Improving Your Python Programming Experience

Tools and Tricks to Make Your Life Easier

Jacob Adams, UGRC

UGIC 2022


RoofBag.com


2019 SHOP TOUR
Katz-Moses
WoodWorking

Virtual Environments via Conda

IDEs

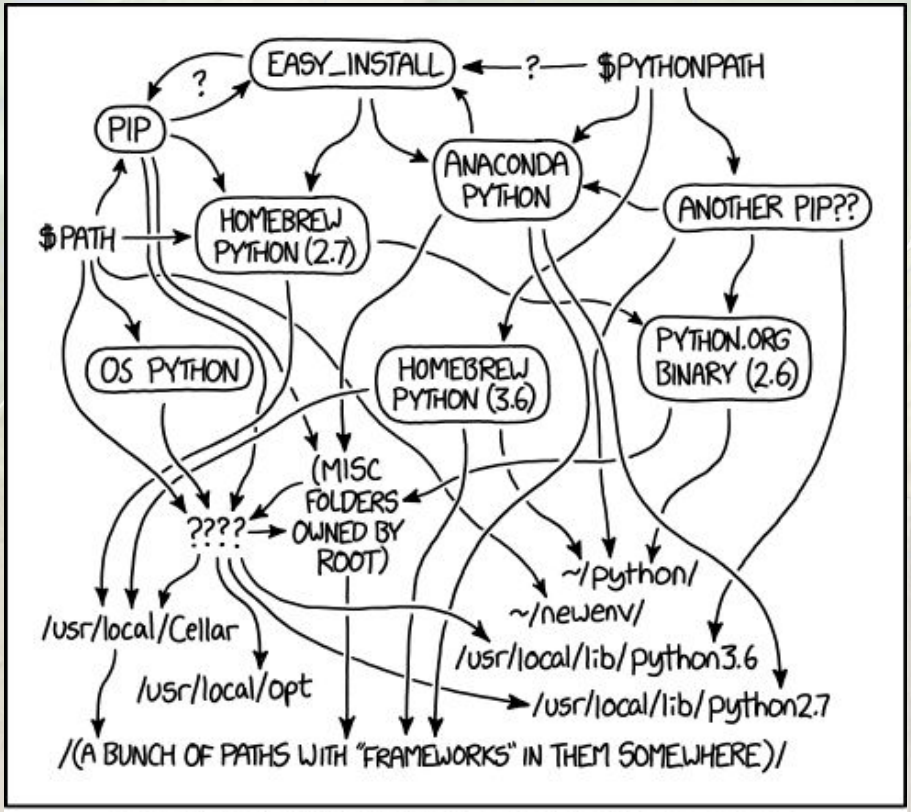
Formatting and Linting

Git & GitHub

Unit Testing


DON'T
PANIC

Managing Your Environments With CONDA



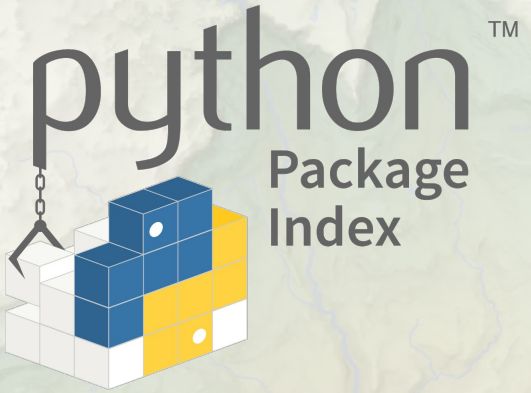
MY PYTHON ENVIRONMENT HAS BECOME SO DEGRADED THAT MY LAPTOP HAS BEEN DECLARED A SUPERFUND SITE.

```
(arcgispro-py3) c:\gis>conda env list
# conda environments:
#
base                     C:\Users\jdadams\AppData\Local\Programs\ArcGIS\Pro\bin\Python
arcgispro-py3         *  C:\Users\jdadams\AppData\Local\Programs\ArcGIS\Pro\bin\Python\envs\arcgispro-py3
auditor                  C:\Users\jdadams\AppData\Local\Programs\ArcGIS\Pro\bin\Python\envs\auditor
erapskid                 C:\Users\jdadams\AppData\Local\Programs\ArcGIS\Pro\bin\Python\envs\erapskid
gsheets                  C:\Users\jdadams\AppData\Local\Programs\ArcGIS\Pro\bin\Python\envs\gsheets
housing                  C:\Users\jdadams\AppData\Local\Programs\ArcGIS\Pro\bin\Python\envs\housing
palletjack               C:\Users\jdadams\AppData\Local\Programs\ArcGIS\Pro\bin\Python\envs\palletjack
rural                    C:\Users\jdadams\AppData\Local\Programs\ArcGIS\Pro\bin\Python\envs\rural
supervisor               C:\Users\jdadams\AppData\Local\Programs\ArcGIS\Pro\bin\Python\envs\supervisor
test                     C:\Users\jdadams\AppData\Local\Programs\ArcGIS\Pro\bin\Python\envs\test
ugic                     C:\Users\jdadams\AppData\Local\Programs\ArcGIS\Pro\bin\Python\envs\ugic
upython                  C:\Users\jdadams\AppData\Local\Programs\ArcGIS\Pro\bin\Python\envs\upython
```

