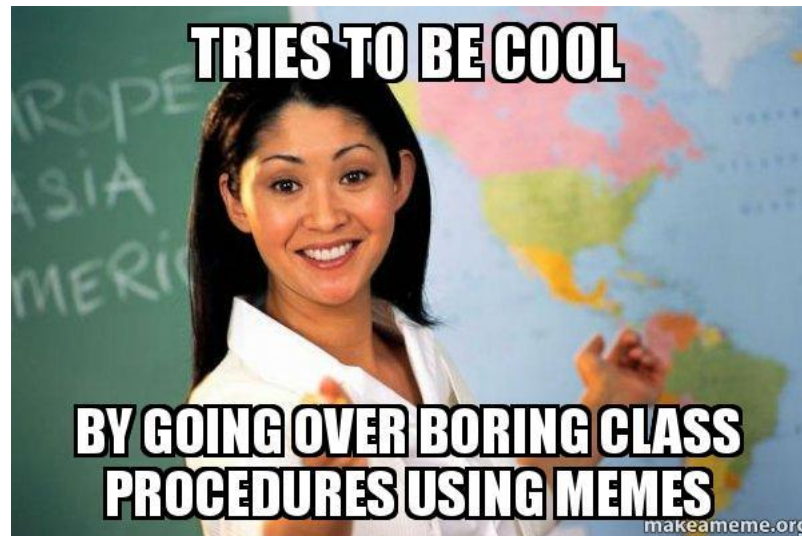


# Becoming Fluent in Data (2022)

---

## Setup



# Software

The course uses different software components for learning.



Statistical  
Software



Integrated  
Development  
Environment



Version  
Control &  
file sharing

git



GitHub

Cloud  
storage



Datacamp  
(Learning Platform)



Datacamp Mobile

\*All work on either Mac or  
Windows operation system.

# Setup

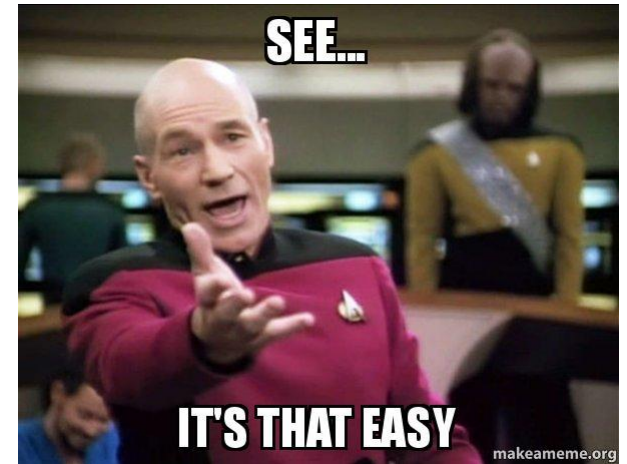
## Steps to complete.

Step 1. Install R.

Step 2. Install RStudio.

Step 3. Install git.

