**Bulk Insert Task in SSIS**

The Bulk Insert task in SSIS can transfer data **only from** a text file into a SQL Server table or view, which is similar to Bulk Insert in SQL Server. If the destination table or view already contains data, the new data is appended to the existing data when the SSIS Bulk Insert task runs. If you want to replace the data, run an Execute SQL task that runs a DELETE or TRUNCATE statement before you run the Bulk Insert task.

### Points to Remember in SSIS bulk insert task

* The Bulk Insert task in SSIS can transfer data **only from** a text file into a [SQL Server Table](https://www.tutorialgateway.org/sql-create-table/) or SQL Server [View](https://www.tutorialgateway.org/views-in-sql-server/).
* Bulk Insert Task supports the Flat file Connection manager to select the text file.
* The Bulk Insert Task in SSIS only supports [OLE DB Connection Manager in SSIS](https://www.tutorialgateway.org/ole-db-connection-manager-in-ssis/) for the destination database.
* Destination table must exist before it is using in the Bulk Insert Task
* Don’t forget to change the **First** **Row** option to 2, if you have your column names in the first row of a text file.
* It is always good practice to set the batch size to insert a large amount of data.

# Data Profiling Task in SSIS

The Data Profiling Task in SSIS used to computes various profiles that help us to become familiar with the data source and to identify the problems in the data (if any) that have to fix. Here, we show you how to profile the source data using the Data Profiling Task in SSIS with example.

The Data Profiling Task in SSIS will work only with the data present in SQL Server. The SSIS Data Profiling Task doesn’t support the data present in the file system, or the third-party data.

# Execute T-SQL Statement Task in SSIS

The Execute T-SQL Statement Task in SSIS is used to run only the T-SQL statements. In this article, we are going to create the New table and send the following data into that newly created table.

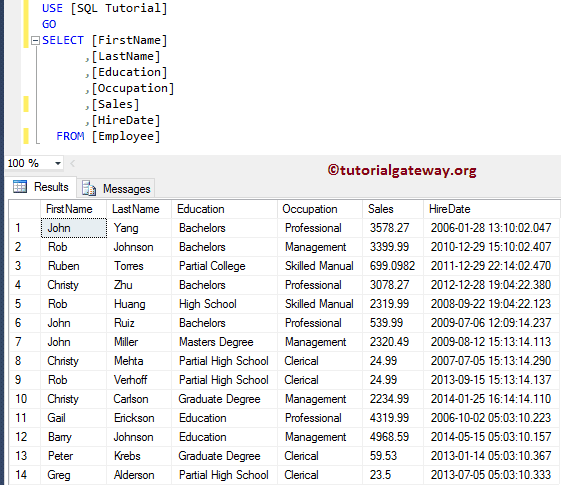
# Execute SQL Task in SSIS Example

How to truncate the SQL tables using the Execute SQL Task in SSIS with an example. It also demonstrates the Result Set option NONE. For this, we are going to use the Employee table present in the SQL Tutorial database

# Execute SQL Task in SSIS Single Rowset

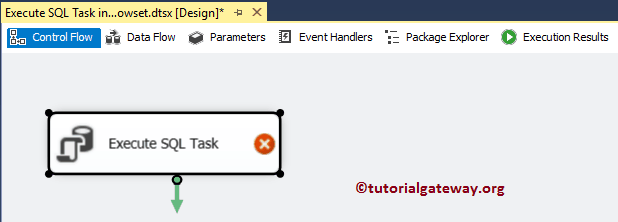
How to return a single rowset (or Single Row) from the SQL tables using the Execute SQL Task in SSIS with an example. It also demonstrates the Result Set option **Single Rowset**. For this SSIS Execute SQL Task Single Rowset demo, we are going to use the Employee table present in the SQL Tutorial database

Before we start working with the Execute SQL Task in SSIS Single Rowset, Let me show you the Employee Table in SQL Server that we are going to use for this example:

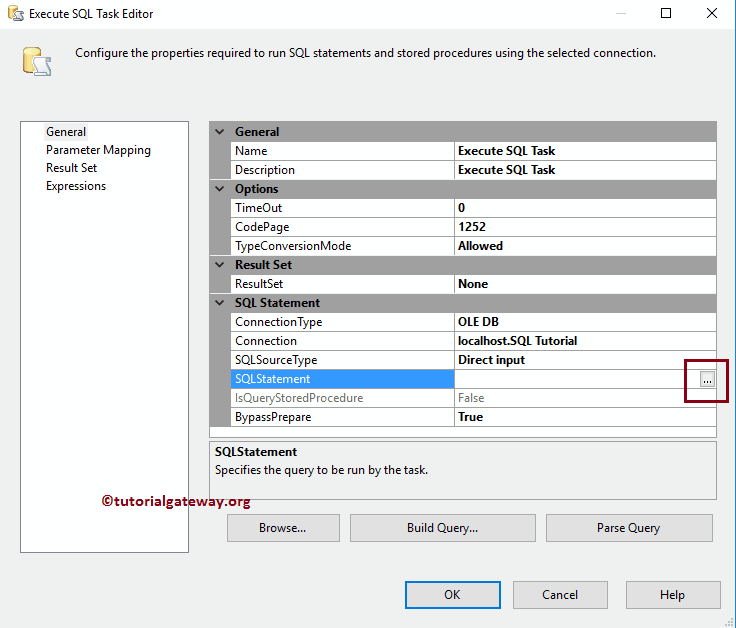


## Execute SQL Task in SSIS Single Rowset

In this example, we will use the Execute SQL Task to select the first row from the Employee table and display the returned single row set in the message box. To do so, Drag and drop the Execute SQL Task from [SSIS](https://www.tutorialgateway.org/ssis/) toolbox to designer space.



Double click on it will open the Execute SQL Task Editor to configure it. Let me select the Connection Type as [OLE DB Connection](https://www.tutorialgateway.org/ole-db-connection-manager-in-ssis/), which is connecting to the SQL Tutorial database. Next, we are using the Direct Input as the [SQL](https://www.tutorialgateway.org/sql/) statement, so click the **…**.



Please write your custom [SQL](https://www.tutorialgateway.org/sql/) statement here. As you can from the below screenshot, we are writing a [SQL Select statement](https://www.tutorialgateway.org/sql-select-statement/) with the [TOP Clause](https://www.tutorialgateway.org/sql-top-clause/) to select the first row from the table.

-- Execute SQL Task in SSIS Single Rowset Example

SELECT TOP 1 [FirstName]

,[LastName]

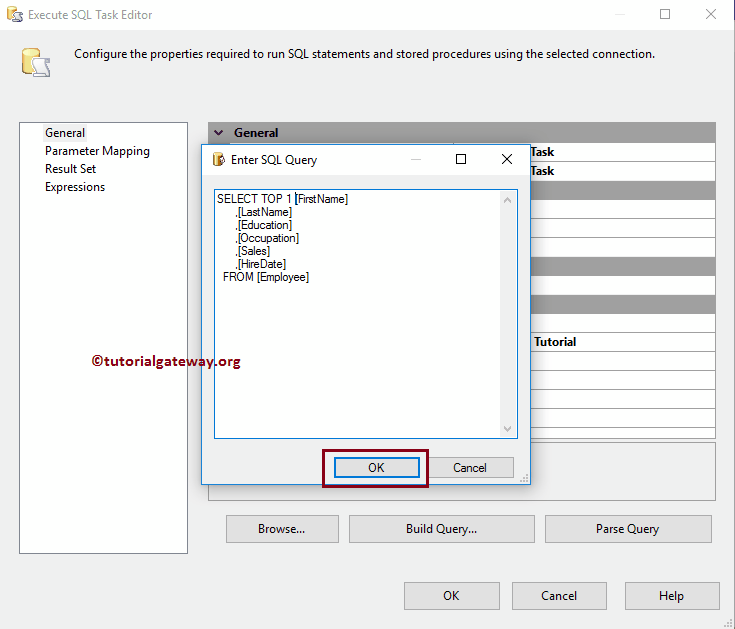
,[Education]

,[Occupation]

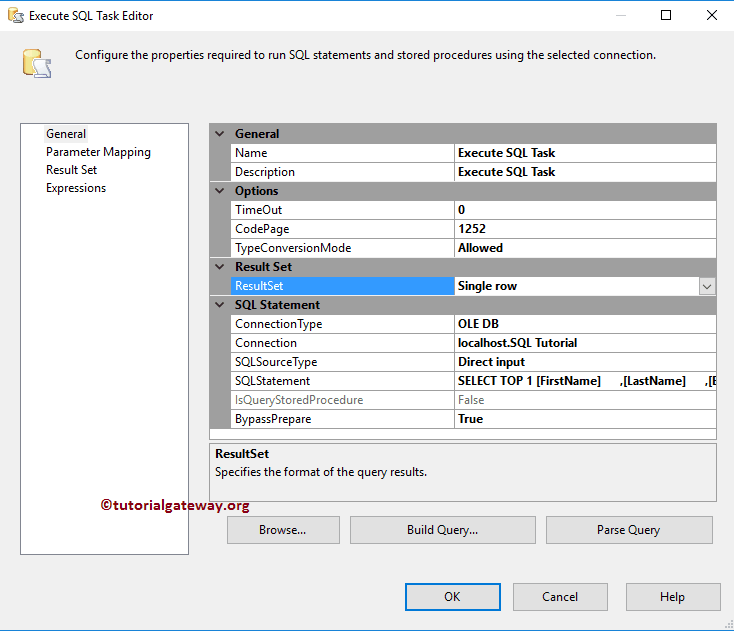
,[Sales]

,[HireDate]

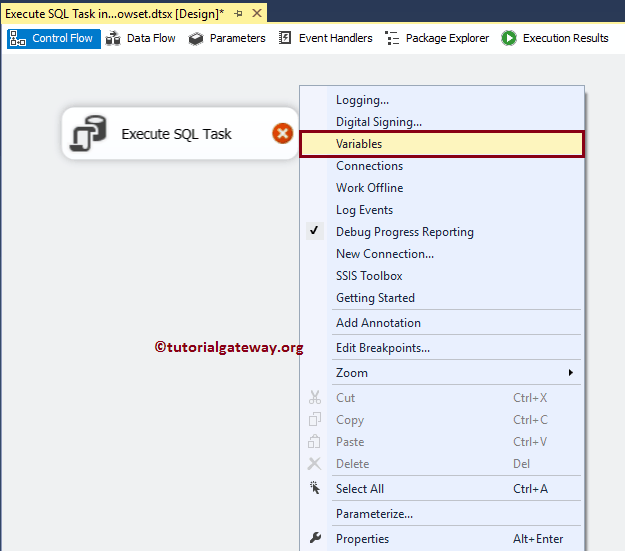
FROM [Employee]



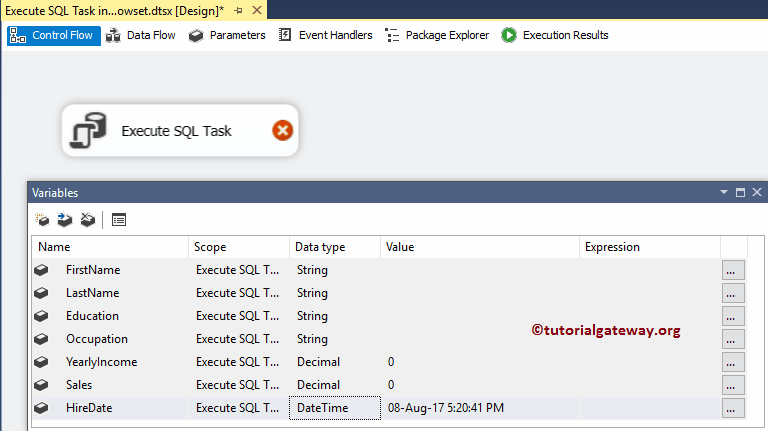
Please change the ResultSet from NONE (default) to Single row, because our select statement will return one row.



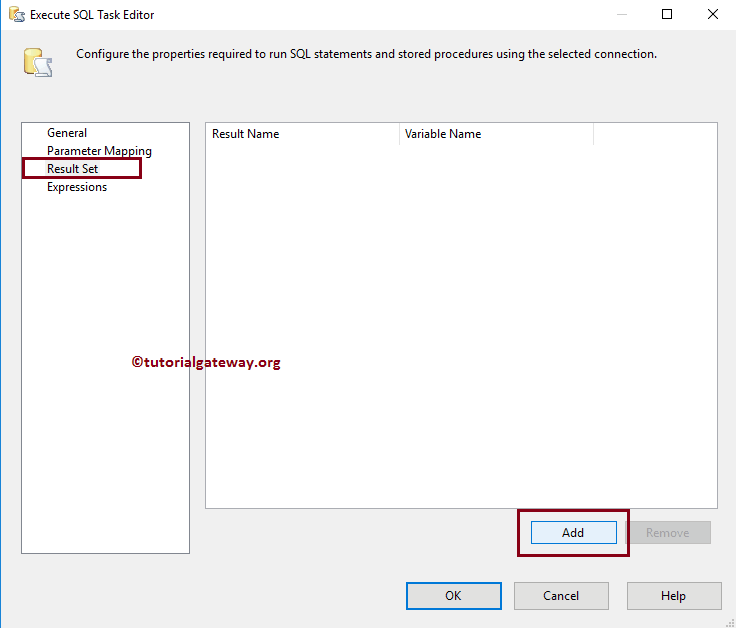
Let us create variables required to hold the return records. First, close the Execute SQL Task editor, and right-click on the design will open the context menu. Please select the Variables option.



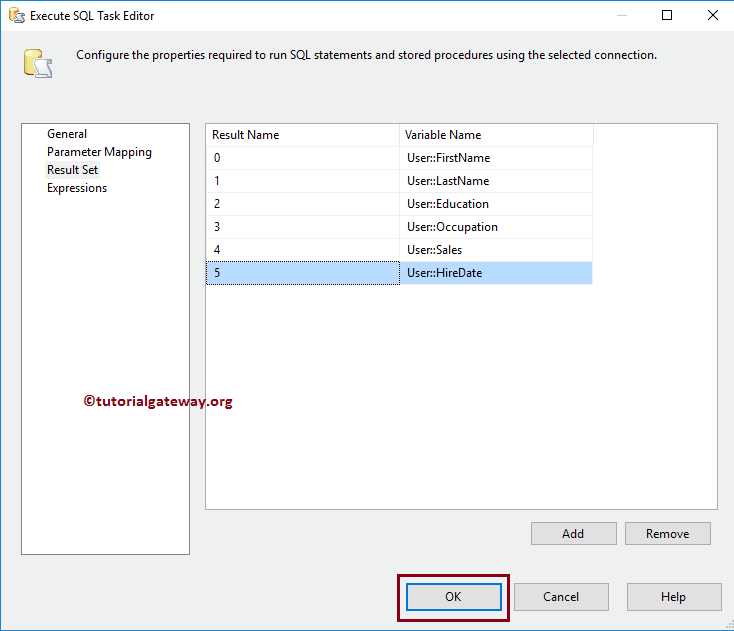
As you can from the below screenshot, we created 7 variables to hold seven columns returned by the select statement.



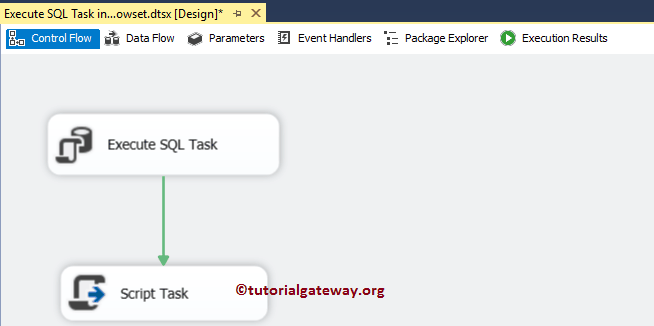
Next, go to the Result Set tab in Execute SQL Task Editor and click the Add button to assign the variables for the return set.



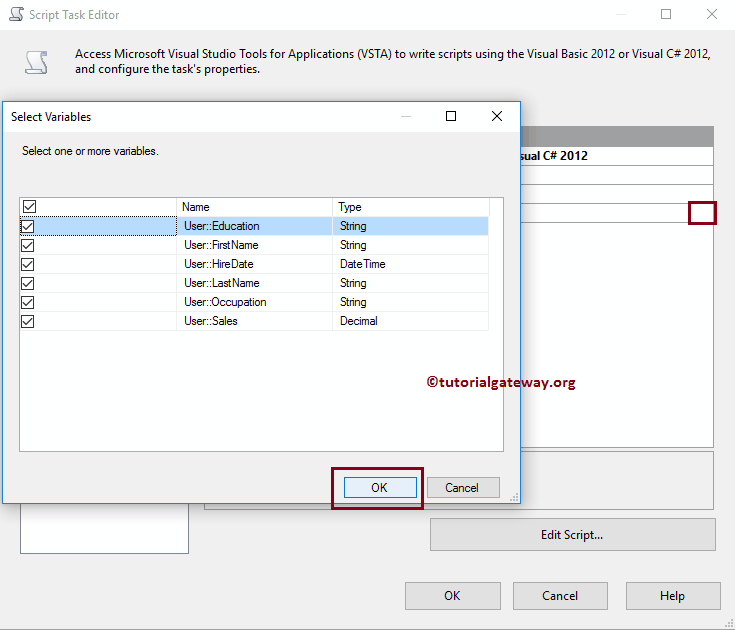
As you can see from the below screenshot, we are assigning the previously created variables to all the columns that are returned by the [SQL](https://www.tutorialgateway.org/sql/) statement.



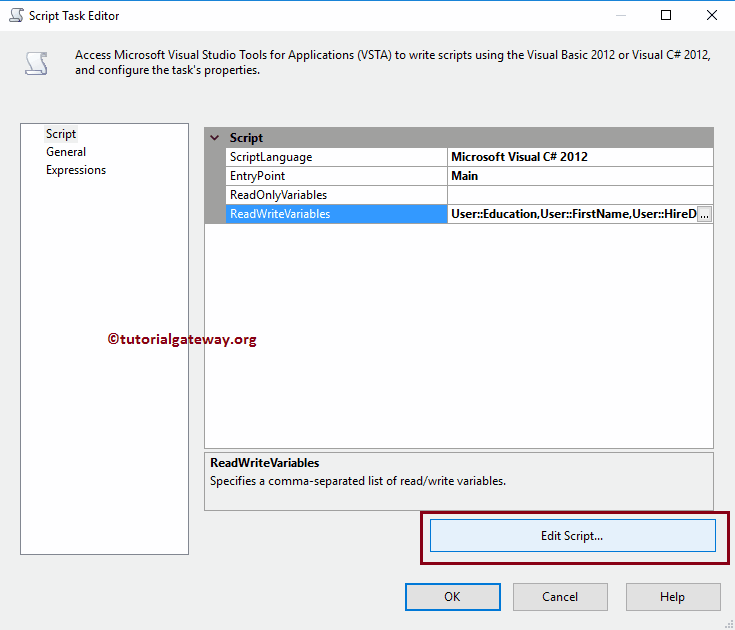
We completed configuring SSIS Execute SQL Task Single Rowset. Let me display the returned result in the Message Box. To do so, drag and drop the [Script Task](https://www.tutorialgateway.org/script-task-in-ssis/) from the [SSIS](https://www.tutorialgateway.org/ssis/) toolbox into the Control Flow region and add the Execute SQL Task output to the [Script Task](https://www.tutorialgateway.org/script-task-in-ssis/).



Double click on the Script task will open the following editor to configure the Script task components. Here we are selecting all the user-defined variables (a variable that we created earlier) as Read and Write variable.



Once you selected the required variable, please click on the ***Edit Script..***button to write the actual C# or VB Script



Here we first declared the local variables to hold the user-defined variables that we are getting from the SSIS package. Next, we are concatenating the value with custom text (“Name:” etc)

C# code we used in the below screenshot is:

-- Execute SQL Task in SSIS Single Rowset Example

String FullName = "Name: " + Dts.Variables["FirstName"].Value.ToString() + " " +

Dts.Variables["LastName"].Value.ToString();

String Occupation = "Occupation: " + Dts.Variables["Occupation"].Value.ToString();

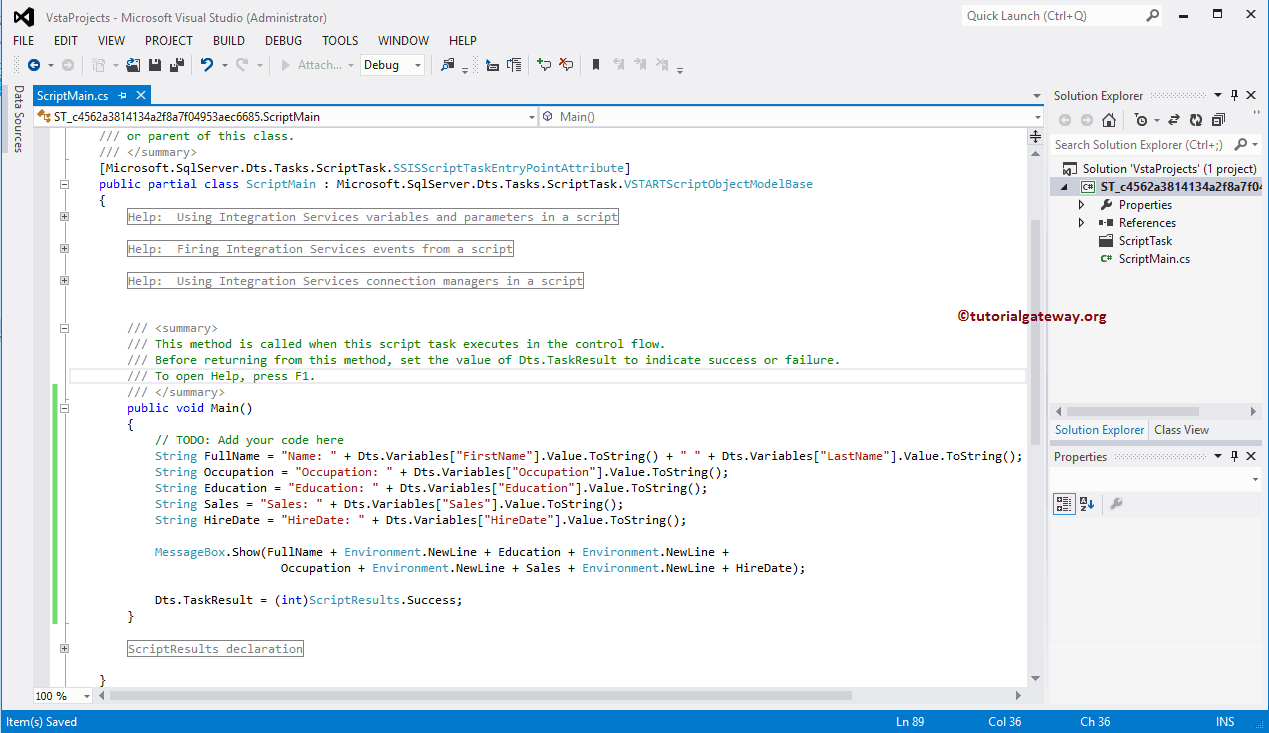
String Education = "Education: " + Dts.Variables["Education"].Value.ToString();

String Sales = "Sales: " + Dts.Variables["Sales"].Value.ToString();

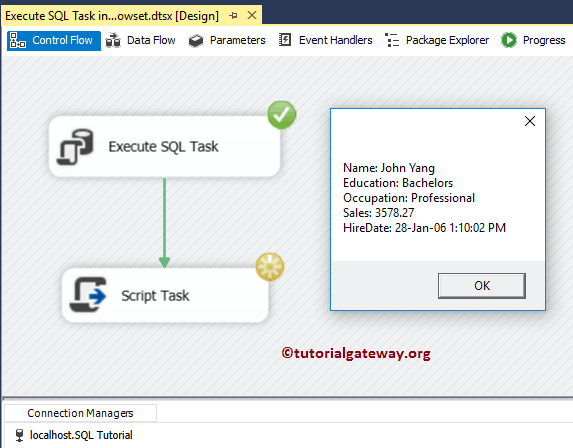
String HireDate = "HireDate: " + Dts.Variables["HireDate"].Value.ToString();

MessageBox.Show(FullName + Environment.NewLine + Education + Environment.NewLine +

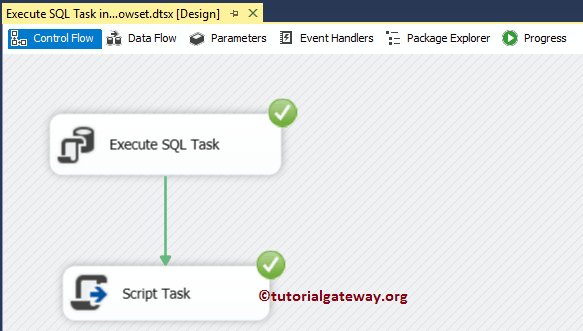
Occupation + Environment.NewLine + Sales + Environment.NewLine + HireDate);



Click OK to finish configuring the Execute SQL Task in SSIS Single Rowset package. Let us Run the package



Click the OK button on the message box to finish the execution process



# Execute Package Task in SSIS

The Execute Package task in SSIS is the most useful in SQL Server Integration Services because it allows us to call other packages from the workflow.

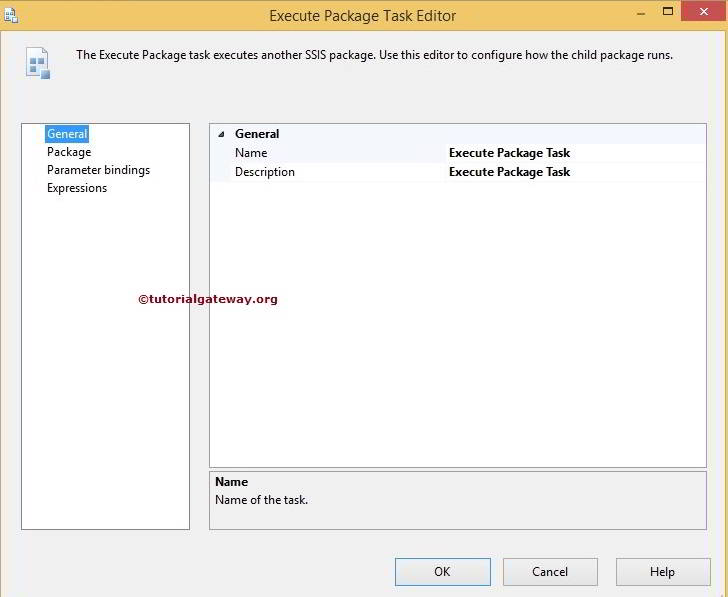
The following are a few advantages of using Execute Package task in SSIS:

* We can break down the complex packages into multiple packages. And then, using the Execute Package task, we can call them at any time. For example, Instead of performing 10 transformations in a single SSIS package. We can break the single package into 10 packages. Then using the Execute Package task, we can call them.
* If we break down the complex packages into multiple packages, we can reuse them numerous times. It means Code usability.
* If we break down the complex packages into multiple packages, maintenance will be easy and cheaper.
* If we break down the complex packages into multiple packages, it helps us to divide the work to the whole team where each individual can work on a single package, and module lead can access all the packages by creating the parent package. Any package that calls other packages (child packages) called to as parent package.

## Configuring Execute Package Task in SSIS

Double click on the SSIS Execute Package task will open the Execute Package task Editor to configure it.

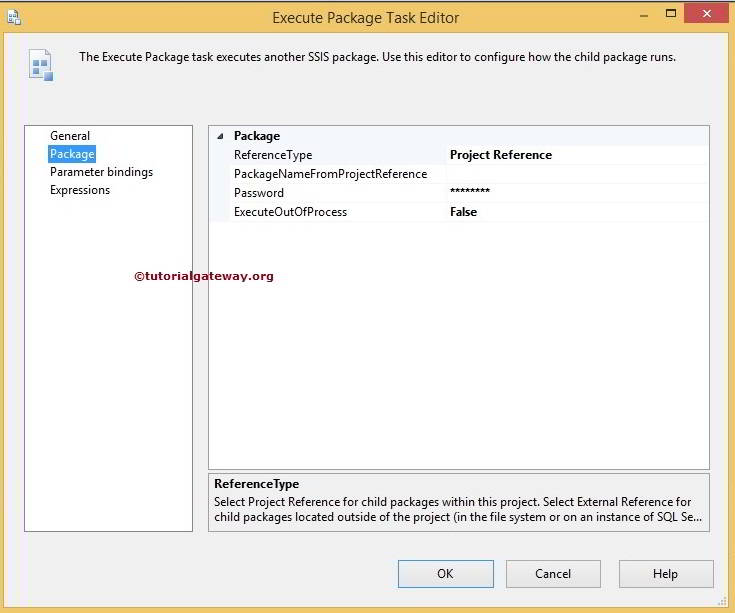
#### General Page



Within the General page, You can configure the Name and Description of the Execute Package Task.

* **Name:** Please provide the Unique name for the Execute Package task.
* **Description:** Please explain what this [SSIS](https://www.tutorialgateway.org/ssis/) task is doing?

#### Package Page



**Reference Type:** The Execute Package task in SSIS allows us to call packages present in the Same Project, File system, and [SQL Server](https://www.tutorialgateway.org/sql/).

* If you select the **ReferenceType** property to **Project Reference,**then the Execute Package task allows us to choose the child package present in the same project that contains the parent package. Please refer [SSIS Execute Package Task Project Reference](https://www.tutorialgateway.org/ssis-execute-package-task-project-reference/) article to understand how to call packages inside the same project.
* If you select the **ReferenceType** property to **External Reference,**the Execute Package task provides two options to choose: **File System** and **SQL**Server
  + **File System:**If we selected the File System, the Execute Package task allows us to choose the child package in the file system. To access the files, it uses the File connection manager. Please refer [Execute Packages in File System using SSIS Execute Package Task](https://www.tutorialgateway.org/execute-packages-in-file-system-using-ssis-execute-package-task/) for calling packages in File System
  + **SQL Server:** If we select the SQL Server, the Execute Package task will allow us to choose the child package stored within the MSDB Database inside our SQL Server. The Execute Package task in SSIS uses the OLE DB Connection Manager to access the packages in the SQL Server. Please refer to [Execute Packages in SQL Server using SSIS Execute Package Task](https://www.tutorialgateway.org/execute-packages-in-sql-server-using-ssis-execute-package-task/) for calling packages in SQL Server.

**Password:** While choosing the child package, If the package protected by a password (Which is common in real-time), then please provide the password for that child package.

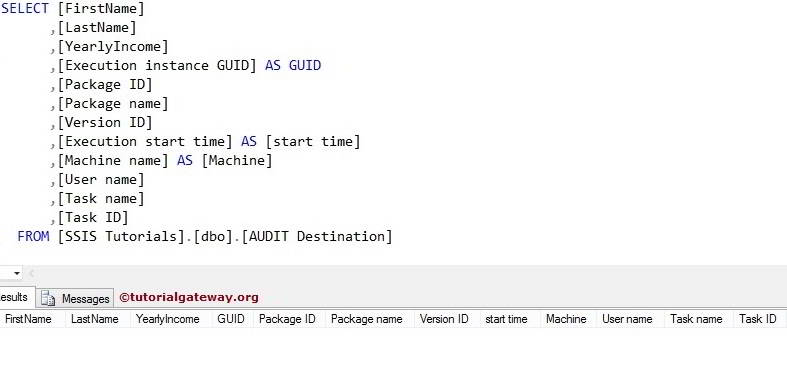
**ExecuteOutOfProcess:**Specify whether you want your child package to run in the same process (with the parent package) or the separate process. If you set this option to False, both parent package and child package will run in the same process. If you set this option to True, both parent package and child package will run in a different process.

# Execute Packages in File System using SSIS Execute Package Task

The Execute Package Task allows us to call other packages present in the SQL Server or File System as a part of its execution. In this article, we are executing Packages present in the File System using SSIS Execute Package Task.

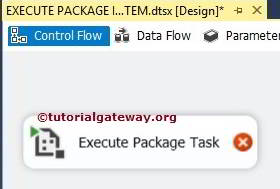
In this example, we are going to execute SSIS Audit Transformation Package present in the File System using Execute Package Task. We already explained about this [Audit Transformation](https://www.tutorialgateway.org/audit-transformation-in-ssis/) package in our previous article

You can see from the below screenshot, [Audit Destination] table is empty. If not, Please truncate the table using T-SQL or add Execute SQL Task.

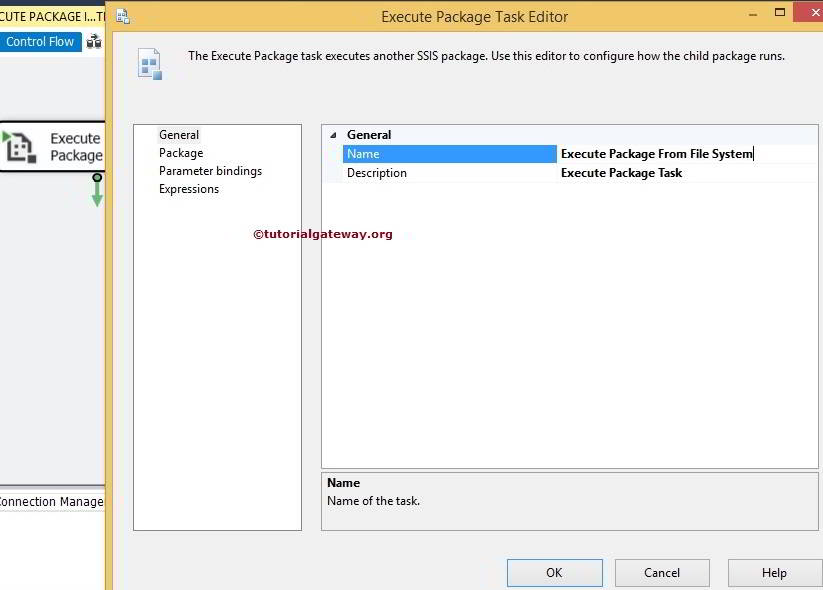


## Execute Packages in File System using SSIS Execute Package Task Example

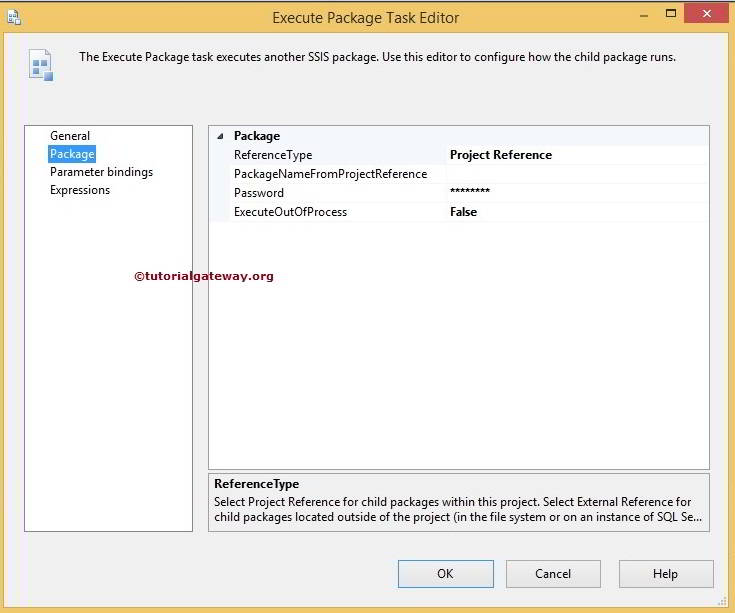
Drag and drop the [Execute Package Task](https://www.tutorialgateway.org/execute-package-task-in-ssis/) from the toolbox to the Control Flow Region.



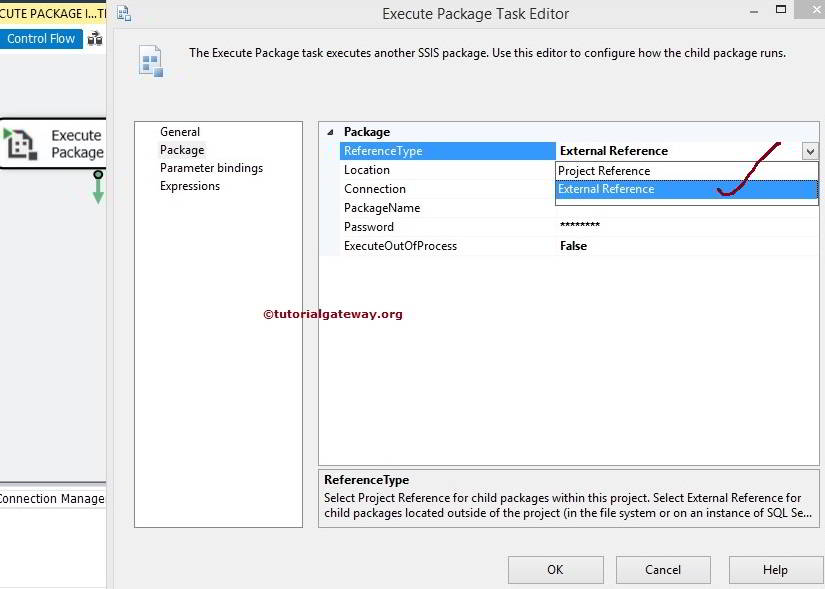
Double click on the Execute Package Task to configure the package. In the General tab, we can change the name and description. As of now, we changed the Name as Execute Package Task from File System and left the description as it is.



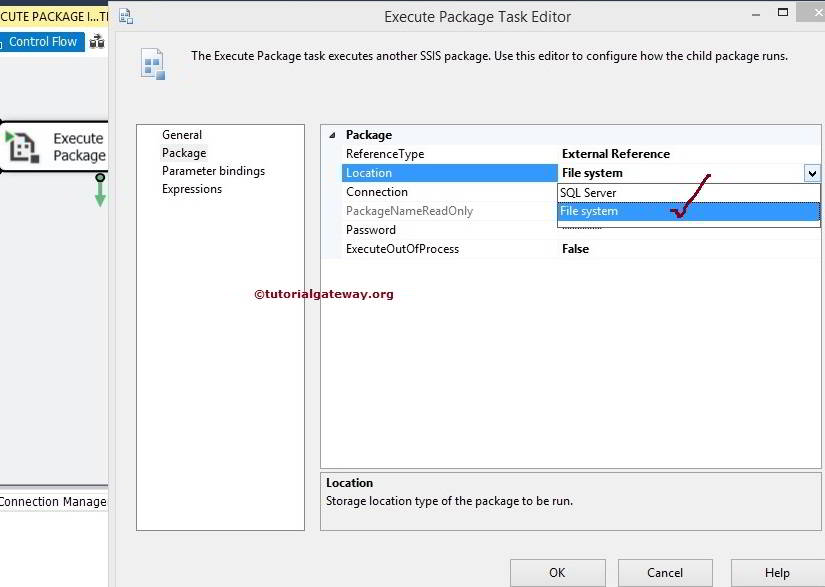
Click on the package tab to configure the package location and connection string



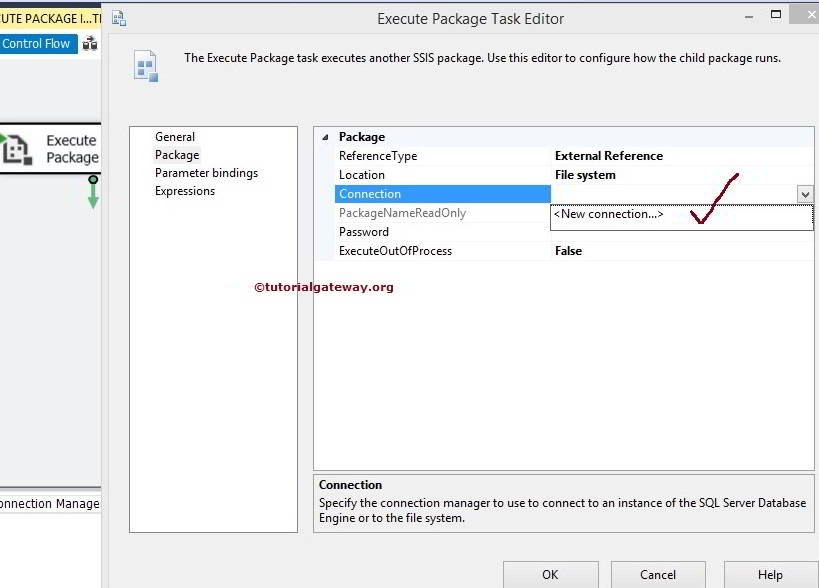
In this example, We are executing the package located in the file system. So, Please change the **ReferenceType** from Project Reference to External reference. Please refer [SSIS Execute Package Task Project Reference](https://www.tutorialgateway.org/ssis-execute-package-task-project-reference/) article for calling packages inside the same project and [Execute Packages in SQL Server using SSIS Execute Package Tas](https://www.tutorialgateway.org/execute-packages-in-sql-server-using-ssis-execute-package-task/)k for calling packages in SQL Server.



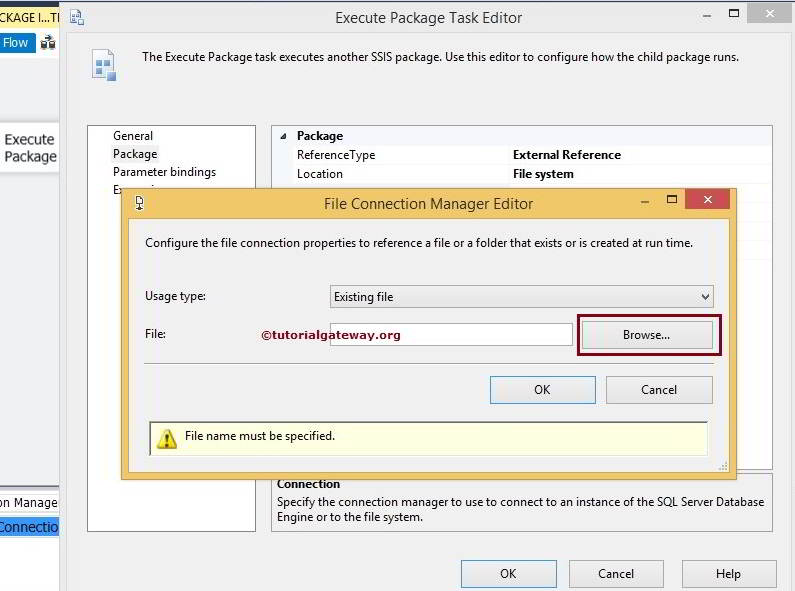
Here let us select the file system as a package source



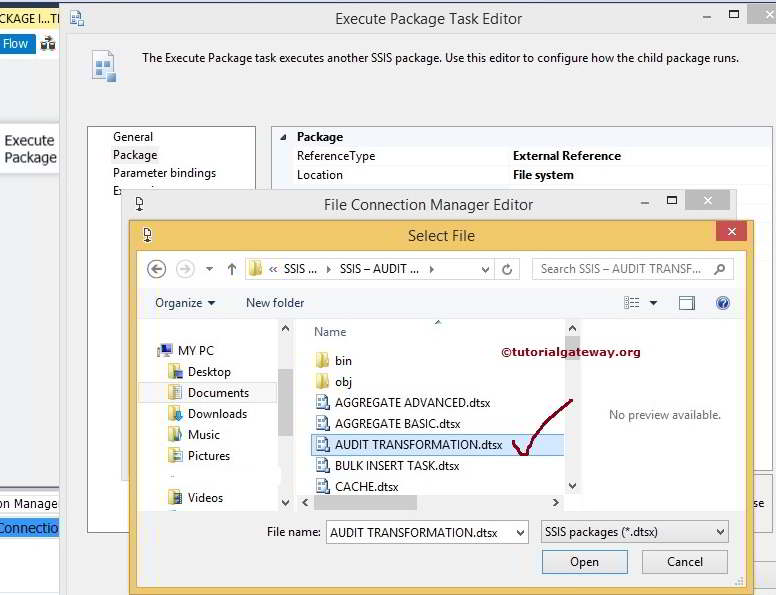
Click on the New Connection to configure or select the file from the file system.



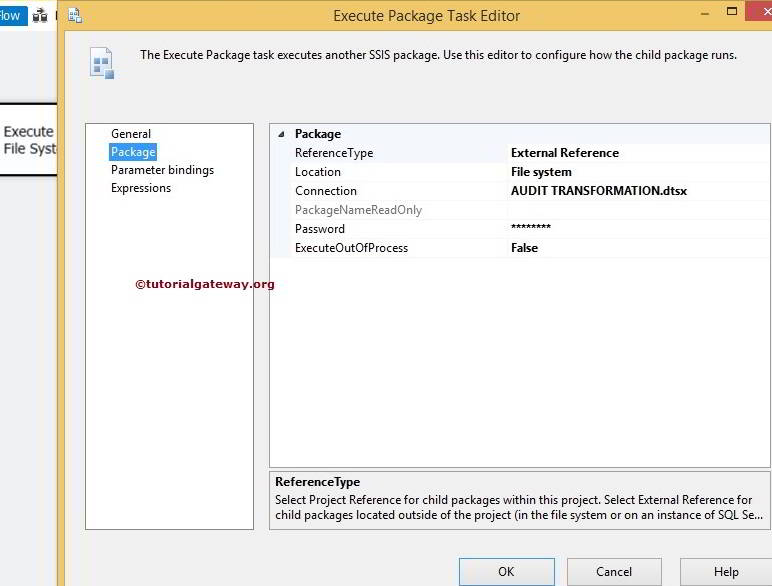
Once you click on the <New Connection…>, it will open the **File Connection Manager Editor** to configure the file in the file system.



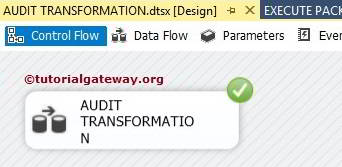
From the below screenshot you can observe, we are selecting AUDIT TRANSFORMATION from the list of [SSIS](https://www.tutorialgateway.org/ssis/) Packages.



Click ok to select the file. If the Audit Transformation package secured with a password (In general, Yes), provide an appropriate password.



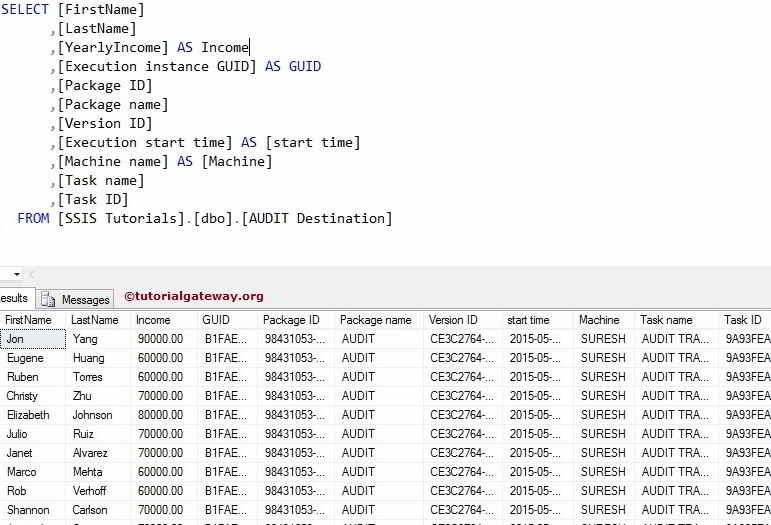
Click ok to finish configuring the Execute Package Task. Let us Run the Package and see, whether we successfully called the package present in the file system using the Execute Package Task or not.



From the above, you can observe that our Execute Package Task is calling Audit Transformation. After this Transformation is successful, the Execute Package Task will execute.



Let us open the [SQL Server Management Studio](https://www.tutorialgateway.org/sql/) and check the result.

