***Group 17 DBS Assignment: APU College System Test Cases***

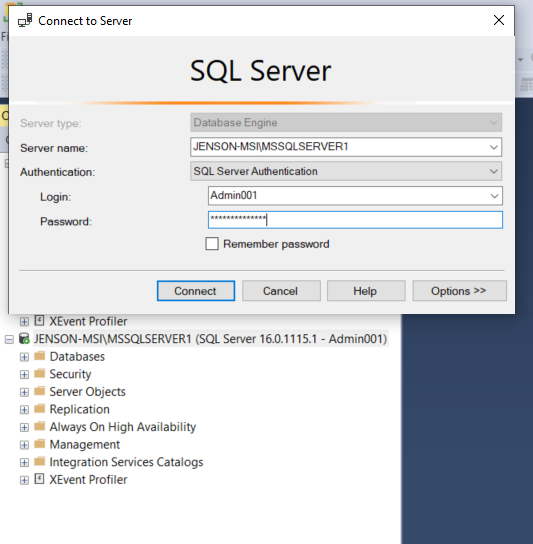
* **DB Admin – Part 1**

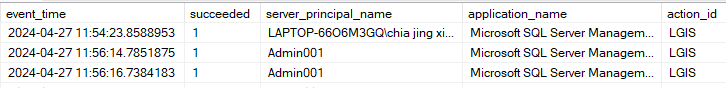
**DB Admin Part 1 Test Case Result Table**

|  |  |  |  |
| --- | --- | --- | --- |
| Test # | Description/SQL Query | Expected Output | Test Result (Pass/Fail) & Comments |
| 1 | Login as DBAdmin | DBAdmin can login successfully | Pass |
| 2 | Create minimum 3 student accounts (logins and users). Implement your permission control as per your Authorization matrix. | All student logins and users are created successfully. | Pass |
| 3 | Create a 4 lecturer accounts (logins and users). Names them Lecturer1, Lecturer2, Lecturer3, Lecturer4. Implement your permission control as per your Authorization matrix. Lecturer1 and Lecturer2 is from Computer Science Department. Lecturer3 and Lecturer4 is from Engineering Department. | All lecturer logins and users are created successfully. | Pass |
| 4 | Create a view based on the Student table. | Table can be created successfully | Pass |
| 5 | Add a new student record with a password. | Query should not be allowed. | Pass |
| 6 | Add a new lecturer record with a password. | Query should not be allowed. | Pass |
| 7 | Add a new student record without a password. | Query should be allowed, and record added successfully. | Pass |
| 8 | Add a new lecturer record without a password. | Query should be allowed, and record added successfully. | Pass |

**DB Admin Part 1 Test Cases’ Output**

1. Test Case 1 - Login as DBAdmin

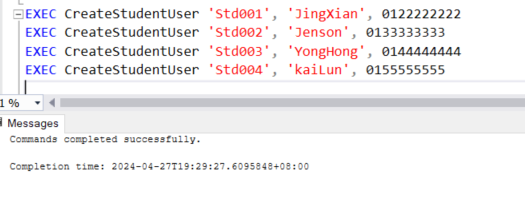




*Figure 1: Login as DB Admin*

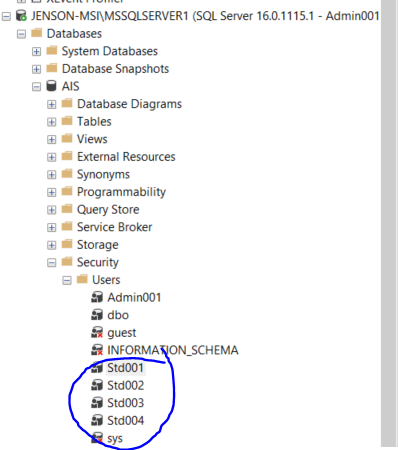
Above output shows the successful Login as an Admin

1. Test Case 2 - Create student accounts

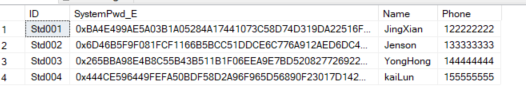


*Figure 2: Admin create Student Account*

Student of 4 had been created by using the “CreateStudentUser” procedure. The process had shown successfully.



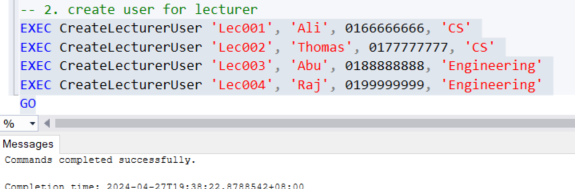
*Figure 3: Student users' instance stored in database*



*Figure 4: Students' accounts data stored into table*

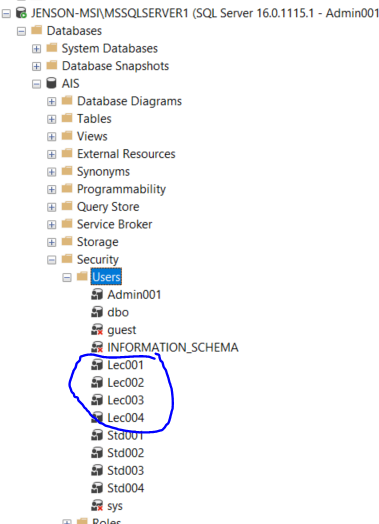
After the creation of student accounts, the user instances had been stored into the database shown in figure 3. The respective students’ data had also been stored into the Student table shown in figure 4.

1. Test Case 3 - Create Lecturer accounts

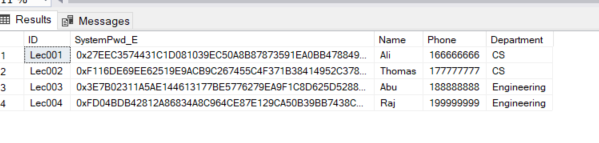


*Figure 5: Admin create lecturer accounts*

Figure 5 shows the process of Admin creating 4 lecturer accounts, which had been shown successful.



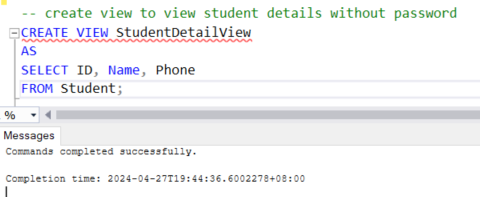
*Figure 6: Lecturer users' instance stored in database*



*Figure 7: Lecturer accounts data stored into table*

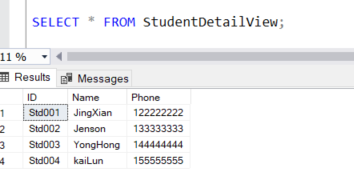
After the creation of lecturer accounts, the lecturer user instances had been stored into the database shown in figure 6. The respective lecturer’s data had also been stored into the Student table shown in figure 7.

1. Create a view based on the Student table



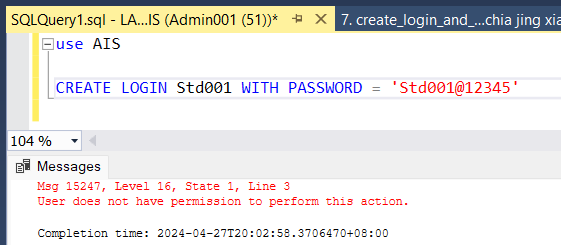
*Figure 8: Student table view for Admin*

Figure 8 had shown the successful creation of a student table view by Admin. The password column had been hidden from the lecturer.



*Figure 9: View student table*

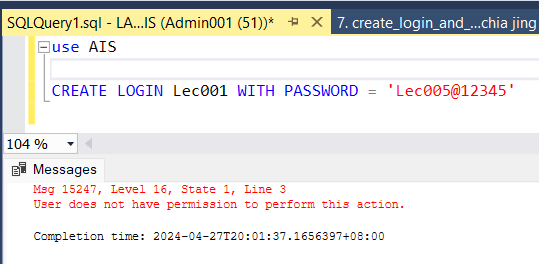
1. Add a new student record with a password.



*Figure 10: Add New Student with Password*

Figure 10 shows that the admin tried to create a login account for student 001 with password, the output shows that the admin did not have the permission to create a server account for student.

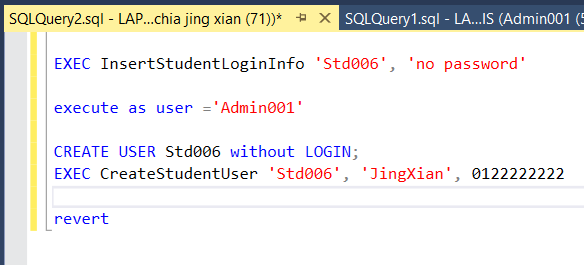
1. Add a new lecturer record with a password.

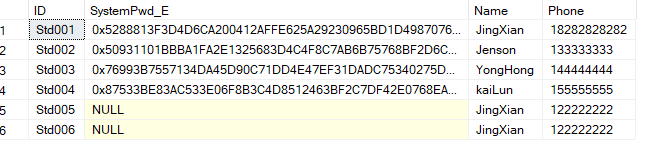


*Figure 11: Add New Lecturer with Password*

Figure 11 shows that the admin tried to create a login account for Lecturer 001 with password, the output shows that the admin did not have the permission to create a server account for Lecturer.

1. Add a new student record without a password.

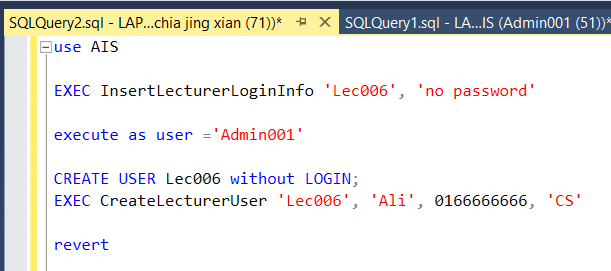


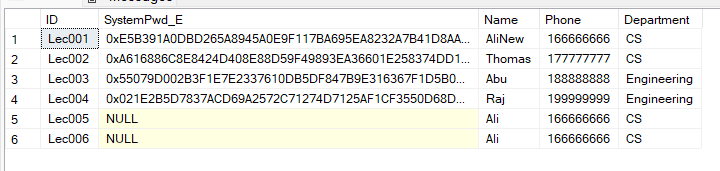


*Figure 12: Add New Student without Password*

Figure 12 shows that the admin tried to create a user account for Student 006 without password in the database, the code is executing the InsertStudentLoginInfo procedure, as the procedure can only be executed by database owner, admin did not have the permission to create login account. The output shows that the admin has the permission to insert or add a new student into the student table, and only display the ID, Name and phone only

1. Add a new lecturer record without a password.





*Figure 13: Add New Lecturer without Password*

Figure 13 shows that the admin can create a user account for Student 006 without password within the database, the output shows that the admin has the permission to insert or add a new student into the student table, and only display the ID, Name and phone only.

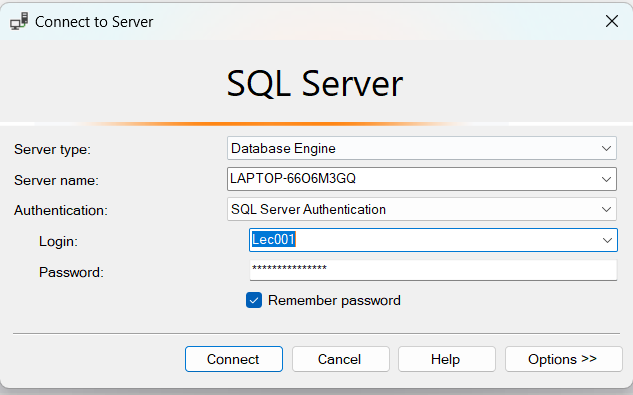
* **Lecturer**

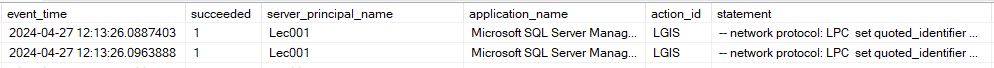
**Lecturer Test Cases Result Table**

|  |  |  |  |
| --- | --- | --- | --- |
| Test # | Description/SQL Query | Expected Output | Test Result (Pass/Fail) |
| 1 | Login as Lecturer1 | Lecturer1 can login successfully | Pass |
| 2 | Login as Lecturer1 and check personal details | Own personal details that are stored in DB are shown without any encryption to the lecturer | Pass |
| 3 | Login as Lecturer1 and update name | New value for name is updated successfully into the database | Pass |
| 4 | Login as Lecturer2, run “Select \* From Student” query | Login successful but Query should not be allowed. | Pass |
| 5 | Login as Lecturer3, run “Select ID, Name, Phone From Student” query | Login successful and all student records shown. | Pass |
| 6 | Login as Lecturer4, run “EXEC readLecturerProcedure” query | Login successful. Query should be allowed but only own data shown | Pass |
| 7 | Login as each lecturer and add 1-2 rows in the Result table | Data added successfully. | Pass |
| 8 | Login as Lecturer2, run “EXEC viewMarkByDepartment” | All rows that belong to the same lecturer department must be shown including rows newly added by Lecturer1. | Pass |
| 9 | Login as Lecturer4, try to delete a row that was newly added by Lecturer3. | Row must NOT be deleted. | Pass |
| 10 | Login as Lecturer3, try to update a row that was newly added by Lecturer3. | Row is deleted successfully. | Pass |

**Lecturer Test Cases’ Output**

1. Login as Lecturer1

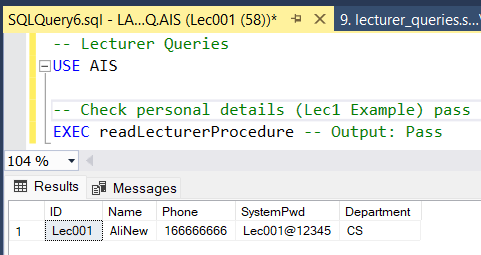




*Figure 14: Lecturer1 Login Successful*

Figure 14 shows the Lecturer 001 login in the server successfully, and the output shows that the Lecturer 001 logged into the database successfully.

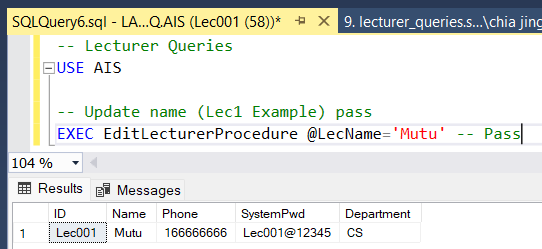
1. Login as Lecturer1 and check personal details



*Figure 15: Lecturer 001 view own data*

Figure 15 shows the Lecturer 001 login in the server successfully and checks the Lecturer 001 data, and the output only shows the Lecturer 001 personal data.

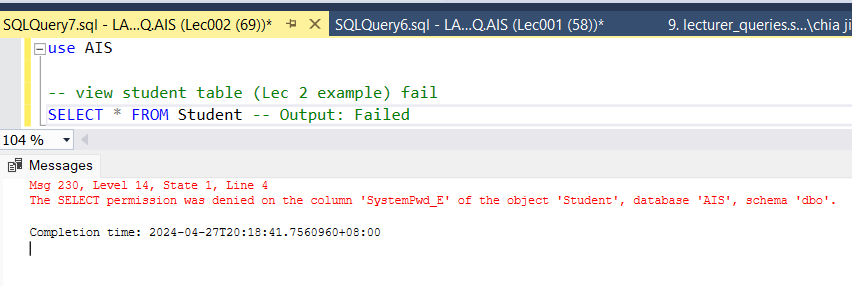
1. Login as Lecturer1 and update name



*Figure 16: Lecturer 001 update own data*

Figure 16 shows the Lecturer 001 logged into the server successfully and updating the Lecturer 001 personal data, and the output shows that the Lecturer 001 updated the Lecturer name from AliNew to Mutu successfully.

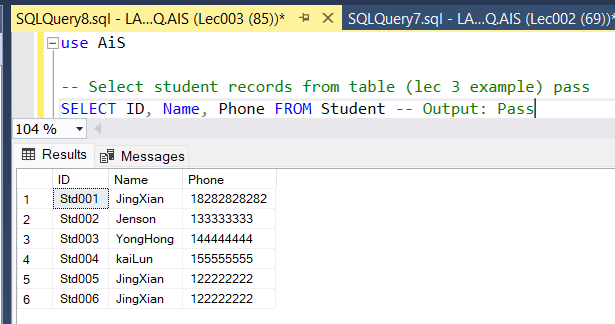
1. Login as Lecturer2, run “Select \* From Student” query



*Figure 17: Lecturer 2 View Student Table*

Figure 17 shows the Lecturer 002 logged into the server successfully, but the Lecturer role cannot view the student information table, Lecturer did not have permission to view the student information table, as student information table included the student password, Lecturer did not have the permission to view student password.

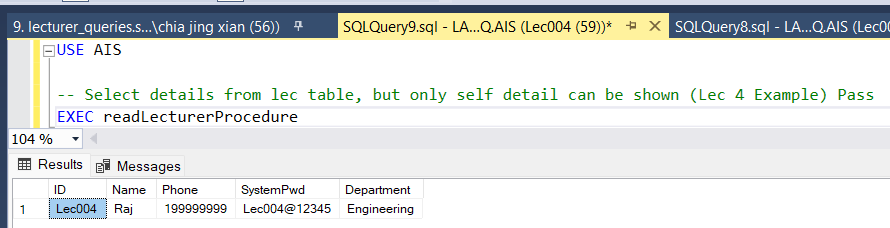
1. Login as Lecturer3, run “Select ID, Name, Phone From Student” query



*Figure 18: Lecturer 3 View General information of Student*

Figure 18 shows the Lecturer 003 login in the server successfully, and view the general information of the student table successfully, without the sensitive information (student password).

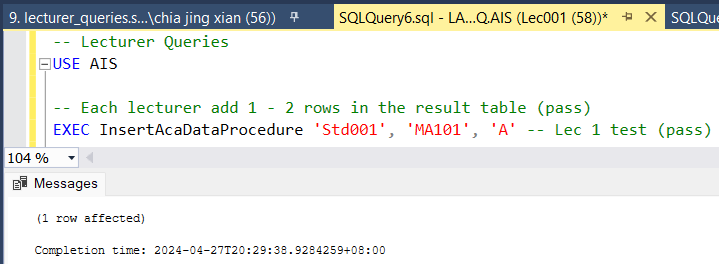
1. Login as Lecturer4, run “EXEC readLecturerProcedure” query



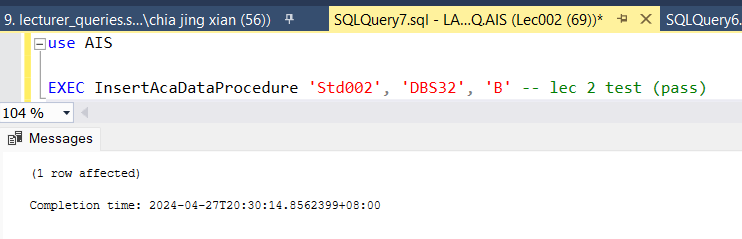
*Figure 19: Lecturer 4 View Own Data*

Figure 19 shows the Lecturer 004 logged into the server and viewing his own data successfully.

