Hands-on Lab of Creating SSIS Package

This article is to show you how to start from zero to create a SSIS package that reads a CSV file and then do a lookup in a DB and finally insert the data into a table.

# Preparation

This section is to prepare an environment in Azure for this lab, including create a VM in Azure with SQL Server 2016. Then we prepare the dataset, and development tool, i.e., Visual Studio and SQL Server Data Tools.

## Create a VM with Microsoft SQL Server 2016 Enterprise Edition

Log into Azure Portal

Select this image: SQL Server 2016 SP1 Enterprise on Windows Server 2016

For Virtual Machine, select Standard D2s v3

Wait for 10 minutes till the VM is fully created.

Log into the VM through RDP

You will find

SSMS has been installed

SQL Server 2016 has been installed with a several services:

* Database Engine
* Analysis Servers
* Integration Services
* Reporting Services

## Database Preparation on the VM

Login into the VM through RDP

Open Internet Explorer (IE) and navigate to <https://github.com/Microsoft/sql-server-samples/releases/tag/adventureworks>

We are going to use the dataset in [**AdventureWorksDW2016\_EXT.bak**](https://github.com/Microsoft/sql-server-samples/releases/download/adventureworks/AdventureWorksDW2016_EXT.bak) .

Download it to the data disk F:\.

Log into SSMS

Databases->Restore Database->Device->File->Add the file AdventureWorksDW2016\_EXT.bak->Ok->Ok.

In a few minutes, we see database AdventureWorksDW2016\_EXT was restored on this SQL server.

## Lab dataset Preparation on the VM

Login into the VM through RDP

Open Internet Explorer (IE) and navigate to <https://archive.codeplex.com/?p=msftisprodsamples>

Click “download archive” button and save the file MSFTISProdSamples.zip for the data strip on F:\.

Unzip the file and navigate to F:\jackdata\MSFTISProdSamples\sourceCode\sourceCode\Katmai\_Trunk\Tutorial\Creating a Simple ETL Package\Sample Data

We see several \*.txt files that we will use in this lab.

## Visual Studio 2017 Community Version on the VM

Log into the VM through RDP

Open Internet Explorer (IE) and navigate to <https://www.visualstudio.com/downloads/>

Click “Free download” in Visual Studio Community 2017.

Install VS2017 by running the downloaded vs\_community\_\_931523988.1522616584.exe

## SQL Server Data Tools on VM

Log into the VM through RDP

Open Internet Explorer (IE) and navigate to <https://docs.microsoft.com/en-us/sql/ssdt/download-sql-server-data-tools-ssdt>

Click the link [Download SSDT for Visual Studio 2017 (15.5.2)](https://go.microsoft.com/fwlink/?LinkId=866452)

Install SSDT on the VM by running SSDT-Setup-ENU.exe

# Creating Basic SSIS Package

To create a SSIS package for ETL purpose, we mainly follow Lesson 1 and Lesson 4 in this tutorial: <https://docs.microsoft.com/en-us/sql/integration-services/ssis-how-to-create-an-etl-package>

## Creating a New Integration Services Project

Log into the VM through RDP

Start Visual Studio

On the **File** menu, point to **New**, and click **Project** to create a new Integration Services project.

In the **New Project** dialog box, expand the **Business Intelligence** node under **Installed Templates**, and select **Integration Services Project** in the **Templates** pane.

In the **Name** box, change the default name to **SSIS Tutorial**. Optionally, clear the **Create directory for solution** check box.

Click **OK**.

By default, an empty package, titled **Package.dtsx**, will be created and added to your project under SSIS Packages.

In **Solution Explorer** toolbar, right-click **Package.dtsx**, click **Rename**, and rename the default package to **Lesson 1.dtsx**.

## Adding and Configuring a Flat File Connection Manager

Right-click anywhere in the **Connection Managers** area, and then click **New Flat File Connection**.

