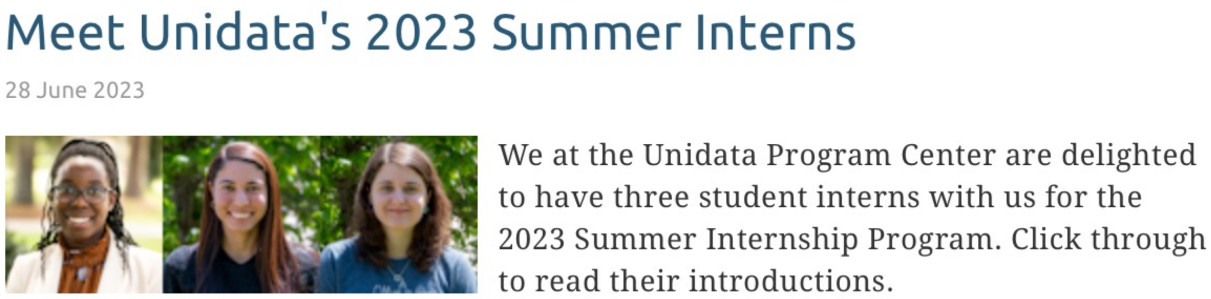
My Summer at Unidata

Erin Rhoades

July 27th, 2023

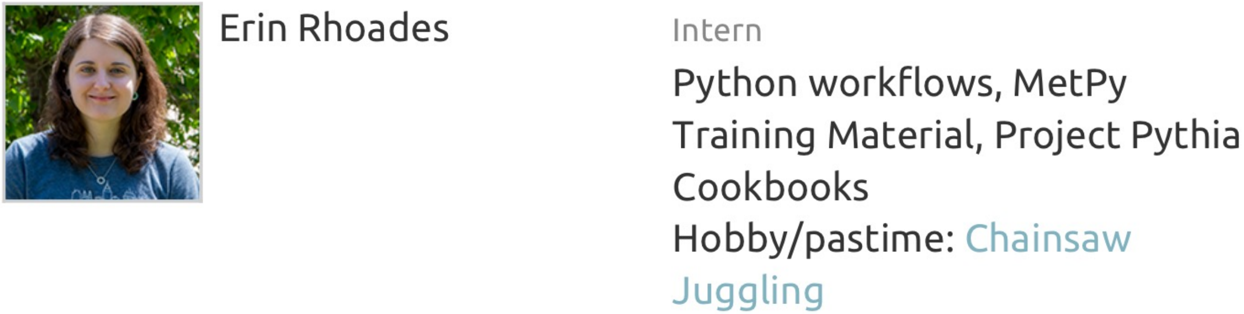


Meet Unidata's 2023 Summer Interns

28 June 2023

We at the Unidata Program Center are delighted to have three student interns with us for the 2023 Summer Internship Program. Click through to read their introductions.

