Managing Python environments

What:

- ➔ Run code from a defined computing environment (the “kernel” for the code notebook)
- ➔ Control the set of packages and specific versions installed

Why?

- ➔ Consistent, deterministic results
- ➔ Avoid software conflicts for smooth operation
- ➔ Use new tools on new projects while ensuring older projects continue to run reliably

Why?

Open-source software:

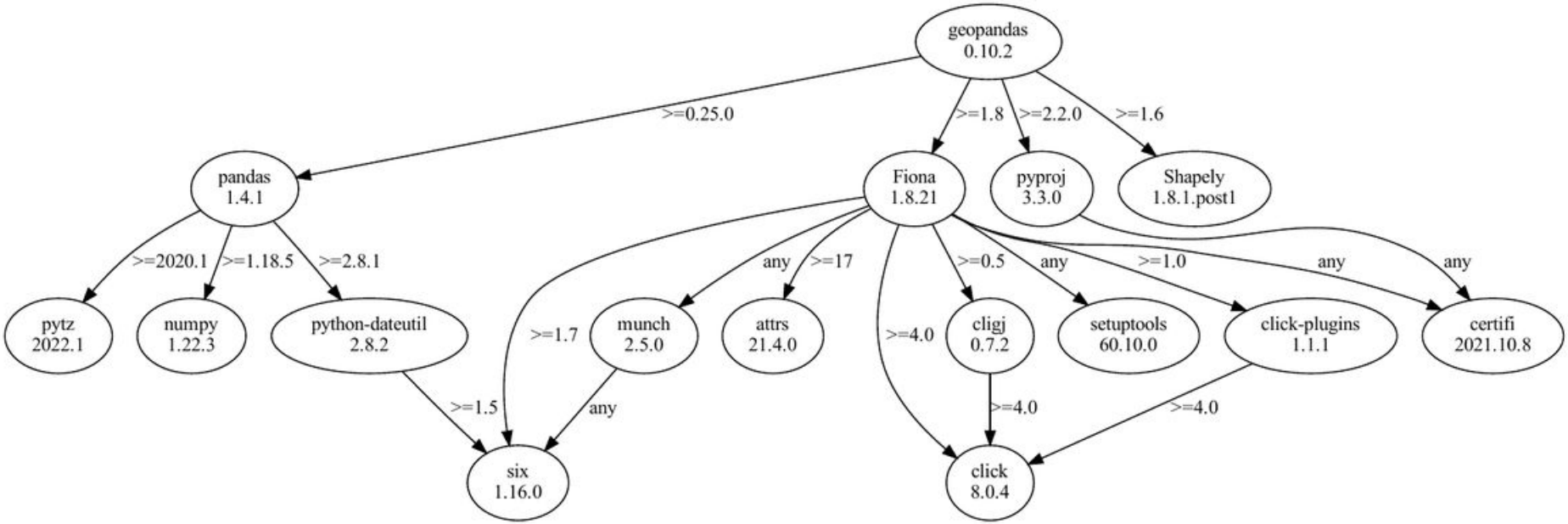
- ➔ Advances quickly
- ➔ Is built on layered dependencies

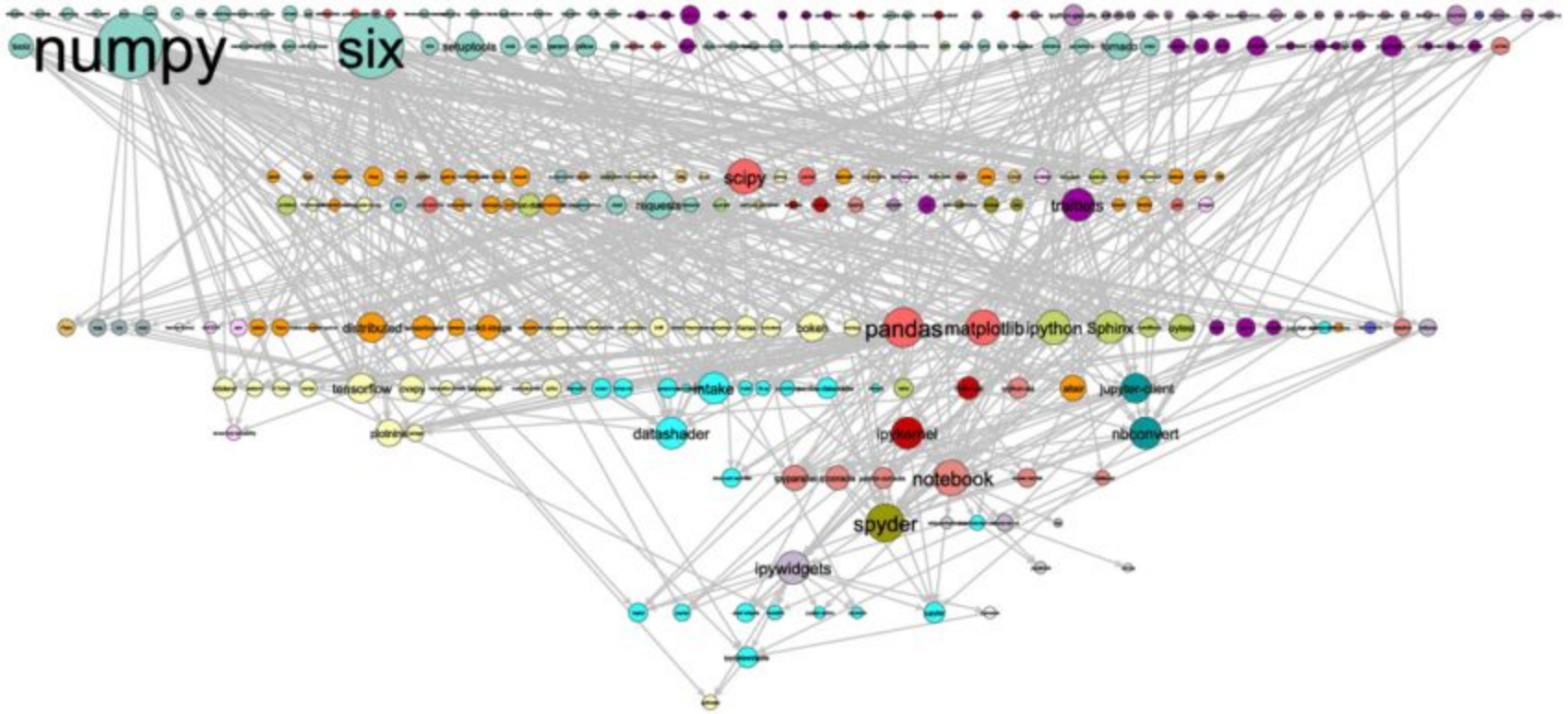
Managing these stacked dependencies is essential for a stable computing platform

