



St. JOSEPH'S
GROUP OF INSTITUTIONS
OMR, CHENNAI - 119



Placement Empowerment Program

Cloud Computing and DevOps Centre

Create a Simple Backup Script: Create a script that backs up your entire Git repository to a local folder daily.

Name:

Samuel James Billy Graham

Department: CSE



St. JOSEPH'S
COLLEGE OF ENGINEERING



St. JOSEPH'S
INSTITUTE OF TECHNOLOGY

AUTONOMOUS INSTITUTIONS, AFFILIATED TO ANNA UNIVERSITY

Introduction

Backing up Git repositories is essential to safeguard your work from accidental deletions, hardware failures, or corruption. Automating this process saves time and ensures regular updates without manual effort.

Overview

This solution automates Git repository backups on a Windows machine using a batch script and Task Scheduler. The script pulls the latest changes daily, stores them in a backup directory, and compresses the repository with a timestamped archive for easy organization. This ensures your codebase and version history remain secure and accessible.

Key Components

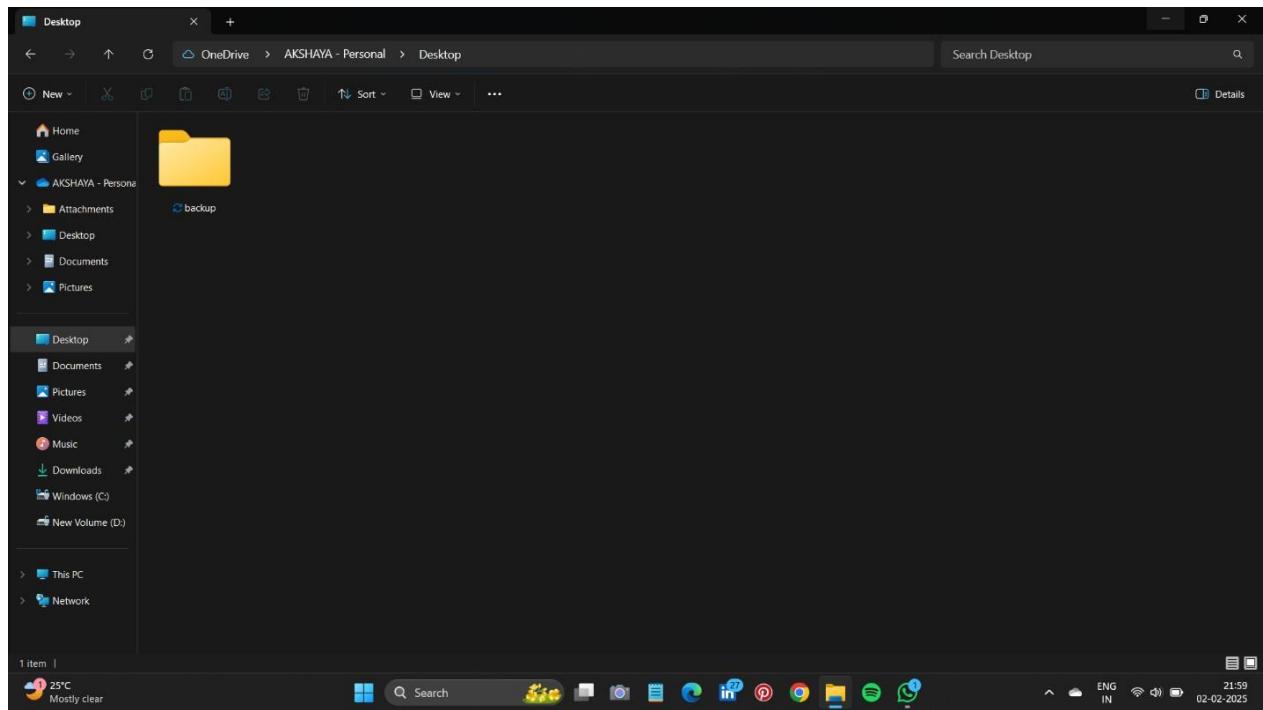
Batch Script: A .bat file is used to execute commands such as cloning the repository, pulling updates, and compressing the backup.

