

# RStudio, Git, & GitHub

## R & RStudio

What is R?

- R is a language and environment for statistical computing and graphics.

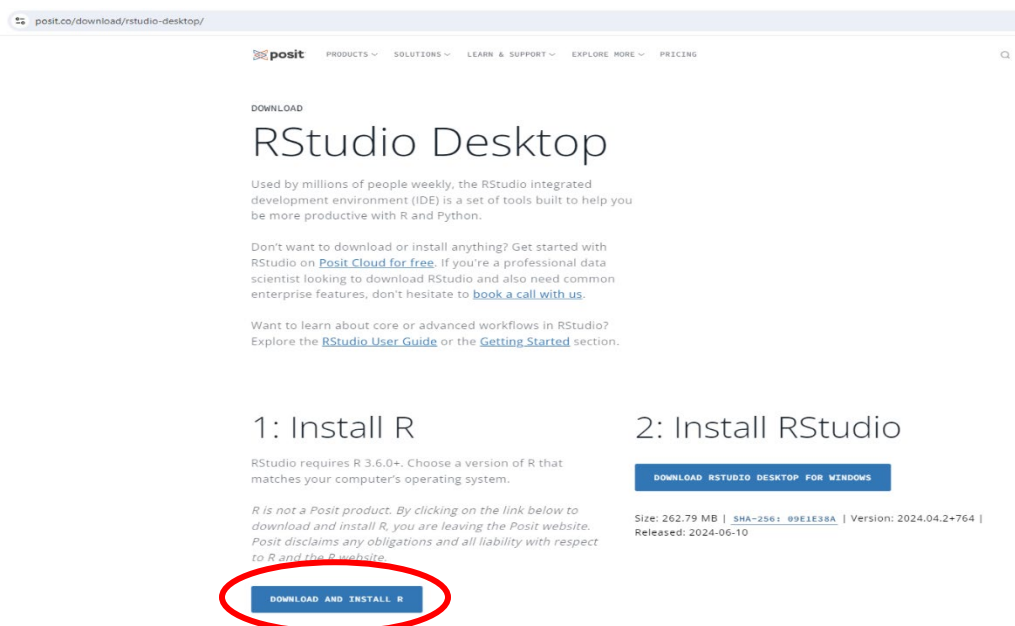
What is RStudio?

- It is a free and open-source integrated development environment (IDE) for R.

## Installing R & RStudio

### For Windows

- Go to: <https://posit.co/download/rstudio-desktop/>
- You must download R first, then download RStudio
- Click on “Download and Install R”.



posit.co/download/rstudio-desktop/

posit PRODUCTS SOLUTIONS LEARN & SUPPORT EXPLORE MORE PRICING

DOWNLOAD

## RStudio Desktop

Used by millions of people weekly, the RStudio integrated development environment (IDE) is a set of tools built to help you be more productive with R and Python.

Don't want to download or install anything? Get started with RStudio on [Posit Cloud for free](#). If you're a professional data scientist looking to download RStudio and also need common enterprise features, don't hesitate to [book a call with us](#).

Want to learn about core or advanced workflows in RStudio? Explore the [RStudio User Guide](#) or the [Getting Started](#) section.

### 1: Install R

RStudio requires R 3.6.0+. Choose a version of R that matches your computer's operating system.

*R is not a Posit product. By clicking on the link below to download and install R, you are leaving the Posit website. Posit disclaims any obligations and all liability with respect to R and the Posit website.*

[DOWNLOAD AND INSTALL R](#)

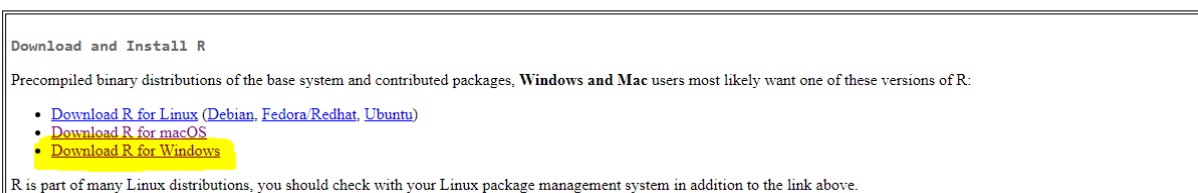
### 2: Install RStudio

[DOWNLOAD RSTUDIO DESKTOP FOR WINDOWS](#)

Size: 262.79 MB | SHA-256: 09E1E38A | Version: 2024.04.2+764 | Released: 2024-06-10

- Click on “Download R for Windows”

### The Comprehensive R Archive Network



Download and Install R

Precompiled binary distributions of the base system and contributed packages, **Windows and Mac** users most likely want one of these versions of R:

- [Download R for Linux \(Debian, Fedora Redhat, Ubuntu\)](#)
- [Download R for macOS](#)
- [Download R for Windows](#)

R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.

