Exercise 4

Container Networking

Q1.
Check all currently available networks.

Q2.
Create a new network named “production”.
Create another new network named “development”.
Create another new network named “deployment”.
Connect the already running container “container3” to network “production”.

Q3.
Connect a new container “container4” with network “development”.
Also bind the port 80 of the container with port 8082 of the host machine.
Check if port binding was successful or not.
Remove all the networks which are unused by any container.

Q4.
Inspect the “production” network.
Delete the “production” network.

Solutions

Q1.

```
[root@localhost ~]# podman network ls
NETWORK ID    NAME        DRIVER
2f259bab93aa  podman      bridge
[root@localhost ~]#
```

Q2.

```
[root@localhost ~]# podman network create production
production
[root@localhost ~]# podman network create development
development
[root@localhost ~]# podman network create deployment
deployment
[root@localhost ~]# podman network ls
NETWORK ID    NAME         DRIVER
593fd3c1768a  deployment   bridge
c2f306c9302b  development  bridge
2f259bab93aa  podman       bridge
fb4191651689  production   bridge
[root@localhost ~]#
```

```
[root@localhost ~]# podman network connect production container3
```

```
[root@localhost ~]# podman inspect container3 | grep -i network
          "NetworkSettings": {
               "Networks": {
                         "NetworkID": "podman",
                         "NetworkID": "production",
               "NetworkMode": "bridge",
```

Q3.

```
[root@localhost ~]# podman run -d --name container4 -p 8082:80 --network=development httpd
b1bc97952d34737ca478dd379fb7171e07e9f5db5b288995c2fead571597f9bf
[root@localhost ~]# podman port --all
b1bc97952d34    80/tcp -> 0.0.0.0:8082
d84f18916306    80/tcp -> 0.0.0.0:8081
[root@localhost ~]#
```

```
[root@localhost ~]# podman network ls
NETWORK ID    NAME         DRIVER
593fd3c1768a  deployment   bridge
c2f306c9302b  development  bridge
865b411f9b56  example      bridge
2f259bab93aa  podman       bridge
[root@localhost ~]# podman network prune
WARNING! This will remove all networks not used by at least one container.
Are you sure you want to continue? [y/N] y
deployment
development
example
[root@localhost ~]#
```

Q4.

```
[root@localhost ~]# podman network inspect production
[
     {
          "name": "production",
          "id": "fb41916516893dccce509a24fdf551d2b8befcd8ade86063b3b0909851b3fc37",
          "driver": "bridge",
          "network_interface": "podman1",
          "created": "2023-09-28T20:21:39.475143182+05:30",
          "subnets": [
               {
                    "subnet": "10.89.0.0/24",
                    "gateway": "10.89.0.1"
               }
          ],
          "ipv6_enabled": false,
          "internal": false,
          "dns_enabled": true,
          "ipam_options": {
               "driver": "host-local"
          }
     }
]
```

```
[root@localhost ~]# podman network rm -f production
production
[root@localhost ~]# podman network ls
NETWORK ID    NAME         DRIVER
593fd3c1768a  deployment   bridge
c2f306c9302b  development  bridge
2f259bab93aa  podman       bridge
[root@localhost ~]#
```