 

**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: Nanthini R Department: ADS



**Introduction**

Automating routine tasks, such as copying files, can significantly enhance productivity by saving valuable time, reducing errors, and ensuring more efficient processes. In this Proof of Concept (PoC), we’ll explore a simple yet effective way to automate the process of transferring files from one folder to another, demonstrating a practical solution for streamlining file management, backups, and data synchronization.

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

### This automation script is built to facilitate the seamless copying of files from a designated source folder to a target destination folder. It can be customized according to different requirements, such as filtering files by type, modification date, or size. Depending on your preference and system environment, you can implement the automation using various programming languages, such as Python, Shell scripting, or PowerShell.

**Objectives**

**Treamline File Management:** Automate the process of moving or copying files between folders, eliminating manual intervention.

**Boost Efficiency:** Automate repetitive tasks, saving time and optimizing workflows.

**Preserve Data Integrity:** Ensure files are transferred without corruption, preserving their original state.

**Customizable Workflow:** Tailor the script to specific needs, whether it's copying only certain file types, based on modification dates, or filtering by size.

**Scalable Solution:** Create a solution that works efficiently even with large volumes of files

.

**Importance**

**Saves Time:** By automating this task, you can bypass hours of manual copying, freeing up time for more important activities.

**Minimizes Errors:** Automation drastically reduces the chance of human mistakes, like missing files or accidentally overwriting important data.

**Ensures Consistency:** Keep your file organization structure organized and consistent, making backups and file management easier to handle.

**Wide Application:** This approach is invaluable to IT professionals, businesses, or anyone who frequently handles file transfers or backups.

**Reusable and Scalable:** Once the script is created, it can be reused for other tasks and scaled up as your file management needs grow.

**Step-by-Step Overview**

**Step1:**Create two folders on your system, labeled "Source" and "Destination."

**Step2:**Add a few files to the "Source" folder as sample data to automate the copying process.

**Step3:**Open a text editor such as Notepad, and write the automation code. Ensure that you specify the correct paths for both the SOURCE and DESTINATION directories within the script.

**Step4:**Save the script as a .bat file (e.g., index.bat) on your desktop

**Step5:**Press Win + R to open the "Run" dialog box.  
Type taskschd.msc and press Enter. This will launch the Task Scheduler window.

**Step6:**In Task Scheduler, look to the right-hand side and click on Create Basic Task to begin the setup process.

**Step 7:**

* Task Name: Give your task a descriptive name, such as "Automate File Copying." Optionally, you can add a description like "Copies files from Source to Destination."
* Click Next to continue.

**Step8:**Choose a schedule for the task to run:

* Daily: Runs every day at a specified time.
* Weekly: Runs once a week.
* One-time: Runs only once at a specific time.

Select the schedule that fits your needs (e.g., Daily) and click Next.

**Step9:**Set the desired time and frequency. If you chose Daily, you’ll need to specify the start date and time (e.g., 10:00 AM). Click Next once you’ve configured the schedule.

**Step10:**Choose the action you want the task to perform.

* Select Start a Program and then click Next.

**Step 11:**

* Program or Script: Click Browse to locate and select your .bat file (e.g., index.bat) that you saved earlier.
* Click Next to proceed.

**Step12:**Review the task details to ensure everything is correct. Once satisfied, click Finish to save and schedule the task.

**Step13:**In the Task Scheduler window, navigate to Task Scheduler Library on the left-hand side. Locate the task you just created (e.g., "Automate File Copying"), right-click it, and select Run. This will trigger the task to execute immediately.

**Step14:**Check the Destination folder to verify that the files have been successfully copied from the Source folder.

**Outcome**

Successfully set up a scheduled task to run automatically according to the specified timing or on demand.

Automated the process of copying files from one folder to another using a batch script.

Gained insight into how Task Scheduler works, including setting triggers, defining actions, and managing tasks.

Saved time by eliminating manual file operations, ensuring that tasks are performed reliably and on time.

Had the opportunity to test and verify that the automation works as expected, ensuring consistency and reliability in your file management.

Gained hands-on experience with automating tasks, enhancing your ability to manage workflows more effectively.