Book Proposal

**Top 20 Python**

**Vision and Audience:** Acquire fundamental Python competencies within the context of ArcGIS. Readers will embark on their Python development journey while optimizing its integration with ArcGIS capabilities. As each Python skill is introduced, we’ll apply it to various aspects of GIS, sometimes with increasing complexity. We’ll also discuss how to approach best designing their GIS Python solutions.

Each chapter will include a project for the user to learn by example. All projects will be themed as if they were assigned as part of their role at the fictional company GeoNinjas Pythonalytics. These assignments will work towards GeoNinjas Pythonalytics' gaining an understanding of their user base, performing market analytics, and managing their Portal and ArcGIS Online organization. In addition to their analysis, users will deliver their work in various ways, including Python scripts, Jupyter Notebooks, ArcGIS Pro toolbox tools, geoprocessing services, and web applications. While each project builds off our work in preceding chapters, each project will be designed to be completed independently, not requiring the reader to have completed previous chapters.

**Authors: Daniel Yaw, GISP**

**Notes:**

**Competitive/comparable titles:**

***Price:***

***Specs:***

***Proposed time frame and due date:***

Proposed Outline or TOC:

**Preface**  **Getting Started**

We’ll give the user a basic understanding of how this book is laid out and get them off to a quick start setting up VS Code.

Concepts Introduction

* How this book is laid out
* Essential tools for this book, including ArcGIS Pro and VS Code
* Brief introduction to the Zen of Python and our approach to code
* Briefly discuss tools to aid development, such as auto linters and auto formatters

Practice Setup VS Code and Create Your Workspace – Set up VS code and create a new Conda environment based on the default ArcGIS Pro Conda env

**Chapter 1**  **Designing Your Python Solution**

Before we jump into the code, we’ll discuss how to understand best your task / project’s needs and the tools at our disposal.

Concepts Understanding What We Need

* Explanation of Esri software referenced in this book
  + ArcGIS Pro
  + ArcGIS Online
  + ArcGIS Enterprise
* What do we need?
* Who will be using it?
* How do we deliver it?

Practice First project -

You’re tasked with developing a script that takes a list of user addresses and creates an ArcGIS Pro project with a map of the user addresses and a PDF of the map. Using some questions and answers from the script's audience, we understand our options and develop an approach that works for our organization. This tool will be created in the following chapter.

**Chapter 2**  Data Structures

Loops, data types, spatial data, etc

Concepts

Practice

**Chapter 3**  ArcPy Basics

Get started with programming using ArcPy.

Concepts Intro to ArcPy

* What is ArcPy?
* Capabilities / Modules
* Creating Projects
* Adding Maps, Layers, etc
* Symbolizing Data

Practice Make a Python script that uses ArcPy - Create a Python script that accomplishes the task discussed in Chapter 1.

**Chapter 4**  ArcGIS Python API Basics

Get started with programming using the ArcGIS Python API.

Concepts Intro to the ArcGIS Python API

* What is the ArcGIS Python API?
  + Capabilities / Modules
* ArcGIS Online and ArcGIS Enterprise Content
* Discovering content
* Creating a web map

Practice Make a Python Script that uses the ArcGIS Python API. Using a sample dataset, which is an output of the script from Chapter 2, publish it to our ArcGIS Online organization as a web map, and share it with our users.

**Chapter 5**  Jupyter Notebooks

Get started with Jupyter Notebooks. We’ll cover examples using Jupyter Notebook, but also introduce other places Jupyter Notebooks can live, such as ArcGIS Pro, VS Code, ArcGIS Online.

Concepts Intro to Jupyter Notebooks

* What’s a Jupyter Notebook?
  + Notebooks in ArcGIS Pro
  + Notebooks in IDEs (VS Code)
  + ArcGIS Notebook Server and ArcGIS Online Notebooks
* Jupyter Notebook Basics
  + iPython and Magic Commands
  + Customizing Output (maps, graphs, etc)

Practice Using the hosted feature service from Chapter 2, create a professional-looking Jupyter Notebook that connects to the data, queries it, and displays it on a map.

**Chapter 6**  Data Manipulation

Manipulate your data to get exactly what you need.

Concepts Retrieving and Handling Data

* Pandas
* Spatially Enabled DataFrames
* Manipulating the Data

Practice Load and manipulate data using Pandas. Export the data from the hosted feature service created in Chapter 2 into a spatially enabled data frame and perform some basic queries and analysis on the data. Load data showing traffic to our website and aggregate it to show important metrics, such as requests per user per day.

**Chapter 7**  Data Engineering

Ingest data from sources into your database.

Concepts Building Your Pipeline

* What’s ETL?
* Ingesting Data from an API
* Scraping
* Loading Data

Practice Building a Python script to extract sensor data from our locations, transform it to match our schema and quality standards, and load it into a file geodatabase.

**Chapter 8**  Parallel Processing

Concepts

Practice

**Chapter 9**  Data Visualization

Create impressive visualizations to share your data and analysis.

Concepts Non-Esri Visualization Tools

* Matplotlib and basic plotting
* Seaborn
* Pandas visualization
* Plotly
* ArcPy and ArcGIS Python API for data visualization

Practice Create visualizations showing our users and trends. Utilize Pandas, Plotly, and the ArcGIS Python API to create a series of visualizations to understand our user base.

**Chapter 10**  Geoprocessing

Perform complex geoprocessing tasks with relative ease using ArcPy

Concepts Data Management

* Exporting data
* Converting data
* Data Analysis
* Overlay toolset
* Extract toolset
* Statistics toolset

Practice Create a script to enrich and analyze data and export the results to share with our stakeholders.

**Chapter 11**  Geoprocessing Toolboxes and Services

Create geoprocessing tools to perform everyday workflows

Concepts Geoprocessing Toolboxes

* What’s a Toolbox?
  + Getting started by using Model Builder
  + Creating a Toolbox
  + Sharing Toolboxes
* Geoprocessing Services
  + What’s a geoprocessing service?
  + Publishing geoprocessing services
  + Consuming geoprocessing services

Practice Turn the geoprocessing script from Chapter 11 into a geoprocessing toolbox and then publish it as a geoprocessing service.

**Chapter 12**  Object-Oriented Programming

Use object-oriented programming (OOP) to create modular, organized, and maintainable code for your GIS Python projects.

Concepts OOP Concepts

* What is OOP?
* Why use OOP in Python?
* OOP Components

Practice Using object-oriented programming, create a script that will take an inventory of our content on our ArcGIS Online organization.

**Chapter 13**  Interacting with Databases

Learn how to interact with databases and perform everyday tasks.

Concepts Working with SQL Databases

* SQLAlchemy
* ORM
* Working with Enterprise Geodatabases
  + Using ArcPy to interact with eGDBs
  + Common maintenance tasks

Practice Design a schema and create a file geodatabase and spatially-enabled SQLite database.

**Chapter 14**  ArcGIS Enterprise and Online

Programmatically work with your ArcGIS Online or ArcGIS Enterprise Organization

Concepts

* Working with Data
  + Loading and extracting feature data
  + Geometry
  + Geoenrichment
  + Geocoding
* Administrative Tasks
  + Roles, Users, Groups
  + Moving Content

Practice Geocode customer data using ArcGIS Online, and enrich it with valuable data

**Chapter 15**  Working with Web APIs

Web APIs can interact with and obtain data from many systems, including ArcGIS Enterprise and ArcGIS Online.

Concepts Intro to APIs

* What’s a REST API?
* Understanding HTTP
* Request and response types

Practice Create a script to use the ArcGIS REST API to search for all content created in the past 30 days and summarize the results.

**Chapter 16**  Web Frameworks

Take your tools to the next level by integrating them with web frameworks

Concepts Intro to Web Frameworks

* What’s a web framework
* Introduction to REST
* Flask
* Talk about other web frameworks
  + Django
  + FastAPI

Exercise 1 Create a Flask app that uses the script from Chapter 14, which allows users to create an inventory of content from our ArcGIS Online organization

Exercise 2 Build further upon our Flask app to enable it to display the data collected in a user-friendly format.

**Chapter 17**  Source Control

Using git and publishing to GitHub

Concepts Git Repos

* Intro to Git
* How Git works
* Common tasks
* GitHub
* Creating a GitHub repo
* Managing GitHub repo

Exercise 1 Create a GitHub repo to share the package from Chapter 16.

Exercise 2 Create a new branch, push data, and merge the branch with main

**Chapter 18** Packaging

Create a Python Package for the tools you develop

Concepts Intro to Packages

* Project structure
* Creating packages with PIP
* Creating packages with Conda

Exercise 1 Organize the code from 16 into a Python package structure

Exercise 2 Turn this code into a Python package that allows users to use and integrate it with other processes easily.

**Chapter 19**  Debugging and Troubleshooting

Debugging is a critical part of development. In this section, we’ll cover some debugging and troubleshooting techniques.

Concepts Working with Your Debugger

* What’s a debugger?
* Setting up your debugger
* Using your debugger

Exercise 1 After some recent updates, the package from Chapter 18 is no longer working as expected, and we need to debug this code.

Exercise 2 We need more information to understand what is happening, analyze network traffic to better understand the problem and how to fix it.

**Chapter 20**  Documentation

Documentation is critical to the success of yourself and your users. We’ll look at how to create and publish documentation effectively

Concepts Documentation Best Practices

* Concepts
* Docstrings
* Creating Documentation
* Static sites - Sphinx and Jupyter Book
* PDF generation

Exercise 1 Create docstrings to document your code

Exercise 2 Use Jupyter Book to automatically create interactive, user-friendly documentation