



INDIAN INSTITUTE OF
INFORMATION
TECHNOLOGY

CS457

DEVOPS TASK - 2

Submitted to:

Dr Uma S

Submitted by:

TEAM 8

Akshaya Reddy 18BCS017

Bhavya Tripathi 18BCS019

Harshitha M 18BCS032

Meghana N 18BCS053

S Namratha 18BCS083

Vinita Yadav 18BCS109

Deepanshu Sachdeva 18BCS114

DEVOPS ASSIGNMENT

TEAM 8

TASK1

Part B

Q. Developing and deploying a Node.js app from Docker to Kubernetes

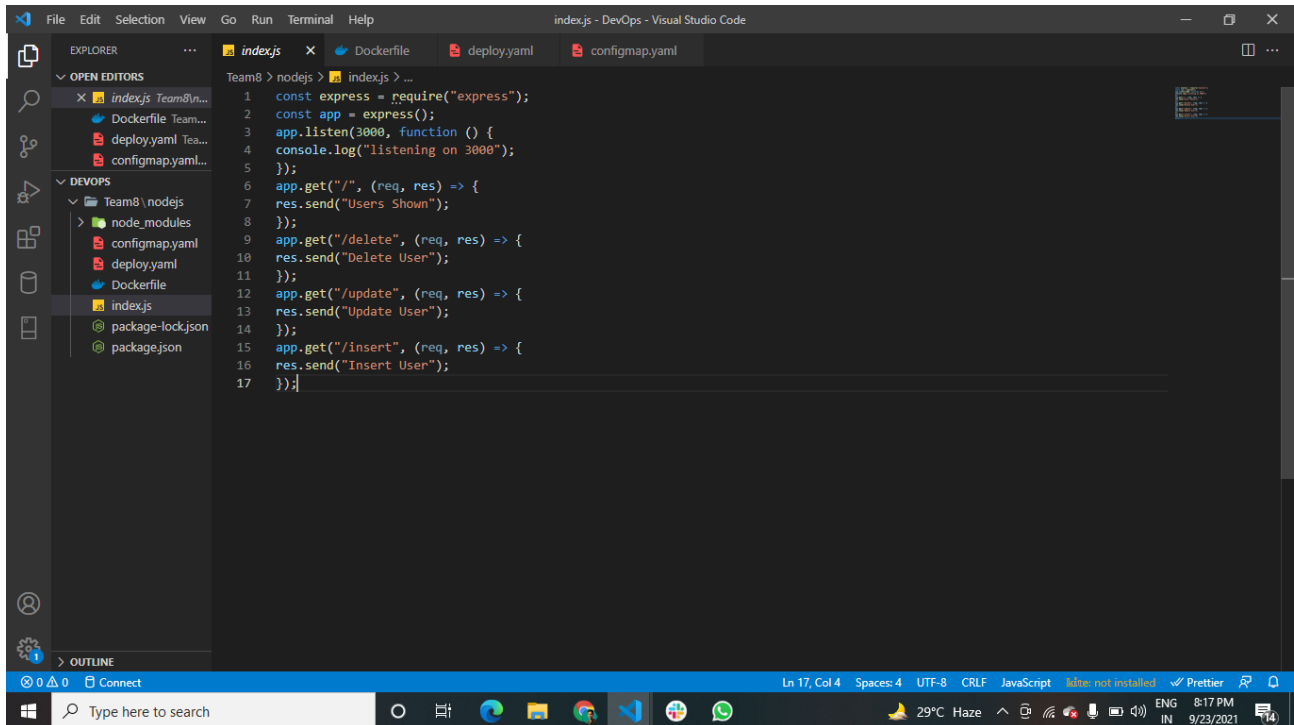
You can find the docker file here :

<https://hub.docker.com/r/deepanshusachdeva5/nodejs-starter>

Github link -

https://github.com/deepanshusachdeva5/CS457_DEVOPS/tree/main/Assignment2

STEP1: NodeJs App

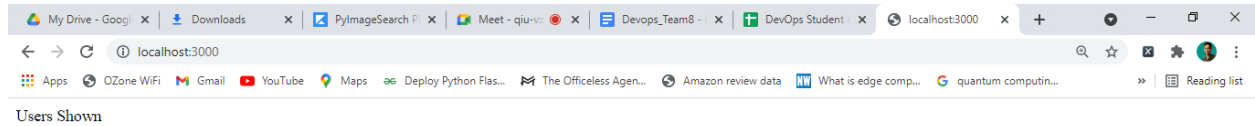


The screenshot shows the Visual Studio Code editor with a project named 'indexjs - DevOps'. The Explorer sidebar on the left shows the file structure: 'Team8 > nodejs > indexjs'. The main editor area displays the 'index.js' file with the following code:

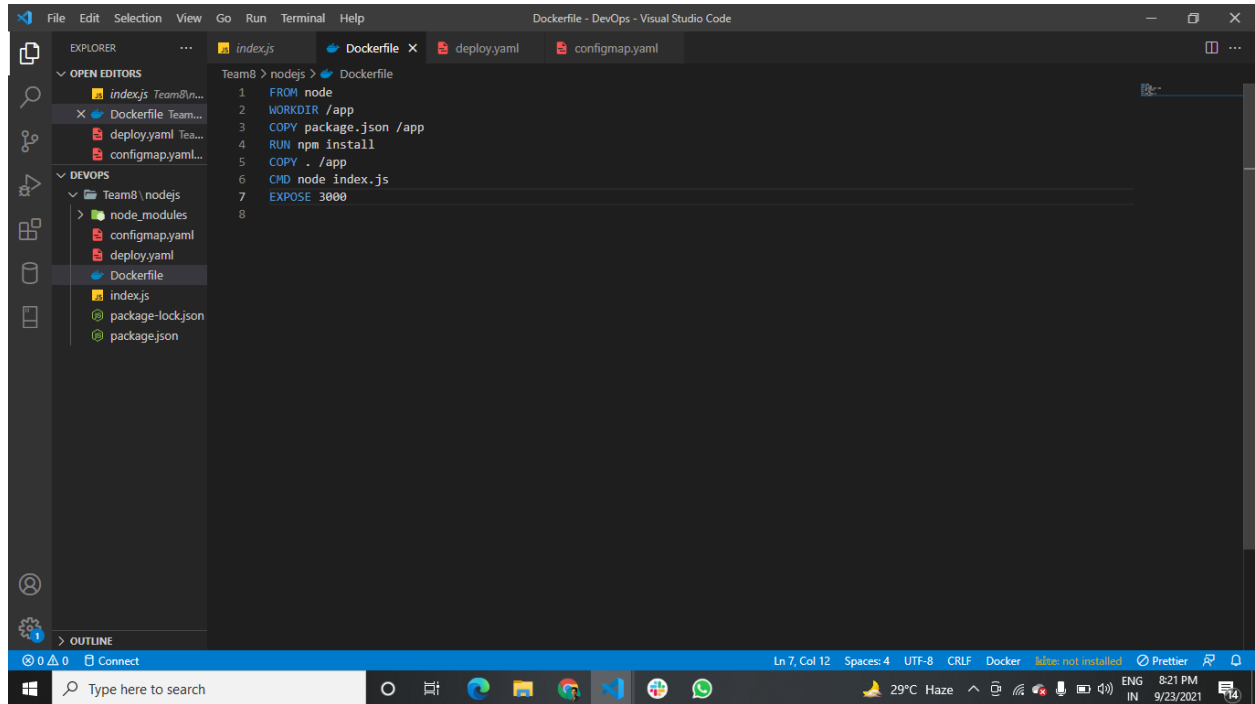
```
1 const express = require("express");
2 const app = express();
3 app.listen(3000, function () {
4   console.log("listening on 3000");
5 });
6 app.get("/", (req, res) => {
7   res.send("Users Shown");
8 });
9 app.get("/delete", (req, res) => {
10  res.send("Delete User");
11 });
12 app.get("/update", (req, res) => {
13  res.send("Update User");
14 });
15 app.get("/insert", (req, res) => {
16  res.send("Insert User");
17 });
```

The status bar at the bottom indicates the current file is 'index.js' at line 17, column 4, using UTF-8 encoding and CRLF line endings. The system tray at the bottom shows the date and time as 9/23/2021, 8:17 PM.

STEP2: Testing NodeJs App using npm start



STEP3: Writing Dockerfile



STEP4: Building Docker Image using docker build -t node-starter

The screenshot shows the Visual Studio Code interface with a file explorer on the left showing 'index.js', 'Dockerfile', and 'deploy.yaml'. The 'deploy.yaml' file is open in the editor, showing a deployment configuration for 'nodejs-deployment'. The terminal window at the bottom shows the output of the 'docker build' command. It starts with '[internal] load .dockerignore', followed by 'failed to solve with frontend dockerfile.v0: failed to read dockerfile: open /var/lib/docker/tmp/buildkit-mount448437805/Dockerfile: no such file or directory'. The user then runs 'cd .\nodejs\' and 'docker build -t node-server'. The build process shows the resolution of the Dockerfile, the transfer of context, and the pulling of the 'node:latest' image from Docker Hub. The final output shows the image 'node-server' is built successfully.

```
Team8 > nodejs > deploy.yaml
1  apiVersion: apps/v1 #1
2  kind: Deployment #2
3  metadata: #3
4  name: nodejs-deployment #4

[+] Building 57.6s (5/10)
-> [internal] load build definition from Dockerfile 0.0s
-> [internal] load .dockerignore 0.0s
-> [internal] load metadata for docker.io/library/node:latest 5.6s
-> [auth] library/node:pull token for registry-1.docker.io 0.0s
-> [1/5] FROM docker.io/library/node@sha256:8710efa3fc8be1cbce674c646c1c64a993a70ca767bd50bae81e232855261973 51.7s
-> resolve docker.io/library/node@sha256:8710efa3fc8be1cbce674c646c1c64a993a70ca767bd50bae81e232855261973 0.1s
-> sha256:b061a801a0f3c17c437c67c49e20c76c8735e205cd165e56ae4fa867f32af1 2.21kB / 2.21kB 0.0s
-> sha256:93f5457b802e5d969ff33aac9fd499fccbef17b9b1d753e701c8cc5c7467dc2d 7.60kB / 7.60kB 0.0s
-> sha256:8710efa3fc8be1cbce674c646c1c64a993a70ca767bd50bae81e232855261973 1.21kB / 1.21kB 0.0s
-> sha256:8f04e8168e3873638397ca4beb7d8484b150eca0d10fe1b033a125202ba57692 9.44MB / 50.44MB 51.6s
-> sha256:82e5f66f5d0e1c97622f33d44fb04efb42bd3562bdc3482537d121040c789f9a 7.83MB / 7.83MB 35.5s
-> sha256:c1c8f1c77d6674046d7deb41be1ca07f25cb43f067f8e79ee79cc586887f0 10.00MB / 10.00MB 44.1s
-> sha256:5095cab277710f0c2883844158323ad986c763ffc37353ddff874dd85585d9b6 1.05MB / 51.84MB 45.9s
-> sha256:ea7fe362a971515971cf53613a30cc824f94d544272a5e061eb6365923ccbc11 0B / 192.39MB 45.9s
-> [internal] load build context 0.0s
-> [internal] load build context: 1.74MB 0.5s
```

STEP5: Running Docker Image

The screenshot shows the Visual Studio Code interface with the 'deploy.yaml' file open. The terminal window at the bottom shows the output of the 'docker run' command. It starts with 'PS C:\Users\HP\Desktop\DevOps\Team8\nodejs> docker run -d --name nodongo -p 3000:3000 node-server'. The output shows the container 'e8400dfb8bf99ac81abb077e8aa6e44e879e2568eb523fbf800358d887d356c5' is created. The user then runs 'docker login' and provides credentials. The output shows the login is successful. The user then runs 'docker tag node-server deepanshusachdeva5/nodejs-starter' and 'docker push deepanshusachdeva5/nodejs-starter:1.1'. The output shows the push is successful, with the image 'c4c3fa1f2b7a' being pushed to the repository 'docker.io/deepanshusachdeva5/nodejs-starter'.

```
PS C:\Users\HP\Desktop\DevOps\Team8\nodejs> docker run -d --name nodongo -p 3000:3000 node-server
e8400dfb8bf99ac81abb077e8aa6e44e879e2568eb523fbf800358d887d356c5
Removing login credentials for https://index.docker.io/v1/
PS C:\Users\HP\Desktop\DevOps\Team8\nodejs> docker login
Login with your Docker ID to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.com to create one.
Username: deepanshusachdeva5
Password:
Login Succeeded
PS C:\Users\HP\Desktop\DevOps\Team8\nodejs> docker tag node-server deepanshusachdeva5/nodejs-starter
PS C:\Users\HP\Desktop\DevOps\Team8\nodejs> docker push deepanshusachdeva5/nodejs-starter:1.1
The push refers to repository [docker.io/deepanshusachdeva5/nodejs-starter]
tag does not exist: deepanshusachdeva5/nodejs-starter:1.1
PS C:\Users\HP\Desktop\DevOps\Team8\nodejs> docker push deepanshusachdeva5/nodejs-starter
Using default tag: latest
The push refers to repository [docker.io/deepanshusachdeva5/nodejs-starter]
c4c3fa1f2b7a: Pushed
55506e955bc5: Pushed
2b2948de99aa: Pushed
120a348a2b30: Pushed
3ebf0ae2cdfb: Layer already exists
d95faf3ef1b0: Layer already exists
0e681c4be8c0: Layer already exists
ac5ca208761a: Layer already exists
6db3d1445d34: Layer already exists
f36cbdef5e3a: Layer already exists
1c721e0a3622: Layer already exists
6680d5caa999: Layer already exists
4dd0c5812f04: Layer already exists
latest: digest: sha256:22088a234e5372daf10af360d1f7f473979fa458c0c7ddb49208107b7e9bcb1b size: 3049
```

STEP6: Pushing image to docker hub after tagging

The screenshot shows the Visual Studio Code interface with a file named `deploy.yaml` open. The file contains a deployment configuration for a Node.js application. The terminal window shows the execution of `docker run` and `docker push` commands, indicating the successful building and pushing of a Docker image to Docker Hub.

```
1 apiVersion: apps/v1 #1
2 kind: Deployment #2
3 metadata: #3
4   name: nodejs-deployment #4
```

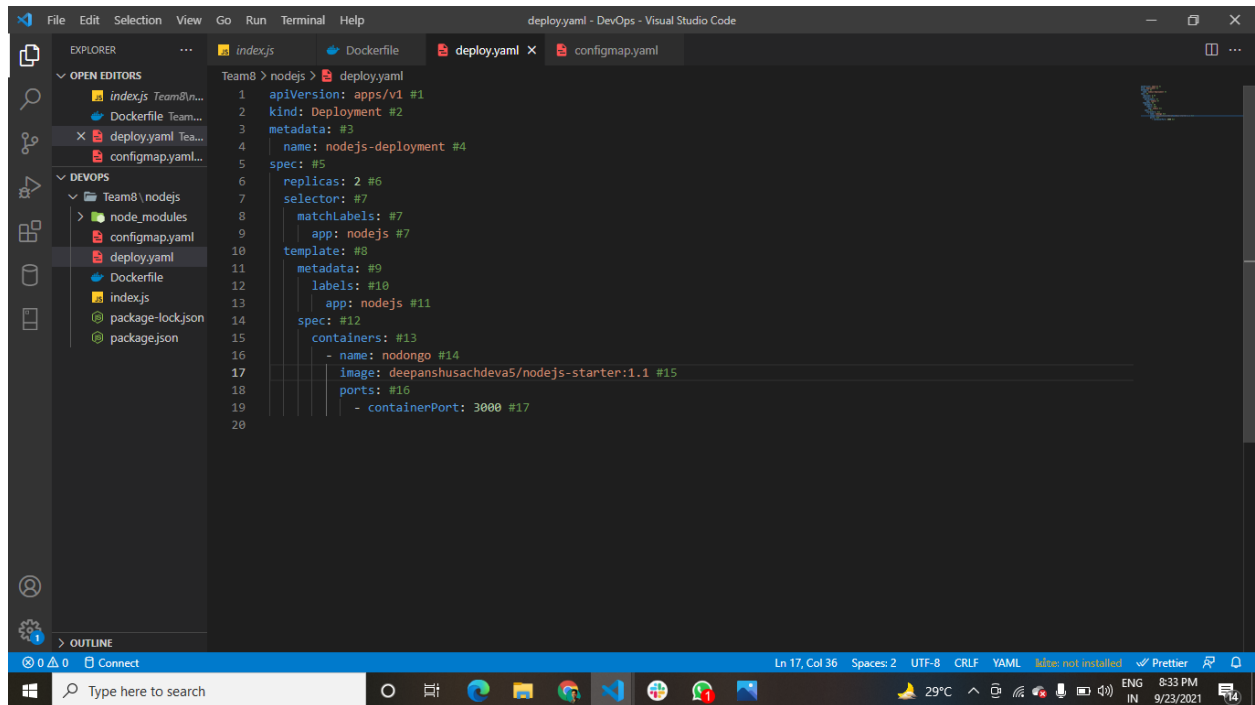
```
PS C:\Users\HP\Desktop\DevOps\Team8\nodejs> docker run -d --name nodongo -p 3000:3000 node-server
e8400dfb8bf99ac01abb077e8aa6e44e879e2568eb523fbf800358d087d356c5
Removing login credentials for https://index.docker.io/v1/
PS C:\Users\HP\Desktop\DevOps\Team8\nodejs> docker login
Login with your Docker ID to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.com to create one.
Username: deepanshusachdeva5
Password:
Login Succeeded
PS C:\Users\HP\Desktop\DevOps\Team8\nodejs> docker tag node-server deepanshusachdeva5/nodejs-starter
PS C:\Users\HP\Desktop\DevOps\Team8\nodejs> docker push deepanshusachdeva5/nodejs-starter:1.1
The push refers to repository [docker.io/deepanshusachdeva5/nodejs-starter]
tag does not exist: deepanshusachdeva5/nodejs-starter:1.1
PS C:\Users\HP\Desktop\DevOps\Team8\nodejs> docker push deepanshusachdeva5/nodejs-starter
Using default tag: latest
The push refers to repository [docker.io/deepanshusachdeva5/nodejs-starter]
c4c3fa1f2b7a: Pushed
55506e955bc5: Pushed
2b2948de99aa: Pushed
120a348a2b30: Pushed
3ebf0ae2cdfb: Layer already exists
d95faf3ef1b0: Layer already exists
0e681c4be8c0: Layer already exists
ac5ca208761a: Layer already exists
6db3d1445d34: Layer already exists
1c721e0a3622: Layer already exists
1c721e0a3622: Layer already exists
6680d5caa699: Layer already exists
4dd0c5812fd4: Layer already exists
latest: digest: sha256:22608a234e5372daf16af360d1f74739379fa458c0c7ddb49200107b7e9bc1b size: 3049
```

STEP7: Minikube Start

The screenshot shows the Visual Studio Code interface with a terminal window displaying the execution of `minikube start` and `kubectl` commands. The output shows the successful installation and configuration of Minikube, followed by the creation of a deployment and the exposure of the service.

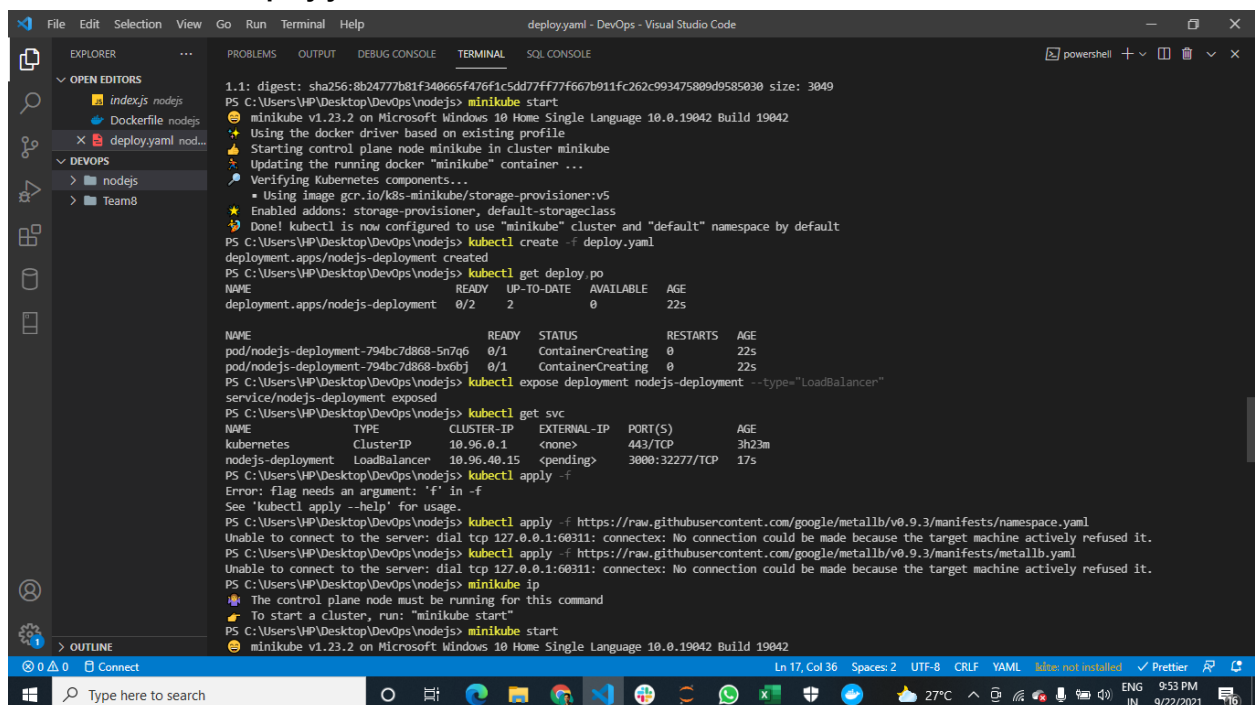
```
PS C:\Users\HP\Desktop\DevOps\nodejs> docker login
Login with your Docker ID to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.com to create one.
Username: deepanshusachdeva5
Password:
Login Succeeded
The push refers to repository [docker.io/deepanshusachdeva5/nodejs-starter]
c6037137d5b6: Pushed
174a12ba5978: Pushed
6e0023cb31fe: Pushed
36bb431f0787: Pushed
3ebf0ae2cdfb: Mounted from library/node
d95faf3ef1b0: Mounted from library/node
0e681c4be8c0: Mounted from library/node
ac5ca208761a: Mounted from library/node
6db3d1445d34: Mounted from library/node
1c721e0a3622: Mounted from library/node
4dd0c5812fd4: Mounted from library/node
1.1: digest: sha256:8b24777b81f340665f476f1c5dd77ff77f667b911fc262c993475809d9585030 size: 3049
PS C:\Users\HP\Desktop\DevOps\nodejs> minikube start
minikube v1.23.2 on Microsoft Windows 10 Home Single Language 10.0.19042 Build 19042
* Using the docker driver based on existing profile
* Starting control plane node minikube in cluster minikube
* Updating the running docker "minikube" container ...
* Verifying Kubernetes components...
  * Using image gcr.io/k8s-minikube/storage-provisioner:v5
  * Enabled addons: storage-provisioner, default-storageclass
  * Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
PS C:\Users\HP\Desktop\DevOps\nodejs> kubectl create -f deploy.yaml
deployment.apps/nodejs-deployment created
PS C:\Users\HP\Desktop\DevOps\nodejs> kubectl get deploy po
NAME READY UP-TO-DATE AVAILABLE AGE
deployment.apps/nodejs-deployment 0/2 2 0 22s
NAME pod/nodejs-deployment-794bc7d868-5n7q6 pod/nodejs-deployment-794bc7d868-bx6bj
READY STATUS RESTARTS AGE
0/1 ContainerCreating 0 22s
0/1 ContainerCreating 0 22s
PS C:\Users\HP\Desktop\DevOps\nodejs> kubectl expose deployment nodejs-deployment --type=LoadBalancer
```

