

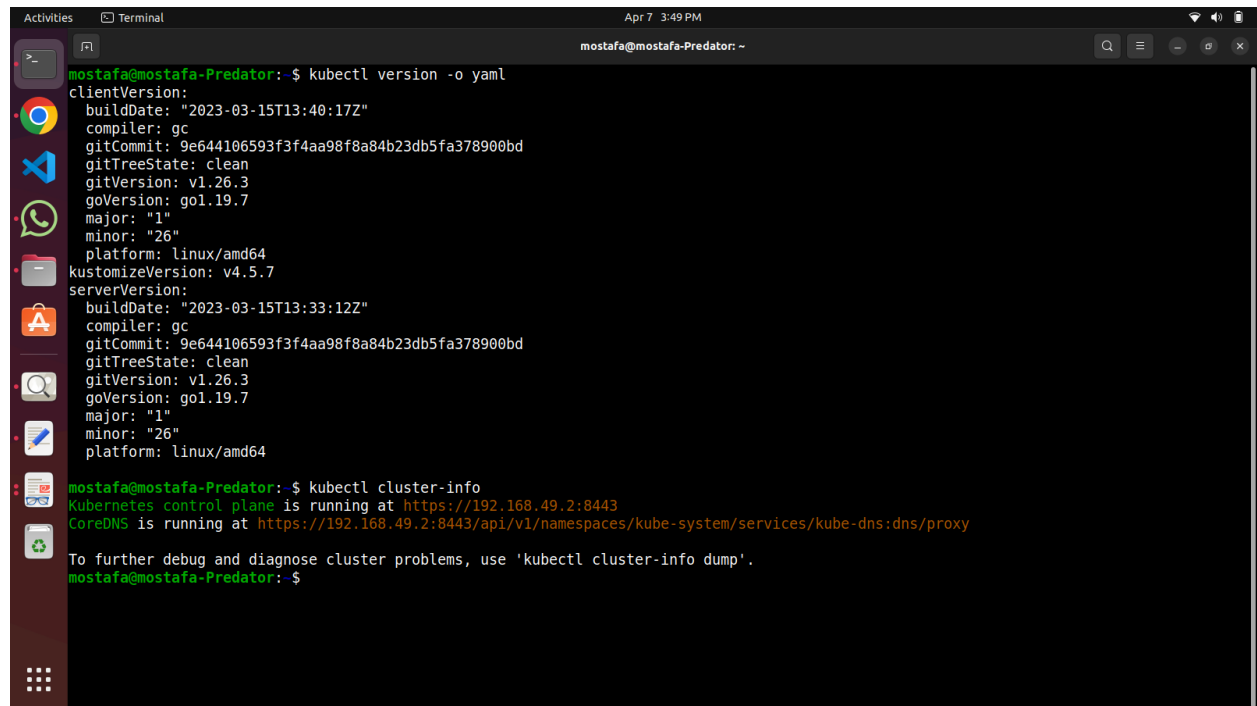
Q1-

```
mostafa@mostafa-Predator:~$ wget https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64
--2023-04-07 15:28:57-- https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64
Resolving storage.googleapis.com (storage.googleapis.com)... 142.250.200.240, 142.250.201.16, 142.250.201.48, ...
Connecting to storage.googleapis.com (storage.googleapis.com)|142.250.200.240|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 83937631 (80M) [application/octet-stream]
Saving to: 'minikube-linux-amd64'

minikube-linux-amd64 28%[=====>] 22.83M 1.07MB/s etminikube-linux-amd64 29%[=====>] 23m
minikube-linux-a 100%[=====] 80.05M 2.97MB/s in 35s

2023-04-07 15:29:33 (2.26 MB/s) - 'minikube-linux-amd64' saved [83937631/83937631]

mostafa@mostafa-Predator:~$ sudo cp minikube-linux-amd64 /usr/local/bin/minikube
mostafa@mostafa-Predator:~$ sudo chmod +x /usr/local/bin/minikube
mostafa@mostafa-Predator:~$ minikube version
minikube version: v1.30.1
commit: 08896fd1dc362c097c925146c4a0d0dac715ace0
mostafa@mostafa-Predator:~$
```

A screenshot of a terminal window titled 'Terminal' with a timestamp of 'Apr 7 3:49 PM'. The terminal shows the output of 'kubectl version -o yaml' and 'kubectl cluster-info'. The 'kubectl version' output shows client and server versions as v1.26.3. The 'kubectl cluster-info' output shows the Kubernetes control plane running at https://192.168.49.2:8443 and CoreDNS running at https://192.168.49.2:8443/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy. The terminal window has a sidebar with application icons and a search bar at the top right.

```
mostafa@mostafa-Predator:~$ kubectl version -o yaml
clientVersion:
  buildDate: "2023-03-15T13:40:17Z"
  compiler: gc
  gitCommit: 9e644106593f3f4aa98f8a84b23db5fa378900bd
  gitTreeState: clean
  gitVersion: v1.26.3
  goVersion: go1.19.7
  major: "1"
  minor: "26"
  platform: linux/amd64
kustomizeVersion: v4.5.7
serverVersion:
  buildDate: "2023-03-15T13:33:12Z"
  compiler: gc
  gitCommit: 9e644106593f3f4aa98f8a84b23db5fa378900bd
  gitTreeState: clean
  gitVersion: v1.26.3
  goVersion: go1.19.7
  major: "1"
  minor: "26"
  platform: linux/amd64

mostafa@mostafa-Predator:~$ kubectl cluster-info
Kubernetes control plane is running at https://192.168.49.2:8443
CoreDNS is running at https://192.168.49.2:8443/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
mostafa@mostafa-Predator:~$
```

Q2-

The screenshot shows a terminal window on the left and a text editor on the right. The terminal shows the user applying a Redis pod configuration, editing it, and then applying it again. The text editor shows the content of the `redis.yaml` file.

```
mostafa@mostafa-Predator:~$ kubectl apply -f redis.yaml
pod/pod-name created
mostafa@mostafa-Predator:~$ kubectl apply -f redis.yaml
pod/redis created
mostafa@mostafa-Predator:~$ gedit redis.yaml
mostafa@mostafa-Predator:~$ kubectl apply -f redis.yaml
pod/redis unchanged
mostafa@mostafa-Predator:~$
```

```
1 apiVersion: v1
2 kind: Pod
3 metadata:
4   name: redis
5 spec:
6   containers:
7     - name: redis-container
8       image: redis:latest
```

The screenshot shows a terminal window on the left and a text editor on the right. The terminal shows the user listing pods, describing the 'nginx' pod, and seeing an error. The text editor shows a new pod definition file named `pod-def.yaml`.

```
mostafa@mostafa-Predator:~$ kubectl get pods
NAME      READY   STATUS    RESTARTS   AGE
nginx     0/1     ErrImagePull  0          37m
pod-name  1/1     Running    0          23m
redis     1/1     Running    0          43m
mostafa@mostafa-Predator:~$ kubectl describe nginx
error: the server doesn't have a resource type "nginx"
mostafa@mostafa-Predator:~$ kubectl describe pod nginx
Name:      nginx
Namespace: default
Priority:   0
Service Account: default
Node:      minikube/192.168.49.2
Start Time: Fri, 07 Apr 2023 16:49:56 +0200
Labels:    <none>
Annotations: <none>
Status:    Pending
IP:        10.244.0.7
IPs:
  IP: 10.244.0.7
Containers:
  nginx-container:
    Container ID:
    Image:        nginx:123
    Image ID:
    Port:         <none>
    Host Port:    <none>
    State:        Waiting
    Reason:       ImagePullBackOff
    Ready:        False
    Restart Count: 0
    Environment:  <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-acce
```

```
1 apiVersion: v1
2 kind: Pod
3 metadata:
4   name: pod-name
5 spec:
6   containers:
7     - name: nginx-container
8       image: nginx:123
```

