Placement Empowerment Program

Cloud Computing and DevOps Centre

Automate File Copying with a Script

Create a script to copy files from one folder to another automatically.

Name: Harshana Perianayaki B Department: IT

St. JOSEPH'S COLLEGE OF ENGINEERING | St. JOSEPH'S INSTITUTE OF TECHNOLOGY
AUTONOMOUS INSTITUTIONS, AFFILIATED TO ANNA UNIVERSITY

INTRODUCTION

Automation is a powerful tool to streamline repetitive tasks, saving time and reducing errors. One common task in file management is copying files from one folder to another, often required for backups, synchronization, or organizing files. By automating this process, users can ensure consistency, efficiency, and reliability in their workflows without manual intervention.

OVERVIEW

This project aims to develop a script that can automatically copy files from a source directory to a destination directory. It includes the following key features:

1. Cross-platform support (Linux and Windows).
2. Scheduled execution using system tools like Cron (Linux) or Task Scheduler (Windows).
3. Error handling to manage missing files, directory permissions, and logging.
4. Optional extensions for advanced requirements like filtering files by type, date, or size.

The process uses basic scripting tools like `cp` for Linux and `xcopy` for Windows, coupled with the appropriate scheduling tools to ensure automation.

OBJECTIVES

The primary objectives of this project are:

1.Simplify File Copying Automate the repetitive task of manually copying files from one location to another.

2.Enhance Efficiency: Ensure all files are copied promptly without delays, minimizing the time spent on manual tasks.

3.Ensure Consistency: Avoid human error by creating a reliable and repeatable process for file copying.

4.Adapt to Different Platforms: Provide solutions that work seamlessly on Linux and Windows.

5.Enable Scheduling: Automate the task to run at regular intervals (daily, weekly, or custom), ensuring routine operations like backups are handled effortlessly.

6.Support Logging: Maintain records of file-copying activities for monitoring and troubleshooting purposes.

IMPORTANCE

1. **Time-Saving**: Automating the process eliminates the need for manual intervention, saving hours of work.
2. **Error Reduction**: Reduces human errors, such as forgetting to copy specific files or overwriting important ones.
3. **Improved Organization**: Helps maintain a consistent structure for file storage and backups.

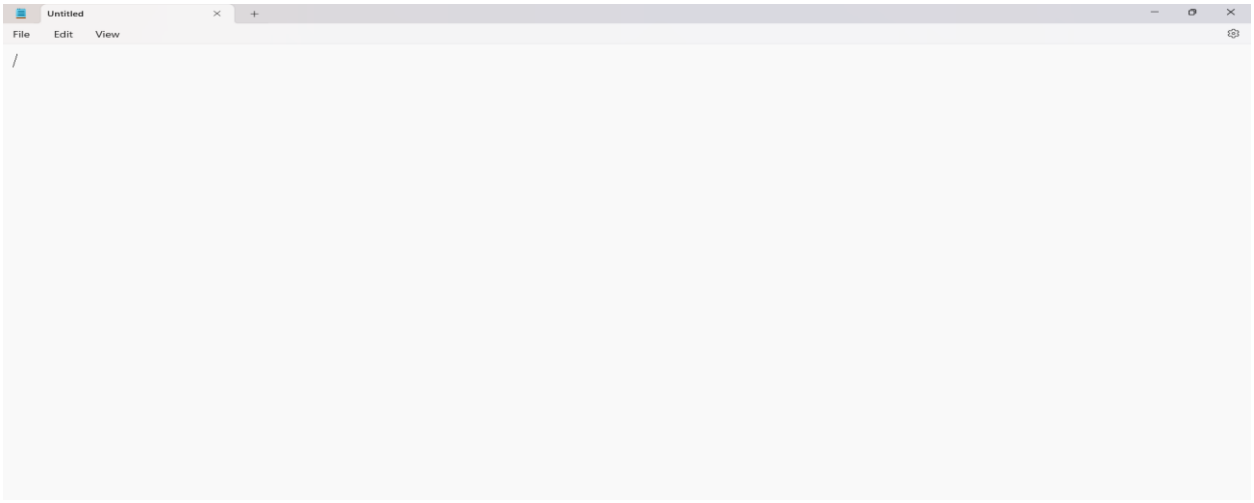
4. Useful in Various Scenarios: This approach is valuable for IT professionals, businesses, and individuals dealing with frequent file transfers or backups.

5. Scalability and Reusability: Once created, the script can be reused and scaled up to handle more complex tasks.

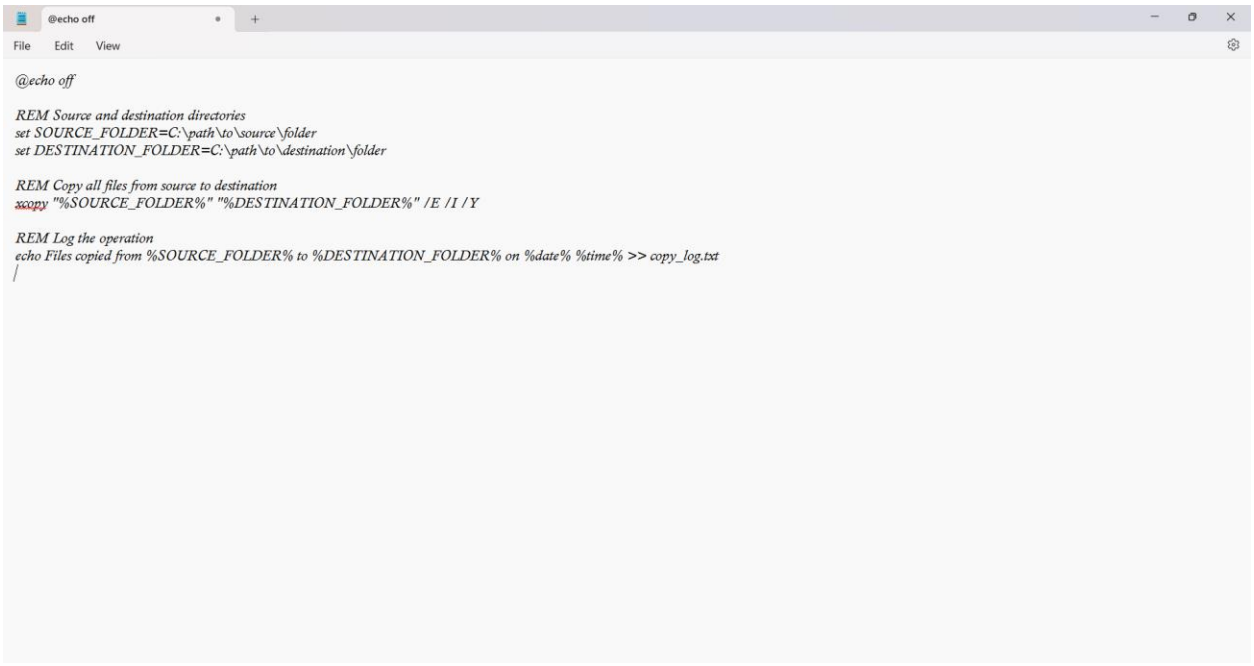
STEP BY STEP OVERVIEW

Step 1: Create a Batch Script

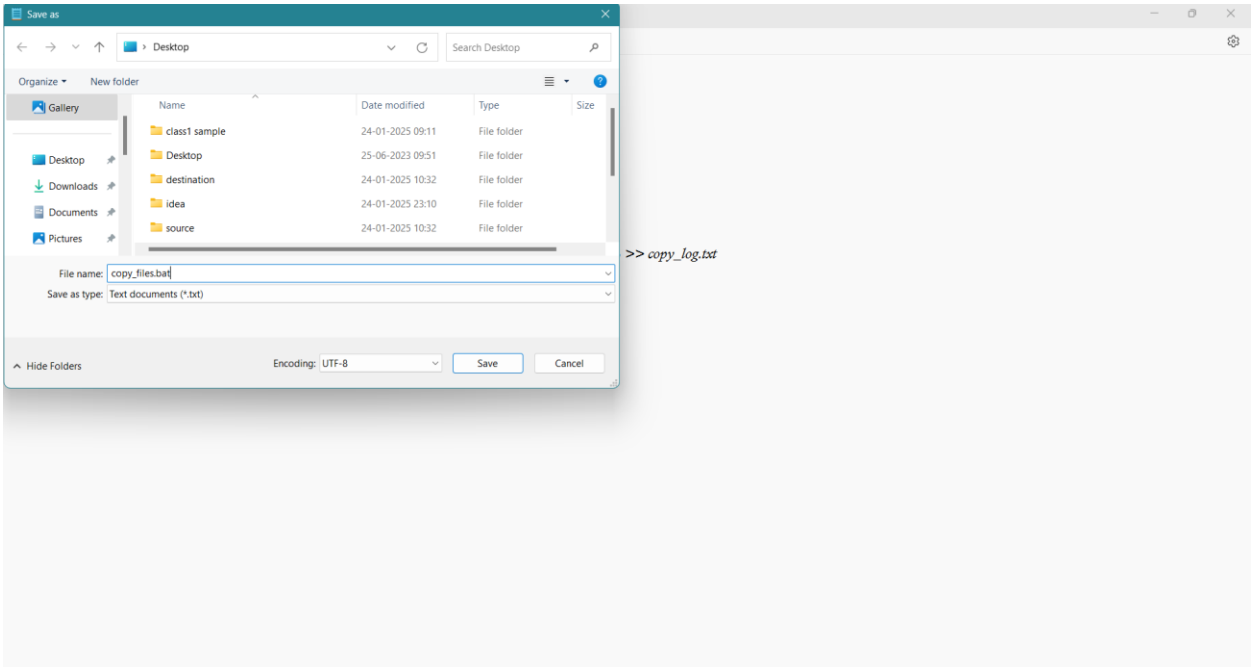
1.Open a text editor (like Notepad)



2.Add the following script:

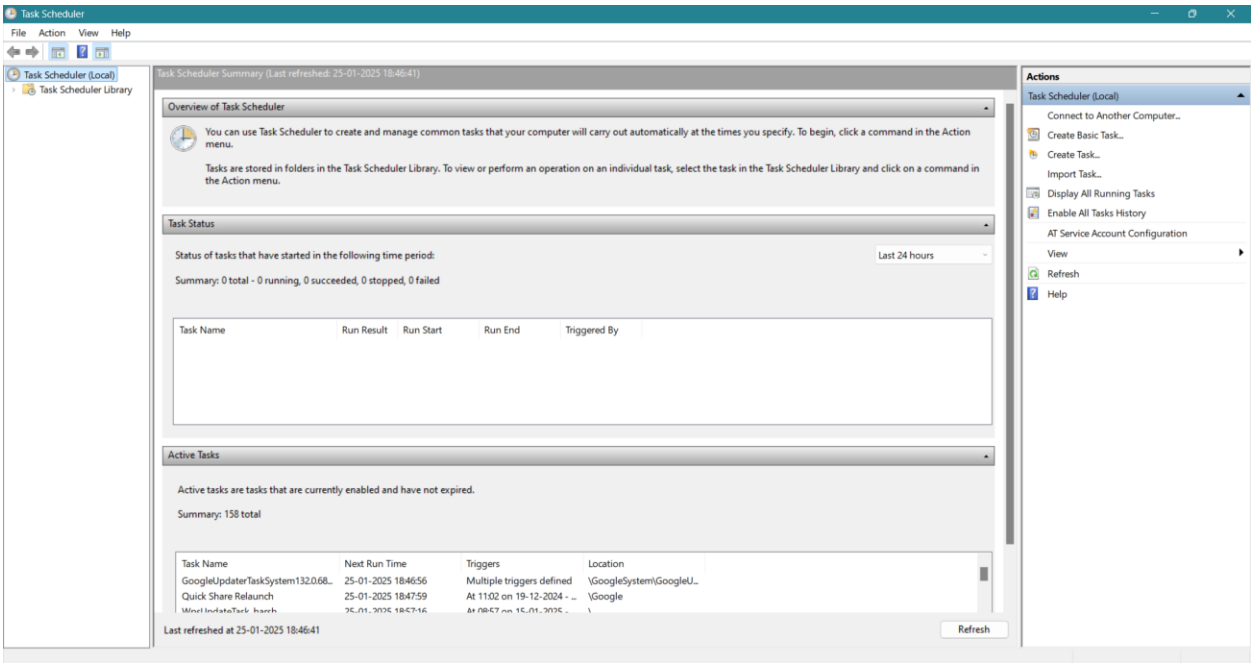


3.Save the file as `copy_files.bat`

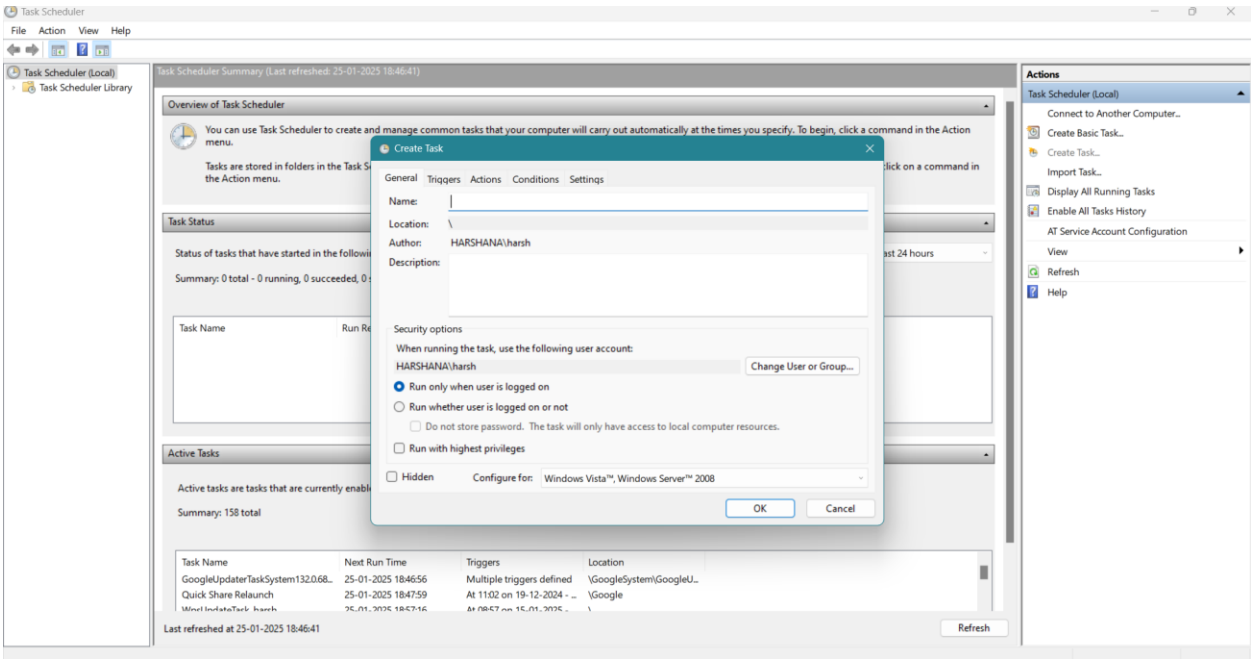


***Step 2*:** Schedule with Task Scheduler.

1.Open Task Scheduler (search for it in the Start menu).



2.Click **Create Task**.



