

Part 1 Dockerfile&Containerize

```
systemctl stop packagekit
```

```
sudo apt update
```

```
systemctl stop httpd.service
```

```
mkdir lab
```

```
cd lab
```

```
app.py python kodunu kullan
```

```
cat > app.py
```

```
from flask import Flask
```

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

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

```
def test():
```

```
    return "<h1>Flask Server is working!</h1>"
```

```
if __name__ == "__main__":
```

```
    app.run(host="0.0.0.0",port=int("5000"), debug=True)
```

```
#python imajini pull et: docker pull python:3.9.23
```

```
#Dockerfile olusturmadan once requirements.txt dosyasi olusturalim
```

```
cat > requirements.txt
```

```
flask
```

```
#simdi Dockerfile: cat > Dockerfile
```

```
FROM python:3.9.23
```

```
COPY ./*.py /
```

```
COPY ./requirements.txt /
```

```
RUN pip install -r requirements.txt
```

```
EXPOSE 8080
```

```
CMD ["python3.9","app.py"]
```

#python kodunu container imajina donustur

docker build -t myimage .

docker image ls

#kendi container'imizi yaratalim

docker container run -d -p 8080:5000 --name mycontainer myimage

localhost:8080

192.168.56.111.:8080

docker container stop mycontainer

docker container prune

#Sayisal loto icin python kodumuz

```
import random as rd
```

```
loto = []
```

```
while len(loto) < 6:
```

```
    sayi = rd.randint(1,99)
```

```
    if sayi not in loto:
```

```
        loto.append(sayi)
```

```
loto.sort()
```

```
print(loto)
```

#bu kodu flask kodumuzu entegre edelim #

bunun adi da myapp2.py olsun

mkdir lab2

cd lab2

```
cat >app2.py
```

```
from flask import Flask
```

```
import random as rd
```

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

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

```
def test():
```

```
    loto = []
```

```
    while len(loto) < 6:
```

```
        sayi = rd.randint(1,99)
```

```
        if sayi not in loto:
```

```
            loto.append(sayi)
```

```
    loto.sort()
```

```
    result=f'Winning Lottery numbers are: {loto}'
```

```
    return "<h1>Flask Server is working!</h1>" + result
```

```
if __name__ == "__main__":
```

```
    app.run(host="0.0.0.0",port=int("8000"), debug=True)
```

```
python3 app2.py
```

```
localhost:8000
```

```
#lab2 directory'sinde
```

```
cat > requirements.txt
```



flask

#Dockerfile

FROM python:3.9.23

COPY ./*.py /

COPY ./requirements.txt /

RUN pip install -r requirements.txt

EXPOSE 80

CMD ["python3.9", "app2.py"]

#python kodunu container imajina donustur

docker build -t myimage2 .

docker image ls

#kendi container'imizi yaratalim

docker container run -d -p 80:8000 --name mycontainer2 myimage2

localhost:80

192.168.56.111

docker image tag myimage2:latest eunvan/mylotoimage2-july:1.2_2025

docker image ls

docker login

docker image push eunvan/mylotoimage2-july:1.2_2025

#docker hub'a push ettigin imaji local repository'inden sil



```
docker image rm eunvan/mylotoimage2-july:1.2_2025
```

#simdi de docker hub'tan ayni imaji pull et

```
docker pull eunvan/mylotoimage2-july:1.2_2025
```

```
docker image ls
```

Part 2 Dockerfile&Containerize An API

```
mkdir project
```

```
cd project
```

```
cat > calculator.py
```

```
from flask import Flask
```

```
from flask_restx import Resource, Api, reqparse
```

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

```
api=Api(app)
```

```
carpim_parser=reqparse.RequestParser()
```

```
carpim_parser.add_argument('a',type=int,required=True,help='a sayisini yaz')
```

```
carpim_parser.add_argument('b',type=int,required=True,help='b sayisini yaz')
```

```
carpim_parser.add_argument('secim',choices=('toplama','cikarma','carpma','bolme'))
```

```
@api.route('/calculator')
```

```
class calculator(Resource):
```

```
    @api.expect(carpim_parser)
```

```
    def get(self):
```

```
        args=carpim_parser.parse_args()
```

```
        if args.secim=='toplama':
```

```
            return args.a+args.b
```

```
        elif args.secim=='cikarma':
```

```
            return args.a-args.b
```

```
        elif args.secim=='carpma':
```

```
            return args.a*args.b
```

```
        elif args.secim=='bolme' and args.b!=0:
```

```
            return args.a/args.b
```

```
        elif args.secim=='bolme' and args.b==0:
```

```
            return 'matematik bu islem olmaz'
```

```
        elif args.secim==None:
```

```
            return 'secim yap'
```

```
if __name__ == "__main__":
```

```
    app.run(host="0.0.0.0", port=int("8080"), debug=True)
```

```
cat > requirements.txt
```

```
flask
```

```
flask-restx
```

```
docker pull python:3.9.23
```

```
#Dockerfile
```

```
FROM python:3.9.23
```

```
COPY ./*.py /
```

```
COPY ./requirements.txt /
```

```
RUN pip install -r requirements.txt
```

```
EXPOSE 80
```

```
CMD ["python3.9","calculator.py"]
```

```
docker build -t calc .
```

```
docker run -d -p 80:8080 --name mycalculator calc
```

```
192.168.56.111
```