

Supercharge Your PowerShell - Harnessing ChatGPT and GitHub Copilot for Advanced Script Authoring

Frank Lesniak

X: @franklesniak

Danny Stutz

X: @danny_stutz

April 7-10, 2025

1 / 33



```
C:\Users\flesniak>whoami  
Frank Lesniak (@franklesniak)  
Tech enthusiast; Microsoft 365 consulting team lead at West Monroe
```

Experience:

Almost 20 years in consulting. PowerShellin' since 2006. Focused on corporate M&A.

**Credentials:**

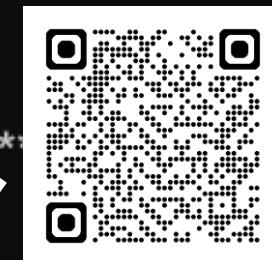
MCSEs. Co-leads Chicago PowerShell Users Group. PowerShell + DevOps Global Summit speaker alumnus.

Ask Me About:

- PowerShell automation (duh!)
- Executing corporate divestitures and integrations – where I spent most of my consulting client time
- File server modernization – moving to Azure Files, Teams/SharePoint, etc.
- Retro video game software and FPGA emulation
- My upcoming home construction project

Contact:

- x.com/franklesniak
- bsky.app/profile/franklesniak.com
- linkedin.com/in/flesniak
- github.com/franklesniak
- infosec.exchange/@franklesniak
- flesniak ATSIGN westmonroe.com



```
C:\Users\flesniak>
```

```
C:\Users\dstutz>whoami  
Danny Stutz (@danny_stutz)  
Computer nerd, M365 consulting and automation leader at West Monroe
```

**Experience:**

6 years working as a M365/Entra/Azure/AWS technical lead

Credentials:

West Monroe PowerShell Interest Group lead.

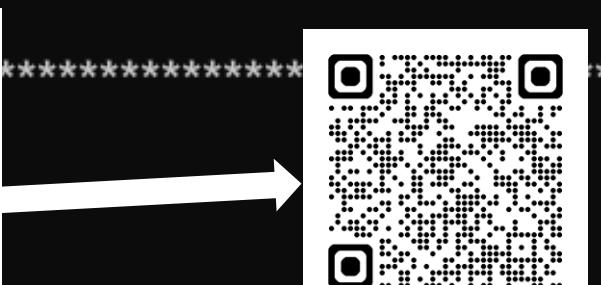
PowerShell + DevOps Global Summit alum

Ask Me About:

- My mechanical keyboard! (Keychron K2, Boba U4's)
- Migration tool automation with PowerShell (ShareGate, MigrationWiz, azcopy, etc...)
- Threat hunting, M365/Entra/Azure/AWS security
- Executing corporate divestiture & integration migration work (where I spend the majority of my consulting time)
- My music taste (send me your favorite tunes on Discord or Slack!)

Contact:

- x.com/danny_stutz
- linkedin.com/in/daniel-stutz
- github.com/danstutz
- dstutz ATSIGN westmonroe.com





DevIT:>_
[DevITJobs.com]



Thank you Sponsors

Key Learning Objectives

- Gain the ability to seamlessly integrate ChatGPT and GitHub Copilot into your PowerShell workflows, enhancing scripting efficiency, accelerating development, and minimizing errors.
- Understand practical prompting tips for generating, debugging PowerShell scripts through live demos
- Master advanced best practices for optimizing and maintaining AI-enhanced PowerShell scripts, including strategies for effective error handling, performance tuning, and ensuring script scalability and reliability.



Agenda

- LLM models and tool comparisons (ChatGPT, Deepseek, etc.)
- Setting up LLM tools in your local development environment
- Generating new scripts
- Script performance optimization techniques
 - Decreasing execution runtime
 - Usability
 - Error handling



LLMs Everywhere

- Cloud based LLMs
 - ChatGPT (OpenAI)
 - Deepseek
 - Gemini 2.5 (Google)
 - Llama 4 (Meta)
 - Grok 3
 - Llama 3.1 Nemotron Ultra (Nvidia)
- Local LLM Software
 - Ollama
 - Microsoft AI Toolkit



