

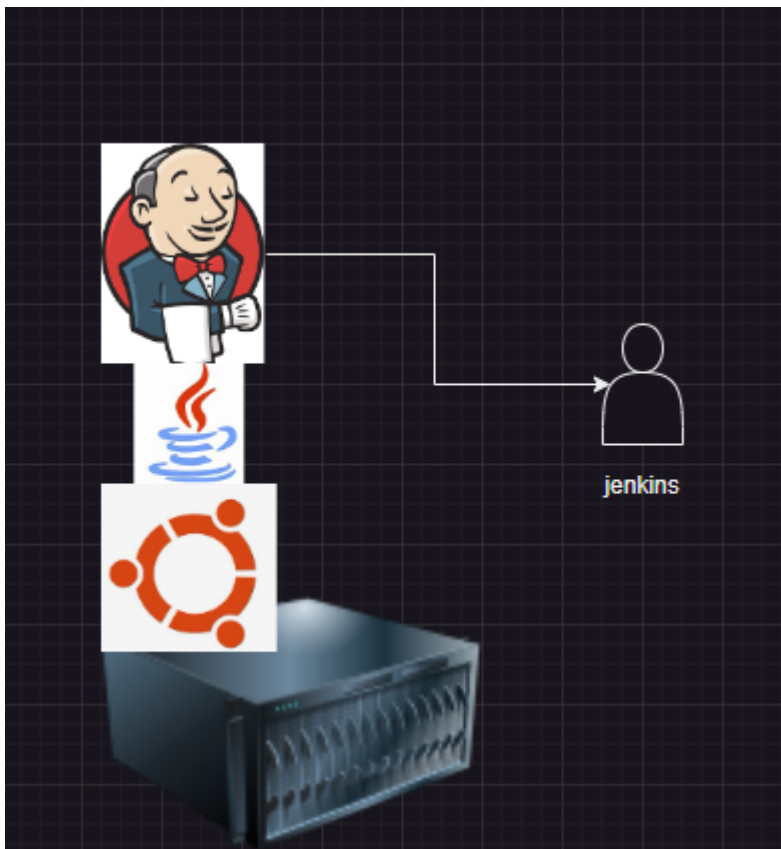
Continuous Integration with Jenkins

- Idea is to provide feedback to the developer who has pushed the commit, regarding its correctness.
- Jenkins is a CI/CD Engine which is developed in Java language.
- Installing jenkins on Linux
 - install jdk 17
 - [Refer Here](#) for the rest of the steps
- Commands

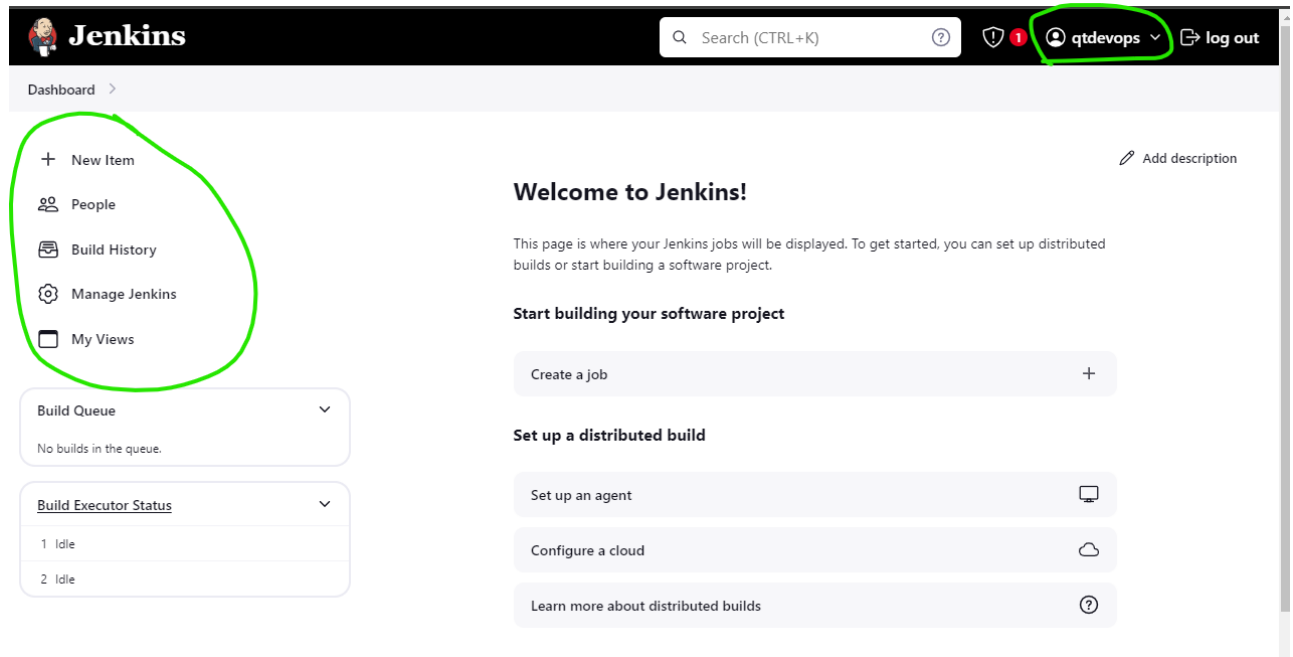
```
sudo apt update
sudo apt install openjdk-17-jdk -y
sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \
  https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key
echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \
  https://pkg.jenkins.io/debian-stable binary/ | sudo tee \
  /etc/apt/sources.list.d/jenkins.list > /dev/null
sudo apt-get update
sudo apt-get install jenkins -y
```

