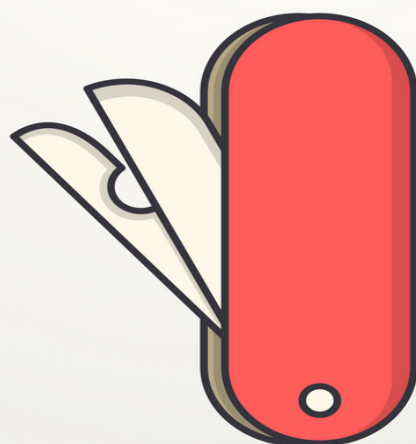


HOW TO UP YOUR SCRIPTING GAME ?

# POWERSHELL GUIDE TO PYTHON

A COMPARATIVE APPROACH TO LEARN  
**TWO LANGUAGES AT ONCE. FAST!**



**PRATEEK SINGH**

# PowerShell Guide to Python

A Comparative Approach to Learn two Scripting Languages at once. Fast!

Prateek Singh

This book is for sale at <http://leanpub.com/PowerShell-to-Python>

This version was published on 2019-07-31



This is a [Leanpub](#) book. Leanpub empowers authors and publishers with the Lean Publishing process. [Lean Publishing](#) is the act of publishing an in-progress ebook using lightweight tools and many iterations to get reader feedback, pivot until you have the right book and build traction once you do.

© 2018 - 2019 Prateek Singh

# Tweet This Book!

Please help Prateek Singh by spreading the word about this book on [Twitter](#)!

The suggested tweet for this book is:

[I just bought this amazing book: #PowerShell guide to #Python!](#)

The suggested hashtag for this book is [#PS2PY](#).

Find out what other people are saying about the book by clicking on this link to search for this hashtag on Twitter:

[#PS2PY](#)

# Contents

<b>1. Introduction</b>	<b>1</b>
<b>2. About the Book</b>	<b>2</b>
2.1 Why did I write this book	2
2.2 Who is the Target Audience	2
2.3 Prerequisites	3
2.4 Feedback	3
<b>3. The Basics</b>	<b>4</b>
3.1 Version	4
3.2 Case Sensitivity	4
3.3 Indentation	4
3.4 Comments	4
3.5 Variables	4
<b>4. Console Input/Output</b>	<b>6</b>
4.1 Printing to Console	6
4.2 Reading from Console	11
4.3 Reading from Console Securely	12
<b>5. Creating your First Script</b>	<b>15</b>
5.1 Saving Commands as a Script File	15
5.2 Running the Script File	15
<b>6. Passing Command-Line Arguments</b>	<b>16</b>
6.1 'sys.argv' in Python	16
6.2 '\$args' Automatic Variable in PowerShell	16
6.3 'Params()' in PowerShell	16
6.4 'argparse' Module in Python	16
6.5 Parameter Help	17
<b>7. Object Introspection</b>	<b>18</b>
7.1 Class, Object, Property, and Method	18
7.2 'Get-Member' cmdlet in PowerShell	19
7.3 'type()' Method in Python	19

## CONTENTS

7.4	'dir()' Method in Python . . . . .	19
7.5	'inspect' Module in Python . . . . .	19
<b>8.</b>	<b>The Help System . . . . .</b>	<b>20</b>
8.1	Get-Help cmdlet in PowerShell . . . . .	20
8.2	'help()' Method in Python . . . . .	20
8.3	Understanding How a Module Works . . . . .	20
<b>9.</b>	<b>Modules . . . . .</b>	<b>21</b>
9.1	Modular Programming . . . . .	21
9.2	Importing Modules . . . . .	21
9.3	Installing Modules . . . . .	22
9.4	Creating Modules . . . . .	22
<b>10.</b>	<b>Data Types . . . . .</b>	<b>23</b>
10.1	Common Data types . . . . .	23
10.2	Checking the Data Type . . . . .	24
10.3	Data type casting . . . . .	24
<b>11.</b>	<b>String Manipulations and Formatting . . . . .</b>	<b>25</b>
11.1	Common String Operations . . . . .	25
11.2	String Formatting . . . . .	25
11.3	Substring or String Slicing . . . . .	25
11.4	Testing String Membership . . . . .	26
11.5	Built-in methods . . . . .	26
<b>12.</b>	<b>Date and Time . . . . .</b>	<b>28</b>
12.1	Get Date and Time . . . . .	28
12.2	Convert String to datetime object . . . . .	28
12.3	DateTime formatting . . . . .	29
12.4	Time Span or Time Delta . . . . .	29
<b>13.</b>	<b>File Handling . . . . .</b>	<b>30</b>
13.1	Reading from a File . . . . .	30
13.2	File Modes . . . . .	30
13.3	Create and write to a text File . . . . .	30
13.4	Appending data to the File . . . . .	30
<b>14.</b>	<b>Arrays, List, ArrayList and Tuples . . . . .</b>	<b>31</b>
14.1	Arrays . . . . .	31
14.2	ArrayList (.Net) . . . . .	31
14.3	Lists . . . . .	31
14.4	Common PowerShell Array and Python Lists Operations . . . . .	31
14.5	Tuples . . . . .	33
14.6	Sets . . . . .	33

## CONTENTS

<b>15. Dictionary and Hashtable</b>	<b>34</b>
15.1 Hash Table	34
15.2 Dictionary	34
<b>16. Conditional Statements</b>	<b>36</b>
16.1 If Statement	36
16.2 Else Statement	36
16.3 The elif and ifelse statement	36
16.4 Nested Conditional Statements	36
<b>17. Loops</b>	<b>37</b>
17.1 For Loop	37
17.2 ForEach and ForeEach-Object	37
17.3 While Loop	37
17.4 Loop Control Statements	38
17.5 Nested Loops	38
<b>18. Functions</b>	<b>39</b>
18.1 Types of Functions	39
18.2 Creating a Function	39
18.3 Calling a Function	39
18.4 Functions vs Methods	39
18.5 Parameters and Arguments	39
18.6 Mandatory and Default parameters	40
18.7 The Anonymous Functions	40
<b>19. Handling XML and JSON Data</b>	<b>41</b>
19.1 Difference between JSON and XML	41
19.2 Reading XML from a file	41
19.3 Accessing XML as Objects	41
19.4 Accessing XML with XPath	41
19.5 Using XML for Object Serialization	41
19.6 Parsing JSON	42
19.7 JSON Serialization and De-Serialization	42
<b>20. Reading and Writing CSV Files</b>	<b>43</b>
20.1 What is CSV	43
20.2 Reading CSV File	43
20.3 Writing CSV File	43
<b>21. Error Handling</b>	<b>45</b>
21.1 Types of Errors	45
21.2 Raising an Exception or Throwing an error	45
21.3 Handling Exceptions using Try..catch..finally	46

## CONTENTS

<b>22. Built-In Functions</b>	<b>47</b>
22.1 Basic Mathematical Operations	47
22.2 Selection and Arrangement	48
22.3 Union and Intersection	49
22.4 Dynamic Execution of Code	49

# 1. Introduction

This PowerShell Scripting guide to Python is **designed to make readers familiar with syntax, semantics and core concepts of Python language**, in an approach that readers can totally relate with the concepts of PowerShell already in their arsenal, to **learn Python fast and effectively, such that it sticks with readers for longer period of time.**

**“Use what you know to learn what you don’t.”** also known as **Associative learning.**

PowerShell has been my scripting language of choice because of multiple reasons which includes its intuitiveness, command chaining with pipelines, smaller code and overall robustness! Undoubtedly it has shaped my career for good. As a scripting language it is very powerful when it comes to get the job done, although I do not intend to say that it is always the best tool to use. I feel when we love something we develop an deep affinity towards it, which seeds a bias towards it; and it grows into a preference and limits our learning somewhere. Which is just not right and I think we should be open to options and ideas, more specifically scripting/programming languages.

While PowerShell is still evolving in last 10 years since its inception, I see a lot of possibilities with more mature languages like Python.

Python is one of the top programming languages and in fast changing IT scenarios to DevOps and Cloud to the future - Data Science, Artificial Intelligence (AI) and Machine Learning Python is a must know. But this PowerShell Scripting guide to Python would be very helpful for you if you already have some knowledge of PowerShell.

*Prateek Singh*<sup>1</sup>

---

<sup>1</sup><https://twitter.com/SinghPrateik>



## 2. About the Book

### 2.1 Why did I write this book

PowerShell is my go to scripting language for almost ANYTHING! but When I started learning Python, I realized that I could see many patterns and concepts that resembled with PowerShell to a great extent. Probably this is because PowerShell is a very new scripting language compared to Python, and has adopted best features from older scripting and programming languages.

Despite of the reasons, the reality is that both Python and PowerShell has lot of similarities and there is a huge deal of overlap in the core concepts. Thus during my journey to learn Python, I could very easily relate it to the concepts of PowerShell that I had cemented in my brain with years of usage and fiddling around. Being able to relate and associate the concepts from two languages, helped me to quick grasp Python in no time.

Science supports this and coins a word around this known as “Associative Learning”, that means you can “Use what you know (PowerShell) to learn what you don’t (Python)”. That is why the book follows a comparative approach to moon shoot the readers to their journey in Python, and it will help if you already know PowerShell.

Another aspect of learning Python could be if you are into System administration then, you start to see how everything that works in PowerShell from a perspective of a programming language. This will in return help you in understanding subtle concepts that seemed obvious in PowerShell, now at a deeper level.

### 2.2 Who is the Target Audience

Python is one of the top programming languages in world and in fast changing IT scenarios to DevOps and Cloud, to the future - Data Science, Artificial Intelligence (AI) and Machine Learning; Python is a must know.

- Any System Administrator who wants to step into Development or Programming roles, and even if you don’t want to be a developer, knowledge of another scripting language will make your skill set more robust.

- Python Developers who want to learn PowerShell scripting and understand its ease of use and importance to manage any platform.

## 2.3 Prerequisites

Some prior experience and knowledge of PowerShell is recommended, but even if you are not experienced in PowerShell this book gives an opportunity to learn PowerShell basics in an effective approach.

