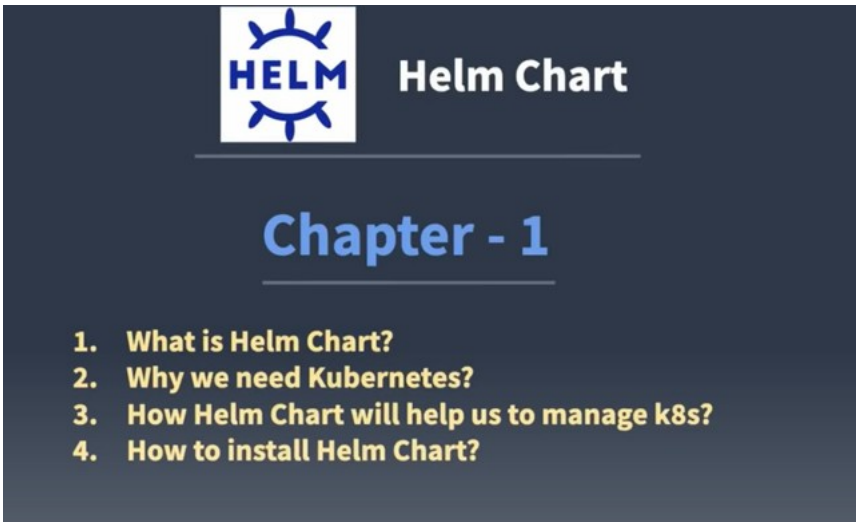
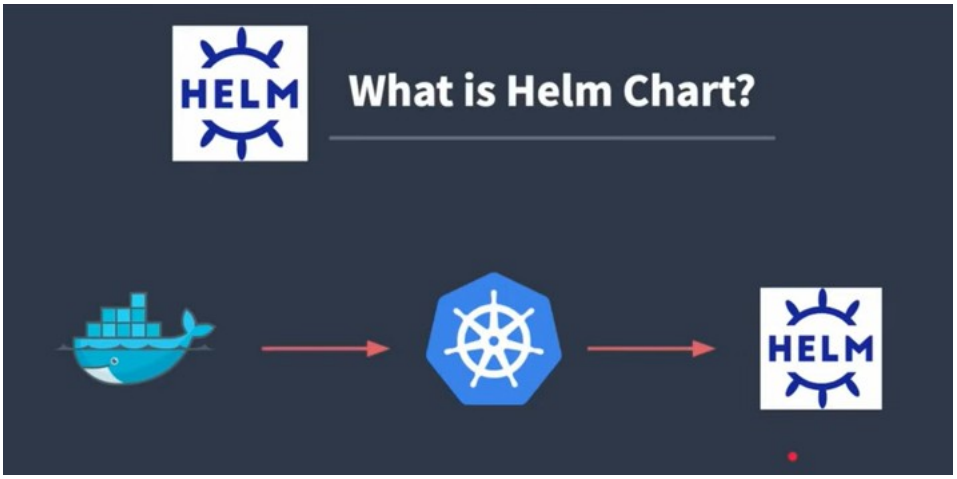
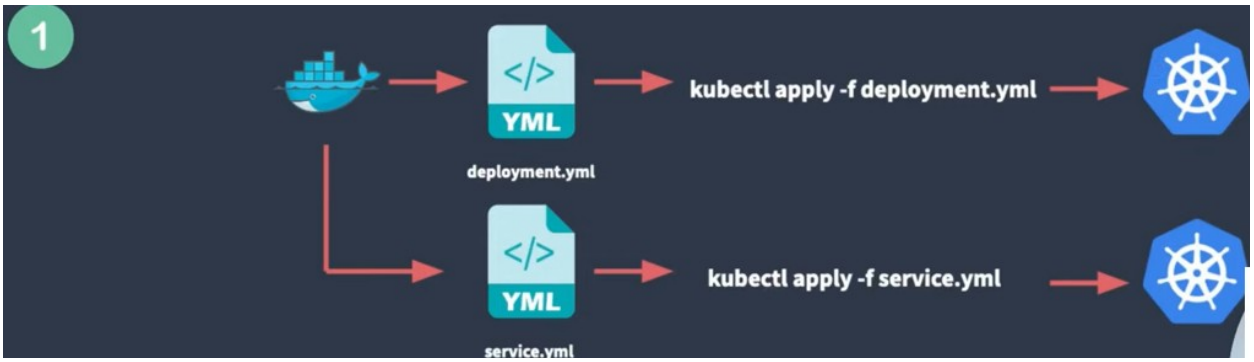
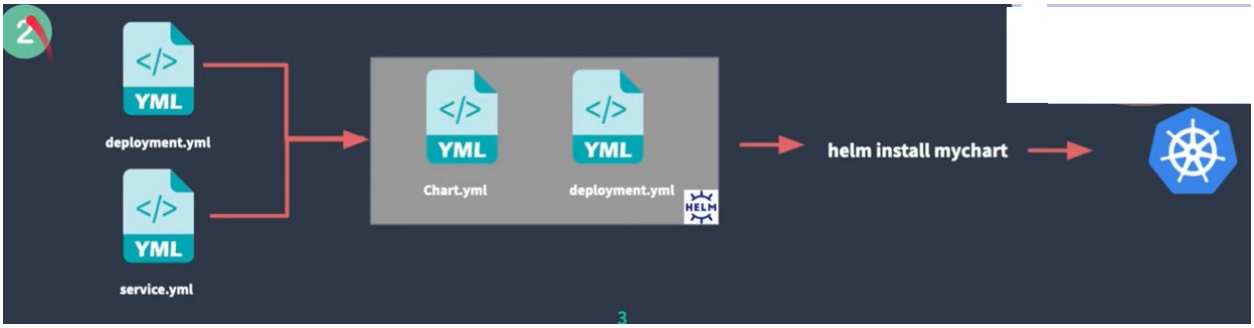


1. Chapter 1 - **Introduction to Helm Chart & Kubernetes**
2. **What is Helm Chart?**
3. **Why is Kubernetes essential?**
4. **The role of Helm Chart in managing K8s Step-by-step guide to installing Helm Chart**







A Helm chart is used to simplify the deployment and management of applications on a Kubernetes cluster by packaging all the necessary resources (like deployments, services, and configurations) into a single, reusable unit, allowing for easier installation, upgrades, and sharing across different environments, significantly reducing manual configuration and potential errors in complex Kubernetes deployments.

Key benefits of using Helm charts:

Simplified deployment:

Instead of writing separate YAML files for each application component, a Helm chart bundles everything together, enabling deployment with a single command.

Versioning and reusability:

Charts can be versioned and shared across teams, allowing consistent deployments across different environments.

Configuration management:

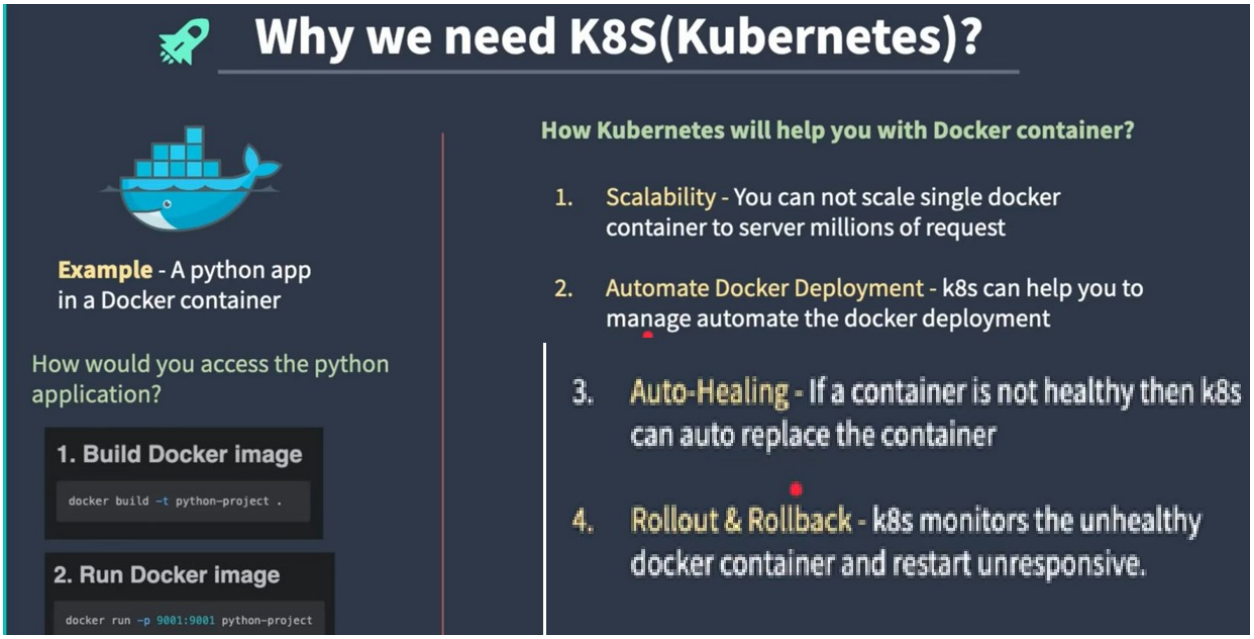
Use "values files" to customize configurations for different environments without modifying the chart itself.

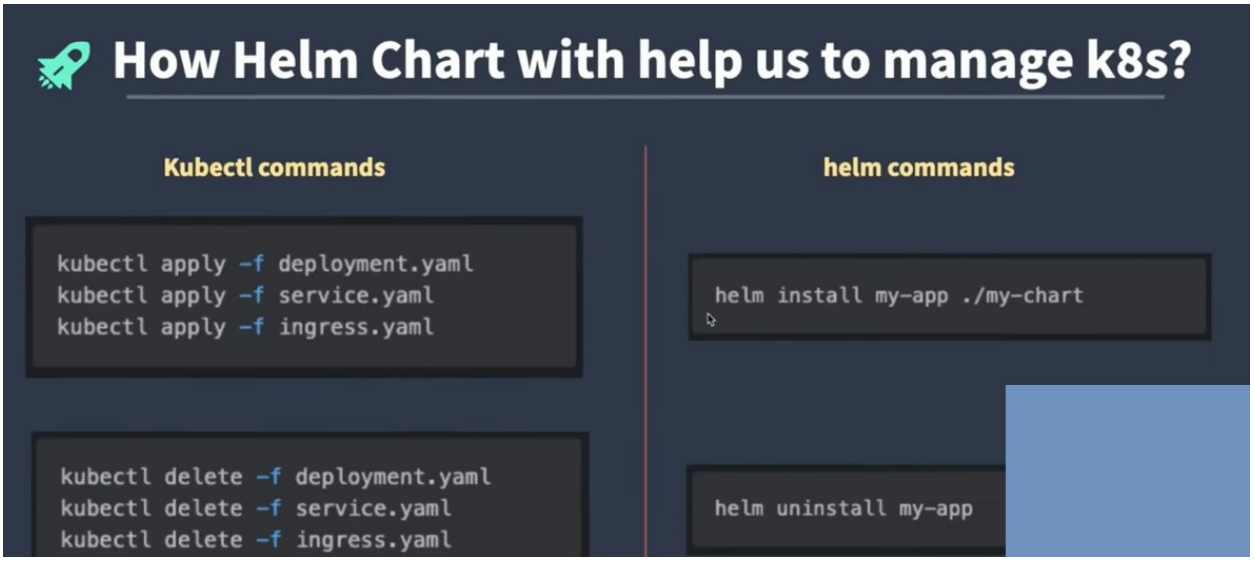
Easy upgrades and rollbacks:

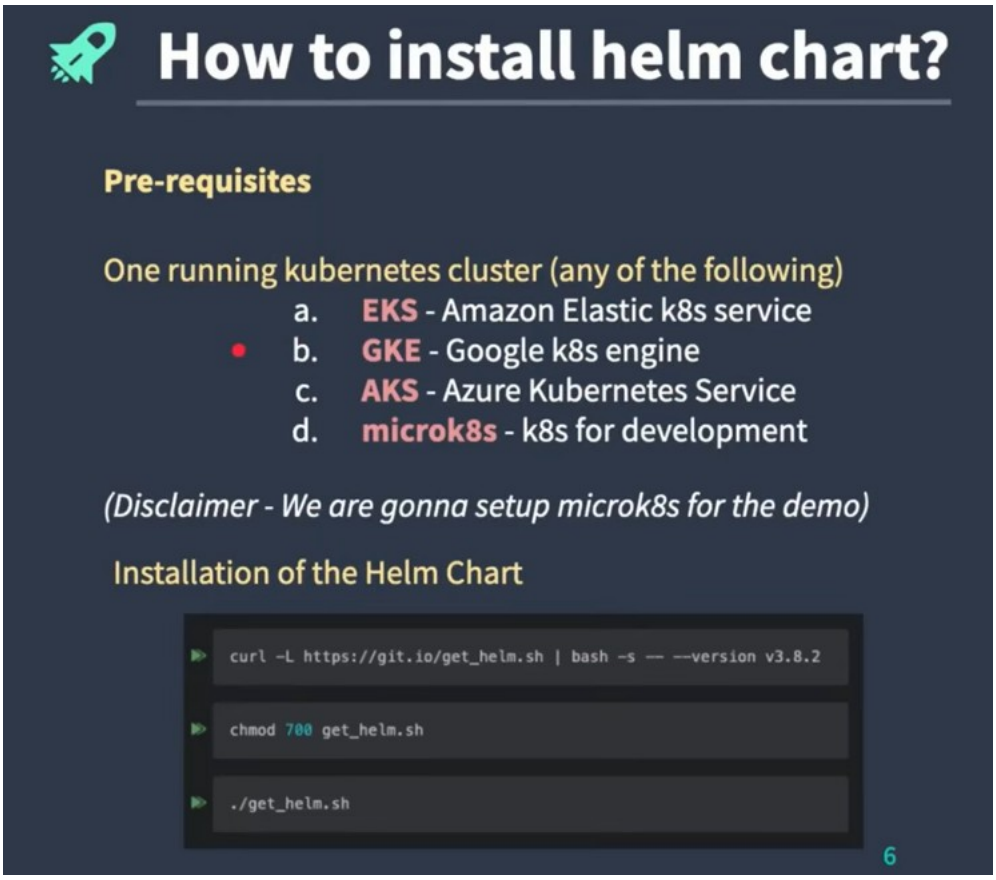
Helm simplifies upgrading applications by managing changes between chart versions, and allows for easy rollbacks if needed.

Reduced complexity:

Helm abstracts away the intricacies of Kubernetes YAML, making application management more accessible to developers and operations teams.



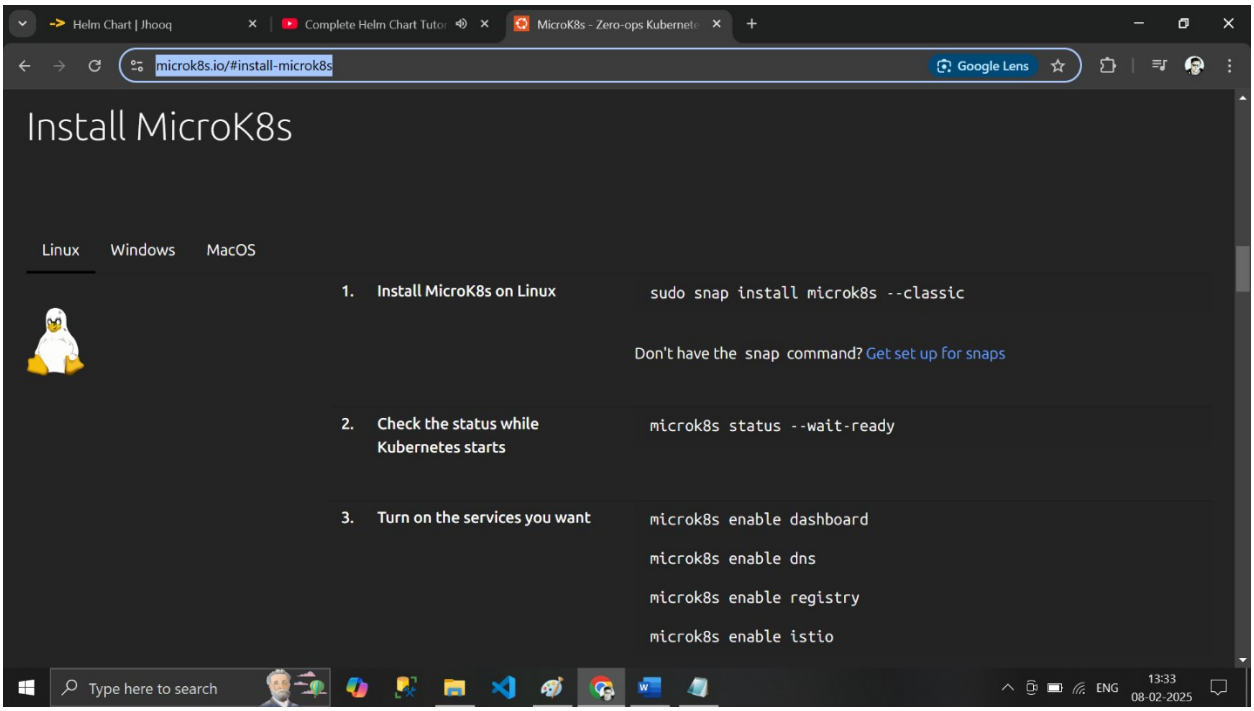


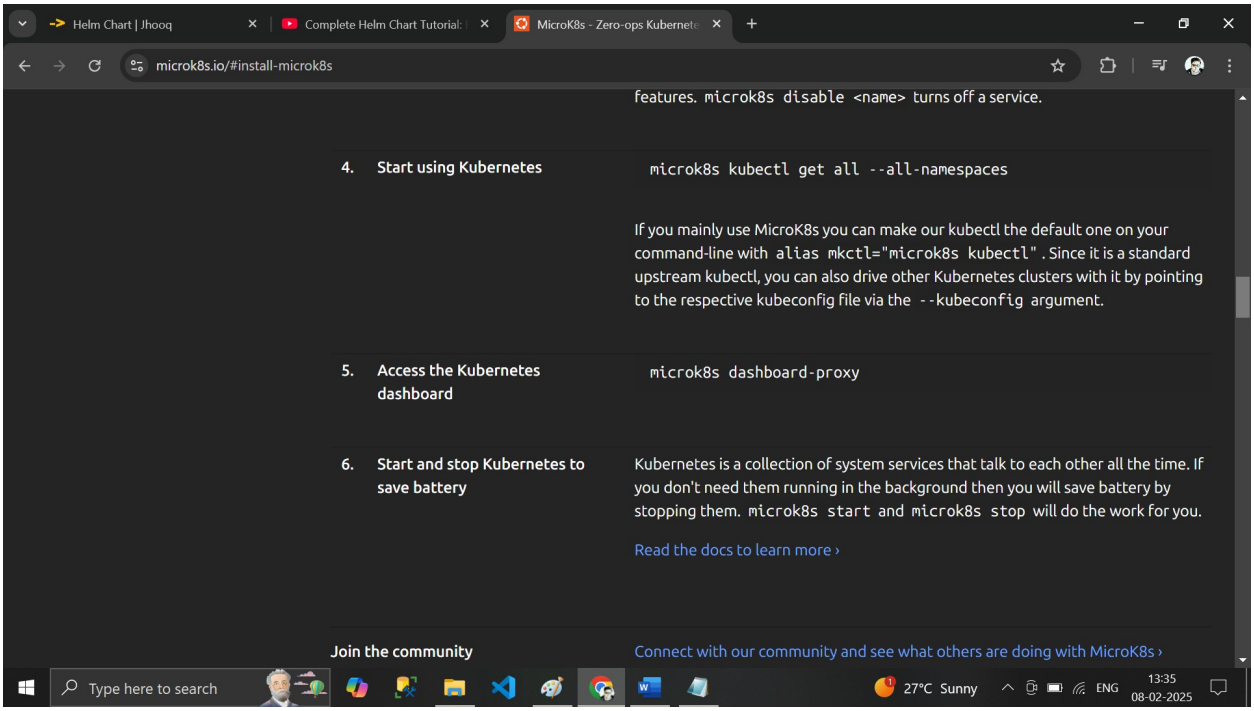


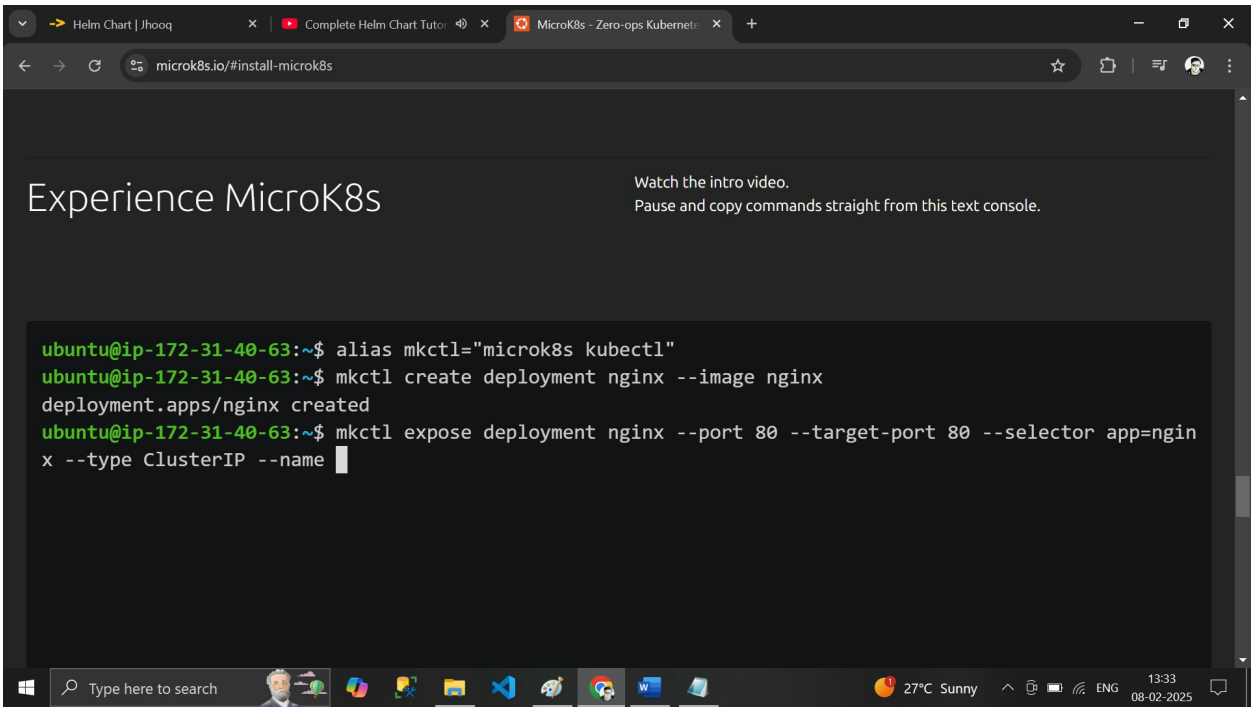
Links - Microk8s - https://microk8s.io/

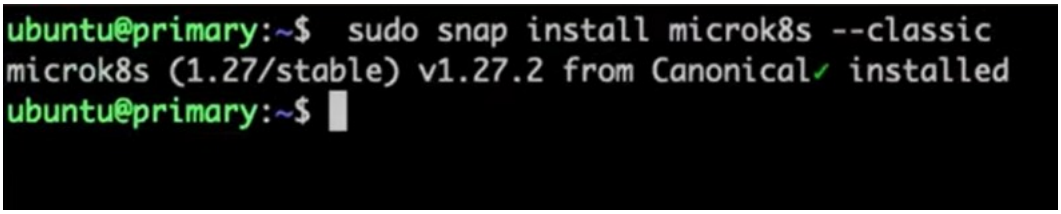
Install kubectl - https://kubernetes.io/docs/tasks/tool...

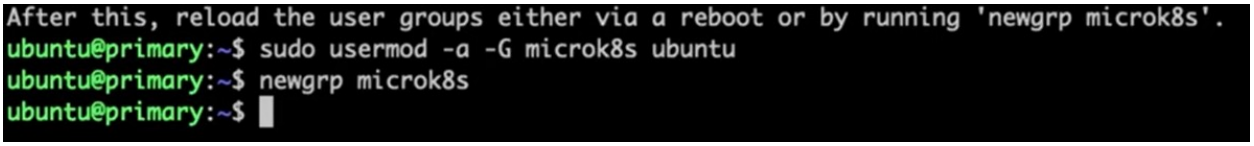
Install Microk8s -with Kubectl - https://microk8s.io/docs/working-with-kubectl


Helm Chart | Jhooq
Complete Helm Chart Tutor
MicroK8s - Zero-ops Kubernete
microk8s.io/#install-microk8s
Google Lens
Install MicroK8s
Linux
Windows
MacOS
1. Install MicroK8s on Linux
sudo snap install microk8s --classic
Don't have the snap command? Get set up for snaps
2. Check the status while Kubernetes starts
microk8s status --wait-ready
3. Turn on the services you want
microk8s enable dashboard
microk8s enable dns
microk8s enable registry
microk8s enable istio
Type here to search
ENG
13:33
08-02-2025

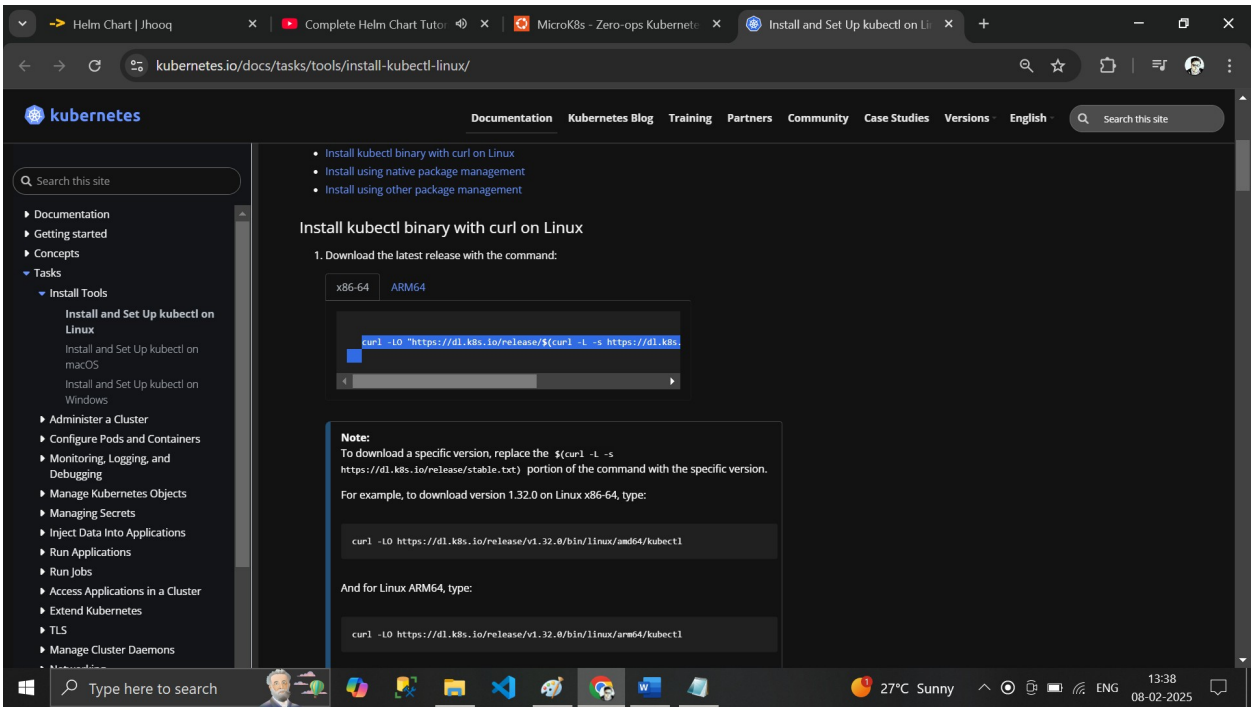

Helm Chart | Jhooq
Complete Helm Chart Tutorial:
MicroK8s - Zero-ops Kubernete
microk8s.io/#install-microk8s
features. microk8s disable <name> turns off a service.
4. Start using Kubernetes
microk8s kubectl get all --all-namespaces
If you mainly use MicroK8s you can make our kubectl the default one on your command-line with alias mkctl="microk8s kubectl" . Since it is a standard upstream kubectl, you can also drive other Kubernetes clusters with it by pointing to the respective kubeconfig file via the --kubeconfig argument.
5. Access the Kubernetes dashboard
microk8s dashboard-proxy
6. Start and stop Kubernetes to save battery
Kubernetes is a collection of system services that talk to each other all the time. If you don't need them running in the background then you will save battery by stopping them. microk8s start and microk8s stop will do the work for you.
Read the docs to learn more ›
Join the community
Connect with our community and see what others are doing with MicroK8s ›
Type here to search
27°C Sunny
ENG
13:35
08-02-2025







https://kubernetes.io/docs/tasks/tools/install-kubectl-linux/




```
ubuntu@primary:~$ curl -LO https://dl.k8s.io/release/v1.23.6/bin/linux/arm64/kubectl
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   138  100   138    0     0    536      0 --:--:-- --:--:-- --:--:--   536
100 44.0M  100 44.0M    0     0  9765k      0  0:00:04  0:00:04 --:--:-- 11.5M
ubuntu@primary:~$ ls
Home  kubectl  snap
ubuntu@primary:~$ sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl
ubuntu@primary:~$
ubuntu@primary:~$
ubuntu@primary:~$ kubectl version
Client Version: version.Info{Major:"1", Minor:"23", GitVersion:"v1.23.6", GitCommit:"ad3338546da947756e8a88
:"clean", BuildDate:"2022-04-14T08:49:13Z", GoVersion:"go1.17.9", Compiler:"gc", Platform:"linux/arm64"}
The connection to the server localhost:8080 was refused - did you specify the right host or port?
ubuntu@primary:~$
```

