In [1]:
```
from flask import Flask, request, redirect, url_for, render_template
from werkzeug.wrappers import Request, Response
import sqlite3
```

In [2]:
```
dataBase= "../DBsqlite/db_base.db"

class DB:
    def ejecutar_consulta(self, consulta, parametros = ()):
        with sqlite3.connect(dataBase) as conn:
            self.cursor = conn.cursor()
            result = self.cursor.execute(consulta, parametros)
            conn.commit()
            return result
```

In [3]:
```
app = Flask(__name__)
```

In [4]:
```
db=DB()

@app.route('/')
@app.route('/home')
def home():
    name = request.args.get('usuario')
    datos = db.ejecutar_consulta("SELECT * FROM producto")
    lista = list()
    for dato in datos:
        lista.append(dato)
    return render_template("index.html", data=lista, name=name)

@app.route('/register')
def registrar():
    return render_template("persona.html")

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

@app.route('/create_person', methods=['POST'])
def create_Person():

    if request.method == 'POST':
        nombre = request.form.get('nombre')
        direccion = request.form.get('direccion')
        telefono = request.form.get('telefono')
        email = request.form.get('email')

        sql = "INSERT INTO cliente (nombre,direccion,telefono,email) VALUES(?,?,?,?)
        parametros = (nombre,direccion,telefono,email)
        db.ejecutar_consulta(sql,parametros)

        return redirect(url_for('home'))

@app.route('/search')
@app.route('/search', methods=['POST'])
def buscar():
    if request.method == 'POST':
        id_cliente = request.form.get('buscar')
        sql="SELECT * FROM cliente WHERE id=?"
        parametros=[id_cliente,]
        cliente = db.ejecutar_consulta(sql,parametros)
        for id_c in cliente:
            if id_c[0] != 0:
                estado = "disabled"
                return render_template("persona.html", dato=str(id_c[0]))
            else:
                print("no")

@app.route('/searchA')
@app.route('/searchA', methods=['POST'])
def buscarA():
    if request.method == 'POST':
        id_cliente = request.form.get('buscar')
        sql="SELECT id FROM cliente WHERE id=?"
        parametros=[id_cliente,]
        cliente = db.ejecutar_consulta(sql,parametros)
        for id_c in cliente:
            if id_c[0] != 0:
                return render_template("update.html", dato=str(id_c[0]))
            else:
                print("no")

@app.route('/delete')
@app.route('/delete/<int:id>')
def delete(id):
    if(id !=0):
        sql="DELETE FROM cliente WHERE id=?"
        parametros = (id,)
        pragma="PRAGMA foreign_keys=ON"
        db.ejecutar_consulta(pragma,)
        db.ejecutar_consulta(sql,parametros)
        return 'Eliminado...'

@app.route('/upDate')
@app.route('/upDate/<int:id>',methods=['POST'])
def upDate(id):
    if request.method == 'POST':
        nombre = request.form.get('nombre')
        direccion = request.form.get('direccion')
        telefono = request.form.get('telefono')
        email = request.form.get('email')

        sql="UPDATE cliente SET nombre=?,direccion=?,telefono=?,email=? WHERE id=?"
        parametros = (nombre,direccion,telefono,email,id)
        db.ejecutar_consulta(sql,parametros)
        return redirect(url_for('home'))


@app.route('/seleccionar/<string:name>/<int:id>')
def seleccionar(id=None, name=None):
    if (id !=None and name != None):
        sql = "SELECT valor FROM producto WHERE id=?"
        parametros = [id,]
        precios = db.ejecutar_consulta(sql,parametros)
        #name = request.args.get('usuario')
        for precio in precios:
            print (name,precio[0])
            return redirect(url_for('pago', usuario=name, valor=precio[0]))



@app.route('/pago')
def pago():
    usuario = request.args.get('usuario')
    valor = request.args.get('valor')
    if (usuario != "" and valor !=None):
        return render_template('factura.html', usuario=usuario,valor=valor)

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

@app.route('/consulta', methods=['POST'])
def consulta():
    if request.method=='POST':
        email = request.form.get('email')
        sql = 'SELECT * FROM cliente WHERE email=?'
        parametros=[email,]
        clientes = db.ejecutar_consulta(sql,parametros)
        for cliente in clientes:
            nombreCliente = cliente[1]
            print(nombreCliente)
            return redirect(url_for('home', usuario=nombreCliente))
```

In [ ]:

```
if __name__ == '__main__':
    from werkzeug.serving import run_simple
    run_simple('localhost', 9001,app)
```

```
 * Running on http://localhost:9001/ (Press CTRL+C to quit)
127.0.0.1 - - [11/May/2021 16:56:48] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [11/May/2021 16:56:51] "GET /seleccionar/None/1 HTTP/1.1" 302 -
127.0.0.1 - - [11/May/2021 16:56:51] "GET /pago?usuario=None&valor=30.0 HTTP/1.1" 20
0 -
None 30.0
127.0.0.1 - - [11/May/2021 16:57:51] "GET /pago?usuario=None&valor=30.0 HTTP/1.1" 20
0 -
127.0.0.1 - - [11/May/2021 16:57:53] "GET /pago?usuario=None&valor=30.0 HTTP/1.1" 20
0 -
127.0.0.1 - - [11/May/2021 16:57:53] "GET /pago?usuario=None&valor=30.0 HTTP/1.1" 20
0 -
127.0.0.1 - - [11/May/2021 16:57:54] "GET /pago?usuario=None&valor=30.0 HTTP/1.1" 20
0 -
127.0.0.1 - - [11/May/2021 16:57:54] "GET /pago?usuario=None&valor=30.0 HTTP/1.1" 20
0 -
127.0.0.1 - - [11/May/2021 16:57:54] "GET /pago?usuario=None&valor=30.0 HTTP/1.1" 20
0 -
127.0.0.1 - - [11/May/2021 16:57:54] "GET /pago?usuario=None&valor=30.0 HTTP/1.1" 20
0 -
127.0.0.1 - - [11/May/2021 16:57:55] "GET /pago?usuario=None&valor=30.0 HTTP/1.1" 20
0 -
127.0.0.1 - - [11/May/2021 16:57:57] "GET /pago?usuario=None&valor=30.0 HTTP/1.1" 20
0 -
127.0.0.1 - - [11/May/2021 16:57:57] "GET /pago?usuario=None&valor=30.0 HTTP/1.1" 20
0 -
```

In [ ]: