**SQL Server Integration Services (SSIS) Tasks (material taken from [1])**

1. **Execute SQL Task**

The Execute SQL Task is one of the most widely used tasks in SSIS for interacting with an RDBMS Data Source. The Execute SQL Task is used for all sorts of things, including truncating a staging data table prior to importing, retrieving row counts to determine the next step in a workflow, or calling stored procedures to perform business logic against sets of staged data.

**Executing a parameterized SQL Query**

In this example, we are going to update a table

1. Drag and drop the Execute SQL Task

2. Double click the Task to open the configuration editor

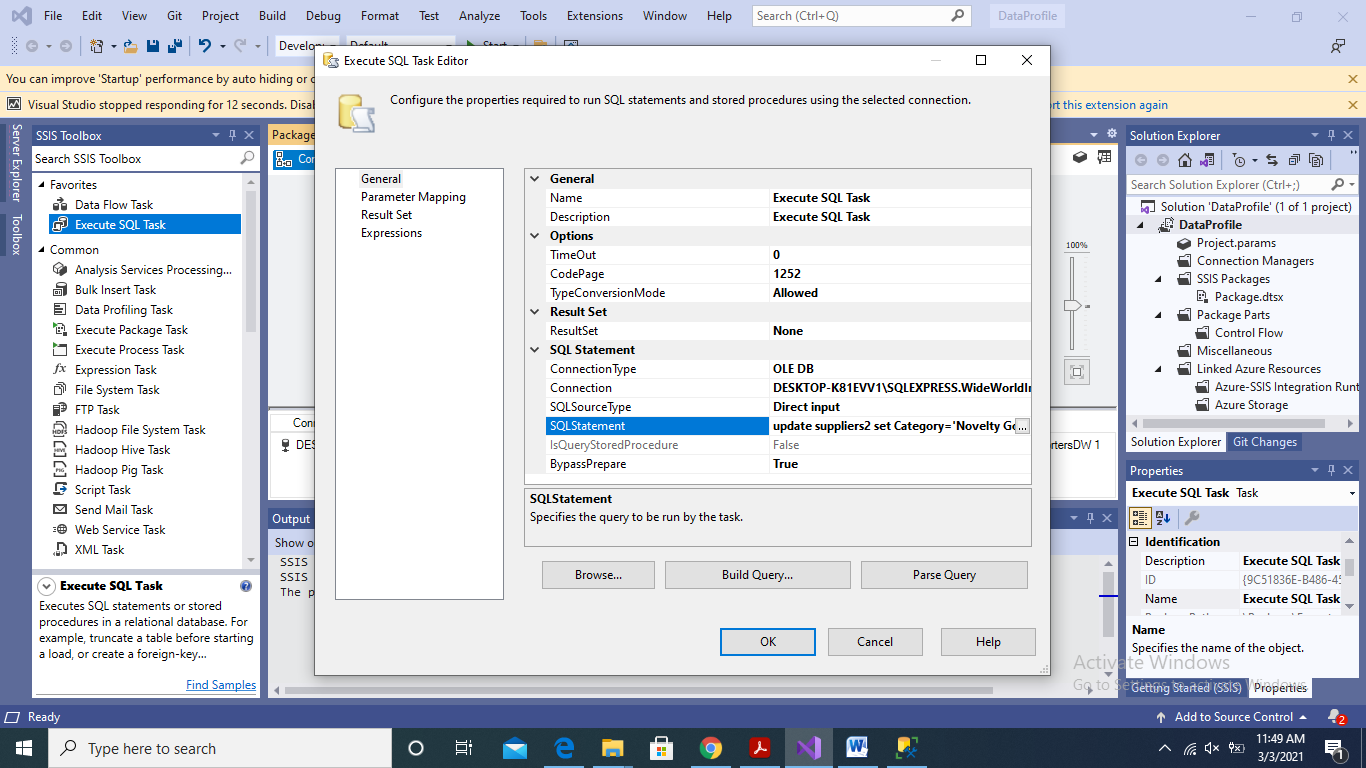
3. Make a connection to a database stored in SQL Server. In my case I have made a connection with WideWorldImportersDW

4. In the SQLStatement write the update query. In my case I have written a query as

update suppliers2 set Category='Novelty Goods Supplier' where [Supplier Key] = 3

5. Click Ok and execute the package. If there is no error then you will see that the SQL query has updated the desired value

6. The following figure shows the configuration editor with the above mentioned configurations



---------------------------------------------------------------------------------------

7. You can also use a parameter to set the value of [Supplier Key] in the above query. In this case the query would be

update suppliers2 set Category=”Courier” where [Supplier Key] = ?

8. In this case make a new User variable. In my case I have made a variable with the name Supplier\_Key and set its value as 3

9. In the Execute SQL Task update the query as above in point no. 7 and then select Parameter Mapping

10. Select the Variable Name as User:Supplier\_Key

11. In the parameter name set its value as 0. This is because we have only one ? mark in the above query. So the 0 means the first ? mark. If there are more ? marks in a query they are referred to with a number starting from 0. This convention is followed in OLEDD connection.

12. Execute the package. If there is no error then the Supplier with [Supplier Key] as 3 will be updated and its Category will be set to

------------------------------------------------------------------------------

13. Next, we can also make a connection as ADO.NET where the query will not execute with ? mark. Rather we must make a named parameter.

14. Make a Connection with ADO.NET and select a database

15. Update the query as update suppliers2 set Category=”Courier2” where [Supplier Key] = @Supplier\_Key

16. In the Parameter Mapping, in the Parameter Name enter value as @ Supplier\_Key

17. Select the data type as Int16

18. Make sure that the variable Supplier\_Key has your desired value

19. Execute the package if there is no error then Category will be updated as Courier2 of the supplier with the provided Supplier\_Key

**Capturing Singleton Results**

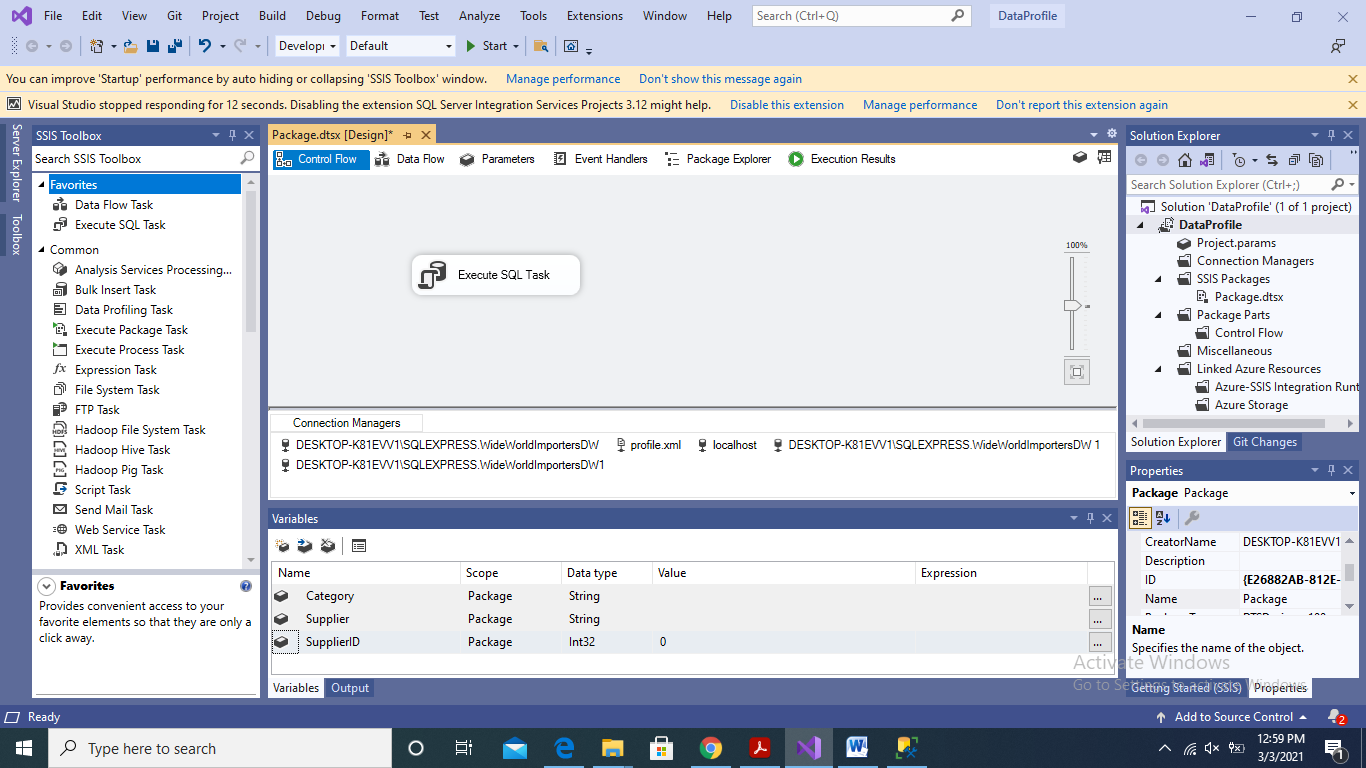
With Execute SQL Task, you can also get a single row result. The result of the query then can be stored in variables that then can be used to make further workflows. For this purpose follow the below given steps.

1. Make a Connection with ADO.NET and select a database

2. SELECT [WWI Supplier ID] ,[Supplier],[Category] FROM [WideWorldImportersDW].[dbo].[suppliers2]

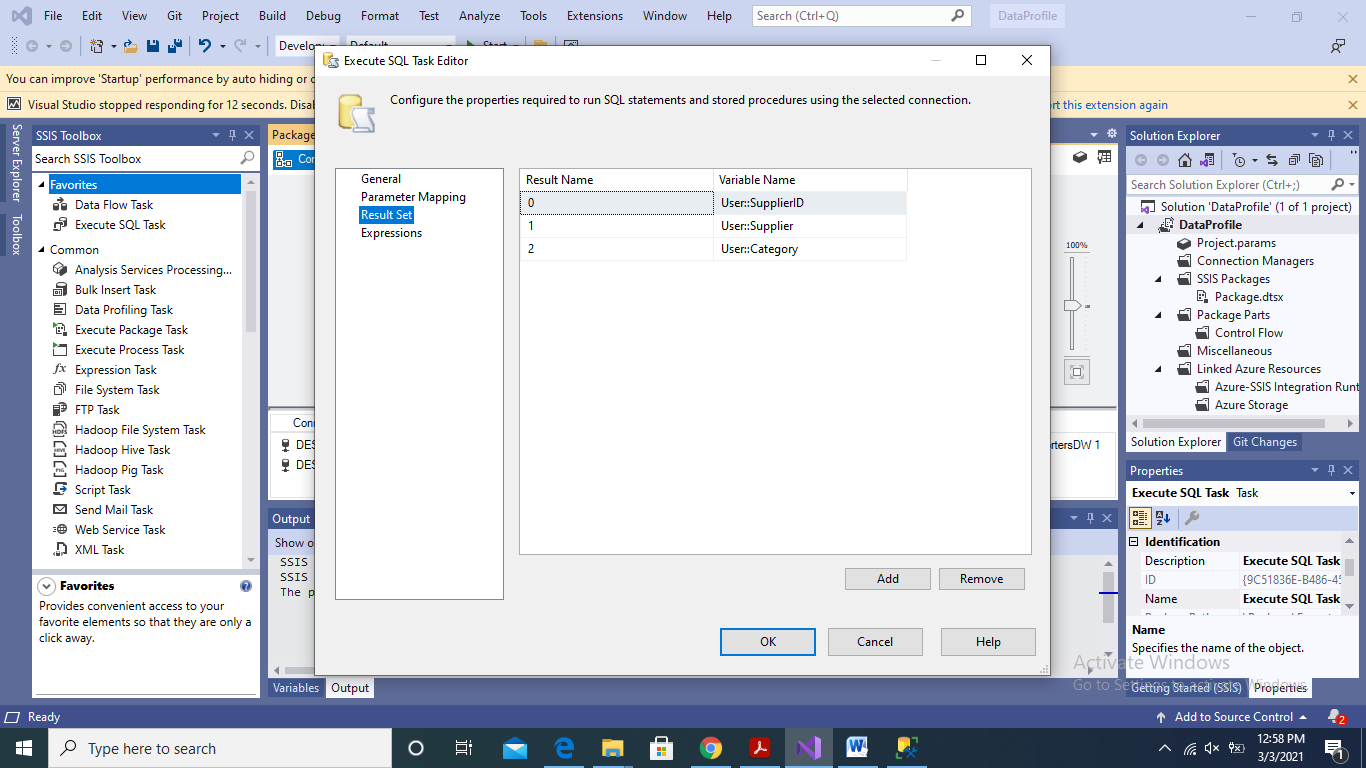
where [Supplier Key] = 3

3. Make the variables as shown in the following figure. Please make sure that you select the correct data type of variables. The selection of correct data type for variables in SSIS is very important to execute tasks property



4. In the ResultSet select Single Row

5. In the Result Set Properties map the results of the query in the defined variables as show in the following figure



6. Execute the package if there is no error then the values of the query will be stored in the variables

**Multi-Row Results**

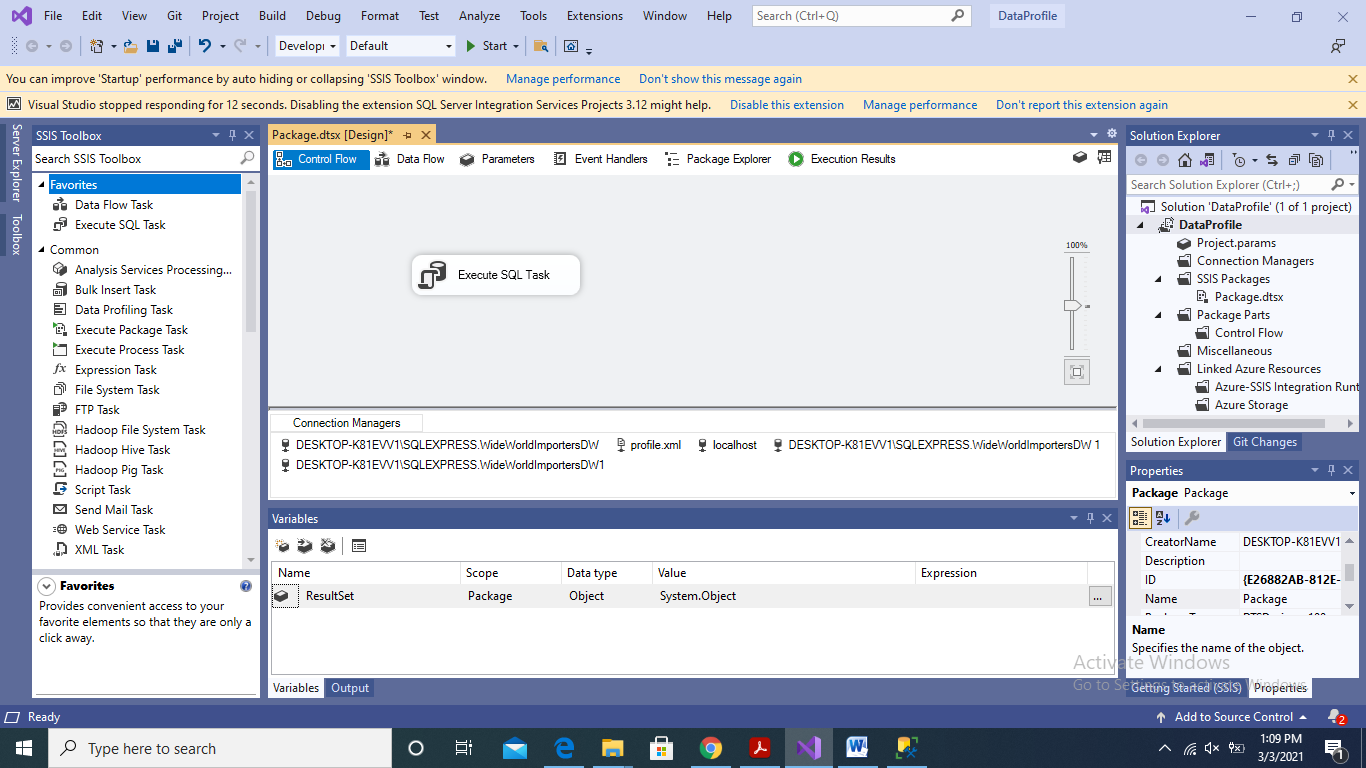
Typically, you capture multi-row results from a database as a recordset or an XML file (particularly between SQL Server Data Sources) to use in another Script Task for analysis or decision-making purposes, to provide an enumerator in a Foreach or Looping Task, or to feed into a Data Flow Task processing.

With Execute SQL Task, you can also get a single row result. The result of the query then can be stored in variables that then can be used to make further workflows. For this purpose follow the below given steps.