Install Microk8s -with Kubectl - https://microk8s.io/docs/working-with-kubectl

 # shakehand with kubectl + microk8s

If you have not already configured kubectl on the host, you can just open a terminal and generate the required config :

```
cd $HOME
mkdir .kube
cd .kube
microk8s config > config
```

```
ubuntu@primary:~$ cd $HOME
ubuntu@primary:~$ mkdir .kube
ubuntu@primary:~$
ubuntu@primary:~$ ls -lart
total 45164
-rw-r--r-- 1 ubuntu ubuntu        807 Jan  6  2022 .profile
-rw-r--r-- 1 ubuntu ubuntu       3771 Jan  6  2022 .bashrc
-rw-r--r-- 1 ubuntu ubuntu        220 Jan  6  2022 .bash_logout
drwxr-x--- 1 ubuntu ubuntu       1824 Jul  8 18:17 Home
drwxr-xr-x 3 root   root         4096 Jul  8 18:23 ..
drwx------ 2 ubuntu ubuntu       4096 Jul  8 18:23 .ssh
drwx------ 2 ubuntu ubuntu       4096 Jul  8 18:23 .cache
-rw-r--r-- 1 ubuntu ubuntu          0 Jul  8 18:23 .sudo_as_admin_successful
-rw------- 1 ubuntu ubuntu          5 Jul  8 18:23 .bash_history
drwx------ 4 ubuntu ubuntu       4096 Jul  8 21:34 snap
-rw-rw-r-- 1 ubuntu microk8s 46202880 Jul  8 21:47 kubectl
drwxrwxr-x 2 ubuntu microk8s     4096 Jul  8 21:51 .kube
drwxr-x--- 7 ubuntu ubuntu       4096 Jul  8 21:51 .
ubuntu@primary:~$
```

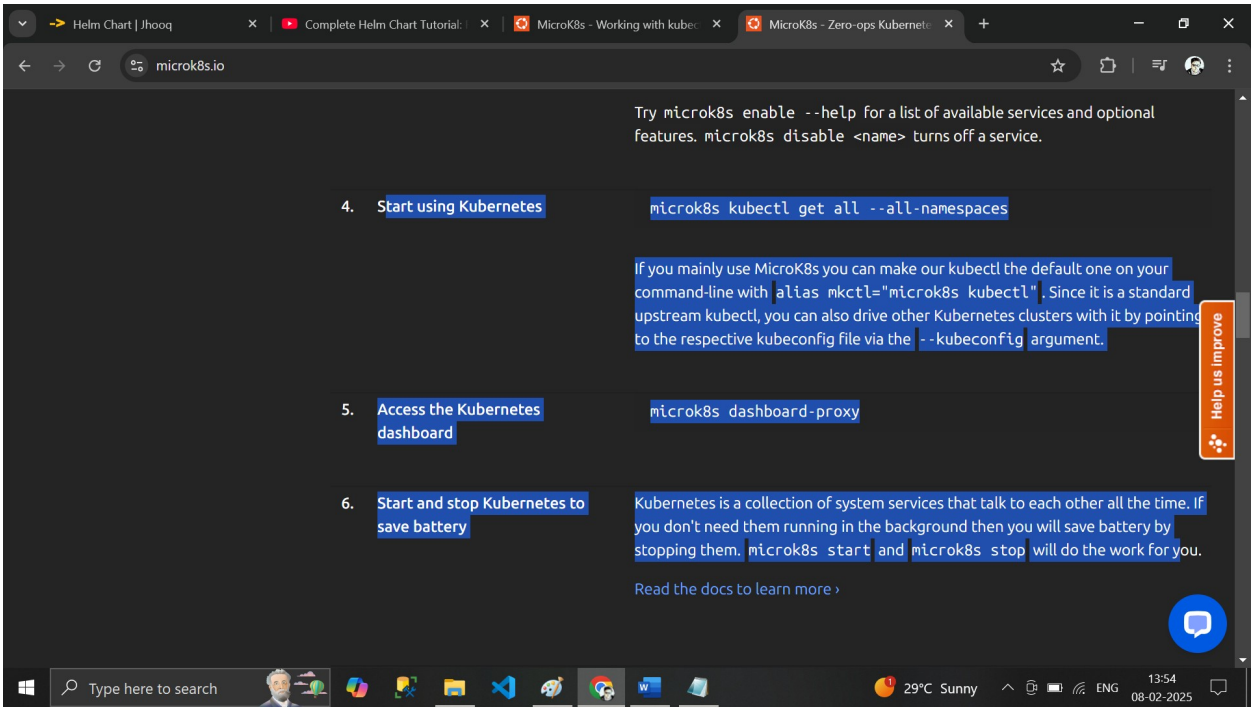
```
ubuntu@primary:~$ cd .kube
ubuntu@primary:~/.kube$ microk8s config > config
ubuntu@primary:~/.kube$
ubuntu@primary:~/.kube$
ubuntu@primary:~/.kube$ ls -lart
total 12
drwxr-x--- 7 ubuntu ubuntu   4096 Jul  8 21:51 ..
drwxrwxr-x 2 ubuntu microk8s 4096 Jul  8 21:51 .
-rw-rw-r-- 1 ubuntu microk8s 1874 Jul  8 21:51 config
ubuntu@primary:~/.kube$ cat con
```

 # certificate

```
clusters:
- cluster:
    certificate-authority-data: LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSUREekNDQWZlZ0F3SUJBZ0lVT2ozTVFtb3JyWGY5VU5TMUJZZUk5QjlhLzNzd0RRWUpL
b1pJaHZjTkFRRUwKQlFBd0Z6RVZNQk1HQTFVRUF3d01NVEF1TVRVeUxqRTRNeTR4TUI0WERUSXpNRGN3T0RFNE16TXlNVm9YRFRNegpNRGN3TlRFNE16TXlNVm93RnpFVk1CTUdBMVVF
QXd3TU1UQXVNVFV5TGpFNE15NHhNSUlCSWpBTkJna3Foa2lHCjl3MEJBUUVGQUFPQ0FROEFNSUlCQ2dLQ0FRRUExV0NZdVRWeE9IZTE2Q3UvRUVHM1NWMVQ4UjFvQnYvR1UxM0IKQ2ZM
VTA5TE8reUVxWnF1eFBUbXIzRmN3cU9tcW5YV1J0Qi95cDF0bXR4QXNHdFpMby9xajdGbS9lUmVuL2NSVgpMK2V0WStKODNGYnJ5bHJpVmh6M0lxc0xPaHlSZlRZb01BWlh4QnIzaVVU
VVIzL015WnBEZHVIVnNnMmhta1BUClVPck0yNjh2bGNpWDNrUW43eUsvU1R6aFEzcUI0NndrdEhLMDZsaTFQZkh6OW5NVWJjMnhIMGJvaWZneXFCVnMKOTBLWTQwZk14Znc3TFhqRE4z
ZWZKL0VnYTlONnVHaVYxeWVpcXgxbEZhSFZINWU3anV1M0htL3BSRDFQaDNBOQp4Rk0vazdzWkFKNkl2b0hBdWZSMTdWNkNCWG96bXN3RWFFblF0Z2E2NEpDczN4VnJnUUlEQVFBQm8x
TXdVVEFkCkJnTlZIUTRFRmdRVUdCbVRhaFVOdzkyTEFLN3hUVDZ1UXVJVDVic3dId1lEVlIwakJCZ3dGb0FVR0JtVGFoVU4KdzkyTEFLN3hUVDZ1UXVJVDVic3dEd1lEVlIwVEFRSC9C
QVV3QXdFQi96QU5CZ2txaGtpRzl3MEJBUXNGQUFPQwpBUUVBTUtnL1BiSzRjeFN5SzgyVytOM1B3ZmVVNFMreTZOWGluVTlyOEhHcytEOFp2MXBGYkNUc3ZJcEw3a3NiCnR3WFN4Z2J5
SUowM0V4V0xwekl5RHFtOGxRdHZNKzFuNHErME53QTJqRGd3SGRubzBOTFY3dVhlQ0tvWjF2UWsKd3Y4ekg5YlczZXhXa1hmOElJanN2U3QwWEpKTE04MndESTdIMDRSdnpWcVQ5SVFs
azhXVkhrYnB5VmwwOUw0MAp1RGRydnFYNkdUNEZPNVg3cFMyeWpXL2tZK1VrM1RhRExGUU01c3J5aTk4dFFTK25tMGFZMXU0ZnZmcWxUUzNKCkRkR1VaVXI4RFlSTzRIZXp0eTYzZmVU
WU0yS1Q0bXIwNGtkZ1NNaTl1SEI0OFpOVVBKTHlaMzZuV1QwOWlzOHcKTG9GWkR6REl1TFFUVnFFMzEzK2FMYW9UQnc9PQotLS0tLUVORCBDRVJUSUZJQ0FURS0tLS0tCg==
    server: https://192.168.64.5:16443
  name: microk8s-cluster
contexts:
- context:
    cluster: microk8s-cluster
    user: admin
  name: microk8s
current-context: microk8s
kind: Config
preferences: {}
users:
```

>>>>>>>>>>>>>>>>>>>>>>>>>RESTART Your machine<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

https://microk8s.io/



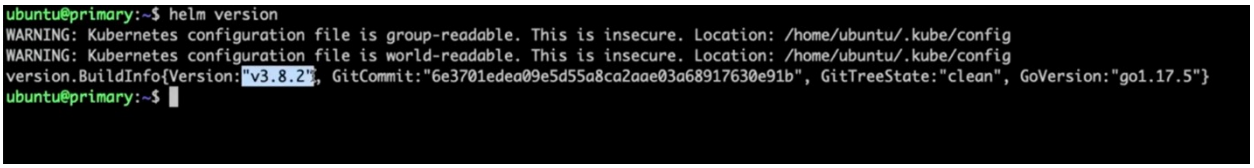
```
ubuntu@primary:~$ kubectl get all --all-namespaces
NAMESPACE     NAME                                       READY   STATUS    RESTARTS        AGE
kube-system   pod/coredns-7745f9f87f-5xp7k               1/1     Running   1 (4m32s ago)   25m
kube-system   pod/calico-kube-controllers-6c99c8747f-wvvzw   1/1     Running   1 (4m32s ago)   25m
kube-system   pod/calico-node-f2dq4                      1/1     Running   1 (4m32s ago)   25m

NAMESPACE     NAME                 TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)                  AGE
default       service/kubernetes   ClusterIP   10.152.183.1    <none>        443/TCP                  25m
kube-system   service/kube-dns     ClusterIP   10.152.183.10   <none>        53/UDP,53/TCP,9153/TCP   25m

NAMESPACE     NAME                         DESIRED   CURRENT   READY   UP-TO-DATE   AVAILABLE   NODE SELECTOR            AGE
kube-system   daemonset.apps/calico-node   1         1         1       1            1           kubernetes.io/os=linux   25m

NAMESPACE     NAME                                       READY   UP-TO-DATE   AVAILABLE   AGE
kube-system   deployment.apps/coredns                    1/1     1            1           25m
kube-system   deployment.apps/calico-kube-controllers    1/1     1            1           25m

NAMESPACE     NAME                                                 DESIRED   CURRENT   READY   AGE
kube-system   replicaset.apps/coredns-7745f9f87f                   1         1         1       25m
kube-system   replicaset.apps/calico-kube-controllers-6c99c8747f   1         1         1       25m
```

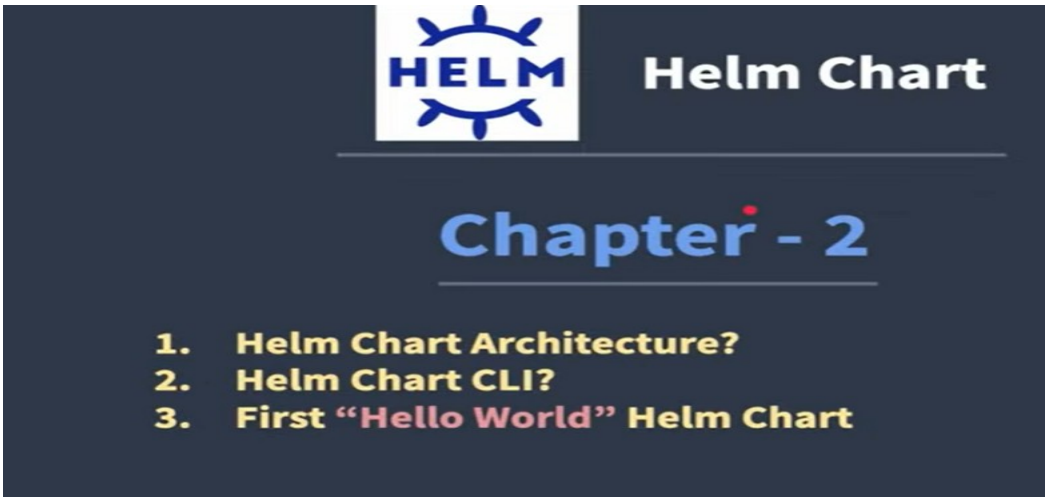
 # helm Chart cmds / installation from Jhooq – Rahul Wagh Sir 😊

https://jhooq.com/getting-start-with-helm-chart/

```
ubuntu@primary:~$ curl -L https://git.io/get_helm.sh | bash -s -- --version v3.8.2
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0
100  6666  100  6666    0     0   8108      0 --:--:-- --:--:-- --:--:-- 95228
Downloading https://get.helm.sh/helm-v3.8.2-linux-arm64.tar.gz
Preparing to install helm and tiller into /usr/local/bin
helm installed into /usr/local/bin/helm
info: tiller binary was not found in this release; skipping tiller installation
Run 'helm init' to configure helm.
```

```
ubuntu@primary:~$ ls -lart
total 45164
-rw-r--r-- 1 ubuntu ubuntu        807 Jan  6  2022 .profile
-rw-r--r-- 1 ubuntu ubuntu       3771 Jan  6  2022 .bashrc
-rw-r--r-- 1 ubuntu ubuntu        220 Jan  6  2022 .bash_logout
drwxr-x--- 1 ubuntu ubuntu       1824 Jul  8 18:17 Home
drwxr-xr-x 3 root   root         4096 Jul  8 18:23 ..
drwx------ 2 ubuntu ubuntu       4096 Jul  8 18:23 .ssh
drwx------ 2 ubuntu ubuntu       4096 Jul  8 18:23 .cache
-rw-r--r-- 1 ubuntu ubuntu          0 Jul  8 18:23 .sudo_as_admin_successful
drwx------ 4 ubuntu ubuntu       4096 Jul  8 21:34 snap
-rw-rw-r-- 1 ubuntu microk8s 46202880 Jul  8 21:47 kubectl
drwxr-x--- 7 ubuntu ubuntu       4096 Jul  8 21:51 .
drwxrwxr-x 3 ubuntu microk8s     4096 Jul  8 21:53 .kube
-rw------- 1 ubuntu ubuntu        639 Jul  8 21:54 .bash_history
ubuntu@primary:~$
```


ubuntu@primary:~$ helm version
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
version.BuildInfo{Version:"v3.8.2", GitCommit:"6e3701edea09e5d55a8ca2aae03a68917630e91b", GitTreeState:"clean", GoVersion:"go1.17.5"}
ubuntu@primary:~$



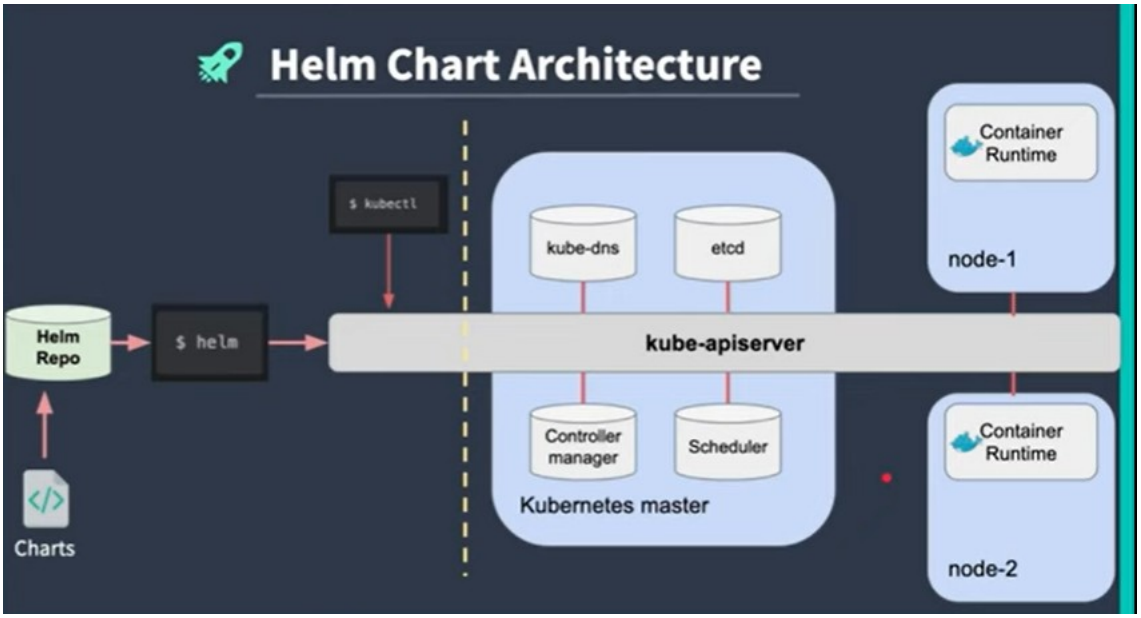
Chapter 2 - Diving Deeper into Helm Chart

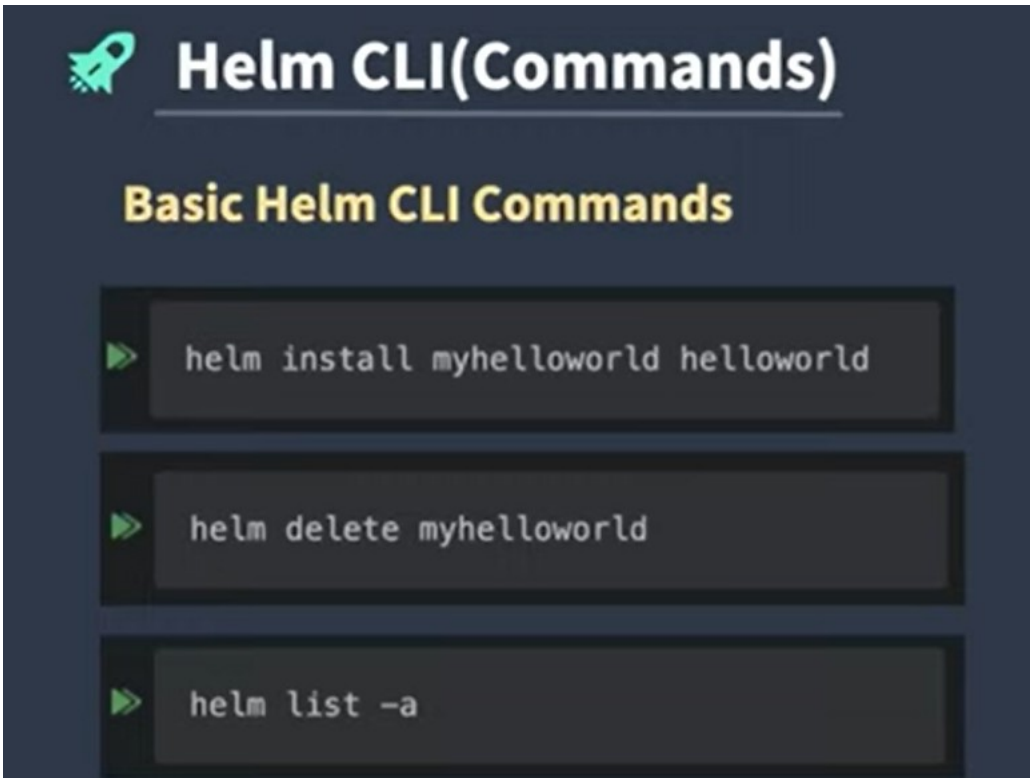
Understand Helm Chart's significance ..The symbiotic relationship between Kubernetes and Helm Chart Streamlining .. K8s management using Helm Chart

complete Practical + notes + cmd execution mentioned below

http://jhooq.com/getting-start-with-helm-chart/

Helm chart Architecture





2. Writing your first Helm Chart for "Hello World"

Now after you have done your Helm Chart installation, we can write our first **"Hello World"** Helm Chart.

To begin with -

2.1: Create your first Helm Chart

We are going to create our first **helloworld** Helm Chart using the following command

BASH

```
1 helm create helloworld
```

It should create a directory `helloworld`, you can verify it by using the following `ls -lart` command

```bash
ls -lart | grep helloworld
```

It should return you with -

```bash
drwxr-xr-x 4 vagrant vagrant     4096 Nov  7 19:57 helloworld
```

To verify the complete directory structure of the HelmChart please do run the command

```bash
tree helloworld
```

```
helloworld
├── charts
├── Chart.yaml
├── templates
│   ├── deployment.yaml
│   ├── _helpers.tpl
│   ├── hpa.yaml
│   ├── ingress.yaml
│   ├── NOTES.txt
│   ├── serviceaccount.yaml
│   ├── service.yaml
│   └── tests
│       └── test-connection.yaml
└── values.yaml
```

Great now you created your first Helm Chart - **helloworld**.

In the next steps we are going to run the **helloworld** Helm Chart

2.2: Update the service.type from `ClusterIP` to `NodePort` inside the values.yml

Before you run your **helloworld** Helm Chart we need to update the `service.type` from `ClusterIP` to `NodePort` .

The reason for this change is - After installing/running the **helloworld** Helm Chart we should be able to access the service outside of the kubernetes cluster. And if you do not change the `service.type` then you will only be able to access the service withing kubernetes cluster.

To update the `values.yml`, first go inside the directory `helloworld`

```bash
cd helloworld
```

2.2.1: Open `values.yml` in `vi`

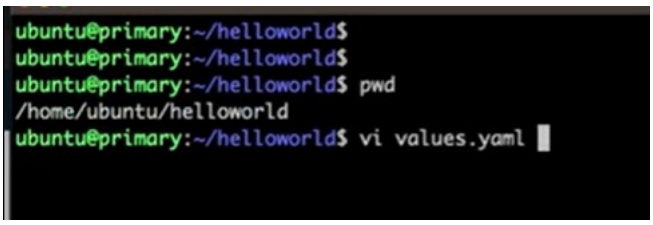
After that open the `values.yml` in `vi`

```
vi values.yaml
```

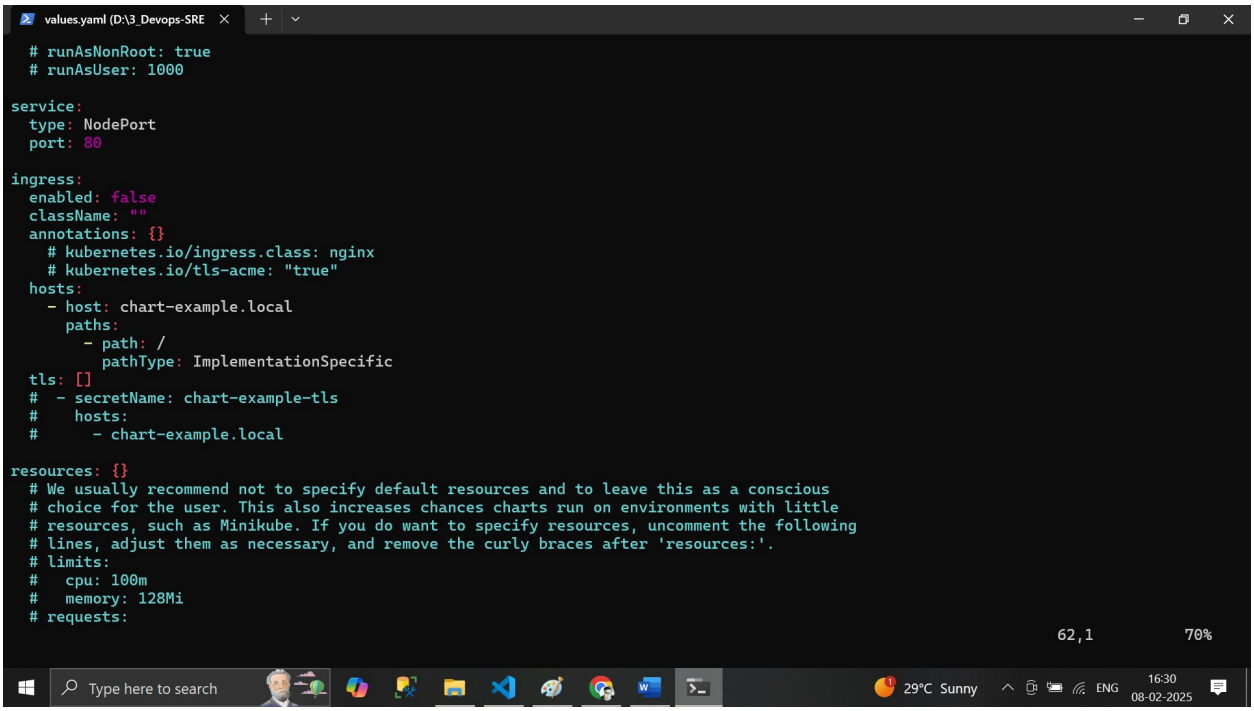
2.2.2: Update `service.type` from `ClusterIP` to `NodePort`

Look for the `service.type` block and update its value to `NodePort`

```
service:
  type: NodePort
```



change the service to NodePort ..



2.3: Install the Helm Chart using command - `helm install`

Now after updating the `values.yml`, you can install the Helm Chart.

Note : The `helm install` command take two arguments -

1. First argument - Release name that you pick
2. Second argument - Chart you want to install

It should look like -

```
helm install <FIRST_ARGUMENT_RELEASE_NAME> <SECOND_ARGUMENT_CHART_NAME>
```

```
helm install myhelloworld helloworld
```

After running the above command it should return you with -

```
NAME: myhellworld
LAST DEPLOYED: Sat Nov  7 21:48:08 2020
NAMESPACE: default
STATUS: deployed
REVISION: 1
NOTES:
1. Get the application URL by running these commands:
  export NODE_PORT=$(kubectl get --namespace default -o jsonpath="{.spec.ports[0].nodePort}" ser
  export NODE_IP=$(kubectl get nodes --namespace default -o jsonpath="{.items[0].status.addresse
  echo http://$NODE_IP:$NODE_PORT
```

2.4: Verify the helm install command

Now you need to verify your helm release .i.e. `myhelloworld` and which can be done by running the `helm list` command.

```
helm list -a
```

It should return you withe release name which you have just installed .i.e. `myhelloworld`

```
NAME          NAMESPACE       REVISION        UPDATED                                 STATUS
myhelloworld  default         1               2020-11-07 21:48:08.8550677 +0000 UTC   deployed
```

2.5: Get kubernetes Service details and port

Lets run the `kubectl get service` command to get the `NodePort`.

```
kubectl get service
```

And the above command should return you -

```
NAME                      TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)        AGE
kubernetes                ClusterIP   10.233.0.1      <none>        443/TCP        14d
myhellworld-helloworld    NodePort    10.233.14.134   <none>        80:30738/TCP   7m10s
```

Note: Keep in mind the `NodePort` number can vary in the range **30000-32767**, so you might get different `NodePort`.