Local Dev Tools

- Ollama
- CodeGPT VS Code Extension
- GitHub Copilot



Ollama

- What is Ollama?
 - Ollama is an open-source platform that allows you to run LLMs on your local machine (now on Windows!) to enhance data privacy and reduce reliance on cloud provider availability
 - Can be used as an API endpoint for you to run curl commands against in your code for looping prompts and running it locally
 - Integrates with GitHub Copilot and CodeGPT
 - <https://ollama.com/download>



CodeGPT VSCode Extension

- Built in VS Code extension that allows you to run local LLM's using Ollama as well as Cloud LLMs. Think of it as a "free" GitHub Copilot



Ollama/CodeGPT Demo

- Laptop experience
- Our thoughts



Microsoft AI Toolkit/NPU Demo

- Running a local LLM using the NPU using Microsoft's AI Toolkit VS Code extension
- Let's see if this "Copilot+ PC" hype is real



Ollama Part Deux

- Let's try again with a bit more horsepower...



GitHub Copilot Demo

- Add context
- Review and comment on suggestions
- Generating unit tests with Pester
- Test using Ollama models with GitHub Copilot



Generating Scripts



April 7-10, 2025

15 / 33

Bad Prompt

- [Link](#)

Write a PowerShell script to export a list of users from active directory including all attributes

Reasoned for a couple of seconds

Below is a simple PowerShell script that will retrieve all user objects from Active Directory and export all **attributes** to a CSV file. Keep in mind that retrieving every property for a large AD environment can result in a very large export. Adjust the command as necessary if you only need certain attributes.

```
powershell
# Ensure the ActiveDirectory module is available
Import-Module ActiveDirectory

# Retrieve all user accounts and all properties
Get-ADUser -Filter * -Properties * |

# Select all properties
Select-Object *

# Export the results to a CSV file
Export-Csv -Path "C:\Temp\AllADUsers.csv" -NoTypeInformation
```



Good Prompt

- Link

Write a PowerShell script to export a list of users from active directory including all attributes, export the results to a CSV. Additionally ensure that all attributes are parsed into strings so that there are no System.* references in the CSV export. Additionally ensure that all required PowerShell module are installed before running the script.

Better Prompt

- [Link](#)

Write a PowerShell script to export a list of users from active directory including all attributes, export the results to a CSV. Additionally ensure that all attributes are parsed into strings so that there are no System.* references in the CSV export. Additionally ensure that all required PowerShell module are installed before running the script.

Also use a method to run this script from a computer that is joined to the Active Directory domain, and not from a domain controller. That is do not use the ActiveDirectory module to make the calls to pull a list of users.

Generating Scripts

Tips & Tricks

- Be verbose
- Include constraints and context
- Use LLMs as your debugging buddy

Generating Scripts

Tips & Tricks

- Be verbose



- Include constraints and context
- Use LLMs as your debugging buddy

- Input exact parameters for how you want things to function so it can build in the logic and produce an accurate script right away
- Specify input and output formats
- Write your prompt out in Notepad or Markdown and fine tune there if you don't get the response you're looking for

Generating Scripts

Tips & Tricks

- **Be verbose** 
- Include constraints and context
- Use LLMs as your debugging buddy

