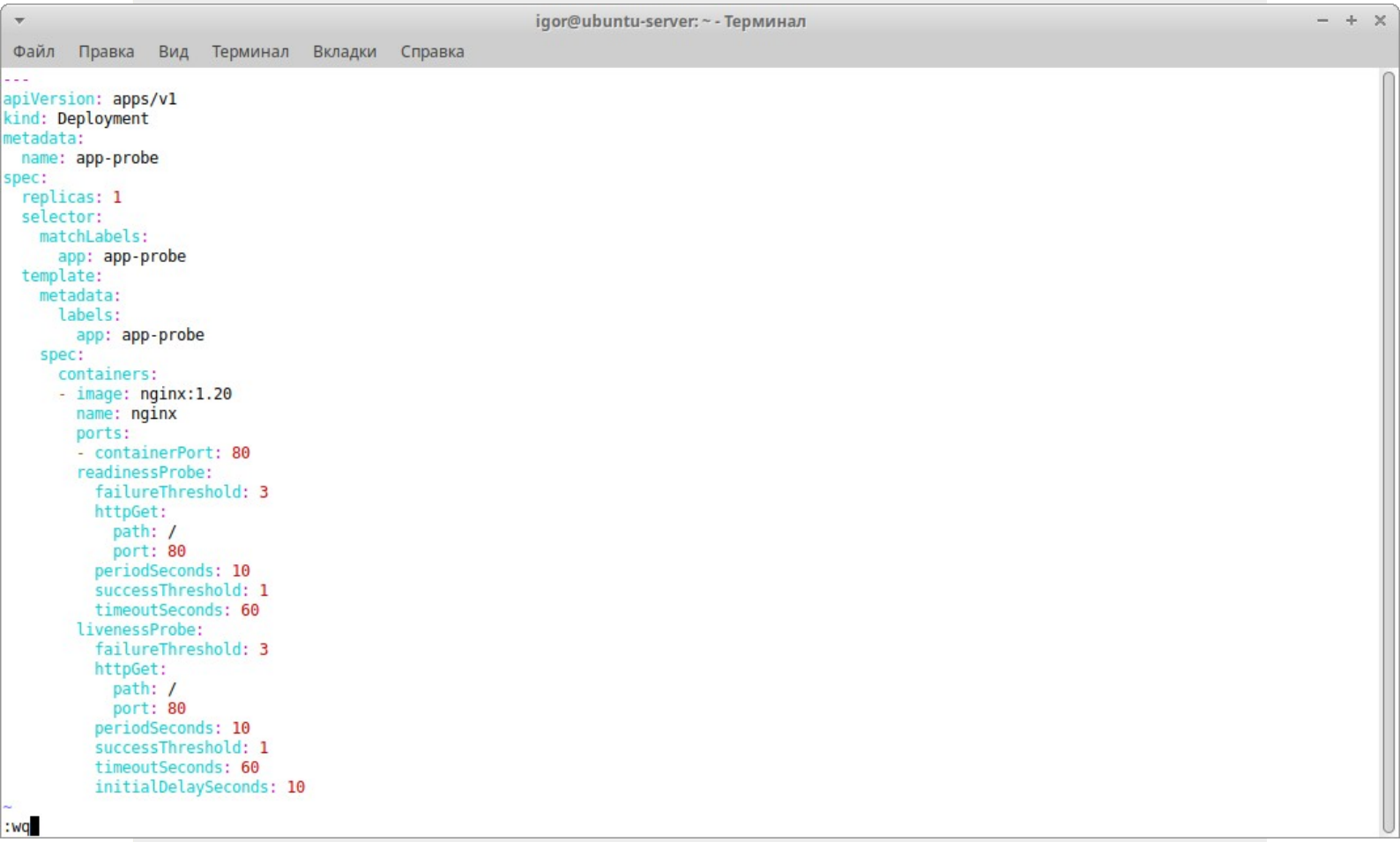
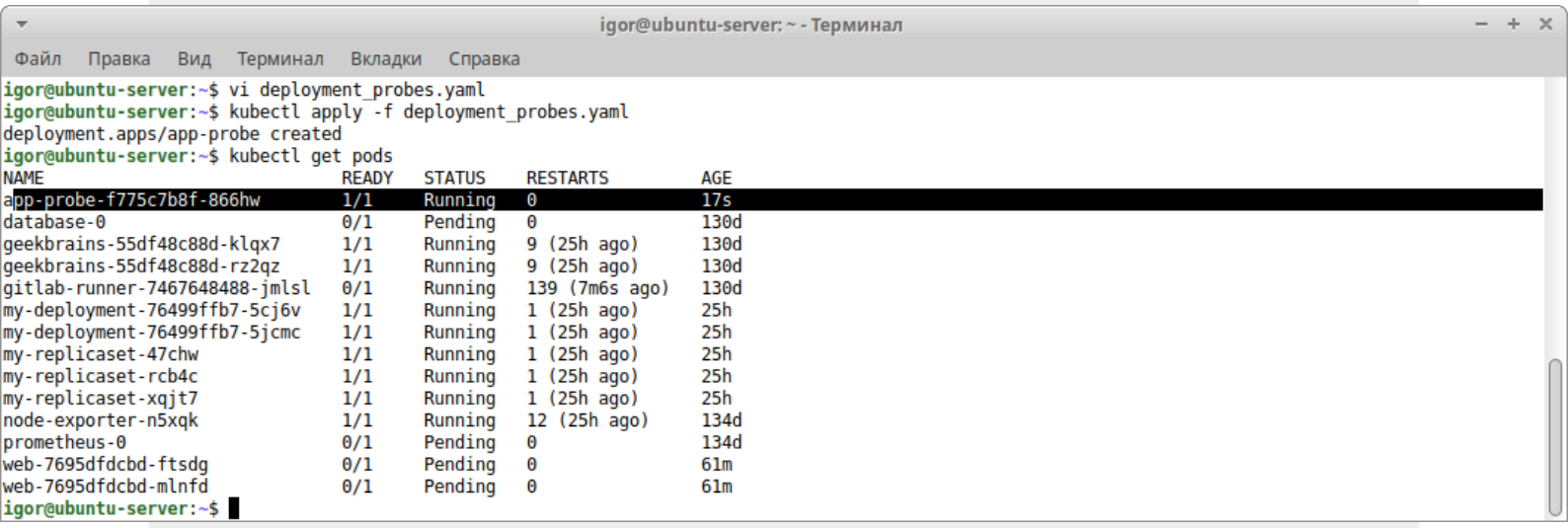
```
vi deployment_probes.yaml
```




```
kubectl apply -f deployment_probes.yaml

kubectl get pods
```



