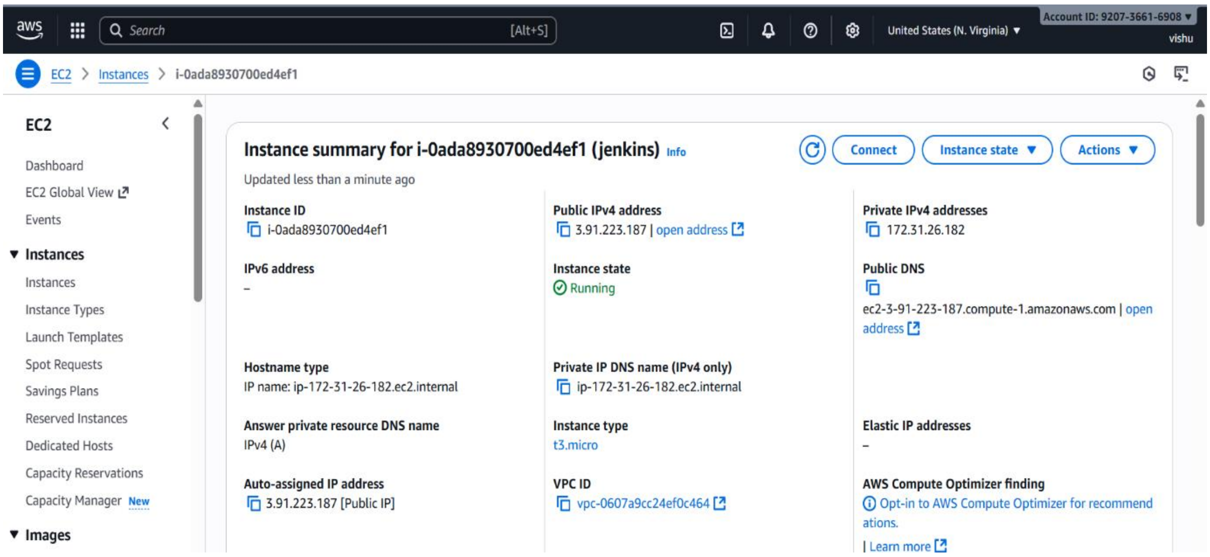
This project demonstrates how to set up a **Jenkins CI/CD pipeline** to automate the test, build and deployment of a simple Python hello-flask-world (e.g., Flask)

Install Jenkins on a Virtual AWS Machine



Instance summary for i-0ada8930700ed4ef1 (jenkins)

Updated less than a minute ago

Instance ID: i-0ada8930700ed4ef1

Public IPv4 address: 3.91.223.187 | open address

Private IPv4 addresses: 172.31.26.182

IPv6 address: –

Instance state: Running

Public DNS: ec2-3-91-223-187.compute-1.amazonaws.com | open address

Hostname type: IP name: ip-172-31-26-182.ec2.internal

Private IP DNS name (IPv4 only): ip-172-31-26-182.ec2.internal

Answer private resource DNS name: IPv4 (A)

Instance type: t3.micro

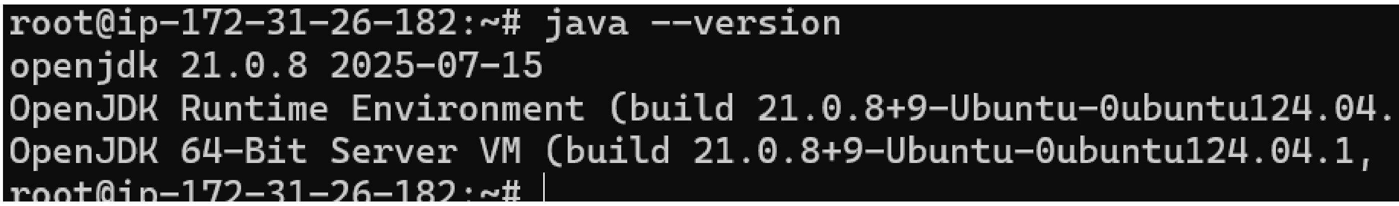
Elastic IP addresses: –

Auto-assigned IP address: 3.91.223.187 [Public IP]

