SSIS Hack – SSIS Migration

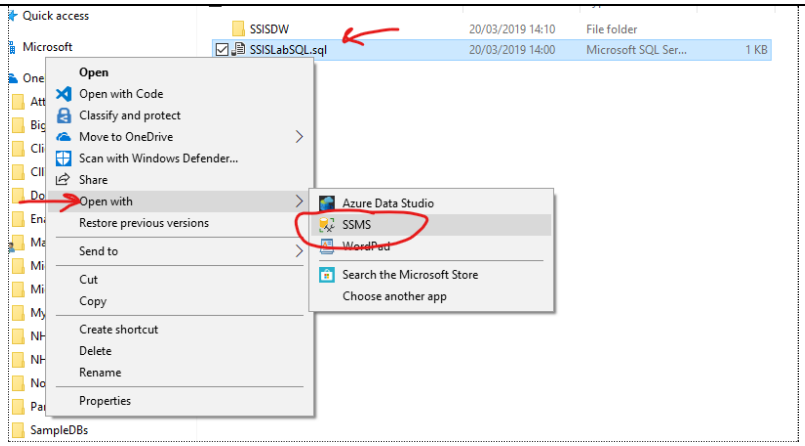
PROBLEM STATEMENT

Now the databases have been migrated to Azure, the Data Warehouse SSIS Packages also need to be migrated from on Premise to Azure SSIS Integration runtime.

LAB INSTARUCTIONS

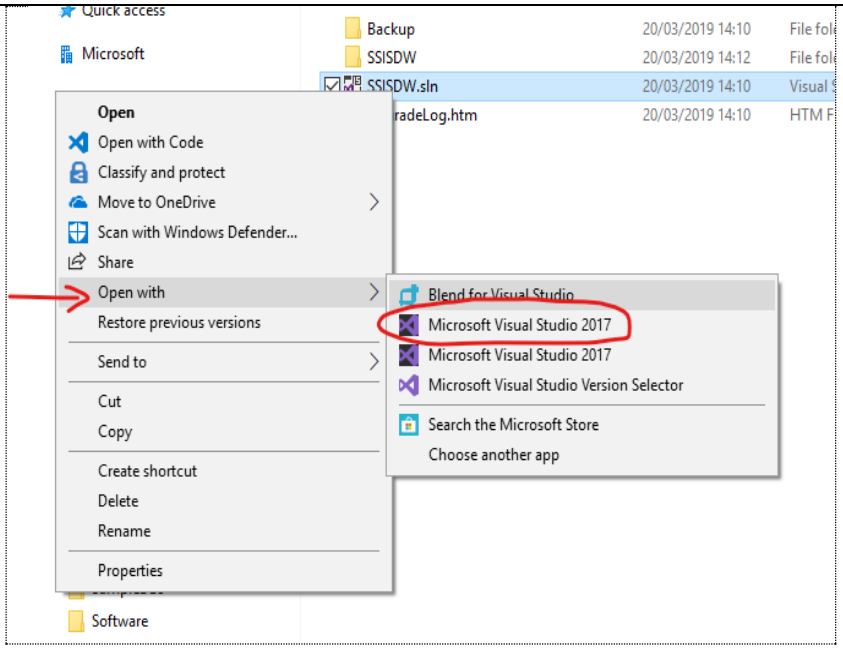
LAB Time 30 Mins

1. Open SQL Script with Management Studio C:_SQLHACK_\LABS\02-SSIS_Migration\SSISLabSQL.sql

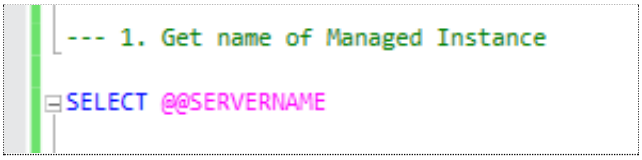
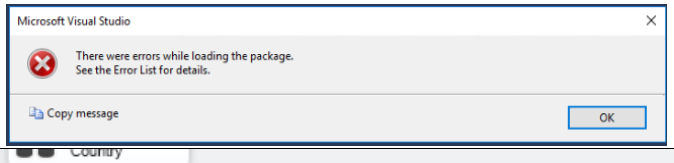
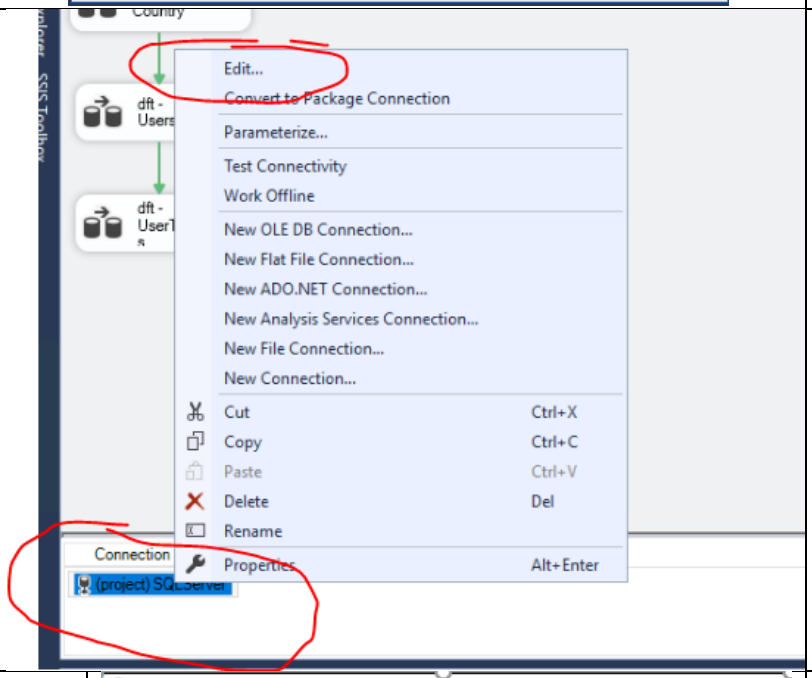
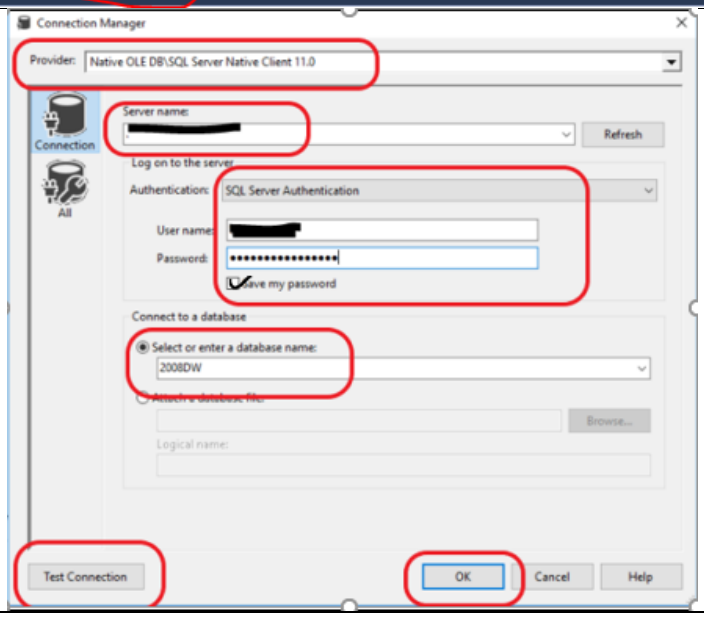
Narrative	Screenshot	Notes
Open SQL Scrip		***Tip**: Right Click SSISLabSQL,sql* • *Open with SSMS*

2. Open SSIS Package with Visual Studio 2017 C:_SQLHACK_\LABS\02-SSIS_Migration\SSISDW\SSISDW.sln

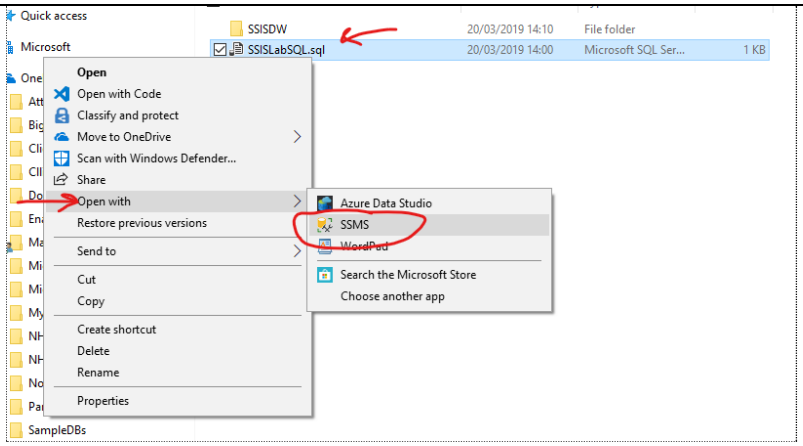
Narrative	Screenshot	Notes

Open SSIS Package using Visual Studio 2017		***Tip**: Right Click SSISDW.sln* • *Open with Visual Studio 2017*

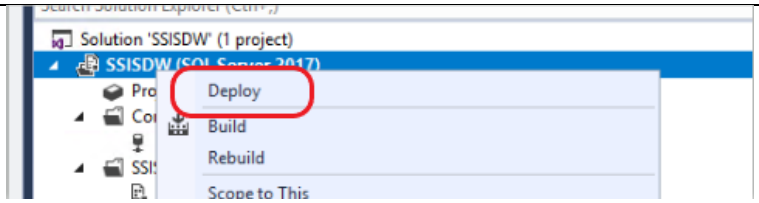
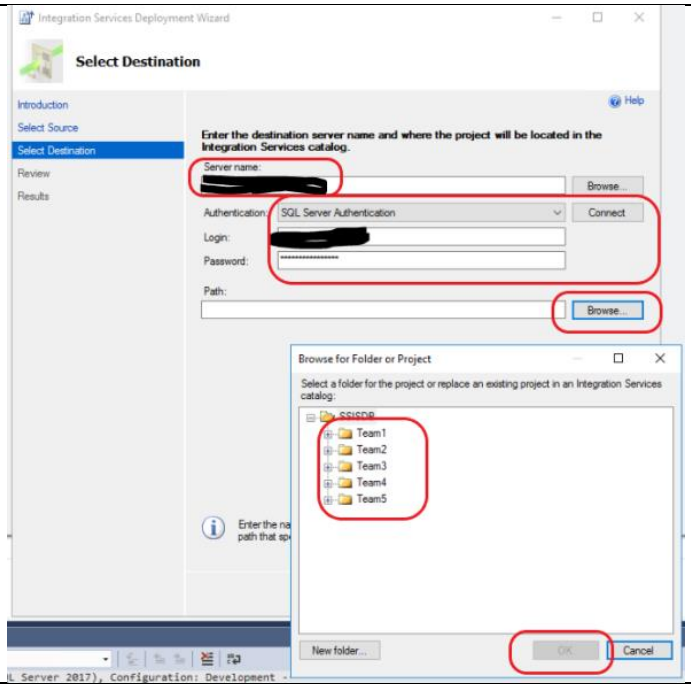
3. Using Visual Studio, check and validate connection string to Managed Instance of SSIS Package.

