Creating Python® Scripts for ArcGIS®

STUDENT EDITION

Copyright © 2020 Esri
All rights reserved.

Course version 5.2. Version release date December 2020.

Printed in the United States of America.

The information contained in this document is the exclusive property of Esri. This work is protected under United States copyright law and other international copyright treaties and conventions. No part of this work may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, or by any information storage or retrieval system, except as expressly permitted in writing by Esri. All requests should be sent to Attention: Director, Contracts and Legal, Esri, 380 New York Street, Redlands, CA 92373-8100, USA.

Export Notice: Use of these Materials is subject to U.S. export control laws and regulations including the U.S. Department of Commerce Export Administration Regulations (EAR). Diversion of these Materials contrary to U.S. law is prohibited.

The information contained in this document is subject to change without notice.

Commercial Training Course Agreement Terms: The Training Course and any software, documentation, course materials or data delivered with the Training Course is subject to the terms of the Master Agreement for Products and Services, which is available at https://www.esri.com/~/media/Files/Pdfs/legal/pdfs/ma-full/ma-full.pdf. The license rights in the Master Agreement strictly govern Licensee's use, reproduction, or disclosure of the software, documentation, course materials and data. Training Course students may use the course materials for their personal use and may not copy or redistribute for any purpose. Contractor/Manufacturer is Esri, 380 New York Street, Redlands, CA 92373-8100, USA.

Esri Trademarks: Esri trademarks and product names mentioned herein are subject to the terms of use found at the following website: https://www.esri.com/legal/copyright-trademarks.html.

Other companies and products or services mentioned herein may be trademarks, service marks or registered marks of their respective mark owners.

Table of Contents

Esri resources for your organization

Course introduction

Course introduction
Course goals
Installing the course data
Icons used in this workbook
Understanding the ArcGIS platform

1 Python automation for your organization

Lesson introduction
Benefits of Python automation
The Python script creation workflow
Integrating Python in ArcGIS Pro
Python development and run environments
Accessing Python
Adding Python packages to ArcGIS Pro
Lesson review
Answers to Lesson 1 questions

2 Using Python to access geoprocessing tools

Lesson introduction
Viewing parameters in a geoprocessing tool
Explore the syntax of a geoprocessing tool
Ways to use variables in a script
Using variables in a script
Exercise 2A: Create a simple script with variables
- Sign in to ArcGIS Pro
- Create a new ArcGIS Pro project
- Add in data for the affected area
- Evaluate the syntax of the Copy Features tool
- Evaluate the syntax of the Kernel Density tool
- Combine the geoprocessing tools in a script

Python troubleshooting techniques
Troubleshooting errors in a script
Exercise 2B: Add troubleshooting techniques to a Python script
- Set up PyCharm
- Add pseudocode
- Add print functions

Comment existing code
Create a variable
Use debugging tools in PyCharm
Lesson review
Answers to Lesson 2 questions

3 Using Describe properties in geoprocessing tools

Lesson introduction
Accessing properties with a Describe function
Identify available Describe function properties
Describe syntax and usage
Accessing Describe properties
Exercise 3: Use the Describe function in a geoprocessing script
Create a new map
Examine the properties of a feature class
Use the Describe function to examine properties
Create a new Python script in PyCharm
Create a Describe object
Use Describe properties to create a new feature class
Verify the new feature class
Using the da.Describe object in a geoprocessing script
Lesson review
Answers to Lesson 3 questions

4 Automating Python scripts with lists

Lesson introduction
List functions in Python
Examine the ListFiles function
Finishing the List function syntax
Exercise 4A: Prepare to automate with Python lists
Create a new map
Create a list of workspaces
Create a list of features
Create a list of fields
Using a for loop
Using a for loop to iterate over lists
Exercise 4B: Automate a geoprocessing workflow using loops
Create a new Python script in PyCharm
Copy feature classes from a workspace
Verify the new feature classes
Lesson review

Answers to Lesson 4 questions

5 Working with cursors

Lesson introduction
Types of cursors
Examine cursors
Determine which cursor to use
Exploring cursor syntax
Using cursors in a workflow
Apply the cursor workflow
Exercise 5: Read and update values in a feature class
- Create a new Python script in PyCharm
- Set up the Python script
- Create an Update cursor
- Create a Search cursor
- Write values to a CSV file
- Run the script in PyCharm
- Create a new map
- Verify the script results

Lesson review
Answers to Lesson 5 questions

6 Geoprocessing with geometry objects

Lesson introduction
Benefits of geometry objects
Workflows to create geometry objects
Discovering polyline geometry objects
Components of a geometry object
Discover geometry object methods
Exercise 6: Convert coordinates into affected area polygons
- Open a Python script in PyCharm
- Evaluate the Python script
- Create the geometry object from a list
- Run the script in PyCharm
- Create a new map
- Verify the script output feature classes

Lesson review
Answers to Lesson 6 questions

7 Error management techniques

Lesson introduction

Types of errors that occur in geoprocessing
Using try-except statements
Error handling techniques
Introducing error handling in Python scripts
Determining the error handling technique
Exercise 7: Apply error handling techniques
    Open the script in PyCharm
    Run the Python script with empty input
    Add a custom Raise exception
    Run a Python script with a Raise exception
    Run a Python script with invalid input
    Add an arcpy.ExecuteError exception
    Run the Python script with all error handling
Lesson review
Answers to Lesson 7 questions

8 Creating a Python script tool

Lesson introduction
Improving script accessibility
Components of a Python script tool
Accepting user input
Parameter identification
Exercise 8: Create a Python script tool
    Prepare the Python script for user input
    Create a new map
    Create a script tool in ArcGIS Pro
    Run the Python script tool
Lesson review
Answers to Lesson 8 questions

9 Adding validation to script tools

Lesson introduction
Customizing script tool behavior
Using validation to create customizations
Validate script tool inputs using ToolValidator methods
Exercise 9: Add custom messaging to a script tool
    Start ArcGIS Pro
    Open the script tool validation properties
    Add code to the validator script
    Verify the validator code
    Verify the updated tool messaging
Lesson review

Answers to Lesson 9 questions

10 Using Python script tools in the platform

Lesson introduction
Methods to share a Python script tool
Determining the appropriate method
Workflow to share a Python script tool
Exercise 10: Share a geoprocessing package
    Start ArcGIS Pro
    Analyze the geoprocessing package
    Modify the metadata
    Share the geoprocessing package
Workflow review
Sharing a web tool within the ArcGIS platform
Answers to Lesson 10 questions

Appendices

Appendix A: Esri data license agreement
Appendix B: Answers to lesson review questions
Appendix C: Additional resources