\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



**Title: eCoaching\_Quality\_Load**

**SSIS Detail Design Document**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |  |
| --- | --- | --- |
| Last Revision | Last Review | Description |
| 09/08/2014 |  | Initial revision |

Prepared by: Susmitha Palacherla Date: 05/01/2014

Department, Location: Health Solutions Division

Approved by: Date:

**Change History Log**

| **Date** | **Version** | **Change Description** | **Author** |
| --- | --- | --- | --- |
| 05/01/2014 | 1.0 | Initial revision | Susmitha Palacherla |
| 07/01/2014 | 2.0 | Updated per SCR 12963 to fix Loaded Count value in File List table | Susmitha Palacherla |
| 07/20/2014 | 3.0 | Updated per SCR 13054 to import Verint Form Name | Susmitha Palacherla |
| 09/08/2014 | 4.0 | Updated to add a variable for File Prefix to support loads into dev and test using a different file name | Susmitha Palacherla |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Contents

[1. Overview 4](#_Toc391973376)

[1.1 Project Description 4](#_Toc391973377)

[1.2 Document Scope 4](#_Toc391973378)

[1.3 Source File 4](#_Toc391973379)

[1.4 Module List 4](#_Toc391973380)

[1.5 Software and Hardware Interfaces 4](#_Toc391973381)

[1.5.1 Software 4](#_Toc391973382)

[1.5.2 Hardware 4](#_Toc391973383)

[1.6 Users 5](#_Toc391973384)

[2. Details 5](#_Toc391973385)

[2.1 Source Files 5](#_Toc391973388)

[2.1.1 eCL\_IQS\_Scorecard\_yyyymmdd.csv.zip.encrypt 5](#_Toc391973389)

[2.2 Module Details 5](#_Toc391973390)

[2.2.1 SQL agent job 5](#_Toc391973391)

[2.2.2 SSIS Package 6](#_Toc391973392)

[2.2.3 Tables 49](#_Toc391973393)

[2.2.4 Procedures 50](#_Toc391973394)

# Overview

## Project Description

Quality is an important metric that contributes to a CCO CSR’s overall Performance. Quality monitoring can be performed internally by Vangent/GDIT staff or external evaluators using an internal application like Lime Survey or Verint Quality Monitoring tools. Irrespective of the source system in which the Quality evaluations are created, the records are fed into the eCoaching database with an overall classification of opportunity or Reinforcement. Once entered into the eCoaching database Reinforcements are simply acknowledged by CSRs and their Supervisors while Opportunities trigger a Coaching and a Coaching workflow is initiated.

## Document Scope

This document describes the load of the Quality feed sourced from all of the source systems in use at the time. Quality evaluations from all sources are consolidated into the IQS system from which a daily feed is generated for loading into the Coaching database.

## Source File

A daily file is generated from IQS and staged in an encrypted form on a file share.

Files have a naming convention eCL\_IQS\_Scorecard\_yyyymmdd.csv.zip.encrypt (prod) or Test\_eCL\_IQS\_Scorecard\_yyyymmdd.csv.zip.encrypt (Test)

and staged at [\\vrivscors01\BCC Scorecards\Coaching\Apps\Encryption\Encrypt\_out\](file:///\\vrivscors01\BCC%20Scorecards\Coaching\Apps\Encryption\Encrypt_out\)

## Module List

* SQL agent job
* SSIS Package
* Tables
* Procedures

## Software and Hardware Interfaces

### Software

* SQL Server 2008 R2 SP1 Suite
* Encryption/Decryption utility

### Hardware

* VRIVFSSDBT02\SCORD01,1437 – Dev DB Instance
* VRIVFSSDBT02\SCORT01,1438 – Test DB Instance
* VDENSSDBP07\SCORP01 – Prod DB Instance
* [\\VRIVSCORS01\BCC Scorecards\](file:///\\VRIVSCORS01\BCC%20Scorecards\) - File staging share

## Users

* CCO CSRs, Supervisors, Managers and support staff.

# Details



## Source Files

### eCL\_IQS\_Scorecard\_yyyymmdd.csv.zip.encrypt

* Description: A daily file containing all active evaluations from the previous day is staged on file staging server. This file is loaded into the eCoaching database and depending on whether the evaluation is an opportunity or Reinforcement a Coaching is initiated or simply acknowledged. The source file is in an encrypted form and decrypted during the load process.
* Source system: IQS
* Staging location: [\\vrivscors01\BCC Scorecards\Coaching\Apps\Encryption\Encrypt\_out\](file:///\\vrivscors01\BCC%20Scorecards\Coaching\Apps\Encryption\Encrypt_out\)
* File name eCL\_IQS\_Scorecard\_yyyymmdd.csv.zip.encrypt
* Frequency: Daily
* File arrival time: ~4:00 AM EST
* Destination Table: Coaching\_Log and Coaching\_Log\_Reason

## Module Details

### SQL agent job

* Production Instance : VDENSSDBP07\SCORP01
* Production Job: NewDailyCoaching
* Production Package: [\\VDENSSDBP07\scorecard-ssis\Coaching\IQS\_Coaching.dtsx](file:///\\VDENSSDBP07\scorecard-ssis\Coaching\IQS_Coaching.dtsx)
* Production Config File: Prod\_IQS\_Coaching.dtsConfig
* Owner: jobs\_admin
* Run As: BccScrdSQLAgent
* Schedule: Daily 6:30 AM EST Daily
* Destination File Location: N/A

### SSIS Package

#### Variables

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Scope** | **DataType** | **Value** |
| Decrypt\_In | IQS\_Coaching | String | \\vrivscors01\BCC Scorecards\Coaching\Apps\Encryption\Decrypt\_in\ |
| Encrypt\_In | IQS\_Coaching | String | \\vrivscors01\BCC Scorecards\Coaching\Apps\Encryption\Encrypt\_in\ |
| Decrypt\_Out | IQS\_Coaching | String | \\vrivscors01\BCC Scorecards\Coaching\Apps\Encryption\Decrypt\_out\ |
| Encrypt\_Out | IQS\_Coaching | String | \\vrivscors01\BCC Scorecards\Coaching\Apps\Encryption\Encrypt\_out\ |
| FileName | IQS\_Coaching | String | @[User::FilePrefix] +  (DT\_STR,4,1252) DatePart("yyyy",getdate())+  Right("0" + (DT\_STR,4,1252) DatePart("m",getdate()),2) +  Right("0" + (DT\_STR,4,1252) DatePart("d",getdate()),2)+".csv" |
| SystemName | IQS\_Coaching | String | Dev |
| LoadedCount | IQS\_Coaching | Int32 | 0 |
| RejectedCount | IQS\_Coaching | Int32 | 0 |
| StagedCount | IQS\_Coaching | Int32 | 0 |
| FilePrefix | IQS\_Coaching | String | Test\_eCL\_IQS\_Scorecard\_(Dev/test) and eCL\_IQS\_Scorecard\_(Prod) |

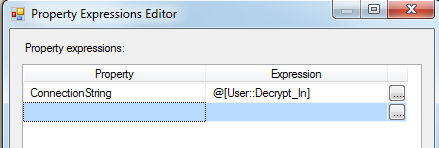
#### Connection Manager Entries

ConnectionStrings

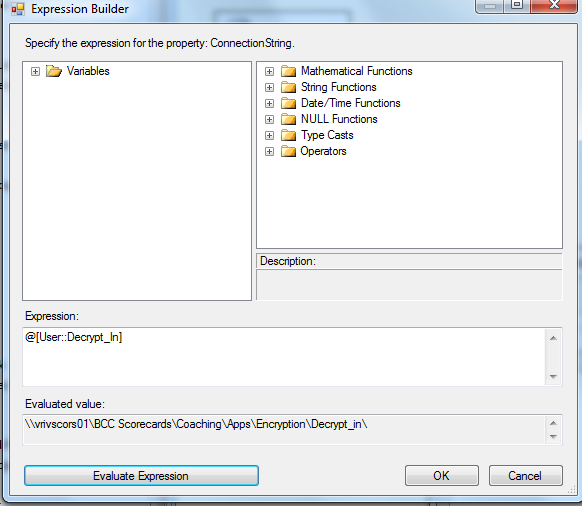
1. Decrypt\_In
2. Encrypt\_out
3. IQS\_File
4. Destinationdb
5. Decrypt\_In

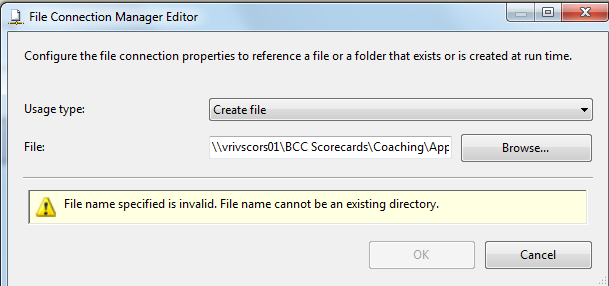
ConnectionString: \\vrivscors01\BCC Scorecards\Coaching\Apps\Encryption\Decrypt\_in\

Expression

: 

Expression Evaluates to:

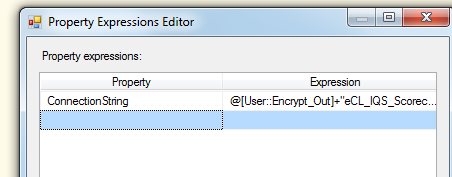


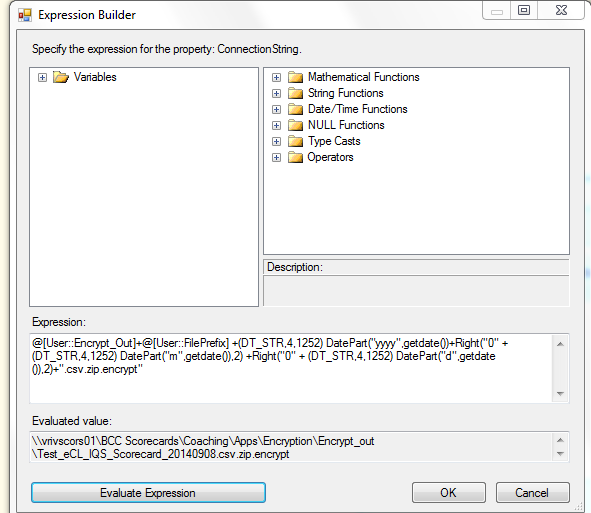


1. Encrypt\_out

ConnectionString: [\\vrivscors01\BCC Scorecards\Coaching\Apps\Encryption\Encrypt\_out\eCL\_IQS\_Scorecard\_20140315.csv.zip.encrypt](file:///\\vrivscors01\BCC%20Scorecards\Coaching\Apps\Encryption\Encrypt_out\eCL_IQS_Scorecard_20140315.csv.zip.encrypt)

Expression

\

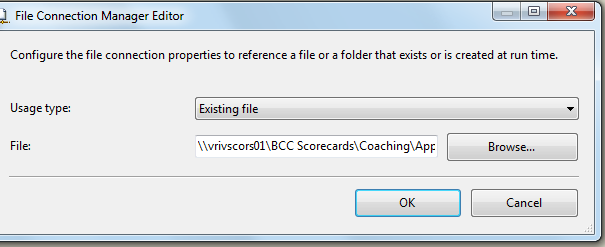


@[User::Encrypt\_Out]+@[User::FilePrefix] +

(DT\_STR,4,1252) DatePart("yyyy",getdate())+

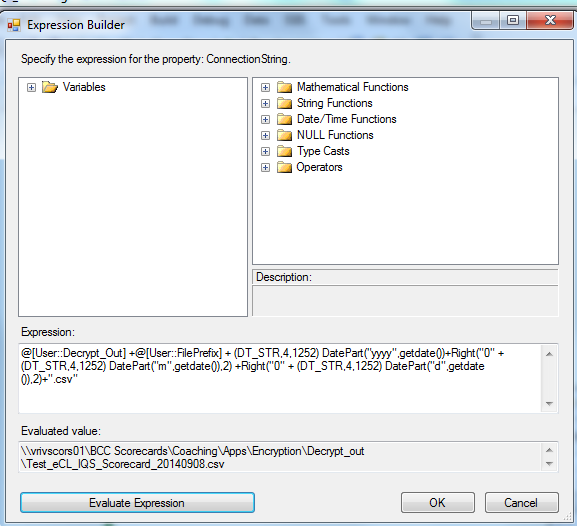
Right("0" + (DT\_STR,4,1252) DatePart("m",getdate()),2) +

Right("0" + (DT\_STR,4,1252) DatePart("d",getdate()),2)+".csv.zip.encrypt"



1. IQS\_File

Expression:

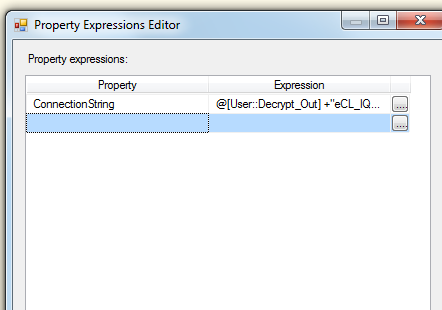


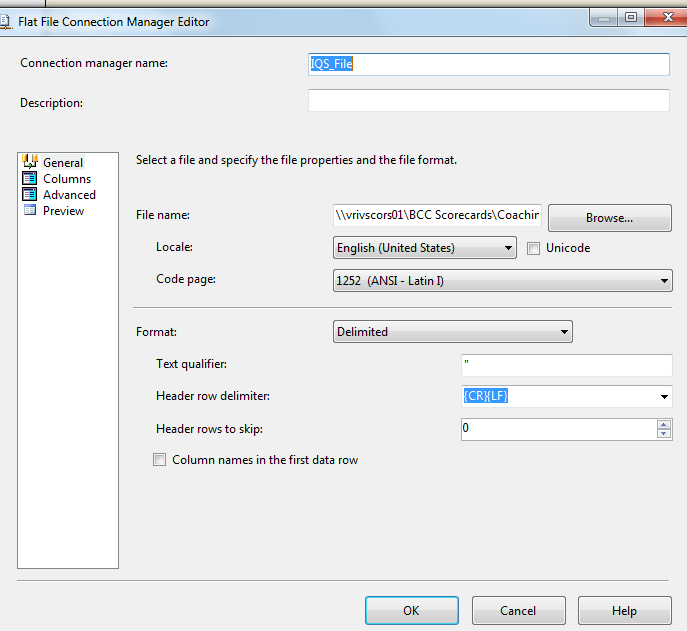
@[User::Decrypt\_Out] +@[User::FilePrefix] +

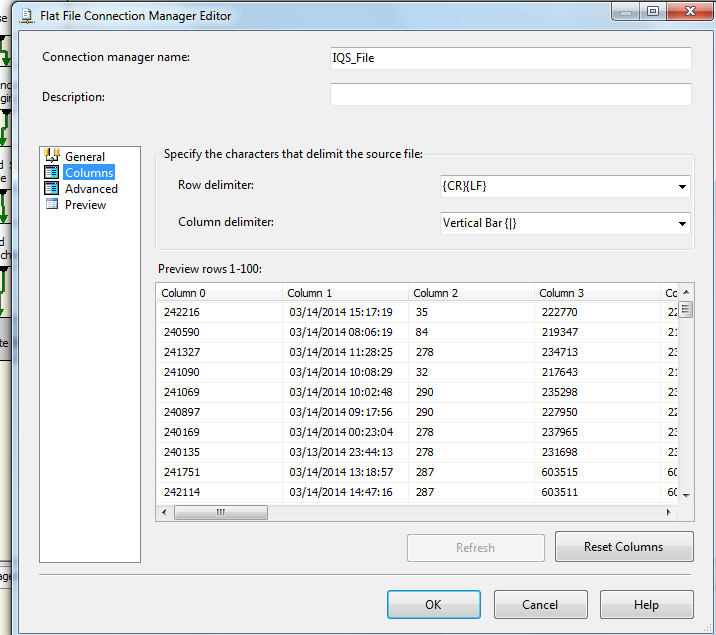
(DT\_STR,4,1252) DatePart("yyyy",getdate())+

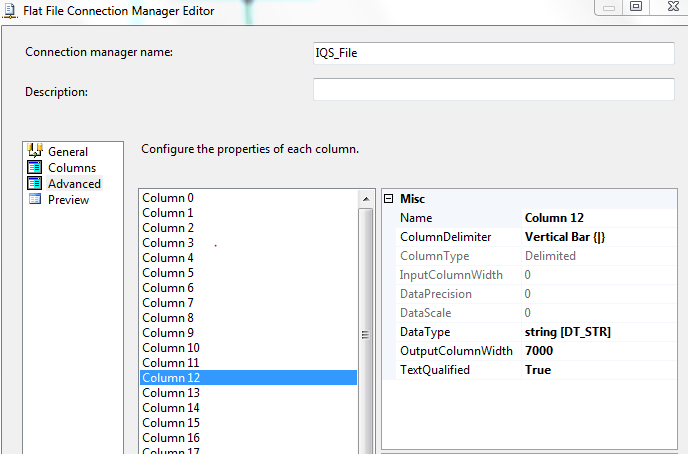
Right("0" + (DT\_STR,4,1252) DatePart("m",getdate()),2) +

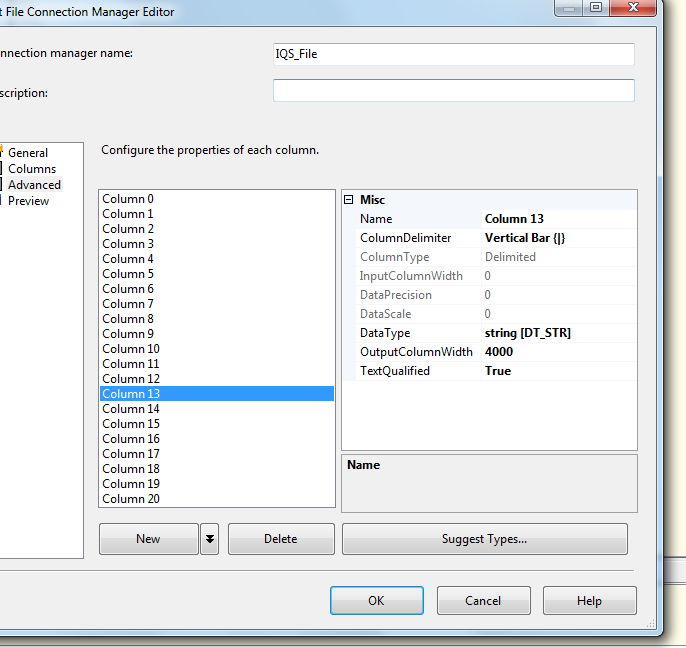
Right("0" + (DT\_STR,4,1252) DatePart("d",getdate()),2)+".csv"









