

ISSC: Building a personal website workshop series

Liza Bolton

Last updated 2021-05-25

Contents

1	Introduction	2
1.1	What we're going to do during the workshop series	2
I	Prerequisites	2
2	What you need to do before the first session	2
2.1	Decide if this is the right way of making a website for you	2
2.2	Technical tasks	2
II	Workshop sessions	3
3	Session 1: Build a simple website with Distill	3
3.1	To do before the workshop	3
3.2	Workshop instructions	4
3.3	Other tips	5
3.4	Setting up a custom domain	5
3.5	Post-workshop tasks	5
4	Session 2: Blogging with Distill	5
5	Session 3: Content development	5
A	Buy a domain name (optional)	5
A.1	But what if you don't buy a domain name?	5

Unless told otherwise, assume everything in this guide is as yet incomplete!

1 Introduction

There are many, many ways to make a website these days. This workshop series aims to help you make a personal website using the `distill` package. This is just ONE way that is used by a lots of folks in statistics and that leverages your existing R and R Markdown skills. It is reasonably beginner friendly and creates a professional looking website.

Note: If you already have a personal website, sessions 2 and/or 3 may still be of interest to you. You can attend as many or as few of these sessions as you would like.

1.1 What we're going to do during the workshop series

- Get set up with R/RStudio/Git and GitHub (if not already)
- Build a basic landing page website with `postcards`
- Create a multi-paged website with `distill`
- Begin to develop an aesthetic for your personal brand
- Learn some basic ways to edit the look and feel of a site with CSS
- Set up and post a first blog post
- Develop and polish professional content for your website and blog
- Connect to additional U of T resources that can help you develop your professional digital presence (TBC)

Part I

Prerequisites

2 What you need to do before the first session

2.1 Decide if this is the right way of making a website for you

In order to make the most of this series, you should think about what the purpose for your personal website is going to be.

- Is it to have a basic hub for your professional links? (A bit like the 'linktree' links you might see on Instagram.)
- Is it to share your technical portfolio and/or demonstrate your communication skills with simple blogs? Especially good if R is one of the main languages you'll be using. Also works with Python with the `reticulate` package.

If yes, to the above, this workshop series using `distill` is probably perfect for you. If you want a more complicated and customizable site checkout `blogdown` (still using an R package) or Bootstrap (HTML/CSS) or if you need to support e-commerce, you might just want to go find which YouTuber you like has a Squarespace sponsorship this week...

2.2 Technical tasks

To be able to engage fully with this series of workshops, you will need to have completed the steps in the Installation part of <https://happygitwithr.com/>. This a great a resource. I also **STRONGLY** recommend that you try to follow the instructions in Chapter 10 so you can easily set up your new project and connect it to GitHub.

- Register a GitHub account
- Get GitHub for education <https://education.github.com/> so you can have free private repositories
- Install or upgrade R and RStudio on your local machine
- Install Git
- Introduce yourself to Git
- Cache your credentials

For the easiest experience with website building and updating, you want to be able to push to a GitHub repo from either the Git pane in RStudio or using the terminal (which you can also access from RStudio, tab next to the console pane).

2.2.1 What is GitHub and why do you want it?

We’re suggesting that if you haven’t already, you get yourself setup with GitHub as a key component of your portfolio building strategy, regardless of your plans to make a website with this series or otherwise. It will supercharge your version control and your ability to collaborate with others AND provides a FREE way to host a website.

Jenny Bryan has a great introduction in her Happy Git with R, so I’ll let her explain the rest:

Git is a version control system. Its original purpose was to help groups of developers work collaboratively on big software projects. Git manages the evolution of a set of files – called a repository – in a sane, highly structured way. If you have no idea what I’m talking about, think of it as the “Track Changes” features from Microsoft Word on steroids. Git has been re-purposed by the data science community. In addition to using it for source code, we use it to manage the motley collection of files that make up typical data analytical projects, which often consist of data, figures, reports, and, yes, source code. A solo data analyst, working on a single computer, will benefit from adopting version control. But not nearly enough to justify the pain of installation and workflow upheaval. There are much easier ways to get versioned back ups of your files, if that’s all you’re worried about. In my opinion, for new users, the pros of Git only outweigh the cons when you factor in the overhead of communicating and collaborating with other people. Who among us does not need to do that? Your life is much easier if this is baked into your workflow, as opposed to being a separate process that you dread or neglect.” - Jenny Bryan, Happy Git with R, Section 1.1: Why Git? <https://happygitwithr.com/big-picture.html> There is also lots of great practical professional advice in here, too, like “Pick a username you will be comfortable revealing to your future boss.” Save gamerangel420 for Reddit. (The first example I thought of I had to change...it actually was someone’s Reddit username) Work through Jenny Bryan’s awesome ‘Happy Git with R’ Check out the DoSS Toolkit lesson “Git outta here” Sign up for GitHub Education Student Developer Pack (access to hundreds of dollars worth of tools and training AND a GitHub pro account while you’re a student)

