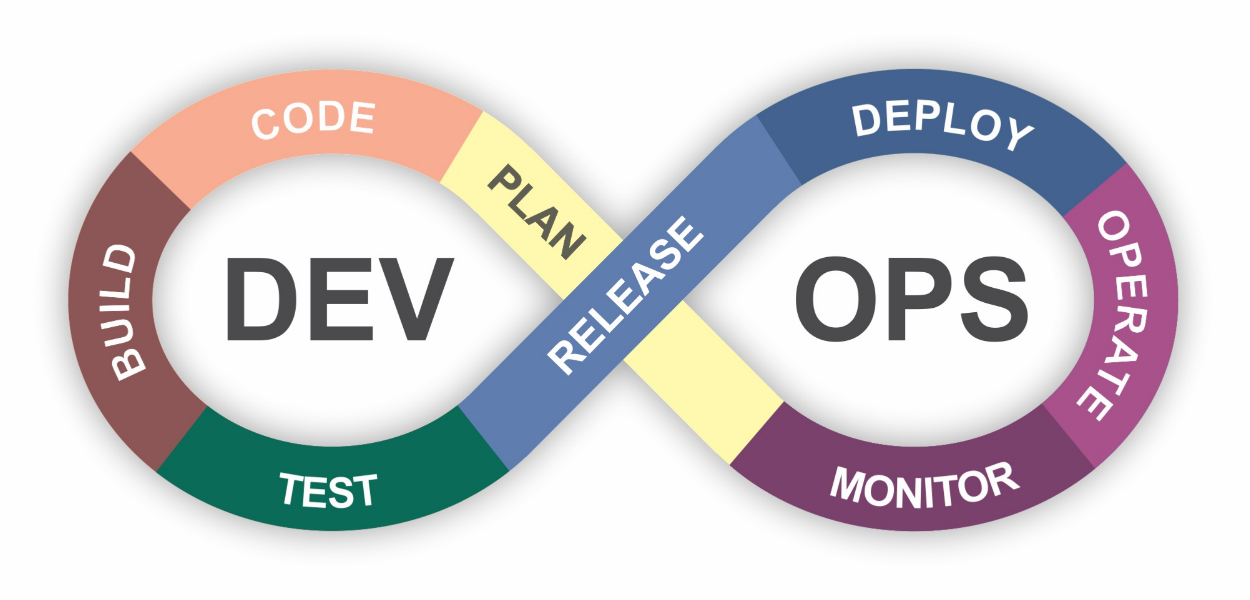
CS 816 Software Production Engineering

Mini Project

Scientific Calculator using DevOps Tools



Submitted By:

Khushal Abrol
MT2021063
M.Tech CSE 2021-23

Table of Content:

Problem Statement:

Create a scientific calculator program with user menu driven operations

- Square root function - √x
- Factorial function – x!
- Natural logarithm (base e) - ln(x)
- Power function - x

Why DevOps?

- DevOps reduce the load of various frequently occurring procedures from developers and operations teams.
- Increase collaboration in all involved parties i.e, developers, testers, operations etc.
- It helps increase deployment frequency.
- Failure rate of new releases is typically less.
- Recovery from failure is very fast.

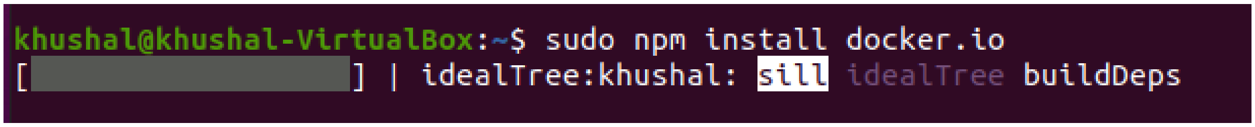
DevOps tool chain used

- **Version Control System**: GitHub
- **Testing**: Jasmine Test Framework
- **Build**: NodeJs
- **Continuous Integration**: Jenkins
- **Containerize**: Docker
- **Pushed docker image**: Docker hub
- **Deployment:** Ansible
- **Deployment Node:** Ubuntu 18.04 server
- **Monitoring**: ELK stack

Step 1: Setting Up Environment (Ubuntu)

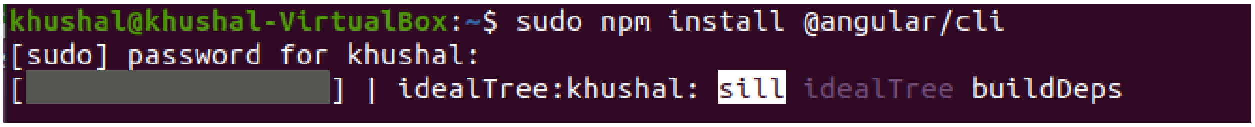
Firstly we need to set up a Jenkins Master environment. We need to install following packages/tools:

- **Install nodejs 16.04**
  - sudo npm install nodejs



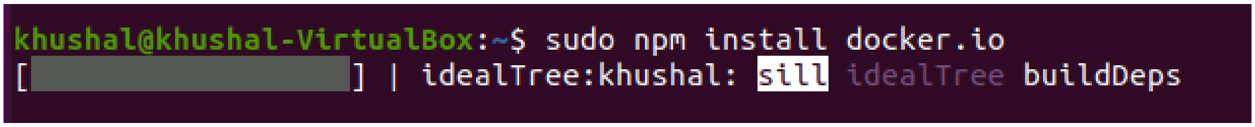
```
khushal@khushal-VirtualBox:~$ sudo npm install docker.io
[                  ] | idealTree:khushal: sill idealTree buildDeps
```

- **Install @angular/cli**
  - sudo npm install @angular/cli



```
khushal@khushal-VirtualBox:~$ sudo npm install @angular/cli
[sudo] password for khushal:
[                  ] | idealTree:khushal: sill idealTree buildDeps
```

- **Install Docker**
  - sudo npm install docker.io

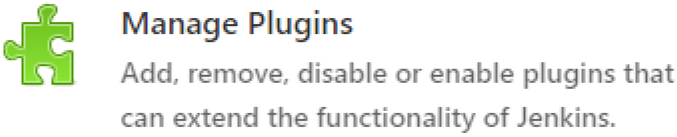


```
khushal@khushal-VirtualBox:~$ sudo npm install docker.io
[                  ] | idealTree:khushal: sill idealTree buildDeps
```

Step 2: Configure Jenkins

We need to configure our jenkins master with the following plugins. First we need to add plugins and then configure them. To manage and use various tools in our project. We also need to add our credentials of github and docker hub so that Jenkins Master will be able to access the remote repositories and can work with them.

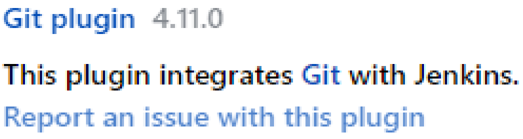
- **Install Plugins**



Manage Plugins

Add, remove, disable or enable plugins that can extend the functionality of Jenkins.

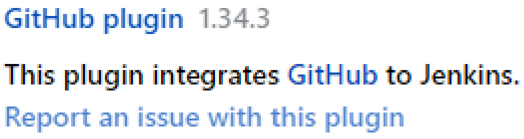
Install the following plugins from Manage Jenkins > Manage Plugins > Available



Git plugin 4.11.0

This plugin integrates Git with Jenkins.

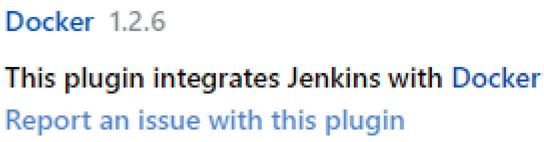
Report an issue with this plugin



GitHub plugin 1.34.3

This plugin integrates GitHub to Jenkins.

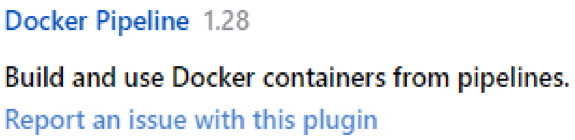
Report an issue with this plugin



Docker 1.2.6

This plugin integrates Jenkins with Docker

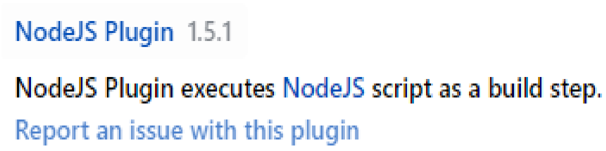
Report an issue with this plugin



Docker Pipeline 1.28

Build and use Docker containers from pipelines.

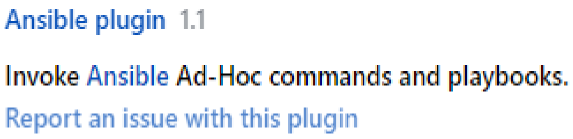
Report an issue with this plugin



NodeJS Plugin 1.5.1

NodeJS Plugin executes NodeJS script as a build step.

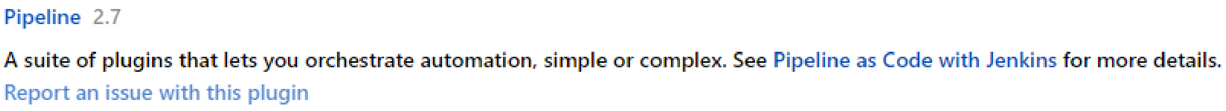
Report an issue with this plugin



Ansible plugin 1.1

Invoke Ansible Ad-Hoc commands and playbooks.

Report an issue with this plugin

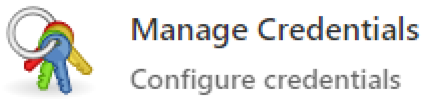


Pipeline 2.7

A suite of plugins that lets you orchestrate automation, simple or complex. See Pipeline as Code with Jenkins for more details.

Report an issue with this plugin

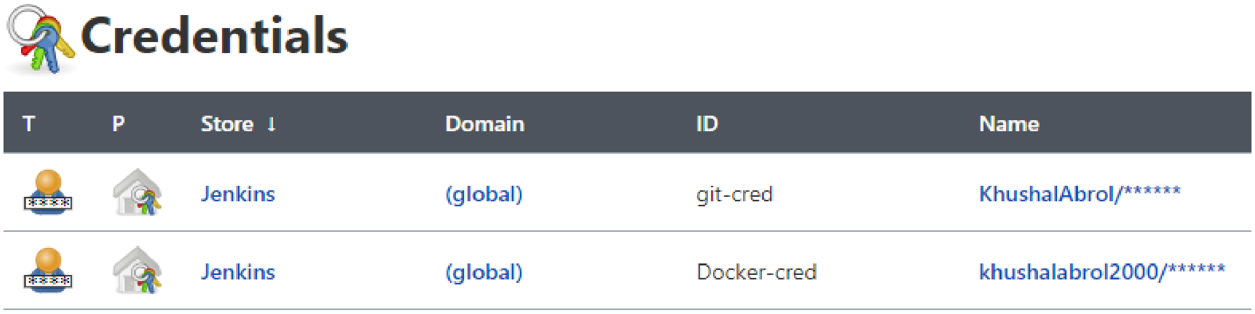
- **Add Credentials**



Manage Credentials

Configure credentials

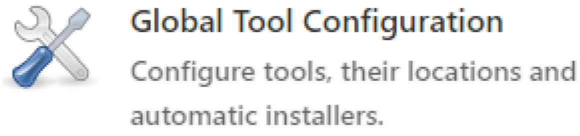
Enter Credentials of github and dockerhub in Manage Jenkins > Manage Credentials > Add Credentials



Credentials

T	P	Store ↓	Domain	ID	Name
		Jenkins	(global)	git-cred	KhushalAbrol/******
		Jenkins	(global)	Docker-cred	khushalabrol2000/******

- **Configure Plugins**

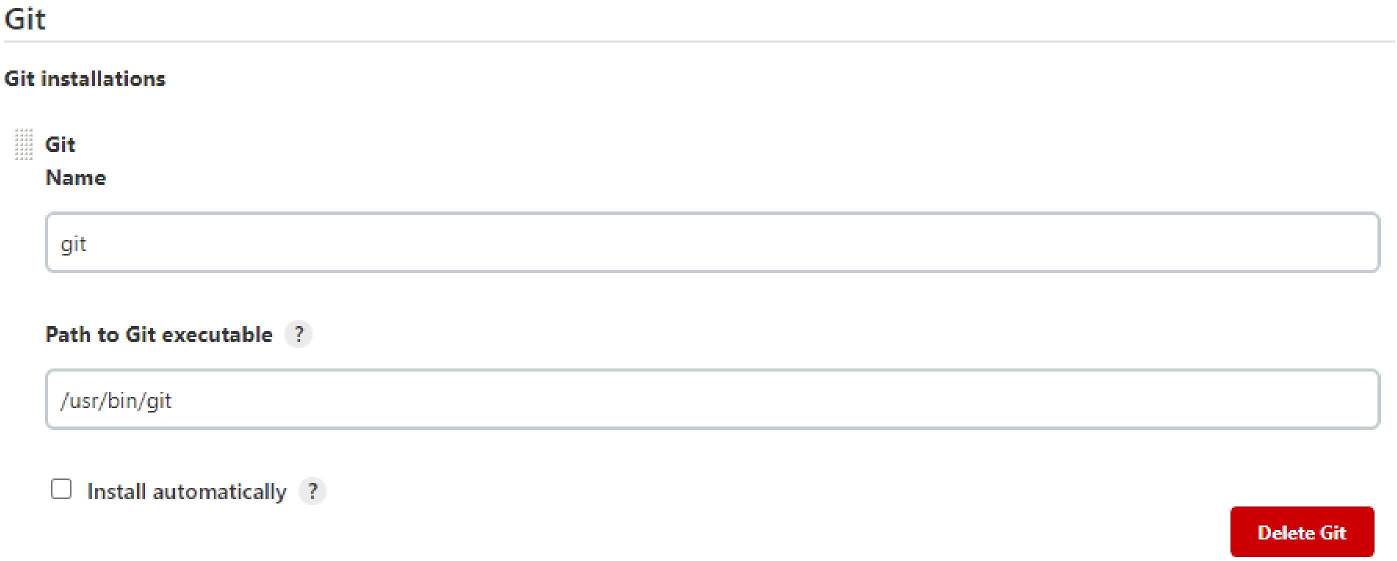


Global Tool Configuration

Configure tools, their locations and automatic installers.

Install/Configure various tools to be used:

➢ Git



Git

Git installations

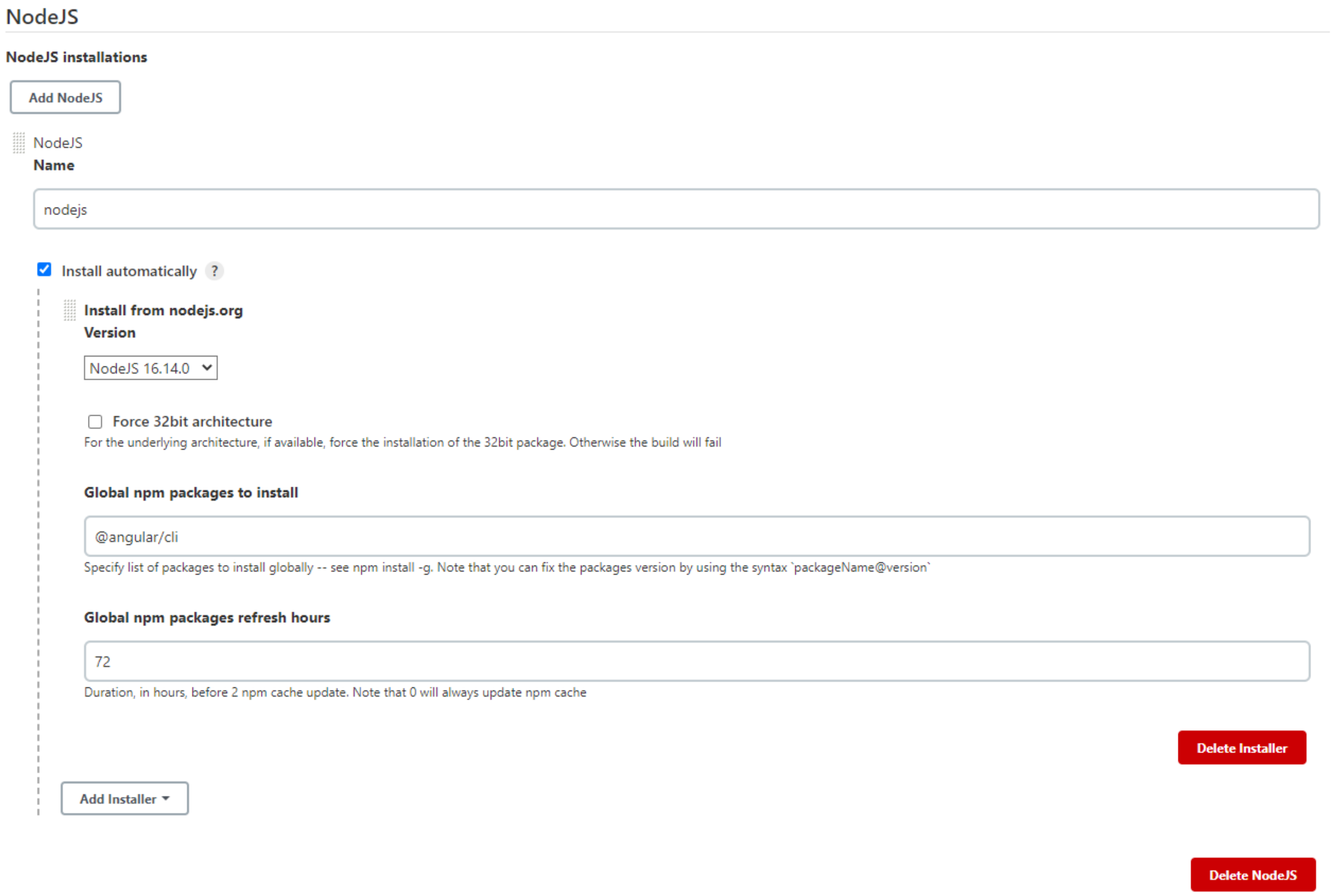
Git

Name: git

Path to Git executable: /usr/bin/git

□ Install automatically

- ➢ NodeJS



NodeJS

NodeJS installations

NodeJS

Name

nodejs

☑ Install automatically ?

Install from nodejs.org

Version

NodeJS 16.14.0

□ Force 32bit architecture

For the underlying architecture, if available, force the installation of the 32bit package. Otherwise the build will fail

Global npm packages to install

@angular/cli

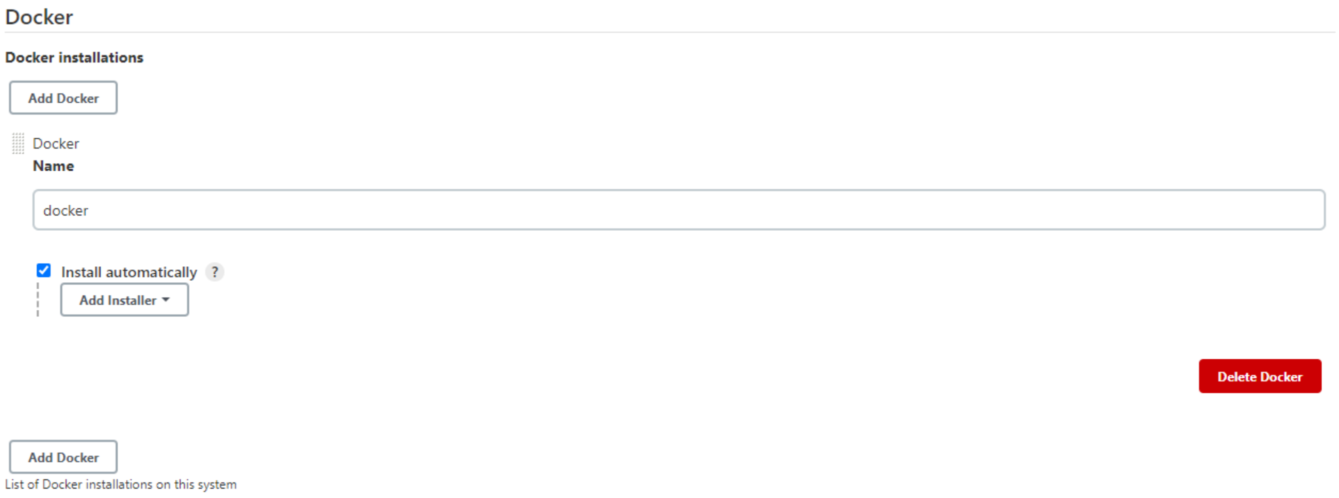
Specify list of packages to install globally -- see npm install -g. Note that you can fix the packages version by using the syntax `packageName@version`

Global npm packages refresh hours

72

Duration, in hours, before 2 npm cache update. Note that 0 will always update npm cache

- ➢ Docker



Docker

Docker installations

Docker

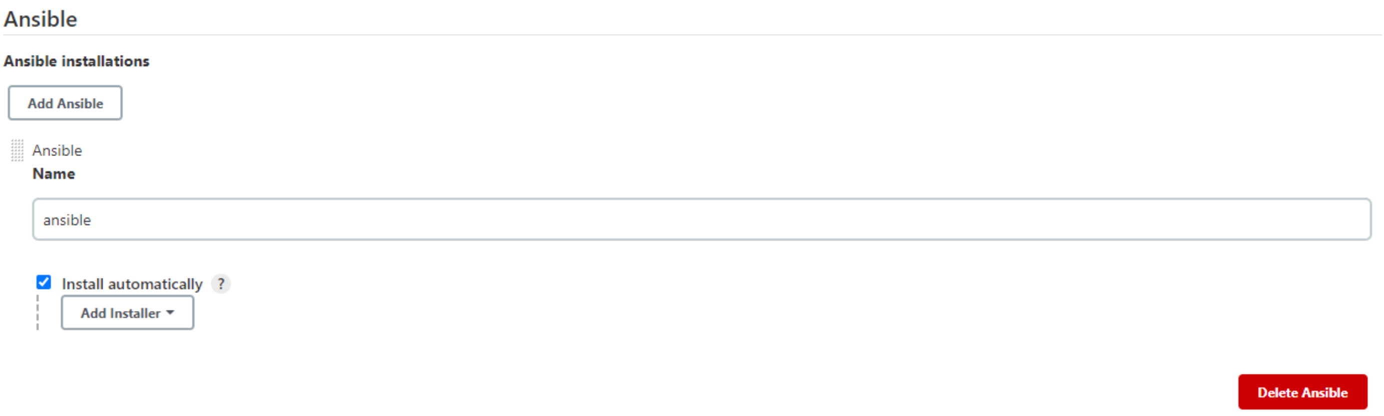
Name

docker

☑ Install automatically ?

List of Docker installations on this system

- Ansible



Ansible

Ansible installations

Add Ansible

Ansible

Name

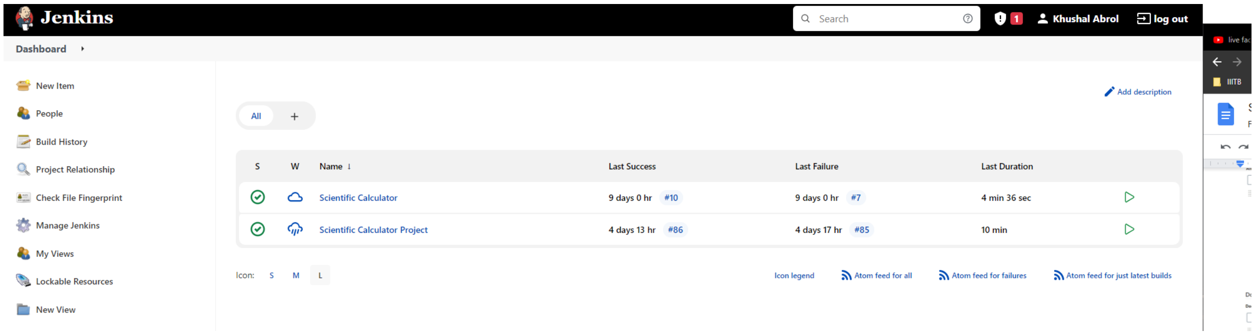
ansible

☑ Install automatically ?

Add Installer

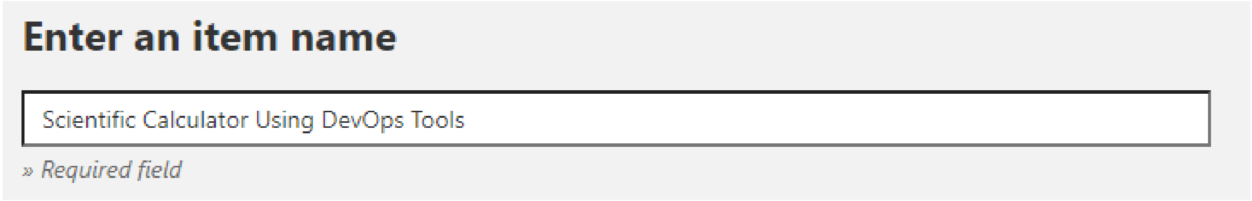
Delete Ansible

- **Creating Jenkins Project**
  Create new Jenkins Pipeline Project from New Item > Pipeline



S	W	Name ↓	Last Success	Last Failure	Last Duration
✓	☁	Scientific Calculator	9 days 0 hr #10	9 days 0 hr #7	4 min 36 sec
✓	☂	Scientific Calculator Project	4 days 13 hr #86	4 days 17 hr #85	10 min

Fig: Jenkins Dashboard

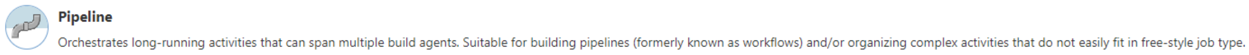


Enter an item name

Scientific Calculator Using DevOps Tools

» Required field

Fig: Entering Project Name

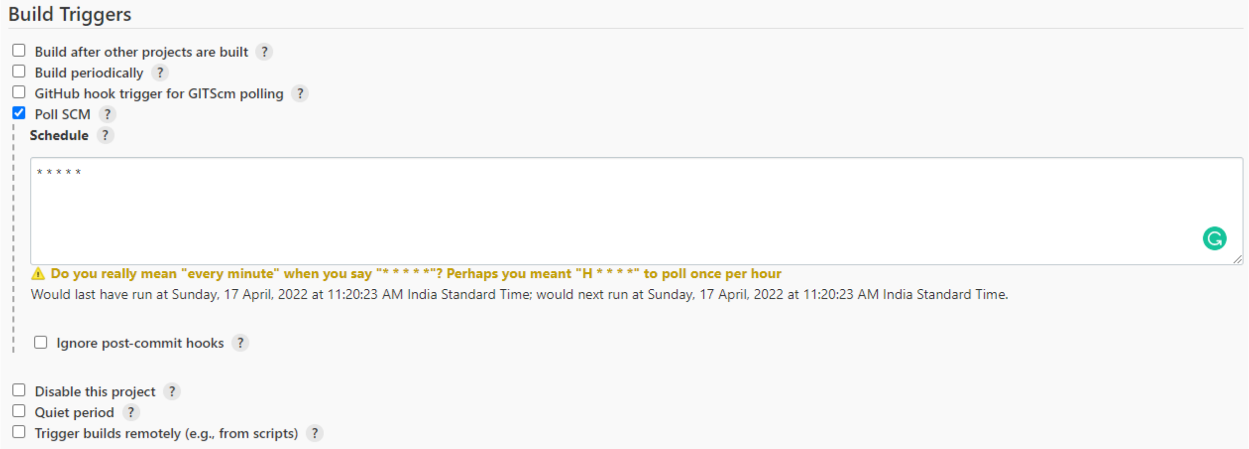


Pipeline

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.

Fig: Selecting Pipeline Project

- **Configuring Jenkins Pipeline**
  Setting **Build Triggers** as SCM Polling and **Schedule** as * * * * * (every minute).
  By setting this Jenkins Master will check github every minute periodically for any changes. If it finds any new commits it will start to run the pipeline and will build, test, deploy the new version of the app automatically.



Build Triggers

□ Build after other projects are built ?
□ Build periodically ?
□ GitHub hook trigger for GITScm polling ?
☑ Poll SCM ?

Schedule ?

```
* * * * *
```

⚠ Do you really mean "every minute" when you say "* * * * *"? Perhaps you meant "H * * * *" to poll once per hour

Would last have run at Sunday, 17 April, 2022 at 11:20:23 AM India Standard Time; would next run at Sunday, 17 April, 2022 at 11:20:23 AM India Standard Time.

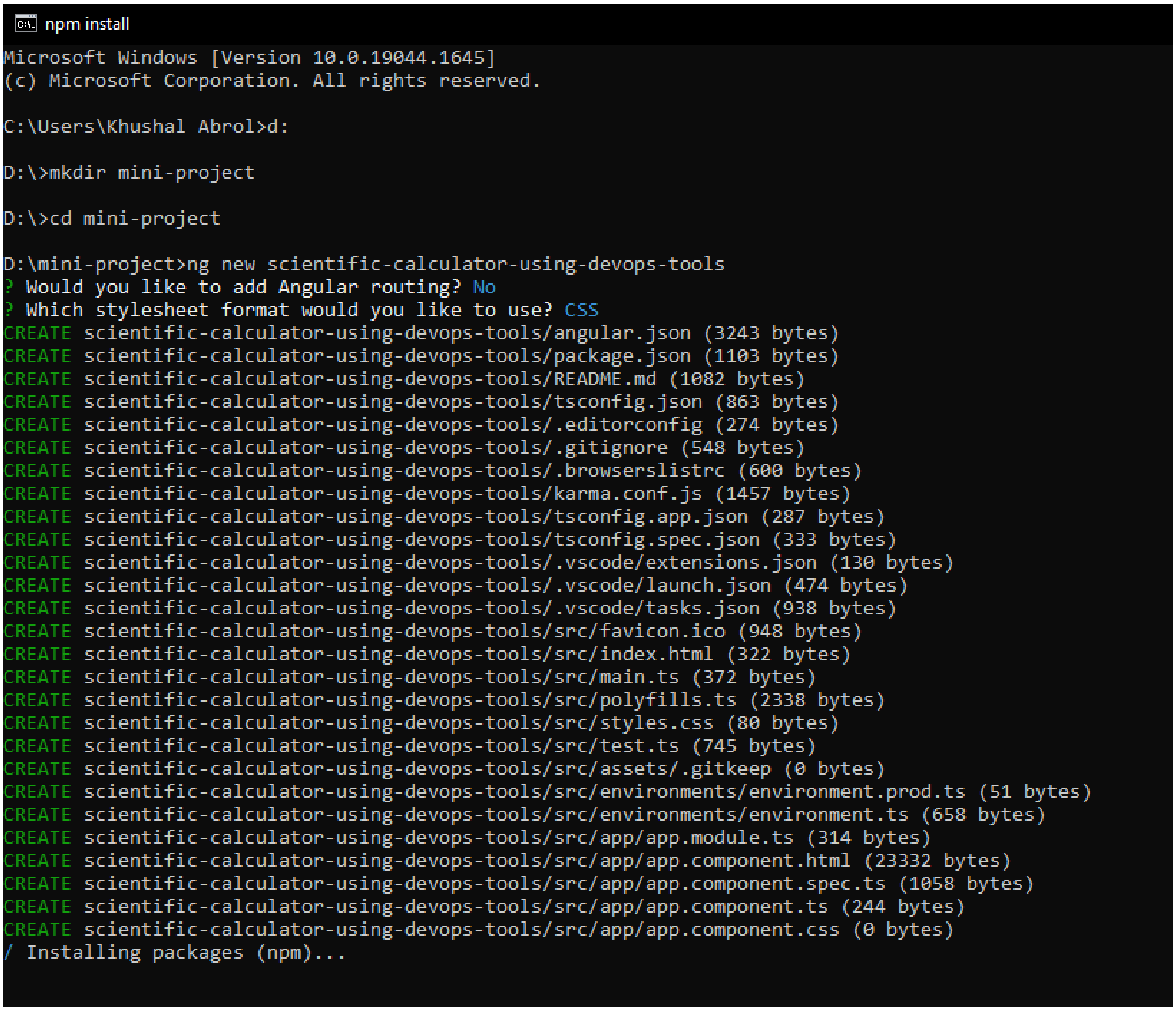
□ Ignore post-commit hooks ?

□ Disable this project ?
□ Quiet period ?
□ Trigger builds remotely (e.g., from scripts) ?

Fig: Build Trigger

Step 3: Setting Up Other Tools

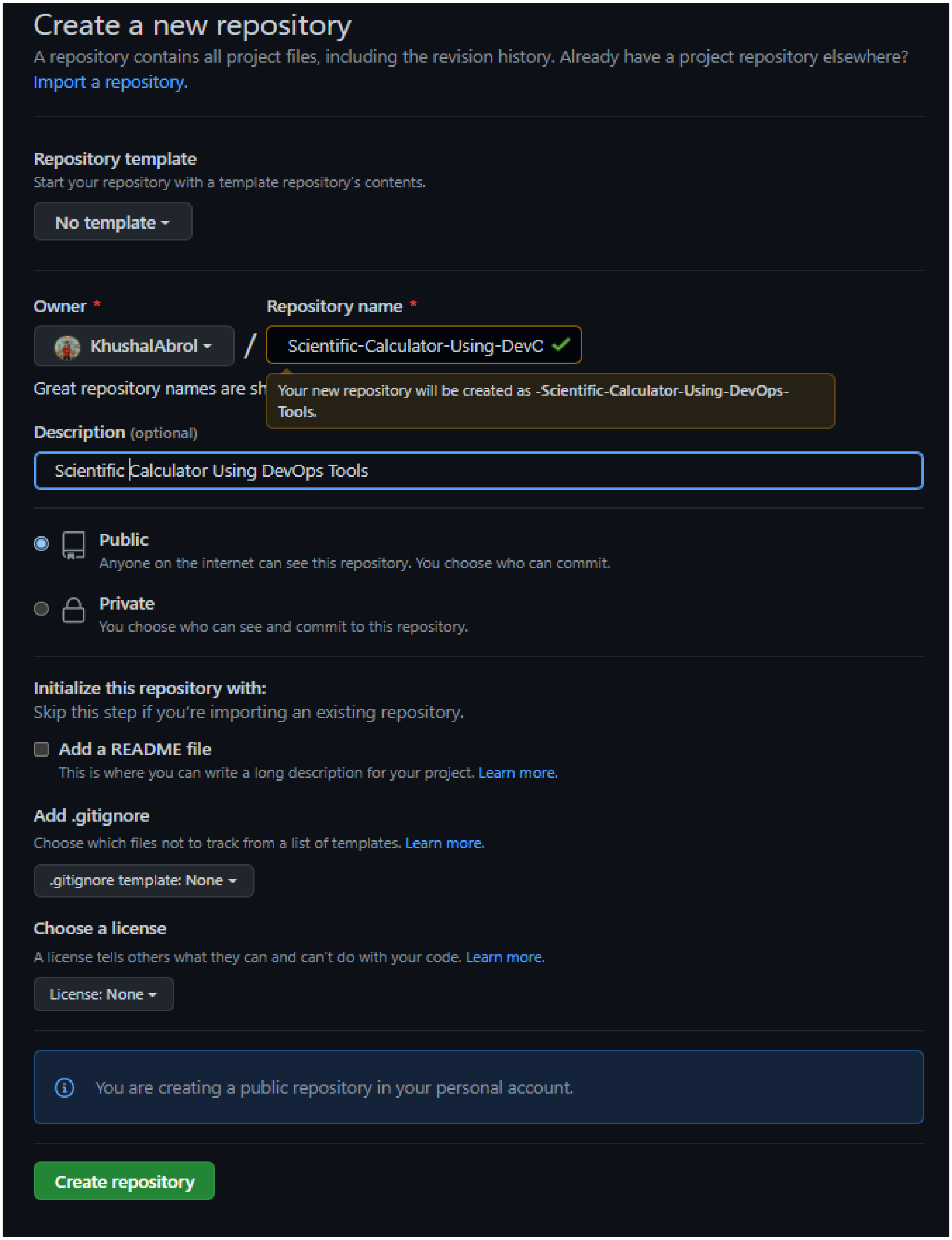
- **Initialize Angular Project**
  ng new scientific-calculator-using-devops-tools



```
Microsoft Windows [Version 10.0.19044.1645]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Khushal Abrol>d:

D:\>mkdir mini-project

D:\>cd mini-project

D:\mini-project>ng new scientific-calculator-using-devops-tools
? Would you like to add Angular routing? No
? Which stylesheet format would you like to use? CSS
CREATE scientific-calculator-using-devops-tools/angular.json (3243 bytes)
CREATE scientific-calculator-using-devops-tools/package.json (1103 bytes)
CREATE scientific-calculator-using-devops-tools/README.md (1082 bytes)
CREATE scientific-calculator-using-devops-tools/tsconfig.json (863 bytes)
CREATE scientific-calculator-using-devops-tools/.editorconfig (274 bytes)
CREATE scientific-calculator-using-devops-tools/.gitignore (548 bytes)
CREATE scientific-calculator-using-devops-tools/.browserslistrc (600 bytes)
CREATE scientific-calculator-using-devops-tools/karma.conf.js (1457 bytes)
CREATE scientific-calculator-using-devops-tools/tsconfig.app.json (287 bytes)
CREATE scientific-calculator-using-devops-tools/tsconfig.spec.json (333 bytes)
CREATE scientific-calculator-using-devops-tools/.vscode/extensions.json (130 bytes)
CREATE scientific-calculator-using-devops-tools/.vscode/launch.json (474 bytes)
CREATE scientific-calculator-using-devops-tools/.vscode/tasks.json (938 bytes)
CREATE scientific-calculator-using-devops-tools/src/favicon.ico (948 bytes)
CREATE scientific-calculator-using-devops-tools/src/index.html (322 bytes)
CREATE scientific-calculator-using-devops-tools/src/main.ts (372 bytes)
CREATE scientific-calculator-using-devops-tools/src/polyfills.ts (2338 bytes)
CREATE scientific-calculator-using-devops-tools/src/styles.css (80 bytes)
CREATE scientific-calculator-using-devops-tools/src/test.ts (745 bytes)
CREATE scientific-calculator-using-devops-tools/src/assets/.gitkeep (0 bytes)
CREATE scientific-calculator-using-devops-tools/src/environments/environment.prod.ts (51 bytes)
CREATE scientific-calculator-using-devops-tools/src/environments/environment.ts (658 bytes)
CREATE scientific-calculator-using-devops-tools/src/app/app.module.ts (314 bytes)
CREATE scientific-calculator-using-devops-tools/src/app/app.component.html (23332 bytes)
CREATE scientific-calculator-using-devops-tools/src/app/app.component.spec.ts (1058 bytes)
CREATE scientific-calculator-using-devops-tools/src/app/app.component.ts (244 bytes)
CREATE scientific-calculator-using-devops-tools/src/app/app.component.css (0 bytes)
/ Installing packages (npm)...
```

Fig: Creating new Angular Project

- **Initialize Git Repository**



Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? Import a repository.

Repository template

Start your repository with a template repository's contents.

Owner * **Repository name** *

KhushalAbrol ▾ / Scientific-Calculator-Using-DevC ✓

Great repository names are sh

Your new repository will be created as -Scientific-Calculator-Using-DevOps-Tools.

Description (optional)

Scientific Calculator Using DevOps Tools

● **Public**
Anyone on the internet can see this repository. You choose who can commit.

○ **Private**
You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.

□ **Add a README file**
This is where you can write a long description for your project. Learn more.

Add .gitignore

Choose which files not to track from a list of templates. Learn more.

.gitignore template: None ▾

Choose a license

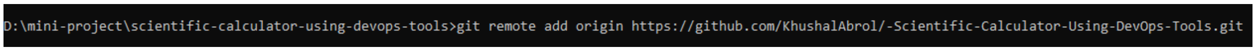
A license tells others what they can and can't do with your code. Learn more.

License: None ▾

ⓘ You are creating a public repository in your personal account.

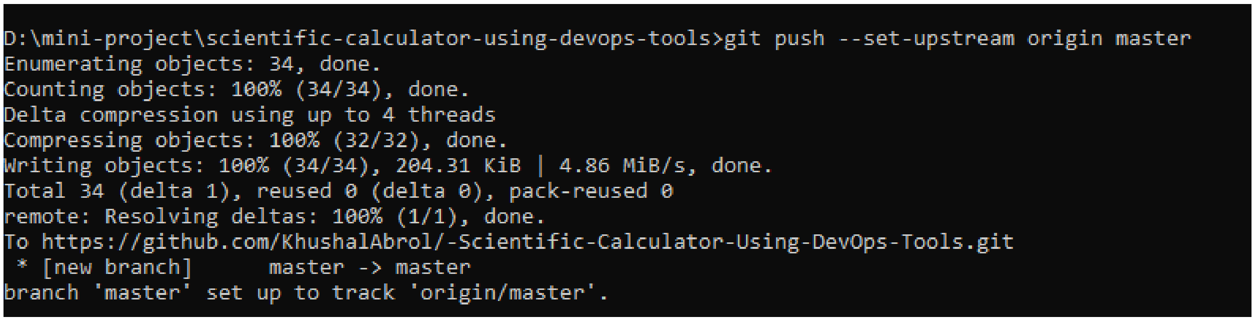
Fig: Creating new Github repository

- **Pushing Code To Github**
  Pushing angular boilerplate code to remote github repository as initial commit.



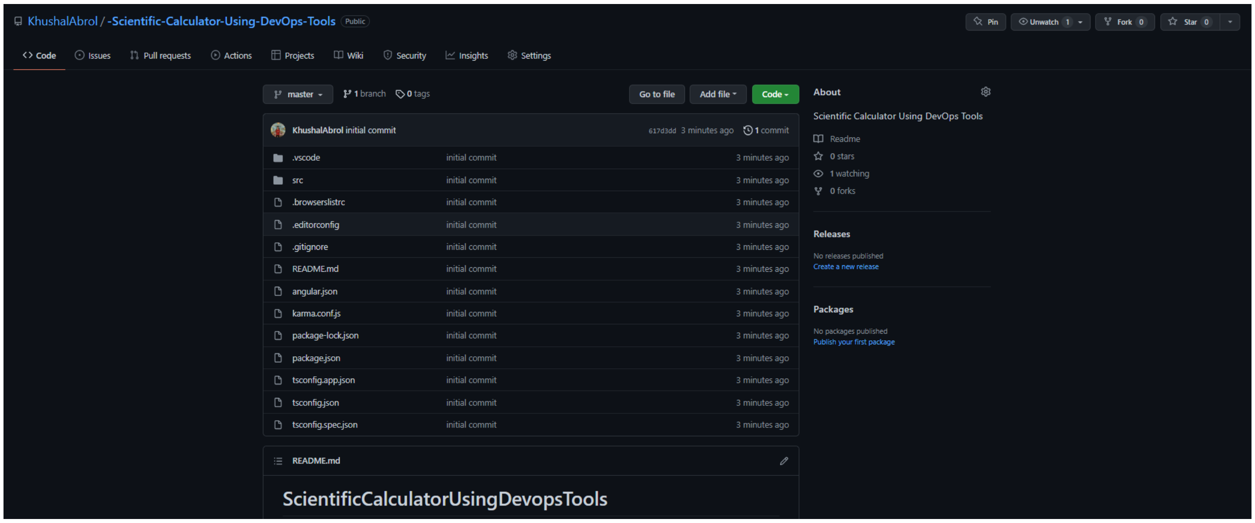
```
D:\mini-project\scientific-calculator-using-devops-tools>git remote add origin https://github.com/KhushalAbrol/-Scientific-Calculator-Using-DevOps-Tools.git
```

Fig: Setting git remote origin



```
D:\mini-project\scientific-calculator-using-devops-tools>git push --set-upstream origin master
Enumerating objects: 34, done.
Counting objects: 100% (34/34), done.
Delta compression using up to 4 threads
Compressing objects: 100% (32/32), done.
Writing objects: 100% (34/34), 204.31 KiB | 4.86 MiB/s, done.
Total 34 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/KhushalAbrol/-Scientific-Calculator-Using-DevOps-Tools.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
```

Fig: Pushing local code to remote repository



KhushalAbrol initial commit 617d3dd 3 minutes ago 1 commit

.vscode	initial commit	3 minutes ago
src	initial commit	3 minutes ago
.browserslistrc	initial commit	3 minutes ago
.editorconfig	initial commit	3 minutes ago
.gitignore	initial commit	3 minutes ago
README.md	initial commit	3 minutes ago
angular.json	initial commit	3 minutes ago
karma.conf.js	initial commit	3 minutes ago
package-lock.json	initial commit	3 minutes ago
package.json	initial commit	3 minutes ago
tsconfig.app.json	initial commit	3 minutes ago
tsconfig.json	initial commit	3 minutes ago
tsconfig.spec.json	initial commit	3 minutes ago

README.md

ScientificCalculatorUsingDevopsTools

About

Scientific Calculator Using DevOps Tools

Readme
0 stars
1 watching
0 forks

Releases

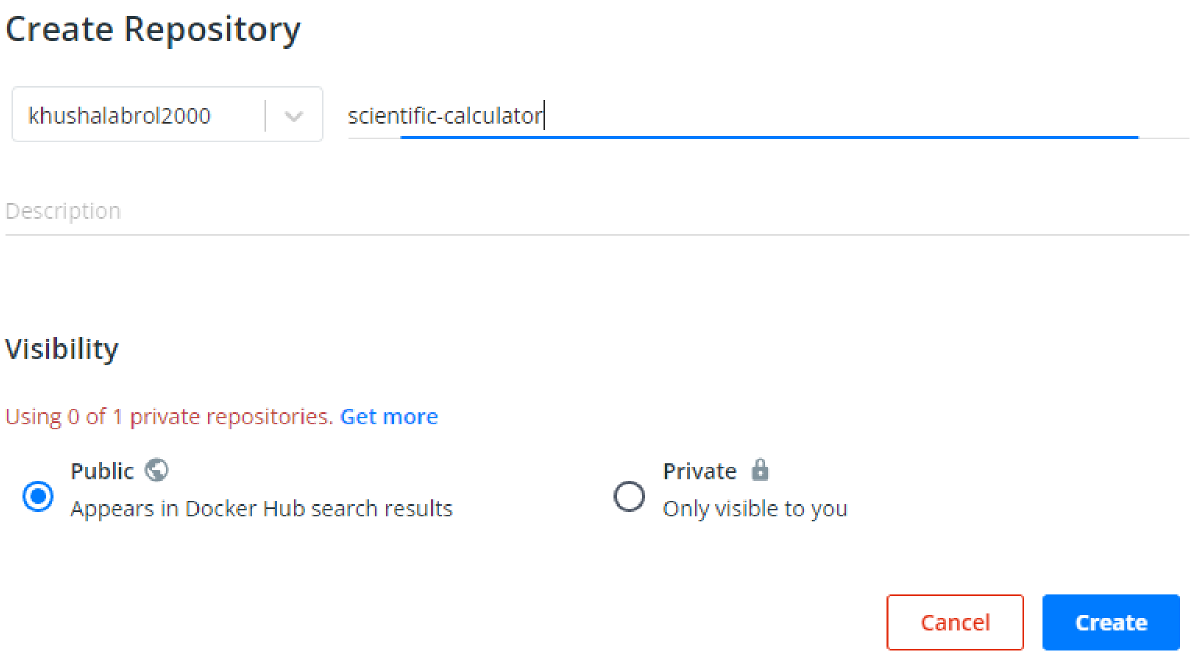
No releases published
Create a new release

Packages

No packages published
Publish your first package

Fig: Initial commit

- **Create Docker Hub Repository**
  Creating docker hub repository https://hub.docker.com/ > Create Repository



Create Repository

khushalabrol2000 / scientific-calculator

Description

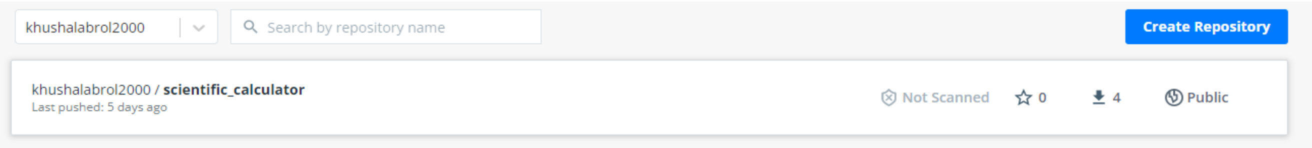
Visibility

Using 0 of 1 private repositories. Get more

● Public — Appears in Docker Hub search results

○ Private — Only visible to you

Fig: Creating Docker Hub Repository



khushalabrol2000

Search by repository name

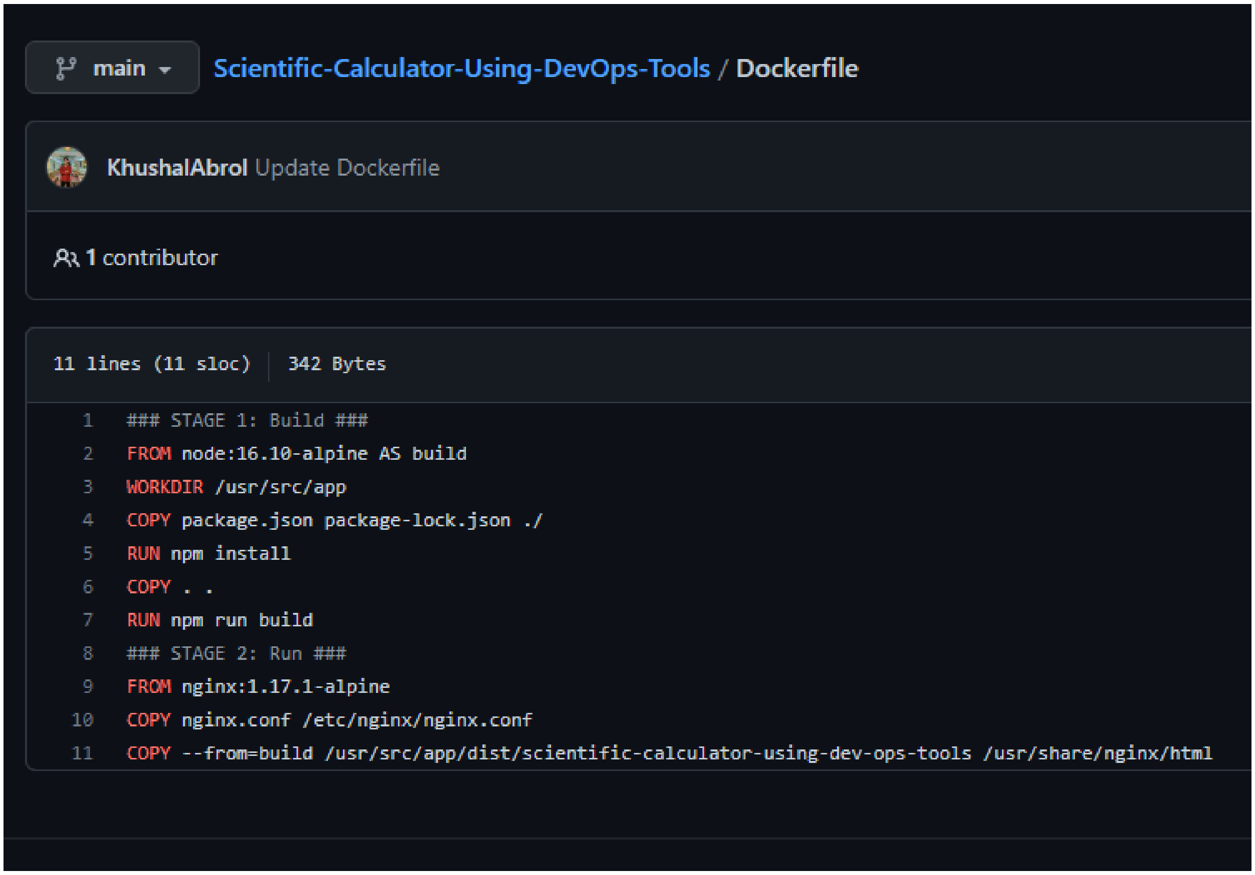
khushalabrol2000 / scientific_calculator
Last pushed: 5 days ago

Not Scanned ☆ 0 ⬇ 4 Public

Fig: Docker Hub Repository is being created

Step 4: Creating Jenkins Pipeline

- **Creating Docker File**
  Docker is an orchestration tool which helps in making an isolated environment for a particular application.
  Docker file contains the instructions about how to make a docker image of the project. The environment (packages/dependencies) needed for the project to work are installed in docker image and their corresponding instructions are present here.



main Scientific-Calculator-Using-DevOps-Tools / Dockerfile

KhushalAbrol Update Dockerfile

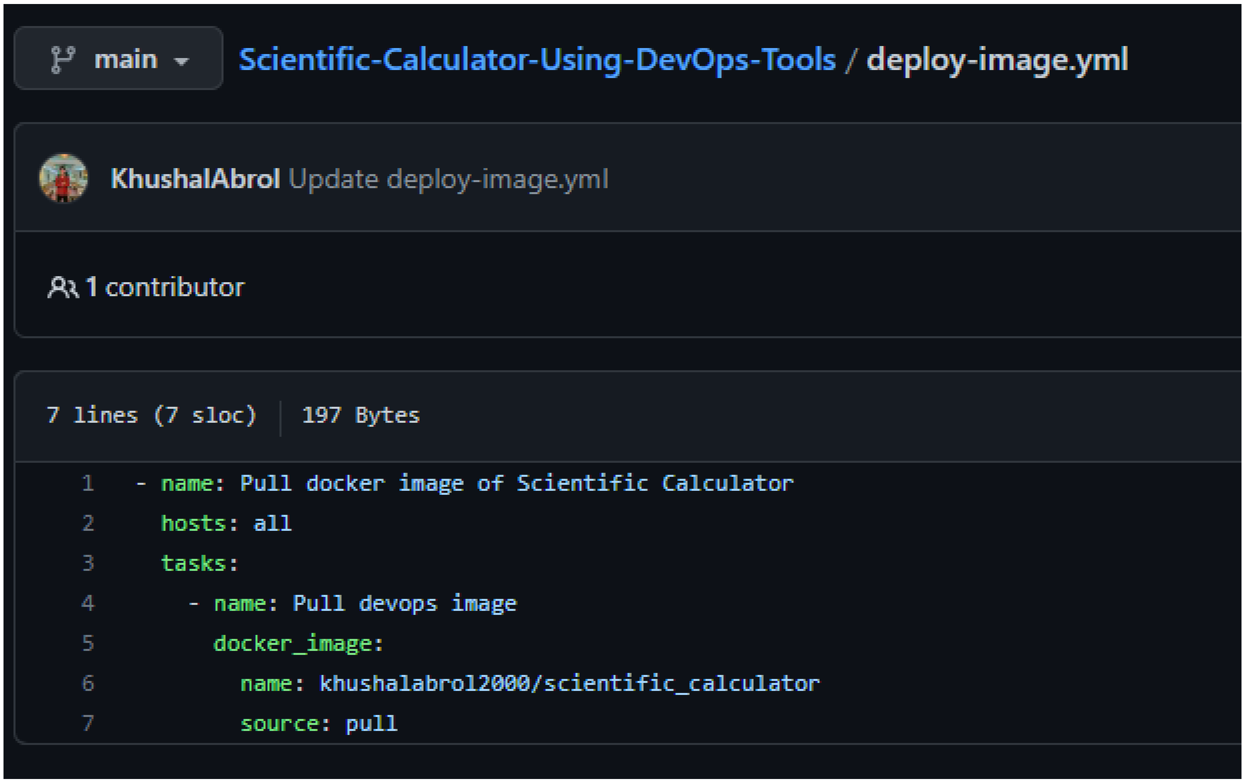
1 contributor

11 lines (11 sloc) | 342 Bytes

```
### STAGE 1: Build ###
FROM node:16.10-alpine AS build
WORKDIR /usr/src/app
COPY package.json package-lock.json ./
RUN npm install
COPY . .
RUN npm run build
### STAGE 2: Run ###
FROM nginx:1.17.1-alpine
COPY nginx.conf /etc/nginx/nginx.conf
COPY --from=build /usr/src/app/dist/scientific-calculator-using-dev-ops-tools /usr/share/nginx/html
```

Fig: Docker File

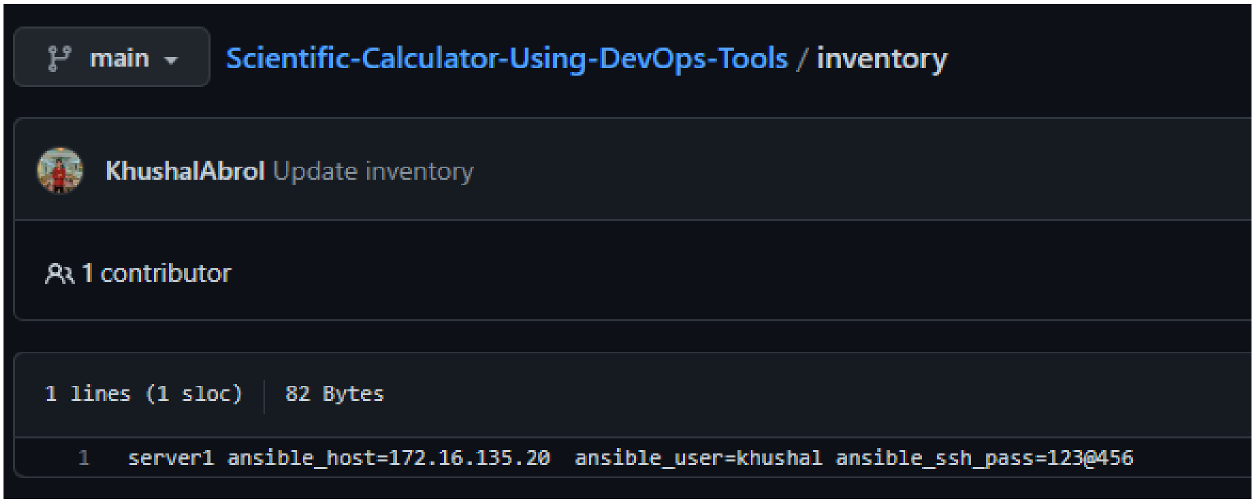
- **Creating Ansible YAML file**
  Ansible is a deployment tool which helps in automating the deployment of the application on a server. For using ansible we don't need to use any corresponding tool on server side that's the main feature ansible has over its competitors.
  Ansible needs a .yml file which contains all the instructions about what app has to be deployed, on which server the app needs to be deployed, and how the app needs to be deployed.
  Here we are instructing ansible to pull image from **khushalabrol2000/scientific_calculator** docker hub repository and deploy it on all hosts.


main
Scientific-Calculator-Using-DevOps-Tools / deploy-image.yml
KhushalAbrol Update deploy-image.yml
1 contributor
7 lines (7 sloc) | 197 Bytes

```yaml
- name: Pull docker image of Scientific Calculator
  hosts: all
  tasks:
    - name: Pull devops image
      docker_image:
        name: khushalabrol2000/scientific_calculator
        source: pull
```

Fig: Ansible deployment file

- **Creating inventory file**
  Ansible also needs an inventory file which contains information about all the nodes/hosts on which we need to deploy our app.
  Here I only have one server so I gave the information of that server for ansible to connect to and deploy the app.

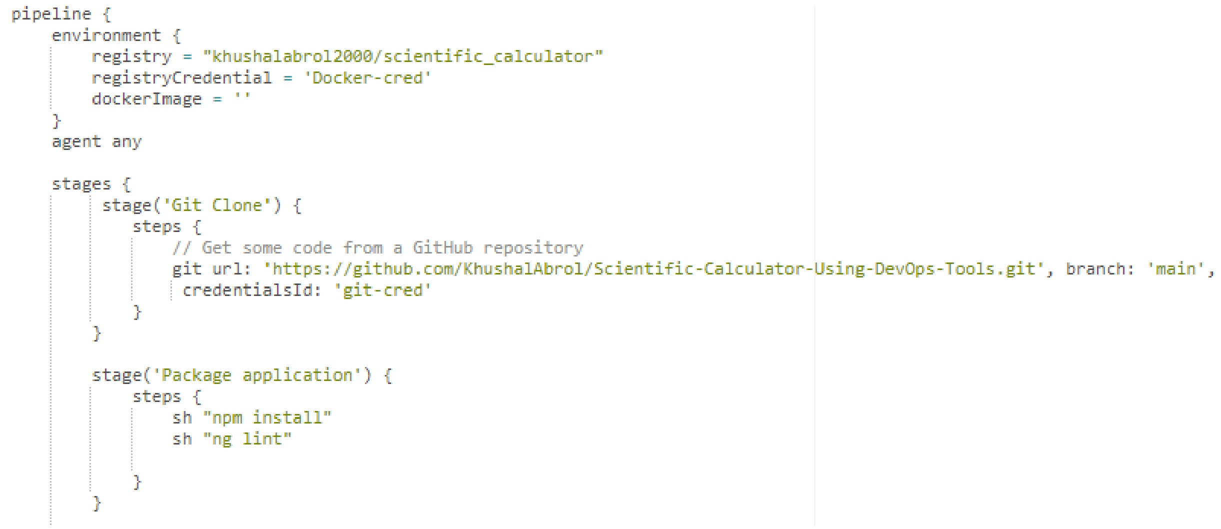

main
Scientific-Calculator-Using-DevOps-Tools / inventory
KhushalAbrol Update inventory
1 contributor
1 lines (1 sloc)
82 Bytes

```
server1 ansible_host=172.16.135.20  ansible_user=khushal ansible_ssh_pass=123@456
```

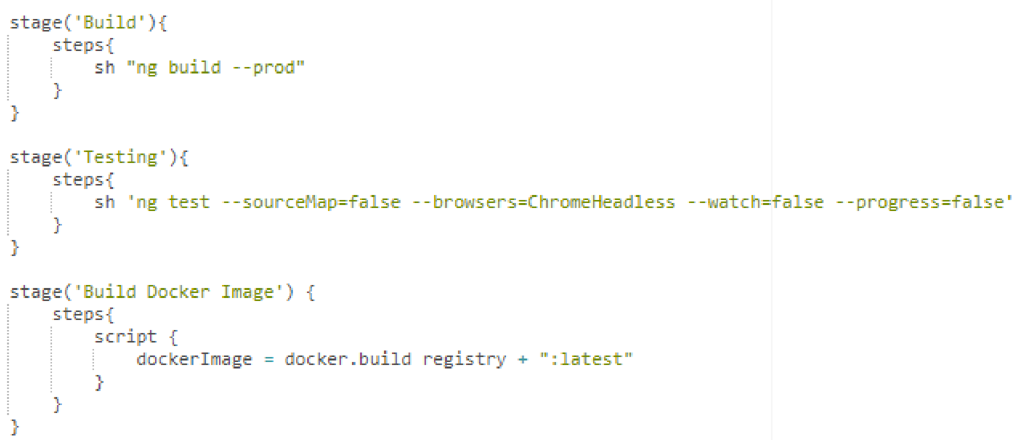
Fig: Ansible inventory file

- **Jenkins Pipeline Script**
  Jenkins pipeline script is written in groovy language. It contains information about all the stages which we need in the app from cloning from git to deploying it to the server.

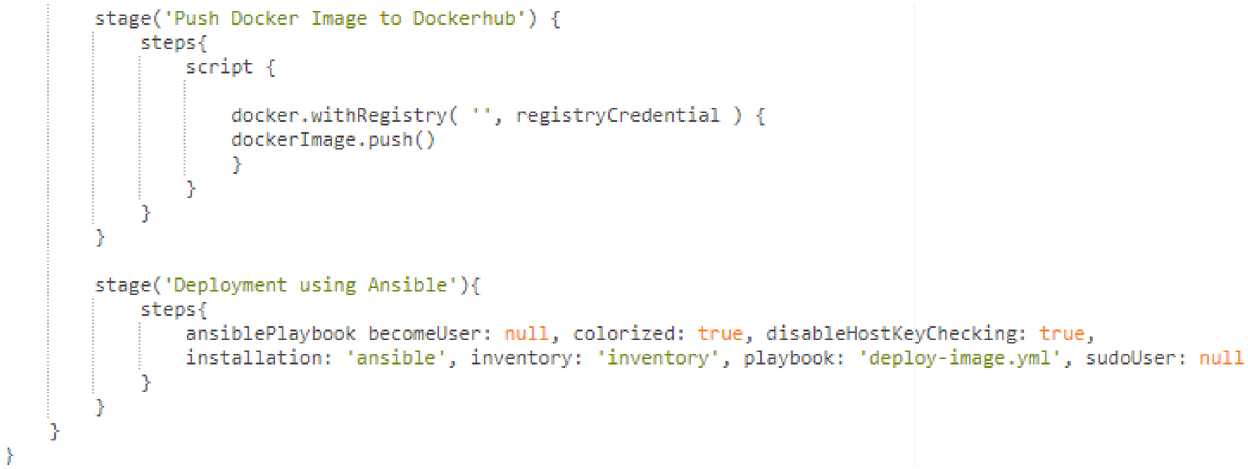
  Here we have 7 Steps:
  - ➢ Cloning the repository from github
  - ➢ Installing relevant packages
  - ➢ Building the app
  - ➢ Testing the app
  - ➢ Building docker image
  - ➢ Pushing docker image to dockerhub
  - ➢ Deploying the app to the server



```
pipeline {
    environment {
        registry = "khushalabrol2000/scientific_calculator"
        registryCredential = 'Docker-cred'
        dockerImage = ''
    }
    agent any

    stages {
         stage('Git Clone') {
            steps {
                // Get some code from a GitHub repository
                git url: 'https://github.com/KhushalAbrol/Scientific-Calculator-Using-DevOps-Tools.git', branch: 'main',
                 credentialsId: 'git-cred'
            }
        }

        stage('Package application') {
            steps {
                sh "npm install"
                sh "ng lint"

            }
        }
```



```
  stage('Build'){
      steps{
          sh "ng build --prod"
      }
  }

  stage('Testing'){
      steps{
          sh 'ng test --sourceMap=false --browsers=ChromeHeadless --watch=false --progress=false'
      }
  }

  stage('Build Docker Image') {
      steps{
          script {
              dockerImage = docker.build registry + ":latest"
          }
      }
  }
```



```
        stage('Push Docker Image to Dockerhub') {
            steps{
                script {

                    docker.withRegistry( '', registryCredential ) {
                    dockerImage.push()
                    }
                }
            }
        }

        stage('Deployment using Ansible'){
            steps{
                ansiblePlaybook becomeUser: null, colorized: true, disableHostKeyChecking: true,
                installation: 'ansible', inventory: 'inventory', playbook: 'deploy-image.yml', sudoUser: null
            }
        }
    }
}
```

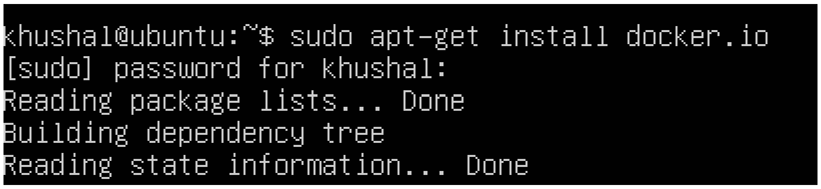
Fig: Jenkins Pipeline Script

Step 5: Setting Up Server

- **Install Docker**

  We need to install docker on the server so that the server can understand the docker image and run it.

  - npm install docker.io



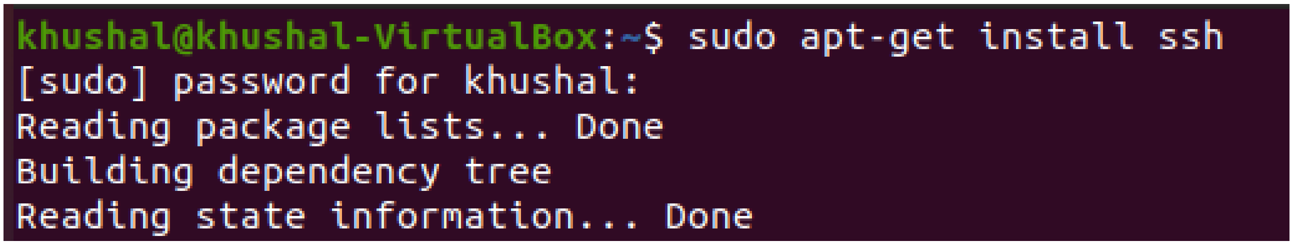
```
khushal@ubuntu:~$ sudo apt-get install docker.io
[sudo] password for khushal:
Reading package lists... Done
Building dependency tree
Reading state information... Done
```

Fig: Installing docker on server

- **Connecting Jenkins Master and Node Server**

  We are using ssh to connect Jenkins Master and server node. First we need to install ssh,

  sudo get-apt install ssh



```
khushal@khushal-VirtualBox:~$ sudo apt-get install ssh
[sudo] password for khushal:
Reading package lists... Done
Building dependency tree
Reading state information... Done
```

Fig: Installing ssh

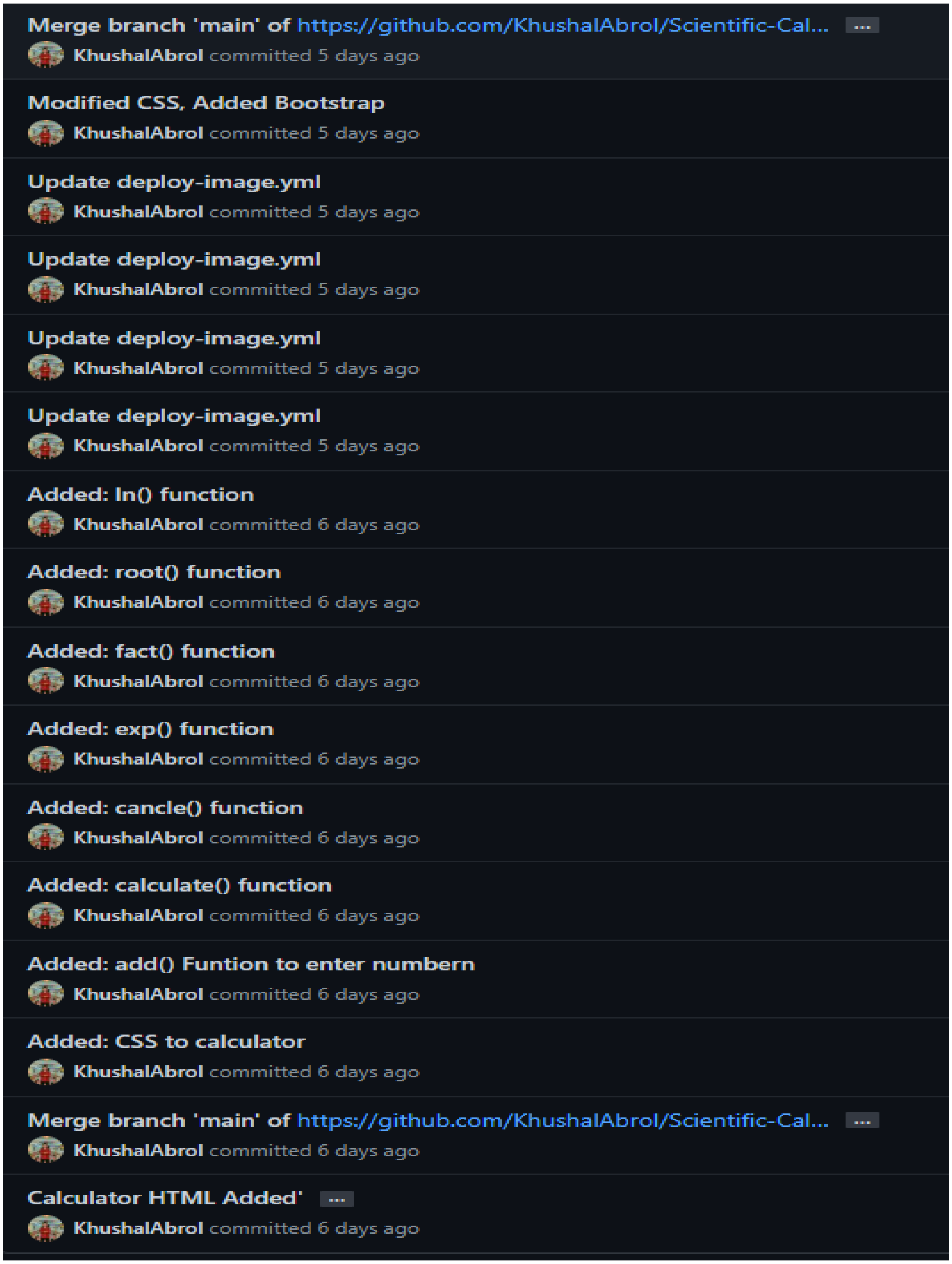
ssh will connect to the server using the inventory file which we have created earlier.

Step 6: Development

Now we come to the development part of the project as everything needed is set up. We develop the code and write tests and jenkins will automatically build, test and deploy the app on the server.

- **Development & Testing**

  Developed the code and wrote tests bit by bit, committed and pushed to github.



Merge branch 'main' of https://github.com/KhushalAbrol/Scientific-Cal... ...
KhushalAbrol committed 5 days ago

Modified CSS, Added Bootstrap
KhushalAbrol committed 5 days ago

Update deploy-image.yml
KhushalAbrol committed 5 days ago

Update deploy-image.yml
KhushalAbrol committed 5 days ago

Update deploy-image.yml
KhushalAbrol committed 5 days ago

Update deploy-image.yml
KhushalAbrol committed 5 days ago

Added: ln() function
KhushalAbrol committed 6 days ago

Added: root() function
KhushalAbrol committed 6 days ago

Added: fact() function
KhushalAbrol committed 6 days ago

Added: exp() function
KhushalAbrol committed 6 days ago

Added: cancle() function
KhushalAbrol committed 6 days ago

Added: calculate() function
KhushalAbrol committed 6 days ago

Added: add() Funtion to enter numbern
KhushalAbrol committed 6 days ago

Added: CSS to calculator
KhushalAbrol committed 6 days ago

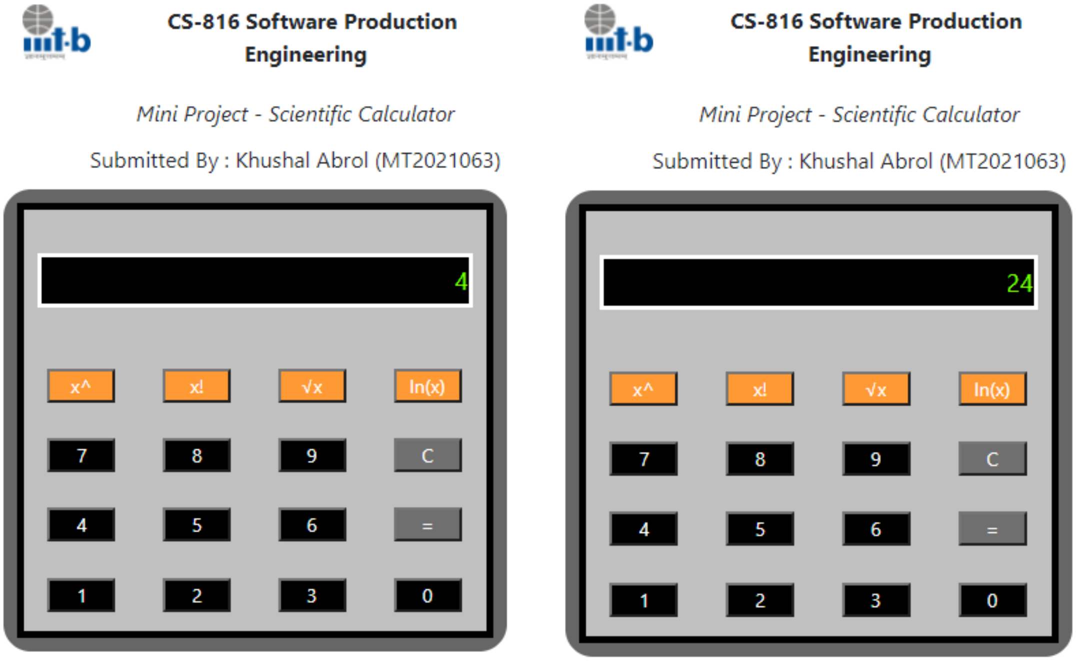
Merge branch 'main' of https://github.com/KhushalAbrol/Scientific-Cal... ...
KhushalAbrol committed 6 days ago

Calculator HTML Added' ...
KhushalAbrol committed 6 days ago

Fig: Git Commits

- **Logic and User Interface**
  Since it is a angular base app, it has a user interface:

  Let’s see how it calculates: 4!



CS-816 Software Production Engineering

Mini Project - Scientific Calculator

Submitted By : Khushal Abrol (MT2021063)

CS-816 Software Production Engineering

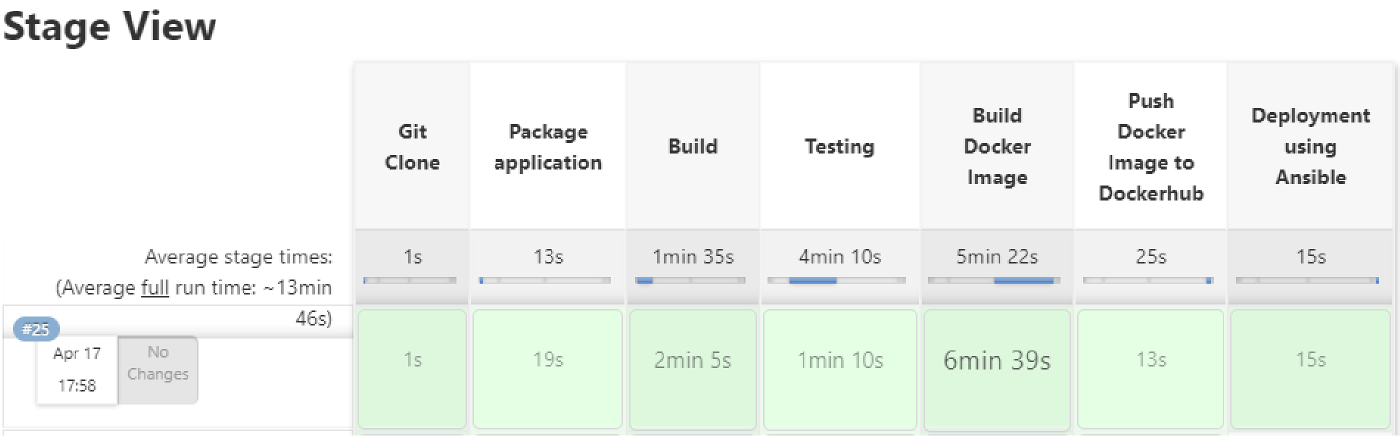
Mini Project - Scientific Calculator

Submitted By : Khushal Abrol (MT2021063)

FIg: Pressed Key 4 > Pressed Factorial Key

  We got the output as 24 when we pressed the factorial key.

- **Running Jenkins Pipeline**
  As we are using SCM polling Jenkins will check for updates in the git repository and when a commit is detected it will start to run the pipeline.
  Here in this version of the app every step ran successfully and our app is deployed on the server.

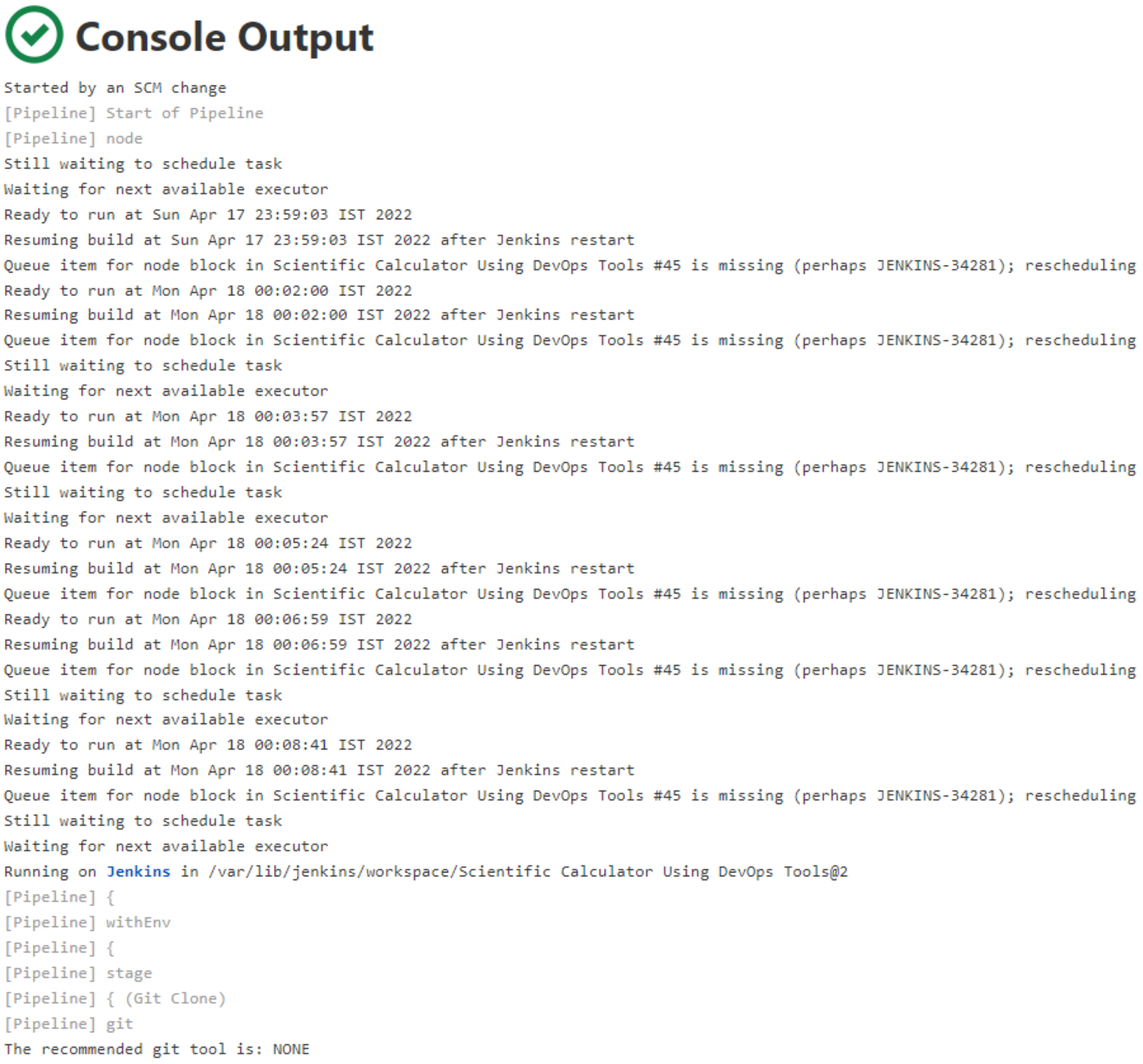


Stage View

	Git Clone	Package application	Build	Testing	Build Docker Image	Push Docker Image to Dockerhub	Deployment using Ansible
Average stage times: (Average full run time: ~13min 46s)	1s	13s	1min 35s	4min 10s	5min 22s	25s	15s
#25 Apr 17 17:58 No Changes	1s	19s	2min 5s	1min 10s	6min 39s	13s	15s

Fig: Jenkins Pipeline

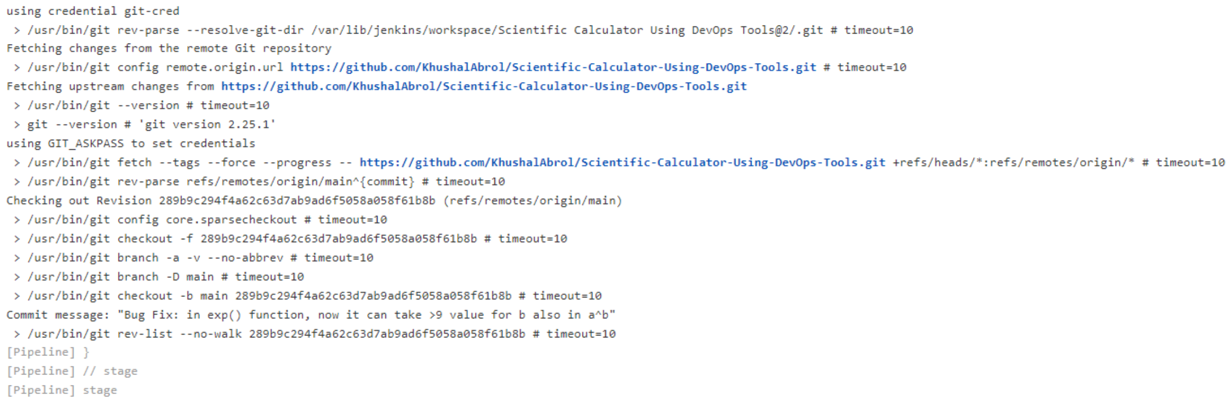
Following are the screenshot of console output of jenkins pipeline.



Console Output

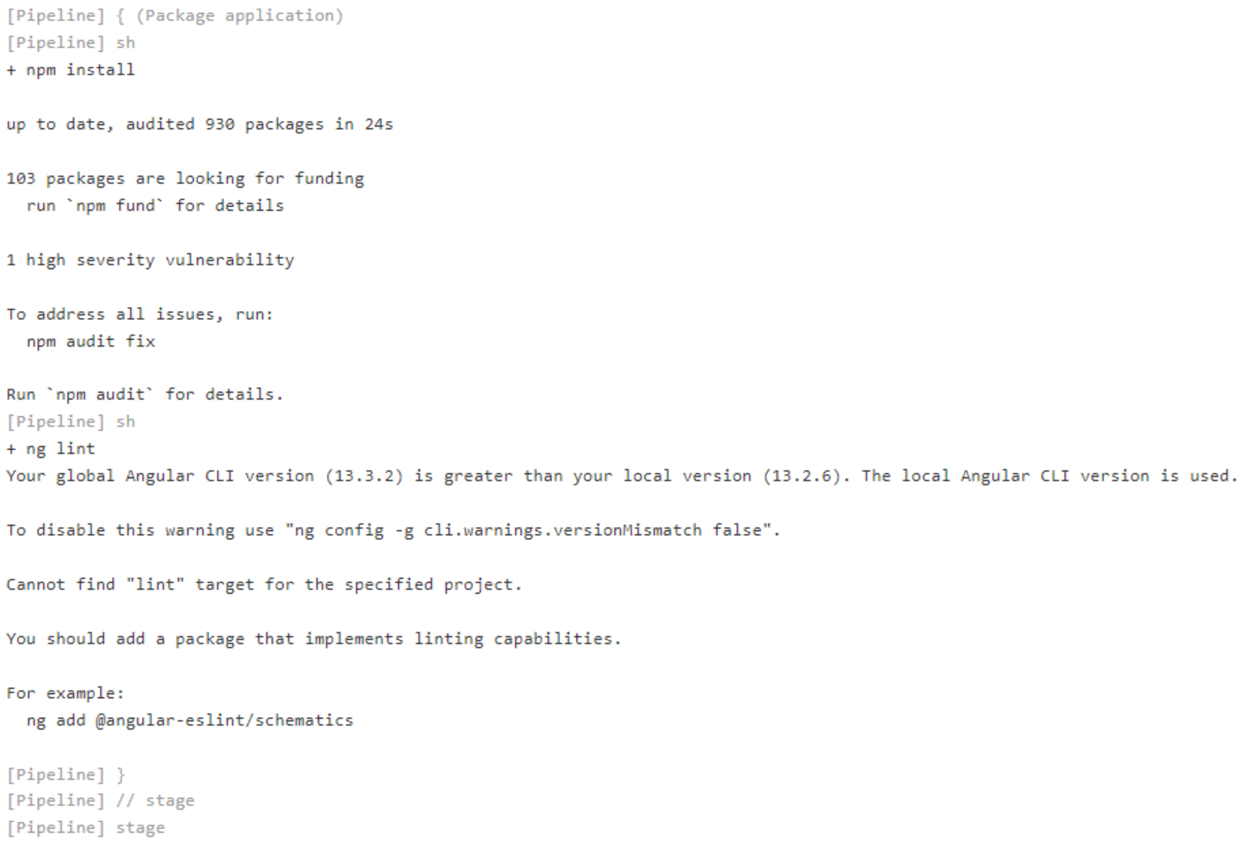
```
Started by an SCM change
[Pipeline] Start of Pipeline
[Pipeline] node
Still waiting to schedule task
Waiting for next available executor
Ready to run at Sun Apr 17 23:59:03 IST 2022
Resuming build at Sun Apr 17 23:59:03 IST 2022 after Jenkins restart
Queue item for node block in Scientific Calculator Using DevOps Tools #45 is missing (perhaps JENKINS-34281); rescheduling
Ready to run at Mon Apr 18 00:02:00 IST 2022
Resuming build at Mon Apr 18 00:02:00 IST 2022 after Jenkins restart
Queue item for node block in Scientific Calculator Using DevOps Tools #45 is missing (perhaps JENKINS-34281); rescheduling
Still waiting to schedule task
Waiting for next available executor
Ready to run at Mon Apr 18 00:03:57 IST 2022
Resuming build at Mon Apr 18 00:03:57 IST 2022 after Jenkins restart
Queue item for node block in Scientific Calculator Using DevOps Tools #45 is missing (perhaps JENKINS-34281); rescheduling
Still waiting to schedule task
Waiting for next available executor
Ready to run at Mon Apr 18 00:05:24 IST 2022
Resuming build at Mon Apr 18 00:05:24 IST 2022 after Jenkins restart
Queue item for node block in Scientific Calculator Using DevOps Tools #45 is missing (perhaps JENKINS-34281); rescheduling
Ready to run at Mon Apr 18 00:06:59 IST 2022
Resuming build at Mon Apr 18 00:06:59 IST 2022 after Jenkins restart
Queue item for node block in Scientific Calculator Using DevOps Tools #45 is missing (perhaps JENKINS-34281); rescheduling
Still waiting to schedule task
Waiting for next available executor
Ready to run at Mon Apr 18 00:08:41 IST 2022
Resuming build at Mon Apr 18 00:08:41 IST 2022 after Jenkins restart
Queue item for node block in Scientific Calculator Using DevOps Tools #45 is missing (perhaps JENKINS-34281); rescheduling
Still waiting to schedule task
Waiting for next available executor
Running on Jenkins in /var/lib/jenkins/workspace/Scientific Calculator Using DevOps Tools@2
[Pipeline] {
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Git Clone)
[Pipeline] git
The recommended git tool is: NONE
```

Fig: Starting Jenkins pipeline



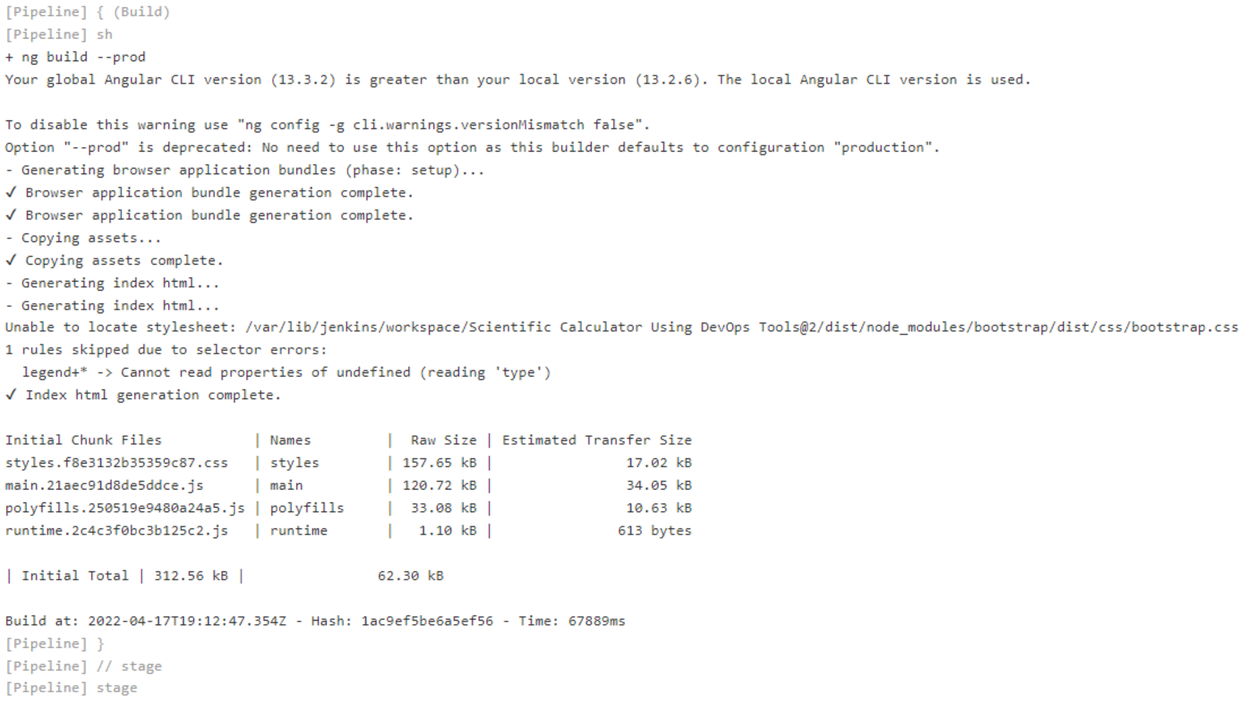
```
using credential git-cred
 > /usr/bin/git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/Scientific Calculator Using DevOps Tools@2/.git # timeout=10
Fetching changes from the remote Git repository
 > /usr/bin/git config remote.origin.url https://github.com/KhushalAbrol/Scientific-Calculator-Using-DevOps-Tools.git # timeout=10
Fetching upstream changes from https://github.com/KhushalAbrol/Scientific-Calculator-Using-DevOps-Tools.git
 > /usr/bin/git --version # timeout=10
 > git --version # 'git version 2.25.1'
using GIT_ASKPASS to set credentials
 > /usr/bin/git fetch --tags --force --progress -- https://github.com/KhushalAbrol/Scientific-Calculator-Using-DevOps-Tools.git +refs/heads/*:refs/remotes/origin/* # timeout=10
 > /usr/bin/git rev-parse refs/remotes/origin/main^{commit} # timeout=10
Checking out Revision 289b9c294f4a62c63d7ab9ad6f5058a058f61b8b (refs/remotes/origin/main)
 > /usr/bin/git config core.sparsecheckout # timeout=10
 > /usr/bin/git checkout -f 289b9c294f4a62c63d7ab9ad6f5058a058f61b8b # timeout=10
 > /usr/bin/git branch -a -v --no-abbrev # timeout=10
 > /usr/bin/git branch -D main # timeout=10
 > /usr/bin/git checkout -b main 289b9c294f4a62c63d7ab9ad6f5058a058f61b8b # timeout=10
Commit message: "Bug Fix: in exp() function, now it can take >9 value for b also in a^b"
 > /usr/bin/git rev-list --no-walk 289b9c294f4a62c63d7ab9ad6f5058a058f61b8b # timeout=10
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
```

Fig: Git Clone Stage



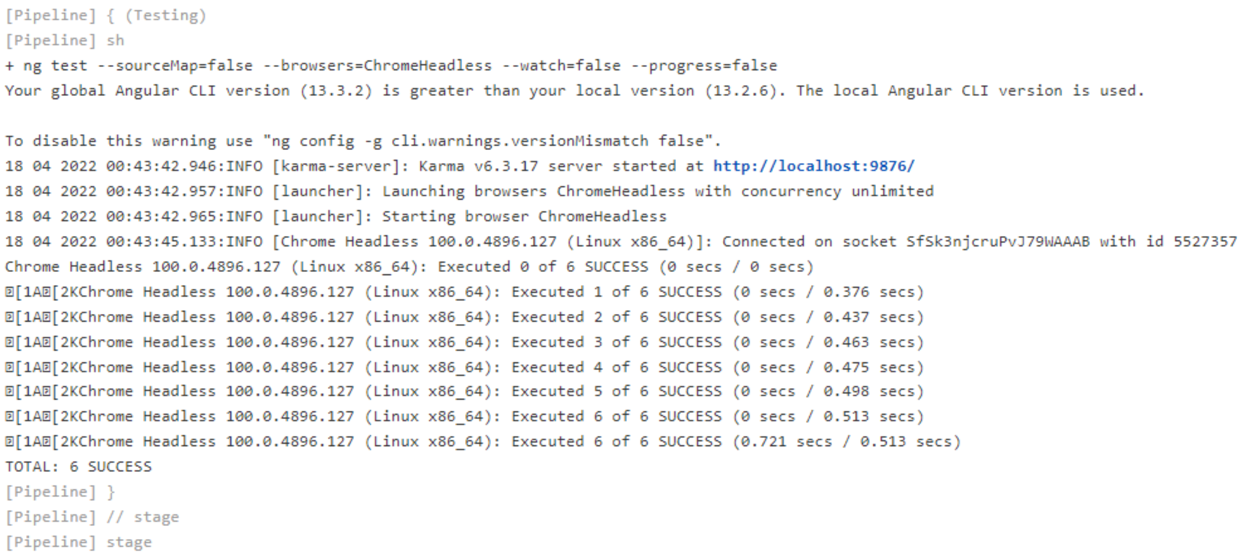
```
[Pipeline] { (Package application)
[Pipeline] sh
+ npm install

up to date, audited 930 packages in 24s

103 packages are looking for funding
  run `npm fund` for details

1 high severity vulnerability

To address all issues, run:
  npm audit fix

Run `npm audit` for details.
[Pipeline] sh
+ ng lint
Your global Angular CLI version (13.3.2) is greater than your local version (13.2.6). The local Angular CLI version is used.

To disable this warning use "ng config -g cli.warnings.versionMismatch false".

Cannot find "lint" target for the specified project.

You should add a package that implements linting capabilities.

For example:
  ng add @angular-eslint/schematics

[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
```

