**Practice Exercises for PowerShell Beginner**

 June 25, 2018   [PowerShell](https://techsckool.com/category/powershell/)

**Are you new to the PowerShell world?**

It is very obvious to think of getting your hands dirty on PowerShell Scripting. However, the challenge is where to start from. I would suggest start with doing tiny automation of the tasks which you are doing manually. Example: Restart your computer once a week.

Here I have listed few basic tasks which you can perform using PowerShell to get initial exposure of scripting and will definitely give you the confidence to approach bigger requirements.

**Note: Please perform each and everything using PowerShell(even for navigation purpose)**

.

**Offer For you:**

Enroll in my online 'Best Selling' Udemy courses and explore the world of PowerShell & Python in an exciting way  I offer every course at the minimum possible price to my readers.   Wish you a very happy scripting journey in advance.

**Plan to invest in your carrier today.** [**Here is the link**](https://techsckool.com/courses/)**.   All my courses are insured by 30 days money back guarantee. No Questions Asked**

**Practice Exercise1: File Operations/Searching/Basic PowerShell Operators**

1.) Create a folder **TestingPurpose** and 3 Subfolders inside it SubFolder1, SubFolder2

2.) Create some test files inside these folders:

TypeATest1.txt, TypeATest2.txt  … TypeATest50.txt into SubFolder1

TypeBTest51.txt, Purpose52Test2.txt … TypeBTest100 into SubFolder2

Needless to say that you have to use logic for creating these files. Not one by one

3.) Move all files which have an odd number in its name to SubFolder2

4.) Move all files which have even number in its name to SubFolder1

5.) Rename folder SubFolder1 to EvenFilesContainer and SubFolder2 to OddFilesContainer

5.) Prepare a list of all files currently existing inside folder **TestingPurpose**

**Example:** MasterFile.txt:

As of YYYYMMDD HH: MM files inside Testing Purpose are:

C:\testingPurpose\EvenFilesContainer\TypeBTest2.txt

.

.

C:\testingPurpose\OddFilesContainer\TypeATest99.txt

6.) Delete all files which start with TypeA

**Practice Exercise 2:  Windows Service Related**

Write PowerShell One-liners for:

1.) Get all services which are stopped

2.) Get all services whose name starts with letter "A"

3.) Get all services which are set to start automatically (look for property      StartType  : Automatic)

4.) Restart-Service Winmgmt

5.) Export the service name and status into a text file.

Example:

Service Name,  Status

Service A,   Running

Service B,    Stopped

6.) Export the service name, StartType, and status into an HTML file.

**Practice Exercise 3:  Windows Process Related**

Write PowerShell One-liners for:

1.) Get all windows processes whose name starts with letter "A"

2.) Get list of processes whose name is **svchost** and PM more than 100MB

3.) Get Process Name, Process ID and handleCount whose PM is more then 100MB and CPU more than 1000s

4.) Export the results of (3) to html and CSV format

**Practice Exercise 4:  User Input**

Simple interest calculation on any principal amount involves variables like Principle amount, Interest rate and tenure for which you are calculating SI

Simple interest can be calculated by using below formula

**SI = P \* R \* T / 100**

where,

P = Principal amount

 R = Rate of Interest

T = Time(in years)

Write a PowerShell to take user inputs and show the results user

**Practice Exercise 5:  Programming using PowerShell**

Write a logic using nested loops(for loop or while loop) to draw the below pattern

#

##

###

####

#####

######

#######

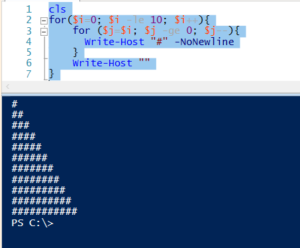
########

#########

##########

###########

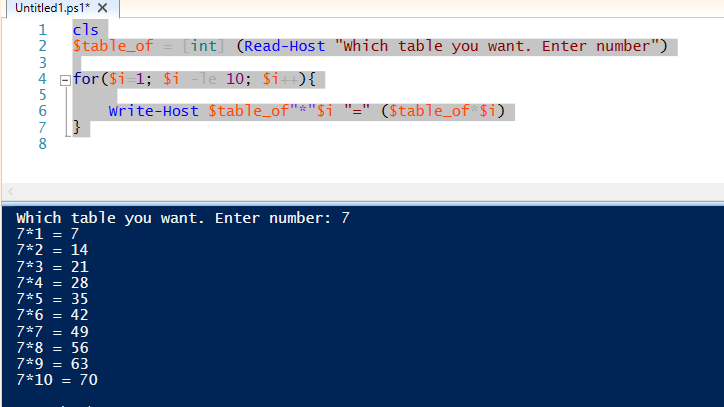
Note: Copy the solution only after trying it yourself



cls  
for($i=0; $i -le 10; $i++){  
for ($j=$i; $j -ge 0; $j--){  
Write-Host "#" -NoNewline  
}  
Write-Host ""  
}

**Practice Exercise 6:  Programming using PowerShell**

Write a logic to produce the mathematical table of a given number. The number should be a dynamic value and provided by the user at the run time.



cls  
$table\_of = [int] (Read-Host "Which table you want. Enter number")

for($i=1; $i -le 10; $i++){

Write-Host $table\_of"\*"$i "=" ($table\_of\*$i)  
}

**Practice Exercise 6:  Programming using PowerShell**

**Practice Exercises for PowerShell Intermediate**

 June 25, 2018   [PowerShell](https://techsckool.com/category/powershell/)

Dear Friend,

If you have already gone through   [practice exercises for PowerShell beginner,](https://techsckool.com/practice-exercises-for-powershell-beginner/)  This is another opportunity to take your Scripting knowledge to next level.

Please go through the problems, develop a clear understanding, make a rough sketch and apply your PowerShell knowledge to solve the problems.  Use PowerShell's help as much possible in case you need, but don't directly search on the internet. Trust me, solving these problems will definitely boost your confidence.

**New User Offer:**

Enroll in my online 'Best Selling' Udemy courses and explore the world of PowerShell & Python in an exciting way  I offer every course at the minimum possible price to my readers.   Wish you a very happy scripting journey in advance.

**Plan to invest in your carrier today.** [**Here is the link**](https://techsckool.com/courses/)**.   All my courses are insured by 30 days money back guarantee. No Questions Asked**

**Problem 1: Service restart on multiple computers and logging.**

Write a PowerShell script to read the computer names from a text file Then,

1. Stop a given service (say Print Spooler service ) and wait for 30 seconds after logging the status into a dedicated log file.
2. Ensure no child process is alive so that graceful stop of service can be confirmed
3. If there is any child process, kill it forcefully and log the information into the log file
4. After waiting for 30 seconds, start the service
5. Check for the service status, log into the log file and come out.

Next level,  You can see the above process might take around 2 minutes of time for gracefully restarting a service on 1 server.  using a single thread, It is not a good solution if we have to gracefully restart 5 services on 1000 machines(ETA: 10,000 minutes).

So, Use multithreading to improvise the solution. (hint: Invoke-Command or Start-Job might help you here)

**Problem 2:   Event Viewer**

Write a quick PowerShell script which,

-> Read multiple server names from a text file

-> Ask the user to specify which log they want to scan- like Application, System etc

-> Upon providing input, ask for event ID which they are looking for

Once this information is provided, Script should scan all the computers event logs for provided eventID and generate a nicely formatted CSV report whose headers should be:

**"MachineName","TimeGenerated","Source","Message"**

**Problem 3: Task Scheduler**

There are scheduled tasks **Important\_Service\_Restart** and **Important\_Job\_Processing**running on 100 machines(you have their names in a text file). Tasks are scheduled to run on the daily basis and they are critical for your business.

You need to write a PowerShell script to run after 15 minutes of task scheduled time and collect the status into a CSV file(each task, each computer).

Next, Read the CSV file and get the name of servers where **the last run**of the task was failed.

Servers on which one task was failed, action should be sending an email to the support asking them to look into this urgently.

Servers on which both tasks were found in failed status, send an email to support and send a separate email to management informing them about the severity of the situation.

Next, Once your standard solution is ready, improvise your solution to decrease the overall script execution time(hint:; use multi threading)

**Problem 4: Daily backup**

You have a folder in your computer(say C:\Important\My Coding Practice ).

Since you do some code changes on daily basis, you want to set up a daily backup using PowerShell.  Preferred time for backup is 10PM.  The format of backup should be .zip with the date appended in the name.

When you are in office, your preferred backup location is a shared folder which is accessible to you by UNC path(say \\XYZ\_CORPS\share\associates\personal\).

But not all the time, you stay in office till 10 PM, and UNC path is not available outside of office environment. In that case, you have to take a local backup to C:\Archives. Whenever you are in office at the time of scheduled backup, your local backup should also be moved to UNC path as it is safer.

Your space in shared folder at work is limited, so you want to ensure that no more than last 30 backups are available at the backup directory.

**Please find a solution using Windows PowerShell and Windows task Scheduler**