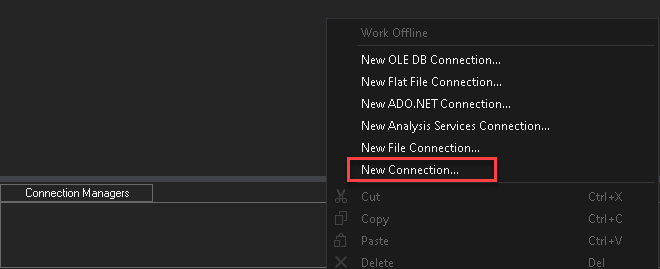
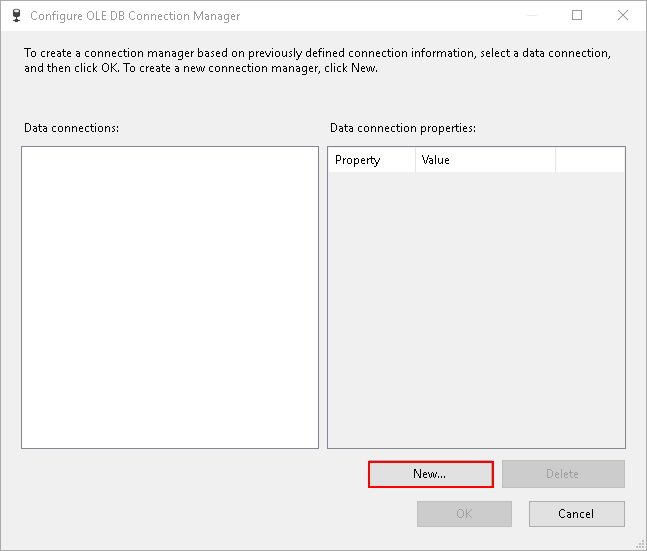
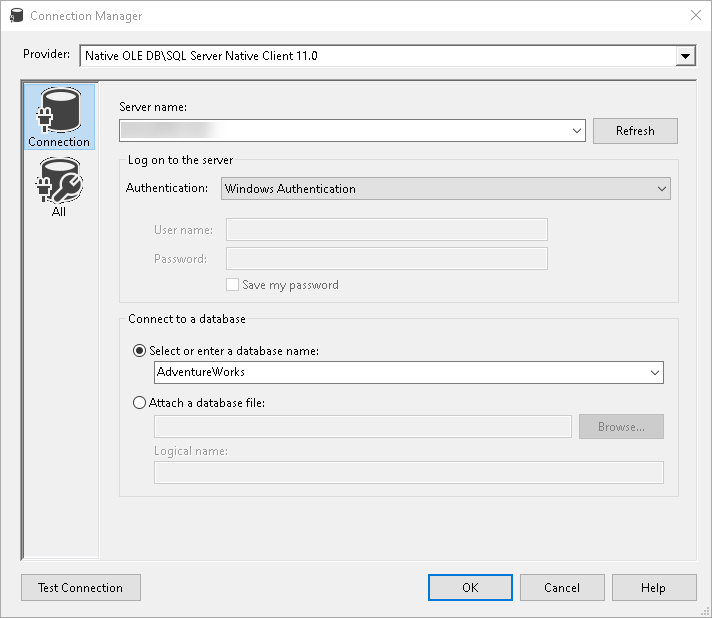
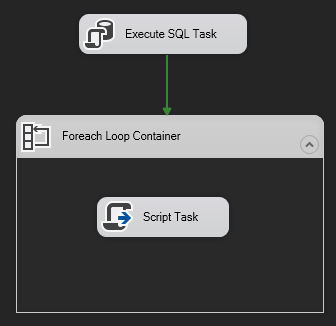
SQL Server Integration Services

# Module 04 – LAB 02: Event handlers

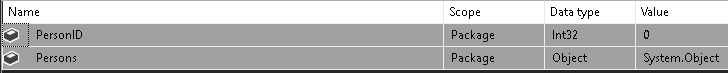
1. Create new integration services project.
2. Set up a connection manager for the database. In the bottom center pane under Connection Manger, right-click select New Ole-DB Connection.  
   
3. In Configure OLE DB Connect Manager, click New.  
   
4. Setup connection to SQL Server pointing to the AdventureWorks database.



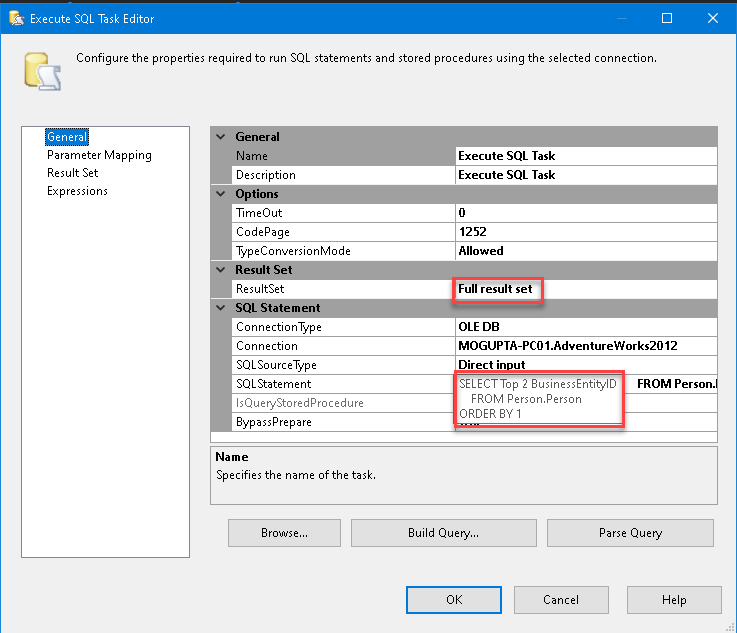
1. Setup the tasks (Execute SQL Task, Foreach Loop Container, and Script Task). and link them as per the diagram below.



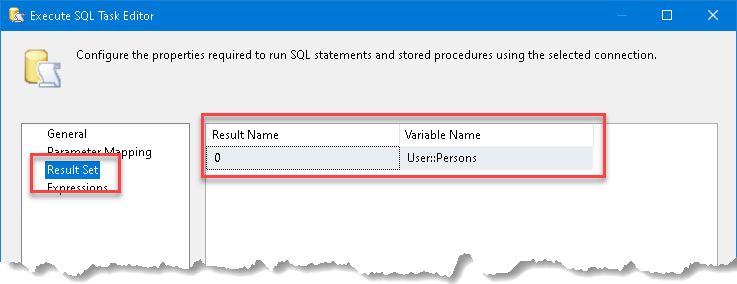
1. Setup two package variables (PersonID and Persons).



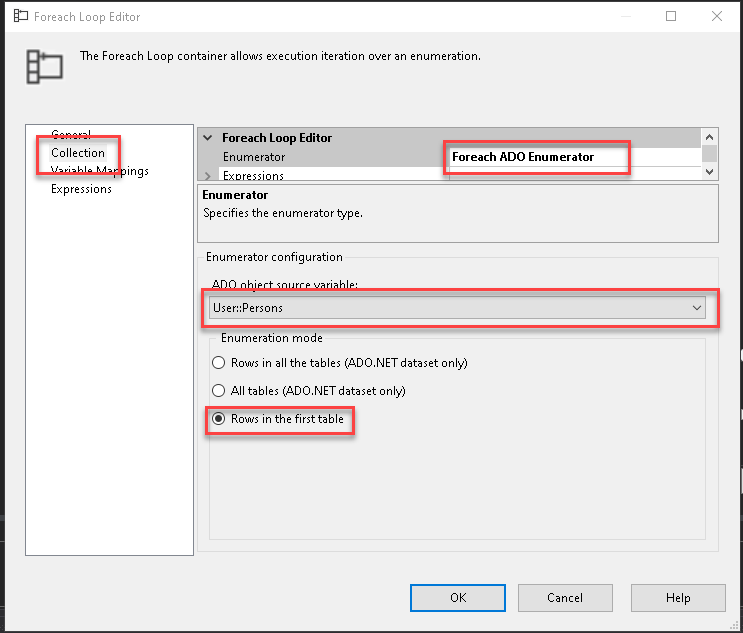
1. Configure the Execute SQL Task linking to connection created in Step #1.
2. Execute the SQL Statement (SELECT TOP 2 BusinessEntityID FROM Person.Person ORDER BY 1), output the results to a “Full result set.”



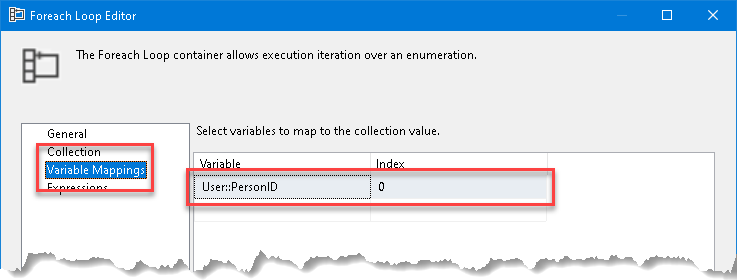
1. Save the result set to a variable created in step #6.



