Website building

Contents

This page was last updated Fri Jan 03.
[Download this page as a PDF]

Learning Objectives

- Recognize the minimal requirements to build a website using R and Markdown.
- Build a basic level website using R and Markdown
- Publish this website to GitHub
- Become aware of how the `blogdown` R package can be used to create responsive websites
- Download and compile a starter website based on blogdown.

Getting started: Basic Site

R Markdown websites like this one can be built with a minimum of two files: a `_site.yml` and an `index.Rmd`.

The `_site.yml` file controls the overall structure of the website, such as the order of the navigation bar and the color theme.

When these files are *rendered* into HTML files, they can be hosted on GitHub for free. Then after turning on a few settings in your GitHub repository (repo) these files can be viewed as a magnificent webpage.

We are going to employ the "GitHub first, R Studio second" approach.

1. Create a public Github repo.

- From your *Repository* page, click the green [NEW] button.

- (*Suggestion*): Name this repo something like `ichps_website` so you know it's a playground.
- keep it a PUBLIC repo
- YES initalize with a README

Click 'Create Repository' to finish, but keep this page open. We'll come back to it a few times.

2. Create a project in R Studio and clone your new repo into this project.

- Top right corner in R Studio click: *New Project –> Checkout from version control –> Git*
- Go back to GitHub, and click on the green [Clone or Download] button.
    - Click the clipboard icon to copy the URL to your repo.
    - Paste this URL into the *Repository URL* box in R Studio.
        * (***Strongly Suggestested***): Name this project the same as the repo name.
        * Subdirectory: Your path, your choice. Just don't put a repo inside a repo.
- Click *Create Project*

3. Create necessary files in R Studio.

- In the *Terminal* window (tab to the right of the *Console*), create the `_site.yml` and `index.Rmd` files using the `touch` command. Type the following commands in the Terminal one at a time, pressing Return or Enter after each line to execute.

```
touch _site.yml
touch index.rmd
```

You can view these files in the *File* pane (lower left). You can open supported file types by clicking on the name of the file from this pane.

- Open both the `_site.yml` and `index.Rmd` files now.

4. Configuring the `_site.yml` file.

This is the file that controls your overall site's configuration. If you are familiar with R Markdown files, this is the same type of 'language' that is used in the header area to control the document structure.

YAML headers, and this file, are very particular about spacing and tabs. To avoid unnecessary grief we are going to copy/paste code from Chapter 10.5.1 of the bookdown website.

- Navigate to the bookdown website by clicking the link above.
- Mouse over the first code chunk in section 10.5.1, and click the *Copy to Clipboard* icon in the top right corner.
- Paste this code into your `_site.yml` file.
- Remove lines 7 and 8 since right now we don't have an "About" page.
- Give your website a name and title!
- Add the following line on a new line `2` (between `name` and `navbar`).
    - This ensures that all rendered HTML files are saved in the top level directory.

```
output_dir: "."
```

5. Configure your index page.

This is your landing page. That is, the first page people see when they go to your website. You can add a title to this page using a YAML header as is shown in the bookdown website, or you can use a pound sign `#` as a first level header to make this title.

- Create a title for your page using *either* a YAML header or a `#` sign. Using both is redundant.
- Write a sentence or two welcoming your viewers to your website.

6. Building & previewing your website locally.

Since we created the `.yml` file manually after creating the R Project, your project doesn't quite know yet that it has a website to build.

- Save your work and close down R studio.
- Navigate to your repository folder on your computer, and open the project by double clicking on the R project file icon (a cube with the letter R).
  - *Note: This is always how you should open your R Project files*

There should now be a **BUILD** tab in the top right pane.

- Click this tab, and then click the *Build Website* button.

An HTML file should now appear in your viewer pane.

- Click the 'Show in new window' button to see your site in a full browser window.

Notes, Tips and Comments -

- Clicking "Build Website" will render every `.md` and `.Rmd` file in your top level project folder into an HTML file.
  - This is important to keep in mind when creating a website with a lot of pages.
  - Code files in sub-folders will not be automatically rendered.
- You can re-render individual files by clicking 'Knit'.
- You can leave this browser window open as you work, refreshing the page to see new changes.
- There are some basic themes you can app

```
output:
  html_document:
    theme: readable
    highlight: pygments
```

Try this now: Make a change to your `index.Rmd`, knit and refresh your browser window.

7. Push to GitHub & publish your website.

Let's get your test site up for the world to see.

- In the *Terminal* window, stage and commit your files to version control by executing the following commands:

```
git add -A
```

This will *stage* all files that have been changed and/or added. *Staging* is the process of adding a file to be tracked under version control.

```
git commit -m "first commit"
```

Every commit needs a message. Try to make it informative, yet brief.

```
git push
```

This will *push* your changes up to GitHub's servers online. You may have to enter your github username and password here. See happy git with R for help storing your github credentials.

- Go back to your GitHub repo page and refresh - you should now see that your repo now contains the new code, and HTML files that you have been working on.
- Click *Settings* in the top right of the navigation bar in your repository.
  - Scroll down to the **GitHub Pages** section.

– Under *Source*, select *Master branch*. (*Note: Chapter 3.3 of the bookdown manual has instructions for publishing something other than the top level folder for a repo, such as a `public` folder*)

After the page refreshes, you will see the URL to your website in at the top of GitHub pages section.

- Click on this link to see what your public facing website looks like!
- Copy this URL, we're going to paste it in the repo settings so it can be found easy.
- Go back to your main repo page by clicking on the name of your repo at the top of the GitHub page.
  – Click **Edit**
  – Add a short description, and paste the URL into the *Website* field.

8. Adding Content.

Let's add an example analysis project to showcase on our website.

- Go back to R Studio.
  – Create a new R Markdown file using *File -> New R Markdown* so we can get the example R Markdown content that includes a plot.
  – Save this file with a name such as `project.Rmd`
  – Knit this file to produce a HTML file.
- Add a link to the project HTML page to your `_site.yml` configuration file by adding a new `text` line. Be mindful about the spacing here.

```
  - text: "My Projects"
    href: project.html
```

- Rebuild your website by clicking the *Build Website* button in the *Build* tab in the top right pane.
- View your changes locally, make a change if you see fit.
- Add your changes to version control from the Terminal:

```
git add -A
git commit -m "add project"
git push
```

- Refresh your live webpage on GitHub to see the results!

Next phase: Blog Aware

In Progress

References

- Github
  – https://www.atlassian.com/git/tutorials/learn-git-with-bitbucket-cloud
  – https://swcarpentry.github.io/git-novice/
  – https://happygitwithr.com/
- Static Sites
  – R Markdown sites from R Studio https://rmarkdown.rstudio.com/lesson-13.html
  – Chapter 10.5 in the Blogdown book https://bookdown.org/yihui/rmarkdown/rmarkdown-site.html
- Blog aware sites

– `blogdown`: Creating Websites with R Markdown https://bookdown.org/yihui/blogdown/ , https://github.com/rstudio/blogdown