

# Professional Website Building

Damon C. Roberts

June, 2020

# What Are My Webhosting Options?

- Google Sites
- Wordpress
- Bluehost
- Github
- And More!
- For more information about the pros and cons of hosting options, see Andy Q. Philip's [github](#)

# **This Workshop is Oriented towards hosting on Github**

# Basics of Github

- Git is a command line tool that is useful for version control
  - Many people working in the tech industry and coding world use git to ensure that they can retrieve old versions of code in case there are problems
  - Github was created to make it less onerous on users and to be a bit more friendly than git. Github has a desktop app and a website where you can visualize your version controlled code.
  - Setting up a github account is really easy. Once you do so, you also need to install the github desktop application on your computer; it will not work well if you try to manually adjust code from the webversion or continuing to upload the files whenever you change something
  - The point of git is to easily update and control code.
  - Github allows you to not have to write the bash or powershell commands.
-

# Github pages

- Github pages is a way for you to easily manage and host a website
- It typically has a few start up costs, but it is a great signal to people on the job market about your technical capabilities. Additionally, you own your data - not some large tech company since github is open source!

# How to get started

- There are many ways you can go with this:
  - Build your own if you know CSS or HTML
  - Use any number of open-source jekyll themes (preferred so you can skip the laborous web dev coding - just find something you like, upload your own content and adjust when needed)
  - Use the Academic theme by Hugo (the tutorial in the next slide will follow this option - the Academic theme is really clean and is bootstrapped for cross-device compatibility.)

# Website building tutorial with Academic Theme

- You can find the theme's materials and documentation [by following this link] (<https://themes.gohugo.io/academic/>)
- You should follow the one-click install. This sets up a domain with netlify - which you can then purchase your own customized domain name later; very clean and straight forward.
- Once you have installed the theme and are set up on netlify, you then need to start changing and updating your content

# Updating and Changing Content

- The Academic theme is a markdown website. This means that rather than writing html and css code each time you create new content for your site, the theme has html and css templates so that when you write new content in a markdown file, you can just invoke the template - and that is it.



# Updating Continued

- The file tree for the theme has a couple tricks to it that make it challenging to navigate sometimes
  - `content/`: is where you will find your file tree to upload content to your site. This varies depending on what you have on your site; but it should all go within this master folder.
  - `static/`: this is where you should put your CV. Your CV should go in `static/files/` and should be titled `cv.pdf`

# Configuring your Menu

- `config/`: is where you will configure your menus and such \*In `config/_default/menus.toml`, you'll be able to customize your menu bar. You will simply have:

```
[[main]]
name = "Home"
url = "#about"
weight = "10"
```

- This bit of code tells the theme to add this to the main theme, name it Home and to use the #about information in your content/ folder to display it, and `weight = 10` tells it to put it first.
- You can also reference a saved file on your site, like your CV by using the following:

```
[[main]]
name = "CV"
url = "files/cv.pdf"
weight = 30
```

- This tells the theme to go to the `files` folder and to link to `cv.pdf`

# Configuring your menu continued

- If you are looking to reference something from a different site like an R shinyapp dashboard of your projects, this is all you have to do:

```
[[main]]  
name = "Project Dashboard"  
url = "https://damoncharlesroberts.shinyapps.io/Current_Projects/"  
weight = 50
```

- Notice that all you had to do was include the link to the page.

# Writing .md (Markdown) Files

- There are many resources to readily find this.
- The yml header information will be slightly different with this theme depending on what you are writing.
- The yml header at the top of your .md file will have some differences depending on the theme template you are using.
  - You can find examples and explanations on the documentation for the theme. You can also check out my website to see what I have done
- Keep in mind your pages will be your subfolders within your content/ folder. For example if you want to make a page dedicated to the courses you teach you will have a folder directory that looks like the following `content/courses/index.md`.
  - Notice how there is an `index.md` file, all of your pages should be titled `index.md`.

# Editing your content in your \*/index.md file

- As mentioned before, each yaml header will look slightly different depending on the template you are using for the theme. for a standard page, though, the yaml header will include:
  - A title: `title:`
  - A summary of the page: `summary:`
  - Tags to help connect similar pages that are listed on their individual lines with a - before them: `tags:`
  - For example `tags:`
    - Webdev
    - Statistics
    - Blog `
- There are some other things to include but to leave blank in your yaml header because of the theme's settings. You can find this information in their documentation or by looking at [this example] (<https://github.com/DamonCharlesRoberts/damonroberts/edit/master/content/project>) by clicking edit this file to see the code.

# Editing \*/index.md file

- To make a H1 Header, the largest header you can simply use the # symbol then a space and the text. For example:

```
# Title
```

- To make a subheader like an H2, you can simply do this

```
## H2 Header
```

- To list objects in a bulleted (unordered list): `
- Unordered list
- Unordered list `
- Another nifty thing about the theme's templates is that you can include mathematical equations by using latex:

```
$$ \hat{\beta} = (X'X)^{-1}X'Y $$
```

# Publishing the Site

- Using netlify, the site should be ready to deploy pretty immediately.
  - Since you have already set up the site with Netlify, you can just login to their site, click the website you just built, and click deploy. Now, whenever you push a commit on github, it will rebuild the site. You can check a deploy status on the Netlify site and will be told of an error if your code is wrong somewhere.
  - If you receive an error that doesn't push your update, there are many reasons for this and you should read the error message carefully.
  - Use Stack Overflow if you run into problems
    - Get used to it if you use anything but Stata
    - You will get faster at finding solutions as you get practice asking coding questions.
-

# The gist of all this:

- READ THE DOCUMENTATION CAREFULLY!
  - Almost everything you need to set up a github site is at your fingertips. The tricky thing is that there are a few things that are not explained well or are present at all. If you have questions, you can likely find a solution.
  - Look at demos, often times this shows you WORKING code. If you can get your hands on it and change things, you'll learn what it is doing
- Github pages do not have to be very technical - if you are not a coder, that is fine! You do not have to code.
  - Just be willing to have to spend some time to get this set up and learn your way around the file structure.
  - Once you have it up, you will have a beautiful site that doesn't cost you an arm and a leg to make or something that you can't customize