Task 1

DATABASE CREATION AND DATA INSERTION

```
CREATE DATABASE STUDENTMANAGEMENT;
 2 •
      USE STUDENTMANAGEMENT;
 3 • ⊖ CREATE TABLE STUDENTS(
 4
       StudentID INT AUTO INCREMENT PRIMARY KEY.
 5
       Name VARCHAR(50) NOT NULL,
      Gender VARCHAR(1) CHECK (Gender IN ('M', 'F')),
      Age INT CHECK (Age >= 5 AND Age <= 25),
 7
      Grade VARCHAR(10) NOT NULL,
 8
 9
      MathScore INT CHECK (MathScore BETWEEN 0 AND 100),
10
       ScienceScore INT CHECK (ScienceScore BETWEEN 0 AND 100),
       EnglishScore INT CHECK (EnglishScore BETWEEN 0 AND 100)
11
12
      ٠);
13 • INSERT INTO Students (Name, Gender, Age, Grade, MathScore, ScienceScore, EnglishScore)
14
       ('GEETA', 'F', 16, 'A', 95, 90, 88),
15
       ('SHYAM', 'M', 17, 'B', 80, 85, 82),
16
       ('RAM', 'M', 15, 'A', 92, 87, 91),
       ('ANUSHKA', 'F', 18, 'C', 70, 75, 78),
18
      ('GYAN', 'M', 16, 'B', 85, 89, 84),
19
20
      ('RIYA', 'F', 17, 'A', 96, 94, 93),
       ('MUKESH', 'M', 15, 'C', 68, 70, 65),
21
       ('SITA', 'F', 16, 'B', 83, 81, 85),
22
       ('ISAAN', 'M', 17, 'A', 98, 97, 96),
23
       ('JACKY', 'M', 18, 'C', 72, 74, 71);
```

PURPOSE OF DATABASE CREATION- This query creates a database named **StudentManagement** and a table **Students** with the necessary fields to store student information, including scores in three subjects.

PURPOSE OF DATA INSERTION: This query inserts 10 sample records representing students with diverse grades, genders, and scores.



```
-- 1. Display all students and their details to get an overview of the data--
  25
         SELECT* FROM STUDENTS;
         CELECT
| Edit: 🚄 📆 Export/Import: 🏣 👸 | Wrap Cell Content:
    StudentID Name
                        Gender
                                           MathScore ScienceScore EnglishScore
                                     Grade
             GEETA
                                            95
                                                      90
                                                                  88
                               16
   1
             SHYAM
                               17
                                                                  82
                       M
                                     В
                                            80
                                                      85
   3
             RAM
                                            92
                                                      87
                                                                  91
                               15
             ANUSHKA
                       F
                              18
                                    C
                                            70
                                                      75
                                                                  78
   5
             GYAN
                       M
                                                      89
                                                                  84
                               16
                                     В
                                            85
             RIYA
                               17
                                            96
                                                      94
                                                                  93
   6
                                    A
             MUKESH
                               15
                                                      70
                                                                  65
                                            68
   8
             SITA
                               16
                                     В
                                            83
                                                      81
                                                                  85
             ISAAN
                       M
                               17
                                            98
                                                      97
                                                                  96
   10
             JACKY
                       M
                                                                  71
NULL
                               18
                                    C
                                           72
NULL
```

Purpose:

gu

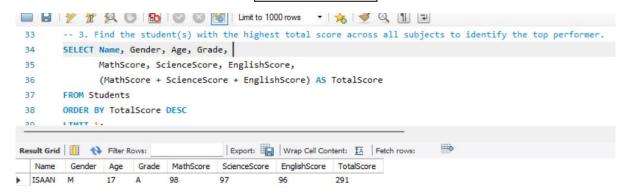
• It Retrieves all student records for an overview of the dataset.

Query No-2

```
27
        -- 2. Calculate the average scores for each subject to understand subject-wise performance
 28 •
        SELECT
 29
            AVG(MathScore) AS Avg_MathScore,
            AVG(ScienceScore) AS Avg_ScienceScore,
 30
 31
            AVG(EnglishScore) AS Avg_EnglishScore
 32
        FROM Students;
         SELECT Name Gondon Age Goods
                                       Export: Wrap Cell Content: $\frac{1}{2}
Avg_MathScore Avg_ScienceScore Avg_EnglishScore
83.9000
                84,2000
                              83,3000
```

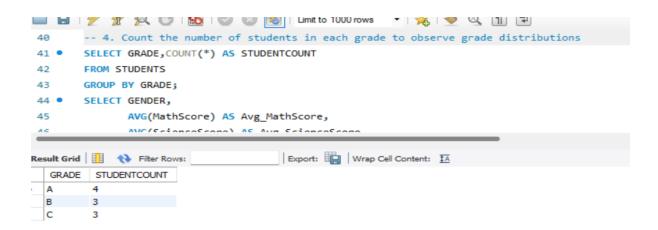
Purpose: It Provides insights into subject-wise performance and can identify which subject students excel in or struggle with.

Query No-3



Purpose: It Identifies the student with the highest total score and helps in recognizing academic excellence.

Query No-4



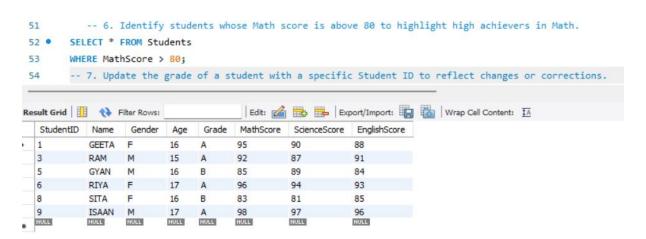
Purpose: It Helps analyze the distribution of students across different grade and Identifies whether most students belong to a specific grade.



```
44
       -- 5. Find the average score for male and female students to compare performance by gender.
45 •
       SELECT GENDER,
46
              AVG(MathScore) AS Avg_MathScore,
47
              AVG(ScienceScore) AS Avg_ScienceScore,
48
              AVG(EnglishScore) AS Avg_EnglishScore
49
              FROM STUDENTS
50
              GROUP BY GENDER;
esult Grid 🔢 💎 Filter Rows:
                                       Export: Wrap Cell Content: 🖽
GRADE STUDENTCOUNT
 В
        3
C
```

Purpose: It Compares male and female students' academic performance and Can highlight potential gender disparities in subject performance.





Purpose: It Lists students who scored above 80 in Math and helps in identifying strong performers in Mathematics.

Query No-7

```
-- 7. Update the grade of a student with a specific Student ID to reflect changes or corrections.
     UPDATE Students
      SET Grade = '8
       WHERE StudentID = 5;
                                                                                                                                                                                             Context Help Snippets
Action Output
 10 14:09:40 SELECT AVG(MathScore) AS Avg_MathScore, AVG(ScienceScore) AS Avg_ScienceScore, AVG(E... 1 row(s) returned
                                                                                                                                                                                                              0.000 sec / 0.000 se
 11 14:11:26 SELECT Name, Gender, Age, Grade,
                                                 MathScore, ScienceScore, EnglishScore,
                                                                                                                                                                                                              0.000 sec / 0.000 se
                                                                                                              1 row(s) returned
 12 14:12:41 SELECT GRADE, COUNT(*) AS STUDENTCOUNT FROM STUDENTS GROUP BY GRADE LIMIT 0, 1000
                                                                                                                                                                                                              0.031 sec / 0.000 se
 13 14:16:52 SELECT * FROM Students WHERE Math Score > 80 LIMIT 0, 1000
                                                                                                              6 row(s) returned
                                                                                                                                                                                                              0.000 sec / 0.000 se
 14 14:17:21 SELECT * FROM Students WHERE Math Score > 80 LIMIT 0, 1000
                                                                                                              6 row(s) returned
                                                                                                                                                                                                              0.000 sec / 0.000 se
 15 14:19:10 UPDATE Students SET Grade = 'B' WHERE StudentID = 5
                                                                                                             0 \text{ row(s)} affected Rows matched: 1 \text{ Changed: } 0 \text{ Warnings: } 0
```

Purpose: Updates a student's grade based on StudentID and Useful for correcting data entry mistakes or reflecting academic improvements.