CREATE A SIMPLE BACKUP SCRIPT

OBJECTIVE:

Create a script that backs up your entire Git repository to a local folder daily.

WHY BACKUP?

A backup is a copy of data that is stored separately from the original to protect against data loss, corruption, or accidental deletion. Backups ensure that important files, systems, or applications can be restored in case of failure, cyberattacks, or disasters.

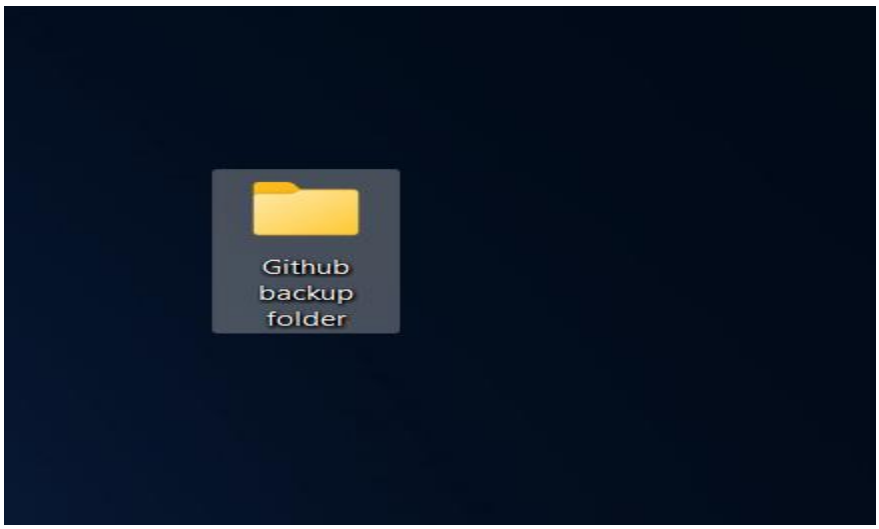
IMPORTANCE:

- Disaster Recovery – Protects against hardware failures, malware, and system crashes.
- Data Security – Prevents permanent loss from cyberattacks like ransomware.
- Version Control – Restores previous versions of files if needed.
- Compliance & Regulations – Some industries require regular backups for data protection.

STEP BY STEP OVERVIEW:

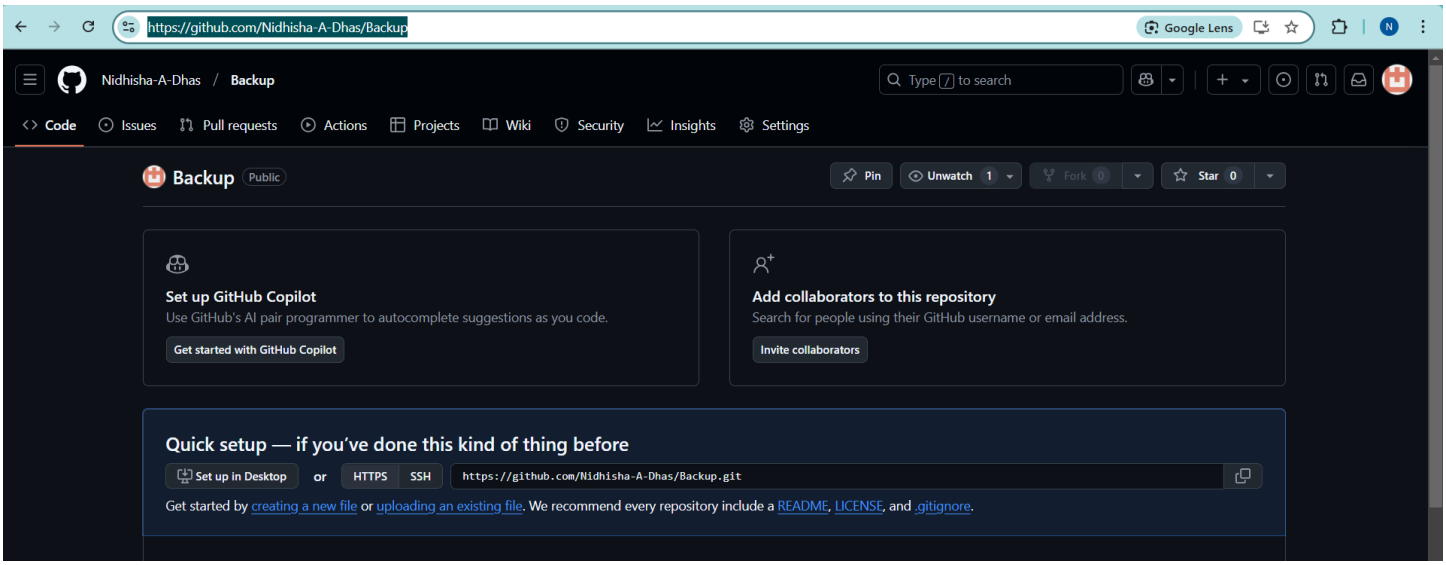
Step 1: CREATE A FOLDER

- Create a folder name ‘Github-backup-folder’ on your desktop.
- This folder stores the backup files.



