https://github.com/dhinojosa/working-with-helm

https://kind.sigs.k8s.io/

No Tiller in Vesion 3

help iniit installs tiller on to your k8s cluster

```
brew install go
brew install helm@2
/usr/local/Cellar/helm@2/2.16.5/bin/helm
helm version

brew install  kind
kind delete cluster
kind create cluster
kubectl cluster-info --context kind-kind

helm init
kubectl config get-contexts
kubectl get svc --all-namespaces

kubectl config use-context kind-kind
kubectl create clusterrolebinding add-on-cluster-admin --clusterrole=cluster-admin --serviceaccount=kube-system:default
helm install stable/mysql
helm ls
helm delete --purge dusty-turkey
helm delete --purge gilded-beetle
```

Install Kubectl (https://kubernetes.io/docs/tasks/tools/install-kubectl/)
* Install Docker (https://docs.docker.com/get-docker/)

Minikube:
* Install Virtualbox (https://www.virtualbox.org/wiki/Downloads), which is a Salesforce recommended product
* Install Minikube (https://kubernetes.io/docs/tasks/tools/install-minikube/)
* Ensure you can connect using $ kubectl config use-context minikube
* Ensure you can run something $ kubectl get pods --all-namespaces

or

Kind:
* Install Kind: (https://kind.sigs.k8s.io/docs/user/quick-start/)
* Run Kind using the following script which integrates with an ingress

```
cat <<EOF | kind create cluster --config=-
kind: Cluster
apiVersion: kind.x-k8s.io/v1alpha4
nodes:
- role: control-plane
  kubeadmConfigPatches:
  - |
    kind: InitConfiguration
    nodeRegistration:
      kubeletExtraArgs:
        node-labels: "ingress-ready=true"
        authorization-mode: "AlwaysAllow"
  extraPortMappings:
  - containerPort: 80
    hostPort: 80
    protocol: TCP
  - containerPort: 443
    hostPort: 443
    protocol: TCP
EOF
```

* Using kubectl, ensure that you are switch context to your kind cluster

```
$ kubectl config use-context kind-kind
```

* Run the following so we can deploy to kind

```
$
```

That is it, hope all is well, and see you tomorrow.
Danno

Helm 3

https://helm.sh/docs/intro/quickstart/

```
helm repo add stable https://kubernetes-charts.storage.googleapis.com/
helm search repo stable
helm repo update
helm install stable/mysql --generate-name
```

or

```
helm install my-mysql stable/mysql
helm list
helm uninstall mysql-1587864254
```

```
helm install happy-panda stable/mariadb
echo '{mariadbUser: user0, mariadbDatabase: user0db}' > config.yaml
helm install -f config.yaml stable/mariadb --generate-name
helm install mine --set name=value
helm get values mine

cat values.yaml
name: value
```

```
helm install mine -f values.yaml .
helm get values mine > myvalues.yaml
```

```
helm get manifest good-puppy
```

Install CloudBees Core

```
helm repo add nginx-stable https://helm.nginx.com/stable
```

https://hub.helm.sh/charts/cloudbees/cloudbees-core

```
helm repo add cloudbees https://charts.cloudbees.com/public/cloudbees
helm repo update

helm install cloudbees-core-222.1.1 cloudbees/cloudbees-core --set nginx.ingress.Enabled=true --set OperationsCenter.HostName=‘’

helm install cloudbees-core-222.1.1 cloudbees/cloudbees-core -f example-values.yaml
```

https://github.com/cloudbees/cloudbees-examples/tree/master/helm-custom-value-file-examples

```
helm template cloudbees-core-2.222 cloudbees/cloudbees-core   --values values.yaml --namespace default > cloudbees-core.yaml
```

```
cat CloudBees/dev-values.yaml
```

```
namespace: cloudbees-ssdev
ingress_class: nginx-cloudbees-ssdev
ingress_name: cjoc-cloudbees-ssdev
oc_image: 460949649053.dkr.ecr.us-
west-2.amazonaws.com/cicd_cjoc:
2.222.1.1_dev
mm_image: 460949649053.dkr.ecr.us-
west-2.amazonaws.com/
cicd_managed_master:2.222.1.1_dev
agent_image: 460949649053.dkr.ecr.us-
west-2.amazonaws.com/
cicd_build_agent:amazonlinux
ingress_url: dev-build.sfdcbt.net
storage_class: cloudbees-ssdev-efs
```

```
helm install --dry-run --debug
cloudbees-core-2.222 --namespace
cloudbees-ssdev -f CloudBees/dev-
values.yaml  ./CloudBees/ >
myyaml.yaml
```