- ## R for Windows

- [base](#)
- [contrib](#)
- [old contrib](#)
- [Rtools](#)

- Binaries for base distribution. This is what you want to [install R for the first time](#)
- Binaries of contributed CRAN packages (for R >= 4.0.x).
- Binaries of contributed CRAN packages for outdated versions of R (for R < 4.0.x).
- Tools to build R and R packages. This is what you want to build your own packages on Windows, or to build R itself.

You may also want to read the [R FAQ](#) and [R for Windows FAQ](#).

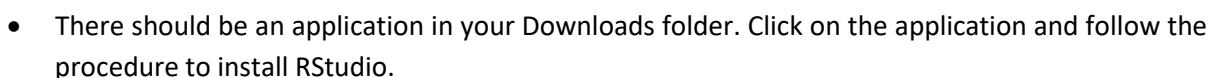
- Click on “Download R-4.4.1 for Windows”

### R-4.4.1 for Windows

[README on the Windows binary distribution](#)  
[New features in this version](#)

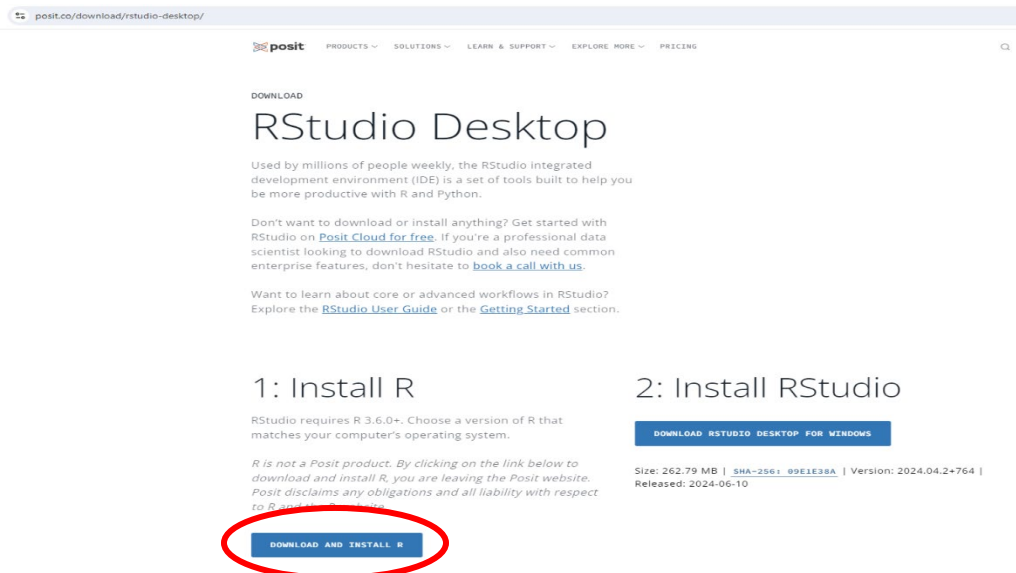
If you want to double-check that the package you have downloaded matches the package distributed by CRAN, you can compare the [md5sum](#) of the .exe to the [fingerprint](#) on the master server.

- There should be an application in your Downloads folder. Click on the application and follow the procedure to install R.
- Now that R is installed, you can proceed with downloading RStudio.
- Go to: <https://posit.co/download/rstudio-desktop/>
- Click on “Download RStudio Desktop for Windows”



## For Mac

- Go to: <https://posit.co/download/rstudio-desktop/>
- You must download R first, then download RStudio
- Click on “Download and Install R”.



- Click on “Download R for macOS”

### The Comprehensive R Archive Network

Download and Install R

Precompiled binary distributions of the base system and contributed packages, **Windows and Mac** users most likely want one of these versions of R:

- [Download R for Linux \(Debian, Fedora, Redhat, Ubuntu\)](#)
- [Download R for macOS](#)
- [Download R for Windows](#)

R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.

