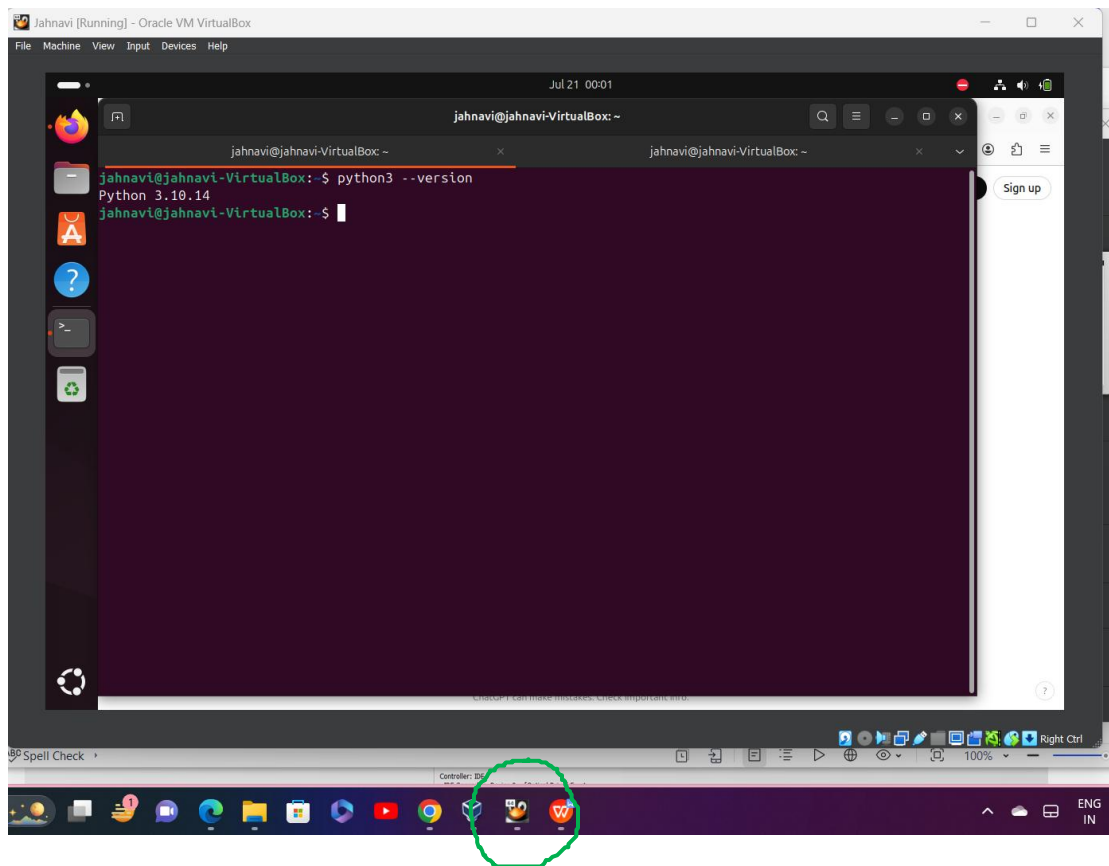
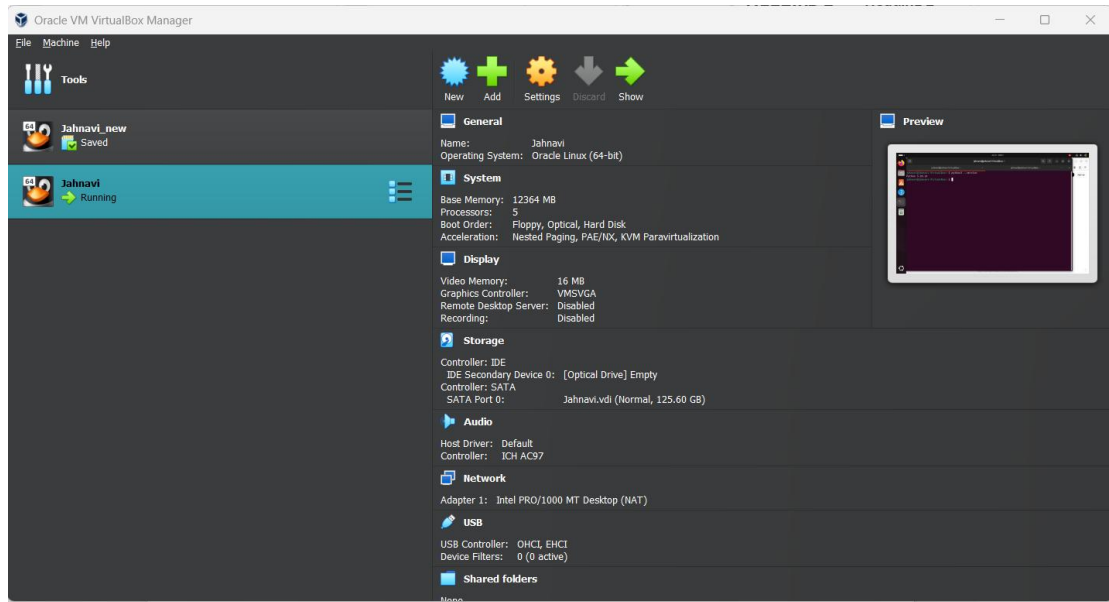


## WEEK10 -WEEK 12 ASSIGNMENT

1. Host a Ubuntu Virtual Machine using Oracle VM Virtual Box

Solution : Hosted Ubuntu Virtual machine using oracle VM Virtual box

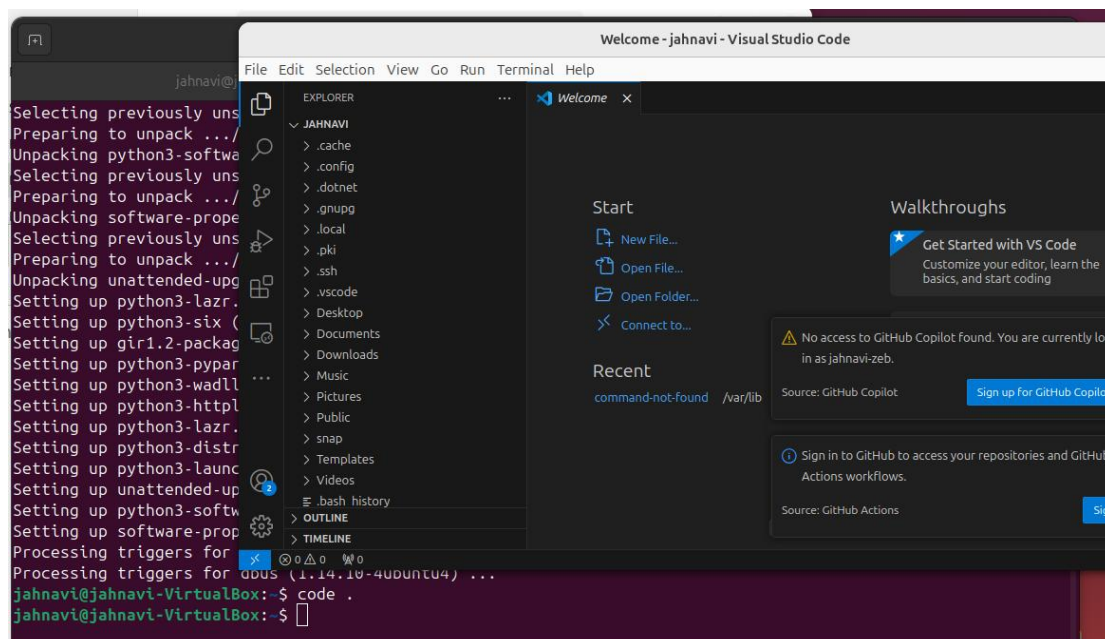


```
jahnavi@jahnavi-VirtualBox: ~  
jahnavi@jahnavi-VirtualBox: ~  
jahnavi@jahnavi-VirtualBox: ~  
jahnavi@jahnavi-VirtualBox:~$ sudo apt install software-properties-common  
[sudo] password for jahnavi:  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following additional packages will be installed:  
  gir1.2-packagekitglib-1.0 python3-distro-info python3-httplib2 python3-launchpadlib  
  python3-lazr.restfulclient python3-lazr.uri python3-pyparsing python3-six python3-software-properties  
  python3-wadllib unattended-upgrades  
Suggested packages:  
  python3-keyring python3-testresources python-pyparsing-doc bsd-mailx default-mta | mail-transport-agent  
  needrestart  
The following NEW packages will be installed:  
  gir1.2-packagekitglib-1.0 python3-distro-info python3-httplib2 python3-launchpadlib  
  python3-lazr.restfulclient python3-lazr.uri python3-pyparsing python3-six python3-software-properties  
  python3-wadllib software-properties-common unattended-upgrades  
0 upgraded, 12 newly installed, 0 to remove and 0 not upgraded.  
Need to get 484 kB of archives.  
After this operation, 3,924 kB of additional disk space will be used.  
Do you want to continue? [Y/n] y  
Get:1 http://in.archive.ubuntu.com/ubuntu noble/main amd64 python3-distro-info all 1.7build1 [7,076 B]  
Get:2 http://in.archive.ubuntu.com/ubuntu noble/main amd64 gir1.2-packagekitglib-1.0 amd64 1.2.8-2build3 [25.6  
kB]  
Get:3 http://in.archive.ubuntu.com/ubuntu noble/main amd64 python3-pyparsing all 3.1.1-1 [86.2 kB]  
Get:4 http://in.archive.ubuntu.com/ubuntu noble/main amd64 python3-httplib2 all 0.20.4-3 [30.4 kB]  
Get:5 http://in.archive.ubuntu.com/ubuntu noble/main amd64 python3-lazr.uri all 1.0.6-3 [13.5 kB]
```

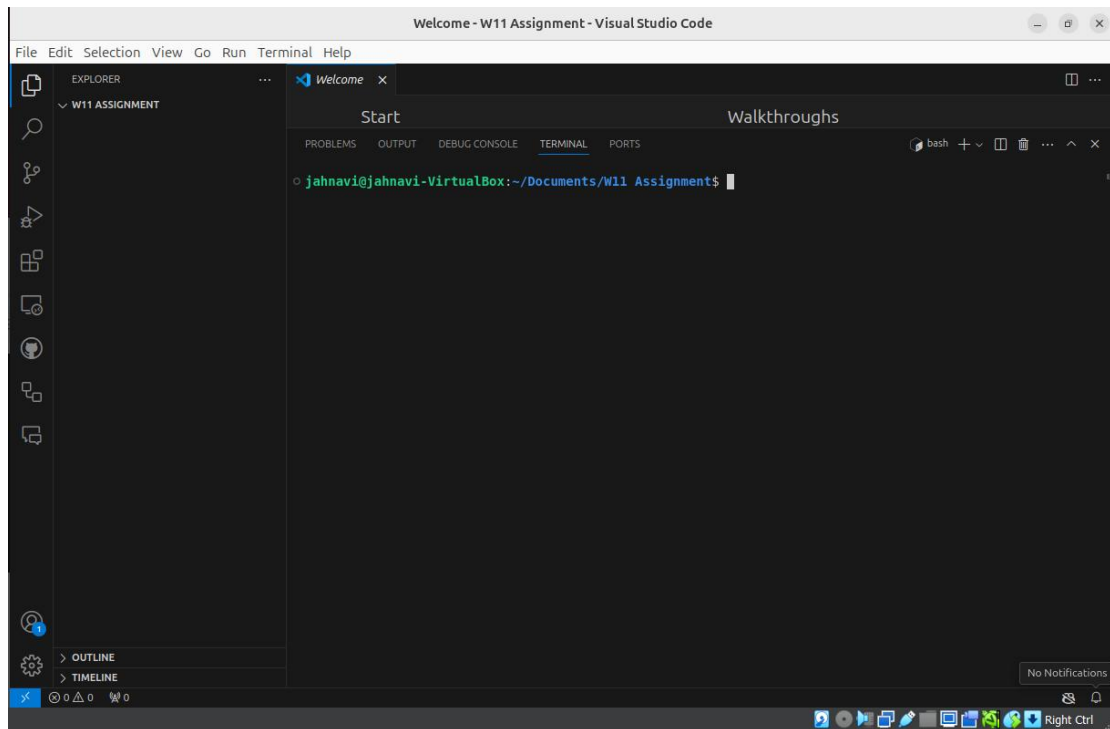
2. Set up Visual Studio code on Ubuntu VM.

Downloaded the VSCode file from <https://code.visualstudio.com/>

Used Terminal to install it.

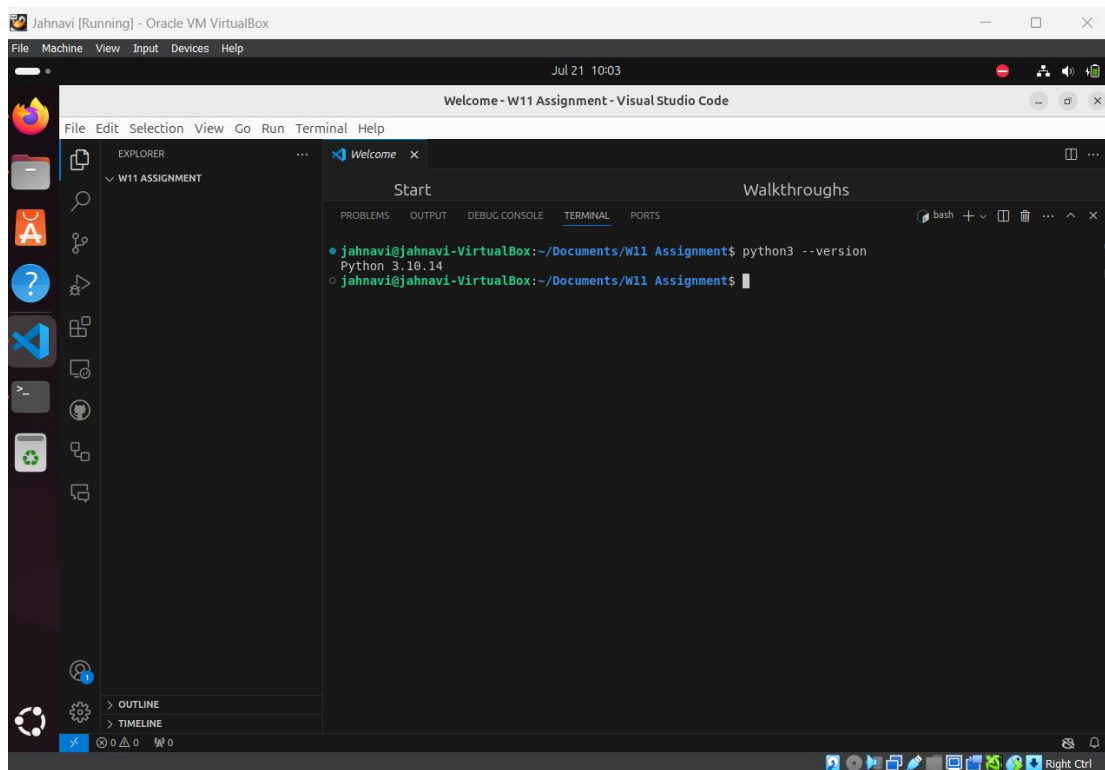


Created a document with Assignment name for this project.



### 3. Set up Python

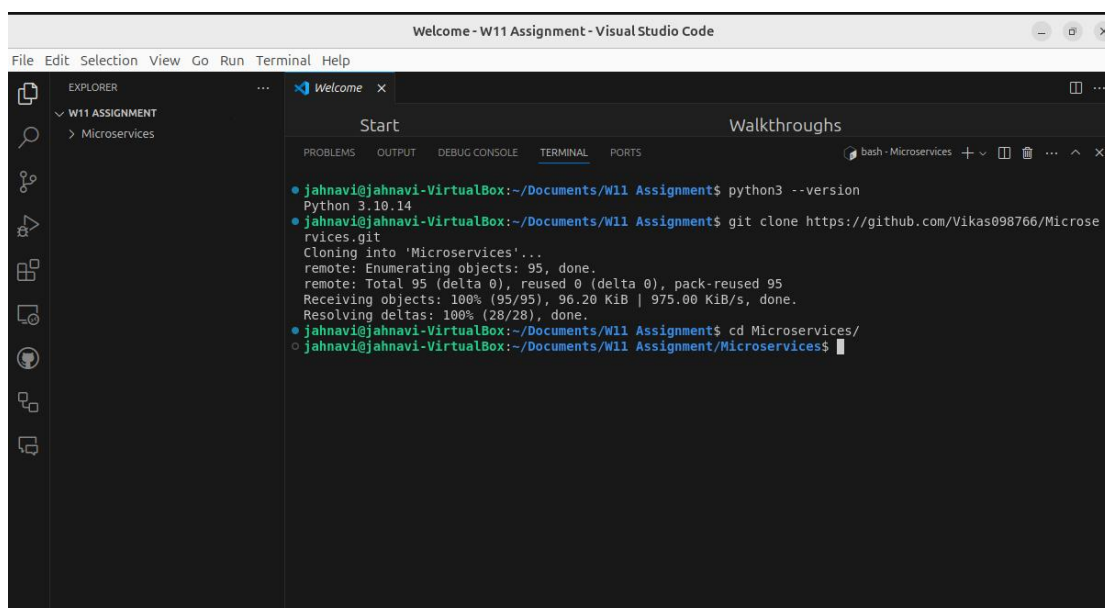
Solution : Python is set up, please find below snap with python version



4. Clone this Github repository  
<https://github.com/Vikas098766/Microservices.git>

Solution : Cloned using the command

● `git clone https://github.com/Vikas098766/Microservices.git`

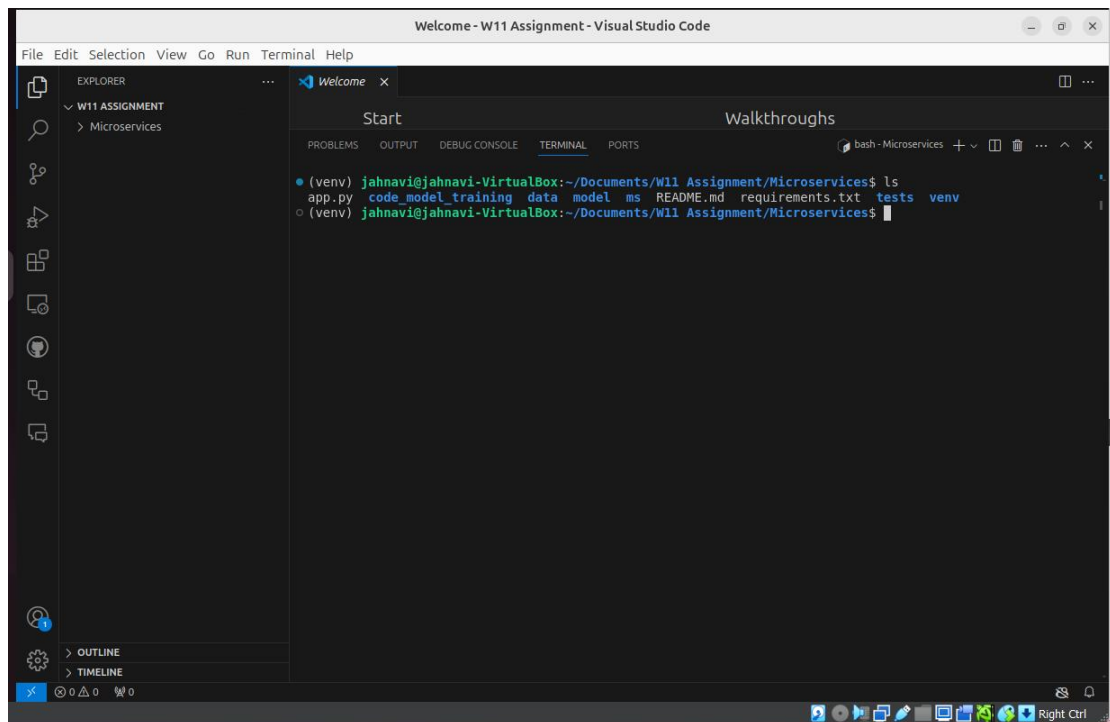


5. Create a Virtual Environment.

Solution : Created Virtual Environment using commands

- `python3 -m venv venv`
- `source venv/bin/activate`

```
jahnavi@jahnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$ python3 -m venv venv
jahnavi@jahnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$ spurge venv/bin/activate
bash: spurge: command not found
jahnavi@jahnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$ source venv/bin/activate
(jenv) jahnavi@jahnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$
```



6. Install the dependencies from requirements.txt file.

Solution : Installed all dependencies present in requirements.txt file using the command

- `Pip install -r requirements.txt`



Visual Studio Code interface showing the terminal output of a pip install command. The terminal is titled 'Welcome - W11 Assignment - Visual Studio Code'. The left sidebar shows the Explorer with 'W11 ASSIGNMENT' and 'Microservices' folders. The main editor area shows the 'Start' tab with the following output:

```
(venv) jahnnavi@jahnnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$ ls
app.py  code_model_training  data  model.ms  README.md  requirements.txt  tests  venv
(venv) jahnnavi@jahnnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$ pip install -r requirements.txt
Collecting click==8.0.3
  Downloading click-8.0.3-py3-none-any.whl (97 kB)
    97.5/97.5 kB 2.7 MB/s eta 0:00:00
Collecting cyclr==0.11.0
  Downloading cyclr-0.11.0-py3-none-any.whl (6.4 kB)
Collecting Flask==2.0.2
  Downloading Flask-2.0.2-py3-none-any.whl (95 kB)
    95.2/95.2 kB 2.0 MB/s eta 0:00:00
Collecting fonttools==4.28.5
  Downloading fonttools-4.28.5-py3-none-any.whl (890 kB)
    890.4/890.4 kB 4.5 MB/s eta 0:00:00
Collecting gunicorn==20.1.0
  Downloading gunicorn-20.1.0-py3-none-any.whl (79 kB)
    79.5/79.5 kB 4.4 MB/s eta 0:00:00
Collecting itsdangerous==2.0.1
  Downloading itsdangerous-2.0.1-py3-none-any.whl (18 kB)
Collecting Jinja2==3.0.3
  Downloading Jinja2-3.0.3-py3-none-any.whl (133 kB)
    133.6/133.6 kB 2.9 MB/s eta 0:00:00
Collecting joblib==1.1.0
  Downloading joblib-1.1.0-py2.py3-none-any.whl (306 kB)
    307.0/307.0 kB 3.6 MB/s eta 0:00:00
Collecting kiwisolver==1.3.2
  Downloading kiwisolver-1.3.2-cp310-cp310-manylinux_2_12_x86_64_manylinux2010_x86_64.whl (1.6 MB)
    1.6/1.6 MB 2.6 MB/s eta 0:00:00
Collecting MarkupSafe==2.0.1
  Downloading MarkupSafe-2.0.1-cp310-cp310-manylinux_2_5_x86_64_manylinux1_x86_64_manylinux_2_12_x86_64
  .manylinux2010_x86_64.whl (30 kB)
Collecting matplotlib==3.5.1
```

