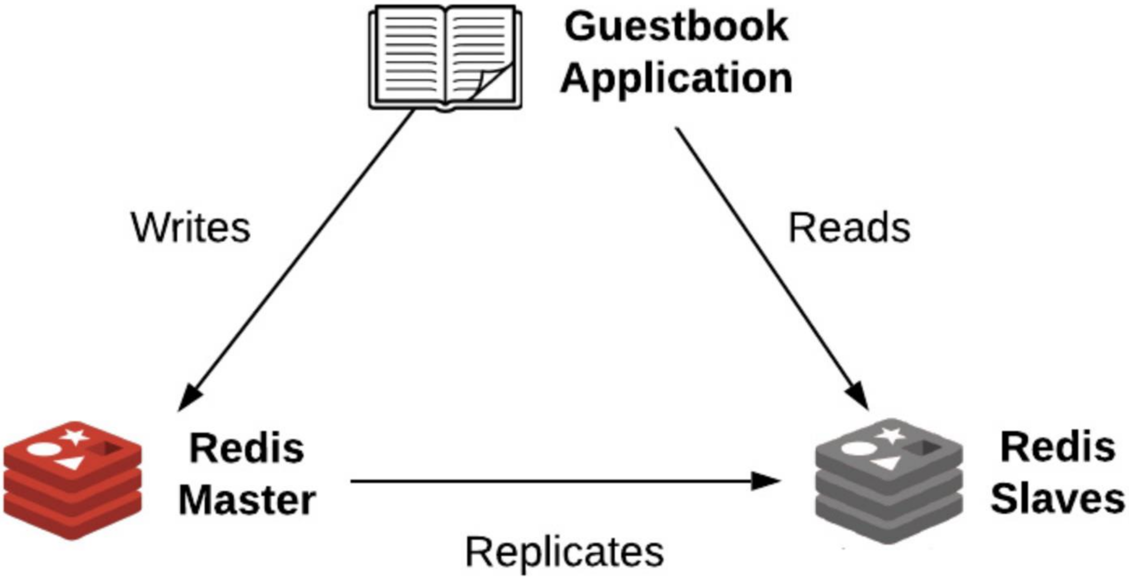
How to add dependency in Helm

we will be deploying this guestbook application using helm and add the `Redis` as a dependency.



Prerequisite

1. Kubernetes Cluster Setup
2. Clone this git repo

Setup a helm project

```
helm create guestbook
rm -rf guestbook/templates/tests
```

Adding a Redis Chart dependency

Chart dependencies are used to install other charts' resources that a Helm chart may depend on.

In this example, we are using Redis as a database so we to need add this as a dependency.

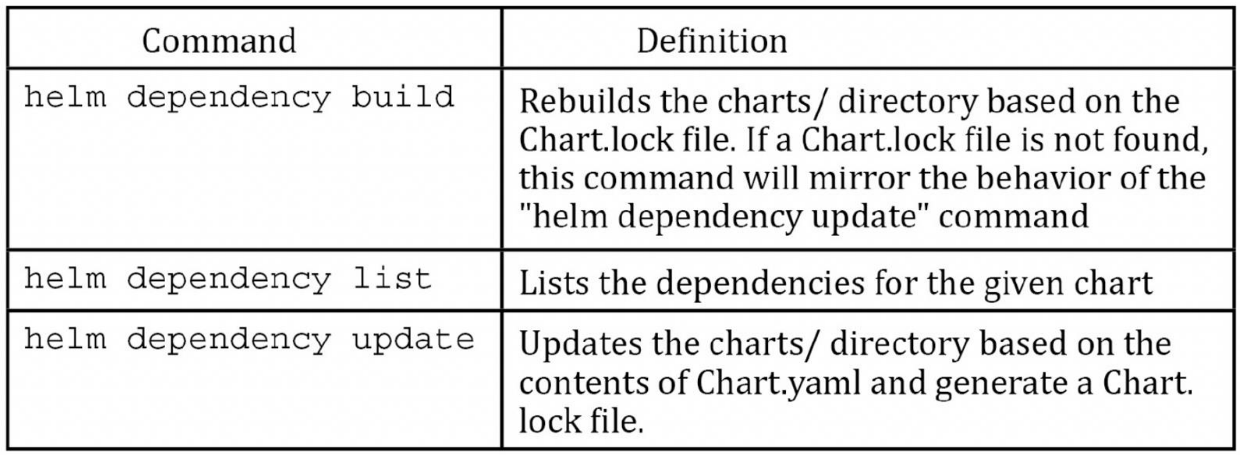
First we can search the charts for `redis`

```
helm search hub redis
```

Now we will add the dependency section in the `Charts.yaml` file

```
dependencies:
  - name: redis
    version: 12.7.x
    repository: https://charts.bitnami.com/bitnami
```

How to download this dependency?

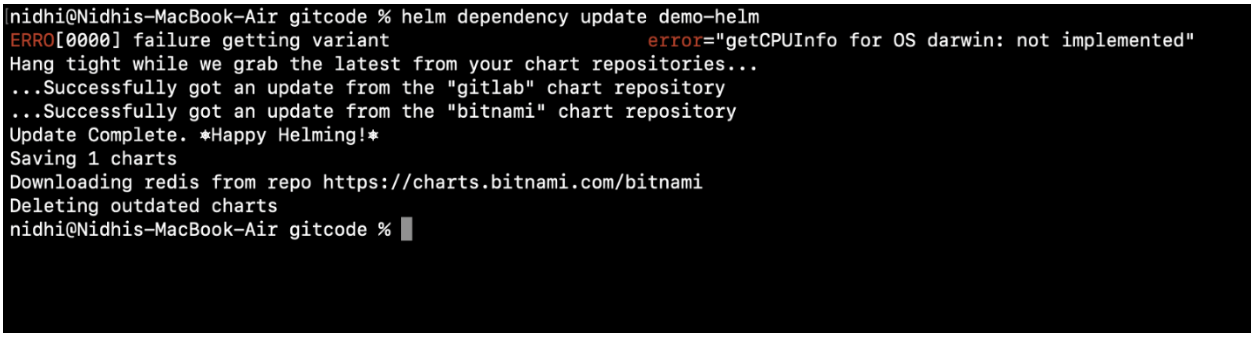


Command	Definition
`helm dependency build`	Rebuilds the charts/ directory based on the Chart.lock file. If a Chart.lock file is not found, this command will mirror the behavior of the "helm dependency update" command
`helm dependency list`	Lists the dependencies for the given chart
`helm dependency update`	Updates the charts/ directory based on the contents of Chart.yaml and generate a Chart. lock file.

When downloading a dependency for the first time, you should use the **helm dependency update** command.

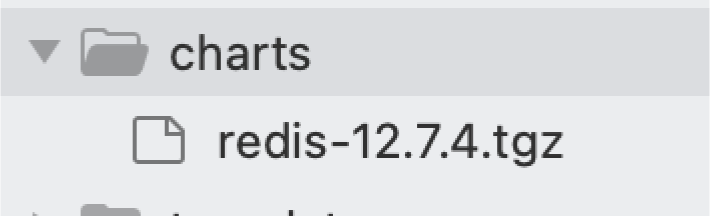
This command will download your dependency to the **charts/** directory and will generate the **Chart.lock** file, which specifies metadata about the chart that was downloaded.

