

## DEVOPS TASK 5

Step 1: Clone the repository in ubuntu using the url :

<https://github.com/Jervinjeno/spring-framework.git>

```
kowsika@kowsi: ~/spring-framework
kowsika@kowsi:~/spring-framework$ git remote remove origin
kowsika@kowsi:~/spring-framework$ git remote add origin https://github.com/KowsikaJ/Springframework_Devops.git
kowsika@kowsi:~/spring-framework$ git remote -v
origin https://github.com/KowsikaJ/Springframework_Devops.git (fetch)
origin https://github.com/KowsikaJ/Springframework_Devops.git (push)
kowsika@kowsi:~/spring-framework$ git push -u origin main
Username for 'https://github.com': KowsikaJ
Password for 'https://github.com':
remote: Support for password authentication was removed on August 13, 2021.
remote: Please see https://docs.github.com/get-started/getting-started-with-git/about-remote-repositories#cloning-with-https-urls for information on currently recommended modes of authentication.
fatal: Authentication failed for 'https://github.com/KowsikaJ/Springframework_Devops.git/'
kowsika@kowsi:~/spring-framework$ git push https://KowsikaJ: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@github.com/KowsikaJ/Springframework_Devops.git
Enumerating objects: 6703, done.
Counting objects: 100% (6703/6703), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2366/2366), done.
Writing objects: 100% (6703/6703), 1.52 MiB | 1.38 MiB/s, done.
Total 6703 (delta 3328), reused 6703 (delta 3328), pack-reused 0
remote: Resolving deltas: 100% (3328/3328), done.
To https://github.com/KowsikaJ/Springframework_Devops.git
 * [new branch]      main -> main
kowsika@kowsi:~/spring-framework$
```

Step 2: Install maven dependency and JDK

```
root@kowsi: ~
E: Could not open lock file /var/lib/dpkg/lock-frontent - open (13: Permission denied)
E: Unable to acquire the dpkg frontend lock (/var/lib/dpkg/lock-frontent), are you root?
kowsika@kowsi:~/java_project$ sudo su -
[sudo] password for kowsika:
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 5.15.167.4-microsoft-standard-WSL2 x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

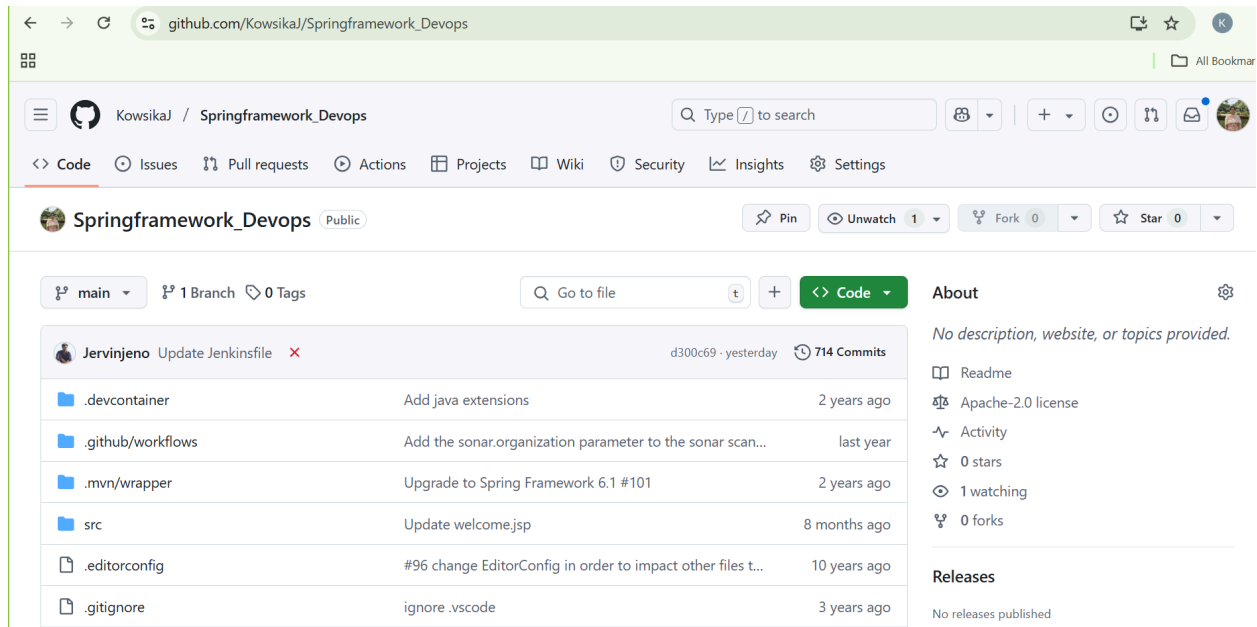
System information as of Fri Mar 21 10:24:04 UTC 2025

System load:  0.09          Processes:    53
Usage of /:   0.7% of 1006.85GB   Users logged in: 2
Memory usage: 37%             IPv4 address for eth0: 192.168.236.233
Swap usage:   0%

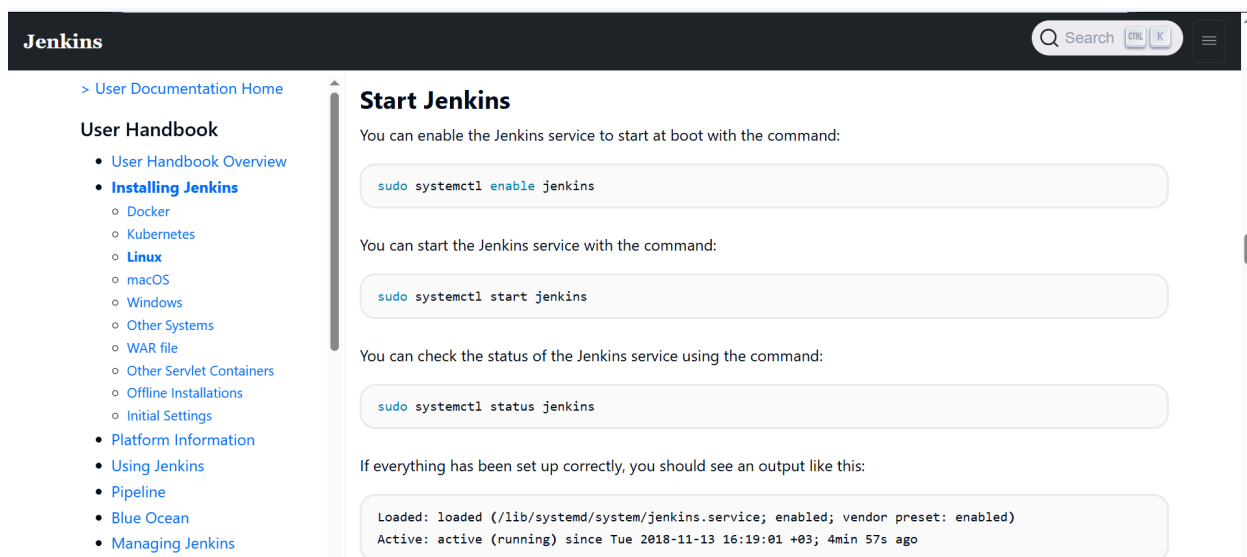
 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.

   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

This message is shown once a day. To disable it please create the
/root/.hushlogin file.
root@kowsi:~# apt-get install maven -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libaopalliance-java libapache-pom-java libatinject-jsr330-api-java libcdi-api-java libcommons-cli-java libcommons-io-java libcommons-lang3-java
  libcommons-parent-java liberror-prone-java libgeronimo-annotation-1.3-spec-java libgeronimo-interceptor-3.0-spec-java libguava-java libguice-java
  libjansi-java libjsr305-java libmaven-parent-java libmaven-resolver-java libmaven-shared-utils-java libmaven3-core-java libplexus-cipher-java
  libplexus-classworlds-java libplexus-component-annotations-java libplexus-interpolation-java libplexus-sec-dispatcher-java libplexus-utils2-java
  libsisu-inject-java libsisu-plexus-java libslf4j-java libwagon-file-java libwagon-http-shaded-java libwagon-provider-api-java
Suggested packages:
  libatinject-jsr330-api-java-doc libel-api-java libcommons-io-java-doc libasm-java libcglib-java libjsr305-java-doc libmaven-shared-utils-java-doc
  liblogback-java libplexus-utils2-java-doc junit4 testng libcommons-logging-java liblog4j1.2-java
The following NEW packages will be installed:
  libaopalliance-java libapache-pom-java libatinject-jsr330-api-java libcdi-api-java libcommons-cli-java libcommons-io-java libcommons-lang3-java
  libcommons-parent-java liberror-prone-java libgeronimo-annotation-1.3-spec-java libgeronimo-interceptor-3.0-spec-java libguava-java libguice-java
  libjansi-java libjsr305-java libmaven-parent-java libmaven-resolver-java libmaven-shared-utils-java libmaven3-core-java libplexus-cipher-java
  libplexus-classworlds-java libplexus-component-annotations-java libplexus-interpolation-java libplexus-sec-dispatcher-java libplexus-utils2-java
  libsisu-inject-java libsisu-plexus-java libslf4j-java libwagon-file-java libwagon-http-shaded-java libwagon-provider-api-java
```



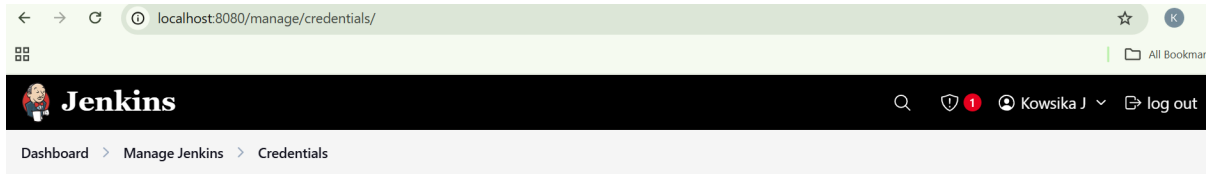
Step 3: In a new ubuntu terminal , start jenkins using the following 3 commands:



Step 4: In chrome type **localhost:8080** and enter .Then give the username and password to login into jenkins.

Step 5: Go to Manage Jenkins and choose **Credentials** from Security.

Step 6: Add new credentials for github & docker-hub.



## Credentials

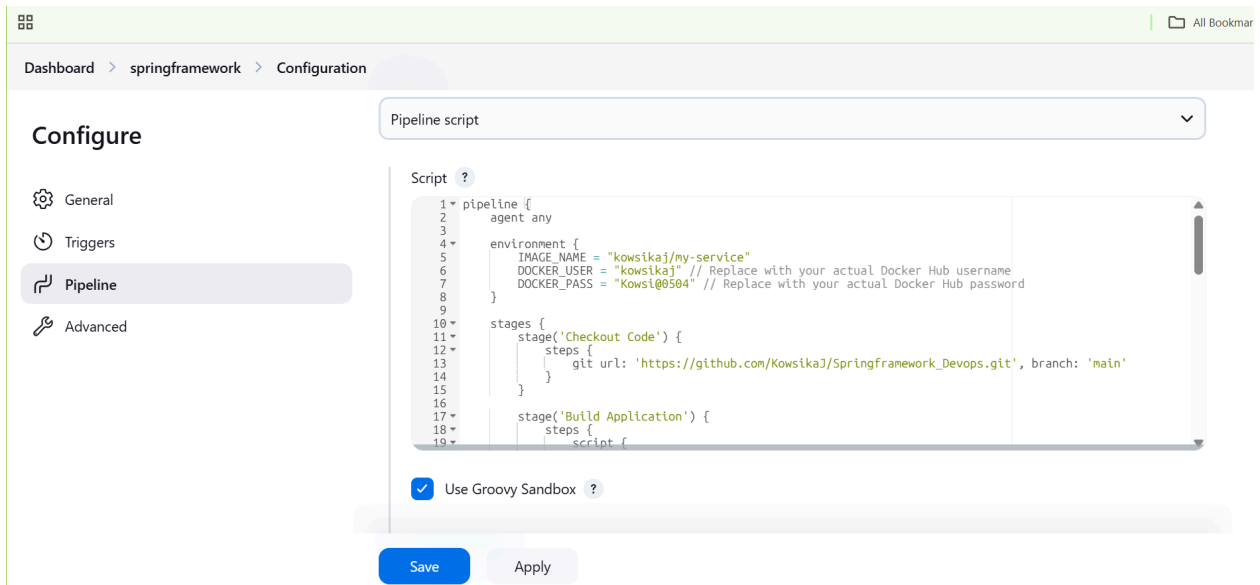
T	P	Store ↓	Domain	ID	Name
		System	(global)	github	KowsikaJ/*****
		System	(global)	dockerhub	kowsikaj/*****

## Stores scoped to Jenkins

P	Store ↓	Domains
	System	(global)

Step 7: Create a new item springframework and choose pipeline project in jenkins.

Step 8: Go to **Configure** in springframework item, and paste the edited jenkins file in the script with proper indentation.



Step 9: Click on Save and Build now.

Step 10: This gives an error message as permission denied

Dashboard > springframework > #11

```
+ docker build -t kowsikaj/my-service:latest .
ERROR: permission denied while trying to connect to the Docker daemon socket at
unix:///var/run/docker.sock: Head "http://%2Fvar%2Frun%2Fdocker.sock/_ping": dial unix
/var/run/docker.sock: connect: permission denied
[Pipeline] }
[Pipeline] // script
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Login to Docker Registry)
Stage "Login to Docker Registry" skipped due to earlier failure(s)
[Pipeline] getContext
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Push Image to Docker Registry)
Stage "Push Image to Docker Registry" skipped due to earlier failure(s)
[Pipeline] getContext
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
```

STEP 11: To overcome the above error execute the below commands

```
[sudo] password for kowsika:
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 5.15.167.4-microsoft-standard-WSL2 x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Sat Mar 22 06:33:21 UTC 2025

System load:  0.75          Processes:           57
Usage of /:   0.8% of 1006.85GB   Users logged in:    2
Memory usage: 40%          IPv4 address for eth0: 192.168.236.233
Swap usage:   0%


 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.

   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

This message is shown once a day. To disable it please create the
/root/.hushlogin file.
root@kowsi:~# chmod 777 /var/run/docker.sock
root@kowsi:~# usermod -aG docker jenkins
root@kowsi:~#
```

## HISTORY OF COMMANDS

```
root@kowsi: /home/kow x + v
95 sudo apt update
96 sudo apt install docker.io
97 sudo apt remove -y containerd
98 sudo apt remove -y docker.io
99 sudo apt remove -y containerd.io
100 sudo apt update && sudo apt upgrade -y
101 sudo systemctl start docker
102 sudo systemctl enable docker
103 sudo docker --version
104 sudo systemctl list-units --type=service | grep docker
105 sudo systemctl daemon-reload
106 sudo systemctl restart docker
107 sudo dockerd
108 cd E-Commerce
109 cd /home/kowsika/E-Commerce
110 ls
111 git pull origin main
112 cat Dockerfile
113 docker build -t ecommerce:latest .
114 sudo systemctl start docker
115 pwd
116 sudo su -
117 kubectl create deployment webapp --image=nginx --port=80
118 kubectl cluster-info
119 minikube status
120 minikube start
121 exit
122 apt-get install maven -y
123 java -version
124 mvn clean install
125 sudo apt install openjdk-11-jdk
126 ll
127 cd /home/kowsika/java_project/
128 ll
129 cd ..
130 cd spring-framework/
131 ll
132 apt-get install maven -y
133 sudo apt install openjdk-11-jdk
134 history
135 sudo systemctl enable jenkins
```

 **Jenkins**

🔍 🛡️ 1 👤 Kowsika J 🚪 log out

Dashboard > springframework > #13

Status

</> Changes

📄 Console Output

📝 Edit Build Information

🗑️ Delete build '#13'

🕒 Timings

📦 Git Build Data

🔗 Pipeline Overview

📄 Pipeline Console

🔄 Restart from Stage

✅ #13 (22 Mar 2025, 06:43:42)

🕒 Started by user [Kowsika J](#)

🕒 This run spent:

- 21 ms waiting;
- 1 min 13 sec build duration;
- 1 min 13 sec total from scheduled to completion.

📦 **git**  
**Revision:** d300c699a2ab5bdc22659fafcb2254bea92bb9d1  
**Repository:** [https://github.com/Kowsika/Springframework\\_Devops.git](https://github.com/Kowsika/Springframework_Devops.git)

- refs/remotes/origin/main

</> No changes.

Add description

Keep this build forever

Started 2 hr 0 min ago  
Took **1 min 13 sec**



All Bookmarks



🔍 🔒 1 👤 Kowsika J 🚪 log out

Dashboard > springframework > #13

📄 Status

</> Changes

📄 Console Output

📄 Edit Build Information

🗑 Delete build '#13'

🕒 Timings

📊 Git Build Data

🔗 Pipeline Overview

📄 Pipeline Console

🔄 Restart from Stage

## ✓ Console Output

📄 Download

📄 Copy

View as plain text

```
Started by user Kowsika J
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/springframework@2
[Pipeline] {
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Checkout Code)
[Pipeline] git
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
Cloning repository https://github.com/KowsikaJ/Springframework_Devops.git
```



All Bookmarks

Dashboard > springframework > #13

```
3359bc3d7a6a: Layer already exists
71ed530e3292: Pushed
latest: digest: sha256:cda1596f6d5b15727b889b9510c38737f135310e5127686fdfaeb45fbbc1b7d1 size: 2413
[Pipeline] }
[Pipeline] // script
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Declarative: Post Actions)
[Pipeline] echo
Pipeline executed successfully!
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```