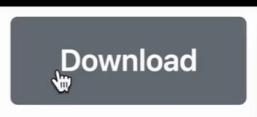
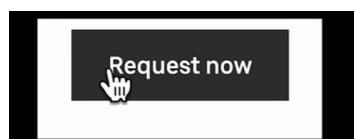


(background color, text color)

3e. LinkedIn



(background color, border width)





## Lesson 5 Exercises - Text Styles

Recreate the following text styles:

(font sizes and colors don't have to be perfect, just close enough)

5a. font = Tahoma

**This is Tahoma Font**

5c. font = Verdana

**HTML CSS Course**

Beginner to Pro

In this course, we'll learn the skills you need to become a developer.

[Get Started](#)

5b. font = Arial

**BIGGEST DEALS OF THE YEAR!**

*Sales end Tuesday*

5d. font = Arial

**SHOPPING FOR YOUR BUSINESS?**

See how Apple at Work can help.

[Learn more >](#)

## Challenge Exercises

All use font-family Arial. (hint: you'll need lots of <span>'s for 5f. and fg.)

5e.



5f.

**1,049.61 USD**  
+18.05 (1.75%) today  
After hours 1,048.00 -1.61 (0.15%)

5g.

**freeCodeCamp.org** @freeCodeCamp 1h

As a web developer, you'll want to make your projects easy to use and navigate around.

Here [@chp\\_it](#) outlines the top skills new developers should have.

## Lesson 6 Exercises - The HTML Structure

Modify some of the previous exercises and:

6a. Add the HTML structure (`<!DOCTYPE html>`, `<html>`, `<head>`, `<body>`)

6b. Add an appropriate `<title>` for each exercise

6c. Move the CSS into a separate file (using the `<link>` element)

---

6d. In a new HTML file, create this design:

### Model 3

Order Online for Touchless Delivery

- Use the HTML structure
- Add a `<title>`
- Use a separate CSS file
- Use the Google Font called "Montserrat" (regular + bold)

Solutions + more exercises in description



## Lesson 7 Exercises - Images and Text Boxes

For these exercises, find any .jpg .png or .webp image on the internet to practice with:

7a. Load the image, set width to 200px and add rounded corners



7b. Make the image a square, center the image within the square



7c. Make the image a circle (using border-radius)



Solutions + more exercises in description



### Create a textbox:

#### 7d. With rounded corners

#### 7e. With a shadow, and no border

### Challenge Exercises

#### 7f. LinkedIn signup

Email

By clicking Agree & Join, you agree to the Privacy Policy.

[Agree & Join](#)

#### 7g. Twitter text box

(hint: use vertical-align: middle on each element to achieve the design below)

 What's happening? [Tweet](#)

Solutions + more exercises in description



## Lesson 8 Exercises - Display Block and Inline-Block

#### 8a. Create 2 text boxes and put them on separate lines (using display: block)

Name

Email

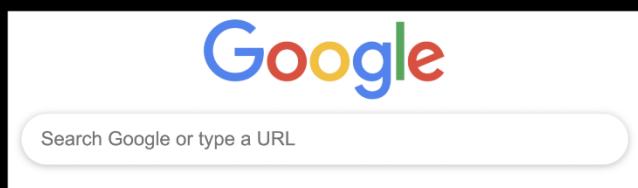
#### 8b. Create a paragraph and 2 buttons and place them on the same line (using width, display: inline-block and vertical-align: middle)

Thanks for chatting with our customer support. Would you like to take our quick survey?

[Yes](#) [No](#)

#### 8c. Download Google's logo and create the google.com search form

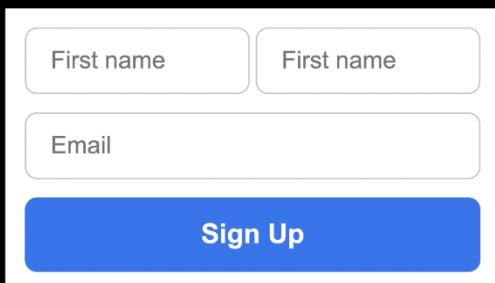
- We created the text box in exercise 7e.
- For now, position the logo with margin (we'll learn better ways to do this later)



## **Challenge Exercises**

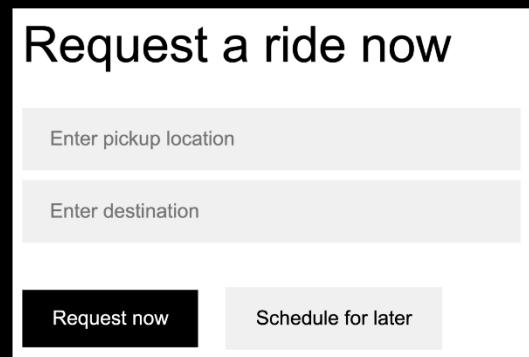
Recreate the following designs:

8d. Sign up form



A sign-up form template consisting of three input fields: 'First name' (two separate fields), 'Email', and a large blue 'Sign Up' button.

8e. Uber request-a-ride form



An Uber request-a-ride form template with two input fields: 'Enter pickup location' and 'Enter destination'. Below the fields are two buttons: 'Request now' and 'Schedule for later'.

**Solutions + more exercises in description**



## **Lesson 9 Exercises - The <div> Element**

First, we'll show that a `<div>` is just a box:

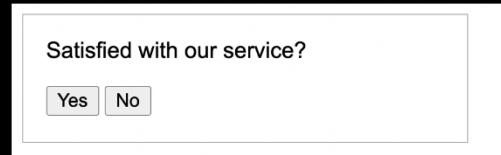
9a. Create a `<div>` and turn it into a red square



9b. Create a `<div>` and turn it into a green circle

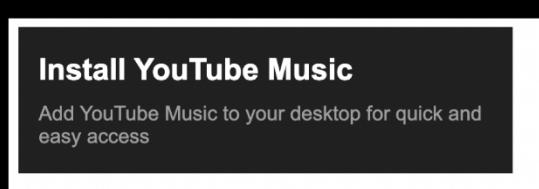


9c. Recreate this design by wrapping the elements in a `<div>` and adding a border



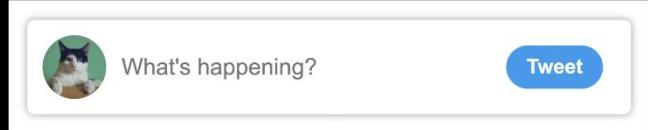
A feedback survey form template with the question 'Satisfied with our service?' and two buttons: 'Yes' and 'No'.

9d. Recreate this design using a `<div>` with 2 paragraphs inside



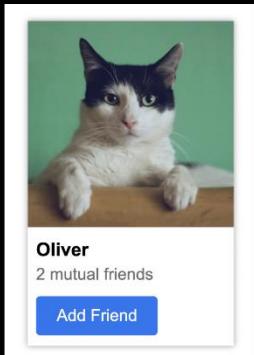
A YouTube Music download card template with the heading 'Install YouTube Music' and the subtext 'Add YouTube Music to your desktop for quick and easy access'.

9e. Take the design created in exercise 7g. Wrap it in a `<div>` and give it a shadow