Welcome Summer Intern Erin Rhoades

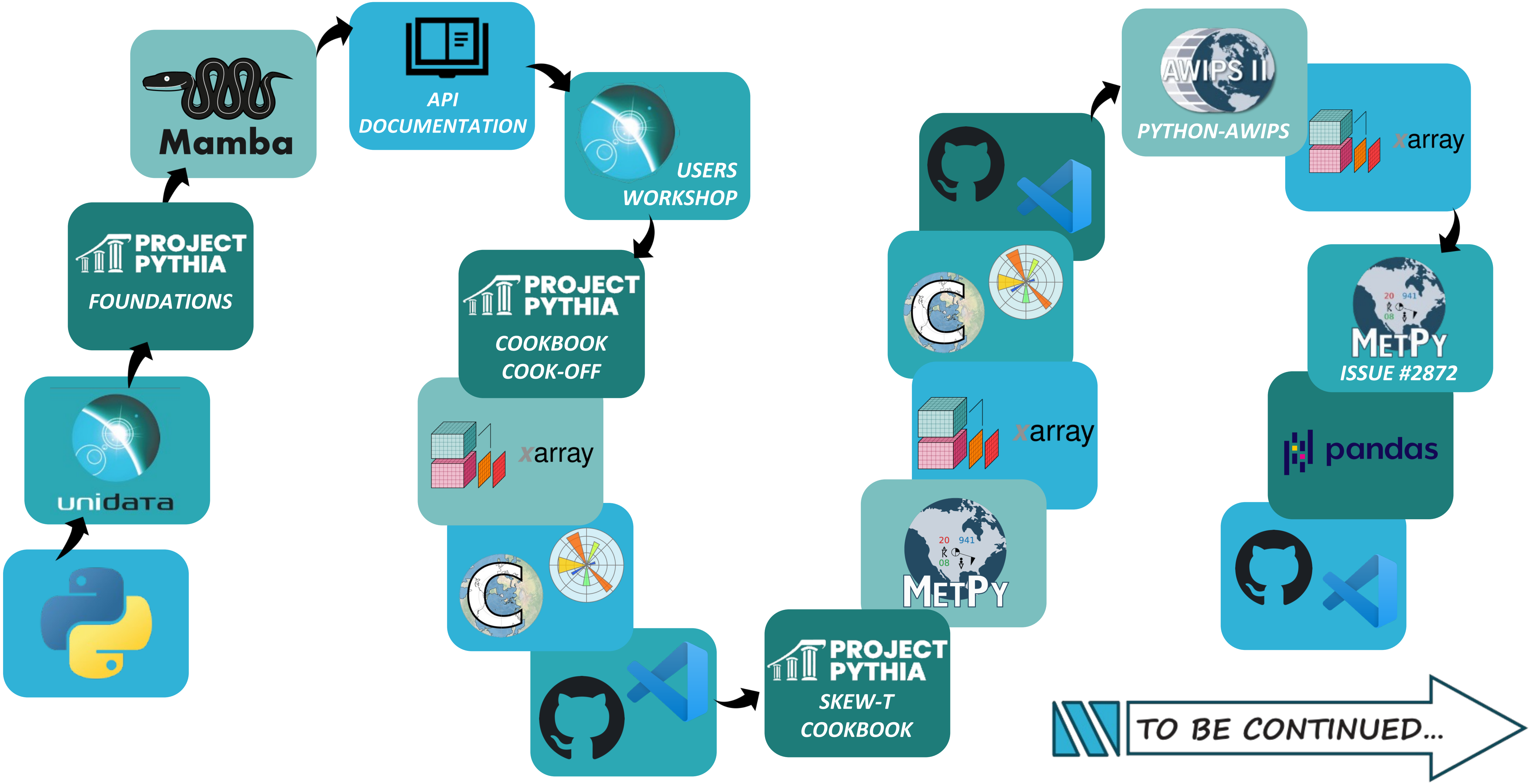


Erin Rhoades

Intern

Python workflows, MetPy Training Material, Project Pythia Cookbooks

Hobby/pastime: Chainsaw Juggling



TO BE CONTINUED...

How I Got Here

- 4th year MSU Denver Meteorology student
- Experience with Unidata products
  - IDV
  - AWIPS CAVE
- Started learning Python in late 2022
- Attended AMS 2023
  - MetPy workshop
- Applied for internship and accepted!

Internship Time

- Summer Goals
  - Solidify fundamental Python skills
  - Create something that can help others

Exploration

- Project Pythia Foundations
  - Work through books
- Environment management
  - Basics
  - Switch to Mamba

PROJECT PYTHIA FOUNDATIONS

Mamba

API Documentation

API (application programming interface) reference is a section in most package documentation. It is a reference guide that describes all the features and details in the package. API information is written inside of the core package code and is auto-generated into a readable format. It is available in the API Reference section in the package documentation site.

Function name to call
Xarray = package function originates from
.align = specific function

Function definition and key information

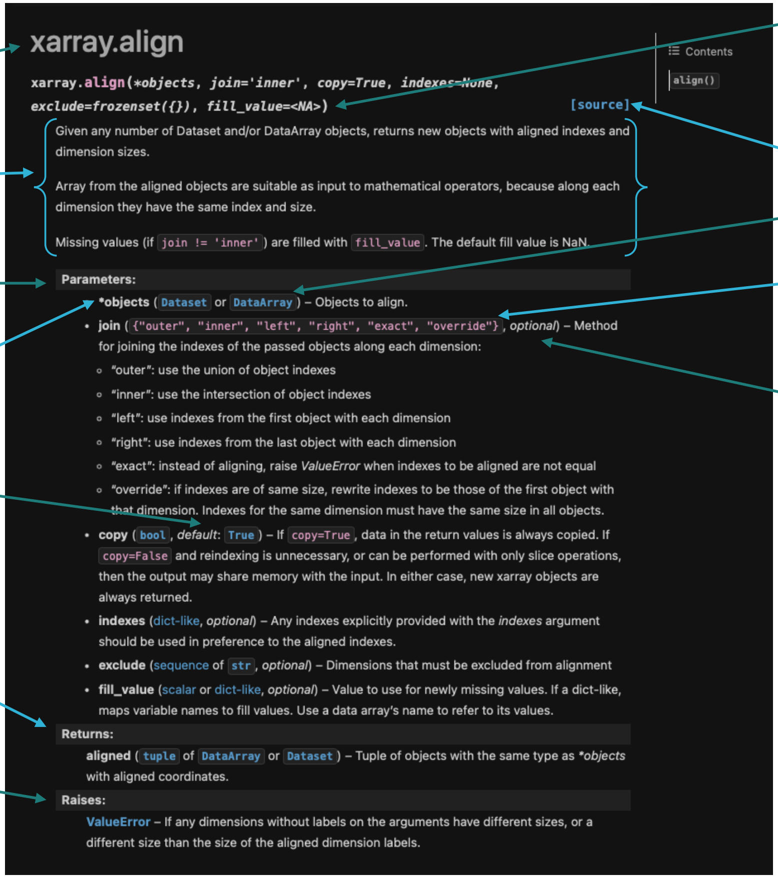
Expanded definitions of each parameter. Will state if optional argument here

* Indicates multiple arguments can be passed

States parameter default behavior

Data structure produced when using the function xarray.align ()

Further information about potential errors and exceptions that could occur



xarray.align

```
xarray.align(*objects, join='inner', copy=True, indexes=None, exclude=frozenset({}), fill_value=<NA>)
```

[source]

Given any number of Dataset and/or DataArray objects, returns new objects with aligned indexes and dimension sizes.

Array from the aligned objects are suitable as input to mathematical operators, because along each dimension they have the same index and size.

Missing values (if `join != 'inner'`) are filled with `fill_value`. The default fill value is NaN.

Parameters:

- ***objects** (`Dataset` or `DataArray`) – Objects to align.
- **join** (`{"outer", "inner", "left", "right", "exact", "override"}`, *optional*) – Method for joining the indexes of the passed objects along each dimension:
    - "outer": use the union of object indexes
    - "inner": use the intersection of object indexes
    - "left": use indexes from the first object with each dimension
    - "right": use indexes from the last object with each dimension
    - "exact": instead of aligning, raise *ValueError* when indexes to be aligned are not equal
    - "override": if indexes are of same size, rewrite indexes to be those of the first object with that dimension. Indexes for the same dimension must have the same size in all objects.
- **copy** (`bool`, *default*: `True`) – If `copy=True`, data in the return values is always copied. If `copy=False` and reindexing is unnecessary, or can be performed with only slice operations, then the output may share memory with the input. In either case, new xarray objects are always returned.
- **indexes** (dict-like, *optional*) – Any indexes explicitly provided with the *indexes* argument should be used in preference to the aligned indexes.
- **exclude** (sequence of `str`, *optional*) – Dimensions that must be excluded from alignment
- **fill_value** (scalar or dict-like, *optional*) – Value to use for newly missing values. If a dict-like, maps variable names to fill values. Use a data array's name to refer to its values.

Returns:

aligned (`tuple` of `DataArray` or `Dataset`) – Tuple of objects with the same type as **objects* with aligned coordinates.

Raises:

ValueError – If any dimensions without labels on the arguments have different sizes, or a different size than the size of the aligned dimension labels.