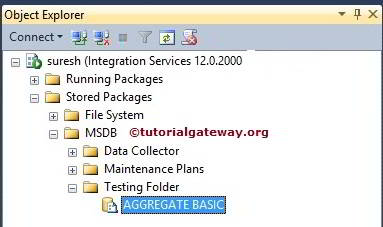


# Execute Packages in SQL Server using SSIS Execute Package Task

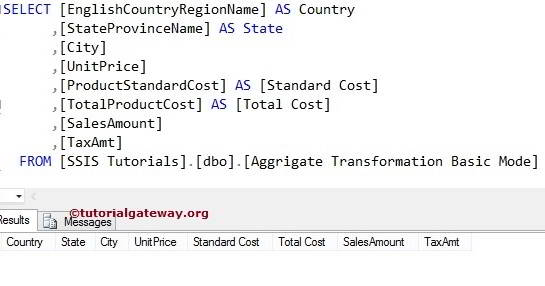
The [Execute Package Task](https://www.tutorialgateway.org/execute-package-task-in-ssis/) allows us to call other packages present in SQL Server, File System, and Packages present in the same project as a part of its execution. In this article, we are executing Packages present in the SQL Server using SSIS Execute Package Task.

In this example, we are going to execute the SSIS Aggregate Transformation Package stored inside the SQL Server using the Execute Package Task. We already explained this [Aggregate Transformation in Basic Mode](https://www.tutorialgateway.org/aggregate-transformation-in-ssis/). So, Please refer to it.

From the below screenshot, you can see that we already published the Aggregate Transformation in the MSDB database.

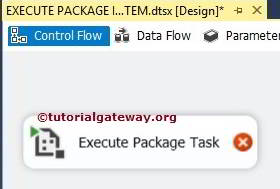


You can see from the below screenshot, [Aggregate Transformation Basic Mode] table is empty. If not, Please truncate the table using T-SQL or add Execute SQL Task.

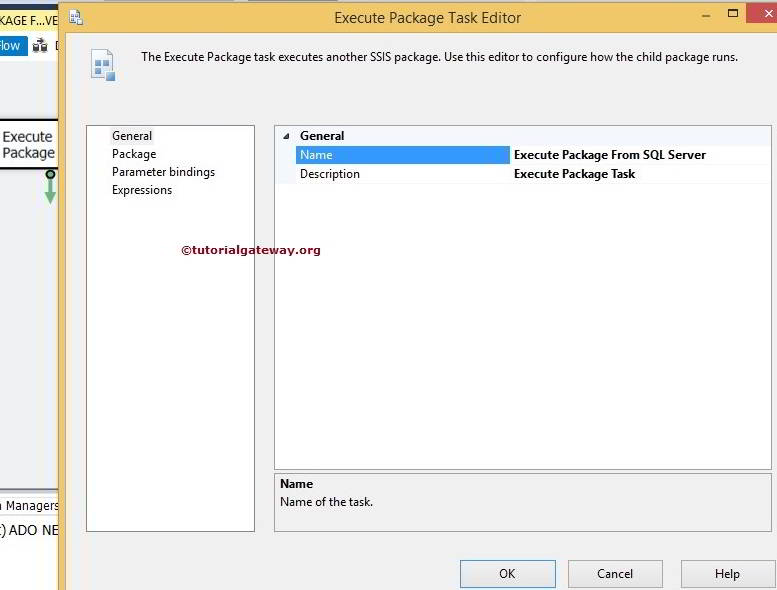


## Execute Packages in SQL Server using SSIS Execute Package Task Example

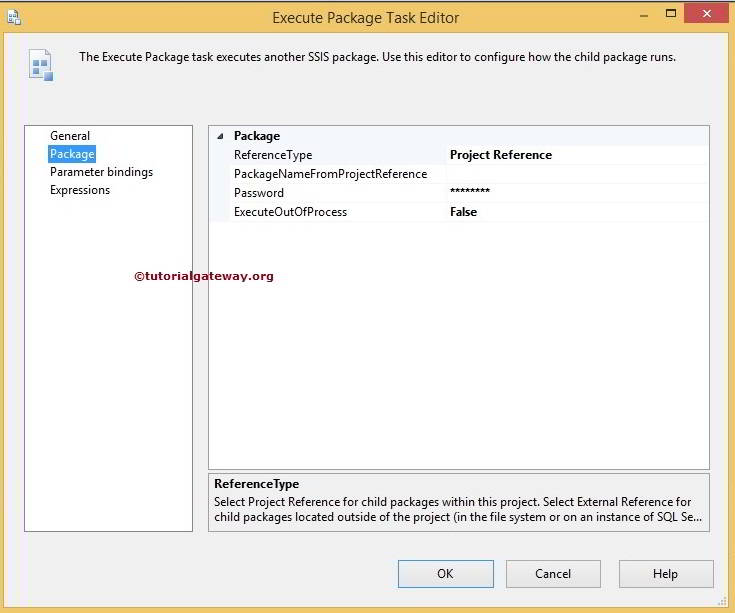
Drag and drop the Execute Package Task from the toolbox to Control Flow Region.



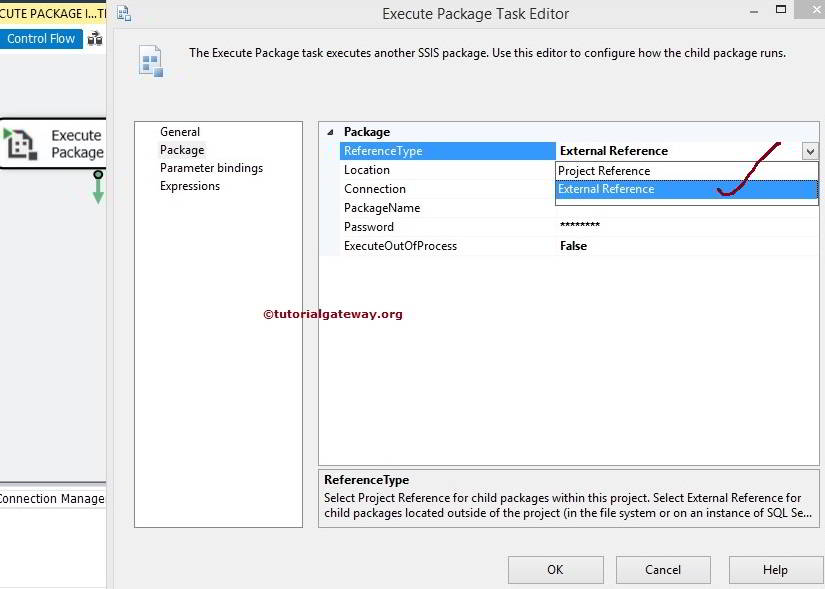
Double click on the Execute Package Task to configure the package. In the General Tab, we can change the name and description. As of now, we changed the Name as Execute Package Task from [SQL Server](https://www.tutorialgateway.org/sql/).



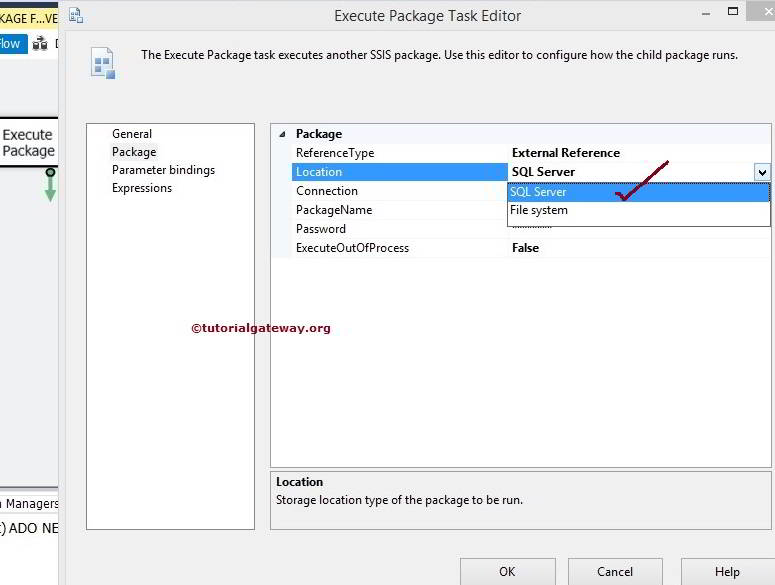
Click on the package tab to configure the package location and connection string



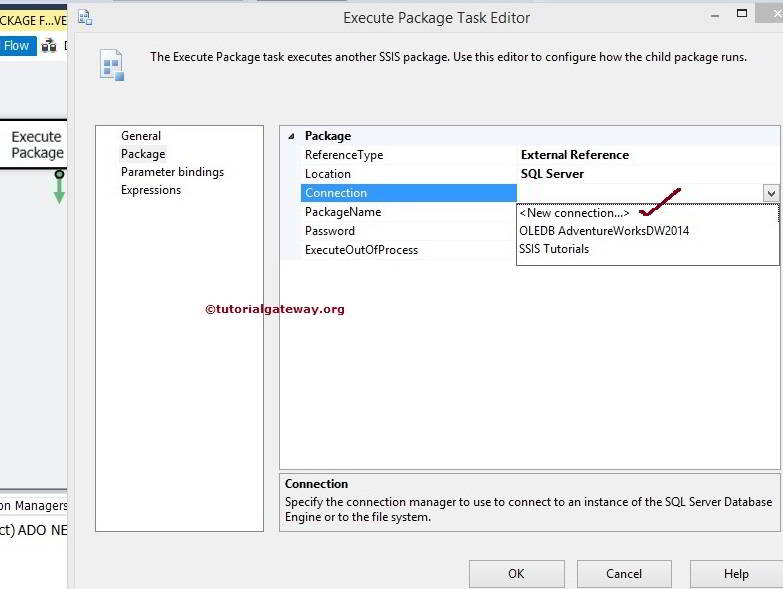
In this example, We are executing the package located in the SQL Server so, Please change the **ReferenceType** from Project Reference to External reference. Please refer [SSIS Execute Package Task Project Reference](https://www.tutorialgateway.org/ssis-execute-package-task-project-reference/) article for calling packages inside the same project and [Execute Packages in File System using SSIS Execute Package Task](https://www.tutorialgateway.org/execute-packages-in-file-system-using-ssis-execute-package-task/) for calling packages in File System



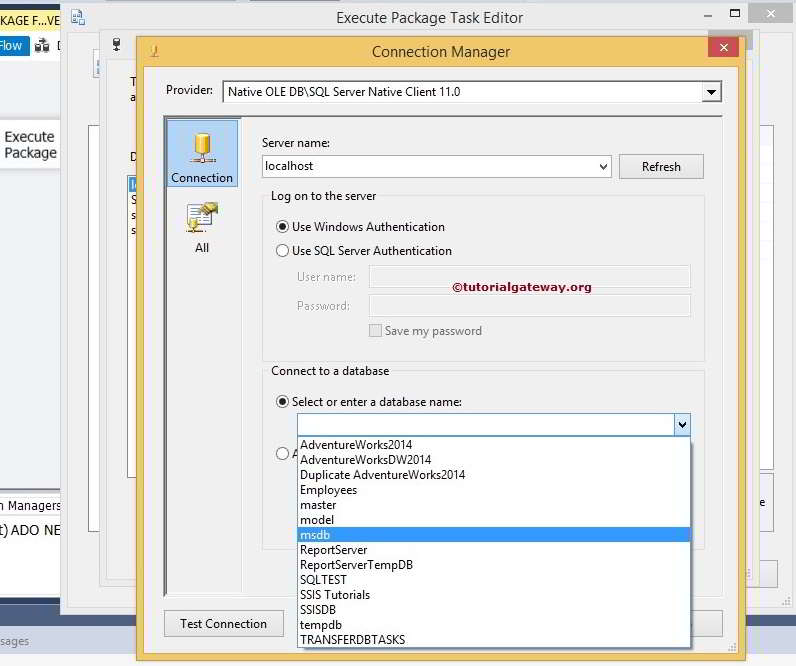
Here let us select the SQL Server as a package source



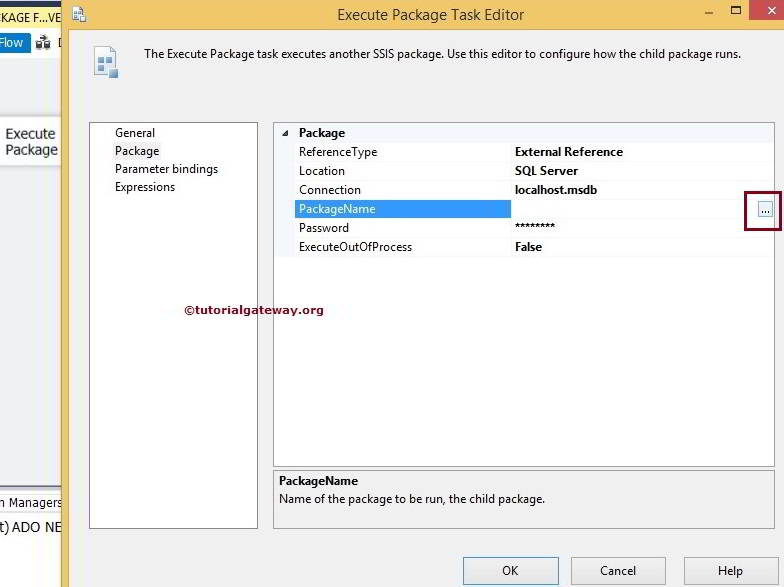
Click on the New Connection to configure or select the Server settings. If you already created the connection before then, it will display the list. You can choose the required connection. For now, click on the <New Connection…>



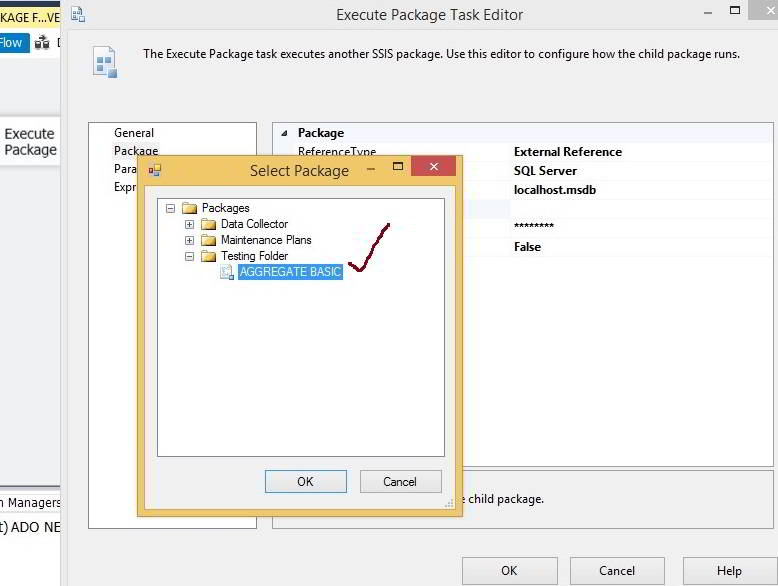
Once you click on the <New Connection…>, it will open the **Connection Manager Editor** to select the Provider Name, Server Name, and Database Name.



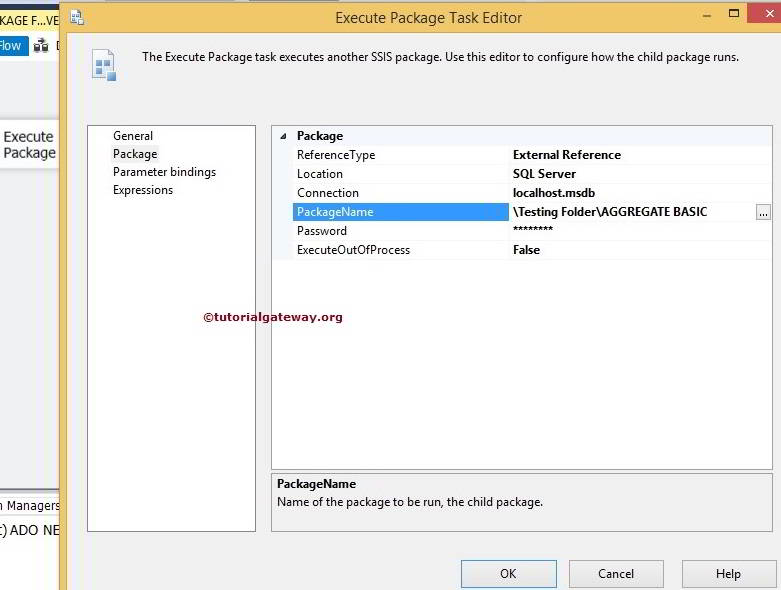
Click ok to finish selecting the Database. Now we have to select the package we want to execute. So, click on the Browse button (**…**) beside the **PackageName** option



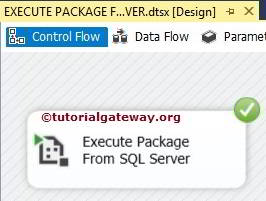
It will open the **Select Package** window to select the required package. From the below screenshot, you can observe, we are selecting AGGREGATE BASIC from the list. We have only one package at the moment. So it is displaying one, but in real-time, it may be more.



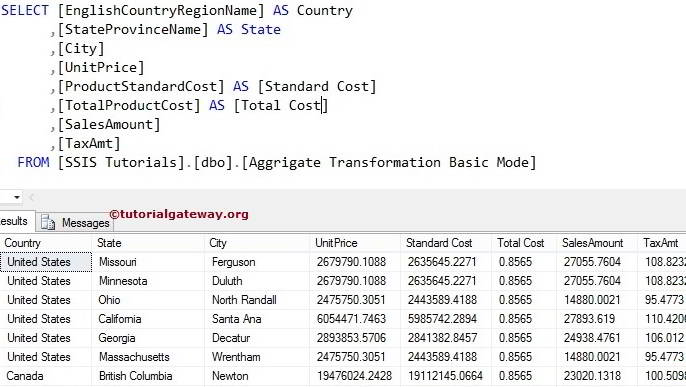
Click ok to select the package. If the Aggregate Transformation package secured with a password (In general, Yes), enter an appropriate password.



Click ok to finish configuring the [SSIS](https://www.tutorialgateway.org/ssis/) Execute Package Task. Let us Run the Package and see whether we successfully called the package present in the SQL Server using Execute Package Task or not.



It seems our Execute Package Task not thrown any errors. Let us open the [SQL Server Management Studio](https://www.tutorialgateway.org/sql/) and check the result.



Well, we successfully called the package present in the SQL Server using SSIS Execute Package Task.

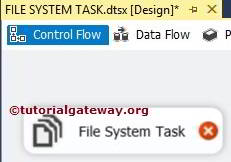
# File System Task in SSIS

The File System Task in SSIS is used to perform a different kinds of operations on Files and Folders (or Directories). For instance, if you want to move the directory content from one location to another, we can use this SSIS File System Task.

**NOTE:**File System Task in SSIS uses the [File Connection Manager](https://www.tutorialgateway.org/file-connection-manager-in-ssis/) to connect with the Files and Folders.

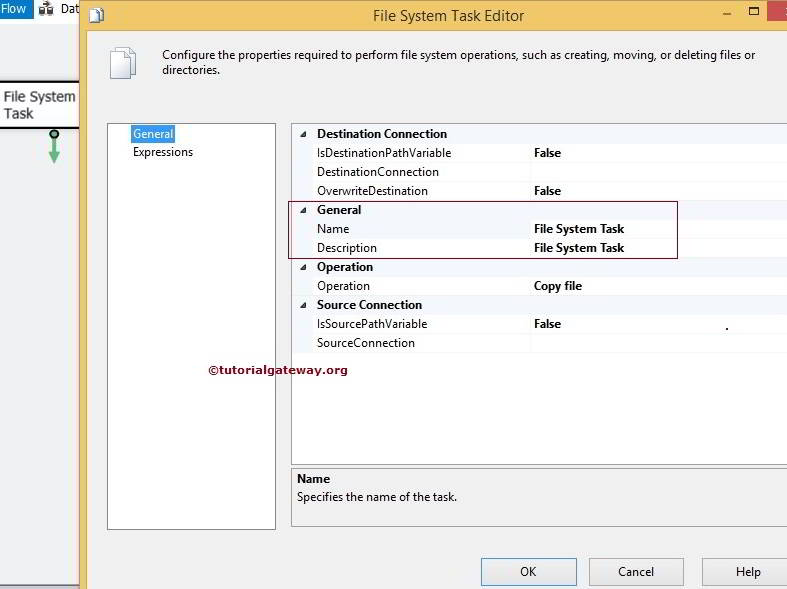
## Configuring File System Task in SSIS

Drag and drop the SSIS File System Task into the Control Flow region

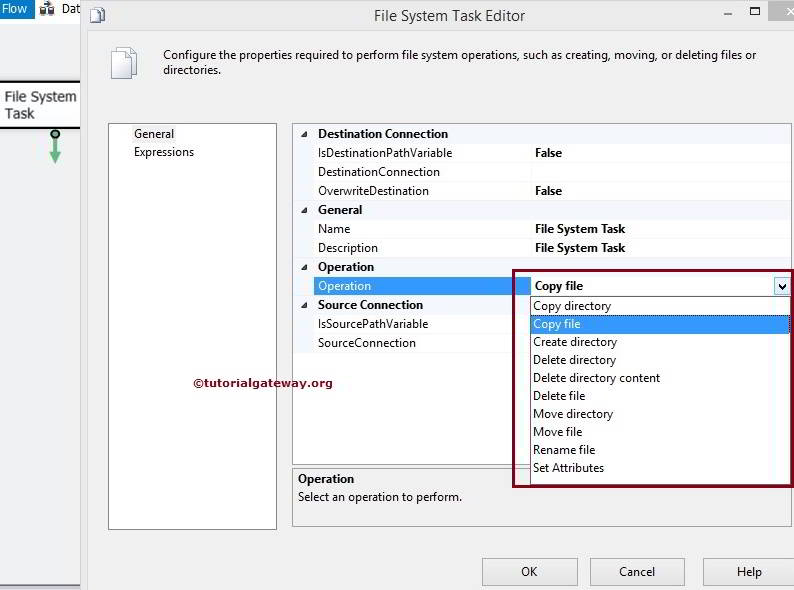


Double click on it will open the SSIS File System Task Editor to configure it.

* **Name:** Please provide the Unique Name
* **Description:** Briefly describe the File system task Functionality.



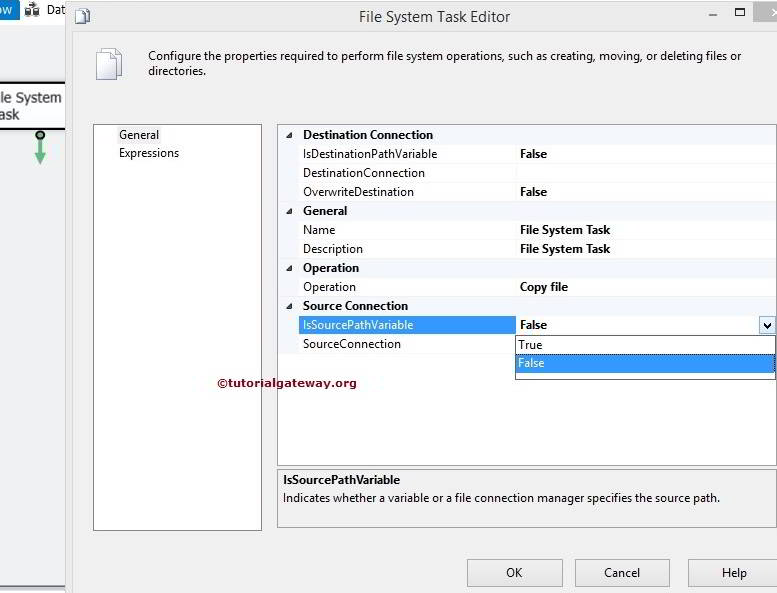
The File System Task in SSIS supports ten different operations on Files and Folders. Please see the available options in below [SSIS](https://www.tutorialgateway.org/ssis/) screenshot



|  |  |
| --- | --- |
| **OPERATION** | **DESCRIPTION** |
| Copy Directory | Copies a folder from one location to another location. Please refer to [Copy Directory Using File System Task](https://www.tutorialgateway.org/copy-directory-using-file-system-task-in-ssis/) article to understand the steps involved in Copying the Directory |
| Copy File | Copies a file from one location to another location. Please refer to [Copy File Using File System Task](https://www.tutorialgateway.org/copy-files-using-file-system-task-in-ssis/) article to understand the steps involved in Copying the Files |
| Create Directory | Using this option, we can create the directory at the specified location. |
| Delete Directory | Deletes the selected folder present in the specified location. Please refer to [Delete Directory Using File System Task](https://www.tutorialgateway.org/delete-directory-using-file-system-task-in-ssis/) article to understand the steps involved in Deleting the Directory |
| Delete Directory Content | It Deletes the content present in a selected folder. Please refer to [Delete Directory Content Using File System Task](https://www.tutorialgateway.org/delete-directory-content-using-file-system-task-in-ssis/) to understand the steps involved in Deleting the Directory Content. |
| Delete File | Deletes the selected file present in the specified location. Please refer [Delete File Using File System Task](https://www.tutorialgateway.org/delete-file-using-file-system-task-in-ssis/) to know the steps involved in Deleting the Files |
| Move Directory | Moves a folder from one location to another location. Please see [Move Directory Using File System Task](https://www.tutorialgateway.org/move-directory-using-file-system-task-in-ssis/) article to understand the steps involved in Moving the Directory |
| Move File | It Moves a file from one location to another location. Please see [Move File Using File System Task](https://www.tutorialgateway.org/move-file-using-file-system-task-in-ssis/) article to know the steps involved in Moving Files |
| Rename File | Moves a file from one location to another location and renames the file name. Please refer [Rename File Using File System Task](https://www.tutorialgateway.org/rename-file-using-file-system-task-in-ssis/) to understand the steps involved in Renaming Files |
| Set Attributes | Using this option, we can set the attributes for the files and folder. This option includes (Hidden, ReadOnly, System and Archive options). Please refer to [Setting Attributes Using File System Task](https://www.tutorialgateway.org/setting-attributes-using-file-system-task-in-ssis/) article to know the steps involved in Setting or altering the attributes of files and folders. |

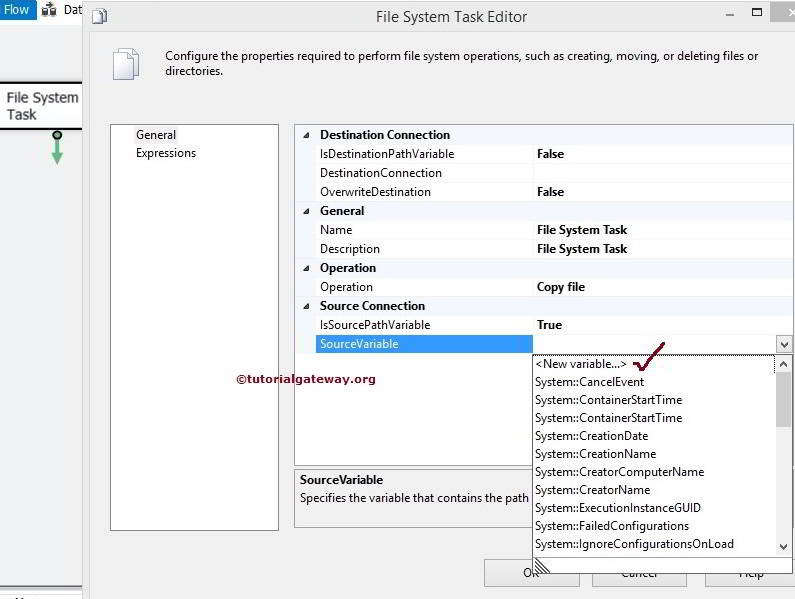
### SSIS File System Task Source Connection

**IsSorcePathVariable:**This property has two options: True and False. If we set this property to true, then the source path is stored in a variable. And, if we set this property to false, we have to select the source path manually using File Connection Manager.

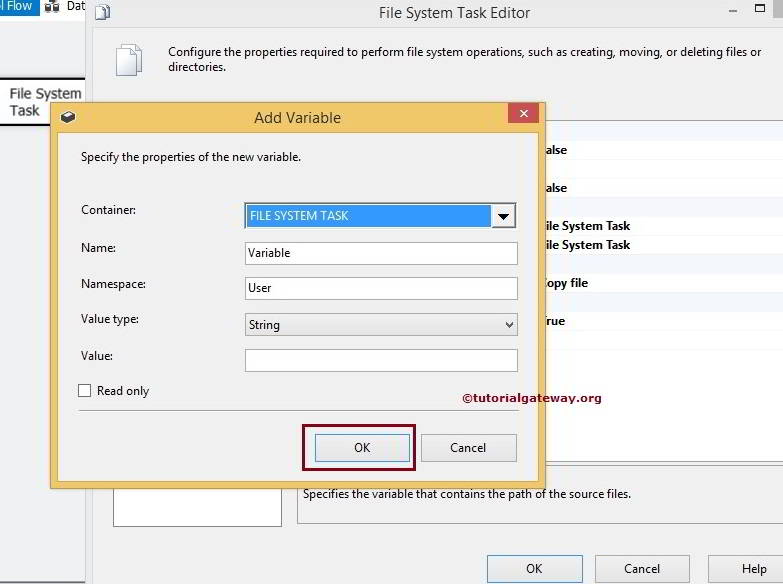


If we set the **IsSorcePathVariable** to true, a new property called **SourceVariable** appeared.

**SourceVariable:**This SSIS File System Task property displays all the available system and user variables. Please select the appropriate variable which is holding the Source Path (File or Folder path). If you haven’t created any variable before, please click on the **<New Variable..>**.

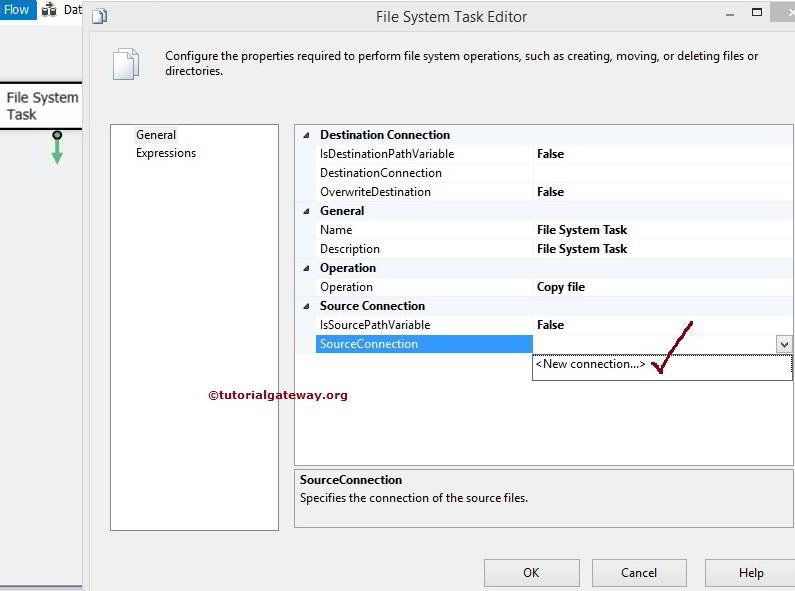


Once you click on the **<New Variable..>,**it will open the new window called Add Variable to create the new variable. Here you can create the new variable which will hold the source path.



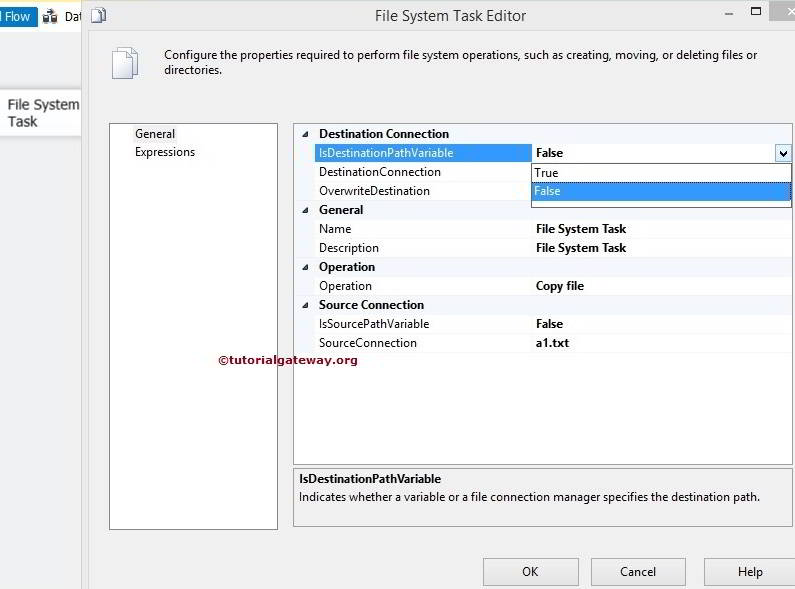
And If we set the **IsSorcePathVariable** False, configure the Source Connection using **SourceConnection**Property. If you already created the [File Connection Manager](https://www.tutorialgateway.org/file-connection-manager-in-ssis/), select the same from the drop-down list.

If you haven’t created any connection Manager before, You have to create by selecting **<New Connection..>**.



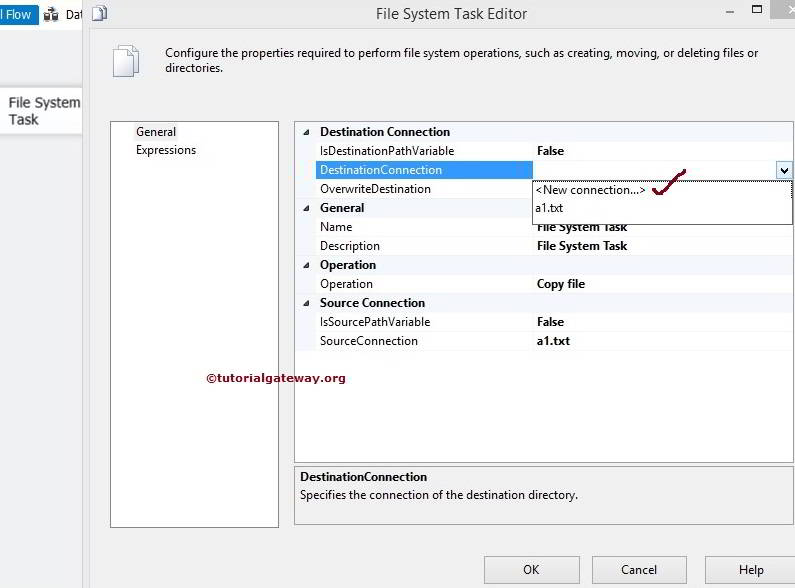
### Config Destination Connection of File System Task in SSIS

**IsDestinationPathVariable:**This SSIS File System Task property has two options: True and False. If we set this property to true, the Destination path stored in a variable. If we set this property to false, select the Destination path manually using File Connection Manager.

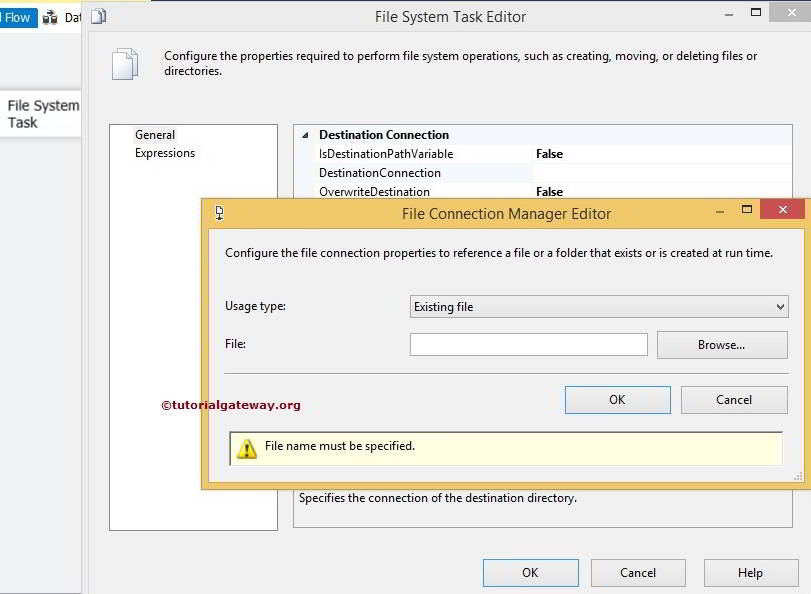


For instance, If we set the **IsDestinationPathVariable** False, we have to configure the Destination Connection using **DestinationConnection**Property. If you previously created the File Connection Manager, you can pick it from the drop-down list.

If you haven’t created any connection Manager before, You have to create by selecting **<New Connection..>**.

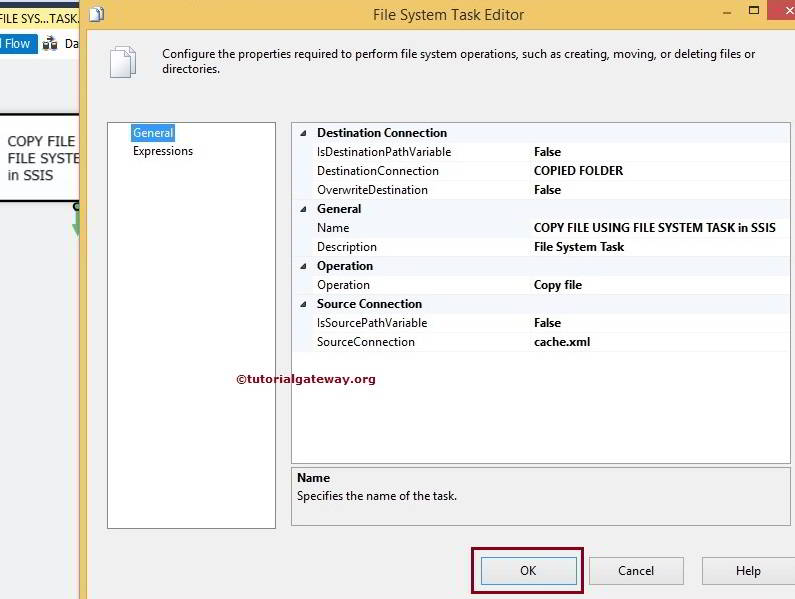


Once you click on the **<New Connection..>** option, a new window called File Connection Manager Editor opened. Using this, we have to configure the Destination Connection.



Please refer to the [File Connection Manager](https://www.tutorialgateway.org/file-connection-manager-in-ssis/) article to understand the configuration.

**OverwriteDetination:**This SSIS File System Task property has two options: True and False. If we set this property to true, the File System Task overwrite the existing files in the Destination path.



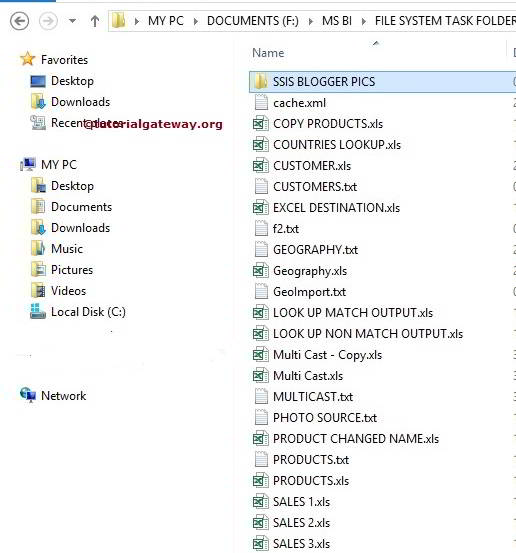
Click OK to finish configuring SSIS File System Task Editor.

# Copy Files Using File System Task in SSIS

In this article, we show you how to Copy Files Using File System Task in SSIS with example. To Copy the complete Directory, Please refer to [Copy Directory Using File System Task](https://www.tutorialgateway.org/copy-directory-using-file-system-task-in-ssis/) article.

**TIP:** In [SSIS](https://www.tutorialgateway.org/ssis/), Please use the [Foreach Container](https://www.tutorialgateway.org/ssis-foreach-loop-file-enumerator/) to Copy multiple files from one location to another location.

We have File System Task Folder inside the MSBI Folder. Our task is to Copy the cache.xml file inside the File System Task Folder to **Copied Folder** inside F Drive.

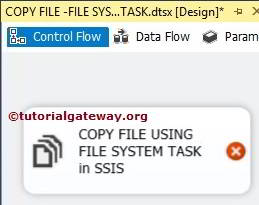


Files inside the Copied folder are:



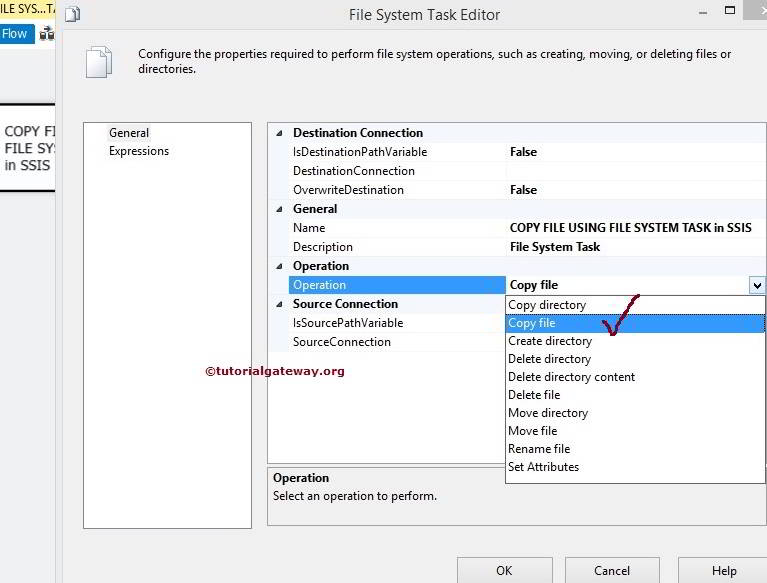
## Copy Files Using File System Task in SSIS

To Copy Files Using File System Task in SSIS, First Drag and drop the File System Task into the Control Flow region and rename it as Copy File Using File System Task in SSIS



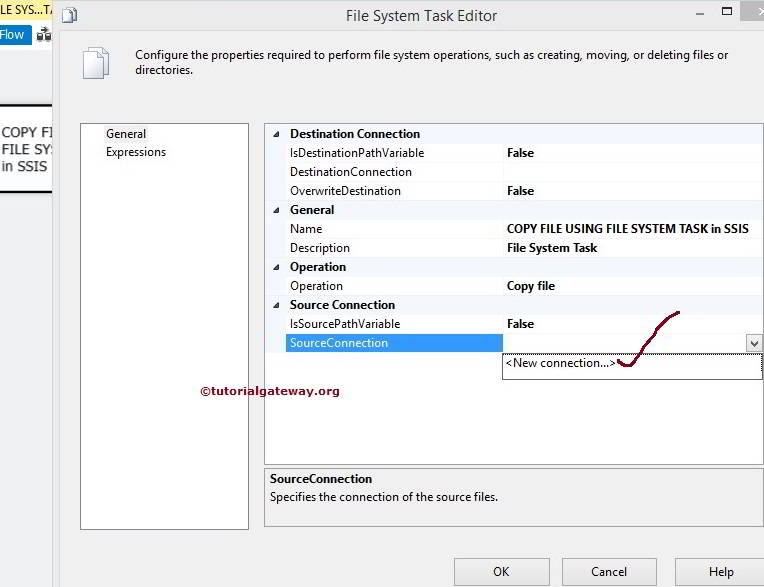
Double click on it will open the [File System Task](https://www.tutorialgateway.org/file-system-task-in-ssis/) Editor to configure it.

In this example, we are Copying a single file. So, change the **operation** property to **Copy File**



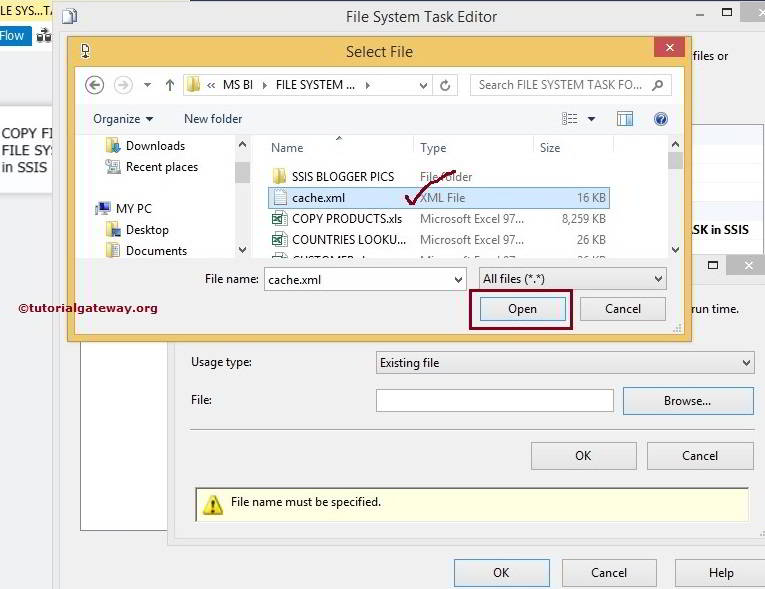
Let us configure the Source Connection by selecting the **SourceConnection**property. If you already created the File Connection Manager before then select the created one or If you stored the Source Connection in the Variable then, please change the **IsSourcePathVariable**property to TRUE and select the Variable Name.

Here, we are selecting **<New Connection..>**.

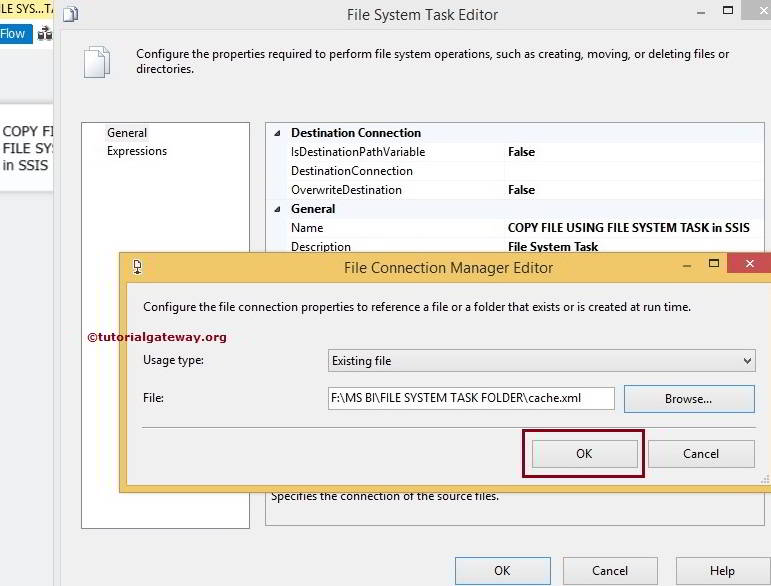


Once you click on the **<New Connection..>**option, File Connection Manager Editor opened to configure. Here, we are Copying the existing file, so select the Existing File option from the**Usage Type.**

Click on the Browse button to select the Existing File from the file system.

