# Get a Web Server Running

You might be wondering why you need to run a web server just to load data in the browser. Well, because JavaScript runs in the browser, the way you load data, whether it's represented as XML, JSON, or CSV, into the browser for your application to use, is using the web server. Typically, you make a request for data using `XMLHttpRequest` or JavaScript's `fetch` API, and the web server responds with the data. You store that data in JavaScript objects, and use it in your program.
In this project, we'll be using D3's `d3.csv()` method to load the CSV data in the GLB.Ts.csv into the browser and make it accessible to our code. So, we need to make sure all the files for the application are served by a web server running on your local machine. Unfortunately, that means you'll have to do a bit of extra work here. The good news is, whether you're running Windows or Mac or Linux you have a few options.
Alternatively, if you don't want to run a web server on your own machine, if you have a hosted web site somewhere, you can always upload all the files and access the application from your hosted web site (which will automatically serve the sound files to the application). Most hosted web sites provide an easy way to upload files to a folder you can access at your domain name.

## Mac Users

**Beginner:** We recommend using Python. Python comes installed on your Mac already. To use Python to run a web server, you'll use the Terminal application. You can find this in your Applications/Utilities folder. Double click the icon to run Terminal.
Find the path to your Warming Stripes project code. For instance, if your code is in a Stripes folder inside your Sites folder, then `Sites/Stripes/` is your path.
At the prompt in the Terminal window type:

```
cd [PATH]
```

where you will substitute your path (e.g. Sites/Stripes/) for [PATH]. Type

```
ls
```

at the prompt and you should see your list of project files, including the data file GLB.Ts.csv. Then type:

```
python3 -m http.server
```

If this doesn't work, try

```
python -m SimpleHTTPServer
```

Leave Python running. This will run a web server at the default port (8000). Then you'll be able to access your project HTML file using the URL:

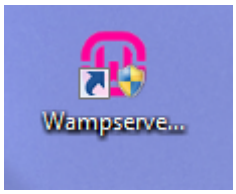
```
<a
href="http://localhost/stripes.html">http://localhost/stripes.html</
a>
```

If you haven't downloaded the project files from github yet, do that now so you can test your web server.

**Advanced:** If you're an advanced user, you can enable the Apache Web Server that's already installed on your Mac. If you're running Yosemite, try following these instructions; if you're running Sierra, try following these instructions. If you're running High Sierra, try following these instructions.

## Windows Users

**Beginner:** We recommend installing WampServer. Follow the instructions at the the WampServer web site to download and install WampServer on your version of Windows.
Once it's installed, you'll find you now have a new desktop icon to start the WampServer manager. Launch the WampServer Manager using the desktop icon.




You'll see a square go from red to green on your taskbar once WampServer is running successfully.

Click on the green square to see the WampServer menu.

Open the www directory, which is installed by WampServer. This is usually at C:\wamp\www, but might also be at C:\wamp64\www. Create a new "Stripes" folder in www and copy all the stripes files into that folder.

Open your browser and go to the URL:

```
<a
href="http://localhost/Stripes/stripes.html">http://localhost/Stripe
s/stripes.html</a>
```
Note that if you used a different folder name than "Stripes" you'll have to change the path above to reflect your folder name. If you haven't downloaded the project files from github yet, do that now so you can test your web server. Just make sure to put them in the correct folder.

You can use the WampServer menu to stop and start the server when you like. Just make sure it's running before you access your Stripes code at the localhost URL.

Another option for Windows users is to install Python, and run the simple server, just like we did on the Mac.

**Advanced:** You can install and run Apache in Windows to run a web server. Follow these instructions (just the Apache section, you don't need PHP or MySQL) or these instructions at Apache.org.

## Linux Users

Hopefully you Linux users know what you're doing and have a web server in mind or already installed.

If not, installing Python is a great option (just like it is on Mac and Windows; see above). Once you have Python installed, you can run the web server at the command line:

Python 2.x:

```
python -m SimpleHTTPServer
```
Python 3.x:

```
python -m http.server 8000
```
**Advanced:** Install and run Apache Web Server on your Linux system.