Data Science – intro 2

Course text, Jupyter Notebooks, arithmetic (the computer way), variables

Course TAs

- Arpan Nayak
- Thomas Roberts
- Poppy Aves

Keep learning

- I have learned and used lots of languages in my data science journey
  - assembly, Basic, FORTRAN, COBOL, Perl, C, C++, R, Python
  - At the same time, I am relatively new to Python
  - I can make mistakes. Making mistakes is part of coding.
  - If it looks like I am doing something wrong, ask. I may be and we will correct it together.
  - Seeing the process of making and correcting mistakes is important (how to find and correct mistakes is one of the main things coders need to learn)
  - The goal is not to make no mistakes. That is impossible.
  - The goal is to learn to test for, find and fix mistakes
- No prior knowledge is assumed for this course

Outline – What we will cover – Getting set up for the course

- Course resources
  - VLab and jupyter notebooks
  - Piazza (Q & A)
  - CANVAS page
  - e-textbook
  - Student Whatsapp group?
- Surviving the computer – some advice
- Our first exercise – Dracula

Course organization – course map

- The course runs on two tracks in parallel:

1) Python programming concepts
2) Data science concepts

These don't always look connected (and sometimes they aren't), but you need both.

More detailed map for each week

Week	Data concepts	Python	Examples
1-2	Sampling and distributions	Python basics: Variables, operations, functions	
3-5	Simulations	Arrays, loops, data frames	Births of boys/girls, Ethnicity of the jury
7	Permutations	Random shuffle, 'if' statement, compare means (t-tests)	McDonalds/Starbucks, Mosquitos and beer
8	Linear relationship	Handling missing data, correlation	Fertility and GDP
9	Linear relationship	Optimization, regression	Cholesterol levels and heart disease
10	Count data	chi-square test	Titanic

Guide to course tools

- CANVAS page – has links to tools and announcements
- Lecture notes and exercises – Vlab - notes have runnable sections of python if you want to experiment with them
- Exercises – Vlab – Make sure you do the exercises
  - This is the one most important tip for success
  - The computer provides feedback to guide you during the exercises
  - If you don't understand something in an exercise, please post to piazza and/or ask in class – Questions are essential and you do everyone else a service by asking them
- Online textbook – Information to support lectures and activities – Make sure you read the weekly sections.
- Piazza discussion board – Q&A

Assessment

- Homework 1 – formative
- Homework 2 – summative (counts for your mark)
- Group project
- Individual reflective statement covering your project work

A note about AI

- AI is NOT always a positive force when learning to program. It depends how you use it.
- I suggest you just use google search at the beginning and not AI
- Why?

- You might have the impression that ChatGPT is a better programmer than you (now)
- You are probably right
- Guess what? There are lots of humans who are also better programmers than you.
- Do you want to learn to program or just learn to copy things?

A note about AI

- There is a level of programming and data analysis that AI can’t do (and has no immediate prospect of doing)
- This is the thinking part
- This is a valuable skill

- You can’t learn it without going through the beginning phase where ChatGPT (and other humans) are better than you
- No change – it has always been this way
- There is no substitute for learning it yourself and that means doing it yourself

A note about AI

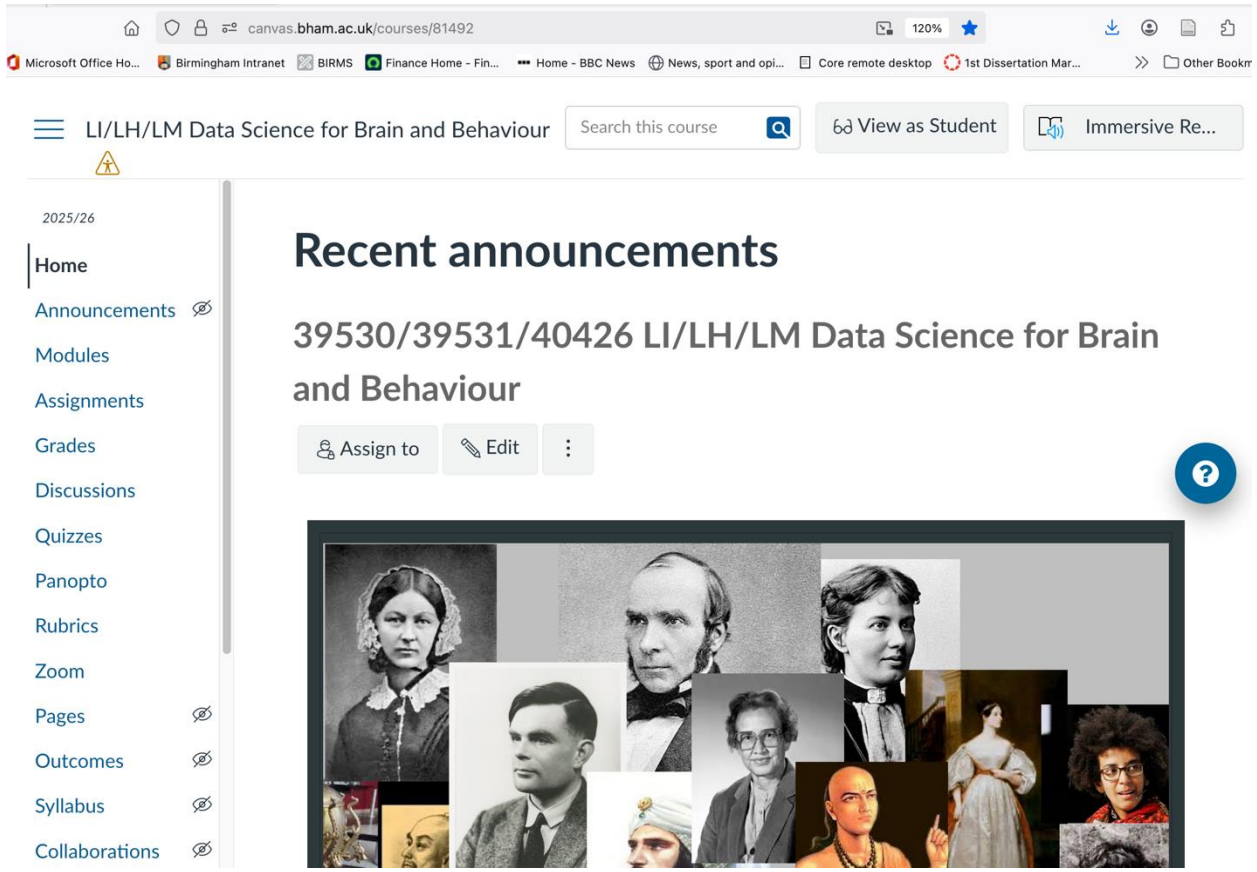
- I cannot determine how you use the course
- If you want a valuable skill, you will do things yourself
- You might be able to get by by doing very little work and copying ChatGPT. Many have tried. Some have failed. You won't know until it is too late if your strategy has worked
- If you adopt the AI-dependency route you will have a 'skill' that is ***guaranteed*** to have ZERO value
- Who will hire you when they can hire ChatGPT

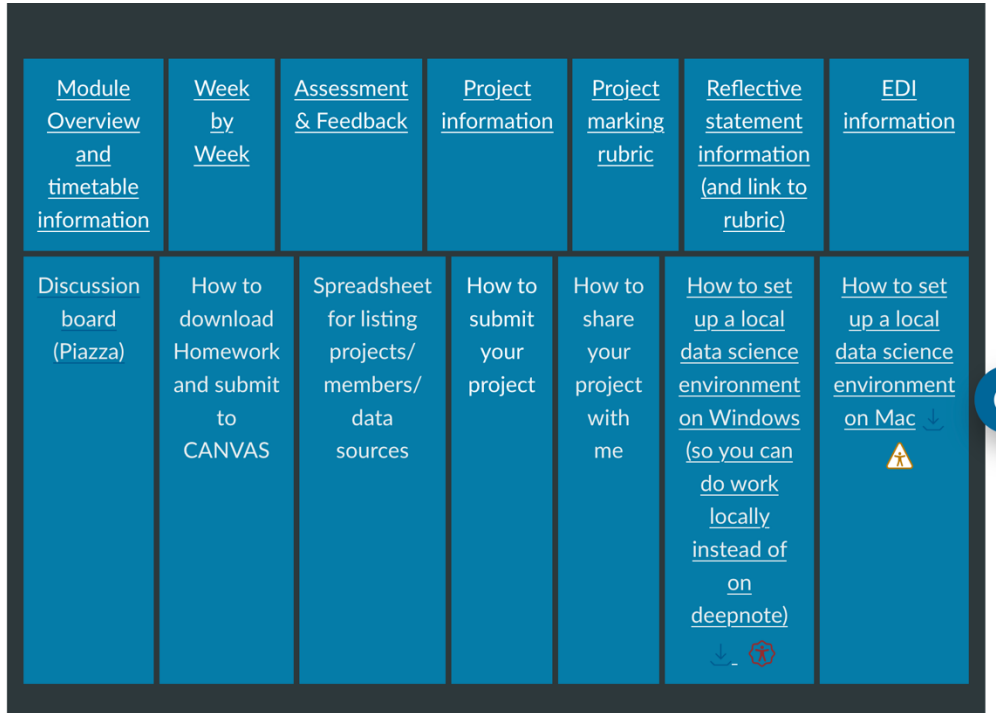
A note about AI

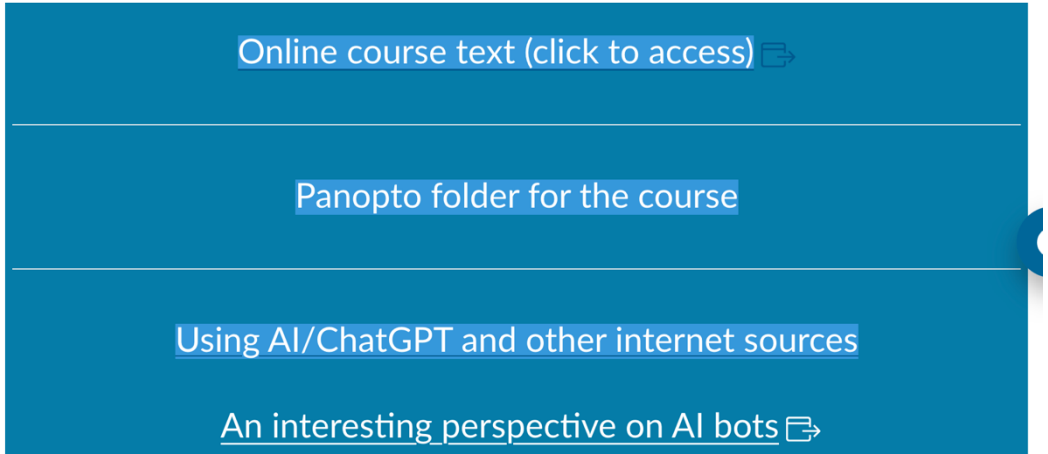
- There is lots of good information for programmers on the internet
- Stackoverflow is one site you will frequently see
- Use these sources, but use them to learn, not to copy

- Our assessments (e.g. projects) are much more weighted to understanding now, and less to coding. Coding is just a basic level on the way to understanding. We will ask you to explain code and also to explain what data mean.

