TASK 7

SUBJECT:

Programming For AI

PROGRAM:

BS DATA SCIENCE

SUBMITTED TO:

Sir Rasikh Ali

SUBMITTED BY:

FIZZA FAROOQ

ROLL NUMBER:

SU92-BSDSM-F23-017

BSDS (4A)

TASK 7

Flask News App
Backend code (python / flask code)

```python
from flask import Flask, render_template, jsonify
import requests

app = Flask(__name__)
API_KEY = "900b473053ca4dd6ab4fa228afa0a050"
def get_news():
    url = f"https://newsapi.org/v2/top-headlines?country=us&apiKey={API_KEY}"
    response = requests.get(url)
    data = response.json()
    return data.get("articles", [])

@app.route('/news', methods=['GET'])
def news():
    articles = get_news()
    return jsonify(articles)

@app.route('/')
def home():
    articles = get_news()
    return render_template('index.html', articles=articles)

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

Overview

This Flask-based News App fetches the latest top headlines from the NewsAPI and displays them in a web interface. It provides both an API endpoint and a front-end web page to view the news articles.

Technologies Used:

- Flask (Python Web Framework)
- HTML & CSS (Bootstrap for Styling)
- NewsAPI (for fetching live news)

<u>Code Implementation</u>

1.app.py (Main Flask Application)

- **Flask:** A lightweight web framework used to create the web application.
- **render_template:** Used to render HTML templates (e.g., index.html).
- **jsonify:** Converts Python data structures (lists/dictionaries) into JSON format for API responses.
- **requests:** A module for making HTTP requests to external APIs (in this case, NewsAPI).

2. Initializing Flask Application

- **app = Flask(__name__):** Creates a new Flask web application instance.
- **API_KEY:** The unique API key required to fetch data from the NewsAPI.

  "900b473053ca4dd6ab4fa228afa0a050"

3. Fetching News Data from NewsAPI

- Constructs the API request URL using the API key.
- Sends a GET request to fetch top news headlines from the US.
- Parses the JSON response.
- Extracts the "articles" field (which contains the news articles) and returns it as a list.

4. API Endpoint for News Data

- Defines a Flask route /news that handles GET requests.
- Calls **get_news()** to fetch the latest news articles.
- Returns the articles as a JSON response, making this a REST API endpoint.

5. Rendering the Home Page

- Defines a route / for the home page.
- Calls **get_news()** to fetch the latest news articles.
- Passes the articles to **index.html**, which displays them.

6. Running the Application

- Ensures the Flask app runs only when executed directly (not when imported as a module).

- debug=True: Enables debug mode, which helps in troubleshooting errors during development.