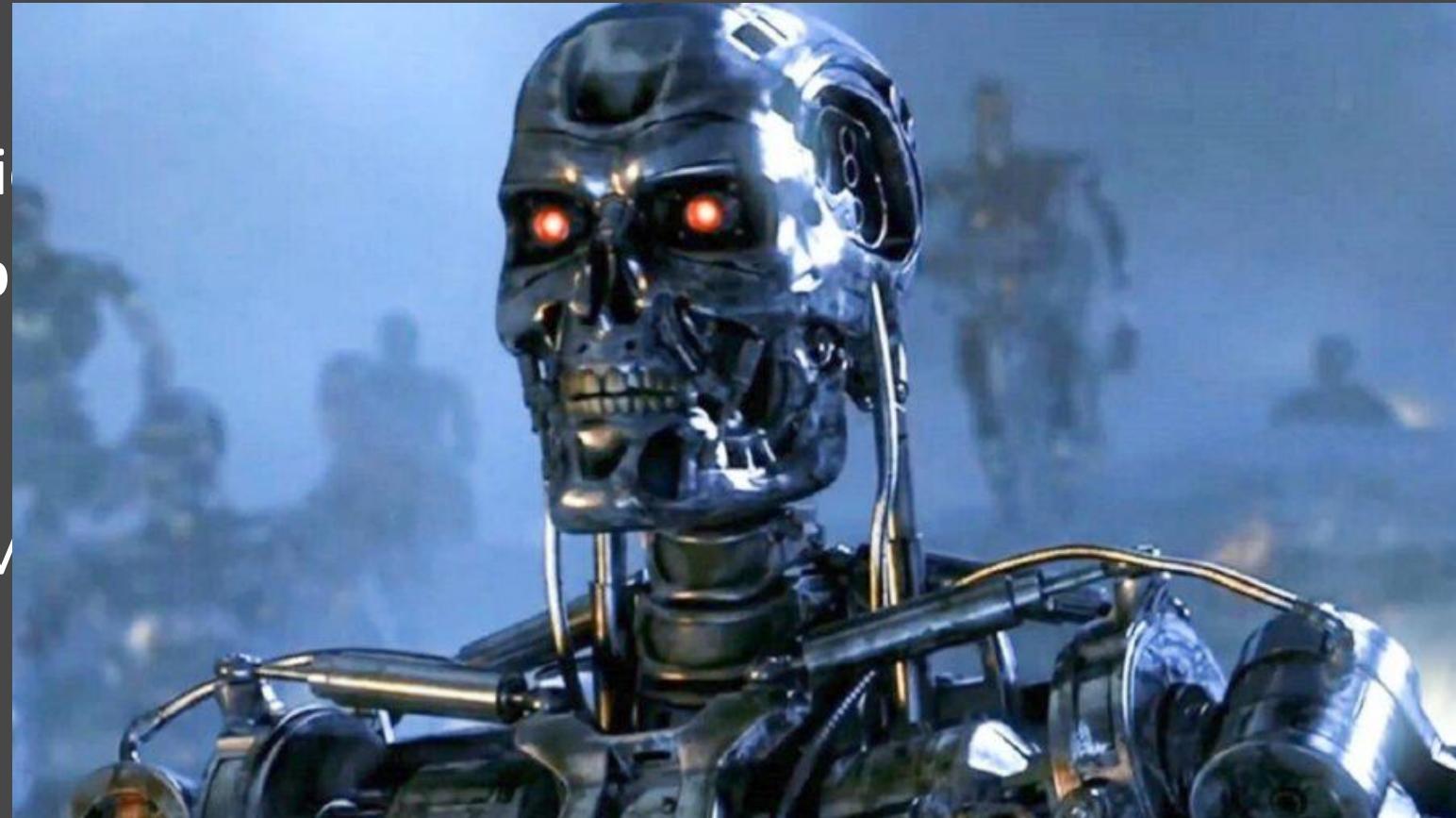
- Be demanding with the LLM, it is there to provide you with the best output it can provide, but it needs enough context to generate the best output to achieve your end result

Generating Scripts

Tips & Tri

- Be verb
- Include context
- Use LLM buddy

how you
it can build
an accurate
formats



Generating Scripts

Tips & Tricks

- Be verbose
- **Include constraints and context**
- Use LLMs as your debugging buddy

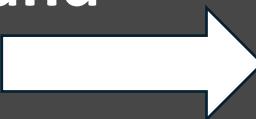


- Include constraints such as using a specific PowerShell module “Use the Microsoft.Graph.Beta PowerShell module”
- Add in what versions of PowerShell you want the script to run on
- Provide an example of code you’ve written in the past

Generating Scripts

Tips & Tricks

- Be verbose
- **Include constraints and context**
- Use LLMs as your debugging buddy



April 7-10, 2025

Customize ChatGPT
Introduce yourself to get better, more personalized responses ⓘ

What should ChatGPT call you?

Name

What do you do?

Enterprise IT Consultant

What traits should ChatGPT have? ⓘ

Explain the thought process and steps to reach the conclusion
ChatGPT is an expert in writing PowerShell and diagnosing bugs in scripts

+ Chatty + Witty + Straight shooting + Encouraging + Gen Z
+ Skeptical + Traditional + Forward thinking + Poetic ⚙️

Anything else ChatGPT should know about you? ⓘ

Interests, values, or preferences to keep in mind

Advanced ▾

Enable for new chats Cancel Save

Generating Scripts

Tips & Tricks

- Be verbose
- Include constraints **and context**
- **Use LLMs as your debugging buddy**



- After you've run the script if you run into any issues with the script, enter the error message into your LLM of choice and ask it to debug the error for you
- Add error handling to scripts using a prompt to handle various issues that may come up during script execution

Generating Scripts

Tips & Tricks

- Be verbose
- Include constraints **and context**
- **Use LLMs as your debugging buddy**



- If one LLM doesn't give you a good output for debugging, try another model or provider!

Script Performance Optimizations

- Using LLMs to assist with script optimizations can help accelerate code development while also increasing efficiency of scripts and methods that are used in the code
- Take any non-proprietary code you want to optimize (make it run faster, simpler to read/maintain) and input it into an LLM of your choice.
 - Certain LLMs will have different recommendations for what to do to your script



Script Performance Optimizations

- Prompt: "Analyze the below PowerShell code and identify opportunities for performance enhancements such as reducing script execution time and enhancing usability of the script for new users."
- Continue to ask for additional recommendations until it starts getting nit-picky or just straight up hallucinating recommendations
- At the end, ask for a prioritized list of recommendations based on all the recommendations provided



Script Performance Optimizations

- Ask for additional optimizations, then organize the optimizations by order of priority/complexity
- Use multiple LLMs to get a variation of options
- Edge cases for improvements



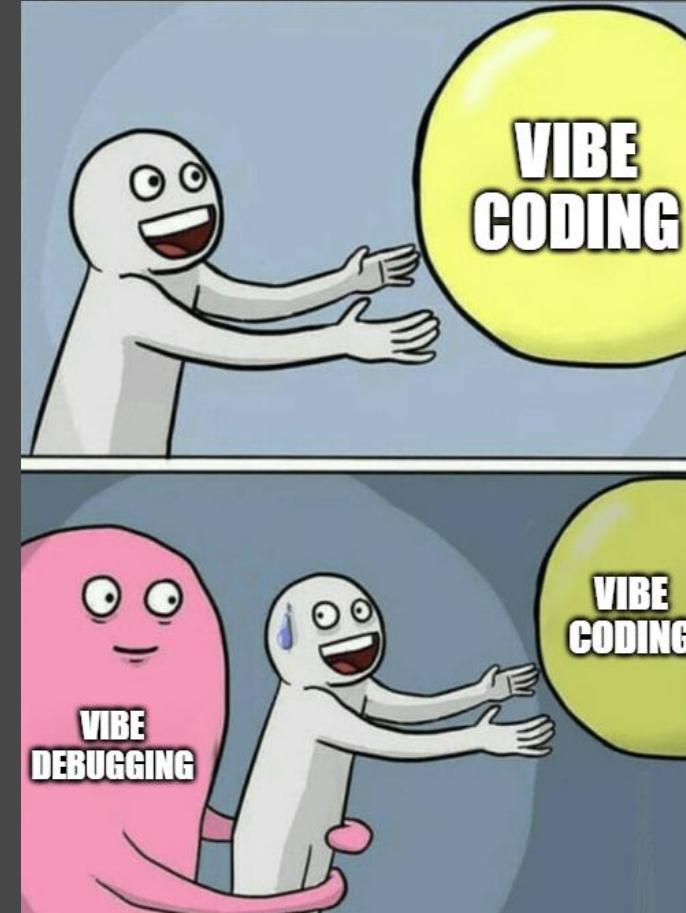
Script Performance Optimizations

- Demo
 - Optimize existing scripts we think are already pretty good
 - Repair-NTFSPermissions.ps1
 - Let's try this on a simple script to see what it produces in terms of recommendations using
 - [Get-AzADPrivilegedRoles.ps1](#)



Script Performance Optimizations

- Error handling
 - Within the context of a chat, ask the LLM to help resolve errors when running your script. Give the LLM the script contents then paste in the error you're looking to resolve



THANK YOU

FEEDBACK IS A
GIFT. PLEASE
REVIEW THIS
SESSION

