Q1) Create a Pod with three busy box containers with commands “ls; sleep 3600;”, “echo Hello World; sleep 3600;” and “echo this is the third container; sleep 3600” respectively and check the status
Ans:

// first create single container pod with dry run flag

$ kubectl run busybox --image=busybox --restart=Never --dry-run=client -o yaml -- bin/sh -c "ls; sleep 3600" > multi-container.yaml

// edit the pod to following yaml and create it

$ vi multi-container.yaml

```
apiVersion: v1
kind: Pod
metadata:
  labels:
    run: busybox
  name: busybox
spec:
  containers:
  - args:
    - bin/sh
    - -c
    - ls; sleep 3600
    image: busybox
    name: busybox1
    resources: {}
  - args:
    - bin/sh
    - -c
    - echo Hello world; sleep 3600
    image: busybox
    name: busybox2
    resources: {}
  - args:
    - bin/sh
    - -c
    - echo this is third container; sleep 3600
    image: busybox
    name: busybox3
```

$ kubectl create -f multi-container.yaml

Verfiy

$ kubectl get pods busybox

```
root@Master:~# kubectl run busybox --image=busybox --restart=Never --dry-run=client -o yaml -- bin/sh -c "ls; sleep 3600" > multi-container.yaml
root@Master:~# vi multi-container.yaml
root@Master:~# cat multi-container.yaml
apiVersion: v1
kind: Pod
metadata:
  labels:
    run: busybox
  name: busybox
spec:
  containers:
  - args:
    - bin/sh
    - -c
    - ls; sleep 3600
    image: busybox
    name: busybox1
    resources: {}
  - args:
    - bin/sh
    - -c
    - echo Hello world; sleep 3600
    image: busybox
    name: busybox2
    resources: {}
  - args:
    - bin/sh
    - -c
    - echo this is third container; sleep 3600
    image: busybox
    name: busybox3

root@Master:~# kubectl create -f multi-container.yaml
pod/busybox created
root@Master:~# kubectl get pods busybox
NAME      READY   STATUS    RESTARTS   AGE
busybox   3/3     Running   0          6s
root@Master:~#
```

Q2) Check the logs of each container that you just created
Ans:

```
$ kubectl logs busybox -c busybox1
$ kubectl logs busybox -c busybox2
$ kubectl logs busybox -c busybox3
```

```
root@Master:~# kubectl logs busybox -c busybox1
bin
dev
etc
home
proc
root
sys
tmp
usr
var
root@Master:~# kubectl logs busybox -c busybox2
Hello world
root@Master:~# kubectl logs busybox -c busybox3
this is third container
root@Master:~#
```

Q3) Run command “ls” in the third container busybox3 of the above pod and check file system
Ans:

```
$ kubectl exec busybox -c busybox3 -- ls
```

```
root@Master:~# kubectl exec busybox -c busybox3 -- ls
bin
dev
etc
home
proc
root
sys
tmp
usr
var
root@Master:~#
```

Q4) Create a Pod with main container busybox and which executes this “while true; do echo ‘Hi I am from Main container’ >> /var/log/index.html; sleep 5; done” and with sidecar container with nginx image which exposes on port 80. Use emptyDir Volume and mount this volume on path /var/log for busybox and on path /usr/share/nginx/html for nginx container. Verify both containers are running.
Ans:

```
// create an initial yaml file with this
$ kubectl run multi-cont-pod --image=busbox --dry-run=client -o yaml > multi-container-new.yaml

// edit the yml as below and create it
$ vi multi-container-new.yaml
```

```
apiVersion: v1
kind: Pod
metadata:
  labels:
    run: multi-cont-pod
  name: multi-cont-pod
spec:
  volumes:
  - name: var-logs
    emptyDir: {}
  containers:
  - image: busybox
    command: ["/bin/sh"]
    args: ["-c", "while true; do echo 'Hi I am from Main container' >> /var/log/index.html; sleep
5;done"]
    name: main-container
    volumeMounts:
    - name: var-logs
      mountPath: /var/log
  - image: nginx
    name: sidecar-container
```

```
    ports:
      - containerPort: 80
    volumeMounts:
    - name: var-logs
      mountPath: /usr/share/nginx/html
```