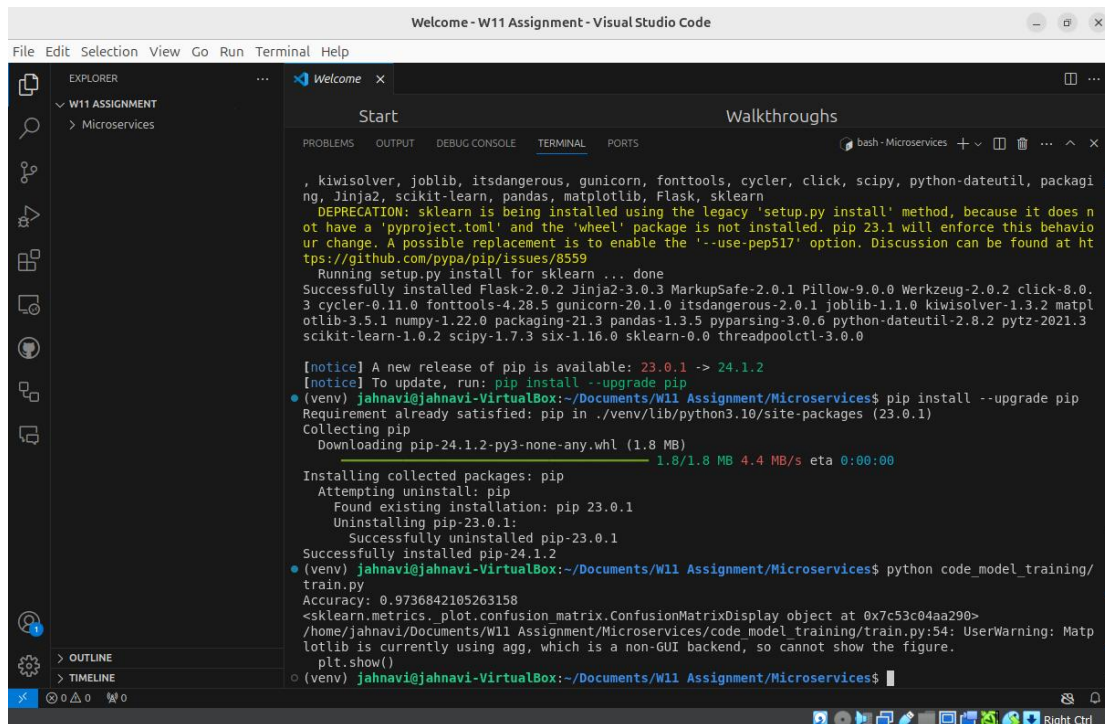
Visual Studio Code interface showing the terminal output of a pip install command. The terminal is titled 'Welcome - W11 Assignment - Visual Studio Code'. The left sidebar shows the Explorer with 'W11 ASSIGNMENT' and 'Microservices' folders. The main editor area shows the 'Start' tab with the following output:

```
Download scikit_learn-1.0.2-cp310-cp310-manylinux_2_17_x86_64_manylinux2014_x86_64.whl (26.5 MB)
    26.5/26.5 MB 3.7 MB/s eta 0:00:00
Collecting scipy==1.7.3
  Downloading scipy-1.7.3-cp310-cp310-manylinux_2_17_x86_64_manylinux2014_x86_64.whl (39.9 MB)
    39.9/39.9 MB 4.2 MB/s eta 0:00:00
Collecting six==1.16.0
  Downloading six-1.16.0-py2.py3-none-any.whl (11 kB)
Collecting sklearn==0.0
  Downloading sklearn-0.0.tar.gz (1.1 kB)
  Preparing metadata (setup.py) ... done
Collecting threadpoolctl==3.0.0
  Downloading threadpoolctl-3.0.0-py3-none-any.whl (14 kB)
Collecting Werkzeug==2.0.2
  Downloading Werkzeug-2.0.2-py3-none-any.whl (288 kB)
    288.9/288.9 kB 4.5 MB/s eta 0:00:00
Requirement already satisfied: setuptools==3.0 in ./venv/lib/python3.10/site-packages (from gunicorn==20.1.0->-r requirements.txt (line 5)) (65.5.0)
Installing collected packages: pytz, Werkzeug, threadpoolctl, six, pyparsing, Pillow, numpy, MarkupSafe, kiwisolver, joblib, itsdangerous, gunicorn, fonttools, cyclr, click, scipy, python-dateutil, packaging, Jinja2, scikit-learn, pandas, matplotlib, Flask, sklearn
  DEPRECATION: sklearn is being installed using the legacy 'setup.py install' method, because it does not have a 'pyproject.toml' and the 'wheel' package is not installed. pip 23.1 will enforce this behavior change. A possible replacement is to enable the '--use-pep517' option. Discussion can be found at https://github.com/pypa/pip/issues/8559
  Running setup.py install for sklearn ... done
Successfully installed Flask-2.0.2 Jinja2-3.0.3 MarkupSafe-2.0.1 Pillow-9.0.0 Werkzeug-2.0.2 click-8.0.3 cyclr-0.11.0 fonttools-4.28.5 gunicorn-20.1.0 itsdangerous-2.0.1 joblib-1.1.0 kiwisolver-1.3.2 matplotlib-3.5.1 numpy-1.22.0 packaging-21.3 pandas-1.3.5 pyparsing-3.0.6 python-dateutil-2.8.2 pytz-2021.3 scikit-learn-1.0.2 scipy-1.7.3 six-1.16.0 sklearn-0.0 threadpoolctl-3.0.0

[notice] A new release of pip is available: 23.0.1 -> 24.1.2
[notice] To update, run: pip install --upgrade pip
(venv) jahnnavi@jahnnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$
```

7. Train and save the model.

Solution : Trained and saved the model. ● Command: python code\_model\_training/train.py



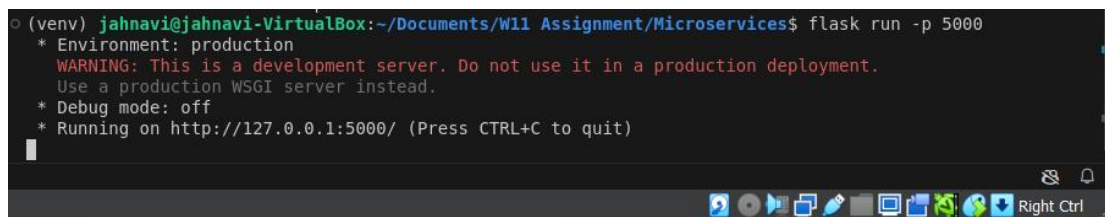
```
Welcome - W11 Assignment - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
W11 ASSIGNMENT
Microservices
Start Walkthroughs
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
bash - Microservices
, kiwisolver, joblib, itsdangerous, gunicorn, fonttools, cyclr, click, scipy, python-dateutil, packagi
ng, Jinja2, scikit-learn, pandas, matplotlib, Flask, sklearn
DEPRECATION: sklearn is being installed using the legacy 'setup.py install' method, because it does n
ot have a 'pyproject.toml' and the 'wheel' package is not installed. pip 23.1 will enforce this behavio
ur change. A possible replacement is to enable the '--use-pep517' option. Discussion can be found at ht
tps://github.com/pypa/pip/issues/8559
Running setup.py install for sklearn ... done
Successfully installed Flask-2.0.2 Jinja2-3.0.3 MarkupSafe-2.0.1 Pillow-9.0.0 Werkzeug-2.0.2 click-8.0.
3 cyclr-0.11.0 fonttools-4.28.5 gunicorn-20.1.0 itsdangerous-2.0.1 joblib-1.1.0 kiwisolver-1.3.2 matpl
otlib-3.5.1 numpy-1.22.0 packaging-21.3 pandas-1.3.5 pyparsing-3.0.6 python-dateutil-2.8.2 pytz-2021.3
scikit-learn-1.0.2 scipy-1.7.3 six-1.16.0 sklearn-0.0 threadpoolctl-3.0.0

[notice] A new release of pip is available: 23.0.1 -> 24.1.2
[notice] To update, run: pip install --upgrade pip
(jahnavi@jahnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$ pip install --upgrade pip
Requirement already satisfied: pip in ./venv/lib/python3.10/site-packages (23.0.1)
Collecting pip
  Downloading pip-24.1.2-py3-none-any.whl (1.8 MB)
    1.8/1.8 MB 4.4 MB/s eta 0:00:00
Installing collected packages: pip
  Attempting uninstall: pip
    Found existing installation: pip 23.0.1
    Uninstalling pip-23.0.1:
      Successfully uninstalled pip-23.0.1
  Successfully installed pip-24.1.2
(jahnavi@jahnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$ python code_model_training/
train.py
Accuracy: 0.9736842105263158
<sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay object at 0x7c53c04aa290>
/home/jahnavi/Documents/W11 Assignment/Microservices/code_model_training/train.py:54: UserWarning: Matp
lotlib is currently using agg, which is a non-GUI backend, so cannot show the figure.
  plt.show()
(jahnavi@jahnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$
```

8. Test the Flask web application.

Solution : Tested web application by running the command.

● flask run -p 5000




```
(jahnavi@jahnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$ flask run -p 5000
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: off
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

9: Tested the the end point /info Command: curl -X GET

http://localhost:5000/info Command: curl -X GET

<http://localhost:5000/health>



```
jahnavi@jahnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$ curl -X GET http://localhost:5000/info
{"name":"Breast Cancer Wisconsin (Diagnostic)","version":"v1.0.0"}
jahnavi@jahnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$ curl -X GET http://localhost:5000/health
okjahnavi@jahnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$
```

**Command USED:** curl-d '[{"radius\_mean": 17.99, "texture\_mean": 10.38, "perimeter\_mean": 122.8, "area\_mean": 1001.0, "smoothness\_mean": 0.1184, "compactness\_mean": 0.2776, "concavity\_mean": 0.3001, "concave points\_mean": 0.1471, "symmetry\_mean": 0.2419, "fractal\_dimension\_mean": 0.07871, "radius\_se": 1.095, "texture\_se": 0.9053, "perimeter\_se": 8.589, "area\_se": 153.4, "smoothness\_se": 0.006399, "compactness\_se": 0.04904, "concavity\_se": 0.05373, "concave points\_se": 0.01587, "symmetry\_se": 0.03003, "fractal\_dimension\_se": 0.006193, "radius\_worst": 25.38, "texture\_worst": 17.33, "perimeter\_worst": 184.6, "area\_worst": 2019.0, "smoothness\_worst": 0.1622, "compactness\_worst": 0.6656, "concavity\_worst": 0.7119, "concave points\_worst": 0.2654, "symmetry\_worst": 0.4601, "fractal\_dimension\_worst": 0.1189}]]' \-H "Content-Type: application/json" \-X POST http://0.0.0.0:5000/predict

```
jahnavi@jahnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$ curl -d '[{"radius_mean": 17.99, "texture_mean": 10.38, "perimeter_mean": 122.8, "area_mean": 1001.0, "smoothness_mean": 0.1184, "compactness_mean": 0.2776, "concavity_mean": 0.3001, "concave points_mean": 0.1471, "symmetry_mean": 0.2419, "fractal_dimension_mean": 0.07871, "radius_se": 1.095, "texture_se": 0.9053, "perimeter_se": 8.589, "area_se": 153.4, "smoothness_se": 0.006399, "compactness_se": 0.04904, "concavity_se": 0.05373, "concave points_se": 0.01587, "symmetry_se": 0.03003, "fractal_dimension_se": 0.006193, "radius_worst": 25.38, "texture_worst": 17.33, "perimeter_worst": 184.6, "area_worst": 2019.0, "smoothness_worst": 0.1622, "compactness_worst": 0.6656, "concavity_worst": 0.7119, "concave points_worst": 0.2654, "symmetry_worst": 0.4601, "fractal_dimension_worst": 0.1189}]]' \-H "Content-Type: application/json" \-X POST http://localhost:5000/predict
{"label": "M", "prediction": 1, "status": 200}
jahnavi@jahnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$
```

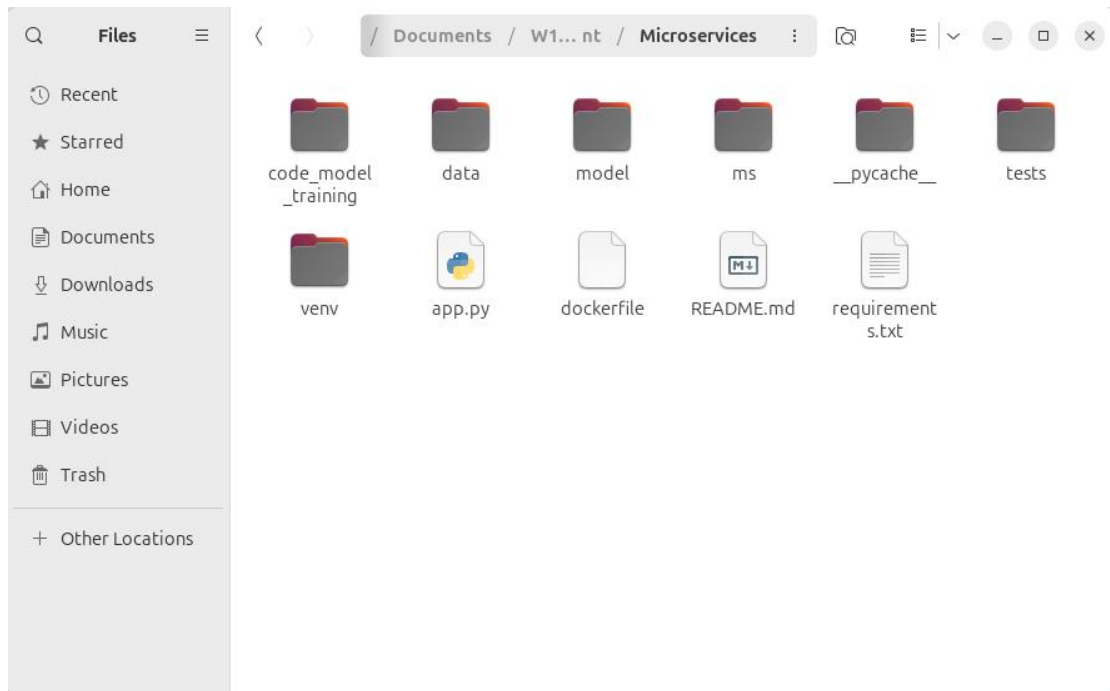
API ENDPOINT with /predict got the output as  
 { "label" : "M" , " prediction" :1, " status" :200}

10.Steps to create a docker image.

1. Created the text file named dockerfile using the command as touch dockerfile

```
jahnavi@jahnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$ touch dockerfile
jahnavi@jahnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$
```





2. Within the txt file adding the following content within it.