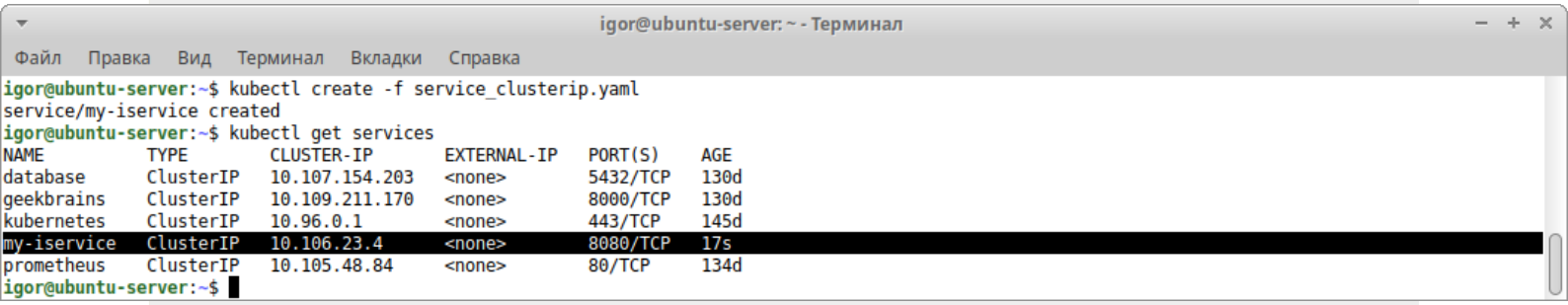

```
kubectl describe pod app-probe-f775c7b8f-866hw
```

```
igor@ubuntu-server: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
igor@ubuntu-server:~$ kubectl describe pod app-probe-f775c7b8f-866hw
Name:         app-probe-f775c7b8f-866hw
Namespace:    default
Priority:     0
Node:         minikube/192.168.49.2
Start Time:   Sat, 26 Mar 2022 21:02:19 +0000
Labels:       app=app-probe
              pod-template-hash=f775c7b8f
Annotations:  <none>
Status:       Running
IP:           172.17.0.18
IPs:
  IP:           172.17.0.18
Controlled By:  ReplicaSet/app-probe-f775c7b8f
Containers:
  nginx:
    Container ID:   docker://992a2b0c4093a3c0a02b233b6e8d31b50be9033acced280bcfafa81b030439fe
    Image:          nginx:1.20
    Image ID:       docker-pullable://nginx@sha256:70ffa5088f7a3be061ae2b0816abc36f52db185c023c4bc039d7fb8fe490fcf8
    Port:           80/TCP
    Host Port:      0/TCP
    State:          Running
      Started:      Sat, 26 Mar 2022 21:02:31 +0000
    Ready:          True
    Restart Count:  0
    Liveness:       http-get http://:80/ delay=10s timeout=60s period=10s #success=1 #failure=3
    Readiness:      http-get http://:80/ delay=0s timeout=60s period=10s #success=1 #failure=3
    Environment:    <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-mr7hb (ro)
Conditions:
  Type              Status
  Initialized       True
  Ready             True
  ContainersReady   True
  PodScheduled      True
Volumes:
  kube-api-access-mr7hb:
    Type:                    Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds:  3607
    ConfigMapName:           kube-root-ca.crt
    ConfigMapOptional:       <nil>
    DownwardAPI:             true
QoS Class:                   BestEffort
Node-Selectors:              <none>
Tolerations:                 node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
                             node.kubernetes.io/unreachable:NoExecute op=Exists for 300s

Events:
  Type    Reason     Age   From               Message
  ----    ------     ----  ----               -------
  Normal  Scheduled  90s   default-scheduler  Successfully assigned default/app-probe-f775c7b8f-866hw to minikube
  Normal  Pulling    90s   kubelet            Pulling image "nginx:1.20"
  Normal  Pulled     81s   kubelet            Successfully pulled image "nginx:1.20" in 8.437542896s
  Normal  Created    79s   kubelet            Created container nginx
  Normal  Started    79s   kubelet            Started container nginx
igor@ubuntu-server:~$
```

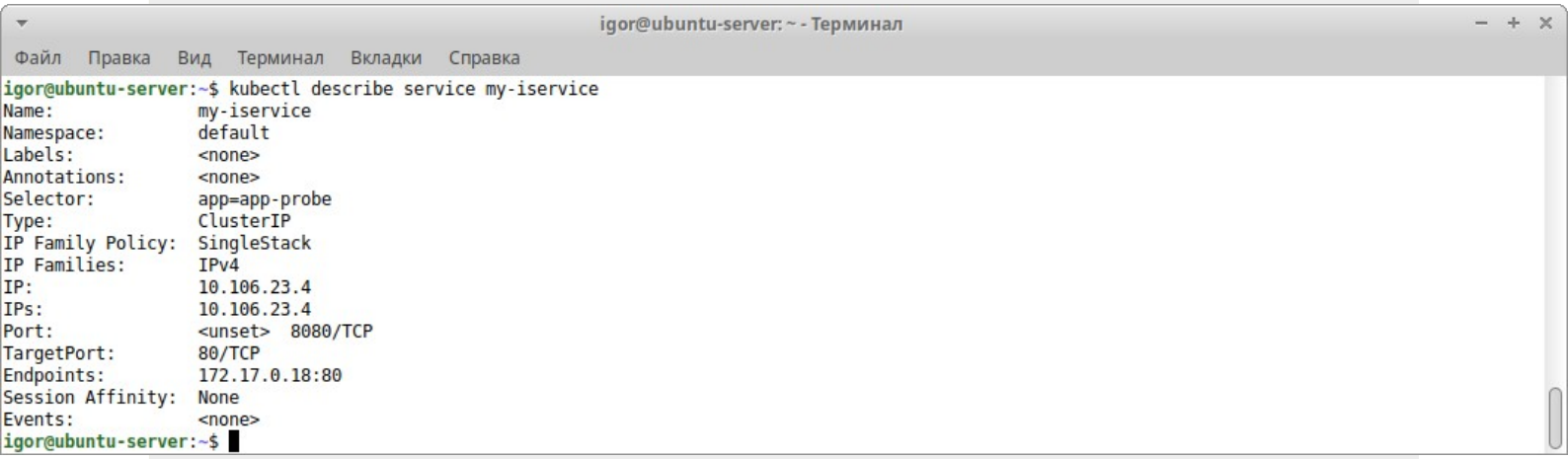
vi service_clusterip.yaml

```
igor@ubuntu-server: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
---
apiVersion: v1
kind: Service
metadata:
  name: my-iservice
spec:
  ports:
  - port: 8080
    targetPort: 80
  selector:
    app: app-probe
  type: ClusterIP
~
:wq
```

```
kubectl create -f service_clusterip.yaml

kubectl get services
```

