Kubernetes

https://kubernetes.io/docs/tasks/tools/install-minikube/#install-kubectl

```
curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key add -
echo "deb https://apt.kubernetes.io/ kubernetes-xenial main" | sudo tee -a
/etc/apt/sources.list.d/kubernetes.list
sudo apt-get update
sudo apt-get install -y kubectl
## Remove this version as we will have it with microk8s
sudo apt-get remove kubectl

kubectl version
kubectl cluster-info

wget
https://github.com/Activiti/activiti-cloud-charts/blob/master/activiti-cloud-full-example/helm-service-account-role.yaml


kubectl apply -f helm-service-account-role.yaml
helm init --service-account helm --upgrade
curl -Lo minikube
https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64 \

curl -Lo minikube
https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64   && chmod +x
minikube
ls
sudo cp minikube /usr/local/bin && rm minikube

minikube start

sudo apt-cache search kubernetes
sudo apt-cache search docker

sudo snap install microk8s --classic

sudo service  apache2 stop
sudo netstat -tpln | grep 80
wget --no-check-certificate http://127.0.0.1/helloWorld
```

```
kubectl get nodes
kubectl start
```

https://microk8s.io/#quick-start

```
microk8s.start
kubectl get nodes
microk8s.enable dashboard
netstat -tpln | grep 8080
kubectl clusterinfo
kubectl cluster-info
microk8s.reset
kubectl cluster-info
microk8s.enable dashboard dns
kubectl cluster-info
kubectl get pods --all
kubectl get pods
kubectl get pods --all-namespaces
kubectl get pods --all-namespaces kubernetes-dashboard-654cfb4879-df8w4
kubectl get pods -n kube-system  kubernetes-dashboard-654cfb4879-df8w4
kubectl describe pods -n kube-system  kubernetes-dashboard-654cfb4879-df8w4
kubectl get services --all-namespace
kubectl get services --all-namespaces
kubectl cluster-info
kubectl apply -f
https://raw.githubusercontent.com/kubernetes/dashboard/v1.10.1/src/deploy/recommended/kubernetes-dashboard.yaml
kubectl proxy
kubectl get services --all-namespaces
kubectl get pods --all-namespaces kubernetes-dashboard-654cfb4879-df8w4
kubectl get pods --all-namespaces
kubectl cluster-info
microk8s.reset
microk8s.enable
microk8s.enable dashboard dns proxy
kubectl cluster-info
kubectl get pods --all-namespaces
kubectl get services --all-namespaces
kubectl proxy &
jobs
kubectl get secrets
kubectl describe  secrets
```

```
touch mynotes.txt
nano mynotes.txt
> notes.txt
ls~
ls
kubectl config set-credentials cluster-admin --help
kubectl config set-credentials cluster-admin --token=bearer_token
nano /snap/microk8s/383/client.config
ls /snap/microk8s/383/
nano /snap/microk8s/383/configs/kubelet.config
nano /snap/microk8s/383/client.config
nano /snap/microk8s/383/basic_auth.csv
nano /snap/microk8s/383/configs/kubelet.config
kubectl get namespaces default
kubectl describe namespaces default
kubectl get namespaces -o yaml default
kubectl get namespaces -o yaml default > isti-env.yaml
nano isti-env.yaml
kubectl apply -f isti-env.yaml
sudo apt-get install docker
ls
kubectl --help~
kubectl create --help
kubectl create deployment --help
kubectl create deployment mydeployment --image=nginx -o yaml > mydeployment.yaml
nano mydeployment.yaml
kubectl apply -f mydeployment.yaml
kubectl get namespaces default
kubectl cluster-info
kubectl get deployments
kubectl get deployments -n isti-env
kubectl get deployments -n default
kubectl describe deployments mydeployment
kubectl describe deployments mydeployment -n default
kubectl describe deployments mydeployment -n isti-env
kubectl delete deployments -n default mydeployment
kubectl describe deployments mydeployment -n default
kubectl describe deployments mydeployment -n isti-env
kubectl expose --help
kubectl get namespaces isti-env
kubectl get pods --all-namespaces
kubectl get services --all-namespaces
kubectl delete deployments -n isti-env mydeployment
kubectl apply -f helloWorld.yaml
kubectl scale --help
kubectl expose --help
kubectl expose deployment helloworld --port=8002 --target-port=80 -n isti-env
kubectl get services helloworld -n isti-env
kubectl get services helloworld -n isti-env -o yaml >> helloWorld.yaml
kubectl delete services helloworld -n isti-env
kubectl apply -f helloWorld.yaml
kubectl get pods --all-namespaces
kubectl get services helloworld -n isti-env
kubectl exec -it helloworld-866b94d7c7-4x6h8 /bin/bash
kubectl exec -it helloworld-866b94d7c7-4x6h8 -n isti-env /bin/bash
kubectl get services helloworld -n isti-env
kubectl get pods --all-namespaces
kubectl get services --all-namespaces
job
jobs
fg 1
jobs
kubectl proxy &
nano helloWorld.yaml
cd /snap/microk8s/383/
kubectl get svc helloworld -n isti-env  -o yaml
kubectl apply -f helloWorld.yaml
kubectl get pods --all-namespaces
kubectl get svc helloworld -n isti-env  -o yaml
microk8s.inspect
sudo iptables -P FORWARD ACCEPT
microk8s.inspect
kubectl delete services helloworld -n isti-env
nano helloWorld.yaml
kubectl apply -f helloWorld.yaml
kubectl get pods --all-namespaces
kubectl get svc helloworld -n isti-env  -o yaml
kubectl delete services helloworld -n isti-env
nano helloWorld.yaml
kubectl apply -f helloWorld.yaml
kubectl delete services helloworld -n isti-env
kubectl expose deployment helloworld -n isti-env --type=LoadBalancer

microk8s.enable ingress
kubectl get services helloworld -n isti-env
kubectl delete services helloworld -n isti-env
kubectl expose deployment helloworld -n isti-env --type=LoadBalancer
kubectl get services helloworld -n isti-env
kubectl delete services helloworld -n isti-env
kubectl expose deployment helloworld -n isti-env --type=LoadBalancer
kubectl get services helloworld -n isti-env
kubectl get services helloworld -n isti-env -o yaml
kubectl get pods -o yaml | grep -i podip
kubectl delete services helloworld -n isti-env
kubectl apply -f helloWorld.yaml
kubectl expose deployment helloworld --port=8002 --target-port=80 -n isti-env
kubectl describe services helloworld -n isti-env
kubectl describe ingress test-ingress -n isti-env
nano helloWorld.yaml
kubectl apply -f helloWorld.yaml
kubectl delete ingress test-ingress -n default
kubectl delete ingress test-ingress -n isti-env
kubectl apply -f helloWorld.yaml
kubectl describe ingress test-ingress  -n isti-env
```

