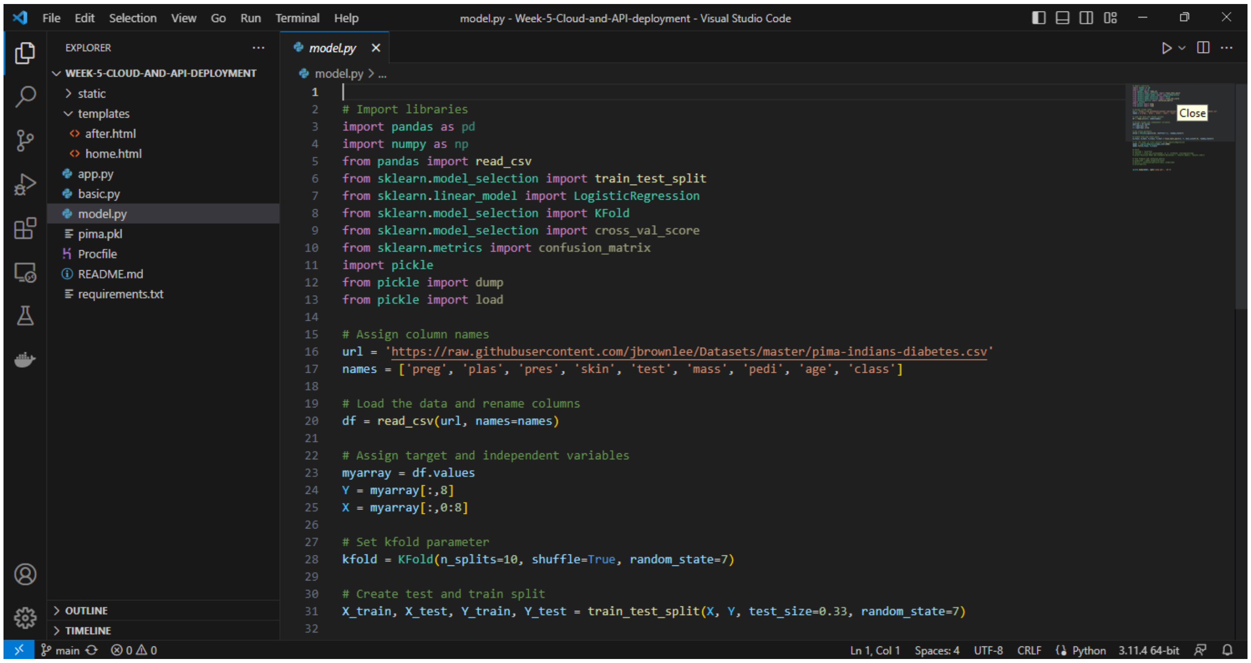
Name: Alexis Collier

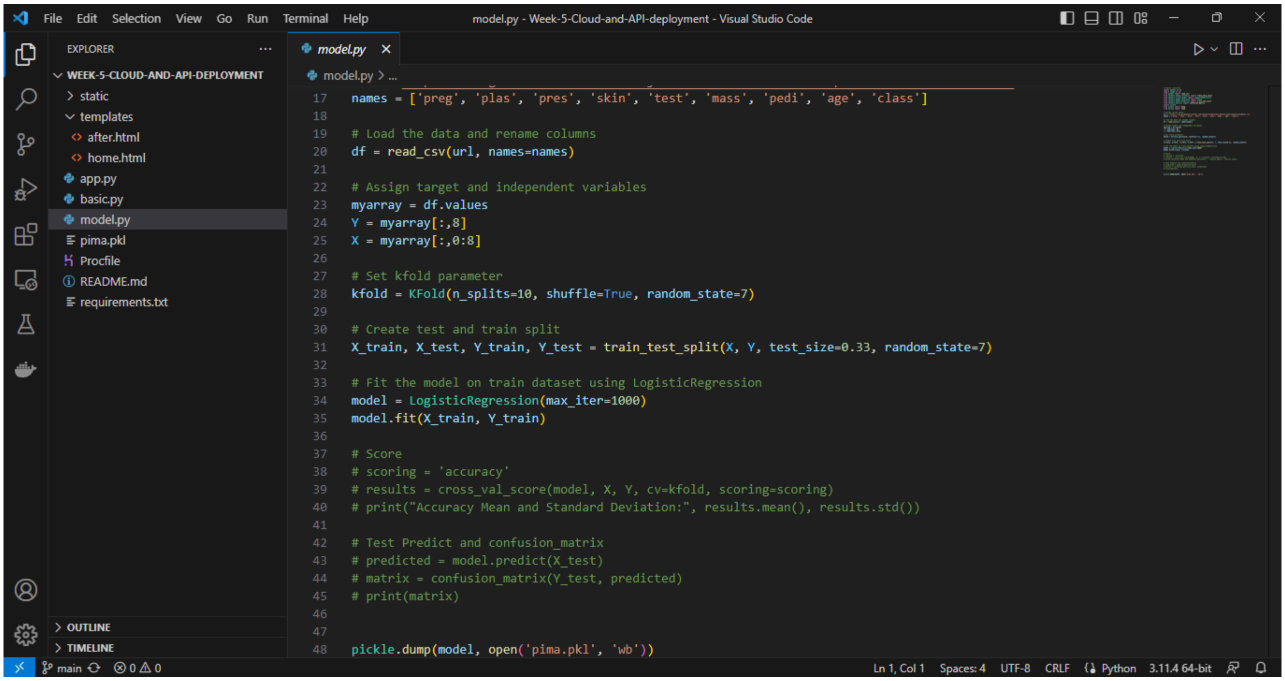
Batch Code: LISUM24

Submission Date: 02 September 2023

- Created the environment
- Create Model
- Import libraries
- Download the dataset
- Identify X and Y (target variable)
- Set kfold parameter
- Create train and test data
- Fit the model (Logistic Regression)
- Dump the model as a pickle file

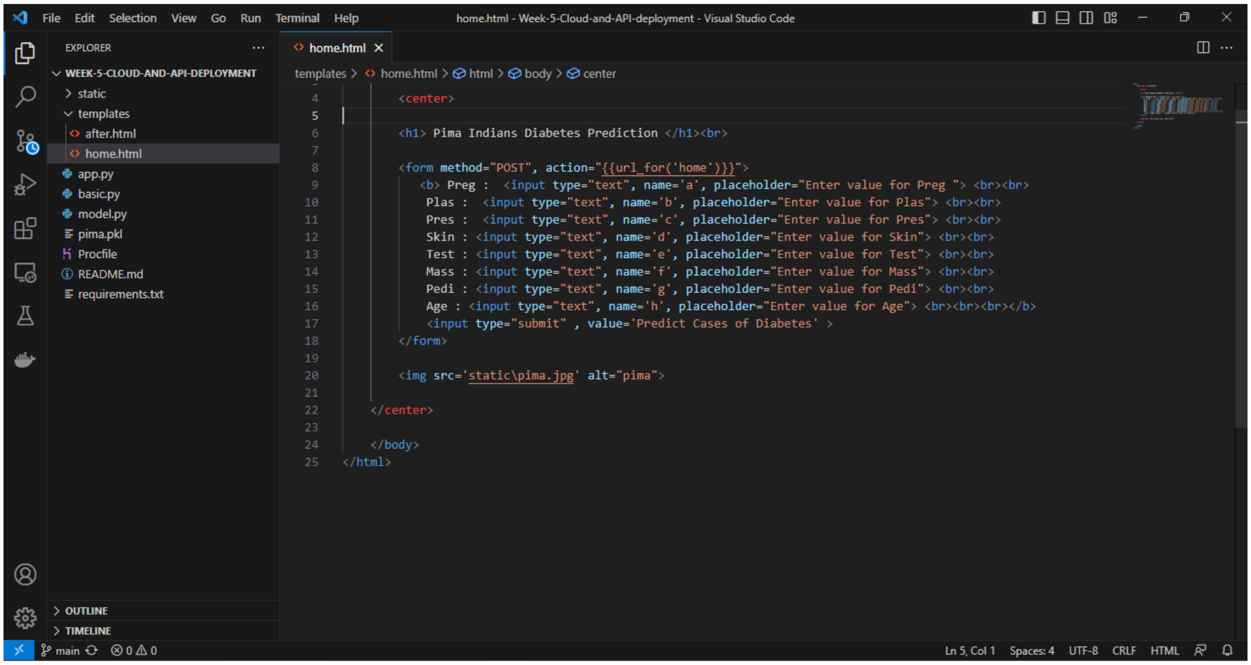


```

# Import libraries
import pandas as pd
import numpy as np
from pandas import read_csv
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import KFold
from sklearn.model_selection import cross_val_score
from sklearn.metrics import confusion_matrix
import pickle
from pickle import dump
from pickle import load

# Assign column names
url = 'https://raw.githubusercontent.com/jbrownlee/Datasets/master/pima-indians-diabetes.csv'
names = ['preg', 'plas', 'pres', 'skin', 'test', 'mass', 'pedi', 'age', 'class']

# Load the data and rename columns
df = read_csv(url, names=names)

# Assign target and independent variables
myarray = df.values
Y = myarray[:,8]
X = myarray[:,0:8]

# Set kfold parameter
kfold = KFold(n_splits=10, shuffle=True, random_state=7)

# Create test and train split
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.33, random_state=7)

```



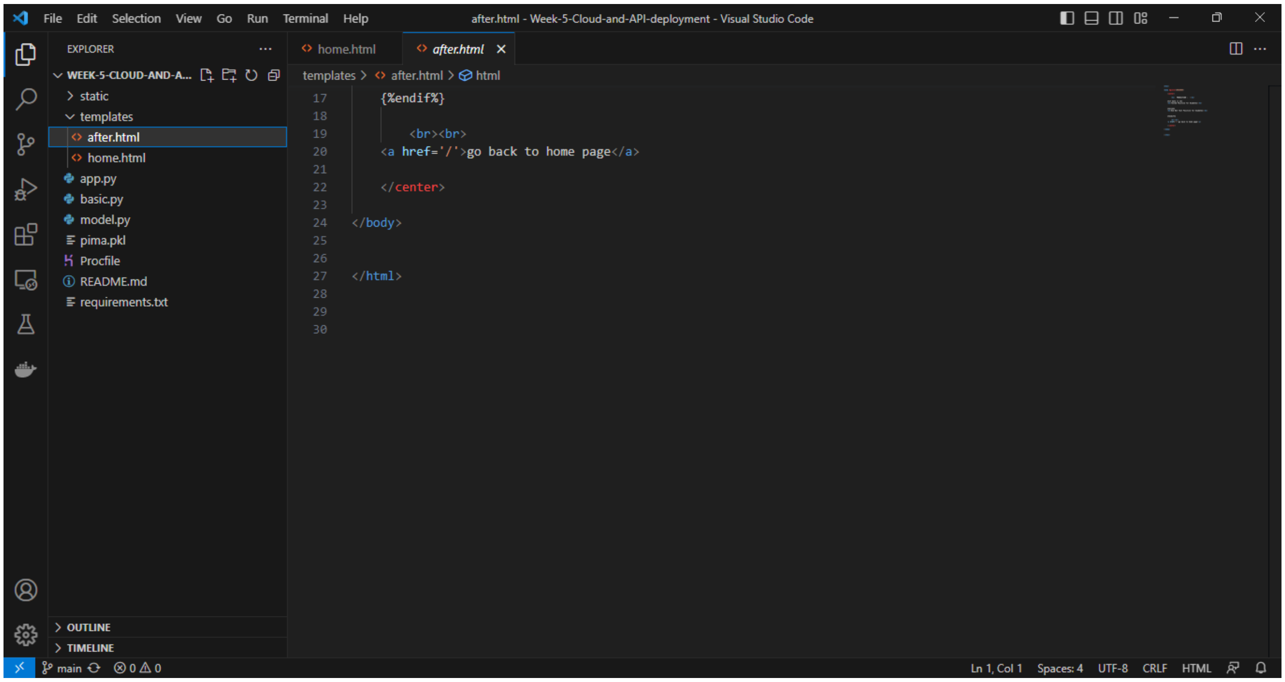
```python
names = ['preg', 'plas', 'pres', 'skin', 'test', 'mass', 'pedi', 'age', 'class']

# Load the data and rename columns
df = read_csv(url, names=names)

# Assign target and independent variables
myarray = df.values
Y = myarray[:,8]
X = myarray[:,0:8]

# Set kfold parameter
kfold = KFold(n_splits=10, shuffle=True, random_state=7)

# Create test and train split
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.33, random_state=7)

# Fit the model on train dataset using LogisticRegression
model = LogisticRegression(max_iter=1000)
model.fit(X_train, Y_train)

# Score
# scoring = 'accuracy'
# results = cross_val_score(model, X, Y, cv=kfold, scoring=scoring)
# print("Accuracy Mean and Standard Deviation:", results.mean(), results.std())

# Test Predict and confusion_matrix
# predicted = model.predict(X_test)
# matrix = confusion_matrix(Y_test, predicted)
# print(matrix)


pickle.dump(model, open('pima.pkl', 'wb'))
```

