SPRINT 3

Date	15 November 2022
Team ID	PNT2022TMID46078
Project Name	Efficient Water Quality Analysis and Prediction Using Machine Learning
Maximum Marks	8 Marks

PYTHON CODE:

App.py

```python
from flask import Flask, render_template, flash, request,session
from cloudant.client import  Cloudant
import pickle


client = Cloudant.iam("d3bd6270-afc4-41f1-a4d3-44f7b38e1f0d-
bluemix","y_O6N6Py64ajGj0sPg4KnnyFHixq3L8svUGBixNv5wjq",connect=True)
my_database = client.create_database("database-dharan")


app = Flask(__name__)
app.config.from_object(__name__)
app.config['SECRET_KEY'] = '7d441f27d441f27567d441f2b6176a'


@app.route("/")
def homepage():

    return render_template('index.html')


@app.route("/userhome")
def userhome():

    return render_template('userhome.html')
@app.route("/addamount")

@app.route("/NewUser")
def NewUser():

    return render_template('NewUser.html')
```

```python
@app.route("/user")
def user():

    return render_template('user.html')


@app.route("/newuse",methods=['GET','POST'])
def newuse():
    if request.method == 'POST':#

        x = [x for x in request.form.values()]
        print(x)
        data = {
            '_id': x[1],
            'name': x[0],
            'psw': x[2]
        }
        print(data)
        query = {'_id': {'Seq': data['_id']}}
        docs = my_database.get_query_result(query)
        print(docs)
        print(len(docs.all()))
        if (len(docs.all()) == 0):
            url = my_database.create_document(data)
            return render_template('goback.html', data="Register, please login using
your details")
        else:
            return render_template('goback.html', data="You are already a member,
please login using your details")

@app.route("/userlog", methods=['GET', 'POST'])
def userlog():
        if request.method == 'POST':

            user = request.form['_id']
            passw = request.form['psw']
            print(user, passw)

            query = {'_id': {'$eq': user}}
            docs = my_database.get_query_result(query)
            print(docs)
            print(len(docs.all()))
            if (len(docs.all()) == 0):
                return render_template('goback.html', pred="The username is not
found.")
            else:
                if ((user == docs[0][0]['_id'] and passw == docs[0][0]['psw'])):
```

```python
                return render_template("userhome.html")
            else:
                return render_template('goback.html',data="user name and password
incorrect")




@app.route("/predict", methods=['GET', 'POST'])
def predict():
    if request.method == 'POST':
        outttt =""

        year = request.form["year"]
        do = request.form["do"]
        ph = request.form["ph"]
        co = request.form["co"]
        bod = request.form["bod"]
        na = request.form["na"]
        tc = request.form["tc"]

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



        total = [[int(year), float(do), float(ph), float(co), float(bod), float(na),
float(tc)]]
        #total = int(year)+ float(do)+ float(ph)+ float(co)+float(bod)+float(na)+
float(tc)
        y_pred = model.predict(total)
        print(y_pred)

        y_pred1 = y_pred[[0][0]]

        y_pred2 = y_pred1[[10][0]]


        print(y_pred2)

        if (y_pred2 >= 95 and y_pred2 <= 100):

            outttt ="Excellent, the Predicted value is " + str(y_pred2)


        elif (y_pred2 >= 89 and y_pred2 <= 94):
            outttt = "Very good, the Predicted value is " + str(y_pred2)

        elif (y_pred2 >= 80 and y_pred2 <= 88):

            outttt="Good, the Predicted value is " + str(y_pred2)
```

```python
        elif(y_pred2 >= 65 and y_pred2 <= 79):
            outttt = "Fair, the Predicted value is " + str(y_pred2)

        elif (y_pred2 >= 45 and y_pred2 <= 64):
            outttt ="Marginal, the Predicted value is " + str(y_pred2)
        else:
            outttt="Poor, the Predicted value is " + str(y_pred2)

        return render_template('userhome.html', prediction=outttt)




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