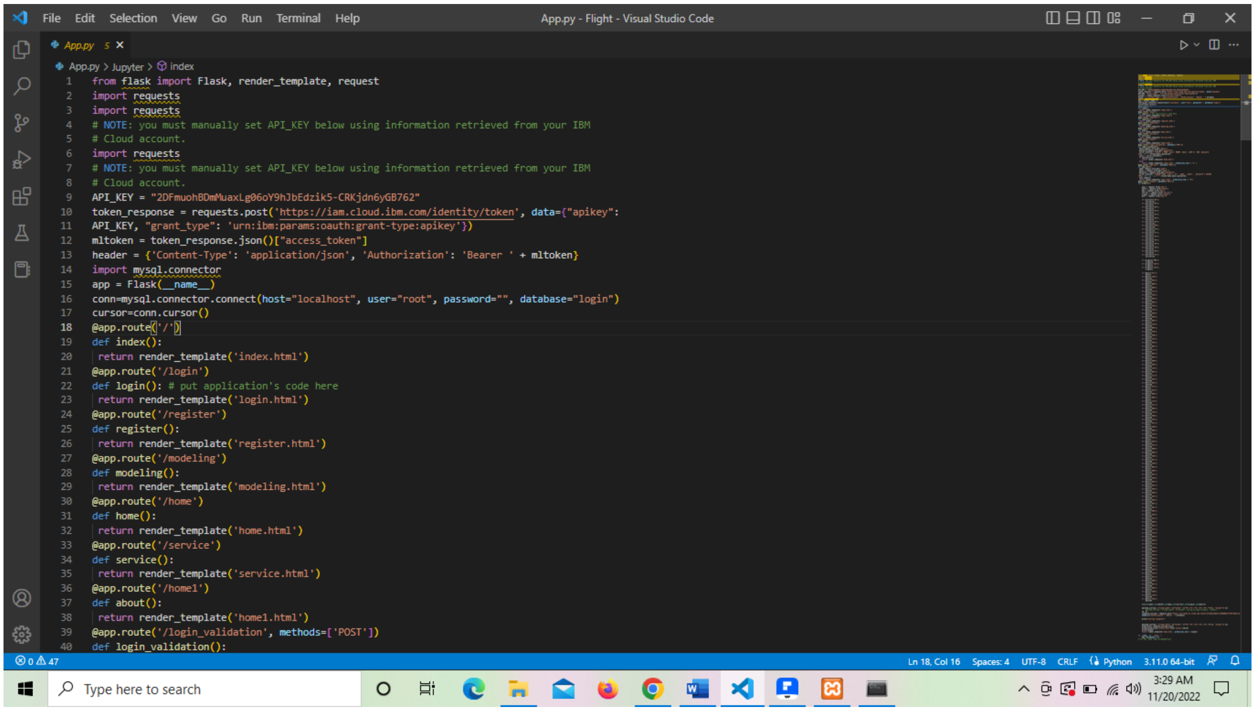
Build Python Code

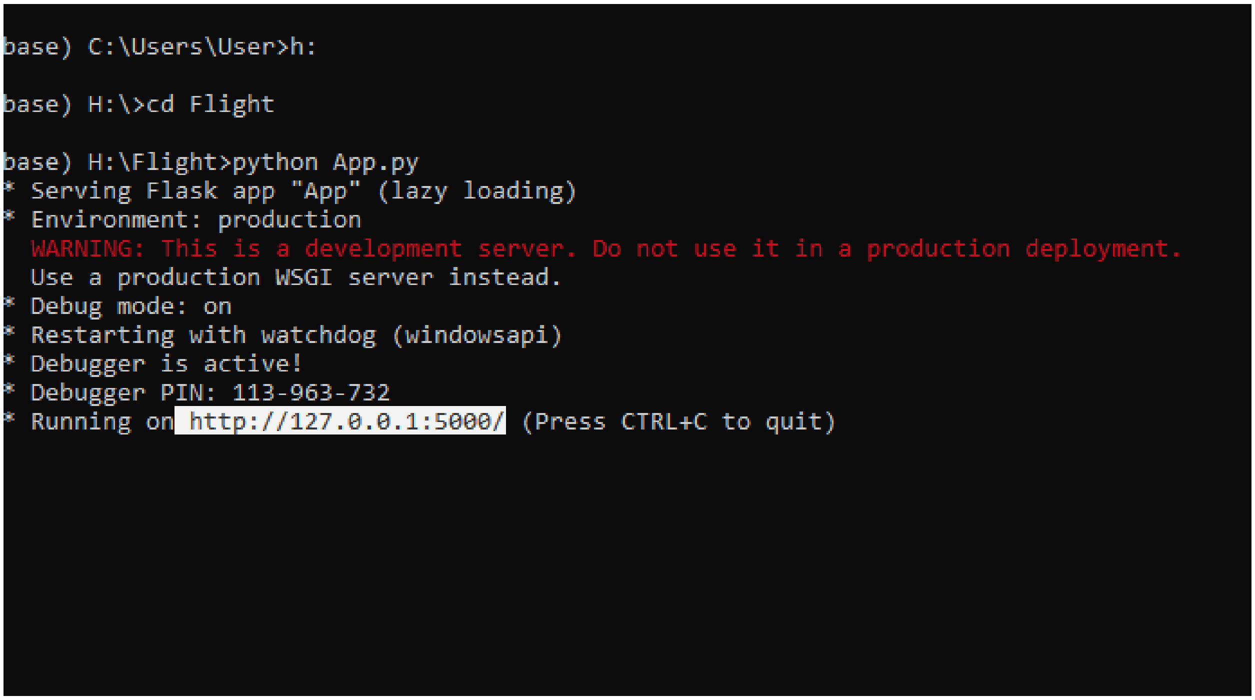
Date	20 November 2022
Team ID	PNT2022TMD52214
Project Name	Developing a Flight Delay Prediction using Machine Learning

- Main Python Script
  - Let us build a flask file ’flight delay analysis. ipynb which is a web framework written in python for server-side scripting. Let’s see step by step procedure for building the backend application.
  - You can also run this on vs Code, given as app.py
  - App starts running when the “__name__” constructor is called in main.
  - Render template is used to return html files.
  - “GET” method is used to take input from the user.
  - “POST” method is used to display the output to the user.
  - Three.html web pages are given.



```
from flask import Flask, render_template, request
import requests
import requests
# NOTE: you must manually set API_KEY below using information retrieved from your IBM
# Cloud account.
import requests
# NOTE: you must manually set API_KEY below using information retrieved from your IBM
# Cloud account.
API_KEY = "2DFmuohBDmMuaxLg06oY9hJbEdzik5-CRKjdn6yGB762"
token_response = requests.post('https://iam.cloud.ibm.com/identity/token', data={"apikey":
API_KEY, "grant_type": 'urn:ibm:params:oauth:grant-type:apikey'})
mltoken = token_response.json()["access_token"]
header = {'Content-Type': 'application/json', 'Authorization': 'Bearer ' + mltoken}
import mysql.connector
app = Flask(__name__)
conn=mysql.connector.connect(host="localhost", user="root", password="", database="login")
cursor=conn.cursor()
@app.route('/')
def index():
 return render_template('index.html')
@app.route('/login')
def login(): # put application's code here
 return render_template('login.html')
@app.route('/register')
def register():
 return render_template('register.html')
@app.route('/modeling')
def modeling():
 return render_template('modeling.html')
@app.route('/home')
def home():
 return render_template('home.html')
@app.route('/service')
def service():
 return render_template('service.html')
@app.route('/home1')
def about():
 return render_template('home1.html')
@app.route('/login_validation', methods=['POST'])
def login_validation():
```

Lastly, we run our app on the local host. Here we are running it on localhost:5000



```
base) C:\Users\User>h:

base) H:\>cd Flight

base) H:\Flight>python App.py
* Serving Flask app "App" (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: on
* Restarting with watchdog (windowsapi)
* Debugger is active!
* Debugger PIN: 113-963-732
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```