

# How-to Run a PowerShell Script – All Options Explained

lazyadmin.nl/powershell/run-a-powershell-script

December 21, 2021

PowerShell scripts are a great way to automate tasks, gather information, or modify properties in bulk. Most PowerShell commands are only a single line that you can just copy and paste into PowerShell. But how do you run a complete PowerShell Script?

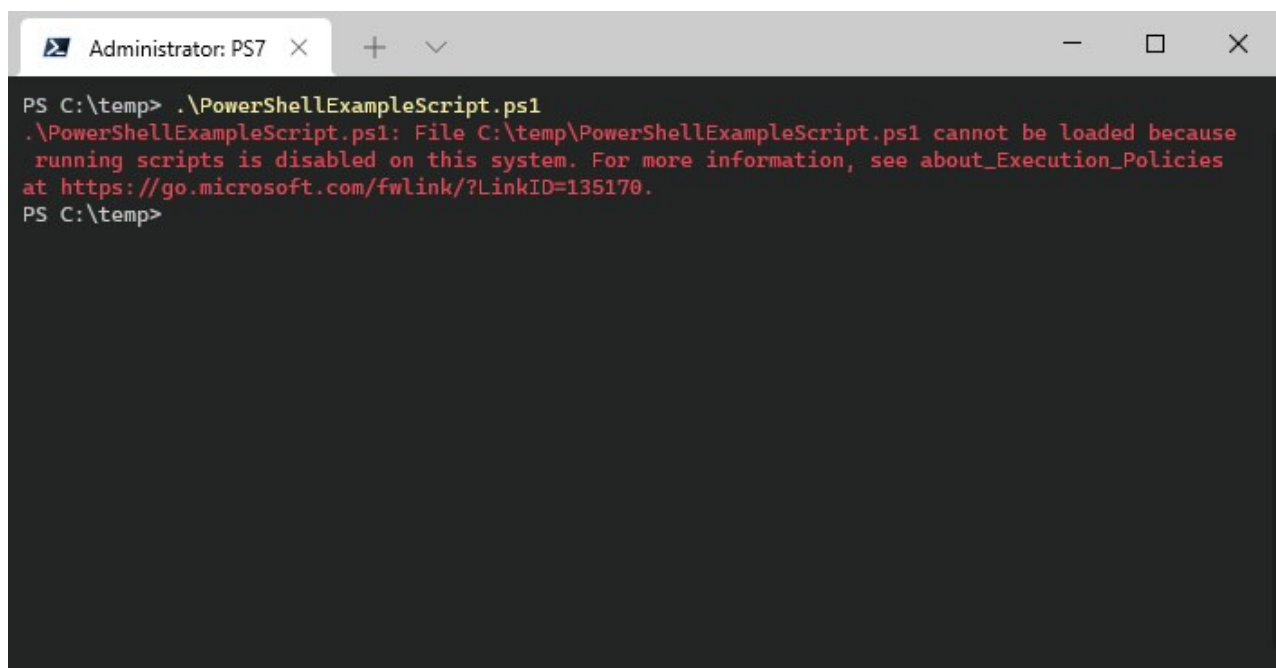
There are a couple of options when it comes to running PowerShell scripts, simply from the cmd, using the PowerShell ISE, or even a scheduled task.

In this article, I will explain the different options to run a PowerShell script. Also, I will help you with common errors, like “running scripts is disabled on this system”.

## Fix Running Scripts is Disabled on this System

We are going to start with a common error that prevents most users from running a PowerShell script, the error “Running scripts is disabled on this system”. This error is caused by the PowerShell Execution Policy.

By default, the PowerShell Execution policy is set to **Restricted**. This means that PowerShell scripts won’t run at all.