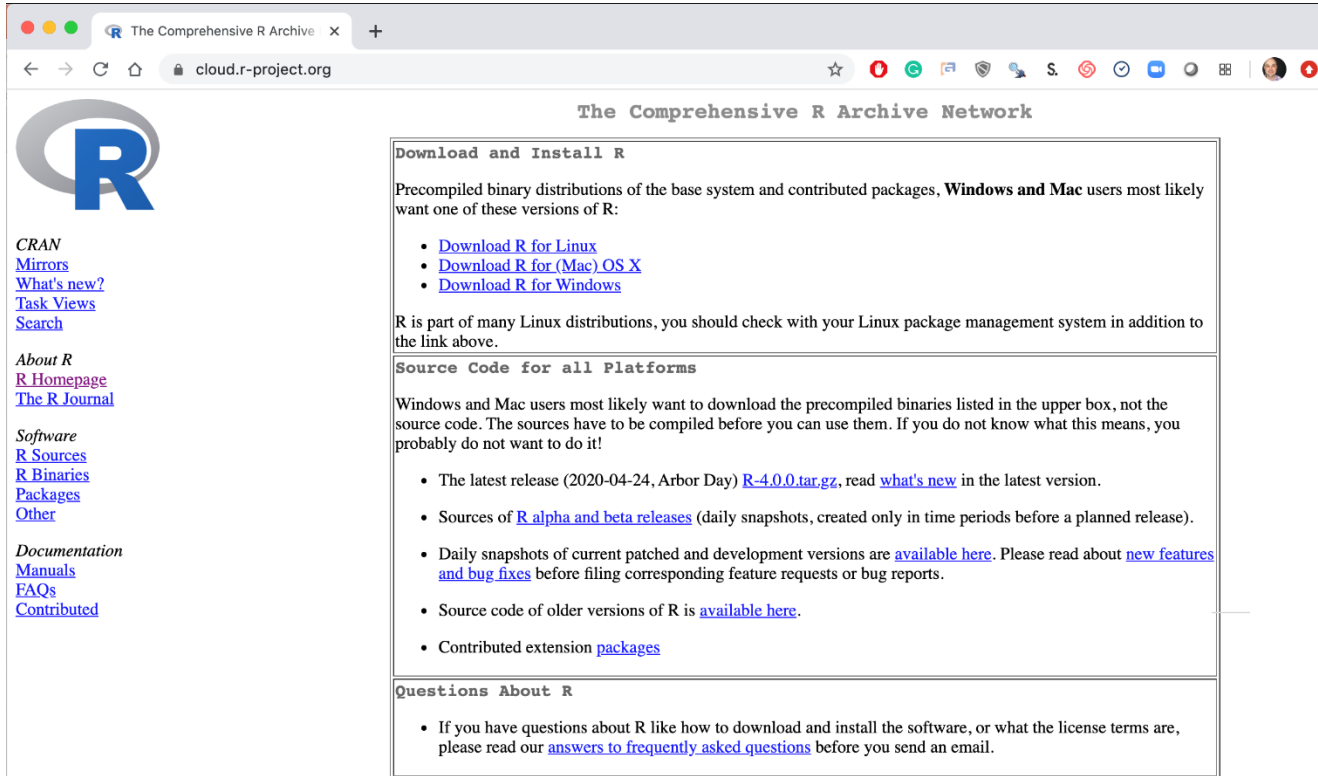
Step 4. Connect RStudio to Github.



\* If you have completed this setup and can run code ...you're off to a great start! And don't worry if you hit a hiccup, we will figure it out.

# Step 1. Install Base -R

## Local Laptop – Windows or Mac



The screenshot shows the website 'The Comprehensive R Archive Network' in a web browser. The browser's address bar shows 'cloud.r-project.org'. The website has a large blue 'R' logo on the left. Below the logo, there are links for 'CRAN', 'Mirrors', 'What's new?', 'Task Views', and 'Search'. Further down, there are links for 'About R', 'R Homepage', and 'The R Journal'. Under 'Software', there are links for 'R Sources', 'R Binaries', 'Packages', and 'Other'. Under 'Documentation', there are links for 'Manuals', 'FAQs', and 'Contributed'. The main content area is titled 'The Comprehensive R Archive Network' and contains a section 'Download and Install R'. This section states: 'Precompiled binary distributions of the base system and contributed packages, **Windows and Mac** users most likely want one of these versions of R:'. It lists three links: 'Download R for Linux', 'Download R for (Mac) OS X', and 'Download R for Windows'. Below this, it says: 'R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.' There is also a section 'Source Code for all Platforms' which states: 'Windows and Mac users most likely want to download the precompiled binaries listed in the upper box, not the source code. The sources have to be compiled before you can use them. If you do not know what this means, you probably do not want to do it!'. It lists four items: 'The latest release (2020-04-24, Arbor Day) [R-4.0.0.tar.gz](#), read [what's new](#) in the latest version.', 'Sources of [R alpha and beta releases](#) (daily snapshots, created only in time periods before a planned release).', 'Daily snapshots of current patched and development versions are [available here](#). Please read about [new features and bug fixes](#) before filing corresponding feature requests or bug reports.', and 'Source code of older versions of R is [available here](#).' There is also a section 'Questions About R' which states: 'If you have questions about R like how to download and install the software, or what the license terms are, please read our [answers to frequently asked questions](#) before you send an email.'

Download the appropriate version Linux, Mac or Windows.

<https://cloud.r-project.org/>

# Step 2. Install RStudio

## Local Laptop – Windows or Mac

	RStudio Desktop Open Source License <b>Free</b> <a href="#">DOWNLOAD</a> <a href="#">Learn more</a>	RStudio Desktop Commercial License <b>\$995 /year</b> <a href="#">BUY</a> <a href="#">Learn more</a>	RStudio Server Open Source License <b>Free</b> <a href="#">DOWNLOAD</a> <a href="#">Learn more</a>	RStudio Server Pro Commercial License <b>\$4,975 /year</b> (5 Named Users) <a href="#">BUY</a> <a href="#">Evaluation</a>   <a href="#">Learn more</a>
Integrated Tools for R	✓	✓	✓	✓
Priority Support		✓		✓
Access via Web Browser			✓	✓
Enterprise Security				✓
Project Sharing				✓
Manage Multiple R Sessions				✓

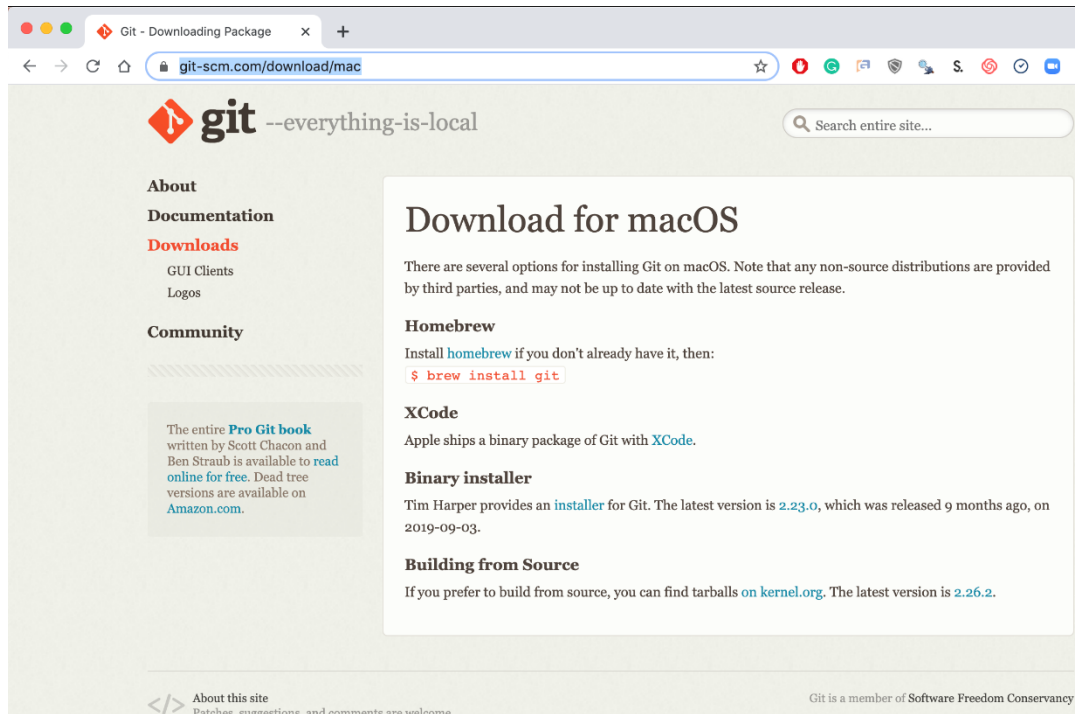
### Download the RStudio Desktop (Free)

<https://rstudio.com/products/rstudio/download/>

<https://rstudio.com/products/rstudio/download/#download>

# Step 3. Install Git (Mac)

## Local Laptop – Mac



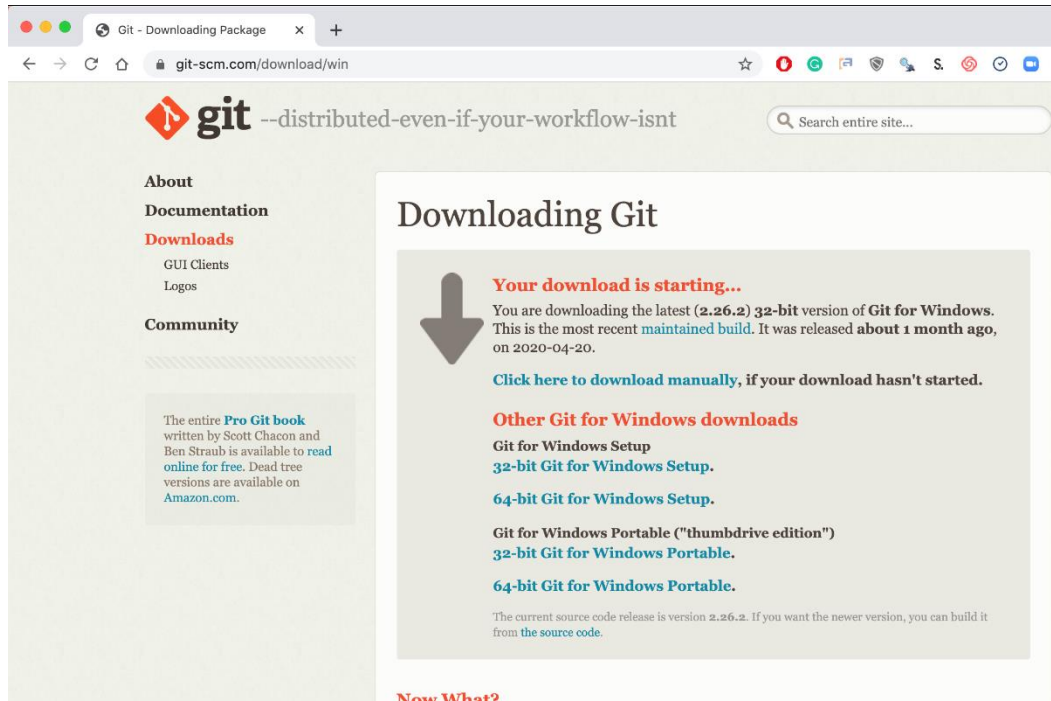
Download the Git Program, then follow prompts for installation.

<https://git-scm.com/download/mac>

<https://www.youtube.com/watch?v=twFo9wCpdSU>

# Step 3. Install Git (Windows)

## Local Laptop – Windows



Download the Git Program, then follow prompts for installation.

<https://git-scm.com/download/win>

<https://www.youtube.com/watch?v=nbFwejl5HIY>

# Magic

---

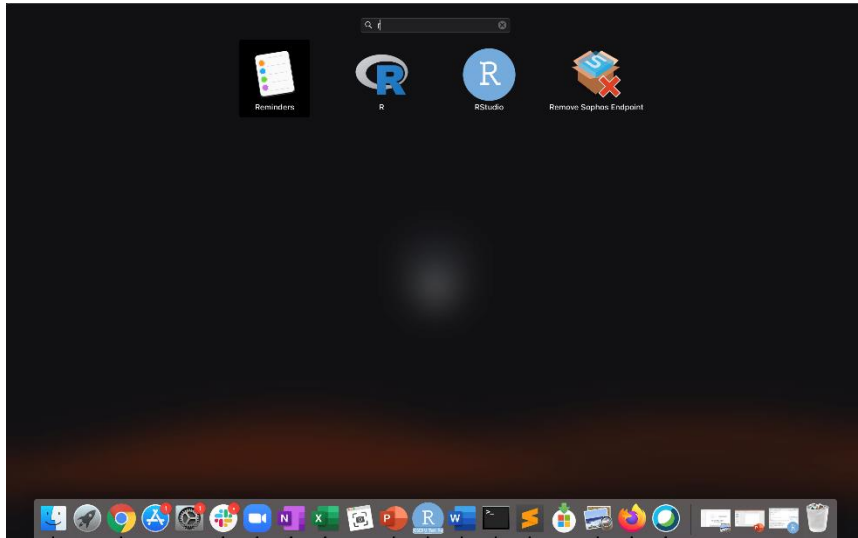


You should now have R, R-Studio and git.