Q3-

```
Activities Terminal Apr 7 5:30 PM
mostafa@mostafa-Predator: ~
mostafa@mostafa-Predator: ~
Reason: ImagePullBackOff
Ready: False
Restart Count: 0
Environment: <none>
Mounts:
  /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-hjg79 (ro)
Conditions:
  Type Status
  Initialized True
  Ready False
  ContainersReady False
  PodScheduled True
Volumes:
  kube-api-access-hjg79:
    Type: Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds: 3607
    ConfigMapName: kube-root-ca.crt
    ConfigMapOptional: <nil>
    DownwardAPI: true
QoS Class: BestEffort
Node-Selectors: <none>
Tolerations:
  node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
  node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
Events:
  Type Reason Age From Message
  ----
  Normal Scheduled 38m default-scheduler Successfully assigned default/nginx to minikube
  Normal Pulling 36m (x4 over 38m) kubelet Pulling image "nginx:123"
  Warning Failed 36m (x4 over 38m) kubelet Failed to pull image "nginx:123": rpc error: code = Unknown desc = Error
  response from daemon: manifest for nginx:123 not found: manifest unknown: manifest unknown
  Warning Failed 36m (x4 over 38m) kubelet Error: ErrImagePull
  Warning Failed 36m (x6 over 38m) kubelet Error: ImagePullBackOff
  Normal BackOff 3m19s (x151 over 38m) kubelet Back-off pulling image "nginx:123"
mostafa@mostafa-Predator: $
```

Q6-

```
Activities Text Editor Apr 7 5:37 PM
mostafa@mostafa-Predator: ~
mostafa@mostafa-Predator: ~
5360ff5cc1670a85acb5bd85ba1b19c0
Port: <none>
Host Port: <none>
State: Running
Started: Fri, 07 Apr 2023 17:34:08 +0200
Ready: True
Restart Count: 0
Environment: <none>
Mounts:
  /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-rwjmd (ro)
Conditions:
  Type Status
  Initialized True
  Ready True
  ContainersReady True
  PodScheduled True
Volumes:
  kube-api-access-rwjmd:
    Type: Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds: 3607
    ConfigMapName: kube-root-ca.crt
    ConfigMapOptional: <nil>
    DownwardAPI: true
QoS Class: BestEffort
Node-Selectors: <none>
Tolerations:
  node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
  node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
Events:
  Type Reason Age From Message
  ----
  Normal Scheduled 3m27s default-scheduler Successfully assigned default/nginxpod to minikube
  Normal Pulling 3m26s kubelet Pulling image "nginx"
  Normal Pulled 3m24s kubelet Successfully pulled image "nginx" in 1.718133611s (1.718140017s including waiting)
  Normal Created 3m24s kubelet Created container nginx
  Normal Started 3m24s kubelet Started container nginx
mostafa@mostafa-Predator: $

pod-def.yaml
1 apiVersion: v1
2 kind: Pod
3 metadata:
4   name: nginxpod
5 spec:
6   containers:
7   - name: nginx
8     image: nginx
```

```

mostafa@mostafa-Predator:~$ kubectl get replicaset
No resources found in default namespace.
mostafa@mostafa-Predator:~$ kubectl get replicaset | wc -l
0
mostafa@mostafa-Predator:~$

```

Q7-

```

mostafa@mostafa-Predator:~$ kubectl get replicaset
NAME          DESIRED   CURRENT   READY   AGE
replica-set-1  3         3         0       2m36s
mostafa@mostafa-Predator:~$

```

```

home > mostafa > Desktop > iti > Kubernetes > ! replicaset2.yaml > {} spec > # replicas
1  apiVersion: apps/v1
2  kind: ReplicaSet
3  metadata:
4    name: replicaset-1
5  spec: # awl 7aga tbd2 tt8ir f no3 l resource
6    replicas: 4
7    selector:
8      matchLabels:
9        app: nginx
10   template:
11     metadata:
12       labels:
13         app: nginx
14     spec:
15       containers:
16       - image: busybox
17         name: bbx-containers

```

```
mostafa@mostafa-Predator:~$ kubectl apply -f replicaset1.yaml
replicaset.apps/replica-set-1 configured
mostafa@mostafa-Predator:~$ kubectl get replicaset
```

NAME	DESIRED	CURRENT	READY	AGE
replica-set-1	5	5	0	7m17s

Q10,11

Still 5 pods, this is because pods are managed by the deployment not the replica set .

