GEOGRAPHIC DATA INTEGRATION AND ANALYSIS PROJECT USING SSIS

ENG. Silva Parraguez Maximo

I. UNDERSTANDING THE DATA AND THE PROJECT.

1. Data source:

For this project, Geographic data will be needed, such as Shapefiles (files that store geographic data) and GPS data.

- Shapefiles: These will be useful for representing transportation routes, geographic boundaries, etc.
- GPS data: I will use this data to analyze specific movements or locations.

To carry out this project, I have collected two main types of geographic data related to Mexico City: Shapefiles and GPS data.

- **Shapefiles** :

  Shapefiles are files containing vector geographic information, such as lines, points, and polygons, that represent geographic features. In this project, I will be using Shapefiles to delineate transportation routes and geographic boundaries within Mexico City. I obtained these files from **Geofabrik , a company that offers OpenStreetMap** data in different formats. Specifically, I downloaded the Shapefiles from their download server for Mexico https://download.geofabrik.de/north-america.html . These files provide a detailed representation of road infrastructure, urban areas, and other geographic features relevant to the analysis.

- **GPS data** :

  To analyze specific movements and locations, I turned to a dataset of taxi routes in Mexico City. This dataset , available on Kaggle https://www.kaggle.com/datasets/mnavas/taxi-routes-for-mexico-city-and-quito?select=mex_clean.csv was collected using the EC Taximeter app between June 2016 and July 2017 and contains detailed information on taxi routes, including GPS coordinates, travel times, and distances traveled. This data will allow me to study mobility patterns and transport behavior in the city, providing a practical and up-to-date perspective to the geospatial analysis of the project.

2. Aim:

The objective of this project is to develop an ETL process in SSIS to integrate, transform and analyze geographic data related to public transport routes, in order to generate useful insights for route analysis, optimization and decision making.

II. PROJECT DEVELOPMENT

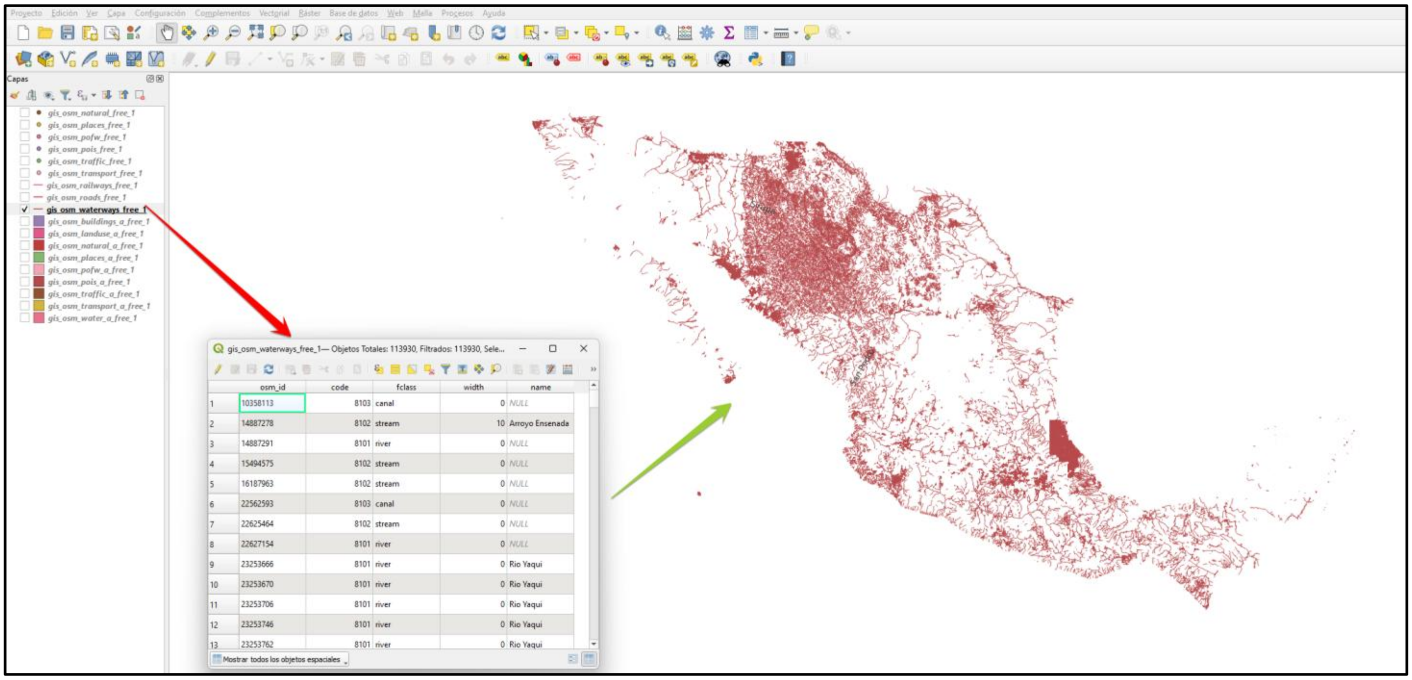
Step 1: Exploring Shapefiles with QGIS:

To examine the data from the Shapefiles I will use the **QGIS application** to visualize the geographic data, as well as its tables and records.

The following files are observed:

1. **gis_osm_natural_free_1:** Data related to natural features (e.g. rivers, lakes, mountains).
2. **gis_osm_places_free_1:** Information about places (e.g. cities, towns).
3. **gis_osm_pofw_free_1:** Points of worship or places of worship (e.g. churches, mosques).
4. **gis_osm_pois_free_1:** Points of interest (e.g. restaurants, hotels).
5. **gis_osm_traffic_free_1:** Traffic related data (e.g. traffic lights, signs).
6. **gis_osm_transport_free_1:** Transport information (e.g. bus stops, train stations).
7. **gis_osm_railways_free_1:** Data on railways.
8. **gis_osm_roads_free_1:** Information about roads and streets.
9. **gis_osm_waterways_free_1:** Data on waterways (e.g. rivers, canals).
10. **gis_osm_buildings_a_free_1:** Buildings.
11. **gis_osm_landuse_a_free_1:** Land use (e.g. residential, industrial areas).
12. **gis_osm_natural_a_free_1:** Natural features in more detail.
13. **gis_osm_places_a_free_1:** Places in more detail.
14. **gis_osm_pofw_a_free_1:** Cult points in more detail.
15. **gis_osm_pois_a_free_1:** Points of interest in more detail.
16. **gis_osm_traffic_a_free_1:** Traffic data in more detail.
17. **gis_osm_transport_a_free_1:** Transport information with more detail.
18. **gis_osm_water_a_free_1:** Data related to water (e.g. lakes, oceans).

For example, in the layers section, I selected the **gis_osm_waterways_free_1 file** , and the file data and also the respective graph are displayed, so, by exploring everything, I get an idea of everything that all the Shapefiles contain .



gis_osm_waterways_free_1— Objetos Totales: 113930, Filtrados: 113930, Sele...

	osm_id	code	fclass	width	name
1	10358113	8103	canal	0	NULL
2	14887278	8102	stream	10	Arroyo Ensenada
3	14887291	8101	river	0	NULL
4	15494575	8102	stream	0	NULL
5	16187963	8102	stream	0	NULL
6	22562593	8103	canal	0	NULL
7	22625464	8102	stream	0	NULL
8	22627154	8101	river	0	NULL
9	23253666	8101	river	0	Rio Yaqui
10	23253670	8101	river	0	Rio Yaqui
11	23253706	8101	river	0	Rio Yaqui
12	23253746	8101	river	0	Rio Yaqui
13	23253762	8101	river	0	Rio Yaqui

Step 2: Exploring GPS data:

Taxi route data in the CSV contains the following columns:

- **id:** Unique identifier of the trip.
- **vendor_id :** Taxi company.
- **pickup_datetime :** Start date and time.
- **dropoff_datetime :** End date and time.
- **pickup_length :** Start Length.
- **pickup_latitude :** Start Latitude.
- **dropoff_length :** End length.
- **dropoff_latitude :** End latitude.
- **store_and_fwd_flag :** Store and forward data flag.
- **trip_duration :** Trip duration in seconds.
- **dist_meters :** Distance traveled in meters.
- **wait_sec :** Timeout in seconds.

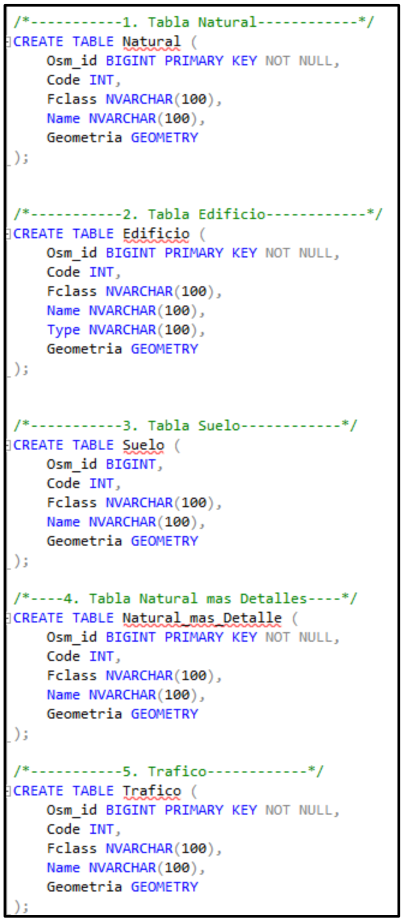
Step 3: Prepare the Database and Tables:

Now I created a database called " **GEOGRAPHIC_DATES ", where I will place all the Shapefiles** files , each one in a different table and also a table with the **GPS data** .

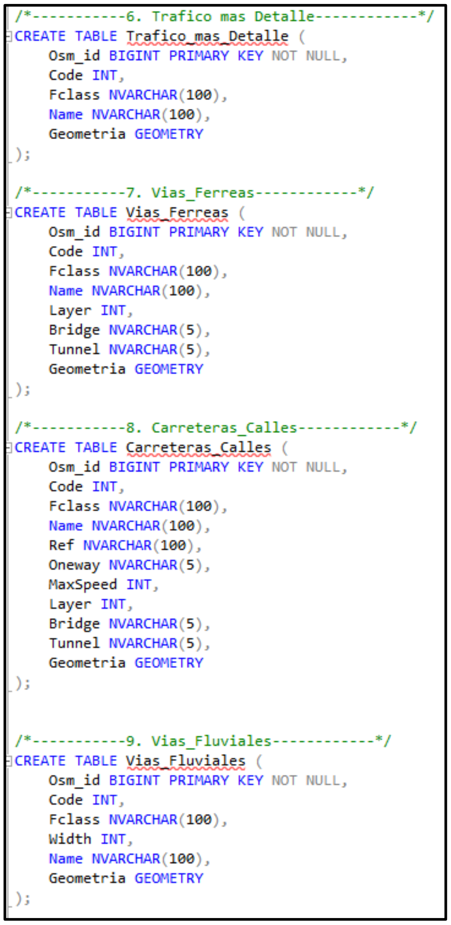
a) Shapefiles data I created 18 tables as follows:

Shapefile	Table in Database
gis_osm_natural_free_1	Natural
gis_osm_buildings_a_free_1	Edificio
gis_osm_landuse_a_free_1	Suelo
gis_osm_natural_a_free_1	Natural_mas_Detalle
gis_osm_traffic_free_1	Trafico

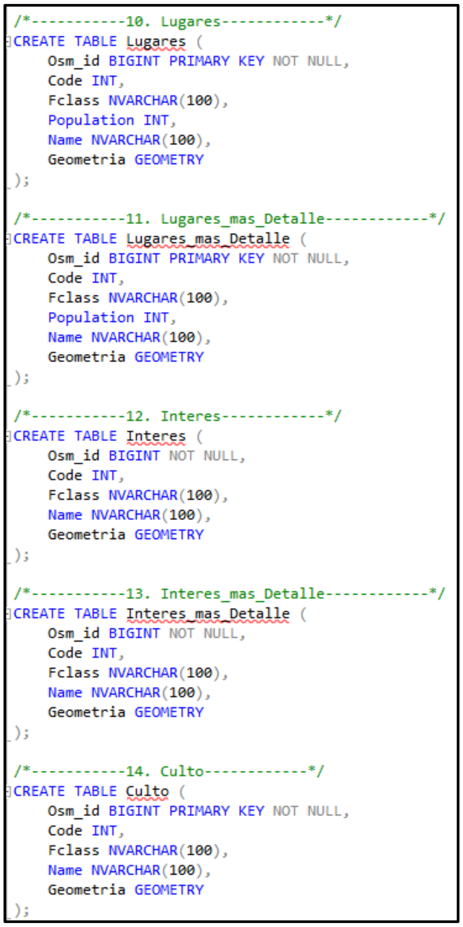
gis_osm_traffic_a_free_1	Trafico_mas_Detalle
gis_osm_railways_free_1	Vias_Ferreas
gis_osm_roads_free_1	Carretera_Calle
gis_osm_waterways_free_1	Vias_Fluviales
gis_osm_places_free_1	Lugares
gis_osm_places_a_free_1	Lugares_mas_Detalle
gis_osm_pois_free_1	Interes
gis_osm_pois_a_free_1	Interes_mas_Detalle
gis_osm_pofw_free_1	Culto
gis_osm_pofw_a_free_1	Culto_mas_Detalle
gis_osm_transport_free_1	Transporte
gis_osm_transport_a_free_1	Transporte_mas_Detalle
gis_osm_water_a_free_1	Agua



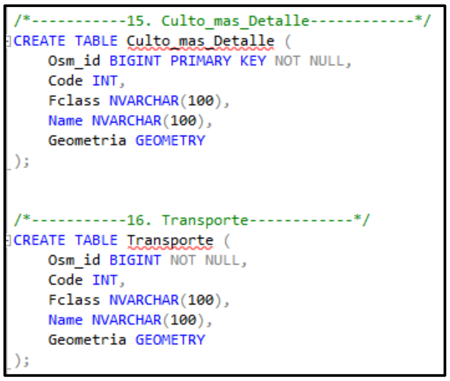
```
/*-----------1. Tabla Natural------------*/
CREATE TABLE Natural (
    Osm_id BIGINT PRIMARY KEY NOT NULL,
    Code INT,
    Fclass NVARCHAR(100),
    Name NVARCHAR(100),
    Geometria GEOMETRY
);


/*-----------2. Tabla Edificio------------*/
CREATE TABLE Edificio (
    Osm_id BIGINT PRIMARY KEY NOT NULL,
    Code INT,
    Fclass NVARCHAR(100),
    Name NVARCHAR(100),
    Type NVARCHAR(100),
    Geometria GEOMETRY
);


/*-----------3. Tabla Suelo------------*/
CREATE TABLE Suelo (
    Osm_id BIGINT,
    Code INT,
    Fclass NVARCHAR(100),
    Name NVARCHAR(100),
    Geometria GEOMETRY
);

/*----4. Tabla Natural mas Detalles----*/
CREATE TABLE Natural_mas_Detalle (
    Osm_id BIGINT PRIMARY KEY NOT NULL,
    Code INT,
    Fclass NVARCHAR(100),
    Name NVARCHAR(100),
    Geometria GEOMETRY
);

/*-----------5. Trafico------------*/
CREATE TABLE Trafico (
    Osm_id BIGINT PRIMARY KEY NOT NULL,
    Code INT,
    Fclass NVARCHAR(100),
    Name NVARCHAR(100),
    Geometria GEOMETRY
);
```



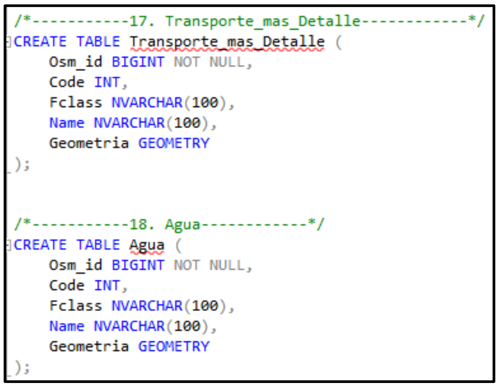
```
/*-----------6. Trafico mas Detalle------------*/
CREATE TABLE Trafico_mas_Detalle (
    Osm_id BIGINT PRIMARY KEY NOT NULL,
    Code INT,
    Fclass NVARCHAR(100),
    Name NVARCHAR(100),
    Geometria GEOMETRY
);


/*-----------7. Vias_Ferreas------------*/
CREATE TABLE Vias_Ferreas (
    Osm_id BIGINT PRIMARY KEY NOT NULL,
    Code INT,
    Fclass NVARCHAR(100),
    Name NVARCHAR(100),
    Layer INT,
    Bridge NVARCHAR(5),
    Tunnel NVARCHAR(5),
    Geometria GEOMETRY
);


/*-----------8. Carreteras_Calles------------*/
CREATE TABLE Carreteras_Calles (
    Osm_id BIGINT PRIMARY KEY NOT NULL,
    Code INT,
    Fclass NVARCHAR(100),
    Name NVARCHAR(100),
    Ref NVARCHAR(100),
    Oneway NVARCHAR(5),
    MaxSpeed INT,
    Layer INT,
    Bridge NVARCHAR(5),
    Tunnel NVARCHAR(5),
    Geometria GEOMETRY
);


/*-----------9. Vias_Fluviales------------*/
CREATE TABLE Vias_Fluviales (
    Osm_id BIGINT PRIMARY KEY NOT NULL,
    Code INT,
    Fclass NVARCHAR(100),
    Width INT,
    Name NVARCHAR(100),
    Geometria GEOMETRY
);
```



```
/*-----------10. Lugares------------*/
CREATE TABLE Lugares (
    Osm_id BIGINT PRIMARY KEY NOT NULL,
    Code INT,
    Fclass NVARCHAR(100),
    Population INT,
    Name NVARCHAR(100),
    Geometria GEOMETRY
);


/*-----------11. Lugares_mas_Detalle------------*/
CREATE TABLE Lugares_mas_Detalle (
    Osm_id BIGINT PRIMARY KEY NOT NULL,
    Code INT,
    Fclass NVARCHAR(100),
    Population INT,
    Name NVARCHAR(100),
    Geometria GEOMETRY
);


/*-----------12. Interes------------*/
CREATE TABLE Interes (
    Osm_id BIGINT NOT NULL,
    Code INT,
    Fclass NVARCHAR(100),
    Name NVARCHAR(100),
    Geometria GEOMETRY
);


/*-----------13. Interes_mas_Detalle------------*/
CREATE TABLE Interes_mas_Detalle (
    Osm_id BIGINT NOT NULL,
    Code INT,
    Fclass NVARCHAR(100),
    Name NVARCHAR(100),
    Geometria GEOMETRY
);


/*-----------14. Culto------------*/
CREATE TABLE Culto (
    Osm_id BIGINT PRIMARY KEY NOT NULL,
    Code INT,
    Fclass NVARCHAR(100),
    Name NVARCHAR(100),
    Geometria GEOMETRY
);
```

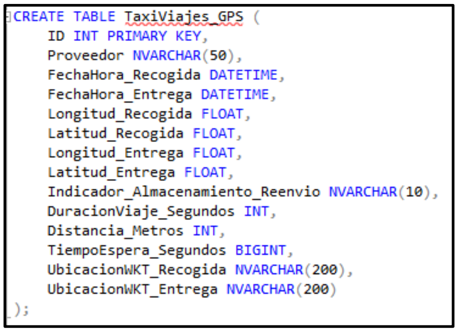


```
/*-----------15. Culto_mas_Detalle------------*/
CREATE TABLE Culto_mas_Detalle (
    Osm_id BIGINT PRIMARY KEY NOT NULL,
    Code INT,
    Fclass NVARCHAR(100),
    Name NVARCHAR(100),
    Geometria GEOMETRY
);


/*-----------16. Transporte------------*/
CREATE TABLE Transporte (
    Osm_id BIGINT NOT NULL,
    Code INT,
    Fclass NVARCHAR(100),
    Name NVARCHAR(100),
    Geometria GEOMETRY
);
```



```
/*-----------17. Transporte_mas_Detalle------------*/
CREATE TABLE Transporte_mas_Detalle (
    Osm_id BIGINT NOT NULL,
    Code INT,
    Fclass NVARCHAR(100),
    Name NVARCHAR(100),
    Geometria GEOMETRY
);


/*-----------18. Agua------------*/
CREATE TABLE Agua (
    Osm_id BIGINT NOT NULL,
    Code INT,
    Fclass NVARCHAR(100),
    Name NVARCHAR(100),
    Geometria GEOMETRY
);
```

b) For the GPS data I inserted it into a table adding two more columns, called **UbicacionWKT_Recogida** and **UbicacionWKT_Entrega** , these columns represent

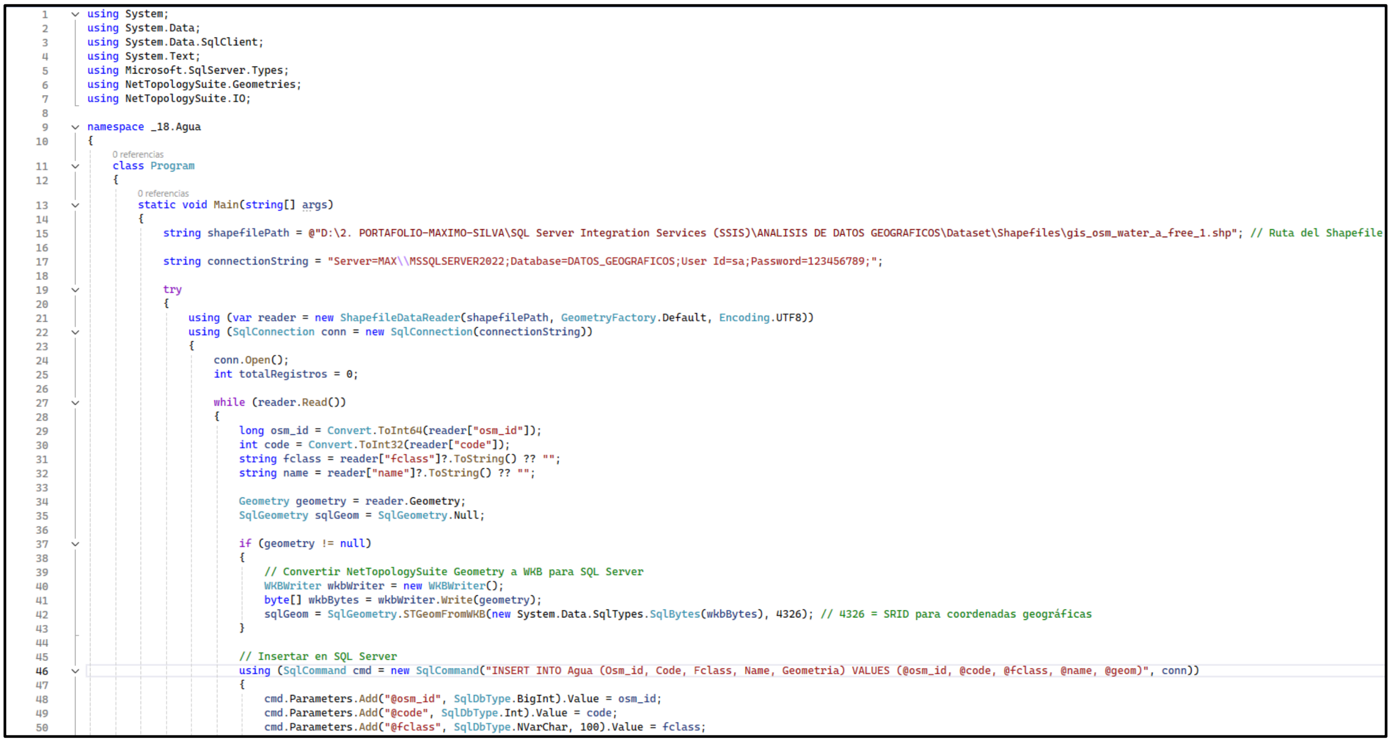


```sql
CREATE TABLE TaxiViajes_GPS (
    ID INT PRIMARY KEY,
    Proveedor NVARCHAR(50),
    FechaHora_Recogida DATETIME,
    FechaHora_Entrega DATETIME,
    Longitud_Recogida FLOAT,
    Latitud_Recogida FLOAT,
    Longitud_Entrega FLOAT,
    Latitud_Entrega FLOAT,
    Indicador_Almacenamiento_Reenvio NVARCHAR(10),
    DuracionViaje_Segundos INT,
    Distancia_Metros INT,
    TiempoEspera_Segundos BIGINT,
    UbicacionWKT_Recogida NVARCHAR(200),
    UbicacionWKT_Entrega NVARCHAR(200)
);
```

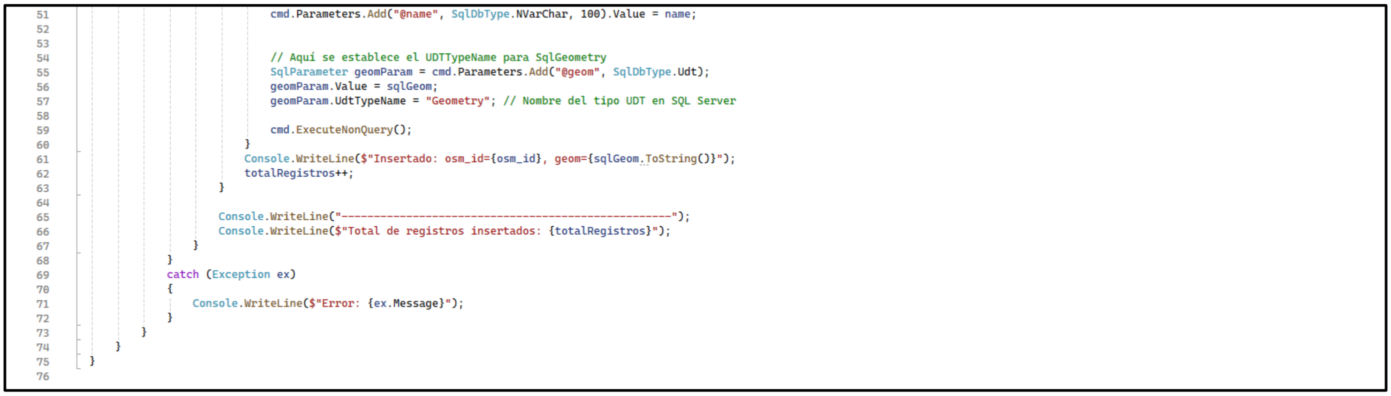
Shapefiles Data into the Database with SSIS.

For this step, I will create a project in SSIS and since there is no native component to insert Shapefiles (. shp) directly into SQL Server, I will use **code in C# (Console Application (.NET Framework) C#)** , where I will use libraries for the correct insertion of data into the DB in SQL Server.

For example, for the Shapefile Water, the C# code would be:

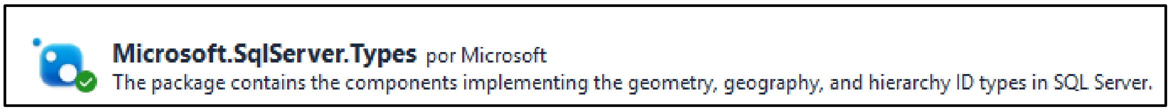


```csharp
using System;
using System.Data;
using System.Data.SqlClient;
using System.Text;
using Microsoft.SqlServer.Types;
using NetTopologySuite.Geometries;
using NetTopologySuite.IO;

namespace _18.Agua
{
    0 referencias
    class Program
    {
        0 referencias
        static void Main(string[] args)
        {
            string shapefilePath = @"D:\2. PORTAFOLIO-MAXIMO-SILVA\SQL Server Integration Services (SSIS)\ANALISIS DE DATOS GEOGRAFICOS\Dataset\Shapefiles\gis_osm_water_a_free_1.shp"; // Ruta del Shapefile

            string connectionString = "Server=MAX\\MSSQLSERVER2022;Database=DATOS_GEOGRAFICOS;User Id=sa;Password=123456789;";

            try
            {
                using (var reader = new ShapefileDataReader(shapefilePath, GeometryFactory.Default, Encoding.UTF8))
                using (SqlConnection conn = new SqlConnection(connectionString))
                {
                    conn.Open();
                    int totalRegistros = 0;

                    while (reader.Read())
                    {
                        long osm_id = Convert.ToInt64(reader["osm_id"]);
                        int code = Convert.ToInt32(reader["code"]);
                        string fclass = reader["fclass"]?.ToString() ?? "";
                        string name = reader["name"]?.ToString() ?? "";

                        Geometry geometry = reader.Geometry;
                        SqlGeometry sqlGeom = SqlGeometry.Null;

                        if (geometry != null)
                        {
                            // Convertir NetTopologySuite Geometry a WKB para SQL Server
                            WKBWriter wkbWriter = new WKBWriter();
                            byte[] wkbBytes = wkbWriter.Write(geometry);
                            sqlGeom = SqlGeometry.STGeomFromWKB(new System.Data.SqlTypes.SqlBytes(wkbBytes), 4326); // 4326 = SRID para coordenadas geográficas
                        }

                        // Insertar en SQL Server
                        using (SqlCommand cmd = new SqlCommand("INSERT INTO Agua (Osm_id, Code, Fclass, Name, Geometria) VALUES (@osm_id, @code, @fclass, @name, @geom)", conn))
                        {
                            cmd.Parameters.Add("@osm_id", SqlDbType.BigInt).Value = osm_id;
                            cmd.Parameters.Add("@code", SqlDbType.Int).Value = code;
                            cmd.Parameters.Add("@fclass", SqlDbType.NVarChar, 100).Value = fclass;
```

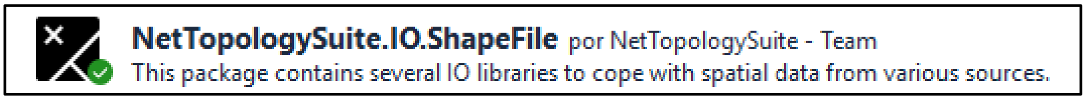


```csharp
                            cmd.Parameters.Add("@name", SqlDbType.NVarChar, 100).Value = name;


                            // Aquí se establece el UDTTypeName para SqlGeometry
                            SqlParameter geomParam = cmd.Parameters.Add("@geom", SqlDbType.Udt);
                            geomParam.Value = sqlGeom;
                            geomParam.UdtTypeName = "Geometry"; // Nombre del tipo UDT en SQL Server

                            cmd.ExecuteNonQuery();
                        }
                        Console.WriteLine($"Insertado: osm_id={osm_id}, geom={sqlGeom.ToString()}");
                        totalRegistros++;
                    }

                    Console.WriteLine("--------------------------------------------------");
                    Console.WriteLine($"Total de registros insertados: {totalRegistros}");
                }
            }
            catch (Exception ex)
            {
                Console.WriteLine($"Error: {ex.Message}");
            }
        }
    }
}
```

For the code to work, two packages need to be installed within the **NuGet Package Manager :**



Microsoft.SqlServer.Types por Microsoft
The package contains the components implementing the geometry, geography, and hierarchy ID types in SQL Server.

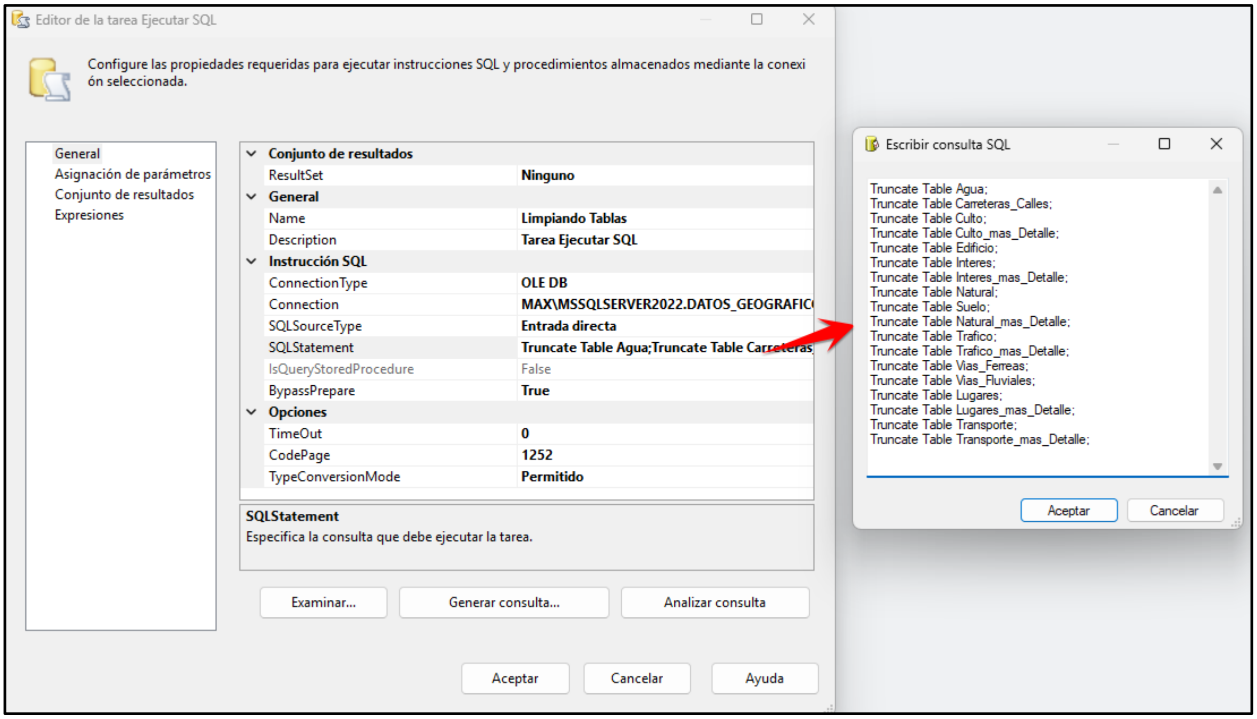


NetTopologySuite.IO.ShapeFile por NetTopologySuite - Team
This package contains several IO libraries to cope with spatial data from various sources.

Once everything was working correctly, I compiled the project for later use.

The same process was performed for all shapefiles .

I then dragged the SQL Task component to perform the truncation of all tables first.



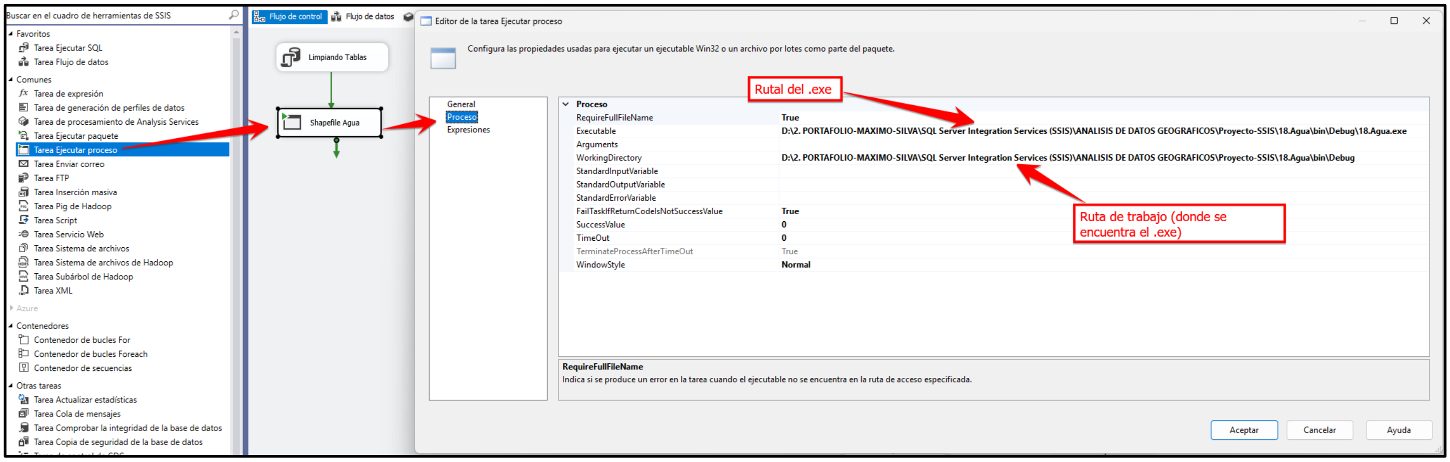
Configure las propiedades requeridas para ejecutar instrucciones SQL y procedimientos almacenados mediante la conexión seleccionada.

Conjunto de resultados	
ResultSet	**Ninguno**
General	
Name	**Limpiando Tablas**
Description	**Tarea Ejecutar SQL**
Instrucción SQL	
ConnectionType	**OLE DB**
Connection	**MAX\MSSQLSERVER2022.DATOS_GEOGRAFIC**
SQLSourceType	**Entrada directa**
SQLStatement	**Truncate Table Agua;Truncate Table Carreteras**
IsQueryStoredProcedure	False
BypassPrepare	**True**
Opciones	
TimeOut	**0**
CodePage	**1252**
TypeConversionMode	**Permitido**

SQLStatement
Especifica la consulta que debe ejecutar la tarea.

```
Truncate Table Agua;
Truncate Table Carreteras_Calles;
Truncate Table Culto;
Truncate Table Culto_mas_Detalle;
Truncate Table Edificio;
Truncate Table Interes;
Truncate Table Interes_mas_Detalle;
Truncate Table Natural;
Truncate Table Suelo;
Truncate Table Natural_mas_Detalle;
Truncate Table Trafico;
Truncate Table Trafico_mas_Detalle;
Truncate Table Vias_Ferreas;
Truncate Table Vias_Fluviales;
Truncate Table Lugares;
Truncate Table Lugares_mas_Detalle;
Truncate Table Transporte;
Truncate Table Transporte_mas_Detalle;
```

I then used the Execute Package task where I set the path of the executable that generated the build:

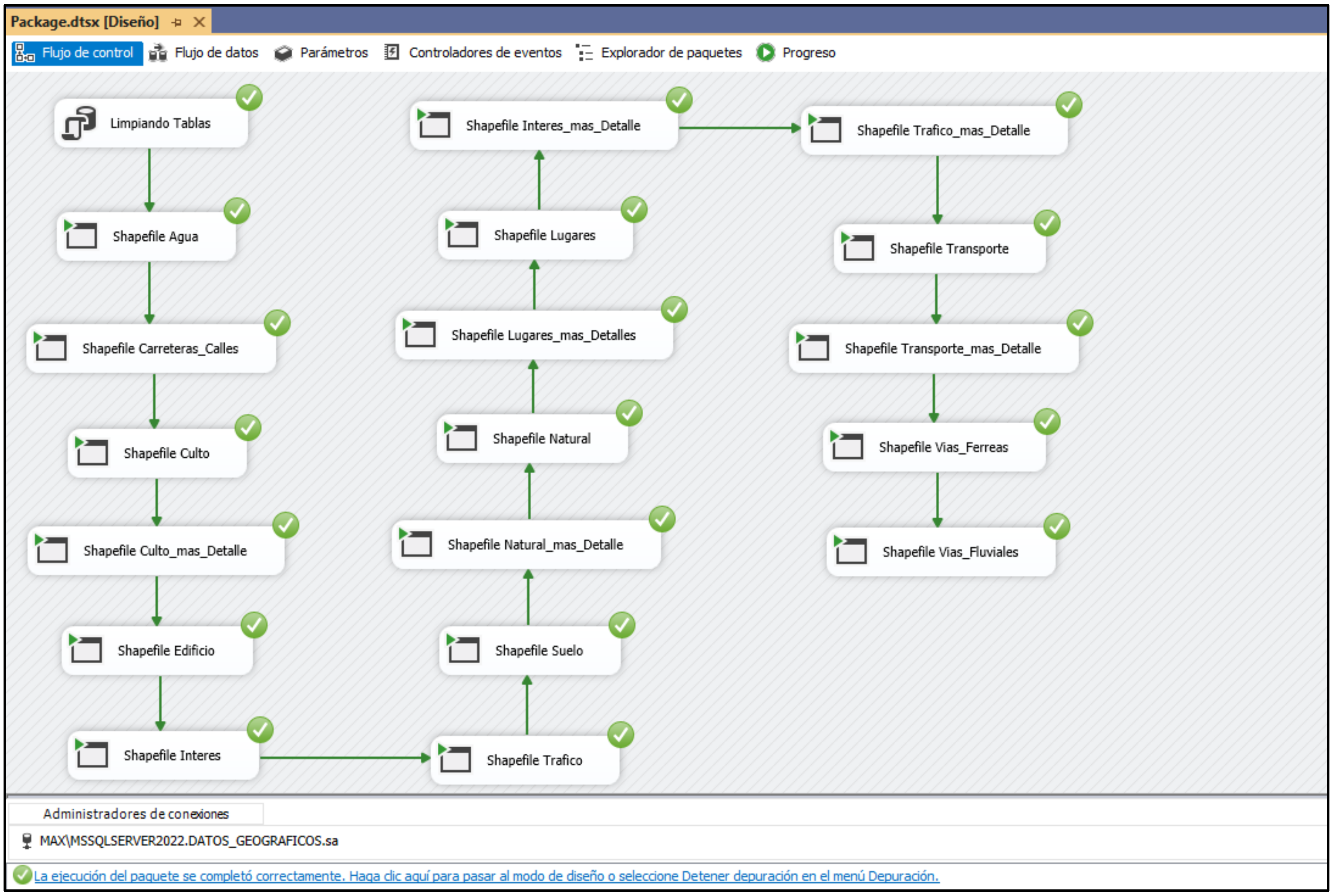


Configura las propiedades usadas para ejecutar un ejecutable Win32 o un archivo por lotes como parte del paquete.

Proceso	
RequireFullFileName	**True**
Executable	**D:\2. PORTAFOLIO-MAXIMO-SILVA\SQL Server Integration Services (SSIS)\ANALISIS DE DATOS GEOGRAFICOS\Proyecto-SSIS\18.Agua\bin\Debug\18.Agua.exe**
Arguments	
WorkingDirectory	**D:\2. PORTAFOLIO-MAXIMO-SILVA\SQL Server Integration Services (SSIS)\ANALISIS DE DATOS GEOGRAFICOS\Proyecto-SSIS\18.Agua\bin\Debug**
StandardInputVariable	
StandardOutputVariable	
StandardErrorVariable	
FailTaskIfReturnCodeIsNotSuccessValue	**True**
SuccessValue	**0**
TimeOut	**0**
TerminateProcessAfterTimeOut	True
WindowStyle	**Normal**

RequireFullFileName
Indica si se produce un error en la tarea cuando el ejecutable no se encuentra en la ruta de acceso especificada.

I did the same for all the files in each Shapefile and ran the project:

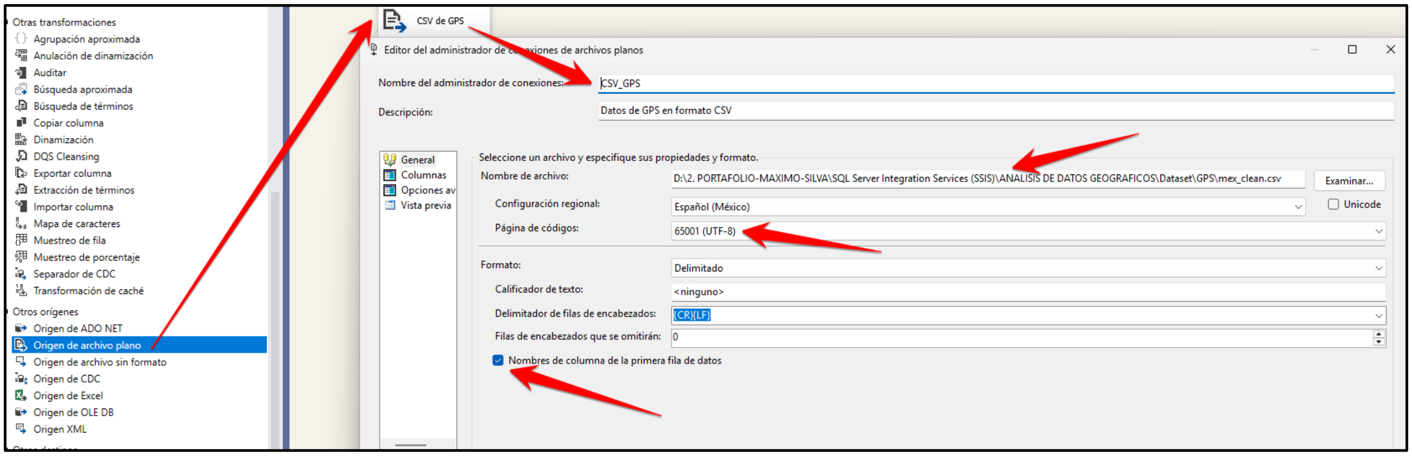


MAX\MSSQLSERVER2022.DATOS_GEOGRAFICOS.sa

La ejecución del paquete se completó correctamente. Haga clic aquí para pasar al modo de diseño o seleccione Detener depuración en el menú Depuración.

Step 5: Inserting GPS Data into the Database with SSIS.

I dragged in a flat file source and set it up with the .CSV containing the GPS data.



Editor del administrador de conexiones de archivos planos

Nombre del administrador de conexiones: CSV_GPS

Descripción: Datos de GPS en formato CSV

Seleccione un archivo y especifique sus propiedades y formato.

Nombre de archivo: D:\2. PORTAFOLIO-MAXIMO-SILVA\SQL Server Integration Services (SSIS)\ANALISIS DE DATOS GEOGRAFICOS\Dataset\GPS\mex_clean.csv

Configuración regional: Español (México) ☐ Unicode

Página de códigos: 65001 (UTF-8)

Formato: Delimitado

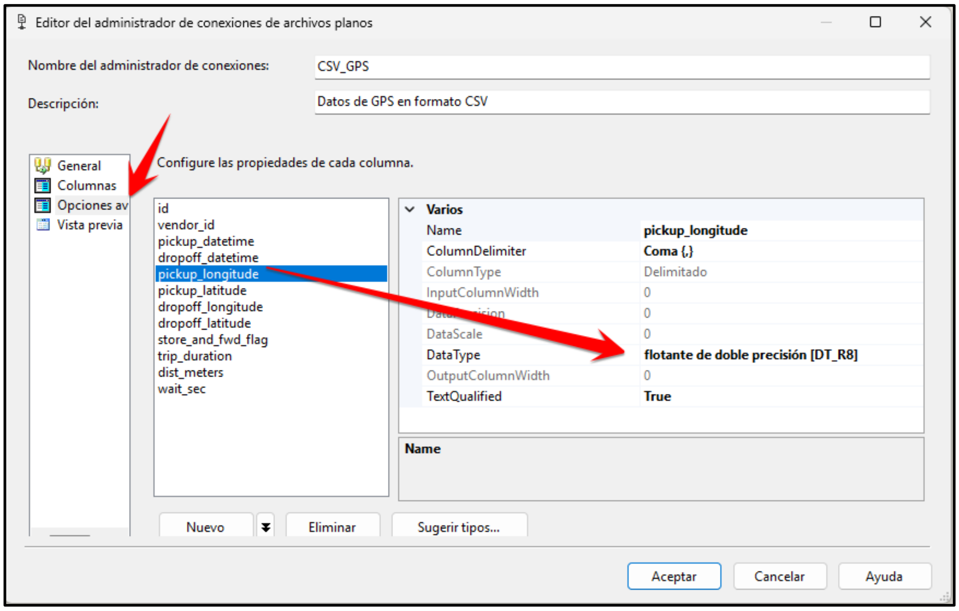
Calificador de texto: <ninguno>

Delimitador de filas de encabezados: {CR}{LF}

Filas de encabezados que se omitirán: 0

☑ Nombres de columna de la primera fila de datos

The corresponding type is placed in each column in SSIS taking into account its data type in SQL Server. For example, **pickup_longitude** is of type **DT_R8** in SSIS so that it can be saved correctly in SQL Server where the type is FLOAT.



Editor del administrador de conexiones de archivos planos

Nombre del administrador de conexiones: CSV_GPS

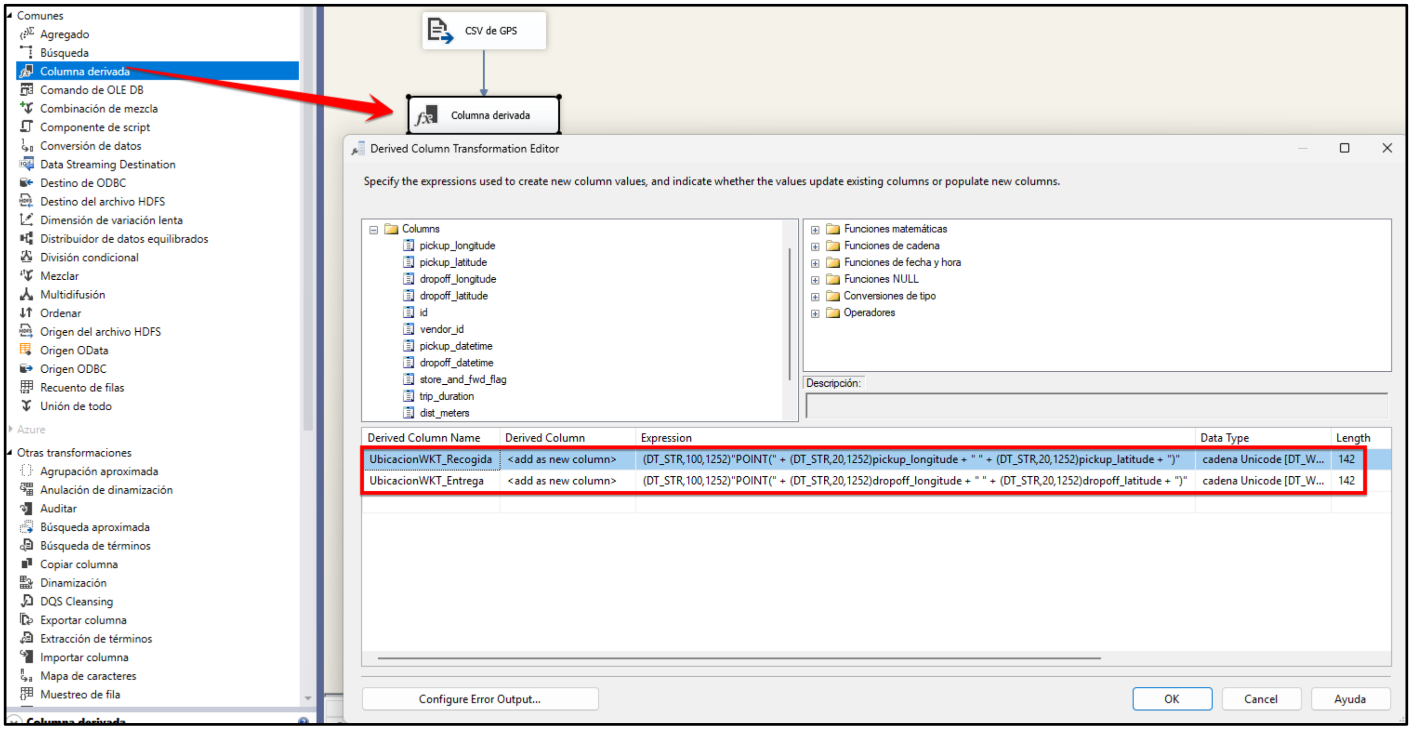
Descripción: Datos de GPS en formato CSV

Configure las propiedades de cada columna.

Varios	
Name	pickup_longitude
ColumnDelimiter	Coma {,}
ColumnType	Delimitado
InputColumnWidth	0
DataPrecision	0
DataScale	0
DataType	flotante de doble precisión [DT_R8]
OutputColumnWidth	0
TextQualified	True

Name

Then I used the Derived Column component and created two columns that will store the WKT format in a text string so that it can later be converted to a GEOGRAPHY data type.

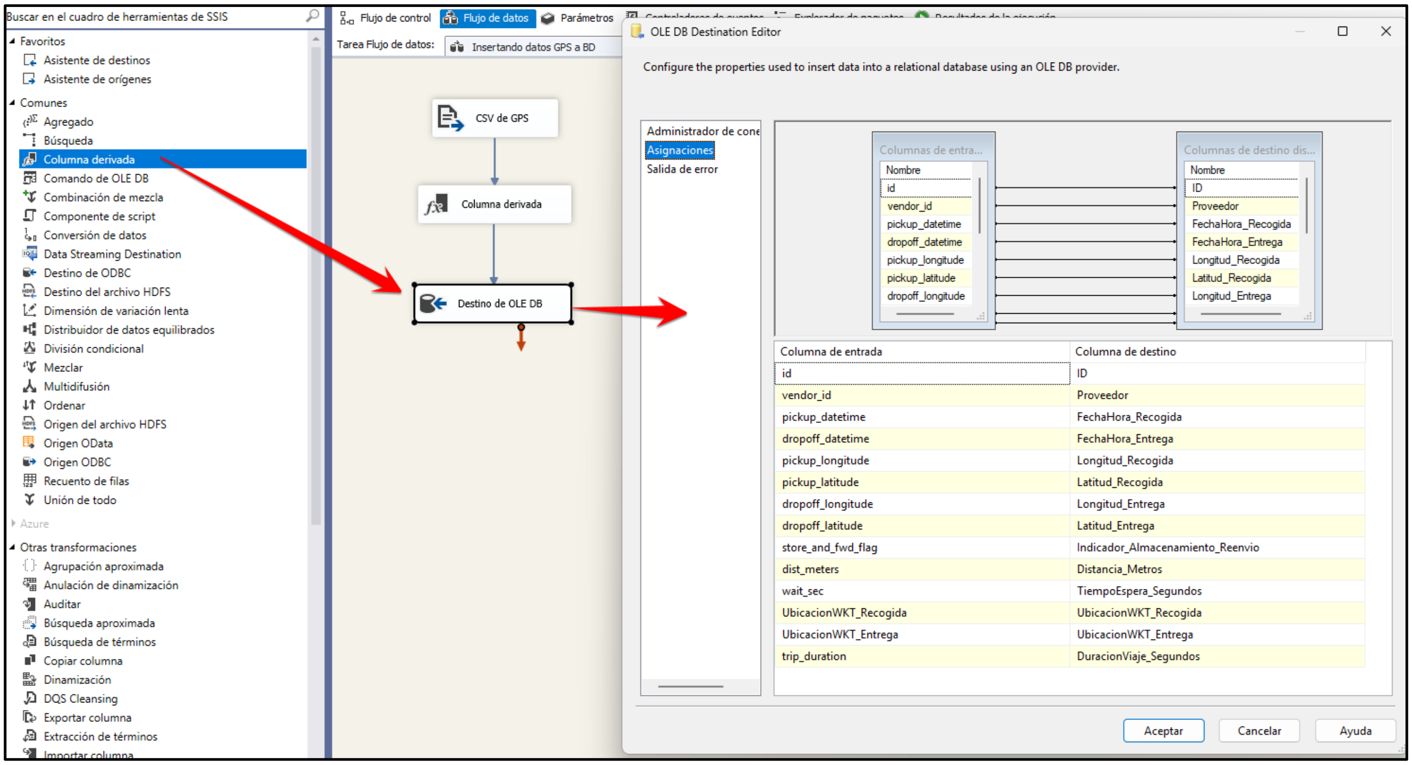


Derived Column Transformation Editor

Specify the expressions used to create new column values, and indicate whether the values update existing columns or populate new columns.

Derived Column Name	Derived Column	Expression	Data Type	Length
UbicacionWKT_Recogida	<add as new column>	(DT_STR,100,1252)"POINT(" + (DT_STR,20,1252)pickup_longitude + " " + (DT_STR,20,1252)pickup_latitude + ")"	cadena Unicode [DT_W...	142
UbicacionWKT_Entrega	<add as new column>	(DT_STR,100,1252)"POINT(" + (DT_STR,20,1252)dropoff_longitude + " " + (DT_STR,20,1252)dropoff_latitude + ")"	cadena Unicode [DT_W...	142

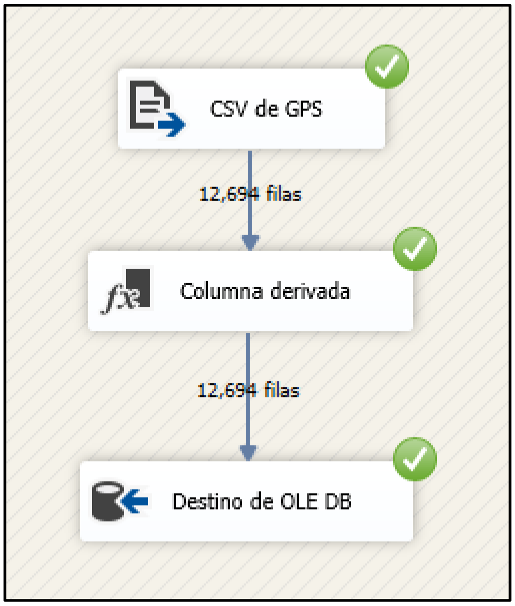
Finally, I set up a destination to a table in SQL Server and run the project.



OLE DB Destination Editor

Configure the properties used to insert data into a relational database using an OLE DB provider.

Columna de entrada	Columna de destino
id	ID
vendor_id	Proveedor
pickup_datetime	FechaHora_Recogida
dropoff_datetime	FechaHora_Entrega
pickup_longitude	Longitud_Recogida
pickup_latitude	Latitud_Recogida
dropoff_longitude	Longitud_Entrega
dropoff_latitude	Latitud_Entrega
store_and_fwd_flag	Indicador_Almacenamiento_Reenvio
dist_meters	Distancia_Metros
wait_sec	TiempoEspera_Segundos
UbicacionWKT_Recogida	UbicacionWKT_Recogida
UbicacionWKT_Entrega	UbicacionWKT_Entrega
trip_duration	DuracionViaje_Segundos



Step 6: Identifying Data to be used for analysis.

Since I won't use all the data entered in the first database, and some data is inconsistent, I will perform an ETL process to another database with only the columns I will use and using clean data. To do this, I created a database called "CLEAN_GEOGRAPHIC_DATA", where I will store all the tables that will be used for various analyses.

You want to export data that answers the following questions:

a) How many taxis pick up passengers near points of interest (hotels, restaurants, transport stations, parks, etc.)?

b) What percentage of trips begin or end near a transport station?

c) What are the areas with the most taxi trips at a specific time of day?

d) Which point of interest contains the largest number of registered trips between Hotels and Restaurants?

e) What are the most used roads to start a trip?

f) What are the places of interest with the most taxi trips?

g) Which cities have the highest taxi activity?

h) List of trips that start near a natural area.

i) What is the average speed of trips on different types of roads in km/h?

j) How many trips end in hospitals?

source tables of the Shapefiles with useful data that will be used for this analysis are:

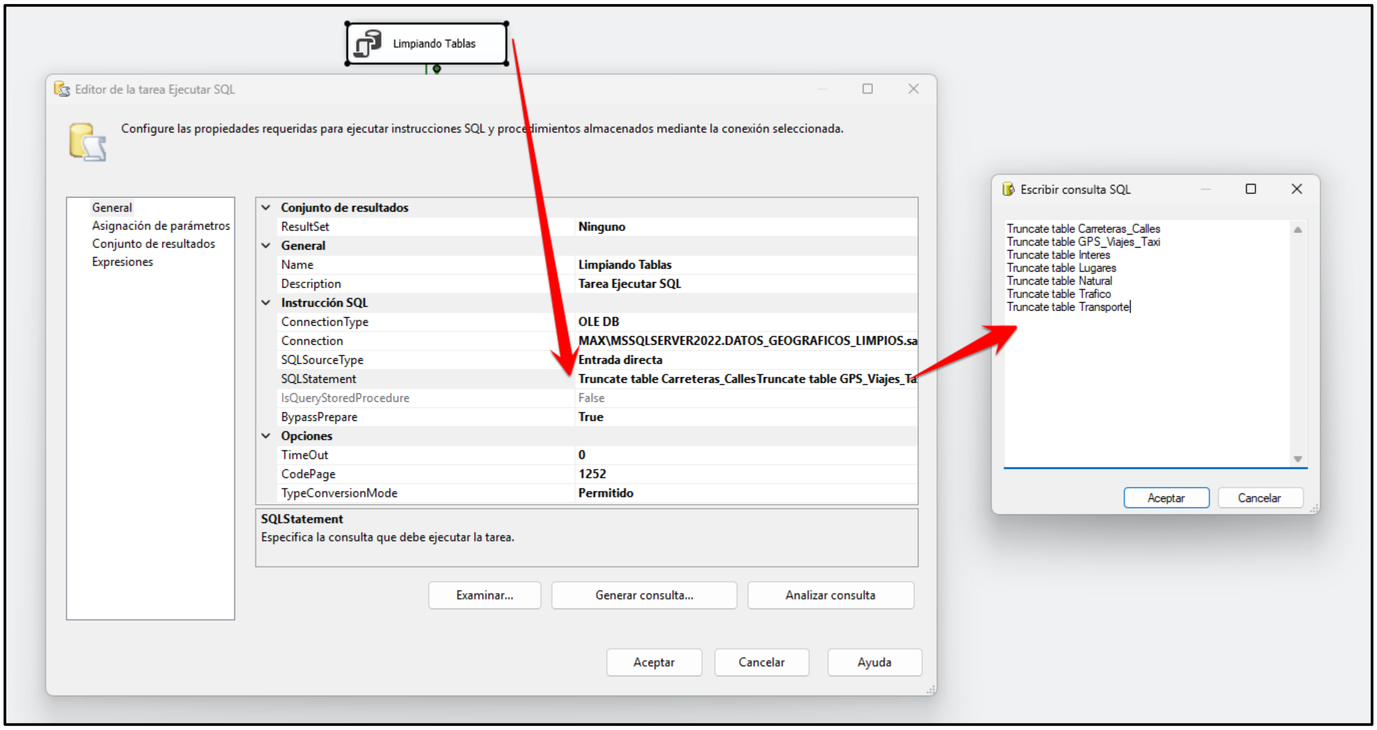
TABLES IN DB GEOGRAPHIC_DATA	DESCRIPTION AND USE	TABLES IN CLEAN_GEOGRAPHIC_DATA_DB
Carretera_Calle	Key to analyzing road infrastructure and routes	**Carretera_Calle**
Interes	Data on hotels, restaurants, train stations, etc.	**Interes**
Lugares	It can help to see key start and end points of the trip.	**Lugares**
Lugares_mas_Detalle	More data from Key Places.	**Lugares**
Agua	Data on lakes, rivers, etc.	**Natural**
Natural	Data relating to natural areas.	**Natural**
Natural_mas_Detalle	More Natural Area Data.	**Natural**
Trafico	Useful for analyzing congestion and travel time	**Trafico**
Trafico_mas_Detalle	More data regarding traffic.	**Trafico**
Transporte	Bus stops, train station, airport.	**Transporte**
Transporte_mas_Detalle	More data on Transport.	**Transporte**

And the columns of GPS data that will be used are:

COLUMNS OF THE TABLE TaxiViajes_GPS BD DATOS_GEOGRAFICOS	DESCRIPTION	GPS_ Viajes_Taxi TABLE COLUMNS BD DATOS_GEOGRAFIOS_LIMPIOS
ID	Code	**Id_Viajes_Taxi**
Proveedor	Taxi Provider	**Proveedor**
FechaHora_Entrega	Date and Time of Start of the Trip	**FechaHora_Inicio**
FechaHora_Recogida	Trip End Date and Time	**FechaHora_Fin**
DuracionViaje_Segundos	Duration of the trip in seconds	**DuracionViaje_Segundos**
Distancia_Metros	Distance in meters of the trip	**Distancia_Metros**
TiempoEspera_Segundos	Wait time in seconds	**TiempoEspera_Segundos**
UbicacionWKT_Recogida (NVARCHAR)	Trip Start Location	**Ubicacion_Inicio (GEOGRAPHY)**
UbicacionWKT_Entrega (NVARCHAR)	Location of the trip destination	**Ubicacion_Fin (GEOGRAPHY)**

Step 7: Transferring data from the Shapefiles to a new DB with only filtered data.

First, I performed a truncation of the tables using the Execute SQL Task component:



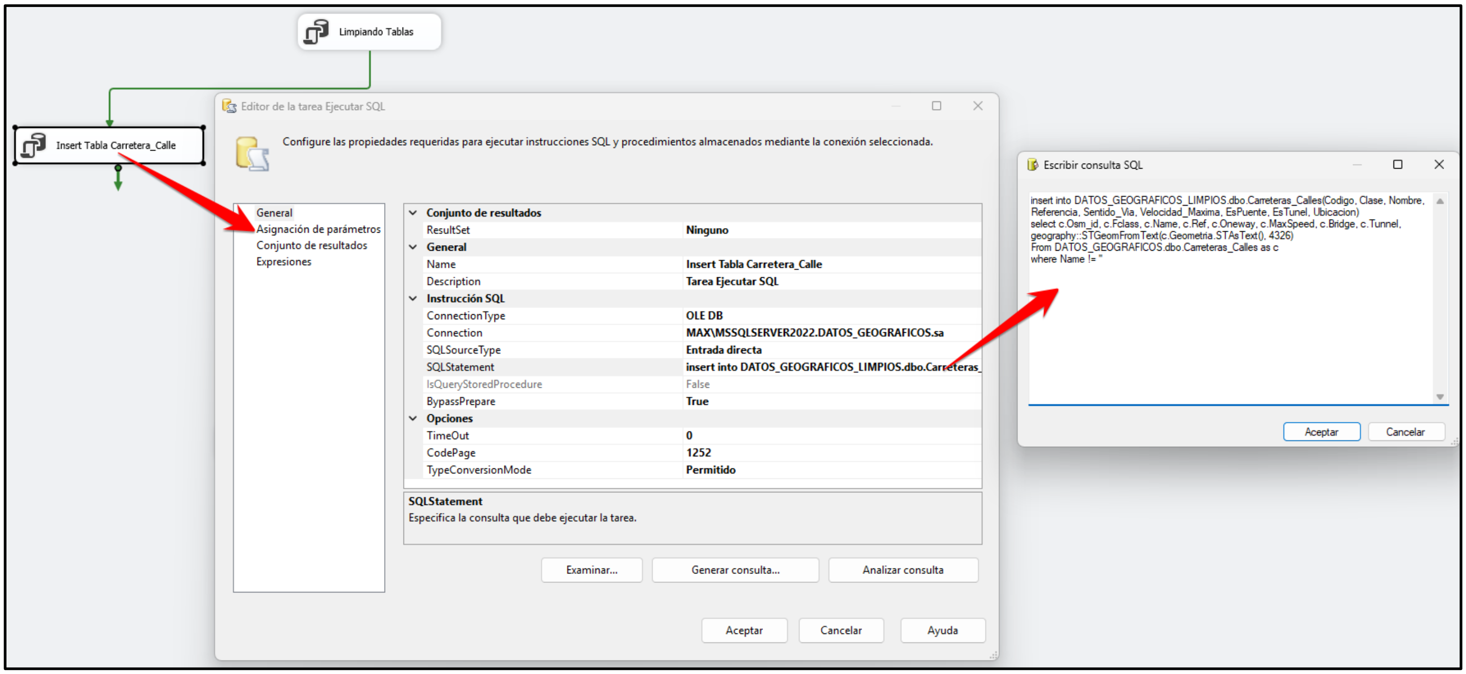
Configure las propiedades requeridas para ejecutar instrucciones SQL y procedimientos almacenados mediante la conexión seleccionada.

Conjunto de resultados	
ResultSet	**Ninguno**
General	
Name	**Limpiando Tablas**
Description	**Tarea Ejecutar SQL**
Instrucción SQL	
ConnectionType	**OLE DB**
Connection	**MAX\MSSQLSERVER2022.DATOS_GEOGRAFICOS_LIMPIOS.sa**
SQLSourceType	**Entrada directa**
SQLStatement	**Truncate table Carreteras_CallesTruncate table GPS_Viajes_Ta**
IsQueryStoredProcedure	False
BypassPrepare	**True**
Opciones	
TimeOut	**0**
CodePage	**1252**
TypeConversionMode	**Permitido**

SQLStatement
Especifica la consulta que debe ejecutar la tarea.

```
Truncate table Carreteras_Calles
Truncate table GPS_Viajes_Taxi
Truncate table Interes
Truncate table Lugares
Truncate table Natural
Truncate table Trafico
Truncate table Transporte
```

Because the Geometry data type is handled in a complex way in SSIS, I chose to perform each data insertion using the Execute SQL Task component as well, **converting the GEOMETRY data to GEOGRAPHY** and making filters, such as passing only data that contains data in the Name field, as in the following image:



Configure las propiedades requeridas para ejecutar instrucciones SQL y procedimientos almacenados mediante la conexión seleccionada.

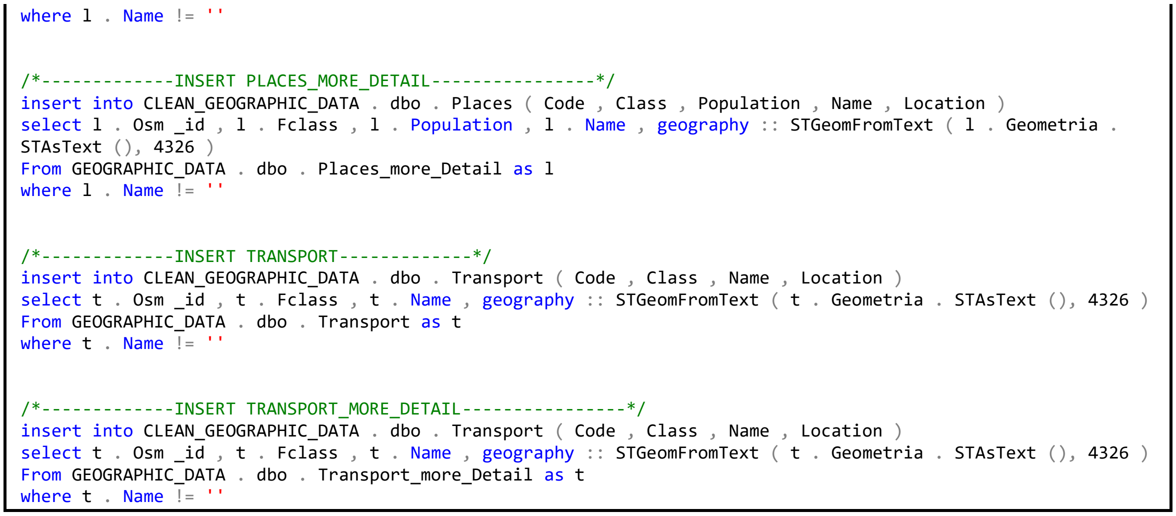
Conjunto de resultados	
ResultSet	Ninguno
General	
Name	Insert Tabla Carretera_Calle
Description	Tarea Ejecutar SQL
Instrucción SQL	
ConnectionType	OLE DB
Connection	MAX\MSSQLSERVER2022.DATOS_GEOGRAFICOS.sa
SQLSourceType	Entrada directa
SQLStatement	insert into DATOS_GEOGRAFICOS_LIMPIOS.dbo.Carreteras_
IsQueryStoredProcedure	False
BypassPrepare	True
Opciones	
TimeOut	0
CodePage	1252
TypeConversionMode	Permitido

SQLStatement
Especifica la consulta que debe ejecutar la tarea.

```
insert into DATOS_GEOGRAFICOS_LIMPIOS.dbo.Carreteras_Calles(Codigo, Clase, Nombre,
Referencia, Sentido_Via, Velocidad_Maxima, EsPuente, EsTunel, Ubicacion)
select c.Osm_id, c.Fclass, c.Name, c.Ref, c.Oneway, c.MaxSpeed, c.Bridge, c.Tunnel,
geography::STGeomFromText(c.Geometria.STAsText(), 4326)
From DATOS_GEOGRAFICOS.dbo.Carreteras_Calles as c
where Name != ''
```

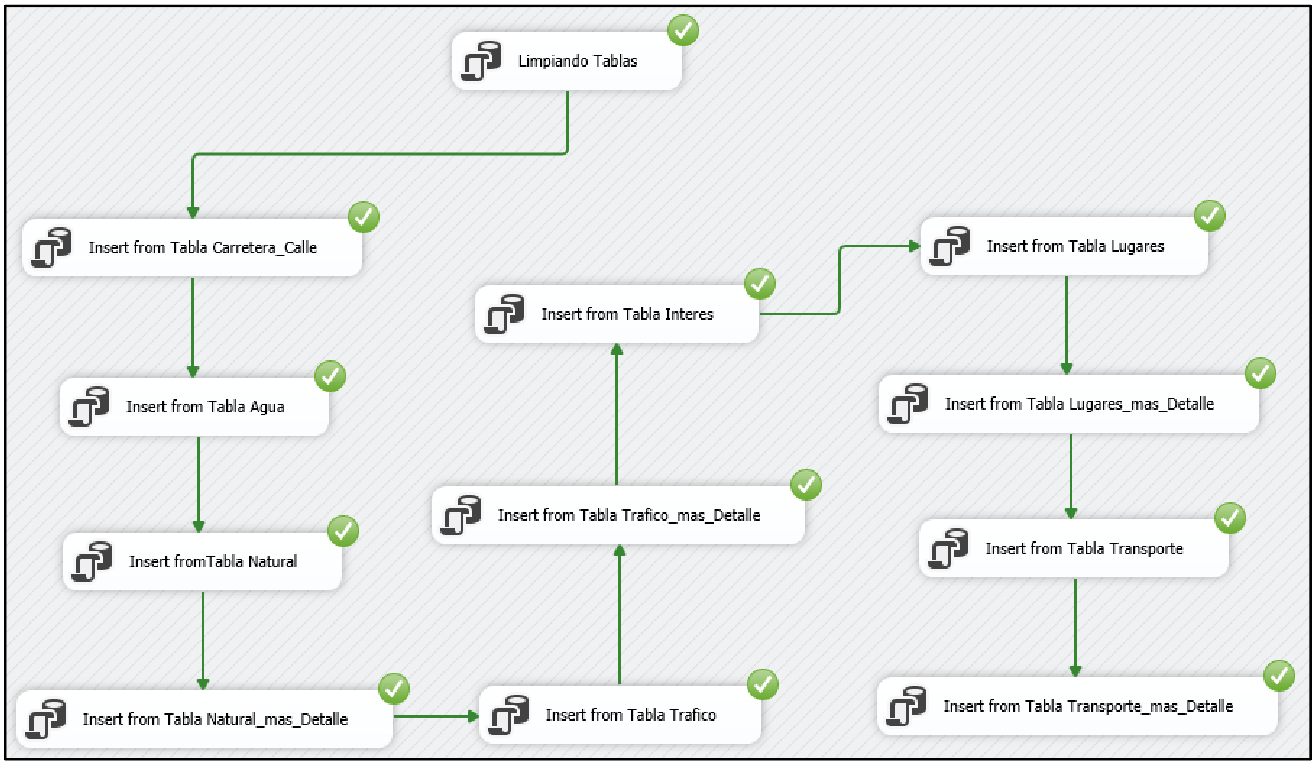
The same process was performed for all tables, using the following queries:

```
/*------------INSERT ROAD_STREET---------------*/
insert into CLEAN_GEOGRAPHIC_DATA . dbo . Roads_Streets ( Code , Class , Name , Reference , Direction_Via ,
Maximum_Speed , It is a bridge , EsTunel , Location )
select c . Osm _id , c . Fclass , c . Name , c . Ref , c . Oneway , c . MaxSpeed , c . Bridge , c . Tunnel ,
geography :: STGeomFromText ( c . Geometria . STAsText (), 4326 )
From GEOGRAPHIC_DATA . dbo . Roads_Streets as c
where Yam ! = ''



/*------------INSERT WATER---------------*/
insert into CLEAN_GEOGRAPHIC_DATA . dbo . Natural ( Code , Class , Name , Location )
select a . Osm _id , a . Fclass , a . Name , geography :: STGeomFromText ( a . Geometria . STAsText (), 4326 )
From GEOGRAPHIC DATA . dbo . Water as a
where a . Name != ''


/*------------INSERT NATURAL------------*/
insert into CLEAN_GEOGRAPHIC_DATA . dbo . Natural ( Code , Class , Name , Location )
select l . Osm _id , l . Fclass , l . Name , geography :: STGeomFromText ( l . Geometria . STAsText (), 4326 )
From GEOGRAPHIC_DATA . dbo . Natural as l
where l . Name != ''

/*------------INSERT NATURAL_MORE_DETAIL------------*/
insert into CLEAN_GEOGRAPHIC_DATA . dbo . Natural ( Code , Class , Name , Location )
select n . Osm _id , n . Fclass , n . Name , geography :: STGeomFromText ( n . Geometria . STAsText (), 4326 )
From GEOGRAPHIC_DATA . dbo . Natural_more_Detail as n
where n . Name != ''


/*------------INSERT INTEREST------------*/
insert into CLEAN_GEOGRAPHIC_DATA . dbo . Interest ( Code , Class , Name , Location )
select i . Osm _id , i . Fclass , i . Name , geography :: STGeomFromText ( i . Geometria . STAsText (), 4326 )
From GEOGRAPHIC_DATA . dbo . Interest So
where i . Name != ''

/*------------INSERT TRAFFIC---------------*/
insert into CLEAN_GEOGRAPHIC_DATA . dbo . Traffic ( Code , Class , Name , Location )
select t . Osm _id , t . Fclass , t . Name , geography :: STGeomFromText ( t . Geometria . STAsText (), 4326 )
From GEOGRAPHIC_DATA . dbo . Traffic as t


/*------------INSERT TRAFFIC_MORE_DETAIL---------------*/
insert into CLEAN_GEOGRAPHIC_DATA . dbo . Traffic ( Code , Class , Name , Location )
select t . Osm _id , t . Fclass , t . Name , geography :: STGeomFromText ( t . Geometria . STAsText (), 4326 )
From GEOGRAPHIC_DATA . dbo . Traffic_more_Details as t


/*------------INSERT PLACES------------*/
insert into CLEAN_GEOGRAPHIC_DATA . dbo . Places ( Code , Class , Population , Name , Location )
select l . Osm _id , l . Fclass , l . Population , l . Name , geography :: STGeomFromText ( l . Geometria .
STAsText (), 4326 )
From GEOGRAPHIC_DATA . dbo . Places as l
```



```
where l . Name != ''


/*-------------INSERT PLACES_MORE_DETAIL----------------*/
insert into CLEAN_GEOGRAPHIC_DATA . dbo . Places ( Code , Class , Population , Name , Location )
select l . Osm _id , l . Fclass , l . Population , l . Name , geography :: STGeomFromText ( l . Geometria .
STAsText (), 4326 )
From GEOGRAPHIC_DATA . dbo . Places_more_Detail as l
where l . Name != ''


/*-------------INSERT TRANSPORT-------------*/
insert into CLEAN_GEOGRAPHIC_DATA . dbo . Transport ( Code , Class , Name , Location )
select t . Osm _id , t . Fclass , t . Name , geography :: STGeomFromText ( t . Geometria . STAsText (), 4326 )
From GEOGRAPHIC_DATA . dbo . Transport as t
where t . Name != ''


/*-------------INSERT TRANSPORT_MORE_DETAIL----------------*/
insert into CLEAN_GEOGRAPHIC_DATA . dbo . Transport ( Code , Class , Name , Location )
select t . Osm _id , t . Fclass , t . Name , geography :: STGeomFromText ( t . Geometria . STAsText (), 4326 )
From GEOGRAPHIC_DATA . dbo . Transport_more_Detail as t
where t . Name != ''
```

The project is executed:

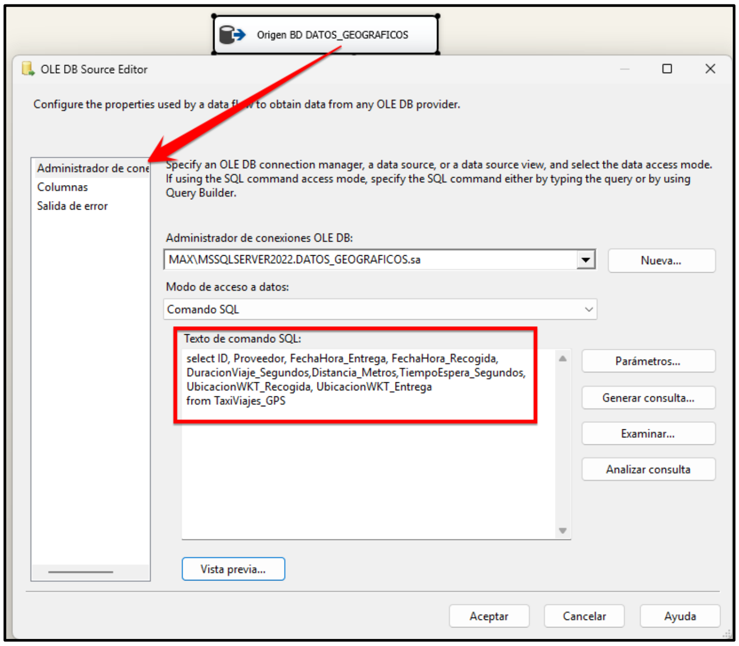

Limpiando Tablas
Insert from Tabla Carretera_Calle
Insert from Tabla Agua
Insert fromTabla Natural
Insert from Tabla Natural_mas_Detalle
Insert from Tabla Trafico
Insert from Tabla Trafico_mas_Detalle
Insert from Tabla Interes
Insert from Tabla Lugares
Insert from Tabla Lugares_mas_Detalle
Insert from Tabla Transporte
Insert from Tabla Transporte_mas_Detalle

Step 8: Moving GPS data to a new DB with only filtered data.

To pass the relevant data and making the corresponding filters, I extracted the fields mentioned above, for this I carried out the following process:

a) Make the query to extract only the columns that I require from the BD DATOS_GEOGRAFICOS, including the Start and End Location data to NVARCHAR fields in the destination database, in the columns: Ubicacion_Inicio_Temporal and Ubicacion_Fin_Temporal .

```
select ID , Supplier , DateTime_Delivery , Pickup_Date_Time ,
TripDuration_Seconds , Distance_Meters , WaitTime_Seconds ,
geography :: STGeomFromText ( LocationWKT_Pickup , 4326 ) ACE Pickup Location
,
geography :: STGeomFromText ( LocationWKT_Delivery , 4326 ) ACE Delivery
Location
from GEOGRAPHIC_DATA . dbo . TaxiViajes_GPS
```



Configure the properties used by a data flow to obtain data from any OLE DB provider.

Specify an OLE DB connection manager, a data source, or a data source view, and select the data access mode. If using the SQL command access mode, specify the SQL command either by typing the query or by using Query Builder.

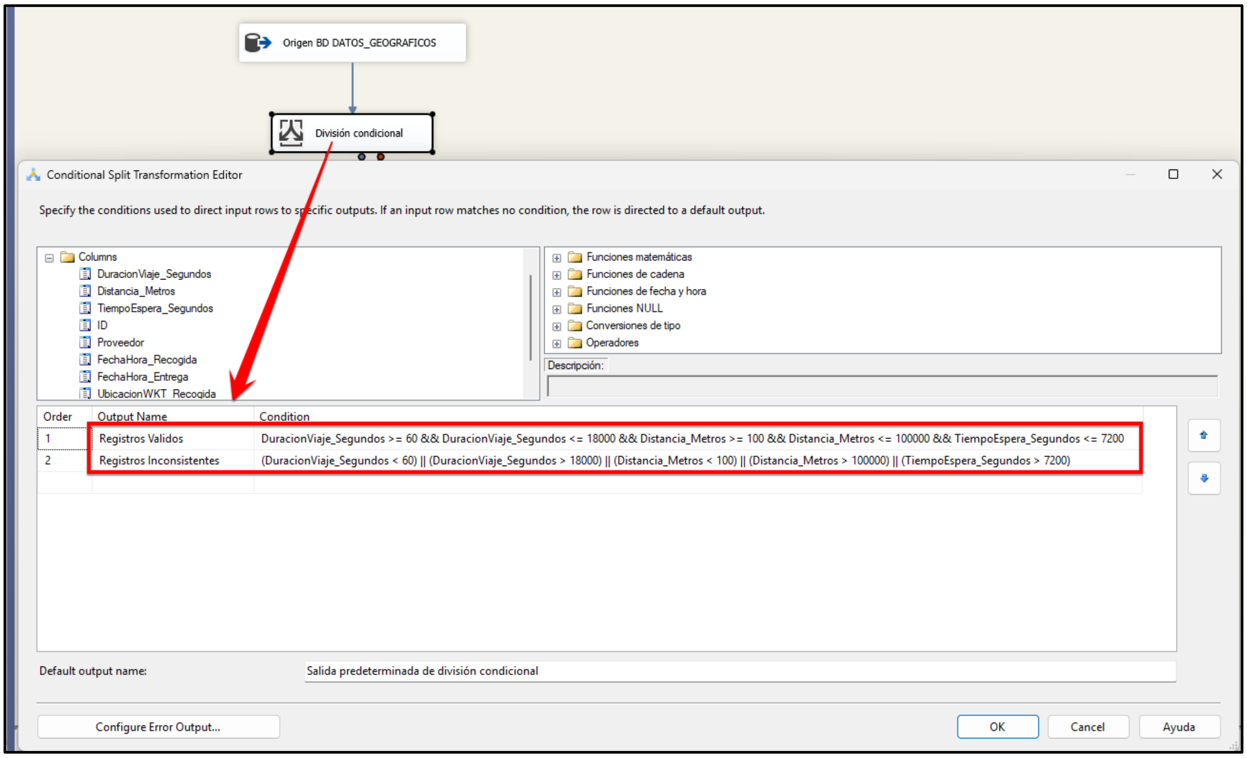
Administrador de conexiones OLE DB: MAX\MSSQLSERVER2022.DATOS_GEOGRAFICOS.sa

Modo de acceso a datos: Comando SQL

Texto de comando SQL:

```
select ID, Proveedor, FechaHora_Entrega, FechaHora_Recogida,
DuracionViaje_Segundos,Distancia_Metros,TiempoEspera_Segundos,
UbicacionWKT_Recogida, UbicacionWKT_Entrega
from TaxiViajes_GPS
```

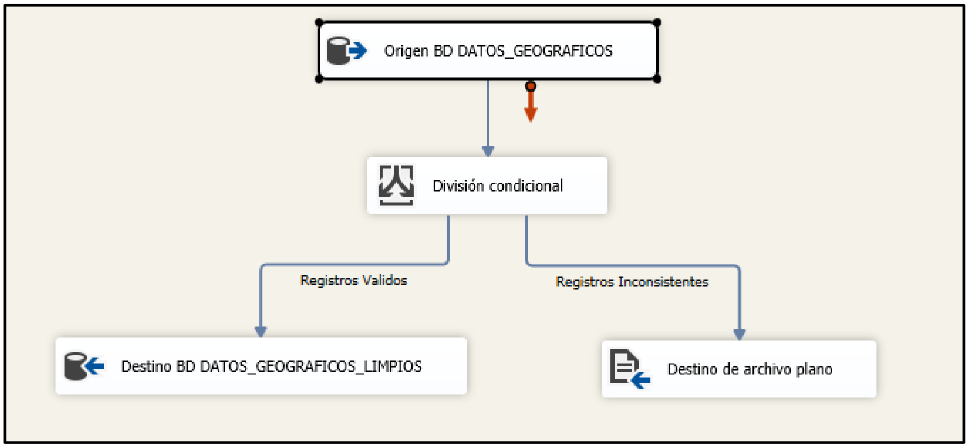
b) Filter inconsistent data using the Conditional Split component which uses the following criteria:
   - **The duration of the trip must be greater than or equal to 60 seconds and less than 5 hours (18,000 seconds),** since it does not make sense for trips to last a few seconds; this would be a type of inconsistency.
   - **The distance in meters must be greater than 100 meters and less than 100,000 meters,** which was probably incorrect data due to the distance indicated.
   - **The waiting time must be less than 2 hours (7200 seconds)** , which, exaggerating, is a time during which you could be waiting for the start of the trip.
   - **Any data that is outside of that range** will be passed to a .CSV to verify that it actually has erroneous data.



Specify the conditions used to direct input rows to specific outputs. If an input row matches no condition, the row is directed to a default output.

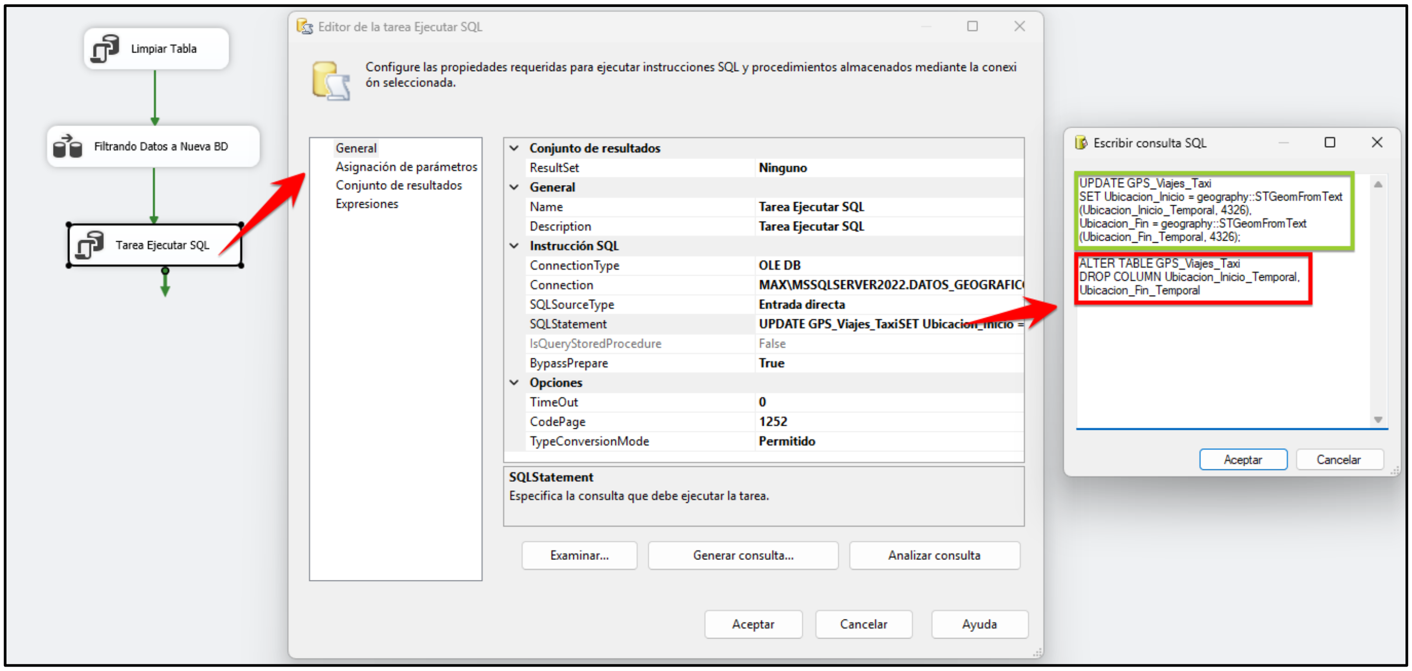
Order	Output Name	Condition
1	Registros Validos	DuracionViaje_Segundos >= 60 && DuracionViaje_Segundos <= 18000 && Distancia_Metros >= 100 && Distancia_Metros <= 100000 && TiempoEspera_Segundos <= 7200
2	Registros Inconsistentes	(DuracionViaje_Segundos < 60) || (DuracionViaje_Segundos > 18000) || (Distancia_Metros < 100) || (Distancia_Metros > 100000) || (TiempoEspera_Segundos > 7200)

Default output name: Salida predeterminada de división condicional



c) Convert and pass the data from **Ubicacion_Inicio_Temporal** and **Ubicacion_Fin_Temporal** to **Ubicacion_Inicio** and **Ubicacion_Fin** respectively.

d) Delete the Temporary columns: Ubicacion_Inicio_Temporal and Ubicacion_Fin_Temporal.



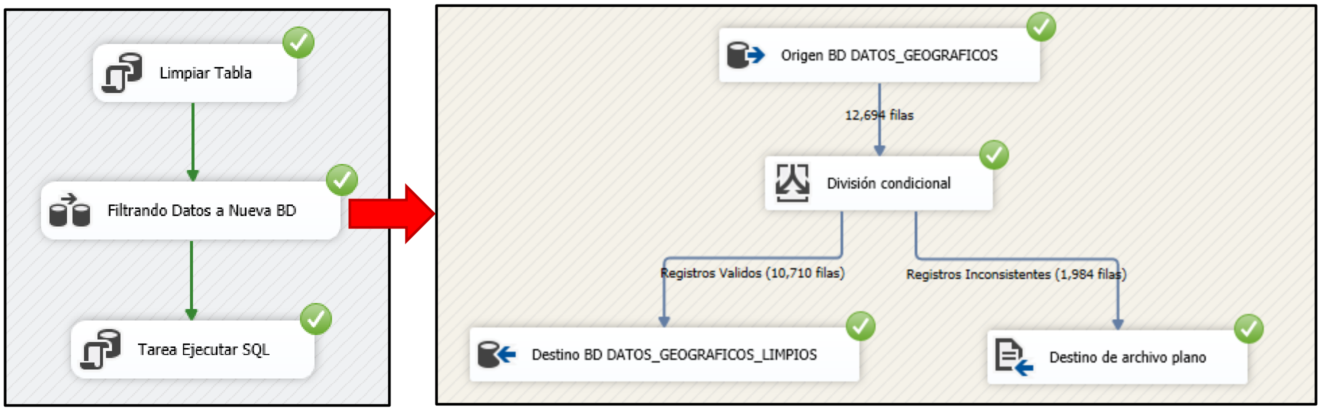
Configure las propiedades requeridas para ejecutar instrucciones SQL y procedimientos almacenados mediante la conexión seleccionada.

Conjunto de resultados	
ResultSet	**Ninguno**
General	
Name	**Tarea Ejecutar SQL**
Description	**Tarea Ejecutar SQL**
Instrucción SQL	
ConnectionType	**OLE DB**
Connection	**MAX\MSSQLSERVER2022.DATOS_GEOGRAFIC**
SQLSourceType	**Entrada directa**
SQLStatement	**UPDATE GPS_Viajes_TaxiSET Ubicacion_Inicio =**
IsQueryStoredProcedure	False
BypassPrepare	**True**
Opciones	
TimeOut	**0**
CodePage	**1252**
TypeConversionMode	**Permitido**

SQLStatement
Especifica la consulta que debe ejecutar la tarea.

```
UPDATE GPS_Viajes_Taxi
SET Ubicacion_Inicio = geography::STGeomFromText
(Ubicacion_Inicio_Temporal, 4326),
Ubicacion_Fin = geography::STGeomFromText
(Ubicacion_Fin_Temporal, 4326);

ALTER TABLE GPS_Viajes_Taxi
DROP COLUMN Ubicacion_Inicio_Temporal,
Ubicacion_Fin_Temporal
```

e) Run the project.



Limpiar Tabla
Filtrando Datos a Nueva BD
Tarea Ejecutar SQL
Origen BD DATOS_GEOGRAFICOS
12,694 filas
División condicional
Registros Validos (10,710 filas)
Registros Inconsistentes (1,984 filas)
Destino BD DATOS_GEOGRAFICOS_LIMPIOS
Destino de archivo plano

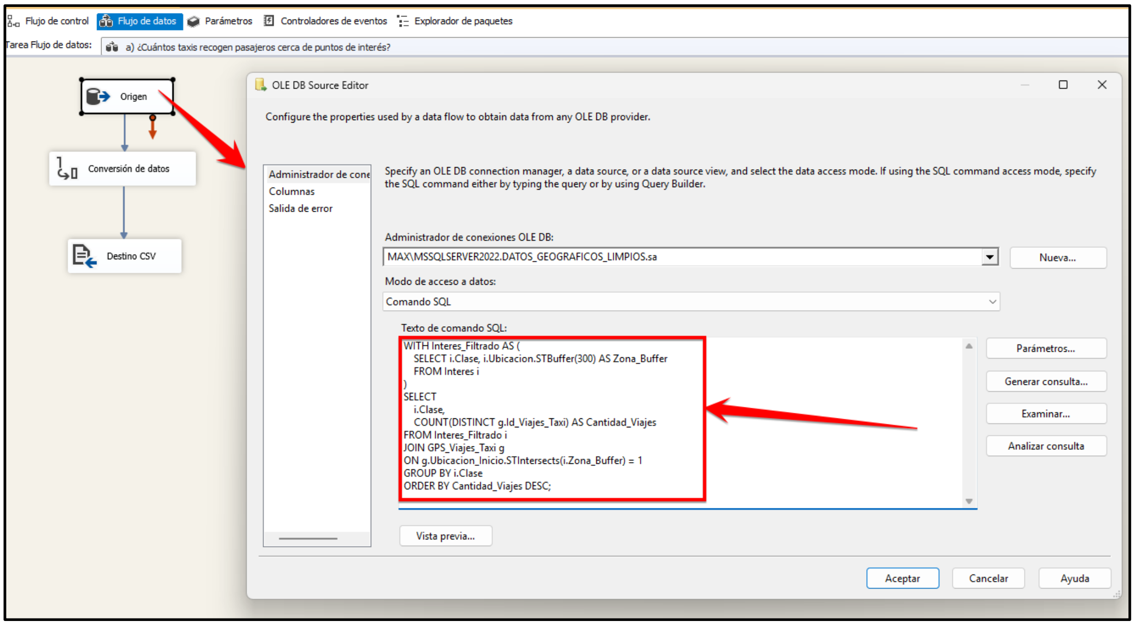
Step 9: Performing Analysis.

To perform the analysis of all the questions, a query was made in SQL Server, which will extract the data that responds to each analysis, then, through an ETL process in SSIS, that data will be saved in a CSV text file.

a) **How many taxis pick up passengers near points of interest (hotels, restaurants, transport stations, parks, etc.)?**

Objective: Identify how many taxis start their trip for each interest group.

Extracting data in SSIS using SQL query:



Tarea Flujo de datos: a) ¿Cuántos taxis recogen pasajeros cerca de puntos de interés?

Configure the properties used by a data flow to obtain data from any OLE DB provider.

Specify an OLE DB connection manager, a data source, or a data source view, and select the data access mode. If using the SQL command access mode, specify the SQL command either by typing the query or by using Query Builder.

Administrador de conexiones OLE DB: MAX\MSSQLSERVER2022.DATOS_GEOGRAFICOS_LIMPIOS.sa

Modo de acceso a datos: Comando SQL

Texto de comando SQL:

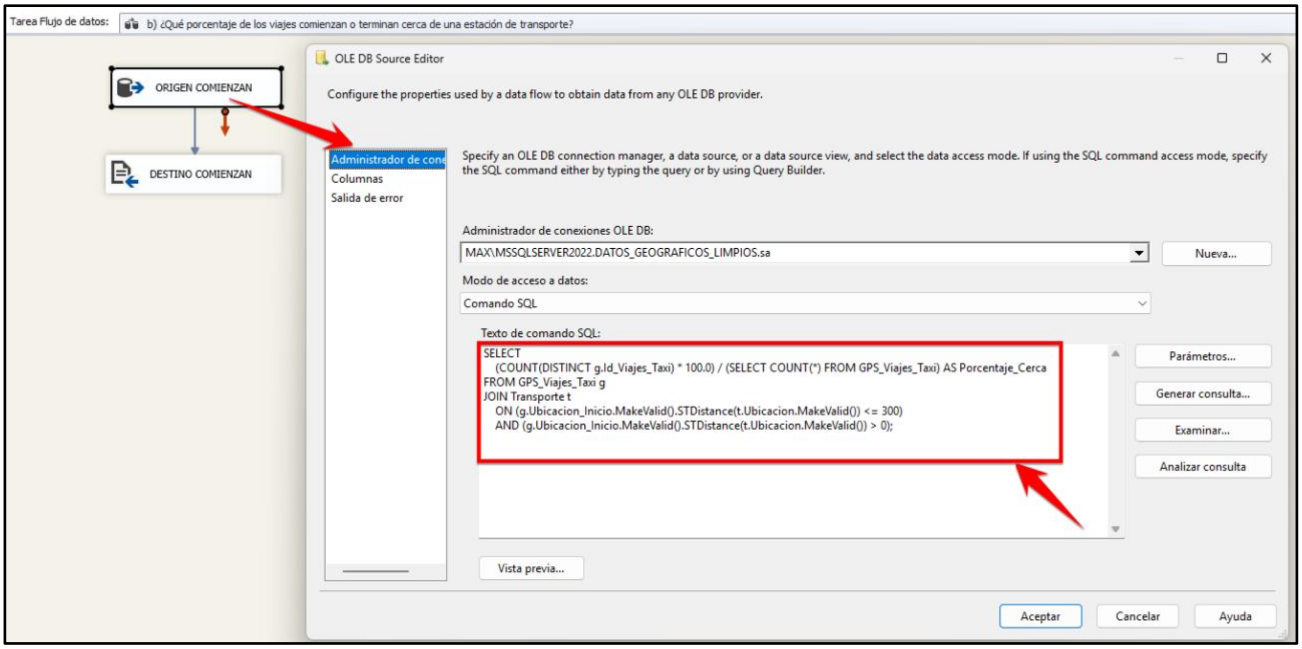
```sql
WITH Interes_Filtrado AS (
    SELECT i.Clase, i.Ubicacion.STBuffer(300) AS Zona_Buffer
    FROM Interes i
)
SELECT
    i.Clase,
    COUNT(DISTINCT g.Id_Viajes_Taxi) AS Cantidad_Viajes
FROM Interes_Filtrado i
JOIN GPS_Viajes_Taxi g
ON g.Ubicacion_Inicio.STIntersects(i.Zona_Buffer) = 1
GROUP BY i.Clase
ORDER BY Cantidad_Viajes DESC;
```

Destination: A CSV destination.

b) What percentage of trips begin or end near a transit station?

Objective: To identify the percentage of trips that begin or end near any transport station.

Extracting data in SSIS using SQL query:



Tarea Flujo de datos: b) ¿Qué porcentaje de los viajes comienzan o terminan cerca de una estación de transporte?

Configure the properties used by a data flow to obtain data from any OLE DB provider.

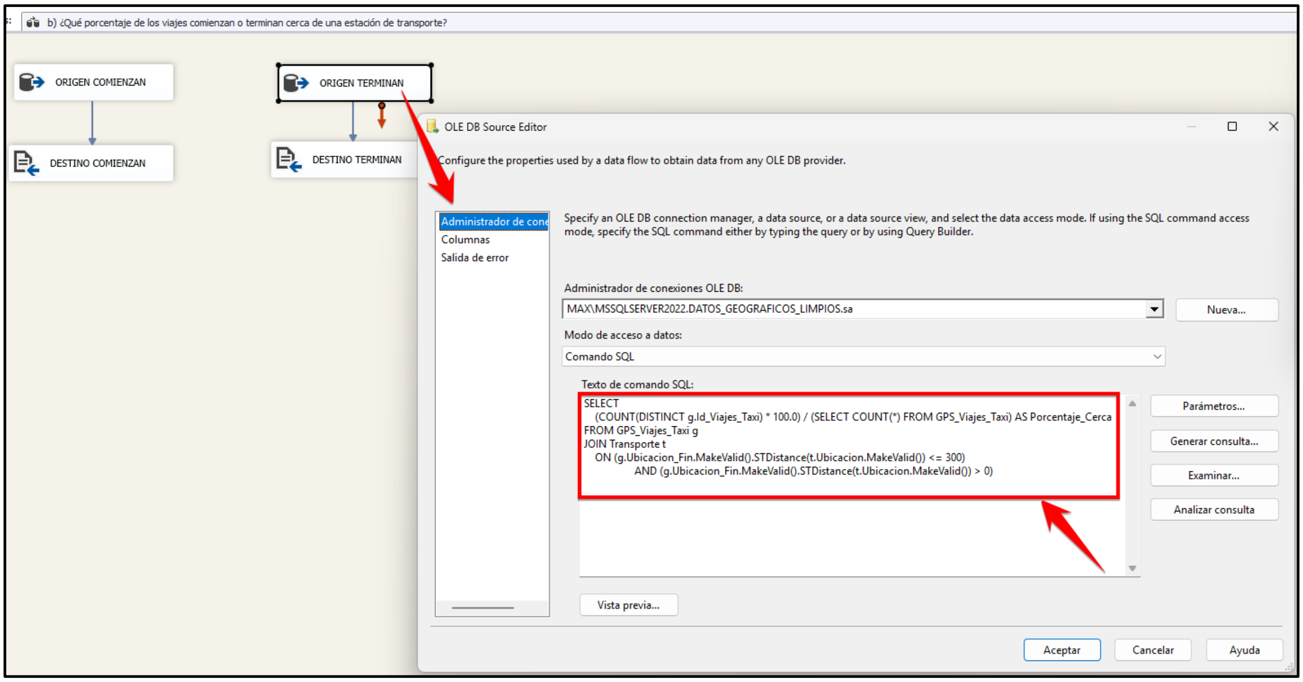
Specify an OLE DB connection manager, a data source, or a data source view, and select the data access mode. If using the SQL command access mode, specify the SQL command either by typing the query or by using Query Builder.

Administrador de conexiones OLE DB: MAX\MSSQLSERVER2022.DATOS_GEOGRAFICOS_LIMPIOS.sa

Modo de acceso a datos: Comando SQL

Texto de comando SQL:

```sql
SELECT
    (COUNT(DISTINCT g.Id_Viajes_Taxi) * 100.0) / (SELECT COUNT(*) FROM GPS_Viajes_Taxi) AS Porcentaje_Cerca
FROM GPS_Viajes_Taxi g
JOIN Transporte t
    ON (g.Ubicacion_Inicio.MakeValid().STDistance(t.Ubicacion.MakeValid()) <= 300)
    AND (g.Ubicacion_Inicio.MakeValid().STDistance(t.Ubicacion.MakeValid()) > 0);
```



b) ¿Qué porcentaje de los viajes comienzan o terminan cerca de una estación de transporte?

OLE DB Source Editor

Configure the properties used by a data flow to obtain data from any OLE DB provider.

Specify an OLE DB connection manager, a data source, or a data source view, and select the data access mode. If using the SQL command access mode, specify the SQL command either by typing the query or by using Query Builder.

Administrador de conexiones OLE DB: MAX\MSSQLSERVER2022.DATOS_GEOGRAFICOS_LIMPIOS.sa

Modo de acceso a datos: Comando SQL

Texto de comando SQL:

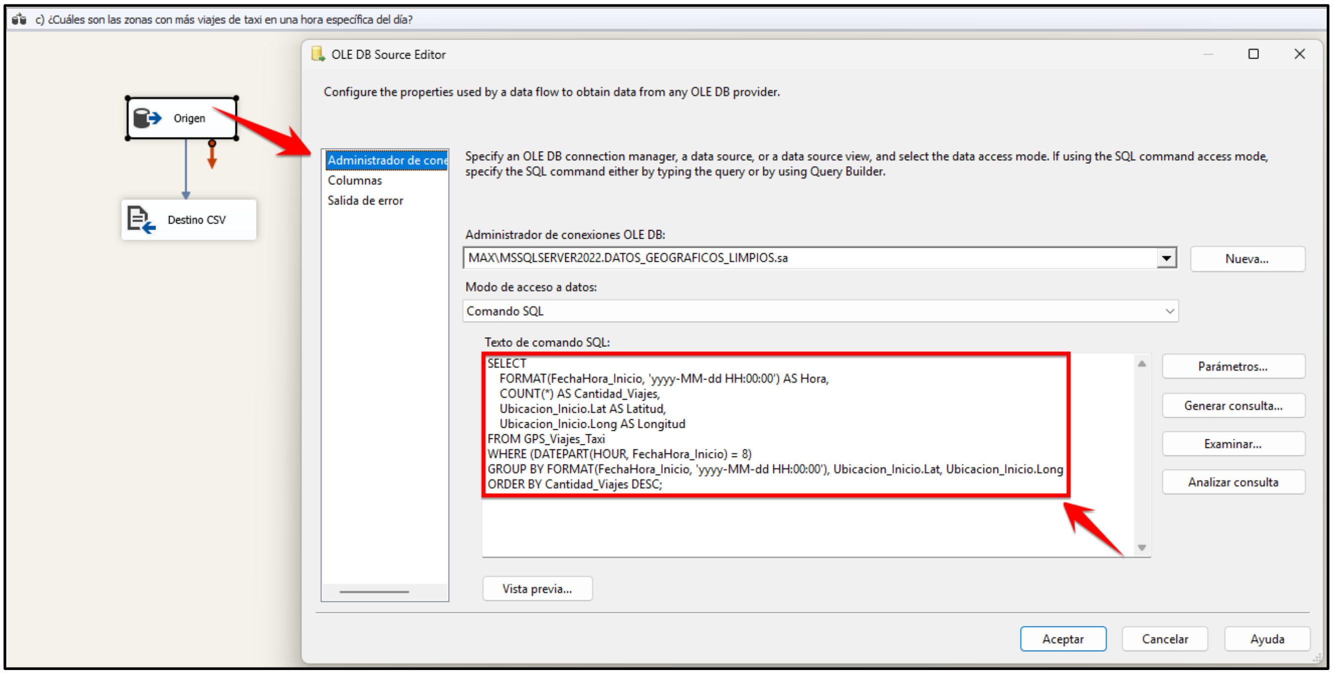
```sql
SELECT
   (COUNT(DISTINCT g.Id_Viajes_Taxi) * 100.0) / (SELECT COUNT(*) FROM GPS_Viajes_Taxi) AS Porcentaje_Cerca
FROM GPS_Viajes_Taxi g
JOIN Transporte t
   ON (g.Ubicacion_Fin.MakeValid().STDistance(t.Ubicacion.MakeValid()) <= 300)
              AND (g.Ubicacion_Fin.MakeValid().STDistance(t.Ubicacion.MakeValid()) > 0)
```

Destination: A CSV destination.

c) What are the areas with the most taxi trips at a specific time of day?

Objective: To identify the areas with the highest concentration of trips at a specific time, in this case at 8:00.

Extracting data in SSIS using SQL query:



c) ¿Cuáles son las zonas con más viajes de taxi en una hora específica del día?

OLE DB Source Editor

Configure the properties used by a data flow to obtain data from any OLE DB provider.

Specify an OLE DB connection manager, a data source, or a data source view, and select the data access mode. If using the SQL command access mode, specify the SQL command either by typing the query or by using Query Builder.

Administrador de conexiones OLE DB: MAX\MSSQLSERVER2022.DATOS_GEOGRAFICOS_LIMPIOS.sa

Modo de acceso a datos: Comando SQL

Texto de comando SQL:

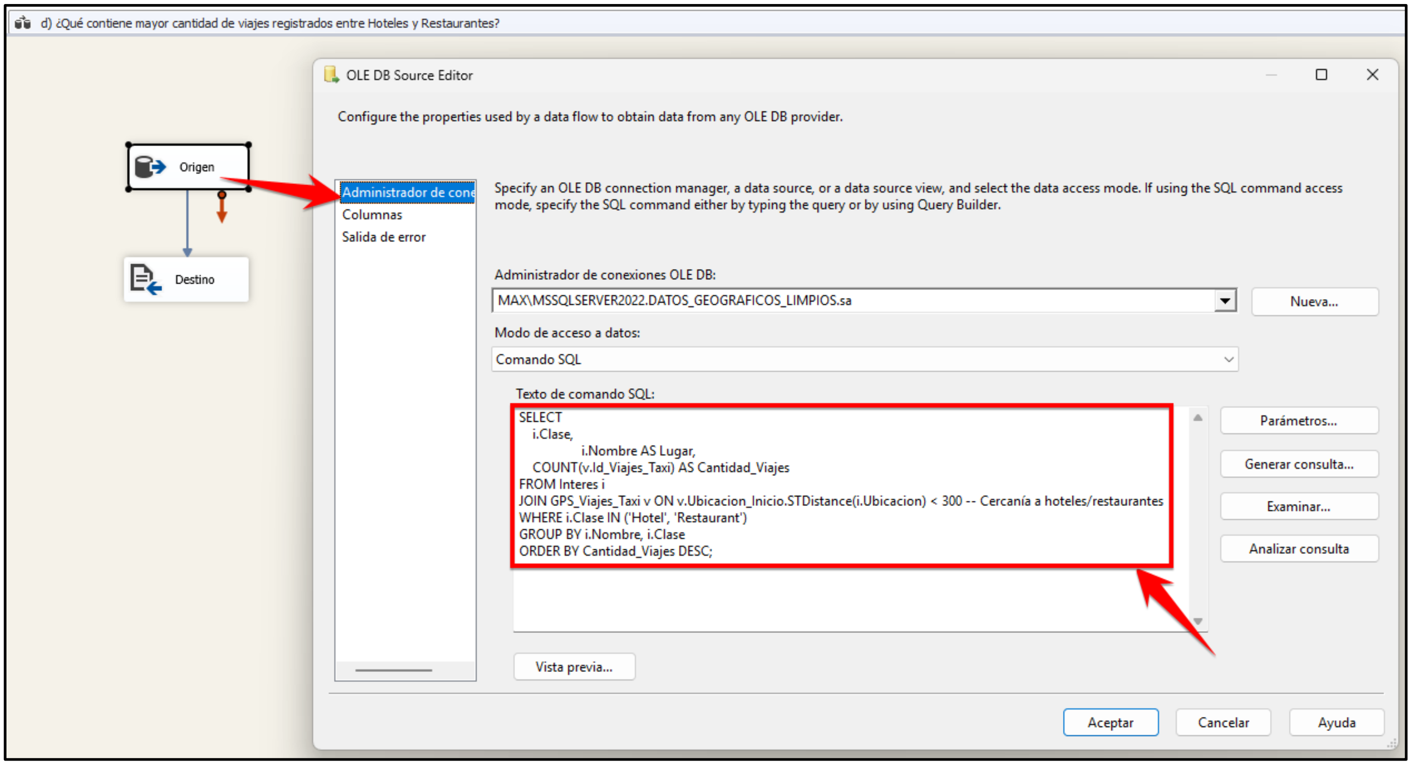
```sql
SELECT
   FORMAT(FechaHora_Inicio, 'yyyy-MM-dd HH:00:00') AS Hora,
   COUNT(*) AS Cantidad_Viajes,
   Ubicacion_Inicio.Lat AS Latitud,
   Ubicacion_Inicio.Long AS Longitud
FROM GPS_Viajes_Taxi
WHERE (DATEPART(HOUR, FechaHora_Inicio) = 8)
GROUP BY FORMAT(FechaHora_Inicio, 'yyyy-MM-dd HH:00:00'), Ubicacion_Inicio.Lat, Ubicacion_Inicio.Long
ORDER BY Cantidad_Viajes DESC;
```

Destination: A CSV destination.

d) **Which point of interest contains the largest number of trips recorded between Hotels and Restaurants?**

Objective: Extract the number of trips between Hotels and Restaurants and identify which one has more.

Extracting data in SSIS using SQL query:



d) ¿Qué contiene mayor cantidad de viajes registrados entre Hoteles y Restaurantes?

OLE DB Source Editor

Configure the properties used by a data flow to obtain data from any OLE DB provider.

Specify an OLE DB connection manager, a data source, or a data source view, and select the data access mode. If using the SQL command access mode, specify the SQL command either by typing the query or by using Query Builder.

Administrador de conexiones OLE DB: MAX\MSSQLSERVER2022.DATOS_GEOGRAFICOS_LIMPIOS.sa

Modo de acceso a datos: Comando SQL

Texto de comando SQL:

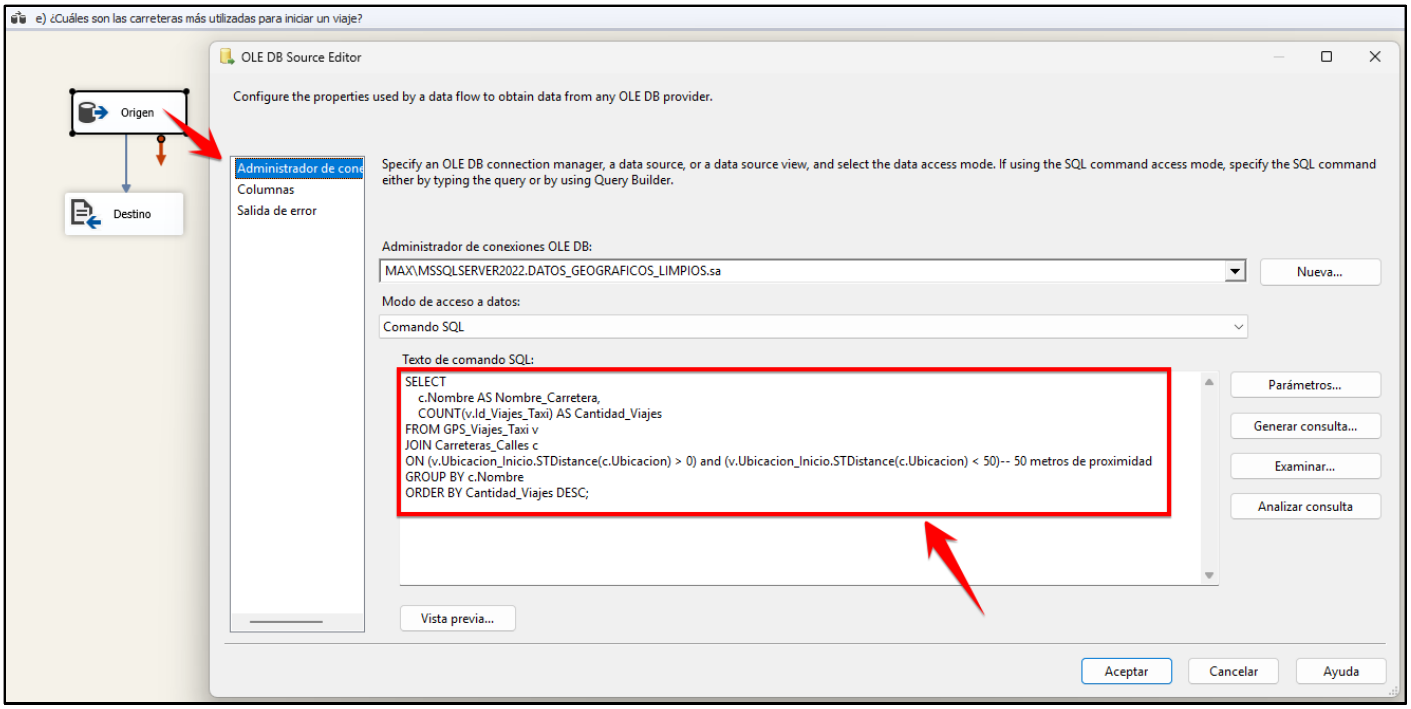
```
SELECT
    i.Clase,
            i.Nombre AS Lugar,
    COUNT(v.Id_Viajes_Taxi) AS Cantidad_Viajes
FROM Interes i
JOIN GPS_Viajes_Taxi v ON v.Ubicacion_Inicio.STDistance(i.Ubicacion) < 300 -- Cercanía a hoteles/restaurantes
WHERE i.Clase IN ('Hotel', 'Restaurant')
GROUP BY i.Nombre, i.Clase
ORDER BY Cantidad_Viajes DESC;
```

Destination: A CSV destination.

e) **What are the most popular roads to start a trip?**

Objective: To determine which roads most trips begin on.

Extracting data in SSIS using SQL query:



Configure the properties used by a data flow to obtain data from any OLE DB provider.

Specify an OLE DB connection manager, a data source, or a data source view, and select the data access mode. If using the SQL command access mode, specify the SQL command either by typing the query or by using Query Builder.

Administrador de conexiones OLE DB: MAX\MSSQLSERVER2022.DATOS_GEOGRAFICOS_LIMPIOS.sa

Modo de acceso a datos: Comando SQL

Texto de comando SQL:

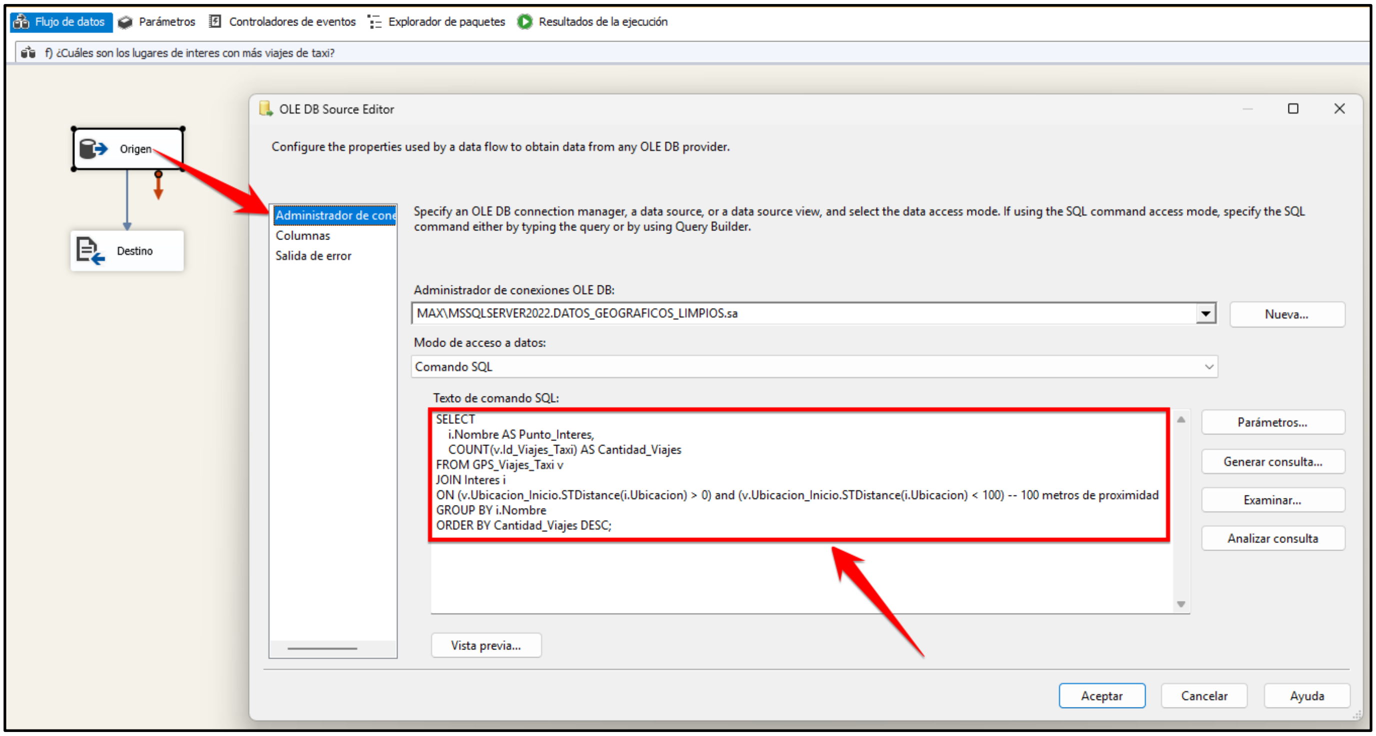
```sql
SELECT
   c.Nombre AS Nombre_Carretera,
   COUNT(v.Id_Viajes_Taxi) AS Cantidad_Viajes
FROM GPS_Viajes_Taxi v
JOIN Carreteras_Calles c
ON (v.Ubicacion_Inicio.STDistance(c.Ubicacion) > 0) and (v.Ubicacion_Inicio.STDistance(c.Ubicacion) < 50)-- 50 metros de proximidad
GROUP BY c.Nombre
ORDER BY Cantidad_Viajes DESC;
```

Destination: A CSV destination.

f) What are the places of interest with the most taxi rides?

Objective: Determine which places of interest generate the most taxi traffic.

Extracting data in SSIS using SQL query:



Configure the properties used by a data flow to obtain data from any OLE DB provider.

Specify an OLE DB connection manager, a data source, or a data source view, and select the data access mode. If using the SQL command access mode, specify the SQL command either by typing the query or by using Query Builder.

Administrador de conexiones OLE DB: MAX\MSSQLSERVER2022.DATOS_GEOGRAFICOS_LIMPIOS.sa

Modo de acceso a datos: Comando SQL

Texto de comando SQL:

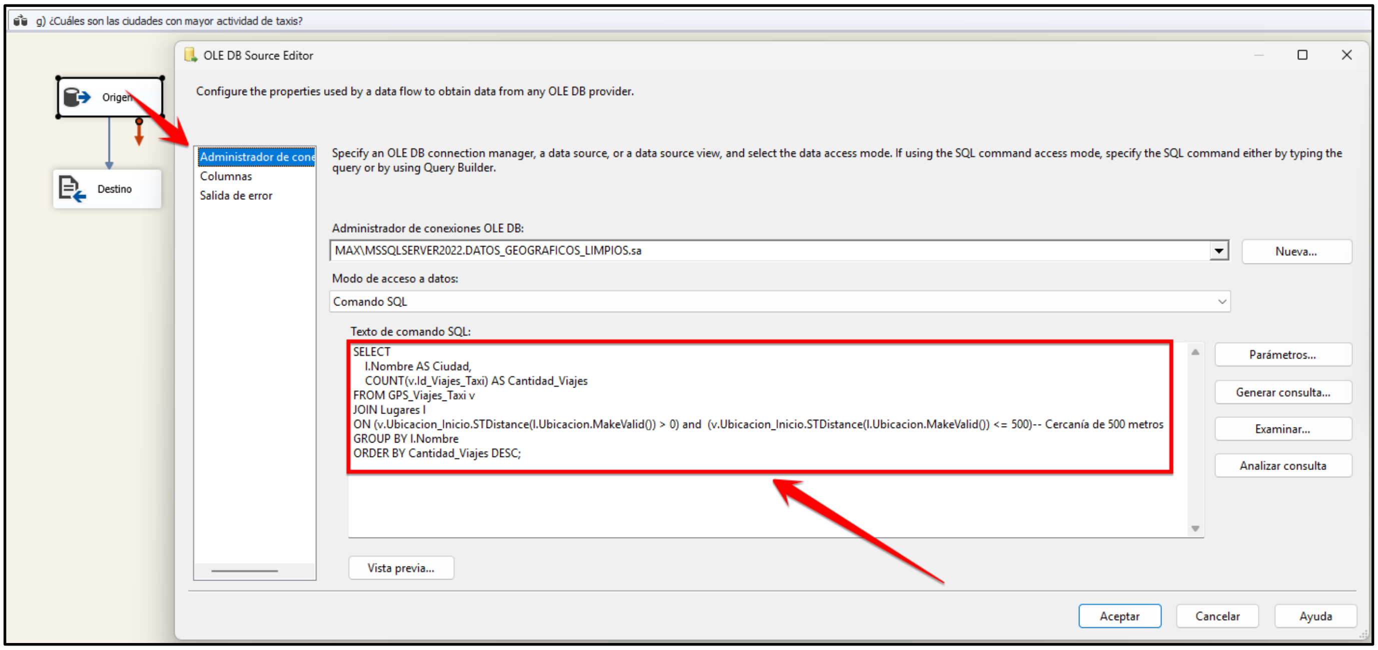
```sql
SELECT
   i.Nombre AS Punto_Interes,
   COUNT(v.Id_Viajes_Taxi) AS Cantidad_Viajes
FROM GPS_Viajes_Taxi v
JOIN Interes i
ON (v.Ubicacion_Inicio.STDistance(i.Ubicacion) > 0) and (v.Ubicacion_Inicio.STDistance(i.Ubicacion) < 100) -- 100 metros de proximidad
GROUP BY i.Nombre
ORDER BY Cantidad_Viajes DESC;
```

Destination: A CSV destination.

g) Which cities have the highest taxi activity?

Objective: Identify which cities have the most trips.

Extracting data in SSIS using SQL query:



Configure the properties used by a data flow to obtain data from any OLE DB provider.

Specify an OLE DB connection manager, a data source, or a data source view, and select the data access mode. If using the SQL command access mode, specify the SQL command either by typing the query or by using Query Builder.

Administrador de conexiones OLE DB: MAX\MSSQLSERVER2022.DATOS_GEOGRAFICOS_LIMPIOS.sa

Modo de acceso a datos: Comando SQL

Texto de comando SQL:

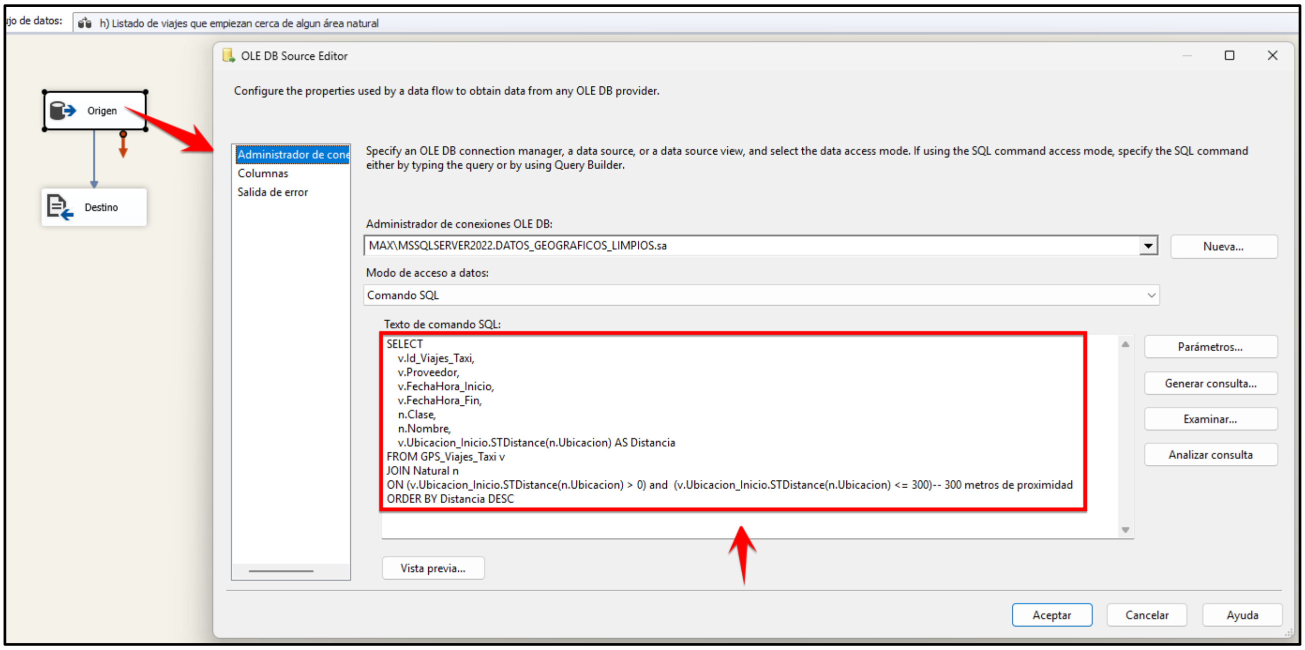
```
SELECT
   l.Nombre AS Ciudad,
   COUNT(v.Id_Viajes_Taxi) AS Cantidad_Viajes
FROM GPS_Viajes_Taxi v
JOIN Lugares l
ON (v.Ubicacion_Inicio.STDistance(l.Ubicacion.MakeValid()) > 0) and  (v.Ubicacion_Inicio.STDistance(l.Ubicacion.MakeValid()) <= 500)-- Cercanía de 500 metros
GROUP BY l.Nombre
ORDER BY Cantidad_Viajes DESC;
```

Destination: A CSV destination.

h) List of trips that start near a natural area.

Objective: Identify trips that begin in a natural area.

Extracting data in SSIS using SQL query:



Configure the properties used by a data flow to obtain data from any OLE DB provider.

Specify an OLE DB connection manager, a data source, or a data source view, and select the data access mode. If using the SQL command access mode, specify the SQL command either by typing the query or by using Query Builder.

Administrador de conexiones OLE DB: MAX\MSSQLSERVER2022.DATOS_GEOGRAFICOS_LIMPIOS.sa

Modo de acceso a datos: Comando SQL

Texto de comando SQL:

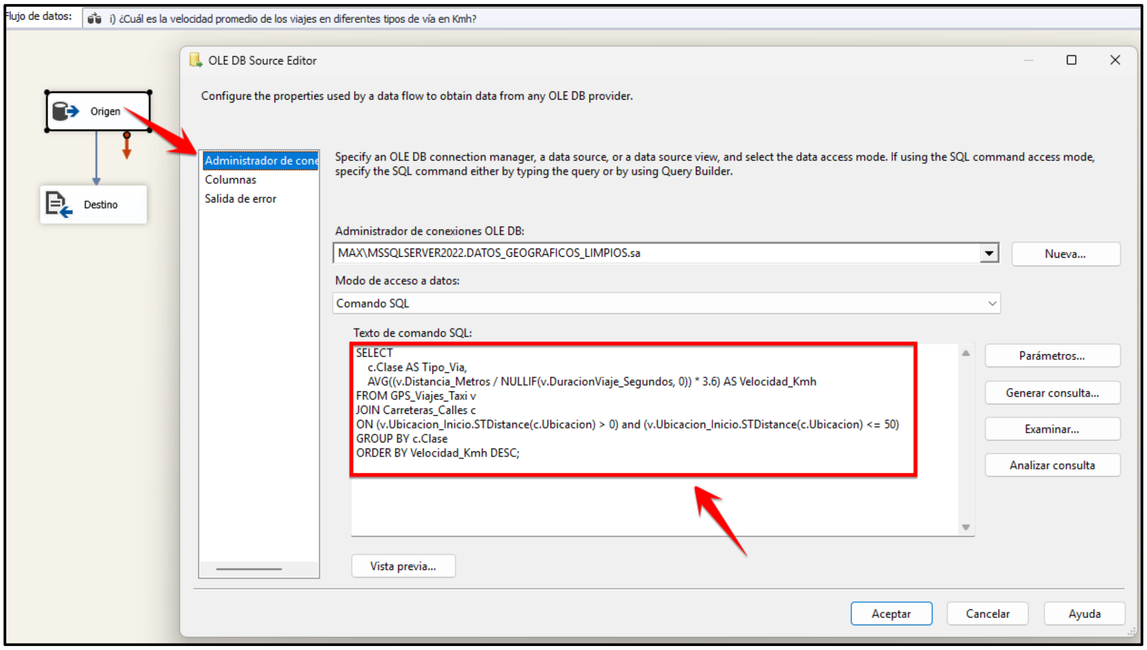
```sql
SELECT
    v.Id_Viajes_Taxi,
    v.Proveedor,
    v.FechaHora_Inicio,
    v.FechaHora_Fin,
    n.Clase,
    n.Nombre,
    v.Ubicacion_Inicio.STDistance(n.Ubicacion) AS Distancia
FROM GPS_Viajes_Taxi v
JOIN Natural n
ON (v.Ubicacion_Inicio.STDistance(n.Ubicacion) > 0) and  (v.Ubicacion_Inicio.STDistance(n.Ubicacion) <= 300)-- 300 metros de proximidad
ORDER BY Distancia DESC
```

Destination: A CSV destination.

i) What is the average speed of travel on different types of roads in km/h?

Objective: Identify trips that begin in a natural area.

Extracting data in SSIS using SQL query:



Configure the properties used by a data flow to obtain data from any OLE DB provider.

Specify an OLE DB connection manager, a data source, or a data source view, and select the data access mode. If using the SQL command access mode, specify the SQL command either by typing the query or by using Query Builder.

Administrador de conexiones OLE DB: MAX\MSSQLSERVER2022.DATOS_GEOGRAFICOS_LIMPIOS.sa

Modo de acceso a datos: Comando SQL

Texto de comando SQL:

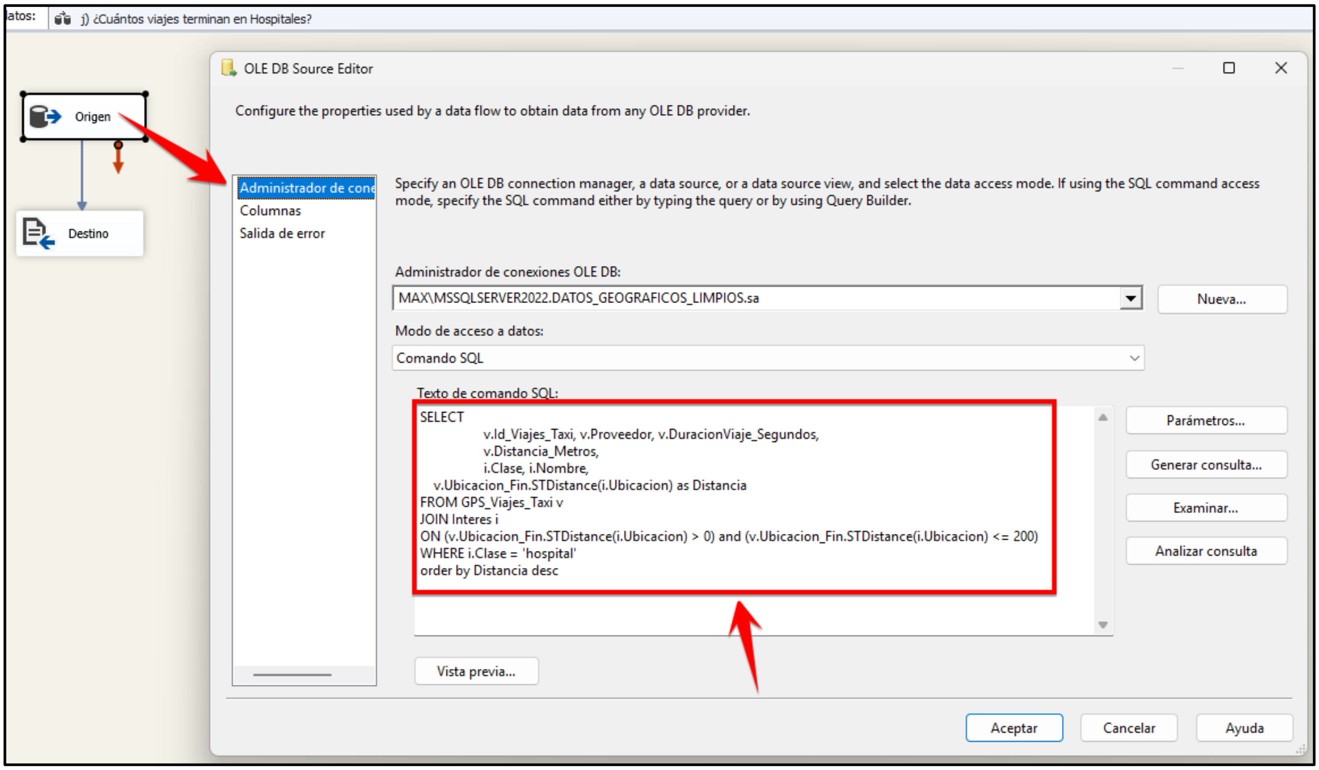
```sql
SELECT
    c.Clase AS Tipo_Via,
    AVG((v.Distancia_Metros / NULLIF(v.DuracionViaje_Segundos, 0)) * 3.6) AS Velocidad_Kmh
FROM GPS_Viajes_Taxi v
JOIN Carreteras_Calles c
ON (v.Ubicacion_Inicio.STDistance(c.Ubicacion) > 0) and (v.Ubicacion_Inicio.STDistance(c.Ubicacion) <= 50)
GROUP BY c.Clase
ORDER BY Velocidad_Kmh DESC;
```

Destination: A CSV destination.

j) How many trips end in hospitals?

Objective: To measure taxi activity in hospital areas.

Extracting data in SSIS using SQL query:



Configure the properties used by a data flow to obtain data from any OLE DB provider.

Specify an OLE DB connection manager, a data source, or a data source view, and select the data access mode. If using the SQL command access mode, specify the SQL command either by typing the query or by using Query Builder.

Administrador de conexiones OLE DB: MAX\MSSQLSERVER2022.DATOS_GEOGRAFICOS_LIMPIOS.sa

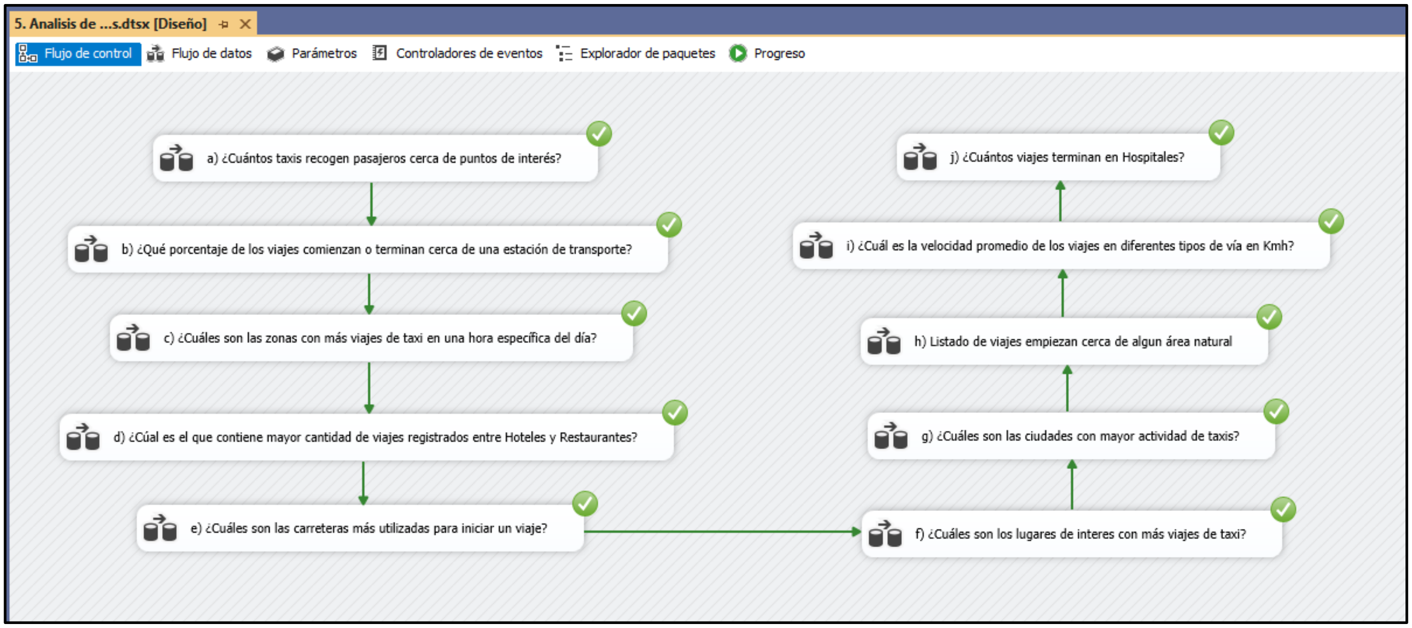
Modo de acceso a datos: Comando SQL

Texto de comando SQL:

```
SELECT
            v.Id_Viajes_Taxi, v.Proveedor, v.DuracionViaje_Segundos,
            v.Distancia_Metros,
            i.Clase, i.Nombre,
   v.Ubicacion_Fin.STDistance(i.Ubicacion) as Distancia
FROM GPS_Viajes_Taxi v
JOIN Interes i
ON (v.Ubicacion_Fin.STDistance(i.Ubicacion) > 0) and (v.Ubicacion_Fin.STDistance(i.Ubicacion) <= 200)
WHERE i.Clase = 'hospital'
order by Distancia desc
```

Destination: A CSV destination.

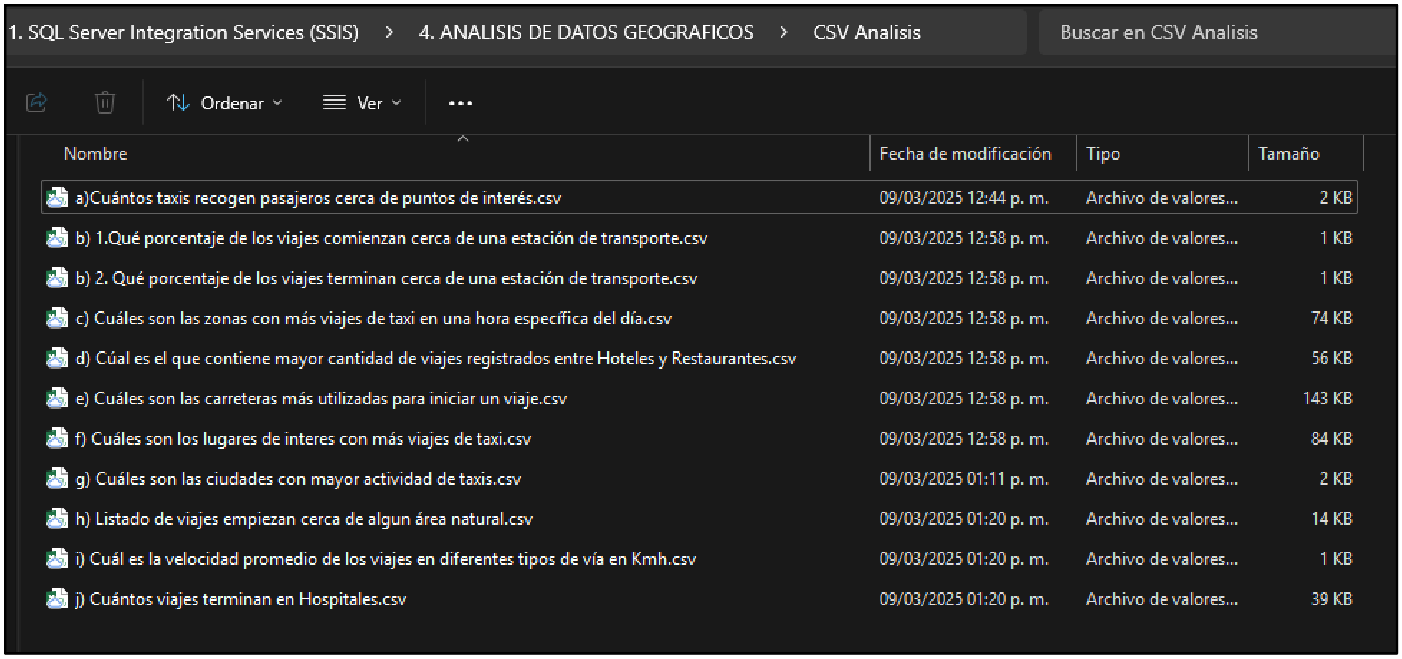
RUNNING THE FLOW TASKS:



5. Analisis de ...s.dtsx [Diseño]
Flujo de control
Flujo de datos
Parámetros
Controladores de eventos
Explorador de paquetes
Progreso
a) ¿Cuántos taxis recogen pasajeros cerca de puntos de interés?
b) ¿Qué porcentaje de los viajes comienzan o terminan cerca de una estación de transporte?
c) ¿Cuáles son las zonas con más viajes de taxi en una hora específica del día?
d) ¿Cúal es el que contiene mayor cantidad de viajes registrados entre Hoteles y Restaurantes?
e) ¿Cuáles son las carreteras más utilizadas para iniciar un viaje?
f) ¿Cuáles son los lugares de interes con más viajes de taxi?
g) ¿Cuáles son las ciudades con mayor actividad de taxis?
h) Listado de viajes empiezan cerca de algun área natural
i) ¿Cuál es la velocidad promedio de los viajes en diferentes tipos de vía en Kmh?
j) ¿Cuántos viajes terminan en Hospitales?

III. REVIEW OF ANALYSIS

When running the project, you can see that each CSV was saved correctly. Let's review each one of them:

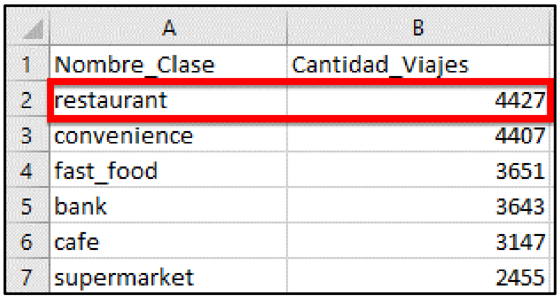

1. SQL Server Integration Services (SSIS) > 4. ANALISIS DE DATOS GEOGRAFICOS > CSV Analisis
Buscar en CSV Analisis
Ordenar
Ver
...

Nombre	Fecha de modificación	Tipo	Tamaño
a)Cuántos taxis recogen pasajeros cerca de puntos de interés.csv	09/03/2025 12:44 p. m.	Archivo de valores...	2 KB
b) 1.Qué porcentaje de los viajes comienzan cerca de una estación de transporte.csv	09/03/2025 12:58 p. m.	Archivo de valores...	1 KB
b) 2. Qué porcentaje de los viajes terminan cerca de una estación de transporte.csv	09/03/2025 12:58 p. m.	Archivo de valores...	1 KB
c) Cuáles son las zonas con más viajes de taxi en una hora específica del día.csv	09/03/2025 12:58 p. m.	Archivo de valores...	74 KB
d) Cúal es el que contiene mayor cantidad de viajes registrados entre Hoteles y Restaurantes.csv	09/03/2025 12:58 p. m.	Archivo de valores...	56 KB
e) Cuáles son las carreteras más utilizadas para iniciar un viaje.csv	09/03/2025 12:58 p. m.	Archivo de valores...	143 KB
f) Cuáles son los lugares de interes con más viajes de taxi.csv	09/03/2025 12:58 p. m.	Archivo de valores...	84 KB
g) Cuáles son las ciudades con mayor actividad de taxis.csv	09/03/2025 01:11 p. m.	Archivo de valores...	2 KB
h) Listado de viajes empiezan cerca de algun área natural.csv	09/03/2025 01:20 p. m.	Archivo de valores...	14 KB
i) Cuál es la velocidad promedio de los viajes en diferentes tipos de vía en Kmh.csv	09/03/2025 01:20 p. m.	Archivo de valores...	1 KB
j) Cuántos viajes terminan en Hospitales.csv	09/03/2025 01:20 p. m.	Archivo de valores...	39 KB

1. **Analyzing Results:**

   a) **How many taxis pick up passengers near points of interest (hotels, restaurants, transport stations, parks, etc.)?**

   CSV Result:

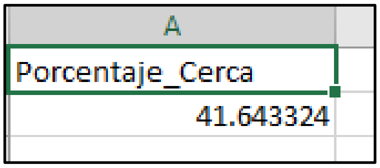


	A	B
1	Nombre_Clase	Cantidad_Viajes
2	restaurant	4427
3	convenience	4407
4	fast_food	3651
5	bank	3643
6	cafe	3147
7	supermarket	2455

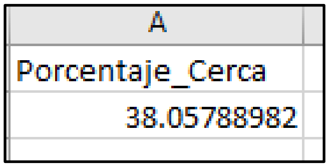
Conclusion: The POI of “Restaurant” has a total of 4427 taxi trips, followed by “ Convenience ” with 4407 trips, and “ fast_food ” with 3651, these are the POIs that have the most trips.

b) What percentage of trips begin or end near a transit station?

CSV Result:



A
Porcentaje_Cerca
41.643324

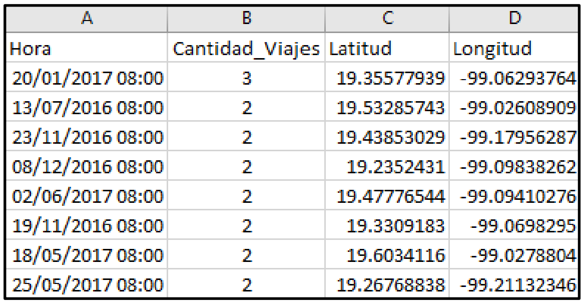


A
Porcentaje_Cerca
38.05788982

Conclusion: 41.64% of trips started near a transport station and 38.05% ended near one.

c) What are the areas with the most taxi trips at a specific time of day?

CSV Result:

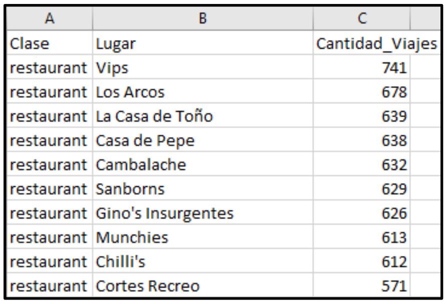


A	B	C	D
Hora	Cantidad_Viajes	Latitud	Longitud
20/01/2017 08:00	3	19.35577939	-99.06293764
13/07/2016 08:00	2	19.53285743	-99.02608909
23/11/2016 08:00	2	19.43853029	-99.17956287
08/12/2016 08:00	2	19.2352431	-99.09838262
02/06/2017 08:00	2	19.47776544	-99.09410276
19/11/2016 08:00	2	19.3309183	-99.0698295
18/05/2017 08:00	2	19.6034116	-99.0278804
25/05/2017 08:00	2	19.26768838	-99.21132346

Conclusion: The area with latitude and longitude shown in the image represents the number of trips that were made in that hour, in this case, 3 trips in the same area, on the same day, at 8:00 am

d) Which point of interest contains the highest number of trips recorded between Hotels and Restaurants?

CSV Result:

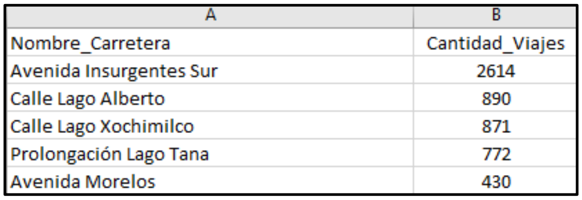


A	B	C
Clase	Lugar	Cantidad_Viajes
restaurant	Vips	741
restaurant	Los Arcos	678
restaurant	La Casa de Toño	639
restaurant	Casa de Pepe	638
restaurant	Cambalache	632
restaurant	Sanborns	629
restaurant	Gino's Insurgentes	626
restaurant	Munchies	613
restaurant	Chilli's	612
restaurant	Cortes Recreo	571

Conclusion: It is observed that the one containing the largest number of trips is made by the Restaurant Class.

e) What are the most popular roads to start a trip?

CSV Result:

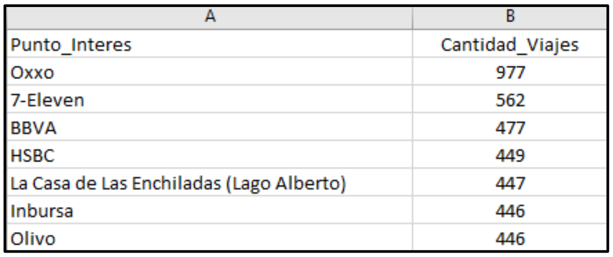


A	B
Nombre_Carretera	Cantidad_Viajes
Avenida Insurgentes Sur	2614
Calle Lago Alberto	890
Calle Lago Xochimilco	871
Prolongación Lago Tana	772
Avenida Morelos	430

Conclusion: Insurgentes Sur Avenue was the most used route as a starting point.

f) What are the places of interest with the most taxi rides?

CSV Result:

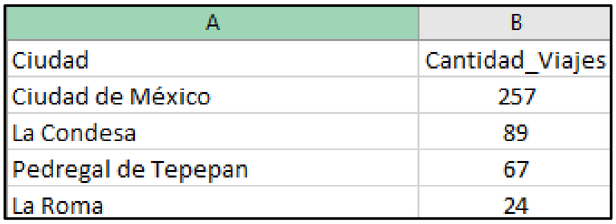


A	B
Punto_Interes	Cantidad_Viajes
Oxxo	977
7-Eleven	562
BBVA	477
HSBC	449
La Casa de Las Enchiladas (Lago Alberto)	447
Inbursa	446
Olivo	446

Conclusion: The Oxxo turned out to be one of the places of interest with the most taxi activity.

g) Which cities have the highest taxi activity?

CSV Result:

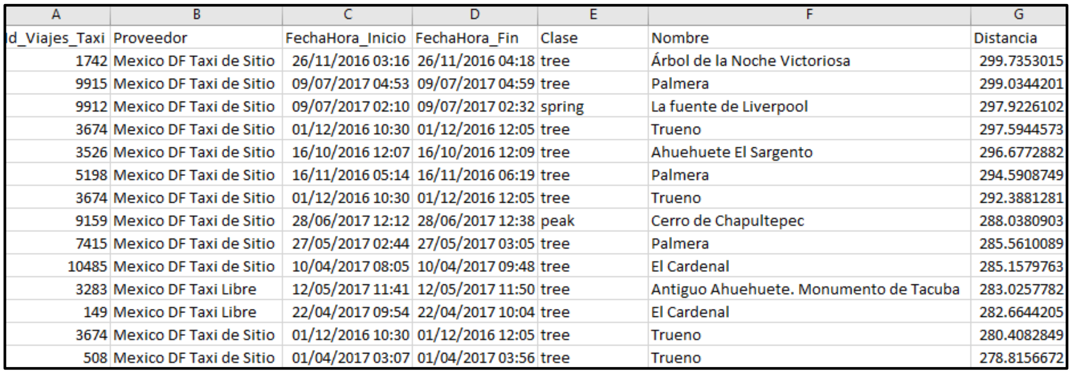


A	B
Ciudad	Cantidad_Viajes
Ciudad de México	257
La Condesa	89
Pedregal de Tepepan	67
La Roma	24

Conclusion: Mexico City concentrated the majority of trip starts.

h) List of trips that start near a natural area.

CSV Result:

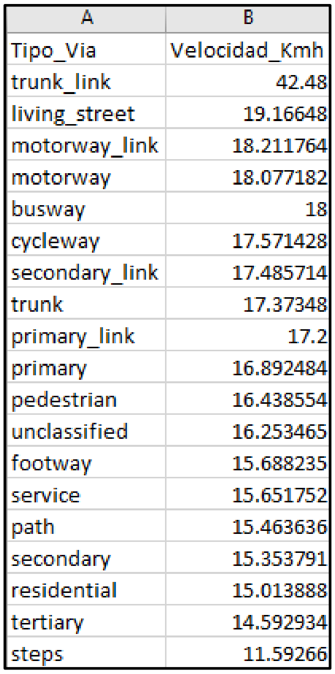


A	B	C	D	E	F	G
Id_Viajes_Taxi	Proveedor	FechaHora_Inicio	FechaHora_Fin	Clase	Nombre	Distancia
1742	Mexico DF Taxi de Sitio	26/11/2016 03:16	26/11/2016 04:18	tree	Árbol de la Noche Victoriosa	299.7353015
9915	Mexico DF Taxi de Sitio	09/07/2017 04:53	09/07/2017 04:59	tree	Palmera	299.0344201
9912	Mexico DF Taxi de Sitio	09/07/2017 02:10	09/07/2017 02:32	spring	La fuente de Liverpool	297.9226102
3674	Mexico DF Taxi de Sitio	01/12/2016 10:30	01/12/2016 12:05	tree	Trueno	297.5944573
3526	Mexico DF Taxi de Sitio	16/10/2016 12:07	16/10/2016 12:09	tree	Ahuehuete El Sargento	296.6772882
5198	Mexico DF Taxi de Sitio	16/11/2016 05:14	16/11/2016 06:19	tree	Palmera	294.5908749
3674	Mexico DF Taxi de Sitio	01/12/2016 10:30	01/12/2016 12:05	tree	Trueno	292.3881281
9159	Mexico DF Taxi de Sitio	28/06/2017 12:12	28/06/2017 12:38	peak	Cerro de Chapultepec	288.0380903
7415	Mexico DF Taxi de Sitio	27/05/2017 02:44	27/05/2017 03:05	tree	Palmera	285.5610089
10485	Mexico DF Taxi de Sitio	10/04/2017 08:05	10/04/2017 09:48	tree	El Cardenal	285.1579763
3283	Mexico DF Taxi Libre	12/05/2017 11:41	12/05/2017 11:50	tree	Antiguo Ahuehuete. Monumento de Tacuba	283.0257782
149	Mexico DF Taxi Libre	22/04/2017 09:54	22/04/2017 10:04	tree	El Cardenal	282.6644205
3674	Mexico DF Taxi de Sitio	01/12/2016 10:30	01/12/2016 12:05	tree	Trueno	280.4082849
508	Mexico DF Taxi de Sitio	01/04/2017 03:07	01/04/2017 03:56	tree	Trueno	278.8156672

Conclusion: Multiple trips were identified that started near natural areas, which could be used to assess transport demand in recreational or rural areas.

i) What is the average speed of travel on different types of roads in km/h?

CSV Result:

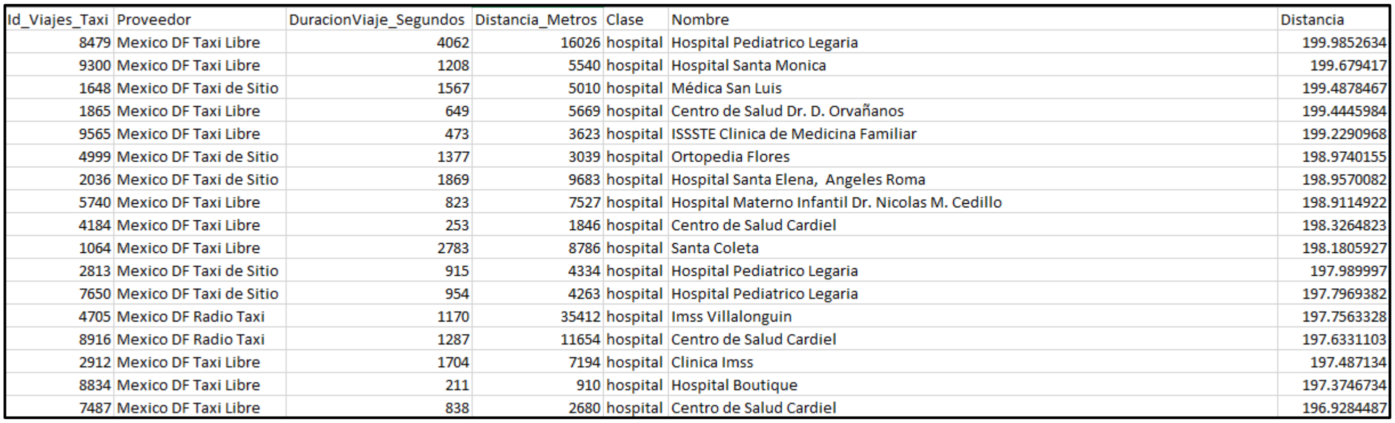


A	B
Tipo_Via	Velocidad_Kmh
trunk_link	42.48
living_street	19.16648
motorway_link	18.211764
motorway	18.077182
busway	18
cycleway	17.571428
secondary_link	17.485714
trunk	17.37348
primary_link	17.2
primary	16.892484
pedestrian	16.438554
unclassified	16.253465
footway	15.688235
service	15.651752
path	15.463636
secondary	15.353791
residential	15.013888
tertiary	14.592934
steps	11.59266

Conclusion: Average speeds by type of road were obtained, useful for traffic analysis.

j) How many trips end in hospitals?

CSV Result:



Id_Viajes_Taxi	Proveedor	DuracionViaje_Segundos	Distancia_Metros	Clase	Nombre	Distancia
8479	Mexico DF Taxi Libre	4062	16026	hospital	Hospital Pediatrico Legaria	199.9852634
9300	Mexico DF Taxi Libre	1208	5540	hospital	Hospital Santa Monica	199.679417
1648	Mexico DF Taxi de Sitio	1567	5010	hospital	Médica San Luis	199.4878467
1865	Mexico DF Taxi Libre	649	5669	hospital	Centro de Salud Dr. D. Orvañanos	199.4445984
9565	Mexico DF Taxi Libre	473	3623	hospital	ISSSTE Clinica de Medicina Familiar	199.2290968
4999	Mexico DF Taxi de Sitio	1377	3039	hospital	Ortopedia Flores	198.9740155
2036	Mexico DF Taxi de Sitio	1869	9683	hospital	Hospital Santa Elena, Angeles Roma	198.9570082
5740	Mexico DF Taxi Libre	823	7527	hospital	Hospital Materno Infantil Dr. Nicolas M. Cedillo	198.9114922
4184	Mexico DF Taxi Libre	253	1846	hospital	Centro de Salud Cardiel	198.3264823
1064	Mexico DF Taxi Libre	2783	8786	hospital	Santa Coleta	198.1805927
2813	Mexico DF Taxi de Sitio	915	4334	hospital	Hospital Pediatrico Legaria	197.989997
7650	Mexico DF Taxi de Sitio	954	4263	hospital	Hospital Pediatrico Legaria	197.7969382
4705	Mexico DF Radio Taxi	1170	35412	hospital	Imss Villalonguin	197.7563328
8916	Mexico DF Radio Taxi	1287	11654	hospital	Centro de Salud Cardiel	197.6331103
2912	Mexico DF Taxi Libre	1704	7194	hospital	Clinica Imss	197.487134
8834	Mexico DF Taxi Libre	211	910	hospital	Hospital Boutique	197.3746734
7487	Mexico DF Taxi Libre	838	2680	hospital	Centro de Salud Cardiel	196.9284487

Conclusion: It shows a list of more than 400 records of taxi trips that have a hospital as their final destination.

IV. CONCLUSIONS

Throughout the development of this project, a complete flow of integration, transformation and analysis of geographic data was successfully implemented using tools such as SSIS, SQL Server, QGIS and languages such as C#. The main objective was to take advantage of spatial data (Shapefiles) and GPS data from taxi trips to obtain valuable information about urban transport behavior in Mexico City.

The main achievements include:

- **Effective integration of geospatial data** into a relational database environment, overcoming the challenges involved in managing GEOGRAPHY and GEOMETRY data.
- **Data cleaning and filtering** , which ensured the quality of the information used in the analyses.
- **Automation of the ETL process** , facilitating future updates or replication of the project in other cities.
- **Obtaining key indicators** , such as areas with the highest demand for taxis, points of interest with the highest activity, road use, and behavior according to type of road.
- **View and export results** in CSV files, useful for executive reports or as inputs for other analytical tools or data visualization.

This project allowed me to realize how the use of spatial data together with analysis tools can provide very valuable information for making decisions on issues such as urban transport, territorial planning or even to better understand how the city moves and how mobility could be improved.

This project was developed as part of my professional portfolio in geographic data analysis.