

# DEVOPS

## TASK 1

### Installation of Nginx.

#### Task Description:

This task involves creating a Jenkins Freestyle job that automates the installation of Nginx on an Ubuntu server using the command `sudo apt-get install -y nginx`. This automation ensures consistent server setup and can be easily repeated, making it ideal for automated environments.

---

#### Steps:

1. **Update Package List:**

- Run the following command to update the package list:
- `sudo apt update`

2. **Install Nginx:**

- Install Nginx using the following command:
- `sudo apt-get install -y nginx`

3. **Verify Installation:**

- Check if Nginx is running with the command:
- `systemctl status nginx`

4. **Create Jenkins Freestyle Job:**

- In Jenkins, click New Item and create a Freestyle project (e.g., Install\_Nginx).

5. **Configure Job:**

- Add a build step of type Execute shell.
- In the shell command box, enter:
- `sudo apt update`
- `sudo apt-get install -y nginx`

6. **Save and Run Job:**

- Save the job and click Build Now to run it.

## 7. Check Console Output:

- After the job completes, view the **Console Output** to confirm the successful installation of Nginx.

### OUTPUT:

The screenshot displays the Jenkins web interface. On the left, a sidebar contains navigation links: 'Changes', 'Console Output' (selected), 'Edit Build Information', 'Delete build #3', 'Timings', and 'Previous Build'. The main area shows the console output for a build. The output text is as follows:

```
Started by user Swathika H
Running as SYSTEM
Building in workspace C:\ProgramData\Jenkins\jenkins\workspace\Nginx
The recommended git tool is: NONE
using credential Dockerhub_Credentials
Cloning the remote Git repository
Cloning repository https://github.com/Swathika-1804/Docker.git
> git.exe init C:\ProgramData\Jenkins\jenkins\workspace\Nginx # timeout=10
> git.exe config remote.origin.url https://github.com/Swathika-1804/Docker.git # timeout=10
> git.exe fetch --tags --force --progress -- https://github.com/Swathika-1804/Docker.git +refs/heads/*:refs/remotes/origin/* # timeout=10
> git.exe config remote.origin.url https://github.com/Swathika-1804/Docker.git # timeout=10
> git.exe config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
> git.exe rev-parse "refs/remotes/origin/master^{commit}" # timeout=10
> git.exe rev-parse "origin/master^{commit}" # timeout=10
debconf: unable to initialize frontend: Readline
debconf: (This frontend requires a controlling tty.)
debconf: falling back to frontend: Teletype
Setting up libbpf1:amd64 (1:1.1.0-1) ...
Setting up libtirpc3:amd64 (1.3.3+ds-1) ...
Setting up iproute2 (6.1.0-3) ...
debconf: unable to initialize frontend: Dialog
debconf: (Dialog frontend will not work on a dumb terminal, an emacs shell buffer, or without a controlling terminal.)
debconf: falling back to frontend: Readline
debconf: unable to initialize frontend: Readline
debconf: (This frontend requires a controlling tty.)
debconf: falling back to frontend: Teletype
Setting up nginx (1.22.1-9+deb12u1) ...
invoke-rc.d: could not determine current runlevel
invoke-rc.d: policy-rc.d denied execution of start.
Processing triggers for libc-bin (2.36-9+deb12u9) ...
Finished: SUCCESS
```

Below the console output, the Jenkins REST API endpoint 'localhost:8080/job/task1/1/console' is visible in the browser's address bar. The bottom of the image shows a Windows taskbar with various application icons and a system tray displaying '31°C Sunny' and the time '9:31 PM 3/20/2025'.