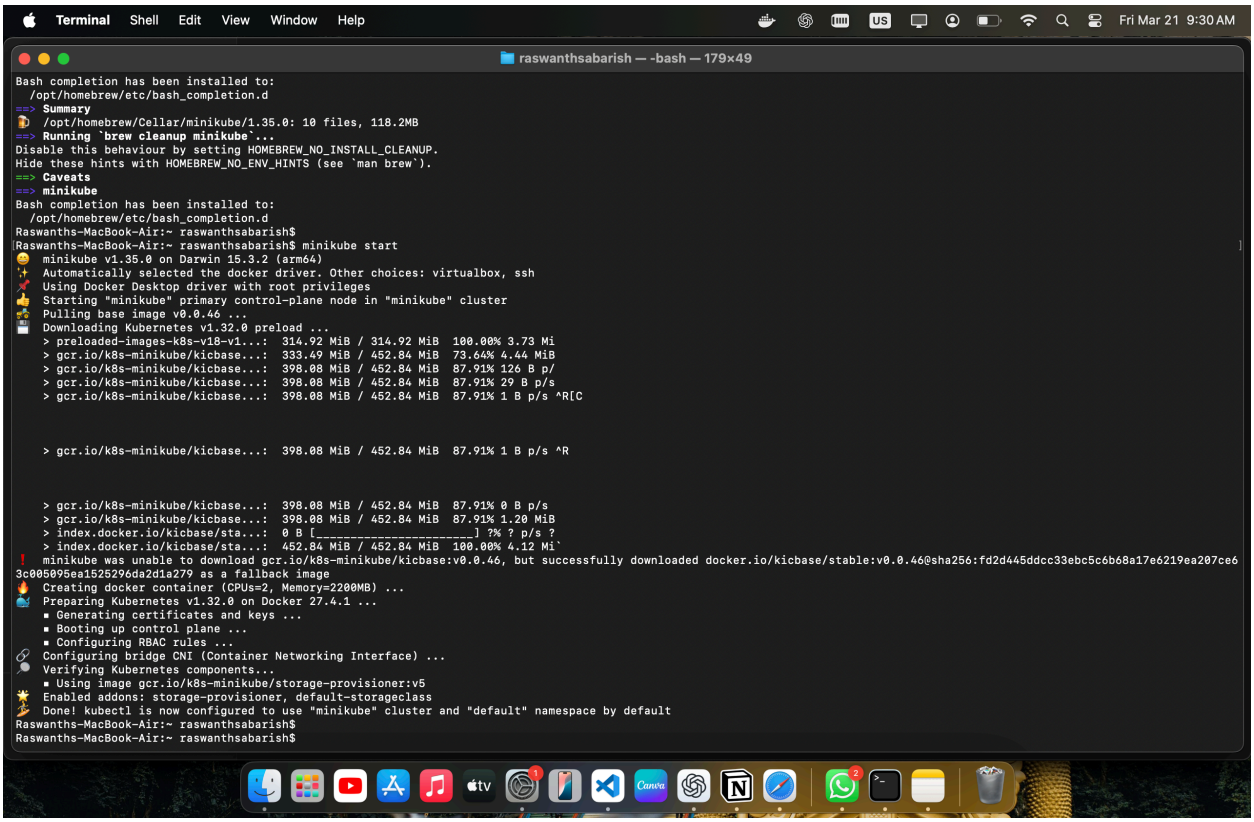
Devops class guvi (DAY-3)

19 March 2025 (DAY-3)

I**nstall Minikube**- on macOS using Homebrew:



wsl tool. (only for windows)

Installing java- sudo apt install fontconfig openjdk-17-jre java -version

Installing Jenkins on Ubuntu/Debian

- Follow the official Jenkins installation guid Jenkins Installation Guide
- Restart and check Jenkins service status

  sudo service jenkins restart

  sudo service jenkins status

Installing Docker

sudo apt install docker.io -y

sudo service docker restart

sudo service docker status

1. Add user to the Docker group
   a. sudo usermod -aG docker $USER
2. Check Docker images and running containers.
   a. sudo chmod 666 /var/run/docker.sock

Installing Kubernetes (kubectl)

Download and install kubectl.

curl -LO https://dl.k8s.io/release/v1.32.0/bin/linux/amd64/kubectl

sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl

chmod +x kubectl

mkdir -p ~/.local/bin

mv ./kubectl ~/.local/bin/kubectl

kubectl version --client

Installing Minikube (Kubernetes)

Download and install Minikube

```
curl -LO https://github.com/kubernetes/minikube/releases/latest/download/minikube-linux-amd64
```

```
sudo install minikube-linux-amd64 /usr/local/bin/minikube && rm minikube-linux-amd64
```

Start Minikube and check status.

```
minikube start
```

```
minikube status
```

Check Kubernetes resources.

```
kubectl get pod
```

```
kubectl get deploy
```

```
kubectl get replica
```

```
kubectl get pod -o wide
```

Docker Compose (Managing Multi-Container Applications)

Install Docker Compose.

```
sudo apt install docker-compose -y
```

Download the latest Docker Compose binary.

```
sudo curl -L "https://github.com/docker/compose/releases/latest/download/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
```

Example docker-compose.yml file for running **NGINX and MySQL**.

yaml code:

```
version: '3'

services:
web:
image: nginx:latest
ports:
- 80:80
```

```
db:
image: mysql:latest
environment:
- MYSQL_ROOT_PASSWORD=secret
```

Running MySQL Inside Docker Container

Enter the MySQL container shell.

```
docker exec -it david-db-1 /bin/bash
```

Login to MySQL

```
mysql -u root -p
```

Jenkins Workspace and Maven Build Location

Path where Jenkins builds and stores the .war file.

```
/var/lib/jenkins/workspace/maven/target/my-app.war
```

Pipelining code for Tomcat

```
pipeline {
agent any
environment {
   DOCKER_CREDENTIALS = credentials('docker-hub-cred')  // Docker Hub
Credentials ID
}
stages {
   stage('SCM') {
      steps {
         git branch: 'main', url: '<https://github.com/MugeshS-
04/guvidevopsday1.git>'
      }
   }
   stage('Build') {
      steps {
```

```
            sh "mvn clean"
            sh "mvn install"
        }
    }
    stage('Build Docker Image') {
        steps {
            script {
                sh 'docker build -t mugeshs04/guvidevopsday1 .'
            }
        }
    }
    stage('Push to Docker Hub') {
        steps {
            script {
                docker.withRegistry('<https://index.docker.io/v1/>', 'docker-hub-
cred') {
                    sh 'docker push mugeshs04/guvidevopsday1'
                }
            }
        }
    }
}}
```