All scripts and code references in this book is either Windows PowerShell v5.1 or Python v3.6, and I'll highly recommend you to install them on your local machines, before proceeding with reading chapters and trying out code samples in this book. Following are links and installation steps:

- [Windows PowerShell v5.1](#)<sup>1</sup>
- Python v3.6 - I'm using [Anaconda](#)<sup>2</sup> that is bunch of useful Python packages that come pre-installed, and [here](#)<sup>3</sup> is a link to the Anaconda user manual if you want to download that.

## 2.4 Feedback

In case you have any questions, comments, or your helpful feedback about this book, please share it via email to [prateek@ridicurious.com](mailto:prateek@ridicurious.com)<sup>4</sup>

---

<sup>1</sup><https://docs.microsoft.com/en-us/powershell/scripting/setup/installing-windows-powershell?view=powershell-6>

<sup>2</sup><https://www.anaconda.com/download/>

<sup>3</sup><https://conda.io/docs/user-guide/install/download.html>

<sup>4</sup><mailto:prateek@ridicurious.com>

## 3. The Basics

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 3.1 Version

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 3.2 Case Sensitivity

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 3.3 Indentation

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 3.4 Comments

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 3.5 Variables

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### **3.5.1 Rules for naming PowerShell variables**

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### **3.5.2 Rules for naming Python variables**

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### **3.5.3 Multiple Variable Assignment**

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 4. Console Input/Output

Python and PowerShell both are interactive scripting languages which can capture the user data from standard input and display the results on standard output or on your console. In this chapter we are going to look into some of these approaches and best practices to read and print to PowerShell and Python consoles.

### 4.1 Printing to Console

Python provides a built-in function: `print()`, which by default prints the value passed to the standard output, or any other specified stream.

```
1  # console output
2  print("Hello World!")
3  # string with single quotes
4  print('Hello World in Single quotes')
5  # \ backslash is the escape char for the single quote
6  print('Hi, We\'re going to store')
7  print('Hello','world') # concatenated printing to console
8  print('Hi'+ ' There') # concatenated string with + plus sign
9  print('I am ',28,'years old') # concatenated Numbers and strings
10 print(19) # print number
```

Just like `Write-Host` or `Write-Output` cmdlet in PowerShell, in fact in PowerShell, it's much simpler and we can directly output strings to the console.

But there is a catch, `Write-Host` sends output directly to the console.

```
1 # console output
2 Write-host "Hello World!"
3
4 # to the output stream
5 Write-Output "Hello World!"
6
7 # Alternatively
8 "Hello World!"
```

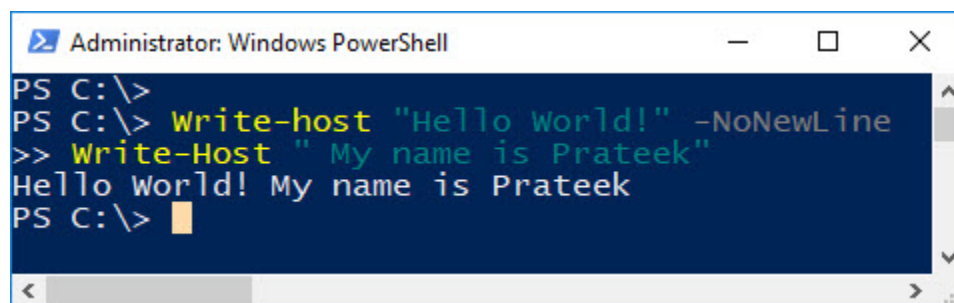
Strings inside quotes are sent to a different PowerShell Stream called Output stream (the default PowerShell stream), which can be piped and be used by cmdlets following the pipeline or can be stored in a variable.

### 4.1.1 Printing without a New-Line

Printing without a new line or in other words without adding a line feed character at the end of the output string is fairly simple in both PowerShell and Python.

In PowerShell when you don't want to add a line-feed you use `Write-Host` cmdlet with a `-NoNewLine` switch parameter, which will continue printing the next item just after the previous one, as demonstrated in the following example:

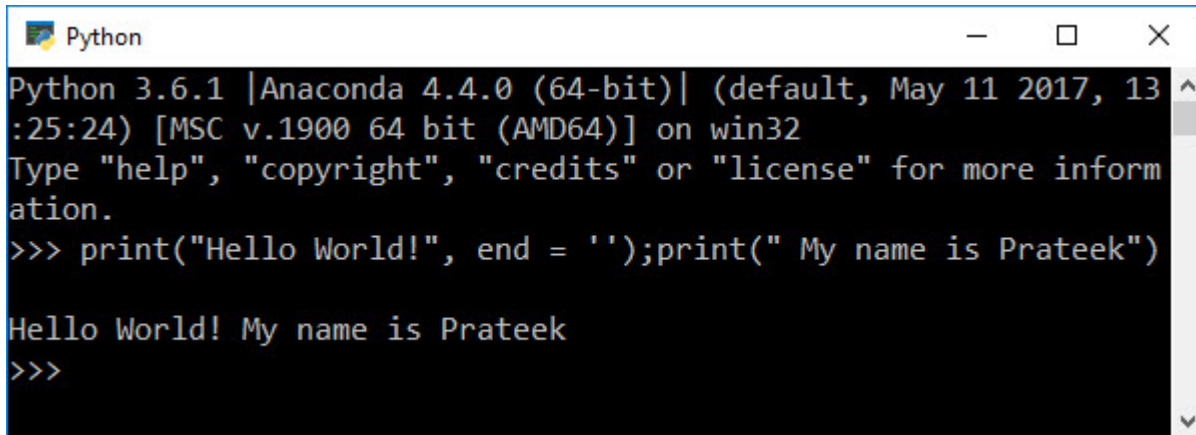
```
1 Write-host "Hello World!" -NoNewLine
2 Write-Host " My name is Prateek"
```

