

ReadMe 'modular.PowerShell_Profile'

Deutsch

Das `modular.PowerShell_Profile`-Modul teilt die `PowerShell`-Profildatei, in mehrere logisch getrennte Bereiche auf.

Dadurch wird das Profil übersichtlicher, leichter zu bearbeiten und besser erweiterbar. Alle Einstellungen und Funktionen können einzeln aktiviert oder deaktiviert werden, indem an erster Stelle einer Zeile mit `#` diese zum Kommentar gesetzt wird.

Meine Motivation:

Ursprünglich wollte ich nur:

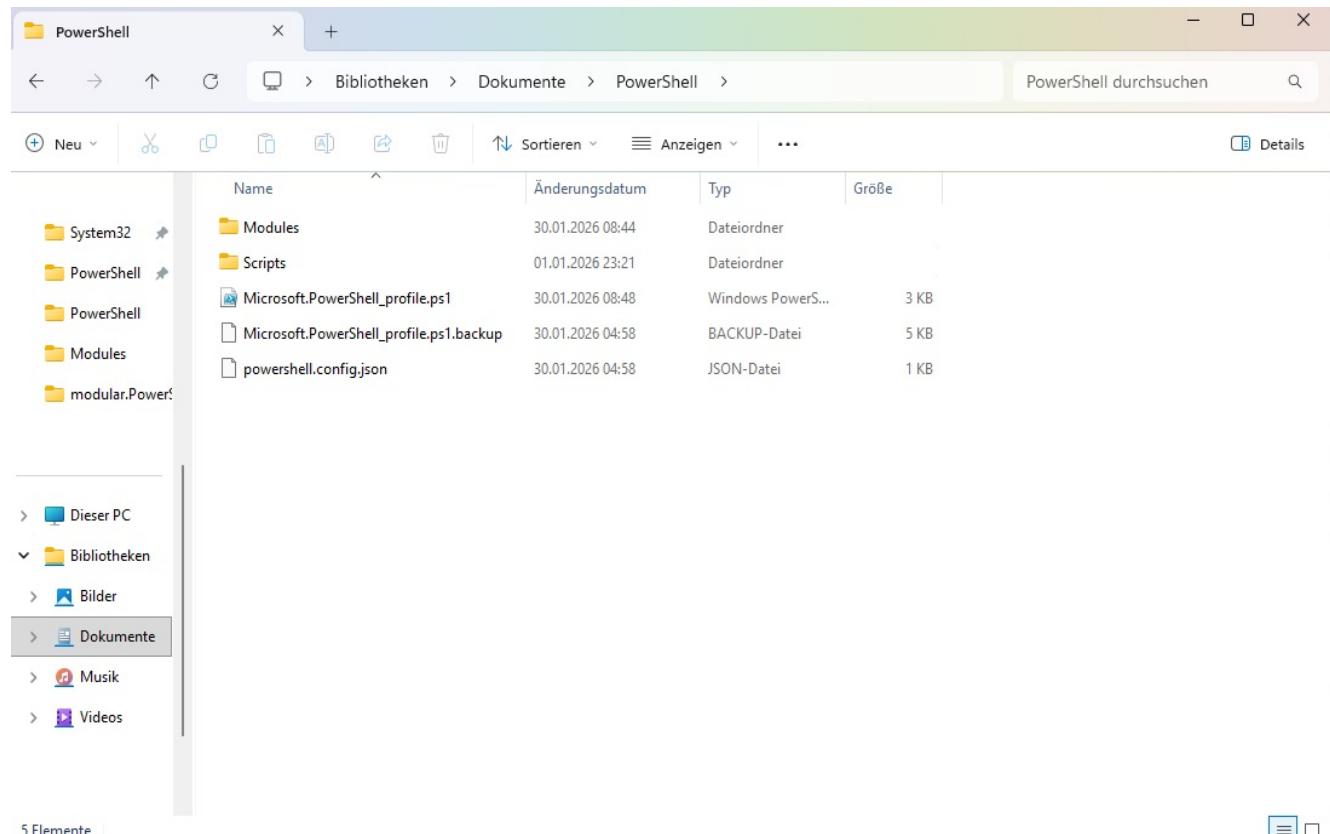
- die letzten 50 Befehle beim Start anzeigen und
- einige `PSReadLine`-Parameter nach meinen Vorstellungen setzen.

Daraus entstand die Idee, das gesamte `Profil` modular aufzubauen und nur die Module zu laden, die tatsächlich benötigt werden. Statt einer langen Liste fest importierter Module, wird nun automatisch das Modul geladen, das den verwendeten Befehl bereitstellt.

Die ursprüngliche Profildatei lade ich als `'Microsoft.PowerShell_profile.old.ps1'` hoch - zur Transparenz.

Aufbau des Moduls:

Das Profil besteht aus einer Hauptdatei



Und die 4 Unterkategorien in die sich das **Profil** dann aufteilt:

"UI.psml"

Hier lässt sich das Aussehen und zum Teil auch das dazu gehörende Verhalten der PowerShell anpassen.

"History.psml"

Lädt die **PSReadLine**-Historydatei

(**AppData\Roaming\Microsoft\Windows\PowerShell\PSReadLine** --> **ConsoleHost_history.txt**)

Um die angezeigte Sprache zu ändern (ist nur eine Zeile)

muss nur der Inhalt bei:

Write-Host "`nLast 50 Commands:"

angepasst werden.

(Wichtig dieser Teil: **Write-Host** "`n muss unverändert bleiben,

genauso wie am Ende die beiden Zeichen :" da es sonst zu Fehlern kommen wird.)

Nach jetzigem Stand der Modulfunktionen,

ist dies aber das einzige an Text was man übersetzen müsste bzw. könnte.

"Functions.psml"

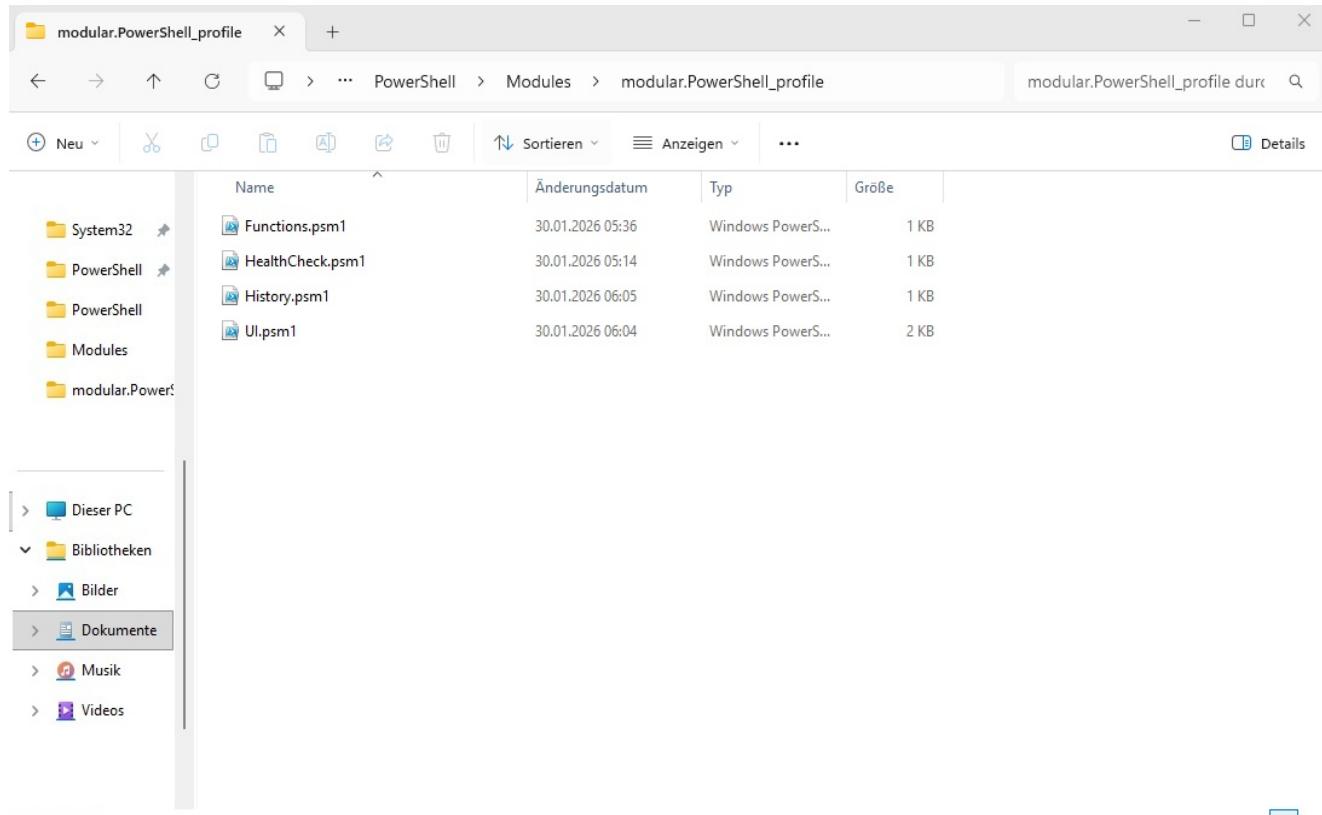
Das eigentliche Kernstück des Moduls,

hier werden die Funktionen und entsprechende Parameter festgelegt und gesteuert

"HealthCheck.psml"

Kurzer Systemcheck:

- Internetverbindung
- PSReadLine installiert?
- WinGet und Git vorhanden?
- Profil beschreibbar und korrekt gesetzt?



4 Elemente

Wenn **Git** nicht genutzt wird, oder nicht verwendet werden soll,

In **HealthCheck.ps1** in der Zeile die '**Git**' enthält an erster Stelle,
also ganz vorne eine Raute '#' setzen oder die Zeile entfernen.

Zusätzlich in der **Microsoft.PowerShell_profile.ps1**

vor **import 'posh-Git'** ebenfalls die Raute setzen oder die Zeile löschen.

Bei der **PowerToys** Funktion **CommandNotFound** sollte ebenfalls so vorgegangen werden,
wenn diese Funktion nicht genutzt werden soll (setzt **PowerToys** voraus)

Da es zwangsläufig zu Fehlern kommen wird, wenn die **PowerShell** versucht
ein Skript, Modul oder eine Funktion zu laden die nicht existiert,
ist dieser Schritt sehr wichtig.

Vorteile des modularen Aufbaus

- deutlich übersichtlicher als eine große Profildatei
- leichter zu erweitern (einfach neue `psm1`-Datei erstellen und einbinden)
- nur tatsächlich benötigte Module werden geladen
- ideal für Einsteiger, da Änderungen sehr einfach übernommen werden können
- sicherer, da fehlerhafte oder fehlende Module nicht automatisch geladen werden

Wichtig

Da die **Profil**datei überschrieben wird, vorher unbedingt ein Backup anlegen, oder die Datei **Microsoft.PowerShell_profile.ps1** umbenennen und ein **.backup** dransetzen.

Beispiel:

(Für **PowerShell** 7.x und höher)

```
C:\Users\%UserName%\Documents\PowerShell\Microsoft.PowerShell_profile.ps1  
--> Microsoft.PowerShell_profile.ps1.backup
```

(Für **WindowsPowerShell**)

```
C:\Users\%UserName%\Documents\WindowsPowerShell\Microsoft.PowerShell_profile.ps1  
--> Microsoft.PowerShell_profile.ps1.backup
```

Installation:

Modul Dateien in den '**PowerShell**' Dokumente-Ordner kopieren
(Alle Dateien des entpackten Zip-Archivs):

(Für die **PowerShell** 7.x)

```
C:\Users\%UserName%\Documents\PowerShell\
```

(Für die **WindowsPowerShell** -nicht empfohlen-)

```
C:\Users\%UserName%\Documents\WindowsPowerShell\
```

Die heruntergeladene Hauptdatei enthält auch den Ordner „**Modules**“.

(Kopieren Sie ihn einfach in den Ordner „**Dokumente\PowerShell**“.)

und das modifizierte **PowerShell \$Profile**

(**Microsoft.PowerShell_profile.ps1**). Diese wird durch das Modul ersetzt.

Die Ordnerstruktur ist so aufgebaut, dass sie direkt übernommen werden kann:

Detaillierte Ordnerstruktur:

```
C:\Users\%UserName%\Documents\PowerShell\ (---Hier die Microsoft.PowerShell_profile.ps1 ersetzen)  
-->Modules  
-->Modulname  
-->Modulversion (Dieser enthält die Moduldateien)
```

Meine Empfehlung ist,

nutzt das Modul nur bei einem von beiden **PowerShell** Versionen.

So dass ihr im Falle das, warum auch immer,

die **PowerShell** funktionsunfähig wird, immer noch ein verwendbares Backup,
oder wie Microsoft sagt, FallBack habt.

Habt ihr Verbesserungsvorschläge oder Ideen für neue Funktionen?

Dann, immer her damit :)

teilt bitte eure Ideen mit mir.

Vielleicht kann dann auch dieses Projekt,
weit über seine eigentlichen Ziele und Aufgaben hinaus wachsen,
größer werden...

Es steht jedem frei die enthaltenen Dateien/Files,
nach eigenem Wunsch anzupassen oder zu erweitern.

Ich bitte nur darum, wenn ihr dieses Modul um Funktionen ergänzt habt,
teilt dann eure Kreativität bitte mit mir und der Community.

Es ist auch möglich ein UserMod Ordner dem Projekt hinzuzufügen,
wo jeder nach kurzer Prüfung auf schädlichem Inhalt,
seine Version hochladen kann.

Als Community Content,
für größtmögliche Vielfalt der Anpassungen und Konfigurationen
(zum leichten "einfach übernehmen ohne groß nachdenken zu müssen")

oder dem Modul integriert, um die Basisversion zu verbessern.

Je nachdem, was mehr Stimmen bekommt ;)

Falls daran Interesse besteht

wichtig zu beachten ist nur, dieses Modul ist und bleibt **Freeware**
und darf weder zum Kauf angeboten werden,
noch anderen zu diesem Zwecke verfügbar gemacht werden.
Dies beinhaltet auch das integrieren dieses Moduls in andere Software,
die mit der Absicht des Verkaufs entwickelt wird.

Details: siehe Licence.txt