Conda virtual environments: separate sandboxes for your projects

Lowes.com

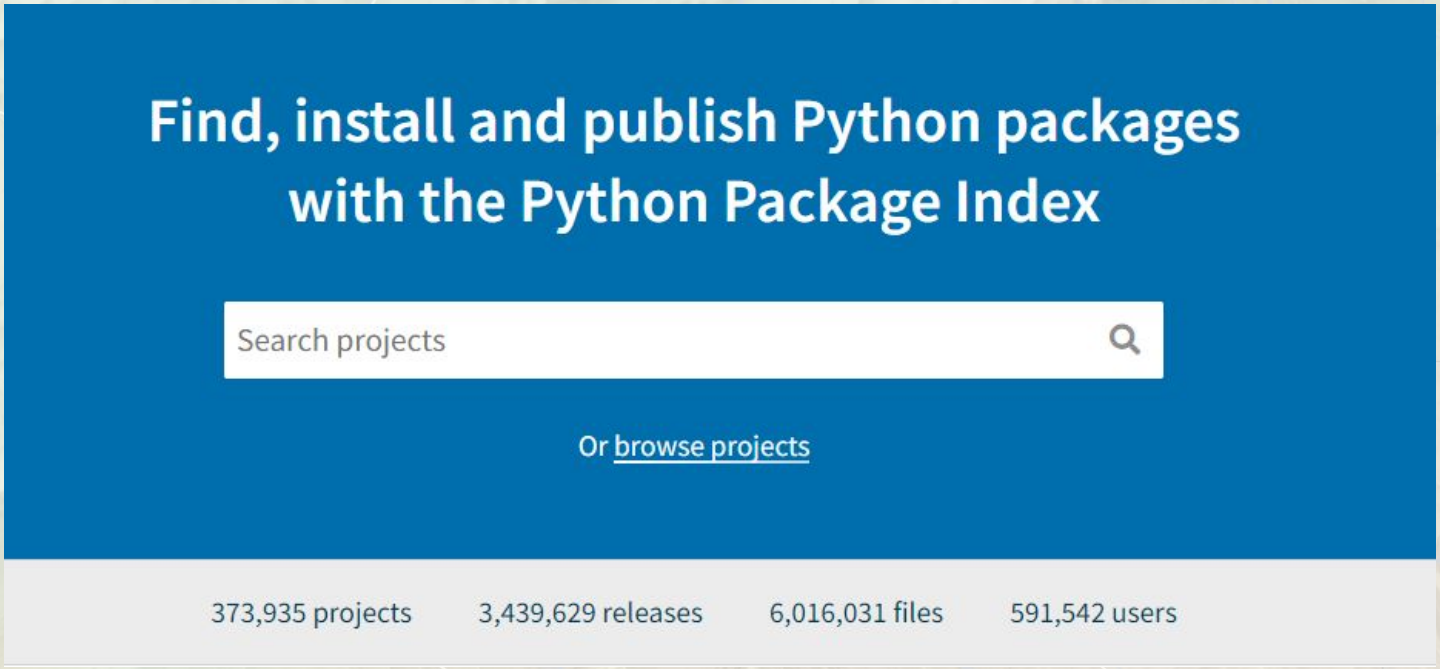
vs



- Installs Python
- Many channels
- Manages environments
- Installed with ArcGIS Pro

- Requires Python
- More packages
- Easy to automate dependencies
- Easily install local packages