```
Open ▾ [icon] • dockerfile
~/Documents/W11 Assignment/Microservices

# Use an official Python runtime as a parent image
FROM python:3.9-slim

# Set the working directory inside the container
WORKDIR /usr/src/app

# Copy the requirements file into the container
COPY requirements.txt ./

# Install dependencies
RUN pip install --no-cache-dir -r requirements.txt

# Copy the rest of the application code into the container
COPY . .

# Expose the port the app runs on
EXPOSE 5000

# Define the command to run the app
CMD ["flask", "run", "--host=0.0.0.0", "--port=5000"]S
```

3. Build the docker image with the name as my-python-app

- Command: `sudo docker build -t my-python-app .`

```
jahnavi@jahnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$ docker --version
Docker version 27.0.3, build 7d4bcd8
jahnavi@jahnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$ sudo docker build -t my-python-app .
[+] Building 8.4s (1/2)                                docker:default
=> [internal] load build definition from dockerfile                                0.1s
=> ==> transferring dockerfile: 540B                                              0.0s
=> [internal] load metadata for docker.io/library/python:3.9-slim                8.3s
```

```
jahnavi@jahnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$ docker --version
Docker version 27.0.3, build 7d4bcd8
jahnavi@jahnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$ sudo docker build -t my-python-app .
[+] Building 68.7s (7/9)                                docker:default
=> ==> transferring context: 2B                                                    0.0s
=> [1/5] FROM docker.io/library/python:3.9-slim@sha256:a6c12ec09f13df9d4b8b4e4d08678c1b212d89885be14b6c7 38.4s
=> ==> resolve docker.io/library/python:3.9-slim@sha256:a6c12ec09f13df9d4b8b4e4d08678c1b212d89885be14b6c7 0.1s
=> ==> sha256:f11c1adaa26e078479ccdd45312ea3b88476441b91be0ec898a7e07bfd05badc 29.13MB / 29.13MB 14.0s
=> ==> sha256:a6c12ec09f13df9d4b8b4e4d08678c1b212d89885be14b6c72b633bee2a520f4 10.41kB / 10.41kB 0.0s
=> ==> sha256:4719115deb9cc7a5479a7d3c57cfceac2be89fcaf0fed8c747e8dfb4b01a79a3 1.94kB / 1.94kB 0.0s
=> ==> sha256:b97320a8c1caf64deebb911ff8eb75bf12f671408a85302dd33b5ede2d1cdc1 6.90kB / 6.90kB 0.0s
=> ==> sha256:c1f67e58a3d2a9d9c5f38c8c3fc629ff3bfd6e0045b935c99e9ffc4182070fa1 3.51MB / 3.51MB 4.1s
=> ==> sha256:9370038d11852cad5a70691e76b0ddc8e669018bc770cad15c23a3def629b874 11.89MB / 11.89MB 10.2s
=> ==> sha256:174cb52e37e55d01d6ea95cdfa00d252ccd883946a96050ff4be1633a7f1712 231B / 231B 4.6s
=> ==> sha256:f259657f3656970a8d44c5a7a8250b746967f869516bb767767c6942eaa60e55 2.77MB / 2.77MB 8.1s
=> ==> extracting sha256:f11c1adaa26e078479ccdd45312ea3b88476441b91be0ec898a7e07bfd05badc 16.0s
=> ==> extracting sha256:c1f67e58a3d2a9d9c5f38c8c3fc629ff3bfd6e0045b935c99e9ffc4182070fa1 1.5s
=> ==> extracting sha256:9370038d11852cad5a70691e76b0ddc8e669018bc770cad15c23a3def629b874 2.8s
=> ==> extracting sha256:174cb52e37e55d01d6ea95cdfa00d252ccd883946a96050ff4be1633a7f1712 0.0s
=> ==> extracting sha256:f259657f3656970a8d44c5a7a8250b746967f869516bb767767c6942eaa60e55 1.8s
=> [internal] load build context                                                    29.6s
=> ==> transferring context: 449.68MB                                              29.3s
=> [2/5] WORKDIR /usr/src/app                                                    2.2s
=> [3/5] COPY requirements.txt ./                                                 0.2s
=> [4/5] RUN pip install --no-cache-dir -r requirements.txt                      19.2s
=> ==> #                                247.7/247.7 kB 5.7 MB/s eta 0:00:00
=> ==> # Collecting pytz==2021.3
=> ==> # Downloading pytz-2021.3-py2.py3-none-any.whl (503 kB)
=> ==> #                                503.5/503.5 kB 6.8 MB/s eta 0:00:00
```