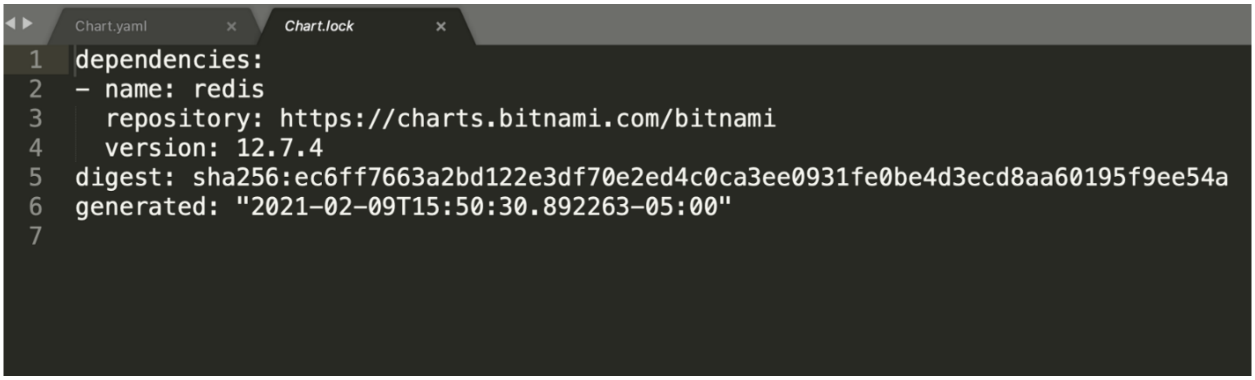
```
helm dependency update guestbook
```



```
nidhi@Nidhis-MacBook-Air gitcode % helm dependency update demo-helm
ERRO[0000] failure getting variant                        error="getCPUInfo for OS darwin: not implemented"
Hang tight while we grab the latest from your chart repositories...
...Successfully got an update from the "gitlab" chart repository
...Successfully got an update from the "bitnami" chart repository
Update Complete. *Happy Helming!*
Saving 1 charts
Downloading redis from repo https://charts.bitnami.com/bitnami
Deleting outdated charts
nidhi@Nidhis-MacBook-Air gitcode %
```

Now you should see that it has downloaded the dependency and also updated the Chart.lock file

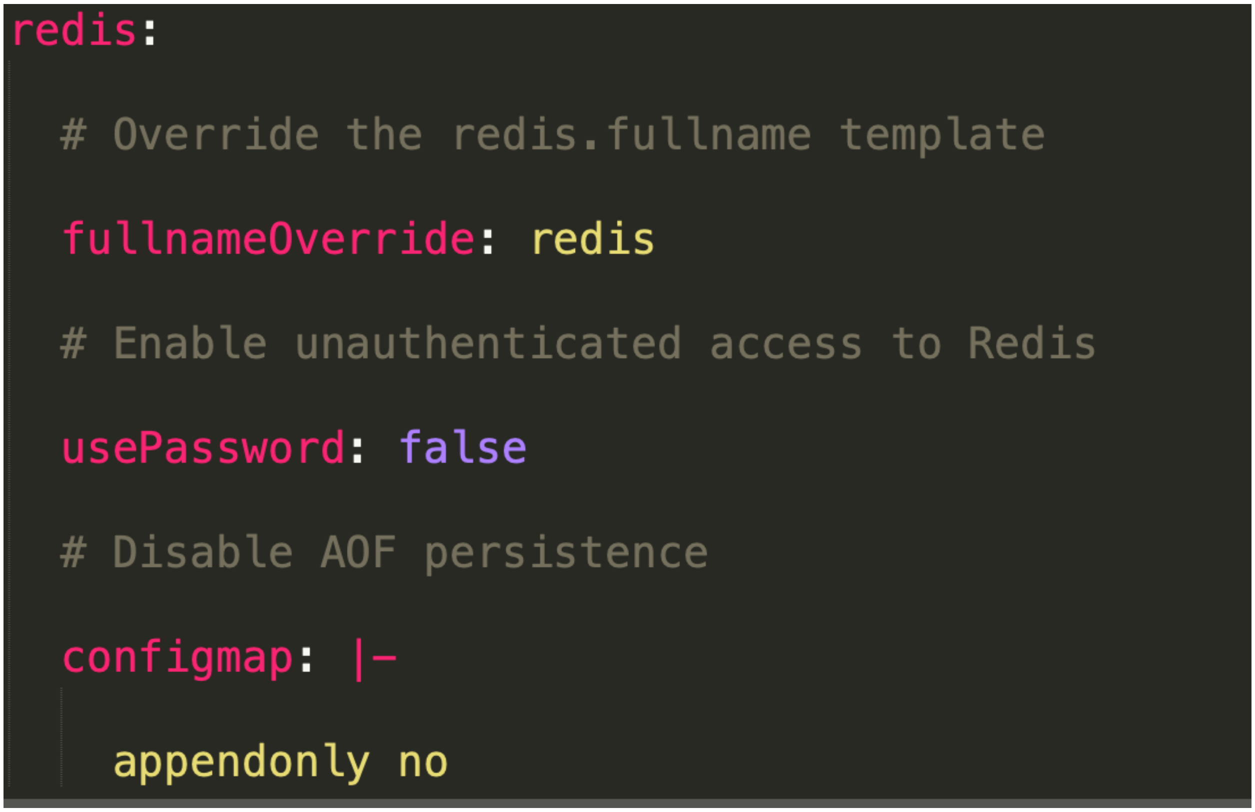




```
dependencies:
- name: redis
  repository: https://charts.bitnami.com/bitnami
  version: 12.7.4
digest: sha256:ec6ff7663a2bd122e3df70e2ed4c0ca3ee0931fe0be4d3ecd8aa60195f9ee54a
generated: "2021-02-09T15:50:30.892263-05:00"
```

