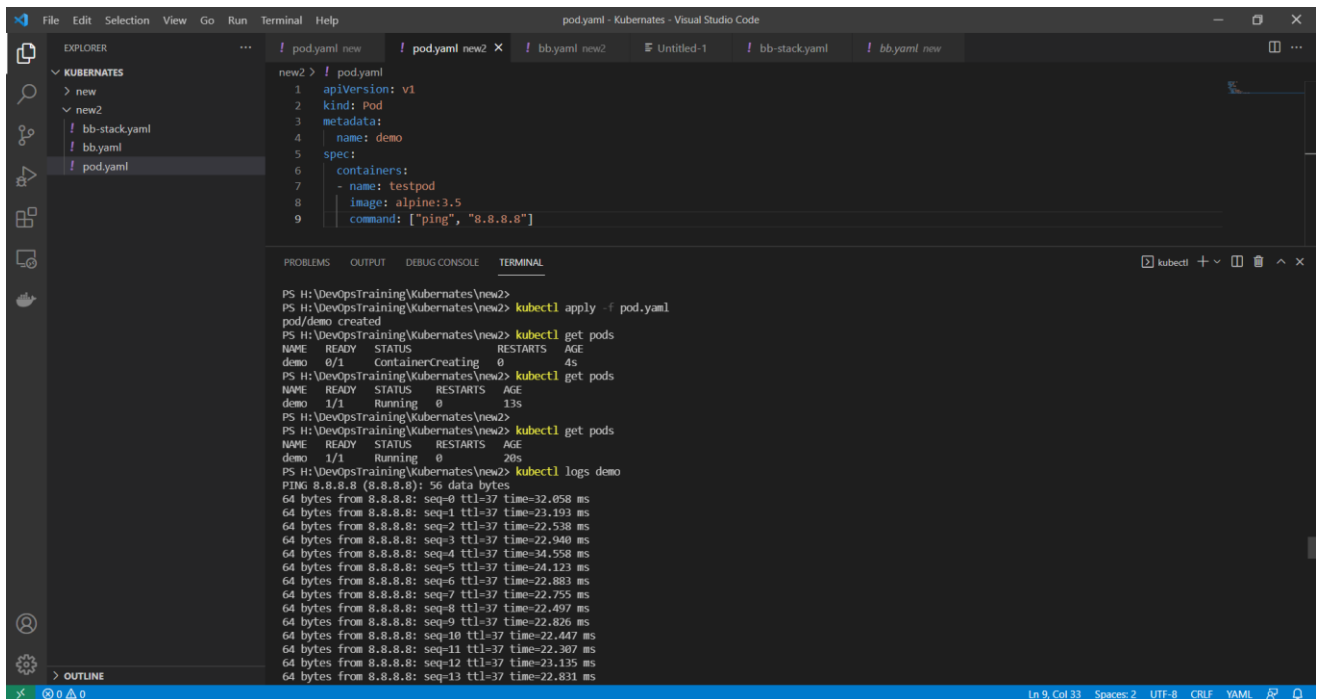


# KUBERNETES OUTPUT

## Output Of Codes in file Kubernetes.txt

### Enabling Kubernetes



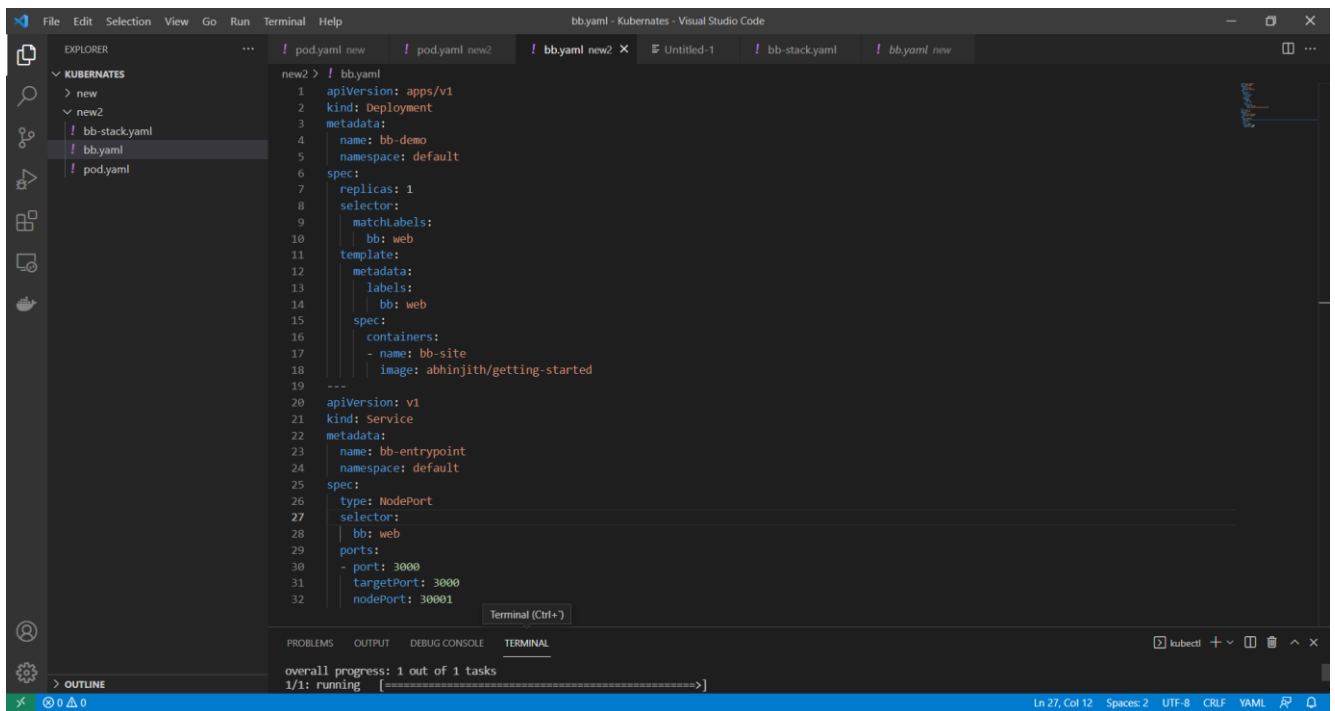
The screenshot shows a Visual Studio Code editor with a file explorer on the left containing a 'KUBERNETES' folder with subfolders 'new' and 'new2'. The 'pod.yaml' file is open in the editor, showing a pod configuration for 'demo' using the 'alpine:3.5' image with a 'ping' command. The terminal at the bottom shows the execution of 'kubectl apply -f pod.yaml', followed by 'kubectl get pods' which shows the pod in a 'Running' state. The 'kubectl logs demo' command is also executed, showing a series of 'ping' outputs from 8.8.8.8.

```
new2 > ! pod.yaml
1 apiVersion: v1
2 kind: Pod
3 metadata:
4   name: demo
5 spec:
6   containers:
7     - name: testpod
8       image: alpine:3.5
9       command: ["ping", "8.8.8.8"]

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

PS H:\DevOpsTraining\Kubernetes\new2>
PS H:\DevOpsTraining\Kubernetes\new2> kubectl apply -f pod.yaml
pod/demo created
PS H:\DevOpsTraining\Kubernetes\new2> kubectl get pods