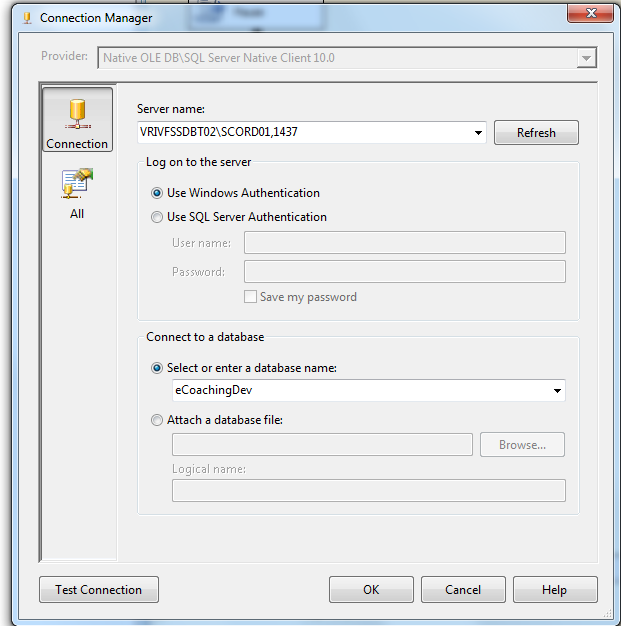


1. destinationdb

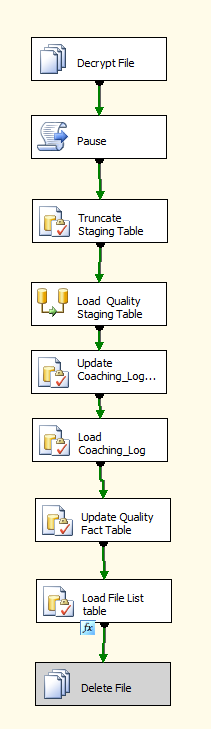
dev - VRIVFSSDBT02\SCORD01,1437 DB – eCoachingdev

Test - VRIVFSSDBT02\SCORT01,1438 DB – eCoachingtest

Prod - VDENSSDBP07\SCORP01 DB - eCoaching



#### Package Content



**Workflow**

**Summary of Steps**

1.Decrypt File in Encrypt\_out and place with that days date in decrypt\_out folder

2. Pause for 90 seconds.

3. Truncate staging table

4. Load staging table

5. Update Coaching Log and Coaching Log Reason tables

6. Load coaching log and Coaching log reason table table

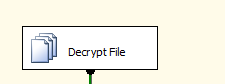
7. Update Quality Fact Table

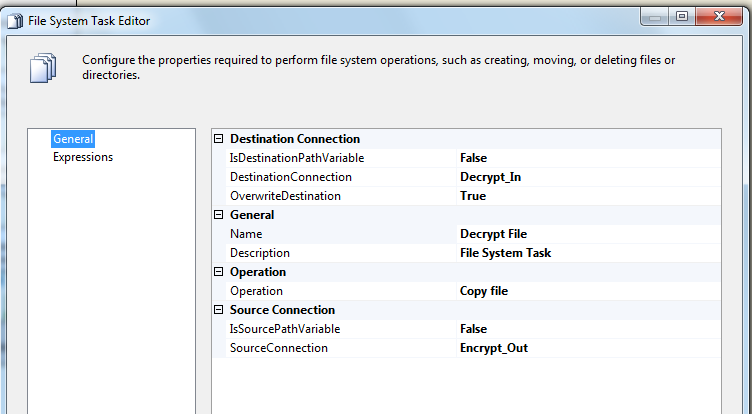
8. Load Quality File List Table

9. Delete file (currently disabled).

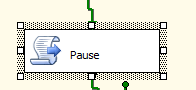
**Details**

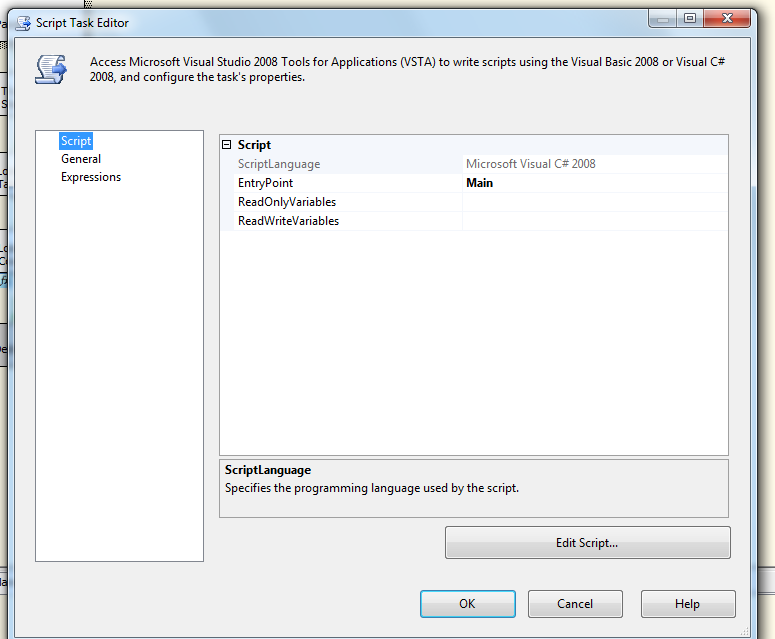
**Step1:** File System task: Decrypt file

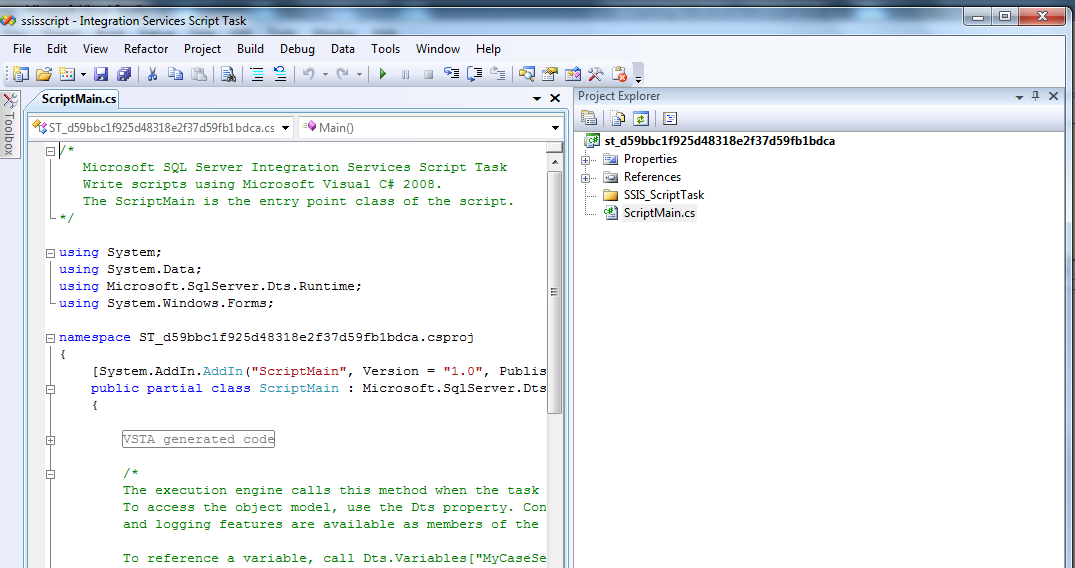




**Step2**: Script Task: Pause (to pause 90 seconds after the file decrypt. This ensures that the file decrypts completely before moving onto the next steps of loading the staging table.







Script:

/\*

Microsoft SQL Server Integration Services Script Task

Write scripts using Microsoft Visual C# 2008.

The ScriptMain is the entry point class of the script.

\*/

using System;

using System.Data;

using Microsoft.SqlServer.Dts.Runtime;

using System.Windows.Forms;

namespace ST\_d59bbc1f925d48318e2f37d59fb1bdca.csproj

{

[System.AddIn.AddIn("ScriptMain", Version = "1.0", Publisher = "", Description = "")]

public partial class ScriptMain : Microsoft.SqlServer.Dts.Tasks.ScriptTask.VSTARTScriptObjectModelBase

{

#region VSTA generated code

enum ScriptResults

{

Success = Microsoft.SqlServer.Dts.Runtime.DTSExecResult.Success,

Failure = Microsoft.SqlServer.Dts.Runtime.DTSExecResult.Failure

};

#endregion

/\*

The execution engine calls this method when the task executes.

To access the object model, use the Dts property. Connections, variables, events,

and logging features are available as members of the Dts property as shown in the following examples.

To reference a variable, call Dts.Variables["MyCaseSensitiveVariableName"].Value;

To post a log entry, call Dts.Log("This is my log text", 999, null);

To fire an event, call Dts.Events.FireInformation(99, "test", "hit the help message", "", 0, true);

To use the connections collection use something like the following:

ConnectionManager cm = Dts.Connections.Add("OLEDB");

cm.ConnectionString = "Data Source=localhost;Initial Catalog=AdventureWorks;Provider=SQLNCLI10;Integrated Security=SSPI;Auto Translate=False;";

Before returning from this method, set the value of Dts.TaskResult to indicate success or failure.

To open Help, press F1.