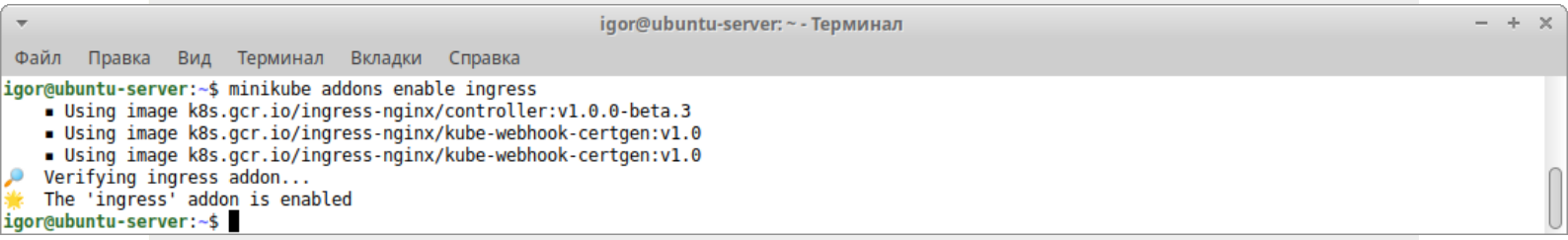



```
kubectl describe service my-iservice
```

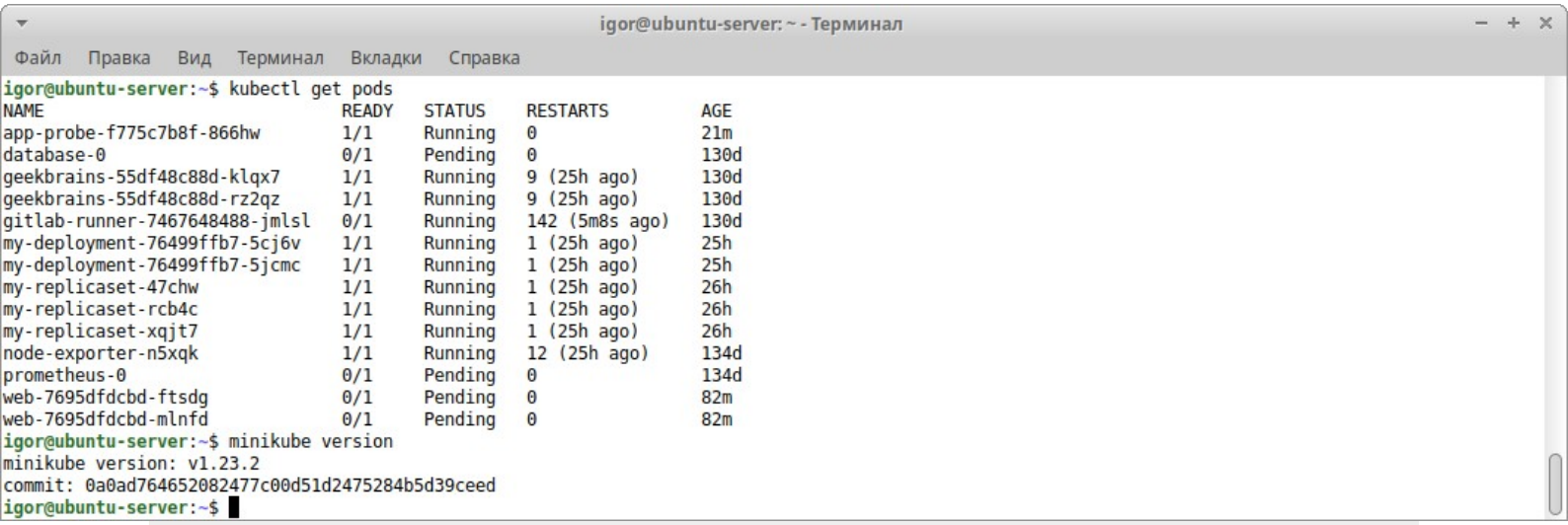


Для выполнения домашнего задания

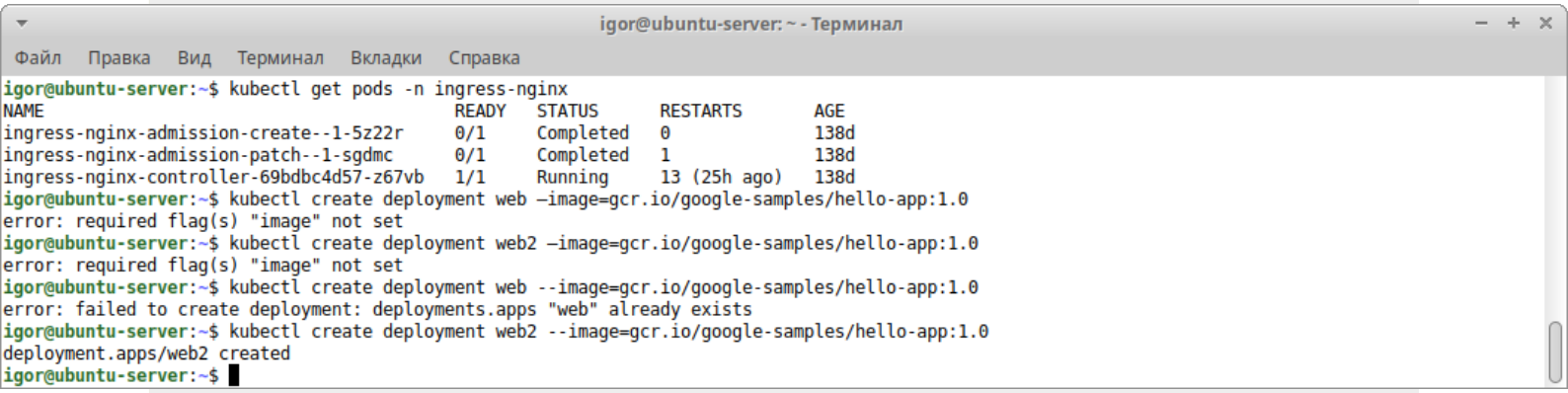
```
minikube addons enable ingress
```



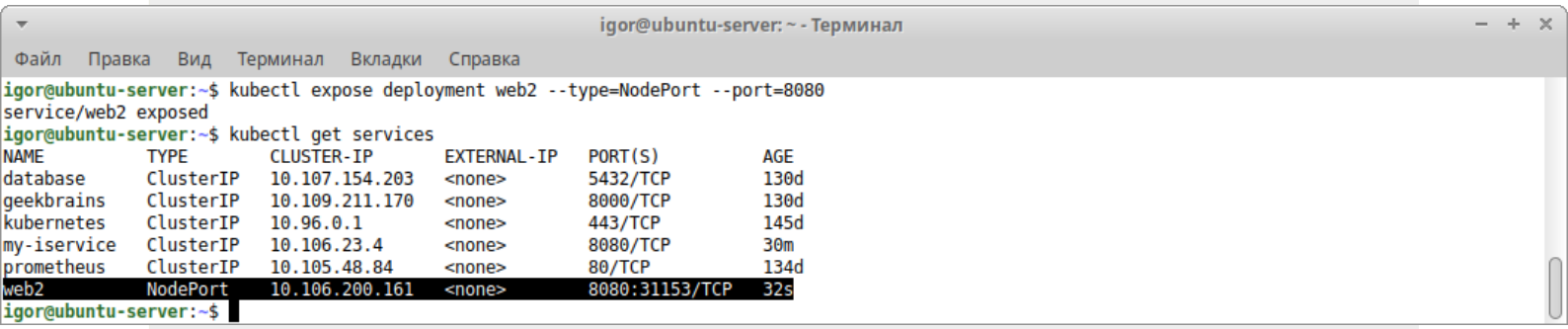

```
kubectl get pods

minikube version
```



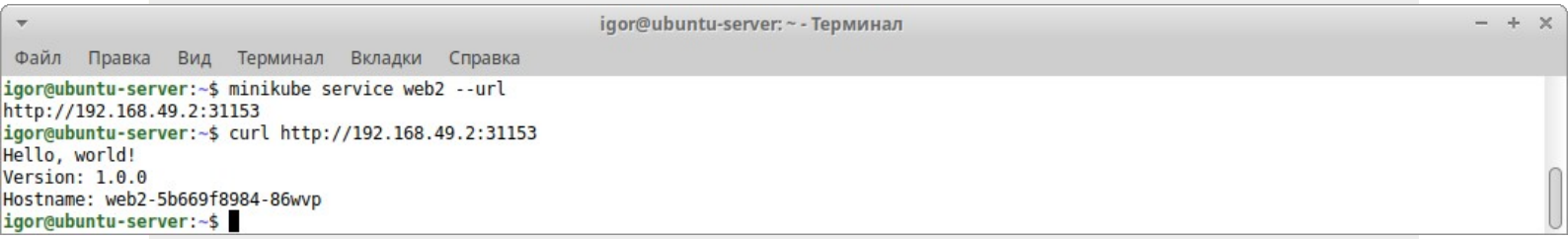

```
kubectl get pods -n ingress-nginx

kubectl create deployment web --image=gcr.io/google-samples/hello-app:1.0

error: required flag(s) "image" not set

kubectl create deployment web2 --image=gcr.io/google-samples/hello-app:1.0
```



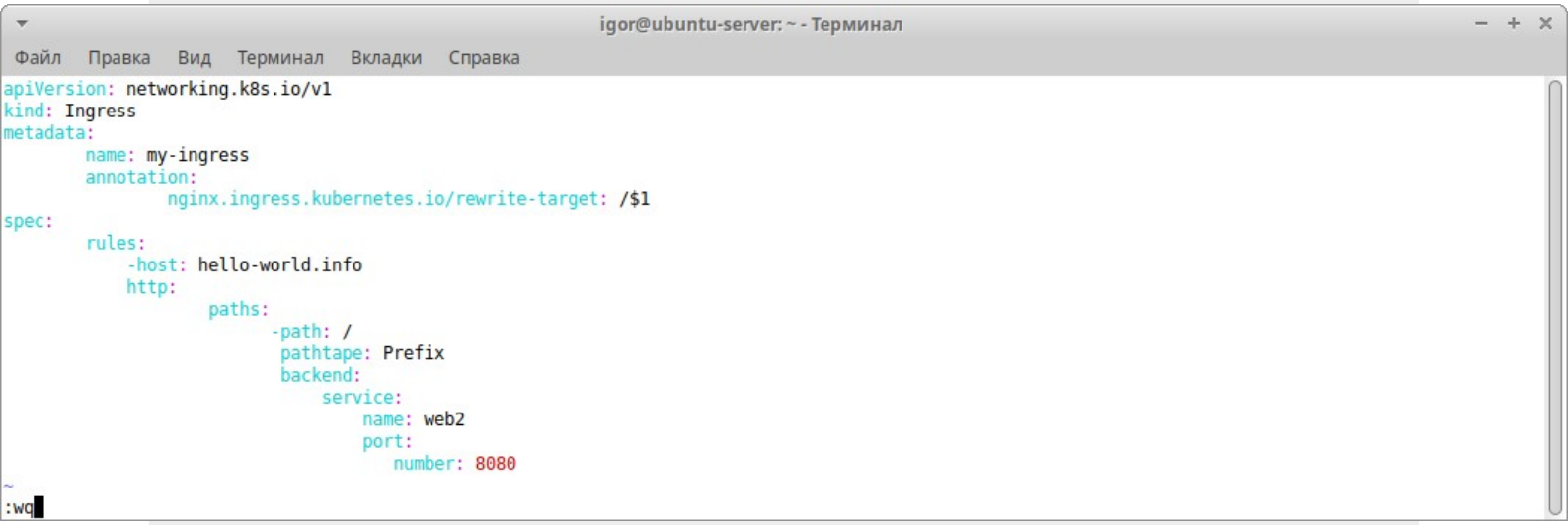

```
kubectl expose deployment web2 --type=NodePort --port=8080

kubectl get services
```



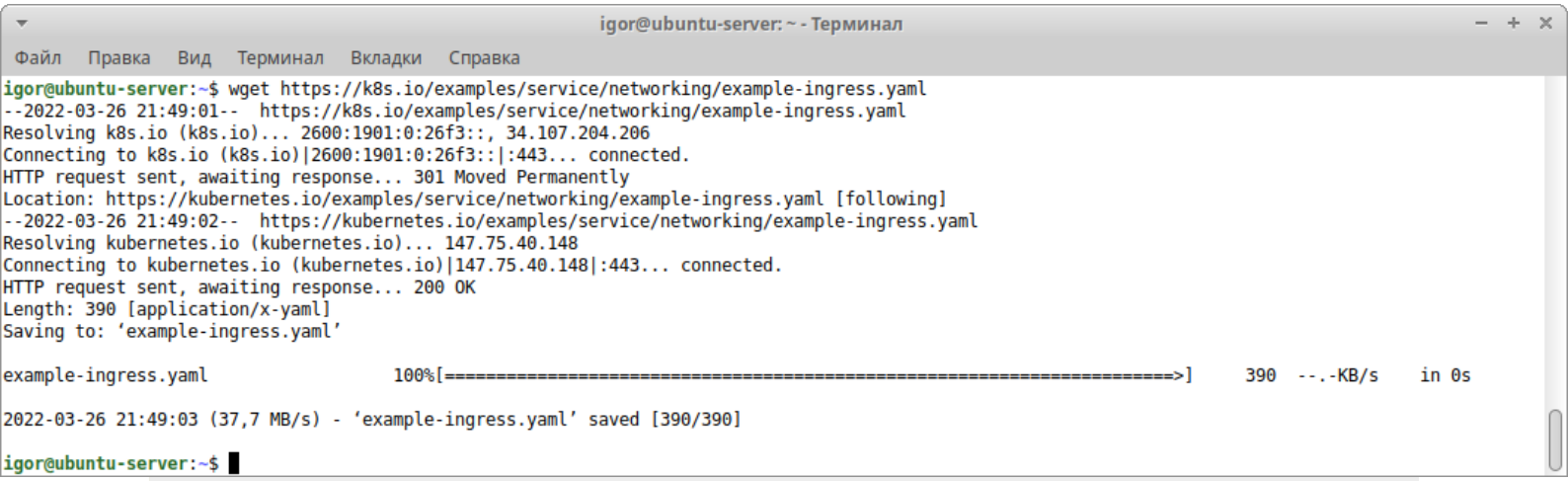
```
minikube service web2 --url

http://192.168.49.2:31153

curl http://192.168.49.2:31153
```



