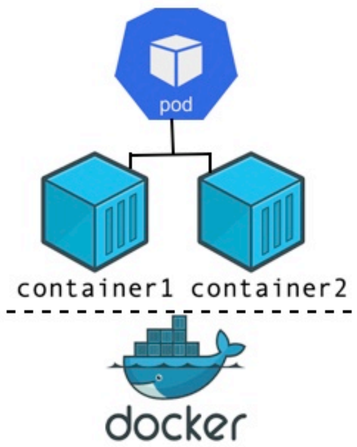
What is a Pod?

A Pod is an smallest entity or unit within the Kubernetes cluster where in one or more containers are kept together and running.

Are these containers that are part of the Pod are isolated from each other?
Pod is an logical group of multiple containers that has common attributes or requirements in running them.

Yes, both the Containers that are part of the Pod are isolated from each other as below:
1. cpu/memory
2. ipc
3. process
4. user
5. utc
6. networking



then why should we have these 2 containers being logical grouped aspart of a Pod?
These applications/containers do share few common resources and lifecycle as below
1. Networking = The applications across these containers should be able to communicate with each other
2. share common volumes or bind mounts
3. lifecycle dependencies (start/stop) = indirectly one cannot exist without the other.

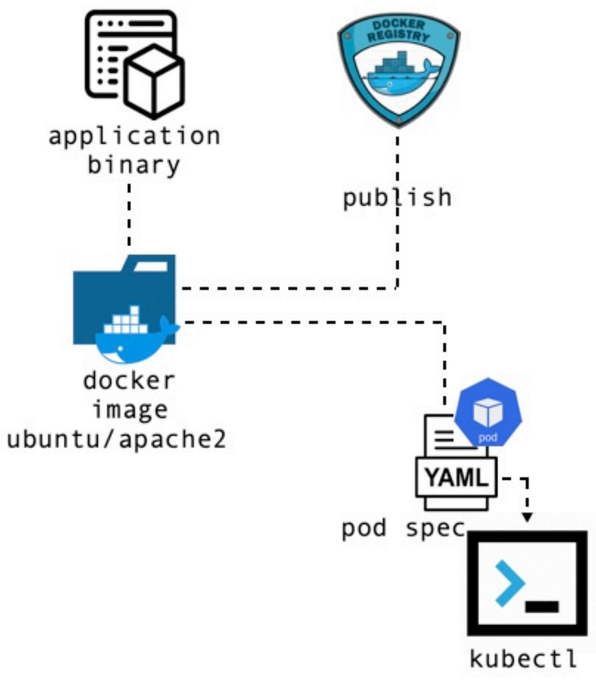
Instead of managing these containers independently, the kuberntes has provided an logical grouping of these containers with Pod.
So any resources or lifecycle operations on pod will be applied across all the containers defined in the Pod and hence managing these containers becomes easy job.

In such case if we have these containers running independently, then the efforts of managing them is duplicated/multiplied because, the same actions has to be performed for both the containers repeatedly
#1
1. start container1
2. immediately start container2

#2
1. stop container1
2. immediately stop container2

#3.
1. run the container1 with volume v1
2. run the container2 with volume v1

How to run a Pod on K8S cluster?



ubuntu/apache2 docker image is already available aspart of the dockerhub, so we can quickly create an podspec file in running the container ontop of the image on the cluster.

apache2-pod.yml

```
---
apiVersion: v1
kind: Pod
metadata:
  name: apache2
  version: 1.0
spec:
  containers:
    - name: apache2container
      image: ubuntu/apache2:latest
      ports:
        - name: http
          containerPort: 80
          protocol: tcp
…
```

apiVersion = indicates the version of the spec template we are using in writing our spec file
kind = type of the k8s object to be created on the cluster
metadata = holds key/value pair of labels, and those are used for identifying or querying the objects from the cluster.
spec = holds the definition information about the object