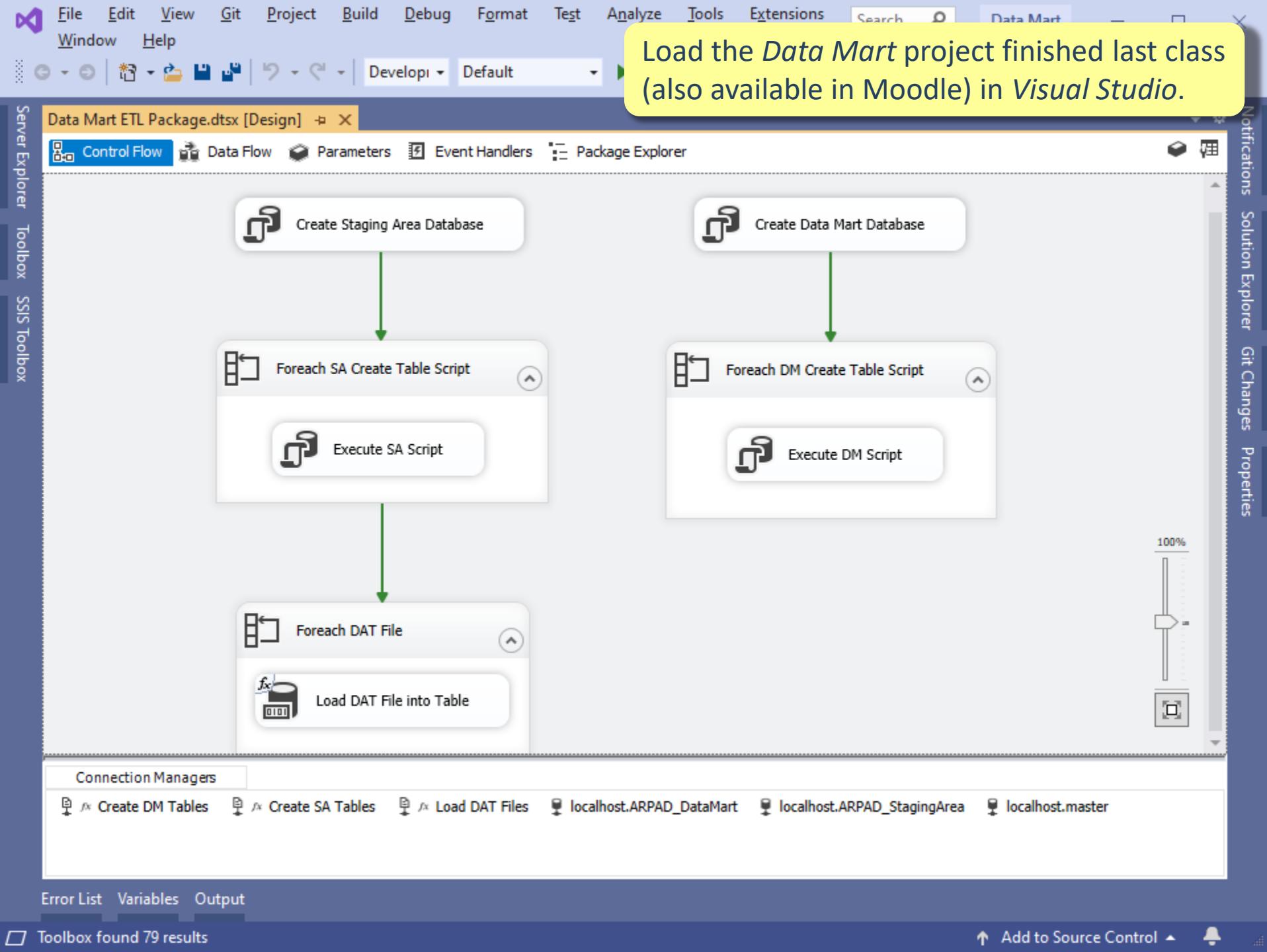


# **ARMDD**

## **EXERCISE 2 TUTORIAL**

Paulo Oliveira  
DEI-ISEP



File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Search... Data Mart

Data Mart ETL Package.dtsx [Design] Control Flow Data Flow

Server Explorer Toolbox SSIS Toolbox

Let's create four parameters for storing the:

- Server name
- Staging Area database name
- Data Mart database name
- Package path

Double click on *Project.params* to open the project parameters window.

Search Solution Explorer (Ctrl+F)

Solution 'Data Mart' (1 of 1 project)

Data Mart (SQL Server 2019)

- Project.params
- Connection Managers
- SSIS Packages
  - Data Mart ETL Package.dtsx
- Package Parts
  - Control Flow
  - Miscellaneous
- Linked Azure Resources
  - Azure-SSIS Integration Runtime
  - Azure Storage

Foreach SA Create Table Script

Execute SA Script

Foreach DM Create Table Script

Execute DM Script

Foreach DAT File

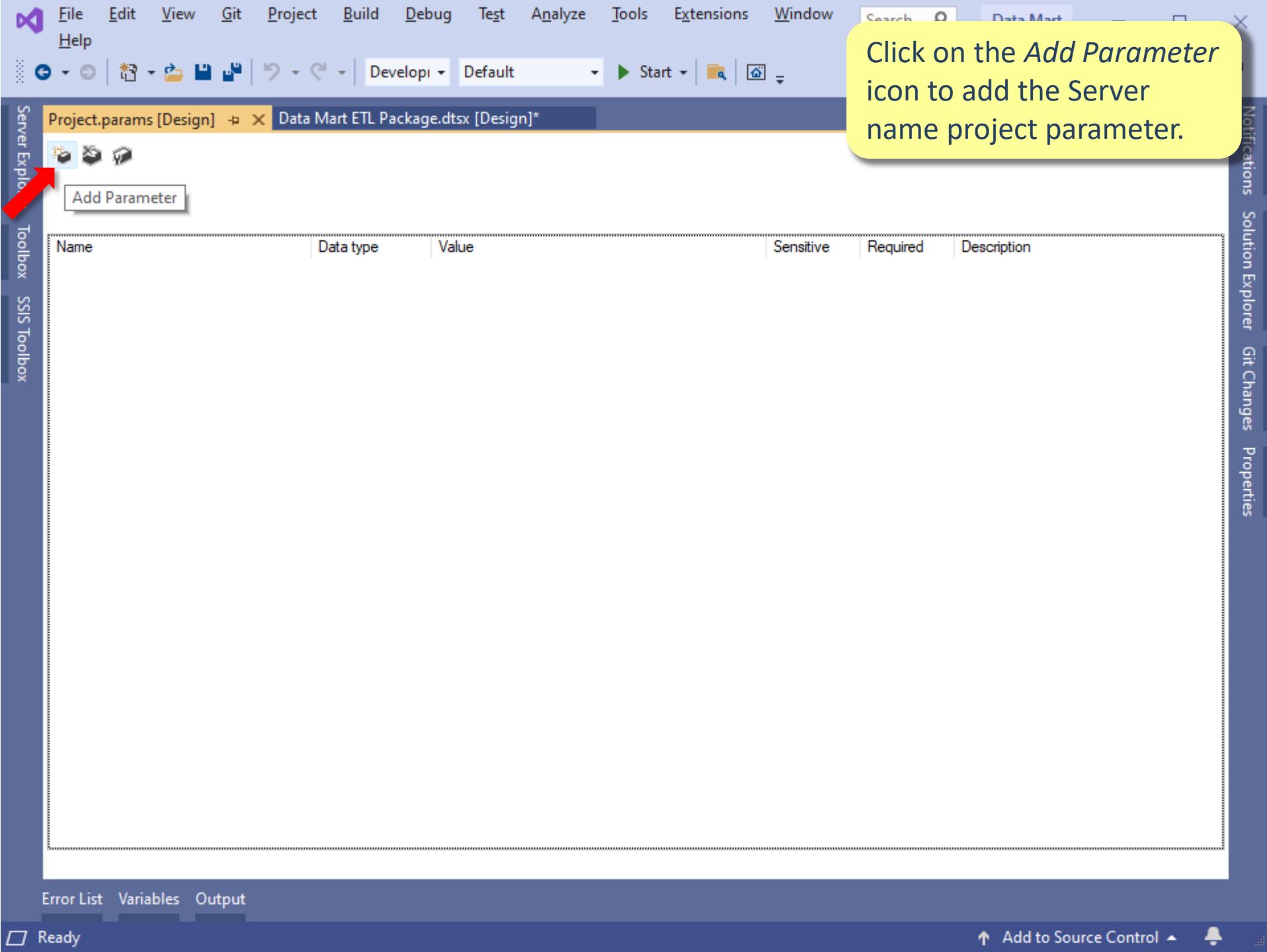
Load DAT File into Table

Connection Managers

Create DM Tables Create SA Tables Load DAT Files localhost.ARPAD\_DataMart localhost.ARPAD\_DM

Error List Variables Output

This item does not support previewing Add to Source Control



File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Data Mart

Help

Project.params [Design]\* Data Mart ETL ...e.dtsx [Design]

SSIS Toolbox Solution Explorer Properties

Output Error List Variables

Ready Add to Source Control Select Repository

Introduce the parameter's data as shown in the picture. The value (*localhost*) can be different if you haven't performed a default SQL Server installation.

Name	Data type	Value	Sensitive	Required	Description
ServerName	String	(local)	False	True	

File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Data Mart

Help

Project.params [Design]\* Data Mart ETL ...e.dtsx [Design]

SSIS Toolbox

Add another project parameter as shown in the picture, now for the Staging Area database name.

Name	Data type	Value	Sensitive	Required	Description
ServerName	String	(local)	False	True	
StagingAreaDBName	String	ARMDD_StagingArea	False	True	

Output Error List Variables

Ready Add to Source Control Select Repository

Solution Explorer Properties

File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Data Mart

Help

Project.params [Design]\* Data Mart ETL ...e.dtsx [Design]

SSIS Toolbox

Add another project parameter as shown in the picture, this time for the Data Mart database name.

Name	Data type	Value	Sensitive	Required	Description
ServerName	String	(local)	False	True	
StagingAreaDBName	String	ARMDD_StagingArea	False	True	
DataMartDBName	String	ARMDD_DataMart	False	True	

Output Error List Variables

Ready Add to Source Control Select Repository

Solution Explorer Properties

File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Data Mart Help

Project.params [Design] \* Data Mart ETL ...e.dtsx [Design]

SSIS Toolbox Solution Explorer Properties

Add yet another project parameter as shown in the picture, now for the Package path. You should use your on path to the Data Mart folder.

Name	Data type	Value	Sensitive	Required	Description
ServerName	String	(local)	False	True	
StagingAreaDBName	String	ARMDD_StagingArea	False	True	
DataMartDBName	String	ARMDD_DataMart	False	True	
PackagePath	String	C:\Temp\ARMDD\Data Mart	False	True	

Output Error List Variables

Add to Source Control Select Repository

File Edit View Git Project Build Debug Test Analyze Tools Extensions Window | Data Mart

Help

SSIS Toolbox

Save the project changes.

Project.params [Design] Save Selected Items (Ctrl+S) ssx [Design]

SSIS Toolbox

Solution Explorer Properties

Name Data type Value Sensitive Required Description

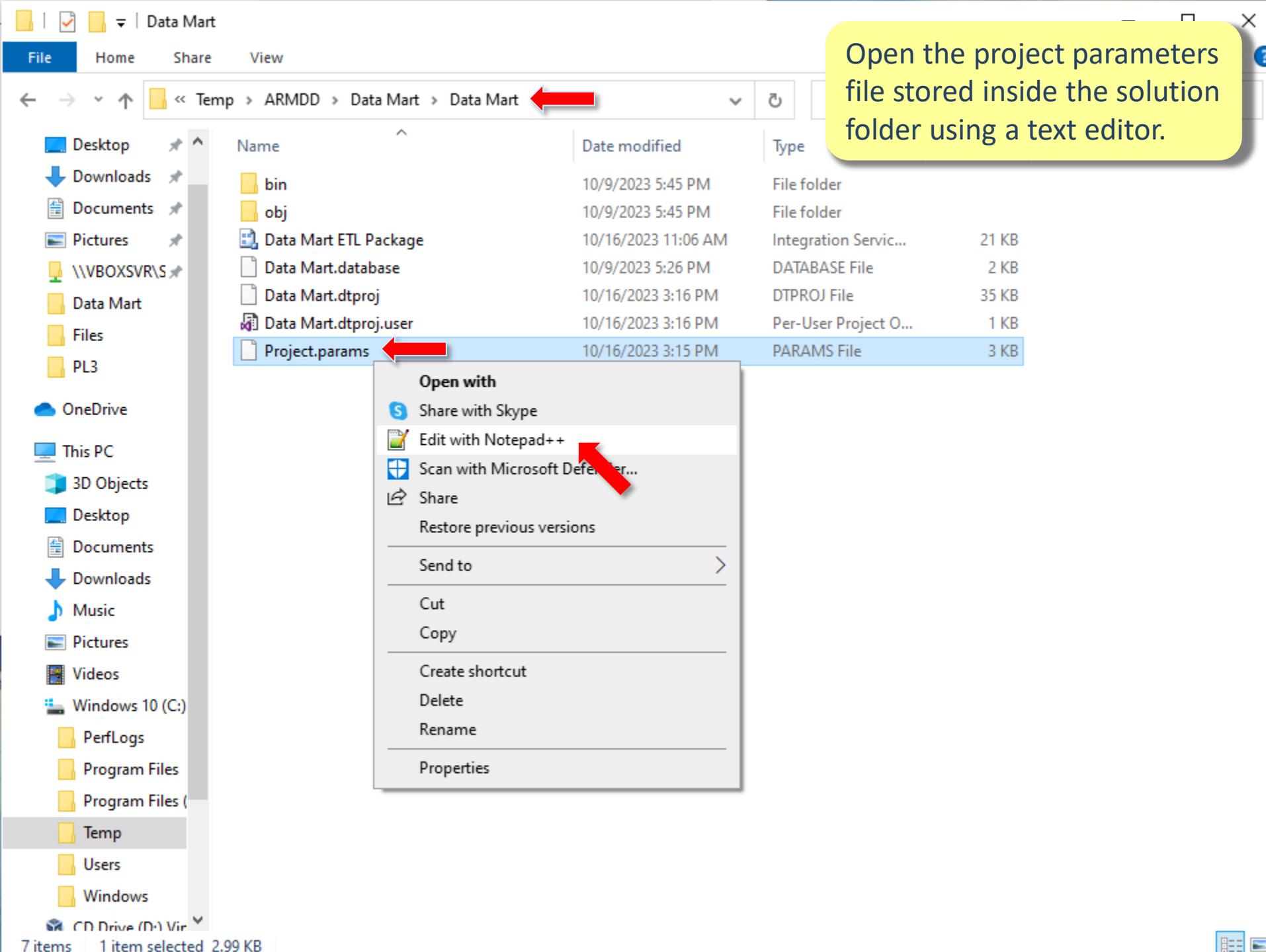
DataMartDBName	String	ARMDD_DataMart	False	True	
PackagePath	String	C:\Temp\ARMDD\Data Mart	False	True	
ServerName	String	(local)	False	True	
StagingAreaDBName	String	ARMDD_StagingArea	False	True	

Output Error List Variables

Ready Add to Source Control Select Repository

Save the project changes.

Open the project parameters file stored inside the solution folder using a text editor.



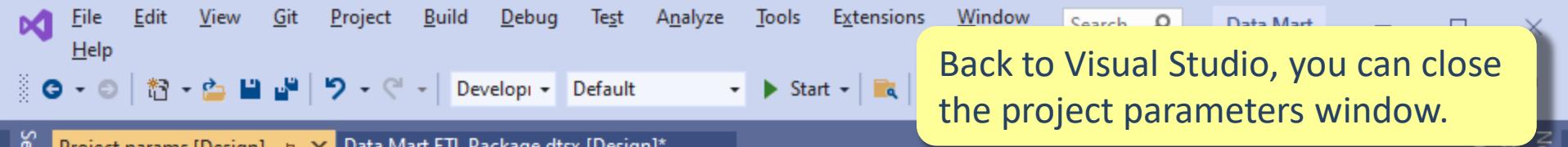
C:\Temp\ARMDD\Data Mart\Data Mart\Project.params - Notepad++

File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window

Project.params

```
<?xml version="1.0"?>
<SSIS:Parameters xmlns:SSIS="www.microsoft.com/SqlServer/SSIS">
    <SSIS:Parameter
        SSIS:Name="ServerName">
        <SSIS:Properties>
            <SSIS:Property
                SSIS:Name="ID">{9a0dbe19-4f4f-41fe-a648-8ed82871348f}</SSIS:Property>
            <SSIS:Property
                SSIS:Name="CreationName"></SSIS:Property>
            <SSIS:Property
                SSIS:Name="Description"></SSIS:Property>
            <SSIS:Property
                SSIS:Name="IncludeInDebugDump">0</SSIS:Property>
            <SSIS:Property
                SSIS:Name="Required">1</SSIS:Property>
            <SSIS:Property
                SSIS:Name="Sensitive">0</SSIS:Property>
            <SSIS:Property
                SSIS:Name="Value">(local)</SSIS:Property>
            <SSIS:Property
                SSIS:Name=" DataType">18</SSIS:Property>
        </SSIS:Properties>
    </SSIS:Parameter>
    <SSIS:Parameter
        SSIS:Name="StagingAreaDBName">
        <SSIS:Properties>
            <SSIS:Property
                SSIS:Name="ID">{60440cca-c723-4cf5-8e46-e9050a802894}</SSIS:Property>
            <SSIS:Property
                SSIS:Name="CreationName"></SSIS:Property>
            <SSIS:Property
                SSIS:Name="Description"></SSIS:Property>
            <SSIS:Property
                SSIS:Name="IncludeInDebugDump">0</SSIS:Property>
            <SSIS:Property
```

You can see the project parameters and their different properties, based on the values previously specified.



Back to Visual Studio, you can close the project parameters window.

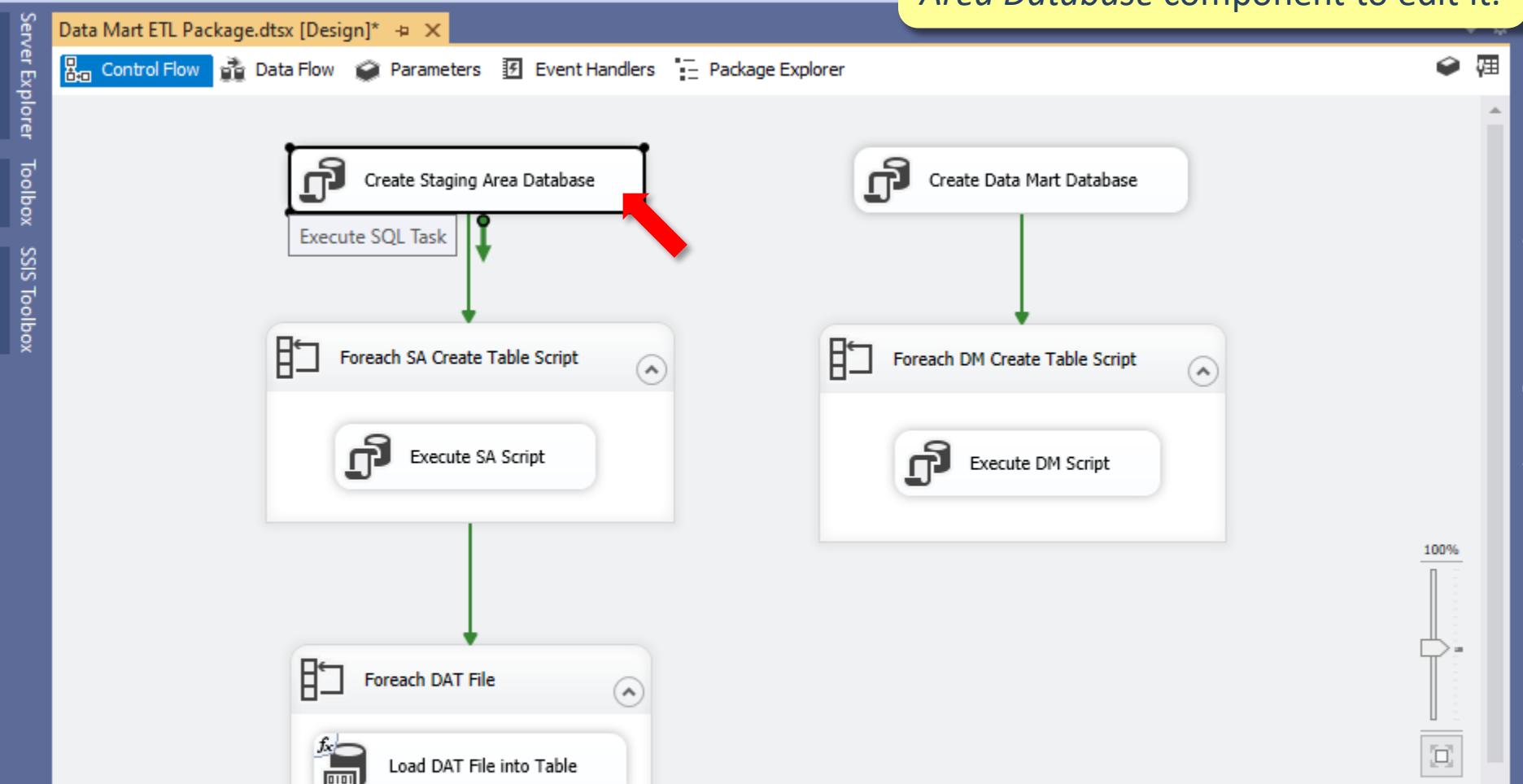
Project.params [Design]  Data Mart ETL Package.dtsx [Design]\*

 Close (Ctrl+F4)

Name	Data type	Value	Sensitive	Required	Description
ServerName	String	localhost	False	True	
StagingAreaDBName	String	ARPAD_StagingArea	False	True	
DataMartDBName	String	ARPAD_DataMart	False	True	
PackagePath	String	C:\Temp\ARPAD\Data Mart	False	True	



Double click on the *Create Staging Area Database* component to edit it.



Connection Managers

- Create DM Tables
- Create SA Tables
- Load DAT Files
- localhost.ARPAD\_DataMart
- localhost.ARPAD\_StagingArea
- localhost.master

File Edit View Git Project Build Debug Format Test Analyze Tools Extensions Search Data Mart

Window Help

Data Mart ETL Control Flow

Server Explorer Toolbox SSIS Toolbox

Notifications Solution Explorer Git Changes Properties

Execute SQL Task Editor

Configure the properties required to run SQL statements and store results.

General Parameter Mapping Result Set Expressions

General Options Result Set SQL Statement

ConnectionType	ADO.NET
Connection	
SQLSourceType	Direct input
SQLStatement	IF NOT EXISTS (SELECT name FROM sys.d
IsQueryStoredProcedure	False
BypassPrepare	False

**ConnectionType**  
Specifies the type of connection manager.

Browse... Build Query... Parse Query

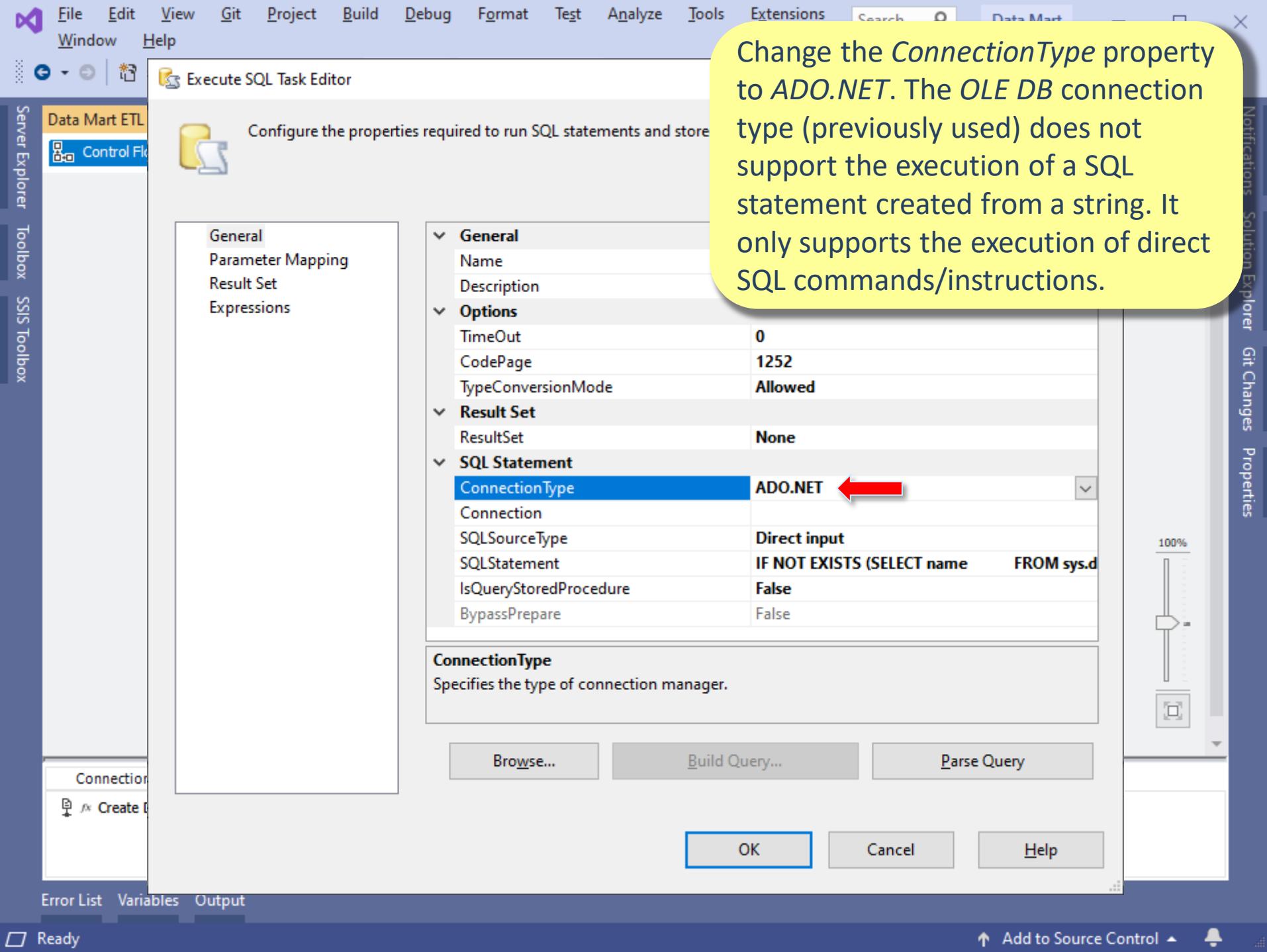
OK Cancel Help

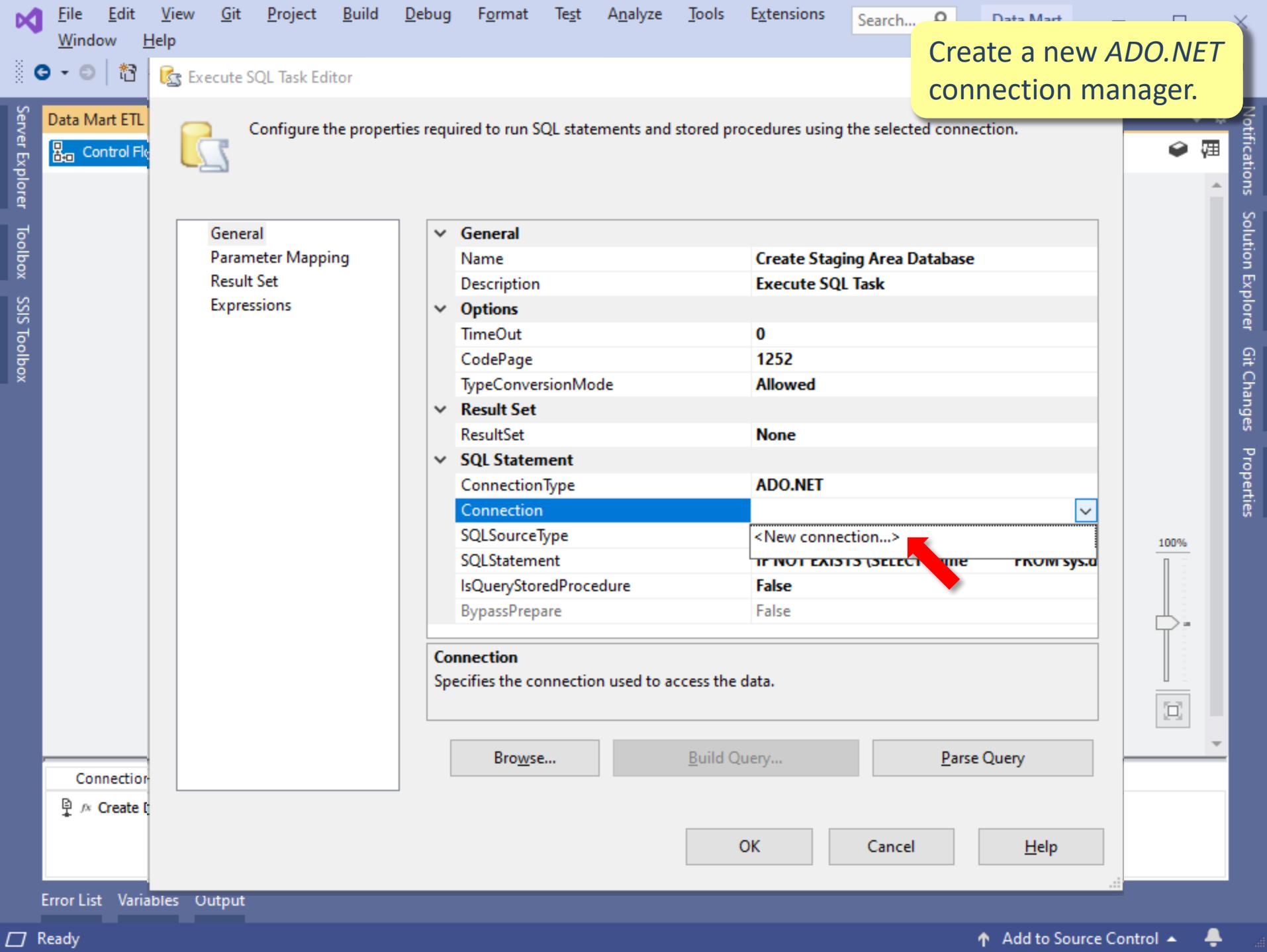
Change the *ConnectionType* property to *ADO.NET*. The *OLE DB* connection type (previously used) does not support the execution of a SQL statement created from a string. It only supports the execution of direct SQL commands/instructions.

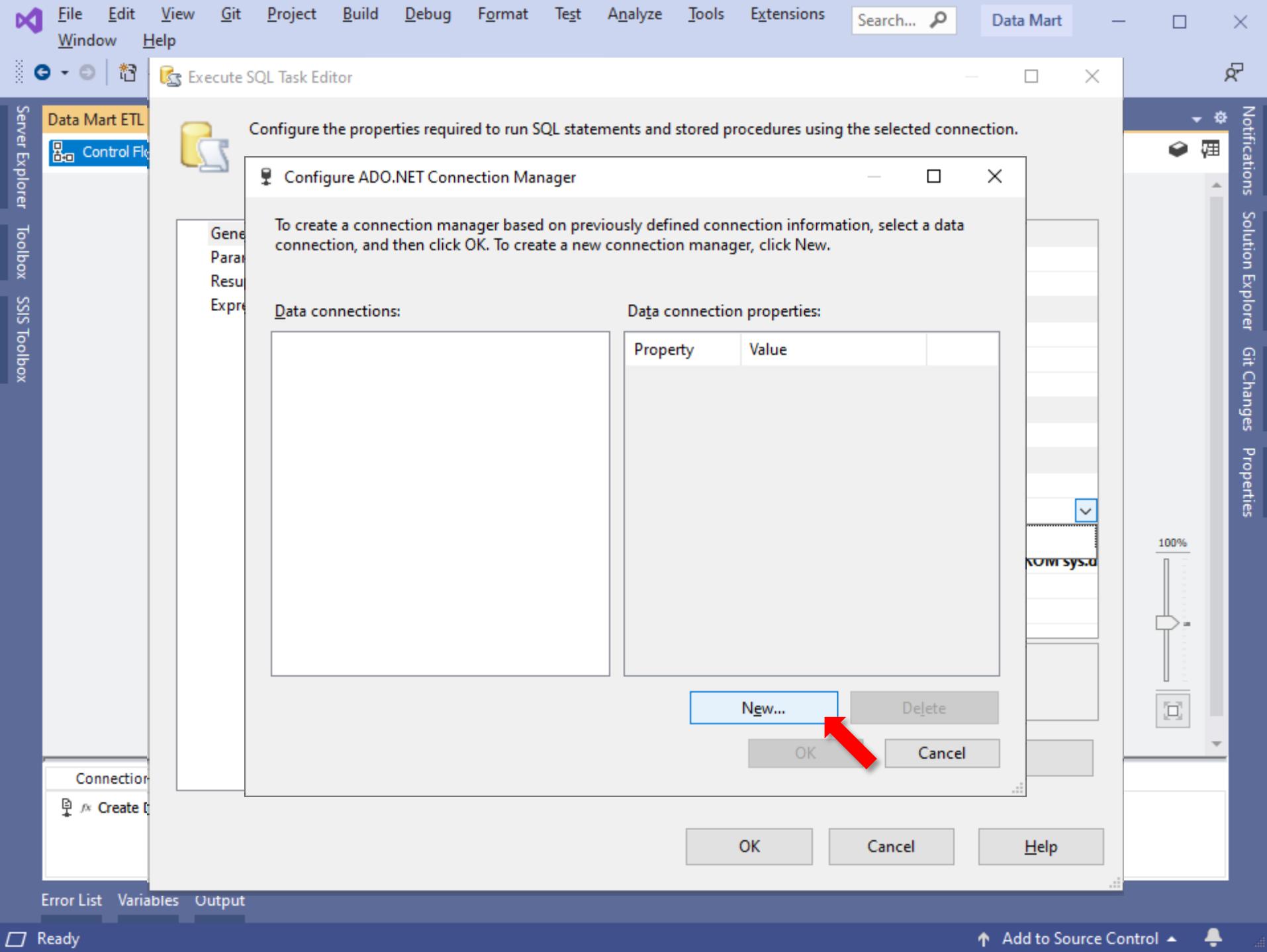
100%

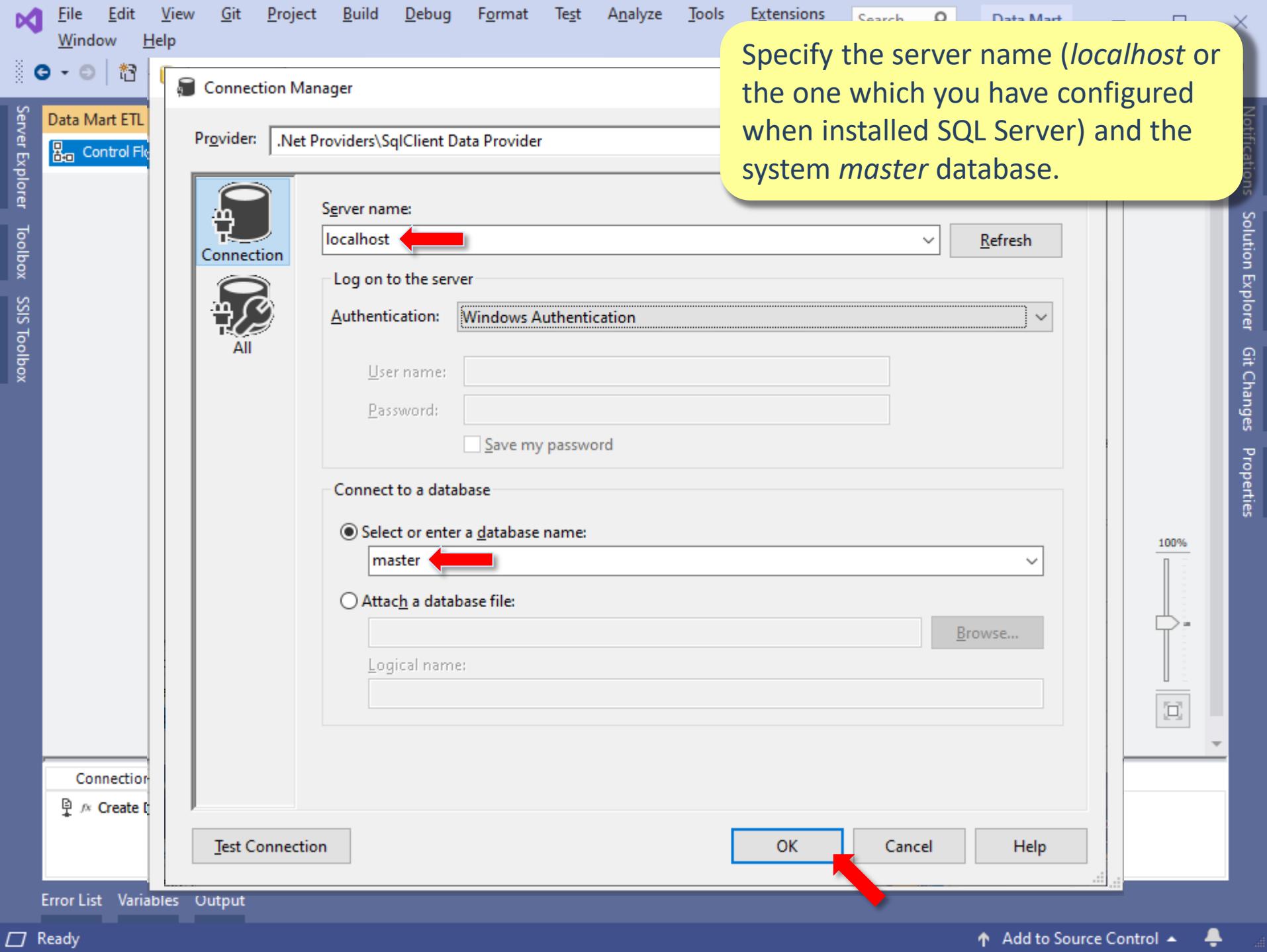
Error List Variables Output

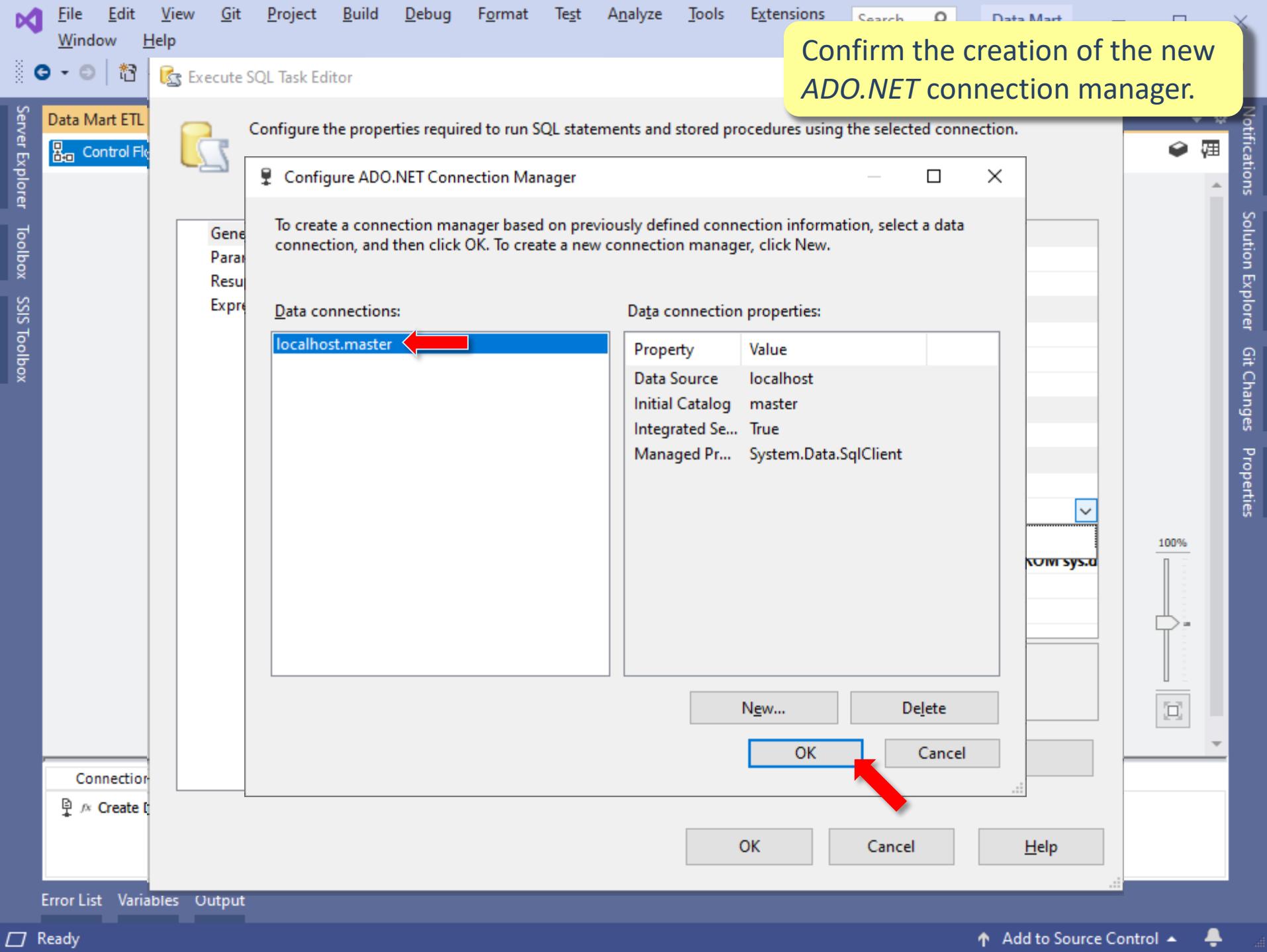
Add to Source Control Ready











Confirm the creation of the new  
ADO.NET connection manager.

Configure the properties required to run SQL statements and stored procedures using the selected connection.

### Configure ADO.NET Connection Manager

To create a connection manager based on previously defined connection information, select a data connection, and then click OK. To create a new connection manager, click New.

#### Data connections:

localhost.master

#### Data connection properties:

Property	Value
Data Source	localhost
Initial Catalog	master
Integrated Se...	True
Managed Pr...	System.Data.SqlClient

New...

Delete

OK

Cancel

OK

Cancel

Help

File

Edit

View

Git

Project

Build

Debug

Format

Test

Analyze

Tools

Extensions

Search

Data Mart

Window Help

Data Mart ETL

Control Flow



General  
Parameters  
Resources  
Expressions

Error List Variables Output



Notifications Solution Explorer Git Changes Properties

Add to Source Control



Ready

File Edit View Git Project Build Debug Format Test Analyze Tools Extensions Search... Data Mart

Data Mart ETL Control Flow

Server Explorer Toolbox SSIS Toolbox

Notifications Solution Explorer Git Changes Properties

Execute SQL Task Editor

Configure the properties required to run SQL statements and stored procedures using the selected connection.

General Parameter Mapping Result Set Expressions

General Options Result Set SQL Statement

Name	Create Staging Area Database
Description	Execute SQL Task
TimeOut	0
CodePage	1252
TypeConversionMode	Allowed
ResultSet	None
ConnectionType	ADO.NET
Connection	localhost.master
SQLSourceType	Direct input
SQLStatement	IF NOT EXISTS (SELECT name FROM sys.d
IsQueryStoredProcedure	False
BypassPrepare	False

Connection  
Specifies the connection used to access the data.

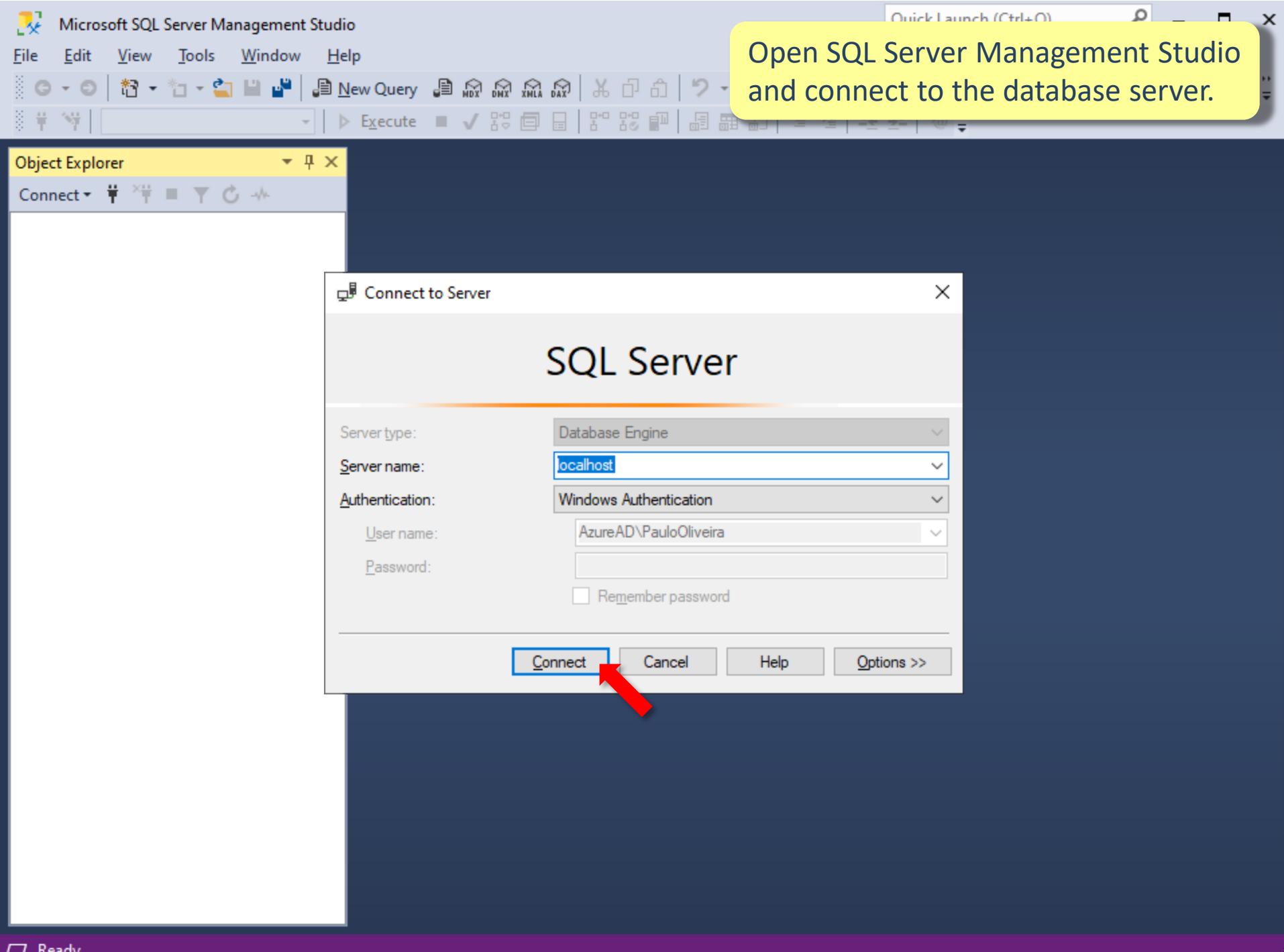
Browse... Build Query... Parse Query

OK Cancel Help

Error List Variables Output

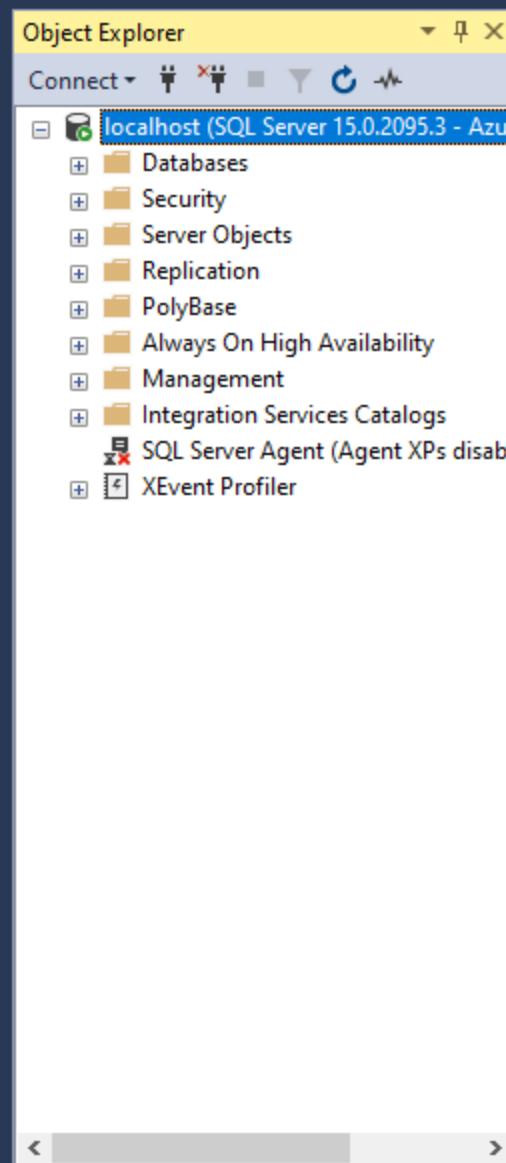
Add to Source Control Ready

The screenshot shows the 'Execute SQL Task Editor' dialog box within the SSIS interface. On the left, there's a navigation pane with 'Data Mart ETL' and 'Control Flow' selected. The main area has a title bar 'Execute SQL Task Editor' and a subtitle 'Configure the properties required to run SQL statements and stored procedures using the selected connection.' Below this is a tree view with 'General', 'Parameter Mapping', 'Result Set', and 'Expressions' nodes. The 'General' node is expanded, showing properties like Name (Create Staging Area Database), Description (Execute SQL Task), TimeOut (0), CodePage (1252), TypeConversionMode (Allowed), ResultSet (None), and ConnectionType (ADO.NET). The 'Connection' property is highlighted with a blue selection bar and a red arrow pointing to it. The value 'localhost.master' is shown in the dropdown. Below this, the 'Connection' section is expanded, explaining it specifies the connection used to access the data. At the bottom are buttons for 'Browse...', 'Build Query...', 'Parse Query...', 'OK', 'Cancel', and 'Help'.





Create a New Query window to write a new SQL instruction.



Write the following instruction that uses a parameter (@DBName), i.e., the database name, that is going to be concatenated with the remaining SQL text and then executed (by the EXEC SQL clause).

The screenshot shows the SSMS interface with the following details:

- Object Explorer:** Shows the database structure under "localhost (SQL Server 15.0.2095.3 - Azure)".
- Query Window:** Contains the following T-SQL code:

```
EXEC('IF NOT EXISTS(SELECT name FROM sys.databases WHERE name = ''' + @DBName + ''')
CREATE DATABASE ' + @DBName)
```

The code is highlighted with a red box.
- Messages Window:** Displays the output:

```
Commands completed successfully.

Completion time: 2022-10-18T15:14:20.9153452+01:00
```
- Status Bar:** Shows "Query executed successfully." and the execution details: "localhost (15.0 RTM) | AzureAD\PauloOliveira ... | master | 00:00:00 | 0 rows".

SQLQuery1.sql - localhost.master (AzureAD\PauloOliveira)

File Edit View Query Project Tools Window

New Query

master Execute

Object Explorer

localhost (SQL Server 15.0.2095.3 - AzureAD) Databases System Databases Database Snapshots AdventureWorksDW2019 ARPAD\_DataMart ARPAD\_StagingArea Security Server Objects Replication PolyBase Always On High Availability Management Integration Services Catalogs SQL Server Agent (Agent XPs disabled) XEvent Profiler

SQLQuery1.sql - loc...\\PauloOliveira (56)\*

```
EXEC('IF NOT EXISTS(SELECT name FROM sys.databases WHERE name = ''' + @DBName + ''')  
CREATE DATABASE ' + @DBName)
```

Messages

Commands completed successfully.

Completion time: 2022-10-18T15:14:20.9153

100 %

Query executed successfully.

localhost (15.0 RTM) | AzureAD\PauloOliveira ... | master | 00:00:00 | 0 rows

Col 1 Ch 1 INS Ln 1

As the `@DBName` parameter is not previously declared, the editor highlights the error. That's not important, so just select and copy the whole SQL instruction/command.

Cut Ctrl+X

**Copy Ctrl+C**

Paste Ctrl+V

Insert Snippet... Ctrl+K, C

Surround With... Ctrl+K, S

Connection

Open Server in Object Explorer Alt+F8

Execute F5

Display Estimated Execution Plan Ctrl+L

IntelliSense Enabled Ctrl+B, I

Trace Query in SQL Server Profiler Ctrl+Alt+T

Analyze Query in Database Engine Tuning Advisor

Design Query in Editor... Ctrl+Shift+D

Include Actual Execution Plan Ctrl+M

Include Live Query Statistics

Include Client Statistics Shift+A

Results To

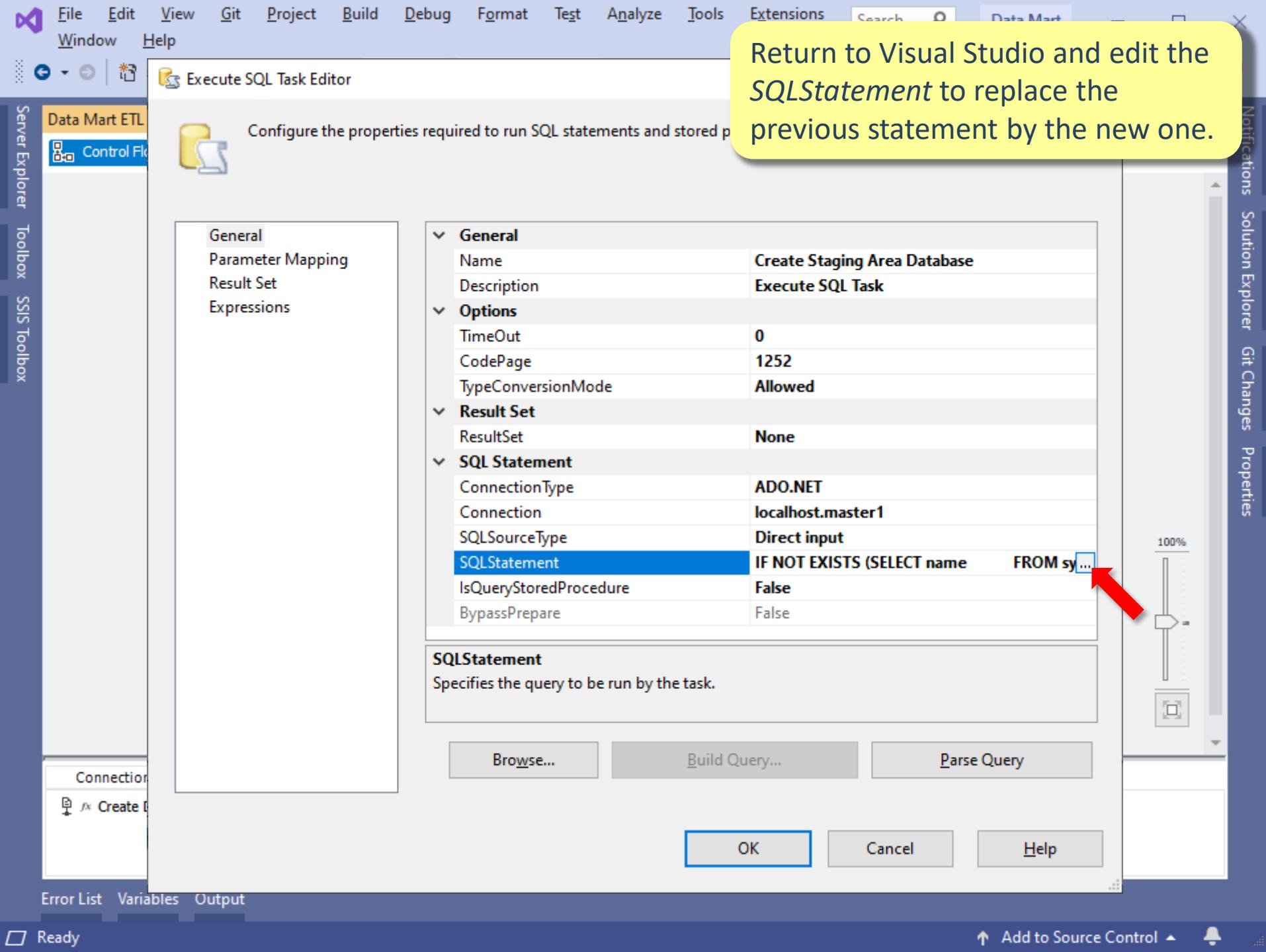
Properties Window F4

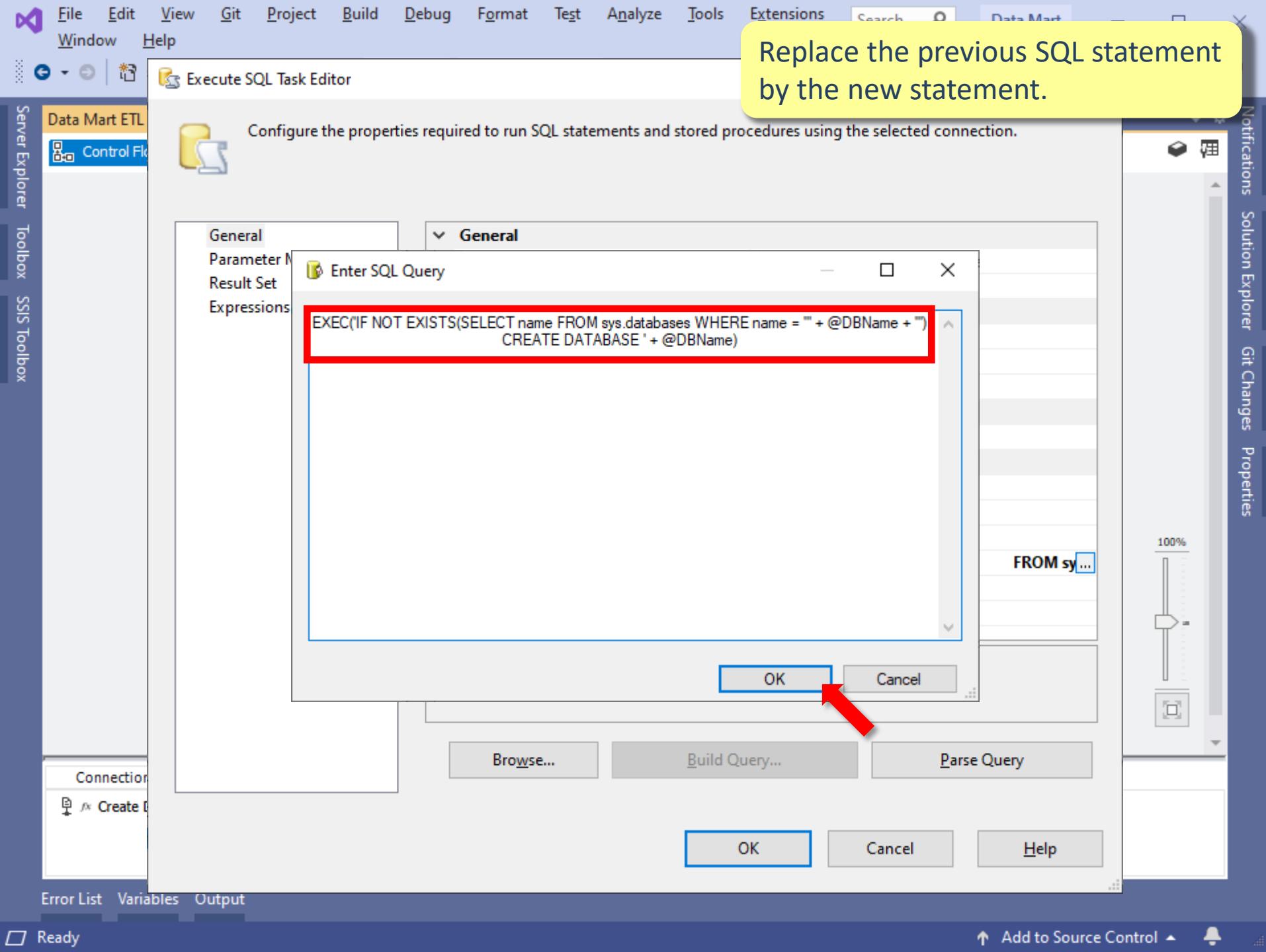
Query Options...

The screenshot shows the SSMS interface with a query in the center pane. The query is as follows:

```
EXEC('IF NOT EXISTS(SELECT name FROM sys.databases WHERE name = ''' + @DBName + ''')  
CREATE DATABASE ' + @DBName)
```

The context menu is open over the query text, and the 'Copy' option is highlighted with a red arrow. The status bar at the bottom indicates "Query executed successfully."





File Edit View Git Project Build Debug Format Test Analyze Tools Extensions Search... Data Mart

Data Mart ETL Control Flow

Server Explorer Toolbox SSIS Toolbox

Notifications Solution Explorer Git Changes Properties

Execute SQL Task Editor

Configure the properties required to run SQL statements and stored procedures using the selected connection.

General Parameter Mapping Result Set Expressions

General Options Result Set SQL Statement

Name	Create Staging Area Database
Description	Execute SQL Task
TimeOut	0
CodePage	1252
TypeConversionMode	Allowed
ResultSet	None
ConnectionType	ADO.NET
Connection	localhost.master1
SQLSourceType	Direct input
SQLStatement	EXEC('IF NOT EXISTS(SELECT name FROM sys.databases WHERE name = 'Staging') CREATE DATABASE [Staging]')
IsQueryStoredProcedure	False
BypassPrepare	False

SQLStatement

Specifies the query to be run by the task.

Browse... Build Query... Parse Query

OK Cancel Help

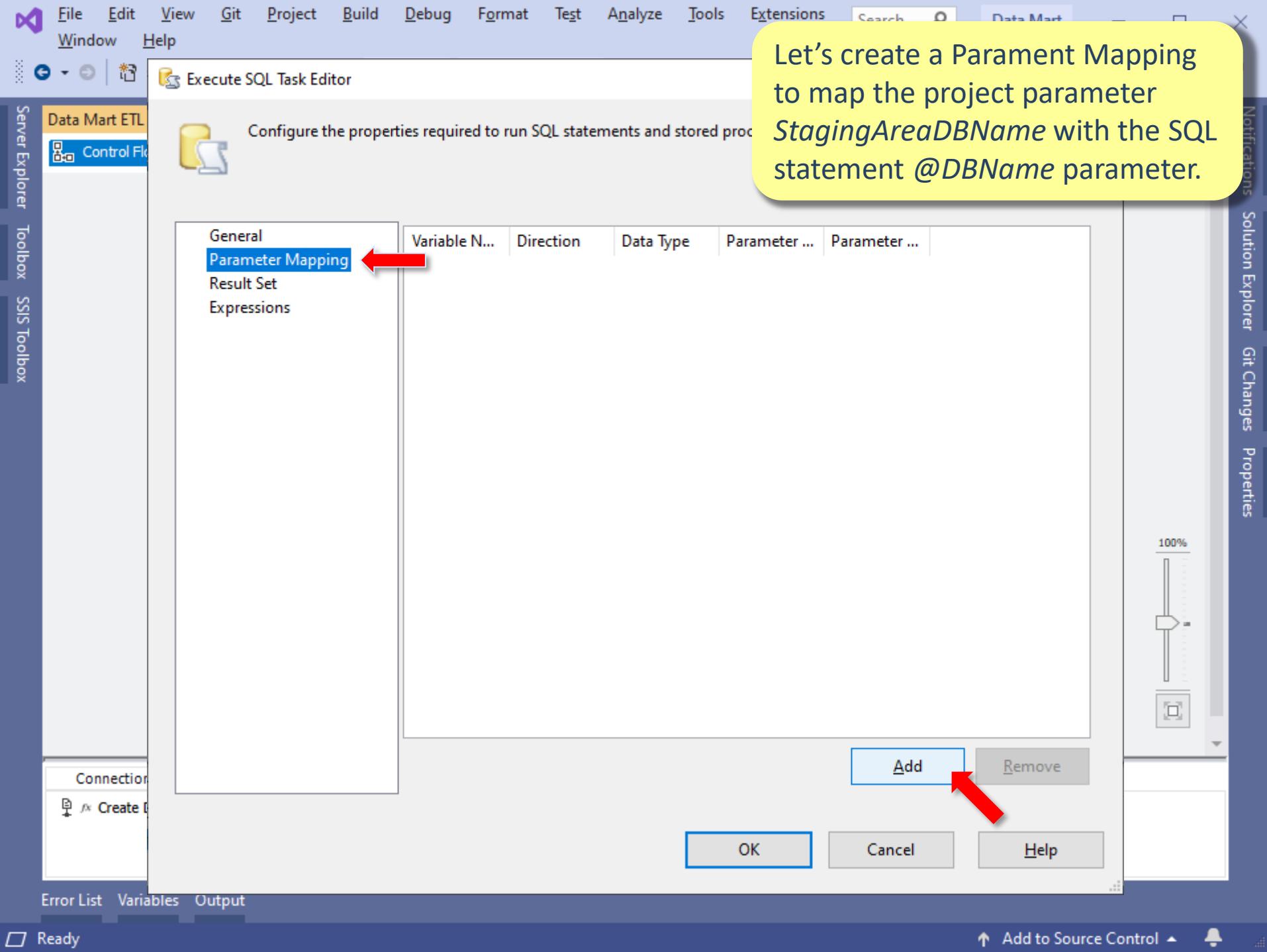
100%

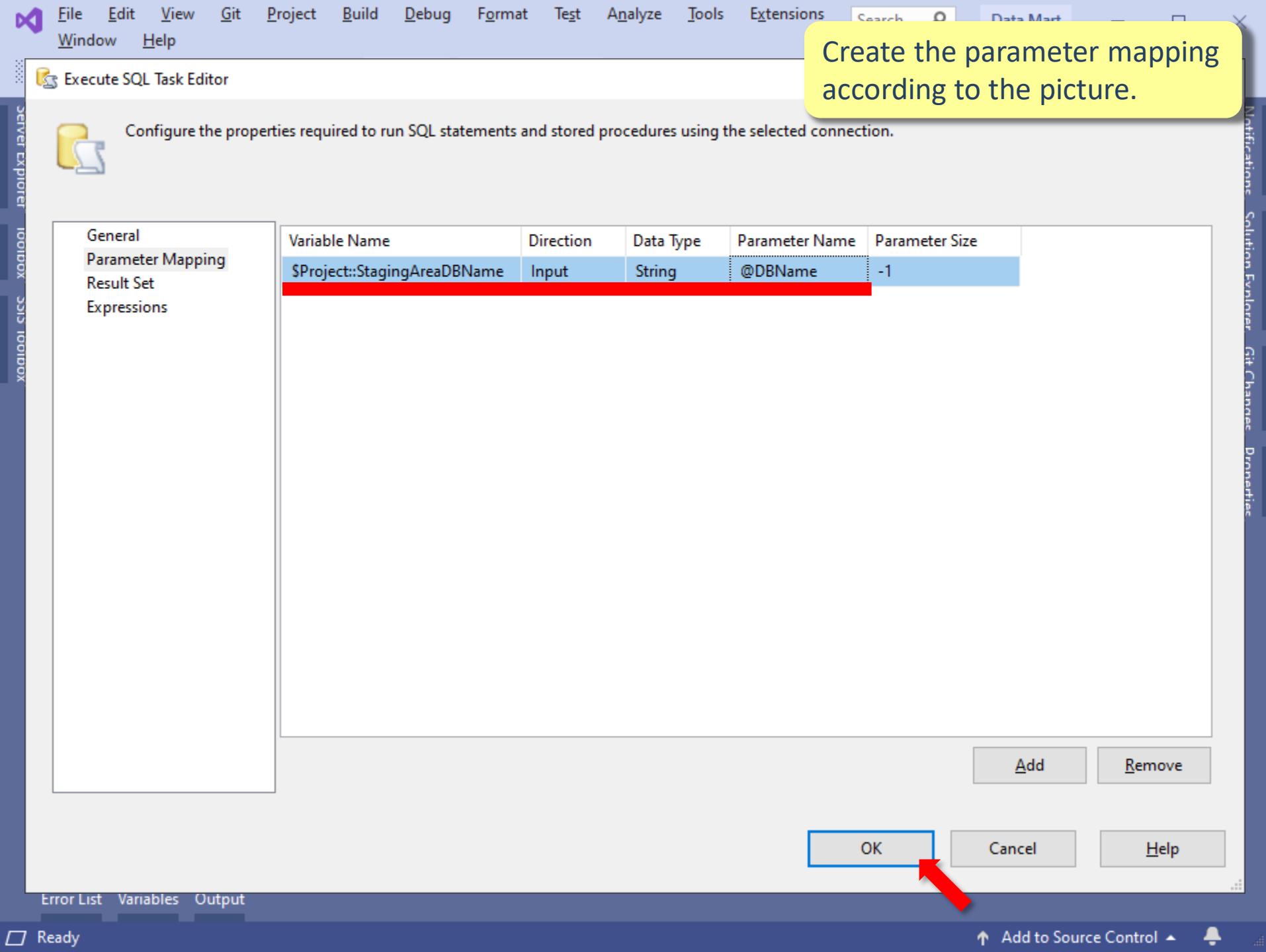
Connection Create D...

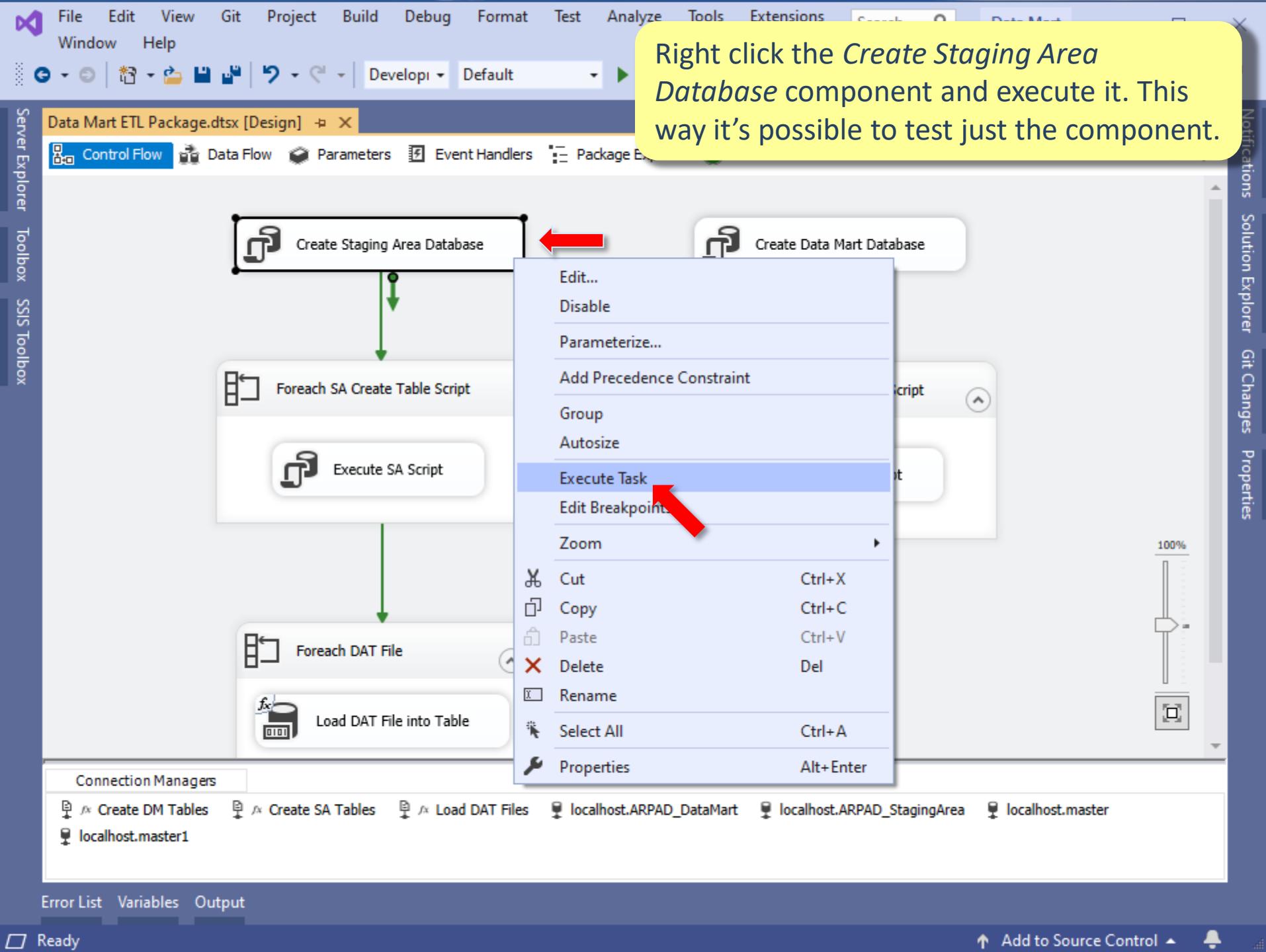
Error List Variables Output

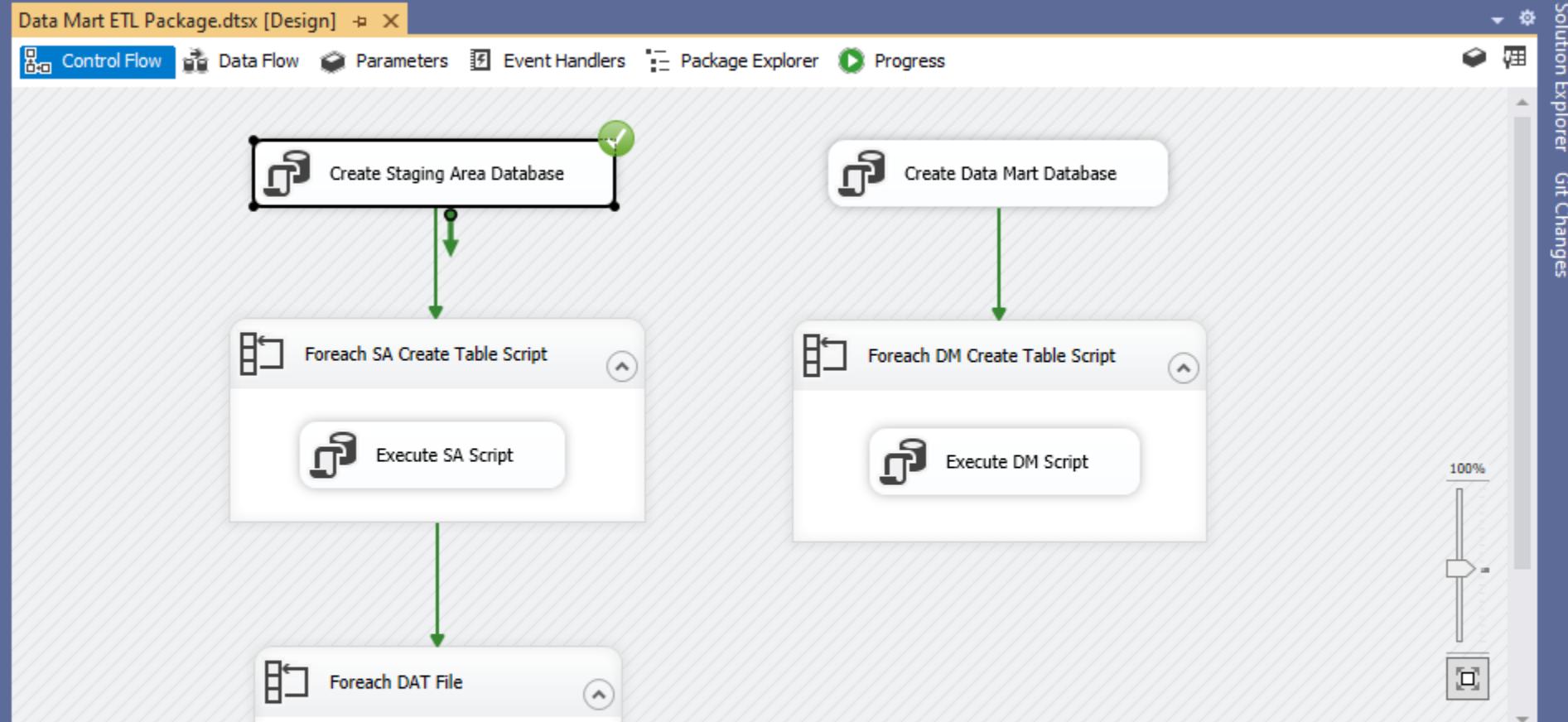
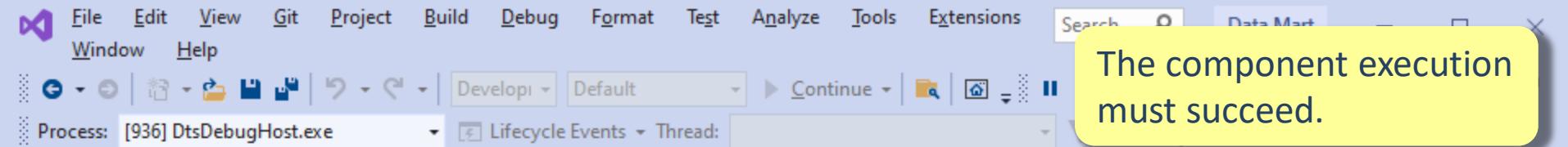
Add to Source Control Ready

The screenshot shows the 'Execute SQL Task Editor' dialog box in Microsoft SQL Server Integration Services (SSIS). The 'General' tab is selected. The 'Name' is set to 'Create Staging Area Database' and the 'Description' is 'Execute SQL Task'. In the 'SQL Statement' section, the 'SQLStatement' property is highlighted with a red arrow pointing to the value 'EXEC('IF NOT EXISTS(SELECT name FROM sys.databases WHERE name = 'Staging') CREATE DATABASE [Staging]')'. Below this, the 'SQLStatement' description states: 'Specifies the query to be run by the task.' At the bottom of the dialog are buttons for 'Browse...', 'Build Query...', and 'Parse Query'. At the very bottom are 'OK', 'Cancel', and 'Help' buttons. The status bar at the bottom shows '100%' and icons for 'Add to Source Control' and 'Ready'.









The Connection Managers pane of the Visual Studio interface. It lists the following connection managers: 'Create DM Tables', 'Create SA Tables', 'Load DAT Files', 'localhost.ARPAD\_DataMart', 'localhost.ARPAD\_StagingArea', 'localhost.master', and 'localhost.master1'. The 'Create DM Tables' and 'Create SA Tables' entries are highlighted with a blue background.

Package execution completed. Click here to switch to design mode, or select Stop Debugging from the Debug menu.

Call Stack Breakpoints Exception Settings Command Window Immediate Window Output Autos Locals Watch 1

Ready

Add to Source Control



File Edit View Git Project Build Debug Format Test Analyze Tools Extensions Search Data Mart

Window Help

Process: [936] DtsDebugHost.exe

Data Mart ETL Package.dtsx [Design]

Control Flow Data Flow Parameters

Edit the *Create Data Mart Database* component and perform exactly the same steps you did for the *Create Staging Area Database* component. You can use the same SQL EXEC instruction, but the *@DBName* parameter now needs to be mapped with the *DataMartDBName* project parameter.

```
graph TD; SA[Create Staging Area Database] --> SA_Foreach[ForEach SA Create Table Script]; SA_Foreach --> SA_Execute[Execute SA Script]; SA_Execute --> DAT[ForEach DAT File]; DM[Create Data Mart Database] --> DM_Foreach[ForEach DM Create Table Script]; DM_Foreach --> DM_Execute[Execute DM Script]; DAT --> End[100%]
```

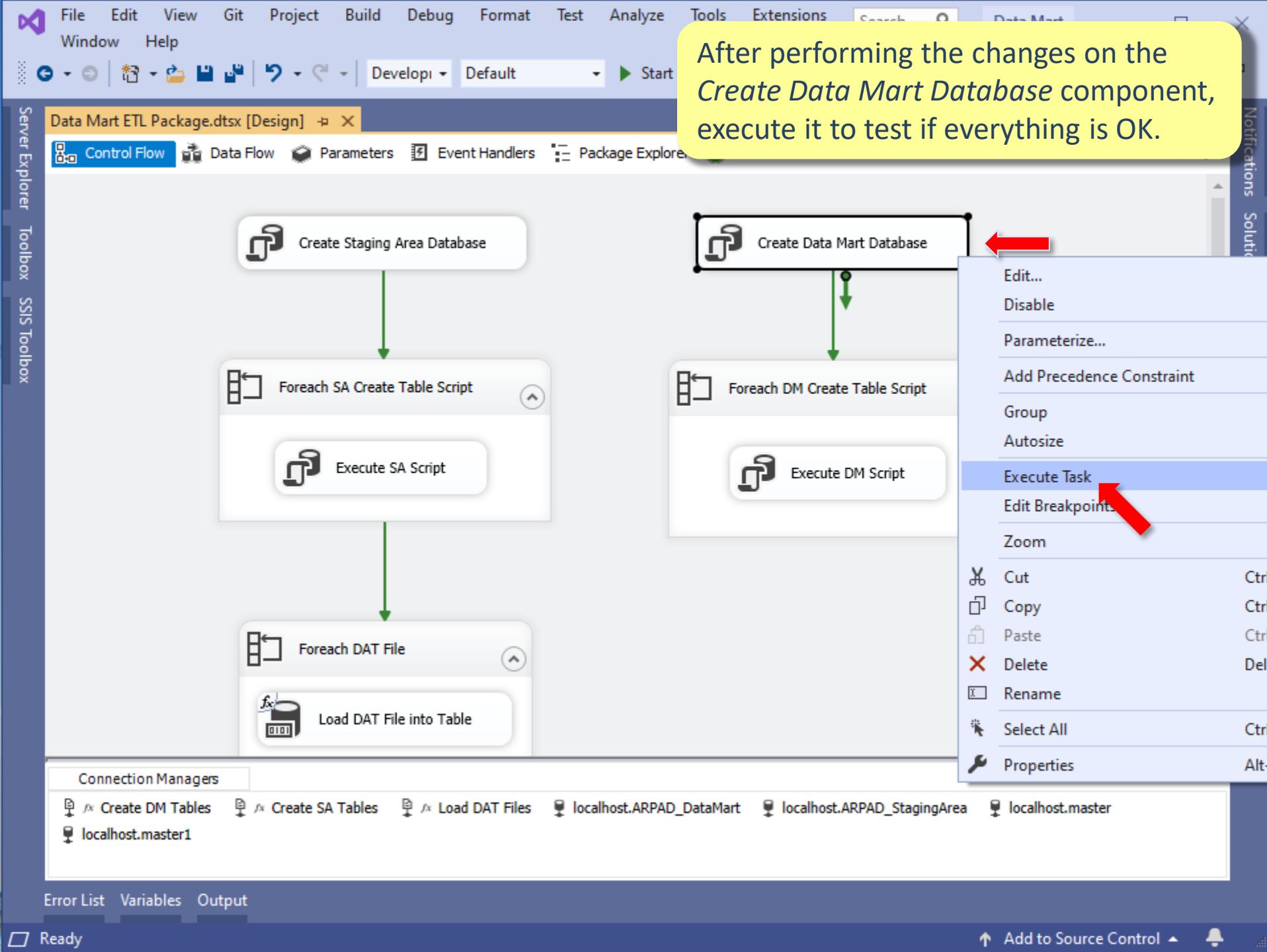
Connection Managers

- Create DM Tables
- Create SA Tables
- Load DAT Files
- localhost.ARPAD\_DataMart
- localhost.ARPAD\_StagingArea
- localhost.master
- localhost.master1

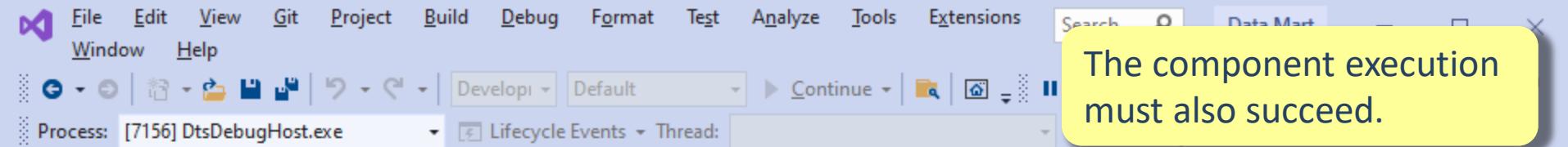
Package execution completed. Click here to switch to design mode, or select Stop Debugging from the Debug menu.

Call Stack Breakpoints Exception Settings Command Window Immediate Window Output Autos Locals Watch 1

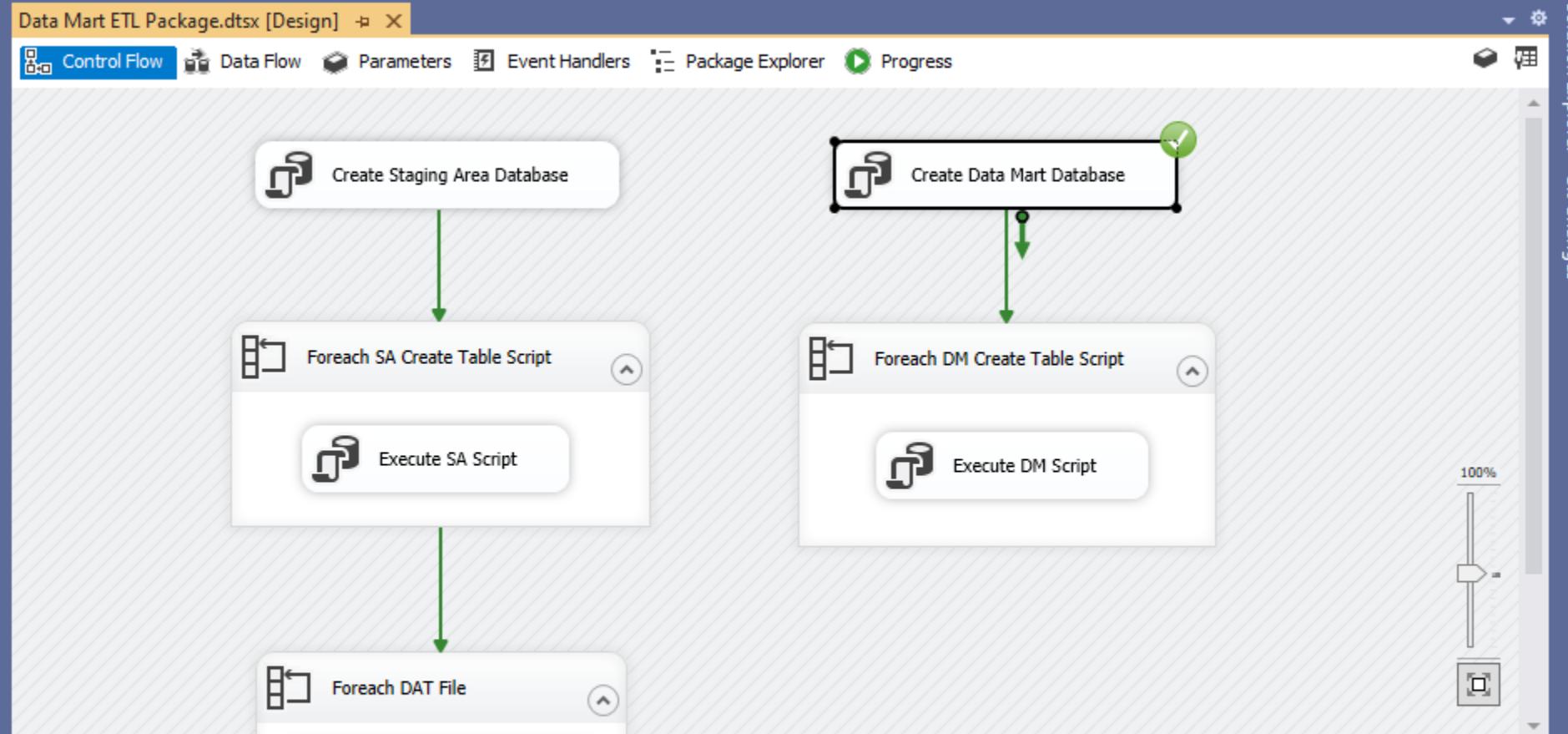
Ready Add to Source Control



After performing the changes on the *Create Data Mart Database* component, execute it to test if everything is OK.



The component execution  
must also succeed.



#### Connection Managers

Create DM Tables	Create SA Tables	Load DAT Files	localhost.ARPAD_DataMart	localhost.ARPAD_StagingArea	localhost.master
localhost.master1					

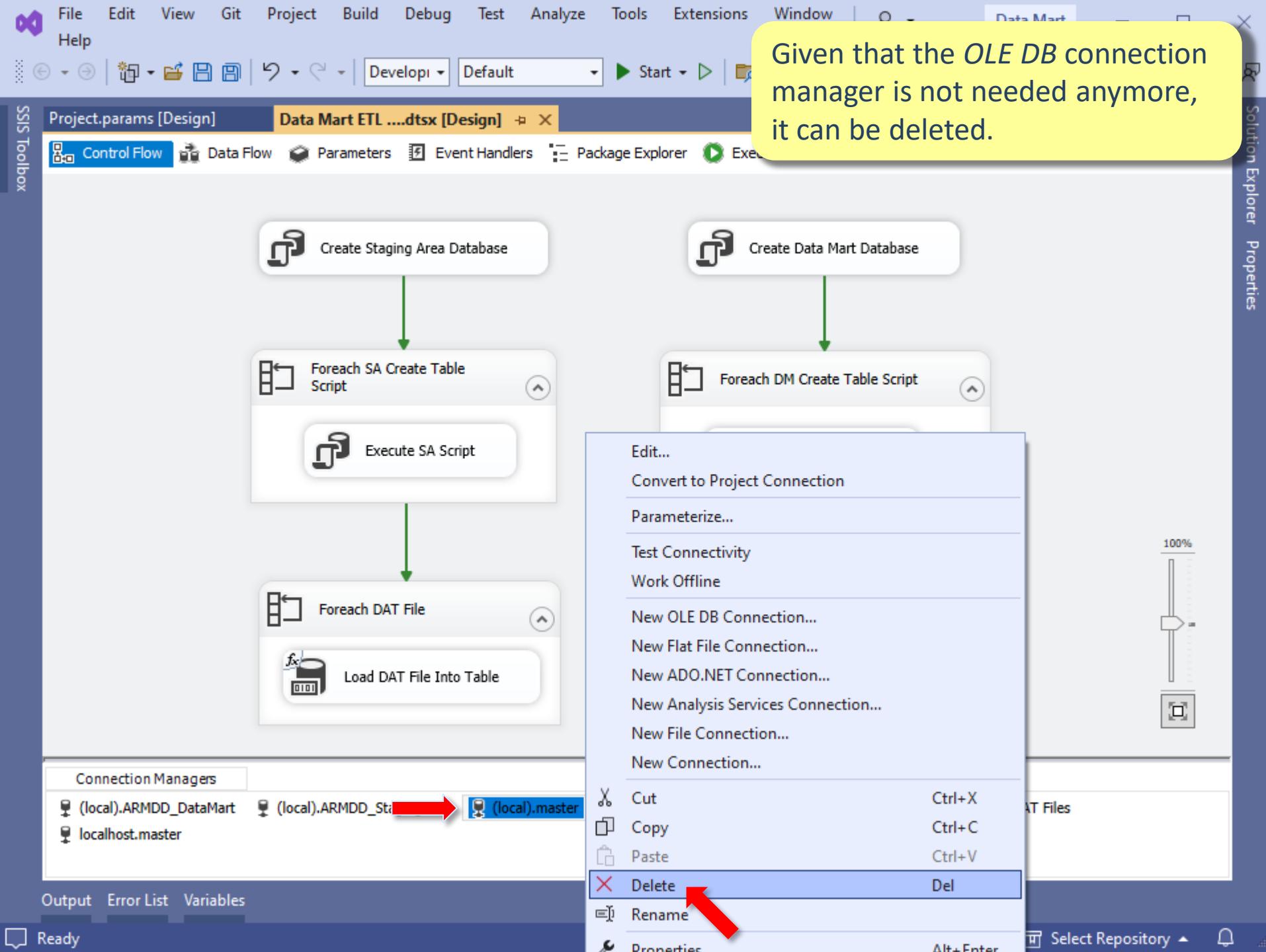
Package execution completed. Click here to switch to design mode, or select Stop Debugging from the Debug menu.

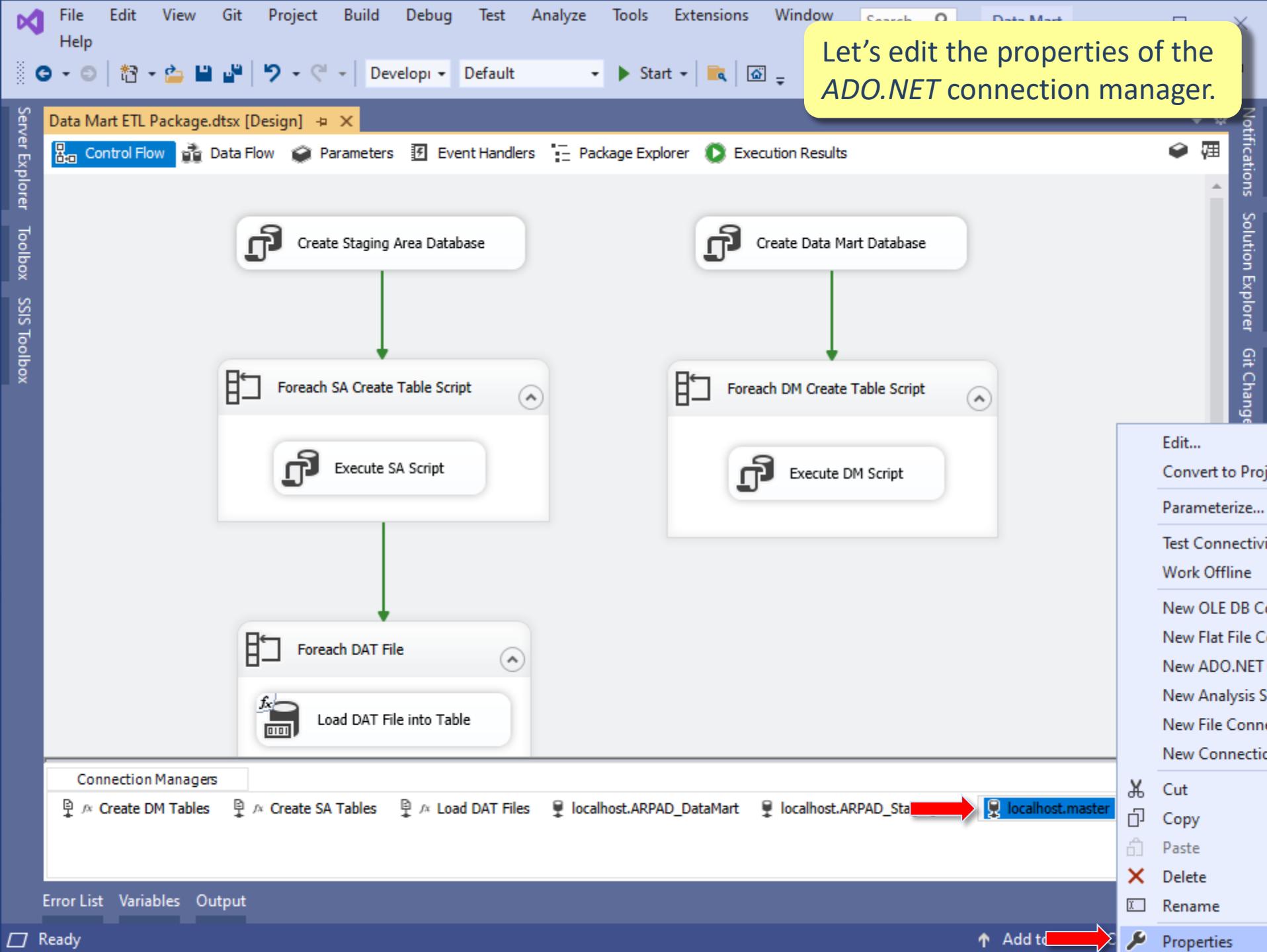
Call Stack Breakpoints Exception Settings Command Window Immediate Window Output Autos Locals Watch 1

Ready

Add to Source Control







File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Search Data Mart

Develop Default Start

Data Mart ETL Package.dtsx [Design]

Control Flow Data Flow Parameters Event Handlers Package Explorer Execution Result

Create Staging Area Database

Identification

PackagePath \Package.Connections[localhost.master]

Misc

ConnectByProxy False

ConnectionString ADO.NET:System.Data.SqlClient

ConnectionManagerType Data Source=localhost;Initial Catalog=master

ConnectUsingManagement False

DataSourceID

DelayValidation False

Description

Expressions

HasExpressions False

ID {725B01AC-1B9F-43A7-8339-\*\*\*\*\*}

InitialCatalog master

Name localhost.master

Password \*\*\*\*\*

Qualifier System.Data.SqlClient.SqlCommand

RetainSameConnection False

ServerName localhost

SupportsDTCTransactions True

UserName

Show Editor

Expressions

A collection of expressions. The evaluation result of each expression is assigned to a property and replac...

Edit the Expressions of the connection manager.

Properties

localhost.master Connection

Identification

PackagePath \Package.Connections[localhost.master]

Misc

ConnectByProxy False

ConnectionString ADO.NET:System.Data.SqlClient

ConnectionManagerType Data Source=localhost;Initial Catalog=master

ConnectUsingManagement False

DataSourceID

DelayValidation False

Description

Expressions

HasExpressions False

ID {725B01AC-1B9F-43A7-8339-\*\*\*\*\*}

InitialCatalog master

Name localhost.master

Password \*\*\*\*\*

Qualifier System.Data.SqlClient.SqlCommand

RetainSameConnection False

ServerName localhost

SupportsDTCTransactions True

UserName

...

Server Explorer Toolbox SSIS Toolbox

Control Flow

ForEach SA Create Table Script

Execute SA Script

ForEach DM Create

Execute

ForEach DAT File

Load DAT File into Table

Connection Managers

Create DM Tables Create SA Tables Load DAT Files localhost.ARPAD\_DataMart localhost.master

Error List Variables Output

Add to Source Control

Ready

Notifications Solution Explorer Git Changes Properties

File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Search Data Mart

Help

Develop Default Start

Data Mart ETL Package.dtsx [Design]

Control Flow Data Flow Parameters Event Handlers Package Explorer Execution Result

localhost.master Connection

Properties

Identification

PackagePath

Package.Connections[localhost.master]

False

ADO.NET:System.Data.SqlClient

Data Source=localhost;Initial Catalog=master;Integrated Security=True

False

False

False

{725B01AC-1B9F-43A7-8339-\*\*\*\*\*}

master

localhost.master

\*\*\*\*\*

System.Data.SqlClient.SqlConnection

False

localhost

True

Create Staging Area Database

Foreach SA Create Table Script

Execute SA Script

Foreach DAT File

Load DAT File into Table

Property Expressions Editor

Property expressions:

Property	Expression
ServerName	<input style="width: 20px; height: 20px;" type="button" value="..."/>

Delete OK Cancel

Connection Managers

Create DM Tables Create SA Tables Load DAT Files localhost.ARPAD\_DataMart localhost.

Show Editor

Expressions

A collection of expressions. The evaluation result of each expression is assigned to a property and replac...

Error List Variables Output

Add to Source Control

Ready

Let's map the *ServerName* property with the *ServerName* project parameter.

The screenshot shows the SSIS Designer interface with a Control Flow diagram. A 'Create Staging Area Database' task is selected. Its properties are displayed in a dialog box. In the 'Properties' section, the 'ServerName' property is being edited. The 'Expression' column contains a placeholder for a mapped expression. A red arrow points to the ellipsis button ('...') in this column, which is used to map the property to a project parameter like 'ServerName'.

File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Search Data Mart

Help

Develop Default Start

Data Mart ETL Package.dtsx [Design]

Properties

Control Flow Data Flow

Expression Builder

Specify the expression for the property: ServerName.

Variables and Parameters

- System Variables
  - \$Project::DataMartDBName
  - \$Project::PackagePath
  - \$Project::ServerName**
  - \$Project::StagingAreaDBName
  - User::DatFilename
  - User::DMScriptFilename
  - User::SAScriptFilename

Mathematical Functions

String Functions

Date/Time Functions

NULL Functions

Type Casts

Operators

Description:

Expression:

Evaluated value:

Evaluate Expression OK Cancel

Connection Managers

Create DM Tables Create DM Tables

Error List Variables Output

Add to Source Control

Drag the Server name parameter to the *Expression* field.

Properties

Action

\Package.Connections[localhost]

False

ADO.NET:System.Data.SqlClient

**Data Source=localhost;Initi**

False

False

False

False

{725B01AC-1B9F-43A7-8339-

master

**localhost.master**

\*\*\*\*\*

**System.Data.SqlClient.SqlCo**

False

localhost

True

Expressions

A collection of expressions. The evaluation result of each expression is assigned to a property and replac...

File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Search... Data Mart

Develop Default Start

Data Mart ETL Package.dtsx [Design]

Properties

Control Flow Data Flow

Expression Builder

Specify the expression for the property: ServerName.

Variables and Parameters

- System Variables
  - \$Project::DataMartDBName
  - \$Project::PackagePath
  - \$Project::ServerName
  - \$Project::StagingAreaDBName
  - User::DatFilename
  - User::DMScriptFilename
  - User::SAScriptFilename

Mathematical Functions

String Functions

Date/Time Functions

NULL Functions

Type Casts

Operators

Description:

Expression: @[\$Project::ServerName] 

Evaluated value: 

Evaluate Expression 

OK Cancel

Connection Managers

Create DM Tables Create

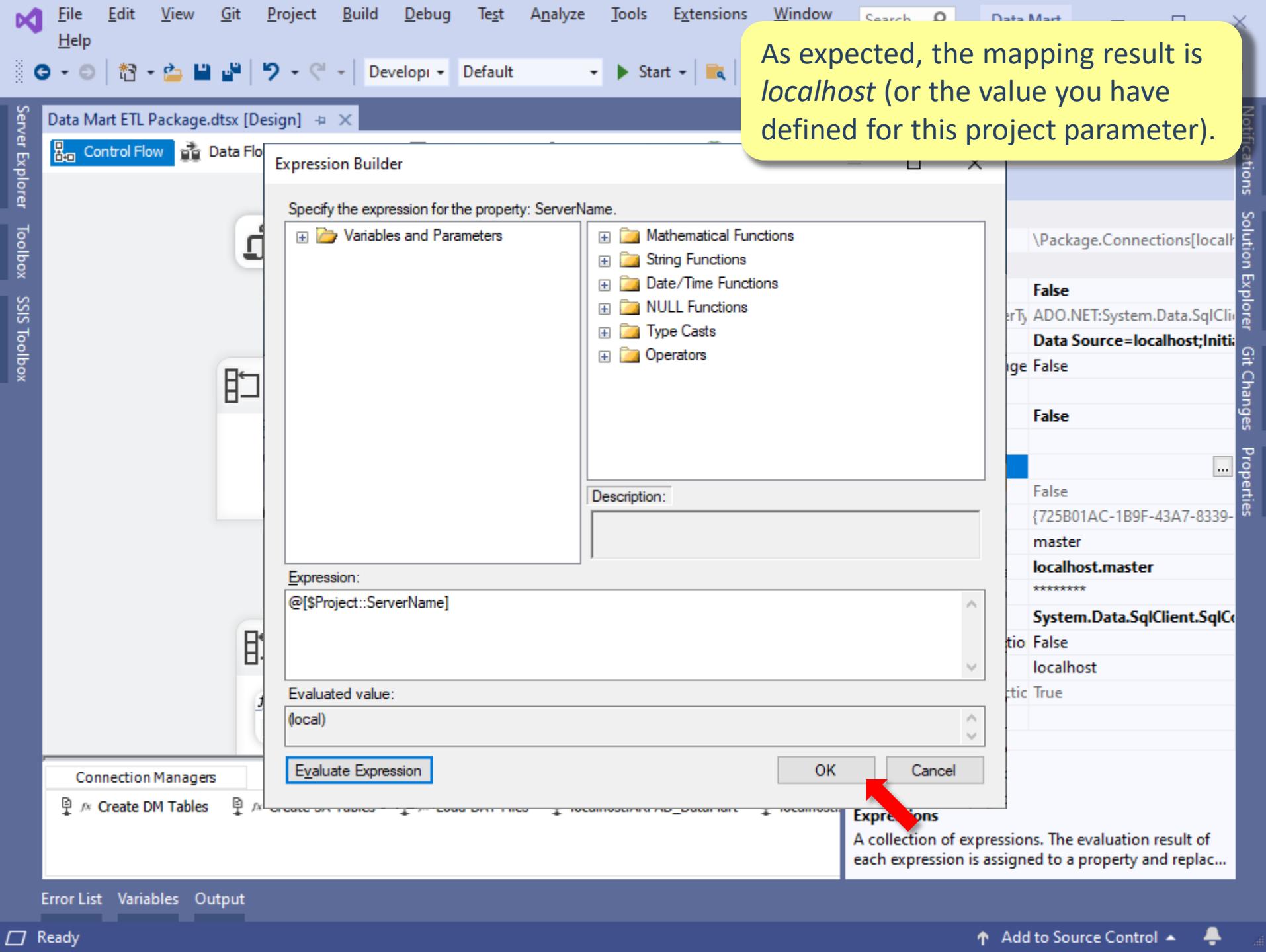
Error List Variables Output

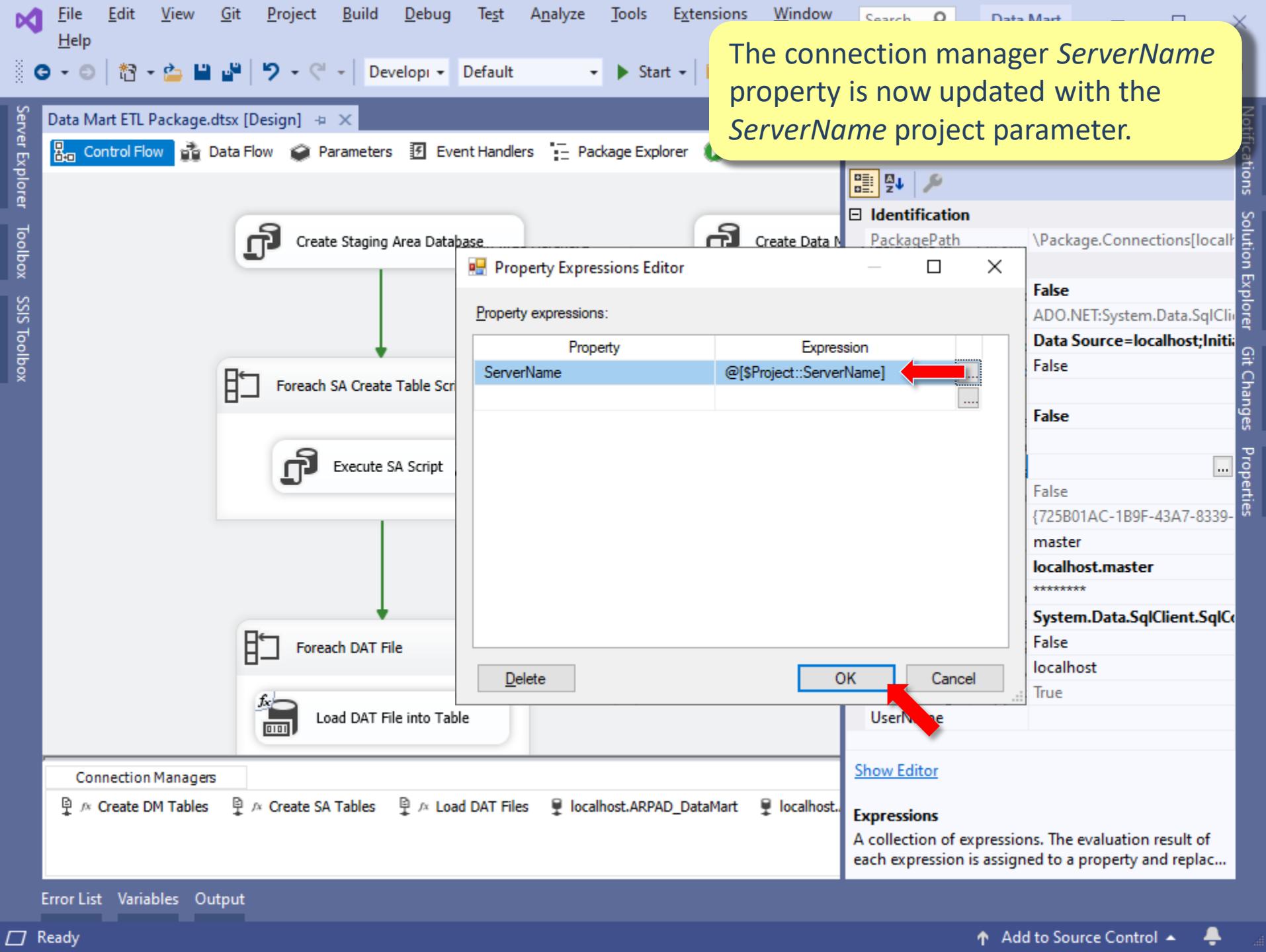
Add to Source Control

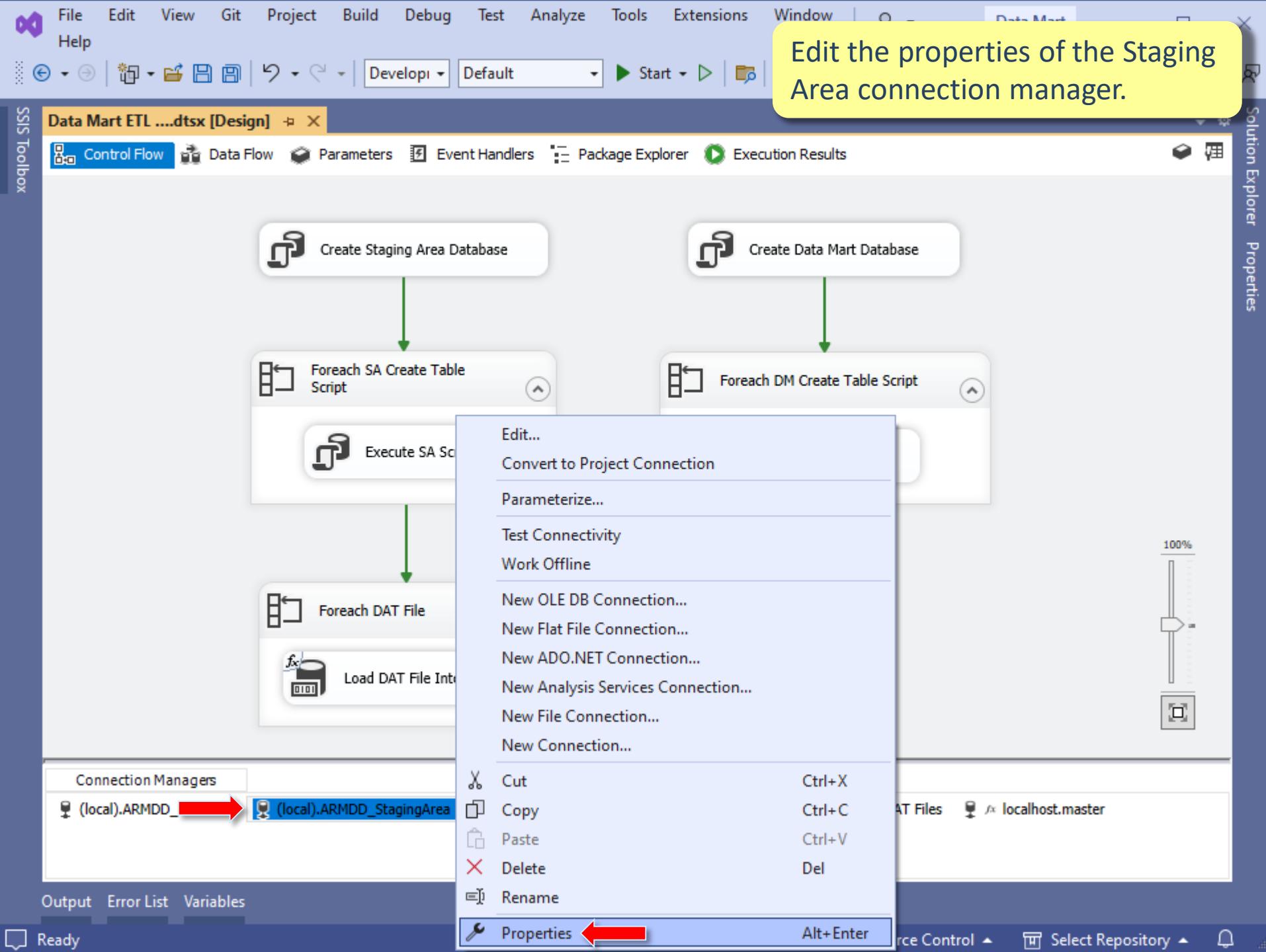
Just to test, evaluate the expression result.

tion  
Package.Connections[localhost]  
False  
ADO.NET:System.Data.SqlClient  
Data Source=localhost;Initia  
False  
False  
False  
False  
False  
master  
localhost.master  
\*\*\*\*\*  
System.Data.SqlClient.SqlCo  
False  
localhost  
True

Expressions  
A collection of expressions. The evaluation result of each expression is assigned to a property and replac...







File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Search Data Mart

Develop Default Start

Data Mart ETL Package.dtsx [Design]\*

Control Flow Data Flow Parameters Event Handlers Package Explorer Execution Results

Server Explorer Toolbox SSIS Toolbox

Edit the Expressions of the connection manager.

Create Staging Area Database

Foreach SA Create Table Script

Execute SA Script

Foreach DM Create

Execute

Foreach DAT File

Load DAT File into Table

Connection Managers

Create DM Tables Create SA Tables Load DAT Files localhost.ARPAD\_DataMart localhost.ARPAD\_StagingArea

Error List Variables Output

Add to Source Control

Properties

localhost.ARPAD\_StagingArea Connection

Misc

- ConnectByProxy False
- ConnectionString OLEDB
- ConnectionStringType Data Source=localhost;In
- ConnectRetryCount 1
- ConnectRetryInterval 5
- ConnectUsingManagement False
- DataSourceID
- DelayValidation False
- Description

Expressions

- HasExpressions False
- ID {1920F007-C34C-4315-B74
- InitialCatalog ARPAD\_StagingArea
- Name localhost.ARPAD\_StagingArea
- Password \*\*\*\*\*
- Qualifier
- RetainSameConnection False
- ServerName localhost
- SupportsDTCTransactions True
- UserName

Show Editor

Expressions

A collection of expressions. The evaluation result of each expression is assigned to a property and replac...

The screenshot shows the Microsoft SQL Server Integration Services (SSIS) Designer interface. The main area displays a Control Flow task sequence consisting of five tasks connected sequentially: 'Create Staging Area Database', 'Foreach SA Create Table Script' (containing 'Execute SA Script'), 'Foreach DM Create' (containing 'Execute'), and 'Foreach DAT File' (containing 'Load DAT File into Table'). The 'Foreach DAT File' task has a yellow warning icon. The 'Execute' task in the 'Foreach DM Create' loop also has a yellow warning icon. The 'Create Staging Area Database' task has a green arrow pointing to the 'Foreach SA Create Table Script' task. The 'Execute SA Script' task has a green arrow pointing to the 'Foreach DAT File' task. The 'Execute' task has a green arrow pointing to the 'Load DAT File into Table' task.

The Properties window on the right is open for the 'localhost.ARPAD\_StagingArea' connection manager. A yellow callout box with the text 'Edit the Expressions of the connection manager.' points to the 'Expressions' section in the Properties window. A red arrow points to the '...' button in the 'Expressions' section. The 'Expressions' section contains the following properties:

- HasExpressions: False
- ID: {1920F007-C34C-4315-B74}
- InitialCatalog: ARPAD\_StagingArea
- Name: localhost.ARPAD\_StagingArea
- Password: \*\*\*\*\*
- Qualifier
- RetainSameConnection: False
- ServerName: localhost
- SupportsDTCTransactions: True
- UserName

The 'Show Editor' link is visible below the 'Expressions' section. The 'Expressions' section is described as follows: 'A collection of expressions. The evaluation result of each expression is assigned to a property and replac...'.

File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Search Data Mart

Help

Develop Default Start

Data Mart ETL Package.dtsx [Design]\*

Control Flow Data Flow Parameters Event Handlers Package Explorer Execution Results

localhost.ARPAD\_StagingArea Connection

Properties

Misc

False

OLEDB

Data Source=localhost;In

1

5

False

False

False

False

{1920F007-C34C-4315-B74

ARPAD\_StagingArea

localhost.ARPAD\_Staging

\*\*\*\*\*

False

localhost

True

Show Editor

Expressions

A collection of expressions. The evaluation result of each expression is assigned to a property and replac...

Create Staging Area Database

Foreach SA Create Table Script

Execute SA Script

Foreach DAT File

Load DAT File into Table

Property Expressions Editor

Property expressions:

Property	Expression
ServerName	<input style="border: none; background-color: transparent; font-size: inherit; padding: 0;" type="button" value="..."/>

Delete OK Cancel

Connection Managers

Create DM Tables Create SA Tables Load DAT Files localhost.ARPAD\_DataMart localhost.ARPAD\_StagingArea

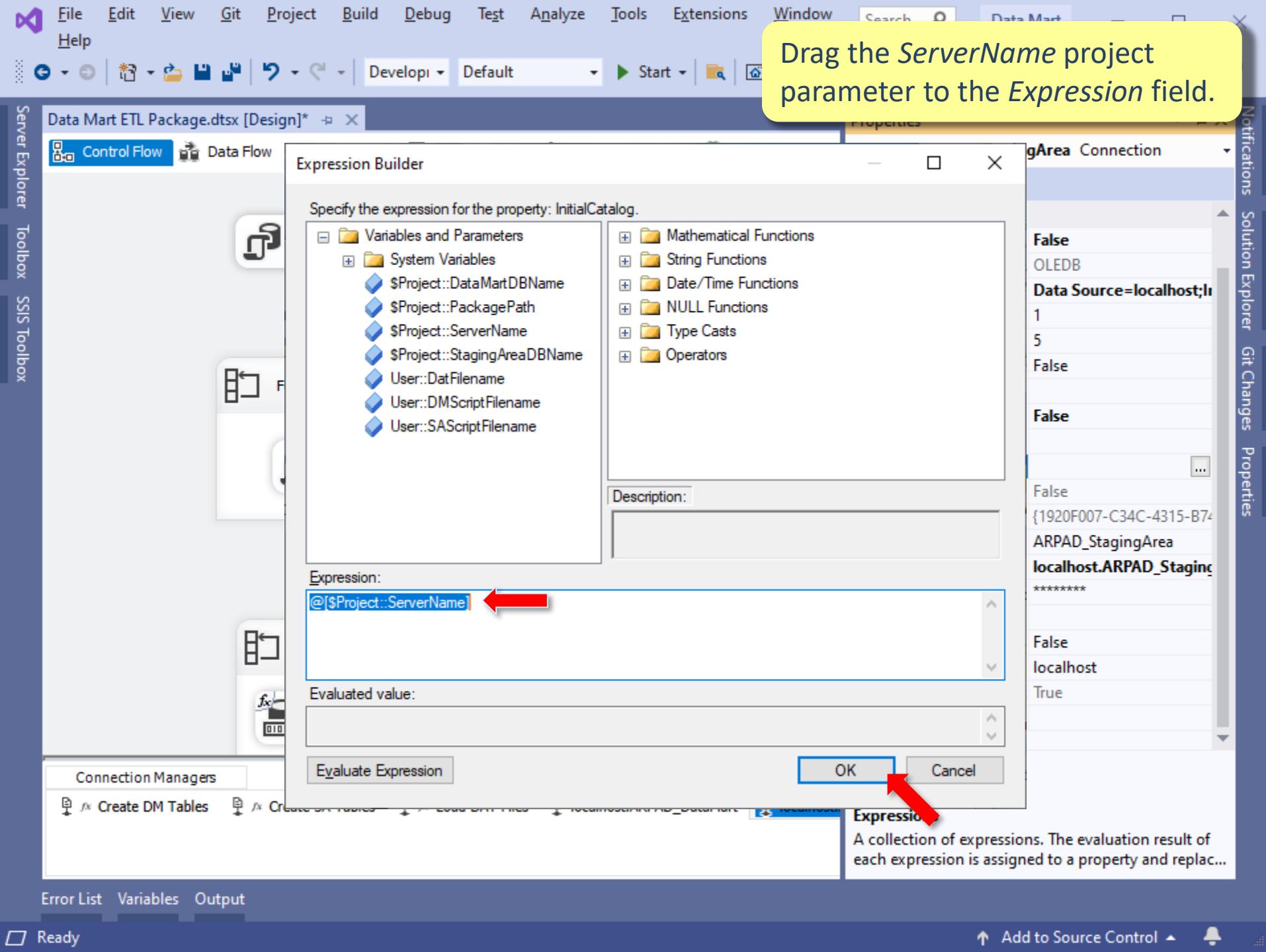
Error List Variables Output

Add to Source Control

Ready

Let's map the *ServerName* property with the *ServerName* project parameter.

The screenshot shows the SSIS Designer interface with a package named "Data Mart ETL Package.dtsx". The Control Flow tab is selected, displaying a sequence of tasks: "Create Staging Area Database", "Foreach SA Create Table Script", "Execute SA Script", "Foreach DAT File", and "Load DAT File into Table". A green arrow points from the first task to the second. Another green arrow points from the fourth task to the fifth. The "Foreach SA Create Table Script" task is currently selected, opening a "Property Expressions Editor" dialog. This dialog lists a single property expression for the "ServerName" property, which is highlighted with a red arrow. To the right of the editor is a properties grid showing connection manager settings for "localhost.ARPAD\_StagingArea". The "ServerName" property is mapped to the "ServerName" project parameter, which is also highlighted with a red arrow. A yellow callout box at the top right contains the text: "Let's map the *ServerName* property with the *ServerName* project parameter."



File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Search Data Mart

Help

Develop Default

Data Mart ETL Package.dtsx [Design]\*

Control Flow Data Flow Parameters Event Handlers

Create Staging Area Database

Foreach SA Create Table Script

Execute SA Script

Foreach DAT File

Load DAT File into Table

Property Expressions Editor

Property expressions:

Property	Expression
ServerName	@[Project::ServerName]
InitialCatalog	@[Project::InitialCatalog]

OK Cancel Delete

Connection Managers

Create DM Tables Create SA Tables Load DAT Files localhost.ARPAD\_DataMart localhost

Error List Variables Output

Add to Source Control

Let's map the *InitialCatalog* property (i.e., the specific database that we want to connect to) to the *StagingAreaDBName* project parameter.

Misc ConnectByProxy

OLEDB  
Data Source=localhost;In 1  
5 False  
False  
False  
False {1920F007-C34C-4315-B74 ARPAD\_StagingArea localhost.ARPAD\_Staging \*\*\*\*\* False localhost True

Show Editor

Expressions

A collection of expressions. The evaluation result of each expression is assigned to a property and replac...

File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Search Data Mart

Help

Develop Default Start

Data Mart ETL Package.dtsx [Design]\*

Control Flow Data Flow

Properties

gArea Connection

False OLEDB Data Source=localhost;InitialCatalog=15;Provider=SQLNCLI11.1;Integrated Security=SSPI;Auto Translate=False

False {1920F007-C34C-4315-B74A-ARPAD\_StagingArea localhost.ARPAD\_StagingArea\*\*\*\*\*

False localhost True

Drag the *StagingAreaDBName* parameter to the *Expression* field.

Expression Builder

Specify the expression for the property: InitialCatalog.

Variables and Parameters

- System Variables
  - \$Project::DataMartDBName
  - \$Project::PackagePath
  - \$Project::ServerName
  - \$Project::StagingAreaDBName
  - User::DatFilename
  - User::DMScriptFilename
  - User::SAScriptFilename

Mathematical Functions

Date/Time Functions

NULL Functions

Type Casts

Operators

Description:

Expression: `@[$Project::StagingAreaDBName]`

Evaluated value:

Evaluate Expression OK Cancel

A collection of expressions. The evaluation result of each expression is assigned to a property and replac...

Connection Managers

Create DM Tables Create DM Tables

Error List Variables Output

Add to Source Control

Ready

File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Search Data Mart

Develop Default Start

Data Mart ETL Package.dtsx [Design]\*

Control Flow Data Flow Parameters Event Handlers Package Explorer Execution Results

localhost.ARPAD\_StagingArea Connection

Properties

Both properties are now updated with the correct project parameters.

Create Staging Area Database

Property Expressions Editor

Property expressions:

Property	Expression
ServerName	@[Project::ServerName]
InitialCatalog	@[Project::StagingAreaDBName]

OK Cancel

Foreach SA

Ex

Foreach

Load DAT File into Table

Connection Managers

Create DM Tables Create SA Tables Load DAT Files localhost.ARPAD\_DataMart localhost.ARPAD\_StagingArea

Error List Variables Output

Add to Source Control

Ready

Notifications Solution Explorer Git Changes Properties

UserName

Show Editor

Expressions

A collection of expressions. The evaluation result of each expression is assigned to a property and replac...

File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Search Data Mart

Develop Default Star

Data Mart ETL Package.dtsx [Design]\*

Control Flow Data Flow Parameters Event Handlers Package Explorer

Server Explorer Toolbox SSIS Toolbox

Create Starting Area Database

Property Expressions Editor

Property expressions:

Property	Expression
ServerName	[\$Project::ServerName]
InitialCatalog	[\$Project::DataMartDBName]

OK Cancel

Connection Managers

Create DM Tables Create SA Tables Load DAT Files localhost.ARPAD\_DataMart

Error List Variables Output

Add to Source Control

Perform exactly the same steps in the Data Mart connection manager to map the *ServerName* and *InitialCatalog* properties. The result should be the one presented in the picture.

ConnectByProxy False  
ConnectionManagerType OLEDB  
ConnectionString Data Source=localhost;In  
ConnectRetryCount 1  
ConnectRetryInterval 5  
ConnectUsingManaged False  
DataSourceID  
Validation False  
ValidationExpressions False  
ValidationCatalog {30BA7684-F250-479F-BD1  
ValidationDatabase ARPAD\_DataMart  
ValidationConnectionString localhost.ARPAD\_DataM:  
ValidationUserName \*\*\*\*\*  
ValidationPassword  
NameConnectivity False  
Name localhost  
Name DTC Transaction Isolation True  
UserName

Show Editor

Expressions

A collection of expressions. The evaluation result of each expression is assigned to a property and replac...

File Edit View Git Project Build Debug  
Window Help

Data Mart ETL Package.dtsx [Design]\*

Control Flow Data Flow Parameters Events

Server Explorer Toolbox SSIS Toolbox

Create Staging Area Database

Foreach SA Create Table Script

Execute SA Script

Create Data Mart Database

Foreach DM Create Table Script

Execute DM Script

Foreach DAT File

Load DAT File into Table

Connection Managers

- Create DM Tables
- Create SA Tables
- Load DAT Files
- localhost.ARPAD\_DataMart
- localhost.ARPAD\_Staging
- localhost.master

Error List Variables Output

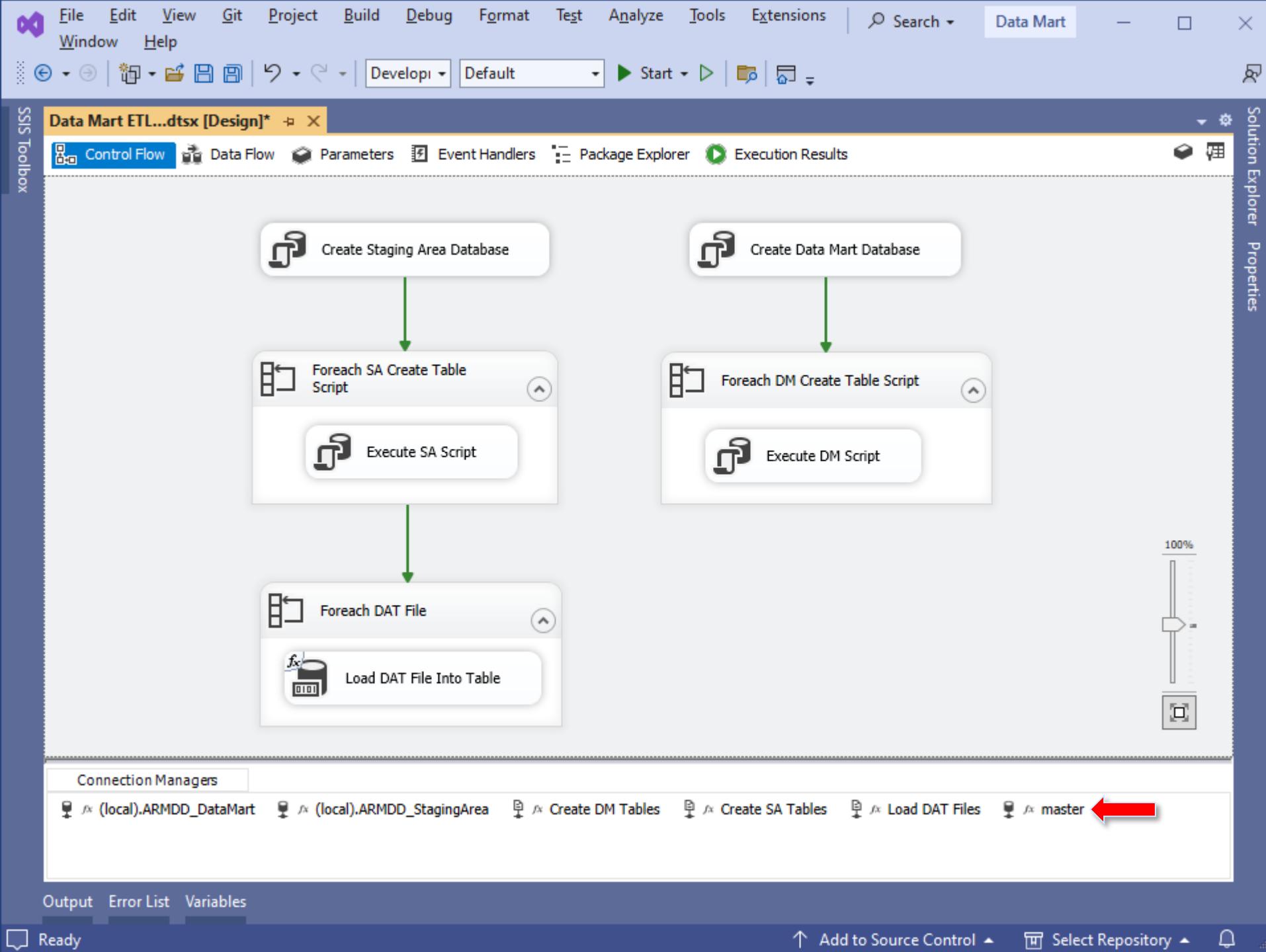
Add to Source Control Properties

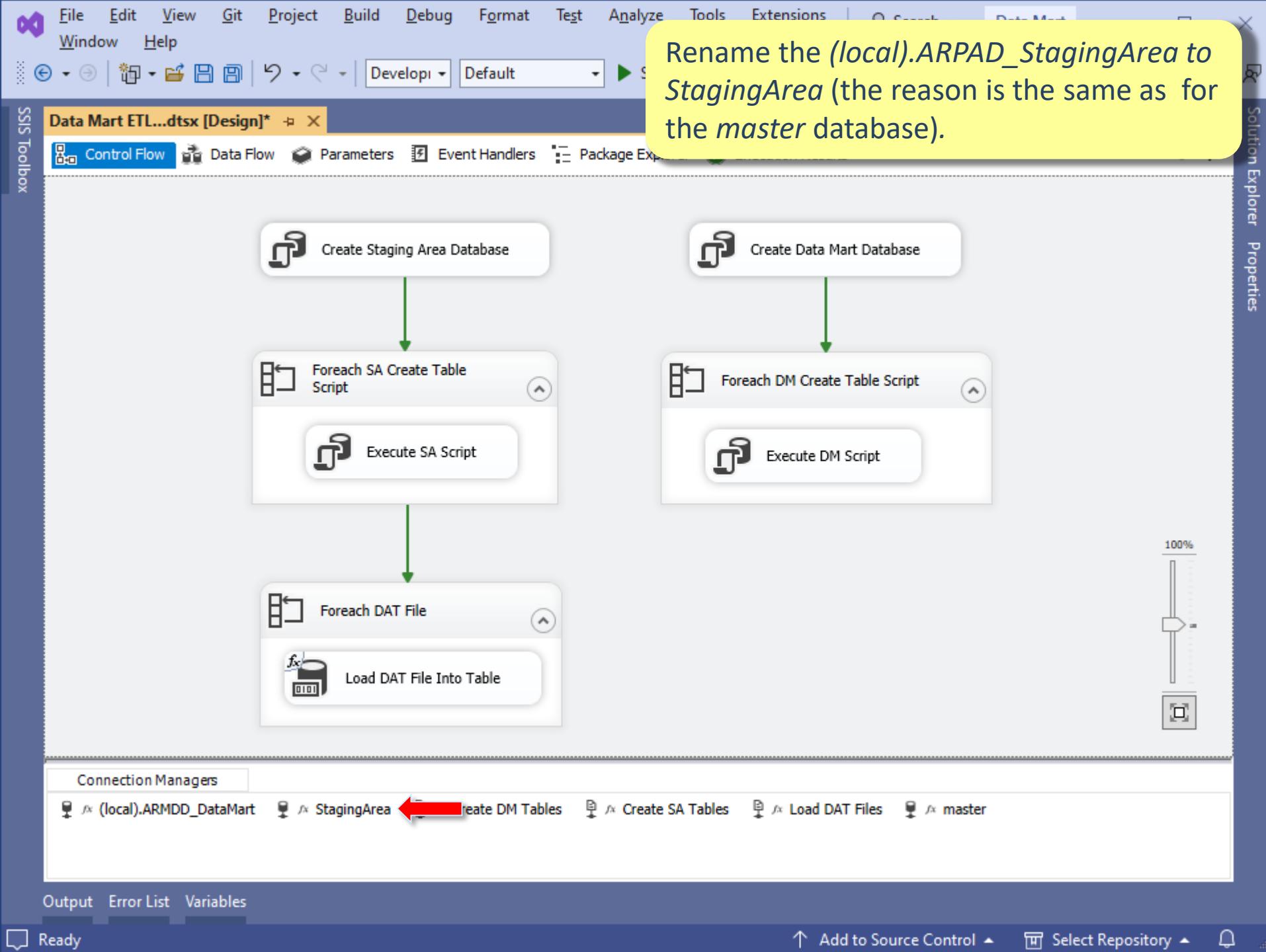
Rename

Notifications Solution Explorer Git Changes

Rename the connection manager *localhost.master* to *master*. After being parameterized, the master database can be in any server (depending just of the *ServerName* project parameter) and not only in *localhost*.

```
graph TD; SA[Create Staging Area Database] --> SA_Foreach[Foreach SA Create Table Script]; SA_Foreach --> SA_Execute[Execute SA Script]; SA_Execute --> DAT_Foreach[Foreach DAT File]; DAT_Foreach --> DAT_Load[Load DAT File into Table]; DM[Create Data Mart Database] --> DM_Foreach[Foreach DM Create Table Script]; DM_Foreach --> DM_Execute[Execute DM Script];
```





File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help

Develop Default Start

Data Mart ETL...dtsx [Design] \* X

Control Flow Data Flow Parameters Event Handlers Package Explorer

Create Staging Area Database

Execute SA Script

Load DAT File Into Table

Create Data Mart Database

Execute DM Script

Foreach SA Create Table Script

Foreach DAT File

Foreach DM Create Table Script

100%

Connection Managers

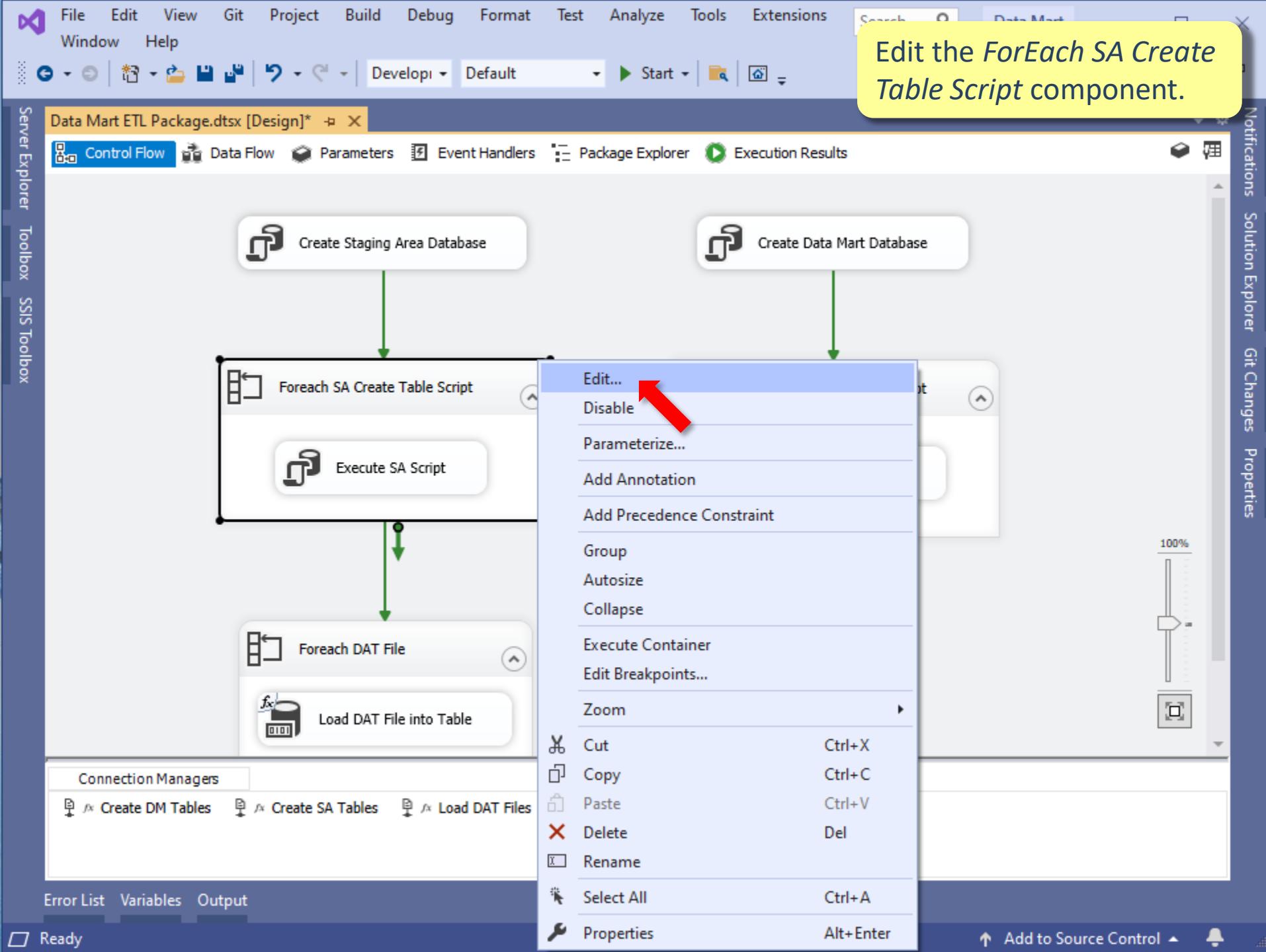
DataMart (selected) ← StagingArea Create DM Tables Create SA Tables Load DAT Files master

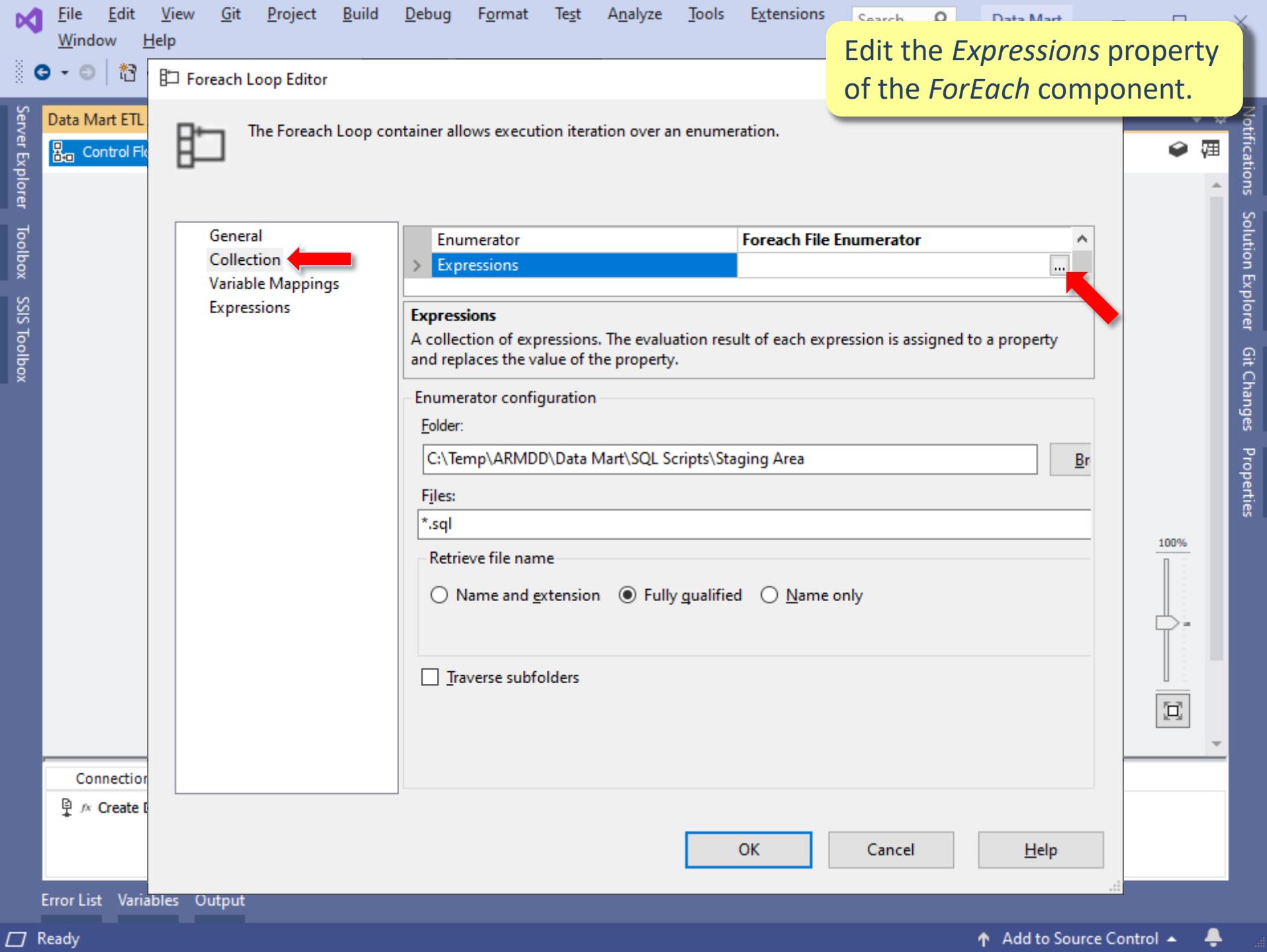
Output Error List Variables

Ready Add to Source Control Select Repository

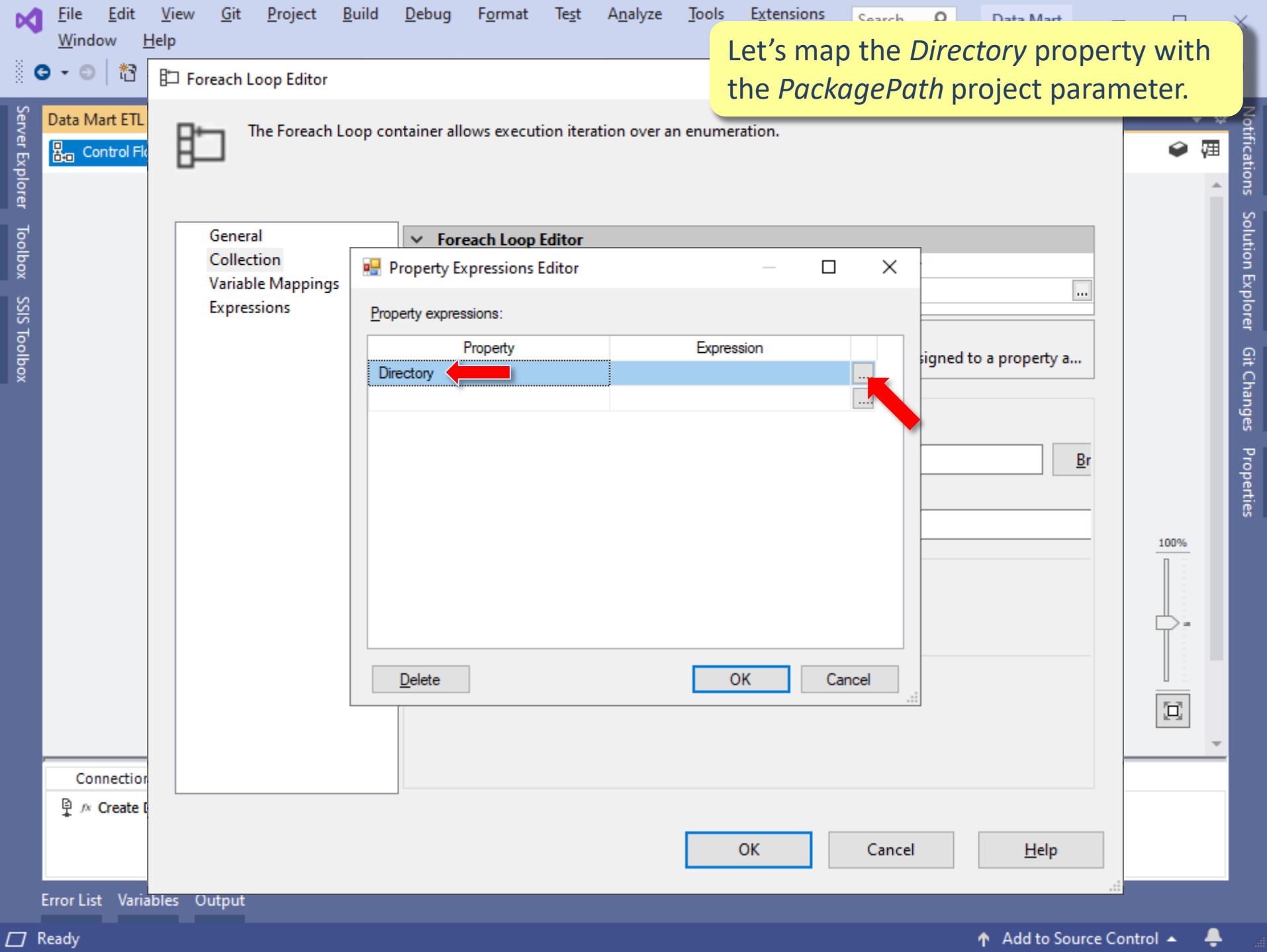
Rename the *(local).ARPAD\_DataMart* to *DataMart* (the reason is the same as for the *master* database).

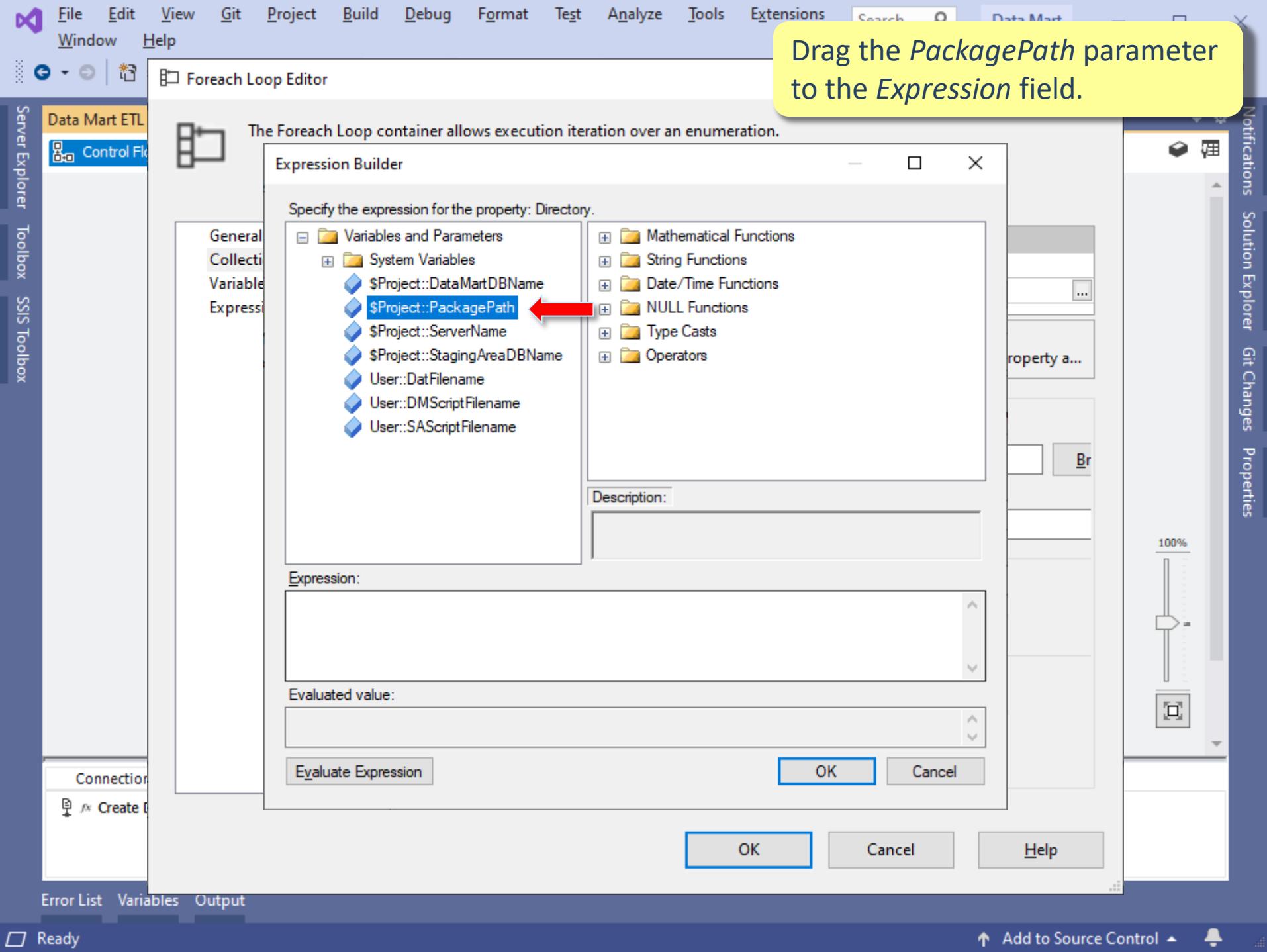
The diagram illustrates the SSIS Control Flow for a Data Mart ETL package. It begins with two parallel tasks: 'Create Staging Area Database' and 'Create Data Mart Database'. Both lead into a 'Foreach SA Create Table Script' loop, which contains an 'Execute SA Script' task. This is followed by a 'Foreach DAT File' loop, which contains a 'Load DAT File Into Table' task. Finally, the process moves to a 'Foreach DM Create Table Script' loop, containing an 'Execute DM Script' task.

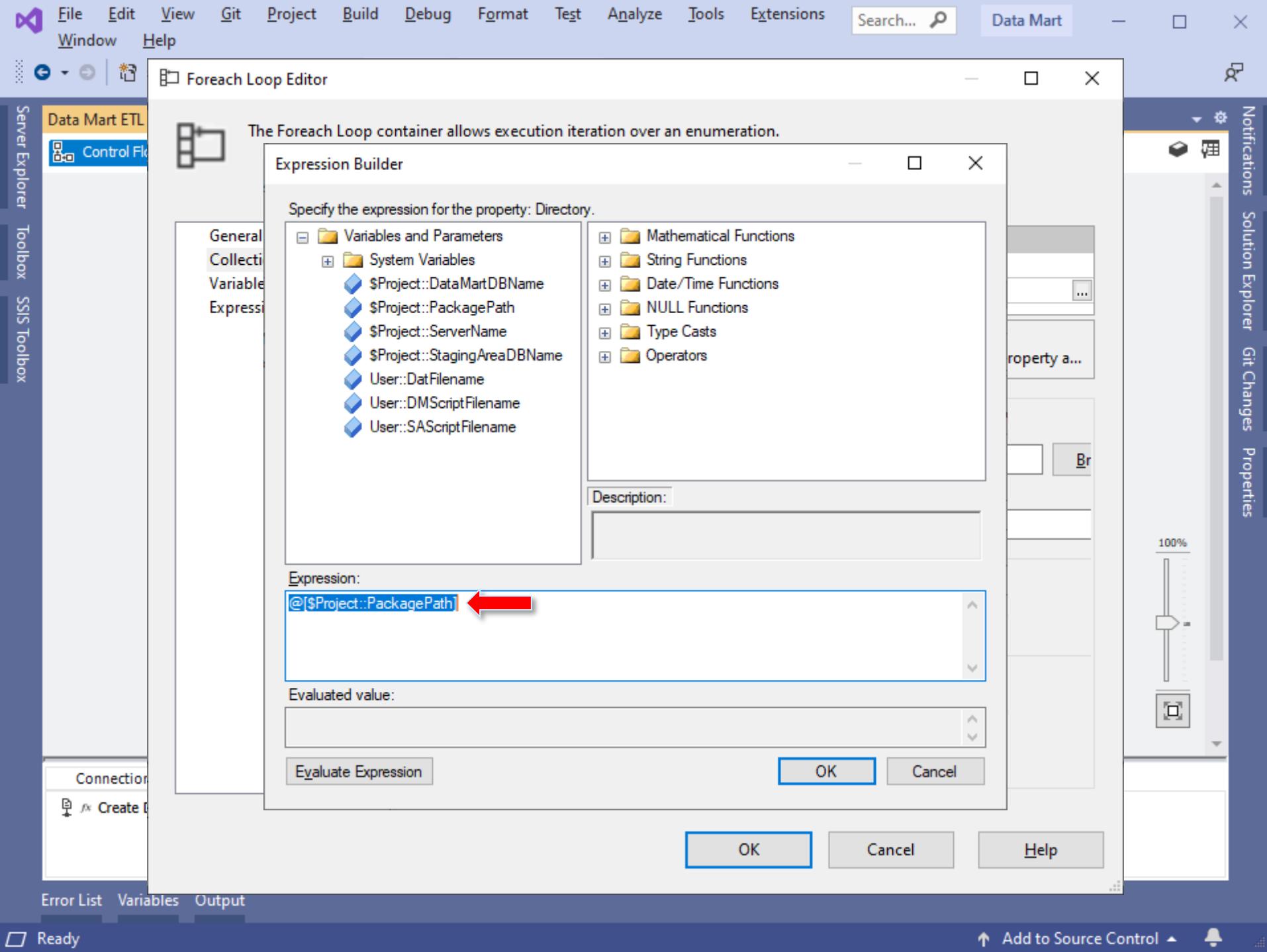


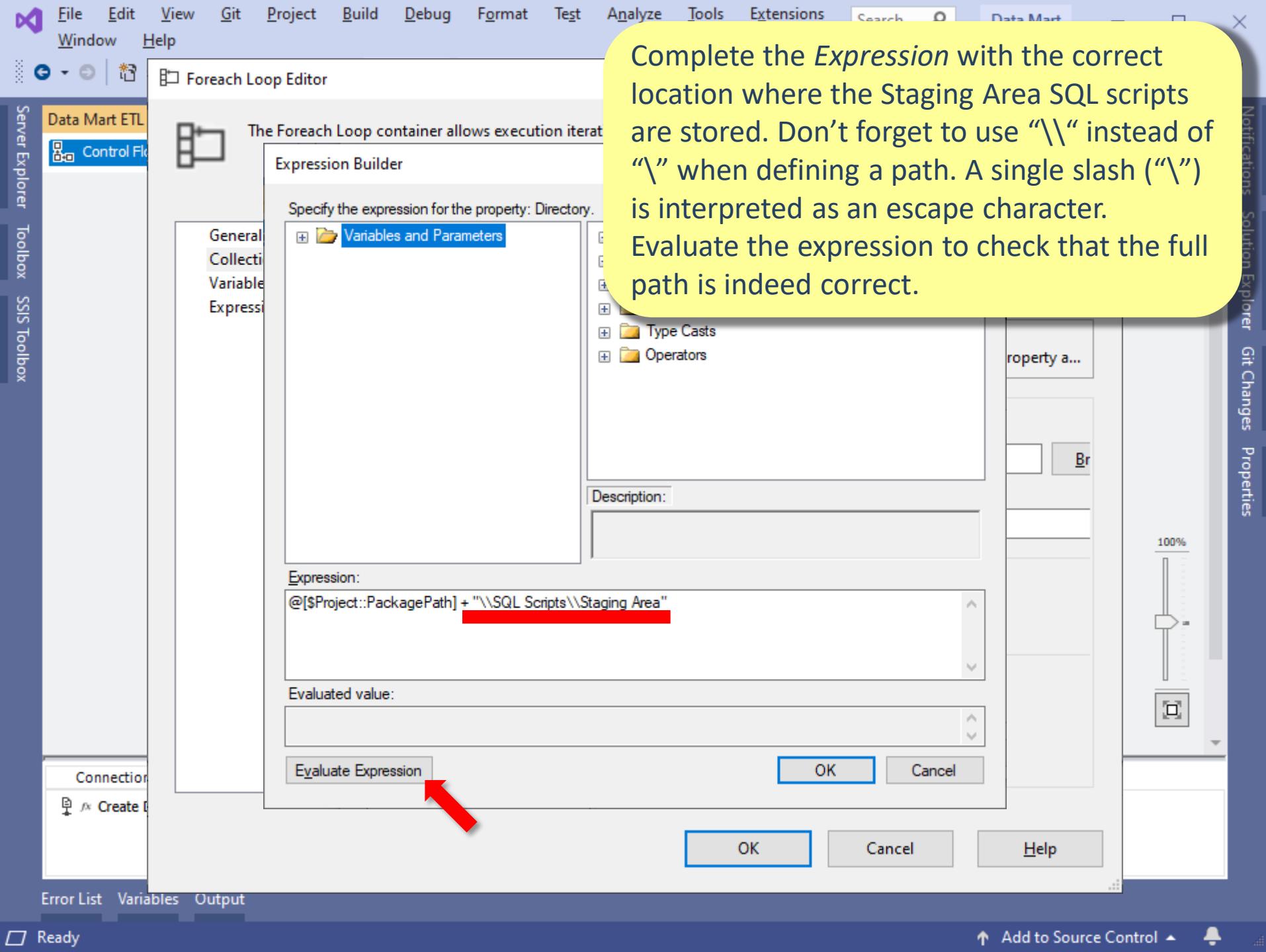


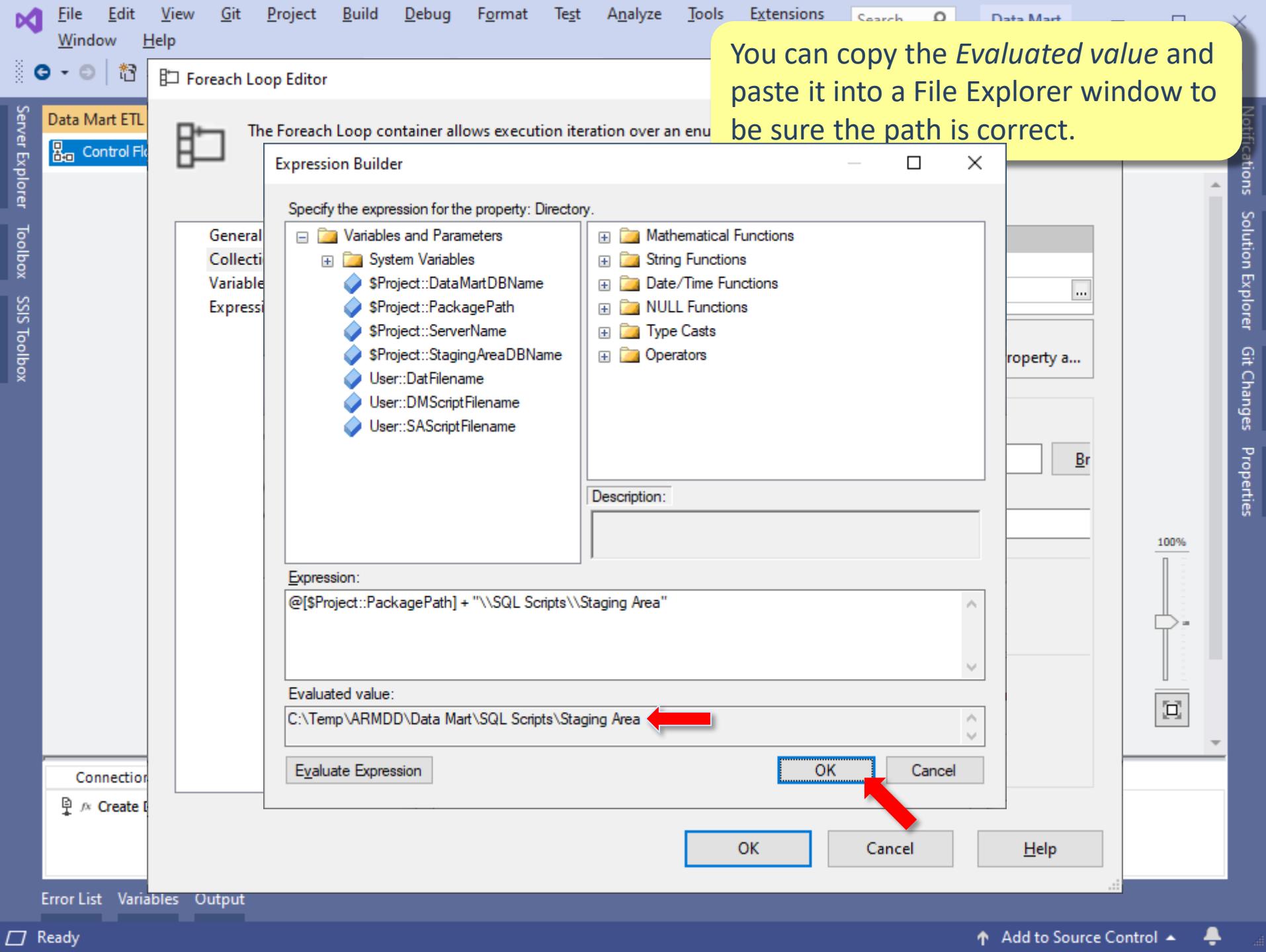
Edit the *Expressions* property of the *ForEach* component.

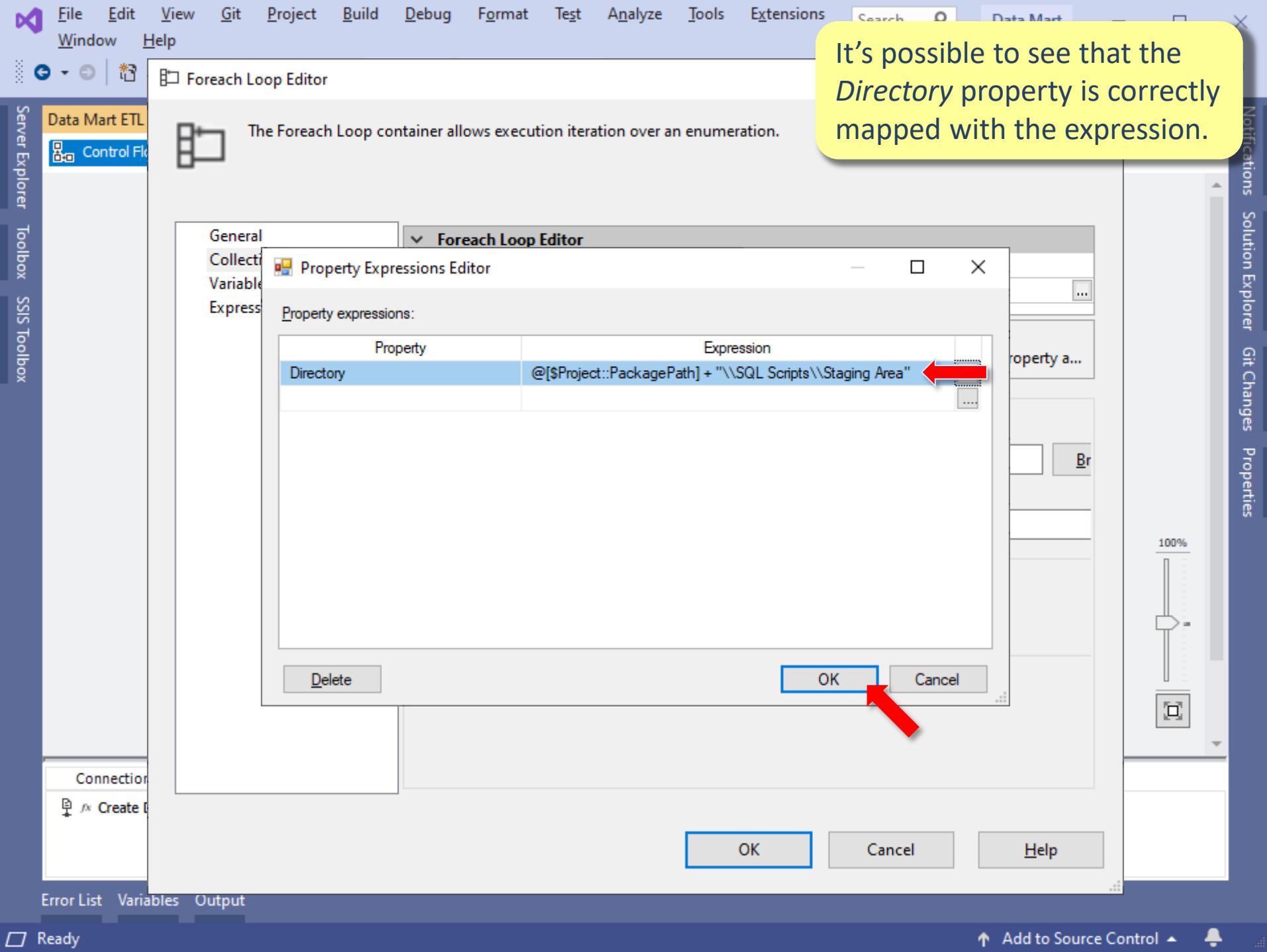


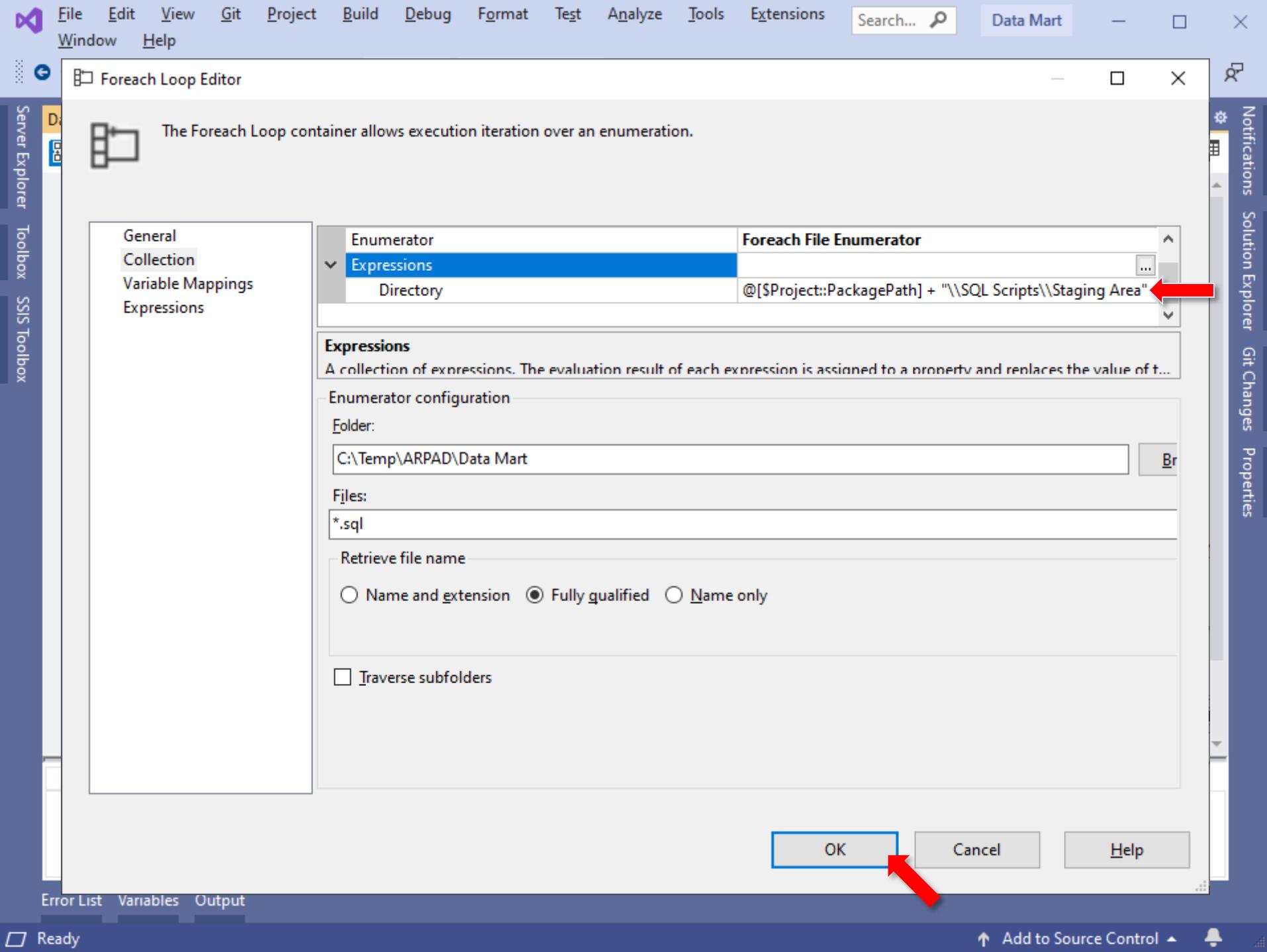


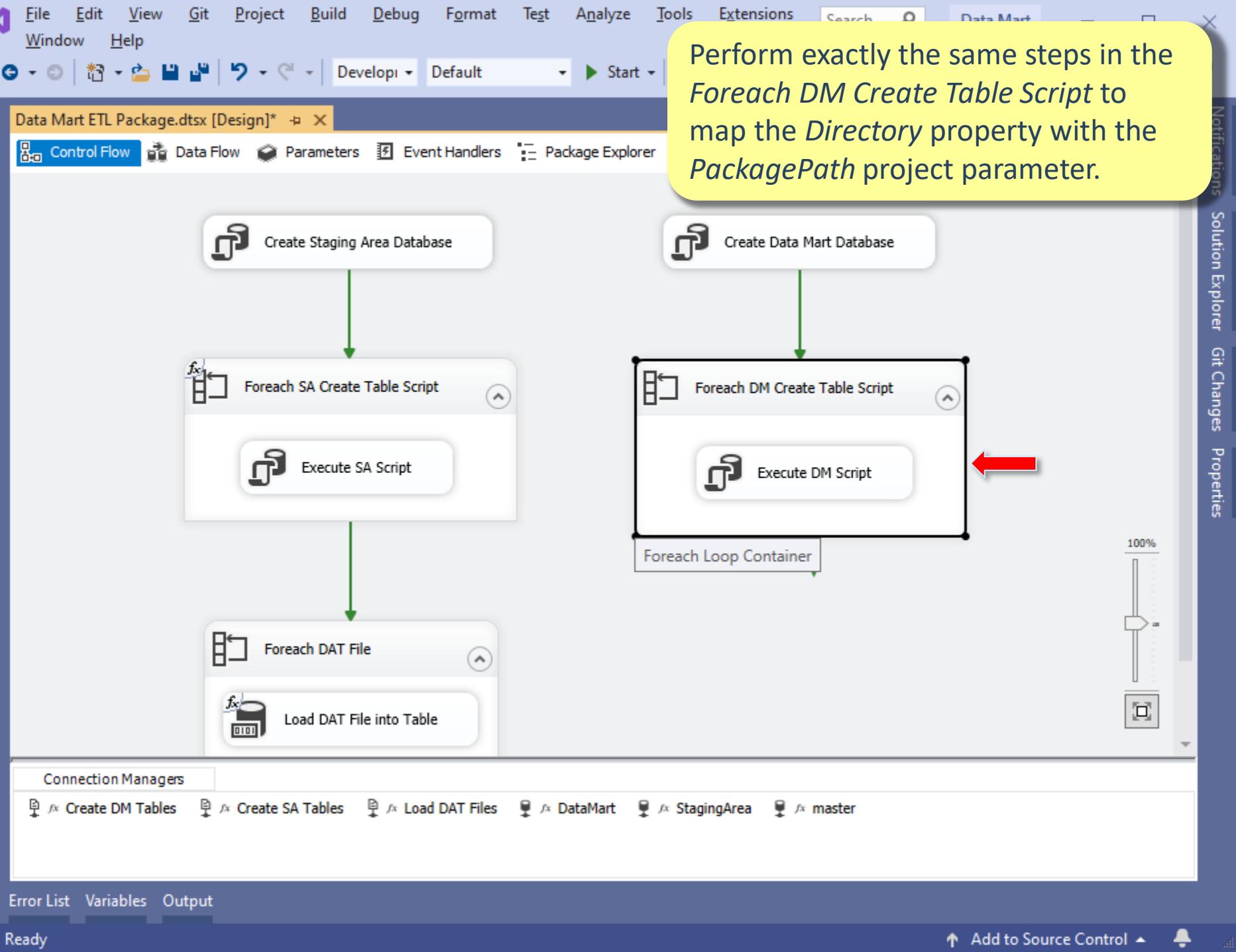


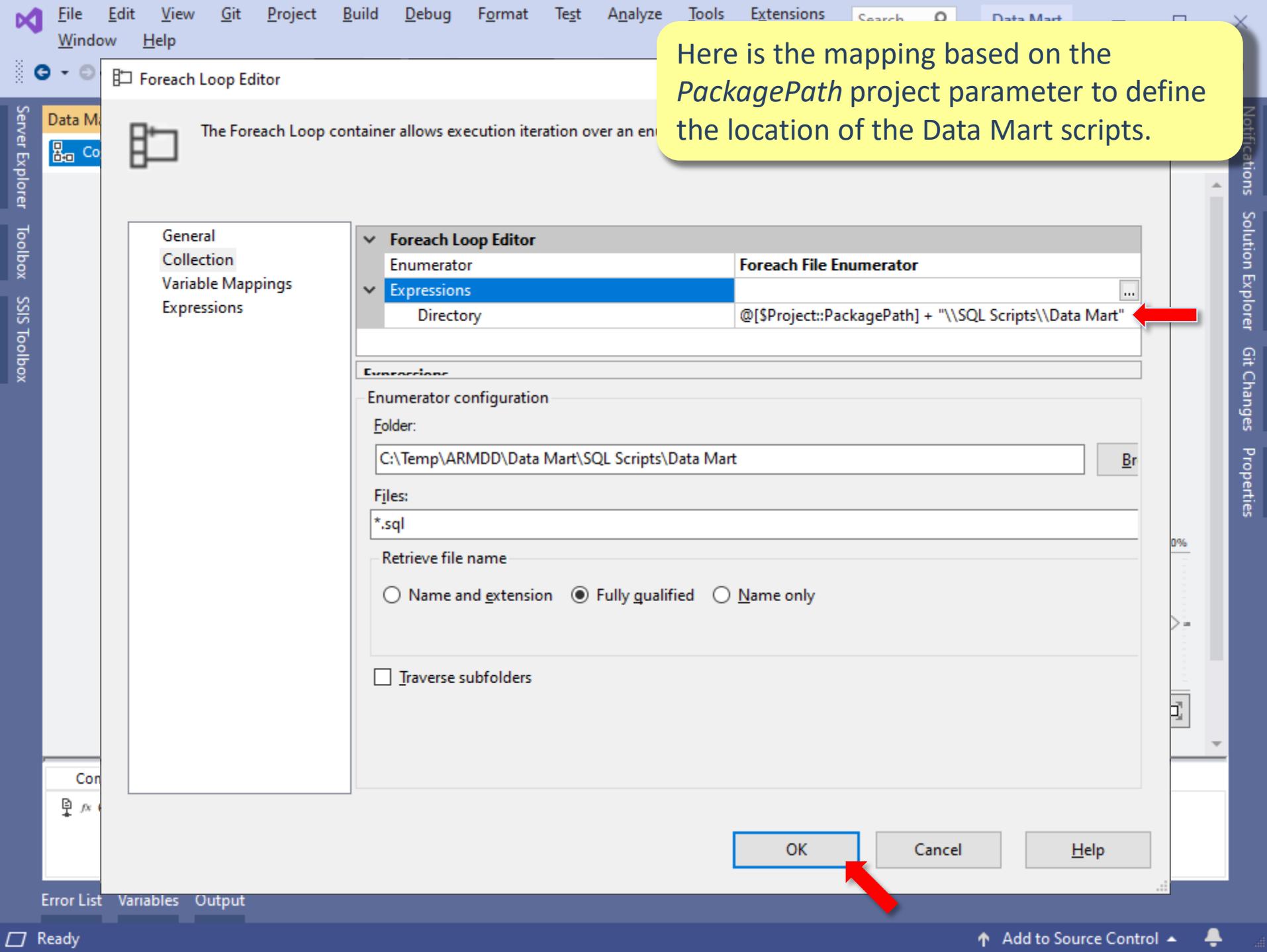




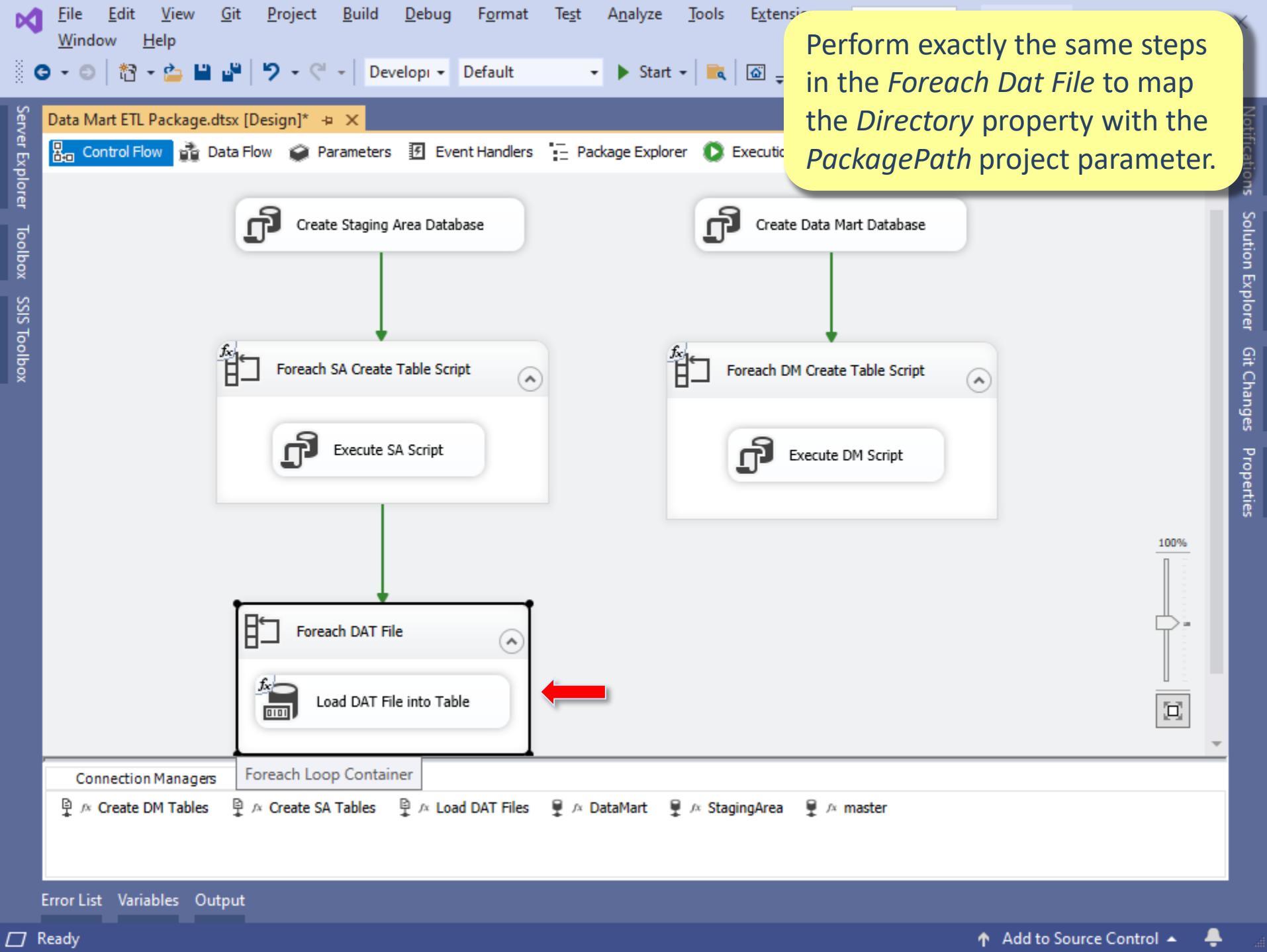


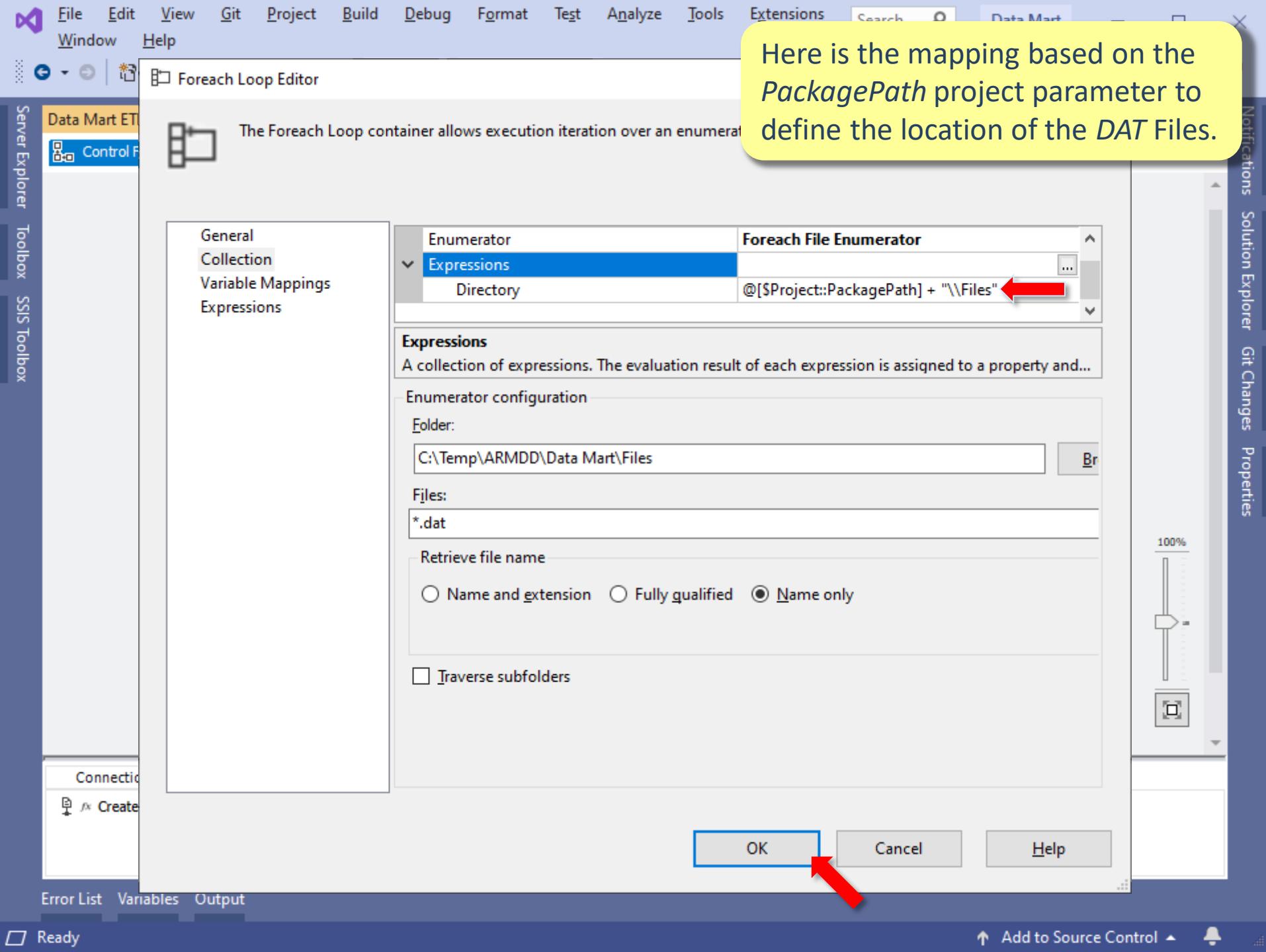


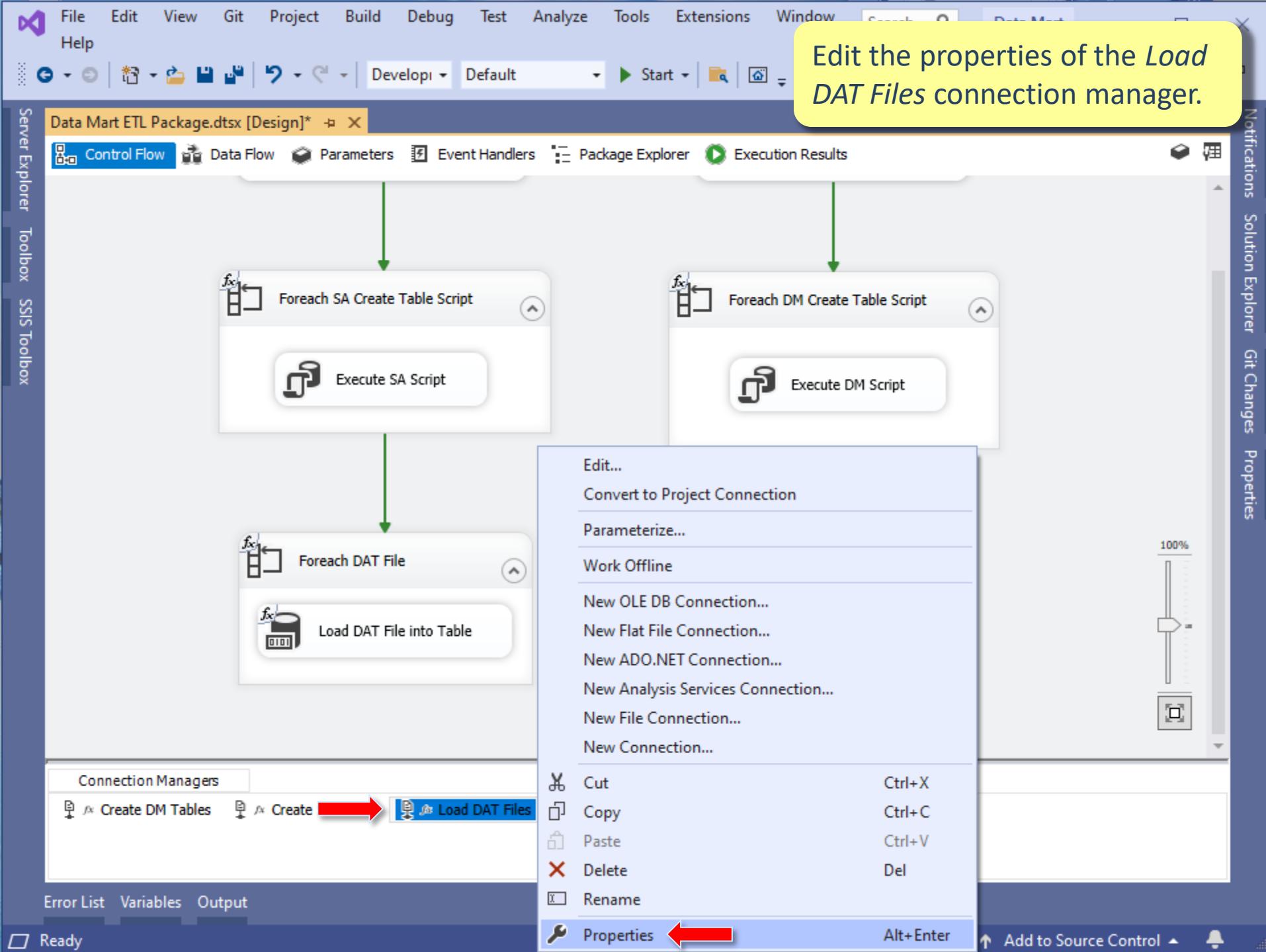




Here is the mapping based on the *PackagePath* project parameter to define the location of the Data Mart scripts.







File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Search... Data Mart

Develop Default Start

Data Mart ETL Package.dtsx [Design]\*

Control Flow Data Flow Parameters Event Handlers Package Explorer Execution Result

Server Explorer Toolbox SSIS Toolbox

Properties

**Load DAT Files Connection**

Identification  
PackagePath \Package.Connections[Load DAT Files]

Misc  
ConnectByProxy False  
ConnectionManagerType FILE  
ConnectionString C:\Temp\ARPAD\Data Mart

Expressions

FileUsageType 0  
HasExpressions True  
ID {0ED7F756-FC1F-4AD2-9127-  
Name Load DAT Files  
Qualifier  
SupportsDTCTransactions False

Edit the Expressions of the component.

Properties

Load DAT Files Connection

Identification  
PackagePath \Package.Connections[Load DAT Files]

Misc  
ConnectByProxy False  
ConnectionManagerType FILE  
ConnectionString C:\Temp\ARPAD\Data Mart

Expressions

FileUsageType 0  
HasExpressions True  
ID {0ED7F756-FC1F-4AD2-9127-  
Name Load DAT Files  
Qualifier  
SupportsDTCTransactions False

Show Editor

Expressions

A collection of expressions. The evaluation result of each expression is assigned to a property and replac...

Connection Managers

Create DM Tables Create SA Tables Load DAT Files DataMart StagingArea

Error List Variables Output

Add to Source Control

Ready

Notifications Solution Explorer Git Changes Properties

```
graph TD; SA[Execute SA Script] --> DAT[Load DAT File into Table]; DAT --> DM[Execute DM Create]; DM --> SA;
```

File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Search Data Mart

Help

Develop Default Start

Control Flow Data Flow Parameters Event Handlers Package Explorer Execution Result

Data Mart ETL Package.dtsx [Design]\*

Properties Load DAT Files Connection

Identification PackagePath \Package.Connections[Load]

ConnectionString "C:\\Temp\\ARMDD\\Data Mart"

ConnectionStringType FILE

ConnectionString C:\\Temp\\ARPAD\\Data Mart

ConnectionStringID {0ED7F756-FC1F-4AD2-9127-000000000000}

ConnectionStringValidation False

ConnectionStringType 0

ConnectionStringExpressions True

ConnectionStringLoad DAT Files

ConnectionStringIsDTCTransaction False

Property Expressions Editor

Property expressions:

Property	Expression
ConnectionString	"C:\\Temp\\ARMDD\\Data Mart"

Delete OK Cancel

Connection Managers

Create DM Tables Create SA Tables Load DAT Files DataMart StagingArea

Show Editor

Expressions

A collection of expressions. The evaluation result of each expression is assigned to a property and replac...

Error List Variables Output

Add to Source Control

Ready

Edit the value of the ConnectionString property.

The screenshot shows the SSIS Designer interface with a 'Property Expressions Editor' dialog open. The dialog lists a single property, 'ConnectionString', with its expression set to a local file path. A red arrow points to the ellipsis button in the expression column, which typically opens a 'ConnectionString Editor' for modification. The background shows a package containing a 'Foreach SA' loop and a 'Load DAT Files' task. The 'ConnectionString' property is also visible in the main Properties pane on the right.

File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Search Data Mart

Help

Develop Default Start

Data Mart ETL Package.dtsx [Design]\*

Properties

Control Flow Data Flow

Expression Builder

Specify the expression for the property: ConnectionString.

+ Variables and Parameters

+ Mathematical Functions  
+ String Functions  
+ Date/Time Functions  
+ NULL Functions  
+ Type Casts  
+ Operators

Description:

Expression:  
`"C:\\Temp\\ARMDD\\Data Mart\\Files\\\" + @[User::DatFilename] + ".dat"`

Evaluated value:

Evaluate Expression OK Cancel

Connection Managers

Create DM Tables

Error List Variables Output

Let's replace the existing path by the **PackagePath** project parameter.

Properties

Action

\Package.Connections[Load]

False

FILE

C:\Temp\ARPAD\Data Mart

False

0

True

{0ED7F756-FC1F-4AD2-9127-}

Load DAT Files

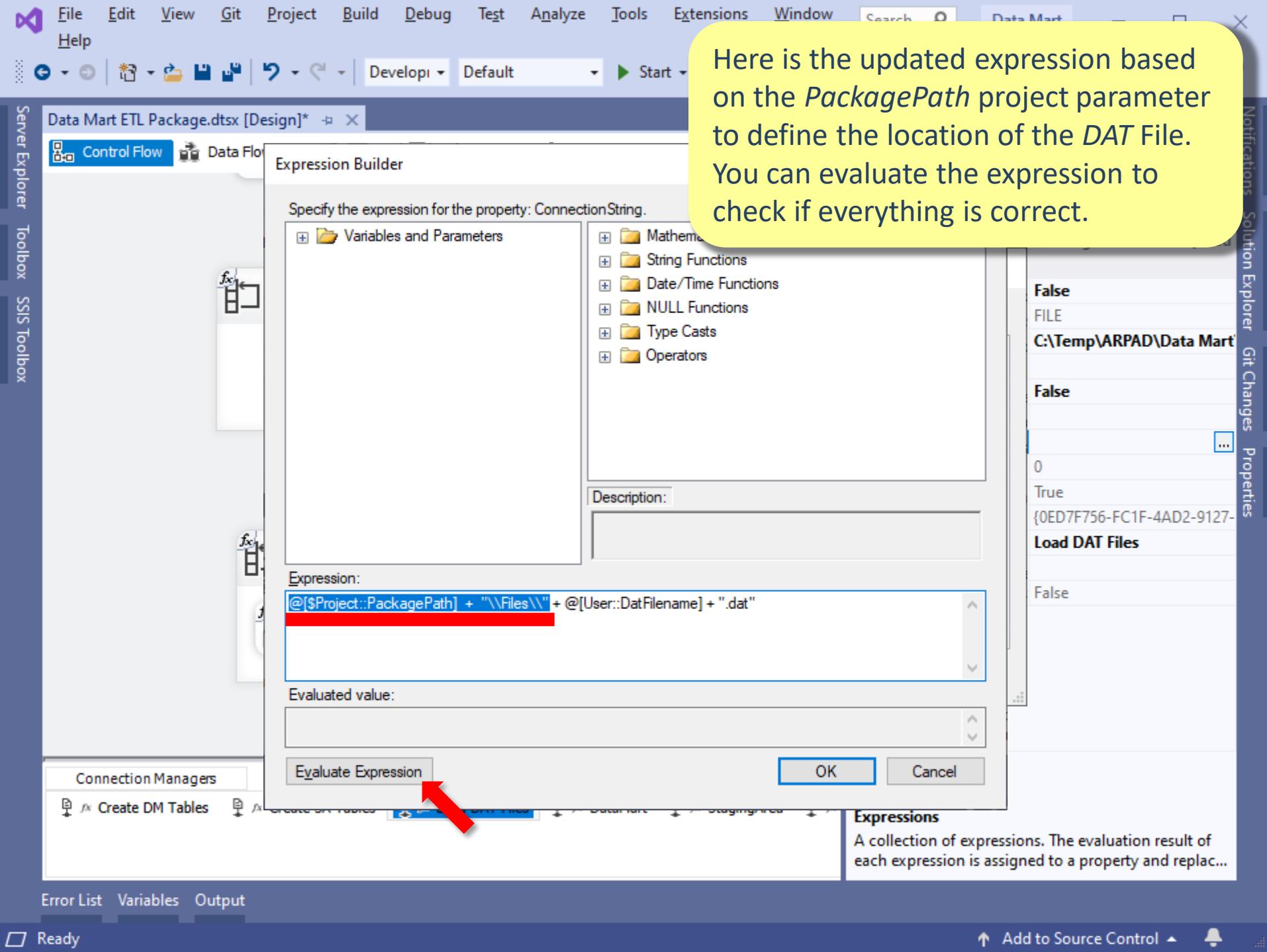
tic False

Add to Source Control

Ready

Notifications Solution Explorer Git Changes Properties

A collection of expressions. The evaluation result of each expression is assigned to a property and replac...



Here is the updated expression based on the *PackagePath* project parameter to define the location of the DAT File. You can evaluate the expression to check if everything is correct.

In the *Evaluated value* field, the *DatFilename* value is missing. This is expected because the variable (created on the last tutorial) only gets its value at runtime.

Specify the expression for the property: ConnectionString

Variables and Parameters

- System Variables
  - \$Project::DataMartDBName
  - \$Project::PackagePath
  - \$Project::ServerName
  - \$Project::StagingAreaDBName
  - User::DatFilename
  - User::DMSScriptFilename
  - User::SAScriptFilename

Mathematical Functions

String Functions

Date/Time Functions

NULL Functions

Type Casts

Operators

Description:

Expression:

```
@[User::DatFilename]
```

Evaluated value:

C:\Temp\ARMDD\Data Mart\Files\dat

Evaluate Expression OK Cancel

Expressions

A collection of expressions. The evaluation result of each expression is assigned to a property and replac...

File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Search Data Mart

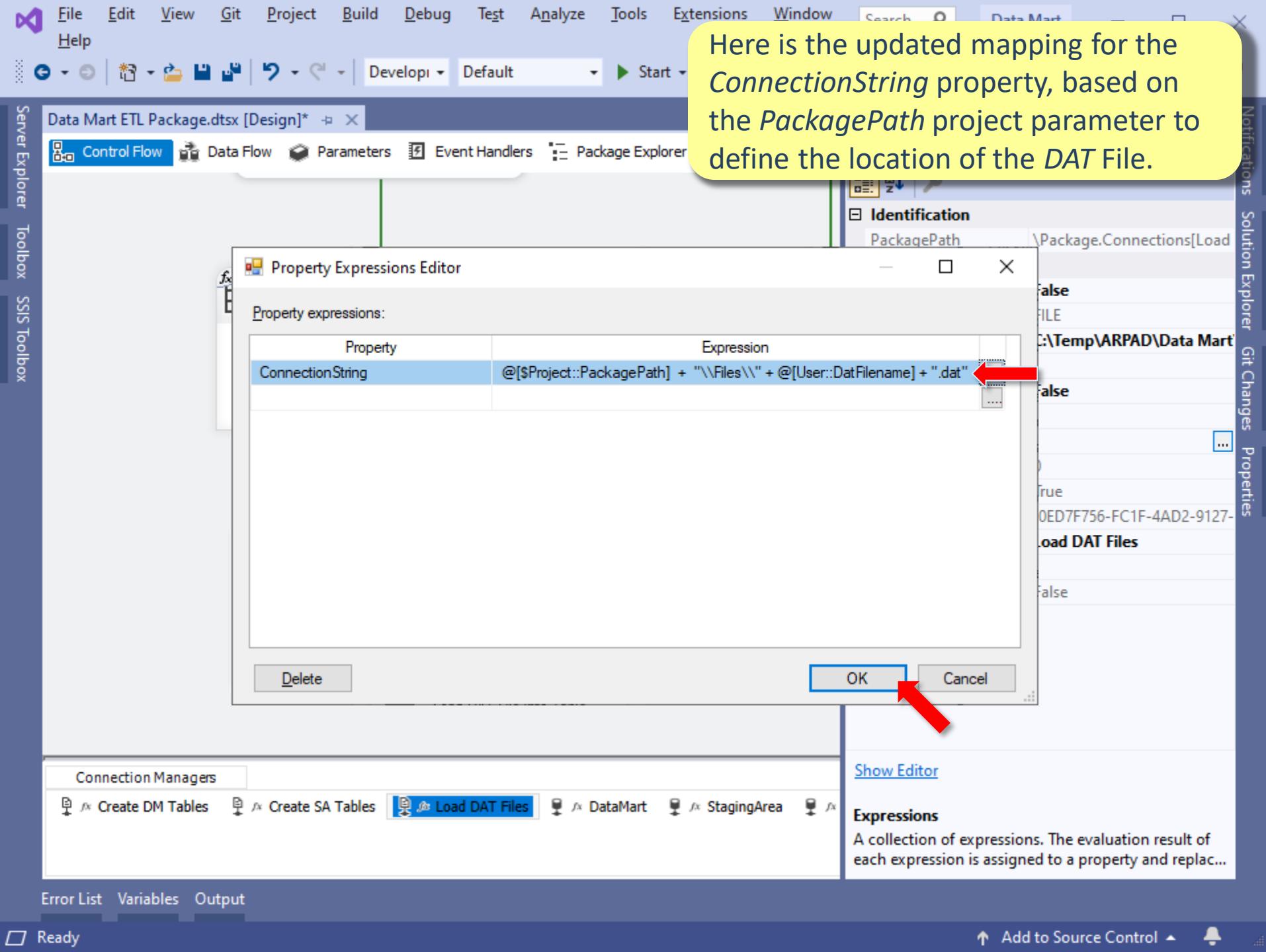
Develop Default

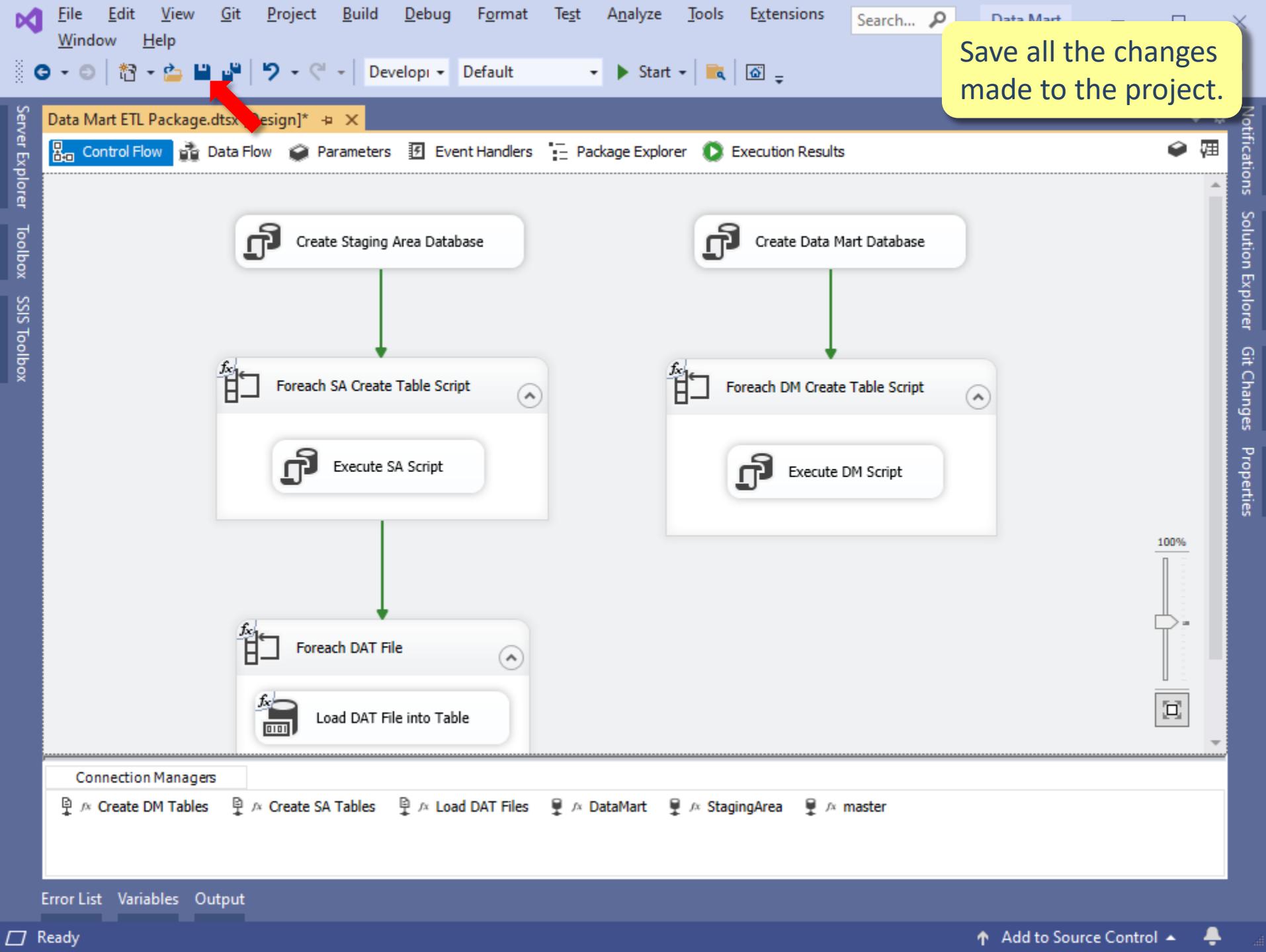
Data Mart ETL Package.dtsx [Design]\*

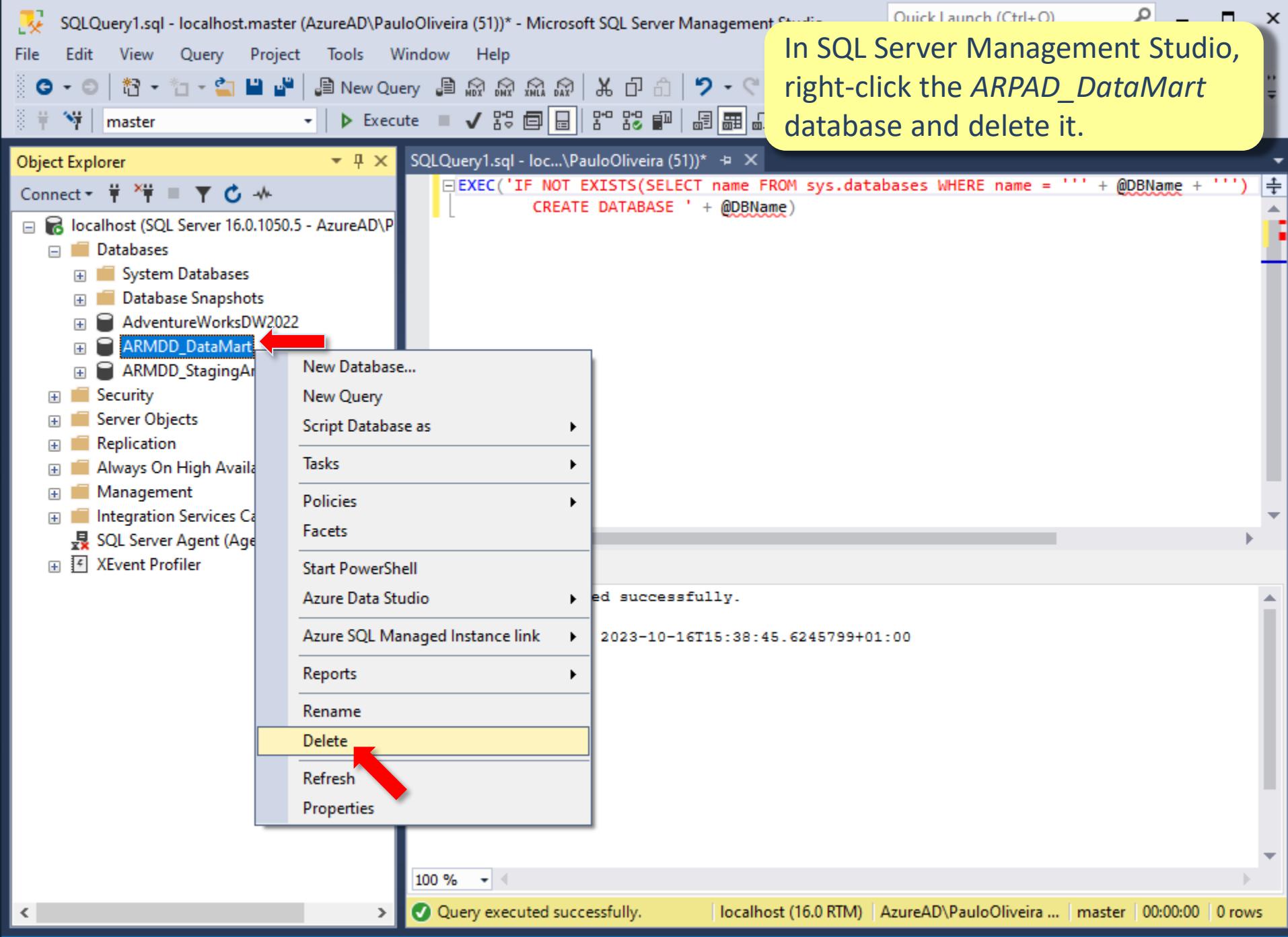
Control Flow Data Flow

Server Explorer Toolbox SSIS Toolbox

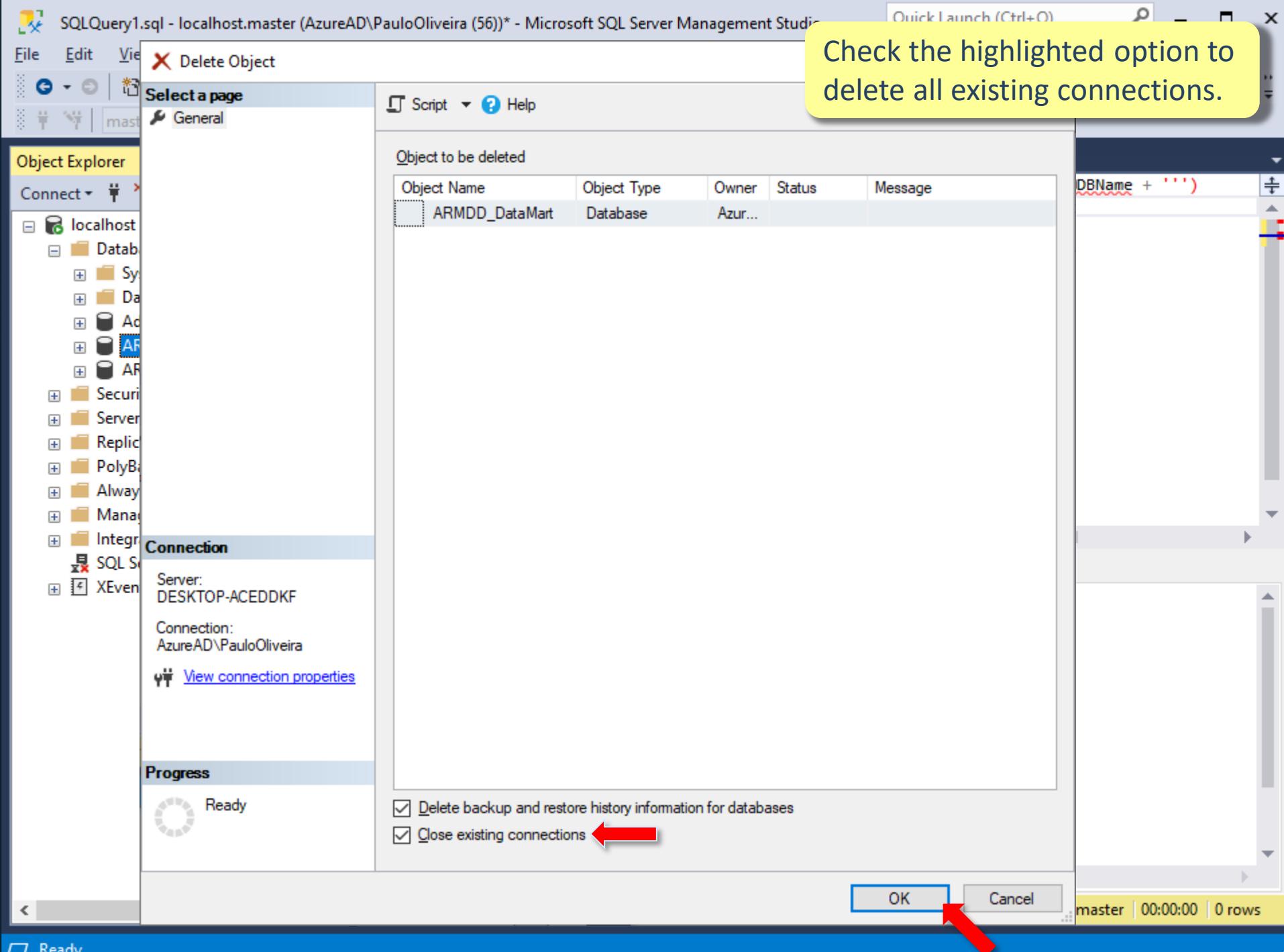
Add to Source Control Ready

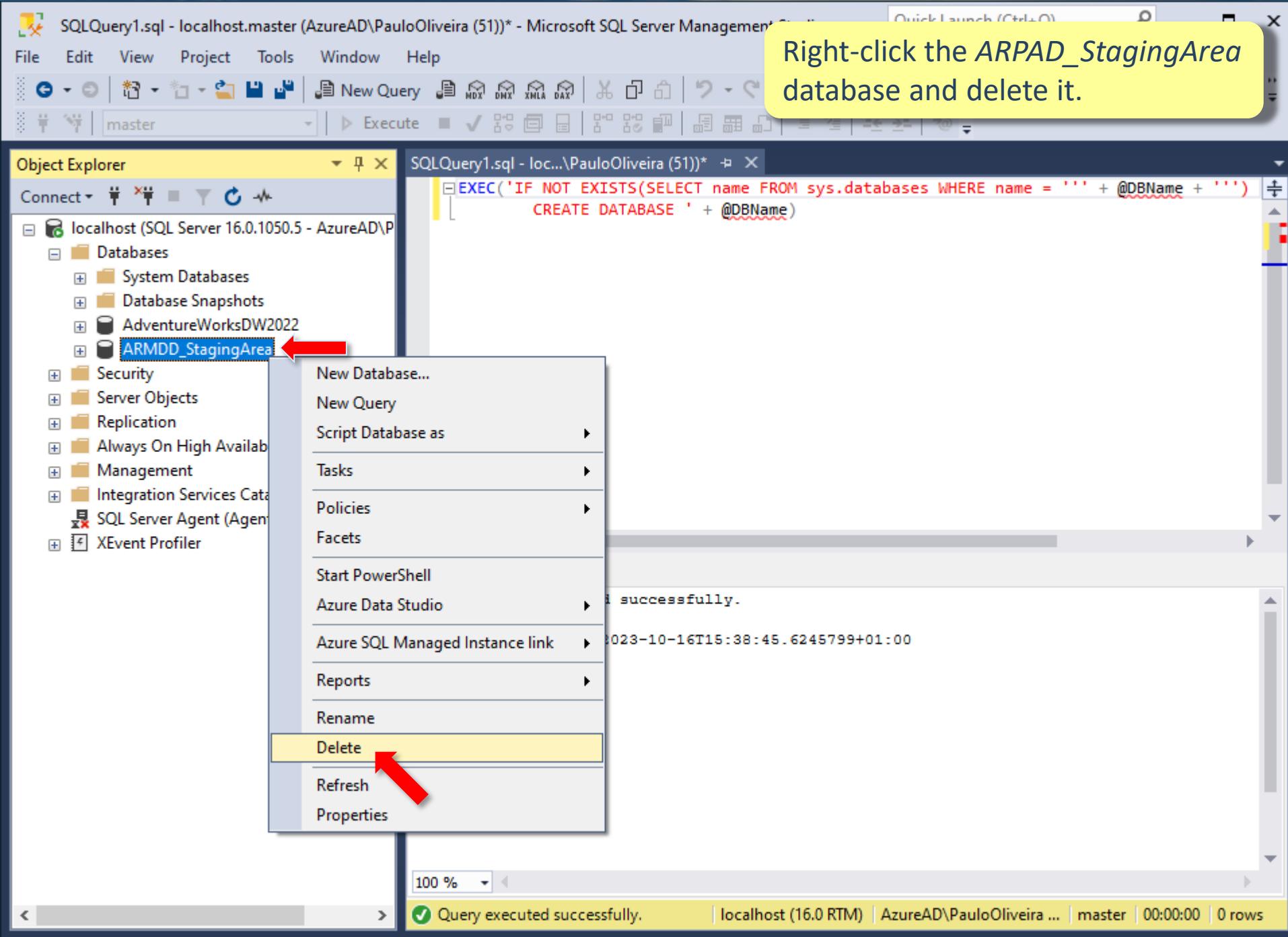


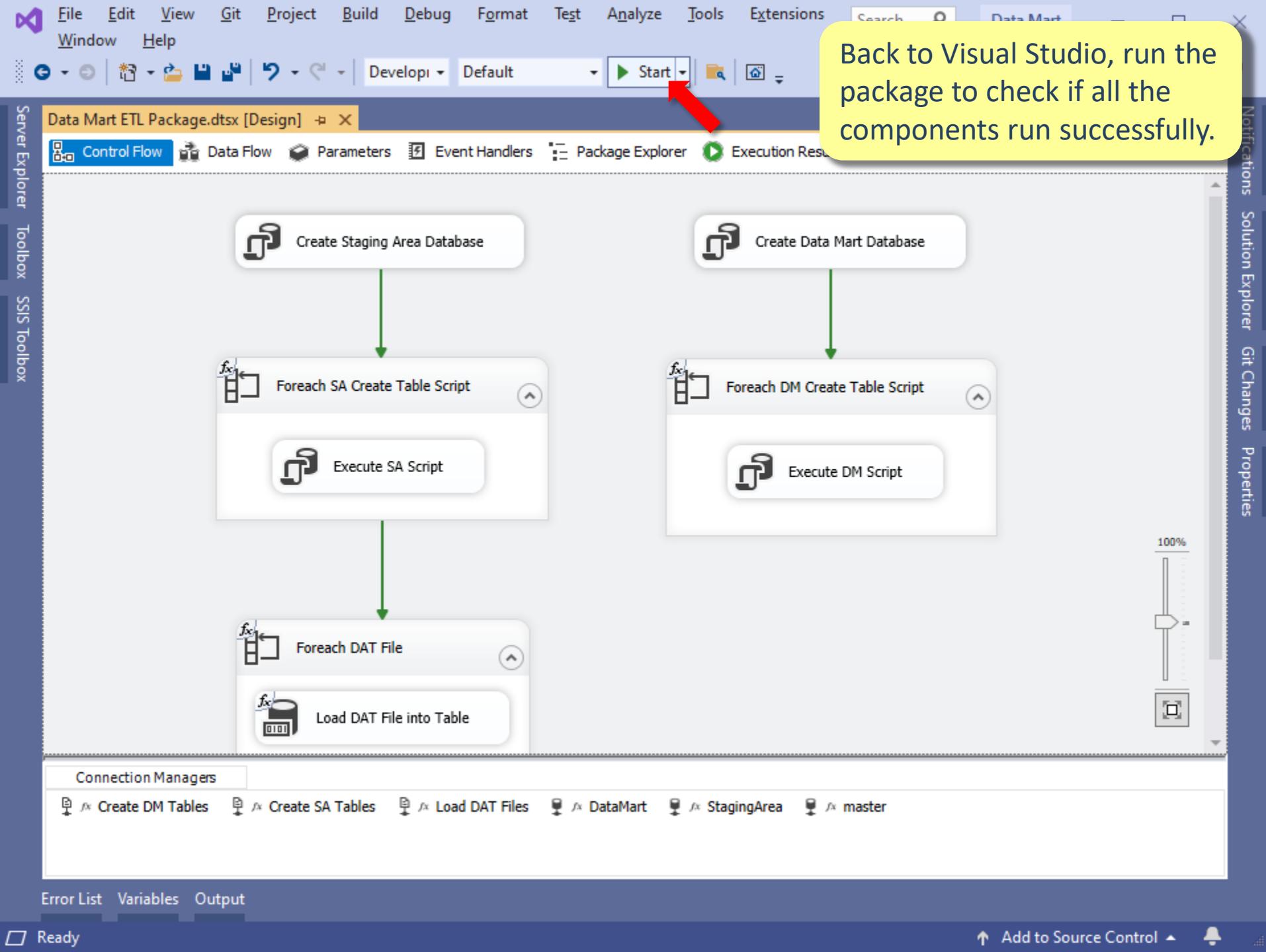


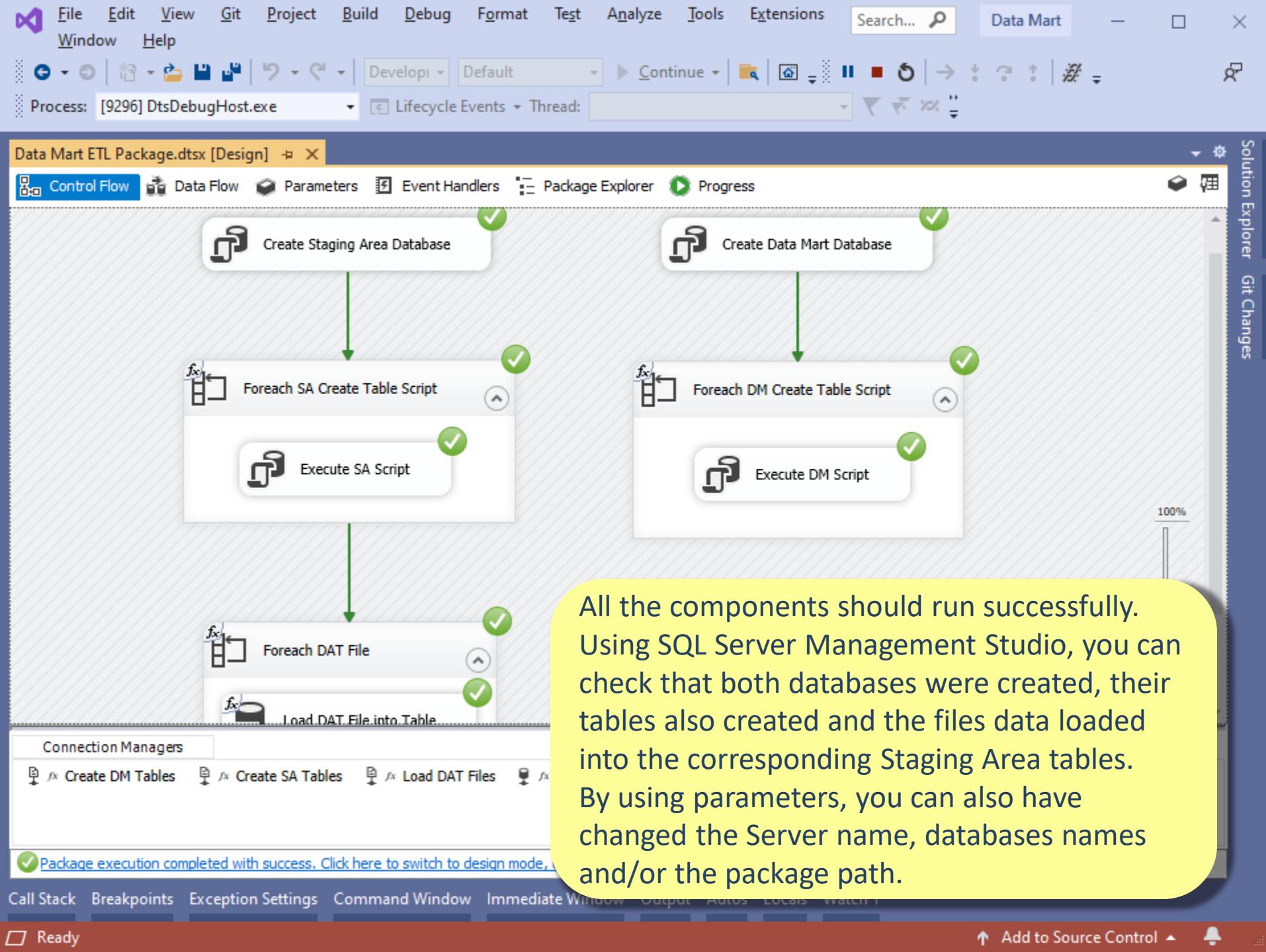


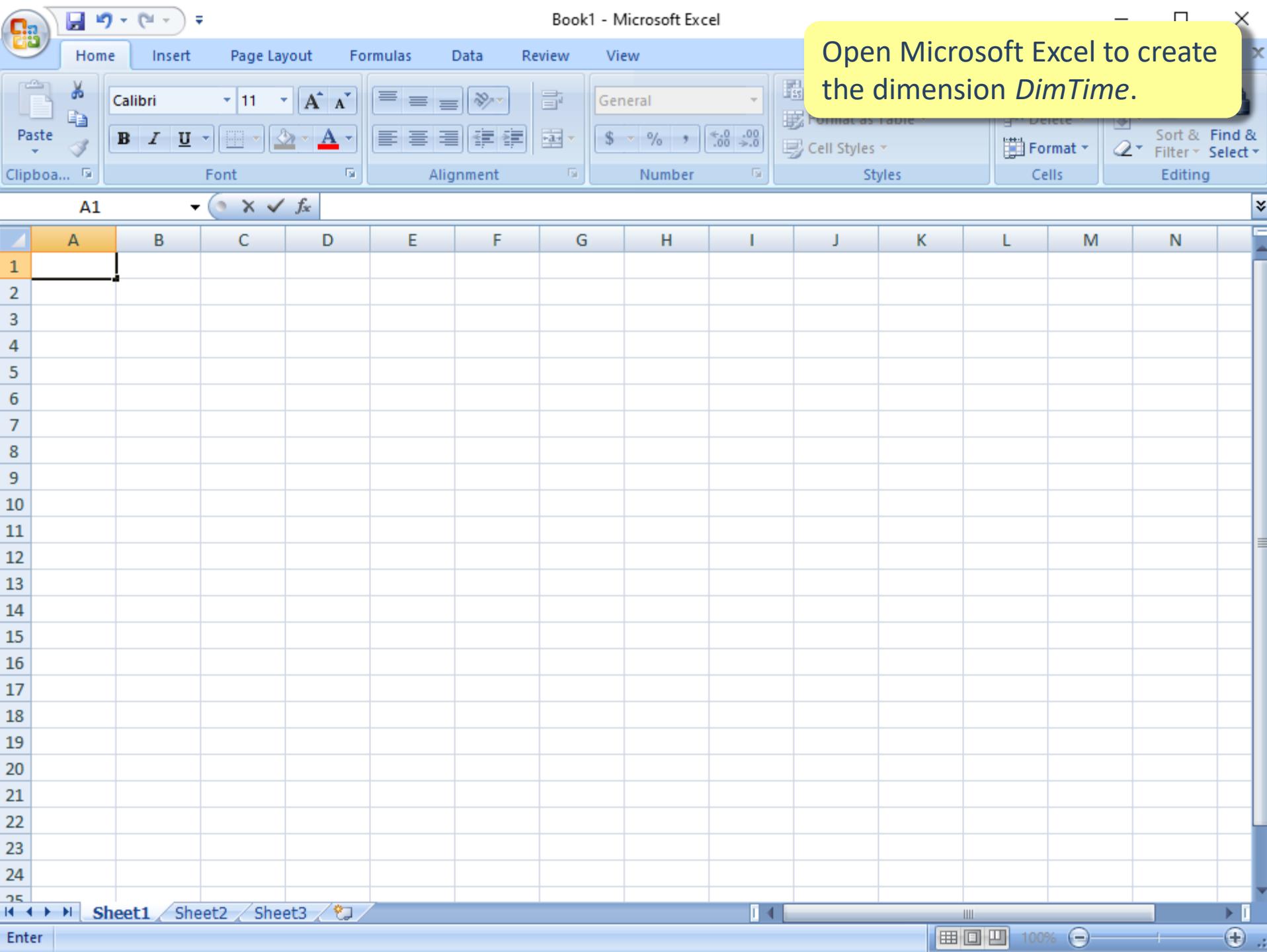
In SQL Server Management Studio, right-click the **ARPAD\_DataMart** database and delete it.



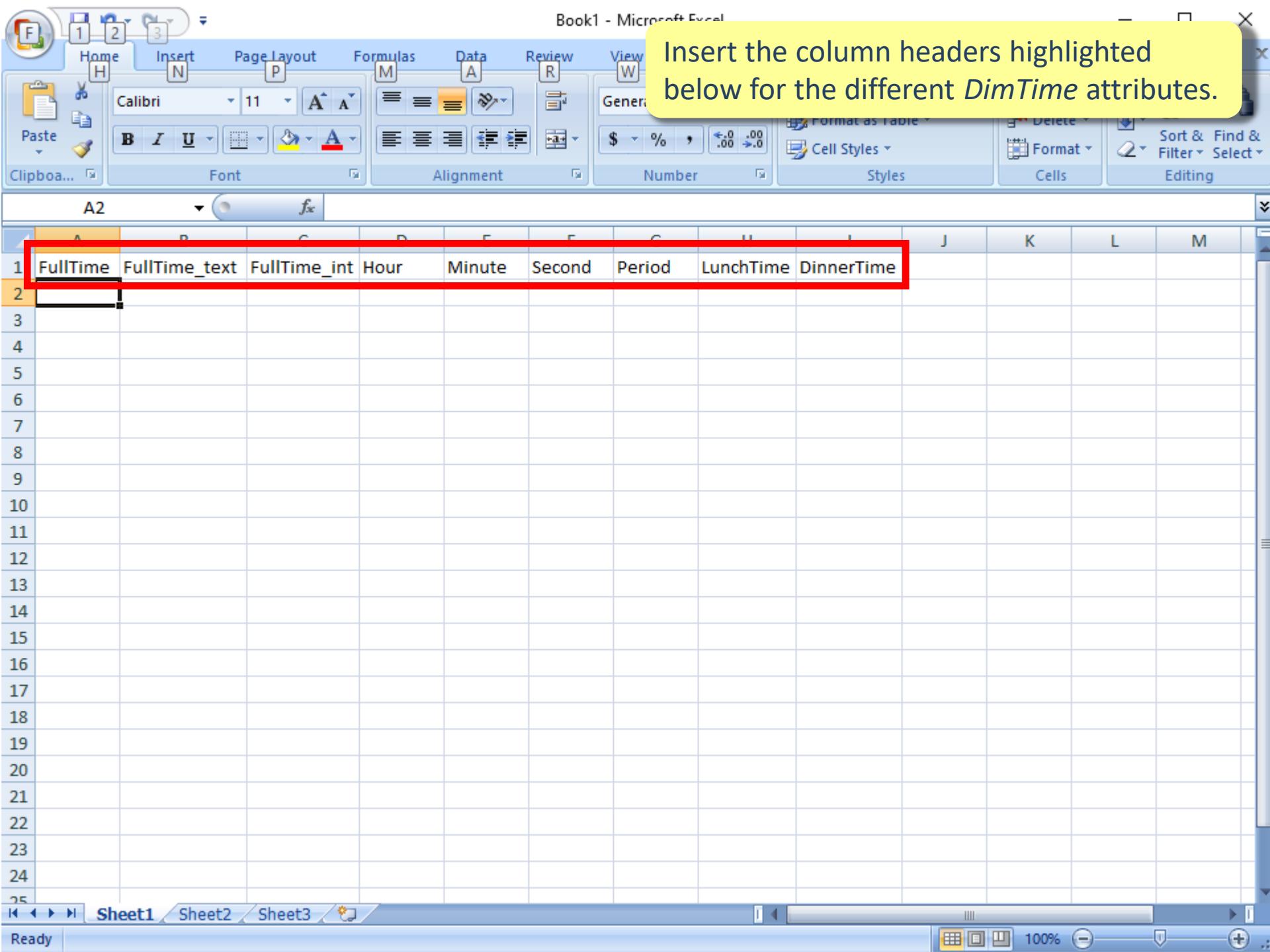








Open Microsoft Excel to create  
the dimension *DimTime*.



Insert the column headers highlighted below for the different *DimTime* attributes.

Book1 - Microsoft Excel

Home Insert Page Layout Formulas Data

Paste Clipboa...

Calibri 11 A A

B I U

Font Alignment

A2 8:00:00 AM

A FullTime FullTime\_text FullTime\_int Hour Minute

1 8:00:00

2 8:00:01

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

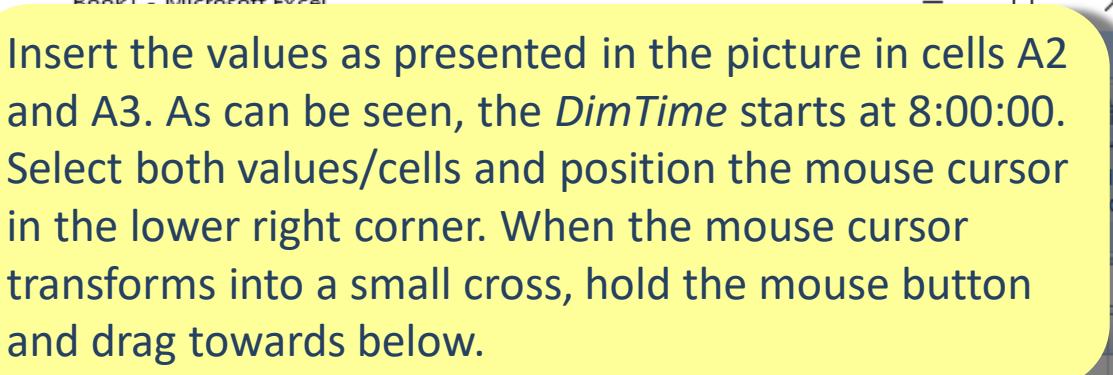
24

25

Sheet1 Sheet2 Sheet3

Average: 8:00:00 Count: 2 Sum: 16:00:01 100%

Insert the values as presented in the picture in cells A2 and A3. As can be seen, the *DimTime* starts at 8:00:00. Select both values/cells and position the mouse cursor in the lower right corner. When the mouse cursor transforms into a small cross, hold the mouse button and drag towards below.



The screenshot shows a Microsoft Excel spreadsheet with a yellow callout bubble containing instructions. The callout bubble has a black border and a white background, pointing from the text above to the bottom-right corner of cell A3. A thick red arrow points downwards from the bottom-right corner of cell A3 towards the bottom edge of the worksheet area. The spreadsheet contains data in rows 1 through 3 and columns A through E. Row 1 has headers: 'FullTime' in A, 'FullTime\_text' in B, 'FullTime\_int' in C, 'Hour' in D, and 'Minute' in E. Row 2 contains the value '8:00:00' in cell A2. Row 3 contains the value '8:00:01' in cell A3. The font used is Calibri, size 11. The text color is black. The background of the spreadsheet is white. The overall interface is consistent with Microsoft Excel's classic ribbon-style design.

Given the initial increment of 1 second, when dragging, Excel automatically increments the values by 1 second. Release the mouse button just when you achieve 23:59:59, i.e., the end time of the *DimTime* being created.

The screenshot shows a Microsoft Excel spreadsheet titled "Book1 - Microsoft Excel". The ribbon menu is visible at the top, showing tabs for Home, Insert, Page Layout, Formulas, Data, and Review. The "Font" and "Alignment" toolbars are also visible above the main workspace. The main workspace contains a single cell, A2, which is highlighted with a red border and contains the text "23:59:59". The cell A2 is part of a larger range that spans from A2 to M2. The row number 57599 is visible on the left side of the screen. The status bar at the bottom of the screen displays the message "Drag outside selection to extend series or fill; drag inside to clear" and "Average: 8:00:00 Count: 2 Sum: 16:00:01".

Book1 - Microsoft Excel

When the mouse button is released, the cells are filled with the values.

A2

8:00:00 AM

	A	B	C	D	E	F	G	H	I	J	K	L	M
57583	23:59:41												
57584	23:59:42												
57585	23:59:43												
57586	23:59:44												
57587	23:59:45												
57588	23:59:46												
57589	23:59:47												
57590	23:59:48												
57591	23:59:49												
57592	23:59:50												
57593	23:59:51												
57594	23:59:52												
57595	23:59:53												
57596	23:59:54												
57597	23:59:55												
57598	23:59:56												
57599	23:59:57												
57600	23:59:58												
57601	23:59:59												
57602													
57603													
57604													
57605													
57606													
57607													

Sheet1 Sheet2 Sheet3

Average: 15:59:59 Count: 57600 Sum: 921592:00:00

Ready 100%

Book1 - Microsoft Excel

Home Insert Page Layout Formulas Data Review View

Paste Clipboa... Font Alignment Number

SUM =TEXT(A2;"hhmmss")

	A	B	C	D	E	F	G
1	FullTime	FullTime_text	FullTime_int	Hour	Minute	Second	Period
2	08:00:00	=TEXT(A2;"hhmmss")					
3	08:00:01						
4	08:00:02						
5	08:00:03						
6	08:00:04						
7	08:00:05						
8	08:00:06						
9	08:00:07						
10	08:00:08						
11	08:00:09						
12	08:00:10						
13	08:00:11						
14	08:00:12						
15	08:00:13						
16	08:00:14						
17	08:00:15						
18	08:00:16						
19	08:00:17						
20	08:00:18						
21	08:00:19						
22	08:00:20						
23	08:00:21						
24	08:00:22						
25	08:00:23						

As we want to transform the *Fulltime* column (hour format) into a *Fulltime\_int* (integer format), let's use an additional column (*FullTime\_text*) to perform this transformation. We'll start by transforming the time into its corresponding text format. For that, in cell B2 input the highlighted formula.

Select the cell and position the mouse cursor in the lower right corner. When the mouse cursor transforms into a small cross, double click to automatically fill the adjacent cells with the same formula. Another (not so good) option is to drag the small cross towards below.

The screenshot shows a Microsoft Excel spreadsheet with the following structure:

	A	B	C	D	E	F	G	H	
1	FullTime	FullTime_text	FullTime_int	Hour	Minute	Second	Period	LunchTime	DinnerTime
2	08:00:00	080000							
3	08:00:01								
4	08:00:02								
5	08:00:03								
6	08:00:04								
7	08:00:05								
8	08:00:06								
9	08:00:07								
10	08:00:08								
11	08:00:09								
12	08:00:10								
13	08:00:11								
14	08:00:12								
15	08:00:13								
16	08:00:14								
17	08:00:15								
18	08:00:16								
19	08:00:17								
20	08:00:18								
21	08:00:19								
22	08:00:20								
23	08:00:21								
24	08:00:22								
25	08:00:23								

The formula `=TEXT(A2;"hhmmss")` is entered in cell B2. A red arrow points from cell B2 to the bottom-right corner of the cell, indicating where the user should click to copy the formula down the column. Another red arrow points downwards from the bottom-right corner of cell B2, indicating the direction of the fill operation.

The values automatically appear in the cells.

Book1 - Microsoft Excel

Home Insert Page Layout Formulas Data Review View

Font Alignment Number Styles Cells Editing

B2 =TEXT(A2;"hhmmss")

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	FullTime	FullTime_text	FullTime_int	Hour	Minute	Second	Period	LunchTime	DinnerTime				
2	08:00:00	080000											
3	08:00:01	080001											
4	08:00:02	080002											
5	08:00:03	080003											
6	08:00:04	080004											
7	08:00:05	080005											
8	08:00:06	080006											
9	08:00:07	080007											
10	08:00:08	080008											
11	08:00:09	080009											
12	08:00:10	080010											
13	08:00:11	080011											
14	08:00:12	080012											
15	08:00:13	080013											
16	08:00:14	080014											
17	08:00:15	080015											
18	08:00:16	080016											
19	08:00:17	080017											
20	08:00:18	080018											
21	08:00:19	080019											
22	08:00:20	080020											
23	08:00:21	080021											
24	08:00:22	080022											
25	08:00:23	080023											

Sheet1 Sheet2 Sheet3

Ready Count: 57600 100%

Book1 - Microsoft Excel

Let's transform the textual time representation into an integer representation. For that, in cell C2, input the highlighted formula.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	FullTime	FullTime_text	FullTime_int	Hour	Minute	Second	Period	LunchTime	DinnerTime				
2	08:00:00	080000	=value(B2)										
3	08:00:01	080001											
4	08:00:02	080002											
5	08:00:03	080003											
6	08:00:04	080004											
7	08:00:05	080005											
8	08:00:06	080006											
9	08:00:07	080007											
10	08:00:08	080008											
11	08:00:09	080009											
12	08:00:10	080010											
13	08:00:11	080011											
14	08:00:12	080012											
15	08:00:13	080013											
16	08:00:14	080014											
17	08:00:15	080015											
18	08:00:16	080016											
19	08:00:17	080017											
20	08:00:18	080018											
21	08:00:19	080019											
22	08:00:20	080020											
23	08:00:21	080021											
24	08:00:22	080022											
25	08:00:23	080023											

SUM      =value(B2)

Enter

Book1 - Microsoft Excel

Home Insert Page Layout Formulas Data Review View

Paste Clipboa...

Font Alignment Number

C2 =VALUE(B2)

A B C D E F G H I J K L M

	FullTime	FullTime_text	FullTime_int	Hour	Minute	Second	Period	LunchTime	DinnerTime					
1	08:00:00	080000	80000											
2	08:00:01	080001												
3	08:00:02	080002												
4	08:00:03	080003												
5	08:00:04	080004												
6	08:00:05	080005												
7	08:00:06	080006												
8	08:00:07	080007												
9	08:00:08	080008												
10	08:00:09	080009												
11	08:00:10	080010												
12	08:00:11	080011												
13	08:00:12	080012												
14	08:00:13	080013												
15	08:00:14	080014												
16	08:00:15	080015												
17	08:00:16	080016												
18	08:00:17	080017												
19	08:00:18	080018												
20	08:00:19	080019												
21	08:00:20	080020												
22	08:00:21	080021												
23	08:00:22	080022												
24	08:00:23	080023												

Sheet1 Sheet2 Sheet3

Select the cell and position the mouse cursor in the lower right corner. When the mouse cursor transforms into a small cross, double click to automatically fill the adjacent cells with the same formula.

The screenshot shows a Microsoft Excel spreadsheet titled 'Book1 - Microsoft Excel'. The ribbon menu is visible at the top. The formula bar shows 'C2' and '=VALUE(B2)'. The spreadsheet has columns labeled A through M. Row 1 contains column headers: FullTime, FullTime\_text, FullTime\_int, Hour, Minute, Second, Period, LunchTime, DinnerTime. Row 2 contains data: 08:00:00, 080000, 80000. A red arrow points to the bottom-right corner of cell C2, indicating where to click to copy the formula down the column. The rest of the rows are empty.

The values automatically appear in the cells.

Book1 - Microsoft Excel

Home Insert Page Layout Formulas Data Review View

Calibri 11 A A

Font Alignment Number

General \$ % , .00 .00

Cells Styles

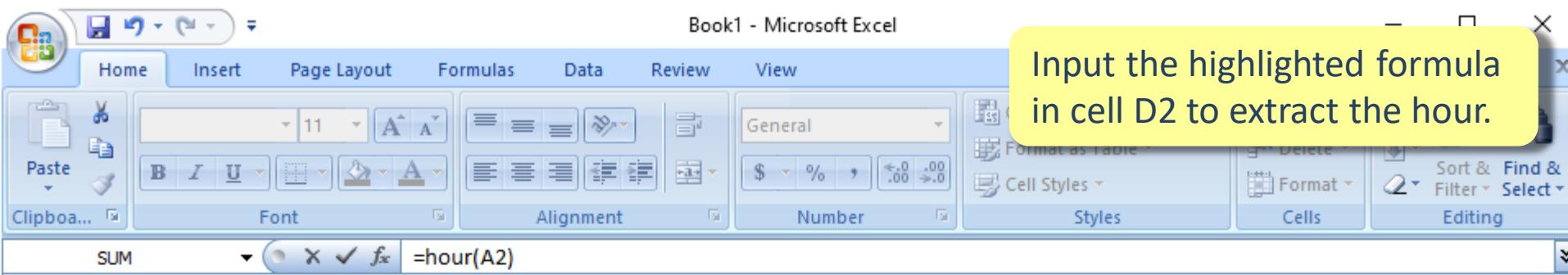
C2 =VALUE(B2)

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	FullTime	FullTime_text	FullTime_int	Hour	Minute	Second	Period	LunchTime	DinnerTime				
2	08:00:00	080000	80000										
3	08:00:01	080001	80001										
4	08:00:02	080002	80002										
5	08:00:03	080003	80003										
6	08:00:04	080004	80004										
7	08:00:05	080005	80005										
8	08:00:06	080006	80006										
9	08:00:07	080007	80007										
10	08:00:08	080008	80008										
11	08:00:09	080009	80009										
12	08:00:10	080010	80010										
13	08:00:11	080011	80011										
14	08:00:12	080012	80012										
15	08:00:13	080013	80013										
16	08:00:14	080014	80014										
17	08:00:15	080015	80015										
18	08:00:16	080016	80016										
19	08:00:17	080017	80017										
20	08:00:18	080018	80018										
21	08:00:19	080019	80019										
22	08:00:20	080020	80020										
23	08:00:21	080021	80021										
24	08:00:22	080022	80022										
25	08:00:23	080023	80023										

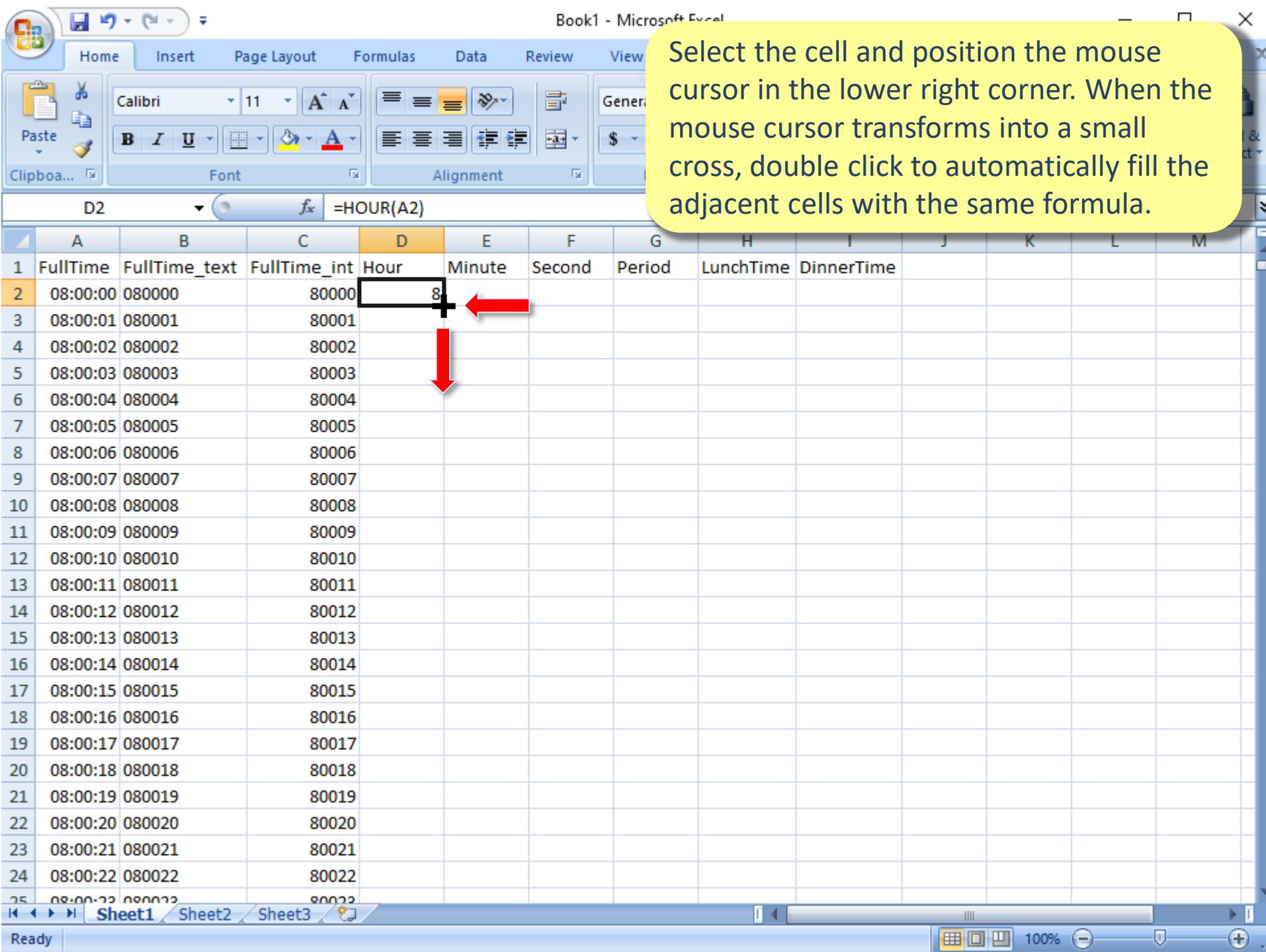
Sheet1 Sheet2 Sheets

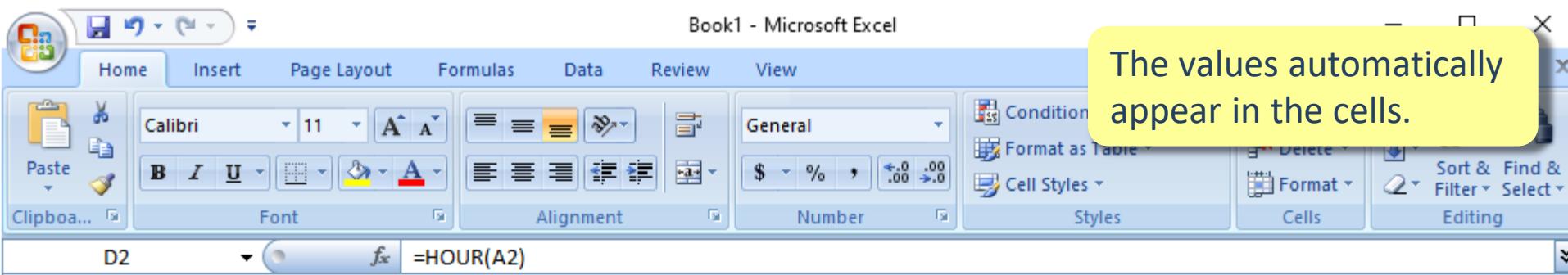
Average: 157979,5 Count: 57600 Sum: 9099619200

100%

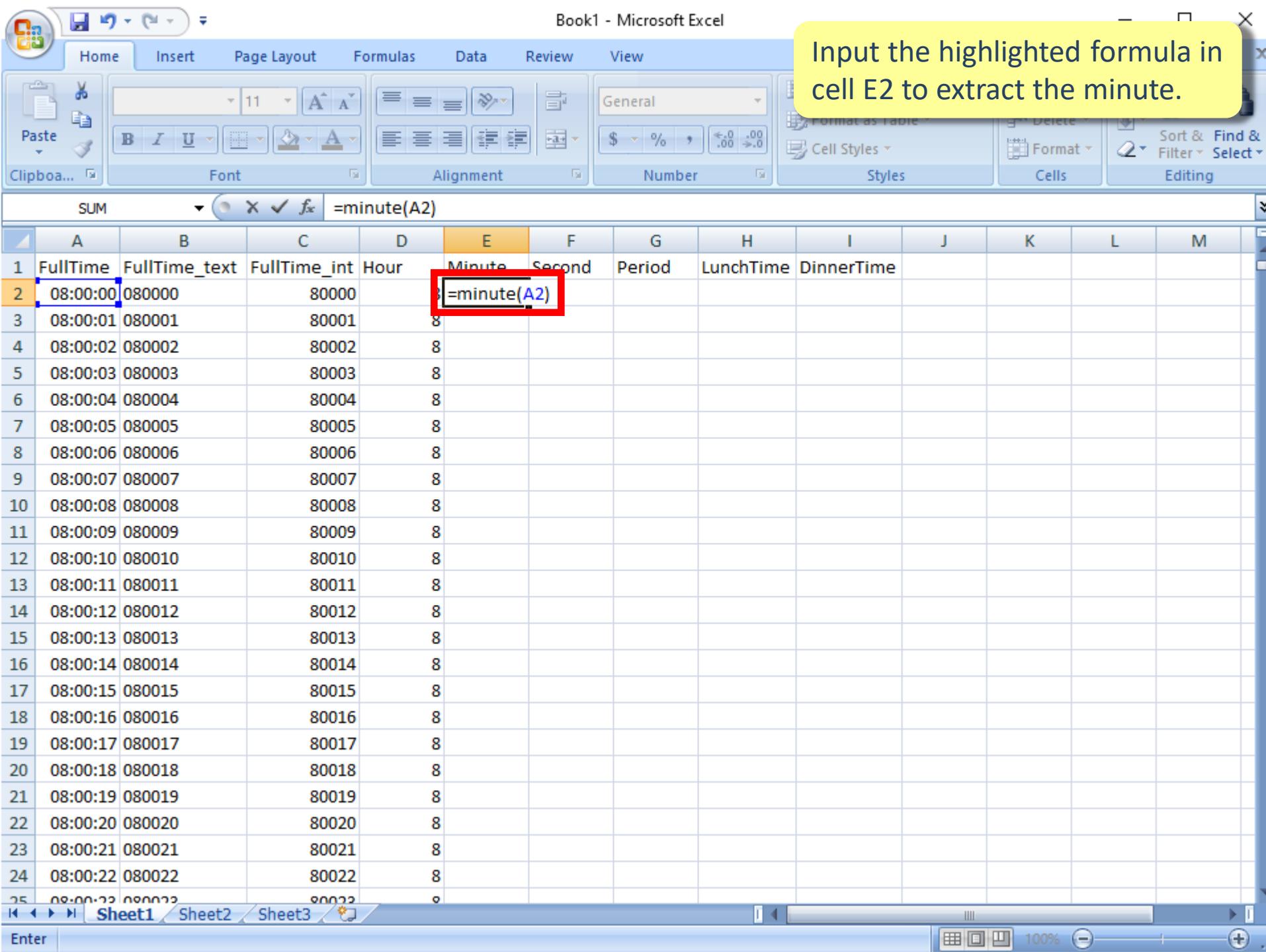


Input the highlighted formula in cell D2 to extract the hour.





The values automatically appear in the cells.



Book1 - Microsoft Excel

Select the cell and position the mouse cursor in the lower right corner. When the mouse cursor transforms into a small cross, double click to automatically fill the adjacent cells with the same formula.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	FullTime	FullTime_text	FullTime_int	Hour	Minute	Second	Period	LunchTime	DinnerTime				
2	08:00:00	080000		80000	8	0							
3	08:00:01	080001		80001	8								
4	08:00:02	080002		80002	8								
5	08:00:03	080003		80003	8								
6	08:00:04	080004		80004	8								
7	08:00:05	080005		80005	8								
8	08:00:06	080006		80006	8								
9	08:00:07	080007		80007	8								
10	08:00:08	080008		80008	8								
11	08:00:09	080009		80009	8								
12	08:00:10	080010		80010	8								
13	08:00:11	080011		80011	8								
14	08:00:12	080012		80012	8								
15	08:00:13	080013		80013	8								
16	08:00:14	080014		80014	8								
17	08:00:15	080015		80015	8								
18	08:00:16	080016		80016	8								
19	08:00:17	080017		80017	8								
20	08:00:18	080018		80018	8								
21	08:00:19	080019		80019	8								
22	08:00:20	080020		80020	8								
23	08:00:21	080021		80021	8								
24	08:00:22	080022		80022	8								
25	08:00:23	080023		80023	8								

Home Insert Page Layout Formulas Data Review View  
Font Alignment General  
Paste Clipboa...  
Font Alignment General  
E2 fx =MINUTE(A2)

Sheet1 Sheet2 Sheet3

Ready 100%

The values automatically appear in the cells.

Book1 - Microsoft Excel

E2      =MINUTE(A2)

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	FullTime	FullTime_text	FullTime_int	Hour	Minute	Second	Period	LunchTime	DinnerTime				
2	08:00:00	080000		80000	8	0							
3	08:00:01	080001		80001	8	0							
4	08:00:02	080002		80002	8	0							
5	08:00:03	080003		80003	8	0							
6	08:00:04	080004		80004	8	0							
7	08:00:05	080005		80005	8	0							
8	08:00:06	080006		80006	8	0							
9	08:00:07	080007		80007	8	0							
10	08:00:08	080008		80008	8	0							
11	08:00:09	080009		80009	8	0							
12	08:00:10	080010		80010	8	0							
13	08:00:11	080011		80011	8	0							
14	08:00:12	080012		80012	8	0							
15	08:00:13	080013		80013	8	0							
16	08:00:14	080014		80014	8	0							
17	08:00:15	080015		80015	8	0							
18	08:00:16	080016		80016	8	0							
19	08:00:17	080017		80017	8	0							
20	08:00:18	080018		80018	8	0							
21	08:00:19	080019		80019	8	0							
22	08:00:20	080020		80020	8	0							
23	08:00:21	080021		80021	8	0							
24	08:00:22	080022		80022	8	0							
25	08:00:23	080023		80023	8	0							

Sheet1 Sheet2 Sheet3

Average: 29,5 Count: 57600 Sum: 1699200 100%

Book1 - Microsoft Excel

Input the highlighted formula in cell F2 to extract the second.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	FullTime	FullTime_text	FullTime_int	Hour	Minute	Second	Period	LunchTime	DinnerTime				
2	08:00:00	080000		80000	8	=second(A2)							
3	08:00:01	080001		80001	8	0							
4	08:00:02	080002		80002	8	0							
5	08:00:03	080003		80003	8	0							
6	08:00:04	080004		80004	8	0							
7	08:00:05	080005		80005	8	0							
8	08:00:06	080006		80006	8	0							
9	08:00:07	080007		80007	8	0							
10	08:00:08	080008		80008	8	0							
11	08:00:09	080009		80009	8	0							
12	08:00:10	080010		80010	8	0							
13	08:00:11	080011		80011	8	0							
14	08:00:12	080012		80012	8	0							
15	08:00:13	080013		80013	8	0							
16	08:00:14	080014		80014	8	0							
17	08:00:15	080015		80015	8	0							
18	08:00:16	080016		80016	8	0							
19	08:00:17	080017		80017	8	0							
20	08:00:18	080018		80018	8	0							
21	08:00:19	080019		80019	8	0							
22	08:00:20	080020		80020	8	0							
23	08:00:21	080021		80021	8	0							
24	08:00:22	080022		80022	8	0							
25	08:00:23	080023		80023	8	0							

Select the cell and position the mouse cursor in the lower right corner. When the mouse cursor transforms into a small cross, double click to automatically fill the adjacent cells with the same formula.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	FullTime	FullTime_text	FullTime_int	Hour	Minute	Second	Period	LunchTime	DinnerTime				
2	08:00:00	080000		8	0	0							
3	08:00:01	080001		8	0	0							
4	08:00:02	080002		8	0	0							
5	08:00:03	080003		8	0	0							
6	08:00:04	080004		8	0	0							
7	08:00:05	080005		8	0	0							
8	08:00:06	080006		8	0	0							
9	08:00:07	080007		8	0	0							
10	08:00:08	080008		8	0	0							
11	08:00:09	080009		8	0	0							
12	08:00:10	080010		8	0	0							
13	08:00:11	080011		8	0	0							
14	08:00:12	080012		8	0	0							
15	08:00:13	080013		8	0	0							
16	08:00:14	080014		8	0	0							
17	08:00:15	080015		8	0	0							
18	08:00:16	080016		8	0	0							
19	08:00:17	080017		8	0	0							
20	08:00:18	080018		8	0	0							
21	08:00:19	080019		8	0	0							
22	08:00:20	080020		8	0	0							
23	08:00:21	080021		8	0	0							
24	08:00:22	080022		8	0	0							
25	08:00:23	080023		8	0	0							

The values automatically appear in the cells.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	FullTime	FullTime_text	FullTime_int	Hour	Minute	Second	Period	LunchTime	DinnerTime				
2	08:00:00	080000		8	0	0							
3	08:00:01	080001		8	0	1							
4	08:00:02	080002		8	0	2							
5	08:00:03	080003		8	0	3							
6	08:00:04	080004		8	0	4							
7	08:00:05	080005		8	0	5							
8	08:00:06	080006		8	0	6							
9	08:00:07	080007		8	0	7							
10	08:00:08	080008		8	0	8							
11	08:00:09	080009		8	0	9							
12	08:00:10	080010		8	0	10							
13	08:00:11	080011		8	0	11							
14	08:00:12	080012		8	0	12							
15	08:00:13	080013		8	0	13							
16	08:00:14	080014		8	0	14							
17	08:00:15	080015		8	0	15							
18	08:00:16	080016		8	0	16							
19	08:00:17	080017		8	0	17							
20	08:00:18	080018		8	0	18							
21	08:00:19	080019		8	0	19							
22	08:00:20	080020		8	0	20							
23	08:00:21	080021		8	0	21							
24	08:00:22	080022		8	0	22							
25	08:00:23	080023		8	0	23							

Book1 - Microsoft Excel

Input the highlighted formula in cell G2 to obtain the day period (morning, afternoon or evening).

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	FullTime	FullTime_text	FullTime_int	Hour	Minute	Second	Period	LunchTime	DinnerTime				
2	08:00:00	080000	80000	8	0	0	=if(D2<12;"Morning";IF(D2<20;"Afternoon";"Evening"))						
3	08:00:01	080001	80001	8	0	0	1						
4	08:00:02	080002	80002	8	0	0	2						
5	08:00:03	080003	80003	8	0	0	3						
6	08:00:04	080004	80004	8	0	0	4						
7	08:00:05	080005	80005	8	0	0	5						
8	08:00:06	080006	80006	8	0	0	6						
9	08:00:07	080007	80007	8	0	0	7						
10	08:00:08	080008	80008	8	0	0	8						
11	08:00:09	080009	80009	8	0	0	9						
12	08:00:10	080010	80010	8	0	0	10						
13	08:00:11	080011	80011	8	0	0	11						
14	08:00:12	080012	80012	8	0	0	12						
15	08:00:13	080013	80013	8	0	0	13						
16	08:00:14	080014	80014	8	0	0	14						
17	08:00:15	080015	80015	8	0	0	15						
18	08:00:16	080016	80016	8	0	0	16						
19	08:00:17	080017	80017	8	0	0	17						
20	08:00:18	080018	80018	8	0	0	18						
21	08:00:19	080019	80019	8	0	0	19						
22	08:00:20	080020	80020	8	0	0	20						
23	08:00:21	080021	80021	8	0	0	21						
24	08:00:22	080022	80022	8	0	0	22						
25	08:00:23	080023	80023	8	0	0	23						

Select the cell and position the mouse cursor in the lower right corner. When the mouse cursor transforms into a small cross, double click to automatically fill the adjacent cells with the same formula.

The screenshot shows a Microsoft Excel spreadsheet titled "Book1 - Microsoft Excel". The ribbon menu is visible at the top, and the formula bar displays the formula =IF(D2<12;"Morning";IF(D2<20;"Afternoon","Night")) in cell G2. The data starts in row 1 with columns A through M. Column A contains "FullTime" and "FullTime\_text" with corresponding values. Column B contains "FullTime\_int" with values ranging from 80000 to 90022. Column C contains "Hour" with all values set to 8. Columns D, E, and F contain "Minute", "Second", and "Period" respectively. Column G contains the formula =IF(D2<12;"Morning";IF(D2<20;"Afternoon","Night")) and is highlighted with a black border. Red arrows point from the text above to the bottom-right corner of this cell, indicating where to click to copy the formula down the column. The rest of the table shows a sequence of times from 08:00:00 to 08:00:22.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	FullTime	FullTime_text	FullTime_int	Hour	Minute	Second	Period	LunchTime	DinnerTime				
2	08:00:00	080000		80000	8	0	0 Morning						
3	08:00:01	080001		80001	8	0	1						
4	08:00:02	080002		80002	8	0	2						
5	08:00:03	080003		80003	8	0	3						
6	08:00:04	080004		80004	8	0	4						
7	08:00:05	080005		80005	8	0	5						
8	08:00:06	080006		80006	8	0	6						
9	08:00:07	080007		80007	8	0	7						
10	08:00:08	080008		80008	8	0	8						
11	08:00:09	080009		80009	8	0	9						
12	08:00:10	080010		80010	8	0	10						
13	08:00:11	080011		80011	8	0	11						
14	08:00:12	080012		80012	8	0	12						
15	08:00:13	080013		80013	8	0	13						
16	08:00:14	080014		80014	8	0	14						
17	08:00:15	080015		80015	8	0	15						
18	08:00:16	080016		80016	8	0	16						
19	08:00:17	080017		80017	8	0	17						
20	08:00:18	080018		80018	8	0	18						
21	08:00:19	080019		80019	8	0	19						
22	08:00:20	080020		80020	8	0	20						
23	08:00:21	080021		80021	8	0	21						
24	08:00:22	080022		80022	8	0	22						
25	08:00:23	080023		80023	8	0	23						

The values automatically appear in the cells.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	FullTime	FullTime_text	FullTime_int	Hour	Minute	Second	Period	LunchTime	DinnerTime				
2	08:00:00	080000		8	0	0	0 Morning						
3	08:00:01	080001		8	0	0	1 Morning						
4	08:00:02	080002		8	0	0	2 Morning						
5	08:00:03	080003		8	0	0	3 Morning						
6	08:00:04	080004		8	0	0	4 Morning						
7	08:00:05	080005		8	0	0	5 Morning						
8	08:00:06	080006		8	0	0	6 Morning						
9	08:00:07	080007		8	0	0	7 Morning						
10	08:00:08	080008		8	0	0	8 Morning						
11	08:00:09	080009		8	0	0	9 Morning						
12	08:00:10	080010		8	0	0	10 Morning						
13	08:00:11	080011		8	0	0	11 Morning						
14	08:00:12	080012		8	0	0	12 Morning						
15	08:00:13	080013		8	0	0	13 Morning						
16	08:00:14	080014		8	0	0	14 Morning						
17	08:00:15	080015		8	0	0	15 Morning						
18	08:00:16	080016		8	0	0	16 Morning						
19	08:00:17	080017		8	0	0	17 Morning						
20	08:00:18	080018		8	0	0	18 Morning						
21	08:00:19	080019		8	0	0	19 Morning						
22	08:00:20	080020		8	0	0	20 Morning						
23	08:00:21	080021		8	0	0	21 Morning						
24	08:00:22	080022		8	0	0	22 Morning						
25	08:00:23	080023		8	0	0	23 Morning						

Book1 - Microsoft Excel

Input the highlighted formula in cell H2 to flag (no/yes) the lunch time period.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	FullTime	FullTime_text	FullTime_int	Hour	Minute	Second	Period	LunchTime	DinnerTime				
2	08:00:00	080000	80000	8	0	0	0 Morning	=IF(OR(D2=12;D2=13); "Yes"; "No")					
3	08:00:01	080001	80001	8	0	0	1 Morning						
4	08:00:02	080002	80002	8	0	0	2 Morning						
5	08:00:03	080003	80003	8	0	0	3 Morning						
6	08:00:04	080004	80004	8	0	0	4 Morning						
7	08:00:05	080005	80005	8	0	0	5 Morning						
8	08:00:06	080006	80006	8	0	0	6 Morning						
9	08:00:07	080007	80007	8	0	0	7 Morning						
10	08:00:08	080008	80008	8	0	0	8 Morning						
11	08:00:09	080009	80009	8	0	0	9 Morning						
12	08:00:10	080010	80010	8	0	0	10 Morning						
13	08:00:11	080011	80011	8	0	0	11 Morning						
14	08:00:12	080012	80012	8	0	0	12 Morning						
15	08:00:13	080013	80013	8	0	0	13 Morning						
16	08:00:14	080014	80014	8	0	0	14 Morning						
17	08:00:15	080015	80015	8	0	0	15 Morning						
18	08:00:16	080016	80016	8	0	0	16 Morning						
19	08:00:17	080017	80017	8	0	0	17 Morning						
20	08:00:18	080018	80018	8	0	0	18 Morning						
21	08:00:19	080019	80019	8	0	0	19 Morning						
22	08:00:20	080020	80020	8	0	0	20 Morning						
23	08:00:21	080021	80021	8	0	0	21 Morning						
24	08:00:22	080022	80022	8	0	0	22 Morning						
25	08:00:23	080023	80023	8	0	0	23 Morning						

Select the cell and position the mouse cursor in the lower right corner. When the mouse cursor transforms into a small cross, double click to automatically fill the adjacent cells with the same formula.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	FullTime	FullTime_text	FullTime_int	Hour	Minute	Second	Period	LunchTime	DinnerTime				
2	08:00:00	080000		80000	8	0	0	Morning	No				
3	08:00:01	080001		80001	8	0	1	Morning					
4	08:00:02	080002		80002	8	0	2	Morning					
5	08:00:03	080003		80003	8	0	3	Morning					
6	08:00:04	080004		80004	8	0	4	Morning					
7	08:00:05	080005		80005	8	0	5	Morning					
8	08:00:06	080006		80006	8	0	6	Morning					
9	08:00:07	080007		80007	8	0	7	Morning					
10	08:00:08	080008		80008	8	0	8	Morning					
11	08:00:09	080009		80009	8	0	9	Morning					
12	08:00:10	080010		80010	8	0	10	Morning					
13	08:00:11	080011		80011	8	0	11	Morning					
14	08:00:12	080012		80012	8	0	12	Morning					
15	08:00:13	080013		80013	8	0	13	Morning					
16	08:00:14	080014		80014	8	0	14	Morning					
17	08:00:15	080015		80015	8	0	15	Morning					
18	08:00:16	080016		80016	8	0	16	Morning					
19	08:00:17	080017		80017	8	0	17	Morning					
20	08:00:18	080018		80018	8	0	18	Morning					
21	08:00:19	080019		80019	8	0	19	Morning					
22	08:00:20	080020		80020	8	0	20	Morning					
23	08:00:21	080021		80021	8	0	21	Morning					
24	08:00:22	080022		80022	8	0	22	Morning					
25	08:00:23	080023		80023	8	0	23	Morning					

The values automatically appear in the cells.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	FullTime	FullTime_text	FullTime_int	Hour	Minute	Second	Period	LunchTime	DinnerTime				
2	08:00:00	080000		80000	8	0	0 Morning	No					
3	08:00:01	080001		80001	8	0	1 Morning	No					
4	08:00:02	080002		80002	8	0	2 Morning	No					
5	08:00:03	080003		80003	8	0	3 Morning	No					
6	08:00:04	080004		80004	8	0	4 Morning	No					
7	08:00:05	080005		80005	8	0	5 Morning	No					
8	08:00:06	080006		80006	8	0	6 Morning	No					
9	08:00:07	080007		80007	8	0	7 Morning	No					
10	08:00:08	080008		80008	8	0	8 Morning	No					
11	08:00:09	080009		80009	8	0	9 Morning	No					
12	08:00:10	080010		80010	8	0	10 Morning	No					
13	08:00:11	080011		80011	8	0	11 Morning	No					
14	08:00:12	080012		80012	8	0	12 Morning	No					
15	08:00:13	080013		80013	8	0	13 Morning	No					
16	08:00:14	080014		80014	8	0	14 Morning	No					
17	08:00:15	080015		80015	8	0	15 Morning	No					
18	08:00:16	080016		80016	8	0	16 Morning	No					
19	08:00:17	080017		80017	8	0	17 Morning	No					
20	08:00:18	080018		80018	8	0	18 Morning	No					
21	08:00:19	080019		80019	8	0	19 Morning	No					
22	08:00:20	080020		80020	8	0	20 Morning	No					
23	08:00:21	080021		80021	8	0	21 Morning	No					
24	08:00:22	080022		80022	8	0	22 Morning	No					
25	08:00:23	080023		80023	8	0	23 Morning	No					

The lunch period is considered to begin at 12:00:00...

Excel ribbon tabs: Home, Insert, Page Layout, Formulas, Data, Review, View.

Font, Alignment, Number, Styles, Cells, Editing buttons.

Formula Bar: H7 =IF(OR(D7=12;D7=13); "Yes"; "No")

	A	B	C	D	E	F	G	H	I	J	K	L	M
14395	11:59:53	115953		115953	11	59	53	Morning	No				
14396	11:59:54	115954		115954	11	59	54	Morning	No				
14397	11:59:55	115955		115955	11	59	55	Morning	No				
14398	11:59:56	115956		115956	11	59	56	Morning	No				
14399	11:59:57	115957		115957	11	59	57	Morning	No				
14400	11:59:58	115958		115958	11	59	58	Morning	No				
14401	11:59:59	115959		115959	11	59	59	Morning	No				
14402	12:00:00	120000		120000	12	0	0	Afternoor	Yes				
14403	12:00:01	120001		120001	12	0	1	Afternoor	Yes				
14404	12:00:02	120002		120002	12	0	2	Afternoor	Yes				
14405	12:00:03	120003		120003	12	0	3	Afternoor	Yes				
14406	12:00:04	120004		120004	12	0	4	Afternoor	Yes				
14407	12:00:05	120005		120005	12	0	5	Afternoor	Yes				
14408	12:00:06	120006		120006	12	0	6	Afternoor	Yes				
14409	12:00:07	120007		120007	12	0	7	Afternoor	Yes				
14410	12:00:08	120008		120008	12	0	8	Afternoor	Yes				
14411	12:00:09	120009		120009	12	0	9	Afternoor	Yes				
14412	12:00:10	120010		120010	12	0	10	Afternoor	Yes				
14413	12:00:11	120011		120011	12	0	11	Afternoor	Yes				
14414	12:00:12	120012		120012	12	0	12	Afternoor	Yes				
14415	12:00:13	120013		120013	12	0	13	Afternoor	Yes				
14416	12:00:14	120014		120014	12	0	14	Afternoor	Yes				
14417	12:00:15	120015		120015	12	0	15	Afternoor	Yes				
14418	12:00:16	120016		120016	12	0	16	Afternoor	Yes				
14419	12:00:17	120017		120017	12	0	17	Afternoor	Yes				

...and ending at 13:59:59.

	A	B	C	D	E	F	G	H	I	J	K	L	M
21588	13:59:46	135946		135946	13	59	46	Afternoor Yes					
21589	13:59:47	135947		135947	13	59	47	Afternoor Yes					
21590	13:59:48	135948		135948	13	59	48	Afternoor Yes					
21591	13:59:49	135949		135949	13	59	49	Afternoor Yes					
21592	13:59:50	135950		135950	13	59	50	Afternoor Yes					
21593	13:59:51	135951		135951	13	59	51	Afternoor Yes					
21594	13:59:52	135952		135952	13	59	52	Afternoor Yes					
21595	13:59:53	135953		135953	13	59	53	Afternoor Yes					
21596	13:59:54	135954		135954	13	59	54	Afternoor Yes					
21597	13:59:55	135955		135955	13	59	55	Afternoor Yes					
21598	13:59:56	135956		135956	13	59	56	Afternoor Yes					
21599	13:59:57	135957		135957	13	59	57	Afternoor Yes					
21600	13:59:58	135958		135958	13	59	58	Afternoor Yes					
21601	13:59:59	135959		135959	13	59	59	Afternoor Yes					
21602	14:00:00	140000		140000	14	0	0	Afternoor No					
21603	14:00:01	140001		140001	14	0	1	Afternoor No					
21604	14:00:02	140002		140002	14	0	2	Afternoor No					
21605	14:00:03	140003		140003	14	0	3	Afternoor No					
21606	14:00:04	140004		140004	14	0	4	Afternoor No					
21607	14:00:05	140005		140005	14	0	5	Afternoor No					
21608	14:00:06	140006		140006	14	0	6	Afternoor No					
21609	14:00:07	140007		140007	14	0	7	Afternoor No					
21610	14:00:08	140008		140008	14	0	8	Afternoor No					
21611	14:00:09	140009		140009	14	0	9	Afternoor No					
21612	14:00:10	140010		140010	14	0	10	Afternoor No					

Book1 - Microsoft Excel

Input the highlighted formula in cell I2 to flag (no/yes) the dinner time period.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	FullTime	FullTime_text	FullTime_int	Hour	Minute	Second	Period	LunchTime	DinnerTime				
2	08:00:00	080000	80000	8	0	0	0 Morning	No	=IF(AND(C2>=193000;C2<220000);"Yes";"No")				
3	08:00:01	080001	80001	8	0	0	1 Morning	No					
4	08:00:02	080002	80002	8	0	0	2 Morning	No					
5	08:00:03	080003	80003	8	0	0	3 Morning	No					
6	08:00:04	080004	80004	8	0	0	4 Morning	No					
7	08:00:05	080005	80005	8	0	0	5 Morning	No					
8	08:00:06	080006	80006	8	0	0	6 Morning	No					
9	08:00:07	080007	80007	8	0	0	7 Morning	No					
10	08:00:08	080008	80008	8	0	0	8 Morning	No					
11	08:00:09	080009	80009	8	0	0	9 Morning	No					
12	08:00:10	080010	80010	8	0	0	10 Morning	No					
13	08:00:11	080011	80011	8	0	0	11 Morning	No					
14	08:00:12	080012	80012	8	0	0	12 Morning	No					
15	08:00:13	080013	80013	8	0	0	13 Morning	No					
16	08:00:14	080014	80014	8	0	0	14 Morning	No					
17	08:00:15	080015	80015	8	0	0	15 Morning	No					
18	08:00:16	080016	80016	8	0	0	16 Morning	No					
19	08:00:17	080017	80017	8	0	0	17 Morning	No					
20	08:00:18	080018	80018	8	0	0	18 Morning	No					
21	08:00:19	080019	80019	8	0	0	19 Morning	No					
22	08:00:20	080020	80020	8	0	0	20 Morning	No					
23	08:00:21	080021	80021	8	0	0	21 Morning	No					
24	08:00:22	080022	80022	8	0	0	22 Morning	No					
25	08:00:23	080023	80023	8	0	0	23 Morning	No					

Enter

Select the cell and position the mouse cursor in the lower right corner. When the mouse cursor transforms into a small cross, double click to automatically fill the adjacent cells with the same formula.

I2

=IF(AND(C2>=193000;C2<220000); "Yes"; "No")

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	FullTime	FullTime_text	FullTime_int	Hour	Minute	Second	Period	LunchTime	DinnerTime				
2	08:00:00	080000		80000	8	0	0 Morning	No	No				
3	08:00:01	080001		80001	8	0	1 Morning	No					
4	08:00:02	080002		80002	8	0	2 Morning	No					
5	08:00:03	080003		80003	8	0	3 Morning	No					
6	08:00:04	080004		80004	8	0	4 Morning	No					
7	08:00:05	080005		80005	8	0	5 Morning	No					
8	08:00:06	080006		80006	8	0	6 Morning	No					
9	08:00:07	080007		80007	8	0	7 Morning	No					
10	08:00:08	080008		80008	8	0	8 Morning	No					
11	08:00:09	080009		80009	8	0	9 Morning	No					
12	08:00:10	080010		80010	8	0	10 Morning	No					
13	08:00:11	080011		80011	8	0	11 Morning	No					
14	08:00:12	080012		80012	8	0	12 Morning	No					
15	08:00:13	080013		80013	8	0	13 Morning	No					
16	08:00:14	080014		80014	8	0	14 Morning	No					
17	08:00:15	080015		80015	8	0	15 Morning	No					
18	08:00:16	080016		80016	8	0	16 Morning	No					
19	08:00:17	080017		80017	8	0	17 Morning	No					
20	08:00:18	080018		80018	8	0	18 Morning	No					
21	08:00:19	080019		80019	8	0	19 Morning	No					
22	08:00:20	080020		80020	8	0	20 Morning	No					
23	08:00:21	080021		80021	8	0	21 Morning	No					
24	08:00:22	080022		80022	8	0	22 Morning	No					
25	08:00:23	080023		80023	8	0	23 Morning	No					

The values automatically appear in the cells.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	FullTime	FullTime_text	FullTime_int	Hour	Minute	Second	Period	LunchTime	DinnerTime				
2	08:00:00	080000	80000	8	0	0	0 Morning	No	No				
3	08:00:01	080001	80001	8	0	0	1 Morning	No	No				
4	08:00:02	080002	80002	8	0	0	2 Morning	No	No				
5	08:00:03	080003	80003	8	0	0	3 Morning	No	No				
6	08:00:04	080004	80004	8	0	0	4 Morning	No	No				
7	08:00:05	080005	80005	8	0	0	5 Morning	No	No				
8	08:00:06	080006	80006	8	0	0	6 Morning	No	No				
9	08:00:07	080007	80007	8	0	0	7 Morning	No	No				
10	08:00:08	080008	80008	8	0	0	8 Morning	No	No				
11	08:00:09	080009	80009	8	0	0	9 Morning	No	No				
12	08:00:10	080010	80010	8	0	0	10 Morning	No	No				
13	08:00:11	080011	80011	8	0	0	11 Morning	No	No				
14	08:00:12	080012	80012	8	0	0	12 Morning	No	No				
15	08:00:13	080013	80013	8	0	0	13 Morning	No	No				
16	08:00:14	080014	80014	8	0	0	14 Morning	No	No				
17	08:00:15	080015	80015	8	0	0	15 Morning	No	No				
18	08:00:16	080016	80016	8	0	0	16 Morning	No	No				
19	08:00:17	080017	80017	8	0	0	17 Morning	No	No				
20	08:00:18	080018	80018	8	0	0	18 Morning	No	No				
21	08:00:19	080019	80019	8	0	0	19 Morning	No	No				
22	08:00:20	080020	80020	8	0	0	20 Morning	No	No				
23	08:00:21	080021	80021	8	0	0	21 Morning	No	No				
24	08:00:22	080022	80022	8	0	0	22 Morning	No	No				
25	08:00:23	080023	80023	8	0	0	23 Morning	No	No				

The dinner period is considered to begin at 19:30:00...

	A	B	C	D	E	F	G	H	I	J	K	L	M
41390	19:29:48	192948		192948	19	29	48	Avond	No				
41391	19:29:49	192949		192949	19	29	49	Avond	No				
41392	19:29:50	192950		192950	19	29	50	Avond	No				
41393	19:29:51	192951		192951	19	29	51	Avond	No				
41394	19:29:52	192952		192952	19	29	52	Avond	No				
41395	19:29:53	192953		192953	19	29	53	Avond	No				
41396	19:29:54	192954		192954	19	29	54	Avond	No				
41397	19:29:55	192955		192955	19	29	55	Avond	No				
41398	19:29:56	192956		192956	19	29	56	Avond	No				
41399	19:29:57	192957		192957	19	29	57	Avond	No				
41400	19:29:58	192958		192958	19	29	58	Avond	No				
41401	19:29:59	192959		192959	19	29	59	Avond	No				
41402	19:30:00	193000		193000	19	30	0	Avond	No				
41403	19:30:01	193001		193001	19	30	1	Avond	No				
41404	19:30:02	193002		193002	19	30	2	Avond	No				
41405	19:30:03	193003		193003	19	30	3	Avond	No				
41406	19:30:04	193004		193004	19	30	4	Avond	No				
41407	19:30:05	193005		193005	19	30	5	Avond	No				
41408	19:30:06	193006		193006	19	30	6	Avond	No				
41409	19:30:07	193007		193007	19	30	7	Avond	No				
41410	19:30:08	193008		193008	19	30	8	Avond	No				
41411	19:30:09	193009		193009	19	30	9	Avond	No				
41412	19:30:10	193010		193010	19	30	10	Avond	No				
41413	19:30:11	193011		193011	19	30	11	Avond	No				
41414	19:30:12	193012		193012	19	30	12	Avond	No				

Book1 - Microsoft Excel											
...and ending at 21:59:59.											
I50394											
A	B	C	D	E	F	G	H	I	J	K	L
50389	21:59:47	215947	215947	21	59	47	Evening	No	Yes		
50390	21:59:48	215948	215948	21	59	48	Evening	No	Yes		
50391	21:59:49	215949	215949	21	59	49	Evening	No	Yes		
50392	21:59:50	215950	215950	21	59	50	Evening	No	Yes		
50393	21:59:51	215951	215951	21	59	51	Evening	No	Yes		
50394	21:59:52	215952	215952	21	59	52	Evening	No	Yes		
50395	21:59:53	215953	215953	21	59	53	Evening	No	Yes		
50396	21:59:54	215954	215954	21	59	54	Evening	No	Yes		
50397	21:59:55	215955	215955	21	59	55	Evening	No	Yes		
50398	21:59:56	215956	215956	21	59	56	Evening	No	Yes		
50399	21:59:57	215957	215957	21	59	57	Evening	No	Yes		
50400	21:59:58	215958	215958	21	59	58	Evening	No	Yes		
50401	21:59:59	215959	215959	21	59	59	Evening	No	Yes		
50402	22:00:00	220000	220000	22	0	0	Evening	No	No		
50403	22:00:01	220001	220001	22	0	1	Evening	No	No		
50404	22:00:02	220002	220002	22	0	2	Evening	No	No		
50405	22:00:03	220003	220003	22	0	3	Evening	No	No		
50406	22:00:04	220004	220004	22	0	4	Evening	No	No		
50407	22:00:05	220005	220005	22	0	5	Evening	No	No		
50408	22:00:06	220006	220006	22	0	6	Evening	No	No		
50409	22:00:07	220007	220007	22	0	7	Evening	No	No		
50410	22:00:08	220008	220008	22	0	8	Evening	No	No		
50411	22:00:09	220009	220009	22	0	9	Evening	No	No		
50412	22:00:10	220010	220010	22	0	10	Evening	No	No		
50413	22:00:11	220011	220011	22	0	11	Evening	No	No		

Book1 - Microsoft Excel

Save the DimTime Excel file.

The screenshot shows a Microsoft Excel spreadsheet titled "Book1 - Microsoft Excel". The ribbon menu is visible at the top, with the "Home" tab selected. A red arrow points to the "Save (Ctrl+S)" button in the top-left corner of the ribbon. A yellow callout bubble on the right side of the screen contains the text "Save the DimTime Excel file." The spreadsheet contains data from rows 50389 to 50412. Column A lists times from 21:59:47 to 22:00:11. Column B lists IDs from 215947 to 220011. Column C contains the formula =IF(AND(C50394>=193000;C50394<220000);"Yes";"No"). Columns D through M show numerical values and text entries such as "Evening" and "No". Row 50394 is highlighted with an orange background, and the cell I50394 is currently selected and has a black border around it. The bottom navigation bar shows tabs for "Sheet1", "Sheet2", and "Sheet3".

	A	B	C	D	E	F	G	H	I	J	K	L	M
50389	21:59:47	215947		215947	21	59	47	Evening	No	Yes			
50390	21:59:48	215948		215948	21	59	48	Evening	No	Yes			
50391	21:59:49	215949		215949	21	59	49	Evening	No	Yes			
50392	21:59:50	215950		215950	21	59	50	Evening	No	Yes			
50393	21:59:51	215951		215951	21	59	51	Evening	No	Yes			
50394	21:59:52	215952		215952	21	59	52	Evening	No	Yes			
50395	21:59:53	215953		215953	21	59	53	Evening	No	Yes			
50396	21:59:54	215954		215954	21	59	54	Evening	No	Yes			
50397	21:59:55	215955		215955	21	59	55	Evening	No	Yes			
50398	21:59:56	215956		215956	21	59	56	Evening	No	Yes			
50399	21:59:57	215957		215957	21	59	57	Evening	No	Yes			
50400	21:59:58	215958		215958	21	59	58	Evening	No	Yes			
50401	21:59:59	215959		215959	21	59	59	Evening	No	Yes			
50402	22:00:00	220000		220000	22	0	0	Evening	No	No			
50403	22:00:01	220001		220001	22	0	1	Evening	No	No			
50404	22:00:02	220002		220002	22	0	2	Evening	No	No			
50405	22:00:03	220003		220003	22	0	3	Evening	No	No			
50406	22:00:04	220004		220004	22	0	4	Evening	No	No			
50407	22:00:05	220005		220005	22	0	5	Evening	No	No			
50408	22:00:06	220006		220006	22	0	6	Evening	No	No			
50409	22:00:07	220007		220007	22	0	7	Evening	No	No			
50410	22:00:08	220008		220008	22	0	8	Evening	No	No			
50411	22:00:09	220009		220009	22	0	9	Evening	No	No			
50412	22:00:10	220010		220010	22	0	10	Evening	No	No			
50413	22:00:11	220011		220011	22	0	11	Evening	No	No			

Select the *Files* folder stored inside the *Data Mart* project folder as the proper location for the file. Name it *DimTime* and select the CSV file format.

File name: DimTime

Save as type: CSV (Comma delimited)

Authors: Paulo Oliveira

Tags: Add a tag

Title: Add a title

Tools Save Cancel

50411	22:00:09	220009		22	0	9	Evening	No	No
50412	22:00:10	220010		22	0	10	Evening	No	No
50413	22:00:11	220011		22	0	11	Evening	No	No

Book1 - Microsoft Excel

Confirm the save of the active sheet.

Font

Alignment

Number

General

Conditional Formatting

Format as Table

Delete

Sort & Filter

Find & Select

Cells

Editing

I50394 =IF(AND(C50394>=193000;C50394<220000); "Yes"; "No")

	A	B	C	D	E	F	G	H	I	J	K	L	M
50389	21:59:47	215947		215947	21	59	47	Evening	No	Yes			
50390	21:59:48	215948		215948	21	59	48	Evening	No	Yes			
50391	21:59:49	215949		215949	21	59	49	Evening	No	Yes			
50392	21:59:50	215950		215950	21	59	50	Evening	No	Yes			
50393	21:59:51	215951		215951	21	59	51	Evening	No	Yes			
50394	21:59:52	215	Microsoft Office Excel										
50395	21:59:53	215											
50396	21:59:54	215											
50397	21:59:55	215											
50398	21:59:56	215											
50399	21:59:57	215											
50400	21:59:58	215											
50401	21:59:59	215959		215959	21	59	59	Evening	No	Yes			
50402	22:00:00	220000		220000	22	0	0	Evening	No	No			
50403	22:00:01	220001		220001	22	0	1	Evening	No	No			
50404	22:00:02	220002		220002	22	0	2	Evening	No	No			
50405	22:00:03	220003		220003	22	0	3	Evening	No	No			
50406	22:00:04	220004		220004	22	0	4	Evening	No	No			
50407	22:00:05	220005		220005	22	0	5	Evening	No	No			
50408	22:00:06	220006		220006	22	0	6	Evening	No	No			
50409	22:00:07	220007		220007	22	0	7	Evening	No	No			
50410	22:00:08	220008		220008	22	0	8	Evening	No	No			
50411	22:00:09	220009		220009	22	0	9	Evening	No	No			
50412	22:00:10	220010		220010	22	0	10	Evening	No	No			
50413	22:00:11	220011		220011	22	0	11	Evening	No	No			

OK Cancel

The selected file type does not support workbooks that contain multiple sheets.  
• To save only the active sheet, click OK.  
• To save all sheets, save them individually using a different file name for each, or choose a file type that supports multiple sheets.

Sheet1 Sheet2 Sheet3

Ready 100%

Book1 - Microsoft Excel

Confirm once again the save in CSV file format.

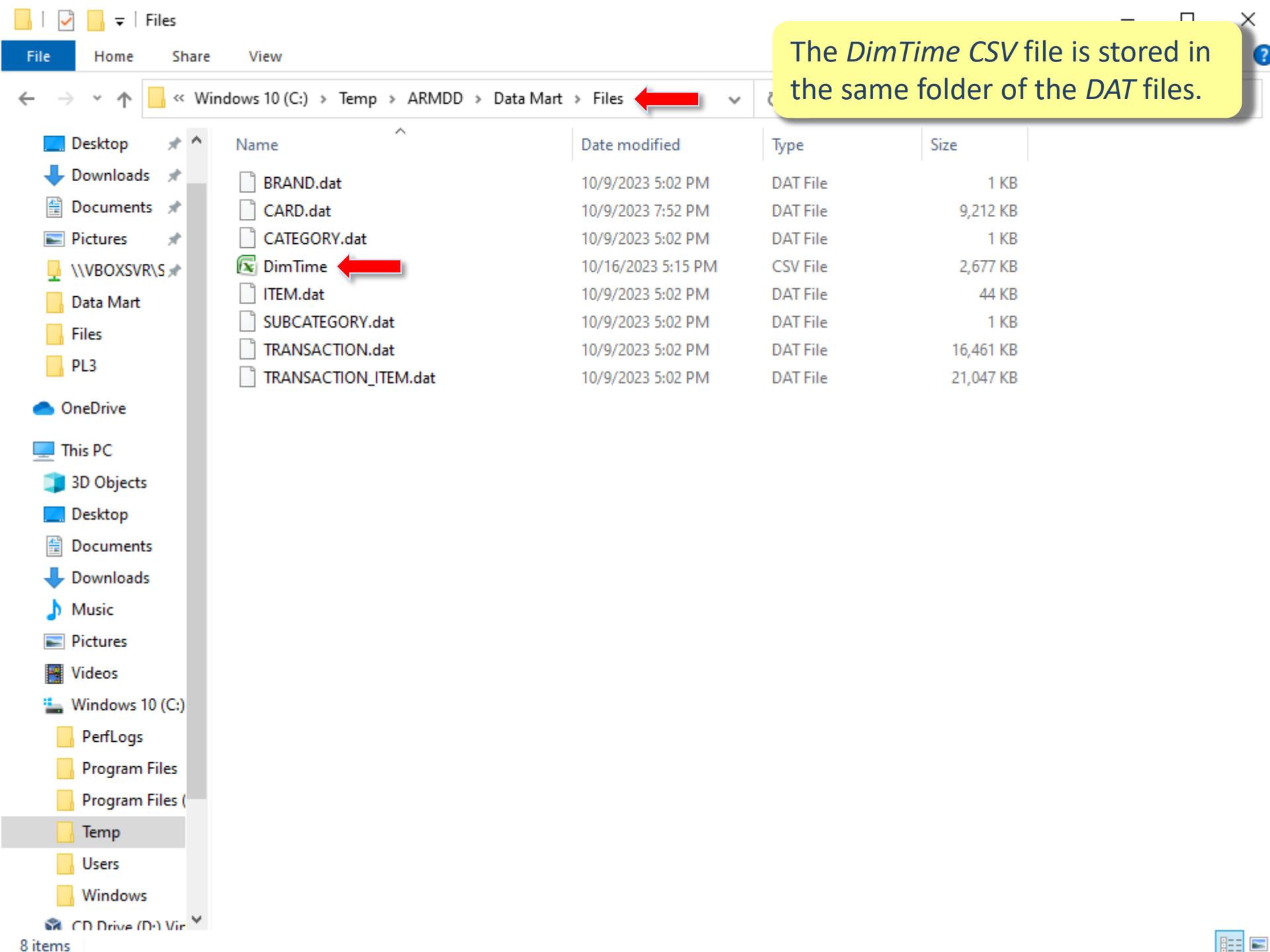
Font: Calibri 11pt, Alignment: General, Number: General

	A	B	C	D	E	F	G	H	I	J	K	L	M
50389	21:59:47	215947		215947	21	59	47	Evening	No	Yes			
50390	21:59:48	215948		215948	21	59	48	Evening	No	Yes			
50391	21:59:49	215949		215949	21	59	49	Evening	No	Yes			
50392	21:59:50	215950		215950	21	59	50	Evening	No	Yes			
50393	21:59:51	215951		215951	21	59	51	Evening	No	Yes			
50394	21:59:52	2	Microsoft Office Excel										
50395	21:59:53	2											
50396	21:59:54	2											
50397	21:59:55	2											
50398	21:59:56	2											
50399	21:59:57	2											
50400	21:59:58	2											
50401	21:59:59	215959		215959	21	59	59	Evening	No	Yes			
50402	22:00:00	220000		220000	22	0	0	Evening	No	No			
50403	22:00:01	220001		220001	22	0	1	Evening	No	No			
50404	22:00:02	220002		220002	22	0	2	Evening	No	No			
50405	22:00:03	220003		220003	22	0	3	Evening	No	No			
50406	22:00:04	220004		220004	22	0	4	Evening	No	No			
50407	22:00:05	220005		220005	22	0	5	Evening	No	No			
50408	22:00:06	220006		220006	22	0	6	Evening	No	No			
50409	22:00:07	220007		220007	22	0	7	Evening	No	No			
50410	22:00:08	220008		220008	22	0	8	Evening	No	No			
50411	22:00:09	220009		220009	22	0	9	Evening	No	No			
50412	22:00:10	220010		220010	22	0	10	Evening	No	No			
50413	22:00:11	220011		220011	22	0	11	Evening	No	No			

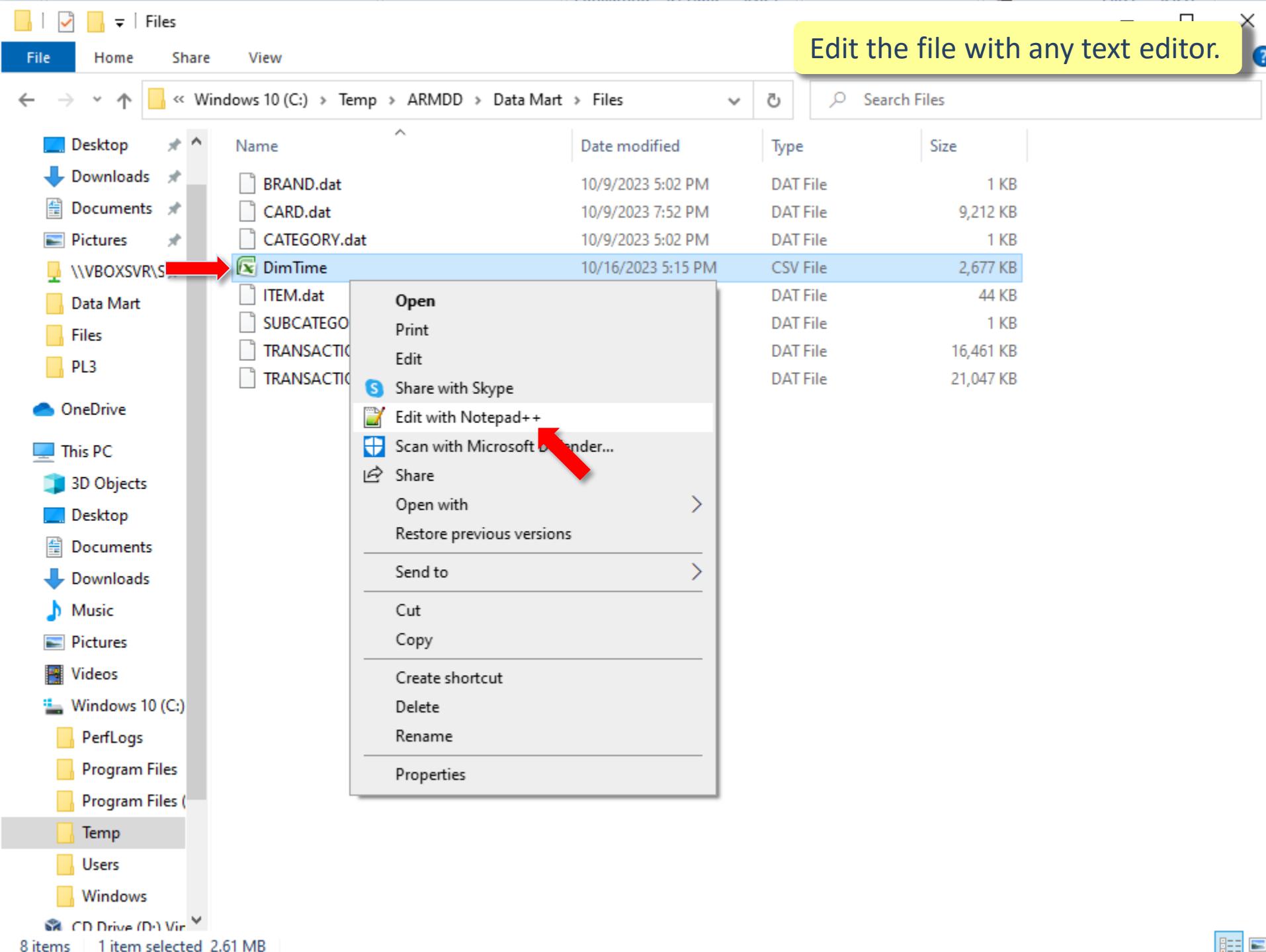
Sheet1 Sheet2 Sheet3

Ready 100%

The *DimTime* CSV file is stored in the same folder of the DAT files.



Edit the file with any text editor.



C:\Temp\ARPAD\Data Mart\Files\DimTime.csv - Notepad++

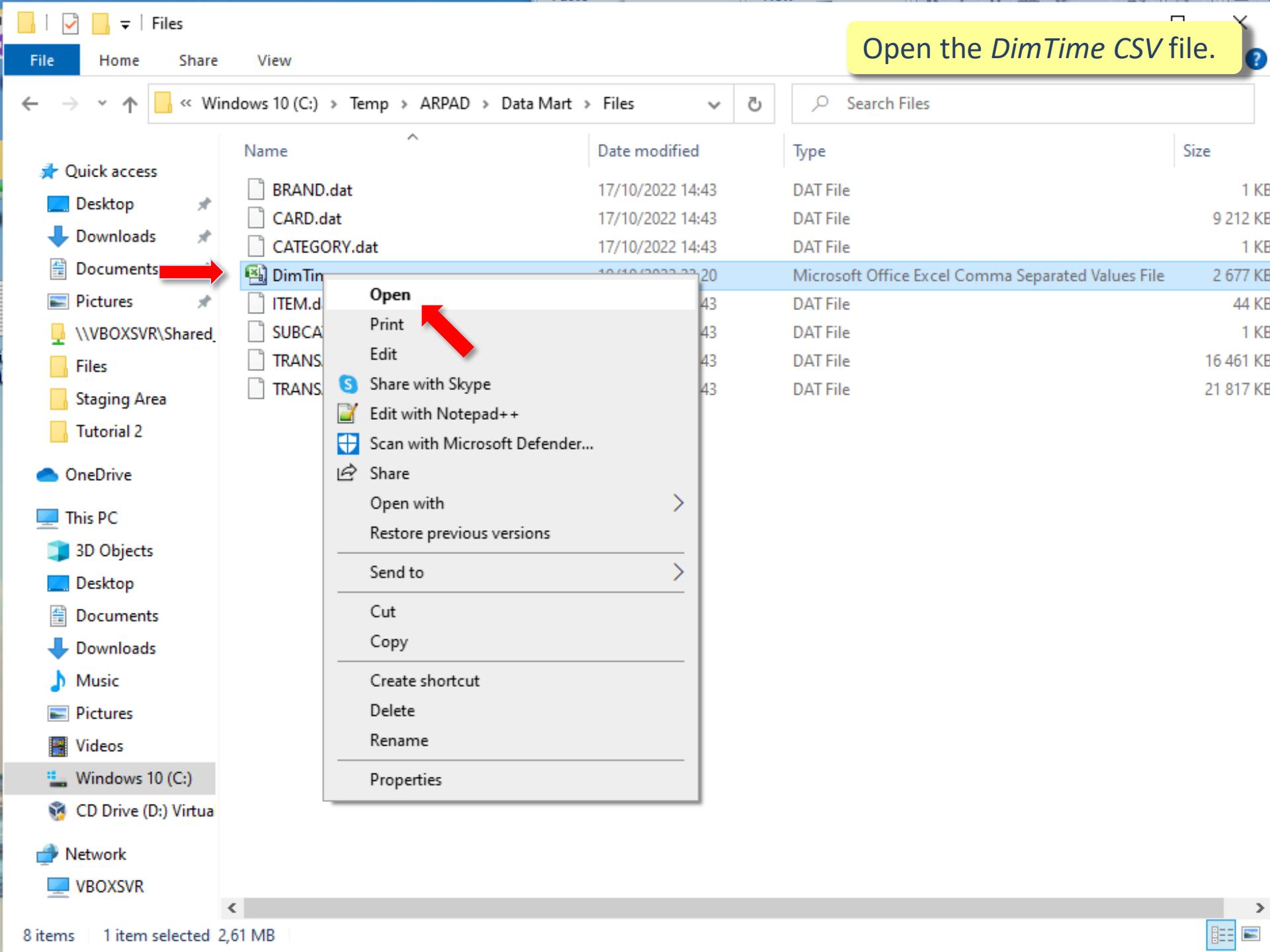
File Edit Search View Encoding Language Settings Tools Macro Run Plugins

change.log Dim Time.csv

1 FullTime;FullTime\_text;FullTime\_int;Hour :Minute;Second;  
2 08:00:00;80000;80000;8;0;0;Morning;No;No  
3 08:00:01;80001;80001;8;0;1;Morning;No;No  
4 08:00:02;80002;80002;8;0;2;Morning;No;No  
5 08:00:03;80003;80003;8;0;3;Morning;No;No  
6 08:00:04;80004;80004;8;0;4;Morning;No;No  
7 08:00:05;80005;80005;8;0;5;Morning;No;No  
8 08:00:06;80006;80006;8;0;6;Morning;No;No  
9 08:00:07;80007;80007;8;0;7;Morning;No;No  
10 08:00:08;80008;80008;8;0;8;Morning;No;No  
11 08:00:09;80009;80009;8;0;9;Morning;No;No  
12 08:00:10;80010;80010;8;0;10;Morning;No;No  
13 08:00:11;80011;80011;8;0;11;Morning;No;No  
14 08:00:12;80012;80012;8;0;12;Morning;No;No  
15 08:00:13;80013;80013;8;0;13;Morning;No;No  
16 08:00:14;80014;80014;8;0;14;Morning;No;No  
17 08:00:15;80015;80015;8;0;15;Morning;No;No  
18 08:00:16;80016;80016;8;0;16;Morning;No;No  
19 08:00:17;80017;80017;8;0;17;Morning;No;No  
20 08:00:18;80018;80018;8;0;18;Morning;No;No  
21 08:00:19;80019;80019;8;0;19;Morning;No;No  
22 08:00:20;80020;80020;8;0;20;Morning;No;No  
23 08:00:21;80021;80021;8;0;21;Morning;No;No  
24 08:00:22;80022;80022;8;0;22;Morning;No;No  
25 08:00:23;80023;80023;8;0;23;Morning;No;No  
26 08:00:24;80024;80024;8;0;24;Morning;No;No  
27 08:00:25;80025;80025;8;0;25;Morning;No;No  
28 08:00:26;80026;80026;8;0;26;Morning;No;No  
29 08:00:27;80027;80027;8;0;27;Morning;No;No  
30 08:00:28;80028;80028;8;0;28;Morning;No;No  
31 08:00:29;80029;80029;8;0;29;Morning;No;No  
32 08:00:30;80030;80030;8;0;30;Morning;No;No  
33 08:00:31;80031;80031;8;0;31;Morning;No;No  
34 08:00:32;80032;80032;8;0;32;Morning;No;No  
35 08:00:33;80033;80033;8;0;33;Morning;No;No  
...  
Normal text file length: 2 / 40 285 lines: 5 / 602 Ln: 12 Col: 42 Pos: 547 Windows (CR LF) UTF-8 INS

The file is a CSV, although the separator is the semicolon instead of a comma. This is normal/expected and results from the Portuguese regional setting.

Open the DimTime CSV file.



DimTime - Microsoft Excel

Although a CSV file, it can easily be opened in Excel to perform changes or add other columns if needed.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	FullTime	FullTime_	FullTime_	Hour	Minute	Second	Period	LunchTime	DinnerTime					
2	08:00:00	80000	80000	8	0	0	0 Morning	No	No					
3	08:00:01	80001	80001	8	0	0	1 Morning	No	No					
4	08:00:02	80002	80002	8	0	0	2 Morning	No	No					
5	08:00:03	80003	80003	8	0	0	3 Morning	No	No					
6	08:00:04	80004	80004	8	0	0	4 Morning	No	No					
7	08:00:05	80005	80005	8	0	0	5 Morning	No	No					
8	08:00:06	80006	80006	8	0	0	6 Morning	No	No					
9	08:00:07	80007	80007	8	0	0	7 Morning	No	No					
10	08:00:08	80008	80008	8	0	0	8 Morning	No	No					
11	08:00:09	80009	80009	8	0	0	9 Morning	No	No					
12	08:00:10	80010	80010	8	0	0	10 Morning	No	No					
13	08:00:11	80011	80011	8	0	0	11 Morning	No	No					
14	08:00:12	80012	80012	8	0	0	12 Morning	No	No					
15	08:00:13	80013	80013	8	0	0	13 Morning	No	No					
16	08:00:14	80014	80014	8	0	0	14 Morning	No	No					
17	08:00:15	80015	80015	8	0	0	15 Morning	No	No					
18	08:00:16	80016	80016	8	0	0	16 Morning	No	No					
19	08:00:17	80017	80017	8	0	0	17 Morning	No	No					
20	08:00:18	80018	80018	8	0	0	18 Morning	No	No					
21	08:00:19	80019	80019	8	0	0	19 Morning	No	No					
22	08:00:20	80020	80020	8	0	0	20 Morning	No	No					
23	08:00:21	80021	80021	8	0	0	21 Morning	No	No					
24	08:00:22	80022	80022	8	0	0	22 Morning	No	No					
25	08:00:23	80023	80023	8	0	0	23 Morning	No	No					