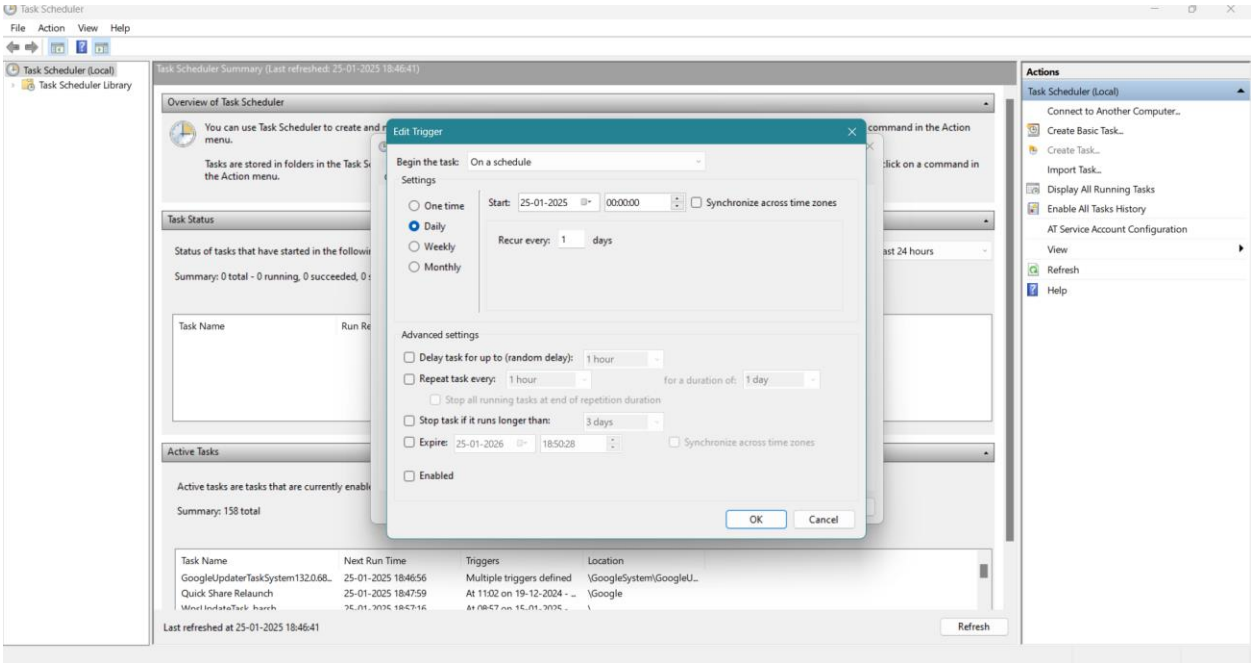
3. Under the **General** tab:

Give your task a name (e.g., "File Copy Automation").

Choose **Run whether user is logged on or not**.

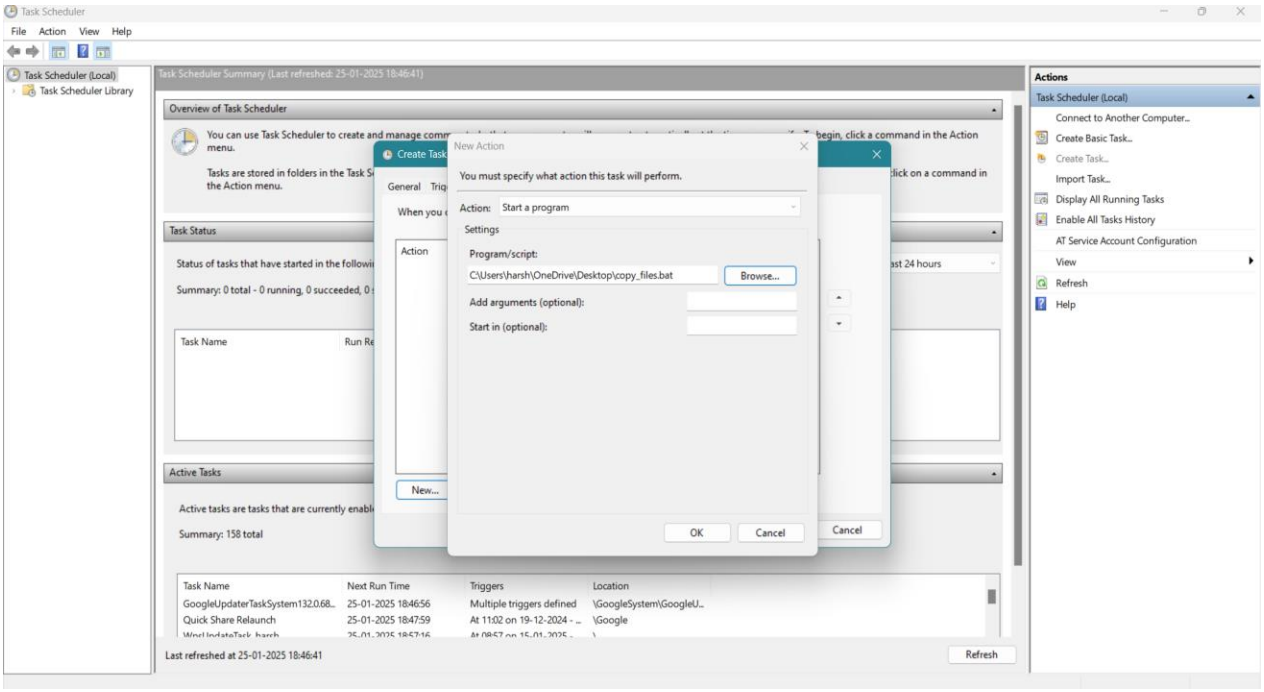
4. Under the **Triggers** tab:

Click **New**, then set a schedule (e.g., daily at midnight).

