Name: Manju

Date: 06/07/2025

Batch: May 25th Batch

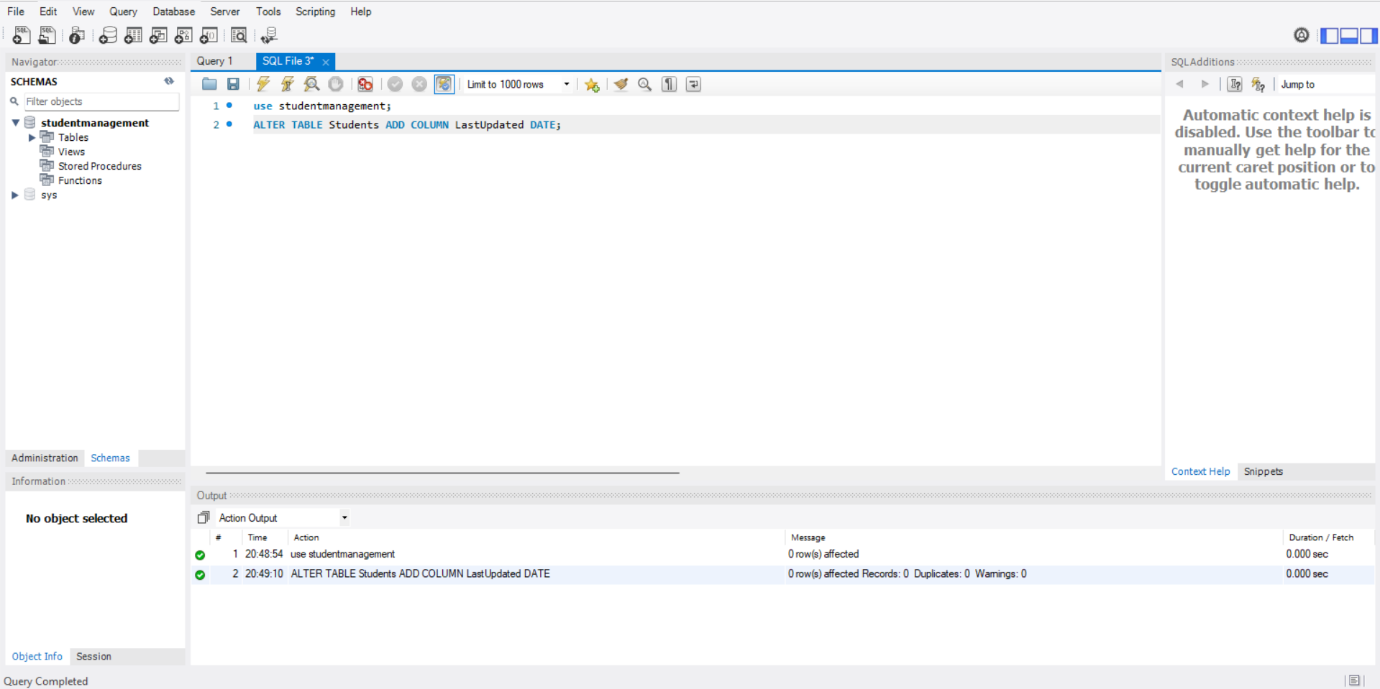
This document contains a set of SQL queries completed for the Week 6 task.

In this task, I worked on modifying the Students table by adding new columns to track when records were last updated and to indicate whether each record is active or inactive. I also updated these columns with appropriate values and removed records based on certain conditions, such as inactive status or outdated information.

Each query is explained in simple terms to describe what it does and why it was used. I’ve included screenshots showing the queries as they were run and the results they produced, following the assignment requirements.

**1.Modify Table Structure**

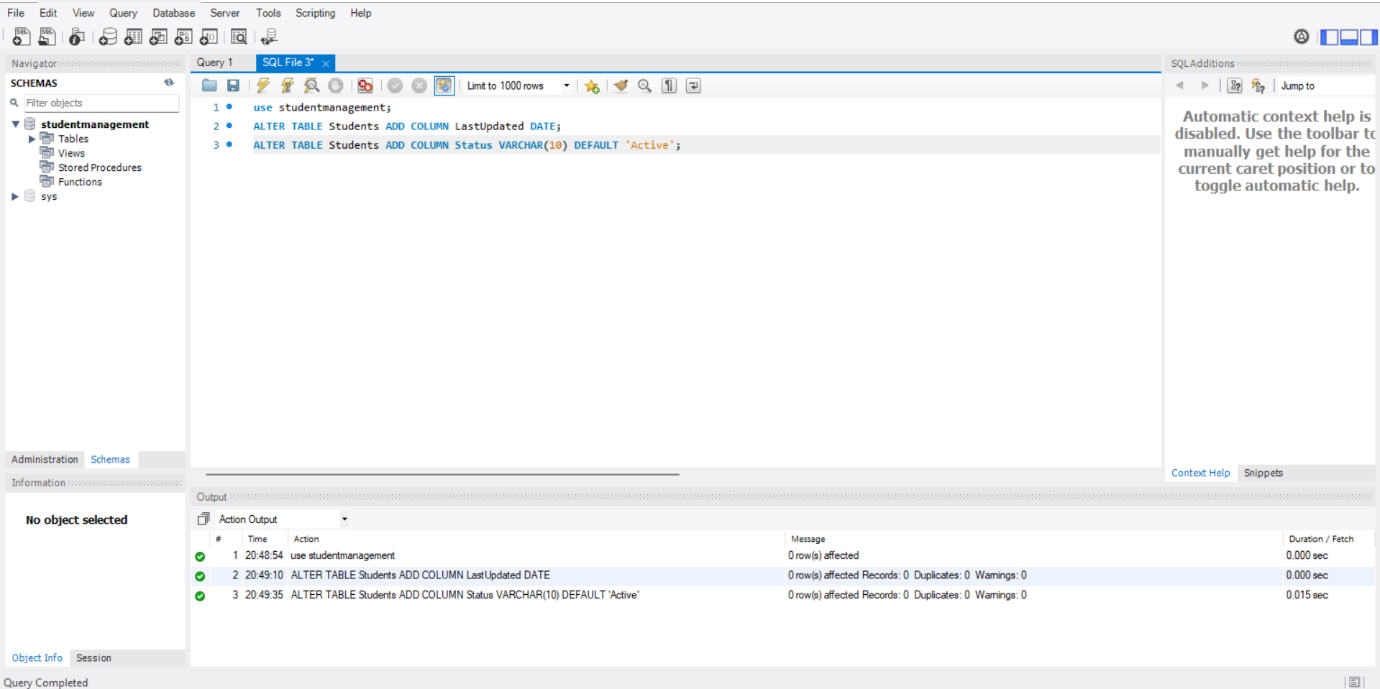
ALTER TABLE Students ADD COLUMN LastUpdated DATE;

**Output:**

**Purpose:** To add a new column named LastUpdated to the Students table.

**Explanation:** The LastUpdated column will store the date when a student record was last modified, enabling you to track updates and manage data retention.

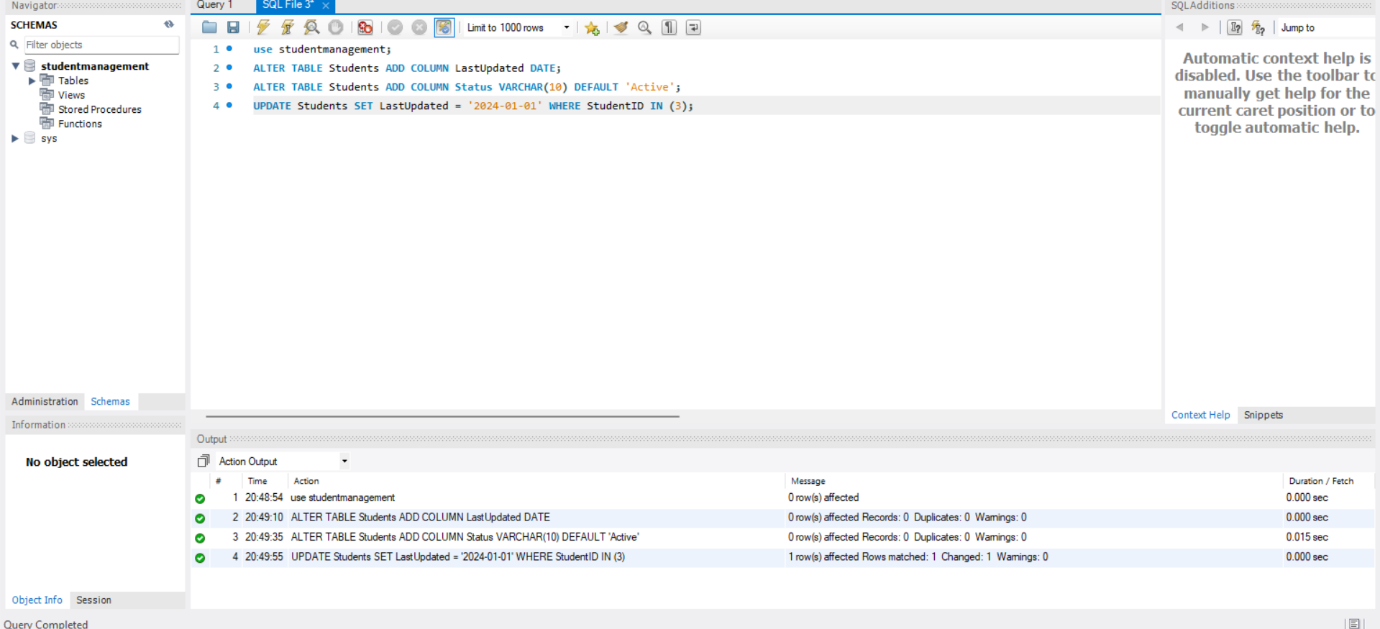
ALTER TABLE Students ADD COLUMN Status VARCHAR(10) DEFAULT 'Active';

**Output:**

**Purpose:** To add a Status column that defaults to 'Active'.

**Explanation:** The Status column is used to categorize records as 'Active' or 'Inactive', making it easier to filter and delete records based on their status.

UPDATE Students SET LastUpdated = '2024-01-01' WHERE StudentID IN (3);