ADDING VALUES TO CONFIGURE THE REDIS CHART

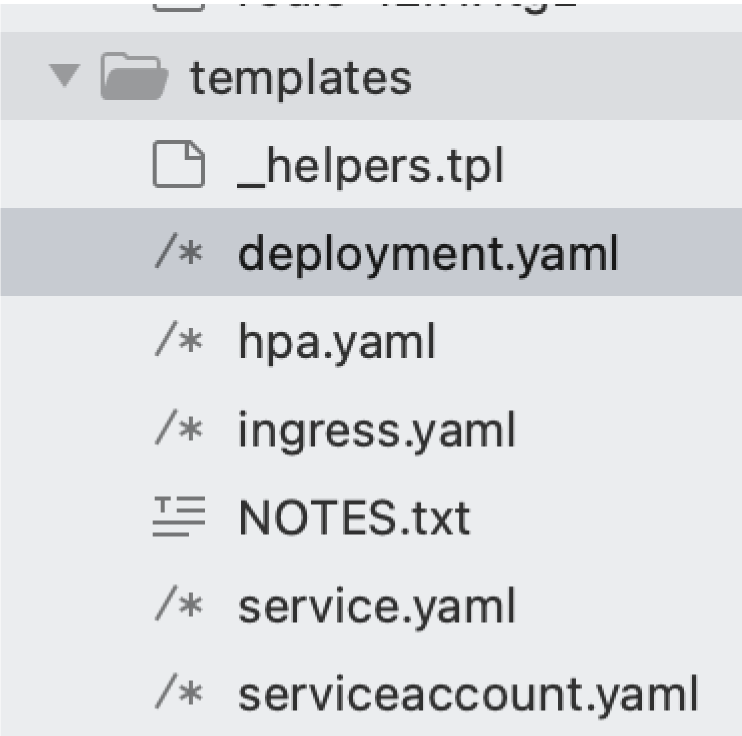
You can override the default values of Redis chart using values.yaml file



```
redis:
  # Override the redis.fullname template
  fullnameOverride: redis
  # Enable unauthenticated access to Redis
  usePassword: false
  # Disable AOF persistence
  configmap: |-
    appendonly no
```

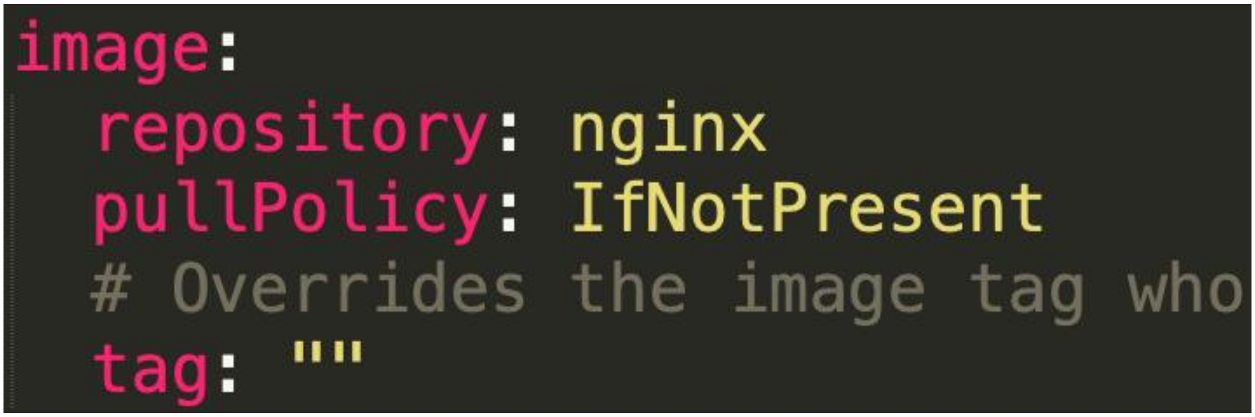
Modify frontend application

By default it has created these templates



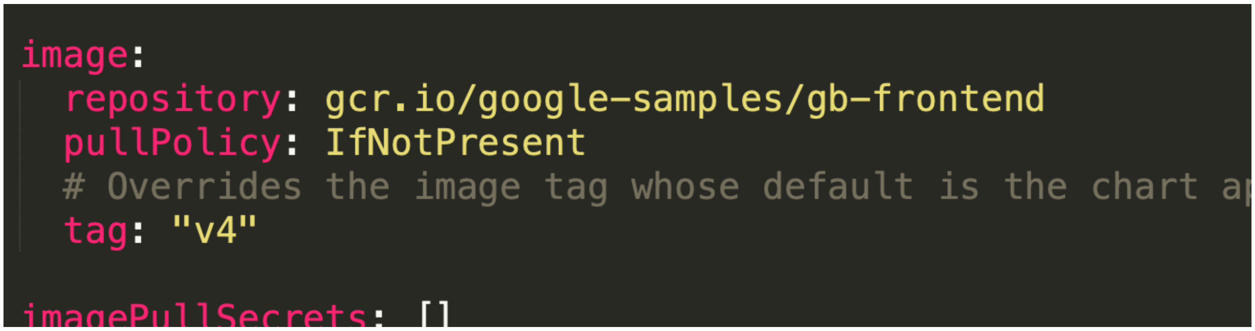
- **deployment.yaml**: Used to deploy the Guestbook application to Kubernetes.
- **ingress.yaml**: Provides one option to access the Guestbook application from outside the Kubernetes cluster.
- **serviceaccount.yaml**: Used to create a dedicated **serviceaccount** for the Guestbook application.
- **service.yaml**: Used to load-balance between multiple instances of the Guestbook application. Can also provide an option to access the Guestbook application from outside the Kubernetes cluster.
- **_helpers.tp**: Provides a set of common templates used throughout the Helm chart.
- **NOTES.txt**: Provides a set of instructions used to access the application after it is installed.

In the `values.yaml` file you can change the default image to the one which you want to use for your application



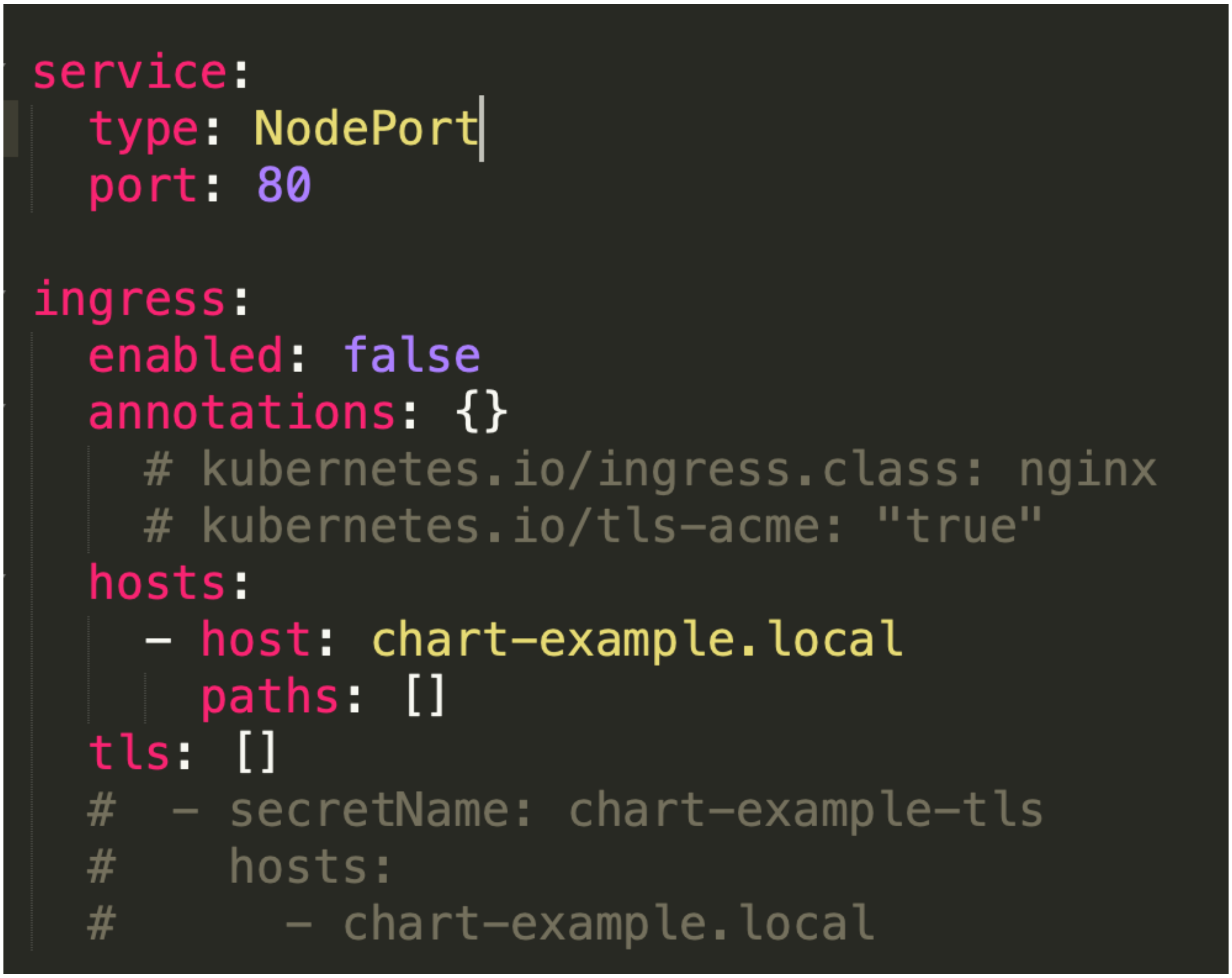
```
image:
  repository: nginx
  pullPolicy: IfNotPresent
  # Overrides the image tag who
  tag: ""
```

Now we will update this image as per our requirement



```
image:
  repository: gcr.io/google-samples/gb-frontend
  pullPolicy: IfNotPresent
  # Overrides the image tag whose default is the chart ap
  tag: "v4"

imagePullSecrets: []
```