VPC ID: vpc-0607a9cc24ef0c464

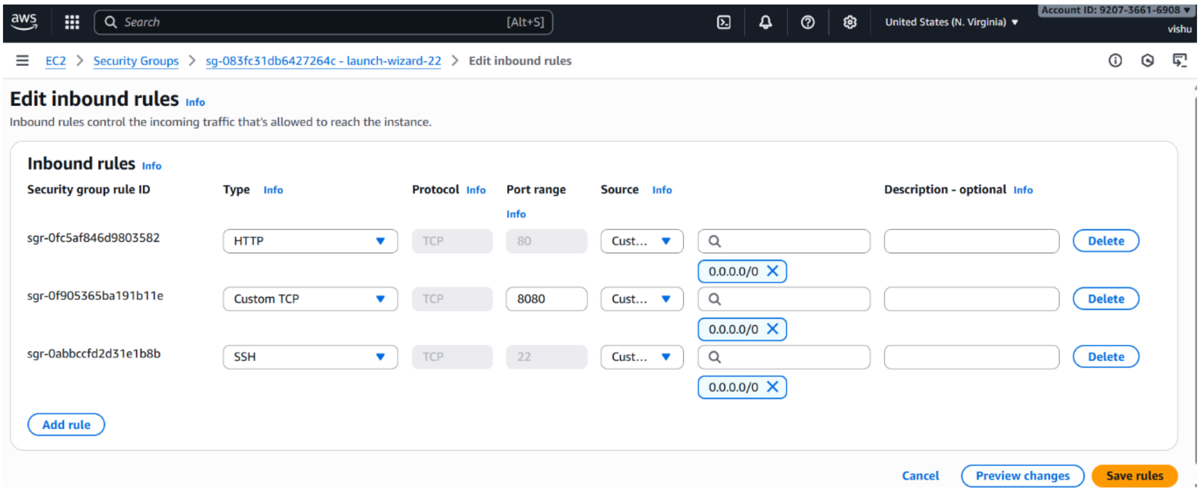
AWS Compute Optimizer finding: Opt-in to AWS Compute Optimizer for recommendations. | Learn more

1. Launch an Ubuntu virtual machine (t3.micro).



```
root@ip-172-31-26-182:~# java --version
openjdk 21.0.8 2025-07-15
OpenJDK Runtime Environment (build 21.0.8+9-Ubuntu-0ubuntu124.04.
OpenJDK 64-Bit Server VM (build 21.0.8+9-Ubuntu-0ubuntu124.04.1,
root@ip-172-31-26-182:~#
```

2. Install java version : openjdk version "21.0.3" 2024-04-16



aws
Search [Alt+S]
United States (N. Virginia) ▼
Account ID: 9207-3661-6908 ▼
vishu
EC2 > Security Groups > sg-083fc31db6427264c - launch-wizard-22 > Edit inbound rules

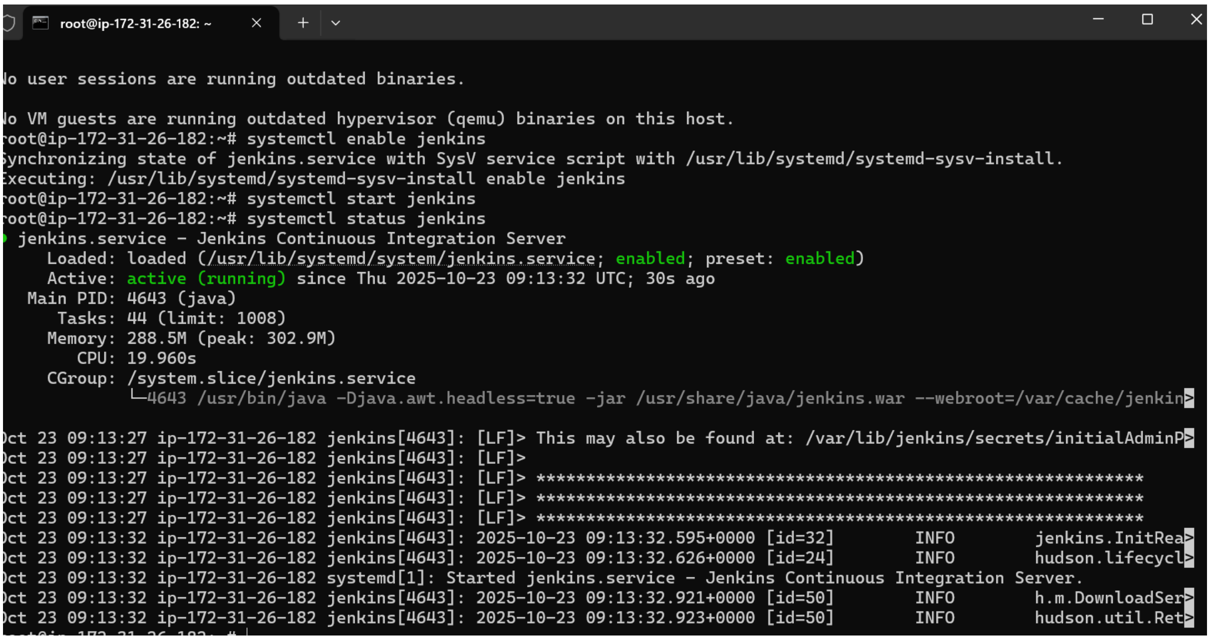
Edit inbound rules Info

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules Info

Security group rule ID	Type Info	Protocol Info	Port range Info	Source Info		Description - optional Info	
sgr-0fc5af846d9803582	HTTP ▼	TCP	80	Cust… ▼	0.0.0.0/0 ✕		Delete
sgr-0f905365ba191b11e	Custom TCP ▼	TCP	8080	Cust… ▼	0.0.0.0/0 ✕		Delete
sgr-0abbccfd2d31e1b8b	SSH ▼	TCP	22	Cust… ▼	0.0.0.0/0 ✕		Delete

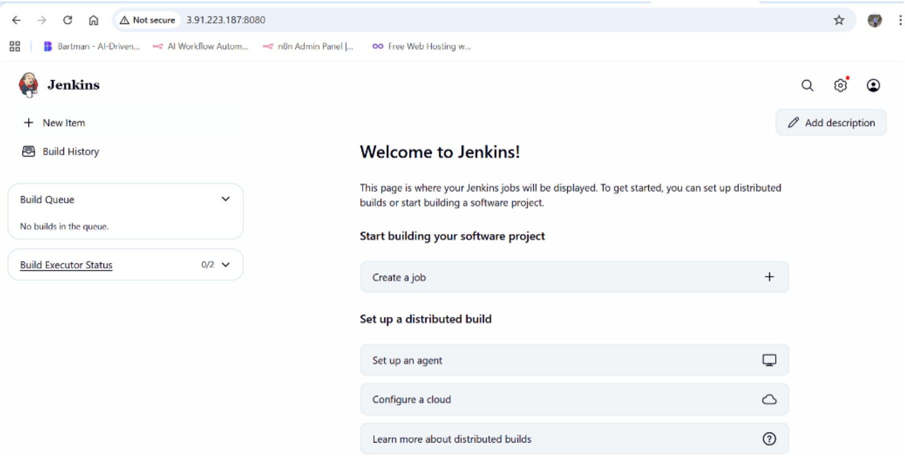
3. Create security groups



```
lo user sessions are running outdated binaries.

lo VM guests are running outdated hypervisor (qemu) binaries on this host.
oot@ip-172-31-26-182:~# systemctl enable jenkins
ynchronizing state of jenkins.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
xecuting: /usr/lib/systemd/systemd-sysv-install enable jenkins
oot@ip-172-31-26-182:~# systemctl start jenkins
oot@ip-172-31-26-182:~# systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
     Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; preset: enabled)
     Active: active (running) since Thu 2025-10-23 09:13:32 UTC; 30s ago
   Main PID: 4643 (java)
      Tasks: 44 (limit: 1008)
     Memory: 288.5M (peak: 302.9M)
        CPU: 19.960s
     CGroup: /system.slice/jenkins.service
             └─4643 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenkin>

Oct 23 09:13:27 ip-172-31-26-182 jenkins[4643]: [LF]> This may also be found at: /var/lib/jenkins/secrets/initialAdminP>
Oct 23 09:13:27 ip-172-31-26-182 jenkins[4643]: [LF]>
Oct 23 09:13:27 ip-172-31-26-182 jenkins[4643]: [LF]> *************************************************************
Oct 23 09:13:27 ip-172-31-26-182 jenkins[4643]: [LF]> *************************************************************
Oct 23 09:13:27 ip-172-31-26-182 jenkins[4643]: [LF]> *************************************************************
Oct 23 09:13:32 ip-172-31-26-182 jenkins[4643]: 2025-10-23 09:13:32.595+0000 [id=32]        INFO        jenkins.InitRea>
Oct 23 09:13:32 ip-172-31-26-182 jenkins[4643]: 2025-10-23 09:13:32.626+0000 [id=24]        INFO        hudson.lifecycl>
Oct 23 09:13:32 ip-172-31-26-182 systemd[1]: Started jenkins.service - Jenkins Continuous Integration Server.
Oct 23 09:13:32 ip-172-31-26-182 jenkins[4643]: 2025-10-23 09:13:32.921+0000 [id=50]        INFO        h.m.DownloadSer>
Oct 23 09:13:32 ip-172-31-26-182 jenkins[4643]: 2025-10-23 09:13:32.923+0000 [id=50]        INFO        hudson.util.Ret>
```