Packaging Your Code



Configuration: `setup.py`

Local Installation: `pip install -e .[tests]`

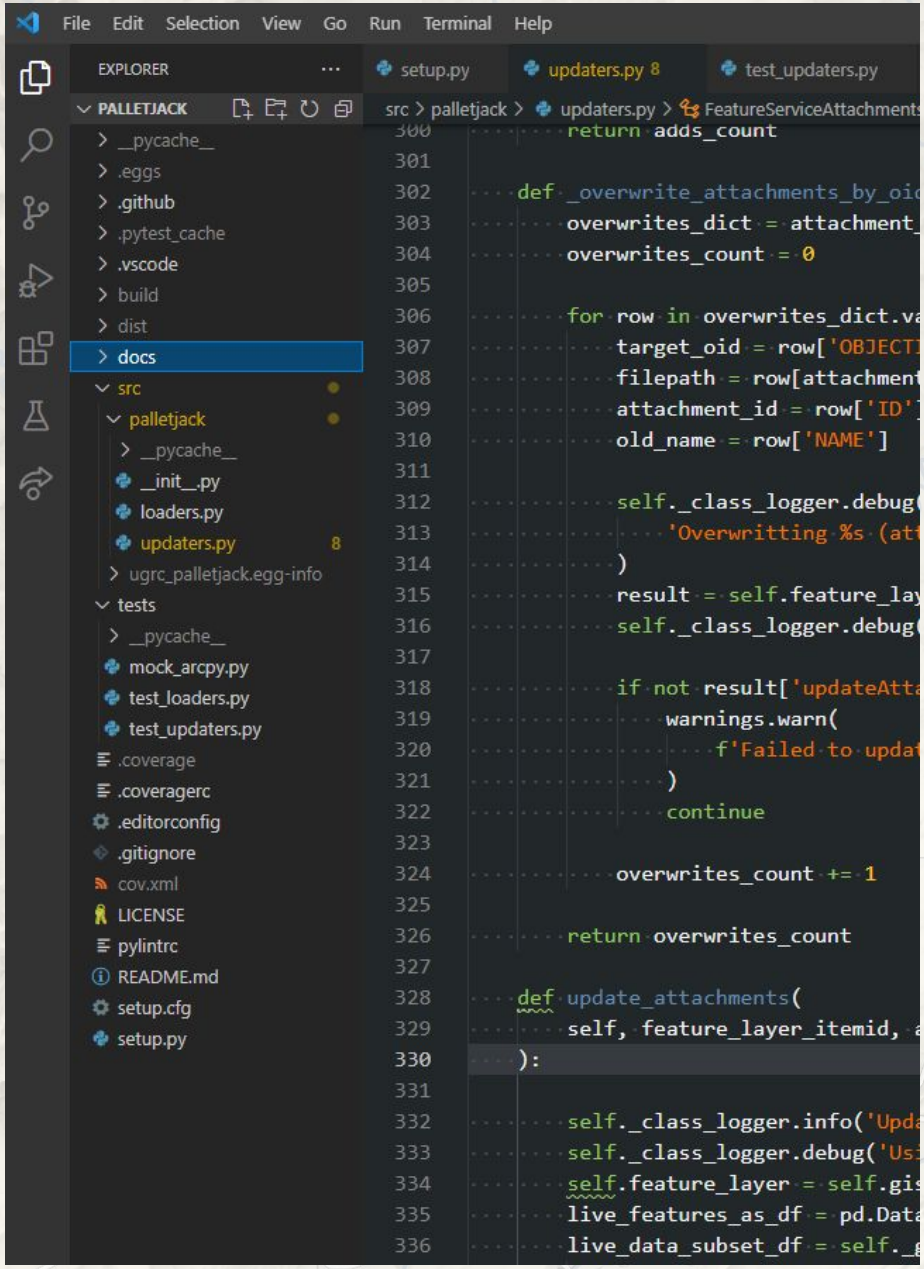
Distribution: Upload to PyPi

Writing with an IDE

https://www.reddit.com/r/Workbenches/comments/lsl4v0/small_but_gets_it_done/

Visual Studio Code:

- Code hinting
- Highlighting
- Multi-select
- Go to definition
- Extensions
- Renaming
- Debugging



Formatting & Linting

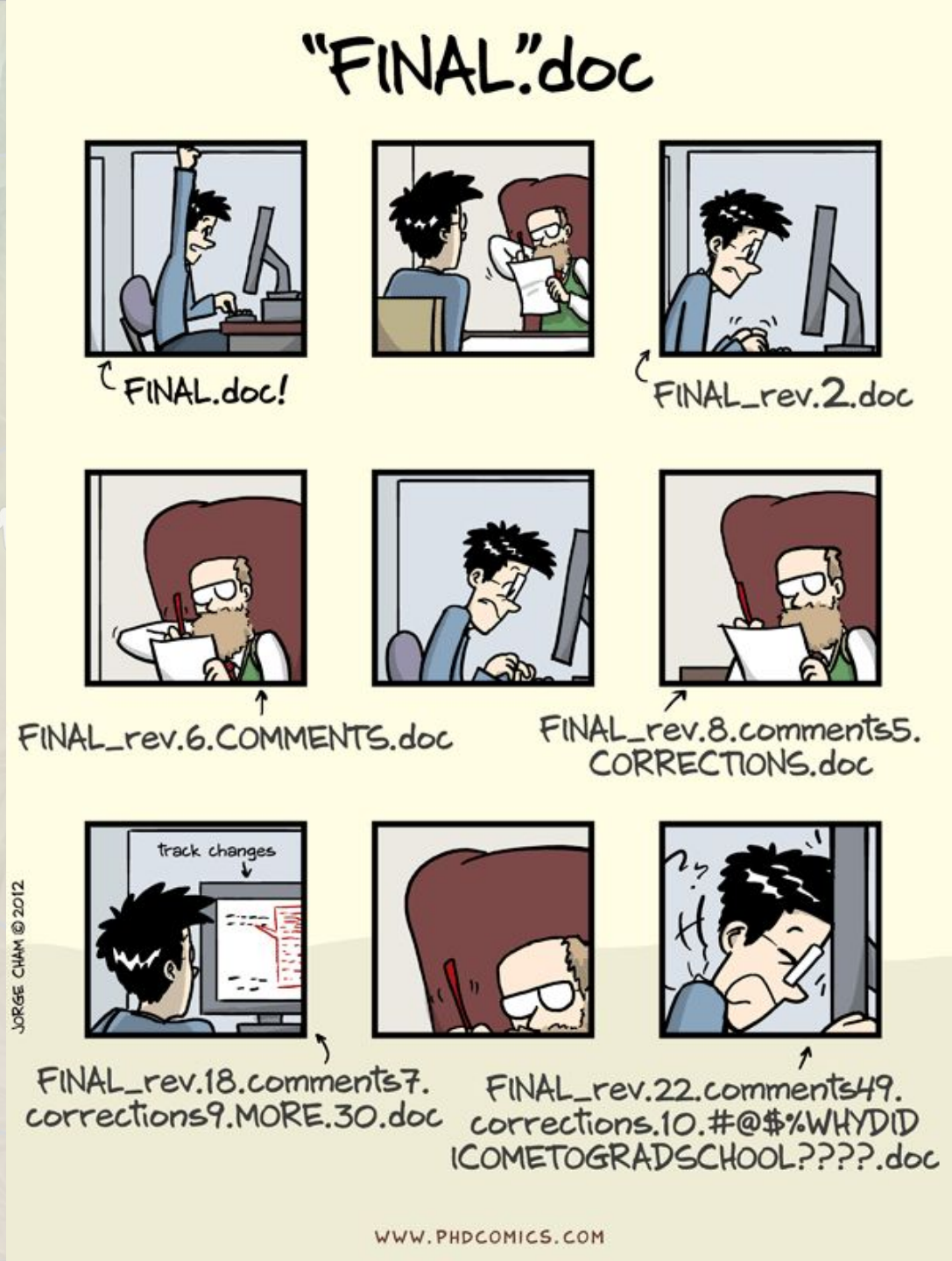
Why Code Style Matters

Formatting vs Linting

- PEP 8
- Consistent style
- Doc strings and comments
- Performed by `yapf`
- Controlled by config file (`setup.cfg`, `pyproject.toml`, ...)

- Detects errors and "code smell"
- Enforces good programming practices
- Performed by `pylint`
- Controlled by `.pylintrc`

Getting a Handle on File Names: Version Control with Git & GitHub



THE MOMENT YOU REALIZE

THE TRUE MEANING OF MEASURE TWICE, CUT ONCE.

git

- Solves the `_v3_try6.py` problem
- Version control
- Branching