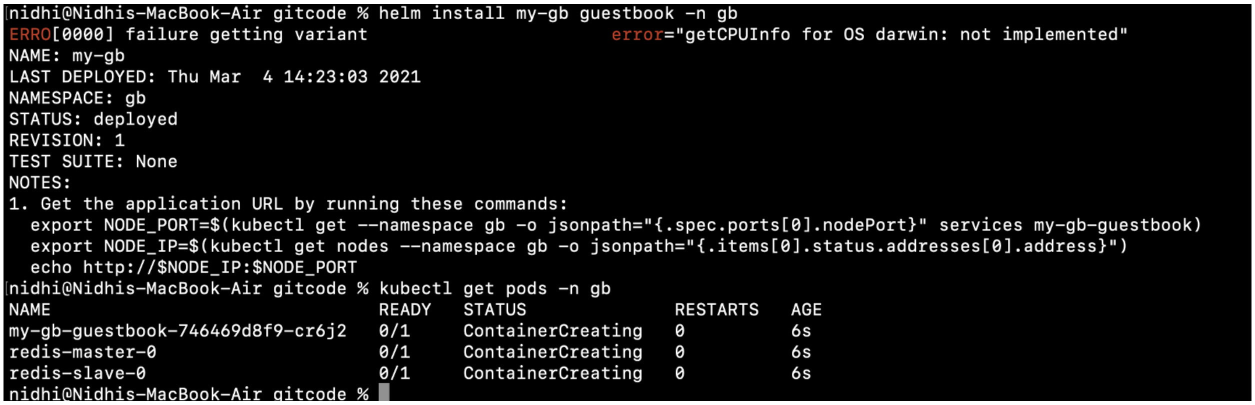
Also, change the NodePort so that we can browse the application



```yaml
service:
  type: NodePort
  port: 80

ingress:
  enabled: false
  annotations: {}
    # kubernetes.io/ingress.class: nginx
    # kubernetes.io/tls-acme: "true"
  hosts:
    - host: chart-example.local
      paths: []
  tls: []
  #  - secretName: chart-example-tls
  #    hosts:
  #      - chart-example.local
```

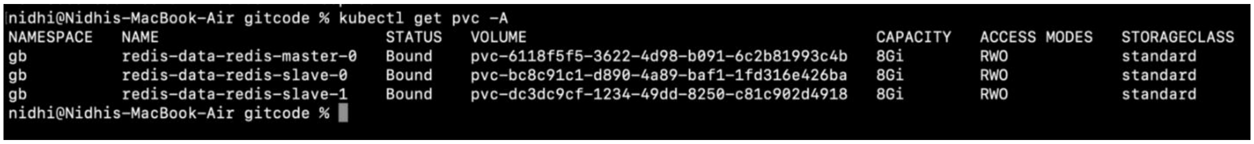
Install the chart

```
kubectl create ns gb( Create the namespace first)

helm install my-gb guestbook -n gb
kubectl get pods -n gb
```



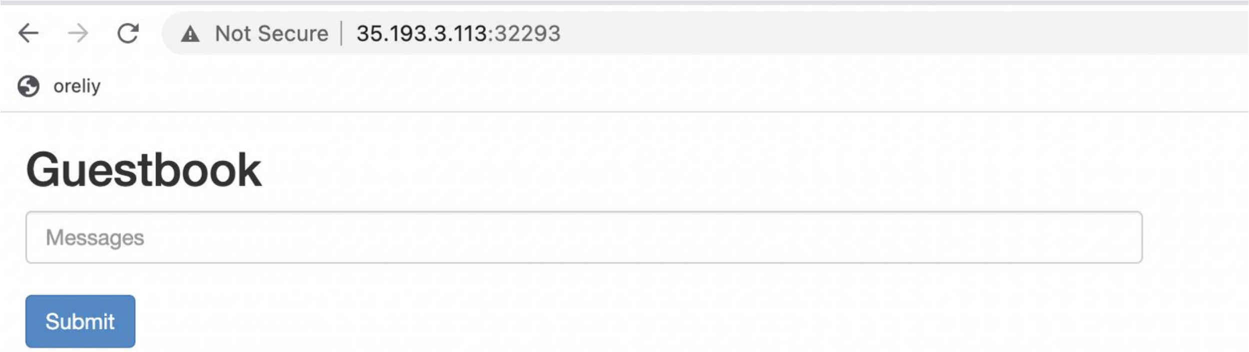
```
nidhi@Nidhis-MacBook-Air gitcode % helm install my-gb guestbook -n gb
ERRO[0000] failure getting variant                       error="getCPUInfo for OS darwin: not implemented"
NAME: my-gb
LAST DEPLOYED: Thu Mar  4 14:23:03 2021
NAMESPACE: gb
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
1. Get the application URL by running these commands:
  export NODE_PORT=$(kubectl get --namespace gb -o jsonpath="{.spec.ports[0].nodePort}" services my-gb-guestbook)
  export NODE_IP=$(kubectl get nodes --namespace gb -o jsonpath="{.items[0].status.addresses[0].address}")
  echo http://$NODE_IP:$NODE_PORT
nidhi@Nidhis-MacBook-Air gitcode % kubectl get pods -n gb
NAME                               READY   STATUS              RESTARTS   AGE
my-gb-guestbook-746469d8f9-cr6j2   0/1     ContainerCreating   0          6s
redis-master-0                     0/1     ContainerCreating   0          6s
redis-slave-0                      0/1     ContainerCreating   0          6s
nidhi@Nidhis-MacBook-Air gitcode %
```