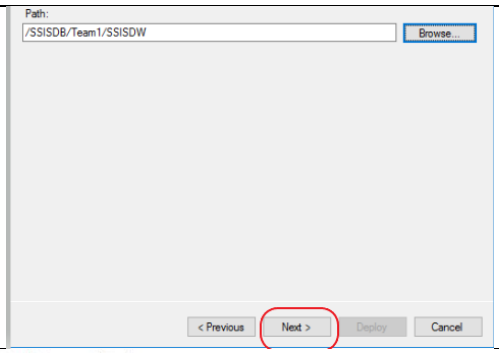
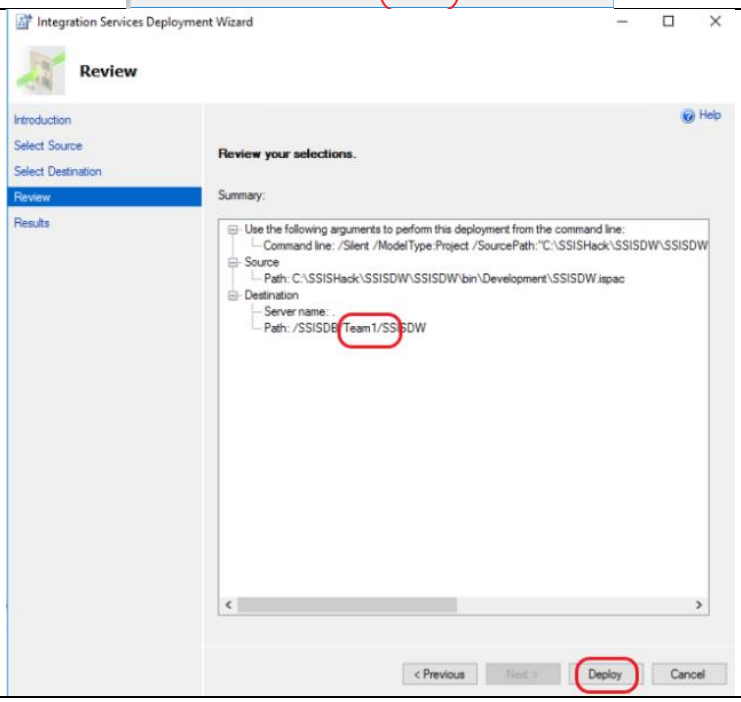
Narrative	Screenshot	Notes
		***Tip:*** *Execute part 3 of SQL script to get name of Managed Instance*
Click OK if you get error messages		We will be fixing the connection strings in the next step
Right Click connection called "(project)SQLServer"		
Check the following 1.Server Name = Full Managed Instance Name 2.Provider as per pic 3.Authentication as per pic with student Username and Password. ***Select save password*** 4.Database name – 2008DW 5.Click "Test Connection" 6.Click OK		

4. Using Visual Studio, build the solution

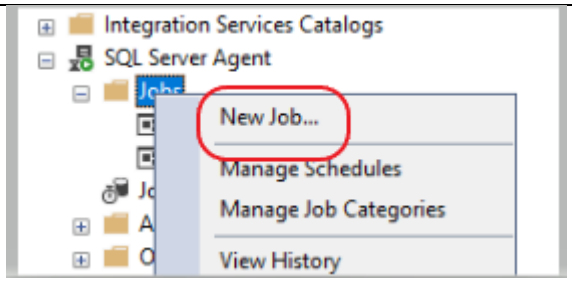
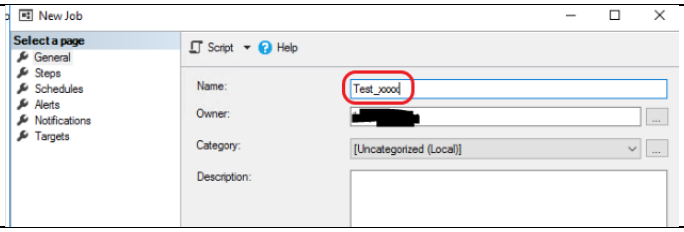
Narrative	Screenshot	Notes
1.Right Click Project 2.Click Build		*You must build a solution before you are allowed to deploy it*