4. Install enkins:

sudo systemctl start Jenkins



No builds in the queue.

Welcome to Jenkins!

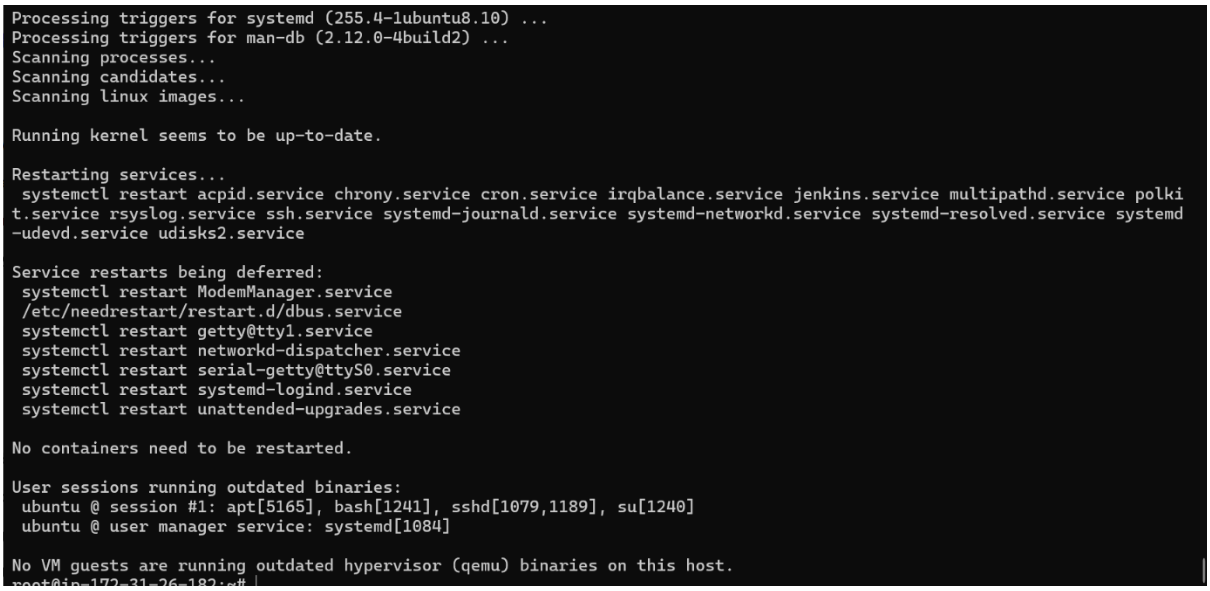
This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.

