```
from flask import Flask, render_template,request,flash,redirect,url_for,session
from newsapi import NewsApiClient
import ibm_db
import requests
from flask_sqlalchemy import SQLAlchemy




app = Flask(__name__)
app.app_context().push()



app.config['DEBUG'] = True
app.config['SQLALCHEMY_DATABASE_URI'] = 'sqlite:///weather.db'


db = SQLAlchemy(app)


class City(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    name = db.Column(db.String(50), nullable=False)


app.secret_key='a'


connection = ibm_db.connect("DATABASE=bludb ; HOSTNAME=54a2f15b-5c0f-46df-8954-
7e38e612c2bd.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud; PORT= 32733; SECURITY=SSL;
```

```
SSLServerertificate=/Users/pradeep/PycharmProjects/Flask SQLite/certificate.crt; UID=ssr73017;
PWD=zQUhCA8HIIVl8nxT;",'','' )



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



@app.route('/register', methods=['GET', 'POST'])
def register():
 if request.method == 'POST':
     try:
      username = request.form['username']
      mail = request.form['mail']
      contact = request.form['contact']
      password = request.form['password']
      query = "INSERT INTO USERS VALUES(?,?,?,?)"
      stmt = ibm_db.prepare(connection, query)
      ibm_db.bind_param(stmt, 1, username)
      ibm_db.bind_param(stmt, 2, mail)
      ibm_db.bind_param(stmt, 3, contact)
      ibm_db.bind_param(stmt, 4, password)
      ibm_db.execute(stmt)
      flash("Registered  Successfully","success")
     except:
        flash("Error in Insert Operation", "danger")
     finally:
```

```
        return redirect(url_for("index"))
        con.close()
 return render_template('register.html')




@app.route('/login', methods=['GET', 'POST'])
def login():
 global userid
 if request.method == "POST":
    username = request.form['username']
    password = request.form['password']
    query = "SELECT * FROM USERS where username=? and password=?"
    stmt = ibm_db.prepare(connection, query)
    ibm_db.bind_param(stmt, 1, username)
    ibm_db.bind_param(stmt, 2, password)
    ibm_db.execute(stmt)
    account = ibm_db.fetch_assoc(stmt)
    if account:
       session['Loggedin'] = True
       session['id'] = account['USERNAME']
       session['username'] = account['USERNAME']
       return redirect("/news")
    else:
       flash("Oops Username and Password Mismatch", "danger")
       return redirect(url_for("index"))
```

```
@app.route("/news")
def home():
    api_key = '25560f3cf53c433c9807c60595b373a6'

    newsapi = NewsApiClient(api_key=api_key)

    top_headlines = newsapi.get_top_headlines(sources="bbc-news")
    all_articles = newsapi.get_everything(sources="bbc-news")

    t_articles = top_headlines['articles']
    a_articles = all_articles['articles']

    news = []
    desc = []
    img = []
    p_date = []
    url = []

    for i in range(len(t_articles)):
        main_article = t_articles[i]

        news.append(main_article['title'])
        desc.append(main_article['description'])
        img.append(main_article['urlToImage'])
        p_date.append(main_article['publishedAt'])
        url.append(main_article['url'])

        contents = zip(news, desc, img, p_date, url)
```

```
    news_all = []
    desc_all = []
    img_all = []
    p_date_all = []
    url_all = []

    for j in range(len(a_articles)):
        main_all_articles = a_articles[j]

        news_all.append(main_all_articles['title'])
        desc_all.append(main_all_articles['description'])
        img_all.append(main_all_articles['urlToImage'])
        p_date_all.append(main_all_articles['publishedAt'])
        url_all.append(main_article['url'])

        all = zip(news_all, desc_all, img_all, p_date_all, url_all)

    return render_template('home.html', contents=contents, all=all)



@app.route('/weather', methods=['GET', 'POST'])
def weather():
    if request.method == 'POST':
        new_city = request.form.get('city')

        if new_city:
            new_city_obj = City(name=new_city)

            db.session.add(new_city_obj)
```

```
        db.session.commit()

    cities = City.query.all()

    url =
'http://api.openweathermap.org/data/2.5/weather?q={}&units=imperial&appid=2a78921b9bffcab5e47
1564c6a553385'

    weather_data = []

    for city in cities:
        r = requests.get(url.format(city.name)).json()

        weather = {
            'city': city.name,
            'temperature': r['main']['temp'],
            'description': r['weather'][0]['description'],
            'icon': r['weather'][0]['icon'],
        }

        weather_data.append(weather)

    return render_template('weather.html', weather_data=weather_data)




@app.route('/logout')
def logout():
    session.clear()
```

```
    return redirect(url_for("index"))


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