**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: Samiya Afreen J 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.

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

The script automates the process of copying files from a source folder to a destination folder. This can be customized to copy specific file types, based on modification date, or other filters. The process will utilize a Shell Script and the cronscheduler to ensure that the task is automated on the Mac system.

**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.

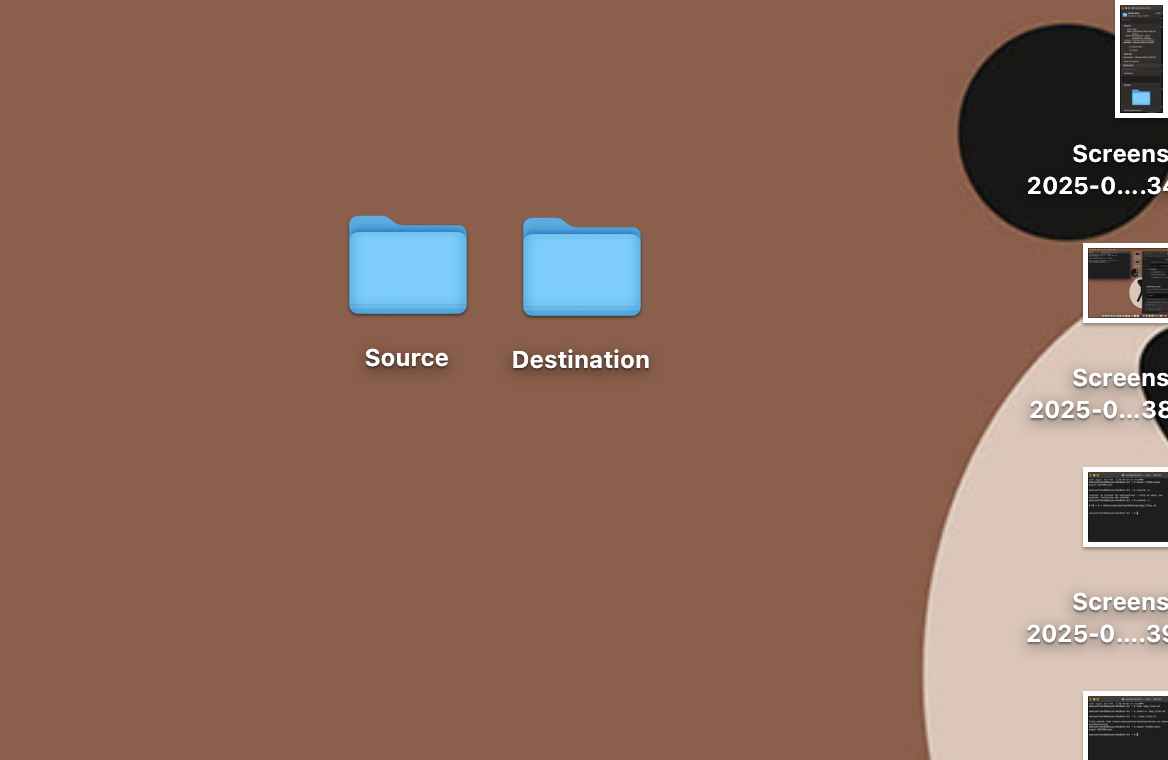
**4. Useful in Various Scenarios**: This approach is valuable for IT professionals, businesses, and individuals dealing with frequent file transfers or backups.

**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

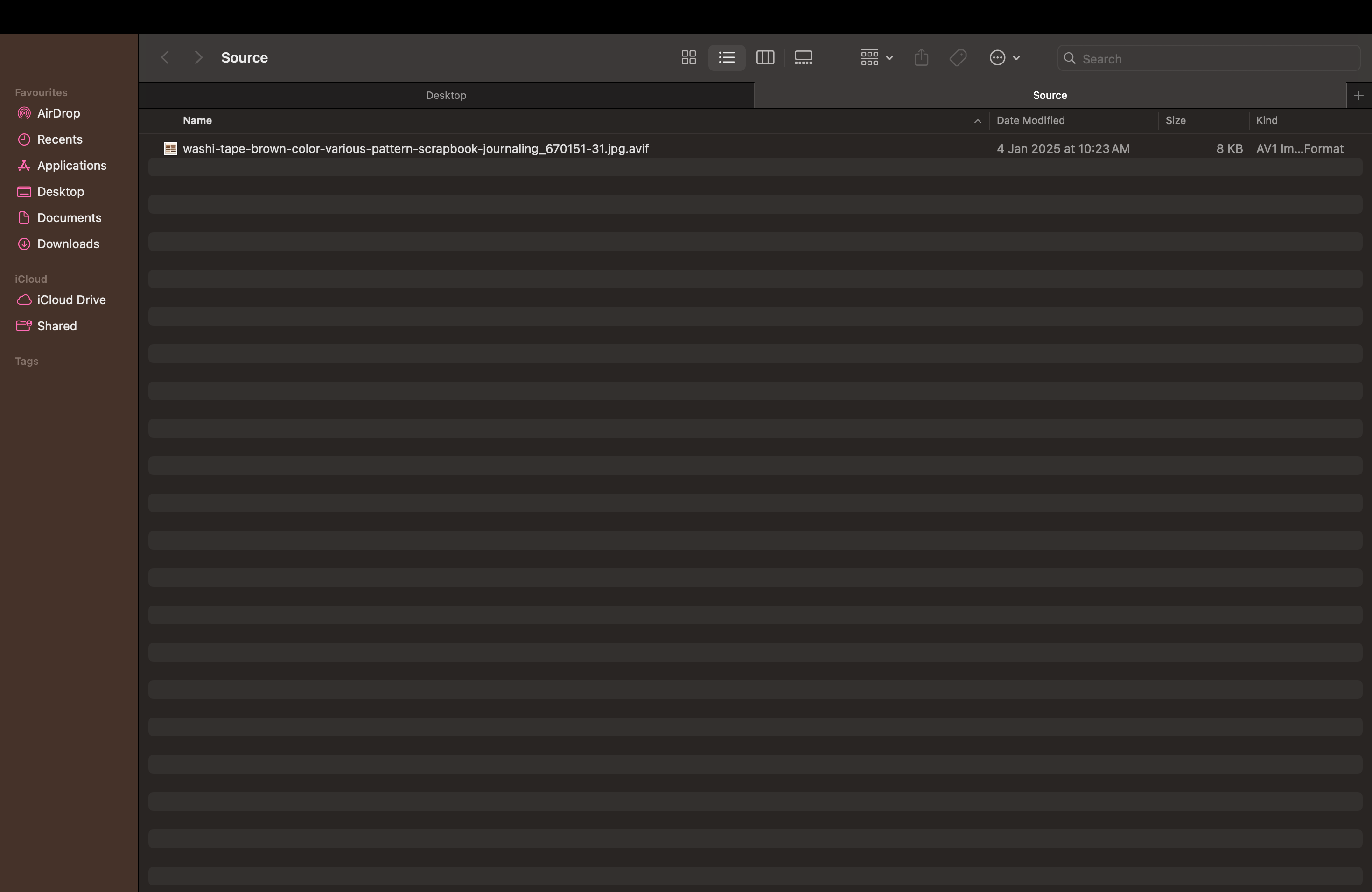
Create two folders named Source and Destination



Step 2

Store Some Files Inside the Source Folder:

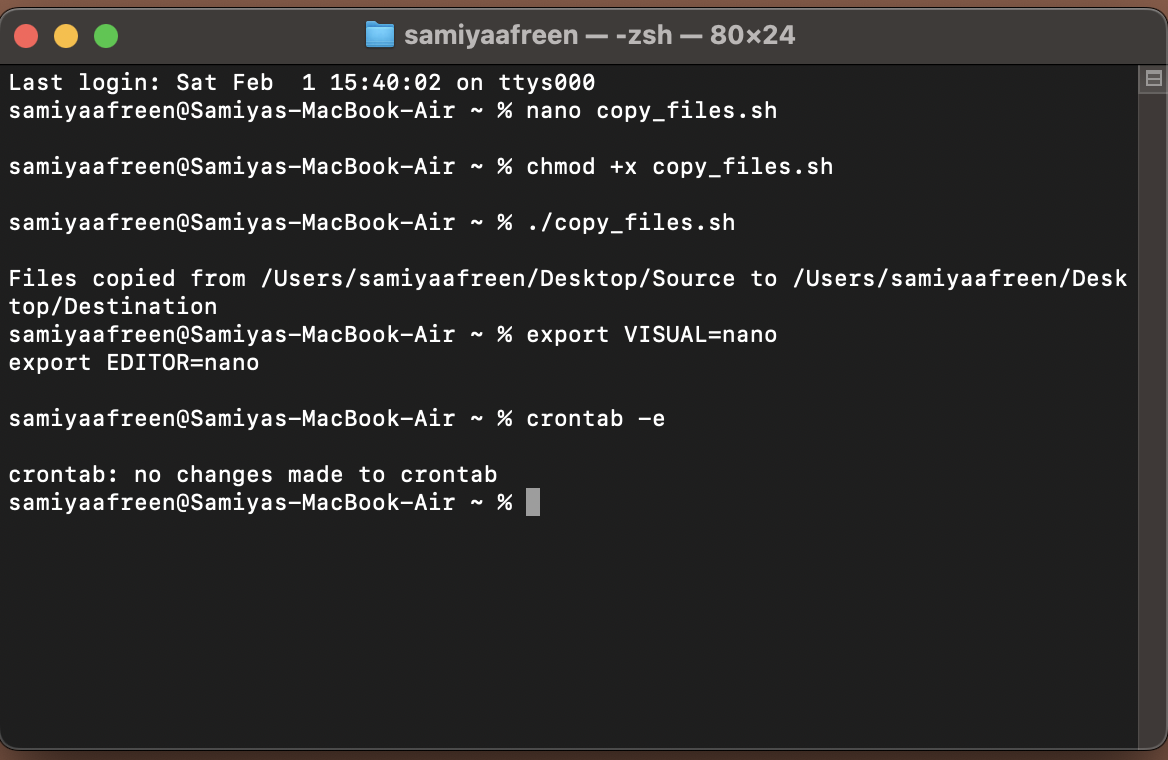
Add a few files into the source folder for testing purposes. These files will be copied to the destination folder automatically.



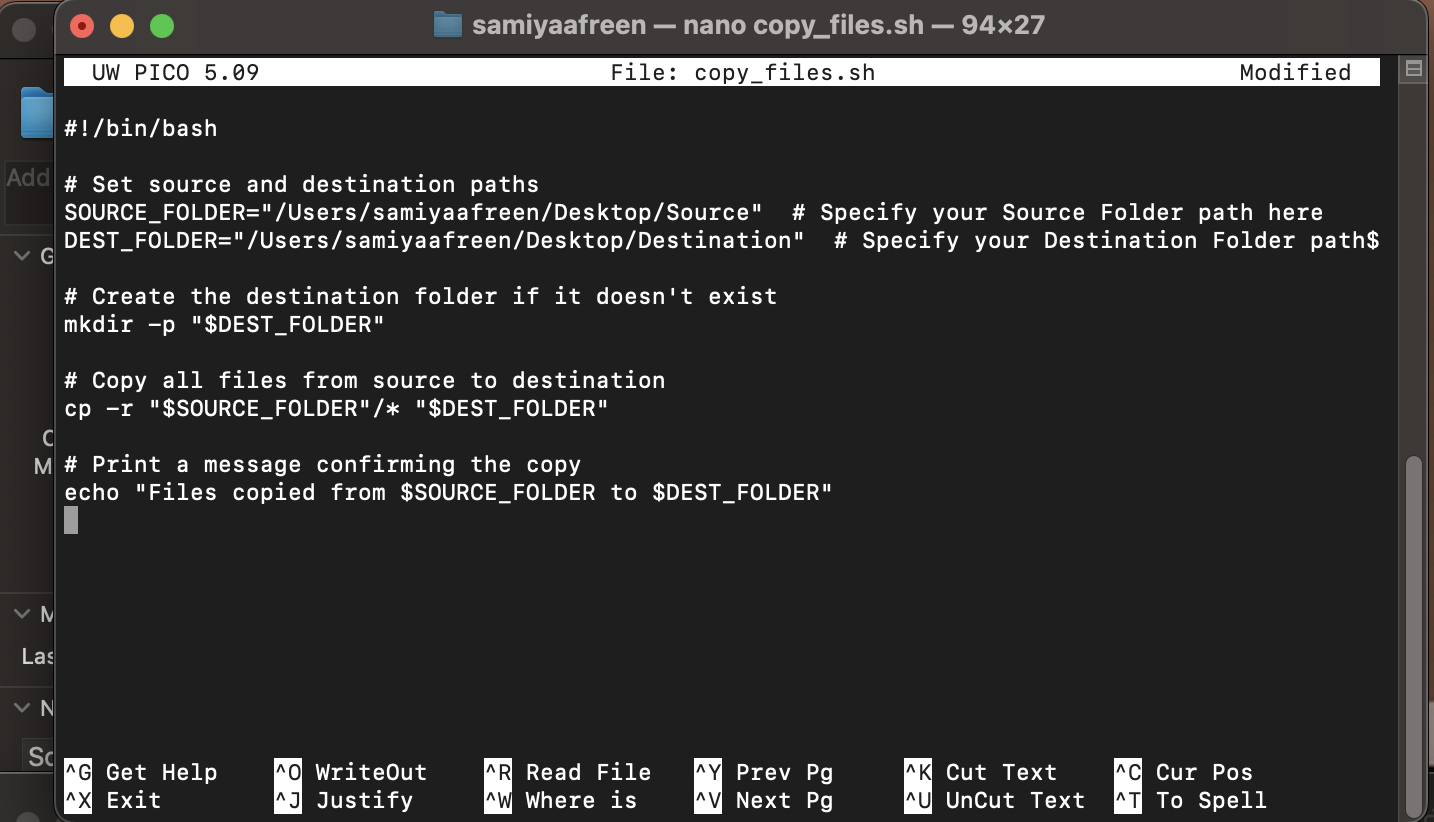
Step 3

**Write the Shell Script for Copying Files:**

Open the Terminal and create a new script using nano or any text editor of your choice:



Write the script to copy files from the source folder to the destination folder:

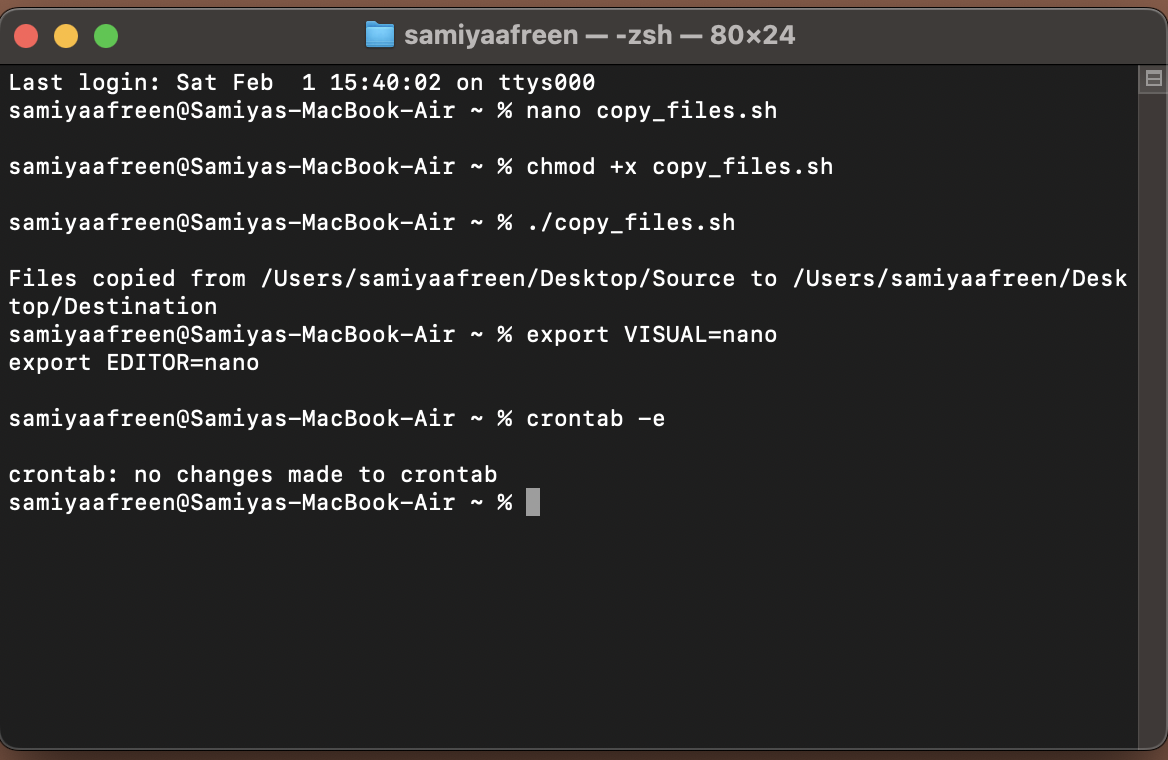


The cp -r command will copy all files, including subdirectories, from the source to the destination.

· Save the script:

· Press Ctrl + X, then Y, and hit Enter to save the file.