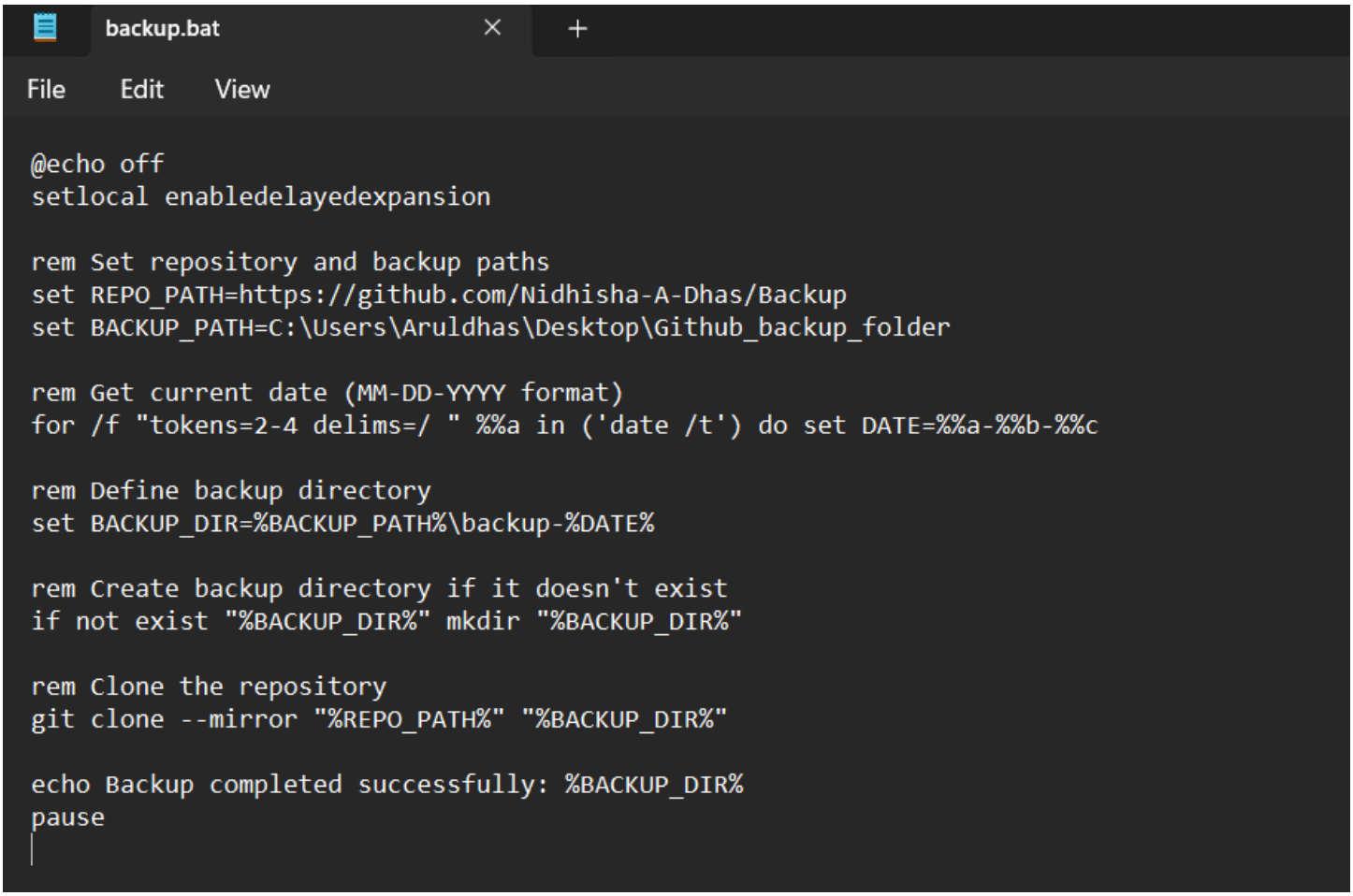
Step 2: CREATE A GIT REPOSITORY

- Login into your ‘Github’ account and create a new repository named ‘Backup’.



Step 2: CREATING A SCRIPT

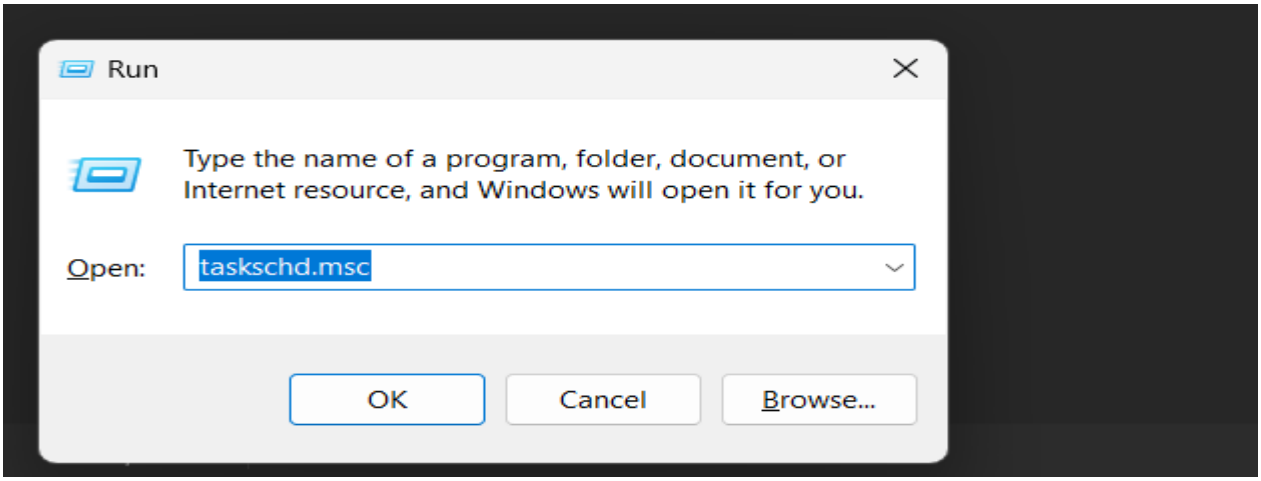
- Open your Notepad and type the script as shown below.



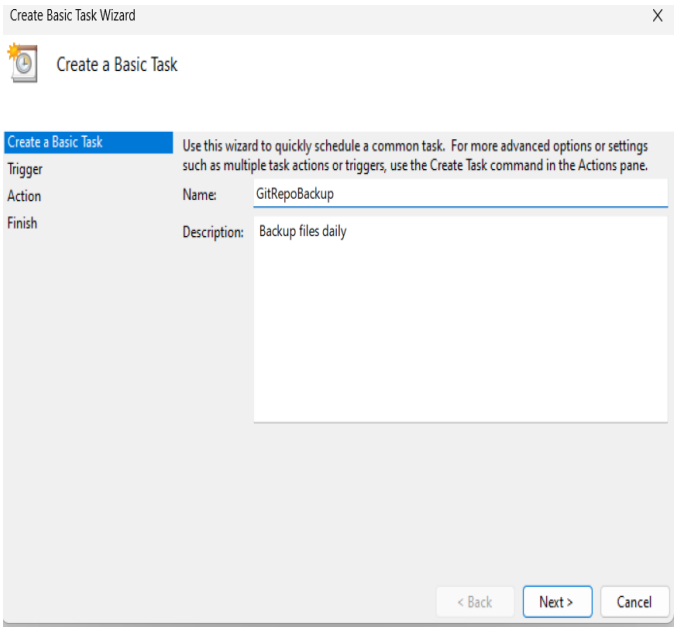
- Save this file in ‘bat’ format eg: backup.bat our desktop.

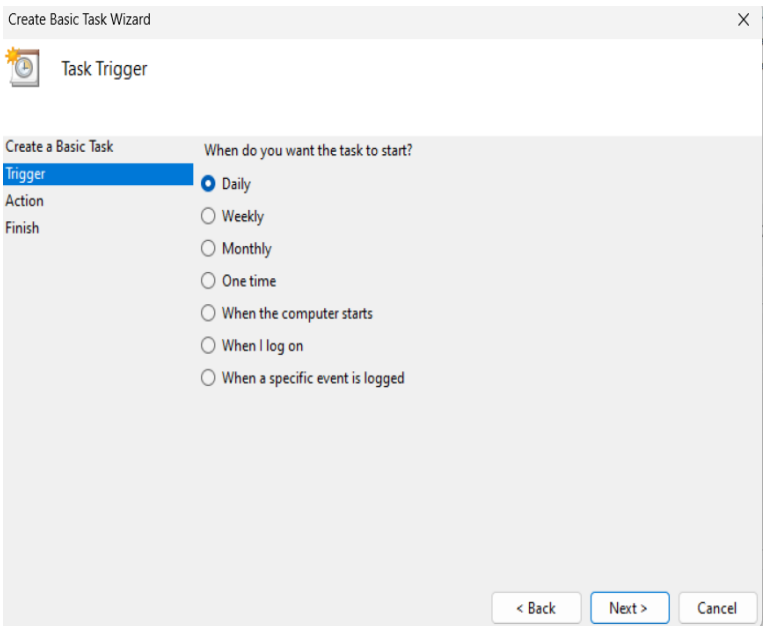
Step 3: TASK SCHEDULAR

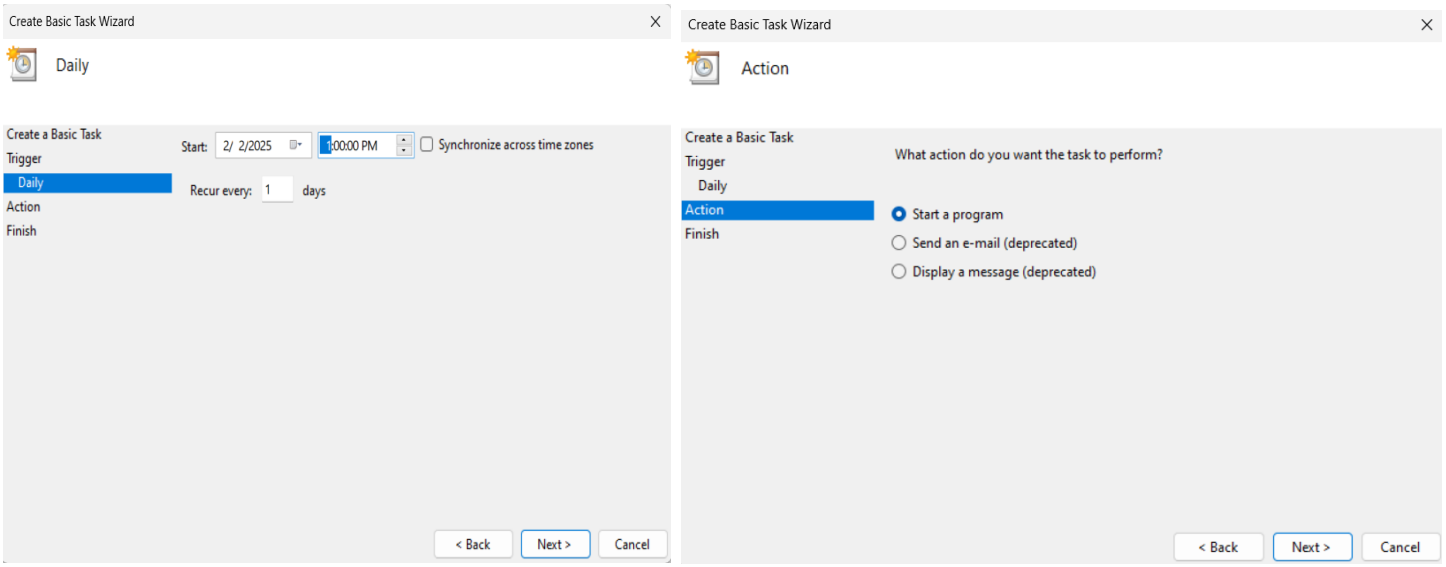
- Click ‘Windows+R’ and this will open a small dialogue box.
- Type ‘taskschd.msc’ in the ‘Run’ box.

