DevOps ASSIGNMENT

TEAM 10

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

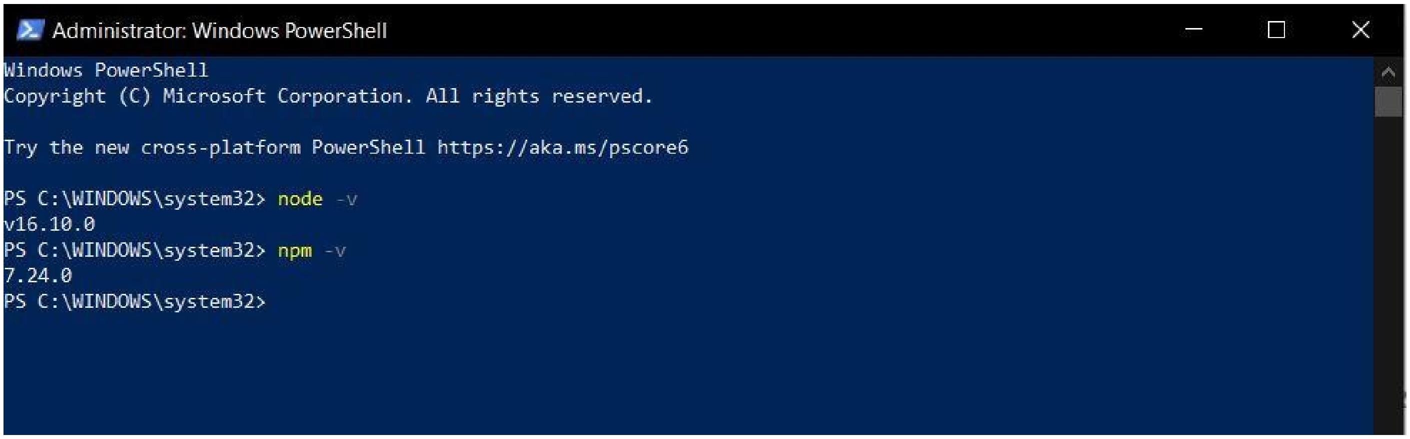
GK Bharath Bhushan-18BCS026
K V S Chaitanya-18bcs045
Neha T-18bcs060
Rahul S-18BCS075
Varun awati -18BCS108
Sushanth B patil-18BCS102
Aqtar Parveez-18BCS010
Trishul K S-18BCS104

TO:
Dr.Uma S

Docker file link: https://hub.docker.com/r/18bcs026/nodejs-starter

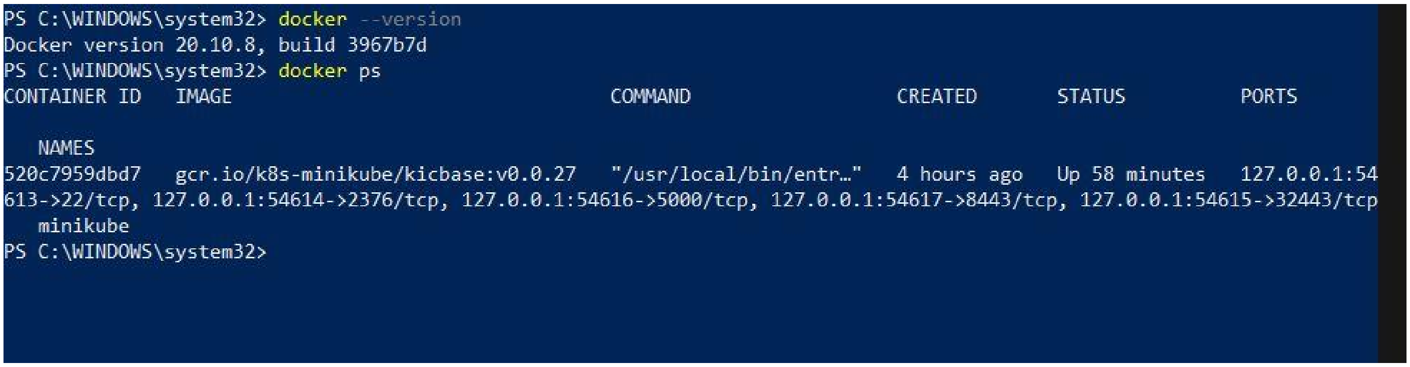
PROCEDURE

Install Node.js and npm



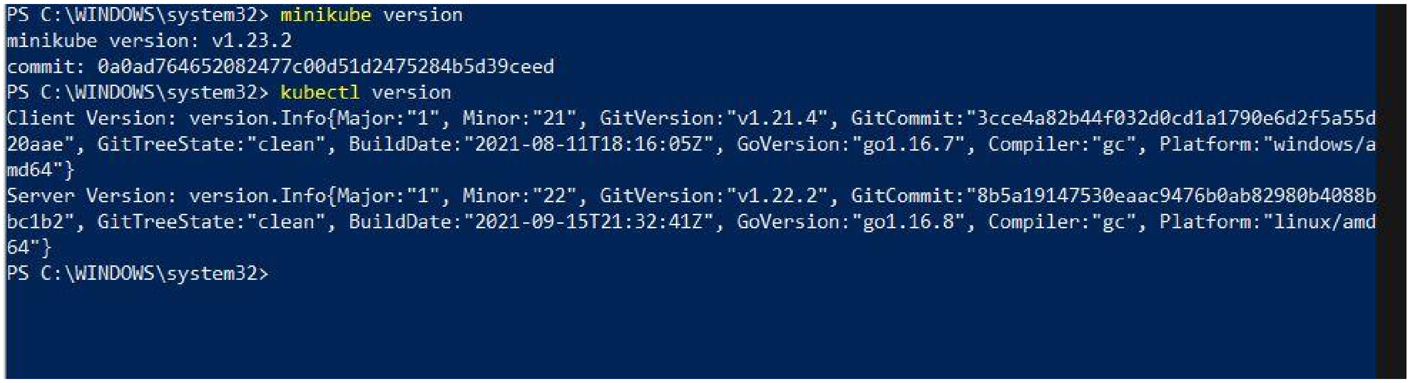
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

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

PS C:\WINDOWS\system32> node -v
v16.10.0
PS C:\WINDOWS\system32> npm -v
7.24.0
PS C:\WINDOWS\system32>
```

Install Docker



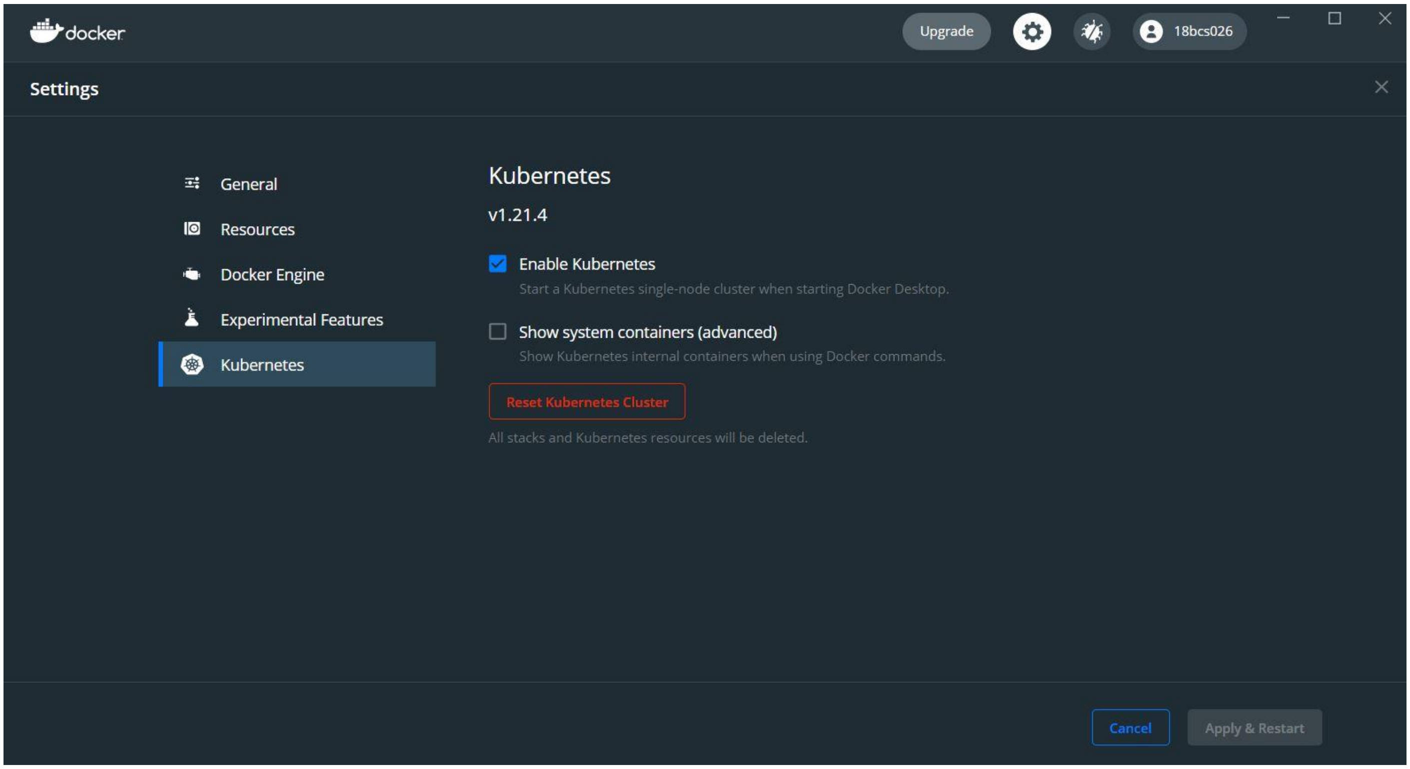
```
PS C:\WINDOWS\system32> docker --version
Docker version 20.10.8, build 3967b7d
PS C:\WINDOWS\system32> docker ps
CONTAINER ID   IMAGE                                 COMMAND                  CREATED       STATUS          PORTS

   NAMES
520c7959dbd7   gcr.io/k8s-minikube/kicbase:v0.0.27   "/usr/local/bin/entr…"   4 hours ago   Up 58 minutes   127.0.0.1:54
613->22/tcp, 127.0.0.1:54614->2376/tcp, 127.0.0.1:54616->5000/tcp, 127.0.0.1:54617->8443/tcp, 127.0.0.1:54615->32443/tcp
   minikube
PS C:\WINDOWS\system32>
```

Minikube and Kubectl installation



```
PS C:\WINDOWS\system32> minikube version
minikube version: v1.23.2
commit: 0a0ad764652082477c00d51d2475284b5d39ceed
PS C:\WINDOWS\system32> kubectl version
Client Version: version.Info{Major:"1", Minor:"21", GitVersion:"v1.21.4", GitCommit:"3cce4a82b44f032d0cd1a1790e6d2f5a55d
20aae", GitTreeState:"clean", BuildDate:"2021-08-11T18:16:05Z", GoVersion:"go1.16.7", Compiler:"gc", Platform:"windows/a
md64"}
Server Version: version.Info{Major:"1", Minor:"22", GitVersion:"v1.22.2", GitCommit:"8b5a19147530eaac9476b0ab82980b4088b
bc1b2", GitTreeState:"clean", BuildDate:"2021-09-15T21:32:41Z", GoVersion:"go1.16.8", Compiler:"gc", Platform:"linux/amd
64"}
PS C:\WINDOWS\system32>
```

Enabled Kubernetes service with Docker



Kubernetes

v1.21.4

☑ Enable Kubernetes

Start a Kubernetes single-node cluster when starting Docker Desktop.

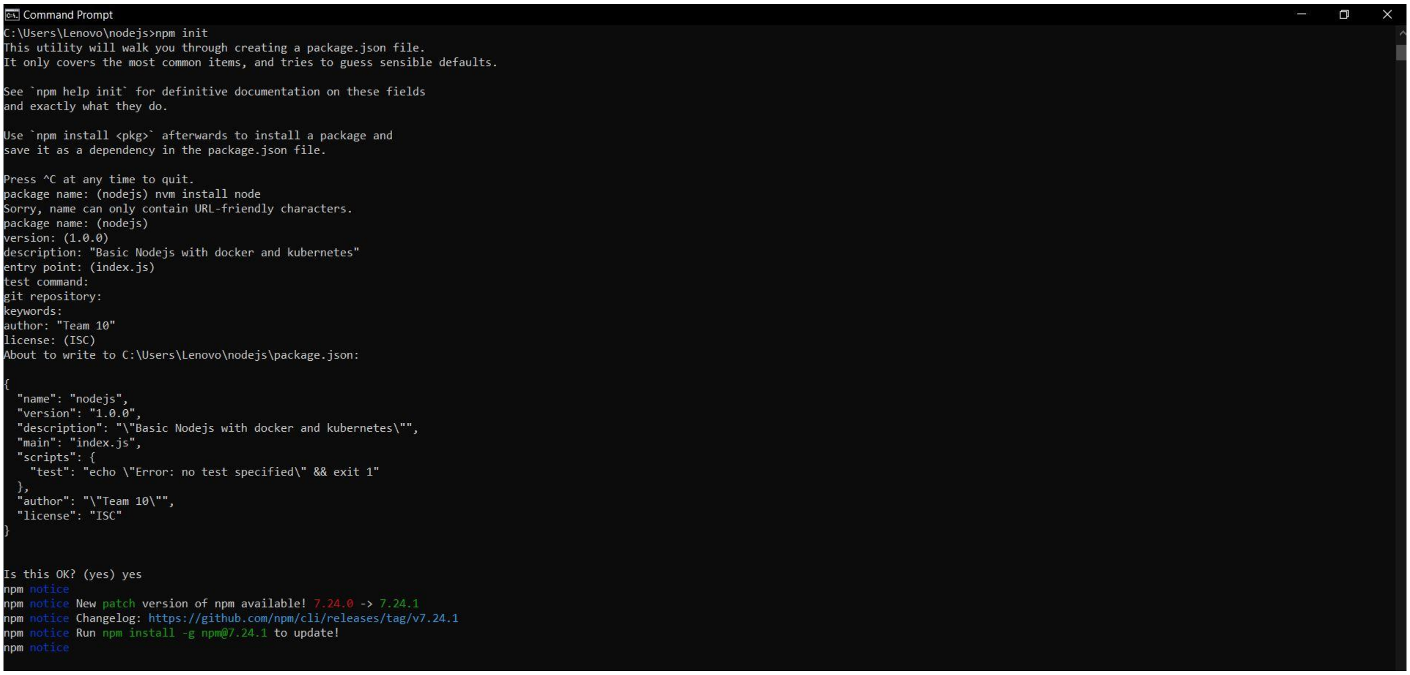
□ Show system containers (advanced)

Show Kubernetes internal containers when using Docker commands.

Reset Kubernetes Cluster

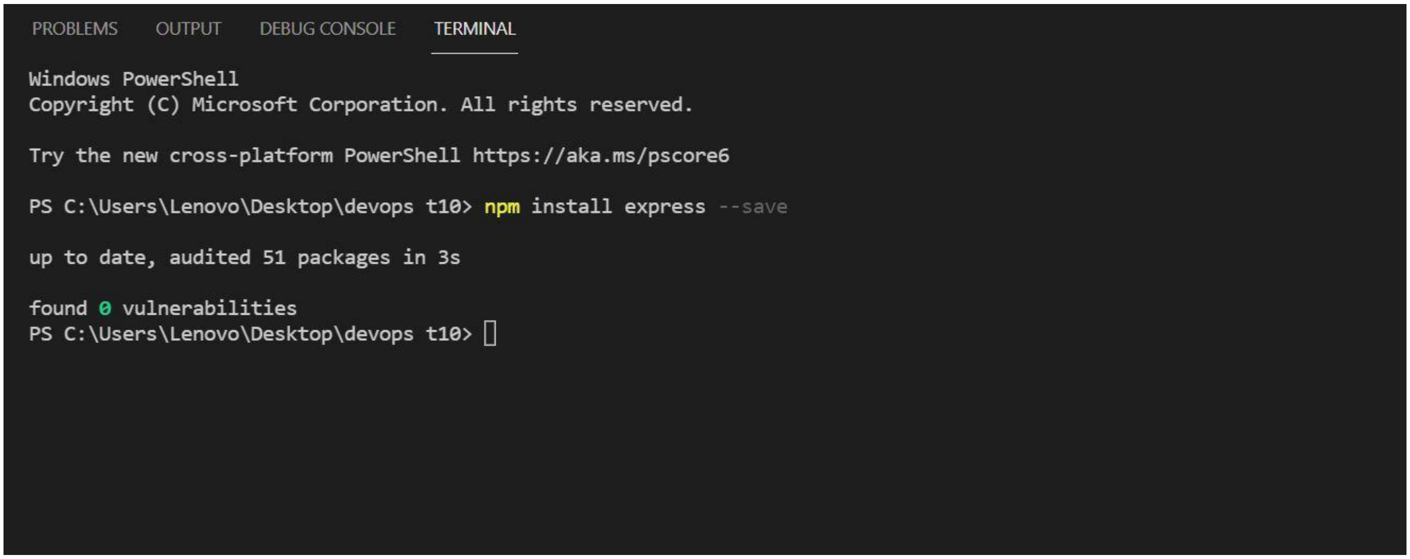
All stacks and Kubernetes resources will be deleted.

Step 1: Make A Separate Directory And Initialize The Node Application



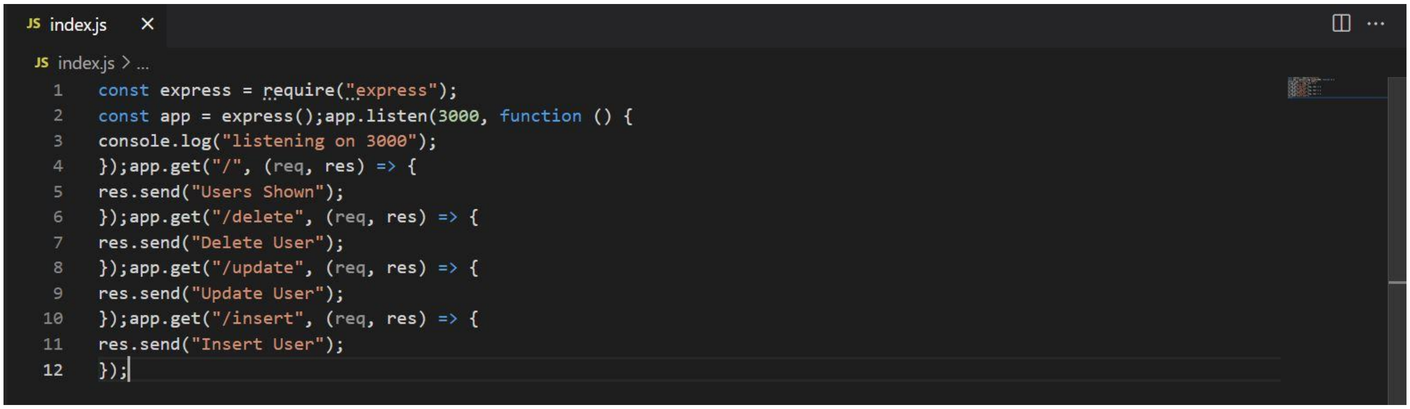
```
C:\Users\Lenovo\nodejs>npm init
This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sensible defaults.

See `npm help init` for definitive documentation on these fields
and exactly what they do.

Use `npm install <pkg>` afterwards to install a package and
save it as a dependency in the package.json file.

Press ^C at any time to quit.
package name: (nodejs) nvm install node
Sorry, name can only contain URL-friendly characters.
package name: (nodejs)
version: (1.0.0)
description: "Basic Nodejs with docker and kubernetes"
entry point: (index.js)
test command:
git repository:
keywords:
author: "Team 10"
license: (ISC)
About to write to C:\Users\Lenovo\nodejs\package.json:

{
  "name": "nodejs",
  "version": "1.0.0",
  "description": "\"Basic Nodejs with docker and kubernetes\"",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "\"Team 10\"",
  "license": "ISC"
}


Is this OK? (yes) yes
npm notice
npm notice New patch version of npm available! 7.24.0 -> 7.24.1
npm notice Changelog: https://github.com/npm/cli/releases/tag/v7.24.1
npm notice Run npm install -g npm@7.24.1 to update!
npm notice
```

Step 2: Installing Express



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

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

PS C:\Users\Lenovo\Desktop\devops t10> npm install express --save

up to date, audited 51 packages in 3s

found 0 vulnerabilities
PS C:\Users\Lenovo\Desktop\devops t10>
```

Step 3: Make index.js File And Write Some Code

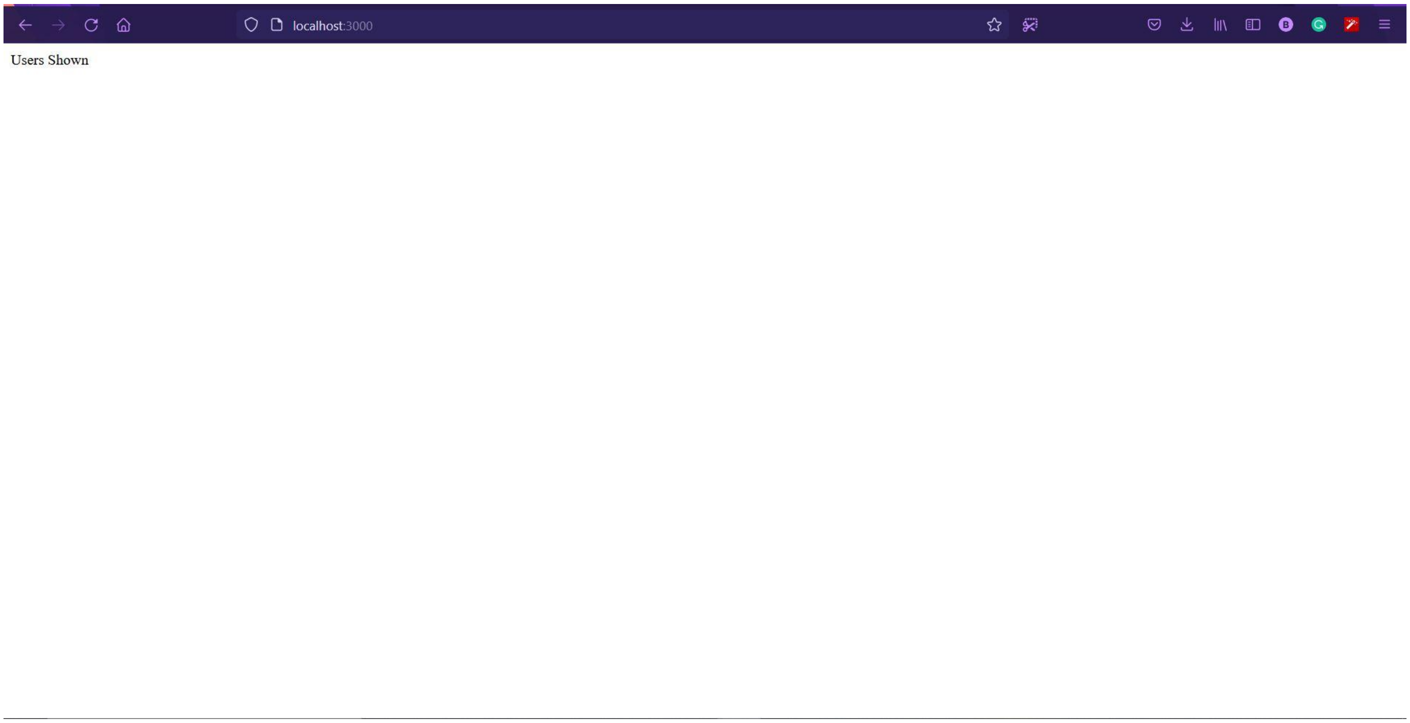


```
const express = require("express");
const app = express();app.listen(3000, function () {
console.log("listening on 3000");
});app.get("/", (req, res) => {
res.send("Users Shown");
});app.get("/delete", (req, res) => {
res.send("Delete User");
});app.get("/update", (req, res) => {
res.send("Update User");
});app.get("/insert", (req, res) => {
res.send("Insert User");
});
```

After writing the code in the index.js file run the following command in the terminal.

$ node index.js

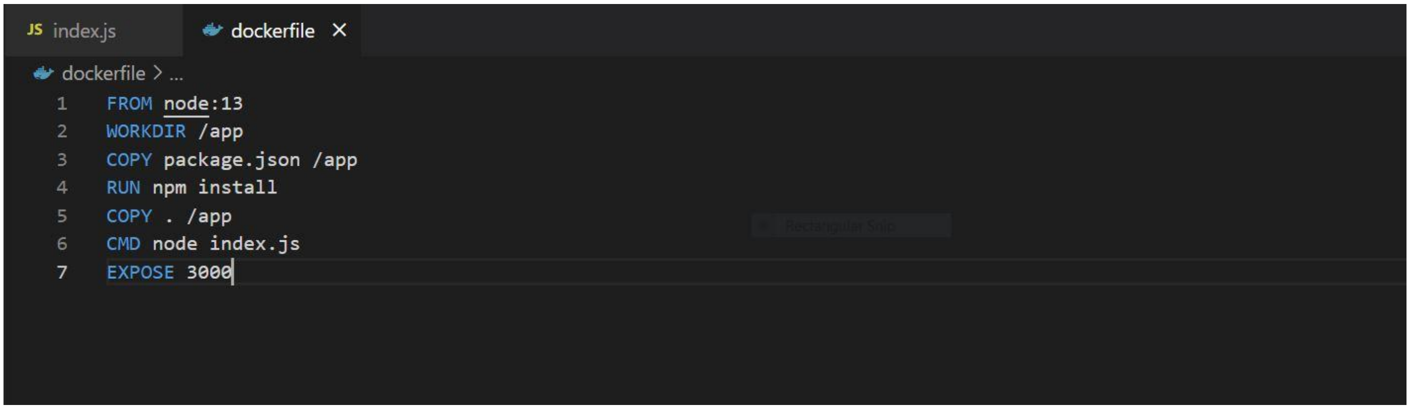
You can now check the server by using the following command and browsing localhost:3000/



