

# MT2022118 - SUBHAM BASU ROY CHOWDHURY

## Calculator with Devops MiniProject

### What is Devops?

DevOps is a set of practices that combines software development (Dev) and information technology operations (Ops) to shorten the systems development life cycle and provide continuous delivery with high software quality. It aims to improve the collaboration and communication between development and operations teams, leading to better and faster software delivery.



## Why do we Need Devops?

There are several benefits of having a Devops Practices over the Agile Practices, some of them are:

- Faster time-to-market: DevOps helps organizations to deliver software faster, reducing the time-to-market and enabling them to respond quickly to changing business needs and customer requirements.
- Improved collaboration and communication: DevOps encourages collaboration and communication between development and operations teams, leading to better alignment, understanding, and trust, and improving the overall efficiency and effectiveness of the software delivery process.
- Better quality and reliability: DevOps includes continuous integration, continuous delivery, and automated testing, which helps to identify and fix issues early in the development process, improving software quality and reliability.
- 
- Reduced costs: DevOps helps organizations to reduce costs by automating repetitive tasks, improving efficiency, and reducing waste and rework.
- 
- Increased agility and scalability: DevOps enables organizations to quickly and easily scale their infrastructure and applications, allowing them to respond rapidly to changing business needs and market demands.

## Problem Statement:

Create a scientific calculator program with user menu driven operations

- Square root function -  $\sqrt{x}$
- Factorial function -  $x!$
- Natural logarithm (base e) -  $\ln(x)$
- Power function -  $x^b$

## Devops Tool Chain Used:

1. Source code management: **GIT AND GITHUB**
2. Testing: **JUnit**
3. Build: **MAVEN**
4. Continuous Integration: **JENKINS**
5. Containerization: **DOCKER, DOCKER HUB**
6. Configuration Management: **ANSIBLE**
7. Deployment: **Localhost**
8. Logging: **LOG4J**
9. Monitoring: **ELK Stack.**

Programming Language Used: **JAVA**

Type of Application: **COMMAND LINE INTERFACE**

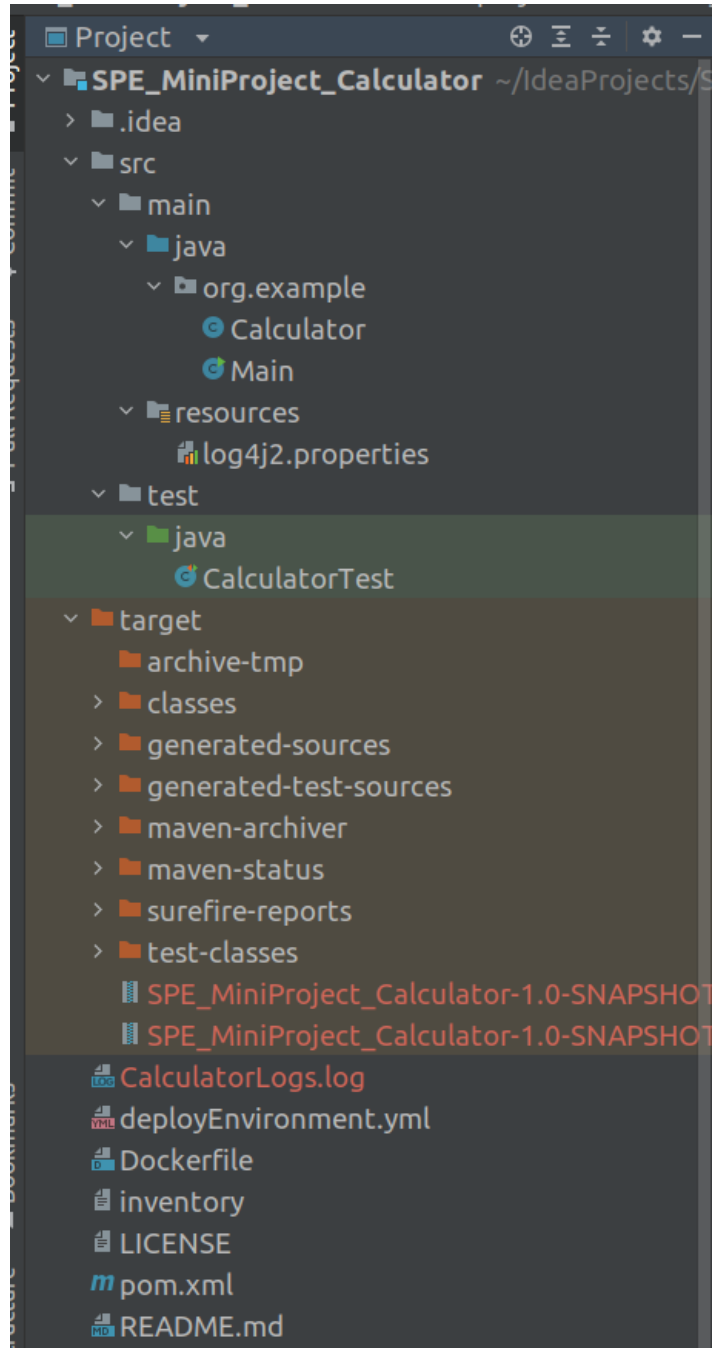
**GITHUB REPO:** [https://github.com/sbrc1996/SPE\\_Calculator](https://github.com/sbrc1996/SPE_Calculator) (master branch)

### **Steps followed in development:**

1. Start the development in local
2. Write the test cases for the application using JUnit
3. Push the code into the guthub repository via git
4. Build the entire project using Maven
5. Create the Pipeline for the application using groovy script in Jenkins
6. Create image in the Docker
7. Push the image in the Docker Hub after creating an account
8. Deploy the code in the Docker container with the help of Ansible
9. Run the code in the local machine through the help of Ansible user
10. Generate the Log file by the help of Log4j2.

## Step 1: Development of the application.

The folder structure:



### **Dependency File: POM.xml**

```
<?xml version="1.0" encoding="UTF-8"?>
<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>org.example</groupId>
  <artifactId>SPE_MiniProject_Calculator</artifactId>
  <version>1.0-SNAPSHOT</version>

  <properties>
    <maven.compiler.source>11</maven.compiler.source>
    <maven.compiler.target>11</maven.compiler.target>

<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  </properties>

  <build>
    <plugins>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-assembly-plugin</artifactId>
        <version>3.5.0</version>
        <executions>
          <execution>
            <phase>package</phase>
            <goals>
              <goal>single</goal>
            </goals>
            <configuration>
              <archive>
                <manifest>

<mainClass>org.example.Main</mainClass>
                </manifest>
              </archive>
            </configuration>
          </execution>
        </executions>
      </plugin>
    </plugins>
  </build>
</project>
```

```

        </archive>
        <descriptorRefs>

<descriptorRef>jar-with-dependencies</descriptorRef>
        </descriptorRefs>
    </configuration>
</execution>
</executions>
</plugin>
</plugins>
</build>

<!-- https://mvnrepository.com/artifact/junit/junit -->
<dependencies>
    <dependency>
        <groupId>junit</groupId>
        <artifactId>junit</artifactId>
        <version>4.13.2</version>
        <scope>test</scope>
    </dependency>
<!-- https://mvnrepository.com/artifact/log4j/log4j -->
    <dependency>
        <groupId>org.apache.logging.log4j</groupId>
        <artifactId>log4j-api</artifactId>
        <version>2.19.0</version>
    </dependency>
    <dependency>
        <groupId>org.apache.logging.log4j</groupId>
        <artifactId>log4j-core</artifactId>
        <version>2.19.0</version>
    </dependency>

</dependencies>

</project>

```

Output of the application:

```
Run: Main
/usr/lib/jvm/java-11.0-openjdk-amd64/bin/java -javaagent:/snap/intellij-idea-ultimate/418/lib/idea_rt.jar=33413:/snap/intellij-idea-ultimate/
2023-03-19 01:24:31,624 main ERROR Unable to locate appender "STDOUT" for logger config "root"
Welcome to Mini Calculator!!
Enter 1. For Addition!
Enter 2. For Subtraction!
Enter 3. For Multiply!
Enter 4. For Square Root!
Enter 5. For Factorial!
Enter 6. For Natural Logarithm!
Enter 7. For Power Function!
Enter 8. To Exit from the calculator!
Enter your choice:-
1
Enter the first number
12
Enter the second number
4
The answer for addition of 12 and 4 is 16
```

## Step 2: Writing Test Cases with JUnit:

```
CalculatorTest.java
import org.example.Calculator;
import org.junit.Test;

import static org.junit.Assert.assertEquals;
import static org.junit.Assert.assertNotEquals;

no usages 1 sbr1996
public class CalculatorTest
{
    12 usages
    Calculator cc = new Calculator();
    no usages 1 sbr1996
    @Test
    public void testAddFunction()
    {
        assertEquals( message: "Verifying the add function for True Positive: ", expected: 26, cc.add( a: 13, b: 13));
        assertEquals( message: "Verifying the add function for False Positive: ", unexpected: 22, cc.add( a: 13, b: 13));
    }
    no usages 1 sbr1996
    @Test
    public void testSubFunction()
    {
        assertEquals( message: "Verifying the add function for True Positive: ", expected: 0, cc.sub( a: 13, b: 13));
        assertEquals( message: "Verifying the add function for False Positive: ", unexpected: 22, cc.sub( a: 13, b: 13));
    }
}
```

## Step 3: SOURCE CODE MANAGEMENT

master 1 branch 0 tags

Go to file Add file <> Code

sbrc1996 Latest Commit with Log4j Files in the master branch...

213a588 yesterday 18 commits

idea	Initial Commit in the master branch....	4 days ago
src	Latest Commit with Log4j Files in the master branch...	yesterday
target/classes/org/example	Latest Commit with Log4j Files in the master branch...	yesterday
Dockerfile	Latest Commit with Log4j Files in the master branch...	yesterday
LICENSE	Initial commit	4 days ago
README.md	Create README.md	4 days ago
deployEnvironment.yml	Latest Commit with Log4j Files in the master branch...	yesterday
inventory	Fifth Commit in the master branch...	3 days ago
pom.xml	Latest Commit with Log4j Files in the master branch...	yesterday

README.md

# SPE\_Calculator

This repository is used to implement a calculator like app using Java 11. Here we show the usecase of how to build a sample project in Java and then Use a VCS like Git & GitHub. Moreover develop a CI pipeline using Jenkins and GitSCM polling to design the continous pipelines for seamless development and testing. Also we use JUnit for writing the unit test cases and Maven for building the entire project. We use Docker for the containerization purpose and Ansible for Configuration Management. We also use the ELK stack here.

### Commit History:

master

Commits on Mar 17, 2023

Latest Commit with Log4j Files in the master branch...  
sbrc1996 committed yesterday

213a588 <> Tip

Latest Commit with Log4j Files in the master branch...  
sbrc1996 committed yesterday

1152bd7 <> Tip

Latest Commit with Log4j Files in the master branch...  
sbrc1996 committed yesterday

3b3f1f2 <> Tip

Latest Commit with Log4j Files in the master branch...  
sbrc1996 committed yesterday

2967e37 <> Tip

Sixth Commit with Log4j Files in the master branch...  
sbrc1996 committed 2 days ago

2d89a02 <> Tip

Sixth Commit with Log4j Files in the master branch...  
sbrc1996 committed 2 days ago

017af77 <> Tip

Sixth Commit with Log4j Files in the master branch...  
sbrc1996 committed 2 days ago

0614aca <> Tip

Commits on Mar 16, 2023

Update deployEnvironment.yml  
sbrc1996 committed 2 days ago

Verified 8006bfa <> Tip

Commits on Mar 15, 2023

Fifth Commit in the master branch...  
sbrc1996 committed 3 days ago

e3c75b3 <> Tip

Fifth Commit in the master branch...  
sbrc1996 committed 3 days ago

ba91e8f <> Tip

Fourth Commit in the master branch...  
sbrc1996 committed 3 days ago

0860b04 <> Tip

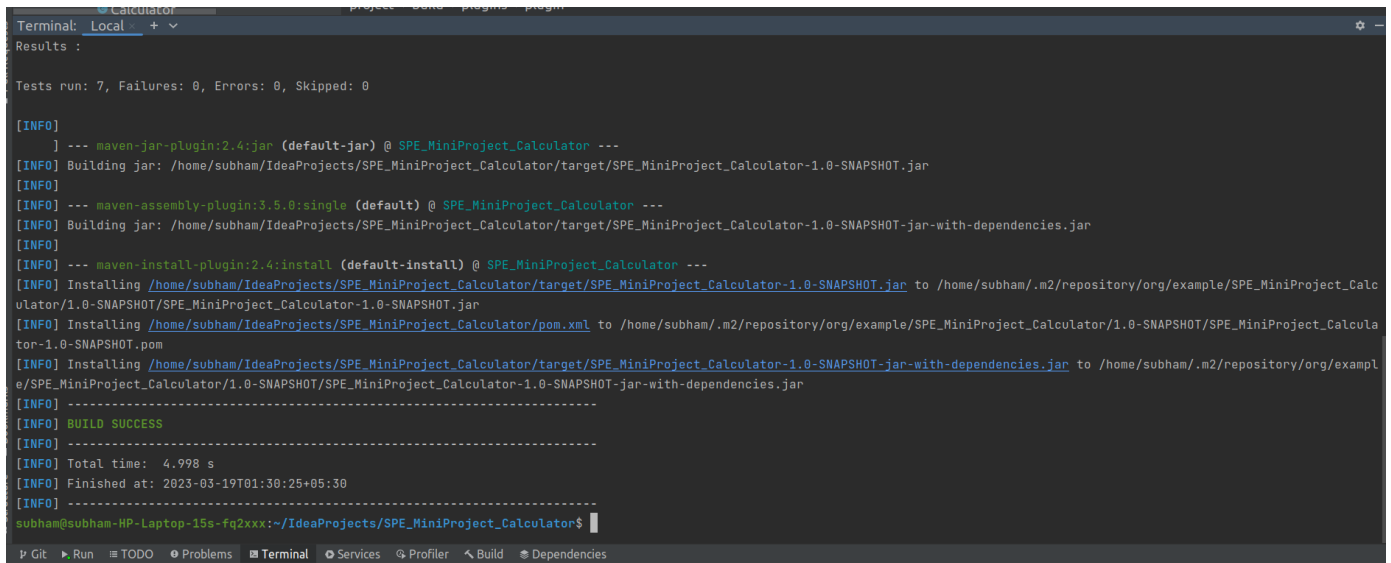


Commands used 👍

- **Git init**
- **Git add .**
- **Git remote add origin**
- **Git commit -m <Commit Message>**
- **Git push origin master**

## Step 4: Build the entire project using Maven

Command used: **mvn clean install**



```
Terminal: Local + v
Results :

Tests run: 7, Failures: 0, Errors: 0, Skipped: 0

[INFO]
] --- maven-jar-plugin:2.4:jar (default-jar) @ SPE_MiniProject_Calculator ---
[INFO] Building jar: /home/subham/IdeaProjects/SPE_MiniProject_Calculator/target/SPE_MiniProject_Calculator-1.0-SNAPSHOT.jar
[INFO]
[INFO] --- maven-assembly-plugin:3.5.0:single (default) @ SPE_MiniProject_Calculator ---
[INFO] Building jar: /home/subham/IdeaProjects/SPE_MiniProject_Calculator/target/SPE_MiniProject_Calculator-1.0-SNAPSHOT-jar-with-dependencies.jar
[INFO]
[INFO] --- maven-install-plugin:2.4:install (default-install) @ SPE_MiniProject_Calculator ---
[INFO] Installing /home/subham/IdeaProjects/SPE_MiniProject_Calculator/target/SPE_MiniProject_Calculator-1.0-SNAPSHOT.jar to /home/subham/.m2/repository/org/example/SPE_MiniProject_Calculator/1.0-SNAPSHOT/SPE_MiniProject_Calculator-1.0-SNAPSHOT.jar
[INFO] Installing /home/subham/IdeaProjects/SPE_MiniProject_Calculator/pom.xml to /home/subham/.m2/repository/org/example/SPE_MiniProject_Calculator/1.0-SNAPSHOT/SPE_MiniProject_Calculator-1.0-SNAPSHOT.pom
[INFO] Installing /home/subham/IdeaProjects/SPE_MiniProject_Calculator/target/SPE_MiniProject_Calculator-1.0-SNAPSHOT-jar-with-dependencies.jar to /home/subham/.m2/repository/org/example/SPE_MiniProject_Calculator/1.0-SNAPSHOT/SPE_MiniProject_Calculator-1.0-SNAPSHOT-jar-with-dependencies.jar
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time: 4.998 s
[INFO] Finished at: 2023-03-19T01:30:25+05:30
[INFO]
subham@subham-HP-Laptop-15s-fq2xxx:~/IdeaProjects/SPE_MiniProject_Calculator$
```

After running the above command we get a 2 JAR file created in the Target folder. We need to execute that jar file using the following command:

**java -jar SPE\_MiniProject\_Calculator-1.0-SNAPSHOT-jar-with-dependencies.jar**

To execute our CLI application:

```
subham@subham-HP-Laptop-15s-fq2xxx:~/IdeaProjects/SPE_MiniProject_Calculator$ cd target/
subham@subham-HP-Laptop-15s-fq2xxx:~/IdeaProjects/SPE_MiniProject_Calculator/target$ java -jar SPE_MiniProject_Calculator-1.0-SNAPSHOT-jar-with-dependencies.jar
2023-03-19 01:32:38,884 main ERROR Unable to locate appender "STDOUT" for logger config "root"
Welcome to Mini Calculator!!
Enter 1. For Addition!
Enter 2. For Subtraction!
Enter 3. For Multiply!
Enter 4. For Square Root!
Enter 5. For Factorial!
Enter 6. For Natural Logarithm!
Enter 7. For Power Function!
Enter 8. To Exit from the calculator!
Enter your choice:-
5
Enter the number
67
The answer for square Root of 67 is 8
```

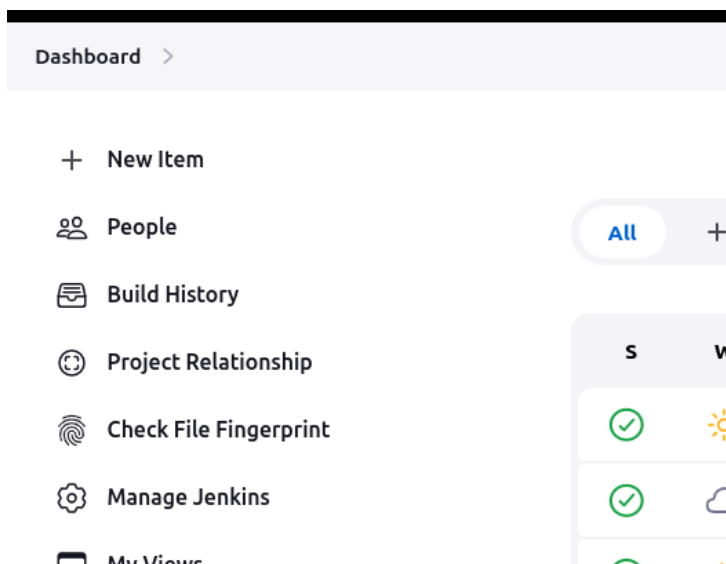
## Step 5: Pipeline Creation in Jenkins

### What is Jenkins?

Jenkins is an open-source automation server. It helps automate the parts of software development related to building, testing, and deploying, facilitating continuous integration and continuous delivery. It is a server-based system that runs in servlet containers such as Apache Tomcat.

### Steps to create a pipeline in Jenkins:


1. Install Jenkins in Ubuntu
2. Click on New Item:




3. Enter a project name and create a project of type Pipeline.

**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.

4. Our Pipeline name here is: SPE\_Calculator\_Pipeline.
5. Steps involved in our pipeline are:
  - Git Pull
  - Maven Build + Junit Scripts execution
  - Docker Image Build
  - Push Docker Image in the Docker Hub by creating a repository
  - Delete the already created Docker image
  - Ansible Deploy

We have used Groovy Scripts to write the pipeline:

Groovy script for:

- **Git Pull:**

```
8 stages {  
9     stage('Git Pull') {  
10         steps {  
11             git 'https://github.com/sbrc1996/SPE_Calculator.git'  
12         }  
13     }  
}
```

- **Maven Build + Junit Scripts execution:**

```
14 stage('Maven Build + JUnit Test Cases') {  
15     steps {  
16         sh 'mvn clean install'  
17     }  
18 }
```

- **Docker Image Build:**

```

19 stage('Docker Build Image') {
20     steps {
21         script{
22             dockering=docker.build("sbrc1996/speminiproject:latest")
23         }
24     }
25 }

```

- **Push Docker Image in the Docker Hub by creating a repository:**

```

26 stage('Push Docker Image') {
27     steps {
28         script{
29             docker.withRegistry('', 'dockerhub'){
30                 dockering.push()
31             }
32         }
33     }
34 }

```

- **Delete the already created Docker image:**

```

35 stage('Delete Docker Image') {
36     steps {
37         script{
38             sh 'docker image rm -f sbrc1996/speminiproject'
39         }
40     }
41 }

```

- **Ansible Deploy:**

```

42 stage('Ansible deploy') {
43     steps {
44         //Ansible Deploy to remote server
45         ansiblePlaybook becomeUser: null, colorized: true, disableHostKeyChecking: true, inventory: 'inventory', playbook: 'deployEnvironment.yml',
46     }
47 }
48

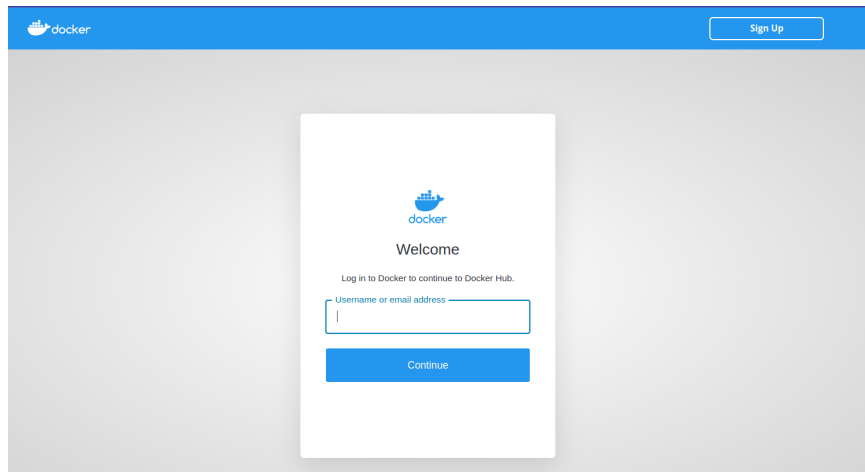
```

## Pipeline Full Run:

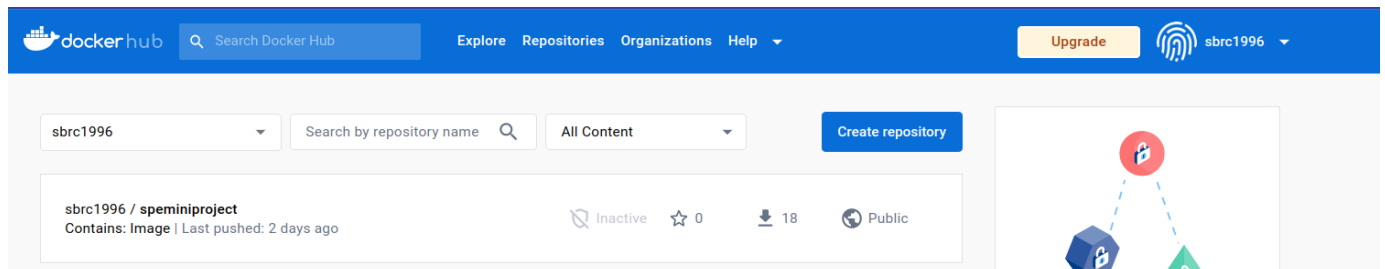
### SPE\_Calculator\_Pipeline - Stage View

			Git Pull	Maven Build + JUnit Test Cases	Docker Build Image	Push Docker Image	Delete Docker Image	Ansible deploy
Average stage times: (Average full run time: ~42s)			1s	7s	1s	16s	330ms	9s
#61	Mar 19	No Changes	1s	8s	2s	20s	444ms	13s
#60	Mar 17	No Changes	960ms	3s	698ms	23s	333ms	10s
#59	Mar 17	1 commit	1s	14s	1s	21s	349ms	8s

Before executing the entire pipeline we have to create an account on the docker Hub.



Your repository will be created here..



We have to provide sudo privileges to run our docker container:

**sudo chmod 666 /var/run/docker.sock**

## Step 6: Execution

After execution of the pipeline, we will have our docker image created and a repository created in the docker hub.

A more detailed view of the Repository:

The screenshot shows the Docker Hub repository page for 'sbrc1996/speminiproject'. The page includes a description section, a 'Tags' section with a table of image tags, and an 'Automated Builds' section. The 'Tags' table lists the 'latest' tag as an 'Image' type, pushed '2 days ago'. The 'Automated Builds' section provides instructions on how to connect to GitHub or Bitbucket for automated builds.

**Repository:** sbrc1996 / speminiproject

**Description:** This repository does not have a description. Last pushed: 2 days ago.

**Tags:** This repository contains 1 tag(s).

Tag	OS	Type	Pulled	Pushed
latest		Image	a day ago	2 days ago

**Automated Builds:** Manually pushing images to Hub? Connect your account to GitHub or Bitbucket to automatically build and tag new images whenever your code is updated, so you can focus your time on creating.

**README:** Repository description is empty. Click [here](#) to edit.

Docker Image creation:

```
subham@subham-HP-Laptop-15s-fq2xxx:~$ sudo chmod 666 /var/run/docker.sock
[sudo] password for subham:
subham@subham-HP-Laptop-15s-fq2xxx:~$ docker images
REPOSITORY          TAG          IMAGE ID          CREATED           SIZE
sbrc1996/speminiproject latest      3374256285ba     50 seconds ago   656MB
sbrc1996/speminiproject <none>      d8571eb06fd0     44 hours ago     656MB
sbrc1996/speminiproject <none>      19063e3b3458     4 days ago       656MB
sbrc1996/speminiproject <none>      22627f61c235     4 days ago       656MB
sbrc1996/speminiproject <none>      29da2144a5db     4 days ago       654MB
sbrc1996/speminiproject <none>      9493cf8dc71c     4 days ago       654MB
sbrc1996/speminiproject <none>      dfa459f1d91c     4 days ago       654MB
sbrc1996/speminiproject <none>      8bf1b7319697     4 days ago       654MB
sbrc1996/speminiproject <none>      83f57a90be1d     5 days ago       654MB
sbrc1996/speminiproject <none>      57c6bbaceb1d     5 days ago       654MB
sbrc1996/speminiproject <none>      f77f937a6f51     5 days ago       654MB
sbrc1996/speminiproject <none>      2df7bd54152d     6 days ago       654MB
sbrc1996/speminiproject <none>      47d489c8843d     6 days ago       654MB
sbrc1996/speminiproject <none>      b2435593f6c1     6 days ago       654MB
sbrc1996/speminiproject <none>      d619ffae5223     6 days ago       654MB
subham1996/speminiproject latest      d668c576ffc1     6 days ago       654MB
sbrc1996/speminiproject <none>      9d4319a285ff     6 days ago       654MB
sbrc1996/speminiproject <none>      52b958325398     6 days ago       654MB
sbrc1996/speminiproject <none>      53202ca58ce0     6 days ago       654MB
sbrc1996/speminiproject <none>      8b9f242f2992     6 days ago       654MB
sbrc1996/speminiproject <none>      939b52b36340     6 days ago       654MB
sbrc1996/speminiproject <none>      21ef7e7e891a     6 days ago       654MB
sbrc1996/speminiproject <none>      3a8c1aeaa20e     6 days ago       654MB
sbrc1996/speminiproject <none>      74c89adaee61     6 days ago       654MB
sbrc1996/speminiproject <none>      5049eedd8aa9     6 days ago       654MB
sbrc1996/speminiproject <none>      6ce43d0a50e2     6 days ago       654MB
sbrc1996/speminiproject <none>      381d08f77756     6 days ago       654MB
sbrc1996/speminiproject <none>      c9165f1e6bd2     6 days ago       654MB
sbrc1996/speminiproject <none>      a94a012e17c1     6 days ago       654MB
sbrc1996/speminiproject <none>      5f652c2ec22e     6 days ago       654MB
sbrc1996/speminiproject <none>      77209b1eda91     6 days ago       654MB
sbrc1996/speminiproject <none>      4826d798ab36     6 days ago       654MB
sbrc1996/speminiproject <none>      dba981f9a9d6     6 days ago       654MB
ubuntu               latest      58db3edaf2be     7 weeks ago      77.8MB
openjdk              11         47a932d998b7     7 months ago     654MB
subham@subham-HP-Laptop-15s-fq2xxx:~$
```

We can run this file by executing the following command

**docker run -it sbrc1996/speminiproject**

```
subham@subham-HP-Laptop-15s-fq2xxx:~$ docker run -it sbrc1996/speminiproject
2023-03-21 13:55:07,733 main ERROR Unable to locate appender "STDOUT" for logger config "root"
Welcome to Mini Calculator!!
Enter 1. For Addition!
Enter 2. For Subtraction!
Enter 3. For Multiply!
Enter 4. For Square Root!
Enter 5. For Factorial!
Enter 6. For Natural Logarithm!
Enter 7. For Power Function!
Enter 8. To Exit from the calculator!
Enter your choice:-
3
Enter the first number
23
Enter the second number
56
The answer for multiplication of 23 and 56 is 1288
Enter 1. For Addition!
Enter 2. For Subtraction!
Enter 3. For Multiply!
Enter 4. For Square Root!
Enter 5. For Factorial!
Enter 6. For Natural Logarithm!
Enter 7. For Power Function!
Enter 8. To Exit from the calculator!
Enter your choice:-
7
Enter the number
9
Enter the power
5
The answer for Power of 9^5 is 59049
Enter 1. For Addition!
Enter 2. For Subtraction!
Enter 3. For Multiply!
Enter 4. For Square Root!
Enter 5. For Factorial!
Enter 6. For Natural Logarithm!
Enter 7. For Power Function!
Enter 8. To Exit from the calculator!
Enter your choice:-
8
Thanks for visiting, see you soon!!!
subham@subham-HP-Laptop-15s-fq2xxx:~$
```

## Step 7: Deployment.

We are not deploying our calculator code into any cloud environment, instead, we are deploying it in our localhost using Ansible.

### What is Ansible?

**Ansible** is a suite of software tools that enables infrastructure as code. It is open-source and the suite includes software provisioning, configuration management, and application deployment functionality. Originally written by Michael DeHaan and acquired by Red Hat in 2015, Ansible is designed to configure both Unix-like systems and Microsoft Windows. Ansible is agentless, relying on temporary remote connections via SSH or Windows Remote Management which allows PowerShell execution. The Ansible control node runs on most Unix-like systems that are able to run Python, including Windows with Windows Subsystem for Linux installed. System configuration is defined in part by using its own declarative language.

#### Steps needed for a successful deployment:

- Install Ansible
- Create a dedicated Ansible user on your own computer.
- Generate SSH key which will be used in the authentication.
- Finally, write the YAML playbook file to install the required dependencies in the docker container.

#### Playbook:

```
13 lines (10 sloc) | 240 Bytes
1 ---
2 - name: Pull docker image of Calculator
3   hosts: all
4   tasks:
5
6     - name: Start docker service
7       service:
8         name: docker
9         state: started
10
11    - name: pull docker image
12      shell: docker pull sbrc1996/speminiproject
13
```

#### Inventory File:

```
2 lines (2 sloc) | 51 Bytes
1 [ubuntu18]
2 172.16.133.72 ansible_user="ansible_usr"
```



Here Ubuntu18 is the group name, the ip-address is the one of my my own computer and the name of the user through which I am going to deploy the code in the container.

Running in the docker container using ansible user: "ansible\_usr"

- Login in as sudo ansible user
- Check the docker images
- Run the docker container

Screenshots for the same:

```
subham@subham-HP-Laptop-15s-fq2xxx:~$  
subham@subham-HP-Laptop-15s-fq2xxx:~$ sudo su ansible_usr  
[sudo] password for subham:  
ansible_usr@subham-HP-Laptop-15s-fq2xxx:/home/subham$ docker images  
REPOSITORY          TAG             IMAGE ID        CREATED         SIZE  
sbrc1996/speminiproject latest        3374256285ba   29 minutes ago  656MB  
sbrc1996/speminiproject <none>       d8571eb06fd0   44 hours ago    656MB  
sbrc1996/speminiproject <none>       19063e3b3458   4 days ago      656MB  
sbrc1996/speminiproject <none>       22627f61c235   4 days ago      656MB  
sbrc1996/speminiproject <none>       29da2144a5db   4 days ago      654MB  
sbrc1996/speminiproject <none>       9493cf8dc71c   4 days ago      654MB  
sbrc1996/speminiproject <none>       dfa459f1d91c   4 days ago      654MB  
sbrc1996/speminiproject <none>       8bf1b7319697   4 days ago      654MB  
sbrc1996/speminiproject <none>       83f57a90be1d   5 days ago      654MB  
sbrc1996/speminiproject <none>       57c6bbaceb1d   5 days ago      654MB  
sbrc1996/speminiproject <none>       f77f937a6f51   5 days ago      654MB  
sbrc1996/speminiproject <none>       2df7bd54152d   6 days ago      654MB  
<none>              <none>       47d489c8843d   6 days ago      654MB
```

```
openjdk             11             47a932d998b7   7 months ago    654MB  
ansible_usr@subham-HP-Laptop-15s-fq2xxx:/home/subham$ docker ps  
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES  
ansible_usr@subham-HP-Laptop-15s-fq2xxx:/home/subham$ docker ps -a  
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES  
23355a41eac5       sbrc1996/speminiproject "java -jar SPE_Minip..." 27 minutes ago     Exited (0) 26 minutes ago           funny_kalam  
77ab2bad1212       d8571eb06fd0       "java -jar SPE_Minip..." 31 hours ago       Exited (129) 31 hours ago           minicalc  
e8d07a1a7ff5       d8571eb06fd0       "java -jar SPE_Minip..." 44 hours ago       Exited (143) 42 hours ago           xyz  
cdbc4250ecda       190               "java -jar SPE_Minip..." 4 days ago         Exited (0) 4 days ago               musing_wozniak  
4c21ab285534       19063e3b3458       "java -jar SPE_Minip..." 4 days ago         Exited (0) 4 days ago               priceless_galois  
cd84be51935c       226               "java -jar SPE_Minip..." 4 days ago         Exited (0) 4 days ago               speminiproject  
d307127edbf6       dfa459f1d91c       "java -jar SPE_Minip..." 4 days ago         Exited (0) 4 days ago               cranky_colden  
d0b90362d040       dfa459f1d91c       "java -jar SPE_Minip..." 4 days ago         Exited (130) 4 days ago             crazy_archimedes  
a0dec8d71a6c       57c6bbaceb1d       "java -jar SPE_Minip..." 5 days ago         Exited (130) 5 days ago             great_goodall  
c183829392be       f77f937a6f51       "java -jar SPE_Minip..." 5 days ago         Exited (143) 5 days ago             upbeat_germain  
3c40dd828c7a       ubuntu            "/bin/bash"         6 days ago         Created                                test1  
48d959370b81       ubuntu            "/bin/bash"         6 days ago         Created                                test  
7774e2a2c057       ubuntu            "/bin/bash"         3 weeks ago        Exited (137) 3 weeks ago            user10  
2bdb38d03dad       ubuntu            "/bin/bash"         3 weeks ago        Exited (137) 3 weeks ago            user9  
53630d5a231e       ubuntu            "/bin/bash"         3 weeks ago        Exited (137) 3 weeks ago            user8  
bc1ef4d9b2f8       ubuntu            "/bin/bash"         3 weeks ago        Exited (137) 3 weeks ago            user7  
ad74ed162fe1       ubuntu            "/bin/bash"         3 weeks ago        Exited (137) 3 weeks ago            user6  
ef3529536654       ubuntu            "/bin/bash"         3 weeks ago        Exited (137) 3 weeks ago            user5  
a561d58973a0       ubuntu            "/bin/bash"         3 weeks ago        Exited (137) 3 weeks ago            user4  
fb4a7658c953       ubuntu            "/bin/bash"         3 weeks ago        Exited (127) 3 weeks ago            user3
```

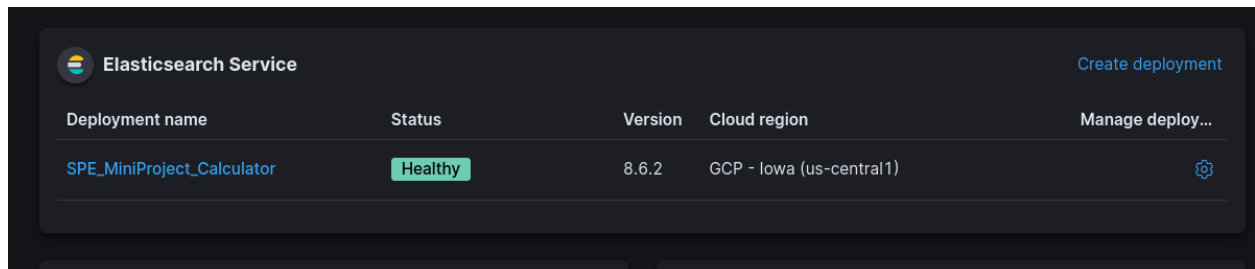
```
ansible_usr@subham-HP-Laptop-15s-fq2xxx:/home/subham$ docker run -it sbrc1996/speminiproject
2023-03-21 14:23:25,599 main ERROR Unable to locate appender "STDOUT" for logger config "root"
Welcome to Mini Calculator!!
Enter 1. For Addition!
Enter 2. For Subtraction!
Enter 3. For Multiply!
Enter 4. For Square Root!
Enter 5. For Factorial!
Enter 6. For Natural Logarithm!
Enter 7. For Power Function!
Enter 8. To Exit from the calculator!
Enter your choice:-
```


Code executing successfully in the docker container 23355a41eac5.

## Step 8: Continuous Monitoring

We will perform continuous monitoring using the ELK stack. We have 2 options with the ELK stack either Install it locally or use the cloud version for a 14-day trail. I manually installed the log file from the Jenkins workspace in the docker container and uploaded it in the ELK dashboard to monitor the file.

Screenshots:



Elasticsearch Service <a href="#">Create deployment</a>				
Deployment name	Status	Version	Cloud region	Manage deploy...
SPE_MiniProject_Calculator	Healthy	8.6.2	GCP - Iowa (us-central1)	

Import the log file from the Local:

## More ways to add data

In addition to adding [integrations](#), you can try our sample data or upload your own data.

[Sample data](#)[Upload file](#)

### CalculatorLogs.log

#### File contents

First 24 lines

```
1 [INFO ] 2023-03-19 18:18:26.547 [main] Main -
2 Start of Logging...
3 [ERROR] 2023-03-19 18:18:26.550 [main] Main -
4 This is an test error message...
5 [FATAL] 2023-03-19 18:18:26.550 [main] Main -
6 This is an test fatal massage...
7 [DEBUG] 2023-03-19 18:18:26.550 [main] Main -
8 Control now in the main loop...
9 [DEBUG] 2023-03-19 18:18:34.577 [main] Main -
10 Control with the addition function....
11 [DEBUG] 2023-03-19 18:18:37.578 [main] Main -
12 Control now in the main loop...
13 [DEBUG] 2023-03-19 18:18:41.222 [main] Main -
14 Control with the Power function....
```

#### Summary

Number of lines analyzed	24
Format	semi_structured_text

[Import](#)[Cancel](#)

View the logs in the Discover dashboard:

1 field sorted

<input checked="" type="checkbox"/>	@timestamp	Document
<input checked="" type="checkbox"/>	Mar 19, 2023 @ 18:21:12.393	@timestamp Mar 19, 2023 @ 18:21:12.393 log.level ERROR message [ERROR] 2023-03-19 18:21:12.393 [main] Main - This is an test error message... _id Ex1ZB1cBZjsJyq6kLQkB _index ddd _score -
<input checked="" type="checkbox"/>	Mar 19, 2023 @ 18:21:12.393	@timestamp Mar 19, 2023 @ 18:21:12.393 log.level FATAL message [FATAL] 2023-03-19 18:21:12.393 [main] Main - This is an test fatal massage... _id FB1ZB1cBZjsJyq6kLQkB _index ddd _score -
<input checked="" type="checkbox"/>	Mar 19, 2023 @ 18:21:12.393	@timestamp Mar 19, 2023 @ 18:21:12.393 log.level DEBUG message [DEBUG] 2023-03-19 18:21:12.393 [main] Main - Control now in the main loop.... _id FR1ZB1cBZjsJyq6kLQkB _index ddd _score -
<input checked="" type="checkbox"/>	Mar 19, 2023 @ 18:21:12.389	@timestamp Mar 19, 2023 @ 18:21:12.389 log.level INFO message [INFO ] 2023-03-19 18:21:12.389 [main] Main - Start of Logging... _id Eh1ZB1cBZjsJyq6kLQkB _index ddd _score -
<input checked="" type="checkbox"/>	Mar 19, 2023 @ 18:18:48.930	@timestamp Mar 19, 2023 @ 18:18:48.930 log.level DEBUG message [DEBUG] 2023-03-19 18:18:48.930 [main] Main - See you later... _id ER1ZB1cBZjsJyq6kLQkB _index ddd _score -
<input checked="" type="checkbox"/>	Mar 19, 2023 @ 18:18:44.786	@timestamp Mar 19, 2023 @ 18:18:44.786 log.level DEBUG message [DEBUG] 2023-03-19 18:18:44.786 [main] Main - Control now in the main loop.... _id EB1ZB1cBZjsJyq6kLQkB _index ddd _score -
<input checked="" type="checkbox"/>	Mar 19, 2023 @ 18:18:41.222	@timestamp Mar 19, 2023 @ 18:18:41.222 log.level DEBUG message [DEBUG] 2023-03-19 18:18:41.222 [main] Main - Control with the Power function.... _id Dx1ZB1cBZjsJyq6kLQkB _index ex ddd _score -
<input checked="" type="checkbox"/>	Mar 19, 2023 @ 18:18:37.578	@timestamp Mar 19, 2023 @ 18:18:37.578 log.level DEBUG message [DEBUG] 2023-03-19 18:18:37.578 [main] Main - Control now in the main loop.... _id Dh1ZB1cBZjsJyq6kLQkB _index ddd _score -
<input checked="" type="checkbox"/>	Mar 19, 2023 @ 18:18:34.577	@timestamp Mar 19, 2023 @ 18:18:34.577 log.level DEBUG message [DEBUG] 2023-03-19 18:18:34.577 [main] Main - Control with the addition function.... _id DR1ZB1cBZjsJyq6kLQkB _index ddd _score -
<input checked="" type="checkbox"/>	Mar 19, 2023 @ 18:18:26.550	@timestamp Mar 19, 2023 @ 18:18:26.550 log.level ERROR message [ERROR] 2023-03-19 18:18:26.550 [main] Main - This is an test error message... _id Ch1ZB1cBZjsJyq6kLQkB _index ddd _score -
<input checked="" type="checkbox"/>	Mar 19, 2023 @ 18:18:26.550	@timestamp Mar 19, 2023 @ 18:18:26.550 log.level FATAL message [FATAL] 2023-03-19 18:18:26.550 [main] Main - This is an test fatal massage... _id Cx1ZB1cBZjsJyq6kLQkB _index ddd _score -
<input checked="" type="checkbox"/>	Mar 19, 2023 @ 18:18:26.550	@timestamp Mar 19, 2023 @ 18:18:26.550 log.level DEBUG message [DEBUG] 2023-03-19 18:18:26.550 [main] Main - Control now in the main loop.... _id DB1ZB1cBZjsJyq6kLQkB _index ddd _score -

