

# **Doc use panavanga intha num ku gpay(only above Rs200): 8056099616**

## **DOCKER**

Installing Docker on Ubuntu - go to official website (Docker docs)

```
mkdir fapp2  
touch app.py
```

### **app.py**

```
from flask import Flask  
app=Flask(__name__)  
@app.route('/')  
def run():  
    return "vicky punda"  
app.run('0.0.0.0',5000)
```

```
touch Dockerfile
```

### **Dockerfile**

```
FROM python:3.10  
WORKDIR /app  
COPY . /app  
RUN pip install -r requirements.txt
```

EXPOSE 5000

CMD ["python3","app.py"]

touch requirements.txt

**requirements.txt**

Flask==2.0.1

Werkzeug==2.0.1

docker build -t fapp2 .

docker run -p 5000:5000 fapp2

**SSH**

ssh username@hostname\_or\_ip\_address

**If key pair not generated already**

ssh-keygen

ssh-copy-id username@hostname\_or\_ip\_address

ssh username@hostname\_or\_ip\_address

# JENKINS

```
sudo apt-get install jenkins
```

```
sudo apt-get install fontconfig openjdk-17-jre (if Java  
doesn't exist)
```

```
sudo systemctl start jenkins
```

```
sudo systemctl status jenkins
```

Go to <https://localhost:8080> - default Jenkins server

It will ask for the authentication key. Copy the path and use command

```
sudo cat "paste_the_path_here"
```

Paste the key from terminal to the Jenkins server

Create an account with username and password

---

Create a repository with a Java file

For eg: <https://github.com/ssnlabs/jenkins.git>

Jenkins

1. Click New Item

2. Give a name and select Freestyle Project
3. Select Github project and paste repository name

## General

Enabled 


Description

Plain text [Preview](#)

☐ Discard old builds [?](#)

☒ GitHub project


Project url [?](#)

Advanced 

☐ This project is parameterized [?](#)

☐ Throttle builds [?](#)

☐ Execute concurrent builds if necessary [?](#)

Advanced 

---

## Source Code Management

## 4. Select Git, Under Credentials, Click Add & Jenkins Change “Branch Specifier” to \*/main

Source Code Management

☐ None

☒ Git ?

Repositories ?

Repository URL ?

https://github.com/ssnlabs/jenkins.git

Credentials ?

- none -

Jenkins Credentials Provider

Jenkins

Advanced

Add Repository

Branches to build ?

Branch Specifier (blank for 'any') ?

\*/main

Add Branch

## 5. Enter your github username and password

Jenkins Credentials Provider: Jenkins

Kind

Username with password

Scope ?

Global (Jenkins, nodes, items, all child items, etc)

Username ?

ssnlabs

☐ Treat username as secret ?

Password ?

\*\*\*\*\*

## 6. Add \* \* \* \* \* under Build Periodically -> Schedule (Cron Job - triggers a build every minute)

### Build Triggers

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

☐ Build after other projects are built ?

☒ Build periodically ?

Schedule ?

\* \* \* \* \*

⚠ Do you really mean "every minute" when you say "\* \* \* \* \*"? Perhaps you meant "H \* \* \* \*" to poll once per hour  
Would last have run at Thursday, 7 November, 2024, 10:03:55 pm India Standard Time; would next run at Thursday, 7 November, 2024, 10:03:55 pm India Standard Time.

☐ GitHub hook trigger for GITScm polling ?

☐ Poll SCM ?

## 7. Under Build Step -> Select Execute Shell and enter the following

### Build Steps

≡ Execute shell ?

Command

See [the list of available environment variables](#)

```
javac hello.java
java hello
```

Advanced ▾

## 8. Give Save and Build Now in next page

## 9. Enjoy!