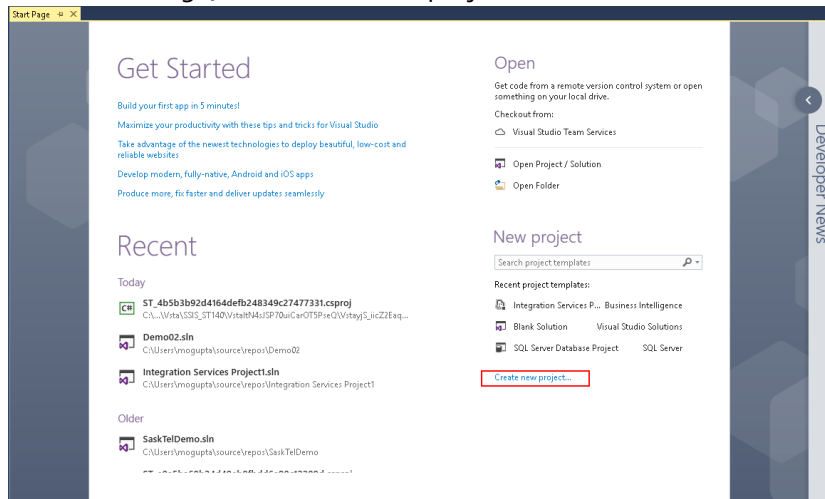


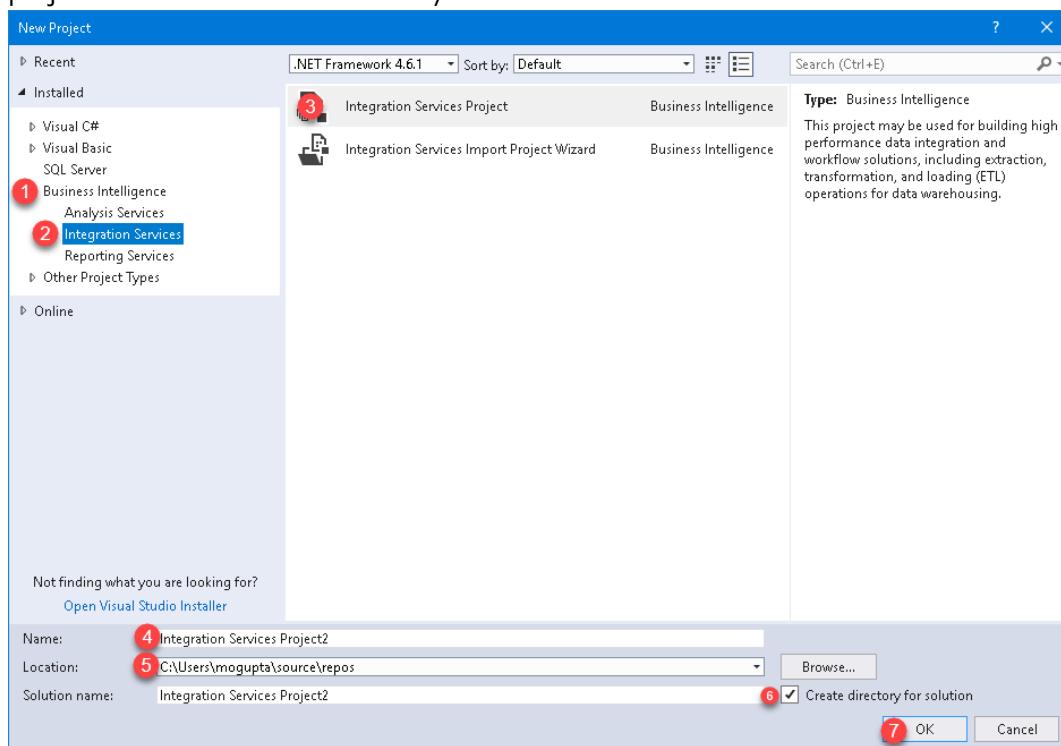
SQL SERVER INTEGRATION SERVICES

MODULE 03: CONTROL FLOW: BRING IT TOGETHER

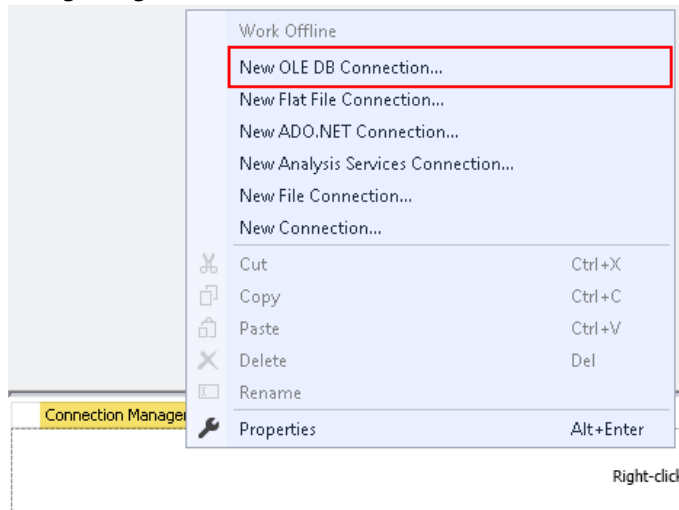
1. Launch SQL Server Data Tools (SSDT), under start menu look for Visual Studio 2017 (SSDT).
2. In the Start Page, click Create new project.



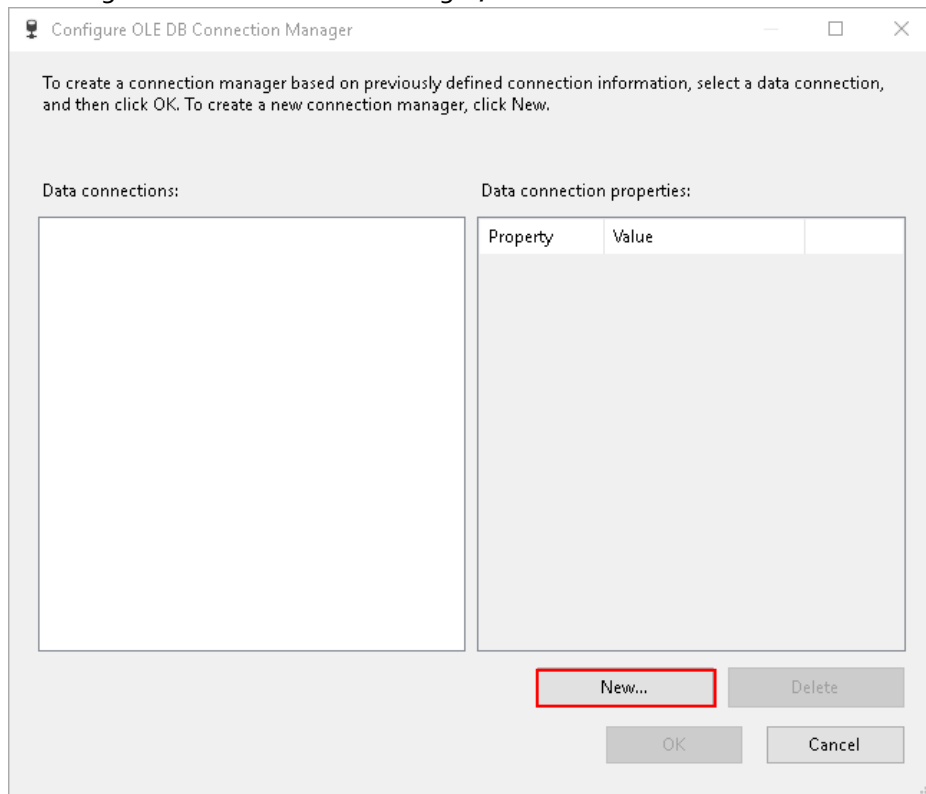
3. In New Project dialog box select Business Intelligence > Integration Services > Integration Services Project. On the bottom enter in project name and location you wish to save the project. Make sure "Create directory for solution" is selected and click OK.



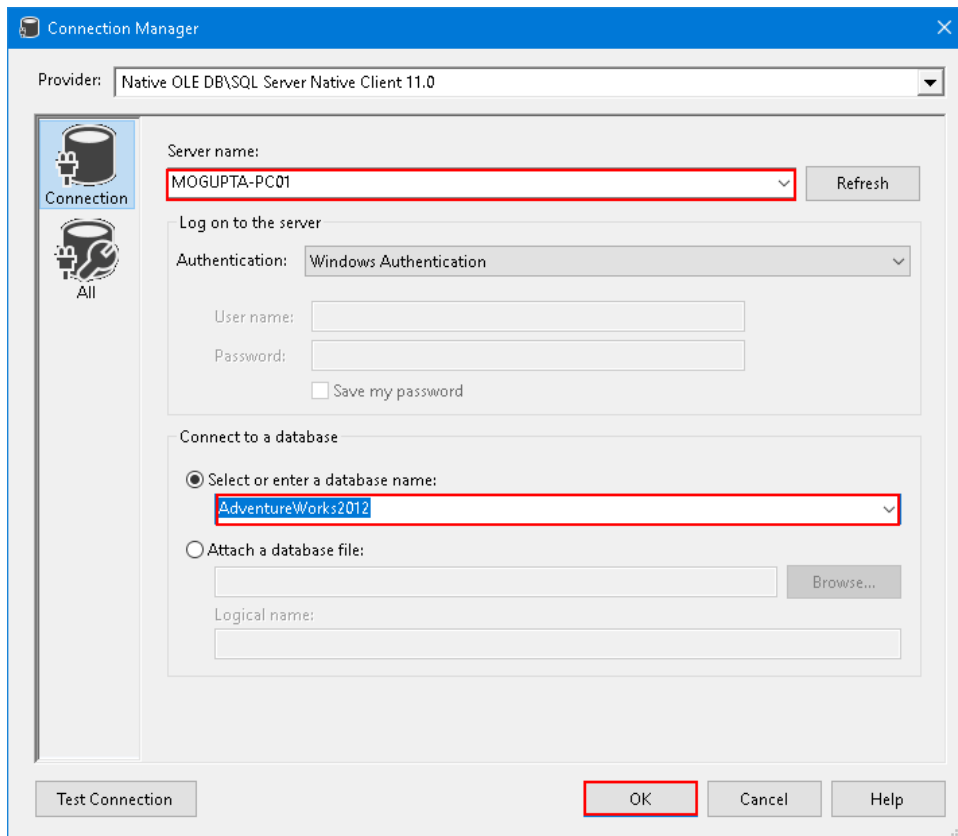
- Let's set up a connection manager to our database. In the bottom center pain under Connection Manger, right-click select New Ole-DB Connection.



- In Configure OLE DB Connect Manager, click New.



- In connection manager, type the server name, select the database "AdventureWorks2012" and click OK.



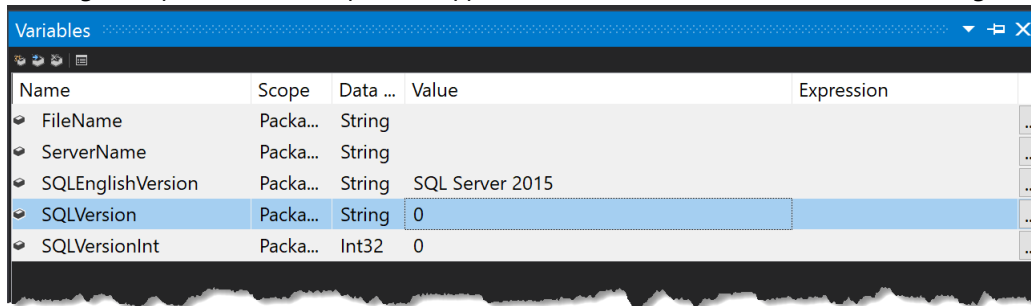
- Click OK in Configure OLE DB Connection Manager. You should see a single connection under Connection Manager.

Hint: Rename the connection manager to OLEDB.ServerName.DatabaseName. This will make it easier to identify which driver is being used for the driver.

- Setup a Package Parameter assign it value "SELECT @@ServerName AS ServerName". Set Required to True.

Package.dtsx [Design]*					
Control Flow Data Flow Paramet... Event Handl... Package Explo... Execution Results					
Name	Data ...	Value	Se...	Re...	Description
SQLStmt	String	SELECT @@ServerName AS ServerName	False	True	

- Setup variables for the project. Go to SSIS Menu > Variables. Then create following variables in Package Scope and their respective types (Note there is default value for SQLEnglishVersion).

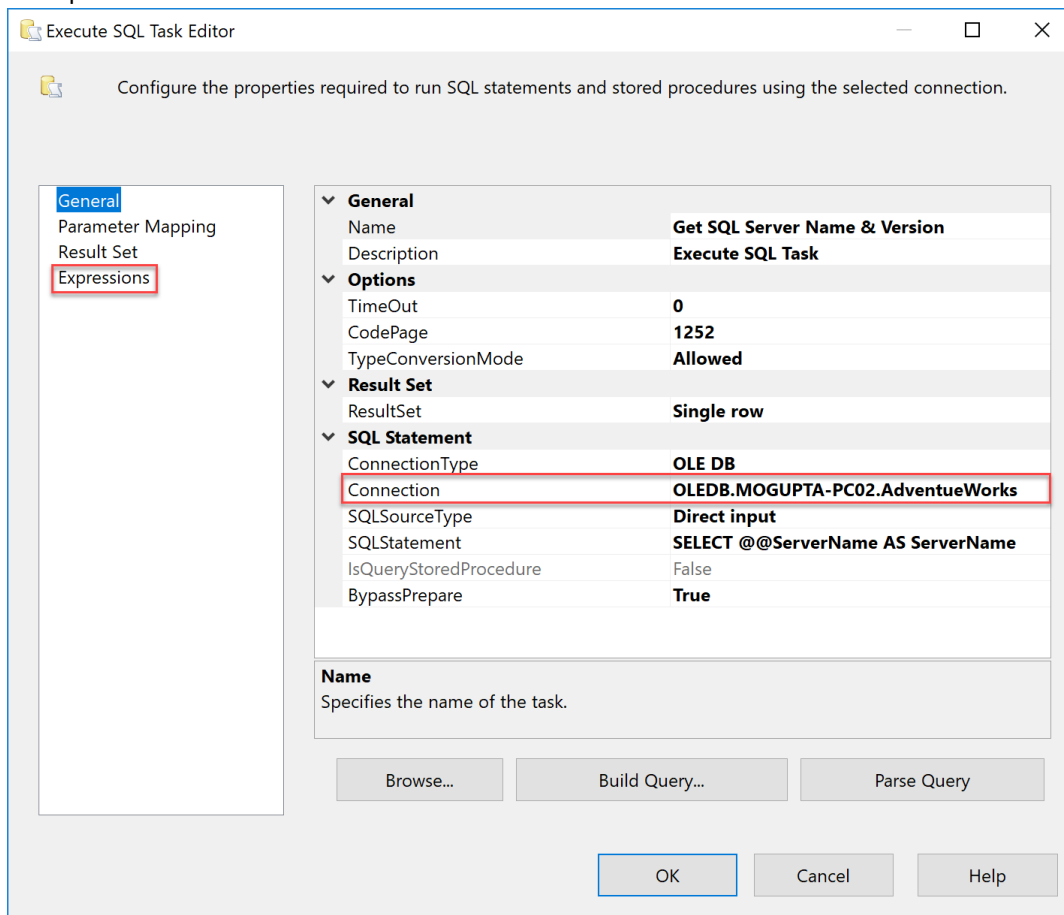


Name	Scope	Data ...	Value	Expression
FileName	Packa...	String		
ServerName	Packa...	String		
SQLEnglishVersion	Packa...	String	SQL Server 2015	
SQLVersion	Packa...	String	0	
SQLVersionInt	Packa...	Int32	0	

- Create an Execute SQL Task. Rename it to "Get SQL Server Name & Version".

Hint: You can rename a task by double-clicking and updating the Name attribute. You can also select the Task and press F2 to rename it.

- Connect the "Get SQL Server Name & Version" to OLEDB connection created in step #7 and go to Expressions.



Execute SQL Task Editor

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

General

Name: **Get SQL Server Name & Version**

Description: **Execute SQL Task**

Options

TimeOut: **0**

CodePage: **1252**

TypeConversionMode: **Allowed**

Result Set

ResultSet: **Single row**

SQL Statement

ConnectionType: **OLE DB**

Connection: **OLEDB.MOGUPTA-PC02.AdventueWorks**

SQLSourceType: **Direct input**

SQLStatement: **SELECT @@ServerName AS ServerName**

IsQueryStoredProcedure: **False**

BypassPrepare: **True**

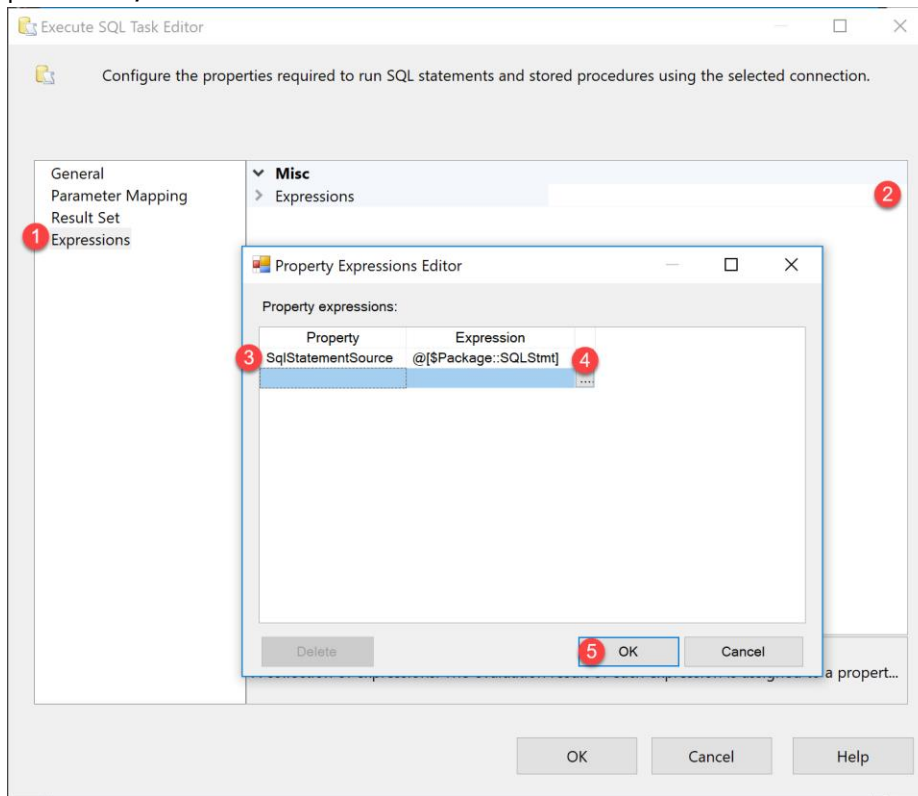
Name

Specifies the name of the task.

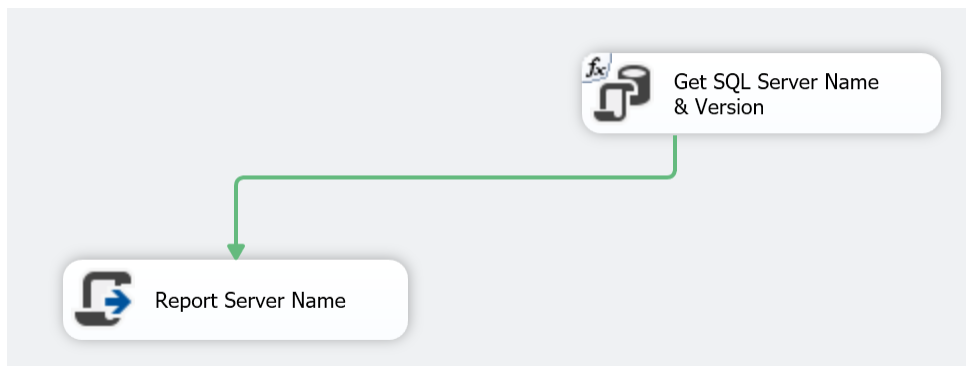
Browse... Build Query... Parse Query

OK Cancel Help

- Under expressions, build an expression and attach the SqlStatementSource to package parameter, **SQLStmt**.

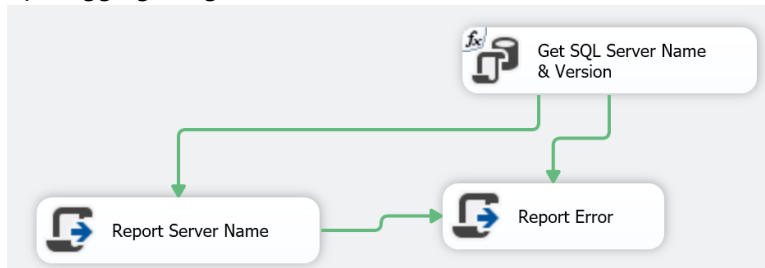


- Create a Script Task. Rename it to "Report Server Name".
- Connect the "Get SQL Server Name & Version" to "Report Server Name" by dragging the green arrow to task.

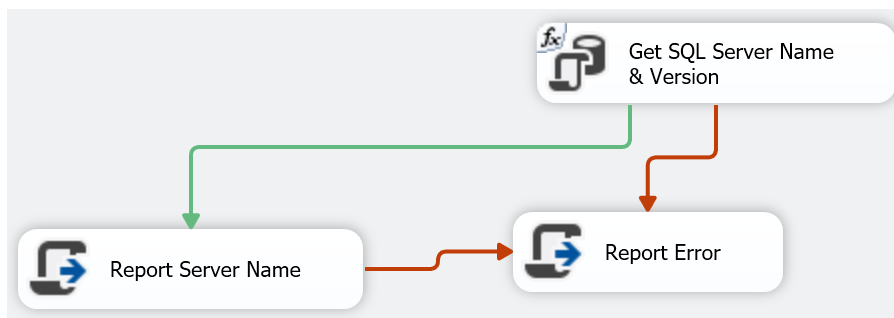
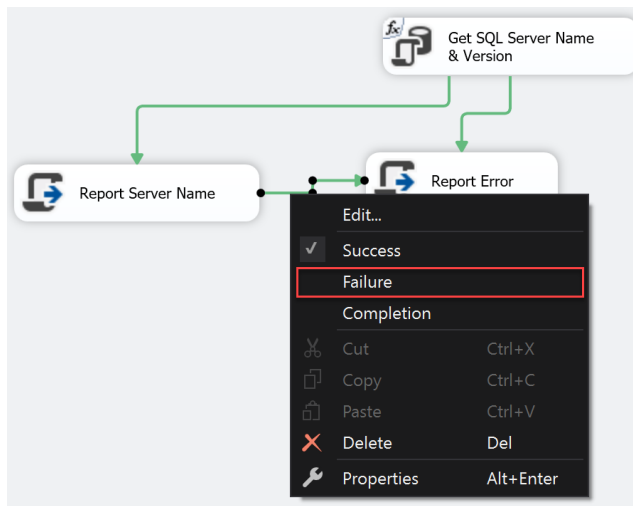


- Create another Script Task. Rename it to "Report Error".

16. Connect both “Report Server Name” and “Get SQL Server Name & Version” to “Report Error” by dragging the green arrows.



17. However, since we want to report the error on a failure, we will change connections to on failure. Right click on the green line connecting each of the tasks to Report Error and Select Failure. After both lines are updated, it should be similar to below.



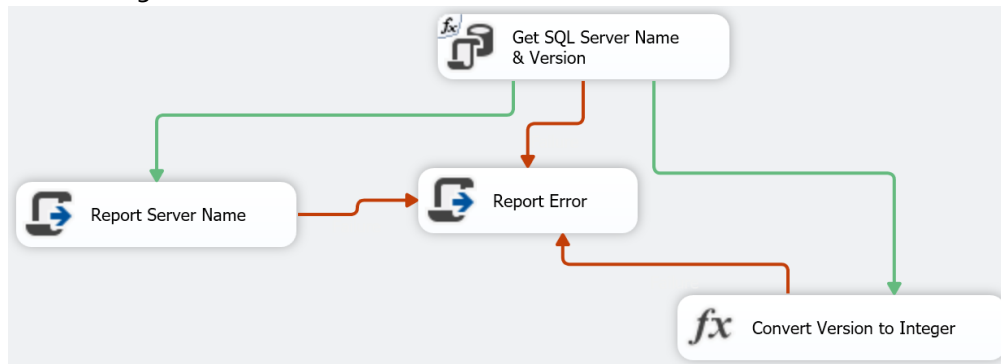
18. Update each Script task with their respect C# Code. Open the script task, assign variable, and type the following script in the Main() function.

Script Task Name	ReadOnlyVariables	Script
Report Server Name	User::ServerName	<code>MessageBox.Show("SQL Server Name: " + Dts.Variables["User::ServerName"].Value.ToString(), "Server Name");</code>
Report Error	Blank-No-Assignment	<code>MessageBox.Show("Uhhh! I think something went wrong!", "Help!");</code>

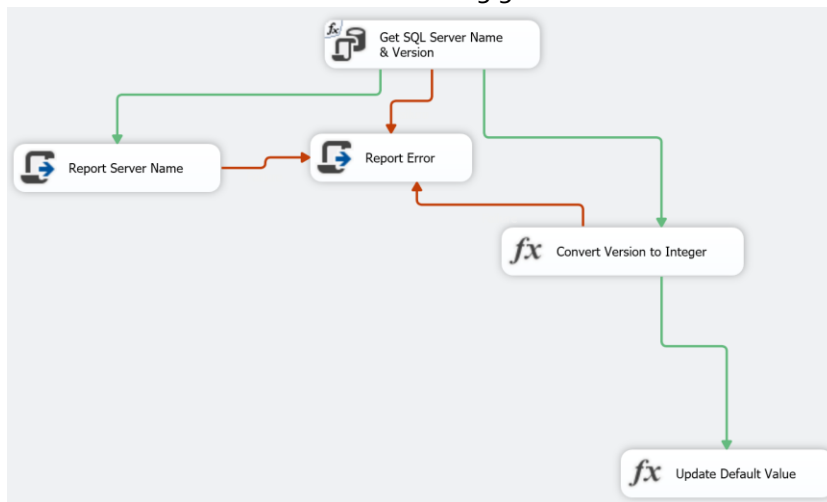
19. Next create an Expression Task, rename it to "Convert Version to Integer" and set the expression to "@[User::SQLVersionInt] = (DT_I4) @[User::SQLVersion]".



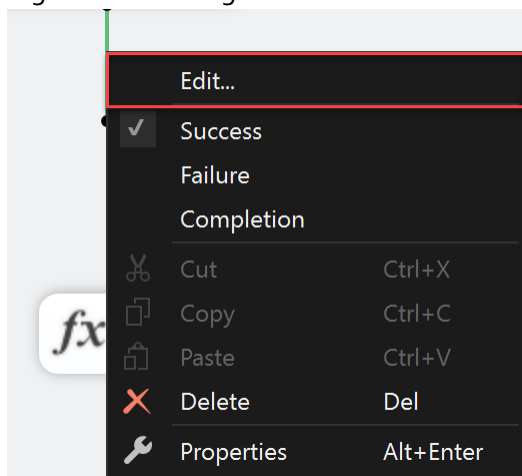
20. After building the express task, connect, "Get SQL Server Name & Version" to "Convert Version to Integer" with Green arrow (Success) and connect "Convert Version to Integer" to "Report Error" using Red arrow (Failure).



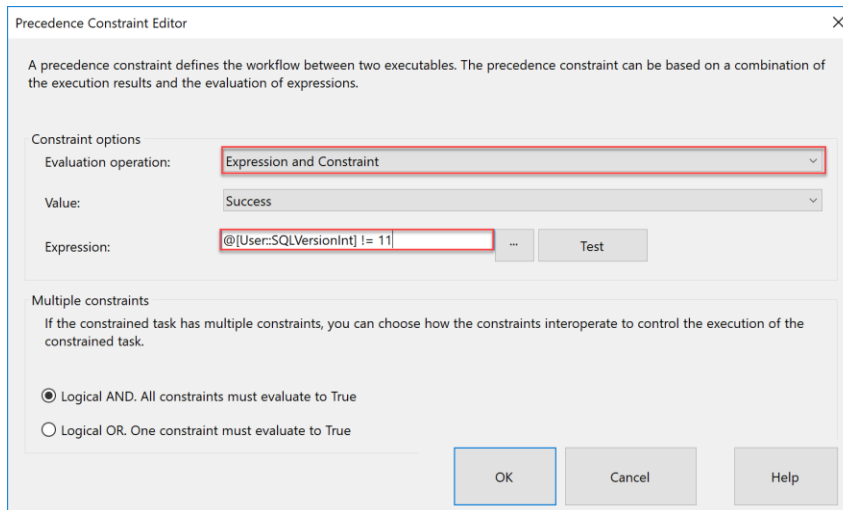
21. Next build another Expression Task, rename it to "Update Default Value". Set the expression to "@[User::SQLEnglishVersion] = "SQL Server 2017"". Connect the "Update Default Value" to "Get SQL Server Name & Version" using green arrow.



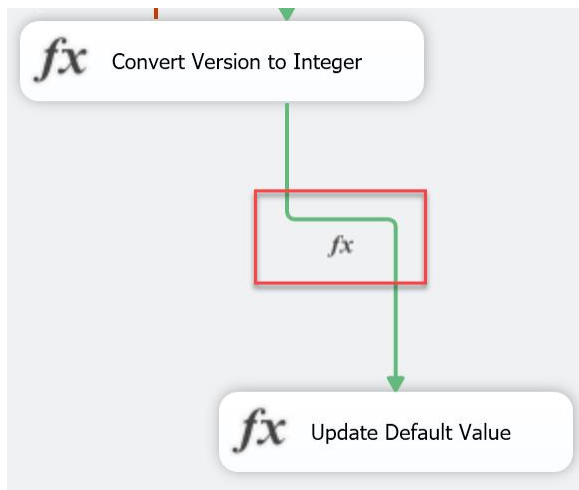
22. We are going to update the expression, for the Update Default Value to make it a bit restrictive. Right click on the green line and select Edit.



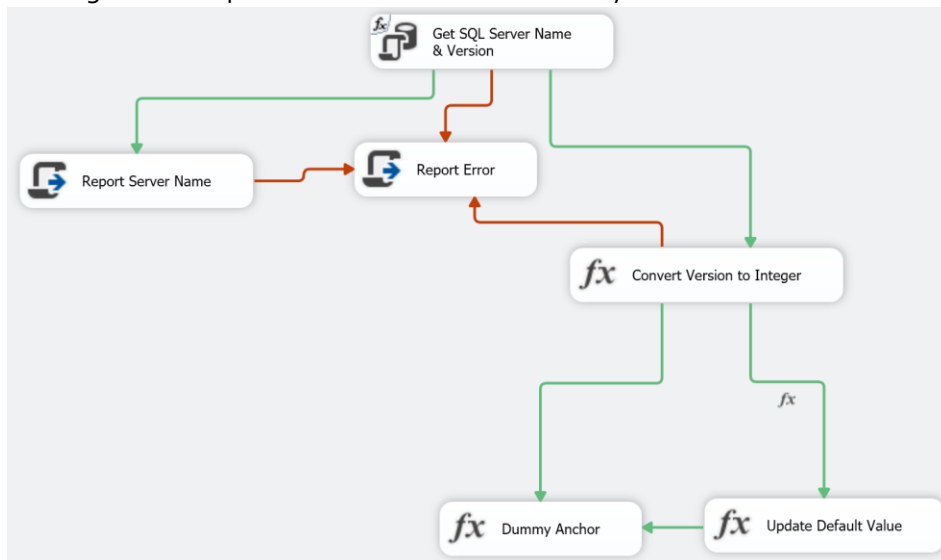
23. Change the Evaluation operation to Expression and Constraint and change the Expression to “@[User::SQLVersionInt] != 11”. Note after the precedence constraint is built an “fx” added to the constraint line.



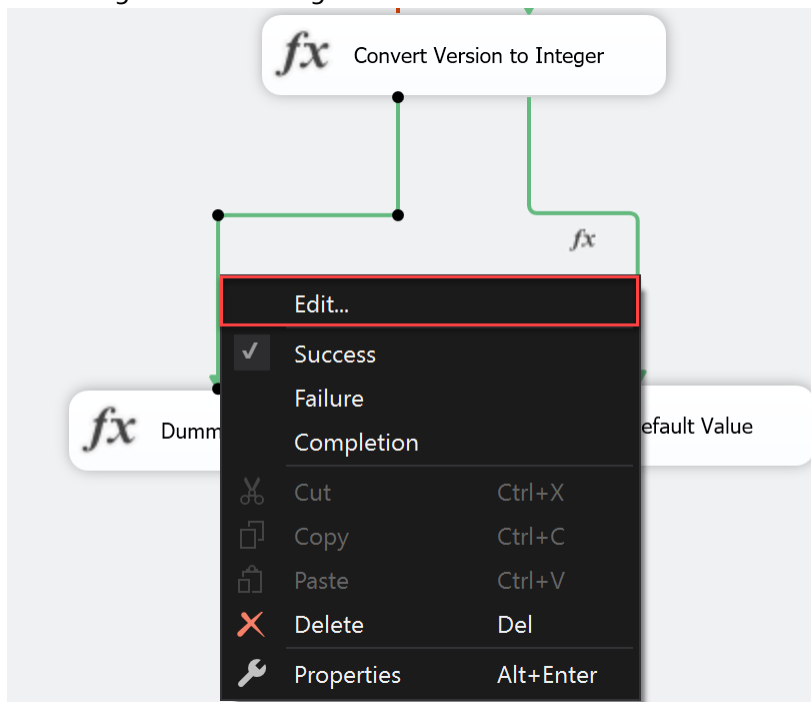
The image shows the 'Precedence Constraint Editor' dialog box. It has a title bar with a close button. The main text explains that a precedence constraint defines the workflow between two executables. The 'Constraint options' section includes a dropdown for 'Evaluation operation' set to 'Expression and Constraint', a dropdown for 'Value' set to 'Success', and a text box for 'Expression' containing '@[User::SQLVersionInt] != 11'. There are '...' and 'Test' buttons next to the expression field. The 'Multiple constraints' section has two radio buttons: 'Logical AND. All constraints must evaluate to True' (selected) and 'Logical OR. One constraint must evaluate to True'. At the bottom are 'OK', 'Cancel', and 'Help' buttons.



24. Create another Expression Task, rename it to “Dummy Anchor”. Set the expression to “@[User::SQLEnglishVersion] = @[User::SQLEnglishVersion]”. Connect the “Convert Version to Integer” and “Update Default Value” to “Dummy Anchor” task.



25. We cannot have both tasks from “Convert” and “Update” going to Dummy Anchor because of function constraints. Therefore, we need to change the Multiple constraints logical operator from ADD Condition to OR condition. Right click on either of the green-lines connecting to the Dummy Anchor and select edit. In precedence constraint editor, select “Logical OR”. Notice the lines green line changed from solid to dotted.



Precedence Constraint Editor

A precedence constraint defines the workflow between two executables. The precedence constraint can be based on a combination of the execution results and the evaluation of expressions.

Constraint options

Evaluation operation: Constraint

Value: Success

Expression: ... Test

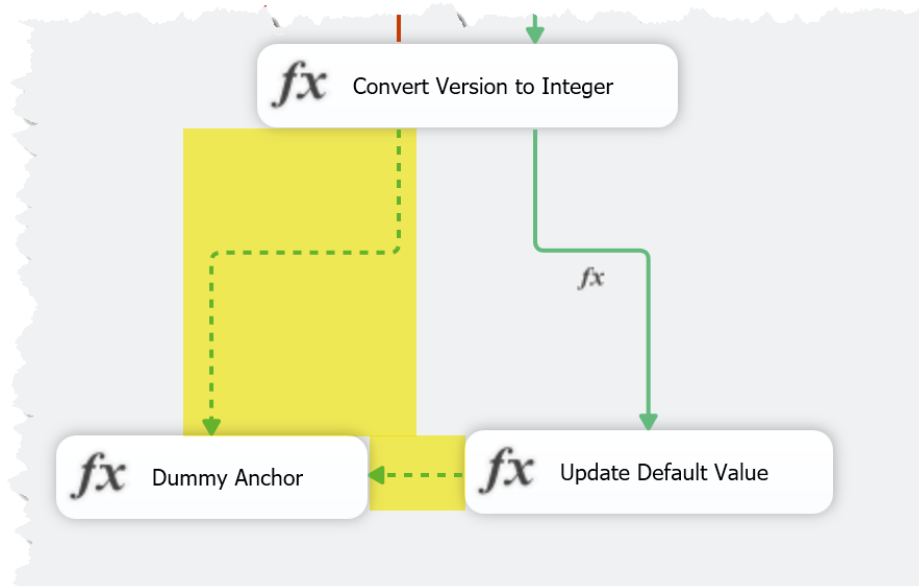
Multiple constraints

If the constrained task has multiple constraints, you can choose how the constraints interoperate to control the execution of the constrained task.

☐ Logical AND. All constraints must evaluate to True

☒ Logical OR. One constraint must evaluate to True

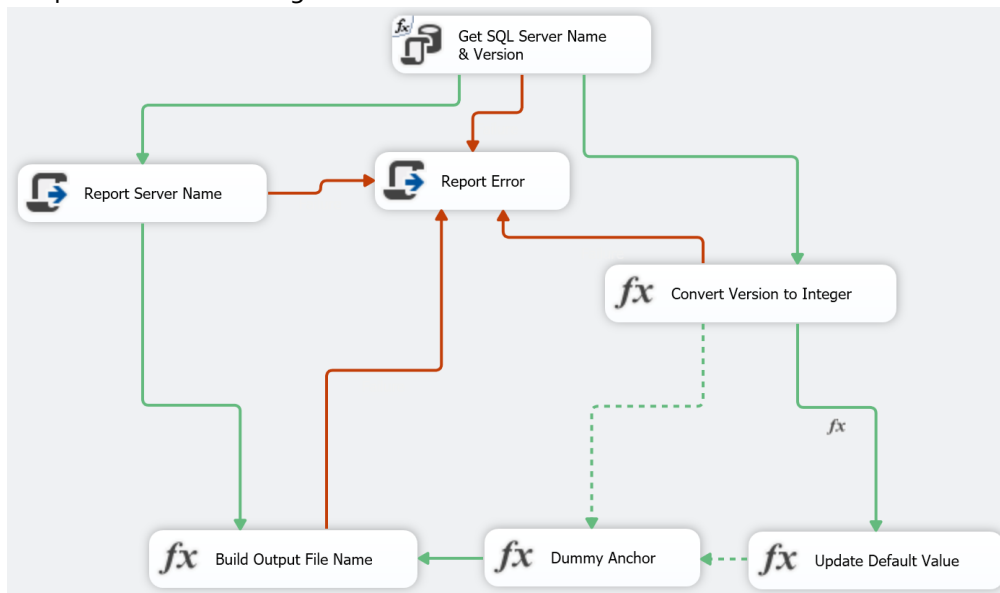
OK Cancel Help



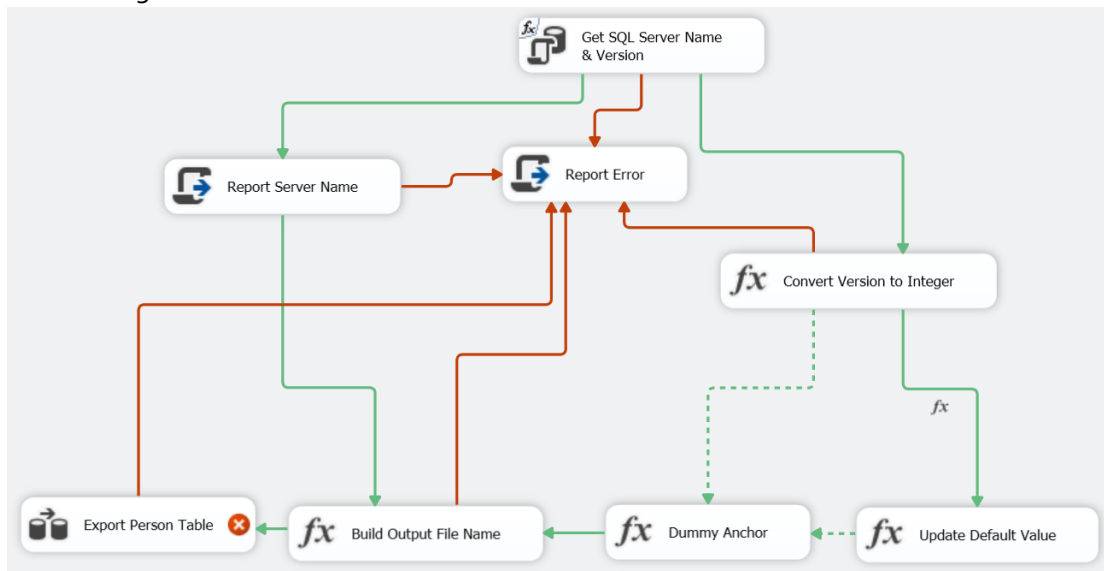
26. Create another Expression Task. Rename it "Build Output File Name". Set the expression to "`@[User::FileName] = "C:\\Temp\\" + @[User::ServerName] + "_" + @[User::SQLEnglishVersion] + "_Persons.txt"`" (Please update the starting path C:\\Temp\\ to your respective directory. Also note to escape each \\ with \\).

```
@[User::FileName] = "C:\\Temp\\" + @[User::ServerName] + "_" + @[User::SQLEnglishVersion] + "_Persons.txt"
```

27. Connect the “Build Output File Name” task to “Report Error” with Red line. Connect “Dummy Anchor” to “Build Output File Name” with Green line. Connect “Report Server Name” to “Build Output File Name” using Green line.

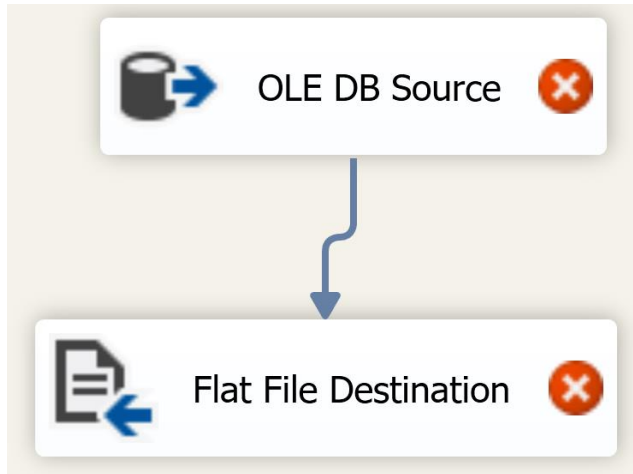


28. Next add a data flow task and rename it to “Export Person Table”. Connect “Build Output File Name” to “Export Person Table” using Green Line. Connect “Export Person Table” to “Report Error” using Red Line.

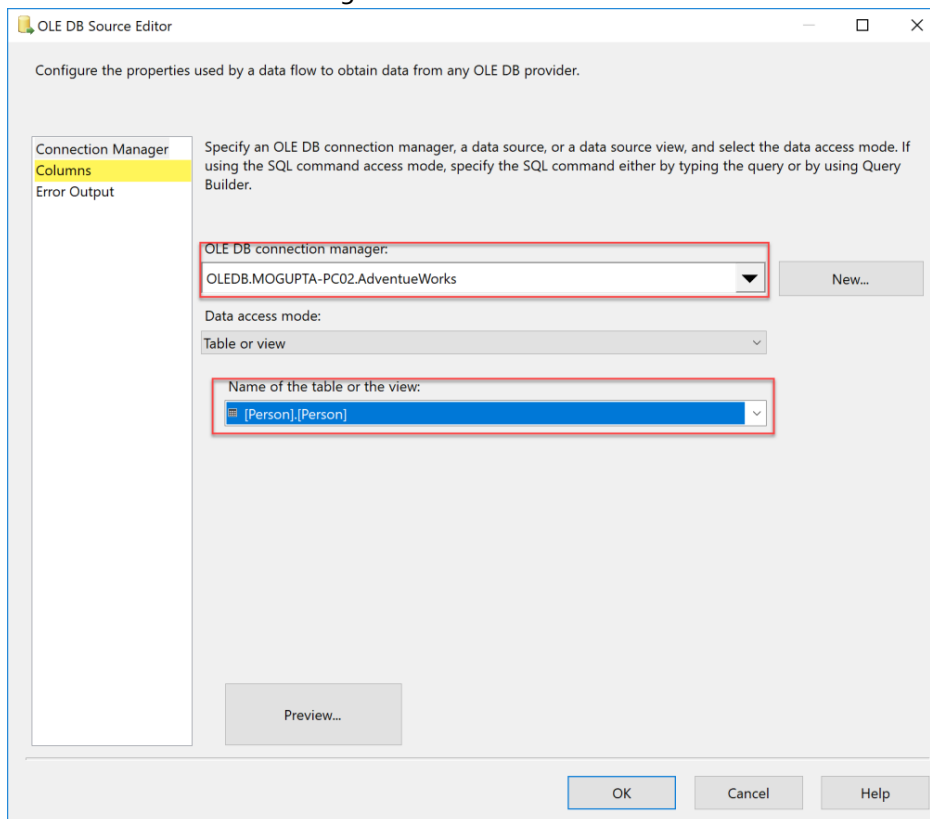


29. Double-click on the “Export Person Table” task to configure the data flow.

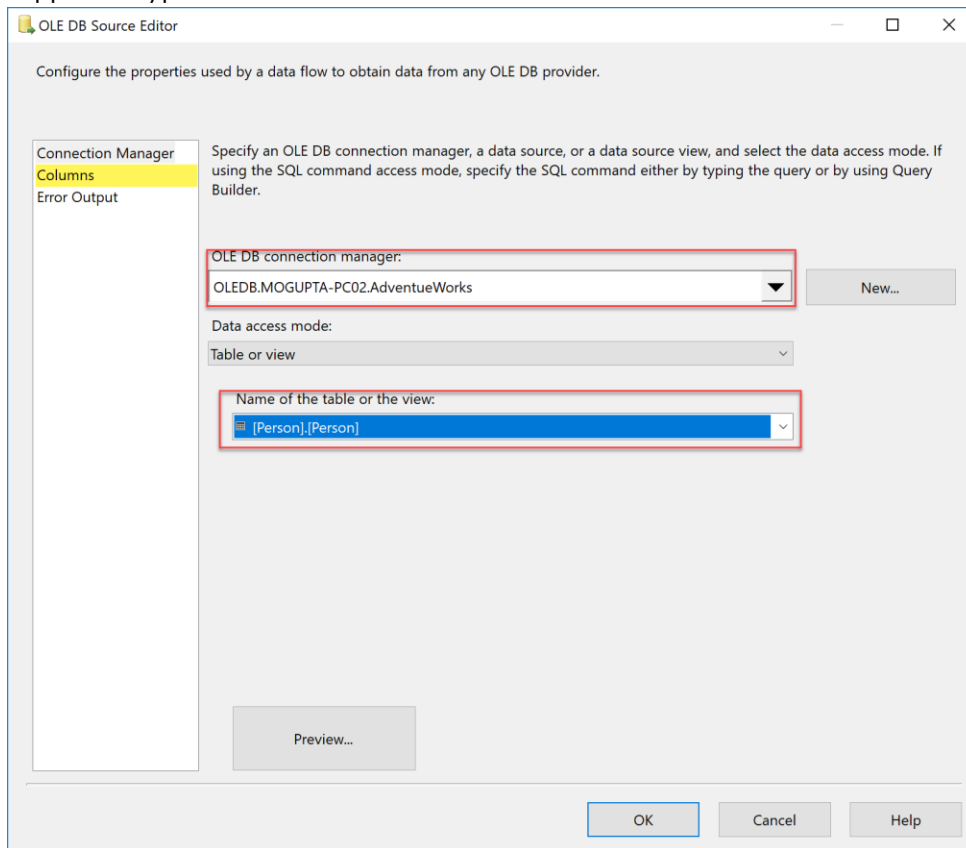
30. Create two tasks “OLE DB Source” and “Flat File Destination” and connect them.



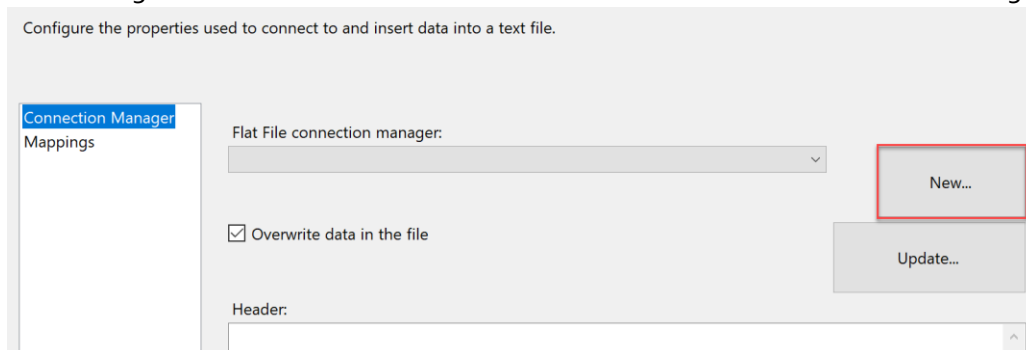
31. Configure the “OLE DB Source” to connect to our connection string and pull data from Person.Person. After setting those select Columns.



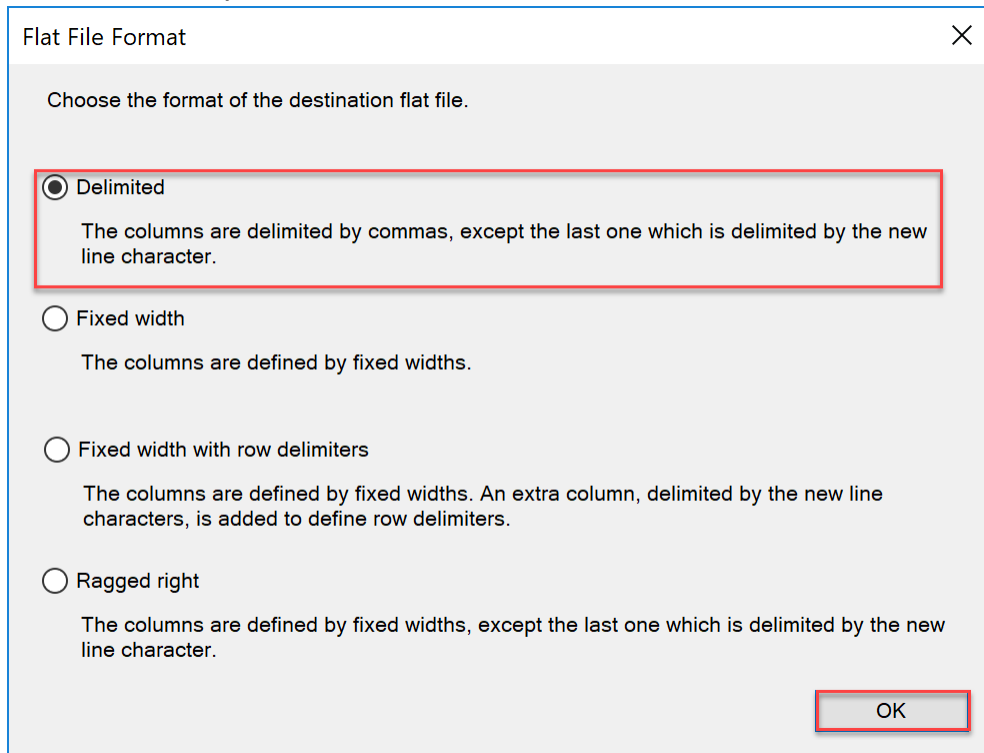
32. Only select columns shown below, we cannot export all columns to CSV because they are not supported types.



33. Next configure the "Flat File Destination". Create a New Flat File Connection Manager.



34. In Flat File Format, Select Delimited and click OK.



The image shows a 'Flat File Format' dialog box with a close button (X) in the top right corner. The main text reads 'Choose the format of the destination flat file.' There are four radio button options, each with a description. The 'Delimited' option is selected and highlighted with a red rectangle. The 'Fixed width' option is also highlighted with a red rectangle. The 'Fixed width with row delimiters' and 'Ragged right' options are not highlighted. The 'OK' button is highlighted with a red rectangle.

Flat File Format

Choose the format of the destination flat file.

☒ Delimited
The columns are delimited by commas, except the last one which is delimited by the new line character.

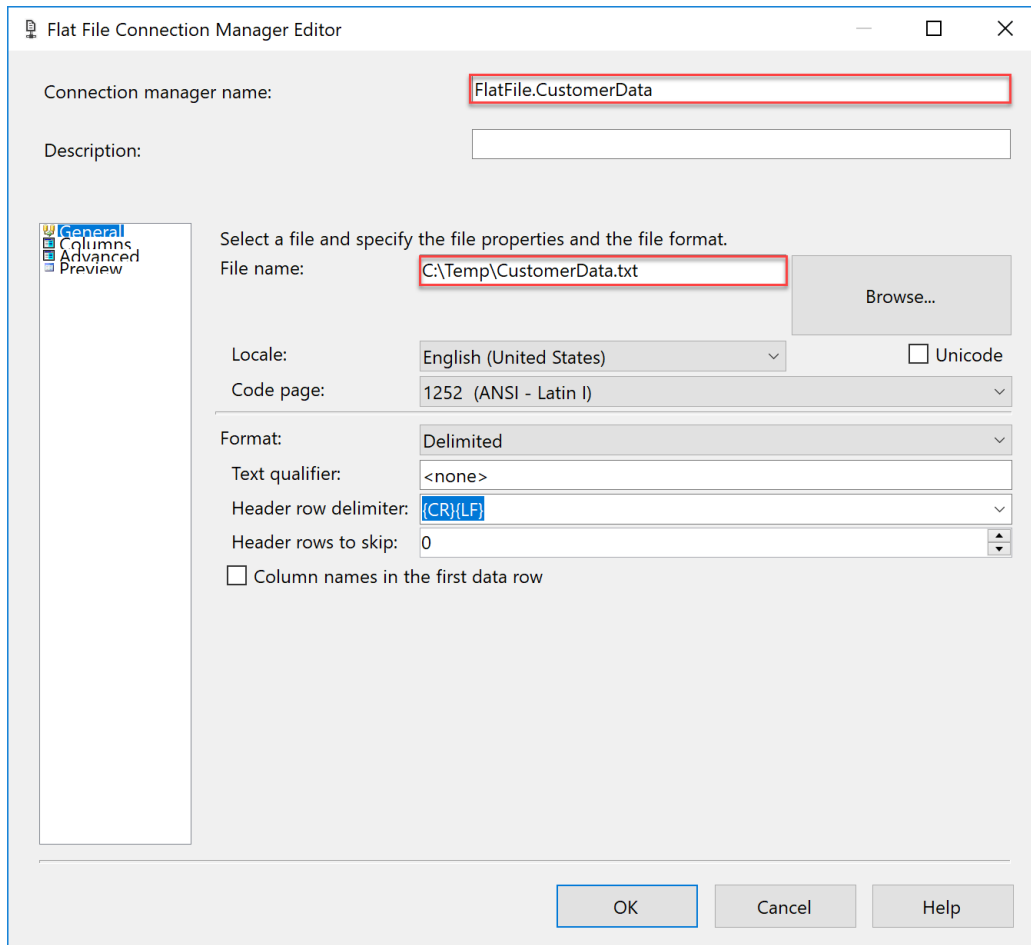
☐ Fixed width
The columns are defined by fixed widths.

☐ Fixed width with row delimiters
The columns are defined by fixed widths. An extra column, delimited by the new line characters, is added to define row delimiters.

☐ Ragged right
The columns are defined by fixed widths, except the last one which is delimited by the new line character.

OK

35. Rename the connection manager and set file name, doesn't have to be valid as we will be changing it via expressions.



The image shows the 'Flat File Connection Manager Editor' dialog box. It has a title bar with a minimize button, a maximize button, and a close button. The dialog is divided into several sections. At the top, there is a 'Connection manager name:' field with the text 'FlatFile.CustomerData' and a 'Description:' field. Below these is a tree view on the left with four items: 'General' (selected), 'Columns', 'Advanced', and 'Preview'. The main area is titled 'Select a file and specify the file properties and the file format.' It contains a 'File name:' field with the text 'C:\Temp\CustomerData.txt' and a 'Browse...' button. Below this are 'Locale:' and 'Code page:' dropdown menus. The 'Locale:' dropdown is set to 'English (United States)' and there is an unchecked 'Unicode' checkbox. The 'Code page:' dropdown is set to '1252 (ANSI - Latin I)'. Below these are 'Format:' and 'Text qualifier:' dropdown menus. The 'Format:' dropdown is set to 'Delimited'. The 'Text qualifier:' dropdown is set to '<none>'. Below these are 'Header row delimiter:' and 'Header rows to skip:' dropdown menus. The 'Header row delimiter:' dropdown is set to '(CR)(LF)' and the 'Header rows to skip:' dropdown is set to '0'. At the bottom, there is an unchecked checkbox labeled 'Column names in the first data row'. At the very bottom of the dialog are three buttons: 'OK', 'Cancel', and 'Help'.

Flat File Connection Manager Editor

Connection manager name: FlatFile.CustomerData

Description:

General
Columns
Advanced
Preview

Select a file and specify the file properties and the file format.

File name: C:\Temp\CustomerData.txt Browse...

Locale: English (United States) ☐ Unicode

Code page: 1252 (ANSI - Latin I)

Format: Delimited

Text qualifier: <none>

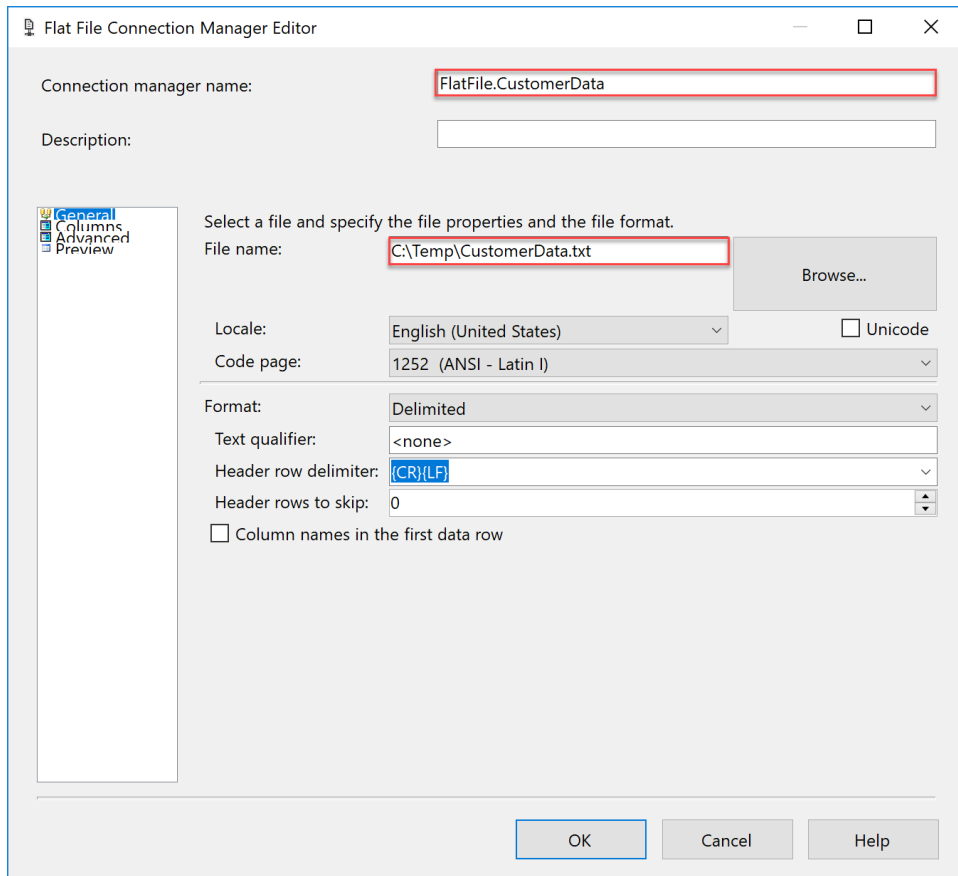
Header row delimiter: (CR)(LF)