Rows per page: 100

Commands used in the ELK Stack:

- `sudo su ansible_usr`
- `docker images`
- `docker run -it --name xyz sbrc1996/speminiproject`
- `docker ps`
- `docker start xyz`
- `sudo docker cp xyz:CalculatorLogs.log /home/subham/Desktop/`

## Challenges Faced:

- While installing the Ansible the config file was not available , hence I had to uninstall the entire Ansible and reinstall from scratch.

Resources followed for the fix:

<https://www.cyberciti.biz/faq/how-to-install-and-configure-latest-version-of-ansible-on-ubuntu-linux/>

For locale issue run:

`export LC_ALL=en_IN.UTF-8`

- Error in the 3rd stage of pipeline stating Docker Daemon socket: Got permission denied.

Fix: `sudo chmod 666 /var/run/docker.sock`

We need to run this command everytime we restart the server machine.

- On running the command `mvn clean install` I got the following error:

```
PLAY RECAP *****
[0;31m172.16.133.72[0m : [0;32mok=3 [0m [0;33mchanged=1 [0m [0m unreachable=0 [0;31mfailed=1 [0m [0m skipped=0 rescued=0 ignored=0

FATAL: command execution failed
hudson.AbortException: Ansible playbook execution failed
    at org.jenkinsci.plugins.ansible.AnsiblePlaybookBuilder.perform(AnsiblePlaybookBuilder.java:262)
    at org.jenkinsci.plugins.ansible.workflow.AnsiblePlaybookSteps$AnsiblePlaybookExecution.run(AnsiblePlaybookStep.java:430)
    at org.jenkinsci.plugins.ansible.workflow.AnsiblePlaybookSteps$AnsiblePlaybookExecution.run(AnsiblePlaybookStep.java:351)
    at org.jenkinsci.plugins.workflow.steps.AbstractSynchronousNonBlockingStepExecution$1$1.call(AbstractSynchronousNonBlockingStepExecution.java:47)
    at hudson.security.ACL.impersonate2(ACL.java:451)
    at hudson.security.ACL.impersonate(ACL.java:463)
    at org.jenkinsci.plugins.workflow.steps.AbstractSynchronousNonBlockingStepExecution$1.run(AbstractSynchronousNonBlockingStepExecution.java:44)
    at java.base/java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:515)
    at java.base/java.util.concurrent.FutureTask.run(FutureTask.java:264)
    at java.base/java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1128)
    at java.base/java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:628)
    at java.base/java.lang.Thread.run(Thread.java:829)

[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
ERROR: Ansible playbook execution failed
Finished: FAILURE
```

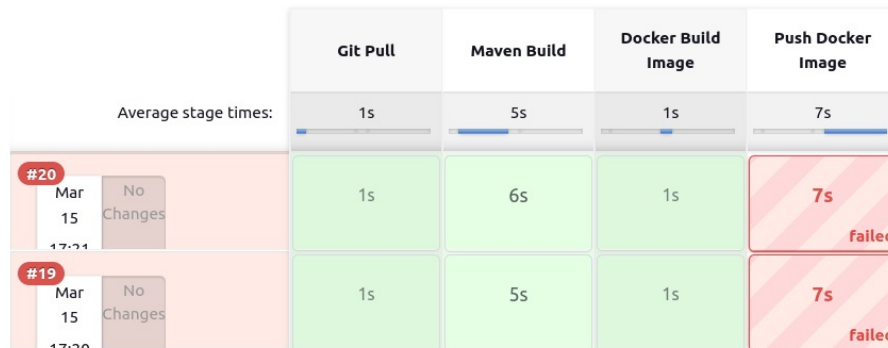
The fix to the problem was to change the dependencies added in the pom.xml file.

- Error in the Push Docker Image stage in the pipeline:

## Pipeline SPE\_Calculator\_Pipeline

This is a pipeline to depict the CI and CD for our calculator mini Project related to SPE.

### Stage View



### Terminal

```

Login Succeeded
[Pipeline] {
[Pipeline] isUnix
[Pipeline] withEnv
[Pipeline] {
[Pipeline] sh
+ docker tag subham1996/speminiproject:latest subham1996/speminiproject:latest
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] isUnix
[Pipeline] withEnv
[Pipeline] {
[Pipeline] sh
+ docker push subham1996/speminiproject:latest
The push refers to repository [docker.io/subham1996/speminiproject]
d59b3e7affa2: Preparing
7b7f3078e1db: Preparing
826c3ddb29c: Preparing
b626401ef603: Preparing
9b55156abf26: Preparing
293d5db30c9f: Preparing
03127cdb479b: Preparing
9c742cd6c7a5: Preparing
03127cdb479b: Waiting
293d5db30c9f: Waiting
9c742cd6c7a5: Waiting
denied: requested access to the resource is denied
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }

```

Fix: in the pipeline groovy script I changed the name from subham1996/speminiproject:latest -> sbrc1996/speminiproject:latest  
Here sbrc1996 is the login id of my DockerHub.com