**Practical Python Training for Modelers**

Intro and Goals

Code organization styles for projects – pros and cons

* Free-form Jupyter notebook
* Functions + procedures
* Classes

Setting up modeling projects for collaboration with users and multiple developers

* A good folder structure for projects
* Using projfiles.py (and a files Class)
  + Manage file and folder locations across a project
  + Toggle between test and production mode
* Code for helping your libraries a) know their own directory location and b) be oriented to other \*.py and project files

Code Architectures in Python – Pros and cons of each and why OOP is “top of the mountain”

OOP aka Object Oriented Programming (a practical level…)

* What is a Python class aka object?
* Using \_\_init\_\_ to kick things off and document what’s in your class
* Getting “self” comfortable
* Methods, procedures and properties

TDD aka Test-Driven Design(a practical level…)

* Setting up a Pytest tests file
* How to write an individual test (assert etc.)
* Hints on test naming practices and docstrings for tests
* Running individual or groups of tests
* Using the Pandas.testing module to check DataFrames

Planning/architecting your modeling projects

Putting it all together: Tuhdoop (aka TDD/OOP) for modeling in Python

References

1. [TDD\_OOP\_Tutorial](https://github.com/jlandgre/TDD_OOP_Tutorial), Github repo, J.D. Landgrebe
2. Gruppetta [extended Twitter thread on Object-Oriented Programming](https://twitter.com/s_gruppetta_ct/status/1641121936414736385)
3. [The Python Coding Book](https://thepythoncodingbook.com/book-outline/) – online book by [Stephen Gruppetta](https://twitter.com/s_gruppetta_ct)
4. [Modeling and Simulation in Python](https://allendowney.github.io/ModSimPy/index.html) by Allen Downey
5. [Pytest](https://docs.pytest.org/en/7.3.x/)
6. [Object-Oriented Programming Wikipedia](https://en.wikipedia.org/wiki/Object-oriented_programming)
7. [Test-Driven Development Wikipedia](https://en.wikipedia.org/wiki/Test-driven_development)
8. Go-to book on user interface design: [About Face, by Alan Cooper et al](https://www.wiley.com/en-us/About+Face:+The+Essentials+of+Interaction+Design,+4th+Edition-p-9781118766576).
   * See “Designing Considerate Products” Chapter 12 in 3rd edition
   * [Blog on characteristics of considerate products](https://www.designprinciplesftw.com/collections/principles-for-considerate-products)