ArcGIS Pro: Scripting with Python

Andrew Chapkowski

2019 ESRI
FEDERAL GIS CONFERENCE
WASHINGTON, DC

Overview

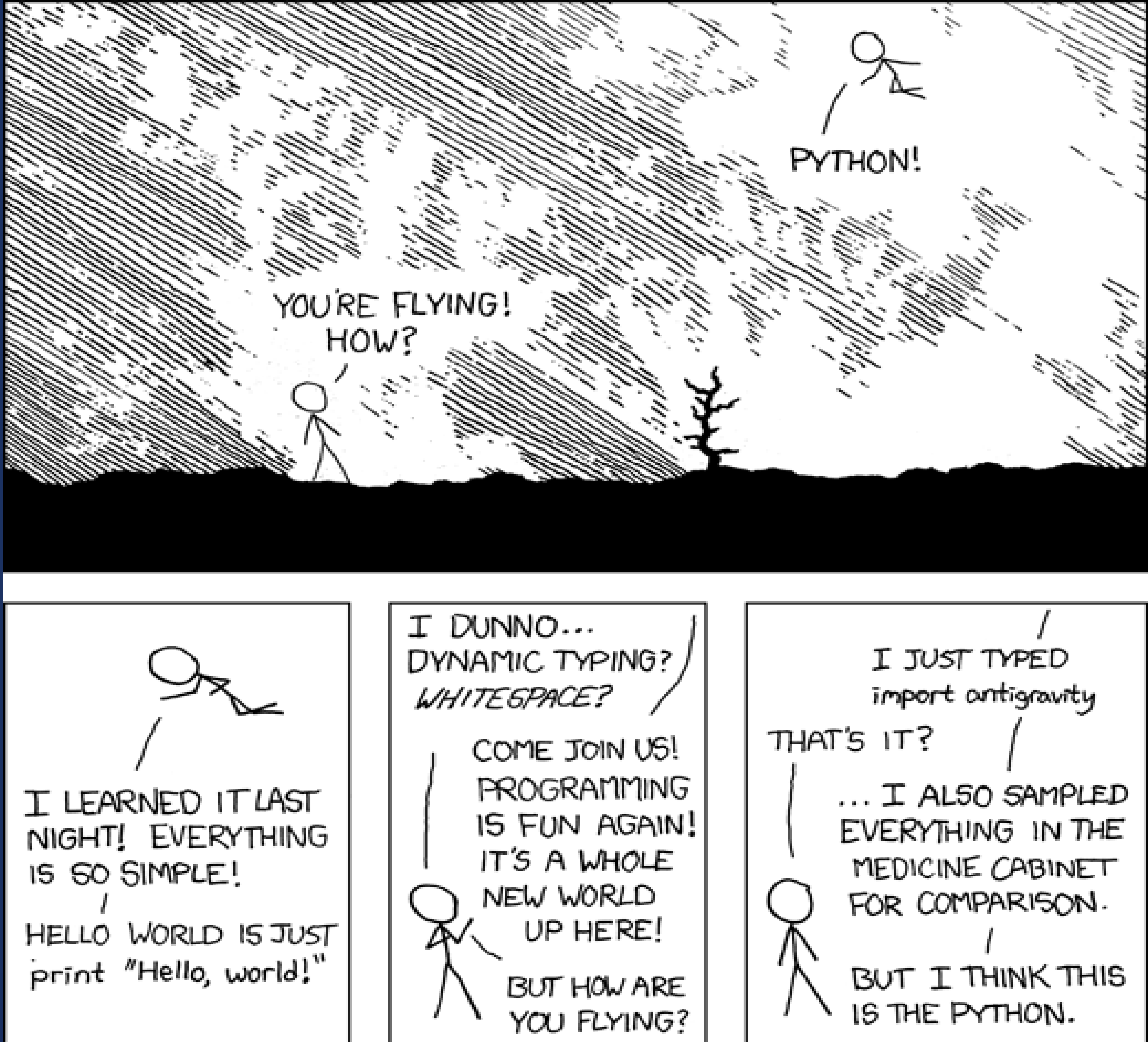
- Preliminaries
  - Why Python?
  - Essential Vocabulary
  - What is ArcPy?
  - Moving from PY2 to PY3

- Python Toolboxes
  - Best Practices
  - Creating Toolboxes
  - Testing

Preliminaries

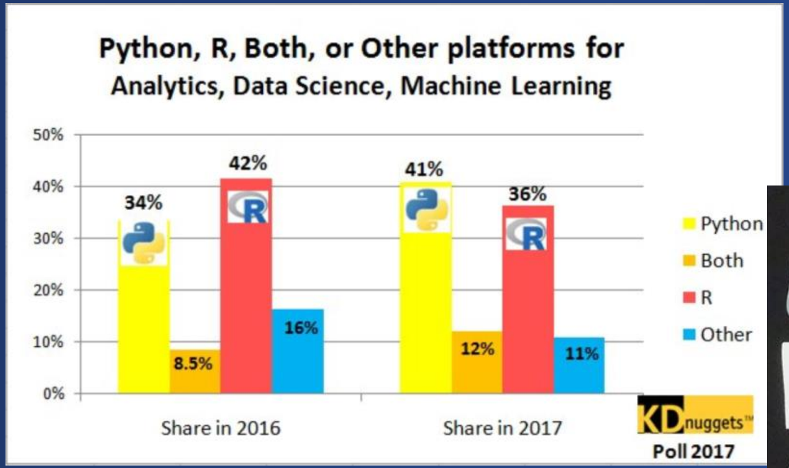
Why Python?

- Why Python?


YOU'RE FLYING! HOW?
PYTHON!
I LEARNED IT LAST NIGHT! EVERYTHING IS SO SIMPLE!
HELLO WORLD IS JUST print "Hello, world!"
I DUNNO... DYNAMIC TYPING? WHITESPACE?
COME JOIN US! PROGRAMMING IS FUN AGAIN! IT'S A WHOLE NEW WORLD UP HERE!
BUT HOW ARE YOU FLYING?
I JUST TYPED import antigravity
THAT'S IT?
... I ALSO SAMPLED EVERYTHING IN THE MEDICINE CABINET FOR COMPARISON.
BUT I THINK THIS IS THE PYTHON.

Why Python?

- Popularity
- Productivity
- Interoperability
- Solves the “two-language” problem
- Scientific Python ecosystem
- Community



CAME FOR THE LANGUAGE STAYED FOR THE COMMUNITY

pandas

$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$

scikit learn

machine learning in Python

jupyter

python

NumPy

OpenCV

TensorFlow

matplotlib

What is ArcPy?

