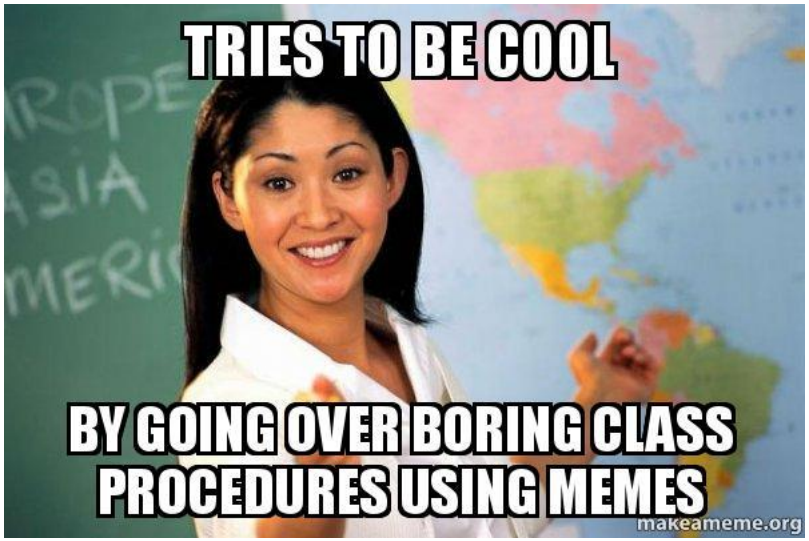
Becoming Fluent in Data (2022)

Setup



Software

The course uses different software components for learning.

Statistical Software

Integrated Development Environment

Version Control & file sharing

Cloud storage

Datacamp (Learning Platform)

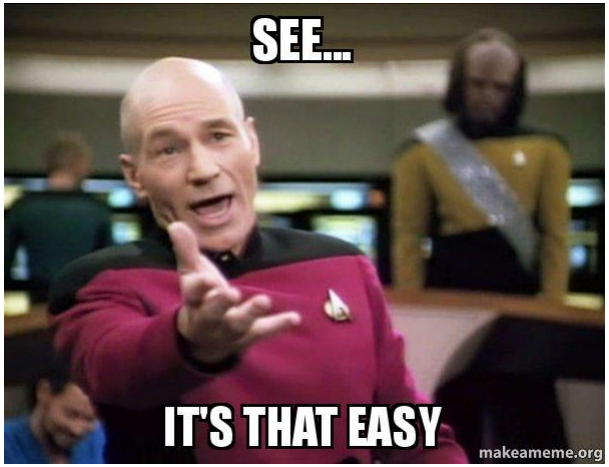
Datacamp Mobile

Assignments & Discussion

*All work on either Mac or Windows operation system.

Setup

Steps to complete.



Step 1. Install MS Teams.
Step 2. Install R.
Step 3. Install RStudio.
Step 4. Install git.
Step 5. 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.

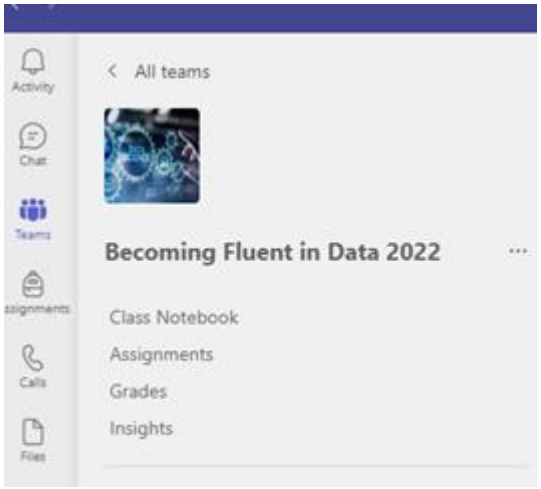
Step 1. Install MS Teams

Microsoft Teams: Office 365 Education

Register Office 365 education with your student E-Mail:
https://www.microsoft.com/de-de/education/products/office

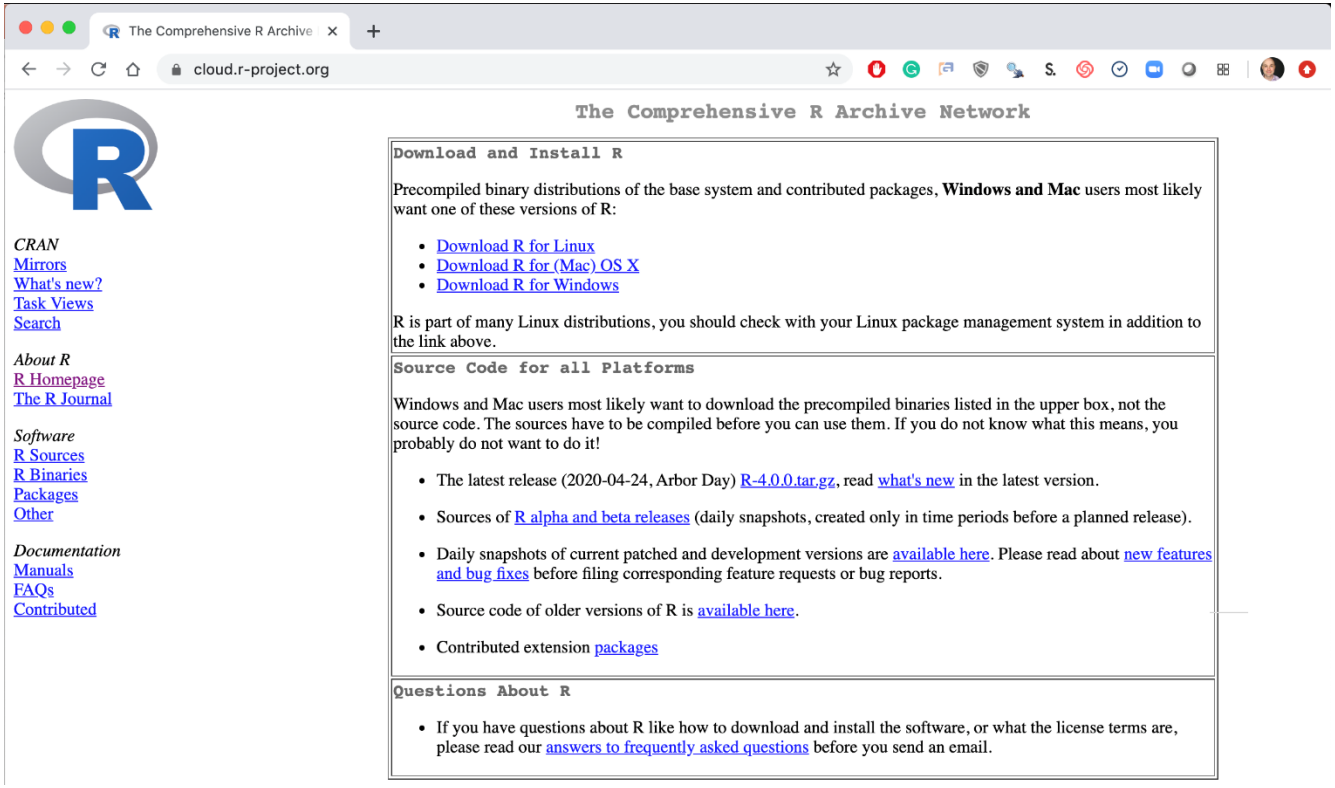
Use Team Link:

https://teams.microsoft.com/l/team/19%3aBL5KRIQ2fKd248VCfkT5xCpa33P_QmTvl_zQWd5YnbE1%40thread.tacv2/conversations?groupId=7eae0536-0949-4fed-b6dc-3c89aac103df&tenantId=41360a17-9216-4d8b-bc2d-26b7cc95fe99



Step 2. Install Base -R

Local Laptop – Windows or Mac

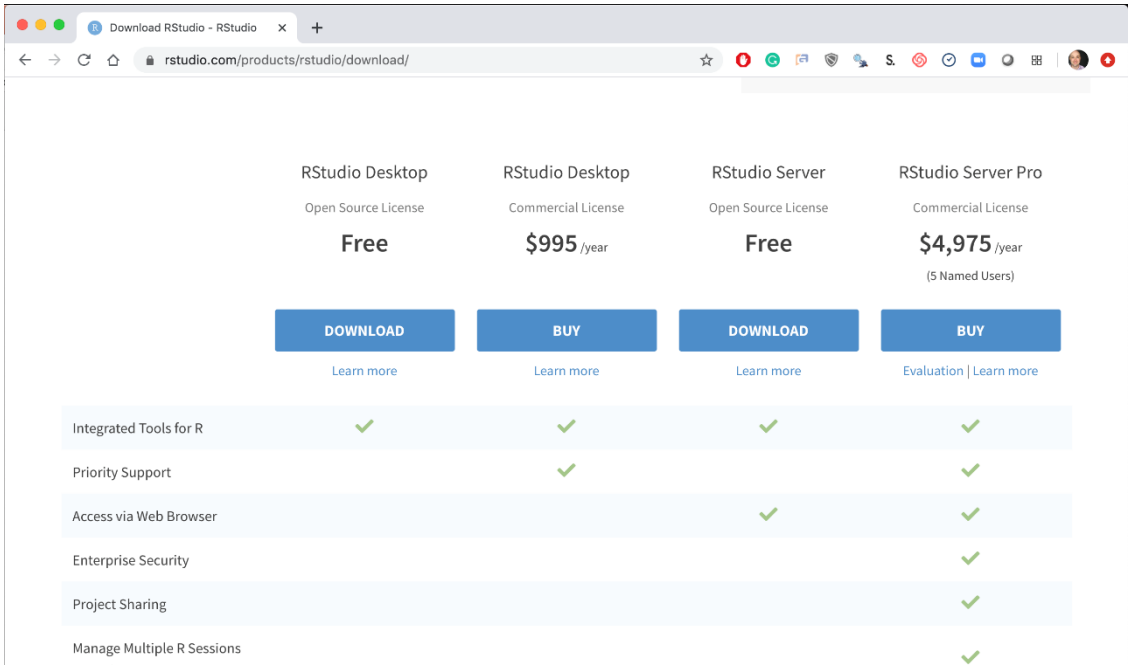


Download the appropriate version Linux, Mac or Windows.

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

Step 3. Install RStudio

Local Laptop – Windows or Mac



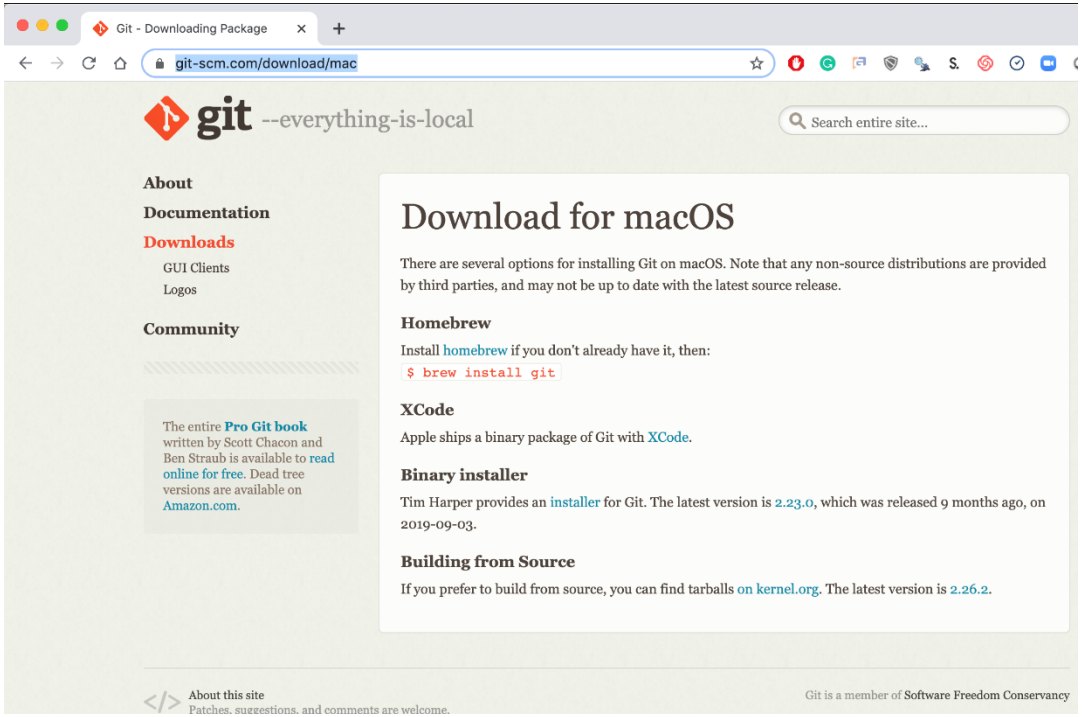
Download the RStudio Desktop (Free)

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

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

Step 4. Install Git (Mac)

Local Laptop – Mac



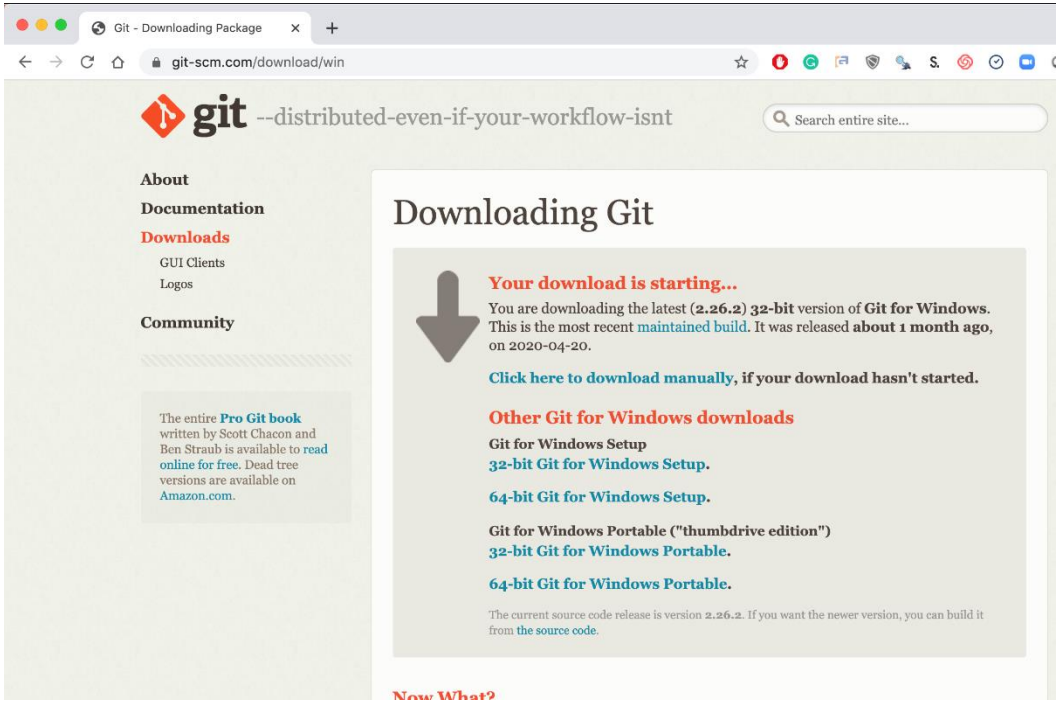
Download the Git Program, then follow prompts for installation.

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

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

Step 4. Install Git (Windows)

Local Laptop – Windows



Download the Git Program, then follow prompts for installation.

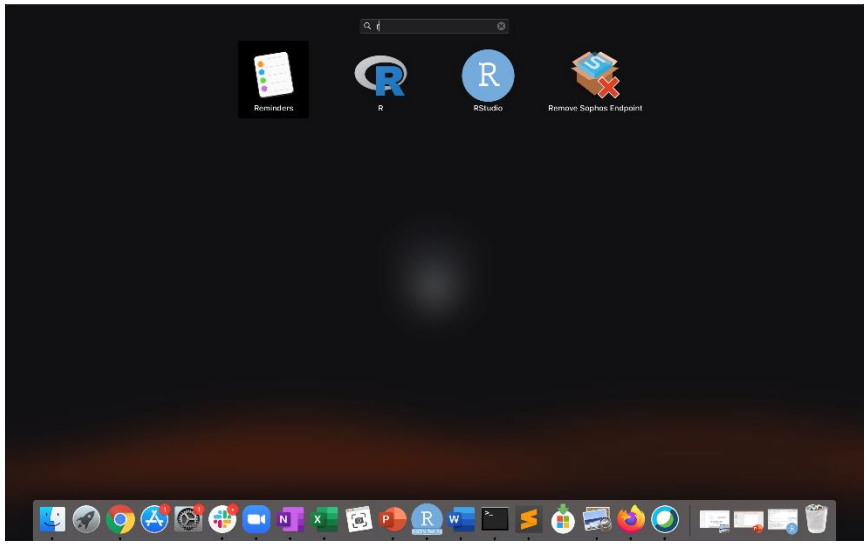
https://git-scm.com/download/win

https://www.youtube.com/watch?v=nbFwejIsHIY

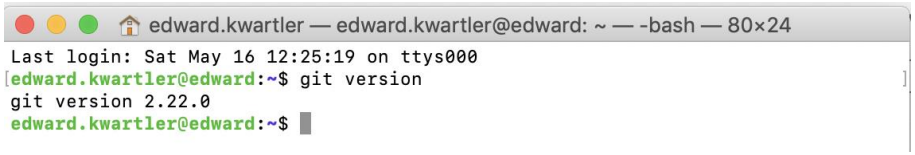
Magic

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

All good?



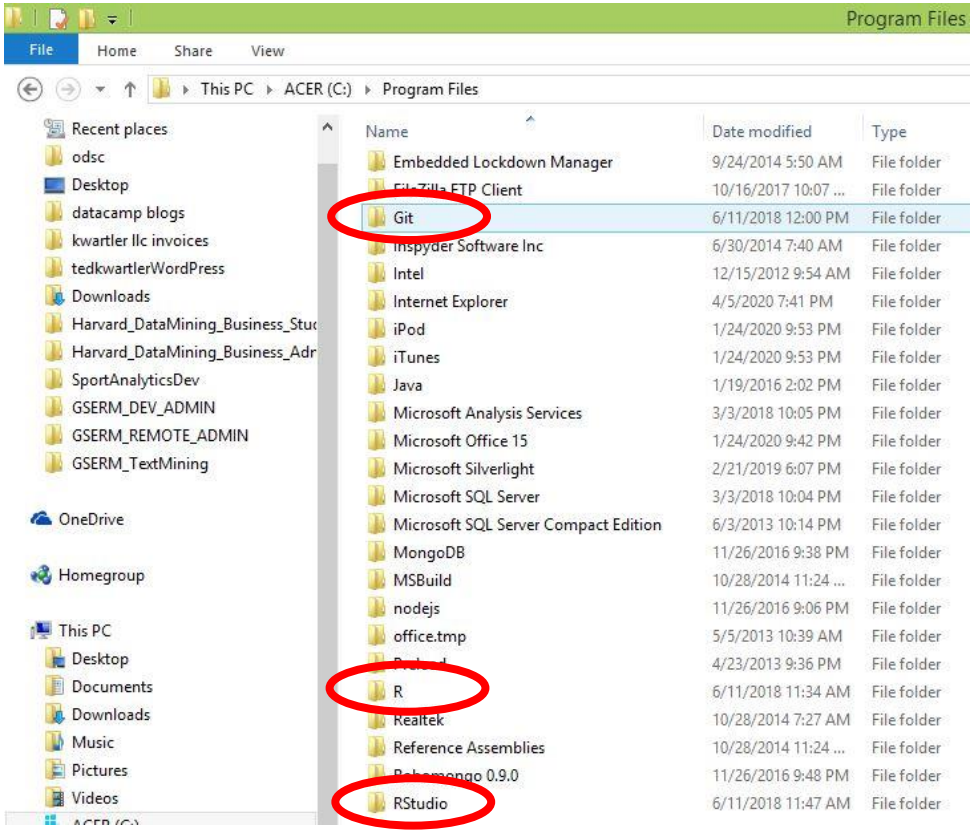
Mac: Use “Launchpad” & Type R



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

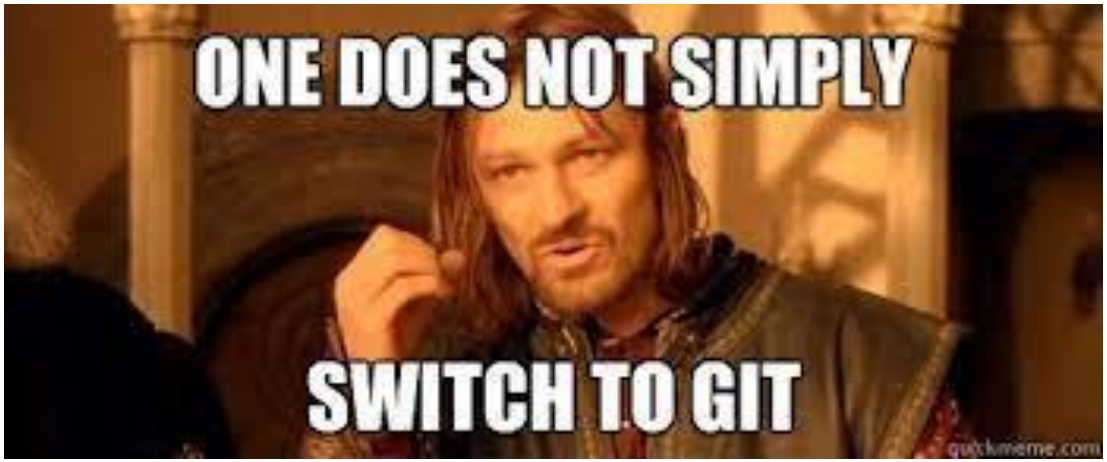
Mac: Look for R and RStudio Icons.

All good?



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

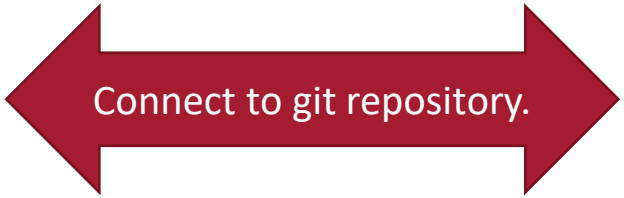
You should now have R, R-Studio and git.



Step 5. 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.



git

Makes the programmatic connection.

GitHub

Class Files hosted on github.com

*Think of GitHub as a dropbox for programmer.

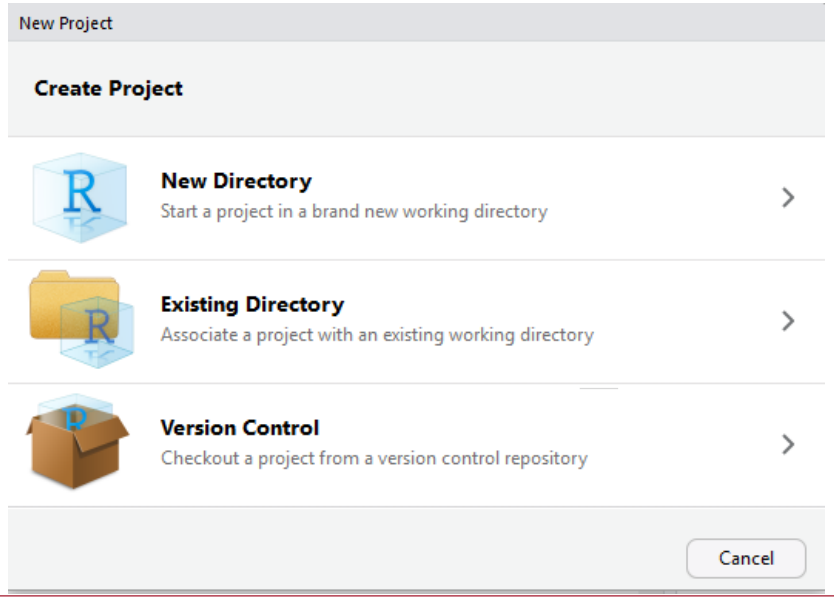
Step 5. Setup a new project & do a “git pull”

For you to look at in browser:
https://github.com/MarcoKuehne/seminars_in_applied_economics

Enter in your Rstudio Instance:
https://github.com/MarcoKuehne/seminars_in_applied_economics.git

Open RStudio

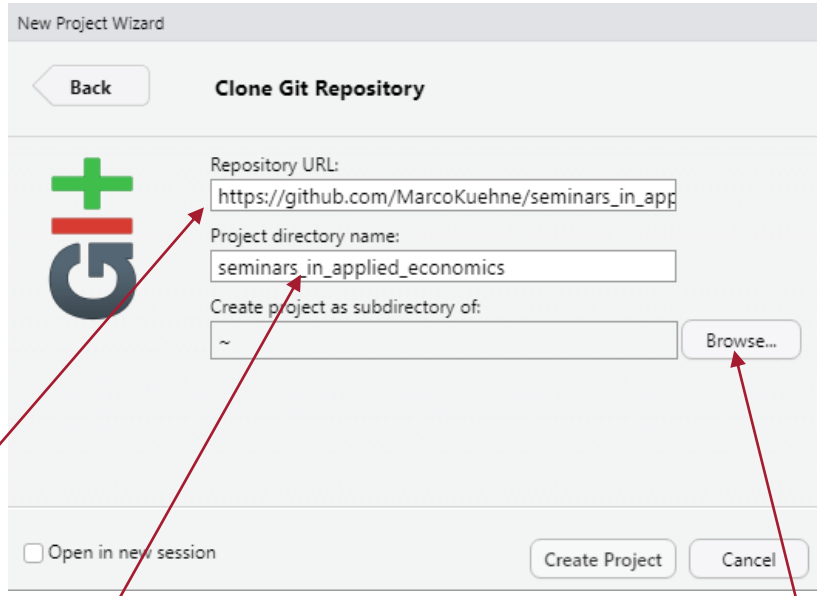
1. File
2. New Project
3. Version Control



Step 5. 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



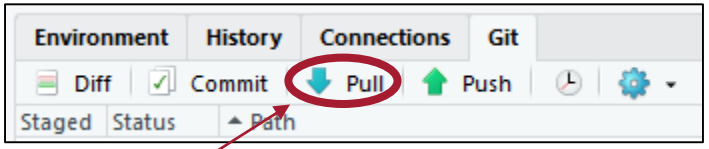
Appears by magic

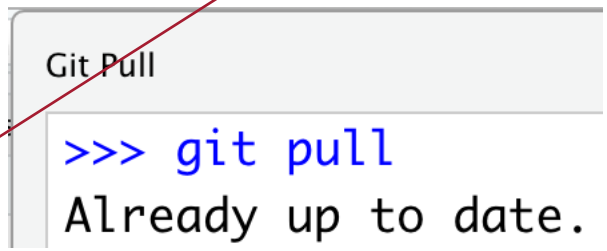
https://github.com/MarcoKuehne/seminars_in_applied_economics.git

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

In RStudio

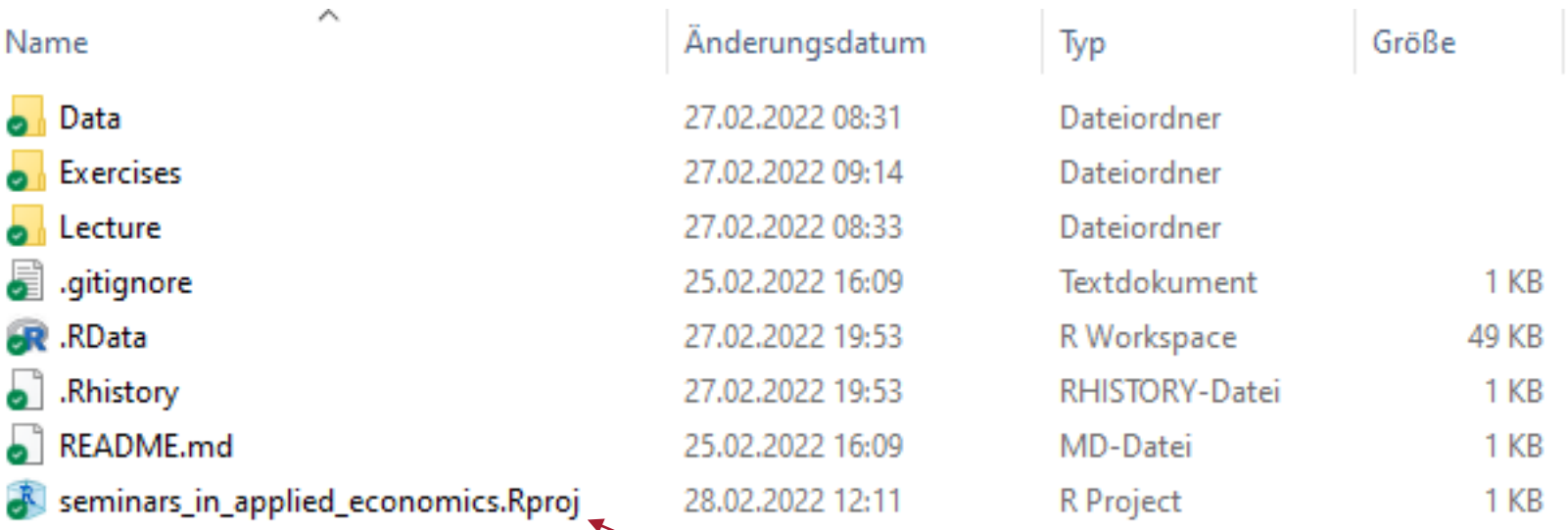
1. File
2. New Project…
3. Version Control
4. Select Git
5. Add URL, folder name & destination for files
6. EACH Class session perform a **Git Pull** to update files





One folder, all data.

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

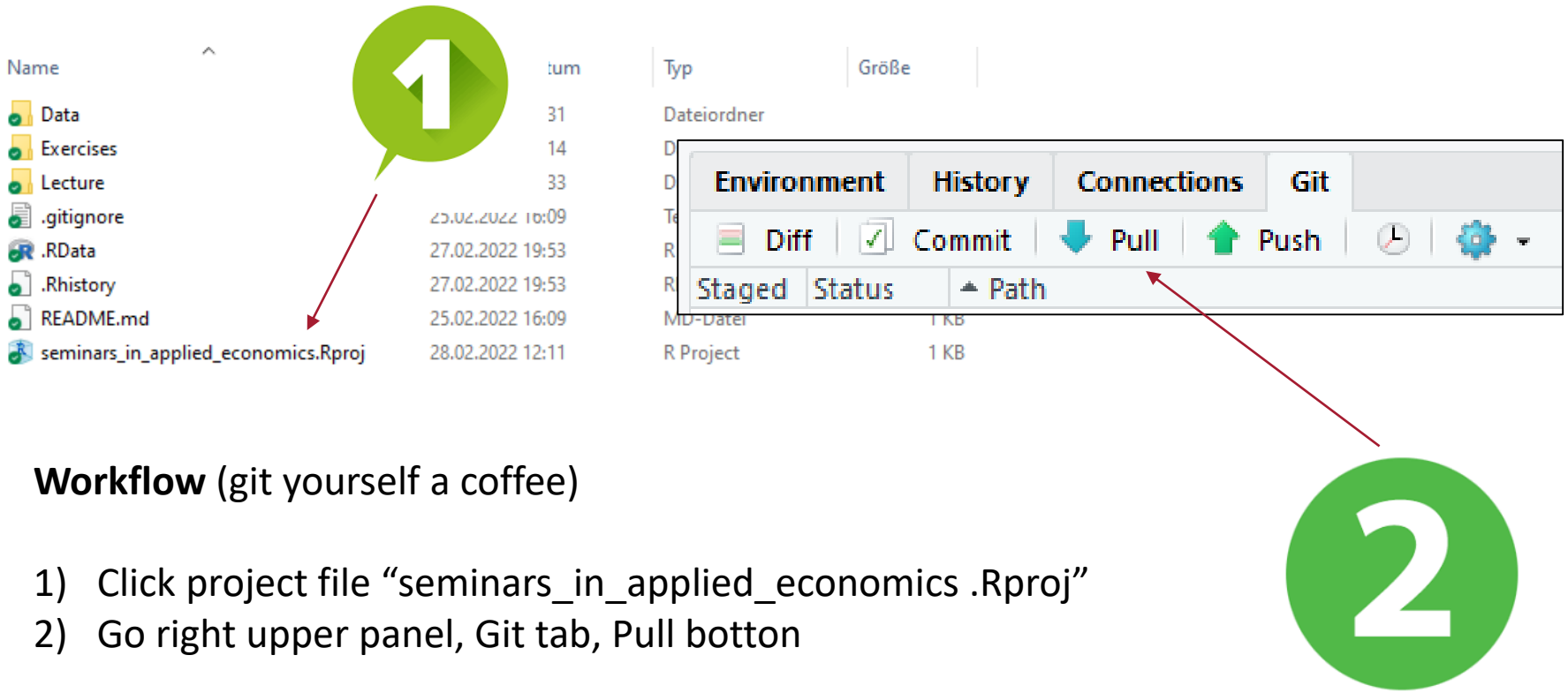


Name	Änderungsdatum	Typ	Größe
Data	27.02.2022 08:31	Dateiordner	
Exercises	27.02.2022 09:14	Dateiordner	
Lecture	27.02.2022 08:33	Dateiordner	
.gitignore	25.02.2022 16:09	Textdokument	1 KB
.RData	27.02.2022 19:53	R Workspace	49 KB
.Rhistory	27.02.2022 19:53	RHISTORY-Datei	1 KB
README.md	25.02.2022 16:09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

Enjoy

True stories …

Mac tot 11:54	Mac dead
Man darf das nicht installieren 11:54	Never ever install this
Nicht gut 11:59	No good
Alles tot 11:59	Everything dead
Lisi kann nichts mehr machen 11:59	Cannot do anything anymore
Mac kaputt 11:59	Mac died
Man darf das nicht tun 11:59	Really, never ever do this
12:00	
😪😪😪😪 12:00	
Ich würde allen Studenten ganz dringend von dieser abraten 12:01	IMHO I advise all students against this .. stuff
Das ist es nicht wert 12:01	It's not worth it

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

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.

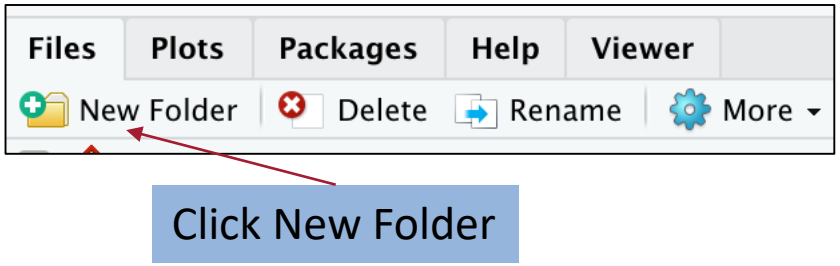
- “Git pull” downloads all new changes from the Github repository.
- There is a “Git push” button that you cannot use.
- The Github repo is public but read-only.

- Create a personal folder that is ignored by git.
- This is were you can work on scripts and homework.

Last item. Create a personal folder.

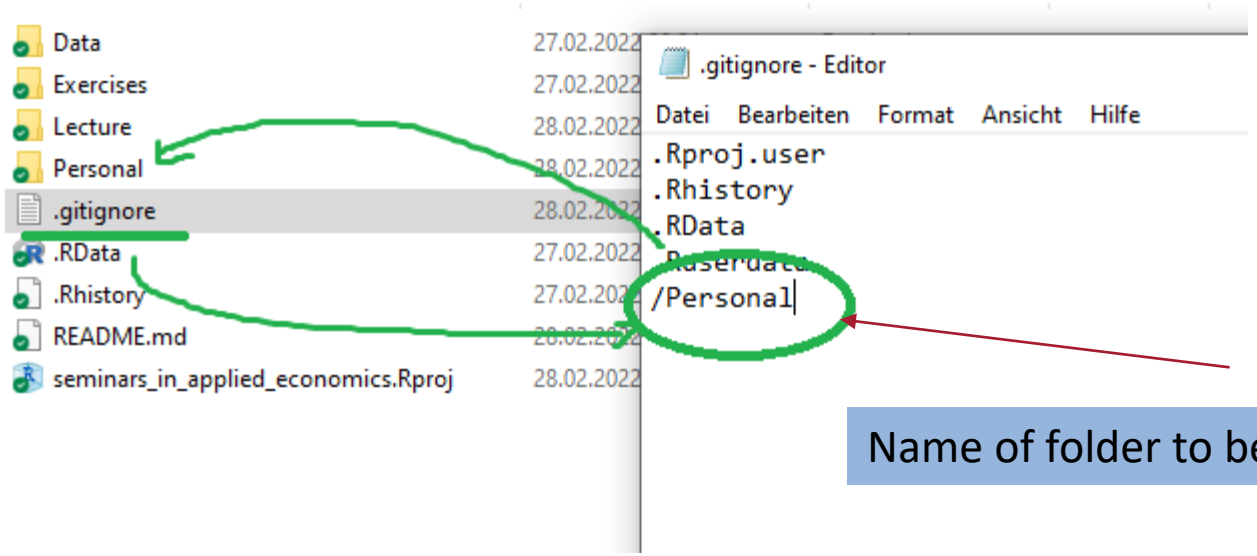
In the lower right file section click “New Folder”. Name it “personal” exactly as in the gitignore (capitalization)

You can create folders from inside Rstudio.



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.*