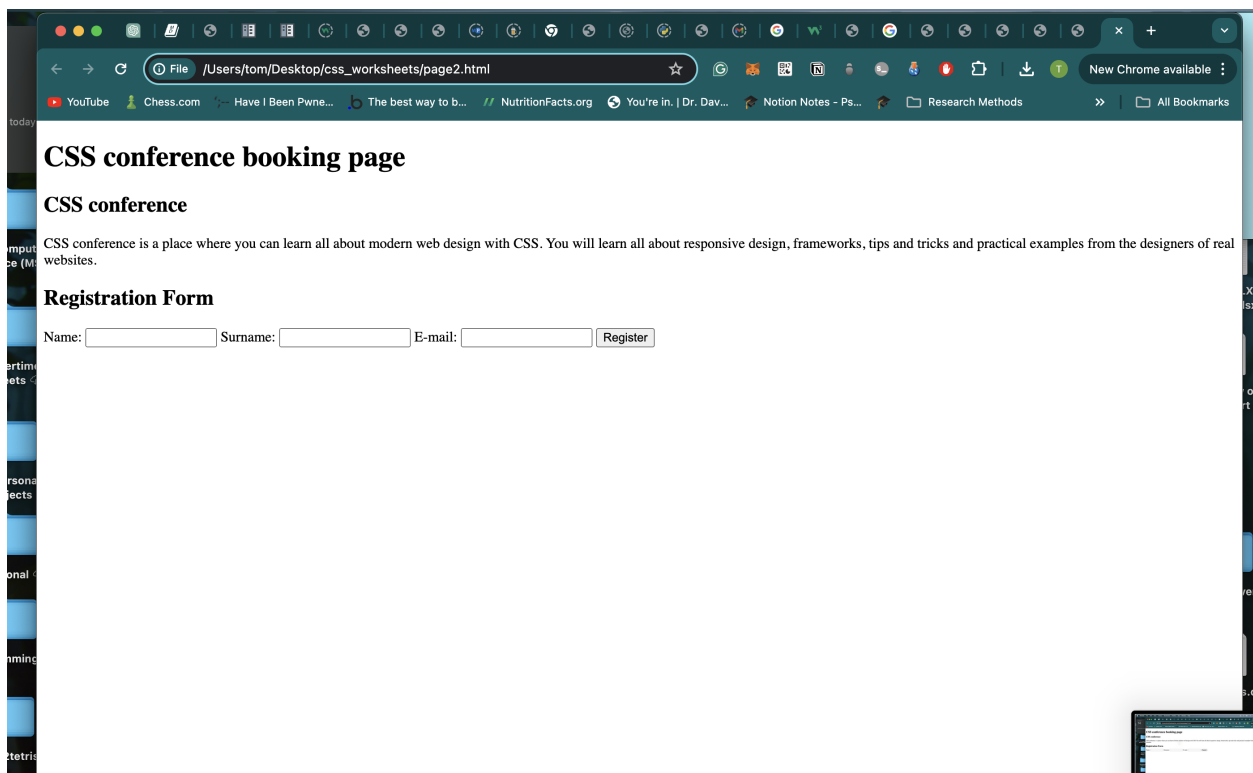


CSS Frameworks (1)

Bulma

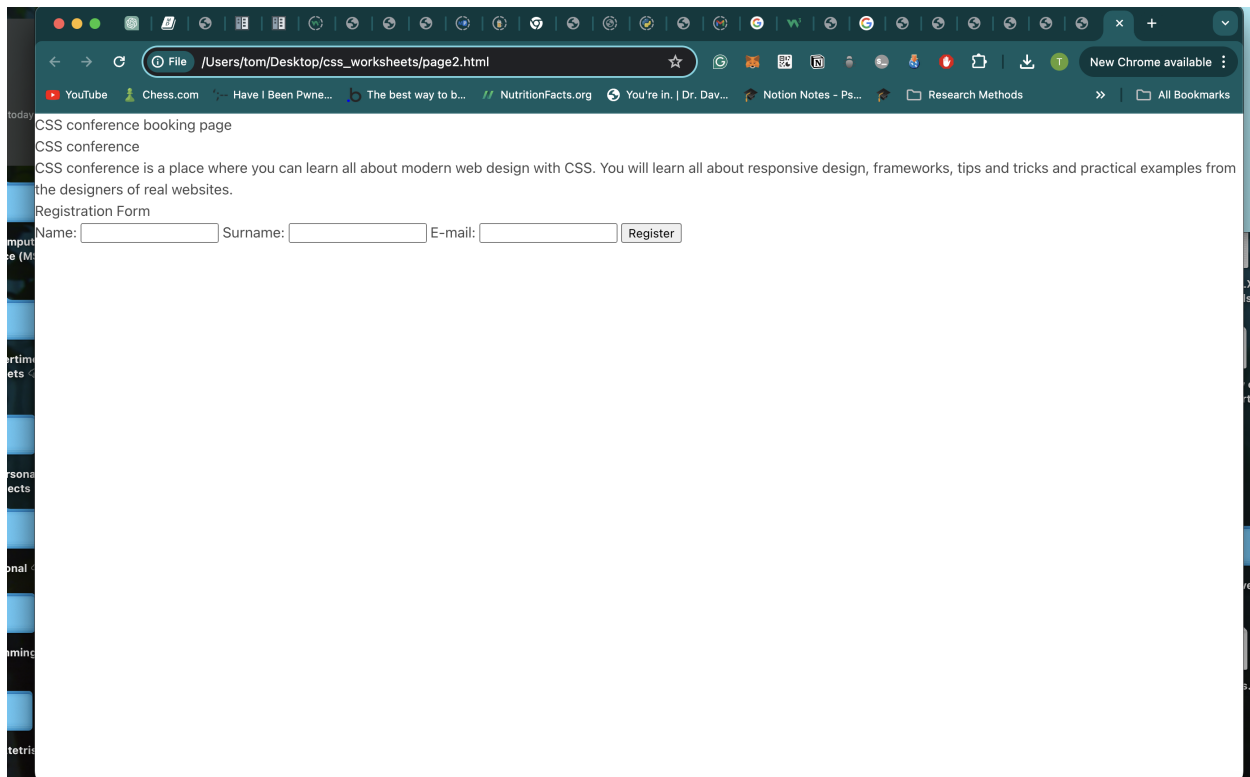
Bulma is a larger but still fairly lightweight CSS framework. Unlike Milligram, it only styles parts of the page you tell it to (but it sets a default font), and it has some more components to build things like menus or panels.



Start with [page2.html](#), have a look at the raw page and then add the following to the header (the viewport tag is already there):

