

BDM 1034 - Application Design for Big Data

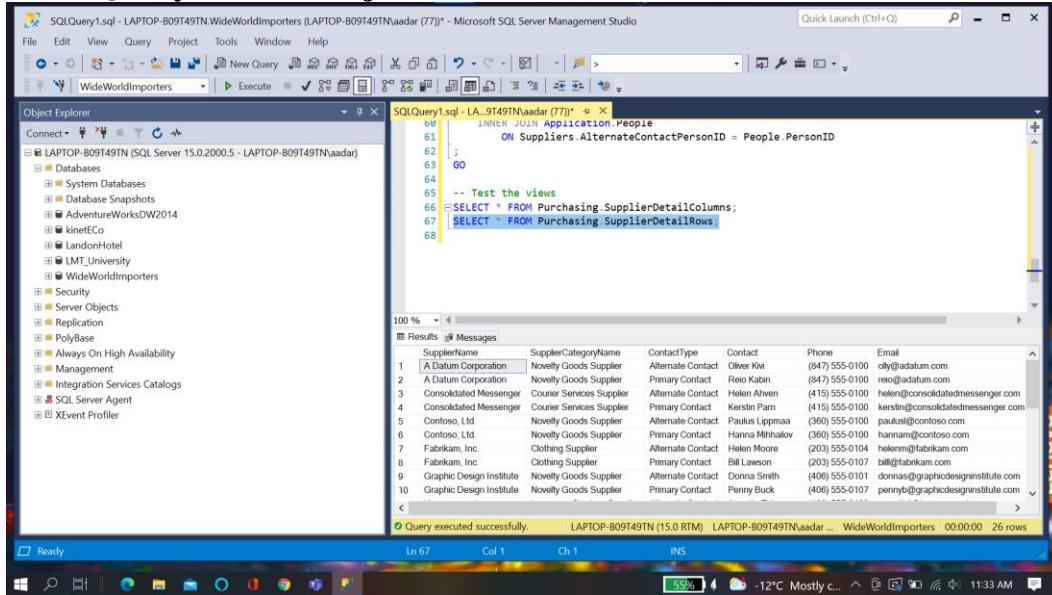
Lab4

**Submitted by: Aadarsha Chapagain
Student ID:C0825975**

Submitted to: Prof. Teresa Zhu

Here I have attached the certificate I achieved from Linkedin learning for Course “**Program Databases with Transact-SQL**” along with the Screenshot of score I achieved

Chapter 1: Save Query as view object



The screenshot shows the Microsoft SQL Server Management Studio (SSMS) interface. The title bar reads "SQLQuery1.sql - LAPTOP-B09T49TN.WideWorldImporters (LAPTOP-B09T49TN\aadar (77)) - Microsoft SQL Server Management Studio". The main window has two panes: Object Explorer on the left and Results grid on the right. The Object Explorer shows the database structure, including databases like WideWorldImporters. The Results grid displays the output of a query that joins the Suppliers and Application.People tables. The query is as follows:

```
SELECT * FROM Purchasing.SupplierDetailColumns;
SELECT * FROM Purchasing.SupplierDetailRows;
```

The results grid shows 26 rows of data with columns: SupplierName, SupplierCategoryName, ContactType, Contact, Phone, Email. Some sample data rows are:

SupplierName	SupplierCategoryName	ContactType	Contact	Phone	Email
A Datum Corporation	Novelty Goods Supplier	Alternate Contact	Oliver Kiri	(847) 555-0100	oily@adatum.com
A Datum Corporation	Novelty Goods Supplier	Primary Contact	Reiko Kabin	(847) 555-0100	reoi@adatum.com
Consolidated Messenger	Courier Services Supplier	Alternate Contact	Helen Ahven	(415) 555-0100	helen@consolidatedmessenger.com
Consolidated Messenger	Courier Services Supplier	Primary Contact	Kerstin Parvi	(415) 555-0100	kerstin@consolidatedmessenger.com
Contoso, Ltd.	Novelty Goods Supplier	Alternate Contact	Paulus Lippmaa	(360) 555-0100	paulusj@contoso.com
Contoso, Ltd.	Novelty Goods Supplier	Primary Contact	Hanna Mihalov	(360) 555-0100	hannam@contoso.com
Fabrikam, Inc.	Clothing Supplier	Alternate Contact	Helen Moore	(203) 555-0104	helene@fabrikam.com
Fabrikam, Inc.	Clothing Supplier	Primary Contact	Bill Lawson	(203) 555-0107	bill@fabrikam.com
Graphic Design Institute	Novelty Goods Supplier	Alternate Contact	Donna Smith	(406) 555-0107	donna@graphicdesigninstitute.com
Graphic Design Institute	Novelty Goods Supplier	Primary Contact	Penny Buck	(406) 555-0107	pennyb@graphicdesigninstitute.com

At the bottom of the SSMS window, it says "Query executed successfully." The taskbar at the bottom of the screen shows the Windows Start button, a search icon, taskbar icons for various applications, and system status icons.

Exploring existing views:

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left lists the database 'WideWorldImporters'. The central pane displays a T-SQL script for creating a view named 'Suppliers' from the 'Purchasing' schema. The script includes columns such as SupplierID, SupplierName, and various contact details. The status bar at the bottom indicates a connection to 'LAPTOP-B09T49TN\aaadar'.

```
1 USE [WideWorldImporters]
2 GO
3 **** Object: View [Website].[Suppliers] Script Date: 2/13/2022 11:39:44 AM ****
4 SET ANSI_NULLS ON
5 GO
6
7 SET QUOTED_IDENTIFIER ON
8 GO
9
10
11 CREATE VIEW [Website].[Suppliers]
12 AS
13     SELECT s.SupplierID,
14             s.SupplierName,
15             sc.SupplierCategoryName,
16             pp.FullName AS PrimaryContact,
17             ap.FullName AS AlternateContact,
18             s.PhoneNumber,
19             s.FaxNumber,
20             s.WebsiteURL,
21             dm.DeliveryMethodName AS DeliveryMethod,
22             c.CityName AS CityName,
23             s.DeliveryLocation AS DeliveryLocation,
24             s.SupplierReference
25
26     FROM Purchasing.Suppliers AS s
27     LEFT OUTER JOIN Purchasing.SupplierCategories AS sc
```

Schema Binding:

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left lists the database 'WideWorldImporters'. The central pane displays a T-SQL script that renames a column and then finds dependent objects. It uses system stored procedures like sp_rename and dm_sql_referencing_entities. The results pane shows two rows of data related to the 'GetTransactionUpdates' and 'InvoiceCustomerOrders' stored procedures.

```
76 EXEC sp_rename 'Sales.CustomerTransactions.PreTaxTotal', 'AmountExcludingTax', 'COLUMN';
77 GO
78
79 -- Find dependent objects
80 SELECT dm_sql_referencing_entities.referencing_schema_name,
81         dm_sql_referencing_entities.referencing_entity_name,
82         dm_sql_referencing_entities.referencing_id,
83         sql_modules.object_id,
84         sql_modules.definition,
85         sql_modules.is_schema_bound
86     FROM sys.dm_sql_referencing_entities ('Sales.CustomerTransactions', 'OBJECT')
87     JOIN sys.sql_modules ON dm_sql_referencing_entities.referencing_id = sql_modules.object_id
88
89 -- In order to change the table, the schemabound view must be altered or dropped
90 DROP VIEW Sales.OutstandingBalance;
```

referencing_schema_name	referencing_entity_name	object_id	definition	is_schema_bound
Integration	GetTransactionUpdates	1950629992	CREATE PROCEDURE Integration.GetTransactionUpd...	0
Website	InvoiceCustomerOrders	1307151987	CREATE PROCEDURE Website.InvoiceCustomerOrder...	0

View Object with indexes:

The screenshot shows the Microsoft SQL Server Management Studio interface. In the Object Explorer, the database 'WideWorldImporters' is selected. In the center pane, a query window titled 'SQLQuery3.sql - LAPTOP-B09T49TN.WideWorldImporters (LAPTOP-B09T49TN\aaadar (51))' contains the following T-SQL code:

```
39 -- Add an index to the view
40 CREATE UNIQUE CLUSTERED INDEX IDX_StockItemDetails
41 ON Warehouse.StockItemDetails (StockItemStockGroupID, StockItemName, SupplierID);
42 GO
43
44 -- Query the view
45 SELECT *
46 FROM Warehouse.StockItemDetails
47 INNER JOIN Purchasing.Suppliers
48 ON Suppliers.SupplierID = StockItemDetails.SupplierID
49 WHERE StockItemDetails.SupplierID = 5;
```

The results pane shows a table with 126 rows, representing the data from the StockItemDetails table filtered by SupplierID 5. The columns are: StockItemStockGroupID, StockItemName, QuantityOnHand, StockGroupName, ColorName, UnitPrice, and SupplierID.

Chapter Quiz

The screenshot shows a summary page for a chapter quiz. On the left, a sidebar lists completed sections with green checkmarks:

- 1. Create Views of the Data
 - Save a query as a view object
 - Explore existing views
 - SchemaBind a view
 - Leverage view objects with indexes
 - Chapter Quiz (4 questions)
- 2. Create User-Defined Functions
 - Deterministic vs. nondeterministic functions
 - Scalar-valued user-defined functions
 - IF ELSE and CASE statements
 - Table-valued user-defined functions

The main content area displays the following message:
You answered 4 of 4 questions correctly.
You successfully completed all questions in this quiz.

Buttons at the bottom include 'Review all answers' and 'Continue watching'.

Chapter 2: Create User defined functions

Deterministic VS Non deterministic function

SQLQuery3.sql - LAPTOP-B09T49TN.WideWorldImporters (LAPTOP-B09T49TN\aadarr (51)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

New Query Execute

WideWorldImporters

Object Explorer

SQLQuery3.sql - LA_9T49TN\aadarr (51) * X

```
13     MIN(Temperature) AS 'Minimum Temp',
14     AVG(Temperature) AS 'Average Temp'
15   FROM Warehouse.ColdRoomTemperatures_Archive;
16
17  -- Some functions require multiple parameters or arguments
18 =SELECT DISTINCT TOP 10
19     InvoiceDate,
20     FORMAT(InvoiceDate, 'd') AS FormattedDate -- standard date formatting for local culture
21   FROM Sales.Invoices;
22
23  -- Functions can be nested together
24 =SELECT FORMAT(GETDATE(), 'MMMM dd, yyyy "at" hh:mm');
```

Results # Messages

	InvoiceDate	FormattedDate
1	2013-01-01	1/1/2013
2	2013-01-02	1/2/2013
3	2013-01-03	1/3/2013
4	2013-01-04	1/4/2013
5	2013-01-05	1/5/2013
6	2013-01-07	1/7/2013
7	2013-01-08	1/8/2013
8	2013-01-09	1/9/2013

Query executed successfully.

LAPTOP-B09T49TN (15.0 RTM) | LAPTOP-B09T49TN\aadarr ... | WideWorldImporters | 00:00:00 | 10 rows

Ln 18 Col 1 Ch 1 INS

Ready

Scalar Valued Used-defined functions

SQLQuery3.sql - LAPTOP-B09T49TN.WideWorldImporters (LAPTOP-B09T49TN\aadarr (51)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

New Query Execute

WideWorldImporters

Object Explorer

SQLQuery3.sql - LA_9T49TN\aadarr (51) * X

```
19     SELECT Application.SquareNumber(3) AS Result;
20
21  -- Use the function with data from the database
22 =SELECT TOP 10 OrderID AS 'A Number',
23     Application.SquareNumber(OrderID) AS 'The Number Squared'
24   FROM Sales.Orders;
25
26  -- Test your function for bugs
27 =SELECT Application.SquareNumber(5.9) AS Result;
28
29  -- Remove the function from the database
30 =DROP FUNCTION Application.SquareNumber;
31 GO
```

Results # Messages

A Number	The Number Squared
1	1
2	256
3	324
4	961
5	2025
6	3249
7	10404
8	25600
9	28224
10	28561

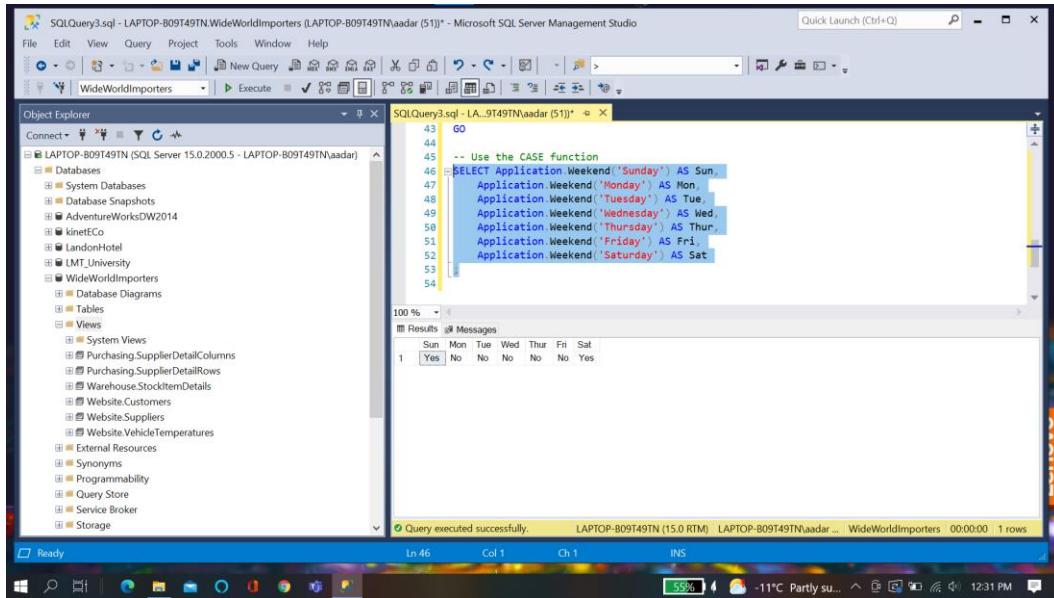
Query executed successfully.

LAPTOP-B09T49TN (15.0 RTM) | LAPTOP-B09T49TN\aadarr ... | WideWorldImporters | 00:00:00 | 10 rows

Ln 22 Col 1 Ch 1 INS

Ready

If Else Case Statement:

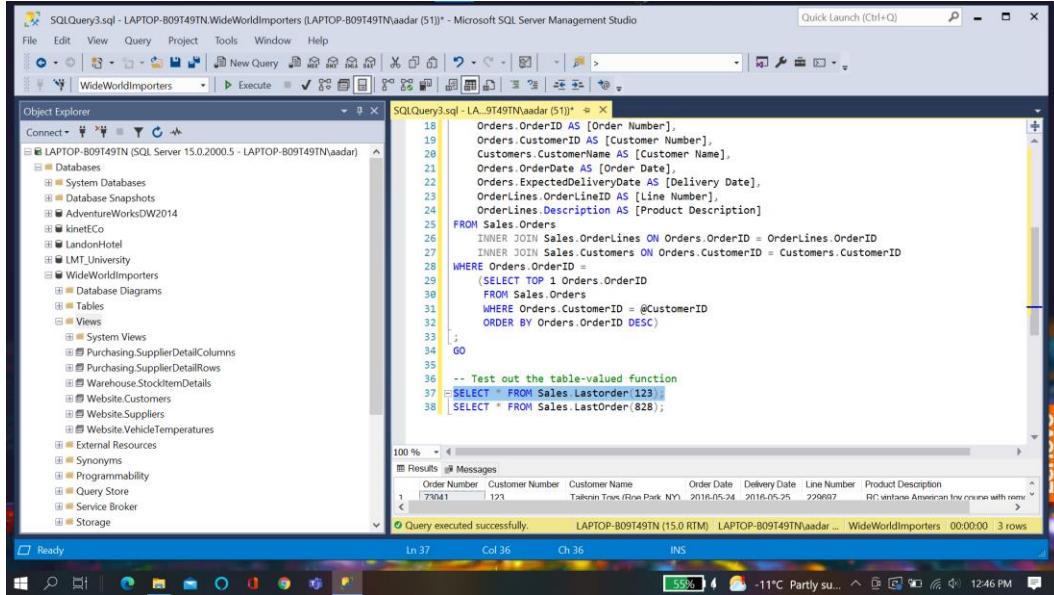


```
-- Use the CASE function
SELECT Application Weekend('Sunday') AS Sun,
       Application Weekend('Monday') AS Mon,
       Application Weekend('Tuesday') AS Tue,
       Application Weekend('Wednesday') AS Wed,
       Application Weekend('Thursday') AS Thur,
       Application Weekend('Friday') AS Fri,
       Application Weekend('Saturday') AS Sat
```

	Sun	Mon	Tue	Wed	Thu	Fri	Sat
1	Yes	No	No	No	No	No	Yes

Query executed successfully.

User Defined Table Valued function:



```
SELECT * FROM Sales.LastOrder 123;
SELECT * FROM Sales.LastOrder 828;
```

Order Number	Customer Number	Customer Name	Order Date	Delivery Date	Line Number	Product Description
73941	123	Tallinn Tree (Ree Park, NY)	2016-05-24	2016-05-25	299697	RC venture American low profile with new

Query executed successfully.

Challenge:

The screenshot shows the Microsoft SQL Server Management Studio interface. In the Object Explorer, the 'WideWorldImporters' database is selected. In the center pane, a query window titled 'SQLQuery3.sql - [LA...9T49TN\adar (51)]*' contains the following T-SQL code:

```
1 CREATE OR ALTER FUNCTION Warehouse.TempSummary (@InputNumber decimal(10,2))
2 RETURNS char(10)
3 AS
4 BEGIN
5     DECLARE @Output char(10);
6     SET @Output =
7         CASE WHEN @InputNumber < 3.5 THEN 'TOO COLD'
8             WHEN @InputNumber > 4 THEN 'TOO HOT'
9             ELSE 'JUST RIGHT'
10    END;
11    RETURN @Output;
12 END;
13 GO
14
15 SELECT ColdRoomSensorNumber AS InputNumber, ColdRoomSensorNumber, Temperature,
16        Warehouse.TempSummary(Temperature) AS 'Summary'
17 FROM Warehouse.ColdRoomTemperatures;
18
19 GO
```

The results pane shows the output of the query:

InputNumber	ColdRoomSensorNumber	Temperature	Summary
1	1	3.72	JUST RIGHT
2	2	4.70	TOO HOT
3	3	3.00	TOO COLD
4	4	3.58	JUST RIGHT

At the bottom of the screen, the taskbar shows the system status: 55%, -11°C, Partly su..., 1:14 PM.

Chapter Quiz:

The screenshot shows a summary page for a chapter quiz. On the left, there is a sidebar with a 'Contents' section listing various topics with their duration: Scalar-valued user-defined functions (6m 4s), IF ELSE and CASE statements (6m 42s), Table-valued user-defined functions (5m 18s), Challenge: Create a function (4m 45s), Solution: Create a function (1m 6s), and Chapter Quiz (9m 28s). Below this is a section for '3. Work with Stored Procedures' with topics: Write and execute a stored procedure (7m 43s), Stored procedure input parameters (7m 38s), Stored procedure output parameters (3m 38s), Create a database trigger (3m 37s), and Challenge: Use function in a stored (7m 38s).

The main area displays the following message: "You answered 4 of 4 questions correctly. You successfully completed all questions in this quiz." It includes two buttons: "Review all answers" and "Continue watching". There is also a decorative illustration of a person sitting at a desk with a computer monitor displaying a star and several certificates or awards hanging on the wall behind them.

Chapter 3: Stored Procedure

Execute Stored procedure

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the database structure for 'WideWorldImporters'. The central pane displays a query window titled 'SQLQuery3.sql' containing the following T-SQL code:

```
28 GO
29 -- Stored procedures can perform multiple tasks
30 CREATE OR ALTER PROCEDURE Application.uspViewData
31 AS
32 SELECT TOP 1 * FROM Application.People;
33 SELECT TOP 1 * FROM Sales.Customers;
34 SELECT TOP 1 * FROM Warehouse.Colors;
35 GO
36
37 -- Execute multi-task stored procedure
38 EXECUTE Application.uspViewData;
39 GO
```

The results pane shows three tables of data:

CustomerID	CustomerName	BillToCustomerID	CustomerCategoryID	BuyingGroupId	PrimaryContactPersonID	AlternateContactPersonID	DeliveryMethodID
1	Talspin Toys (Head Office)	1	3	1	1001	1002	3

ColorID	ColorName	LastEditedBy	ValidFrom	ValidTo
1	Azure	1	2013-01-01 00:00:00.0000000	9999-12-31 23:59:59.9999999

At the bottom of the results pane, it says 'Query executed successfully.'

Stored Procedure Output Parameter:

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the database structure for 'WideWorldImporters'. The central pane displays a query window titled 'SQLQuery3.sql' containing the following T-SQL code:

```
4
5 -- Create an output variable to pass a message to the calling application
6 CREATE OR ALTER PROCEDURE Application.uspSimpleProcedure (@OutputMessage AS nvarchar(200) OUTPUT)
7 AS
8 SET @OutputMessage = N'This message was returned by the stored procedure on ' + FORMAT(GETDATE(), 'd')
9
10 GO
11
12 DECLARE @MyLocalMessage nvarchar(200)
13 EXEC Application.uspSimpleProcedure
14 @OutputMessage = @MyLocalMessage OUTPUT,
15 PRINT @MyLocalMessage
16 GO
```

The results pane shows the output message:

This message was returned by the stored procedure on 2/13/2022.

Completion time: 2022-02-13T15:20:39.1055446-05:00

At the bottom of the results pane, it says 'Query executed successfully.'

Trigger:

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the database structure for 'WideWorldImporters'. A query window titled 'SQLQuery3.sql' contains the following T-SQL code:

```
38 |     WHERE ColorID = @ColorID
39 |     ORDER BY ColorID DESC;
40 |
41 | GO
42 |
43 | -- Use the stored procedure to add a color to the table
44 | EXEC Warehouse.uspInsertColor 'Banana Yellow';
45 |
46 | -- The table's trigger inserts a new row to the audit table
47 | SELECT * FROM Warehouse.ColorAudit;
```

The results pane shows a single row inserted into the 'Warehouse.ColorAudit' table:

AuditID	ColorName	TimeAdded
1	Banana Yellow	2022-02-13 15:28:54.2900000

At the bottom, the status bar indicates 'Query executed successfully.'

Challenge:

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the database structure for 'WideWorldImporters'. A query window titled 'SQLQuery3.sql' contains the following T-SQL code:

```
31 | INSERT INTO Sales.CustomerAccountAudit (CustomerID, ReviewDate)
32 |     VALUES (@Customer, GETDATE());
33 |
34 | GO
35 |
36 | -- Test the stored procedure
37 | EXEC Sales.SalesInfo 915;
38 | EXEC Sales.SalesInfo 874;
39 |
40 | -- Review the audit table
41 | SELECT * FROM Sales.CustomerAccountAudit;
```

The results pane shows a single row inserted into the 'Sales.CustomerAccountAudit' table:

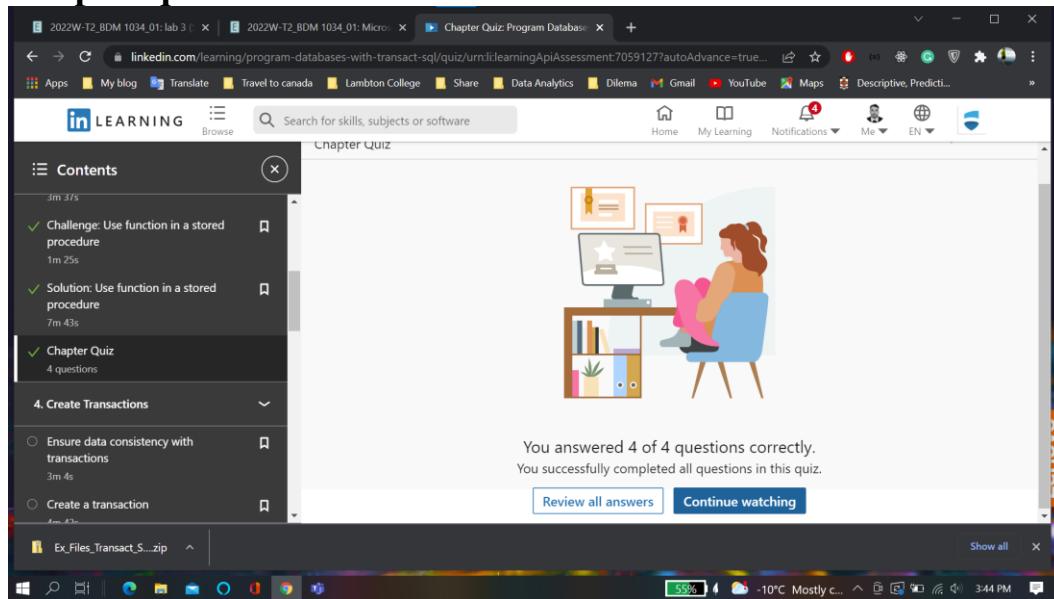
CustomerID	CustomerName	PhoneNumber
874	Daniel Martensson	(225) 555-0100

Below it, another results pane shows multiple rows inserted into the 'Sales.CustomerAccountAudit' table:

OrderID	CustomerID	ReviewDate
1	852	2013-01-15
2	1636	2013-01-31
3	2690	2013-02-26
4	2973	2013-03-04
5	3371	2013-03-11
6	4129	2013-03-22
7	4270	2013-03-26
8	5195	2013-04-12
9	5282	2013-04-15

At the bottom, the status bar indicates 'Query executed successfully.'

Chapter quiz:



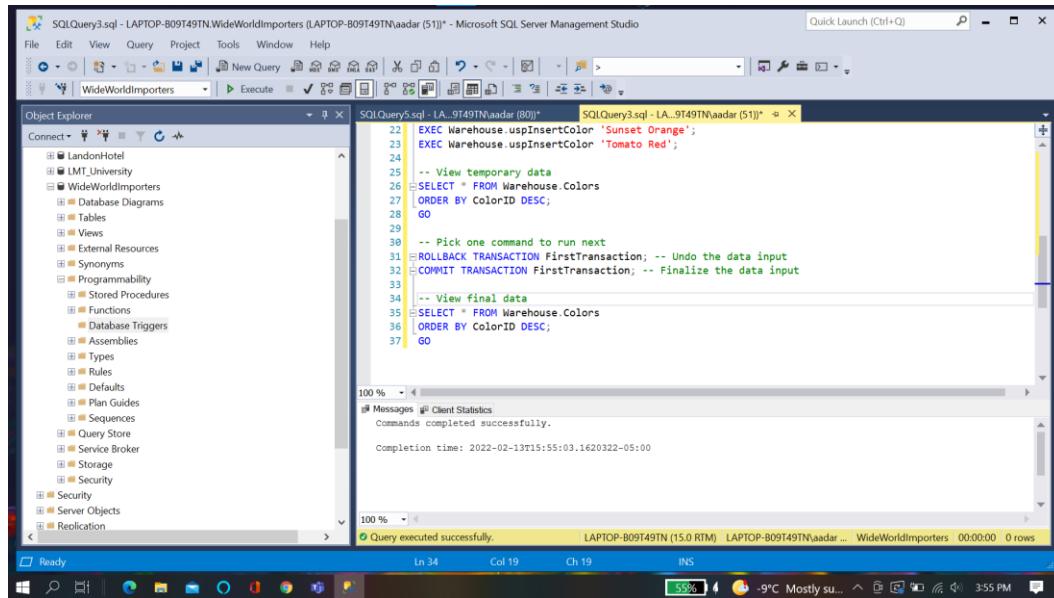
A screenshot of a Microsoft Edge browser window showing the LinkedIn Learning platform. The user has completed a chapter quiz with 4 questions, answered correctly. The quiz content includes sections like 'Challenge: Use function in a stored procedure' and 'Create Transactions'. A cartoon illustration of a person sitting at a desk with a computer monitor is displayed above the results.

You answered 4 of 4 questions correctly.
You successfully completed all questions in this quiz.

[Review all answers](#) [Continue watching](#)

Chapter 4: Transactions

Create Transaction



A screenshot of Microsoft SQL Server Management Studio (SSMS) showing a query window titled 'SQLQuery3.sql - LAPTOP-B09T49TN.WideWorldImporters (LAPTOP-B09T49TN\aadar (51))'. The code in the window demonstrates a transaction with two insert statements into the 'Colors' table, a select statement to view the data, and a rollback command to undo the changes.

```
22 EXEC Warehouse.uspInsertColor 'Sunset Orange';
23 EXEC Warehouse.uspInsertColor 'Tomato Red';
24
25 -- View temporary data
26 SELECT * FROM Warehouse.Colors
27 ORDER BY ColorID DESC;
28 GO
29
30 -- Pick one command to run next
31 ROLLBACK TRANSACTION FirstTransaction; -- Undo the data input
32 COMMIT TRANSACTION FirstTransaction; -- Finalize the data input
33
34 -- View final data
35 SELECT * FROM Warehouse.Colors
36 ORDER BY ColorID DESC;
37 GO
```

Completion time: 2022-02-13T15:55:03.1620322-05:00

Query executed successfully.

Transactions Save point:

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows databases like LandonHotel, LMT_University, and WideWorldImporters. The SQL Query window displays the following T-SQL code:

```
28 | ORDER BY ColorID DESC;
29 | GO
30 |
31 | -- Revert to the savepoint
32 | ROLLBACK TRANSACTION SavePointOne;
33 |
34 | -- View the number of open transactions
35 | SELECT @@TRANCOUNT AS 'Open Transactions';
36 |
37 | -- Commit the transaction
38 | COMMIT TRANSACTION;
39 |
40 | -- View final data
41 | SELECT * FROM Warehouse.Colors
42 | ORDER BY ColorID DESC;
43 | GO
```

The Results pane shows the output of the query, which includes 40 rows of color data from the Warehouse.Colors table, ordered by ColorID.

Automatic Rollback Transactions

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows databases like LandonHotel, LMT_University, and WideWorldImporters. The SQL Query window displays the following T-SQL code:

```
31 | EXEC Warehouse.uspInsertColor 'Glittering Gold';
32 | EXEC Warehouse.uspInsertColor 'Glittering Gold';
33 |
34 | COMMIT TRANSACTION;
35 |
36 | -- View data
37 | SELECT * FROM Warehouse.Colors
38 | ORDER BY ColorID DESC;
39 |
40 | GO
41 |
42 | -- Disable XACT_ABORT
43 | SET XACT_ABORT OFF;
```

The Results pane shows the output of the query, which includes 41 rows of color data from the Warehouse.Colors table, ordered by ColorID.

Challenge:

The screenshot shows the Microsoft SQL Server Management Studio interface. In the Object Explorer, the database 'WideWorldImporters' is selected. In the center pane, a query window titled 'SQLQuery3.sql - (A_9149TN\vaadar (51))' contains the following T-SQL code:

```
31 SET XACT_ABORT OFF;
32 ;
33 GO
34
35 EXEC dbo.TransferFunds 1, 3, 50.00;
36
37 SELECT * FROM dbo.BankAccounts;
```

The results pane shows a table with three rows:

AccountID	Balance
1	50.00
2	200.00
3	350.00

At the bottom of the results pane, it says 'Query executed successfully.' The status bar at the bottom right indicates 'LAPTOP-B09T49TN (15.0 RTM) LAPTOP-B09T49TN\vaadar... WideWorldImporters 00:00:00 3 rows'.

Quiz:

The screenshot shows a LinkedIn Learning quiz completion summary for the 'Program Databases with Transact-SQL' chapter. The quiz has 4 questions and was completed correctly. The summary includes a cartoon illustration of a person sitting at a desk with a computer monitor displaying a star.

Program Databases with Transact-SQL
Chapter Quiz

You answered 4 of 4 questions correctly.
You successfully completed all questions in this quiz.

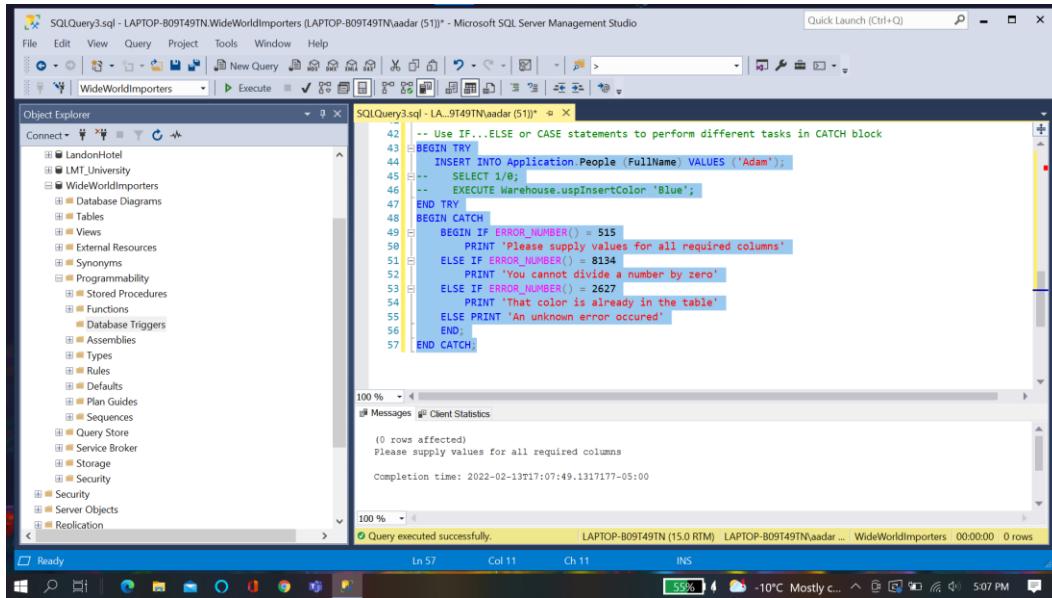
[Review all answers](#) [Continue watching](#)

The left sidebar shows the course structure:

- 3m 46s
- Automatically roll back transactions
- 4m 44s
- Challenge: Create a transaction
- 1m 10s
- Solution: Create a transaction
- 8m 50s
- Chapter Quiz** (4 questions)
- 5. Implement Error Handling
- What is error handling?
- 1m 52s
- Capture errors with TRY and CATCH
- 4m 54s
- Generate errors with THROW

Chapter 5: Error Handling

Errors with try catch:

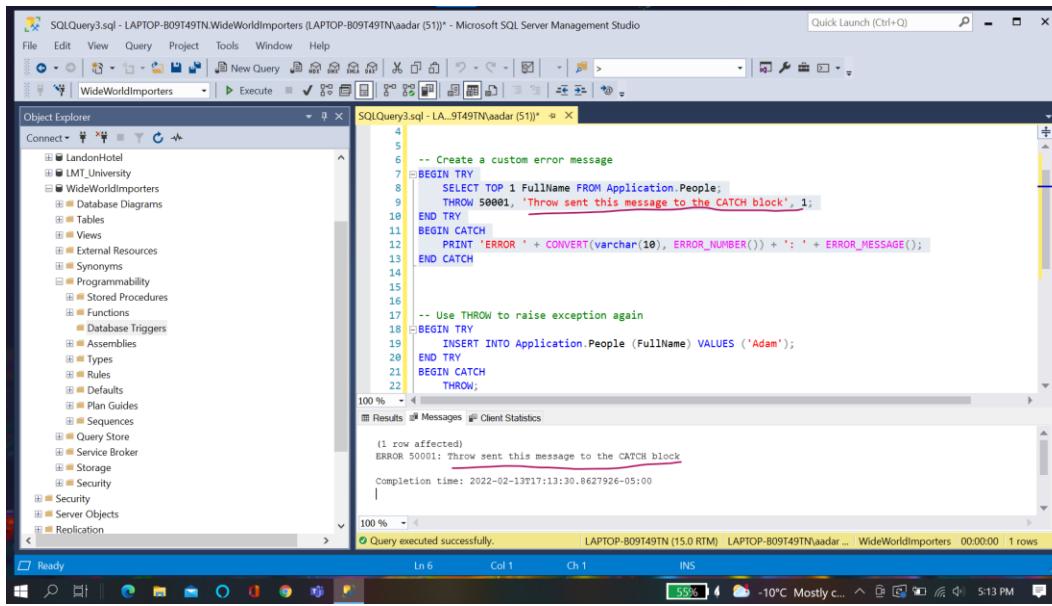


```
-- Use IF...ELSE or CASE statements to perform different tasks in CATCH block
BEGIN TRY
    INSERT INTO Application_People (FullName) VALUES ('Adam');
    SELECT 1/0;
    EXECUTE Warehouse.uspInsertColor 'Blue';
END TRY
BEGIN CATCH
    BEGIN IF ERROR_NUMBER() = 515
        PRINT 'Please supply values for all required columns';
    ELSE IF ERROR_NUMBER() = 8334
        PRINT 'You cannot divide a number by zero!';
    ELSE IF ERROR_NUMBER() = 2627
        PRINT 'That color is already in the table!';
    ELSE PRINT 'An unknown error occurred.';
    END;
END CATCH;
```

(0 rows affected)
Please supply values for all required columns
Completion time: 2022-02-13T17:07:49.1317177-05:00

Query executed successfully.

Errors with throw:



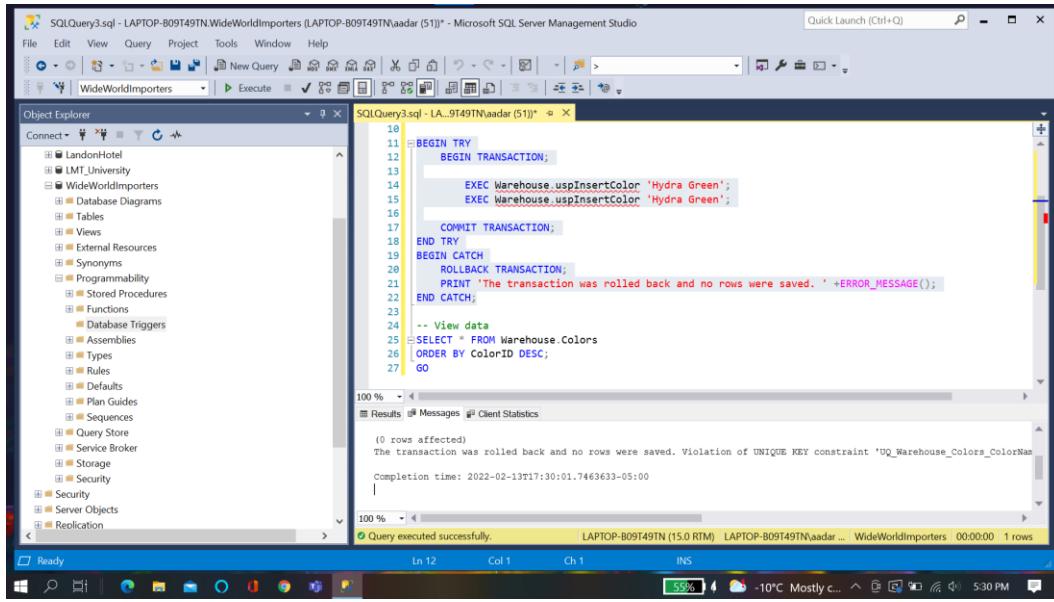
```
-- Create a custom error message
BEGIN TRY
    SELECT TOP 1 FullName FROM Application.People;
    THROW 50001, 'Throw sent this message to the CATCH block', 1;
END TRY
BEGIN CATCH
    PRINT 'ERROR ' + CONVERT(varchar(10), ERROR_NUMBER()) + ':' + ERROR_MESSAGE();
END CATCH

-- Use THROW to raise exception again
BEGIN TRY
    INSERT INTO Application.People (FullName) VALUES ('Adam');
END TRY
BEGIN CATCH
    THROW;
END CATCH;
```

(1 row affected)
ERROR 50001: Throw sent this message to the CATCH block
Completion time: 2022-02-13T17:13:30.8627926-05:00

Query executed successfully.

Transaction Control:

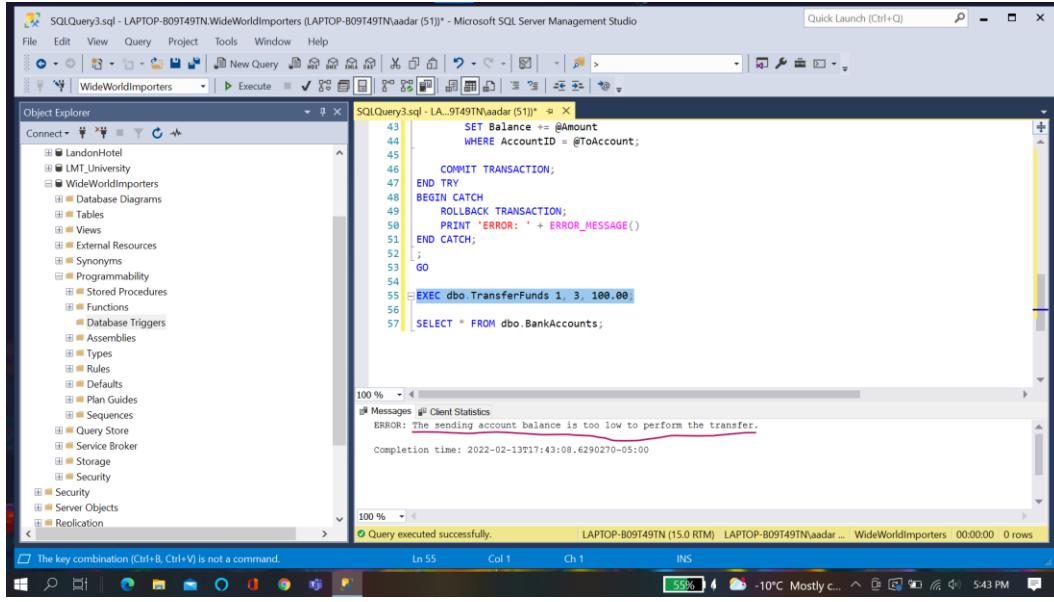


The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left lists databases: LandonHotel, LMT_University, and WideWorldImporters. The WideWorldImporters database is selected. The center pane displays a query window titled 'SQLQuery3.sql - LAPTOP-B09T49TN.WideWorldImporters (51)*'. The code in the query window is as follows:

```
10 BEGIN TRY
11     BEGIN TRANSACTION;
12
13         EXEC Warehouse.uspInsertColor 'Hydra Green';
14         EXEC Warehouse.uspInsertColor 'Hydra Green';
15
16         COMMIT TRANSACTION;
17     END TRY
18     BEGIN CATCH
19         ROLLBACK TRANSACTION;
20         PRINT 'The transaction was rolled back and no rows were saved.' + ERROR_MESSAGE();
21     END CATCH;
22
23
24     -- View data
25     SELECT * FROM Warehouse.Colors
26     ORDER BY ColorID DESC;
27 GO
```

The results pane below the code shows the output of the query. It indicates '0 rows affected' and 'The transaction was rolled back and no rows were saved. Violation of UNIQUE KEY constraint 'UQ_Warehouse_Colors_colorName''. The status bar at the bottom right shows 'Query executed successfully.'

Challenge:

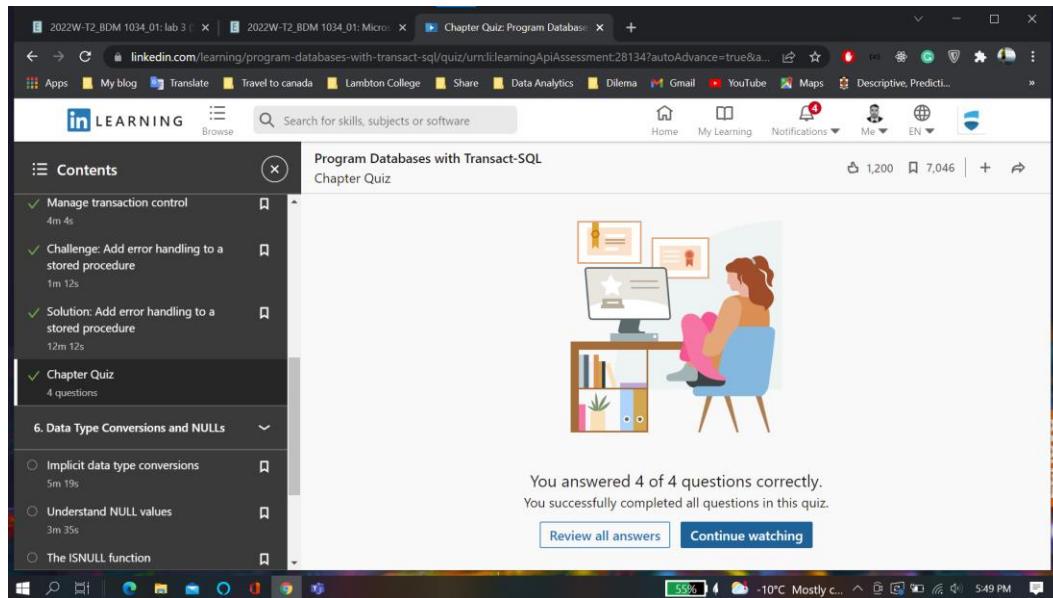


The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left lists databases: LandonHotel, LMT_University, and WideWorldImporters. The WideWorldImporters database is selected. The center pane displays a query window titled 'SQLQuery3.sql - LAPTOP-B09T49TN.WideWorldImporters (51)*'. The code in the query window is as follows:

```
43     SET Balance += @Amount
44     WHERE AccountID = @ToAccount;
45
46     COMMIT TRANSACTION;
47 END TRY
48 BEGIN CATCH
49     ROLLBACK TRANSACTION;
50     PRINT 'ERROR: ' + ERROR_MESSAGE();
51 END CATCH;
52 ;
53 GO
54 EXEC dbo.TransferFunds 1, 3, 100.00;
55
56 SELECT * FROM dbo.BankAccounts;
```

The results pane below the code shows the output. It includes the message 'ERROR: The sending account balance is too low to perform the transfer.' The status bar at the bottom right shows 'Query executed successfully.'

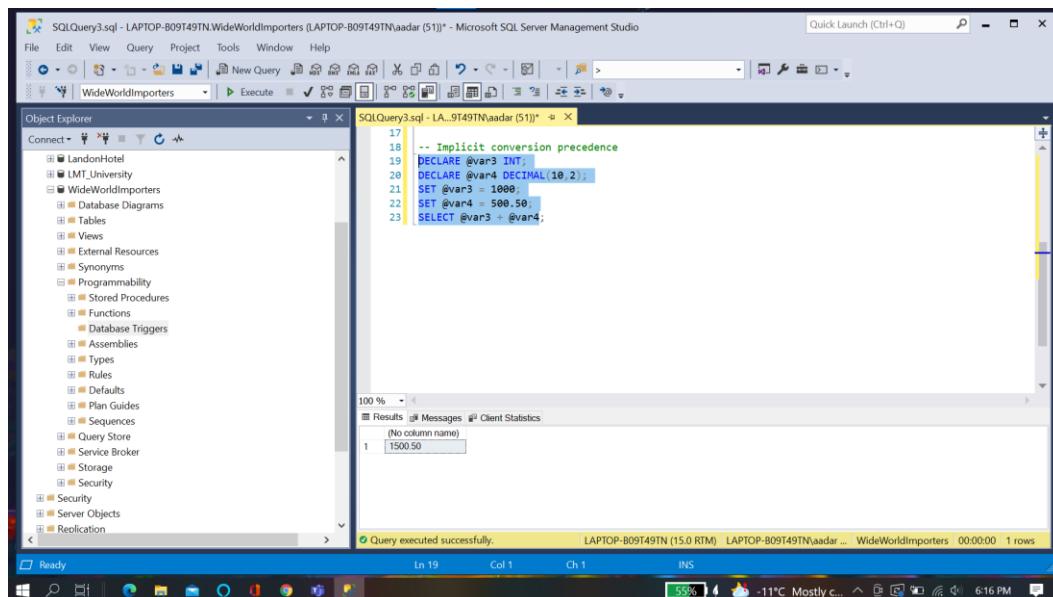
Chapter Quiz:



A screenshot of a Microsoft Windows desktop showing a browser window for LinkedIn Learning. The page displays the results of a 'Chapter Quiz: Program Databases with Transact-SQL'. The quiz contained 4 questions, all of which were answered correctly. The user successfully completed all questions in this quiz. There are two buttons at the bottom: 'Review all answers' and 'Continue watching'. The browser toolbar shows several tabs open, including 'Chapter Quiz: Program Databases with Transact-SQL' and 'Chapter Quiz: Program Databases with Transact-SQL (28134)'. The taskbar at the bottom shows various pinned icons and the system tray.

Chapter 6: Data Type Conversion and NULL

Implicit Data types:



A screenshot of Microsoft SQL Server Management Studio (SSMS) showing a query results window. The query demonstrates implicit data type conversion. The code is as follows:

```
17 -- Implicit conversion precedence
18 DECLARE @var3 INT;
19 DECLARE @var4 DECIMAL(10,2);
20 SET @var3 = 1000;
21 SET @var4 = 500.50;
22
23 SELECT @var3 + @var4;
```

The results window shows the output of the query:

1	1500.50
1	1500.50

Below the results, a message states: 'Query executed successfully.' The status bar at the bottom indicates the query was run on 'LAPTOP-B09T49TN (15.0 RTM)' at '00:00:00' on 'WideWorldImporters' database.

Null Values:

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the database structure, including the WideWorldImporters database. The main window displays a T-SQL script in the Query Editor:

```
13     END AS 'Equality';
14
15
16
17 -- Consider avoiding nulls by adding an excessively out-of-range default value to table designs
18 CREATE TABLE Birthdays (
19     PersonName nvarchar(100) NOT NULL,
20     Birthday date NOT NULL DEFAULT '1/1/1800'
21 )
22 GO
23 INSERT INTO Birthdays (PersonName)
24     VALUES ('Bethany');
25 INSERT INTO Birthdays (PersonName, Birthday)
26     VALUES ('Jin', '5/17/1982');
27 GO
28 SELECT * FROM Birthdays;
```

The Results pane shows the output of the query:

PersonName	Birthday
Bethany	1800-01-01
Jin	1982-05-17

At the bottom, it says "Query executed successfully."

IsNull Function

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the database structure, including the WideWorldImporters database. The main window displays a T-SQL script in the Query Editor:

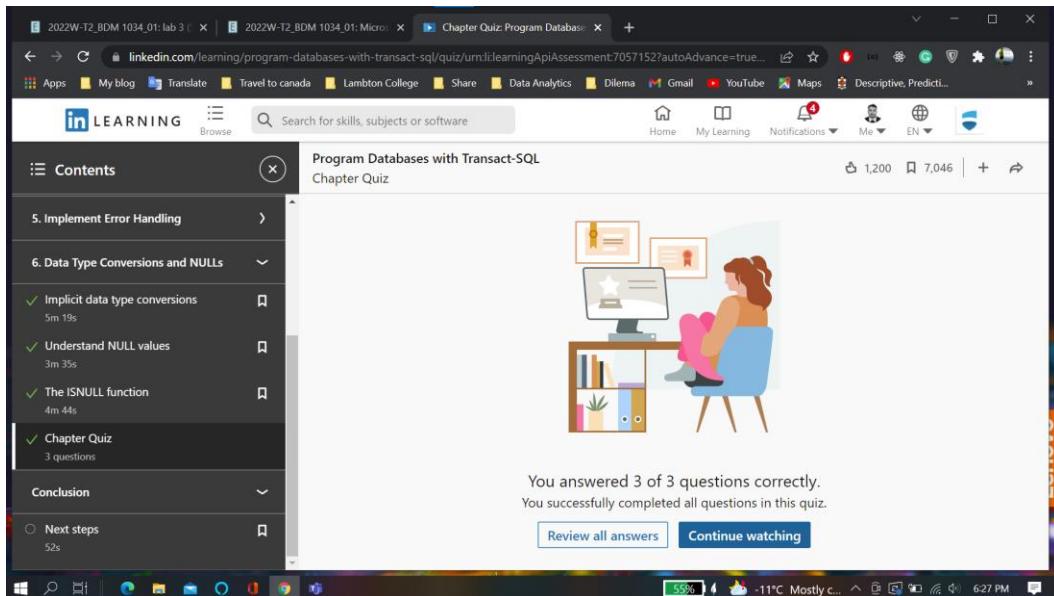
```
39
40
41 -- Instead, perform the substitution in the SELECT clause
42 SELECT Suppliers.SupplierID,
43        Suppliers.SupplierName,
44        Suppliers.DeliveryMethodID,
45        ISNULL(DeliveryMethods.DeliveryMethodName, 'Post') AS DeliveryMethodName
46 FROM Purchasing.Suppliers
47 LEFT JOIN Application.DeliveryMethods
48 ON Suppliers.DeliveryMethodID = DeliveryMethods.DeliveryMethodID;
```

The Results pane shows the output of the query:

SupplierID	SupplierName	DeliveryMethodID	DeliveryMethodName
1	A Datum Corporation	7	Road Freight
2	Contoso, Ltd.	9	Refrigerated Road Freight
3	Consolidated Messenger	NULL	Post
4	Fabrikam, Inc.	7	Road Freight
5	Graphic Design Institute	10	Refrigerated Air Freight
6	Humongous Insurance	NULL	Post
7	Litware, Inc.	2	Courier
8	Lucerne Publishing	10	Refrigerated Air Freight
9	No! Publishers	10	Refrigerated Air Freight
10	Northeast Electric Cars	8	Air Freight
11	Trey Research	NULL	Post
12	The Phone Company	7	Road Freight
13	Woodgrove Bank	NULL	Post

At the bottom, it says "Query executed successfully."

Chapter quiz:



A screenshot of a Microsoft Windows desktop showing a browser window on LinkedIn Learning. The page displays the results of a 'Chapter Quiz: Program Databases with Transact-SQL'. The quiz summary indicates 3 questions answered correctly out of 3. Below this, a message says 'You successfully completed all questions in this quiz.' There are two buttons at the bottom: 'Review all answers' and 'Continue watching'. The browser's address bar shows the URL for the quiz. The taskbar at the bottom includes icons for various applications like File Explorer, Edge, and Task View.

Course Certification:

