# Module Four

# Scenario pt. 2

# Grandfield College

Andrew Robinson

IT 650: Principles of Database Design

Southern New Hampshire University

August 22, 2024

Daniel Foster

## Test Plan

To set up a test plan for the Grandfield College database, follow these steps to ensure that your database adheres to the business rules and functions correctly. Here’s a structured approach to testing the database:

#### Simple SELECT Queries

Rule 1: Verify that software details are correctly stored in the database.

SQL Query:

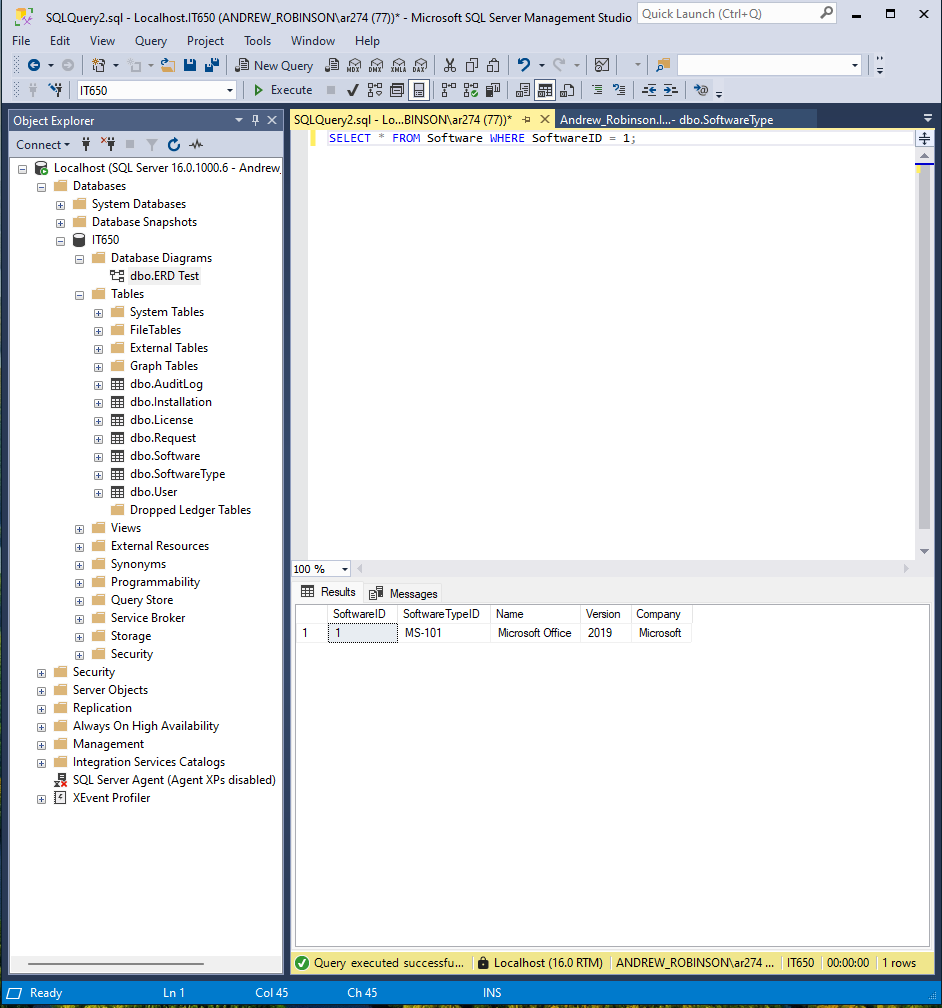
SELECT \* FROM Software WHERE SoftwareID = 1;

Expected Result:

- Displays details of the software with `SoftwareID` 1.

Pass/Fail:

- Pass: The output shows correct details for the specified `SoftwareID`.



Rule 2: Check that all installations for a specific user are retrieved.

SQL Query:

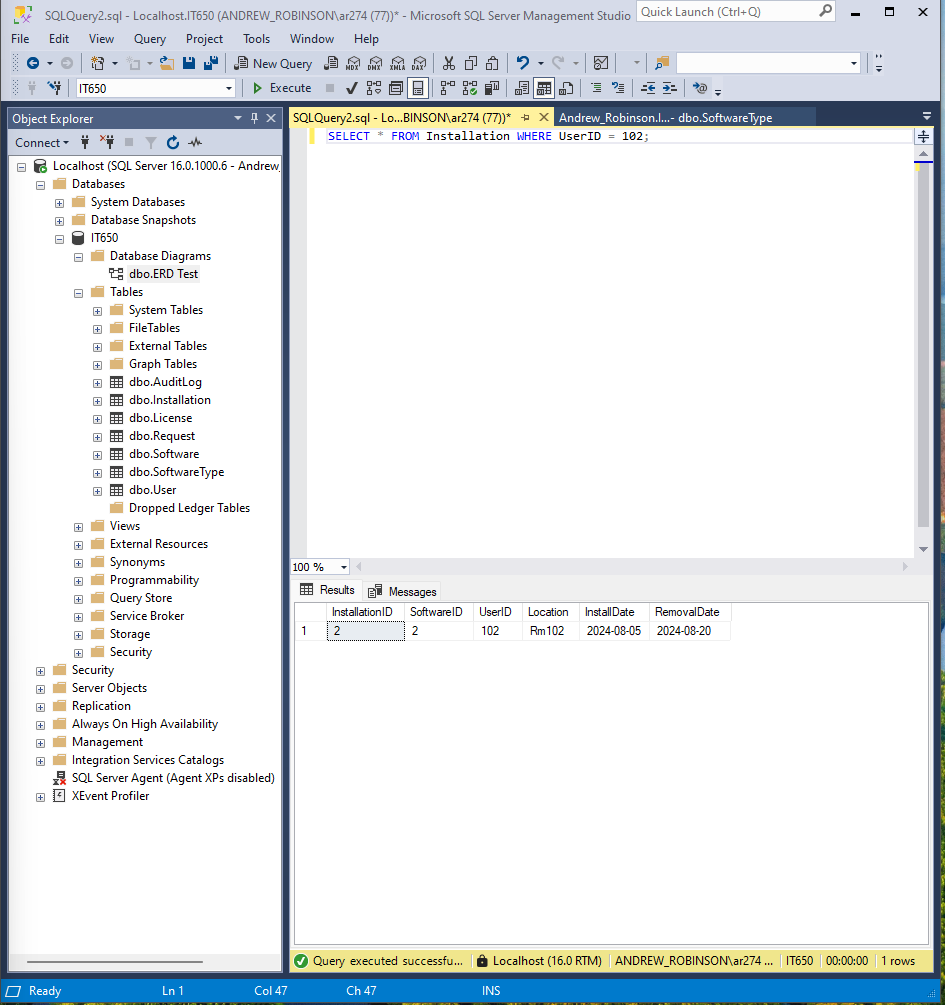
SELECT \* FROM Installation WHERE UserID = 102;

Expected Result:

- Displays all installations assigned to the user with `UserID` 102.

Pass/Fail:

- Pass: The output matches the expected installations for the user.



#### Aggregate Functions

Rule 3: Calculate the total number of software installations.

SQL Query:

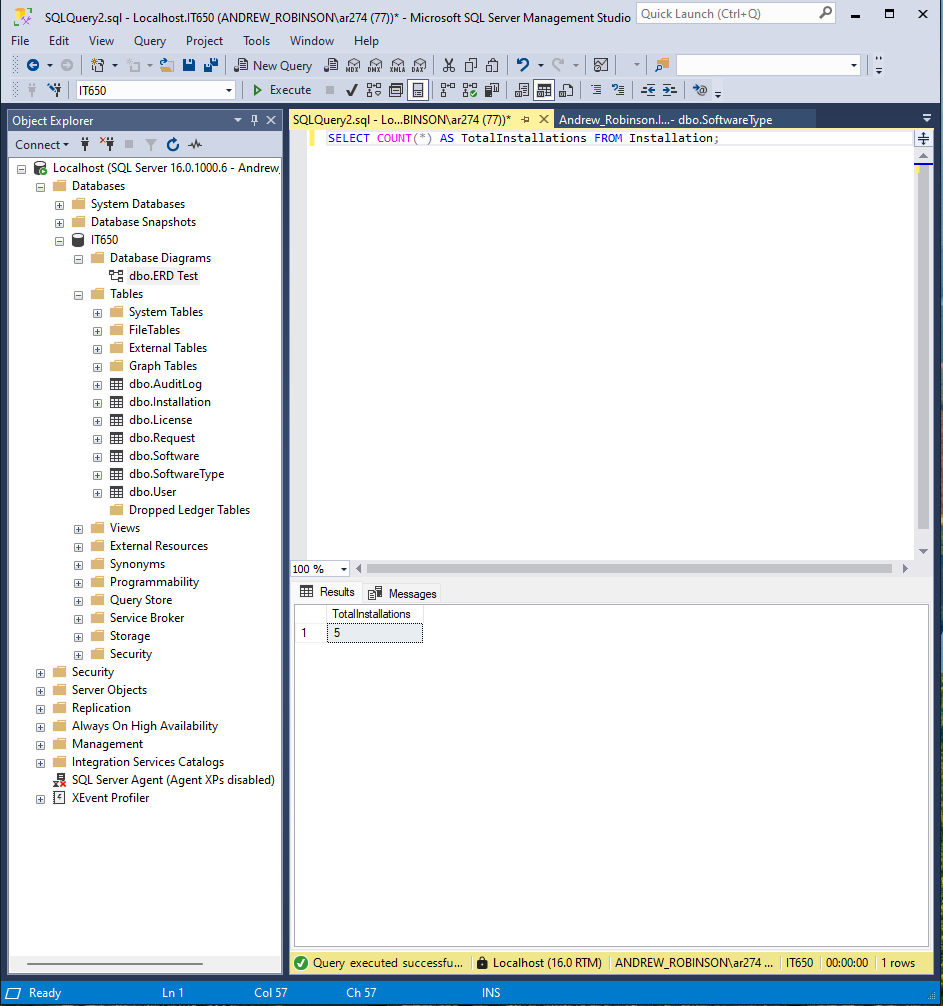
SELECT COUNT(\*) AS TotalInstallations FROM Installation;

Expected Result:

- Displays the total number of installations recorded in the `Installation` table.

Pass/Fail:

- Pass: The count matches the expected number of installations.



Rule 4: Find the average pricing of software licenses.

SQL Query:

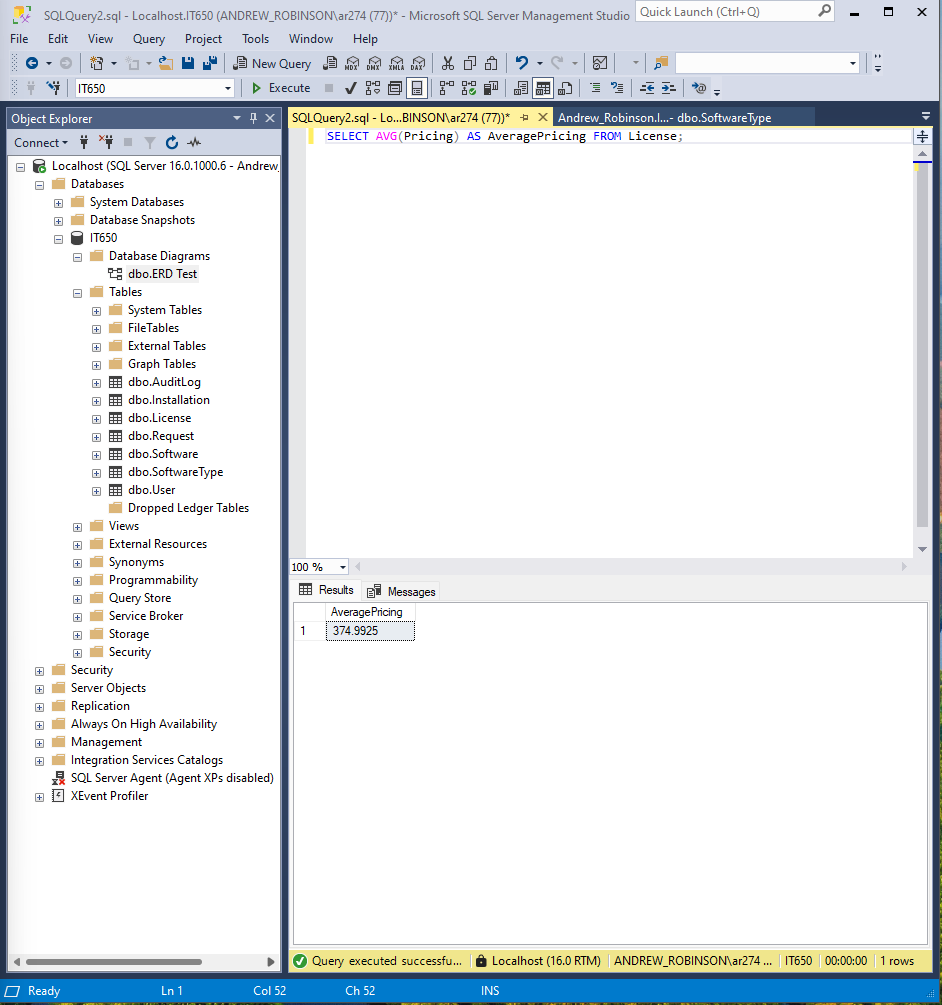
SELECT AVG(Pricing) AS AveragePricing FROM License;

Expected Result:

- Displays the average pricing of licenses from the `License` table.

Pass/Fail:

- Pass: The average price is accurate according to the data.



#### 

#### Queries Using Joins

Rule 5: Retrieve all software installations along with user names.

SQL Query:

SELECT i.InstallationID, u.Name, s.Name AS SoftwareName

FROM Installation i

JOIN [User] u ON i.UserID = u.UserID

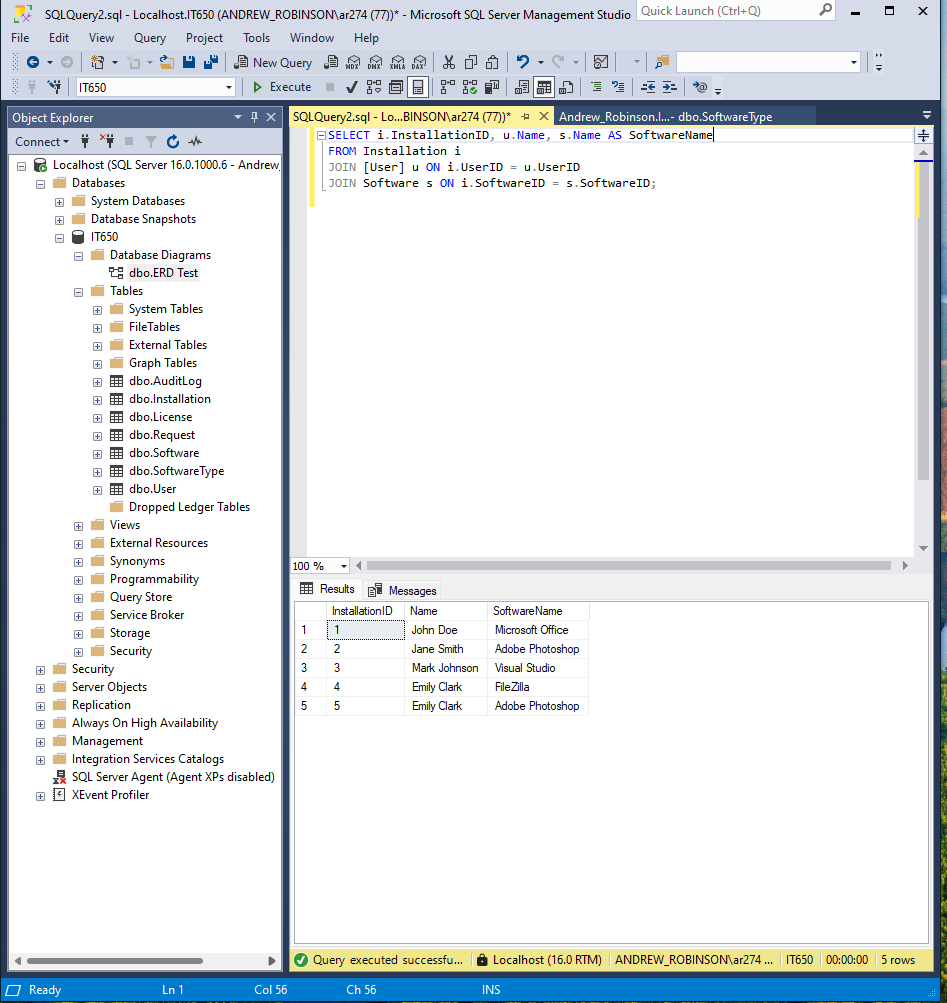
JOIN Software s ON i.SoftwareID = s.SoftwareID;

Expected Result:

- Displays installation details with corresponding user names and software names.

Pass/Fail:

- Pass: The output correctly shows the user names and software details associated with each installation.



Rule 6: List all software and their licensing information.

SQL Query:

SELECT s.Name AS SoftwareName, l.LicenseType, l.StartDate, l.EndDate

FROM Software s

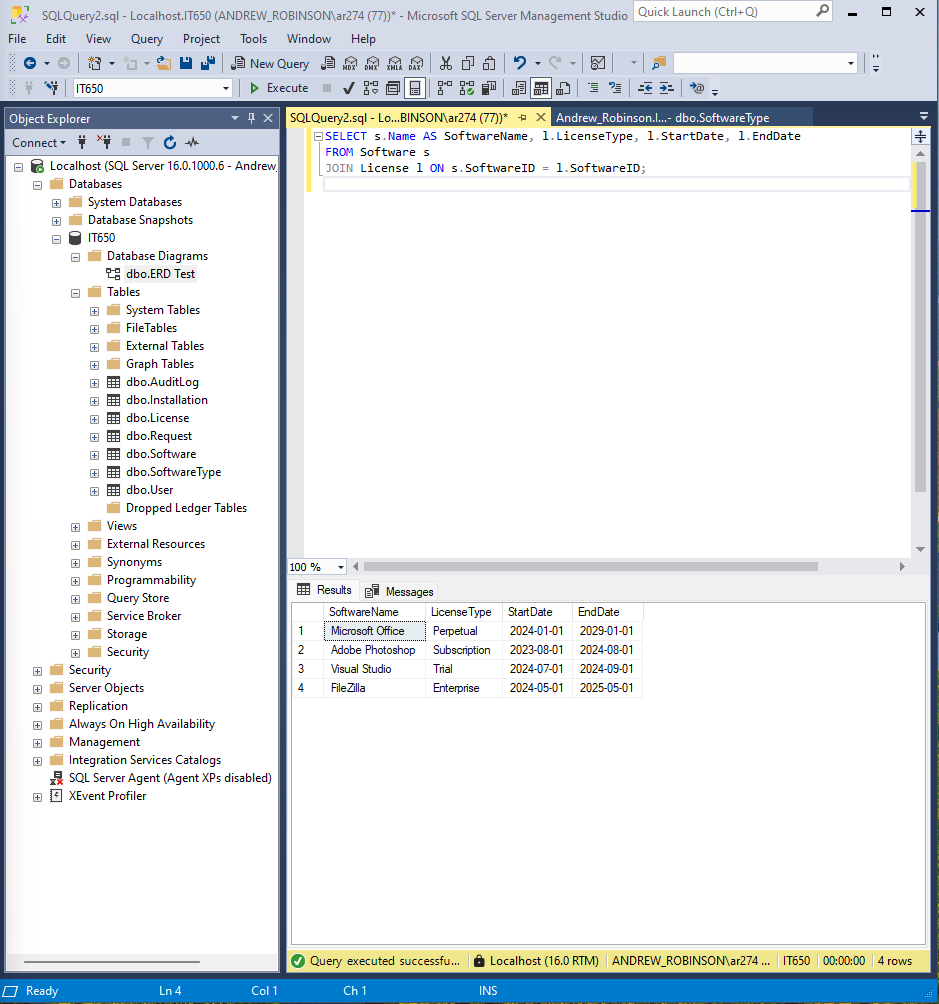
JOIN License l ON s.SoftwareID = l.SoftwareID;

Expected Result:

- Displays software names along with their corresponding licensing details.

Pass/Fail:

- Pass: The output matches the expected software and license information.



#### INSERT Statements

Rule 7: Add a new software record.

SQL Query:

INSERT INTO Software (SoftwareID, SoftwareTypeID, Name, Version, Company)

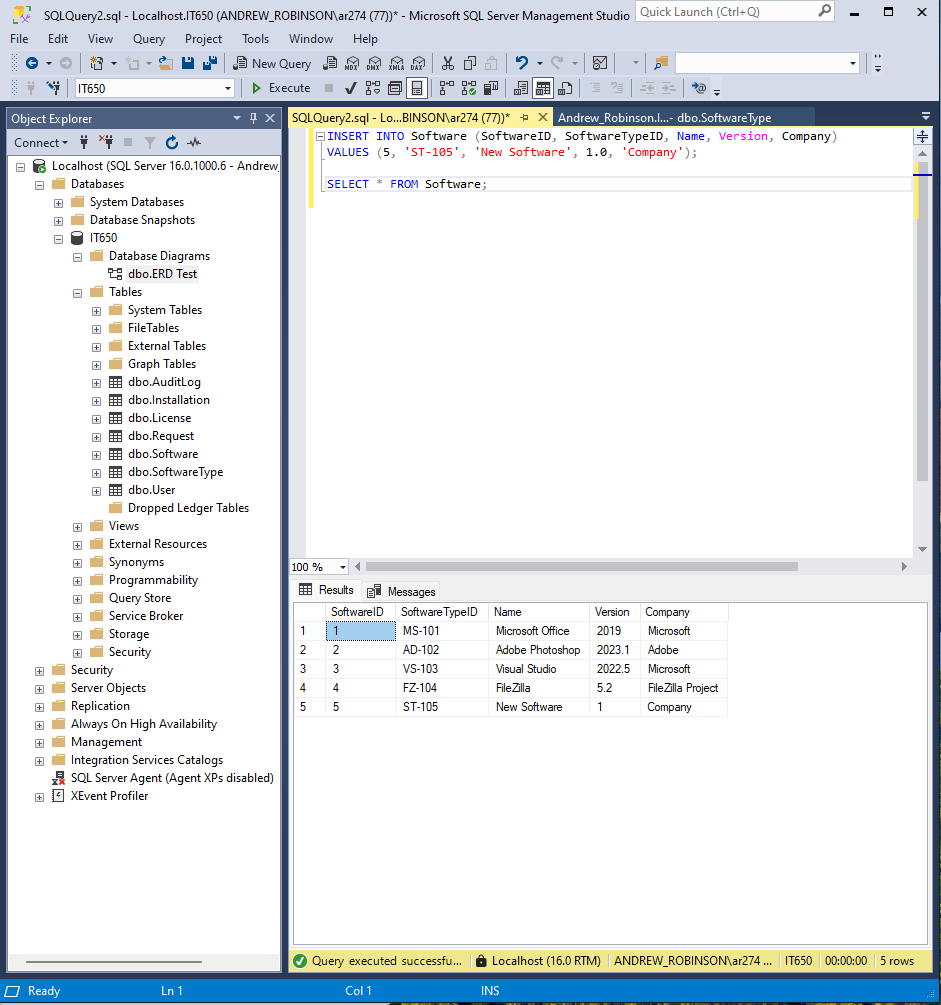
VALUES (5, 'ST-105', 'New Software', 1.0, 'Company');

Expected Result:

- New software record is inserted into the `Software` table.

Pass/Fail:

- Pass: The new record appears in the `Software` table.



Rule 8: Add a new installation record.

SQL Query:

INSERT INTO Installation (InstallationID, SoftwareID, UserID, Location, InstallDate, RemovalDate)

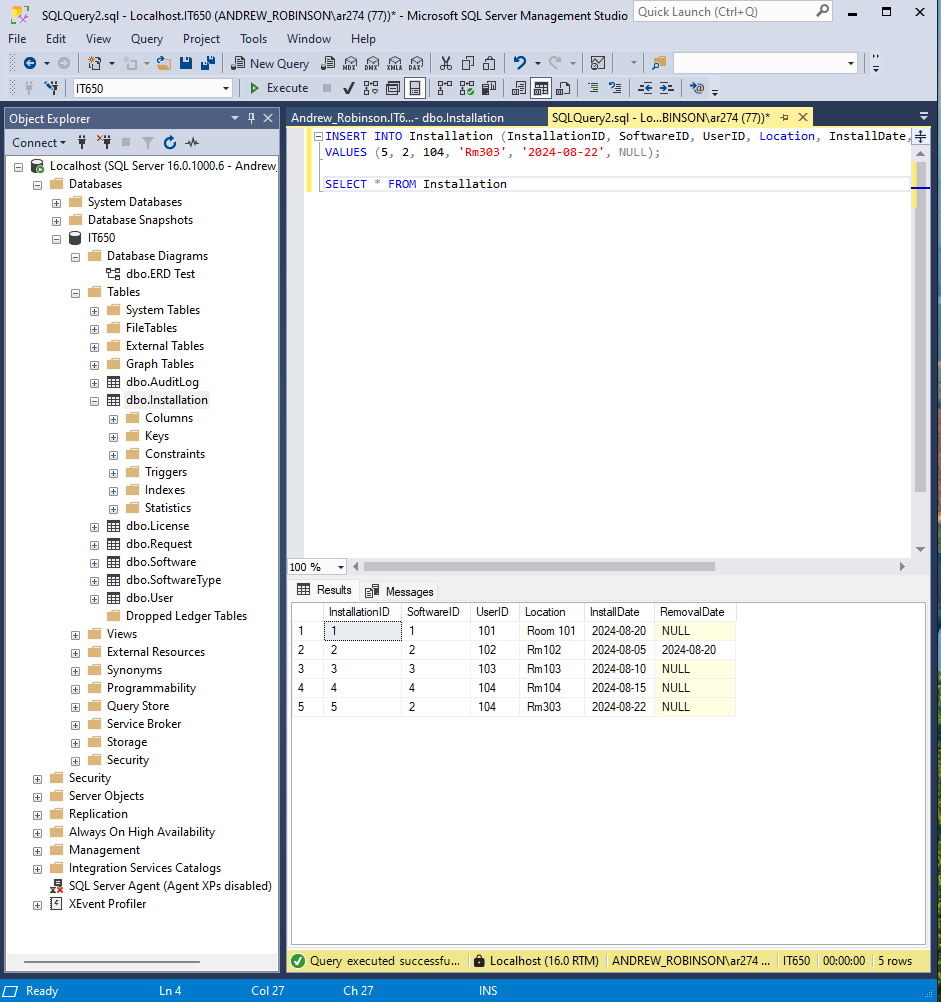
VALUES (5, 2, 104, 'Rm303', '2024-08-22', NULL);

Expected Result:

- New installation record is inserted into the `Installation` table.

Pass/Fail:

- Pass: The new record appears in the `Installation` table.



#### UPDATE Statements

Rule 9: Update the software version.

SQL Query:

UPDATE Software

SET Version = 4.0

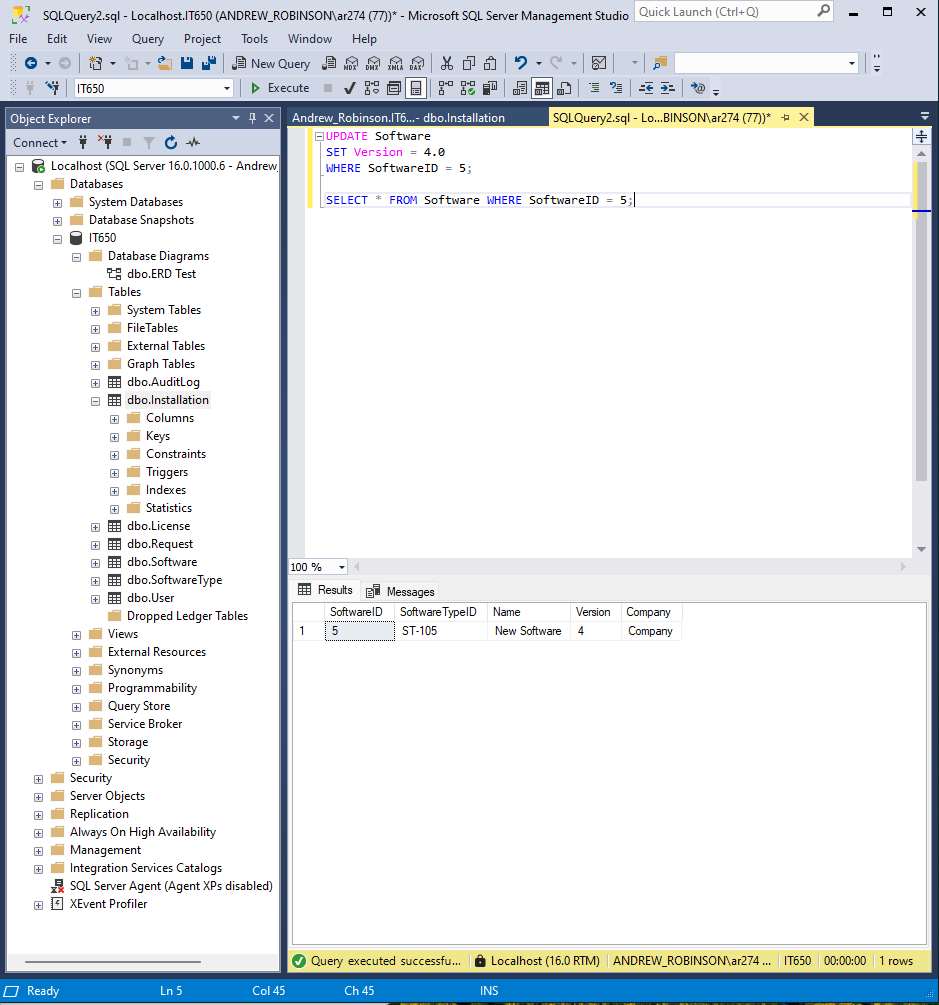
WHERE SoftwareID = 5;

Expected Result:

- The `Version` for `SoftwareID` 5 should be updated to 2.0.

Pass/Fail:

- Pass: The `Version` field is updated correctly.



## References

Conger, S. (2014). Hands-On Database (2nd ed.). Pearson Education (US). https://mbsdirect.vitalsource.com/books/9780133927078