- CLI and desktop clients
- Distributed repositories

GitHub

- Online home for repositories
- Pull Requests
- Issue tracking
- Readmes and documentation
- Continuous Integration (CI)
- A whole lot more


THIS IS GIT. IT TRACKS COLLABORATIVE WORK ON PROJECTS THROUGH A BEAUTIFUL DISTRIBUTED GRAPH THEORY TREE MODEL.
COOL. HOW DO WE USE IT?
NO IDEA. JUST MEMORIZE THESE SHELL COMMANDS AND TYPE THEM TO SYNC UP. IF YOU GET ERRORS, SAVE YOUR WORK ELSEWHERE, DELETE THE PROJECT, AND DOWNLOAD A FRESH COPY.

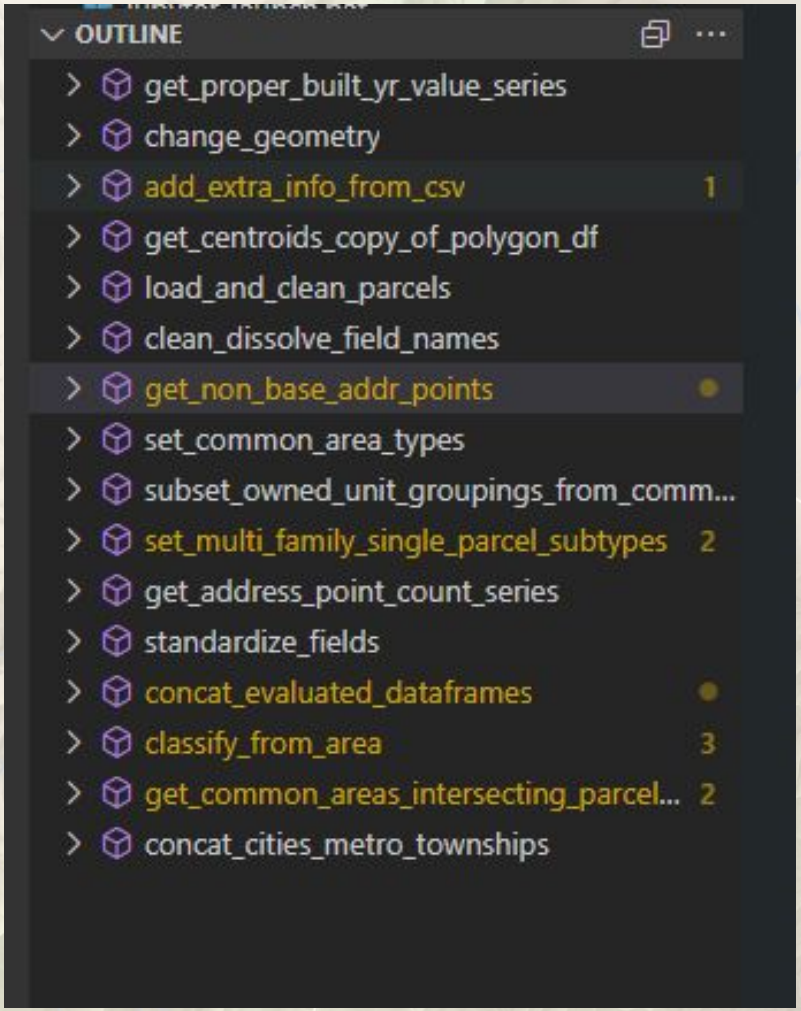
Unit Testing: Knowing When You Break Things

https://reddit.com/u/Patchen35

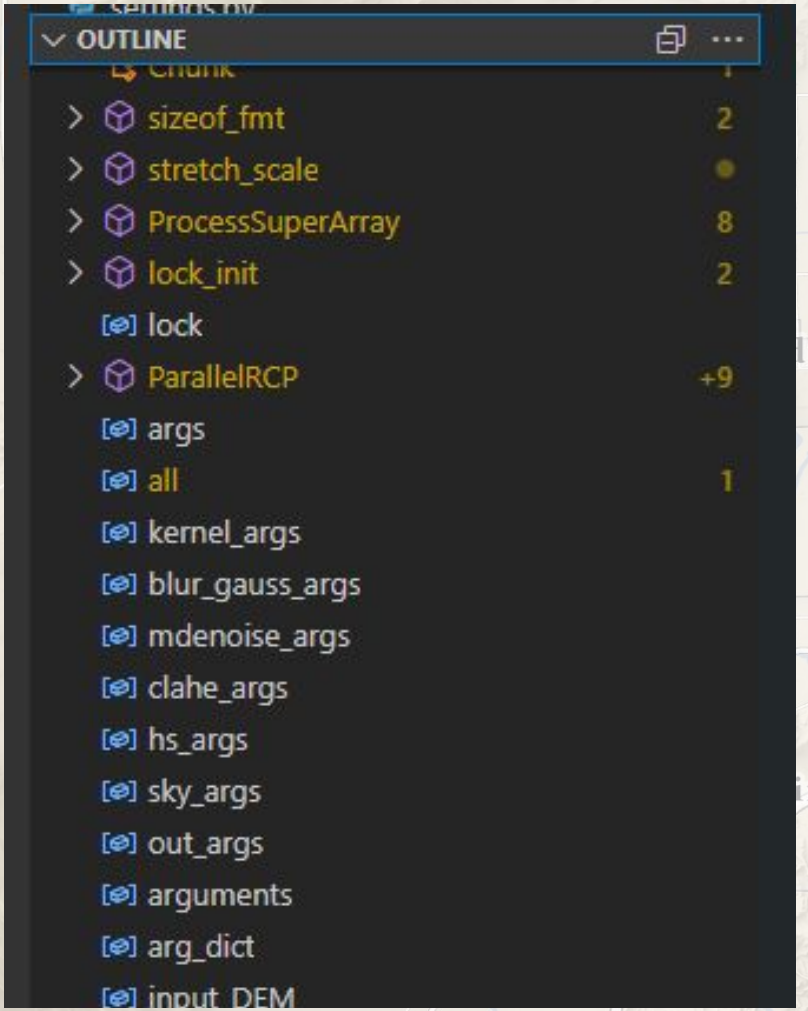
Benefits of Unit Tests

- Trusting your code
- Understanding your code
- Catching errors faster
  - Especially when updating
- Catching more edge cases
- Writing better code

