

## ENUNCIADO 9

### Creo primero el master

```
PS C:\Users\palonso\Desktop\Workspace_VSC\KUBERNETESSEMANALES\kubernetes-exercices-PilarAlonsoSTEMDO\soluciones\ejercicio9> kubectl apply -f redis-master.yaml
deployment.apps/redis-master created
PS C:\Users\palonso\Desktop\Workspace_VSC\KUBERNETESSEMANALES\kubernetes-exercices-PilarAlonsoSTEMDO\soluciones\ejercicio9>
```

### Creo después el servicio

```
PS C:\Users\palonso\Desktop\Workspace_VSC\KUBERNETESSEMANALES\kubernetes-exercices-PilarAlonsoSTEMDO\soluciones\ejercicio9> kubectl apply -f redis-master-service.yaml
service/redis-master-svc created
PS C:\Users\palonso\Desktop\Workspace_VSC\KUBERNETESSEMANALES\kubernetes-exercices-PilarAlonsoSTEMDO\soluciones\ejercicio9>
```

```
service/redis-master-svc created
PS C:\Users\palonso\Desktop\Workspace_VSC\KUBERNETESSEMANALES\kubernetes-exercices-PilarAlonsoSTEMDO\soluciones\ejercicio9> kubectl get svc
NAME                TYPE                CLUSTER-IP      EXTERNAL-IP      PORT(S)          AGE
colors-service      ClusterIP           10.101.24.228    <none>            8080/TCP         27m
kubernetes           ClusterIP           10.96.0.1        <none>            443/TCP          26h
redis-master-svc     ClusterIP           10.111.254.159   <none>            6379/TCP         42s
web-svc             NodePort            10.107.31.92     <none>            80:30002/TCP     65m
PS C:\Users\palonso\Desktop\Workspace_VSC\KUBERNETESSEMANALES\kubernetes-exercices-PilarAlonsoSTEMDO\soluciones\ejercicio9>
```

### Ahora crearemos los pods esclavos que son los que accederán al master:

```
PS C:\Users\palonso\Desktop\Workspace_VSC\KUBERNETESSEMANALES\kubernetes-exercices-PilarAlonsoSTEMDO\soluciones\ejercicio9> kubectl apply -f redis-slave.yaml
deployment.apps/redis-slave created
PS C:\Users\palonso\Desktop\Workspace_VSC\KUBERNETESSEMANALES\kubernetes-exercices-PilarAlonsoSTEMDO\soluciones\ejercicio9> kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
colors-54fbfdf68d-5nxx9             1/1     Running   0           42m
colors-54fbfdf68d-hfp5z             1/1     Running   0           42m
colors-54fbfdf68d-lmw8h             1/1     Running   0           42m
redis-master-5f49bfcd7-r4xb5        1/1     Running   0           17m
redis-slave-6ddcd47dc9-hx25c        0/1     ErrImagePull 0           12s
redis-slave-6ddcd47dc9-kz15h        0/1     ErrImagePull 0           12s
web-d-fcfd454f8-kgdp8              1/1     Running   0           80m
web-d-fcfd454f8-pw7hh              1/1     Running   0           80m
PS C:\Users\palonso\Desktop\Workspace_VSC\KUBERNETESSEMANALES\kubernetes-exercices-PilarAlonsoSTEMDO\soluciones\ejercicio9>
```

