#-------------------------------------------------------------------------------------------------#

# A real world example of using PowerShell with SQL Server #

#-------------------------------------------------------------------------------------------------#

# Before we begin, load up the provider and SMO

. 'C:\PS\03 - SQL\02 - Load the Provider and SMO.ps1'

#-----------------------------------------------------------------------------------------------#

# Real World Example:

# Looking for columns of a certain data type using the SQL Provider

#-----------------------------------------------------------------------------------------------#

$machine = $env:COMPUTERNAME + "\SQL2012"

# Grab the start time so we can get some metrics on how long this runs

$Start = Get-Date

$matches = 0

$dbCollection = (Get-Item SQLSERVER:\sql\$machine\databases -Force).Collection

foreach($db in $dbCollection)

{

$rootPath = "SQLSERVER:\sql\$machine\databases\$($db.Name)\"

$tablePath = "$rootPath\tables"

$tableCollection = (Get-Item $tablePath -Force).Collection

foreach($table in $tableCollection)

{

$tableName = "$($db.Name)\$($table.schema).$($table.name)"

$columnPath = "$rootPath\tables\$($table.Schema).$($table.Name)\Columns"

$columnCollection = (Get-Item $columnPath).Collection

foreach($column in $columnCollection)

{

if($column.DataType.ToString() -eq 'bigint' )

{

"$tableName.$($column) is a BigInt"

$matches++

}

}

}

}

$End = Get-Date # Stop the timer

"`n"

"$matches Matches"

# The end-start results in a date-time object, which you can get the

# various properties of, including total milliseonds or seconds

$elapsed = $end - $start

"Elapsed Time $($elapsed.TotalSeconds) Seconds ( $($elapsed.TotalMilliseconds) Milliseconds)"

# 44 Matches

# Elapsed Time 73.063179 Seconds ( 73063.179 Milliseconds)

##

#-----------------------------------------------------------------------------------------------#

# Real World Example:

# Looking for columns of a certain data type using SMO

#-----------------------------------------------------------------------------------------------#

$machine = $env:COMPUTERNAME + "\SQL2012"

$Start = Get-Date

$matches = 0

$Server = New-Object Microsoft.SqlServer.Management.Smo.Server("$machine")

foreach($database in $Server.Databases)

{

foreach($table in $database.Tables)

{

$tableName = "$($database.Name)\$($table.schema).$($table.Name)"

foreach($column in $table.Columns)

{

if($column.DataType.ToString() -eq "bigint" )

{

"$tableName.$($column.Name) is a BigInt"

$matches++

}

}

}

}

$End = Get-Date

"`n"

"$matches Matches"

$elapsed = $end - $start

"Elapsed Time $($elapsed.TotalSeconds) Seconds ( $($elapsed.TotalMilliseconds) Milliseconds)"

# My test on my system:

# 44 Matches

# Elapsed Time 14.1718106 Seconds ( 14171.8106 Milliseconds)

# Same but write it to a file

$machine = $env:COMPUTERNAME + "\SQL2012"

$Start = Get-Date

$report = "" # Holds the output for our report file

$finds = 0

$Server = New-Object Microsoft.SqlServer.Management.Smo.Server("$machine")

$dbcnt = $Server.Databases.Count

Clear-Host

foreach($database in $Server.Databases)

{

Write-Host $("{0:00} Databases left to process" -f $dbcnt)

foreach($table in $database.Tables)

{

$hasHeaderPrinted = $false

[string]$tableName = "$($database.Name)\$($table.schema).$($table.Name)"

if($tableName.Length > 100)

{$padDash = 2}

else

{$padDash = 100 - $tableName.Length}

foreach($column in $table.Columns)

{

if($column.DataType.ToString() -eq "bigint" )

{

if($hasHeaderPrinted -eq $false)

{

$report += "`r`n -- $tableName $("-" \* $padDash) `r`n" # Just to see it nicely

$hasHeaderPrinted = $true

}

$report += " {0:0000}: $tableName.$($column.Name) is a BigInt`r`n" -f ++$finds

}

}

}

$dbcnt--

}

Set-Content -Value $report -Path "C:\PS\SQL Report.txt"

$End = Get-Date

"`n"

$elapsed = $end - $start

"Elapsed Time $($elapsed.TotalSeconds) Seconds ( $($elapsed.TotalMilliseconds) Milliseconds)"

# Display content in output pane

Get-Content "C:\PS\SQL Report.txt"

# Show content in Notepad

notepad "C:\PS\SQL Report.txt"

##

###