Problem Statement

- **Objective**: To develop a machine learning model that converts natural language questions into SQL queries, enabling users to easily interact with databases without needing to understand SQL syntax. This model aims to simplify data retrieval processes by allowing users to express their queries in everyday language.

Summary

- The application uses a combination of SQLite for database management and a Generative AI model to convert natural language questions into SQL queries. Users can input questions about student data, and the application retrieves relevant records from the SQLite database.

Tools and Technologies Used

- **SQLite**: A lightweight database engine used to store and manage student data.
- **Streamlit**: A web application framework that allows for the easy creation of interactive web applications in Python.
- **Google Generative AI (Gemini)**: A language model utilized to convert natural language questions into SQL queries.
- **Python**: The programming language used for developing the application.
- **dotenv**: A library for loading environment variables from a .env file to securely manage API keys.

Working

- The application performs the following tasks:
    1. **Database Setup**:
        - Establishes a connection to a SQLite database named student.db.
        - Creates a table named student with columns for student name, class, section, and marks if it does not already exist.
        - Inserts sample student records into the database while handling potential duplicates.
    2. **User Input**:
        - Uses Streamlit to create a user interface where users can input questions related to student data.
    3. **Query Generation**:
        - Sends the user's input to the Google Generative AI model to convert it into a corresponding SQL query.
    4. **Data Retrieval**:
        - Executes the generated SQL query on the SQLite database and retrieves the relevant records.
    5. **Display Results**:

        - Outputs the results of the SQL query back to the user through the Streamlit interface.

Key Tasks

- **Database Management**:
    - Create and manage an SQLite database to store student data.
    - Handle record insertion and avoid duplication errors.
- **Natural Language Processing**:
    - Implement a Generative AI model to translate natural language questions into SQL queries.
- **Web Application Development**:
    - Develop a user-friendly interface using Streamlit for interaction with the application.
- **Data Query Execution**:
    - Execute SQL queries and fetch results from the SQLite database.
- **Error Handling**:
    - Implement error handling for database operations to manage potential issues gracefully.

VAISHNAVI SONI

BCA AI-DS

2ND YEAR