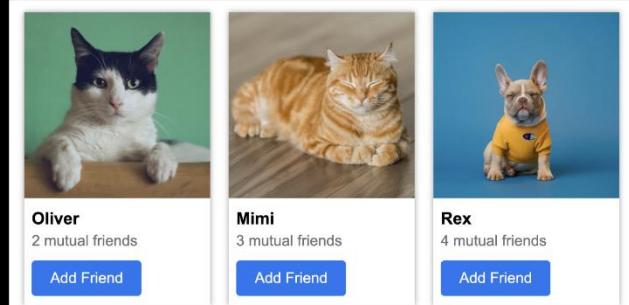


### Challenge Exercises

9f. Recreate this design



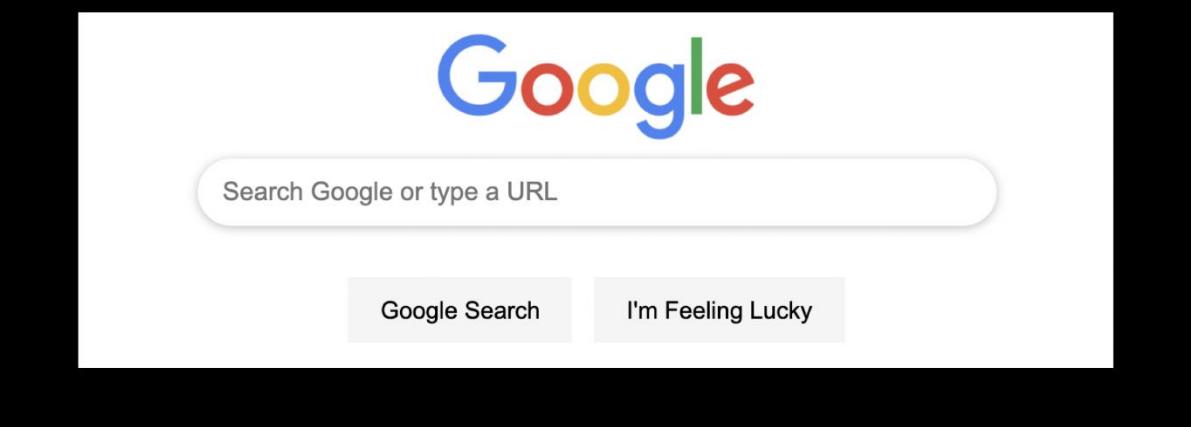
9g. For exercise 9f. set `display: inline-block` on the `<div>`, and duplicate the design a few times (you can use different images than me)



9h. Continuing from exercise 8c. we'll finish the Google Search design

#### To center everything:

- Wrap each line with a `<div>` and use `text-align: center` on the `<div>`
- Remove any uses of `display: block` on the image/input (display: block causes elements to take up the entire line, so they can't be centered)



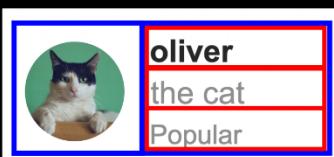
## Lesson 10 Exercises - Nested Layouts Technique

Using an image editing software, break down the following designs into a series of vertical and horizontal layouts (like we did in the lesson)

Download the designs at:

<https://supersimple.dev/css/nested-layouts-technique>

**Here's an example:**



red = vertical layout

blue = horizontal layout

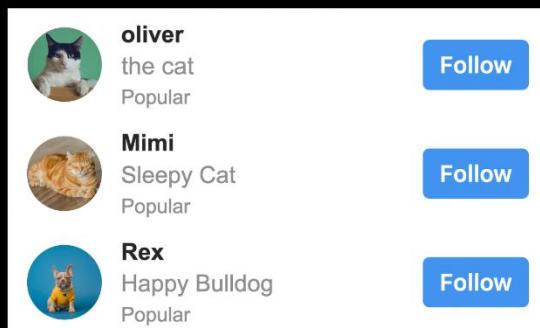
10a. Draw the layout for this design



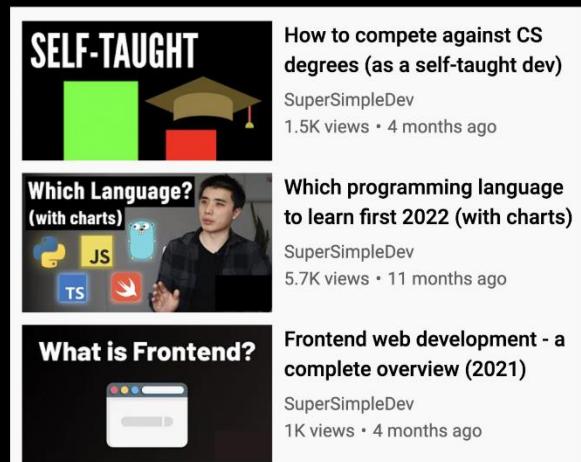
**Solutions + more exercises in description**



10b. Draw the layout



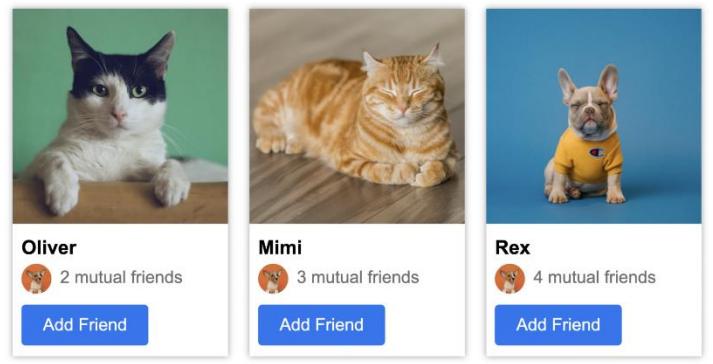
10c. Draw the layout



**Solutions + more exercises in description**



#### **10d. Draw the layout**



### **Challenge Exercise**

**10e. Build the design in exercise 10d. with code (most of this was already built in exercise 9g.)**

- Create vertical layouts with <div>
  - Create horizontal layouts with <div> and display: inline-block;