Q12

```
mostafa@mostafa-Predator:~/Desktop/iti/Kubernetes$ kubectl get deploy
```

NAME	READY	UP-TO-DATE	AVAILABLE	AGE
nginx-deployment	1/1	1	1	61m

```
mostafa@mostafa-Predator:~/Desktop/iti/Kubernetes$ kubectl get rs
```

NAME	DESIRED	CURRENT	READY	AGE
nginx-deployment-7f9ff58b4b	1	1	1	62m
nginx-replicaset	1	1	1	82m
replica-set-1	3	3	0	44h
replicaset-1	5	5	0	8m28s

```
mostafa@mostafa-Predator:~/Desktop/iti/Kubernetes$
```

Q13

```
mostafa@mostafa-Predator:~/Desktop/iti/Kubernetes$ kubectl get rs
```

NAME	DESIRED	CURRENT	READY	AGE
deploment-1-556bc9699d	3	3	0	24s
deployment-1-8685c55bc6	3	3	0	6m5s
deployment-13-8685c55bc6	3	3	0	5m19s
nginx-deployment-7f9ff58b4b	1	1	1	72m
nginx-replicaset	1	1	1	93m
replica-set-1	3	3	0	44h
replicaset-1	5	5	0	19m

```
mostafa@mostafa-Predator:~/Desktop/iti/Kubernetes$ kubectl get deploy
```

NAME	READY	UP-TO-DATE	AVAILABLE	AGE
deploment-1	0/3	3	0	34s
deployment-1	0/3	3	0	6m15s
deployment-13	0/3	3	0	5m29s
nginx-deployment	1/1	1	1	72m

```

! deployment2.yaml x
home > mostafa > Desktop > iti > Kubernetes > ! deployment2.yaml > {} spec > {} template > {} spec > [ ] containers > {} 0 > image
1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    name: deploement-1
5    labels:
6      app: nginx
7  spec:
8    replicas: 3
9    selector:
10     matchLabels:
11       app: nginx
12    template:
13     metadata:
14       labels:
15         app: nginx
16     spec:
17       containers:
18         - name: busybox-cont
19           image: busybox

```

## Q15

```

mostafa@mostafa-Predator:~/Desktop/iti/Kubernetes$ kubectl apply -f deployment2.yaml
deployment.apps/deploement-1 configured
mostafa@mostafa-Predator:~/Desktop/iti/Kubernetes$ kubectl get deploy
NAME                READY   UP-TO-DATE   AVAILABLE   AGE
deploement-1        1/3     2            1           12m
deployment-1        0/3     3            0           18m
deployment-13       0/3     3            0           17m
nginx-deployment    1/1     1            1           85m
mostafa@mostafa-Predator:~/Desktop/iti/Kubernetes$

```

## Q16

```

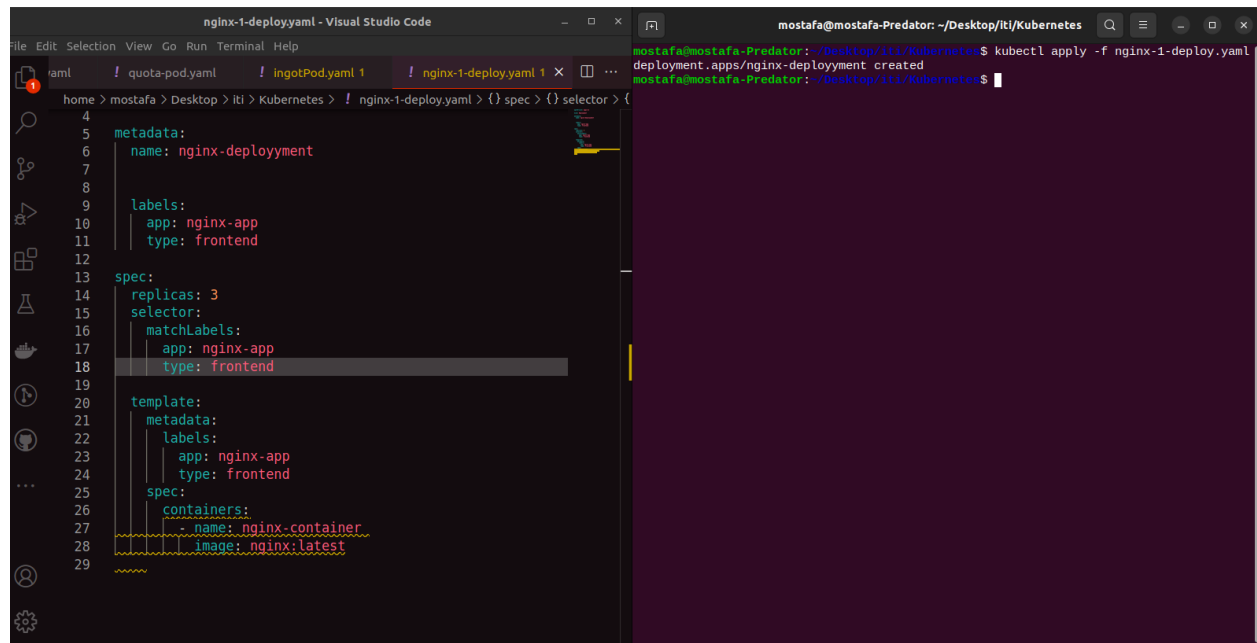
Events:
Type     Reason              Age   From              Message
----     -
Normal   ScalingReplicaSet   18m   deployment-controller   Scaled up replica set deploement-1-556bc9699d to 3
Normal   ScalingReplicaSet   5m25s deployment-controller   Scaled up replica set deploement-1-75775fd9fd to 1
Normal   ScalingReplicaSet   5m22s deployment-controller   Scaled down replica set deploement-1-556bc9699d to 2 from 3
Normal   ScalingReplicaSet   5m22s deployment-controller   Scaled up replica set deploement-1-75775fd9fd to 2 from 1
Normal   ScalingReplicaSet   5m18s deployment-controller   Scaled down replica set deploement-1-556bc9699d to 1 from 2
Normal   ScalingReplicaSet   5m18s deployment-controller   Scaled up replica set deploement-1-75775fd9fd to 3 from 2
Normal   ScalingReplicaSet   5m15s deployment-controller   Scaled down replica set deploement-1-556bc9699d to 0 from 1

```

Rolling Deployment: replace pods running busybox with nginx.

Kubectl rollout undo deploymentname

Q18:



The image shows a Visual Studio Code editor window with a file named `nginx-1-deploy.yaml` open. The file contains a Kubernetes deployment manifest for an nginx application. The manifest includes metadata, labels, a spec with 3 replicas, and a template for the container. The terminal window on the right shows the command `kubectl apply -f nginx-1-deploy.yaml` being executed, resulting in the deployment being created.

```
nginx-1-deploy.yaml - Visual Studio Code
File Edit Selection View Go Run Terminal Help

home > mostafa > Desktop > iti > Kubernetes > ! nginx-1-deploy.yaml > {} spec > {} selector > {}

4
5
6 metadata:
7   name: nginx-deployment
8
9
10 labels:
11   app: nginx-app
12   type: frontend
13
14 spec:
15   replicas: 3
16   selector:
17     matchLabels:
18       app: nginx-app
19       type: frontend
20
21 template:
22   metadata:
23     labels:
24       app: nginx-app
25       type: frontend
26   spec:
27     containers:
28       - name: nginx-container
29         image: nginx:latest

mostafa@mostafa-Predator: ~/Desktop/iti/Kubernetes$ kubectl apply -f nginx-1-deploy.yaml
deployment.apps/nginx-deployment created
mostafa@mostafa-Predator: ~/Desktop/iti/Kubernetes$
```