In the **Flat File Connection Manager Editor** dialog box, for **Connection manager name**, type **Sample Flat File Source Data**.

Click **Browse**.

In the **Open** dialog box, locate the SampleCurrencyData.txt file on your machine.

Name columns as

|  |  |
| --- | --- |
| Column Name | Data Type |
| AverageRate | [DT\_R4] |
| CurrencyID | [DT\_WSTR] |
| CurrencyDate | [DT\_DBDATE] |
| EndOfDayRate | [DT\_R4] |

## Adding and Configuring an OLE DB Connection Manager

Right-click anywhere in the **Connection Managers** area, and then click **New OLE DB Connection**.

In the **Configure OLE DB Connection Manager** dialog box, click **New**.

For **Server name**, enter **localhost**.

When you specify localhost as the server name, the connection manager connects to the default instance of SQL Server on the local computer. To use a remote instance of SQL Server, replace localhost with the name of the server to which you want to connect.

In the **Log on to the server** group, verify that **Use Windows Authentication** is selected.

In the **Connect to a database** group, in the **Select or enter a database name** box, type or select **AdventureWorksDW2012**.

Click **Test Connection** to verify that the connection settings you have specified are valid.

Click **OK**.

Click **OK**.

In the **Data Connections** pane of the **Configure OLE DB Connection Manager** dialog box, verify that **localhost.AdventureWorksDW2016** is selected.

Click **OK**.

## Adding a Data Flow Task to the Package

Click the **Control Flow** tab.

In the **SSIS Toolbox**, expand **Favorites**, and drag a **Data Flow Task** onto the design surface of the **Control Flow** tab.

On the **Control Flow** design surface, right-click the newly added **Data Flow Task**, click **Rename**, and change the name to **Extract Sample Currency Data**.

## Adding and Configuring the Flat File Source

Open the **Data Flow** designer, either by double-clicking the **Extract Sample Currency Data** data flow task or by clicking the **Data Flow tab**.

In the **SSIS Toolbox**, expand **OtherSources**, and then drag a **Flat File Source** onto the design surface of the **Data Flow** tab.

On the **Data Flow** design surface, right-click the newly added **Flat File Source**, click **Rename**, and change the name to **Extract Sample Currency Data**.

Double-click the Flat File source to open the Flat File Source Editor dialog box.

In the **Flat file connection manager** box, select **Sample Flat File Source Data**.

Click **Columns** and verify that the names of the columns are correct.

Click **OK**.

## Adding and Configuring the Lookup Transformations

In the **SSIS Toolbox**, expand **Common**, and then drag **Lookup** onto the design surface of the **Data Flow** tab. Place Lookup directly below the **Extract Sample Currency Data** source.

Click the **Extract Sample Currency Data** flat file source and drag the green arrow onto the newly added **Lookup** transformation to connect the two components.

On the **Data Flow** design surface, click **Lookup** in the **Lookup** transformation, and change the name to **Lookup Currency Key**.

Double-click the **Lookup CurrencyKey** transformation to display the Lookup Transformation Editor.

On the **General** page, make the following selections:

Select **Full cache**.

In the **Connection type** area, select **OLE DB connection manager**.

On the **Connection** page, make the following selections:

In the **OLE DB connection manager** dialog box, ensure that **localhost.AdventureWorksDW2016** is displayed.

Select **Use results of an SQL query**, and then type or copy the following SQL statement:

SQL Copy

SELECT \* FROM [dbo].[DimCurrency]

WHERE [CurrencyAlternateKey]

IN ('ARS', 'AUD', 'BRL', 'CAD', 'CNY',

'DEM', 'EUR', 'FRF', 'GBP', 'JPY',

'MXN', 'SAR', 'USD', 'VEB')

On the **Columns** page, make the following selections:

In the **Available Input Columns** panel, drag **CurrencyID** to the **Available Lookup Columns** panel and drop it on **CurrencyAlternateKey**.

In the **Available Lookup Columns** list, select the check box to the left of **CurrencyKey**.

