

Powershell Interview Questions and Answers

Q1) What do you think is Microsoft PowerShell?

Ans: Microsoft PowerShell is a command-line shell and scripting language built on .NET. It helps system admins and power-users to quickly manage OS. (Windows, MacOS and Linux) and processes.

Q2) Define the key features of PowerShell?

Ans: The prime characteristics of Microsoft PowerShell can be summed up as follows -

- PowerShell is a scripting environment
- PowerShell commands are customizable
- The programming language isn't text-based. It's object-based.

Q3) What makes up the variables in PowerShell?

Ans: Variables contain objects, integers, and strings.

Q4) In PowerShell, what according to you is the implication of brackets?

Ans: We use () Parentheses Brackets in compulsory arguments.

We use {} Braces Brackets for blocked statements

We do not normally use [] Square Brackets. These brackets specify optional items.

Q5) What is your understanding of cmdlet?

Ans: Cmdlet are commands scripted in languages such as PowerShell VB or C#.

Q6) Are you aware of PowerShell Loop?

Ans: When repetitive tasks are automated using PowerShell, it is called PowerShell Loop.

Q7) Can PowerShell scripts be created for the deployment of SharePoint components?

Ans: If the web part is created employing VS2010, SharePoint components can be deployed with the help of ctrl+f5. Features of web part can be activated through PowerShell script.

Q8) What is your idea about comparison operators in PowerShell?

Ans: Comparison operators in PowerShell compare values. Comparison operators can be classified into four types - Match, Equality, Replace, and Containment. To declare variables, -eq comparison is used. Similarly, -lt is used for 'less than', -gt for 'greater than', and 'not equal' is denoted by -ne.

Q9) What is the use of the PowerShell pipeline?

Subscribe to our youtube channel to get new updates..!

Ans: When we have to join two statements, we use the PowerShell pipeline. In this case, one statement's output emerges as the second statement's input.

Q10) What do you understand by get-command in PowerShell?

Ans: PowerShell's get-command is applied to fetch cmdlets. For instance, if anyone is searching cmdlet between E and K, then the get-command would look like -
get-command [E-K]*

Q11) Can you explain how network drive is mapped in PowerShell?

Ans: For mapping network drive, the following PowerShell commands have to be executed -
PowerShell Map Network Drive

```
1 $Net = $( New-Object - ComObject Wscript.Network )  
2 $Net.MapNetworkDrive( "S:", expertmyTectra )
```

Q12) Do you know the methods to 'Select' in PowerShell?

Ans: The simplest way is WQL (WMI Query Language) statement. This technique uses the '-query' in order to introduce 'Select * from'.

Select-String is the next method to 'Select' in PowerShell. In this scenario, matching pattern, phrase, and word is checked by the cmdlet.

The last way is through Select-Object.

Q13) What is PowerShell's Get-ServiceStatus function?

Ans: The functions of Get-ServiceStatus enable filtering of window services. PowerShell lists the services that are 'Running', also, the services that are 'Stopped' through scripting.

Powershell Interview Questions For Experienced

Q14) Since you mentioned scripting, define what Scripting in PowerShell is?

PowerShell Certification Training!

Explore Curriculum

Ans: PowerShell files contain an array of commands, where each command appears in separate lines. When using text file, the filename will have the .PS1 extension. PowerShell script can be run by -
Typing commands in text editor
Saving file with the .ps1 extension
Executing the file

Q15 - Can you explain the hash table functions in PowerShell?

Ans: PowerShell's hash table can be referred as a dictionary. Basically, it is a series that enables users in storing data with a pair association known as "key-value". Here, both "key", as well as "value", might comprise of any length and data. In order to declare hash table users will have to apply @ in front of curly braces.

Q16) Can you explain the uses of PowerShell Array?

Ans: PowerShell Array is used to run scripts on remote computers. To create PowerShell Array, users will need to create variable, as well as they will have to assign the PowerShell Array. PowerShell Arrays are expressed with a "@" symbol, and illustrated as hashtable.
For instance, it would be like -

```
1 | $arrmachine = @ ( "machine1" , "machine2" , "machine3")
```

Q17) Define what commands to use in order to get child folders into a specified folder?

Ans: We will have to apply parameter recurse to get child folders into a specified folder. It will be -
Get-ChildItem C:Scripts -recurse

Q18) Can you explain the ways to convert the object to HTML?

Ans: In order to convert the object to HTML, we will have to execute -

```
1 | Get-Process | Sort-object - property CPU -descending | convert to - HTML |
```



Q19) Can you define how you will assign a different name to a variable?

Ans: A variable can be renamed by using -

Rename-Item- Path Env: MyVariable –NewName MyRenamedVar

Q20) Are you aware of \$input variable? What are its functions?

Ans: \$input variable activates a function that enables pipeline data access.

Q21) Can you illustrate the code used to locate names of installed applications on a current computer?

Ans: We will have to employ the following code -

```
1 | Get-WmiObject-Class Win32_Product
```

- ComputerName. | Format-wide-column1

Q22) How do you think you will find that sql services in PowerShell are in one server?

Ans: We can do this in two ways -

```
1 | get-wmiobject win32_service | where-object {$_.name-like "*sql*"}
2
3 | get-service sql*
```

Q23) What according to you is an efficient way to locate all sql services in one server?

Ans: In this case also, we can do it in two ways -

```
1 | get-wmiobject win32_service | where-object {$_.name -like "*sql*"}
2 | get-service sql*
```

Q24) Are you aware of the class that can enable us to identify if mc is 64 bit or 32 bit?

```
1 | win32_computersystem.
```

We can use this as follows -

```
1 | PS C:> $server = gwmi -cl win32_computersystem
2 | PS C:> $server.SystemType X86-based PC
```

Q25) What is your idea about Variable Interpolation?

Ans: When a variable is added to double-quoted strings, then PowerShell changes the name of that variable through its value. Typically, this feature in PowerShell is known as variable interpolation.