



## Project Work

**Student Name:** Abhishek Kumar

**UID:** 24MCC20006

**Branch:** MCA-CCD

**Section/Group:** 1-A

**Semester:** 1

**Date of Performance:** 30.10.2024

**Subject Name:** Linux Administration Lab

**Subject Code:** 24CAP-607

### Project: File Backup Script

#### 1.Aim:

To create a shell script that automates the process of backing up specific files and directories to a designated backup location.

#### 2.Overview:

The backup script will:

- Allow the user to specify which files or directories to back up.
- Copy those files to a specified backup location.
- Include options for timestamped backups to prevent overwriting previous backups.

#### 3.Hardware and Software Requirements:

- **Operating System:** Any Linux distribution
- **Shell:** Bash (most Linux distributions use Bash by default)
- **Storage:** Sufficient disk space for backups
- **Basic Command-Line Knowledge:** Familiarity with using the terminal

#### 4.Tools Overview:

- **Bash Scripting:** For writing the backup script.



- **Commands:**

- cp - to copy files and directories
- mkdir - to create backup directories if they don't exist
- date - to add timestamps to backup folders

### 5.Steps:

1. Open a Terminal:

**Start by opening your terminal.**

2. Create a New Script File:

Navigate to your home directory (or a directory of your choice) and create a new script file:

```
cd ~
```

```
nano backup_script.sh
```

3. Write the Script:

Add the following code to the backup\_script.sh file:

```
#!/bin/bash
```

```
# Check if the correct number of arguments is provided
```

```
if [ "$#" -ne 2 ]; then
```

```
    echo "Usage: $0 <source_directory> <backup_directory>"
```

```
    exit 1
```

```
fi
```

```
SOURCE_DIR="$1"
```

```
BACKUP_DIR="$2"
```

```
# Get the current date and time for timestamping
```

```
TIMESTAMP=$(date +"%Y%m%d_%H%M%S")
```



```
BACKUP_DEST="$BACKUP_DIR/backup_$TIMESTAMP"
```

```
# Create the backup directory if it doesn't exist
```

```
mkdir -p "$BACKUP_DEST"
```

```
# Copy files from the source directory to the backup directory
```

```
cp -r "$SOURCE_DIR/"* "$BACKUP_DEST/"
```

```
echo "Backup of '$SOURCE_DIR' completed successfully to '$BACKUP_DEST'"
```

4. Save and Exit:

**Press Ctrl + O to save the file, then Ctrl + X to exit the editor.**

5. Make the Script Executable:

Run the following command to make your script executable:

```
chmod +x backup_script.sh
```

- **How to Run the Script**

1. Execute the Script:

Run the script with the source directory and backup directory as arguments. For example:

```
./backup_script.sh /path/to/source /path/to/backup
```

2. Example Usage:

If you want to back up a folder called my\_files located in your home directory to a backup folder called my\_backup, use:

```
./backup_script.sh ~/my_files ~/my_backup
```

## 6.Output:

```
abhishek@192:~/Backup
abhishek@192:~$ ls
backup_script.sh  Documents  Music      Public     Videos
Desktop          Downloads  Pictures   Templates
abhishek@192:~$ rm backup_script.sh
abhishek@192:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
abhishek@192:~$ nano backup_script.sh
abhishek@192:~$ chmod +x backup_script.sh
abhishek@192:~$ ./backup_script.sh ~/Music ~/Backup
cp: cannot stat '/home/abhishek/Music/*': No such file or directory
Backup of '/home/abhishek/Music' completed successfully to '/home/abhishek/Backup/backup_20241030_022629'
abhishek@192:~$ ls ~/mu
ls: cannot access '/home/abhishek/mu': No such file or directory
abhishek@192:~$ ls ~/music
ls: cannot access '/home/abhishek/music': No such file or directory
abhishek@192:~$ ls
Backup      Desktop    Downloads  Pictures   Templates
backup_script.sh  Documents  Music      Public     Videos
abhishek@192:~$ ls ~/Music/
abhishek@192:~$ ./backup_script.sh ~/Music ~/Backup
cp: cannot stat '/home/abhishek/Music/*': No such file or directory
Backup of '/home/abhishek/Music' completed successfully to '/home/abhishek/Backup/backup_20241030_022805'
```

```
abhishek@192:~/Backup
abhishek@192:~$ chmod +x backup_script.sh
abhishek@192:~$ ./backup_script.sh ~/Music ~/Backup
cp: cannot stat '/home/abhishek/Music/*': No such file or directory
Backup of '/home/abhishek/Music' completed successfully to '/home/abhishek/Backup/backup_20241030_022629'
abhishek@192:~$ ls ~/mu
ls: cannot access '/home/abhishek/mu': No such file or directory
abhishek@192:~$ ls ~/music
ls: cannot access '/home/abhishek/music': No such file or directory
abhishek@192:~$ ls
Backup      Desktop    Downloads  Pictures   Templates
backup_script.sh  Documents  Music      Public     Videos
abhishek@192:~$ ls ~/Music/
abhishek@192:~$ ./backup_script.sh ~/Music ~/Backup
cp: cannot stat '/home/abhishek/Music/*': No such file or directory
Backup of '/home/abhishek/Music' completed successfully to '/home/abhishek/Backup/backup_20241030_022805'
abhishek@192:~$ ls
Backup      Desktop    Downloads  Pictures   Templates
backup_script.sh  Documents  Music      Public     Videos
abhishek@192:~$ cd Backup/
abhishek@192:~/Backup$ ls
backup_20241030_022629  backup_20241030_022805
abhishek@192:~/Backup$
```

## 6. Conclusion

This project helps you understand file management and shell scripting basics. You can further enhance the script by adding features such as:



UNIVERSITY INSTITUTE *of*  
**COMPUTING**  
*Asia's Fastest Growing University*

NAAC  
GRADE **A+**  
ACCREDITED UNIVERSITY

- Logging backup activity to a file.
- Including error handling for cases where the source directory does not exist.
- Providing options to compress the backup files.

#### **7.Learning outcomes (What I have learnt):**

- Basics of shell scripting in Bash.
- File and directory manipulation commands.
- Understanding of how to automate tasks in Linux.
- Basic error handling and input validation in scripts.

Feel free to modify the script according to your needs or add features as you learn! Let me know if you have any questions or need further assistance.

---

**Teacher's Signature**