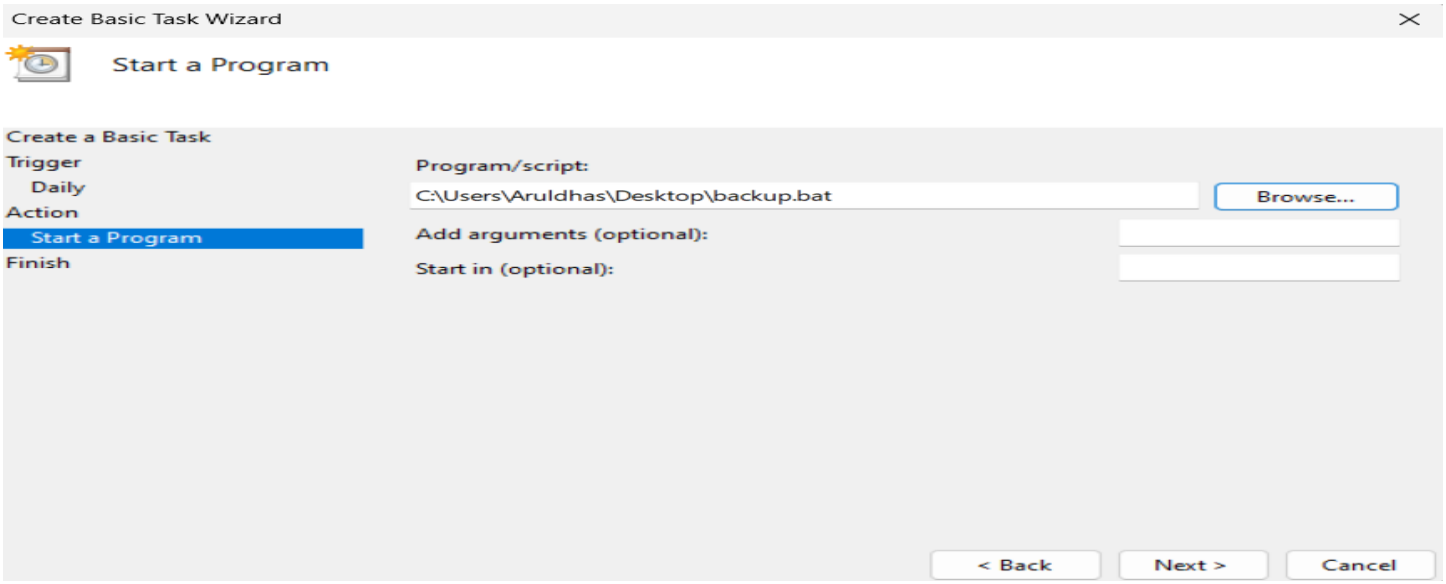


- In the task Schedular window, click on ‘Create Basic Task’.
- A wizard will open to guide you through the step.
- Give the name of the Basic task and specify description if needed.





