1. Download **kubectl**

[**https://kubernetes.io/docs/tasks/tools/install-kubectl-windows/**](https://kubernetes.io/docs/tasks/tools/install-kubectl-windows/)

### Install kubectl binary with curl on Windows

### Download the [latest release v1.23.0](https://dl.k8s.io/release/v1.23.0/bin/windows/amd64/kubectl.exe).

### <https://minikube.sigs.k8s.io/docs/start/>

### rename file with minikube.exe

### Create a folder inside any drive name minikube

### Put both downloaded file inside this folder

### Add path inside environment variable example path=c:/minikube

1. check on cmd by command **minikube version**
2. Start minikube: we can start minikube using multipal driver example: hiper-v, virtualBox or docker. Hiper-v is not abailable all window 10, its available on window 10 professional ,enterprise

VirtualBox need lot of memory min 16GB ram.so starting using docker driver.

1. minikube start –driver=docker // once you start minikube it will create single node cluster
2. check the status minikube status
3. kubectl cluster-info //it will give information about cluster
4. kubectl get nodes //verify node
5. minikube docker-env //to allow minikube to access docker local repo
6. copy the last line like “@FOR /f "tokens=\*" %i IN ('minikube -p minikube docker-env') DO @%i” and execute this
7. verify docker images
8. build the docker image of any spring boot application using command “docker build -t springboot-k8s-demo:1.0 . “ // go to the spring boot folder create Dockerfile and build
9. Now run this image in a POD for that create a deployment object
10. Use command **kubectl create deployment <any name> --image=<imageName>:tag –port=8080**
11. **kubectl get deployment //verify deployment**
12. **kubectl describe deployment <deployment name> //verify the deployment**
13. **kubectl get pod //get the status of pod**
14. **kubectl logs <POD name> //verify the log of POD**
15. **Now create a service to expose outside cluster so that we can access**
16. use command **kubectl expose deployment <deploymentName> --type=NodePort // here NodePort is servicetype, we have lot of service type like loadbalancer**
17. **kubectl get service // To get Service details**
18. **minikube service <serviceName> --url**
19. **now you can access your service through url**
20. **minikube dashboard // to view node cluster on browser**
21. **No clean everything**
22. **kubectl delete service <serviceName> //To delete service**
23. **kubectl delete deployment <deploymentName> //delete deployment**
24. **minikube stop**
25. **minikube delete //it will delete minikube**