Users Shown

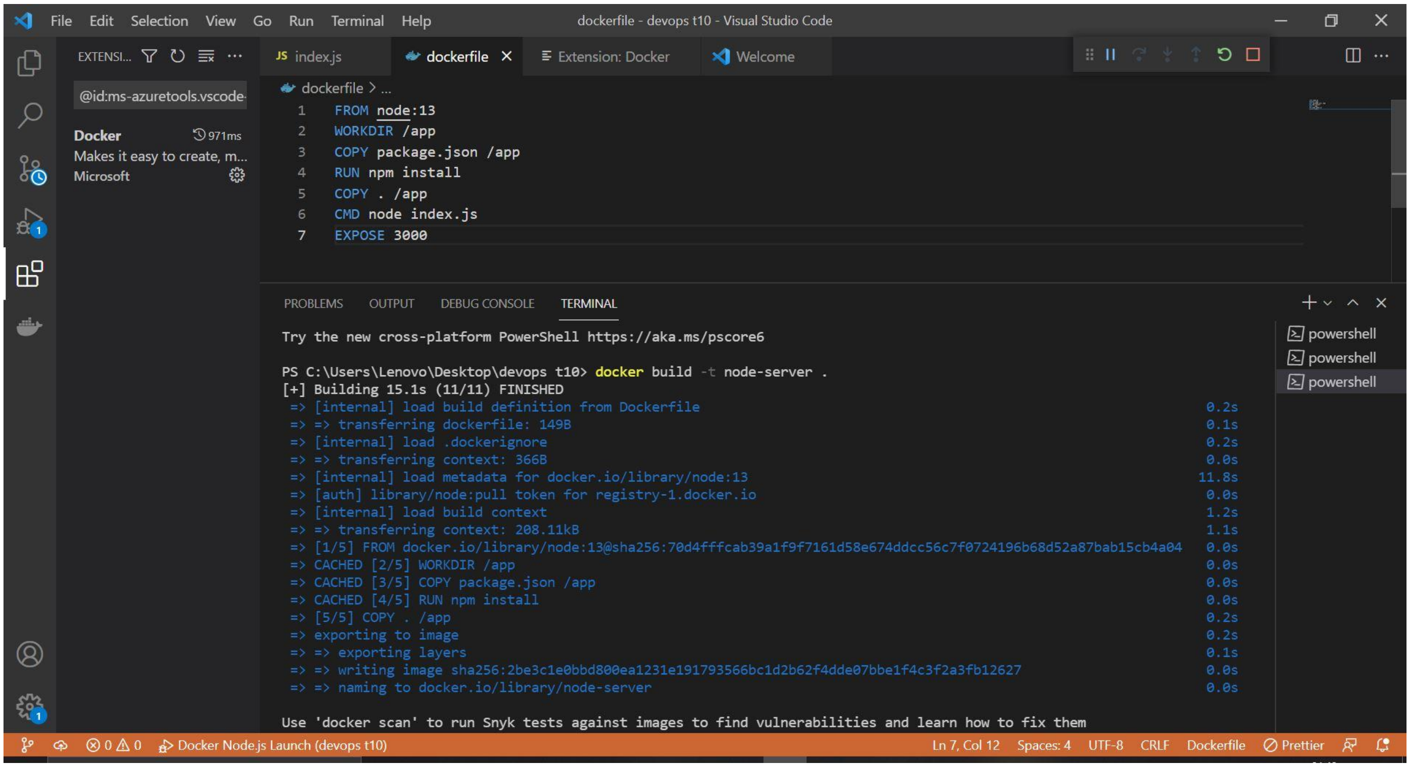
Step 4: Dockerizing The Node Server

For creating the Dockerfile run the following command on terminal: **$ code dockerfile**



```
FROM node:13
WORKDIR /app
COPY package.json /app
RUN npm install
COPY . /app
CMD node index.js
EXPOSE 3000
```

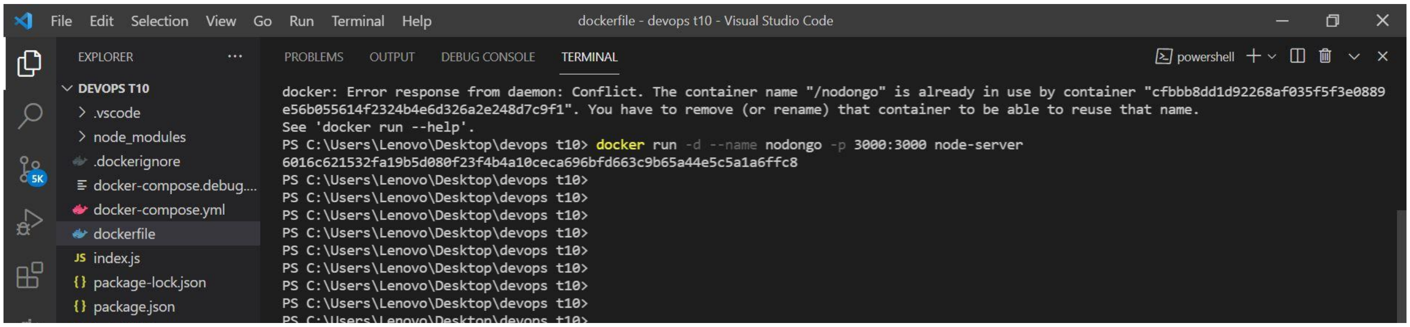
Start building our image by running the following command on terminal: **$ docker build -t node-server .**



```
FROM node:13
WORKDIR /app
COPY package.json /app
RUN npm install
COPY . /app
CMD node index.js
EXPOSE 3000
```

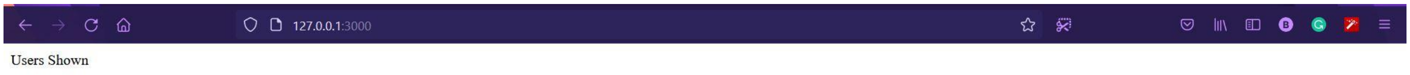
```
Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Lenovo\Desktop\devops t10> docker build -t node-server .
[+] Building 15.1s (11/11) FINISHED
 => [internal] load build definition from Dockerfile                                             0.2s
 => => transferring dockerfile: 149B                                                             0.1s
 => [internal] load .dockerignore                                                                0.2s
 => => transferring context: 366B                                                                0.0s
 => [internal] load metadata for docker.io/library/node:13                                      11.8s
 => [auth] library/node:pull token for registry-1.docker.io                                      0.0s
 => [internal] load build context                                                                1.2s
 => => transferring context: 208.11kB                                                            1.1s
 => [1/5] FROM docker.io/library/node:13@sha256:70d4fffcab39a1f9f7161d58e674ddcc56c7f0724196b68d52a87bab15cb4a04  0.0s
 => CACHED [2/5] WORKDIR /app                                                                    0.0s
 => CACHED [3/5] COPY package.json /app                                                          0.0s
 => CACHED [4/5] RUN npm install                                                                 0.0s
 => [5/5] COPY . /app                                                                            0.2s
 => exporting to image                                                                           0.2s
 => => exporting layers                                                                          0.1s
 => => writing image sha256:2be3c1e0bbd800ea1231e191793566bc1d2b62f4dde07bbe1f4c3f2a3fb12627     0.0s
 => => naming to docker.io/library/node-server                                                   0.0s

Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them
```

Step 5: Create And Run The Container



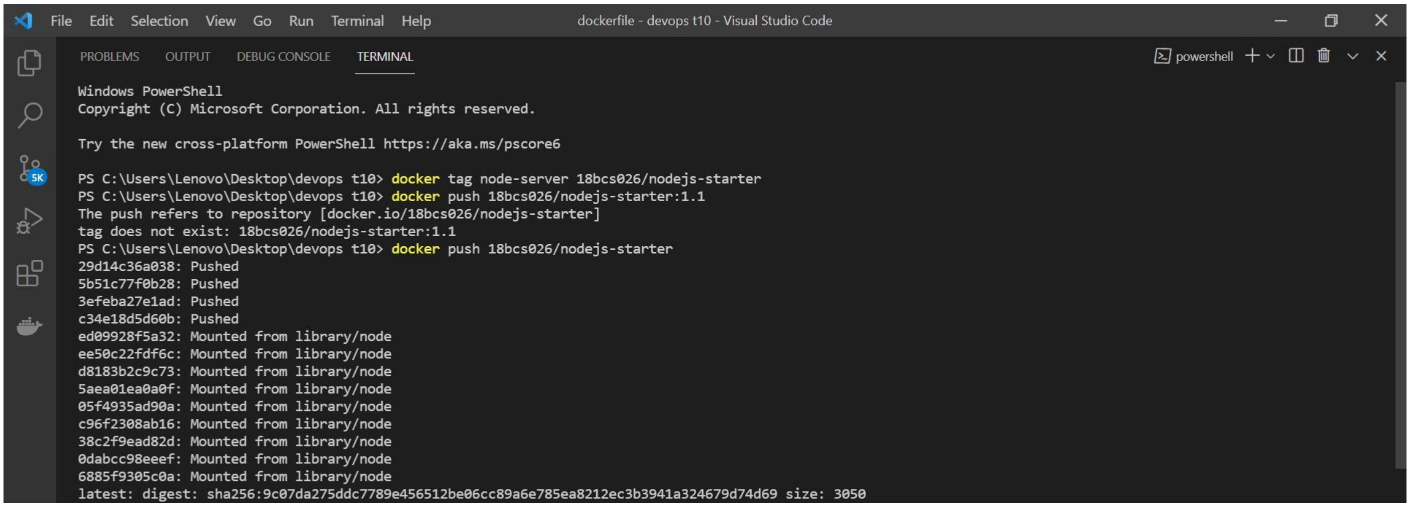
```
docker: Error response from daemon: Conflict. The container name "/nodongo" is already in use by container "cfbbb8dd1d92268af035f5f3e0889
e56b055614f2324b4e6d326a2e248d7c9f1". You have to remove (or rename) that container to be able to reuse that name.
See 'docker run --help'.
PS C:\Users\Lenovo\Desktop\devops t10> docker run -d --name nodongo -p 3000:3000 node-server
6016c621532fa19b5d080f23f4b4a10ceca696bfd663c9b65a44e5c5a1a6ffc8
PS C:\Users\Lenovo\Desktop\devops t10>
PS C:\Users\Lenovo\Desktop\devops t10>
PS C:\Users\Lenovo\Desktop\devops t10>
PS C:\Users\Lenovo\Desktop\devops t10>
PS C:\Users\Lenovo\Desktop\devops t10>
PS C:\Users\Lenovo\Desktop\devops t10>
PS C:\Users\Lenovo\Desktop\devops t10>
PS C:\Users\Lenovo\Desktop\devops t10>
PS C:\Users\Lenovo\Desktop\devops t10>
```

Go to the browser and browse the following address **127.0.0.1:3000** to test that it’s running.

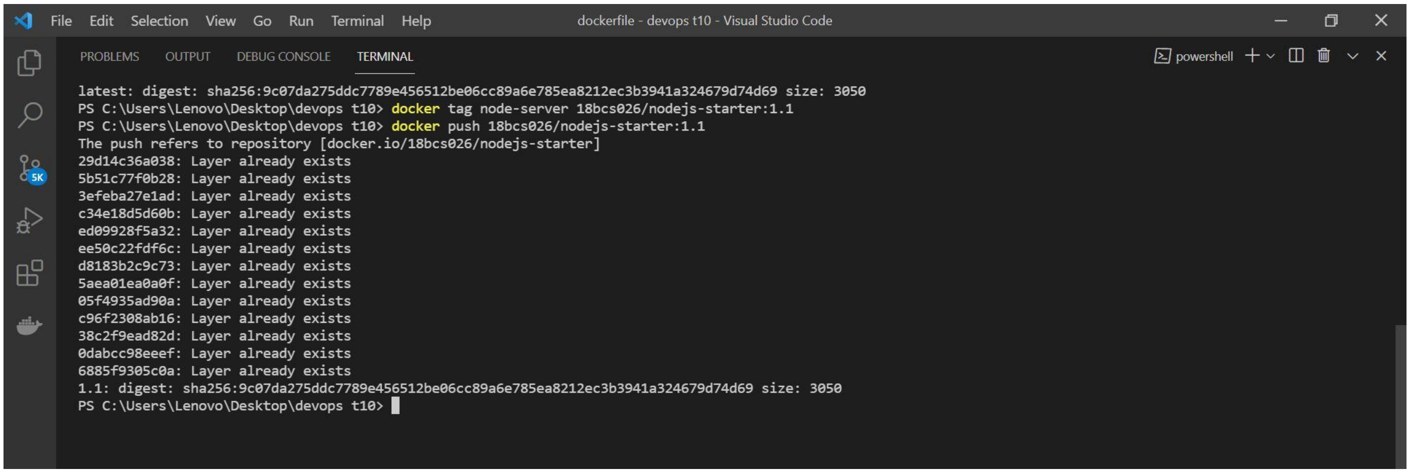


Users Shown

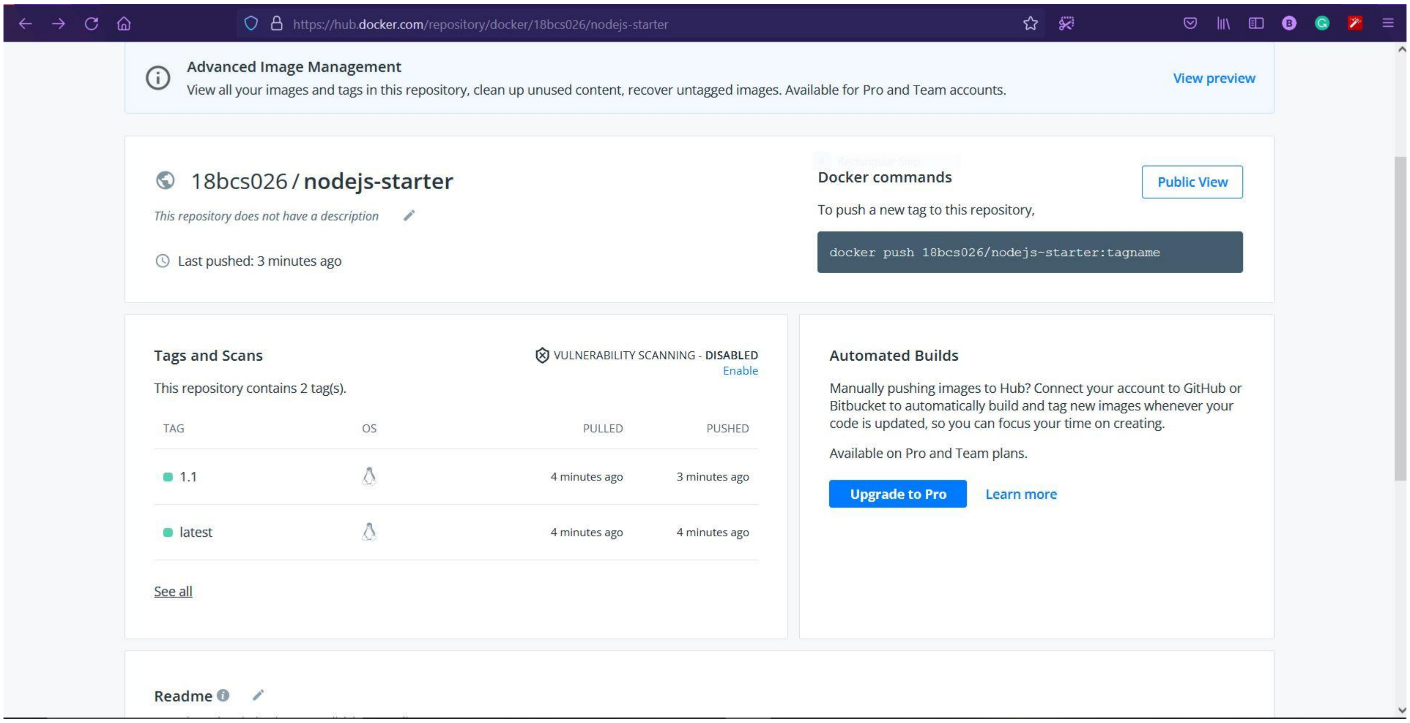
Step 6: Upload The Image To Docker Registry Docker Hub



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

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

PS C:\Users\Lenovo\Desktop\devops t10> docker tag node-server 18bcs026/nodejs-starter
PS C:\Users\Lenovo\Desktop\devops t10> docker push 18bcs026/nodejs-starter:1.1
The push refers to repository [docker.io/18bcs026/nodejs-starter]
tag does not exist: 18bcs026/nodejs-starter:1.1
PS C:\Users\Lenovo\Desktop\devops t10> docker push 18bcs026/nodejs-starter
29d14c36a038: Pushed
5b51c77f0b28: Pushed
3efeba27e1ad: Pushed
c34e18d5d60b: Pushed
ed09928f5a32: Mounted from library/node
ee50c22fdf6c: Mounted from library/node
d8183b2c9c73: Mounted from library/node
5aea01ea0a0f: Mounted from library/node
05f4935ad90a: Mounted from library/node
c96f2308ab16: Mounted from library/node
38c2f9ead82d: Mounted from library/node
0dabcc98eeef: Mounted from library/node
6885f9305c0a: Mounted from library/node
latest: digest: sha256:9c07da275ddc7789e456512be06cc89a6e785ea8212ec3b3941a324679d74d69 size: 3050
```



```
latest: digest: sha256:9c07da275ddc7789e456512be06cc89a6e785ea8212ec3b3941a324679d74d69 size: 3050
PS C:\Users\Lenovo\Desktop\devops t10> docker tag node-server 18bcs026/nodejs-starter:1.1
PS C:\Users\Lenovo\Desktop\devops t10> docker push 18bcs026/nodejs-starter:1.1
The push refers to repository [docker.io/18bcs026/nodejs-starter]
29d14c36a038: Layer already exists
5b51c77f0b28: Layer already exists
3efeba27e1ad: Layer already exists
c34e18d5d60b: Layer already exists
ed09928f5a32: Layer already exists
ee50c22fdf6c: Layer already exists
d8183b2c9c73: Layer already exists
5aea01ea0a0f: Layer already exists
05f4935ad90a: Layer already exists
c96f2308ab16: Layer already exists
38c2f9ead82d: Layer already exists
0dabcc98eeef: Layer already exists
6885f9305c0a: Layer already exists
1.1: digest: sha256:9c07da275ddc7789e456512be06cc89a6e785ea8212ec3b3941a324679d74d69 size: 3050
PS C:\Users\Lenovo\Desktop\devops t10>
```


https://hub.docker.com/repository/docker/18bcs026/nodejs-starter

Advanced Image Management

View all your images and tags in this repository, clean up unused content, recover untagged images. Available for Pro and Team accounts.

View preview

18bcs026 / **nodejs-starter**

This repository does not have a description

Last pushed: 3 minutes ago

Docker commands

Public View

To push a new tag to this repository,

```
docker push 18bcs026/nodejs-starter:tagname
```

Tags and Scans

VULNERABILITY SCANNING - **DISABLED**
Enable

This repository contains 2 tag(s).

TAG	OS	PULLED	PUSHED
🟩 1.1	🐧	4 minutes ago	3 minutes ago
🟩 latest	🐧	4 minutes ago	4 minutes ago

See all

Automated Builds

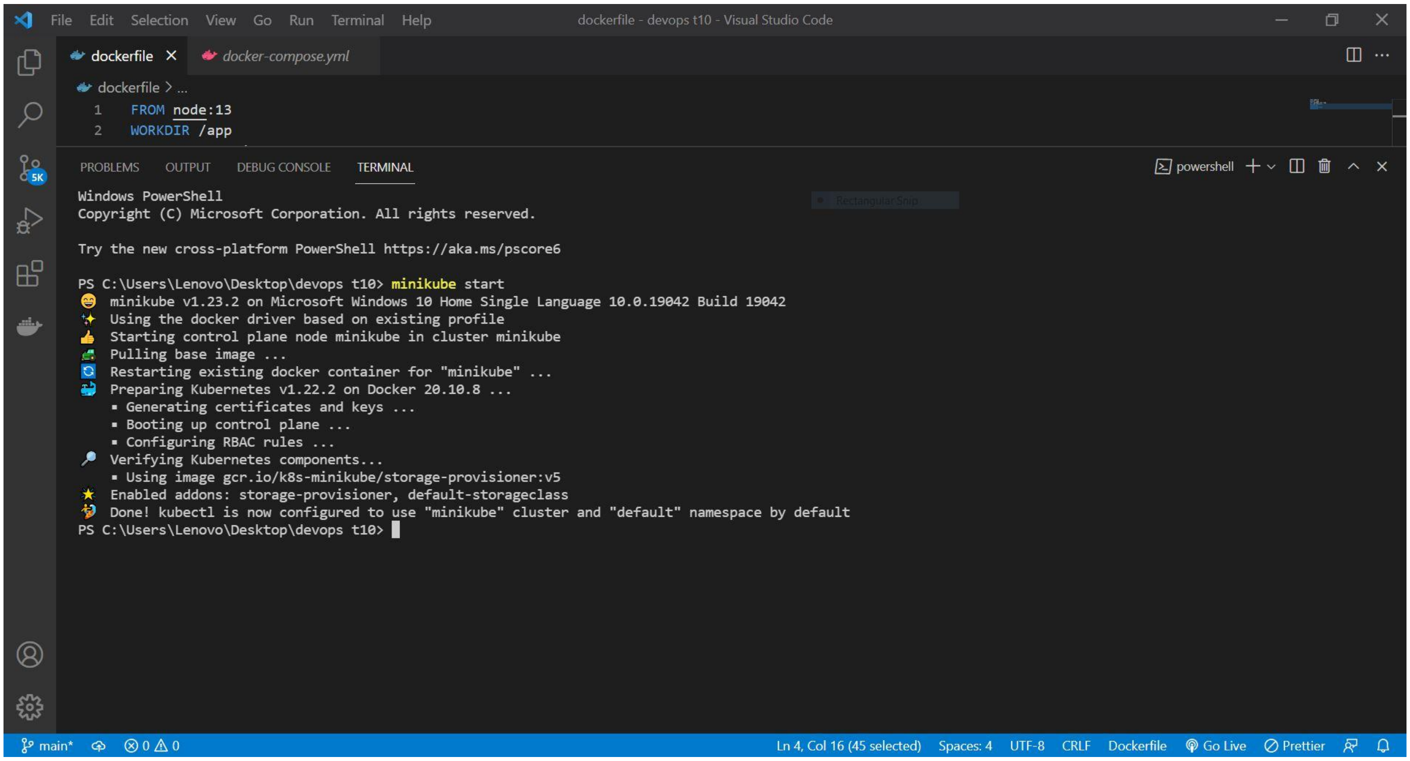
Manually pushing images to Hub? Connect your account to GitHub or Bitbucket to automatically build and tag new images whenever your code is updated, so you can focus your time on creating.

Available on Pro and Team plans.

Upgrade to Pro
Learn more

Readme

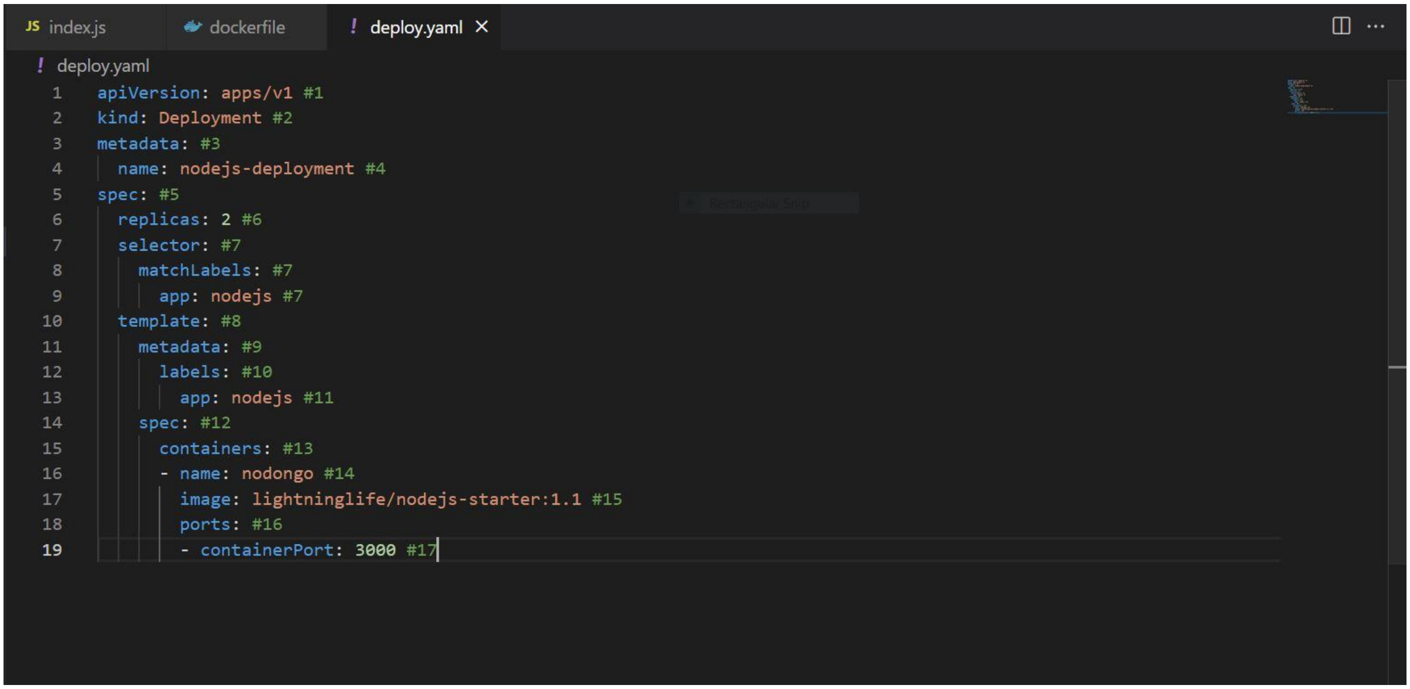
Step 7: Start The Kubernetes Cluster



```
FROM node:13
WORKDIR /app
```

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

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

PS C:\Users\Lenovo\Desktop\devops t10> 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
🚜  Pulling base image ...
🔄  Restarting existing docker container for "minikube" ...
🐳  Preparing Kubernetes v1.22.2 on Docker 20.10.8 ...
    ▪ Generating certificates and keys ...
    ▪ Booting up control plane ...
    ▪ Configuring RBAC rules ...
🔎  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\Lenovo\Desktop\devops t10> 
```

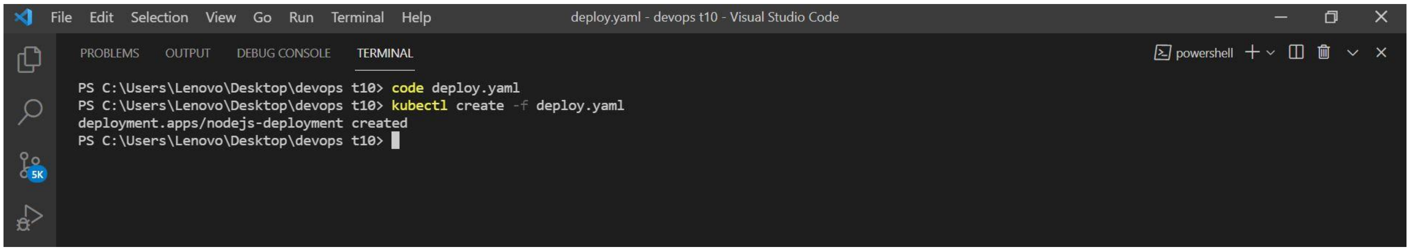
Step 8: Define YAML File To Create A Deployment In Kubernetes Cluster



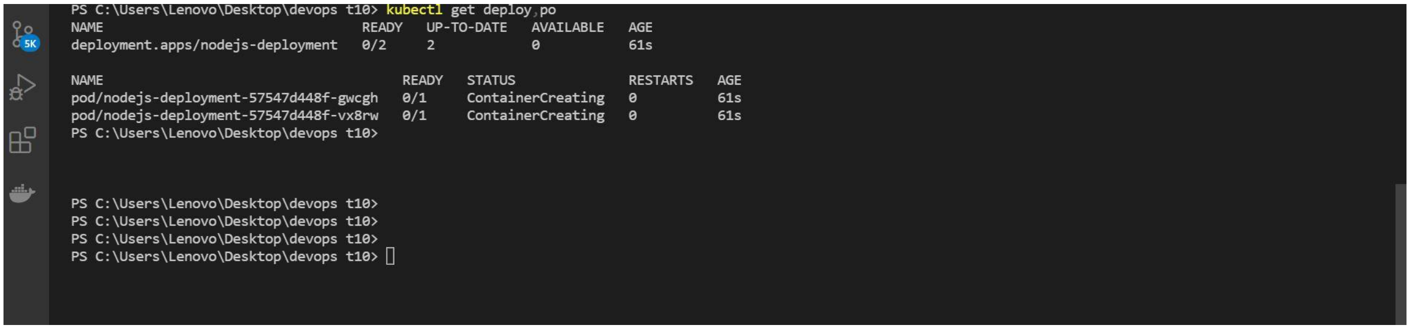
```yaml
apiVersion: apps/v1 #1
kind: Deployment #2
metadata: #3
  name: nodejs-deployment #4
spec: #5
  replicas: 2 #6
  selector: #7
    matchLabels: #7
      app: nodejs #7
  template: #8
    metadata: #9
      labels: #10
        app: nodejs #11
    spec: #12
      containers: #13
      - name: nodongo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

Step 9: Create Deployment In Kubernetes Cluster

create a deployment from this YAML file.
$ kubectl create -f deploy.yaml

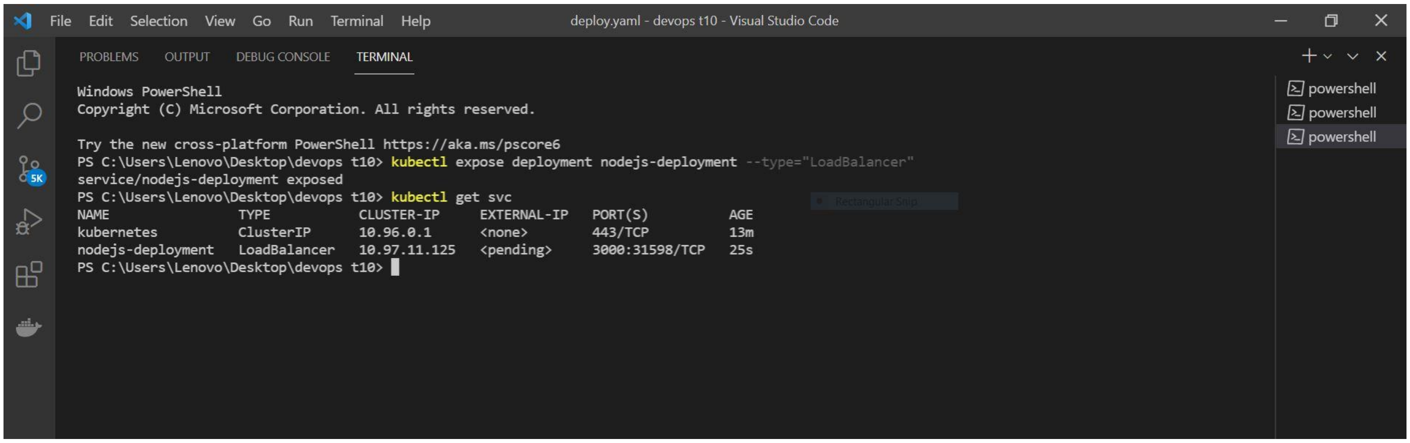