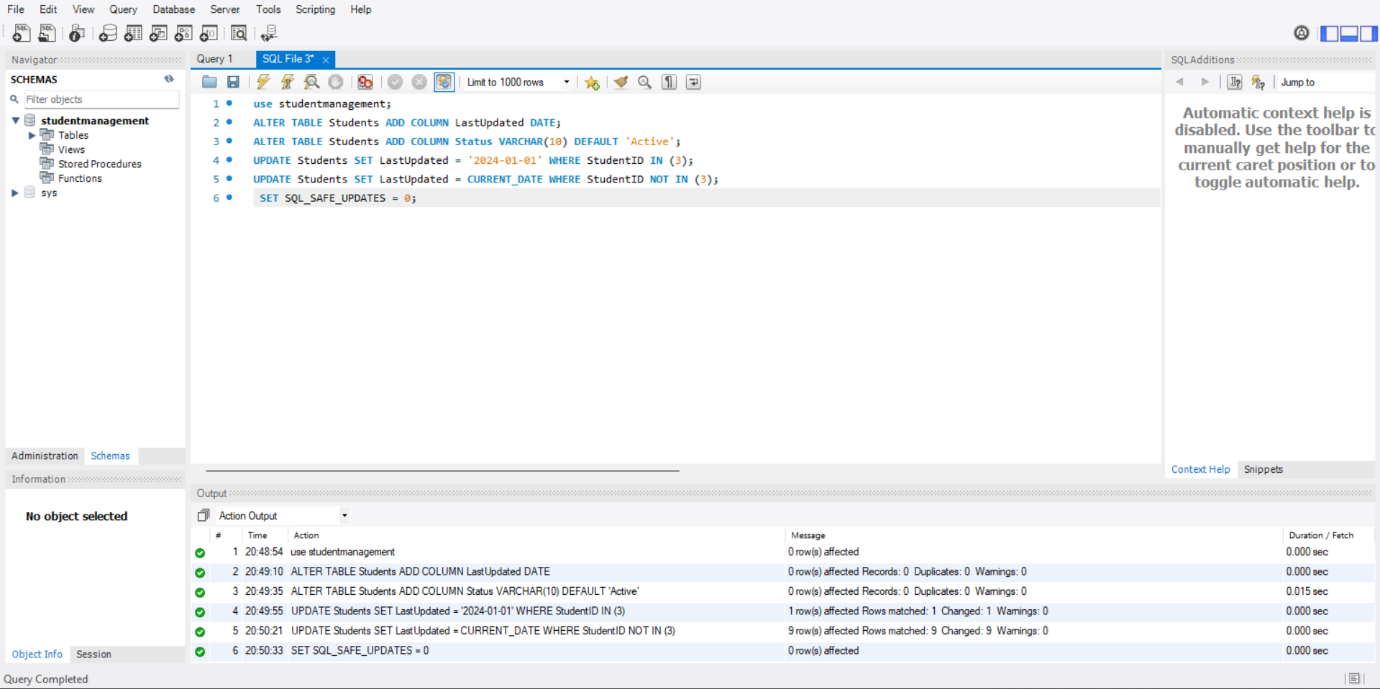
**Output:**

**Purpose:** To assign a specific date (2024-01-01) to the LastUpdated field for a particular student (StudentID = 3).

**Explanation:** Explanation:This simulates older data for StudentID 3, helping you later test deletion of records based on outdated timestamps.

UPDATE Students SET LastUpdated = CURRENT\_DATE WHERE StudentID NOT IN (3);

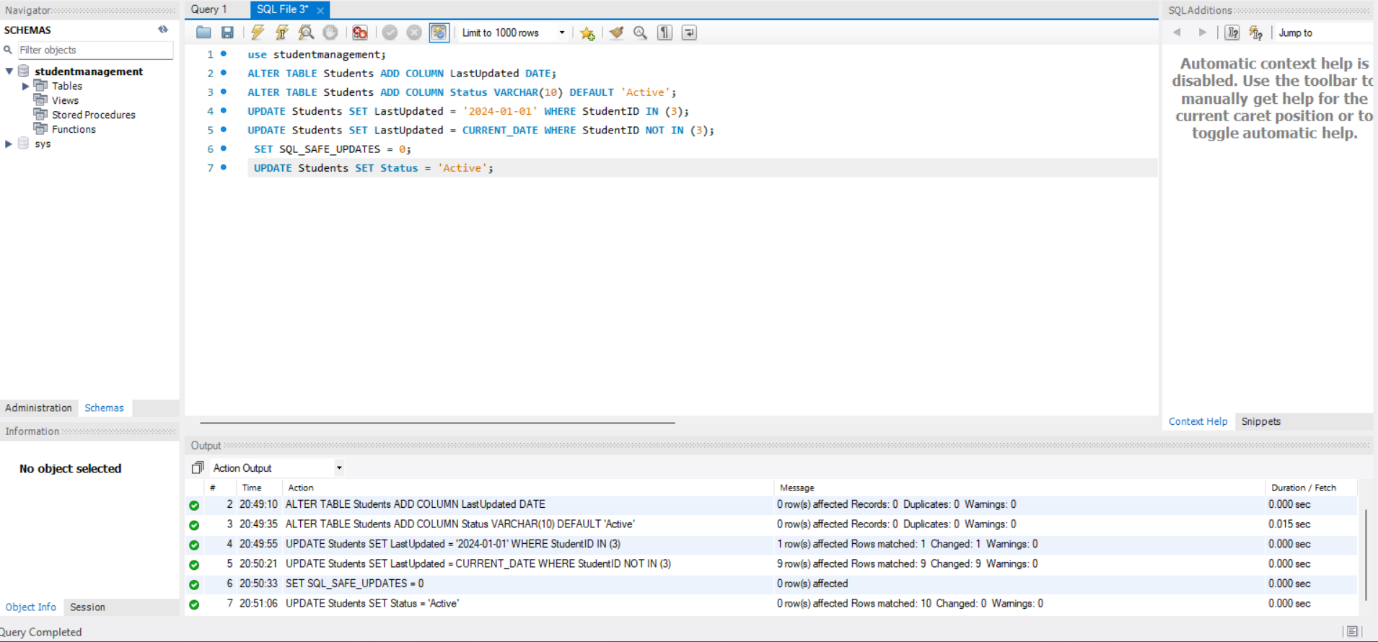
**Output:**

****

**Purpose:** To set the LastUpdated field to today’s date for all students except StudentID 3.

**Explanation:** This keeps most records current, while retaining an older date for StudentID 3 to test date-based filtering and deletion.

UPDATE Students SET Status = 'Active';

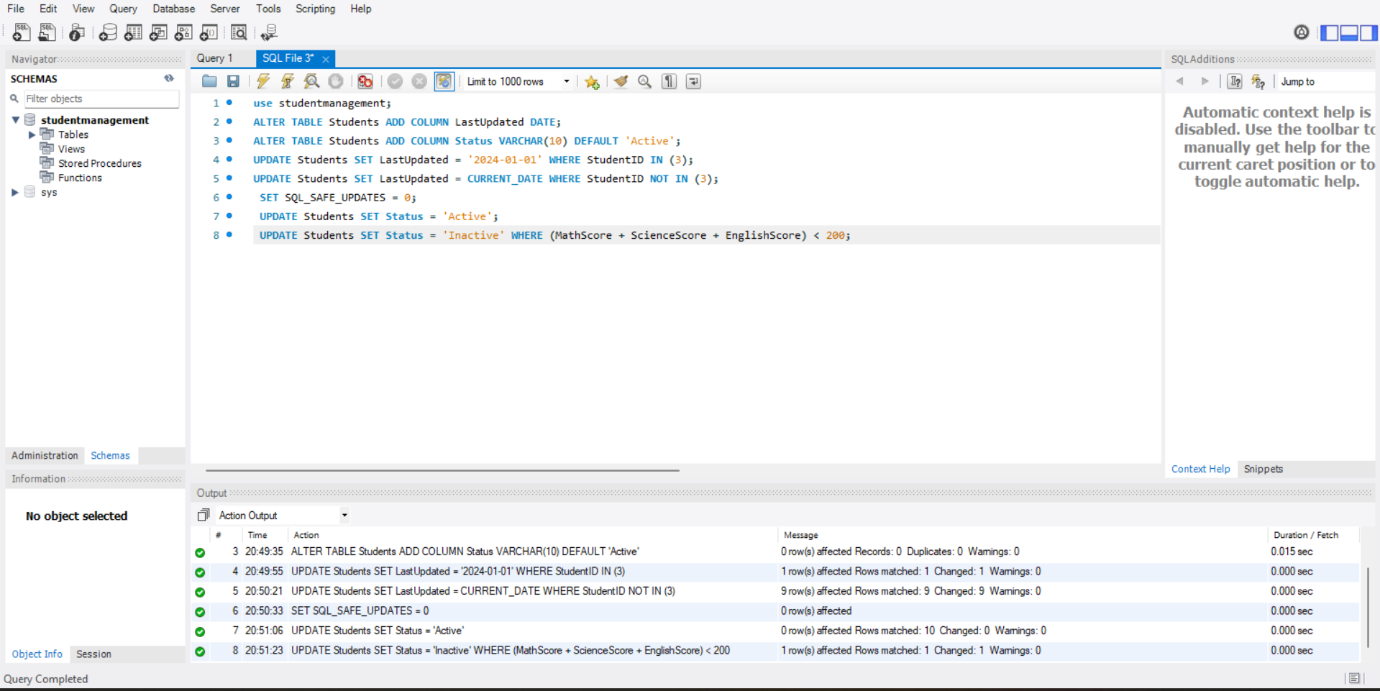
**Output:**

**Purpose:** To reset the Status column to 'Active' for all students.

**Explanation:** This ensures that before marking inactive records, all statuses are consistently set to 'Active'.

UPDATE Students SET Status = 'Inactive' WHERE (MathScore + ScienceScore + EnglishScore) < 200;

**Output:**

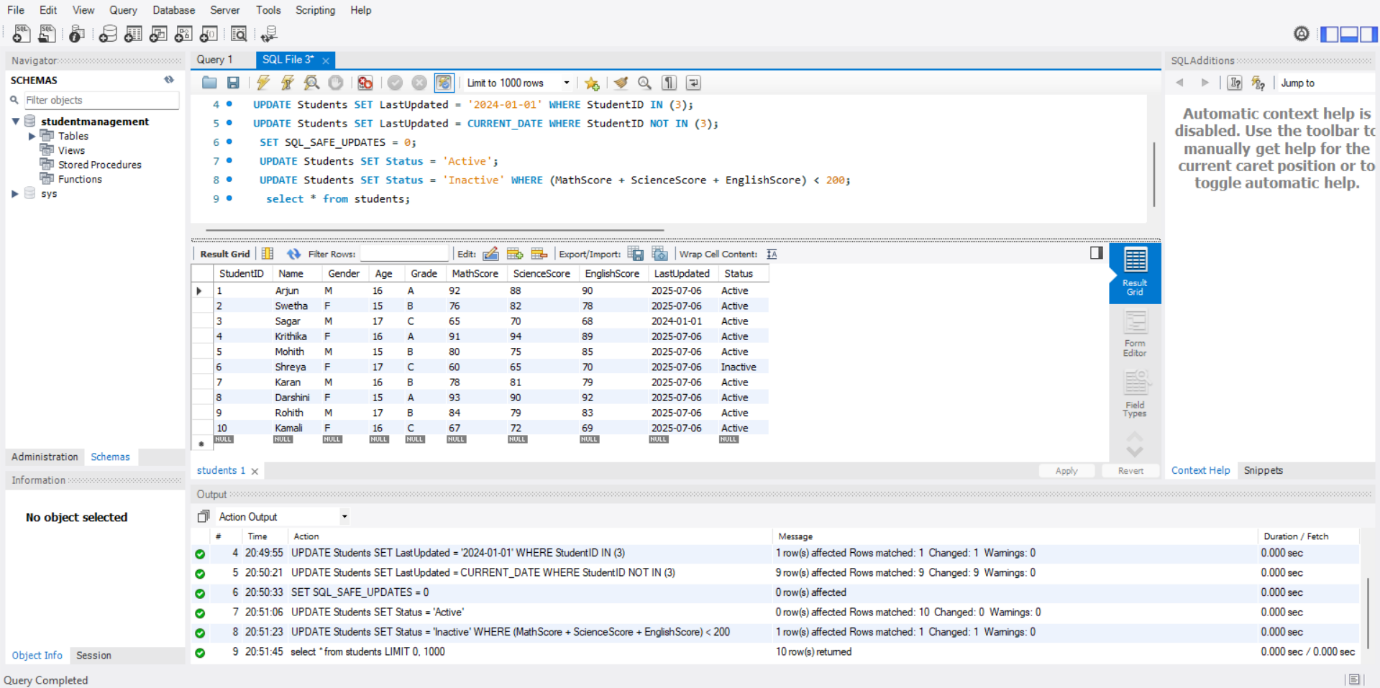
****

**Purpose:** To mark students as 'Inactive' if their total score across subjects is below 200.

**Explanation:** This provides a business rule to identify low-performing students and prepare their records for potential deletion.