A screenshot of a PowerShell console window titled "Administrator: PS7". The window has a dark background. The command prompt shows the user is in the C:\temp directory and has entered the command ".\PowerShellExampleScript.ps1". The output is a red error message: ".\PowerShellExampleScript.ps1: File C:\temp\PowerShellExampleScript.ps1 cannot be loaded because running scripts is disabled on this system. For more information, see about\_Execution\_Policies at https://go.microsoft.com/fwlink/?LinkID=135170." followed by a new prompt "PS C:\temp>".

```
PS C:\temp> .\PowerShellExampleScript.ps1
.\PowerShellExampleScript.ps1: File C:\temp\PowerShellExampleScript.ps1 cannot be loaded because
running scripts is disabled on this system. For more information, see about_Execution_Policies
at https://go.microsoft.com/fwlink/?LinkID=135170.
PS C:\temp>
```

So to run a PowerShell script, we first need to change the execution policy. For the policy we have a couple of options:

Execution Policy	Description
<b>Restricted</b>	Default option – does not allow to run any PowerShell script
<b>Unrestricted</b>	Can run any script, shows warning for downloaded scripts
<b>RemoteSigned</b>	Requires a digital signature for downloaded scripts. You can run locally written scripts. You can unblock downloaded scripts to run them without signature
<b>ByPass</b>	You can run all scripts and no warnings are displayed
<b>AllSigned</b>	You can only run signed scripts from trusted publishers

Execution Policies are not designed as a security model, but more to prevent the accidental execution of a PowerShell script. The best option is to use the **RemoteSigned** policy, this way you can't run a downloaded script by accident (you will have to unblock it first). Read more about the [execution policy in this article](#).

Tip

*Learn more about writing your own PowerShell scripts in [this complete guide](#)*

To change the execution policy in PowerShell (and fix “Running scripts is disabled on this system”) we need to start PowerShell with administrator privilege:

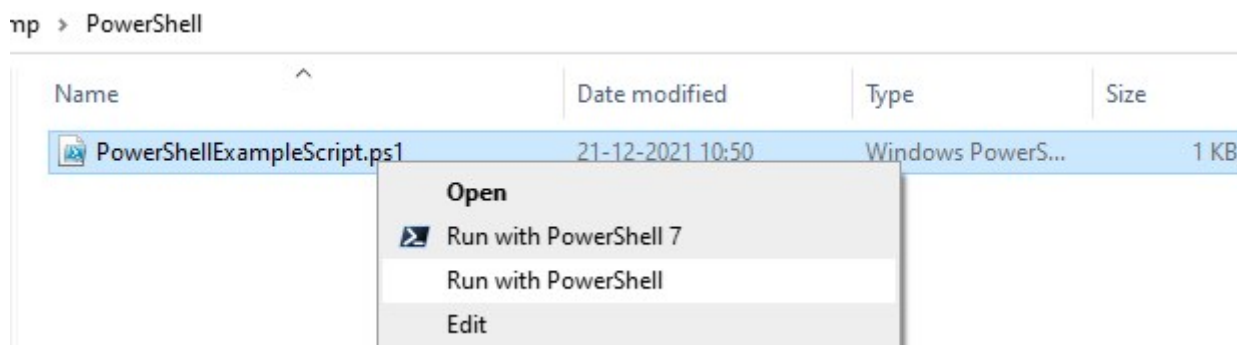
1. Press **Windows key + X** (or right click on the start menu)
2. Choose **Windows PowerShell (admin)**
3. **Run the command below:**

```
Set-ExecutionPolicy RemoteSigned
```

You should now be able to run PowerShell Scripts.

## Run a PowerShell Script

There are a couple of options to run a PowerShell script. The most convenient way is to simply right-click the file and choose **Run with PowerShell**.



Run PowerShell Script

But this method comes with a downside. By default, most PowerShell scripts will close the PowerShell window automatically when the script is done. So the script gets executed, but you won't be able to read any errors or output.

