# ReadMe 'modular.PowerShell_Profile'

## English

The modular.PowerShell_Profile module splits
the PowerShell profile into
several logically separated sections.

This makes the profile easier to read,
easier to maintain, and more flexible to extend.

All settings and functions can be enabled or disabled
individually by commenting out a line with '#'
at the beginning of the line.

The hash symbol marks the **Command** or setting as a comment,
preventing it from being loaded into PowerShell.

### My motivation:

Originally, I only wanted to:
- display the last 50 commands at startup, and
- adjust a few PSReadLine parameters to match my preferences.

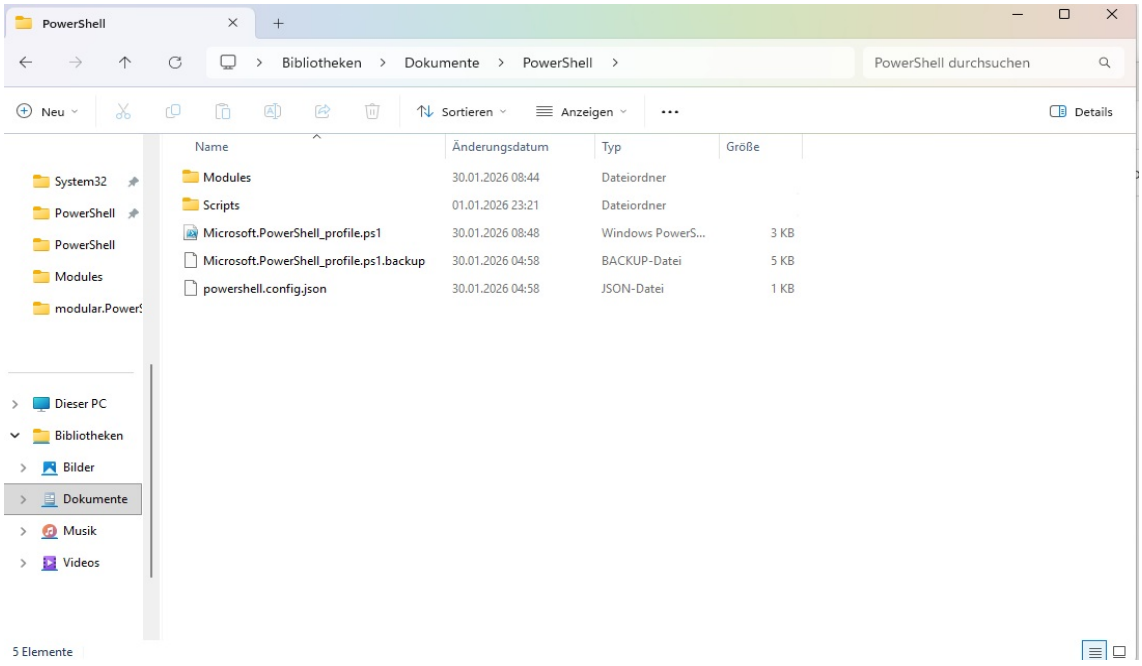
From this, the idea emerged to build the entire **profile**
in a modular way and load only the modules
that are actually needed.

Instead of a long list of fixed imports,
the module now automatically loads:
**whichever module provides the command currently being used.**

For transparency, i will upload the original profile file
as '**Microsoft.PowerShell_profile.old.ps1**'

### Structure of the module:

The **profile** consists of one main file



And then there are the four subcategories into which the **profile** is divided:

"**UI.psm1**"
This file allows you to adjust the appearance and,
to some extent, the behavior of **PowerShell**.

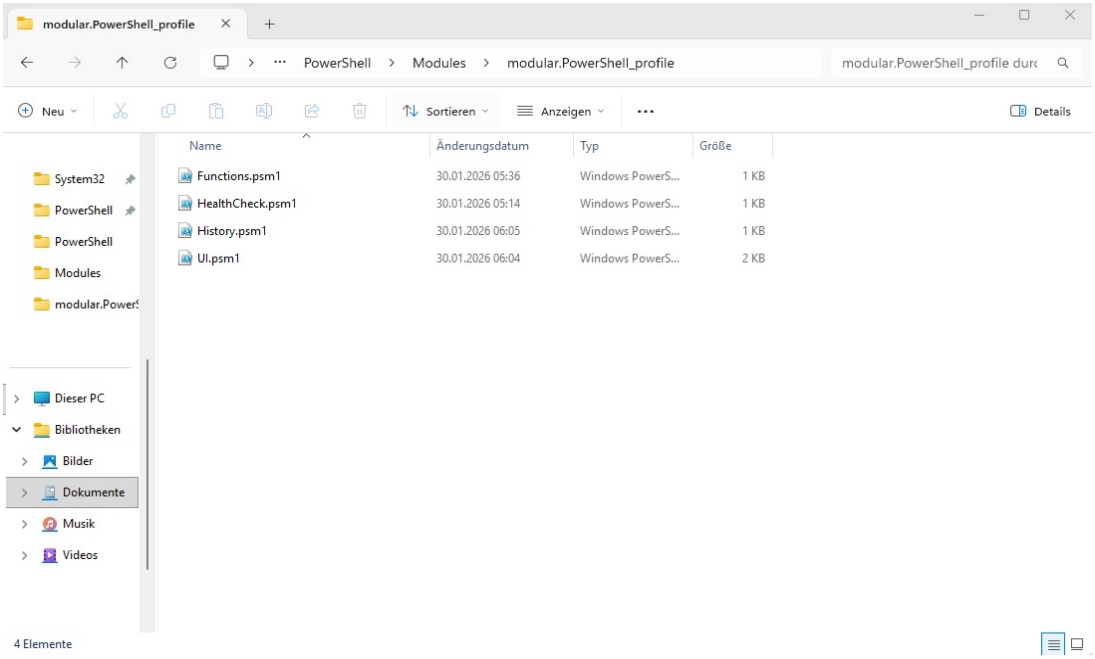
"**History.psm1**"
This file is responsible for loading the **PSReadLine** history file
(which contains the complete command history – provided saving is enabled)
in every session.
**(AppData\Roaming\Microsoft\Windows\PowerShell\PSReadLine**
**--> ConsoleHost_history.txt)**

To change the displayed language (only one line), modify the text in:
**Write-Host "`nLast 50 Commands:"**
(Important: the part **Write-Host "`n** must remain unchanged,
as well as the two characters :" at the end, otherwise errors will occur.)

At the current state of the module, this is the only text that may need
translation.

"**Functions.psm1**"
The core of the module
defines and manages the functions and their parameters.

"**HealthCheck.psm1**"
Performs a quick system check:
- Internet connection
- PSReadLine installed?
- WinGet and Git available?
- Profile writable and correctly set?




If **Git** is not used or should not be used, comment out
the line containing "**Git**" in **HealthCheck.psm1** or remove it.

Additionally, in **Microsoft.PowerShell_profile.ps1**,
comment out or remove the **import posh-git** line.

For the **PowerToys CommandNotFound** feature,
proceed the same way if you do not want to use it (**PowerToys required**).

This step is important, because **PowerShell** will otherwise attempt
to load scripts or modules that do not exist, which will cause errors.

## Advantages of the modular structure:

- much clearer than a single large profile file
- easy to extend (simply create a new .psm1 file and include it)
- only modules that are actually needed are loaded
- ideal for beginners, as changes can be applied very easily
- safer, because missing or faulty modules are not loaded automatically

## Important

Since the profile file will be overwritten, always create a backup first,
or rename the **Microsoft.PowerShell_profile.ps1** file and append **.backup**.

**Example**:
(For **PowerShell** 7.x and higher)
**C:\Users\%UserName%\Documents\PowerShell\Microsoft.PowerShell_profile.ps1**
**--> Microsoft.PowerShell_profile.ps1.backup**

(For **WindowsPowerShell**)
**C:\Users\%UserName%\Documents\WindowsPowerShell\Microsoft.PowerShell_profile.ps1**
**--> Microsoft.PowerShell_profile.ps1.backup**

## Installation:

Copy all module files into your **PowerShell** documents folder
(all files from the extracted ZIP):

(For **PowerShell** 7.x)
**C:\Users\%UserName%\Documents\PowerShell**

(For **WindowsPowerShell** -not recommended-)
**C:\Users\%UserName%\Documents\WindowsPowerShell**

The downloaded main file also contains the “Modules” folder.
Simply copy it into your **Documents\PowerShell** folder.
The included **PowerShell** profile (**Microsoft.PowerShell_profile.ps1**)
will replace your existing one.

The folder structure is designed so it can be copied directly:

## Detailed folder structure:

**C:\Users\%UserName%\Documents\WindowsPowerShell** (The profile file must be replaced here)
-->**Modules**
-->**Module Name**
-->**Module Version** (This contains the module files)

**Recommendation**
Use the module in only one of the two **PowerShell** versions.
If something breaks for any reason,
you will still have a working backup – or as Microsoft calls it, a fallback.

This **$Profile** module is not only easier to edit
due to its fully modular structure, but also offers the advantage
that if new functions are added, or if you want to add
functions and settings that do not fit into any of the existing categories,
you can simply create a new .psm1 file and add it to the list in
**Microsoft.PowerShell_profile.ps1**.

This module is also easily expandable, which is what made it
so interesting for me to script it in a modular and freely extendable way.

## Community

**Do you have suggestions or ideas for new functions?**

Feel free to share them :)
Seriously – please share your ideas with me.
Maybe this project can grow far beyond its original goals.

Everyone is free to modify or extend the included files.
I only ask that if you add new functions, please share your creativity with me and the community.
A “UserMod” folder may be added to the project where users can upload their versions after a brief review.

This allows for a wide variety of configurations and adjustments (for easy “just use it without thinking too much”).
Depending on community feedback, these additions may also be integrated into the base version.

## License