- Click on “R-4.4.1-arm64.pkg” for Apple silicon (M1-3) or “R-4.4.1-x86\_64.pkg” for older Intel Macs.

#### R for macOS

This directory contains binaries for the base distribution and of R and packages to run on macOS. R and package binaries for R versions older than 4.0.0 are only available from the [CRAN archive](https://cran.r-project.org) so users of such versions should adjust the CRAN mirror setting (<https://cran.r-project.org>) accordingly.

Note: Although we take precautions when assembling binaries, please use the normal precautions with downloaded executables.

R 4.4.1 "Race for Your Life" released on 2024/06/14

Please check the integrity of the downloaded package by checking the signature:

```
pkgutil --check-signature R-4.4.1-arm64.pkg
in the Terminal application. If Apple tools are not available you can check the SHA1 checksum of the downloaded image:
openssl sha1 R-4.4.1-arm64.pkg
```

#### Latest release:

R 4.4.1 binary for macOS 11 (Big Sur) and higher, signed and notarized packages.

Contains R 4.4.1 framework, R app GUI 1.80, Tcl/Tk 8.6.12 X11 libraries and Texinfo 6.8. The latter two components are optional and can be omitted when choosing "custom install", they are only needed if you want to use the `tcRtk` R package or build package documentation from sources.

macOS Ventura users: there is a known bug in Ventura preventing installations from some locations without a prompt. If the installation fails, move the downloaded file away from the *Downloads* folder (e.g., to your home or Desktop).

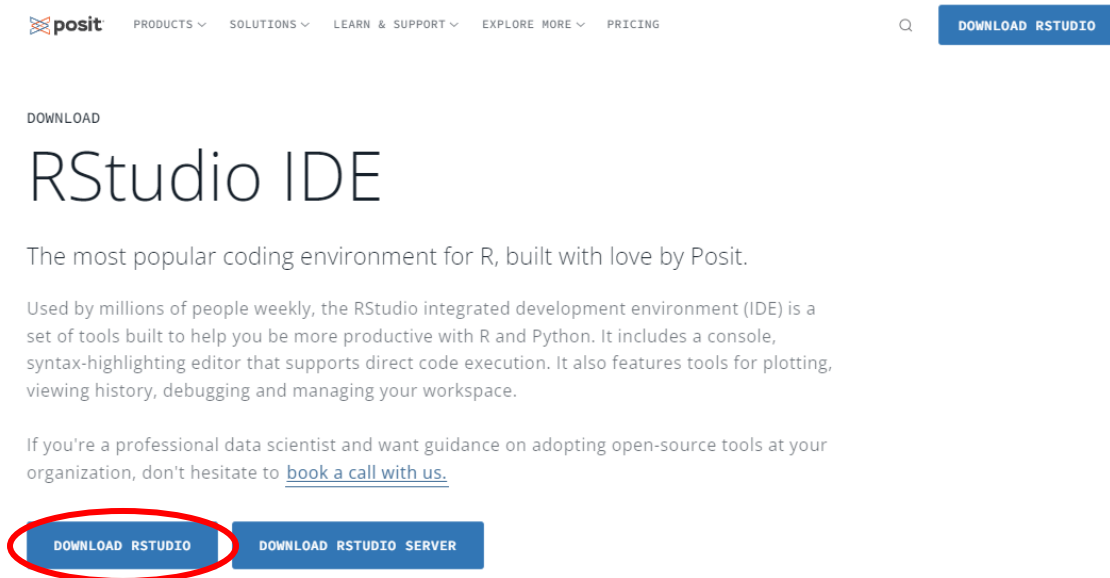
Note: the use of X11 (including `tcRtk`) requires [XQuartz](#) (version 2.8.5 or later). Always re-install XQuartz when upgrading your macOS to a new major version.

This release uses Xcode 14.2/14.3 and GNU Fortran 12.2. If you wish to compile R packages which contain Fortran code, you may need to download the corresponding GNU Fortran compiler from <https://mac.R-project.org/tools>. Any external libraries and tools are expected to live in `/opt/R/arm64` (Apple silicon) or `/opt/R/x86_64` (Intel).