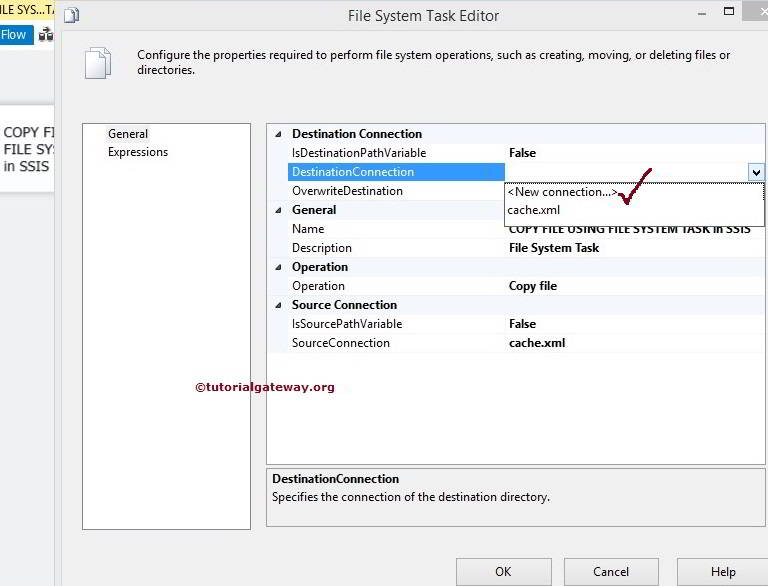


From the above screenshot, you can see we selected the Cache.xml file

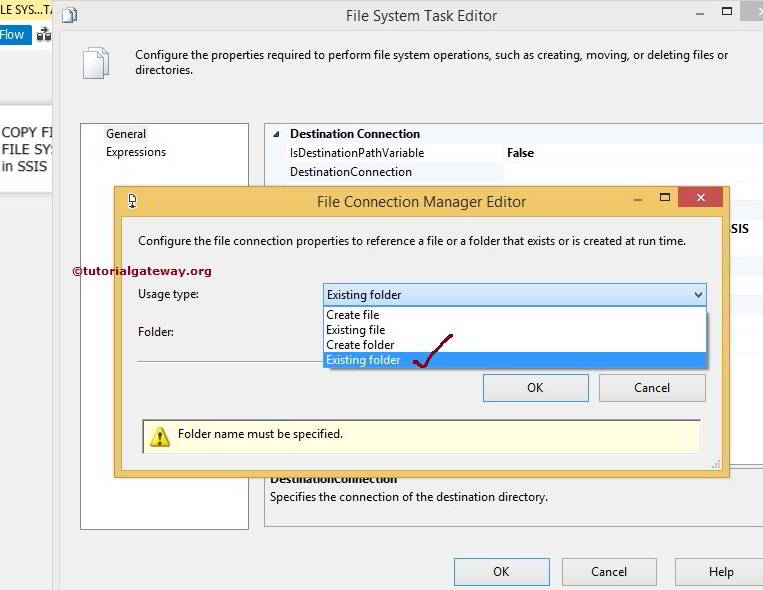


Click Ok to finish configuring the Source connection. Now we have to set the Destination Connection. So, select the **DestinationConnection**property. If you have already created the File Connection Manager before, select it. Or If you saved the Destination Connection in a Variable, change the **IsDestinationPathVariable**field to TRUE and select the Variable Name.

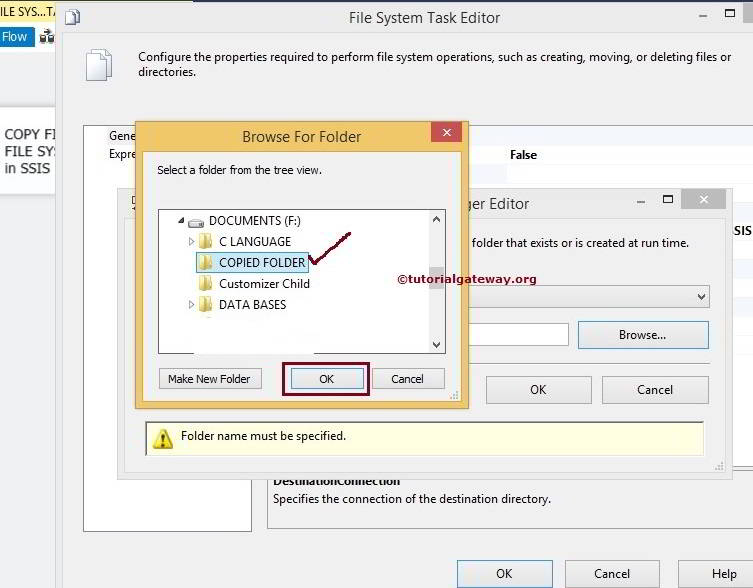
We don’t have any connection Manager. So, We are selecting **<New Connection..>**.



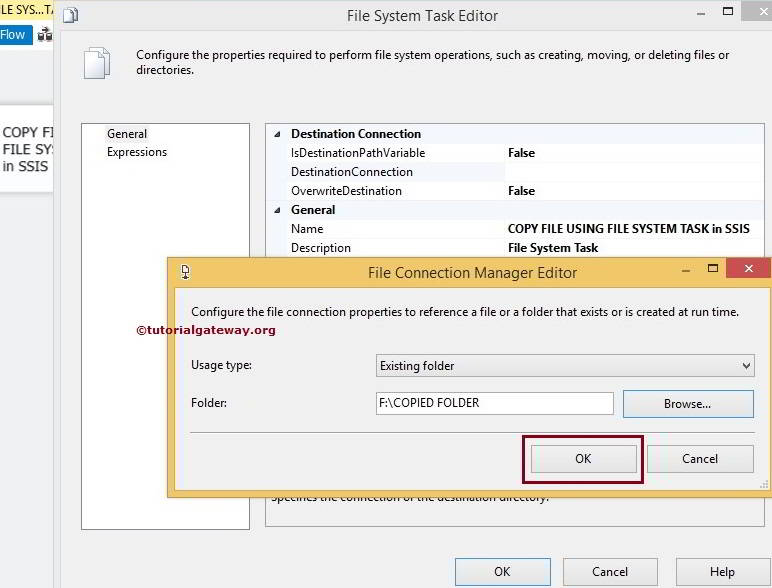
In this example, we are Copying the existing file to an already existing folder. So, select the Existing Folder option from the**Usage Type.**



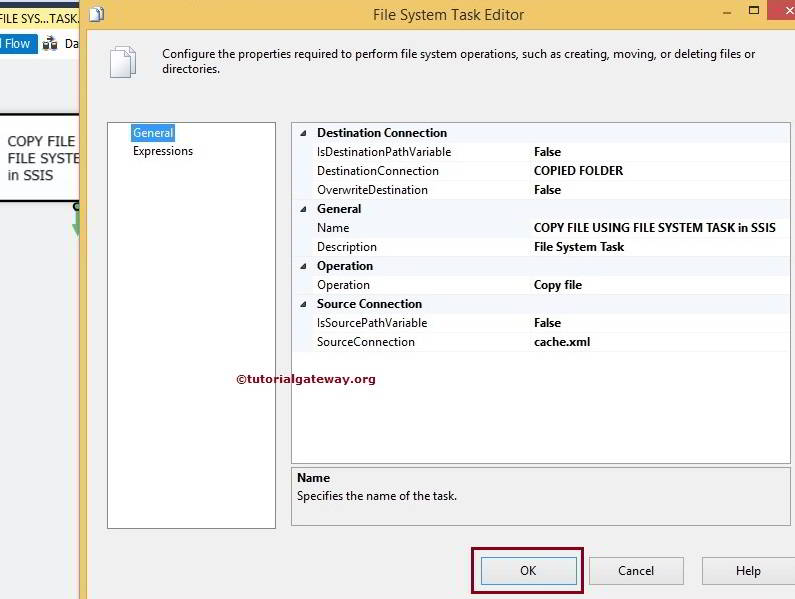
Click on the Browse button to select the Existing Folder from the file system.



From the preceding screenshot, you can observe that we selected the Copied Folder as the destination folder. Click the Ok button to select it.



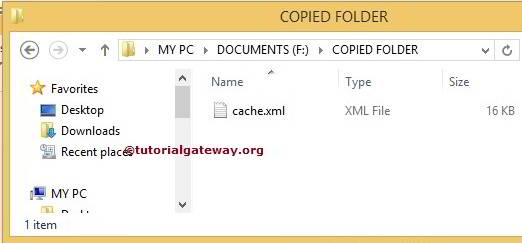
Click Ok to finish configuring the File Connection Manager for the destination.



Next, Click Ok to finish configuring the Copy Files Using File System Task in SSIS package. Let’s run and see whether we successfully Copied the Cache.xml file using the File System Task or Not.



Well, We successfully Copied the Cache.xml file present in the File System Task Folder to the Copied Folder.

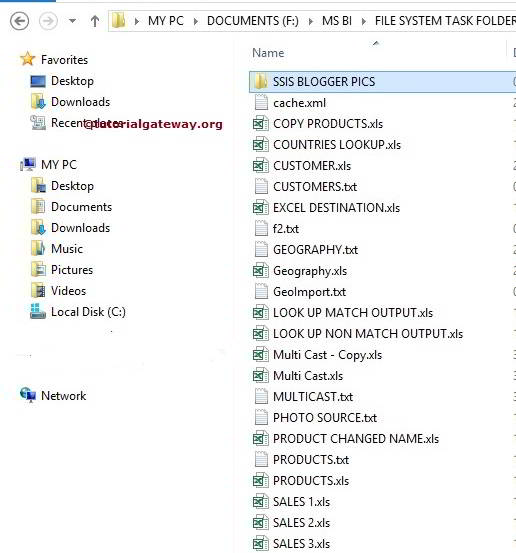


# Move File Using File System Task in SSIS

In this article, we show you how to Move Directory Using File System Task in SSIS from one location to another location with example. To delete the complete Directory, Please refer to [Delete Directory Using File System Task](https://www.tutorialgateway.org/delete-directory-using-file-system-task-in-ssis/) article in the [SSIS](https://www.tutorialgateway.org/ssis/) page.

**TIP:** Please use the [Foreach Container](https://www.tutorialgateway.org/ssis-foreach-loop-file-enumerator/) to move multiple files from one location to another location. You can refer to [Move Multiple files using File System Task](https://www.tutorialgateway.org/move-multiple-files-using-file-system-task-in-ssis/) for the practical example.

We have File System Task Folder inside the MSBI Folder. The below screenshot shows you the data inside that folder. Our SSIS task is to move the CUSTOMERS.txt file inside the File System Task Folder to **Copied Folder** inside F Drive.

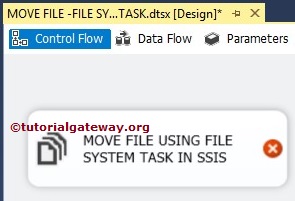


Files inside the Copied folder are:



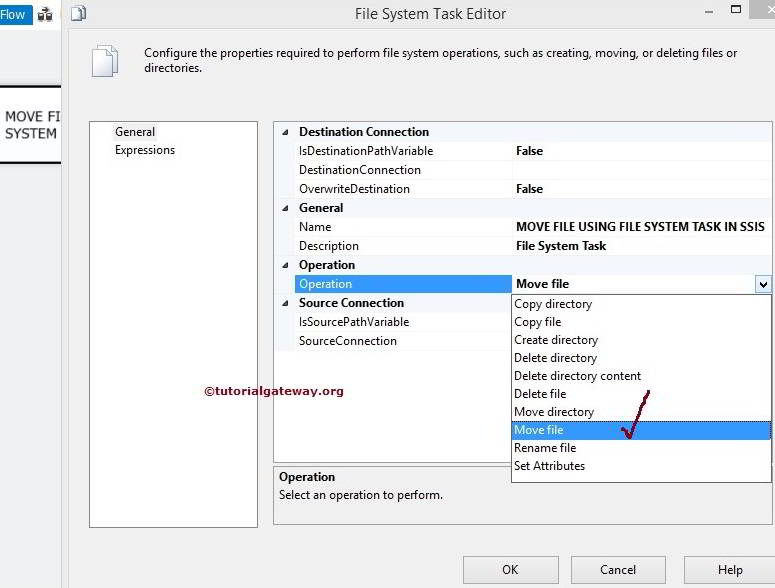
## Move File Using File System Task in SSIS

To move file using file system task in ssis, Please Drag and drop the File System Task into the Control Flow region and rename it as Move File Using File System Task in SSIS



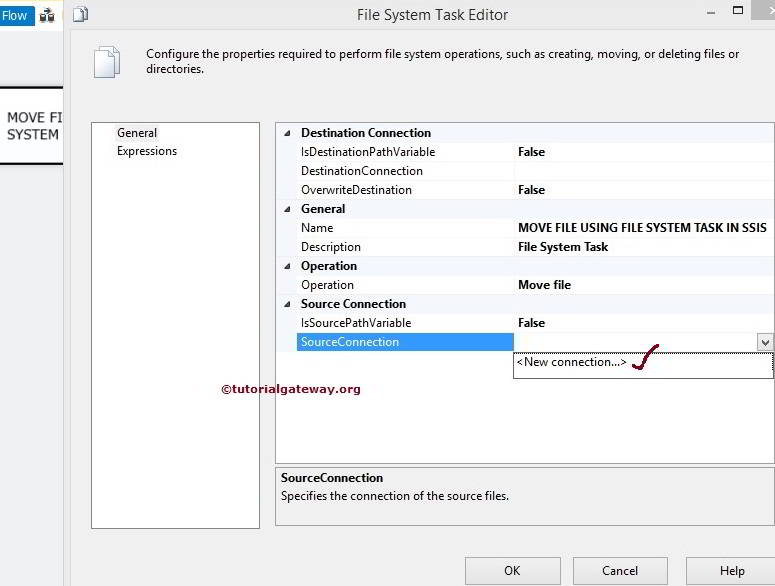
Double click on it will open the [File System Task](https://www.tutorialgateway.org/file-system-task-in-ssis/) Editor to configure it.

In this example, We are Moving a single file. So, please change the **operation** property to **Move File**as shown in the below screenshot



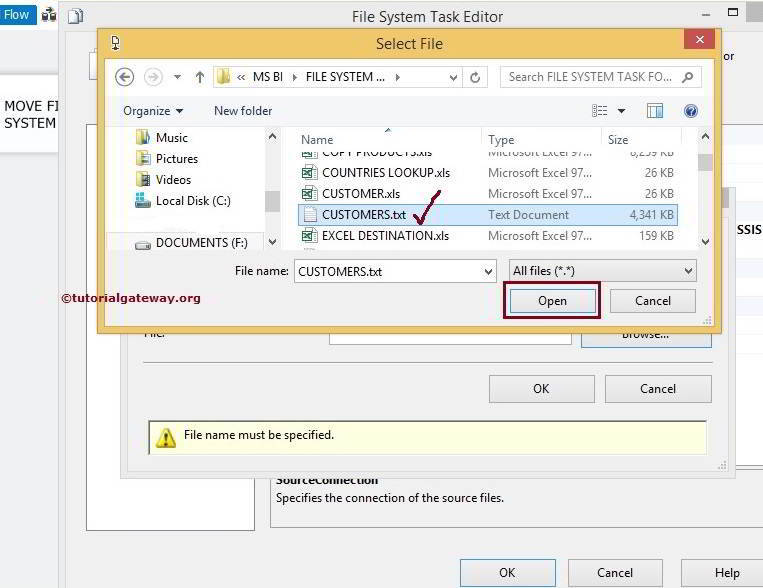
Let us configure the Source Connection by selecting the **SourceConnection**property. If you already created the File Connection Manager, select the created one. Or If you saved the Source Connection in a Variable, please change the **IsSourcePathVariable**property to TRUE and select the Variable Name.

Here, We haven’t created any connection Manager before so, We are selecting **<New Connection..>**.

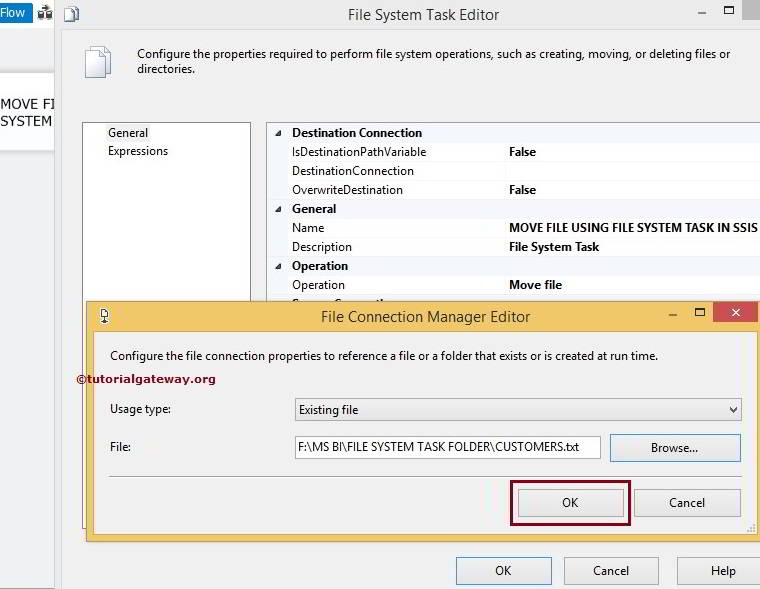


Once you click on the **<New Connection..>**option, File Connection Manager Editor will open to configure it. Here, we are Moving the existing file, so select the Existing File option from the**Usage Type.**

Click on the Browse button to select the Existing File from the file system.



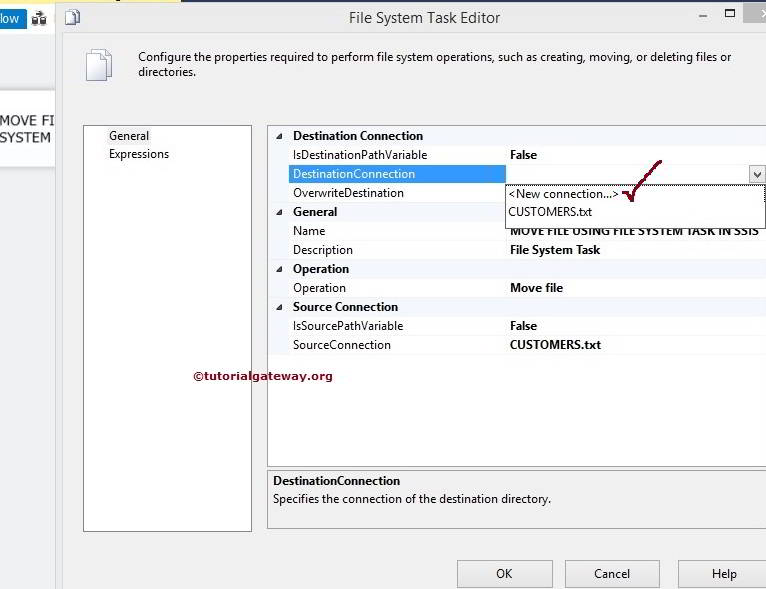
From the above screenshot, you can see we selected the CUSTOMERS.tx file inside the File System Task Folder



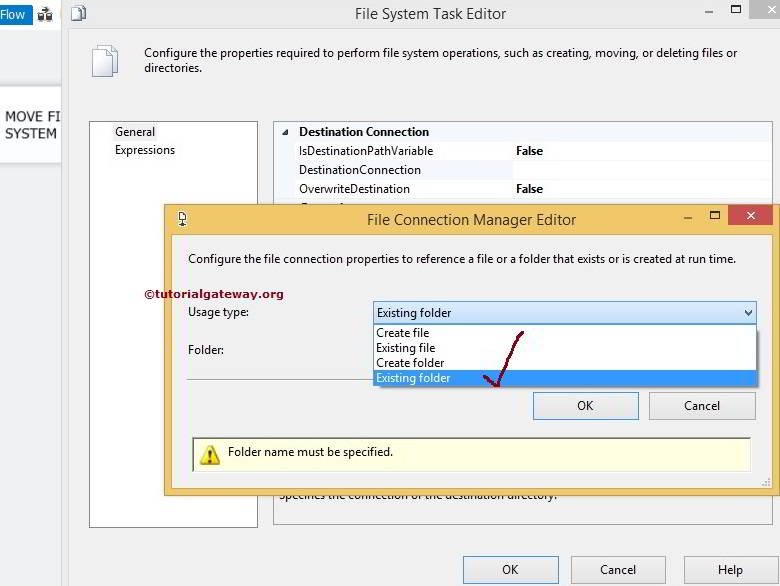
Click Ok to finish configuring the Source connection.

Now we have to configure the Destination Connection so, Please select the **DestinationConnection**property. If you already created the File Connection Manager before then select the created one or If you stored the Destination Connection in the Variable then, please change the **IsDestinationPathVariable**property to TRUE and select the Variable Name.

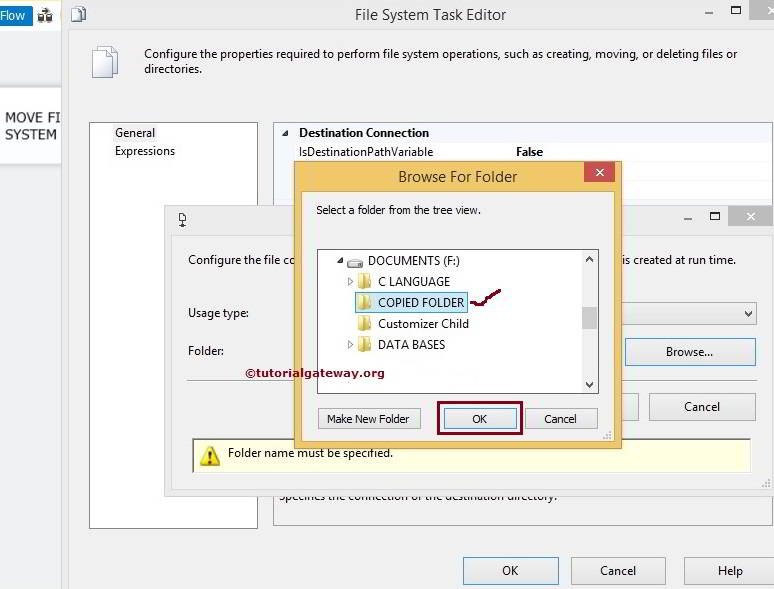
Here, We haven’t created any connection Manager before. So, we are selecting **<New Connection..>**.



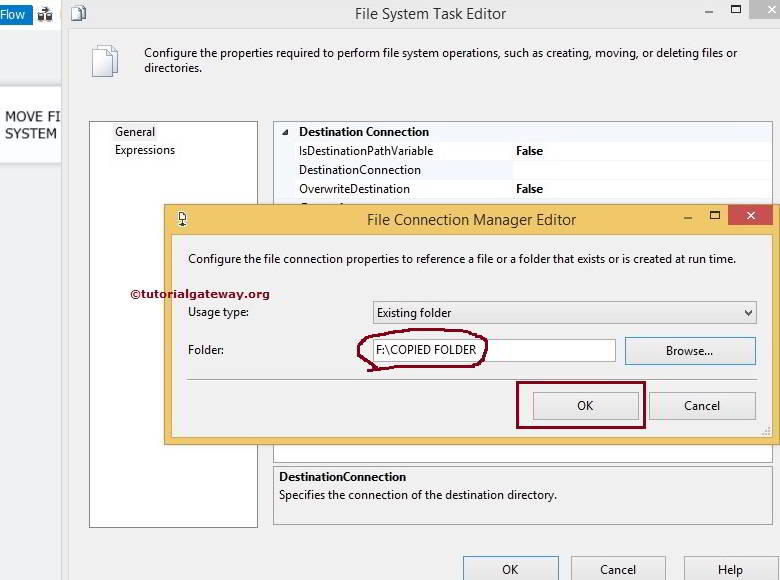
In this example, we are Moving the existing file to an already existing folder. So, select the Existing Folder option from the**Usage Type.**



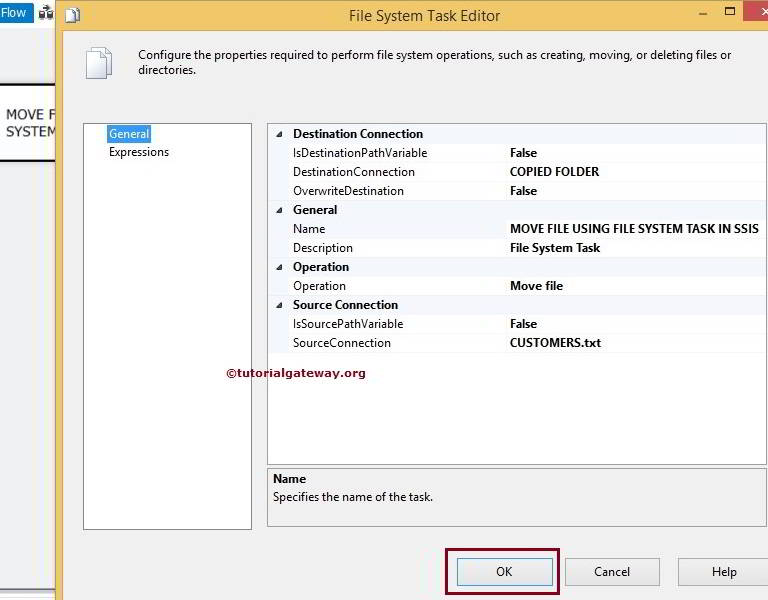
Click on the Browse button to select the Existing Folder from the file system.



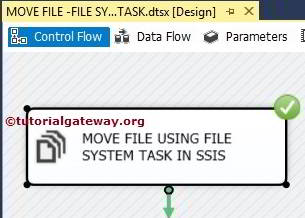
As you see, we selected the Copied Folder as the destination folder. Click the Ok button to select it.



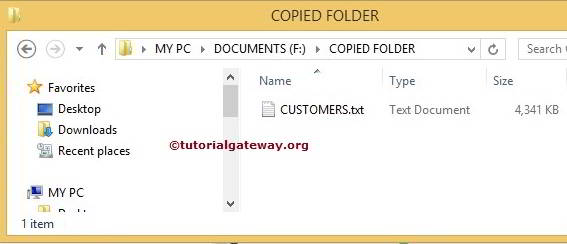
Next, Click Ok to complete configuring the File Connection Manager for the destination.



Click Ok to finish configuring Move File using File System Task in SSIS package. Let’s run and see whether we successfully Moved the Customers.txt file using the File System Task or Not.



Well, We successfully Moved the Customers.txt file present in the File System Task Folder to the Copied Folder.

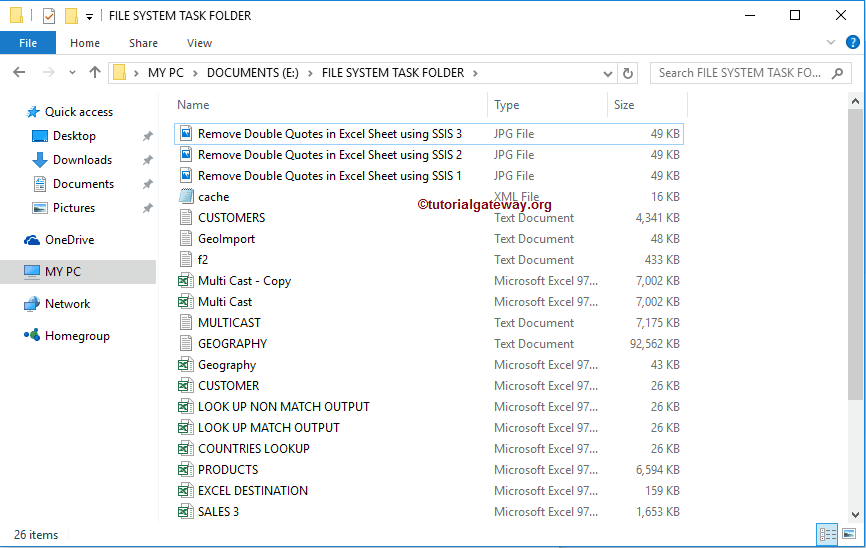


# SSIS FTP Task Send Files

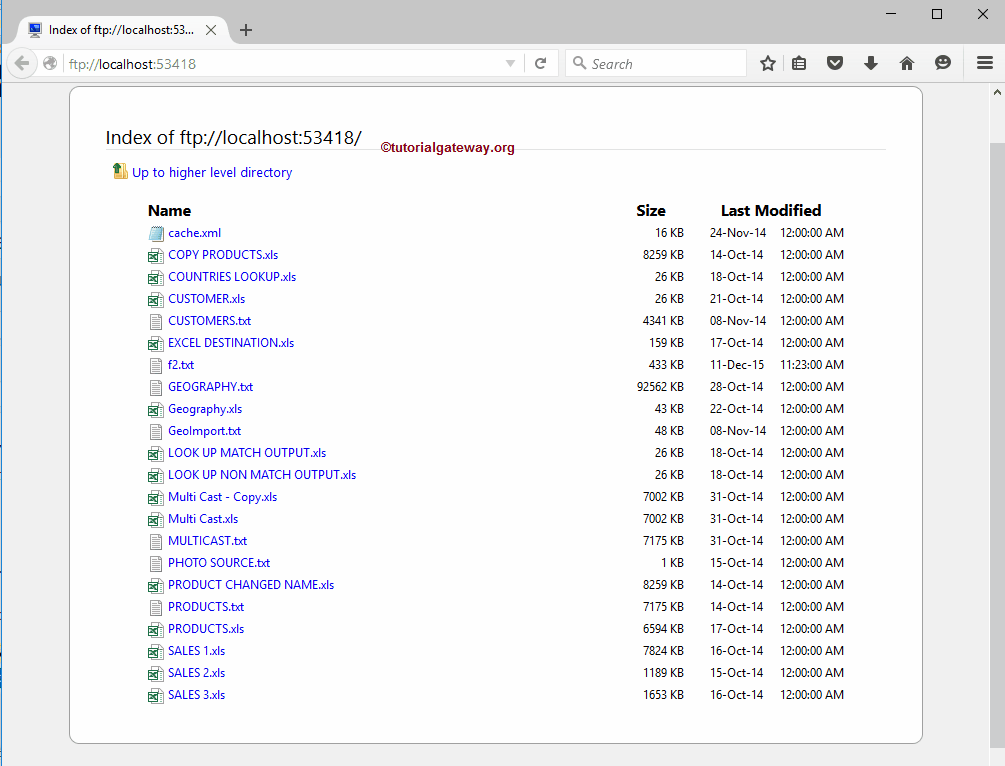
The [SSIS FTP Task](https://www.tutorialgateway.org/ssis-ftp-task/) is used to perform different kinds of operations on Files and Folders (or Directories). In this article we will show you the steps involved in configuring the SSIS FTP task send files with example.

**NOTE:**The [SSIS](https://www.tutorialgateway.org/ssis/) FTP Task uses the [FTP Connection Manger](https://www.tutorialgateway.org/ssis-ftp-connection-manager/) to connect with the Remote (FTP Server) Files and Folders.

Following screenshot shows you the data inside the local folder. Our task is to Copy Remove Double Quotes in Excel Sheet using SSIS 1.jpg file present inside the **File System Folder** to FTP Server’s Root directory.

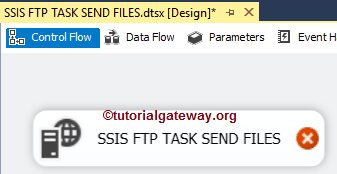


Below screenshot shows you the data inside the FTP Server.

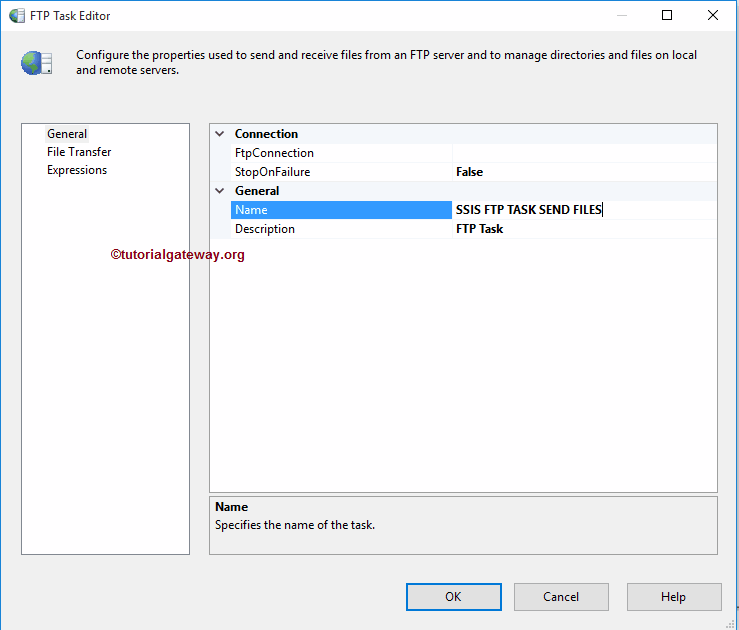


## SSIS FTP Task Send Files

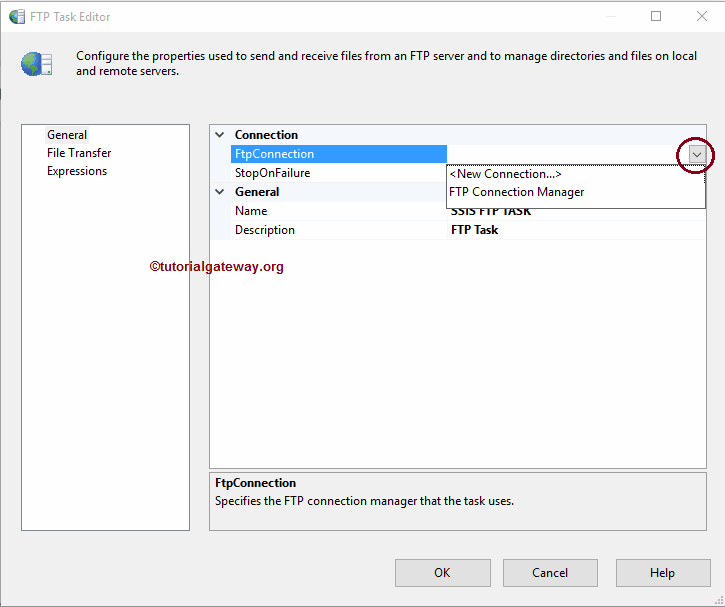
In order to send files from local computer to FTP Server, First Drag and drop the FTP Task into the Control Flow region and rename it as SSIS FTP Task Send Files



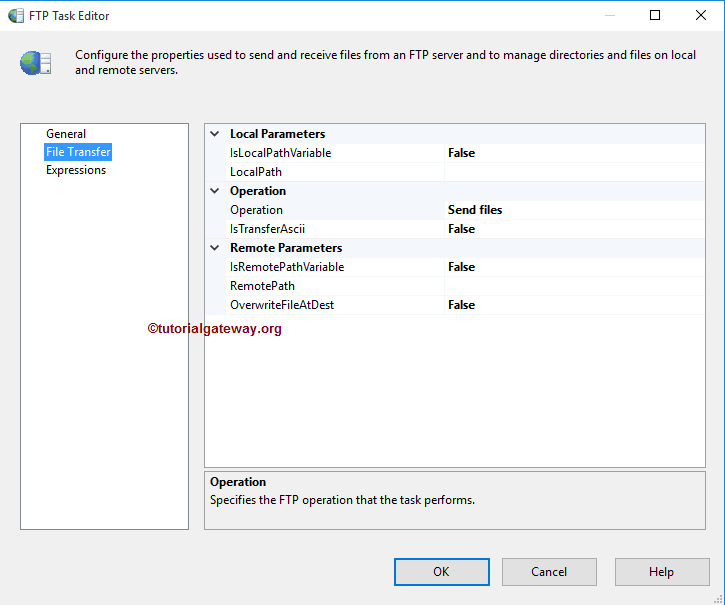
Double click on it will open the FTP Task Editor to configure it. Please change the task name according to your requirements and provide the valid description.



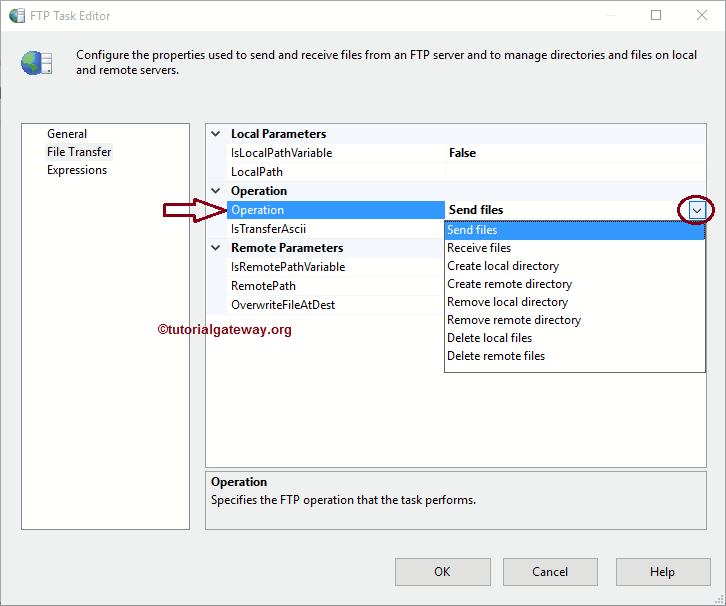
Within the Connection section, We have one property called **FTPConnection.** By clicking on the drop down arrow will show you the already created FTP Connections (If any) or please click on the ***<New Connection..>*** option to create one. In this example, We are using the already created Connection. Please refer [SSIS FTP Connection Manager](https://www.tutorialgateway.org/ssis-ftp-connection-manager/) article to understand the connection settings



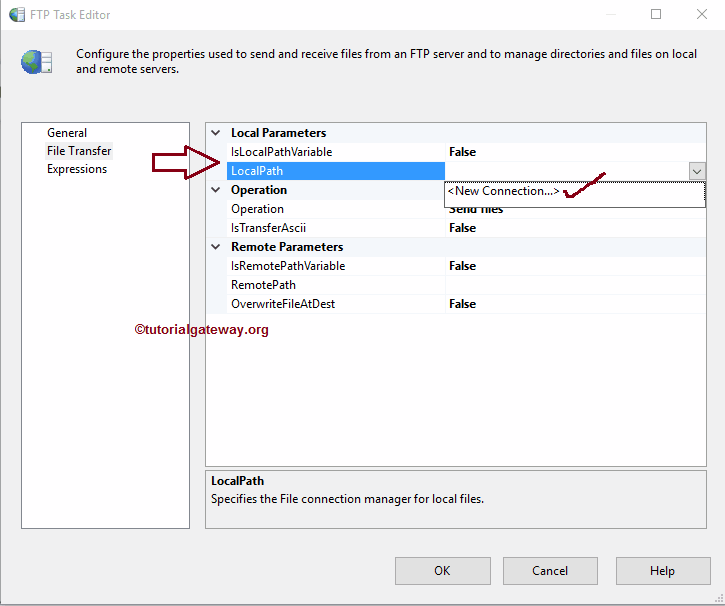
Please click on the ***File Transfer*** tab to configure the FTP operations. Following screenshot will show you the available properties in this tab.



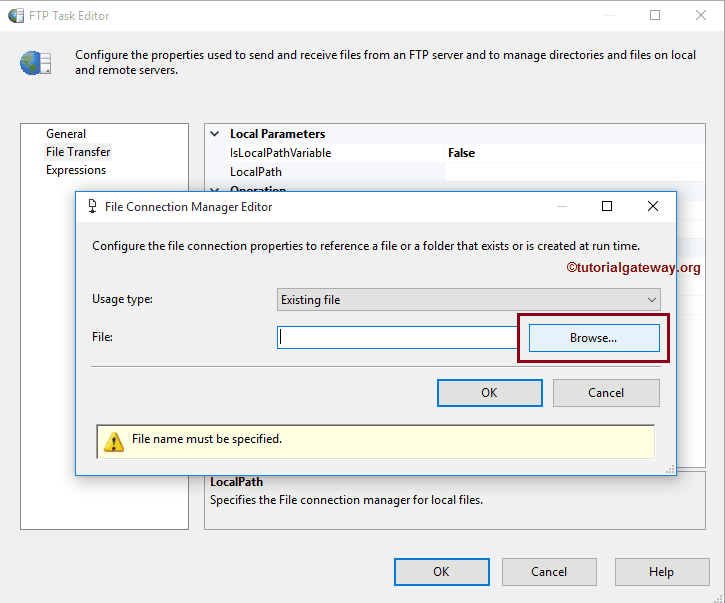
The FTP task in SSIS supports eight different operation on Files and Folders. In this example we want to send file to FTP Server so, We are selecting the ***Send Files*** option from the Operations property



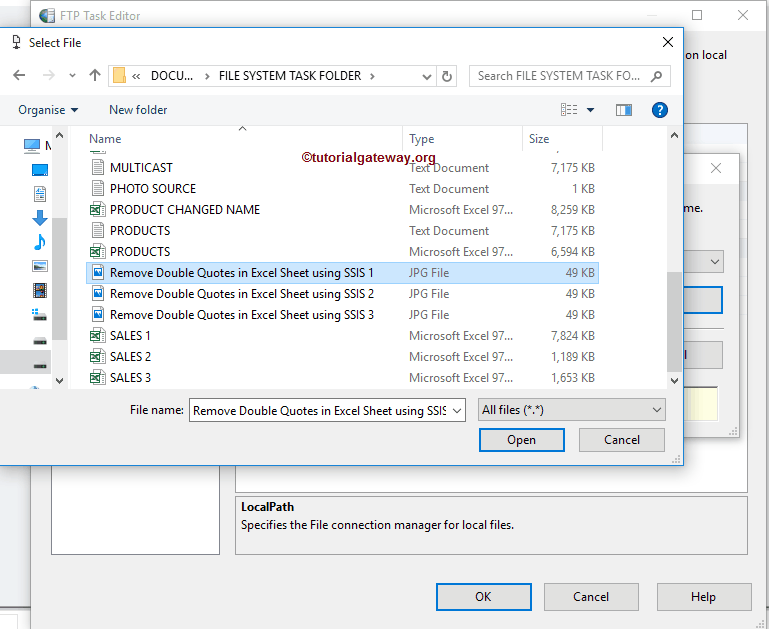
For the time being we are declaring the Local path manually but in real-time, we recommend variables. If we set the **IsLocalPathVariable** to False, we have to configure the Source Connection using **LocalPath**Property. If you already created the [File Connection Manager](https://www.tutorialgateway.org/file-connection-manager-in-ssis/) you can simply select it from the drop down list otherwise, You have to create by selecting **<New Connection..>**. as shown in below screenshot.



Once you selected the **<New Connection..>**for files and folders, File Connection Manager Editor will be opened in new window. Using this editor we have to configure the connection with files and folders.

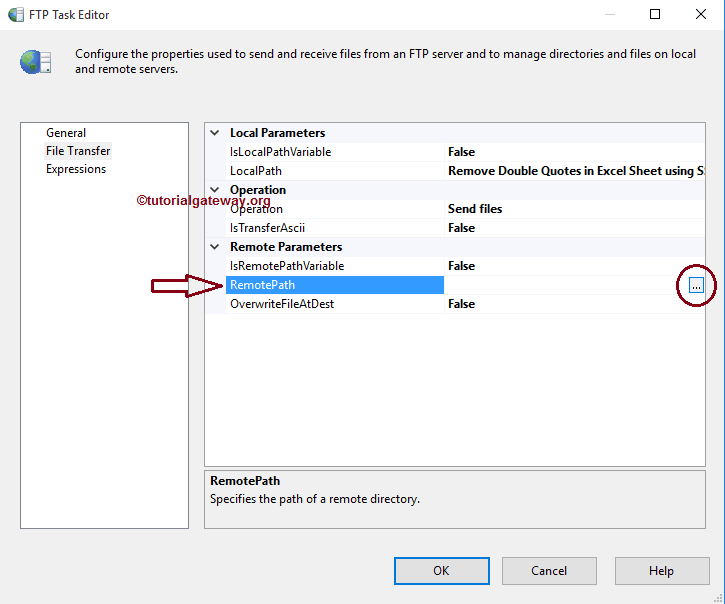


From the below screenshot you can observe that, we are selecting the Remove Double Quotes in Excel Sheet using SSIS 1.jpg file present inside the **File System Folder**

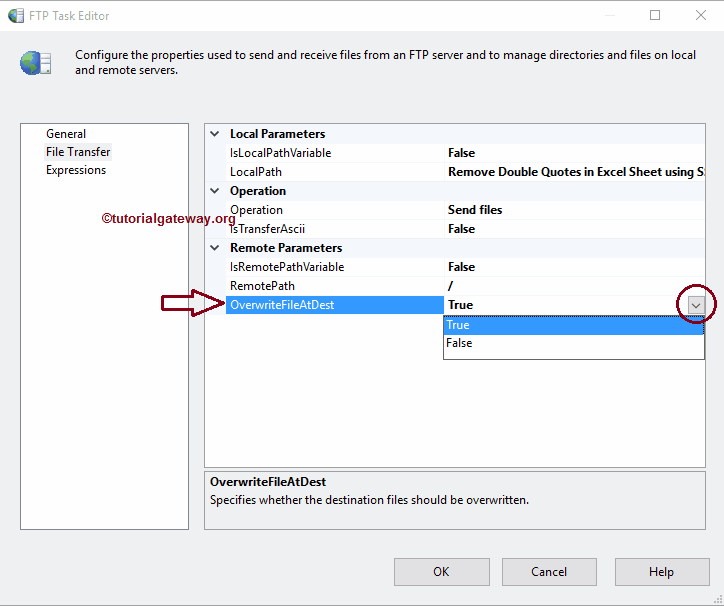


Once you selected the required file, Click on the OK button.

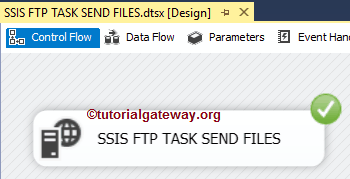
For the time being we are declaring the Remote path manually but in real-time, we recommend using variables. Please click on the browse (**…**) button besides this option to open the remote directory.



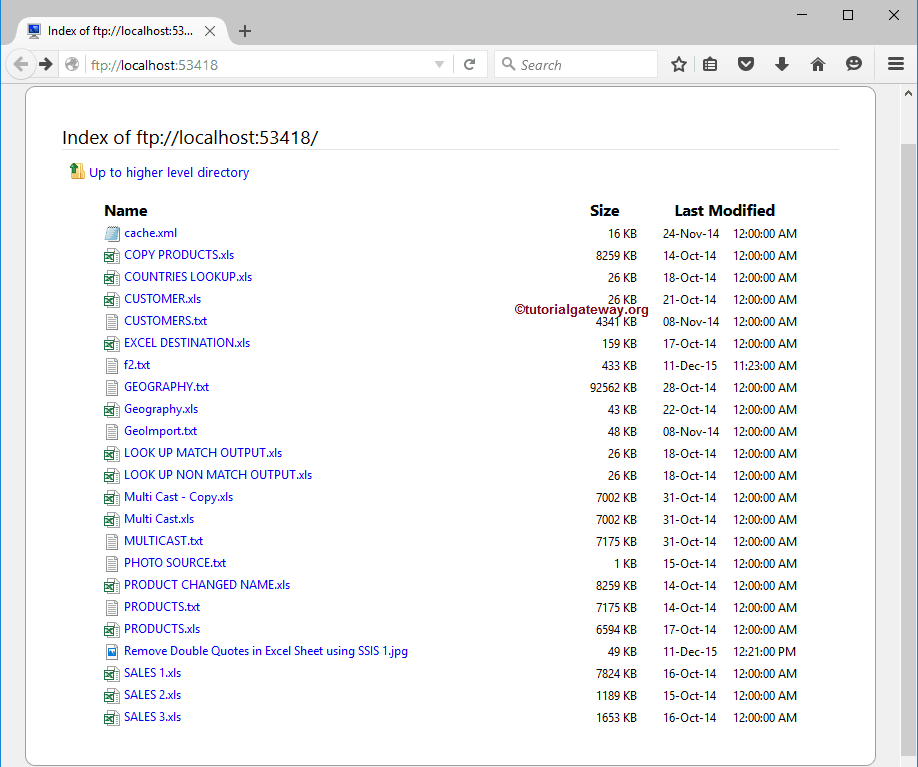
Currently we don’t have any sub folders in our FTP Server so, we are selecting the root directory**/**as the location. Next we are setting the **OverwriteDetination** property to true. It means SSIS FTP Task will overwrite the existing files (if any) in the FTP Server.



Click Ok to finish configuring the SSIS FTP task send files package. Let’s run and see whether we successfully Copied the JPEG file from the local File System to FTP Server or Not.



Well, We successfully Copied the Remove Double Quotes in Excel Sheet using SSIS 1.jpg file present inside the **File System Folder** to FTP Server’s Root directory..



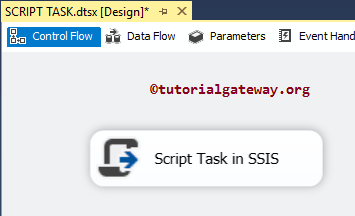
# SSIS Script Task

The SSIS Script Task gives an option to implement functions that are not available or possible in the SSIS toolbox (both in built-in Tasks and transformations). The SSIS script task utilizes the Microsoft VSTA (Visual Studio Tools for Applications) as the code environment in which you can write the C# or VB Script.

**TIP:** SSIS Script Task VSTA provides all the standard futures available in general Visual Studio Environment.

## SSIS Script Task Send Email

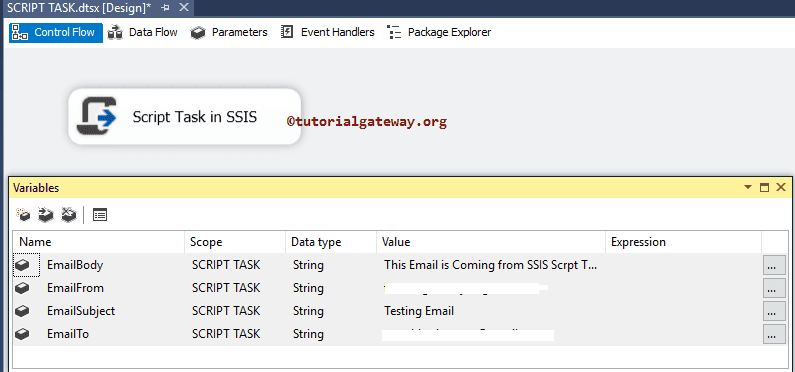
Drag and drop the Script Task from SSIS toolbox into the Control Flow region



Before we start configuring the SSIS Script task, let us create four variables. To create a variable, right-click on the control flow region, and it will open the context menu to select the variable option. Once you choose the variable option, it will open the new window called ***Variables*** to create the new variable.

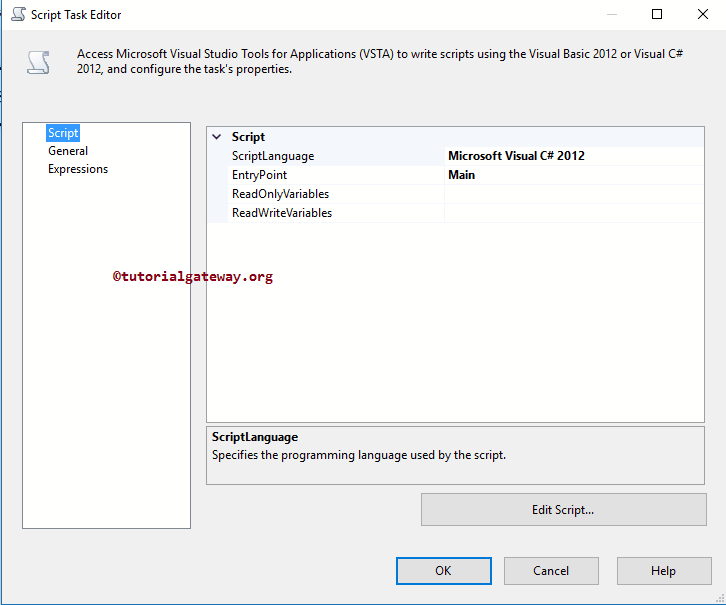
Here you can create **n** variables

* **EmailFrom:** Please specify the Email ID from where you want to send the Email.
* **EmailTo:** Please specify the Email ID to whom you want to send the Email.
* **EmailSubject:** Please specify the Subject you want to include
* **EmailBody:** Please specify the Message. It can be Plain Text or HTML Message



Double click on the Script task will open the following editor to configure the SSIS Script task components

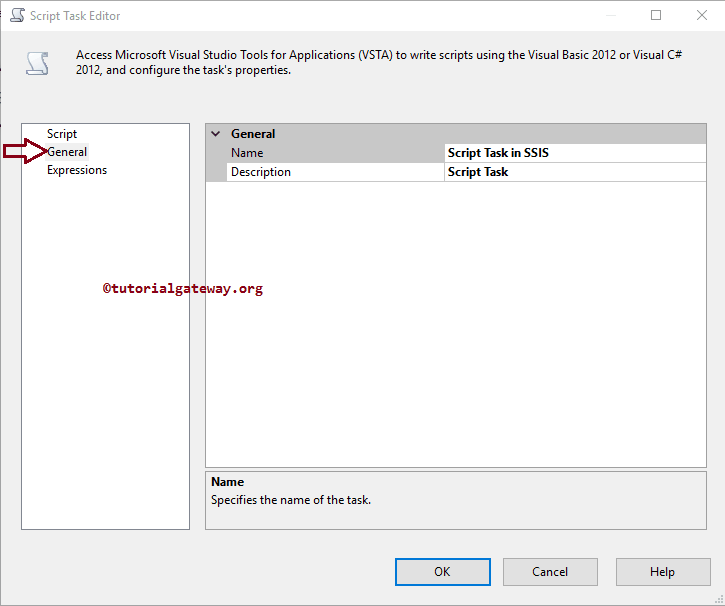
* **EntryPoint:** Please specify the Method name that the [SSIS](https://www.tutorialgateway.org/ssis/) run-time calls as the entry point. The method name you specify here must be in ScriptMain class. Remember, you can change the name as per your requirement, and when you change it you have to change it in ScriptMain class



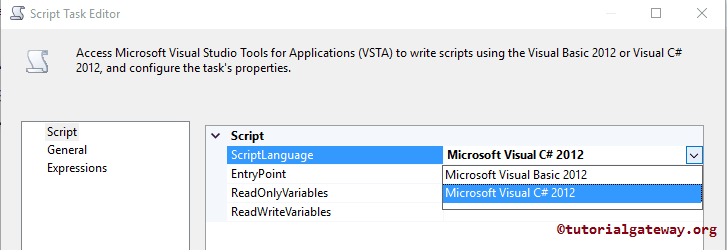
### SSIS Script Task General tab

Please click on the General tab to change the default Name and Description.

* **Name:** Please provide the Unique Name
* **Description:** Briefly describe the SSIS Script Task Functionality. It is always a good practice to provide a valid description.

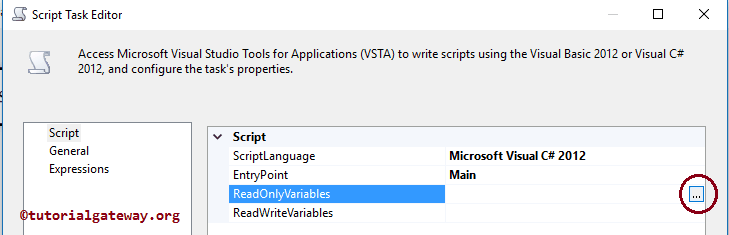


**ScriptLanguage:** Microsoft provides two of its popular languages: Visual Basic (VB) and C# to use as a scripting language. I am very much familiar with C# so, I am selecting C# as my script language.

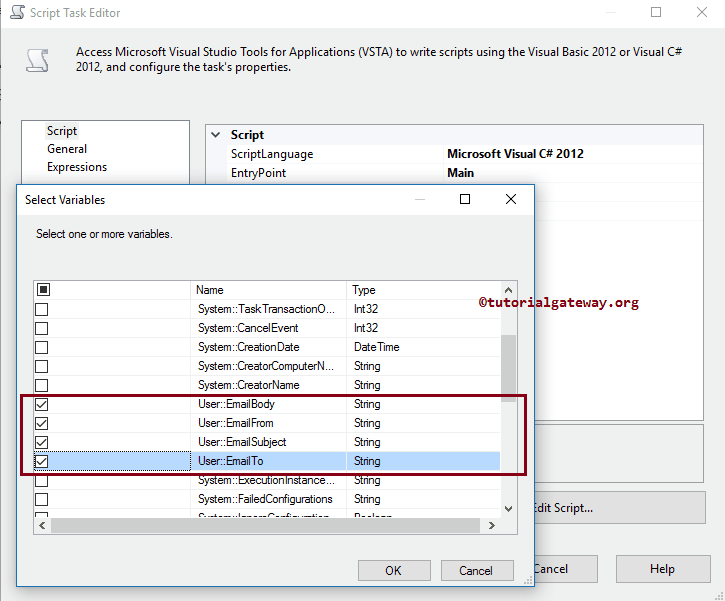


* **ReadOnlyVariables:** Please select the variables that you want to use in the SSIS Script Task, and they may be user-defined variables or System default variables. Remember, variables selected as *ReadOnlyVariables* used for the Read-only purpose (we can’t alter them)
* **ReadWriteVariables:** Please select the variables that you want to use in the Script Task, and they may be user-defined variables or System default variables. Remember, variables selected as *ReadWriteVariables* can alter according to our requirement

To select the variables, please click on the Eclipse (…) button *ReadOnlyVariables*option

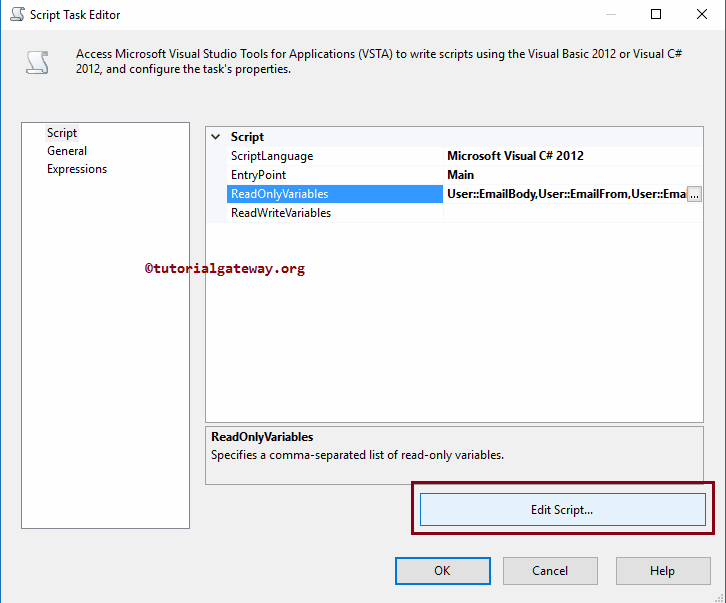


Once you click on the Eclipse (…) button, Select the Variables window will open. Please select the previously created variables



### Add C# code in SSIS Script Task

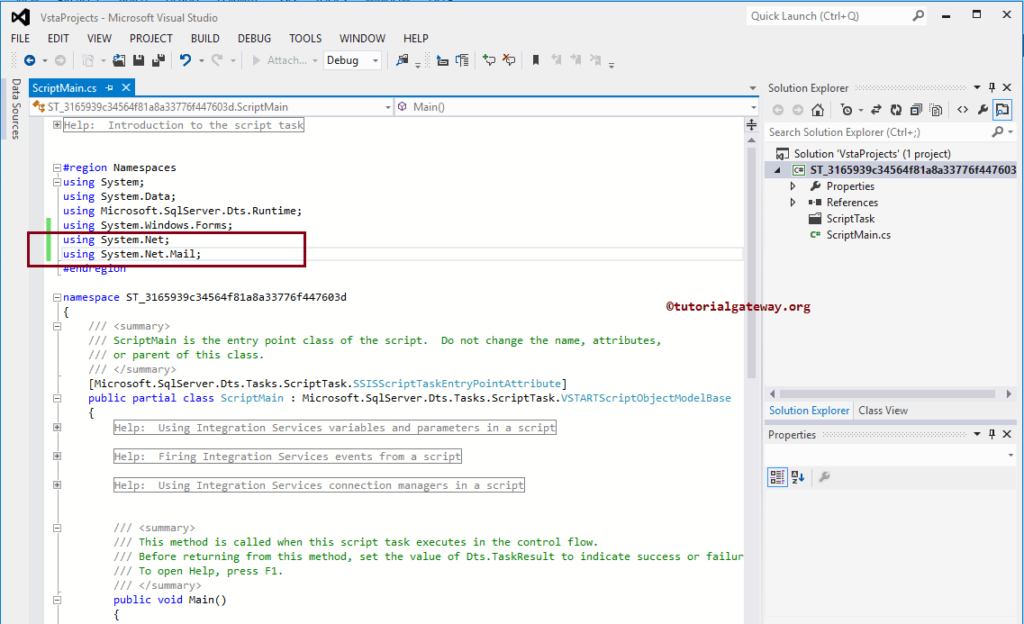
Once you selected the required variable, please click on the ***Edit Script..***button to write the actual C# Script



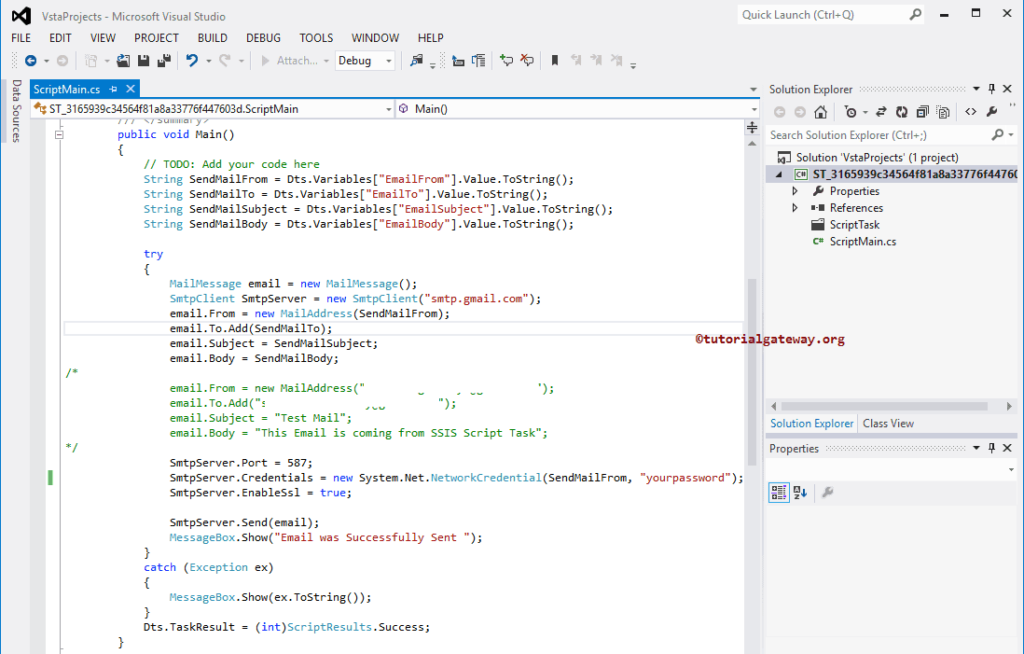
Once you click on the ***Edit*** Script, it will open the ScriptMain.cs class file to write the C# code.

To send an email from C# we need to add two references or import references called: ***using System.Net***; and ***using System.Net.Mail;***

**TIP:** If you are VB developer then it should be something like ScriptMain.vb



Next, within the Main() function add your Code. Remember, if your code is long or robust, try to divide the code by creating methods



C# code we used for this SSIS Script Task in the above screenshot is:

// C# Script for Script Task in SSIS

String SendMailFrom = Dts.Variables["EmailFrom"].Value.ToString();

String SendMailTo = Dts.Variables["EmailTo"].Value.ToString();

String SendMailSubject = Dts.Variables["EmailSubject"].Value.ToString();

String SendMailBody = Dts.Variables["EmailBody"].Value.ToString();

try

{

MailMessage email = new MailMessage();

SmtpClient SmtpServer = new SmtpClient("smtp.gmail.com");

// START

email.From = new MailAddress(SendMailFrom);

email.To.Add(SendMailTo);

email.Subject = SendMailSubject;

email.Body = SendMailBody;

//END

SmtpServer.Port = 587;

SmtpServer.Credentials = new System.Net.NetworkCredential(SendMailFrom, "your password");

SmtpServer.EnableSsl = true;

SmtpServer.Send(email);

MessageBox.Show("Email was Successfully Sent ");

}

catch (Exception ex)

{

MessageBox.Show(ex.ToString());

}

Dts.TaskResult = (int)ScriptResults.Success;

If you forgot to create Variables (or if you find it difficult to create variables) then, you can remove the first four lines of code and replace the code between //Start… END with the below-shown code

// SSIS Script Task Example

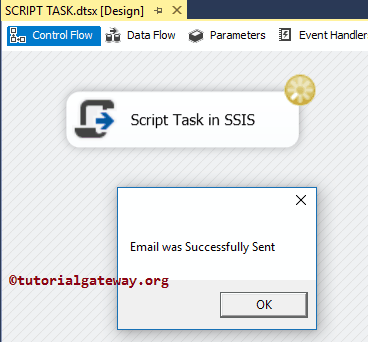
email.From = new MailAddress("xxxxxxxxxxxx@gmail.com");

email.To.Add("xxxxxxxxxxxxxy@gmail.com");

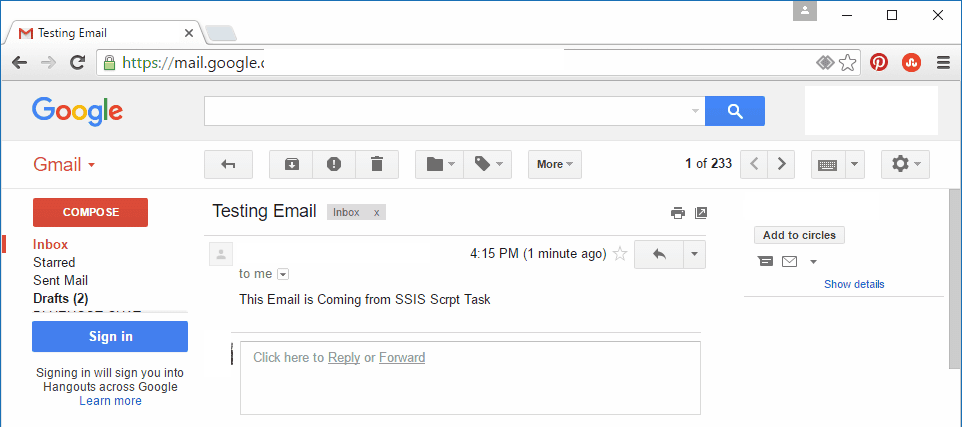
email.Subject = "Test Mail";

email.Body = "This Email is coming from SSIS Script Task";

Once you finished editing the Script, Please close the ScriptMain.cs file and Script Task Editor. Let us run the SSIS Script Task package



From the above screenshot, you can observe that SSIS Script Task is successful, and we got the message box saying that *Email was Successfully sent.*Let me open my Gmail



From the above screenshot, you can see that we got the Email with the message we specified.

### Points to Remember for this Script Task :

* If you want to send an email to your Manager (or to yourself) after every Data Load then, you can use this SSIS Script task along with the Data Flow Task.
* Always use the Variables for storing Emails, Passwords, Subject, or SMTP credentials.
* If there is an Error stating Secure Connection then, Go To **Less Secure Apps** in your Gmail account settings and Turn on the ***Access for Less Secure Apps***option.

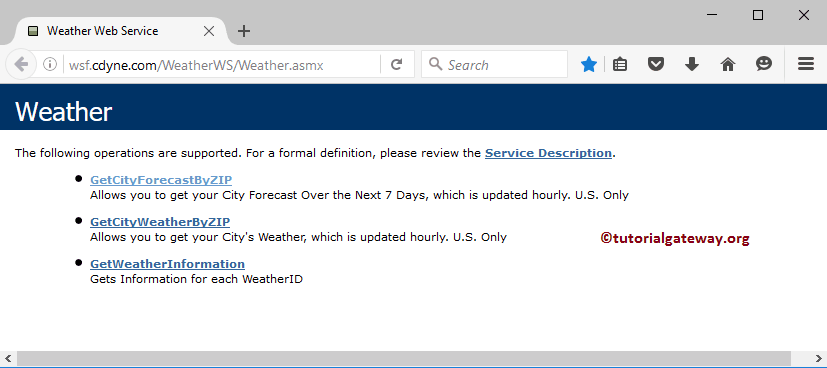
# SSIS Web Service Task

The SSIS Web Service Task is used to execute the web service methods. For instance, if you want to store the weather report as an SQL extra column, then consume the Web service provided by the Yahoo or MSN weather, and store the output in a variable so that you can pass the variable to Data Flow Task.

In this article, we show you how to consume the web service methods using the SSIS Web Service Task with example. Before we start configuring the Web Serve Task in SSIS, let us see the list of the available method in our service.

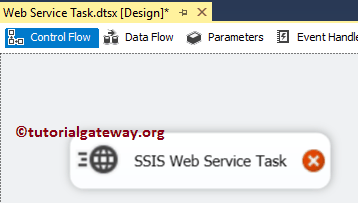
**NOTE:**The SSIS Web Service Task uses the HTTP Connection Manager to connect with the Web service and to send or receive Files.

The below screenshot shows you the Methods available in this SSIS web service Task. Our task is to consume the GetCityForecastByZip method available in this web service. Please visit https://wsf.cdyne.com/WeatherWS/Weather.asmx website to use the same service.



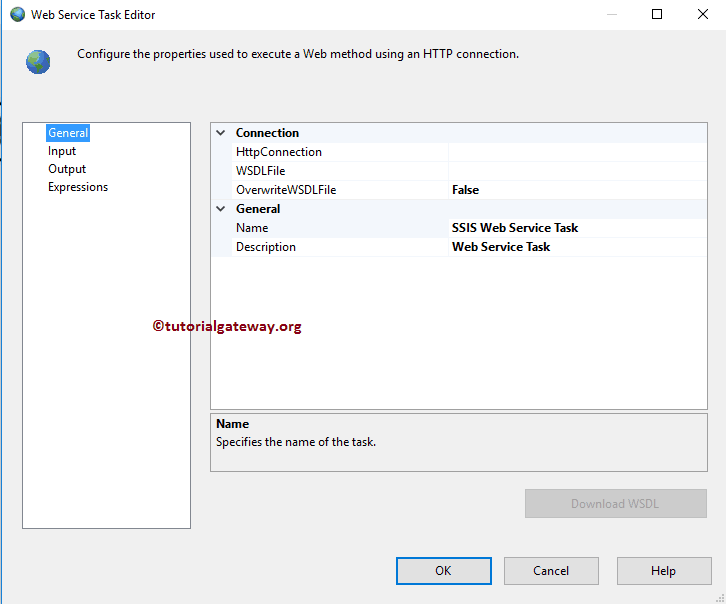
## SSIS Web Service Task Example

Drag and drop the Web Service Task into the Control Flow region and rename it as the SSIS Web Service Task

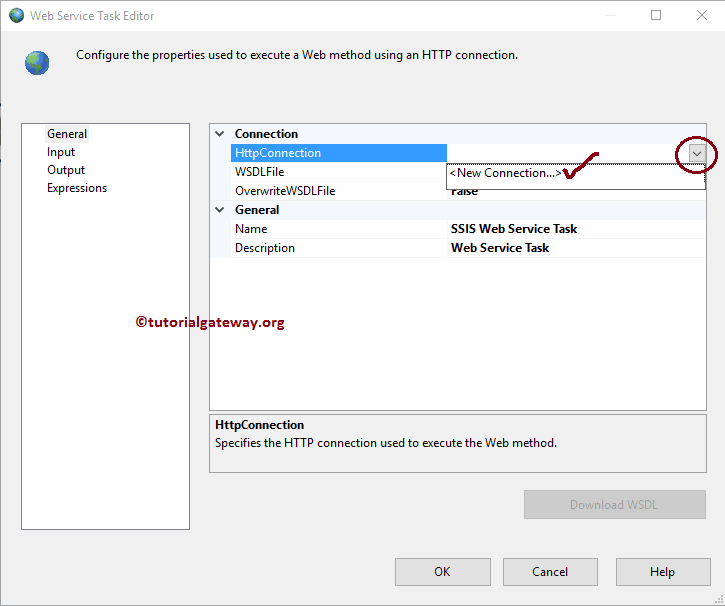


Double click on it will open the Web Service Task Editor to configure it.

* **Name:** Please provide the Unique Name for this web service task
* **Description:** Briefly describe the SSIS Web Service Task Functionality. It is always a good practice to provide a valid description.

