Selected files

1 printable files

LAB 4 .md

LAB 4 .md

Objective: Automate file management.

backup.sh script

```bash
#!/bin/bash

# Create backup directory if it doesn't exist
mkdir -p backup

# Get current timestamp (YYYYMMDD_HHMMSS format)
timestamp=$(date +"%Y%m%d_%H%M%S")

# Find all .txt files in the current folder
for file in *.txt; do
    if [ -f "$file" ]; then
        # Extract filename without extension
        base=$(basename "$file" .txt)
        # Copy to backup/ with timestamp
        cp "$file" "backup/${base}_${timestamp}.txt"
        echo "Backed up $file → backup/${base}_${timestamp}.txt"
    fi
done
```

🔷 LAB4.md Documentation

LAB4 – Backup Script

🎯 Script Name

`backup.sh`

📌 Purpose

The script finds all `.txt` files in the current folder and copies them into a `backup/` directory with a timestamp added to the filename.
This ensures that no backups overwrite each other.

How the Script Works

1. `mkdir -p backup` → Creates a folder called `backup` if it doesn't already exist.
2. `timestamp=$(date +"%Y%m%d_%H%M%S")` → Stores the current date and time.
   Example: `20250910_231530`
3. The script loops over all `.txt` files in the current folder:
   - `basename "$file" .txt` → Gets the filename without extension.
   - `cp "$file" "backup/${base}_${timestamp}.txt"` → Copies the file into `backup/`, appending the timestamp.
4. Prints confirmation of each backup created.

▶ Example Run

Step 1: Create some `.txt` files

```
echo "Hello World" > notes.txt
echo "Shopping List" > list.txt
```

Step 2: Run the script

```
./backup.sh
```

Output Backed up notes.txt → backup/notes_20250910_231530.txt Backed up list.txt → backup/list_20250910_231530.txt

Step 3: Check backup folder

```
ls backup/
```

Result:

```
notes_20250910_231530.txt
list_20250910_231530.txt
```

Extra Questions

What is the difference between cp, mv, and rsync?

- 1. cp (copy) Copies files or directories from one place to another. Original file remains; a duplicate is created.
- 2. mv (move/rename)
- Moves files/directories (like cut-paste).
- Removes them from the original location.
- Also used for renaming.
- 3. rsync (remote sync / robust sync)
- A powerful tool for copying/synchronizing files.
- Works locally and over a network/SSH.
- Faster than cp because it only copies changes (not the entire file every time).
- Supports progress display, compression, mirroring directories, etc.

How can you schedule scripts to run automatically?

- 1. Using cron (Linux scheduler) cron runs tasks at fixed times/dates.
- 2. Using at (one-time scheduling) Runs a command/script once at a given time.
- 3. Using systemd timers (modern Linux) Alternative to cron. More powerful for recurring jobs. Uses .service and .timer files.