Good Program Design

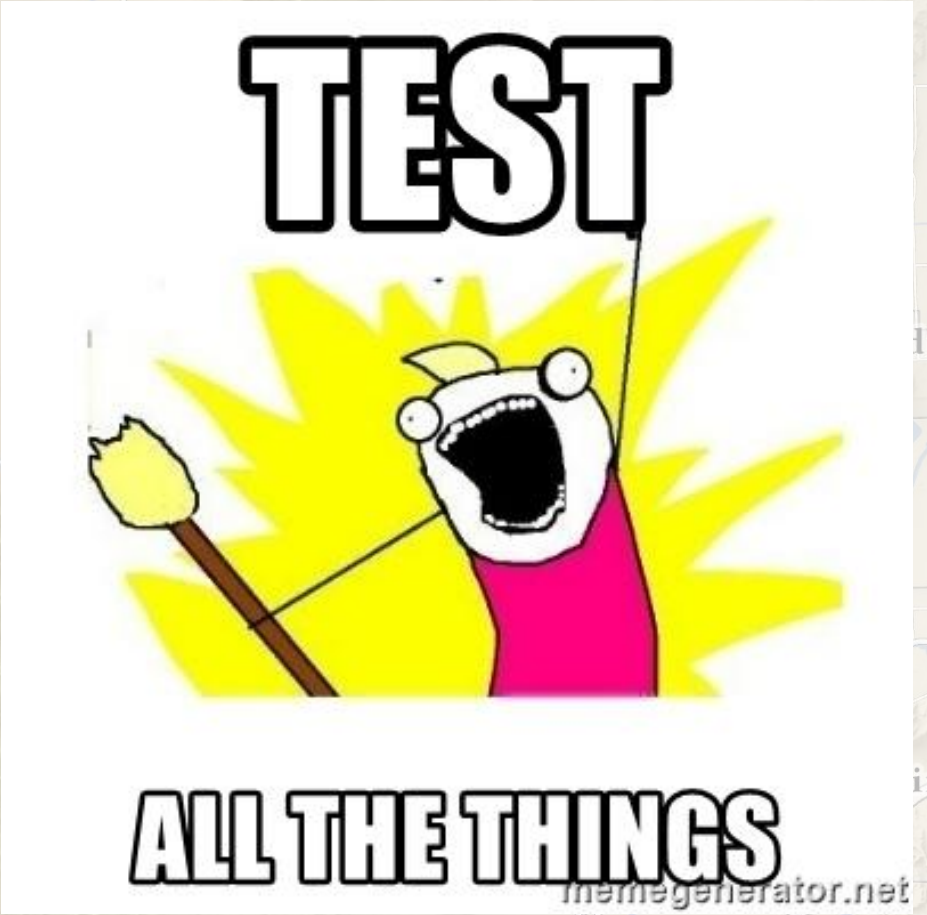


Bad Program Design



Basic Steps of a Unit Test

- Arrange
- Act
- Assert



Unit Test Tools

`unittest`

`pytest-mock`

`pytest-cov`

UGRC

Utah Geospatial Resource Center

jdadams@utah.gov

gis.utah.gov/presentations

Oakley

Kamas

Hideout

Resources

- **UGRC Python template GitHub repo**
  - https://github.com/agrc/python
- **Conda docs**
  - https://docs.conda.io/en/latest/
- **`pip` docs**
  - https://pip.pypa.io/en/stable/
- **Virtual Studio Code**
  - https://code.visualstudio.com/docs
- **`pylint` docs**
  - https://pylint.pycqa.org/en/latest/
- **`yapf` docs**
  - https://github.com/google/yapf

Resources

- **Git docs**
  - https://git-scm.com/doc
- **Some decent git tutorials**
  - https://www.atlassian.com/git/tutorials
- **GitHub docs**
  - https://docs.github.com/en
- **`unittest.mock` reference**
  - https://docs.python.org/3/library/unittest.mock.html
- **pytest docs**
  - https://docs.pytest.org/en/6.2.x/contents.html
- **`pytest-mock` docs**
  - https://pypi.org/project/pytest-mock/
- **Some patchy notes on testing**
  - https://github.com/jacobdadams/notes/blob/master/pytest.md