Guide to the CANVAS page:
https://canvas.bham.ac.uk/courses/81492





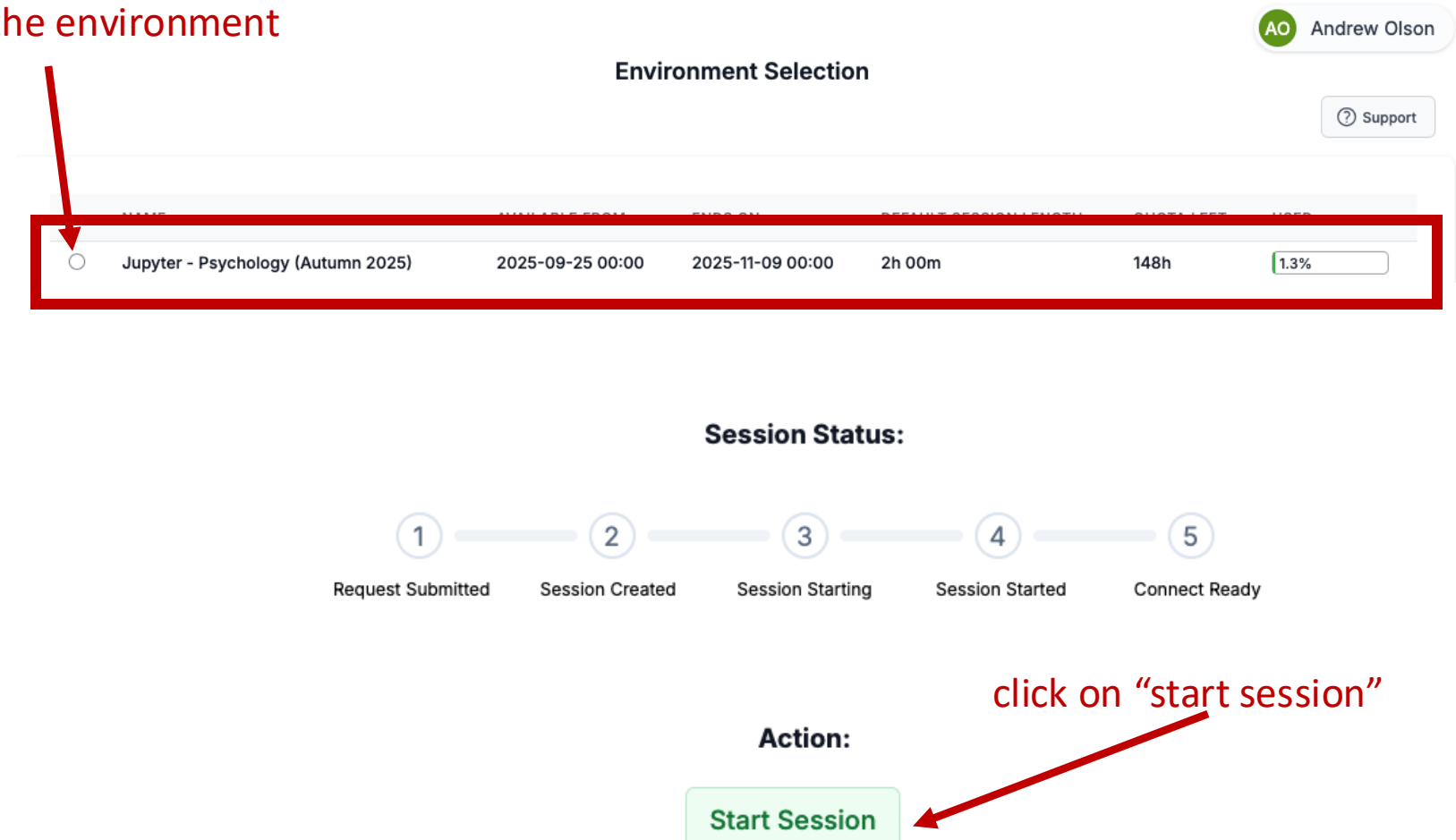


Accessing VLab

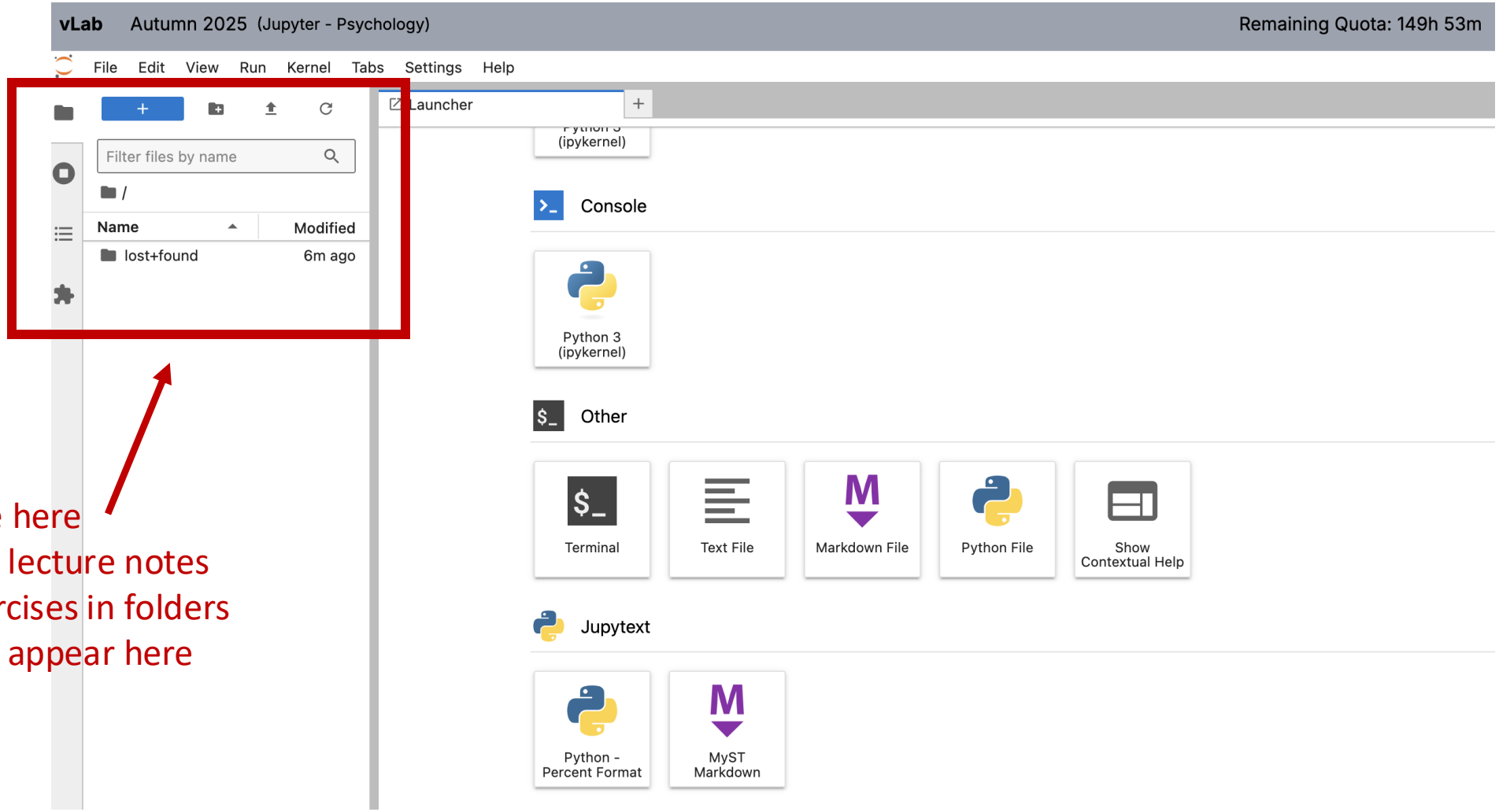
- You should be able to sign in to Vlab with your normal login. Try it.
  - https://vlab.oc1.aws.cs.bham.ac.uk/
  - This gives you a ‘personal space’ to:
    - Do exercises
    - Do your homework
    - Do your projects
    - View lectures (and run examples from lecture)
  - You need to copy exercises from a course repository to your personal space in order to work on them
  - We will set that up together today.
- If you can’t log in using your UoB login, don’t panic
  - There are always some people who are not on the course lists for one reason or another.
  - Let me know your email address and I will get you put on
  - You will not be falling far behind.