Since my **cluster ip** is `100.0.0.2` and **NodePort** is `30738`, so I can access my Nginx page of my `myhelloworld` Helm Chart

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

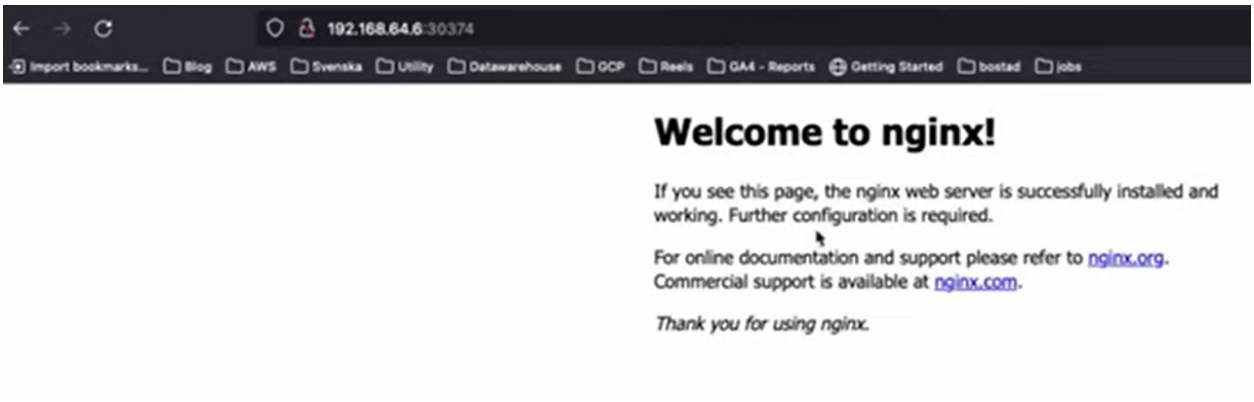
For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

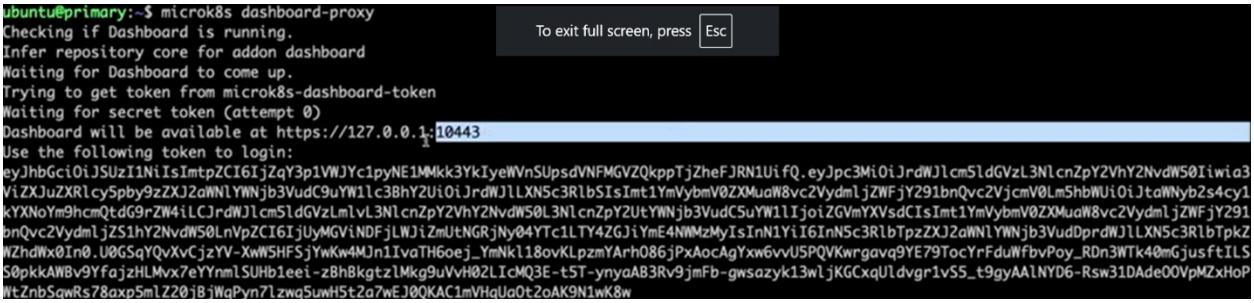
```
ubuntu@primary:~/helloworld$ pwd
/home/ubuntu/helloworld
ubuntu@primary:~/helloworld$ cd ..
ubuntu@primary:~$
ubuntu@primary:~$
ubuntu@primary:~$ pwd
/home/ubuntu
ubuntu@primary:~$ ls
Home  helloworld  kubectl  snap
ubuntu@primary:~$
```

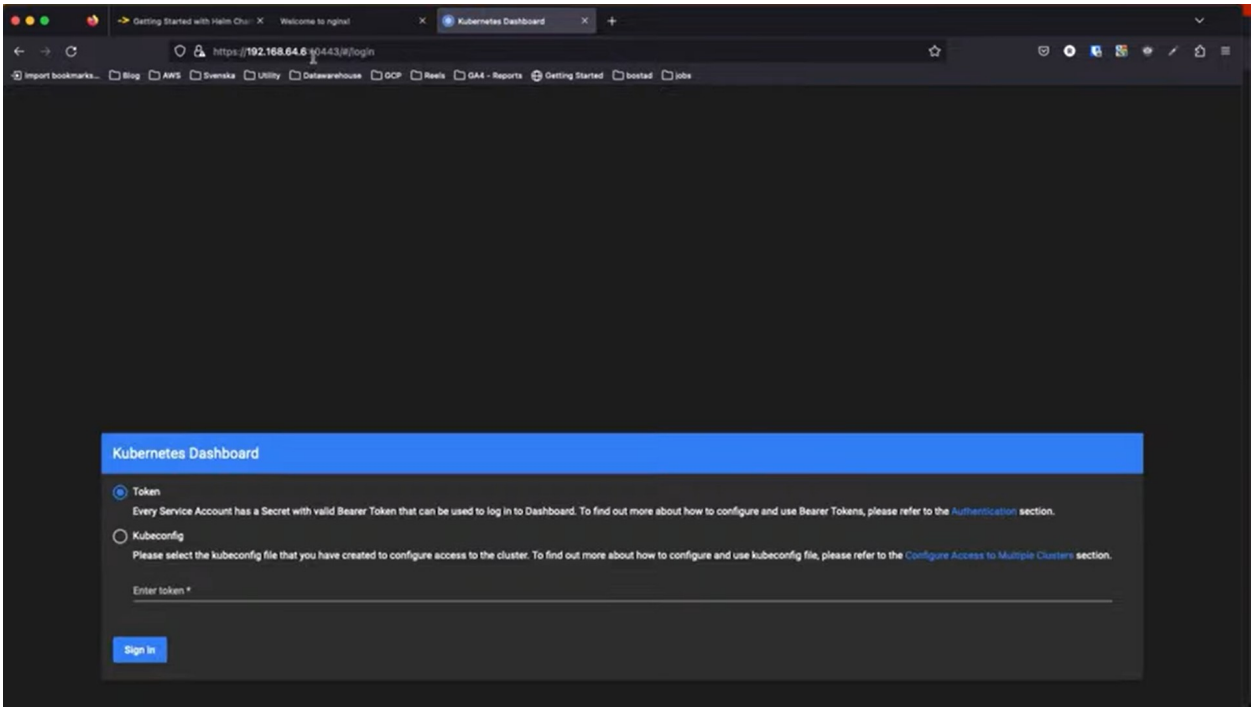
```
ubuntu@primary:~$ helm install myhelloworld helloworld
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
NAME: myhelloworld
LAST DEPLOYED: Sun Jul  9 16:33:35 2023
NAMESPACE: default
STATUS: deployed
REVISION: 1
NOTES:
1. Get the application URL by running these commands:
  export NODE_PORT=$(kubectl get --namespace default -o jsonpath="{.spec.ports[0].nodePort}" services myhelloworld)
  export NODE_IP=$(kubectl get nodes --namespace default -o jsonpath="{.items[0].status.addresses[0].address}")
  echo http://$NODE_IP:$NODE_PORT
ubuntu@primary:~$
ubuntu@primary:~$
ubuntu@primary:~$ helm list -a
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
NAME            NAMESPACE       REVISION        UPDATED                                 STATUS          CHART                   APP V
ERSION
myhelloworld    default         1               2023-07-09 16:33:35.294021765 +0300 EEST        deployed        helloworld-0.1.0        1.16.
0
ubuntu@primary:~$
```

```
ubuntu@primary:~$ kubectl get service
NAME           TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)        AGE
kubernetes     ClusterIP   10.152.183.1    <none>        443/TCP        97m
myhelloworld   NodePort    10.152.183.18   <none>        80:30374/TCP   4m7s
ubuntu@primary:~$ hostname -I
192.168.64.6 10.1.226.128 fd16:8590:7284:4e59:5054:ff:fe7f:b17d
ubuntu@primary:~$
```



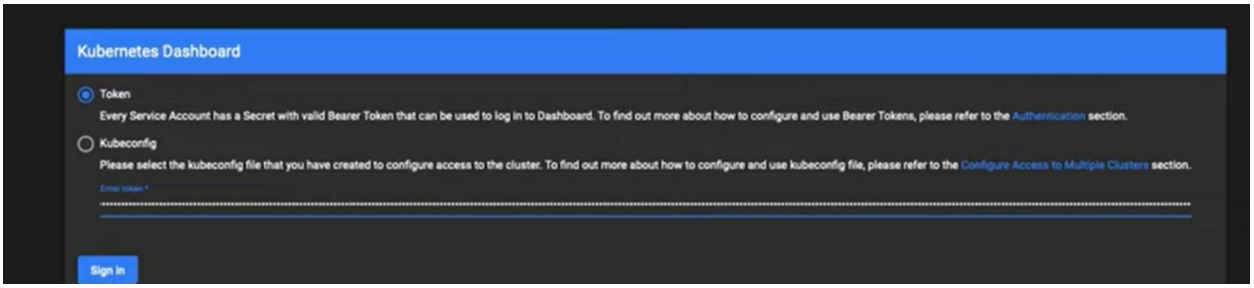
Kubernetes -dashboard



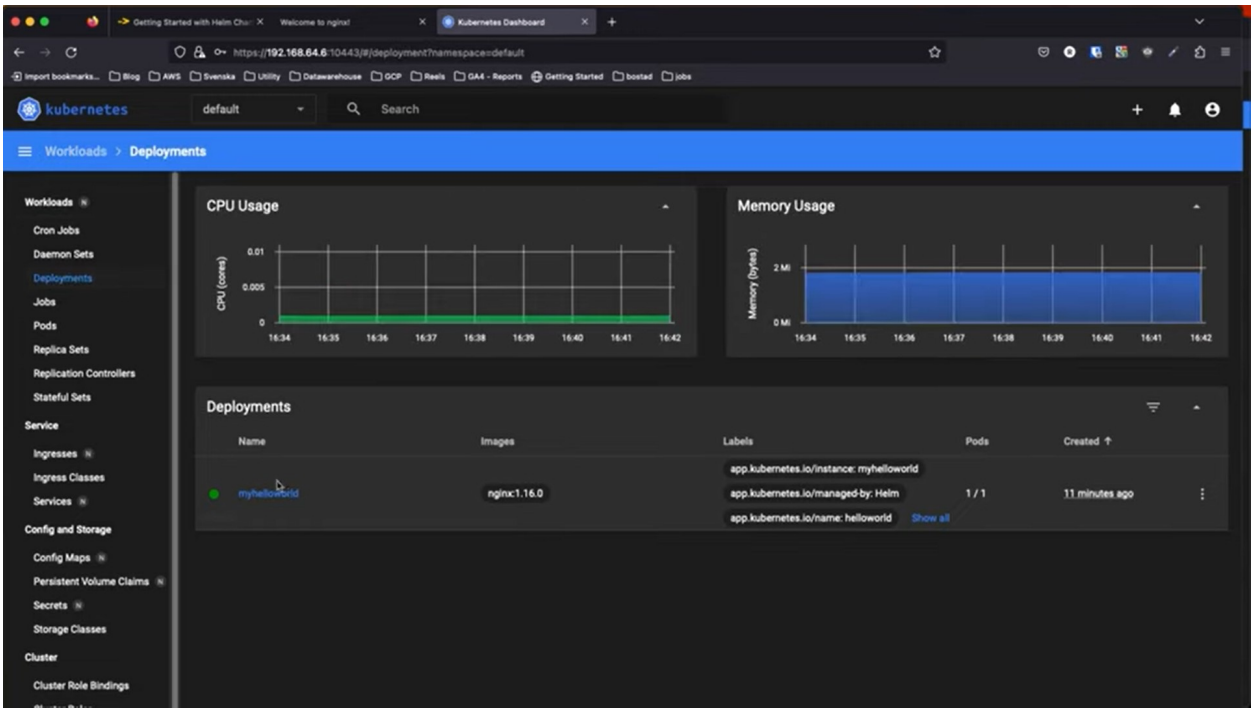


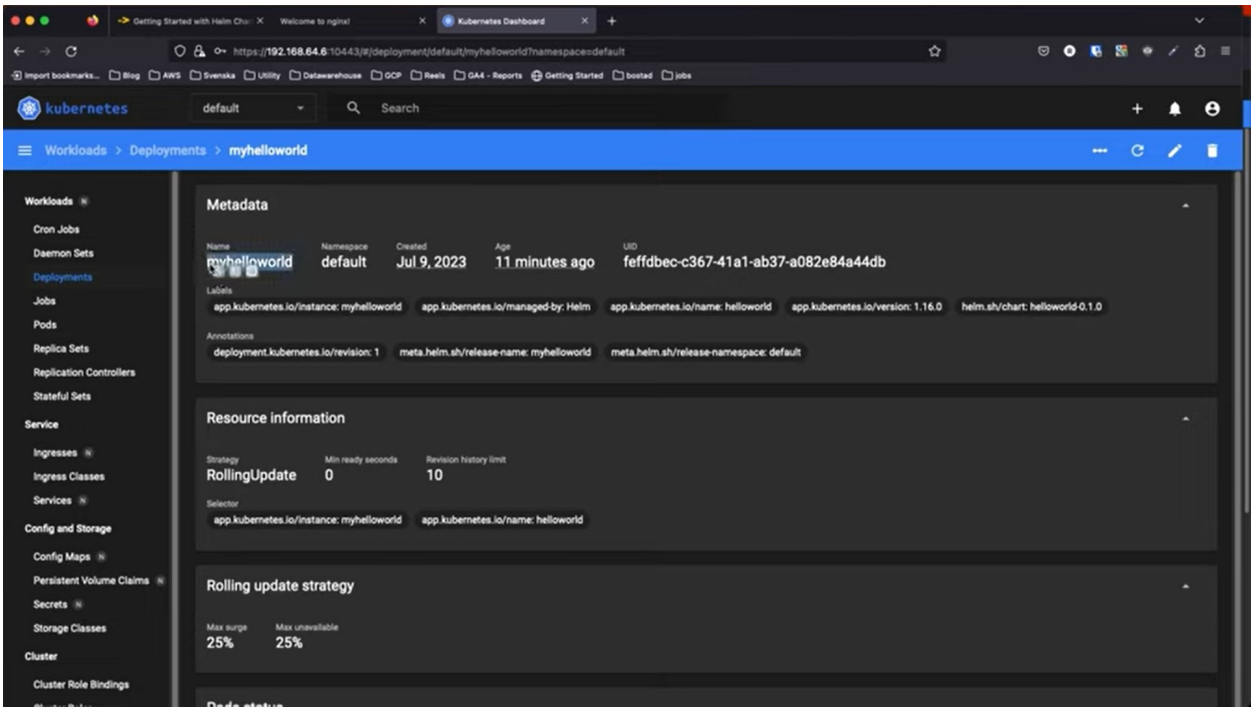
cp the token + Sign in

```
ubuntu@primary:~$ microk8s dashboard-proxy
Checking if Dashboard is running.
Infer repository core for addon dashboard
Waiting for Dashboard to come up.
Trying to get token from microk8s-dashboard-token
Waiting for secret token (attempt 0)
Dashboard will be available at https://127.0.0.1:10443
Use the following token to login:
eyJhbGciOiJSUzI1NiIsImtpZCI6IjZqY3p1VWJYc1pyNE1MMkk3YkIyeWVnSUpsdVNFMGVZQkppTjZheFJRN1UifQ.eyJpc3MiOiJrdWJlcm5ldGVzL3NlcnZpY2VhY2NvdW50Iiwia3
ViZXJuZXRlcy5pby9zZXJ2aWNlYWNjb3VudC9uYW1lc3BhY2UiOiJrdWJlLXN5c3RlbSIsImt1YmVybmV0ZXMuaW8vc2VydmljZWFjY291bnQvc2VjcmV0Lm5hbWUiOiJtaWNyb2s4cy1
kYXNoYm9hcmQtdG9rZW4iLCJrdWJlcm5ldGVzLmlvL3NlcnZpY2VhY2NvdW50L3NlcnZpY2UtYWNjb3VudC5uYW1lIjoiZGVmYXVsdCIsImt1YmVybmV0ZXMuaW8vc2VydmljZWFjY291
bnQvc2VydmljZS1hY2NvdW50LnVpZCI6IjUyMGViNDFjLWJiZmUtNGRjNy04YTc1LTY4ZGJiYmE4NWMzMyIsInN1YiI6InN5c3RlbTpzZXJ2aWNlYWNjb3VudDprdWJlLXN5c3RlbTpk
WZhdWx0In0.U0GSqYQvXvCjzYV-XwW5HFSjYwKw4MJn1IvaTH6oej_YmNkl18ovKLpzmYArhO86jPxAocAgYxw6vvU5PQVKwrgavq9YE79TocYrFduWfbvPoy_RDn3WTk40mGjusftILS
S0pkkAWBv9YfajzHLMvx7eYYnmlSUHb1eei-zBhBkgtzlMkg9uVvH02LIcMQ3E-t5T-ynyaAB3Rv9jmFb-gwsazyk13wljKGCxqUldvgr1vS5_t9gyAAlNYD6-Rsw31DAdeOOVpMZxHoP
WtZnbSqwRs78axp5mlZ20jBjWqPyn7lzwq5uwH5t2a7wEJ0QKAC1mVHqUaOt2oAK9N1wKGw
```



deployment



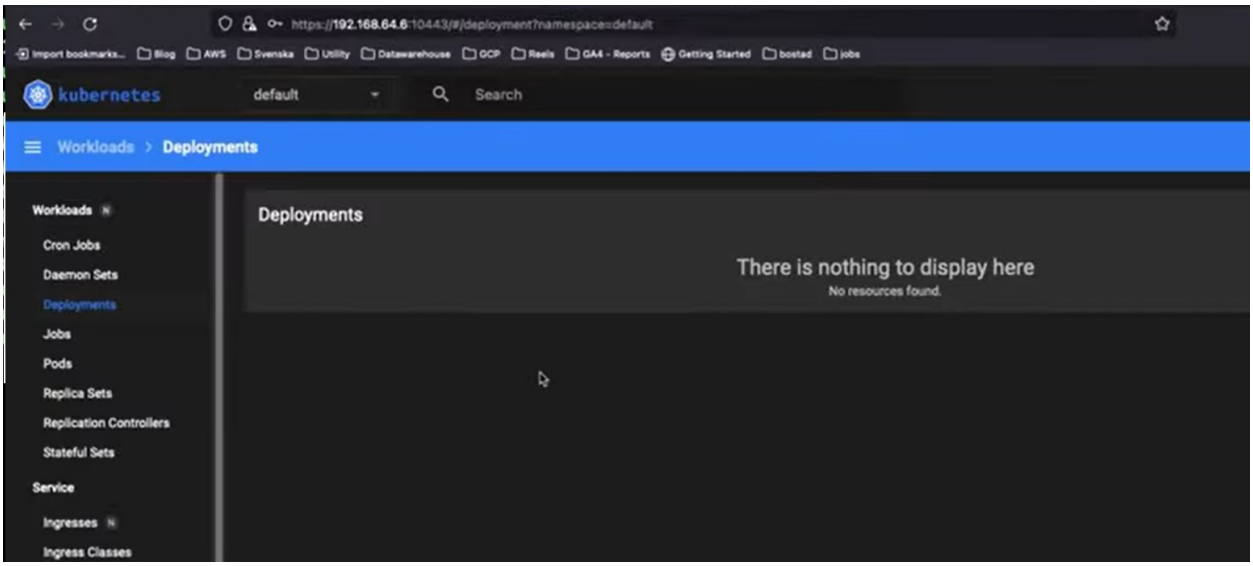



```
ubuntu@primary:~$ helm list -a
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
NAME            NAMESPACE       REVISION        UPDATED                                 STATUS          CHART                   APP V
ERSION
myhelloworld    default         1               2023-07-09 16:33:35.294021765 +0300 EEST        deployed        helloworld-0.1.0        1.16.
0
ubuntu@primary:~$
ubuntu@primary:~$
ubuntu@primary:~$
```

uninstall

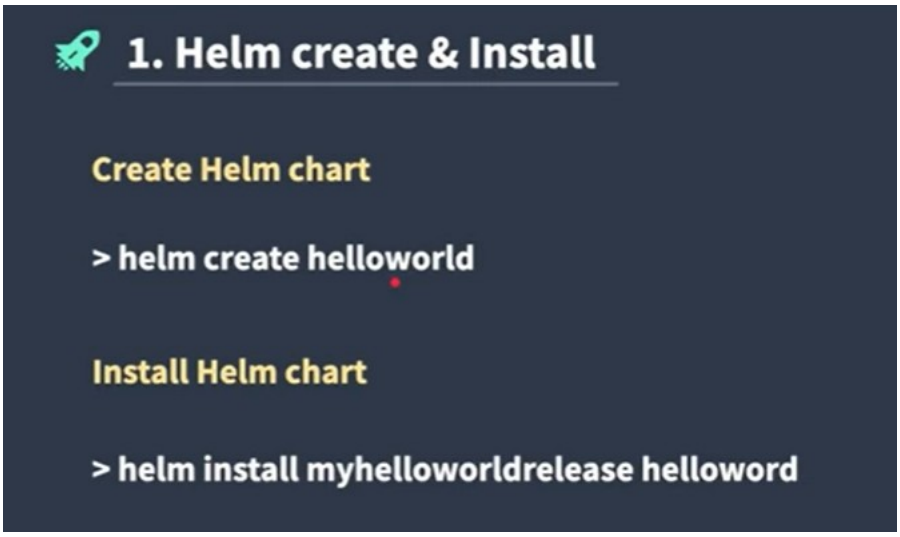
```
ubuntu@primary:~$ helm uninstall myhelloworld
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
release "myhelloworld" uninstalled
ubuntu@primary:~$
```

go to K8s dashboard



Chapter 3 - Helm CLI Commands Mastery:

1. helm create
2. helm install
3. helm upgrade
4. helm rollback
5. Mastering debugging with helm --debug --dry-run
6. helm template, helm lint, helm uninstall and more!




```
ubuntu@primary:~$ helm create helloworld
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
Creating helloworld
ubuntu@primary:~$ ls -lart
total 56
-rw-r--r-- 1 ubuntu ubuntu  807 Jan  6  2022 .profile
-rw-r--r-- 1 ubuntu ubuntu 3771 Jan  6  2022 .bashrc
-rw-r--r-- 1 ubuntu ubuntu  220 Jan  6  2022 .bash_logout
drwxr-xr-x 3 root   root   4096 Jul  9 14:15 ..
drwx------ 2 ubuntu ubuntu 4096 Jul  9 14:15 .ssh
drwx------ 2 ubuntu ubuntu 4096 Jul  9 14:15 .cache
-rw-r--r-- 1 ubuntu ubuntu    0 Jul  9 14:15 .sudo_as_admin_successful
drwxr-xr-x 1 ubuntu ubuntu   96 Jul  9 14:55 common-work-directory
drwx------ 4 ubuntu ubuntu 4096 Jul  9 15:00 snap
drwxrwxr-x 3 ubuntu ubuntu 4096 Jul  9 15:01 .kube
-rw------- 1 ubuntu ubuntu 2303 Jul  9 16:32 .viminfo
-rw------- 1 ubuntu ubuntu 1920 Jul 11 20:10 .bash_history
drwxr-x--- 1 ubuntu ubuntu 1760 Jul 11 21:06 Home
drwxr-xr-x 4 ubuntu ubuntu 4096 Jul 11 21:25 helloworld
drwxr-x--- 9 ubuntu ubuntu 4096 Jul 11 21:25 .
ubuntu@primary:~$ ls -lart helloworld/
```

```
ubuntu@primary:~$ ls -lart helloworld/
total 28
-rw-r--r-- 1 ubuntu ubuntu 1877 Jul 11 21:25 values.yaml
drwxr-xr-x 3 ubuntu ubuntu 4096 Jul 11 21:25 templates
drwxr-xr-x 2 ubuntu ubuntu 4096 Jul 11 21:25 charts
-rw-r--r-- 1 ubuntu ubuntu 1146 Jul 11 21:25 Chart.yaml
-rw-r--r-- 1 ubuntu ubuntu  349 Jul 11 21:25 .helmignore
drwxr-x--- 9 ubuntu ubuntu 4096 Jul 11 21:25 ..
drwxr-xr-x 4 ubuntu ubuntu 4096 Jul 11 21:25 .
ubuntu@primary:~$
```

```
ubuntu@primary:~$ helm install myhelloworldrelease helloworld
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
NAME: myhelloworldrelease
LAST DEPLOYED: Tue Jul 11 21:29:56 2023
NAMESPACE: default
STATUS: deployed
REVISION: 1
NOTES:
1. Get the application URL by running these commands:
  export POD_NAME=$(kubectl get pods --namespace default -l "app.kubernetes.io/name=helloworld,app.kubernetes.io/instance=myhelloworld
" -o jsonpath="{.items[0].metadata.name}")
  export CONTAINER_PORT=$(kubectl get pod --namespace default $POD_NAME -o jsonpath="{.spec.containers[0].ports[0].containerPort}")
  echo "Visit http://127.0.0.1:8080 to use your application"
  kubectl --namespace default port-forward $POD_NAME 8080:$CONTAINER_PORT
ubuntu@primary:~$
```

```
ubuntu@primary:~$ helm list -a
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
NAME                    NAMESPACE       REVISION        UPDATED                                 STATUS          CHART
PP VERSION
myhellowdrldrelease     default         1               2023-07-11 21:29:56.004749205 +0300 EEST        deployed        helloworld-0.1.0
.16.0
ubuntu@primary:~$
```

vim values.yaml

values.yaml + (D:\3_Devops-Sl

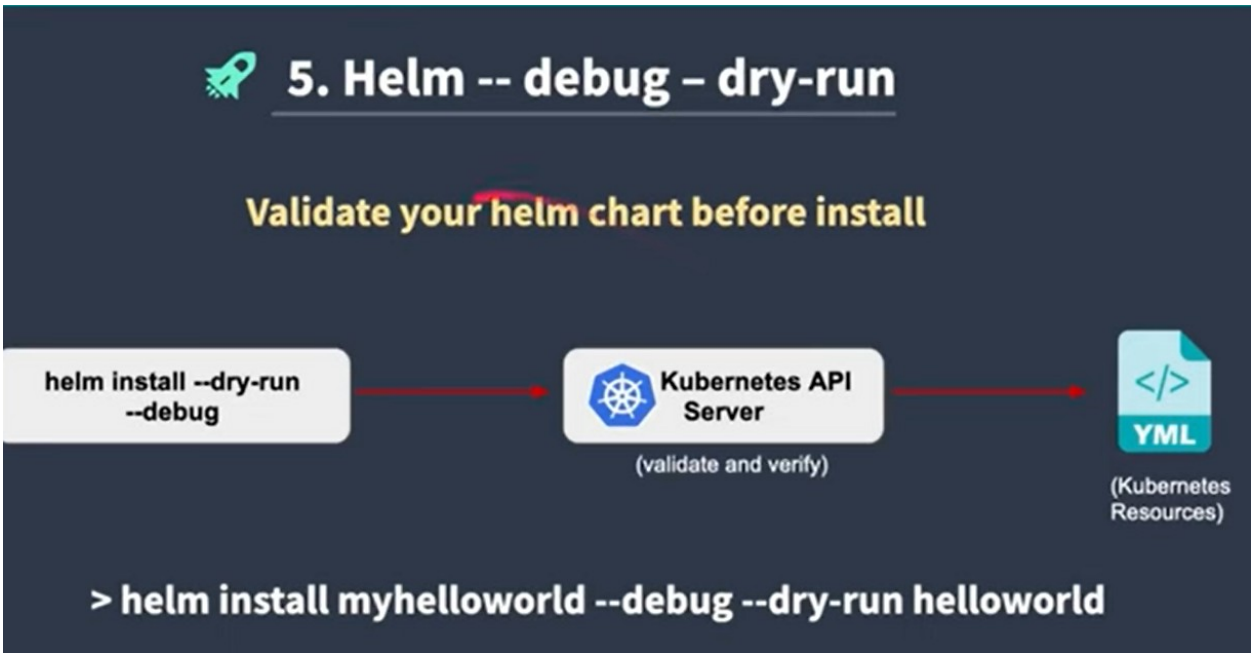
```
# Default values for helloworld.
# This is a YAML-formatted file.
# Declare variables to be passed into your templates.

replicaCount: 2

image:
  repository: nginx
  pullPolicy: IfNotPresent
  # Overrides the image tag whose default is the chart appVersion.
  tag: ""

imagePullSecrets: []
nameOverride: ""
fullnameOverride: ""

serviceAccount:
  # Specifies whether a service account should be created
  create: true
  # Annotations to add to the service account
  annotations: {}
  # The name of the service account to use.
  # If not set and create is true, a name is generated using the fullname template
```

```
ubuntu@primary:~$ helm upgrade myhelloworldrelease helloworld
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
Release "myhelloworldrelease" has been upgraded. Happy Helming!
NAME: myhelloworldrelease
LAST DEPLOYED: Tue Jul 11 21:33:31 2023
NAMESPACE: default
STATUS: deployed
REVISION: 2
```

```
ubuntu@primary:~$ helm list -a
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
NAME                    NAMESPACE       REVISION        UPDATED                                 STATUS          CHART
PP VERSION
myhelloworldrelease     default         2               2023-07-11 21:33:31.869754402 +0300 EEST        deployed        helloworld-0.1.0
.16.0
ubuntu@primary:~$
```

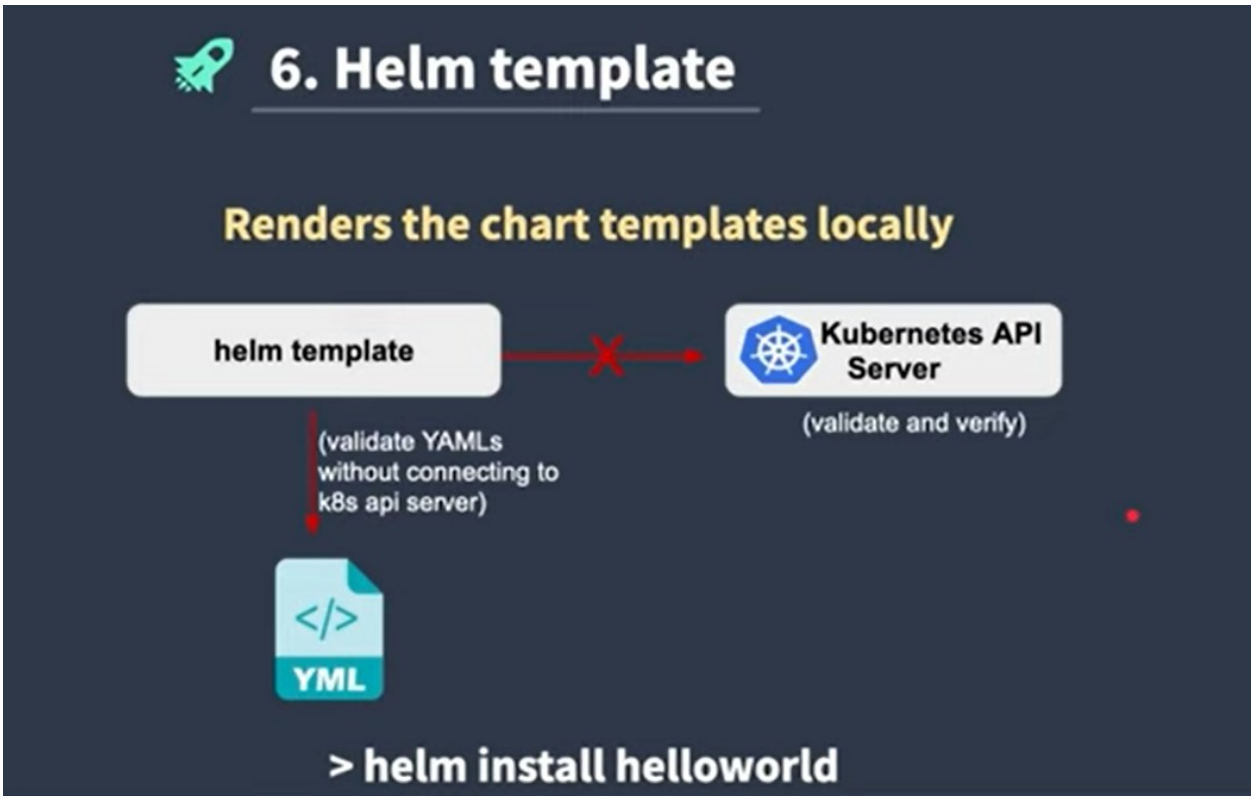

```
ubuntu@primary:~$ helm rollback myhelloworldrelease 1
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
Rollback was a success! Happy Helming!
ubuntu@primary:~$
```

```
ubuntu@primary:~$ helm list -a
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
NAME                    NAMESPACE       REVISION        UPDATED                                 STATUS
ERSION
myhelloworldrelease     default         3               2023-07-11 21:36:37.74212636 +0300 EEST deployed
0
ubuntu@primary:~$
```



```
ubuntu@primary:~$ helm install myhelloworldrelease --debug --dry-run helloworld
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
install.go:178: [debug] Original chart version: ""
install.go:195: [debug] CHART PATH: /home/ubuntu/helloworld

NAME: myhelloworldrelease
LAST DEPLOYED: Tue Jul 11 21:43:32 2023
NAMESPACE: default
STATUS: pending-install
REVISION: 1
USER-SUPPLIED VALUES:
{}

COMPUTED VALUES:
affinity: {}
autoscaling:
  enabled: false
  maxReplicas: 100
  minReplicas: 1
  targetCPUUtilizationPercentage: 80
fullnameOverride: ""
image:
```

```
---
# Source: helloworld/templates/deployment.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: myhelloworldrelease
  labels:
    helm.sh/chart: helloworld-0.1.0
    app.kubernetes.io/name: helloworld
    app.kubernetes.io/instance: myhelloworldrelease
    app.kubernetes.io/version: "1.16.0"
    app.kubernetes.io/managed-by: Helm
spec:
  replicas: 2
  selector:
    matchLabels:
      app.kubernetes.io/name: helloworld
      app.kubernetes.io/instance: myhelloworldrelease
  template:
    metadata:
      labels:
```




```
ubuntu@primary:~$ helm template helloworld
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
---
# Source: helloworld/templates/serviceaccount.yaml
apiVersion: v1
kind: ServiceAccount
metadata:
  name: release-name-helloworld
  labels:
    helm.sh/chart: helloworld-0.1.0
    app.kubernetes.io/name: helloworld
    app.kubernetes.io/instance: release-name
    app.kubernetes.io/version: "1.16.0"
    app.kubernetes.io/managed-by: Helm
---
# Source: helloworld/templates/service.yaml
apiVersion: v1
kind: Service
metadata:
  name: release-name-helloworld
  labels:
    helm.sh/chart: helloworld-0.1.0
    app.kubernetes.io/name: helloworld
    app.kubernetes.io/instance: release-name
    app.kubernetes.io/version: "1.16.0"
    app.kubernetes.io/managed-by: Helm
spec:
  type: ClusterIP
  ports:
```

```
  name: release-name-helloworld
  labels:
    helm.sh/chart: helloworld-0.1.0
    app.kubernetes.io/name: helloworld
    app.kubernetes.io/instance: release-name
    app.kubernetes.io/version: "1.16.0"
    app.kubernetes.io/managed-by: Helm
spec:
  type: ClusterIP
  ports:
    - port: 80
      targetPort: http
      protocol: TCP
      name: http
  selector:
    app.kubernetes.io/name: helloworld
    app.kubernetes.io/instance: release-name
---
# Source: helloworld/templates/deployment.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: release-name-helloworld
  labels:
    helm.sh/chart: helloworld-0.1.0
    app.kubernetes.io/name: helloworld
    app.kubernetes.io/instance: release-name
    app.kubernetes.io/version: "1.16.0"
    app.kubernetes.io/managed-by: Helm
spec:
  replicas: 2
  selector:
    matchLabels:
      app.kubernetes.io/name: helloworld
```

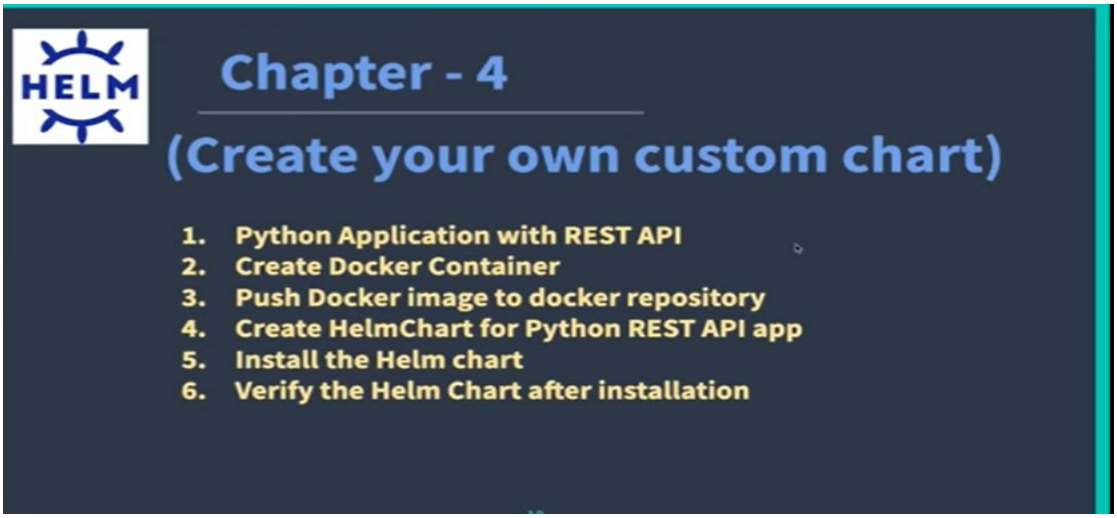
7. Helm Lint

Find any errors or misconfiguration

> helm lint helloworld

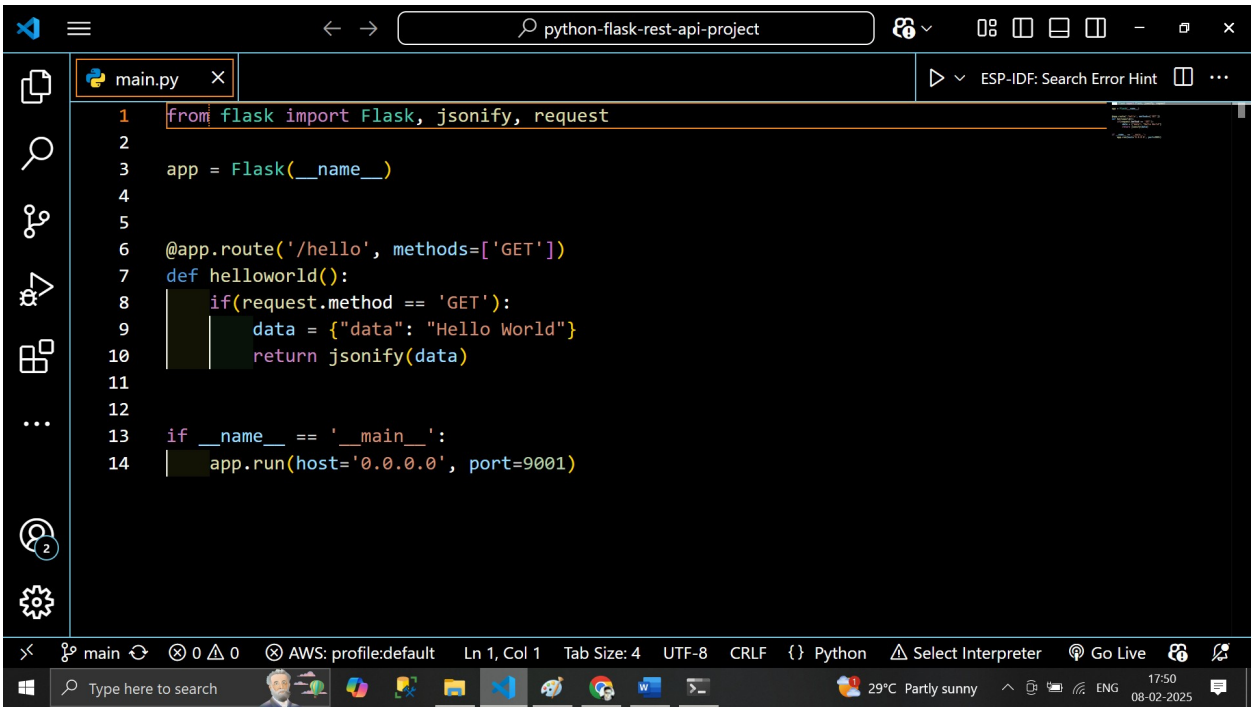
```
ubuntu@primary:~$ helm lint helloworld
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube
==> Linting helloworld
[INFO] Chart.yaml: icon is recommended

1 chart(s) linted, 0 chart(s) failed
ubuntu@primary:~$
```

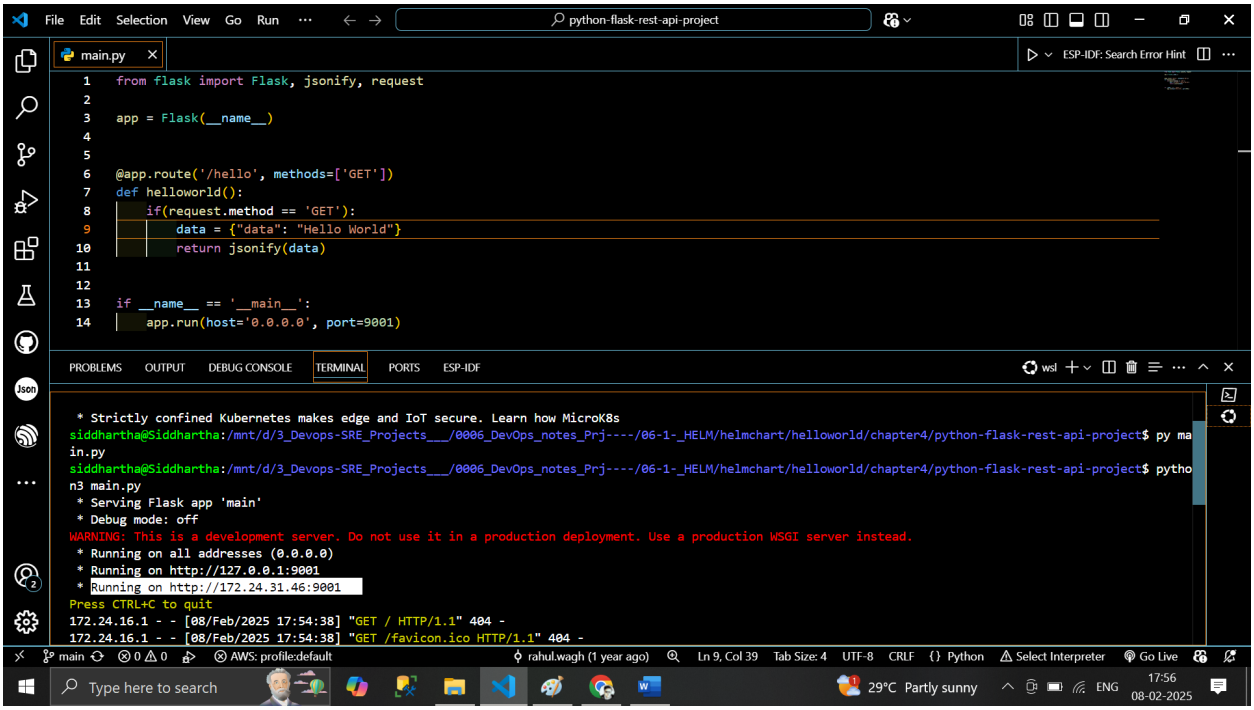

```
ubuntu@primary:~$ helm uninstall myhelloworldrelease
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
release "myhelloworldrelease" uninstalled
ubuntu@primary:~$
ubuntu@primary:~$
ubuntu@primary:~$ helm list -a
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
NAME    NAMESPACE       REVISION        UPDATED STATUS  CHART   APP VERSION
ubuntu@primary:~$
```

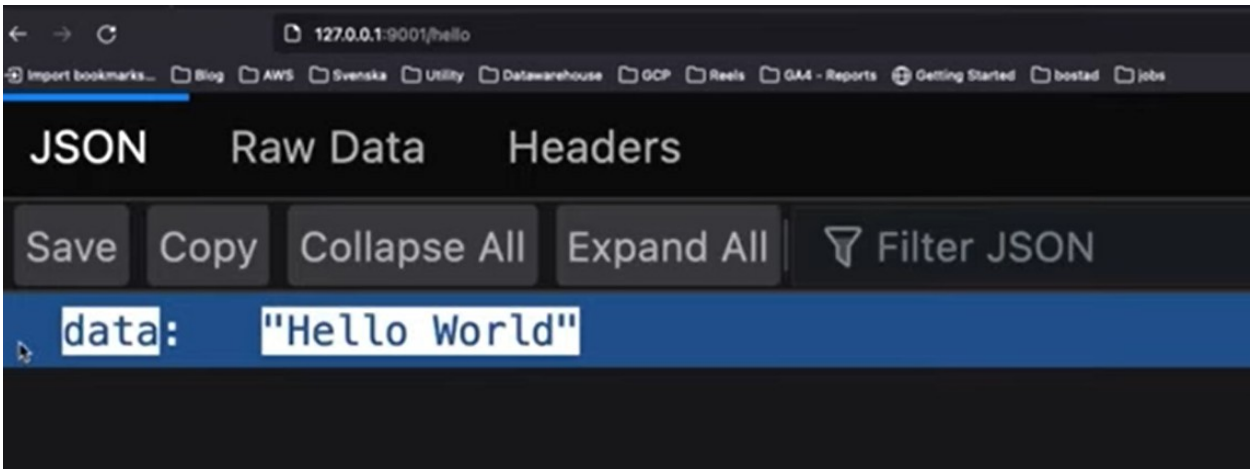


Chapter 4 - Creating Custom Helm Charts:

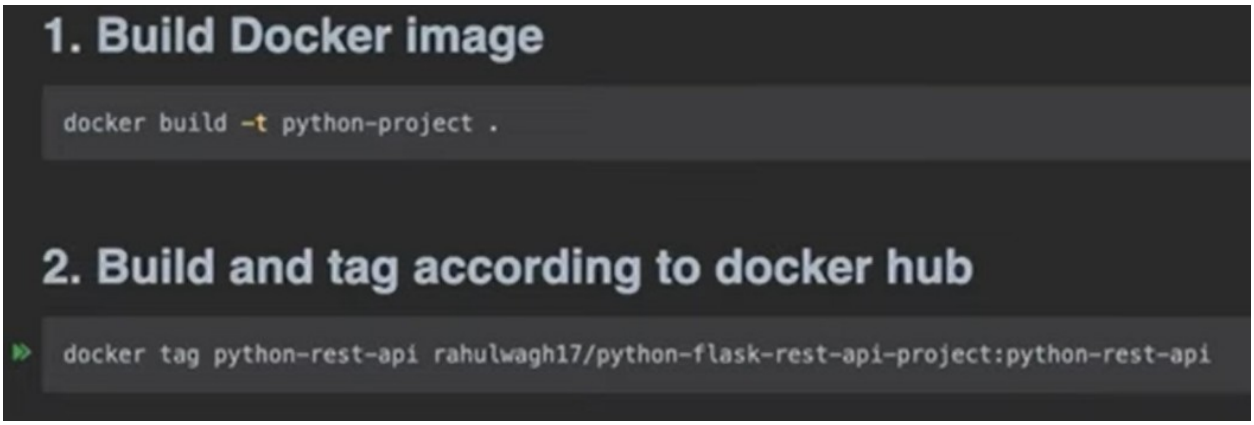
1. Designing a Python application with REST API
2. Docker Container creation and management
3. Pushing Docker images to repositories
4. Crafting Helm Charts for your Python REST API app
5. Helm chart installation and post-installation verification

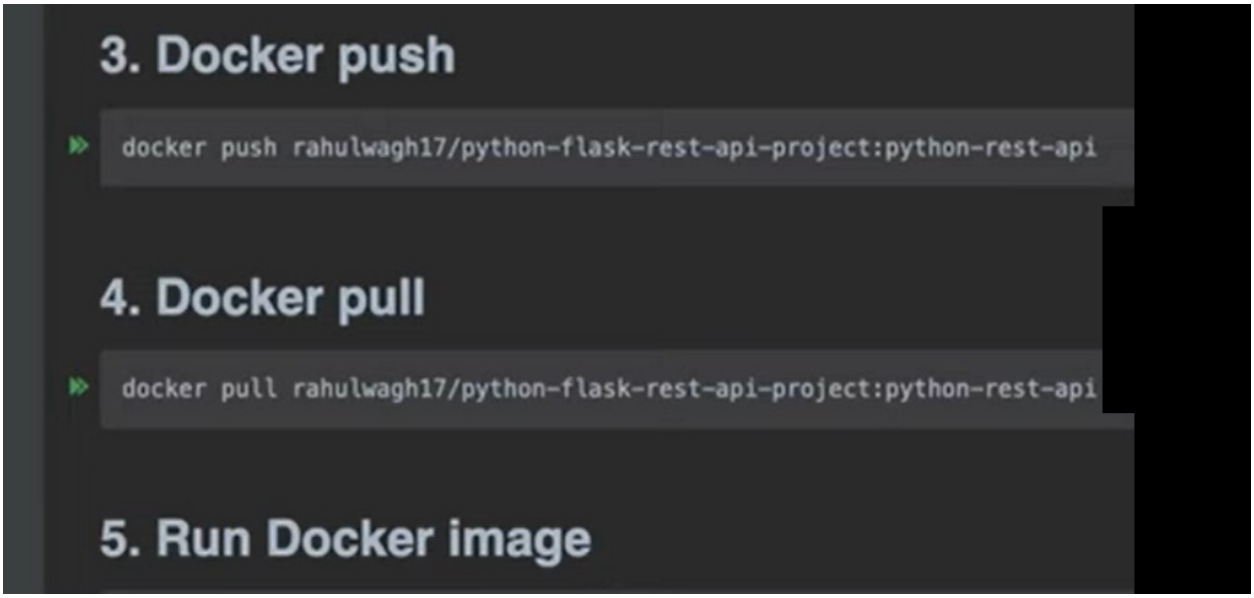

python-flask-rest-api-project
main.py
ESP-IDF: Search Error Hint
1 from flask import Flask, jsonify, request
2
3 app = Flask(__name__)
4
5
6 @app.route('/hello', methods=['GET'])
7 def helloworld():
8 if(request.method == 'GET'):
9 data = {"data": "Hello World"}
10 return jsonify(data)
11
12
13 if __name__ == '__main__':
14 app.run(host='0.0.0.0', port=9001)
main
0 0
AWS: profile:default
Ln 1, Col 1
Tab Size: 4
UTF-8
CRLF
Python
Select Interpreter
Go Live
Type here to search
29°C Partly sunny
ENG
17:50
08-02-2025


File Edit Selection View Go Run
python-flask-rest-api-project
main.py
ESP-IDF: Search Error Hint
1 from flask import Flask, jsonify, request
2
3 app = Flask(__name__)
4
5
6 @app.route('/hello', methods=['GET'])
7 def helloworld():
8 if(request.method == 'GET'):
9 data = {"data": "Hello World"}
10 return jsonify(data)
11
12
13 if __name__ == '__main__':
14 app.run(host='0.0.0.0', port=9001)
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS ESP-IDF
wsl
* Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
siddhartha@Siddhartha:/mnt/d/3_Devops-SRE_Projects___/0006_DevOps_notes_Prj----/06-1-_HELM/helmchart/helloworld/chapter4/python-flask-rest-api-project$ py ma
in.py
siddhartha@Siddhartha:/mnt/d/3_Devops-SRE_Projects___/0006_DevOps_notes_Prj----/06-1-_HELM/helmchart/helloworld/chapter4/python-flask-rest-api-project$ pytho
n3 main.py
* Serving Flask app 'main'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:9001
* Running on http://172.24.31.46:9001
Press CTRL+C to quit
172.24.16.1 - - [08/Feb/2025 17:54:38] "GET / HTTP/1.1" 404 -
172.24.16.1 - - [08/Feb/2025 17:54:38] "GET /favicon.ico HTTP/1.1" 404 -
main
0 0
AWS: profile:default
rahul.wagh (1 year ago)
Ln 9, Col 39
Tab Size: 4
UTF-8
CRLF
Python
Select Interpreter
Go Live
Type here to search
29°C Partly sunny
ENG
17:56
08-02-2025

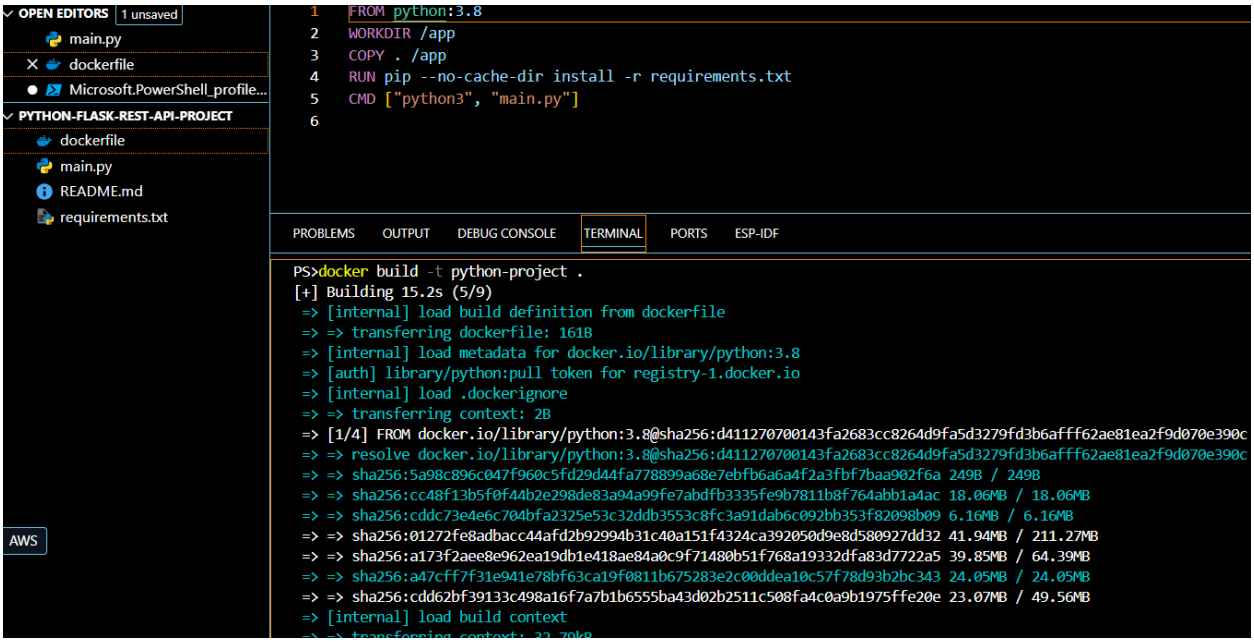


run the docker container locally run the docker img & push the docker image



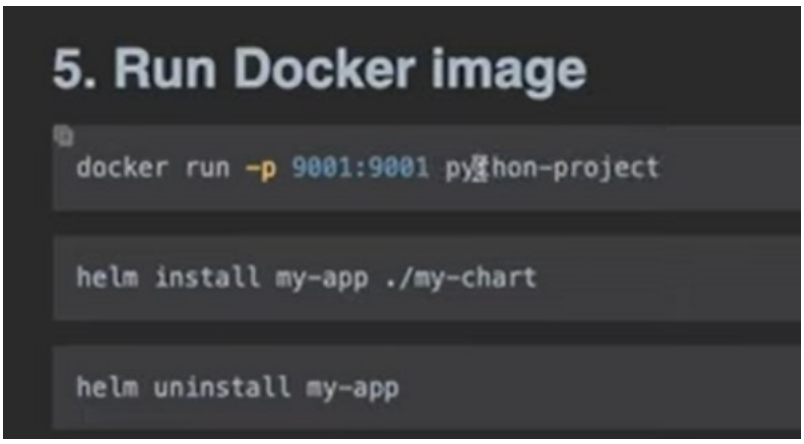


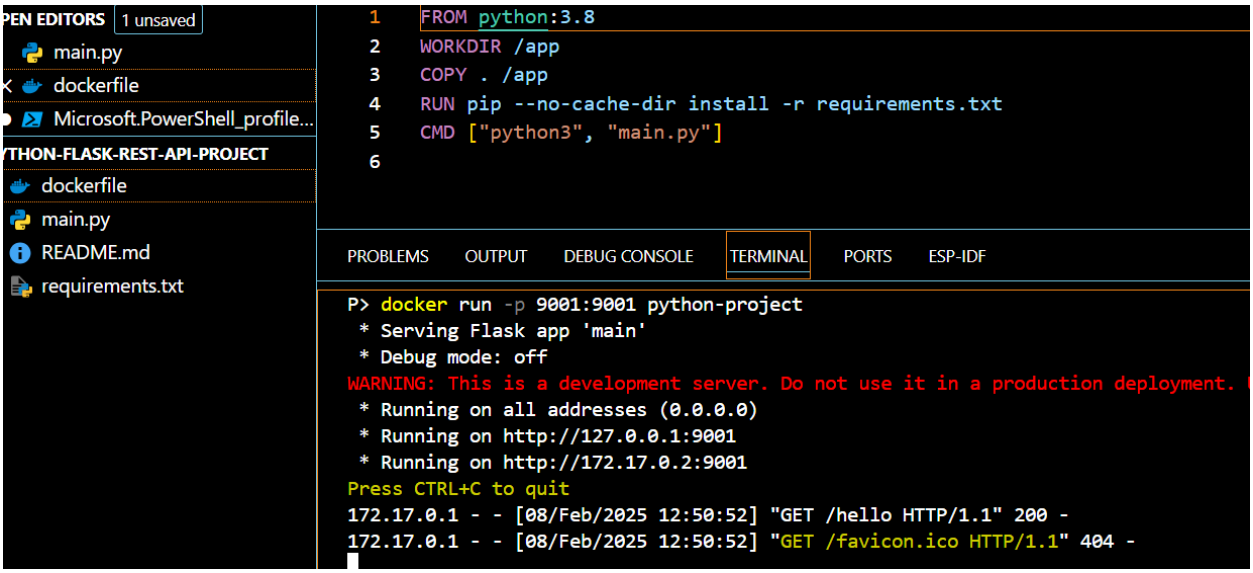
Practicals

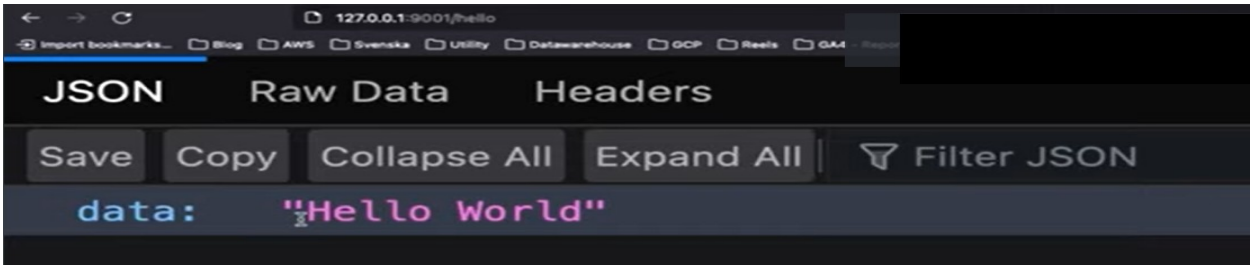


docker img

```
P> docker image ls
REPOSITORY                    TAG       IMAGE ID       CREATED             SIZE
python-project                latest    463f7785835b   About a minute ago   1.46GB
siddhartha082/go-web-app      v1        9170efd33aa6   3 months ago         45.9MB
gcr.io/k8s-minikube/kicbase   <none>    81df28859520   5 months ago         1.81GB
PS>docker ps -a
CONTAINER ID   IMAGE                                  COMMAND                  CREATED        STATUS                      PORTS
     NAMES
e13d821d8b64   gcr.io/k8s-minikube/kicbase:v0.0.45   "/usr/local/bin/entr…"   3 months ago   Exited (130) 3 months ago
     minikube
PS>
```





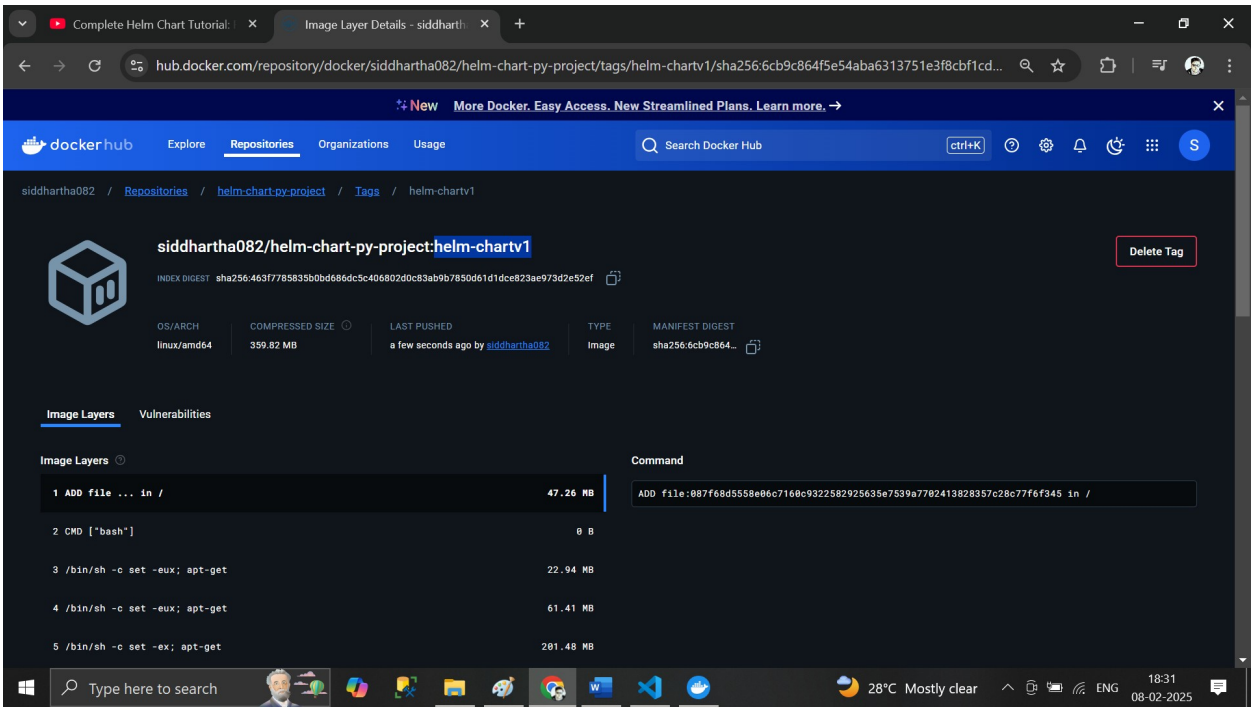


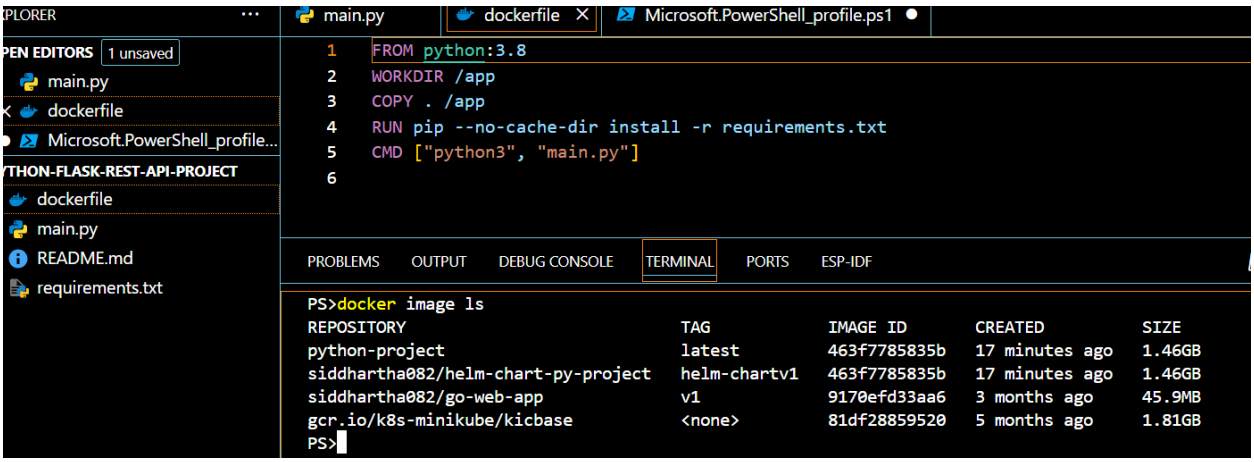
Push the docker img to docker hub

```
P> docker image ls
REPOSITORY                    TAG       IMAGE ID       CREATED          SIZE
python-project                latest    463f7785835b   11 minutes ago   1.46GB
siddhartha082/go-web-app      v1        9170efd33aa6   3 months ago     45.9MB
gcr.io/k8s-minikube/kicbase   <none>    81df28859520   5 months ago     1.81GB
P> docker login
Authenticating with existing credentials...
Login Succeeded
P> docker tag python-project siddhartha082/helm-chart-py-project:helm-chartv1
```

