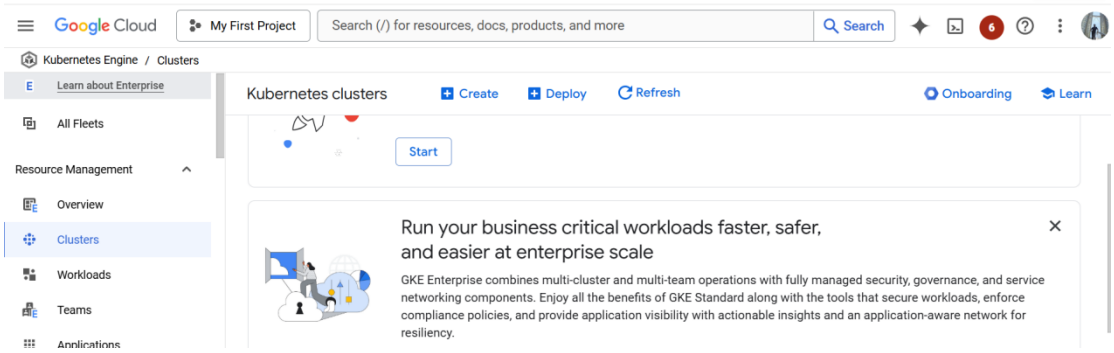
- Create the **google cloud console** free account
- It is a two step process
- It is deducting the 2 rupees from your account and it will give the 330$ free credit points.

**NOTE: Don’t active the full account**

- Once the account is created u can login to google cloud console




- NOW CREATE THE KUBERNETES CLUSTER

Open the cloud shell


To see the cluster list run the below command

Gcloud container clusters list  ( no clusters are there)

```
sarvesh_ambala@cloudshell:~ (hidden-solstice-454006-n0)$ gcloud container clusters list
sarvesh_ambala@cloudshell:~ (hidden-solstice-454006-n0)$
```

You create the cluster with below command

gcloud container clusters create my-cluster --zone us-central1-a

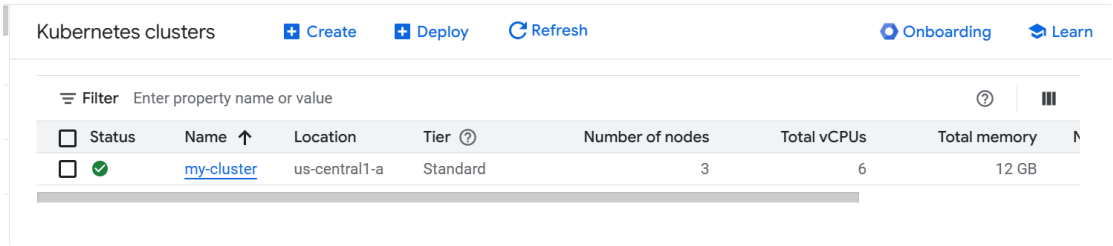
Cluster creation is taking 5 to 10 mints time

```
sarvesh_ambala@cloudshell:~ (hidden-solstice-454006-n0)$ gcloud container clusters create my-cluster --zone us-central1-a
Note: The Kubelet readonly port (10255) is now deprecated. Please update your workloads to use the recommended alternatives. See https://cloud.google.com/kube
rnetes-engine/docs/how-to/disable-kubelet-readonly-port for ways to check usage and for migration instructions.
Note: Your Pod address range (`--cluster-ipv4-cidr`) can accommodate at most 1008 node(s).
Creating cluster my-cluster in us-central1-a...working.
```

Once the cluster is created u can see the below message automatically

```
To inspect the contents of your cluster, go to: https://console.cloud.google.com/kubernetes/workload_/gcloud/us-central1-a/my-cluster?project=hidden-solstice-
454006-n0
kubeconfig entry generated for my-cluster.
NAME: my-cluster
LOCATION: us-central1-a
MASTER_VERSION: 1.31.6-gke.1020000
MASTER_IP: 34.66.171.95
MACHINE_TYPE: e2-medium
NODE_VERSION: 1.31.6-gke.1020000
NUM_NODES: 3
STATUS: RUNNING
sarvesh_ambala@cloudshell:~ (hidden-solstice-454006-n0)$
```

Now u go and check kubernetes engine--->cluster , you can see the my-cluster is running




Run the below command

gcloud container clusters get-credentials my-cluster --zone us-central1-a

```
sarvesh_ambala@cloudshell:~ (hidden-solstice-454006-n0)$ gcloud container clusters get-credentials my-cluster --zone us-central1-a
Fetching cluster endpoint and auth data.
kubeconfig entry generated for my-cluster.
sarvesh_ambala@cloudshell:~ (hidden-solstice-454006-n0)$
```

To see the list  of nodes

```
sarvesh_ambala@cloudshell:~ (hidden-solstice-454006-n0)$ kubectl get nodes
NAME                                          STATUS   ROLES    AGE     VERSION
gke-my-cluster-default-pool-020e8447-03bv   Ready    <none>   8m43s   v1.31.6-gke.1020000
gke-my-cluster-default-pool-020e8447-1510   Ready    <none>   8m43s   v1.31.6-gke.1020000
gke-my-cluster-default-pool-020e8447-rv2r   Ready    <none>   8m43s   v1.31.6-gke.1020000
sarvesh_ambala@cloudshell:~ (hidden-solstice-454006-n0)$
```

Create the pods

Kubectl run --image tomcat webserver

```
sarvesh_ambala@cloudshell:~ (hidden-solstice-454006-n0)$ kubectl run --image tomcat webserver
pod/webserver created
```

To see the pods list

```
sarvesh_ambala@cloudshell:~ (hidden-solstice-454006-n0)$ kubectl get pods
NAME        READY   STATUS    RESTARTS   AGE
webserver   1/1     Running   0          20m
sarvesh_ambala@cloudshell:~ (hidden-solstice-454006-n0)$
```

To get the list of pods along with ip address and which node the pod is running

Kubectl get pods -o wide

```
sarvesh_ambala@cloudshell:~ (hidden-solstice-454006-n0)$ kubectl get pods -o wide
NAME        READY   STATUS    RESTARTS   AGE   IP          NODE                                          NOMINATED NODE   READINESS GATES
webserver   1/1     Running   0          22m   10.40.0.4   gke-my-cluster-default-pool-020e8447-1510   <none>           <none>
sarvesh_ambala@cloudshell:~ (hidden-solstice-454006-n0)$
```

Actually u can create the pod using definition file

Create pd-df1.yaml

Vim pd-df1.yaml

```
apiVersion: v1
kind: Pod
metadata:
  name: jenkins-pod
spec:
  containers:
  - name: myjenkins
    image: jenkins/jenkins
    ports:
    - containerPort: 8080
      hostPort: 8080
```

for accessing the application u need to open the port

How to open the port

gcloud compute firewall-rules create rule2 --allow tcp:8080

```
sarvesh_ambala@cloudshell:~/sample (hidden-solstice-454006-n0)$ gcloud compute firewall-rules create rule2 --allow tcp:8080
Creating firewall...working..Created [https://www.googleapis.com/compute/v1/projects/hidden-solstice-454006-n0/global/firewalls/rule2].
Creating firewall...done.
NAME: rule2
NETWORK: default
DIRECTION: INGRESS
PRIORITY: 1000
ALLOW: tcp:8080
DENY:
DISABLED: False
```

Kubectl create -f pd-df1.yaml

Kubectl get pods -o wide

```
sarvesh_ambala@cloudshell:~/sample1 (hidden-solstice-454006-n0)$ kubectl get pods -o wide
NAME           READY   STATUS             RESTARTS   AGE     IP          NODE                                              NOMINATED NODE   READINESS GATES
jenkins-pod    0/1     ImagePullBackOff   0          3h25m   10.40.0.5   gke-my-cluster-default-pool-020e8447-1510          <none>           <none>
jenkins-pod1   0/1     ImagePullBackOff   0          3h8m    10.40.0.6   gke-my-cluster-default-pool-020e8447-1510          <none>           <none>
jk-pod         1/1     Running            0          4m33s   10.40.1.5   gke-my-cluster-default-pool-020e8447-rv2r          <none>           <none>
webserver      1/1     Running            0          4h5m    10.40.0.4   gke-my-cluster-default-pool-020e8447-1510          <none>           <none>
```