Header rows to skip: 0

☐ Column names in the first data row

OK Cancel Help

36. Before you close the Flat File Destination Editor, click on Mappings to confirm column mapping and click OK.



The image shows the 'Flat File Connection Manager Editor' dialog box. It has a title bar with a document icon, the text 'Flat File Connection Manager Editor', and standard window controls. The dialog is divided into several sections. At the top, there is a 'Connection manager name:' field with the text 'FlatFile.CustomerData' and a 'Description:' field. Below these is a tabbed interface with three tabs: 'General' (selected), 'Columns', and 'Advanced'. The 'General' tab contains the following fields: 'File name:' with the text 'C:\Temp\CustomerData.txt' and a 'Browse...' button; 'Locale:' with a dropdown menu set to 'English (United States)' and an unchecked 'Unicode' checkbox; 'Code page:' with a dropdown menu set to '1252 (ANSI - Latin I)'; 'Format:' with a dropdown menu set to 'Delimited'; 'Text qualifier:' with a text box containing '<none>'; 'Header row delimiter:' with a dropdown menu set to '(CR)(LF)'; and 'Header rows to skip:' with a spinner box set to '0'. There is also an unchecked checkbox labeled 'Column names in the first data row'. At the bottom of the dialog are three buttons: 'OK', 'Cancel', and 'Help'.

Flat File Connection Manager Editor

Connection manager name: FlatFile.CustomerData

Description:

General Columns Advanced Preview

Select a file and specify the file properties and the file format.

File name: C:\Temp\CustomerData.txt Browse...

Locale: English (United States) ☐ Unicode

Code page: 1252 (ANSI - Latin I)

Format: Delimited

Text qualifier: <none>

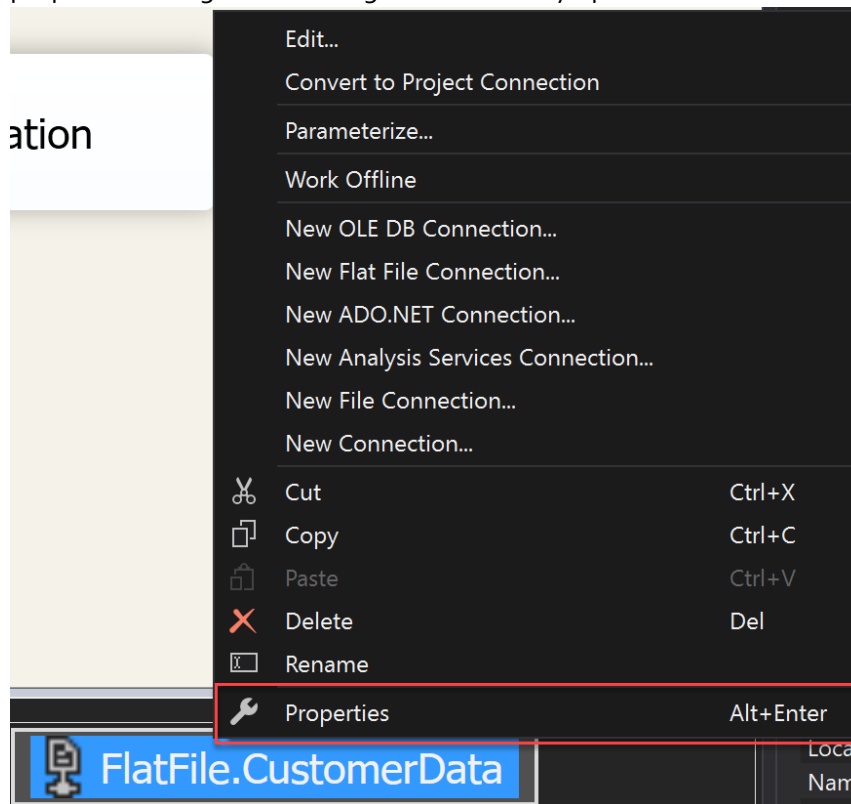
Header row delimiter: (CR)(LF)

Header rows to skip: 0

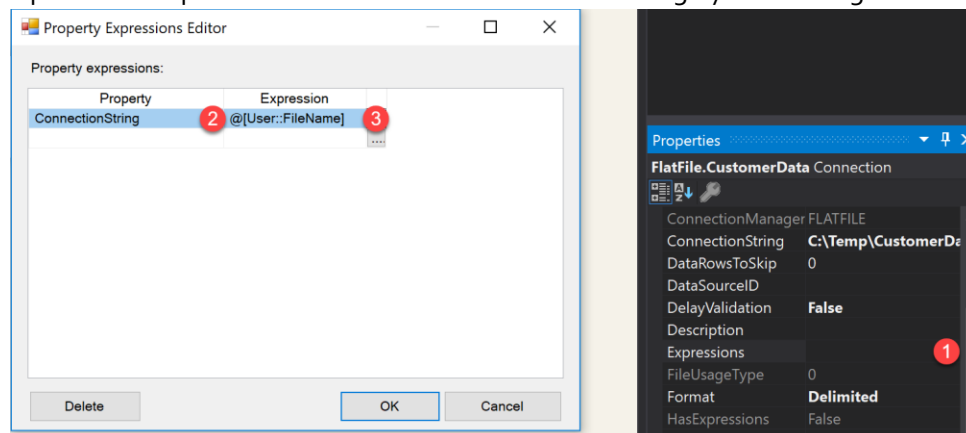
☐ Column names in the first data row

OK Cancel Help

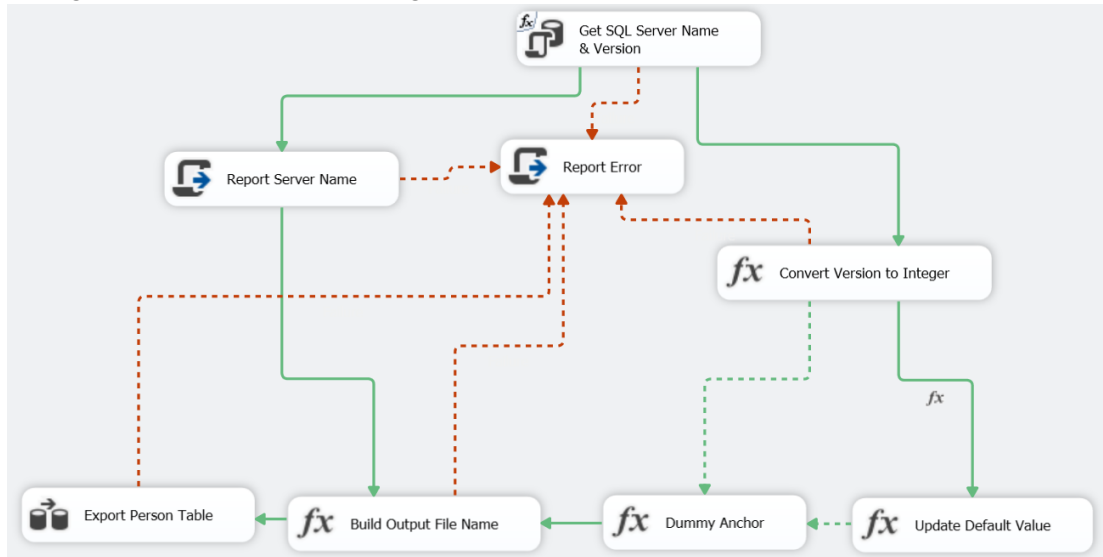
37. Right click on the FlatFile.CustomerData connection and select properties. This will open properties dialog in bottom right if not already open..



38. Update the expression information for connection string by connecting it to FileName variable.



39. Change the multiple constraint logical condition to OR for Red Lines. So, it looks like below.



40. Execute the package, did it execute successfully? If not why?

41. The parameter value is incorrect, update it to "SELECT @@ServerName AS ServerName, SERVERPROPERTY('ProductMajorVersion') AS SQLVersion".

42. Execute package again to see the execution path.