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:Nantha Krishnan.G.K.
Department:AML

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

Introduction

Backing up your Git repositories is essential for effective version control management. It ensures the security and accessibility of your work, even in cases of unexpected data loss, such as accidental deletions, hardware failures, or repository corruption. Automating this backup process not only saves time but also minimizes manual effort, ensuring that your repositories are consistently backed up with regular updates.

Overview

This task involves setting up an automated backup system for your Git repository on a Windows machine using a batch script and Task Scheduler. The batch script will pull the latest changes from your repository every day and store them in a designated backup directory. It will also create a timestamped archive of the repository for organized backups. This process ensures that both your codebase and version history are securely stored locally for safekeeping.

Objectives

1. Automate Backups: Develop a script to back up the entire Git repository daily.

2. Minimize Data Loss: Safeguard the repository from accidental deletions or hardware failures.

3. Ease of Management: Create timestamped backups for quick identification and restoration.

4. Hands-Free Automation: Leverage Task Scheduler to eliminate the need for manual executio

Importance

1. Disaster Recovery: In case of repository failures or accidental deletions, you can quickly restore your work from the local backup.

2. Version History Preservation: All changes and version history are secured, ensuring no progress is lost.

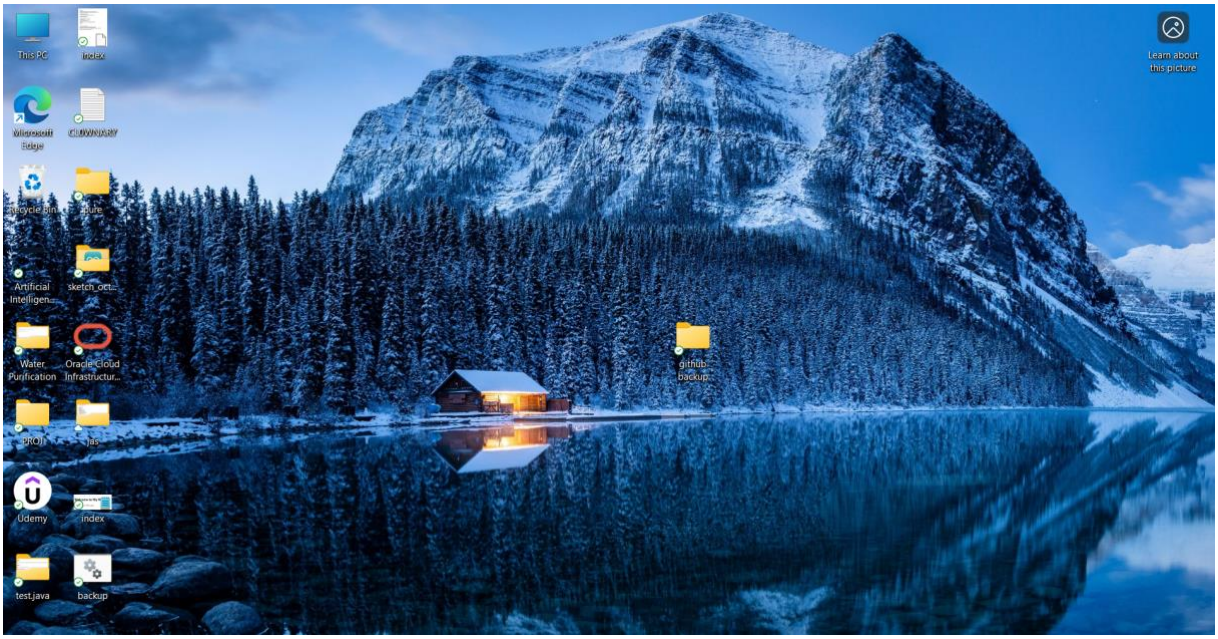
3. Efficient Workflow: Automating the process allows you to focus on development tasks instead of managing backups manually.

4. Organization: Timestamped backups provide a clear, structured way to keep track of changes over time

Step-by-Step Overview

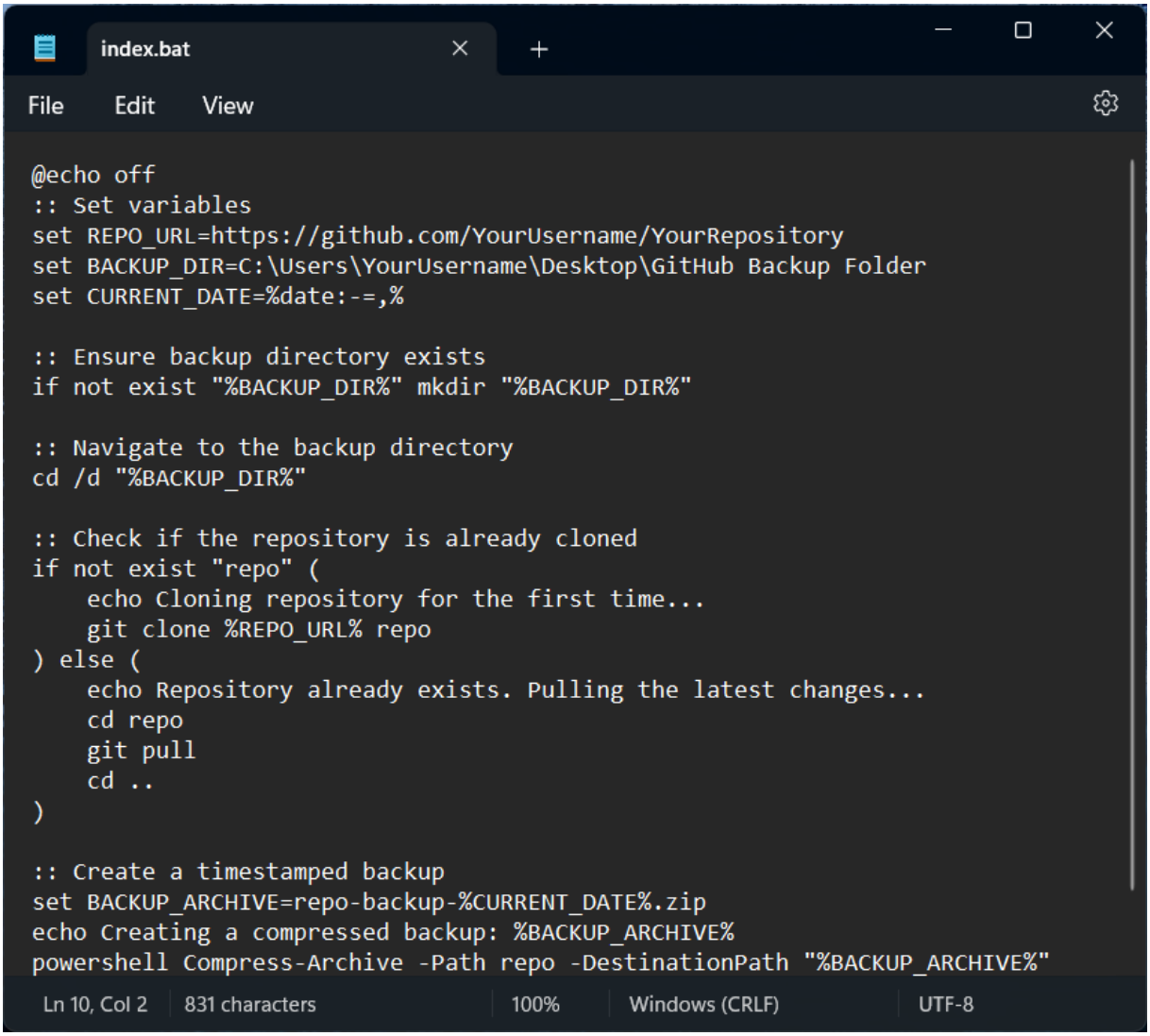
Step 1:

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) in Desktop



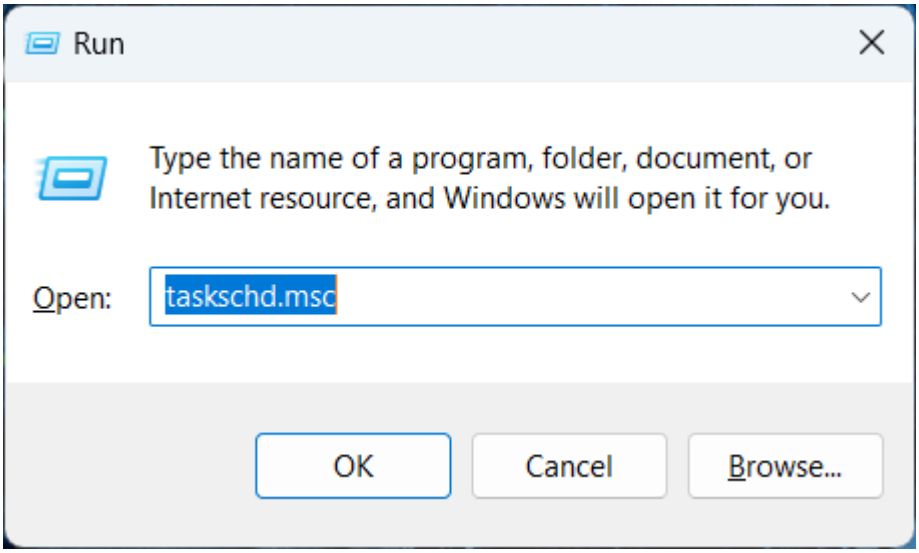
Step 3:

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.



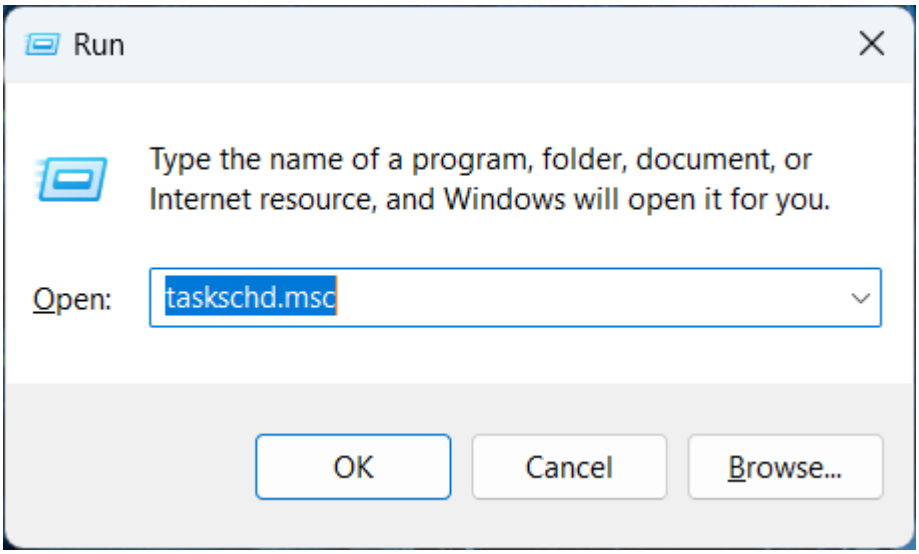
Step 3:

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.

