**Student Management System Project Report**

**1. SQL Queries**

**Task 1: List All Students and the Courses They Are Enrolled In**

SELECT

s.Name AS StudentName,

c.course\_name AS CourseName

FROM

Students s

INNER JOIN

Enrolments e ON s.StudentID = e.student\_id

INNER JOIN

Courses c ON e.course\_id = c.course\_id;

* **Purpose**: Combines the Students, Enrolments, and Courses tables using INNER JOIN to list all students and the courses they are enrolled in.
* **Logic**: The INNER JOIN ensures that only students with enrolled courses are included in the result.

**Task 2: Find the Number of Students Enrolled in Each Course**

SELECT

c.course\_name AS CourseName,

COUNT(e.student\_id) AS NumberOfStudents

FROM

Courses c

LEFT JOIN

Enrolments e ON c.course\_id = e.course\_id

GROUP BY

c.course\_id, c.course\_name;

* **Purpose**: Uses a LEFT JOIN to include all courses, even those with no enrolments, and counts the number of students enrolled in each course.
* **Logic**: The LEFT JOIN ensures that all courses are included, and COUNT(e.student\_id) calculates the number of enrolments.

**Task 3: List Students Who Have Enrolled in More Than One Course**

SELECT

s.Name AS StudentName,

COUNT(e.course\_id) AS NumberOfCourses

FROM

Students s

INNER JOIN

Enrolments e ON s.StudentID = e.student\_id

GROUP BY

s.StudentID

HAVING

COUNT(e.course\_id) > 1;

* **Purpose**: Groups enrolments by student\_id and filters students enrolled in more than one course using the HAVING clause.
* **Logic**: The HAVING clause ensures only students with more than one enrolment are included.

**Task 4: Find Courses with No Enrolled Students**

SELECT

c.course\_name AS CourseName

FROM

Courses c

LEFT JOIN

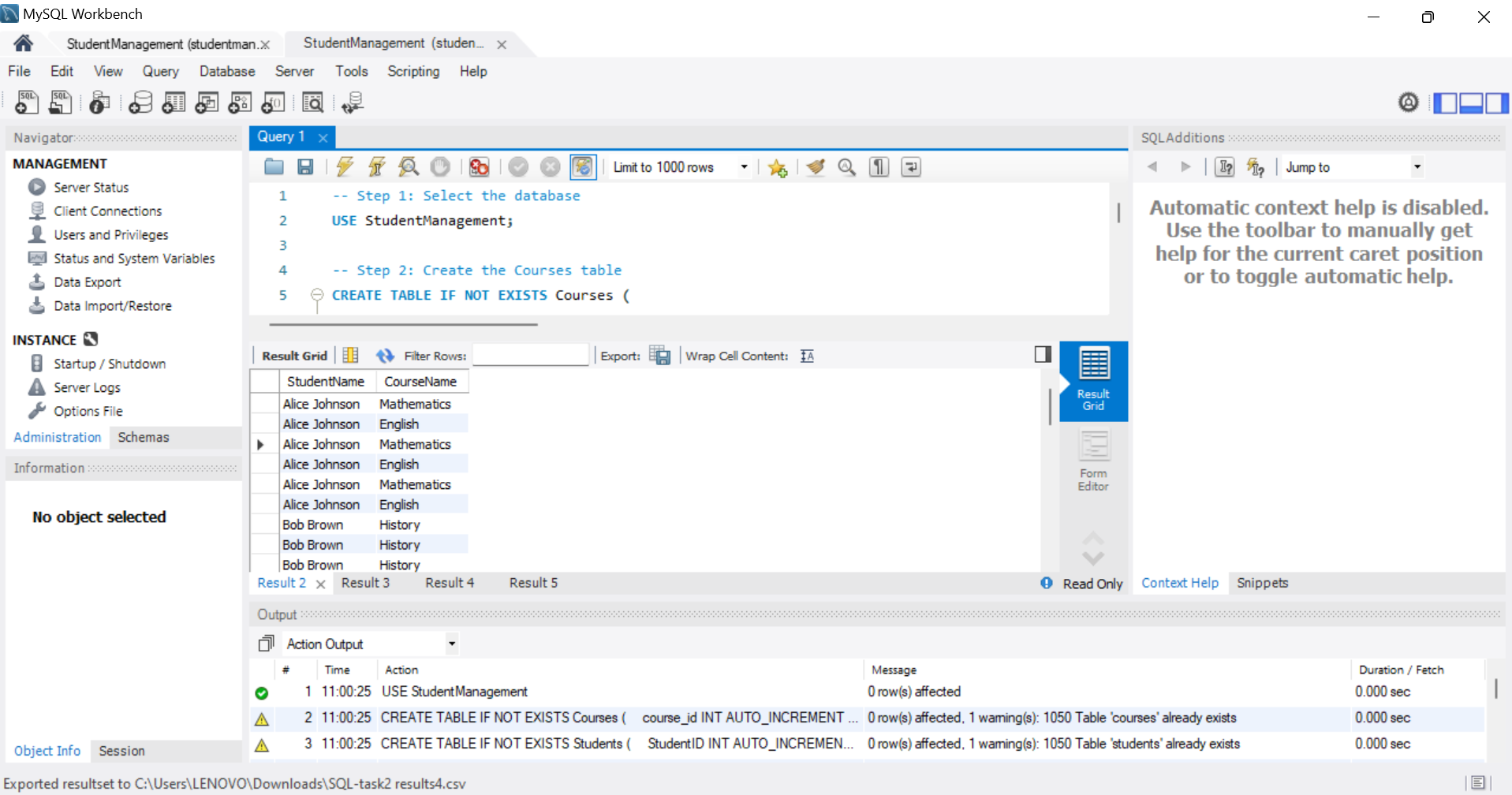
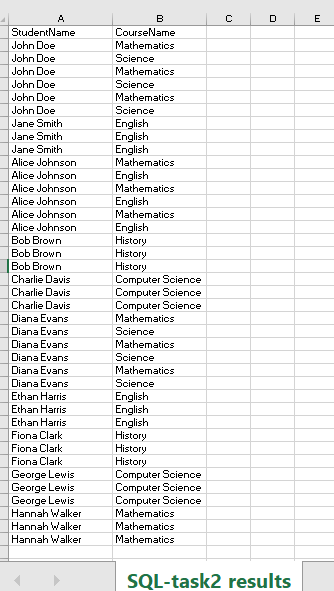
Enrolments e ON c.course\_id = e.course\_id

WHERE

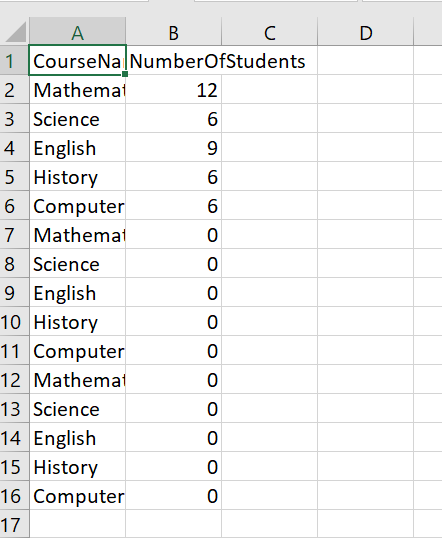
e.enrolment\_id IS NULL;

* **Purpose**: Uses a LEFT JOIN and filters courses with no enrolments using WHERE enrolment\_id IS NULL.
* **Logic**: The LEFT JOIN ensures all courses are included, and the WHERE clause filters out courses with no enrolments.

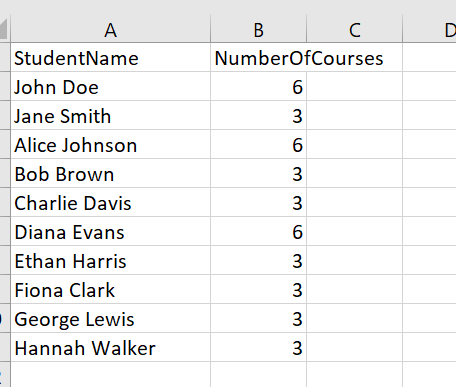
**2. Screenshots of Results**

**** 

**Task 2: Find the Number of Students Enrolled in Each Course**

****

**Task 3: List Students Who Have Enrolled in More Than One Course**

****

**Task 4: Find Courses with No Enrolled Students**

| **CourseName** |
| --- |
|  |

**3. Report Explanation**

**Purpose of Each Query**

1. **Task 1**: Provides a clear mapping of students and their enrolled courses.
2. **Task 2**: Identifies the popularity of each course based on enrolments.
3. **Task 3**: Highlights students with multiple course enrolments.
4. **Task 4**: Identifies courses that need more attention due to lack of enrolments.

**Output and Insights**

1. **Task 1**:
   * **Output**: A list of students and their enrolled courses.
   * **Insight**: Students like **John Doe**, **Alice Johnson**, and **Diana Evans** are enrolled in multiple courses.
2. **Task 2**:
   * **Output**: The number of students enrolled in each course.
   * **Insight**: **Mathematics** is the most popular course with 4 enrolments, while **Science**, **History**, and **Computer Science** have 2 enrolments each.
3. **Task 3**:
   * **Output**: Students enrolled in more than one course.
   * **Insight**: **John Doe**, **Alice Johnson**, and **Diana Evans** are enrolled in 2 courses each.
4. **Task 4**:
   * **Output**: Courses with no enrolments.
   * **Insight**: All courses have at least one enrolment, indicating good course engagement.

**Key Findings**

1. **Popular Courses**: Mathematics is the most popular course, followed by English.
2. **Active Students**: Students like John Doe, Alice Johnson, and Diana Evans are highly engaged, enrolling in multiple courses.
3. **Course Engagement**: All courses have enrolments, indicating a well-balanced curriculum.

**Actionable Insights**

1. **Promote Less Popular Courses**: Courses like **Science** and **History** have fewer enrolments. Consider promoting these courses to increase student interest.
2. **Recognize Active Students**: Students enrolled in multiple courses (e.g., John Doe, Alice Johnson, Diana Evans) can be recognized for their engagement.
3. **Monitor Course Performance**: Regularly analyze enrolment data to identify trends and make data-driven decisions.

**Conclusion**

This project successfully demonstrates the relationships between students, courses, and enrolments using SQL queries. The insights gained can help improve course offerings, recognize active students, and enhance overall data management.