

Deploy an EKS cluster called playground-cluster with 3 nodes in us-west-2

```

Events:
  Type    Reason      Age    From          Message
  ----    -
  Normal  Scheduled   16m    default-scheduler
  Normal  Pulling     15m    kubelet        Successfully assigned default/apache-deployment-6d6c5765b9-3kljk to ip-192-168-15-127.us-west-2.compute.internal
  Normal  Pulling     15m    kubelet        Pulling image "justin/justin_nginx_image:0.1"
  Normal  Created     15m    kubelet        Successfully pulled image "justin/justin_nginx_image:0.1" in 14.65s (14.65s including waiting). Image size: 190700008 bytes.
  Normal  Started     15m    kubelet        Created container apache-container
  Normal  Completed   15m    kubelet

```

```

Name:         apache-deployment-6d6c5765b9-wnkvr
Namespace:    default
Priority:      0
Service Account:
Node:         ip-192-168-59-188.us-west-2.compute.internal/192.168.59.188
Start Time:   Sat, 24 Aug 2024 05:24:14 -0400
Labels:       app=apache
Annotations:   pod-template-hash=6d6c5765b9
Status:       Running
IP:           192.168.42.101
IPs:
  IP:         192.168.42.101
Docker ID:    592188c42101
Controlled By: ReplicaSet/apache-deployment-6d6c5765b9
Containers:
  apache-container:
    Container ID:   containerd://5677430d6e12593b06848937ed15906f706efc48976c6827ca3383bc
    Image:          docker.io/justin/justin_nginx_image:0.1
    Image ID:       docker.io/justin/justin_nginx_image:sha256-448651482aeb3ec36ba39f3ff7af72ac2fdef07fd6412ccc9b0e5e5
    Port:           80/TCP
    Host Port:      80/TCP
    State:          Running
    Started:        Sat, 24 Aug 2024 05:24:14 -0400
    Ready:          True
    Restart Count:  0
    Environment:    none
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-gkx78 (ro)
Conditions:
  Type              Status
  PodReadyToStartContainers    True
  Initialized         True
  Ready               True
  ContainersReady     True
  PodScheduled        True
Volumes:
  kube-api-access-gkx78:
    Type:              Projected (a volume that contains injected data from multiple sources)
    Type:               Mount
    TokenExpirationSeconds: 3600
    ConfigMapName:      kube-root-ca.crt
    ConfigMapOptional:  
    DownwardAPI:        true
    OwnerReference:      
  DefaultStorage:
    Type:               EmptyDir
    Medium:              Memory
    SizeLimit:           
  DefaultCache:
    Type:               EmptyDir
    Medium:              Memory
    SizeLimit:           
Tolerations:
  node.kubernetes.io/not-ready: NoExecute op=Exists for 300s
  node.kubernetes.io/unreachable: NoExecute op=Exists for 300s

```

```
23767@LAPTOP-458419V5 MINGW64 ~/Desktop/AWS-COURSES/git-github/github-repository/K8s (master)
$ kubect! get service
NAME          TYPE          CLUSTER-IP    EXTERNAL-IP    PORT(S)        AGE
kubernetes    ClusterIP     10.100.0.1    <none>         443/TCP        22m
```

```

17/76LAPTOP-418419V3 MINGW64 ~/Desktop/AWS-COURSES/git-github/github-repository/K8s (master)
$ kubectl get pod
NAME                                READY    STATUS    RESTARTS   AGE
apache-deployment-68dc37b2b9-11g1k  1/1      Running   0           4m10s
apache-deployment-68dc37b2b9-wmhr   1/1      Running   0           4m10s

17/76LAPTOP-418419V3 MINGW64 ~/Desktop/AWS-COURSES/git-github/github-repository/K8s (master)
$ kubectl describe service
Name:                               kubernetes
Namespace:                          default
Labels:                              component=apiserver,provider=kubernetes
Annotations:                         none
Selector:                            none
Type:                                ClusterIP
IP Family Policy:                    SingleStack
IP Families:                         IPv4
IPs:                                 10.100.0.1
Ports:                               80/TCP, 443/TCP
TargetPort:                          80/TCP
Session Affinity:                    None
Events:                              none

17/76LAPTOP-418419V3 MINGW64 ~/Desktop/AWS-COURSES/git-github/github-repository/K8s (master)
$

```

Create a Load Balancer service to expose our apache application on port 80

```

17/76LAPTOP-418419V3 MINGW64 ~/Desktop/AWS-COURSES/git-github/github-repository/K8s (master)
$ kubectl get service
NAME                TYPE        CLUSTER-IP      EXTERNAL-IP      PORT(S)          AGE
apache-service      LoadBalancer  10.100.52.220    afd580abde9404199a4a8bb5aafab42a-1419943033.us-west-2.elb.amazonaws.com  80:30245/TCP    2m10s
kubernetes          ClusterIP    10.100.0.1      none             443/TCP          40s

17/76LAPTOP-418419V3 MINGW64 ~/Desktop/AWS-COURSES/git-github/github-repository/K8s (master)
$

```

The application tested on the browser

The screenshot shows a web browser window with the address bar displaying a URL from Amazon AWS. The page content is a simple text message: "Hello, here our first docker image running apache server !!!". The browser's address bar shows the URL: <https://www.rentfact.com>. The browser's developer tools are open, showing the console with the message: "Hello, here our first docker image running apache server !!!".

Create a GITHUB repository called kubernetes-apache-webapp and upload Dockerfile with all the kubernetes manifest

<https://github.com/Justoaws>