

Assignment -1

| | |
|---------------------|-------------------|
| Assignment Date | 27 September 2022 |
| Student Name | C.Saranya |
| Student Roll Number | 912619104024 |
| Maximum Marks | 2 Marks |

Question-1:

Create registration page in html with username, email and phone number and by using POST method display it in next html page.

```
<link rel="stylesheet" href="styles.css" type="text/css">

<div class="login-box">

    <h2>Sign Up</h2>

    <form action="/login" method="post">

        <div class="user-box"><input type="text" id="username" name="username"
required=""><label>Enter Username</label></div>

        <div class="user-box"><input type="email" id="mail" name="mail"
required=""><label>Enter Email id</label></div>

        <div class="user-box"> <input type="number" id="number" name="number"
required=""><label>Enter contact no</label></div>

        <button class="btn" >

            <input type="submit" value="SignUp"/></button>

    </form>

</div>
```

```
from flask import Flask,render_template,redirect, url_for, request

app = Flask(__name_)
```

```
@app.route('/')

def signup():

    return render_template('flask.html')

@app.route('/success/<name>/<email>/<contact>')

def success(name,email,contact):

    # return 'Welcome %s %s %d' % name %email

    return f'welcome {name} {email} {contact}'
```

```

@app.route('/login',methods=['POST','GET'])
def login():
    if request.method=='POST':
        user=request.form['username']
        eemail=request.form['mail']
        contactno=request.form['number']
        return redirect(url_for('success',name=user,email=eemail,contact=contactno))

    else:
        user=request.args.get('username')
        eemail=request.args.get('mail')
        contactno=request.args.get('number')
        return redirect(url_for('success',name=user,email=eemail,contact=contactno))

if __name__ == '__main__':
    app.run()

```

Output:



Question-2:

Develop a flask program which should contain at least 5 packages used from pypi.org.

```

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import pendulum

```

```

from flask import Flask
import io
from flask import Response
from matplotlib.backends.backend_agg import FigureCanvasAgg as FigureCanvas
from matplotlib.figure import Figure
import seaborn as sns

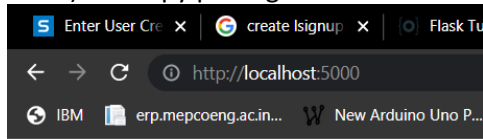
app = Flask(__name__)
@app.route('/')
def check():
    x = 5
    y=6
    z=np.add(x,y)
    return '%d' %z
@app.route('/plot')
def plot_png():
    plt.rcParams["figure.figsize"] = [7.50, 3.50]
    plt.rcParams["figure.autolayout"] = True
    fig = Figure()
    axis = fig.add_subplot(1, 1, 1)
    xs = np.random.rand(100)
    ys = np.random.rand(100)
    axis.plot(xs, ys)
    output = io.BytesIO()
    FigureCanvas(fig).print_png(output)
    return Response(output.getvalue(), mimetype='image/png')
@app.route('/pandas')
def pandas():
    ser=pd.Series([0.25,0.7,0.5])
    return '%f' %ser[0]
@app.route('/seaborn')
def seaborn():
    fig=Figure()
    x=[i for i in range(100)]
    y=[i for i in range(100)]
    sns.set()
    fig,ax=plt.subplots(1,1)
    sns.lineplot(x,y)
    img = io.BytesIO()
    FigureCanvas(fig).print_png(img)
    return Response(img.getvalue(),mimetype='img/png')
@app.route('/pendulum')
def pendulum1():
    local = pendulum.local(2020, 11,27)
    zone=local.timezone.name
    return '%s' %zone

```

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

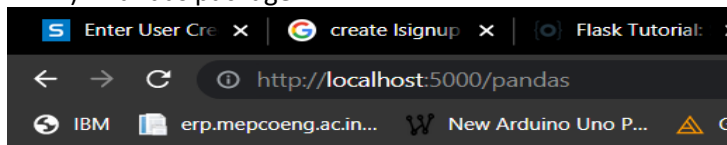
Output:

1) Numpy package



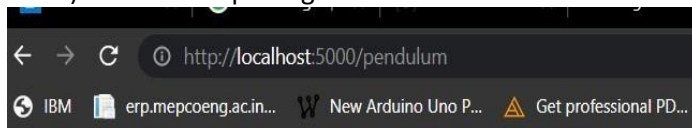
Numpy 11

2) Pandas package



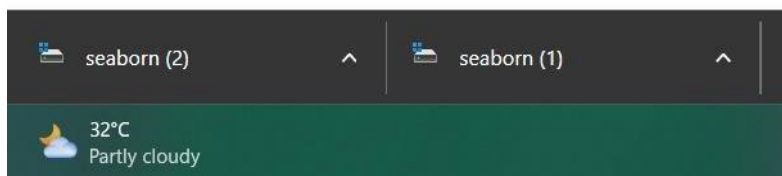
0.250000

3) Pendulum package



Asia/Calcutta

4) using seaborn package



5) Matplotlib package