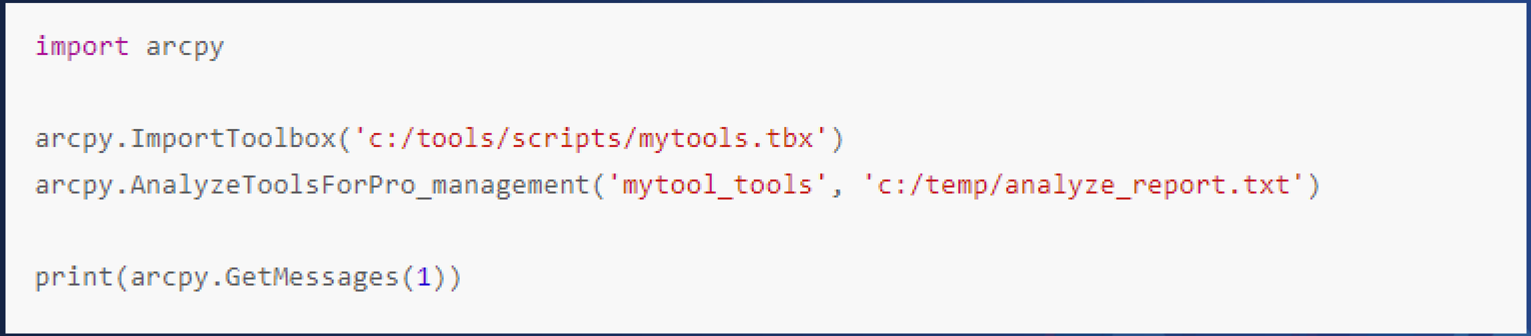
- Python Package for ArcGIS Desktop
- Provides a way to perform geographic data analysis
- Provides a rich and native Python experience
- Work can be done in Toolboxes and Scripts

Essential Vocabulary

- Package/Library – a collection of Python code containing logic. An import statement is used to access this logic
- Module – a file ending in .py. Contains Python logic
- Feature Class – a spatial table
- Parameter – A parameter is a special kind of variable that is used to pass information between functions or procedures
- Exception – an error
- Variable – an assigned space in memory (RAM) containing information

Moving from ArcMap to ArcGIS Pro

- ArcGIS Pro uses Python 3.x
- Most code works out of the box!
- Checking compatibility
  - 2to3 – Python’s upgrade checker
  - Analyze Tools For Pro – ArcGIS Pro’s Python Validator



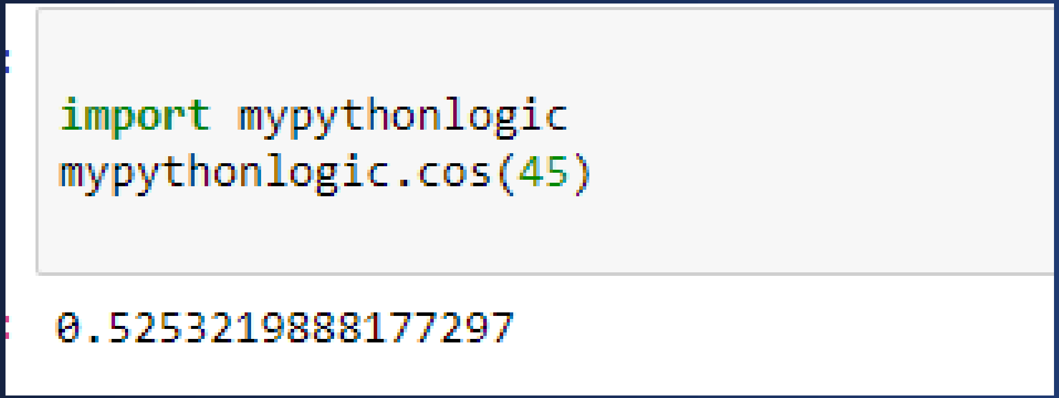
```python
import arcpy

arcpy.ImportToolbox('c:/tools/scripts/mytools.tbx')
arcpy.AnalyzeToolsForPro_management('mytool_tools', 'c:/temp/analyze_report.txt')

print(arcpy.GetMessages(1))
```

Python Toolboxes

```
            , user1, user2):
    elf.filter(from_user=user1, to_user=user2).
        return True
    if self.filter(from_user=user2, to_user=user1).
        return True
    return False

def remove(self, user1, user2):
    """
    Deletes proper object regardless of the order of
    """
    connection = self.filter(from_user=user1, to_user=
    if not connection:
        connection = self.filter(from_user=user2, to_u
    connection.delete()
--:---  models.py        Top l1
```

What is a Python Toolbox?

- Geoprocessing toolboxes that are created entirely in Python
- Look and behave like other toolboxes
- The Python Toolbox is an ascii file
- Supports logic in PYT file or from .py file



```
import mypythonlogic
mypythonlogic.cos(45)
```

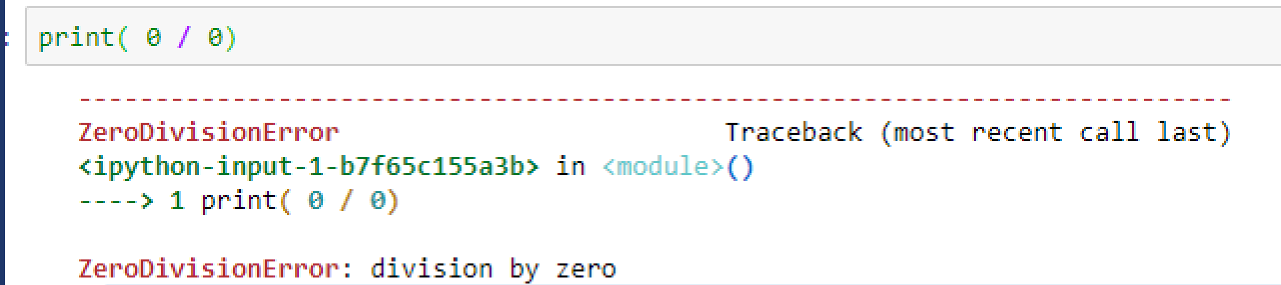
```
0.5253219888177297
```

What Goes in the Toolbox?

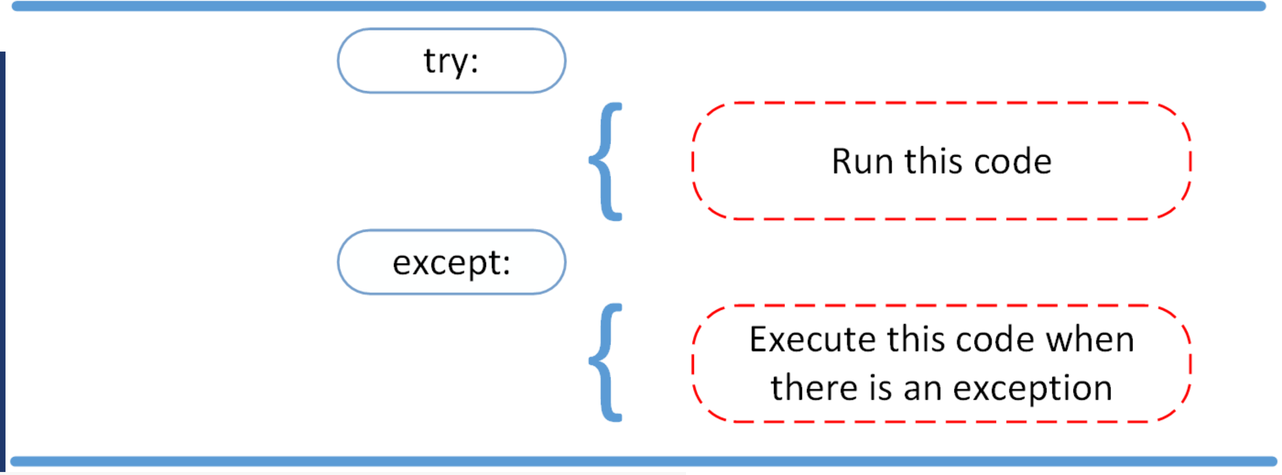
- Error handling
- Logging
- Code documentation
- Python logic

Error Handling

- **The try/except blocks handle allow for the handling of exceptions**
- **Python executes the code in the ‘try’ first and if an error is raised, then it does the ‘except block**



```
print( 0 / 0)

---------------------------------------------------------------------------
ZeroDivisionError                         Traceback (most recent call last)
<ipython-input-1-b7f65c155a3b> in <module>()
----> 1 print( 0 / 0)

ZeroDivisionError: division by zero
```

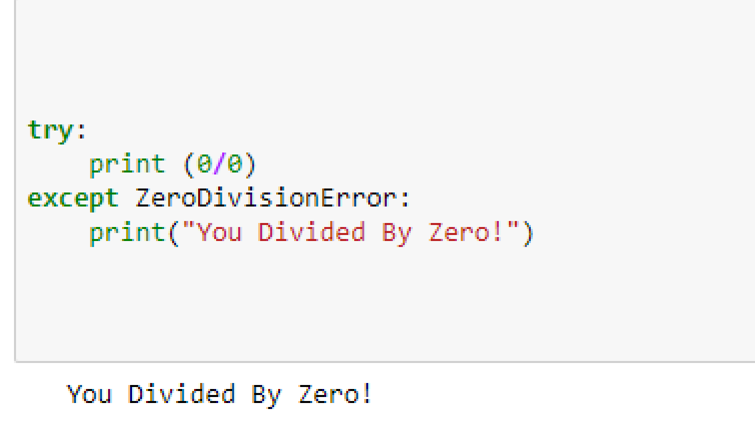


try:

Run this code

except:

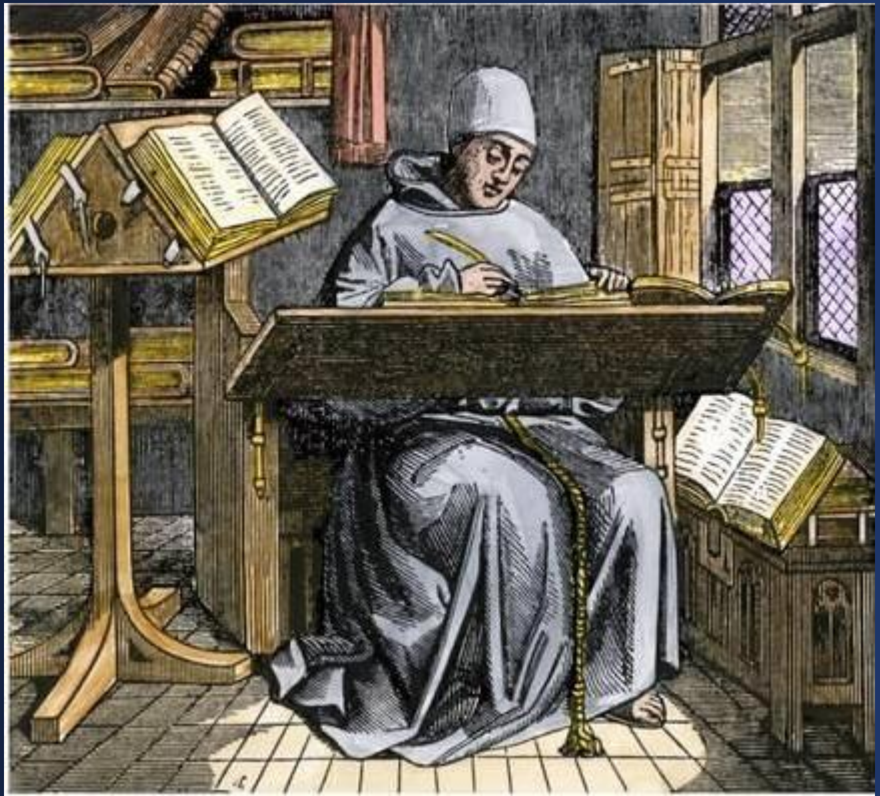
Execute this code when there is an exception



```
try:
    print (0/0)
except ZeroDivisionError:
    print("You Divided By Zero!")
```

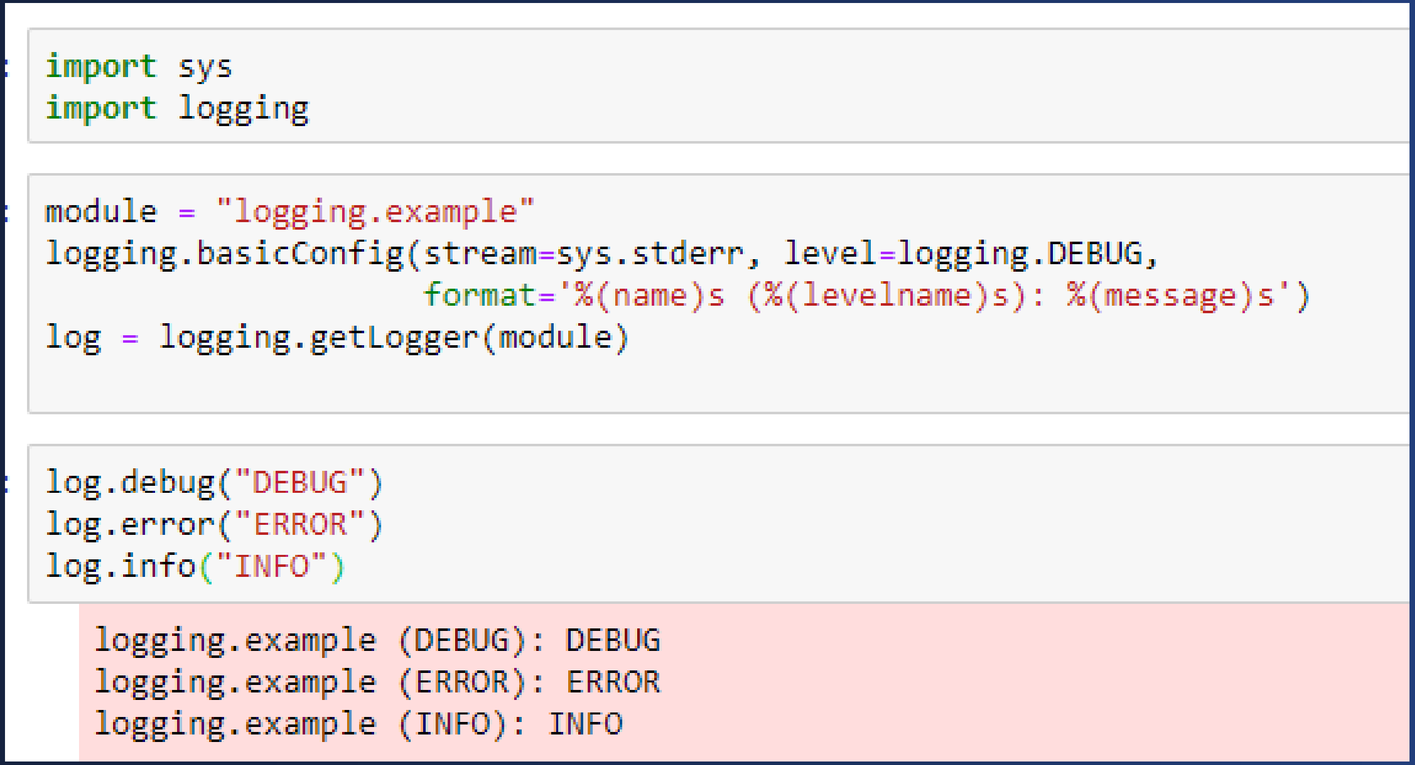
```
You Divided By Zero!
```

Logging



- Logging provides insight to how your code is running
- Captures information when you are not around!
- When you log information it should help in debugging an issue quicker

Logging



```
import sys
import logging
```

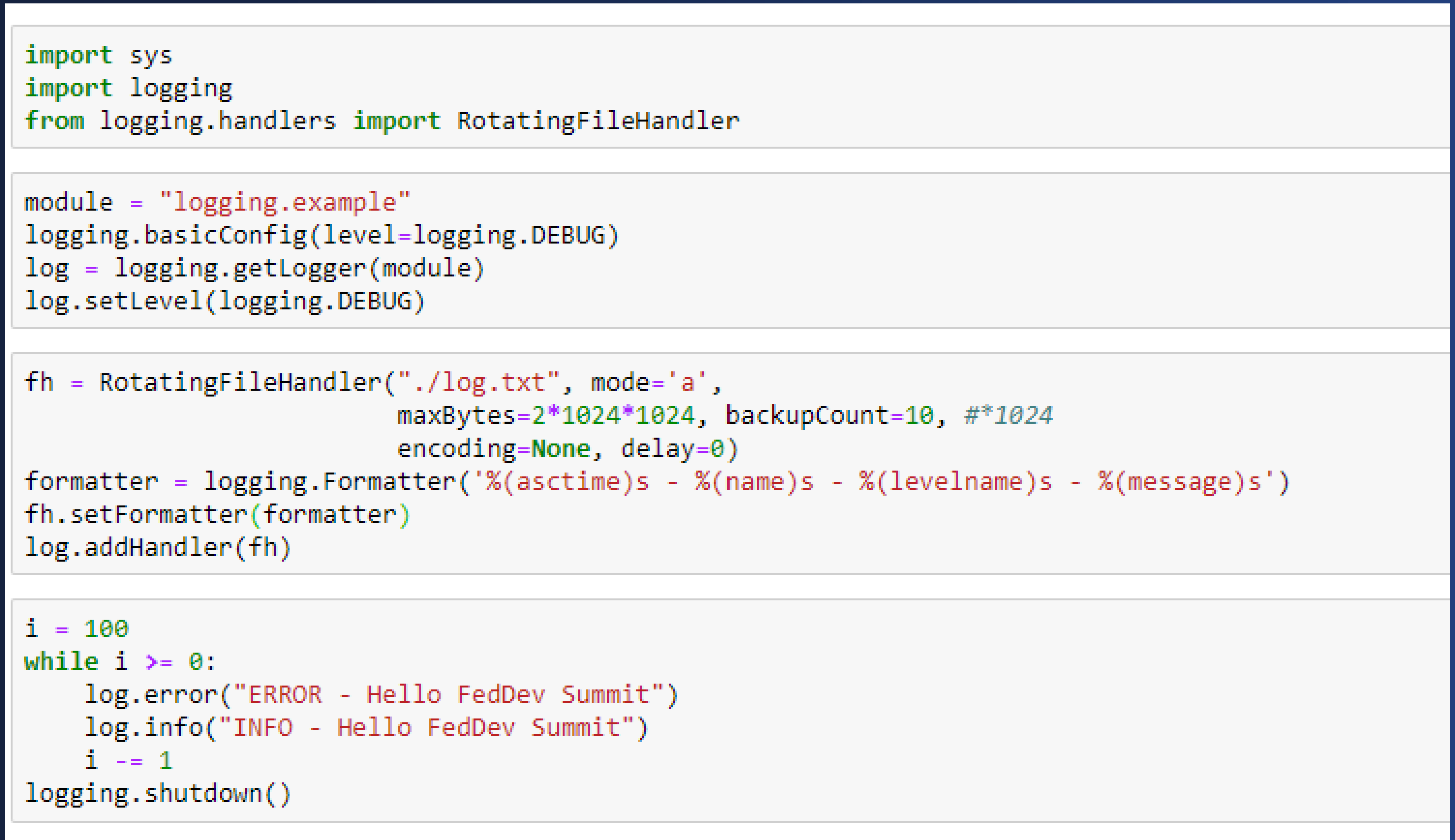
```
module = "logging.example"
logging.basicConfig(stream=sys.stderr, level=logging.DEBUG,
                    format='%(name)s (%(levelname)s): %(message)s')
log = logging.getLogger(module)
```

```
log.debug("DEBUG")
log.error("ERROR")
log.info("INFO")
```

```
logging.example (DEBUG): DEBUG
logging.example (ERROR): ERROR
logging.example (INFO): INFO
```

- Starts by importing logging
- Configure some basic information
  - Format
  - Default level
- Use getLogger() to grab the logger by name