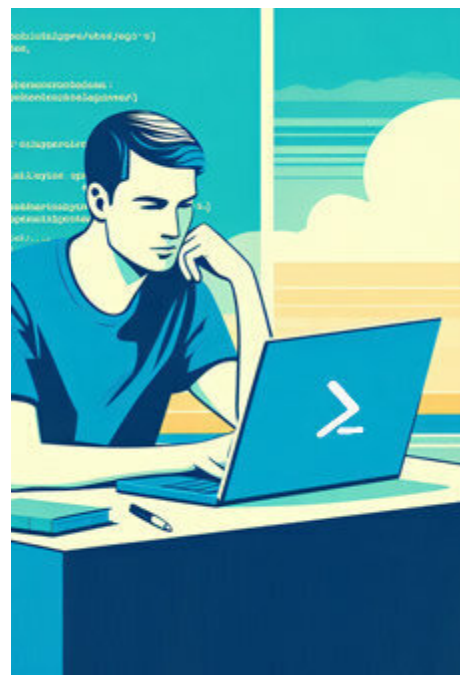
You can solve this by adding the following line to the end of the PowerShell script, but that is not always a convenient option:

Read-Host -Prompt "Press Enter to exit"

## FREE EMAIL SERIES!

## Level Up with PowerShell

## 5 Emails, Endless Skills

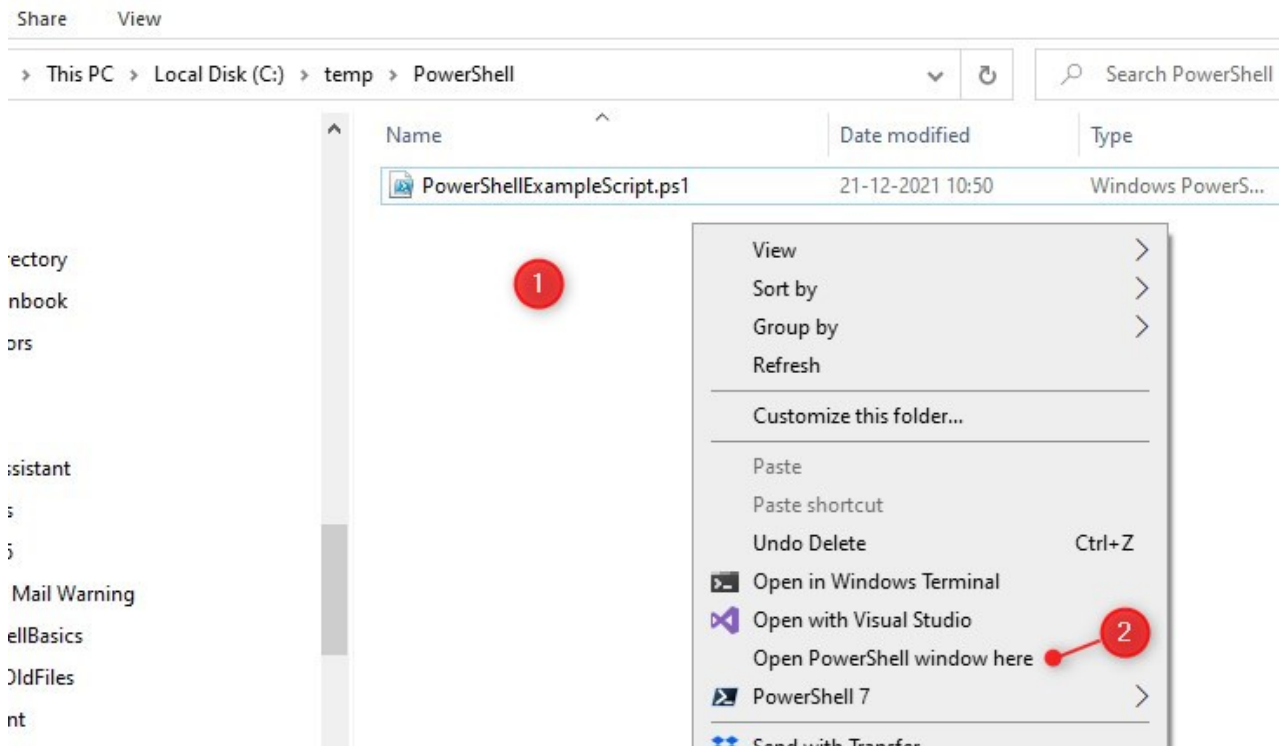


## Use Windows PowerShell

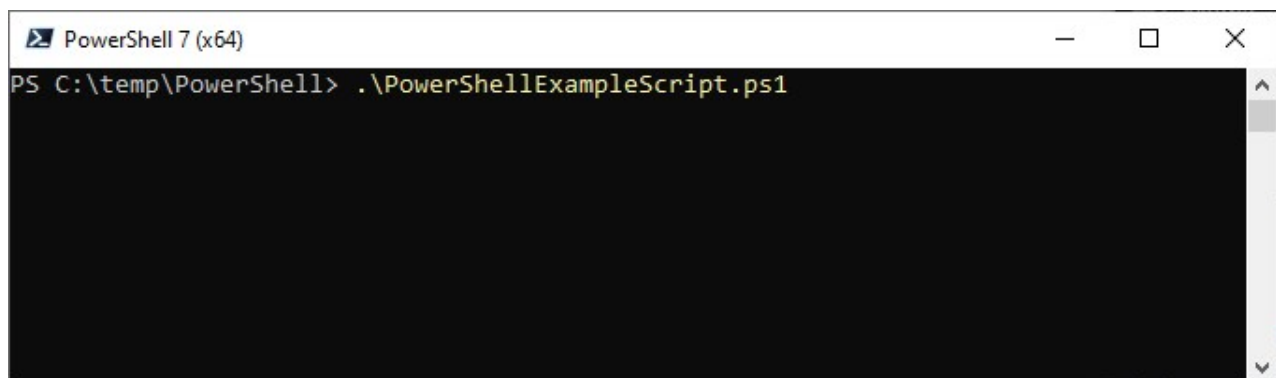
Personally, I prefer to start PowerShell scripts from the command line in Windows PowerShell itself. There are two ways to do this, you can first start PowerShell and navigate to the script or open PowerShell in the correct location from the explorer.

We start with the latter, opening the PowerShell in the correct location from the explorer. In the explorer, locate the script that you want to run, in this case, `c:\temp\powershell`.

1. **Right-click** on the blank area in Explorer (not on the PowerShell file)
  2. Choose **Open PowerShell window here**
- (In Windows 11 you will need to select Open in Windows Terminal)

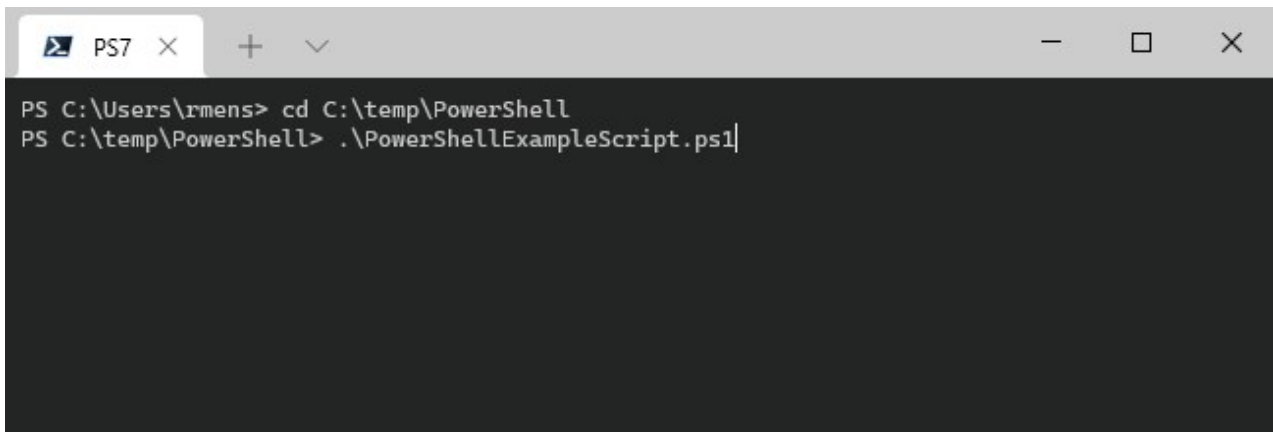


As you can see, PowerShell is started in the folder `c:\temp\PowerShell`. We can now run the script by simply typing the filename (type the first few letters and press Tab to autocomplete it)



You can also first start Windows PowerShell:

1. Right-click on the start menu (or press **Windows key + X**)
2. Choose **Windows PowerShell**
3. **Navigate** to the folder where the script is located  
`cd c:\path\to\script <enter>`
4. **Run** the PowerShell script  
`.\PowerShellExampleScript.ps1 <enter>`



```
PS C:\Users\rmens> cd C:\temp\PowerShell
PS C:\temp\PowerShell> .\PowerShellExampleScript.ps1
```

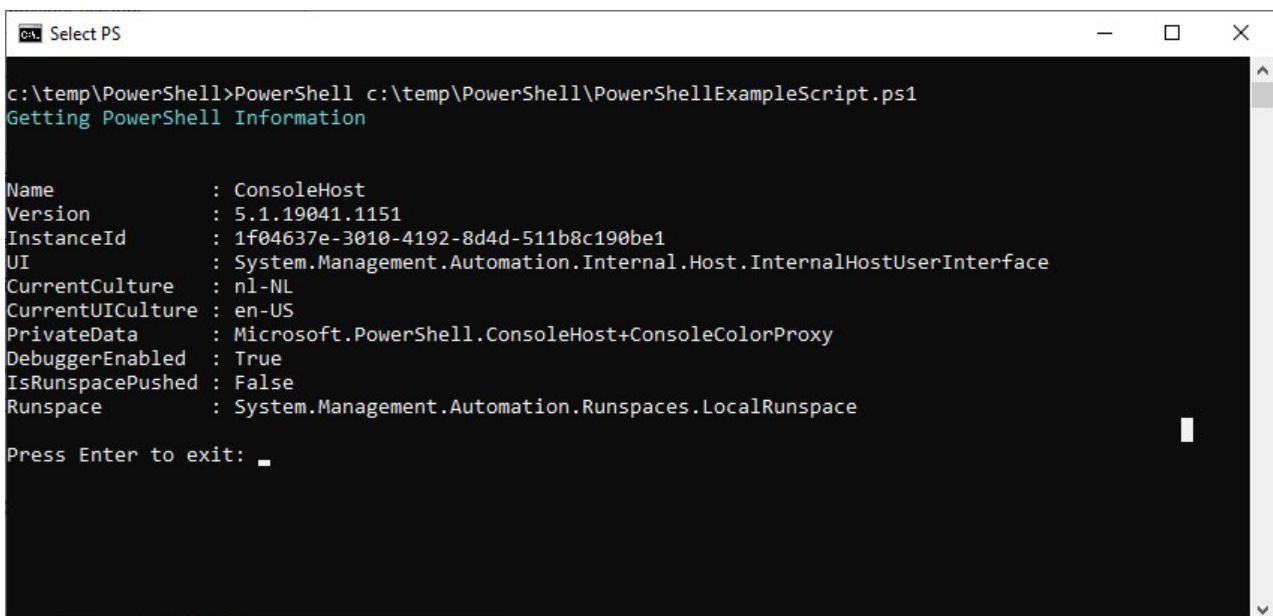
## Run PowerShell Script from CMD

---

If you have tried to run a PowerShell from cmd, then you might have noticed that it will just open the script in notepad. Not really the result we are looking for.

