Kubernetes Setup and Installation Document

To start with Kubernetes you need to have installed the following on your computer :

- Docker - the latest version of docker for Mac users already have kubernetes plugins.
  Windows users, please check if you have kubernetes plugins. If not please uninstall current Docker installed on your machine first and then install Docker Edge
- MiniKube - Minikube is a tool that makes it easy to run Kubernetes locally. Minikube runs a single-node Kubernetes cluster inside a VM on your laptop for users looking to try out Kubernetes or develop with it day-to-day.
  https://github.com/kubernetes/minikube
- kubectl - is a must because this is how you interact with Kubernetes using the command line. It's supported on Windows and Mac, too.
  https://kubernetes.io/docs/tasks/tools/install-kubectl/

You'll need one docker image to deploy it on kubernetes. For demo purpose, I am using tutum/hello-world image. It is already present on docker. The image contains a simple hello world program with tomcat plugins to run it on the web.

You can add your own application image instead.

After done with the installation, test your docker image first. Run it on your machine and check the output.

Now, when you have a final image of your application, you can start the process of deployment.

1. Start the VM: To deploy your application you need to start your VM first.

Command: minikube start

2. Service:

service.yaml

```yaml
apiVersion: v1
kind: Service
metadata:
  name: helloworldservice
spec:
  selector:
    app: hello-world
  ports:
  - protocol: TCP
    #ports accessible inside the cluster
    port: 8080
    #ports to forward to inside the pod
    targetPort: 80
    #ports accessible outside cluster
    nodePort: 32767
  type: LoadBalancer
```

This is the service that I am using for my demo. You can change the name of service. And make sure to change the selector app, give the name of your application instead.

To create service use following command:
kubectl create -f service.yml

3. Deployment

deployment.yaml

```yaml
apiVersion: apps/v1beta1
kind: Deployment
metadata:
  name: hello-world
spec:
  replicas: 5
  template:
    metadata:
      labels:
        app: hello-world
    spec:
      containers:
      - name: hello-world
        image: tutum/hello-world
        ports:
        - containerPort: 80
```

This is the deployment file that I have used for the demo. Please make sure to change the image name with your own application image name. You can mention as many as replicas you need of your application.

To create deployment use following command:
kubectl create -f deployment.yml

After deployment, you can check all the status of your Kubernetes cluster on your browser by using the command:

```
minikube dashboard
```

This command will take you to the browser and you can see all the procedures that are running in Kubernetes cluster.

It provides you, a user interface to interact with pods, services, and deployments without using the command line.

Basic Commands:

Below are some kubectl commands that will useful for you.

kubectl get pods - to get running pods

kubectl get service - to get services

kubectl get deployment - to get deployment

kubectl get nodes - to get available nodes

kubectl apply -f service.yaml - to update service/file

kubectl delete svc service.yaml - to delete service

kubectl delete deployment deployment.yaml - to delete deployment