QUES 6 : **Create a Custom Docker Bridge Network**

SOLN :

1. Create a custom bridge network

```
docker network create my-bridge-network
```

---

2. Run the backend container

Let's assume your backend is listening on port `5000` inside the container:

```
docker run -d \
  --name backend \
  --network my-bridge-network \
  -p 5000:5000 \
  your-backend-image
```

This will expose backend internally to other containers as `http://backend:5000`.

---

3. Run the frontend container

Now run the frontend and connect it to the same network:

```
docker run -d \
  --name frontend \
  --network my-bridge-network \
  -p 3000:3000 \
  -e BACKEND_URL=http://backend:5000 \
  your-frontend-image
```

> Replace `-e BACKEND_URL=...` with the correct env variable if your frontend expects backend URL that way.

```
C:\Users\prate>docker run -d --name backend --network my-bridge-network -p 5000:5000 myprotfolio-backend
ea32a731564df3c78c8f0aacecf24061621a10f01dea7c0ed169cb20c2b192aa

C:\Users\prate>docker run -d --name frontend --network my-bridge-network -p 3000:3000 -e BACKEND_URL=http://backend:5000 myprotfolio-frontend
070c892461926f476548ee713d4f36a9eedff22e84c3f0bce8ff3147ad504c07

C:\Users\prate>
```

---

4. Verify both containers

```
docker ps
```

```
C:\Users\prate>docker ps -a
CONTAINER ID   IMAGE                    COMMAND                  CREATED          STATUS          PORTS                      NAMES
070c89246192   myprotfolio-frontend     "/docker-entrypoint.…"   52 seconds ago   Up 51 seconds   80/tcp, 0.0.0.0:3000->3000/tcp   frontend
ea32a731564d   myprotfolio-backend      "docker-entrypoint.s…"   About a minute ago   Up About a minute   0.0.0.0:5000->5000/tcp   backend
```

Check logs if needed:

```
docker logs frontend
docker logs backend
```

◈ 5. Test inter-container communication

You can test from inside one container:

```
docker exec -it frontend ping backend
```

Or curl the backend API:

```
docker exec -it frontend apk add curl  # for Alpine-based images
docker exec -it frontend curl http://backend:5000/your-endpoint
```

BRIGED NETWORK WORKING

pinged backend from frontend container

```
C:\Users\prate>docker exec -it frontend ping backend
PING backend (172.19.0.2): 56 data bytes
64 bytes from 172.19.0.2: seq=0 ttl=64 time=2.027 ms
64 bytes from 172.19.0.2: seq=1 ttl=64 time=0.317 ms
64 bytes from 172.19.0.2: seq=2 ttl=64 time=0.325 ms
64 bytes from 172.19.0.2: seq=3 ttl=64 time=0.253 ms
64 bytes from 172.19.0.2: seq=4 ttl=64 time=0.297 ms
```

BUT FIRST OF ALL LETS CREATE AN UPDATED

DOCKER-COMPOSE.YML FILE

```yaml
version: '3.9'

services:
  backend:
    image: my-backend:latest
    container_name: backend
    build:
      context: ./backend
      dockerfile: Dockerfile
    ports:
      - "5000:5000"
    networks:
      - my-network

  frontend:
    image: my-frontend:latest
    container_name: frontend
    build:
      context: ./frontend
      dockerfile: Dockerfile
    ports:
      - "3000:3000"
    environment:
      - BACKEND_URL=http://backend:5000
    depends_on:
      - backend
    networks:
      - my-network

networks:
  my-network:
    driver: bridge
```

AND IF WE WANT TO ASSIGN AN IP TO THE PORT THE FILE WILL BE

नेक ; ocr_output.t cleane output (1).tx output (2).tx output (4).tx output.txt newne Paragr

File Edit View

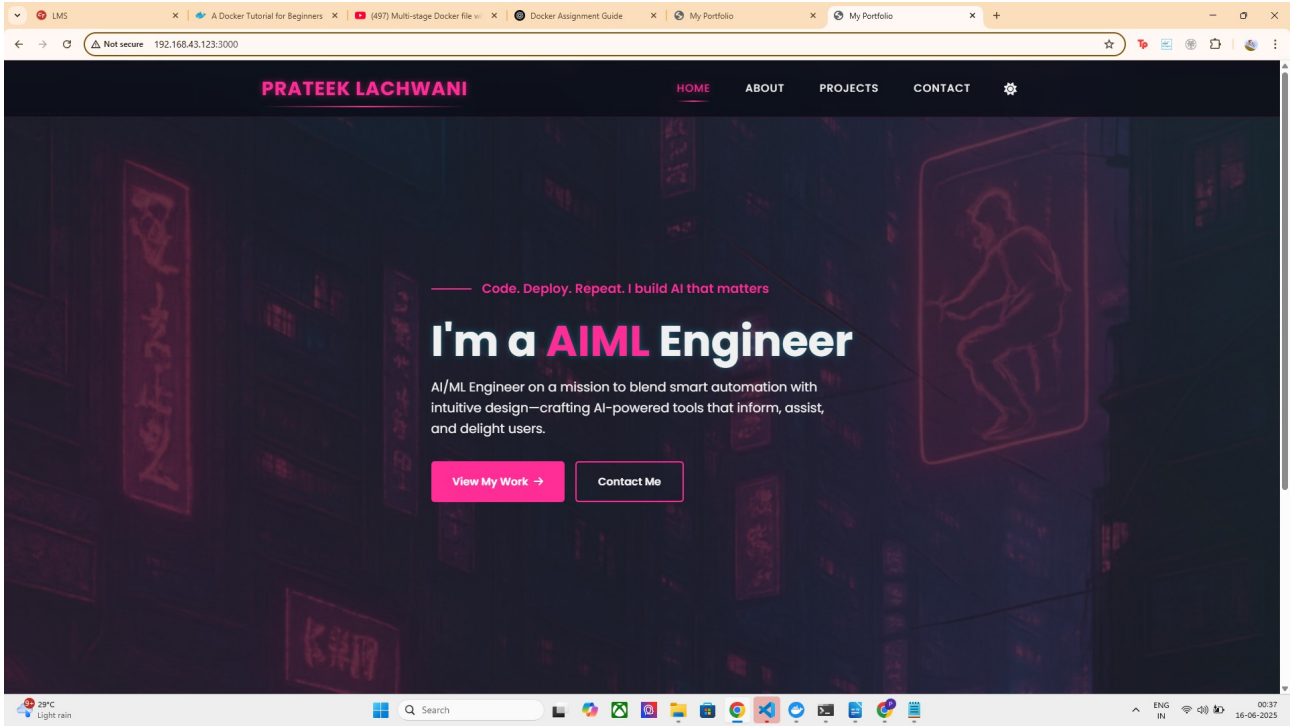
```yaml
version: '3.9'

services:
  backend:
    image: my-backend:latest
    container_name: backend
    build:
      context: ./backend
      dockerfile: Dockerfile
    ports:
      - "5000:5000"
    networks:
      my-custom-network:
        ipv4_address: 172.20.0.2

  frontend:
    image: my-frontend:latest
    container_name: frontend
    build:
      context: ./frontend
      dockerfile: Dockerfile
    ports:
      - "3000:80"
    environment:
      - BACKEND_URL=http://172.20.0.2:5000
    depends_on:
      - backend
    networks:
      my-custom-network:
        ipv4_address: 172.20.0.3

networks:
  my-custom-network:
    driver: bridge
    ipam:
      config:
        - subnet: 172.20.0.0/16
```

Ln 23, Col 17 701 characters

9+ 29°C Light rain Search

docker-compose up --build -d

SITE WORKING ON LOCAL IP : http://192.168.43.123:3000/



TO VERIFY THE BRIGED NETWORK CONFIGURATION

COMMAND USED : docker network inspect myprotfolio_my-custom-network

OUTPUT :


```
[
  {
    "Name": "myprotfolio_my-custom-network",
    "Id": "a7dd79034dfbf10205a08e106ef765617c589fad9dcd309216f35a3b927560d9",
    "Created": "2025-06-15T19:00:50.68327118Z",
    "Scope": "local",
    "Driver": "bridge",
    "EnableIPv4": true,
    "EnableIPv6": false,
```

