

Static Files

The authentication views and templates work, but they look very plain right now. Some [CSS](#) can be added to add style to the HTML layout you constructed. The style won't change, so it's a *static* file rather than a template.

Flask automatically adds a `static` view that takes a path relative to the `flaskr/static` directory and serves it. The `base.html` template already has a link to the `style.css` file:

```
{{ url_for('static', filename='style.css') }}
```

Besides CSS, other types of static files might be files with JavaScript functions, or a logo image. They are all placed under the `flaskr/static` directory and referenced with `url_for('static', filename='...')`.

This tutorial isn't focused on how to write CSS, so you can just copy the following into the `flaskr/static/style.css` file:

`flaskr/static/style.css`

```
html { font-family: sans-serif; background: #eee; padding: 1rem; }
body { max-width: 960px; margin: 0 auto; background: white; }
h1 { font-family: serif; color: #377ba8; margin: 1rem 0; }
a { color: #377ba8; }
hr { border: none; border-top: 1px solid lightgray; }
nav { background: lightgray; display: flex; align-items: center; padding: 0 1rem; }
nav h1 { flex: auto; margin: 0; }
nav h1 a { text-decoration: none; padding: 0.25rem 0.5rem; }
nav ul { display: flex; list-style: none; margin: 0; padding: 0; }
nav ul li a, nav ul li span, header .action { display: block; padding: 0.5rem 1rem; }
.content { padding: 0 1rem 1rem; }
.content > header { border-bottom: 1px solid lightgray; display: flex; justify-content: space-between; align-items: center; padding: 0.5rem 1rem; }
.content > header h1 { flex: auto; margin: 0 0.25rem 0 0; }
.flash { margin: 1em 0; padding: 1em; background: #cae6f6; border: 1px solid #add8e6; }
.post > header { display: flex; align-items: flex-end; font-size: 0.85em; padding: 0.5rem 1rem; }
.post > header > div:first-of-type { flex: auto; }
.post > header h1 { font-size: 1.5em; margin-bottom: 0; }
.post .about { color: slategray; font-style: italic; }
.post .body { white-space: pre-line; }
.content:last-child { margin-bottom: 0; }
.content form { margin: 1em 0; display: flex; flex-direction: column; }
.content label { font-weight: bold; margin-bottom: 0.5em; }
.content input, .content textarea { margin-bottom: 1em; }
.content textarea { min-height: 12em; resize: vertical; }
```

```
input.danger { color: #cc2f2e; }  
input[type=submit] { align-self: start; min-width: 10em; }
```

You can find a less compact version of `style.css` in the [example code](#).

Go to <http://127.0.0.1:5000/auth/login> and the page should look like the screenshot below.



The screenshot shows a web browser window displaying the Flaskr application. The header is a grey bar with the text 'Flaskr' in blue, and two links, 'Register' and 'Log In', in blue. Below the header, the main content area is white. It starts with the title 'Log In' in blue. Underneath, there are two input fields: one for 'Username' and one for 'Password'. Both fields are empty. Below the password field is a button labeled 'Log In'.

You can read more about CSS from [Mozilla's documentation](#). If you change a static file, refresh the browser page. If the change doesn't show up, try clearing your browser's cache.

Continue to [Blog Blueprint](#).