Task Scheduler: A built-in Windows tool is used to automate the script, ensuring it runs daily without manual intervention.

Step-by-Step Overview

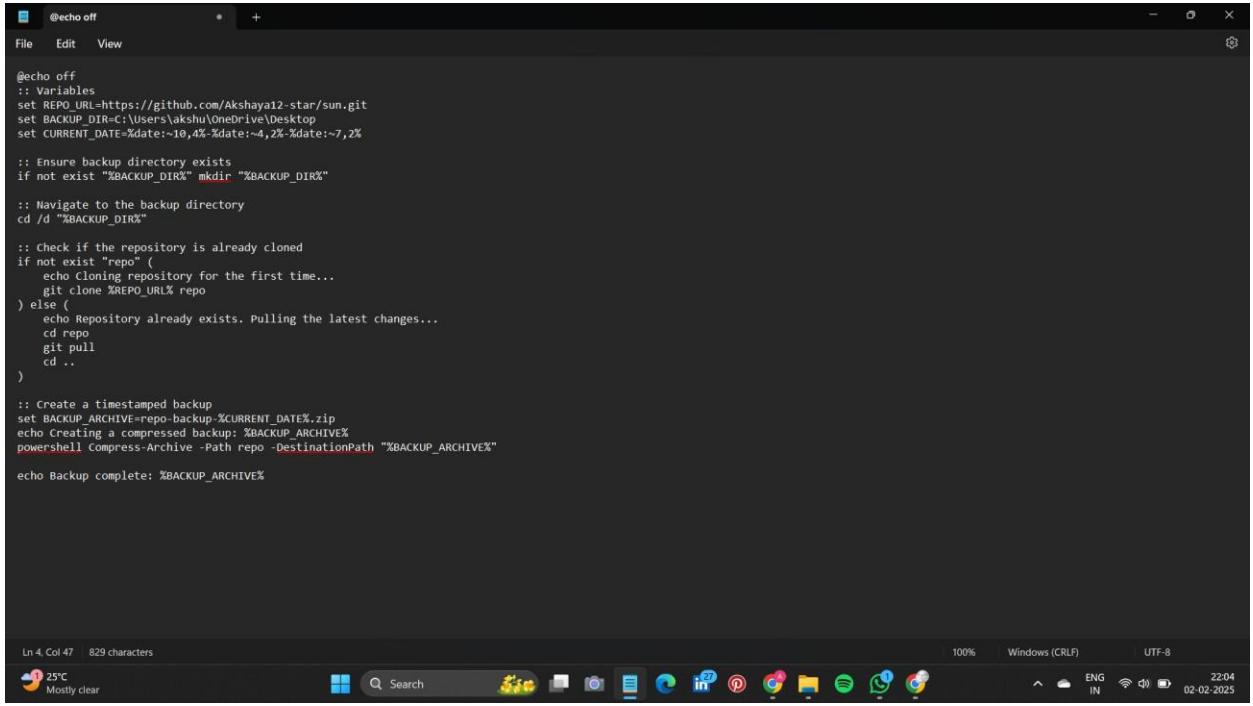
Step 1:

Create a folder named GitHub Backup Folder to store your Backup files



Step 2:

Open Notepad and type this script . Make sure that in set REPO_URL give the URL of the repository you want to backup and in set BACK_DIR give the file path of the folder which you created in first step . Then save it as **.bat format** (eg:backup.bat)



```
@echo off
:: Variables
set REPO_URL=https://github.com/Akshaya12-star/sun.git
set BACKUP_DIR=c:\Users\akshu\OneDrive\Desktop
set CURRENT_DATE=%date:~10,4%-%date:~4,2%-%date:~7,2%
:: Ensure backup directory exists
if not exist "%BACKUP_DIR%" mkdir "%BACKUP_DIR%"
:: Navigate to the backup directory
cd /d "%BACKUP_DIR%"
:: Check if the repository is already cloned
if not exist "repo" (
    echo Cloning repository for the first time...
    git clone %REPO_URL% repo
) else (
    echo Repository already exists. Pulling the latest changes...
    cd repo
    git pull
    cd ..
)
:: Create a timestamped backup
set BACKUP_ARCHIVE=repo-backup-%CURRENT_DATE%.zip
echo Creating a compressed backup: %BACKUP_ARCHIVE%
powershell Compress-Archive -Path repo -DestinationPath "%BACKUP_ARCHIVE%"

echo Backup complete: %BACKUP_ARCHIVE%
```

Ln 4, Col 47 | 829 characters

25°C Mostly clear

Search

100% Windows (CRLF) UTF-8

22:04 ENG IN 02-02-2025

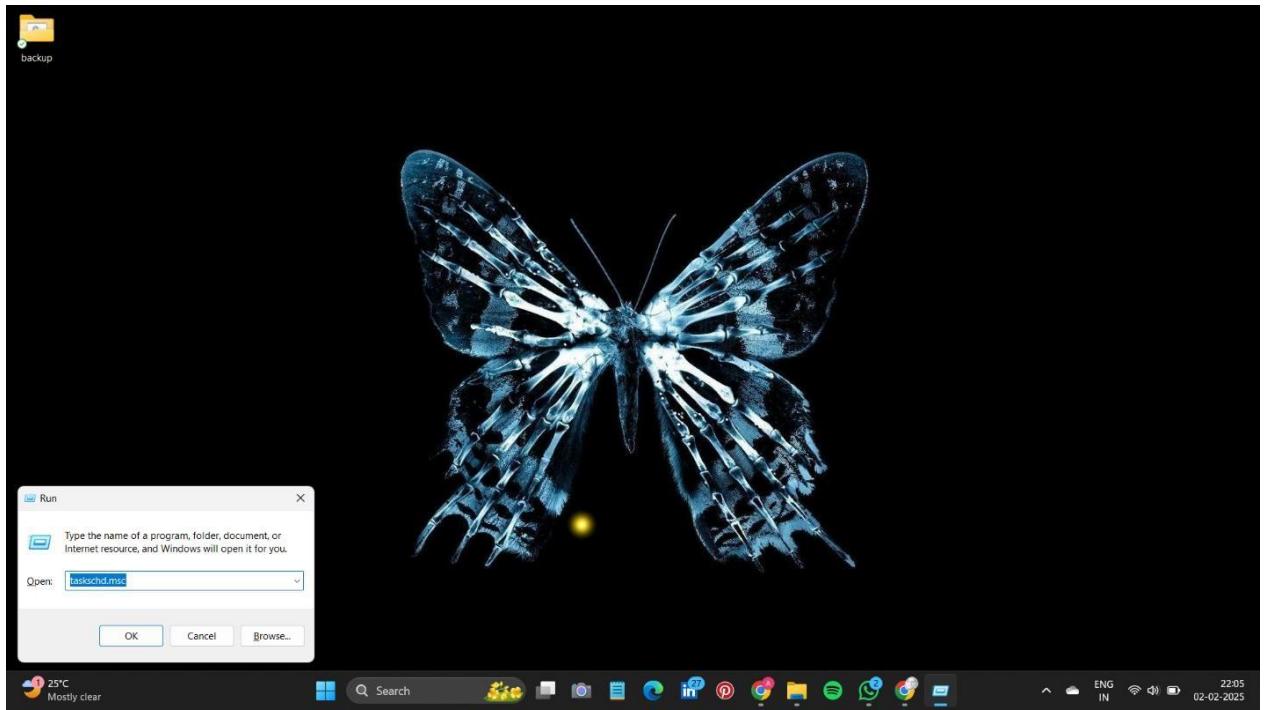
Step 3:

Press **Win + R** on your keyboard.

A small "Run" dialog box will pop up.

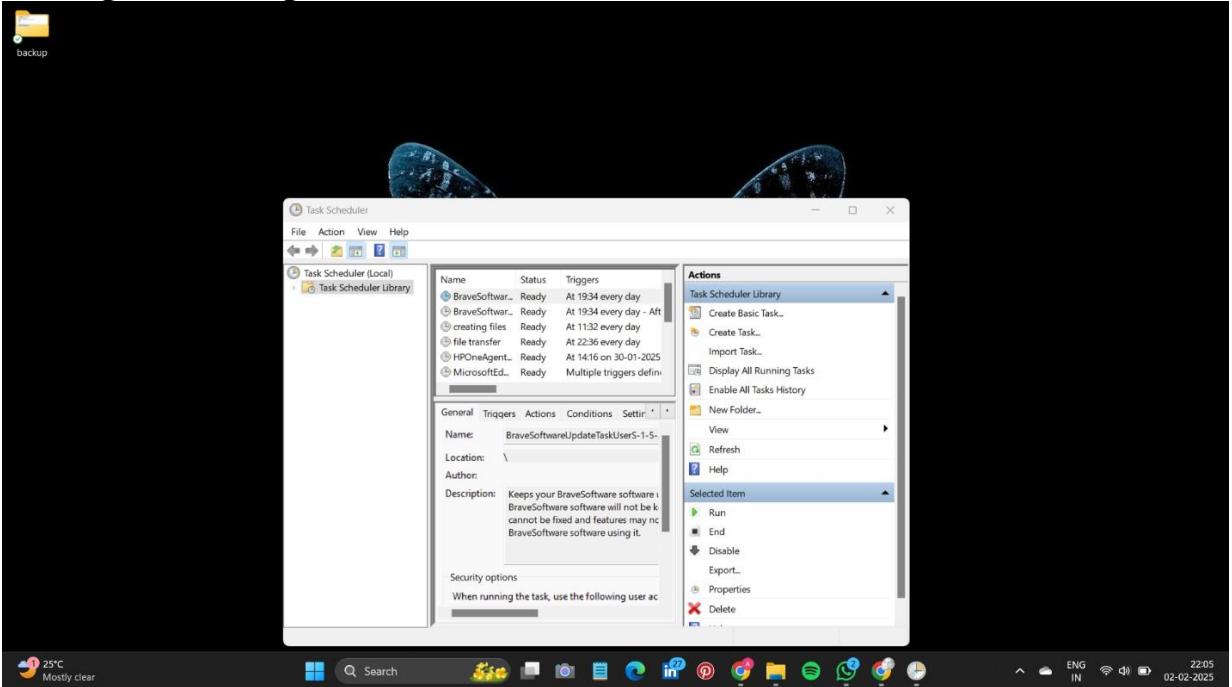
Type **taskschd.msc** (without quotes) in the Run box.

Press Enter or click OK. This will open the Task Scheduler window.



Step 4:

In the Task Scheduler window, look to the right-hand side for a button called "Create Basic Task". Click it. A wizard will open to guide you through the setup.