A screenshot of a Windows PowerShell console window titled "Administrator: Windows PowerShell". The prompt is "PS C:\>". The user enters the command "Write-host 'Hello World!' -NoNewLine". The prompt changes to ">>". The user enters the second command "Write-Host ' My name is Prateek'". The output displayed is "Hello World! My name is Prateek" on a single line. The cursor is at the end of the second line of input.

No New-Line in PowerShell

Whereas in Python, you've to use the named argument "end" to explicitly specify the end of line string, which should be an empty string ( ' ' ) if you want to continue printing in the same line. The default value of the named parameter is '\n' which is a new-line character in Python and passing an empty string overwrites this value.

```
1  # use the named argument "end" to explicitly specify the end of line string
2  print("Hello World!", end = '')
3  print(" My name is Prateek")
```



```
Python
Python 3.6.1 |Anaconda 4.4.0 (64-bit)| (default, May 11 2017, 13:25:24) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Hello World!", end = '');print(" My name is Prateek")
Hello World! My name is Prateek
>>>
```

No New-Line in Python

## 4.1.2 Output Field Separator

PowerShell offers an Automatic Variable `$ofs` that can store a separator string, which is used as the 'Output Field Separator' between the objects printed on the console. The default value of this automatic variable is " ", that is why when you convert an array of `[int]` objects to string and print it to console it adds a space between each integer object in the output.

```
1  PS C:\> 1,2,3,4
2  1
3  2
4  3
5  4
6  PS C:\> # adds space when converted to string output
7  PS C:\> [string] (1,2,3,4)
8  1 2 3 4
9  PS C:\> "$(1,2,3,4)"
10 1 2 3 4
11 PS C:\> $ofs = '**'
12 PS C:\> [string] (1,2,3,4)
13 1**2**3**4
14 PS C:\> "$(1,2,3,4)"
```

```

15 1**2**3**4
16 PS C:\>

```

```

Administrator: Windows PowerShell
PS C:\> 1,2,3,4
1
2
3
4
PS C:\> # adds space when converted to string output
PS C:\> [string] (1,2,3,4)
1 2 3 4
PS C:\> "$(1,2,3,4)"
1 2 3 4
PS C:\> $ofs = '**'
PS C:\> [string] (1,2,3,4)
1**2**3**4
PS C:\> "$(1,2,3,4)"
1**2**3**4
PS C:\>

```

### \$OFS in PowerShell

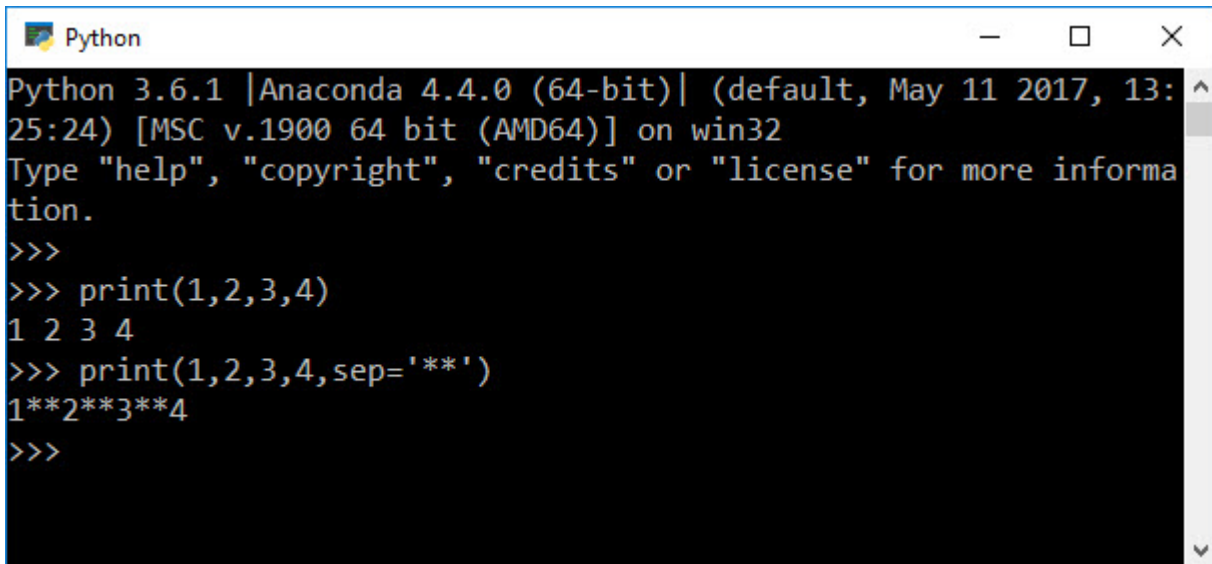
`print()` method in Python also provides similar functionality to explicitly define the separator, using the named argument `sep` which has a default value `sep=' '`. This is the reason for extra space introduced between integers printed using the `print()` function.

```

1 >>> print(1,2,3,4)
2 1 2 3 4
3 >>> print(1,2,3,4, sep='**')
4 1**2**3**4
5 >>>