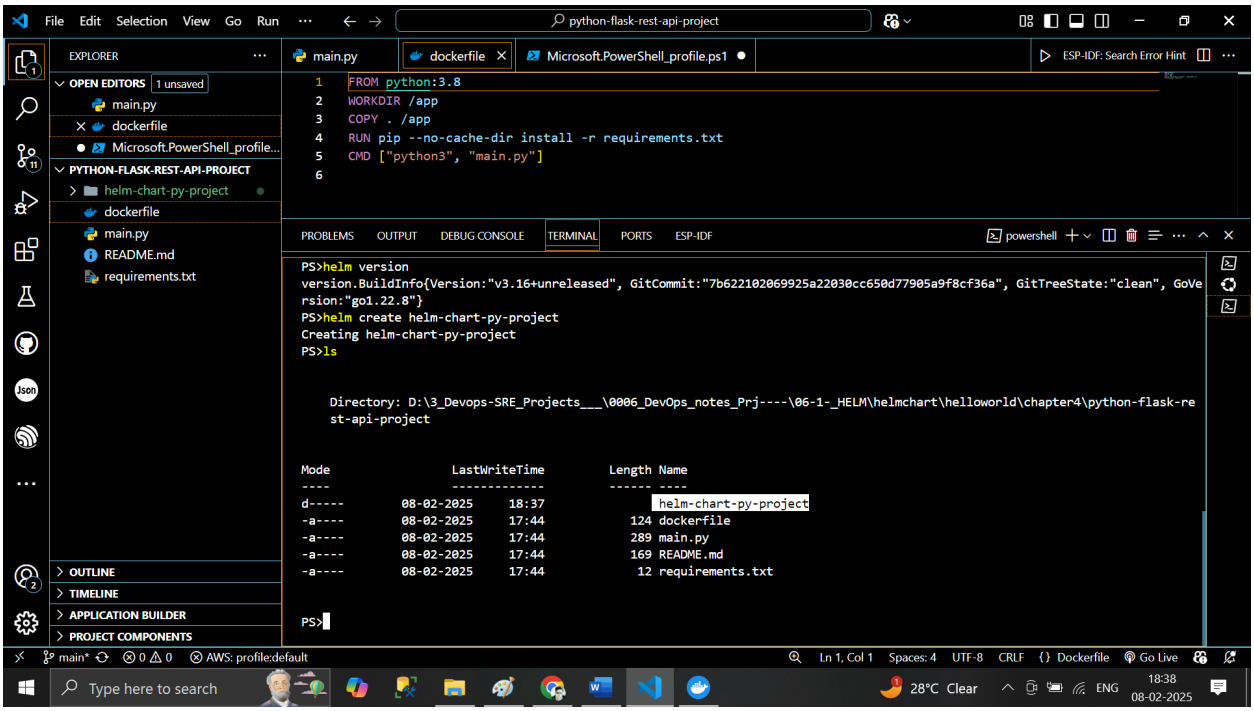
```
PS>docker push siddhartha082/helm-chart-py-project:helm-chartv1
The push refers to repository [docker.io/siddhartha082/helm-chart-py-project]
ff9518d9817c: Pushed
a173f2aee8e9: Pushed
01272fe8adba: Pushed
cc48f13b5f0f: Pushed
5a98c896c047: Pushed
2c08678988eb: Pushed
e50e07929188: Pushed
cdd62bf39133: Pushed
a47cff7f31e9: Pushed
cddc73e4e6c7: Pushed
511f39350be2: Pushed
helm-chartv1: digest: sha256:463f7785835b0bd686dc5c406802d0c83ab9b7850d61d1dce823ae973d2e52ef size: 856
P>
```

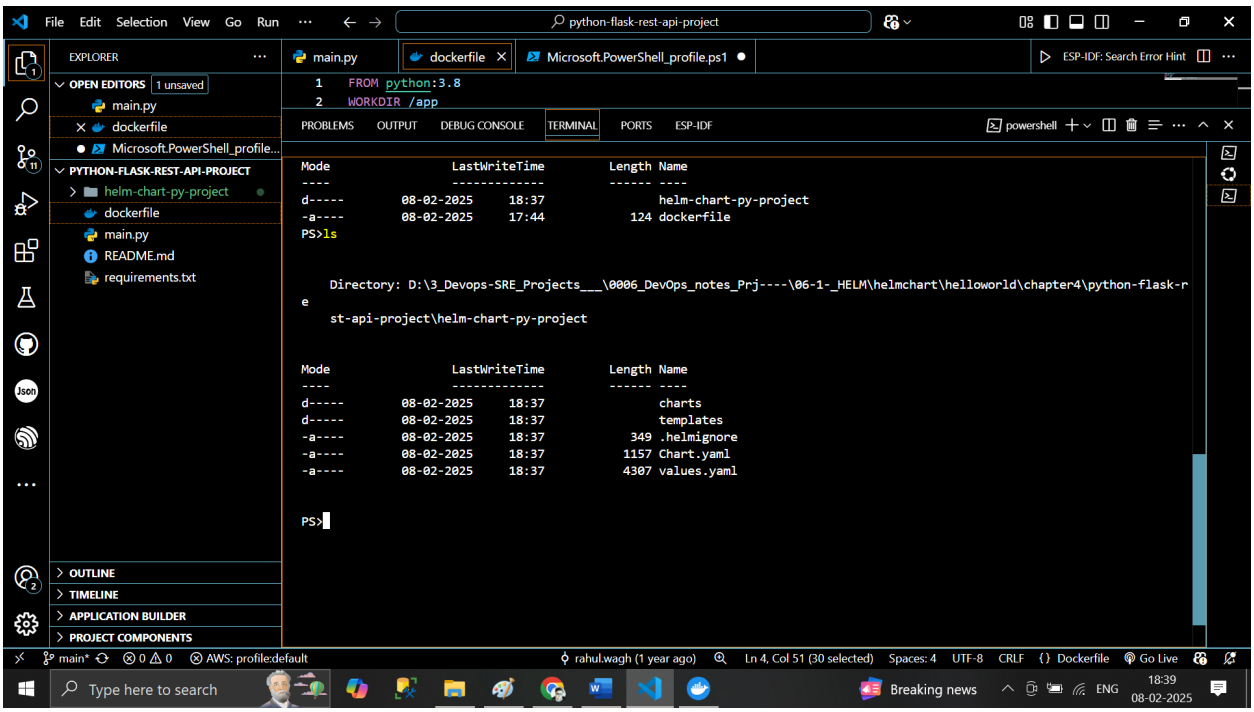
successfully img pushed in docker Hub

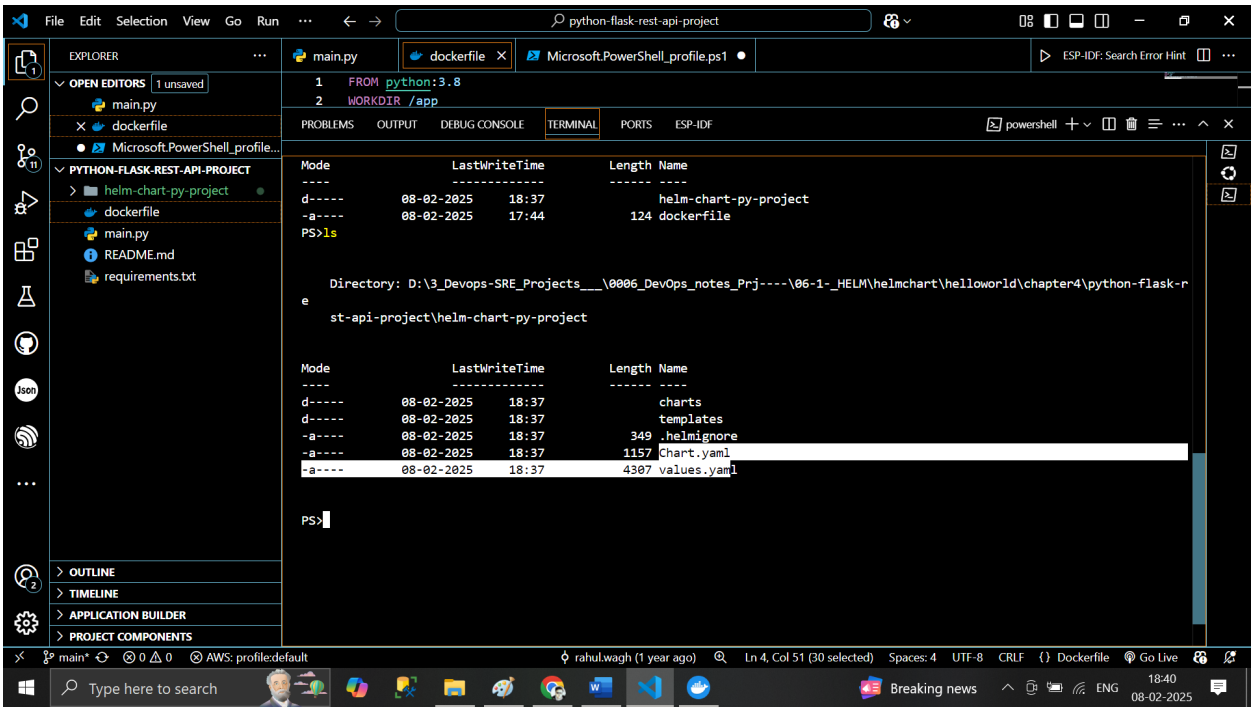




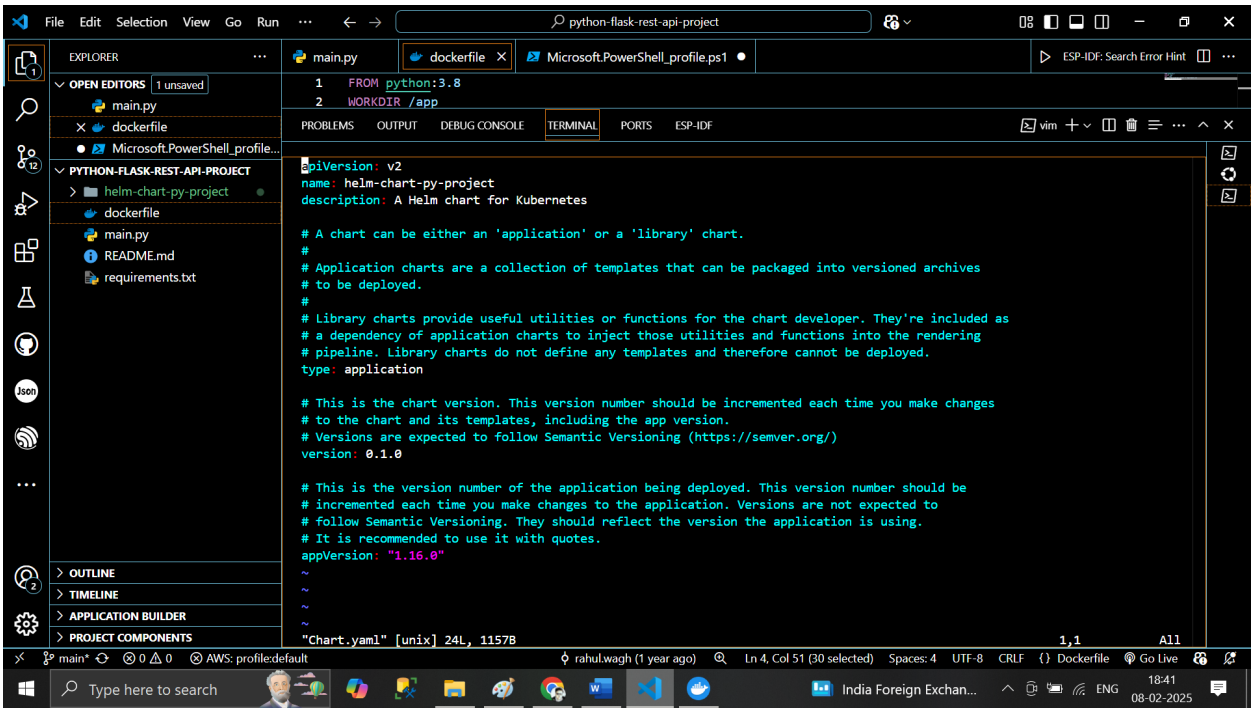
create helm chart


File Edit Selection View Go Run
python-flask-rest-api-project
EXPLORER
OPEN EDITORS 1 unsaved
main.py
dockerfile
Microsoft.PowerShell_profile...
PYTHON-FLASK-REST-API-PROJECT
helm-chart-py-project
dockerfile
main.py
README.md
requirements.txt
OUTLINE
TIMELINE
APPLICATION BUILDER
PROJECT COMPONENTS
main.py
dockerfile
Microsoft.PowerShell_profile.ps1
ESP-IDF: Search Error Hint
1 FROM python:3.8
2 WORKDIR /app
3 COPY . /app
4 RUN pip --no-cache-dir install -r requirements.txt
5 CMD ["python3", "main.py"]
6
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS ESP-IDF
powershell
PS>helm version
version.BuildInfo{Version:"v3.16+unreleased", GitCommit:"7b622102069925a22030cc650d77905a9f8cf36a", GitTreeState:"clean", GoVersion:"go1.22.8"}
PS>helm create helm-chart-py-project
Creating helm-chart-py-project
PS>ls
Directory: D:\3_Devops-SRE_Projects___\0006_DevOps_notes_Prj----\06-1-_HELM\helmchart\helloworld\chapter4\python-flask-rest-api-project
Mode LastWriteTime Length Name
d----- 08-02-2025 18:37 helm-chart-py-project
-a---- 08-02-2025 17:44 124 dockerfile
-a---- 08-02-2025 17:44 289 main.py
-a---- 08-02-2025 17:44 169 README.md
-a---- 08-02-2025 17:44 12 requirements.txt
PS>
main* 0 0 AWS: profile:default
Ln 1, Col 1 Spaces: 4 UTF-8 CRLF {} Dockerfile Go Live
Type here to search
28°C Clear
ENG
18:38
08-02-2025


File Edit Selection View Go Run
python-flask-rest-api-project
EXPLORER
OPEN EDITORS 1 unsaved
main.py
dockerfile
Microsoft.PowerShell_profile...
PYTHON-FLASK-REST-API-PROJECT
helm-chart-py-project
dockerfile
main.py
README.md
requirements.txt
OUTLINE
TIMELINE
APPLICATION BUILDER
PROJECT COMPONENTS
main.py
dockerfile
Microsoft.PowerShell_profile.ps1
ESP-IDF: Search Error Hint
1 FROM python:3.8
2 WORKDIR /app
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS ESP-IDF
powershell
Mode LastWriteTime Length Name
d----- 08-02-2025 18:37 helm-chart-py-project
-a---- 08-02-2025 17:44 124 dockerfile
PS>ls
Directory: D:\3_Devops-SRE_Projects___\0006_DevOps_notes_Prj----\06-1-_HELM\helmchart\helloworld\chapter4\python-flask-re
st-api-project\helm-chart-py-project
Mode LastWriteTime Length Name
d----- 08-02-2025 18:37 charts
d----- 08-02-2025 18:37 templates
-a---- 08-02-2025 18:37 349 .helmignore
-a---- 08-02-2025 18:37 1157 Chart.yaml
-a---- 08-02-2025 18:37 4307 values.yaml
PS>
main* 0 0 AWS: profile:default
rahul.wagh (1 year ago)
Ln 4, Col 51 (30 selected) Spaces: 4 UTF-8 CRLF {} Dockerfile Go Live
Type here to search
Breaking news
ENG
18:39
08-02-2025

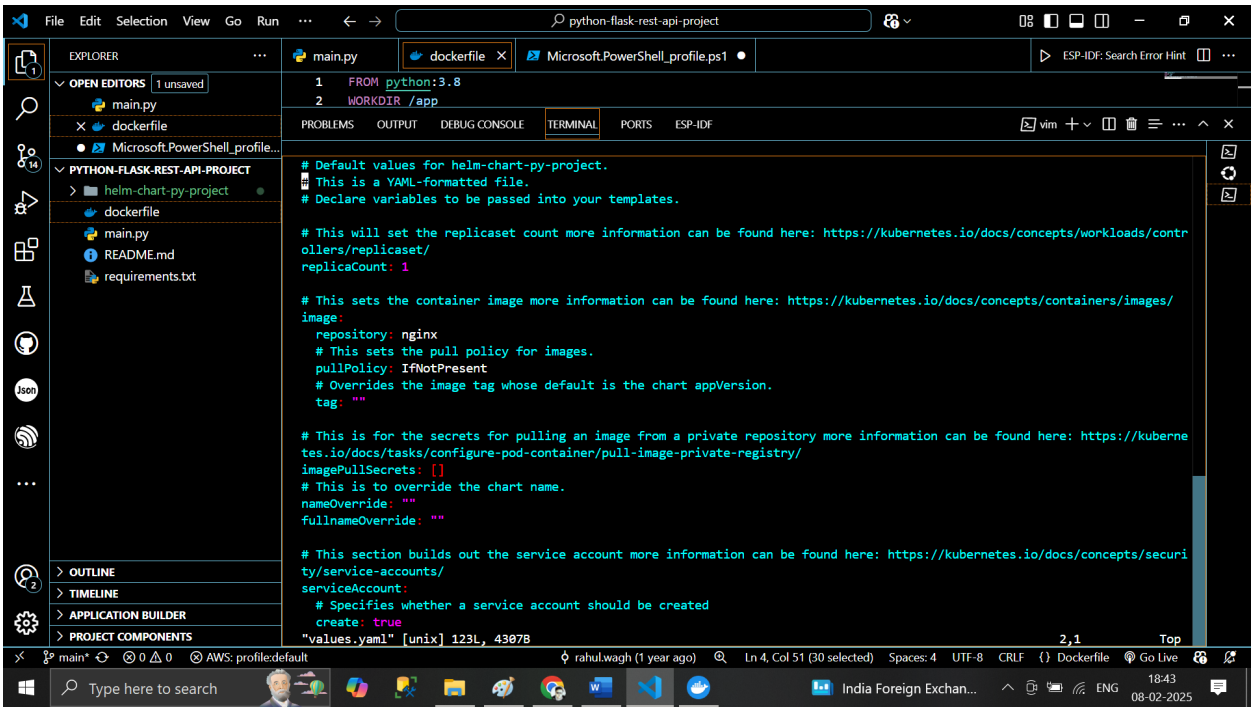


update the chart.yml edit mode

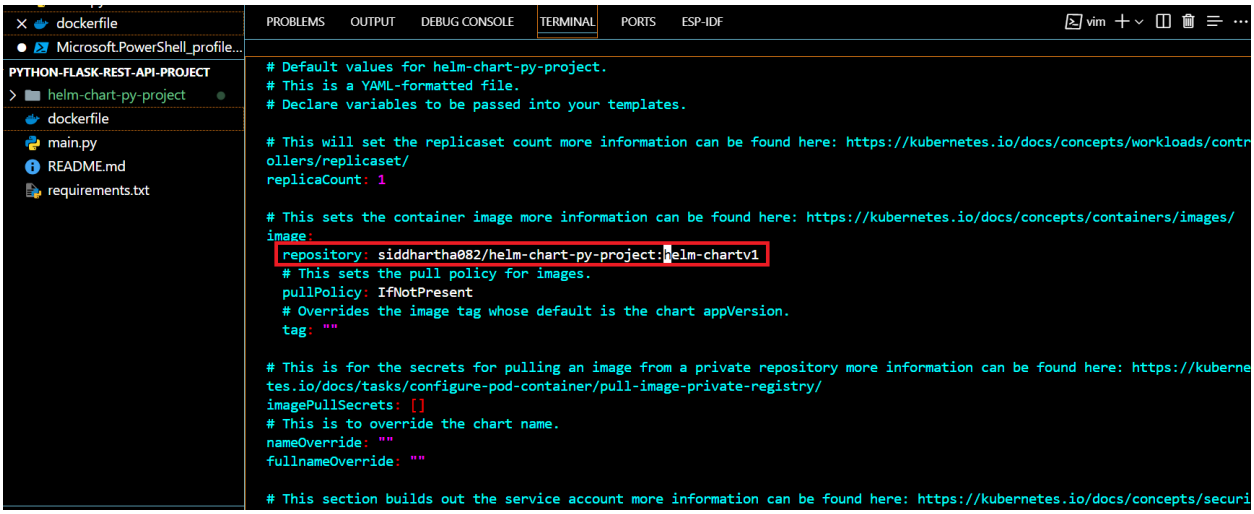


set comment the app_version .save + quit

now update the same for values.yml file

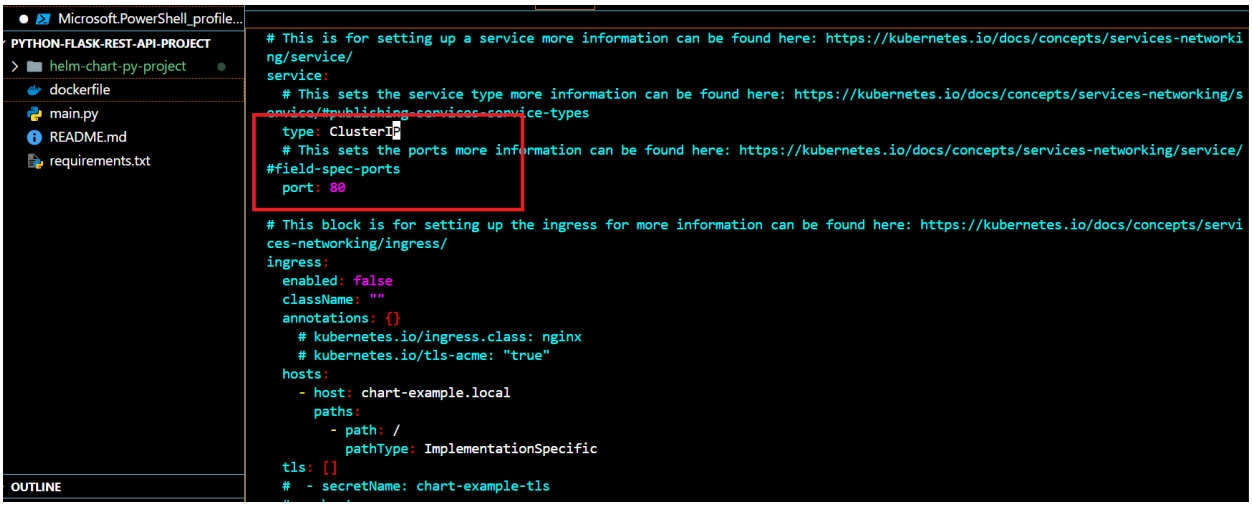


change the repo – ngnix to your docker repo name : tagname too save + quit

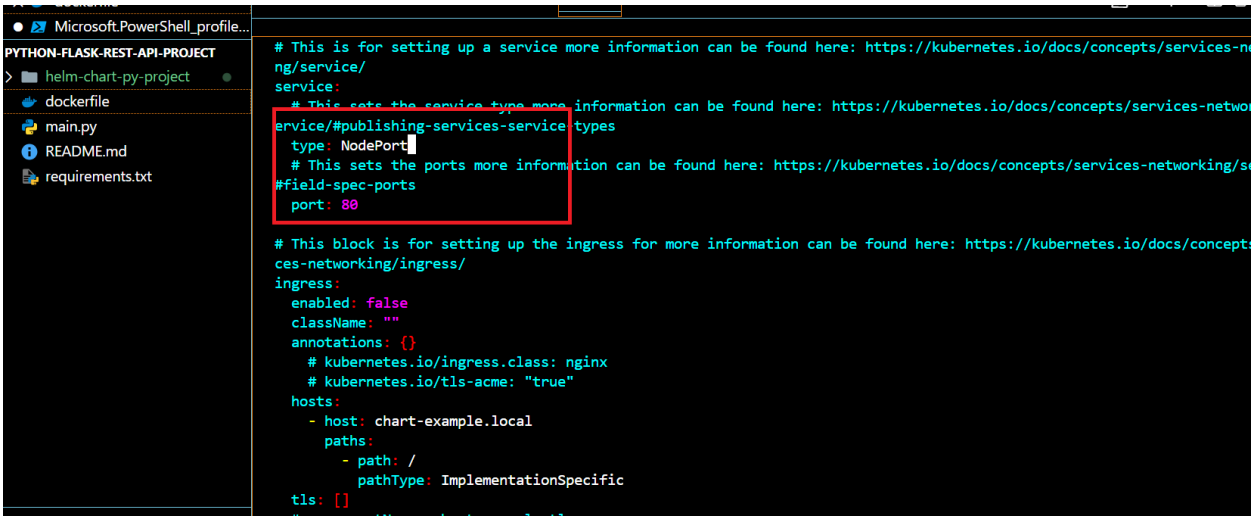


we have updated helm chart + values yaml ..

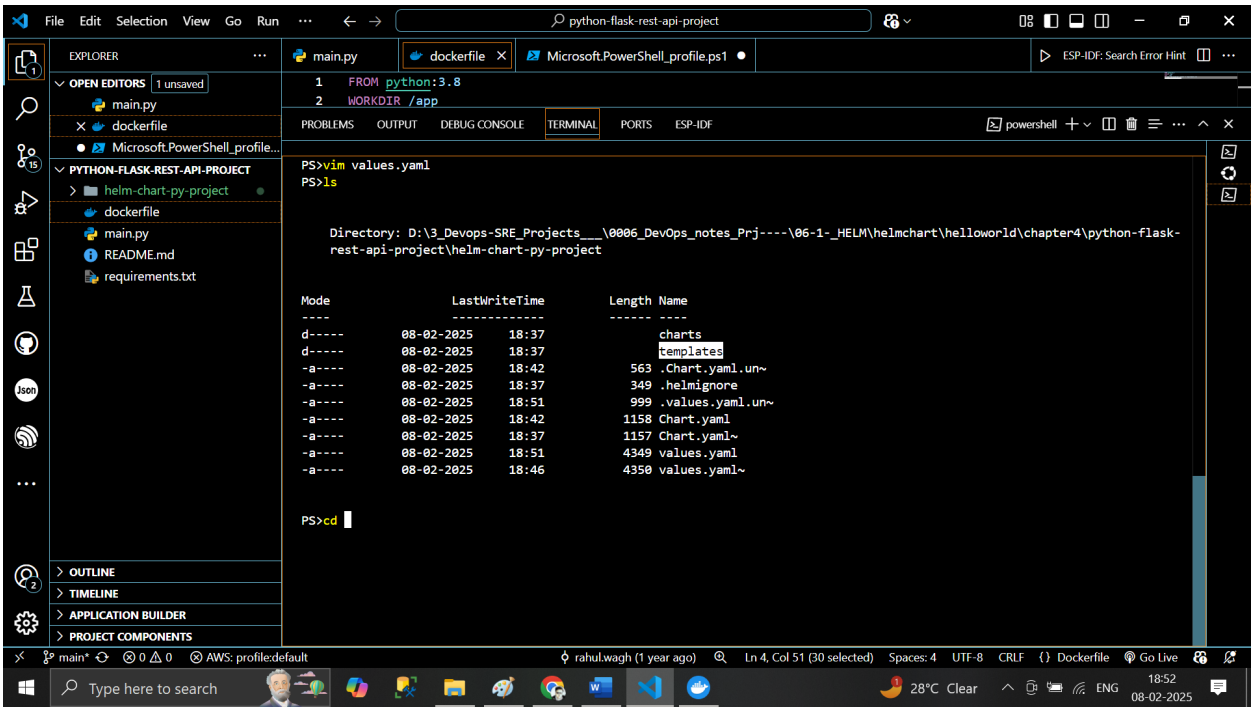
in Values.yaml change the clusterIP to NodePort .. for port forwarding

