Scientific Programming Best Practices

Week 8
Data Science Workshop for
NGA LTER REU Students

Review of Last Week

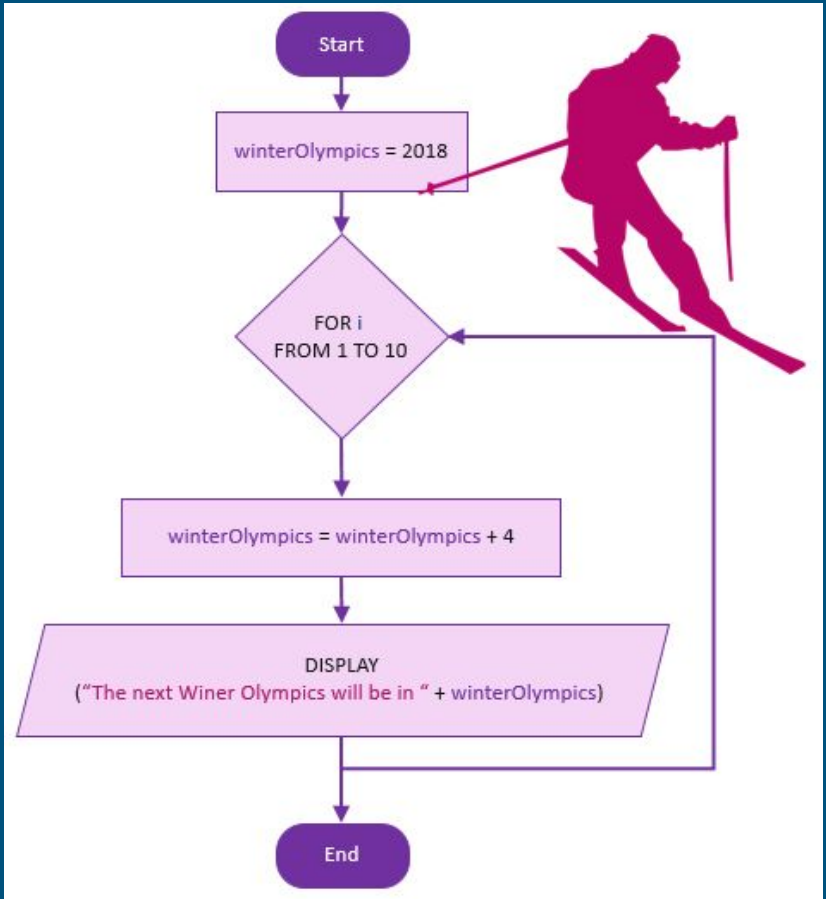
- Learned about GitHub
- Used ssh keys to tell GitHub to trust our computers
- Started a new repository and copied some stuff into it

Goals for this week

- Talk about coding
  - Normalize whatever problems you are having.
- Chat about data stuff
- How have these topics applied to your work?

Why Coding?

- Code reduces errors
- Documented code creates a record of processing steps
- Other people's code = less work!
  - Use standard libraries whenever possible - don't reinvent



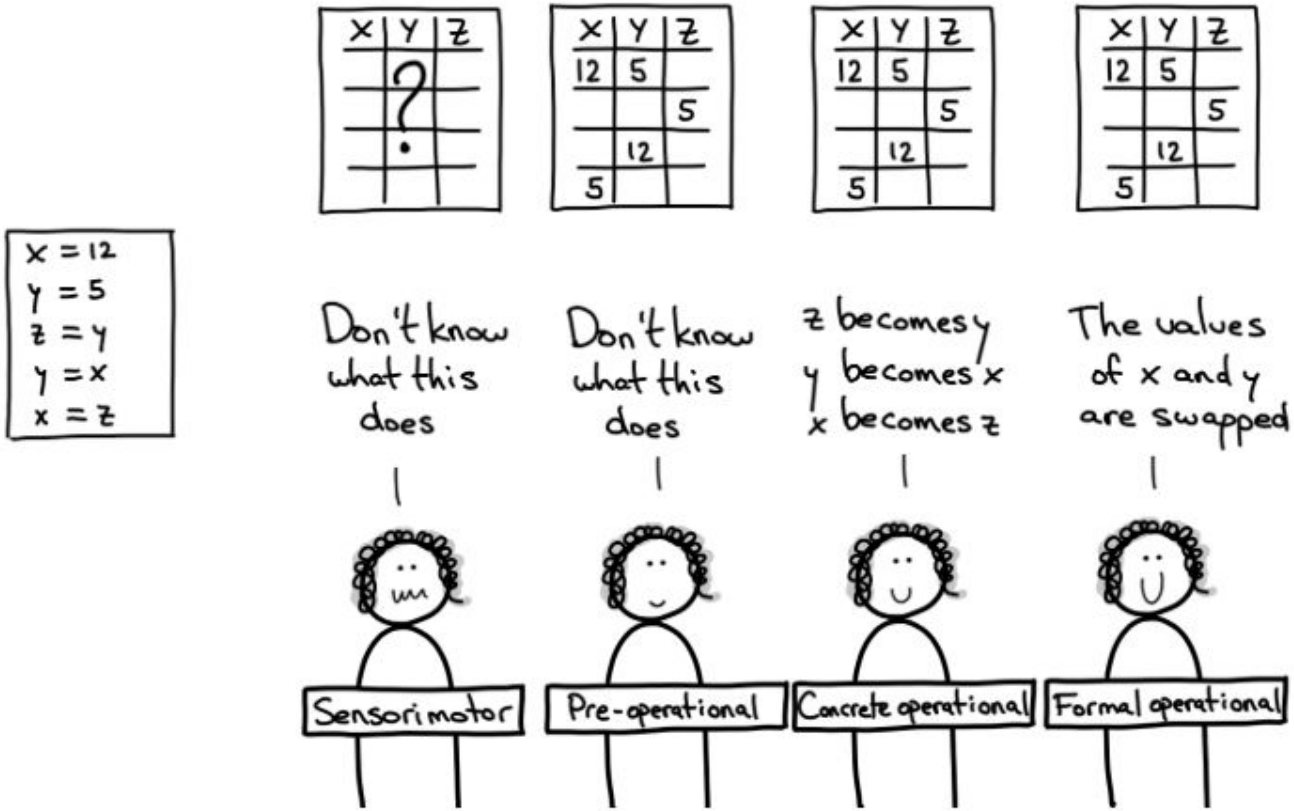
How Coding?

- Eventually, you will try coding
- Learn a new language to do a different kind of task

How do you learn to code?

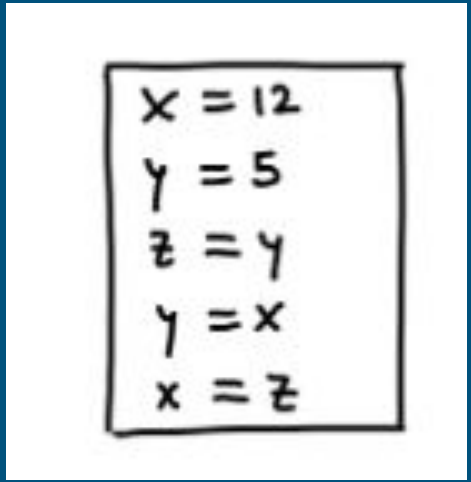
- Not where do you go for lessons,
- But what is the process? What happens in your brain?

Figure 13.1 Overview of the four different neo-Piagetian levels for programming.



Credit: [The Programmer's Brain]

Cognitive Load

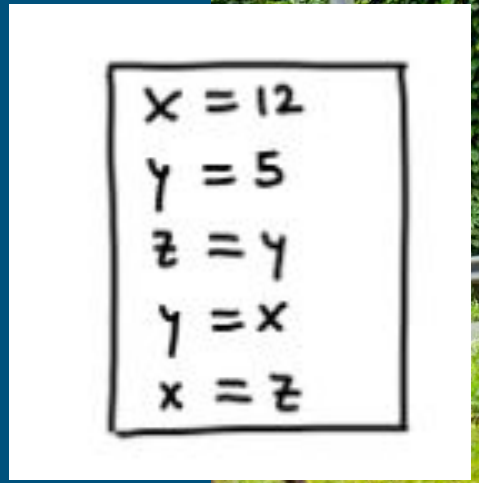


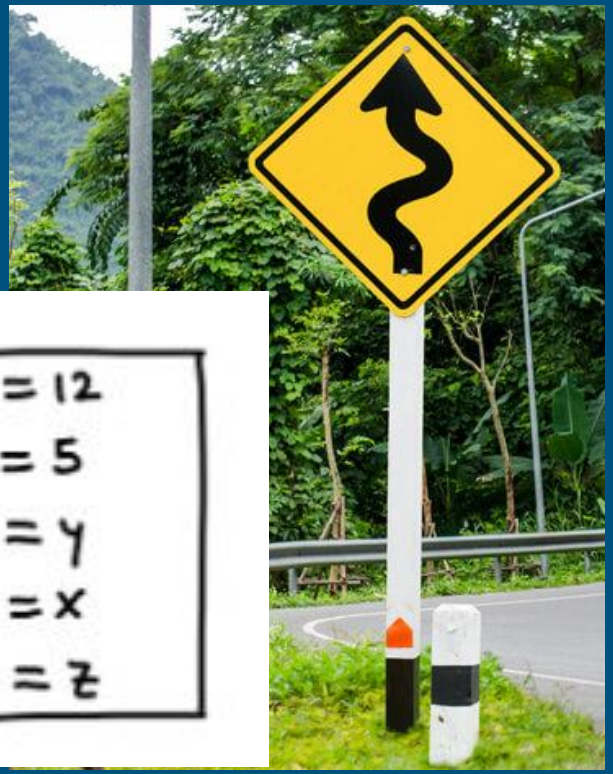
Credit: Rita Sa

Comments = write code for people

Comments should document design and purpose.
Not every step.

“Swap the values of x and y using z as temporary storage”

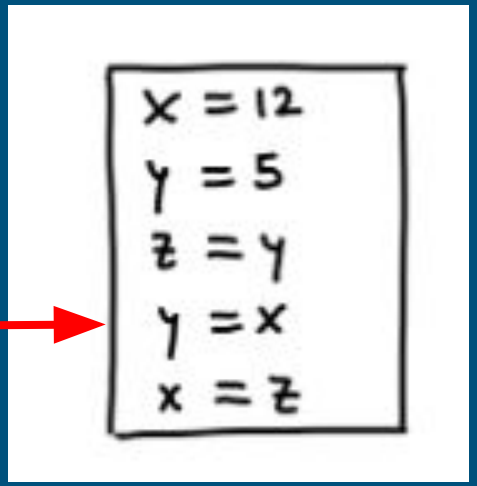




Good names make code “self-documenting”

Avoid cryptic variable names
Make them descriptive, even if long

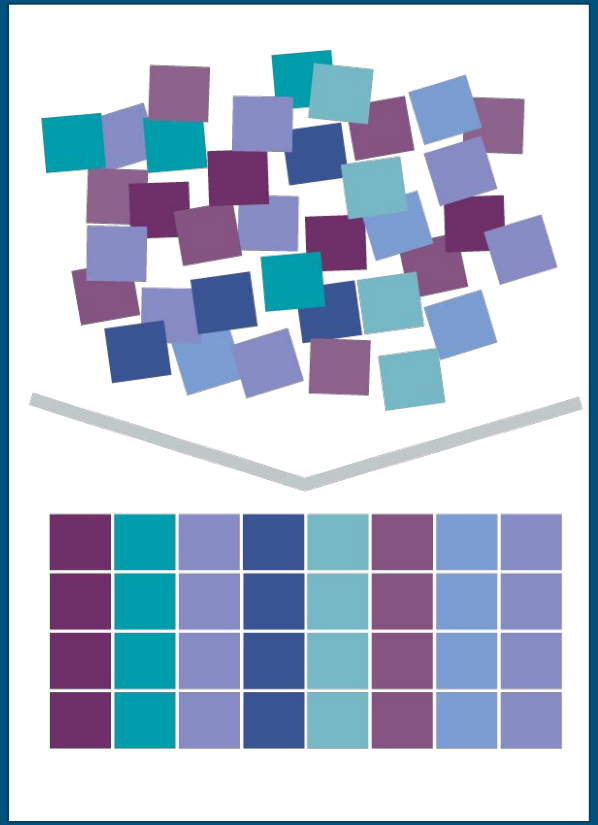
second_value = first_value



Modularize the code

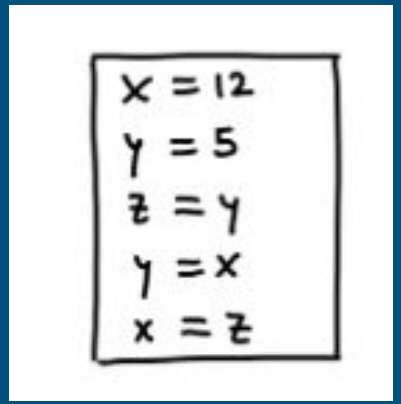
If the task is repeated or forms a logical unit, make a function.

swap_values(x, y)

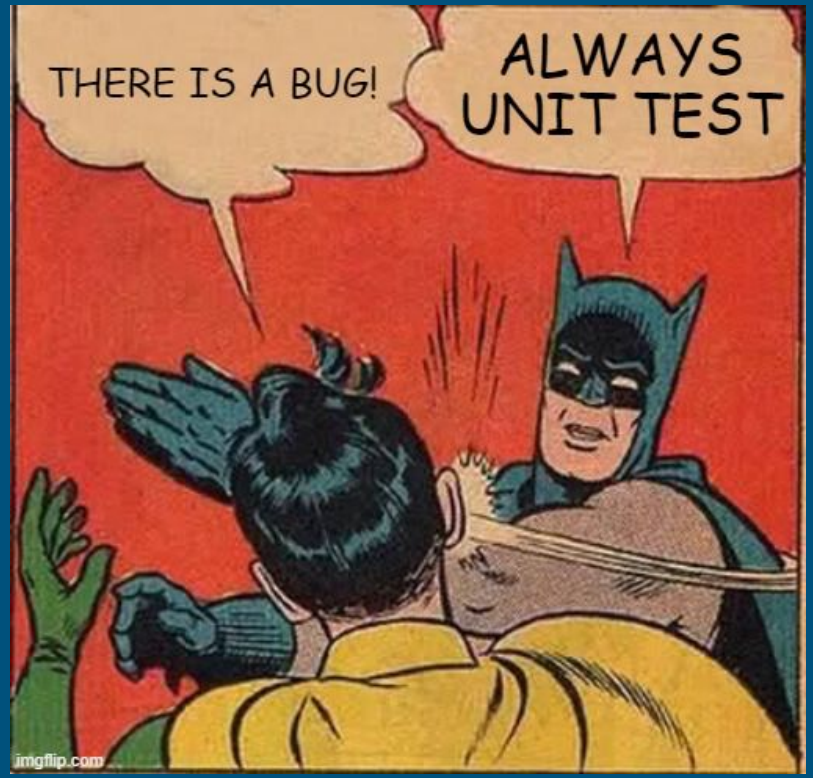


Unit Testing

Test the individual modules to make sure they give the answer you expect.



assert x == 5




KEEP
CALM
AND
TELL YOUR
DUCK
KeepCalmAndPosters.com

Other bug fixing tips:

- Make small, incremental changes
- Collaborate - other eyes see problems you don’t
- Keep records of your changes = version control !

In Review

1. Use comments to document the approach
2. Choosing good names creates more documentation
3. Break the larger task into smaller units
4. Test each unit
5. Expect bugs (Keep calm and tell your duck)

That’s it!

Resources

- Balaban G, Grytten I, Rand KD, Scheffer L, Sandve GK (2021) Ten simple rules for quick and dirty scientific programming. PLOS Computational Biology 17(3): e1008549. https://doi.org/10.1371/journal.pcbi.1008549
- Wilson G, Aruliah DA, Brown CT, Chue Hong NP, Davis M, et al. (2014) Best Practices for Scientific Computing. PLOS Biology 12(1): e1001745. https://doi.org/10.1371/journal.pbio.1001745
- Wilson G, Bryan J, Cranston K, Kitzes J, Nederbragt L, et al. (2017) Good enough practices in scientific computing. PLOS Computational Biology 13(6): e1005510. https://doi.org/10.1371/journal.pcbi.1005510
- The Programmer's Brain