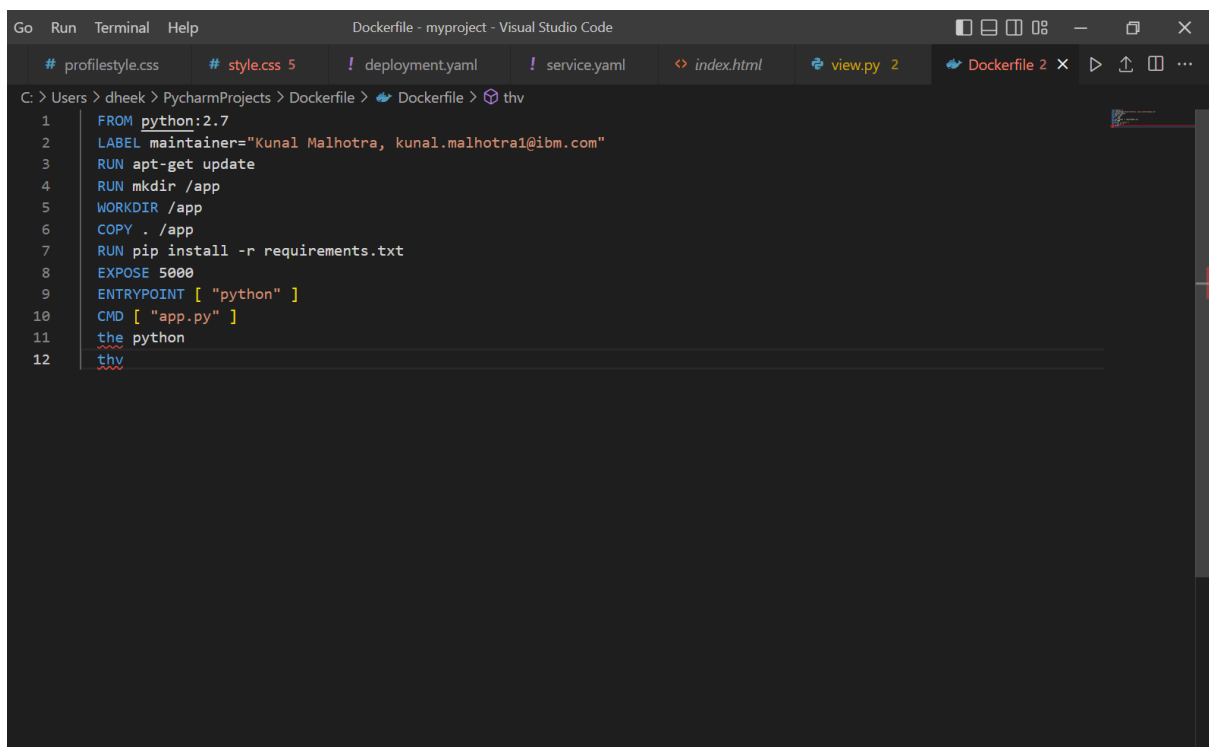


CONTAINERIZE THE APP

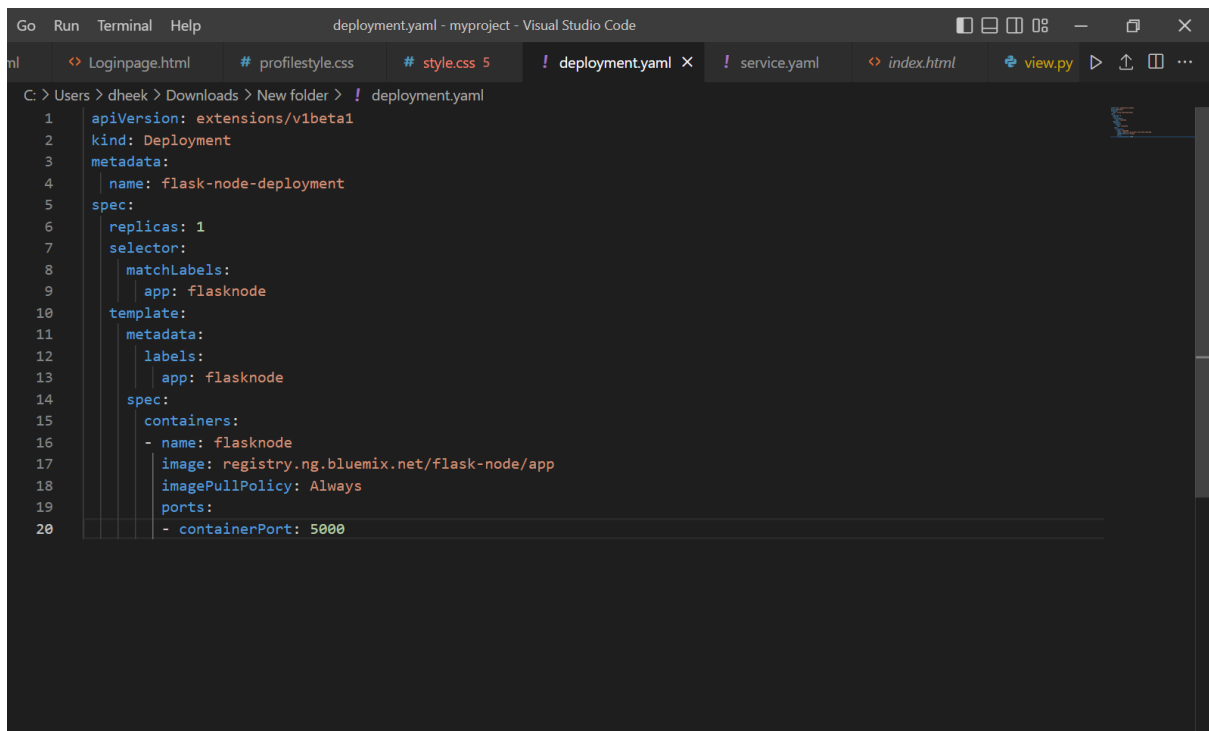
| | |
|--------------|---------------------------------|
| Date | 18 November |
| Team id | PNT2022TMID29412 |
| Project name | Nutrition assistant application |

