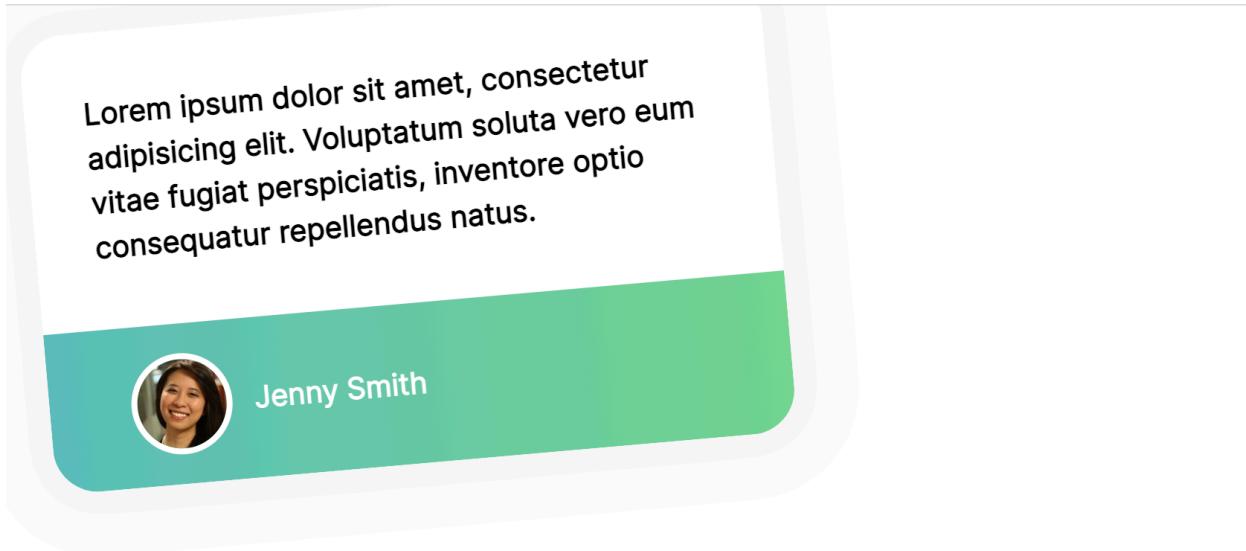


Exercises

Card

Build a card similar to the image below. Read the hints on the next page for more details.



Hints

- Open [/Solutions/Card/index.html](#) and hover over the card to see the animation. Use what you've learned in this section to implement this feature.
- Use the **Inter** font (available on Google Fonts).
- You can get a random user's image from: <https://randomuser.me/photos>
- The image should be 60x60.
- The card is initially rotated 5 degrees counter-clockwise. When we hover on it, it should rotate back to its normal position. The thumbnail image should also be scaled up by 1.15x.
- Use <https://cssgradient.io/> to create a gradient for the footer. The colors I've used are `#74d690` and `#5cbfb9`.
- Give the card a border radius of 30px. Once you set the background for the card footer, you'll see that the round corners are gone. This happens as a result of overflowing because the edges of the footer go beyond the round corners of the card. To bring back the round corners, you need to set **overflow** to **hidden** on the card.
- Add a subtle shadow around the card so it's separated from the page.
 - Shadow offset: `0`
 - Blur: `20px`
 - Spread: `30px`
 - Color: `#f9f9f9`
- The card should have a maximum width of 500px. Use the responsive mode in Chrome DevTools to view the page on various screen sizes.

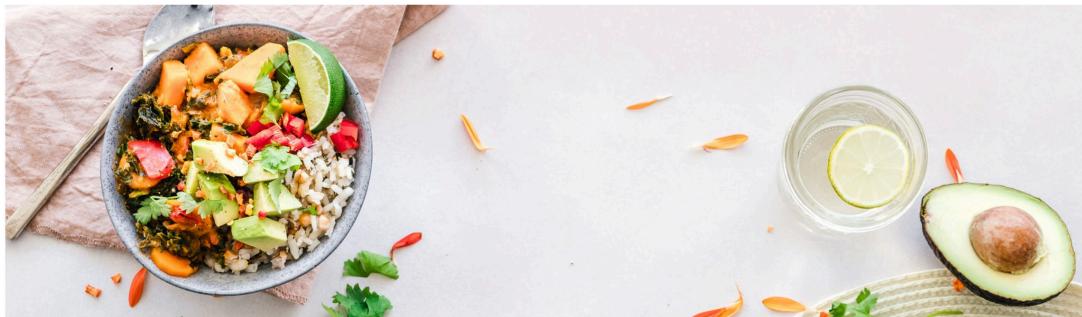
Exercises

Blog Page

Create a page for a recipe website similar to the image below. Read the next page for hints on typography.

Sweet Potato and Kale Bowl

Lorem ipsum dolor sit amet consectetur, adipisicing elit. Dolorem corrupti earum voluptates iure magni soluta facere adipisci dicta, facilis vel.



Ingredients

For the sweet potato

- 2 tablespoons olive oil
- 1 head cauliflower
- 3 cloves garlic
- 1 cup milk

For the kale

- 1 tablespoon bacon fat
- 3 cups chopped kale
- 3 cloves garlic minced

Hints

- Open [/Solution/blog.html](#) to see the final result.

Fonts

- Use Google Font to find two sans serif and one serif font.
- The serif font (in italic) is used for the lead paragraph before the image.
- The two sans serif fonts are used for the body text and headings.
- Reset the base font size to **62.5%** so rem calculations can be easier.
- Use [type-scale.com](#) to determine the size of all headings (h1 to h6).

Container

- Add the entire content of the body in a container with the maximum width of **1140px**.
- To ensure that the container is in the middle, set the horizontal margin to **auto**. This way any extra space will be equally distributed to the left and right margins.

Page Elements

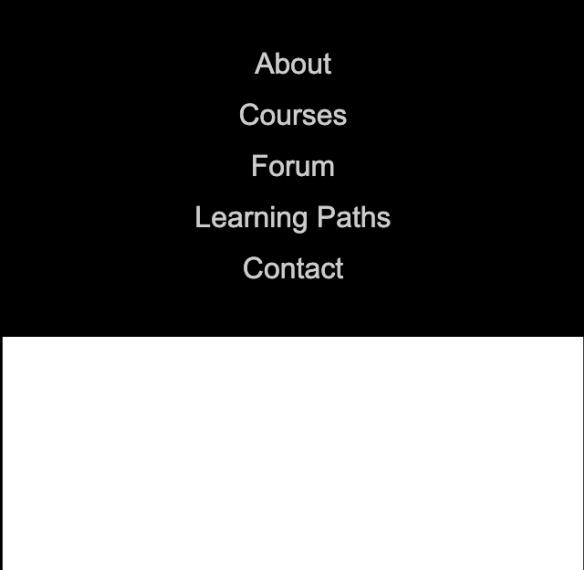
- Make sure the heading elements have a proper hierarchy. There must be one and only one h1 element on the page. Don't skip headings.
- The lead paragraph (above the image) should be slightly larger than the body text.
- You can get the image from here: <https://unsplash.com/photos/C1Q3qOTlegg>
- Give the image a width of 100% to make it responsive.
- Ensure everything looks good on smaller screens.
- Review my implementation. What styles did you miss? Investigate the impact of these styles on the page layout .

Exercises

Navigation Bar

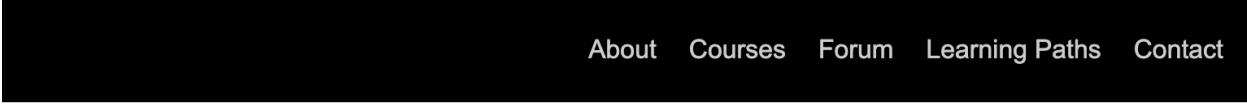
Create a navigation bar similar to the image below. Read the next page for the details.

Mobile

A black rectangular box representing a mobile navigation bar. It contains five white text links: "About", "Courses", "Forum", "Learning Paths", and "Contact".

- About
- Courses
- Forum
- Learning Paths
- Contact

Tablets (768px) and wider screens

A black horizontal bar representing a navigation menu. It contains five white text links: "About", "Courses", "Forum", "Learning Paths", and "Contact".

- About
- Courses
- Forum
- Learning Paths
- Contact

Hints

1. Open [/Solution/NavBar/index.html](#) to see the final result.
2. Launch **Chrome DevTools**. On the **Elements** tab, click on the **Toggle Device Mode** icon to go to the device mode.



3. Click on various parts of the the device bar on the top to view the layout on different screen sizes.



4. As you can see, on mobiles, the items are listed vertically and are center-aligned. On tablets and wider screens, they are listed horizontally and pushed to the end of the navigation bar.
5. Build this navigation bar from scratch using the mobile-first approach. Make the browser small or use the device mode in DevTools to ensure the navigation bar has the right look and feel on mobile devices.
6. To change the default font, apply the following style to the **body** element. We'll talk about fonts in detail later in the course.

```
font-family: Arial, Helvetica, sans-serif;
```

7. Use media queries to re-style the navigation bar for tablets and wider screen. You can use **768px** as the breakpoint. That's a standard breakpoint for tablets.
8. Once you're done, compare your HTML/CSS with mine. It's almost impossible that your solution would be identical to mine, and that's totally expected. We all think differently. As long as your solution works, you're on the right track. Study my HTML/CSS code to see if there are areas you can improve your implementation.

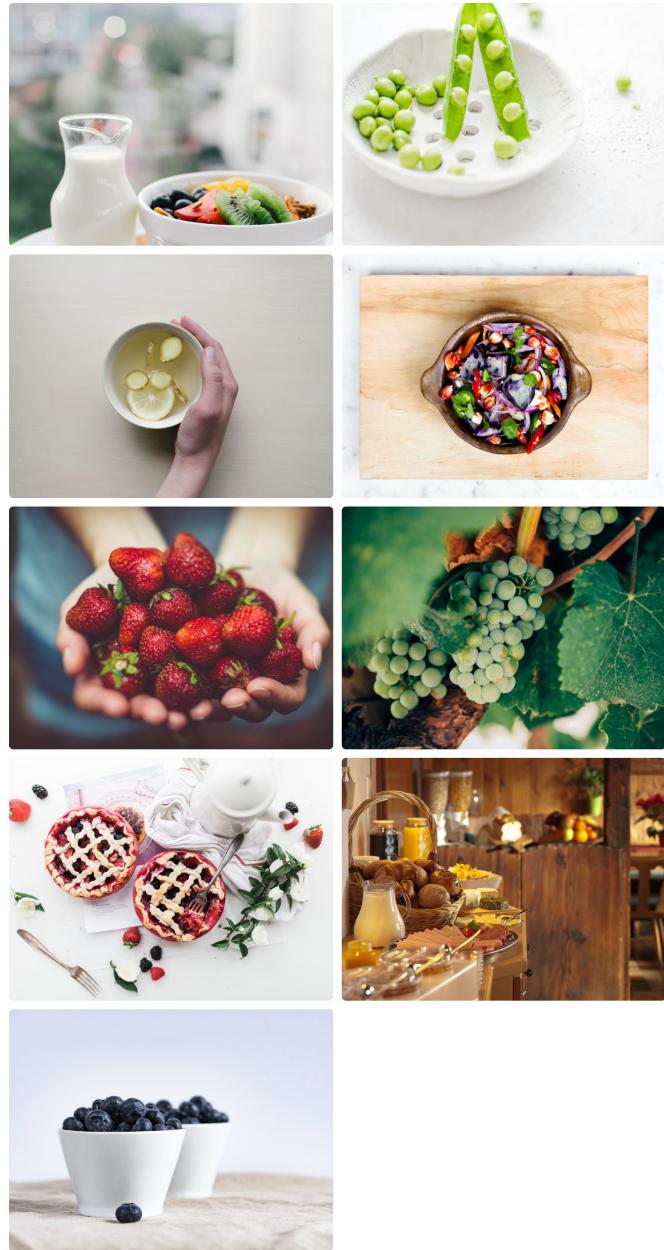
Photo Gallery

Create a photo gallery similar to the image below. Read the next page for the details.

Mobiles



Tablets (768px)



Laptops (1024px) and wider screens

Hints

1. Open </Solution/PhotoGallery/index.html> to see the final result.
2. Use the **Device Mode** in **DevTools** to view the layout on various screen sizes.
3. Note that on mobile, we have a single column. On tablets, we have two columns. On laptops and wider screens, we have three columns and the third image takes up the space of four images.



4. Build this gallery from scratch using the mobile-first approach.
5. You can set your image source to <https://source.unsplash.com/collection/190727/800x600> to get a random image.
6. Sometimes you may get duplicate images as a result of caching. To prevent this, append a **question mark and a sequence number** to the URL of the image source: <https://source.unsplash.com/collection/190727/800x600?1>
7. You can represent the container for the images using a **div** or a **ul** element. It's up to you.
8. To make the images fit the containing cell, you should give them a **width** of **100%** and set their **object-fit** property to **cover** so they don't get squashed.
9. Once done, compare your implementation with mine.

Exercises

Hubble Banner

Create a banner (also called a hero) like the image below. Read the next page for hints.



Hints

- Open `/Solutions/Hubble/index.html` to see the final result.

Layout

- The banner should take up the entire height of the viewport.
- The heading and the tagline should be horizontally and vertically in the center of the banner.

Image

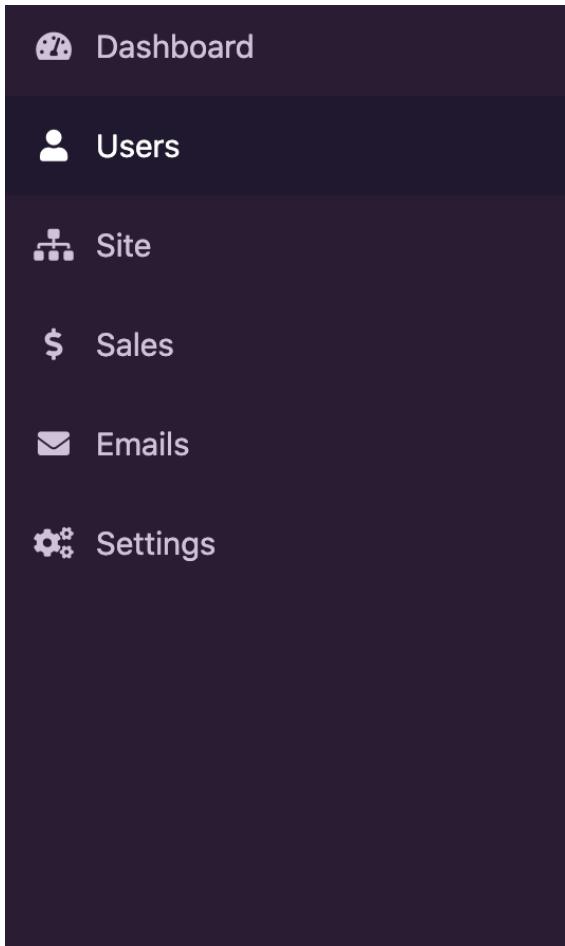
- Get the image (medium size) from here: <https://unsplash.com/photos/E0AHdsENmDg>
- The medium-sized image has the resolution of **1920x1311** and the file size of **700KB**. Serving this image to smaller screens like mobiles and tablets is far from optimal. All the extra bytes downloaded by the browser will be wasted and the browser has to load this large image for resizing. Use <https://www.responsivebreakpoints.com/> to create 4 versions of this image for screens with the resolution of 320 - 1440.
- The width of generated images will be **320px** (for small mobiles), **853px** (for tablets), **1228px** (for tablets) and **1440px** (for desktops). Use media queries to set the right background image depending on the viewport width.
- Remember: the more variations of the image we supply, the better we can optimize for various screen sizes but 4 is enough for this exercise.
- Use the **Device Mode** in **DevTools** to ensure the page looks good on various screen sizes.
- Open the **Network tab** and ensure that the right image is downloaded for the right screen.

Text

- Use the **Poppins** font (available on Google Fonts) with the weights of **900** (for the heading) and **300** (for the tagline).
- To make the heading better stand out on the background, give it a small black shadow.
- To make the heading more readable, increase the space between the words.
- Note that the tagline includes a semi-transparent black background to improve readability. To make the background semi-transparent, use an `rgba` color.

Menu with Icons

Create a menu with icons similar to the image below. Read the next page for hints.



- Open `/Solutions/Menu/index.html` to see the final result.
- Use the **system font stack**.
- The background for the menu is `#2a2138`.
- The background for the selected item and the hover effect is `#221a2e`.
- On mobiles, the menu should have a width of 100% and be tall enough to fit its content.
- On tablets or wider screens, the menu should have a width of 20% and take up 100% of the height of the viewport.
- Use Font Awesome to display the icons.
 - Dashboard: `fa-tachometer-alt`
 - Users: `fa-user`
 - Site: `fa-sitemap`
 - Sales: `fa-dollar-sign`
 - Emails: `fa-envelope`
 - Settings: `fa-cogs`

Exercises

Sign in form

Build a sign in form similar to the image below. Read the next page for hints.

The image shows a sign-in form with the following elements:

- Title:** Please sign in
- Email address:** An input field with a light blue border.
- Password:** An input field with a light gray border.
- Remember me:** A small checkbox labeled "Remember me".
- Sign in:** A large blue button with white text.
- Copyright:** Copyright © 2020 at the bottom of the form.

Hints

- Open `/Solutions/Login/index.html` to see the final result.

Layout

- The form should be horizontally and vertically in the center of the screen.
- The background of the page is `#f5f5f5`.
- Use the **system font stack**.

Inputs

- Note that the input fields and the button have the same width. One way to achieve this is by applying the same width (eg 200px) to both these elements. But what if we need shorter or longer input fields on a different page? A more flexible approach is to give these elements a width of 100% so they always stretch and fill their container. The actual width can then be applied on the container.
- The first input field should be automatically in focus when the page loads.
- Border color: `#ced4da`
- Focus state:
 - Border color: `#86b7fe`
 - Box shadow: `#b9d3fa`

Buttons

- Background: `#0d6efd`