Media Query

```html
<!DOCTYPE html>
<html>
  <head>
    <title>Media Query for Screen</title>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <link rel="stylesheet" type="text/css" href="./assets/css/style.css" />
  </head>
  <body>
    <!-- Header -->
    <header>
      <h1>This Header</h1>
      <p>will change colors when the screen size hits 992px or smaller.</p>
    </header>
    <!-- Navbar -->
    <nav>
      <p>
        This Navbar will change colors when the screen size hits 768px or
        smaller.
      </p>
    </nav>
    <!-- Body -->
    <main>
      <aside>
        <h2>This is a sidebar.</h2>
        <div class="sample-img">Image</div>
        <p>When the screen size hits 576px or smaller...</p>
      </aside>
      <section>
        <h2>This is the body.</h2>
        <div class="sample-img">Image</div>
        <p>..all of the image placeholders will change colors!</p>
        <br />
        <h2>But the real problem is...</h2>
        <div class="sample-img">Image</div>
        <p>
          What are we going to do about the sidebar and body columns becoming
          squished on a smaller screen?
        </p>
      </section>
    </main>
    <!-- Footer -->
    <footer>
      <h3>
        Open Chrome DevTools --> Toggle device toolbar --
> Resize the browser
        window!
      </h3>
    </footer>
  </body>
</html>
```
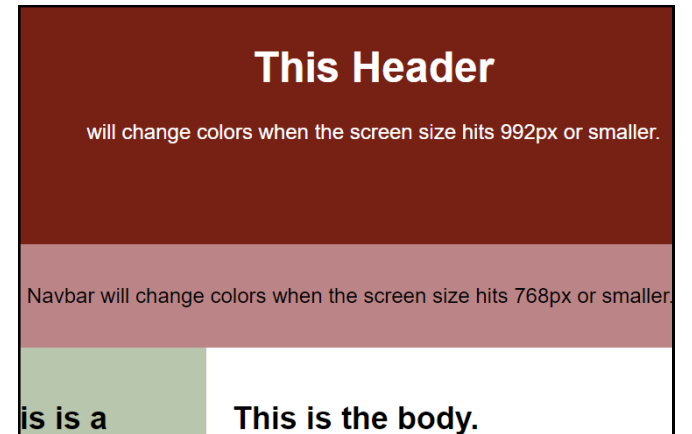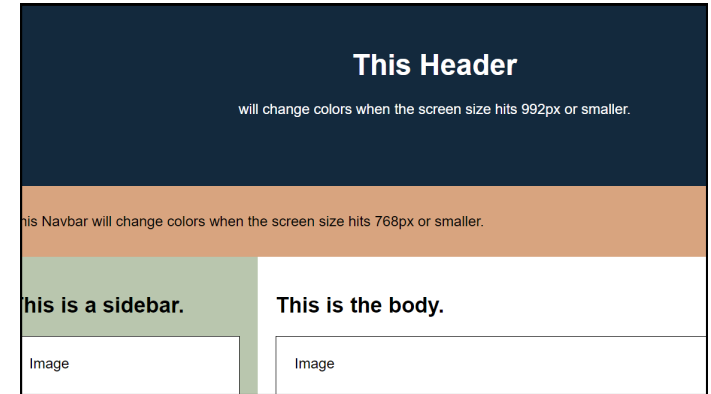
```css
* {
  box-sizing: border-box;
}
body {
  font-family: Arial;
  margin: 0;
}
header {
  padding: 60px;
  text-align: center;
  background: #13293d;
  color: #fff;
}
nav {
  display: flex;
  background-color: #d8a47f;
}
nav p {
  padding: 14px 20px;
  text-decoration: none;
}
main {
  display: flex;
}
aside {
  flex: 30%;
  background-color: #b9c6ae;
  padding: 20px;
  flex-wrap: wrap;
}
section {
  flex: 70%;
  background-color: #fff;
  padding: 20px;
}
footer {
  padding: 20px;
  text-align: center;
  background: #13293d;
  color: #fff;
}
/* Image Placeholders */
.sample-img {
  background-color: #fff;
  width: 100%;
  padding: 20px;
  height: 200px;
  border: 1px solid black;
}
/* When using `max-
width` as the breakpoint, you must go from widest to narrowest to avoid conflict */
/* Tablets - header changes color */
@media screen and (max-width: 992px) {
  header {
    background: #772014;
  }
}
/* Large smartphones - navbar changes color */
@media screen and (max-width: 768px) {
  nav {
    background-color: #bb8588;
  }
}
/* Small smartphones - image placeholders change color */
@media screen and (max-width: 576px) {
  .sample-img {
    background-color: #8ac4ff;
  }
}
```


This Header
will change colors when the screen size hits 992px or smaller.

This Navbar will change colors when the screen size hits 768px or smaller.

This is a sidebar.   This is the body.

Image   Image

This Header

will change colors when the screen size hits 992px or smaller.

Navbar will change colors when the screen size hits 768px or smaller.

is is a   This is the body.

