Introduction to HTML + CSS

Google

Google Search   I'm Feeling Lucky

- Account
- Search
- Maps
- YouTube
- Play
- News
- Gmail
- Contacts
- Drive
- Calendar
- Translate
- Photos
- Shopping

More

- Docs
- Books
- Blogger
- Duo
- Hangouts
- Keep
- Jamboard
- Earth
- Collections

Even more from Google
Sign in

Google Search   I'm Feeling Lucky

HTML

💡 Building a web page

The structure of a web page is determined by the **HTML**.

HTML is made up of many different **elements**, each serving a different purpose.

An element usually consists of a **start tag** and **end tag**, with content inserted in between.

HTML Elements

The building blocks of the page.

HTML

Head

Title

Navigation

Body

Division

Footer

Heading

Paragraph

Anchor

Image

```html
<html>
    <head>
        <title>She Codes Cupcakes</title>
    </head>
    <body>
    <header>
        <h1>She Codes Cupcakes.</h1>
        <nav>
            <a href="index.html">Home</a>
        </nav>
    </header>
    <div>
        <p>Below is a photo of the cupcakes we have at most She Codes events.</p>
        <img src="images/cupcake-1.jpg" alt="She Codes branded Cupcakes.">
    </div>
    <footer>
        She Codes Cupcakes.
    </footer>
    </body>
</html>
```

👉 Building a web page

Open `introduction-to-html` in your code editor.

The folder contains two web pages, `index.html` and `cupcake-effect.html`.

Open `index.html` in your code editor and a web browser.

👉 `<html>`

Add the following to `index.html`

```
<!doctype html>
<html lang="en">

</html>
```

👉 Add a head section

Add the following to `index.html`

```
<!doctype html>
<html lang="en">
    <head>
        <meta charset="utf-8">
        <title>She Codes Cupcakes</title>
        <meta name="description" content="She Codes Cupcakes">
        <meta name="author" content="She Codes">
    </head>
</html>
```

✅ Open `index.html` in a browser!

👉 Link to a stylesheet

```html
<!doctype html>
<html lang="en">
    <head>
        <meta charset="utf-8">
        <title>She Codes Cupcakes</title>
        <meta name="description" content="She Codes Cupcakes">
        <meta name="author" content="She Codes">
        <link rel="stylesheet" href="styles.css">
    </head>
</html>
```

✅ Refresh your browser and see what changed!

👉 Add a body section

```
    </head>
    <body>
    <!-- the content for our webpage will go here -->
    </body>
</html>
```

✅ Refresh your browser and see what changed!

👉 Add a header section

```
    </head>
    <body>
        <header>
        </header>
    </body>
</html>
```

👉 Add some headings

```
    </head>
    <body>
        <header>
            <h1>She Codes</h1>
            <h2>{Cupcakes}</h2>
        </header>
    </body>
</html>
```

✅ Refresh your browser and see what changed!

👉 Add a navigation section

```
<body>
    <header>
        <h1>She Codes</h1>
        <h2>{Cupcakes}</h2>
        <nav>
        </nav>
    </header>
</body>
</html>
```

👉 **Add some links to the navigation**

```
<body>
    <header>
        <h1>She Codes</h1>
        <h2>{Cupcakes}</h2>
        <nav>
            <a href="index.html">Home</a>
            <a href="cupcake-effect.html">The Cupcake Effect</a>
        </nav>
    </header>
</body>
```

✅ Refresh your browser and see what changed!

👉 Add a div

```
        </header>
        <div>
        </div>
    </body>
</html>
```

👉 Add a paragraph

```
        </header>
        <div>
            <p>Check out our awesome designer cupcakes!</p>
        </div>
    </body>
</html>
```

✅ Refresh your browser and see what changed!

👉 Add an image

```
    </header>
    <div>
        <p>Check out our awesome designer cupcakes!</p>
        <img src="cupcake-1.jpg" alt="She Codes Cupcakes.">
    </div>
  </body>
</html>
```

✅ Refresh your browser and see what changed!

👉 Change the image path

```
        </header>
        <div>
            <p>Check out our awesome designer cupcakes!</p>
            <img src="images/cupcake-1.jpg" alt="She Codes Cupcakes.">
        </div>
    </body>
</html>
```

✅ Refresh your browser and see what changed!

👉 Add a list

```
<div>
    <p>Check out our awesome designer cupcakes!</p>
    <img src="images/cupcake-1.jpg" alt="She Codes Cupcakes.">
    <p>Our Cupcakes are:</p>
    <ul>
        <li>Colourful</li>
        <li>Creative</li>
        <li>Delicious</li>
    </ul>
</div>
```

✅ Refresh your browser and see what changed!

👉 Add a footer

```
        </div>
        <footer>
            She Codes Cupcakes.
        </footer>
    </body>
</html>
```

✅ Refresh your browser and see what changed!

🤔 Challenge

Complete the "Cupcake Effect" page. It needs to include:

- 1 image in a div
- A div containing the text content for the page.
  - This div will contain three more divs.
  - Each of these three divs will include a name (h3 tag) and a quote (paragraph tag).

Complete the "Recipe" page. It needs to include:

- The same header and footer, with "Cupcake Recipe" as the heading.
- A div containing all of the content for the page:
  - Ingredients and Method headings (h3 tags)
  - An unordered list of ingredients.
  - An ordered list of steps for the method.
  - A picture of our cupcakes

Look in `content.txt` for the quotes and recipe.

She Codes
The Cupcake Effect

Home   The Cupcake Effect   Recipe

Kate Kirwin

The cupcake effect is the way an event looks and makes you feel. Its when you get your cupcake at the workshop and its got the logo, glitter, sparkles, and makes you feel joy and makes you feel special, like the event was made for you.

Holly Keogh

A way to make everyone feel special and loved.

Hayley van Waas

The feeling of accomplishment you get when you learn something new and realise you CAN do this. The warm fuzzies you get when you find likeminded people. The pride in showcasing something that you made.

She Codes Cupcakes.

She
Cupca

Home   The

Ingredients

- 2 cups self-raising flour, sifted
- 3/4 cup caster sugar
- 2 eggs, beaten
- 3/4 cup milk
- 125g unsalted butter, melted, cooled
- 1 teaspoon vanilla essence
- 1 1/2 cups pure icing sugar
- 1-1 1/2 tablespoons water
- Pink food colouring, optional

Method

1. Preheat oven to 200C or 180C fan-forced. Grease a 12 x Combine flour and caster sugar in a bowl. Make a well i
2. Add milk, butter, eggs and vanilla to flour mixture. Using
3. Spoon mixture into prepared muffin pan. Bake for 12 to 1 Stand in pan for 5 minutes before transferring to a wire r
4. Make icing: Sift icing sugar into a bowl. Add food colouri cupcakes. Decorate with sprinkles.

She

💡 HTML vs CSS

Comment out the link to the stylesheet

```
<head>
    <meta charset="utf-8">
    <title>She Codes Cupcakes</title>
    <meta name="description" content="She Codes Cupcakes">
    <meta name="author" content="She Codes">
    <!-- link rel="stylesheet" href="styles.css" -->
</head>
```

✅ Refresh your browser and see what changed!

CSS

💡 Styling a web page.

A web page is styled using **CSS**.

CSS defines the **styles**, including the design and layout.

Choosing what part of the HTML to style is done by using **selectors** to access a particular element(s), and then we define what **properties** it will have (e.g. colour, font, borders…).

```css
body {
    margin: 50px;
}


h1 {
    font-size: 3rem;
    color: #ff006f;
    margin: 0;
}


h2 {
    margin: 0;
}


header {
    display: flex;
    justify-content: space-between;
    align-items: center;
    border-bottom: 1px solid #000;
    margin-bottom: 20px;
}
```

CSS Selectors

Accessing an HTML element.

Type

Class

ID

———

👉 Styling a web page

Open `introduction-to-css-selectors` in your code editor.

The folder contains a webpage, made up of `index.html` and `styles.css`.

Open `index.html` in your code editor and a web browser.

H1 Title

H2 Title

Nav Item 1 Nav Item 2 Nav Item 3

A paragraph tag, in a div tag.

Another paragraph tag, in a div.

- List Item 1
- List Item 2
- List Item 3

A footer tag.

💡 Type Selectors

Selects all the elements that match the given **tag** name.

e.g. Given the following html:

```
<div>
    <h2>About Me</h2>
    <p>Here is some info about me.</p>
</div>
```

h2 would match the **<h2>** element.