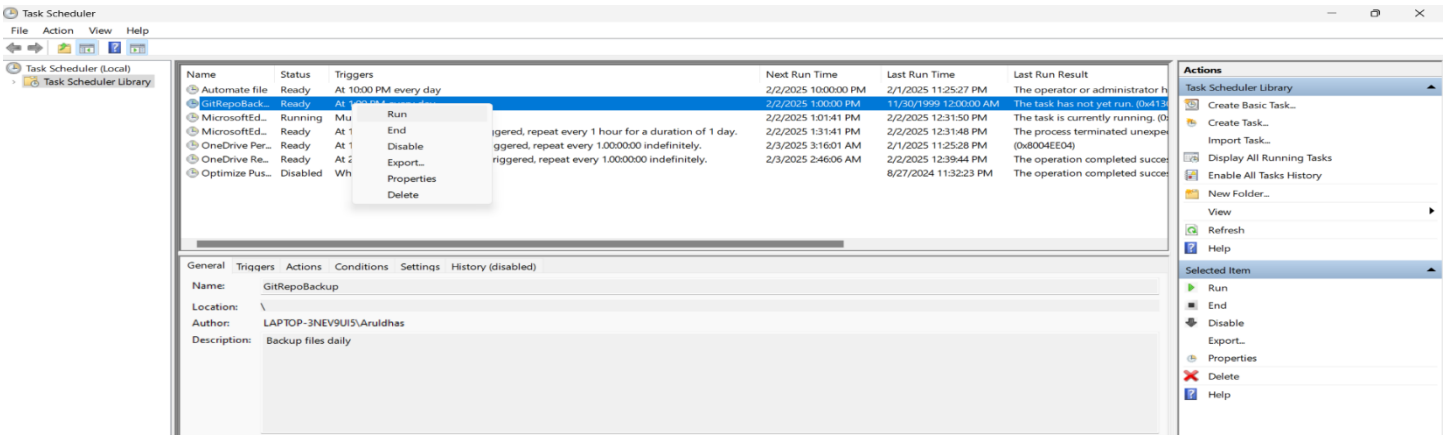




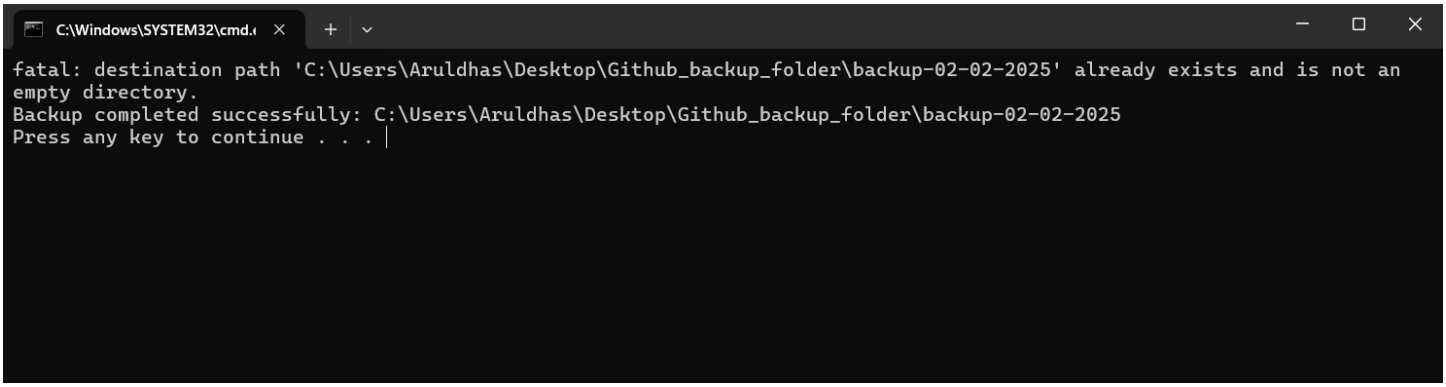
- Review and click finish to save and schedule the task.

Step 4: TRIGGER THE TASK

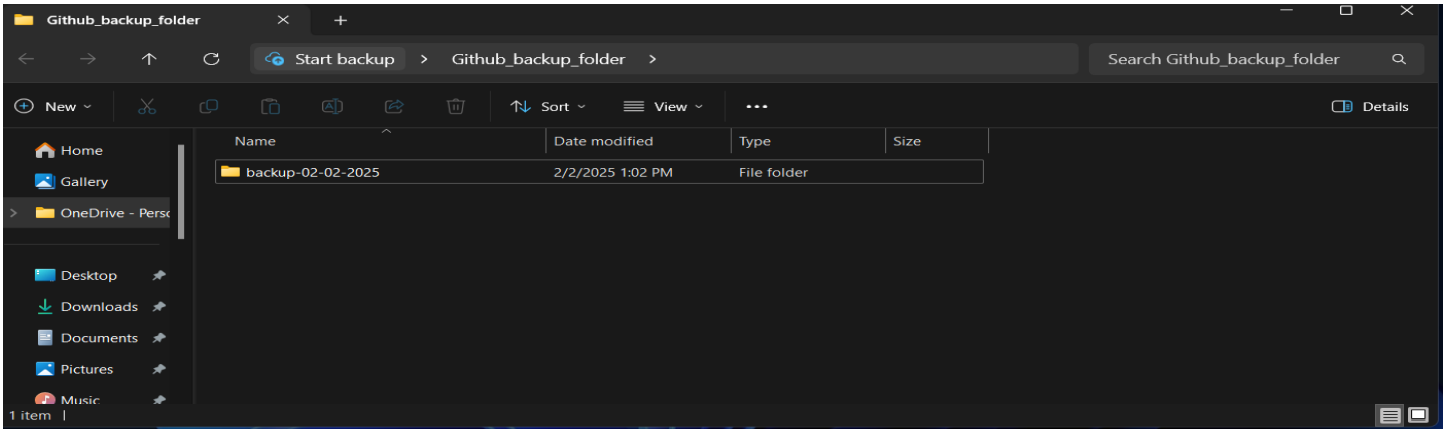
- On the ‘Task Scheduler Library’ find the task you have just created and right-click on it to run the task.



- This will manually trigger the task immediately.



- The file has been successfully backup-ed.



CONCLUSION:

Thus, from this POC, we have learnt

- To successfully implement a automate backup system for Git repositories.
- Task Scheduler's automation capabilities.