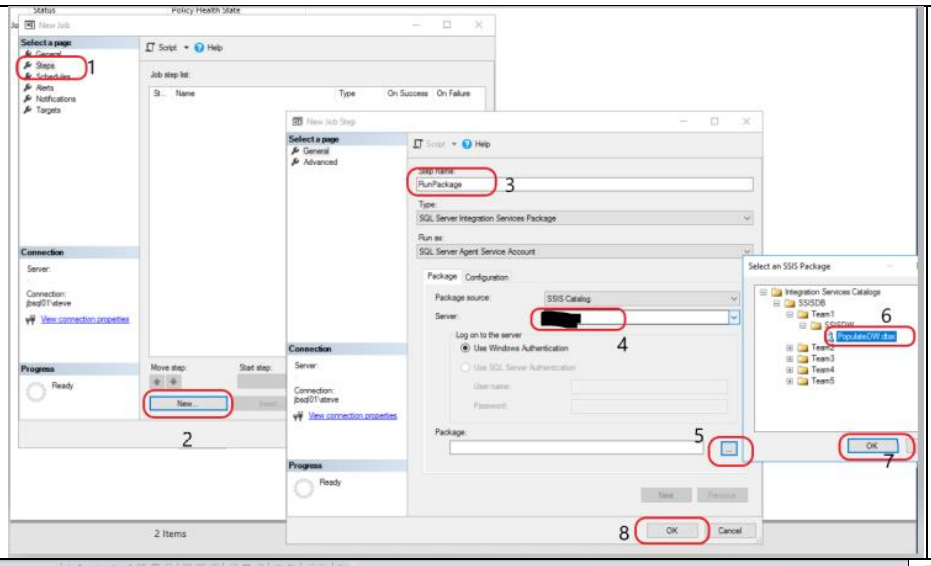
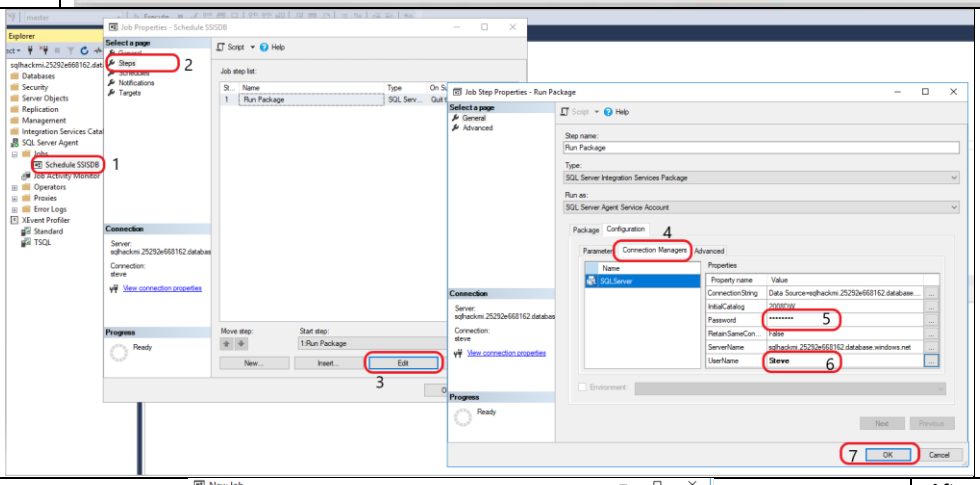
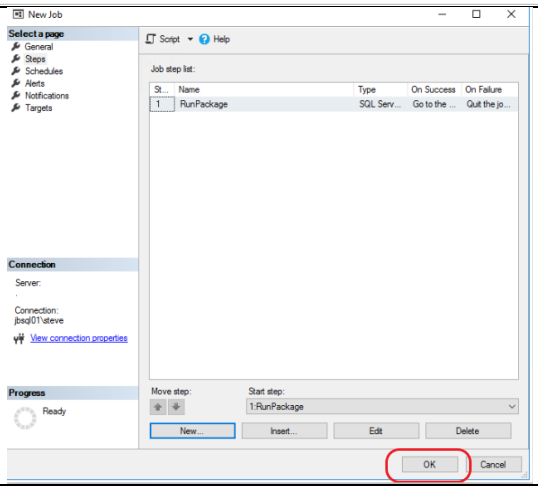
5. Using Visual Studio, deploy the solution to your folder under SSISDB

Narrative	Screenshot	Notes
1.Right Click Project 2.Click Deploy		
Fill in as per diag.. **1. Server Name** = Full Managed Instance Name 2.SQL Authentication + Username + Password 3.Click connect 4.Click Browse 5.Select ***correct*** team folder 6.Click OK		You will not see the project folder list until you authenticate. If the team folder does not exist…click New folder and createit

Click Next		
1.Check team name is correct 2.Click Deploy		Deployment might take a couple of minutes