Release notes

This is the list of changes to pandas between each release. For ful
logs. For install and upgrade instructions, see Installation.

Version 2.2


geopandas
0.10.2
>=0.25.0
>=1.8
>=2.2.0
>=1.6
pandas
1.4.1
Fiona
1.8.21
pyproj
3.3.0
Shapely
1.8.1.post1
>=2020.1
>=1.18.5
>=2.8.1
any
>=17
>=0.5
any
>=1.0
any
any
pytz
2022.1
numpy
1.22.3
python-dateutil
2.8.2
>=1.7
munch
2.5.0
attrs
21.4.0
>=4.0
cligj
0.7.2
setuptools
60.10.0
click-plugins
1.1.1
certifi
2021.10.8
>=1.5
any
>=4.0
>=4.0
six
1.16.0
click
8.0.4


numpy
six
scipy
requests
tornado
distributed
bokeh
pandas
matplotlib
ipython
Sphinx
tensorflow
intake
jupyter-client
datashader
ipykernel
nbconvert
notebook
spyder
ipywidgets

How:

- Virtual environments
- Package managers
  - conda
  - uv

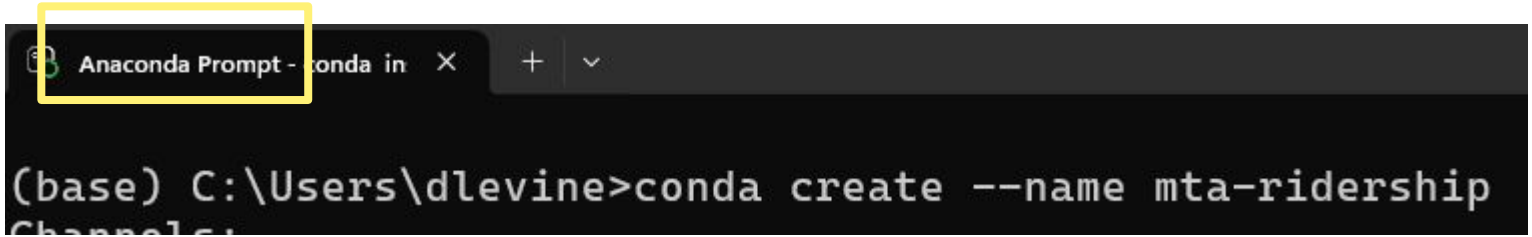
conda

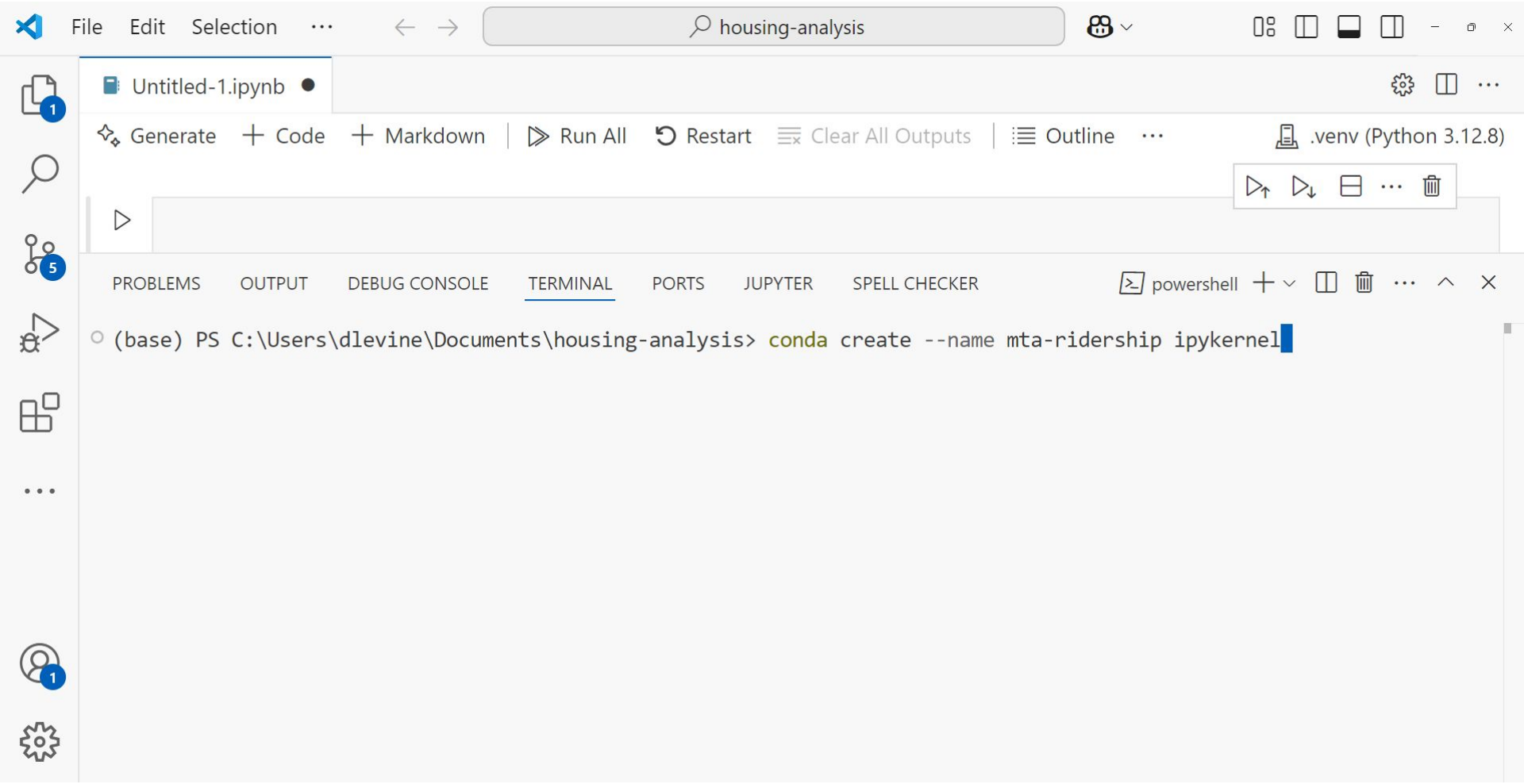
Anaconda (conda)

- Create reusable, named environments
- Stable tool, frequently used by scientists and data scientists

Introduction to environments: https://www.anaconda.com/docs/tools/working-with-conda/environments

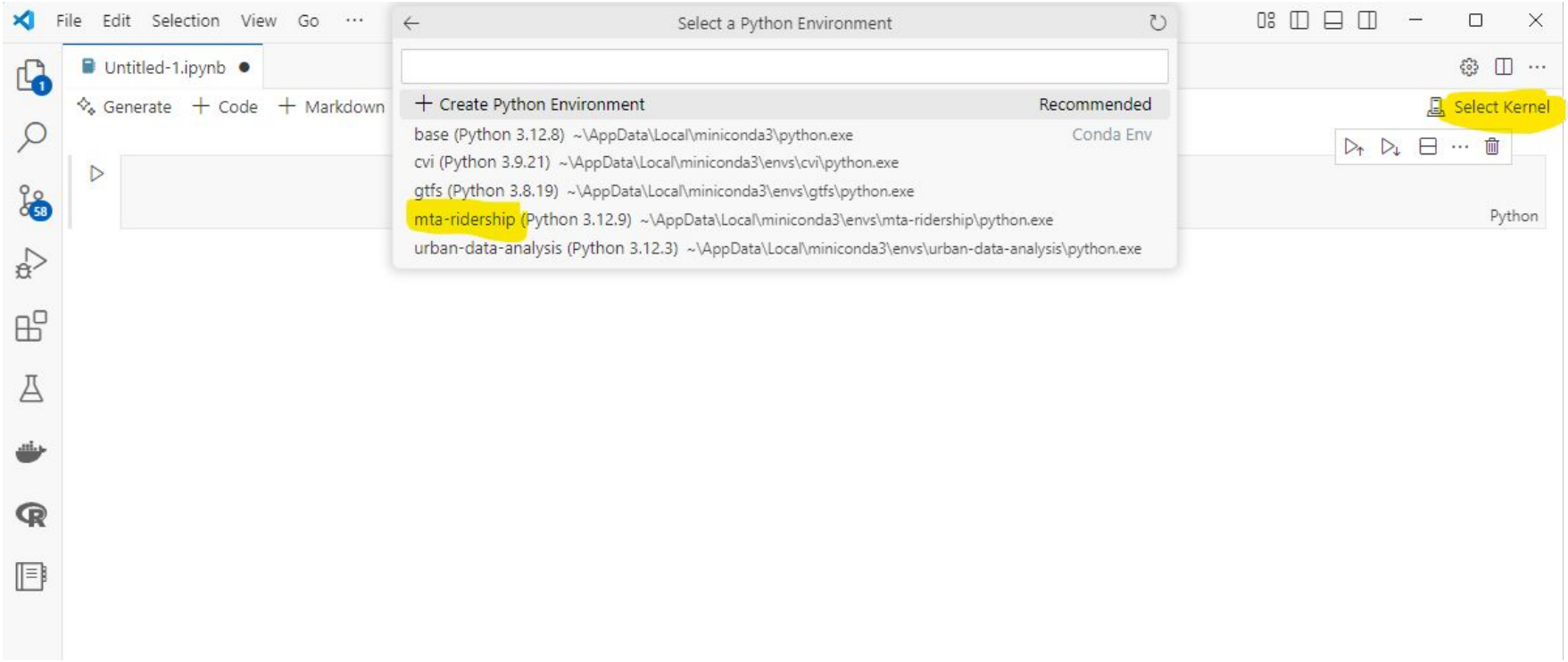
```
# Replace <ENV_NAME> with a name for your environment
# Replace <PACKAGE> with your desired package
# Replace <VERSION> with your desired version (optional)
conda create --name <ENV_NAME> <PACKAGE>=<VERSION>
```


Anaconda Prompt -
(base) C:\Users\dlevine>conda create --name mta-ridership


File
Edit
Selection
housing-analysis
Untitled-1.ipynb
Generate
Code
Markdown
Run All
Restart
Clear All Outputs
Outline
.venv (Python 3.12.8)
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
JUPYTER
SPELL CHECKER
powershell
(base) PS C:\Users\dlevine\Documents\housing-analysis> conda create --name mta-ridership ipykernel

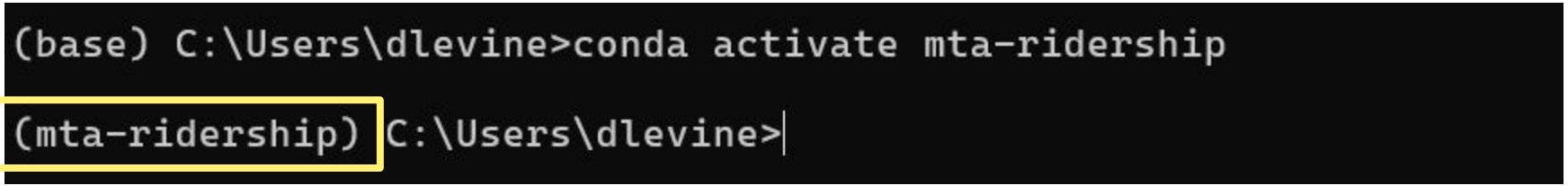
```
Proceed ([y]/n)?


Downloading and Extracting Packages:

Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#     $ conda activate mta-ridership
#
# To deactivate an active environment, use
#
#     $ conda deactivate
```

Use the environment:



Update the environment

```
(base) C:\Users\dlevine>conda activate mta-ridership
```


(base) C:\Users\dlevine>conda activate mta-ridership
(mta-ridership) C:\Users\dlevine>

```
(mta-ridership) C:\Users\dlevine>conda install seaborn matplotlib
```

```
(mta-ridership) C:\Users\dlevine>conda install pandas=2.2
```

```
## Package Plan ##

  environment location: C:\Users\dlevine\AppData\Local\miniconda3\envs\mta-ridershp

  added / updated specs:
    - pandas=2.2


The following packages will be UPDATED:

  pandas                              2.1.4-py312hc7c4135_0 --> 2.2.3-py312h5da7b3


Proceed ([y]/n)?
```

Export environment

```
(mta-ridership) C:\Users\dlevine>conda env export --from-history
name: mta-ridership
channels:
  - https://repo.anaconda.com/pkgs/main
  - https://repo.anaconda.com/pkgs/r
  - https://repo.anaconda.com/pkgs/msys2
dependencies:
  - python=3.12
  - pandas==2.2.3
  - seaborn
  - ipykernel
prefix: C:\Users\dlevine\AppData\Local\miniconda3\envs\mta-ridership

(mta-ridership) C:\Users\dlevine>conda env export --from-history > environment.yml
```

Recreate environment

```
(mta-ridership) C:\Users\dlevine>conda env create -f environment.yml
```

uv

uv

- Create project environment
- Works *much* faster
- New tool, still in development, likely to change

