

## Bonus: Multiple Entry Points

In the example project, we only have **one main entry point**: `app.js`.

In bigger projects - with multiple HTML pages - you **might have multiple scripts** for the different pages (HTML files) you might be building. Hence you **might need more than one entry point** because you want to build more than one bundle (i.e. not every HTML page uses the same script).

**This can easily be configured with Webpack:**

Instead of

```
entry: './src/app.js'
```

use

```
entry: {  
  welcome: './src/welcome-page/welcome.js',  
  about: './src/about-page/about.js',  
  // etc.  
}
```

Now Webpack will **look up all these entry points** and create one bundle per entry point - you can then link to these bundles in your respective HTML files.

**A simple rule that makes sense for most projects is:**

*One entry point per HTML file because you typically have one script per HTML file.*

If you share a script across multiple HTML files or you have a file that does not need any script, you of course can deviate from that rule.

You can **learn more about multiple entry points** with these two resources:

- Code Splitting (i.e. generating more than one bundle): <https://webpack.js.org/guides/code-splitting/>
- Entry Point Configuration: <https://webpack.js.org/concepts/#entry>

And in general, check out the official Webpack docs to dive into it in detail:

<https://webpack.js.org/guides/>