AIR UNIVERSITY, ISLAMABAD

Department of Cyber Security

Secure Software Design & Development Lab (CY- 256L)

Assignment-3

Submitted to: Sir Mahaz Khan

Submitted by: Mirza Shaheer Baig (231330)

CLASS: BSCYS-4 A

Submitted on: **13/06/2025**

Docker:

What is Docker?

Docker is an open-source platform that allows you to develop, ship, and run applications inside lightweight, portable containers.

- A Docker container is a standardized unit that packages your code, runtime, system tools, libraries, and settings—everything the app needs to run.
- Containers are isolated from each other and the host system, but they share the OS kernel, making them faster and more efficient than virtual machines (VMs).

How Docker Works

1. **Dockerfile**: You write a file that defines what goes into the container (like a recipe).
2. **Image**: Docker builds an image from the Dockerfile.
3. **Container**: You run an image to create a container (an instance of the image).

Uses of Docker

1. **Development Consistency**
    - Developers can run the same code on any machine without worrying about dependencies or environment setup issues.
    - "It works on my machine" problems are eliminated.
2. **Microservices Architecture**
    - Each microservice can run in its own container with its own dependencies.
    - Easy to scale and manage.
3. **CI/CD Pipelines**
    - Integrates seamlessly into Continuous Integration/Continuous Deployment workflows.
    - Automated testing, building, and deployment are simplified.
4. **Environment Replication**
    - Easily replicate production-like environments for testing or staging.
5. **Isolation and Security**
    - Containers isolate applications from each other and from the host system.
6. **Portability**

- Docker containers can run on any system that supports Docker: Linux, Windows, cloud providers (AWS, Azure, GCP), and even your laptop.

7. **Resource Efficiency**
   - Containers are more lightweight than VMs and use fewer system resource

Docker Setup:

Step 1: Install Prerequisites

First, let's make sure your system is up to date and install required dependencies:

```
Hit:1 https://packages.wazuh.com/4.x/apt stable InRelease
Get:2 https://download.docker.com/linux/ubuntu noble InRelease [48.8 kB]
Hit:3 http://archive.ubuntu.com/ubuntu noble InRelease
Get:4 https://download.docker.com/linux/ubuntu noble/stable amd64 Packages [24.0
 kB]
Hit:5 http://archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:6 http://security.ubuntu.com/ubuntu noble-security InRelease
Hit:7 http://archive.ubuntu.com/ubuntu noble-backports InRelease
Fetched 72.8 kB in 1s (52.2 kB/s)
Reading package lists... Done
```

Step 2: Add Docker's Official GPG Key

Now let's install Docker and related components:

```
Setting up docker-compose-plugin (2.35.1-1~ubuntu.24.04~noble) ...
Setting up docker-ce-cli (5:28.1.1-1~ubuntu.24.04~noble) ...
Setting up libslirp0:amd64 (4.7.0-1ubuntu3) ...
Setting up pigz (2.8-1) ...
Setting up git-man (1:2.43.0-1ubuntu7.2) ...
Setting up docker-ce-rootless-extras (5:28.1.1-1~ubuntu.24.04~noble) ...
Setting up slirp4netns (1.2.1-1build2) ...
Setting up docker-ce (5:28.1.1-1~ubuntu.24.04~noble) ...
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /us
r/lib/systemd/system/docker.service.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /usr/li
b/systemd/system/docker.socket.
Setting up git (1:2.43.0-1ubuntu7.2) ...
Processing triggers for man-db (2.12.0-4build2) ...
Processing triggers for libc-bin (2.39-0ubuntu8.4) ...
root@Ubuntu-Vbox:/home/ubuntu#
```

Step 3: Verify Docker Installation

Let's make sure Docker is installed correctly by running the hello-world image:

```
root@Ubuntu-Vbox:/home/ubuntu# sudo docker run hello-world

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.
```

Step 4: Working with Busybox

```
root@Ubuntu-Vbox:/home/ubuntu# docker pull busybox
Using default tag: latest
latest: Pulling from library/busybox
Digest: sha256:3308bdfbc80b8e960219232df14f233a3c56979f392f56b0d9a8bc290c7dfd76
Status: Image is up to date for busybox:latest
docker.io/library/busybox:latest
root@Ubuntu-Vbox:/home/ubuntu# docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
hello-world   latest    74cc54e27dc4   4 months ago   10.1kB
busybox       latest    6d3e4188a38a   7 months ago   4.28MB
root@Ubuntu-Vbox:/home/ubuntu#
```

```
root@Ubuntu-Vbox:/home/ubuntu# docker run busybox echo "hello from busybox"
hello from busybox
root@Ubuntu-Vbox:/home/ubuntu#
```

```
root@Ubuntu-Vbox:/home/ubuntu# docker run -it busybox sh
/ # ls
bin    etc    lib    proc   sys    usr
dev    home   lib64  root   tmp    var
/ # uptime
 04:20:25 up 19 min,  0 users,  load average: 0.37, 0.45, 0.81
/ #
```

Step 5: Exiting Docker:

```
root@Ubuntu-Vbox:/home/ubuntu# docker ps -a
CONTAINER ID   IMAGE         COMMAND                  CREATED         STATUS
                 PORTS     NAMES
3e0b28b8e1be   busybox       "sh"                     2 minutes ago   Exited (0)
 7 seconds ago             elated_jones
bd2ffe1bef8e   busybox       "echo 'hello from bu…"   2 minutes ago   Exited (0)
 2 minutes ago             inspiring_hodgkin
6e8e3fff292e   busybox       "sh"                     5 minutes ago   Exited (0)
 4 minutes ago             vigorous_noyce
fadbe7777371   busybox       "echo 'hello from bu…"   5 minutes ago   Exited (0)
 5 minutes ago             crazy_archimedes
e32d85d37528   hello-world   "/hello"                 7 minutes ago   Exited (0)
 7 minutes ago             bold_mclaren
d69f93f46519   hello-world   "/hello"                 7 minutes ago   Exited (0)
 7 minutes ago             happy_bose
root@Ubuntu-Vbox:/home/ubuntu#
```

Step 6: Deleting a container:

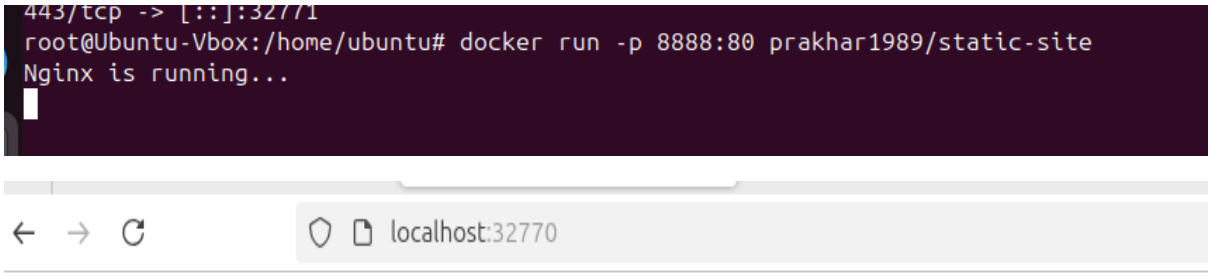
```
root@Ubuntu-Vbox:/home/ubuntu# docker ps -a
CONTAINER ID   IMAGE         COMMAND                  CREATED          STATUS
3e0b28b8e1be   busybox       "sh"                     6 minutes ago    Exited (0) 4 minut
s
bd2ffe1bef8e   busybox       "echo 'hello from bu…"   6 minutes ago    Exited (0) 6 minut
odgkin
6e8e3fff292e   busybox       "sh"                     9 minutes ago    Exited (0) 8 minut
yce
fadbe7777371   busybox       "echo 'hello from bu…"   9 minutes ago    Exited (0) 9 minut
medes
e32d85d37528   hello-world   "/hello"                 11 minutes ago   Exited (0) 11 minu
n
root@Ubuntu-Vbox:/home/ubuntu# docker rm 3e0b28b8e1be
3e0b28b8e1be
root@Ubuntu-Vbox:/home/ubuntu#
```

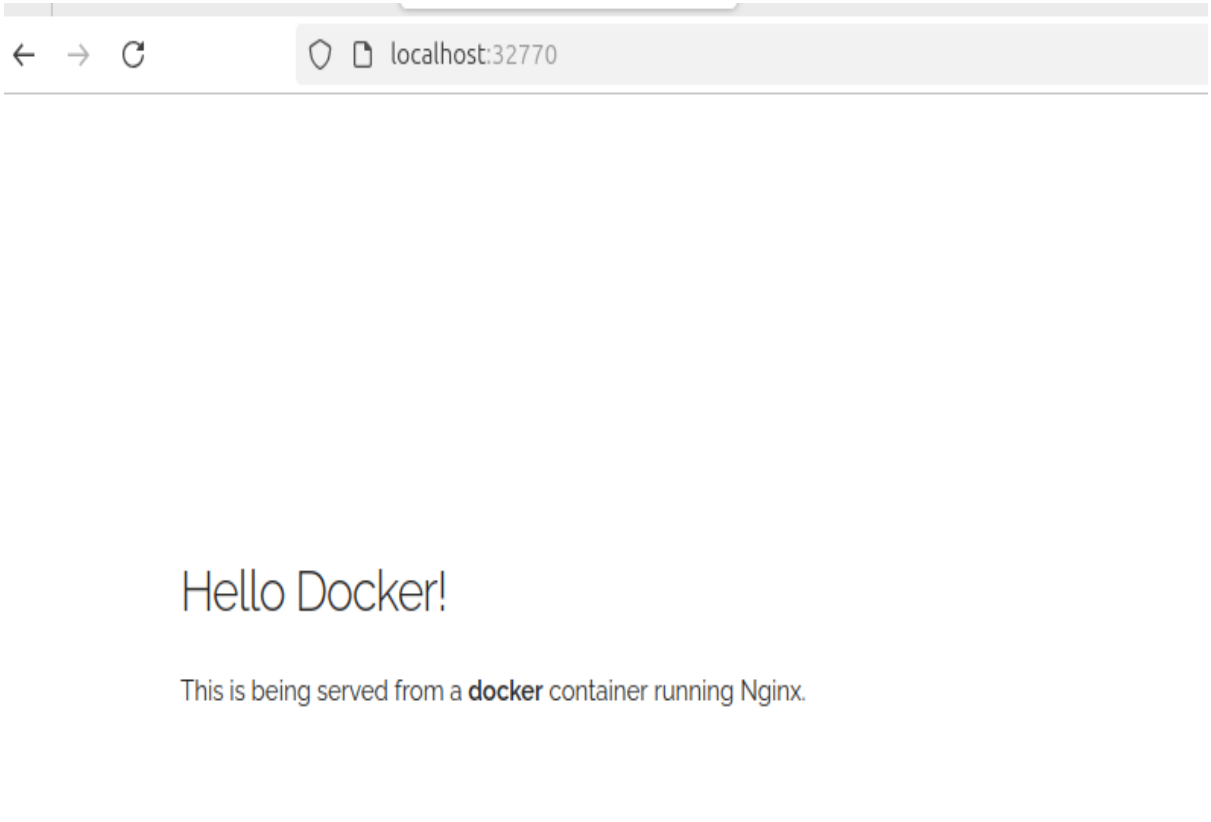
Step 7: Running a Web Application

Let's try running a simple web application in a container:

```
root@Ubuntu-Vbox:/home/ubuntu# docker run -d -p 8080:80 --name my-site nginx
Unable to find image 'nginx:latest' locally
latest: Pulling from library/nginx
61320b01ae5e: Pull complete
670a101d432b: Pull complete
405bd2df85b6: Pull complete
cc80efff8457: Pull complete
2b9310b2ee4b: Pull complete
6c4aa022e8e1: Pull complete
abddc69cb49d: Pull complete
Digest: sha256:fb39280b7b9eba5727c884a3c7810002e69e8f961cc373b89c92f14961d903a0
Status: Downloaded newer image for nginx:latest
66cbab800dfa34eef79a6f8924e8731384c4bd51f13565b1276d3ac00bd69cd7
root@Ubuntu-Vbox:/home/ubuntu#
```

We can also specify a custom port to which the client will forward connections to the container.





Stopping my website and deleting the docker container

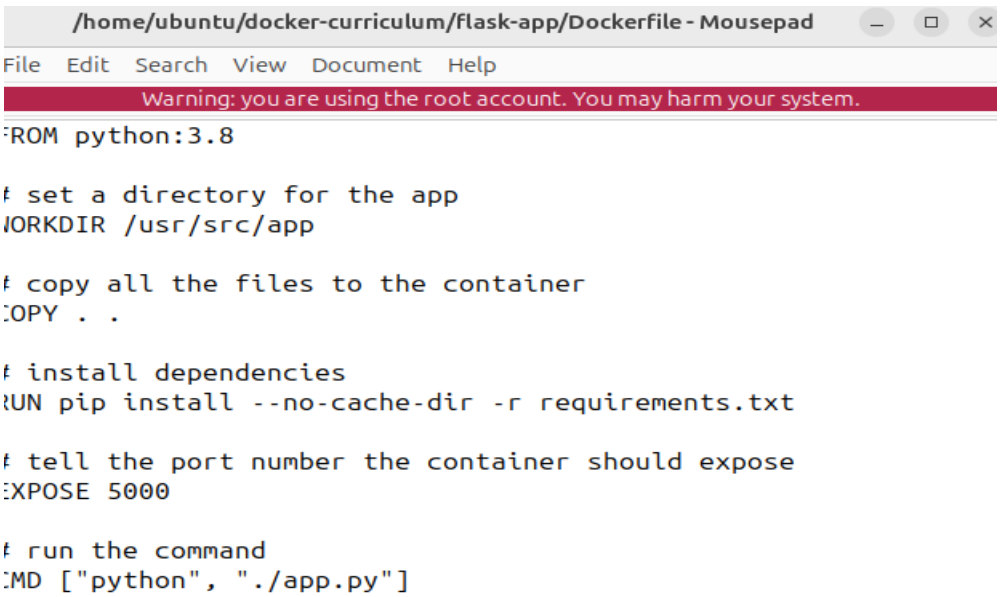
```
root@Ubuntu-Vbox:/home/ubuntu# docker stop static-site
static-site
root@Ubuntu-Vbox:/home/ubuntu#
```

```
root@Ubuntu-Vbox:/home/ubuntu# docker ps -a
CONTAINER ID   IMAGE                     COMMAND                  CREATED          STATUS
     NAMES
0742006d3ca4   prakhar1989/static-site   "./wrapper.sh"           3 minutes ago    Exited
     fervent_panini
116e39dfd6cf   prakhar1989/static-site   "./wrapper.sh"           9 minutes ago    Exited
     static-site
bd2ffe1bef8e   busybox                   "echo 'hello from bu…"   33 minutes ago   Exited
     inspiring_hodgkin
6e8e3fff292e   busybox                   "sh"                     35 minutes ago   Exited
     vigorous_noyce
fadbe7777371   busybox                   "echo 'hello from bu…"   35 minutes ago   Exited
     crazy_archimedes
e32d85d37528   hello-world               "/hello"                 37 minutes ago   Exited
     bold_mclaren
root@Ubuntu-Vbox:/home/ubuntu# docker rm 0742006d3ca4
0742006d3ca4
root@Ubuntu-Vbox:/home/ubuntu#
```

Step 8: Creating Your Own Image

```
root@Ubuntu-Vbox:/home/ubuntu# git clone https://github.com/prakhar1989/docker-curriculum.git
Cloning into 'docker-curriculum'...
remote: Enumerating objects: 1737, done.
remote: Counting objects: 100% (71/71), done.
remote: Compressing objects: 100% (54/54), done.
remote: Total 1737 (delta 57), reused 17 (delta 17), pack-reused 1666 (from 3)
Receiving objects: 100% (1737/1737), 9.13 MiB | 90.00 KiB/s, done.
Resolving deltas: 100% (961/961), done.
root@Ubuntu-Vbox:/home/ubuntu# cd docker-curriculum/flask-app
```

With that, our Docker file is now ready. This is how it looks



Now that we have our Dockerfile, we can build our image. The docker build command does the heavy-lifting of creating a Docker image from a Dockerfile.

```
FROM python:3.8

RUN pip install flask

COPY . /opt/

EXPOSE 5000

WORKDIR /opt

ENTRYPOINT ["python"]
CMD ["app.py"]
```

Step 9: Building Image:

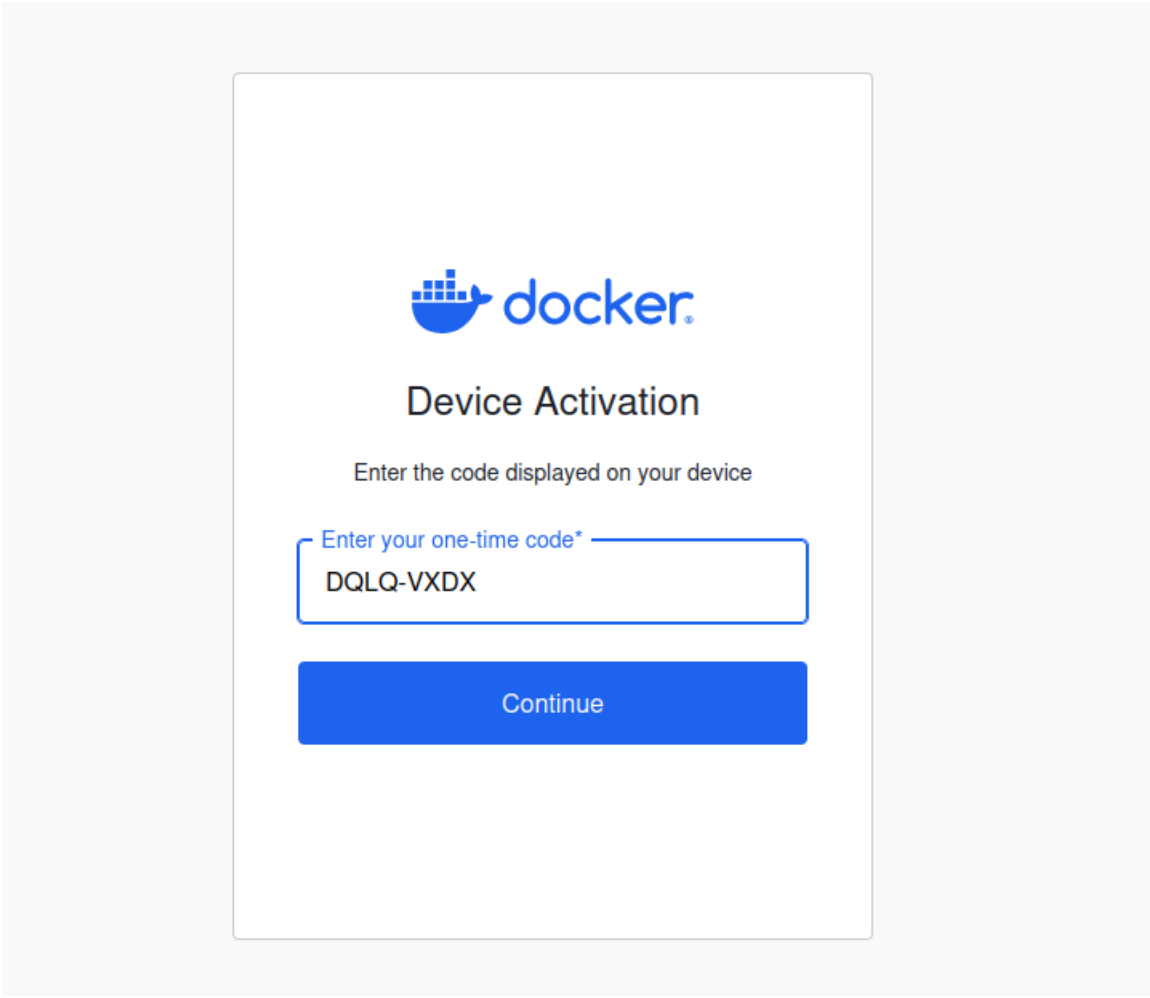
```
root@Ubuntu-Vbox:/home/ubuntu/docker-curriculum/flask-app# docker build -t root/catnip .
[+] Building 190.7s (9/9) FINISHED                                                                                doc
 => [internal] load build definition from Dockerfile
 => => transferring dockerfile: 154B
 => [internal] load metadata for docker.io/library/python:3.8
 => [internal] load .dockerignore
 => => transferring context: 2B
 => [1/4] FROM docker.io/library/python:3.8@sha256:d411270700143fa2683cc8264d9fa5d3279fd3b6afff62ae81ea2f9d0
 => => resolve docker.io/library/python:3.8@sha256:d411270700143fa2683cc8264d9fa5d3279fd3b6afff62ae81ea2f9d0
 => => sha256:3ea6eaad4f175bd42f39dae10098b1820ee522628ff04670b3ac8b89fe00c1c8 5.97kB / 5.97kB
 => => sha256:d411270700143fa2683cc8264d9fa5d3279fd3b6afff62ae81ea2f9d070e390c 10.35kB / 10.35kB
 => => sha256:7aa279fb41dad2962d3c915aa6f6615134baa412ab5aafa9d4384dcaaa0af15d 2.32kB / 2.32kB
 => => sha256:cdd62bf39133c498a16f7a7b1b6555ba43d02b2511c508fa4c0a9b1975ffe20e 49.56MB / 49.56MB
 => => sha256:a173f2aee8e962ea19db1e418ae84a0c9f71480b51f768a19332dfa83d7722a5 64.39MB / 64.39MB
 => => sha256:a47cff7f31e941e78bf63ca19f0811b675283e2c00ddea10c57f78d93b2bc343 24.05MB / 24.05MB
 => => sha256:01272fe8adbacc44afd2b92994b31c40a151f4324ca392050d9e8d580927dd32 211.27MB / 211.27MB
 => => sha256:cddc73e4e6c704bfa2325e53c32ddb3553c8fc3a91dab6c092bb353f82098b09 6.16MB / 6.16MB
 => => extracting sha256:cdd62bf39133c498a16f7a7b1b6555ba43d02b2511c508fa4c0a9b1975ffe20e
 => => extracting sha256:a47cff7f31e941e78bf63ca19f0811b675283e2c00ddea10c57f78d93b2bc343
 => => extracting sha256:a173f2aee8e962ea19db1e418ae84a0c9f71480b51f768a19332dfa83d7722a5
 => => sha256:cc48f13b5f0f44b2e298de83a94a99fe7abdfb3335fe9b7811b8f764abb1a4ac 18.06MB / 18.06MB
 => => sha256:5a98c896c047f960c5fd29d44fa778899a68e7ebfb6a6a4f2a3fbf7baa902f6a 249B / 249B
 => => extracting sha256:01272fe8adbacc44afd2b92994b31c40a151f4324ca392050d9e8d580927dd32
 => => extracting sha256:cddc73e4e6c704bfa2325e53c32ddb3553c8fc3a91dab6c092bb353f82098b09
 => => extracting sha256:cc48f13b5f0f44b2e298de83a94a99fe7abdfb3335fe9b7811b8f764abb1a4ac
 => => extracting sha256:5a98c896c047f960c5fd29d44fa778899a68e7ebfb6a6a4f2a3fbf7baa902f6a
 => [internal] load build context
 => => transferring context: 3.23kB
 => [2/4] RUN pip install flask
 => [3/4] COPY . /opt/
 => [4/4] WORKDIR /opt
```

Step 10: Running Image:

```
root@Ubuntu-Vbox:/home/ubuntu/docker-curriculum/flask-app# docker run -p 8888:5000 root/catnip
 * Serving Flask app 'app'
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI serv
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:5000
 * Running on http://172.17.0.2:5000
Press CTRL+C to quit
```

Docker push

```
ubuntu@Ubuntu-Vbox:~$ sudo su
[sudo] password for ubuntu:
root@Ubuntu-Vbox:/home/ubuntu# docker login

USING WEB-BASED LOGIN

i Info → To sign in with credentials on the command line, use 'docker login -u <username>'


Your one-time device confirmation code is: DQLQ-VXDX
Press ENTER to open your browser or submit your device code here: https://login.docker.com/activate

Waiting for authentication in the browser…
```



Security Checks

- Run the commands given in Rule 2, Rule 3, Rule 8 to see if they work fine.

Check that containers can't access host processes (Rule 2)

```
root@Ubuntu-Vbox:/home/ubuntu/docker-curriculum/flask-app# docker run --rm ubuntu ps aux
USER         PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root           1  7.1  0.0   7888  3776 ?        Rs   05:22   0:00 ps aux
root@Ubuntu-Vbox:/home/ubuntu/docker-curriculum/flask-app#
```

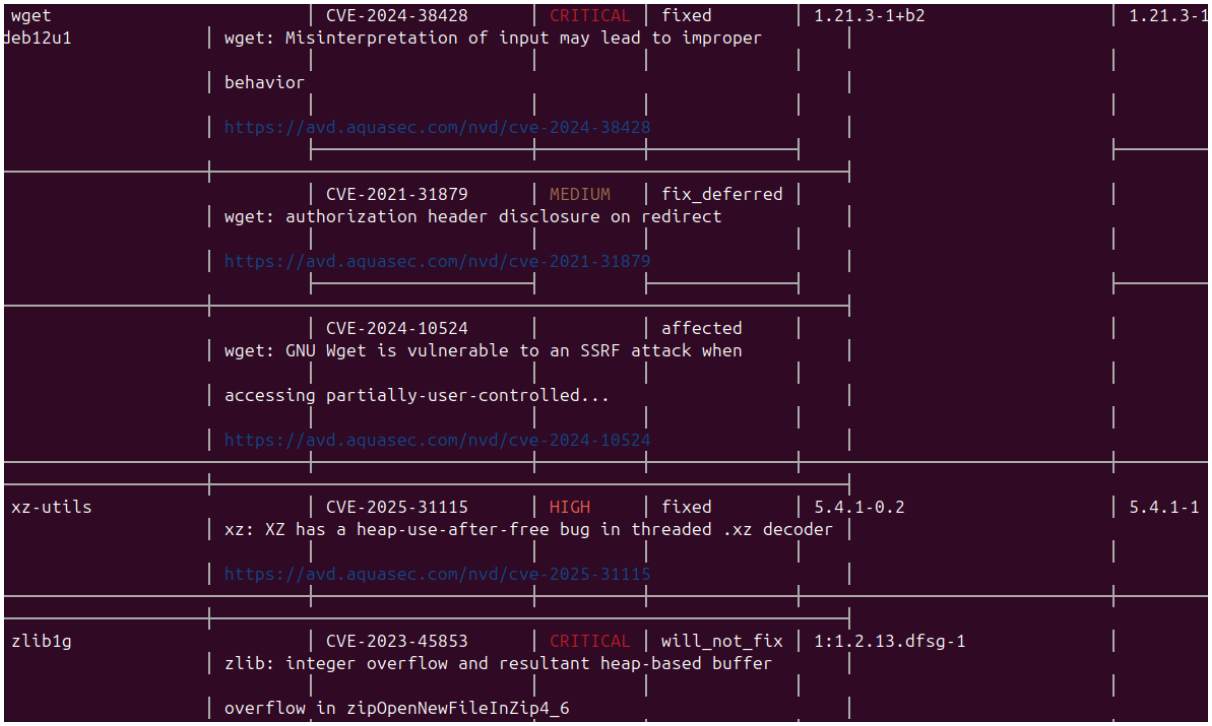
Check user namespace isolation (Rule 3)

```
root@Ubuntu-Vbox:/home/ubuntu/docker-curriculum/flask-app# docker run --rm ubuntu cat /proc/self/uid_map
         0          0 4294967295
root@Ubuntu-Vbox:/home/ubuntu/docker-curriculum/flask-app#
```

Check for container resource limits (Rule 8)

- In Rule 9 tools are suggested to detect containers with known vulnerabilities - scan images.
- Use any of these free tools (preferably Trivy) to test the image you have created above in the task or any other docker image.

```
uration injectable in repo revision |
|                   |              |        |       |                |            | when installing via p
ip                                  |
|                   |              |        |       |                |            | https://avd.aquasec.c
om/nvd/cve-2023-5752                |
├───────────────────┼──────────────┼────────┤       ├────────────────┼────────────┼──────────────────────
────────────────────────────────────┤
| setuptools (METADATA) | CVE-2022-40897 | HIGH |   | 57.5.0         | 65.5.1     | pypa-setuptools: Regu
lar Expression Denial of Service    |
|                   |              |        |       |                |            | (ReDoS) in package_in
dex.py                              |
|                   |              |        |       |                |            | https://avd.aquasec.c
om/nvd/cve-2022-40897               |
|                   ├──────────────┤        |       |                ├────────────┼──────────────────────
────────────────────────────────────┤
|                   | CVE-2024-6345 |       |       |                | 70.0.0     | pypa/setuptools: Remo
te code execution via download      |
|                   |              |        |       |                |            | functions in the pack
age_index module in...              |
|                   |              |        |       |                |            | https://avd.aquasec.c
om/nvd/cve-2024-6345                |
|                   ├──────────────┤        |       |                ├────────────┼──────────────────────
────────────────────────────────────┤
|                   | CVE-2025-47273 |      |       |                | 78.1.1     | setuptools: Path Trav
ersal Vulnerability in setuptools   |
|                   |              |        |       |                |            | PackageIndex
|                   |              |        |       |                |            | https://avd.aquasec.c
om/nvd/cve-2025-47273               |
└───────────────────┴──────────────┴────────┴───────┴────────────────┴────────────┴──────────────────────
────────────────────────────────────┘
```



What is Jenkins?

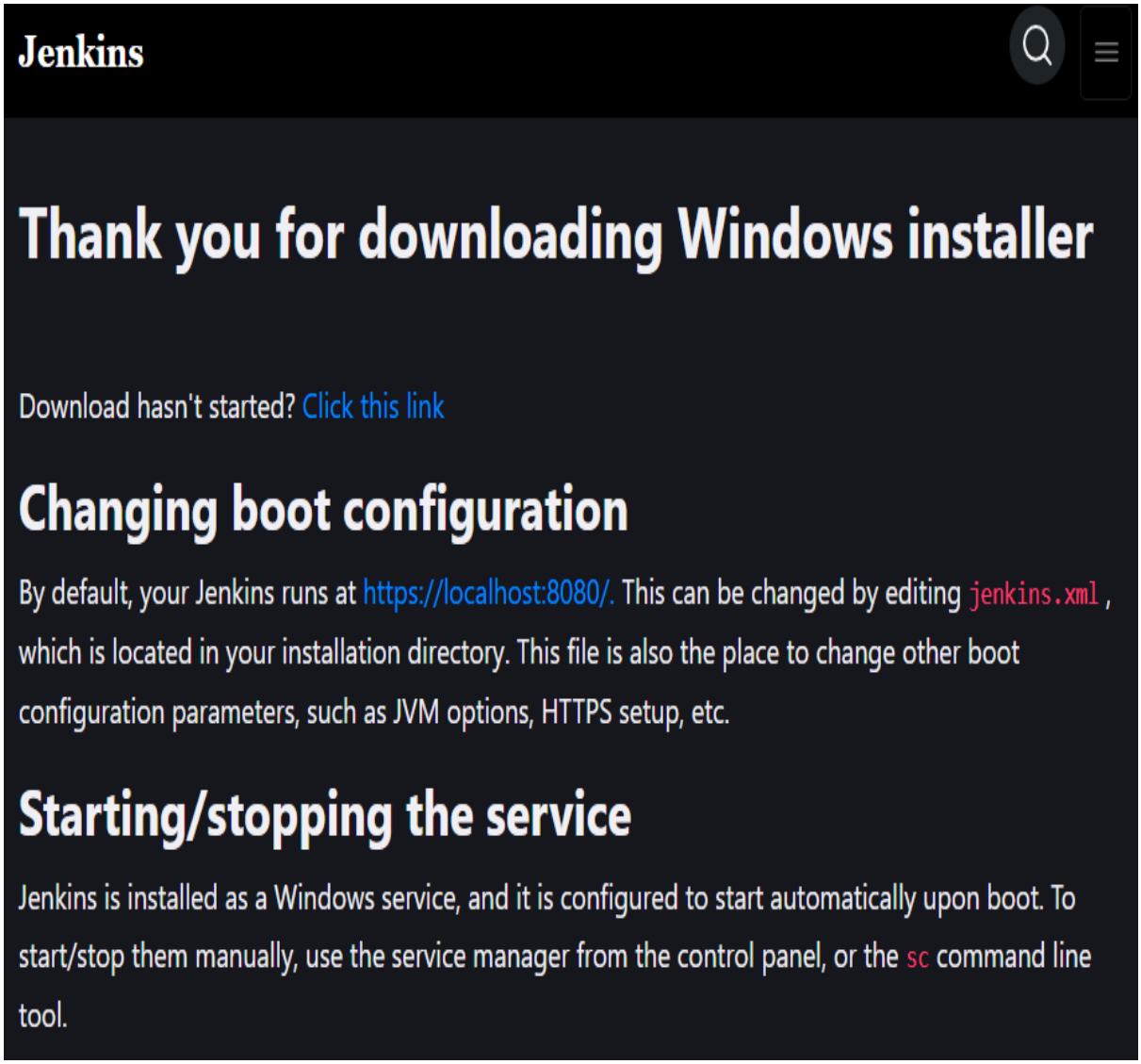
Jenkins is a powerful application that allows continuous integration and continuous delivery of projects, regardless of the platform you are working on. It is a free source that can handle any kind of build or continuous integration. You can integrate Jenkins with a number of testing and deployment

Pre requisites:

Java should be installed on your machine.

```
C:\Users\Shaheer Baig>java -version
openjdk version "21.0.7" 2025-04-15 LTS
OpenJDK Runtime Environment Temurin-21.0.7+6 (build 21.0.7+6-LTS)
OpenJDK 64-Bit Server VM Temurin-21.0.7+6 (build 21.0.7+6-LTS, mixed mode, sharing)
```

Download Jenkins:



Starting Jenkins

Open the command prompt. From the command prompt, browse to the directory where the jenkins.war file is present. Run the following command

D:\ –jar Jenkins.war

```
D:\Downloads>Java -jar jenkins.war
Running from: D:\Downloads\jenkins.war
webroot: C:\Users\Shaheer Baig\.jenkins\war
2025-05-30 04:49:05.384+0000 [id=1]     INFO    winstone.Logger#logInternal:
 Beginning extraction from war file
2025-05-30 04:49:06.595+0000 [id=1]     WARNING o.e.j.ee9.nested.ContextHand
ler#setContextPath: Empty contextPath
2025-05-30 04:49:06.657+0000 [id=1]     INFO    org.eclipse.jetty.server.Ser
ver#doStart: jetty-12.0.21; built: 2025-05-09T00:32:00.688Z; git: 1c4719601e
31b05b7d68910d2edd980259f1f53c; jvm 21.0.7+6-LTS
2025-05-30 04:49:12.723+0000 [id=1]     INFO    o.e.j.e.w.StandardDescriptor
Processor#visitServlet: NO JSP Support for /, did not find org.eclipse.jetty
.ee9.jsp.JettyJspServlet
2025-05-30 04:49:12.784+0000 [id=1]     INFO    o.e.j.s.DefaultSessionIdMana
ger#doStart: Session workerName=node0
2025-05-30 04:49:13.256+0000 [id=1]     INFO    hudson.WebAppMain#contextIni
tialized: Jenkins home directory: C:\Users\Shaheer Baig\.jenkins found at: $
user.home/.jenkins
2025-05-30 04:49:13.363+0000 [id=1]     INFO    o.e.j.s.handler.ContextHandl
er#doStart: Started oeje9n.ContextHandler$CoreContextHandler@3051e0b2{Jenkin
s v2.512,/,b=file:///C:/Users/Shaheer%20Baig/.jenkins/war/,a=AVAILABLE,h=oej
e9n.ContextHandler$CoreContextHandler$CoreToNestedHandler@1f52eb6f{STARTED}}
2025-05-30 04:49:13.413+0000 [id=1]     INFO    o.e.j.server.AbstractConnect
or#doStart: Started ServerConnector@6f19d20d{HTTP/1.1, (http/1.1)}{0.0.0.0:8
080}
```

After the command is run, various tasks will run, one of which is the extraction of the war file which is done by an embedded webserver called winstone.

Once the processing is complete without major errors, the following line will come in the output of the command prompt.

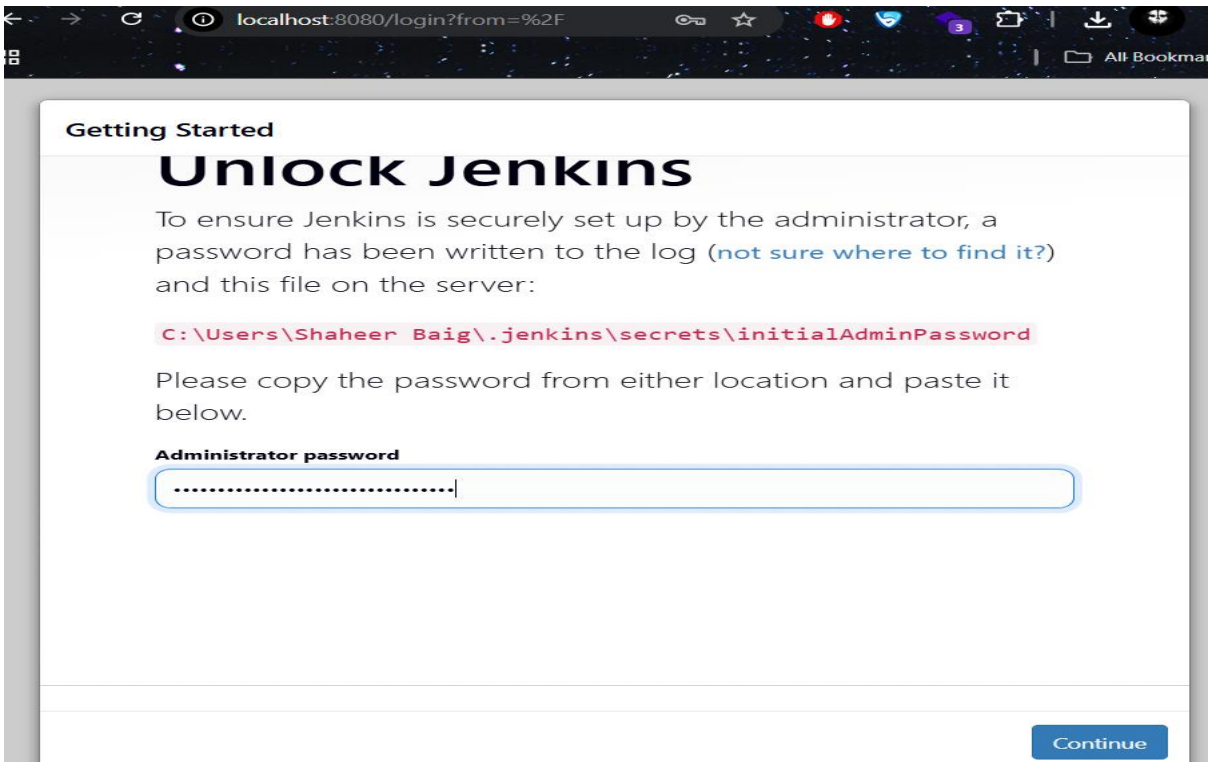
Copy the password and save it somewhere for later use.

Accessing Jenkins

Once Jenkins is up and running, one can access Jenkins from the

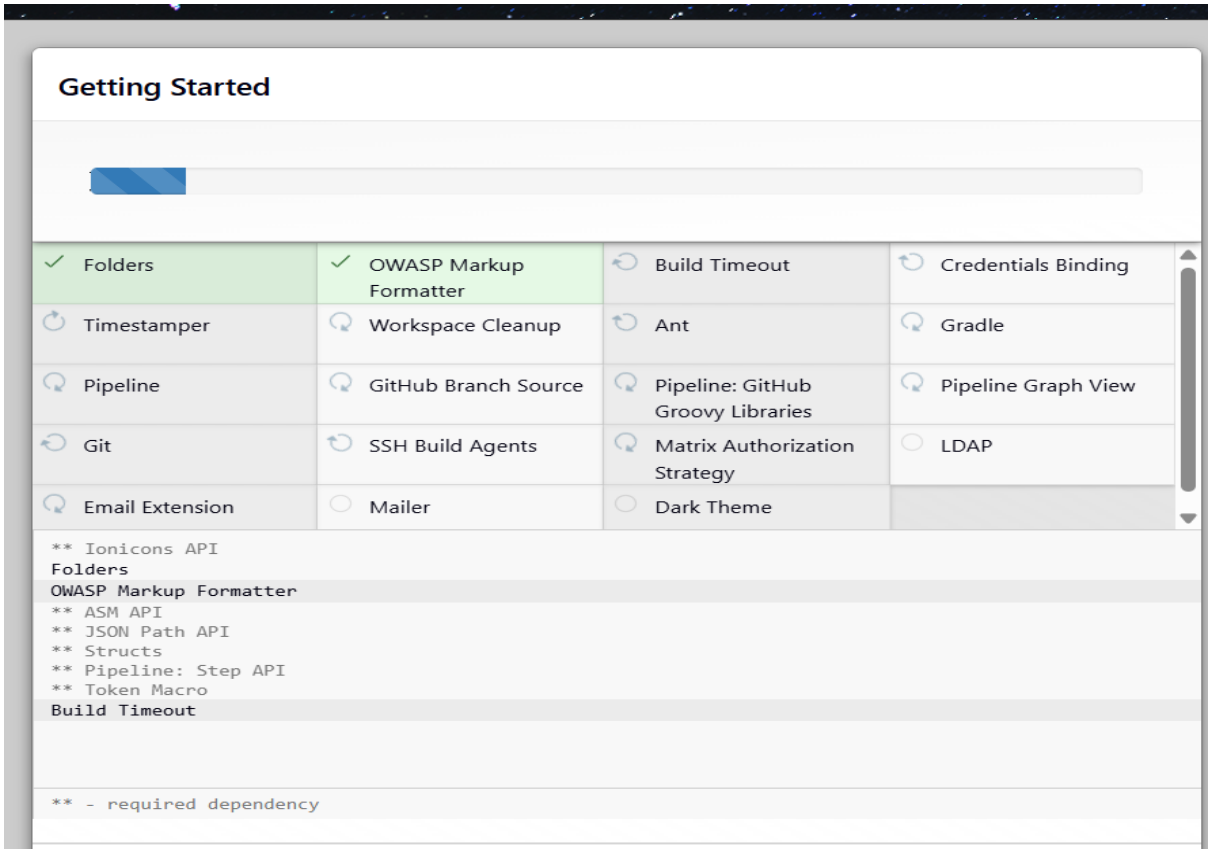
link − http://localhost:8080

This link will bring up the Jenkins dashboard. Write password that you saved previously.



Select and install suggested plugins.

Submitted to: Sir Mahaz Khan



Username

Thunder_31

User name must only contain alphanumeric characters, underscore and dash

Password

••••••••••

Confirm password

••••••••••

Full name

Mirza Shaheer Baig

E-mail address

231330@students.au.edu.pk

Submitted to: Sir Mahaz Khan

Submitted to: Sir Mahaz Khan

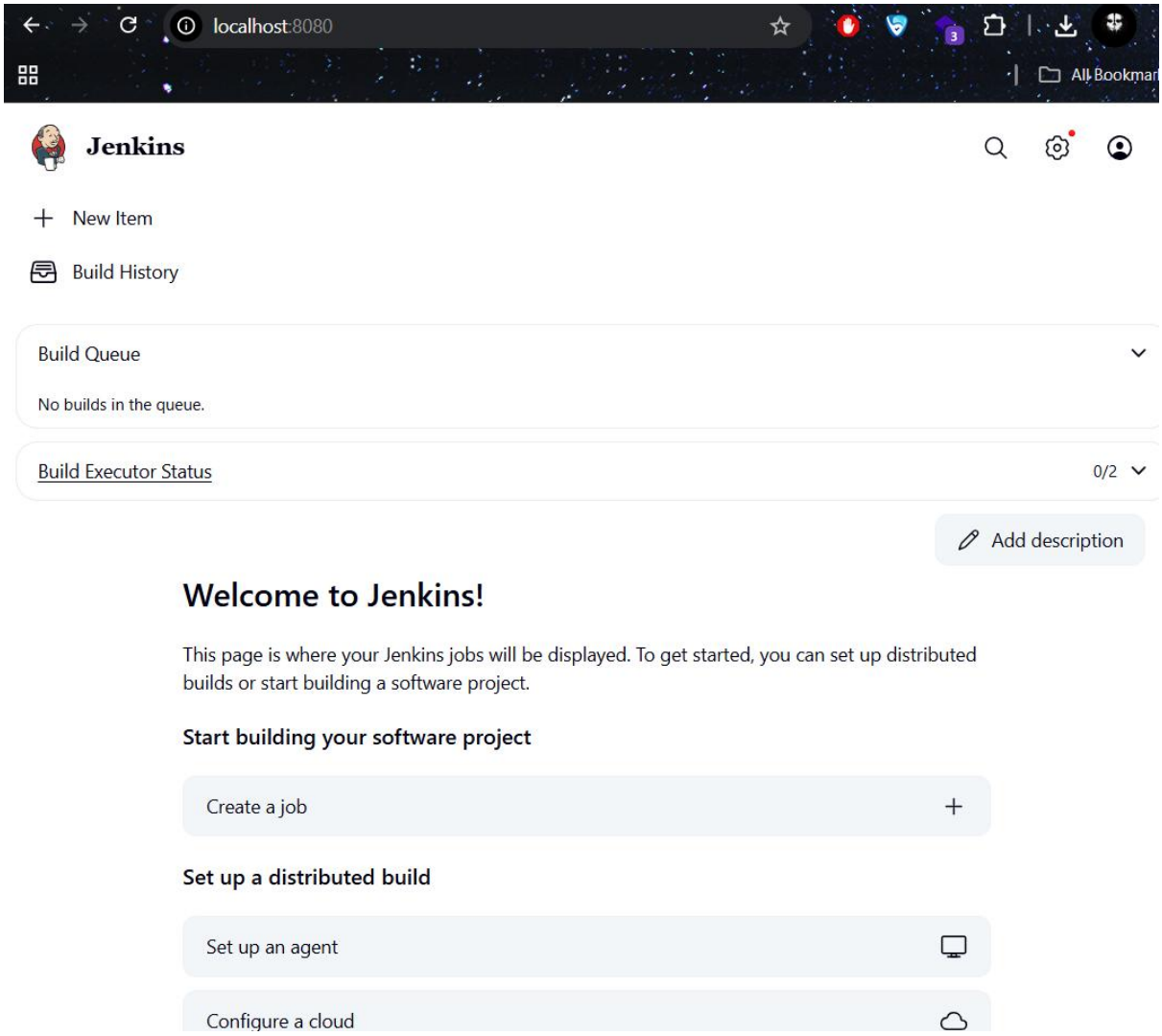
Getting Started

Jenkins is ready!

Your Jenkins setup is complete.

Start using Jenkins

Jenkins 2.512

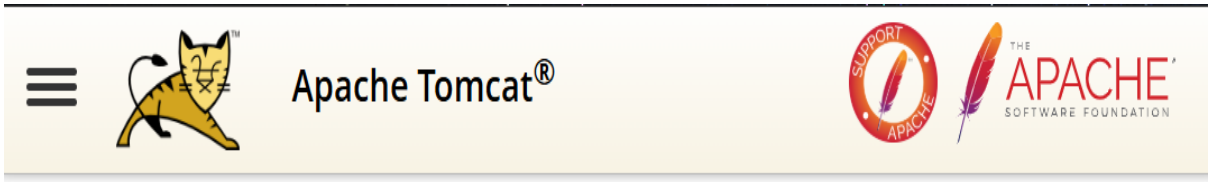


What is Tomcat?

Tomcat is an open-source web server and servlet. The Apache Software Foundation has developed it. It is used widely for hosting Java-based applications on the web. It is built on Java technologies and implements the Java Servlet and JavaServer Pages (JSP) specifications. Tomcat acts as a bridge between web servers and Java-based applications, facilitating the execution of dynamic content and processing client requests.

Download Tomcat

The official website for tomcat is Tomcat. If you click the given link, you can get the home page of the tomcat official website as shown below.



Apache Tomcat

The Apache Tomcat® software is an open source implementation of the Jakarta Servlet, Jakarta Pages, Jakarta Expression Language, Jakarta WebSocket, Jakarta Annotations and Jakarta Authentication specifications. These specifications are part of the Jakarta EE platform.

The Jakarta EE platform is the evolution of the Java EE platform. Tomcat 10 and later implement specifications developed as part of Jakarta EE. Tomcat 9 and earlier implement specifications developed as part of Java EE.

The Apache Tomcat software is developed in an open and participatory environment and released under the Apache License version 2. The Apache Tomcat project is intended to be a collaboration of the best-of-breed developers from around the world. We invite you to participate in this open development project. To learn more about getting involved, click here.

Apache Tomcat software powers numerous large-scale, mission-critical web applications across a diverse range of industries and organizations. Some of these users and their stories are listed on the PoweredBy wiki page.

Apache Tomcat, Tomcat, Apache, the Apache feather, and the Apache Tomcat project logo are trademarks of the Apache Software Foundation.

Tomcat Native 2.0.9 Released 2024-05-29

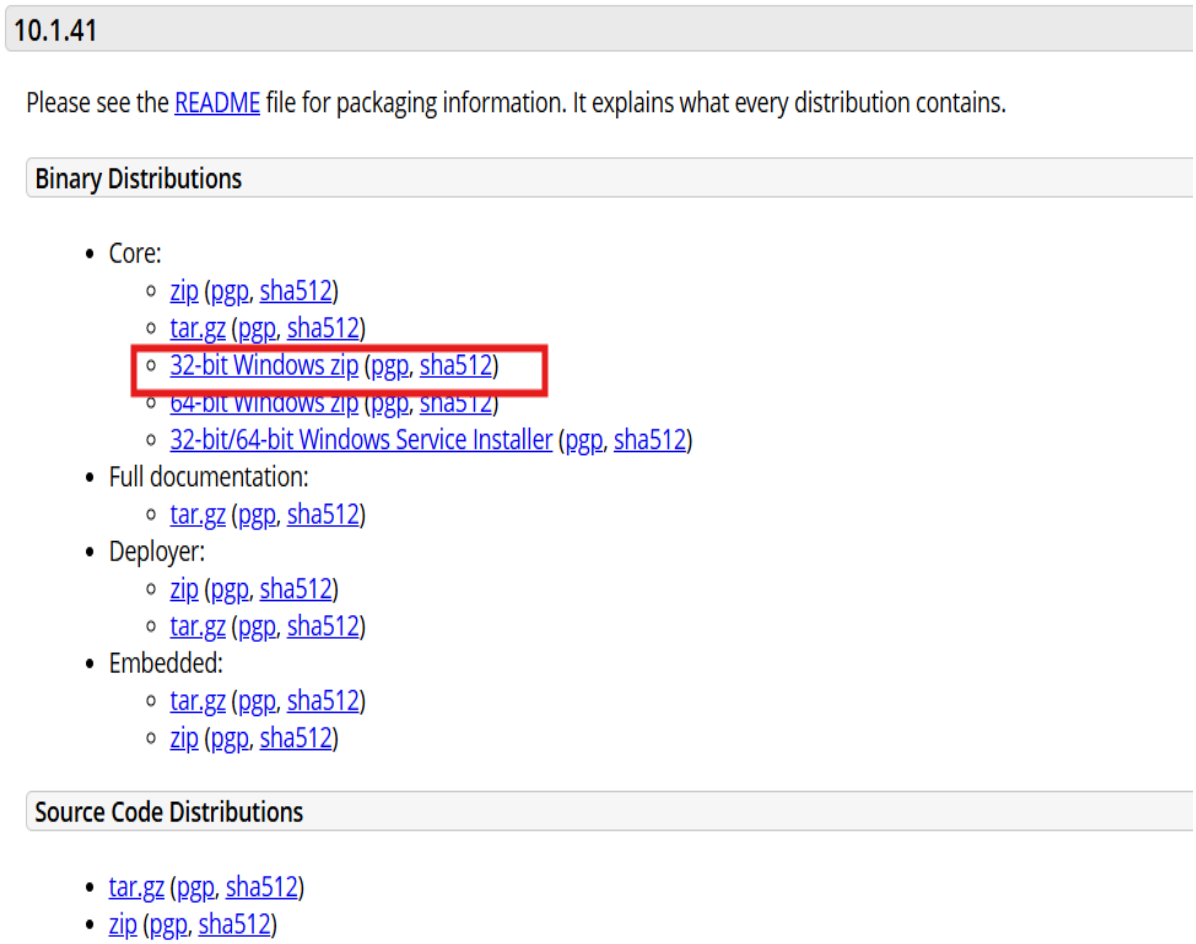
The Apache Tomcat Project is proud to announce the release of version 2.0.9 of Tomcat Native. The notable changes compared to 2.0.8 include:

- Update the Windows build environment to use Visual Studio 2022.
- The windows binaries in this release have been built with OpenSSL 3.5.0 and APR 1.7.6

Download | Change log for 2.0.9

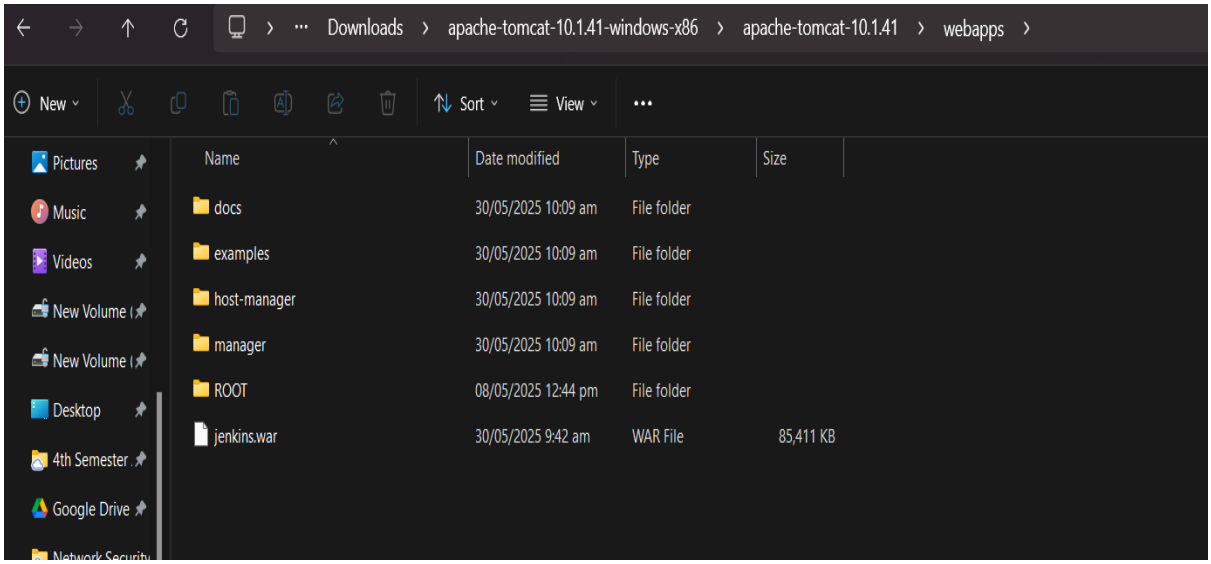
Tomcat 11.0.7 Released 2025-05-13

Browse to the link https://tomcat.apache.org/download-70.cgi to get the download for tomcat.



Go to the ‘Binary Distributions’ section. Download the 32-bit Windows zip file.

Then unzip the contents of the downloaded zip file.



Jenkins and Tomcat Setup

Copy the Jenkis.war file which was downloaded from the previous section and copy it to the webapps folder in the tomcat folder.

Now open the command prompt. From the command prompt, browse to the directory where the tomcat7 folder is location. Browse to the bin directory in this folder and run the start.bat file

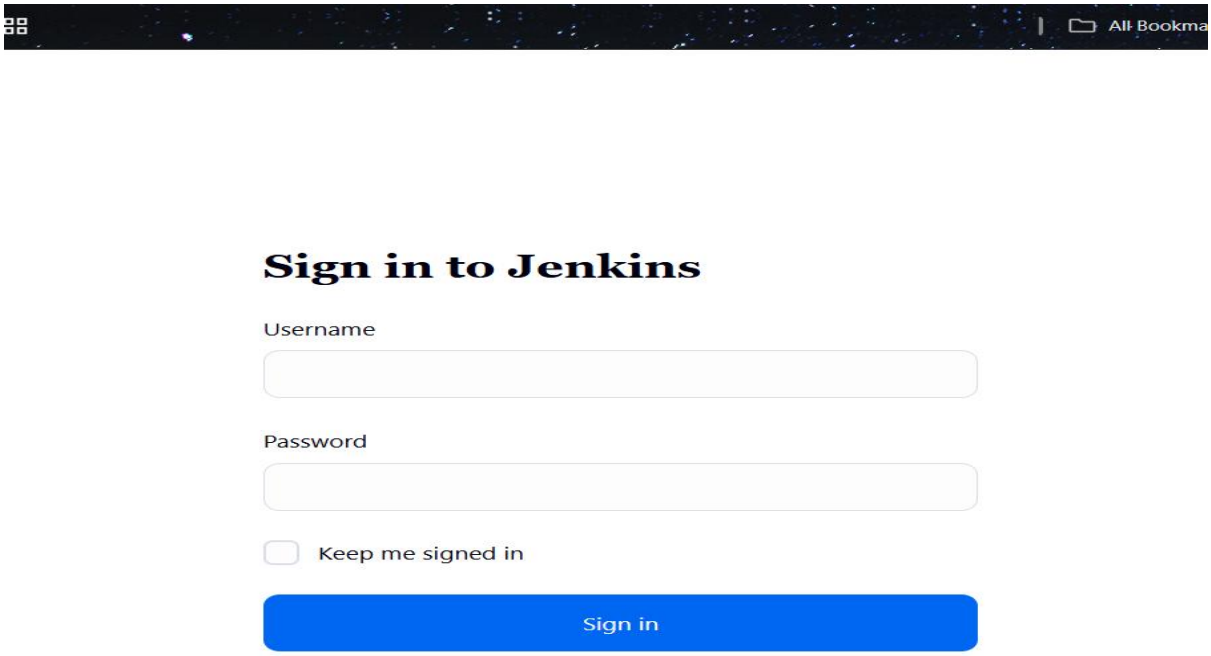
E:\Apps\tomcat7\bin>startup.bat

```
08/05/2025  12:44 pm    <DIR>          work
               7 File(s)        122,106 bytes
               9 Dir(s)  269,982,982,144 bytes free

D:\Downloads\apache-tomcat-10.1.41-windows-x86\apache-tomcat-10.1.41>cd bin

D:\Downloads\apache-tomcat-10.1.41-windows-x86\apache-tomcat-10.1.41\bin>sta
rtup.bat
```

```
Tomcat
30-May-2025 10:23:03.667 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web applicatio
n directory [D:\Downloads\apache-tomcat-10.1.41-windows-x86\apache-tomcat-10.1.41\webapps\host-manager] has finished in
[72] ms
30-May-2025 10:23:03.674 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application di
rectory [D:\Downloads\apache-tomcat-10.1.41-windows-x86\apache-tomcat-10.1.41\webapps\manager]
30-May-2025 10:23:03.753 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web applicatio
n directory [D:\Downloads\apache-tomcat-10.1.41-windows-x86\apache-tomcat-10.1.41\webapps\manager] has finished in [82]
ms
30-May-2025 10:23:03.754 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application di
rectory [D:\Downloads\apache-tomcat-10.1.41-windows-x86\apache-tomcat-10.1.41\webapps\ROOT]
30-May-2025 10:23:03.811 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web applicatio
n directory [D:\Downloads\apache-tomcat-10.1.41-windows-x86\apache-tomcat-10.1.41\webapps\ROOT] has finished in [57] ms
30-May-2025 10:23:03.818 INFO [main] org.apache.coyote.AbstractProtocol.start Starting ProtocolHandler ["http-nio-8080"]
30-May-2025 10:23:03.831 INFO [pool-7-thread-3] jenkins.InitReactorRunner$1.onAttained Started initialization
30-May-2025 10:23:03.877 INFO [main] org.apache.catalina.startup.Catalina.start Server startup in [19268] milliseconds
30-May-2025 10:23:04.186 INFO [pool-7-thread-3] jenkins.InitReactorRunner$1.onAttained Listed all plugins
30-May-2025 10:23:11.673 INFO [pool-7-thread-4] jenkins.InitReactorRunner$1.onAttained Prepared all plugins
30-May-2025 10:23:11.719 INFO [pool-7-thread-6] jenkins.InitReactorRunner$1.onAttained Started all plugins
30-May-2025 10:23:11.736 INFO [pool-7-thread-3] jenkins.InitReactorRunner$1.onAttained Augmented all extensions
30-May-2025 10:23:12.795 INFO [GitSCM.onLoaded] hudson.plugins.build_timeout.global.GlobalTimeOutConfiguration.load glob
al timeout not set
30-May-2025 10:23:14.988 INFO [pool-7-thread-3] jenkins.InitReactorRunner$1.onAttained System config loaded
30-May-2025 10:23:14.988 INFO [pool-7-thread-3] jenkins.InitReactorRunner$1.onAttained System config adapted
30-May-2025 10:23:15.068 INFO [pool-7-thread-7] jenkins.InitReactorRunner$1.onAttained Loaded all jobs
30-May-2025 10:23:15.094 INFO [pool-7-thread-7] jenkins.InitReactorRunner$1.onAttained Configuration for all jobs update
d
30-May-2025 10:23:15.178 INFO [pool-7-thread-5] jenkins.InitReactorRunner$1.onAttained Completed initialization
30-May-2025 10:23:15.241 INFO [Jenkins initialization thread] hudson.lifecycle.Lifecycle.onReady Jenkins is fully up and
 running
```

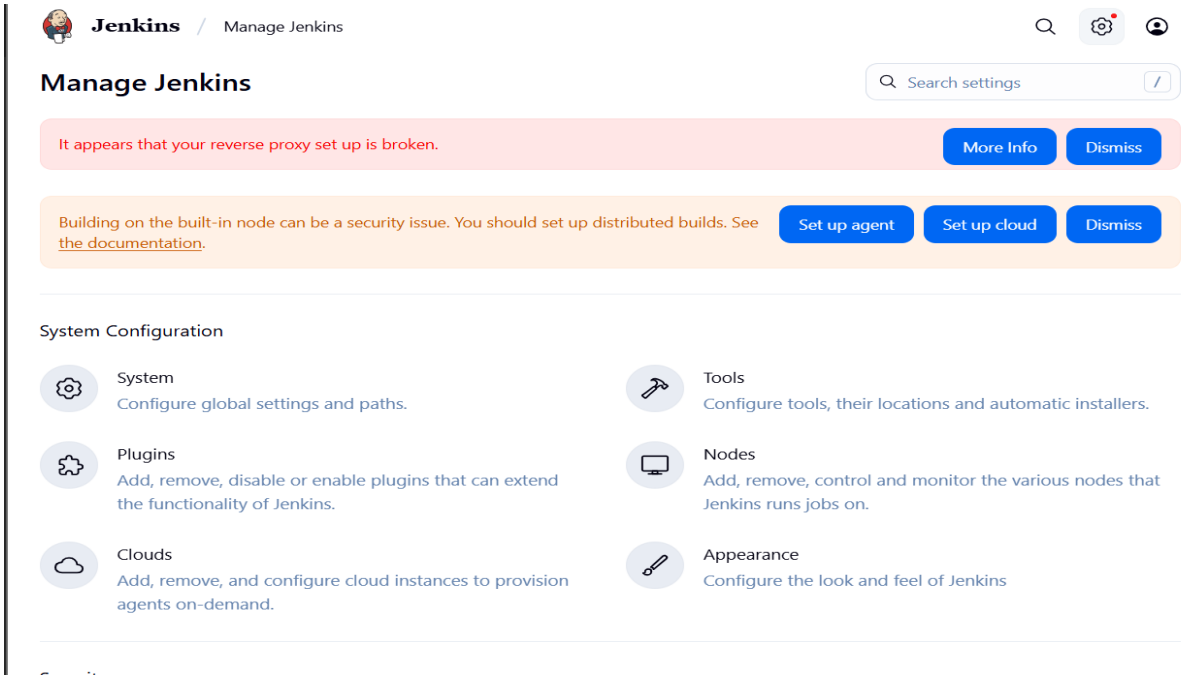
Open the browser and go to the link − **http://localhost:8080/jenkins.**

Jenkins will be up and running on tomcat.



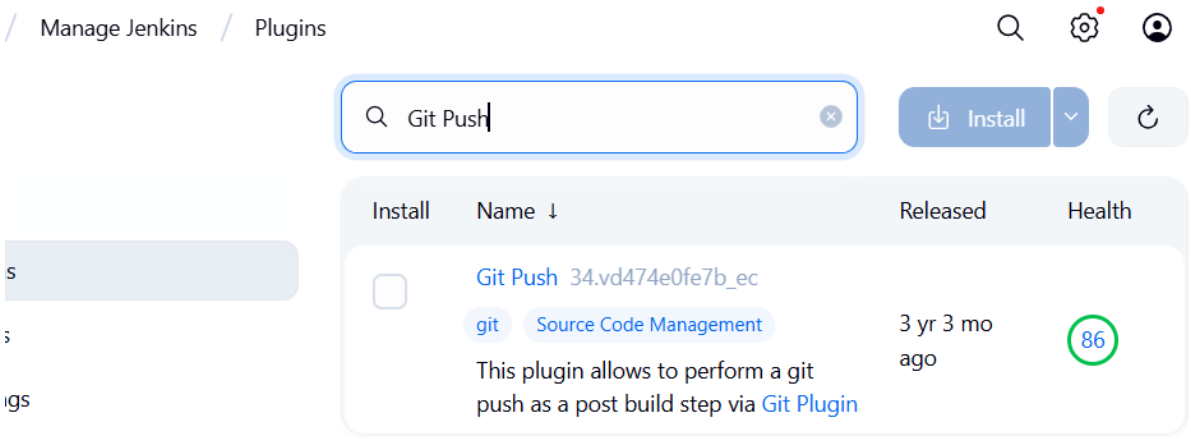
Jenkins - Git Setup

For this exercise, you have to ensure that Internet connectivity is present from the machine on which Jenkins is installed. In your Jenkins Dashboard (Home screen), click the Manage Jenkins option on the left hand side.

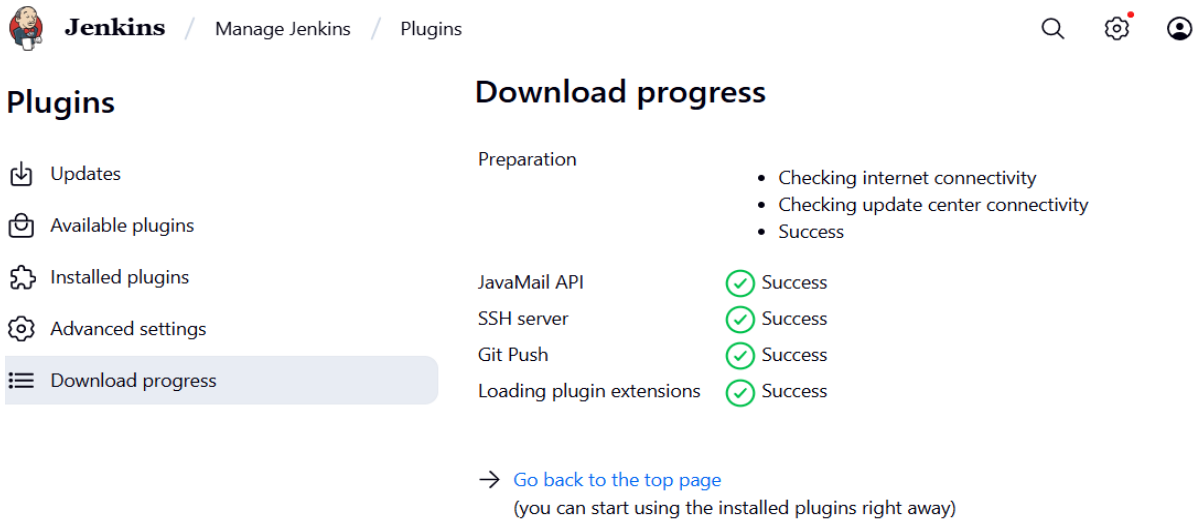


In the next screen, click the ‘Plugins’ option.

In the next screen, click the Available tab. This tab will give a list of plugins which are available for downloading. In the ‘Filter’ tab type ‘Git Push’



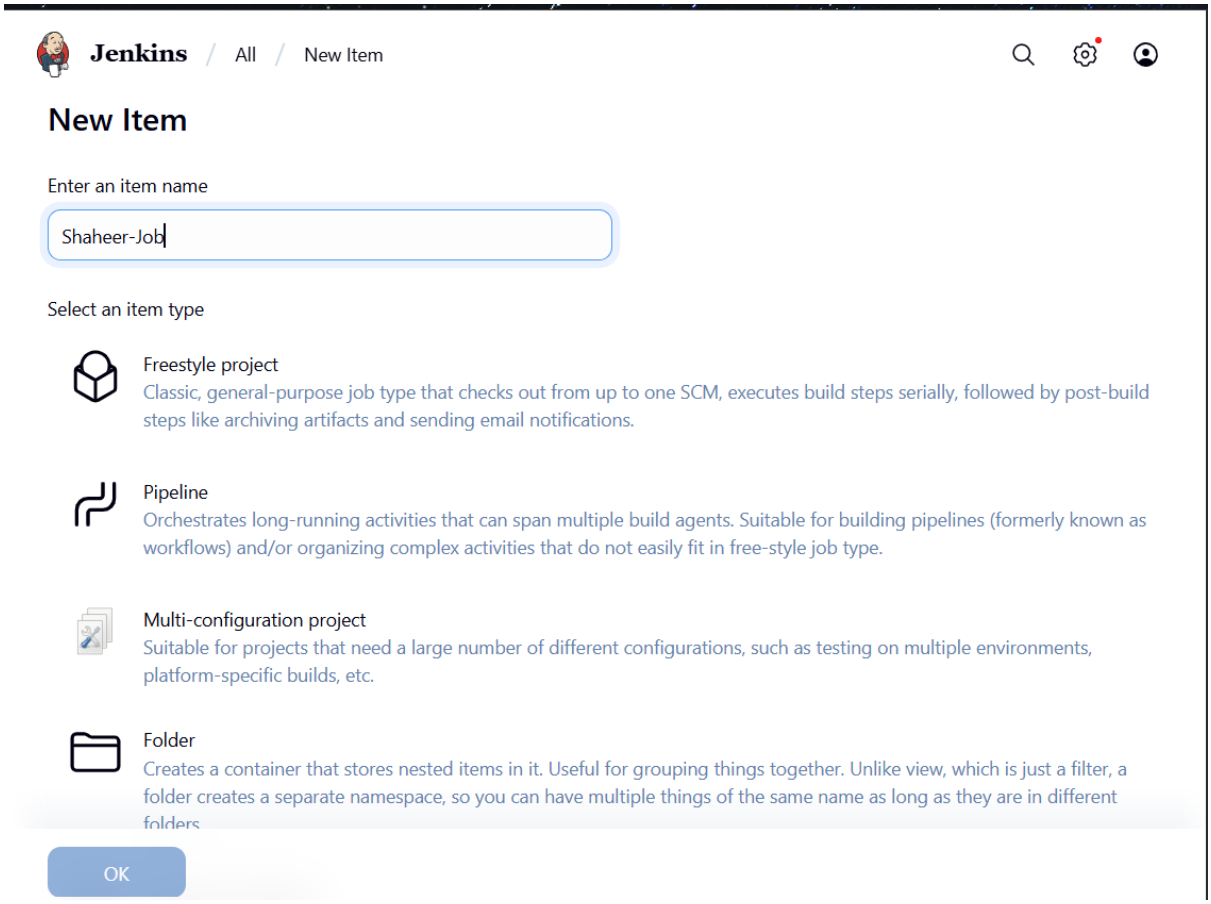
The installation will then begin and the screen will be refreshed to show the status of the download.



Once all installations are complete, restart Jenkins by issue the following command in the

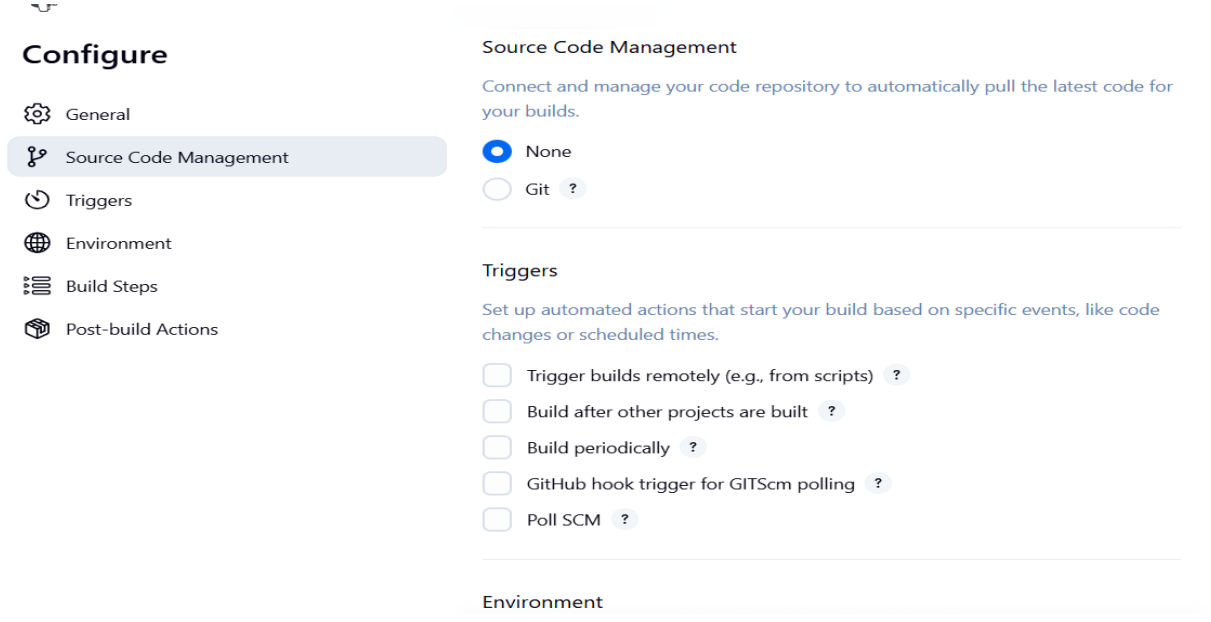
browser. http://localhost:8080/jenkins/restart

Submitted to: Sir Mahaz Khan

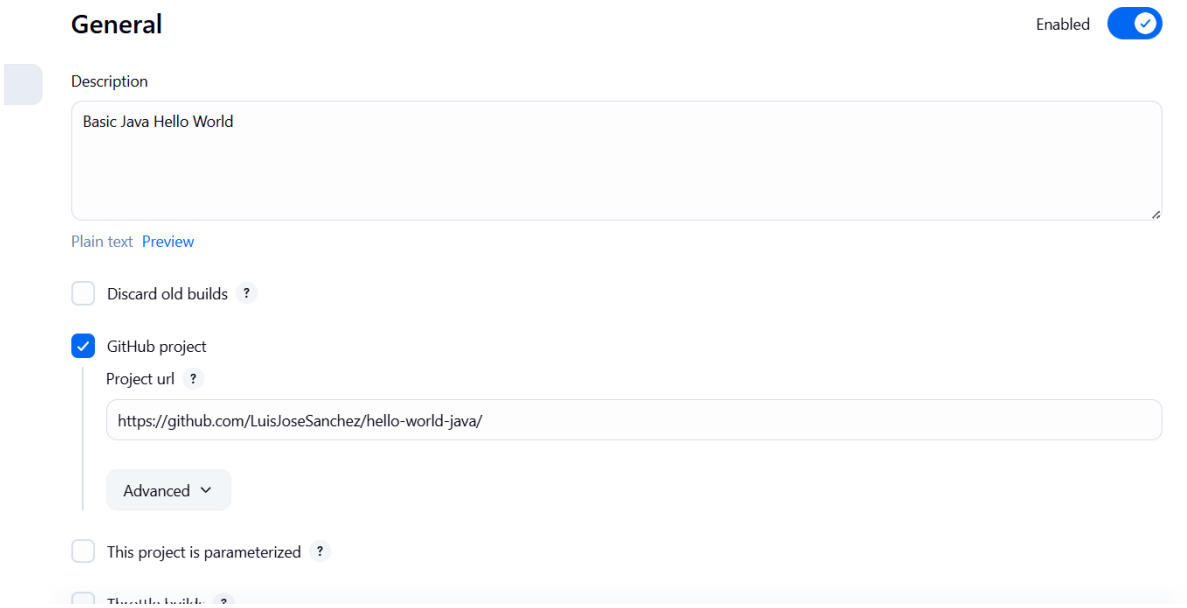


Jenkins - Setup Build Jobs

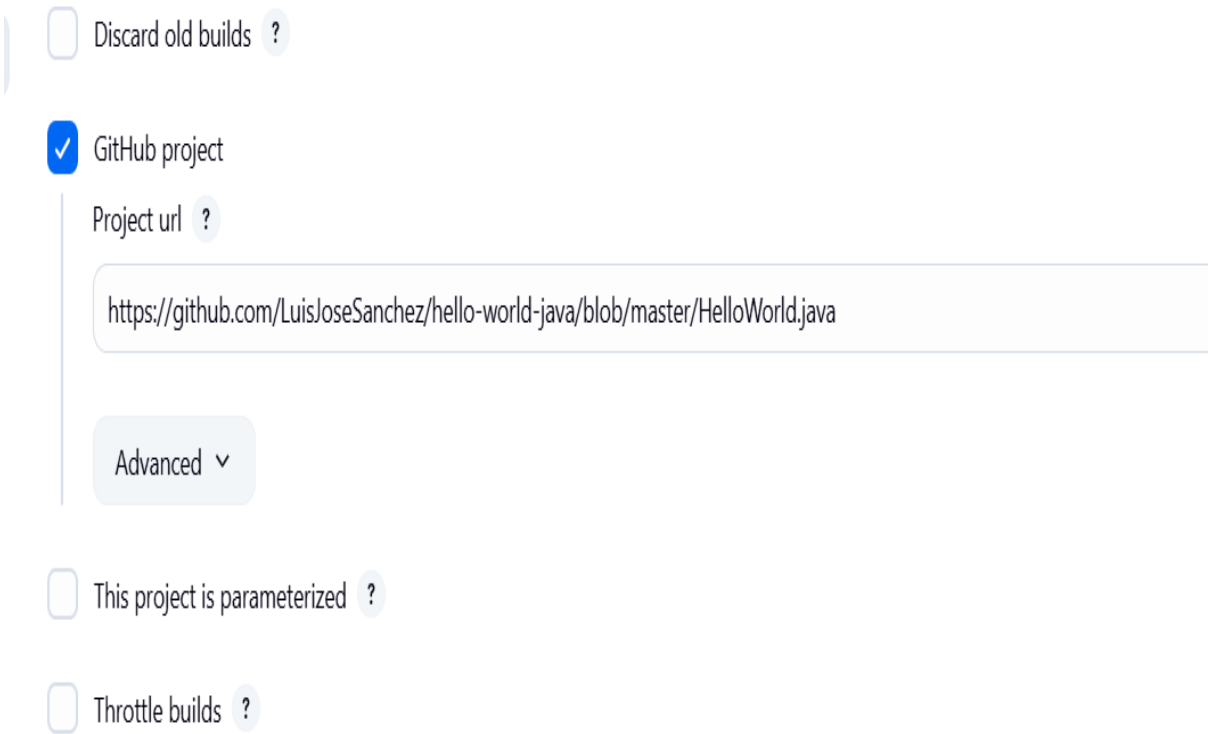
For this exercise, we will create a job in Jenkins which picks up a simple HelloWorld application, builds and runs the java program.



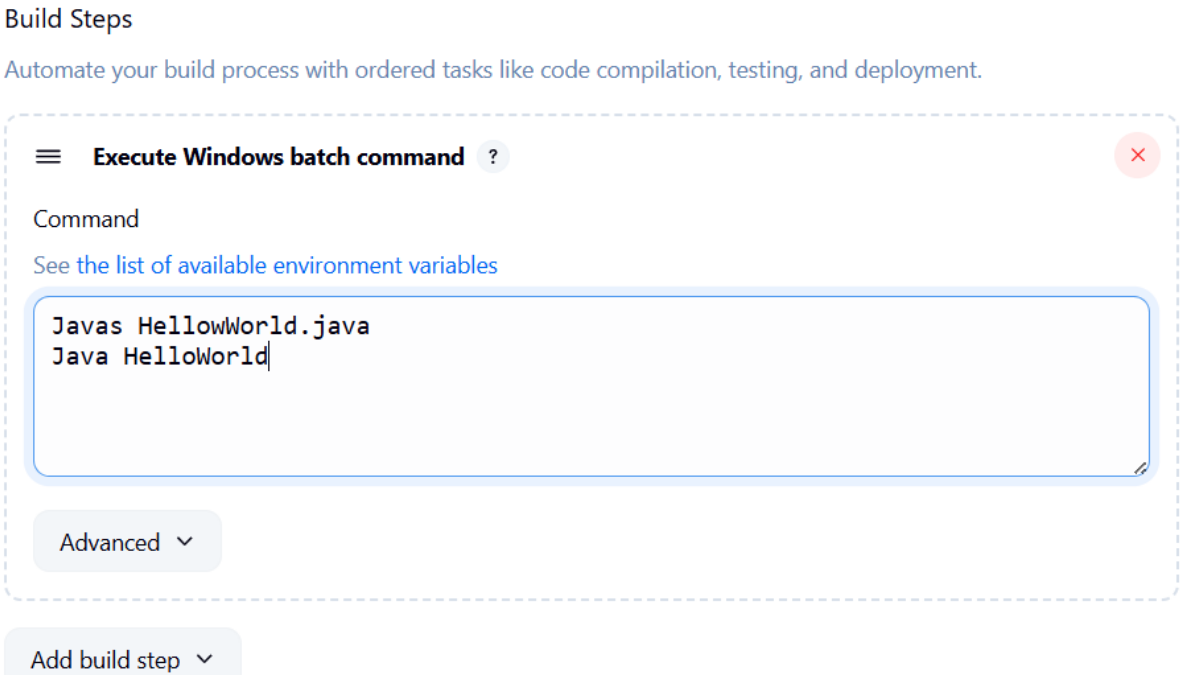
The following screen will come up in which you can specify the details of the job.



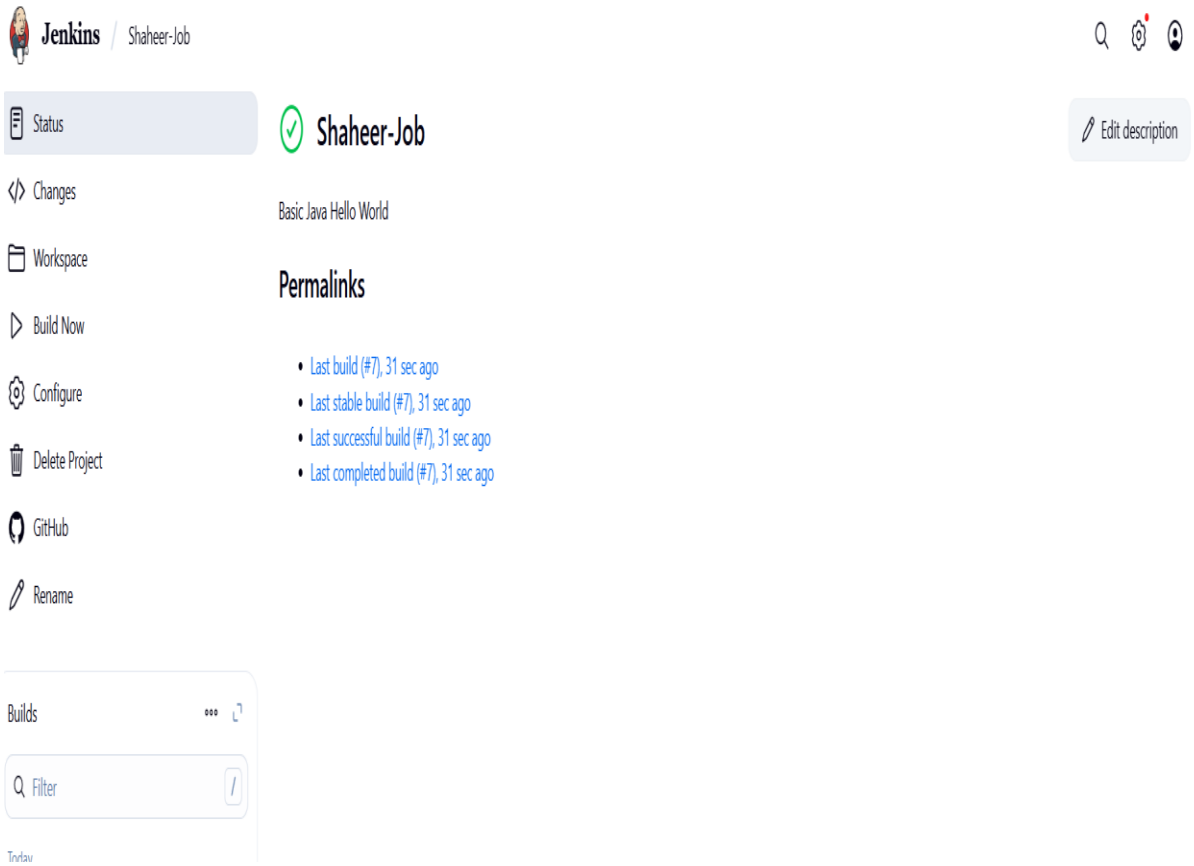
We need to specify the location of files which need to be built. If you repository if hosted on Github, you can also enter the url of that repository here:



Now go to the Build section and click on Add build step → Execute Windows batch command



Once saved, you can click on the Build Now option to see if you have successfully defined the job.



Pre requisites:

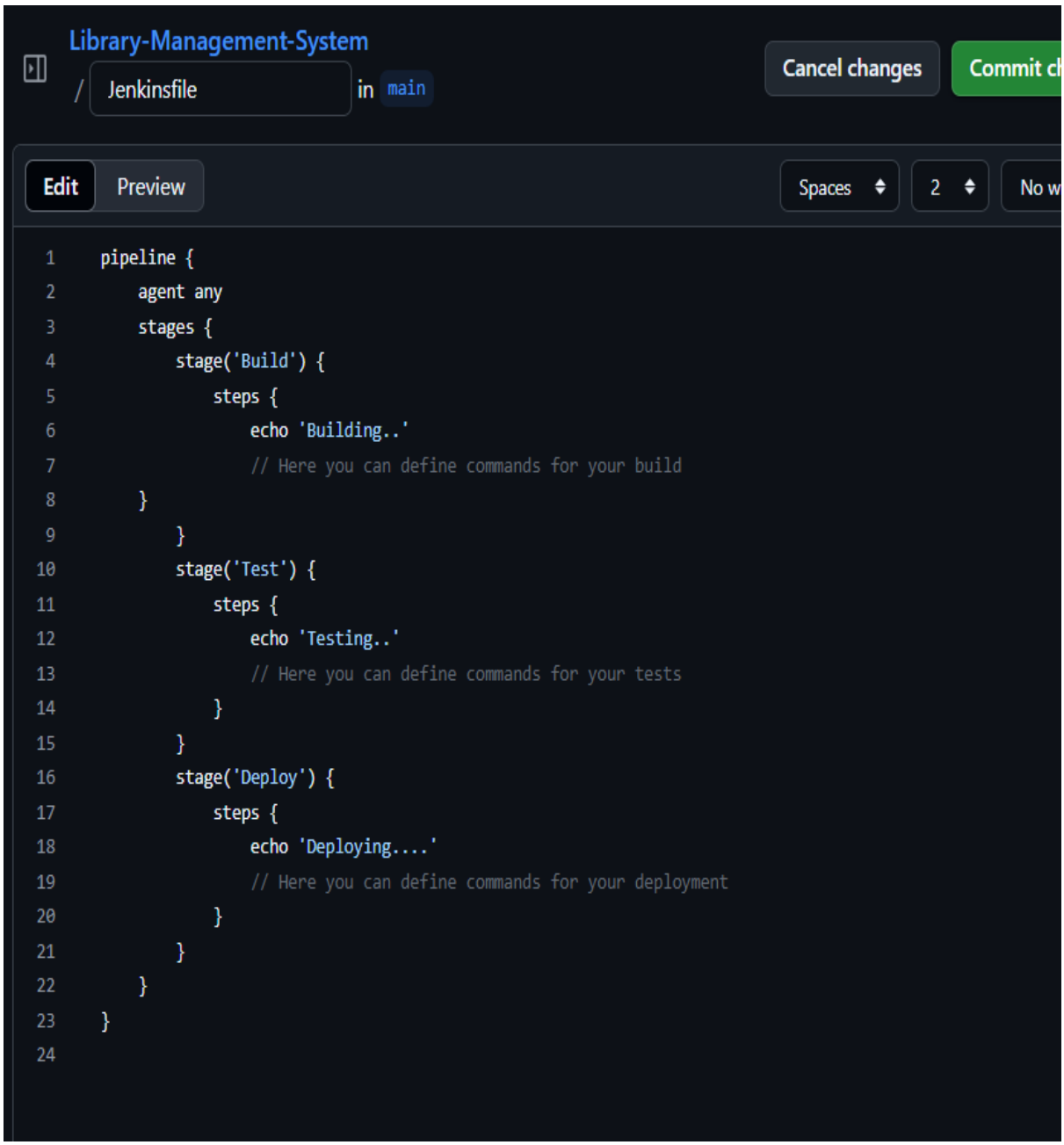
Jenkins should be installed and configured on your machine.

Now you can begin with setting up the pipeline:

What is Jenkins File?

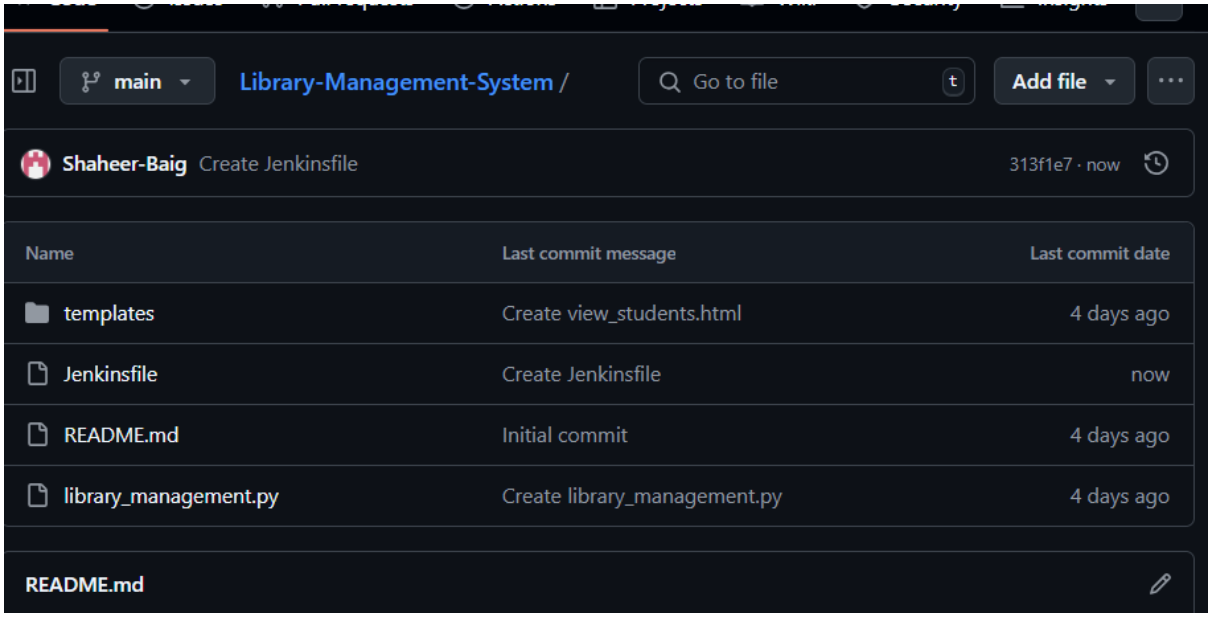
A Jenkins file is a script written in the Groovy programming language that defines the steps to be executed by a Jenkins pipeline. The pipeline is a series of steps executed in a particular order.

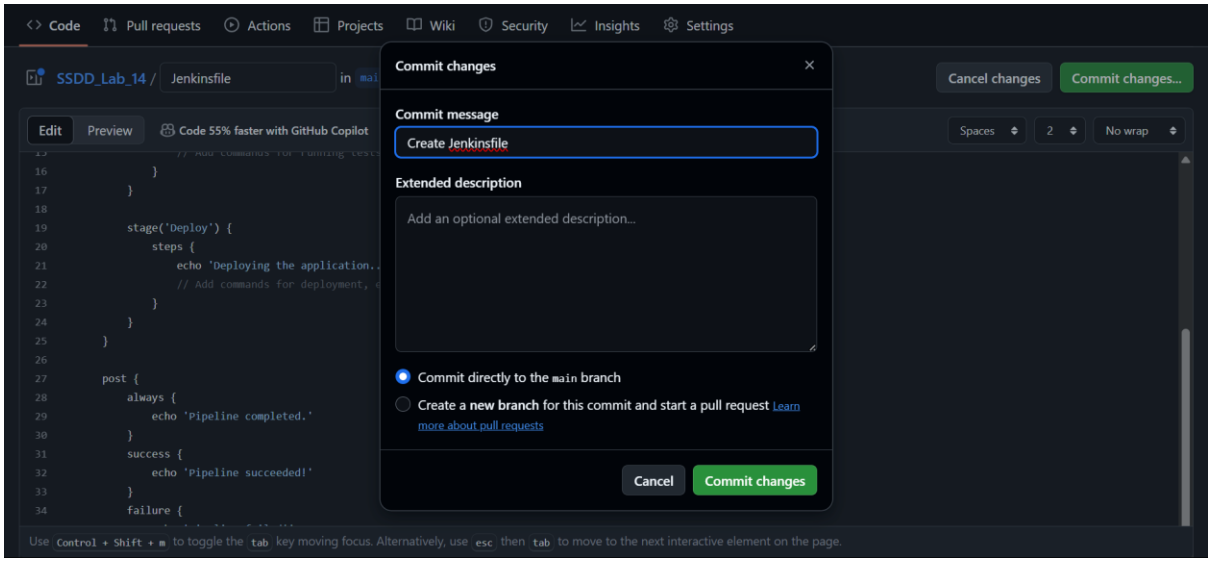
Make a new file named Jenkinsfile in any of your existing GitHub repository and paste the code




```
pipeline {
    agent any
    stages {
        stage('Build') {
            steps {
                echo 'Building..'
                // Here you can define commands for your build
    }
        }
        stage('Test') {
            steps {
                echo 'Testing..'
                // Here you can define commands for your tests
            }
        }
        stage('Deploy') {
            steps {
                echo 'Deploying....'
                // Here you can define commands for your deployment
            }
        }
    }
}
```

- Click on Commit Changes
- The file will appear in your repository:





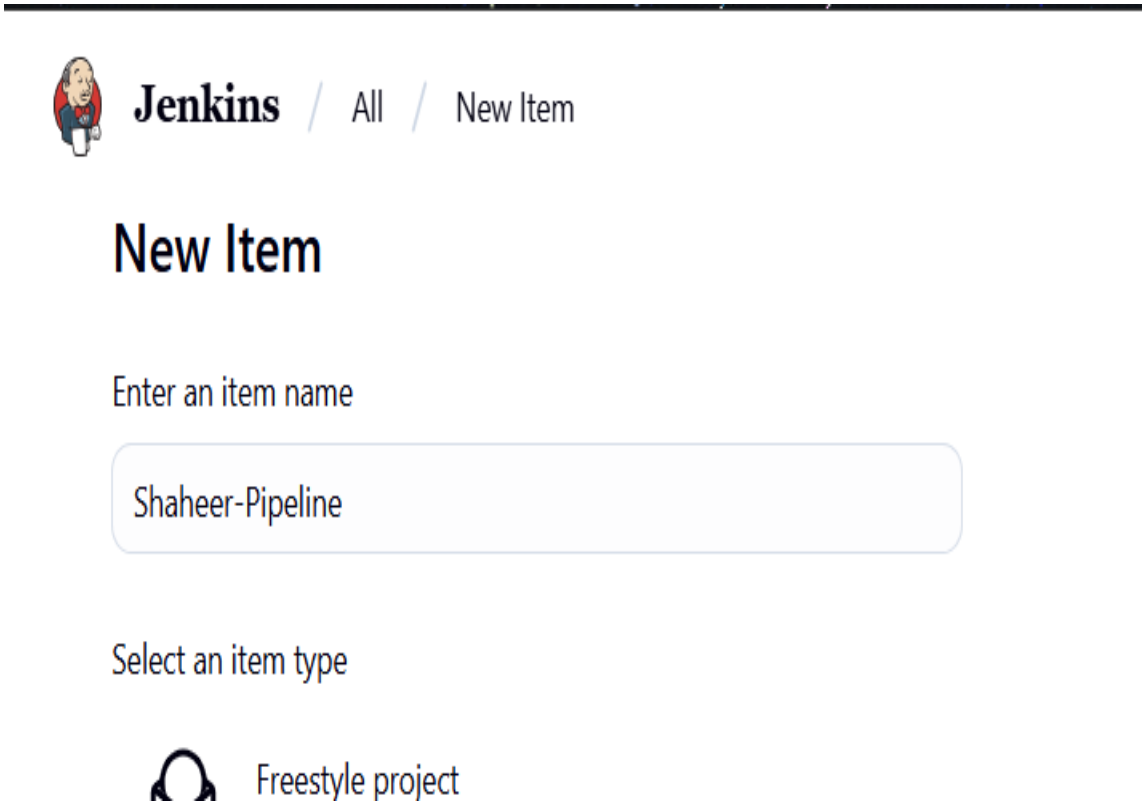
Create Pipeline:

Now start Jenkins

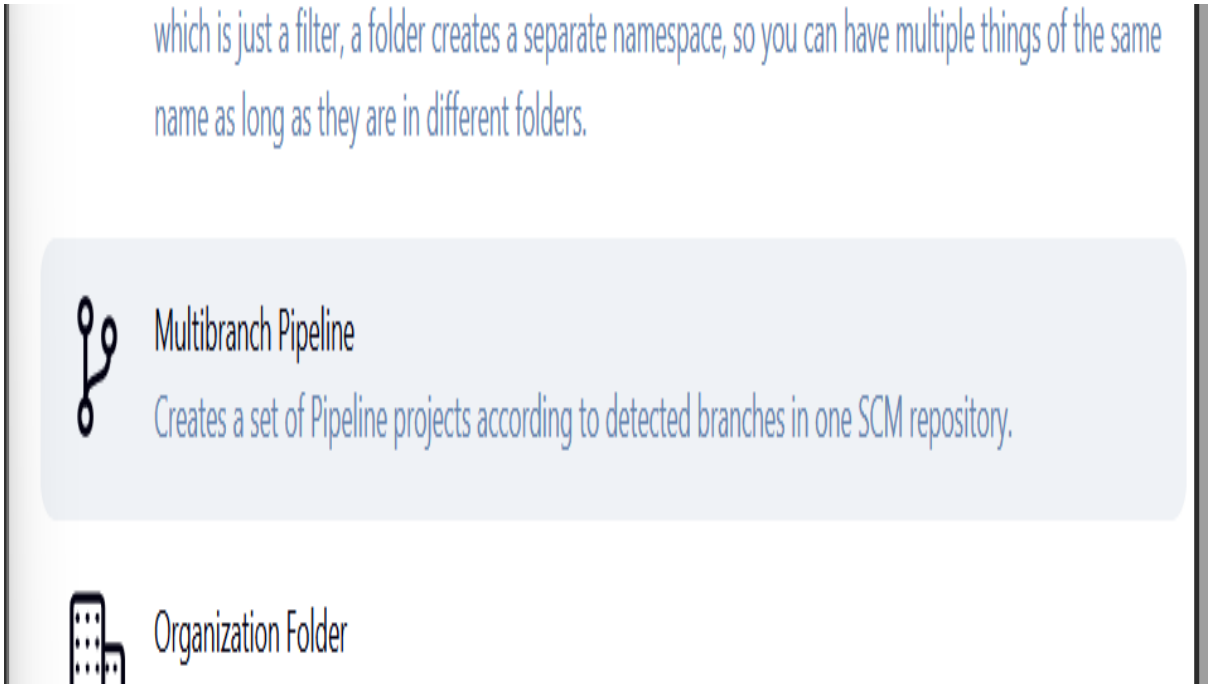
- Go to Jenkins dashboard.

Click on new item.

Add name of your pipeline as **myfirstpipeline.**

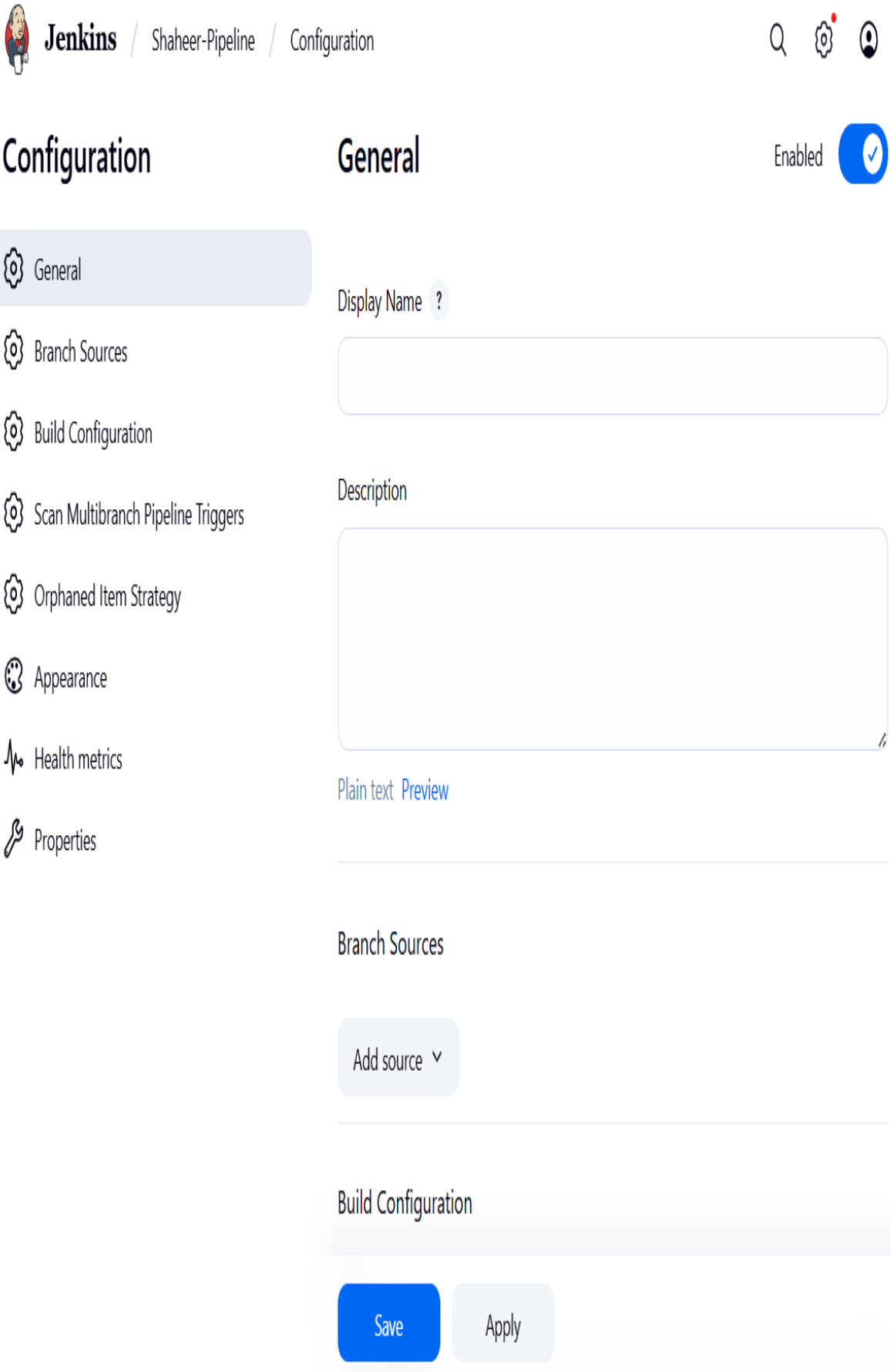


Select multibranch pipeline option:

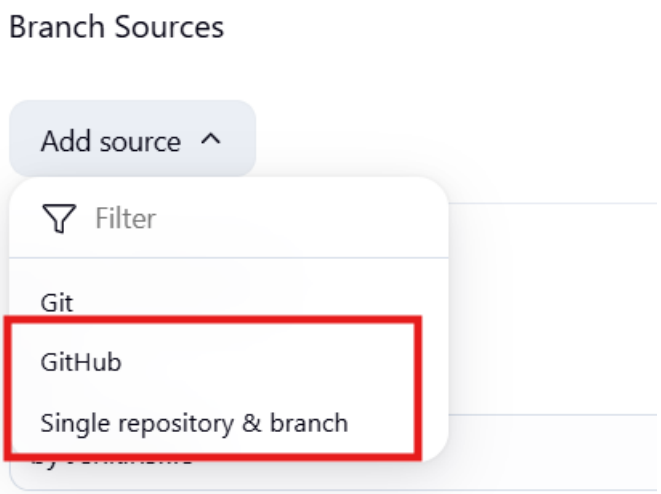


- A new pipeline will be created:

Submitted to: Sir Mahaz Khan

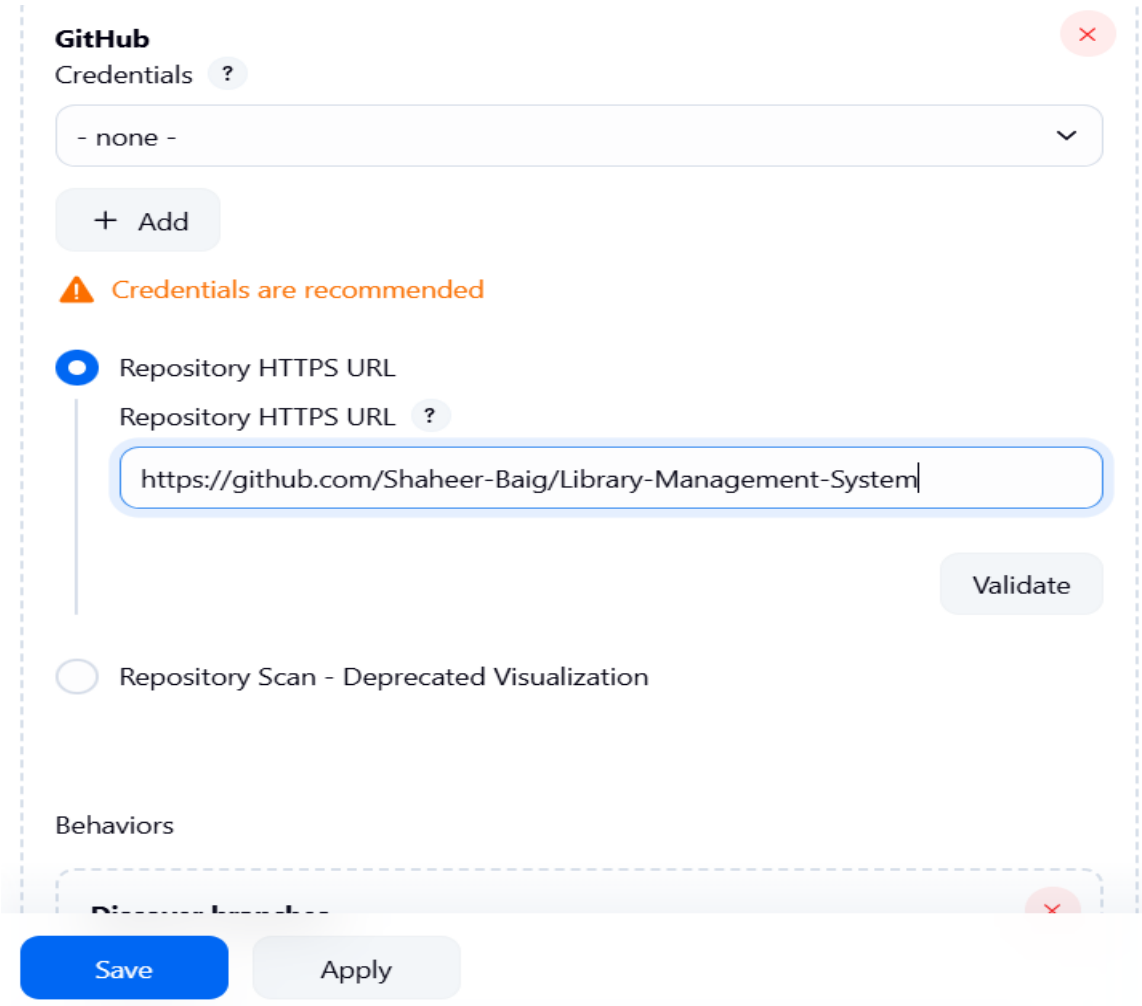


- Now click on add source option:



- Paste the link of your github repository where you created the JenkinsFile

Validate URL:



Validate the link to see if it is correct.

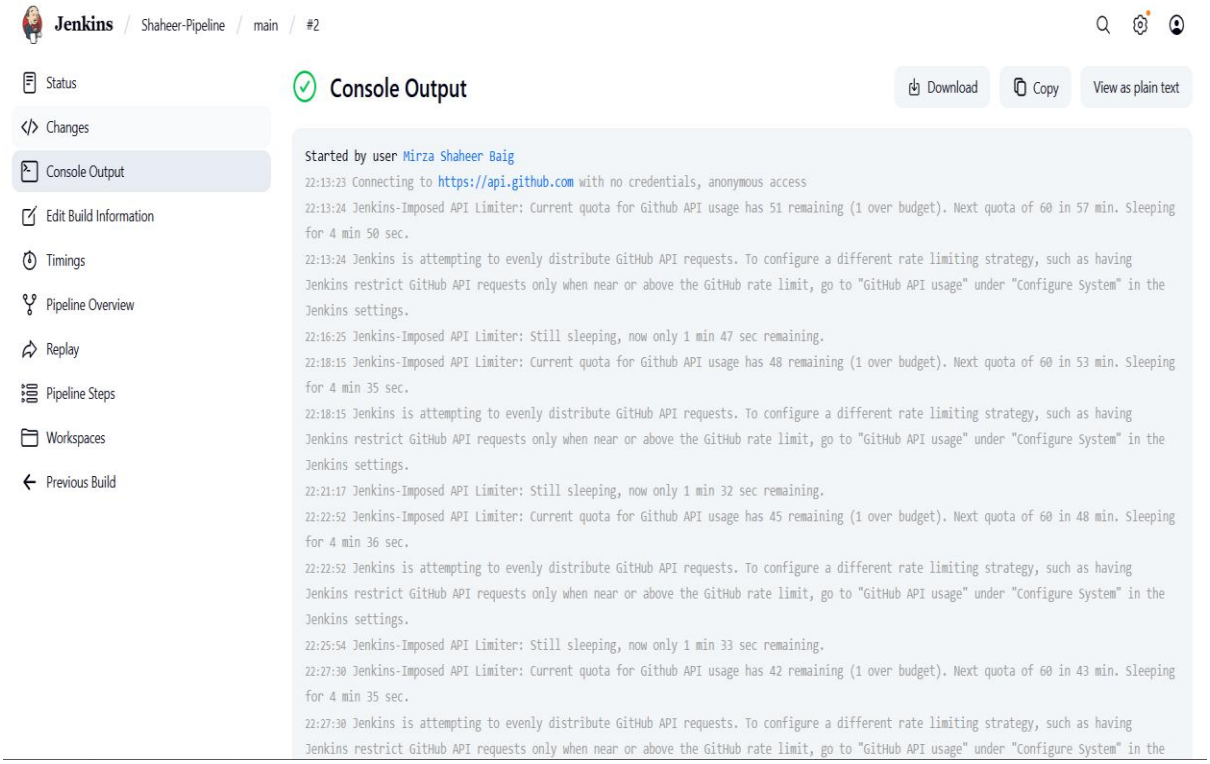
And click on **save.**

- It will start scanning the repository.
- It will scan all the branches one by one to look for Jenkinsfile
- When it is found, the scanning stops and the exits successfully.

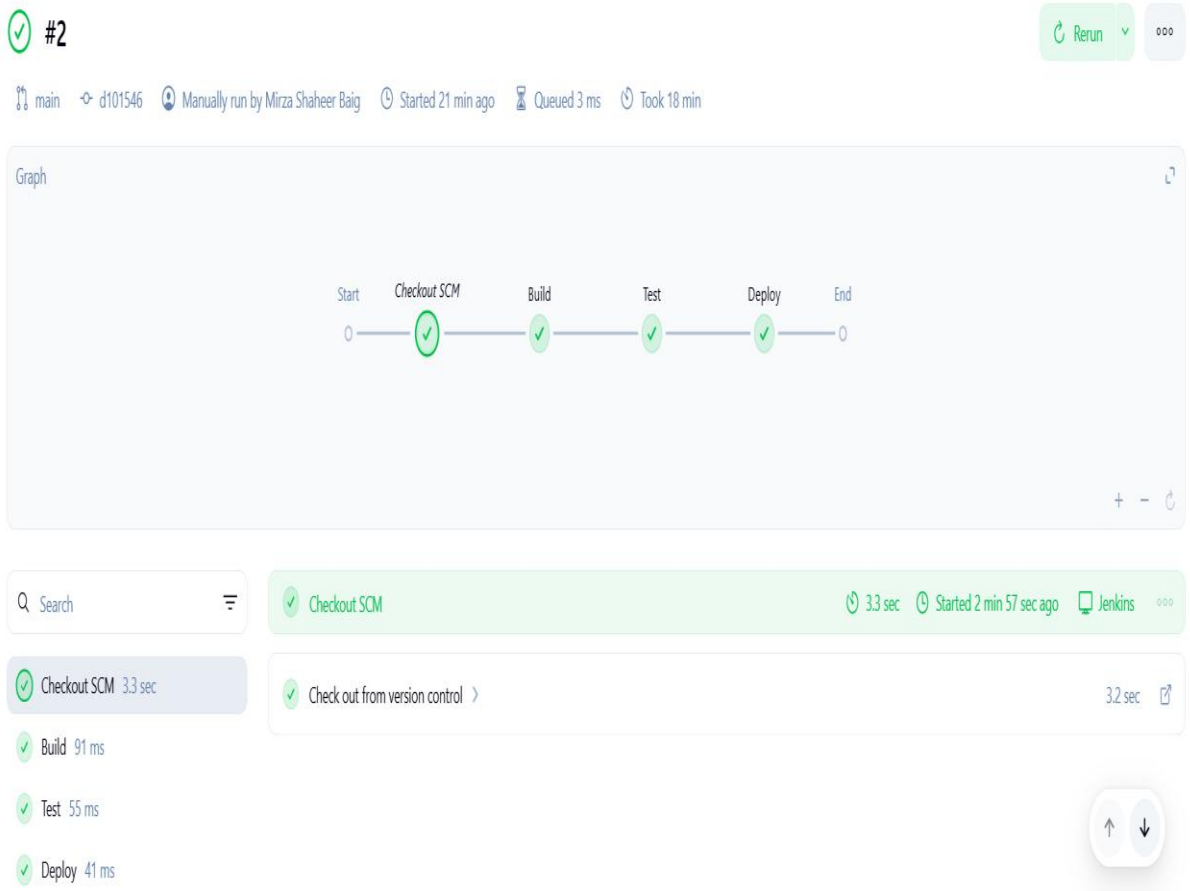
Scan Repository Log

```
Started by user Mirza Shaheer Baig
[Fri Jun 13 20:44:53 PKT 2025] Starting branch indexing...
20:44:54 Connecting to https://api.github.com with no credentials,
anonymous access
Examining Shaheer-Baig/Library-Management-System

  Checking branches...

    Getting remote branches...

    Checking branch main

    Getting remote pull requests...
        'Jenkinsfile' found
      Met criteria
Scheduled build for branch: main

    1 branches were processed

    Checking pull-requests...

    0 pull requests were processed
```

Submitted to: Sir Mahaz Khan



Stages:

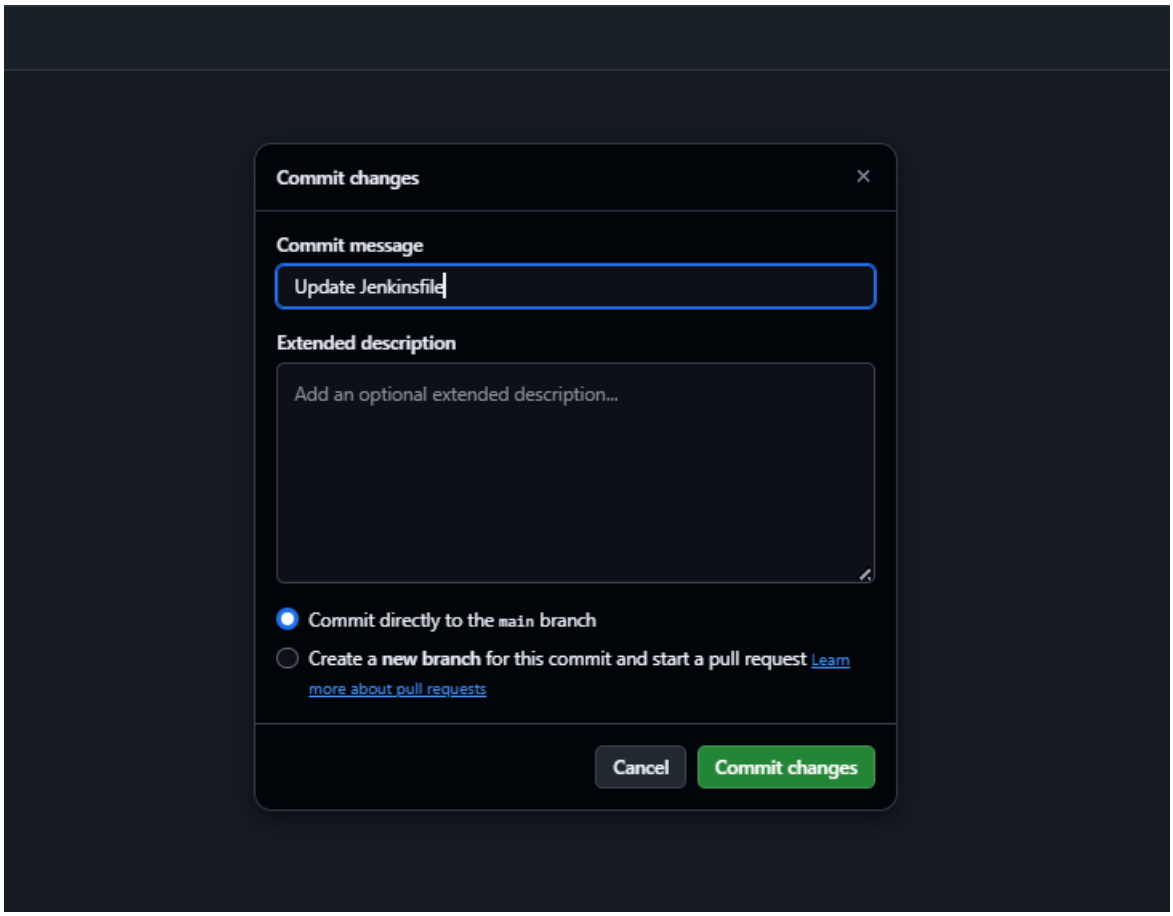


Submitted to: Sir Mahaz Khan

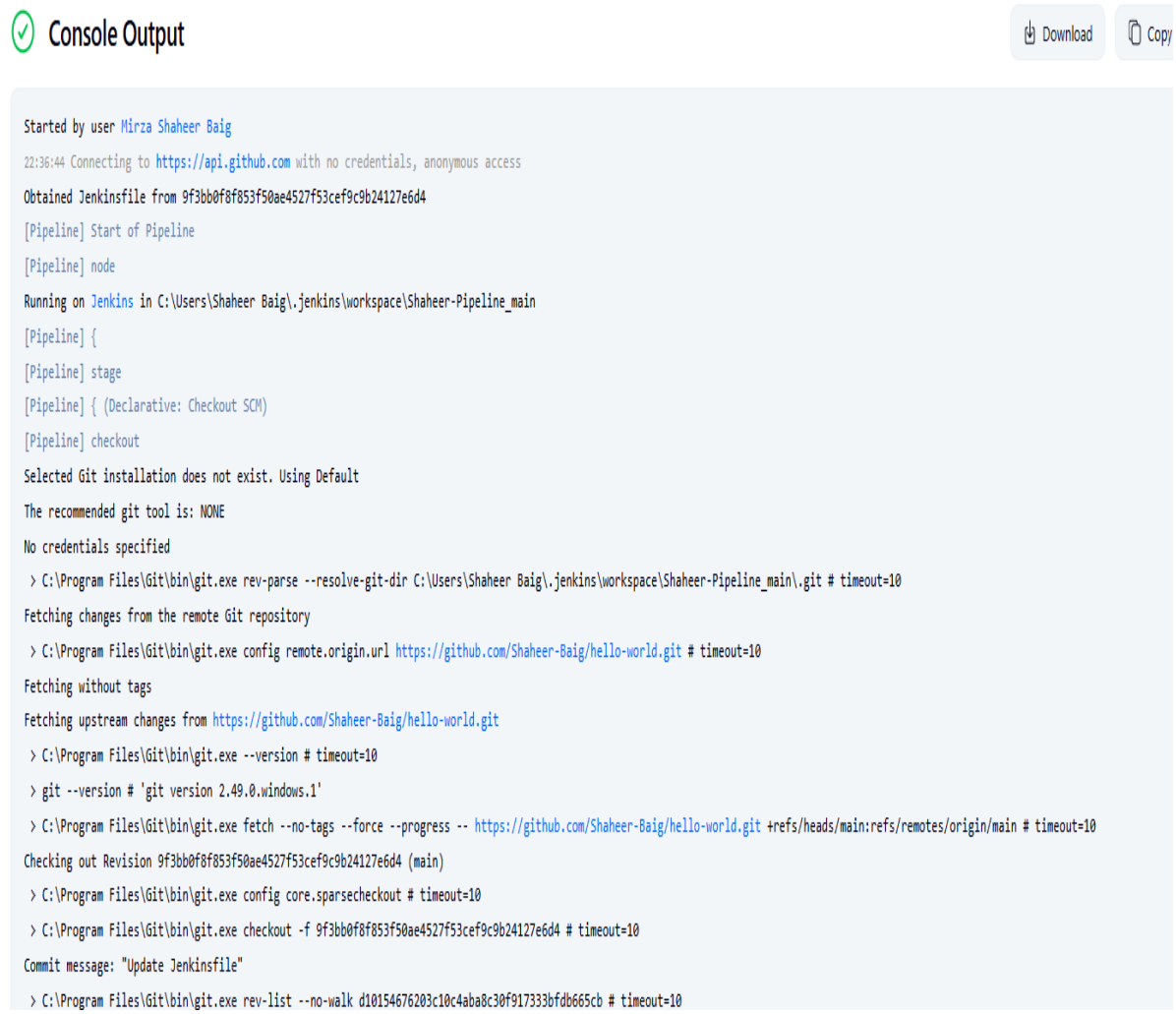
Post Build Actions:

Edit Preview

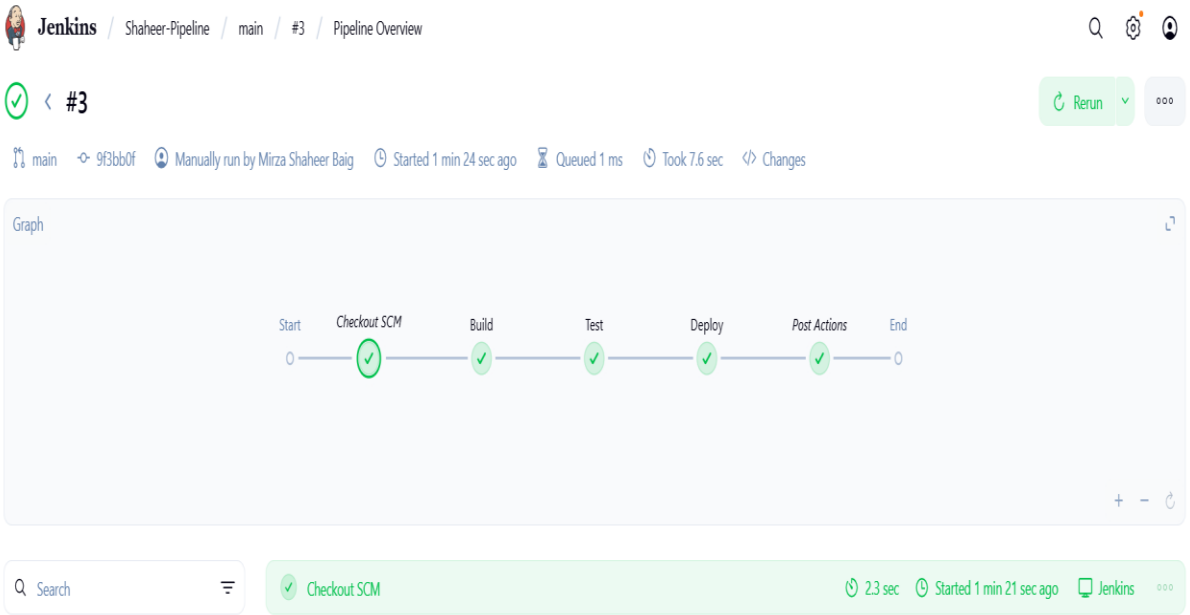
```groovy
pipeline {
    agent any

    stages {
        stage('Build') {
            steps {
                echo 'Building...'
                // Add your build steps here
            }
        }
        stage('Test') {
            steps {
                echo 'Testing...'
                // Add your test steps here
            }
        }
        stage('Deploy') {
            steps {
                echo 'Deploying...'
                // Add your deploy steps here
            }
        }
    }

    post {
        success {
            echo 'Build succeeded!'
        }
        failure {
            echo 'Build failed! Notifying the team...'
            // Add failure-specific steps here, e.g., sending an email or Slack notification
        }
        always {
            echo 'Cleaning up workspace...'
            cleanWs() // Cleans up the workspace after the build
        }
    }
}
```



Build Again:



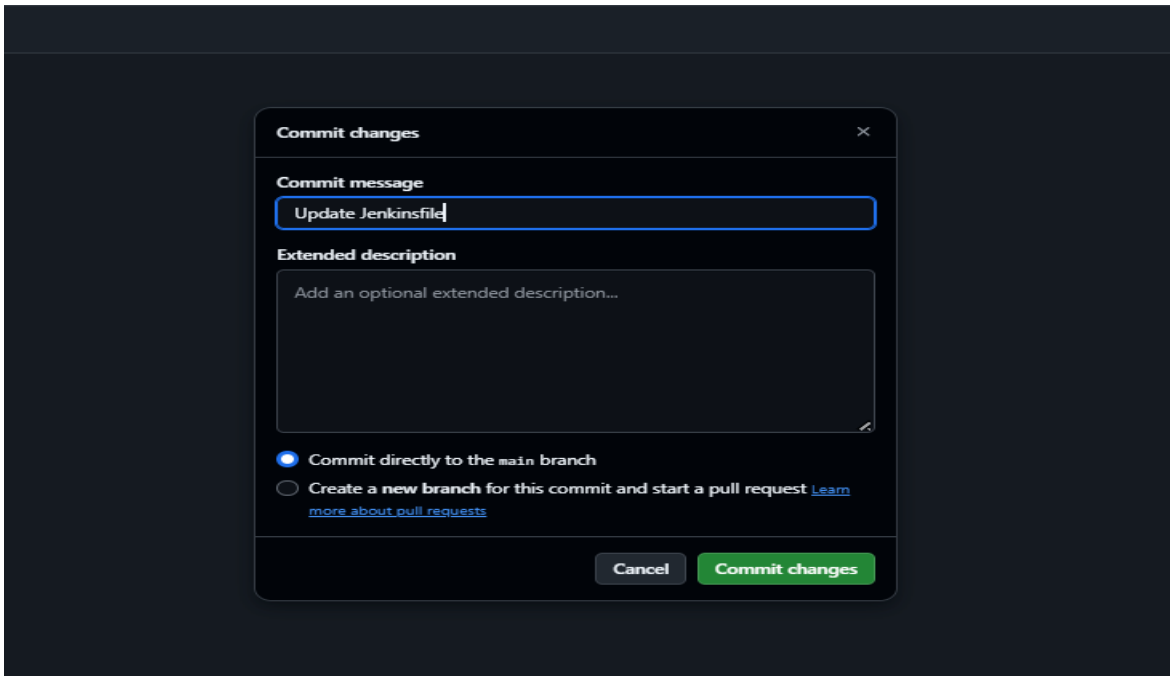
Stages:



Submitted to: Sir Mahaz Khan

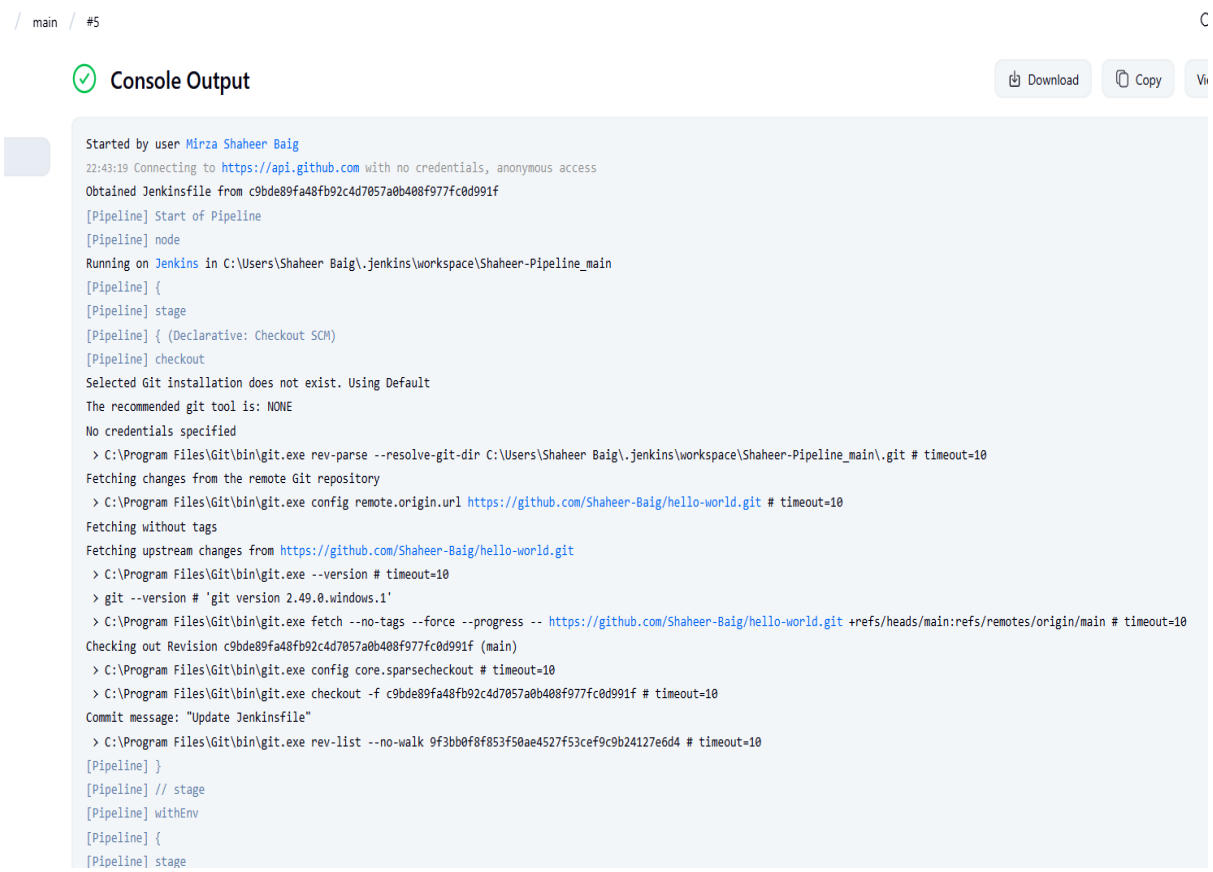
Define Conditionals for each stage:

```
pipeline {
    agent any

    stages {
        stage('Build') {
            steps {
                echo 'Building...'
                bat 'echo Build process completed!'
            }
        }

        stage('Test') {
            when {
                allOf {
                    expression { env.BRANCH_NAME == 'main' }
                    not {
                        expression { env.SKIP_TESTS == 'true' }
                    }
                }
            }
            steps {
                echo 'Testing...'
                bat 'echo Running test cases!'
            }
        }

        stage('Deploy') {
            steps {
                echo 'Deploying...'
                bat 'echo Deployment process completed!'
            }
        }
    }
}
```

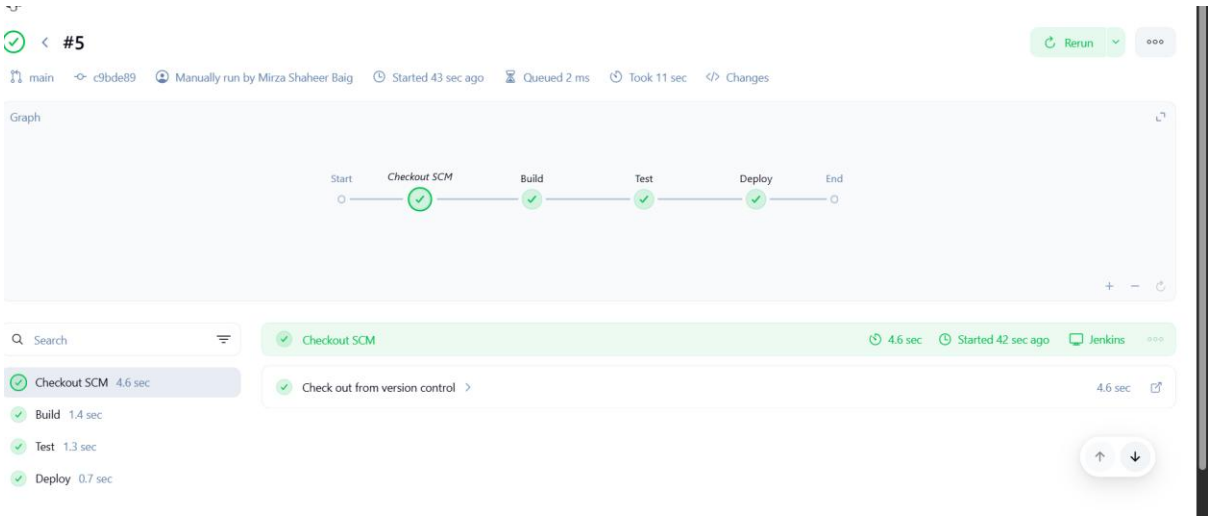


Submitted to: Sir Mahaz Khan

Build Again:

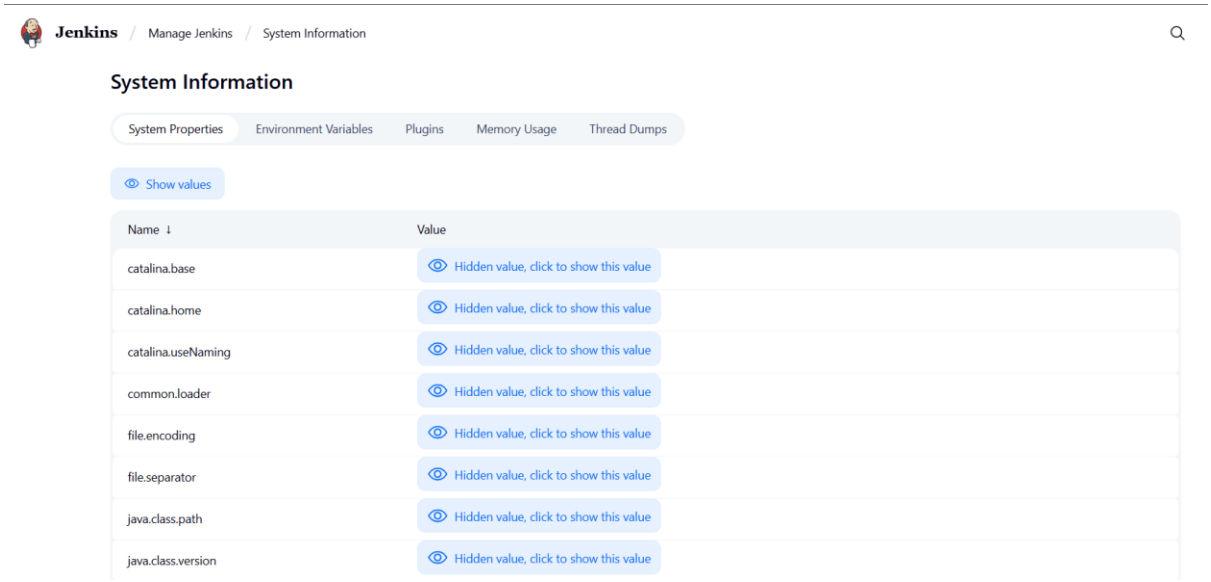


Stages:

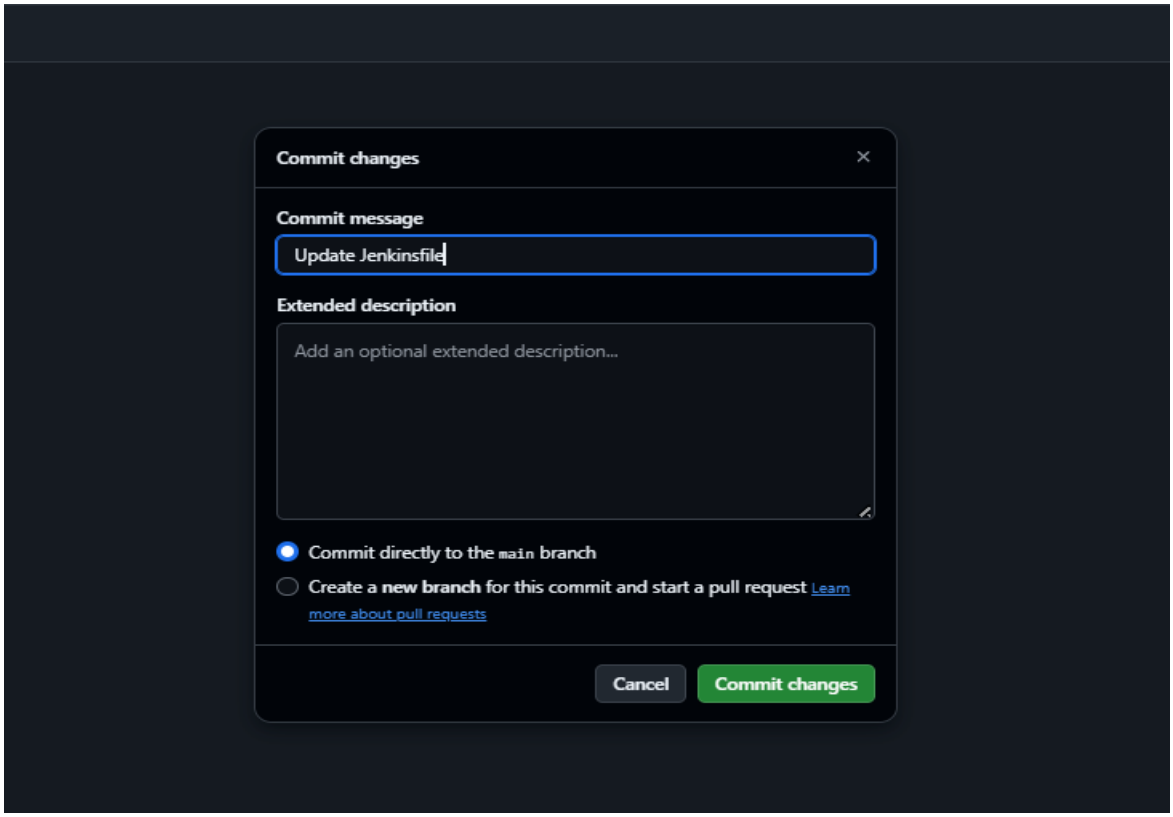


Submitted to: Sir Mahaz Khan

Environment Variables:




```
    // Define custom environment variables
    environment {
        APP_VERSION = '1.0.3'       // Example application version
        DEPLOY_ENV = 'staging'      // Deployment environment
        AUTHOR_NAME = 'Shaheer Baig' // Example author name
    }

    stages {
        stage('Build') {
            steps {
                echo "Building version: ${env.APP_VERSION}"
                echo "Author: ${env.AUTHOR_NAME}"
                // Your build logic here
                bat 'echo Build successful'
            }
        }

        stage('Test') {
            steps {
                echo "Running tests for version: ${env.APP_VERSION}"
                // Your test logic here
                bat 'echo All tests passed'
            }
        }

        stage('Deploy') {
            steps {
                echo "Deploying version: ${env.APP_VERSION} to environment: ${env.DEPLOY_ENV}"
                // Your deployment logic here
                bat 'echo Deployment done'
            }
        }
    }
}
SS
```



Built Again:

```
[Pipeline] echo
Testing...
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Deploy)
[Pipeline] echo
Deploying...
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Declarative: Post Actions)
[Pipeline] echo
Cleaning up workspace...
[Pipeline] cleanWs
[WS-CLEANUP] Deleting project workspace...
[WS-CLEANUP] Deferred wipeout is used...
[WS-CLEANUP] done
[Pipeline] echo
Build succeeded!
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```

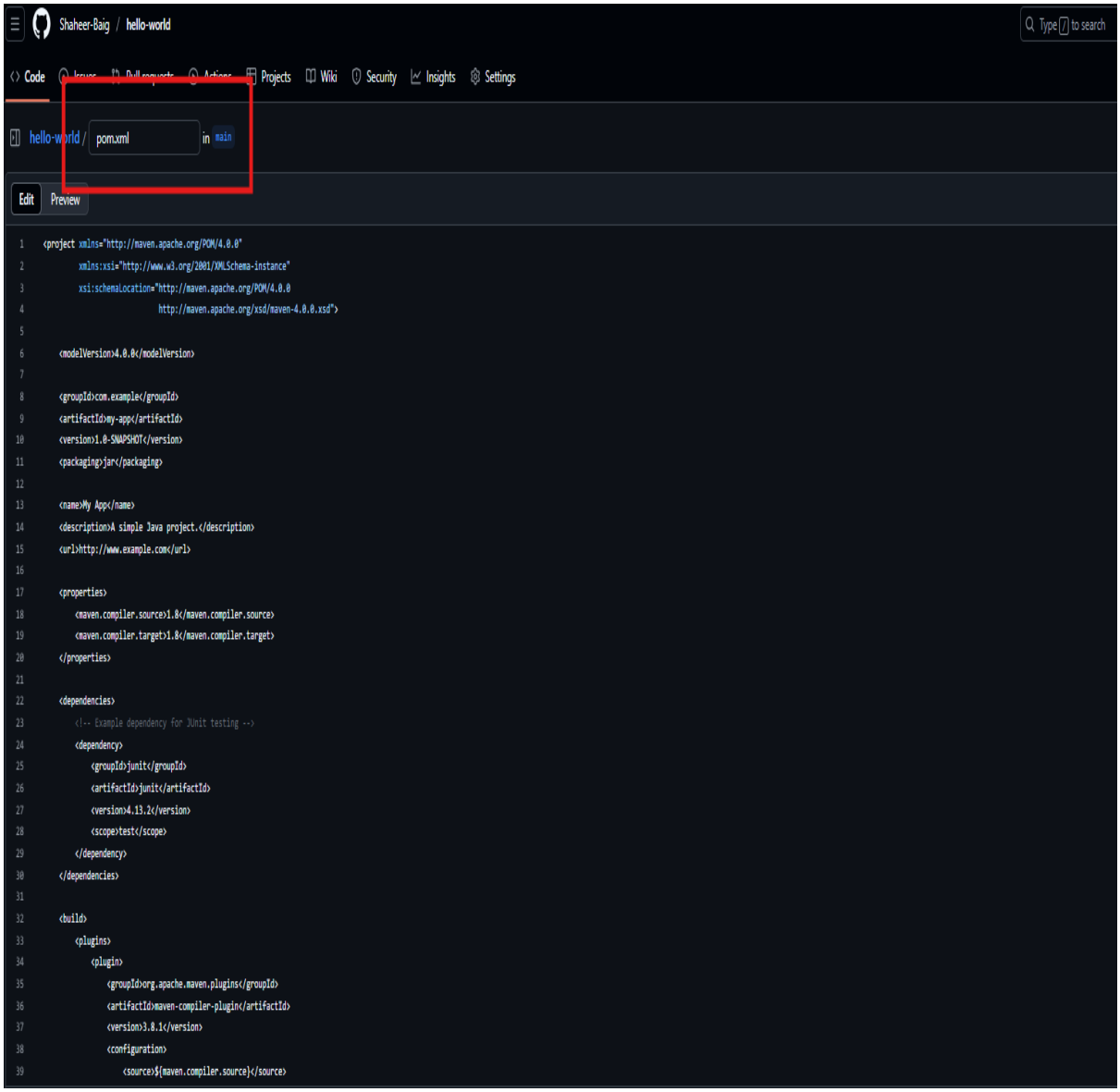
Submitted to: Sir Mahaz Khan

Tools attribute for build tools:

```
pipeline {
    agent any

    tools {
        // Use the exact name of your Maven configuration as defined in Jenkins Global Tool Configuration
        maven 'Maven'
    }

    environment {
        // Define any environment variables you may need
        APP_VERSION = '1.0.3'
    }

    stages {
        stage('Setup') {
            steps {
                echo "Setting up environment for Maven..."
            }
        }

        stage('Build') {
            steps {
                // For Windows, use 'bat' instead of 'sh'
```

Use Control + Shift + m to toggle the tab key moving focus. Alternatively, use esc then tab to move to the next interactive element on the page.

Create pom.xml:



Shaheer-Baig / hello-world

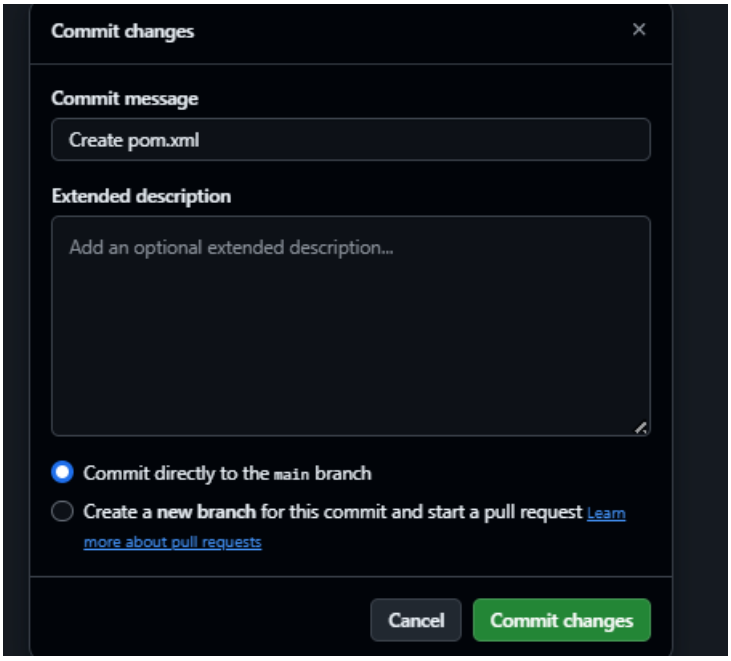
Code Issues Pull requests Actions Projects Wiki Security Insights Settings

hello-world / pom.xml in main

Edit Preview

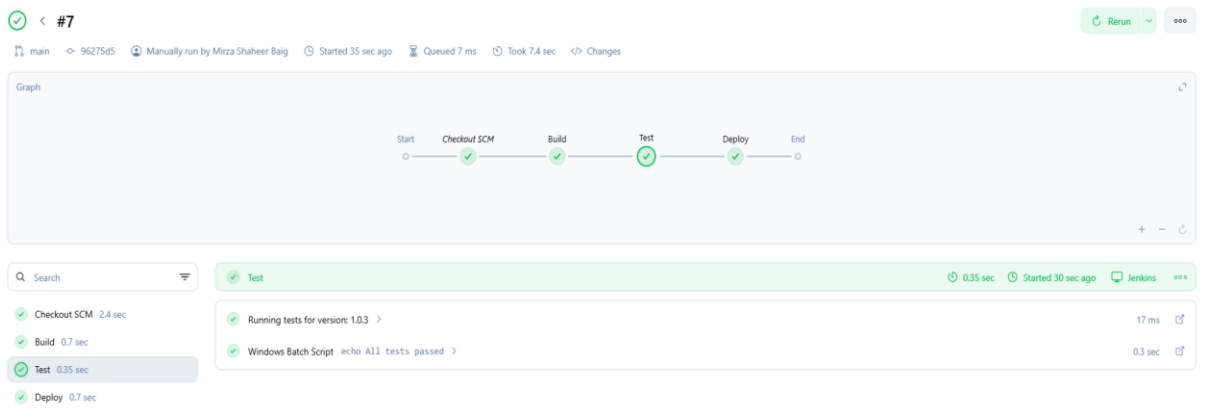
```
<project xmlns="http://maven.apache.org/POM/4.0.0"
         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
         xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
                             http://maven.apache.org/xsd/maven-4.0.0.xsd">

    <modelVersion>4.0.0</modelVersion>

    <groupId>com.example</groupId>
    <artifactId>my-app</artifactId>
    <version>1.0-SNAPSHOT</version>
    <packaging>jar</packaging>

    <name>My App</name>
    <description>A simple Java project.</description>
    <url>http://www.example.com</url>

    <properties>
        <maven.compiler.source>1.8</maven.compiler.source>
        <maven.compiler.target>1.8</maven.compiler.target>
    </properties>

    <dependencies>
        <!-- Example dependency for JUnit testing -->
        <dependency>
            <groupId>junit</groupId>
            <artifactId>junit</artifactId>
            <version>4.13.2</version>
            <scope>test</scope>
        </dependency>
    </dependencies>

    <build>
        <plugins>
            <plugin>
                <groupId>org.apache.maven.plugins</groupId>
                <artifactId>maven-compiler-plugin</artifactId>
                <version>3.8.1</version>
                <configuration>
                    <source>${maven.compiler.source}</source>
```

Submitted to: Sir Mahaz Khan



Build Again:

Stages:



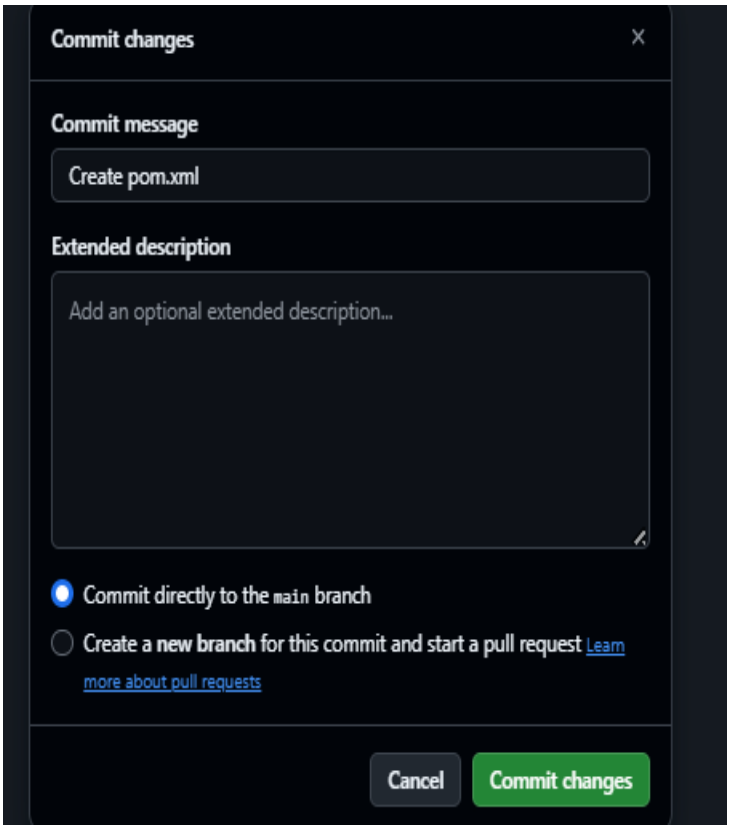
Submitted to: Sir Mahaz Khan

Parameters in Jenkins:

hello-world / Jenkinsfile in main

Edit Preview

```
pipeline {
    agent any

    parameters {
        booleanParam(
            name: 'executeTests',
            defaultValue: true,
            description: 'Run tests during the pipeline build?'
        )
    }

    stages {
        stage('Setup') {
            steps {
                echo 'Setting up the build environment...'
            }
        }

        stage('Build') {
            steps {
                echo 'Building the project...'
            }
        }

        stage('Test') {
            when {
                expression { params.executeTests }
            }
            steps {
                echo 'Running tests...'
            }
        }

        stage('Deploy') {
            steps {
                echo 'Deploying the project...'
            }
        }
    }
```



Submitted to: Sir Mahaz Khan

Build Again:

Console Output

Download Copy View as plain text

```
Started by user Mirza Shaheer Baig
22:55:19 Connecting to https://api.github.com with no credentials, anonymous access
22:55:21 Jenkins is restricting GitHub API requests only when near or above the rate limit. To configure a different rate limiting strategy, such as having Jenkins attempt to evenly distribute GitHub API
requests, go to "GitHub API usage" under "Configure System" in the Jenkins settings.
22:55:21 Jenkins-Imposed API Limiter: Current quota for Github API usage has 14 remaining (1 over budget). Next quota of 60 in 15 min. Sleeping until reset.
22:58:22 Jenkins-Imposed API Limiter: Still sleeping, now only 13 min remaining.
23:01:24 Jenkins-Imposed API Limiter: Still sleeping, now only 10 min remaining.
23:04:25 Jenkins-Imposed API Limiter: Still sleeping, now only 7 min 14 sec remaining.
23:07:27 Jenkins-Imposed API Limiter: Still sleeping, now only 4 min 12 sec remaining.
23:10:29 Jenkins-Imposed API Limiter: Still sleeping, now only 1 min 10 sec remaining.
Obtained Jenkinsfile from 3057e3287b7b6be73058994f614d290ec40a897a
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in C:\Users\Shaheer Baig\.jenkins\workspace\Shaheer-Pipeline_main
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Declarative: Checkout SCM)
[Pipeline] checkout
Selected Git installation does not exist. Using Default
The recommended git tool is: NONE
No credentials specified
 > C:\Program Files\Git\bin\git.exe rev-parse --resolve-git-dir C:\Users\Shaheer Baig\.jenkins\workspace\Shaheer-Pipeline_main\.git # timeout=10
Fetching changes from the remote Git repository
 > C:\Program Files\Git\bin\git.exe config remote.origin.url https://github.com/Shaheer-Baig/hello-world.git # timeout=10
Fetching without tags
Fetching upstream changes from https://github.com/Shaheer-Baig/hello-world.git
 > C:\Program Files\Git\bin\git.exe --version # timeout=10
 > git --version # 'git version 2.49.0.windows.1'
 > C:\Program Files\Git\bin\git.exe fetch --no-tags --force --progress -- https://github.com/Shaheer-Baig/hello-world.git +refs/heads/main:refs/remotes/origin/main # timeout=10
Checking out Revision 3057e3287b7b6be73058994f614d290ec40a897a (main)
 > C:\Program Files\Git\bin\git.exe config core.sparsecheckout # timeout=10
 > C:\Program Files\Git\bin\git.exe checkout -f 3057e3287b7b6be73058994f614d290ec40a897a # timeout=10
Commit message: "Update Jenkinsfile"
 > C:\Program Files\Git\bin\git.exe rev-list --no-walk 96275d51604820cc83e280c7b3086d449702791c # timeout=10
[Pipeline] }
[Pipeline] // stage
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Setup)
[Pipeline] echo
Setting up the build environment...
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Build)
[Pipeline] echo
Building the project...
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Test)
[Pipeline] echo
Running tests...
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Deploy)
[Pipeline] echo
Deploying the project...
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```

Stages: