Step 1

Display existing Docker networks and names on your host

$ docker network ls

```
NETWORK ID      NAME            DRIVER      SCOPE

3092c22e2f99    bridge          bridge      local

5ee8aa70a57c    docker_gwbridge bridge      local

d50effa7e823    host            host        local

koijigc21abd    ingress         overlay     swarm

1c1832dfeb9f    none            null        local
```

Step 2

Create a new network as follows:

* use the 'bridge' driver

* assigns IP addresses in the '192.168.1.0/24' network range

* uses a gateway address of 192.168.1.250

* called 'dev_bridge'

$ docker network create --driver=bridge --subnet=192.168.1.0/24 --gateway=192.168.1.250 dev_bridge

```
9172f9fed208df5717df937dc7d5e284ce6a32b2e7ce72ffeaa325c169fdb842
```

Step 3

Display all Docker networks on the host

$ docker network ls

```
NETWORK ID      NAME            DRIVER      SCOPE

3092c22e2f99    bridge          bridge      local

9172f9fed208    dev_bridge      bridge      local

5ee8aa70a57c    docker_gwbridge bridge      local

d50effa7e823    host            host        local

koijigc21abd    ingress         overlay     swarm

1c1832dfeb9f    none            null        local
```

Step 4

Pull the 'httpd' image and install locally

$ docker pull httpd

```
Using default tag: latest

latest: Pulling from library/httpd

f49cf87b52c1: Pull complete

02ca099fb6cd: Pull complete

de7acb18da57: Pull complete

770c8edb393d: Pull complete

0e252730aeae: Pull complete

6e6ca341873f: Pull complete

2daffd0a6144: Pull complete

Digest: sha256:b5f21641a9d7bbb59dc94fb6a663c43fbf3f56270ce7c7d51801ac74d2e70046

Status: Downloaded newer image for httpd:latest
```

Step 5

Create a container called 'testweb' based on the image in the previous step as follows:

* assigned to new 'dev_bridge' network on start

$ docker run -d --name testweb --network=dev_bridge httpd

```
8d6ce33684248269da414fc3f8cd2b084ce6b7c7874612786736377 64c008a6a
```

Step 6

Using the appropriately formatted Docker command output, display the container's IP(s) to include the new network

$ docker container inspect --format="{{.NetworkSettings.Networks.dev_bridge.IPAddress}}" testweb

```
192.168.1.1
```