```
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bul
```

```
ma@0.9.3/css/bulma.css">
```



Advanced Note

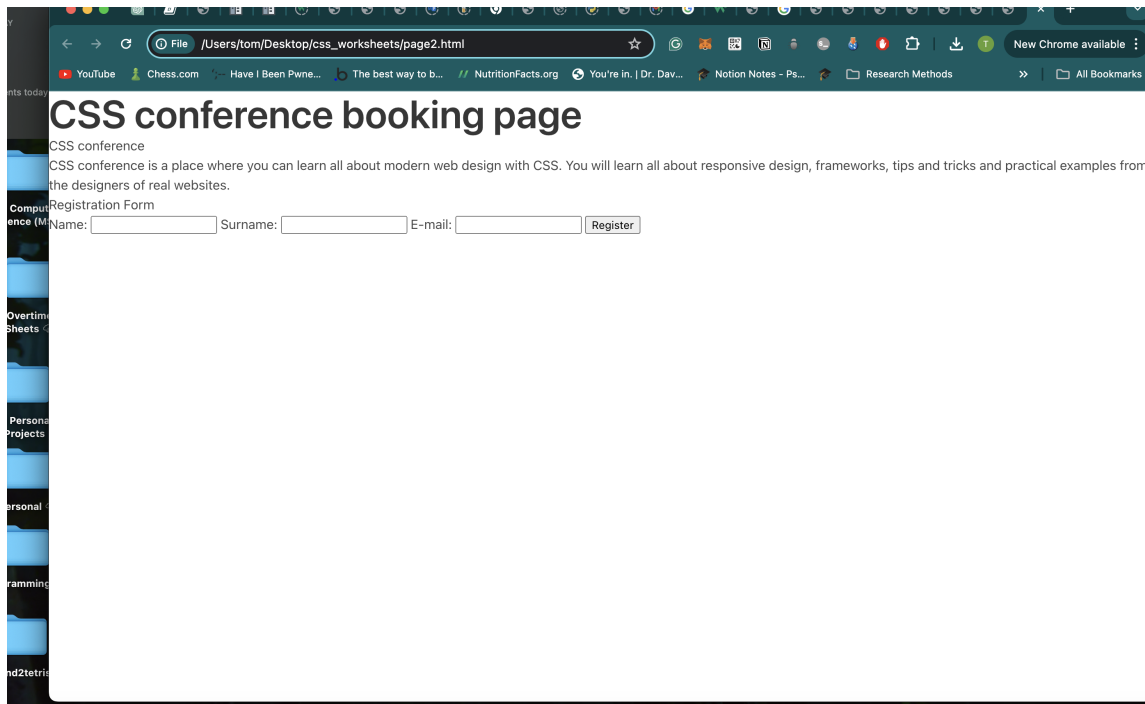
Bulma, like most frameworks, comes in two formats: `bulma.css` is the CSS file with human-readable formatting, and `bulma.min.css` (at the same location) is the minified version with all unnecessary whitespace removed and so is slightly quicker to download.

