

Top 5 PowerShell Modules You Need to Know About

lazyadmin.nl/powershell/top-5-powershell-modules

May 30, 2024

I have been using PowerShell for quite some years now, and over the years I have installed a lot of PowerShell modules. Most are purely functional, to connect to Exchange Online for example, but others are just really handy tools to have.

Even though there are a lot more PowerShell Modules that are worth to install, these are probably the best PowerShell modules that I used the most. If you have some good suggestion, make sure you let me know in the comments.

In this article, I will show you my Top 5 PowerShell modules that you should need to know about.

Oh-My-Posh

We of course have to start with one of the best PowerShell modules, Oh My Posh. This is a custom prompt engine that can be used with any shell and allows you to fully customize your terminal.

The latest version of Oh My Posh works with theme files, allowing you to easily create and share your custom theme. There is an [extensive list](#) of themes available, allowing you to easily pick one and get started.



The screenshot shows a PowerShell terminal window titled 'scripts (pwsh)'. The prompt is a colorful Oh My Posh prompt showing the current directory as '/SysAdmin/scripts', the Git status as 'master', and other system information like time and battery. The user has entered two commands: 'Connect-ExchangeOnline -UserPrincipalName lazyadmin@lazydev.onmicrosoft.com -ShowBanner:\$false' and 'Get-Exomailbox -Identity meganb | ft'. The output of the second command is a table with columns: ExternalDirectoryObjectId, UserPrincipalName, Alias, DisplayName, and EmailAddresses. The table contains one row of data for Megan Bowen.

ExternalDirectoryObjectId	UserPrincipalName	Alias	DisplayName	EmailAddresses
0ee2adb2-54a6-4196-9f27-9c3ec0ace47a	MeganB@lazydev.onmicrosoft.com	MeganB	Megan Bowen	{SPO:SPO_8f784c95-0845-4e58-90...

You can not only add colors to your terminal but also useful information, like the Git status, user information, or system information.

To get started with Oh My Posh, you can install it using the following command:

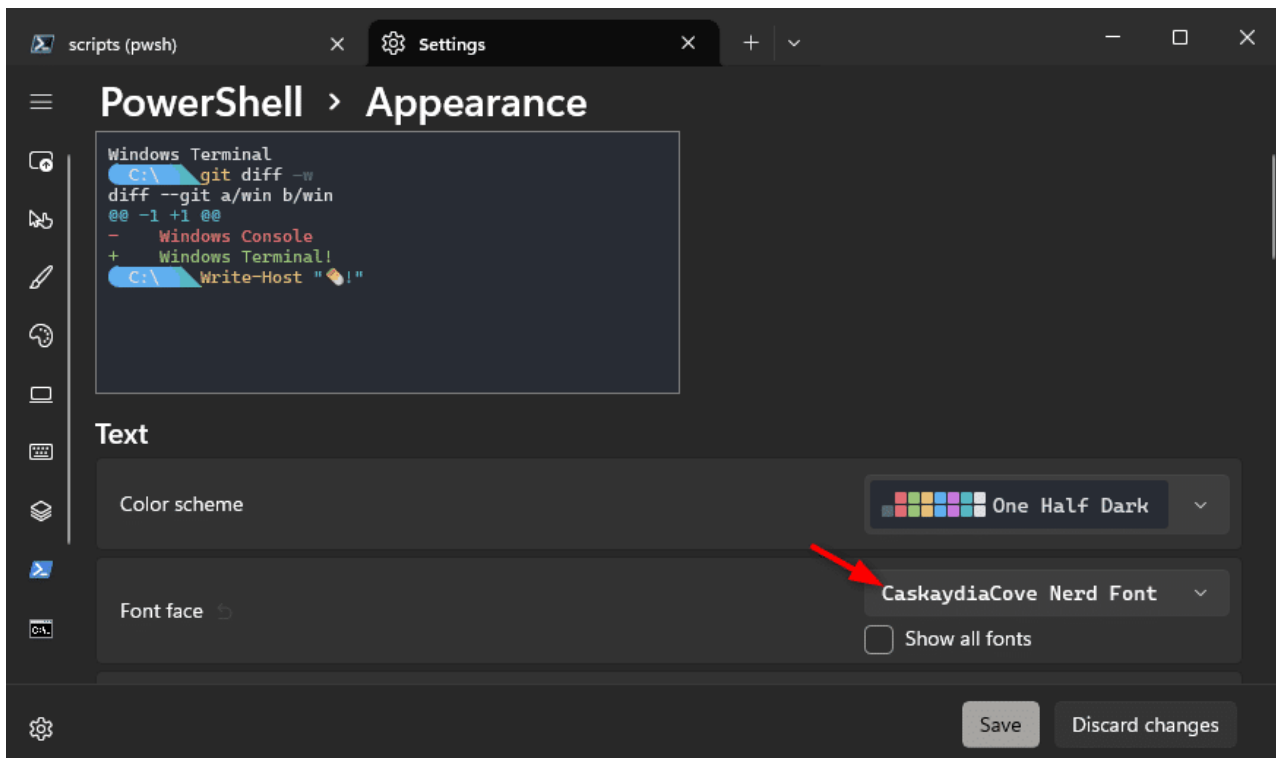
```
winget install JanDeDobbeleer.OhMyPosh -s winget
```