To run the PowerShell script, we need to tell the command prompt to open the script with PowerShell. We can use the following cmd for this:

PowerShell c:\path\to\script\PowerShellScript.ps1



```
c:\temp\PowerShell>PowerShell c:\temp\PowerShell\PowerShellExampleScript.ps1
Getting PowerShell Information

Name           : ConsoleHost
Version        : 5.1.19041.1151
InstanceId     : 1f04637e-3010-4192-8d4d-511b8c190be1
UI             : System.Management.Automation.Internal.Host.InternalHostUserInterface
CurrentCulture : nl-NL
CurrentUICulture : en-US
PrivateData    : Microsoft.PowerShell.ConsoleHost+ConsoleColorProxy
DebuggerEnabled : True
IsRunspacePushed : False
Runspace       : System.Management.Automation.Runspace.LocalRunspace

Press Enter to exit: _
```

Note that you will need to **specify the full path** to the PowerShell script. The result of the script is displayed in the command prompt.

If you want to keep the PowerShell session open, you can add the **-noexit** parameter to the command:

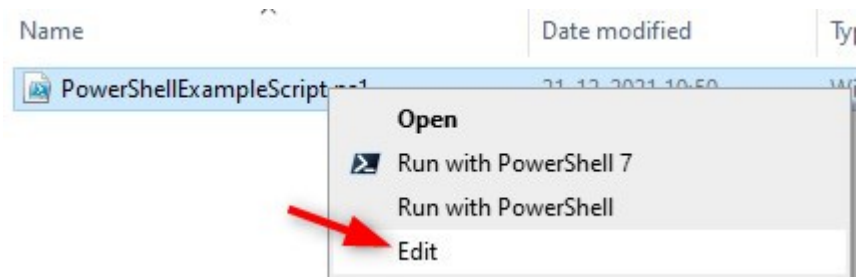
PowerShell -noexit c:\temp\PowerShell\PowerShellExampleScript.ps1

## Using PowerShell ISE

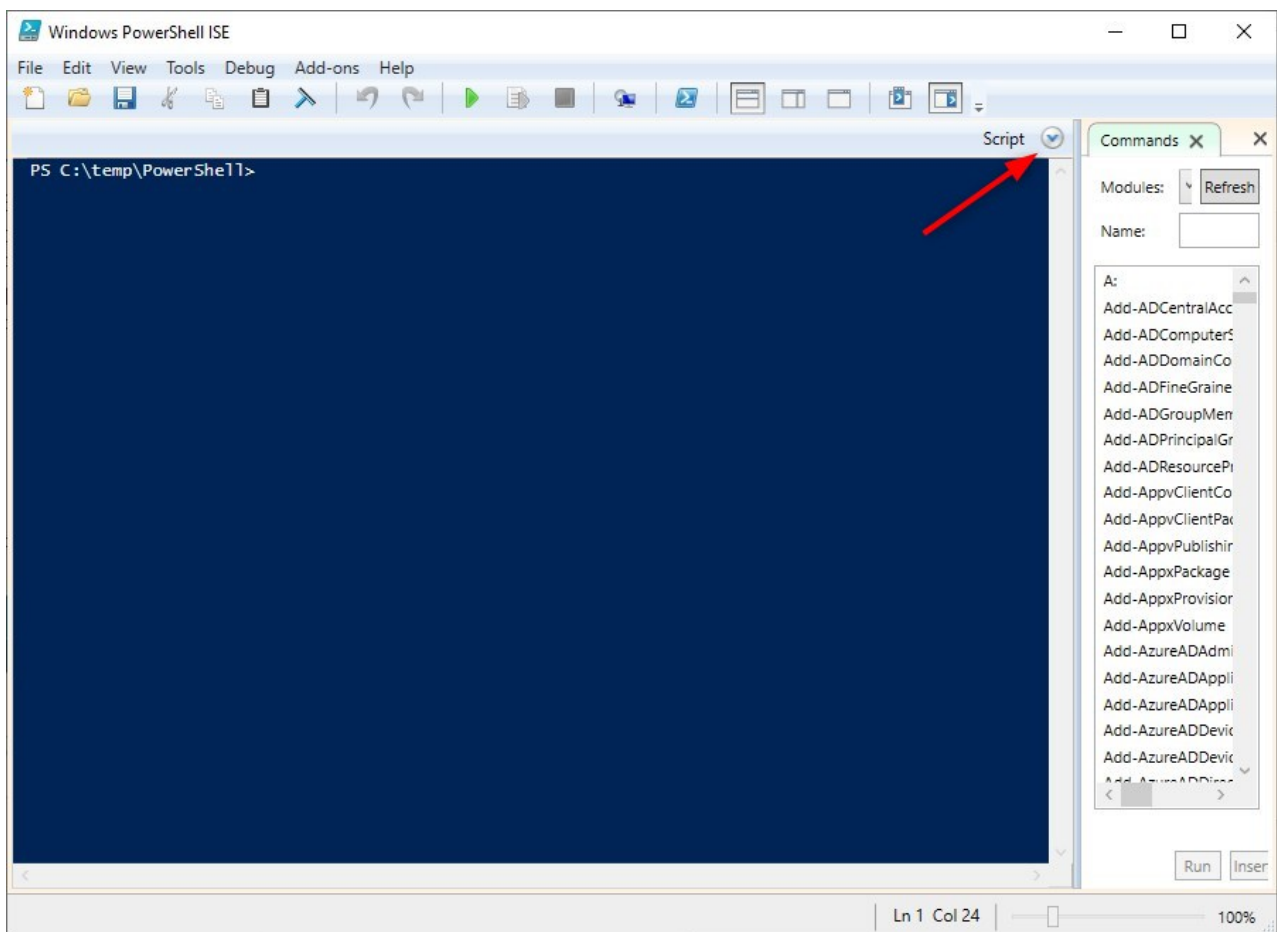
---

When you download a script from the internet it's always a good idea to review the script before you run it. An easy way to review and run a PowerShell script is to use the PowerShell ISE.

To open a PowerShell file in the editor, simply right-click it and choose **Edit**