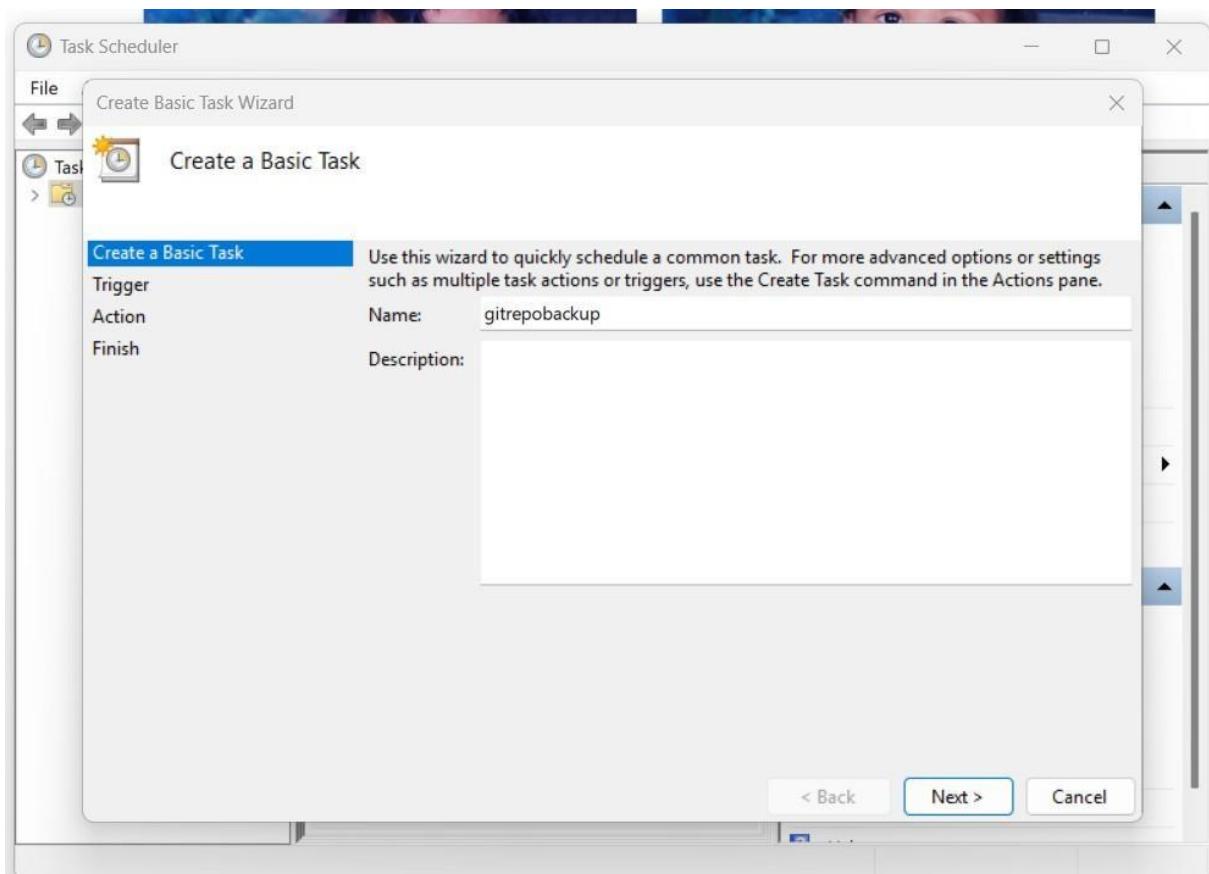
Step 5:

1. Enter a Name for the Task:

For example: "GitRepoBackup".(This can be anything that helps you remember what the task does.)

Optionally, you can add a description like "Backup files daily".

Click Next to continue.



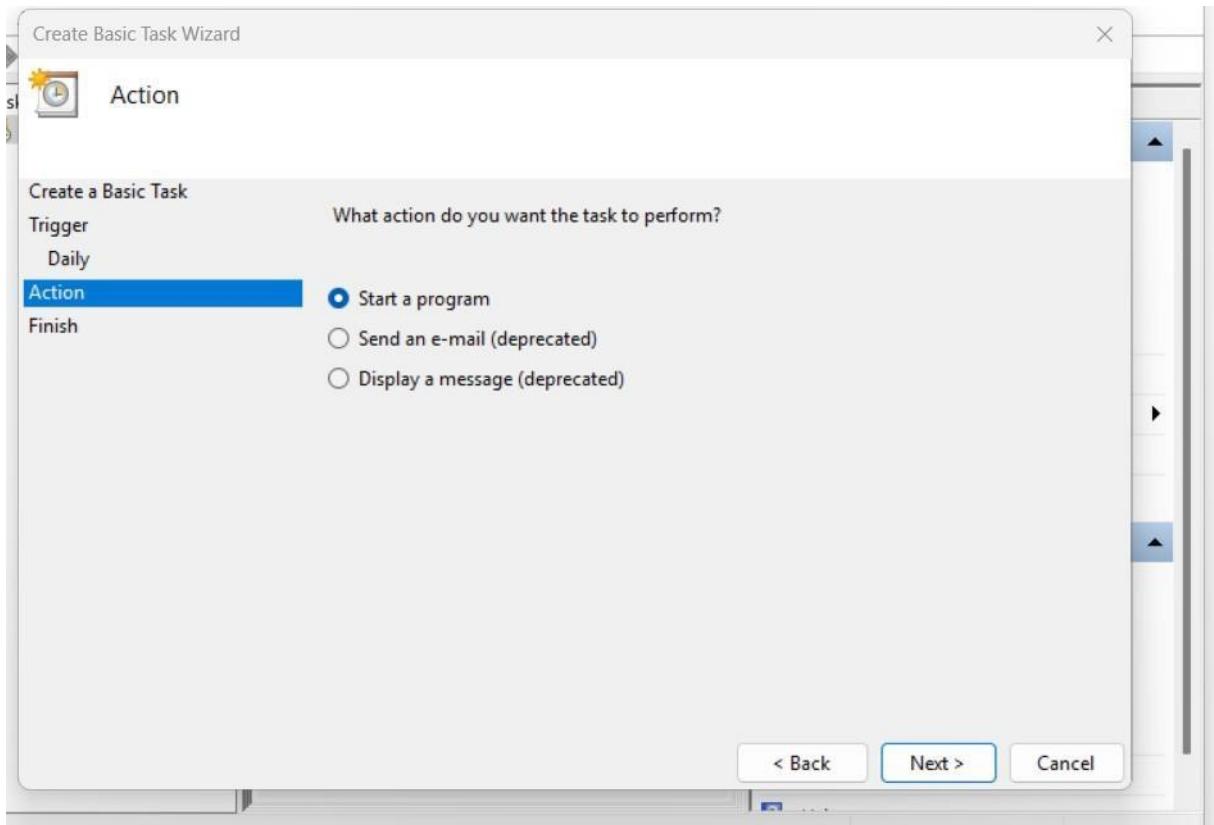
Step 6:

Set the Action

Now, we tell Task Scheduler what to do when it runs.

Select "Start a Program":

On the "Action" screen, select the option "Start a Program" and click Next.



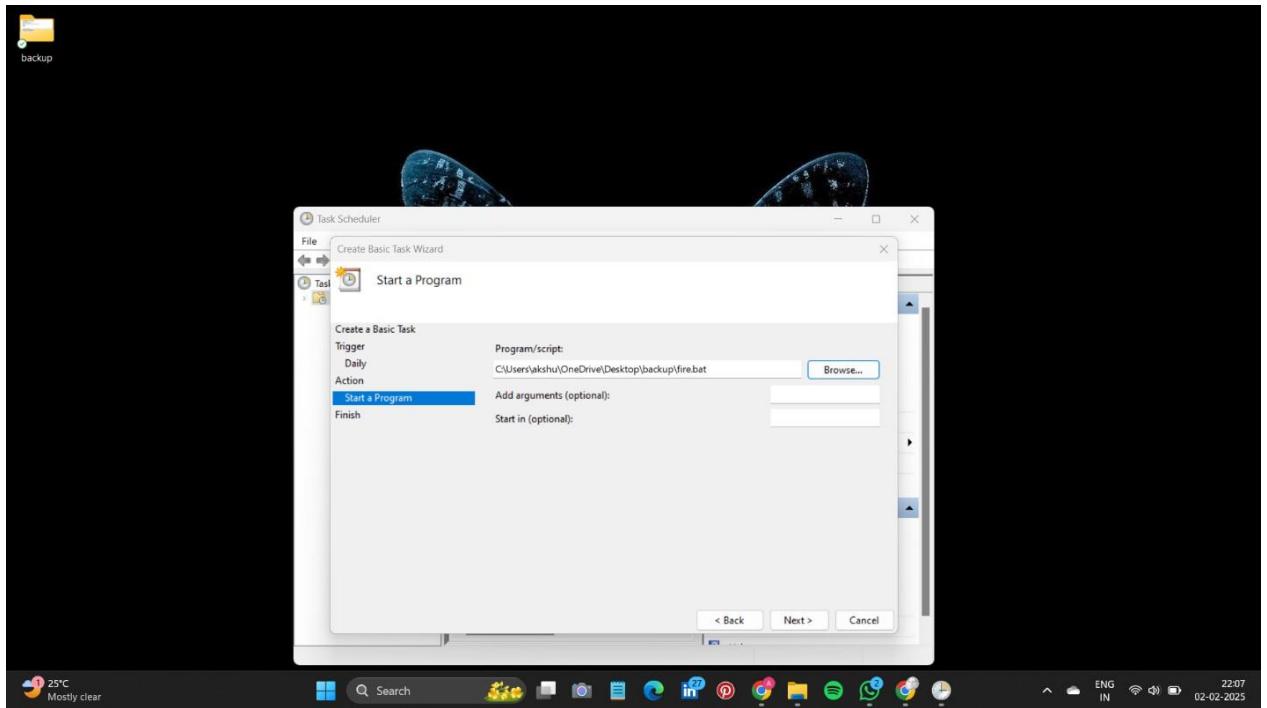
Step 7:

Point to the Program or Script:

In the Program/script field, click **Browse** and navigate to the location of your .bat file.

