

Team Id - PNT2022TMID39717

Integrate Flask with Scoring Endpoint:

```
import requests
import flask
from flask import request, render_template, Flask
from flask_cors import CORS
import requests

API_KEY = "1LuQwMcCdkqNEAHL6QzF6F5zfSDv8Az4s0HlpzHirICX"
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}

app=flask.Flask(__name__)

@app.route('/')
def index():
    return render_template("a.html")

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

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

@app.route('/contact')
def contact():
    return render_template('contact.html')
```

```
@app.route('/Register',methods=['POST'])

def register():

    cyl=float(request.form['12'])
    dis=float(request.form['34'])
    hp=float(request.form['56'])
    w=float(request.form['3'])
    a=float(request.form['4'])
    my=float(request.form['5'])
    ori=float(request.form['6'])
    X = [[cyl, dis, hp, w, a, my, ori]]
    payload_scoring = {"input_data": [{"field": [['12', '34', '56', '3', '4', '5', '6']], "values": X}]}
    response_scoring = requests.post('https://us-south.ml.cloud.ibm.com/ml/v4/deployments/b15feb78-c8da-49a8-9d5d-ac409a0470eb/predictions?version=1.0',
    json=payload_scoring, headers={'Authorization': 'Bearer ' + mltoken})
    print(response_scoring)
    predictions = response_scoring.json()
    predict = predictions['predictions'][0]['values'][0][0]
    print("Final prediction :", predict)
    if(predict >46 ):
        return render_template("result.html",Efficiency = "Very High performance with mileage"+ str(predict)+" You can plan for a tour")
    elif(predict >29 and predict<=46):
        return render_template("result.html",Efficiency = "High performance with mileage "+ str(predict)+" Go for a healthy ride")
    elif(predict >17.5 and predict<=29):
        return render_template("result.html",Efficiency = "Medium performance with mileage "+ str(predict)+" Go for a ride nearby")
    elif(predict >9 and predict<=17.5):
        return render_template("result.html",Efficiency = "Low performance with mileage "+ str(predict)+" Don't go to a long distance")
    else:
        return render_template("result.html",Efficiency = "Worst performance with mileage "+ str(predict)+" carry extra fuel")

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