Create Home.Html Page



```html
        <center>

        <h1> Pima Indians Diabetes Prediction </h1><br>

        <form method="POST", action="{{url_for('home')}}">
           <b> Preg :  <input type="text", name='a', placeholder="Enter value for Preg "> <br><br>
            Plas :  <input type="text", name='b', placeholder="Enter value for Plas"> <br><br>
            Pres :  <input type="text", name='c', placeholder="Enter value for Pres"> <br><br>
            Skin : <input type="text", name='d', placeholder="Enter value for Skin"> <br><br>
            Test : <input type="text", name='e', placeholder="Enter value for Test"> <br><br>
            Mass : <input type="text", name='f', placeholder="Enter value for Mass"> <br><br>
            Pedi : <input type="text", name='g', placeholder="Enter value for Pedi"> <br><br>
            Age : <input type="text", name='h', placeholder="Enter value for Age"> <br><br><br></b>
            <input type="submit" , value='Predict Cases of Diabetes' >
        </form>

        <img src='static\pima.jpg' alt="pima">

    </center>

    </body>
</html>
```

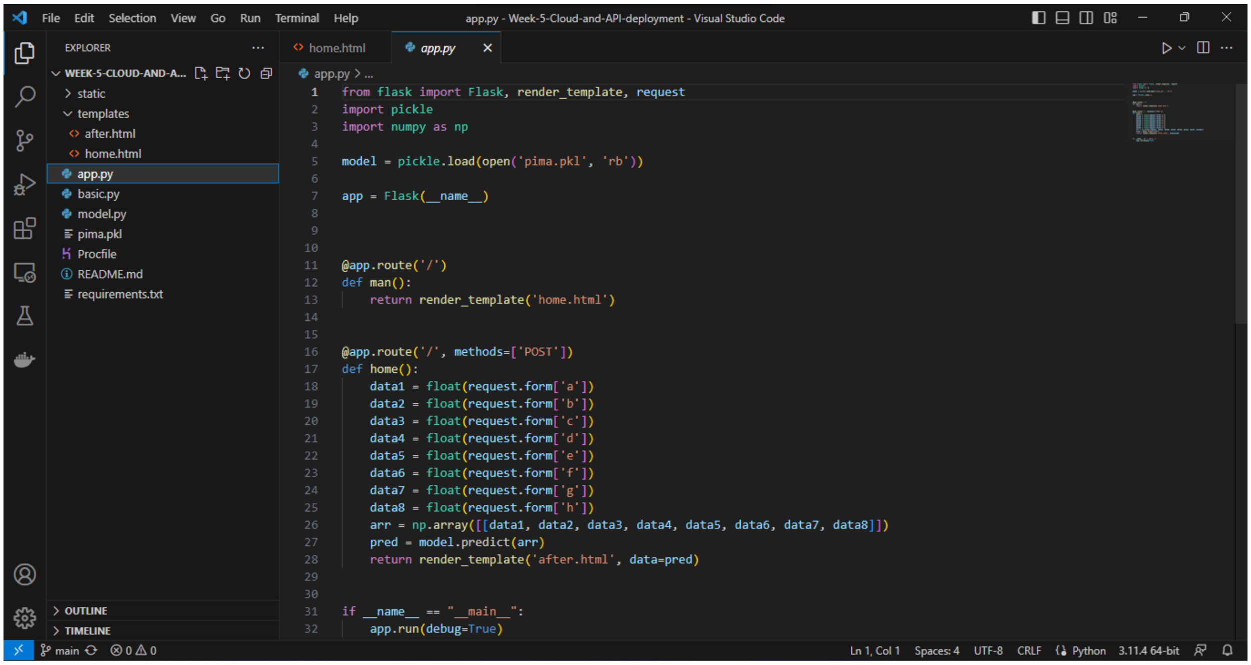
Create After.Html Page



after.html - Week-5-Cloud-and-API-deployment - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
WEEK-5-CLOUD-AND-A...
static
templates
after.html
home.html
app.py
basic.py
model.py
pima.pkl
Procfile
README.md
requirements.txt
OUTLINE
TIMELINE
home.html
after.html
templates > after.html > html
main
Ln 1, Col 1 Spaces: 4 UTF-8 CRLF HTML

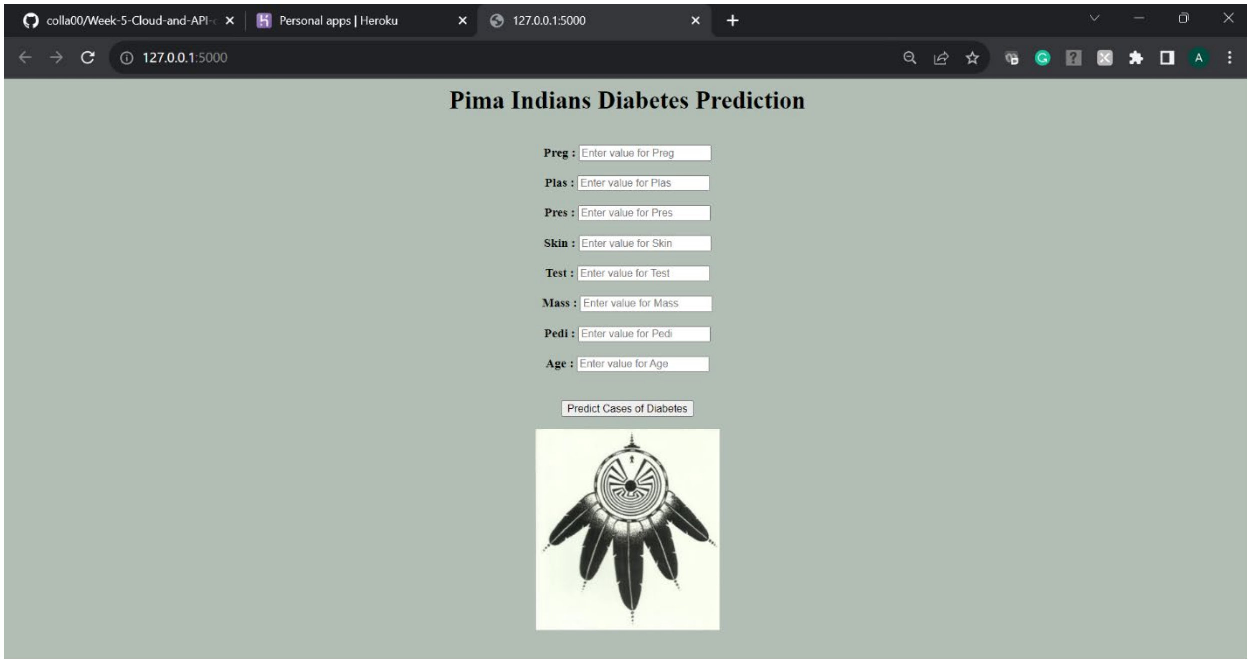
```html
    {%endif%}

        <br><br>
    <a href='/'>go back to home page</a>

    </center>

</body>


</html>



```

Create App.Py



```python
from flask import Flask, render_template, request
import pickle
import numpy as np

model = pickle.load(open('pima.pkl', 'rb'))

app = Flask(__name__)



@app.route('/')
def man():
    return render_template('home.html')


@app.route('/', methods=['POST'])
def home():
    data1 = float(request.form['a'])
    data2 = float(request.form['b'])
    data3 = float(request.form['c'])
    data4 = float(request.form['d'])
    data5 = float(request.form['e'])
    data6 = float(request.form['f'])
    data7 = float(request.form['g'])
    data8 = float(request.form['h'])
    arr = np.array([[data1, data2, data3, data4, data5, data6, data7, data8]])
    pred = model.predict(arr)
    return render_template('after.html', data=pred)


if __name__ == "__main__":
    app.run(debug=True)
```

Test App.Py



Pima Indians Diabetes Prediction

Preg : Enter value for Preg

Plas : Enter value for Plas

Pres : Enter value for Pres

Skin : Enter value for Skin

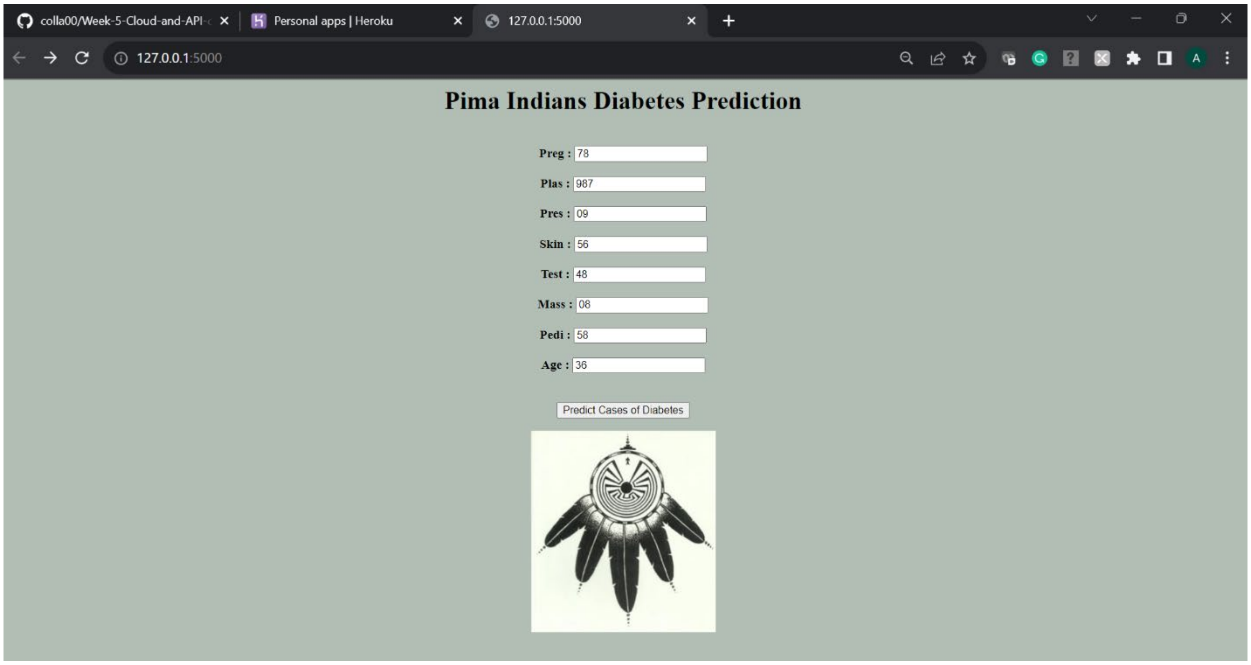
Test : Enter value for Test

Mass : Enter value for Mass

Pedi : Enter value for Pedi

Age : Enter value for Age

Enter Values:



Pima Indians Diabetes Prediction

Preg : 78

Plas : 987

Pres : 09

Skin : 56

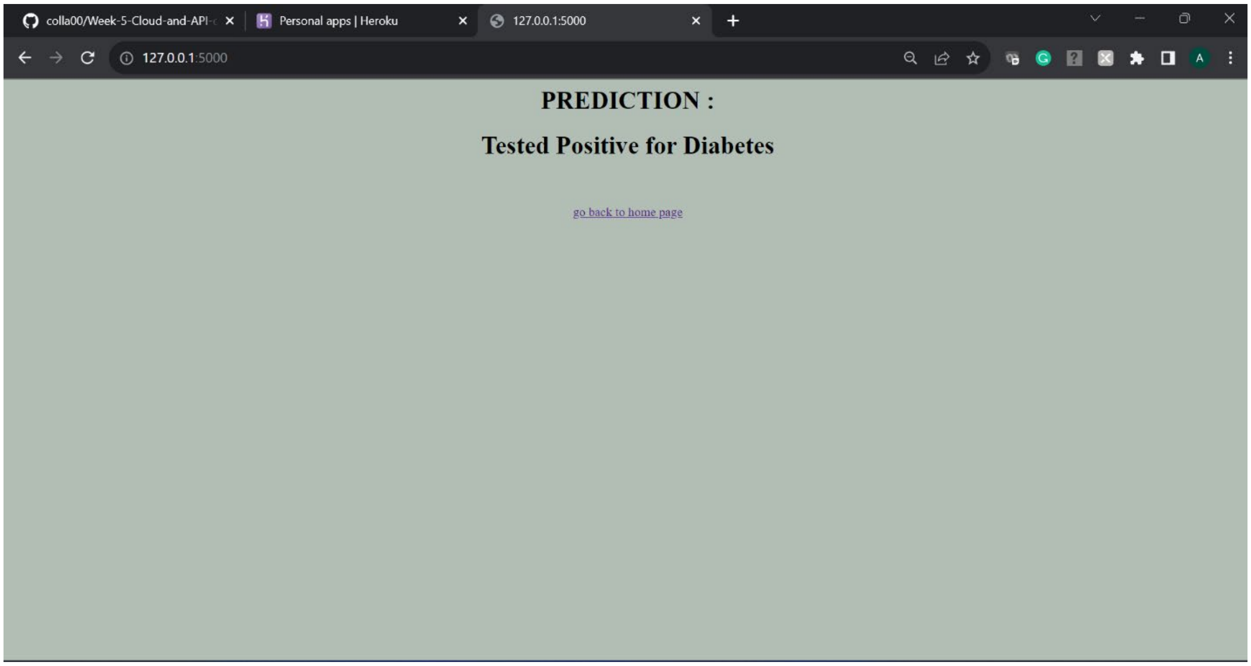
Test : 48

Mass : 08

Pedi : 58

Age : 36

See Prediction Results:

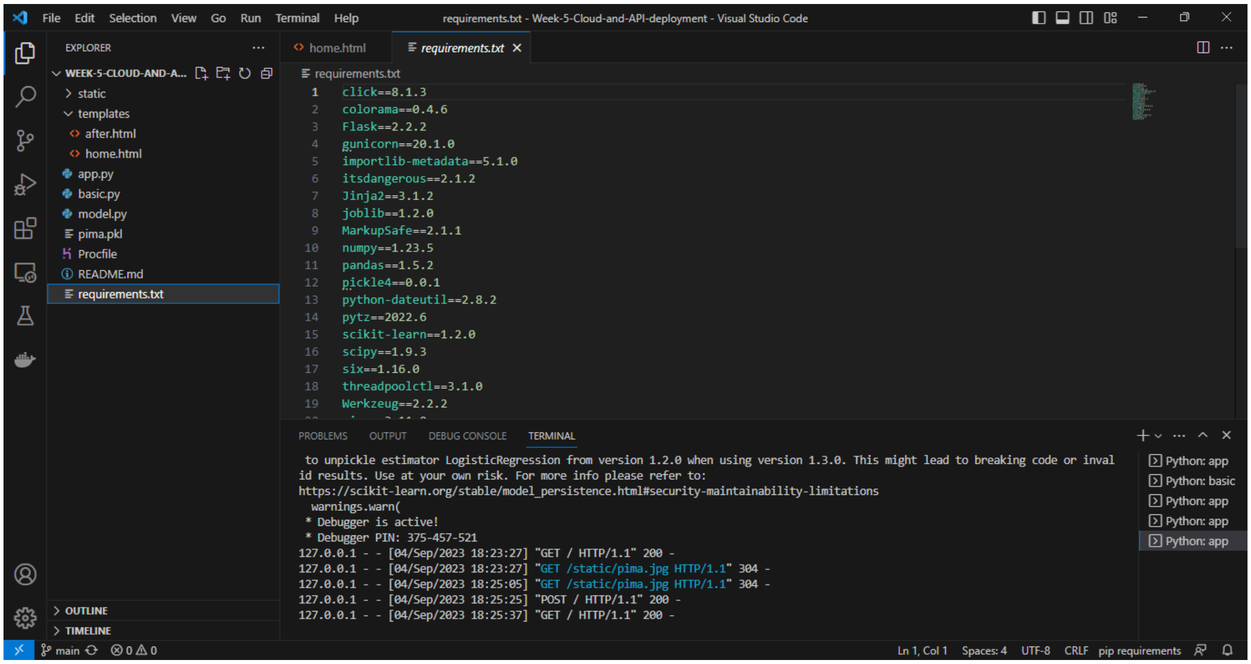


PREDICTION :

Tested Positive for Diabetes

go back to home page

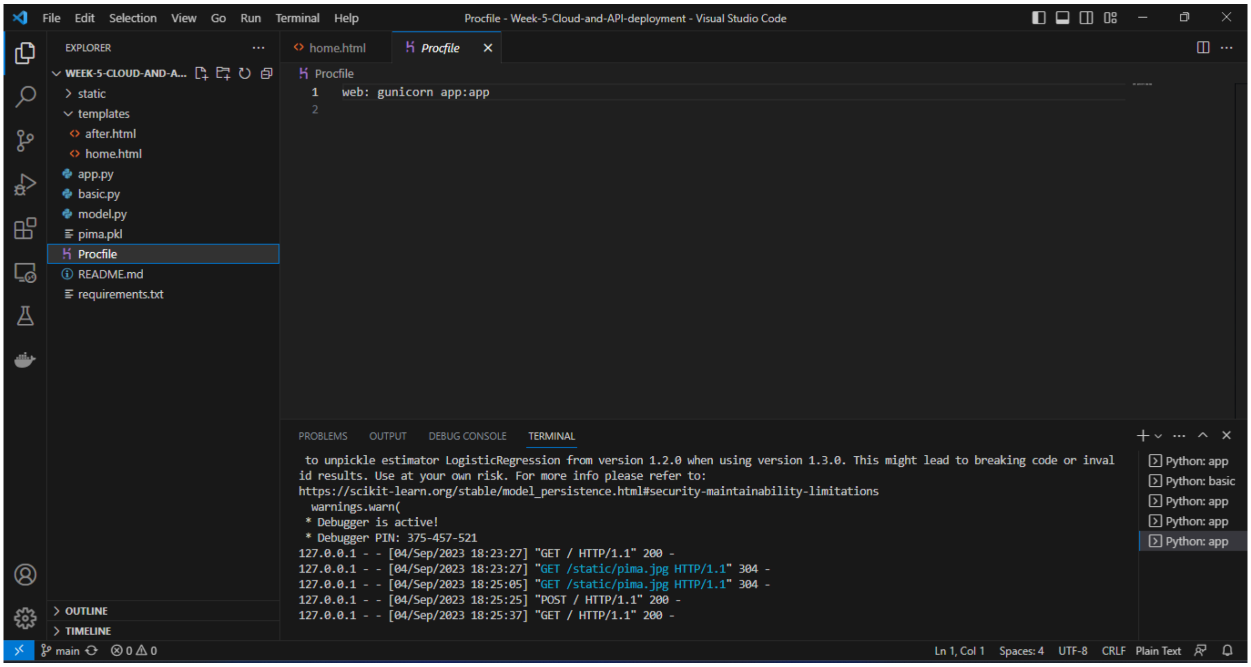
Create Requirement.Txt File



```
click==8.1.3
colorama==0.4.6
Flask==2.2.2
gunicorn==20.1.0
importlib-metadata==5.1.0
itsdangerous==2.1.2
Jinja2==3.1.2
joblib==1.2.0
MarkupSafe==2.1.1
numpy==1.23.5
pandas==1.5.2
pickle4==0.0.1
python-dateutil==2.8.2
pytz==2022.6
scikit-learn==1.2.0
scipy==1.9.3
six==1.16.0
threadpoolctl==3.1.0
Werkzeug==2.2.2
```

```
 to unpickle estimator LogisticRegression from version 1.2.0 when using version 1.3.0. This might lead to breaking code or inval
id results. Use at your own risk. For more info please refer to:
https://scikit-learn.org/stable/model_persistence.html#security-maintainability-limitations
  warnings.warn(
 * Debugger is active!
 * Debugger PIN: 375-457-521
127.0.0.1 - - [04/Sep/2023 18:23:27] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [04/Sep/2023 18:23:27] "GET /static/pima.jpg HTTP/1.1" 304 -
127.0.0.1 - - [04/Sep/2023 18:25:05] "GET /static/pima.jpg HTTP/1.1" 304 -
127.0.0.1 - - [04/Sep/2023 18:25:25] "POST / HTTP/1.1" 200 -
127.0.0.1 - - [04/Sep/2023 18:25:37] "GET / HTTP/1.1" 200 -
```

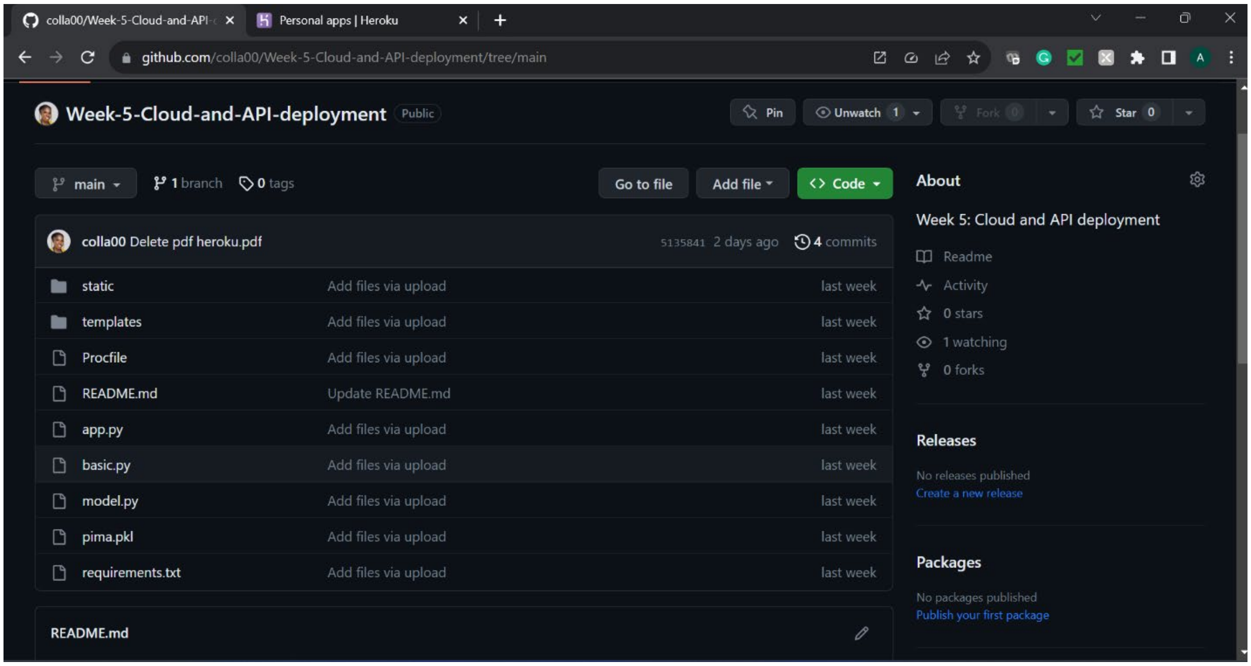
Create Procfile



```
web: gunicorn app:app
```

```
 to unpickle estimator LogisticRegression from version 1.2.0 when using version 1.3.0. This might lead to breaking code or inval
id results. Use at your own risk. For more info please refer to:
https://scikit-learn.org/stable/model_persistence.html#security-maintainability-limitations
  warnings.warn(
 * Debugger is active!
 * Debugger PIN: 375-457-521
127.0.0.1 - - [04/Sep/2023 18:23:27] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [04/Sep/2023 18:23:27] "GET /static/pima.jpg HTTP/1.1" 304 -
127.0.0.1 - - [04/Sep/2023 18:25:05] "GET /static/pima.jpg HTTP/1.1" 304 -
127.0.0.1 - - [04/Sep/2023 18:25:25] "POST / HTTP/1.1" 200 -
127.0.0.1 - - [04/Sep/2023 18:25:37] "GET / HTTP/1.1" 200 -
```

Create a Repository in GitHub. Upload Files and Commit.


colla00/Week-5-Cloud-and-API-
Personal apps | Heroku
github.com/colla00/Week-5-Cloud-and-API-deployment/tree/main
Pin
Unwatch 1
Fork 0
Star 0
main
1 branch
0 tags
Go to file
Add file
Code

Week-5-Cloud-and-API-deployment Public

colla00 Delete pdf heroku.pdf 5135841 2 days ago 4 commits

static	Add files via upload	last week
templates	Add files via upload	last week
Procfile	Add files via upload	last week
README.md	Update README.md	last week
app.py	Add files via upload	last week
basic.py	Add files via upload	last week
model.py	Add files via upload	last week
pima.pkl	Add files via upload	last week
requirements.txt	Add files via upload	last week

README.md

About

Week 5: Cloud and API deployment

Readme
Activity
0 stars
1 watching
0 forks

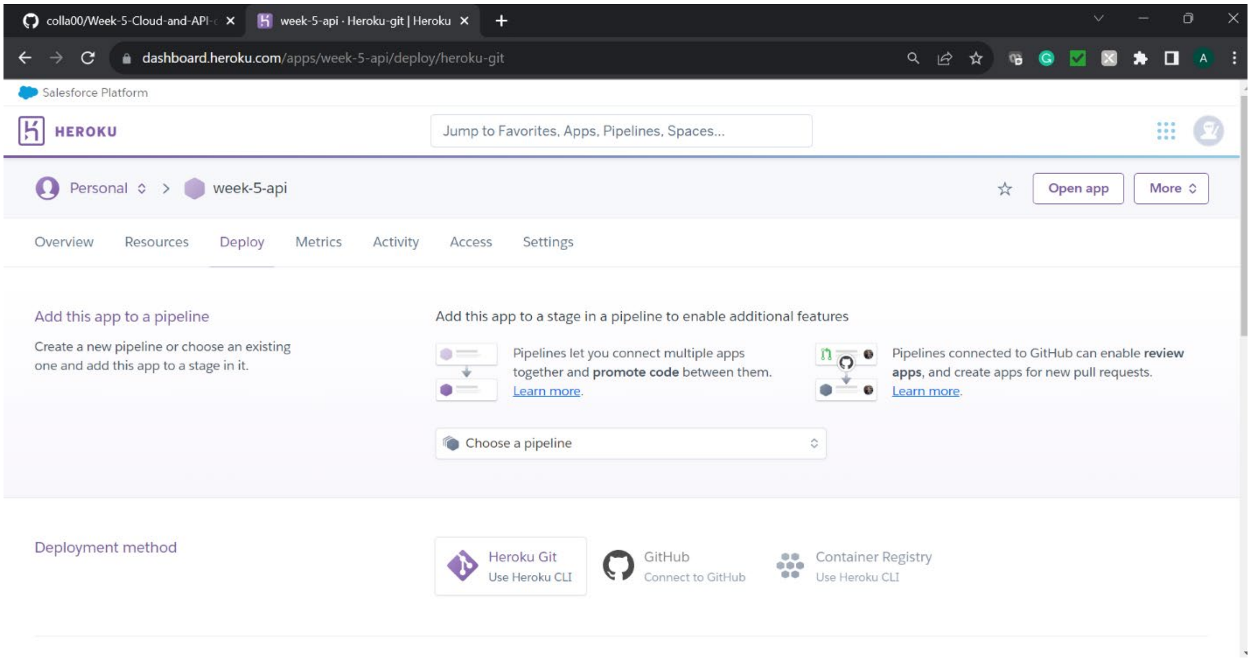
Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Login to Heroku, Select New, Create a New app And Create App Name


colla00/Week-5-Cloud-and-API-
week-5-api · Heroku-git | Heroku
dashboard.heroku.com/apps/week-5-api/deploy/heroku-git
Salesforce Platform
HEROKU
Jump to Favorites, Apps, Pipelines, Spaces...
Personal
week-5-api
Open app
More
Overview
Resources
Deploy
Metrics
Activity
Access
Settings

Add this app to a pipeline

Create a new pipeline or choose an existing one and add this app to a stage in it.

Add this app to a stage in a pipeline to enable additional features

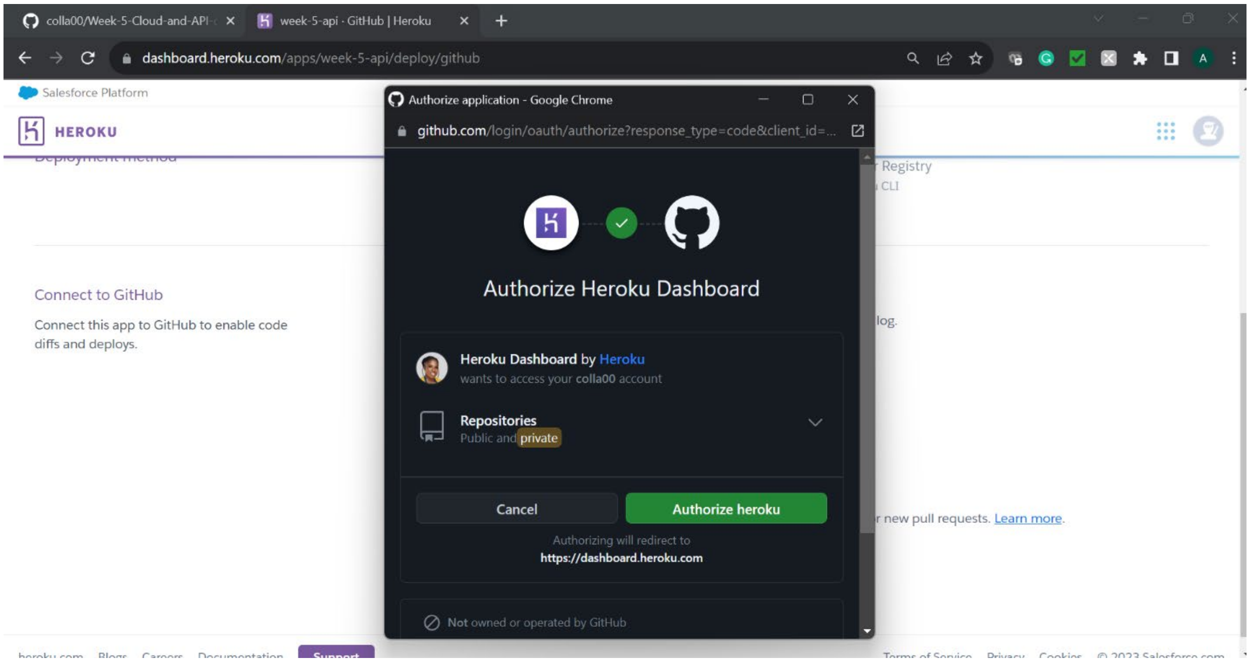
Pipelines let you connect multiple apps together and **promote code** between them. Learn more.

Pipelines connected to GitHub can enable **review apps**, and create apps for new pull requests. Learn more.

Choose a pipeline

Deployment method

Connect To GitHub and Repository



Authorize Heroku Dashboard

Heroku Dashboard by Heroku
wants to access your colla00 account

Repositories
Public and private

Authorizing will redirect to
https://dashboard.heroku.com

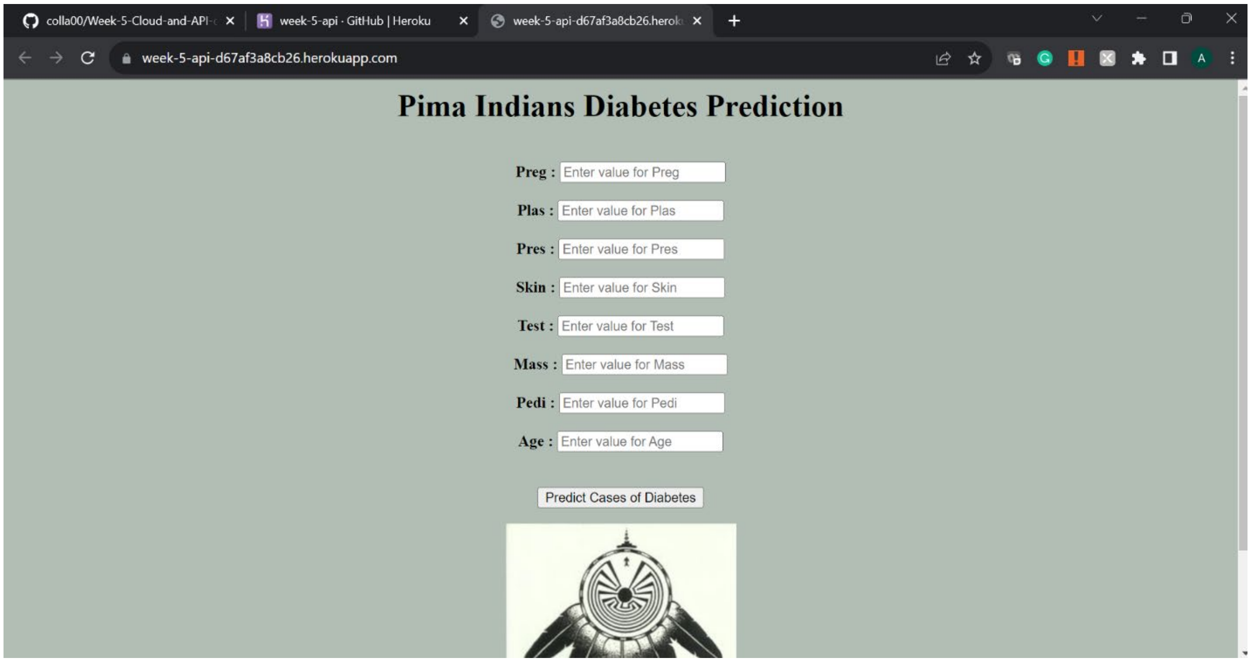
Not owned or operated by GitHub

Connect to GitHub

Connect this app to GitHub to enable code diffs and deploys.

colla00/Week-5-Cloud-and-API-
week-5-api - GitHub | Heroku
dashboard.heroku.com/apps/week-5-api/deploy/github
Salesforce Platform
HEROKU
Authorize application - Google Chrome
github.com/login/oauth/authorize?response_type=code&client_id=...
Cancel
Authorize heroku
r Registry
CLI
log.
r new pull requests. Learn more.
Support

Deploy Branch and View Deployed Branch


colla00/Week-5-Cloud-and-API-
week-5-api · GitHub | Heroku
week-5-api-d67af3a8cb26.heroku
week-5-api-d67af3a8cb26.herokuapp.com

Pima Indians Diabetes Prediction

Preg : Enter value for Preg

Plas : Enter value for Plas

Pres : Enter value for Pres

Skin : Enter value for Skin

Test : Enter value for Test

Mass : Enter value for Mass

Pedi : Enter value for Pedi

Age : Enter value for Age

Predict Cases of Diabetes