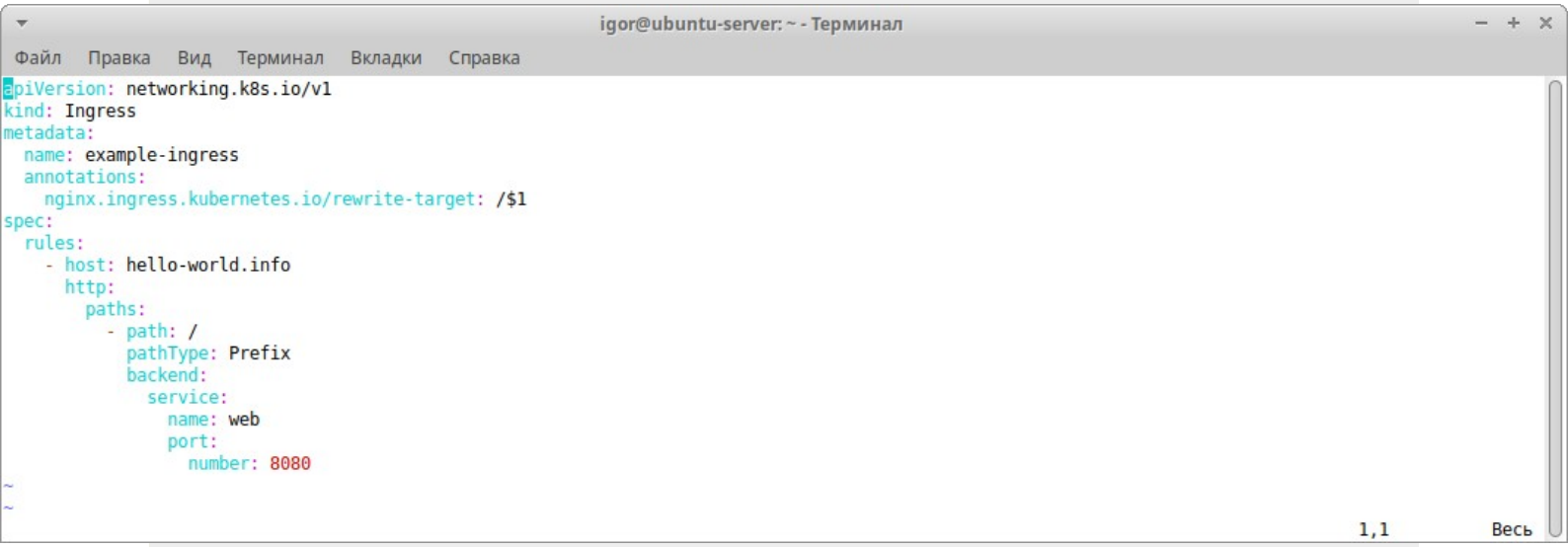

```
vi hw.yaml
```




```
kubectl apply -f hw.yaml

error: error parsing hw.yaml: error converting YAML to JSON: yaml:
line 13: mapping values are not allowed in this context

wget https://k8s.io/examples/service/networking/example-
ingress.yaml
```



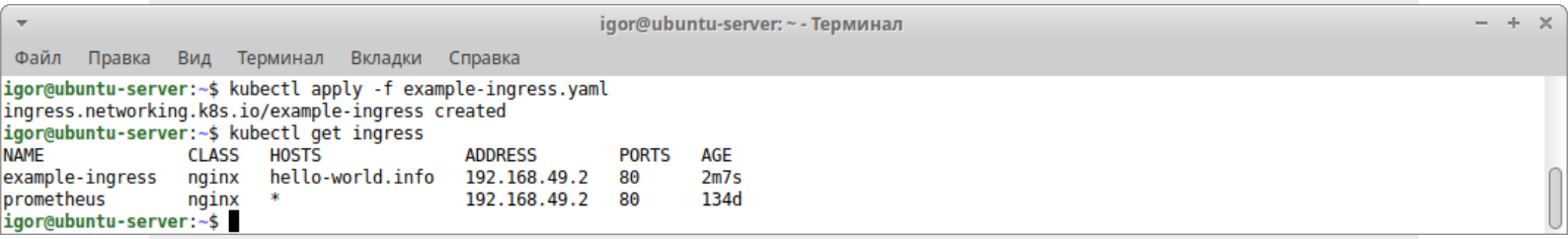
vi example-ingress.yaml



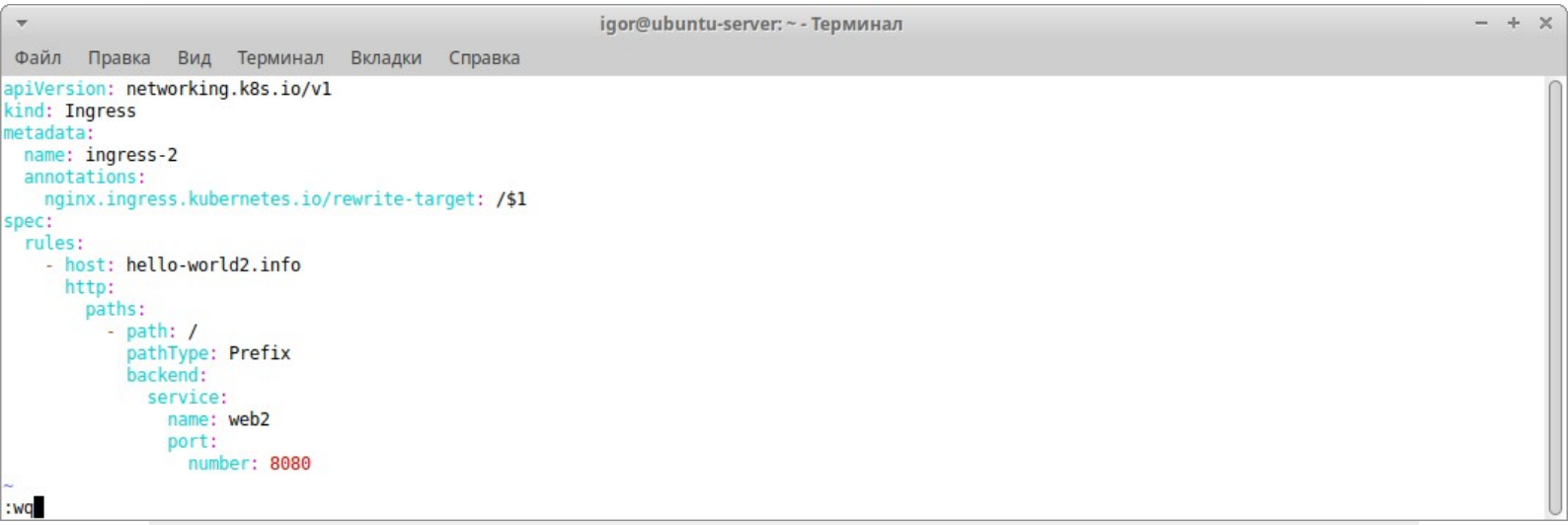
kubectl apply -f example-ingress.yaml

ingress.networking.k8s.io/example-ingress created

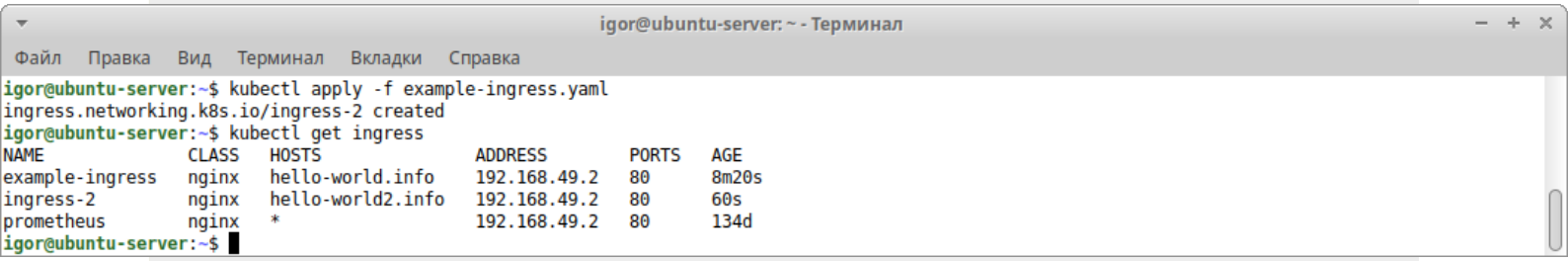
kubectl get ingress



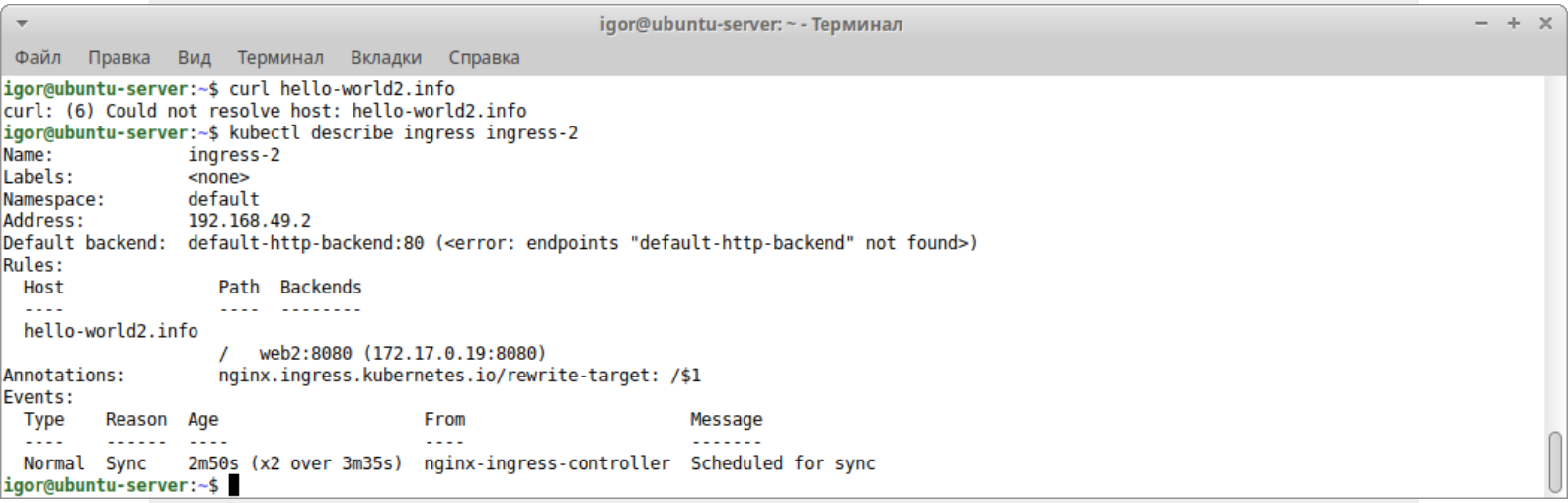
vi example-ingress.yaml




```
kubectl apply -f example-ingress.yaml

ingress.networking.k8s.io/ingress-2 created

kubectl get ingress
```

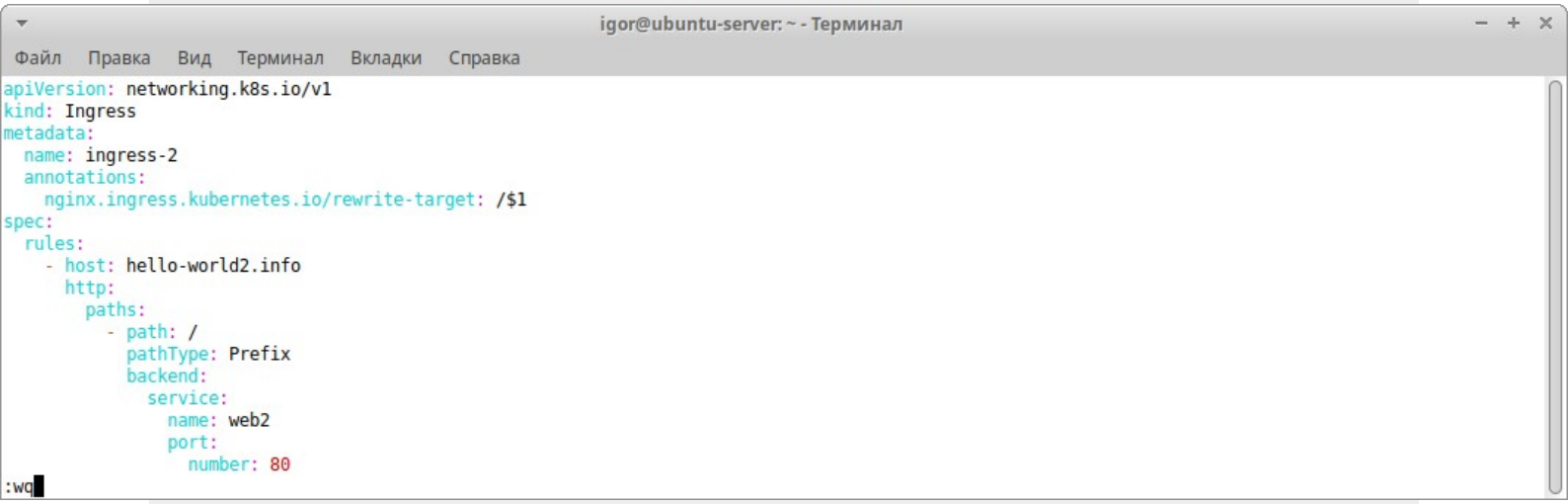



```
curl hello-world2.info

curl: (6) Could not resolve host: hello-world2.info

kubectl describe ingress ingress-2
```

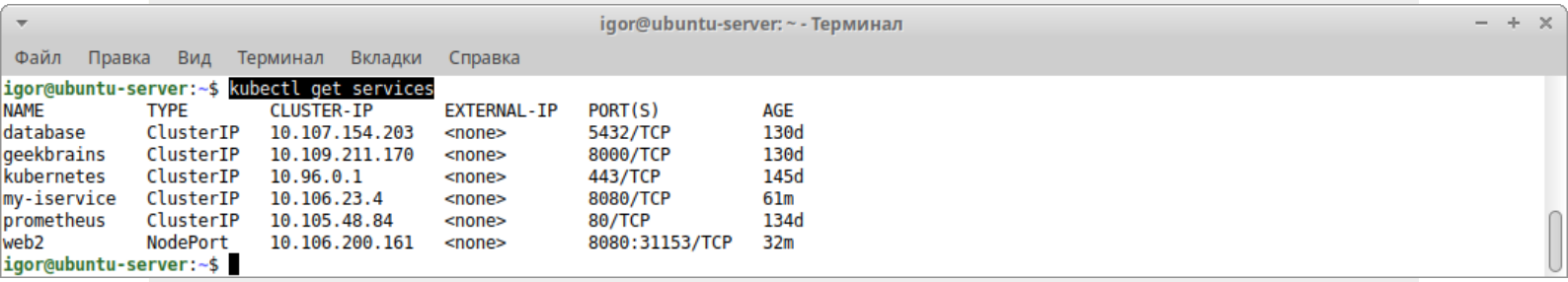


```
172.17.0.19
```

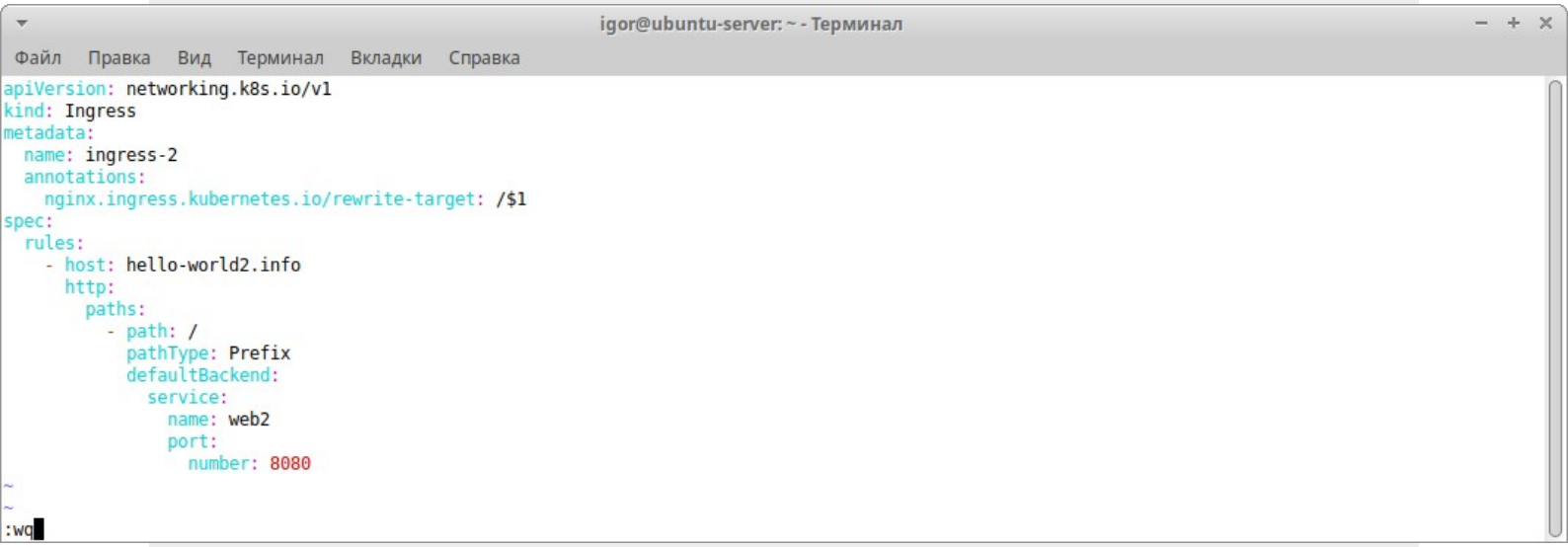
```
vi example-ingress.yaml
```




```
kubectl get services
```



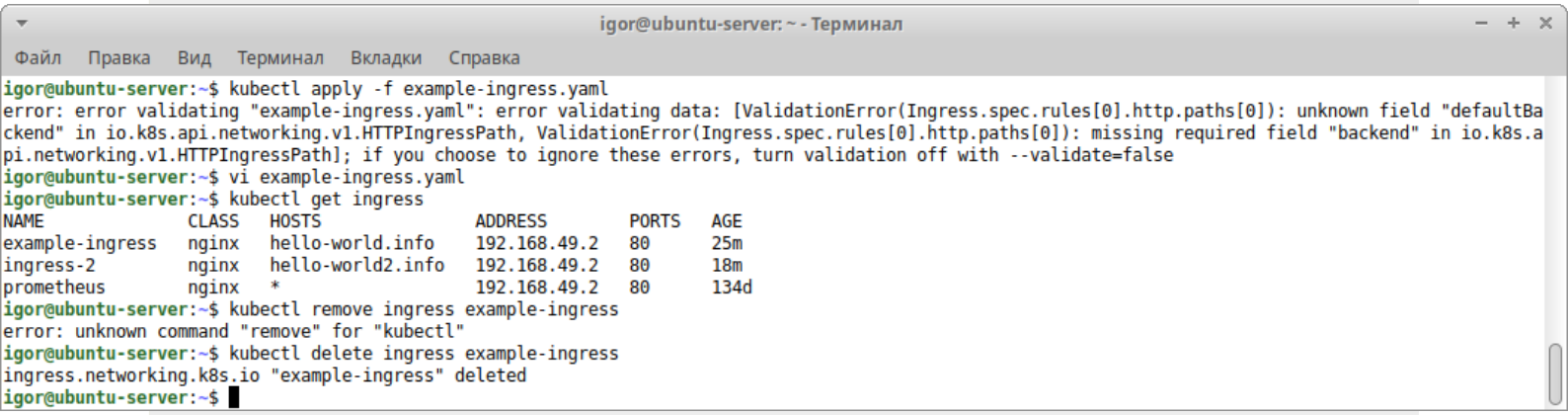

```
vi example-ingress.yaml
```



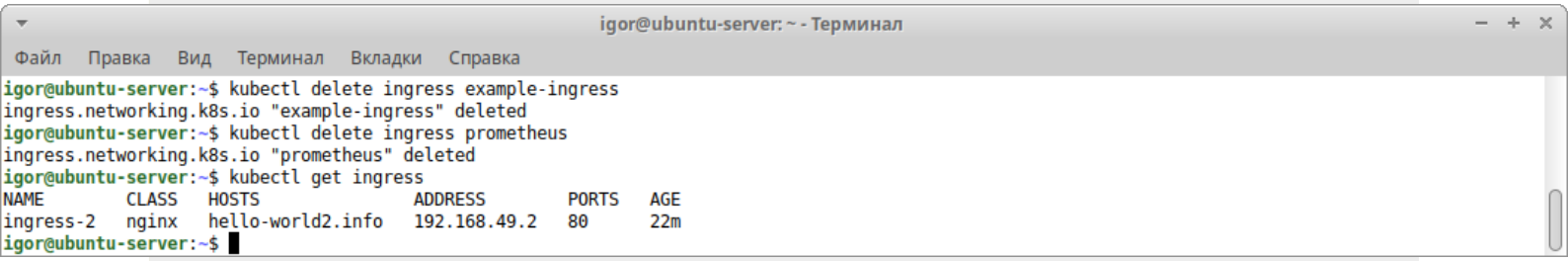

```
kubectl apply -f example-ingress.yaml
```

```
error: error validating "example-ingress.yaml": error validating
data: [ValidationError(Ingress.spec.rules[0].http.paths[0]):
unknown field "defaultBackend" in
io.k8s.api.networking.v1.HTTPIngressPath,
ValidationError(Ingress.spec.rules[0].http.paths[0]): missing
required field "backend" in
io.k8s.api.networking.v1.HTTPIngressPath]; if you choose to ignore
these errors, turn validation off with --validate=false
```

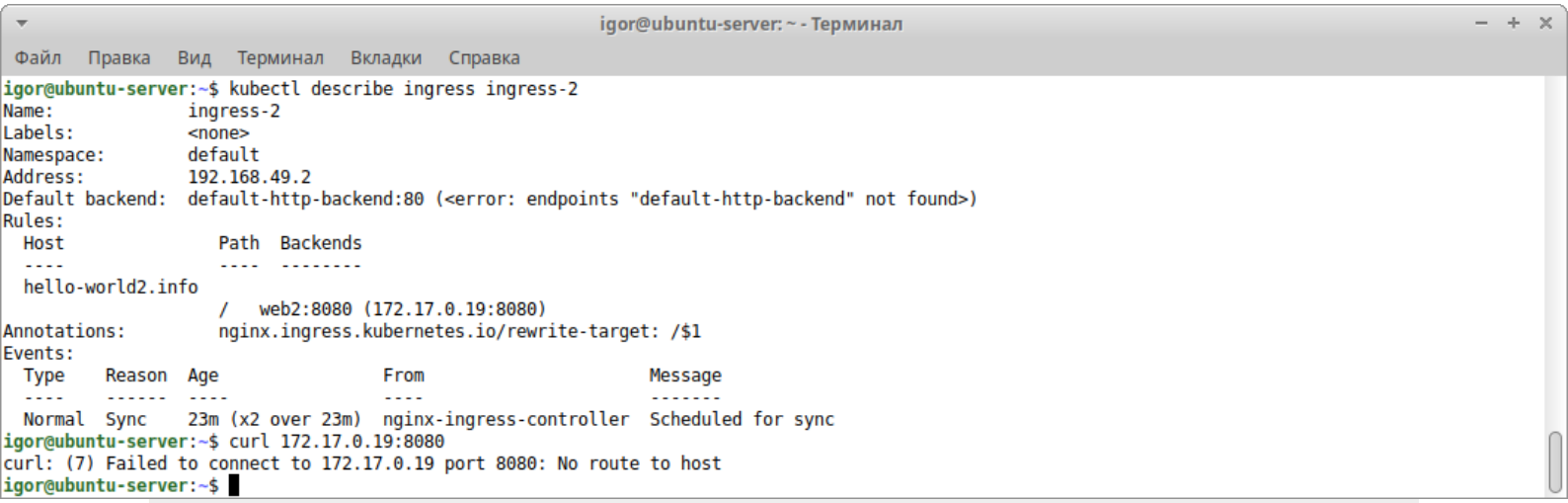
```
vi example-ingress.yaml

kubectl get ingress

kubectl delete ingress example-ingress
```




```
kubectl delete ingress example-ingress

kubectl delete ingress prometheus

kubectl get ingress
```




```
kubectl describe ingress ingress-2

curl 172.17.0.19:8080

curl: (7) Failed to connect to 172.17.0.19 port 8080: No route to
host
```


igor@ubuntu-server: ~ - Терминал
Файл Правка Вид Терминал Вкладки Справка
igor@ubuntu-server:~$ kubectl describe ingress ingress-2
Name: ingress-2
Labels: <none>
Namespace: default
Address: 192.168.49.2
Default backend: default-http-backend:80 (<error: endpoints "default-http-backend" not found>)
Rules:
  Host   Path Backends
  ----   ---- --------
  hello-world2.info
                     /   web2:8080 (172.17.0.19:8080)
Annotations: nginx.ingress.kubernetes.io/rewrite-target: /$1
Events:
  Type   Reason Age   From   Message
  ----   ------ ----   ----   -------
  Normal Sync   23m (x2 over 23m) nginx-ingress-controller Scheduled for sync
igor@ubuntu-server:~$ curl 172.17.0.19:8080
curl: (7) Failed to connect to 172.17.0.19 port 8080: No route to host
igor@ubuntu-server:~$