Click **OK** to return to the **Data Flow** design surface.

## Adding and Configuring the OLE DB Destination

In the **SSIS Toolbox**, expand **Other Destinations**, and drag **OLE DB Destination** onto the design surface of the **Data Flow** tab. Place the OLE DB destination directly below the **Lookup Date Key** transformation.

Click the **Lookup Date Key** transformation and drag the green arrow over to the newly added **OLE DB Destination** to connect the two components together.

In the **Input Output Selection** dialog box, in the **Output** list box, click **Lookup Match Output**, and then click **OK**.

On the **Data Flow** design surface, click **OLE DB Destination** in the newly added **OLE DB Destination** component, and change the name to **Sample OLE DB Destination**.

Double-click **Sample OLE DB Destination**.

In the **OLE DB Destination Editor** dialog box, ensure that **localhost.AdventureWorksDW2012** is selected in the **OLE DB Connection manager** box.

In the **Name of the table or the view** box, type or select **[dbo].[FactCurrencyRate]**.

Click the **New** button to create a new table. Change the name of the table in the script to read **NewFactCurrencyRate**. Click **OK**.

Upon clicking **OK**, the dialog will close and the **Name of the table or the view** will automatically change to **NewFactCurrencyRate**.

Click **Mappings**.

Verify that the **AverageRate**, **CurrencyKey**, **EndOfDayRate**, and **DateKey** input columns are mapped correctly to the destination columns. If same-named columns are mapped, the mapping is correct.

Click **OK**.

## Testing the Package So-far

On the **Debug** menu, click **Start Debugging**.

The package will run, resulting in 1097 rows successfully added into the **FactCurrency** fact table in **AdventureWorksDW2012**.

After the package has completed running, on the **Debug** menu, click **Stop Debugging**

You should see more than 1,000 rows having been inserted into table **NewFactCurrencyRate.**

# Enhancing the SSIS Package

In this section we will build up our package created in the previous section by adding a exception branch in it to write out the rows with unknown CurrencyID.

## Data File Preparation

Make a copy of file SampleCurrencyData.txt and then change the first two lines of this file to

1 XXX 7/1/2001 0:00 0.99980004

1 YYY 7/2/2001 0:00 1.000900811

The CurrencyID XXX and YYY are not in the lookup, and thus should be errored out.

## Adding Error Flow Redirection

In the **SSIS Toolbox**, expand **Common**, and then drag **Script Component** onto the design surface of the **Data Flow** tab. Place **Script** to the right of the **Lookup Currency Key** transformation.

In the **Select Script Component Type** dialog box, click **Transformation**, and click **OK**.

Click the **Lookup Currency Key** transformation and then drag the red arrow onto the newly added **Script** transformation to connect the two components.

The red arrow represents the error output of the **Lookup Currency Key** transformation. By using the red arrow to connect the transformation to the Script component, you can redirect any processing errors to the Script component, which then processes the errors and sends them to the destination.

In the **Configure Error Output** dialog box, in the **Error** column, select **Redirect row**, and then click **OK**.

On the **Data Flow** design surface, click **Script Component** in the newly added **ScriptComponent**, and change the name to **Get Error Description**.

Double-click the **Get Error Description** transformation.

In the **Script Transformation Editor** dialog box, on the **Input Columns** page, select the **ErrorCode** column.

On the **Inputs and Outputs** page, expand **Output 0**, click **Output Columns**, and then click **Add Column**.

In the **Name** property, type **ErrorDescription** and set the **DataType** property to **Unicode string [DT\_WSTR]**.

On the **Script** page, verify that the **LocaleID** property is set to **English (United States.**

Click **Edit Script** to open Microsoft Visual Studio Tools for Applications (VSTA). In the **Input0\_ProcessInputRow** method, type or paste the following code.

Row.ErrorDescription = this.ComponentMetaData.GetErrorDescription(Row.ErrorCode);

On the **Build** menu, click **Build Solution** to build the script and save your changes, and then close VSTA.

