DOCUMENTATION

TASK 1:
Write shell script to install jenkins
- Independent of os dependency
- Must provide version to install

In this Script we need to first check weather the local machine has its JAVA installed in it or not if not then we must run a script to install jdk then proceed towards our jenkins installation

Here we install jenkins from the official site of its latest version

Thus , a script was created and it does all its required work to be useful by installing/updating the jenkins in the local machine

For a more detailed view you can also checkout my script click here
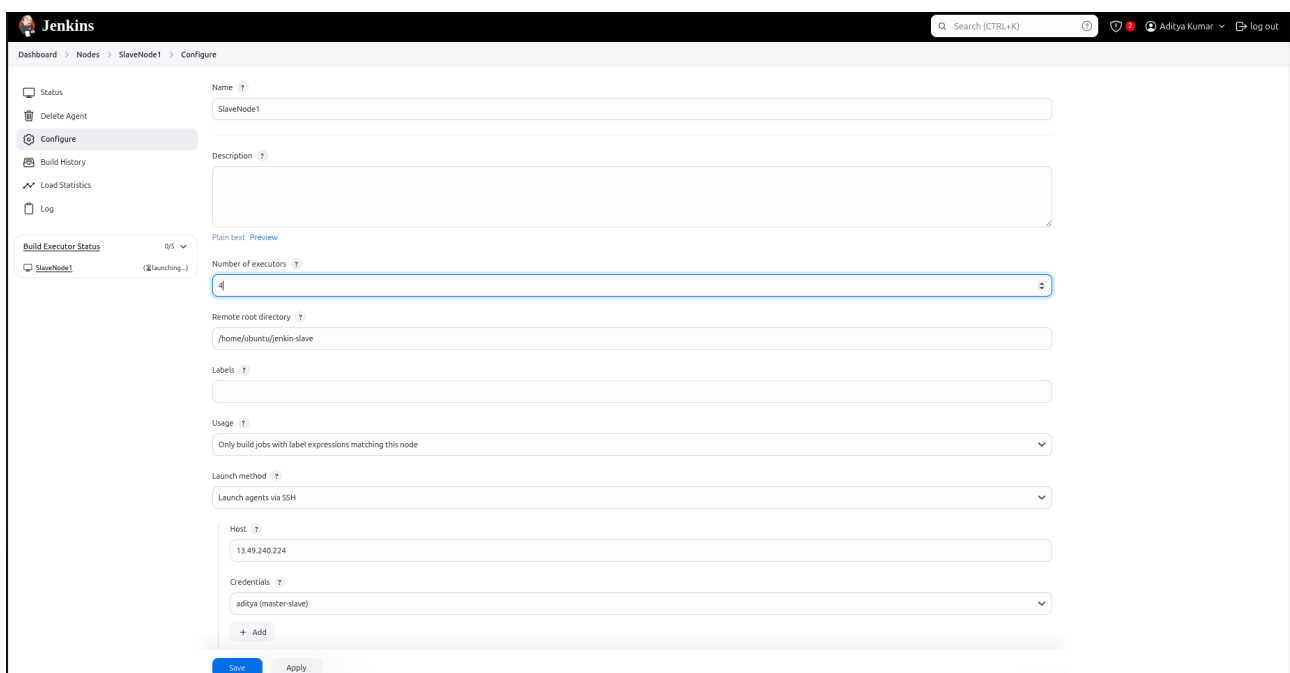

TASK 2:
Setup Master-slave architecture
        It should be capable of executing 5 tasks at a time

This task is to be done on jenkins by setting up credentials of the local machine and implement SSH on it and as well as the slave

The slave must have jdk installed in it

For details you can checkout the photo attached below:

Here I have set up the Global Credential with my private key and connected it via ssh thus making a master-slave architect

TASK 3:
Setup Role based authorization

In this task after the slave node was created we assign it some tags. And then we go into MANAGE JENKINS and modify the roles we need to give it to it.

Manage Roles

Global roles

	Overall		Credentials					Agent							Job									Run			View				SCM	Metrics		
Role	Administer	Read	Create	Delete	ManageDomains	Update	View	Build	Configure	Connect	Create	Delete	Disconnect	Provision	Build	Cancel	Configure	Create	Delete	Discover	Move	Read	Workspace	Delete	Replay	Update	Configure	Create	Delete	Read	Tag	HealthCheck	ThreadDump	View
admin	✓																																	
employee		✓																																

Role to add

[ ]

Add

Item roles

Role	Pattern	Template	Credentials					Job									Run			View				SCM	Metrics		
			Create	Delete	ManageDomains	Update	View	Build	Cancel	Configure	Create	Delete	Discover	Move	Read	Workspace	Delete	Replay	Update	Configure	Create	Delete	Read	Tag	HealthCheck	ThreadDump	View
development	"dev-.*"							✓		✓	✓				✓				✓								
devlopment-readonly	"dev-.*"														✓												
devops	"devops-.*"							✓		✓	✓				✓				✓								
testing	"test-.*"							✓		✓	✓				✓				✓								
testing-readonly	"test-.*"														✓												

Assign Roles

Global roles

User/Group	admin	employee
Anonymous		
Authenticated Users		
developer-1		✓
developer-2		✓
devops-1		✓
devops-2		✓
testing-1		✓
testing-2		✓
Aditya Kumar	✓	

Add User   Add Group

Item roles

User/Group	development	devlopment-readonly	devops	testing	testing-readonly
Anonymous					
Authenticated Users					
developer-1	✓				
developer-2	✓				
devops-1		✓	✓		✓
devops-2		✓	✓		✓
testing-1		✓		✓	
testing-2		✓		✓	

TASK 4:
Create a pipeline to execute a shell script
       * on git
       * on scripting task
       * Monitor disk utilization and send mail if > 80%
       * Process Management

Here I have made a script in which it monitors the memory utilization and then via my command line interface uploaded to my git repository .

After that I have initiated a jenkins pipeline and linked it with by git hub repository and executed my file there.

Dashboard > TASK MONITORING > #3

🗐 Status
</> Changes
▣ Console Output
✎ Edit Build Information
🗑 Delete build '#3'
⏱ Timings
◈ Git Build Data
⅄ Pipeline Overview
▣ Pipeline Console
↻ Restart from Stage
↪ Replay
:≣ Pipeline Steps
🗀 Workspaces

✓ **Build #3 (Nov 27, 2024, 11:01:41 AM)**

Started by user Aditya Kumar

This run spent:

- 37 ms waiting;
- 3.5 sec build duration;
- 3.5 sec total from scheduled to completion.

◈ git   **Revision**: 725b90ea0ff2104b054b3412cd138958e11d7ae8
       Repository: https://github.com/Rootcoderafk/DEVOPS.git

- refs/remotes/origin/main

```
Advanced ∨

Pipeline

Definition

Pipeline script                                                    ∨

Script  ?
   1 ▾ pipeline {
   2        agent any
   3 ▾      stages {
   4 ▾          stage('CHECKOUT SCRIPT') {
   5 ▾              steps {
   6                    git branch: 'main',
   7                        changelog: false,
   8                        poll: false,
   9                        url: 'https://github.com/Rootcoderafk/DEVOPS.git'
  10                }
  11            }
  12
  13 ▾          stage('RUN SCRIPT') {
  14 ▾              steps {
  15                    sh './monitor.sh'   // Run the shell script
  16                }
  17            }
  18        }
  19    }
  20

☑ Use Groovy Sandbox  ?
```

These are the pipeline ss whereas the script is as follows.
click here

TASK 5:
create another shell script that automates the backup of important files and directories to a backup location.

This was a basic shell scripting I created a script and then accepted input from user for the file location and destination and according to that I have managed to complete the script.

You can verify the code and check it by executing the script click here and after every time you run the script a entry in the log file would be made which keeps managing itself by deleting files after 7 days. You can check the log file click here

Thank You,
Aditya Kumar
E23CSEU0196