Docker file:

A screenshot of a Visual Studio Code editor window showing a Dockerfile. The window title is "Dockerfile - myproject - Visual Studio Code". The file explorer on the left shows a project structure with files like profilestyle.css, style.css, deployment.yaml, service.yaml, index.html, view.py, and Dockerfile. The Dockerfile content is as follows:

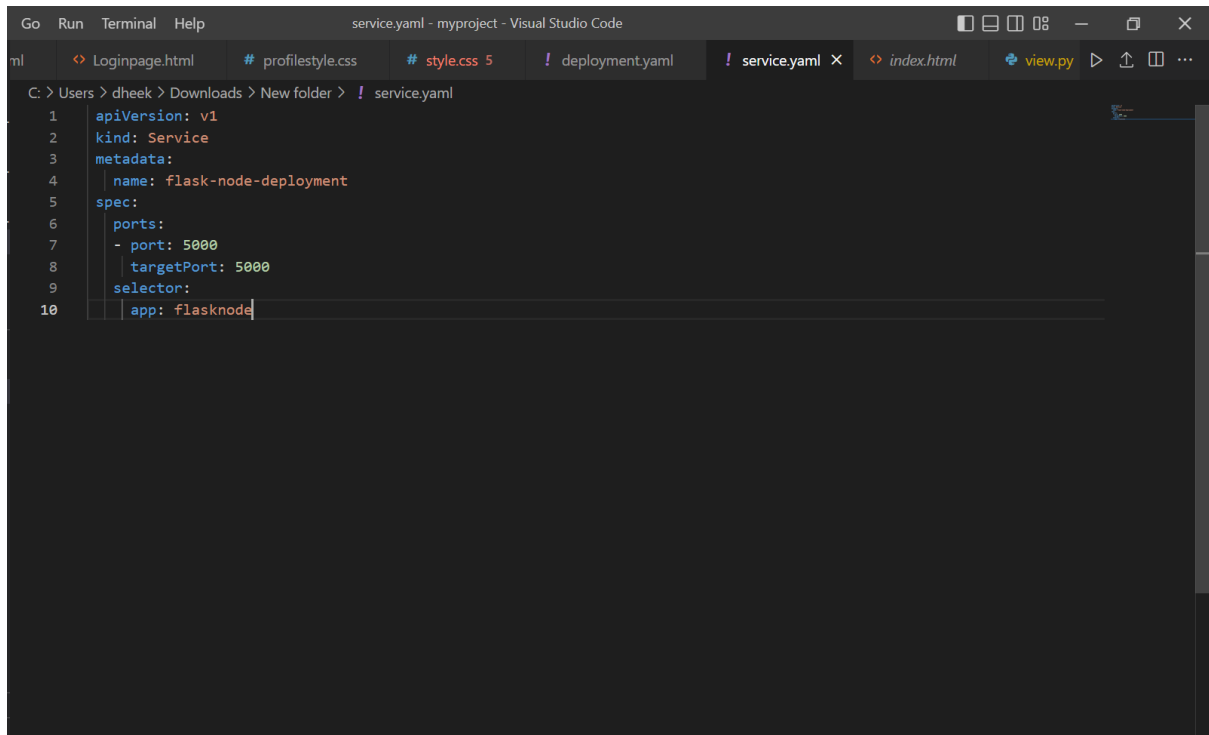
```
1 FROM python:2.7
2 LABEL maintainer="Kunal Malhotra, kunal.malhotra1@ibm.com"
3 RUN apt-get update
4 RUN mkdir /app
5 WORKDIR /app
6 COPY . /app
7 RUN pip install -r requirements.txt
8 EXPOSE 5000
9 ENTRYPOINT [ "python" ]
10 CMD [ "app.py" ]
11 the python
12 thv
```

Deployment.yaml:



```
1  apiVersion: extensions/v1beta1
2  kind: Deployment
3  metadata:
4    name: flask-node-deployment
5  spec:
6    replicas: 1
7    selector:
8      matchLabels:
9        app: flasknode
10   template:
11     metadata:
12       labels:
13         app: flasknode
14     spec:
15       containers:
16       - name: flasknode
17         image: registry.ng.bluemix.net/flask-node/app
18         imagePullPolicy: Always
19         ports:
20         - containerPort: 5000
```

Service.yaml:

A screenshot of the Visual Studio Code editor interface. The title bar at the top reads "service.yaml - myproject - Visual Studio Code". The editor has a dark theme. The file explorer on the left shows a project structure with files like "Loginpage.html", "profilestyle.css", "style.css", "deployment.yaml", "service.yaml", "index.html", and "view.py". The "service.yaml" file is open in the main editor area. The code is a Kubernetes Service manifest. The terminal at the bottom shows the command "C: > Users > dheek > Downloads > New folder > ! service.yaml".

```
1  apiVersion: v1
2  kind: Service
3  metadata:
4    name: flask-node-deployment
5  spec:
6    ports:
7    - port: 5000
8      targetPort: 5000
9    selector:
10   app: flasknode
```