To fully use all the capabilities of Oh My Post, you will need to install a Nerd Font as well. Nerd Fonts are popular fonts where icons are added too. Installing one of the Nerd Fonts is pretty easy, simply type:

```
oh-my-posh font install
```

You can select one of the fonts to install. The recommended font is [Meslo LGM NF](#), but in the screenshot above I have used CaskaydiaCove Nerd Font. After you have installed the font, you will need to set it as your default font for your PowerShell sessions.

- Open [Windows Terminal](#)
- Click on the dropdown arrow and choose **Settings**
- Select **PowerShell** in the sidebar, and open **Appearance** (all the way at the bottom)
- Set the **Font Face** to your installed Nerd Font



With the fonts installed, we can select one of the Themes to use or as a starting point to customize. To view all the Themes, you can use the cmdlet `Get-PoshThemes`. This will show examples of all the themes in your console.

To install the theme, I recommend saving the JSON file locally. You can then enable the theme with:

```
oh-my-posh init pwsh --config 'd:\scripts\lazyadmin\lazyadmin.omp.json' | Invoke-Expression
```

Terminal Icons

The Terminal Icons PowerShell modules really accomplish the Oh My Post themes. Terminal Icons will add files and folder icons (and colors) in your shell. This really helps with quickly finding the right file or folder.



Terminal Icons uses, just like Oh My Posh, a Nerd Font for all the icons. If you already have installed Oh My Posh, then you can simply install the Terminal Icons module with:

```
# Install the module
```

```
Install-Module -Name Terminal-Icons -Repository PSGallery
```

```
# Load the module - add this to your PowerShell Profile:
```

```
Import-Module -Name Terminal-Icons
```

If you are not using Oh My Posh, then make sure that you install a Nerd Font as well. You can checkout [this guide](#) for more information on how to get it to work in PowerShell.

ImportExcel – Import and Export Excel

When you have used PowerShell for quite sometime then you probably have created an report more then once. Most will use the `Export-CSV` cmdlet to export data from PowerShell and use in Excel. But there is an easier way to do this.

The `ImportExcel` module does not only allow you to import data from an Excel file, but it can also export to an Excel, with a fully functional table init as well. And it doesn't stop there, charts, pivot tables, conditional formatting, is all possible.

	A	B	C	D	
1	ExternalDirectoryObjectId	UserPrincipalName	Alias	DisplayName	EmailAddresses
2	7a3b301d-0462-41b6-8468-19a38:	AdeleV@lazydev.onmicrosoft.com	AdeleV	Adele Vance	SMTP:AdeleV@lazydev.onmicrosof
3	4d72ae58-1792-4aeb-a776-1d432	beamert@lazydev.onmicrosoft.com	beamert	beamer	SMTP:beamert@lazydev.onmicrosc
4	88692ac9-42ad-417d-8b15-03c35c	boardroom@lazydev.onmicrosoft.com	boardroom	boardroom	SMTP:boardroom@lazydev.onmicr
5	eb30c2a0-0d58-4c0e-a95b-bf12e	branch-office@lazydev.onmicrosoft.com	branch-office	Branch Office	SMTP:branch-office@lazydev.onmi
6	8563c068-2c1b-47b6-9d15-debac	DiegoS@lazydev.onmicrosoft.com	DiegoS	Diego Siciliani	SPO:SPO_5ffeacc7-a863-4098-9d39-
7		DiscoverySearchMailbox{D919BA05-46A6-4	DiscoverySearchM	Discovery Search Mailbox	SMTP:DiscoverySearchMailbox{D91
8	220406ad-5e79-4750-a560-e399d	GradyA@lazydev.onmicrosoft.com	GradyA	Grady Archie	SPO:SPO_7f167796-829f-4aa0-8f3a-
9	5a207d6e-9750-4829-9abe-0cc72	HenriettaM@lazydev.onmicrosoft.com	HenriettaM	Henrietta Mueller	SPO:SPO_0603014f-864b-4a58-ad5b
10	4747e48d-e573-45d3-8bb6-67ec0	IsaiahL@lazydev.onmicrosoft.com	IsaiahL	Isaiah Langer	SIP:IsaiahL@lazydev.onmicrosoft.c
11	343dd6f0-bb31-4b64-b7e4-89217	JohannaL@lazydev.onmicrosoft.com	JohannaL	Johanna Lorenz	SPO:SPO_70acea42-6194-4e0a-be9
12	38157e3d-16a3-4086-9c40-99c22b	JoniS@lazydev.onmicrosoft.com	JoniS	Joni Sherman	SPO:SPO_08066db4-2381-44c9-8116
13	1a711acf-5ea5-4b49-b8bb-061fc8	lazyadmin@lazydev.onmicrosoft.com	lazyadmin	Rudy Mens	smtp:lab23@lazydev.onmicrosoft.c
14	0593a29f-ae6a-4a6f-8683-ee11f8	LeeG@lazydev.onmicrosoft.com	LeeG	Lee Gu	SIP:LeeG@lazydev.onmicrosoft.cor
15	0ee2adb2-54a6-4196-9f27-9c3ec	MeganB@lazydev.onmicrosoft.com	MeganB	Megan Bowen	SPO:SPO_8f784c95-0845-4e58-9024-
16	27e74ef7-f754-46aa-89dc-5aa754	MiriamG@lazydev.onmicrosoft.com	MiriamG	Miriam Graham	SPO:SPO_452dc8ba-86b8-47c4-8a7c
17	e1512872-9f54-4742-b34a-602187	PattiF@lazydev.onmicrosoft.com	PattiF	Patti Fernandez	SIP:PattiF@lazydev.onmicrosoft.co
18	f17d1aec-6407-436f-90c2-b4b6b0	PradeepG@lazydev.onmicrosoft.com	PradeepG	Pradeep Gupta	SIP:PradeepG@lazydev.onmicrosof
19	c8a7c469-caac-4668-80f9-e536fdc	office@lazydev.onmicrosoft.com	office	The Office	SMTP:office@lazydev.onmicrosof.
20					

To quickly export your data to Excel and format it into a table, you can use the following command:

```
# First install the module
Install-Module -Name ImportExcel
# Get some data and export it to Excel with a formatted table
Get-EXOMailbox | Export-Excel -AutoSize -BoldTopRow -FreezeTopRow
```

PSWriteHtml – Export to HTML

Whether you are using PowerShell to automate tasks or to create reports, you often need to communicate the results. One way to do this is by sending an email from PowerShell with the results.

