**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: Paripurani S P Department: IT



### Introduction

Repetitive tasks like file copying can be time-consuming, error-prone, and inefficient when done manually. Automating this process offers a practical solution for managing backups, organizing files, and syncing data between directories. doc script demonstrates a straightforward yet powerful method to automate file transfers from one folder to another.

### ****Overview****

### The script simplifies the process of transferring files between directories by automating the task. It ensures all files in the source directory are efficiently copied to the destination folder while maintaining data integrity. The approach is versatile and can be tailored to meet specific needs, such as filtering files by type, size, or modification date. Depending on the environment, the script can be implemented using programming languages like Python, Shell scripting, or PowerShell.

**Objectives**

The key objectives of this automation task are:

**Streamlining File Management:** Minimize manual effort in file transfers.

**Boosting Efficiency:** Automate repetitive tasks to save time and increase productivity.

**Ensuring Data Integrity:** Accurately copy files without data corruption or loss.

**Customizing Workflows:** Allow for flexibility in filtering or organizing files based on specific criteria (e.g., file type, date).

**Enabling Scalability**: Handle large volumes of files or more complex workflows effortlessly.

**Importance**

**Time Savings**: Automation eliminates the need for manual interventions, saving hours of effort.

**Error Prevention:** Reduces risks like skipping files or overwriting important data.

**Enhanced Organization**: Maintains consistent file storage structures and ensures seamless backups.

**Wide Applicability:** Beneficial for IT professionals, businesses, and individuals managing frequent file transfers or backups.

**Scalable and Reusable Solution:** A well-designed script can be easily adapted for other scenarios or scaled up for larger tasks.

**Step-by-Step Overview**

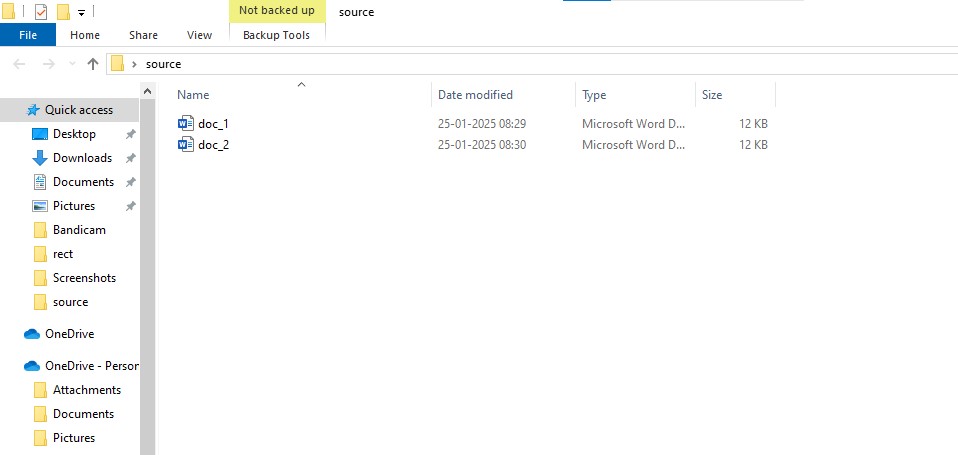
Step 1

Create two folders named Source and Destination



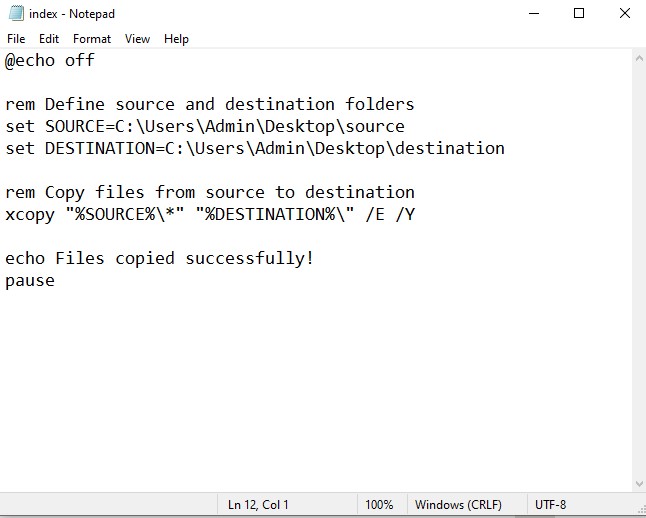
Step 2

Store some files inside source file.



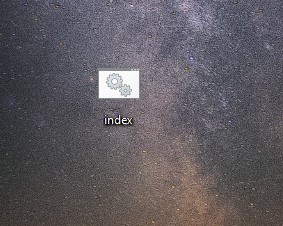
Step 3

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



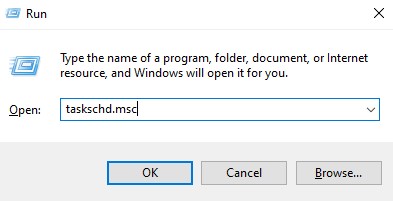
Step 5

In taskbar search box search as Run or Press (**Window + R**)

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.

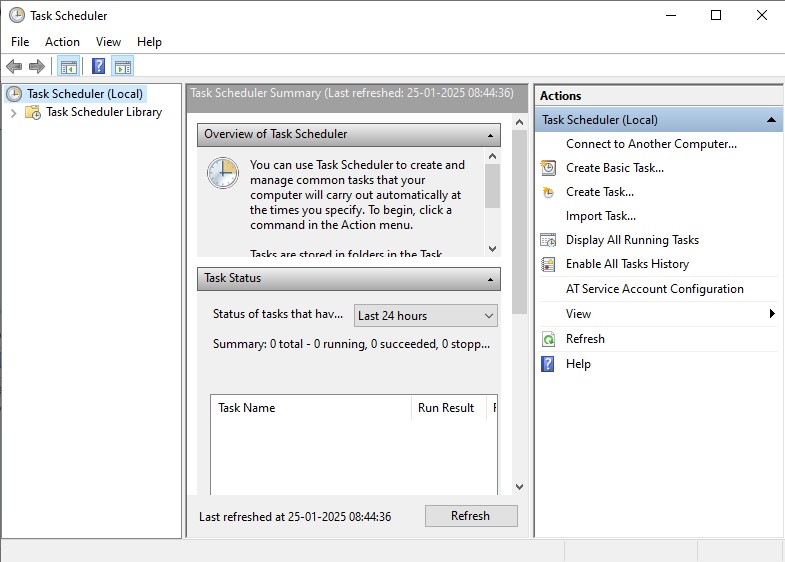


Step 6

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.



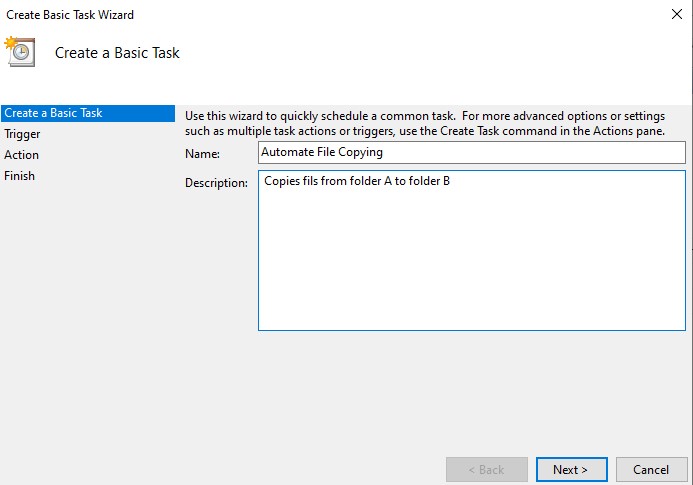
Step 7

1. Enter a Name for the Task:

For example: "Automate File Copying".

Optionally, you can add a description like "Copies files from folder A to folder B".

1. Click Next to continue.



Step 8

Choose a Schedule:

You will see options like:

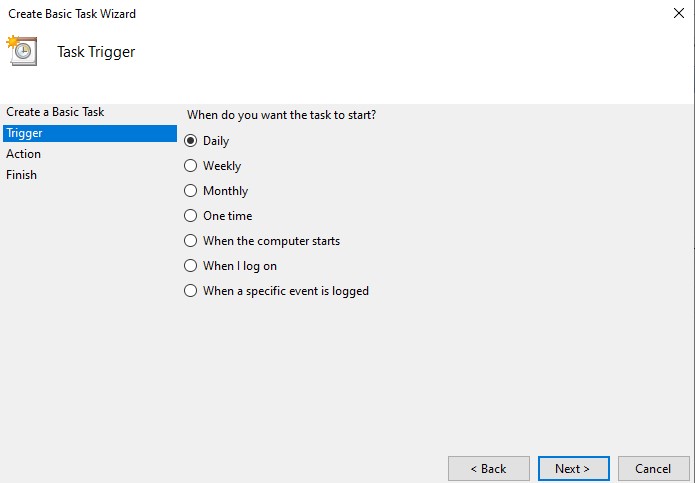
Daily

Weekly

Monthly

One time

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



Step 9

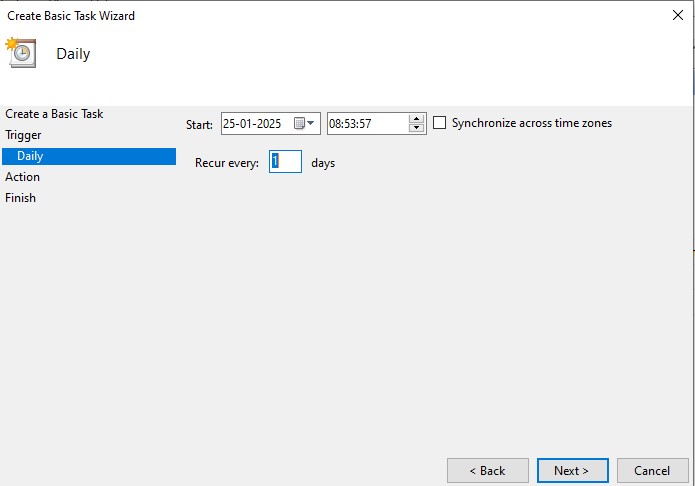
Set the Time and Frequency:

If you chose Daily, specify:

The start date (it defaults to today).

The time (e.g., 8:00 AM).

Click Next to move on.

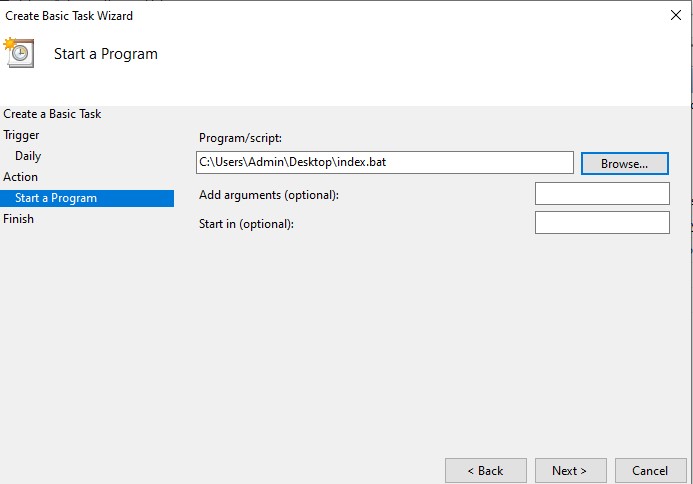


Step 10

Point to the Program or Script:

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

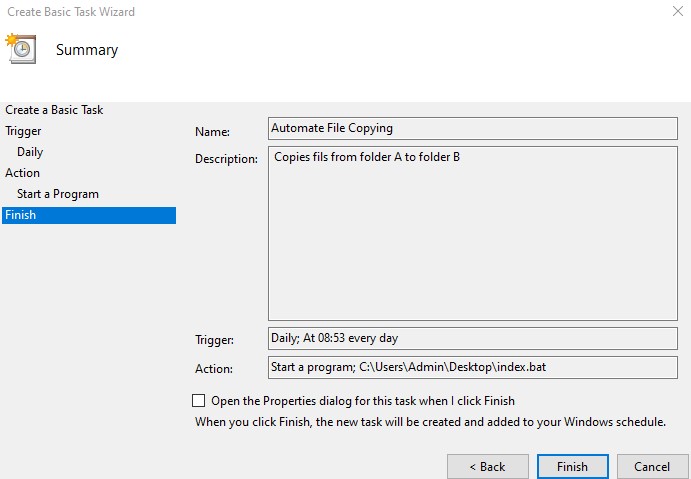
Click Next.



Step 11

Review and Finish

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



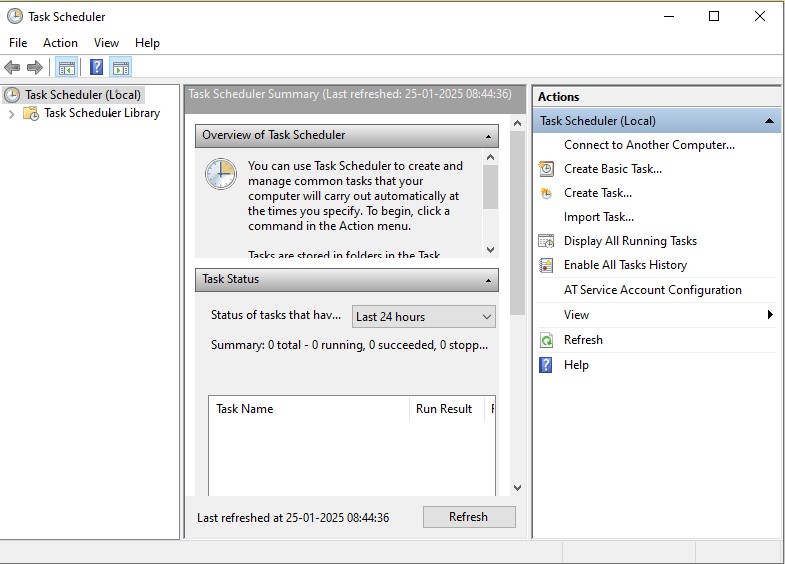
Step 13

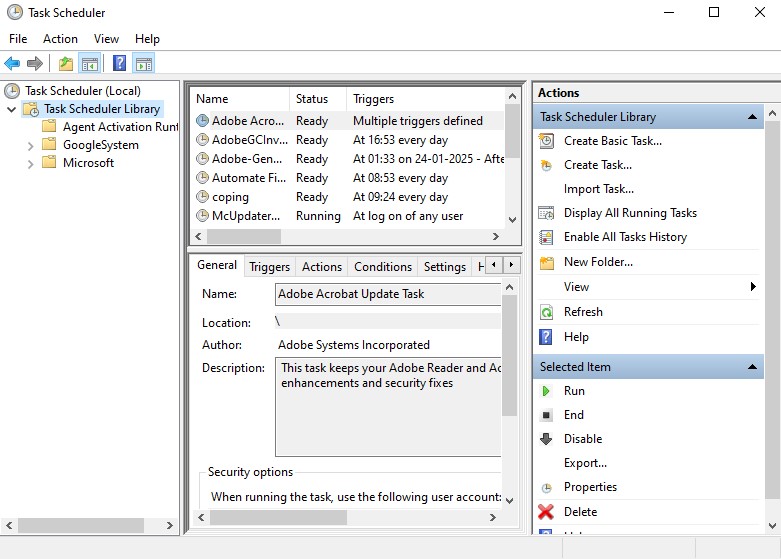
In Task Scheduler, go to the **Task Scheduler Library** (on the left-hand side).

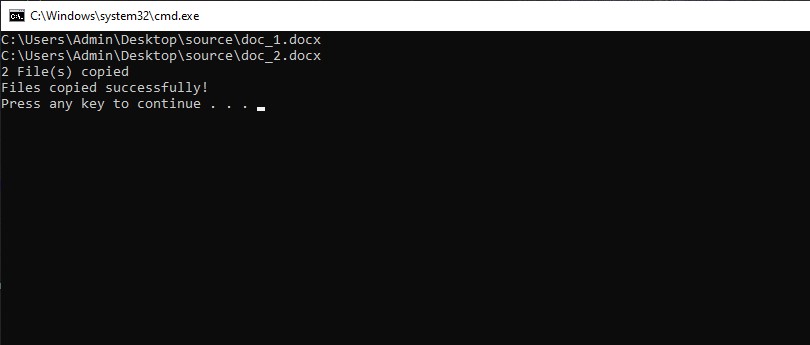
Find your task (it should have the name you gave it, e.g., "Automate File Copying").

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

This will manually trigger the task immediately

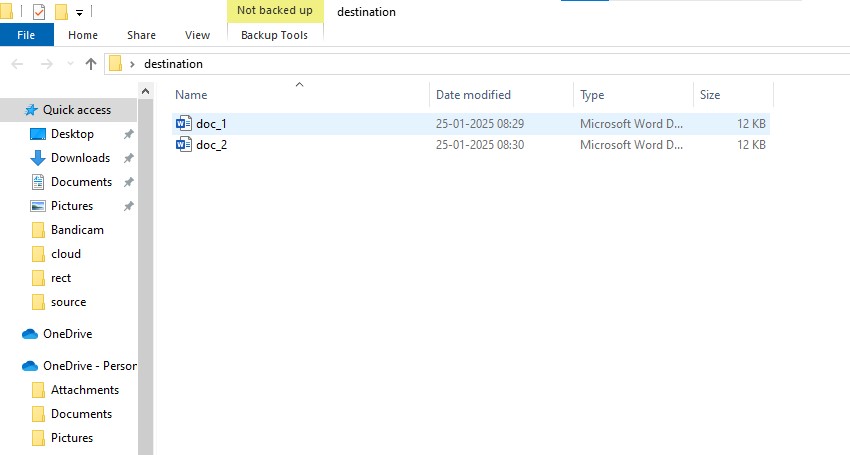






Step 14

If your task was set up to copy files, go to the destination folder and confirm that the files have been copied.



**Outcome**

By implementing this automated file transfer solution, the following outcomes can be achieved:

* Time spent on manual file copying is significantly reduced, enabling users to focus on more critical tasks.
* Automation ensures accurate and reliable file transfers, eliminating common human errors.
* A consistent and streamlined process for organizing files is established, improving overall efficiency.
* The script can be tailored to handle a variety of use cases, from simple tasks to complex workflows, and can scale up to manage large datasets.
* By maintaining data integrity, the risk of file corruption or loss during transfers is minimized.
* Once created, the script becomes a reusable asset for future tasks, saving time in the long term.