**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: Vijaya Nandana M Department: CSE



**Introduction**

Automating repetitive tasks like file copying can save time, reduce errors, and improve efficiency in various scenarios, such as managing backups, organizing files, or syncing data between directories. This POC script will demonstrate how to copy files automatically from one folder to another, using a simple yet effective approach.

**Objectives**

The key objectives of this automation task are:

**Simplify File Management**: Reduce manual effort in moving or copying files between folders.

**Improve Efficiency**: Automate the process to save time and improve productivity.

**Ensure Data Integrity**: Accurately copy files without any corruption or loss.

**Customizable Workflow**: Enable flexibility to meet specific requirements (e.g., filtering files by type or date).

**Scalable Solution**: Allow the script to handle a large number of files efficiently.

**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.

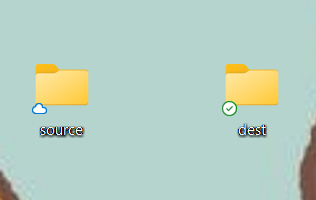
**4. Useful in Various Scenarios**: This approach is valuable for IT professionals, businesses, and individuals dealing with frequent file transfers or backups.

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

**Step-by-Step Overview**

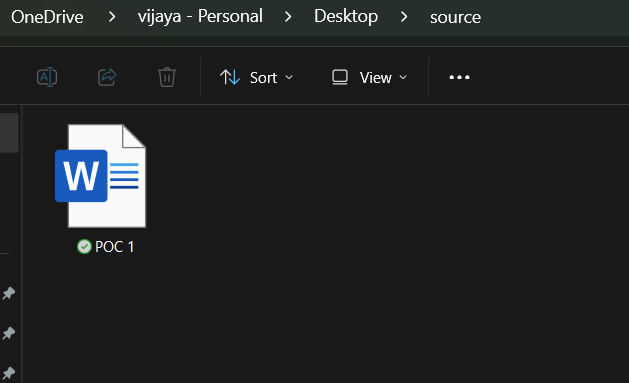
Step 1

Create two folders named Source and Destination



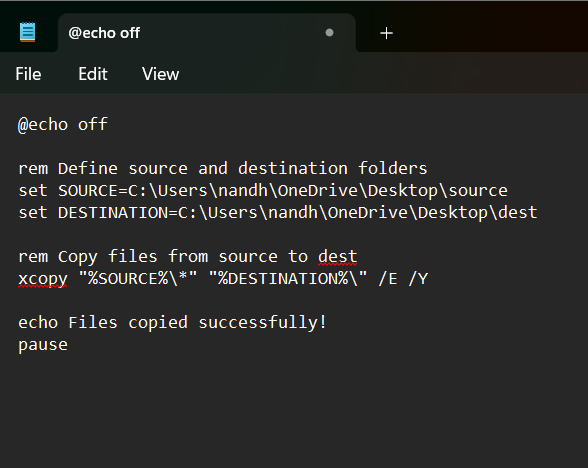
Step

Store some files inside it to automate copying.



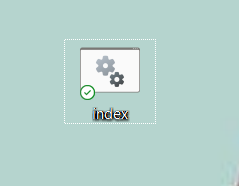
Step 3

Open the note pad and type the code and make sure that in set SOURCE give your Source folder address and in set DESTINATION give your Destination Folder address.



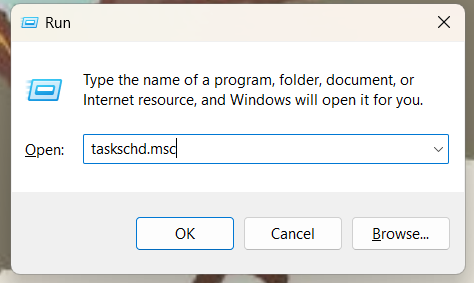
Step 4

Then save the file in desktop with .bat extension (eg : index.bat) so the file looks like this.



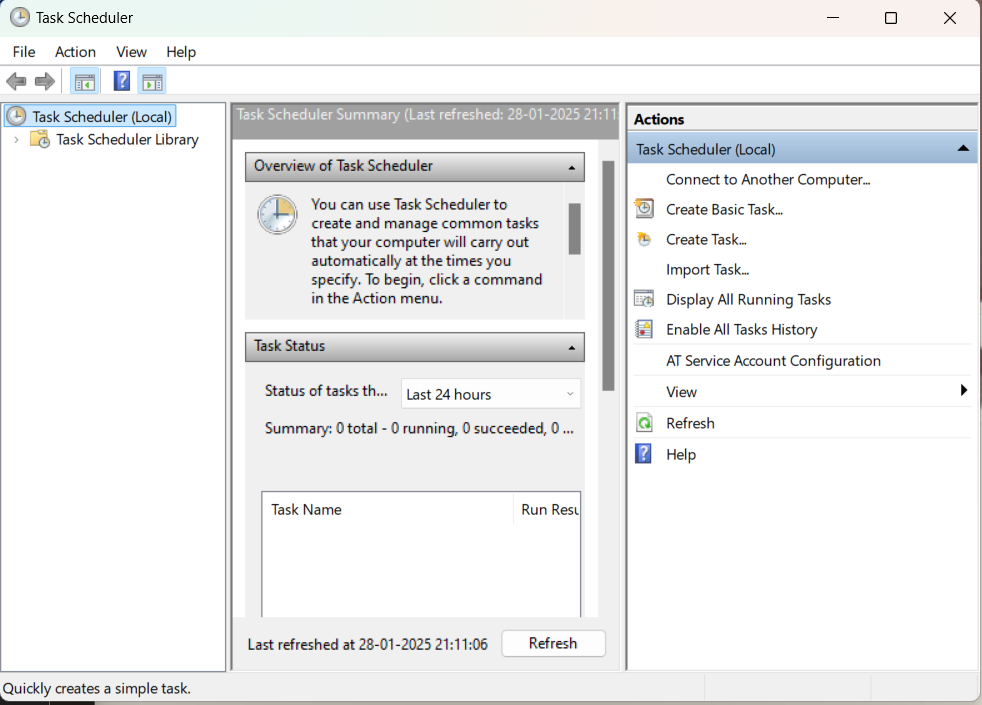
Step 5

* 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.



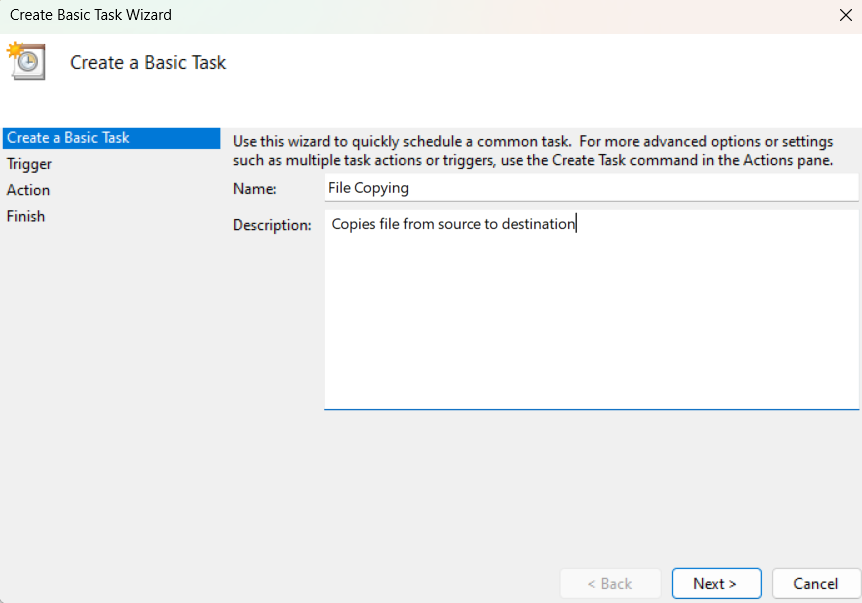
Step 6

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 7

Enter a Name for the Task and click Next to continue.



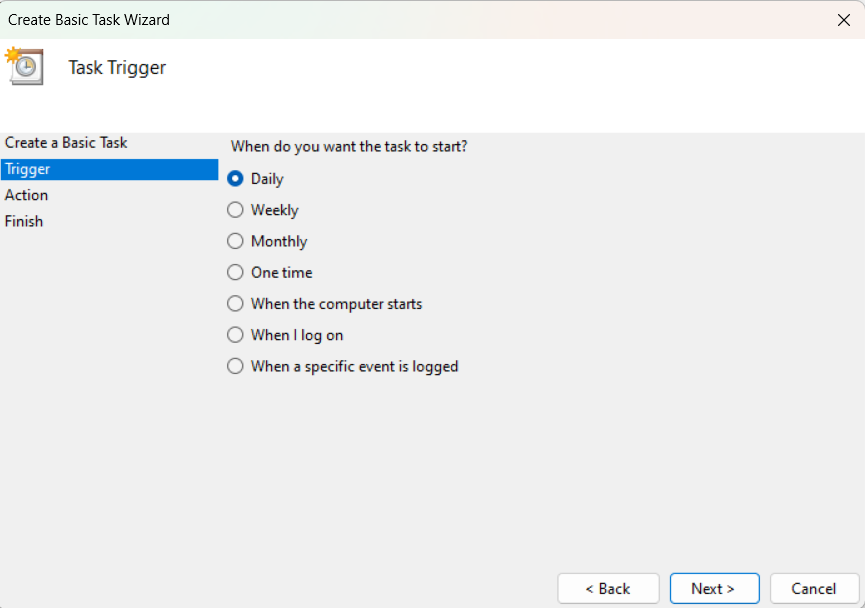
Step 8

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)

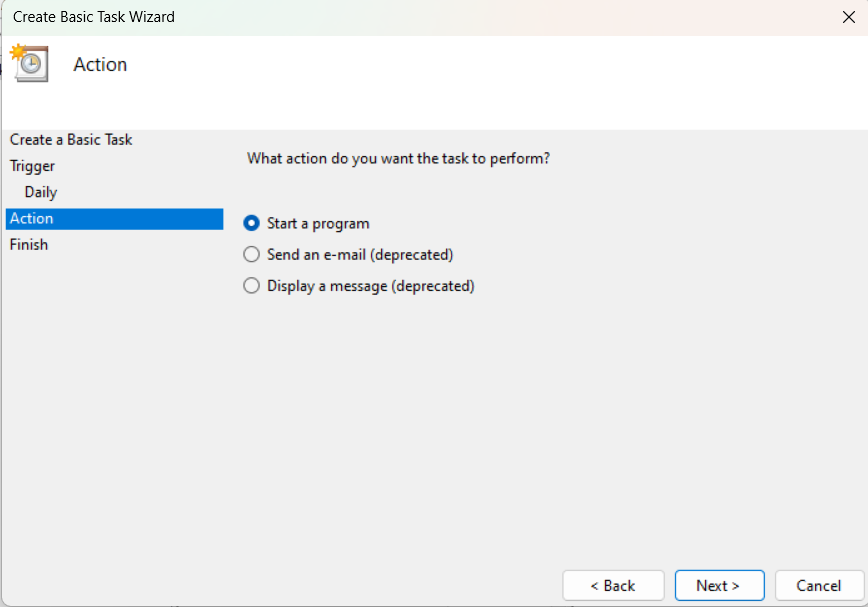
Choose what works for you (e.g., Daily) and click Next.



Step 9

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

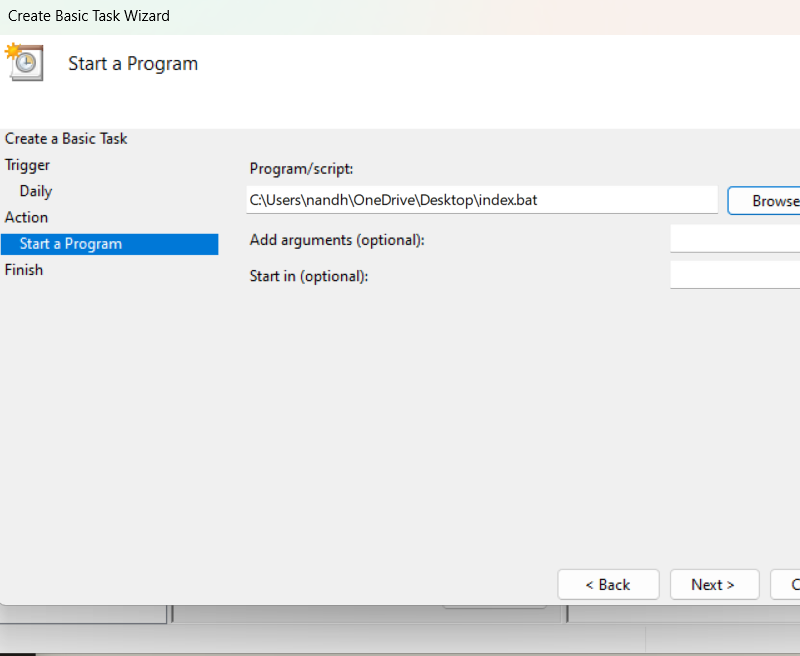
On the "Action" screen, select the option **"**Start a Program**"** and click Next.



Step 10

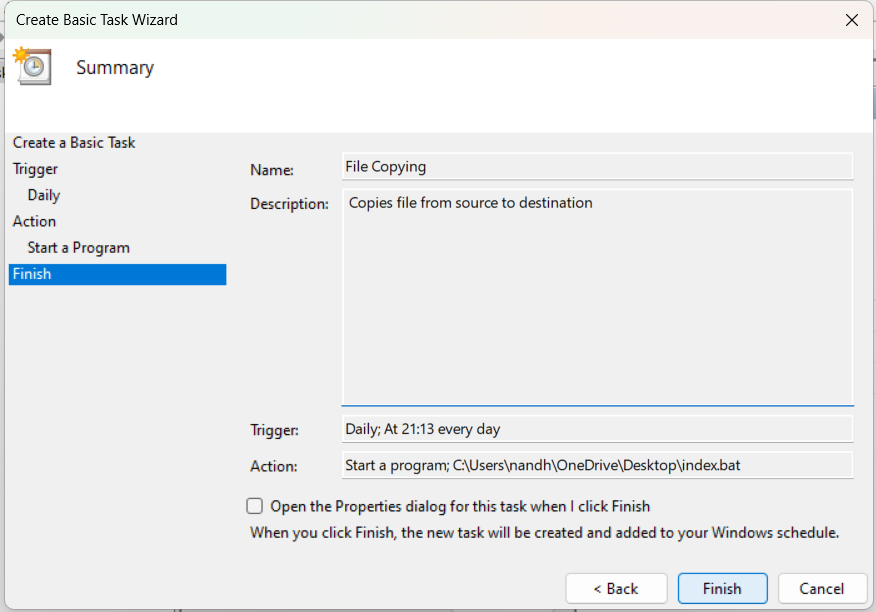
Point to the Program or Script:

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



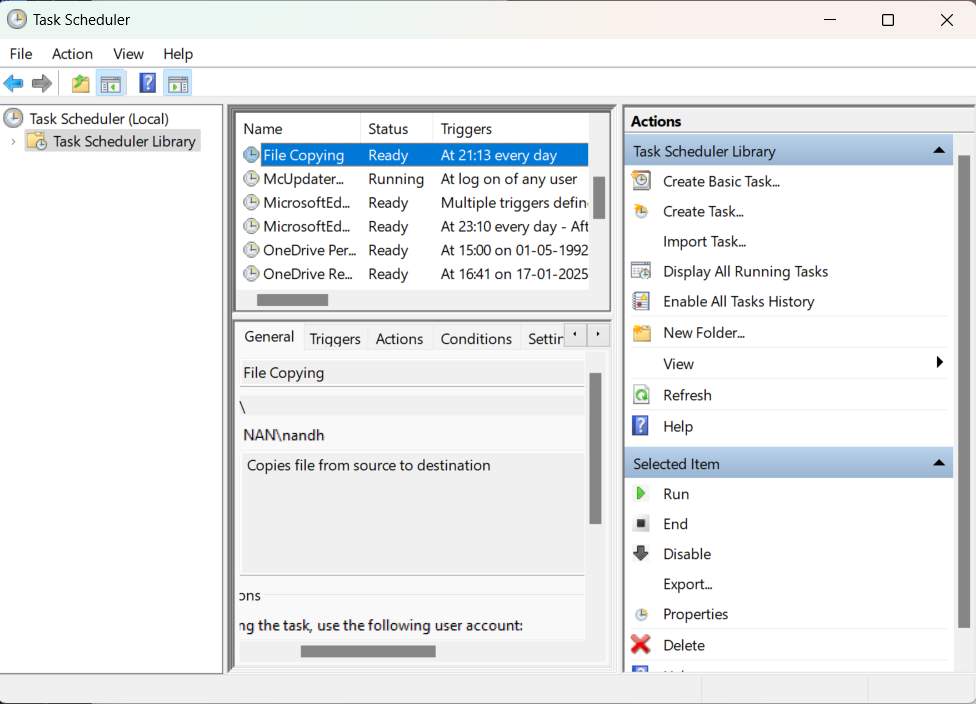
Step 11

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



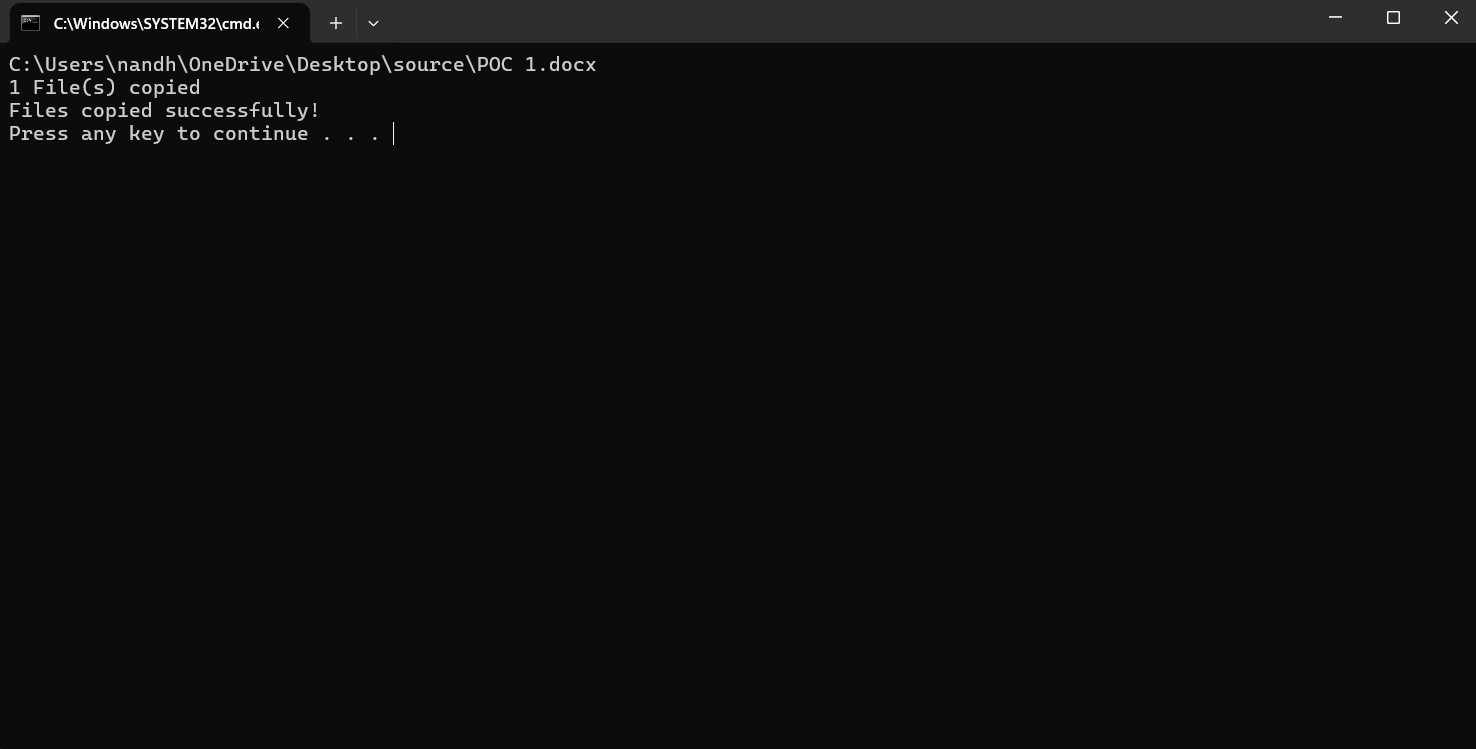
Step 12

* In Task Scheduler, go to the **Task Scheduler Library** (on the left-hand side).
* Find your task
* Right-click the task and select **Run**.



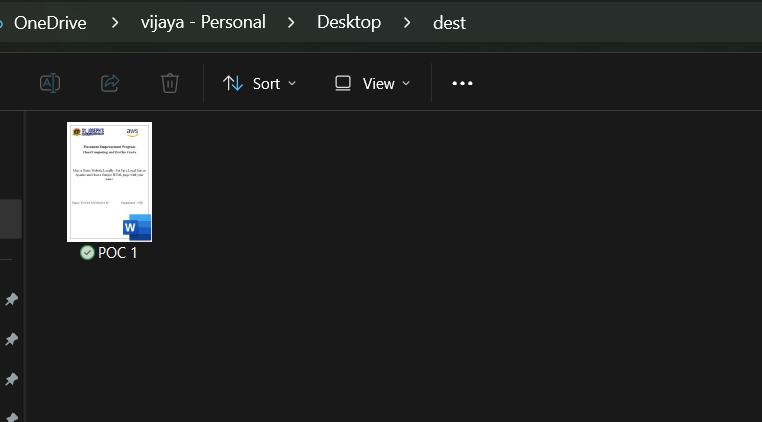
Step 13

This will manually trigger the task immediately.



Step 14

Since we set up the task to copy files from source file to dest file, it was successfully automated.



**Outcome**

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. Enhance your workflow automation skills with practical hands-on experience.