NAME READY STATUS RESTARTS AGE
demo 0/1 ContainerCreating 0 4s
PS H:\DevOpsTraining\Kubernetes\new2> kubectl get pods
NAME READY STATUS RESTARTS AGE
demo 1/1 Running 0 20s
PS H:\DevOpsTraining\Kubernetes\new2> kubectl get pods
NAME READY STATUS RESTARTS AGE
demo 1/1 Running 0 20s
PS H:\DevOpsTraining\Kubernetes\new2> kubectl logs demo
PING 8.8.8.8 (8.8.8.8): 56 data bytes
64 bytes from 8.8.8.8: seq=0 ttl=37 time=32.058 ms
64 bytes from 8.8.8.8: seq=1 ttl=37 time=23.193 ms
64 bytes from 8.8.8.8: seq=2 ttl=37 time=22.538 ms
64 bytes from 8.8.8.8: seq=3 ttl=37 time=22.940 ms
64 bytes from 8.8.8.8: seq=4 ttl=37 time=34.558 ms
64 bytes from 8.8.8.8: seq=5 ttl=37 time=24.123 ms
64 bytes from 8.8.8.8: seq=6 ttl=37 time=22.883 ms
64 bytes from 8.8.8.8: seq=7 ttl=37 time=22.755 ms
64 bytes from 8.8.8.8: seq=8 ttl=37 time=22.497 ms
64 bytes from 8.8.8.8: seq=9 ttl=37 time=22.826 ms
64 bytes from 8.8.8.8: seq=10 ttl=37 time=22.447 ms
64 bytes from 8.8.8.8: seq=11 ttl=37 time=22.307 ms
64 bytes from 8.8.8.8: seq=12 ttl=37 time=23.135 ms
64 bytes from 8.8.8.8: seq=13 ttl=37 time=22.831 ms
```