$ kubectl create -f multi-container.yaml

verify if pods created or not

$ kubectl get pods multi-cont-pod

```
root@Master:~# kubectl run multi-cont-pod --image=busbox --dry-run=client -o yaml > multi-container-new.yaml
root@Master:~# vi multi-container-new.yaml
root@Master:~# cat multi-container
cat: multi-container: No such file or directory
root@Master:~# cat multi-container-new.yaml
apiVersion: v1
kind: Pod
metadata:
  labels:
    run: multi-cont-pod
  name: multi-cont-pod
spec:
  volumes:
  - name: var-logs
    emptyDir: {}
  containers:
  - image: busybox
    command: ["/bin/sh"]
    args: ["-c", "while true; do echo 'Hi I am from Main container' >> /var/log/index.html; sleep 5;done"]
    name: main-container
    volumeMounts:
    - name: var-logs
      mountPath: /var/log
  - image: nginx
    name: sidecar-container
    ports:
      - containerPort: 80
    volumeMounts:
    - name: var-logs
      mountPath: /usr/share/nginx/html
root@Master:~# kubectl create -f multi-container-new.yaml
pod/multi-cont-pod created
root@Master:~# kubectl get pods multi-cont-pod
NAME             READY   STATUS              RESTARTS   AGE
multi-cont-pod   0/2     ContainerCreating   0          10s
root@Master:~#
```

Q5) Exec into both containers and verify that index.html exist and query the index.html from sidecar container with curl localhost
Ans:

// exec into main container
$ kubectl exec -it multi-cont-pod -c main-container -- sh
$ cat /var/log/index.html

$ exit

```
// exec into sidecar container
$ kubectl exec -it  multi-cont-pod -c sidecar-container -- sh
$ cat /usr/share/nginx/html/index.html

$ exit

// install curl and get default page
$ kubectl exec -it  multi-cont-pod -c sidecar-container -- sh
# apt-get update && apt-get install -y curl
# curl localhost
```

```
root@Master:~# kubectl exec -it  multi-cont-pod -c main-container -- sh
/ # cat /var/log/main.txt
cat: can't open '/var/log/main.txt': No such file or directory
/ # cat /var/log/index.html
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
/ #
```

```
root@Master:~# kubectl exec -it  multi-cont-pod -c sidecar-container -- sh
# cat /usr/share/nginx/html/index.html
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
```

```
root@Master:~# kubectl exec -it  multi-cont-pod -c sidecar-container -- sh
# apt-get update && apt-get install -y curl
Get:1 http://deb.debian.org/debian buster InRelease [121 kB]
Get:2 http://deb.debian.org/debian buster-updates InRelease [51.9 kB]
Get:3 http://security.debian.org/debian-security buster/updates InRelease [65.4 kB]
Get:4 http://deb.debian.org/debian buster/main amd64 Packages [7907 kB]
Get:5 http://security.debian.org/debian-security buster/updates/main amd64 Packages [269 kB]
Get:6 http://deb.debian.org/debian buster-updates/main amd64 Packages [9504 B]
Fetched 8424 kB in 2s (4203 kB/s)
Reading package lists... Done
Reading package lists... Done
Building dependency tree
Reading state information... Done
curl is already the newest version (7.64.0-4+deb10u1).
0 upgraded, 0 newly installed, 0 to remove and 5 not upgraded.
# curl localhostcurl localhost^H^H^H^H^H^H^H^H^H^H^H^H^H^H^H^H^H^H^C
# ^[[A^C
# curl localhost
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
Hi I am from Main container
```