Part II

Workshop sessions

3 Session 1: Build a simple website with Distill

3.1 To do before the workshop

- Complete all the steps in the prerequisites section.
- Decide if you want to buy a domain name (see appendix). Buying a domain name is **totally optional**, and you don’t have to decide now. But, if you know you want to, you might as well get set up now.

- Decide what links to external things you want to share on your website. Have them easily accessible to copy and paste. *GitHub? LinkedIn? Twitter?*
- Choose a profile picture for your website. A headshot with a simple background is best. If you don't have one/don't want to show a photo, you may want to find some other form of an avatar or placeholder picture. Have the PNG or JPEG easily available to copy to the file you'll need.
- Download the R Markdown cheat sheet and pay special attention to the '**Pandoc's Markdown**' section on the lefthand side of the second page.
- Install the below R packages. I.e., copy and paste the below code into your R console and run it. If you're not sure what this means or how to do it, come talk to use during a TidyTuesday and Talk session (Tuesdays, from 2:00 to 3:00 p.m. ET, on Zoom, see the Events list)

```
install.packages('postcards')
install.packages('distill')
```

3.1.1 Thank you, Rohan!

Major credit to Prof. Rohan Alexander for his teaching notes about making a website. You can find them at: <https://www.tellingstorieswithdata.com/interactive-communication.html#making-a-website>

His personal website is also made with distill, <https://rohanalexander.com/>.

3.2 Workshop instructions

3.2.1 Landing page with postcards

Let's start with the most basic thing we can do, create a landing page with a brief bio and some key links. It is a bit like a business card for the digital age.

This page can later become the home page of your website, or you might find you want to do something more customized.

1. Make sure you have installed packages you needed to.
2. Create a new R project using the postcard template:
 - a. 'File -> New Project -> New Directory -> Postcards Website.'
 - b. Choose a sensible location for this project to live on your computer.
 - c. Pick a theme, I'd recommend trying 'trestles' for now, but you can see all the different 'looks' here (scroll down).
 - d. Choose 'Open in new session', just to be safe.
3. Run this code in your console: `file.create('.nojekyll')`. It tell GitHub something important about how *not* to build the site.
4. Navigate to the **Build** tab in the environment pane (usually the top right pane in your R Studio if you haven't customized the layout) and click **Build Website**. This should create a pop-up that allows you to preview what the basic template builds.
- 5.

3.2.1.1 Example I made the ISSC a landing page. You can view it at <https://uoft-doss-issc.github.io/> and the code is available on GitHub.

3.2.2 Multipage site

3.3 Other tips

- Add `.DS_Store` to your `.gitignore` file (if you're on a Mac)

3.4 Setting up a custom domain

It can take a while (like, a day or so) to update the redirection between your GitHub page and your domain, so be prepared to be patient!

These instructions are quite good: <https://richpauloo.github.io/2019-11-17-Linking-a-Custom-Domain-to-Github-Pages/>

If you're using `domain.com`

3.5 Post-workshop tasks

- Check-in with yourself:
- What is

4 Session 2: Blogging with Distill

More information coming soon.

5 Session 3: Content development

More information coming soon.

A Buy a domain name (optional)

You **do not have to buy anything** to build and host a professional personal website as a student. BUT, if you've always wanted to own `www.firstnamelastname.com`, here are some instructions. Note, if you're not sure what a domain name is there are heaps of explainer articles online, just search "what is a domain name". You don't really need to know too much.

A.0.1 Some caveats before we continue

- I'd recommend doing some of your own further research to make sure this is actually the best approach for you and what you actually want or need.
- This isn't legal or financial advice. Duh.

A.1 But what if you don't buy a domain name?

If you don't want to buy a domain name, your personal website's address will appear as: `your-github-username.github.io`. Even more reason to choose a good username! See more in the prerequisite section.

For example, I made a simple landing page for the ISSC at <https://uoft-doss-issc.github.io/>.