R-Ladies Baltimore | Building websites in Rstudio <https://www.youtube.com/watch?v=RYf5HqU1pI4&t=1s>

<https://www.tes.com/resources/search/?&q=R%20markdown>

<https://greenoak.bham.sch.uk/r-girls-school-network/>

Before getting started

* Go to google fonts and remove previously selected fonts
* Have r girls github open – show the website and show the code
* Have images folder on desktop so it’s easy to show what’s in it but then can simply just move into the working directory

Hi everyone and welcome back to building websites in R with distill! In today’s tutorial we are going to jump straight into building our website in RStudio and getting everything set up with Github.

1. Create github account
   * Rgirls
2. Sign in to github account
3. Create repo – important note: the name will end up being part of the URL of your website
   * repo name
   * public
   * add a readme file
   * create repo
4. clone repo
   * green code button 🡪 copy URL
   * open up Rstudio
   * project 🡪 new project 🡪 version control 🡪 git 🡪 paste
   * browse to save your project in an appropriate place
   * create project
   * right now this is cloning an empty repo (only has the readme.md file but by setting up the project this way allows us to interact with github and save all of our changes and eventually host our website on github
   * please don’t worry if you’ve never used github before, I’ll walk you through all of the steps along the way!
   * Close out of rstudio and open back up from desktop

Create Distill site

1. install.packages(“distill”)
2. library(distill)
3. create\_website(dir = “.”, title = “R Girls Website”, gh\_pages = TRUE) – this is a function from distill
   * dir “.” = current working directory which we’ve already set up
   * gh\_pages = we’ll host our website on github pages
4. this will create your website! May take a minute or two to fully create
5. go to Files to see everything that’s been created
   * site = configuration for the site; putting all the pieces together
     1. output\_dir = “docs”; docs folder is important because github will take all the files located here to create/build the website (I don’t recommend changing things in this folder)
     2. navbar will adjust this
   * index = default homepage for website (you always need to have an index file)
   * about
6. build website
   * you may have to install some packages the first time you try to build
   * you will likely have to restart your R session or even close out and re-open R
   * once you reopen your project, you should see a build tab in the top right
   * click build website and let it run
   * open in new tab to see the navbar and fullscreen of site
   * we’ll build the website pretty frequently to check on our changes and make sure there are no issues

BUILD SITE!

Customize website

1. google github seankross postcards: for different templates: <https://github.com/seankross/postcards>
2. delete index file because we’re going to create a new homepage!
   * If you try to create a new index file when one is already there, you will get an error message so make sure to delete the index file first
   * **May have to install postcards package**
3. create\_article(file = “index”, template = “trestles”, package = “postcards”)
   * we have to add one line of code to the yaml:
     + Text

       Description automatically generatedsite: distill::distill\_website (after output)
   * this creates the template for the homepage
   * This created a index.Rmd file
   * Now this is our new homepage
   * Change title and knit to see a preview of just the homepage

Push to github

create personal github token – error when pushing? Says something like don’t have access or authorization – chances are you’ll need to create what’s called a personal access token

1. In R: usethis::create\_github\_token() (will be redirected to github)
   1. log in to github
   2. provide name (optional)
   3. can change settings (but I keep the default)
   4. select create token
2. copy token
3. in R: gitcreds::gitcreds\_set()
4. paste token

Push to Github

1. <https://happygitwithr.com/rstudio-git-github.html>
2. In Rstudio
   * Install.packages(“usethis”)
   * usethis::create\_github\_token()
     1. log in to github
     2. provide name (optional)
     3. can change settings (but I keep the default)
     4. select create token
     5. copy token
   * gitcreds::gitcreds\_set()
   * copy token from github and paste it when prompted
3. push everything to github
   * stage, commit, push
4. stage = go to the Git tab and check all the boxes
   * Tools shell
   * Git add -A (to add everything)
   * Close out and click commit and everything will be staged
5. commit = click the commit button and you’ll see everything is checked. Here you have to leave a message! This is just a reminder of the changes you’ve just made. Then click commit
6. push to github!
7. Go to your repo on github to see all of your files
8. In your repo, click on settings 🡪 pages (on left)
   * Select “main” branch
   * Select the “/docs” folder
   * May take some time for github to build the website – can try to refresh

Alright so this looks great! Well done if you’ve made it this far and congrats on your new website! In the next tutorial we will modify our homepage so stay tuned! Thanks so much for watching and see you next time.