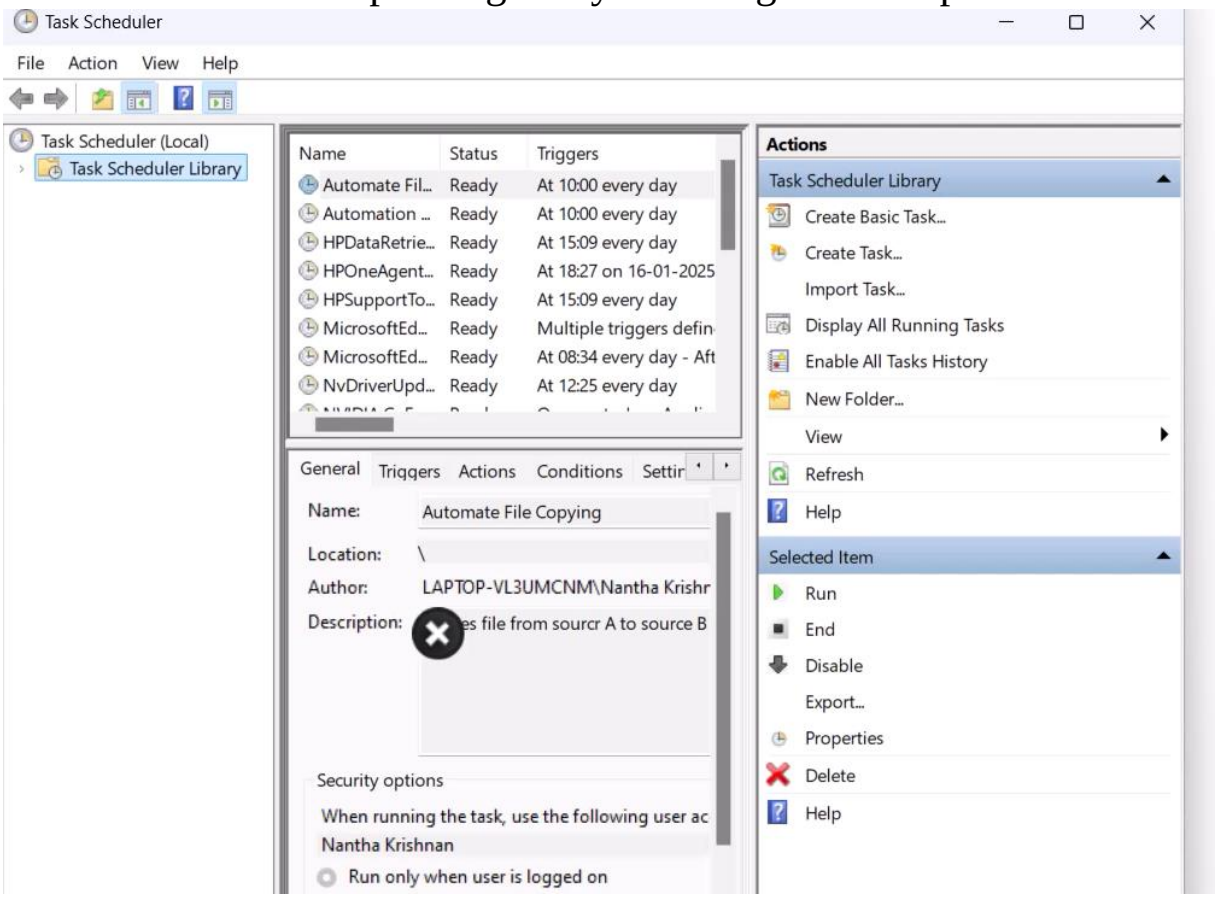


Step 4:

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.



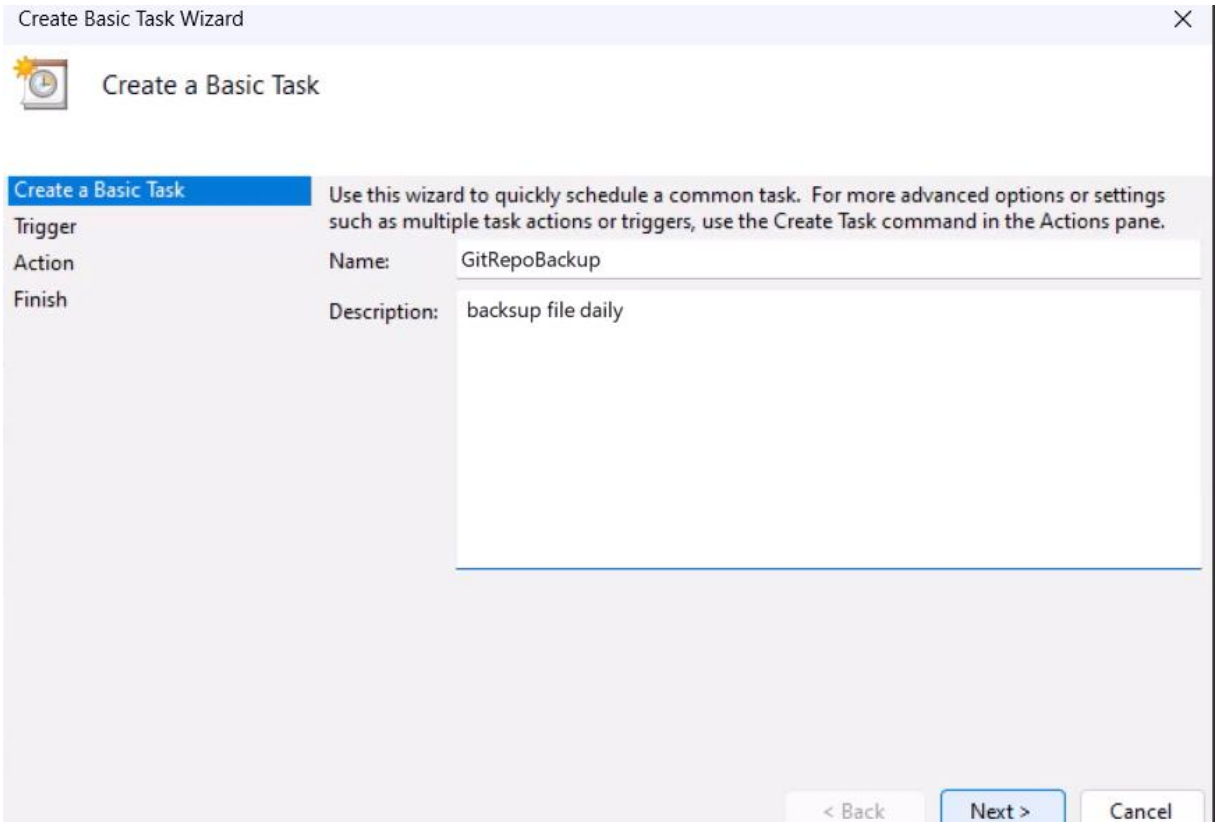
Step 5:

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 "Backsup files daily".

2. Click Next to continue.



Step 6:

Choose a Schedule:

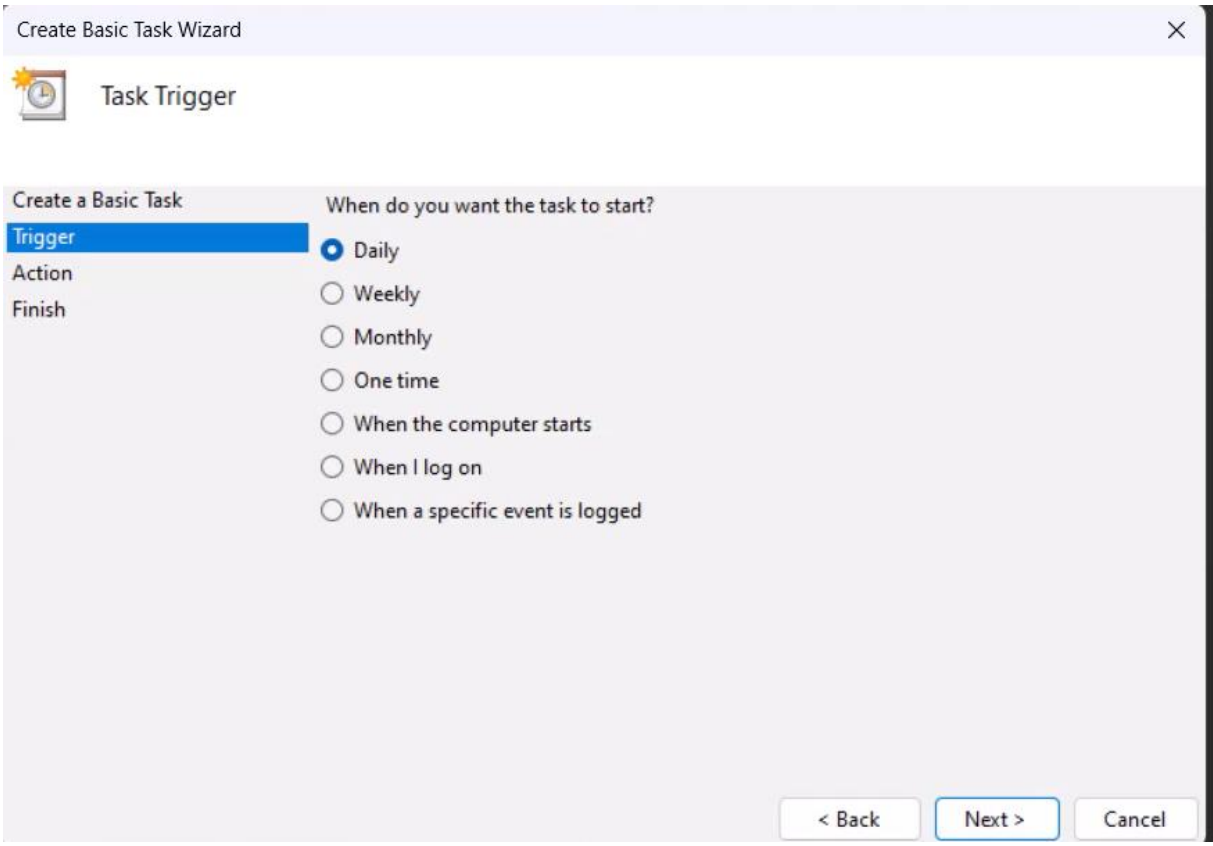
You will see options like:

Daily (runs every day).

Weekly (runs once a week).

One time (runs only once at a specific time).

Choose what works for you (e.g., Daily) and click Next.



Step 7:

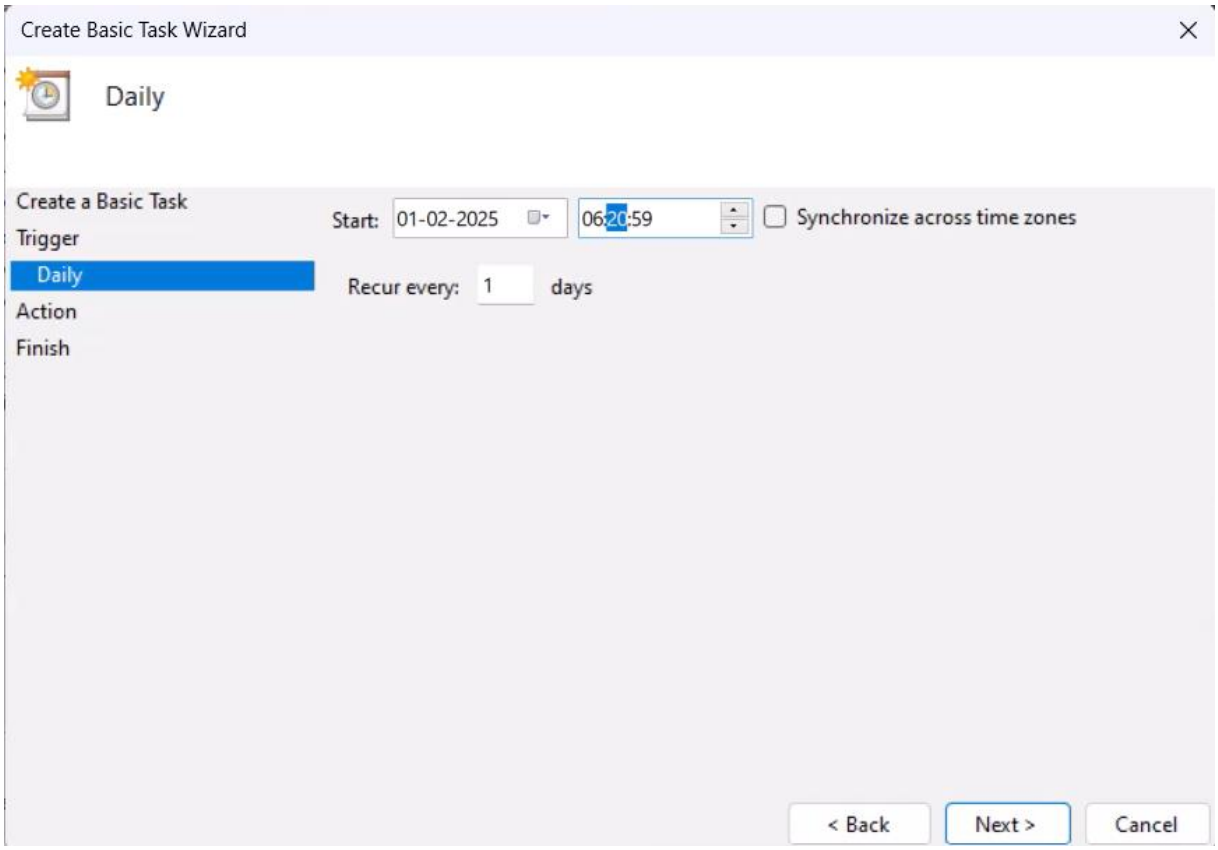
Set the Time and Frequency:

If you chose Daily, specify:

The start date (it defaults to today).

The time (e.g., 7.00 PM). Click

Next to move on.



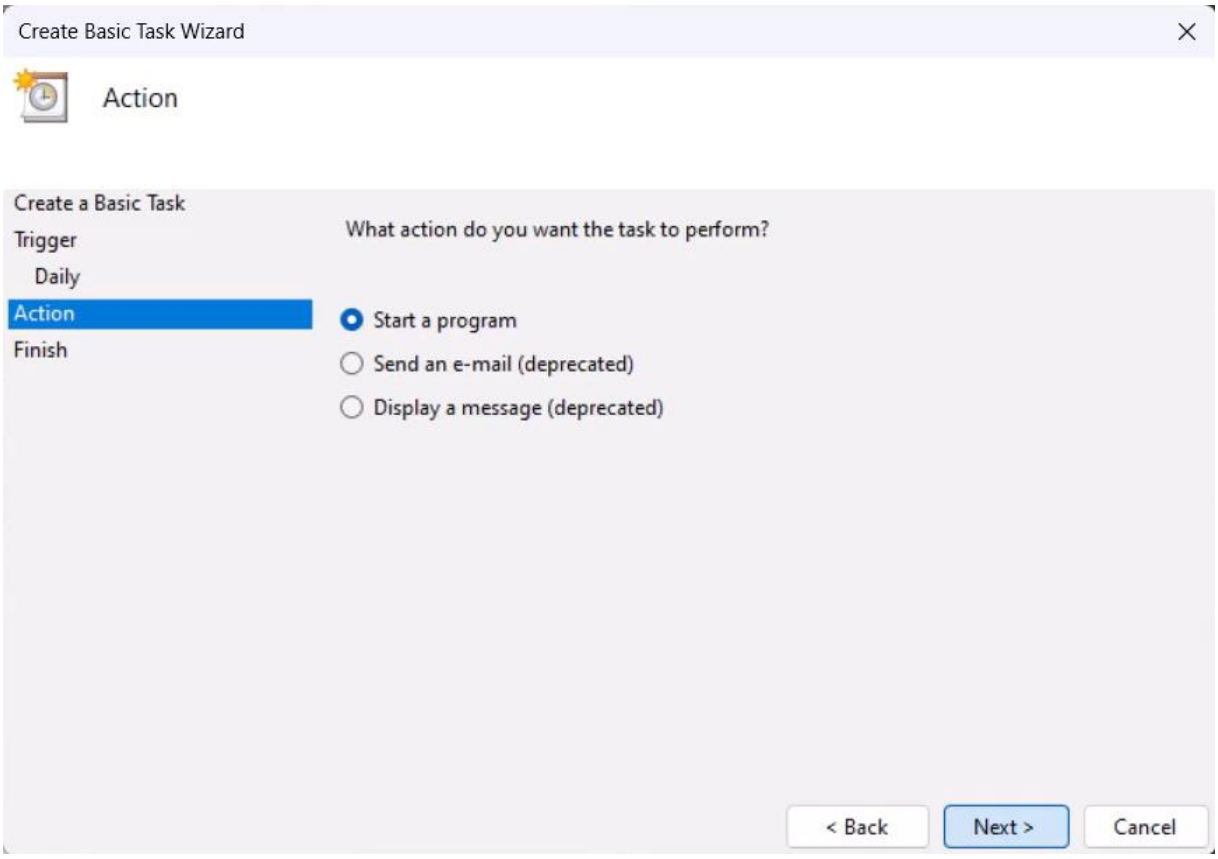
Step 8:

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.



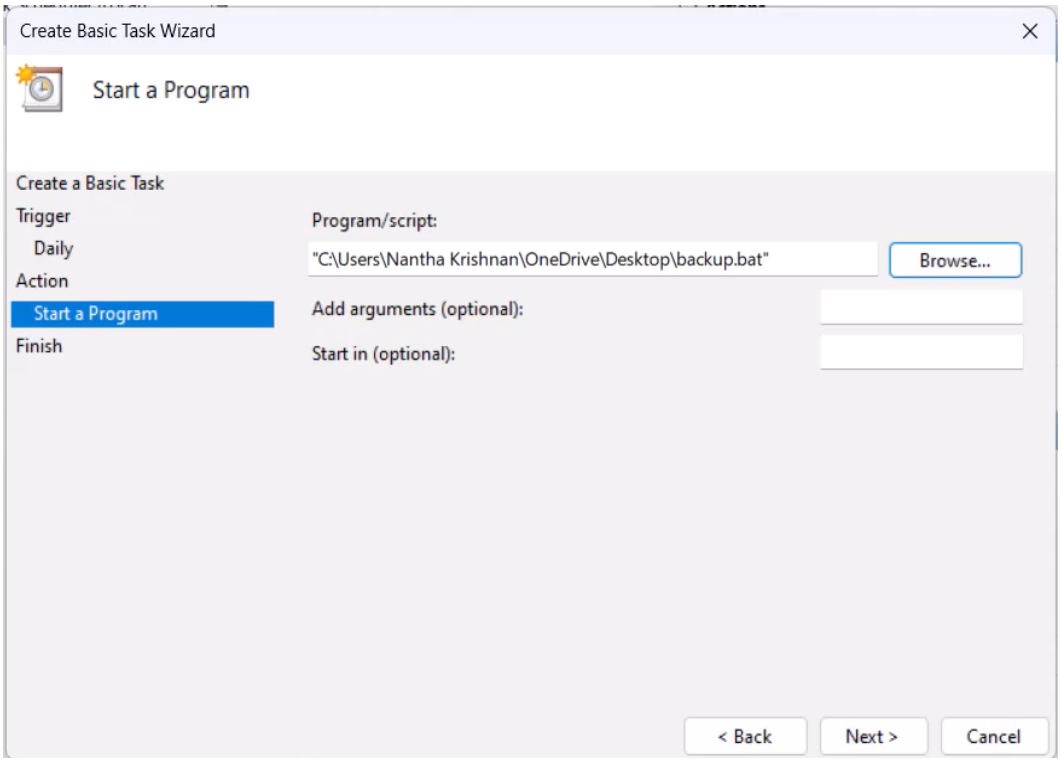
Step 9:

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.