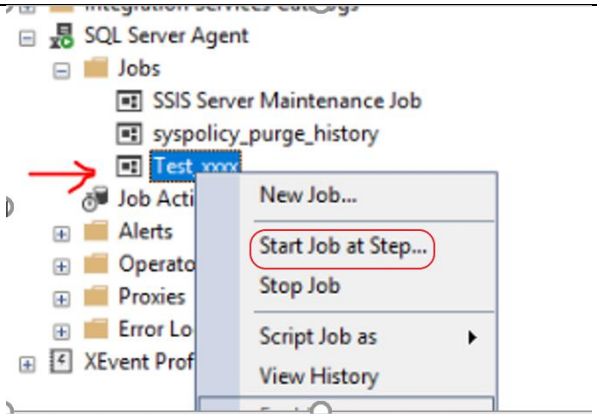
6. Use SSMS to set up SQL Job to schedule SSISPackage and execute it under your teams credentials

Narrative	Screenshot	Notes
Right click Jobs Click New Jobs		*Jobs are under SQL Server Agent*
Name = Test_'team name' Ie: Test_Team1		

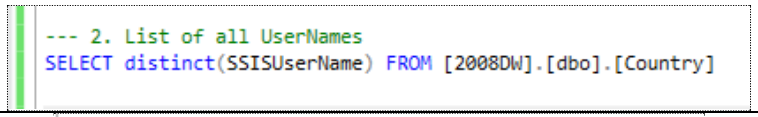
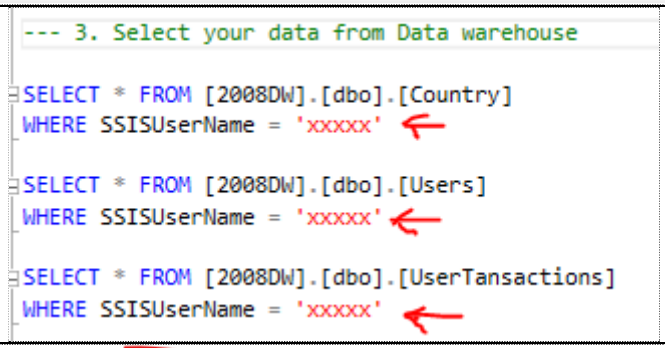
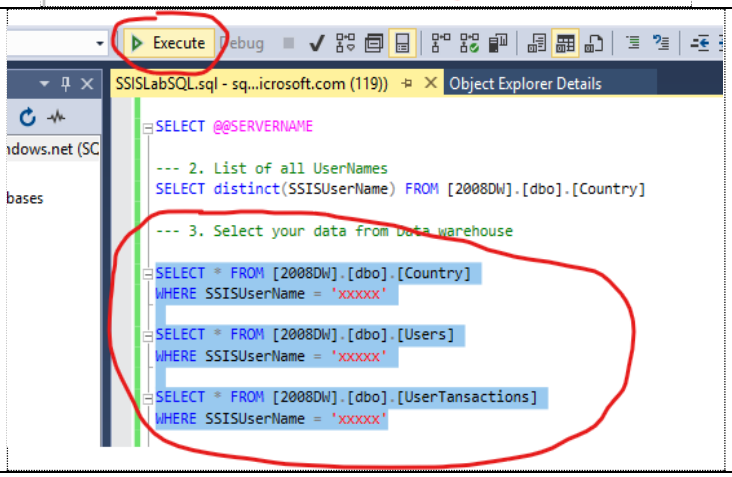
1.Click Steps 2.Click New 3.Step Name = "RunPackage" 4.Server = Full MI Name 5.Click browse button 6.Select package ***from correct team*** folder 7.OK 8.OK		
1.Double click the job you have just created 2.Select Steps 3.Click Edit 4.Select "Connection Manager" 5.Update UserName and Password to your 6.OK 7.OK		*This allows the package to execute under the team username*
Click OK to create job		*After this step the job will be create*

7. Use SSMS to manually Run Job

Narrative	Screenshot	Notes

1.Right click the job you have just created 2.Click “Start Job at Step…”		

8. Use SSMS to check data in Managed Instance

Narrative	Screenshot	Notes
1.Run this script--		This will give a list of all the username and the student can cut and paste it into the WHERE clause below
Update section 3 of script to is the students Team Name.		SELECT * FROM …. WHERE SSISUsername = 'Team12'
Highlight and execute section 3 of the script		Student should see their data There is a TimeStamp of when the row was created.