The Notebook:

`[Python]` has always been enough

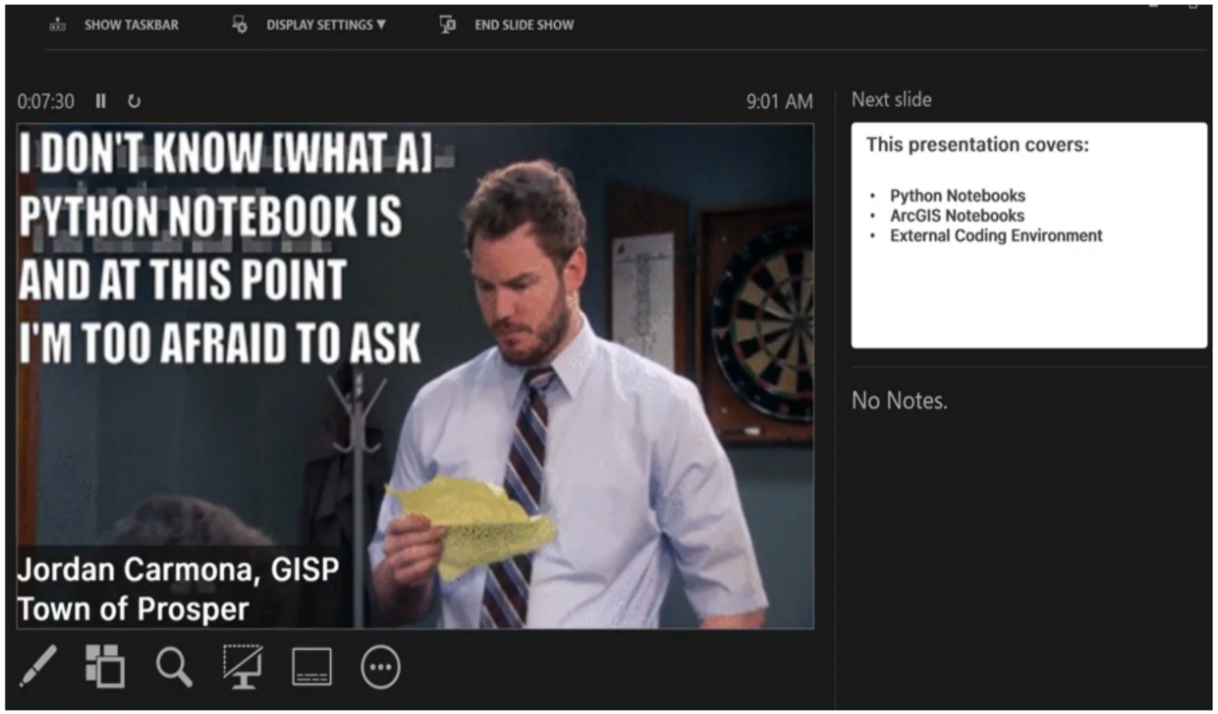
A love letter, or rather, 365 love letters to the Python Notebook.

[2]:
```
import datetime

print(f"""
A presentation by Jordan Carmona, GISP
Washington GIS Conference, Room 407-408
{datetime.datetime.now():%A, %B %d, %Y %H:%M}
""")
```

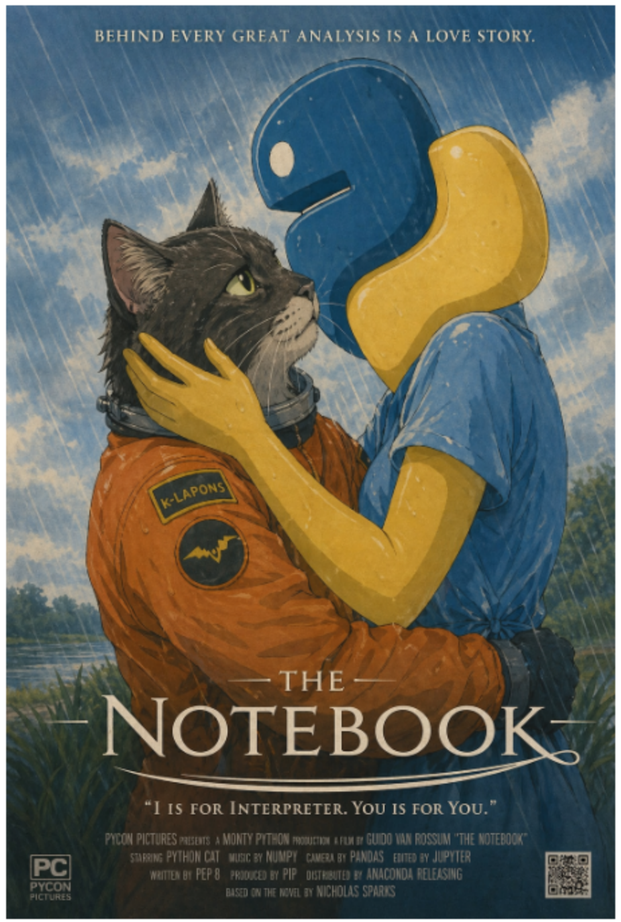
```
A presentation by Jordan Carmona, GISP
Washington GIS Conference, Room 407-408
Wednesday, May 20, 2026 14:40
```

[1]:
```
import urllib3
import pandas as pd
import arcgis
import arcpy
urllib3.disable_warnings(urllib3.exceptions.InsecureRequestWarning)
pd.options.display.max_columns = 10
```

[ ]:
```

```


SHOW TASKBAR
DISPLAY SETTINGS
END SLIDE SHOW
0:07:30
9:01 AM
I DON'T KNOW [WHAT A] PYTHON NOTEBOOK IS AND AT THIS POINT I'M TOO AFRAID TO ASK
Jordan Carmona, GISP
Town of Prosper
Next slide
This presentation covers:
Python Notebooks
ArcGIS Notebooks
External Coding Environment
No Notes.

About Me



Fun Things

- banishing invasive plants with a weed wrench
- eating random berries
- amatuer ukuleleist
- cat fancier

Work Things

- part 107 sUAS remote pilot
- data engineer & automation expert
- 12 years of professional experience
- private sector, public sector, education, and retail

I traveled the road many times, sat on the porch, played the games, been the dog, everything. I was even the scarecrow for a while... ...because I hadn't done it.

this presentation will cover the:

- when

- where

- how

to use Python Notebooks

- gotchas to watch out for

- quality of life tweaks to make

lastly, preserving your romance for your future self through

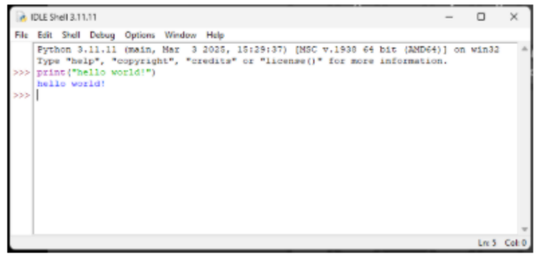
- effective version control

we'll click this later

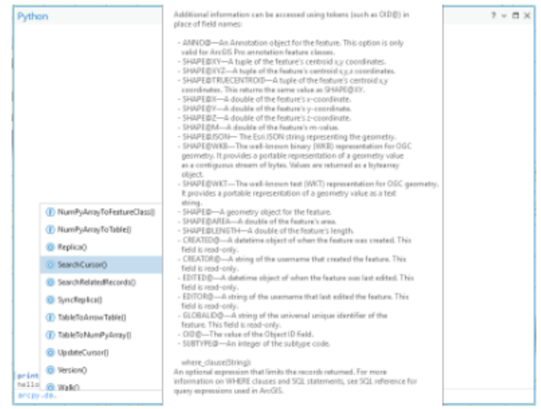
Places you may have previously used Python

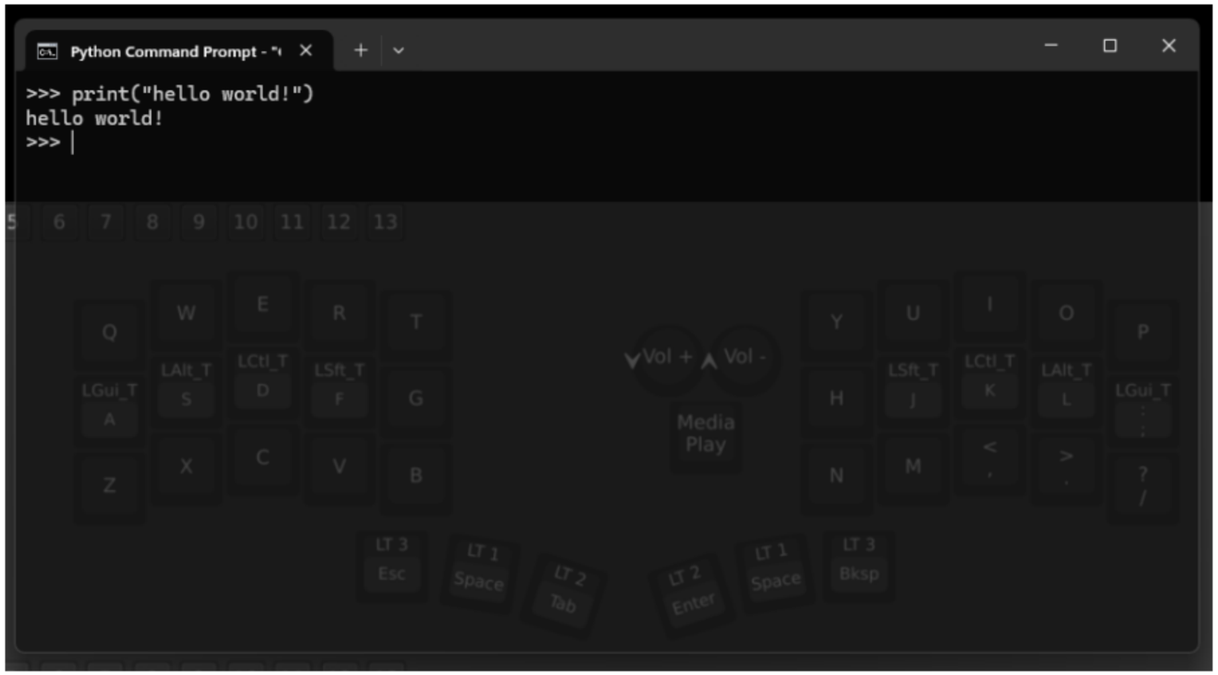
Python Terminal

Python IDLE

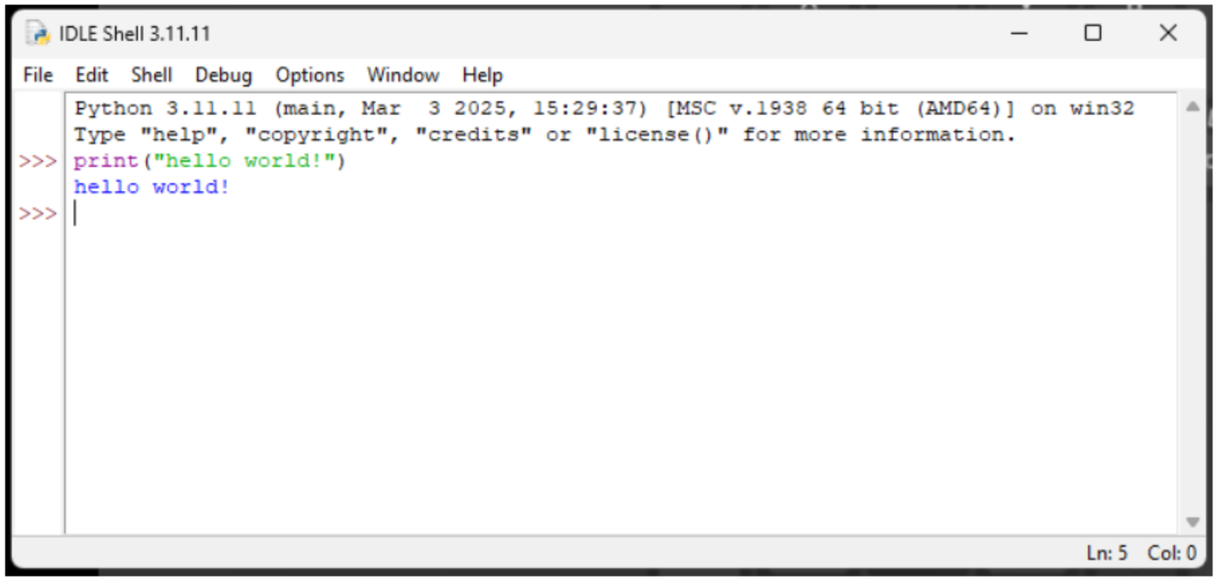


Python Window

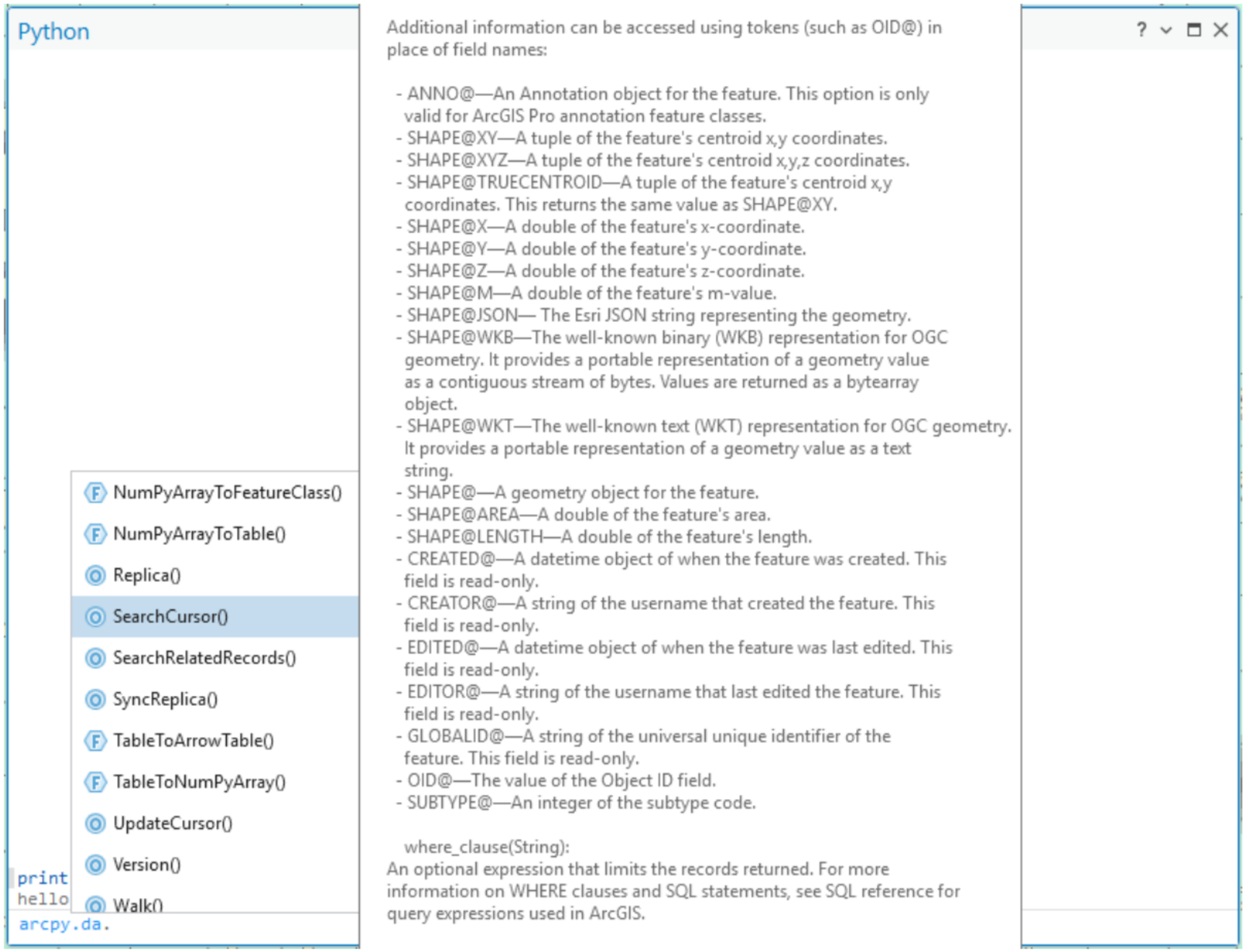




- one-liners
- checking if arcpy will import
- setting keyring keys



- you're *pretty sure* you just need to fix one thing
- quickly reading some code
- checking that something can run



- quick question about the documentation
- playing around with some logic:
  - GP tools
  - metadata
  - CIM attributes
  - etc

- when
- where
- how

when

when to use *The Notebook*

or rather... is *The Notebook* ~~the~~ [an] appropriate format for *this* work?

I'm doing `X` task should I...?

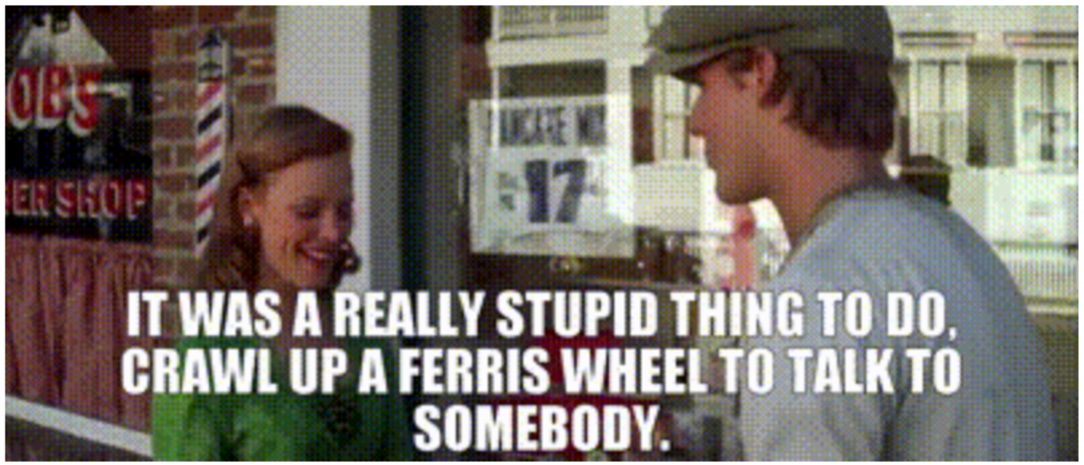
- testing Python syntax or logic ------------ yes
- exploratory data analysis ----------------- yes
- data visualization ------------------------ yes
- data engineering -------------------------- yes
- data science ------------------------------ yes
- benchmarking ------------------------------ yes
- writing a scheduled task ------------------ yes
- repeatable workflow for myself ------------ yes
- repeatable workflow for my team ----------- yes
- repeatable workflow for an end-user -------- yes
- ad hoc tasks ------------------------------ yes
- writing documentation --------------------- yes
- AGO / Portal administration ---------------- yes
- writing a geoprocessing tool --------------- yes*
- presentations ----------------------------- maybe
- running a scheduled task ------------------ maybe
- CI/CD ------------------------------------- no
- production-level cartography --------------- no

* you will eventually need to use a `.py` file

for most everything, the answer is, *likely*, yes.

does that make it, *the best*, format for anything you want to do? No.

- when
- where
- how


IT WAS A REALLY STUPID THING TO DO,
CRAWL UP A FERRIS WHEEL TO TALK TO
SOMEBODY.

where

where to use *The Notebook*

ArcGIS Pro

ArcGIS Online

Jupyter

- Notebook
- Lab
- GIS

IDE

- I only recognize VSCode; there is no other IDE

Google Colab

Marimo

ArcGIS Notebooks in ArcGIS Pro

- shares the main application thread
- variable state conflict
- no auto-complete

ArcGIS Notebooks in ArcGIS Pro

- loading notebooks is painfully slow
- does not implement full jupyter options
- text formatting is wonky at times

ArcGIS Notebooks in ArcGIS Online

- credits: scheduling, arcpy, GPU-support
- session timeouts
- limited access to local files

The Jupyter Ecosystem

- Notebook
- Lab
- Extensions

Notebooks in an IDE

- when
- where
- how

how

Things I've used *The Notebook* for:

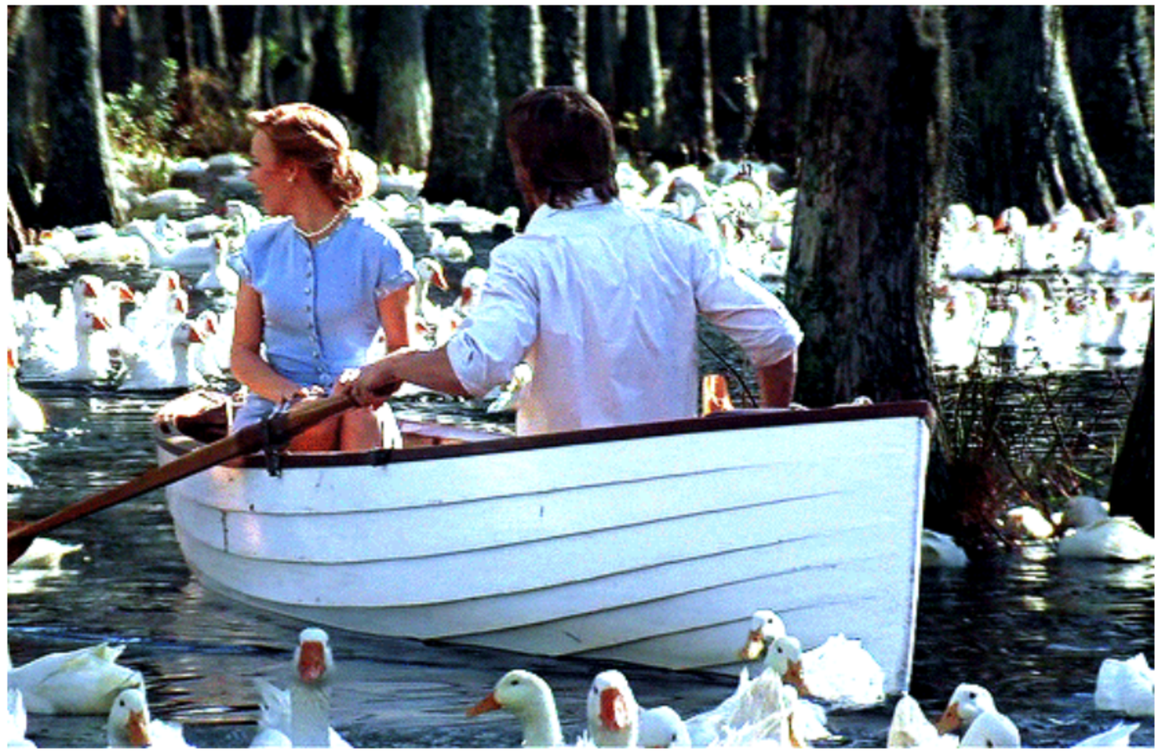
- prototyping scripts
    - noodling through logic
    - exploring variables
    - a function workspace
- irregular processes
    - narrative; hey, you only do this process every x months
    - tools needed
    - any weird manual things or gotchas
    - packaging items as necessary
- documentation
- very specific workflows for users
    - our elections coordinator uses a notebook to load jobs into ArcGIS Workforce
    - dataframe to csv

but really, this presentation isn't about that

it's about all [*The Notebook*] we made along the way

- gotchas to watch out for
- quality of life tweaks to make

. gotchas to watch out for



run order determines the state

[6]:
```
count = 0
```

[5]:
```
count += 1
print(count)
```

```
3
```

[7]:
```
print(count)
```

```
0
```

[ ]:

it's only unexpected if you don't know how it works

[7]:
```
from pathlib import Path

Path(".").resolve()
```

[7]:
```
WindowsPath('C:/Users/jcarmon/OneDrive - Pierce County/Presentations/The Notebook')
```

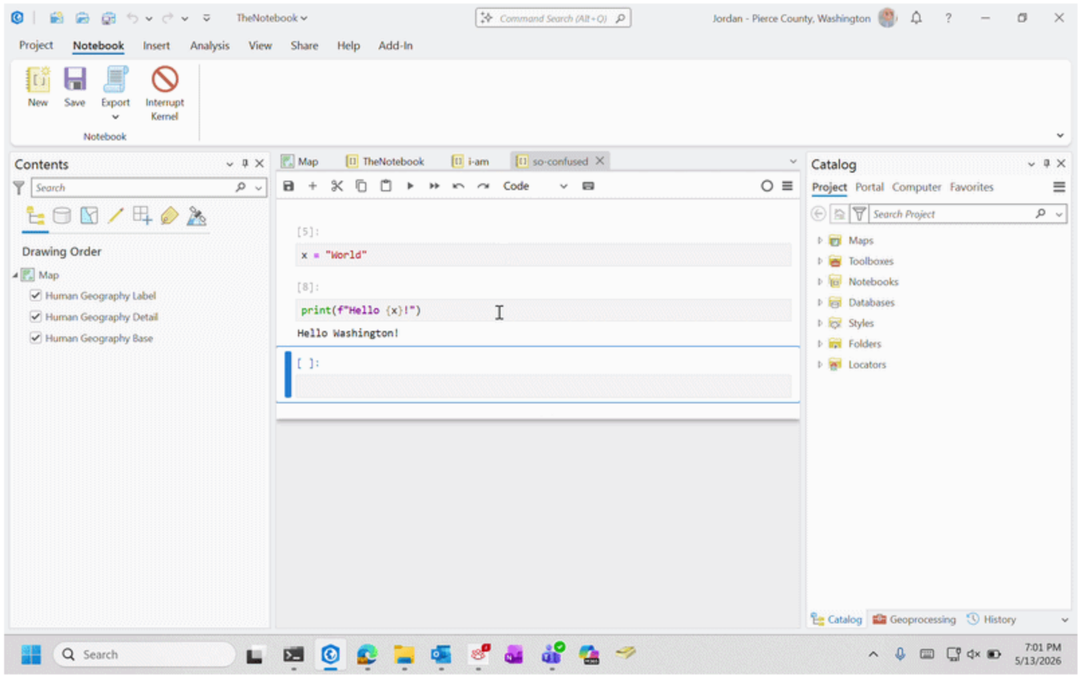
[2]:
```
# you opened ArcGIS Pro using the task bar or search toolbar
from pathlib import Path

Path(".").resolve()
```