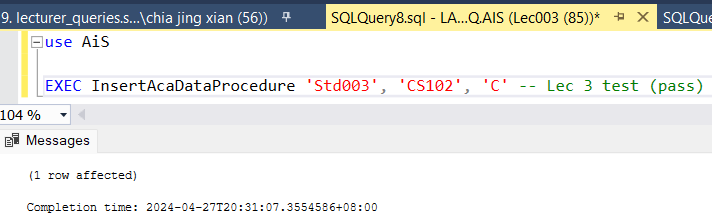
1. Login as each lecturer and add 1-2 rows in the Result table



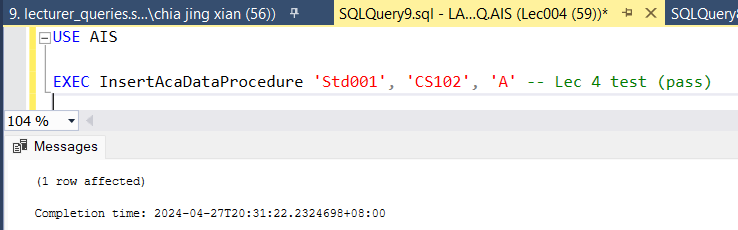
*Figure 20: Lecturer 1 insert academic data into result table*



*Figure 21: Lecturer 2 insert academic data into result table*

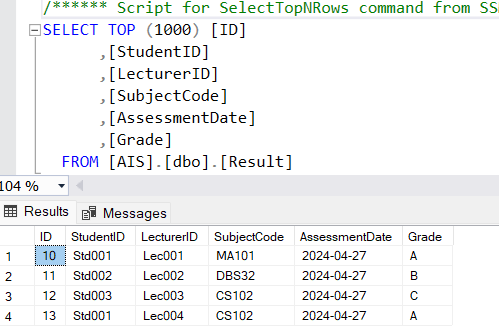


*Figure 22: Lecturer 3 insert academic data into result table*



*Figure 23: Lecturer 4 insert academic data into result table*

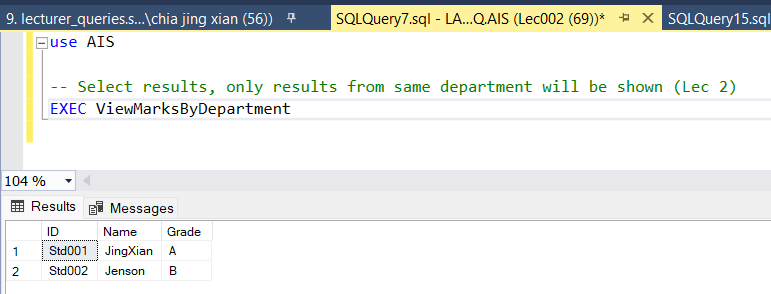
Figure 20, Figure 21, Figure 22, Figure23 show that the 4 Lecturers (001 - 004) insert different student, subject and academic data successfully into the Result table.



*Figure 24: View Result table*

Figure 24 shows that the Academic data was inserted successfully by the 4 Lecturers (001 - 004), including the student id, lecturer is, subject code, assessment data and grade of each course. the student 001 inserted a different subject and with its grade successfully into the Result table.

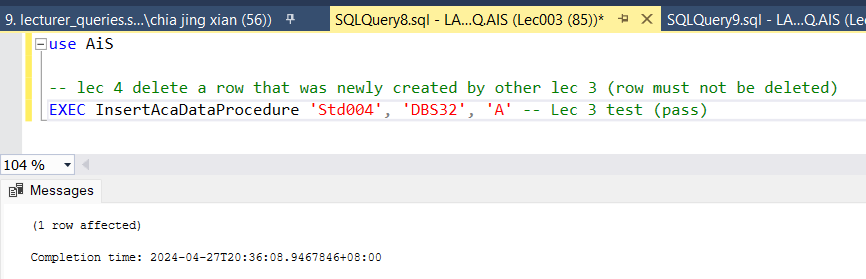
1. Login as Lecturer2, run “EXEC viewMarkByDepartment”



*Figure 25: Lecturer 2 view the student in same department*

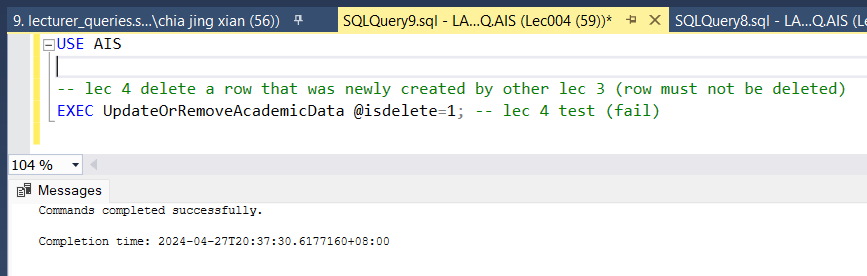
Figure 25 shows the Lecturer 002 viewing the student and academic data in the same department successfully.

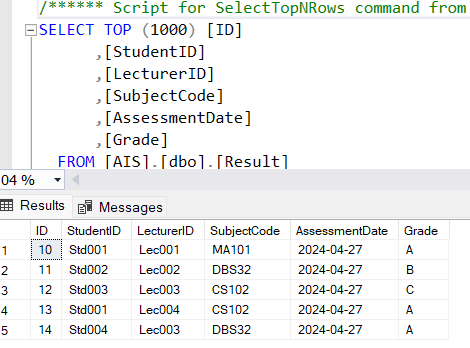
1. Login as Lecturer4, try to delete a row that was newly added by Lecturer3.



*Figure 26: Lecturer 3 insert Result*

Figure 26 shows the Lecturer 003 insert a new academic data into the result table successfully.

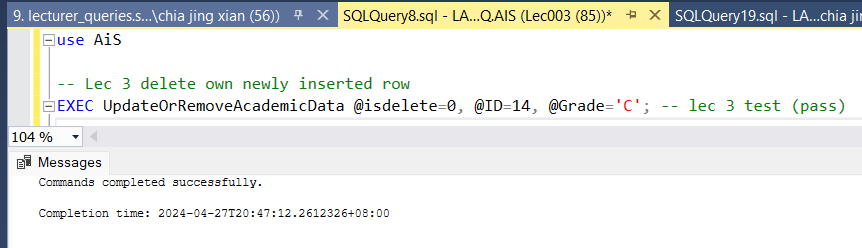


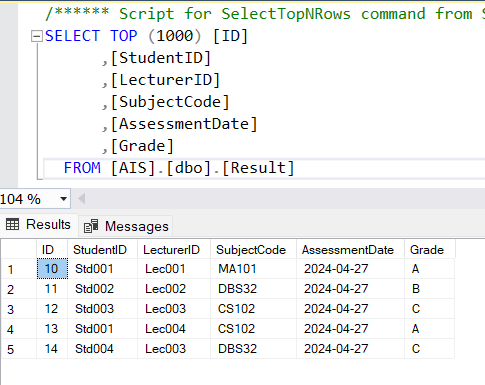


*Figure 27: Lecturer 4 delete the latest result*

Figure 27 shows that Lecturer 004 cannot delete the data that was inserted by Lecturer 003 or other Lecturer.

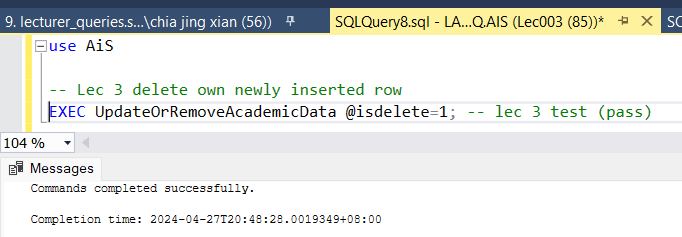
1. Login as Lecturer3, try to update a row that was newly added by Lecturer3.

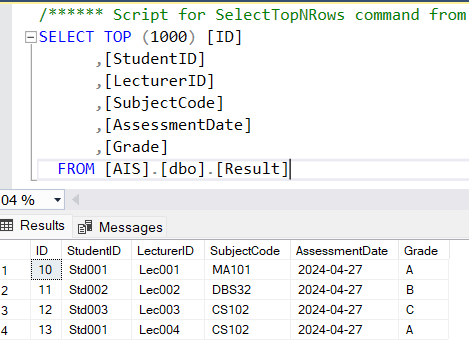




*Figure 28: Lecturer 3 update the academic data*

Figure 28 shows that the Lecturer 003 update the specific row (id =14) academic data successfully, from A to C.





*Figure 29: Lecturer 3 delete own student academic data*

Figure 29 shows that Lecturer 3 deleted the Lecturer 3’s student academic data successfully.

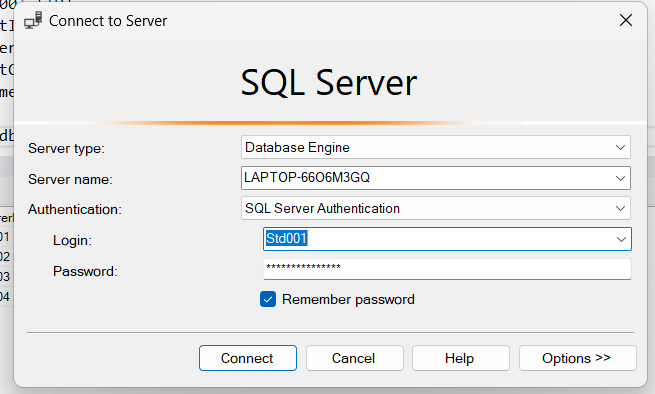
* **Student**

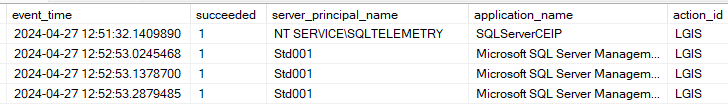
**Student Test Cases’ Result Table**

|  |  |  |  |
| --- | --- | --- | --- |
| Test # | Description/SQL Query | Expected Output | Test Result (Pass/Fail) |
| 1 | Login as Student1 | Student1 can login successfully | Pass |
| 2 | Login as Student1 and check personal details | Own personal details that are stored in DB are shown without any encryption to the student | Pass |
| 3 | Login as Student1 and update phone number | New phone number is updated successfully into the database | Pass |
| 4 | Login as Student2, run “EXEC readuserprocedure” query | Login successful. Query should be allowed but only own data shown | Pass |
| 5 | Login as Student3, run “EXEC readStudentResultProcedure” query | Login successful. Query should be allowed but only own data shown | Pass |
| 6 | Login as Student4, run “Delete From Result” query | Login successful but Query should not be allowed | Pass |
| 7 | Login as Student2, run “Select \* From Lecturer” query | Login successful but Query should not be allowed | Pass |
| 8 | Login as Student2 and run “Insert Into Student (ID, SystemPwd, Name, Phone) Values (‘S1111’,’S1111’,’Ramasamy’,’012345678’) | Login successful but Query should not be allowed | Pass |

**Student Test Cases’ Output**

1. Login as Student1

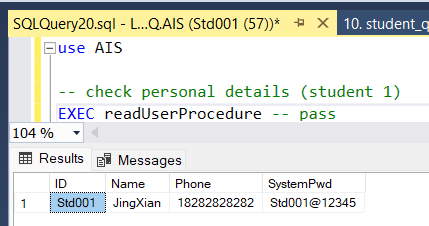




*Figure 30: Login Student 001 successfully*

Figure 30 shows that Student 001 login to the server successfully, the output shows that the student 001 login to the server successfully.

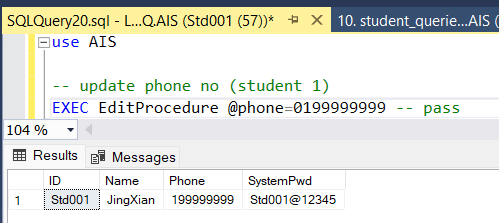
1. Login as Student1 and check personal details



*Figure 31: View student 001 personal data*

Figure 31 shows the student 001 only can view the own data only.

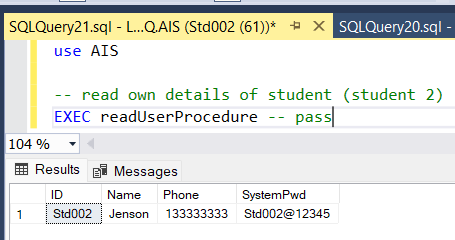
1. Login as Student1 and update phone number



*Figure 32: Update student table*

Figure 32 shows student 001 updating the personal data into the student table successfully.

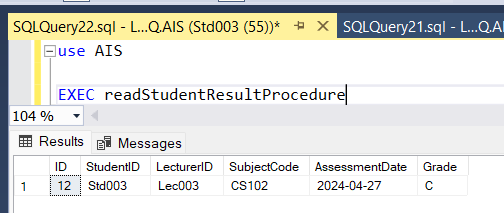
1. Login as Student2, run “EXEC readuserprocedure” query



*Figure 33: View student 2 personal data*

Figure 33 shows that student 002 viewed personal data successfully.

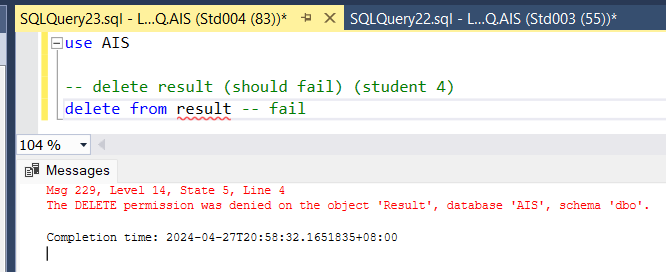
1. Login as Student3, run “EXEC readStudentResultProcedure” query



*Figure 34: View student 3 own academic result*

Figure 34 shows that the Student 001 only can view the student's own academic data successfully.

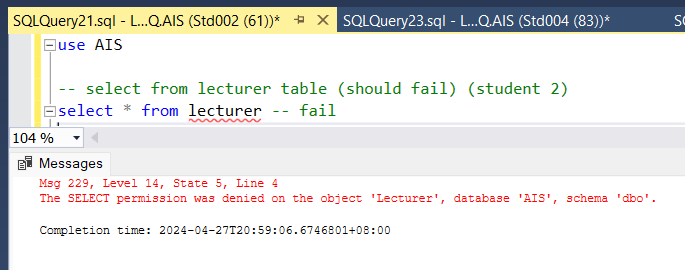
1. Login as Student4, run “Delete From Result” query



*Figure 35: Student delete result table*

Figure 35 shows that the student 4 or student cannot delete the result table.