Introduction to projects:
https://docs.astral.sh/uv/guides/projects/

Install

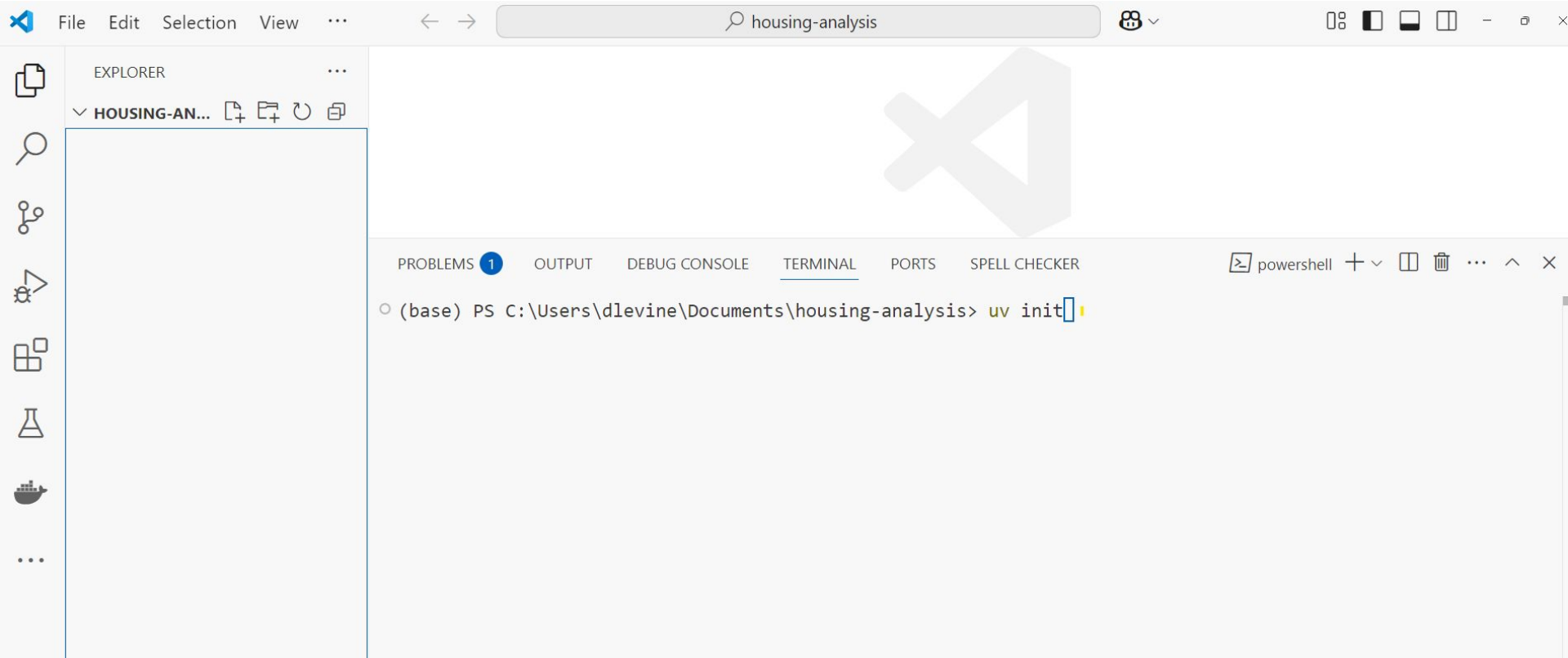
- ➔ Instructions here: https://docs.astral.sh/uv/getting-started/installation/
- ➔ (Copy and paste into terminal and run)
- ➔ You may need to restart terminal

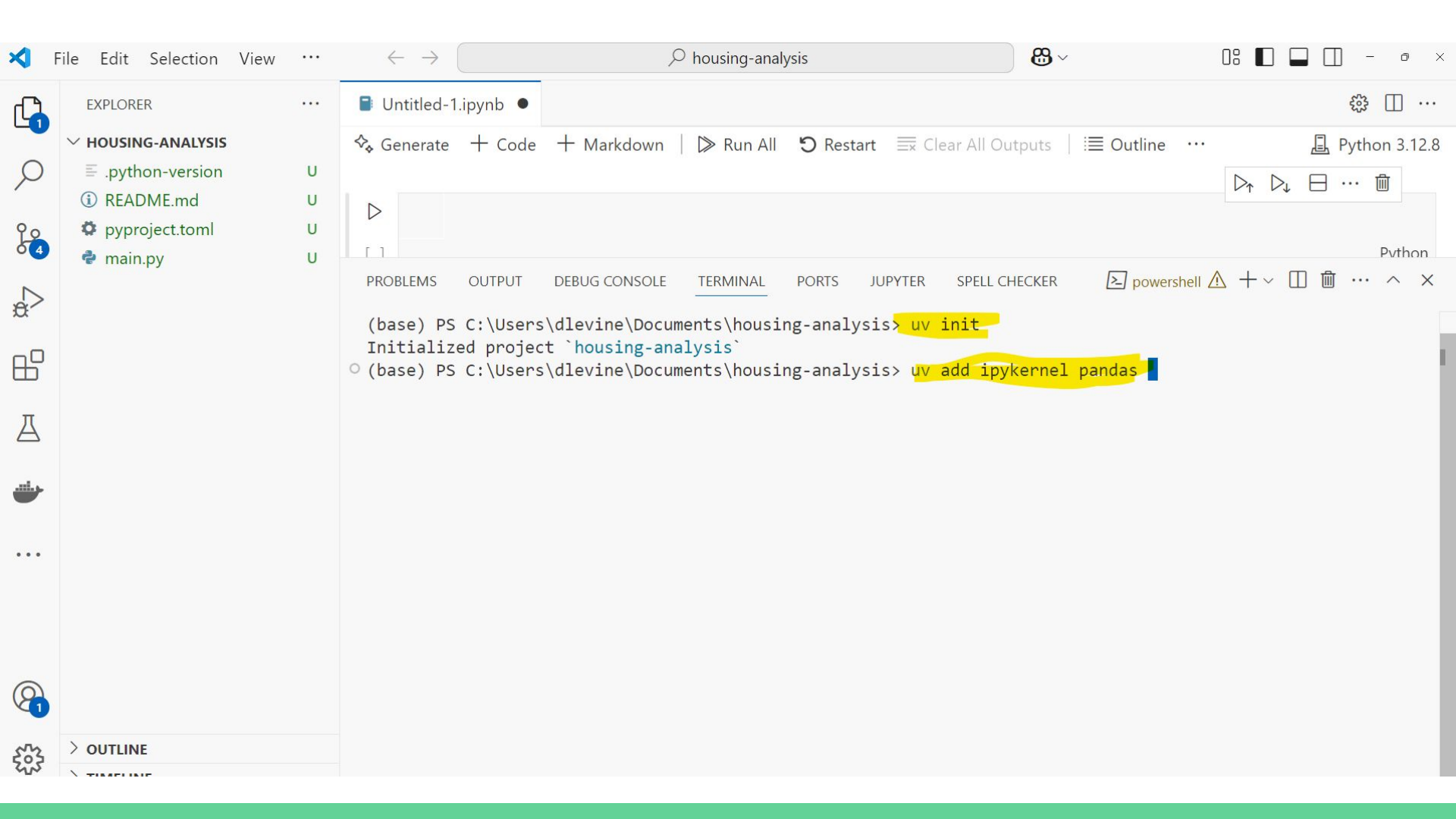
`cd` to new folder and run:

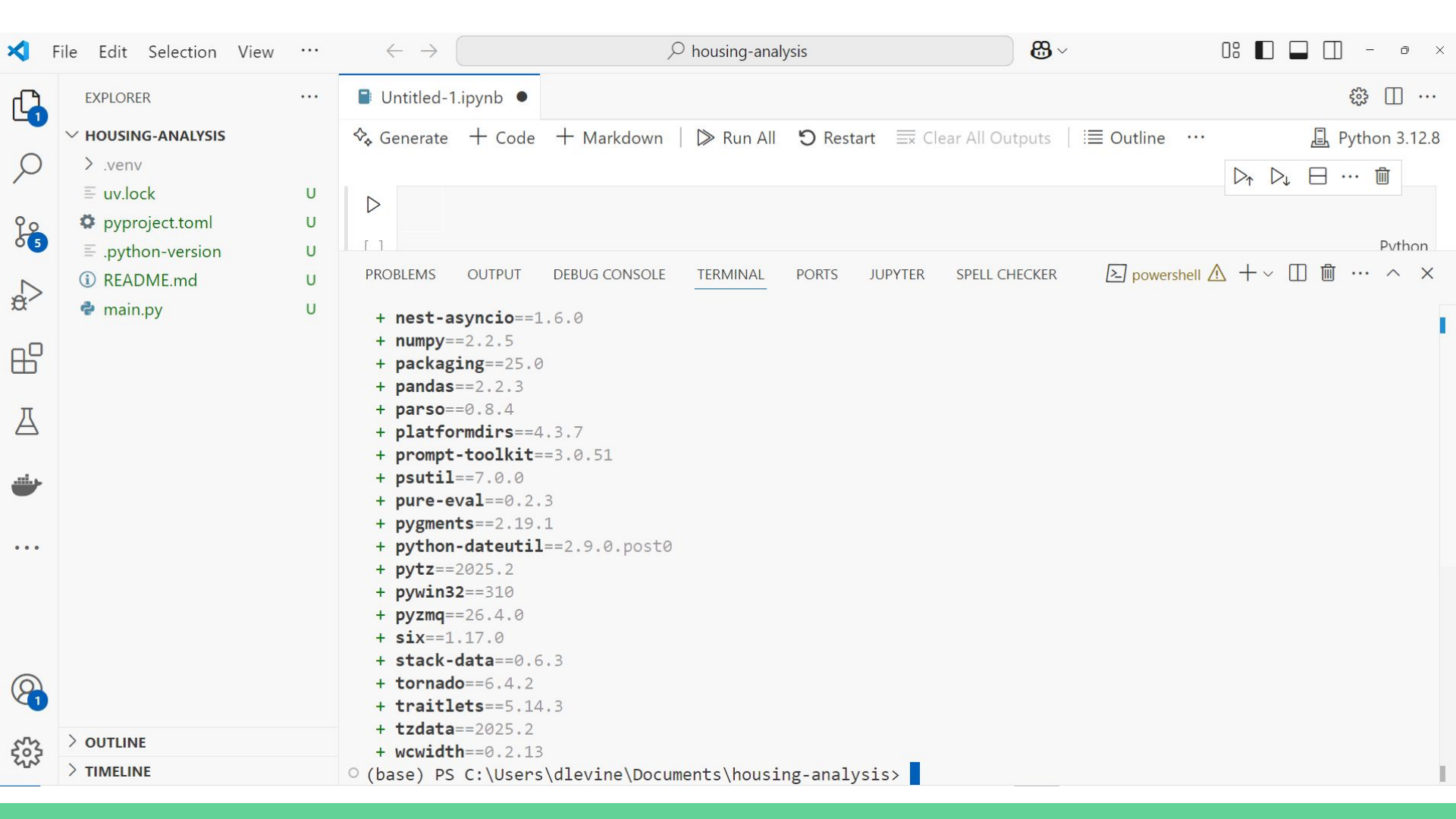
```
(base) PS C:\Users\dlevine\Documents\housing-analysis> uv init --name housing-analysis
```

```
(base) PS C:\Users\dlevine\Documents\housing-analysis> uv add ipykernel pandas geopandas altair
```

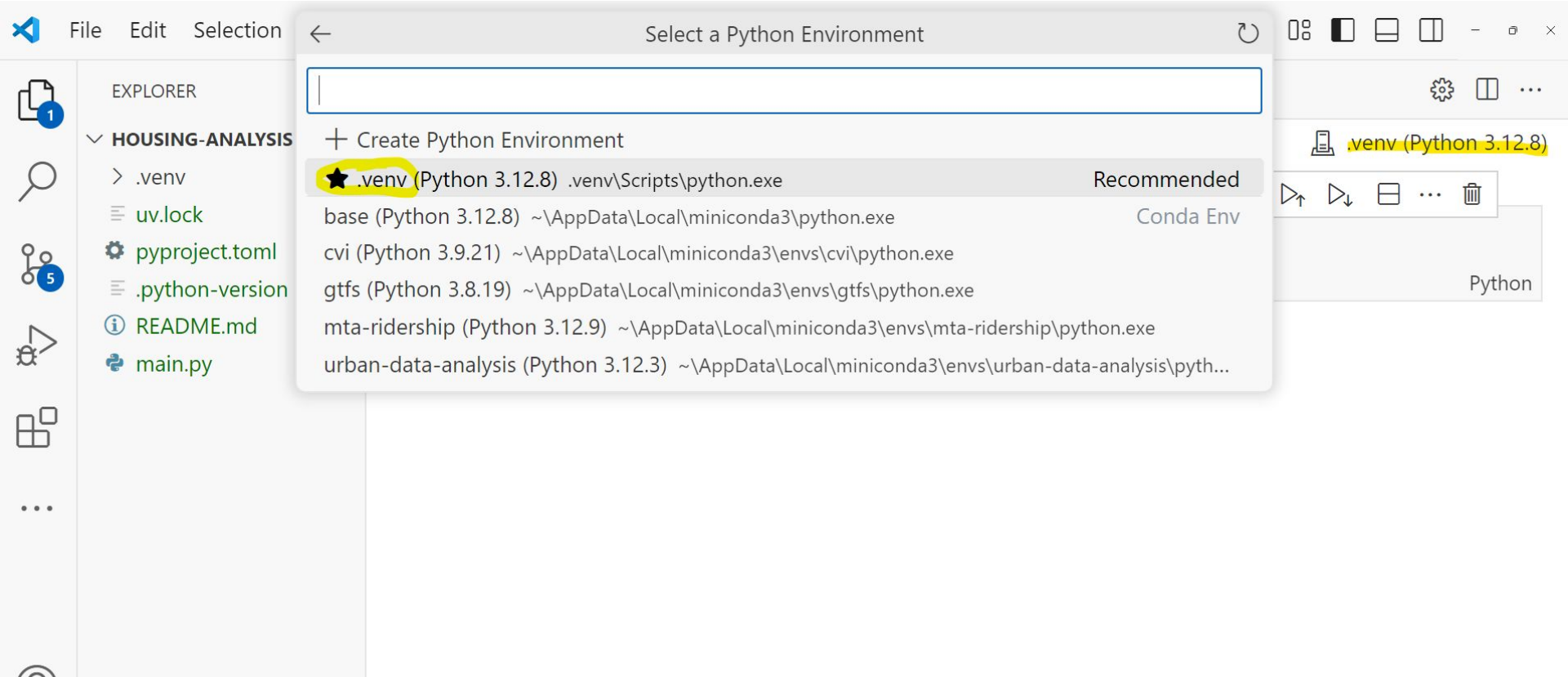
Or open a folder in VSCode and run in the terminal there:



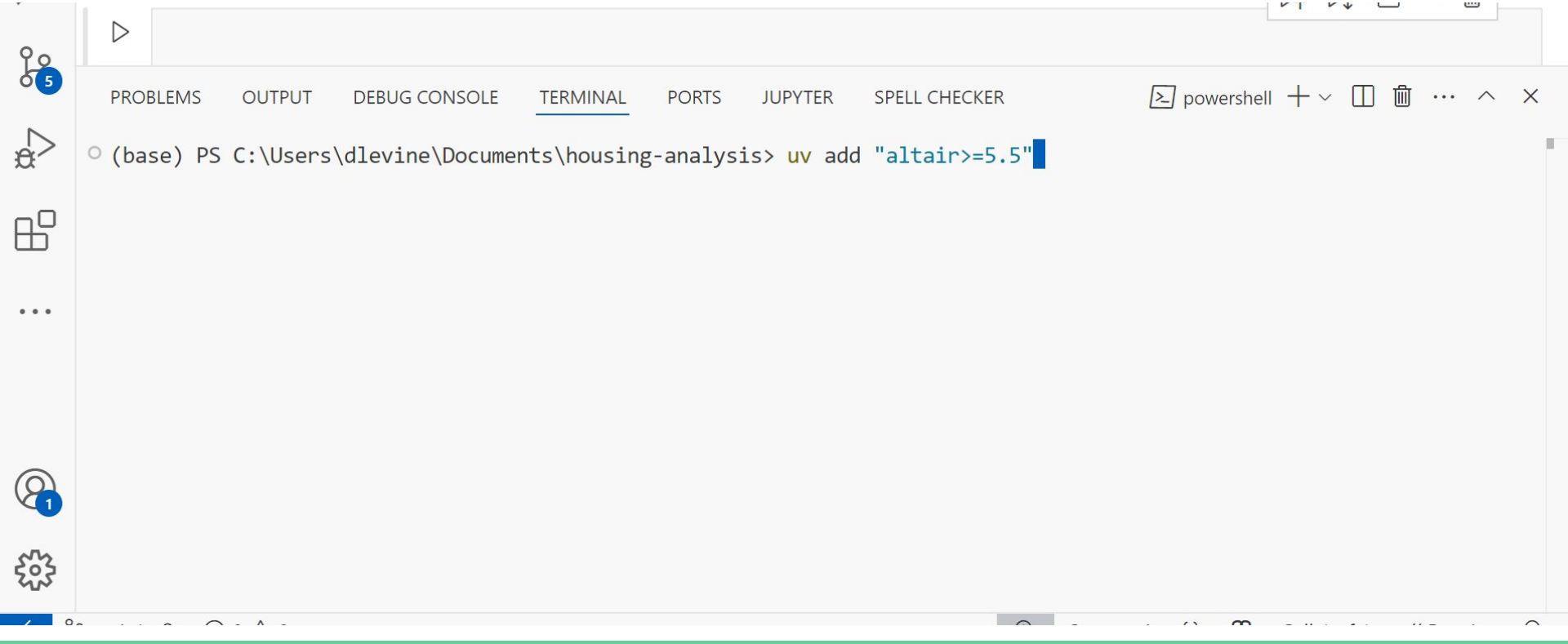

File
Edit
Selection
View
housing-analysis
EXPLORER
HOUSING-ANALYSIS
.python-version U
README.md U
pyproject.toml U
main.py U
Untitled-1.ipynb
Generate
Code
Markdown
Run All
Restart
Clear All Outputs
Outline
Python 3.12.8
Python
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
JUPYTER
SPELL CHECKER
powershell
(base) PS C:\Users\dlevine\Documents\housing-analysis> uv init
Initialized project `housing-analysis`
(base) PS C:\Users\dlevine\Documents\housing-analysis> uv add ipykernel pandas
OUTLINE


File
Edit
Selection
View
housing-analysis
EXPLORER
HOUSING-ANALYSIS
.venv
uv.lock U
pyproject.toml U
.python-version U
README.md U
main.py U
OUTLINE
TIMELINE
Untitled-1.ipynb
Generate
Code
Markdown
Run All
Restart
Clear All Outputs
Outline
Python 3.12.8
Python
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
JUPYTER
SPELL CHECKER
powershell
+ nest-asyncio==1.6.0
+ numpy==2.2.5
+ packaging==25.0
+ pandas==2.2.3
+ parso==0.8.4
+ platformdirs==4.3.7
+ prompt-toolkit==3.0.51
+ psutil==7.0.0
+ pure-eval==0.2.3
+ pygments==2.19.1
+ python-dateutil==2.9.0.post0
+ pytz==2025.2
+ pywin32==310
+ pyzmq==26.4.0
+ six==1.17.0
+ stack-data==0.6.3
+ tornado==6.4.2
+ traitlets==5.14.3
+ tzdata==2025.2
+ wcwidth==0.2.13
(base) PS C:\Users\dlevine\Documents\housing-analysis>

Use the environment



Update the environment



Necessary & useful packages

Useful packages

`ipykernel`	required to run Jupyter notebook
`pandas`	standard DataFrame tool
`seaborn, matplotlib, altair`	data visualization
`openpyxl, xlrd`	read/write Excel files (not included in default pandas installation)
`geopandas`	geospatial data
`scipy, statsmodels`	statistical analysis