# Placement Empowerment Program

***Cloud Computing and DevOps Centre***

Write the Shell Script to Monitor Logs : Create a script that monitors server logs for errors and alert you

Name: Shakthi Lakshmanan V Department: CSE



## **Introduction**

Log files record system activities and errors, helping IT teams identify problems. Monitoring these logs is important for detecting issues early. Automating this process makes it easier and ensures nothing is missed.

This project shows how to use a simple PowerShell script to watch a log file in real-time. The script will look for specific words (like "error") and alert the user when they appear.

## **Overview**

In this project, we will:

1. **Read a log file continuously** to check for updates.
2. **Look for specific keywords** like "error" in new log entries.
3. **Display an alert** when a keyword is found.

This method is useful for IT professionals to quickly catch issues in system logs.

## **Objectives**

1. Automate log file monitoring.
2. Learn how to create and run PowerShell scripts.
3. Detect keywords like "error" in logs as they appear.
4. Provide quick alerts to improve troubleshooting.

## **Why It’s Useful**

* **Find issues early** – Detect errors as soon as they happen.
* **Learn PowerShell** – A useful tool for automation.
* **Free & simple** – No need for expensive software.
* **Saves time** – No more manual log checking.
* **Can be expanded** – Monitor multiple log files if needed.

# Step-by-Step Overview

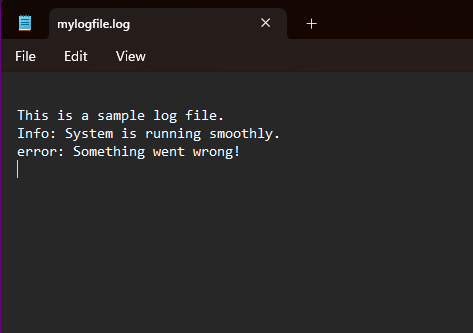
## Step 1:

Create a Folder called logs for Your Logs and Script



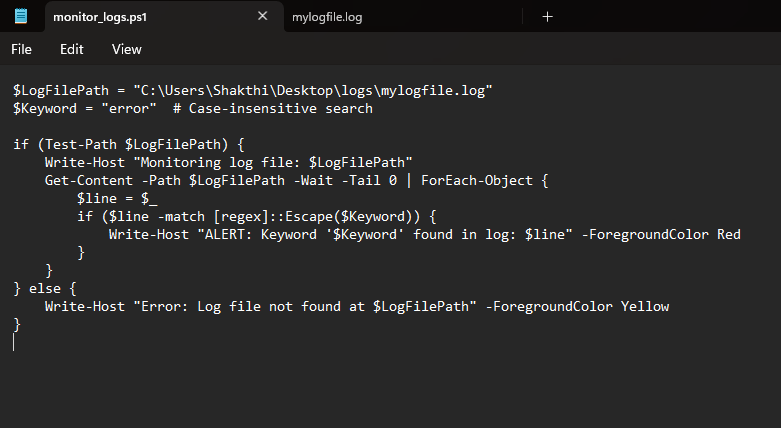
## Step 2:

Open Notepad and Add the following sample text to it and Save the file as **mylogfile.log** inside the logs folder