FlexBox

```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Flexbox</title>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <link rel="stylesheet" type="text/css" href="./assets/css/style.css" />
  </head>
  <body>
    <header class="main-header">
      <h1>Welcome to the store!</h1>
      <h2>We sell stuff!</h2>
    </header>
    <main>
      <section class="card camera">
        <header>Camera</header>
      </section>
      <section class="card">
        <header>Tablet</header>
      </section>
      <section class="card">
        <header>Toilet Paper</header>
      </section>
      <section class="card">
        <header>Wooden Spoons</header>
      </section>
    </main>
  </body>
</html>
```
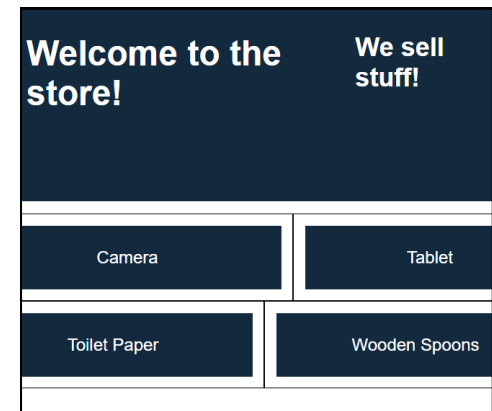
```css
* {
  box-sizing: border-box;
}
body {
  font-family: Arial;
  margin: 0;
}
main {
  /* Turns this element into a flexbox */
  display: flex;
  /* Forces elements to the next line if they don't fit */
  flex-wrap: wrap;
  margin-top: 2%;
}
.main-header {
  display: flex;
  /* Defines how to distribute the empty space between child elements */
  justify-content: space-between;
  padding: 60px;
  background: #13293d;
  color: #fff;
}
.card {
  /* A flex-basis of 200px sets the starting width */
  /* A flex-grow value of 1 allows the elements to
          evenly grow to fill the remaining space */
  flex: 1 0 200px;
  padding: 10px;
  text-align: center;
  border-style: solid;
  border-width: 1px;
}
.card header {
  padding: 20px;
  background: #13293d;
  color: #fff;
}
.card.camera {
  /* With a flex-grow value of 2, this element grows twice as wide as others */
  /* The second value, flex-shrink, defines how the element will shrink
          if it can't fit */
  flex: 2 1 200px;
}
```

```html
<!DOCTYPE html>
<html>
  <head>
    <title>Responsive Web Design</title>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <link rel="stylesheet" type="text/css" href="./assets/css/style.css" />
  </head>
  <body>
    <header>
      <h1>Responsive Web Design</h1>
      <p>Watch what happens to the navbar links when the screen hits 780px!</p>
    </header>
    <nav>
      <a href="#">Link 1</a>
      <a href="#">Link 2</a>
      <a href="#">Link 3</a>
      <a href="#">Link 4</a>
    </nav>
    <main>
      <aside>
        <h2>This is a sidebar.</h2>
        <div class="sample-img">Image</div>
        <p>
          It will stack BELOW the body once the screen hits 768px or smaller.
        </p>
      </aside>
      <section>
        <h2>This is the body.</h2>
        <div class="sample-img">Image</div>
        <p>
          It will stack on top of the sidebar once the screen hits 768px or
          smaller.
        </p>
      </section>
    </main>
    <footer>
      <h3>
        Open Chrome DevTools --> Toggle device toolbar --> Resize the browser
        window!
      </h3>
    </footer>
  </body>
</html>
```
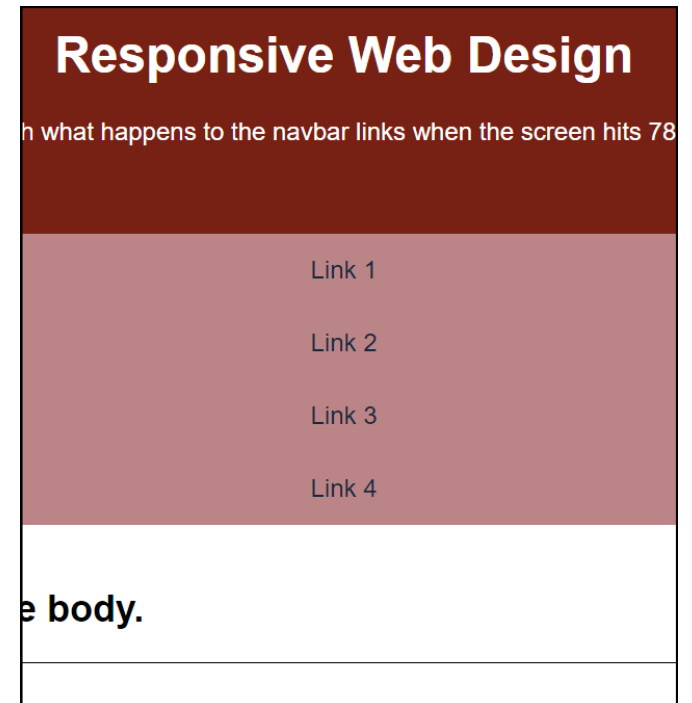
```css
* {
  box-sizing: border-box;
}
body {
  font-family: Arial;
  margin: 0;
}
header {
  padding: 40px;
  text-align: center;
  background: #13293d;
  color: #fff;
}
nav {
  display: flex;
  background-color: #d8a47f;
}
nav a {
  color: #13293d;
  padding: 14px 20px;
  text-decoration: none;
  text-align: center;
}
main {
  display: flex;
  flex-wrap: wrap;
}
aside {
  flex: 1;
  background-color: #b9c6ae;
  padding: 20px;
}
section {
  flex: 3;
  background-color: #fff;
  padding: 20px;
}
.sample-img {
  background-color: #fff;
  width: 100%;
  padding: 20px;
  height: 200px;
  border: 1px solid black;
}
footer {
  padding: 20px;
  text-align: center;
  background: #13293d;
  color: #fff;
}
@media screen and (max-width: 768px) {
  /* Use flex-direction to stack the navbar, sidebar, and body */
  main,
  nav {
    flex-direction: column;
  }
  /* Make the sidebar stack below the body */
  aside {
    order: 2;
  }
  /* Change the header background */
  header {
    background: #772014;
  }
  /* Change the navbar and footer background */
  nav,
  footer {
    background-color: #bb8588;
  }
}
```

Responsive Design

Flexbox Placement

```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Flexbox Alignment</title>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-
width, initial-scale=1" />
    <link rel="stylesheet" type="text/css" href="./assets/css/
style.css" />
  </head>
  <body>
    <div class="products">
      <section class="card">
        <header>Centered Camera</header>
        <img src="./assets/images/camera.jpg" alt="camera" />
      </section>
      <section class="card">
        <header>Centered Tablet</header>
        <img src="./assets/images/tablet.jpg" alt="tablet" />
      </section>
    </div>
  </body>
</html>
```
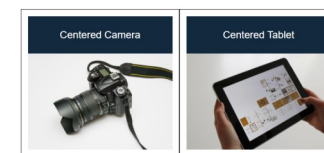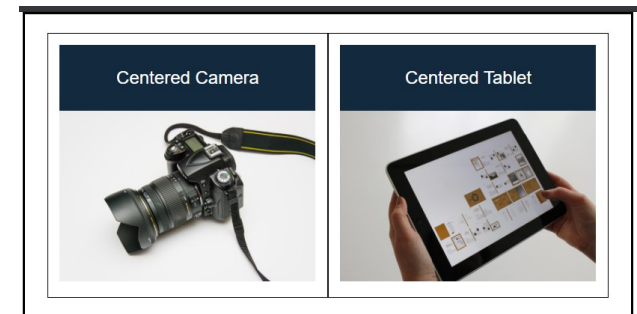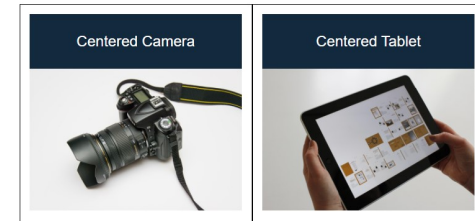
```css
* {
  box-sizing: border-box;
}
body {
  font-family: Arial;
  margin: 0;
}
.products {
  display: flex;
  flex-direction: row;
  /* Centers along the main axis determined by flex-
direction */
  justify-content: center;
  /* Centers along the opposite axis */
  align-items: center;
  width: 90%;
  /* Uses the height of the viewport */
  height: 100vh;
  margin: 0 auto;
  border-style: solid;
  border-width: 2px;
}
.card {
  padding: 10px;
  width: 250px;
  text-align: center;
  border-style: solid;
  border-width: 1px;
}
.card header {
  padding: 20px;
  background: #13293d;
  color: #fff;
}
.card img {
  width: 100%;
}
```

Box Styling

```html
<!DOCTYPE html>
<html>
  <head>
    <title>CSS Box Styling</title>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <link rel="stylesheet" type="text/css" href="./assets/css/style.css" />
  </head>
  <body>
    <header>
      <h1>CSS Box Styling</h1>
      <p>Here are different ways we can transform boxes!</p>
    </header>
    <nav><br /></nav>
    <main>
      <section>
        <div class="photo rotate">
          <h2>Rotate</h2>
          <img
            src="assets/images/teddy-bear.jpg"
            alt="teddy bear rotated to the right"
          />
          <p>
            We can rotate to the right or to the left!
          </p>
        </div>
      </section>
      <section>
        <div class="photo scale">
          <h2>Scale</h2>
          <img
            src="assets/images/canned-coffee.jpg"
            alt="canned coffee stretched horizontally"
          />
          <p>
            We can stretch it vertically or horizontally!
          </p>
        </div>
      </section>
      <section>
        <div class="photo skew">
          <h2>Skew</h2>
          <img
            src="assets/images/cookie-tin.jpg"
            alt="cookie tin skewed to the left"
          />
          <p>
            We can skew to the right or to the left!
          </p>
        </div>
      </section>
    </main>
    <footer>
      <h3>
        Open Chrome DevTools --> Toggle device toolbar --> Resize the browser
        window!
      </h3>
    </footer>
  </body>
</html>
```
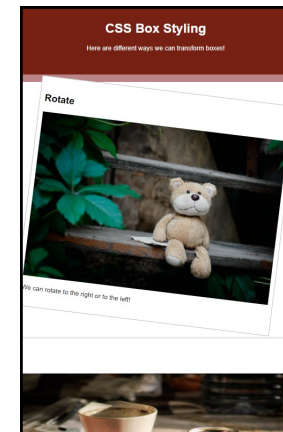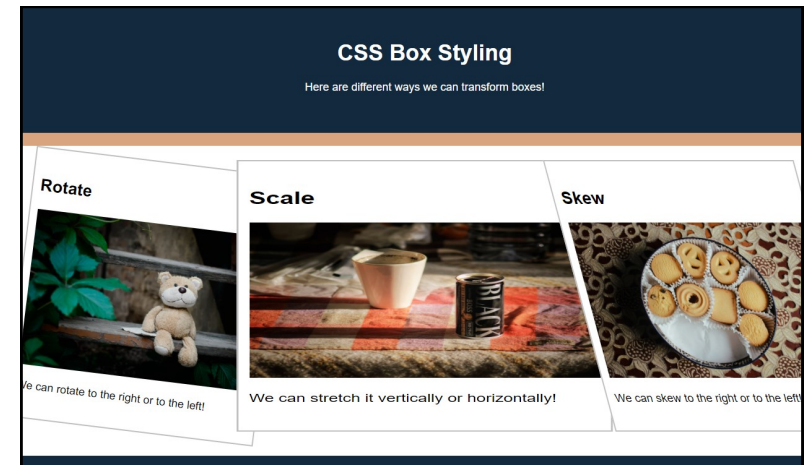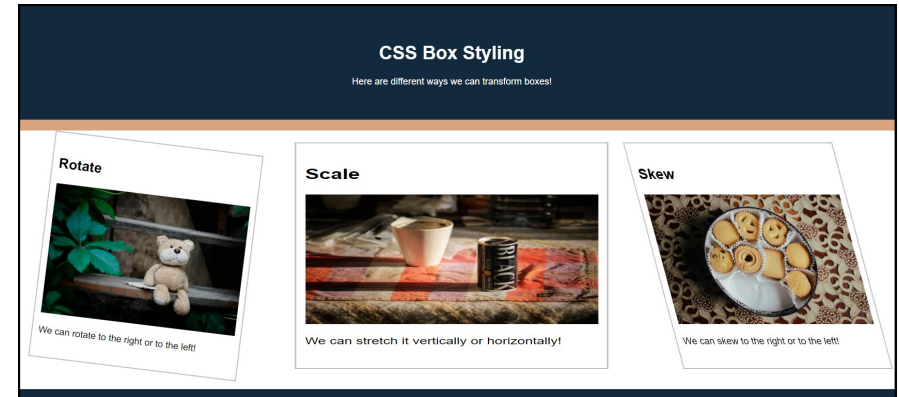
```css
* {
  box-sizing: border-box;
}
body {
  font-family: Arial;
  margin: 0;
  display: flex;
  flex-direction: column;
  min-height: 100vh;
  justify-content: flex-start;
}
header {
  padding: 40px;
  text-align: center;
  background: #13293d;
  color: #fff;
}
nav {
  display: flex;
  background-color: #d8a47f;
}
main {
  display: flex;
  flex-wrap: wrap;
  justify-content: space-around;
}
section {
  flex: 0 0 400px;
  background-color: #fff;
  padding: 20px;
}
img {
  width: 100%;
}
footer {
  flex-shrink: 0;
  padding: 20px;
  text-align: center;
  background: #13293d;
  color: #fff;
  margin-top: auto;
}
/* Added some padding and styling for the photos */
.photo {
  padding: 20px 10px;
  border: 2px solid #bfbfbf;
  background-color: #fff;
}
/* We can use rotate to rotate the photo to the left or right */
.rotate {
  transform: rotate(7deg);
}
/* We can use scale to stretch the photo vertically or horizontally */
.scale {
  transform: scale(1.5, 1);
}
/* We can use skew to skew the photo to the left or right */
.skew {
  transform: skew(15deg);
}
@media screen and (max-width: 768px) {
  main,
  nav {
    flex-direction: column;
  }
  header {
    background: #772014;
  }
  nav,
  footer {
    background-color: #bb8588;
  }
}
```

```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>CSS Resets</title>
  <!-- <link rel="stylesheet" href="./assets/css/reset.css" /> -->
</head>
<body>
  <header>
    <h1>The CSS Blog</h1>
    <nav>
     <ul>
      <li><a href="#">Home</a></li>
      <li><a href="#">About</a></li>
      <li><a href="#">Contact</a></li>
      <li><a href="#">Resources</a></li>
     </ul>
    </nav>
  </header>
  <main>
    <section>
      <article>
       <h2>Building Responsive Layouts</h2>
       <div>
        <p>Lorem ipsum dolor sit amet consectetur adipisicing elit.
             Nihil natus fuga blanditiis accusamus tempore!</p>
        <ul>
         <li>Lorem ipsum dolor sit amet consectetur adipisicing elit</li>
         <li>Nihil natus recusandae amet nostrum odit</li>
         <li>Eligendi vitae fuga blanditiis accusamus tempore</li>
        </ul>
        <p>Lorem ipsum dolor sit amet consectetur adipisicing elit.
             Ni-
hil natus recusandae amet nostrum odit, ut unde non sint, quia sed ullam!</p>
       </div>
      </article>
      <article>
       <h2>Border Radius Tricks</h2>
       <div>
        <p>Lorem ipsum dolor sit amet consectetur adipisicing elit.
             Nihil natus recusandae amet nostrum Omnis quam?</p>
        <p>Lorem ipsum dolor sit amet consectetur adipisicing elit.
             Nihil natus recusandae amet nostrum blanditiis accusamus tempore!
</p>
       </div>
      </article>
    </section>
    <aside>
     <h2>Sign up to Learn More!</h2>
     <form>
      <input type="text" placeholder="E-mail address" />
      <button>Submit</button>
     </form>
    </aside>
  </main>
  <footer>
   <span>Thanks for visiting!</span>
  </footer>
</body>
</html>
```

```css
/* Resets the box size of every element */
* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}
/* Gives the body element a parent height to compare against */
html {
  height: 100%;
}
body {
  min-height: 100%;
  line-height: 1;
  font-family: sans-serif;
}
h1, h2, h3, h4, h5, h6 {
  font-size: 100%;
}

/* Matches the font of special elements to the rest of the page */
input, select, option, optgroup, textarea, button,
pre, code {
  font-size: 100%;
  font-family: inherit;
}

/* Removes default bullet points from lists */
ol, ul {
  list-style: none;
}
```

Before:

The CSS Blog

- Home
- About
- Contact
- Resources

Building Responsive Layouts

Lorem ipsum dolor sit amet consectetur adipisicing elit. Nihil na
unde non sint, quia sed inventore odio, delectus iure! Eligendi vi

- Lorem ipsum dolor sit amet consectetur adipisicing elit
- Nihil natus recusandae amet nostrum odit
- Eligendi vitae fuga blanditiis accusamus tempore

Lorem ipsum dolor sit amet consectetur adipisicing elit. Nihil na
unde non sint, quia sed inventore odio, delectus iure! Eligendi vi
Lorem ipsum dolor sit amet consectetur adipisicing elit. Consect
esse voluptatum, dolores dolor aspernatur neque laborum doloru
eveniet sequi corporis tempore ullam!

Border Radius Tricks

Lorem ipsum dolor sit amet consectetur adipisicing elit. Nihil na
unde non sint, quia sed inventore odio, delectus iure! Eligendi vi

After:

The CSS Blog
Home
About
Contact
Resources
Building Responsive Layouts
Lorem ipsum dolor sit amet consectetur adipisicing elit. Nihil natus recusandae amet nostrum odit, ut
unde non sint, quia sed inventore odio, delectus iure! Eligendi vitae fuga blanditiis accusamus tempore!
Lorem ipsum dolor sit amet consectetur adipisicing elit
Nihil natus recusandae amet nostrum odit
Eligendi vitae fuga blanditiis accusamus tempore
Lorem ipsum dolor sit amet consectetur adipisicing elit. Nihil natus recusandae amet nostrum odit, ut
unde non sint, quia sed inventore odio, delectus iure! Eligendi vitae fuga blanditiis accusamus tempore!
Lorem ipsum dolor sit amet consectetur adipisicing elit. Consectetur, nisi, saepe quas sint consequuntur
esse voluptatum, dolores dolor aspernatur neque laborum dolorum dolorem repudiandae. Dignissimos
eveniet sequi corporis tempore ullam!
Border Radius Tricks
Lorem ipsum dolor sit amet consectetur adipisicing elit. Nihil natus recusandae amet nostrum odit, ut
unde non sint, quia sed inventore odio, delectus iure! Eligendi vitae fuga blanditiis accusamus tempore!
Lorem ipsum dolor sit amet consectetur adipisicing elit. Omnis cumque voluptatum fugiat autem nesciunt
harum amet eaque, officiis quidem dignissimos fuga excepturi voluptas impedit ipsam voluptate sequi
labore architecto quam?
Lorem ipsum dolor sit amet consectetur adipisicing elit. Nihil natus recusandae amet nostrum odit, ut
unde non sint, quia sed inventore odio, delectus iure! Eligendi vitae fuga blanditiis accusamus tempore!
Sign up to Learn More!
E-mail address | Submit
Thanks for visiting!

```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>CSS Typography</title>
    <link rel="stylesheet" href="./assets/css/reset.css" />
    <link rel="stylesheet" href="./assets/css/style.css" />
  </head>
  <body>
    <header>
      <h1>CSS Typography</h1>
      <nav>
        <ul>
          <li><a href="#">Home</a></li>
          <li><a href="#">About</a></li>
          <li><a href="#">Contact</a></li>
          <li><a href="#">Resources</a></li>
        </ul>
      </nav>
    </header>
    <main>
      <section>
        <article>
          <h2>Typography</h2>
          <div>
            <p>
              We can use different CSS properties to make the text on our
              website more readable. The basic fundamentals of text styling
              includes setting the font weight, font family, and font style, as
              well as text alignment, line and letter spacing.
            </p>
            <p>For this activity, we will demo the following:</p>
            <ul>
              <li>Font weight, family, style, size</li>
              <li>Text transform</li>
              <li>Text decoration</li>
              <li>Margins</li>
            </ul>
          </div>
        </article>
        <article>
          <h2>Style this text!</h2>
          <div>
            <p class="one">
              This text currently has a 'font-weight' of 900, 'font-family' of
              sans-serif, and 'font-style' of oblique.
            </p>
            <p class="two">
              This text currently has a 'font-weight' of 500, 'font-family' of
              serif, and 'font-style' of italic.
            </p>
            <p class="three">
              This text currently has a 'font-weight' of 100, 'font-family' of
              monospace, and 'font-style' of normal.
            </p>
            <p class="four">
              Lorem ipsum dolor sit amet consectetur adipisicing elit. Possimus,
              praesentium at. Inventore, consequatur!
            </p>
          </div>
        </article>
      </section>
      <aside>
        <h2>Sign up to Learn More!</h2>
        <form>
          <input type="text" placeholder="E-mail address" />
          <button>Submit</button>
        </form>
      </aside>
    </main>
    <footer>
      <span>Thanks for visiting!</span>
    </footer>
  </body>
</html>
```

```css
body {
  display: flex;
  flex-direction: column;
  line-height: 1.3;
}
header {
  display: flex;
  justify-content: space-between;
  align-items: center;
  padding: 1%;
  background-color: #13293d;
  color: #fff;
}
/* Change font size of headings */
h1 {
  font-size: 200%;
}
/* Change the margin */
h2 {
  margin: 2% 0;
  font-size: 150%;
}
nav ul {
  display: flex;
  justify-content: space-between;
  min-width: 300px;
}
nav a {
  color: #b9c6ae;
  font-weight: bold;
  text-decoration: none;
}
main {
  display: flex;
  flex: 1 1 0;
  max-width: 1200px;
  margin: 2% auto;
}
section {
  flex: 1 1 0;
  padding-right: 5%;
}
article {
  border-bottom: 1px solid #d8a47f;
}
article p {
  margin: 2% 0;
  font-size: 110%;
}
/* Change the margin specifically */
article ul {
  margin-left: 5%;
  margin-bottom: 2%;
  font-size: 110%;
  list-style: circle;
}
/* Implement font stacks */
.one {
  font-family: 'Helvetica', 'Arial', sans-serif;
  font-weight: 900;
  font-style: oblique;
}
.two {
  font-family: 'Times New Roman', Times, serif;
  font-weight: 500;
  font-style: italic;
}
.three {
  font-family: 'Menlo', 'Lucida Console', monospace;
  font-weight: 100;
  font-style: normal;
}
/* Using DevTools, change the following properties
to demonstrate the differences */
.four {
  /* Change font family: serif, sans-serif, monospace, cursive */
  font-family: serif;
  /* Change font weight: bold, 100-900 */
  font-weight: 100;
  /* Change the font style: normal, italic, oblique */
  font-style: normal;
  /* Transform the text: uppercase, lowercase, capitalize */
  text-transform: none;
  /* Set text decorations on fonts: underline, overline, line-
through */
  text-decoration: none;
}
aside {
  width: 25%;
}
aside h2 {
  font-size: 150%;
}
input,
button {
  width: 100%;
  margin: 2% 0;
  padding: 1%;
}
footer {
  display: flex;
  justify-content: center;
  padding: 2%;
  background-color: #13293d;
  color: #fff;
}
@media screen and (max-width: 768px) {
  header,
  main {
    flex-direction: column;
  }
  section {
    padding-right: 0;
  }
  aside {
    width: 100%;
    text-align: center;
  }
}
```
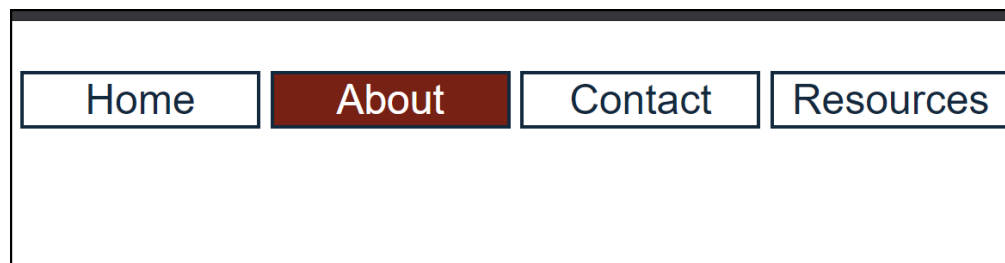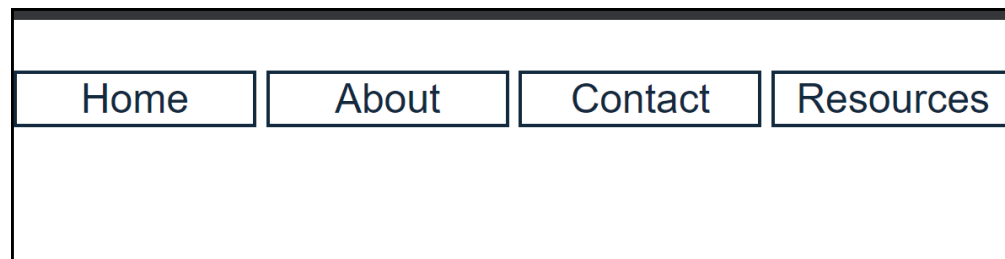
Typography

CSS Typography

Typography

We can use different CSS properties to make the text on our website more readable. The basic fundamentals of text styling includes setting the font weight, font family, and font style, as well as text alignment, line and letter spacing.

For this activity, we will demo the following:

- Font weight, family, style, size
- Text transform
- Text decoration
- Margins

Style this text!

This text currently has a 'font-weight' of 900, 'font-family' of sans-serif, and 'font-style' of oblique.

This text currently has a 'font-weight' of 500, 'font-family' of serif, and 'font-style' of italic.

`This text currently has a 'font-weight' of 100, 'font-family' of monospace, and 'font-style' of normal.`

Lorem ipsum dolor sit amet consectetur adipisicing elit. Possimus, excepturi temporibus rem adipisci optio rerum facere blanditiis, ipsam iure esse accusantium eligendi aut est deserunt. In, praesentium at. Inventore, consequatur!

Pseudo-Classes

```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>CSS Pseudo-Classes</title>
  <link rel="stylesheet" href="./assets/css/style.css" />
</head>
<body>
  <nav>
    <button>Home</button>
    <button>About</button>
    <button>Contact</button>
    <button>Resources</button>
  </nav>
</body>
</html>
```

```css
nav {
  display: flex;
  justify-content: center;
  margin-top: 5%;
}
button {
  width: 250px;
  margin: 0 .5%;
  font-size: 200%;
  color: #13293d;
  background-color: #fff;
  border: 3px solid #13293d;
}
/* The user's cursor is positioned over the element */
button:hover {
  background-color: #772014;
  color: #fff;
}
/* The user is actively pressing down on the element */
button:active {
  font-size: 180%;
  box-shadow: 0 0 10px #000 inset;
}
/* The element is currently in focus, either by tabbing to or clicking on it */
button:focus {
  outline: none;
  border-color: #8ac4ff;
}
```

Pseudo-Elements

```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>CSS Pseudo Elements</title>
    <link rel="stylesheet" href="./assets/css/reset.css" />
    <link rel="stylesheet" href="./assets/css/style.css" />
  </head>
  <body>
    <header>
      <h1>CSS</h1>
      <nav>
        <ul>
          <li><a href="#">Home</a></li>
          <li><a href="#">About</a></li>
          <li><a href="#">Contact</a></li>
          <li><a href="#">Resources</a></li>
        </ul>
      </nav>
    </header>
    <main>
      <section>
        <article>
        <h2>Pseudo Elements</h2>
        <div>
          <p>
            We can do some cool stuff with the pseudo elements ::before and
            ::after. For every element on the page, we basically get two more!
            Let's try a few here.
          </p>
          <h3>Header</h3>
          <p>
            Notice the header above? If you look at our 'index.html' file, the
            header text is simply "CSS". But using the pseudo element ::after
            on the header, we added the words "Pseudo Elements" and gave it a
            different font and color.
          </p>
          <h3>H3 headings</h3>
          <p>
            Notice the h3 headings? Using the pseudo element ::before, we added
            an asterisk before each of the h3 headings.
          </p>
          <h3>Nav Items</h3>
          <p>
            Notice the navbar items? Using both ::before and ::after pseudo
            elements on the navbar items, we added red symbols around each
            one.
          </p>
        </div>
        </article>
      </section>
      <aside>
        <h2>Sign up to Learn More!</h2>
        <form>
          <input type="text" placeholder="E-mail address" />
          <button>Submit</button>
        </form>
      </aside>
    </main>
    <footer>
      <span>Thanks for visiting!</span>
    </footer>
  </body>
</html>
```



```css
body {
  display: flex;
  flex-direction: column;
  line-height: 1.3;
}

header {
  display: flex;
  justify-content: space-between;
  align-items: center;
  padding: 1%;
  background-color: #13293d;
  color: #fff;
}
h1 {
  font-size: 200%;
}
/* Added the rest of the header text */
h1::after {
  font-family: Impact;
  content: 'Pseudo Elements';
  color: #8ac4ff;
  margin-left: 5px;
}
h2 {
  margin: 2% 0;
  font-size: 180%;
}
h3 {
  margin: 2% 0;
  font-size: 150%;
}
/* Added asterisk before every h3 heading */
h3::before {
  font-family: Impact;
  content: '\273A';
  color: #772014;
  margin-right: 5px;
}
```

```css
/* Added symbols before and after the nav links */
nav li::before,
nav li::after {
  content: '\2766';
  color: #8ac4ff;
  padding: 1%;
  font-family: cursive;
}
nav ul {
  display: flex;
  justify-content: space-between;
  min-width: 300px;
}
nav a {
  color: #b9C6ae;
  font-weight: bold;
  text-decoration: none;
}
nav a:hover {
  color: #fff;
}
main {
  display: flex;
  flex: 1 1 0;
  max-width: 1200px;
  margin: 2% auto;
}
section {
  flex: 1 1 0;
  padding-right: 5%;
}
article {
  border-bottom: 1px solid #d8a47f;
}
article p {
  margin: 2% 0;
  font-size: 110%;
}
```

```css
article ul {
  margin-left: 5%;
  margin-bottom: 2%;
  font-size: 110%;
  list-style: circle;
}
aside {
  width: 25%;
}
aside h2 {
  font-size: 150%;
}
input,
button {
  width: 100%;
  margin: 2% 0;
  padding: 1%;
}
footer {
  display: flex;
  justify-content: center;
  padding: 2%;
  background-color: #13293d;
  color: #fff;
}
@media screen and (max-width: 768px) {
  header, main {
    flex-direction: column;
  }
  section {
    padding-right: 0;
  }
  aside {
    width: 100%;
    text-align: center;
  }
}
```

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Default Form Styles</title>
</head>
<body>
  <form>
    <h1>Default Form Styles</h1>
    <p>
      Use the Chrome DevTools to inspect the elements. Try to force <code>:focus</code> state on some of them and look for CSS properties like <code>appearance</code>, <code>outline</code>, and <code>cursor</code>.
    </p>
    <p>
      <label for="username">Name:</label>
      <!-- This element has a CSS declaration of `cursor: text` -->
      <input id="username" type="text" placeholder="First Last" />
    </p>
    <p>
      <label  for="email">E-mail:</label>
      <input id="email" type="text" placeholder="me@example.com" />
    </p>
    <p>
      <label for="expertise">CSS level of expertise:</label>
      <!-- This element has a CSS declaration of `appearance: menulist` -->
      <select id="expertise">
        <option>Beginner</option>
        <option selected>Intermediate</option>
        <option>Advanced</option>
      </select>
    </p>
    <p>
      <label for="comment">Comment:</label>
      <textarea id="comment" placeholder="Great article!"></textarea>
    </p>
    <p>
      <input class="checkbox" id="agreement" type="checkbox" />
      <label for="agreement">I agree to share my e-mail address with The CSS Blog</label>
    </p>
    <p>
      <!-- When this element is in a :focus state, the CSS `outline` property is applied -->
      <button>Submit</button>
    </p>
  </form>
</body>
</html>
```

Default Form Styles

Use the Chrome DevTools to inspect the elements. Try to force `:focus` state on some of them and look for CSS properties like `appearance`, `outline`, and `cursor`.

Name: `First Last`

E-mail: `me@example.com`

CSS level of expertise: Intermediate ▾

Comment: `Great article!`

☐ I agree to share my e-mail address with The CSS Blog

Submit

Wire Framing

CSS Snippet Cheatsheet Wireframe

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum quam augue, suscipit eu accumsan non, ultricies vitae massa.

The areas outlined in red represent rows.

The areas outlined in blue represent columns.

CSS Selectors

