Working with variables, data types, and operators



Andrew Mallett

Author and software developer



Course introduction

Using variables

Data types

Test conditions

Printing OS information

Text chunking: four items

This is the first talking point

This is the second talking point

This is the third talking point

This is the fourth talking point



Variables

Variables are placeholders for information that you can refer to when needed. Variables are prefixed with the $ symbol. There are system variables and variables you can set.

Variables

```
Get-Variable
Get-Command gv
gv | more
$hosts = "c:\windows\system32\drivers\etc\hosts"
gv hosts
Get-Content $hosts
Remove-Variable hosts
```

We can list variables with Get-Variable or the alias gv. Using the = operator we can assign a value to a variable. The variable simplifies access to a file in this case.



Demo

Working with variables

(Optional) Describe your demo on one line here



Data types

All values must be of a specific data type, numbers or text are common types of data. Each type will map to objects which have the appropriate properties and methods,

Data types

```
$a = 1
$a.GetType()
$a = $a.ToString()
$a.GetType()
$a=1
$a="1"
$a=1.0
```

Variables can be assigned using the = operator. Their values can be reassigned.



Demo

Understanding data types

(Optional) Describe your demo on one line here



Testing values

Using simple logic and the if statement we can test values.

Conditions

```
if ($HOME -EQ $PWD) {
  Write-Host "Your working directory is also your home directory"
}
```

Conditional statements can be very useful. Here we test to see if you are working in your home directory.



Demo

Conditional logic

Using the if statement



Gathering system information

Some variables hold useful system information.

System information

```
Write-Host "Computername: $env:COMPUTERNAME"
Write-Host "User: $env:USERNAME"
$ip = (Get-NetIPAddress -AddressFamily IPv4 |
       Where-Object {$_.IPAddress -notlike "127.*"} |
       Select-Object -First 1).IPAddress

Write-Host "IP: $ip"
```

Some information we can grab easily from environment variables. The IP Address needs a little more work.



Demo

Printing system information