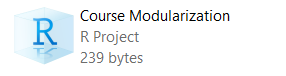
**Guide to building a website using R, GitHub, and Netlify**

This GitHub repo contains all course materials and files for building the website:

<https://github.com/DARTH-git/Course-Modularization>

To access these files, the repo has to be cloned to local computer.

After cloning the repo to a local folder, open the R project file to build and manage the website.



All course materials are stored in:

*Course-Modularization\static\Course\_Modularization*

You can add, remove, or edit the course materials in this directory directly. Make sure to keep an organized structure.

Modify the “config.toml” file to modify the name, tabs and Slack button/URL of the website. I would only make necessary changes and not touch other stuff to avoid breaking it.

Make sure to also update the Slack URL in the file, “doc\_toc\_foot”, each time there is a new course. The file is in: *Course-Modularization\layouts\partials*

To modify the content of the website, modify the folders and files in the “content” folder. Ignore the “post” folder.

Make sure the R package “blogdown” is installed.

After making changes to a .Rmd file in the “content” folder, run the following command:

blogdown:::build\_rmds("path to .Rmd file")

Examples:

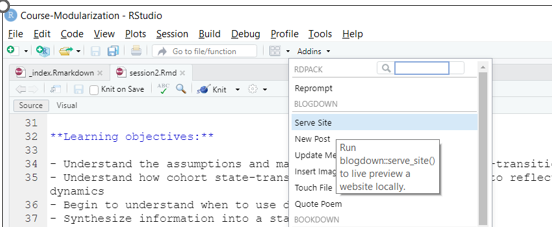
blogdown:::build\_rmds("content/days/day0/session1.Rmd")

blogdown:::build\_rmds("content/days/day1/\_index.Rmarkdown")

If you don’t run the above code, the changes won’t apply.

To ensure the links to the course files work, make sure the file path and extension are correct. Look at existing code/ file path structure for reference.

To update the website after making changes, do the following:



Each time after running “Serve Site”, I recommend typing the following command in the console if you want to make more changes (or R becomes really slow from trying to update your changes real-time):

blogdown::stop\_server()

I recommend running “Serve Site” and the above code after a series of changes have been made.

Aside from the folders and files mentioned above, I would leave everything else alone to avoid breaking it.

To put the website onto Netlify, first create a Netlify account and link it to your GitHub account. I recommend logging into Netlify with your GitHub account directly.

<https://www.netlify.com/>

Then, click on “Add new site” and follow the instructions. The instructions are easy to follow.