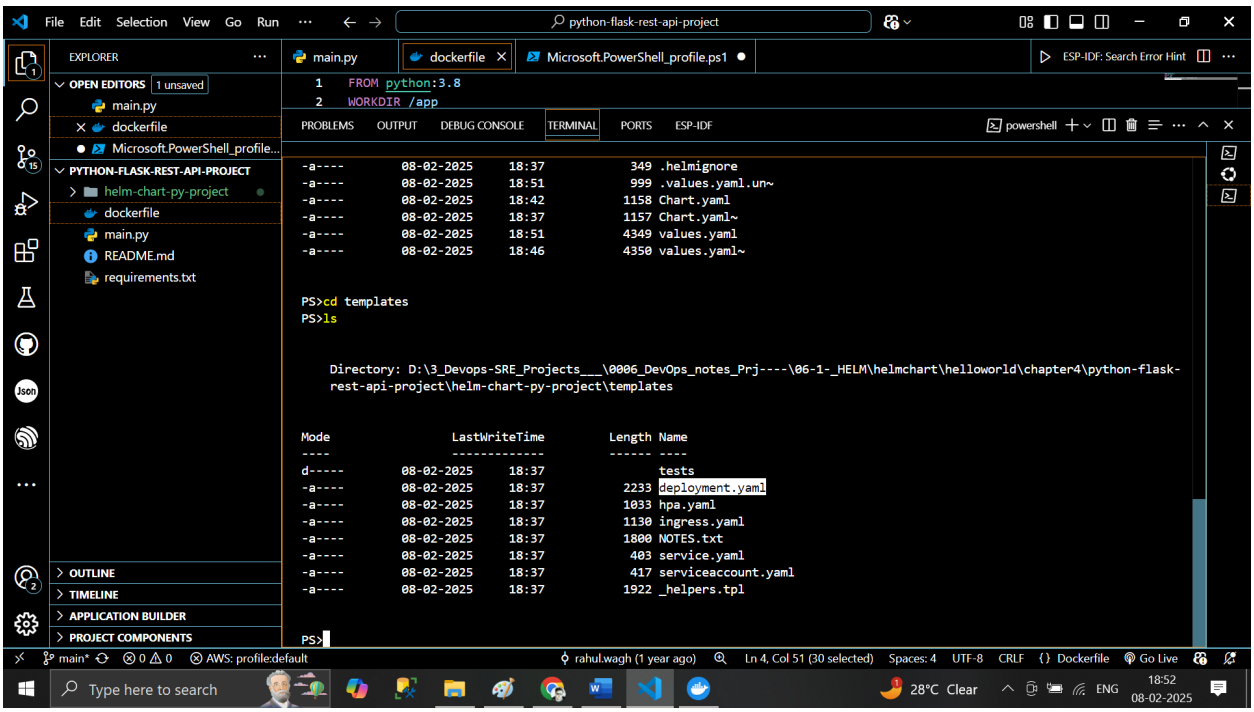


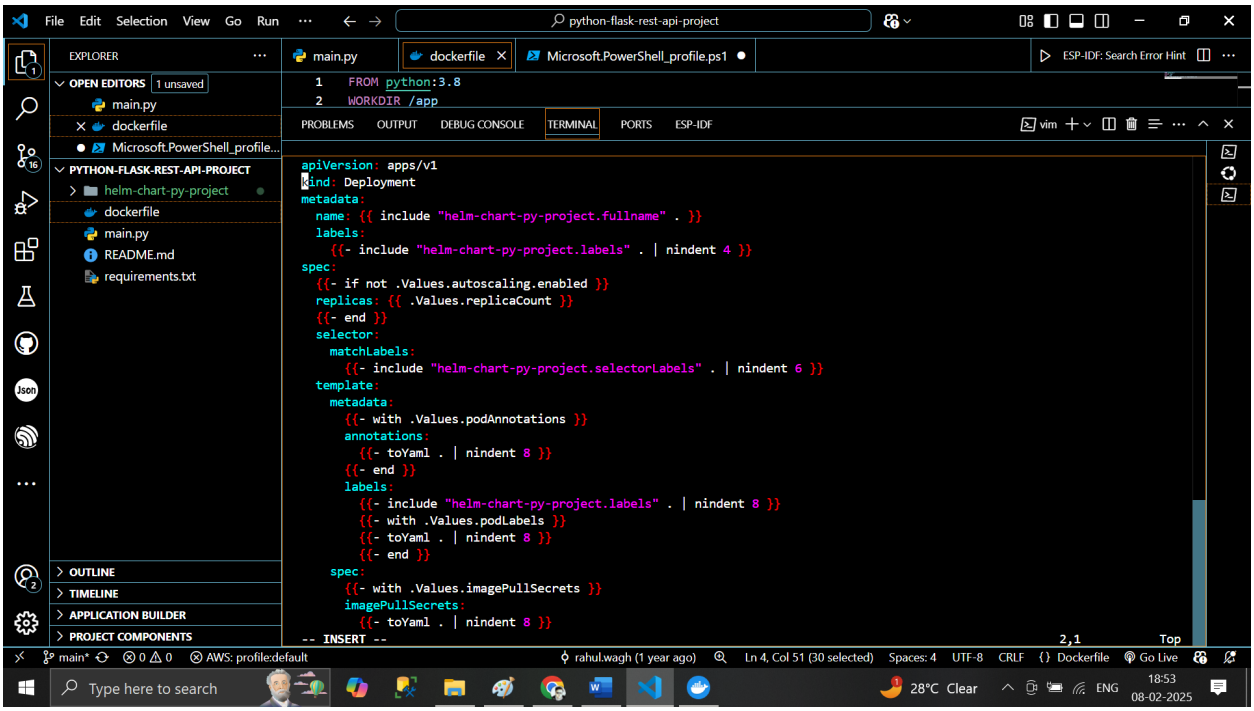
NodePort

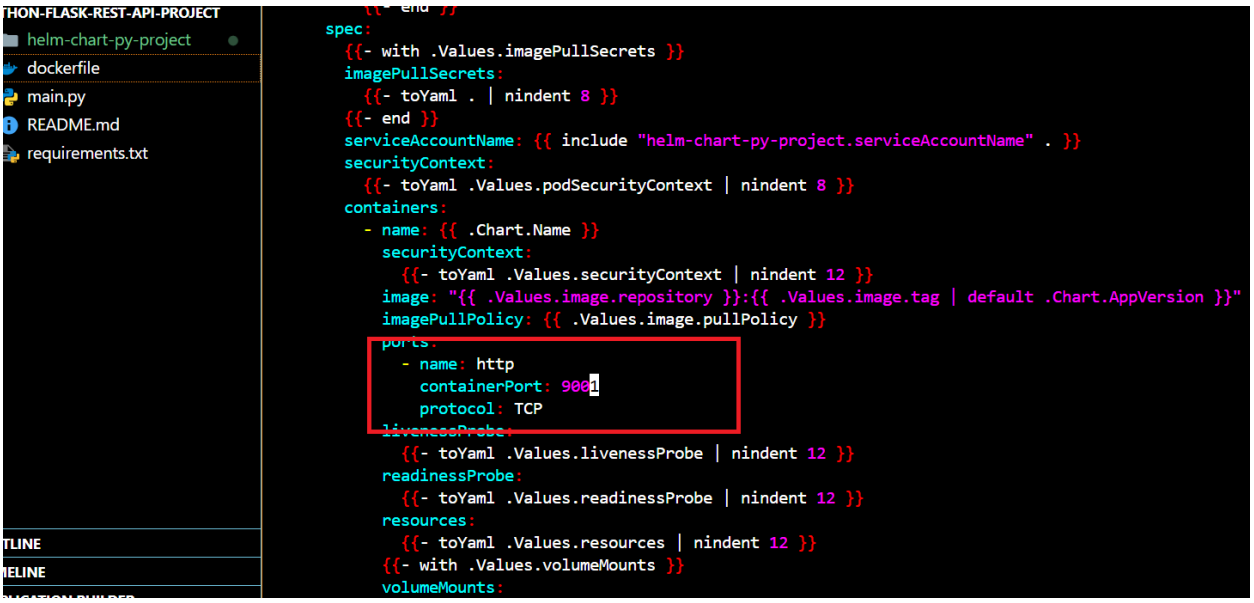


Go inside the template . deployment.yml file

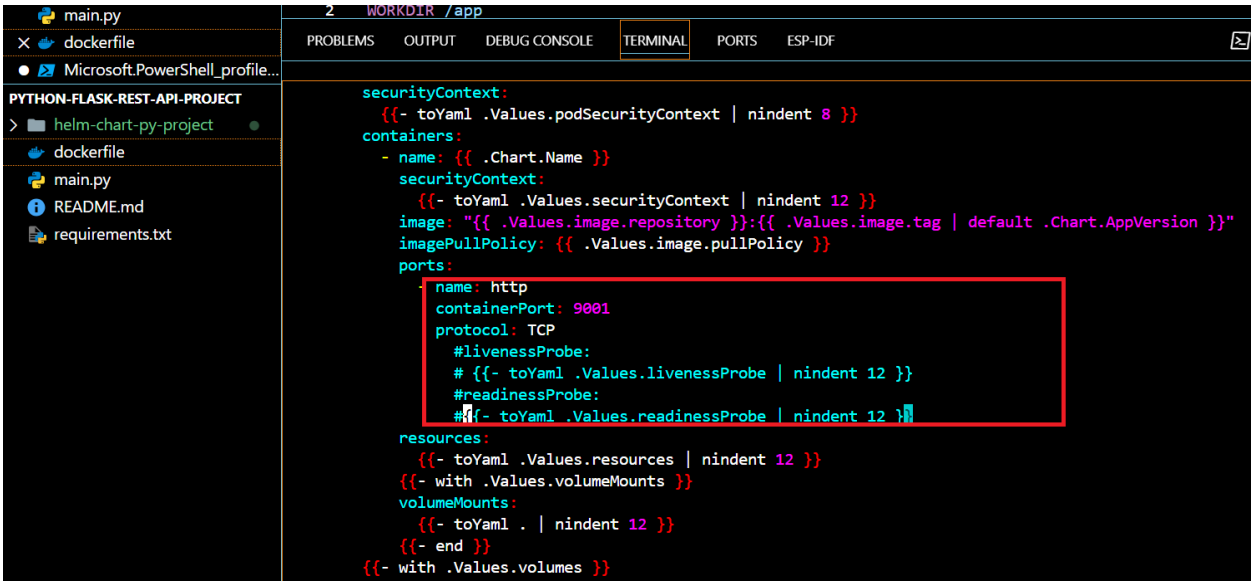

File Edit Selection View Go Run
python-flask-rest-api-project
EXPLORER
OPEN EDITORS 1 unsaved
main.py
dockerfile
Microsoft.PowerShell_profile...
PYTHON-FLASK-REST-API-PROJECT
helm-chart-py-project
dockerfile
main.py
README.md
requirements.txt
OUTLINE
TIMELINE
APPLICATION BUILDER
PROJECT COMPONENTS
main.py
dockerfile
Microsoft.PowerShell_profile.ps1
ESP-IDF: Search Error Hint
1 FROM python:3.8
2 WORKDIR /app
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS ESP-IDF
powershell
PS>vim values.yaml
PS>ls
Directory: D:\3_Devops-SRE_Projects___\0006_DevOps_notes_Prj----\06-1-_HELM\helmchart\helloworld\chapter4\python-flask-rest-api-project\helm-chart-py-project
Mode LastWriteTime Length Name
d----- 08-02-2025 18:37 charts
d----- 08-02-2025 18:37 templates
-a---- 08-02-2025 18:42 563 .Chart.yaml.un~
-a---- 08-02-2025 18:37 349 .helmignore
-a---- 08-02-2025 18:51 999 .values.yaml.un~
-a---- 08-02-2025 18:42 1158 Chart.yaml
-a---- 08-02-2025 18:37 1157 Chart.yaml~
-a---- 08-02-2025 18:51 4349 values.yaml
-a---- 08-02-2025 18:46 4350 values.yaml~
PS>cd
main* 0 0 AWS: profile:default
rahul.wagh (1 year ago) Ln 4, Col 51 (30 selected) Spaces: 4 UTF-8 CRLF Dockerfile Go Live
Type here to search
28°C Clear
ENG
18:52
08-02-2025


File Edit Selection View Go Run
python-flask-rest-api-project
EXPLORER
OPEN EDITORS 1 unsaved
main.py
dockerfile
Microsoft.PowerShell_profile...
PYTHON-FLASK-REST-API-PROJECT
helm-chart-py-project
dockerfile
main.py
README.md
requirements.txt
OUTLINE
TIMELINE
APPLICATION BUILDER
PROJECT COMPONENTS
main.py
dockerfile
Microsoft.PowerShell_profile.ps1
ESP-IDF: Search Error Hint
1 FROM python:3.8
2 WORKDIR /app
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS ESP-IDF
powershell
-a---- 08-02-2025 18:37 349 .helmignore
-a---- 08-02-2025 18:51 999 .values.yaml.un~
-a---- 08-02-2025 18:42 1158 Chart.yaml
-a---- 08-02-2025 18:37 1157 Chart.yaml~
-a---- 08-02-2025 18:51 4349 values.yaml
-a---- 08-02-2025 18:46 4350 values.yaml~
PS>cd templates
PS>ls
Directory: D:\3_Devops-SRE_Projects___\0006_DevOps_notes_Prj----\06-1-_HELM\helmchart\helloworld\chapter4\python-flask-rest-api-project\helm-chart-py-project\templates
Mode LastWriteTime Length Name
d----- 08-02-2025 18:37 tests
-a---- 08-02-2025 18:37 2233 deployment.yaml
-a---- 08-02-2025 18:37 1033 hpa.yaml
-a---- 08-02-2025 18:37 1130 ingress.yaml
-a---- 08-02-2025 18:37 1800 NOTES.txt
-a---- 08-02-2025 18:37 403 service.yaml
-a---- 08-02-2025 18:37 417 serviceaccount.yaml
-a---- 08-02-2025 18:37 1922 _helpers.tpl
PS>
main* 0 0 AWS: profile:default
rahul.wagh (1 year ago) Ln 4, Col 51 (30 selected) Spaces: 4 UTF-8 CRLF Dockerfile Go Live
Type here to search
28°C Clear
ENG
18:52
08-02-2025

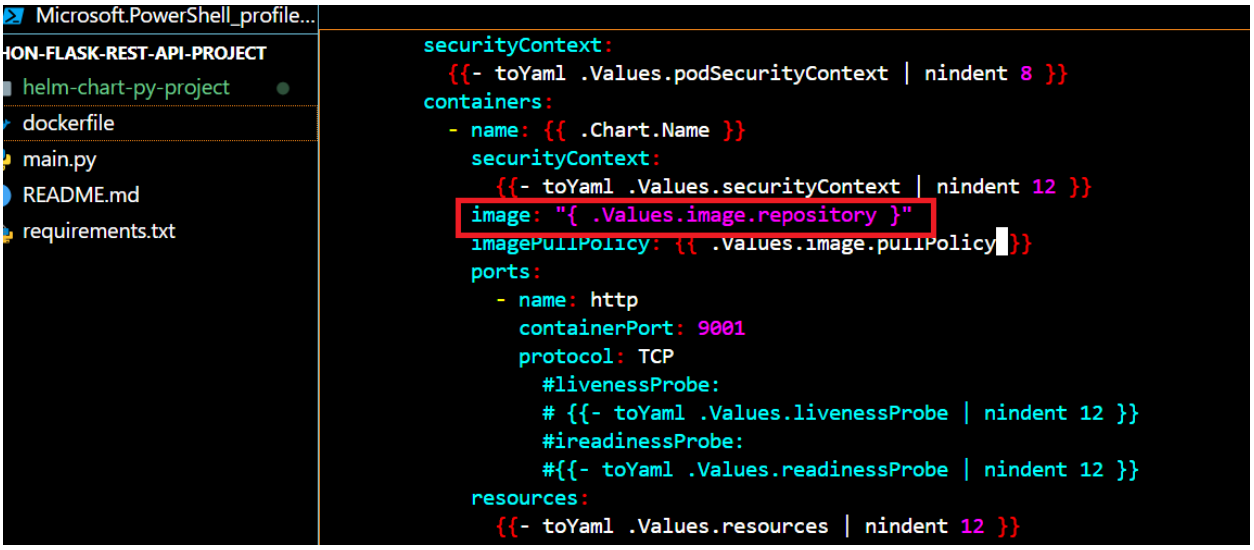


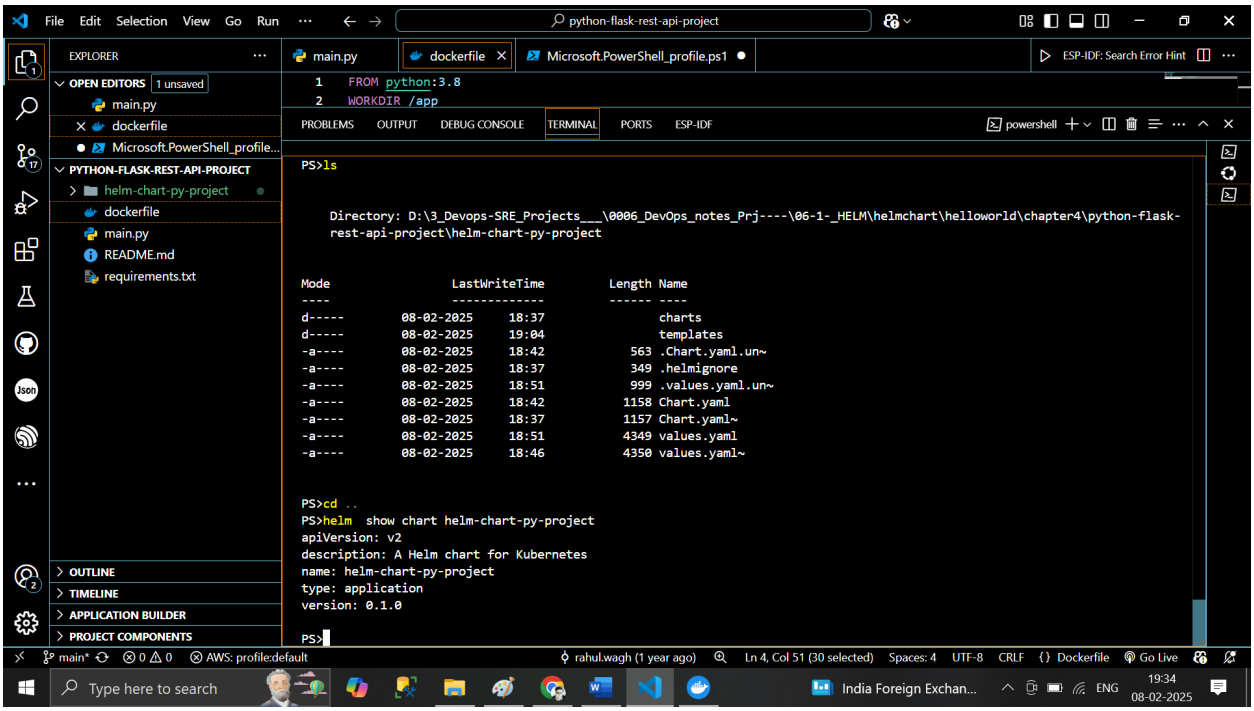


Now Comment out probes .. liveness + readiness probes



as in values.yml fiel we have commented the app_version



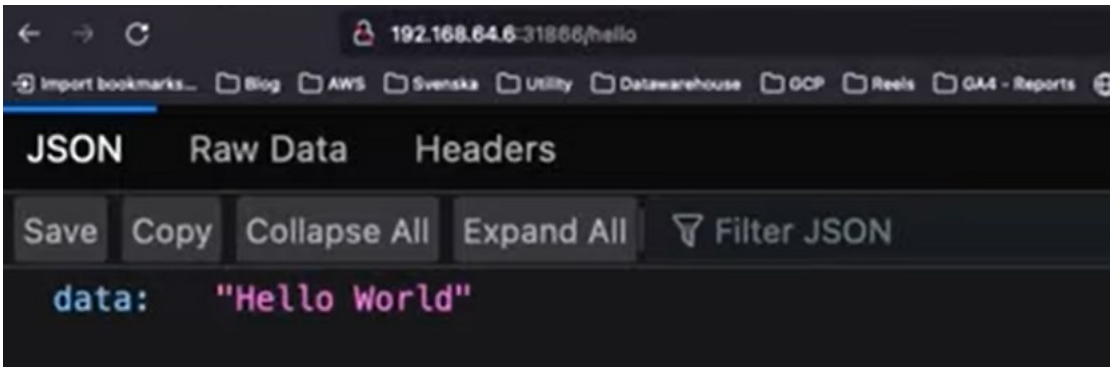


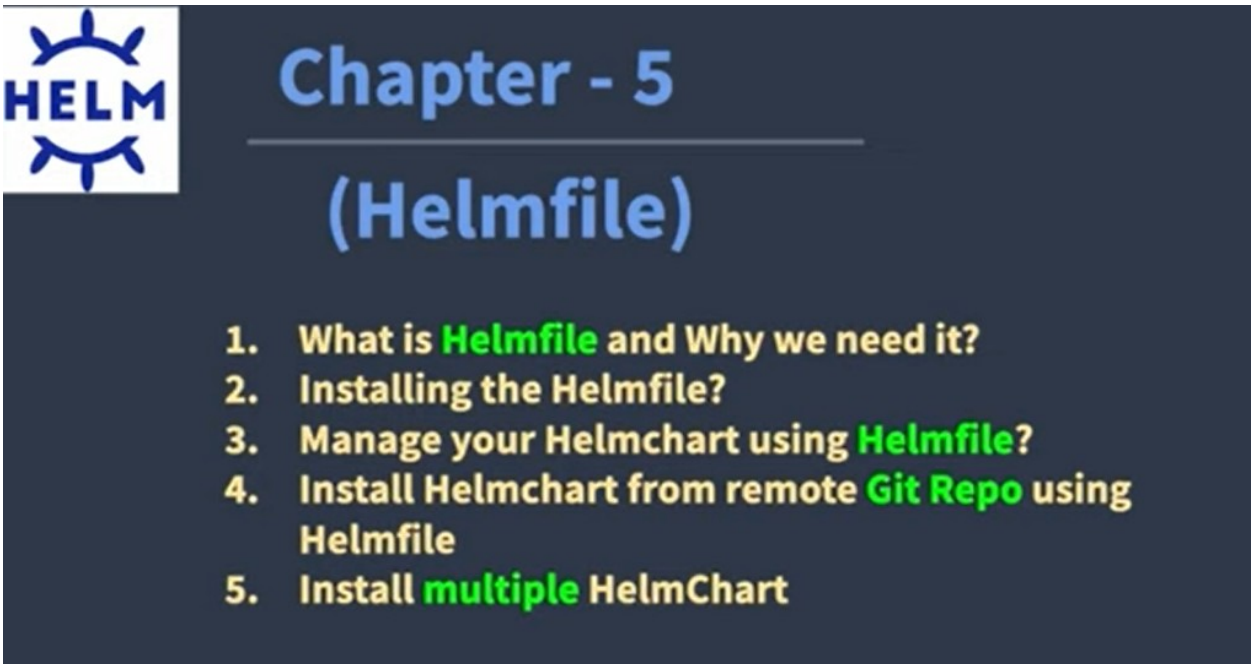

```
ubuntu@primary:~$ helm install mypythonhelmchart python-flask-rest-api-project
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
NAME: mypythonhelmchart
LAST DEPLOYED: Tue Jul 11 23:29:34 2023
NAMESPACE: default
STATUS: deployed
REVISION: 1
NOTES:
1. Get the application URL by running these commands:
  export NODE_PORT=$(kubectl get --namespace default -o jsonpath="{.spec.ports[0].nodePort}" services mypythonhelmch
-project)
  export NODE_IP=$(kubectl get nodes --namespace default -o jsonpath="{.items[0].status.addresses[0].address}")
  echo http://$NODE_IP:$NODE_PORT
ubuntu@primary:~$
```

```
ubuntu@primary:~$ helm list -a
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
NAME                    NAMESPACE       REVISION        UPDATED                                 STATUS          CHART
                APP VERSION
mypythonhelmchart       default         1               2023-07-11 23:29:34.825975291 +0300 EEST        deployed        python-flask-rest-
-project-0.1.0
ubuntu@primary:~$
```

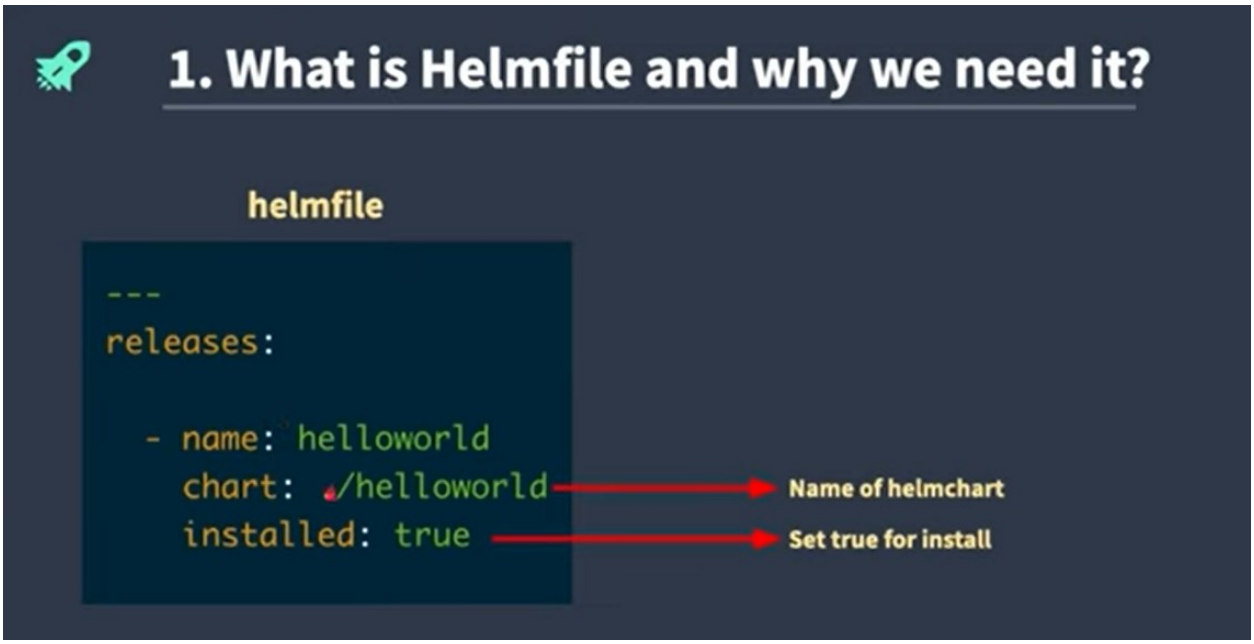
```
ubuntu@primary:~$ kubectl get deployments
NAME                                            READY   UP-TO-DATE   AVAILABLE   AGE
mypythonhelmchart-python-flask-rest-api-project   1/1     1            1           64s
ubuntu@primary:~$
ubuntu@primary:~$
ubuntu@primary:~$ kubectl get service
NAME                                            TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)        AGE
kubernetes                                      ClusterIP   10.152.183.1     <none>        443/TCP        2d8h
mypythonhelmchart-python-flask-rest-api-project   NodePort    10.152.183.123   <none>        80:31866/TCP   87s
ubuntu@primary:~$
```

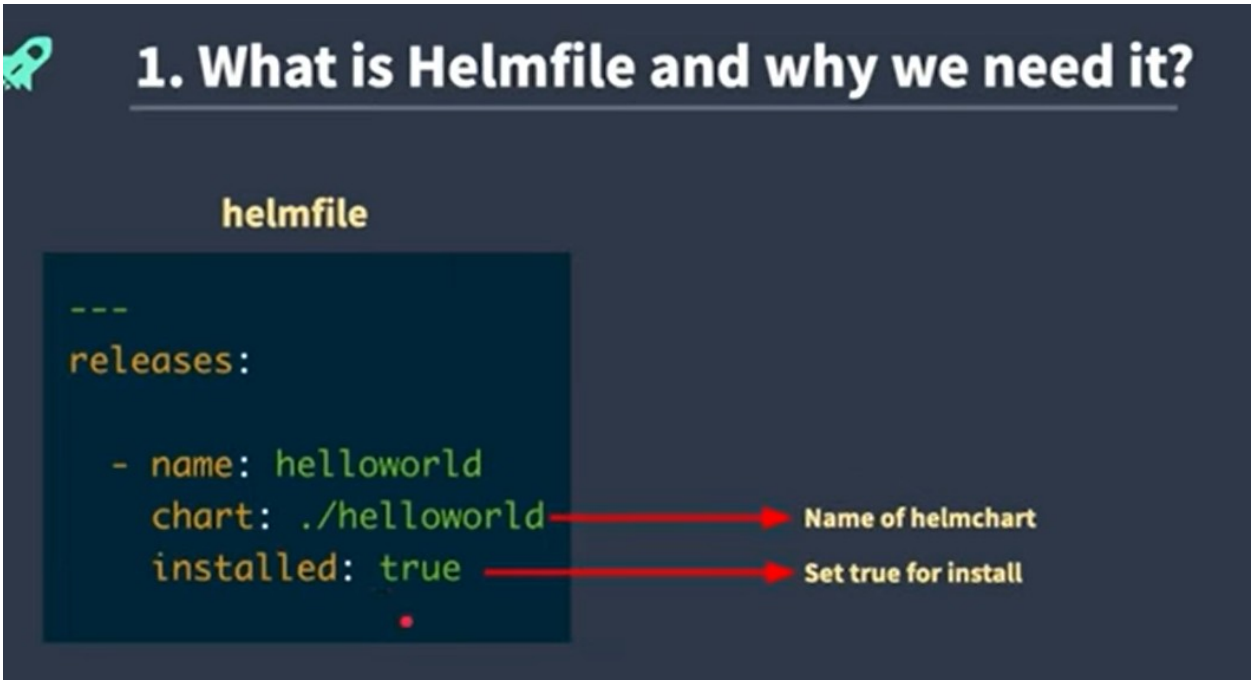
```
ubuntu@primary:~$ hostname -I
192.168.64.6 10.1.226.128 fd16:8590:7284:4e59:5054:ff:fe7f:b17d
ubuntu@primary:~$
```




HELM
Chapter - 5
(Helmfile)
1. What is Helmfile and Why we need it?
2. Installing the Helmfile?
3. Manage your Helmchart using Helmfile?
4. Install Helmchart from remote Git Repo using Helmfile
5. Install multiple HelmChart


1. What is Helmfile and why we need it?
helm commands
helm install my-app ./my-chart
helm uninstall my-app
$ helmfile sync


1. What is Helmfile and why we need it?
helmfile
---
releases:
- name: helloworld
chart: ./helloworld
installed: true
Name of helmchart
Set true for install


1. What is Helmfile and why we need it?
helmfile
---
releases:
- name: helloworld
chart: ./helloworld
installed: true
Name of helmchart
Set true for install

https://jhooq.com/helmfile-manage-helmchart/

How to use Helmfile for managing helm chart?

Helmfile is an another wrapper working on top of Helm Chart. Just like Helm Chart, Helmfile also uses the YAML for writing the configurations.

But why do you need **Helmfile** if you are working with Kubernetes and Helm Chart?

Here are some key benfits of using **Helmfile** -

1. You can bundle several Helm Charts into a Single Helmfile to manage your kubernetes eco system
2. Helmfile helps you to keep isolation between the different environments(developemnt, staging, production)
3. It can help you to identify the differences between the new changes which you want to apply against the existing running deployment inside kubernetes cluster
4. Helmfile uses the Go Templates which lets you templatify your **Helmfile** and also you can use Sprig Library with functions - requiredEnv, exec, readFile, toYaml, fromYaml, setValueAtPath, get, tpl, required, fetchSecretValue, expandSecretRefs
5. With the help of HelmFile you can deploy multi-tier applications inside kubernetes cluster.