Kubernetes Dashboard links

http://localhost:8001/api/v1/namespaces/kube-system/services/https:kubernetes-dashboard:/proxy/#!/cluster?namespace=_all
https://github.com/kubernetes/dashboard
https://kubernetes.io/docs/tasks/access-application-cluster/web-ui-dashboard/#accessing-the-dashboard-ui

Kubernetes Links

https://kubernetes.io/docs/concepts/services-networking/service/
https://kubernetes.io/docs/concepts/services-networking/ingress/
https://kubernetes.io/docs/concepts/containers/images/

Docker links

https://docs.docker.com/install/linux/docker-ce/ubuntu/

Git

```
git config --global user.email "musak.istvan@gmail.com"
git config --global user.name "Musak Istvan"
git config --global core.editor nano
git config --list
git config --global color.status auto
git config --global color.branch auto
git config --global color.interactive auto
git config --global color.diff auto
ssh-keygen -o
cat id_rsa.pub
git status
git add .
git status
git commit
git push origin master
git checkout origin/master
git checkout master
git checkout -b origin/poc_email
git branch
git checkout master
git branch -D origin/poc_email
git checkout origin/poc_email poc_email
git checkout -b origin/poc_email poc_email
git fetch
git fetch origin
git branch
git branch -v -a
git --version
git config --version
git config --get
```

Helm

https://docs.helm.sh/using_helm/#securing-your-helm-installation

```
sudo snap install helm --classic
```

```
helm init
helm init --canary-image
```