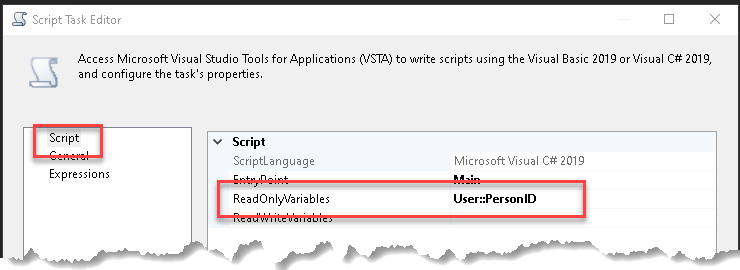
1. Configure the ForEach loop. Under Enumerator select “Foreach ADO Enumerator”. From ADO object select the variable User::Persons.

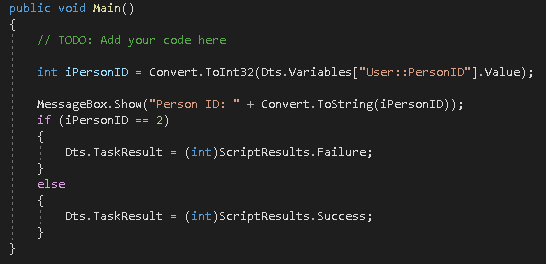


1. Capture the output of each row in result set and save it to variable User::PersonID.



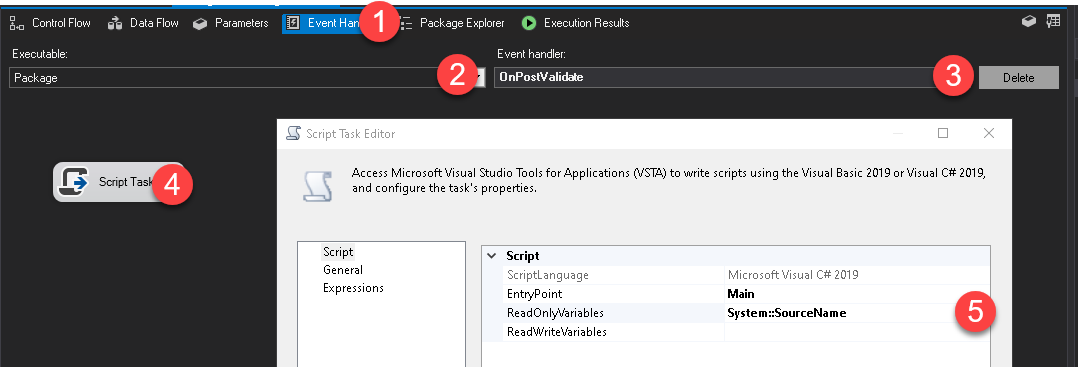
1. Modify the script task, pass in User::PersonID as ReadOnlyVariable. Add the following code in the script:



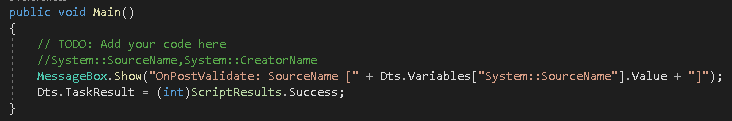


|  |
| --- |
| int iPersonID = Convert.ToInt32(Dts.Variables["User::PersonID"].Value);  MessageBox.Show("Person ID: " + Convert.ToString(iPersonID));  if (iPersonID == 2)  {  Dts.TaskResult = (int)ScriptResults.Failure;  }  else  {  Dts.TaskResult = (int)ScriptResults.Success;  } |

1. After setting up the script test, it out. It should come back with two messages and fail because of result status returned from script.
2. Next add the following steps for each of the events under package.

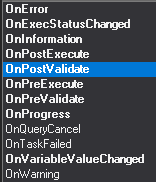


* 1. Go to Event Handler under package.
  2. Select Package from Executable.
  3. Select first event in list.
  4. Create Script Task.
  5. In script task pass in ReadOnlyVariable, System::SourceName.
  6. Create the following script.



|  |
| --- |
| MessageBox.Show("OnPostValidate: SourceName [" + Dts.Variables["System::SourceName"].Value + "]"); |

* 1. Repeat the above steps for each event under Package.



1. Execute the package and review what order the events are raised. Notice how events are bubbled up from deepest execution “Script Task” in Container to Package level.