**# Day 5 Task: Advanced Linux Shell Scripting for DevOps Engineers with User management**

**1) You have to do the same using Shell Script i.e using either Loops or command with start day and end day variables using arguments** -

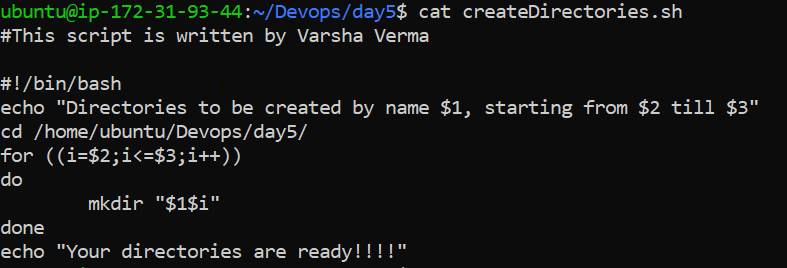
**Solution:**

**Commands:**

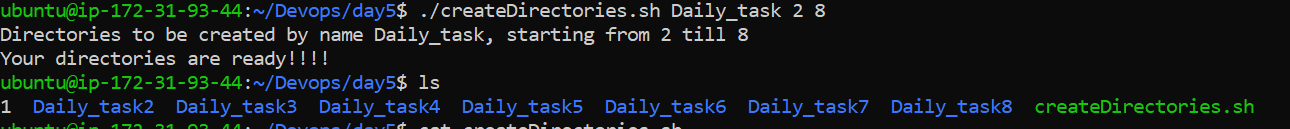
vi createDirectory.sh

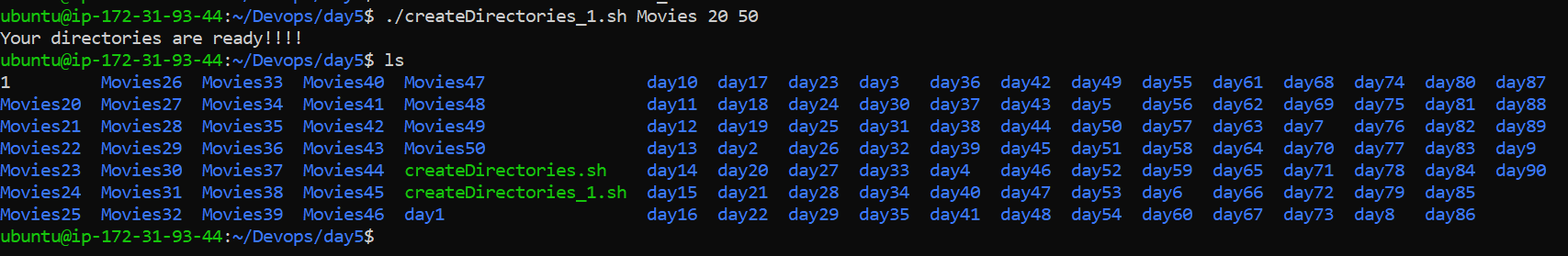
chmod u+x createDirectory.sh

**Script:**



**Output:**





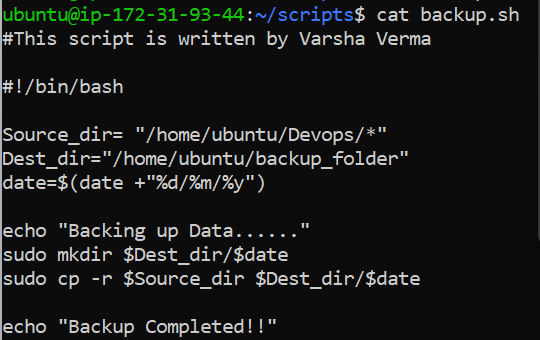
**2)** **Create a Script to backup all your work done till now**.

Backups are an important part of DevOps Engineers day to Day activities

The video in References will help you to understand How a DevOps Engineer takes backups (it can feel a bit difficult but keep trying, Nothing is impossible.)

Watch [this video](https://youtu.be/aolKiws4Joc)

**Solution:**

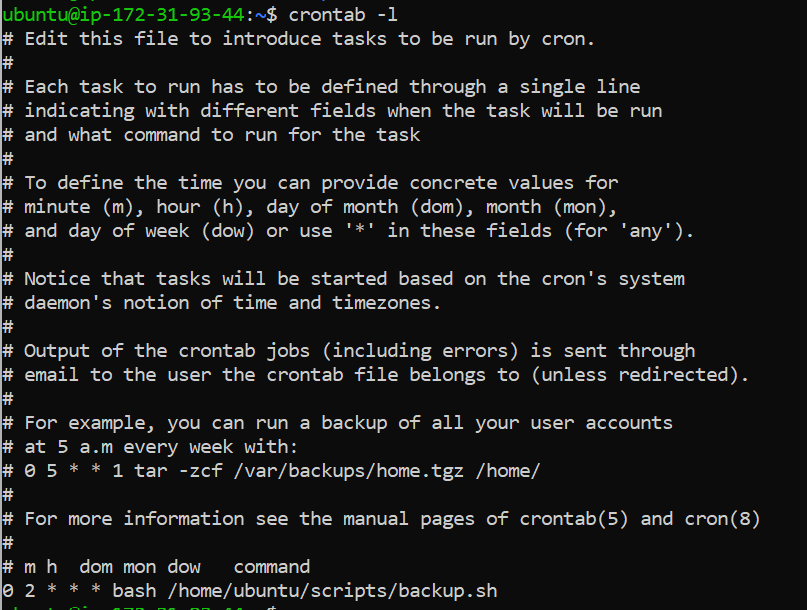


**3) Read About Cron and Crontab, to automate the backup Script**

Cron is the system's main scheduler for running jobs or tasks unattended. A command called crontab allows the user to submit, edit or delete entries to cron. A crontab file is a user file that holds the scheduling information.

Instead of doing **recurring tasks on your system** at particular duration is something that Linux Admins/ DevOps Enginers must have to perform but we can also achieve this by using cron to automate this by scheduling via crontab.

**Cron is a system daemon run on any Linux system that is responsible for detecting cron jobs and executing them at given intervals.** Cron runs every minute and it will inspect a set of pre-defined directories on your filesystem to see if jobs need to be run.



**4) Read about User Management**

A user is an entity, in a Linux operating system, that can manipulate files and perform several other operations. Each user is assigned an ID that is unique for each user in the operating system. In this post, we will learn about users and commands which are used to get information about the users. After installation of the operating system, the ID 0 is assigned to the root user and the IDs 1 to 999 (both inclusive) are assigned to the system users and hence the ids for local user begins from 1000 onwards.

**5) Create 2 users and just display their Usernames**

**Solution:**

**Commands:**

sudo useradd user1

sudo useradd user2

**Script:**

cat /etc/passwd -> this is used to view the file but here we have to display only username hence used **awk**  for it

**Output:**

