Assignment-2

-Name: Sangamitra S

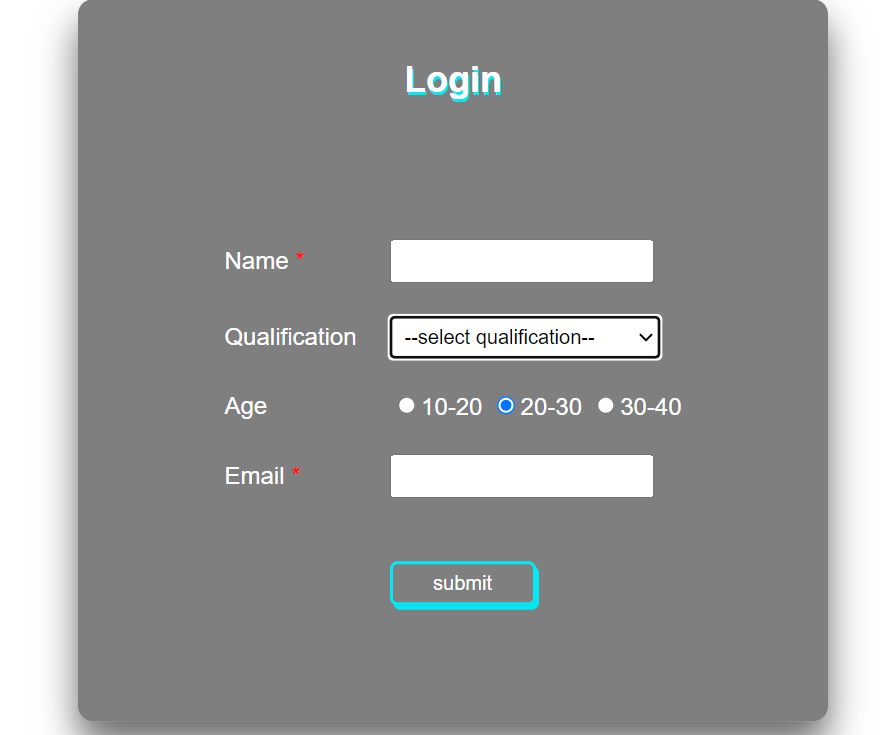
-Reg No: 711119104052

1. Create form of type input text, email, password, radio button text Area, drop down and navigate to success page and display files of form in table (CSS, HTML).

**PROGRAM:**

|  |
| --- |
| <html> |
|  | <head> |
|  | <title> Input Form |
|  | </title> |
|  | <meta charset="utf-8"> |
|  | <meta name="viewport" content="width=device-width, initial-scale=1"> |
|  | <link rel="stylesheet" href="[style.css](file:///C:\Users\SANGAMITRA\Downloads\login\login\style.css)"> |
|  |  |
|  |  |
|  | <body> |
|  | <div class="login-box"> |
|  | <h2>Login</h2> |
|  | <form> |
|  | <table align="center" style="padding-top:50px;"> |
|  |  |
|  | <tr> |
|  | <td> |
|  | <label>Name<span style="color:red;"> \* </span></label></td> |
|  | <td><input type="text" name="" required=""></td> |
|  | </tr> |
|  |  |
|  |  |
|  | <tr> |
|  | <td><label>Qualification</label></td> |
|  | <td> <select name="qualify" id="qualify" style="width:180px;" > |
|  | <option value="">--select qualification--</option> |
|  | <option value="tenth">10th</option> |
|  | <option value="twelvth">12th</option> |
|  | <option value="diploma">Diploma</option> |
|  | <option value="bachelor">Bachelor's Degree</option> |
|  | <option value="master">Master's Degree</option> |
|  |  |
|  |  |
|  | </select></td> |
|  | </tr> |
|  |  |
|  |  |
|  | <tr> |
|  | <td> <label>Age</label></td> |
|  | <td> <input type="radio" name="age10" required="" ><span style="color:#fff">10-20</span>&nbsp;<input type="radio" name="age20" required="" ><span style="color:#fff">20-30</span>&nbsp;<input type="radio" name="age30" required=""><span style="color:#fff">30-40</span></td> |
|  | </tr> |
|  |  |
|  | <tr> |
|  | <td><label>Email<span style="color:red;"> \* </span></label></td> |
|  | <td> <input type="email" name="email" required=""></td> |
|  | <tr> |
|  | <tr> |
|  | <td></td> |
|  | <td><button type="submit" name="submit" class="btn" value="submit">submit</button></td> |
|  | </tr> |
|  |  |
|  | </table> |
|  | </form> |
|  | </div> |
|  |  |
|  | </body> |
|  | </html> |

Output:



1. For CSS create external style sheet for above task (separate CSS file and link that in html).

tr,td {

padding:10px;

border-radius:3px;

}

body {

font-family: sans-serif;

}

.login-box {

position: absolute;

top: 50%;

left: 50%;

width: 500px;

padding: 40px;

color:#000;

transform: translate(-50%, -50%);

background: rgba(0,0,0,.5);

box-sizing: border-box;

box-shadow: 0 15px 25px rgba(0,0,0,.6);

border-radius: 10px;

}

.login-box h2 {

margin: 0 0 30px;

color: #fff;

text-align: center;

text-shadow: 1px 2px #03e9f4;

}

.login-box label {

margin: 0 0 30px;

color: #fff;

text-align: center;

}

.login-box input {

color: #fff;

padding:5px;

}

.login-box select {

padding:5px;

}

.btn {

width: 50%;

background:transparent;

border:2px solid #03e9f4;

color:#fff;

margin: 18px 0 9px 0;

padding:5px 5px 5px 5px;

border-radius:5px;

box-shadow: 2px 3px #03e9f4;

}

}

1. Create sample program for Flask HTTP methods (list or map and Perform operations of PUT, GET, DELETE and POST.

Program:

from flask import Flask, render\_template,request,flash,redirect,url\_for

import sqlite3 as sql

app = Flask(\_\_name\_\_)

app.secret\_key='admin123'

con=sql.connect("db\_web.db")

con.execute("CREATE TABLE IF NOT EXISTS users(pid INTEGER PRIMARY KEY ,UNAME TEXT,CONTACT TEXT)")

con.close()

app=Flask(\_\_name\_\_)

@app.route("/")

@app.route("/index")

def index():

    con=sql.connect("db\_web.db")

    con.row\_factory=sql.Row

    cur=con.cursor()

    cur.execute("select \* from users")

    data=cur.fetchall()

    return render\_template("index.html",datas=data)

@app.route("/add\_user",methods=['POST','GET'])

def add\_user():

    if request.method=='POST':

        uname=request.form['uname']

        contact=request.form['contact']

        con=sql.connect("db\_web.db")

        cur=con.cursor()

        cur.execute("insert into users(UNAME,CONTACT) values (?,?)",(uname,contact))

        con.commit()

        # flash('User Added','success')

        return redirect(url\_for("index"))

    return render\_template("add\_user.html")

@app.route("/edit\_user/<string:uid>",methods=['POST','GET'])

def edit\_user(uid):

    if request.method=='POST':

        uname=request.form['uname']

        contact=request.form['contact']

        con=sql.connect("db\_web.db")

        cur=con.cursor()

        cur.execute("update users set UNAME=?,CONTACT=? where pid=?",(uname,contact,uid))

        con.commit()

        # flash('User Updated','success')

        return redirect(url\_for("index"))

    con=sql.connect("db\_web.db")

    con.row\_factory=sql.Row

    cur=con.cursor()

    cur.execute("select \* from users where pid=?",(uid,))

    data=cur.fetchone()

    return render\_template("edit\_user.html",datas=data)

@app.route("/delete\_user/<string:uid>",methods=['GET'])

def delete\_user(uid):

    con=sql.connect("db\_web.db")

    cur=con.cursor()

    cur.execute("delete from users where pid=?",(uid,))

    con.commit()

    # flash('User Deleted','warning')

    return redirect(url\_for("index"))

if \_\_name\_\_=='\_\_main\_\_':

    app.secret\_key='admin123'

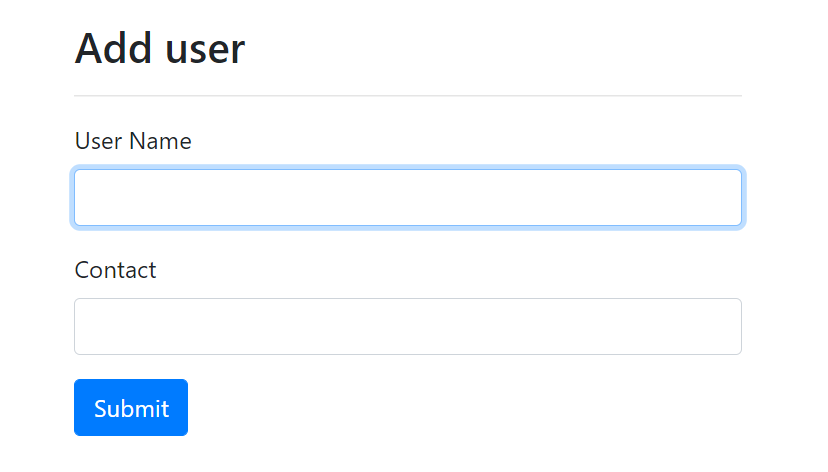
    app.config['SESSION\_TYPE'] = 'filesystem'

    app.config['SECRET\_KEY'] = 'admin123'

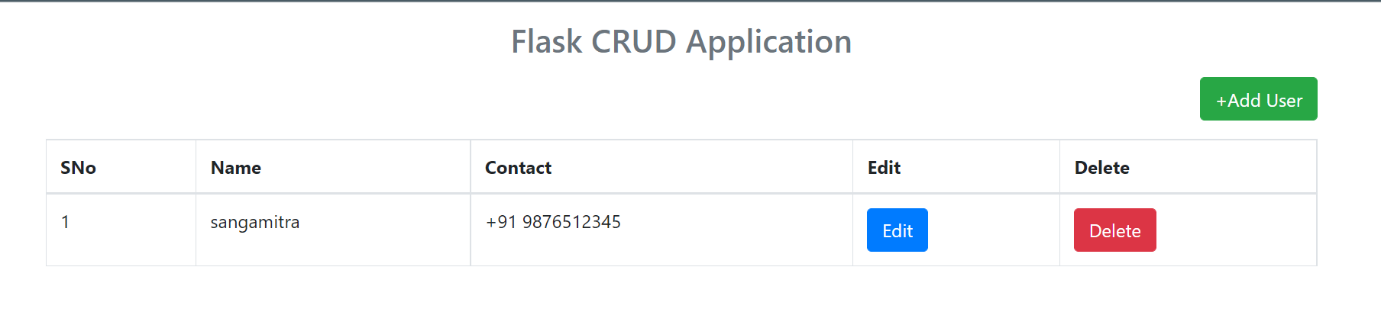
    app.run(debug=True)

Flask HTTP method :

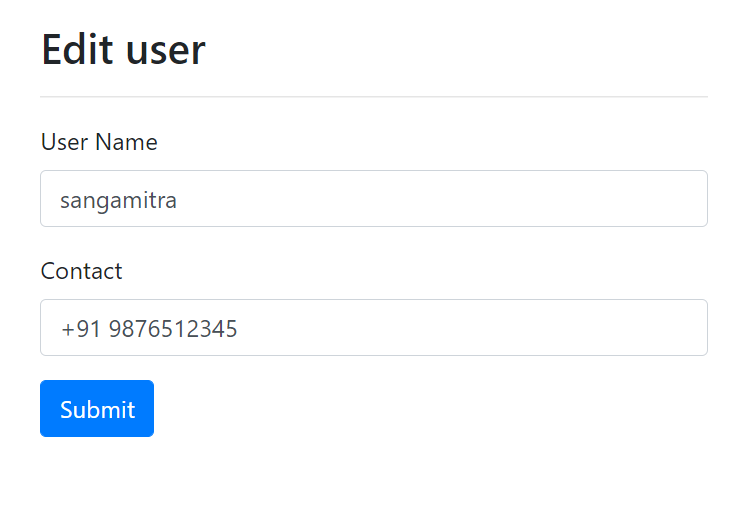
1. Login page



1. Data stored that can be edit, delete and add another id.



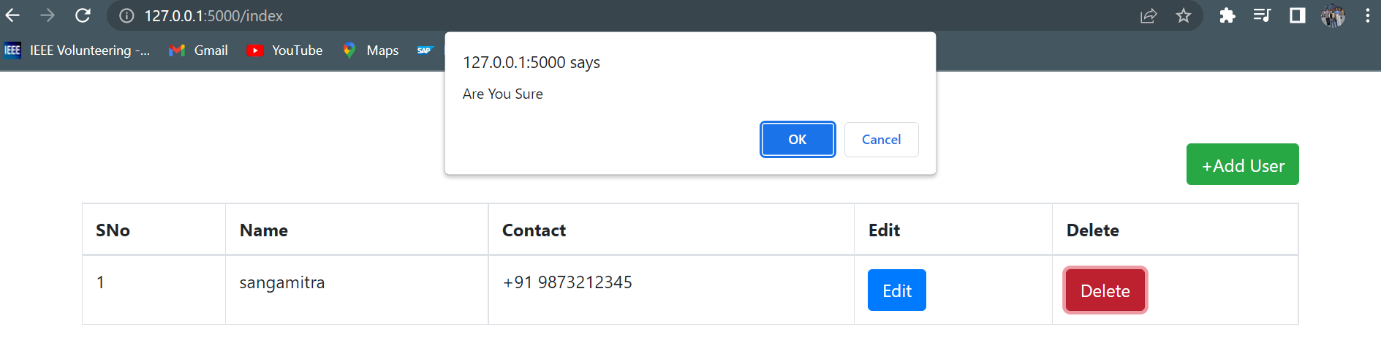
1. Can Edit the Use ID



1. Can see the changes



1. If needed can delete



1. It will be deleted from DB

