Creating Python® Scripts for ArcGIS®

STUDENT EDITION

Copyright © 2018–2026 Esri

All rights reserved.

Course version 10.0. Version release date January 2026.

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 found at the following website: esri.com/TrainingTerms. 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 Marks: Esri marks and product names mentioned herein are subject to the terms of use found at the following website: esri.com/EsriMarks.

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
Training Services account credentials
Icons used in this workbook

1 Introduction to Python and ArcGIS

Lesson introduction
Benefits of Python automation
ArcGIS Python libraries
Python script creation workflow
Python development and run environments in ArcGIS Pro
Introduction to ArcGIS Notebooks for Python
Exercise 1: Explore Python code in ArcGIS Notebooks
- Open an ArcGIS Notebooks notebook
- Explore the ArcGIS Notebooks interface
- Examine markdown cell content
- Run code cells
- Challenge: Format markdown text in a notebook
- Challenge: Write code in ArcGIS Notebooks
- Challenge: Export a notebook to a .py file

Lesson review
Answers to lesson 1 questions

2 Using Python to access geoprocessing tools

Lesson introduction
Using Python to automate a geoprocessing workflow
Geoprocessing tool syntax
ArcGIS resources for Python scripting

Exercise 2A: Use Python resources to discover geoprocessing tool syntax
Review the geoprocessing workflow
Send a geoprocessing tool's history to the Python window
Open a geoprocessing tool's documentation using ArcGIS Pro
Locate a geoprocessing tool's documentation in a web browser
Challenge: Locate nearby features using Python code
Challenge: Create a feature class using Python code
Challenge: Reproject a layer into another coordinate system using Python code
Variable use in geoprocessing scripts
Exercise 2B: Create a geoprocessing script using Python variables
Add data to a map
Evaluate the syntax of geoprocessing tools
Call a geoprocessing tool using variables
Combine the geoprocessing tools in a script using variables
Challenge: Change a geoprocessing tool call to use named or positional parameters
Challenge: Change a geoprocessing tool's parameters
Challenge: Add another tool to the Python script
Lesson review
Answers to lesson 2 questions

3 Accessing data properties in Python

Lesson introduction
Accessing data properties with ArcPy
ArcPy Describe and Data Access Describe functions
Accessing property values from a dictionary
Identify available Data Access Describe properties
Data Access Describe (da.Describe) syntax
Accessing da.Describe values
Exercise 3: Use the da.Describe function in a geoprocessing script
Create a map
Examine the properties of a feature class
Use the da.Describe function to examine properties
Create a notebook in ArcGIS Pro
Create a da.Describe object
Use da.Describe properties to create a feature class

Verify the new feature class
Challenge: Examine the Describe object syntax for a data type
Challenge: Call properties from the Describe object
Challenge: Adapt the script's syntax to use the Describe object syntax
Lesson review
Answers to lesson 3 questions

4 Automating Python scripts with lists

Lesson introduction
List functions in Python
Examine a list function
Finishing the list function syntax
Exercise 4A: Prepare to automate with Python lists
Create a map
Create a list of workspaces
Create a list of features
Create a list of fields
Challenge: Refine a list function
Challenge: Refine a list function with a wildcard
Challenge: Refine a list function to show strings
Using a for loop
Using a for loop to iterate over lists
Exercise 4B: Automate a geoprocessing workflow with loops
Create a notebook in ArcGIS Pro
Create a list function on the data
Create a loop on the list function result
Run a geoprocessing tool on each file in the list
Challenge: Run the Buffer tool on each file in a list
Challenge: Project files with different spatial references conditionally
Lesson review
Answers to lesson 4 questions

5 Working with cursors

Lesson introduction
ArcPy cursor use

Types of cursors
Determine which cursor to use
Exploring cursor syntax
Examine a cursor's syntax using ArcGIS resources
Exercise 5: Prepare tabular data using cursors
Inspect feature data in a map
Review data using a search cursor
Delete outdated rows
Update incorrect data values
Calculate column values
Insert features into a feature class
Challenge: Fix a typo in many rows
Challenge: Fix incorrect values in a subset of rows
Challenge: Sort a search cursor's contents
Challenge: Write selected values from a feature class to a CSV file
Lesson review
Answers to lesson 5 questions

6 Geoprocessing with geometry objects

Lesson introduction
Introduction to ArcPy geometry objects
Geometry object creation workflows
Examine access to existing geometry properties
Geometry object syntax
Using geometry objects in a data processing workflow
Examine geometry object methods
Exercise 6: Use geometry objects as inputs in a geoprocessing tool
Prepare a notebook
Read a CSV file of coordinates into a list
Create a Multipoint geometry object
Use a method on a geometry object
Add geometry object analysis results to a feature class
Review analysis results
Challenge: Use a spatial join with a geometry object
Challenge: Return impacted service areas using a spatial filter

Challenge: Write results to a CSV file
Lesson review
Answers to lesson 6 questions

7 Preparing ArcGIS notebooks for sharing

Lesson introduction
Benefits of ArcGIS Notebooks for sharing code
Preparing cells for sharing
Examine notebook elements for sharing success
Exercise 7: Prepare an ArcGIS notebook for sharing
Adjust notebook code cells
Add descriptive text to a notebook
Add links to a notebook
Add an image to a notebook
Challenge: Open the notebook in Jupyter Notebooks
Challenge: Embed a web page into a notebook
Challenge: Add an iPy widget to a notebook
Methods for sharing an ArcGIS notebook with other users
Lesson review
Answers to lesson 7 questions

8 Creating a Python script tool

Lesson introduction
Introduction to Python script tools
Improving script accessibility
Properties of a Python script tool
Accepting user input
Exercise 8: Create a Python script tool
Prepare the Python script for user input
Create a script tool in ArcGIS Pro
Set script tool parameters in ArcGIS Pro
Run the Python script tool
Challenge: Adjust a script tool parameter's properties
Challenge: Add a message to a Python script tool's geoprocessing history
Challenge: Implement a GetParameter() function in a script tool

Challenge: Control a Python script tool's environment settings
Lesson review

9 Adding validation to script tools

Lesson introduction
Customizing script tool behavior
ToolValidator structure
Choose a property to implement a behavior
Exercise 9A: Examine ToolValidator methods for implementing custom behavior
Locate the ToolValidator documentation
Examine sample code for setting a custom message
Examine sample code for enabling and disabling a parameter
Examine sample code for setting a default value for a parameter
Challenge: Examine how to set parameter dependencies
Challenge: Review an existing Python script tool's validation script
Use validation to create customizations
Exercise 9B: Add custom behavior to a Python script tool
Open the script tool validation properties
Add code to the validator script
Verify the updated tool messaging
Challenge: Create validation code to set a default value
Challenge: Improve script tool responsiveness by editing validation code
Challenge: Use a parameter's name to call a parameter object
Challenge: Add a filter to the Python script tool
Lesson review
Answers to lesson 9 questions

10 Working with Python environments in ArcGIS Pro

Lesson introduction
Libraries in Python
ArcGIS Pro and Python environments
Identifying the Python libraries used to create a script
Exercise 10: Use ArcGIS Pro to examine Python environments
Access the ArcGIS Pro package manager
View the default environment

Activate a clone environment
Review available packages in the environment
Challenge: Locate the clone environment in File Explorer
Challenge: Install a package to the clone environment
Challenge: Clone an ArcGIS Pro Python environment
Python environment sharing considerations
Lesson review
Answers to lesson 10 questions

11 Using Python script tools in ArcGIS

Lesson introduction
Methods for sharing a Python script tool
Determining the appropriate sharing method
Sharing considerations for Python script tools
Workflow for sharing a Python script tool
Exercise 11: Share a Python script tool as a geoprocessing package
Run the Python script tool
Analyze the geoprocessing package
Modify the metadata
Share the geoprocessing package
Challenge: Import the geoprocessing package
Challenge: Review ArcGIS Enterprise web tool sharing resources
Workflow review
Lesson review
Answers to lesson 11 questions

Appendices

Appendix A: Data license agreement
Appendix B: Detailed descriptions and examples of ToolValidator class components
Appendix C: Answers to lesson review questions
Appendix D: Additional resources
Appendix E: Lesson stories