

# SSIS from Oracle to SQL Server

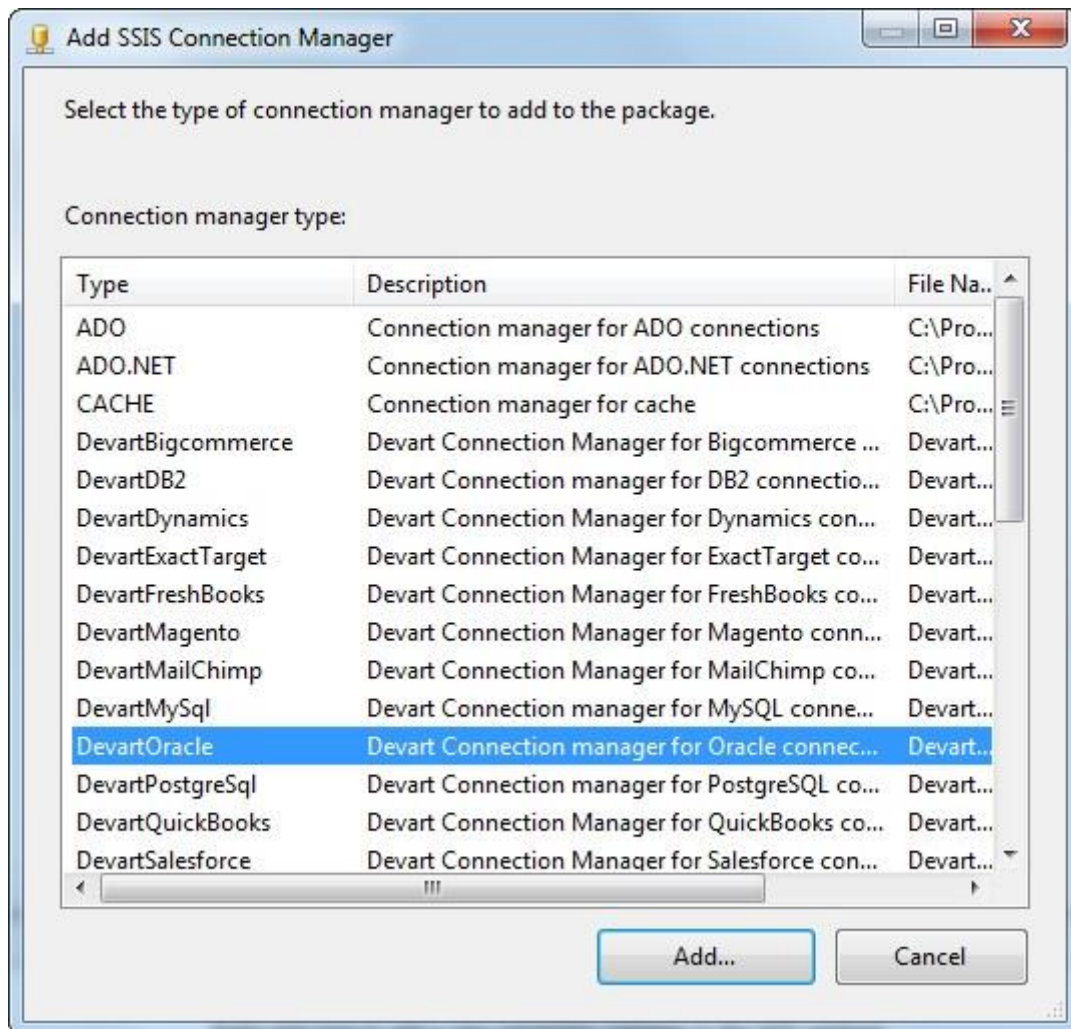
## SSIS Package to Convert Oracle Data for Transfer to SQL Server Table

Using Devart's SSIS Data Flow Components, this article will demonstrate how to connect to a Devart Oracle Data Source, use a Derived Column, use a Data Conversion Task to connect to a SQL Server 2014 table and transfer the converted data into it.

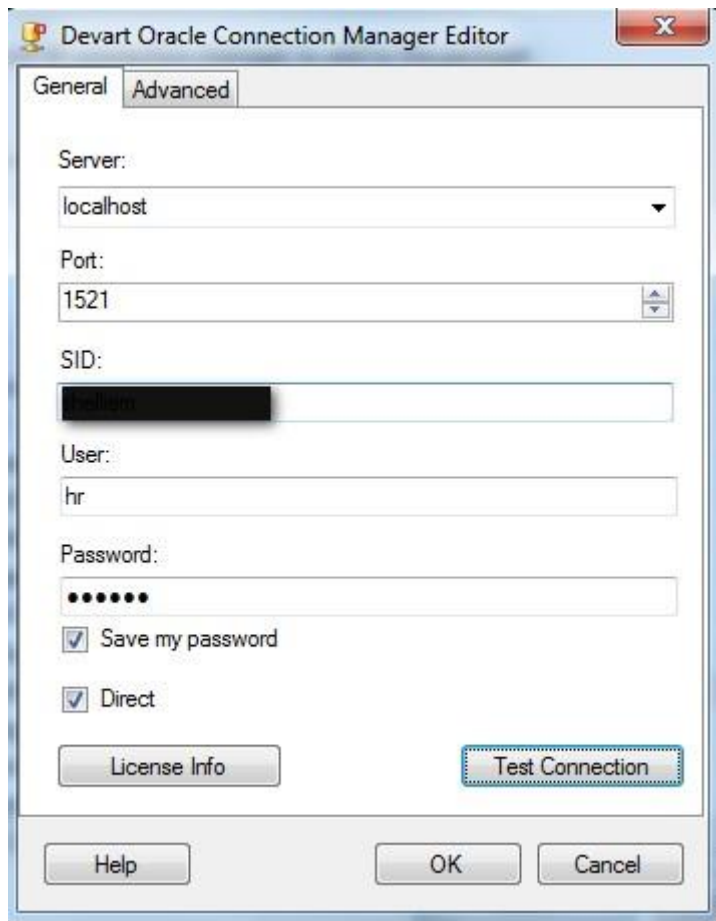
Begin by adding a Data Flow component to the Control Flow tab in your SSIS Project. Click on the Data Flow tab to continue with the project.

## SSIS Oracle Connection Setup

Add a New Oracle connection by Right clicking the Connection Manager and selecting DevartOracle Connection.

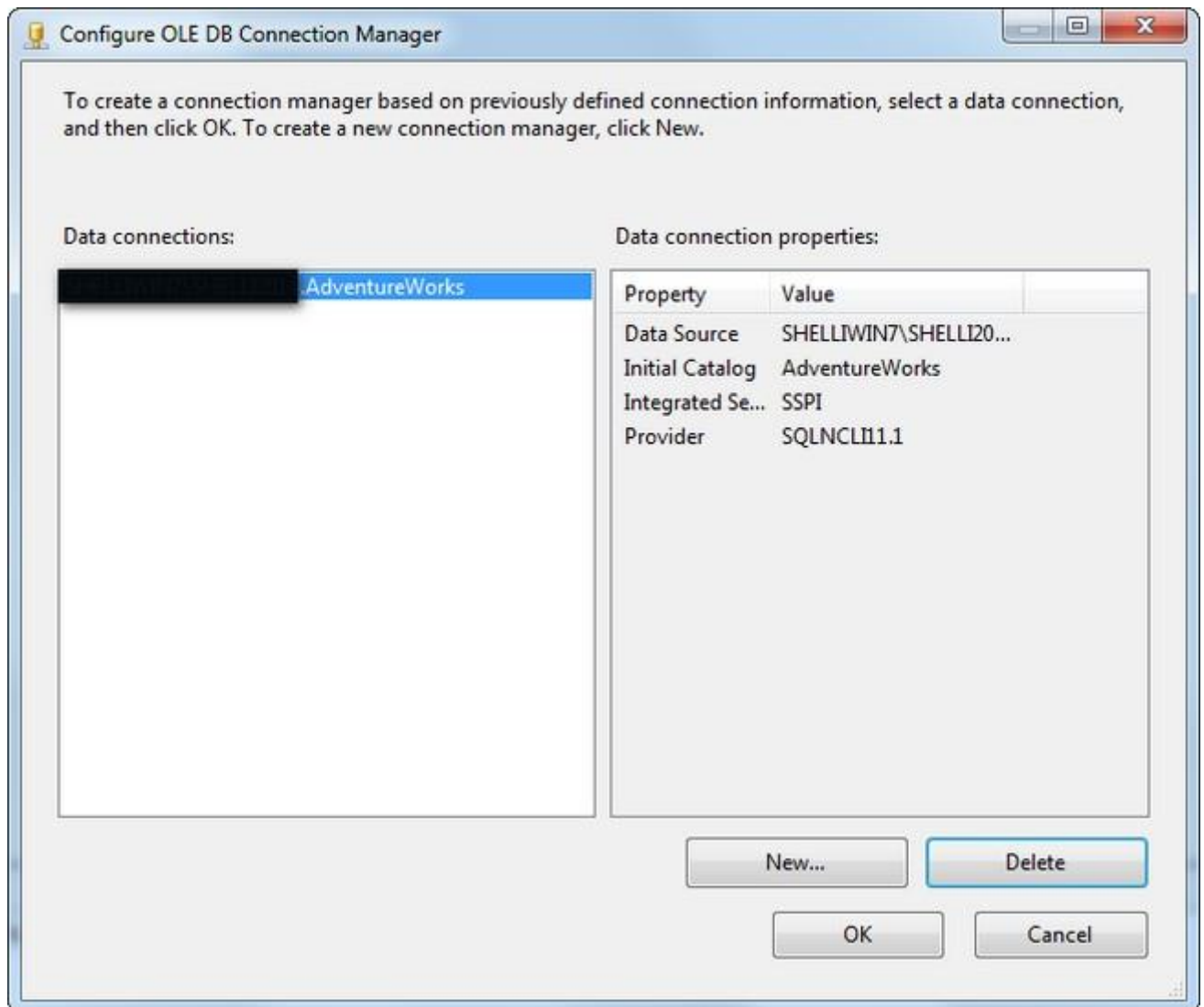


Click on the Direct checkbox to get the additional fields for the Port and SID boxes to show up. The user in this exercise is HR.



## SSIS SQL Server OLE DB Connection Setup

Right click on the Connection Manager and click on the New OLE DB Connection. Click New to select the database of your choice. The database used here is AdventureWorks. Click OK to close the window.



In SQL Server Management Studio, create a table in the instance of SQL Server database that will be used in this package. This table is named OracleEmps.

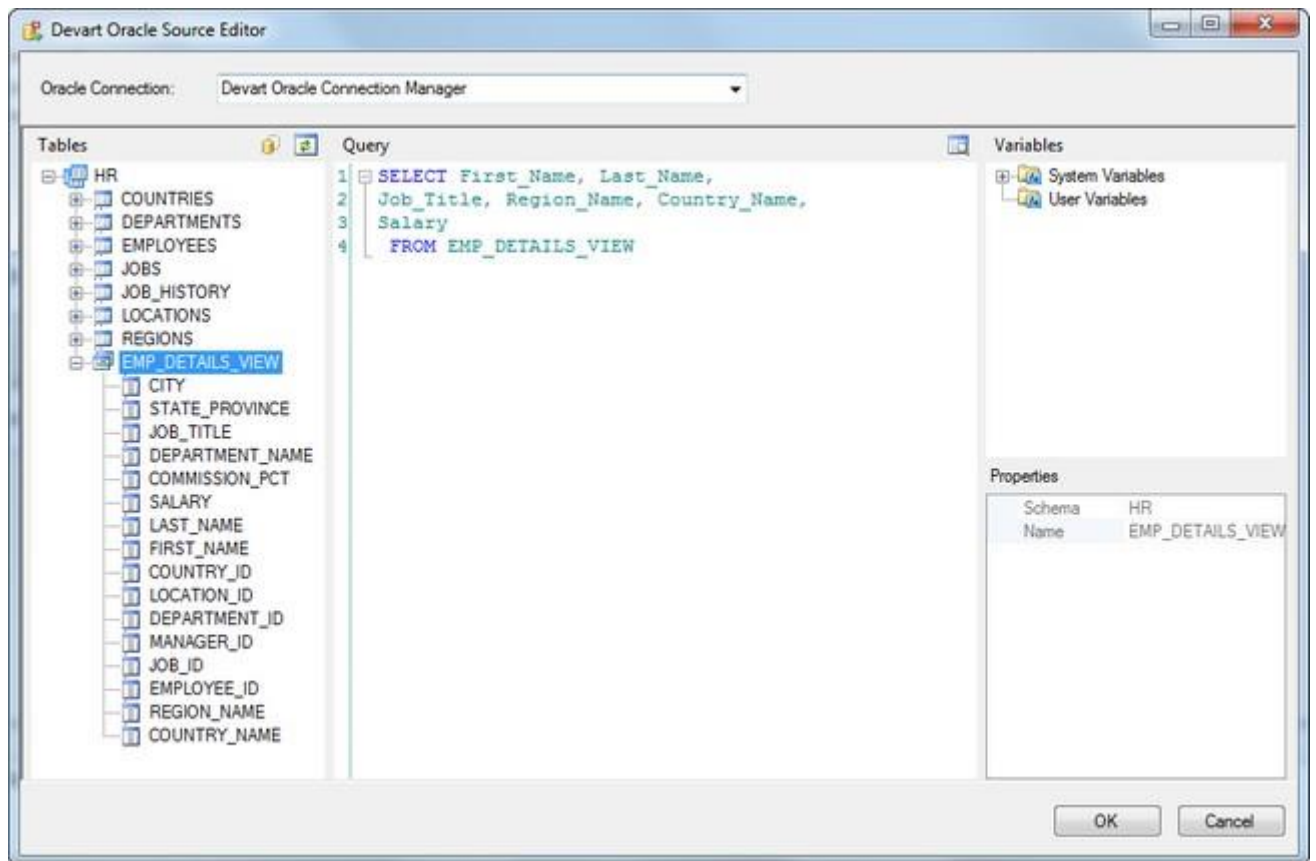
```

Create Table OracleEmps
(
  id int identity(1,1) not null primary key clustered,
  EmpName varchar(255) null,
  jobTitle varchar(100) null,
  Salary money null,
  Region varchar(100) null,
  Country varchar(100) null
)

```

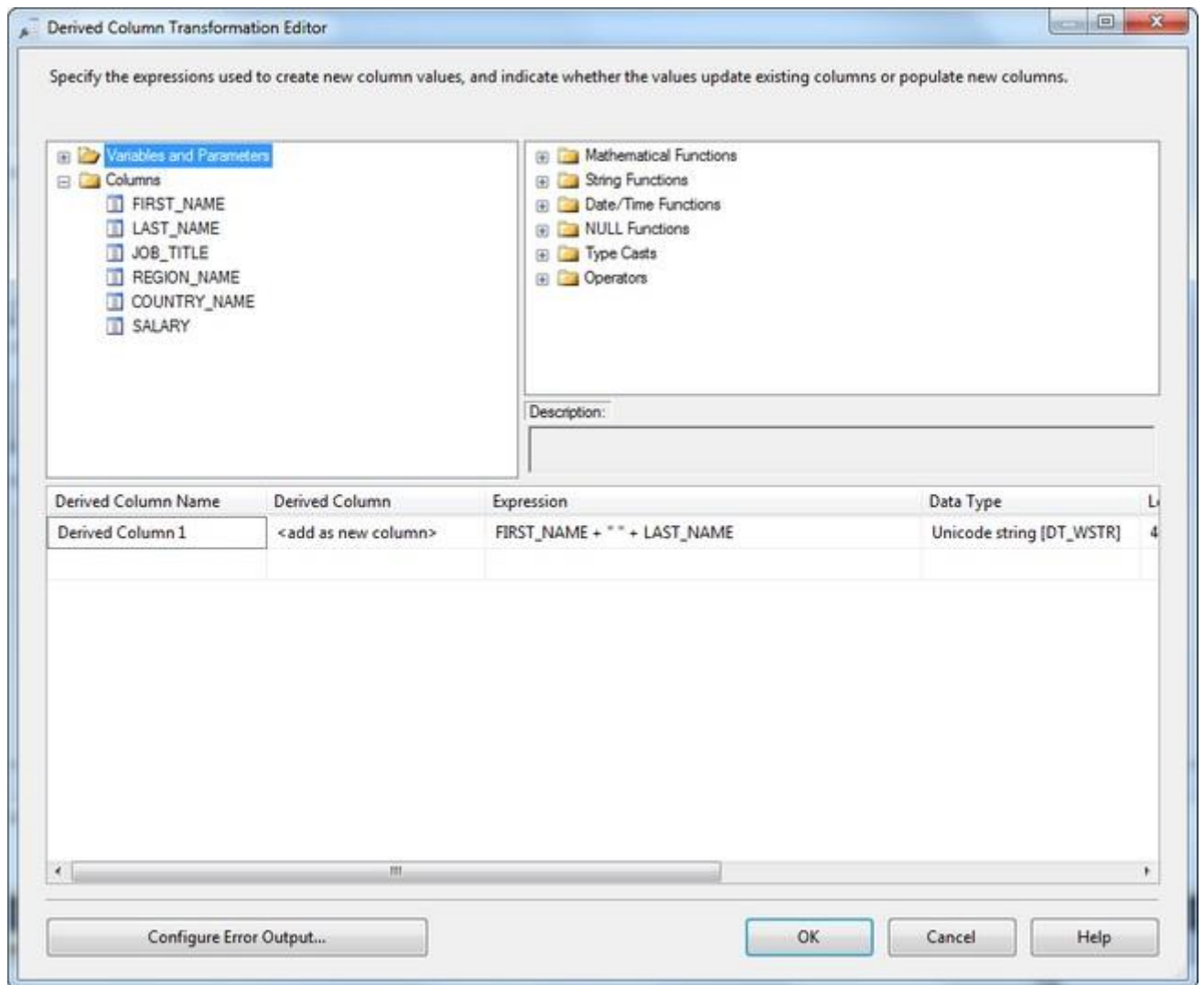
## Devart Oracle Source Configuration

Add a Devart Oracle Source to the Data Flow. Right click and select Edit. Create a query as the data source and click OK to Close.

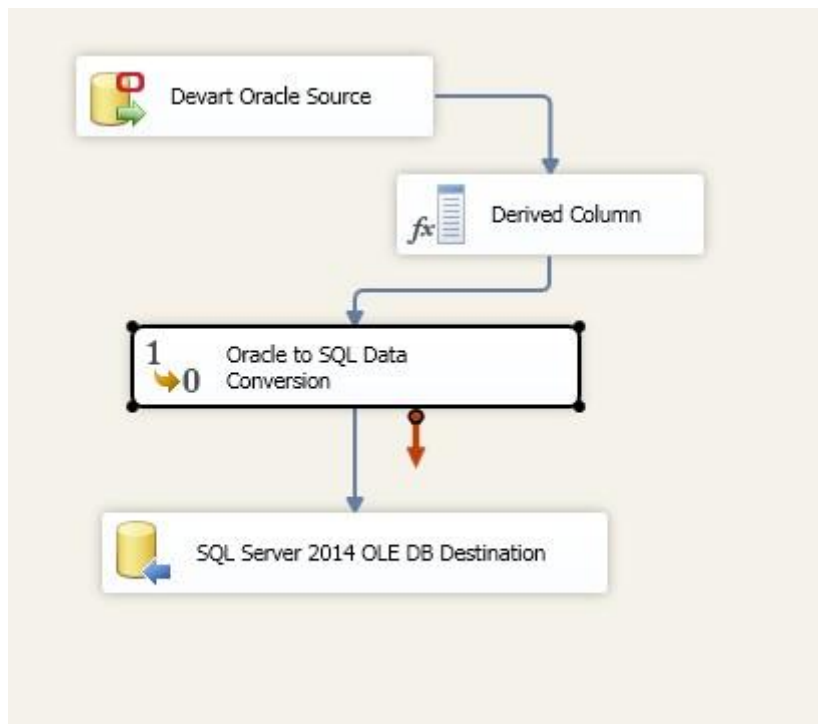


Add a Derived Column to the Data Flow tab and connect the Devart Oracle Source to it.

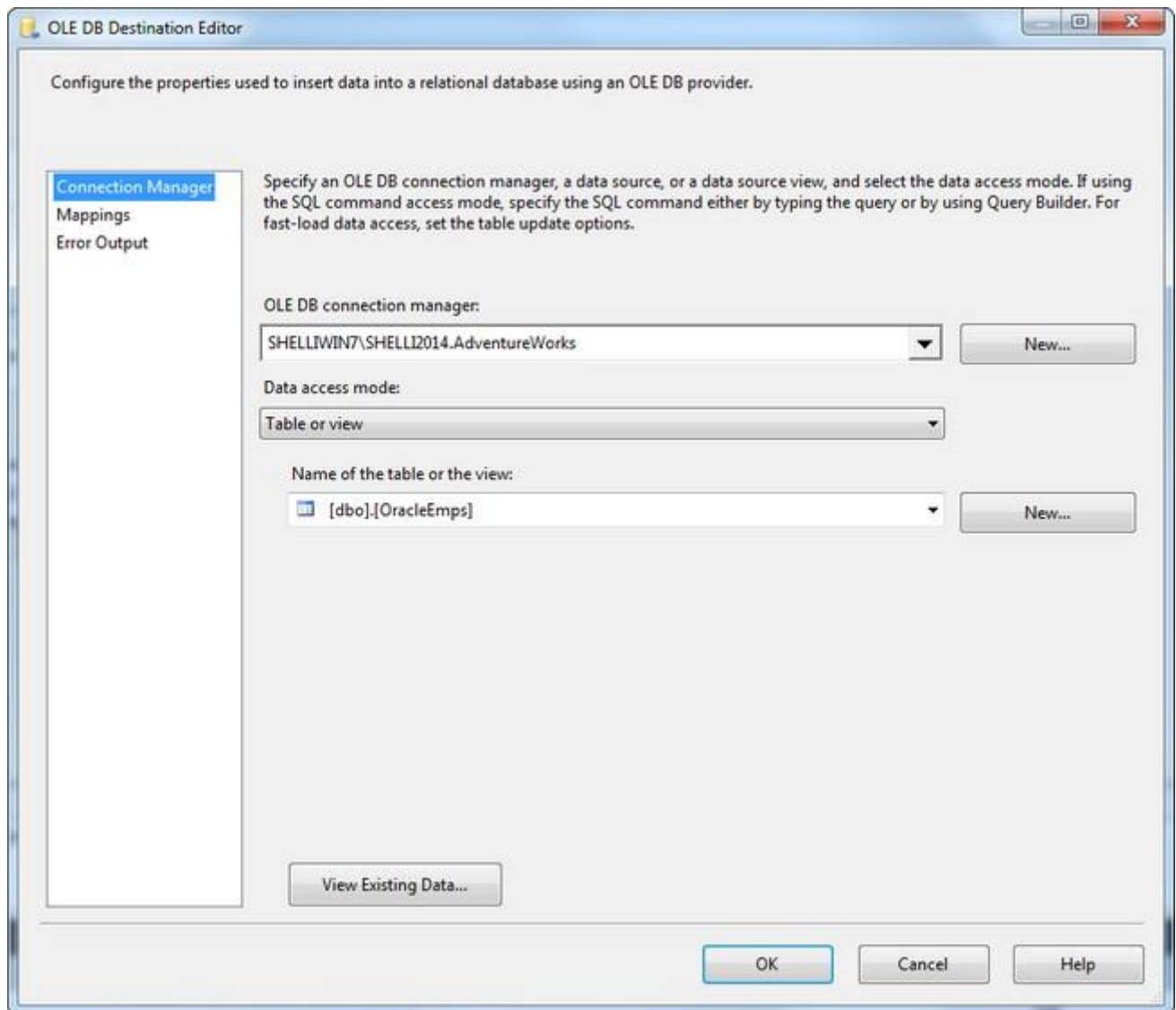
Right click to Edit the Derived Column. To Concatenate the First\_name and Last\_Name field, type `FIRST_NAME + " " + LAST_NAME` in the Expression box. Click OK to continue.



Add a Data Conversion task and SQL Server OLE DB Destination to the Data Flow tab and connect them.



Right click on the OLE DB Destination and click Edit. Select the SQL Server connection previously added and select the new table created in SSMS. Click OK to close. We will return to the Destination to setup the Mappings after fixing the Data Conversion step.



## Setting up the Data Conversion

Right click on the Data Conversion Transformation Editor and click Edit. In each of the boxes under Data Type, change from Unicode String [DT\_WSTR] to String [DT\_STR] and add the Length for each based on the Varchar lengths used in the CREATE TABLE statement in SSMS. The columns need to be converted from Unicode to Non Unicode strings to process the data into SQL Server. If they are left to the Default value of Unicode String, the task will fail.

Data Conversion Transformation Editor

Configure the properties used to convert the data type of an input column to a different data type. Depending on the data type to which the column is converted, set the length, precision, scale, and code page of the column.

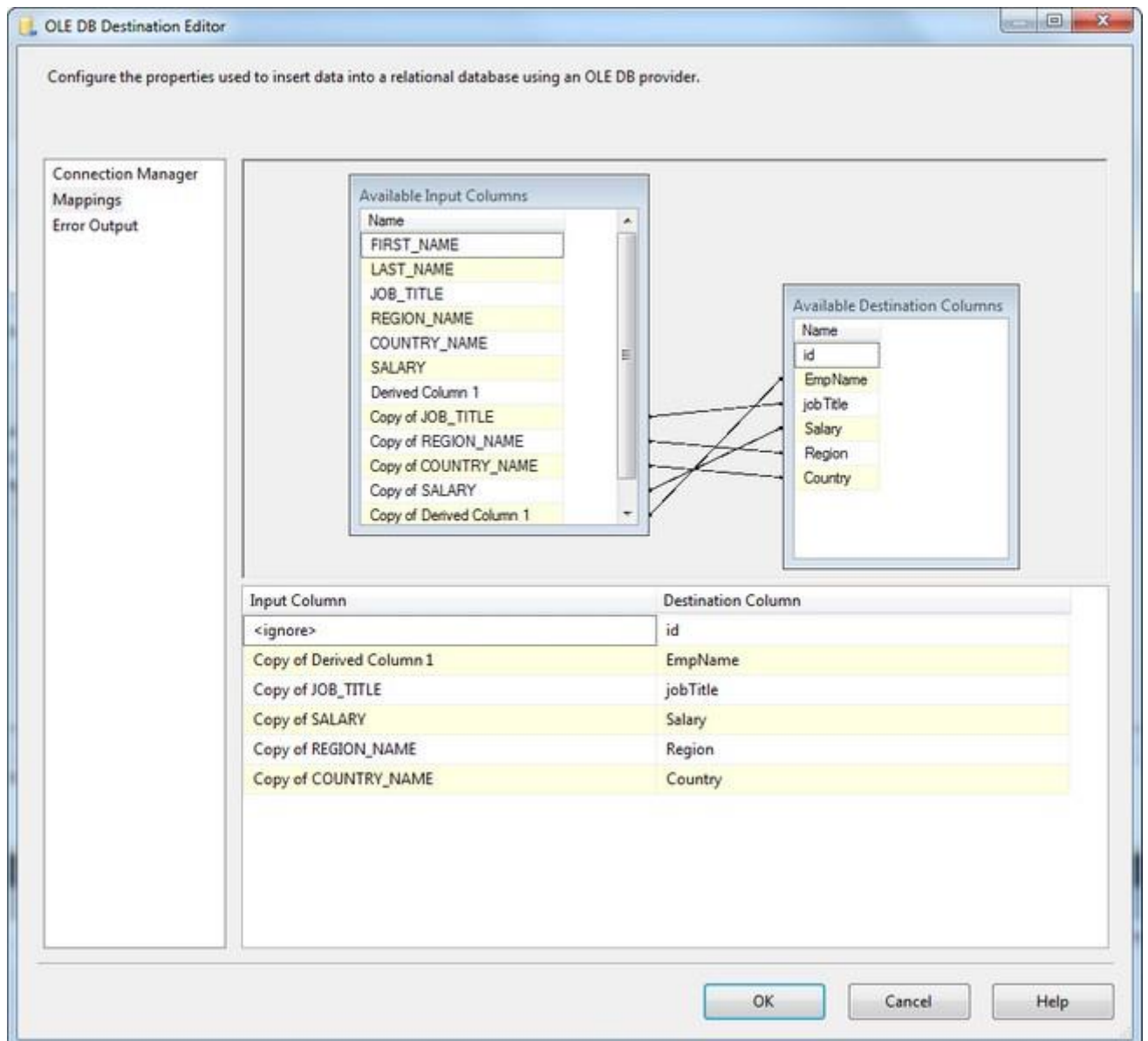
Available Input Columns

- ☒ Name
- ☐ FIRST\_NAME
- ☐ LAST\_NAME
- ☒ JOB\_TITLE
- ☒ REGION\_NAME
- ☒ COUNTRY\_NAME
- ☒ SALARY
- ☒ Derived Column 1