💡 Type Selectors

Selects all the elements that match the given **tag** name.

e.g. Given the following html:

```html
<div>
    <h2>About Me</h2>
    <p>Here is some info about me.</p>
</div>
```

p   would match the `<p>` element.

👉 Type Selectors

Selects all the elements that match the given **tag** name.

Add the following to `styles.css`:

```css
body {
    background-color: lavender;
}


div {
    background-color: peachpuff;
}
```

✅ Refresh your browser and see what changed!

💡 Class Selectors

Selects all the elements that match the given `class` attribute.

e.g. Given the following html:

```
<div class="box">
    <h2>About Me</h2>
    <p class="body-text">Here is some info about me.</p>
</div>
```

`.box` would match the `<div class="box">` element.

💡 Class Selectors

Selects all the elements that match the given `class` attribute.

e.g. Given the following html:

```
<div class="box">
    <h2>About Me</h2>
    <p class="body-text">Here is some info about me.</p>
</div>
```

`.body-text` would match the `<p class="body-text">` element.

👉 Class Selectors

Add the following to `index.html`:

```html
<ul>
    <li class="list-item">List Item 1</li>
    <li class="list-item">List Item 2</li>
    <li>List Item 3</li>
</ul>
```

Add the following to `styles.css`:

```css
.list-item {
    background-color: darkseagreen;
}
```

✅ Refresh your browser and see what changed!

👉 Class Selectors

Add the following to `index.html`:

```html
<header>
    <h1 class="main-heading">H1 Title</h1>
    <h2 class="main-heading">H2 Title</h2>
    <nav>
```

Add the following to `styles.css`:

```css
.main-heading {
    background-color: lightskyblue ;
}
```

✅ Refresh your browser and see what changed!

💡 ID Selectors

Selects all the elements that match the given **id** attribute.

e.g. Given the following html:

```
<div class="box">
    <p id="first-paragraph" class="body-text">Some text.</p>
    <p class="body-text">Here is some more text.</p>
</div>
```

`#first-paragraph` would match the
`<p id="first-paragraph" class="body-text">` element.

👉 ID Selectors

Add the following to `index.html`:

```html
<div>
    <p id="first-paragraph">A paragraph tag, in a div tag.</p>
    <img src="logo.png" alt="She Codes Logo.">
    <p>Another paragraph tag, in a div.</p>
    <ul>
```

Add the following to `styles.css`:

```css
#first-paragraph {
    background-color: lightcoral;
}
```

✅ Refresh your browser and see what changed!

💡 Class vs ID

We use **class** if we want **multiple** elements to have the same styles applied.

We use **id** if we want only **one** element to have styles applied.

🤔 Challenge

- Change the background of the first two `nav` items to `mediumpurple`.

- Change the background of the last `nav` item to `gold`.

- Update the `div` styling to use an `id` instead of a tag.

CSS Properties

Styling an HTML element.

Display

Width and Height

Margin and Padding

Border

Color

Font

_____

👉 Styling a web page

Open `introduction-to-css` in your code editor.

The folder contains the same web pages we were working with earlier.

Open `index.html` and `styles.css` in your code editor.

Open `index.html` in a web browser.

💡 display

Defines the layout used for its children.

```
display: block;
display: inline;
```

`block` means that the element is displayed as a block, and does not tolerate HTML elements next to it, unless ordered otherwise.

`inline` means that the element is displayed inside the current block, on the same line.
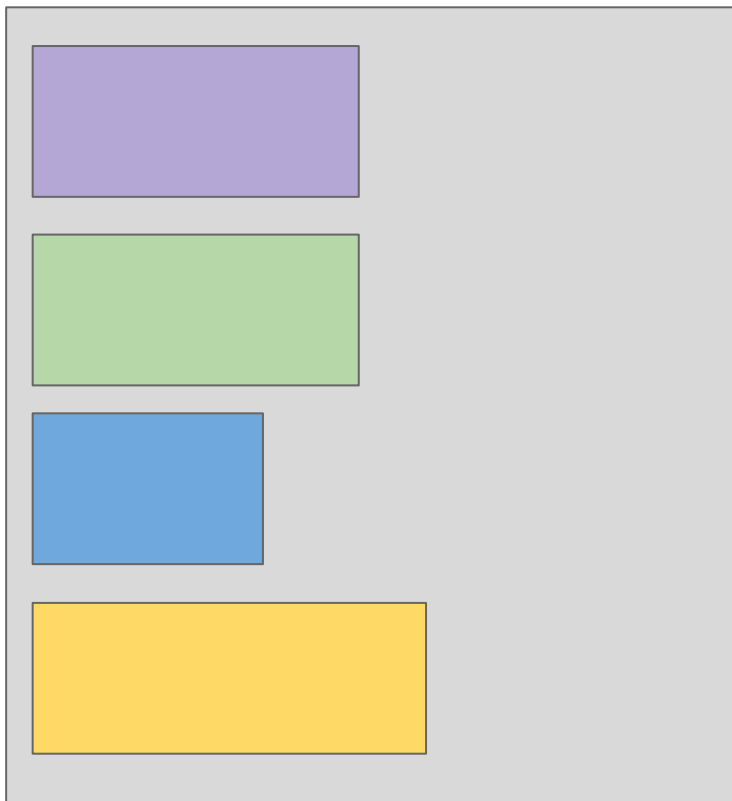
`display: block;`
Block elements stack, regardless of their width.
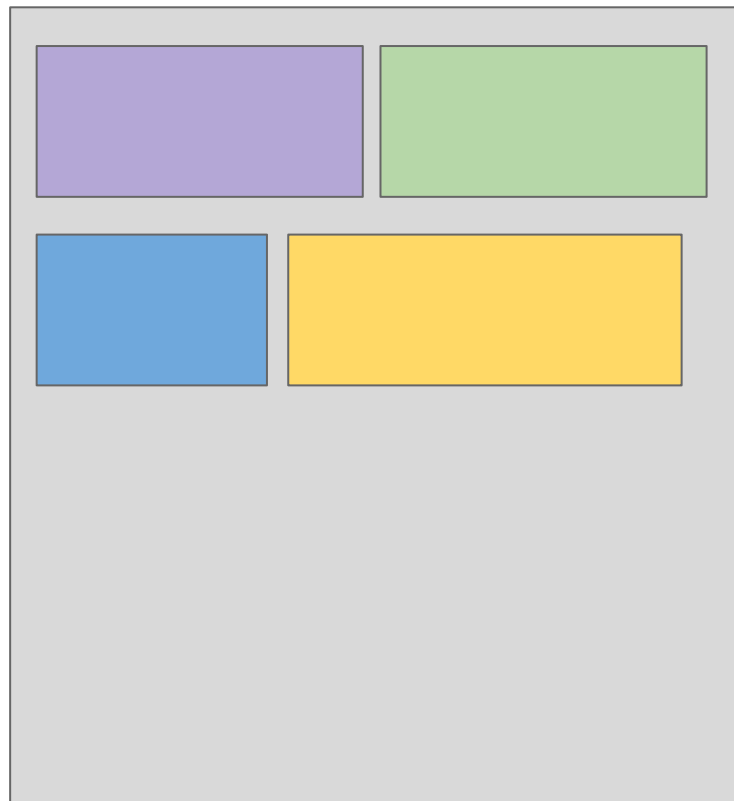
`display: inline;`
Inline elements flow from one line to the next.

👉 display

Add the following to `styles.css`:

```
nav a {
    display: block;
}
```

✅ Refresh your browser and see what changed!

✅ Change block to inline, refresh your browser and see what changed!

💡 width and height

`width` sets an element's width. By default, it uses the width of the parent.

`Height` sets an element's height. By default, it uses the height of the parent.

```
width: 200px | 50% | 20em | auto;
height: 200px | 50% | 20em | auto;
```

👉 `width` and `height`

Add the following lines to `styles.css` to change the size of the image:

```
nav {
    width: 200px;
}
```

✅ Refresh your browser and see what changed!

🤔 Challenge

Change the size of the image on the home page only.

The width should be 300px.

The height should be 200px.

Hint

The ID attribute might be useful here!

💡 **margin and padding**

margin creates extra space **around** an element.

padding creates extra space **within** an element.

Margin
Border
Padding
Content
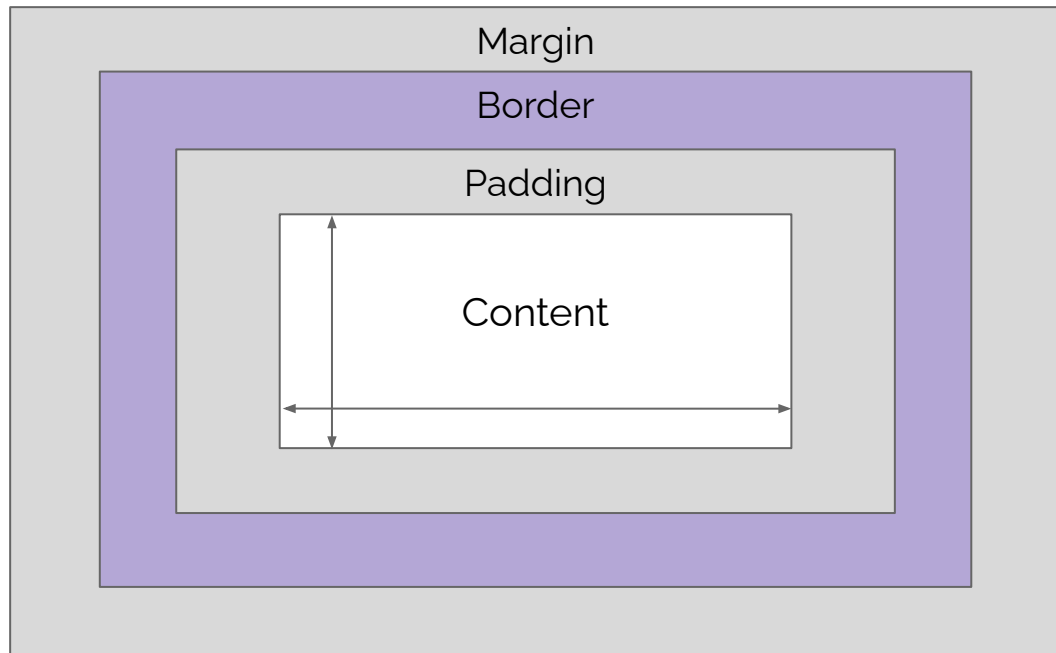
```
margin: <top> <right> <bottom> <left>;
padding: <top> <right> <bottom> <left>;
```

👉 margin and padding

Add the following lines to `styles.css` to change the padding and margins of the header:

```css
header {
    margin: 20px;
    padding: 20px;
}
```

✅ Refresh your browser and see what changed! What happens when you change the values? Have only one of margin or padding?

💡 If you're having trouble seeing the difference, try adding borders to every element on the page:

```css
* {
    border: 1px solid grey;
}
```

💡 **border**

An element can have a border, which has a width, style and color:

```
border: <border-width> <border-style> <border-color>;
```

We can use this shorthand:

```
border: 1px dotted red;
```

Or the longhand:

```
border-width: 1px;
border-style: dotted;
Border-color: red;
```

👉 border

Add a border to the home page image:

```css
img#your-id {
    width: 300px;
    height: 200px;
    border: 3px dashed orange;
}
```

✅ Refresh your browser and see what changed!

Try changing the values and adding borders to different elements!

Let's add some...

...colour!

💡👉 color

We can customise the colour of text, backgrounds, borders, etc.

There are several different ways we can specify colour, here's just 3 ways to specify the same colour:

Name	Hex	RGB
`mediumvioletred`	`#c71585`	`rgb(199,21,133)`

Try searching for `color picker` in Google to play with colours!

Change up the colours already in `styles.css` and add some more of your own!

csfieldguide.org.nz/en/interactives/pixel-viewer/

💡 font

You can set the font and all the attributes that come along with it! For example...

```
font-family: Arial;
font-size: 12px | 2em | 20%;
font-weight: normal | lighter | 500;
line-height: 1rem;
color: #fff | red | rgb(23, 123, 256);
```

👉 font

Add the following lines to `styles.css` to change the font and font size of the header text:

```css
header {
    background-color: lightcoral;
    font-family: Arial;
    font-size: 20px;
}
```

✅ Refresh your browser and see what changed!

Try changing the font and font size of other elements!

💡 **Google** Fonts

Categories ▾   Language ▾   Font Properties ▾

Viewing **960** of 960 font families

Sort by: Trending now ▾

Roboto   ⊕
Christian Robertson (12 styles)

All their equipment and instruments are alive.

Yeon Sung   ⊕
Woowahan brothers (1 style)

A red flare silhouetted the jagged edge of a wing.

McLaren   ⊕
Astigmatic (1 style)

I watched the storm, so beautiful yet terrific.