Select \* from students;

**Output:**

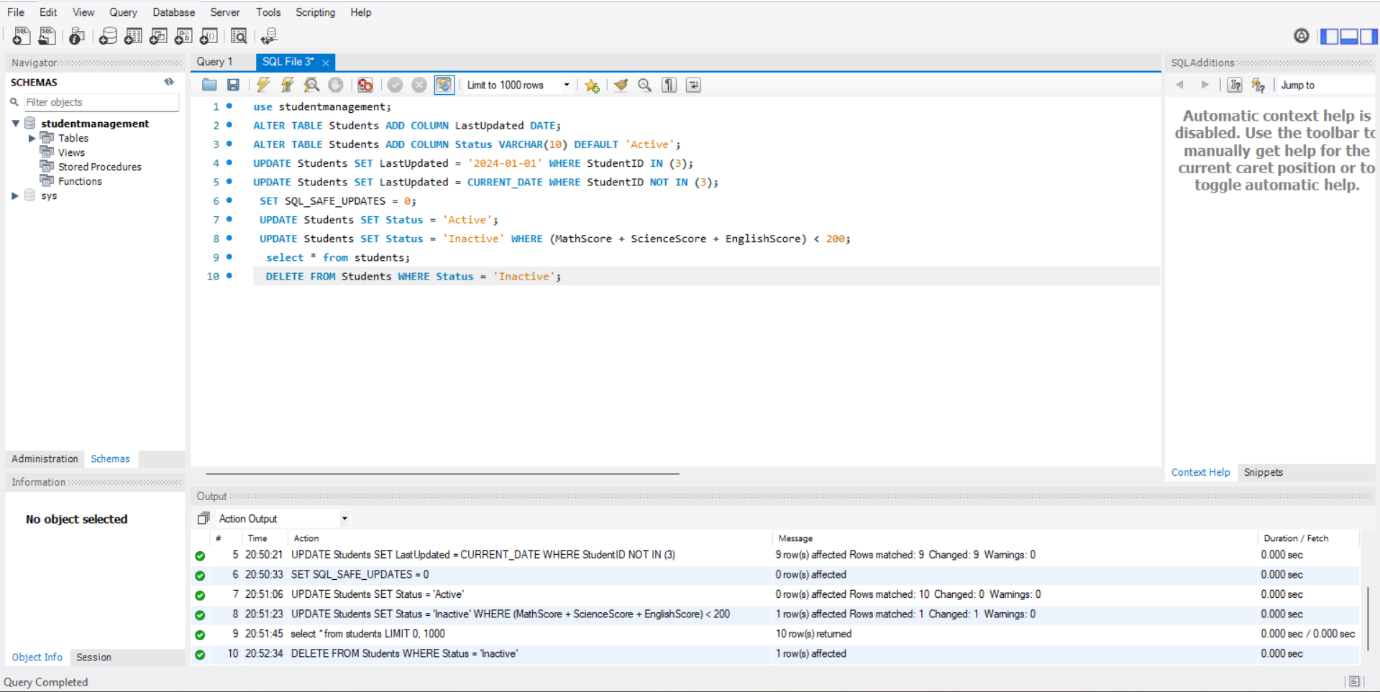
****

**Purpose:** To display all records in the Students table.

**Explanation:** This allows you to review and confirm which students have been marked 'Inactive' before deletion.

**2.Deleting recoreds based on condition:**

DELETE FROM Students WHERE Status = 'Inactive';

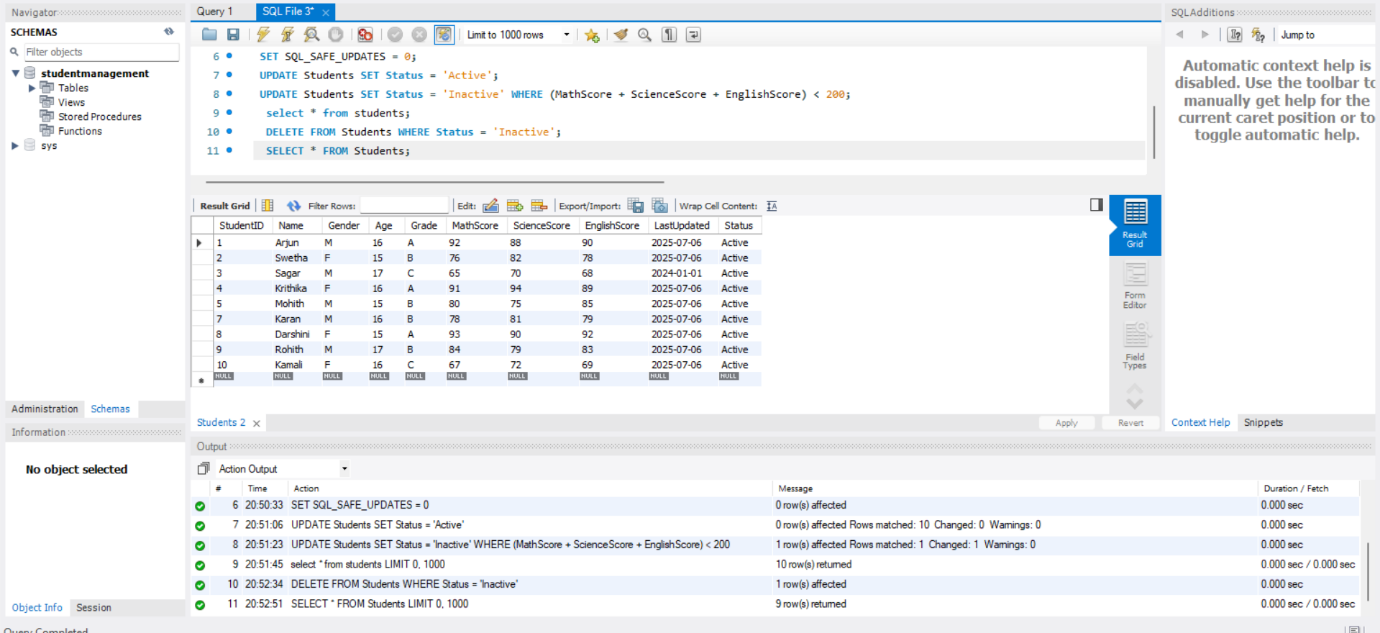
** Output:**

**Purpose:** To remove all students who were marked 'Inactive'.

**Explanation:** This deletes low-performing student records as identified in the previous step.

select \* from students;

**Output:**

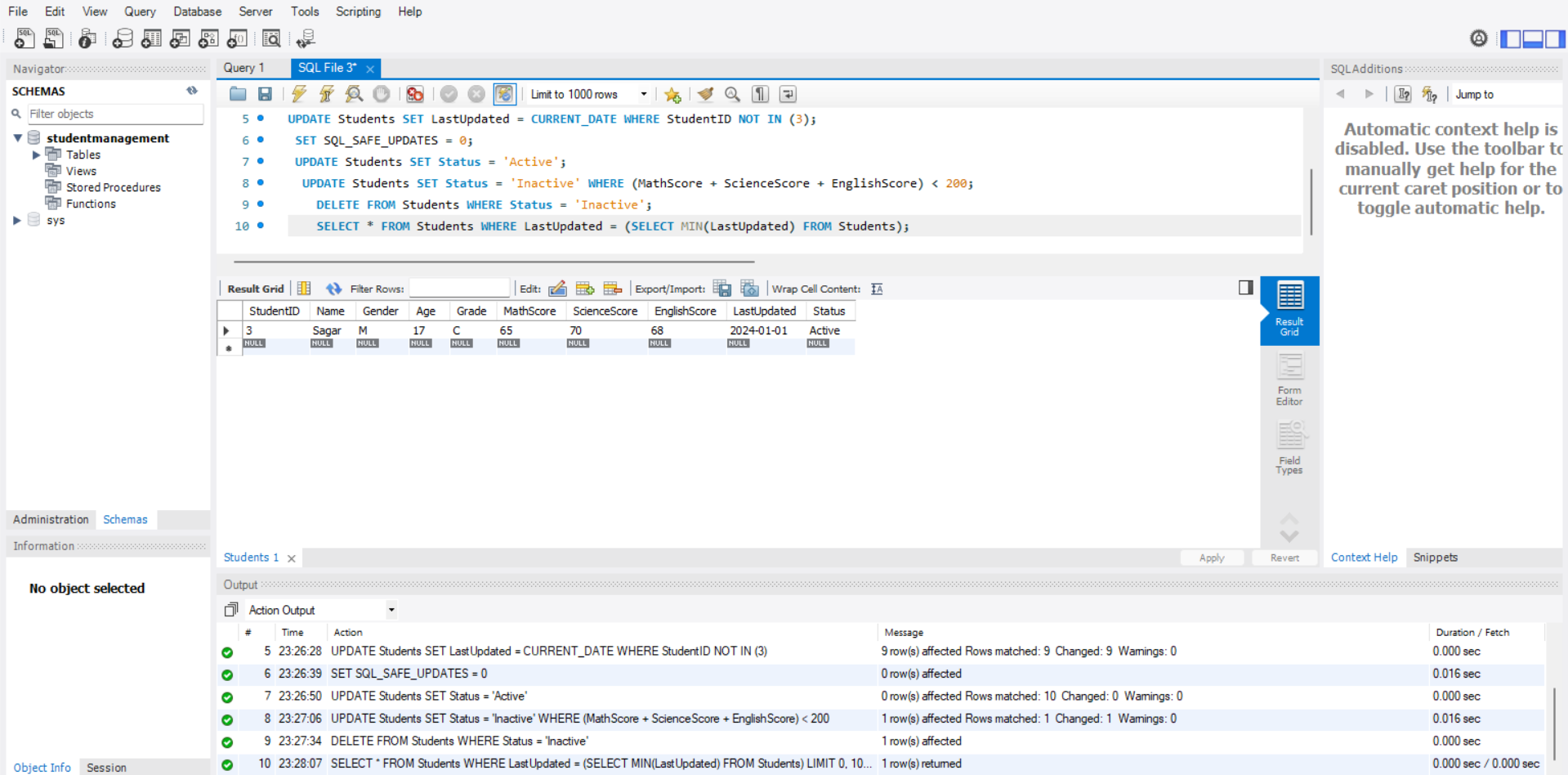
****

**Purpose:** To review the Students table after deletion.

**Explanation:** This confirms that only the intended records were deleted and others remain intact.

SELECT \* FROM Students WHERE LastUpdated = (SELECT MIN(LastUpdated) FROM Students);