In this guide I am assuming that you have not used the Helmfile before so the first most appropriate step which I think would be to *Installing Helmfile?*

(Note :- But there is a prerequisite before installing the Helmfile, you must install the helmchart)

1. Download the Helmfile from GitHub

2. Rename the file from - **helmfile_linux_amd64** to **helmfile**

```bash
mv helmfile_linux_amd64 helmfile
```

3. Change the permission of the file and make it executable

```bash
chmod 777 helmfile
```

4. Move the file from current location to ***/usr/local/bin***

```
mv helmfile /usr/local/bin
```

5.After moving the file verify the Helmfile installation by running the command - ***$ helmfile --version***

```
$ helmfile --version
helmfile version v0.141.0
```

Helmfile version after installation

https://github.com/roboll/helmfile/releases/download/v0.144.0/helmfile_windows_amd64.exe

```
ubuntu@primary:~$ wget https://github.com/roboll/helmfile/releases/download/v0.144.0/helmfile_linux_arm64
--2023-07-17 22:20:33--  https://github.com/roboll/helmfile/releases/download/v0.144.0/helmfile_linux_arm64
Resolving github.com (github.com)... 140.82.121.3
Connecting to github.com (github.com)|140.82.121.3|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://objects.githubusercontent.com/github-production-release-asset-2e65be/74499101/933a6501-5232-466b-a39f-75697ec7ecb2?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIWNJYAX4CSVEH53A%2F20230717%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20230717T192033Z&X-Amz-Expires=300&X-Amz-Signature=bef381931436f0319fb49f3e8e6fa3b0ff69ad7539044726fbaf520d4b268fdb&X-Amz-SignedHeaders=host&actor_id=0&key_id=0&repo_id=74499101&response-content-disposition=attachment%3B%20filename%3Dhelmfile_linux_arm64&response-content-type=application%2Foctet-stream [following]
--2023-07-17 22:20:34--  https://objects.githubusercontent.com/github-production-release-asset-2e65be/74499101/933a6501-5232-466b-a39f-75697ec7ecb2?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIWNJYAX4CSVEH53A%2F20230717%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20230717T192033Z&X-Amz-Expires=300&X-Amz-Signature=bef381931436f0319fb49f3e8e6fa3b0ff69ad7539044726fbaf520d4b268fdb&X-Amz-SignedHeaders=host&actor_id=0&key_id=0&repo_id=74499101&response-content-disposition=attachment%3B%20filename%3Dhelmfile_linux_arm64&response-content-type=application%2Foctet-stream
Resolving objects.githubusercontent.com (objects.githubusercontent.com)... 185.199.108
Connecting to objects.githubusercontent.com (objects.githubusercontent.com)|185.199.10
HTTP request sent, awaiting response... 200 OK
Length: 45861899 (44M) [application/octet-stream]
Saving to: 'helmfile_linux_arm64'
```

```
ubuntu@primary:~$ ls
Home  common-work-directory  helmfile_linux_arm64  python-flask-rest-api-project  snap
ubuntu@primary:~$
ubuntu@primary:~$ mv helmfile_linux_arm64 helmfile
ubuntu@primary:~$ ls
Home  common-work-directory  helmfile  python-flask-rest-api-project  snap
ubuntu@primary:~$
```

```
ubuntu@primary:~$ ls -lart
total 44860
-rw-r--r-- 1 ubuntu ubuntu      807 Jan  6  2022 .profile
-rw-r--r-- 1 ubuntu ubuntu     3771 Jan  6  2022 .bashrc
-rw-r--r-- 1 ubuntu ubuntu      220 Jan  6  2022 .bash_logout
-rw-rw-r-- 1 ubuntu ubuntu 45861899 Mar 31  2022 helmfile
drwxr-xr-x 3 root   root       4096 Jul  9 14:15 ..
drwx------ 2 ubuntu ubuntu     4096 Jul  9 14:15 .ssh
-rw-r--r-- 1 ubuntu ubuntu        0 Jul  9 14:15 .sudo_as_admin_successful
drwxr-xr-x 1 ubuntu ubuntu       96 Jul  9 14:55 common-work-directory
drwx------ 4 ubuntu ubuntu     4096 Jul  9 15:00 snap
drwxrwxr-x 3 ubuntu ubuntu     4096 Jul  9 15:01 .kube
drwx------ 3 ubuntu ubuntu     4096 Jul 11 21:28 .cache
drwxr-xr-x 4 ubuntu ubuntu     4096 Jul 11 23:28 python-flask-rest-api-project
-rw------- 1 ubuntu ubuntu    14252 Jul 11 23:28 .viminfo
-rw------- 1 ubuntu ubuntu     3249 Jul 11 23:37 .bash_history
drwxr-x--- 1 ubuntu ubuntu     1760 Jul 17 19:15 Home
-rw-rw-r-- 1 ubuntu ubuntu      165 Jul 17 22:20 .wget-hsts
drwxr-x--- 9 ubuntu ubuntu     4096 Jul 17 22:21 .
ubuntu@primary:~$
```

```
ubuntu@primary:~$ chmod 777 helmfile
ubuntu@primary:~$
ubuntu@primary:~$ ls -lart
total 44860
-rw-r--r-- 1 ubuntu ubuntu      807 Jan  6  2022 .profile
-rw-r--r-- 1 ubuntu ubuntu     3771 Jan  6  2022 .bashrc
-rw-r--r-- 1 ubuntu ubuntu      220 Jan  6  2022 .bash_logout
-rwxrwxrwx 1 ubuntu ubuntu 45861899 Mar 31  2022 helmfile
drwxr-xr-x 3 root   root       4096 Jul  9 14:15 ..
drwx------ 2 ubuntu ubuntu     4096 Jul  9 14:15 .ssh
-rw-r--r-- 1 ubuntu ubuntu        0 Jul  9 14:15 .sudo_as_admin_successful
drwxr-xr-x 1 ubuntu ubuntu       96 Jul  9 14:55 common-work-directory
drwx------ 4 ubuntu ubuntu     4096 Jul  9 15:00 snap
drwxrwxr-x 3 ubuntu ubuntu     4096 Jul  9 15:01 .kube
drwx------ 3 ubuntu ubuntu     4096 Jul 11 21:28 .cache
drwxr-xr-x 4 ubuntu ubuntu     4096 Jul 11 23:28 python-flask-rest-api-project
-rw------- 1 ubuntu ubuntu    14252 Jul 11 23:28 .viminfo
-rw------- 1 ubuntu ubuntu     3249 Jul 11 23:37 .bash_history
drwxr-x--- 1 ubuntu ubuntu     1760 Jul 17 19:15 Home
-rw-rw-r-- 1 ubuntu ubuntu      165 Jul 17 22:20 .wget-hsts
drwxr-x--- 9 ubuntu ubuntu     4096 Jul 17 22:21 .
ubuntu@primary:~$
```

#Mv this file usr/local/bin dir

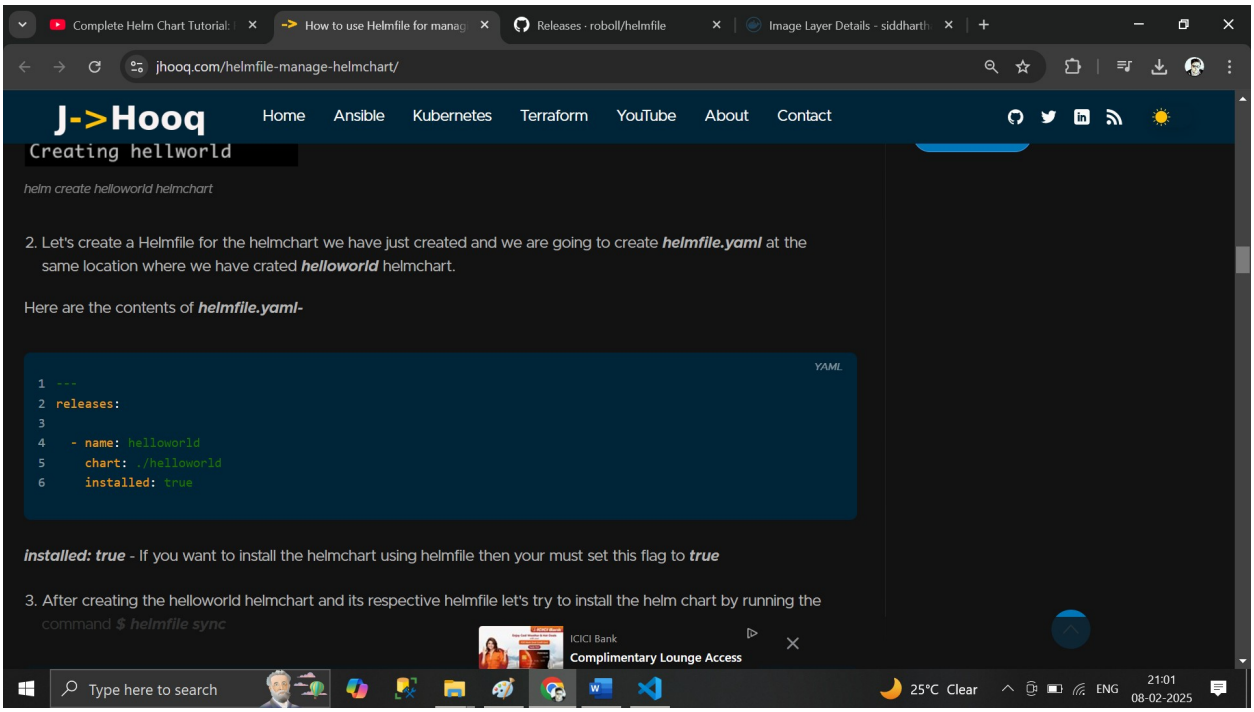
```
ubuntu@primary:~$ mv helmfile /usr/local/bin
mv: cannot move 'helmfile' to '/usr/local/bin/helmfile': Permission denied
ubuntu@primary:~$ sudo mv helmfile /usr/local/bin
ubuntu@primary:~$ ls -lart /usr/local/bin
total 133500
-rwxrwxrwx  1 ubuntu ubuntu 45861899 Mar 31  2022 helmfile
drwxr-xr-x 10 root   root       4096 Jun 16 05:11 ..
-rwxr-xr-x  1 root   root   46202880 Jul  9 14:59 kubectl
-rwxr-xr-x  1 root   root   44630016 Jul  9 15:18 helm
drwxr-xr-x  2 root   root       4096 Jul 17 22:23 .
ubuntu@primary:~$
```

```
ubuntu@primary:~$ helmfile -version
helmfile version v0.144.0
ubuntu@primary:~$
```

```
ubuntu@primary:~$ helm create helloworld
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
Creating helloworld
ubuntu@primary:~$ ls
Home  common-work-directory  helloworld  python-flask-rest-api-project  snap
ubuntu@primary:~$
```

create helmfile .yml

```
ubuntu@primary:~$ vi helmfile.yaml
```




```
---
releases:

  - name: helloworldreleas
    chart: ./helloworld
    installed: true
~
~
```

```
ubuntu@primary:~$ ls
Home  common-work-directory  helloworld  python-flask-rest-api-project  snap
ubuntu@primary:~$ vi helmfile.yaml
ubuntu@primary:~$ ls
Home  common-work-directory  helloworld  helmfile.yaml  python-flask-rest-api-project  snap
ubuntu@primary:~$ cat helmfile.yaml
---
releases:

  - name: helloworldreleas
    chart: ./helloworld
    installed: true
```

```
ubuntu@primary:~$ helmfile sync
Building dependency release=helloworldreleas, chart=helloworld
Affected releases are:
  helloworldreleas (helloworld) UPDATED

Upgrading release=helloworldreleas, chart=helloworld
Release "helloworldreleas" does not exist. Installing it now.
NAME: helloworldreleas
LAST DEPLOYED: Mon Jul 17 22:28:07 2023
NAMESPACE: default
STATUS: deployed
REVISION: 1
NOTES:
1. Get the application URL by running these commands:
  export POD_NAME=$(kubectl get pods --namespace default -l "app.kubernetes.io/name=hellowor
dreleas" -o jsonpath="{.items[0].metadata.name}")
  export CONTAINER_PORT=$(kubectl get pod --namespace default $POD_NAME -o jsonpath="{.spec.
  echo "Visit http://127.0.0.1:8080 to use your application"
  kubectl --namespace default port-forward $POD_NAME 8080:$CONTAINER_PORT

Listing releases matching ^helloworldreleas$
helloworldreleas        default         1               2023-07-17 22:28:07.696860344 +0300
0.1.0  1.16.0


UPDATED RELEASES:
NAME               CHART        VERSION
helloworldreleas   helloworld     0.1.0
```

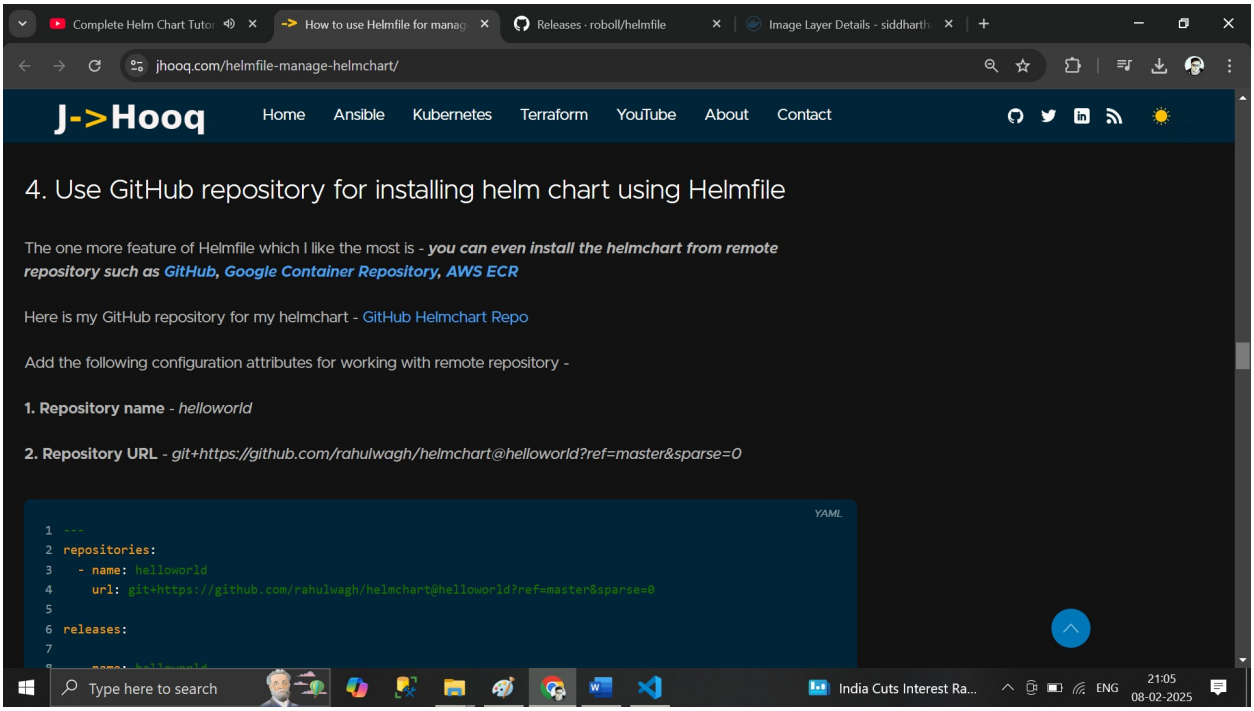
```
ubuntu@primary:~$ helm list -a
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
NAME                NAMESPACE       REVISION        UPDATED                                 STATUS          CHART
       APP VERSION
helloworldreleas    default         1               2023-07-17 22:28:07.696860344 +0300 EEST    deployed        hellowor
0.1.0   1.16.0
ubuntu@primary:~$
```

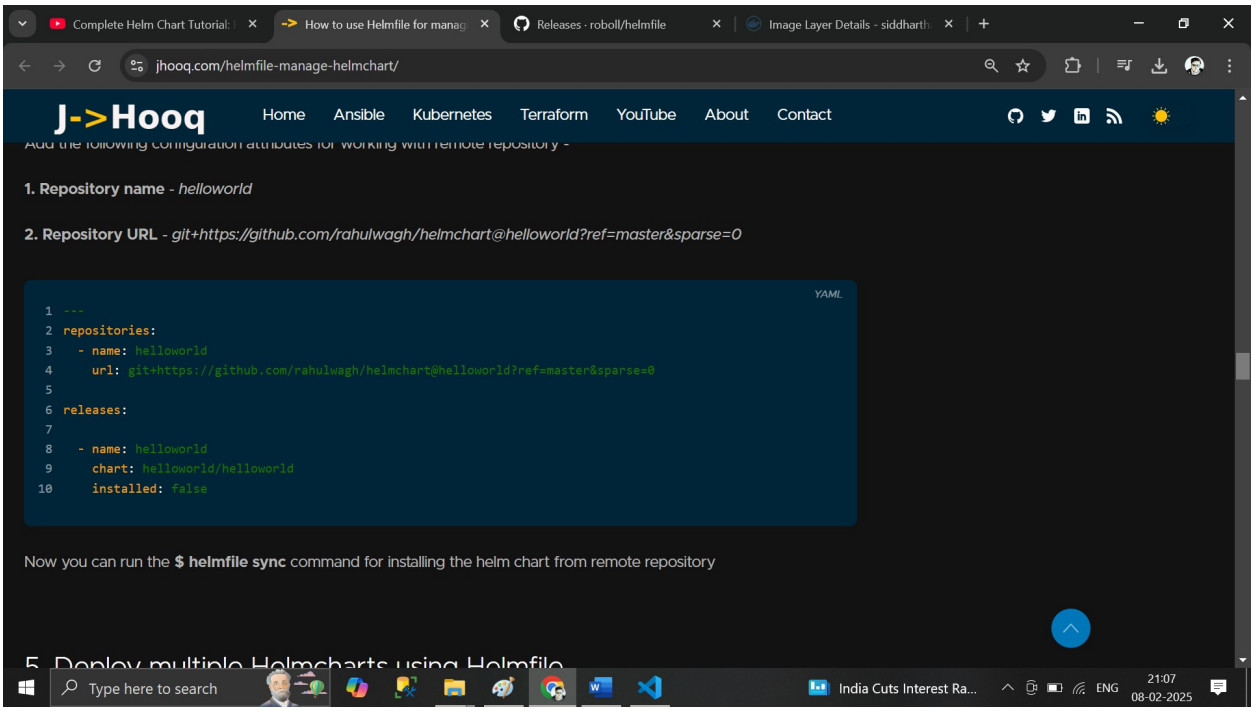
del

```
ubuntu@primary:~$ vi helmfile.yaml
```

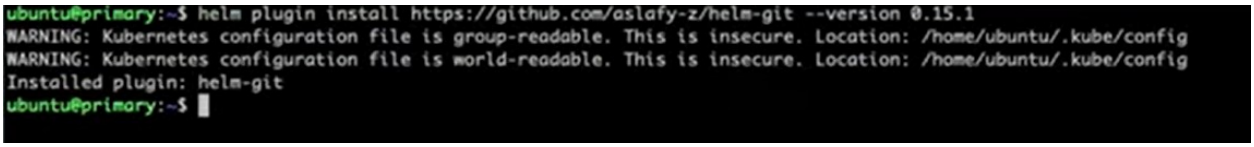
Make it false

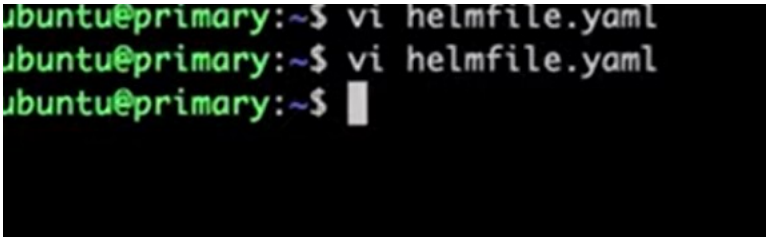

```
ubuntu@primary:~$ helmfile sync
Listing releases matching ^helloworldreleas$
helloworldreleas        default         1               2023-07-17 22:28:07.696860344 +0300 EEST        deployed        helloworl
0.1.0   1.16.0

Affected releases are:
  helloworldreleas (./helloworld) DELETED

Deleting helloworldreleas
release "helloworldreleas" uninstalled


DELETED RELEASES:
NAME
helloworldreleas
ubuntu@primary:~$
```



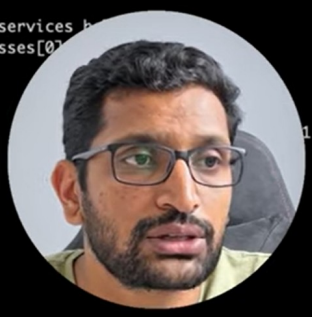


https://github.com/helm/helm





```
ubuntu@primary:~$ helmfile sync
Adding repo helloworld git+https://github.com/rahulwagh/helmchart@helloworld?ref=master&sparse=0
"helloworld" has been added to your repositories

Affected releases are:
  helloworld (helloworld/helloworld) UPDATED

Upgrading release=helloworld, chart=helloworld/helloworld
Release "helloworld" does not exist. Installing it now.
NAME: helloworld
LAST DEPLOYED: Mon Jul 17 22:38:18 2023
NAMESPACE: default
STATUS: deployed
REVISION: 1
NOTES:
1. Get the application URL by running these commands:
  export NODE_PORT=$(kubectl get --namespace default -o jsonpath="{.spec.ports[0].nodePort}" services h
  export NODE_IP=$(kubectl get nodes --namespace default -o jsonpath="{.items[0].status.addresses[0
  echo http://$NODE_IP:$NODE_PORT

Listing releases matching ^helloworld$
helloworld      default         1               2023-07-17 22:38:18.313915226 +0300 EEST                                1.0  1
.16.0


UPDATED RELEASES:
NAME         CHART                   VERSION
helloworld   helloworld/helloworld     0.1.0

ubuntu@primary:~$
```



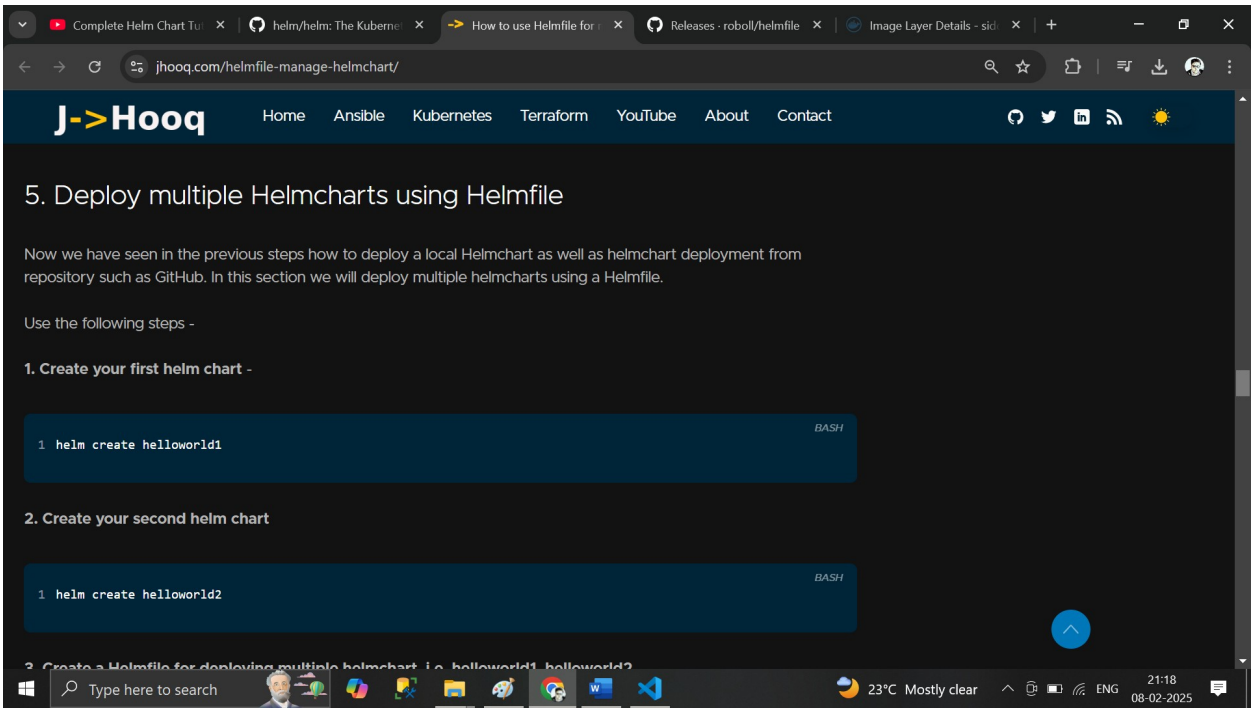
```
ubuntu@primary:~$ helmfile sync
Adding repo helloworld git+https://github.com/rahulwagh/helmchart@helloworld?ref=master&sparse=0
"helloworld" has been added to your repositories

Affected releases are:
  helloworld (helloworld/helloworld) UPDATED

Upgrading release=helloworld, chart=helloworld/helloworld
Release "helloworld" does not exist. Installing it now.
NAME: helloworld
LAST DEPLOYED: Mon Jul 17 22:38:18 2023
NAMESPACE: default
STATUS: deployed
REVISION: 1
NOTES:
1. Get the application URL by running these commands:
  export NODE_PORT=$(kubectl get --namespace default -o jsonpath="{.spec.ports[0].nodePort}"
  export NODE_IP=$(kubectl get nodes --namespace default -o jsonpath="{.items[0].status.addre
  echo http://$NODE_IP:$NODE_PORT

Listing releases matching ^helloworld$
helloworld      default         1               2023-07-17 22:38:18.313915226 +0300 EEST
.16.0


UPDATED RELEASES:
NAME         CHART                   VERSION
helloworld   helloworld/helloworld     0.1.0

ubuntu@primary:~$
```

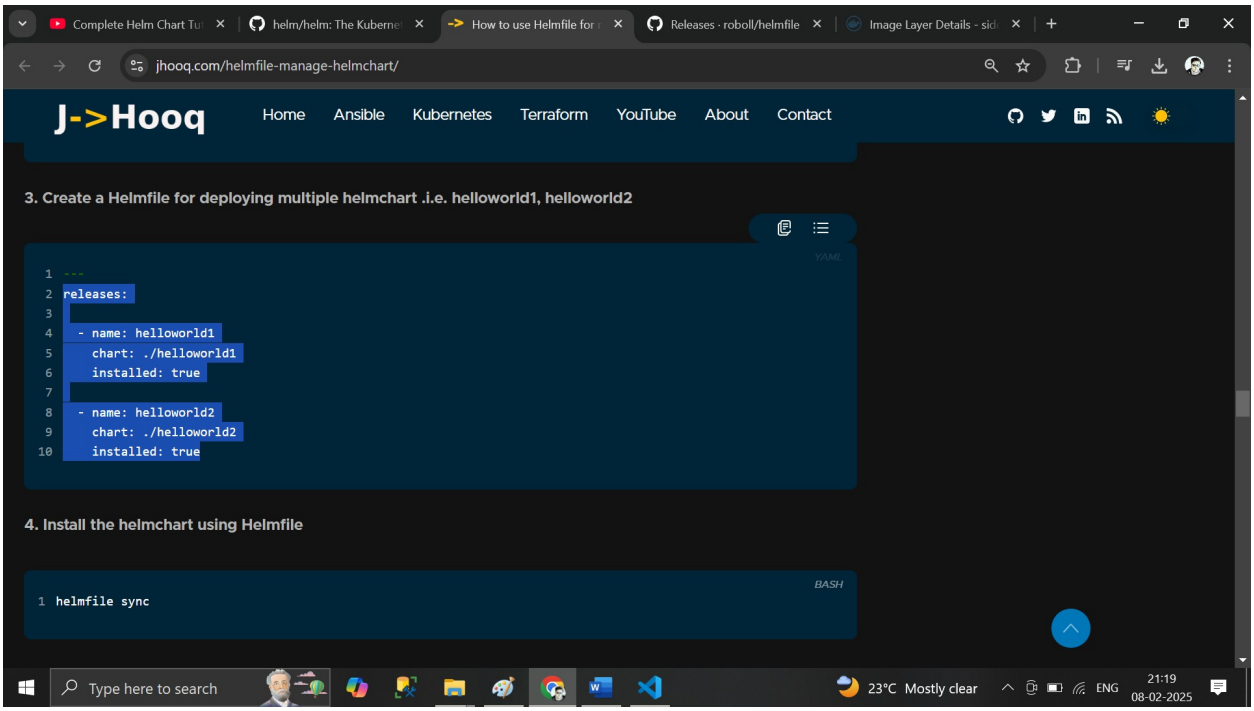
```
ubuntu@primary:~$ helm list -a
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
NAME            NAMESPACE       REVISION        UPDATED                                 STATUS          CHART
PP VERSION
helloworld      default         1               2023-07-17 22:38:18.313915226 +0300 EEST        deployed        hellowo
.16.0
```

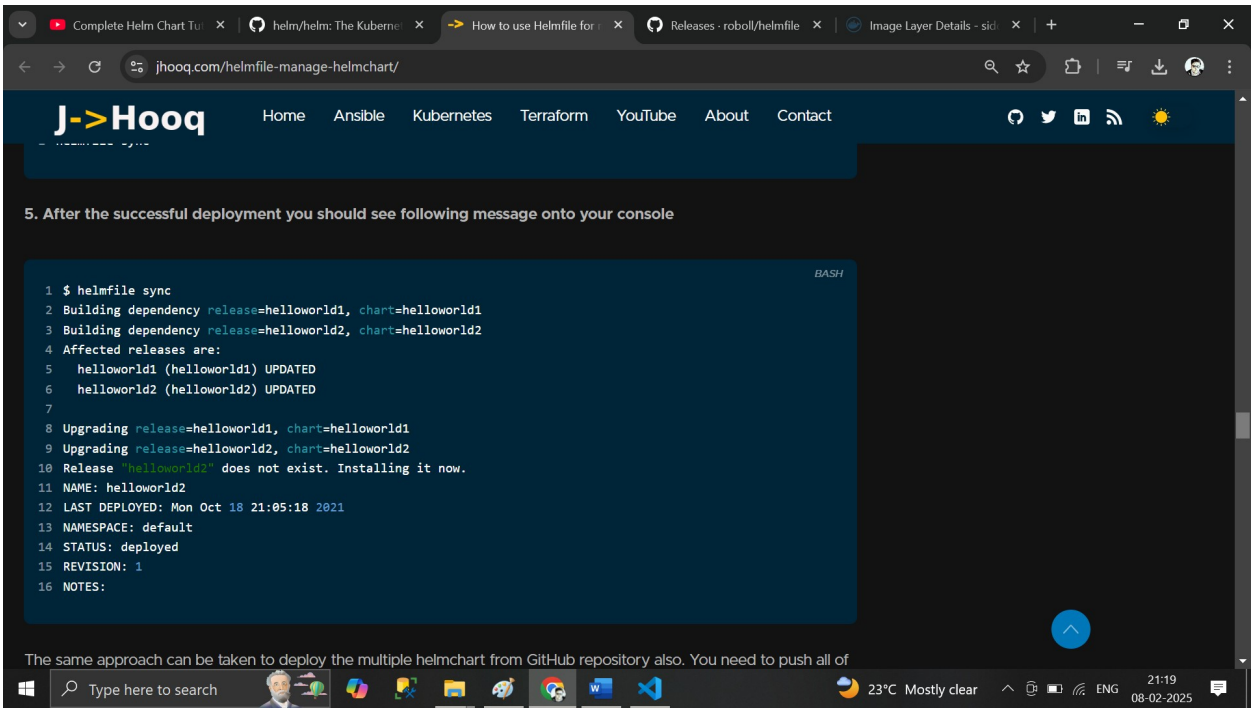
```
repositories:
  - name: helloworld
    url: git+https://github.com/rahulwagh/helmchart@helloworld?ref=master&sparse=0

releases:
  - name: helloworld
    chart: helloworld/helloworld
    installed: false
```

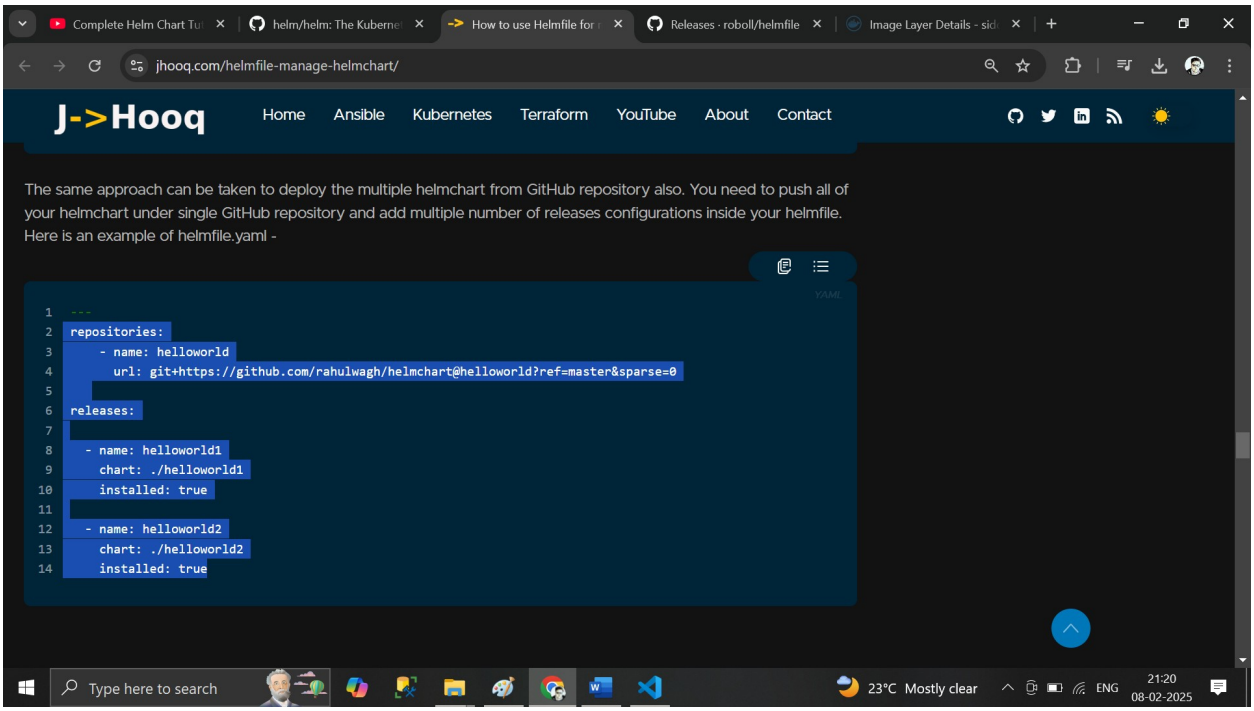
uninstall

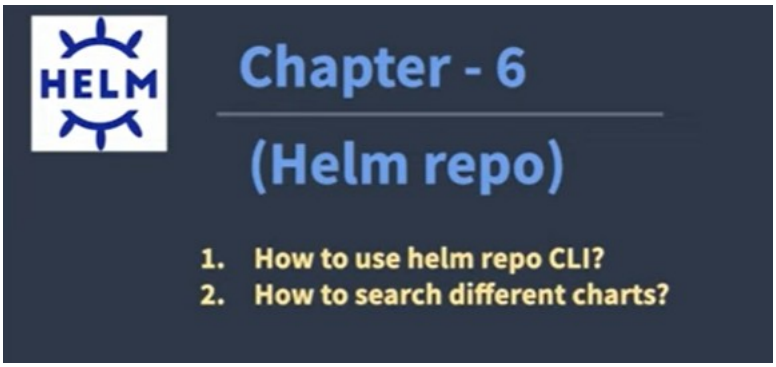
```
ubuntu@primary:~$ helmfile sync
Adding repo helloworld git+https://github.com/rahulwagh/helmchart@helloworld?ref=master&sparse=0
"helloworld" has been added to your repositories

Listing releases matching ^helloworld$
helloworld      default         1               2023-07-17 22:38:18.313915226 +0300 EEST        deployed        helloworld-0.1
.16.0

Affected releases are:
  helloworld (helloworld/helloworld) DELETED

Deleting helloworld
release "helloworld" uninstalled


DELETED RELEASES:
NAME
helloworld
ubuntu@primary:~$
```




Complete Helm Chart Tu
helm/helm: The Kuberne
How to use Helmfile for
Releases · roboll/helmfile
Image Layer Details - sid
jhooq.com/helmfile-manage-helmchart/
J->Hooq
Home
Ansible
Kubernetes
Terraform
YouTube
About
Contact
3. Create a Helmfile for deploying multiple helmchart .i.e. helloworld1, helloworld2
YAML
---
releases:
- name: helloworld1
chart: ./helloworld1
installed: true
- name: helloworld2
chart: ./helloworld2
installed: true
4. Install the helmchart using Helmfile
BASH
helmfile sync
Type here to search
23°C Mostly clear
ENG
21:19
08-02-2025


Complete Helm Chart Tu
helm/helm: The Kuberne
How to use Helmfile for
Releases · roboll/helmfile
Image Layer Details - sid
jhooq.com/helmfile-manage-helmchart/
J->Hooq
Home
Ansible
Kubernetes
Terraform
YouTube
About
Contact
5. After the successful deployment you should see following message onto your console
BASH
$ helmfile sync
Building dependency release=helloworld1, chart=helloworld1
Building dependency release=helloworld2, chart=helloworld2
Affected releases are:
helloworld1 (helloworld1) UPDATED
helloworld2 (helloworld2) UPDATED
Upgrading release=helloworld1, chart=helloworld1
Upgrading release=helloworld2, chart=helloworld2
Release "helloworld2" does not exist. Installing it now.
NAME: helloworld2
LAST DEPLOYED: Mon Oct 18 21:05:18 2021
NAMESPACE: default
STATUS: deployed
REVISION: 1
NOTES:
The same approach can be taken to deploy the multiple helmchart from GitHub repository also. You need to push all of
Type here to search
23°C Mostly clear
ENG
21:19
08-02-2025


Complete Helm Chart Tu
helm/helm: The Kuberne
How to use Helmfile for
Releases · roboll/helmfile
Image Layer Details - sid
jhooq.com/helmfile-manage-helmchart/
J->Hooq
Home
Ansible
Kubernetes
Terraform
YouTube
About
Contact
The same approach can be taken to deploy the multiple helmchart from GitHub repository also. You need to push all of your helmchart under single GitHub repository and add multiple number of releases configurations inside your helmfile. Here is an example of helmfile.yaml -
YAML
---
repositories:
- name: helloworld
url: git+https://github.com/rahulwagh/helmchart@helloworld?ref=master&sparse=0
releases:
- name: helloworld1
chart: ./helloworld1
installed: true
- name: helloworld2
chart: ./helloworld2
installed: true
Type here to search
23°C Mostly clear
ENG
21:20
08-02-2025



3.1 Search for the 'wordpress' repo

```bash
helm search hub wordpress
```

After running the above command it should return you with the list of repos available on the Helm Hub.

```bash
URL                                               CHART VERSION  APP VERSION    DESCRIPTION
https://hub.helm.sh/charts/groundhog2k/wordpress  0.1.3          5.5.1-apache   A Helm chart for Wordpress
https://hub.helm.sh/charts/bitnami/wordpress      10.0.3         5.5.3          Web publishing platform for
https://hub.helm.sh/charts/seccurecodebox/old-w... 2.1.0         4.0            Insecure & Outdated Wordpre
https://hub.helm.sh/charts/fasterbytecharts/wor... 0.8.4         v0.8.4         FasterBytes WordPress Opera
https://hub.helm.sh/charts/presslabs/wordpress-... 0.10.5        0.10.5         Presslabs WordPress Operato
https://hub.helm.sh/charts/presslabs/wordpress-... 0.10.3        v0.10.3        A Helm chart for deploying
https://hub.helm.sh/charts/fasterbytecharts/wor... 0.10.2        v0.10.2        A Helm chart for deploying
https://hub.helm.sh/charts/seccurecodebox/wpscan  2.1.0          latest         A Helm chart for the WordPr
https://hub.helm.sh/charts/presslabs/stack        0.10.3         v0.10.3        Open-Source WordPress Infra
https://hub.helm.sh/charts/fasterbytecharts/stack 0.10.2         v0.10.2        Open-Source WordPress Infra
```

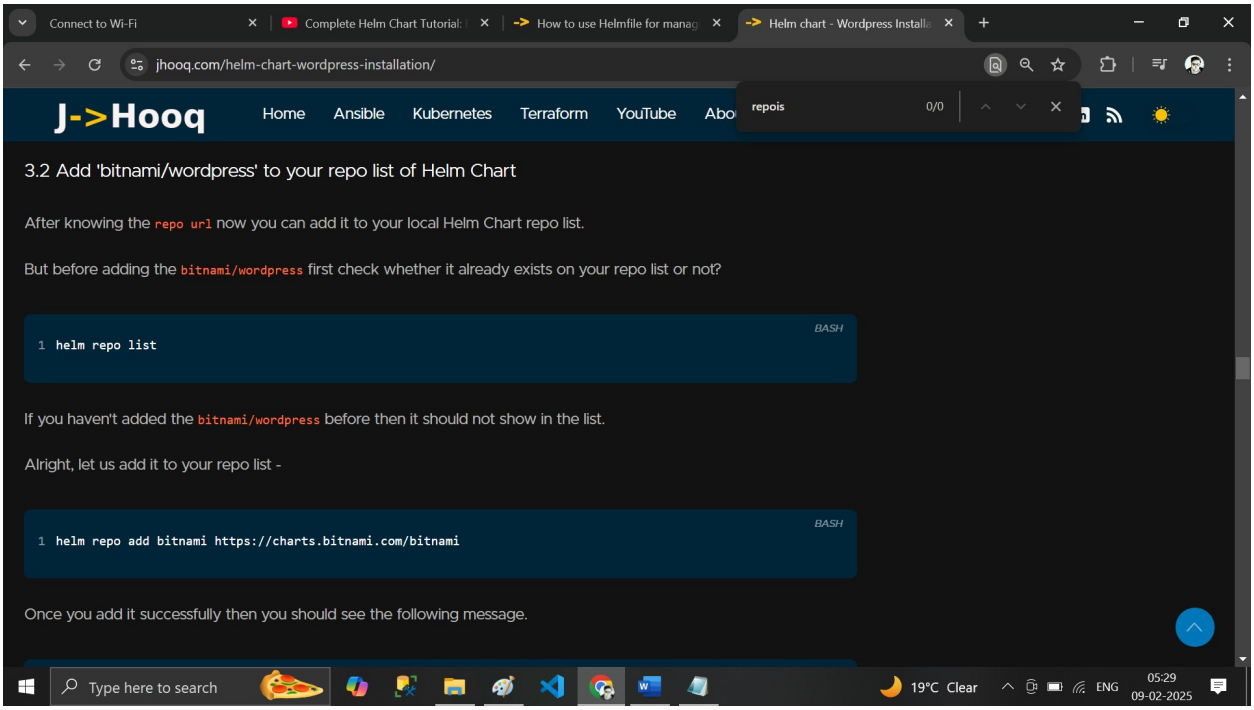
If you look carefully at the results then we are interested in `https://hub.helm.sh/charts/bitnami/wordpress`.

In case if the URL is too long to see then you can put `--max-col-width=0`, so that you can view the complete URL

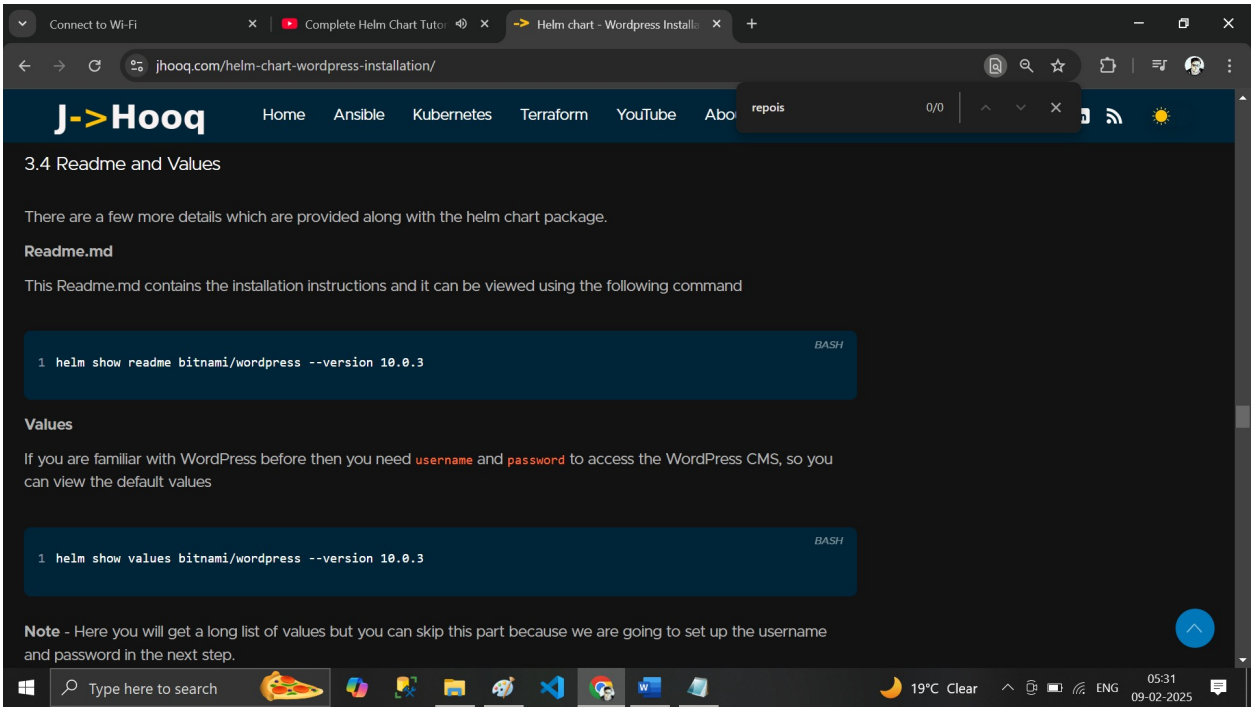
```bash
helm search hub wordpress  --max-col-width=0
```

```
is is my wordpress package

ttps://artifacthub.io/packages/helm/bitnami-aks/wordpress                    15.2.13          6.1.0            W
rdPress is the world's most popular blogging and content management platform. Powerful yet simple, everyone from stu
ents to global corporations use it to build beautiful, functional websites.
ttps://artifacthub.io/packages/helm/shubham-wordpress/wordpress              0.1.0            1.16.0           A
Helm chart for Kubernetes

ttps://artifacthub.io/packages/helm/bitnami/wordpress                        16.1.27          6.2.2            W
rdPress is the world's most popular blogging and content management platform. Powerful yet simple, everyone from stu
ents to global corporations use it to build beautiful, functional websites.
ttps://artifacthub.io/packages/helm/truecharts/wordpress                     2.0.39           6.2.2            T
e WordPress rich content management system can utilize plugins, widgets, and themes.
```

```
ubuntu@primary:~$ helm repo list
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
NAME            URL
helloworld      git+https://github.com/rahulwagh/helmchart@helloworld?ref=master&sparse=0
ubuntu@primary:~$
```



```
ubuntu@primary:~$ helm repo add bitnami https://charts.bitnami.com/bitnami
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
"bitnami" has been added to your repositories
ubuntu@primary:~$
ubuntu@primary:~$ helm repo list
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
NAME            URL
helloworld      git+https://github.com/rahulwagh/helmchart@helloworld?ref=master&sparse=0
bitnami         https://charts.bitnami.com/bitnami
ubuntu@primary:~$
```




```
### To 11.0.0

The [Bitnami WordPress](https://github.com/bitnami/containers/tree/main/bitnami/wordpress) image was refactored and
ow the source code is published in GitHub in the `rootfs` folder of the container image.

Compatibility is not guaranteed due to the amount of involved changes, however no breaking changes are expected.

### To 10.0.0

[On November 13, 2020, Helm v2 support was formally finished](https://github.com/helm/charts#status-of-the-project),
this major version is the result of the required changes applied to the Helm Chart to be able to incorporate the dif
erent features added in Helm v3 and to be consistent with the Helm project itself regarding the Helm v2 EOL.

[Learn more about this change and related upgrade considerations](https://docs.bitnami.com/kubernetes/apps/wordpress
administration/upgrade-helm3/).
```

Chapter - 7

(Hooks & Test)

1. What is Helm Hooks ?
2. When to use the Helm Hooks?
3. Demo - Helm Hooks?
4. What is Helm Test?
5. How to use Helm Test?

1. What is Helm Hooks?

Helm hooks can be any k8s resource?

```yaml
apiVersion: batch/v1
kind: Job
metadata:
  name: pi
spec:
  template:
    spec:
      containers:
      - name: pi
        image: perl:5.34.0
```

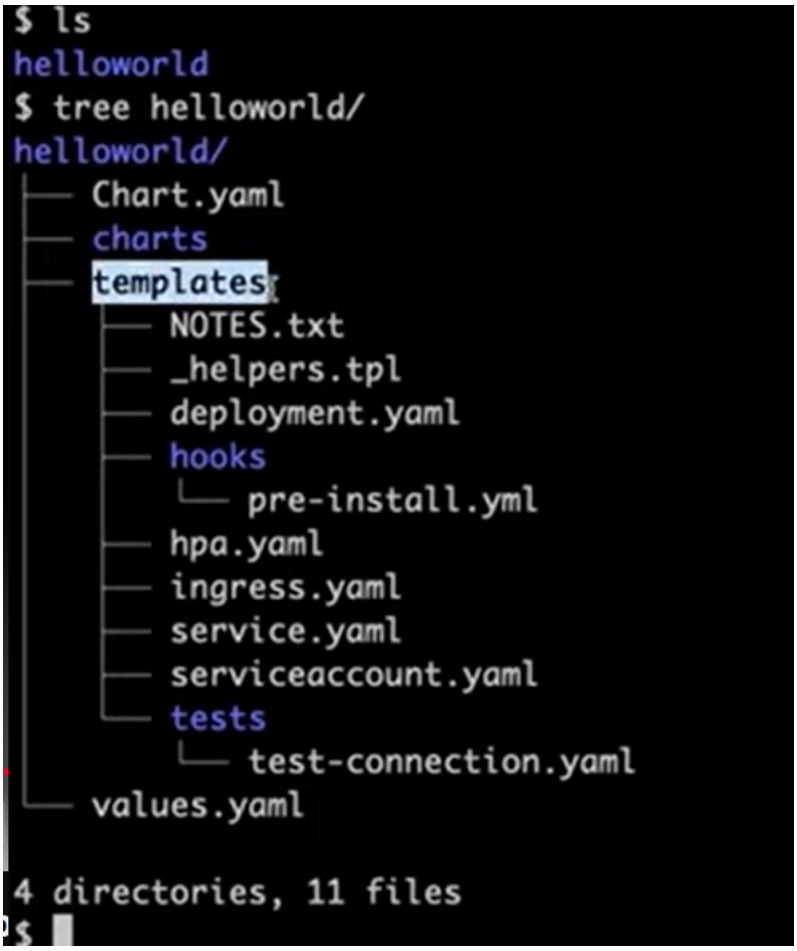
Batch

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  labels:
    app: nginx
spec:
  replicas: 3
  selector:
    matchLabels:
```

Deployment

```yaml
apiVersion: v1
kind: Service
metadata:
  name: my-service
spec:
  selector:
    app.kubernetes.io/name: MyApp
  ports:
    - protocol: TCP
      port: 80
```

Service

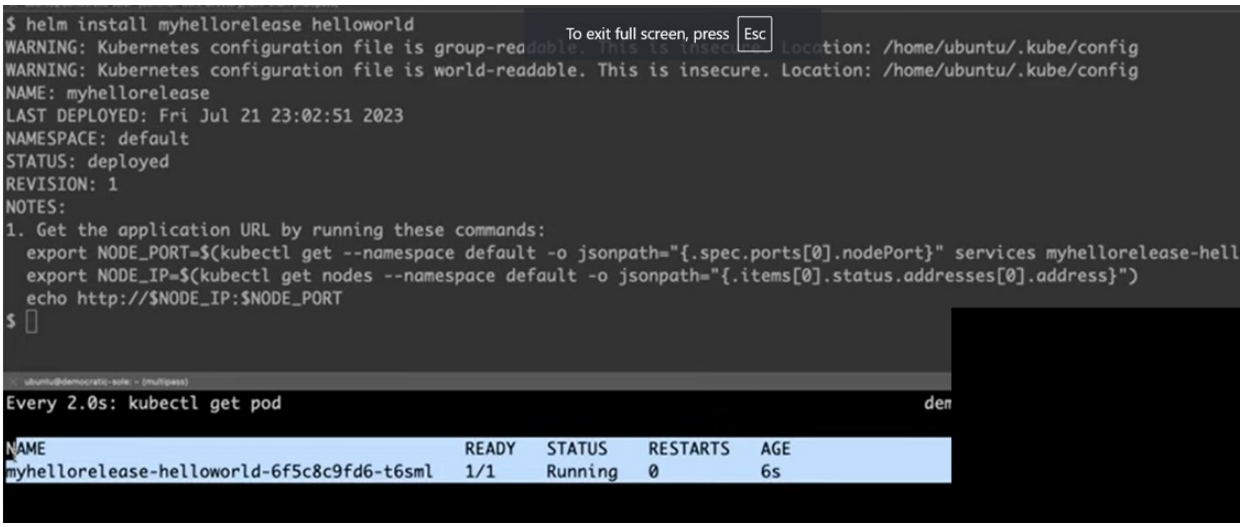

$ ls
helloworld
$ tree helloworld/
helloworld/
├── Chart.yaml
├── charts
├── templates
│   ├── NOTES.txt
│   ├── _helpers.tpl
│   ├── deployment.yaml
│   ├── hooks
│   │   └── pre-install.yml
│   ├── hpa.yaml
│   ├── ingress.yaml
│   ├── service.yaml
│   ├── serviceaccount.yaml
│   └── tests
│   └── test-connection.yaml
└── values.yaml
4 directories, 11 files
$

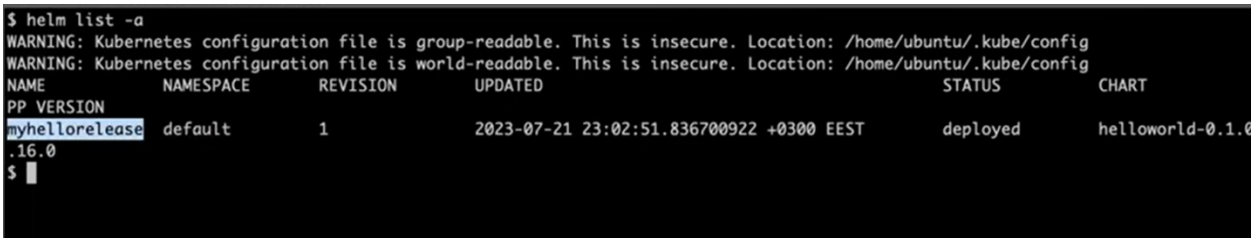
```
apiVersion: batch/v1
kind: Job
metadata:
  name: "{{ include "helloworld.fullname" . }}-pre-install-job-hook"
  labels:
    {{- include "helloworld.labels" . | nindent 4 }}
  annotations:
    "helm.sh/hook": "pre-install"
    "helm.sh/hook-weight": "0"
    "helm.sh/hook-delete-policy": hook-succeeded
spec:
  template:
    spec:
      containers:
      - name: pre-install
        image: busybox
        imagePullPolicy: IfNotPresent
        command: ['sh', '-c', 'echo pre-install Pod is Running ; sleep 10']
      restartPolicy: OnFailure
      terminationGracePeriodSeconds: 0
  backoffLimit: 3
  completions: 1
  parallelism: 1
```

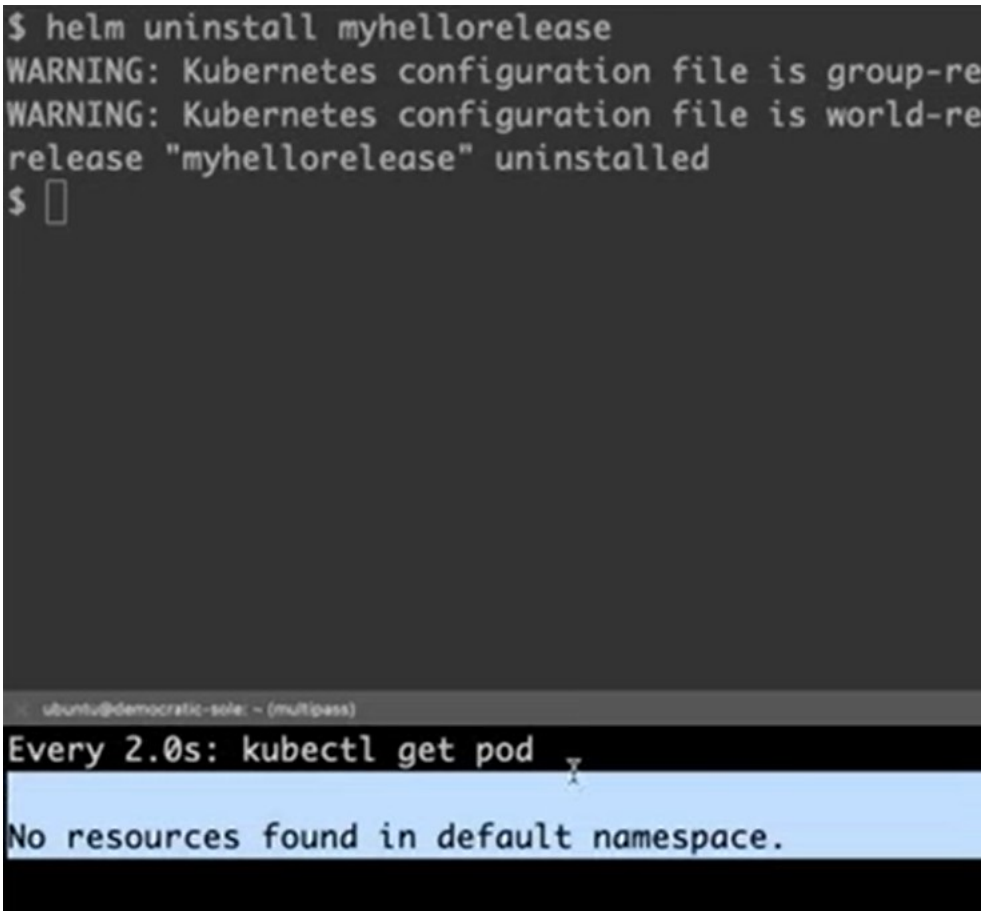
```
ubuntu@democratic-sole: ~ (multipass)
Every 2.0s: kubectl get pod

NAME                                        READY   STATUS      RESTARTS   AGE
myhellorelease-helloworld-test-connection   0/1     Completed   0          24h
```

```
$ helm install myhellorelease helloworld
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
```

```
Every 2.0s: kubectl get pod                                                    democ

NAME                                                READY   STATUS    RESTARTS   AGE
myhellorelease-helloworld-pre-install-job-hookkm28v   1/1     Running   0          7s
```


To exit full screen, press Esc
$ helm install myhellorelease helloworld
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
NAME: myhellorelease
LAST DEPLOYED: Fri Jul 21 23:02:51 2023
NAMESPACE: default
STATUS: deployed
REVISION: 1
NOTES:
1. Get the application URL by running these commands:
export NODE_PORT=$(kubectl get --namespace default -o jsonpath="{.spec.ports[0].nodePort}" services myhellorelease-hell
export NODE_IP=$(kubectl get nodes --namespace default -o jsonpath="{.items[0].status.addresses[0].address}")
echo http://$NODE_IP:$NODE_PORT
$
Every 2.0s: kubectl get pod
NAME READY STATUS RESTARTS AGE
myhellorelease-helloworld-6f5c8c9fd6-t6sml 1/1 Running 0 6s


$ helm list -a
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
NAME NAMESPACE REVISION UPDATED STATUS CHART
PP VERSION
myhellorelease default 1 2023-07-21 23:02:51.836700922 +0300 EEST deployed helloworld-0.1.0
.16.0
$


$ helm uninstall myhellorelease
WARNING: Kubernetes configuration file is group-re
WARNING: Kubernetes configuration file is world-re
release "myhellorelease" uninstalled
$
Every 2.0s: kubectl get pod
No resources found in default namespace.

```
$ tree helloworld
helloworld
├── Chart.yaml
├── charts
├── templates
│   ├── NOTES.txt
│   ├── _helpers.tpl
│   ├── deployment.yaml
│   ├── hooks
│   │   └── pre-install.yml
│   ├── hpa.yaml
│   ├── ingress.yaml
│   ├── service.yaml
│   ├── serviceaccount.yaml
│   └── tests
│       └── test-connection.yaml
└── values.yaml

4 directories, 11 files
$
```

```
$ cat helloworld/templates/tests/test-connection.yaml
apiVersion: v1
kind: Pod
metadata:
  name: "{{ include "helloworld.fullname" . }}-test-connection"
  labels:
    {{- include "helloworld.labels" . | nindent 4 }}
  annotations:
    "helm.sh/hook": test
spec:
  containers:
    - name: wget
      image: busybox
      command: ['wget']
      args: ['{{ include "helloworld.fullname" . }}:{{ .Values.service.port }}']
  restartPolicy: Never
$
```

```
$ helm list -a
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
NAME    NAMESPACE       REVISION        UPDATED STATUS  CHART   APP VERSION
$
```

```
$ helm install myhellorelease helloworld
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
NAME: myhellorelease
LAST DEPLOYED: Fri Jul 21 23:12:01 2023
NAMESPACE: default
STATUS: deployed
REVISION: 1
NOTES:
1. Get the application URL by running these commands:
  export NODE_PORT=$(kubectl get --namespace default -o jsonpath="{.spec.ports[0].nodePort}" services myhellorelease-hellowo
  export NODE_IP=$(kubectl get nodes --namespace default -o jsonpath="{.items[0].status.addresses[0].address}")
  echo http://$NODE_IP:$NODE_PORT
$
```

```
$ kubectl get service
NAME                         TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)        AGE
kubernetes                   ClusterIP   10.152.183.1     <none>        443/TCP        25h
myhellorelease-helloworld    NodePort    10.152.183.217   <none>        80:31836/TCP   23s
$
```

```
$ hostname -I
192.168.64.7 10.1.242.64 fd16:8590:7284:4e59:5054:ff:fe59:58f5
$
```

192.168.64.7:31836

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

```
$ helm list -a
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
NAME            NAMESPACE       REVISION        UPDATED                                         STATUS          CHART
PP VERSION
myhellorelease  default         1               2023-07-21 23:12:01.304746807 +0300 EEST        deployed        helloworld-
.16.0
$
$ helm test myhellorelease
WARNING: Kubernetes configuration file is group-readable. This is insecure. Location: /home/ubuntu/.kube/config
WARNING: Kubernetes configuration file is world-readable. This is insecure. Location: /home/ubuntu/.kube/config
NAME: myhellorelease
LAST DEPLOYED: Fri Jul 21 23:12:01 2023
NAMESPACE: default
STATUS: deployed
REVISION: 1
TEST SUITE:     myhellorelease-helloworld-test-connection
Last Started:   Fri Jul 21 23:14:19 2023
Last Completed: Fri Jul 21 23:14:22 2023
Phase:          Succeeded
NOTES:
1. Get the application URL by running these commands:
  export NODE_PORT=$(kubectl get --namespace default -o jsonpath="{.spec.ports[0].nodePort}"
  export NODE_IP=$(kubectl get nodes --namespace default -o jsonpath="{.items[0].status.addr
  echo http://$NODE_IP:$NODE_PORT
$
```