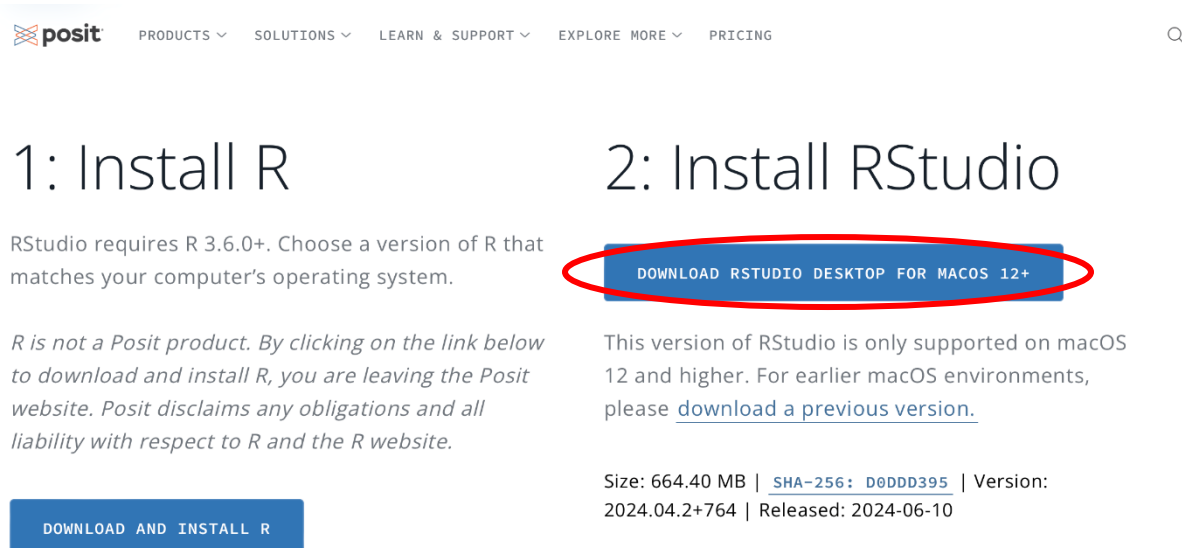
[NEWS](#) (for Mac GUI)

News features and changes in the R.app Mac GUI

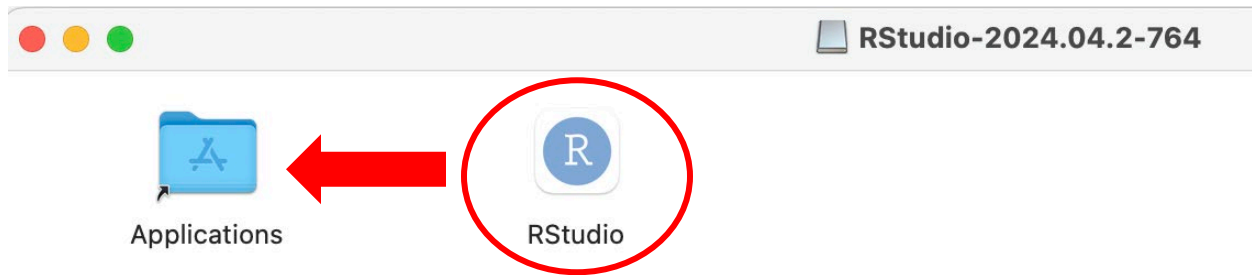
- There should be an application in your Downloads folder. Click on the application and follow the procedure to install R.
- Once the installation of R is successful, download RStudio.
- Go to: <https://posit.co/downloads/>
- Click on “Download RStudio”



- Click on “Download RStudio Desktop for MACOS 12+”



- There should be an application in your Downloads folder. Click on the application to open the installer, then drag RStudio into your Applications folder.



## Git

Git is a version control system that helps you keep track of changes in your code or any other files. It's like a time machine for your work. Git allows you to:

- **Track Changes:** You can record every change you make to your files. This includes adding new files, editing existing ones, and deleting files.
- **Collaborate with Others:** Git is designed for collaboration. Multiple people can work on the same project simultaneously, and Git helps to manage changes and merge them seamlessly (ish).
- **Create Backups:** Git acts as a backup system. If something goes wrong, you can easily restore your project to a previous state.
- **Work Offline:** You don't need an internet connection to use Git. You can commit changes to your local repository, and when you're online, you can sync those changes with a remote repository (like GitHub).

## GitHub

GitHub is a web-based platform that uses Git for version control. It's like a social network for programmers and a hosting platform for Git repositories. Here's what you can do with GitHub:

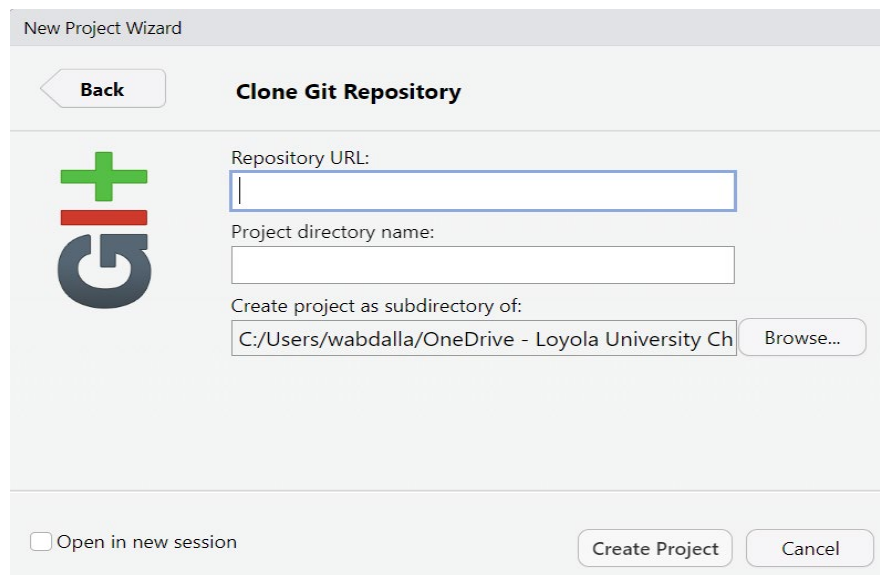
- **Store Code in the Cloud:** You can store your Git repositories online. This is particularly useful because it acts as a backup, and you can access your code from anywhere.
- **Collaborate with Others:** GitHub is an excellent platform for team collaboration. You can invite team members to your project, and everyone can work on the same codebase. This is what we will be using for our class to access notes and labs.
- **Share Your Work:** GitHub allows you to share your code with the world. You can make your repositories public so that anyone can see your code and contribute to it, or you can keep them private for restricted access.
- **Issue Tracking:** GitHub provides tools for issue tracking and project management. You can use it to keep track of tasks, bugs, and feature requests.