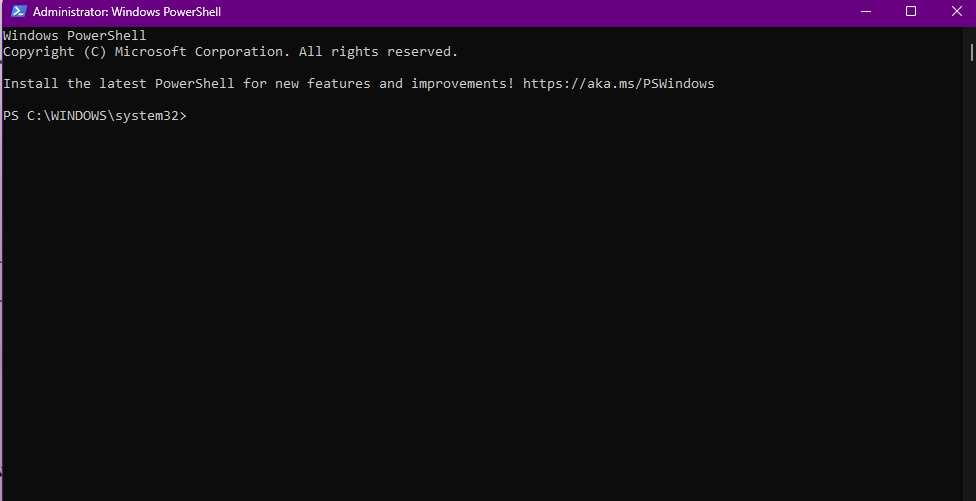
## Step 3:

Open Notepad and Type the following PowerShell script into it and Set the $LogFilePath address to the mylogfile.log which you saved in logs folder. Save the file as monitor\_logs.ps1 inside the same logs folder



## Step 4:

Click the Windows Key and Search for Windows PowerShell and click Run as Administrator.

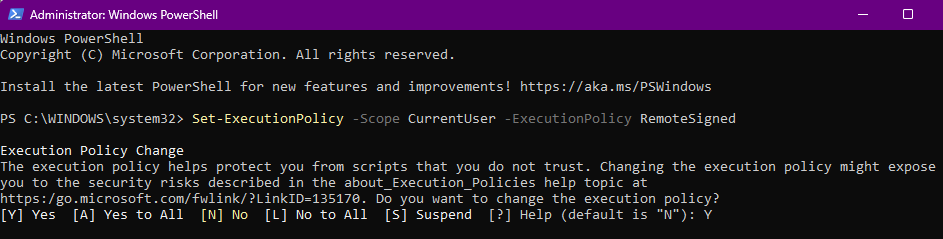


## Step 5:

Run the following command to allow script execution:

### Set-ExecutionPolicy -Scope CurrentUser -ExecutionPolicy RemoteSigned

When prompted, type Y and press Enter.



## Step 6:

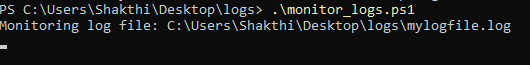
Navigate to the logs folder



## Step 7:

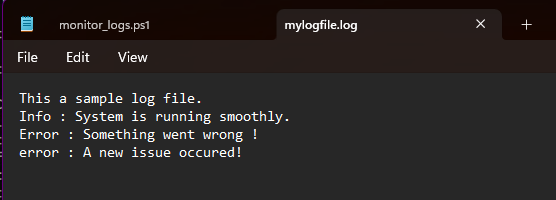
Run the script:

**.\monitor\_logs.ps1**

****

## Step 8:

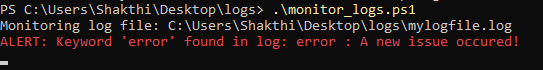
Open mylogfile.log in Notepad and Add a new line with the word "error" and Save the file.



## Step 9:

Check PowerShell — you should see an alert like:

**ALERT: Keyword 'error' found in log: error: A new issue occurred!**

****

## Explanation:

1. When the script is running, it continuously monitors the log file.
2. If any line containing the keyword "error" is added to the file, it immediately triggers an alert in PowerShell.

## Outcome:

By completing this Proof of Concept (PoC), we will:

* 1. Successfully create and execute a PowerShell script to monitor log files in real time.
  2. Detect and alert on predefined keywords (e.g., "error") to highlight critical events.
  3. Gain hands-on experience with PowerShell scripting and automation on a Windows system.
  4. Understand the importance of log monitoring in proactive system maintenance and troubleshooting.
  5. Learn to customize and scale the script for more advanced monitoring scenarios in future projects.