

# Jenkins

## Assignment – 1

### Task to be performed

1. Trigger a pipeline using Git when push on develop branch
2. Pipeline should pull Git content to a folder

### Steps and Commands

- Create 2 instance names them as master and slave
- Connect to both master and slave instance
- Update both the machine by running the command **sudo apt update**
- Next install Java in both master and slave by running the command **sudo apt install openjdk-11-jdk -y**
- To see the java installed in the both the machine by running the command **java --version**
- Navigate to master instance
- Next create a file to install Jenkins by running the command **sudo nano (provide the name for creating file.sh)**
- Provide the necessary command to install Jenkins
  - a) **sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \ https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key**
  - b) **echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \ https://pkg.jenkins.io/debian-stable binary/ | sudo tee \ /etc/apt/sources.list.d/jenkins.list > /dev/null**
  - c) **sudo apt-get update**
  - d) **sudo apt-get install jenkins -y**
- To run the commands which was provided in file by running the command **sudo bash (provide the name for created file.sh)**
- To see the Jenkins installed in the machine by running the command **jenkins --version**
- Now to see the Jenkins page put the (**public IP Address:8080**) of the master instance in browser Jenkins pages will be reflected
- Navigate to Jenkins page in browser
- Provide the necessary configuration
- Provide the user name, password and email address in Jenkins page
- Now the Jenkins dashboard page will be reflected
- Go inside manage Jenkins and click the nodes
- To create a node click the new node

- Provide the name for the node
- Provide /home/Ubuntu/Jenkins/ in the remote root directory
- Select launch agents via SSH in launch method
- Provide the private IP DNS name of slave instance in hosts
- Provide the credentials and necessary configuration
- Finally click on save
- Navigate to master instance
- Create a directory by running the command **mkdir (directory name)**
- Get into the directory by running the command **cd (directory name)**
- To Initialize the git by running the command **git init**
- To create the files by running the command **touch (creating file name)**
- To stage files by running the command **git add (created file name)**
- And to check the files which are in status by running the command **git status**
- To push the files to commit stage by running command **git commit -m "necessary notes"**
- Now the master branch will be created and to see the master branch by running the command **git branch**
- To create develop branches by running command **git branch develop**
- To check the list of branch by running command **git branch**
- Next to switch from master branch to develop branch by running the command **git checkout develop**
- If we run the command **ls** in develop branch the files which was created in master branch will be reflected
- To create the files by running the command **touch (creating file name)**
- To stage files by running the command **git add (created file name)**
- And to check the files which are in status by running the command **git status**
- To push the files to commit stage by running command **git commit -m "necessary notes"**
- Navigate to Github account
- Go Inside the **GITHUB** account
  - a) Click the New in the top repositories
  - b) Provide the name for repository
  - c) Finally click the create repository
- Navigate to master machine
- Next to link the repository with local machine by running the command **git remote add origin (repository URL link)**
- To push the all the branches and files into the repository by running the command **git push --all**
- Provide the username and password token which was assigned by github
- Navigate to github repository which was created

- Now all the branches and files which was created in master machine will be reflected in Github repository in Github account
- Navigate to Jenkins pages in browser
- To create the job click on new item
- Provide the name for the Job
- Provide the (**repository URL link**) in the Github project
- Provide the node which was created in restrict where this project can be run
- Provide the (**repository URL link**) in source code management under git repositories
- Provide develop branch in branches to build
- Navigate to github repository which was created
- To create the webhook by following steps
  - a) Click the settings of created repository
  - b) Go inside the webhooks and click the add webhook
  - c) Provide the Jenkins page URL/github-webhook/
  - d) Finally click on add webhook
- Navigate to Jenkins pages in browser
- Select the Github hook trigger for GITScm polling in build triggers
- Finally click on apply and save
- Navigate to master instance
- Next to switch from master branch to develop branch by running the command **git checkout develop**
- To create the files by running the command **touch (creating file name)**
- To stage files by running the command **git add (created file name)**
- And to check the files which are in status by running the command **git status**
- To push the files to commit stage by running command **git commit -m "necessary notes"**
- To push the develop branches and files into the repository by running the command **git push origin develop**
- Provide the username and password token which was assigned by github
- Navigate to github repository which was created
- Now the file created in develop branch which was created in master machine will be reflected in Github repository in Github account
- Navigate to Jenkins pages in browser
- Now the job which was created will be triggered and run in the node which was created because push was made to repository in develop branch
- Navigate to slave machine
- To see the file in slave machine which was created in master machine and pushed to github repository by running the command **cd Jenkins/workspace/job/**

- To see the file by running the command **ls** the file which was created in the master machine will be reflected in slave machine with the help of Jenkins

```
Swap usage: 0%
Expanded Security Maintenance for Applications is not enabled.
0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-53-47:~$ sudo apt update

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

ubuntu@ip-172-31-53-47:~$ sudo apt-get install openjdk-11-jdk -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
alsa-topology-conf alsuaucm-conf at-spi2-core ca-certificates-java dconf-gsettings-backend fontconfig-config
fonts-dejavu-core fonts-dejavu-extra gsettings-desktop-schemas java-common libasound2 libasound2-data libatk-bridge2.0-0
libatk-wrapper-java libatk-wrapper-java-jni libatk1.0-0 libatk1.0-0-data libatspi2.0-0 libavahi-client3 libavahi-common-data
libavahi-common3 libcurl5 libcurl5-dev libdconf1 libdrm-amdgpu libdrm-intel libdrm-nouveau2 libdrm-radeon1 libfontconfig1 libfontenc1 libgif7 libgl1
libgl1-amber-dri libgl1-mesa-dri libglapi-mesa libglvnd0 libglx0 libgraphite2-3 libharfbuzz0b libice-dev libice6
libjpegturbo libjpegturbo liblcms2-2 liblvm15 libpciaccess0 libpcsc-lite1 libpthread-stubs0-dev libsensors-config libsensors5 libsm-dev
libsm6 libx11-dev libx11-xcb1 libxau-dev libxaw7 libxcb-dri2-0 libxcb-dri3-0 libxcb-glx0 libxcb-present0 libxcb-randr0 libxcb-shape0
libxcb-shm0 libxcb-sync1 libxcb-xfixes0 libxcb1-dev libcomposite1 libxdmcp-dev libxf86libxft2 libxi6 libxinerama1 libxkbfile1
libxml2 libxml2-dev libxrender1 libxshmfence1 libxt-dev libxt6 libxtst6 libxv1 libxxf86dg1 libxxf86vm1 openjdk-11-jdk-headless
openjdk-11-jre openjdk-11-jre-headless session-migration x11-common x11-utils x11proto-dev xorg-sgml-doctools xtrans-dev
Suggested packages:
default-jre libasound2-plugins alsu-utils cups-common libice-doc liblcms2-utils pcscd lm-sensors libsm-doc libx11-doc libxcb-doc
libxt-doc openjdk-11-demo openjdk-11-source visualvm libnss-mdns fonts-ipafont-gothic fonts-ipafont-mincho fonts-wqy-microhei
fonts-wqy-zenhei fonts-indic mesa-utils
The following NEW packages will be installed:
alsa-topology-conf alsuaucm-conf at-spi2-core ca-certificates-java dconf-gsettings-backend fontconfig-config
fonts-dejavu-core fonts-dejavu-extra gsettings-desktop-schemas java-common libasound2 libasound2-data libatk-bridge2.0-0
libatk-wrapper-java libatk-wrapper-java-jni libatk1.0-0 libatk1.0-0-data libatspi2.0-0 libavahi-client3 libavahi-common-data
libavahi-common3 libcurl5 libcurl5-dev libdconf1 libdrm-amdgpu libdrm-intel libdrm-nouveau2 libdrm-radeon1 libfontconfig1 libfontenc1 libgif7 libgl1
libgl1-amber-dri libgl1-mesa-dri libglapi-mesa libglvnd0 libglx0 libgraphite2-3 libharfbuzz0b libice-dev libice6

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

GNU nano 6.2                                     task.sh *
sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \
https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key
echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \
https://pkg.jenkins.io/debian-stable binary/ | sudo tee \
/etc/apt/sources.list.d/jenkins.list > /dev/null
sudo apt-get update
sudo apt-get install jenkins -y

^G Help   ^O Write Out  ^W Where Is  ^R Cut      ^I Execute  ^C Location  M-U Undo  M-A Set Mark  M-J To Bracket
^X Exit   ^F Read File  ^V Replace  ^U Paste    ^J Justify  ^Y Go To Line M-E Redo  M-G Copy   ^Q Where Was

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
```

```
ubuntu@ip-172-31-53-47:~$ sudo nano task.sh  
ubuntu@ip-172-31-53-47:~$ sudo bash task.sh
```

```
ubuntu@ip-172-31-53-47:~$ jenkins --version
2.440.3
ubuntu@ip-172-31-53-47:~$ java --version
openjdk 11.0.22 2024-01-16
OpenJDK Runtime Environment (build 11.0.22+7-post-Ubuntu-0ubuntu222.04.1)
OpenJDK 64-Bit Server VM (build 11.0.22+7-post-Ubuntu-0ubuntu222.04.1, mixed mode, sharing)
ubuntu@ip-172-31-53-47:~$
```

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

The screenshot shows the Jenkins dashboard. At the top left is the Jenkins logo and the word "Jenkins". To the right is a search bar with placeholder text "Search (CTRL+K)" and a help icon. Further right are icons for notifications (bell), a shield, and user status, followed by the text "jenkinss" and a "log out" button. Below the header, the word "Dashboard" is followed by a right-pointing arrow. On the left side, there's a sidebar with several items: "+ New Item", "People" (with a user icon), "Build History" (with a document icon), "Manage Jenkins" (with a gear icon), and "My Views" (with a folder icon). Below this is a box titled "Build Queue" with a dropdown arrow; it contains the message "No builds in the queue.". At the bottom left is another box titled "Build Executor Status" with a dropdown arrow; it shows "1 Idle". The main content area has a title "Welcome to Jenkins!" with a "Add description" link. It includes a descriptive paragraph about displaying Jenkins jobs and starting projects. Below this is a section titled "Start building your software project" with a "Create a job" button and a "+" icon. There are also links for "Set up a distributed build", "Set up an agent", and "Configure a cloud", each accompanied by a small icon.

```
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-57-161:~$ sudo apt update
```

i-0f8cb97731801d164 (Jenkins-slave1)

PublicIPs: 100.25.143.216 PrivateIPs: 172.31.57.161

[CloudShell](#) [Feedback](#)

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

```
ubuntu@ip-172-31-57-161:~$ sudo apt-get install openjdk-11-jdk -y
```

i-0f8cb97731801d164 (Jenkins-slave1)

PublicIPs: 100.25.143.216 PrivateIPs: 172.31.57.161

[CloudShell](#) [Feedback](#)

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

```
ubuntu@ip-172-31-57-161:~$ java --version
openjdk 11.0.22 2024-01-16
OpenJDK Runtime Environment (build 11.0.22+7-post-Ubuntu-0ubuntu222.04.1)
OpenJDK 64-Bit Server VM (build 11.0.22+7-post-Ubuntu-0ubuntu222.04.1, mixed mode, sharing)
ubuntu@ip-172-31-57-161:~$
```

i-0f8cb97731801d164 (Jenkins-slave1)

PublicIPs: 100.25.143.216 PrivateIPs: 172.31.57.161

[CloudShell](#) [Feedback](#)

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)



# Jenkins

Search (CTRL+K) ? ! ! jenkinss log out

Dashboard > Manage Jenkins > Nodes > New node

## New node

Node name

Type

Permanent Agent

Adds a plain, permanent agent to Jenkins. This is called "permanent" because Jenkins doesn't provide higher level of integration with these agents, such as dynamic provisioning. Select this type if no other agent types apply — for example such as when you are adding a physical computer, virtual machines managed outside Jenkins, etc.

Create

Dashboard > Manage Jenkins > Nodes >

Description ?

Plain text [Preview](#)

Number of executors ?

Remote root directory ?

! Remote directory is mandatory

Labels ?

Save

Dashboard > Manage Jenkins > Nodes >

Launch

### Jenkins Credentials Provider: Jenkins

Username

Treat username as secret ?

Private Key

Enter directly

Key

Enter New Secret Below

```
-----BEGIN RSA PRIVATE KEY-----  
MIGfMA0GCSqGSIb3DQEBAQEAoB...  
-----END RSA PRIVATE KEY-----
```

Save

Dashboard > Manage Jenkins > Nodes >

Launch method ?

Launch agents via SSH

Host ?  
ip-172-31-57-161.ec2.internal

Credentials ?  
ubuntu

+ Add ▾

Host Key Verification Strategy ?  
Known hosts file Verification Strategy

Advanced ▾

**Save**

Jenkins

Search (CTRL+K) ? 🔍 🔔 ⚡ 2 jenkinsss log out

Dashboard > Manage Jenkins > Nodes >

**Nodes**

+ New Node Configure Monitors ⚡

S	Name ↓	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
💻	Built-In Node	Linux (amd64)	In sync	4.66 GiB	! 0 B	4.66 GiB	0ms 🛡️
💻	slave1	Linux (amd64)	In sync	5.19 GiB	! 0 B	5.19 GiB	32ms 🛡️
		Data obtained	1 min 49 sec	1 min 49 sec	1 min 49 sec	1 min 49 sec	1 min 49 sec

Build Queue  
No builds in the queue.

Build Executor Status  
Built-In Node  
1 Idle  
2 Idle  
slave1  
1 Idle

Icon: S M L

```
ubuntu@ip-172-31-57-161:~$ ls
jenkins
ubuntu@ip-172-31-57-161:~$
```



```
ubuntu@ip-172-31-57-161:~$ ls
jenkins
ubuntu@ip-172-31-57-161:~$ cd jenkins
ubuntu@ip-172-31-57-161:~/jenkins$ ls
remoting  remoting.jar
ubuntu@ip-172-31-57-161:~/jenkins$
```

i-0f8cb97731801d164 (Jenkins-slave1)

Public IPs: 100.25.143.216 Private IPs: 172.31.57.161

X

[CloudShell](#) [Feedback](#)

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

```
ubuntu@ip-172-31-53-47:~/task$ git init
Reinitialized existing Git repository in /home/ubuntu/task/.git/
ubuntu@ip-172-31-53-47:~/task$
```

[CloudShell](#) [Feedback](#)

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

```
ubuntu@ip-172-31-53-47:~/task$ git init
Reinitialized existing Git repository in /home/ubuntu/task/.git/
ubuntu@ip-172-31-53-47:~/task$ touch masterfile
ubuntu@ip-172-31-53-47:~/task$ ls
masterfile
ubuntu@ip-172-31-53-47:~/task$ git add masterfile
ubuntu@ip-172-31-53-47:~/task$ git commit -m "this ismasterfile"
[master (root-commit) 145ff3b] this ismasterfile
Committer: Ubuntu <ubuntu@ip-172-31-53-47.ec2.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

  git config --global --edit

After doing this, you may fix the identity used for this commit with:

  git commit --amend --reset-author

1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 masterfile
ubuntu@ip-172-31-53-47:~/task$
```

[CloudShell](#) [Feedback](#)

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

```
ubuntu@ip-172-31-53-47:~/task$ ls
masterfile
ubuntu@ip-172-31-53-47:~/task$ git branch
* master
ubuntu@ip-172-31-53-47:~/task$ git branch develop
  develop
* master
ubuntu@ip-172-31-53-47:~/task$ git checkout develop
Switched to branch 'develop'
ubuntu@ip-172-31-53-47:~/task$ ls
masterfile
ubuntu@ip-172-31-53-47:~/task$ touch developfile
ubuntu@ip-172-31-53-47:~/task$ git add developfile
```

[CloudShell](#) [Feedback](#)

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

```
* master
ubuntu@ip-172-31-53-47:~/task$ git checkout develop
Switched to branch 'develop'
ubuntu@ip-172-31-53-47:~/task$ ls
masterfile
ubuntu@ip-172-31-53-47:~/task$ touch developfile
ubuntu@ip-172-31-53-47:~/task$ git add developfile
ubuntu@ip-172-31-53-47:~/task$ git commit -m "this is developfile"
[develop ac22a8b] this is developfile
  Committer: Ubuntu <ubuntu@ip-172-31-53-47.ec2.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

  git config --global --edit

After doing this, you may fix the identity used for this commit with:

  git commit --amend --reset-author

1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 developfile
ubuntu@ip-172-31-53-47:~/task$
```

[CloudShell](#) [Feedback](#)

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

```
ubuntu@ip-172-31-53-47:~/task$ git remote add origin https://github.com/new001001001/assignment.git
ubuntu@ip-172-31-53-47:~/task$ git push --all
Username for 'https://github.com': new001001001
Password for 'https://new001001001@github.com':
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 2 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (5/5), 454 bytes | 454.00 KiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/new001001001/assignment.git
 * [new branch]      develop -> develop
 * [new branch]      master -> master
ubuntu@ip-172-31-53-47:~/task$
```

[CloudShell](#) [Feedback](#)

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

The screenshot shows a GitHub repository page for 'assignment'. At the top, there's a search bar with placeholder text 'Type [ ] to search'. Below the search bar are navigation links: Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. The repository name 'assignment' is displayed with a yellow gear icon, and it's marked as 'Public'. Below the repository name, there are buttons for Pin, Unwatch (with a count of 1), Fork (with a count of 0), Star (with a count of 0), and a settings gear icon. The repository has 2 branches and 0 tags. A commit list shows three commits from 'Ubuntu': 'this is developfile' (28 minutes ago), 'developfile' (28 minutes ago), and 'masterfile' (34 minutes ago). On the right side, there's an 'About' section with a note: 'No description, website, or topics provided.' It also shows activity metrics: 2 commits, 0 stars, 1 watching, and 0 forks. Below the 'About' section are sections for 'Releases' (no releases published) and 'Packages'.

The screenshot shows the Jenkins dashboard. At the top, there's a search bar with placeholder text 'Search (CTRL+K)' and a log out link. Below the search bar, there are notifications for 1 build and 2 errors. The main area is titled 'Enter an item name' and contains a text input field with 'job1' typed in. Below the input field, there's a note: '» Required field'. There are three project creation options listed: 'Freestyle project' (represented by a house icon), 'Pipeline' (represented by a blue ribbon icon), and 'Multi-configuration project' (represented by a blue button icon). The 'Freestyle project' description states: 'Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.' The 'Pipeline' description states: 'Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.' The 'Multi-configuration project' description states: 'Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific'.

The screenshot shows the Jenkins configuration screen for 'job1'. The URL is 'Dashboard > job1 > Configuration'. The configuration tabs include General, Source Code Management, Build Triggers, Build Environment, Build Steps, and Post-build Actions. The 'General' tab is selected and highlighted in grey. Under the 'General' tab, there are several configuration options: 'Discard old builds' (unchecked), 'GitHub project' (checked), 'Project url' (set to 'https://github.com/new001001001/assignment.git'), 'Advanced' (a dropdown menu), 'This project is parameterized' (unchecked), 'Throttle builds' (unchecked), 'Execute concurrent builds if necessary' (unchecked), and 'Restrict where this project can be run' (unchecked). At the bottom of the configuration screen are 'Save' and 'Apply' buttons.

Dashboard > job1 > Configuration

## Configure

Execute concurrent builds if necessary ?

Restrict where this project can be run ?

Label Expression ?

slave1

Label slave1 matches 1 node. Permissions or other restrictions provided by plugins may further reduce that list.

Advanced ▾

**Source Code Management**

None

Git ?

**Save** **Apply**

This screenshot shows the Jenkins configuration interface for a job named 'job1'. In the 'General' section, there is a checkbox for 'Restrict where this project can be run?' which is checked. Below it, a 'Label Expression' field contains 'slave1'. A note states that this label matches one node and that plugin permissions might further restrict it. Under 'Source Code Management', the 'None' option is selected. At the bottom are 'Save' and 'Apply' buttons.

Dashboard > job1 > Configuration

## Configure

General

Source Code Management

Build Triggers

Build Environment

Build Steps

Post-build Actions

Advanced ▾

Add Repository

Branches to build ?

Branch Specifier (blank for 'any') ?

\*/Develop

Add Branch

Repository browser ?

(Auto)

**Save** **Apply**

This screenshot shows the Jenkins configuration interface for a job named 'job1'. In the 'Source Code Management' section, the 'Add Repository' button is visible. Below it, a 'Branches to build' section has a 'Branch Specifier' field containing '\*/Develop'. There is also an 'Add Branch' button and a 'Repository browser' dropdown set to '(Auto)'. At the bottom are 'Save' and 'Apply' buttons.

new001001001 / assignment

Type ⌥ to search

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

General

Access

Collaborators

Moderation options

Code and automation

Branches

Tags

Rules

Actions

Webhooks

Environments

Codespaces

Webhooks / Add webhook

We'll send a POST request to the URL below with details of any subscribed events. You can also specify which data format you'd like to receive (JSON, x-www-form-urlencoded, etc). More information can be found in our developer documentation.

Payload URL \*

http://54.89.117.86:8080/github-webhook/

Content type

application/x-www-form-urlencoded

Secret

Which events would you like to trigger this webhook?

This screenshot shows the GitHub repository settings for a repository named 'new001001001'. On the left, a sidebar lists 'General', 'Access', 'Collaborators', 'Moderation options', 'Code and automation', 'Branches', 'Tags', 'Rules', 'Actions', 'Webhooks', 'Environments', and 'Codespaces'. The 'Webhooks' section is currently active. On the right, a 'Webhooks / Add webhook' form is displayed. It includes fields for 'Payload URL' (set to 'http://54.89.117.86:8080/github-webhook/'), 'Content type' (set to 'application/x-www-form-urlencoded'), and a 'Secret' field. A note at the top explains that the webhook will send POST requests with event details. At the bottom, a question asks 'Which events would you like to trigger this webhook?'. The top navigation bar shows links for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings.

application/x-www-form-urlencoded

Secret

Which events would you like to trigger this webhook?

Just the push event.

Send me everything.

Let me select individual events.

Active

We will deliver event details when this hook is triggered.

Add webhook

© 2024 GitHub, Inc. Terms Privacy Security Status Docs Contact Manage cookies Do not share my personal information

Dashboard > job1 > Configuration

## Configure

General

Source Code Management

Build Triggers

Build Environment

Build Steps

Add build step ▾

Post-build Actions

Add post-build action ▾

Save Apply

```
ubuntu@ip-172-31-53-47:~/task$ git branch
* develop
  master
ubuntu@ip-172-31-53-47:~/task$ touch newfile
ubuntu@ip-172-31-53-47:~/task$ git add newfile
ubuntu@ip-172-31-53-47:~/task$ git commit -m "newfile"
[develop 203c955] newfile
Committer: Ubuntu <ubuntu@ip-172-31-53-47.ec2.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

git config --global --edit

After doing this, you may fix the identity used for this commit with:

git commit --amend --reset-author

1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 newfile
ubuntu@ip-172-31-53-47:~/task$
```

```

ubuntu@ip-172-31-53-47:~/task$ git branch
* develop
  master
ubuntu@ip-172-31-53-47:~/task$ git push origin develop
Username for 'https://github.com': new001001001
Password for 'https://new001001001@github.com':
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 2 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 238 bytes | 238.00 KiB/s, done.
Total 2 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/new001001001/assignment.git
 ac22a8b..203c955  develop -> develop
ubuntu@ip-172-31-53-47:~/task$ 

```

CloudShell   Feedback   © 2024, Amazon Web Services, Inc. or its affiliates.   Privacy   Terms   Cookie preferences

GitHub repository screenshot showing code, issues, pull requests, actions, projects, wiki, security, insights, and settings tabs. It shows a commit from Ubuntu with a message "this is developfile" and another from masterfile with "this ismasterfile". A README file is present.

Jenkins Console Output screenshot showing build logs for job1 #3. The logs show the Jenkins pipeline executing Git commands to clone the repository and check out the develop branch.

```
ubuntu@ip-172-31-57-161:~/jenkins$ ls
remoting  remoting.jar  workspace
ubuntu@ip-172-31-57-161:~/jenkins$ cd workspace
ubuntu@ip-172-31-57-161:~/jenkins/workspace$ ls
job1
ubuntu@ip-172-31-57-161:~/jenkins/workspace$ cd job1
ubuntu@ip-172-31-57-161:~/jenkins/workspace/job1$ ls
developfile  masterfile  newfile
ubuntu@ip-172-31-57-161:~/jenkins/workspace/job1$
```

i-0f8cb97731801d164 (Jenkins-slave1)

Public IPs: 100.25.143.216 Private IPs: 172.31.57.161

X

 CloudShell    Feedback

© 2024, Amazon Web Services, Inc. or its affiliates.    Privacy    Terms    Cookie preferences

## Assignment – 2

### Task to be performed

1. Add 2 nodes to Jenkins master
2. Create 2 jobs with the following job
  - a) Push to test
  - b) Push to prod
3. Once a push is made to test branch, copy git files to test server
4. Once a push is made to master branch, copy git files to prod server

### Steps and Commands

- Create 2 instance names them as test server and prod server
- Connect to both test server and prod server instance
- Update both the machine by running the command **sudo apt update**
- Next install Java in both test server and prod server instance by running the command **sudo apt install openjdk-11-jdk -y**
- To see the java installed in the both the machine by running the command **java --version**
- Navigate to master instance
- Go inside manage Jenkins and click the nodes
- To create a 2 node click the new node
- Provide the name for the node as test node and prod node
- Provide /home/Ubuntu/Jenkins/ in the remote root directory
- Select launch agents via SSH in launch method
- Provide the private IP DNS name of test server instance in test node and prod server instance in prod node in hosts
- Provide the credentials and necessary configuration
- Finally click on save
- Now test node and prod node have been created
- Navigate to master instance
- Create a directory by running the command **mkdir (directory name)**
- Get into the directory by running the command **cd (directory name)**
- To Initialize the git by running the command **git init**
- To create the files by running the command **touch (creating file name)**
- To stage files by running the command **git add (created file name)**
- And to check the files which are in status by running the command **git status**
- To push the files to commit stage by running command **git commit -m "necessary notes"**

- Now the master branch will be created and to see the master branch by running the command **git branch**
- To create test branches by running command **git branch test**
- To check the list of branch by running command **git branch**
- Next to switch from master branch to test branch by running the command **git checkout test**
- If we run the command **ls** in test branch the files which was created in master branch will be reflected
- To create the files by running the command **touch (creating file name)**
- To stage files by running the command **git add (created file name)**
- And to check the files which are in status by running the command **git status**
- To push the files to commit stage by running command **git commit -m "necessary notes"**
- Navigate to Github account
- Go Inside the **GITHUB** account
  - d) Click the New in the top repositories
  - e) Provide the name for repository
  - f) Finally click the create repository
- Navigate to master machine
- Next to link the repository with local machine by running the command **git remote add origin (repository URL link)**
- To push the all the branches and files into the repository by running the command **git push --all**
- Provide the username and password token which was assigned by github
- Navigate to github repository which was created
- Now the all the branches and files which was created in master machine will be reflected in Github repository in Github account
- Navigate to Jenkins pages in browser
- To create 2 job click on new item
- Provide the name for the job as test-Job and prod-Job
- Provide the **(repository URL link)** in the Github project
- Provide the test node for test-job and prod node for prod-job which was created in restrict where this project can be run
- Provide the **(repository URL link)** in source code management under git repositories
- Provide test branch for test-job and master branch for job-prod in branches to build
- Navigate to github repository which was created
- To create the webhook by following steps
  - e) Click the settings of created repository
  - f) Go inside the webhooks and click the add webhook
  - g) Provide the Jenkins page URL/github-webhook/

h) Finally click on add webhook

- Navigate to Jenkins pages in browser
- Select the Github hook trigger for GITScm polling in build triggers
- Finally click on apply and save
- Now test-Job and prod-Job have been created
- Navigate to master instance
- To create the files in master branch by running the command **touch prodfile**
- To stage files by running the command **git add prodfile**
- And to check the files which are in status by running the command **git status**
- To push the files to commit stage by running command **git commit -m "necessary notes"**
- Next to switch from master branch to test branch by running the command **git checkout test**
- To create the files by running the command **touch testingfile**
- To stage files by running the command **git add testingfile**
- And to check the files which are in status by running the command **git status**
- To push the files to commit stage by running command **git commit -m "necessary notes"**
- To push the all the branches and files into the repository by running the command **git push --all**
- Provide the username and password token which was assigned by github
- Navigate to github repository which was created
- Now the all branch and file which was in master machine instance will be reflected in Github repository in Github account
- Navigate to Jenkins pages in browser
- Now 2 job which was created will be triggered test-job will be run in test node and prod-Job will be run in prod node which was created because push was made to the repository
- Navigate to test server
- To see the testingfile in test server machine by running the command **cd Jenkins/workspace/test-job/**
- To see the testingfile by running the command **ls**
- Navigate to prod server
- To see the prodfile in test server machine by running the command **cd Jenkins/workspace/prod-job/**
- To see the prodfile by running the command **ls**

```
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-31-196:~$ sudo apt update
```

i-0a681404964306096 (test\_server)  
PublicIPs: 34.229.150.0 PrivateIPs: 172.31.31.196

[CloudShell](#) [Feedback](#) © 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

```
ubuntu@ip-172-31-31-196:~$ sudo apt-get install openjdk-11-jdk -y
```

i-0a681404964306096 (test\_server)  
PublicIPs: 34.229.150.0 PrivateIPs: 172.31.31.196

[CloudShell](#) [Feedback](#) © 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

```
ubuntu@ip-172-31-31-196:~$ java --version
openjdk 11.0.22 2024-01-16
OpenJDK Runtime Environment (build 11.0.22+7-post-Ubuntu-0ubuntu22.04.1)
OpenJDK 64-Bit Server VM (build 11.0.22+7-post-Ubuntu-0ubuntu22.04.1, mixed mode, sharing)
ubuntu@ip-172-31-31-196:~$
```

i-0a681404964306096 (test\_server)  
PublicIPs: 34.229.150.0 PrivateIPs: 172.31.31.196

[CloudShell](#) [Feedback](#) © 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

```
System information as of Sat Apr 27 15:08:30 UTC 2024

System load: 0.19287109375    Processes:          113
Usage of /: 20.5% of 7.57GB   Users logged in:      0
Memory usage: 21%           IPv4 address for eth0: 172.31.23.194
Swap usage:  0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update
```

i-0214a6c68f09e2058 (prod\_server)  
PublicIPs: 34.207.60.16 PrivateIPs: 172.31.23.194

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

```
ubuntu@ip-172-31-23-194:~$ sudo apt-get install openjdk-11-jdk -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  alsatopology-conf alsau-cumc-conf at-spi2-core ca-certificates-java dconf-gsettings-backend dconf-service fontconfig-config
  fonts-dejavu-core fonts-dejavu-extra gsettings-desktop-schemas java-common libasound2 libasound2-data libatk-bridge2.0-0
  libatk-wrapper-java libatk-wrapper-java-jni libatk1.0-0 libatkspi2.0-0 libavahi-client3 libavahi-common-data
  libavahi-common3 libcupsc2 libdconf1 libdrm-amdgpu libdrm-intel libdrm-nouveau2 libdrm-radeon1 libfontconfig libfontenc1 libgif7 libgl1
  libgl1-amber-dri libgl1-mesa-dri libglapi-mesa libglvnd0 libglx-mesa0 libglx0 libgraphite2-3 libharfbuzz0b libice-dev libice6
  libjpeg-turbo8 libjpeg8 liblcms2-2 liblilmv15 libpaciaccesso libpcsccli1 libpthread-stubs0-dev libsensors-conf libsm-dev
  libsm6 libx11-dev libx11-xcb1 libxau-dev libxaw7 libxcb-dr1-0 libxcb-dr1-3 libxcb-glx0 libxcb-present0 libxcb-randr0 libxcb-shape0
  libxcb-shm0 libxcb-sync1 libxcb-xfixes0 libxcb1-dev libxcompositel libxdmcp-dev libxfixed3 libxf2 libxi6 libxinerama1 libxkbfile1
  libxml2 libxpm4 libxrandr2 libxrender1 libxshmfence1 libxt-dev libxt6 libxtst6 libxv1 libxf86dg1 libxf86vm1 openjdk-11-jdk-headless
  openjdk-11-jre openjdk-11-jre-headless session-migration x11-common x11-utils x11proto-dev xorg-sgml-doctools xtrans-dev
Suggested packages:
  default-jre libasound2-plugins alsau-utils cups-common libice-doc liblcms2-utils pscsd lm-sensors libsm-doc libx11-doc libxcb-doc
  libxt-doc openjdk-11-demo openjdk-11-source visualvm libnss-mdns fonts-ipafont-gothic fonts-ipafont-mincho fonts-wqy-microhei
```

i-0214a6c68f09e2058 (prod\_server)  
PublicIPs: 34.207.60.16 PrivateIPs: 172.31.23.194

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

```
ubuntu@ip-172-31-23-194:~$ java --version
openjdk 11.0.22 2024-01-16
OpenJDK Runtime Environment (build 11.0.22+7-post-Ubuntu-0ubuntu222.04.1)
OpenJDK 64-Bit Server VM (build 11.0.22+7-post-Ubuntu-0ubuntu222.04.1, mixed mode, sharing)
ubuntu@ip-172-31-23-194:~$
```

i-0214a6c68f09e2058 (prod\_server)  
PublicIPs: 34.207.60.16 PrivateIPs: 172.31.23.194

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences



# Jenkins

Search (CTRL+K) ? ! ! jenkins log out

Dashboard > Manage Jenkins > Nodes > New node

## New node

Node name

Type

Permanent Agent

Adds a plain, permanent agent to Jenkins. This is called "permanent" because Jenkins doesn't provide higher level of integration with these agents, such as dynamic provisioning. Select this type if no other agent types apply — for example such as when you are adding a physical computer, virtual machines managed outside Jenkins, etc.

**Create**

Dashboard > Manage Jenkins > Nodes >

Plain text [Preview](#)

Number of executors ?

Remote root directory ?

! Remote directory is mandatory

Labels ?

Usage ?

**Save**

Dashboard > Manage Jenkins > Nodes >

Usage ?

Launch method ?

Host ?

Credentials ?

+ Add ▼

Host Key Verification Strategy ?

**Save**

Dashboard > Manage Jenkins > Nodes >

Advanced ▾

Availability ?

Keep this agent online as much as possible

Node Properties

Disable deferred wipeout on this node ?

Disk Space Monitoring Thresholds

Environment variables

Tool Locations

**Save**

REST API Jenkins 2.440.3

## New node

Node name

prod-node

Type

 Permanent Agent

Adds a plain, permanent agent to Jenkins. This is called "permanent" because Jenkins doesn't provide higher level of integration with these agents, such as dynamic provisioning. Select this type if no other agent types apply — for example such as when you are adding a physical computer, virtual machines managed outside Jenkins, etc.

 Copy Existing Node**Create**

Number of executors ?

1

Remote root directory ?

/home/ubuntu/jenkins/

**! Remote directory is mandatory**

Labels ?

Usage ?

Use this node as much as possible

Launch method ?

**Save**

Launch method ?

Launch agents via SSH

Host ?

ip-172-31-23-194.ec2.internal

Credentials ?

ubuntu

+ Add ▾

Host Key Verification Strategy ?

Non verifying Verification Strategy

Advanced ▾

**Save**

Jenkins

Search (CTRL+K)

jenkinss log out

Dashboard > Manage Jenkins > Nodes >

Nodes

Clouds

Build Queue

No builds in the queue.

Build Executor Status

Built-In Node

1 Idle

2 Idle

prod-node

1 Idle

Nodes

+ New Node

Configure Monitors

S	Name	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
	Built-In Node	Linux (amd64)	In sync	4.58 GiB	! 0 B	4.58 GiB	0ms 🌐
	prod-node	Linux (amd64)	In sync	5.19 GiB	! 0 B	5.19 GiB	58ms 🌐
	test-node	Linux (amd64)	In sync	5.19 GiB	! 0 B	5.19 GiB	53ms 🌐

Data obtained 27 sec 27 sec 27 sec 27 sec 27 sec 27 sec

Icon: S M L

```
ubuntu@ip-172-31-31-196:~$ ls
jenkins
ubuntu@ip-172-31-31-196:~$ cd jenkins
ubuntu@ip-172-31-31-196:~/jenkins$ ls
remoting remoting.jar
ubuntu@ip-172-31-31-196:~/jenkins$
```

i-0a681404964306096 (test\_server)

PublicIPs: 34.229.150.0 PrivateIPs: 172.31.31.196

CloudShell Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

```
ubuntu@ip-172-31-23-194:~$ ls
jenkins
ubuntu@ip-172-31-23-194:~$ cd jenkins
ubuntu@ip-172-31-23-194:~/jenkins$ ls
remoting remoting.jar
ubuntu@ip-172-31-23-194:~/jenkins$
```

i-0214a6c68f09e2058 (prod\_server)

PublicIPs: 34.207.60.16 PrivateIPs: 172.31.23.194

CloudShell Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

```

ubuntu@ip-172-31-53-47:~/assignment2$ git init
Reinitialized existing Git repository in /home/ubuntu/assignment2/.git/
ubuntu@ip-172-31-53-47:~/assignment2$ touch masterfile
ubuntu@ip-172-31-53-47:~/assignment2$ git add masterfile
ubuntu@ip-172-31-53-47:~/assignment2$ git commit -m "masterfile"
[master (root-commit) a91b4fc] masterfile
Committer: Ubuntu <ubuntu@ip-172-31-53-47.ec2.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

git config --global --edit

After doing this, you may fix the identity used for this commit with:

git commit --amend --reset-author

1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 masterfile
ubuntu@ip-172-31-53-47:~/assignment2$ 

```

[CloudShell](#) [Feedback](#)

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

```

ubuntu@ip-172-31-53-47:~/assignment2$ ls
masterfile
ubuntu@ip-172-31-53-47:~/assignment2$ git branch test
ubuntu@ip-172-31-53-47:~/assignment2$ git branch
* master
  test
ubuntu@ip-172-31-53-47:~/assignment2$ git checkout test
Switched to branch 'test'
ubuntu@ip-172-31-53-47:~/assignment2$ touch testfile
ubuntu@ip-172-31-53-47:~/assignment2$ git add testfile
ubuntu@ip-172-31-53-47:~/assignment2$ git commit -m "testfile"
[test 9c9d578] testfile
Committer: Ubuntu <ubuntu@ip-172-31-53-47.ec2.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

git config --global --edit

After doing this, you may fix the identity used for this commit with:

git commit --amend --reset-author

```

[CloudShell](#) [Feedback](#)

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

## Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere?

[Import a repository.](#)

Required fields are marked with an asterisk (\*).

Owner \*

new001001001

Repository name \*

assignment2 is available.

Great repository names are short and memorable. Need inspiration? How about [fluffy-spoon](#)?

Description (optional)

Public

Anyone on the internet can see this repository. You choose who can commit.

Private

You choose who can see and commit to this repository.

---

Initialize this repository with:

[Add a README file](#)  
This is where you can write a long description for your project. [Learn more about READMEs.](#)

[Add .gitignore](#)  
.gitignore template: [None](#)

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

[Choose a license](#)  
License: [None](#)

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

ⓘ You are creating a public repository in your personal account.

[Create repository](#)

© 2024 GitHub, Inc. [Terms](#) [Privacy](#) [Security](#) [Status](#) [Docs](#) [Contact](#) [Manage cookies](#) [Do not share my personal information](#)

```
ubuntu@ip-172-31-53-47:~/assignment2$ git remote add origin https://github.com/new001001001/assignment2.git
ubuntu@ip-172-31-53-47:~/assignment2$ git push --all
Username for 'https://github.com': new001001001
Password for 'https://new001001001@github.com':
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 2 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (5/5), 437 bytes | 437.00 KiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/new001001001/assignment2.git
 * [new branch]      master -> master
 * [new branch]      test -> test
ubuntu@ip-172-31-53-47:~/assignment2$
```

[CloudShell](#) [Feedback](#) © 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

The screenshot shows a GitHub repository page for 'assignment2'. At the top, there's a navigation bar with links for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. Below the navigation bar, the repository name 'assignment2' is displayed, along with a 'Public' badge. A yellow banner at the top indicates that 'test' had recent pushes 30 seconds ago. On the left, there's a sidebar showing the 'master' branch, 2 branches, and 0 tags. The main content area shows a commit by 'Ubuntu' titled 'masterfile' with a timestamp of '6 minutes ago'. Below the commit, there's a link to 'README'. On the right side, there are sections for 'About', 'Releases', and 'Packages'. The 'About' section notes 'No description, website, or topics provided.' The 'Releases' section says 'No releases published' and has a link to 'Create a new release'. The 'Packages' section is currently empty.

The screenshot shows a GitHub repository page for 'assignment2'. At the top, there's a search bar and various navigation links like 'Code', 'Issues', 'Pull requests', 'Actions', 'Projects', 'Wiki', 'Security', 'Insights', and 'Settings'. Below the header, the repository name 'assignment2' is displayed with a 'Public' status. A yellow banner at the top indicates that 'test' had recent pushes 30 seconds ago. On the left, a sidebar shows the current branch is 'master' with 2 branches and 0 tags. A 'Switch branches/tags' dropdown is open, showing 'master' and 'test' with 'default' selected. The main area displays a file named 'masterfile' with a commit 'a91b4fc - 6 minutes ago' and '1 Commits'. To the right, sections for 'About', 'Activity', 'Releases', and 'Packages' are visible.

The screenshot shows the Jenkins configuration interface for a job named 'job1-test'. At the top, there's a search bar and user information. Below it, a form asks 'Enter an item name' with 'job1-test' entered. It lists two project types: 'Freestyle project' (selected) and 'Pipeline'. The 'Freestyle project' section includes a description and an 'OK Configuration project' button. The configuration page for 'job1-test' shows the 'General' tab selected. It includes fields for 'Source Code Management' (set to 'GitHub project' with URL 'https://github.com/new001001001/assignment2.git'), 'Build Triggers', 'Build Environment', 'Build Steps', and 'Post-build Actions'. At the bottom, there are 'Save' and 'Apply' buttons.

Dashboard > job1-test > Configuration

## Configure

General

Source Code Management

Build Triggers

Build Environment

Build Steps

Post-build Actions

Advanced

This project is parameterized ?

Throttle builds ?

Execute concurrent builds if necessary ?

Restrict where this project can be run ?

Label Expression ?

test-node

Label test-node matches 1 node. Permissions or other restrictions provided by plugins may further reduce that list.

Advanced

Save Apply

Dashboard > job1-test > Configuration

## Configure

None

Git ?

Repositories ?

Repository URL ?

https://github.com/new001001001/assignment2.git

Credentials ?

- none -

+ Add ▾

Advanced ▾

Save Apply

new001001001 / assignment2

Type ⌘ to search

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

General

Access

Collaborators

Moderation options

Code and automation

Branches

Tags

Rules

Actions

Webhooks

Environments

Codespaces

Webhooks / Add webhook

We'll send a post request to the URL below with details of any subscribed events. You can also specify which data format you'd like to receive (JSON, x-www-form-urlencoded, etc). More information can be found in our developer documentation.

Payload URL \*

http://35.153.50.175:8080/github-webhook/

Content type

application/x-www-form-urlencoded

Secret

Which events would you like to trigger this webhook?

Dashboard > job1-test > Configuration

With Ant ?

## Configure

General Source Code Management Build Triggers Build Environment Build Steps Post-build Actions

Add build step ▾ Add post-build action ▾

**Build Steps**

**Post-build Actions**

**Build Steps**

**Post-build Actions**

**Save** **Apply**

**Saved**

REST API Jenkins 2.44.0

Jenkins

Search (CTRL+K)  jenkinsss log out

Dashboard > All >

### Enter an item name

job2-prod » Required field

**Freestyle project**  
Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.

**Pipeline**  
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

**Multi-configuration project**  
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific

Dashboard > job2-prod > Configuration

## Configure

### General

Enabled

General Source Code Management Build Triggers Build Environment Build Steps Post-build Actions

Description  
push to prod

Plain text [Preview](#)

Discard old builds ?

GitHub project

**Save** **Apply**

Dashboard > job2-prod > Configuration

Plain text [Preview](#)

## Configure

Discard old builds ?

GitHub project

Project url ?

Advanced ▾

This project is parameterized ?

Throttle builds ?

Execute concurrent builds if necessary ?

Restrict where this project can be run ?

[Save](#) [Apply](#)

Dashboard > job2-prod > Configuration

## Configure

None

Git ?

Repositories ?

Repository URL ?  
  
① Please enter Git repository.

Credentials ?  
  
+ Add ▾

[Save](#) [Apply](#)

Dashboard > job2-prod > Configuration

## Configure

Advanced ▾

This project is parameterized ?

Throttle builds ?

Execute concurrent builds if necessary ?

Restrict where this project can be run ?

Label Expression ?  
  
Label prod-node matches 1 node. Permissions or other restrictions provided by plugins may further reduce that list.

Advanced ▾

Quiet period ?

[Save](#) [Apply](#)

Dashboard > job2-prod > Configuration

## Configure

General

Source Code Management

Build Triggers

Build Environment

Build Steps

Post-build Actions

Add repository

Branches to build ?

Branch Specifier (blank for 'any') ?

\*/master

Add Branch

Repository browser ?

(Auto)

Additional Behaviours

Add ▾

**Save** **Apply**

Dashboard > job2-prod > Configuration

## Configure

General

Source Code Management

Build Triggers

Build Environment

Build Steps

Post-build Actions

### Build Triggers

Trigger builds remotely (e.g., from scripts) ?

Build after other projects are built ?

Build periodically ?

GitHub hook trigger for GITScm polling ?

Poll SCM ?

### Build Environment

Delete workspace before build starts

Use secret text(s) or file(s) ?

Add timestamps to the Console Output

**Save** **Apply**

Dashboard > job2-prod > Configuration

## Configure

General

Source Code Management

Build Triggers

Build Environment

Build Steps

Post-build Actions

With Ant ?

### Build Steps

Add build step ▾

### Post-build Actions

Add post-build action ▾

**Save** **Apply**

Dashboard >

+ New Item Add description

People All +

Build History All +

S	W	Name	Last Success	Last Failure	Last Duration	More
...	...	job1-test	N/A	N/A	N/A	
...	...	job2-prod	N/A	N/A	N/A	

Project Relationship

Check File Fingerprint

Manage Jenkins

My Views

Icon: S M L Icon legend  Atom feed for all Atom feed for failures Atom feed for just latest builds

Build Queue ▼

No builds in the queue.

```
ubuntu@ip-172-31-53-47:~/assignment2$ git branch
* master
  test
ubuntu@ip-172-31-53-47:~/assignment2$ touch prodfile
ubuntu@ip-172-31-53-47:~/assignment2$ git add prodfile
ubuntu@ip-172-31-53-47:~/assignment2$ git commit -m "prodfile"
[master eef6b7e] prodfile
Committer: Ubuntu <ubuntu@ip-172-31-53-47.ec2.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

git config --global --edit

After doing this, you may fix the identity used for this commit with:

git commit --amend --reset-author

1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 prodfile
ubuntu@ip-172-31-53-47:~/assignment2$
```

CloudShell Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

```
ubuntu@ip-172-31-53-47:~/assignment2$ git checkout test
Switched to branch 'test'
ubuntu@ip-172-31-53-47:~/assignment2$ touch testingfile
ubuntu@ip-172-31-53-47:~/assignment2$ git add testingfile
ubuntu@ip-172-31-53-47:~/assignment2$ git commit -m "testingfile"
[test cf8eb13] testingfile
Committer: Ubuntu <ubuntu@ip-172-31-53-47.ec2.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

git config --global --edit

After doing this, you may fix the identity used for this commit with:

git commit --amend --reset-author

1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 testingfile
ubuntu@ip-172-31-53-47:~/assignment2$ ls
masterfile testfile testingfile
ubuntu@ip-172-31-53-47:~/assignment2$
```

CloudShell Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

```

ubuntu@ip-172-31-53-47:~/assignment2$ git push origin master
Username for 'https://github.com': new001001001
Password for 'https://new001001001@github.com':
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 2 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 249 bytes | 249.00 KiB/s, done.
Total 2 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/new001001001/assignment2.git
  a9lb4fc..eef6b7e master -> master
ubuntu@ip-172-31-53-47:~/assignment2$ git push origin test
Username for 'https://github.com': new001001001
Password for 'https://new001001001@github.com':
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 2 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 257 bytes | 257.00 KiB/s, done.
Total 2 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/new001001001/assignment2.git
  9c9d578..cf8eb13 test -> test
ubuntu@ip-172-31-53-47:~/assignment2$ 

```

[CloudShell](#) [Feedback](#) © 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

The screenshot shows the GitHub repository page for 'assignment2'. The repository is public and has two branches: 'master' and 'test'. The 'master' branch has one commit by 'Ubuntu' with the commit message 'prodfile' and timestamp 'eef6b7e - 8 minutes ago'. It also has a 'masterfile' file updated 41 minutes ago. The 'test' branch has one commit by 'Ubuntu' with the commit message 'prodfile' and timestamp '8 minutes ago'. There is a 'README' file. On the right, there's an 'About' section with no description, 0 stars, 1 watching, and 0 forks. The 'Releases' section shows no releases published. The 'Packages' section shows no packages published.

The screenshot shows the GitHub repository page for 'assignment2'. The repository is public and has two branches: 'test' and 'master'. The 'test' branch is ahead of 'master' by 2 commits. The 'test' branch has three commits by 'Ubuntu': 'testingfile' (timestamp 'cf8eb13 - 5 minutes ago'), 'testfile' (timestamp '38 minutes ago'), and 'testingfile' (timestamp '5 minutes ago'). It also has a 'masterfile' file updated 41 minutes ago and a 'README' file. On the right, there's an 'About' section with no description, 0 stars, 1 watching, and 0 forks. The 'Releases' section shows no releases published. The 'Packages' section shows no packages published.

Dashboard > job1-test > #2 > Console Output

Status

Changes

**Console Output**

View as plain text

Edit Build Information

Delete build '#2'

Polling Log

Git Build Data

← Previous Build

**Console Output**

```

Started by GitHub push by new001001001
Running as SYSTEM
Building remotely on test-node in workspace /home/ubuntu/jenkins/workspace/job1-test
The recommended git tool is: NONE
No credentials specified
> git rev-parse --resolve-git-dir /home/ubuntu/jenkins/workspace/job1-test/.git # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://github.com/new001001001/assignment2.git # timeout=10
Fetching upstream changes from https://github.com/new001001001/assignment2.git
> git --version # timeout=10
> git --version # 'git version 2.34.1'
> git fetch --tags --force --progress -- https://github.com/new001001001/assignment2.git
+refs/heads/*:refs/remotes/origin/* # timeout=10
> git rev-parse refs/remotes/origin/test^{commit} # timeout=10
Checking Out Revision cf8eb13ca41b99a8751ec41b9e1ce7cd3174d64e (refs/remotes/origin/test)
> git config core.sparsecheckout # timeout=10
> git checkout -f cf8eb13ca41b99a8751ec41b9e1ce7cd3174d64e # timeout=10
Commit message: "testingfile"
> git rev-list --no-walk 9c9d57895a44bd6a6481b8a7f63fc9762fc5573d # timeout=10
Finished: SUCCESS

```

Dashboard > job2-prod > #2 > Console Output

**Console Output**

View as plain text

Edit Build Information

Delete build '#2'

Git Build Data

← Previous Build

**Console Output**

```

Running as SYSTEM
Building remotely on prod-node in workspace /home/ubuntu/jenkins/workspace/job2-prod
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
Cloning repository https://github.com/new001001001/assignment2.git
> git init /home/ubuntu/jenkins/workspace/job2-prod # timeout=10
Fetching upstream changes from https://github.com/new001001001/assignment2.git
> git --version # timeout=10
> git --version # 'git version 2.34.1'
> git fetch -t --force --progress -- https://github.com/new001001001/assignment2.git
+refs/heads/*:refs/remotes/origin/* # timeout=10
> git config remote.origin.url https://github.com/new001001001/assignment2.git # timeout=10
> git config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
> git rev-parse refs/remotes/origin/master^{commit} # timeout=10
Checking Out Revision eef6b7e786bfab091435aeaa3441956cc2e3b06f (refs/remotes/origin/master)
> git config core.sparsecheckout # timeout=10
> git checkout -f eef6b7e786bfab091435aeaa3441956cc2e3b06f # timeout=10
Commit message: "prodfile"
> git rev-list --no-walk eef6b7e786bfab091435aeaa3441956cc2e3b06f # timeout=10
Finished: SUCCESS

```

```

ubuntu@ip-172-31-31-196:~$ ls
jenkins
ubuntu@ip-172-31-31-196:~$ cd jenkins/
ubuntu@ip-172-31-31-196:~/jenkins$ ls
remoting remoting.jar workspace
ubuntu@ip-172-31-31-196:~/jenkins$ cd workspace/
ubuntu@ip-172-31-31-196:~/jenkins/workspace$ ls
job1-test
ubuntu@ip-172-31-31-196:~/jenkins/workspace$ cd job1-test
ubuntu@ip-172-31-31-196:~/jenkins/workspace/job1-test$ ls
masterfile testfile testingfile
ubuntu@ip-172-31-31-196:~/jenkins/workspace/job1-test$ 

```

i-0a681404964306096 (test\_server)

PublicIPs: 34.229.150.0 PrivateIPs: 172.31.31.196

```
ubuntu@ip-172-31-23-194:~$ ls
jenkins
ubuntu@ip-172-31-23-194:~$ cd jenkins
ubuntu@ip-172-31-23-194:~/jenkins$ ls
remoting  remoting.jar  workspace
ubuntu@ip-172-31-23-194:~/jenkins$ cd workspace
ubuntu@ip-172-31-23-194:~/jenkins/workspace$ ls
job2-prod
ubuntu@ip-172-31-23-194:~/jenkins/workspace$ cd job2-prod
ubuntu@ip-172-31-23-194:~/jenkins/workspace/job2-prod$ ls
masterfile  proffile
ubuntu@ip-172-31-23-194:~/jenkins/workspace/job2-prod$
```

i-0214a6c68f09e2058 (prod\_server)  
Public IPs: 34.207.60.16 Private IPs: 172.31.23.194

X

 CloudShell   Feedback

© 2024, Amazon Web Services, Inc. or its affiliates.   Privacy   Terms   Cookie preferences

## Assignment – 3

### Task to be performed

1. Created a pipeline in Jenkins
2. Once push is made to “develop” a branch in Git, trigger job “test” This will copy Git files to test node
3. If test job is successful, then prod job should be triggered
4. Prod jobs should copy files to prod node

### Steps and Commands

- Navigate to master instance
- Create a directory by running the command **mkdir (directory name)**
- Get into the directory by running the command **cd (directory name)**
- To Initialize the git by running the command **git init**
- To create the files by running the command **touch (creating file name)**
- To stage files by running the command **git add (created file name)**
- And to check the files which are in status by running the command **git status**
- To push the files to commit stage by running command **git commit -m "necessary notes"**
- Now the master branch will be created and to see the master branch by running the command **git branch**
- To create test branches by running command **git branch develop**
- To check the list of branch by running command **git branch**
- Next to switch from master branch to test branch by running the command **git checkout develop**
- If we run the command **ls** in develop branch the files which was created in master branch will be reflected
- To create the files by running the command **touch (creating file name)**
- To stage files by running the command **git add (created file name)**
- And to check the files which are in status by running the command **git status**
- To push the files to commit stage by running command **git commit -m "necessary notes"**
- Navigate to Github account
- Go Inside the **GITHUB** account
  - g) Click the New in the top repositories
  - h) Provide the name for repository
  - i) Finally click the create repository
- Navigate to master machine

- Next to link the repository with local machine by running the command **git remote add origin (repository URL link)**
- To push the all the branches and files into the repository by running the command **git push --all**
- Provide the username and password token which was assigned by github
- Navigate to github repository which was created
- Now the all the branches and files which was created in master machine will be reflected in Github repository in Github account
- Navigate to Jenkins pages in browser
- To create 2 job click on new item
- Provide the name for the job as test-Job and prod-Job
- Provide the (**repository URL link**) in the Github project
- Provide the test node for test-job and prod node for prod-job which was created in restrict where this project can be run
- Provide the (**repository URL link**) in source code management under git repositories
- Provide develop branch in branches to build
- Navigate to github repository which was created
- To create the webhook by following steps
  - i) Click the settings of created repository
  - j) Go inside the webhooks and click the add webhook
  - k) Provide the Jenkins page URL/github-webhook/
  - l) Finally click on add webhook
- Navigate to Jenkins pages in browser
- In test-Job select the Github hook trigger for GITScm polling in build triggers
- In test-Job provide the prod-Job in post build action under project to build when trigger only if build is stable (It help to trigger prod-Job only if the test-Job is run successful)
- Finally click on apply and save
- Now test-Job and prod-Job have been created
- Next go to manage Jenkins and download build pipeline
- Next click new view and Provide the name for pipeline
- Select build pipeline view in type
- Provide the necessary configuration
- Finally click apply and Ok
- Navigate to master instance
- Next to switch from master branch to develop branch by running the command **git checkout develop**
- To create the files in master branch by running the command **touch newfile**
- To stage files by running the command **git add newfile**

- And to check the files which are in status by running the command **git status**
- To push the files to commit stage by running command **git commit -m "necessary notes"**
- To push the develop branches and files into the repository by running the command **git push origin develop**
- Provide the username and password token which was assigned by github
- Navigate to github repository which was created
- Now the file created in develop branch will be reflected in Github repository in Github account
- Navigate to Jenkins pages in browser
- Now test-job will be triggered and run in the test node once it the test-Job is successful.
- Next prod-Job will be triggered and run in prod node because test-job is successful.
- Now in build pipeline page we can see the test-job and prod-job in pipeline format.

```
ubuntu@ip-172-31-53-47:~/assignment3$ git init
Reinitialized existing Git repository in /home/ubuntu/assignment3/.git/
ubuntu@ip-172-31-53-47:~/assignment3$ touch masterfile
ubuntu@ip-172-31-53-47:~/assignment3$ git add masterfile
ubuntu@ip-172-31-53-47:~/assignment3$ git commit -m "masterfile"
[master (root-commit) fd0bd0] masterfile
Committer: Ubuntu <ubuntu@ip-172-31-53-47.ec2.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

git config --global --edit

After doing this, you may fix the identity used for this commit with:

git commit --amend --reset-author

2 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 master
create mode 100644 masterfile
ubuntu@ip-172-31-53-47:~/assignment3$
```

[CloudShell](#) [Feedback](#) [© 2024, Amazon Web Services, Inc. or its affiliates.](#) [Privacy](#) [Terms](#) [Cookie preferences](#)

```
ubuntu@ip-172-31-53-47:~/assignment3$ git branch
* master
ubuntu@ip-172-31-53-47:~/assignment3$ git branch develop
ubuntu@ip-172-31-53-47:~/assignment3$ git checkout develop
Switched to branch 'develop'
ubuntu@ip-172-31-53-47:~/assignment3$ touch newfile
ubuntu@ip-172-31-53-47:~/assignment3$ git add newfile
ubuntu@ip-172-31-53-47:~/assignment3$ git commit -m "newfile"
[develop 5eba23f] newfile
Committer: Ubuntu <ubuntu@ip-172-31-53-47.ec2.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

git config --global --edit

After doing this, you may fix the identity used for this commit with:

git commit --amend --reset-author

1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 newfile
ubuntu@ip-172-31-53-47:~/assignment3$
```

[CloudShell](#) [Feedback](#) [© 2024, Amazon Web Services, Inc. or its affiliates.](#) [Privacy](#) [Terms](#) [Cookie preferences](#)

 New repository

Type  to search

## Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk (\*).

Owner \*  Repository name \*   
     assignment3 is available.

Great repository names are short and memorable. Need inspiration? How about [redesigned-octo-train](#) ?

Description (optional)

 Anyone on the internet can see this repository. You choose who can commit.  
  You choose who can see and commit to this repository.

Initialize this repository with:

Add a README file  
This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

 You are creating a public repository in your personal account.

 © 2024 GitHub, Inc. [Terms](#) [Privacy](#) [Security](#) [Status](#) [Docs](#) [Contact](#) [Manage cookies](#) [Do not share my personal information](#)

```
ubuntu@ip-172-31-53-47:~/assignment3$ git remote add origin https://github.com/new001001001/assignment3.git
```

```
ubuntu@ip-172-31-53-47:~/assignment3$ git push --all
Username for 'https://github.com': new00100101
Password for 'https://new00100101@github.com':
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 2 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (5/5), 402 bytes | 402.00 KiB/s, done.
Total 5 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/new00100101/assignment3.git
 * [new branch]      develop -> develop
 * [new branch]      master -> master
ubuntu@ip-172-31-53-47:~/assignment3$
```

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

This screenshot shows a GitHub repository page for a project named 'assignment3'. The repository is public and has two branches: 'master' and 'develop'. The 'master' branch has two commits from 'Ubuntu' and one commit from 'master'. There is also a 'masterfile' file. The 'About' section indicates no description, website, or topics are provided. The 'Activity' section shows 1 commit, 0 stars, 1 watching, and 0 forks. The 'Releases' section shows no releases published, with a link to 'Create a new release'. The 'Packages' section is empty.

This screenshot shows the same GitHub repository page for 'assignment3', but with a modal dialog open over the main content. The dialog is titled 'Switch branches/tags' and contains a search bar with the placeholder 'Find or create a branch...'. Below the search bar are tabs for 'Branches' and 'Tags', with 'Branches' currently selected. It lists two branches: 'develop' and 'master'. A 'View all branches' link is at the bottom of the list. The main repository page content is visible in the background, showing the same details as the previous screenshot.

Dashboard > All >

### Enter an item name

job1-test » Required field

 **Freestyle project**  
Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.

 **Pipeline**  
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

 **Configuration project**  
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific

Dashboard > job1-test > Configuration

## Configure

### General

Enabled

**Description**  
push is made to develop branch test job will be triggered

Plain text [Preview](#)

Discard old builds ?

GitHub project

[Save](#) [Apply](#)

Dashboard > job1-test > Configuration

## Configure

### General

GitHub project

**Project url** ?

Advanced ▾

This project is parameterized ?

Throttle builds ?

Execute concurrent builds if necessary ?

Restrict where this project can be run ?

Advanced ▾

[Save](#) [Apply](#)

Dashboard > job1-test > Configuration

## Configure

- General
- Source Code Management**
- Build Triggers
- Build Environment
- Build Steps
- Post-build Actions

- none -

+ Add ▾

Advanced ▾

Add Repository

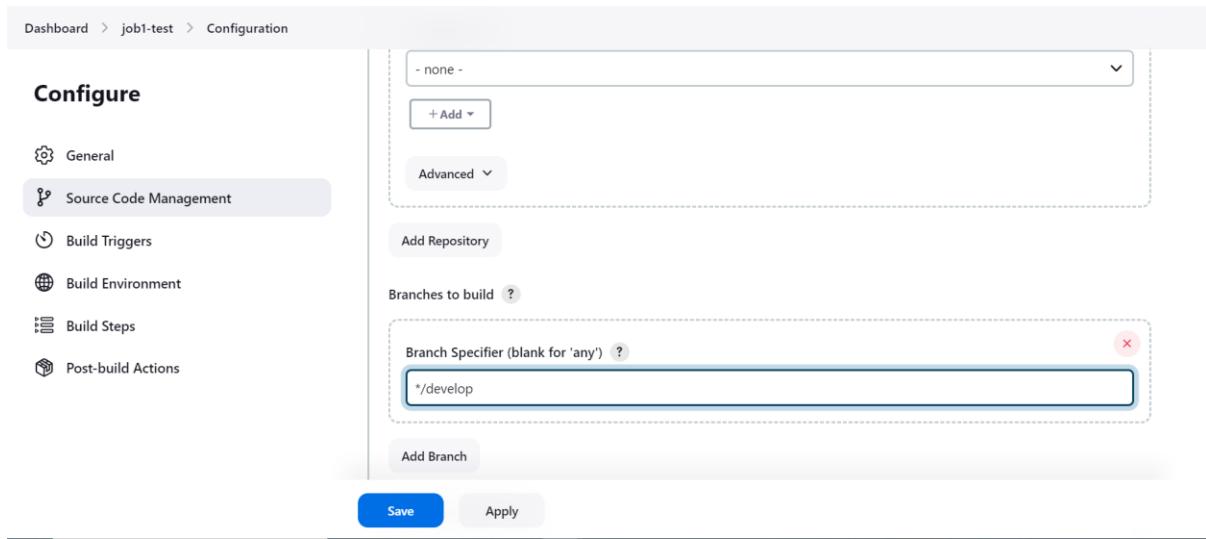
Branches to build ?

Branch Specifier (blank for 'any') ?

\*/develop

Add Branch

Save Apply



Dashboard > job1-test > Configuration

## Configure

- General
- Source Code Management
- Build Triggers
- Build Environment
- Build Steps
- Post-build Actions

Advanced ▾

This project is parameterized ?

Throttle builds ?

Execute concurrent builds if necessary ?

Restrict where this project can be run ?

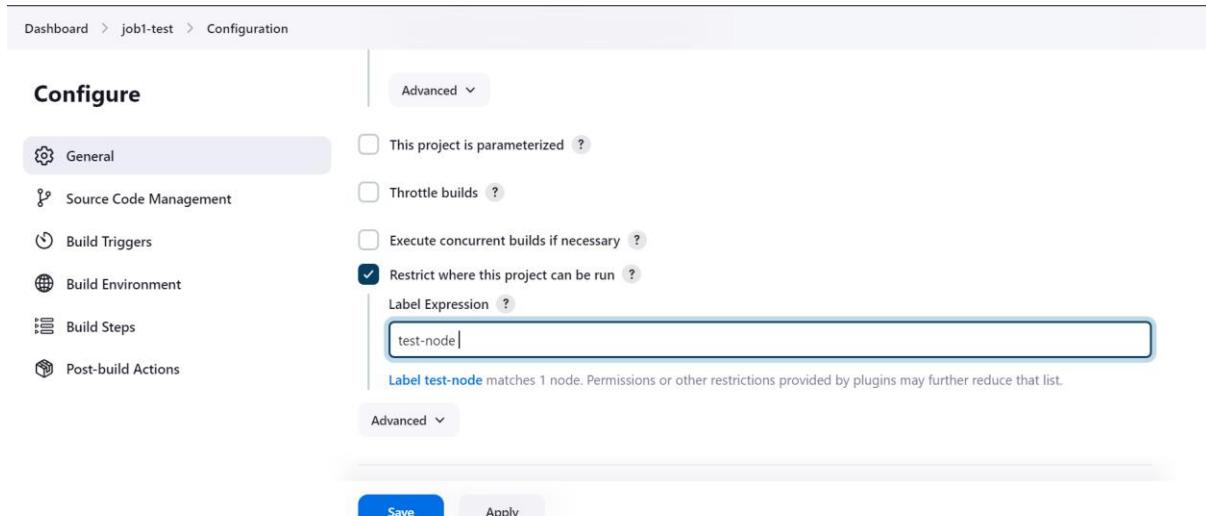
Label Expression ?

test-node

Label test-node matches 1 node. Permissions or other restrictions provided by plugins may further reduce that list.

Advanced ▾

Save Apply



new001001001 / assignment3

Type ⌘ to search

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

General

Access

Collaborators

Moderation options

Code and automation

Branches

Tags

Rules

Actions

**Webhooks**

Environments

Codespaces

Webhooks / Add webhook

We'll send a post request to the URL below with details of any subscribed events. You can also specify which data format you'd like to receive (JSON, x-www-form-urlencoded, etc). More information can be found in our developer documentation.

Payload URL \*

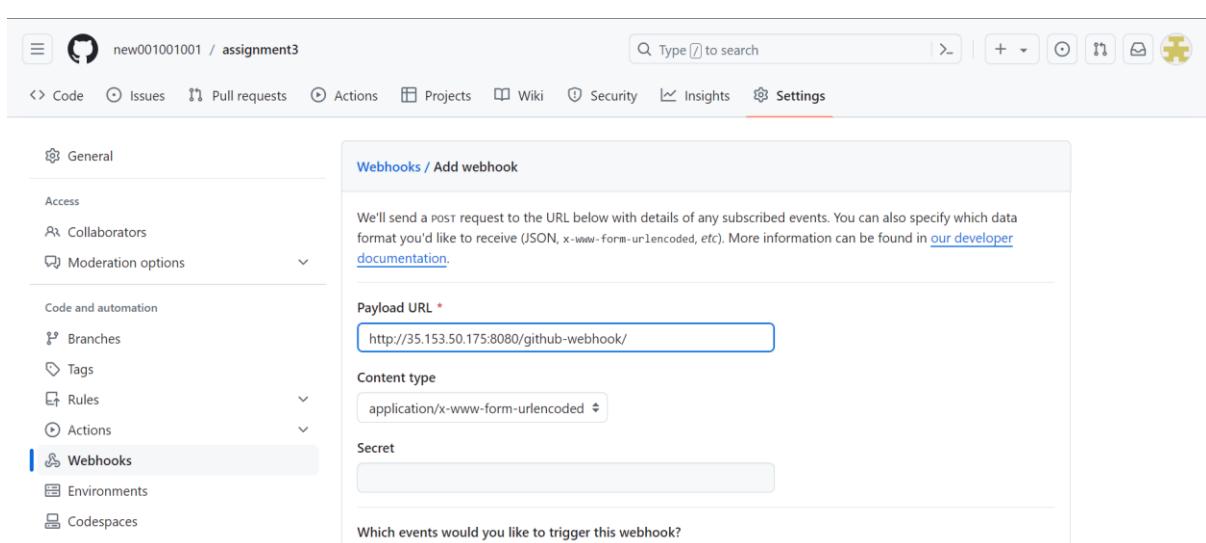
http://35.153.50.175:8080/github-webhook/

Content type

application/x-www-form-urlencoded

Secret

Which events would you like to trigger this webhook?



The screenshot shows the GitHub 'Webhooks' configuration page. On the left, a sidebar lists various GitHub features: Actions, Webhooks (selected), Environments, Codespaces, Pages, Security, Code security and analysis, Deploy keys, Secrets and variables (selected), Integrations, GitHub Apps, and Email notifications. The main area is titled 'application/x-www-form-urlencoded'. It contains a 'Secret' input field, a section for selecting events ('Just the push event' is selected), and a checkbox for 'Active' status. A green 'Add webhook' button is at the bottom.