## Installing Git

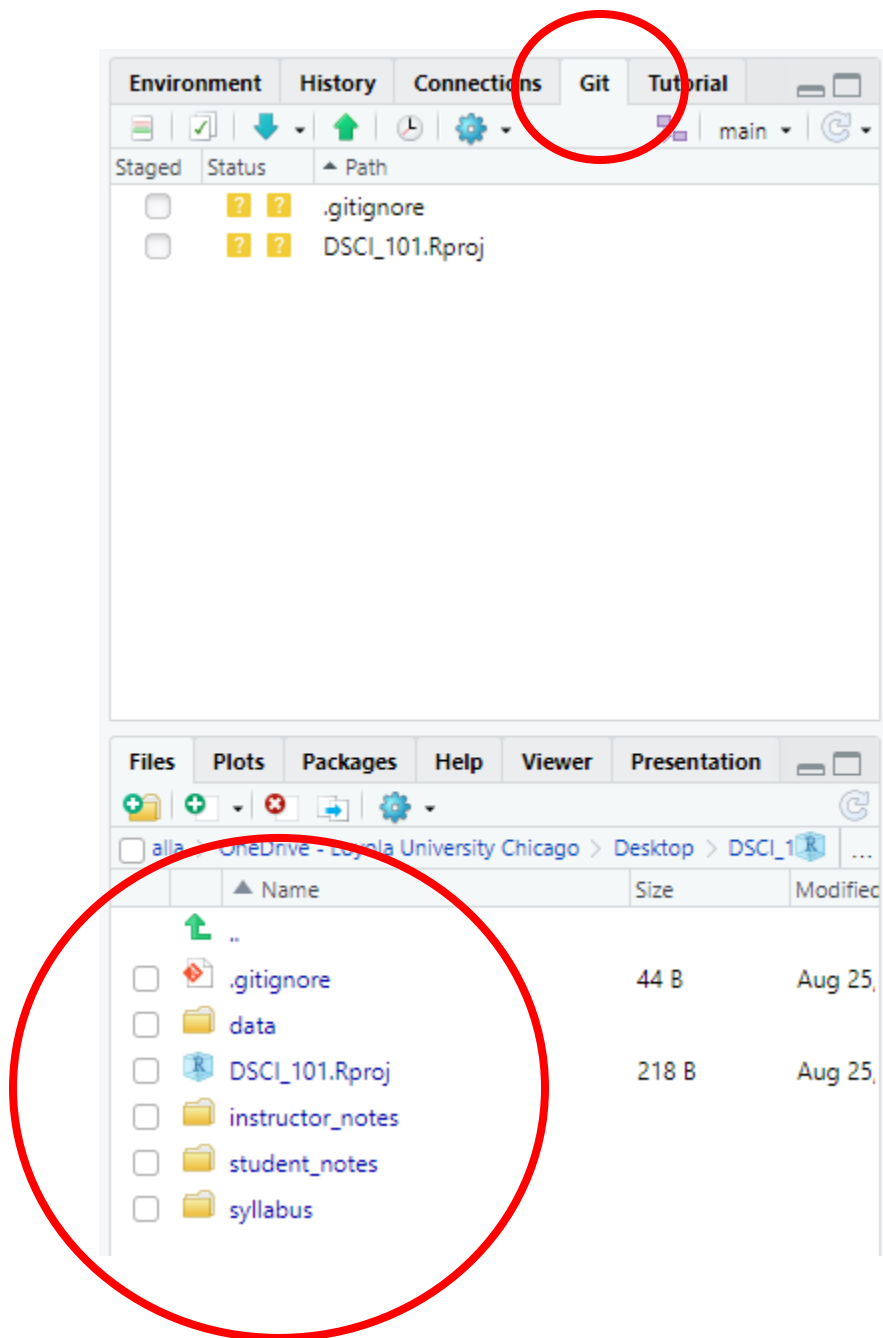
1. To install Git on your computer, go to <https://git-scm.com/downloads>
2. Under "downloads" select Windows or MacOS.
3. For Windows, click on "click here to download".
4. For Mac, go to "Binary installer" and click on "installer".
5. Follow all the steps to install git. Once git is downloaded successfully, you will need to close RStudio.
6. If you have trouble downloading/installing git, you can refer to the following video: <https://www.youtube.com/watch?v=F02LEVYEmQw>

## RStudio and GitHub

- Cloning a repository into RStudio:
  1. Go to <https://github.com/wabdalla>
  2. Click on the repository named DSCI\_101
  3. Go to “<> Code” and copy the directory of the repository.
  4. Open RStudio: File > New Project...
  5. Click on “Version Control”, then click on “Git”. You will get a window that looks like this:



6. Paste the URL that you copied in Step 3 from github where it says “Repository URL”.
7. The “Project directory name” should automatically be named DSCI\_101. If it’s not automatically named, then type “DSCI\_101” under “Project directory name”.
8. Click on “Browse” and choose the location where you’d like to save this folder. I recommend choosing the desktop since it’s easier to access.
9. Click on “Create Project” and wait for it to load.
10. Your RStudio should now look like this:





- Pull the Latest Changes: Before working on any classwork/homework, pull in the git window to get the latest version of the repository. The pull is the blue (down) arrow under the Git tab. You will not be committing or pushing on this class. Just *pull*.
- If you were working with others in a collaborative manner, you could be committing and pushing changes after you pull.
  - Commits:
    - In Git, a “commit” is like taking a snapshot of your project at a specific moment. It records all the changes you’ve made since the last snapshot.
    - Think of commits as checkpoints in your project’s history, showing what was changed and who made the changes.
    - Every commit has a unique ID.
    - You write a brief message to explain what each commit does.
  - Pushing:
    - “Pushing” means sharing your local project changes with others on GitHub.
    - It’s like uploading your work online so that your team members or collaborators can see and use it.
    - When you push, you’re keeping the shared project up to date.
  - Typical Workflow:
    - Pull Down Latest Work
    - Make Changes
    - Record Changes (Commit)
    - Share Changes (Push)
    - Stay Updated (Pull)

This cycle helps everyone stay coordinated when working on the same project.

**Things for you to do before next class:**

- Read the RStudio, Git, & GitHub document.
- Install R & RStudio
- Install Git
- Clone the class repository (under RStudio and GitHub section)
- Take the class [survey](#)