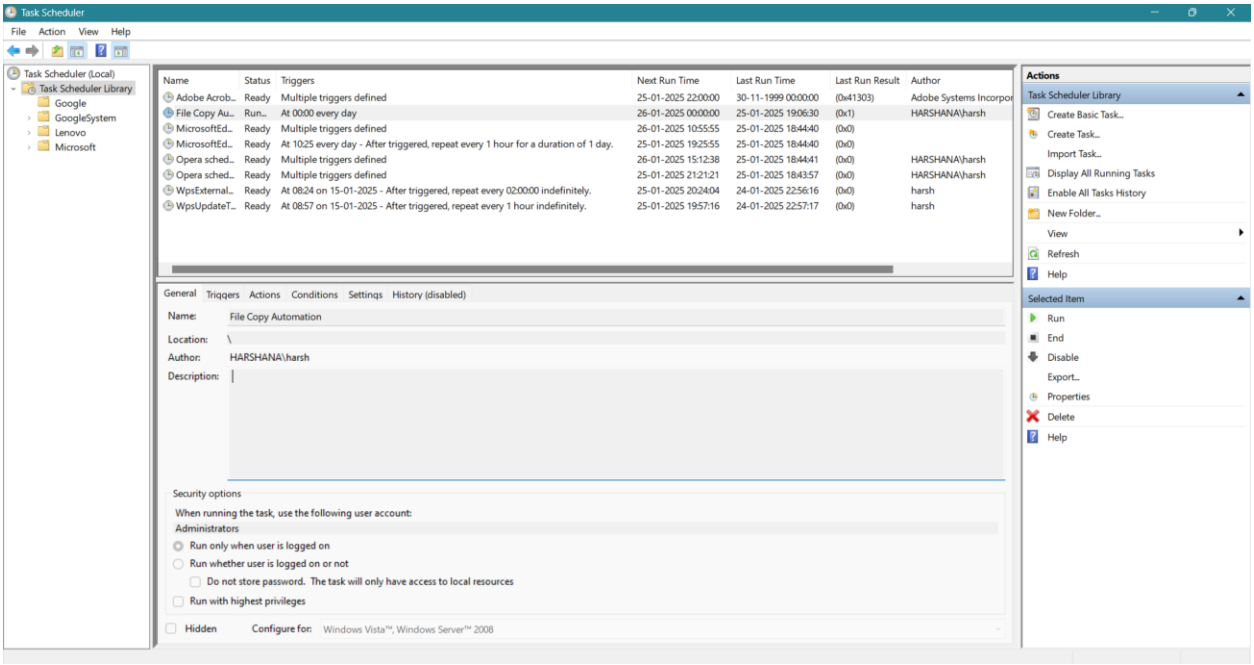


5. Under the **Actions** tab:

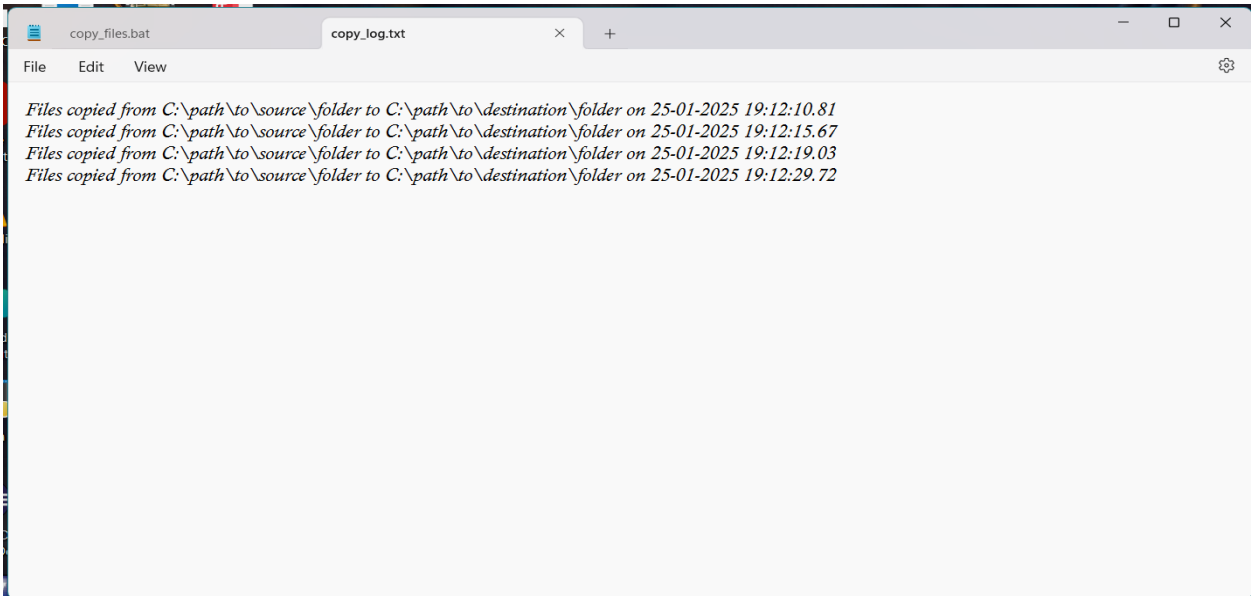
Click New, then browse to your copy_files.bat file.



6. Save and enable the task.



At last, file is copied from one folder to another automatically.



OUTCOMES

By completing this Proof of Concept (PoC) of automating a task using Task Scheduler, you will:

1. Successfully set up an automated task that triggers on a schedule or when manually run.
2. Execute a batch script to copy files from a source folder to a destination folder automatically.
3. Understand how to use Task Scheduler to automate repetitive tasks in Windows.
4. Gain familiarity with task triggers, actions, and conditions in Task Scheduler.
5. Save time and ensure consistent execution of file operations without manual intervention.
6. Optionally test the task to verify that it runs as expected and achieves the desired outcome.
7. Enhance your workflow automation skills with practical hands-on experience.