Instalar libreria

pip install Flask-SQLALchemy

posterior usar el import

from flask-sqlalchemy import SQLALchemy

from **sqlalchemy.orm** import Declarativebase

from **sqlalchemy.orm** import Mappedm mapped\_column

config = dotenv\_values(´.env´)

app=Flas(\_\_name\_\_)

crear la cadena de conexión

app.confing[SQLALCHEMY\_DATABASE\_URI“]= “squite://weather.sqlite” ## indica donde va a estar la base de datos

db=SQLALchemy(app) ## vinculo de la base de datos con la app

##creación de modelo

Class City(db.Model):

id: Mapped[int] mapped\_column(db.Integer, primay\_key=True,autoincrement=True)

name: Mapped[str] = mapped\_column(db.String, unique=True, nullable=False)

#sentencia que crea tablas

with app.app\_context():

db.create\_all()

desde aquí para abajo van los códigos del programa **app**

**después de return\_template(about.html)**

**#metodos disponibles**

**@app.route(´/clima´, methods=[´GET´,´POST´])**

def **clima():**

**#Si es post es que pulsan el boton Agregar Ciudad**

**If request.method == ´POST´:**

**New\_city = request.form.get(´city´)**

**If new\_city:**

**obj= City=name=new\_city)**

**db.session.add(obj)**

**db.session.commit()**

**#Llamo a todas las ciudades select**

**cities = City.queery.all()**

**weather\_data = []**

**for city in cities:**

**r=get\_weather\_data(city.name**

**weather = {**

**´city : city.name,**

**´temparature´ : r[´main´][´temp´],**

**´description´ : r[´weather´][0][´desription´],**

**´icon´ : r[´weather´][0][´icon´],**

**}**

**Weather\_data.append(weather)**

**Desde aquí va el return render\_template**

**PARA CREAR LA BASE DE DATOS SE**

**SQL LITE 3 en la pc**

**DB.Browser.for.SQLite**

Get= traer

Post=enviar