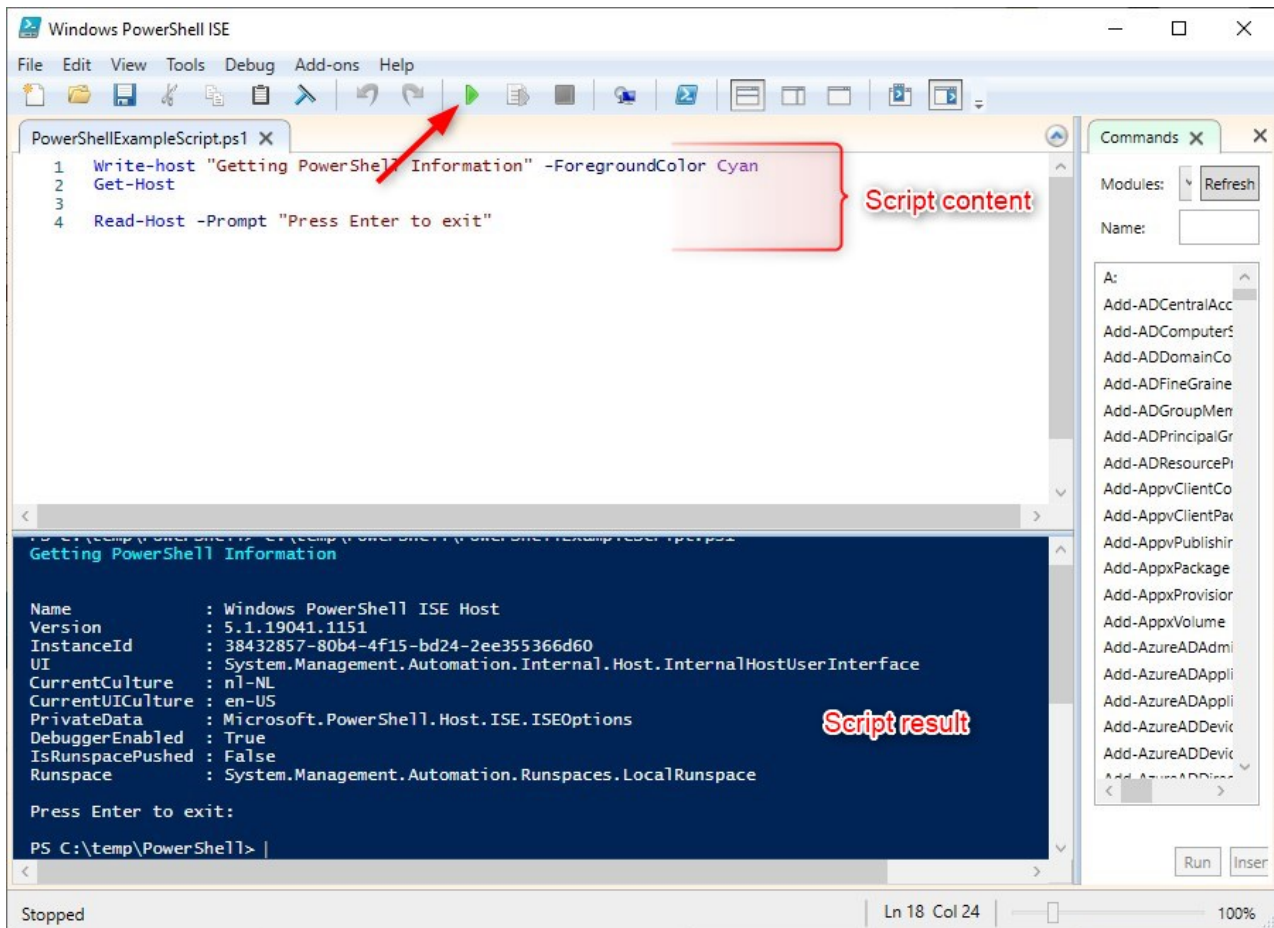


When you use the editor for the first time you might need to expand the script pane first, to see the content of the PowerShell script:



To run a script in PowerShell ISE simply click on the green play button in the toolbar. The results of the scripts are displayed in the console:





## Run PowerShell Script as Administrator

Some PowerShell scripts require administrator privilege to run correctly. They might need access to all folders on your system drive, or need to interact with other domain computers or servers.

The easiest way to run PowerShell scripts as an administrator is to start Windows PowerShell as administrator.

1. Press **Windows key + X** (or right-click the start menu)
2. Choose **Windows PowerShell (admin)**
3. Click **Yes** on the User Account Security prompt

You can now run any PowerShell command or script with Administrator privilege. Simply navigate to the script (or even drag and drop the script) to run it.

You can also run a PowerShell script as administrator from the command line. When you have PowerShell open, we can start a new PowerShell process and run it as Administrator. Optionally we can even specify the file that we want to run:

# Open a new PowerShell windows with Administrator Privilege

```
Start-Process PowerShell -Verb runAs
```

# Run PowerShell script with Administrator Privilege

```
Start-Process PowerShell -Verb runAs -ArgumentList "-file  
c:\temp\powershell\powershellexamplescript.ps1"
```

## Wrapping Up

---

I hope you were able to run your PowerShell script with the steps above. Make sure that you have set the Execution policy correctly before you run a PowerShell script. You can also schedule PowerShell scripts [using the Task Scheduler](#).

If you have any questions, just drop a comment below. Make sure you also read more about [setting up your PowerShell profile](#).

Did you **Liked** this **Article**?

Get the latest articles like this **in your mailbox**

or share this article

I hate spam to, so you can unsubscribe at any time.