It has created `PVC` also as shown below



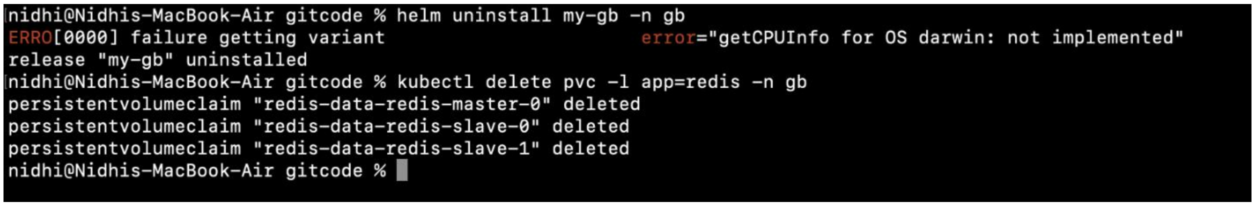
```
nidhi@Nidhis-MacBook-Air gitcode % kubectl get pvc -A
NAMESPACE   NAME                         STATUS   VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS
gb          redis-data-redis-master-0    Bound    pvc-6118f5f5-3622-4d98-b091-6c2b81993c4b   8Gi        RWO            standard
gb          redis-data-redis-slave-0     Bound    pvc-bc8c91c1-d890-4a89-baf1-1fd316e426ba   8Gi        RWO            standard
gb          redis-data-redis-slave-1     Bound    pvc-dc3dc9cf-1234-49dd-8250-c81c902d4918   8Gi        RWO            standard
nidhi@Nidhis-MacBook-Air gitcode %
```

```
export NODE_PORT=$(kubectl get --namespace helm1 -o jsonpath="{.spec.ports[0].nodePort}" services frontend-demo-helm)

export NODE_IP=$(kubectl get nodes --namespace helm1 -o jsonpath="{.items[0].status.addresses[0].address}")

echo http://$NODE_IP:$NODE_PORT
```



Uninstall the chart

```
helm uninstall my-gb -n gb
kubectl delete pvc -l app=redis -n gb
```



```
nidhi@Nidhis-MacBook-Air gitcode % helm uninstall my-gb -n gb
ERRO[0000] failure getting variant                       error="getCPUInfo for OS darwin: not implemented"
release "my-gb" uninstalled
nidhi@Nidhis-MacBook-Air gitcode % kubectl delete pvc -l app=redis -n gb
persistentvolumeclaim "redis-data-redis-master-0" deleted
persistentvolumeclaim "redis-data-redis-slave-0" deleted
persistentvolumeclaim "redis-data-redis-slave-1" deleted
nidhi@Nidhis-MacBook-Air gitcode %
```