



UNIVERSIDAD PRIVADA DE TACNA
FACULTAD DE INGENIERÍA
Escuela Profesional de Ingeniería de Sistemas
Laboratorio 02
“Importación, Data Flow, Traslado de archivos”

CURSO:

Inteligencia de Negocios

DOCENTE:

Mag. Patrick Jose Cuadros Quiroga

ALUMNO:

Lipa Calabilla, Abraham

(2019064039)

Tacna - Perú

2022



Índice

1. OBJETIVOS	2
2. REQUERIMIENTOS	2
3. CONSIDERACIONES INICIALES	2
4. DESARROLLO	3
4.1. Tarea 1: Importación de datos usando el Wizard - SQL Management	3
4.2. Tarea 2: Creamos nuestro primer paquete DTSX	5
4.3. Cargar datos de Clientes en la Tabla Customer de BD Adventure Work.	12
4.4. Ejemplo: conexión con AdventureWork (INSERTAR DATOS EN UNA TABLA) . .	12
5. CONCLUSIONES	16



IMPORTACIÓN, DATA FLOW, TRASLADO DE ARCHIVOS

1. OBJETIVOS

- Importar datos usando el Wizard
- Desarrollar mis primeros paquetes DTSX

2. REQUERIMIENTOS

- Conocimientos
Para el desarrollo de esta práctica se requerirá de los siguientes conocimientos básicos:
 - Conocimientos básicos de administración de base de datos Microsoft SQL Server.
 - Conocimientos básicos de Visual Studio 2019.
- Software
Así mismo se necesitan los siguientes aplicativos:
 - SQL Server Integration Services
 - Base de datos Adventure Work (OLTP Y Datawarehouse) y AdventureWorksLT2012
 - Microsoft SQL Server 2016 o superior
 - Visual Studio 2019

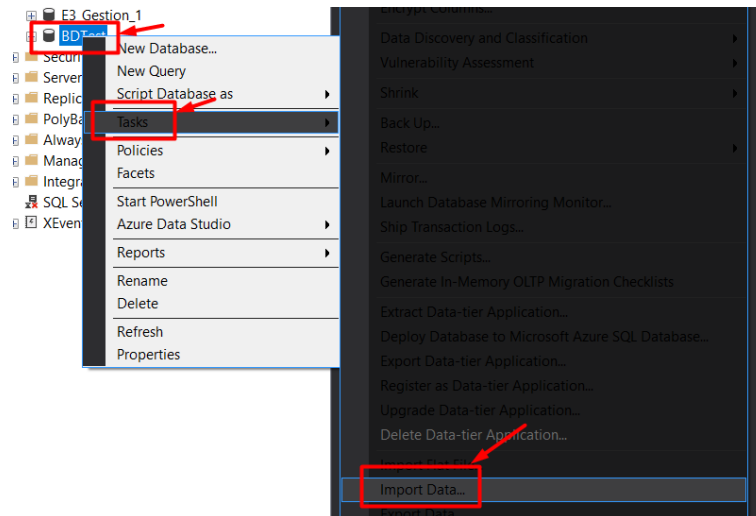
3. CONSIDERACIONES INICIALES

- Tener todos los componentes considerados en los requerimientos de modo que se puedan crear Integration Services Projects

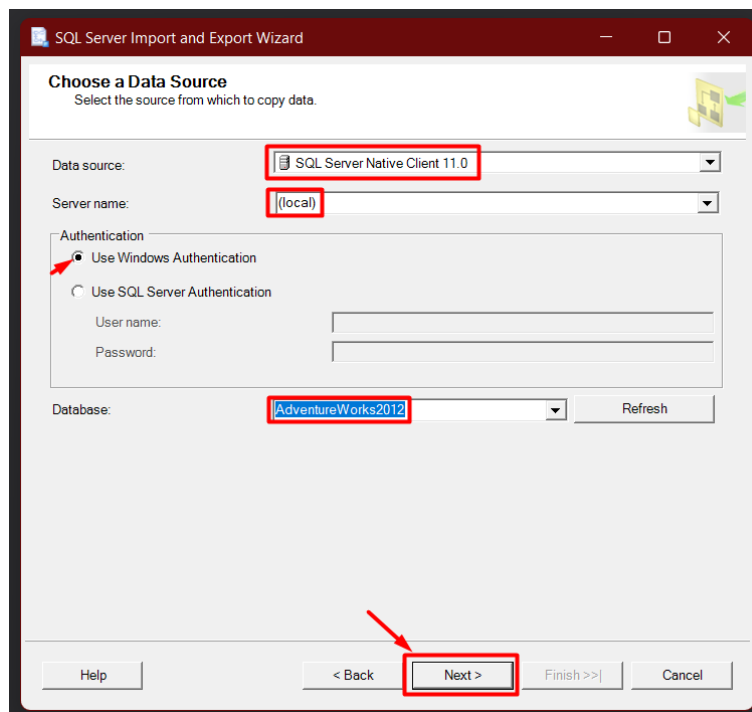
4. DESARROLLO

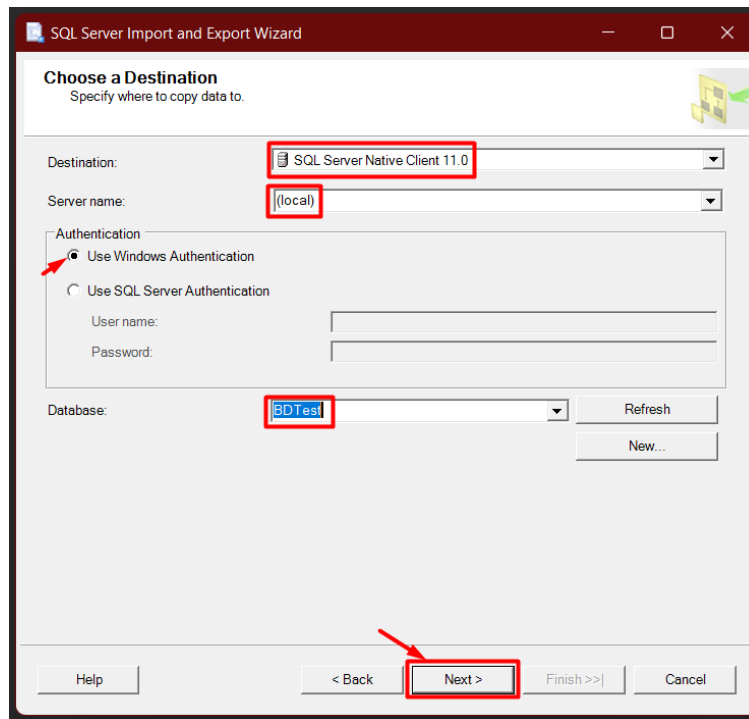
4.1. Tarea 1: Importación de datos usando el Wizard - SQL Management

Crear una base de datos - BDTEST Importar Datos desde AdventureWorks



Next y escribir el Servidor y seleccionar la base de datos





Data Source: La base de donde vamos a importar

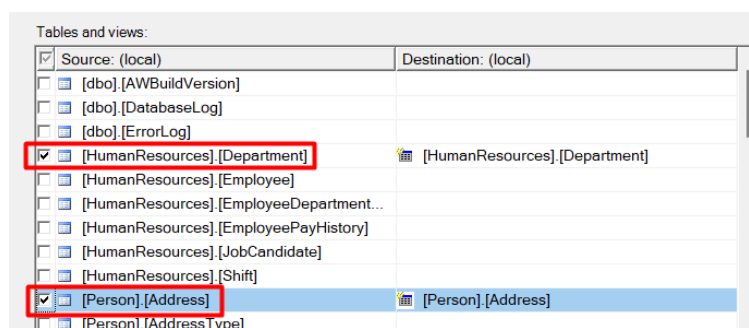
Destination: La Base donde vamos a cargar la data

Seleccionamos la primera opción

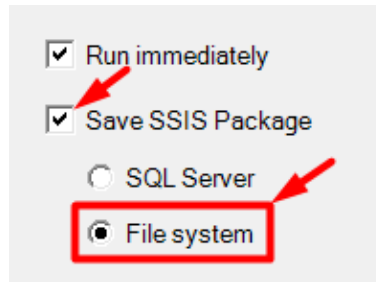
Copy data from one or more tables or views

Use this option to copy all the data from the existing tables or views in the source database.

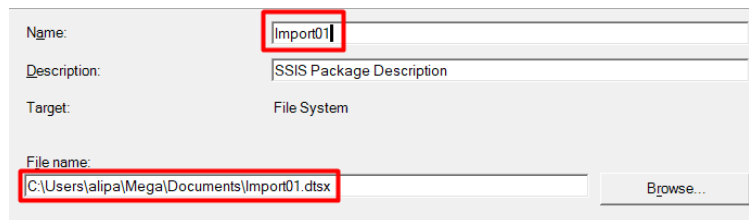
Seleccionamos las dos tablas indicadas



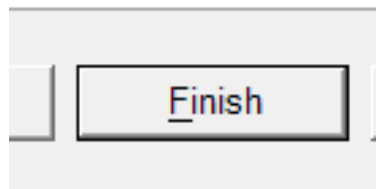
Seleccionamos la opción para guardar el paquete SSIS como archivo



Ingresamos el nombre y la ubicación



Hacemos clic en Finish



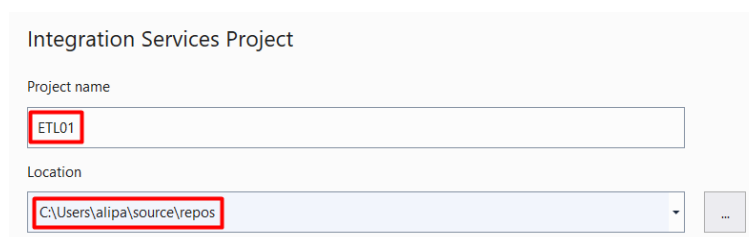
Al finalizar, tenemos el resumen de la ejecución.

- Hemos Generado nuestro 1 paquete de forma automática.
- Podemos actualizar la base de datos
- BdTest y encontraremos las tablas ya importadas.

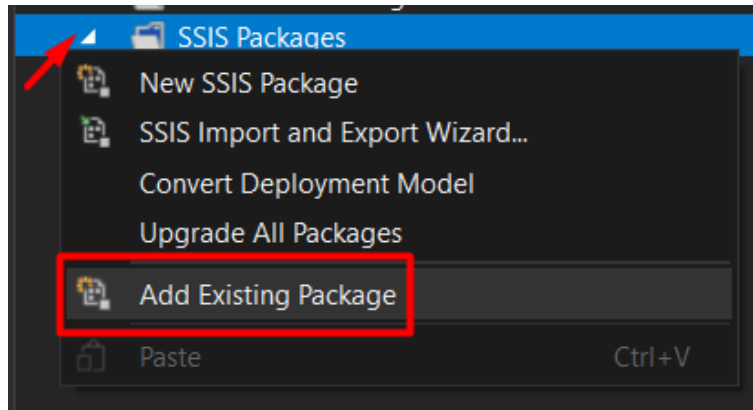
Con este procedimiento puede usted importar 5 tablas de BD Adventure Works a nuestra base de datos.

4.2. Tarea 2: Creamos nuestro primer paquete DTSX

Creamos un nuevo Integration Services Project



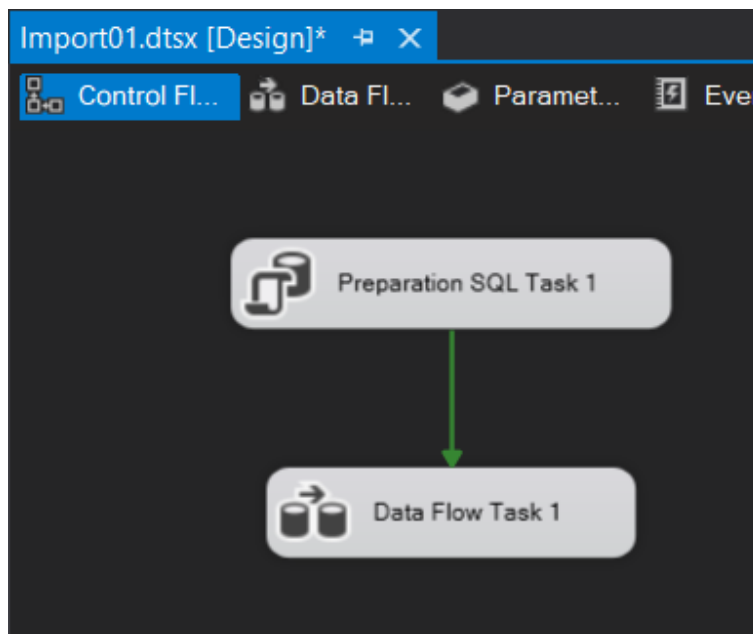
Añadimos un paquete SSIS



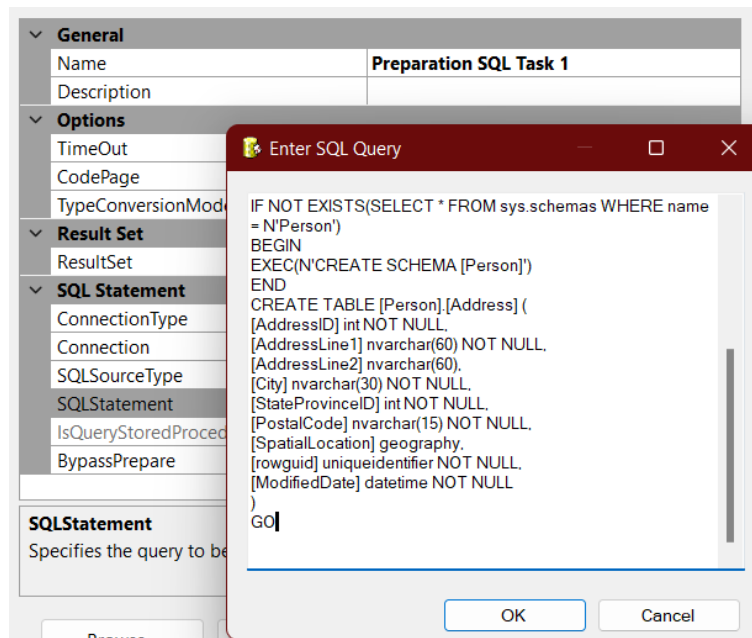
Buscamos el paquete creado y hacemos clic en Ok



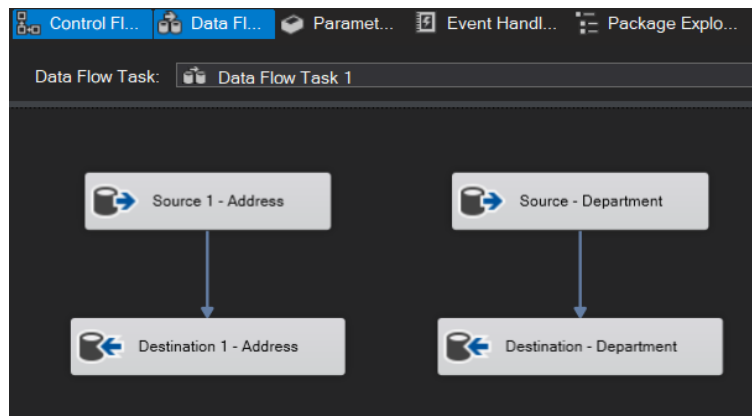
Nos mostrará el diseño del paquete



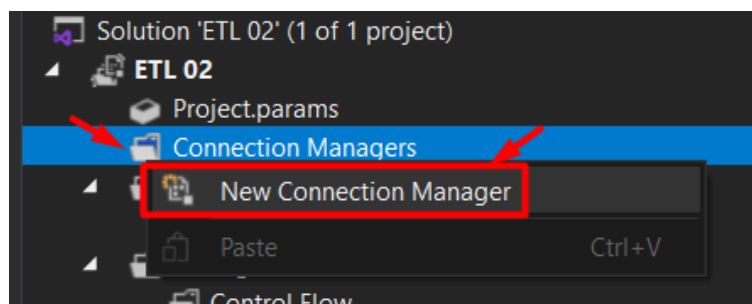
Al hacer clic en el primer control, podemos ver las propiedades incluyendo el código SQL



Al hacer clic sobre el otro control, vemos el flujo de datos



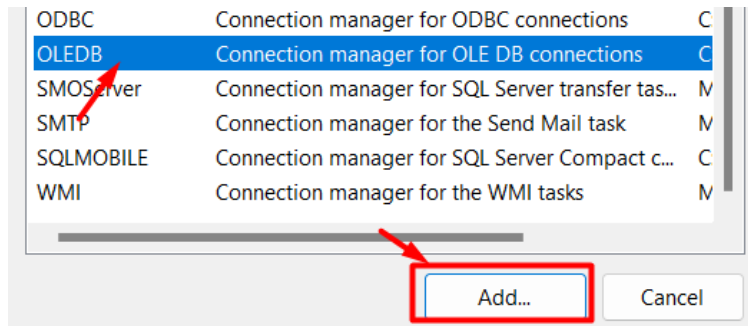
Creamos un nuevo Integration Services Project, hacemos clic derecho en Connection Managers y luego clic en New Connection Manager



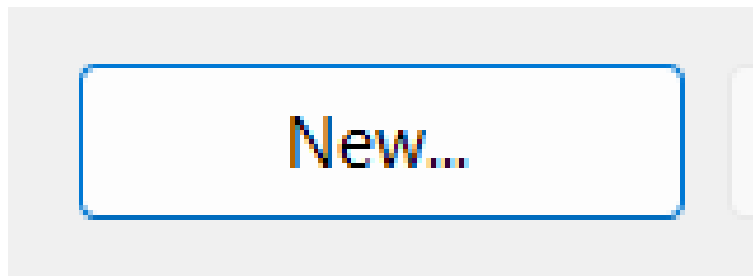
Seleccionamos OLEDB y luego Add...



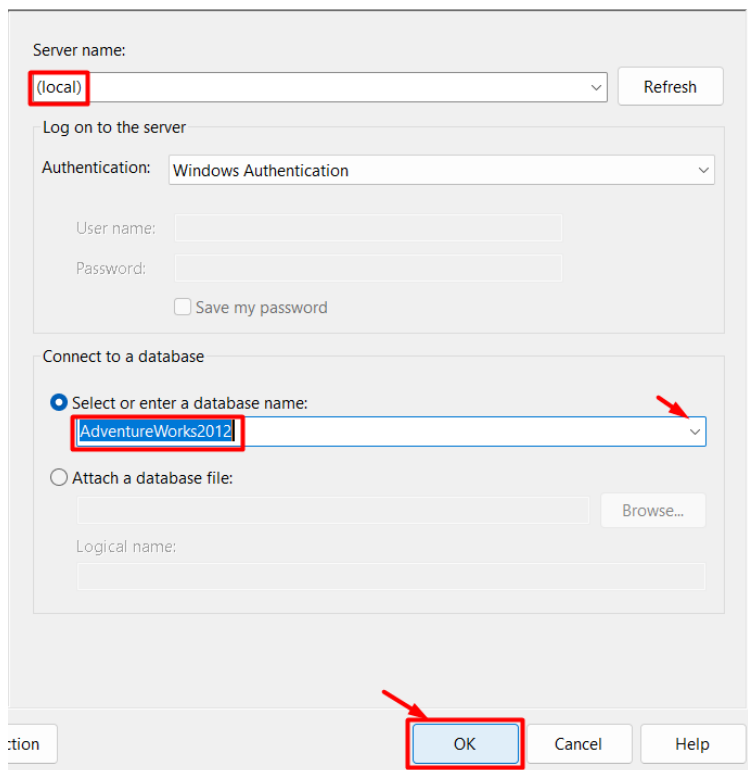
Laboratorio 02 - Importación, Data Flow, Traslado de archivos



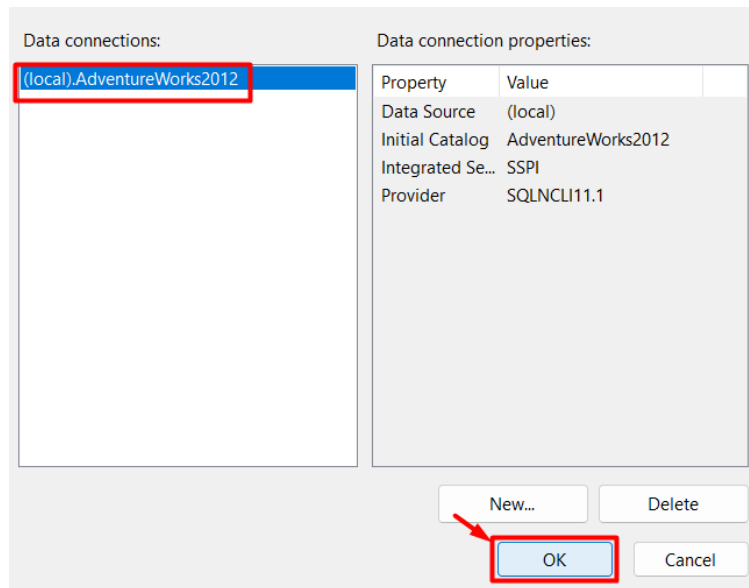
Hacemos clic en New...



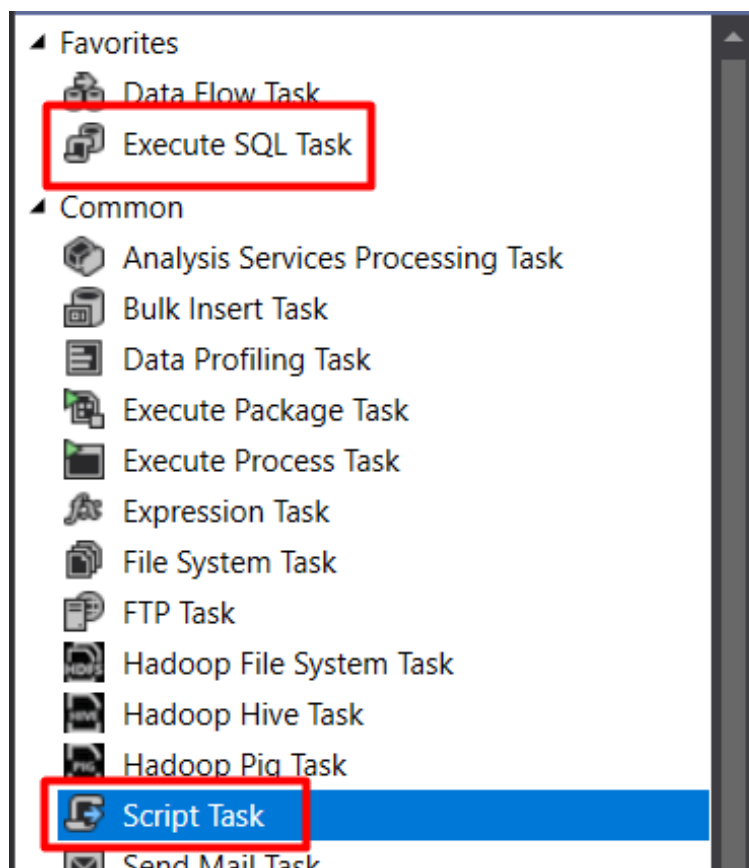
Buscamos la Base de datos de AdventureWorks2012 y luego clic en Ok



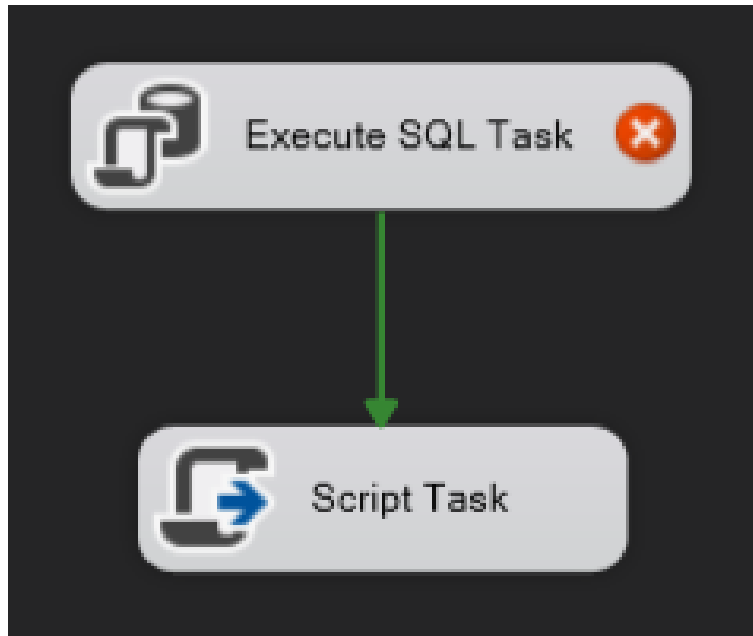
Hacemos clic en Ok



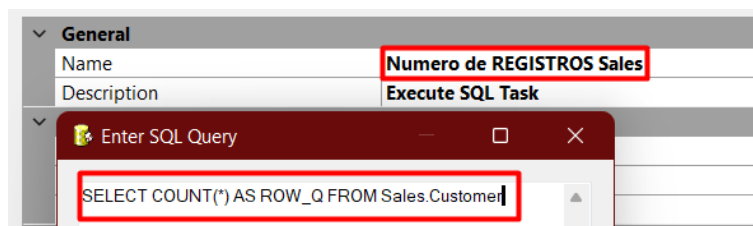
Agregamos los controles indicados



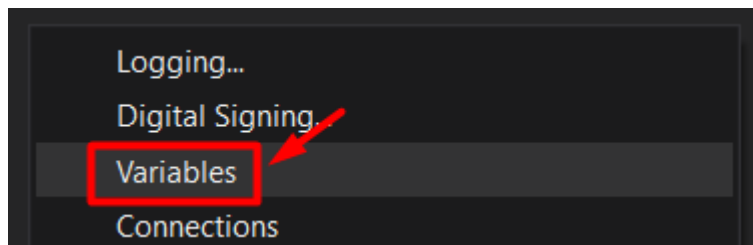
Los enlazamos y quedarían así



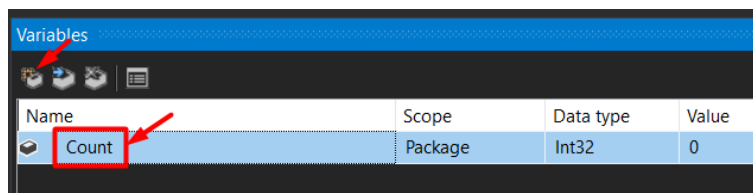
Hacemos clic en el control, le cambiamos el nombre y la consulta SQL



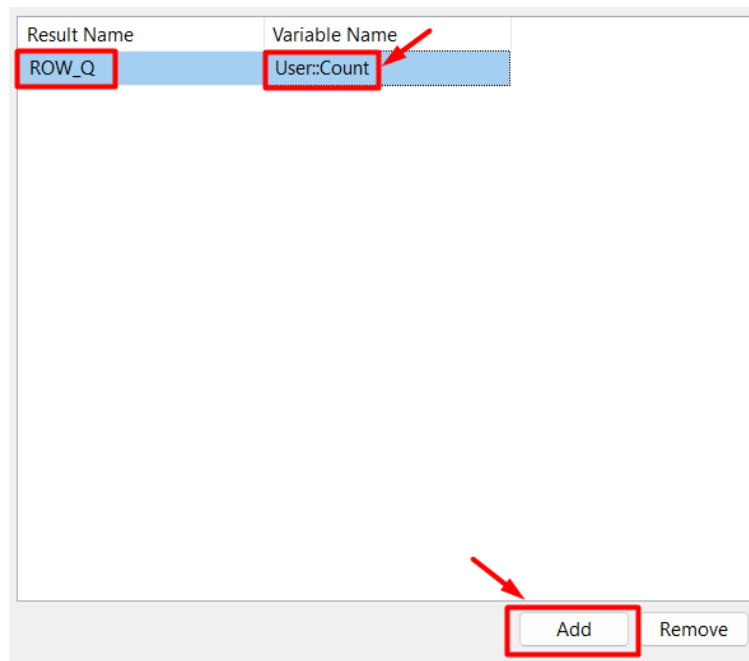
Hacemos clic derecho en un lugar vacío y luego Variables



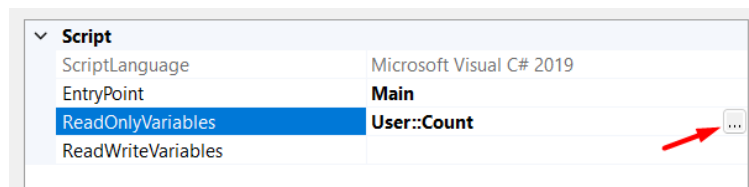
Agregamos una nueva variable Count



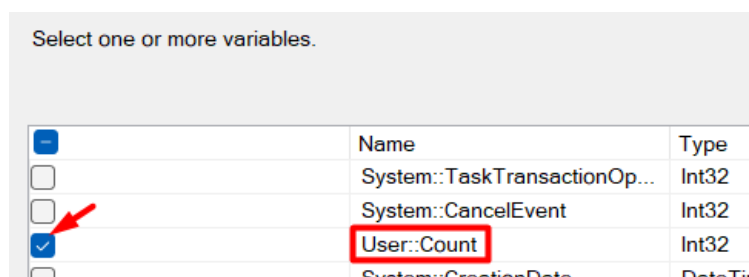
En las opciones del Numero de REGISTROS Sales, hacemos clic en Result Set y agregamos uno



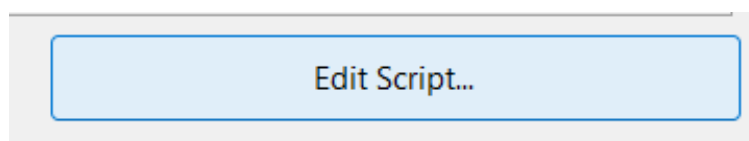
Hacemos clic en el control Script Task y hacemos clic en los tres puntos de ReadOnlyVariables



Seleccionamos User::Count



Hacemos clic en Edit Script...



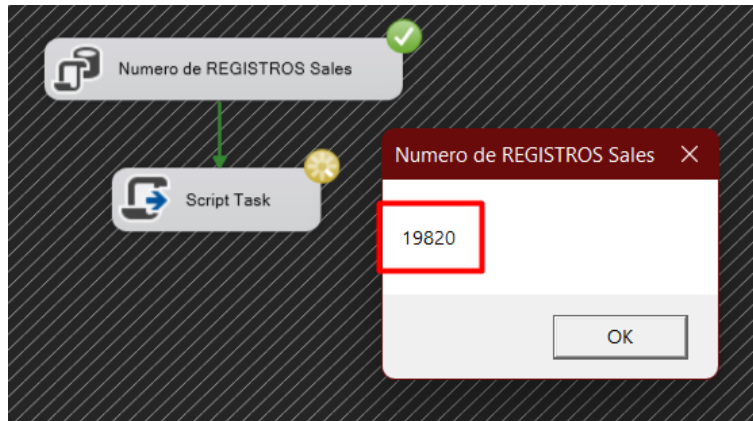
Agregamos el código en el método Main() para que muestre el valor de la variable en un MessageBox



```
public void Main()
{
    // TODO: Add your code here
    Dts.TaskResult = (int)ScriptResults.Success;

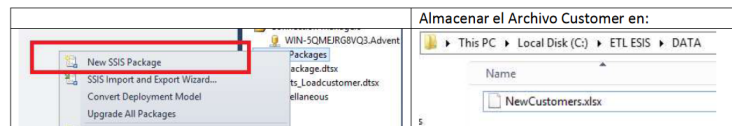
    MessageBox.Show(Dts.Variables[0].Value.ToString(), "Numero de REGISTROS Sales", MessageBoxButtons.OK);
}
```

Al ejecutar el paquete, podemos ver una ventana con el numero de registros

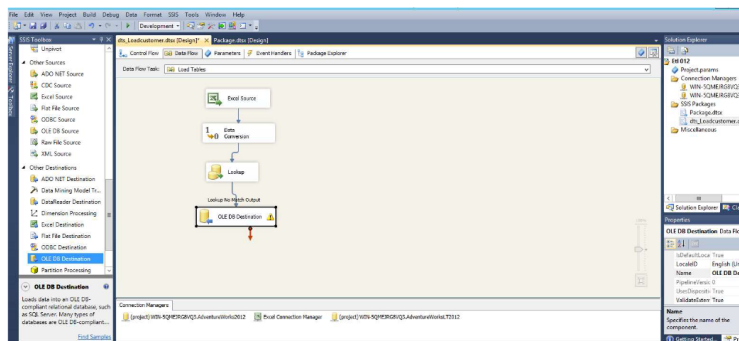


4.3. Cargar datos de Clientes en la Tabla Customer de BD Adventure Work.

Crear una nueva conexión para la base de datos AdventureWorksLT2012
Crear un paquete nuevo Paquete.

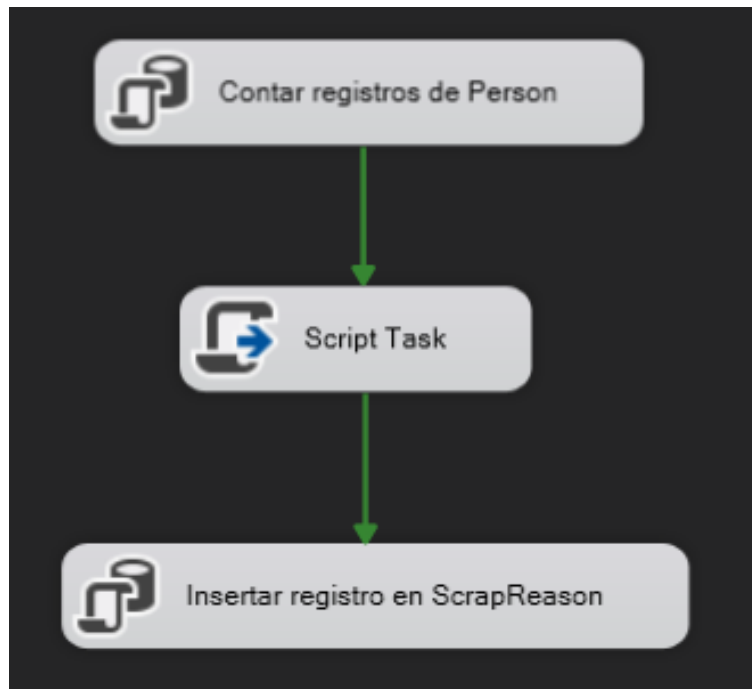


Este es el Diagrama Final que lo haremos Juntos en CLASE.

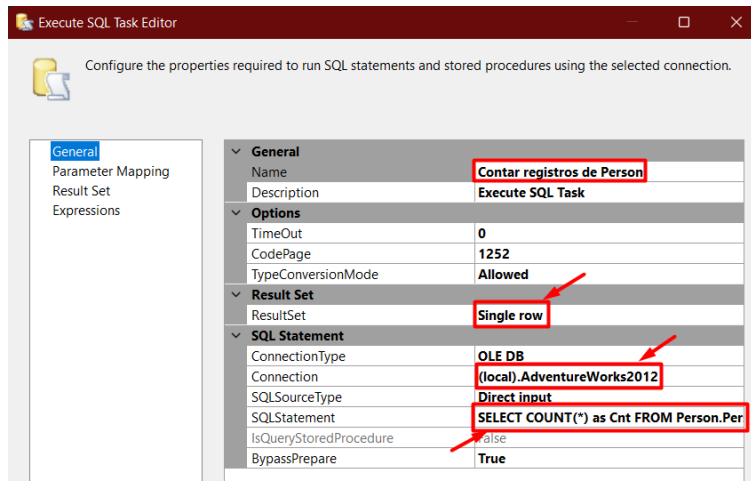


4.4. Ejemplo: conexión con AdventureWork (INSERTAR DATOS EN UNA TABLA)

Agregamos los componentes indicados previamente



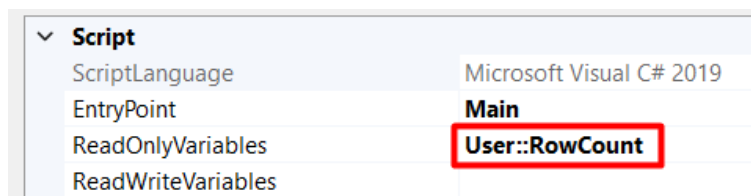
Cambiamos el nombre, el Result Set a Single Row, elegimos la conexión y escribimos el código SQL indicado



En Result Set, agregamos la variable de lectura

Result Name	Variable Name
Cnt	User::RowCount

Hacemos clic en Script Task y en ReadOnlyVariables elegimos la variable RowCount





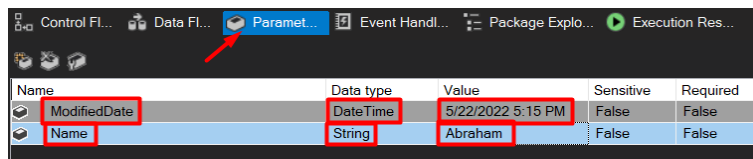
Hacemos clic en Edit Script... y agregamos el código en el método Main()

```
public void Main()
{
    // TODO: Add your code here

    Dts.TaskResult = (int)ScriptResults.Success;

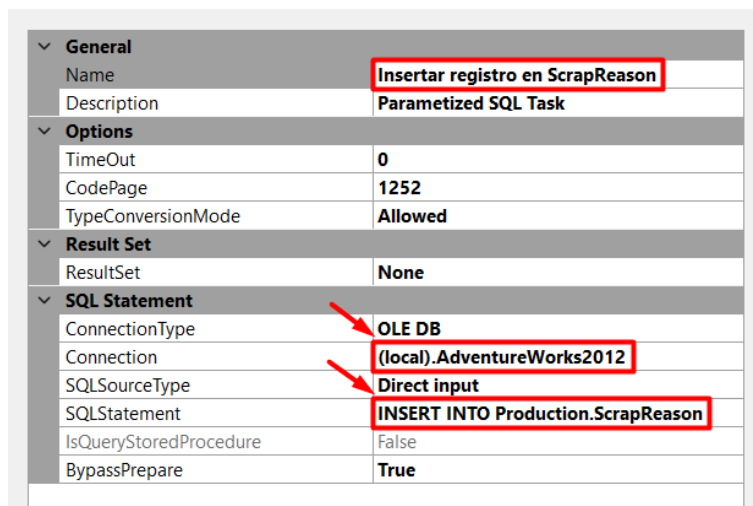
    MessageBox.Show(Dts.Variables["User::RowCount"].Value.ToString());
}
```

Hacemos clic en el botón Parameters y agregamos los dos parámetros con las propiedades



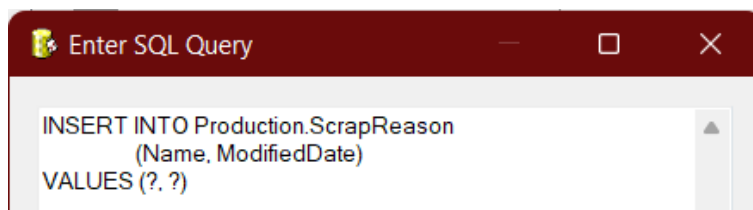
Name	Data type	Value	Sensitive	Required
ModifiedDate	DateTime	5/22/2022 5:15 PM	False	False
Name	String	Abraham	False	False

Hacemos clic en la última tarea, cambiamos el nombre y seleccionamos la conexión



General	
Name	Insertar registro en ScrapReason
Description	Parametized SQL Task
Options	
TimeOut	0
CodePage	1252
TypeConversionMode	Allowed
Result Set	
ResultSet	None
SQL Statement	
ConnectionType	OLE DB
Connection	(local).AdventureWorks2012
SQLSourceType	Direct input
SQLStatement	INSERT INTO Production.ScrapReason
IsQueryStoredProcedure	False
BypassPrepare	True

Ingresamos el siguiente código SQL



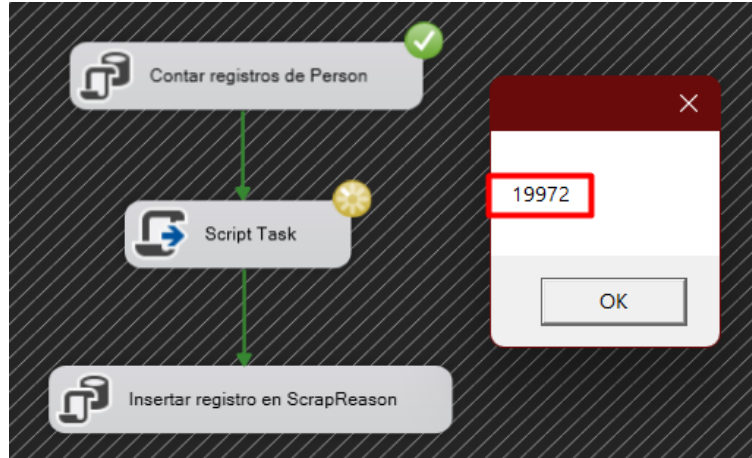
```
INSERT INTO Production.ScrapReason
    (Name, ModifiedDate)
VALUES (?, ?)
```

En el apartado Parameter Mapping, agregamos dos variables con las siguientes propiedades:

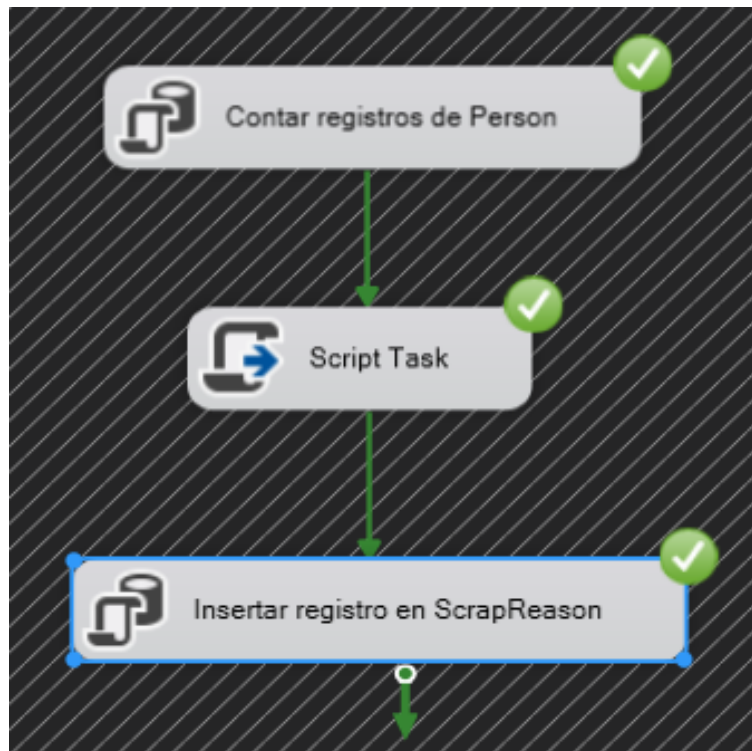
Variable Name	Direction	Data Type	Paramet...	Paramet...
\$Package::ModifiedDate	Input	DATE	0	-1
\$Package::Name	Input	NVARCHAR	1	-1



Ejecutamos el paquete y veremos que se muestra el número de registros en Person.Person



Finalmente, todas las tareas se ejecutan correctamente



En la base de datos, se ve el registro insertado

	ScrapReasonID	Name	ModifiedDate
1	19	Abraham	2022-05-22 17:15:00.000
2	1	Brake assembly not as ordered	2008-04-30 00:00:00.000
3	2	Color incorrect	2008-04-30 00:00:00.000
4	3	Gouge in metal	2008-04-30 00:00:00.000
5	4	Drill pattern incorrect	2008-04-30 00:00:00.000



5. CONCLUSIONES

- Se logró crear la base de datos para la inserción de los datos
- Se creó la conexión necesaria para que las tareas se apliquen a la BD
- Se crearon las tareas con cálculos y operaciones de la BD