\*/

public void Main()

{

// Sleep for 90 seconds (\*1000)

System.Threading.Thread.Sleep(90000);

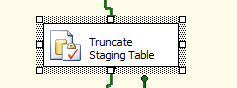
Dts.TaskResult = (int)ScriptResults.Success;

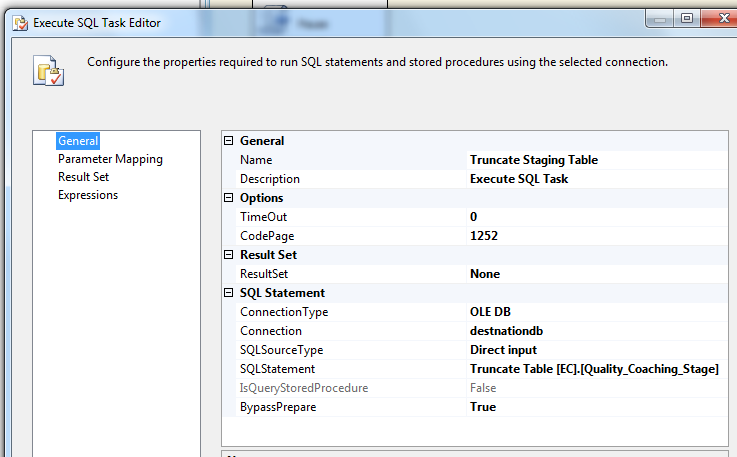
}

}

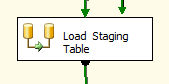
}

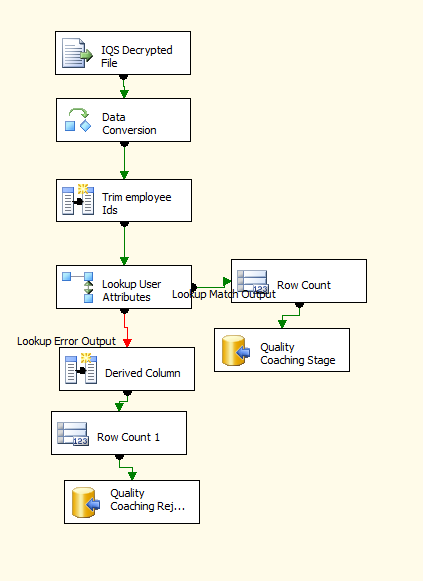
**Step 3: SQL task -** Truncate staging table (Quality\_coaching\_Stage)





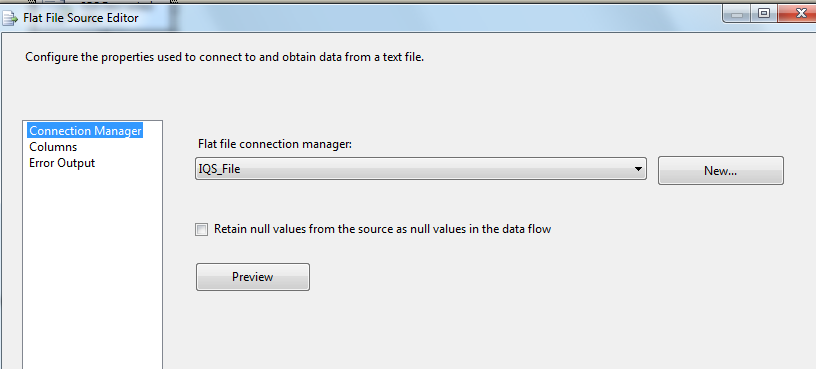
**Step 4: DFT –** Load IQS\_Coaching\_Stage table

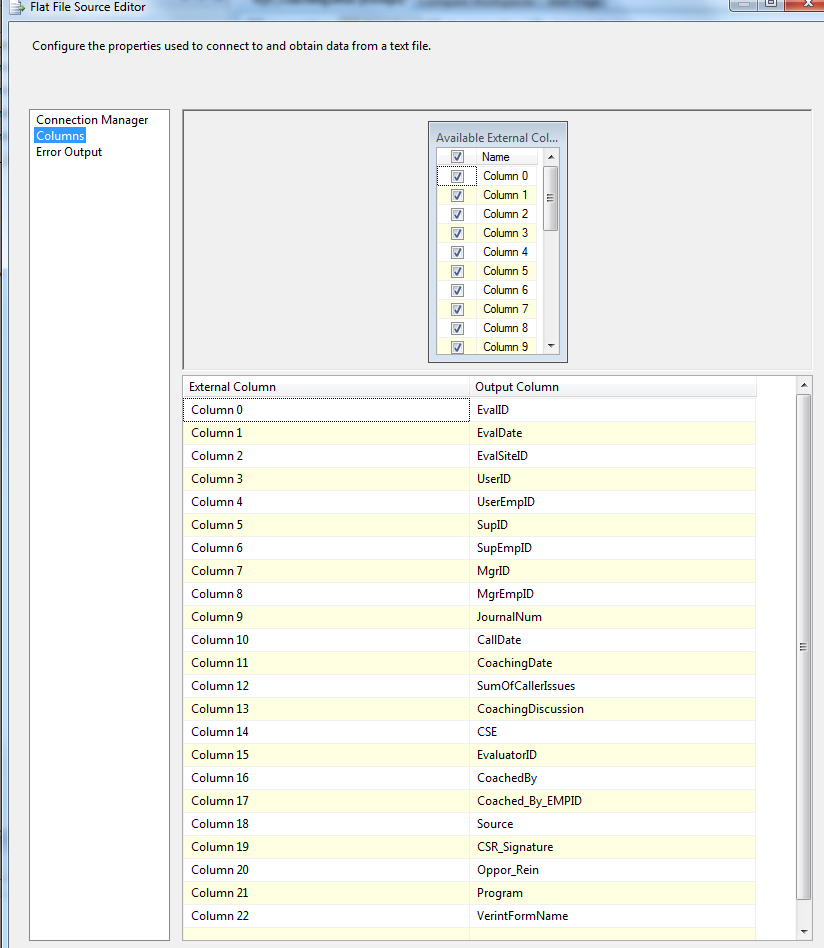


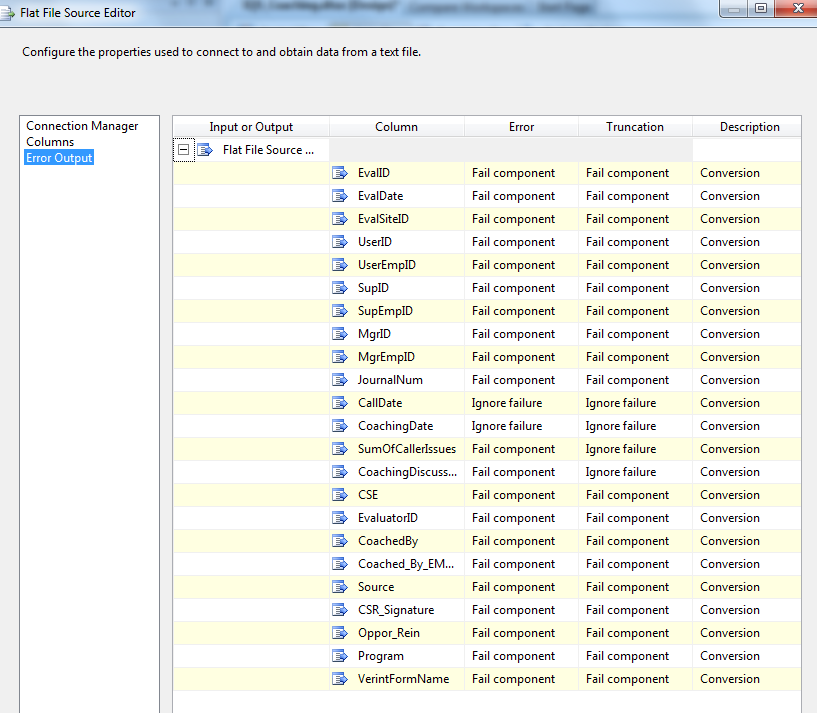


Process flow

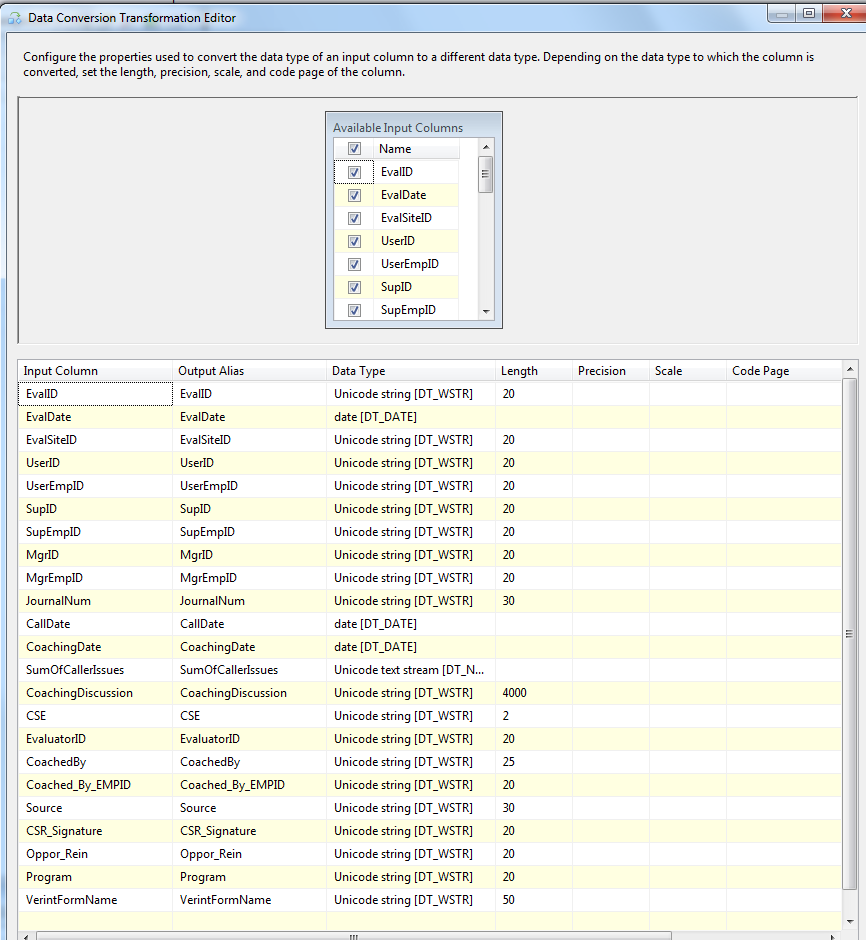
1. Source data
2. Data conversion on string values
3. Derived column to trim employee ids and get system date for date inserted
4. Lookup transformation to lookup Employee lanid based on Employee id
5. Load lookup match output to Coaching\_log table
6. Load lookup error output to IQS Rejected table
   1. Source data file



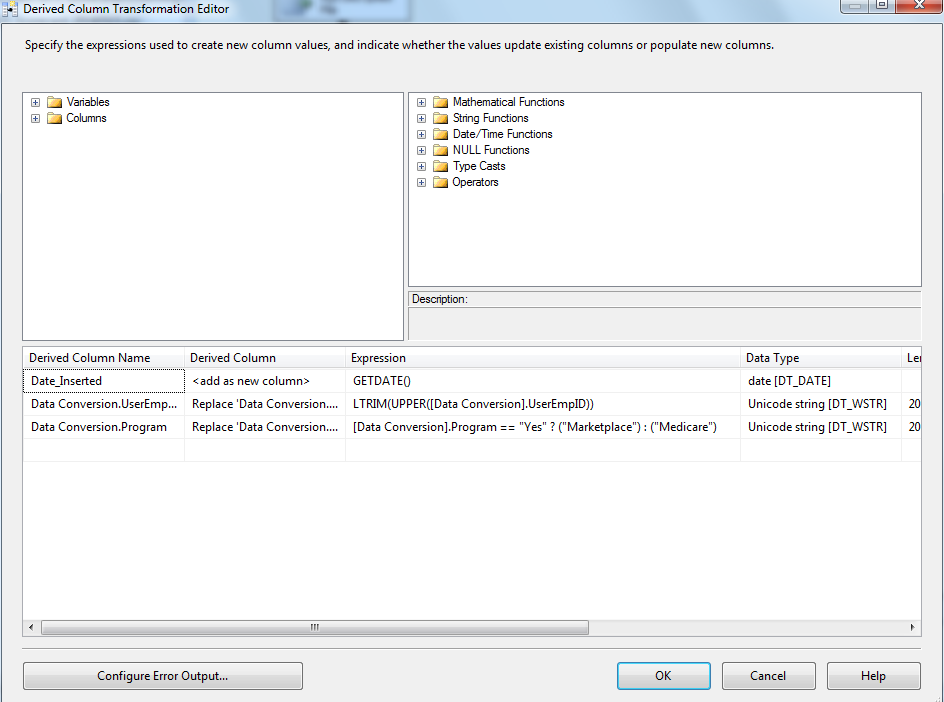




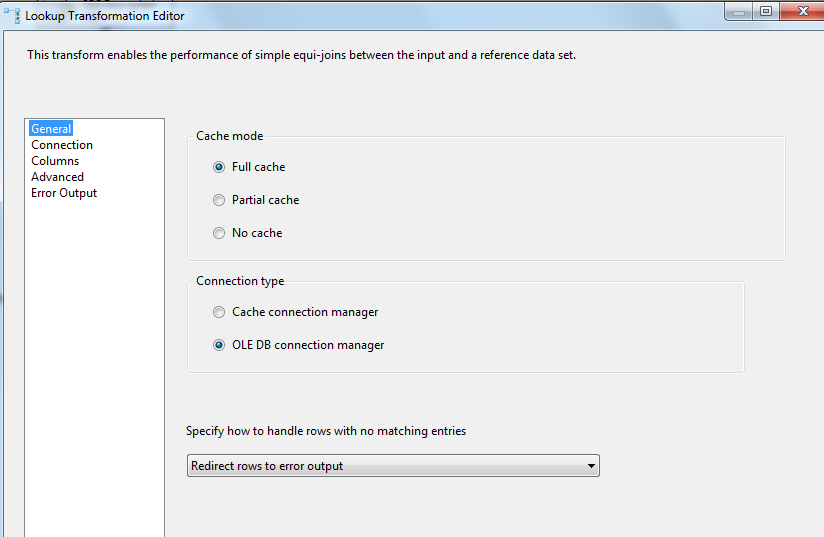
* 1. Data conversion of string values

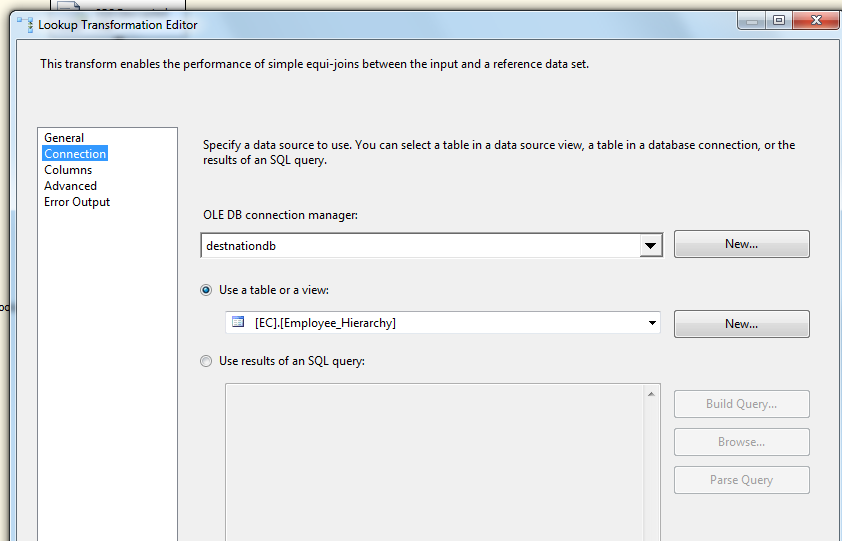


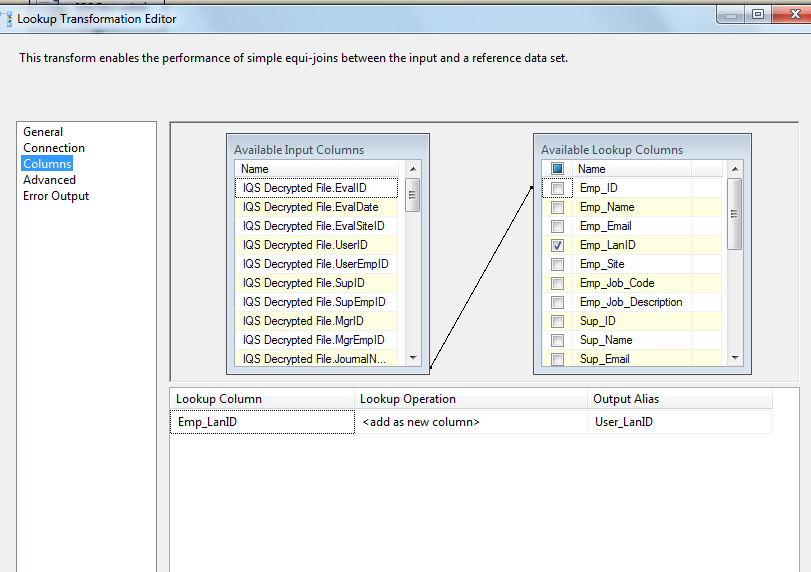
* 1. Derived columns for employee id trim and getdate()

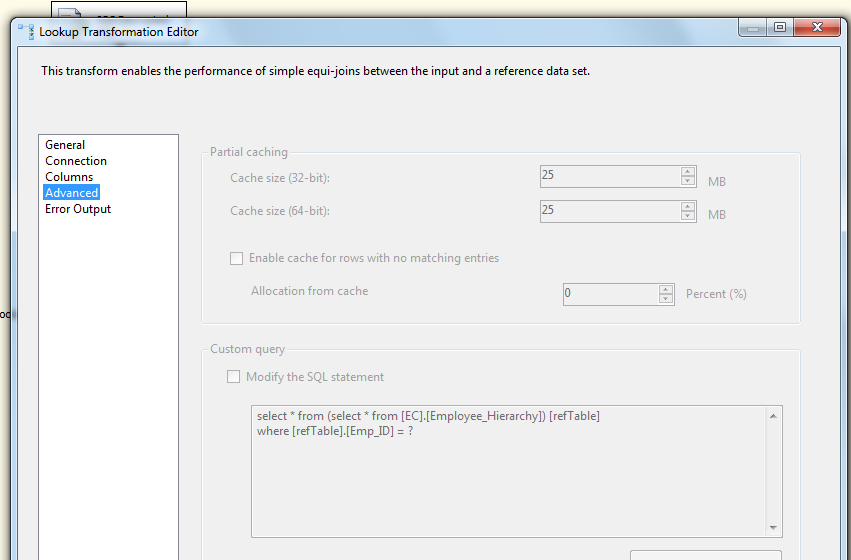


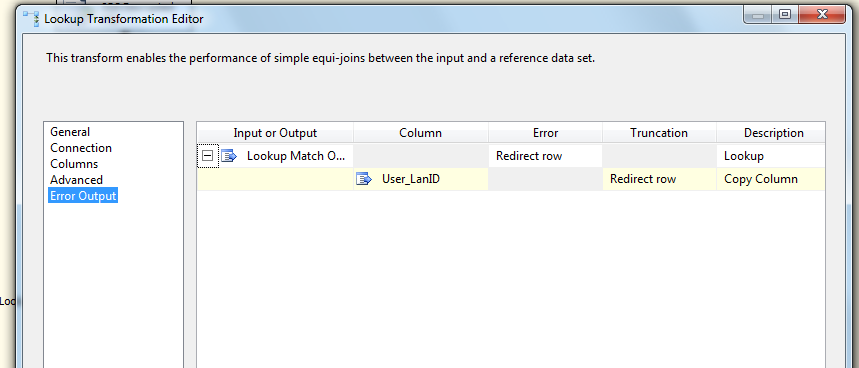
* 1. Lookup transformation to lookup Employee lanid based on Employee id

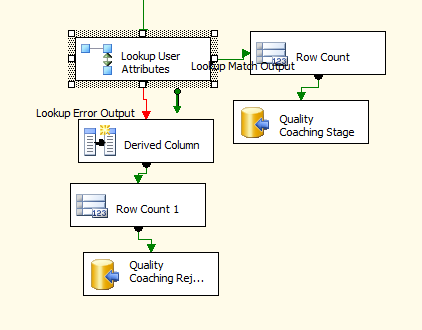


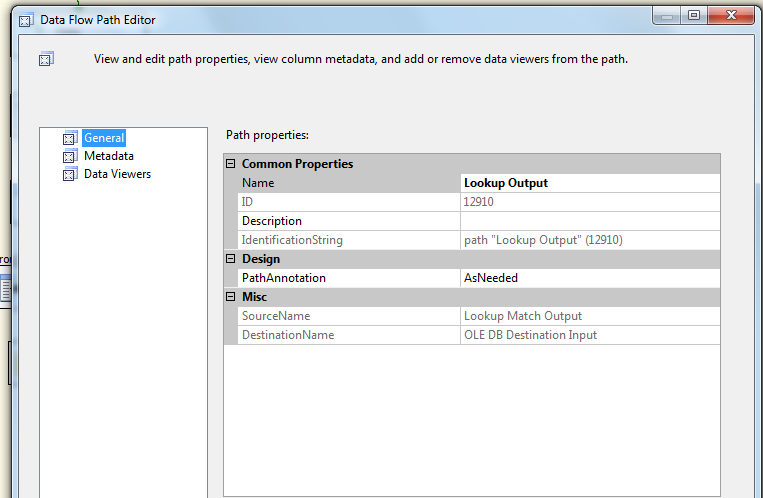


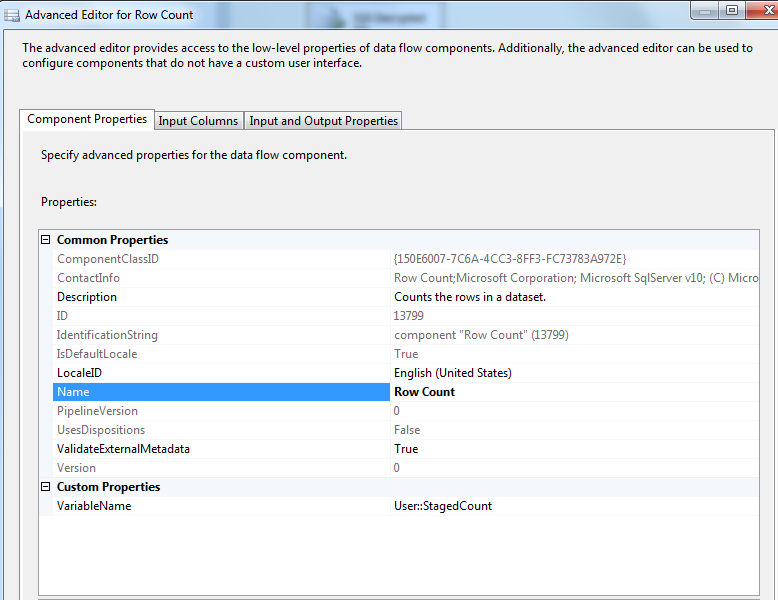


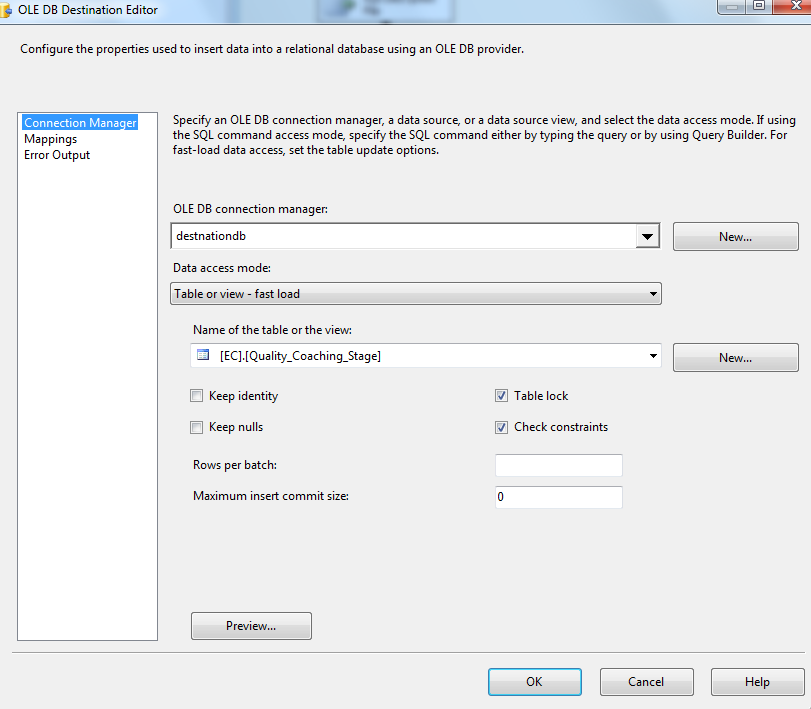


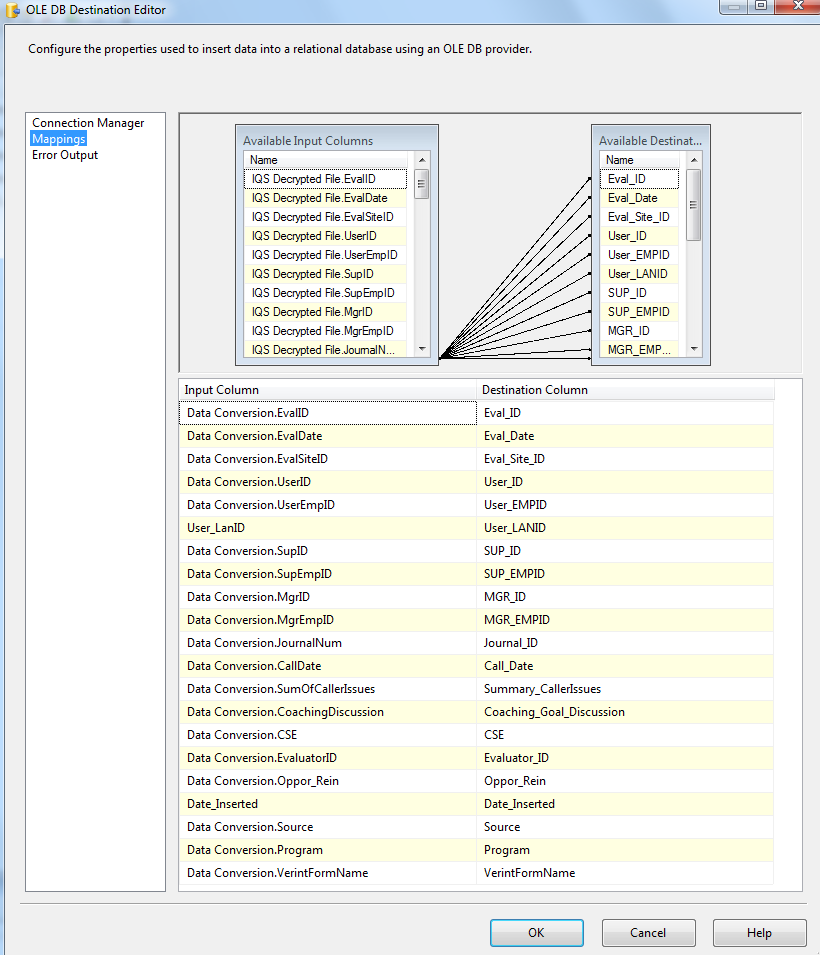


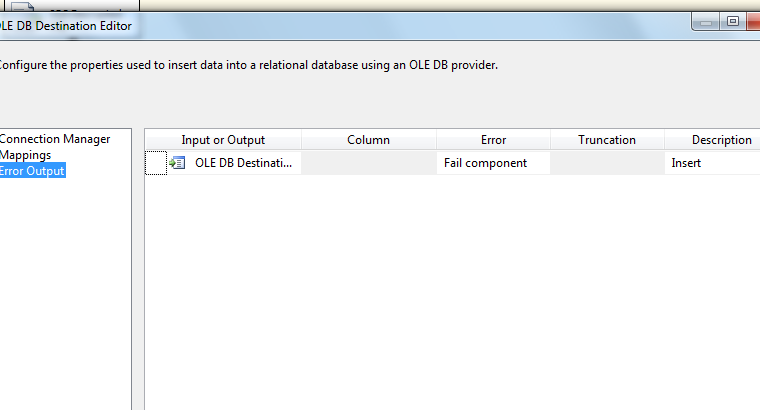


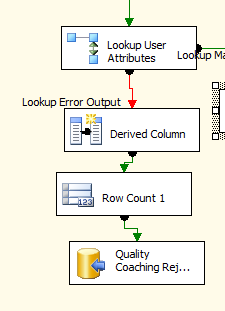


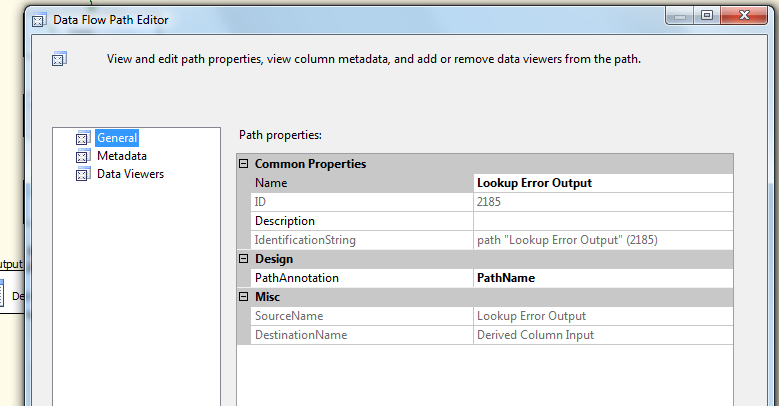


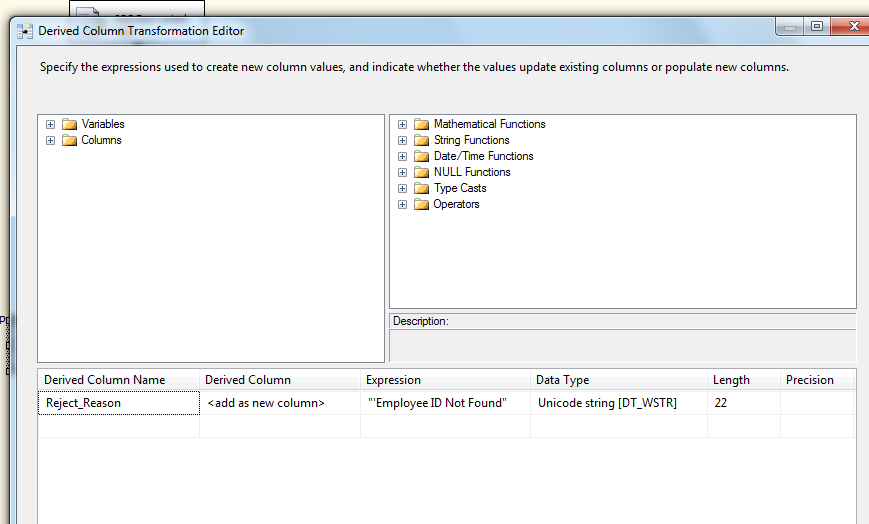


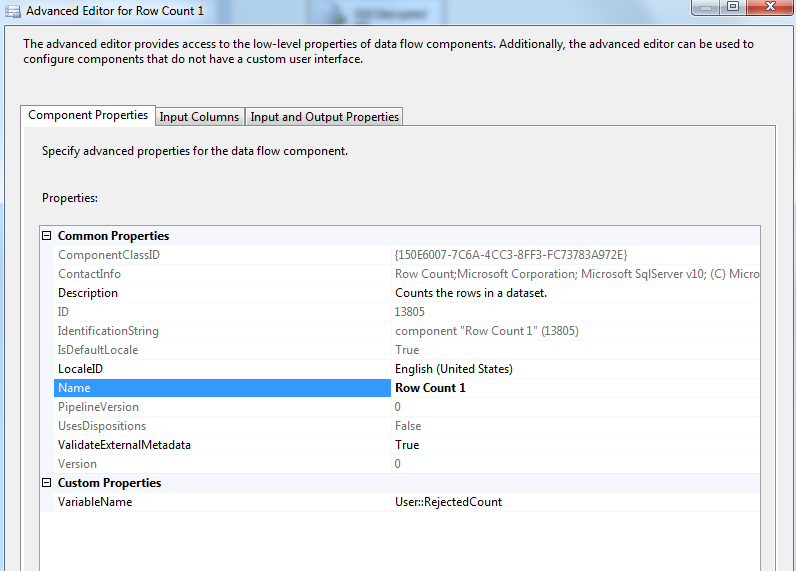


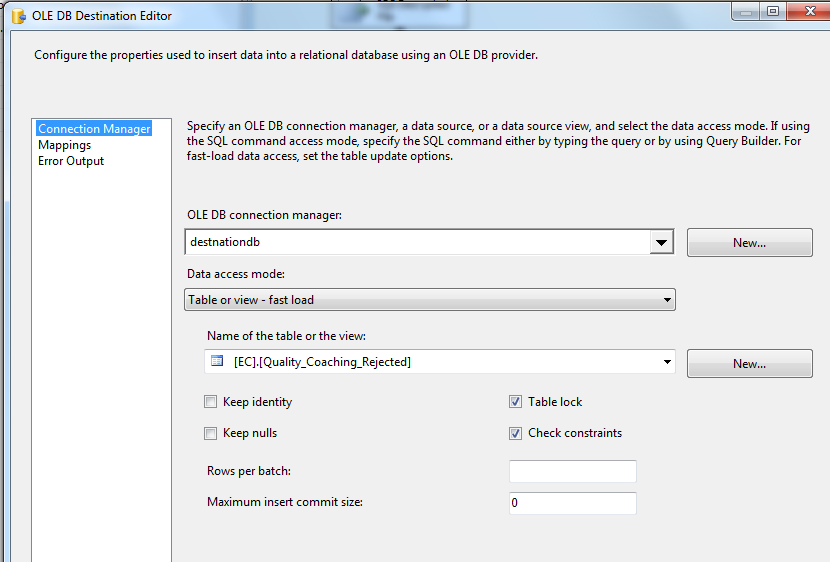


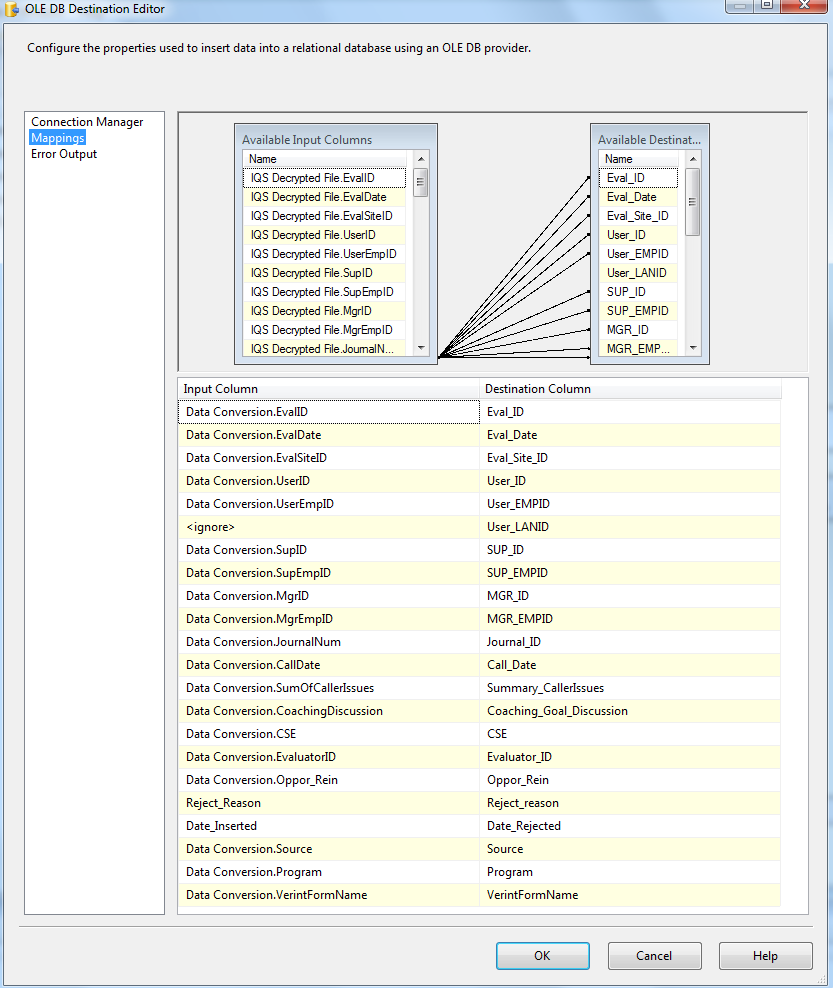


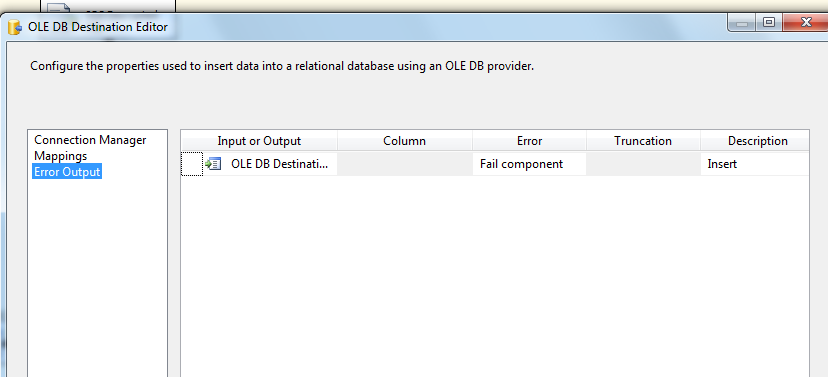




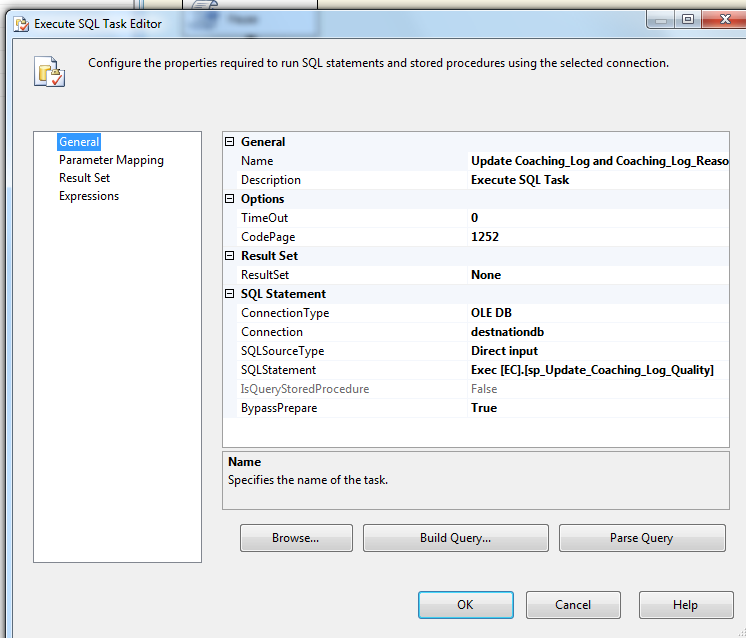




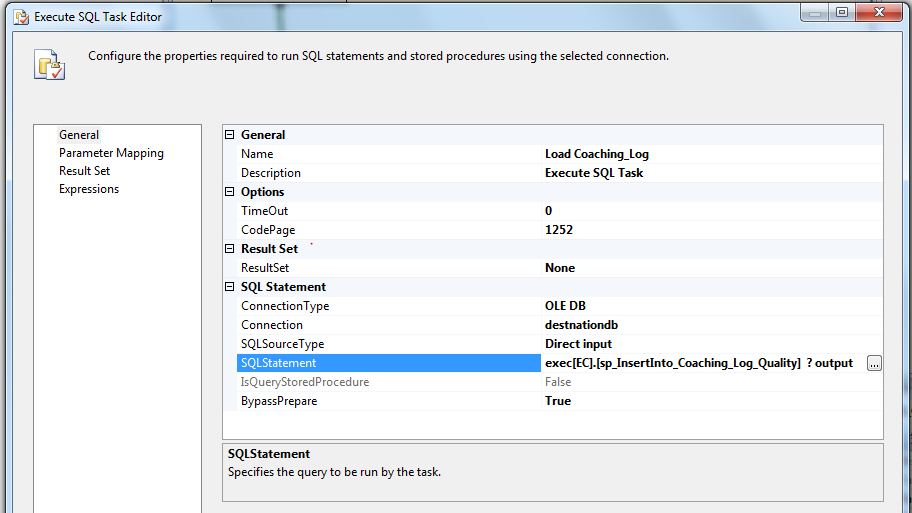




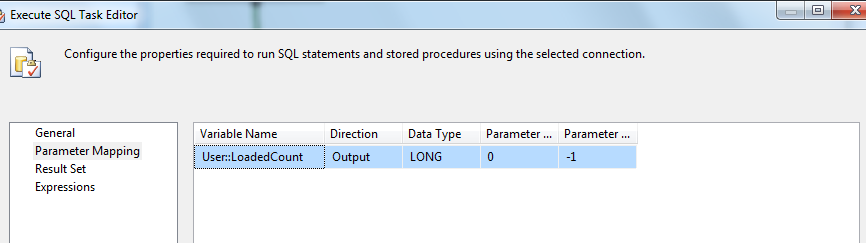
**Step 5: sql task –** Update Coaching\_Log and Coaching\_log\_reason tables



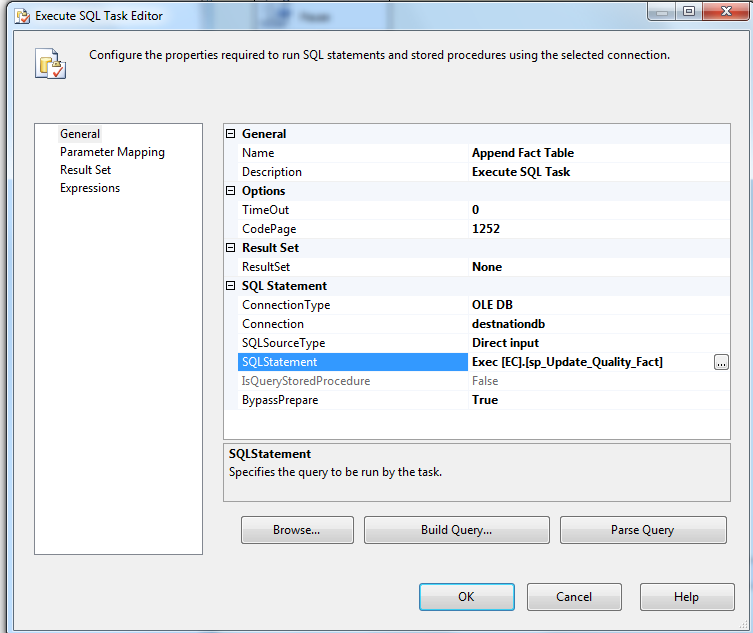
**Step 6: sql task –** Load Coaching\_Log and Coaching\_log\_reason tables



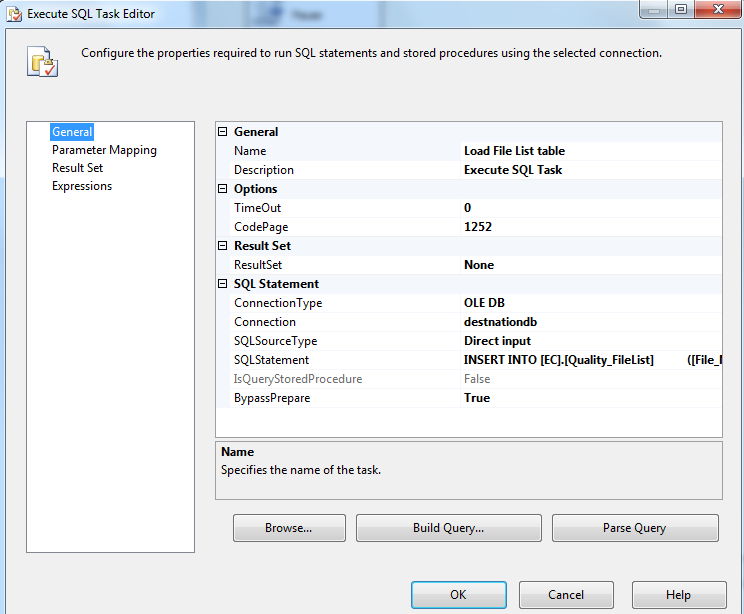
SQL statement - exec[EC].[sp\_InsertInto\_Coaching\_Log\_Quality] ? output



**Step 7: SQL task –** Update Quality Fact Table



**Step 8: SQL task –** Load File List Table



SQL Statement –

INSERT INTO [EC].[Quality\_FileList]

([File\_Name]

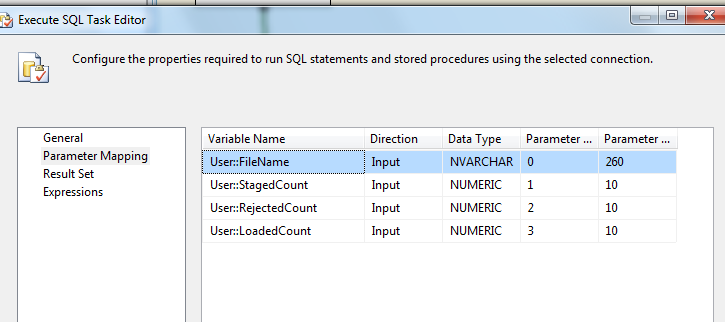
,[File\_LoadDate]

,[Count\_Staged]

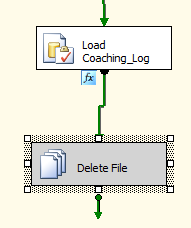
,[Count\_Rejected]

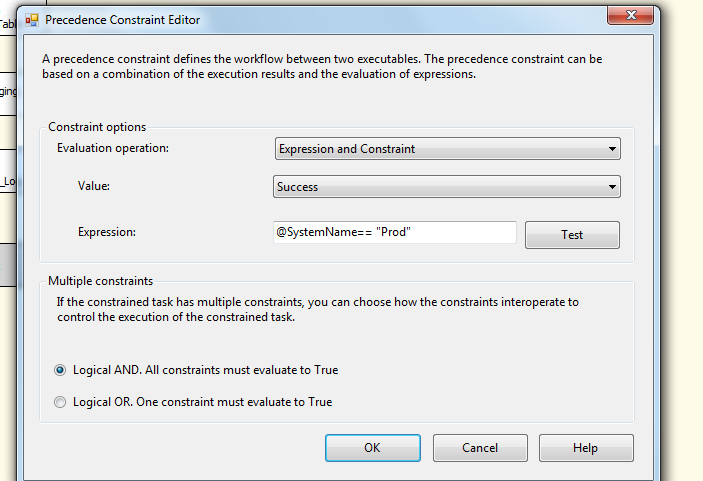
,[Count\_Loaded])

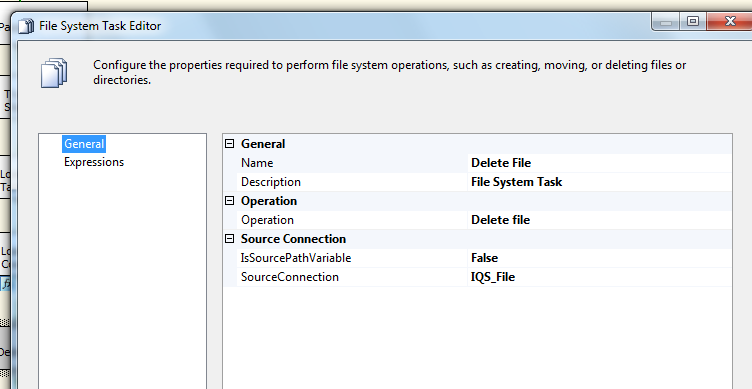
VALUES (?,GetDate(), ?, ?,?)



**Step 9: SQL task Delete File if Environment is Prod (Disabled)**







### Tables

|  | **Table Name** | **Table Description** |
| --- | --- | --- |
| 1. | Quality\_Coaching\_Stage | Table used for staging the Quality evaluations from IQS during the insert into Coaching\_Log table. |
| 2. | Quality\_Coaching\_Rejected | Table used for storing the rejected Quality evaluations from IQS during the insert into Coaching\_Log table. Logs are rejected when a Lan ID cannot be looked up for an incoming Employee ID in the file. |
| 3. | Quality\_Coaching\_Fact | Table used for storing Quality source records loaded into the Coaching Log table. |
| 4. | Quality\_FileList | Table used for storing Quality file names and Load dates. |
| 5. |  |  |

Table structure and details in document: eCoaching\_Database\_DD.docx

### Procedures

|  | **Procedure Name** | **Usage** |
| --- | --- | --- |
| 1. | sp\_InsertInto\_Coaching\_Log\_Quality | Load: Quality Data |
| 2. | sp\_Update\_Coaching\_Log\_Quality | Load: Quality Data |
| 3. | sp\_Update\_Quality\_Fact | Load: Quality Data |
| 4. |  |  |

Reference Document for Objects and Code: cms\eCoaching\Documentation\ eCoaching\_Quality\_Create.sql