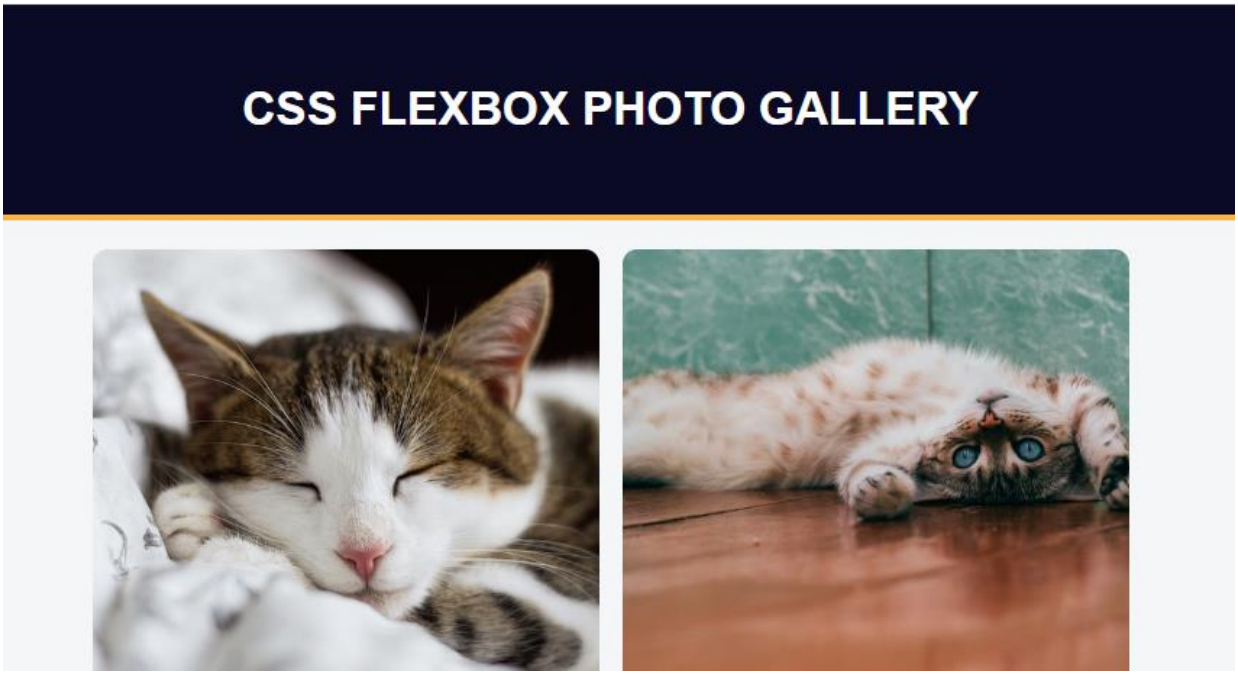
Building a Photo Gallery

In this practice project, you'll learn HTML and CSS flex box by building a photo gallery. The preview of the Photo Gallery will look as shown below:



1. Begin with your standard HTML boilerplate. Add a DOCTYPE declaration, an html element, a head element, and a body element.

2. Within your head element, add a meta tag with the name set to viewport and the content set to width=device-width, initial-scale=1.

Also add a meta tag with the charset set to UTF-8.

3. Within your head element, add a title element with the text set to Photo Gallery, and a link element to add your styles.css file to the page.

4. Within the body element, create a div with the class set to header.

Inside the .header element nest an h1 element with the text CSS FLEXBOX PHOTO GALLERY.

5. Below your .header element, create a new div element with the id set to gallery.

In that #gallery element, create ten img elements.

6. Now you need to give each img element a src attribute. You are going to use the format **./images/#.jpg** replacing # with the img element order.

So your first img element would use 1.jpg, your second would use 2.jpg, and so on.

7. Normalize your box model by creating a * selector and setting the box-sizing property to border-box.

8. Your images are currently too large, and you will not be able to see your CSS changes.

Create a #gallery img selector to target your images. Give it a width property set to 25%.

Also set the height property to 300px to keep your images a uniform size.

9. Remove the margin from your body element, set the font-family to sans-serif, and give it a background-color of #f5f6f7 as the value.

10. Align your .header text in the center, give it a padding of 32px on all sides, and set the background to #E0DDDD.

11. Flexbox is a one-dimensional CSS layout approach that focuses on the flow of content. It offers the ability to control the way items are spaced and aligned within a container.

To set an element to use Flexbox, you give it a display property set to flex. Create a #gallery selector and give it that property.

12. Flexbox can be thought of as having two axes, the main axis and the cross axis. The main axis is determined by the flex-direction property. If flex-direction is set to row or row-reverse, the main axis is horizontal. If flex-direction is set to column or column-reverse, the main axis is vertical.

Give your #gallery selector a flex-direction property set to row.

13. You may have noticed that your images have all moved onto the same row.

The flex-wrap property determines how your items should behave when the flex container is too small. Setting this property to wrap will allow your items to wrap to the next row/column (depending on your main axis), where nowrap will

prevent your items from wrapping. When this is set to nowrap, items may either shrink to fit or overflow.

Give the #gallery selector a flex-wrap property set to wrap. You should see your images take a four-column layout. This is because you set their width to 25% in an earlier step.

14. The justify-content property determines how the items inside a flex container are positioned along the main axis, affecting their position and the space around them.

Give your #gallery selector a justify-content property set to center.

15. The align-items property positions the flex content along the cross axis. In this case, with your flex-direction set to row, your cross axis would be vertical.

To vertically center your images, give your #gallery selector an align-items property set to center.

16. Give your #gallery selector a padding property set to 0 4px.

17. Notice how some of your images have become distorted. This is because the images have different aspect ratios. Rather than setting each aspect ratio individually, you can use the object-fit property to determine how images should behave.

Give your #gallery img selector the object-fit property and set it to cover. This will tell the image to fill the img container while maintaining aspect ratio, resulting in cropping to fit.

18. Your images need some space between them.

Give your #gallery img selector a margin-top property of 8px and a padding property of 0 4px.

19. Smooth out your images a bit by giving the #gallery img selector a border-radius property of 10px.

20. Create a media query for screens smaller than 800px in width. In that media query, create a #gallery img rule and set the width property to 50%. This will convert your gallery to a two-column layout.

```
@media (max-width: 800px) {
```

```
  #gallery img {
    width: 50%;
  }
}
```

21. Resize the preview to see the layout change from your media query.

Finally, create another media query for screens smaller than 600px wide. In that media query, create a #gallery img rule and set the width property to 100%. This will give your gallery a single-column layout.

Your CSS Flexbox Photo Gallery is now complete.