© 2024 GitHub, Inc. Terms Privacy Security Status Docs Contact Manage cookies Do not share my personal information

The screenshot shows the Jenkins 'Configure' screen for a job named 'job1-test'. The left sidebar includes General, Source Code Management, Build Triggers (selected), Build Environment, Build Steps, and Post-build Actions. The main area shows 'Projects to build' set to 'job2-prod.' and 'Trigger only if build is stable' selected. Below this is an 'Add post-build action' dropdown. At the bottom are 'Save' and 'Apply' buttons.

[REST API](#) Jenkins 2.440.3

The screenshot shows the Jenkins 'Enter an item name' screen with 'job2-prod' entered. Below it are three project type options: 'Freestyle project' (selected), 'Pipeline', and 'or multi-configuration project'. Each option has a brief description and a small icon.

**Configure****General**Enabled  General Source Code Management Build Triggers Build Environment Build Steps Post-build Actions

## Description

If test job is successful then job2-prod job should be triggered

Plain text [Preview](#) Discard old builds [?](#) GitHub project**Save****Apply****Configure** General Source Code Management Build Triggers Build Environment Build Steps Post-build Actions

If test job is successful then job2-prod job should be triggered

Plain text [Preview](#) Discard old builds [?](#) GitHub projectProject url [?](#)

https://github.com/new001001001/assignment3.git

Advanced [▼](#) This project is parameterized [?](#)**Save****Apply****Configure** None Git [?](#)Repositories [?](#)Repository URL [?](#)

https://github.com/new001001001/assignment3.git

X

! Please enter Git repository.

Credentials [?](#)

- none -

+ Add [▼](#)**Save****Apply**

Dashboard > job2-prod > Configuration

## Configure

General

Source Code Management

Build Triggers

Build Environment

Build Steps

Post-build Actions

Execute concurrent builds if necessary ?

Restrict where this project can be run ?

Label Expression ?

prod-node

Label prod-node matches 1 node. Permissions or other restrictions provided by plugins may further reduce that list.

Advanced ▾

### Source Code Management

None

Git ?

Repositories ?

**Save** **Apply**

Dashboard > job2-prod > Configuration

## Configure

General

Source Code Management

Build Triggers

Build Environment

Build Steps

Post-build Actions

With Ant ?

### Build Steps

Add build step ▾

### Post-build Actions

Add post-build action ▾

**Save** **Apply**

REST API Jenkins 2.440.3

Dashboard > job1-test >

## job1-test

Status

</> Changes push is made to develop branch test job will be triggered

Workspace

▷ Build Now

Configure

Delete Project

GitHub Hook Log

GitHub

Rename

Edit description

Disable Project

### Downstream Projects

job2-prod

### Permalinks

Dashboard > job2-prod >

**job2-prod**

Status Changes If test job is successful then job2-prod job should be triggered

Workspace Build Now

Configure Delete Project GitHub Rename

**Upstream Projects**

job1-test

**Permalinks**

Build History trend

Jenkins

Search (CTRL+K) ? 🔔 1 ⚡ 2 jenkinsss log out

Dashboard > Manage Jenkins > Plugins

**Plugins**

Updates Available plugins Installed plugins Advanced settings Download progress

**Download progress**

Preparation

- Checking internet connectivity
- Checking update center connectivity
- Success

Pipeline timeline

- Success

→ Go back to the top page  
(you can start using the installed plugins right away)

Jenkins

Search (CTRL+K) ? 🔔 1 ⚡ 2 jenkinsss log out

Dashboard > New view

+ New Item

People Build History Project Relationship Check File Fingerprint Manage Jenkins My Views

**New view**

Name assignment

Type

**Build Pipeline View**  
Shows the jobs in a build pipeline view. The complete pipeline of jobs that a version propagates through are shown as a row in the view.

**List View**  
Shows items in a simple list format. You can choose which jobs are to be displayed in which view.

Build Queue ▾  
No builds in the queue.

Create

Dashboard > assignment > Configure

Edit View

Delete View

Project Relationship

Check File Fingerprint

Manage Jenkins

My Views

Build Queue

No builds in the queue.

Build Executor Status

Built-In Node

1 Idle

Plain text [Preview](#)

Filter build queue ?

Filter build executors ?

Build Pipeline View Title

Pipeline Flow

Layout

Dashboard > assignment > Configure

Do not show any column headers

Refresh frequency (in seconds)

URL for custom CSS files

Console Output Link Style

Lightbox

REST API Jenkins 2.440.3

```
ubuntu@ip-172-31-53-47:~/assignment3$ git branch
* master
ubuntu@ip-172-31-53-47:~/assignment3$ git checkout develop
Switched to branch 'develop'
ubuntu@ip-172-31-53-47:~/assignment3$ touch prodfile
ubuntu@ip-172-31-53-47:~/assignment3$ git add prodfile
ubuntu@ip-172-31-53-47:~/assignment3$ git commit -m "prodfile"
[develop 18d7739] prodfile
Committer: Ubuntu <ubuntu@ip-172-31-53-47.ec2.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

git config --global --edit

After doing this, you may fix the identity used for this commit with:

git commit --amend --reset-author

1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 prodfile
ubuntu@ip-172-31-53-47:~/assignment3$
```

```

ubuntu@ip-172-31-53-47:~/assignment3$ git push origin develop
Username for 'https://github.com': new001001001
Password for 'https://new001001001@github.com':
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 2 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 240 bytes | 240.00 KiB/s, done.
Total 2 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/new001001001/assignment3.git
  5eba23f..18d7739  develop -> develop
ubuntu@ip-172-31-53-47:~/assignment3$ 

```

CloudShell   Feedback   © 2024, Amazon Web Services, Inc. or its affiliates.   Privacy   Terms   Cookie preferences

[new001001001 / assignment3](#)

Type  to search

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

[assignment3](#) (Public)

develop ▾ 2 Branches 0 Tags

Go to file Add file Code

**About**

No description, website, or topics provided.

Activity 0 stars 1 watching 0 forks

**Releases**

No releases published [Create a new release](#)

**Packages**

Dashboard > job2-prod >

</> Changes If test job is successful then job2-prod job should be triggered

Workspace Build Now Configure Delete Project GitHub Rename

[Edit description](#) [Disable Project](#)

**Upstream Projects**

[job1-test](#)

**Permalinks**

- Last build (#1), 2 min 37 sec ago
- Last stable build (#1), 2 min 37 sec ago
- Last successful build (#1), 2 min 37 sec ago
- Last completed build (#1), 2 min 37 sec ago

**Build History** trend

Filter... #1 Apr 27, 2024, 7:39 PM

Dashboard > job1-test >

[Build Now](#) [Configure](#) [Delete Project](#) [GitHub Hook Log](#) [GitHub](#) [Rename](#) [Disable Project](#)

### Downstream Projects

[job2-prod](#)

### Permalinks

- Last build (#1), 32 sec ago
- Last stable build (#1), 32 sec ago
- Last successful build (#1), 32 sec ago
- Last completed build (#1), 32 sec ago

Build History trend

Filter... /

#1 | Apr 27, 2024, 7:39 PM

[Atom feed for all](#) [Atom feed for failures](#)

Dashboard > job1-test > #1 > Console Output

</> Changes

[Console Output](#) (selected) [View as plain text](#) [Edit Build Information](#) [Delete build '#1'](#) [Polling Log](#) [Git Build Data](#)

```

Started by GitHub push by new001001001
Running as SYSTEM
Building remotely on test-node in workspace /home/ubuntu/jenkins/workspace/job1-test
The recommended git tool is: NONE
No credentials specified
> git rev-parse --resolve-git-dir /home/ubuntu/jenkins/workspace/job1-test/.git # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://github.com/new001001001/assignment3.git # timeout=10
Fetching upstream changes from https://github.com/new001001001/assignment3.git
> git --version # timeout=10
> git --version # 'git version 2.34.1'
> git fetch --tags --force --progress -- https://github.com/new001001001/assignment3.git
+refs/heads/*:refs/remotes/origin/* # timeout=10
> git rev-parse refs/remotes/origin/develop^{commit} # timeout=10
Checking out Revision 18d7739ca349d9be6b3f641c48be827ebc268d79 (refs/remotes/origin/develop)
> git config core.sparsecheckout # timeout=10
> git checkout -f 18d7739ca349d9be6b3f641c48be827ebc268d79 # timeout=10
Commit message: "prodfile"
First time build. Skipping changelog.
Triggering a new build of job2-prod
Finished: SUCCESS

```

Dashboard > job2-prod > #5 > Console Output

</> Changes

[Console Output](#) (selected) [View as plain text](#) [Edit Build Information](#) [Delete build '#5'](#) [Git Build Data](#) [Aggregated Test Result](#) [← Previous Build](#)

```

Started by upstream project "job1-test" build number 4
originally caused by:
Started by user Jenkins
Running as SYSTEM
Building remotely on prod-node in workspace /home/ubuntu/jenkins/workspace/job2-prod
The recommended git tool is: NONE
No credentials specified
> git rev-parse --resolve-git-dir /home/ubuntu/jenkins/workspace/job2-prod/.git # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://github.com/new001001001/assignment3.git # timeout=10
Fetching upstream changes from https://github.com/new001001001/assignment3.git
> git --version # timeout=10
> git --version # 'git version 2.34.1'
> git fetch --tags --force --progress -- https://github.com/new001001001/assignment3.git
+refs/heads/*:refs/remotes/origin/* # timeout=10
> git rev-parse refs/remotes/origin/develop^{commit} # timeout=10
Checking out Revision 18d7739ca349d9be6b3f641c48be827ebc268d79 (refs/remotes/origin/develop)
> git config core.sparsecheckout # timeout=10
> git checkout -f 18d7739ca349d9be6b3f641c48be827ebc268d79 # timeout=10
Commit message: "prodfile"
First time build. Skipping changelog.
Finished: SUCCESS

```

Jenkins

Search (CTRL+K)

Dashboard > assignment >

## Build Pipeline: pipeline

Run History Configure Add Step Delete Manage

Pipeline #1

#1 job1-test  
Apr 27, 2024 7:39:36 PM  
0.44 sec

#1 job2-prod  
Apr 27, 2024 7:39:46 PM  
0.3 sec

The Jenkins interface shows a build pipeline named 'pipeline'. It contains two jobs: 'job1-test' and 'job2-prod'. Both jobs were run on April 27, 2024, at different times. Job 1 took 0.44 seconds, and Job 2 took 0.3 seconds.

REST API Jenkins 2.440.3