# All good?

---



Mac: Use “Launchpad” & Type R

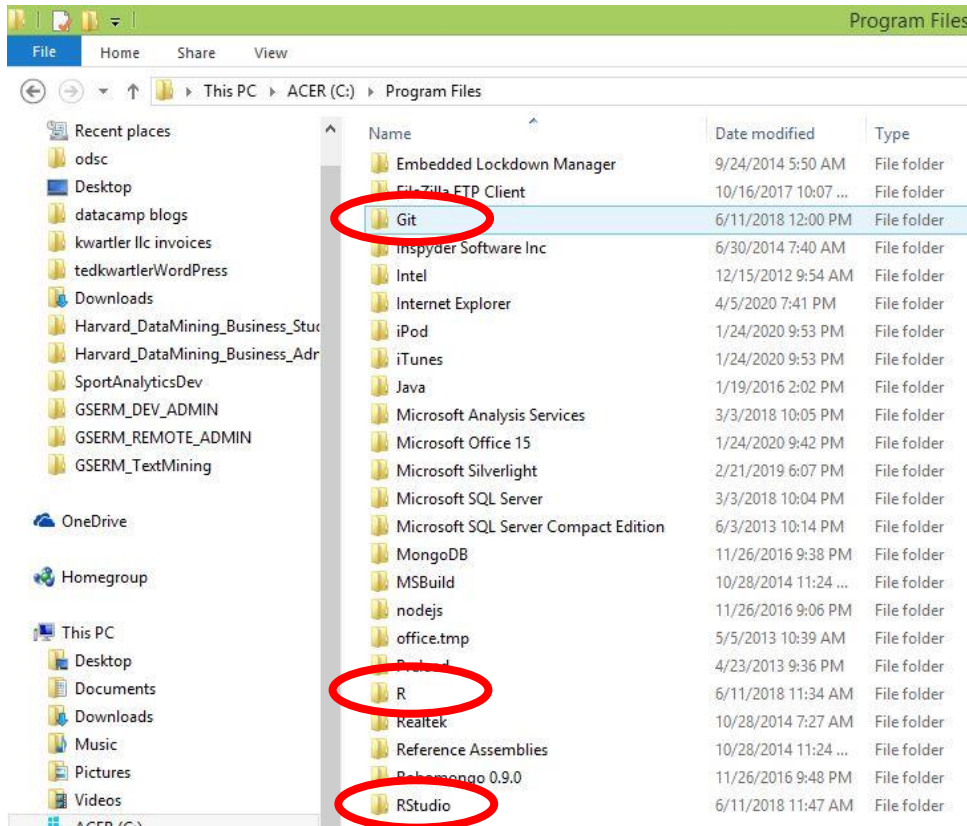


```
edward.kwartler — edward.kwartler@edward: ~ — -bash — 80x24
Last login: Sat May 16 12:25:19 on ttys000
edward.kwartler@edward:~$ git version
git version 2.22.0
edward.kwartler@edward:~$
```

Mac: Use: “Terminal” & Type “git version”

Mac: Look for R and RStudio Icons.

# All good?



**Windows: Navigate to the program folder look for git, R and RStudio**

# You should now have R, R-Studio and git.

---



# Step 4. Connect Rstudio to Repo.

Now that you have R, R-Studio & git, time to connect to our class repository.



R Studio sits on top of base R & adds functionality.



Class Files hosted on [github.com](https://github.com)



Makes the programmatic connection.

\*Think of GitHub as a dropbox for programmer.

# Step 4. Setup a new project & do a “git pull”

For you to look at in browser:

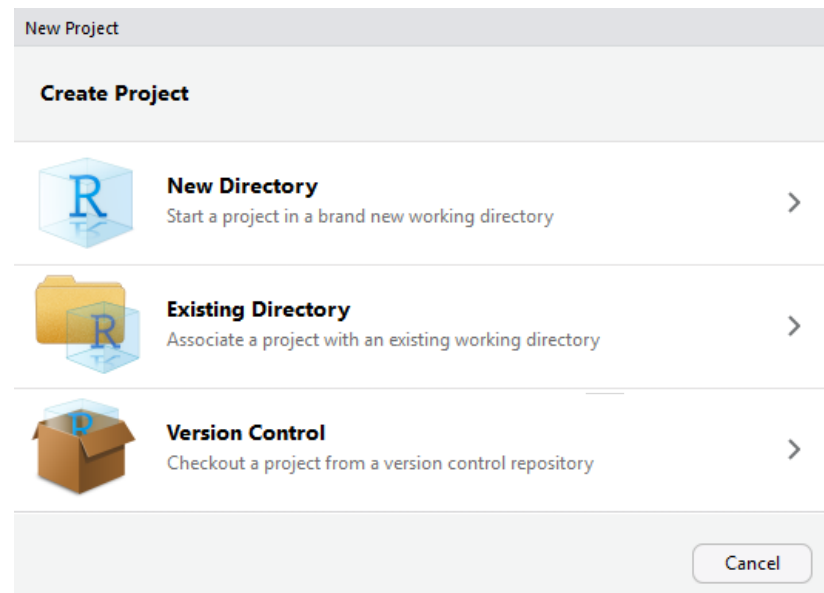
[https://github.com/MarcoKuehne/seminars\\_in\\_applied\\_economics](https://github.com/MarcoKuehne/seminars_in_applied_economics)

Enter in your Rstudio Instance:

[https://github.com/MarcoKuehne/seminars\\_in\\_applied\\_economics.git](https://github.com/MarcoKuehne/seminars_in_applied_economics.git)

Open RStudio

1. File
2. New Project
3. Version Control

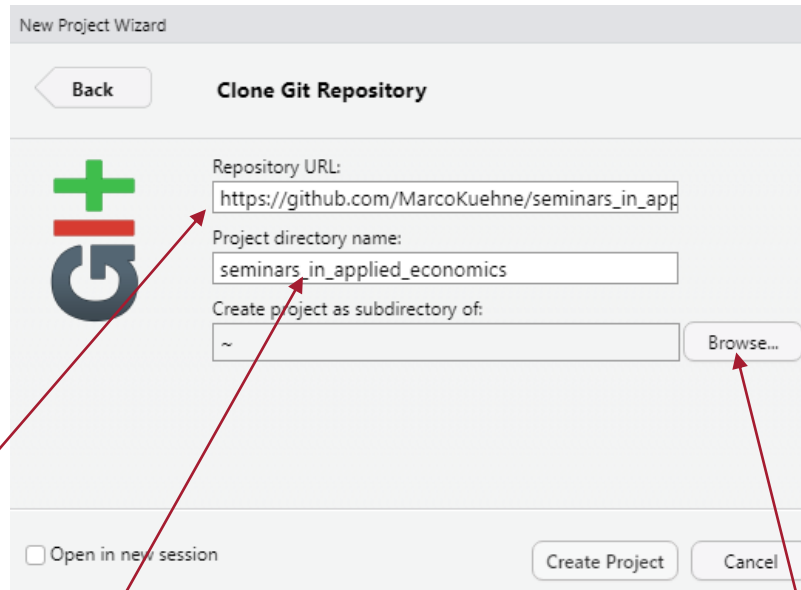


# Step 4. Setup a new project & do a “git pull”

In RStudio

1. File
2. New Project...
3. Version Control
4. Select Git
5. Add URL, folder name & destination for files

Appears by magic



[https://github.com/MarcoKuehne/seminars\\_in\\_applied\\_economics.git](https://github.com/MarcoKuehne/seminars_in_applied_economics.git)

# Each day we will perform a “git pull” in case there are updates to curriculum to aid learning.

---

## In RStudio

1. File
2. New Project...
3. Version Control
4. Select Git
5. Add URL, folder name & destination for files
6. EACH Class session perform a **Git Pull** to update files



```
Git Pull  
  
>>> git pull  
Already up to date.
```

# One folder, all data.

Somewhere in your favorite destination you now synchronized all course material (data, homework, lectures, etc.).

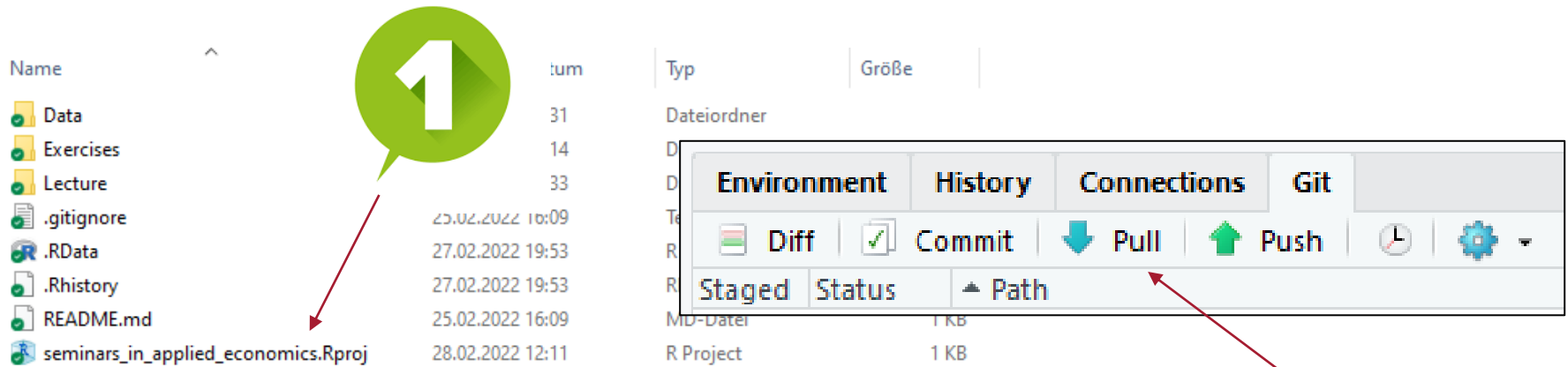
Name	Änderungsdatum	Typ	Größe
Data	27.02.2022 08:31	Dateiordner	
Exercises	27.02.2022 09:14	Dateiordner	
Lecture	27.02.2022 08:33	Dateiordner	
.gitignore	25.02.2022 16:09	Textdokument	1 KB
.RData	27.02.2022 19:53	R Workspace	49 KB
.Rhistory	27.02.2022 19:53	RHISTORY-Datei	1 KB
README.md	25.02.2022 16:09	MD-Datei	1 KB
seminars_in_applied_economics.Rproj	28.02.2022 12:11	R Project	1 KB

Everytime you start to work, click this project file “.Rproj”  
EACH Class session perform a **Git Pull** to update files



# One folder, all data.

Again.      Start “.Rproj”.      Do “git pull”.      Awesome.



## Workflow (git yourself a coffee)

- 1) Click project file “seminars\_in\_applied\_economics .Rproj”
- 2) Go right upper panel, Git tab, Pull button

Enjoy

# True stories ...

Mac tot 11:54

Man darf das nicht installieren 11:54

Nicht gut 11:59

Alles tot 11:59

Lisi kann nichts mehr machen 11:59

Mac kaputt 11:59

Man darf das nicht tun 11:59

12:00

😭😭😭😭 12:00

Ich würde allen Studenten ganz dringend von dieser abraten 12:01

Das ist es nicht wert 12:01

**Mac dead**

**Never ever install this**

**No good**

**Everything dead**

**Cannot do anything anymore**

**Mac died**

**Really, never ever do this**

**IMHO I advise all students against this .. stuff**

**It's not worth it**



# Now just run some code (we will explain it later)

---

[https://github.com/MarcoKuehne/seminars\\_in\\_applied\\_economics](https://github.com/MarcoKuehne/seminars_in_applied_economics) has the R code.