Logging to a File

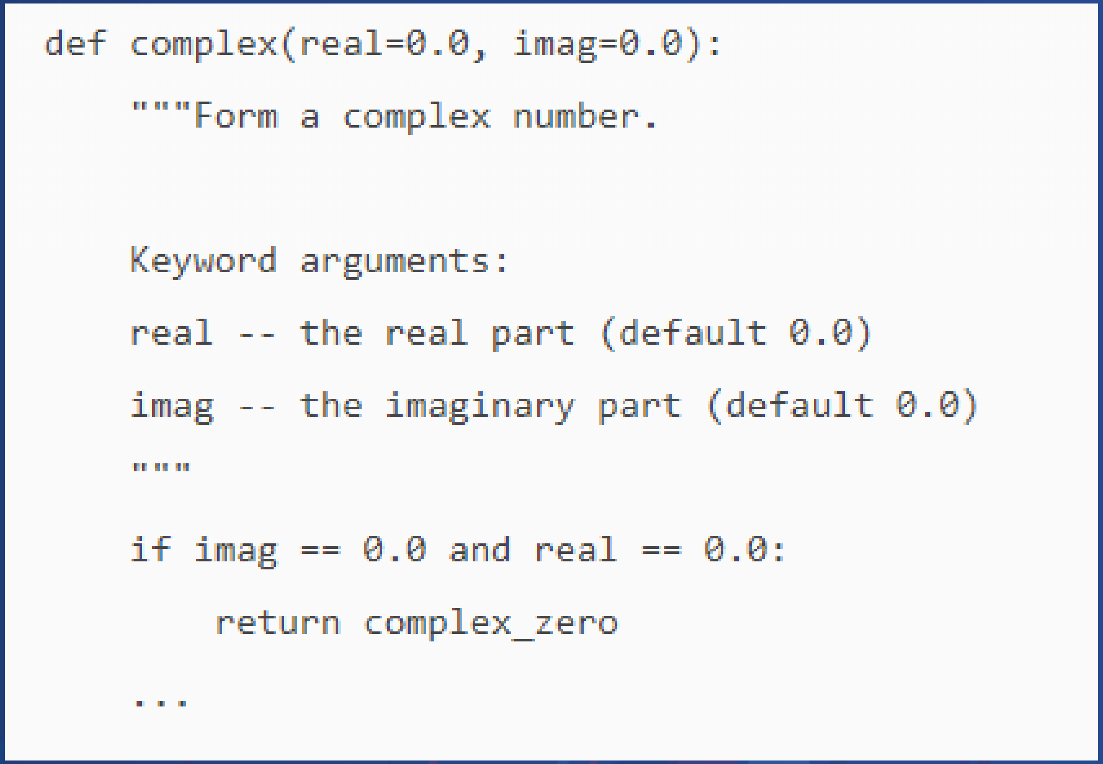


```python
import sys
import logging
from logging.handlers import RotatingFileHandler

module = "logging.example"
logging.basicConfig(level=logging.DEBUG)
log = logging.getLogger(module)
log.setLevel(logging.DEBUG)

fh = RotatingFileHandler("./log.txt", mode='a',
                         maxBytes=2*1024*1024, backupCount=10, #*1024
                         encoding=None, delay=0)
formatter = logging.Formatter('%(asctime)s - %(name)s - %(levelname)s - %(message)s')
fh.setFormatter(formatter)
log.addHandler(fh)

i = 100
while i >= 0:
    log.error("ERROR - Hello FedDev Summit")
    log.info("INFO - Hello FedDev Summit")
    i -= 1
logging.shutdown()
```

- Starts by importing logging
- Configure some basic information
  - Format
  - Default level
- Use getLogger() to grab the logger by name
- FileHandler objects handle the writing to disk

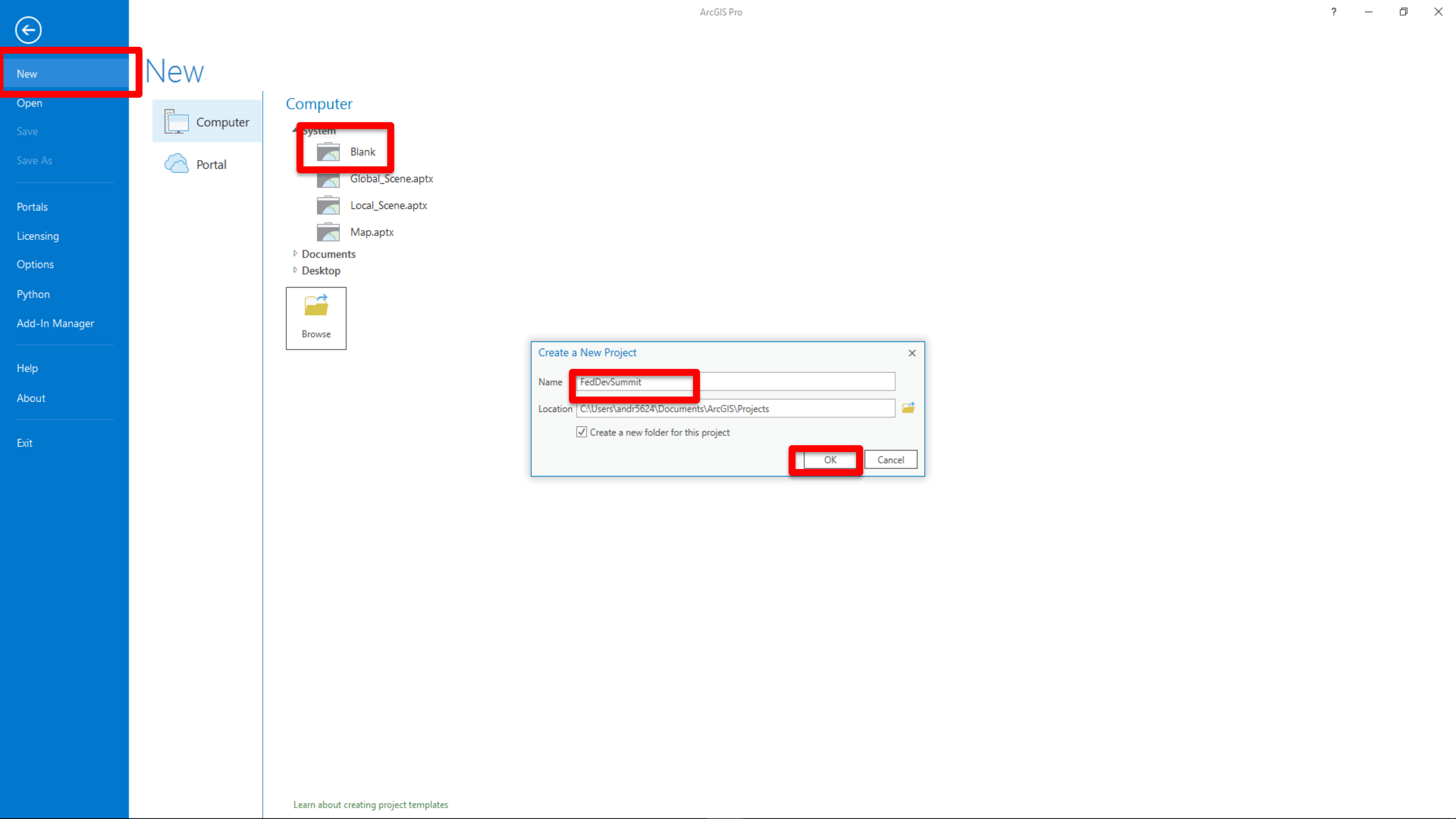
Documentation AKA Docstrings

- Every method, class and module should have them
- A docstring is a string literal that occurs as the first statement in a module, function, class, or method definition
- Reminds developers what your code does
- Can auto-build API documentation
  - Sphinx



```
def complex(real=0.0, imag=0.0):
    """Form a complex number.

    Keyword arguments:
    real -- the real part (default 0.0)
    imag -- the imaginary part (default 0.0)
    """
    if imag == 0.0 and real == 0.0:
        return complex_zero
    ...
```

Building Python Toolboxes

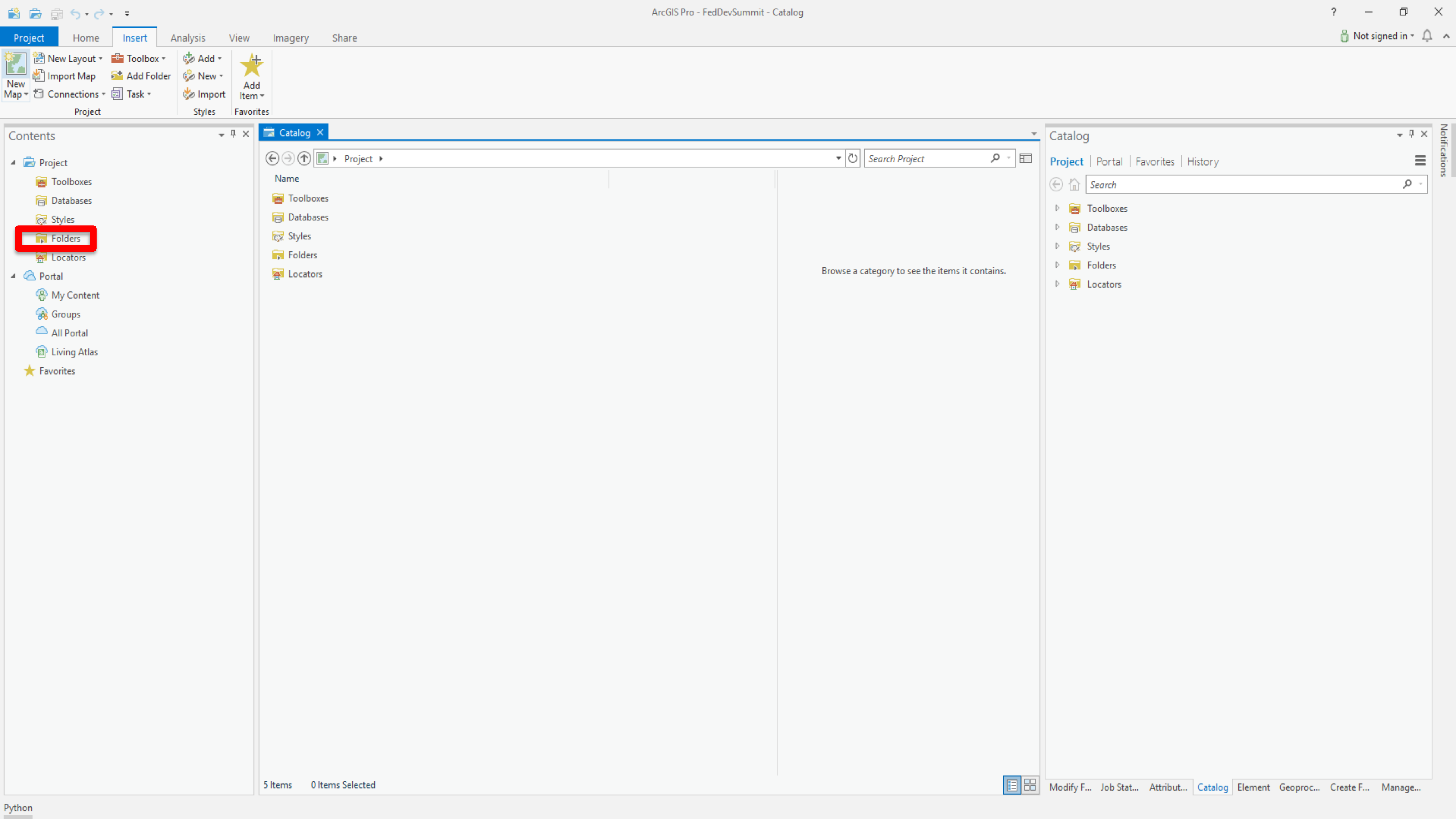


Create a New Project

Name: FedDevSummit

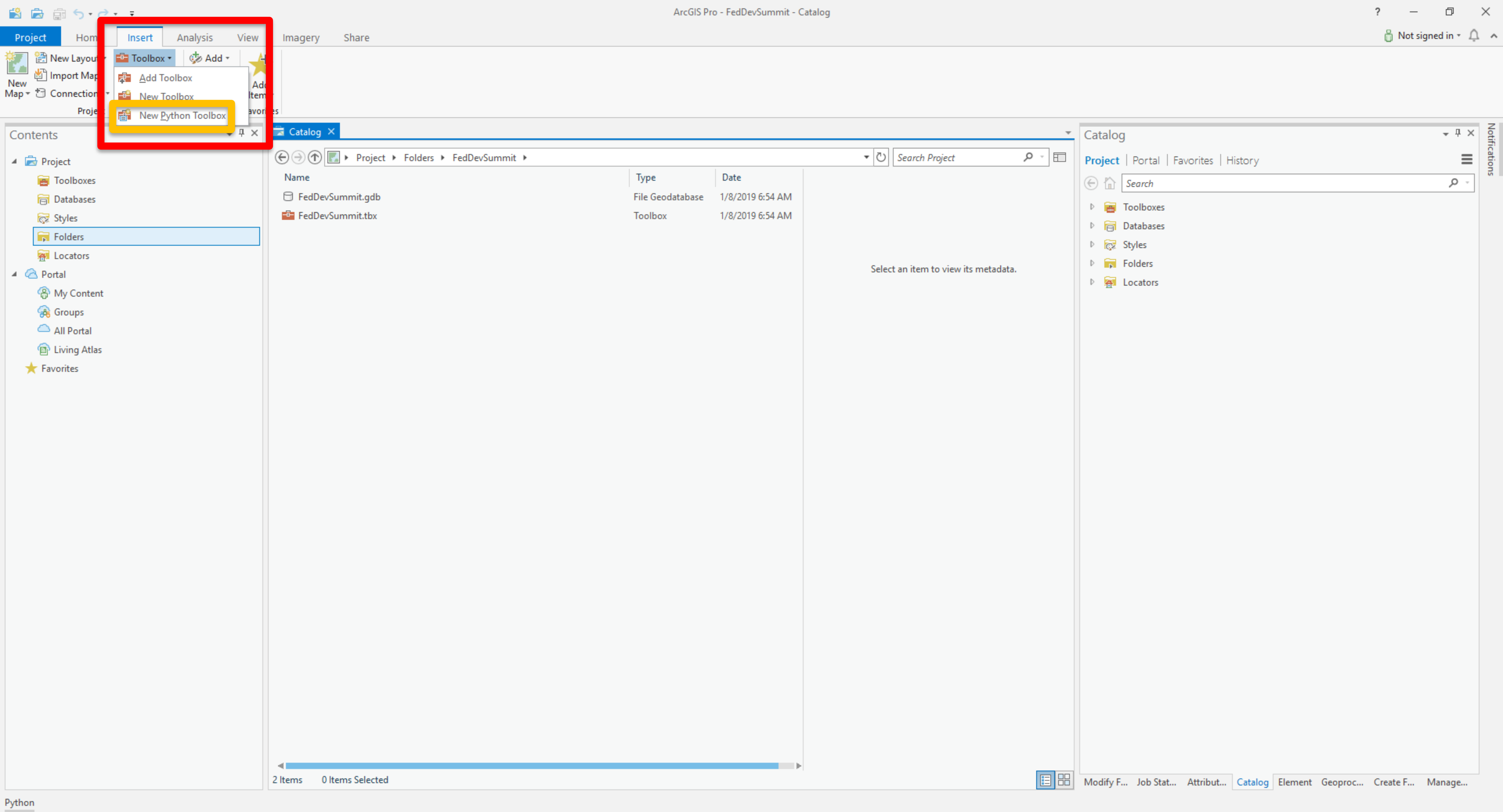
Location: C:\Users\andr5624\Documents\ArcGIS\Projects

☑ Create a new folder for this project



Browse a category to see the items it contains.

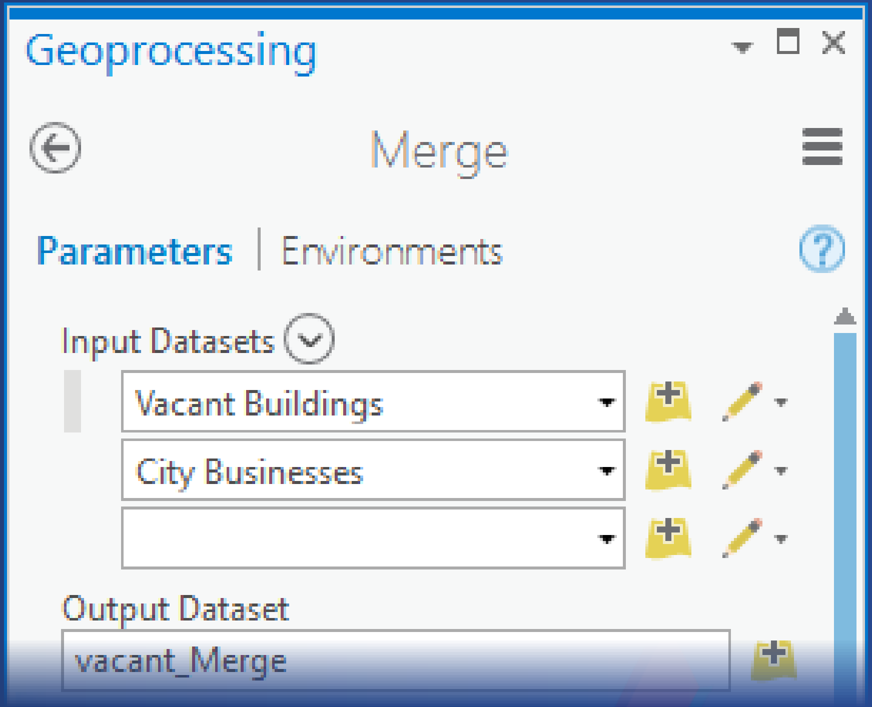
5 Items 0 Items Selected



Name	Type	Date
FedDevSummit.gdb	File Geodatabase	1/8/2019 6:54 AM
FedDevSummit.tbx	Toolbox	1/8/2019 6:54 AM

arcpy.Parameter

- ArcPy’s Parameter class defines inputs/outputs
- Requires the following:
  - Name
  - Display Name
  - Data type
  - Direction
  - Parameter Type (required, optional or derived)
- Import Link: http://pro.arcgis.com/en/pro-app/arcpy/geoprocessing_and_python/defining-parameter-data-types-in-a-python-toolbox.htm



Input Datasets

Vacant Buildings

City Businesses

Output Dataset

vacant_Merge

A Python Toolbox Framework

Building a solid tool

Questions?

Thank you
Andrew Chapkowski

Print Your Certificate of Attendance

Print Stations Located at L Street Bridge

Tuesday

12:30 pm – 6:30 pm
GIS Solutions Expo
Hall D

5:15 pm – 6:30 pm
GIS Solutions Expo Social
Hall D

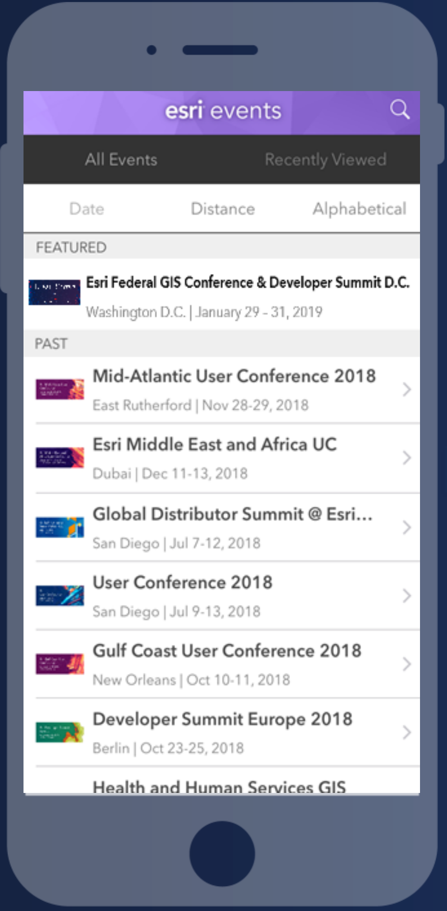
Wednesday

10:45 am – 5:15 pm
GIS Solutions Expo
Hall D

6:30 pm – 9:00 pm
Networking Reception
National Museum of Natural History

Please Take Our Survey on the App

Download the Esri Events app and find your event



esri events

All Events | Recently Viewed

Date | Distance | Alphabetical

FEATURED

Esri Federal GIS Conference & Developer Summit D.C.
Washington D.C. | January 29 - 31, 2019

PAST

Mid-Atlantic User Conference 2018
East Rutherford | Nov 28-29, 2018

Esri Middle East and Africa UC
Dubai | Dec 11-13, 2018

Global Distributor Summit @ Esri...
San Diego | Jul 7-12, 2018

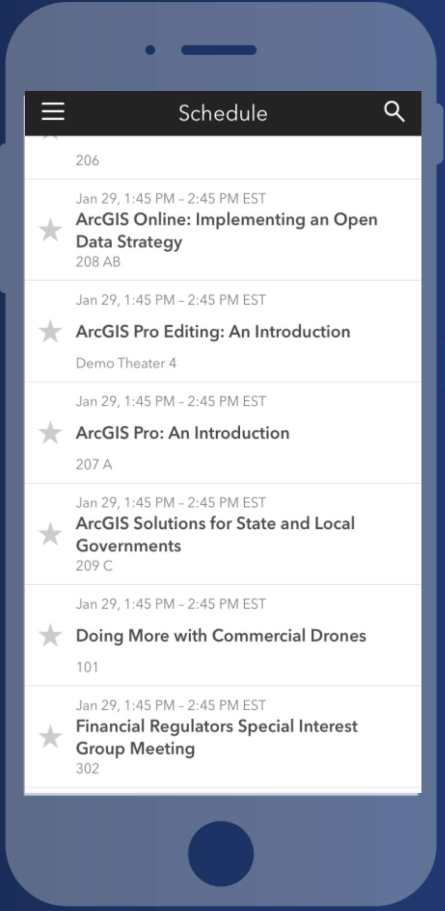
User Conference 2018
San Diego | Jul 9-13, 2018

Gulf Coast User Conference 2018
New Orleans | Oct 10-11, 2018

Developer Summit Europe 2018
Berlin | Oct 23-25, 2018

Health and Human Services GIS

Select the session you attended



Schedule

206

Jan 29, 1:45 PM - 2:45 PM EST
ArcGIS Online: Implementing an Open Data Strategy
208 AB

Jan 29, 1:45 PM - 2:45 PM EST
ArcGIS Pro Editing: An Introduction
Demo Theater 4

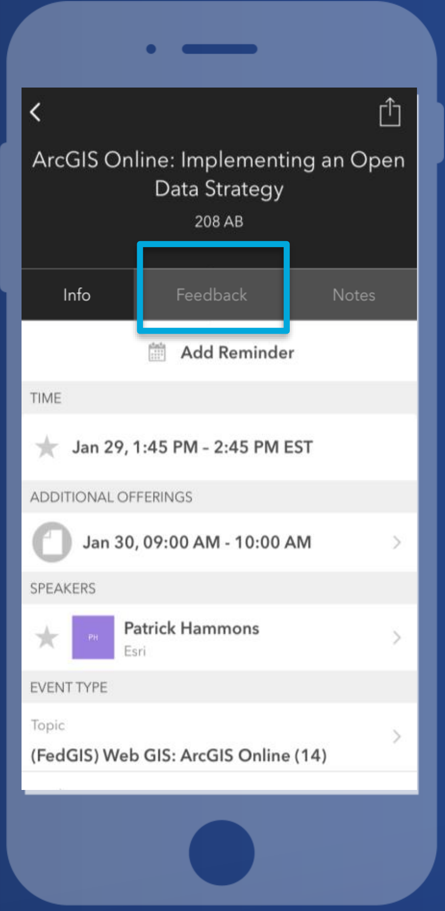
Jan 29, 1:45 PM - 2:45 PM EST
ArcGIS Pro: An Introduction
207 A

Jan 29, 1:45 PM - 2:45 PM EST
ArcGIS Solutions for State and Local Governments
209 C

Jan 29, 1:45 PM - 2:45 PM EST
Doing More with Commercial Drones
101

Jan 29, 1:45 PM - 2:45 PM EST
Financial Regulators Special Interest Group Meeting
302

Scroll down to find the feedback section



ArcGIS Online: Implementing an Open Data Strategy
208 AB

Info | Feedback | Notes

Add Reminder

TIME

Jan 29, 1:45 PM - 2:45 PM EST

ADDITIONAL OFFERINGS

Jan 30, 09:00 AM - 10:00 AM

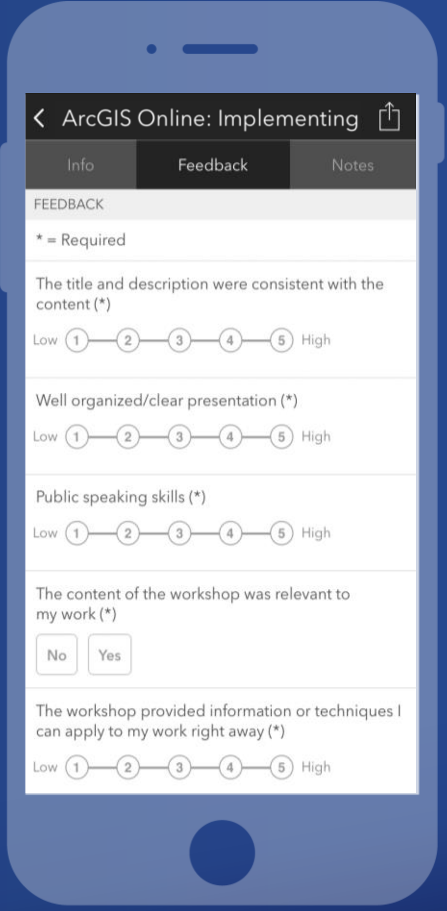
SPEAKERS

Patrick Hammons
Esri

EVENT TYPE

Topic
(FedGIS) Web GIS: ArcGIS Online (14)

Complete answers and select “Submit”



ArcGIS Online: Implementing

Info | Feedback | Notes

FEEDBACK

* = Required

The title and description were consistent with the content (*)
Low 1 2 3 4 5 High

Well organized/clear presentation (*)
Low 1 2 3 4 5 High

Public speaking skills (*)
Low 1 2 3 4 5 High

The content of the workshop was relevant to my work (*)
No | Yes

The workshop provided information or techniques I can apply to my work right away (*)
Low 1 2 3 4 5 High

esri

THE
SCIENCE
OF
WHERE