```
PS C:\Users\Lenovo\Desktop\devops t10> code deploy.yaml
PS C:\Users\Lenovo\Desktop\devops t10> kubectl create -f deploy.yaml
deployment.apps/nodejs-deployment created
PS C:\Users\Lenovo\Desktop\devops t10>
```



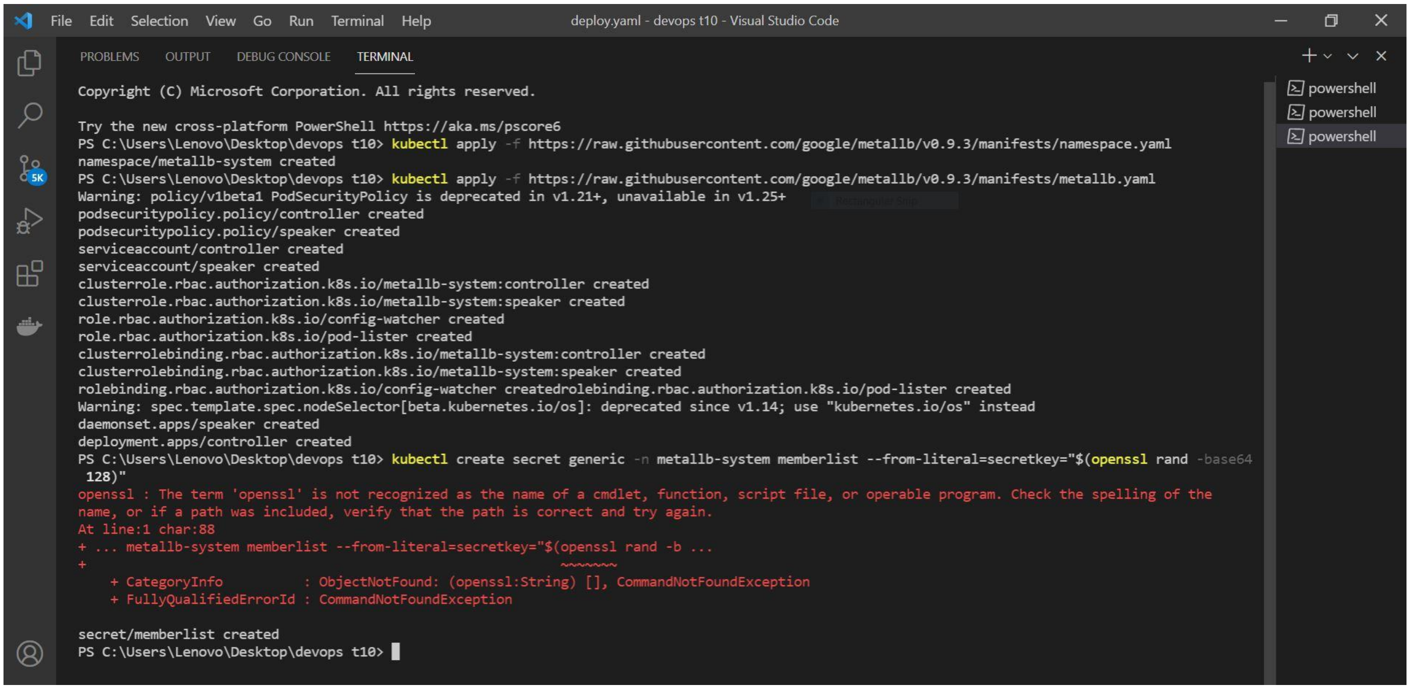
```
PS C:\Users\Lenovo\Desktop\devops t10> kubectl get deploy,po
NAME                                READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/nodejs-deployment   0/2     2            0           61s

NAME                                     READY   STATUS              RESTARTS   AGE
pod/nodejs-deployment-57547d448f-gwcgh   0/1     ContainerCreating   0          61s
pod/nodejs-deployment-57547d448f-vx8rw   0/1     ContainerCreating   0          61s
PS C:\Users\Lenovo\Desktop\devops t10>



PS C:\Users\Lenovo\Desktop\devops t10>
PS C:\Users\Lenovo\Desktop\devops t10>
PS C:\Users\Lenovo\Desktop\devops t10>
PS C:\Users\Lenovo\Desktop\devops t10>
```

Step 10: Expose The Deployment To The Internet

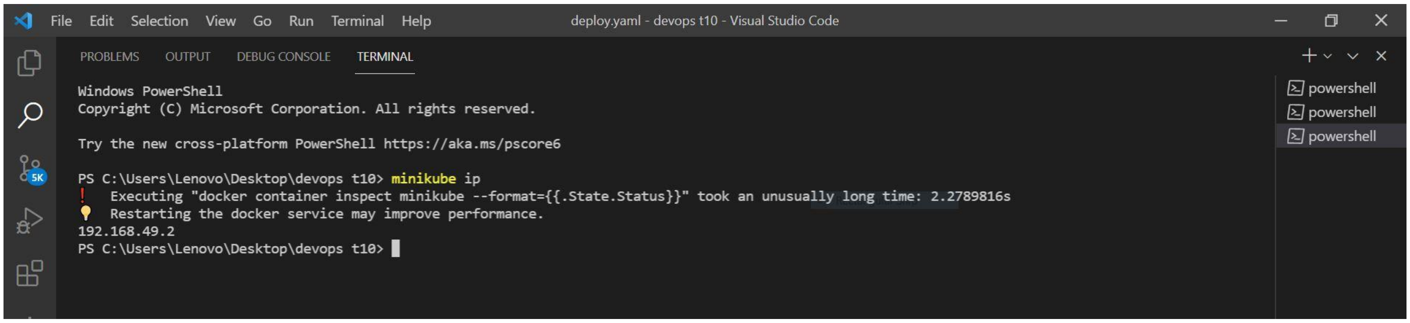


```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Users\Lenovo\Desktop\devops t10> kubectl expose deployment nodejs-deployment --type="LoadBalancer"
service/nodejs-deployment exposed
PS C:\Users\Lenovo\Desktop\devops t10> kubectl get svc
NAME                TYPE           CLUSTER-IP     EXTERNAL-IP   PORT(S)          AGE
kubernetes          ClusterIP      10.96.0.1      <none>        443/TCP          13m
nodejs-deployment   LoadBalancer   10.97.11.125   <pending>     3000:31598/TCP   25s
PS C:\Users\Lenovo\Desktop\devops t10>
```

Step 11: Using MetalLB In Your Minikube Environment

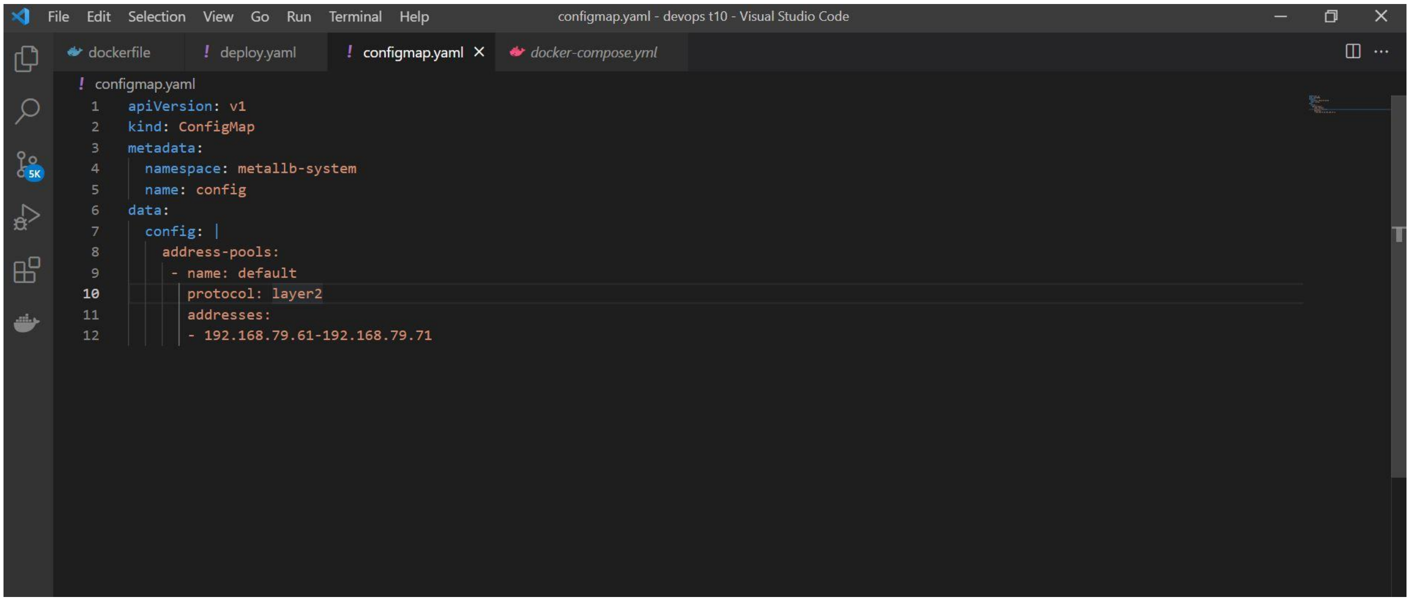

File Edit Selection View Go Run Terminal Help
deploy.yaml - devops t10 - Visual Studio Code
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
powershell
powershell
powershell

```
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Users\Lenovo\Desktop\devops t10> kubectl apply -f https://raw.githubusercontent.com/google/metallb/v0.9.3/manifests/namespace.yaml
namespace/metallb-system created
PS C:\Users\Lenovo\Desktop\devops t10> kubectl apply -f https://raw.githubusercontent.com/google/metallb/v0.9.3/manifests/metallb.yaml
Warning: policy/v1beta1 PodSecurityPolicy is deprecated in v1.21+, unavailable in v1.25+
podsecuritypolicy.policy/controller created
podsecuritypolicy.policy/speaker created
serviceaccount/controller created
serviceaccount/speaker created
clusterrole.rbac.authorization.k8s.io/metallb-system:controller created
clusterrole.rbac.authorization.k8s.io/metallb-system:speaker created
role.rbac.authorization.k8s.io/config-watcher created
role.rbac.authorization.k8s.io/pod-lister created
clusterrolebinding.rbac.authorization.k8s.io/metallb-system:controller created
clusterrolebinding.rbac.authorization.k8s.io/metallb-system:speaker created
rolebinding.rbac.authorization.k8s.io/config-watcher createdrolebinding.rbac.authorization.k8s.io/pod-lister created
Warning: spec.template.spec.nodeSelector[beta.kubernetes.io/os]: deprecated since v1.14; use "kubernetes.io/os" instead
daemonset.apps/speaker created
deployment.apps/controller created
PS C:\Users\Lenovo\Desktop\devops t10> kubectl create secret generic -n metallb-system memberlist --from-literal=secretkey="$(openssl rand -base64
 128)"
openssl : The term 'openssl' is not recognized as the name of a cmdlet, function, script file, or operable program. Check the spelling of the
name, or if a path was included, verify that the path is correct and try again.
At line:1 char:88
+ ... metallb-system memberlist --from-literal=secretkey="$(openssl rand -b ...
+                                                          ~~~~~~~
    + CategoryInfo          : ObjectNotFound: (openssl:String) [], CommandNotFoundException
    + FullyQualifiedErrorId : CommandNotFoundException

secret/memberlist created
PS C:\Users\Lenovo\Desktop\devops t10> 
```


File Edit Selection View Go Run Terminal Help
deploy.yaml - devops t10 - Visual Studio Code
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
powershell
powershell
powershell

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

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

PS C:\Users\Lenovo\Desktop\devops t10> minikube ip
!    Executing "docker container inspect minikube --format={{.State.Status}}" took an unusually long time: 2.2789816s
💡    Restarting the docker service may improve performance.
192.168.49.2
PS C:\Users\Lenovo\Desktop\devops t10> 
```

create a config map for the address pool by running the following command: **$ code configmap.yaml**

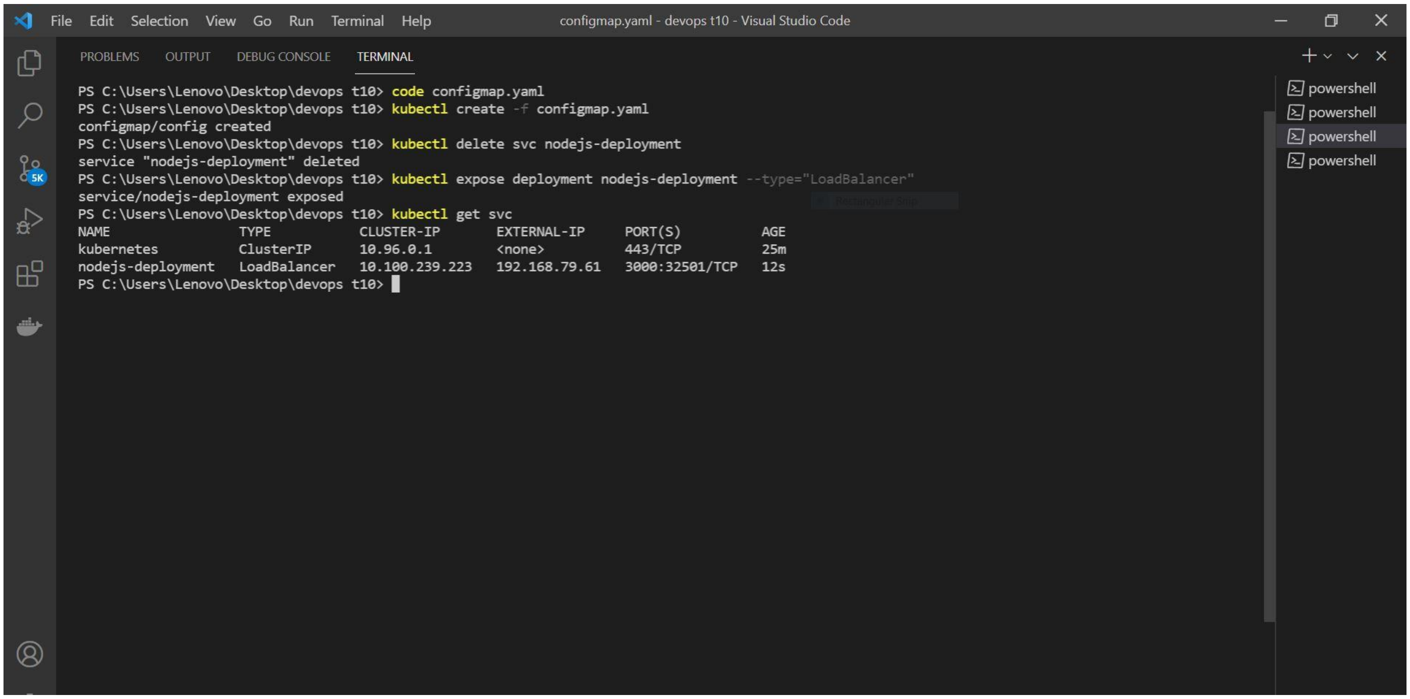

File Edit Selection View Go Run Terminal Help
configmap.yaml - devops t10 - Visual Studio Code
dockerfile
deploy.yaml
configmap.yaml
docker-compose.yml
configmap.yaml

```yaml
apiVersion: v1
kind: ConfigMap
metadata:
  namespace: metallb-system
  name: config
data:
  config: |
    address-pools:
    - name: default
      protocol: layer2
      addresses:
      - 192.168.79.61-192.168.79.71
```

In this configuration, MetalLB is instructed to hand out addresses from **192.168.79.61** to **192.168.79.71**. After that, we'll create a config map in the metallb-system namespace.

```
$ kubectl create -f configmap.
$ yaml kubectl delete svc nodejs-deployment
$ kubectl expose deployment nodejs-deployment--type="LoadBalancer"
```

Now that's done, you'll be getting External IP.



```
PS C:\Users\Lenovo\Desktop\devops t10> code configmap.yaml
PS C:\Users\Lenovo\Desktop\devops t10> kubectl create -f configmap.yaml
configmap/config created
PS C:\Users\Lenovo\Desktop\devops t10> kubectl delete svc nodejs-deployment
service "nodejs-deployment" deleted
PS C:\Users\Lenovo\Desktop\devops t10> kubectl expose deployment nodejs-deployment --type="LoadBalancer"
service/nodejs-deployment exposed
PS C:\Users\Lenovo\Desktop\devops t10> kubectl get svc
NAME                TYPE           CLUSTER-IP       EXTERNAL-IP     PORT(S)          AGE
kubernetes          ClusterIP      10.96.0.1        <none>          443/TCP          25m
nodejs-deployment   LoadBalancer   10.100.239.223   192.168.79.61   3000:32501/TCP   12s
PS C:\Users\Lenovo\Desktop\devops t10>
```

"Thank you"

***************************** END *****************************