Open Sans   ⊕
Steve Matteson (10 styles)

Almost before we knew it, we had left the ground

Lato   ⊕
Łukasz Dziedzic (10 styles)

A shining crescent far beneath the flying vessel

Jomolhari   ⊕
Christopher J. Fynn (1 style)

It was going to be a lonely trip back.

👉 Google Fonts

Go to `fonts.google.com` and select a font you like.

👉 Google Fonts

Go to `fonts.google.com` and select a font you like.

Add the link to `styles.css` and apply it to the element you want to use that font:

```css
@import url('https://fonts.googleapis.com/css2?family=Raleway&display=swap');


html {
    font-family: 'Raleway', sans-serif;
    background-color: #fff;
}
```

✅ Refresh your browser and see the new font in action!

Specificity

💡 Specificity

Specificity is a **weighting** that is applied to CSS declarations. **Browsers** use this weighting to determine which CSS properties to apply.

There are three categories for specificity:

1. ID Selectors
2. Class Selectors
3. Type Selectors

💡 **Calculating Specificity**

```
.nav-link {}
```

0 IDs

1 Classes

0 Types

💡 👉 **Calculating Specificity**
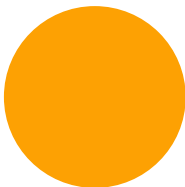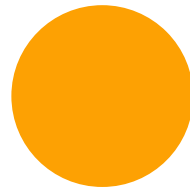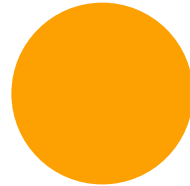
`.nav-link {}`

0 IDs

1 Classes

0 Types

`nav a.nav-link {}`
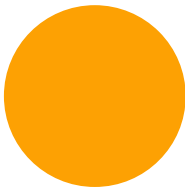
IDs
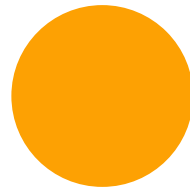
Classes

Types

💡 👉 Calculating Specificity

| `.nav-link {}` | `nav a.nav-link {}` | `nav a.nav-link#link-1 {}` |

0 IDs **0** IDs ⬤ IDs

1 Classes **1** Classes ⬤ Classes

0 Types **2** Types ⬤ Types

💡 👉 **Calculating Specificity**

Most important

Least important

`.nav-link {}`	`nav a.nav-link {}`	`nav a.nav-link#link-1 {}`
0 IDs	**0** IDs	**1** IDs
1 Classes	**1** Classes	**1** Classes
0 Types	**2** Types	**2** Types

🤔Challenge

What is the colour of the following text?

1. Nav link 1
2. Nav link 2
3. Nav link 3

```html
<html>
    <body>
        <header>
            <nav>
                <a class="nav-link" id="link-1" href="#">Nav link 1</a>
                <a class="nav-link" href="#">Nav link 2</a>
                <a class="nav-link" href="#">Nav link 3</a>
            </nav>
        </header>
    </body>
</html>
```

```css
nav a {
    color: red;
}

.nav-link {
    color: black;
}

nav a.nav-link {
    color: green;
}

nav a.nav-link#link-1 {
    color: purple;
}
```

🤔Challenge

What is the colour of the following text?

1. Paragraph 1
2. Paragraph 2
3. Paragraph 3

```html
<html>
    <body>
        <div>
            <p class="paragraph-link" id="paragraph-1">Paragraph 1</p>
            <p class="paragraph-link">Paragraph 2</p>
            <p class="paragraph-link">Paragraph 3</p>
        </div>
    </body>
</html>
```

```css
html div p {
    color: darkgrey;
}

body div p {
    color: hotpink;
}
```

🤔 Challenge

What is the colour of the following text?

1. Paragraph 1
2. Paragraph 2
3. Paragraph 3

```html
<html>
    <body>
        <div>
            <p class="paragraph-link" id="paragraph-1">Paragraph 1</p>
            <p class="paragraph-link">Paragraph 2</p>
            <p class="paragraph-link">Paragraph 3</p>
        </div>
    </body>
</html>
```

```css
body div p {
    color: hotpink;
}

html div p {
    color: darkgrey;
}
```

💡 👉 Calculating Specificity

Open `css-specificity` you code editor and play with the different CSS declarations.

Hovering over a CSS declaration in VS Code will reveal the elements it matches, as well as its specificity.