

Team ID	PNT2022TMID16005
Project Name	Efficient Water Quality Analysis and Prediction using Machine Learning

Python code

```

1 import numpy as np
2 from flask import Flask,render_template,request
3 import pickle
4 import requests
5
6 API_KEY = "THZC3nURVpmMSRhpBOMUeqJsJ6p40DZ4pp2FSnNkwyfY"
7 token_response = requests.post('https://iam.cloud.ibm.com/identity/token', data={"apikey":
8 API_KEY, "grant_type": 'urn:ibm:params:oauth:grant-type:apikey'})
9 mltoken = token_response.json()["access_token"]
10
11 header = {'Content-Type': 'application/json', 'Authorization': 'Bearer ' + mltoken}
12
13
14 app = Flask(__name__)
15 model = pickle.load(open('wqi.pkl','rb'))
16 @app.route('/',methods=['GET'])
17 def home():
18     return render_template("index.html")
19 @app.route('/login',methods = ['POST'])
20 def login():
21     year = request.form["year"]
22     do = request.form["do"]
23     ph = request.form["ph"]
24     co = request.form["co"]
25     bod = request.form["bod"]
26     na = request.form["na"]
27     tc = request.form["tc"]
28     total = [[int(year),float(do),float(ph),float(co),float(bod),float(na),float(tc)]]
29
30     payload_scoring = {"input_data": [{"fields": [['year','do','ph','co','bod','na','tc']], "values": total]}}
31
32     response_scoring = requests.post('https://us-south.ml.cloud.ibm.com/ml/v4/deployments/735973ab-d35c-4182-90f9-ca418497ced0/predictions?version=
33 headers={'Authorization': 'Bearer ' + mltoken})
34     print("Scoring response")
35     print(response_scoring.json())
36     predictions=response_scoring.json()
37     y_pred=predictions['predictions'][0]['values'][0][0]

```

```

72
73     if(y_pred >= 95 and y_pred <=100):
74         return render_template("index.html",showcase = "Excellent, The Predicted Value is "+str(y_pred))
75     elif(y_pred >=89 and y_pred <=94):
76         return render_template("index.html",showcase = "Very Good, The Predicted Value is "+str(y_pred))
77     elif(y_pred >=80 and y_pred <=88):
78         return render_template("index.html",showcase = "Good, The Predicted Value is "+str(y_pred))
79     elif(y_pred>=65 and y_pred<=79):
80         return render_template("index.html",showcase = "Fair, The Predicted Value is "+str(y_pred))
81     elif(y_pred>=45 and y_pred<=64):
82         return render_template("index.html",showcase = "Marginal, The Predicted Value is "+str(y_pred))
83     else:
84         return render_template("index.html",showcase = "Poor, The Predicted Value is "+str(y_pred))
85
86 if __name__ == '__main__':
87     app.run(debug = True,port = 5000)

```