- This installation is used to run jenkins as a service/daemon on ubuntu.
- Installation of jenkins on linux leads to creation of a user `jenkins`

```
jenkins:x:115:122:Jenkins,,,:/var/lib/jenkins:/bin/bash
```



- Jenkins runs on 8080 port
- Navigate to <http://<ip>:8080> and enter the password as suggested and proceed with defaults.



- Jenkins can be defined as CRON on steroids
- In Jenkins we create projects and schedule them to run based on
 - manual trigger
 - periodic trigger
 - based on version control trigger
- Jenkins Projects are of 3 types
 - Free style project (UI Based)
 - Scripted Pipeline
 - Declarative Pipeline

Free Style Project

Exploration - 1

- Lets create a project which prints the current user and environmental variables when executed manually
- Manual commands

```
whoami
printenv
```

- Free style Project has following sections
 - General
 - Source Code Management
 - Build Triggers
 - Build Environment

- Build Steps
- Post Build Actions
- Lets acheive the above by configuring build steps

The screenshot shows the Jenkins Configuration page for a project named 'firstproject'. The breadcrumb trail at the top is 'Dashboard > firstproject > Configuration'. On the left, the 'Configure' section has a sidebar with options: General, Source Code Management, Build Triggers, Build Environment (highlighted with a green circle), Build Steps, and Post-build Actions. The main content area is titled 'Configure' and contains several checkboxes: 'Add timestamps to the Console Output', 'Inspect build log for published build scans', 'Terminate a build if it's stuck', and 'With Ant' (with a help icon). Below these is the 'Build Steps' section, which is also highlighted with a green circle. It contains a single step named 'Execute shell' with a help icon. The 'Command' field for this step contains the text 'whoami' and 'printenv'. A link 'See the list of available environment variables' is provided. At the bottom of the configuration page are 'Save' and 'Apply' buttons.

Dashboard > firstproject > Configuration

Configure

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

☐ Add timestamps to the Console Output

☐ Inspect build log for published build scans

☐ Terminate a build if it's stuck

☐ With Ant ?

Build Steps

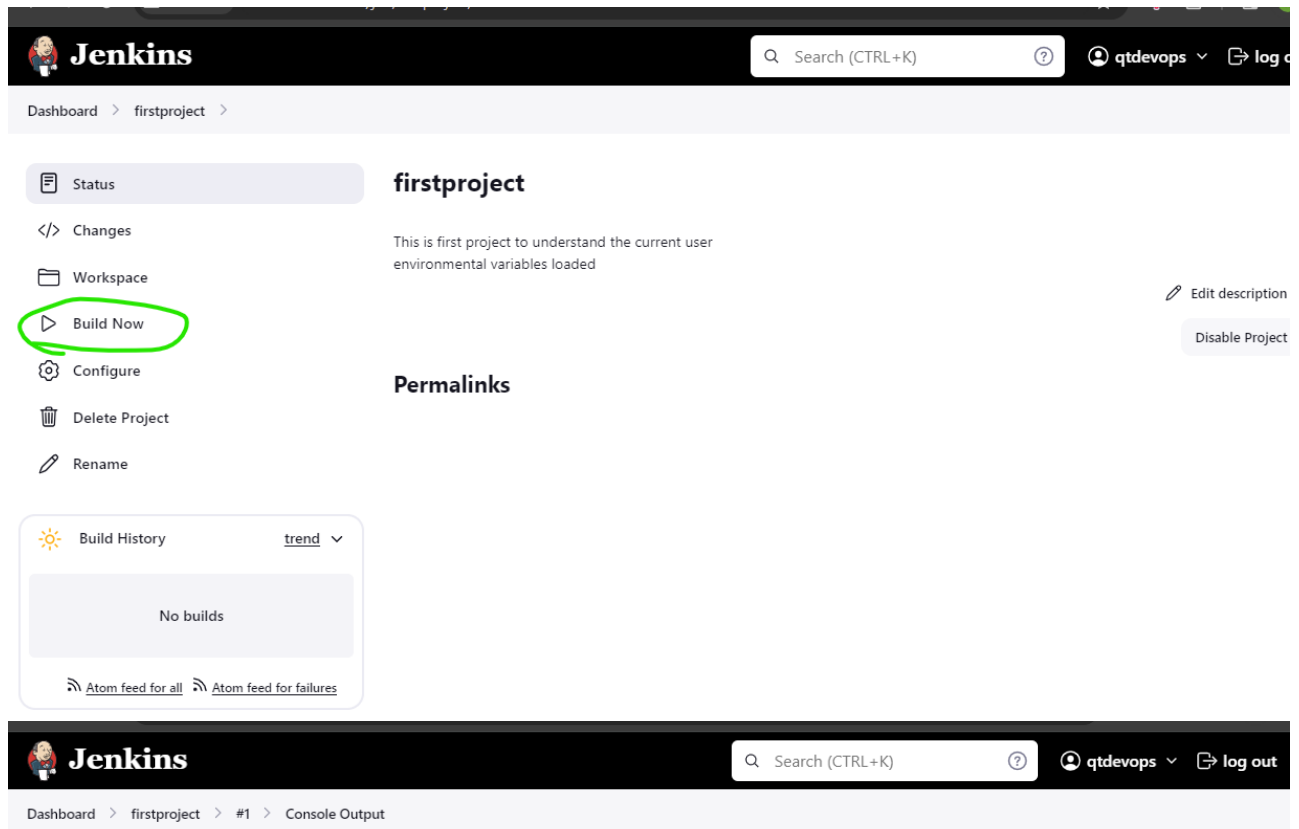
Execute shell ?

Command

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

```
whoami
printenv
```

Save **Apply**



The screenshot shows the Jenkins web interface. At the top, there's a search bar and a user profile dropdown. The main content area is divided into two sections. The left section, titled 'firstproject', contains a list of actions: Status, Changes, Workspace, Build Now (circled in green), Configure, Delete Project, and Rename. The right section, titled 'firstproject', contains a description: 'This is first project to understand the current user environmental variables loaded'. Below this, there's a 'Permalinks' section. At the bottom, there's a 'Build History' section showing 'No builds'. The bottom section of the screenshot shows the 'Console Output' tab selected, displaying the output of a build. The output includes the command 'whoami' and its result 'jenkins', which is circled in green.

Give sudo permissions to jenkins user with nopasssword

- Add permissions

```
sudo visudo
```

```
#Defaults:%sudo env_keep += "SSH_AGENT_PID SSH_AUTH_SOCK"

# Ditto for GPG agent
#Defaults:%sudo env_keep += "GPG_AGENT_INFO"

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL:ALL) ALL

# Members of the admin group may gain root privileges
%admin  ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL
jenkins ALL=(ALL:ALL) NOPASSWD:ALL
# See sudoers(5) for more information on "@include" directives:

@includedir /etc/sudoers.d
```

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute
 ^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify

Install maven and configure in jenkins

- Install maven
- Navigate to Jenkins Home => Manage Jenkins

Dashboard > Manage Jenkins > Tools

Maven installations

Add Maven

≡ Maven

Name

MAVEN_HOME

⚠ opt/maven is not a directory on the Jenkins controller (but perhaps it exists on some agents)

☐ Install automatically ?

Add Maven

Save Apply

Configure jenkins to build spring petclinic on freestyle project

- Create a new free style project

Configure

General

Source Code Management

Build Triggers

Build Environment

Build Steps

Post-build Actions

Git

Repositories

Repository URL

https://github.com/spring-projects/spring-petclinic.git

Credentials

- none -

+ Add

Advanced

Add Repository

Branches to build

Branch Specifier (blank for 'any')

*/main

Add Branch

Save

Apply

- In Build steps select top level maven targets

Dashboard > spc-freestyle > Configuration

Configure

General

Source Code Management

Build Triggers

Build Environment

Build Steps

Post-build Actions

☐ With Ant

Build Steps

Invoke top-level Maven targets

Maven Version

MAVEN_3.9.6

Goals

package

Advanced

Add build step

Save

Apply

6 / 8

← → ↻ Not secure 13.235.244.66:8080/job/spc-freestyle/2/console

Dashboard > spc-freestyle > #2 > Console Output

Progress (1): 1.0/1.0 MB
Progress (1): 1.0/1.0 MB
Progress (1): 1.0/1.0 MB
Progress (1): 1.0/1.0 MB
Progress (1): 1.0 MB

Downloaded from central: <https://repo.maven.apache.org/maven2/org/apache/commons/commons-compress/1.21/commons-compress-1.21.jar> (1.0 MB at 24 MB/s)
[INFO] Building jar: /var/lib/jenkins/workspace/spc-freestyle/target/spring-petclinic-3.2.0-SNAPSHOT.jar
[INFO] --- spring-boot:3.2.1:repackage (repackage) @ spring-petclinic ---
[INFO] Replacing main artifact /var/lib/jenkins/workspace/spc-freestyle/target/spring-petclinic-3.2.0-SNAPSHOT.jar with repackaged archive, adding nested dependencies in BOOT-INF/.
[INFO] The original artifact has been renamed to /var/lib/jenkins/workspace/spc-freestyle/target/spring-petclinic-3.2.0-SNAPSHOT.jar.original
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 06:23 min
[INFO] Finished at: 2024-04-05T02:50:24Z
[INFO] -----
Finished: SUCCESS

REST API Jenkins 2.440.2

Configure Jenkins to build nopcommerce on free style project

- Install dotnet 8

Dashboard > nop-freestyle > Configuration

Configure

General

Source Code Management

Build Triggers

Build Environment

Build Steps

Post-build Actions

repositories

Repository URL ?

<https://github.com/nopSolutions/nopCommerce.git>

Credentials ?

- none -

+ Add

Advanced

Add Repository

Branches to build ?

Branch Specifier (blank for 'any') ?

*/develop

Save Apply

Dashboard > nop-freestyle > Configuration

Configure

General

Source Code Management

Build Triggers

Build Environment

Build Steps

Post-build Actions

Build Steps

Execute shell ?

Command

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

dotnet build -c Release src/Presentation/Nop.Web/Nop.Web.csproj

Advanced

Add build step

Save

Apply

Dashboard > nop-freestyle > #1 > Console Output

Nop.Data -> /var/lib/jenkins/workspace/nop-freestyle/src/Libraries/Nop.Data/bin/Release/net8.0/Nop.Data.dll
/var/lib/jenkins/workspace/nop-freestyle/src/Libraries/Nop.Services/Connected
Services/EuropaCheckVatService/Reference.cs(328,52): warning CS0108: 'checkVatPortTypeClient.CloseAsync()' hides
inherited member 'ClientBase<checkVatPortType>.CloseAsync()'. Use the new keyword if hiding was intended.
[/var/lib/jenkins/workspace/nop-freestyle/src/Libraries/Nop.Services/Nop.Services.csproj]
Nop.Services -> /var/lib/jenkins/workspace/nop-
freestyle/src/Libraries/Nop.Services/bin/Release/net8.0/Nop.Services.dll
Nop.Web.Framework -> /var/lib/jenkins/workspace/nop-
freestyle/src/Presentation/Nop.Web.Framework/bin/Release/net8.0/Nop.Web.Framework.dll
Nop.Web -> /var/lib/jenkins/workspace/nop-freestyle/src/Presentation/Nop.Web/bin/Release/net8.0/Nop.Web.dll

Build succeeded.

/var/lib/jenkins/workspace/nop-freestyle/src/Libraries/Nop.Services/Connected
Services/EuropaCheckVatService/Reference.cs(328,52): warning CS0108: 'checkVatPortTypeClient.CloseAsync()' hides
inherited member 'ClientBase<checkVatPortType>.CloseAsync()'. Use the new keyword if hiding was intended.
[/var/lib/jenkins/workspace/nop-freestyle/src/Libraries/Nop.Services/Nop.Services.csproj]
1 Warning(s)
0 Error(s)

Time Elapsed 00:02:11.29
Finished: SUCCESS

REST API

Jenkins 2.44