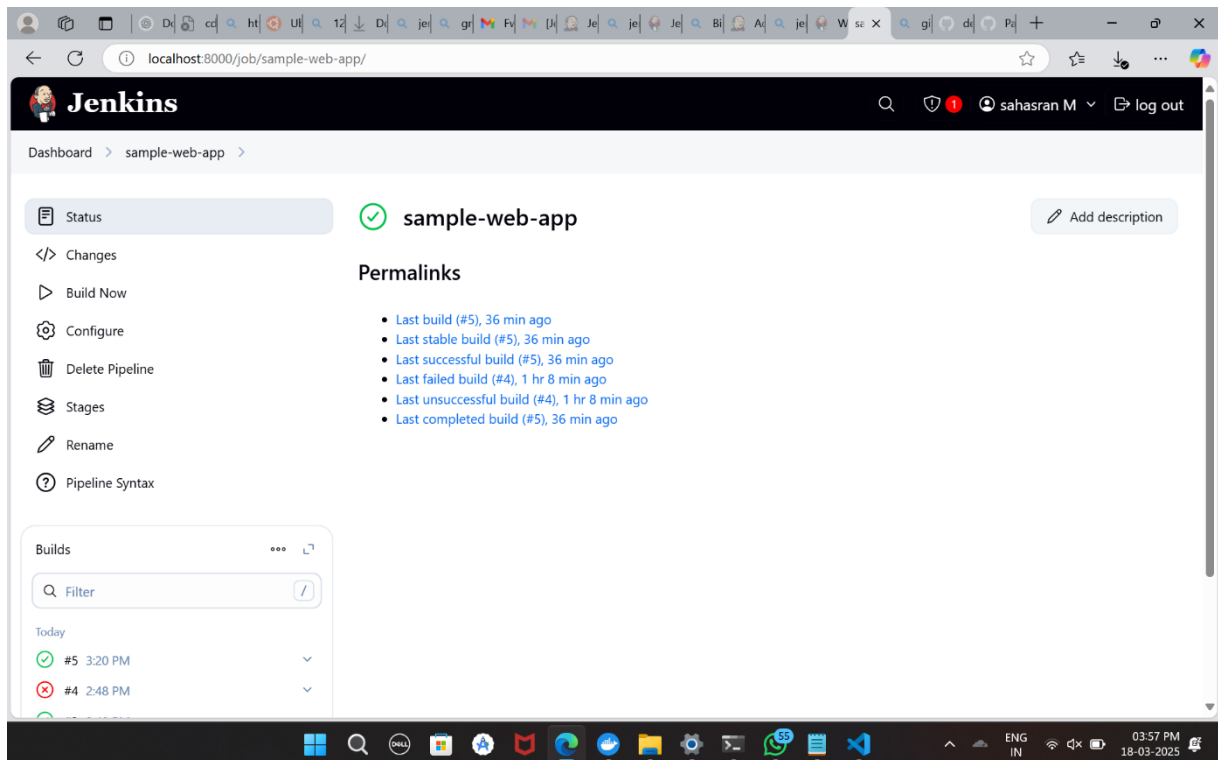


DevOps-Day 06:

SAHASRAN M 22CSR167- III CSE

Devops class guvi (DAY-6)

22 March 2025 - MiniKube Project




```
ubuntu@DESKTOP-MJGHIPO: ~$ minikube ip
92.168.49.2

ubuntu@DESKTOP-MJGHIPO: ~$ minikube status
minikube
type: Control Plane
host: Stopped
kubeadm: Stopped
apiserver: Stopped
kubeproxy: Stopped

ubuntu@DESKTOP-MJGHIPO: ~$ minikube start
minikube v1.35.0 on Ubuntu 24.04 (amd64)
Using the docker driver based on existing profile
Starting "minikube" primary control-plane node in "minikube" cluster
Pulling base image v0.0.46 ...
Restarting existing docker container for "minikube" ...
Preparing Kubernetes v1.32.0 on Docker 27.4.1 ...
Verifying Kubernetes components...
  Using image docker.io/kubernetesui/dashboard:v2.7.0
  Using image docker.io/kubernetesui/metrics-scraper:v1.0.8
  Using image gcr.io/k8s-minikube/storage-provisioner:v5
Some dashboard features require the metrics-server addon. To enable all features please run:

  minikube addons enable metrics-server

Enabled addons: storage-provisioner, dashboard, default-storageclass
Done! kubect is now configured to use "minikube" cluster and "default" namespace by default
ubuntu@DESKTOP-MJGHIPO: ~$ kubectl config current-context
minikube
ubuntu@DESKTOP-MJGHIPO: ~$ kubectl cluster-info
Kubernetes control plane is running at https://127.0.0.1:32769
CoreDNS is running at https://127.0.0.1:32769/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
ubuntu@DESKTOP-MJGHIPO: ~$ kubectl config set-cluster minikube --server=https://<minikube-ip>:<port>
bash: syntax error near unexpected token `newline'
ubuntu@DESKTOP-MJGHIPO: ~$ kubectl config set-cluster minikube --server=https://192.168.49.2:8443
command 'Kubernetes' not found, did you mean:
  command 'kubernetes' from deb kubernetes (1.0)
ubuntu@DESKTOP-MJGHIPO: ~$ kubectl config set-cluster minikube --server=https://192.168.49.2:8443
cluster "minikube" set.
ubuntu@DESKTOP-MJGHIPO: ~$ kubectl get node
NAME                STATUS    ROLES    AGE     VERSION
minikube            Ready    control-plane   2d1h   v1.32.0

E0321 08:09:09.801688 2737 memcache.go:265] "Unhandled Error" err="couldn't get current server
E0321 08:09:09.804085 2737 memcache.go:265] "Unhandled Error" err="couldn't get current server
E0321 08:09:09.806448 2737 memcache.go:265] "Unhandled Error" err="couldn't get current server
E0321 08:09:09.808472 2737 memcache.go:265] "Unhandled Error" err="couldn't get current server
The connection to the server 127.0.0.1:32769 was refused - did you specify the right host or port?
ubuntu@DESKTOP-MJGHIPO: ~$ minikube status
minikube
type: Control Plane
host: Stopped
kubeadm: Stopped
apiserver: Stopped
kubeproxy: Stopped

ubuntu@DESKTOP-MJGHIPO: ~$ minikube start
minikube v1.35.0 on Ubuntu 24.04 (amd64)
Using the docker driver based on existing profile
Starting "minikube" primary control-plane node in "minikube" cluster
Pulling base image v0.0.46 ...
Restarting existing docker container for "minikube" ...
Preparing Kubernetes v1.32.0 on Docker 27.4.1 ...
Verifying Kubernetes components...
  Using image docker.io/kubernetesui/dashboard:v2.7.0
  Using image docker.io/kubernetesui/metrics-scraper:v1.0.8
  Using image gcr.io/k8s-minikube/storage-provisioner:v5
Some dashboard features require the metrics-server addon. To enable all features please run:

  minikube addons enable metrics-server
```

```
saha@SAHASRAN: ~$ minikube start
apiserver: Stopped
kubeconfig: Stopped

saha@SAHASRAN:~$ minikube start
minikube v1.35.0 on Ubuntu 22.04 (amd64)
Using the docker driver based on existing profile
Starting "minikube" primary control-plane node in "minikube" cluster
Pulling base image v0.0.46 ...
Restarting existing docker container for "minikube" ...
Preparing Kubernetes v1.32.0 on Docker 27.4.1 ...
Verifying Kubernetes components...
  * Using image gcr.io/k8s-minikube/storage-provisioner:v5
Enabled addons: default-storageclass, storage-provisioner
Done! kubectrl is now configured to use "minikube" cluster and "default" namespace by default
saha@SAHASRAN:~$ minikube service my-service -n default
✖ Exiting due to SVC_NOT_FOUND: Service 'my-service' was not found in 'default' namespace.
You may select another namespace by using 'minikube service my-service -n <namespace>'. Or list out all the services using 'minikube service list'

saha@SAHASRAN:~$ minikube service list
+-----+-----+-----+-----+
| NAMESPACE | NAME      | TARGET PORT | URL      |
+-----+-----+-----+-----+
| default    | kubernetes | No node port |          |
| kube-system | kube-dns   | No node port |          |
+-----+-----+-----+-----+
```

```
127.0.0.1:46509/my-web/ - Brave
http://127.0.0.1:46509/my-web/
kavin's bookmark Ragu bookmarks ServiceNow - Now L... Date and Time Sele... Microsoft Certified...
Hello World!
```

Commands:

```
jenkins ALL=(ALL) NOPASSWD: ALL sudo systemctl restart ssh.service sudo systemctl restart
sshd.service sudo apt update sudo apt install openssh-server sudo systemctl restart ssh
sudo systemctl status ssh ls /etc/systemd/system/sshd.service or ls
/usr/lib/systemd/system/sshd.service sudo systemctl daemon-reload sudo systemctl status
ssh sudo systemctl restart ssh.service cat /home/david/.minikube/ca.crt | base64 -w 0;
echo sudo chmod 666 /var/run/docker.sock
[https://192.168.39.226:8443](https://192.168.39.226:8443/) sh
'kubectl apply -f deployment.yml --validate=false' minikube service my-
service --url | xargs curl
```

Pipeline codes:

```
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 {
            docker.withRegistry('<https://index.docker.io/v1/>', 'docker-hub-cred') {
                script {
                    'docker push mugeshs04/guvidevopsday1'
                }
            }
        }
    }
}

}

```

```

pipeline { agent
    any

```

```

    stages {
        stage('SCM') {
            steps {
                git branch: 'main', url:
                '<https://github.com/PraneshC2005/DevOps_simpleweb-app.git>'
            }
        }

        stage('Build-clean') {
            steps{
                sh 'mvn clean'
            }
        }

        stage('Build-validate') {
            steps{
                sh 'mvn validate'
            }
        }
    }
}

```

```

stage('Build-compile') {
    steps{
        sh 'mvn compile'
    }
}
stage('Build-test') {
    steps{
        sh 'mvn test'
    }
}
stage('Build-package') {
    steps{
        sh 'mvn package'
    }
}
stage('build to images') {
    steps {
        script{
            sh "docker build
-t praneshc/webapplication ."
        }
    }
}
stage('docker push hub') {
    steps {
        script{
            withDockerRegistry(credentialsId: 'cred-2', url:
'<https://index.docker.io/v1/>') {
                sh 'docker push praneshc/webapplication'
            }
        }
    }
}
}
}

```