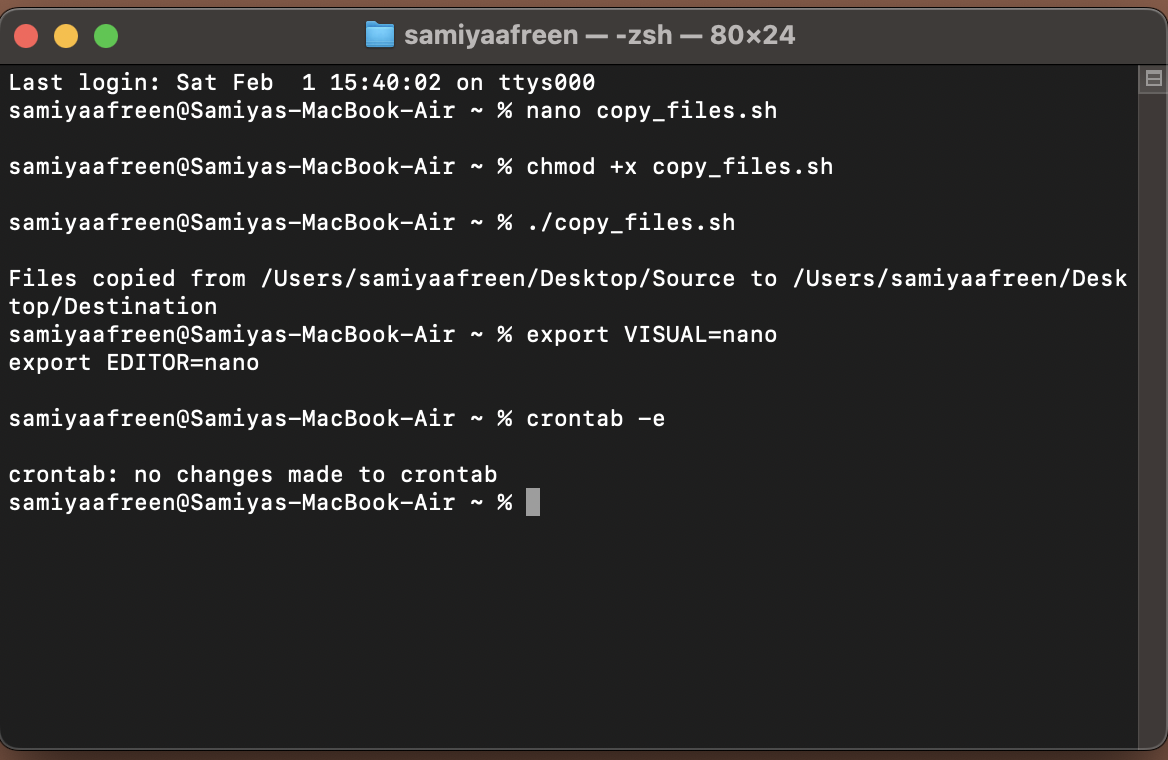
· **Make the script executable:**



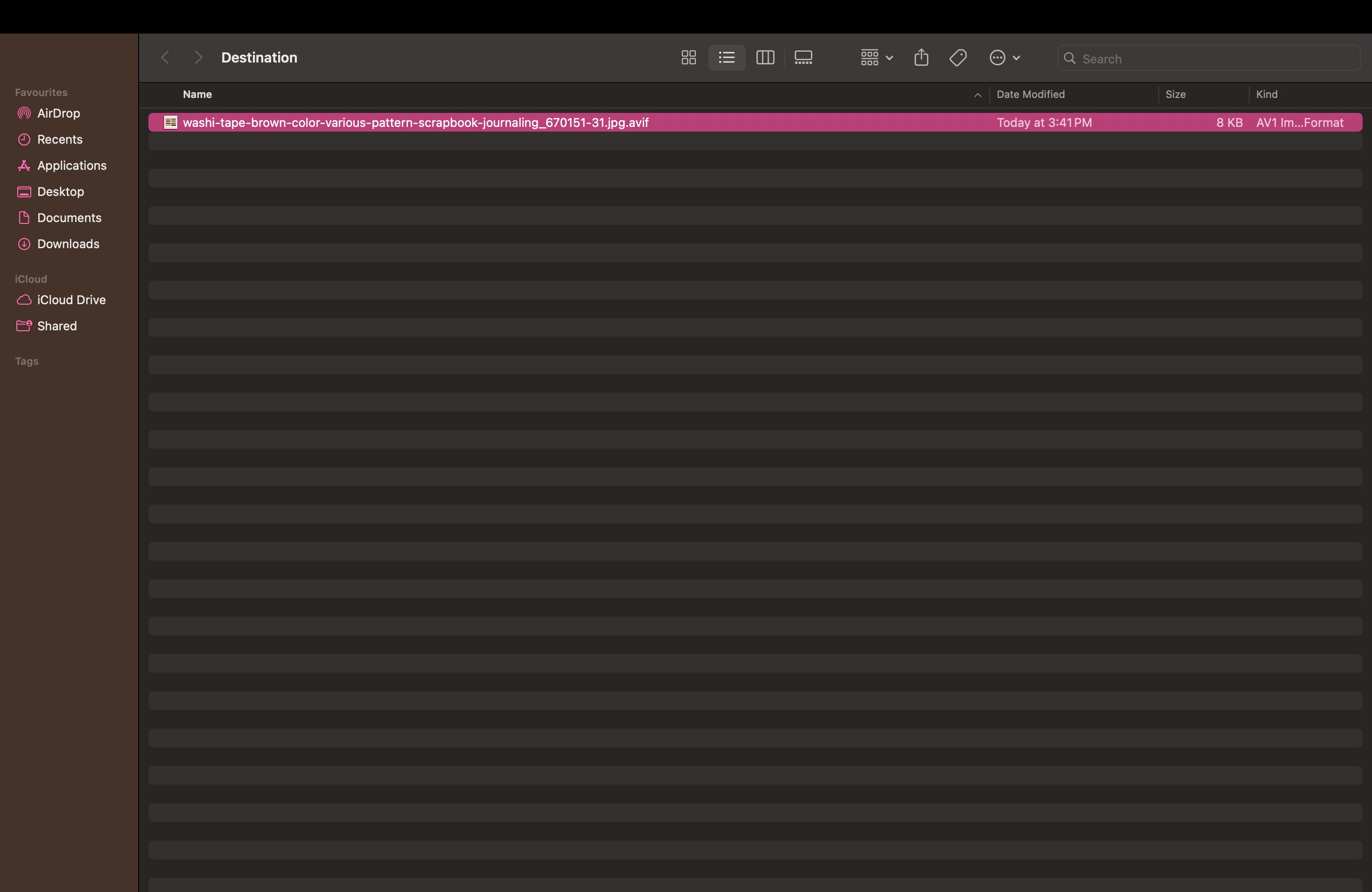
Step 4

**Test the Script:**

Run the script manually to test if it works.



Check the destination folder to confirm that the files have been copied.

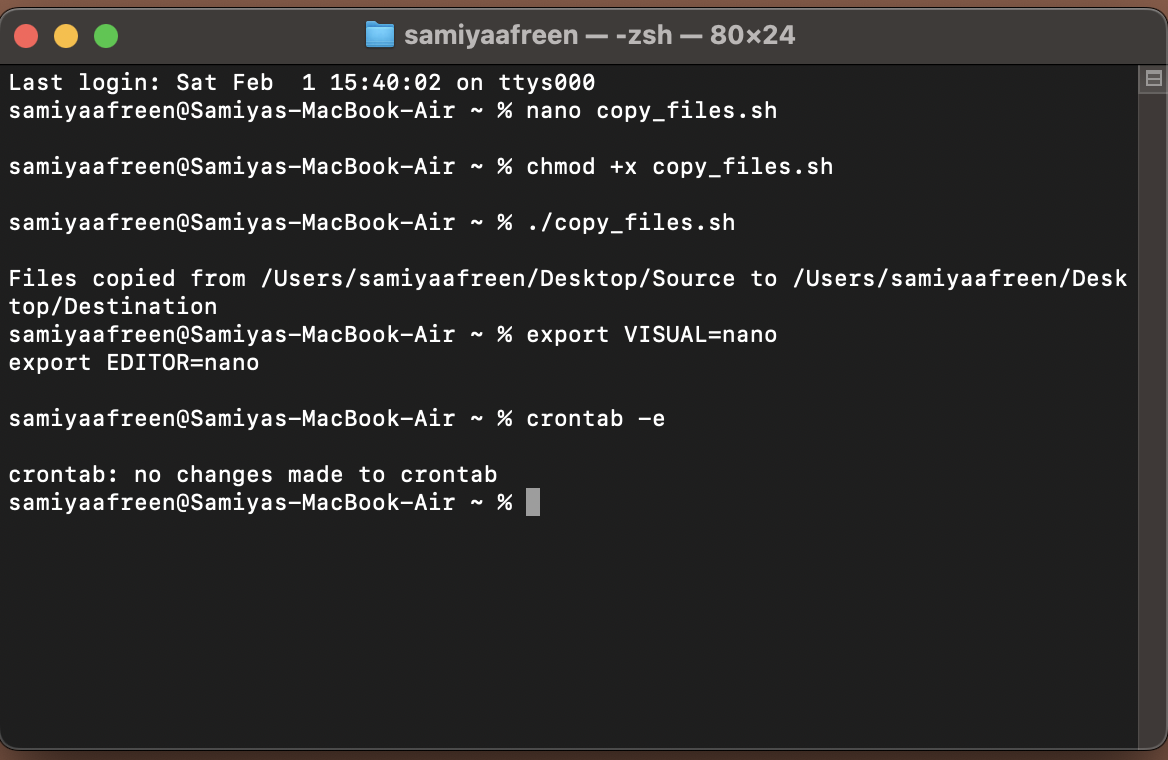


Step 5

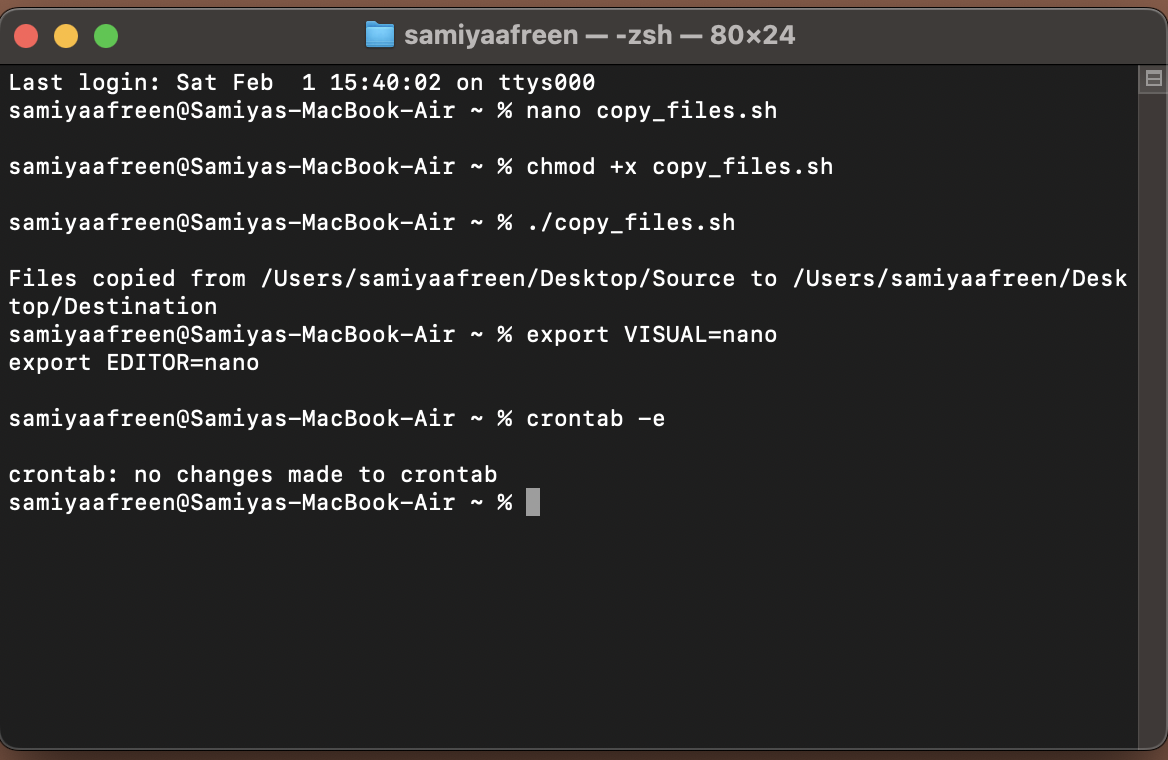
**Set Up Cron to Automate the Script:**

Set nano as the default editor (if it isn't already):

Run the following command.



After setting nano, run crontab -e:

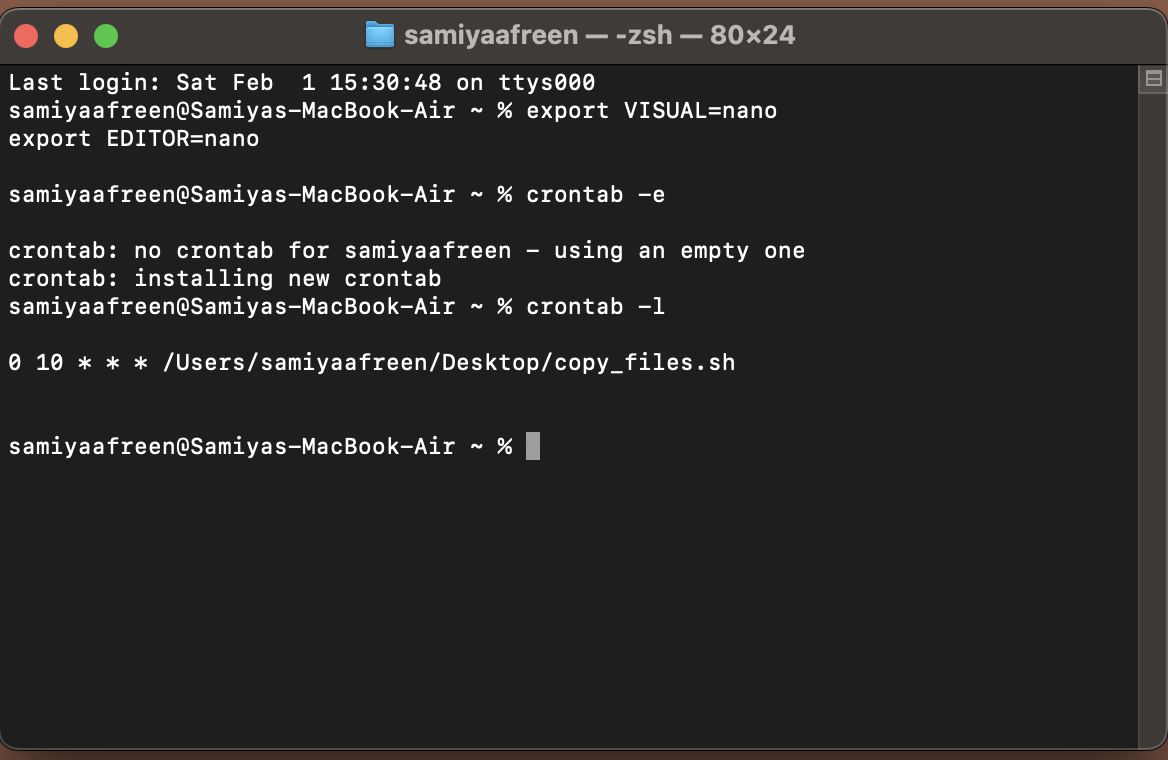


Add a cron job to run the script automatically.For example, if you want your script to run every day at 10:00 AM, add this line:

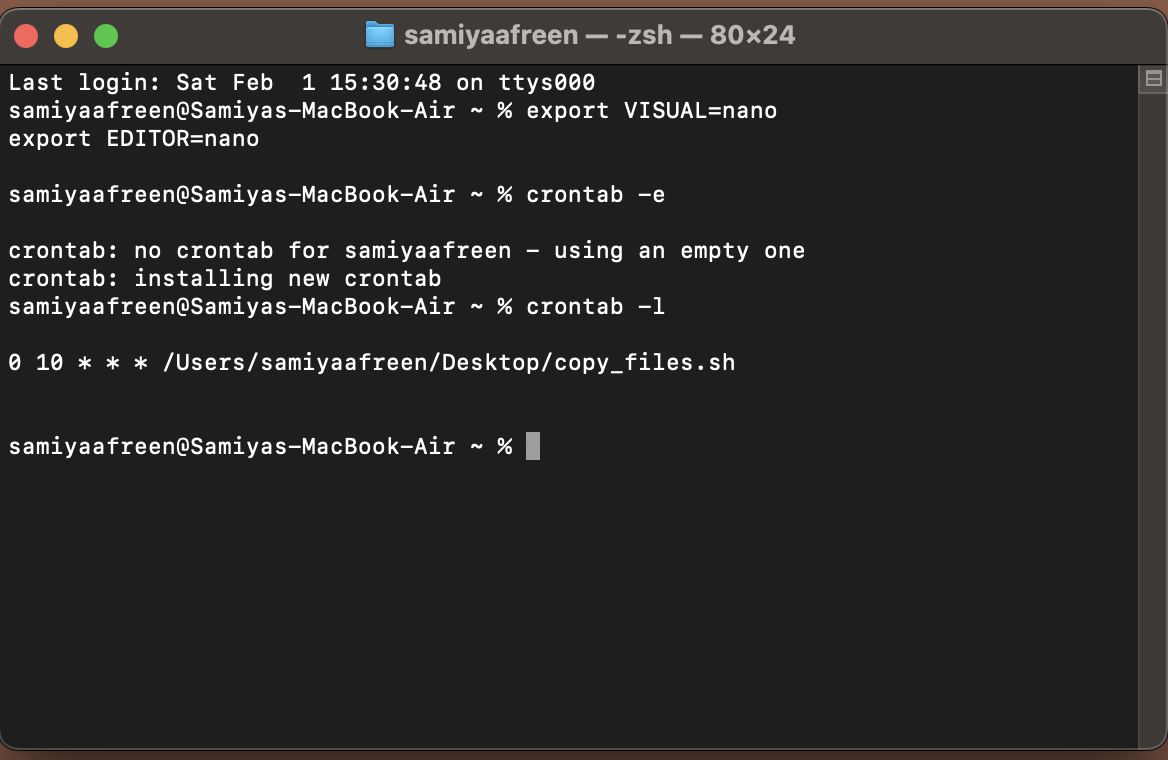


Save and Exit.

Verify the cron job is set correctly by running :



This command lists all active cron jobs. You should see your scheduled task there.

**Outcome**

By completing this PoC, you will have successfully:

1. Created a shell script to copy files from one folder to another.

2. Scheduled the task using cron to run automatically at a specified time.

3. Tested and verified the automated task.