[2]:
```
WindowsPath('C:/Windows/System32')
```

[1]:
```
# you opened ArcGIS Pro using an .aprx
from pathlib import Path

Path(".").resolve()
```

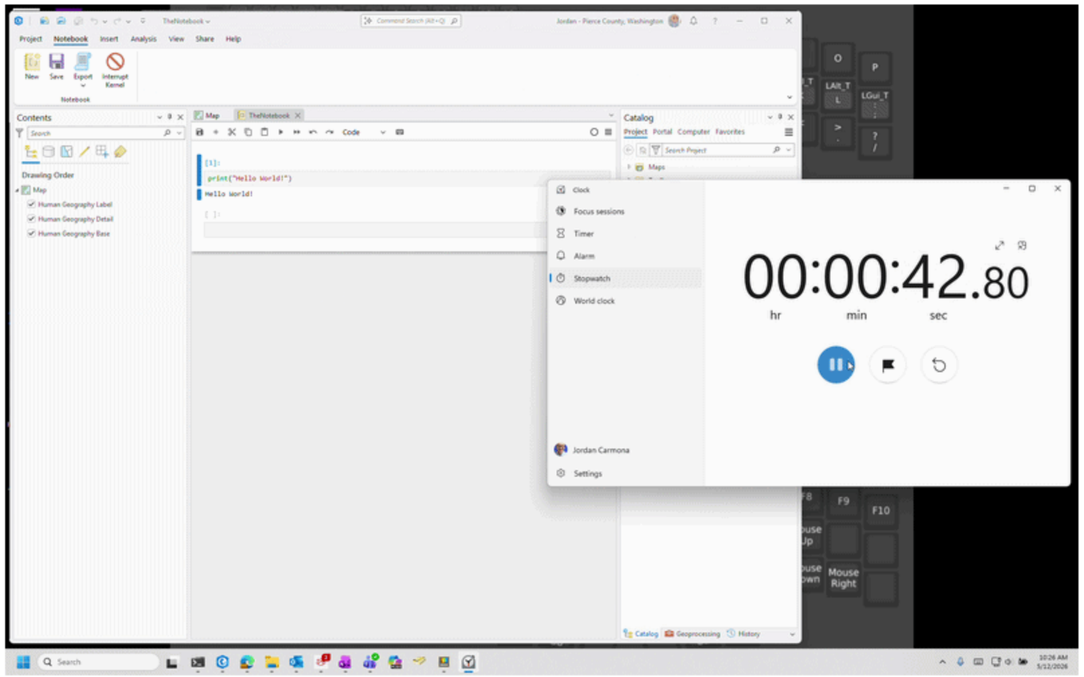
[1]:
```
WindowsPath('C:/Users/jcarmon/Documents/ArcGIS/DAHPProject')
```

basically that spiderman meme



just don’t leave it open

a notebook as a tab in ArcGIS Pro



they get ”old” fast, like, ”long in the tooth”

reading a long notebook is not fun, leave the infinite scrolling to social media

reading a long notebook is not fun, leave the infinite scrolling to social media

the consequences, could be dire

with a lot of convenience functions like `shift+enter` to run a cell, or accidentally clicking a run all cells. If you're not in a development environment, you could accidentally set off an unfortunate series of events

- gotchas to watch out for
- quality of life tweaks to make

. quality of life tweaks to make

use VSCode

- extensions for days
- integrated Copilot
- integrated terminal
- git blame -> version control

ArcGIS Online

- `concurrent.futures`
- `arcgis.notebook.execute_notebook`
- `arcgis.gis.nb._dataaccess.NotebookDataAccess`

concurrent.futures

```
from concurrent.futures import ThreadPoolExecutor

with ThreadPoolExecutor(max_workers=1) as ex:
    print(f"Loading current item dependency graph from .gml at itemid: {graph_item}")
    future_graph = ex.submit(load_from_file, path=temp_graph, gis=gis, include_items=True)
    while not future_graph.done():
        print(
            f"\t[heartbeat] {(time.perf_counter()-_t0)/60:.1f} min elapsed",
            flush=True,
        )
        time.sleep(120)  # a blue whale's heart can slow to one beat every two minutes
item_graph = future_graph.result()
```

```
    future_build_graph = ex.submit(
        create_dependency_graph,
        gis=gis,
        item_list=modified_items,
        outside_org=False,
    )
    while not future_build_graph.done():
        print(
            f"\t[heartbeat] {(time.perf_counter()-start_time)/60:.1f} min elapsed",
            flush=True,
        )
        time.sleep(120)
    modified_graph = future_build_graph.result()
end_time = time.perf_counter()
elapsed_seconds = end_time - start_time
print(f"Org-wide graph created in {(elapsed_seconds / 60):.2f} minutes.")
```

```
        [heartbeat] 0.0 min elapsed
        [heartbeat] 5.0 min elapsed
        [heartbeat] 10.0 min elapsed
        [heartbeat] 15.0 min elapsed
        [heartbeat] 17.0 min elapsed
Org-wide graph created in 19.00 minutes.
```

arcgis.notebook.execute_notebook

execute_notebook

arcgis.notebook.execute_notebook(*item: Item, *, timeout: int = 50, update_portal_item: bool = True, parameters: list = None, save_parameters: bool = False, server_index: int = 0, gis: GIS = None, future: bool = False*) → **dict | Future**

The Execute Notebook operation allows administrators and users with the *Create and Edit Notebooks* privilege to remotely run a notebook that they own. The notebook specified in the operation will be run with all cells in order.

Using this operation, you can schedule the execution of a notebook, either once or with a regular occurrence. This allows you to automate repeating tasks such as data collection and cleaning, content updates, and portal administration. On Linux machines, use a cron job to schedule the executeNotebook operation; on Windows machines, you can use the Task Scheduler app.

https://developers.arcgis.com/python/latest/api-reference/arcgis.gis.nb.html#notebookdataaccess

arcgis.gis.nb._dataaccess.NotebookDataAccess

```
gis = arcgis.gis.GIS()
connected_user = gis.users.me
nb_da_url = connected_user.generate_direct_access_url(store_type="notebook")
nb_daccess = NotebookDataAccess(url=nb_da_url["url"], gis=gis)
nb_daccess.upload(fr"{local_dir}\updating_graph.gml")
```

https://community.esri.com/t5/arcgis-api-for-python-questions/attempting-to-upload-files-to-notebook-files-using/td-p/1288227

https://developers.arcgis.com/python/latest/api-reference/arcgis.gis.nb.html#notebookdataaccess

Dataframes

are just the coolest

[8]:
```
url = "https://services.arcgis.com/6lCKYNJLvwTXqrmp/arcgis/rest/services/TCP/FeatureServer/0"
lyr = arcgis.features.FeatureLayer(url)
sedf = pd.DataFrame.spatial.from_layer(lyr)
sedf = sedf.drop(columns="AlternateSiteNames")
sedf.spatial.to_featureclass(location=r".\arcgispro\the-notebook.geodatabase\main.cleanup_sites", overwrite=True, has_z=None, has_m=None, sanitize_columns=True)
```

[8]:
```
'C:\\Users\\jcarmon\\OneDrive - Pierce County\\Presentations\\The Notebook\\arcgispro\\the-notebook.geodatabase\\main.cleanup_sites'
```

[ ]:

[8]:
```
some_fc = r".\arcgispro\the-notebook.geodatabase\main.cleanup_sites"
fld = arcpy.ListFields(some_fc)
fld
for fld in arcpy.ListFields(some_fc):
    print(fld.name, fld.type)
```

```
OBJECTID OID
Shape Geometry
cleanup_site_id Integer
facility_site_id Integer
cleanup_site_name String
site_status String
site_rank String
has_institutional_control String
current_vcp String
address String
city String
zip_code String
county String
latitude Double
longitude Double
region String
responsible_unit String
```

[10]:
```
sedf = pd.DataFrame.spatial.from_featureclass(r".\arcgispro\the-notebook.geodatabase\main.cleanup_sites")
sedf.dtypes
```

[10]:
```
OBJECTID                        Int64
cleanup_site_id                 Int32
facility_site_id                Int32
cleanup_site_name          string[python]
site_status                string[python]
site_rank                  string[python]
has_institutional_control  string[python]
current_vcp                string[python]
address                    string[python]
city                       string[python]
zip_code                   string[python]
county                     string[python]
latitude                          Float64
longitude                         Float64
region                     string[python]
responsible_unit           string[python]
SHAPE                            geometry
dtype: object
```

set your display appropriately

it's a pandas thing

it's a pandas thing

[11]:
```
sedf
```

[11]:

	OBJECTID	cleanup_site_id	facility_site_id	cleanup_site_name	site_status	...	latitude	longitude	region	responsible_unit	SHAPE
0	1	127	157	US Navy Keyport	Cleanup Started	...	47.70469	-122.62681	Northwest	Headquarters	{"x": 1116749.2756605595, "y": 871885.39742189...
1	2	109	178	US NAVY PSNS OUC	Construction Complete-Performance Monitoring	...	47.561828	-122.643421	Northwest	Headquarters	{"x": 1111244.2666574717, "y": 819898.74024055...
2	3	134	208	American Crossarm & Conduit	Cleanup Complete-Active O&M/Monitoring	...	46.65767	-122.96926	Southwest	Southwest	{"x": 1020502.4720915556, "y": 492640.77500714...
3	4	131	189	FUDS Fort Canby	No Further Action	...	46.27868	-124.05303	Southwest	Headquarters	{"x": 742350.160943225, "y": 364960.9854197204...
4	5	133	207	US NAVY PORT HADLOCK AREA 34	No Further Action	...	48.09305	-122.72338	Southwest	Headquarters	{"x": 1096971.2036682218, "y": 1014180.4310727...
...	...	...	...	...	...	...	...	...	...	...	...
14634	14635	17436	100005795	NB I-5 Exit 49	Awaiting Cleanup	...	46.28332	-122.90186	Southwest	Southwest	{"x": 1033261.9032655507, "y": 355667.61785164...
14635	14636	17406	100005457	ExxonMobil 3448	Awaiting Cleanup	...	47.546336	-122.37676	Northwest	Northwest	{"x": 1176930.7350217998, "y": 812571.41106322...
14636	14637	17438	100005800	Clark PUD Upper Utility Poles ROW	Awaiting Cleanup	...	45.71293	-122.72437	Southwest	Southwest	{"x": 1072251.7928052992, "y": 146437.18362306...
14637	14638	17443	100006086	Still Harbor Dock House	Awaiting Cleanup	...	47.21348	-122.66665	Southwest	Southwest	{"x": 1102020.9091199785, "y": 693038.91029931...
14638	14639	17440	100005889	10160 Oak Bay Release	Cleanup Started	...	47.91827	-122.71432	Southwest	Southwest	{"x": 1097395.0210620612, "y": 950369.70534797...

14639 rows × 17 columns

[12]:
```
pd.options.display.max_columns = None
pd.options.display.max_rows = 5
# lots of options here: https://pandas.pydata.org/docs/user_guide/options.html
```

[ ]:

attribute table who?

let's just never open the GUI again

[ ]:

attribute table who?

let's just never open the GUI again ¶

[11]:
```
# go to the notebook view for this
from ipydatagrid import DataGrid
DataGrid(sedf)
```

[11]:

key	...ECT	...site_	...site_	..._nar	...statu	site_ra	...cont	...nt_v	addres	city	zip_co	county	latitude	longitu	region	...le_u	SHAPI
0	1	127	157	US Na...	Clean...				HWY ...	KEYP...	98345-...	Kitsap	47.704...	-122.6...	North...	Headq...	[object...
1	2	109	178	US NA...	Constr...	0 - NP...			1ST ST	BREM...	98314	Kitsap	47.561...	-122.6...	North...	Headq...	[object...
2	3	134	208	Americ...	Clean...	0 - NP...	True		100 C...	CHEH...	98532	Lewis	46.657...	-122.9...	South...	South...	[object...
3	4	131	189	FUDS ...	No Fur...				1 MI S...	ILWACO	98624	Pacific	46.278...	-124.0...	South...	Headq...	[object...
4	5	133	207	US NA...	No Fur...				INDIA...	PORT ...	98339-...	Jefferson	48.093...	-122.7...	South...	Headq...	[object...
5	6	130	175	US Na...	Clean...		True		1ST ST	BREM...	98314	Kitsap	47.558...	-122.6...	North...	Headq...	[object...
6	7	148	161	US NA...	Constr...				CLEA...	SILVE...	98315	Kitsap	47.763...	-122.7...	North...	Headq...	[object...
7	8	132	190	EDB 2...	Constr...	1 - Hig...			N of S...	Mount ...	98237	Skagit	48.421...	-122.4...	North...	North...	[object...
8	9	150	179	US NA...	Clean...				Farrag...	Breme...		Kitsap	47.561...	-122.6...	North...	Headq...	[object...
9	10	114	235	USAF ...	No Fur...	5 - Lo...			62 CE...	MCCH...	98438-...	Pierce	47.0901	-122.4...	South...	Headq...	[object...
10	11	149	168	US Na...	Clean...				CLEA...	SILVE...	98315	Kitsap	47.699...	-122.7...	North...	Headq...	[object...
11	12	113	212	Puyall...	Tracke...	0 - NP...			250-36...	TACO...	98421	Pierce	47.264...	-122.3...	South...	EPA	[object...
12	13	116	244	REST...	No Fur...	3 - Mo...			2725 9...	OLYM...	98512	Thurston	46.951...	-122.9...	South...	South...	[object...
13	14	115	237	USAF ...	No Fur...				62 CE...	MCCH...	98438-...	Pierce	47.0901	-122.4...	South...	Headq...	[object...
14	15	152	193	EDB 3...	Constr...	3 - Mo...			NORT...	LYNDEN	98264	Whatc...	48.963...	-122.4...	North...	North...	[object...
15	16	110	201	US BP...	No Fur...	0 - NP...			5411 N...	VANC...	98666	Clark	45.658...	-122.6...	South...	EPA	[object...
16	17	112	210	US C...	No Fur...				100 F...	ILWACO	98624	Pacific	46.282...	-124.0...	South...	Headq...	[object...
17	18	120	305	EAST...	No Fur...				E 10T...	KENN...	99336	Benton	46.197...	-119.0...	Central	Central	[object...
18	19	107	171	US Na...	No Fur...		True		ORCH...	MANC...	98353	Kitsap	47.563...	-122.5...	North...	Headq...	[object...
19	20	135	222	Well 12A	Constr...				3011 S...	TACO...	98421	Pierce	47.231...	-122.4...	South...	EPA	[object...
20	21	108	176	US NA...	Constr...				1ST ST	BREM...	98314	Kitsap	47.552...	-122.6...	North...	Headq...	[object...
21	22	137	239	USAF ...	Constr...				62AB...	MCCH...	98438-...	Pierce	47.133...	-122.5...	South...	EPA	[object...
22	23	128	158	US Na...	Constr...				610 D...	KEYP...	98345-...	Kitsap	47.698...	-122.6...	North...	Headq...	[object...
23	24	106	169	US Na...	No Fur...		True		ORCH...	MANC...	98353	Kitsap	47.563...	-122.5...	North...	Headq...	[object...

[ ]:

map who?

I actually never do this. I never need to visualize things in a notebook. If you wanted to, you could.

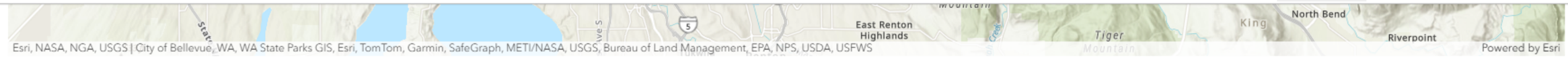
[12]:
```
# go to the notebook view for this
from arcgis.map import Map

new_map = Map(location="Bellevue, WA")
sedf_filtered = sedf[sedf["city"].str.upper() == "BELLEVUE"].copy()
new_map.content.add(sedf_filtered)
new_map
```

[12]:

map who?

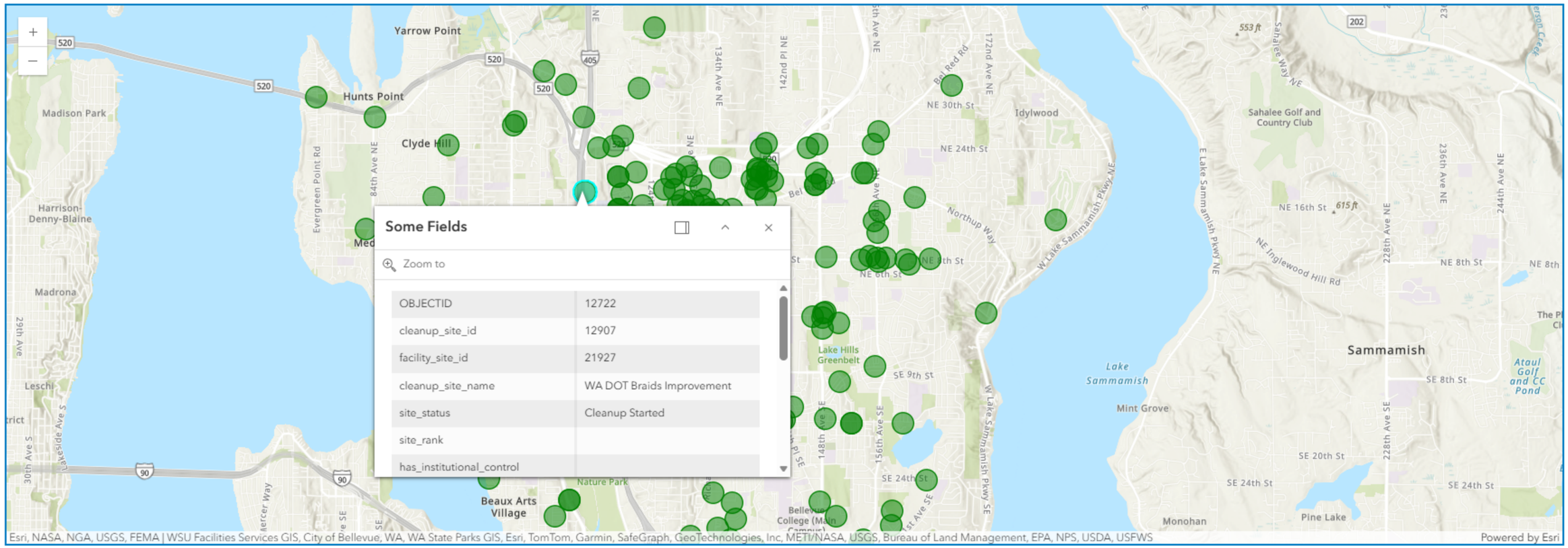
I actually never do this. I never need to visualize things in a notebook. If you wanted to, you could.

I actually never do this. I never need to visualize things in a notebook. If you wanted to, you could.



[13]:
```
from arcgis.map import Map
from arcgis.map.popups import PopupInfo, FieldInfo

new_map = Map(location="Bellevue, WA")
sedf_filtered = sedf[sedf["city"].str.upper() == "BELLEVUE"].copy()
new_map.content.add(sedf_filtered)
pop_up = new_map.content.popup(0)
pop_up.disable_popup = False
fields = [FieldInfo(field_name=column_name) for column_name in sedf_filtered.columns]
pop_up.edit("Some Fields", field_infos=fields)
new_map
```

[13]:



If it must be long

Code Folding

[16]:
```
def howdy():
    print("Hello World!")
    return
```

If it must be long

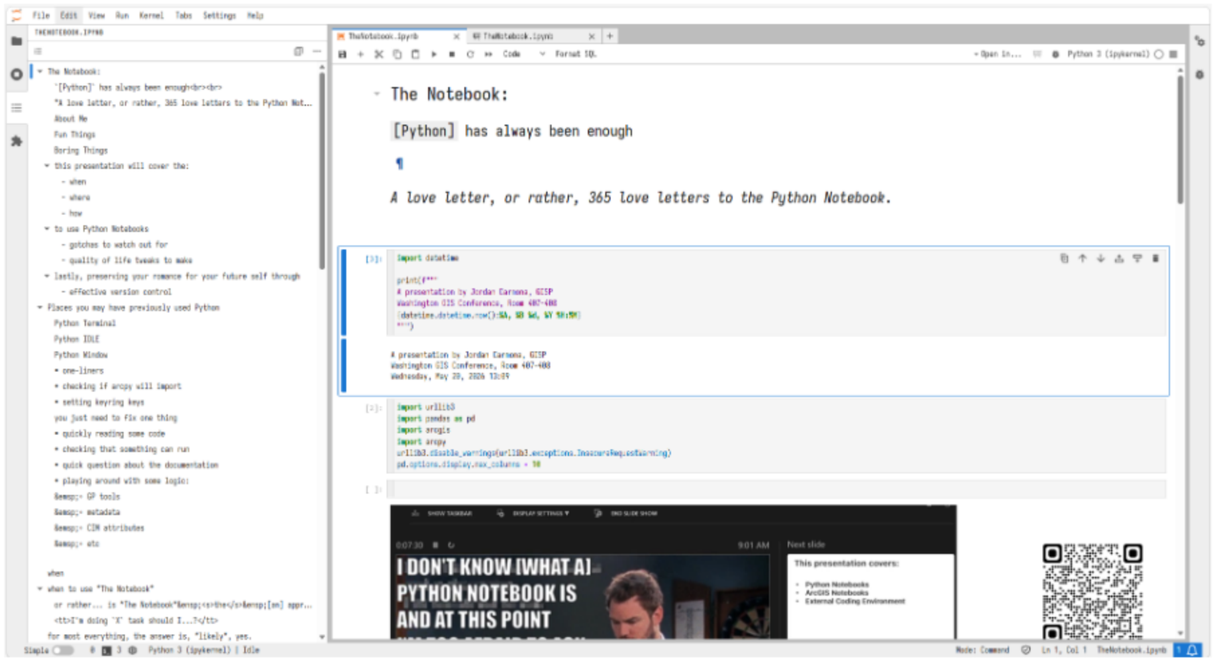
Code Folding

[16]:
```
def howdy():...


howdy()
```

```
Hello World!
```

[ ]:
```

```

Markdown Table of Contents



Notebook Extensions

- jupyter RISE (this kind of thing)
- jupyter slides (an html export of a jupyter notebook)
- pytutor
- variable inspector

https://jupyterlab-contrib.github.io/extensions.html

Notebook Magic

You may not know it, but you and me go waaaaaay back

[17]:
```
%run arcgispro/so-confused.ipynb
```

```
Hello World!
```

[18]:
```
%load_ext sql
```

[19]:
```
import os
from sqlalchemy import create_engine
db_path = os.path.abspath("./arcgispro/the-notebook.geodatabase")
engine = create_engine(f'sqlite+pysqlite:///{db_path}')
%sql engine
%config SqlMagic.displaylimit = 10
```

[20]:

```
%%sql
SELECT
  facility_site_id,
  cleanup_site_name,
  site_status,
  address,
  city,
  county,
  *
FROM
  main.cleanup_sites
WHERE
  OBJECTID IN (
    SELECT
      MAX(OBJECTID)
    FROM
      main.cleanup_sites
    GROUP BY
      facility_site_id
  )
  AND LOWER(city) = "bellevue"
ORDER BY
  facility_site_id
```

Running query in 'sqlite+pysqlite:///C:\Users\jcarmon\OneDrive - Pierce County\Presentations\The Notebook\arcgispro\the-notebook.geodatabase'

[20]:

facility_site_id	cleanup_site_name	site_status	address	city	county	OBJECTID	Shape	cleanup_site_id	facility_site_id_1	cleanup_site_name_1	site_status_1	site_rank	has
2017	Eastgate Landfill	Cleanup Started	2805 160th Ave SE	Bellevue	King	4250	b'do\x0b\x00\x01\x00\x00\x00\x04\x01\x0c\x00\x00\x00\x01\x00\x00\x00\xa5\xce\x9f\xb8\x88\x15\x89\xe6\xb8\xc9\xd7\x11'	4425	2017	Eastgate Landfill	Cleanup Started		
2090	Bellevue Plating Co Inc	No Further Action	1513 128TH PL NE	BELLEVUE	King	2498	b'do\x0b\x00\x01\x00\x00\x00\x04\x01\x0c\x00\x00\x00\x01\x00\x00\x00\x99\xf9\xf9\x9b\x88\x15\x98\xf2\xc6\xf3\xd7\x11'	791	2090	Bellevue Plating Co Inc	No Further Action		
2113	Crossroads Dry Cleaners	No Further Action	NE 8TH & 156TH AVE	BELLEVUE	King	153	b'do\x0b\x00\x01\x00\x00\x00\x04\x01\x0c\x00\x00\x00\x01\x00\x00\x00\xa4\x9d\xe2\xb6\x88\x15\x8c\x9b\xe5\xeb\xd7\x11'	83	2113	Crossroads Dry Cleaners	No Further Action		
2119	Factoria Pit Sunset Park	Awaiting Cleanup	132ND AVE SE & SE 38TH ST	BELLEVUE	King	102	b'do\x0b\x00\x01\x00\x00\x00\x04\x01\x0c\x00\x00\x00\x01\x00\x00\x00\x83\x81\xc6\xa0\x88\x15\xb2\xf6\x8c\xc0\xd7\x11'	35	2119	Factoria Pit Sunset Park	Awaiting Cleanup		
2162	Poles Incorporated	No Further Action	826 102ND AVE NE	BELLEVUE	King	2460	b'do\x0b\x00\x01\x00\x00\x00\x04\x01\x0c\x00\x00\x00\x01\x00\x00\x00\xaa\x88\xb4\x83\x88\x15\x88\xb4\xfc\xed\xd7\x11'	827	2162	Poles Incorporated	No Further Action		
2310	King County Metro Bellevue	No Further Action	2000 118TH AVE SE	BELLEVUE	King	4553	b'do\x0b\x00\x01\x00\x00\x00\x04\x01\x0c\x00\x00\x00\x01\x00\x00\x00\x9f\xca\xa5\x92\x88\x15\xa2\x94\xf2\xd1\xd7\x11'	5086	2310	King County Metro Bellevue	No Further Action		
2319	Town & Country Cleaners Bellevue	Cleanup Started	310 105th Ave NE	Bellevue	King	1040	b'do\x0b\x00\x01\x00\x00\x00\x04\x01\x0c\x00\x00\x00\x01\x00\x00\x00\xba\xfb\xaf\x86\x88\x15\x9e\xc9\xbc\xe8\xd7\x11'	1880	2319	Town & Country Cleaners Bellevue	Cleanup Started		
2352	Tiki Car Wash	Cleanup Started	11909 NE 8TH ST	BELLEVUE	King	4585	b'do\x0b\x00\x01\x00\x00\x00\x04\x01\x0c\x00\x00\x00\x01\x00\x00\x00\xab\x90\xcf\x93\x88\x15\x99\xab\x95\xec\xd7\x11'	5096	2352	Tiki Car Wash	Cleanup Started	3 - Moderate Risk	
2358	The Court at Crossroads	No Further Action	15616 NE 15TH	BELLEVUE	King	1077	b'do\x0b\x00\x01\x00\x00\x00\x04\x01\x0c\x00\x00\x00\x01\x00\x00\x00\xb6\xa1\xa1\xb7\x88\x15\x8e\xb4\xa1\xf2\xd7\x11'	1996	2358	The Court at Crossroads	No Further Action		
2403	Trivestors Partnership Property	No Further Action	12700 NE BEL RED RD	BELLEVUE	King	2284	b'do\x0b\x00\x01\x00\x00\x00\x04\x01\x0c\x00\x00\x00\x01\x00\x00\x00\xa6\xcc\x92\x9b\x88\x15\xa3\xf7\xa6\xf0\xd7\x11'	2932	2403	Trivestors Partnership Property	No Further Action		

Truncated to displaylimit of 10.

[ ]:

[21]:
```
%whos
```

```
Variable          Type             Data/Info
-----------------------------------------------
DataGrid          MetaHasTraits    <class 'ipydatagrid.datagrid.DataGrid'>
FieldInfo         ModelMetaclass   <class 'arcgis.map.datacl<...>models.popups.FieldInfo'>
Map               MetaHasTraits    <class 'arcgis.map.map_widget.Map'>
NamespaceMagics   MetaHasTraits    <class 'IPython.core.magi<...>mespace.NamespaceMagics'>
Path              type             <class 'pathlib.Path'>
PopupInfo         ModelMetaclass   <class 'arcgis.map.datacl<...>models.popups.PopupInfo'>
arcgis            module           <module 'arcgis' from 'C:<...>es\\arcgis\\__init__.py'>
arcpy             module           <module 'arcpy' from 'C:\<...>cPy\\arcpy\\__init__.py'>
collections       module           <module 'collections' fro<...>ollections\\__init__.py'>
count             int              1
create_engine     function         <function create_engine at 0x0000023ED6DEE160>
datetime          module           <module 'datetime' from '<...>y-3-5\\Lib\\datetime.py'>
db_path           str              C:\Users\jcarmon\OneDrive<...>\the-notebook.geodatabase
engine            Engine           Engine(sqlite+pysqlite://<...>the-notebook.geodatabase)
fields            list             n=17
fld               list             n=17
get_ipython       function         <function get_ipython at 0x0000023EA341D9E0>
howdy             function         <function howdy at 0x0000023EB6E01DA0>
islice            type             <class 'itertools.islice'>
json              module           <module 'json' from 'C:\\<...>\Lib\\json\\__init__.py'>
lyr               FeatureLayer     <FeatureLayer url:"https:<...>ces/TCP/FeatureServer/0">
math              module           <module 'math' (built-in)>
new_map           Map              Map(center=[6042827.61214<...>: 3857, 'wkid': 102100}})
numpy             module           <module 'numpy' from 'C:\<...>ges\\numpy\\__init__.py'>
os                module           <module 'os' from 'C:\\Us<...>\\arcpy-3-5\\Lib\\os.py'>
pd                module           <module 'pandas' from 'C:<...>es\\pandas\\__init__.py'>
pop_up            PopupManager     PopupManager for: a0bf78
sedf              DataFrame              OBJECTID  cleanup_<...>[14639 rows x 17 columns]
sedf_filtered     DataFrame              OBJECTID  cleanup_<...>\n[281 rows x 17 columns]
some_fc           str              .\arcgispro\the-notebook.<...>tabase\main.cleanup_sites
sys               module           <module 'sys' (built-in)>
time              module           <module 'time' (built-in)>
url               str              https://services.arcgis.c<...>vices/TCP/FeatureServer/0
urllib3           module           <module 'urllib3' from 'C<...>s\\urllib3\\__init__.py'>
x                 str              World
```

[ ]:

[22]:
```
%pip install lckr_jupyterlab_variableinspector
```

```
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: lckr_jupyterlab_variableinspector in c:\users\jcarmon\appdata\roaming\python\python311\site-packages (3.2.4)
Note: you may need to restart the kernel to use updated packages.
```

[ ]:

[23]:
```
%history
```

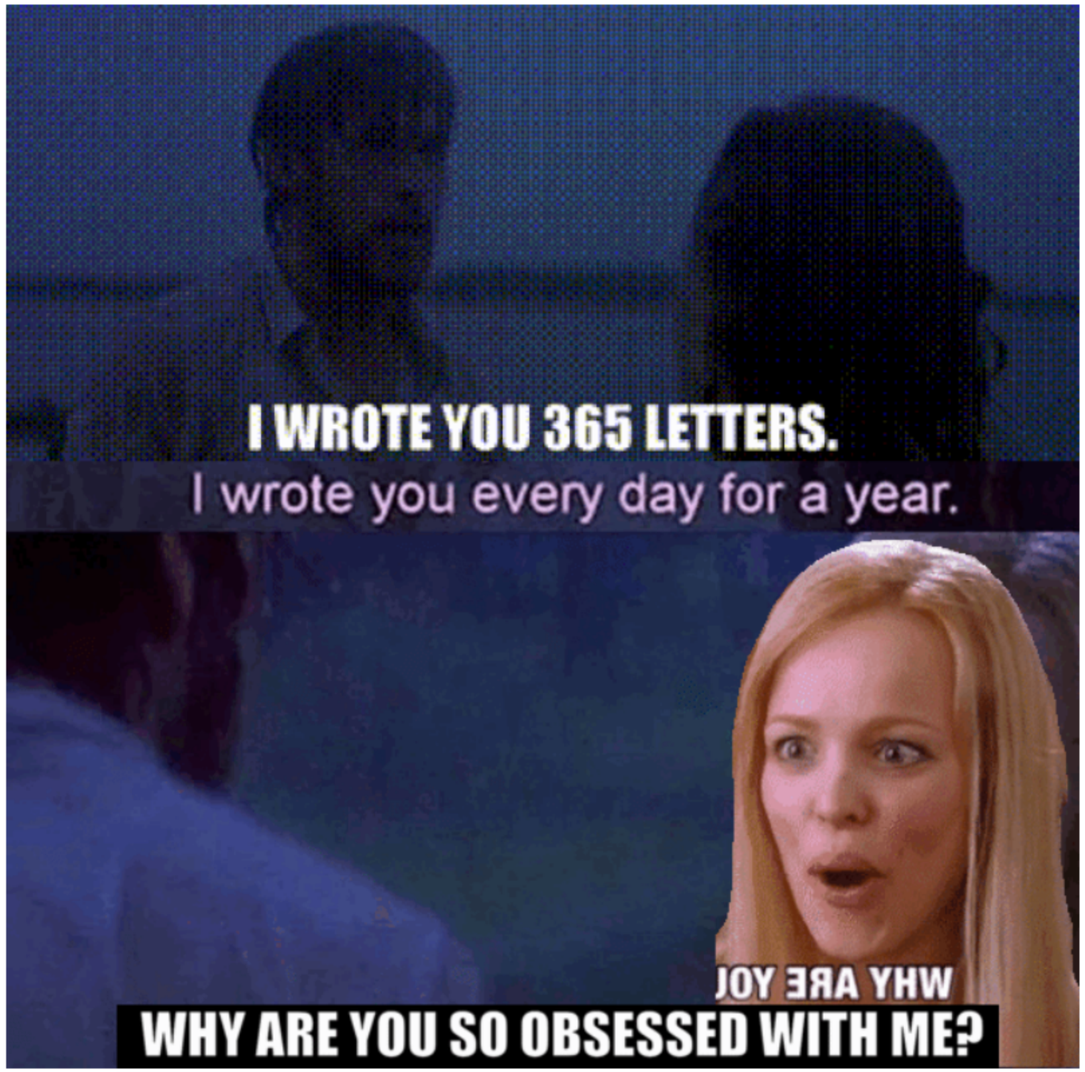
```
import urllib3
import pandas as pd
import arcgis
import arcpy
urllib3.disable_warnings(urllib3.exceptions.InsecureRequestWarning)
pd.options.display.max_columns = 10
import datetime

print(f"""
A presentation by Jordan Carmona, GISP
Washington GIS Conference, Room 407-408
{datetime.datetime.now():%A, %B %d, %Y %H:%M}
""")
import urllib3
import pandas as pd
import arcgis
import arcpy
urllib3.disable_warnings(urllib3.exceptions.InsecureRequestWarning)
pd.options.display.max_columns = 10
count = 0
count += 1
print(count)
print(count)
from pathlib import Path

Path(".").resolve()
url = "https://services.arcgis.com/6lCKYNJLvwTXqrmp/arcgis/rest/services/TCP/FeatureServer/0"
lyr = arcgis.features.FeatureLayer(url)
sedf = pd.DataFrame.spatial.from_layer(lyr)
sedf = sedf.drop(columns="AlternateSiteNames")
sedf.spatial.to_featureclass(location=r".\arcgispro\the-notebook.geodatabase\main.cleanup_sites", overwrite=True, has_z=None, has_m=None, sanitize_columns=True)
some_fc = r".\arcgispro\the-notebook.geodatabase\main.cleanup_sites"
fld = arcpy.ListFields(some_fc)
fld
#for fld in arcpy.ListFields(some_fc):
#    print(fld.name, fld.type)
sedf = pd.DataFrame.spatial.from_featureclass(r".\arcgispro\the-notebook.geodatabase\main.cleanup_sites")
sedf.dtypes
sedf
pd.options.display.max_columns = None
pd.options.display.max_rows = 5
# lots of options here: https://pandas.pydata.org/docs/user_guide/options.html
# go to the notebook view for this
from ipydatagrid import DataGrid
DataGrid(sedf)
# go to the notebook view for this
from arcgis.map import Map

new_map = Map(location="Bellevue, WA")
sedf_filtered = sedf[sedf["city"].str.upper() == "BELLEVUE"].copy()
new_map.content.add(sedf_filtered)
new_map
from arcgis.map import Map
from arcgis.map.popups import PopupInfo, FieldInfo

new_map = Map(location="Bellevue, WA")
sedf_filtered = sedf[sedf["city"].str.upper() == "BELLEVUE"].copy()
new_map.content.add(sedf_filtered)
pop_up = new_map.content.popup(0)
pop_up.disable_popup = False
fields = [FieldInfo(field_name=column_name) for column_name in sedf_filtered.columns]
pop_up.edit("Some Fields", field_infos=fields)
new_map
def howdy():
    print("Hello World!")
    return

howdy()
%run arcgispro/so-confused.ipynb
```

- effective version control

. effective version control

you version your data

you should version your code


I WROTE YOU 365 LETTERS.
I wrote you every day for a year.
WHY ARE YOU SO OBSESSED WITH ME?

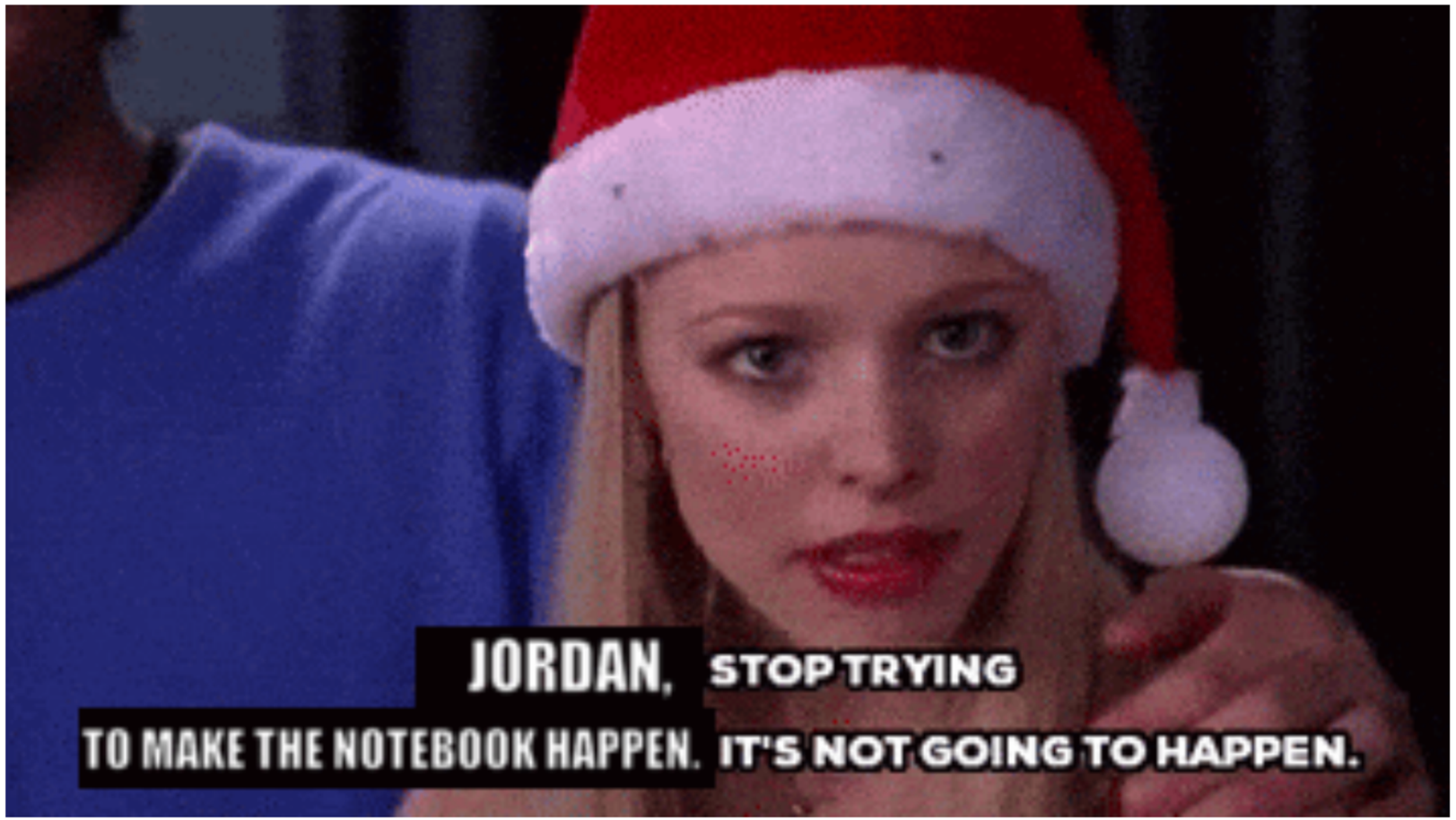
[24]:
```
%load arcgispro/so-confused.ipynb
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[24], line 32
      1 # %load arcgispro/so-confused.ipynb
      2 {
      3  "cells": [
      4   {
      5    "cell_type": "code",
      6    "execution_count": 5,
      7    "metadata": {},
      8    "outputs": [],
      9    "source": [
     10     "x = \"World\""
     11    ]
     12   },
     13   {
     14    "cell_type": "code",
     15    "execution_count": 8,
     16    "metadata": {},
     17    "outputs": [
     18     {
     19      "name": "stdout",
     20      "output_type": "stream",
     21      "text": [
     22       "Hello Washington!\n"
     23      ]
     24     }
     25    ],
     26    "source": [
     27     "print(f\"Hello {x}!\")"
     28    ]
     29   },
     30   {
     31    "cell_type": "code",
---> 32    "execution_count": null,
     33    "metadata": {},
     34    "outputs": [],
     35    "source": []
     36   }
     37  ],
     38  "metadata": {
     39   "kernelspec": {
     40    "display_name": "ArcGISPro",
     41    "language": "python",
     42    "name": "python3"
     43   },
     44   "language_info": {
     45    "file_extension": ".py",
     46    "mimetype": "text/x-python",
     47    "name": "python",
     48    "version": "3.11.10"
     49   }
     50  },
     51  "nbformat": 4,
     52  "nbformat_minor": 4
     53 }

NameError: name 'null' is not defined
```

[ ]:

jupytext

click me



linkedin.com/in/jordancarmona

jordan.carmona@piercecountywa.gov

[ ]: