

Placement Empowerment Program

Cloud Computing and DevOps Centre

Create a Simple Backup Script

Create a script that backups up your entire Git repository to a local folder daily

Name: Raichal Maria P

Department: IT

Introduction

Automating Git repository backups helps prevent data loss by regularly saving copies of your repositories. This guide will walk you through creating a simple script using tar (Linux/macOS) or robocopy (Windows) to back up your Git repositories to a local folder daily, with automation via cron or Task Scheduler.

Overview

This project involves creating an automated backup script for Git repositories using shell scripting (tar) on Linux/macOS or batch scripting (robocopy) on Windows. The script will copy repositories to a designated backup folder daily, ensuring data safety. Automation will be handled using cron jobs (Linux/macOS) or Task Scheduler (Windows), eliminating the need for manual backups.

Objectives

Objectives:

- Automate daily backups of Git repositories to a local folder.
- Use shell scripting (tar) for Linux/macOS and batch scripting (robocopy) for Windows.
- Ensure data safety by maintaining up-to-date repository copies.
- Implement task automation using cron (Linux/macOS) or Task Scheduler (Windows).
- Minimize manual effort while securing repository data efficiently.

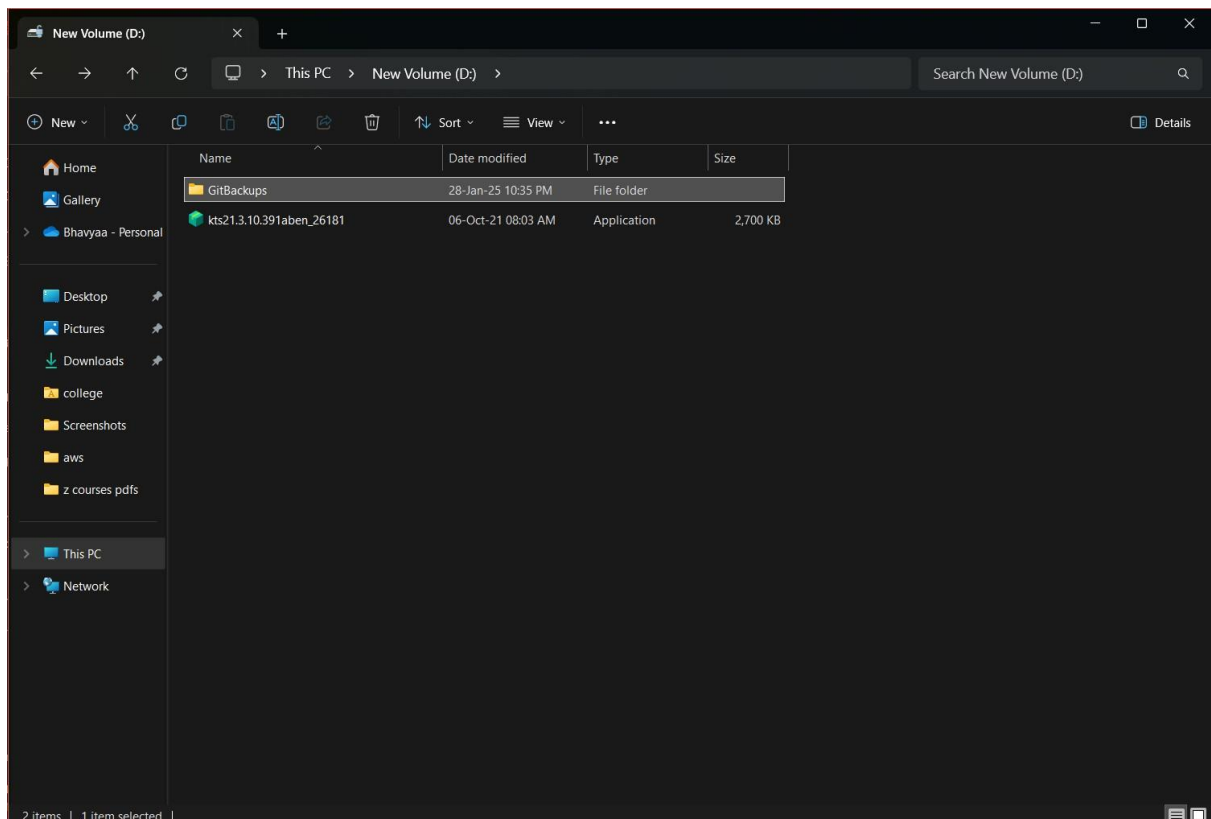
Step-by-Step Overview Step

1

Choose Your Backup Location

Decide where you want to store your backup files.

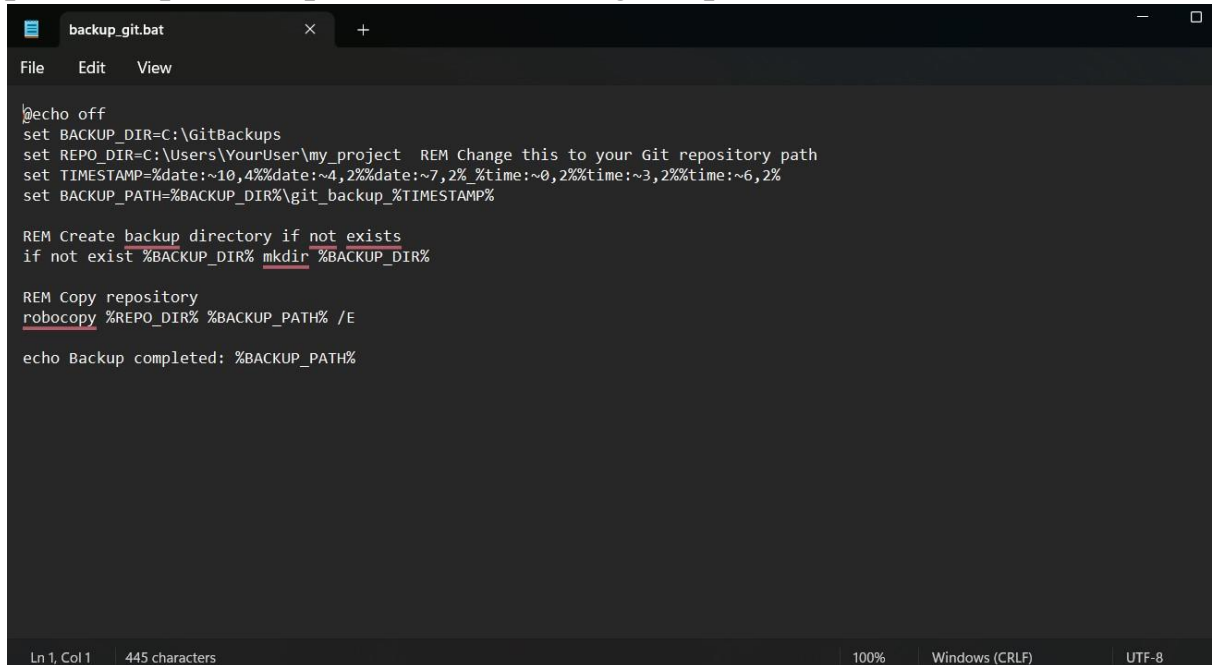
- Example for **Windows**: C:\GitBackups\



Step 2

Create the Backup Script

1. Open Notepad and paste the following script:



```
backup_git.bat
File Edit View

@echo off
set BACKUP_DIR=C:\GitBackups
set REPO_DIR=C:\Users\YourUser\my_project REM Change this to your Git repository path
set TIMESTAMP=%date:~10,4%%date:~4,2%%date:~7,2% %time:~0,2%%time:~3,2%%time:~6,2%
set BACKUP_PATH=%BACKUP_DIR%\git_backup_%TIMESTAMP%

REM Create backup directory if not exists
if not exist %BACKUP_DIR% mkdir %BACKUP_DIR%

REM Copy repository
robocopy %REPO_DIR% %BACKUP_PATH% /E

echo Backup completed: %BACKUP_PATH%

Ln 1, Col 1 | 445 characters | 100% | Windows (CRLF) | UTF-8
```

2. Save the file as backup_git.bat.
3. Double-click the script to test if it runs successfully.

Step 3

Automate the Backup

Open **Task Scheduler** (Win + R, then type taskschd.msc).

Click **Create Basic Task**.

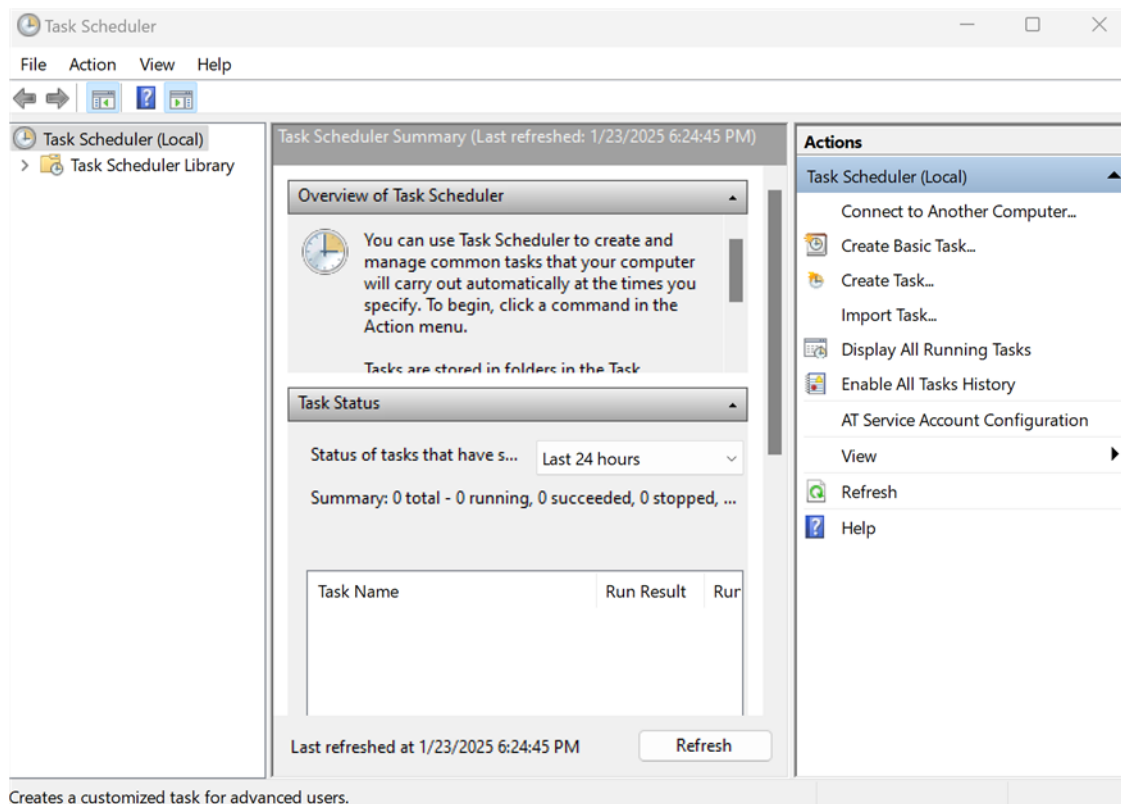
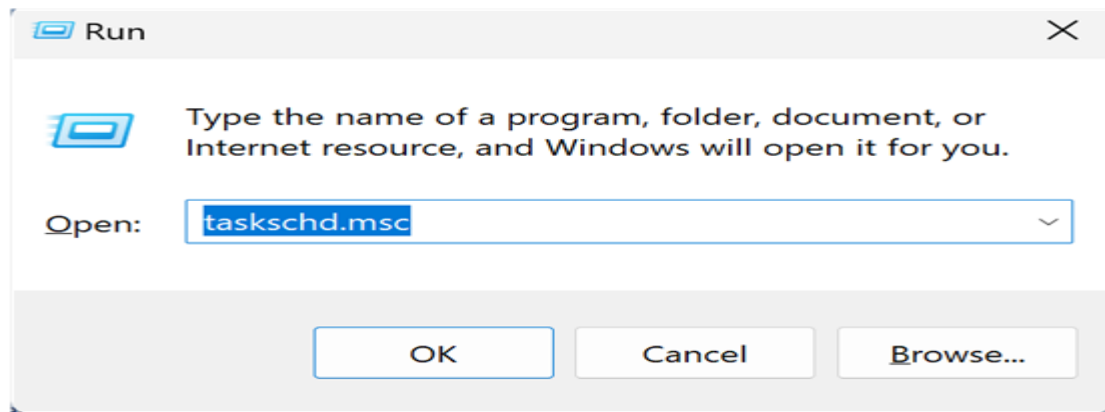
Name it **Git Backup** and click **Next**.

Choose **Daily**, then click **Next**.

Set the time you want the script to run (e.g., **2:00 AM**), then click **Next**.

Choose **Start a Program**, then **Browse** for backup_git.bat.

Click **Finish**.



Step 4

Step 4: Verify Your Backups

- Check the **backup directory** (~/.git_backups on Linux/macOS or C:\GitBackups on Windows) to ensure backups are created.
- Open a backup file to confirm it contains your Git repository.