END TUTORIAL 2 HERE

1. Updating homepage
   * Can test out by knitting as we go
   * Title: R Girls Website - Tutorial
   * Image: images/rgirls-logo.jpg (if you have issues here, try saving as a different file type for example if .png doesn’t work, try .jpg)
     + Save r girls logo into a new folder (images in WD)
   * Copy and paste links from r girls index.rmd file
   * Start with twitter and Github basics
     + Twitter and url <https://twitter.com/R_Girls_School>
     + Github and url <https://github.com/R-Girls>
   * But I wanted to add the icons!
   * <https://www.pipinghotdata.com/>
   * Check out her github 🡪 pipinghotdata\_distill repo 🡪 index file 🡪 copy and paste the label for twitter
   * Add twitter URL
   * Knit doc and test it out
   * target="\_blank" after URL to open in new window
   * copy and paste the rest from the r girls index file
   * Couldn’t figure out how to do this on my own so I turned to google and looked at other distill websites
     + Twitter: <https://twitter.com/R_Girls_School>
     + Github: https://github.com/R-Girls
     + Email: [rgirlsschoolnetwork@gmail.com](mailto:rgirlsschoolnetwork@gmail.com)
     + Slack
   * Add content to rmd – copy and paste from r girls index.Rmd file

Text

Description automatically generated

END TUTORIAL 3 HERE

Hi everyone and welcome back to building websites in R with distill! In today’s tutorial we are going create a custom color theme for our website! This is one of my favorite parts of creating websites, but I will warn you, you can easily spend hours going through and selecting different fonts and different color palettes. Okay, so to start let’s open up our project and build our website to see what it currently looks like.

Create a new theme

* Library(distill)
* Create\_theme(“rgirls-theme”)
  + Site.yml 🡪 add line of code after outpit\_dir:
    - theme: rgirls-theme.css
  + First thing we’ll do is change our font
  + Import google fonts right at the top
    - <https://fonts.google.com/>
    - Note for me: remove previously selected fonts
    - Search open sans 🡪 regular 400, select this style 🡪 use on the web select @import 🡪 copy the code in between the style tags
    - Remove selected fonts
    - Search Amaranth 🡪 select this style 🡪 use on the web select @import 🡪 copy code (refresh if you don’t see the code)
  + Font sizes
    - Title size = 40px
    - Body size = 18px
    - Code size = 16
  + Choose color palette
    - <https://coolors.co/>
    - Make a palette
    - Can lock colors or press space bar for new colors
    - Export to pdf (HEX and RGB)
    - Open up rgirls color palette
  + Main font colors all = #01161E
  + Specify custom fonts
    - Heading font = Amaranth;
    - Mono font = monospace;
    - Body font = “Open Sans” needs quotes
    - Text

      Description automatically generatedNavbar font = Amaranth
  + Keep everything the same until .distill-site-header
    - Title size = 28px
    - Text color = stays the same
    - Text size = 28px
    - Hover color = white
    - Background color = #C34A79
  + Copy and paste from R girls website code…
    - Customize button colors
    - Add url color and hover color
  + Site.yml – title Home

END TUTORIAL 4 HERE

Hi everyone and welcome back to building websites in R with distill! In today’s tutorial we are going to modify our navigation bar and then build the Getting Started page. Let’s start off by checking out the R Girls website.

Nav bar setup

1. Library(distill)
2. Site.yml
3. Comment out the home tab because the link on the left is the home button
4. The menu argument allows you to add dropdown options so the main button will say lesson plans and then you can have the choice to pick which type of lesson plan
5. Make sure the spaces are properly formatted or you’ll get an error (even one tab off on menu as example)
6. Note: the links won’t work yet because we haven’t created the pages yet

Text

Description automatically generated

BUILD SITE!

