Docker Lab Guide for Students

Lab 1: Install Docker

On Ubuntu:

```
sudo apt update
sudo apt install docker.io -y
sudo systemctl start docker
sudo systemctl enable docker
sudo usermod -aG docker $USER
```

Then log out and log back in.

Test: docker --version

Lab 2: Run Your First Container

```
docker run hello-world
```

This checks if Docker is installed and works.

Lab 3: Run an Interactive Container

```
docker run -it alpine sh
```

Inside the container:

```
echo "Welcome to Docker"
exit
```

Lab 4: List Images and Containers

```
docker images        # List images

docker ps            # Running containers

docker ps -a         # All containers
```

Lab 5: Build Your Own Image

Dockerfile:

```
FROM alpine

RUN echo "Hello from custom Docker image!" > /hello.txt

CMD ["cat", "/hello.txt"]
```

Build and run:

```
docker build -t my-alpine .

docker run my-alpine
```

Lab 6: Port Mapping with Nginx

```
docker run -d -p 8080:80 nginx
```

Open http://localhost:8080 in browser.

Lab 7: Docker Volumes

```
docker volume create mydata

docker run -it -v mydata:/data alpine sh
```

Inside:

```
echo "saved data" > /data/file.txt
```

```
exit
```

Check with:

```
docker run -it -v mydata:/data alpine cat /data/file.txt
```

Lab 8: Docker Networking

```
docker network create mynet
docker run -dit --name c1 --network mynet alpine sh
docker run -dit --name c2 --network mynet alpine sh

docker exec -it c1 sh
ping c2
```

Lab 9: Docker Compose Example

docker-compose.yml:

```
version: '3'
services:
  web:
    image: nginx
    ports:
      - "8080:80"
  redis:
    image: redis
```

Run: docker-compose up -d

Lab 10: Clean Up

```
docker rm -f $(docker ps -aq)
docker rmi -f $(docker images -q)
docker volume prune -f
docker network prune -f
```

Bonus Lab: Flask App in Docker

Files: app.py, requirements.txt, Dockerfile

app.py:

```
from flask import Flask
app = Flask(__name__)
@app.route("/")
def hello(): return "Hello from Flask in Docker!"
```

requirements.txt:

```
flask
```

Dockerfile:

```
FROM python:3.9
WORKDIR /app
COPY . .
```

```
RUN pip install -r requirements.txt

CMD ["python", "app.py"]
```

Build: docker build -t flaskapp .

Run: docker run -p 5000:5000 flaskapp

Access: http://localhost:5000