

1) Creating an Image in docker

```
docker image build -t myhello .
```

2) Creating and Running a Container

```
docker container run -p 9999:8888 myhello
```

3) Creating an image in Docker Hub

```
docker image tag myhello ravish1608/myhello
```

4) Push an image in Docker Hub

```
docker image push ravish1608/myhello
```

5) Creating an Container and run it in Docker hub.

```
docker container run -p 9999:8888 ravish1608/myhello
```

Kubernetes

1) Creating Pods

```
kubectI run demo --image=ravish1608/myhello --port=9999 --labels app=demo1
```

2) Forwarding Ports

```
kubectI port-forward pod/demo 9999:8888
```

3) Displaying Pods

```
get pods --selector app=demo1
```

```
kubectI get pod
```

4) Stop or delete pod

```
kubectI delete pod demo1
```

Dockerfile

Use the official Python image from the Docker Hub

FROM python:3.9-alpine

Set the working directory in the container

WORKDIR /usr/src/app

Copy the requirements file into the container

COPY requirements.txt ./

Install any needed packages specified in requirements.txt

RUN pip install --no-cache-dir -r requirements.txt

Copy the rest of the application code into the container

COPY . .

Run the application

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

App.py

```
from flask import Flask
```

```
app = Flask(__name__)
```

```
@app.route('/')
```

```
def hello_world():
```

```
    return 'Ravish'
```

```
if __name__ == '__main__':
```

```
    app.run(host='0.0.0.0', port=8888)
```

Requirement.txt

Flask

Werkzeug