Build site with error in YAML

Create Getting Started page – this is one page or we can even think of it as one rmd file so here we’ll use the create\_article() function

1. Create\_article(“getting\_started”) – will create an .rmd file in the root of the directory
   1. This should match up with the name of the href but don’t worry about the extension yet
   2. This automatically creates an rmd file
2. Title: “Resources to help you begin your R coding journey”
3. Comment out everything else in the yaml besides the date and output (ctrl + shift + C)
4. Copy text from r girls website getting\_started.Rmd
   1. ## My first R lessons using R Markdown – delete the links because it won’t work
5. Important: go to site.yml and make sure the href matches up with the name of the rmd file… in this case getting\_started. Even though the file we edited is .rmd when you build the website it will automatically create a .html file which is what you want to reference here
6. **Before build – go to docs there is no getting\_started.html**

BUILD SITE!

1. Now if you go to docs, you’ll see there is a getting\_started.html file!

Add Table of Content – rgirls website getting\_started.rmd

1. Go to getting\_started.Rmd and add a few lines to the YAML

Change colors – go to rgirls-theme.css

1. d-article h1
2. d-article a
3. d-article li

Embed Youtube videos

1. knit getting started page
2. click go to Youtube
3. click the share button
4. click embed
5. click copy

END TUTORIAL 5 HERE

Hi everyone and welcome back to building websites in R with distill! In today’s tutorial we are going create a couple of the lesson plans from the R Girls website. So let’s start off by building our website and seeing where we left off. In the last tutorial we built the Getting started page, which as a reminder we used the create\_article function from distill. Now in today’s tutorial we’ll create a few lesson plans. The first one is called My 1st R Markdown lesson and then we’ll also create a couple of Math Lessons. Let’s take a look at the R Girls website to see what these lessons look like.

We can see that My 1st R Markdown lesson is a single lesson, so when we click on it we are taken directly to the lesson. However, when we click on Math Lessons we are taken to a new page where we can see a list of all the different math lessons. In distill, each lesson can be thought of as a unique post. Once you click on math lessons, you still need to choose the lesson plan you want to view. Then, you are taken to the specific lesson and once again I’ll note that this is created with an Rmd file.

Alright, so let’s start off with creating My 1st R Markdown lesson because it’s a bit simpler and actually uses the same create\_article function we used in the previous tutorial.

Create R Markdown Lesson 1

1. create\_artice(“rmarkdown1”) – of course you can call this whatever you want but you will eventually need to make sure the .html file aligns with what you have written in your site.yml file.
   1. Rmarkdown1.html
   2. Title: **Introduction to R Markdown**
   3. Description: **An introduction to R using R Markdown aimed at secondary school students (age 11-16) and for complete novices. A complete lesson plan to introduce the R coding language using R Markdown to deliver instructions.**
   4. Remove author
   5. Keep date
   6. ~~Change the output YML~~

Text

Description automatically generated

* 1. Knit file

1. Add actual content
   1. copy and paste all the content from Github
   2. go to RGirls website and show a reminder of the links to download files
   3. outside scope of these tutorials but I will give a very brief overview of this code
2. walk through the code at the top – may have to install here package
   1. copy and paste whole rmarkdown1 folder into WD
3. one more reminder to make sure your href aligns with the name of your Rmd file
4. Build your website – if you don’t see your changes, try to knit your .rmd file first and then build the website again and hopefully that will work.

Create Math Lesson

1. Okay, so I’m sure there are several different ways to go through and create posts but I’ll show you the workflow that has worked for me.
2. Create an .Rmd file first that will live in the current working directory
   1. file new .Rmd title – Math Lessons
   2. save as “math\_lessons.Rmd” – must match the href in your site.yml
   3. go to R Girls website and show the different lessons. For each category of posts, you’ll need to create a unique Rmd file. So for example, for the R girls website, we would need an Rmd file for math, science and geography. In this tutorial I’m using math as the primary example, but you can follow the same steps for the others
   4. can delete the output in the yaml
   5. add “listing: math”
   6. Graphical user interface, text

      Description automatically generatedDelete all content so only yaml
3. This is just part of the initial set up to create the different posts we want. We won’t actually put any other code or content in these Rmd files.
4. go to site.yml and notice that once again the href needs to match the title of the rmd files we just created

Create Math posts

1. create\_post(“boxplots\_post”, collection = “math”) – ~~needs to be different from “boxplot” because we’ll upload the rmd file called “boxplot” separately and in addition to the post~~
2. created a new \_math folder
3. inside the folder, there’s a folder for the boxplot lesson plan
4. The rmd file that is automatically created will be the post on the website
   1. Add title and author if applicable
5. Open up the rmd with all the text and copy everything starting at the first chunk and paste into the boxplotspost.rmd
6. A headers #### to ##
7. Add title
8. Add description from the website
9. Comment out authors for now at least
10. To create the download links include the lines of code below:
11. I believe – you have to knit first before building the website to create the html post

Create straightline\_post

1. create\_post(“generate\_seq1\_post”, collection = “math”) Readme file in Lesson R project with instructions
2. Add YML by hand
3. title: **Equations of Straight Line Graphs - Part 1**
4. delete author
5. description: **This lesson is part 1 of 2 and demonstrates how to use R to plot straight line graphs (y=mx+c) and to observe what happens when the value of m changes**
6. copy full RMD content file **straightline1post** – remove section with files to download

Update preview photo on lessons/posts for straightline post

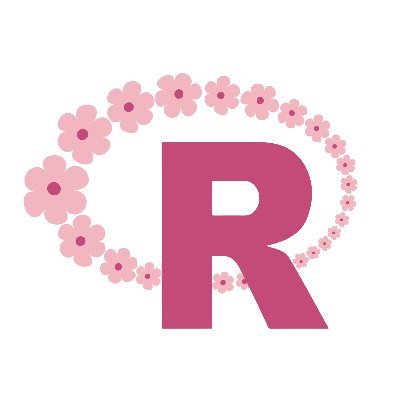
1. Open up and run the rmarkdown code for the lesson you want
2. Take a screen shot of a plot/graph that you want to use or even a few lines of code would work
3. Save the image in the lesson folder
4. In the lesson post, add “preview: straightline1\_photo.png” to the last line of the YAML

END TUTORIAL 6 HERE

Setting up Google Analytics

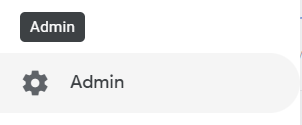
1. You can add [Google Analytics](https://analytics.google.com/) to your website by adding a google\_analytics tracking ID to your \_site.yml file
2. Add “google\_analytics: measurement\_id” to yaml
   1. Log in to goolge analytics
   2. Set up a data stream by going to Admin -> data streams -> choose a platform select web
   3. Add the website URL and choose a name
   4. Create stream
   5. Copy the measurement ID (e.g., G-93KHV17RG5)
   6. Open up the r project and navigate to the site.yml file and add the google analytics measurement id details
3. The Google Analytics tracking code will be automatically included on all pages of your website



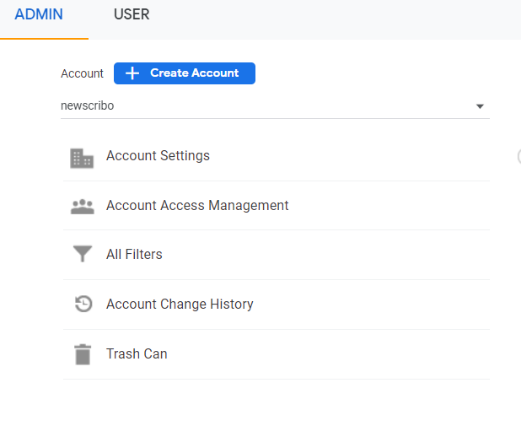


Google Analytics

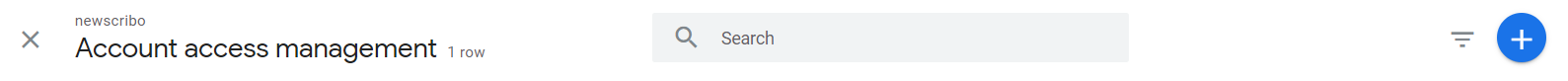
1. Go to google analytics <https://analytics.google.com/analytics/web/>
2. Sign in with the rgirls gmail account login
3. Navigate to and click the admin button in the bottom left-hand corner



1. Select “Account Access Management”



1. Click the blue + button and select “add users”



1. Enter my email: [jfsloane92@gmail.com](mailto:jfsloane92@gmail.com)
2. Check the box to notify new users by email
3. Select editor for the roles and data restrictions (this is all within google analytics only, I won’t have access to the gmail account)
4. Don’t worry about the data restriction boxes (just leave them unchecked)
5. Click the blue add button