"IPAM": {
"Driver": "default",
"Options": null,
"Config": [
{
"Subnet": "172.20.0.0/16"
}
]
},
"Internal": false,
"Attachable": false,
"Ingress": false,
"ConfigFrom": {
"Network": ""
},
"ConfigOnly": false,
"Containers": {
"60ac0f2025394560497b1e822ce7d9df47c6a964c3b6a22e235661b1304a33c7": {
"Name": "backend",
"EndpointID": "10b641152aa45cef507e299669cf47d75b05cd9abfcb017ee5eb1c8e4fca3f18",
"MacAddress": "72:11:cd:4c:22:f1",
"IPv4Address": "172.20.0.2/16",
"IPv6Address": ""
},
"a112a04efbbb0f034c1f9efcdc8a38cf7ef3740b3642743d5b4fd7fe60250f68": {
"Name": "frontend",
"EndpointID": "81a43b4b2bf32b0c127acd1ddd443f8dcd5683174ee0a7d6102828629a72903c",
"MacAddress": "9e:e5:9f:8d:14:e9",
"IPv4Address": "172.20.0.3/16",
"IPv6Address": ""

```
        }
      },
      "Options": {},
      "Labels": {
        "com.docker.compose.config-hash": "be36c59f09b3b8739be5efee8ae97efb106038a1f1446730109dff71f303606d",
        "com.docker.compose.network": "my-custom-network",
        "com.docker.compose.project": "myprotfolio",
        "com.docker.compose.version": "2.34.0"
      }
    }
]
```


HENCE ITS SUCCESSFULLY WORKING :

Criteria	Output
Network Name	`myprotfolio_my-custom-network`
Driver	`bridge` (standard for internal container networking)
Containers Connected	`backend`, `frontend`
IP Addresses Assigned	`172.20.0.2` (backend), `172.20.0.3` (frontend)
Visible in Docker Network Inspect	✔ Yes
Docker Compose Labels	✔ Present (e.g., `com.docker.compose.network`)

```
CONTAINER ID   IMAGE                                          COMMAND                  CREATED          STATUS          PORTS                    NAMES
a112a04efbbb   my-frontend:latest                             "/docker-entrypoint.…"   12 minutes ago   Up 12 minutes   0.0.0.0:3000->80/tcp     frontend
60ac0f202539   my-backend:latest                              "docker-entrypoint.s…"   12 minutes ago   Up 12 minutes   0.0.0.0:5000->5000/tcp   backend
aa669a9e549c   myreistryforimage.azurecr.io/my-image-acr:latest   "/docker-entrypoint.…"   41 minutes ago   Up 41 minutes   0.0.0.0:7777->80/tcp     serene_buck
a191430c6cb2   prateek2004/my-frontend:latest                 "/docker-entrypoint.…"   3 hours ago      Up 3 hours      0.0.0.0:8080->80/tcp     angry_shamir
PS D:\OneDrive\Desktop\my protfolio> docker exec -it frontend ping backend
PING backend (172.20.0.2): 56 data bytes
64 bytes from 172.20.0.2: seq=0 ttl=64 time=1.613 ms
64 bytes from 172.20.0.2: seq=1 ttl=64 time=0.273 ms
64 bytes from 172.20.0.2: seq=2 ttl=64 time=0.313 ms
64 bytes from 172.20.0.2: seq=3 ttl=64 time=0.265 ms
64 bytes from 172.20.0.2: seq=4 ttl=64 time=0.314 ms
64 bytes from 172.20.0.2: seq=5 ttl=64 time=0.530 ms
64 bytes from 172.20.0.2: seq=6 ttl=64 time=0.319 ms
64 bytes from 172.20.0.2: seq=7 ttl=64 time=0.080 ms
```