### Ahora crearé el servicio vinculado a los esclavos

```
PS C:\Users\palonso\Desktop\Workspace_VSC\KUBERNETESSEMANALES\kubernetes-exercices-PilarAlonsoSTEMDO\soluciones\ejercicio9> kubectl apply -f redis-slave-service.yaml
service/redis-slave created
PS C:\Users\palonso\Desktop\Workspace_VSC\KUBERNETESSEMANALES\kubernetes-exercices-PilarAlonsoSTEMDO\soluciones\ejercicio9> kubectl get svc
NAME                TYPE                CLUSTER-IP      EXTERNAL-IP      PORT(S)          AGE
colors-service      ClusterIP           10.101.24.228    <none>            8080/TCP         40m
kubernetes           ClusterIP           10.96.0.1        <none>            443/TCP          26h
redis-master-svc     ClusterIP           10.111.254.159   <none>            6379/TCP         13m
redis-slave          ClusterIP           10.98.1.197      <none>            6379/TCP         26s
web-svc             NodePort            10.107.31.92     <none>            80:30002/TCP     78m
PS C:\Users\palonso\Desktop\Workspace_VSC\KUBERNETESSEMANALES\kubernetes-exercices-PilarAlonsoSTEMDO\soluciones\ejercicio9>
```

### Creamos el frontend:

```
PS C:\Users\palonso\Desktop\Workspace_VSC\KUBERNETESSEMANALES\kubernetes-exercices-PilarAlonsoSTEMDO\soluciones\ejercicio9> kubectl apply -f frontend.yaml
deployment.apps/frontend created
PS C:\Users\palonso\Desktop\Workspace_VSC\KUBERNETESSEMANALES\kubernetes-exercices-PilarAlonsoSTEMDO\soluciones\ejercicio9> kubectl get pods
NAME                                READY   STATUS             RESTARTS   AGE
colors-54fbfdf68d-5nxx9             1/1     Running            0           45m
colors-54fbfdf68d-hfp5z             1/1     Running            0           45m
colors-54fbfdf68d-lmw8h             1/1     Running            0           45m
frontend-784547756c-9hn5x           0/1     ContainerCreating  0           5s
frontend-784547756c-m2k6q           0/1     ErrImagePull       0           5s
frontend-784547756c-svhzp           0/1     ErrImagePull       0           5s
redis-master-5f49bfcd7-r4xb5        1/1     Running            0           20m
redis-slave-6ddcd47dc9-hx25c        0/1     ErrImagePull       0           3m22s
redis-slave-6ddcd47dc9-kz15h        0/1     ImagePullBackOff   0           3m22s
web-d-fcfd454f8-kgdp8              1/1     Running            0           83m
web-d-fcfd454f8-pw7hh              1/1     Running            0           83m
PS C:\Users\palonso\Desktop\Workspace_VSC\KUBERNETESSEMANALES\kubernetes-exercices-PilarAlonsoSTEMDO\soluciones\ejercicio9>
```

### Creamos el servicio del frontend para poder acceder desde el exterior:

```
PS C:\Users\palonso\Desktop\workspace_VSC\KUBERNETESSEMANALES\kubernetes-exercises-PilarAlonsoSTEMDO\soluciones\ejercicio9> kubectl apply -f frontend-service.yaml
service/frontend created
PS C:\Users\palonso\Desktop\workspace_VSC\KUBERNETESSEMANALES\kubernetes-exercises-PilarAlonsoSTEMDO\soluciones\ejercicio9> kubectl get svc
NAME                TYPE        CLUSTER-IP      EXTERNAL-IP      PORT(S)          AGE
colors-service      ClusterIP   10.101.24.228   <none>            8080/TCP          48m
frontend            NodePort    10.102.175.172   <none>            80:32311/TCP      9s
kubernetes           ClusterIP   10.96.0.1        <none>            443/TCP           26h
redis-master-svc     ClusterIP   10.111.254.159   <none>            6379/TCP          21m
redis-slave          ClusterIP   10.98.1.197      <none>            6379/TCP          8m32s
web-svc              NodePort    10.107.31.92     <none>            80:30002/TCP      86m
PS C:\Users\palonso\Desktop\workspace_VSC\KUBERNETESSEMANALES\kubernetes-exercises-PilarAlonsoSTEMDO\soluciones\ejercicio9>
```

Tuve que cambiar los tags de las imágenes porque era lo que me daba error

Una vez teniendo todos los pods running hice minikube service frontend y ya funcionó

