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

app = Flask(__name__)
model = pickle.load(open('CKD.pkl', 'rb'))


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


@app.route('/Prediction', methods=['POST', 'GET'])
def prediction():
    return render_template('indexnew.html')


@app.route('/Home', methods=['POST', 'GET'])
def my_home():
    return render_template('home.html')


@app.route('/predict', methods=['POST'])
def predict():
    #input_features = ([int(x) for x in request.form.values()])
    blood_urea = request.form["blood_urea"]
    blood_glucose_random = request.form["blood_glucose_random"]
    anemia = request.form["Anemia"]
    if (anemia == "no"):
        anemia = 0
    if (anemia == "yes"):
        anemia = 1
    coronary_artery_disease = request.form["coronary_artery_disease"]
    if (coronary_artery_disease == "no"):
        coronary_artery_disease = 0
    if(coronary_artery_disease == "yes"):
        coronary_artery_disease = 1

    pus_cell = request.form["pus_cell"]
    if (pus_cell == "no"):
        pus_cell = 0
    if (pus_cell == "yes"):
        pus_cell = 1
```

```
    red_blood_cell = request.form["red_blood_cell"]
    if (red_blood_cell == "no"):
        red_blood_cell = 0
    if (red_blood_cell == "yes"):
        red_blood_cell = 1

    diabetics_mellitus = request.form["diabetics_mellitus"]
    if (diabetics_mellitus == "no"):
        diabetics_mellitus = 0
    if (diabetics_mellitus == "yes"):
        diabetics_mellitus = 1

    pedal_edema = request.form["pedal_edema"]
    if (pedal_edema == "no"):
        pedal_edema = 0
    if (pedal_edema == "yes"):
        pedal_edema = 1

    input_features =
[int(blood_urea),int(blood_glucose_random),int(anemia),int(coronary_artery_di
sease),int(pus_cell),int(red_blood_cell),int(diabetics_mellitus),int(pedal_ed
ema)]
    #input_features =
[int(red_blood_cell),int(pus_cell),int(blood_glucose_random),int(blood_urea),
int(pedal_edema),int(anemia),int(diabetics_mellitus),int(coronary_artery_dise
ase)]
    print(input_features)
    features_value = [np.array(input_features)]


    #features_name = ['red_blood_cells','pus_cell','blood glucose
random','blood_urea','pedal_edema','anemia','diabetesmellitus','coronary_arte
ry_disease']
    features_name = ['blood_urea','blood glucose
random','anemia','coronary_artery_disease','pus_cell','red_blood_cells','diab
etesmellitus','pedal_edema' ]
    df = pd.DataFrame(features_value, columns=features_name)
    output = model.predict(df)
    return render_template('result.html', prediction_text=output)


# Press the green button in the gutter to run the script.
if __name__ == '__main__':
    app.run(host='localhost', debug=True)

# See PyCharm help at https://www.jetbrains.com/help/pycharm/
```