Type the link in your browser (or click on it)
You will see something like this…

Select the environment



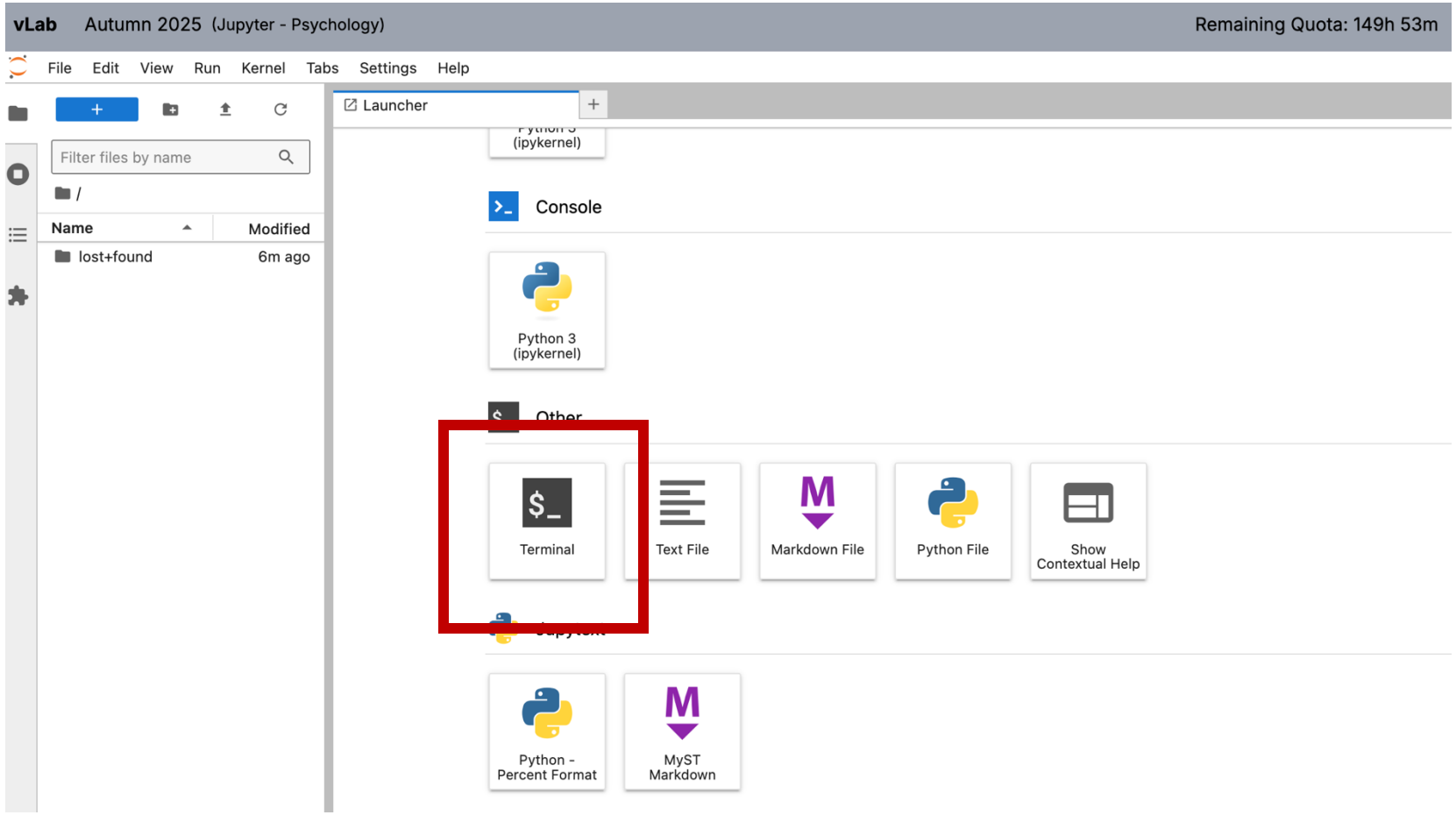
When the environment starts you will see:



navigate here
Look for lecture notes
and exercises in folders
that will appear here

Setting up for getting exercises and notes

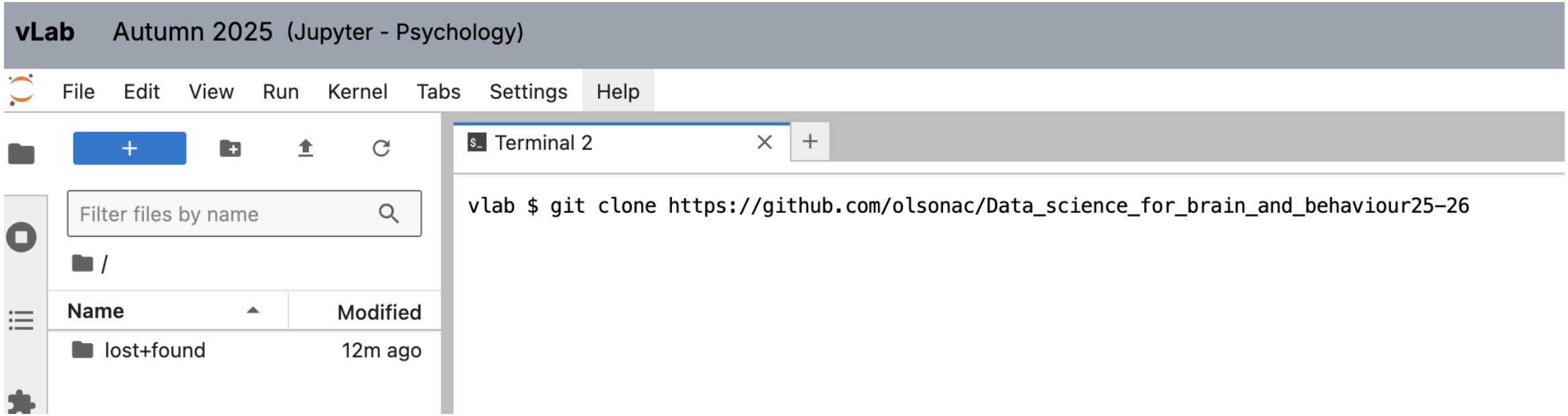
- We use a popular system for sharing code – ‘git’ and ‘github’
- We only need to set up once
- Start a terminal – Click on the ‘terminal’ icon



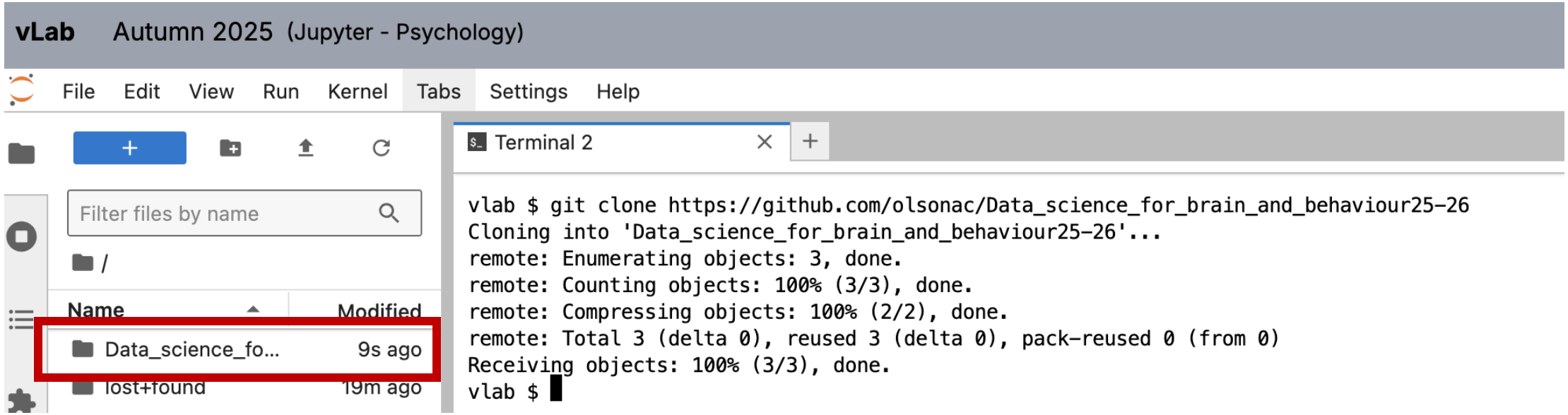
Type the ‘git clone’ command to connect to the ‘repository’ where files are stored on the cloud

- Copy the command below and paste into your terminal

git clone https://github.com/olsonac/Data_science_for_brain_and_behaviour25-26.git



The result should match this:



Notice that you now have a folder for the course. This is where exercises and lecture notes will appear

We only do that setup step once – from now on, each week…

- Start a terminal
- type 'git pull'
- new material will be added to your course folders
- e.g.

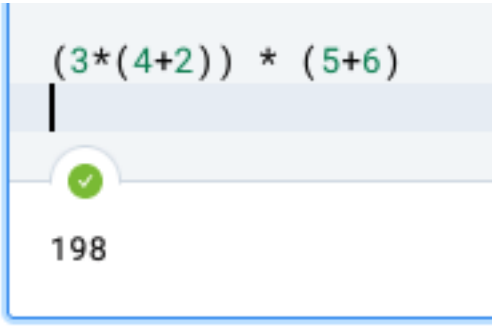
```
vlab $ git pull
remote: Enumerating objects: 13, done.
remote: Counting objects: 100% (13/13), done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 12 (delta 4), reused 12 (delta 4), pack-reused 0 (from 0)
Unpacking objects: 100% (12/12), 53.18 KiB | 6.65 MiB/s, done.
From https://github.com/olsonac/Data_science_for_brain_and_behaviour25-26
   ba4ebee..82598af  main       -> origin/main
Updating ba4ebee..82598af
Fast-forward
 Exercises/dracula/.Rhistory                                            |   0
 Exercises/dracula/.ipynb_checkpoints/dracula-checkpoint.ipynb          | 365 +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 Exercises/dracula/.ipynb_checkpoints/dracula_template-checkpoint.Rmd   | 269 +++++++++++++++++++++++++++++++++++++++++++++
 Exercises/dracula/.ipynb_checkpoints/dracula_template-checkpoint.ipynb | 521 ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 Exercises/dracula/dracula.Rmd                                          | 258 ++++++++++++++++++++++++++++++++++++++++++++
 Exercises/dracula/dracula.ipynb                                        | 379 ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 Exercises/dracula/play_arrow.png                                       | Bin 0 -> 13663 bytes
 7 files changed, 1792 insertions(+)
 create mode 100644 Exercises/dracula/.Rhistory
 create mode 100644 Exercises/dracula/.ipynb_checkpoints/dracula-checkpoint.ipynb
 create mode 100644 Exercises/dracula/.ipynb_checkpoints/dracula_template-checkpoint.Rmd
 create mode 100644 Exercises/dracula/.ipynb_checkpoints/dracula_template-checkpoint.ipynb
 create mode 100644 Exercises/dracula/dracula.Rmd
 create mode 100644 Exercises/dracula/dracula.ipynb
 create mode 100644 Exercises/dracula/play_arrow.png
```

Our tools: Do I need to install python on my own machine? Jupyter notebooks?

- Most of our work is done through vlab.
- Still, you might want to have a local copy of python and jupyter notebooks (e.g. for working offline)
- If you are interested I can help with that – contact me

Jupyter notebook – how does it work?

- VLab is run through a web browser on your computer
- When you run a cell, it doesn't do the computation on your machine.
- When you click on "run" or <cmd><enter>
  - The browser sends the text you have typed in to the vlab machine somewhere in the networld (maybe California). The vlad machine is running your "kernel".
  - The "kernel" is what executes your commands
  - the kernel does the arithmetic and sends back the answer
  - The answer comes over the web to your browser.
  - Your browser gets the message and shows it to you

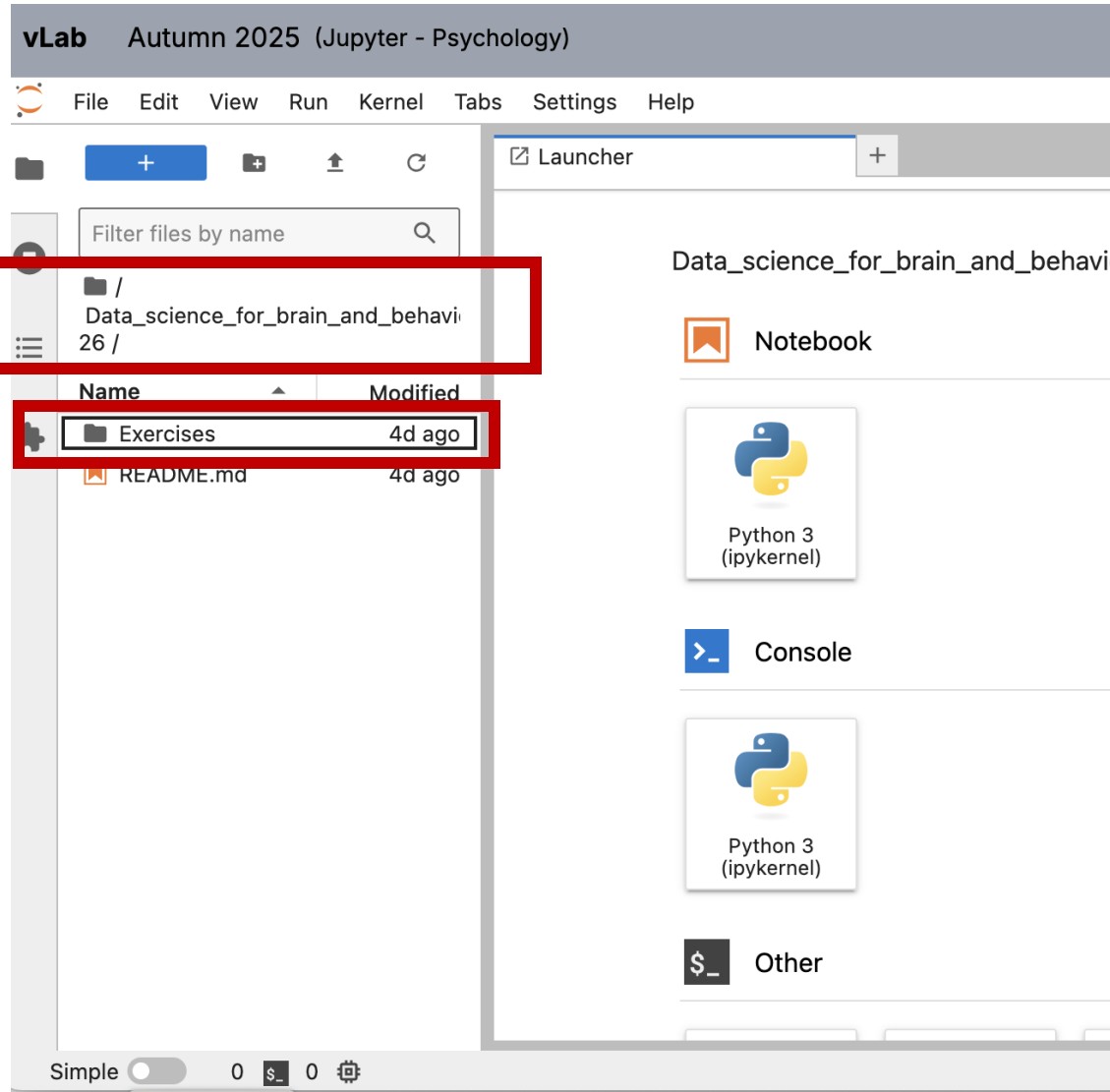


Why do I need to know that?

- Two reasons
- 1) VLab doesn’t work without a web connection
  - That may have been obvious already
- 2) There are two processes looking after you.
  - One on your machine (your web browser).
  - One out in the cloud where your kernel is running

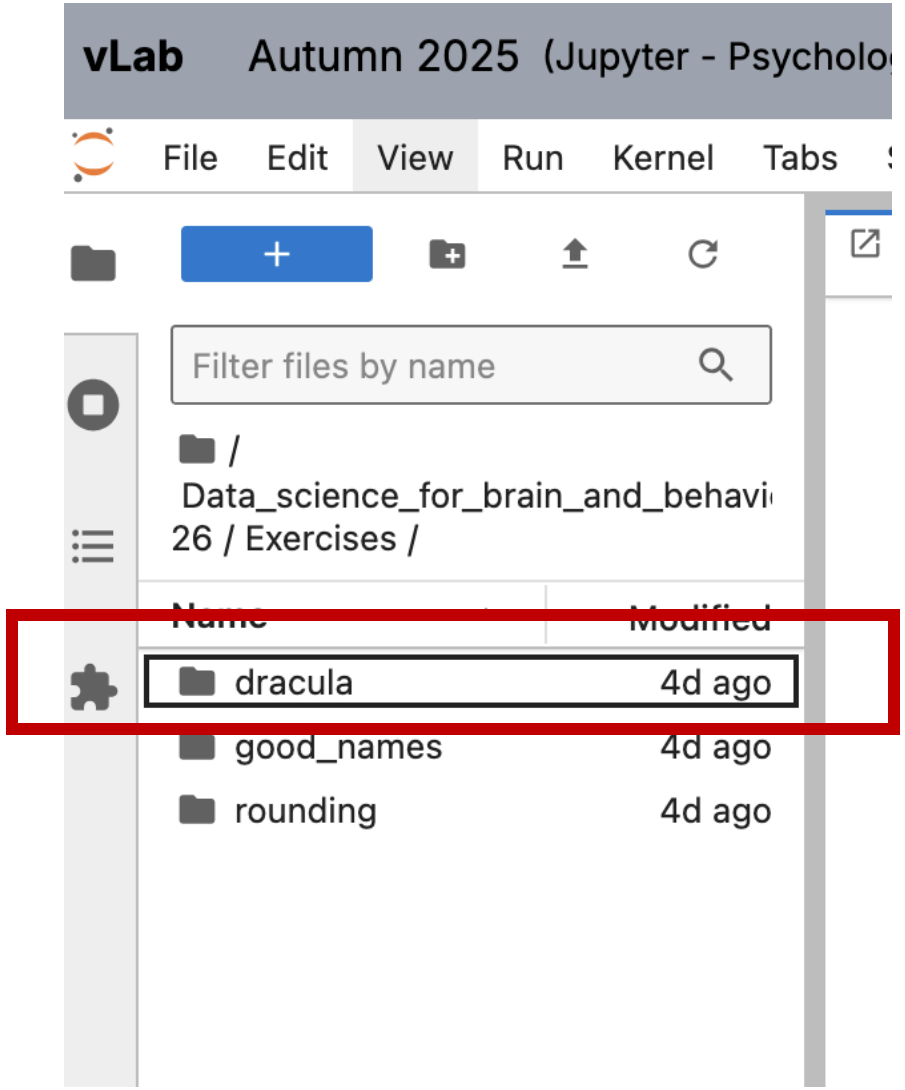
Your first exercise: Dracula

- Click on the Data_science_for_brain_and_behaviour_25-26 folder
- Click on the exercises folder underneath this



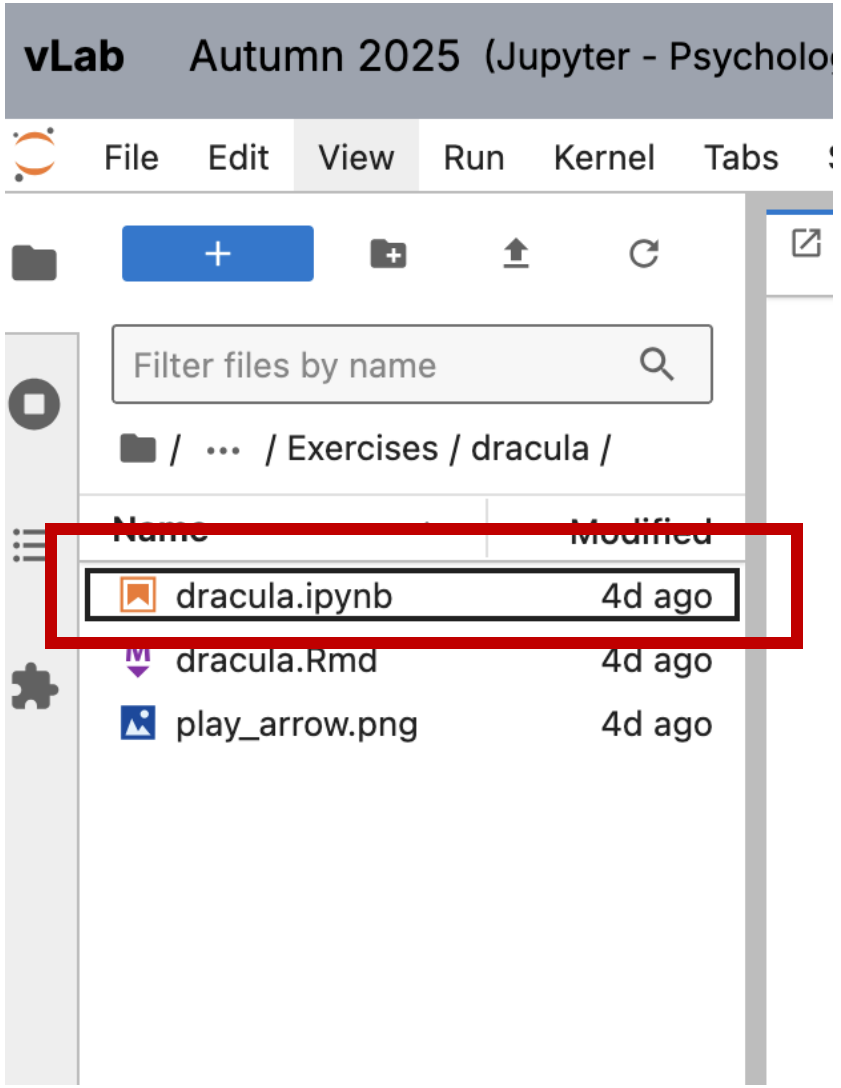
click on the ‘Dracula’ folder

- Dracula is our first exercise

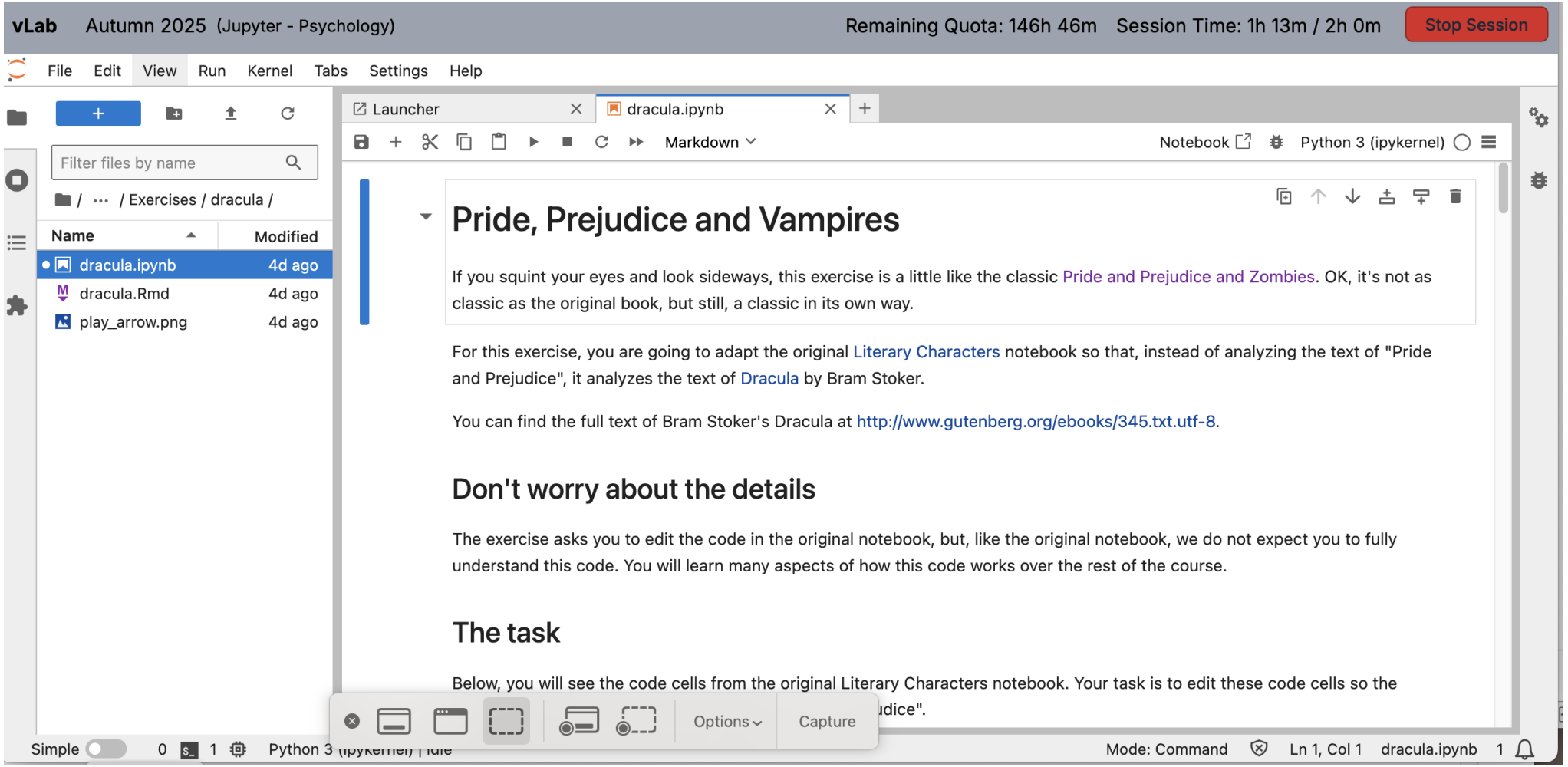


Double click on the ‘dracula.ipynb’ file

- .ipynb files are jupyter notebooks
- We will be using this type of file for our work
- Double clicking on the file will open it



You should now see the notebook:



End of vlab setup

- Now you are set up to use vlab to get exercises and notes
- New material will appear each week
- Remember to do ‘git pull’ before starting your work
- Then click on folders to find the notebook you need
- We will introduce notebooks in a separate step

Piazza – For questions/discussions

- Ask questions!
- There are no silly questions
- sign up to Piazza using this link:

- https://piazza.com/bham.ac.uk/fall2025/3953039531404 26/home

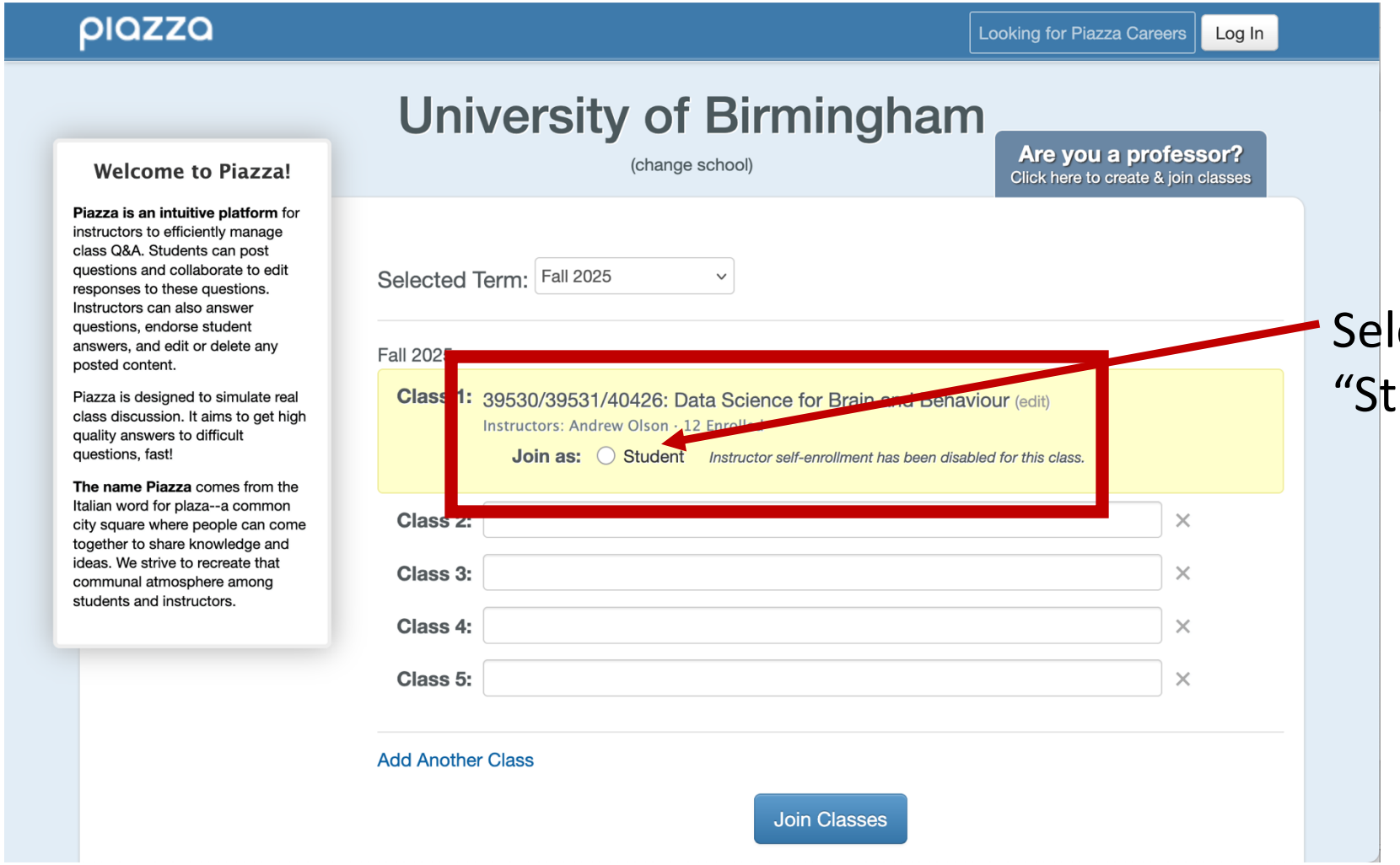
There are no silly questions about DS

- If you have a question, several other people have it too
  - Guaranteed

- Questions help me
  - Sometimes I don’t know how people are thinking about an issue
  - “Ooooh, I see how you’re thinking about it”
  - That can be very helpful

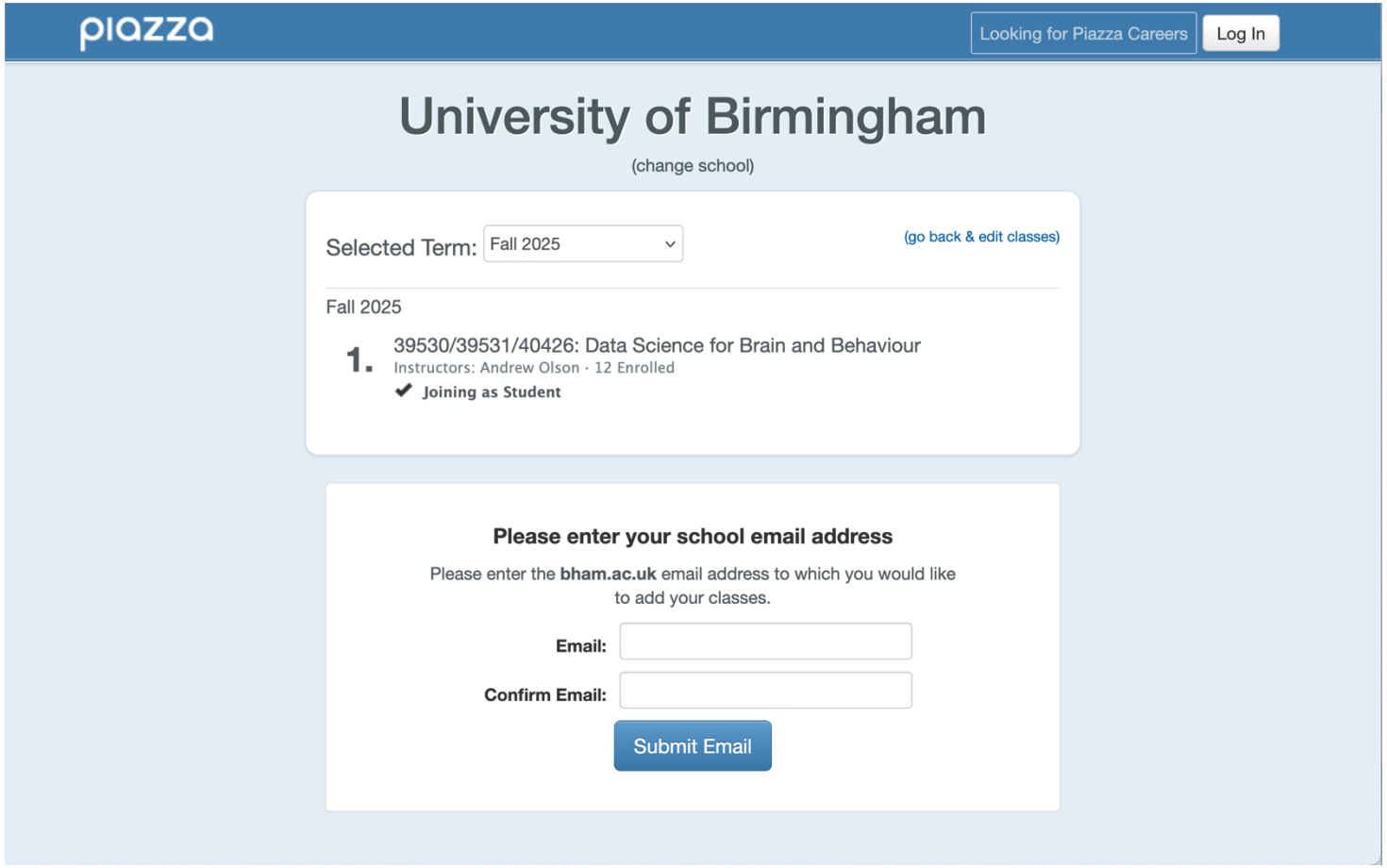
Signing up for piazza – enter your UoB email address

- Signup page:



Select "Student"

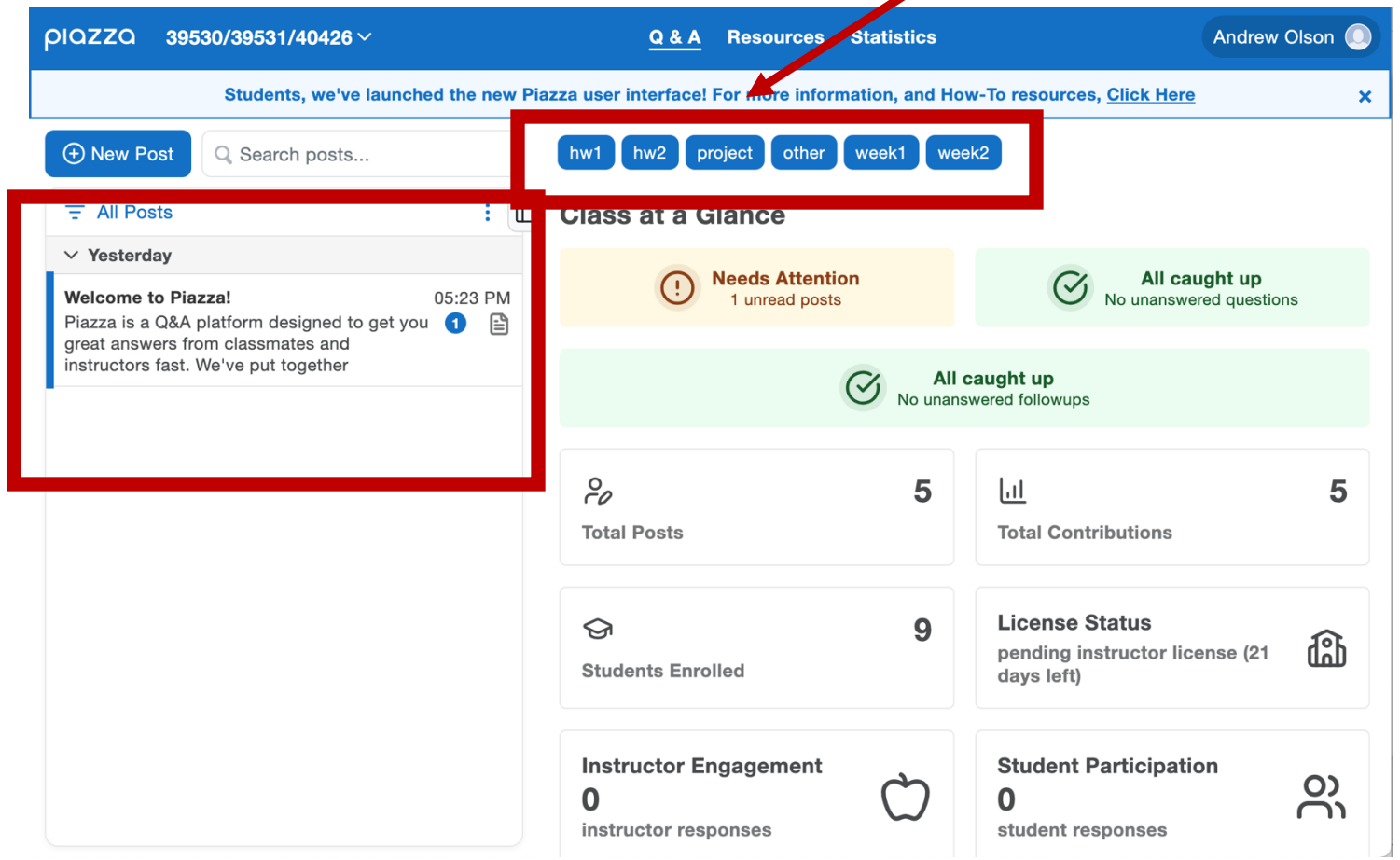
sign up with UoB email (needs to be UoB)



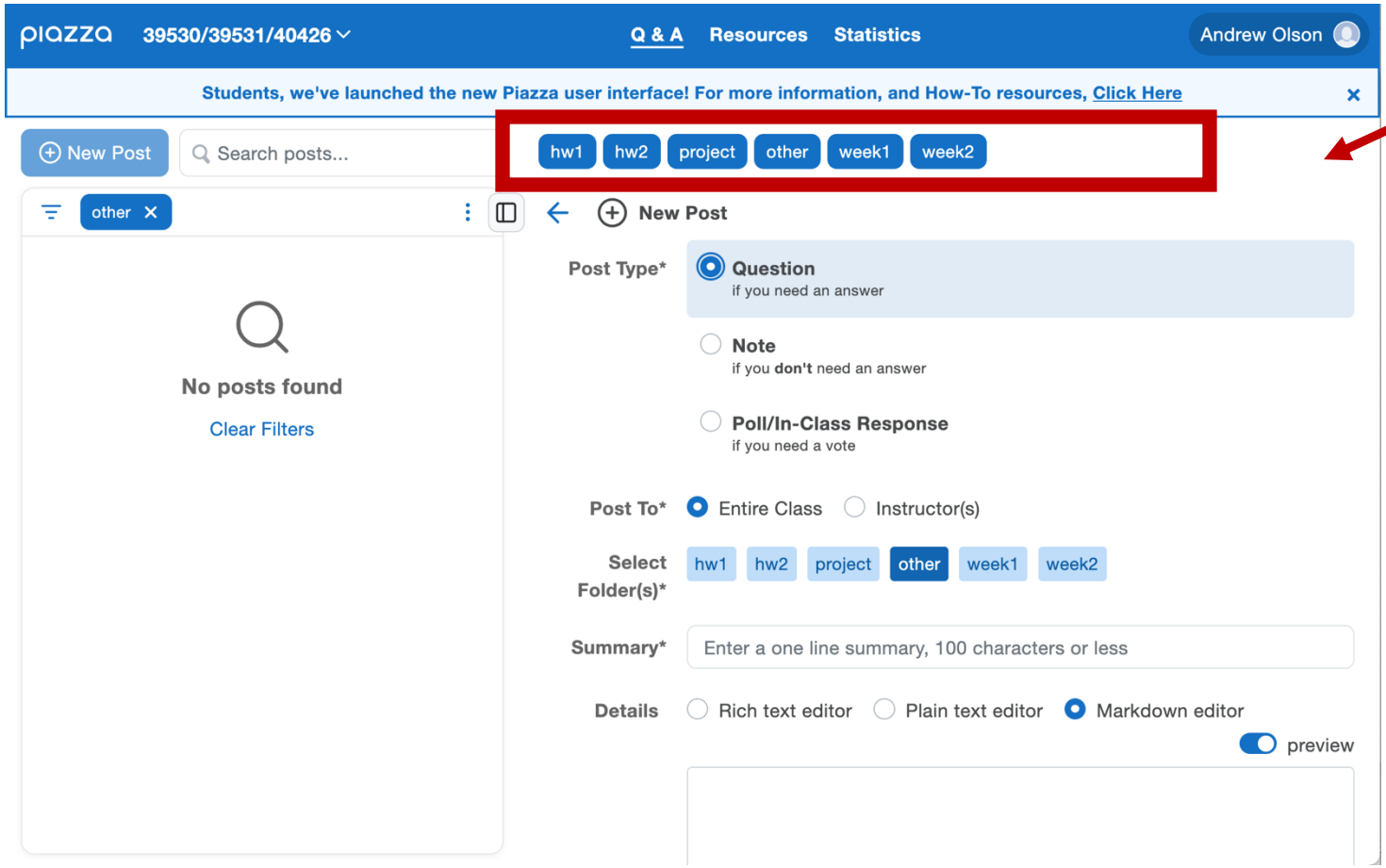
Piazza course page

folders with categories of questions

questions listed here



Ask a question

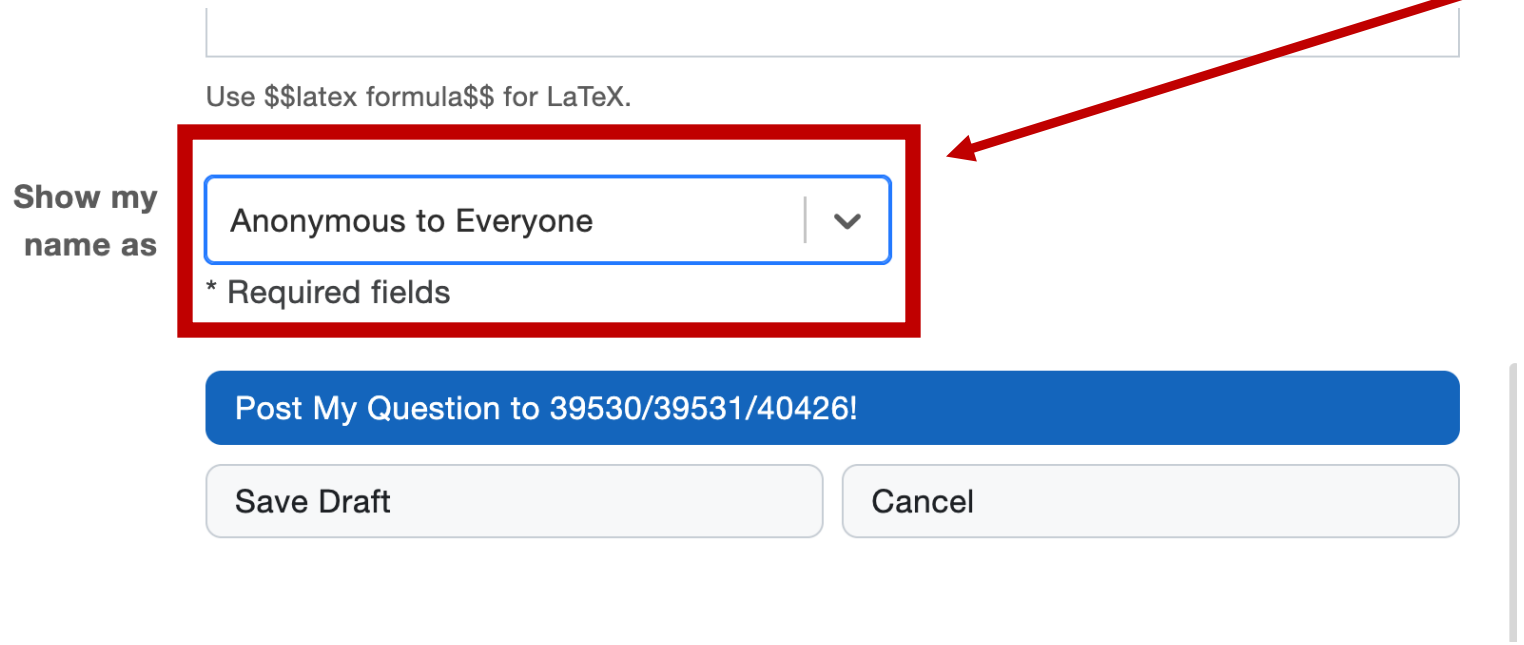


Choose a folder that fits your question

To make it anonymous

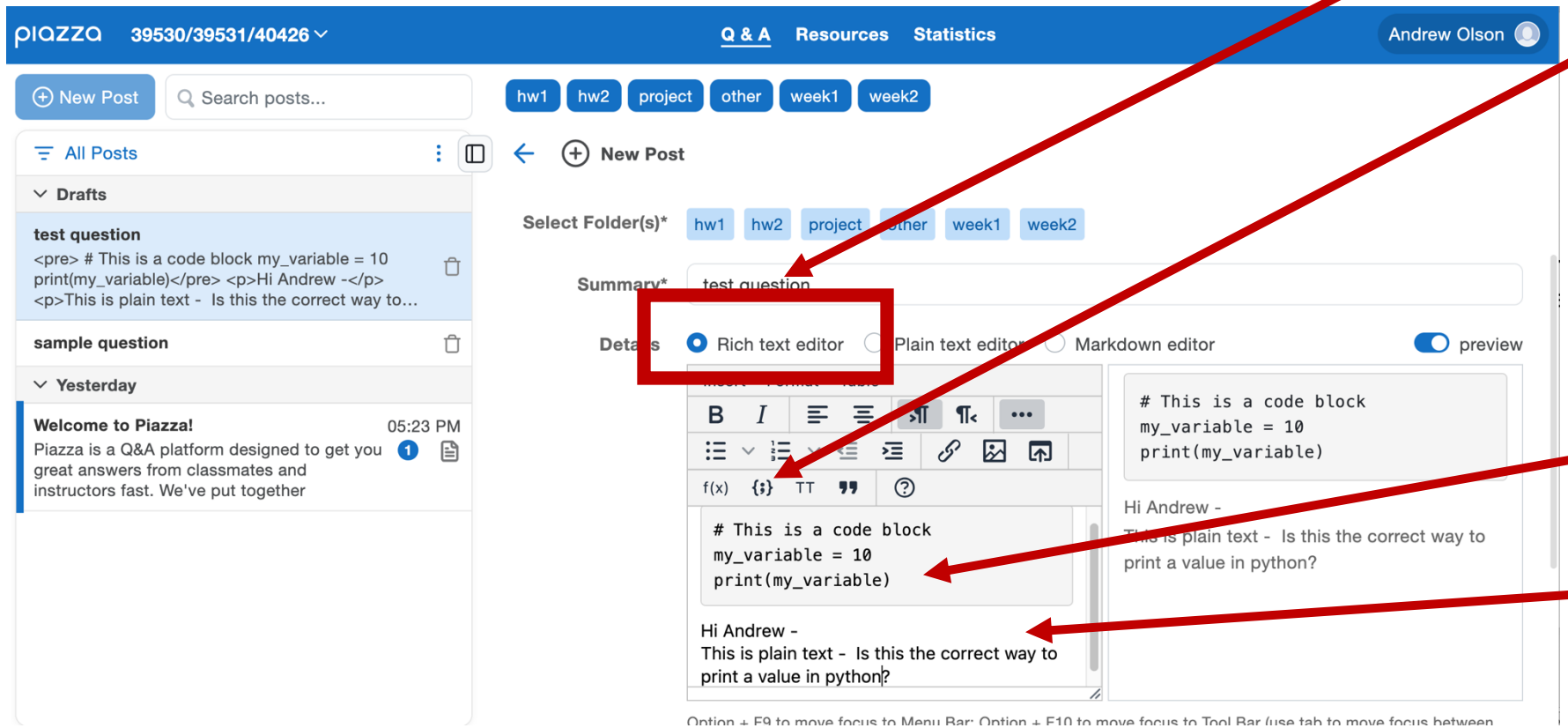
- See the bottom of the post section:

Change this to be anonymous



Piazza allows you to have code and text blocks.

- This can make your question easier to read



1) Use the rich text editor

2) Click here to create a code block

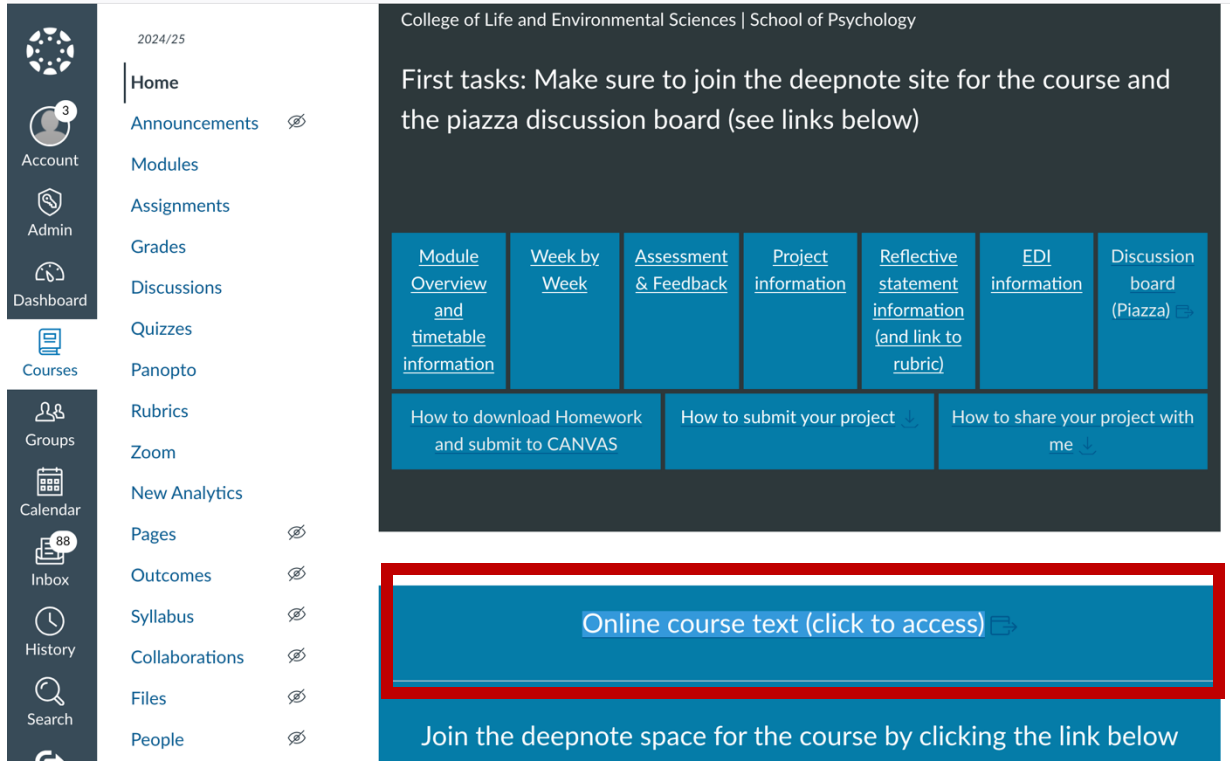
3) the grey box is code

The white part is text

The CANVAS page and link to the online course text

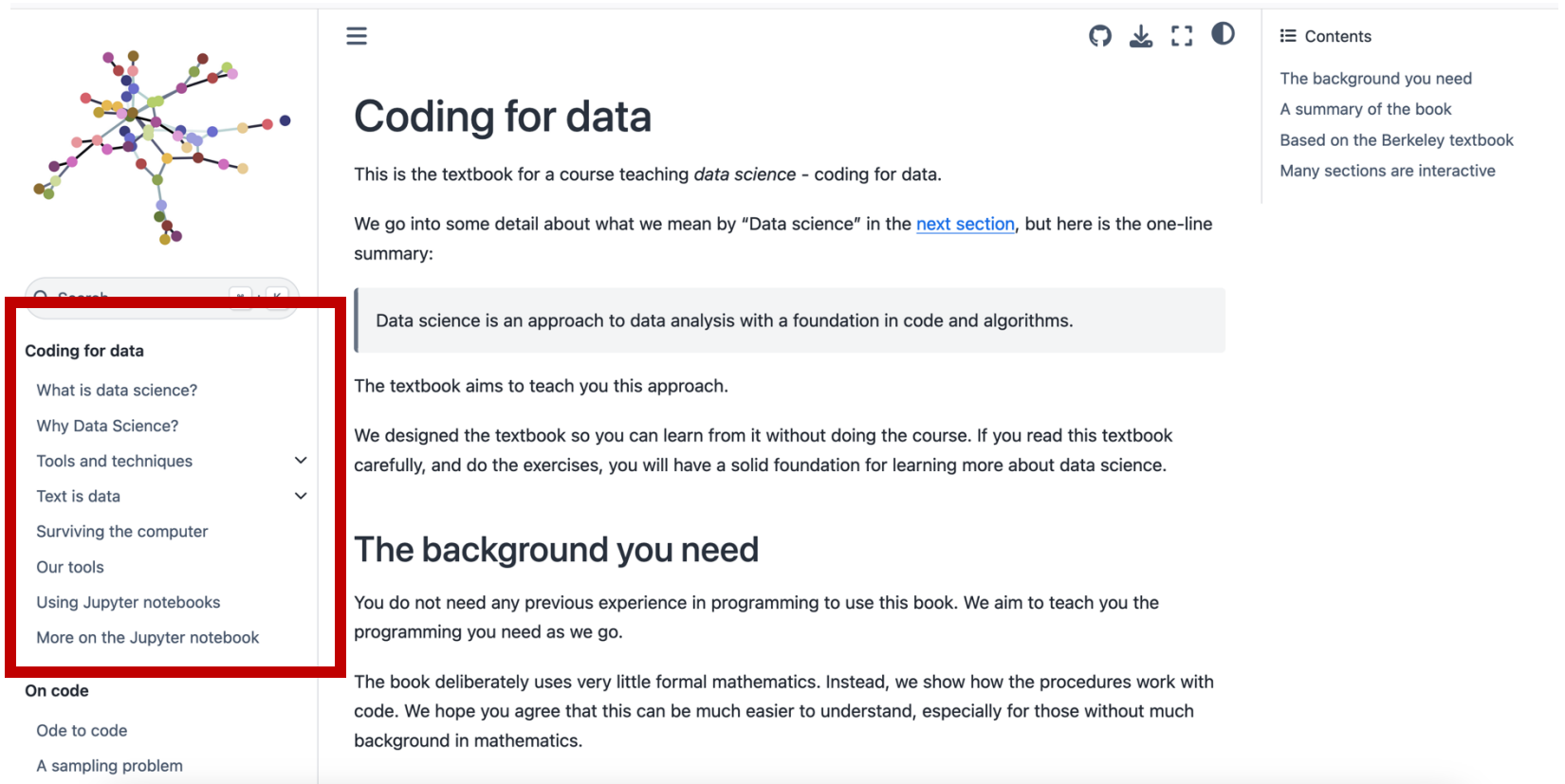
- Link on the CANVAS page – also here:
- https://olsonac.github.io/textbook/intro.html

- Assessment and feedback information
- Link to the e-textbook
- Link to the piazza discussion board
- How to submit homework (TBA)



Course text page

- Read the entries in the first section: “Coding for data”



Student organized Whatsapp group

- Students have organized a Whatsapp group in the past
- If you want to do that I will distribute the link
  - Send me an email and I will put it on the CANVAS page
- I will not participate in the whatsapp group
  - no lurking

Surviving the computer

- Good advice

Have a piece of paper and a pen next to you

When *working* on the computer, have a piece of paper and a pen next to you on your desk. When you notice that you don't fully understand an error or a task on the computer, stop, move the laptop out of the way, and write down your problem on a piece of paper. Give yourself some time to reflect on what the problem is, and how to solve it. Only then, you can return to the computer and try your solutions, or do more research. You will find that disengaging from the computer is a) difficult and b) very productive in releasing you from the various mental traps that it is easy to fall into when you get stuck on the computer.

- Take a break. Get away from your machine. Think about something else. Then come back
- Writing things down can also help you to think in a different way (e.g. diagrams)

The first exercise: Dracula Plotting the classics

- You don't have to understand the code (yet)
- Checks that you can open a notebook and run the sections
- Has some examples of what you can do…
  - Get text from the internet
  - break it up into chunks (words, chapters)
  - Count how often the principle characters are mentioned
  - Make a summary table by chapter (of cumulative numbers - important for interpretation [not numbers for each chapter])
  - Plot the summary
  - See what you can learn from the summary plot
    - e.g. who is the principle character
    - who is the "love interest"
    - when are secondard characters important

Summary

- We’ve looked at VLab
- We’ve seen the canvas page
- We’ve seen how to access the course text
- We’ve seen what jupyter notebooks look like and how to open them
- We’ve seen piazza – Ask questions!

- Dracula – Our first exercise
  - Plotting the classics (you don’t have to understand the code parts – we don’t expect that yet)
- Just see what is possible, getting text from the internet, breaking it up, plotting the results
- See what you can understand about a book from even a few lines of python code
- Practice understanding what a graph is showing you (first identify the parts clearly)