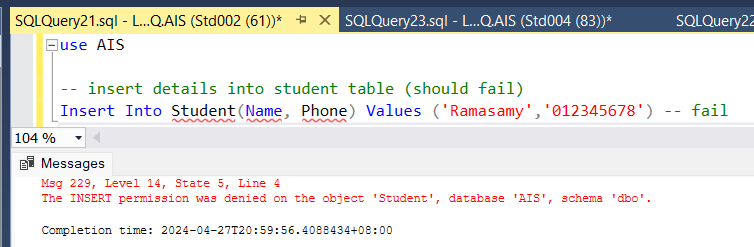
1. Login as Student2, run “Select \* From Lecturer” query



*Figure 36: Student View Lecturer table*

Figure 36 shows that student 002 or student cannot view lecturer data.

1. Login as Student2 and run “Insert Into Student (ID, SystemPwd, Name, Phone) Values (‘S1111’,’S1111’,’Ramasamy’,’012345678’)



*Figure 37: Student Insert data into student table*

Figure 37 shows that students cannot insert new student data into the student table.

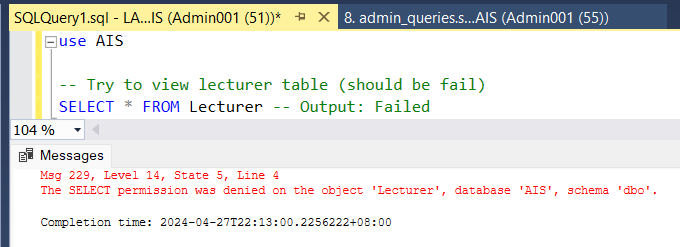
* **DBAdmin - Part 2**

**DBAdmin Part 2 Test Case’s Result Table**

|  |  |  |  |
| --- | --- | --- | --- |
| Test # | Description/SQL Query | Expected Output | Test Result (Pass/Fail) |
| 1 | Check lecturer data. Run “Select \* From Lecturer” | Query should not be allowed. | Pass |
| 2 | Check lecturer data without password. “SELECT \* FROM LecturerDetailView;” | List of lecturer name and phone is shown. | Pass |
| 3 | Check data from the result table. Run “Select \* From Result” | Query should not be allowed. | Pass |
| 4 | Remove academic data. Run “Delete From Result” | Query should not be allowed. | Pass |

**DBAdmin Part 2 Test Cases’ Output**

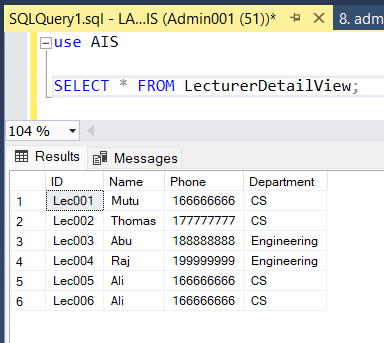
1. Check lecturer data. Run “Select \* From Lecturer”



*Figure 38: Admin View Lecturer table*

Figure 38 shows that Admin cannot view lecturer data, as admin did not have permission to view the lecturer table, in the lecturer table include lecturer password.

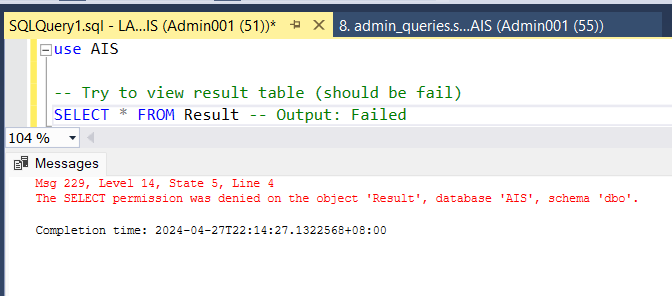
1. Check lecturer data without password. “SELECT \* FROM LecturerDetailView;”



*Figure 39: Admin View general Lecturer data*

Figure 39 shows that admin can view the general Lecturer data in the Lecturer table.

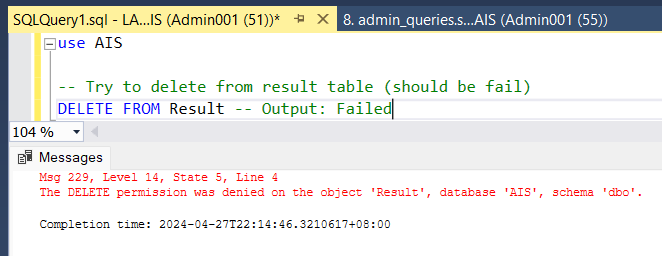
1. Check data from the result table. Run “Select \* From Result”



*Figure 40: Admin view result table*

Figure 40 shows that admin cannot view the academic data in the Result table.

1. Remove academic data. Run “Delete From Result”



*Figure 41: Admin delete academic data from result table*

Figure 41 shows that admin cannot delete the academic data in the Result table, as admin did not have permission to delete the academic data.