### Deploy to Kubernetes

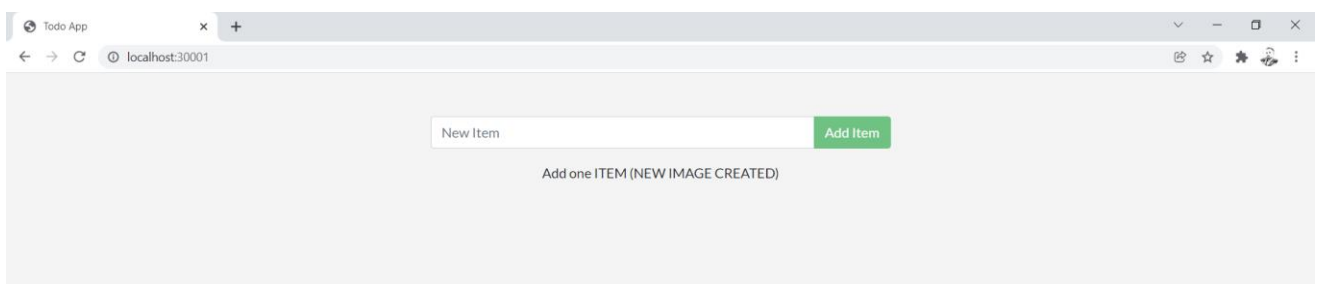


The screenshot shows a Visual Studio Code editor with a file explorer on the left containing a 'KUBERNETES' folder with subfolders 'new' and 'new2'. The 'bb.yaml' file is open in the editor, showing a deployment configuration for 'bb-demo' using the 'abhinjith/getting-started' image. The terminal at the bottom shows the execution of 'kubectl apply -f bb.yaml', followed by 'kubectl get pods' which shows the pod in a 'Running' state. The 'kubectl logs demo' command is also executed, showing a series of 'ping' outputs from 8.8.8.8.

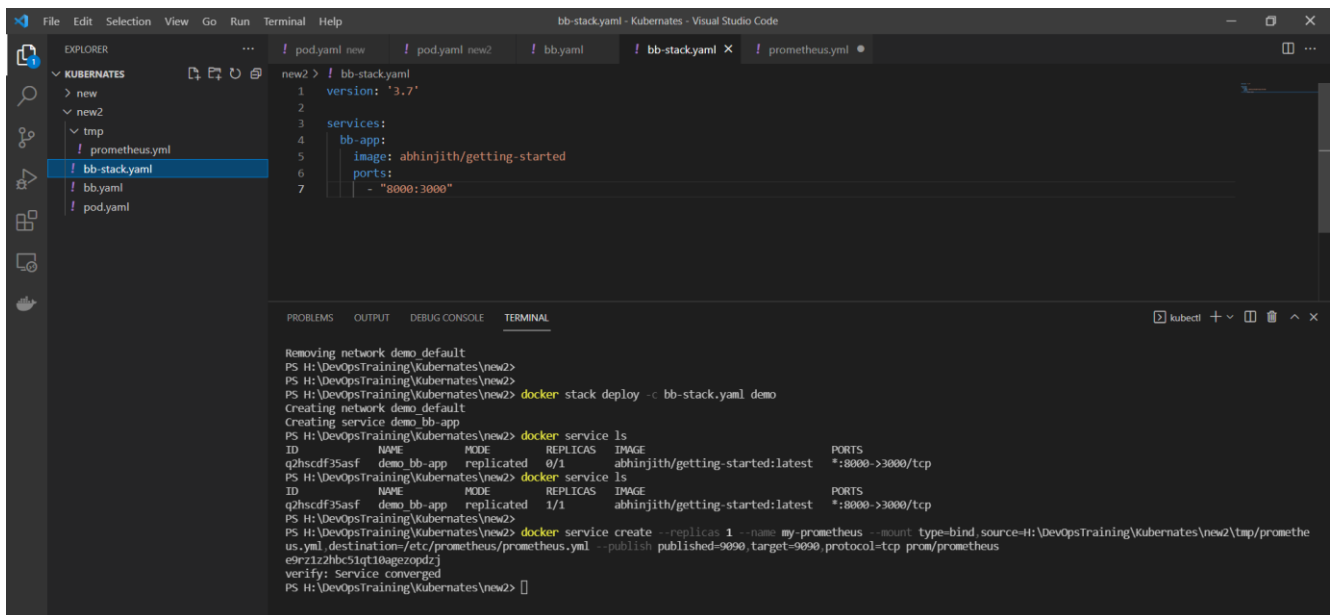
```
new2 > ! bb.yaml
1 apiVersion: apps/v1
2 kind: Deployment
3 metadata:
4   name: bb-demo
5   namespace: default
6 spec:
7   replicas: 1
8   selector:
9     matchLabels:
10      bb: web
11   template:
12     metadata:
13       labels:
14         bb: web
15     spec:
16       containers:
17         - name: bb-site
18           image: abhinjith/getting-started
19 ---
20 apiVersion: v1
21 kind: Service
22 metadata:
23   name: bb-entrypoint
24   namespace: default
25 spec:
26   type: NodePort
27   selector:
28     bb: web
29   ports:
30     - port: 3000
31       targetPort: 3000
32       nodePort: 30001

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

overall progress: 1 out of 1 tasks
1/1: running [=====>]
```

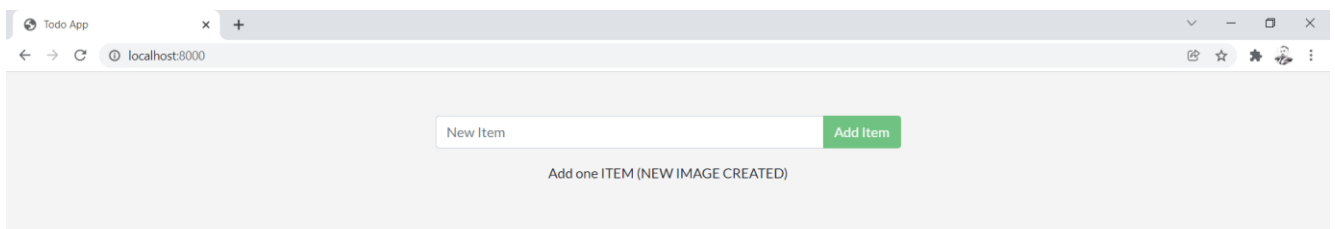


## Deploy To Swarm

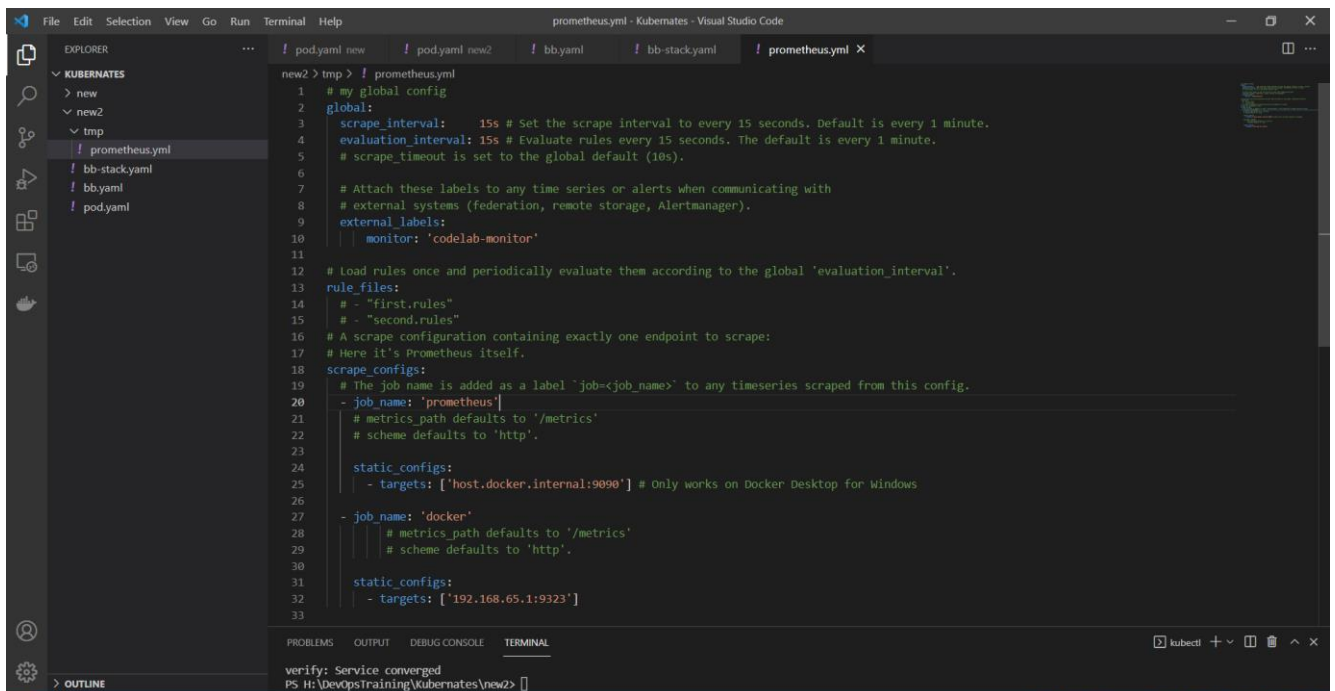


```
new2 > ! bb-stack.yaml
1 version: '3.7'
2
3 services:
4   bb-app:
5     image: abhinjith/getting-started
6     ports:
7       - "8000:3000"
```

```
Removing network demo/default
PS H:\DevOpsTraining\Kubernetes\new2>
PS H:\DevOpsTraining\Kubernetes\new2> docker stack deploy -c bb-stack.yaml demo
Creating network demo/default
Creating service demo/bb-app
PS H:\DevOpsTraining\Kubernetes\new2> docker service ls
ID                NAME                MODE                REPLICAS            IMAGE                PORTS
q2hscdf35asf     demo-bb-app         replicated          0/1                 abhinjith/getting-started:latest *:*8000->3000/tcp
PS H:\DevOpsTraining\Kubernetes\new2> docker service ls
ID                NAME                MODE                REPLICAS            IMAGE                PORTS
q2hscdf35asf     demo-bb-app         replicated          1/1                 abhinjith/getting-started:latest *:*8000->3000/tcp
PS H:\DevOpsTraining\Kubernetes\new2> docker service create --replicas 1 --name my-prometheus --mount type=bind,source=H:\DevOpsTraining\Kubernetes\new2\tmp\prometheus.yml,destination=/etc/prometheus/prometheus.yml --publish published=9090,target=9090,protocol=tcp prom/prometheus
verify: Service converged
PS H:\DevOpsTraining\Kubernetes\new2> []
```



## Collect Docker metrics with Prometheus



```
new2 > tmp > ! prometheus.yml
1 # my global config
2 global:
3   scrape_interval: 15s # Set the scrape interval to every 15 seconds. Default is every 1 minute.
4   evaluation_interval: 15s # Evaluate rules every 15 seconds. The default is every 1 minute.
