WORKING WITH PYTHON IN ARCGIS PRO

David Wynne
Working with Python in ArcGIS Pro
SDCC - Tech Theater 19 Exhibit Hall A

Come learn about recent developments in the Python experience within ArcGIS Pro. We will highlight various aspects of Python, arcpy, and other Python-related functionalities, focusing on the most recent additions.

Categories - - Technical Workshops, Performing Analysis

Focus mainly on changes with ArcGIS Pro 1.4 and 2.0

- Conda
- Charting
- arcpy.mp
- Describe
Conda – Why Packages?

- Software is composed of many smaller components, often called packages or libraries
- It’s often better to reuse code that solves a problem well rather than recreating it
- But, sharing code is a **hard problem**.
  - Do you have the same packages of the same versions as the developer did?
Package Management for Python

- Why not pip, wheels, virtualenvs?

- Don’t handle the harder problem of system dependencies, considered out of scope by Python packagers – does it end up in site-packages?
Why Conda?

• Scientific Python community identified that there was a gap not being addressed by the core Python infrastructure, limiting their ability to get packages into the hands of users

• Industry standard built by people who care about this space — Continuum Analytics
Why Conda?

- It solves the hard problem:
  - Handles dependencies for many languages
  - Built for Python first, but it really solves a much broader infrastructural issue.

- Gateway to data science — scientific, analytics, integrated software ecosystem for organizations
Conda
Demo
Charts

- In ArcGIS Pro, you can make several types of charts that visualize various characteristics and relationships in data

- The Chart class defines an ArcGIS Pro chart
  - Supports different types of charts, including bar charts, line charts, histograms, and scatter plots
  - Can use the class to define the chart title, axes, and other properties

1.4
- **Modify feature layer renderers**
  - Change renderer properties and modifying the symbol properties
  - SimpleRenderer, GraduatedColorsRenderer, GraduatedSymbolsRenderer, and UniqueValueRenderer
- **Add basemaps with the Map’s addBasemap method**
  - Also identify which layers are basemap layers using the Layer’s isBasemapLayer property
- **Add file-based and service layers using the Map’s addDataFromPath method**
  - It allows you to add a layer to a map using a path or URL

2.0
- **Legend synchronization options with its map layers**
- **Support for the RasterClassifyColorizer and RasterUniqueValueColorizer.**
Describing data

• `arcpy.Describe` is a critical part of many scripts
  - Provide info about data that can be used to control the flow of a Python script

• But the object can be *cumbersome* to work with
  - It’s dynamic, but provides no drop-downs
  - Have to “know” or keep running to the documentation

• `arcpy.da.Describe` provides all the same information but returns it as a dictionary
arcpy.da.Describe

Demo
Please Take Our Survey on the **Esri Events App**!

Download the Esri Events app and find your event

Select the session you attended

Scroll down to find the survey

Complete Answers and Select “Submit”