```



A screenshot of a Python console window titled "Python". The window shows the following text: "Python 3.6.1 |Anaconda 4.4.0 (64-bit)| (default, May 11 2017, 13:25:24) [MSC v.1900 64 bit (AMD64)] on win32", "Type 'help', 'copyright', 'credits' or 'license' for more information.", and a prompt ">>>". The user enters "print(1,2,3,4)" and the output is "1 2 3 4". Then the user enters "print(1,2,3,4,sep='\*\*')", and the output is "1\*\*2\*\*3\*\*4". The prompt ">>>" is shown again.

```
Python 3.6.1 |Anaconda 4.4.0 (64-bit)| (default, May 11 2017, 13:25:24) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
>>> print(1,2,3,4)
1 2 3 4
>>> print(1,2,3,4,sep='**')
1**2**3**4
>>>
```

**'sep' attribute in Python's print() function**

### 4.1.3 Console Output to a File

`print()` method can also redirect output to a file, but by default, it goes to standard output. There is a named argument `'file'` that allows you to specify a file name so that the output can be redirected.

```
1 # notice the forward slashes in the path
2 filename = open('c:/temp/file.txt', 'w')
3 print('Pretty cool!', file = filename)
4 filename.close()
```

On another hand, same can be achieved in PowerShell by piping the output string to `Out-File` cmdlet, which will then save the string in a file. A new file would be created if it does not exist.

```
1 # unlike in python the path is in backward slashes
2 "Way easier in PowerShell" | Out-File C:\temp\file1.txt
```

Output Streams can also be redirected to a file using `'>'` greater-than operator and `'*'` in the following command represent a wildcard for all the output streams.

```
1 # redirect the stream to a file
2 Write-Host "Using stream redirection" *> c:\temp\file2.txt
```

### 4.1.4 Append Console Output to a File

There would be some scenarios where you want to append the console output to a file and in order to achieve this in Python, you have to first open the file in append mode ('a') and then simply print the output to the file.

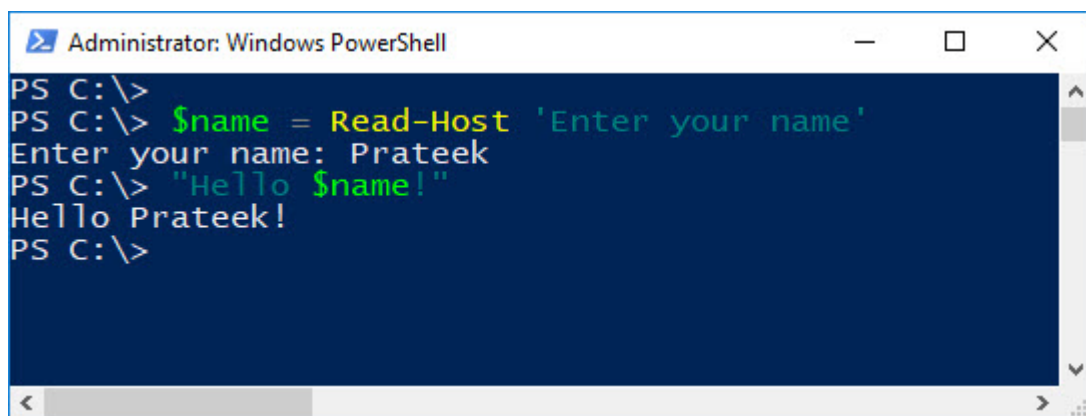
Whereas, in PowerShell you can use '>>' double greater-than operators to append the console output, while redirecting the streams. But with the `Out-File` cmdlet you should use the `-Append` switch parameter.

```
1 Write-Host "Using stream redirection" *>> c:\temp\file2.txt
2 "This is a string" *>> c:\temp\file2.txt
3 "Just add append switch" | Out-File c:\temp\file1.txt -Append
```

## 4.2 Reading from Console

Prompting input from users in the PowerShell console is as easy as using the cmdlet `Read-Host`, which can read a line of input from the console. You must specify a meaningful prompt string to the user for the input, and then capture user input to a variable.

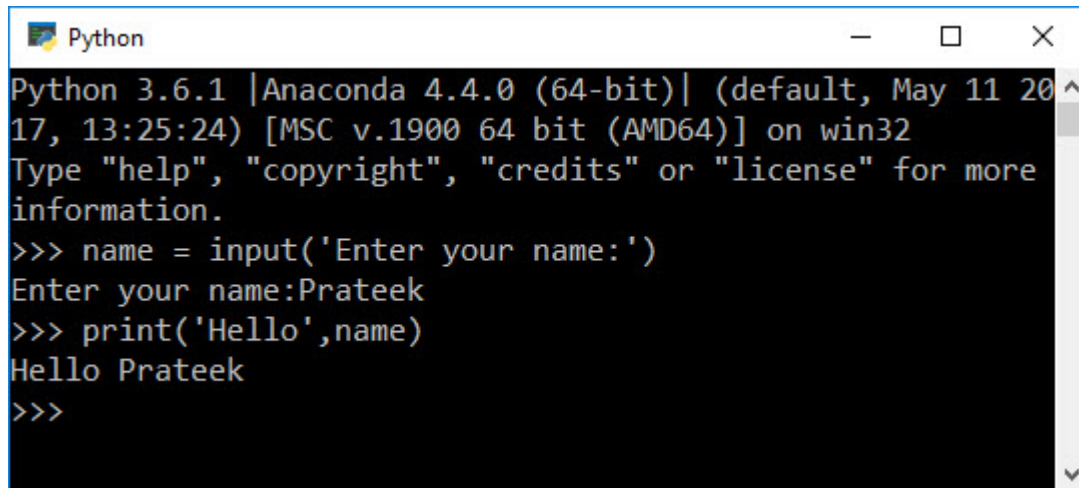
```
1 $name = Read-Host 'Enter your name'
2 "Hello $name!"
```



Reading from the host in PowerShell

Python on other hand provides a built-in method called `input()` to read a string from standard input from the console and the trailing new-line character (Line-Feed) when the user hits enter, is stripped away.

```
1 name = input('Enter your name:')
2 print('Hello', name)
```

A screenshot of a Python console window titled "Python". The window shows the following text: "Python 3.6.1 |Anaconda 4.4.0 (64-bit)| (default, May 11 2017, 13:25:24) [MSC v.1900 64 bit (AMD64)] on win32", "Type 'help', 'copyright', 'credits' or 'license' for more information.", and the execution of the code from the previous block. The prompt is ">>>". The user input "Prateek" is shown on the line following the first prompt. The output "Hello Prateek" is shown on the line following the second prompt. The prompt ">>>" appears again at the end.

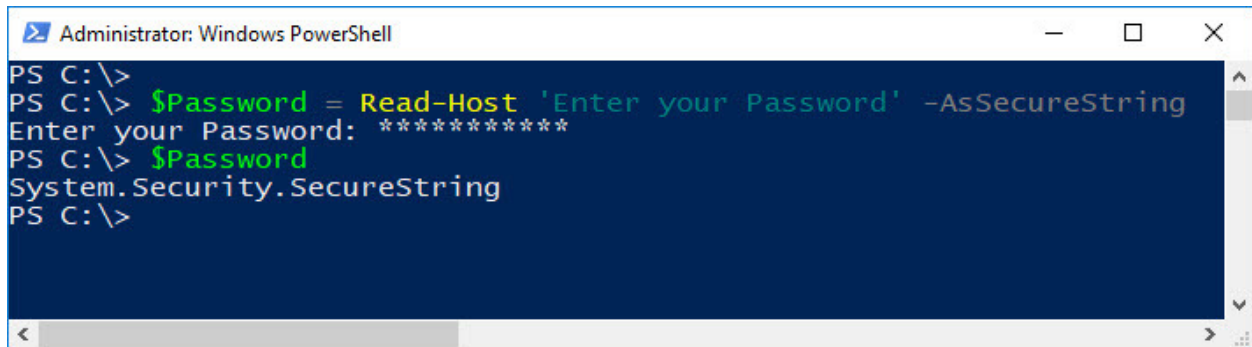
```
Python 3.6.1 |Anaconda 4.4.0 (64-bit)| (default, May 11 2017, 13:25:24) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> name = input('Enter your name:')
Enter your name:Prateek
>>> print('Hello',name)
Hello Prateek
>>>
```

Reading from the host in Python

## 4.3 Reading from Console Securely

PowerShell's `Read-Host` cmdlet when used with `-AsSecureString` switch parameter will hide your input in case you prompt users for secure data, such as passwords.

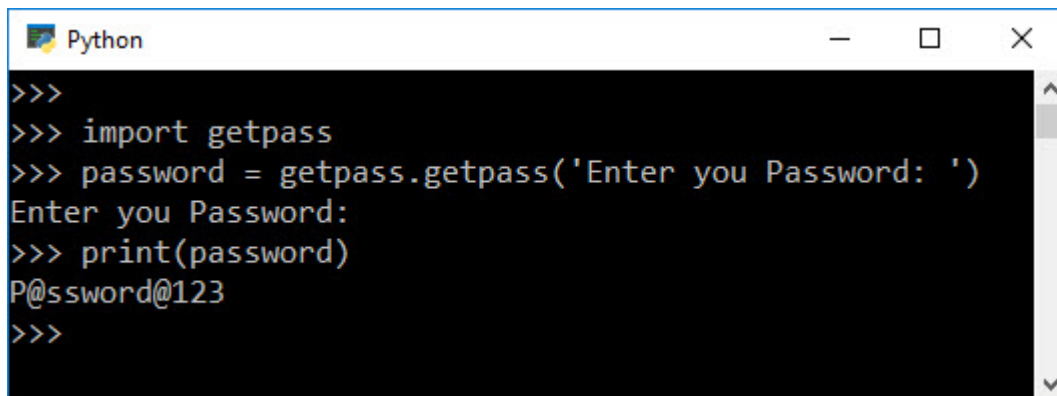
```
1 $Password = Read-Host 'Enter your Password' -AsSecureString
2 $Password
```

A screenshot of a Windows PowerShell window titled "Administrator: Windows PowerShell". The console shows the following commands and output: 

```
PS C:\>
PS C:\> $Password = Read-Host 'Enter your Password' -AsSecureString
Enter your Password: *****
PS C:\> $Password
System.Security.SecureString
PS C:\>
```

Reading from host securely in PowerShell

While in Python you can use `getpass()` method to turn off echo while prompting users a secure data, but data captured can still be viewed in plain text. Before using the method you've to first import the 'getpass' module into the current Python session.

A screenshot of a Python console window titled "Python". The console shows the following commands and output: 

```
>>>
>>> import getpass
>>> password = getpass.getpass('Enter you Password: ')
Enter you Password:
>>> print(password)
P@ssword@123
>>>
```

Reading from host securely in Python

## Key Pointers

- \* Python utilises `print()` function and PowerShell has `Write-Host` or `Write-Output` cmdlets to print output.

- \* Printing without a new-line

- \* PowerShell

```
1 Write-host "Hello World!" -NoNewLine
```

### \* Python

```
1      print("Hello World!", end = '')
```

- Built-in function `input()` is Python equivalent of PowerShell cmdlet: `Read-Host`
- `Read-Host` cmdlet in PowerShell with `-AsSecureString` switch parameter will hide user input to make it secure. Python has a `getpass()` Utility to get a password from a user with echo turned off.

## Reading Recommendations

To learn more, I'd advise you to please go through following PowerShell and Python help documentations from the console and/or find links to some additional external reading resources.

### PowerShell Help

- `Get-Help Write-Host`
- `Get-Help Write-Output`
- `Get-Help Read-Host`
- `Get-Help about_Redirection`

### Python Help

- `help(print)`
- `help(input)`
- `import getpass; help(getpass)`
- `help(open)`

### Links

- [Understanding Streams and Redirection<sup>a</sup>](https://blogs.technet.microsoft.com/heyscriptingguy/2014/03/30/understanding-streams-redirection-and-write-host-in-powershell/)

<sup>a</sup><https://blogs.technet.microsoft.com/heyscriptingguy/2014/03/30/understanding-streams-redirection-and-write-host-in-powershell/>

## 5. Creating your First Script

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 5.1 Saving Commands as a Script File

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 5.2 Running the Script File

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 6. Passing Command-Line Arguments

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 6.1 'sys.argv' in Python

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 6.2 '\$args' Automatic Variable in PowerShell

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 6.3 'Params()' in PowerShell

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 6.4 'argparse' Module in Python

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

#### 6.4.1 Creating the Argument Parser

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 6.4.2 Adding the arguments

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 6.4.3 Adding Multiple Arguments

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 6.4.4 Optional Arguments, Short Options, and Parameter Aliases

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 6.4.5 Default Values

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 6.5 Parameter Help

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.



## 7. Object Introspection

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 7.1 Class, Object, Property, and Method

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

#### 7.1.1 Class

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

#### 7.1.2 Object

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

#### 7.1.3 Attributes or Properties

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

#### 7.1.4 Functionalities or Methods

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 7.1.5 Instantiating the Class

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 7.1.6 Accessing Attributes and Functions of an Objects

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 7.2 'Get-Member' cmdlet in PowerShell

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 7.3 'type()' Method in Python

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 7.4 'dir()' Method in Python

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 7.5 'inspect' Module in Python

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 8. The Help System

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 8.1 Get-Help cmdlet in PowerShell

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 8.2 'help()' Method in Python

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

#### 8.2.1 String Argument passed to help() method

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

#### 8.2.2 No Argument passed to help() method

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 8.3 Understanding How a Module Works

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 9. Modules

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 9.1 Modular Programming

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 9.2 Importing Modules

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

#### 9.2.1 Importing Single Module

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

#### 9.2.2 Importing Multiple Modules

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

#### 9.2.3 Importing Sub-Modules

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 9.2.4 Module Aliases

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 9.2.5 PYTHONPATH and \$Env:PSModulePath

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

# 9.3 Installing Modules

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 9.3.1 Install-Module

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 9.3.2 PIP

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

# 9.4 Creating Modules

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

# 10. Data Types

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 10.1 Common Data types

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 10.1.1 Numbers

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

#### 10.1.1.1 Integers

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

#### 10.1.1.2 Floating-Point Numbers

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 10.1.2 Booleans

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 10.1.3 Strings

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

#### 10.1.3.1 String Interpolation or Variable Substitution

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

#### 10.1.3.2 Multi-line String

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

#### 10.1.3.3 Escape Characters in String

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 10.2 Checking the Data Type

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 10.3 Data type casting

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

# 11. String Manipulations and Formatting

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 11.1 Common String Operations

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 11.2 String Formatting

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 11.2.1 Format Operator in PowerShell : ' -f '

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 11.2.2 Format Method in Python : ' .format() '

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 11.3 Substring or String Slicing

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.



### 11.3.1 Using Range operator

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 11.3.2 Using Substring() method

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 11.4 Testing String Membership

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 11.5 Built-in methods

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 11.5.1 Length of string and Finding Character at an Index

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 11.5.2 Changing the Case

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 11.5.3 Split and Join Strings

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 11.5.4 Reversing a string

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 11.5.5 Removing Whitespace

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 11.5.6 Text Alignment

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

# 12. Date and Time

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 12.1 Get Date and Time

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 12.1.1 System.DateTime

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 12.1.2 'time' Module

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 12.1.3 'datetime' Module

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 12.2 Convert String to datetime object

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 12.3 DateTime formatting

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 12.4 Time Span or Time Delta

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

# 13. File Handling

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 13.1 Reading from a File

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 13.1.1 Reading the file line-by-line

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 13.2 File Modes

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 13.3 Create and write to a text File

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 13.4 Appending data to the File

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

# 14. Arrays, List, ArrayList and Tuples

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 14.1 Arrays

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 14.2 ArrayList (.Net)

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 14.3 Lists

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 14.4 Common PowerShell Array and Python Lists Operations

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 14.4.1 Finding Length

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 14.4.2 Indexing and access/modify elements

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 14.4.3 Insert/Add elements

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 14.4.4 Reverse elements

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 14.4.5 One to Many: Array to the Variable assignment

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 14.4.6 Remove elements

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 14.4.7 Array/List slicing

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 14.4.8 Multi Dimensional Array/List

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

#### 14.4.8.1 Two Dimensional Arrays

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

#### 14.4.8.2 Two Dimensional Lists

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 14.5 Tuples

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 14.6 Sets

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.



# 15. Dictionary and Hashtable

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 15.1 Hash Table

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 15.2 Dictionary

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 15.2.1 Accessing Key-Value pair

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 15.2.2 Nested HashTable and Dictionary

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 15.2.3 Adding Key-Value pair

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 15.2.4 Finding Keys and Values

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 15.2.5 Iterating through key-value pairs

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 15.2.6 Deleting a key-value pair

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 15.2.7 Sorting hashtable

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

# 16. Conditional Statements

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 16.1 If Statement

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 16.2 Else Statement

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 16.3 The elif and ifelse statement

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 16.4 Nested Conditional Statements

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

# 17. Loops

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 17.1 For Loop

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 17.1.1 Using Range() Method

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 17.1.2 Using List

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 17.2 ForEach and ForeEach-Object

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 17.3 While Loop

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 17.4 Loop Control Statements

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 17.4.1 Continue

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 17.4.2 Break

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 17.4.3 Pass

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 17.5 Nested Loops

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

# 18. Functions

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 18.1 Types of Functions

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 18.2 Creating a Function

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 18.3 Calling a Function

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 18.4 Functions vs Methods

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 18.5 Parameters and Arguments

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 18.6 Mandatory and Default parameters

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 18.7 The Anonymous Functions

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

# 19. Handling XML and JSON Data

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 19.1 Difference between JSON and XML

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 19.2 Reading XML from a file

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 19.3 Accessing XML as Objects

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 19.4 Accessing XML with XPath

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 19.5 Using XML for Object Serialization

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.



## 19.6 Parsing JSON

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 19.7 JSON Serialization and De-Serialization

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 20. Reading and Writing CSV Files

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 20.1 What is CSV

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 20.2 Reading CSV File

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

#### 20.2.1 Using PowerShell cmdlet `Import-CSV`

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

#### 20.2.2 Using Python's `csv` Library

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 20.3 Writing CSV File

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 20.3.1 Using PowerShell cmdlet `Export-CSV`

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 20.3.2 Using Python's `csv` Library

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

# 21. Error Handling

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 21.1 Types of Errors

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 21.1.1 Syntax errors

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 21.1.2 Runtime errors

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 21.1.3 Logic errors

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 21.2 Raising an Exception or Throwing an error

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 21.3 Handling Exceptions using Try..catch..finally

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 21.3.1 Try Block

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 21.3.2 Catch and Except Block

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 21.3.3 Finally Block

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 22. Built-In Functions

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 22.1 Basic Mathematical Operations

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

#### 22.1.1 Constants: PI and E

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

#### 22.1.2 Count and Sum: len() and sum()

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

#### 22.1.3 Absolute and Rounded Value: abs() and round()

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

#### 22.1.4 Maximum and Minimum: max() and min()

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 22.1.5 Powers and Square Roots: `pow()` and `sqrt()`

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 22.1.6 Trigonometric Functions: `sin()`, `cos()` and `tan()`

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 22.1.7 Logarithm: `log()` and `log10()`

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 22.1.8 Floor and Ceiling: `floor()` and `ceil()`

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 22.2 Selection and Arrangement

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 22.2.1 Random Selection

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

### 22.2.2 Sorting in Ascending and Descending Order: `sort()`

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 22.3 Union and Intersection

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.

## 22.4 Dynamic Execution of Code

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/PowerShell-to-Python>.