## 🔗 R packages

---

Here are the first R packages that we will need. Run this install once on your system.

```
# Easiest method to run in your console
install.packages('pacman')
pacman::p_load(tidyverse, cowplot, magick, haven, DT, beepr, fun, cowsay, plotly)

# You can install packages individually such as below if pacman fails.
install.packages('tidyverse')

# Or using base functions use a nested `c()`
install.packages(c("fun", "beepr", "cowsay"))
```

`install.packages('pacman')`

`pacman::p_load(tidyverse, cowplot, magick, haven, DT, beepr, fun, cowsay, plotly)`

# Last item. Create a personal folder.

---

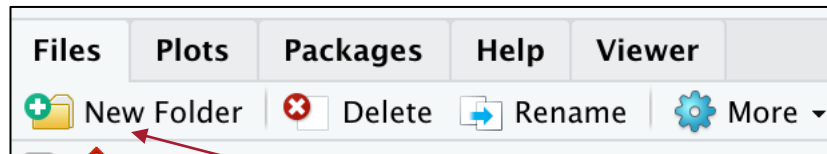
## Why?

- Create a personal folder inside the R project.
- “Git pull” downloads all new changes from the Github repository.
- There is a “Git push” button that you cannot use.
- The Github repo is public but read-only.
- Create a personal folder that is ignored by git.
- This is where you can work on scripts and homework.

# Last item. Create a personal folder.

In the lower right file section click “New Folder”. Name it “personal” exactly as in the gitignore (capitalization)

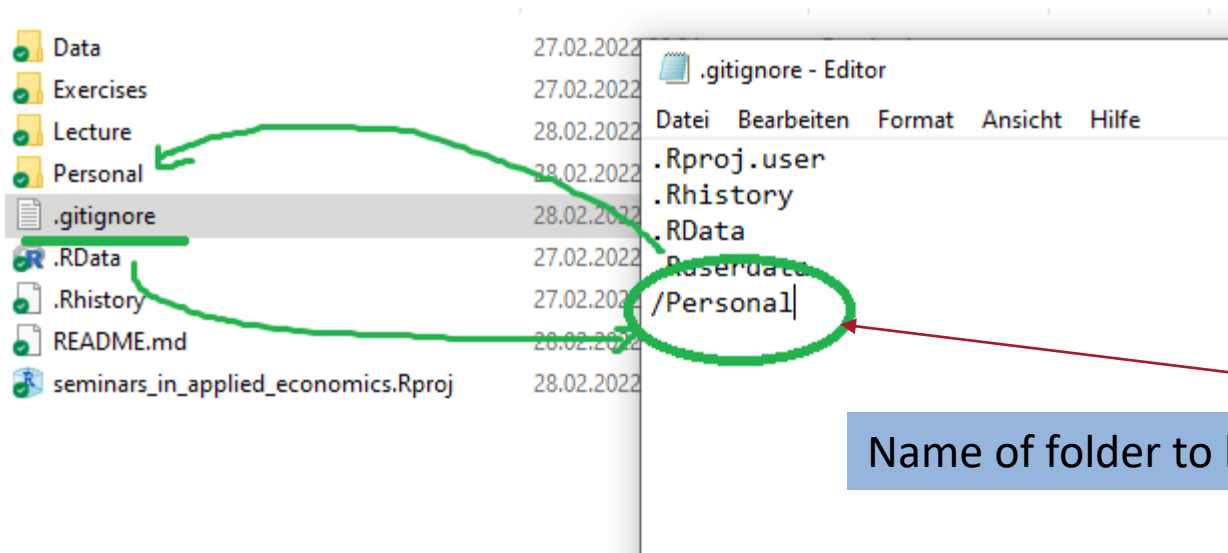
You can create folders from inside Rstudio.



Click New Folder

# Last item. Create a personal folder.

Open the “gitignore” file. Add slash personal



Name of folder to be ignored “Personal”

When you work and make changes to scripts, SAVE AS to a personal folder that is ignored by git.

Almost setup...*don't worry if you need help live.*

---