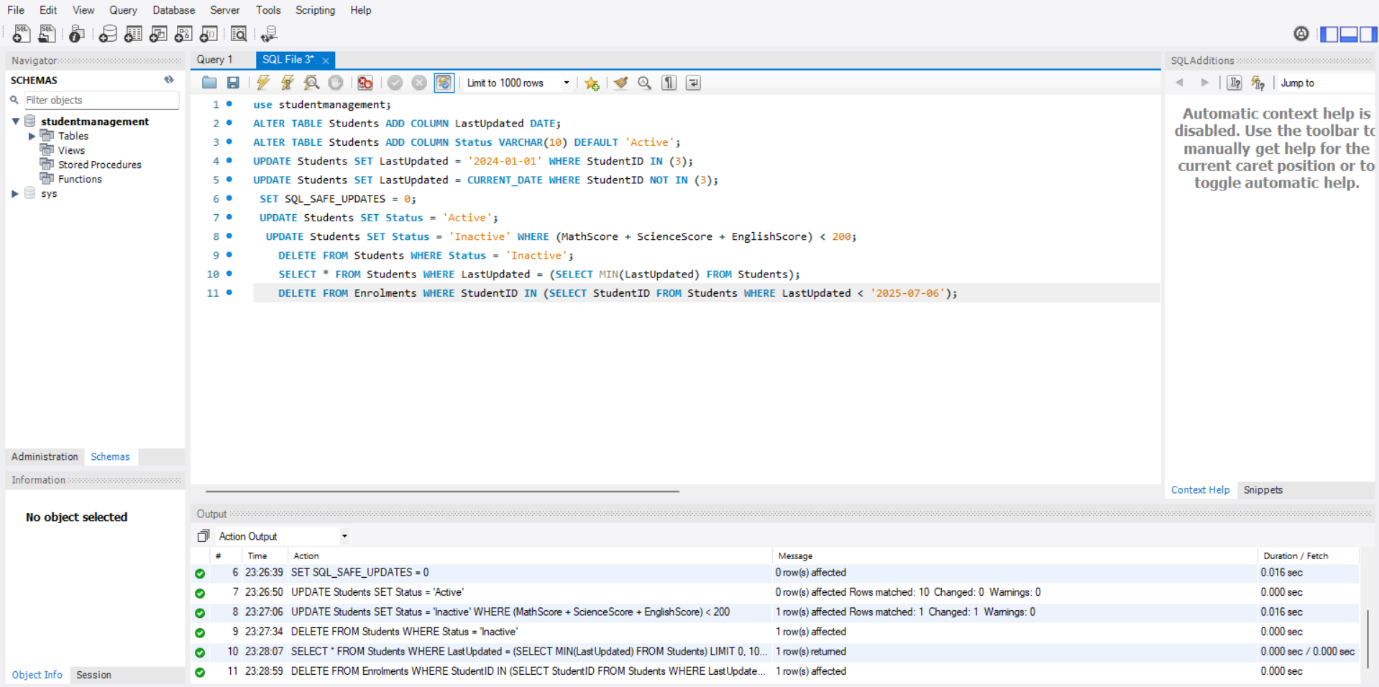
**Output:**

****

**Purpose:** To find the record(s) with the oldest LastUpdated date.

**Explanation:** This helps identify which records are the oldest in the table, potentially for further review or deletion.

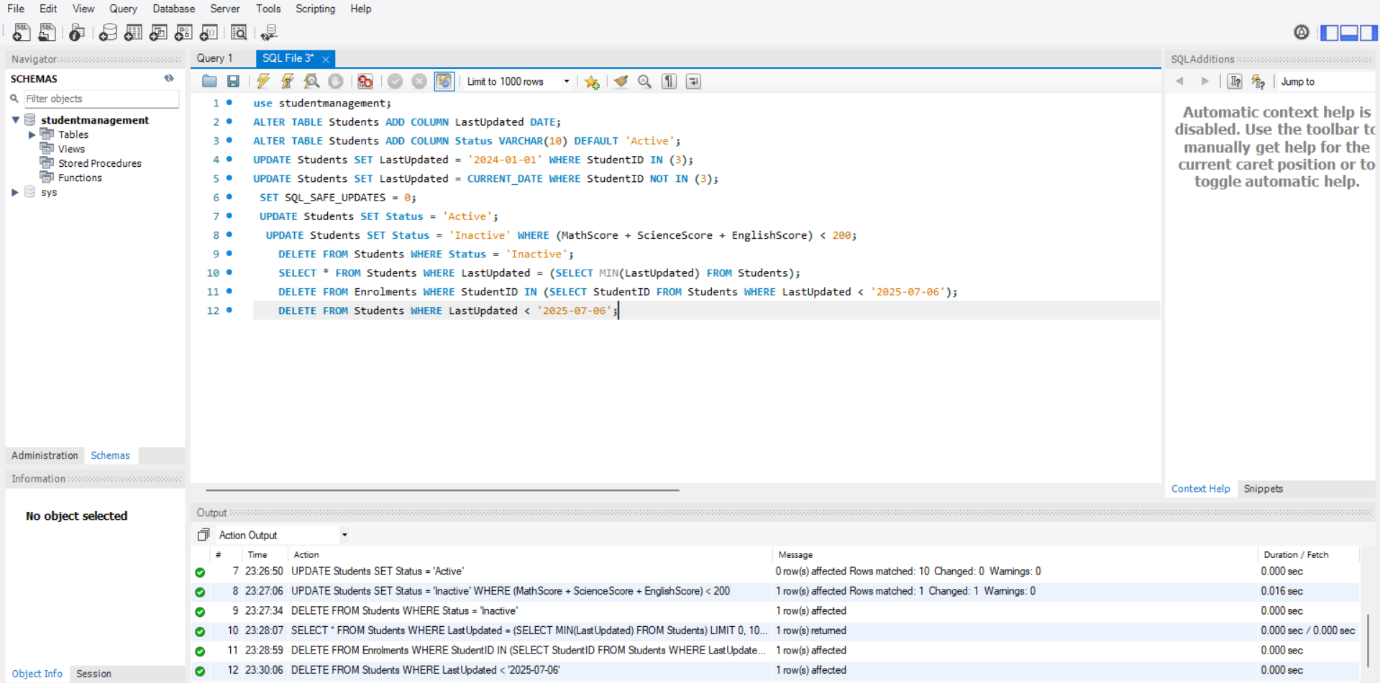
DELETE FROM Enrolments WHERE StudentID IN (SELECT StudentID FROM Students WHERE LastUpdated < '2025-07-06');

**Output:**

**Purpose:** To delete enrolment records linked to students with outdated records.

**Explanation:** This ensures referential consistency—removing enrolments tied to students you plan to delete.

DELETE FROM Students WHERE LastUpdated < '2025-07-06';

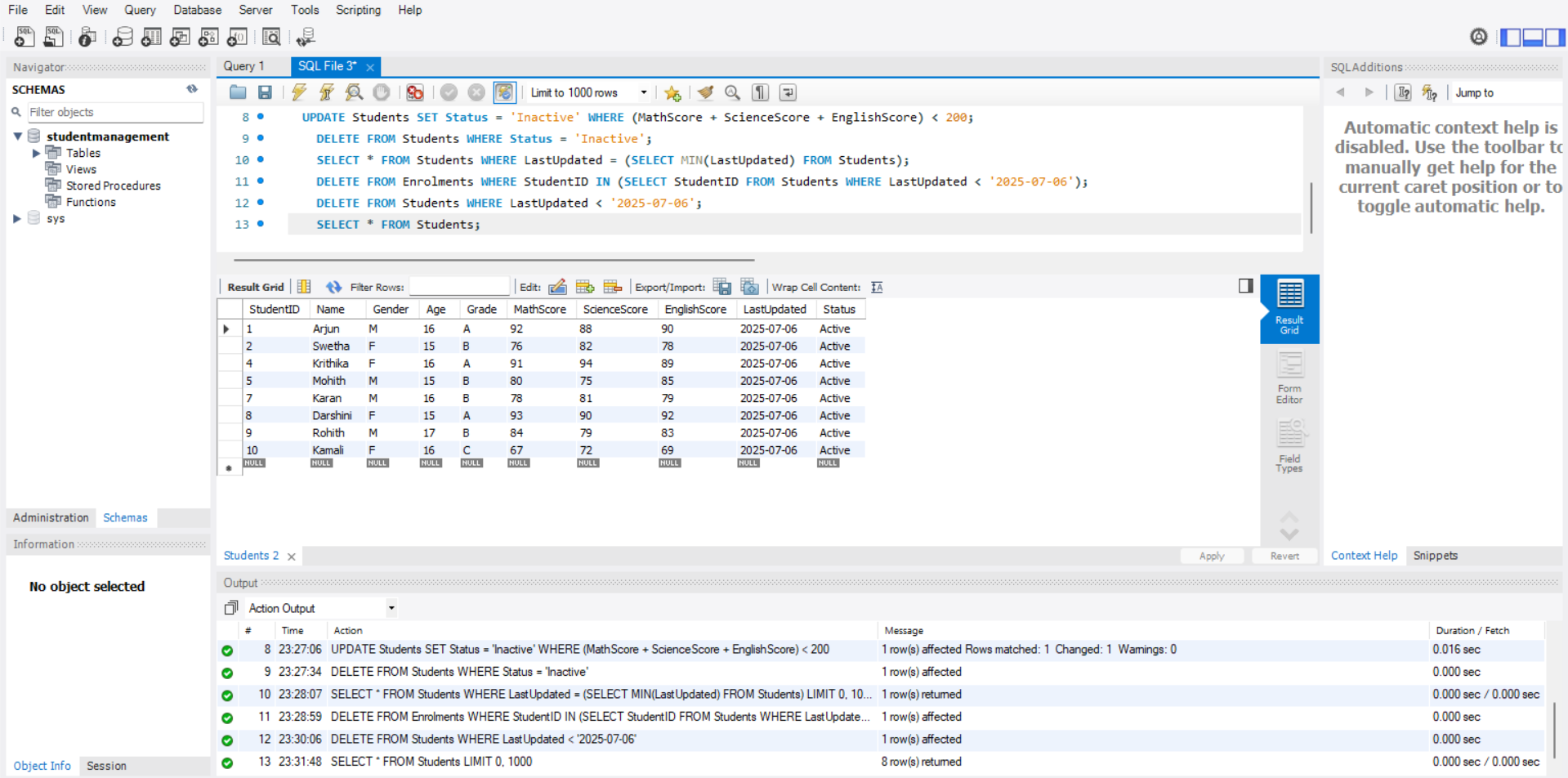
**Output:**

**Purpose:** To delete all student records older than a specified date regardless of enrolment status.

**Explanation:** This performs a final cleanup of outdated records, ensuring only recent data remains.

SELECT \* FROM Students;

**Output:**

****

**Purpose:** To show the final state of the Students table.

**Explanation:** This final check ensures that all deletions were applied as expected and no unintended records remain.

**Summary and Keyfindings**

**Summary & Key Findings:**

1. **Two new columns** were added to the Students table:

LastUpdated (a date field to track the last update of student records).

Status (a text field with a default value 'Active' to indicate if a student is currently active or not).

1. **Initial Update**:

Only the student with StudentID = 3 was marked with a fixed update date (2024-01-01).

All **other students** were marked with the **current date** as LastUpdated.

1. **Safe update mode was disabled** (SET SQL\_SAFE\_UPDATES = 0) to allow broader update operations.
2. **Status Updates**:

Initially, **all students** were marked as 'Active'.

Then, students whose **total marks (Math + Science + English) were less than 200** were marked as 'Inactive'.

1. **Viewing and Cleaning**:

After checking all student data, students marked as 'Inactive' were **deleted** from the database.

Then, the database was **viewed again** to confirm the changes.

1. **Oldest Record Check**:

The student(s) with the **oldest LastUpdated date** was retrieved.

1. **Further Cleanup**:

Any **enrolment records** linked to students who were **last updated before July 6, 2025**, were deleted.

The same condition was used to **delete those students** as well.

1. **Final Result**:

The final list of students left in the database was shown using SELECT \* FROM Students.