Jenkins Fundamentals

Jenkins Fundamentals:

Objectives:

- Understand what Jenkins is and how it works
- Set up Jenkins on an EC2 instance
- Run basic Freestyle and Pipeline jobs

1. What is Jenkins?

Jenkins is an **open-source automation server** used to:

- Build, test, and deploy applications
- Automate any part of your software delivery process (CI/CD)
- Integrate with hundreds of tools (GitHub, Docker, AWS, Terraform, etc.)

Key Features:

Feature	Purpose
Freestyle Jobs	GUI-based jobs with basic logic
Pipeline Jobs	Code-defined pipelines using Jenkinsfile
Plugins	Extend Jenkins (e.g., Git, Docker, AWS CLI, Slack)
Distributed Builds	Run builds on multiple agents (Jenkins master/agent)

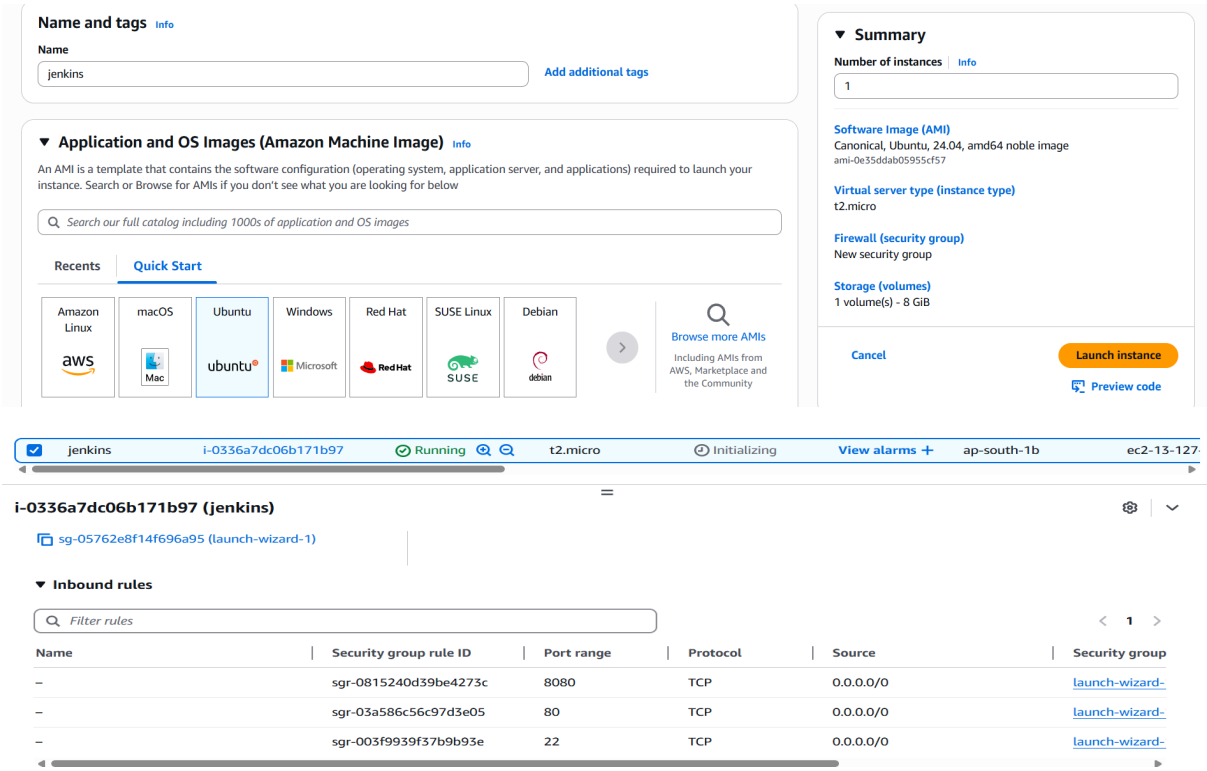
Feature	Purpose
Webhook Integration	Trigger jobs automatically from GitHub/GitLab etc.

2. Provision Jenkins on AWS EC2

We'll use an **Ubuntu-based EC2 instance** to host Jenkins.

Step-by-Step Setup

1. **Launch EC2 Instance**:
    - AMI: Ubuntu
    - Instance type: t2.micro(free-tier)
    - Key pair: Use an existing one or create a new one
    - Security Group:
        - Allow: **SSH (22)**, **HTTP (80)**, **Jenkins UI (8080)**
        - Optional later: **HTTPS (443)**



2. **Connect via SSH**:

```
ssh -i /path/to/your-key.pem ubuntu@<your-ec2-public-ip>
```

3. **Install Java (Jenkins dependency)**:

   sudo apt update

   sudo apt install -y openjdk-17-jdk

4. **Install Jenkins**:

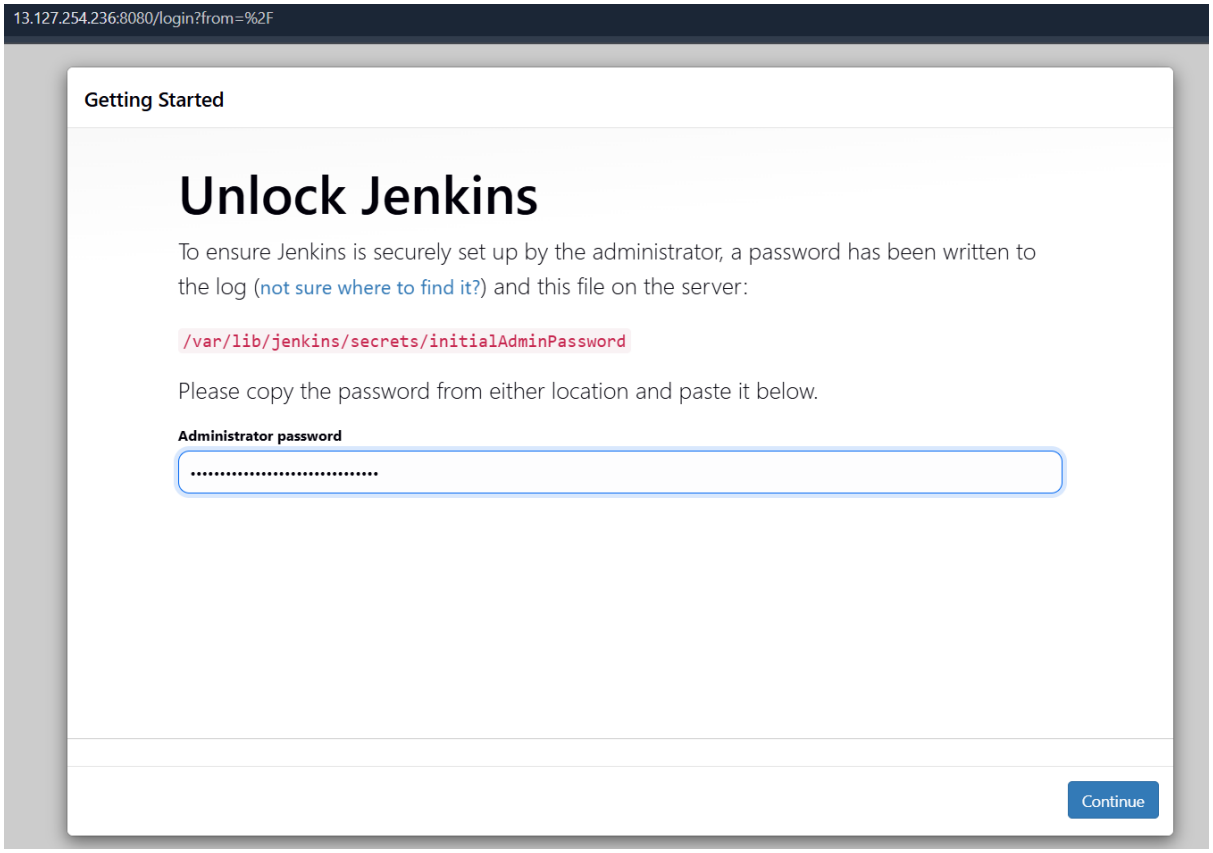
```
-> curl -fsSL https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key | sudo tee \
  /usr/share/keyrings/jenkins-keyring.asc > /dev/null


-> 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 update
-> sudo apt install -y jenkins
```

5. **Start and enable Jenkins**:

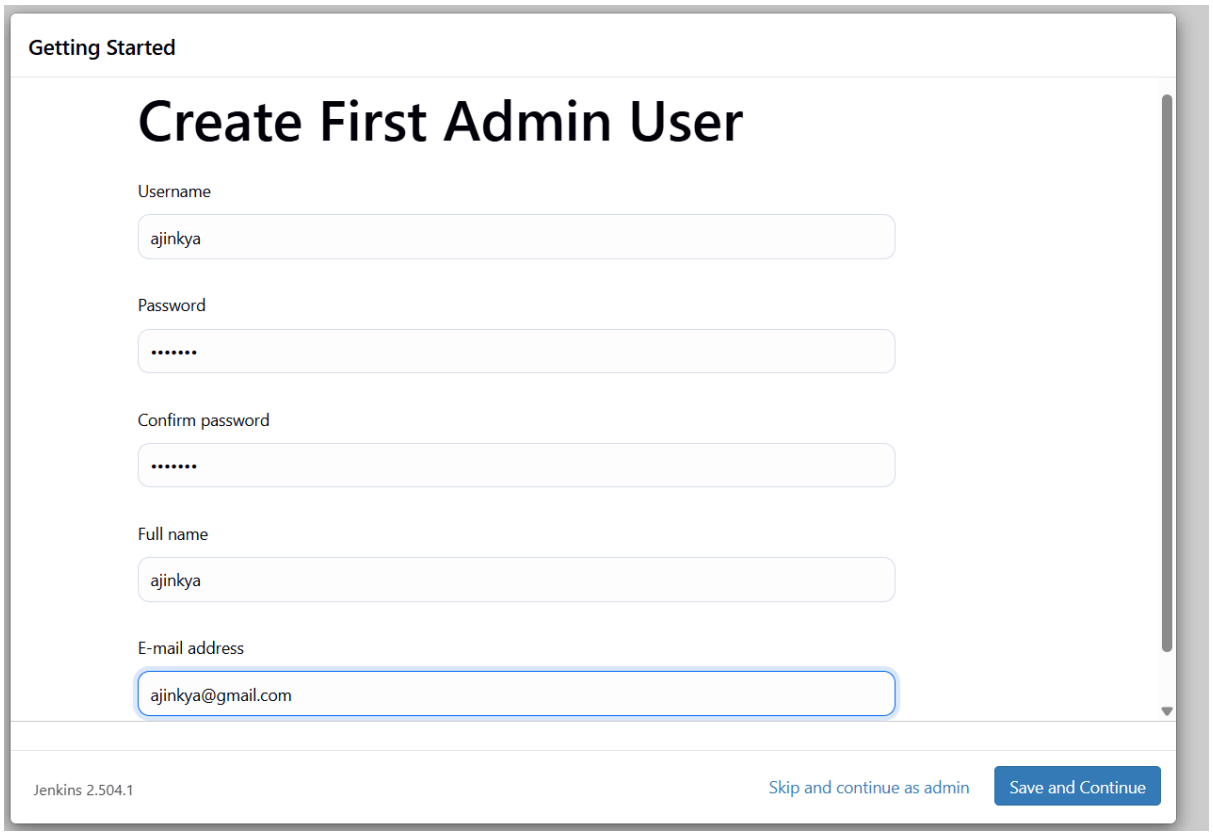
```
sudo systemctl enable jenkins
sudo systemctl start jenkins
```

6. **Access Jenkins UI**:
   - Go to: http://<your-ec2-public-ip>:8080
   - Unlock Jenkins with:

```
sudo cat /var/lib/jenkins/secrets/initialAdminPassword
```

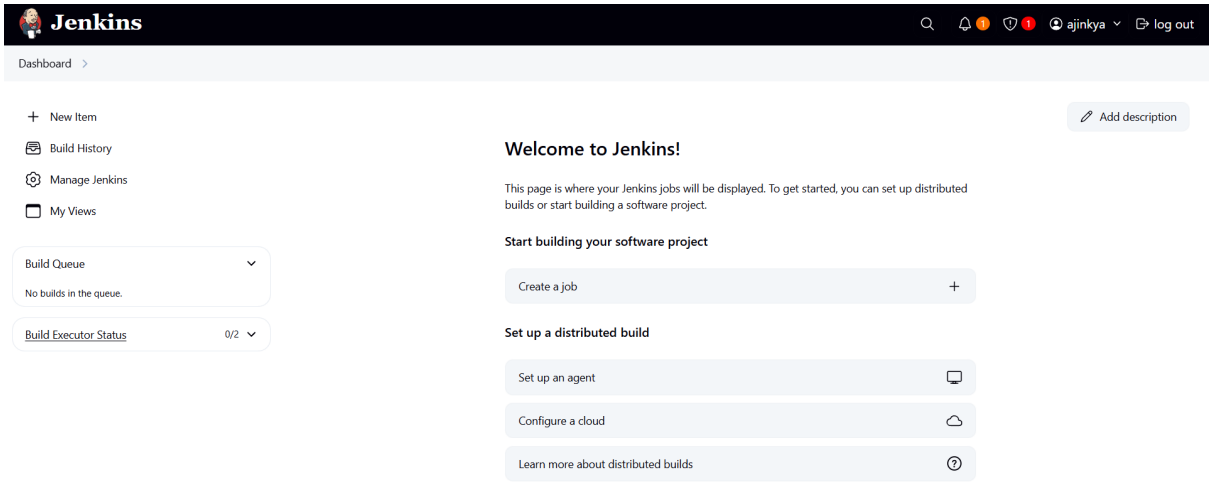


- Install **Suggested Plugins**
- Create **admin user**



3. Jenkins UI Concepts

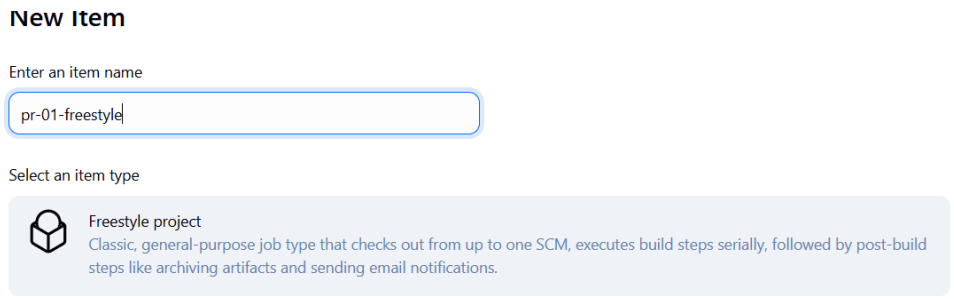
Component	Description
Dashboard	View and manage all jobs
New Item	Create jobs (freestyle, pipeline, multi-branch, etc.)
Build Now	Trigger a job manually
Build History	See past builds (logs, console output, status)
Configure	Set up job details (SCM, build steps, triggers)
Manage Jenkins	Global settings, plugins, nodes, security, etc.



Practical 01: First Jenkins job using freestyle and pipeline

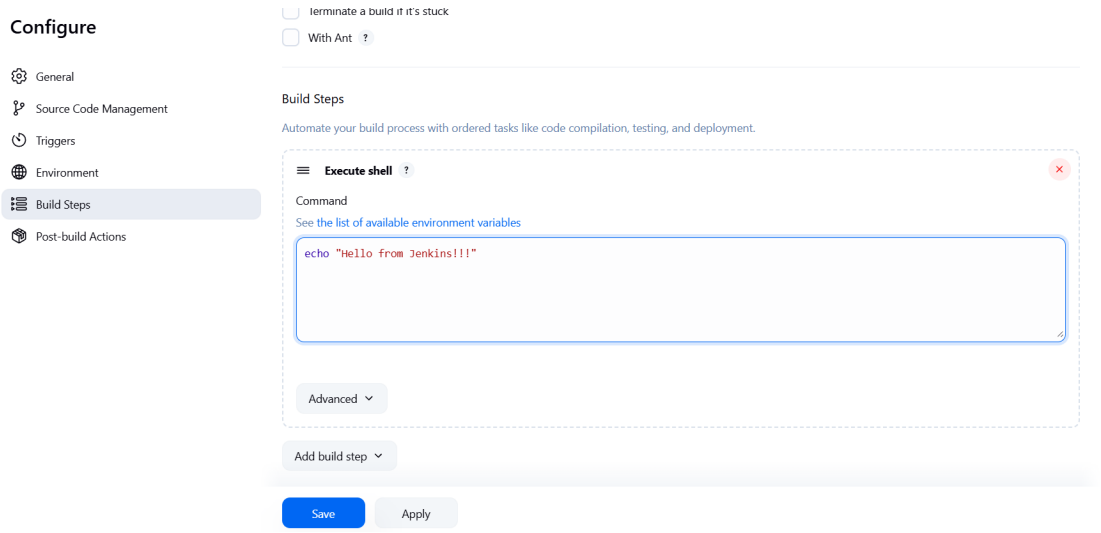
a) Freestyle Job

- Create job → "Freestyle Project"



- Add a **Build Step** → Execute Shell
- Example script:

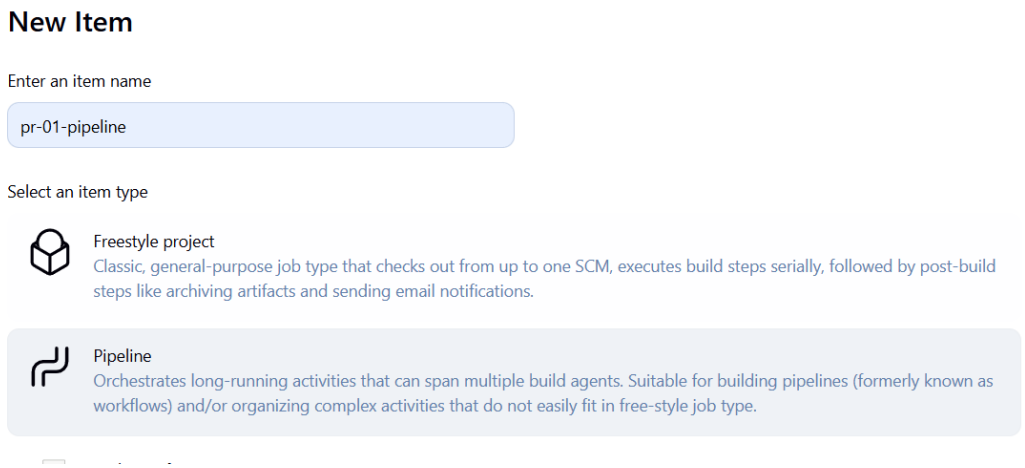
```
echo "Hello from Jenkins!"
```



- Save and build the job using build now.
- Go to build #1 to see console output and our message.

b) Pipeline Job

- Create job → “Pipeline”



- Paste in this example Jenkinsfile:

```
pipeline {
  agent any
  stages {
    stage('Greet') {
      steps {
        echo 'Hello from a pipeline!'
      }
    }
  }
}
```

- Save and build the job using build now.
- Go to build #1 to see console output and our message.

We will see below things further,

Core Jenkins Skills for DevOps Jobs

Goals:

- Master pipeline scripting (Declarative + Scripted)
- Use Git, GitHub, Docker, and AWS CLI in Jenkins
- Automate builds, tests, and deployments

Projects:

1. **CI/CD for a Node.js or Python app**
    - GitHub integration (webhook)
    - Build → Test → Dockerize → Deploy (to EC2 or ECS)
2. **Use Jenkins Shared Libraries**
    - Write reusable pipeline functions