**Solutions + more exercises in description**



## Lesson 11 Exercises - CSS Grid

11a. Create a grid with 2 columns with widths of 200px and 75px

200px

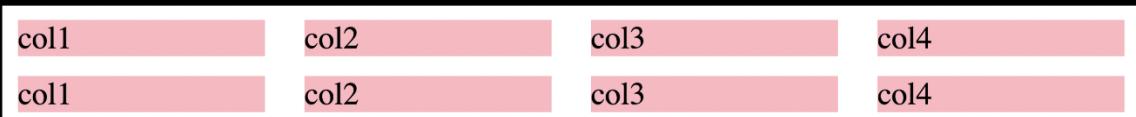
75px

11b. Create a grid with 3 columns. Widths = 50px | remaining space | 75px

col1 col2

col13

11c. Create a grid with 4 equal columns that take up the entire page.  
Space between columns = 20px, space between rows = 10px

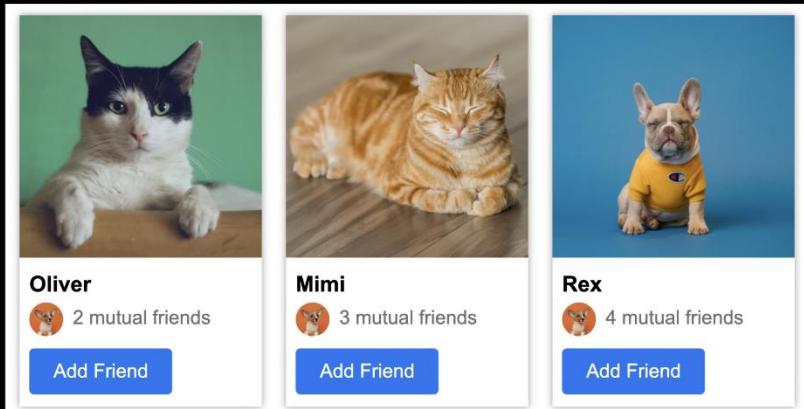


**Solutions + more exercises in description**



11d. Continuing from exercise 10e. create this design using CSS Grid

- Create a grid with 3 columns and put the elements into the Grid
  - Replace any uses of display: inline-block
  - To align elements vertically in a grid, use align-items: center;



## Lesson 12 Exercises - Flexbox

12a. Create a flexbox with 2 items. Widths = 200px | 75px

200px

75px

12b. A flexbox with 3 items. Widths = 50px | remaining space | 75px

item1 item2

item3

12c. A flexbox with 4 items, spread evenly across the page

item1

item2

item3

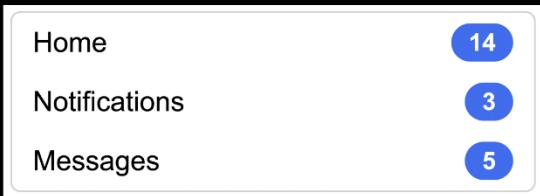
item4

12d. Create the flexbox below with height = 50px and a gray border



## Challenge Exercises

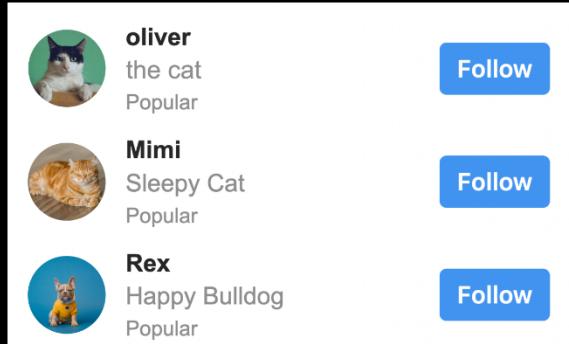
12e. Create this design



12f. Create this design



12g. Create this design



Solutions + more exercises in description



## Lesson 13 Exercises - Nested Flexbox

13a. Create a flexbox with 2 elements (width 100px and 200px). Then make the first element a flexbox to create this design:



13b. Similar to exercise 13a. use nested flexbox (flexbox inside flexbox) to create the design below:



Solutions + more exercises in description



## **Challenge Exercises**

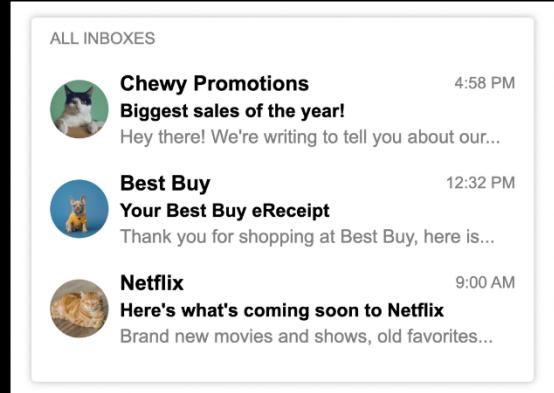
Here's a design from Gmail iOS



**13c. Use nested layouts technique to break down this design into horizontal and vertical layouts**

**13d. Code the design. Use flexbox for horizontal layouts and regular <div>s (display: block) for vertical layouts**

**13e. Make a few copies of exercise 13d. and create this design**

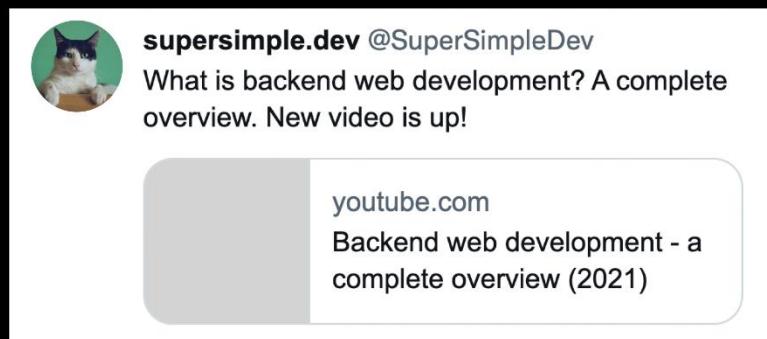


**Solutions + more exercises in description**



Here's a design from Twitter

**13f. Break down this design into horizontal and vertical layouts**



**13g. Code this design using nested flexbox (flexbox inside flexbox)**

**Solutions + more exercises in description**



## **Lesson 14 Exercises - Position Fixed**

For these exercises, set the <body> height to 3000px so you can scroll

14a. Create an element that sticks to the page when scrolling (20px from the bottom and 20px from the right)



14b. Create a sidebar on the right side of the page, width = 100px

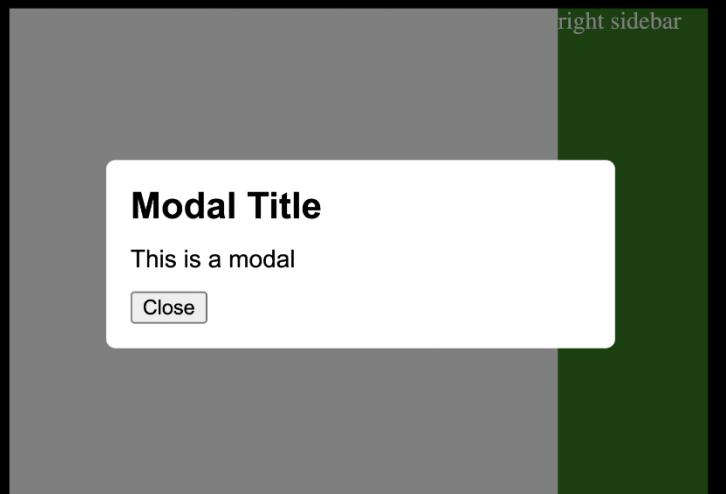


14c. Create an element that covers the entire page (0px on all sides) and has background-color black

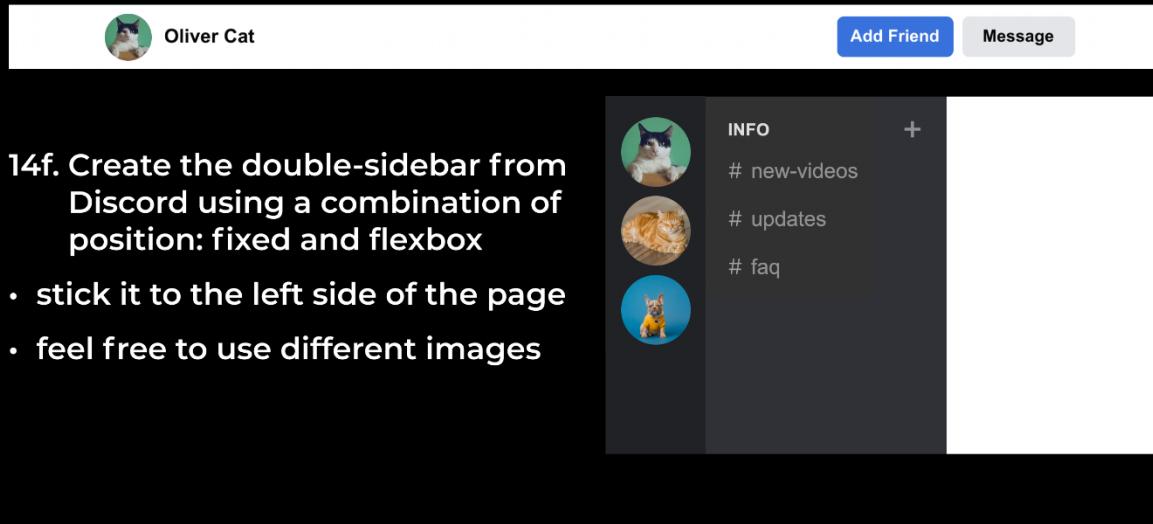
## **Challenge Exercises**

14d. We'll create a modal

- start from exercise 14b.
- create an element that covers the entire page, with background-color: rgba(0, 0, 0, 0.8); (this is called an overlay)
- make the overlay a flexbox and center its contents horizontally and vertically
- add a <div> in the overlay and create the modal



14e. Use a combination of position: fixed and flexbox to create this header from Facebook and stick it to the top of the page (feel free to use a different image/name)



The image shows a screenshot of a Facebook-like header. At the top left is a circular profile picture of a cat named "Oliver Cat". To its right are two buttons: "Add Friend" and "Message". Below the header is a dark sidebar with three circular icons (cat, orange cat, dog) and a "INFO" section containing links: "# new-videos", "# updates", and "# faq". The main content area is blank white space.

### Lesson 15 Exercises - Position Absolute and Relative

For these exercises, set the <body> height to 3000px so you can scroll

15a. Create a “Close Button” like this design and place it at the top-right corner of the page using position: absolute



15b. Using position: fixed, stick a <div> near the bottom-right of the page

- give the <div> a width of 320px and a height of 180px



Solutions + more exercises in description

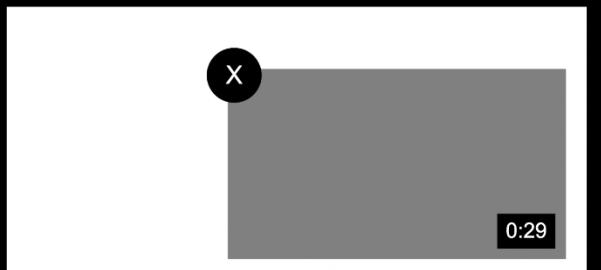


15c. Take the “Close Button” created in exercise 15a. and put it inside the <div> created in exercise 15b.

- Using position: absolute inside position: fixed move the “Close Button” to the top-left corner
- Use negative px for top and left and adjust the sizing if needed



15d. Imagine exercise 15c. is a video ad in the bottom-right corner. Add a timestamp in the corner like this design



### Challenge Exercises

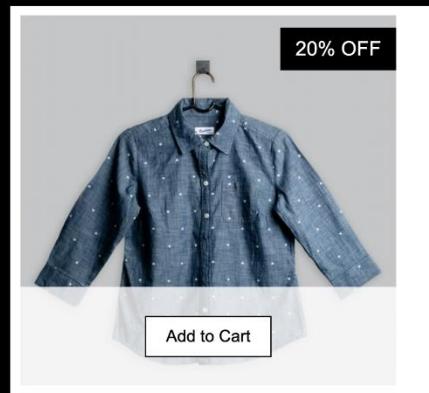
15e. Find an image of clothing online and create the design below

