 

**Placement Empowerment Program**

***Cloud Computing and DevOps Centre***

**Create a simple Backup Script**

Create a script that backs up your entire Git repository

to a local folder daily

Name: Tudor Savio J Department: ECE



**Introduction :**

Version control with Git ensures that your projects, including websites or software applications, are securely managed and stored. However, sometimes you might want to create **backups** of your Git repositories to keep a copy in case of data loss, or to create a snapshot of your project at a specific point in time.

**Objectives :**

**Create a backup script** that will back up your Git repository to a local folder.

1. **Robocopy** (Windows) for copying and compressing files.
2. **Automate the backup process** to run on a daily schedule using **Task Scheduler (Windows)**.
3. Understand the basic commands involved in **scripting**, **backups**, and **task automation**.

**Step 1: Creating a Backup Script for Your Git Repository**

Before automating the backup, you first need a script that will create the backup. This script will copy the contents of your Git repository to a backup folder. You can either compress the backup into an archive or copy the files directly.

**1.2 For Windows: Using robocopy**

Windows systems typically use robocopy for file copying and backup purposes. Below is a Windows script using robocopy to back up your Git repository.

1. **Create the backup directory** for storing backups (e.g., C:\GitBackups).
2. **Create the backup script** (e.g., backup\_git\_repo.bat) with the following content:

REM Set the repository and backup directory

set REPO\_DIR=C:\path\to\your\git\repository

set BACKUP\_DIR=C:\GitBackups

set DATE=%DATE:/=-%

set BACKUP\_NAME=git\_repo\_backup\_%DATE%.zip

REM Create a backup using robocopy

robocopy %REPO\_DIR% %BACKUP\_DIR%\%BACKUP\_NAME% /E /COPYALL /Z /R:3 /W:5

REM Optional: Remove backups older than 30 days

forfiles /p %BACKUP\_DIR% /s /m \*.zip /D -30 /C "cmd /c del @path"

1. **Explanation**:
   * REPO\_DIR: The path to your local Git repository.
   * BACKUP\_DIR: The directory where the backups will be stored.
   * DATE: The current date, formatted for the filename.
   * robocopy: This command copies files from the repository to the backup folder.
   * The forfiles command deletes backups older than 30 days.
2. **Save the script** with a .bat extension, such as backup\_git\_repo.bat.

**Step 2: Automating the Backup Process**

**2.2 Automate Backups with Task Scheduler on Windows**

Task Scheduler is the built-in tool for automating tasks on Windows. Here’s how to set it up to run the backup script daily.

1. Open **Task Scheduler** by searching for it in the Start menu.
2. In the **Actions** panel, click **Create Task**.
3. In the **General** tab:
   * Name the task (e.g., Git Repo Backup).
   * Choose **Run with highest privileges** if needed.
4. In the **Triggers** tab:
   * Click **New** and set the trigger to **Daily** at a time (e.g., 2:00 AM).
5. In the **Actions** tab:
   * Choose **Start a program**.
   * Browse to the location of your .bat backup script (e.g., backup\_git\_repo.bat).
6. Click **OK** to save the task. Your backup script will now run daily at the specified time.

**Step 3: Verifying the Backup**

Once you've set up the backup script and scheduling:

1. **Manually test the script** by running it once.

For Windows, double-click the .bat file to run it manually.

1. Check the backup folder to confirm the backup has been created (e.g., a .zip file for Windows).
2. Check that backups are being created on the scheduled time by reviewing the backup folder the next day.

**Conclusion**

1. Created a **backup script** that copies your Git repository files to a local folder using either robocopy for Windows.
2. **Task Scheduler** (Windows) to automate the backup process to run daily.
3. Learned how to **verify** your backups and ensure that your data is safely stored and protected.