Create IBM DB2 and connect with python

```python
from flask import Flask, render_template, requests
from newsapi import NewsApiClient

# init flask app
app = Flask(__name__)

# Init news api
newsapi = NewsApiClient(api_key='73ebcf7998bd494c9ae7ef215cdb11fd')

# helper function
def get_sources_and_domains():
        all_sources = newsapi.get_sources()['sources']
        sources = []
        domains = []
        for e in all_sources:
                id = e['id']
                domain = e['url'].replace("http://", "")
                domain = domain.replace("https://", "")
                domain = domain.replace("www.", "")
                slash = domain.find('/')
                if slash != -1:
                        domain = domain[:slash]
                sources.append(id)
                domains.append(domain)
        sources = ", ".join(sources)
        domains = ", ".join(domains)
        return sources, domains

@app.route("/", methods=['GET', 'POST'])
def home():
        if request.method == "POST":
                sources, domains = get_sources_and_domains()
                keyword = request.form["keyword"]
                related_news = newsapi.get_everything(q=keyword,
                                                        sources=sources,
                                                        domains=domains,
                                                        language='en',
                                                        sort_by='relevancy')
                no_of_articles = related_news['totalResults']
                if no_of_articles > 100:
                        no_of_articles = 100
                all_articles = newsapi.get_everything(q=keyword,
                                                        sources=sources,
                                                        domains=domains,
                                                        language='en',
                                                        sort_by='relevancy',
                                                        page_size =
no_of_articles)['articles']
                return render_template("home.html", all_articles = all_articles,
                                                keyword=keyword)
        else:
                top_headlines = newsapi.get_top_headlines(country="in", language="en")
                total_results = top_headlines['totalResults']
                if total_results > 100:
```

```python
            total_results = 100
        all_headlines = newsapi.get_top_headlines(country="in",

    language="en",

    page_size=total_results)['articles']
            return render_template("home.html", all_headlines = all_headlines)
        return render_template("home.html")

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