|  |
| --- |
| [GSP318] : Deploy to Kubernetes in Google Cloud: Challenge Lab :- |
|  |  |
|  | ---------------------------------------------------------------------------------------------------------------------------------------------- |
|  |  |
|  | Task - 1 : Create a Docker image and store the Dockerfile :- |
|  |  |
|  | gcloud auth list |
|  | gsutil cat gs://cloud-training/gsp318/marking/setup\_marking.sh | bash |
|  | gcloud source repos clone valkyrie-app |
|  | cd valkyrie-app |
|  | cat > Dockerfile <<EOF |
|  | FROM golang:1.10 |
|  | WORKDIR /go/src/app |
|  | COPY source . |
|  | RUN go install -v |
|  | ENTRYPOINT ["app","-single=true","-port=8080"] |
|  | EOF |
|  | docker build -t valkyrie-app:v0.0.1 . |
|  | cd .. |
|  | cd marking |
|  | ./step1.sh |
|  |  |
|  | ---------------------------------------------------------------------------------------------------------------------------------------------- |
|  |  |
|  | Task - 2 : Test the created Docker image :- |
|  |  |
|  | cd .. |
|  | cd valkyrie-app |
|  | docker run -p 8080:8080 valkyrie-app:v0.0.1 & |
|  | cd .. |
|  | cd marking |
|  | ./step2.sh |
|  |  |
|  | ---------------------------------------------------------------------------------------------------------------------------------------------- |
|  |  |
|  | Task - 3 : Push the Docker image in the Google Container Repository :- |
|  |  |
|  | cd .. |
|  | cd valkyrie-app |
|  | docker tag valkyrie-app:v0.0.1 gcr.io/$GOOGLE\_CLOUD\_PROJECT/valkyrie-app:v0.0.1 |
|  | docker push gcr.io/$GOOGLE\_CLOUD\_PROJECT/valkyrie-app:v0.0.1 |
|  | sed -i s#IMAGE\_HERE#gcr.io/$GOOGLE\_CLOUD\_PROJECT/valkyrie-app:v0.0.1#g k8s/deployment.yaml |
|  |  |
|  | ---------------------------------------------------------------------------------------------------------------------------------------------- |
|  |  |
|  | Task - 4 : Create and expose a deployment in Kubernetes :- |
|  |  |
|  | sed -i s#IMAGE\_HERE#gcr.io/$GOOGLE\_CLOUD\_PROJECT/valkyrie-app:v0.0.1#g k8s/deployment.yaml |
|  | gcloud container clusters get-credentials valkyrie-dev --zone us-east1-d |
|  | kubectl create -f k8s/deployment.yaml |
|  | kubectl create -f k8s/service.yaml |
|  |  |
|  | git merge origin/kurt-dev |
|  | ---------------------------------------------------------------------------------------------------------------------------------------------- |
|  |  |
|  | Task - 5 : Increase the replicas from 1 to 3 :- |
|  |  |
|  | kubectl edit deployment valkyrie-dev |
|  |  |
|  | // Press "i" to edit and change "replicas" from 1 to 3. |
|  | Press "Esc" -> ":wq" -> Enter |
|  |  |
|  | ---------------------------------------------------------------------------------------------------------------------------------------------- |
|  |  |
|  | Task - 6 : Update the deployment with a new version of valkyrie-app :- |
|  |  |
|  | docker build -t gcr.io/$GOOGLE\_CLOUD\_PROJECT/valkyrie-app:v0.0.2 . |
|  | docker push gcr.io/$GOOGLE\_CLOUD\_PROJECT/valkyrie-app:v0.0.2 |
|  | kubectl edit deployment valkyrie-dev |
|  |  |
|  | // Press 'i' to edit and change image to "image: gcr.io/YOUR\_PROJECT\_ID/valkyrie-app:v0.0.2". |
|  | Press "Esc" -> ":wq" -> Enter |
|  |  |
|  | docker ps |
|  |  |
|  | ---------------------------------------------------------------------------------------------------------------------------------------------- |
|  |  |
|  | Task - 7 : Create a pipeline in Jenkins to deploy your app :- |
|  |  |
|  | docker kill container\_id |
|  |  |
|  | export POD\_NAME=$(kubectl get pods --namespace default -l "app.kubernetes.io/component=jenkins-master" -l "app.kubernetes.io/instance=cd" -o jsonpath="{.items[0].metadata.name}") |
|  | kubectl port-forward $POD\_NAME 8080:8080 >> /dev/null & |
|  | printf $(kubectl get secret cd-jenkins -o jsonpath="{.data.jenkins-admin-password}" | base64 --decode);echo |
|  |  |
|  | gcloud source repos list |
|  |  |
|  | // Open Jenkins Web View -> Preview on port 8080 |
|  | Username : admin |
|  | Password : {Code output from previous command} |
|  |  |
|  | -> Manage Jenkins -> Manage Credentials -> Global -> add credentials -> Kind: Google Service Account from metadata -> OK |
|  | -> Jenkins -> New Item -> Name : valkyrie-app -> Pipeline -> OK |
|  | -> Pipeline -> Script: Pipeline script from SCM -> SCM: Git |
|  | -> Repository URL: {Url from previous command} -> Credentials: {Project id} |
|  | -> Apply -> Save |
|  |  |
|  | // In cloud shell , run :- |
|  |  |
|  | sed -i "s/green/orange/g" source/html.go |
|  |  |
|  | sed -i "s/YOUR\_PROJECT/$GOOGLE\_CLOUD\_PROJECT/g" Jenkinsfile |
|  | git config --global user.email "you@example.com" // Email |
|  | git config --global user.name "student..." // Username |
|  | git add . |
|  | git commit -m "built pipeline init" |
|  | git push |
|  |  |
|  | // In Jenkins click Build and wait to get score. |