Start building your software project

Set up a distributed build

Access Jenkins at:

👉 http://<3.91.223.187>:8080



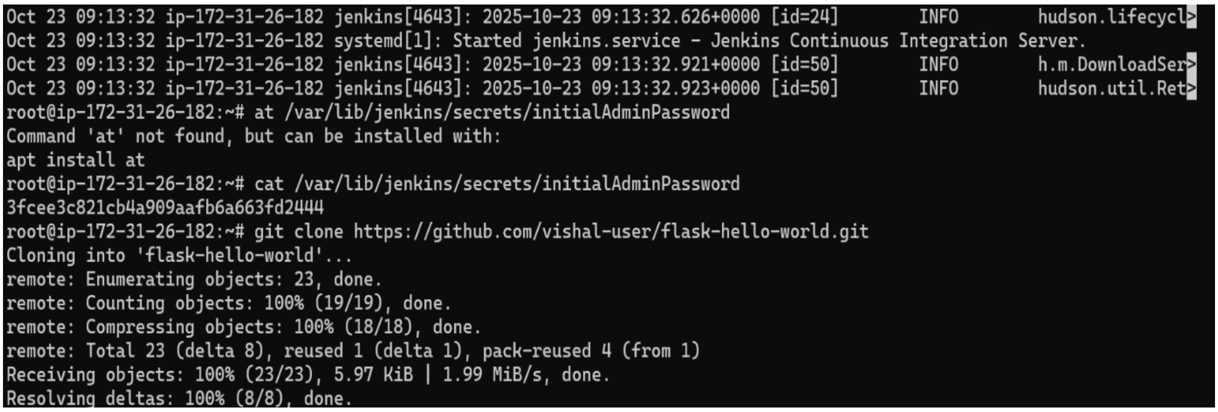
```
Processing triggers for systemd (255.4-1ubuntu8.10) ...
Processing triggers for man-db (2.12.0-4build2) ...
Scanning processes...
Scanning candidates...
Scanning linux images...

Running kernel seems to be up-to-date.

Restarting services...
 systemctl restart acpid.service chrony.service cron.service irqbalance.service jenkins.service multipathd.service polki
t.service rsyslog.service ssh.service systemd-journald.service systemd-networkd.service systemd-resolved.service systemd
-udevd.service udisks2.service

Service restarts being deferred:
 systemctl restart ModemManager.service
 /etc/needrestart/restart.d/dbus.service
 systemctl restart getty@tty1.service
 systemctl restart networkd-dispatcher.service
 systemctl restart serial-getty@ttyS0.service
 systemctl restart systemd-logind.service
 systemctl restart unattended-upgrades.service

No containers need to be restarted.

User sessions running outdated binaries:
 ubuntu @ session #1: apt[5165], bash[1241], sshd[1079,1189], su[1240]
 ubuntu @ user manager service: systemd[1084]

No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-172-31-26-182:~#
```

Configure Python

Install Python and necessary libraries on your Jenkins node:

bash

Copy code

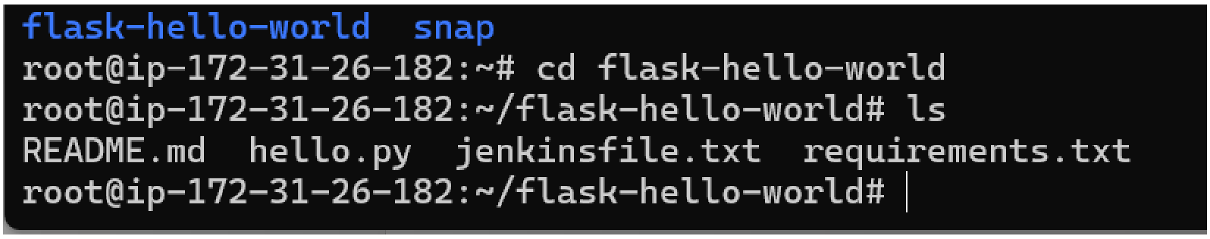


```
Oct 23 09:13:32 ip-172-31-26-182 jenkins[4643]: 2025-10-23 09:13:32.626+0000 [id=24]        INFO        hudson.lifecycl>
Oct 23 09:13:32 ip-172-31-26-182 systemd[1]: Started jenkins.service - Jenkins Continuous Integration Server.
Oct 23 09:13:32 ip-172-31-26-182 jenkins[4643]: 2025-10-23 09:13:32.921+0000 [id=50]        INFO        h.m.DownloadSer>
Oct 23 09:13:32 ip-172-31-26-182 jenkins[4643]: 2025-10-23 09:13:32.923+0000 [id=50]        INFO        hudson.util.Ret>
root@ip-172-31-26-182:~# at /var/lib/jenkins/secrets/initialAdminPassword
Command 'at' not found, but can be installed with:
apt install at
root@ip-172-31-26-182:~# cat /var/lib/jenkins/secrets/initialAdminPassword
3fcee3c821cb4a909aafb6a663fd2444
root@ip-172-31-26-182:~# git clone https://github.com/vishal-user/flask-hello-world.git
Cloning into 'flask-hello-world'...
remote: Enumerating objects: 23, done.
remote: Counting objects: 100% (19/19), done.
remote: Compressing objects: 100% (18/18), done.
remote: Total 23 (delta 8), reused 1 (delta 1), pack-reused 4 (from 1)
Receiving objects: 100% (23/23), 5.97 KiB | 1.99 MiB/s, done.
Resolving deltas: 100% (8/8), done.
```

Fork the sample Python Flask repository:

git clone https://github.com/vishal-user/flask-hello-world.git

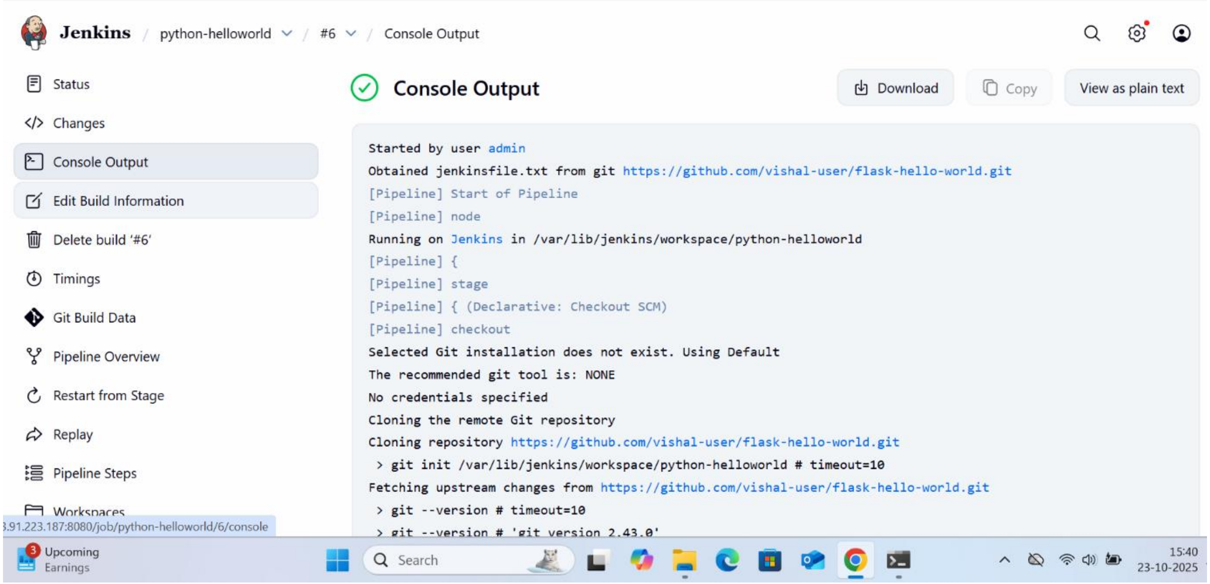
cd flask-hello-world



```
flask-hello-world  snap
root@ip-172-31-26-182:~# cd flask-hello-world
root@ip-172-31-26-182:~/flask-hello-world# ls
README.md  hello.py  jenkinsfile.txt  requirements.txt
root@ip-172-31-26-182:~/flask-hello-world#
```

⚙️ 4. Jenkins Pipeline Configuration

Create a file named Jenkinsfile in the root of your repository with the following content:



Console Output

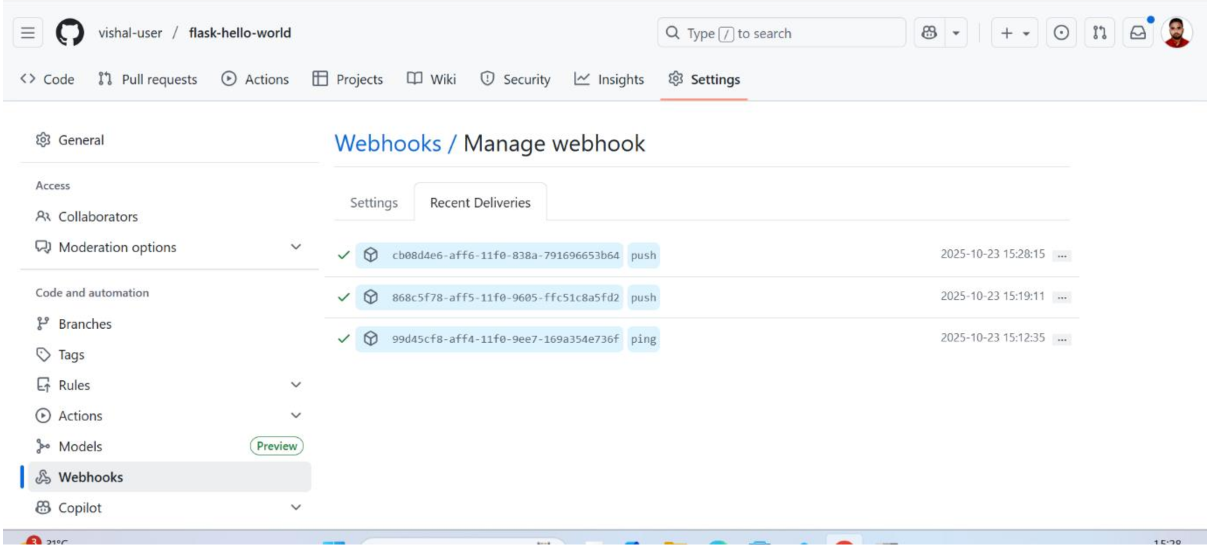
```
Started by user admin
Obtained jenkinsfile.txt from git https://github.com/vishal-user/flask-hello-world.git
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/python-helloworld
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Declarative: Checkout SCM)
[Pipeline] checkout
Selected Git installation does not exist. Using Default
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
Cloning repository https://github.com/vishal-user/flask-hello-world.git
 > git init /var/lib/jenkins/workspace/python-helloworld # timeout=10
Fetching upstream changes from https://github.com/vishal-user/flask-hello-world.git
 > git --version # timeout=10
 > git --version # 'git version 2.43.0'
```

🔁 5. Triggers (Automatic Build on Code Push)

Enable automatic builds on new commits to the master branch.

In Jenkins, open pipeline job → Configure → Build Triggers.

Select GitHub hook trigger for GITScm polling.



Webhooks / Manage webhook

✓ cb08d4e6-aff6-11f0-838a-791696653b64 push 2025-10-23 15:28:15

✓ 868c5f78-aff5-11f0-9605-ffc51c8a5fd2 push 2025-10-23 15:19:11

✓ 99d45cf8-aff4-11f0-9ee7-169a354e736f ping 2025-10-23 15:12:35

In GitHub repo → Settings → Webhooks:

Payload URL: http://<jenkins-server-ip>:8080/github-webhook/

Content type: application/json

Events: “Just the push event”

Save.

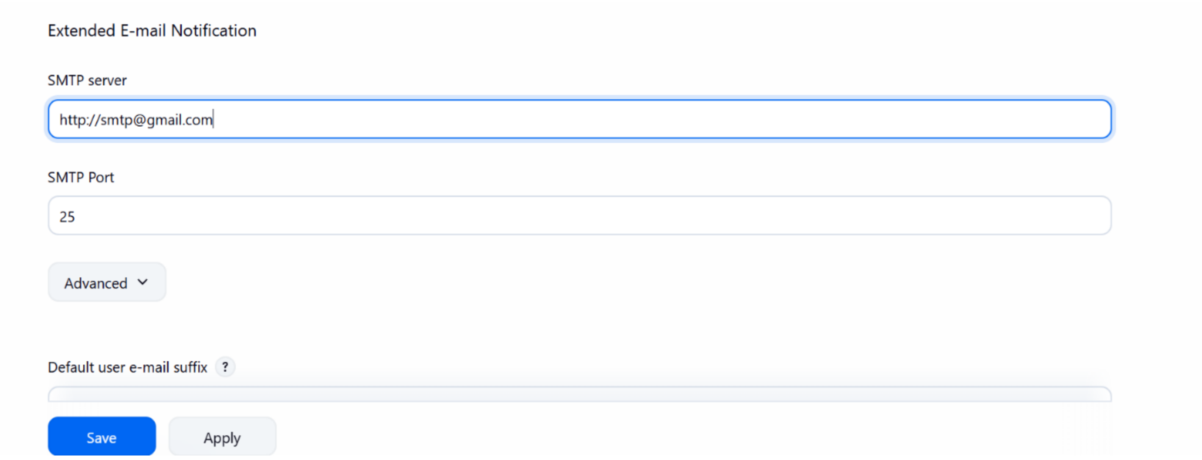
Now Jenkins will automatically trigger builds on every new push to the main branch.

📨 6. Notifications (Email Alerts)

Set up email notifications for build success or failure.

SMTP Configuration:

Go to Manage Jenkins → Configure System → E-mail Notification.



Extended E-mail Notification

SMTP server

http://smtp@gmail.com

SMTP Port

25

Enter SMTP settings (vddogra96@gmail.com):

SMTP Server: smtp.gmail.com

Port: 587

Credentials: Add Gmail App Password via Jenkins credentials.

Test email sending to ensure setup works.

Email-ext Configuration:

Install the Email Extension Plugin.

Add recipients in your pipeline (emailext to: line).

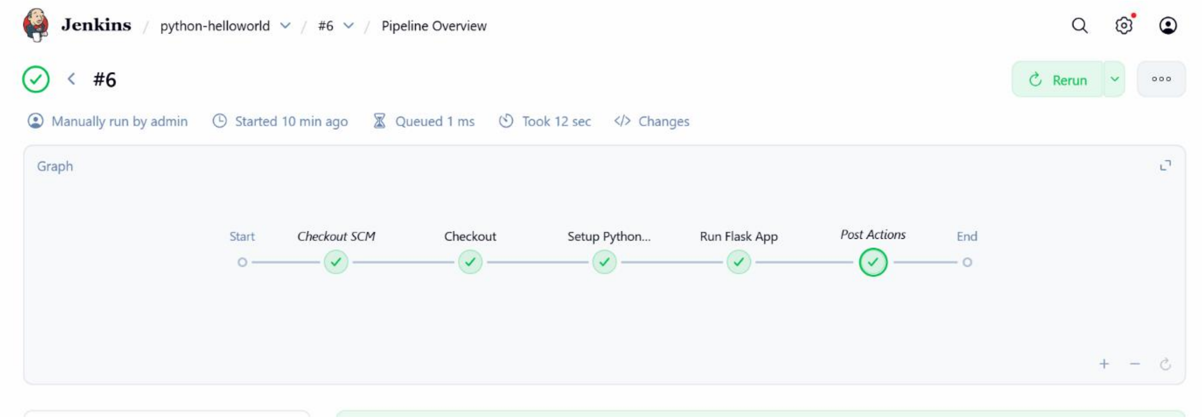
Jenkins will send emails after each build with results and console logs.

□ 7. Run the Pipeline

Commit and push the Jenkinsfile to your repository.

✅ 8. Example Output

stages in Jenkins like this:



#6

Manually run by admin · Started 10 min ago · Queued 1 ms · Took 12 sec · Changes

csharp

Copy code

```
[Pipeline] Start of Pipeline

[Stage] Checkout

[Stage] Build

[Stage] Test

[Stage] Deploy

[Pipeline] End of Pipeline

Finished: SUCCESS
```