

JENKINS DOCUMENTATION

What is JENKINS?

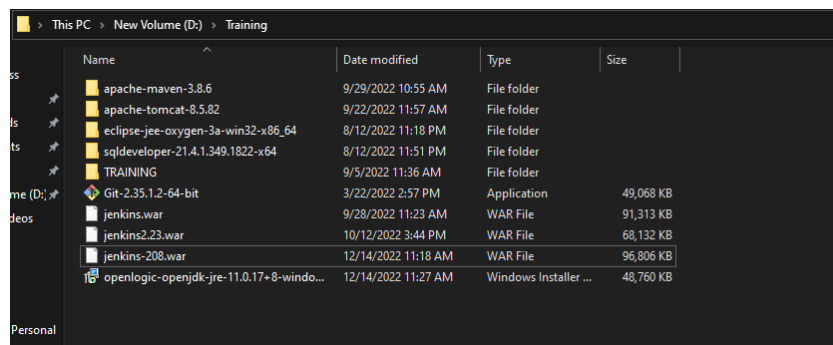
Jenkins is a tool that is used for automation, and it is an open-source server that allows all the developers to build, test and deploy software. It works or runs on java as it is written in java. By using Jenkins we can make a continuous integration of projects(jobs) or end-to-endpoint automation.

Installation of Jenkins in windows:

1. Before installing Jenkins existence of java on working systems as above said it works on java, so it's important to install and for that, installed version jdk8.
2. Now, navigating towards the official website of Jenkins i.e. www.jenkins.io. Here will have two options one is an LTS release and one is a weekly release. Download Jenkins for windows from the LTS section(recommended).
3. After download, open that zip(compressed) file, and by right click and choose the all extract option extract the file. We are doing extraction to unzip the compressed file.
4. After extracting, we can see a new folder of Jenkins. Open it and install Jenkins. After installing, it will automatically take you to the browser where localhost:8080 is running. 8080 is the by default port of Jenkins. And if the browser automatically doesn't open then simply search localhost:8080.

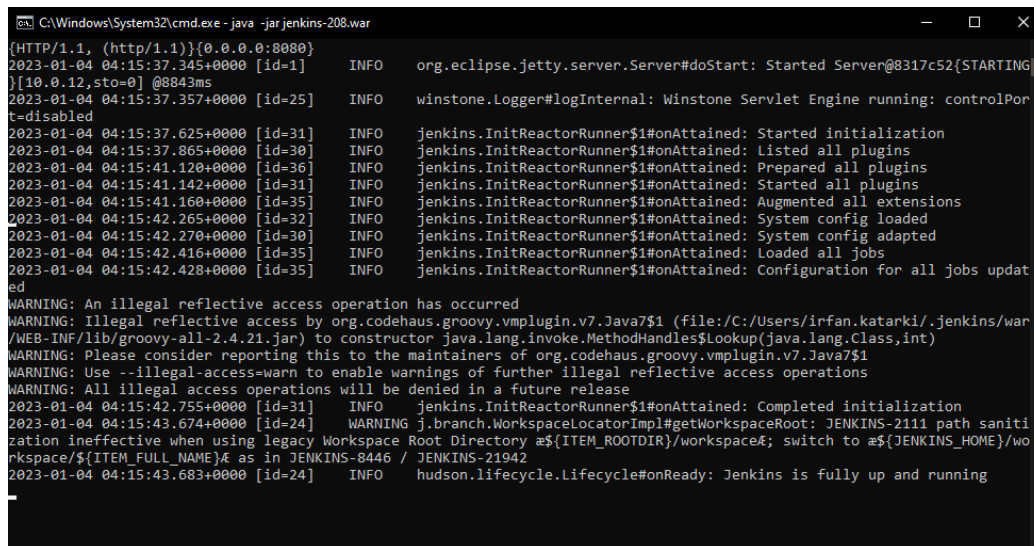
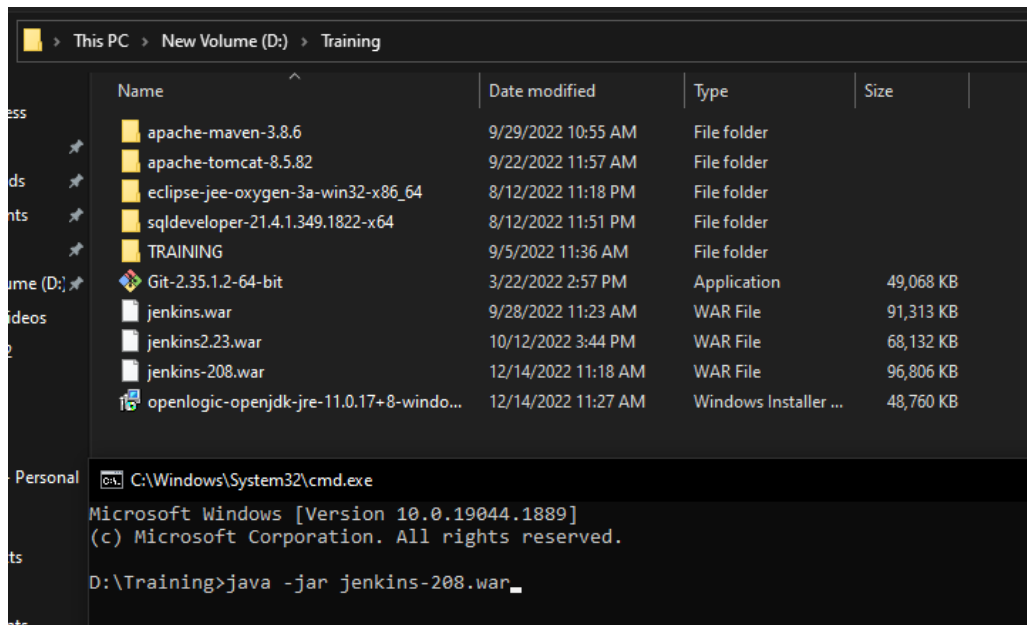
Usage of Jenkins for the running the MAVEN PROJECT build using JAVA JDK VERSION 1.8.

1. Download the [latest Jenkins WAR file](#) to an appropriate directory on the machine.



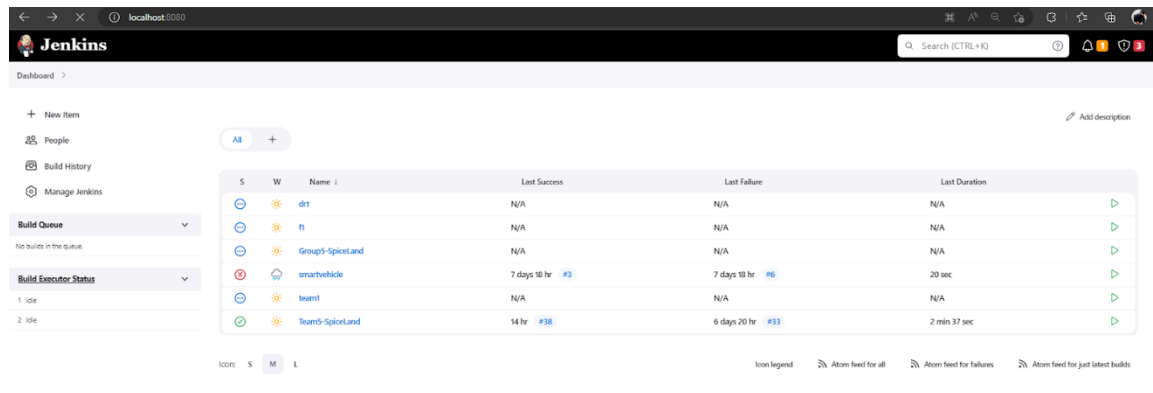
2. Running the command `java -jar jenkins.war`