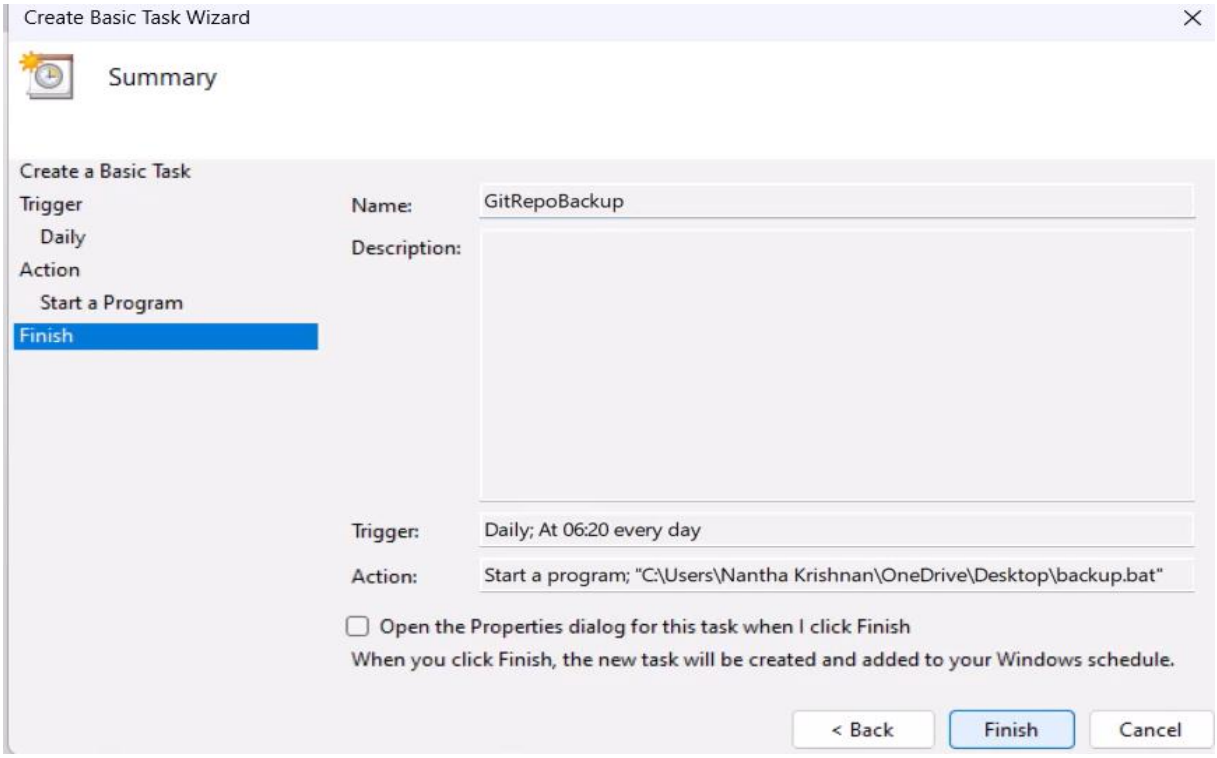
Click Next.



Step 10:

Review and Finish

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



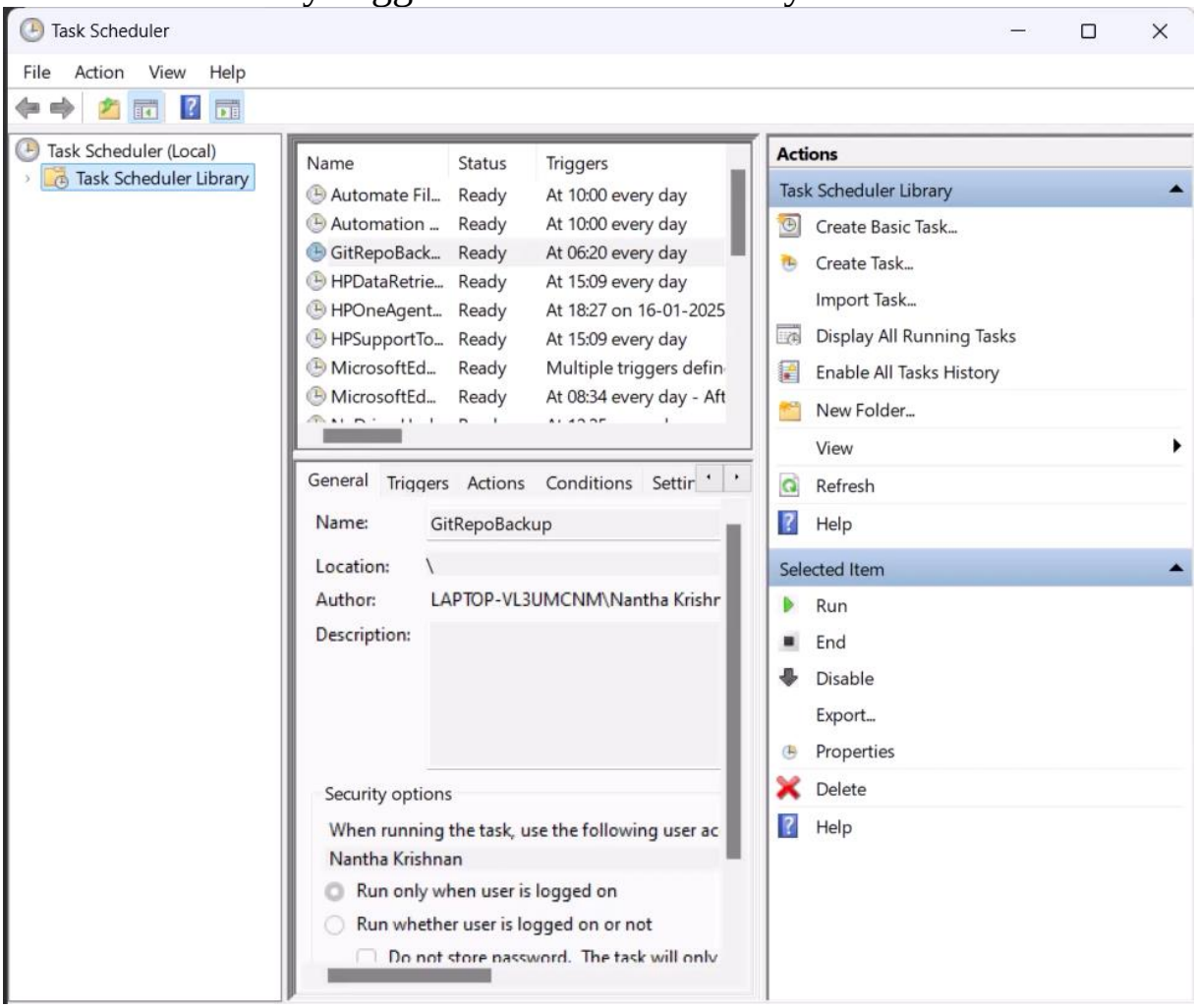
Step 11:

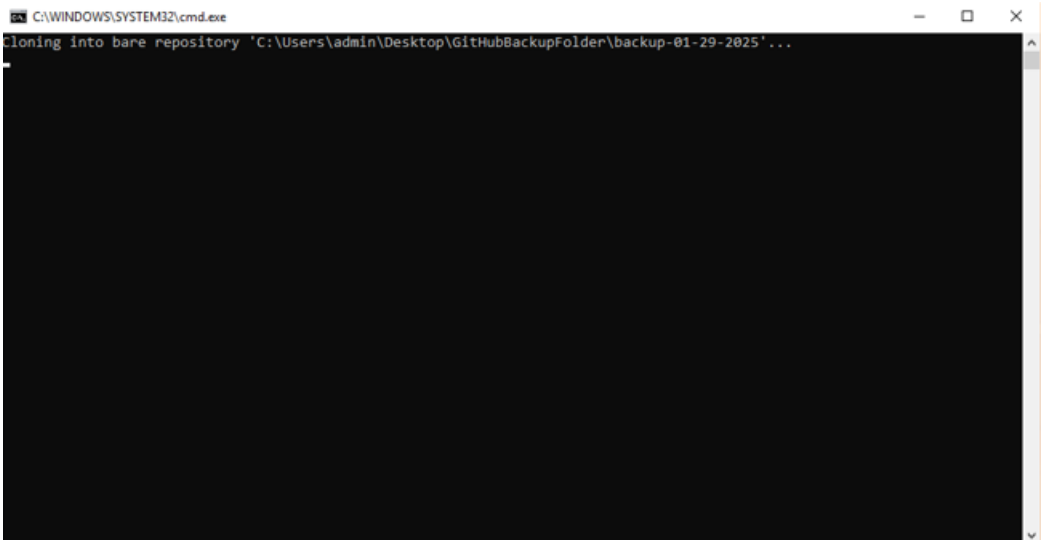
In Task Scheduler, go to the **Task Scheduler Library** (on the left-hand side).

Find your task (it should have the name you gave it, e.g., "GitRepoBackup").

Right-click the task and select **Run**.

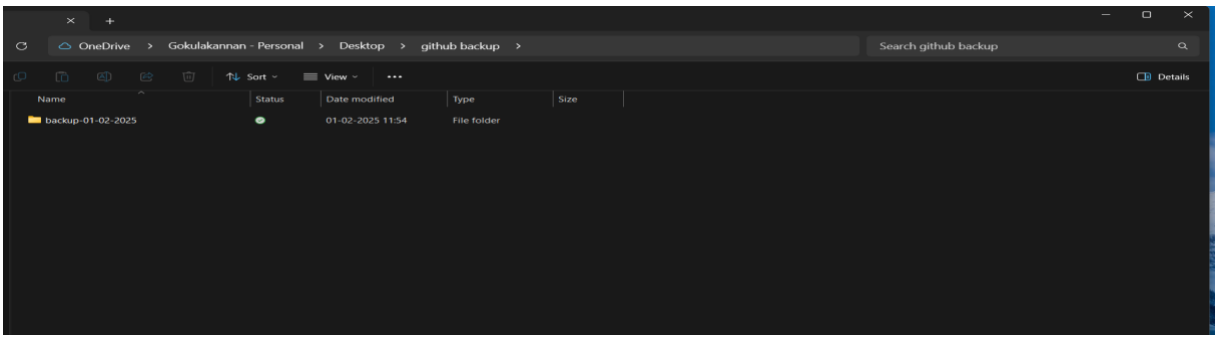
This will manually trigger the task immediately





Step 12:

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.



Outcomes

1. **Automated Git Repository Backups**
   - Successfully automated the process of creating daily backups for Git repositories, ensuring all updates and changes are securely stored in a local folder.
2. **Proficiency in Batch Scripting**
   - Gained the ability to create and execute .bat scripts for automating tasks, including cloning, pulling updates, and compressing Git repositories into timestamped backup archives.
3. **Experience with Task Scheduler**
   - Developed practical knowledge of Task Scheduler for automating repetitive tasks, including setting up triggers, defining actions, and configuring conditions on a Windows system.