5   # scrape_timeout is set to the global default (10s).
6
7   # Attach these labels to any time series or alerts when communicating with
8   # external systems (federation, remote storage, Alertmanager).
9   external_labels:
10    monitor: 'codelab-monitor'
11
12 # Load rules once and periodically evaluate them according to the global 'evaluation_interval'.
13 rule_files:
14   # - "first.rules"
15   # - "second.rules"
16 # A scrape configuration containing exactly one endpoint to scrape:
17 # Here it's Prometheus itself.
18 scrape_configs:
19   # The job name is added as a label 'job=<job_name>' to any timeseries scraped from this config.
20   - job_name: 'prometheus'
21     # metrics_path defaults to '/metrics'
22     # scheme defaults to 'http'.
23
24     static_configs:
25       - targets: ['host.docker.internal:9090'] # Only works on Docker Desktop for Windows
26
27   - job_name: 'docker'
28     # metrics_path defaults to '/metrics'
29     # scheme defaults to 'http'.
30
31     static_configs:
32       - targets: ['192.168.65.1:9323']
33
```

```
verify: Service converged
PS H:\DevOpsTraining\Kubernetes\new2> []
```

Prometheus Time Series Collecti...

localhost:9090/targets

Prometheus

Alerts

Graph

Status

Help

Classic UI

## Targets

All

Unhealthy

Collapse All

docker (0/1 up)

show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://192.168.65.1:9323/metrics	DOWN	instance="192.168.65.1:9323" job="docker"	22.589s ago	10.1s	Get "http://192.168.65.1:9323/metrics": context deadline exceeded

prometheus (1/1 up)

show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://host.docker.internal:9090/metrics	UP	instance="host.docker.internal:9090" job="prometheus"	12.664s ago	19.672ms	

## Graph

