Creating a Kubernetes Cluster with Jenkins X And Importing App

This lesson provides the link to gists that you can use to create a Kubernetes cluster with Jenkins X. Moreover, it also gives some steps to follow before we continue with the chapter.

We'll cover the following

- Create a new cluster and install Jenkins X
  - GKE
  - EKS
  - AKS
- Install Jenkins X in an existing serverless cluster
- Before we get started

You know what to do; create a new Jenkins X cluster unless you kept the one from before.

> All the commands from this chapter are available in the 07-dev.sh Gist.

Create a new cluster and install Jenkins X

For your convenience, the Gists from the previous chapter are available below as well.

GKE

Create a new serverless **GKE** cluster: gke-jx-serverless.sh

EKS

Create a new serverless **EKS** cluster:
eks-jx-serverless.sh

AKS

Create a new serverless **AKS** cluster:
aks-jx-serverless.sh

Install Jenkins X in an existing serverless cluster

Use an **existing** serverless cluster: install-serverless.sh

Before we get started

We'll continue using the *go-demo-6* application. Please enter the local copy of the repository, unless you're there already.

```
cd go-demo-6
```

> The commands that follow will reset your master branch with the contents of the `buildpack` branch that contains all the changes we did so far. Please execute them only if you are unsure whether you did all the exercises correctly.

```
git pull

git checkout buildpack-tekton

git merge -s ours master --no-edit

git checkout master
git merge buildpack-tekton


git push
```

If you restored the branch, the chances are that there is a reference to my user (`vfarcic`). We'll change that to Google project since that's what is the expected location of container images.

> ⚠ Please execute the commands that follow only if you are using **GKE** and if you ever restored a branch at the beginning of a chapter (like in the snippet above).

```
# If GKE
export REGISTRY_OWNER=$PROJECT

# If EKS or AKS
# Replace `[...]` with your GitHub user
export REGISTRY_OWNER=[...]

cat charts/go-demo-6/Makefile \
    | sed -e \
    "s@vfarcic@$REGISTRY_OWNER@g" \
    | sed -e \
    "s@devops-26@$REGISTRY_OWNER@g" \
    | tee charts/go-demo-6/Makefile

cat charts/preview/Makefile \
    | sed -e \
    "s@vfarcic@$REGISTRY_OWNER@g" \
    | sed -e \
    "s@devops-26@$REGISTRY_OWNER@g" \
    | tee charts/preview/Makefile

cat skaffold.yaml \
    | sed -e \
    "s@vfarcic@$REGISTRY_OWNER@g" \
    | sed -e \
    "s@devops-26@$REGISTRY_OWNER@g" \
    | tee skaffold.yaml
```

> 🔍 If you destroyed the cluster at the end of the previous chapter, we'll need to import the *go-demo-6* application again. Please execute the commands that follow only if you created a new cluster specifically for the exercises from this chapter.

```
jx import --pack go --batch-mode


jx get activities \
    --filter go-demo-6 \
    --watch
```

Please wait until the activity of the application shows that all the steps were executed successfully, and stop the watcher by pressing *ctrl+c*.

Now we can explore how to leverage Jenkins X for our development environments.