Example: If your script is named backup.bat and saved on the desktop, navigate to that file and select it.

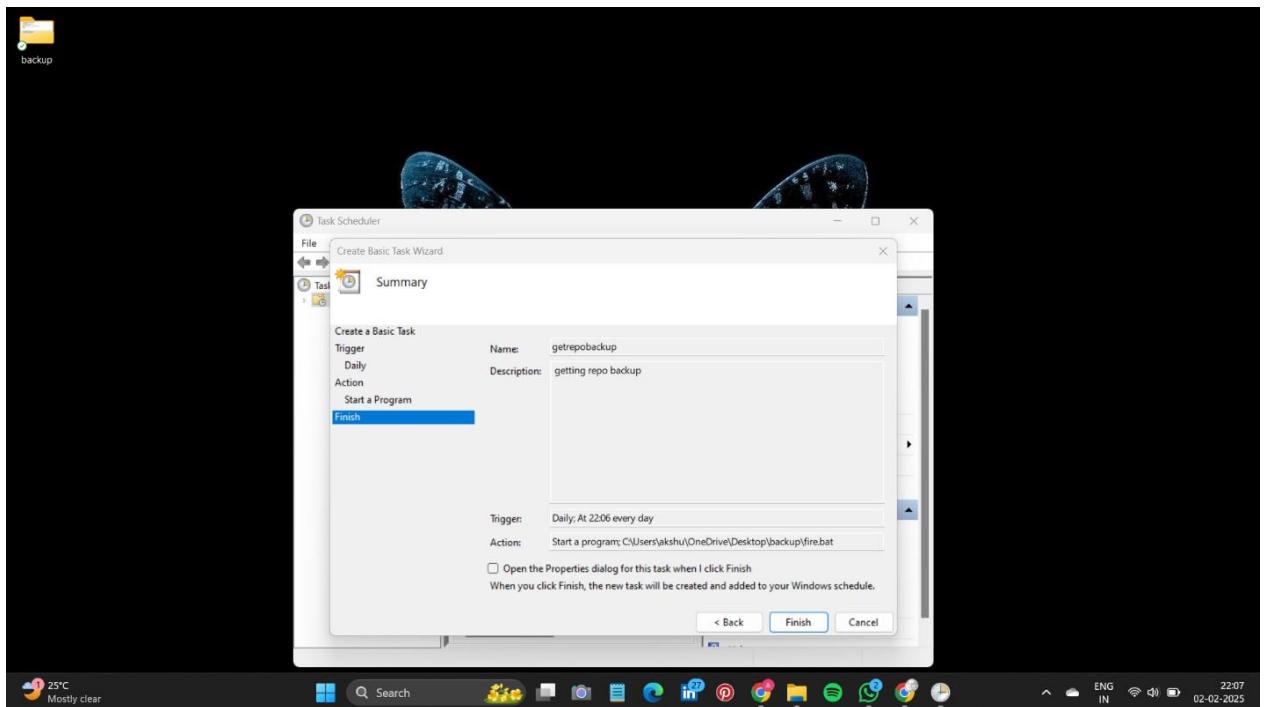
Click Next.



Step 8:

Review and Finish

Click **Finish** to save and schedule the task.



Step 9:

In Task Scheduler, go to the **Task Scheduler Library** (on the left-hand side).

```
Finding matching tasks... (it could take a few moments)
+ | v
Cloning repository for the first time...
Cloning into 'repo'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 3 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (3/3), done.
Creating a compressed backup: repo-backup-2025-01-24.zip
Backup complete: repo-backup-2025-01-24.zip

[process exited with code 0 (0x00000000)]
You can now close this terminal with Ctrl+D, or press Enter to restart.
|
```

Step 10:

Now u can see the folder which you created (GitHub Backup Folder) in the first step will now contains the files which is in your repository.