JENKINS DOCUMENTATION

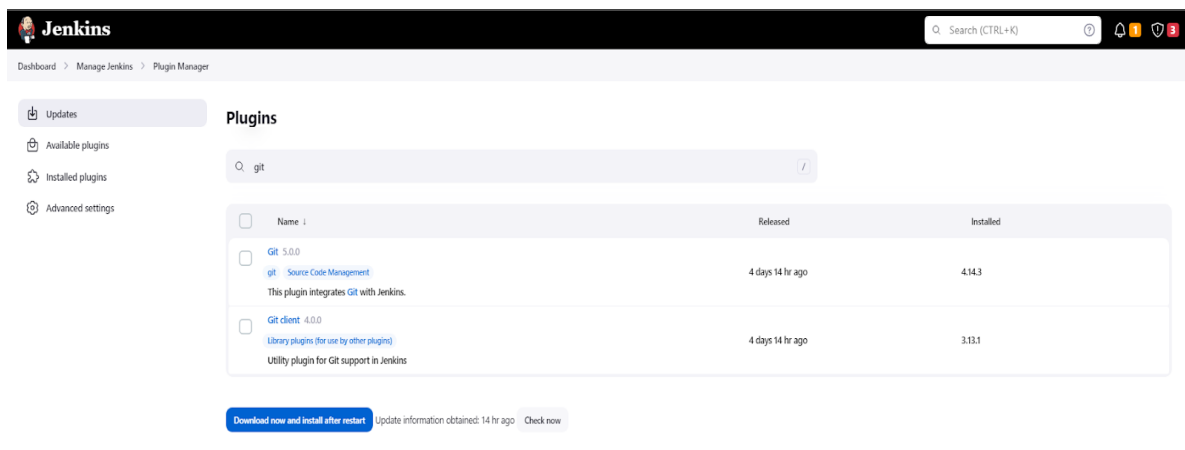


3. Browse to <http://localhost:8080> and after this, we will see the dashboard or interface of Jenkins. If we are installing it the first time then it will show some plugins which are recommended to install. And then, we are ready to use Jenkins.

JENKINS DOCUMENTATION



4. To set up git we need to install plugins by going under [Manage Jenkins](#) > [Manage Plugins](#) page in Jenkins.



5. After successful installation of git, we need to configure the email notifications by going under [Manage Jenkins](#) > [Configure Systems](#) page in Jenkins. This is the global configuration email-notification from sender to get notified for recipients.

JENKINS DOCUMENTATION

E-mail Notification

SMTP server

smtp.gmail.com

Default user e-mail suffix [?](#)

☒ Use SMTP Authentication [?](#)

User Name

irfanahmadn@gmail.com

Password

Concealed

[Change Password](#)

☒ Use SSL [?](#)

☐ Use TLS

SMTP Port [?](#)

465

Reply-To Address

Charset


UTF-8


☐ Test configuration by sending test e-mail


- Click on a new item, and then we will have a page on which we have to give the name to the job and choose 'freestyle project' or any other option according to the project need and then click 'ok'.


Enter an item name


= Required field


**Freestyle project**
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.


**Maven project**
Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.

**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 builds, etc.

**Folder**
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.

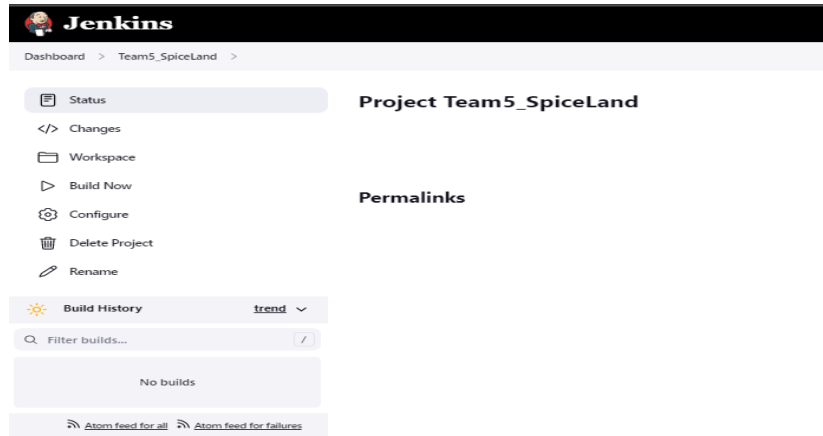
**Multibranch Pipeline**
Creates a set of Pipeline projects according to detected branches in one SCM repository.

**Organization Folder**
Creates a set of multibranch project subfolders by scanning for repositories.

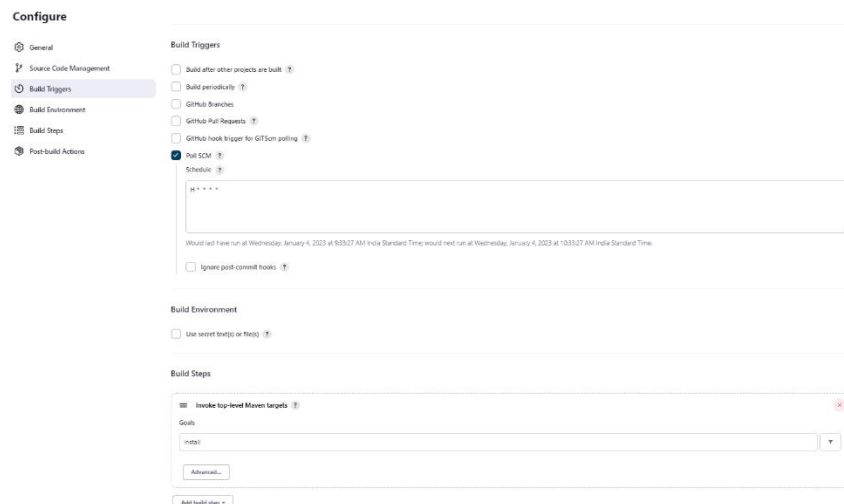
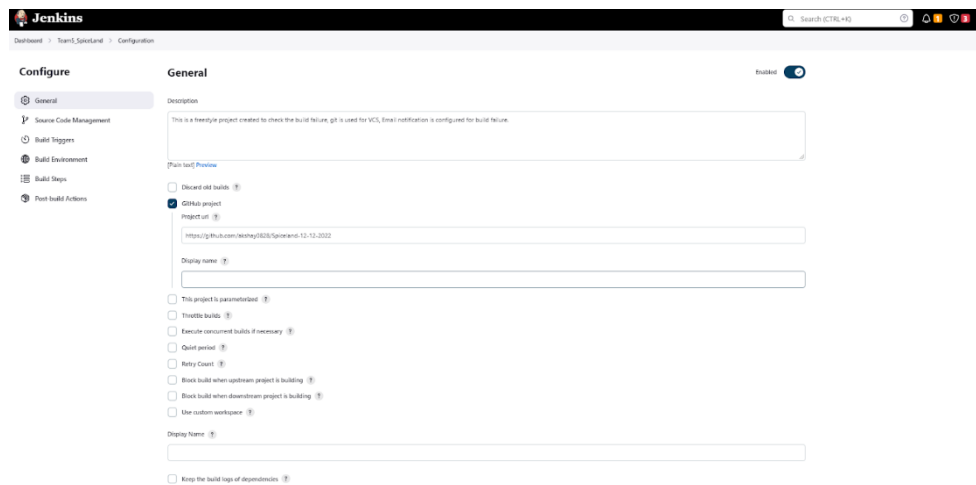
If you want to create a new item from other existing, you can use this option:

**Copy from**

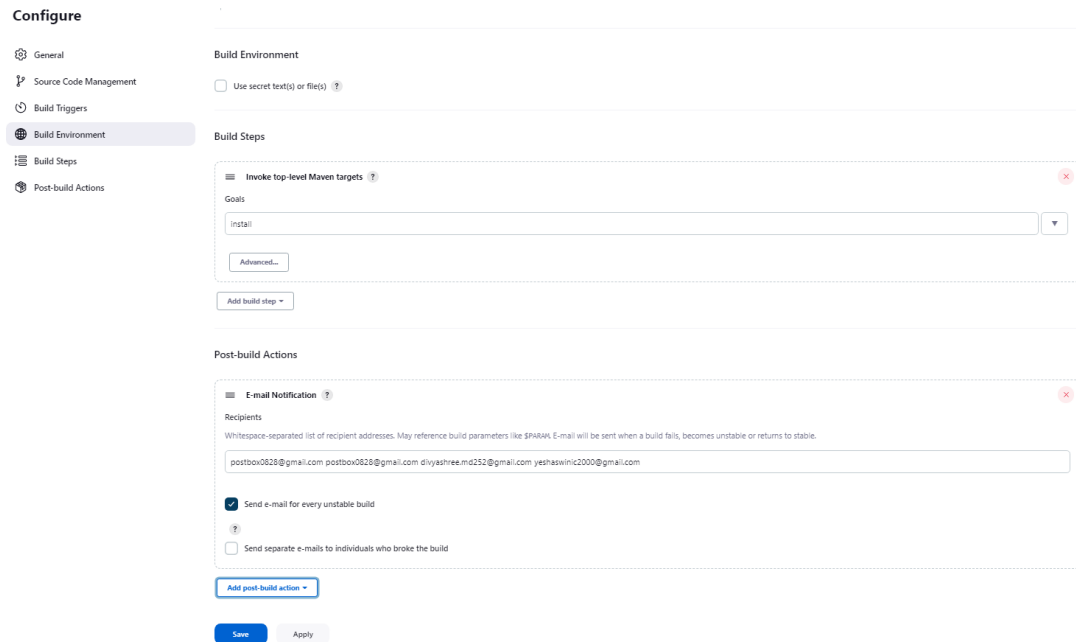
JENKINS DOCUMENTATION



- After this, we will reach a page where you have different options (like build, build triggers, source code management) that help us manage the created job.



JENKINS DOCUMENTATION



The screenshot shows the Jenkins configuration page for a new job. The left sidebar contains the following menu items: General, Source Code Management, Build Triggers, Build Environment (selected), Build Steps, and Post-build Actions. The main content area is divided into three sections: Build Environment, Build Steps, and Post-build Actions. In the Build Environment section, there is a checkbox for 'Use secret text(s) or file(s)' which is unchecked. The Build Steps section contains a single step named 'Invoke top-level Maven targets' with a goal of 'install'. The Post-build Actions section contains a single action named 'E-mail Notification' with a list of recipients: 'postbox0828@gmail.com postbox0828@gmail.com divyashree.md252@gmail.com yeshaswinic2000@gmail.com'. The 'Send e-mail for every unstable build' checkbox is checked. At the bottom, there are 'Save' and 'Apply' buttons.

Configure

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

Build Environment

☐ Use secret text(s) or file(s) ?

Build Steps

Invoke top-level Maven targets ?

Goals

install

Advanced...

Add build step

Post-build Actions

E-mail Notification ?

Recipients

Whitespace-separated list of recipient addresses. May reference build parameters like \$PARAM. E-mail will be sent when a build fails, becomes unstable or returns to stable.

postbox0828@gmail.com postbox0828@gmail.com divyashree.md252@gmail.com yeshaswinic2000@gmail.com

☒ Send e-mail for every unstable build

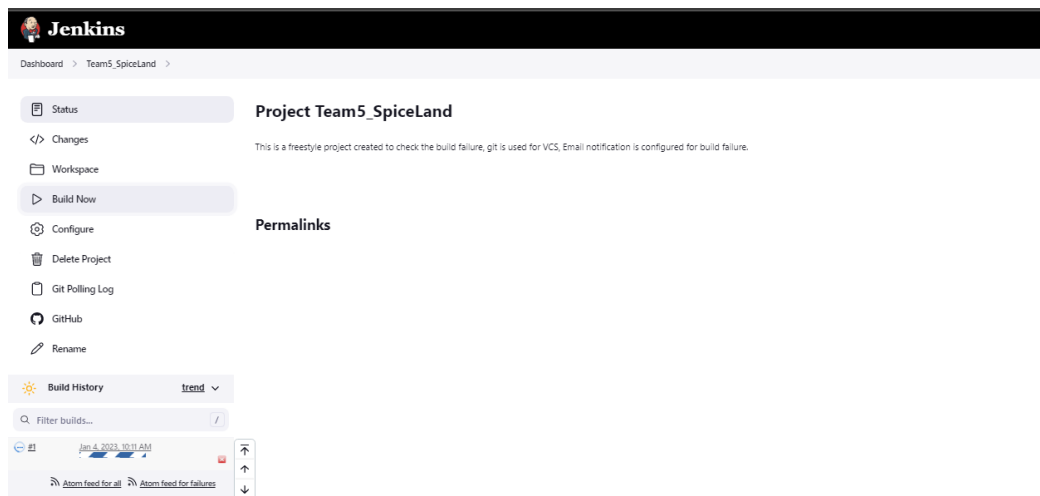
☐ Send separate e-mails to individuals who broke the build

Add post-build action

Save Apply

Then click apply and save. So, the first job is created.

- Now, we will run it for which click the 'build now' option and a building history will be created then click on it.



The screenshot shows the Jenkins dashboard for a project named 'Team5_SpiceLand'. The top bar is black with the Jenkins logo and the project name. Below the bar, there is a navigation menu on the left with the following items: Status, Changes, Workspace, Build Now (highlighted), Configure, Delete Project, Git Polling Log, GitHub, and Rename. The main content area is divided into two sections: 'Project Team5_SpiceLand' and 'Permalinks'. The 'Project Team5_SpiceLand' section contains a description: 'This is a freestyle project created to check the build failure, git is used for VCS, Email notification is configured for build failure.' The 'Permalinks' section is empty. At the bottom, there is a 'Build History' section with a 'trend' dropdown and a search bar. The build history shows a single build with the name 'Jan 4, 2023, 10:11 AM' and a status of 'Success'. There are also links for 'Atom feed for all' and 'Atom feed for failures'.

Jenkins

Dashboard > Team5_SpiceLand >

Project Team5_SpiceLand

This is a freestyle project created to check the build failure, git is used for VCS, Email notification is configured for build failure.

Permalinks

Build History trend

Filter builds...

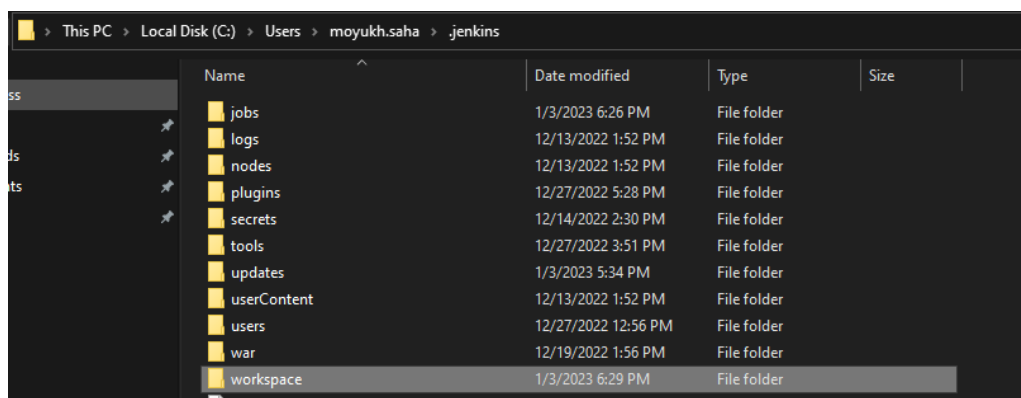
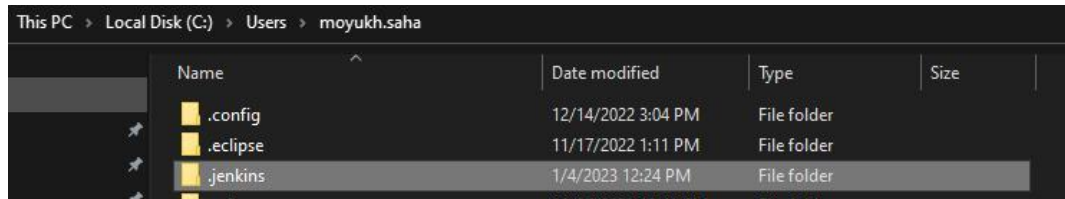
Jan 4, 2023, 10:11 AM

Atom feed for all Atom feed for failures

JENKINS DOCUMENTATION

- Now, Click on console output, and we can see our output. Also, by console output, you can see whether your job is failed or successful.

The first run of the Jenkins war file will create a folder named .jenkins under the user's home directory, and a configuration routine will run.



```
Dashboard > Team5.SpiceLand > #1

> git.exe checkout -f 1d0a1d99838ad25f5d99087ef4e8999ef7032 # timeout=10
Commit message: "Each seller can add each product only once."
First time build. Skipping changelog.
[workspace] $ cmd.exe /C "rem -f Spice-Land\pom.xml install && exit %ERRORLEVEL%"
[INFO] Scanning for projects...
[INFO]
[INFO] -----< com.valtech.spring.boot:Spice-Land >-----
[INFO] Building SpringSecurity 4.0.1-SNAPSHOT
[INFO]
[INFO] -----[ jar ]-----
[INFO]
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-resources-plugin/3.2.0/maven-resources-plugin-3.2.0.jar
[WARNING] The POM for org.apache.maven.plugins:maven-resources-plugin:3.2.0 is missing, no dependency information available
[INFO]
[INFO] --- maven-resources-plugin:3.2.0:resources (default-resources) @ Spice-Land ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] Using 'UTF-8' encoding to copy filtered properties files.
[INFO] Copying 1 resource
[INFO] Copying 30 resources
[INFO]
[INFO] --- maven-compiler-plugin:3.10.1:compile (default-compile) @ Spice-Land ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 25 source files to C:\Users\irfan.katarik\jenkins\jobs\Team5.SpiceLand\workspace\Spice-Land\target\classes
[INFO] C:\Users\irfan.katarik\jenkins\jobs\Team5.SpiceLand\workspace\Spice-Land\src\main\java\com\valtech\spring\security\controllers\DeliveryController.java: Some input files use or override a deprecated API.
[INFO] C:\Users\irfan.katarik\jenkins\jobs\Team5.SpiceLand\workspace\Spice-Land\src\main\java\com\valtech\spring\security\controllers\DeliveryController.java: Recompile with -Xlint:deprecation for details.
[INFO]
[INFO] -----
[INFO] COMPILATION ERROR :
[INFO]
[INFO] /C:/Users/irfan.katarik/.jenkins/jobs/Team5.SpiceLand/workspace/Spice-Land/src/main/java/com/valtech/spring/security/controllers/HelloController.java:[5,22] package java.awt does not exist
[INFO] 1 error
[INFO]
[INFO] -----
[INFO] BUILD FAILURE
[INFO]
[INFO] Total time: 39.880 s
[INFO] Finished at: 2023-01-04T08:12:46+05:30
[INFO]
[ERROR] Failed to execute goal org.apache.maven.plugins:maven-compiler-plugin:3.10.1:compile (default-compile) on project Spice-Land: Compilation failure
[ERROR] /C:/Users/irfan.katarik/.jenkins/jobs/Team5.SpiceLand/workspace/Spice-Land/src/main/java/com/valtech/spring/security/controllers/HelloController.java:[5,22] package java.awt does not exist
[ERROR]
[ERROR] -- [help 1]
[ERROR]
[ERROR] To see the full stack trace of the errors, re-run Maven with the -e switch.
[ERROR] Re-run Maven using the -s switch to enable full debug logging.
[ERROR]
[ERROR] For more information about the errors and possible solutions, please read the following articles:
[ERROR] [help 1] http://wiki.apache.org/maven/MavenBuildException
Build step 'Invoke top-level Maven targets' marked build as failure
Sending e-mails to: postboud2@gmail.com dnyashree.nd2@gmail.com yeshashwin2000@gmail.com
```

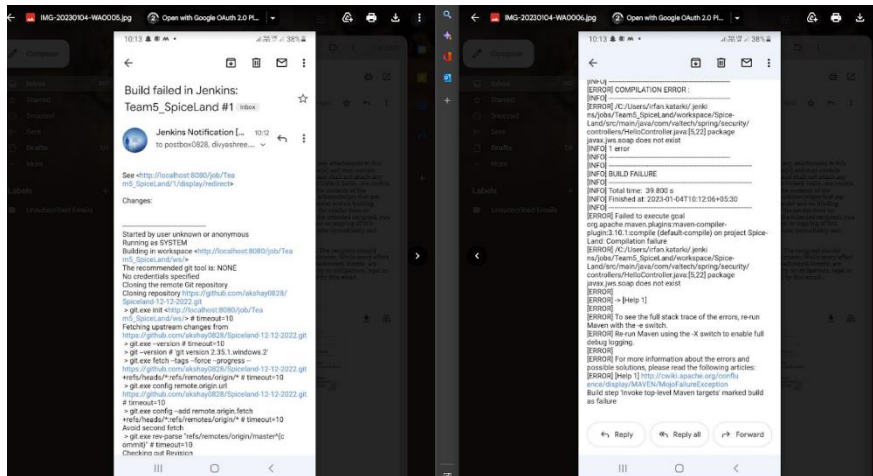
- Once the compilation error occurs recipients are notified through email from sender.

```
Build failed in Jenkins: Team5_SpiceLand #1 >

Jenkins Notification [ High Priority ] - info@madrix@gmail.com -
to:postman@kodi, dmytro.mak@kodi, yefimov@kodi<>
See <https://localhost:8080/job/Team5_SpiceLand/1/display/redirc>

Changes:

-----
Started by user unknown or anonymous
Running as SYSTEM
Building in workspace <https://localhost:8080/job/Team5_SpiceLand/ws/>
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository:
Cloning repository: https://github.com/yahya0629/SpiceLand-12-12-2022.git
> git exec init <https://localhost:8080/job/Team5_SpiceLand/ws/> # timeout=10
Fetching upstream changes from https://github.com/yahya0629/SpiceLand-12-12-2022.git
> git exec --version # timeout=10
> git exec --version # git version 2.35.1 Windows 7
> git exec fetch --tags --force --progress - https://github.com/yahya0629/SpiceLand-12-12-2022.git refs/head/master:refs/remotes/origin/* # timeout=10
> git exec config remote origin url https://github.com/yahya0629/SpiceLand-12-12-2022.git # timeout=10
> git exec config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
> git exec rm -rfC .&#x2F;.git&#x2F;remotes&#x2F;origin&#x2F;master # timeout=10
Checking out Revision fddaf0693dca255f8959167e46d7996af732 (refs/remotes/origin/master)
> git exec config core.sparsecheckout # timeout=10
> git exec checkout -f fddaf0693dca255f8959167e46d7996af732 # timeout=10
Commit message: "Each seller can add each product only once."
First time build. Skipping changelog
[workspace] $ cmd.exe /C "cm .\Spice-Land\pom.xml && %N%ERRORLEVEL% %N%"
INFO: Scanning for projects...
INFO:
INFO: ~~~~~
INFO: Building SpringSecurity 6.0.1-SNAPSHOT
INFO: ~~~~~
INFO: Downloading from central: https://repo.maven.apache.org/maven2/org-commons-jpmorgan-1.0/pom.xml 1.0 pom
[WARNING] The POM for org-commons:jpmorgan:jar:1.0 is missing, no dependency information available
INFO:
```



```

[INFO] ALREADY EXIST
[INFO] Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.07 s - in com.valtech.spring.security.test.UserServiceTest
2023-01-03 19:01:20.502 INFO 33784 --- [onShutdownHook] j.LocalContainerEntityManagerFactoryBean : Closing JPA EntityManagerFactory for persistence unit 'default'
2023-01-03 19:01:20.504 INFO 33784 --- [onShutdownHook] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Shutdown initiated...
2023-01-03 19:01:20.505 INFO 33784 --- [onShutdownHook] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Shutdown completed.
2023-01-03 19:01:20.569 INFO 33784 --- [onShutdownHook] j.LocalContainerEntityManagerFactoryBean : Closing JPA EntityManagerFactory for persistence unit 'default'
2023-01-03 19:01:20.570 INFO 33784 --- [onShutdownHook] com.zaxxer.hikari.HikariDataSource : HikariPool-2 - Shutdown initiated...
2023-01-03 19:01:20.610 INFO 33784 --- [onShutdownHook] com.zaxxer.hikari.HikariDataSource : HikariPool-2 - Shutdown completed.
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 11, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO]
[INFO]
[INFO] --- maven-war-plugin:3.3.2:war (default-war) @ Spice-Land ---
[INFO] Packaging webapp
[INFO] Assembling webapp [Spice-Land] in [C:\Users\Irfan.katarik\jenkins\jobs\Teams-SpiceLand\workspace\Spice-Land\target\Spice-Land-0.0.1-SNAPSHOT]
[INFO] Processing war project
[INFO] Copying webapp resources [C:\Users\Irfan.katarik\jenkins\jobs\Teams-SpiceLand\workspace\Spice-Land\src\main\webapp]
[INFO] Building war: C:\Users\Irfan.katarik\jenkins\jobs\Teams-SpiceLand\workspace\Spice-Land\target\Spice-Land-0.0.1-SNAPSHOT.war
[INFO]
[INFO]
[INFO] --- spring-boot-maven-plugin:2.7.4:repackage (repackage) @ Spice-Land ---
[INFO] Replacing main artifact with repackaged archive
[INFO]
[INFO]
[INFO] --- maven-install-plugin:2.5.2:install (default-install) @ Spice-Land ---
[INFO] Installing C:\Users\Irfan.katarik\jenkins\jobs\Teams-SpiceLand\workspace\Spice-Land\target\Spice-Land-0.0.1-SNAPSHOT.war to C:\Users\Irfan.katarik\m2\repository\com\valtech\spring\boot\Spice-Land\0.0.1-SNAPSHOT\Spice-Land-0.0.1-SNAPSHOT.war
[INFO] Installing C:\Users\Irfan.katarik\jenkins\jobs\Teams-SpiceLand\workspace\Spice-Land\pom.xml to C:\Users\Irfan.katarik\m2\repository\com\valtech\spring\boot\Spice-Land\0.0.1-SNAPSHOT\Spice-Land-0.0.1-SNAPSHOT.pom
[INFO] .....
[INFO] BUILD SUCCESS
[INFO] .....
[INFO] Total time: 02:19 min
[INFO] Finished at: 2023-01-03T19:01:46+05:30
[INFO] .....
[INFO] Finished: SUCCESS

```