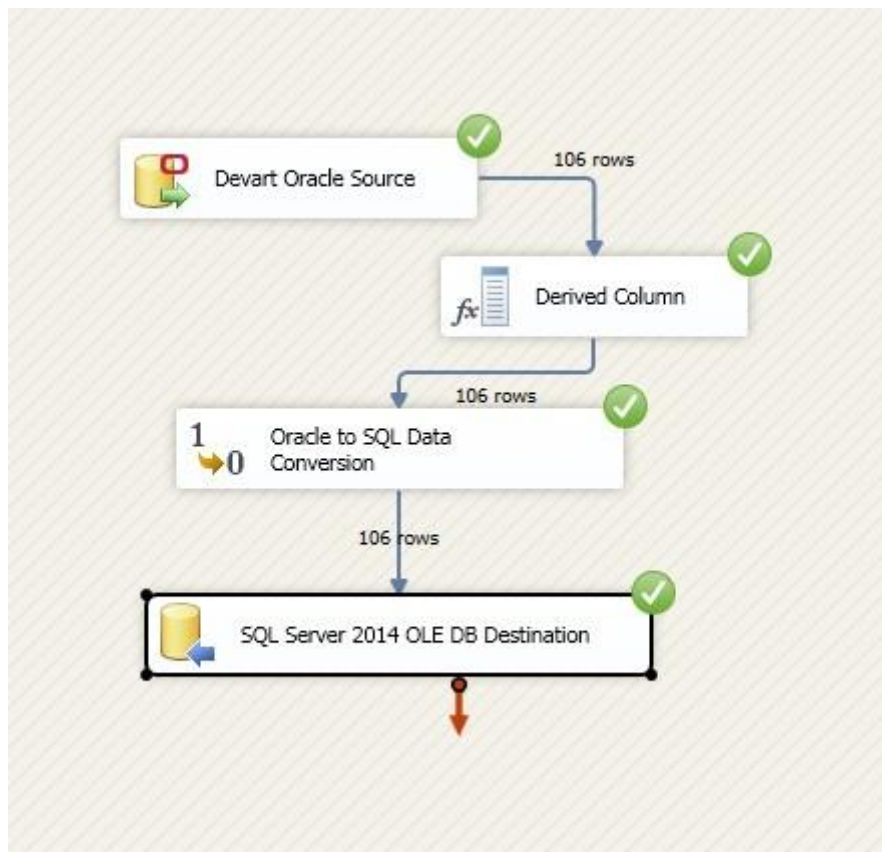
Input Column	Output Alias	Data Type	Length	Precision	Scale
JOB_TITLE	Copy of JOB_TITLE	string [DT_STR]	100		
REGION_NAME	Copy of REGION_NAME	string [DT_STR]	100		
COUNTRY_NAME	Copy of COUNTRY_NAME	string [DT_STR]	100		
SALARY	Copy of SALARY	double-precision float [DT_R8]			
Derived Column 1	Copy of Derived Colum...	string [DT_STR]	255		

Configure Error Output... OK Cancel Help

Right click the SQL OLE DB Destination and Select Edit and click on the Mappings. Click and map the Destination SQL Server Columns to the "Copy of" columns that are set to be Converted to Non Unicode Strings (i.e., Copy of Job\_Title, Copy of Region\_Name, etc.)



Run the job and verify that the data was transferred from the Oracle database through the Data Conversion task and out to SQL Server.



Run a Query in SSMS to verify the data transfer in the OracleEmps table.

```
select * from oracleemps;
```

	id	EmpName	jobTitle	Salary	Region	Country
1	1	Bruce Ernst	Programmer	6000.00	Americas	United States of America
2	2	David Austin	Programmer	4800.00	Americas	United States of America
3	3	Valli Pataballa	Programmer	4800.00	Americas	United States of America
4	4	Diana Lorentz	Programmer	4200.00	Americas	United States of America
5	5	Alexander Hunold	Programmer	9000.00	Americas	United States of America
6	6	Donald OConnell	Shipping Clerk	2600.00	Americas	United States of America
7	7	Douglas Grant	Shipping Clerk	2600.00	Americas	United States of America
8	8	Winston Taylor	Shipping Clerk	3200.00	Americas	United States of America
9	9	Jean Fleaur	Shipping Clerk	3100.00	Americas	United States of America
10	10	Martha Sullivan	Shipping Clerk	2500.00	Americas	United States of America
11	11	Girard Geoni	Shipping Clerk	2800.00	Americas	United States of America
12	12	Nandita Sarchand	Shipping Clerk	4200.00	Americas	United States of America
13	13	Alexis Bull	Shipping Clerk	4100.00	Americas	United States of America
14	14	Julia Dellinger	Shipping Clerk	3400.00	Americas	United States of America

