

Explanation of Task 2: Working with databases

I have used Django Web Framework to implement Task 2. Here, I used the default DBMS in Django, which is sqlite3. I have created a model named *Student* that represents the structure of the "Student" table in the database. Each field in the model corresponds to a column in the table. This model is used to create, read, update, and delete student records in the database using Django's Object-Relational Mapping (ORM). I have written a Python script in views.py to interact with it. The script is capable of adding, retrieving, updating, and deleting data from the Student table.

The database structure in my Django project consists of a single table called "Student" with fields for student information such as name, roll number, address, and mobile number. This structure was chosen for its simplicity, flexibility, and alignment with Django's conventions. It allows for easy data manipulation and maintenance, making it suitable for small to medium-sized applications. However, for more complex scenarios, additional tables and relationships may be necessary.