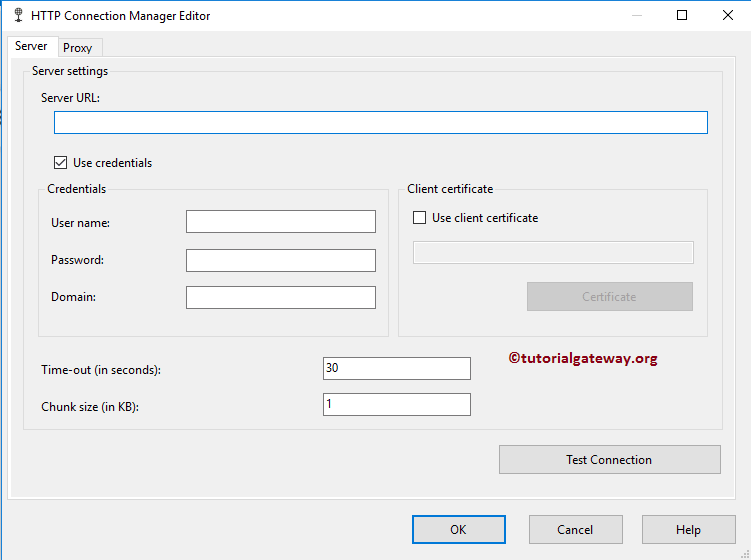


**HttpConnection:**An HTTP Connection enables this [SSIS](https://www.tutorialgateway.org/ssis/) package to access web services. It used the HTTP to send and receive the files. By clicking on the drop-down arrow, will show you the already created HTTP Connections (If any), or please click on the ***<New Connection..>*** option to create one. Let us see what happen when click on the ***<New Connection..>*** option

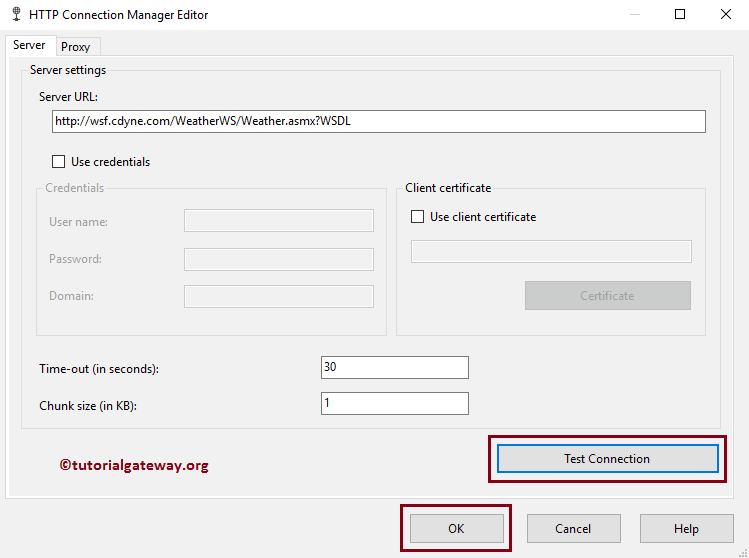


Once you click on the **<New Connection..>**option, HTTP Connection Manager Editor will be opened to configure it.

* **Server URL:** Please specify the URL of the web service. If you are planning to download the WSDL file using the **Download WSDL** button, type the URL of the WSDL file; otherwise, choose the service URL.
* **Use Credentials:** If you want the HTTP Connection Manager to use the security credentials of a user, Please check mark this option.
* **User name:** Please specify the user name to access the Service.
* **Password:**Please specify the password to access the Service.
* **Domain:** Please specify the domain here.
* **Use client Certificate:** Please specify whether you want to use the client certificate as the authentication or not.
* **Time-out (in seconds):** Please specify the web service connection time out in seconds. If the connection takes more than this time, then the connection will fail.
* **Chunk size(in KB):** Please specify the chunk size for the writing data.
* **Test Connection:** This button will help us to check whether we successfully established the connection to web service or not. Based on the result, we can alter the connection settings.



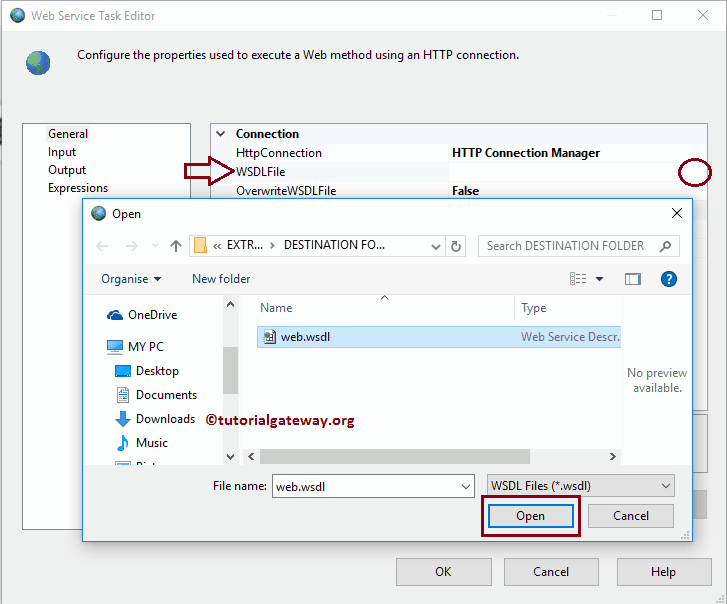
In this example, we are selecting the Web service from a free source. So, we don’t have to configure the credentials and proxy settings. From the below screenshot, you can observe that we are providing the Service URL: *https://wsf.cdyne.com/WeatherWS/Weather.asmx?WSDL*because we want to download the WSDL file.



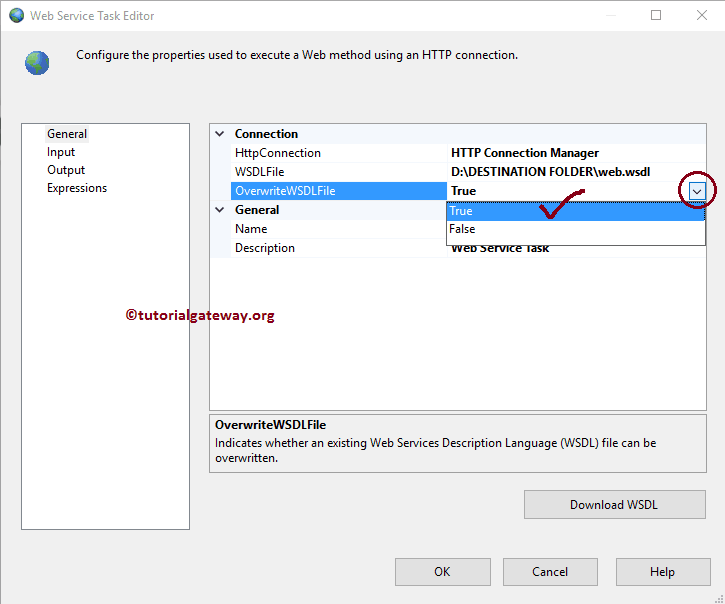
Click OK to close the HTTP Connection Manager.

**WSDL File:**If you already downloaded the WSDL file, then we have to select the .wsdl file from your local file system. Otherwise, create an empty file with WSDL extension in your local hard drive and then click on the **….** button beside the **WSDLFile** option.

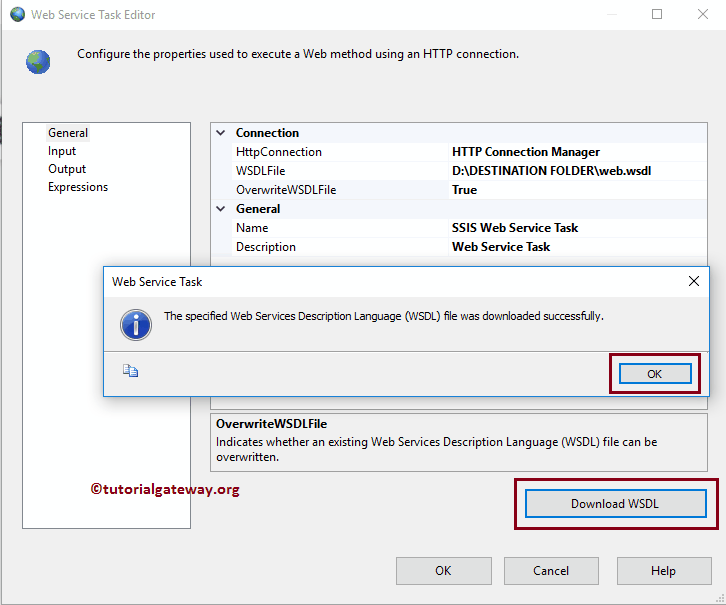
Once you click on the **….** option, a new window will open. Using this, we can create a New WSDL file in your local file system or select the Existing File from the file system. From the below screenshot you can observe that we are choosing the existing file called web.wsdl



**OverwriteWSDLFile:**This SSIS Web Service Task property has two options: True and False. If we set this property to true, the Web Service Task will overwrite the existing WSDL files in the local file system.

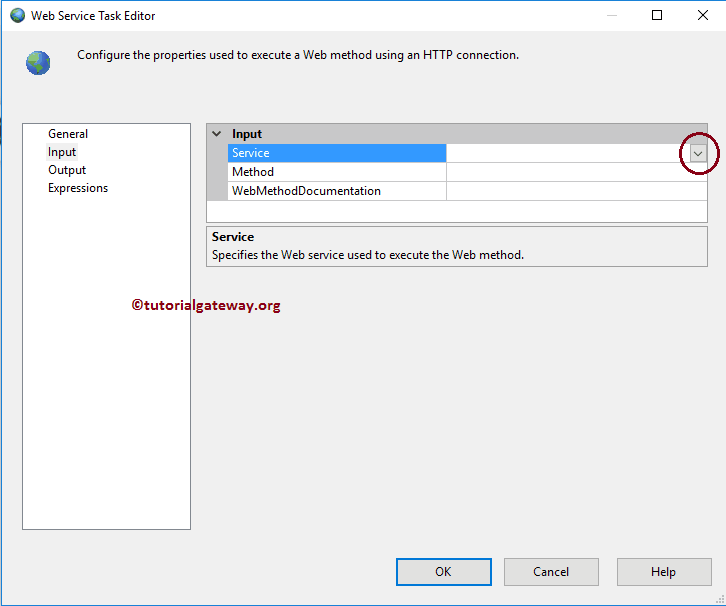


Next, click on the Download WSDL button to download the WSDL file, and if it exists, it will overwrite the existing file.

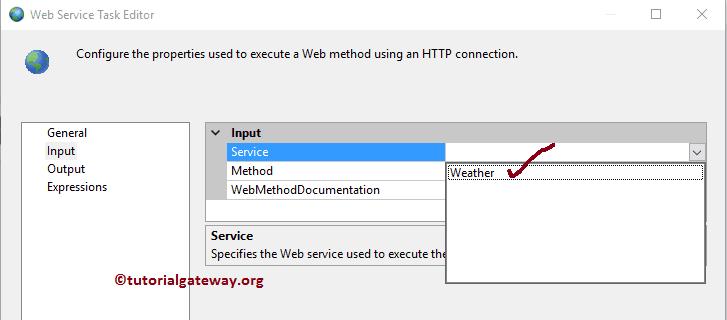


Next, click on the **input** tab to Specify the service Service and methods.

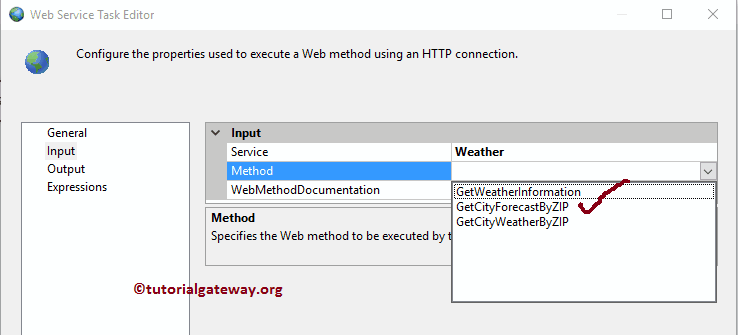
* **Service:** Please select the required service from the list of available web services.
* **Method:** Here, it displays the list of available Methods in above-selected web services. So, please select the required method



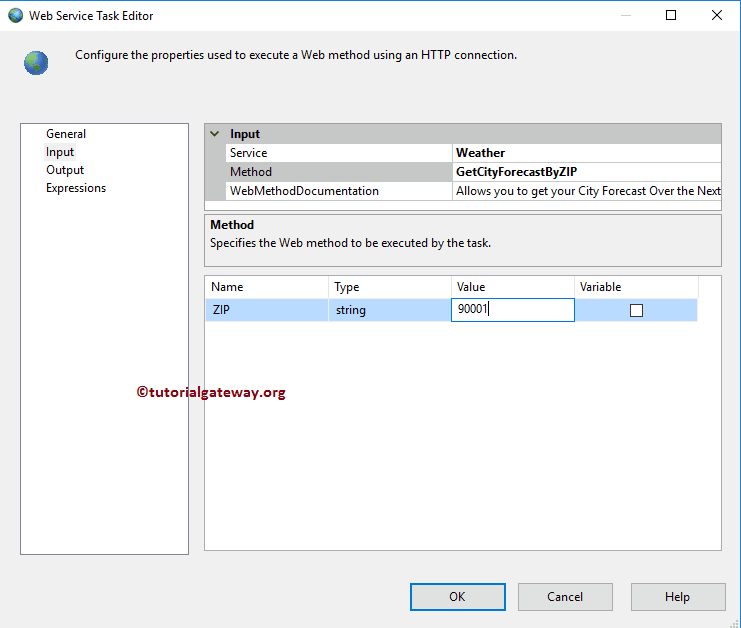
**SSIS Web Service Task Service:**By clicking on the drop-down arrow will show you the available services. So, please select the required one. Here we have only one service, and we are choosing the same.



**Method:** By clicking on the drop-down arrow will show you the list of available methods. From the below screenshot, you can observe that we are selecting the GetCityForcastByZip method. This method accepts the Zip code as a parameter and displays the weather forecast.

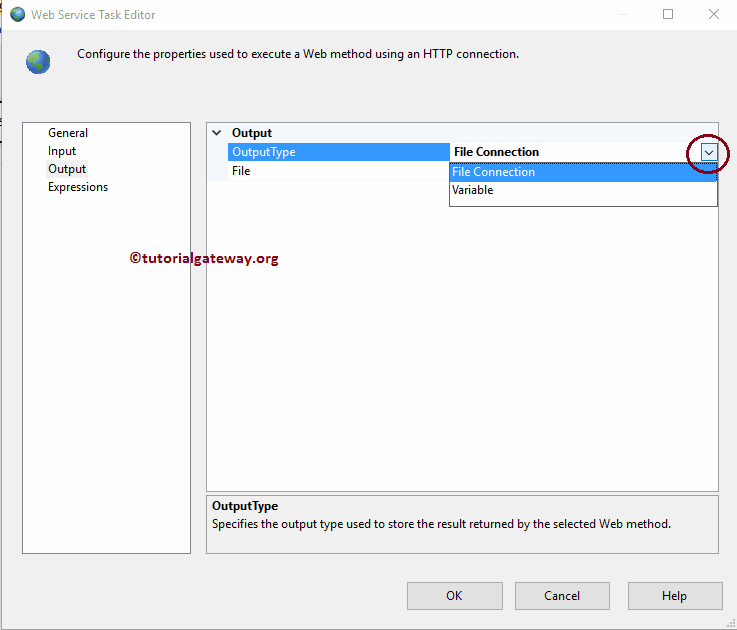


Since this method accepts the Zip code as a parameter, we are providing the Zip code of California. Here we have an option to choose the variable also.

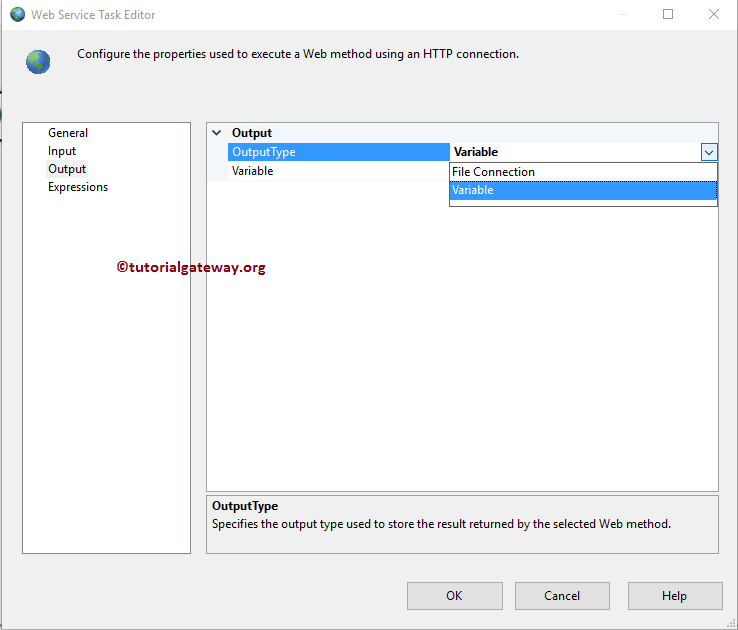


Next, click on the SSIS Web Service Task **Output** tab to configure the Output data.

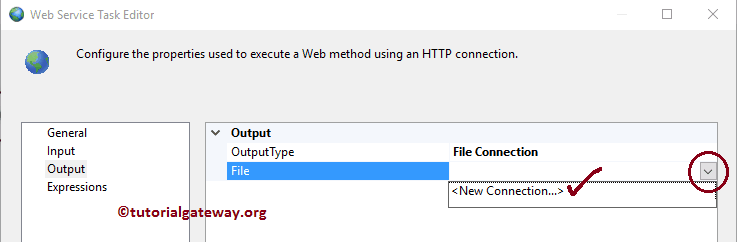
If we set the **OutputType** to File connection, we have to configure the Output Connection as File Connection Manager. It means the output XML data stored in a local file system.



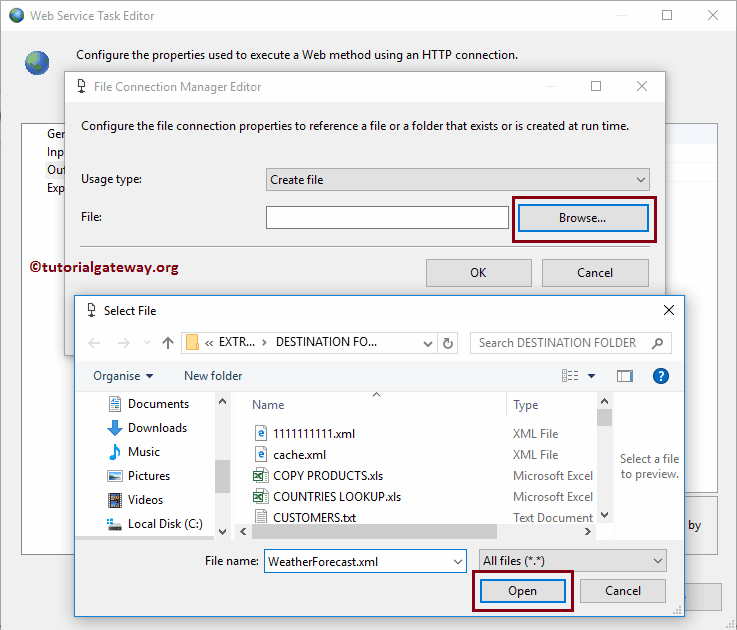
If we set the **OutputType** to Variable, then we have to configure the Output Connection as Variable. It means the output XML data stored in a variable. This option is handy because we can use this XML data in other tasks or transformations.



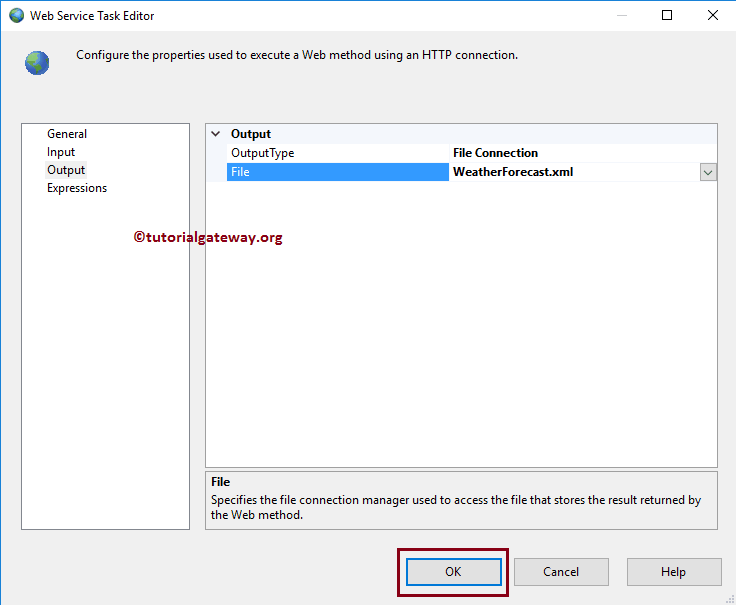
Once you click on the **<New Connection..>** option, a new window called File Connection Manager Editor opened. Using this, we have to configure the Output Connection.



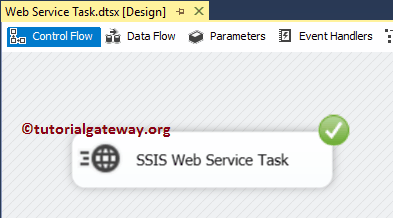
Click on the Browse button to create a New XML file available in your local file system or select the Existing File from the file system. Please refer to the [File Connection Manager](https://www.tutorialgateway.org/file-connection-manager-in-ssis/) article to understand the configuration.



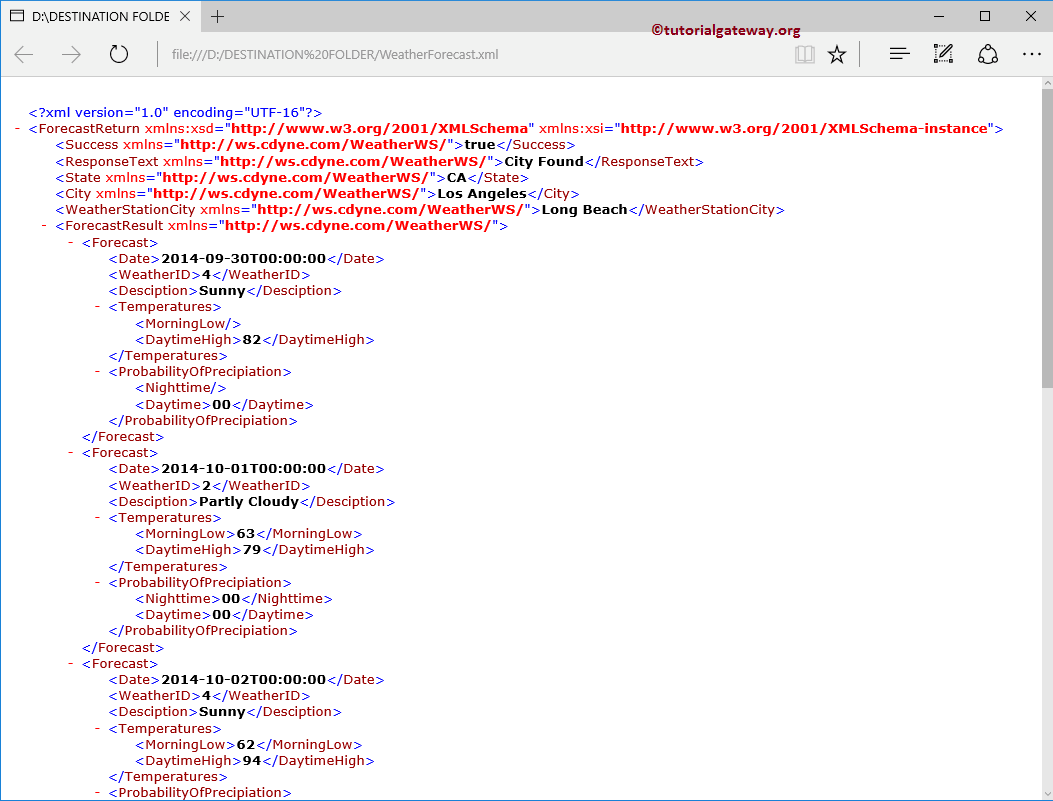
From the above screenshot, you can observe that we are creating a new file called WeatherForecast.xml file.



Click OK to finish configuring and closing SSIS Web Service Task Editor. Let us run the package to check whether we consumed the web service or not.



Let us open the WeatherForecast.xml file and see.



From the above screenshot, you can see that we achieved the result.