```
=> [internal] load metadata for docker.io/library/python:3.9-slim                8.4s
=> [internal] load .dockerignore                                                    0.1s
=> ==> transferring context: 2B                                                    0.0s
=> [1/5] FROM docker.io/library/python:3.9-slim@sha256:a6c12ec09f13df9d4b8b4e4d08678c1b212d89885be14b6c7 38.4s
=> ==> resolve docker.io/library/python:3.9-slim@sha256:a6c12ec09f13df9d4b8b4e4d08678c1b212d89885be14b6c7 0.1s
=> ==> sha256:f11c1adaa26e078479ccdd45312ea3b88476441b91be0ec898a7e07bfd05badc 29.13MB / 29.13MB 14.0s
=> ==> sha256:a6c12ec09f13df9d4b8b4e4d08678c1b212d89885be14b6c72b633bee2a520f4 10.41kB / 10.41kB 0.0s
=> ==> sha256:4719115deb9cc7a5479a7d3c57cfceac2be89fcaf0fed8c747e8dfb4b01a79a3 1.94kB / 1.94kB 0.0s
=> ==> sha256:b97320a8c1caf64deebb911ff8eb75bf12f671408a85302dd33b5ede2d1cdc1 6.90kB / 6.90kB 0.0s
=> ==> sha256:c1f67e58a3d2a9d9c5f38c8c3fc629ff3bfd6e0045b935c99e9ffc4182070fa1 3.51MB / 3.51MB 4.1s
=> ==> sha256:9370038d11852cad5a70691e76b0ddc8e669018bc770cad15c23a3def629b874 11.89MB / 11.89MB 10.2s
=> ==> sha256:174cb52e37e55d01d6ea95cdfa00d252ccd883946a96050ff4be1633a7f1712 231B / 231B 4.6s
=> ==> sha256:f259657f3656970a8d44c5a7a8250b746967f869516bb767767c6942eaa60e55 2.77MB / 2.77MB 8.1s
=> ==> extracting sha256:f11c1adaa26e078479ccdd45312ea3b88476441b91be0ec898a7e07bfd05badc 16.0s
=> ==> extracting sha256:c1f67e58a3d2a9d9c5f38c8c3fc629ff3bfd6e0045b935c99e9ffc4182070fa1 1.5s
=> ==> extracting sha256:9370038d11852cad5a70691e76b0ddc8e669018bc770cad15c23a3def629b874 2.8s
=> ==> extracting sha256:174cb52e37e55d01d6ea95cdfa00d252ccd883946a96050ff4be1633a7f1712 0.0s
=> ==> extracting sha256:f259657f3656970a8d44c5a7a8250b746967f869516bb767767c6942eaa60e55 1.8s
=> [internal] load build context                                                    29.6s
=> ==> transferring context: 449.68MB                                              29.3s
=> [2/5] WORKDIR /usr/src/app                                                    2.2s
=> [3/5] COPY requirements.txt ./                                                 0.2s
=> [4/5] RUN pip install --no-cache-dir -r requirements.txt                      59.7s
=> [5/5] COPY . .                                                                  6.7s
=> ==> exporting to image
=> ==> exporting layers                                                            5.2s
=> ==> writing image sha256:e22ffde6c082935ebee56dbc3c77c0e55540620d476df69a4e3b51ebbc3a0ed7 5.1s
=> ==> naming to docker.io/library/my-python-app                                0.0s
jahnavi@jahnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$
```

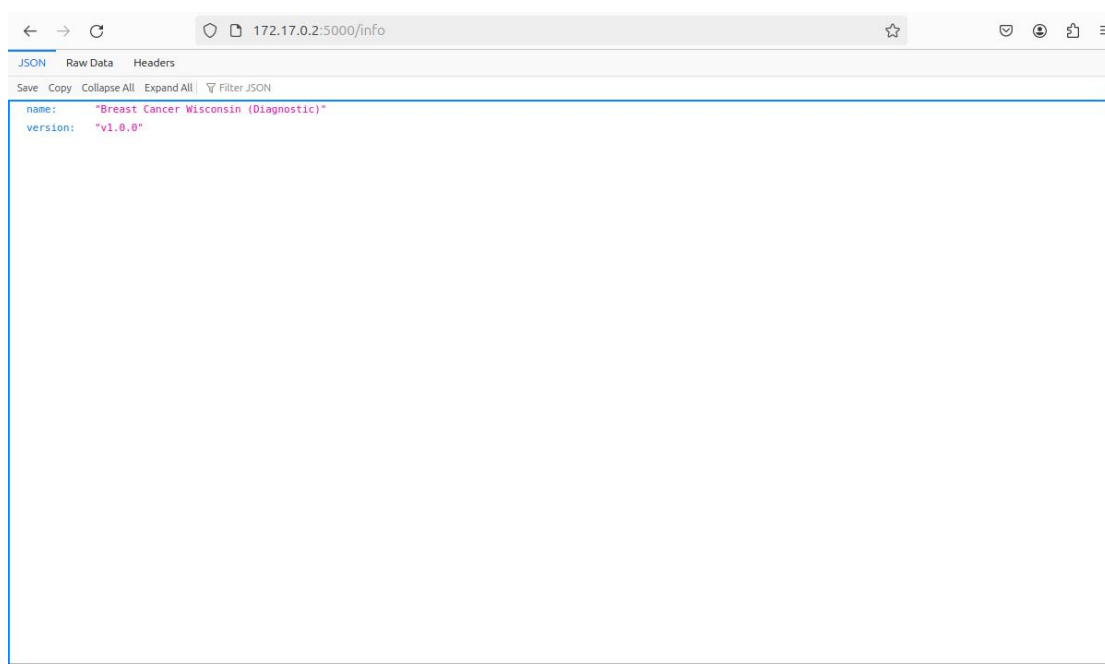
#### 4. Run the Docker Container

● Command: `sudo docker run -p 5000:5000 my-python-app`

```
jahnnavi@jahnnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$ sudo lsof -i :5000
jahnnavi@jahnnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$ sudo docker run -p 5000:5000 my-python-app
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: off
* Running on all addresses.
  WARNING: This is a development server. Do not use it in a production deployment.
* Running on http://172.17.0.2:5000/ (Press CTRL+C to quit)
```

11. To check the Docker image service locally with the help of POSTMAN end points as

1. `/info`



2. `/health`



### 3. /predict

```
jahnavi@jahnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$ curl -d '[{"radius_mean": 17.99, "texture_mean": 10.38, "perimeter_mean": 122.8, "area_mean": 1001.0, "smoothness_mean": 0.1184, "compactness_mean": 0.2776, "concavity_mean": 0.3001, "concave points_mean": 0.1471, "symmetry_mean": 0.2419, "fractal_dimension_mean": 0.07871, "radius_se": 1.095, "texture_se": 0.9053, "perimeter_se": 8.589, "area_se": 153.4, "smoothness_se": 0.006399, "compactness_se": 0.04904, "concavity_se": 0.05373, "concave points_se": 0.01587, "symmetry_se": 0.03003, "fractal_dimension_se": 0.006193, "radius_worst": 25.38, "texture_worst": 17.33, "perimeter_worst": 184.6, "area_worst": 2019.0, "smoothness_worst": 0.1622, "compactness_worst": 0.6656, "concavity_worst": 0.7119, "concave points_worst": 0.2654, "symmetry_worst": 0.4601, "fractal_dimension_worst": 0.1189}]" \
-H "Content-Type: application/json" \
-X POST http://localhost:5000/predict
{"label": "M", "prediction": 1, "status": 200}
jahnavi@jahnavi-VirtualBox:~/Documents/W11 Assignment/Microservices$
```