But a simple, plain text email isn't going to do it anymore. We want it nicely formatted, maybe a table in it, or some colors. That is where HTML comes in. Now you can write and format your own HTML, but the PowerShell module PSWriteHTML makes it a lot easier for you.

Another great feature of PSWriteHTML is that you can quickly export to results to an HTML file. So instead of Out-GridView, you can now use Out-HtmlView, which gives you an interactive HTML file. From the HTML view you can quickly export it again to CSV or PDF, or just share the HTML view if needed.

Copy Excel CSV PDF Show 15 rows Search Builder

Search:

ExternalDirectoryObjectId	UserPrincipalName	Alias	DisplayName	EmailAddresses
▶ 7a3b301d-0462-41b6-8468-19a3837b8ad1	AdeleV@lazydev.onmicrosoft.com	AdeleV	Adele Vance	SMTP:AdeleV@lazydev.onmicrosoft.com SIP:adelev@lazydev.onmicrosoft.com SPO:SPO_ac96b1fb-c38c-4728-a276-b44596863962@SPO_11e55098-68ad-4992-aaf8-c5fdceb3b6da
▶ 4d72ae58-1792-4aeb-a776-1d43226f5498	beamert@lazydev.onmicrosoft.com	beamert	beamer	SMTP:beamert@lazydev.onmicrosoft.com
▶ 88692ac9-42ad-417d-8b15-03c35d98f549	boardroom@lazydev.onmicrosoft.com	boardroom	boardroom	SMTP:boardroom@lazydev.onmicrosoft.com
▶ eb30c2a0-0d58-4c0e-a95b-bf12e4001df1	branch-office@lazydev.onmicrosoft.com	branch-office	Branch Office	SMTP:branch-office@lazydev.onmicrosoft.com
▶ 8563c068-2c1b-47b6-9d15-debacd005268	DiegoS@lazydev.onmicrosoft.com	DiegoS	Diego Siciliani	SPO:SPO_5ffeacc7-a863-4098-9d39-ee2ce3652691@SPO_11e55098-68ad-4992-aaf8-c5fdceb3b6da SIP:diegos@lazydev.onmicrosoft.com SMTP:DiegoS@lazydev.onmicrosoft.com
▶	DiscoverySearchMailbox(D919BA05-46A6-415F-80AD-7E09334BB852)@lazydev.onmicrosoft.com	DiscoverySearchMailbox(D919BA05-46A6-415F-80AD-7E09334BB852)	Discovery Search Mailbox	SMTP:DiscoverySearchMailbox(D919BA05-46A6-415F-80AD-7E09334BB852)@lazydev.onmicrosoft.com
▶ 220406ad-5e79-4750-a560-e399df31433e	GradyA@lazydev.onmicrosoft.com	GradyA	Grady Archie	SPO:SPO_7f167796-829f-4aa0-8f3a-4933aad079d5@SPO_11e55098-68ad-4992-aaf8-c5fdceb3b6da SIP:GradyA@lazydev.onmicrosoft.com SMTP:GradyA@lazydev.onmicrosoft.com
▶ 5a207d6e-9750-4829-9abe-0cc72ed40252	HenriettaM@lazydev.onmicrosoft.com	HenriettaM	Henrietta Mueller	SPO:SPO_0603014f-864b-4a58-ad5b-77d709a621ef@SPO_11e55098-68ad-4992-aaf8-c5fdceb3b6da SIP:HenriettaM@lazydev.onmicrosoft.com SMTP:HenriettaM@lazydev.onmicrosoft.com
▶ 4747e48d-e573-45d3-8bb6-67ec0dea617b	IsaiahL@lazydev.onmicrosoft.com	IsaiahL	Isaiah Langer	SIP:IsaiahL@lazydev.onmicrosoft.com SPO:SPO_40bc66f6-4761-47a8-9f55-719229866bc6@SPO_11e55098-68ad-4992-aaf8-c5fdceb3b6da SMTP:IsaiahL@lazydev.onmicrosoft.com

