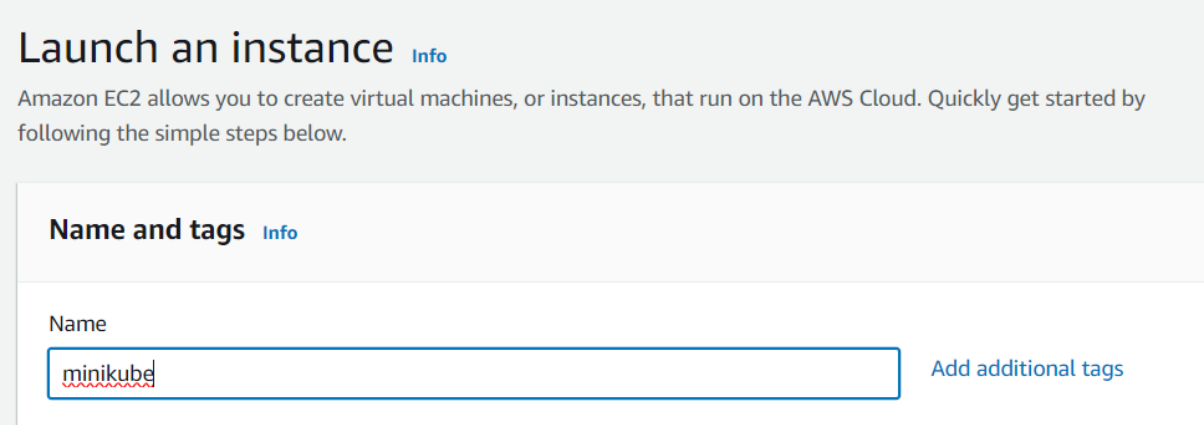
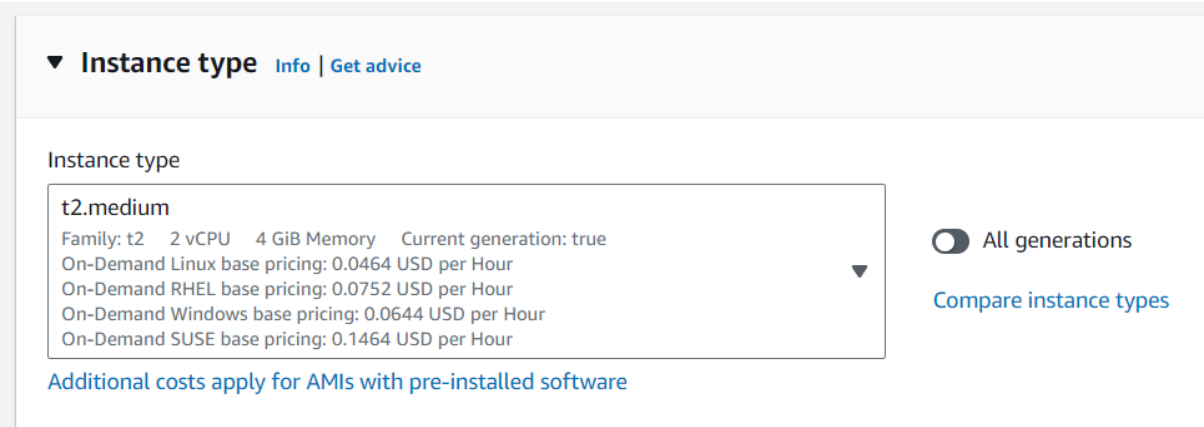
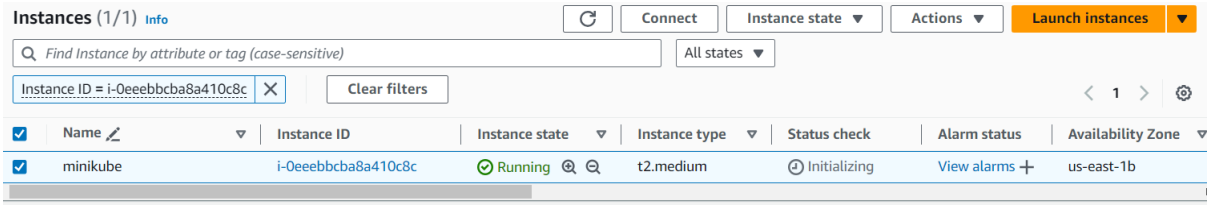
# Task 19: Setup minikube at your local and explore creating namespaces (Go through official documentation)