The bulma sources [on github](#) are actually in SASS format, a language that compiles to CSS but offers extra features in the source file like nested blocks and variables. (After SASS was created, variables were introduced in CSS too, but with a different syntax.)

Bulma works with custom class names. For example, we are going to make the first heading into a banner that spans the whole page width - called a *hero* in web design circles. Bulma has classes for this.

- First, add `class="title"` to the h1 heading, save and reload the page.

This tells bulma to style it as a title/heading element. On the [title element page](#) you can see that the correct way to get a top-level heading is with the classes `"title is-1"`, whereas a `"title is-2"` would be a second-level heading (slightly smaller font):

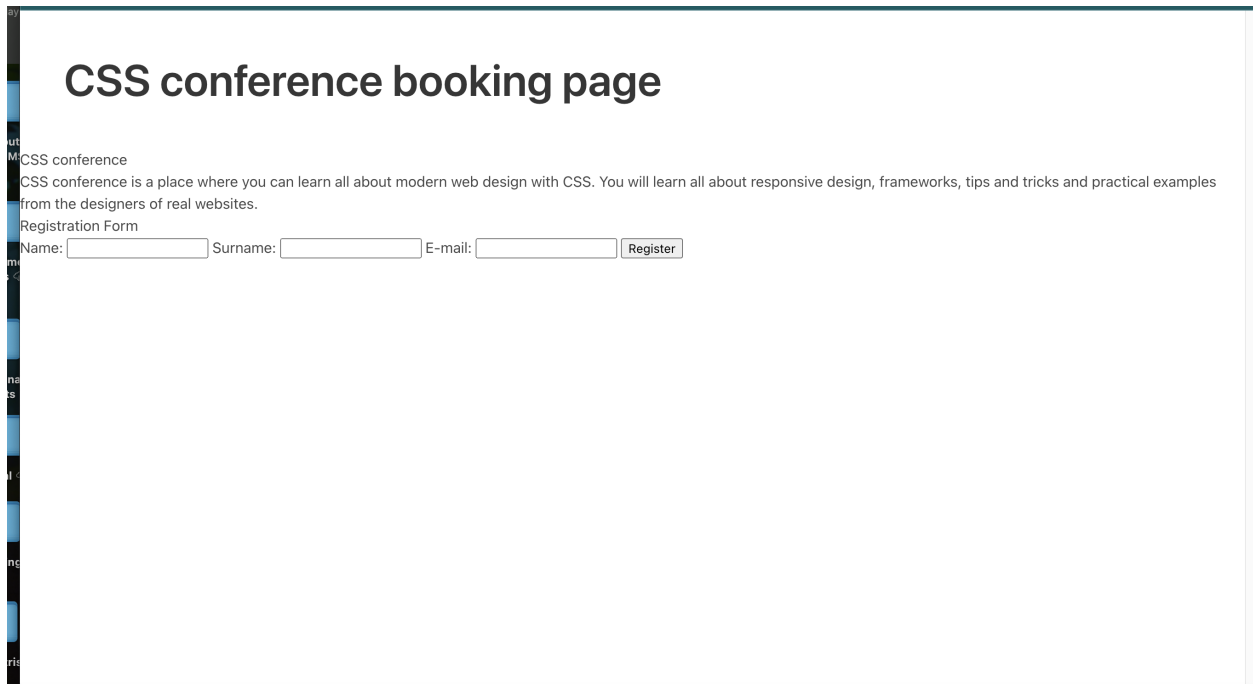


This looks like you are saying the same thing twice, and you are, but it lets you keep the structure and the styling separate - tag names give structure, class names give styling. Note that this method is a convention by the bulma designers, which other designers might not agree with, not part of the HTML/CSS standard. Now to the hero part. According to the bulma documentation for the [hero component](#), we need a container with class `hero` and a direct child with class `hero-body`, so

- Add class `hero` to the header tag.
- Add a child div `hero-body` inside it that wraps the `h1` tag.
- Set the classes

`title is-1` on the h1 tag itself.

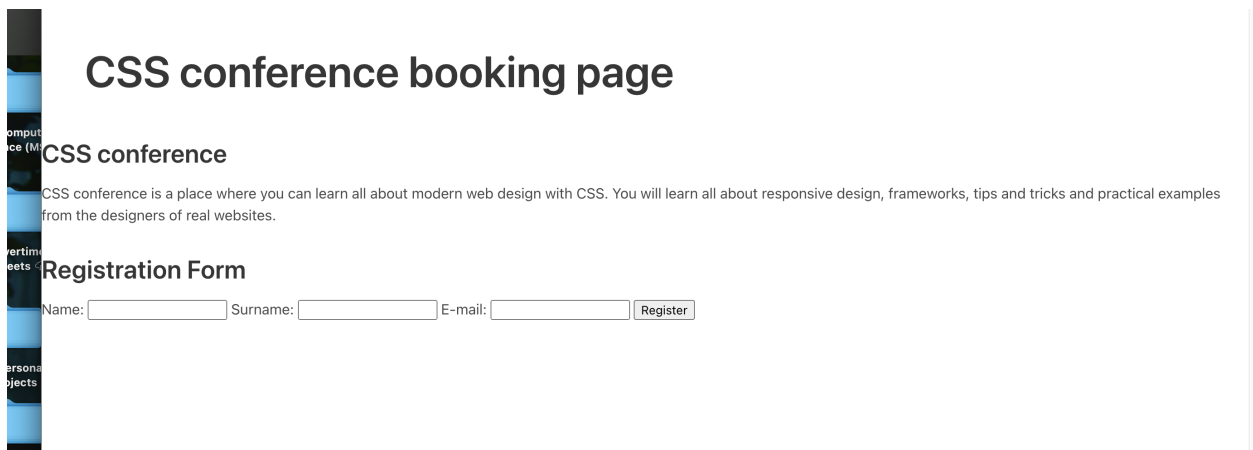
You now have an eye-catching element at the top of your page:



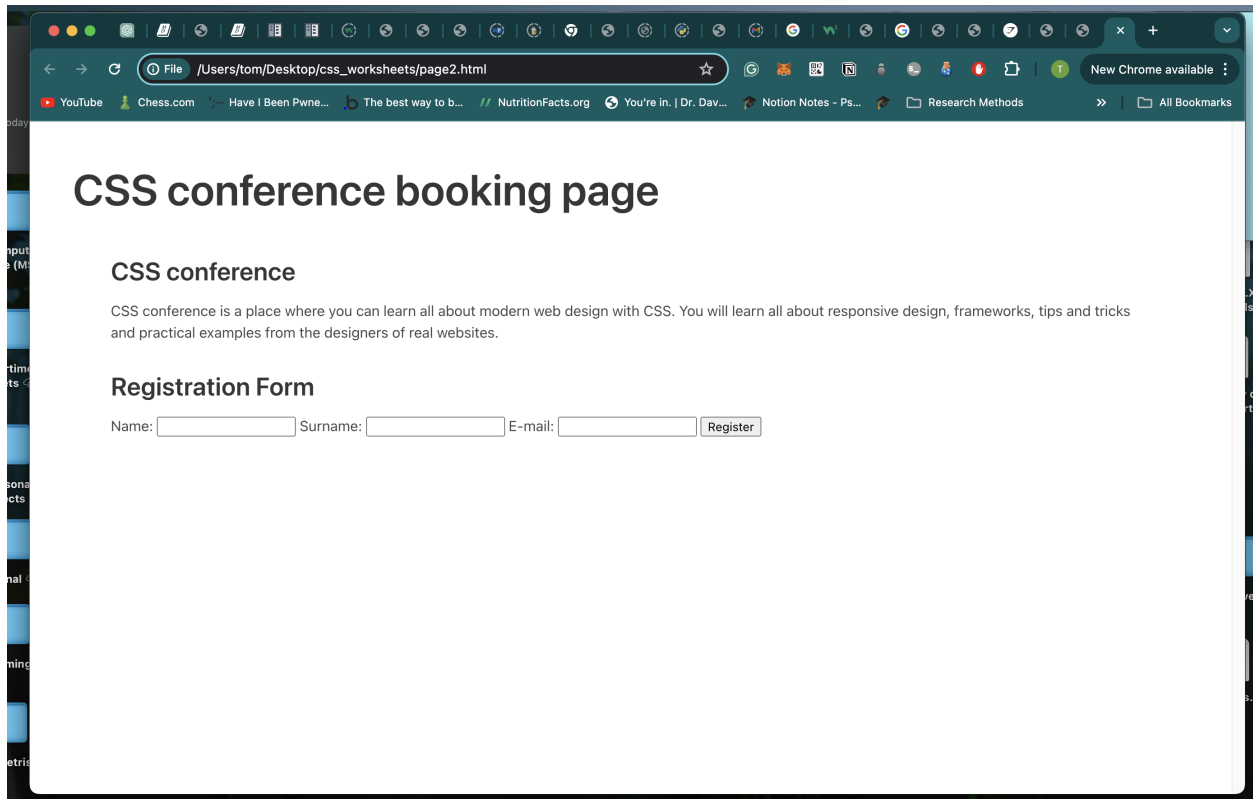
Bulma also lets you give an element the class `content` to tell it to style everything within that tag based on the tag names. Class `section` creates a section with spacing before/after, and class `container` makes something fixed width and centered.

- Add class

`content` on the `main` tag. Reload the page and observe the changes:

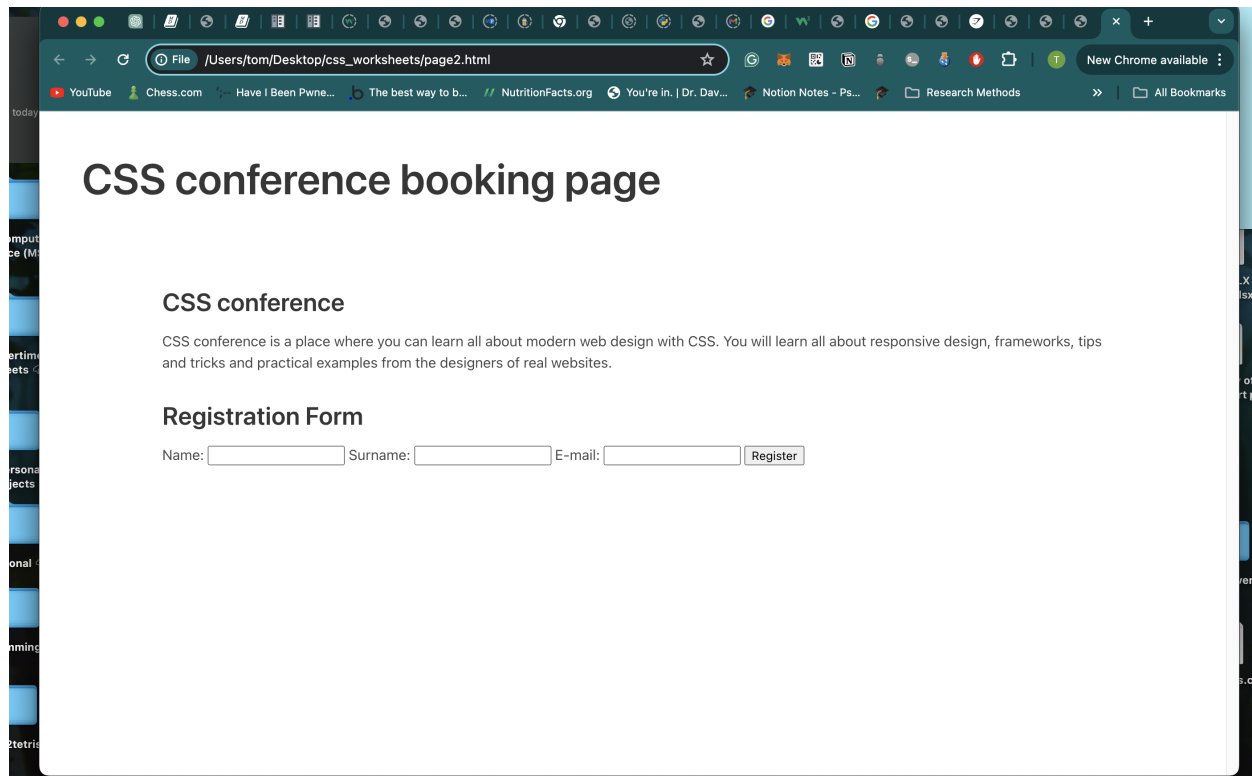


- Add a `div` inside main that wraps all the content inside, and give it class `container`. Reload again and observe the change:



- Notice that the first h2 heading text touches the bottom of the hero area, which is not good design - text needs space between it and edges of coloured areas. Create a section tag as the direct child of the container, and give it class `section` so your code should look like this:

```
<main class="content">
  <div class="container">
    <section class="section">
      ... content ...
    </section>
  </div>
</main>
```



For the next exercise, style the form.

- Put another section tag around the form and its heading (so the container tag will have two direct children, both of them sections).

- Using the bulma

[form documentation](#), add tags and classes to the form as necessary so it looks similar to the Milligram one (full width fields, labels immediately above the fields, coloured button with rounded corners). You don't have to make the button purple, use `is-primary` to make it blue.

Finally, add and style a

[bulma component](#) of your own choice on the page.

Bootstrap

The most popular CSS framework is

[bootstrap](#), developed by Twitter. If you are interested and have some spare time, you can have a look at it - it has lots of different components and options to customise the styling further with themes.

Bootstrap includes a JavaScript file as well as a stylesheet: for some components

like tabbed pages, you need the scripts to make clicking on a tab switch to the correct content.

Bootstrap

The most popular CSS framework is bootstrap, developed by Twitter. If you are interested and have some spare time, you can have a look at it - it has lots of different components and options to customise the styling further with themes.

Bootstrap includes a JavaScript file as well as a stylesheet: for some components like tabbed pages, you need the scripts to make clicking on a tab switch to the correct content.

Code from the above:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <title>CSS example</title>
    <meta name="viewport" content="width=device-width, init:
    <link rel="stylesheet" href="https://cdn.jsdelivr.net/nj
  </head>
  <body>
    <header class="hero">
```

```

        <div class="hero-body">
            <h1 class="title is-1">CSS conference booking page</h1>
        </div>

    </header>
    <main class="content">
        <div class="container">
            <section class="section">

                <h2 class="titles is-2">CSS conference</h2>
                <p>CSS conference is a place where you can learn</p>

                <h2>Registration Form</h2>
                <section>
                    <form method="POST">
                        <label for="name">Name:</label>
                        <input type="text" id="name" name="name">
                        <label for="surname">Surname:</label>
                        <input type="text" id="surname" name="surname">
                        <label for="email">E-mail:</label>
                        <input type="text" id="email" name="email">
                        <input type="submit" value="Register" />
                    </form>
                </section>

            </section>
        </div>

    </main>

</body>
</html>

```