STEP8: Creating deploy.yaml file



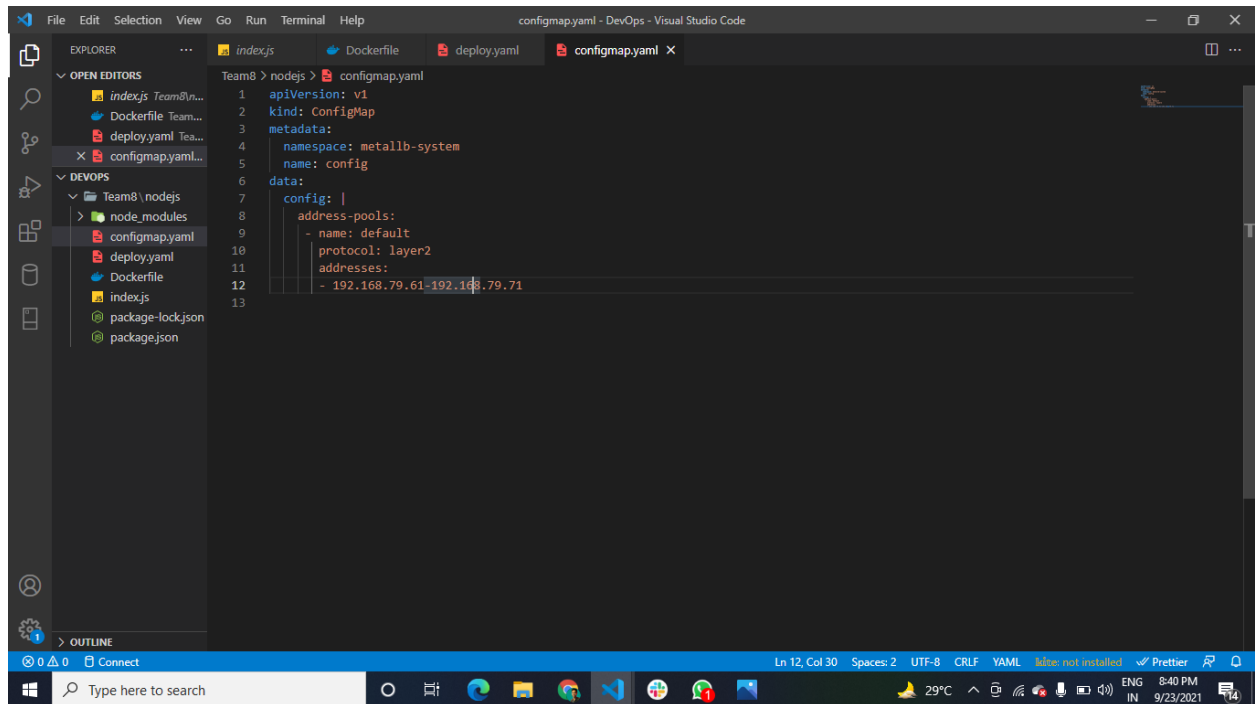
```
1 apiVersion: apps/v1 #1
2 kind: Deployment #2
3 metadata: #3
4   name: nodejs-deployment #4
5 spec: #5
6   replicas: 2 #6
7   selector: #7
8     matchLabels: #7
9     app: nodejs #7
10  template: #8
11    metadata: #9
12      labels: #10
13        app: nodejs #11
14    spec: #12
15      containers: #13
16        - name: nodongo #14
17          image: deepanshusachdeva5/nodejs-starter:1.1 #15
18          ports: #16
19            - containerPort: 3000 #17
```

STEP9: Kubectl -f deploy.yaml

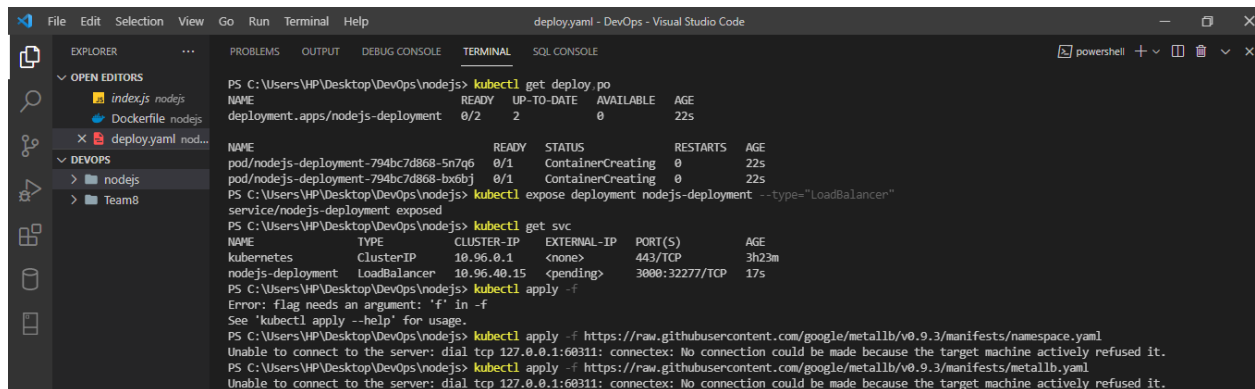


```
1.1: digest: sha256:8b24777b81f348665f476f1c5dd77ff77f667b911fc262c993475809d9585030 size: 3049
PS C:\Users\HP\Desktop\DevOps\nodejs> minikube start
minikube v1.23.2 on Microsoft Windows 10 Home Single Language 10.0.19042 Build 19042
* Using the docker driver based on existing profile
* Starting control plane node minikube in cluster minikube
* Updating the running docker "minikube" container ...
* Verifying Kubernetes components...
  * Using image gcr.io/k8s-minikube/storage-provisioner:v5
  * Enabled addons: storage-provisioner, default-storageclass
  * Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
PS C:\Users\HP\Desktop\DevOps\nodejs> kubectl create -f deploy.yaml
deployment.apps/nodejs-deployment created
PS C:\Users\HP\Desktop\DevOps\nodejs> kubectl get deploy po
NAME READY UP-TO-DATE AVAILABLE AGE
deployment.apps/nodejs-deployment 0/2 2 0 22s
NAME READY STATUS RESTARTS AGE
pod/nodejs-deployment-794bc7d868-5n7q6 0/1 ContainerCreating 0 22s
pod/nodejs-deployment-794bc7d868-bx6bj 0/1 ContainerCreating 0 22s
PS C:\Users\HP\Desktop\DevOps\nodejs> kubectl expose deployment nodejs-deployment --type=LoadBalancer
service/nodejs-deployment exposed
PS C:\Users\HP\Desktop\DevOps\nodejs> kubectl get svc
NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE
kubernetes ClusterIP 10.96.0.1 <none> 443/TCP 3h23m
nodejs-deployment LoadBalancer 10.96.40.15 <pending> 3000:32277/TCP 17s
PS C:\Users\HP\Desktop\DevOps\nodejs> kubectl apply -f
Error: flag needs an argument: 'f' in -f
See 'kubectl apply --help' for usage.
PS C:\Users\HP\Desktop\DevOps\nodejs> kubectl apply -f https://raw.githubusercontent.com/google/metallb/v0.9.3/manifests/namespace.yaml
Unable to connect to the server: dial tcp 127.0.0.1:60311: connect: No connection could be made because the target machine actively refused it.
PS C:\Users\HP\Desktop\DevOps\nodejs> kubectl apply -f https://raw.githubusercontent.com/google/metallb/v0.9.3/manifests/metallb.yaml
Unable to connect to the server: dial tcp 127.0.0.1:60311: connect: No connection could be made because the target machine actively refused it.
PS C:\Users\HP\Desktop\DevOps\nodejs> minikube ip
minikube v1.23.2 on Microsoft Windows 10 Home Single Language 10.0.19042 Build 19042
The control plane node must be running for this command
* To start a cluster, run: "minikube start"
PS C:\Users\HP\Desktop\DevOps\nodejs> minikube start
minikube v1.23.2 on Microsoft Windows 10 Home Single Language 10.0.19042 Build 19042
```

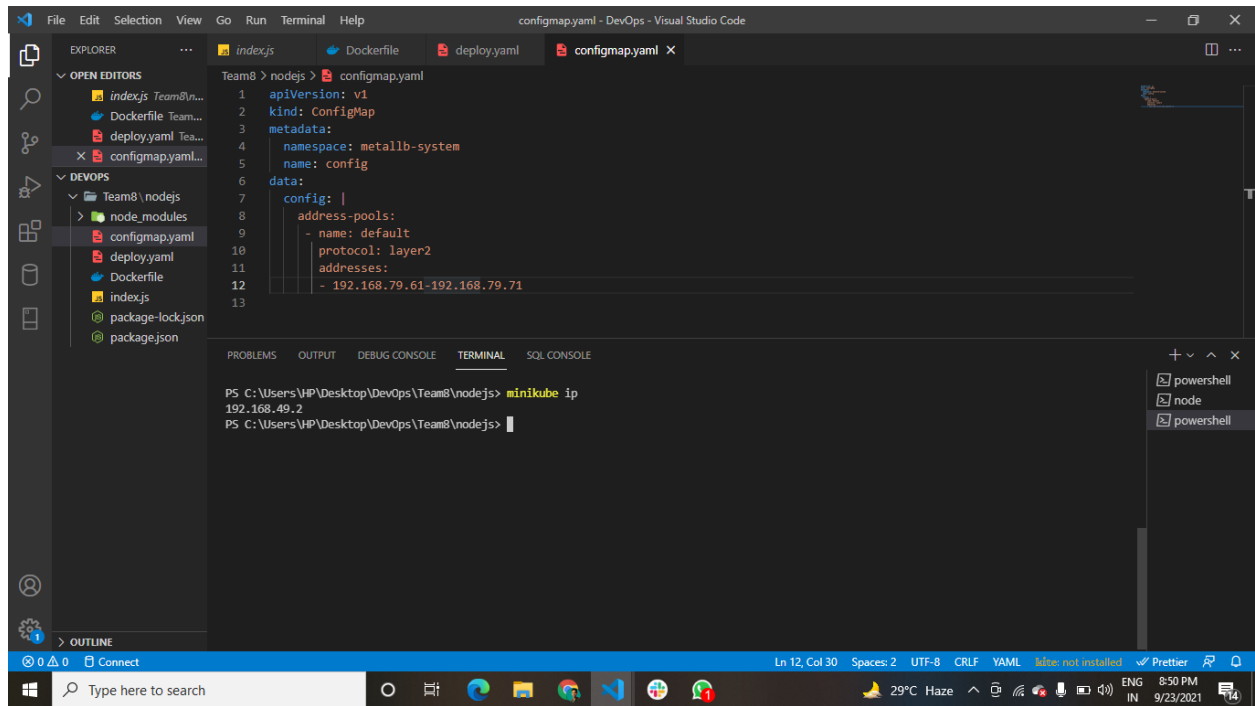
STEP10: configmap.yaml



STEP11: Deploying on kubernetes using minikube



STEP12: Checking minikube ip



STEP13: Verifying external IP

File Edit Selection View Go Run Terminal Help deploy.yaml - Team8 - Visual Studio Code

EXPLORER

OPEN EDITORS

- deploy.yaml nodejs

TEAMS

- nodejs
 - node_modules
 - configmap.yaml
 - deploy.yaml
 - Dockerfile
 - index.js
 - package-lock.json
 - package.json

nodejs > deploy.yaml

```
1 apiVersion: apps/v1 #1
2 kind: Deployment #2
3 metadata: #3
4   name: nodejs-deployment #4
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL SQL CONSOLE

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell <https://aka.ms/pscore6>

```
PS C:\Users\HP\Desktop\DevOps\Team8> kubectl version
Client Version: version.Info{Major:"1", Minor:"22", GitVersion:"v1.22.2", GitCommit:"8b5a19147530eaac9476b0ab82980b4088bbc1b2", GitTreeState:"clean", BuildDate:"2021-09-15T21:38:50Z", GoVersion:"go1.16.8", Compiler:"gc", Platform:"windows/amd64"}
Server Version: version.Info{Major:"1", Minor:"22", GitVersion:"v1.22.2", GitCommit:"8b5a19147530eaac9476b0ab82980b4088bbc1b2", GitTreeState:"clean", BuildDate:"2021-09-15T21:32:41Z", GoVersion:"go1.16.8", Compiler:"gc", Platform:"linux/amd64"}
PS C:\Users\HP\Desktop\DevOps\Team8> kubectl get namespace
NAME          STATUS   AGE
default       Active  11h
kube-node-lease Active  11h
kube-public   Active  11h
kube-system   Active  11h
metallb-system Active  11h
PS C:\Users\HP\Desktop\DevOps\Team8> kubectl get service
NAME          TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)          AGE
kubernetes    ClusterIP   10.96.0.1     <none>         443/TCP           11h
nodejs-deployment LoadBalancer 10.102.240.100 192.168.79.61 3000:31642/TCP    11h
PS C:\Users\HP\Desktop\DevOps\Team8>
```

Ln 19, Col 38 Spaces: 2 UTF-8 CRLF YAML kube: not installed ✓ Prettier

Type here to search

31°C 12:20 PM 9/23/2021

DevOps Assignment - 2

Ansible Playbook Exercise

Set-Up:

- Installing ansible.
- Creating a remote server using AWS EC2 Instance.
Chosen platform: Red Hat Enterprise Linux with High Availability.
- Connected to the server using ssh client.

Step 1: Configuring Git login

Using username and security token.

- `https://user:token@github.com/path`

Step 2: Creating Ansible vault to Store the Git username and token

Creating a vault and setting a vault password and creating a file called `secrets.yml` that stores the GitHub username and security token.

The `secrets.yml` file will be encrypted.

```
Sep 21 22:44
bhavya@bhavya-Inspiron-5570: ~/Desktop/Assignment2
bhavya@bhavya-Inspiron-5570:~/Desktop/Assignment2$ ansible-vault create secrets.yml
New Vault password:
Confirm New Vault password:
bhavya@bhavya-Inspiron-5570:~/Desktop/Assignment2$ cat secrets.yml
$ANSIBLE_VAULT;1.1;AES256
3235333530396666646463396366343232666639306362306661373262663265646264663464633
0
3838623762616435386166343163336539326231306636330a39666635623639656536363433343
5
3633386462363164303334333662633037326333613434356265313630313135396666336635663
6
6663656631383463340a36656232393132323963343861383933333264633466306637363136666
4
3231333231343733336662343135646332633132323139356438356262306532373965376463306
3
3635653432333131323935316161353336333831656464393863383965613231346233393763353
1
6536326263386638653361356262626462363038646436643561613562353737393331336336653
7
65633334366632306533
bhavya@bhavya-Inspiron-5570:~/Desktop/Assignment2$ echo "ENCRYPTED"
ENCRYPTED
```

We can decrypt the file using the following commands:

```
Sep 21 22:48
bhavya@bhavya-Inspiron-5570: ~/Desktop/Assignment2
bhavya@bhavya-Inspiron-5570:~/Desktop/Assignment2$ ansible-vault decrypt secrets.yml
Vault password:
Decryption successful
bhavya@bhavya-Inspiron-5570:~/Desktop/Assignment2$ cat secrets.yml
gituser: Bhavya-Tripathi
gitpass: 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
bhavya@bhavya-Inspiron-5570:~/Desktop/Assignment2$ echo "DECRYPTED"
DECRYPTED
bhavya@bhavya-Inspiron-5570:~/Desktop/Assignment2$
```

Step 3: The Ansible Git Example Playbook

Here we have created a nodejs app and uploaded it in a private GitHub repository.

Then we create the Ansible playbook. (gitexample.yml)

The GitHub username and token we created in the secrets.yml file.

```

- name: Install and Launch the Simple NodeJS Application
  hosts: nodeserver
  vars_files:
    - secrets.yml
  vars:
    - destdir: /apps/SampleNodeApp
  tasks:

    - name : install Node and NPM
      become: yes
      register: ymrepo
      yum:
        name: nodejs
        state: latest

    - name : validate the nodejs installation
      debug: msg="Installation of node is Successfull"
      when: ymrepo is changed

    - name: Version of Node and NPM
      shell:
        "npm -v && node -v"
      register: versioninfo

    - name: Version Info
      debug:
        msg: "Version info {{ versioninfo.stdout_lines }}"
      when: versioninfo is changed

    - name: Download the NodeJS code from the GitRepo
      become: yes
      git:
        repo: 'https://{{gituser}}:{{gitpass}}@github.com/Bhavya-Tripathi/DevOpsAssignment2.git'
        dest: "{{ destdir }}"

    - name: Change the ownership of the directory
      become: yes
      file:
        path: "{{destdir}}"
        owner: "ec2-user"
      register: chgrpout

    - name: Install Dependencies with NPM install command
      shell:
        "npm install"
      args:
        chdir: "{{ destdir }}"
      register: npminstlout

    - name: Debug npm install command
      debug: msg='{{npminstlout.stdout_lines}}'

    - name: Start the App
      async: 10
      poll: 0
      shell:
        "(node index.js > nodesrv.log 2>&1 &)"
      args:
        chdir: "{{ destdir }}"
      register: appstart

    - name: Validating the port is open
      tags: nodevalidate
      wait_for:
        host: "localhost"
        port: 3002
        delay: 10
        timeout: 30
        state: started
        msg: "NodeJS server is not running"

```

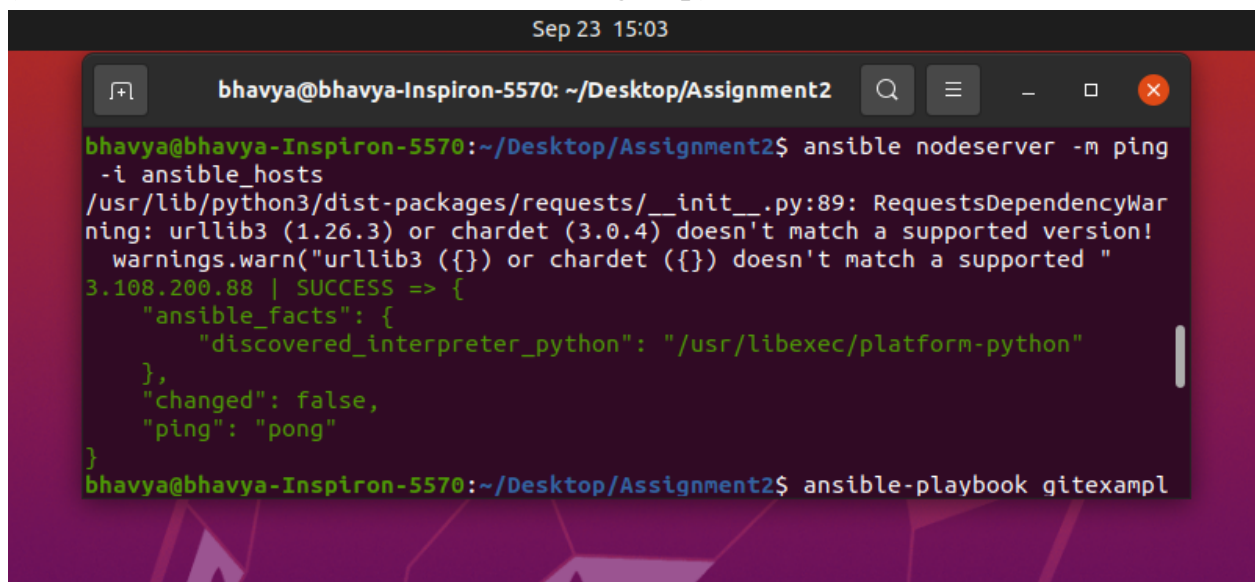
Step 4: Launch the Playbook with Ansible Git

Now we launch the playbook using the `ansible-playbook` command

```
ansible-playbook gitexample.yml --ask-vault-pass
```

Method:

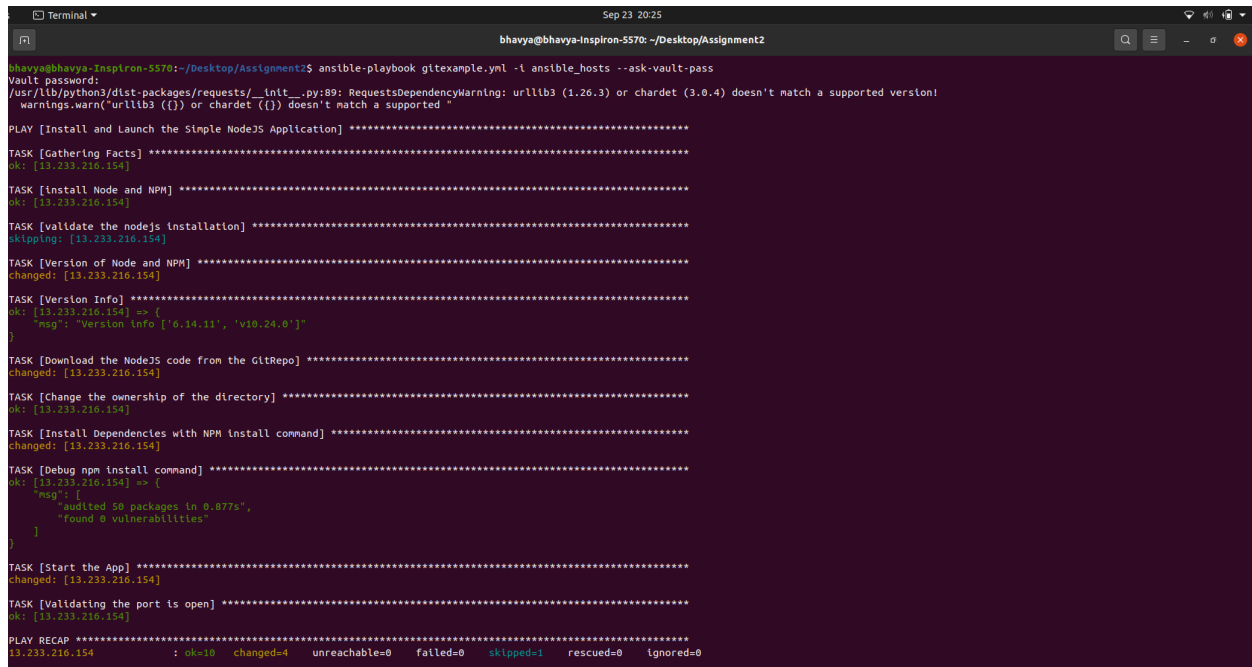
First we check if the hostgroup is reachable using the following command:
(here, `nodeserver` is the name of our hostgroup)

A terminal window titled 'bhavya@bhavya-Inspiron-5570: ~/Desktop/Assignment2' with a timestamp 'Sep 23 15:03'. The terminal shows the command 'ansible nodeserver -m ping -i ansible_hosts' being executed. The output includes a warning about urllib3 and chardet versions, followed by a successful ping result for 3.108.200.88. The JSON output shows 'ansible_facts' with 'discovered_interpreter_python' and 'ping' status 'pong'. The prompt then changes to 'bhavya@bhavya-Inspiron-5570:~/Desktop/Assignment2\$' as the user enters 'ansible-playbook gitexampl'.

```
Sep 23 15:03
bhavya@bhavya-Inspiron-5570: ~/Desktop/Assignment2
bhavya@bhavya-Inspiron-5570:~/Desktop/Assignment2$ ansible nodeserver -m ping -i ansible_hosts
/usr/lib/python3/dist-packages/requests/__init__.py:89: RequestsDependencyWarning: urllib3 (1.26.3) or chardet (3.0.4) doesn't match a supported version!
  warnings.warn("urllib3 ({}), or chardet ({}), doesn't match a supported "
3.108.200.88 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/libexec/platform-python"
  },
  "changed": false,
  "ping": "pong"
}
bhavya@bhavya-Inspiron-5570:~/Desktop/Assignment2$ ansible-playbook gitexampl
```

`Ansible_hosts` is an inventory file that contains the name of the host group, public IP address of the host server etc.

Now we launch the playbook:

A terminal window titled "Terminal" with a subtitle "bhavya@bhavya-Inspiron-5570: ~/Desktop/Assignment2". The terminal shows the execution of an Ansible playbook named "gitexample.yml". The command is "ansible-playbook gitexample.yml -i ansible_hosts --ask-vault-pass". The output shows various tasks being executed, including gathering facts, installing Node and NPM, validating NodeJS installation, downloading NodeJS code from GitHub, changing directory ownership, installing dependencies with NPM, and starting the app. The final output shows a recap of the play: "PLAY RECAP" with statistics: "13.233.216.154 : ok=10 changed=4 unreachable=0 failed=0 skipped=1 rescued=0 ignored=0".

```
Terminal
Sep 23 20:25
bhavya@bhavya-Inspiron-5570: ~/Desktop/Assignment2

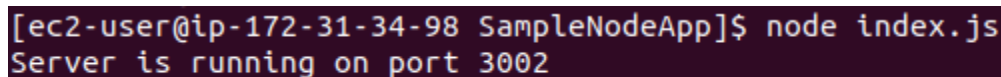
bhavya@bhavya-Inspiron-5570:~/Desktop/Assignment2$ ansible-playbook gitexample.yml -i ansible_hosts --ask-vault-pass
Vault passwords:
/usr/lib/python3/dist-packages/requests/_init_.py:89: RequestsDependencyWarning: urllib3 (1.26.3) or chardet (3.0.4) doesn't match a supported version!
warnings.warn("urllib3 ({}), or chardet ({}), doesn't match a supported version!".format(urllib3.__version__, chardet.__version__), RequestsDependencyWarning)

PLAY [Install and Launch the Simple NodeJS Application] *****
TASK [Gathering Facts] *****
ok: [13.233.216.154]
TASK [Install Node and NPM] *****
ok: [13.233.216.154]
TASK [validate the nodejs installation] *****
skipping: [13.233.216.154]
TASK [Version of Node and NPM] *****
changed: [13.233.216.154]
TASK [Version Info] *****
ok: [13.233.216.154] => {
  "msg": "Version Info ['6.14.11', 'v10.24.0']"
}
TASK [Download the NodeJS code from the GitRepo] *****
changed: [13.233.216.154]
TASK [Change the ownership of the directory] *****
ok: [13.233.216.154]
TASK [Install Dependencies with NPM (install command)] *****
changed: [13.233.216.154]
TASK [Debug npm install command] *****
ok: [13.233.216.154] => {
  "msg": [
    "audited 50 packages in 0.877s",
    "found 0 vulnerabilities"
  ]
}
TASK [Start the App] *****
changed: [13.233.216.154]
TASK [Validating the port is open] *****
ok: [13.233.216.154]

PLAY RECAP *****
13.233.216.154 : ok=10 changed=4 unreachable=0 failed=0 skipped=1 rescued=0 ignored=0
```

The playbook ran successfully.

This means our github repository has been cloned to our server in the /apps/SampleNodeApp directory.

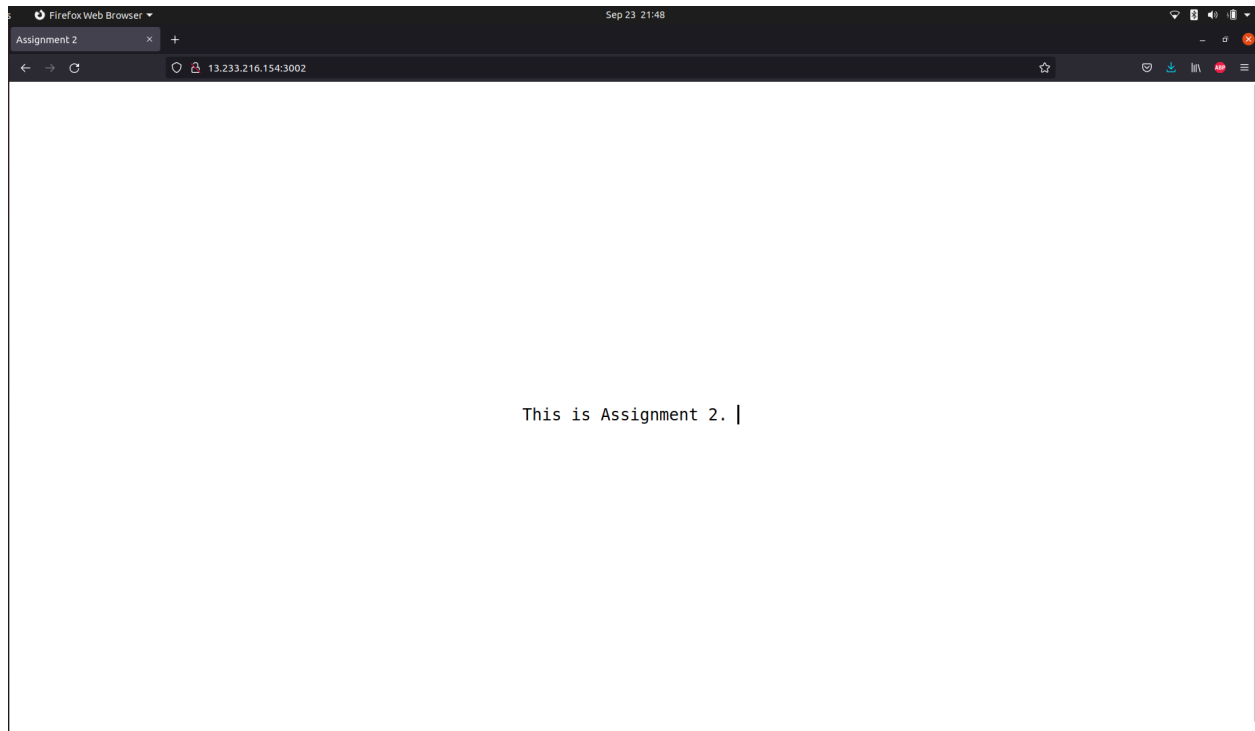
A terminal window showing the command "node index.js" being executed in the directory "/apps/SampleNodeApp". The output is "Server is running on port 3002".

```
[ec2-user@ip-172-31-34-98 SampleNodeApp]$ node index.js
Server is running on port 3002
```

The server is running successfully!

Step 5: Validate the Deployment

The remote server here is 13.233.216.154, now we can access the URL via <http://13.233.216.154:3002>.



Our Node Website has been deployed successfully!

Assignment-3 Add users to EC2 instance with SSH access using ansible

The task consists of 2 steps:

- 1.Create a new user account on all these EC2 instances for the user.
- 2.Copy the user's SSH public key into the newly created account for them to login without a password

1. Configuring the host file with multiple hosts

Creating a host group named `hosts_to_add_key` with host ip address,username.

[`hosts_to_add_key`]

172.99.1.82 ansible_user=ec2-user ansible_port=22

172.99.1.56 ansible_user=ec2-user ansible_port=22

[`hosts_to_add_key:vars`]

ansible_ssh_common_args="-o StrictHostKeyChecking=no"

2. Checking if the host are available

```
Select harshitha@DESKTOP-D1349KB: ~/Team8_devops
harshitha@DESKTOP-D1349KB:~/Team8_devops$ ansible hosts_to_add_key -m ping
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'
[WARNING]: Could not match supplied host pattern, ignoring: hosts_to_add_key
harshitha@DESKTOP-D1349KB:~/Team8_devops$ ansible hosts_to_add_key -m ping -i ansible_hosts
18.221.249.41 | UNREACHABLE! => {
  "changed": false,
  "msg": "Failed to connect to the host via ssh: load pubkey \"~/home/harshitha/.ssh/id_rsa\": invalid format\r\nharshitha@18.221.249.41: Permission denied (publickey,gssapi-keyex,gssapi-with-mic).",
  "unreachable": true
}
3.137.204.136 | UNREACHABLE! => {
  "changed": false,
  "msg": "Failed to connect to the host via ssh: load pubkey \"~/home/harshitha/.ssh/id_rsa\": invalid format\r\nWarning: Permanently added '3.137.204.136' (ECDSA) to the list of known hosts.\r\nharshitha@3.137.204.136: Permission denied (publickey,gssapi-keyex,gssapi-with-mic).",
  "unreachable": true
}
harshitha@DESKTOP-D1349KB:~/Team8_devops$ ls
add-key.yml  ansible_hosts  keypair  keypair.pem
harshitha@DESKTOP-D1349KB:~/Team8_devops$ ansible hosts_to_add_key -m ping -i ansible_hosts -i keypair.pem
[WARNING]: * Failed to parse ~/home/harshitha/Team8_devops/keypair.pem with ini plugin: /home/harshitha/Team8_devops/keypair.pem:1: Expected key=value host variable assignment, got: RSA
[WARNING]: Unable to parse /home/harshitha/Team8_devops/keypair.pem as an inventory source
3.137.204.136 | UNREACHABLE! => {
  "changed": false,
  "msg": "Failed to connect to the host via ssh: load pubkey \"~/home/harshitha/.ssh/id_rsa\": invalid format\r\nharshitha@3.137.204.136: Permission denied (publickey,gssapi-keyex,gssapi-with-mic).",
  "unreachable": true
}
18.221.249.41 | UNREACHABLE! => {
  "changed": false,
  "msg": "Failed to connect to the host via ssh: load pubkey \"~/home/harshitha/.ssh/id_rsa\": invalid format\r\nharshitha@18.221.249.41: Permission denied (publickey,gssapi-keyex,gssapi-with-mic).",
  "unreachable": true
}
harshitha@DESKTOP-D1349KB:~/Team8_devops$ ansible hosts_to_add_key -m ping -i ansible_hosts --user ec2-user --key-file keypair.pem
[WARNING]: Platform linux on host 18.221.249.41 is using the discovered Python interpreter at /usr/bin/python, but future installation of another Python interpreter could change this. See
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information.
18.221.249.41 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python"
  },
  "changed": false,
  "ping": "pong"
}
[WARNING]: Platform linux on host 3.137.204.136 is using the discovered Python interpreter at /usr/bin/python, but future installation of another Python interpreter could change this. See
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information.
3.137.204.136 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python"
  },
  "changed": false,
  "ping": "pong"
}
harshitha@DESKTOP-D1349KB:~/Team8_devops$
```

3.Playbook to add users in Ec2 instance and copy SSH key

Task-1 to create a group of users

Task-2 to create a user and map to a group created in the previous step

Create a keypair for each user

```

harshitha@DESKTOP-D1349KB: ~/.ssh
Enter file in which to save the key (/home/harshitha/.ssh/id_rsa): /home/harshitha/Team8_devops
/home/harshitha/Team8_devops already exists.
Overwrite (y/n)? ^C
harshitha@DESKTOP-D1349KB:~$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/harshitha/.ssh/id_rsa): /home/harshitha/.ssh/harshitha_rsa
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/harshitha/.ssh/harshitha_rsa
Your public key has been saved in /home/harshitha/.ssh/harshitha_rsa.pub
The key fingerprint is:
SHA256:YhuCWNcYUL9Pj/k2zi7xQA04Vdn0rlnWorLbNyxJuM harshitha@DESKTOP-D1349KB
The key's randomart image is:
+---[RSA 3072]-----+
|      .      |
|      o      |
|    o.o+o    |
|    +o o.+   |
|    .+.S+    |
|      o++    |
|    .o+o     |
|    o*Bo+    |
|    oE==     |
+---[SHA256]-----+
harshitha@DESKTOP-D1349KB:~$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/harshitha/.ssh/id_rsa): /home/harshitha/.ssh/akshaya_rsa
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/harshitha/.ssh/akshaya_rsa
Your public key has been saved in /home/harshitha/.ssh/akshaya_rsa.pub
The key fingerprint is:
SHA256:Fk6ceCdGh3iSh2YDfNwJ2MwuL5eyvmAL7ipG6vNc harshitha@DESKTOP-D1349KB
The key's randomart image is:
+---[RSA 3072]-----+
|      .      |
|      o      |
|    o.o+o    |
|    +o o.+   |
|    .+.S+    |
|      o++    |
|    .o+o     |
|    o*Bo+    |
|    oE==     |
+---[SHA256]-----+
harshitha@DESKTOP-D1349KB:~$ cd ~/.ssh
harshitha@DESKTOP-D1349KB:~/.ssh$ ls
akshaya_rsa  akshaya_rsa.pub  harshitha_rsa  harshitha_rsa.pub  id_rsa  id_rsa.pub  known_hosts
harshitha@DESKTOP-D1349KB:~/.ssh$

```

Task-3 to copy the user's SSH key to their newly created user IDs on the EC2 instance for them to able to log in.

4. Running the playbook

```

harshitha@DESKTOP-D1349KB: ~/Team8_devops
ok: [18.216.105.64] => (item={'name': 'akshaya', 'key': '~/.ssh/akshaya_rsa.pub', 'userstate': 'present'})
changed: [18.217.6.207] => (item={'name': 'akshaya', 'key': '~/.ssh/akshaya_rsa.pub', 'userstate': 'present'})

PLAY RECAP *****
18.216.105.64      : ok=4  changed=0  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0
18.217.6.207      : ok=4  changed=3  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0

harshitha@DESKTOP-D1349KB:~/Team8_devops$ ansible-playbook add-key-private.yml -i ansible_hosts_private --key-file=keypair.pem
[WARNING]: * Failed to parse /home/harshitha/Team8_devops/ansible_hosts_private with yaml plugin: Syntax Error while loading YAML. did not find expected <document start> The error appears to be in
/home/harshitha/Team8_devops/ansible_hosts_private: line 2, column 1, but may be elsewhere in the file depending on the exact syntax problem. The offending line appears to be:
18.217.6.207 ansible_user=ec2-user ansible_port=22 ^ here
[WARNING]: * Failed to parse /home/harshitha/Team8_devops/ansible_hosts_private with ini plugin: /home/harshitha/Team8_devops/ansible_hosts_private:6: Section [hosts_to_add_key:vars] not valid for undefined
group: hosts_to_add_key
[WARNING]: Unable to parse /home/harshitha/Team8_devops/ansible_hosts_private as an inventory source
[WARNING]: No inventory was parsed, only implicit localhost is available
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [Playbook to Create User and Add Key to EC2 Instance] *****

TASK [Gathering Facts] *****
[WARNING]: Platform linux on host 18.216.105.64 is using the discovered Python interpreter at /usr/bin/python, but future installation of another Python interpreter could change this. See
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information.
ok: [18.216.105.64]
[WARNING]: Platform linux on host 18.217.6.207 is using the discovered Python interpreter at /usr/bin/python, but future installation of another Python interpreter could change this. See
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information.
ok: [18.217.6.207]

TASK [Create Groups] *****
ok: [18.216.105.64] => (item=adminteam)
ok: [18.217.6.207] => (item=adminteam)
ok: [18.216.105.64] => (item=devopsteam)
ok: [18.217.6.207] => (item=devopsteam)

TASK [Create a user] *****
ok: [18.216.105.64] => (item={'name': 'harshitha', 'group': 'adminteam'})
ok: [18.217.6.207] => (item={'name': 'harshitha', 'group': 'adminteam'})
ok: [18.216.105.64] => (item={'name': 'akshaya', 'group': 'devopsteam'})
ok: [18.217.6.207] => (item={'name': 'akshaya', 'group': 'devopsteam'})

TASK [Copy the authorized key file from] *****
ok: [18.216.105.64] => (item={'name': 'harshitha', 'key': '~/.ssh/harshitha_rsa.pub', 'userstate': 'present'})
ok: [18.217.6.207] => (item={'name': 'harshitha', 'key': '~/.ssh/harshitha_rsa.pub', 'userstate': 'present'})
ok: [18.216.105.64] => (item={'name': 'akshaya', 'key': '~/.ssh/akshaya_rsa.pub', 'userstate': 'present'})
ok: [18.217.6.207] => (item={'name': 'akshaya', 'key': '~/.ssh/akshaya_rsa.pub', 'userstate': 'present'})

PLAY RECAP *****
18.216.105.64      : ok=4  changed=0  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0
18.217.6.207      : ok=4  changed=0  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0

harshitha@DESKTOP-D1349KB:~/Team8_devops$

```

5. Playbook has been executed successfully and the user can log in/SSH now with his private key. I am executing id command with SSH connection and using my private key (akshaya_rsa) file to login without password.

```
harshitha@DESKTOP-D1349KB: ~/Team8_devops
18.216.105.64 : ok=4 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
18.217.6.207 : ok=4 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

harshitha@DESKTOP-D1349KB:~/Team8_devops$ ssh -i ~/.ssh/harshitha_rsa harshitha@18.217.6.207 "id harshitha"
uid=1001(harshitha) gid=1001(adminteam) groups=1001(adminteam)
harshitha@DESKTOP-D1349KB:~/Team8_devops$ ansible-playbook add-key-private.yml -i ansible_hosts_private --key-file=keypair.pem -e "random=~/.ssh/akshaya_rsa"
[WARNING]: * Failed to parse /home/harshitha/Team8_devops/ansible_hosts_private with yaml plugin: Syntax Error while loading YAML. did not find expected <document start> The error appears to be in
/home/harshitha/Team8_devops/ansible_hosts_private: line 2, column 1, but may be elsewhere in the file depending on the exact syntax problem. The offending line appears to be: [hosts_to_add_key_private]
18.217.6.207 ansible_user=ec2-user ansible_port=22 ^ here
[WARNING]: * Failed to parse /home/harshitha/Team8_devops/ansible_hosts_private with ini plugin: /home/harshitha/Team8_devops/ansible_hosts_private:6: Section [hosts_to_add_key:vars] not valid for undefined
group: hosts_to_add_key
[WARNING]: Unable to parse /home/harshitha/Team8_devops/ansible_hosts_private as an inventory source
[WARNING]: No inventory was parsed, only implicit localhost is available
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [Playbook to Create User and Add Key to EC2 Instance] *****

TASK [Gathering Facts] *****
[WARNING]: Platform linux on host 18.216.105.64 is using the discovered Python interpreter at /usr/bin/python, but future installation of another Python interpreter could change this. See
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information.
ok: [18.216.105.64]
[WARNING]: Platform linux on host 18.217.6.207 is using the discovered Python interpreter at /usr/bin/python, but future installation of another Python interpreter could change this. See
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information.
ok: [18.217.6.207]

TASK [Create Groups] *****
ok: [18.216.105.64] => (item=adminteam)
ok: [18.217.6.207] => (item=adminteam)
ok: [18.217.6.207] => (item=devopsteam)
ok: [18.216.105.64] => (item=devopsteam)

TASK [Create a user] *****
ok: [18.216.105.64] => (item={'name': 'harshitha', 'group': 'adminteam'})
ok: [18.217.6.207] => (item={'name': 'harshitha', 'group': 'adminteam'})
ok: [18.216.105.64] => (item={'name': 'akshaya', 'group': 'devopsteam'})
ok: [18.217.6.207] => (item={'name': 'akshaya', 'group': 'devopsteam'})

TASK [Copy the authorized key file from] *****
ok: [18.216.105.64] => (item={'name': 'harshitha', 'key': '~/.ssh/harshitha_rsa.pub', 'userstate': 'present'})
ok: [18.217.6.207] => (item={'name': 'harshitha', 'key': '~/.ssh/harshitha_rsa.pub', 'userstate': 'present'})
ok: [18.216.105.64] => (item={'name': 'akshaya', 'key': '~/.ssh/akshaya_rsa.pub', 'userstate': 'present'})
ok: [18.217.6.207] => (item={'name': 'akshaya', 'key': '~/.ssh/akshaya_rsa.pub', 'userstate': 'present'})

PLAY RECAP *****
18.216.105.64 : ok=4 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
18.217.6.207 : ok=4 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

harshitha@DESKTOP-D1349KB:~/Team8_devops$ ssh -i ~/.ssh/akshaya_rsa akshaya@18.217.6.207 "id akshaya"
uid=1002(akshaya) gid=1002(devopsteam) groups=1002(devopsteam)
harshitha@DESKTOP-D1349KB:~/Team8_devops$
```

Account has been created successfully.