

# C O N T E N T S

<b>1</b>	<i>Introduction</i>	
	Welcome to Introduction to Geoprocessing Scripts Using Python	1-1
	Course objectives	1-2
	Day 1 course content	1-3
	Day 2 course content	1-4
	Course materials	1-5
	Additional resources	1-6
	ESRI Geoprocessing Resource Center	1-7
	ESRI Developer Network	1-8
	ESRI Support Center	1-9
	ArcGIS: A complete GIS	1-10
	Logistics	1-11
	Introductions	1-12
	Exercise goals	1-13
<b>2</b>	<i>The geoprocessing framework</i>	
	The geoprocessing framework	2-1
	Learning objectives	2-2
	What is geoprocessing?	2-3
	Accessing geoprocessing tools	2-4
	Inside ArcToolbox	2-5
	Supported data types	2-6
	System toolboxes	2-7
	Tools with the same name	2-8
	Tools and licensing	2-9
	The geoprocessing framework	2-10
	Class focus	2-11
	Why write scripts?	2-12
	Points of interest	2-13
	The Python scripting language	2-14
	Exercise goals (SBS)	2-15
	Lesson review	2-16
<b>3</b>	<i>The basics of Python</i>	
	The basics of Python	3-1
	Learning objectives	3-2
	Where to write code	3-3
	PythonWin interface	3-4
	IDLE interface	3-5
	Comments	3-6

Variables in Python	3-7
Strings	3-8
Numbers and lists	3-9
Line continuation	3-10
Built-in functions	3-11
Accessing modules	3-12
Statements	3-13
Decision making syntax	3-14
Looping syntax	3-15
Case sensitivity rules	3-16
Common shortcut keys for PythonWin	3-17
Python resources	3-18
Exercise goals (SBS)	3-19
Lesson review	3-20
<b>4</b> <i>Accessing tools and environment settings in scripts</i>	
Accessing tools and environment settings in scripts	4-1
Learning objectives	4-2
Common ArcObjects	4-3
Interacting with ArcObjects	4-4
The GeoProcessor ArcObject	4-5
Creating the geoprocessor	4-6
Syntax for properties and methods	4-7
Demonstration: Finding the scripting syntax	4-8
Toolbox aliases	4-9
The Select tool	4-10
The CreateFeatureClass tool	4-11
The (raster) Clip tool	4-12
The Buffer tool	4-13
The Union tool	4-14
Learning how to populate tool arguments	4-15
Environment settings	4-16
Running multiple tools	4-17
Exercise goals	4-18
Lesson review	4-19
<b>5</b> <i>The Geoprocessor Programming Model and geoprocessor object</i>	
The Geoprocessor Programming Model and geoprocessor object	5-1
Learning objectives	5-2
What you know	5-3
What you may not know	5-4
The Geoprocessor Programming Model	5-5
Finding the model	5-6
Object symbols	5-7

Property and method symbols	5-8
Object colors	5-9
Methods and properties on the geoprocessor	5-10
Adding and removing toolboxes	5-11
Checking whether data exists	5-12
Overwriting the output	5-13
Getting help	5-14
Exercise goals	5-15
Lesson review	5-16
<b>6</b> <i>The describe objects</i>	
The describe objects	6-1
Learning objectives	6-2
Returning standard data	6-3
Returning objects	6-4
The describe objects	6-5
The Describe method	6-6
The describe objects	6-7
Dataset properties	6-8
Feature class properties	6-9
Raster dataset properties	6-10
Describe object properties	6-11
Demonstration: Help for describe objects	6-12
Exercise goals	6-13
Lesson review	6-14
<b>7</b> <i>The list objects</i>	
The list objects	7-1
Learning objectives	7-2
The list objects	7-3
The list methods	7-4
The list objects	7-5
The list objects	7-6
List feature class example	7-7
List dataset example	7-8
Looping through a list of data	7-9
The Field object	7-10
Activity: Complete the code	7-11
Exercise goals	7-12
Lesson review	7-13
<b>8</b> <i>The cursor objects</i>	
The cursor objects	8-1

Learning objectives	8-2
The cursor objects	8-3
The cursor methods	8-4
The cursor objects	8-5
The SearchCursor object	8-6
Search cursor example	8-7
The UpdateCursor object	8-8
Update cursor example	8-9
The InsertCursor object	8-10
Insert cursor example	8-11
The Geometry object	8-12
Geometry example	8-13
Data locks	8-14
Exercise goals	8-15
Lesson review	8-16
<b>9</b> <i>Running scripts from ArcToolbox</i>	
Running scripts from ArcToolbox	9-1
Learning objectives	9-2
Making scripts dynamic	9-3
Creating arguments: sys.argv[]	9-4
Creating arguments: GetParameterAsText()	9-5
Running scripts with arguments	9-6
Why attach a script to a tool?	9-7
Attaching a script to a tool	9-8
General settings	9-9
Setting the script source	9-10
Parameter properties	9-11
Filter property	9-12
Obtained From property	9-13
Final output	9-14
Script tools and the geoprocessing framework	9-15
Demonstration: Show existing script tools	9-16
Exercise goals	9-17
Lesson review	9-18
<b>10</b> <i>Debugging code and handling errors</i>	
Debugging code and handling errors	10-1
Lesson 10A learning objectives	10-2
Step 1: Where to test code	10-3
Step 2: Check for syntax errors	10-4
Step 3: Understand the script	10-5
Step 4: Enter any arguments	10-6
Step 5: Narrow error to block or line of code	10-7

Interactive Window	10-8
Print statements	10-9
Comment code	10-10
Debugger toolbar	10-11
Step 6: Visually examine the line of code	10-12
Activity A: Find seven errors	10-13
Activity B: Find four errors	10-14
Activity C: Find four errors	10-15
Step 7: Check data in ArcCatalog	10-16
A note of caution	10-17
Exercise 10A goals (SBS/SD)	10-18
Lesson 10B learning objectives	10-19
Why handle errors?	10-20
Checking for specific errors	10-21
Geoprocessor messaging methods	10-22
Testing whether data exists (PythonWin)	10-23
Testing whether data exists (ArcToolbox)	10-24
Wrong type of data	10-25
Checking for all errors	10-26
Try...except statement	10-27
Getting all messages from the last tool	10-28
Filtering messages	10-29
Returning individual messages	10-30
Other geoprocessor error-handling methods	10-31
Licensing methods	10-32
Extension example	10-33
Product example	10-34
Exercise 10B goals (SBS)	10-35
Lesson review	10-36

## *11 Course conclusion*

Course conclusion	11-1
Course objectives	11-2
ESRI educational support	11-3
Where do you go from here?	11-4
Learning pathways	11-5
Course evaluation	11-6
Thank you for attending	11-7