To quickly generate a view like the example above, you can use the following PowerShell code:

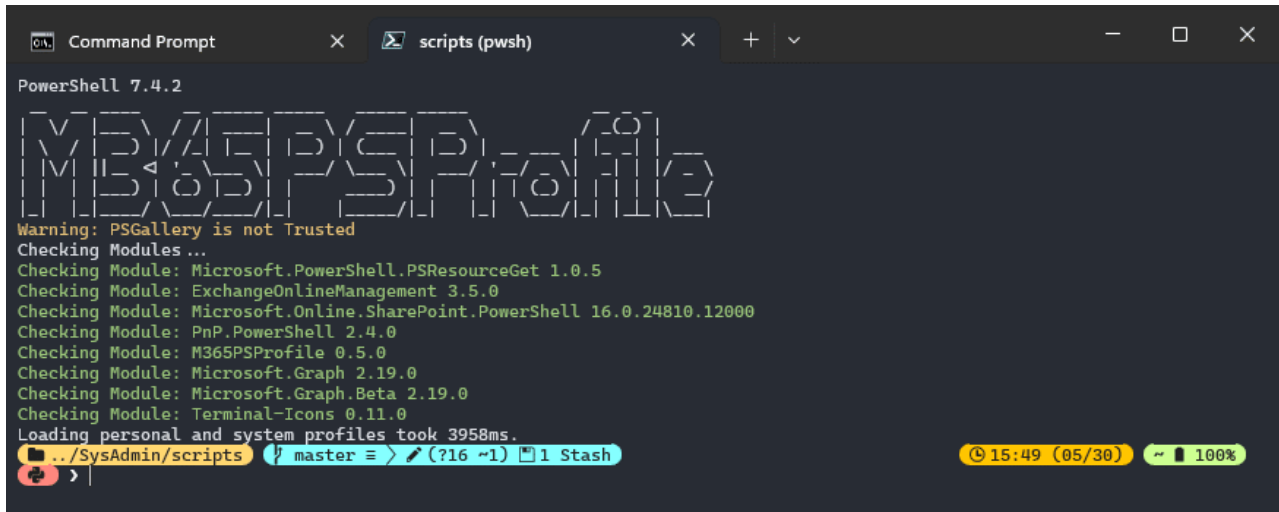
```
# Install the module
Install-Module -Name PSWriteHTML
# Gather some data and output to HTML
Get-Exomailbox | Out-HtmlView
```

M365PSProfile – Keeping your Modules up-to-date

The last module that I wanted to share with your is the module M365PSProfile. To goal of this module is to simplify the installation and updating of all the Microsoft 365 related PowerShell modules.

Although the intention of the module is to only manage the Microsoft 365 modules, you can also use it to automatically install and keep other modules up to date.

The module will check all the listed modules when you open PowerShell and make sure that you have the latest version installed.



To get started with the module, you will of course need to install it first. When done, you can add a list of modules in your PowerShell Profile that you want to keep up-to-date:

Install the module

```
Install-Module -Name M365PSProfile -Scope CurrentUser
```

#Install or Updates the Modules in the Array

```
Install-M365Module -Modules
```

```
@("ExchangeOnlineManagement","Microsoft.Online.SharePoint.PowerShell","PnP.PowerShell",  
"M365PSProfile","Microsoft.Graph","Microsoft.Graph.Beta","Terminal-Icons")
```

Wrapping Up

PowerShell modules are a great way to make your work a lot easier or working with PowerShell a lot nicer to look at. There are of course a lot more great PowerShell modules, these are only some of the best module in my opinion.

If you have a great PowerShell module that you highly recommend, then please let me know in the comments below!

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.