Kubectl get nodes -o wide

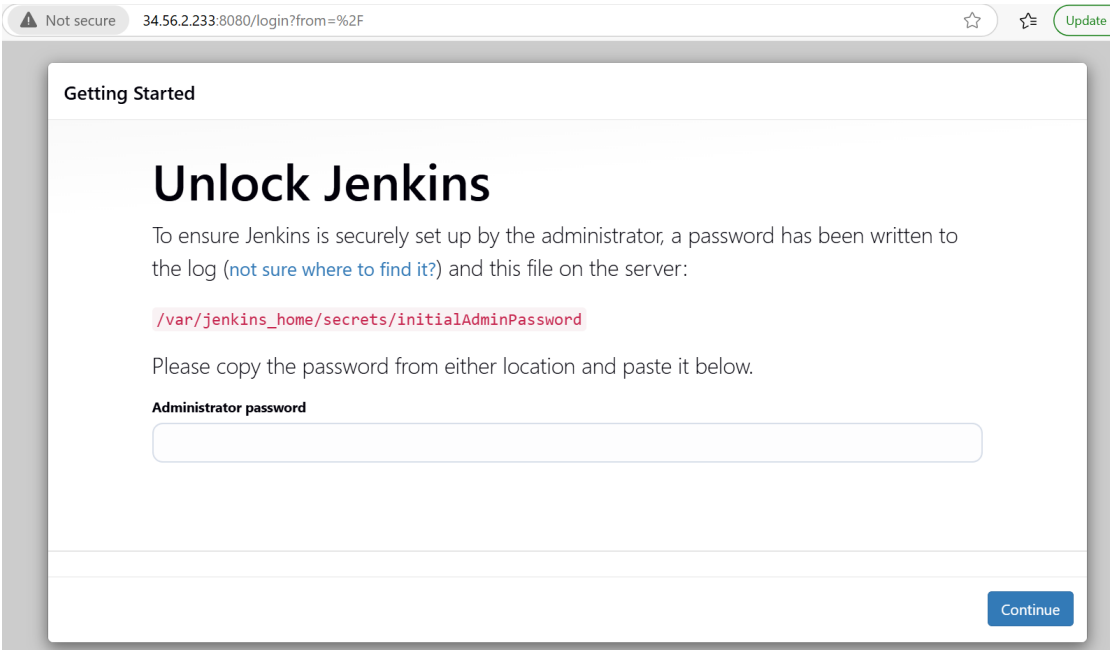
```
sarvesh_ambala@cloudshell:~/sample1 (hidden-solstice-454006-n0)$ kubectl get nodes -o wide
NAME                                              STATUS   ROLES    AGE     VERSION               INTERNAL-IP   EXTERNAL-IP     OS-IMAGE
KERNEL-VERSION   CONTAINER-RUNTIME
gke-my-cluster-default-pool-020e8447-03bv   Ready    <none>   4h19m   v1.31.6-gke.1020000   10.128.0.15   35.225.52.28    Container-Optimized OS from Google
6.6.72+          containerd://1.7.24
gke-my-cluster-default-pool-020e8447-1510   Ready    <none>   4h19m   v1.31.6-gke.1020000   10.128.0.13   34.60.92.233    Container-Optimized OS from Google
6.6.72+          containerd://1.7.24
gke-my-cluster-default-pool-020e8447-rv2r   Ready    <none>   4h19m   v1.31.6-gke.1020000   10.128.0.14   34.56.2.233     Container-Optimized OS from Google
6.6.72+          containerd://1.7.24
sarvesh_ambala@cloudshell:~/sample1 (hidden-solstice-454006-n0)$
```

How can we access the pod

Take the external ip  add the port no 8080

Open the browser paste ipaddress:8080

Now u can able to see the jenkins


Not secure
34.56.2.233:8080/login?from=%2F
Update
Getting Started
Unlock Jenkins
To ensure Jenkins is securely set up by the administrator, a password has been written to the log (not sure where to find it?) and this file on the server:
/var/jenkins_home/secrets/initialAdminPassword
Please copy the password from either location and paste it below.
Administrator password
Continue

# PROMETHEUS SETUP:

helm repo add prometheus  https://prometheus-community.github.io/helm-charts

```
sarvesh_ambala@cloudshell:~ (hidden-solstice-454006-n0)$ helm repo add prometheus  https://prometheus-community.github.io/helm-charts
"prometheus" has been added to your repositories
```

helm repo update

```
sarvesh_ambala@cloudshell:~ (hidden-solstice-454006-n0)$ helm repo update
Hang tight while we grab the latest from your chart repositories...
...Successfully got an update from the "grafana" chart repository
...Successfully got an update from the "prometheus1" chart repository
...Successfully got an update from the "prometheus" chart repository
...Successfully got an update from the "prometheus-community" chart repository
...Successfully got an update from the "pc" chart repository
```

helm install prometheus prometheus-community/kube-prometheus-stack --namespace monitoring --create-namespace

This will install prometheus,alermanager and grafana

```
sarvesh_ambala@cloudshell:~ (hidden-solstice-454006-n0)$ helm install prometheus prometheus-community/kube-prometheus-stack --namespace monitoring --create-na
mespace
```

Check the prometheus pods and services

kubectl get pods -n monitoring

```
sarvesh_ambala@cloudshell:~ (hidden-solstice-454006-n0)$ kubectl get pods -n monitoring
NAME                                                     READY   STATUS    RESTARTS   AGE
alertmanager-prometheus-kube-prometheus-alertmanager-0   2/2     Running   0          3m12s
prometheus-grafana-75bb7d6986-rzg4q                      3/3     Running   0          3m19s
prometheus-kube-prometheus-operator-65c669f8f9-qcjwc     1/1     Running   0          3m19s
prometheus-kube-state-metrics-645c667b6-6bww1            1/1     Running   0          3m19s
prometheus-prometheus-kube-prometheus-prometheus-0       2/2     Running   0          3m11s
prometheus-prometheus-node-exporter-f87x4                1/1     Running   0          3m20s
prometheus-prometheus-node-exporter-gz8rs                1/1     Running   0          3m20s
prometheus-prometheus-node-exporter-vnb86                1/1     Running   0          3m20s
```

kubectl get svc -n monitoring

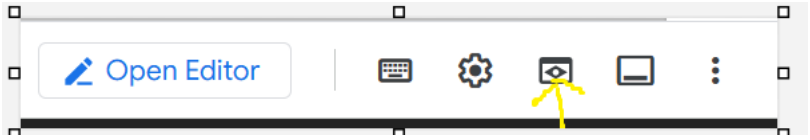
```
sarvesh_ambala@cloudshell:~ (hidden-solstice-454006-n0)$ kubectl get svc -n monitoring
NAME                                      TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)                      AGE
alertmanager-operated                     ClusterIP   None             <none>        9093/TCP,9094/TCP,9094/UDP   4m27s
prometheus-grafana                        ClusterIP   34.118.235.147   <none>        80/TCP                       4m36s
prometheus-kube-prometheus-alertmanager   ClusterIP   34.118.225.122   <none>        9093/TCP,8080/TCP            4m36s
prometheus-kube-prometheus-operator       ClusterIP   34.118.232.165   <none>        443/TCP                      4m36s
prometheus-kube-prometheus-prometheus     ClusterIP   34.118.232.239   <none>        9090/TCP,8080/TCP            4m36s
prometheus-kube-state-metrics             ClusterIP   34.118.228.217   <none>        8080/TCP                     4m36s
prometheus-operated                       ClusterIP   None             <none>        9090/TCP                     4m26s
prometheus-prometheus-node-exporter       ClusterIP   34.118.232.54    <none>        9100/TCP                     4m36s
sarvesh_ambala@cloudshell:~ (hidden-solstice-454006-n0)$
```

Access prometheus and port forwarding

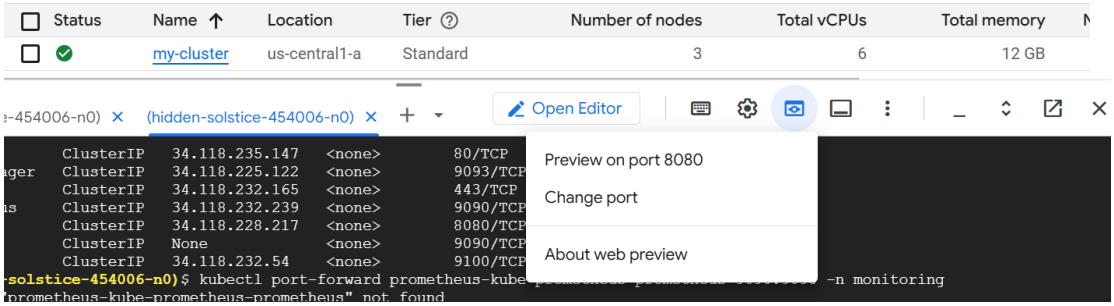
kubectl port-forward svc/prometheus-kube-prometheus-prometheus 9090:9090 -n monitoring

```
sarvesh_ambala@cloudshell:~ (hidden-solstice-454006-n0)$ kubectl port-forward svc/prometheus-kube-prometheus-prometheus 9090:9090 -n monitoring
Forwarding from 127.0.0.1:9090 -> 9090
```

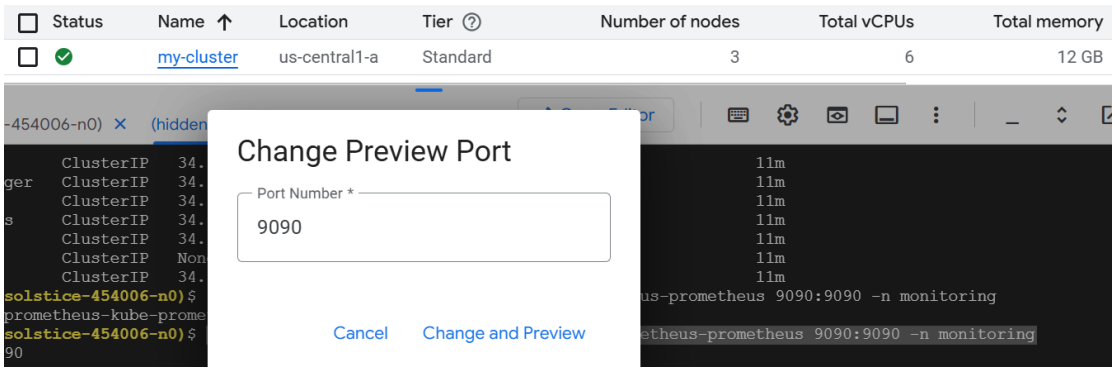
Click on the  webpreview



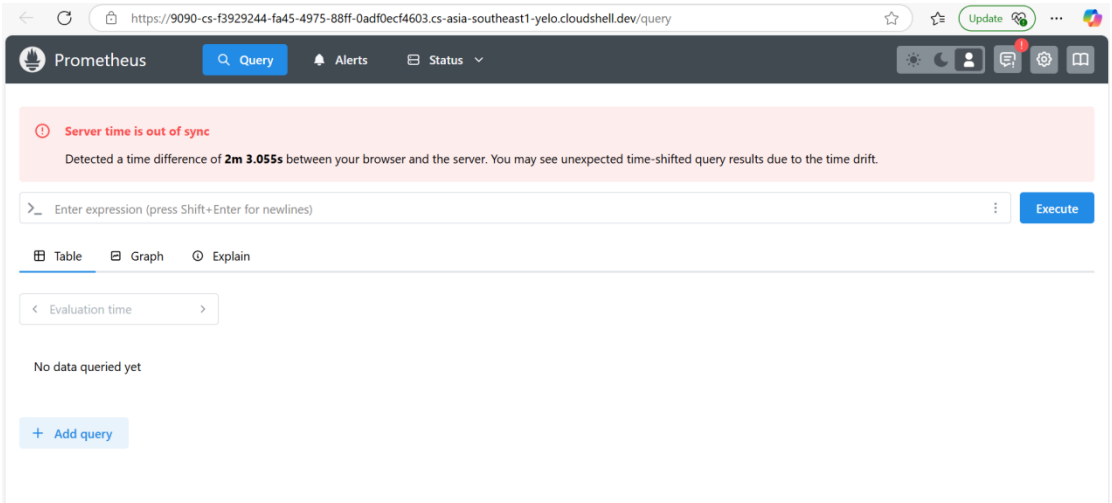

Change port no to 9090




Click on change and preview



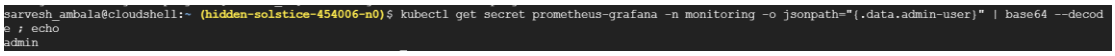

Now u can able to see prometheus in the browser



# ACCESS GRAFANA :

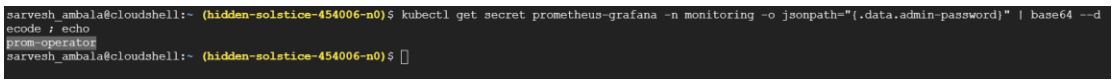
kubectl get secret prometheus-grafana -n monitoring -o jsonpath="{.data.admin-user}" | base64 --decode ; echo

If you run the above command u can see the username for grafana (**admin**)



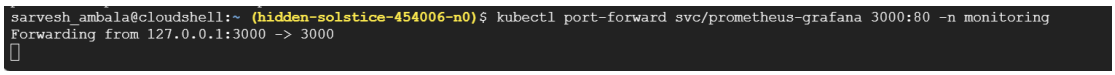

kubectl get secret prometheus-grafana -n monitoring -o jsonpath="{.data.admin-password}" | base64 --decode ; echo

If you run the abvoe command u can see the password for grafana (**prom-operator**)

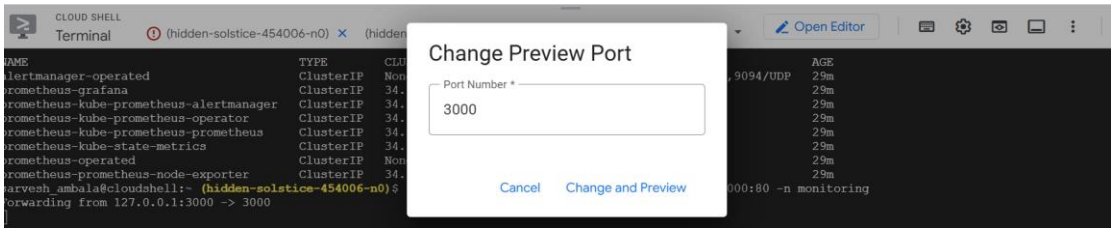


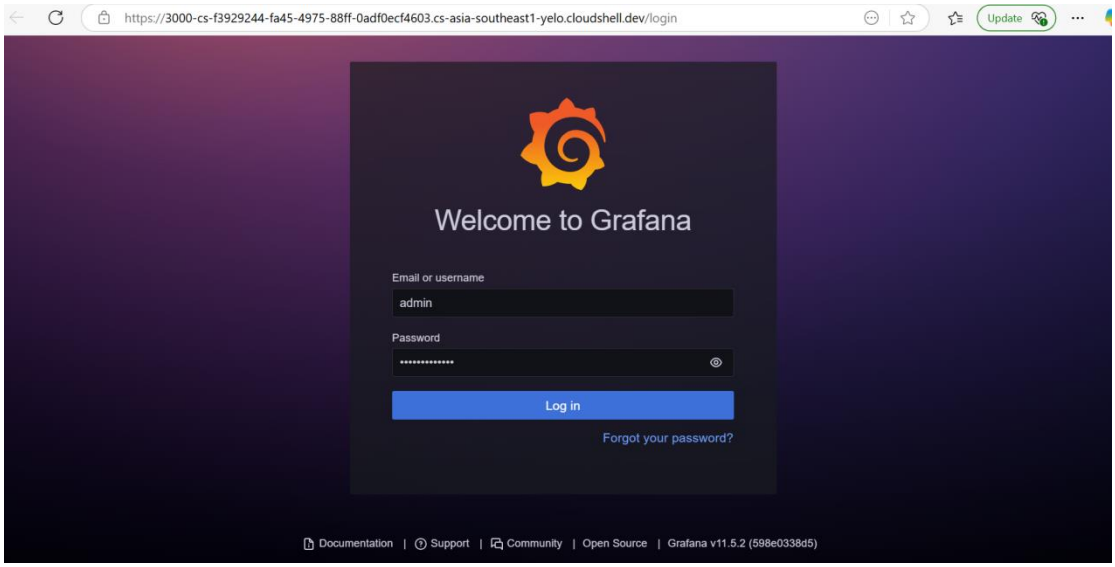

PORT FORWARDING

kubectl port-forward svc/prometheus-grafana 3000:80 -n monitoring




Click on the web preview give the port no 3000 and click on change and preview u can see the grafana






You can login with admin and  prom-operator