```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Advanced Selectors with CSS</title>
  <link rel="stylesheet" href="./assets/css/reset.css" />
  <link rel="stylesheet" href="./assets/css/style.css" />
</head>
<body>
  <header>
    <h1>The CSS Blog</h1>
    <nav>
      <ul>
        <li><a href="#">Home</a></li>
        <li><a href="#">About</a></li>
        <li><a href="#">Contact</a></li>
        <li><a href="#">Resources</a></li>
      </ul>
    </nav>
  </header>
  <main>
    <section>
      <article>
        <h2>A List of Learning Resources</h2>
        <div>
          <p>A successful web developer is one that knows that there's always room
            to grow and learn. You may not use every single tool that's offered to
            you, but it's always good to know that certain tools exist. Check out
            these resources to broaden your knowledge:</p>
          <ul>
            <li>
              Mozilla Developer Network's documentation on <a
              href="https://developer.mozilla.org/en-US/docs/Web/CSS">
              Cascading Style Sheets (CSS)</a>
              is a good resource for all things CSS
              and web development in general.
            </li>
            <li>Our own <a href="./assets/css/reset.css">reset.css</a>
              file shows how to set default browser styles.
            </li>
            <li>Another reset style sheet called
              <a href="https://necolas.github.io/normalize.css/8.0.1/normalize.css">
              normalize.css</a>
              is popular in the web development community</li>
            <li>Our own <a href="./assets/css/style.css">style.css</a>
              file shows how we've designed the page we're reading!
            </li>
            <li>The popular blog, <a href="https://css-tricks.com/">CSS Tricks,</a>
              has a lot of articles and resources to make you a better developer!
            </li>
          </ul>
          <p>Lorem ipsum dolor sit amet consectetur adipisicing elit.
            Nihil natus recusandae amet nostrum odit!</p>
          <p>Lorem ipsum dolor sit amet consectetur adipisicing elit.
            Nihil natus recusandae amet nostrum</p>
        </div>
      </article>
    </section>
  </main>
  <footer>
    <span>Thanks for visiting!</span>
  </footer>
</body>
</html>
```
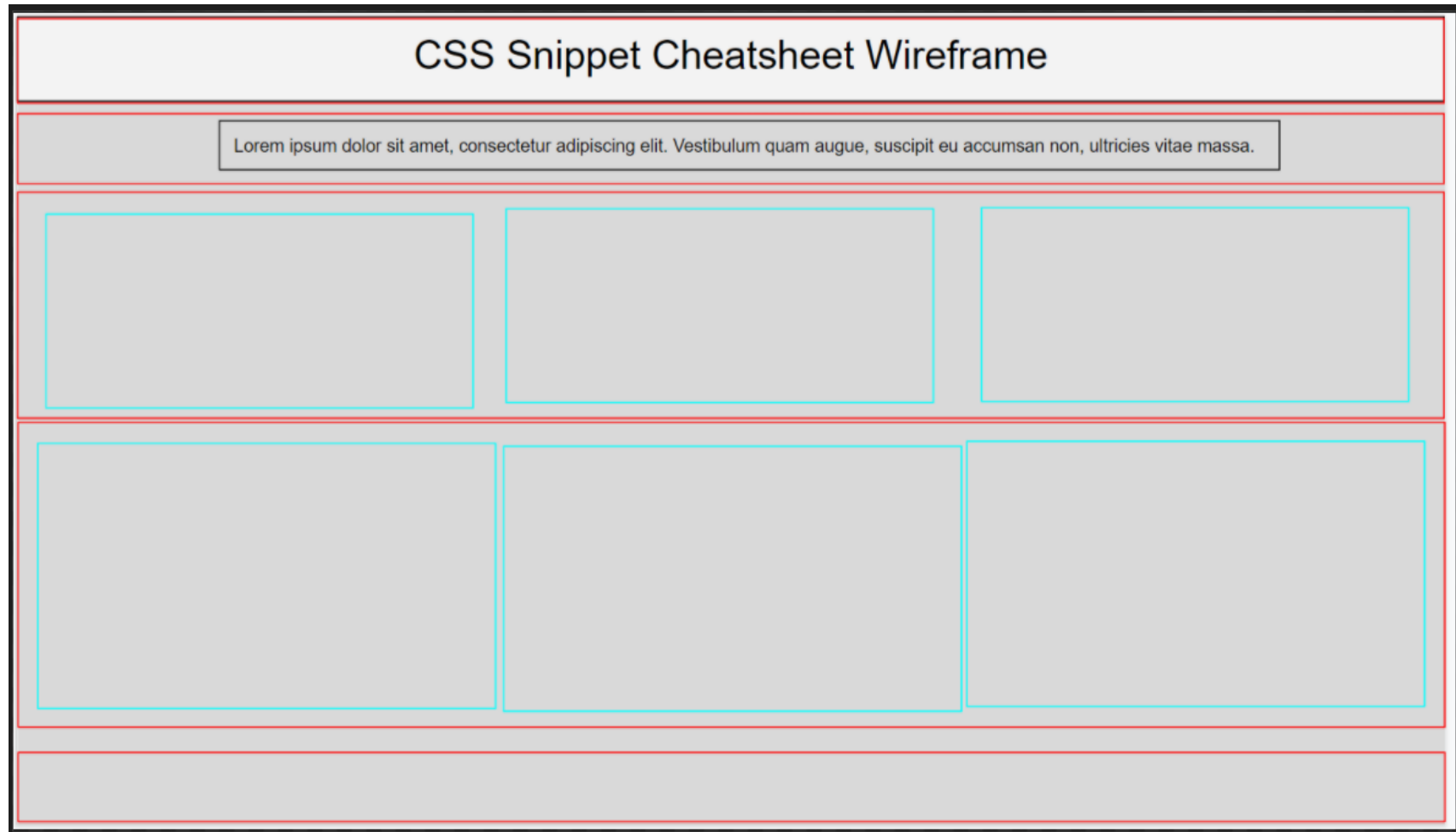
```css
body {
  display: flex;
  flex-direction: column;
  line-height: 1.3;
}
header {
  display: flex;
  justify-content: space-between;
  align-items: center;
  padding: 1%;
  background-color: #13293d;
  color: #fff;
}
h1 {
  font-size: 200%;
}
h2 {
  margin: 2% 0;
  font-size: 250%;
}
nav ul {
  display: flex;
  justify-content: space-between;
  min-width: 300px;
}
nav a {
  color: #b9c6ae;
  font-weight: bold;
  text-decoration: none;
}
nav a:hover {
  color: #fff;
}
main {
  display: flex;
  flex: 1 1 0;
  max-width: 96%;
  margin: 2% auto;
}
article {
  border-bottom: 1px solid #13293d;
}
article p {
  margin: 2% 0;
  font-size: 110%;
}
```

```css
article ul {
  margin-left: 5%;
  font-size: 110%;
  list-style: circle;
}
article li {
  margin: 5px 0;
}
/* Use :nth-child() to select every even list item */
article ul li:nth-child(even) {
  background-color: #b9c6ae;
  color: #13293d;
}
/* Use adjacent sibling combinator (+)
to select all <p> elements that come immediately after a <ul>
element inside of an <article> element */
article ul + p {
  background-color: #13293d;
  color: #fff;
}
/* Use child combinator (>) with a wildcard selector (*)
to select every single element that is a direct descendant (or child)
of any <div> element in an <article> element */
article div > * {
  padding: 20px;
  border: 4px solid #b9c6ae;
}
footer {
  display: flex;
  justify-content: center;
  padding: 2%;
  background-color: #13293d;
  color: #fff;
}
@media screen and (max-width: 768px) {
  header {
    flex-direction: column;
  }
  main {
    max-width: 1200px;
  }
}
```

The CSS Blog   Home   About   Contact   Resources

A List of Learning Resources

A successful web developer is one that knows that there's always room to grow and learn. You may not use every single tool that's offered to you, but it's always good to know that certain tools exist. Check out these resources to broaden your knowledge:

- Mozilla Developer Network's documentation on Cascading Style Sheets (CSS) is a good resource for all things CSS and web development in general.
- Our own reset.css file shows how to set default browser styles.
- Another reset style sheet called normalize.css is popular in the web development community
- Our own style.css file shows how we've designed the page we're reading!
- The popular blog, CSS Tricks, has a lot of articles and resources to make you a better developer!

Lorem ipsum dolor sit amet consectetur adipisicing elit. Nihil natus recusandae amet nostrum odit, ut unde non sint, quia sed inventore odio, delectus iure! Eligendi vitae fuga blanditiis accusamus tempore! Lorem ipsum dolor sit amet consectetur adipisicing elit. Consectetur, nisi, saepe quas sint consequuntur esse voluptatum, dolores dolor aspernatur neque laborum dolorum dolorem repudiandae. Dignissimos eveniet sequi corporis tempore ullam!

Lorem ipsum dolor sit amet consectetur adipisicing elit. Nihil natus recusandae amet nostrum odit, ut unde non sint, quia sed inventore odio, delectus iure! Eligendi vitae fuga blanditiis accusamus tempore!

```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Advanced Selectors with CSS</title>
  <link rel="stylesheet" href="./assets/css/reset.css" />
  <link rel="stylesheet" href="./assets/css/style.css" />
</head>
<body>
  <header>
    <h1>The CSS Blog</h1>
    <nav>
      <ul>
        <li><a href="#">Home</a></li>
        <li><a href="#">About</a></li>
        <li><a href="#">Contact</a></li>
        <li><a href="#">Resources</a></li>
      </ul>
    </nav>
  </header>
  <main>
    <section>
      <article>
        <h2>A List of Learning Resources</h2>
        <div>
          <p>A successful web developer is one that knows that there's always room to grow and learn.
            You may not use every single tool that's offered to you,
            but it's always good to know that certain tools exist.
            Check out these resources to broaden your knowledge:</p>
          <ul>
            <li>
              Mozilla Developer Network's documentation on <a href="
                https://developer.mozilla.org/en-US/docs/Web/CSS">Cascading Style Sheets (CSS)</a>
                is a good resource for all things CSS and web development in general.
            </li>
            <li>Our own <a href="./assets/css/reset.css">reset.css</a>
                file shows how to set default browser styles.
            </li>
            <li>Another reset style sheet called
              <a href="https://necolas.github.io/normalize.css/8.0.1/normalize.css">normalize.css</a>
                is popular in the web development community</li>
            <li>Our own <a href="./assets/css/style.css">style.css</a>
                file shows how we've designed the page we're reading!
            </li>
            <li>The popular blog, <a href="https://css-tricks.com/">CSS Tricks,</a>
                has a lot of articles and resources to make you a better developer!
            </li>
          </ul>
          <p>Lorem ipsum dolor sit amet consectetur adipisicing elit.
              Nihil natus recusandae amet nostrum odit, ut !</p>
          <p>Lorem ipsum dolor sit amet consectetur adipisicing elit.
              Nihil natus recusandae amet nostrum odit, ut unde tempore!</p>
        </div>
      </article>
    </section>
  </main>
  <footer>
    <span>Thanks for visiting!</span>
  </footer>
</body>
</html>
```



```css
/* create custom CSS variables (also called custom properties)
and apply them to the root of the page, making them accessible to all other CSS */
:root {
  /* declare a variable with `--` syntax */
  --dark: #13293d;
  --light: #fff;
  --navlink-color: #b9c6ae;
}
body {
  display: flex;
  flex-direction: column;
  line-height: 1.3;
}
header {
  display: flex;
  justify-content: space-between;
  align-items: center;
  padding: 1%;
  /* use CSS var() function to reference a variable created above */
  background-color: var(--dark);
  color: var(--light);
}
h1 {
  font-size: 200%;
}
h2 {
  margin: 2% 0;
  font-size: 250%;
}
a[href*='.css']::after {
  content: '□';
  display: inline-block;
  margin-left: 3px;
}
nav ul {
  display: flex;
  justify-content: space-between;
  min-width: 300px;
}
nav a {
  background-color: unset;
  color: var(--navlink-color);
  font-weight: bold;
  text-decoration: none;
}

nav a:hover {
  color: var(--light);
}
main {
  display: flex;
  flex: 1 1 0;
  max-width: 96%;
  margin: 2% auto;
}
article {
  /* If we were to change the value of `--
dark`, then all references to it would change automatically */
  border-bottom: 1px solid var(--dark);
}
article p {
  margin: 2% 0;
  font-size: 110%;
}
article ul {
  margin-left: 5%;
  font-size: 110%;
  list-style: circle;
}
footer {
  display: flex;
  justify-content: center;
  padding: 2%;
  background-color: var(--dark);
  color: var(--light);
}
@media screen and (max-width: 768px) {
  header {
    flex-direction: column;
  }
  main {
    max-width: 1200px;
  }
}
```

```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>CSS Snippet Cheatsheet</title>
    <link rel="stylesheet" href="assets/css/reset.css" />
    <link rel="stylesheet" href="./assets/css/style.css" />
  </head>
  <body>
    <header>
      <h1 class="page-title">CSS Snippet Cheatsheet</h1>
      <p>
        Ever have trouble recalling the exact syntax for your favorite CSS code?
        Give it a permanent home and add it to this page! Select any snippet
        below and it'll automatically select all of the code for you to copy.
      </p>
    </header>
    <main>
      <section class="row justify-center">
        <div class="card-column">
          <figure class="card code-card">
            <h2 class="card-header">Flexbox Row</h2>
            <div class="card-body">
              <p>Use these three properties to create a Flexbox row layout.</p>
            </div>
            <!-- with the <pre> element, it counts all spaces literally,
                 so proper code indentation cannot be applied in this case -->
            <pre class="code-block"><code>.row {
                display: flex;
                flex-direction: row;
                flex-wrap: wrap;
            }</code></pre>
          </figure>
        </div>
        <div class="card-column">
          <figure class="card code-card">
            <h2 class="card-header">Flexbox Column</h2>
            <div class="card-body">
              <p>Use this to create a Flexbox column layout.</p>
            </div>
            <pre class="code-block"><code>.column {
                display: flex;
                flex-direction: column
            }</code></pre>
          </figure>
        </div>
        <div class="card-column">
          <figure class="card code-card">
            <h2 class="card-header">CSS Grid Layout</h2>
            <div class="card-body">
              <p>Build a 12-column layout using CSS Grid</p>
            </div>
            <pre class="code-block"><code>.grid {
                display: grid;
                width: 100%;
                grid-template-columns: repeat(12, 1fr);
            }</code></pre>
          </figure>
        </div>
        <div class="card-column">
          <figure class="card code-card">
            <h2 class="card-header">Linear Gradients</h2>
            <div class="card-body">
              <p>
                This will create a background linear gradient from top to
                bottom.
              </p>
            </div>
            <pre class="code-block"><code>.linear-gradient-background {
                background-image: linear-gradient(
                  rgba(232, 102, 236, 0.3) 0%,
                  rgba(232, 102, 236, 0.6) 100%
                );
            }</code></pre>
          </figure>
        </div>
        <div class="card-column">
          <figure class="card code-card">
            <h2 class="card-header">Box Transition Glow</h2>
            <div class="card-body">
              <p>Use transition and box shadows to glow on hover.</p>
            </div>
            <pre class="code-block"><code>.code-card .card-header {
                border-radius: 8px;
                transition: all 0.5s ease-in-out;
            }

            .code-card:hover,
            .code-card:hover .card-header {
                box-shadow: inset 0px 0px 8px
                rgba(232, 102, 236, 1), 0 0 15px rgba(232, 102, 236, 1);
            }</code></pre>
          </figure>
        </div>
        <div class="card-column">
          <figure class="card code-card">
            <h2 class="card-header">Overlay Card with Title</h2>
            <div class="card-body">
              <p>
                Use position properties and negative margins to raise elements
                higher than their natural starting point.
              </p>
            </div>
            <pre class="code-block"><code>.card-header {
                position: relative;
                margin-top: -20px;
            }</code></pre>
          </figure>
        </div>
      </section>
    </main>
    <footer>
      <h3>Made with <span role="img" aria-label="heart">♡</span> and CSS</h3>
    </footer>
  </body>
</html>
```

```css
/* color scheme sourced from http://khroma.co/generator/ */
:root {
  --outer-space: white;
  --bright-turquoise: #2aefcb;
  --gin: black;
  --heliotrope: #e866ec;}
/* set content highlight color and background */
::selection {
  background-color: var(--bright-turquoise);
  color: var(--outer-space);}
html {
  /* set 1rem to 10px instead of native 16px to make it easier to calculate font-
size values
                    for the rest of the page (e.g. 1.6rem is 16px, 2.0rem is 20px, etc.)
*/
  font-size: 62.5%;}
body {
  /* Sans Serif font stack is implemented to help select the best font
                        for each type of computer operating system */
  font-family: -apple-
system, BlinkMacSystemFont, "Segoe UI", Roboto, Helvetica,
  Arial, sans-serif, "Apple Color Emoji", "Segoe UI Emoji", "Segoe UI Symbol";
  /* set body font size to 1.75rem, which is 1.75x
                        more than the font-size of 10px, making it 17.5px  */
  font-size: 1.75rem;
  line-height: 1.5;
  color: var(--gin);
  background-color: var(--outer-space);}
h1, h2, h3, h4, h5, h6 {
  margin-bottom: 0.625rem;
  color: var(--heliotrope);
  font-weight: 700;
  line-height: 1.1;}
h1 {
  /* 4rem = 40px (4 * 10px) */
  font-size: 4rem;}
h2 {
  font-size: 3.5rem;}
h3 {
  font-size: 3rem;}
h4 {
  font-size: 2.5rem;}
h5 {
  font-size: 2rem;}
h6 {
  font-size: 1.5rem;}
p {
  margin-bottom: 1rem;}
/* Monospace font stack is used for our code blocks
                        to make it look more like code */
code {
  font-family: Consolas, "Andale Mono WT", "Andale Mono", "Lucida Console",
  "Lucida Sans Typewriter", "DejaVu Sans Mono", "Bitstream Vera Sans Mono",
  "Liberation Mono", "Nimbus Mono L", Monaco, "Courier New", Courier,
  monospace;}

header {
  width: 96%;
  margin: 0 auto;
  padding: 2.5rem;
  text-align: center;}
header h1.page-title {
  padding: 1rem 1.2rem;
  font-size: 4.5rem;}
main {
  width: 96%;
  margin: 2rem auto;}
footer {
  margin: 1.5rem;
  text-align: center;}
/* create our main grid with flexbox, use negative margins
                        to offset the padding on the .card-column class */
.row {
  display: flex;
  flex-wrap: wrap;
  flex-direction: row;
  margin-left: -1.5rem;
  margin-right: -1.5rem;}
.justify-center {
  justify-content: center;}
/* using a mobile-first approach, set base column width to be 100%
                        and we'll update it using media queries for larger screen sizes */
.card-column {
  flex: 1 0 100%;
  padding: 1.5rem;
  /* when the screen resizes, make it so our columns don't jump in size */
  transition: flex-basis 500ms linear;
  margin-bottom: 1.5rem;}
/* make each code card a flexbox column to align and justify content within it */
.code-card {
  display: flex;
  flex-direction: column;
  justify-content: flex-start;
  min-height: 100%;
  padding: 2rem;
  color: var(--heliotrope);
  /* outline is like border, but on the outside of the box instead of inside */
  outline: 2px dashed var(--gin);
  outline-offset: -2px;
  transition: all 0.5s ease-in-out;}
.code-card .card-header {
  /* use position and negative margins to move title of card over the card's outline
                        and create an overlap effect */
  position: relative;
  margin: -4rem 0 1.5rem 0;
  padding: 0.5rem 1rem;
  background-color: var(--outer-space);
  color: var(--heliotrope);
  border: 2px solid var(--gin);
  border-radius: 8px;
  transition: all 0.5s ease-in-out;}

/* set an outer and inner drop shadow on the card
                        to create a glowing effect on hover */
.code-card:hover,
.code-card:hover .card-header {
  box-shadow: inset 0px 0px 8px var(--heliotrope), 0 0 15px var(--heliotrope);}
/* each card's code block grows to the height of the largest code block
                        on the page, making each card even in height */
.code-card .code-block {
  flex-grow: 1;}
.code-card pre {
  /* set styles to make it so code wraps in <pre> instead of overflowing */
  white-space: pre-wrap;
  overflow: auto;
  tab-size: 4;
  padding: 1.2rem 1rem;
  color: var(--gin);
  border-radius: 8px;
  /* use linear-gradient() function to create a fading background  */
  background-image: linear-gradient(
    rgba(232, 102, 236, 0.3) 0%,
    rgba(232, 102, 236, 0.6) 100%);
  display: flex;
  align-items: center;}
/* on <pre> click, select all contents so the code is highlighted */
.code-card pre:active {
  user-select: all;}
/* using a mobile-first approach, use `min-width` values
                        going from narrow -> wide for a responsive design */
@media screen and (min-width: 768px) {
  /* on tablets, set columns to only be 50% of the width and not grow */
  .card-column {
    flex: 0 0 50%;
    max-width: 50%;}
  /* select the last child to grow to full width if it has space to grow */
  .card-column:last-child {
    flex-grow: 1;}
}
/* media query for larger screens */
@media screen and (min-width: 992px) {
  header {
    width: 75%;}
  .card-column {
    flex: 0 0 33.333%;
    max-width: 33.333%;}
}
```

Rendered preview

CSS Snippet Cheatsheet

Ever have trouble recalling the exact syntax for your favorite CSS code? Give it a permanent home and add it to this page! Select any snippet below and it'll automatically select all of the code for you to copy.

Flexbox Row

Use these three properties to create a Flexbox row layout.

```css
.row {
    display: flex;
    flex-direction: row;
    flex-wrap: wrap;
}
```

Flexbox Column

Use this to create a Flexbox column layout.

```css
.column {
    display: flex;
    flex-direction: column
}
```

CSS Grid Layout

Build a 12-column layout using CSS Grid

```css
.grid {
    display: grid;
    width: 100%;
    grid-template-columns: repeat(12, 1fr);
}
```

Linear Gradients

This will create a background linear gradient from top to bottom.

```css
.linear-gradient-background {
    background-image: linear-gradient(
        rgba(232, 102, 236, 0.3) 0%,
        rgba(232, 102, 236, 0.6) 100%
    );
}
```

Box Transition Glow

Use transition and box shadows to glow on hover.

```css
.code-card .card-header {
    border-radius: 8px;
    transition: all 0.5s ease-in-out;
}

.code-card:hover,
.code-card:hover .card-header {
    box-shadow: inset 0px 0px 8px rgba(232, 102, 236, 1), 0 0 15px rgba(232, 102, 236, 1);
}
```

Overlay Card with Title

Use position properties and negative margins to raise elements higher than their natural starting point.

```css
.card-header {
    position: relative;
    margin-top: -20px;
}
```

Made with ♥ and CSS