- you'll need position: absolute inside position: relative

- set the position: relative container to display: inline-block



15f. Create the design below



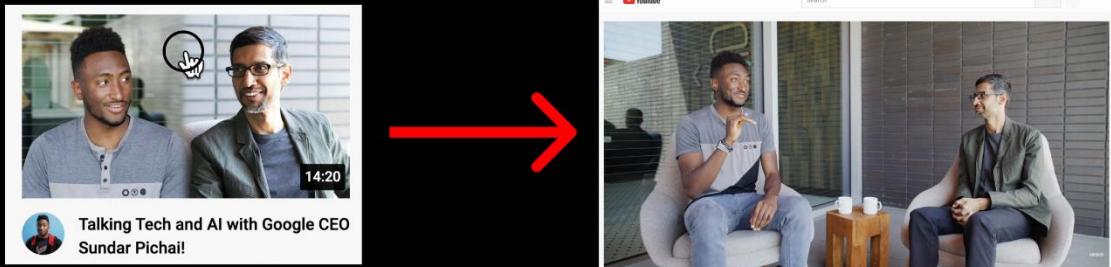
## **Lesson 16 Exercises - Finish the Project**

Congrats, we're done! Some finishing touches you can add for practice:

16a. Add the remaining tooltips in the header

16b. Make the video title a link to the actual YouTube video (you can click the video titles in the final design to get the URL)

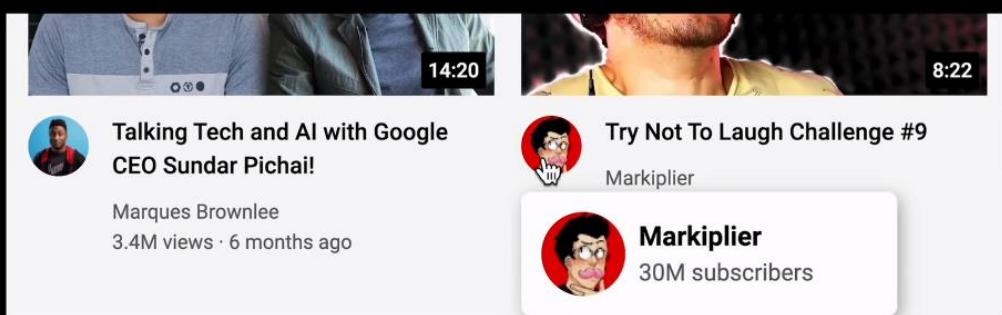
16c. Turn the video thumbnails and channel pictures into links  
(wrap the in an  element)



No 16d, lengkapnya ada di video, bisa di hover

### **Challenge Exercise**

16d. Using everything we learned in this course, create tooltips for the channel pictures like the design below (you'll need to use position, flexbox, hovers, transitions, and shadows)



- Extra: wrap the channel picture in a `<div>` with `display: inline-block`. Use this `<div>` as the `position: relative` element

## Lesson 17 Exercises - More CSS Features

17a. On wider screens, our text is too small. Use @media queries to increase font-size and line-height when the page is > 1000px wide

- For video titles, set font-size = 16px, line-height = 24px
- For other text, set font-size = 14px



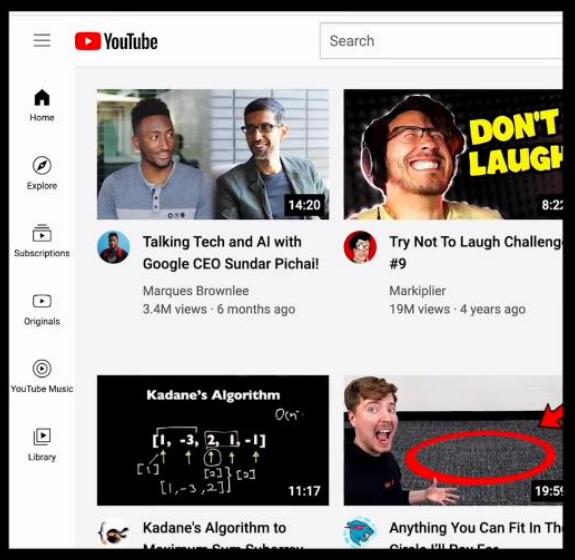
17b ada divideo detailnya

17b. (Optional) For more practice, add the last 6 videos from the final design into our grid so we have 12 videos in total

### Challenge Exercises

17c. Using responsive design, create this expanded sidebar design when the page is  $\geq 1200\text{px}$  wide

- Wrap all the CSS code in sidebar.css in `@media (max-width: 1200px) { ... }`
- Re-style the sidebar for the wider screen and put that code inside `@media (min-width: 1200.1px) { ... }`



Try recreating other well-known websites using the techniques we learned in this course.

17d. Twitter

17e. Facebook

17f. Instagram

17g. YouTube video page (the page with the video and comments)

These are very challenging, but you can do it!

You don't have to match the websites perfectly, just close enough!