Fig: Package Install Stage



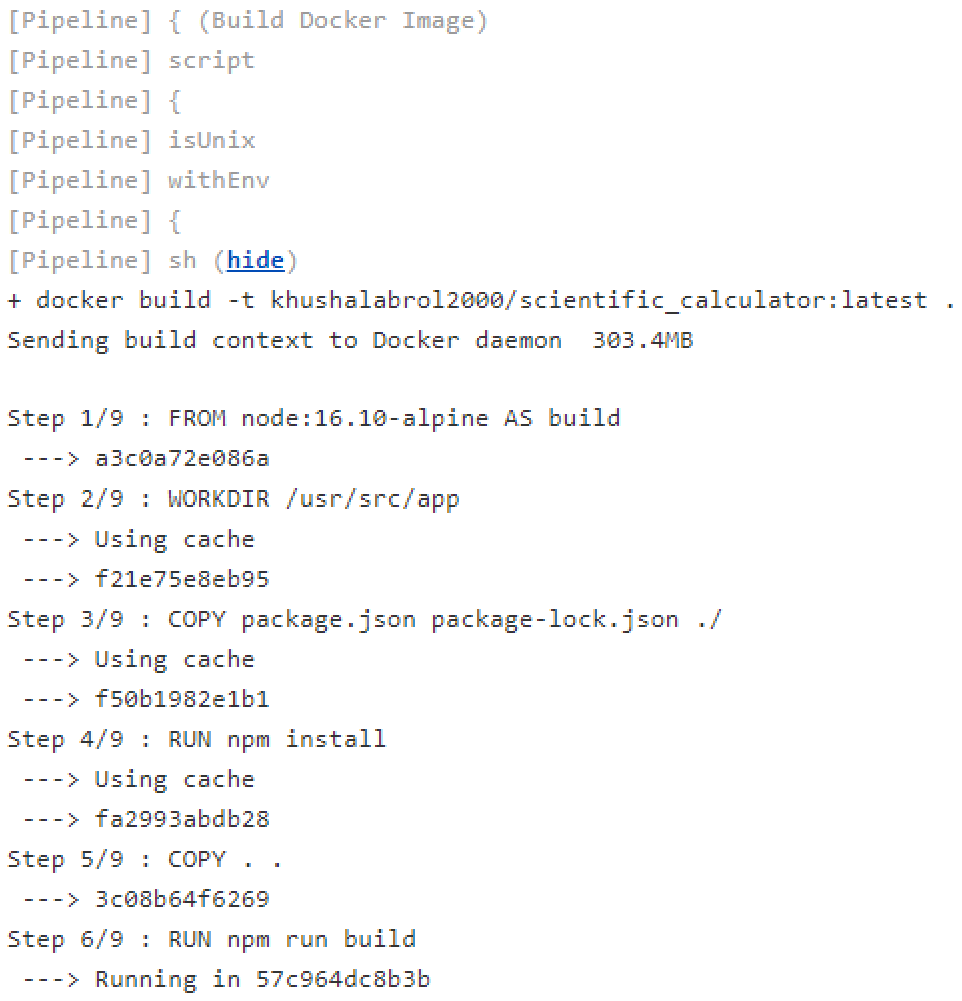
```
[Pipeline] { (Build)
[Pipeline] sh
+ ng build --prod
Your global Angular CLI version (13.3.2) is greater than your local version (13.2.6). The local Angular CLI version is used.

To disable this warning use "ng config -g cli.warnings.versionMismatch false".
Option "--prod" is deprecated: No need to use this option as this builder defaults to configuration "production".
- Generating browser application bundles (phase: setup)...
✓ Browser application bundle generation complete.
✓ Browser application bundle generation complete.
- Copying assets...
✓ Copying assets complete.
- Generating index html...
- Generating index html...
Unable to locate stylesheet: /var/lib/jenkins/workspace/Scientific Calculator Using DevOps Tools@2/dist/node_modules/bootstrap/dist/css/bootstrap.css
1 rules skipped due to selector errors:
  legend+* -> Cannot read properties of undefined (reading 'type')
✓ Index html generation complete.

Initial Chunk Files           | Names         |  Raw Size | Estimated Transfer Size
styles.f8e3132b35359c87.css   | styles        | 157.65 kB |                17.02 kB
main.21aec91d8de5ddce.js      | main          | 120.72 kB |                34.05 kB
polyfills.250519e9480a24a5.js | polyfills     |  33.08 kB |                10.63 kB
runtime.2c4c3f0bc3b125c2.js   | runtime       |   1.10 kB |               613 bytes

| Initial Total | 312.56 kB |                62.30 kB

Build at: 2022-04-17T19:12:47.354Z - Hash: 1ac9ef5be6a5ef56 - Time: 67889ms
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
```

Fig: Build Stage

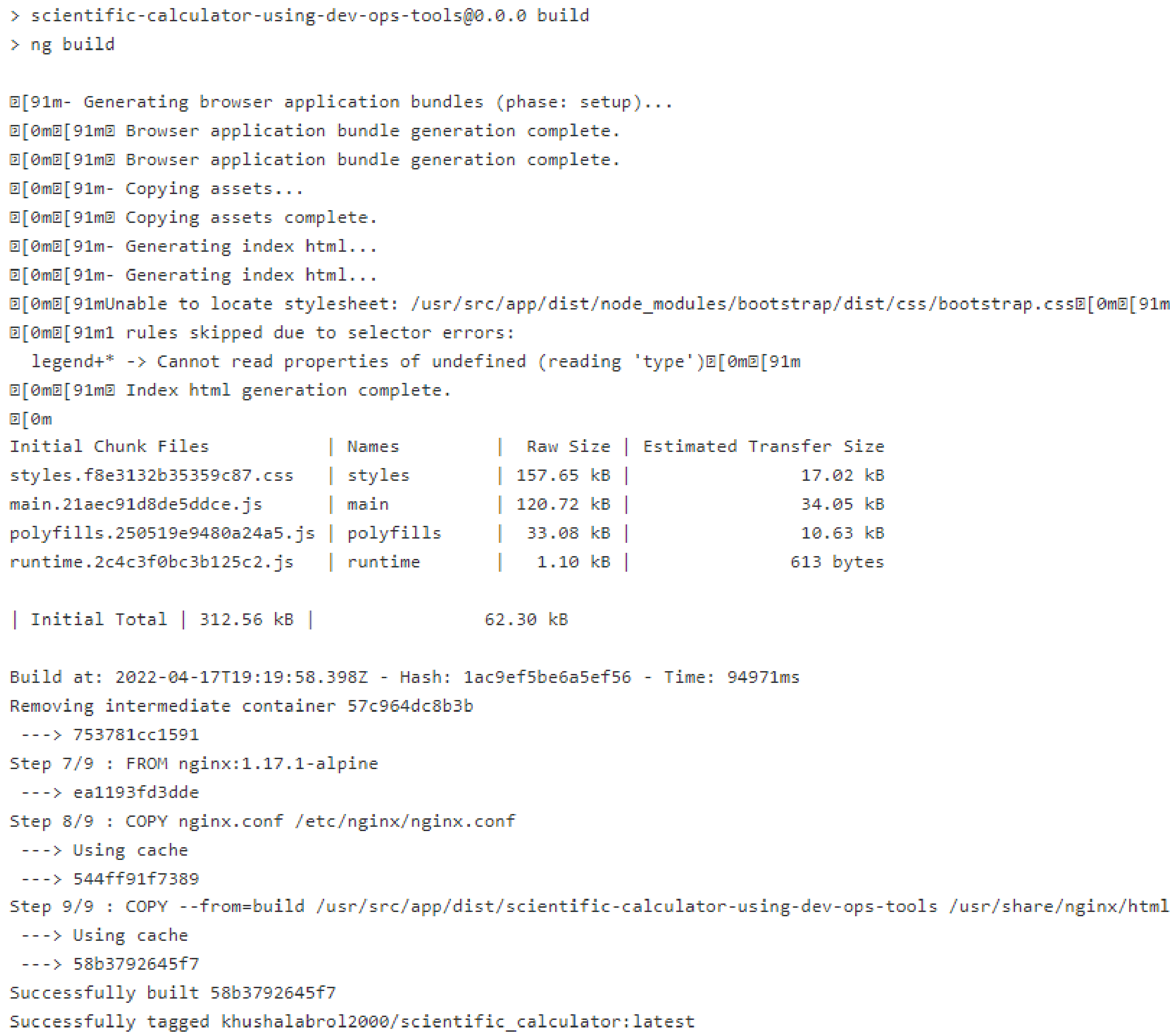


```
[Pipeline] { (Testing)
[Pipeline] sh
+ ng test --sourceMap=false --browsers=ChromeHeadless --watch=false --progress=false
Your global Angular CLI version (13.3.2) is greater than your local version (13.2.6). The local Angular CLI version is used.

To disable this warning use "ng config -g cli.warnings.versionMismatch false".
18 04 2022 00:43:42.946:INFO [karma-server]: Karma v6.3.17 server started at http://localhost:9876/
18 04 2022 00:43:42.957:INFO [launcher]: Launching browsers ChromeHeadless with concurrency unlimited
18 04 2022 00:43:42.965:INFO [launcher]: Starting browser ChromeHeadless
18 04 2022 00:43:45.133:INFO [Chrome Headless 100.0.4896.127 (Linux x86_64)]: Connected on socket SfSk3njcruPvJ79WAAAB with id 5527357
Chrome Headless 100.0.4896.127 (Linux x86_64): Executed 0 of 6 SUCCESS (0 secs / 0 secs)
□[1A□[2KChrome Headless 100.0.4896.127 (Linux x86_64): Executed 1 of 6 SUCCESS (0 secs / 0.376 secs)
□[1A□[2KChrome Headless 100.0.4896.127 (Linux x86_64): Executed 2 of 6 SUCCESS (0 secs / 0.437 secs)
□[1A□[2KChrome Headless 100.0.4896.127 (Linux x86_64): Executed 3 of 6 SUCCESS (0 secs / 0.463 secs)
□[1A□[2KChrome Headless 100.0.4896.127 (Linux x86_64): Executed 4 of 6 SUCCESS (0 secs / 0.475 secs)
□[1A□[2KChrome Headless 100.0.4896.127 (Linux x86_64): Executed 5 of 6 SUCCESS (0 secs / 0.498 secs)
□[1A□[2KChrome Headless 100.0.4896.127 (Linux x86_64): Executed 6 of 6 SUCCESS (0 secs / 0.513 secs)
□[1A□[2KChrome Headless 100.0.4896.127 (Linux x86_64): Executed 6 of 6 SUCCESS (0.721 secs / 0.513 secs)
TOTAL: 6 SUCCESS
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
```

Fig: Testing Stage

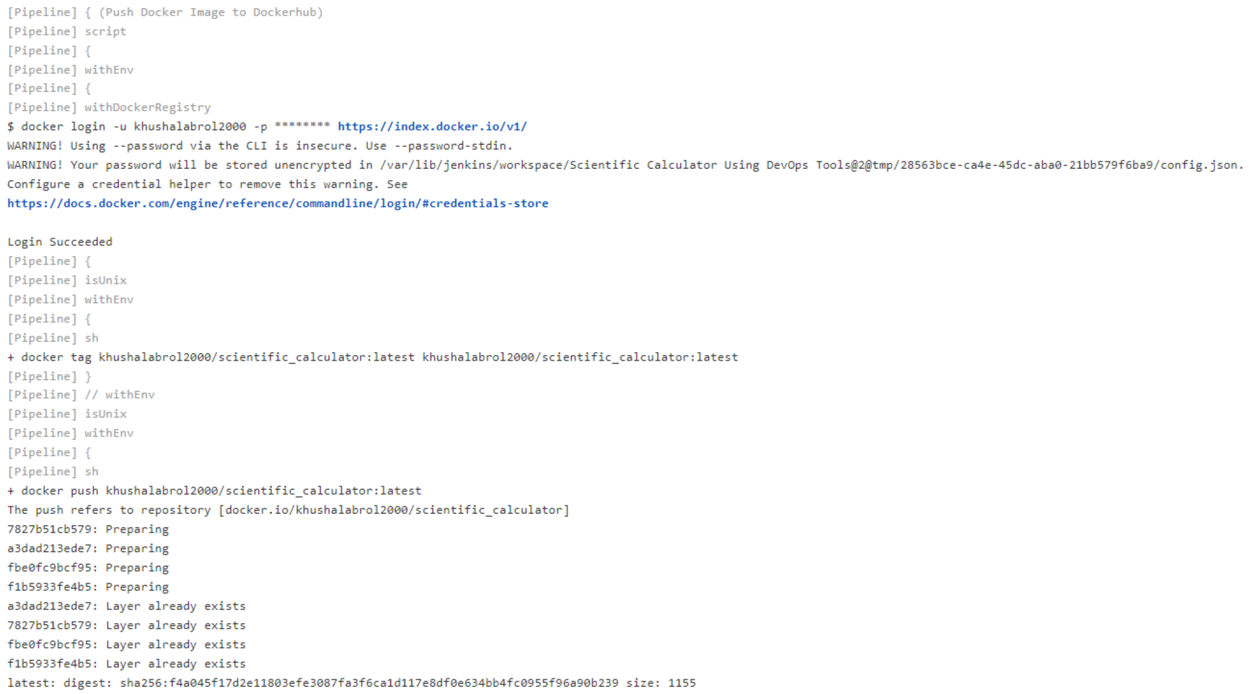


```
[Pipeline] { (Build Docker Image)
[Pipeline] script
[Pipeline] {
[Pipeline] isUnix
[Pipeline] withEnv
[Pipeline] {
[Pipeline] sh (hide)
+ docker build -t khushalabrol2000/scientific_calculator:latest .
Sending build context to Docker daemon  303.4MB

Step 1/9 : FROM node:16.10-alpine AS build
 ---> a3c0a72e086a
Step 2/9 : WORKDIR /usr/src/app
 ---> Using cache
 ---> f21e75e8eb95
Step 3/9 : COPY package.json package-lock.json ./
 ---> Using cache
 ---> f50b1982e1b1
Step 4/9 : RUN npm install
 ---> Using cache
 ---> fa2993abdb28
Step 5/9 : COPY . .
 ---> 3c08b64f6269
Step 6/9 : RUN npm run build
 ---> Running in 57c964dc8b3b
```



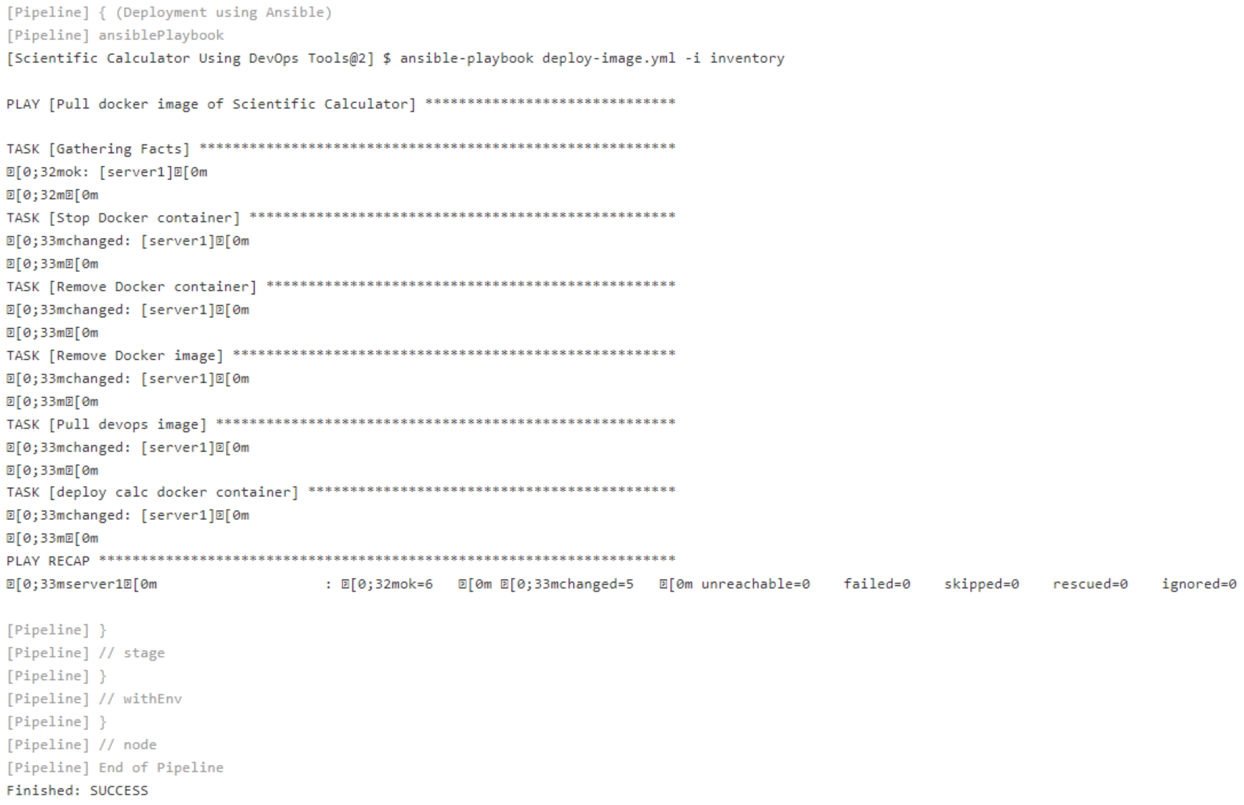
```
> scientific-calculator-using-dev-ops-tools@0.0.0 build
> ng build

⍰[91m- Generating browser application bundles (phase: setup)...
⍰[0m⍰[91m⍰ Browser application bundle generation complete.
⍰[0m⍰[91m⍰ Browser application bundle generation complete.
⍰[0m⍰[91m- Copying assets...
⍰[0m⍰[91m⍰ Copying assets complete.
⍰[0m⍰[91m- Generating index html...
⍰[0m⍰[91m- Generating index html...
⍰[0m⍰[91mUnable to locate stylesheet: /usr/src/app/dist/node_modules/bootstrap/dist/css/bootstrap.css⍰[0m⍰[91m
⍰[0m⍰[91m1 rules skipped due to selector errors:
  legend+* -> Cannot read properties of undefined (reading 'type')⍰[0m⍰[91m
⍰[0m⍰[91m⍰ Index html generation complete.
⍰[0m
Initial Chunk Files            | Names         |  Raw Size | Estimated Transfer Size
styles.f8e3132b35359c87.css    | styles        | 157.65 kB |                17.02 kB
main.21aec91d8de5ddce.js       | main          | 120.72 kB |                34.05 kB
polyfills.250519e9480a24a5.js  | polyfills     |  33.08 kB |                10.63 kB
runtime.2c4c3f0bc3b125c2.js    | runtime       |   1.10 kB |               613 bytes

| Initial Total | 312.56 kB |                62.30 kB

Build at: 2022-04-17T19:19:58.398Z - Hash: 1ac9ef5be6a5ef56 - Time: 94971ms
Removing intermediate container 57c964dc8b3b
 ---> 753781cc1591
Step 7/9 : FROM nginx:1.17.1-alpine
 ---> ea1193fd3dde
Step 8/9 : COPY nginx.conf /etc/nginx/nginx.conf
 ---> Using cache
 ---> 544ff91f7389
Step 9/9 : COPY --from=build /usr/src/app/dist/scientific-calculator-using-dev-ops-tools /usr/share/nginx/html
 ---> Using cache
 ---> 58b3792645f7
Successfully built 58b3792645f7
Successfully tagged khushalabrol2000/scientific_calculator:latest
```

Fig: Docker Image Build Stage



```
[Pipeline] { (Push Docker Image to Dockerhub)
[Pipeline] script
[Pipeline] {
[Pipeline] withEnv
[Pipeline] {
[Pipeline] withDockerRegistry
$ docker login -u khushalabrol2000 -p ******** https://index.docker.io/v1/
WARNING! Using --password via the CLI is insecure. Use --password-stdin.
WARNING! Your password will be stored unencrypted in /var/lib/jenkins/workspace/Scientific Calculator Using DevOps Tools@2@tmp/28563bce-ca4e-45dc-aba0-21bb579f6ba9/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
[Pipeline] {
[Pipeline] isUnix
[Pipeline] withEnv
[Pipeline] {
[Pipeline] sh
+ docker tag khushalabrol2000/scientific_calculator:latest khushalabrol2000/scientific_calculator:latest
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] isUnix
[Pipeline] withEnv
[Pipeline] {
[Pipeline] sh
+ docker push khushalabrol2000/scientific_calculator:latest
The push refers to repository [docker.io/khushalabrol2000/scientific_calculator]
7827b51cb579: Preparing
a3dad213ede7: Preparing
fbe0fc9bcf95: Preparing
f1b5933fe4b5: Preparing
a3dad213ede7: Layer already exists
7827b51cb579: Layer already exists
fbe0fc9bcf95: Layer already exists
f1b5933fe4b5: Layer already exists
latest: digest: sha256:f4a045f17d2e11803efe3087fa3f6ca1d117e8df0e634bb4fc0955f96a90b239 size: 1155
```

Fig: Docker Image Push Stage



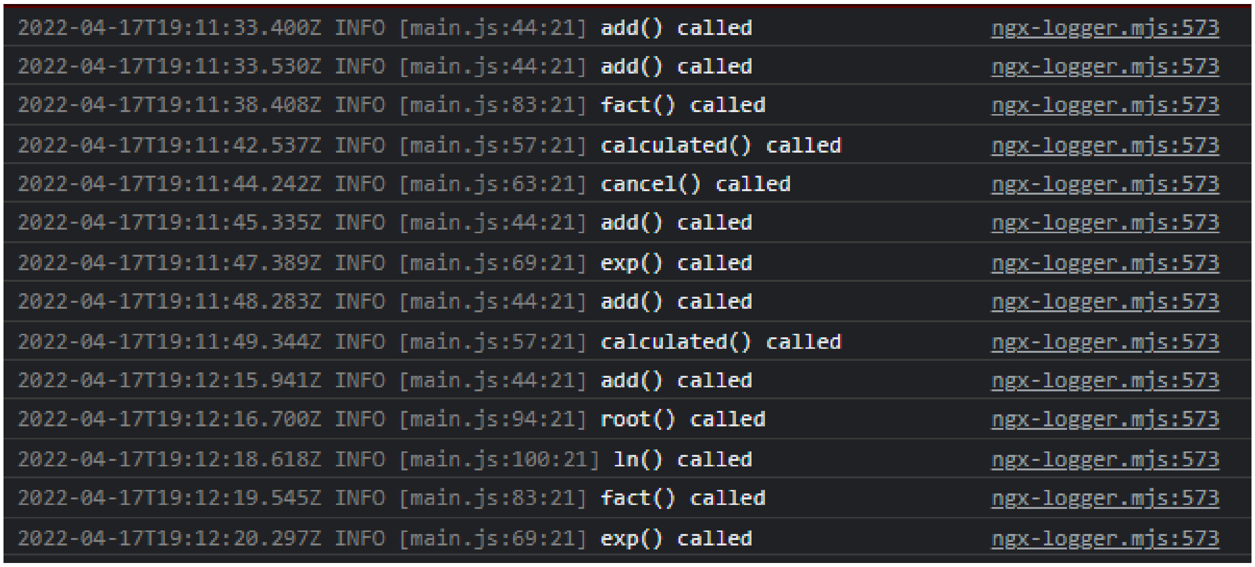
```
[Pipeline] { (Deployment using Ansible)
[Pipeline] ansiblePlaybook
[Scientific Calculator Using DevOps Tools@2] $ ansible-playbook deploy-image.yml -i inventory

PLAY [Pull docker image of Scientific Calculator] ******************************

TASK [Gathering Facts] *********************************************************
⍰[0;32mok: [server1]⍰[0m
⍰[0;32m⍰[0m
TASK [Stop Docker container] ***************************************************
⍰[0;33mchanged: [server1]⍰[0m
⍰[0;33m⍰[0m
TASK [Remove Docker container] *************************************************
⍰[0;33mchanged: [server1]⍰[0m
⍰[0;33m⍰[0m
TASK [Remove Docker image] *****************************************************
⍰[0;33mchanged: [server1]⍰[0m
⍰[0;33m⍰[0m
TASK [Pull devops image] *******************************************************
⍰[0;33mchanged: [server1]⍰[0m
⍰[0;33m⍰[0m
TASK [deploy calc docker container] ********************************************
⍰[0;33mchanged: [server1]⍰[0m
⍰[0;33m⍰[0m
PLAY RECAP *********************************************************************
⍰[0;33mserver1⍰[0m                    : ⍰[0;32mok=6    ⍰[0m ⍰[0;33mchanged=5    ⍰[0m unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```

Fig: Ansible Deployment Stage

Pipeline ended With **SUCCESS** Message.

Step 7: Logging and Monitoring

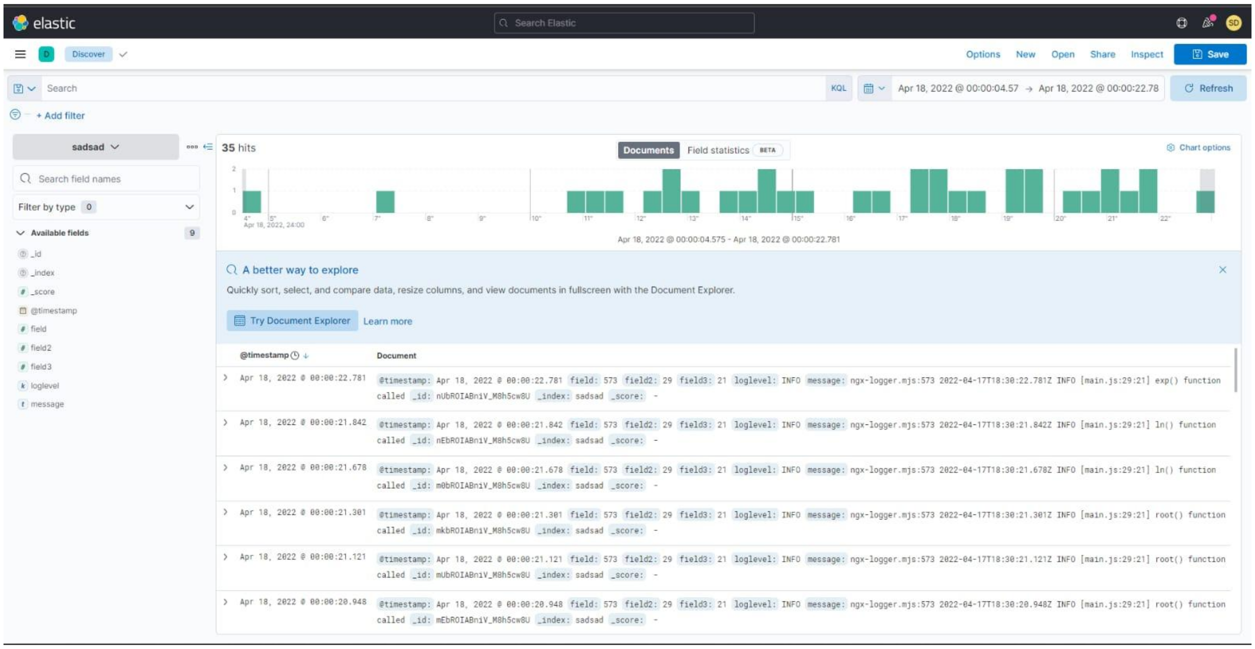
Ngx-logger is use here to do logging of the function called. And all the actions performed.



```
2022-04-17T19:11:33.400Z INFO [main.js:44:21] add() called              ngx-logger.mjs:573
2022-04-17T19:11:33.530Z INFO [main.js:44:21] add() called              ngx-logger.mjs:573
2022-04-17T19:11:38.408Z INFO [main.js:83:21] fact() called             ngx-logger.mjs:573
2022-04-17T19:11:42.537Z INFO [main.js:57:21] calculated() called       ngx-logger.mjs:573
2022-04-17T19:11:44.242Z INFO [main.js:63:21] cancel() called           ngx-logger.mjs:573
2022-04-17T19:11:45.335Z INFO [main.js:44:21] add() called              ngx-logger.mjs:573
2022-04-17T19:11:47.389Z INFO [main.js:69:21] exp() called              ngx-logger.mjs:573
2022-04-17T19:11:48.283Z INFO [main.js:44:21] add() called              ngx-logger.mjs:573
2022-04-17T19:11:49.344Z INFO [main.js:57:21] calculated() called       ngx-logger.mjs:573
2022-04-17T19:12:15.941Z INFO [main.js:44:21] add() called              ngx-logger.mjs:573
2022-04-17T19:12:16.700Z INFO [main.js:94:21] root() called             ngx-logger.mjs:573
2022-04-17T19:12:18.618Z INFO [main.js:100:21] ln() called              ngx-logger.mjs:573
2022-04-17T19:12:19.545Z INFO [main.js:83:21] fact() called             ngx-logger.mjs:573
2022-04-17T19:12:20.297Z INFO [main.js:69:21] exp() called              ngx-logger.mjs:573
```

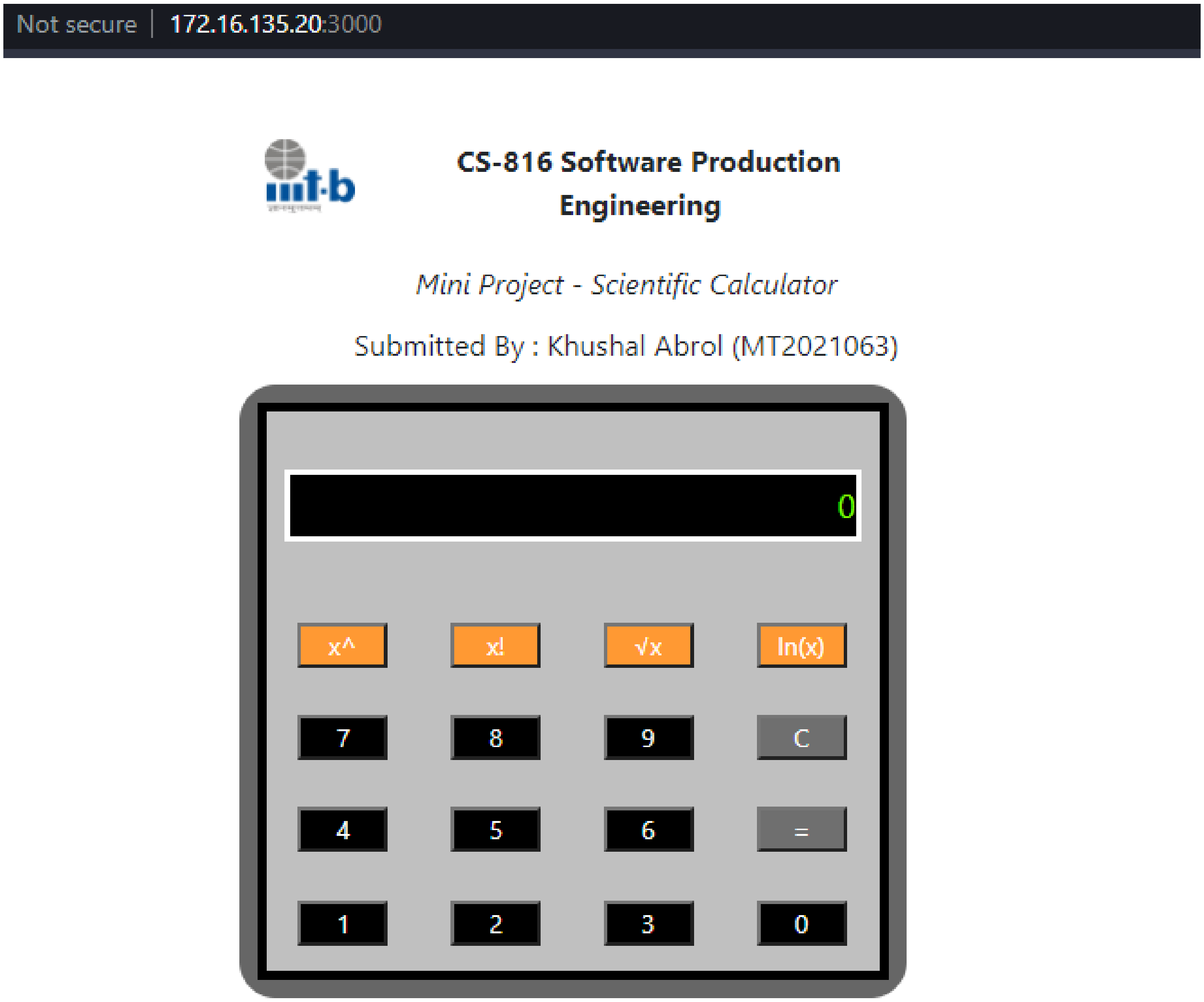
Fig: Chrome console log

ELK (Elasticsearch Logstash Kibana) is a monitoring tool. We have uploaded the app logs on ELK and this is the output.



```
@timestamp                  Document
Apr 18, 2022 @ 00:00:22.781 @timestamp: Apr 18, 2022 @ 00:00:22.781 field: 573 field2: 29 field3: 21 loglevel: INFO message: ngx-logger.mjs:573 2022-04-17T18:30:22.781Z INFO [main.js:29:21] exp() function
                            called _id: nUbROIABniV_M8h5cw8U _index: sadsad _score: -
Apr 18, 2022 @ 00:00:21.842 @timestamp: Apr 18, 2022 @ 00:00:21.842 field: 573 field2: 29 field3: 21 loglevel: INFO message: ngx-logger.mjs:573 2022-04-17T18:30:21.842Z INFO [main.js:29:21] ln() function
                            called _id: nEbROIABniV_M8h5cw8U _index: sadsad _score: -
Apr 18, 2022 @ 00:00:21.678 @timestamp: Apr 18, 2022 @ 00:00:21.678 field: 573 field2: 29 field3: 21 loglevel: INFO message: ngx-logger.mjs:573 2022-04-17T18:30:21.678Z INFO [main.js:29:21] ln() function
                            called _id: m0bROIABniV_M8h5cw8U _index: sadsad _score: -
Apr 18, 2022 @ 00:00:21.301 @timestamp: Apr 18, 2022 @ 00:00:21.301 field: 573 field2: 29 field3: 21 loglevel: INFO message: ngx-logger.mjs:573 2022-04-17T18:30:21.301Z INFO [main.js:29:21] root() function
                            called _id: mkbROIABniV_M8h5cw8U _index: sadsad _score: -
Apr 18, 2022 @ 00:00:21.121 @timestamp: Apr 18, 2022 @ 00:00:21.121 field: 573 field2: 29 field3: 21 loglevel: INFO message: ngx-logger.mjs:573 2022-04-17T18:30:21.121Z INFO [main.js:29:21] root() function
                            called _id: mUbROIABniV_M8h5cw8U _index: sadsad _score: -
Apr 18, 2022 @ 00:00:20.948 @timestamp: Apr 18, 2022 @ 00:00:20.948 field: 573 field2: 29 field3: 21 loglevel: INFO message: ngx-logger.mjs:573 2022-04-17T18:30:20.948Z INFO [main.js:29:21] root() function
                            called _id: mEbROIABniV_M8h5cw8U _index: sadsad _score: -
```

Output: Scientific Calculator: [Github Repository Link]



Not secure | 172.16.135.20:3000

CS-816 Software Production Engineering

Mini Project - Scientific Calculator

Submitted By : Khushal Abrol (MT2021063)

Fig: Deployed and running at server