All possible parameters inputs, not all required. Parameters section below will state if optional

Link to source code on GitHub

Data structure type input options

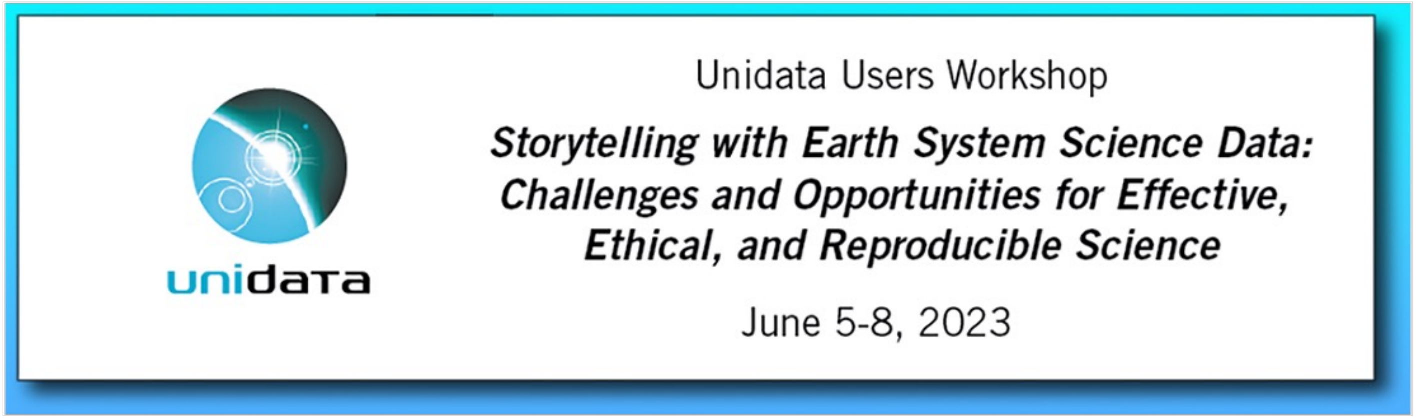
Argument options for specific parameter

Indicates optional parameter

In general, blue text (color varies depending on documentation) links to a pythonic dictionary definition or to the API reference section for the term

API DOCUMENTATION

Unidata Users Workshop



Unidata Users Workshop

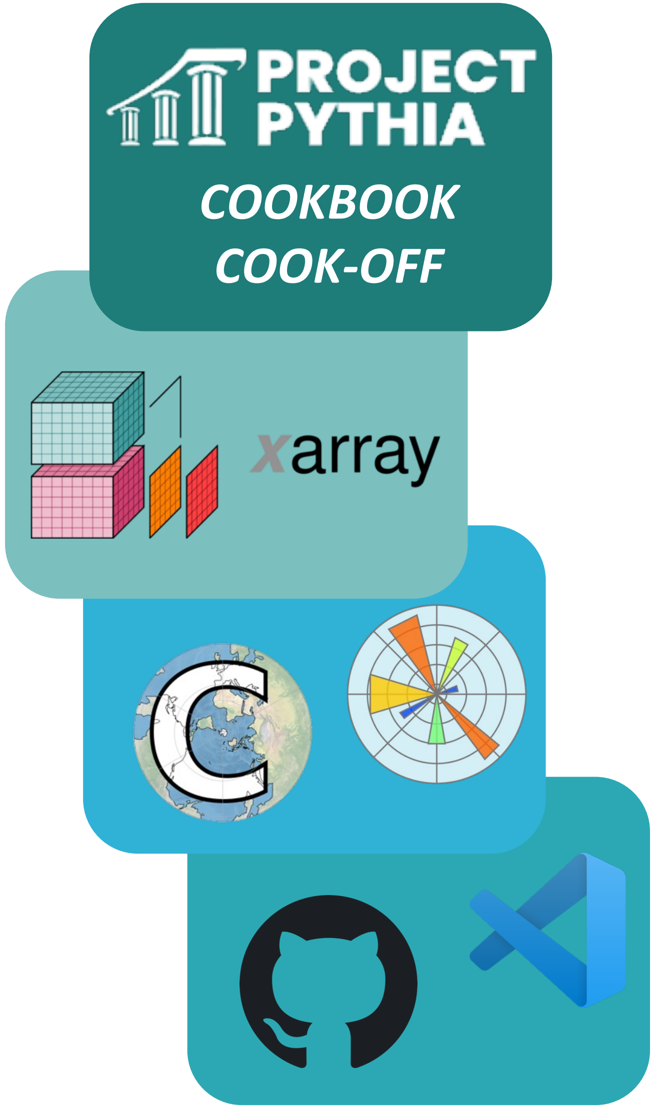
Storytelling with Earth System Science Data: Challenges and Opportunities for Effective, Ethical, and Reproducible Science

June 5-8, 2023

USERS WORKSHOP

Project Pythia – Cookbook Cook-off

- Cookbook 101
- Remote data access with xarray ‘mini’ cookbook
- Several Python packages
  - xarray
  - Cartopy
  - Matplotlib
- Visual Studio Code
- GitHub

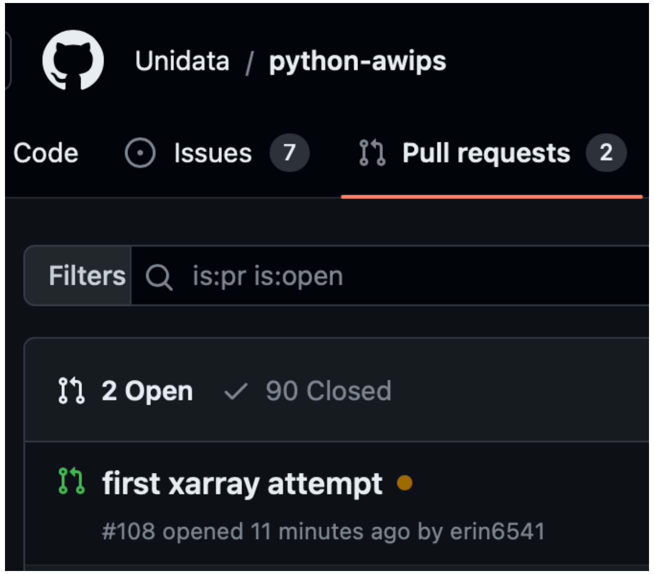


Project Pythia – Skew-T Cookbook

- Full tutorial with example workflows
- Encompasses MetPy's main purposes
- Several Python packages
  - MetPy
  - xarray
  - Matplotlib
  - Cartopy
- Visual Studio Code
- GitHub

Python-AWIPS

- Explore development side of Python
- Learn more about AWIPS
  - Unidata eLearning modules
- Change EDEX data output from NumPy arrays to xarray datasets
- Work in progress



2 Open ✓ 90 Closed

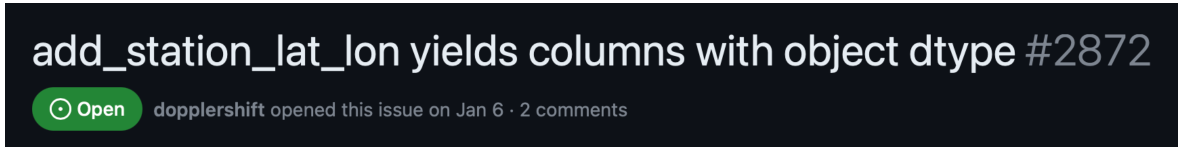
first xarray attempt ●

#108 opened 11 minutes ago by erin6541

PYTHON-AWIPS

MetPy Issue #2872

- Learn how to contribute to MetPy
- Change latitude and longitude data types from object to float32
- MetPy and pandas
- Visual Studio Code
- GitHub
- Work in progress


add_station_lat_lon yields columns with object dtype #2872

Open dopplershift opened this issue on Jan 6 · 2 comments

THANK YOU!

- This summer at Unidata was incredible. I learned so much. I was able to accomplish my goals of furthering my personal Python knowledge and creating things to help others.
- Everyone has been so welcoming and made me feel like a part of the Unidata family. Thank you all for taking time out of your summer to help me grow.
- Big thank you to my mentors, Drew and Shay, for all your guidance throughout the summer!