

DEVOPS

ASSIGNMENT 1

Installation of Nginx.

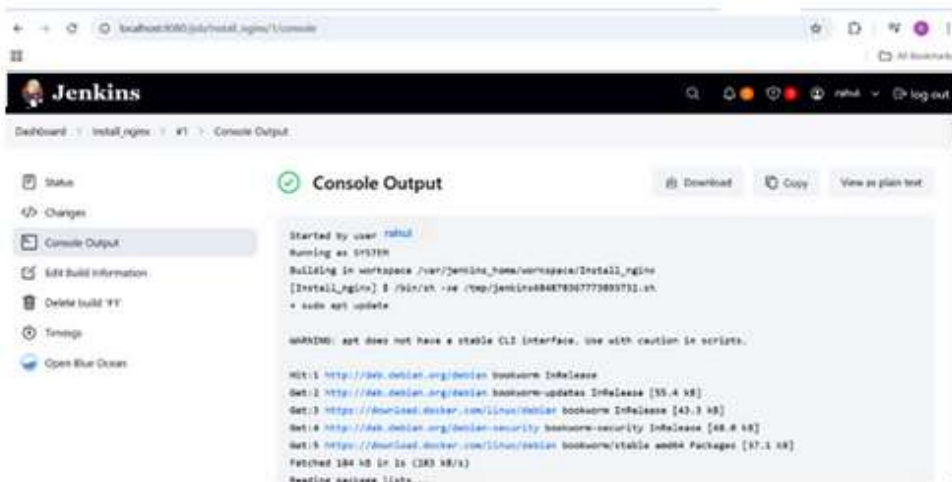
Task Description:

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:

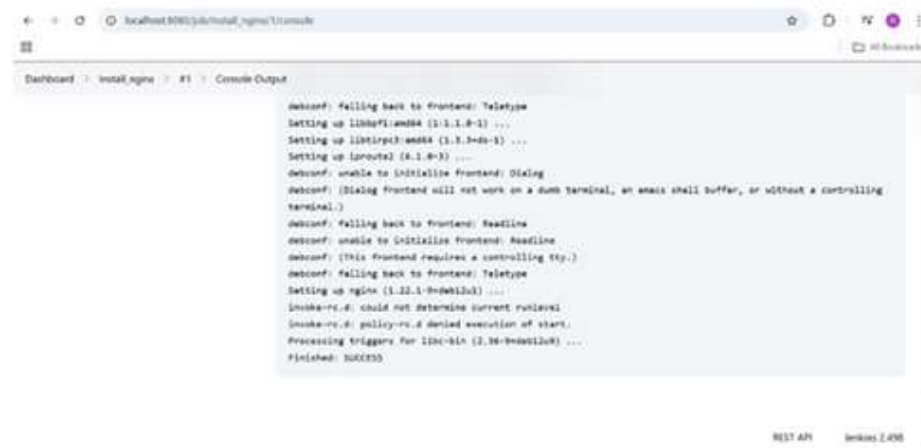


The screenshot shows the Jenkins web interface with the 'Console Output' tab selected for a build named 'install_nginx'. The output text is as follows:

```
Started by user rnhul
Running as SYSTEM
Building in workspace /var/jenkins_home/workspace/install_nginx
[install_nginx] $ /bin/bash -xe /tmp/jenkins68487834777889733.sh
+ sudo apt update

WARNING: apt does not have a stable CLI interface. Use with caution in scripts.

Hit:1 http://deb.debian.org/debian bookworm InRelease
Get:2 http://deb.debian.org/debian bookworm-updates InRelease [55.4 kB]
Get:3 http://deb.debian.org/debian-security bookworm-security InRelease [43.9 kB]
Get:4 http://deb.debian.org/debian-security bookworm-security InRelease [48.8 kB]
Get:5 http://deb.debian.org/debian bookworm/stable amd64 Packages [17.1 kB]
Fetched 184 kB in 1s (128 kB/s)
Reading package lists...
```



The screenshot continues the Jenkins console output from the previous one. The output text is as follows:

```
debconf: falling back to frontend: Teletype
Setting up libbpf1:amd64 (1:1.1.0-1) ...
Setting up libtirpc3:amd64 (1.3.3-4+deb11~) ...
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 (1.22.1-9deb12u3) ...
invoke-rc.d: could not determine current runlevel
invoke-rc.d: policy-rc.d denied execution of start.
Processing triggers for libc-bin (2.36-9deb12u8) ...
Finished: SUCCESS
```

At the bottom right of the console output area, the text 'REST API Jenkins 2.400' is visible.