1. Make a Connection with ADO.NET and select a database

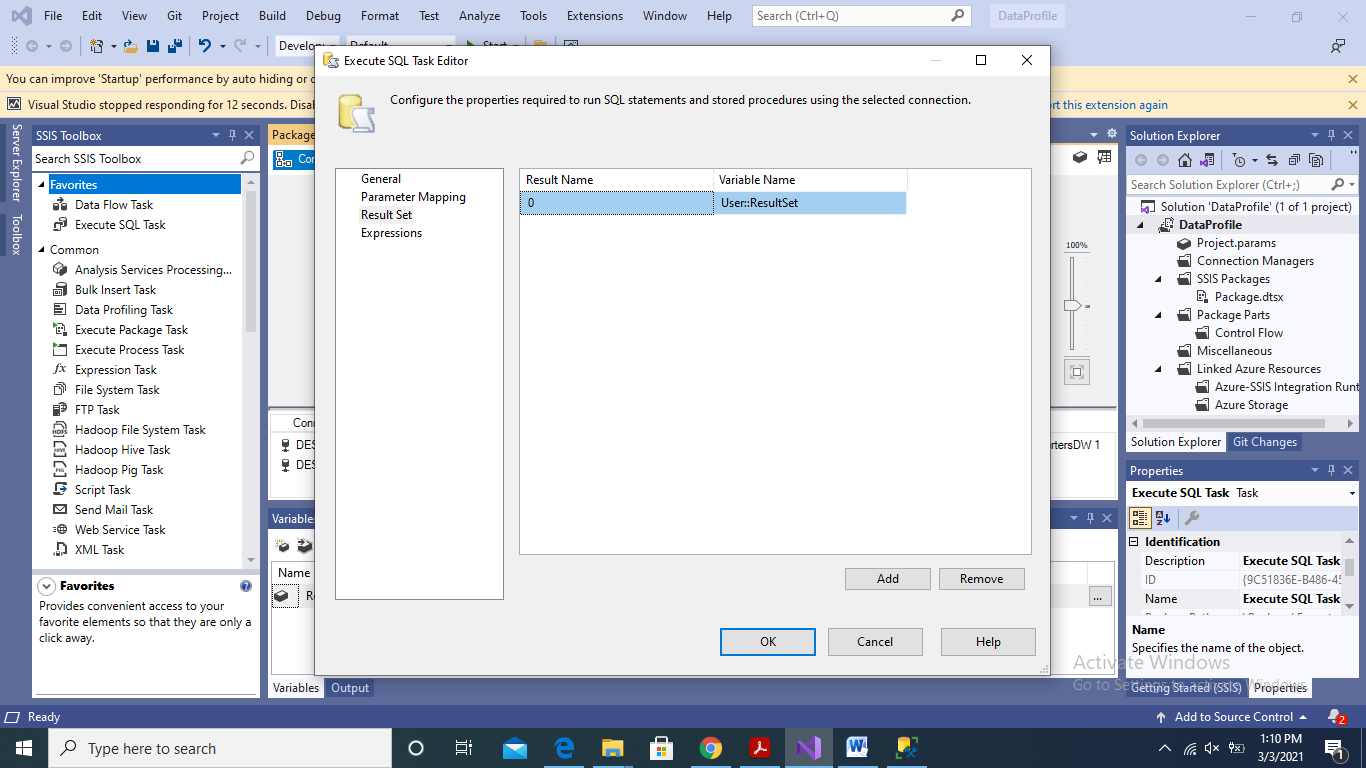
2. SELECT [WWI Supplier ID] ,[Supplier],[Category] FROM [WideWorldImportersDW].[dbo].[suppliers2]

3. Make a variable with object as data type as shown in the following figure.



4. In the ResultSet select Full result set

5. In the Result Set tab map the results of the query in the defined variables as show in the following figure



6. Execute the package if there is no error then the values of the query will be stored in the ResultSet variable. This variable then can be used as an input to data flow tasks for further processing or transformations.

7. The XML version of the same task is further very easy. In that case an XML of the result set will be created which then can be parsed by XML Source in Data Flow tasks to perform further processing or trasnformations.

References:

[1] Knight, B., Knight, D., Moss, J. M., Davis, M., and Rock, C.: “Professional Microsoft SQL Server 2014 Integration Services”. Publisher: Wrox