Click **OK** to close the **Script Transformation Editor** dialog box.

## Adding a Flat File Destination for Errors

Click the **Data Flow** tab.

In the **SSIS Toolbox**, expand **Other**, and drag **Flat File Destination** onto the data flow design surface. Put the **Flat File Destination** directly underneath the **Get Error Description** transformation.

Click the **Get Error Description** transformation, and then drag the green arrow onto the new **Flat File Destination**.

On the **Data Flow** design surface, click **Flat File Destination** in the newly added **Flat File Destination** transformation, and change the name to **Failed Rows**.

Right-click the **Failed Rows** transformation, click **Edit**, and then in the **Flat File Destination Editor**, click **New**.

In the **Flat File Format** dialog box, verify that **Delimited** is selected, and then click **OK**.

In the **Flat File Connection Manager Editor**, in the **Connection Manager Name** box type **Error Data**.

In the **Flat File Connection Manager Editor** dialog box, click **Browse**, and locate the folder in which to store the file.

In the **Open** dialog box, for **File name**, type **ErrorOutput.txt**, and then click **Open**.

In the **Flat File Connection Manager Editor** dialog box, verify that the **Locale** box contains English (United States) and **Code page** contains 1252 (ANSI -Latin I).

In the options pane, click **Columns**.

Click **OK**.

In the **Flat File Destination Editor**, clear the **Overwrite data in the file** check box.

In the **Flat File Destination Editor**, click **Mappings** to verify that all the columns are correct. Optionally, you can rename the columns in the destination.

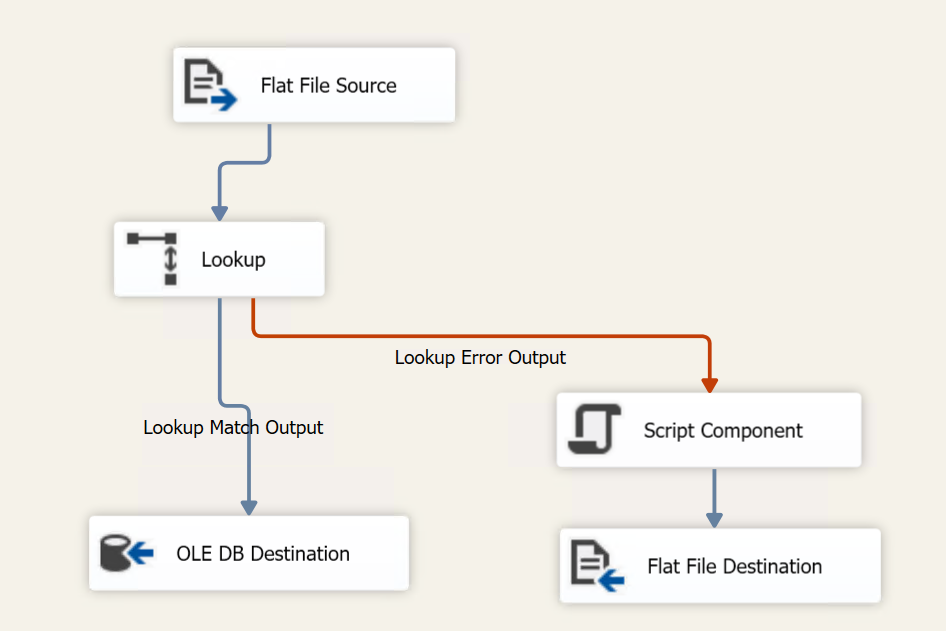
Click **OK**.

## Testing the Package again

On the **Debug** menu, click **Start Debugging**.

After the package has completed running, on the **Debug** menu, click **Stop Debugging**.

The Data Flow should look like this:



The error file should have lines like this:

1,YYY,2001-07-02,1.0009009,-1071607778,0,Row yielded no match during lookup.