

# Skills tip for Lab 2

---

CORS ERRORS

# An annoying little thing called “cross-origin request sharing errors”

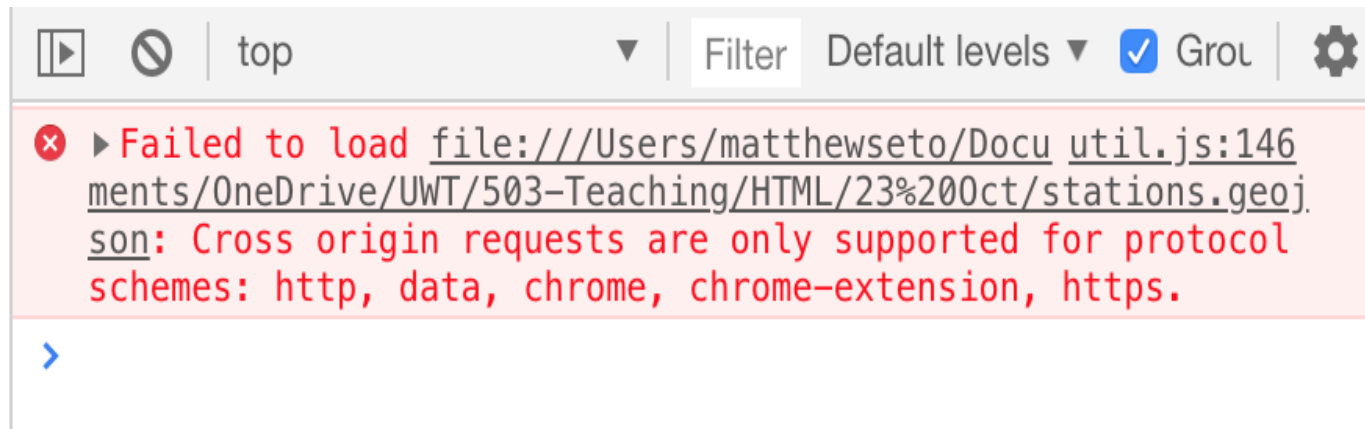
---

- There are security protocols built into HTTP that restrict documents from one origin from loading scripts from another origin
- The origin is basically the first part of a file url, e.g.:
  - http://
  - https://
  - file:///

# An annoying little thing called “cross-origin request sharing errors”

---

CORS protocols prevent a locally hosted file (origin = local:////) from loading JSON data, so we need to host the file on a web server (origin= http:// or https://) in order for geoJSON data to display



# Getting around this with Atom

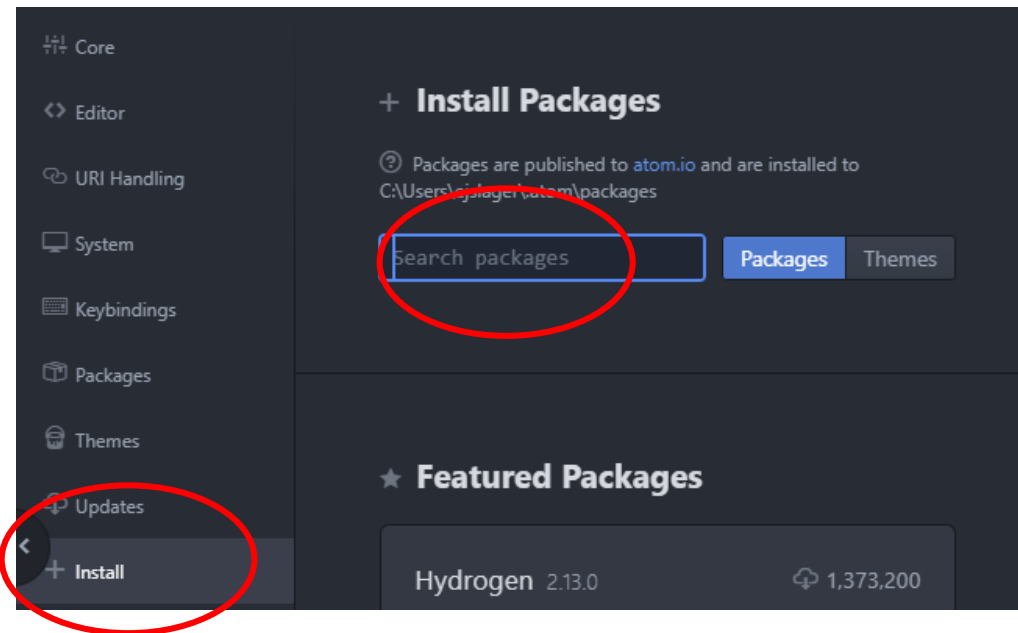
---

- When you upload your files to your web server (UW Server, GitHub Pages, etc.), this won't be a problem. But having to upload files in order to view changes really slows down the development process...
- We can use an Atom plugin package called atom-live-server that helps us get around this
- It launches a test server for you to view your pages on an http:// origin as you develop them

# Installing and using atom-live-server

---

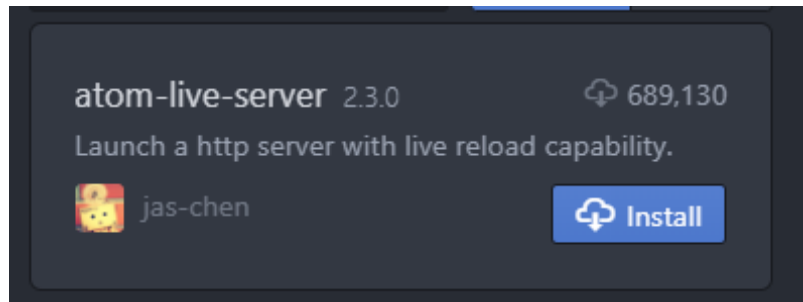
- Open Atom, click File > Settings
- Click the '+Install' button and enter atom-live-server in the 'search packages' bar



# Installing and using atom-live-server

---

Click the 'install' button for the atom-live-server developed by jas-chen (it should be the top result). The installation may take a moment.



Now, go to Packages > atom-live-server > start server

# Installing and using atom-live-server

- Your default web browser will open, and the live server will load the index page of whatever project is the top project in your Projects side bar. Open other pages using their path extensions (see screenshot).
- When you save, the page in the live server will reload automatically
- For help, see <https://webdesign.tutsplus.com/tutorials/quick-tip-setup-local-previews-with-atom--cms-24348>