## 1. Launch an ec2 with t2. medium:




## 2. Install docker:

```
ubuntu@ip-172-31-26-245:~$ sudo apt install docker.io
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
```

```
ubuntu@ip-172-31-26-245:~$ sudo docker version
Client:
 Version:           24.0.7
 API version:       1.43
 Go version:        go1.21.1
 Git commit:        24.0.7-0ubuntu2~22.04.1
 Built:             Wed Mar 13 20:23:54 2024
 OS/Arch:           linux/amd64
 Context:           default

Server:
 Engine:
  Version:          24.0.7
  API version:      1.43 (minimum version 1.12)
  Go version:       go1.21.1
  Git commit:       24.0.7-0ubuntu2~22.04.1
  Built:            Wed Mar 13 20:23:54 2024
  OS/Arch:          linux/amd64
  Experimental:     false
 containerd:
  Version:          1.7.12
```

```
ubuntu@ip-172-31-26-245:~$ sudo systemctl start docker
sudo systemctl enable docker
sudo systemctl status docker
● docker.service - Docker Application Container Engine
     Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
     Active: active (running) since Wed 2024-08-07 12:06:19 UTC; 6min ago
TriggeredBy: ● docker.socket
       Docs: https://docs.docker.com
   Main PID: 2563 (dockerd)
      Tasks: 8
     Memory: 25.8M
        CPU: 341ms
     CGroup: /system.slice/docker.service
             └─2563 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock
```

```
ubuntu@ip-172-31-26-245:~$ sudo usermod -aG docker ubuntu
```

## 3. Install kubectl:

```
ubuntu@ip-172-31-26-245:~$ curl -o kubectl https://amazon-eks.s3.us-west-2.amazonaws.com/1.19.6/2021-01-05/bin/linux/amd64/kubectl
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 57.4M  100 57.4M    0     0   9.7M      0  0:00:05  0:00:05 --:--:-- 13.5M
ubuntu@ip-172-31-26-245:~$ chmod +x ./kubectl
ubuntu@ip-172-31-26-245:~$ sudo mv ./kubectl /usr/local/bin
ubuntu@ip-172-31-26-245:~$ kubectl version --short --client
Client Version: v1.19.6-eks-49a6c0
```

## 4. Install eksctl:

```
ubuntu@ip-172-31-26-245:~$ curl --silent --location "https://github.com/weaveworks/eksctl/releases/latest/download/eksctl_$(uname -s)_amd64.tar.gz" | tar xz -C /t
mp
ubuntu@ip-172-31-26-245:~$ sudo mv /tmp/eksctl /usr/local/bin
ubuntu@ip-172-31-26-245:~$ eksctl version
0.188.0
```

## 5. Install minikube:

```
ubuntu@ip-172-31-26-245:~$ curl -LO https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64
sudo install minikube-linux-amd64 /usr/local/bin/minikube && rm minikube-linux-amd64
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 91.1M  100 91.1M    0     0   113M      0 --:--:-- --:--:-- --:--:--  113M
```

## 6. Minikube start:

```
ubuntu@ip-172-31-26-245:~$ minikube start
* minikube v1.33.1 on Ubuntu 22.04 (xen/amd64)
* Automatically selected the docker driver. Other choices: none, ssh
* Using Docker driver with root privileges
* Starting "minikube" primary control-plane node in "minikube" cluster
* Pulling base image v0.0.44 ...
* Downloading Kubernetes v1.30.0 preload ...
    > preloaded-images-k8s-v18-v1...:  342.90 MiB / 342.90 MiB  100.00% 66.28 M
    > gcr.io/k8s-minikube/kicbase...:  481.58 MiB / 481.58 MiB  100.00% 70.46 M
* Creating docker container (CPUs=2, Memory=2200MB) ...
* Preparing Kubernetes v1.30.0 on Docker 26.1.1 ...
  - Generating certificates and keys ...
  - Booting up control plane ...
  - Configuring RBAC rules ...
* Configuring bridge CNI (Container Networking Interface) ...
* Verifying Kubernetes components...
  - Using image gcr.io/k8s-minikube/storage-provisioner:v5
* Enabled addons: default-storageclass, storage-provisioner

! /usr/local/bin/kubectl is version 1.19.6-eks-49a6c0, which may have incompatibilities with Kubernetes 1.30.0.
  - Want kubectl v1.30.0? Try 'minikube kubectl -- get pods -A'
* Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
```

```
ubuntu@ip-172-31-26-245:~$ minikube status
minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured
```

## 7. Namespace:

```
ubuntu@ip-172-31-26-245:~$ kubectl create namespace my-namespace
namespace/my-namespace created
ubuntu@ip-172-31-26-245:~$ kubectl get namespaces
NAME              STATUS   AGE
default           Active   4m17s
kube-node-lease   Active   4m17s
kube-public       Active   4m17s
kube-system       Active   4m17s
my-namespace      Active   14s
```

## 8. Namespace.yaml file:

```yaml
apiVersion: v1
kind: Namespace
metadata:
  name: my-namespace
```

```
ubuntu@ip-172-31-26-245:~$ vi namespace.yaml
ubuntu@ip-172-31-26-245:~$ kubectl apply -f namespace.yaml
Warning: kubectl apply should be used on resource created by either kubectl create --save-config or kubectl apply
namespace/my-namespace configured
```