Building Sales Data Mart Using SSIS

First, we downloaded an AdventureWorks2014 database to make it our data source.

Then we made our DWH database called Eo-AdventureWorksDW2014 to make it as destination database.

Start to make the tables that I will store the data on it like Product, customer, territory, date as dimension tables and sales as fact tables.

Make each column with its constraints and add some columns like:

- source system code to know each row comes from which system.
- slowly changing dimension (SCD) like start date, end date and is_current column for the historical attributes.

Make our primary keys and foreign keys and some indexes for the table to improve the performance of selecting and retrieving the data.

Data Modeling: make our data model and relationships for these tables with star schema.

Then we start to make our SQL Server Integration Service (SSIS) project:

To run packages and use Data Flow to get the data from our source and make some transformations for it then put it in our DWH destination.
and using connection manager as OLE DB for the source and destination database.

For example:

1. Dim Product:

- get the data from the source as OLE DB Provider SQL Server
- then use lookup for adding some columns from other tables that work as left join
- using derived columns for replacing nulls
- using SCD and make for each column the specific Type for it (Type0: Fixed attribute, Type1: changing attribute and Type2: historical attribute)
- then loaded the data into the dim_product table as destination DWH.

2. Dim Customer:

Same steps as Dim Product and load the data into the dim_customer table as destination DWH.

3. Dim Date:

We create an excel file with python code to get the date from 1/1/2000 to 12/31/2030

then use data conversion to convert each column from source with the same datatype as destination

then load our data into dim_date table as destination DWH.

4. Dim Territory:

We first created a table called Lookup Country: To put in it every abbreviation with the full name of the country

Then we selected the data source from the database and used a lookup in SSIS project to merge the territory table with Lookup Country table to use only the full name of the country not the abbreviation and then loaded it into dim_territory table as destination DWH.

5. Fact Sales:

- **Full Load:**
  First, we get the data from two data sources: SalesOrderHeader and SalesOrderDetail table
  Then we make a merge join and should all the data source be order by the same Column so I order it with SalesOrderID column in the SQL command
  Note:
  1. I should also use the show advanced editor and go to input and output properties then OLE DB Source Output and make the is Sorted to be true and click on the output columns and SalesOrderID to make the SortKeyPosition to be 1 for two data sources
  2. If I didn't use orderby SalesOrderID in the SQL command, then I should use the sort from the ssis toolbox

  We used Lookup join to get the PK for every Dimension

  then use the Derived column to replace every null for the PK with 0 and this 0 refer to the unknown row that I created or inserted in each dimension this null may occur because I use the ignore failure in the lookup join

  Then use Derived column to get the extended_sales and extened_cost by multiplying the quantity by cost and unit price and then load all this data into the Fact Sales table as the destination DWH.

  Note: in the full load if I run the package again without truncate the data will give me error because I put duplicated data in the primary key so to solve this problem we use Execute SQL Task and connect with destination connection manager and put the SQL command (truncate table fact_sales) before the Data Flow to remove the data before run the package and it will work.

- **Incremental Load:**
  The idea of the load is that I choose a specific column (ModifiedDate) from the source to build the entire package about it and put it in new table called meta_control_table to compare this column with the last load data to make the incremental load.

  First, we will create a new table called meta_control_table to put in it the name of the fact table and the specific column that I will compare the data with.

We will copy the Full Load package and paste it to an incremental load but will make some modifications to it.

We truncate the table and remove the relation between the Truncate Fact Sales and Data Flow and then make it disable.

We use a new Execute SQL Task to get the latest load data:

- We wrote the SQL command to get the last_load_date from the meta_control_table and the result of this query will be stored in a single row.
- We made the ResulSet to be single row and open the Result Set and make a new variable called last_load_date.

We connect a new Execute SQL command with the Data Flow and modify it:
we will open every data source and modify the SQL command to compare the ModifiedDate column with 2 parameters

- The first parameter is the variable that we created last_load_date
- The second parameter is the start time from the system

So, this will take the initial value that I create for the last_load_date that's '1900-01-01' but after this I want to get the new value to make incremental load

So, I will make Execute SQL Task and update the last_load_date column and make a parameter to reference to the start time or the last time that I run the package.

This will update the meta_control_table to update the last_load_date to get the last time that is run the package and when I run the package again it will get the last time, I run it before so it will make an incremental load not full load.

So, if I insert some data in the data source and run the package again it will only run the specific rows that I inserted.

Note: why I choose the value 31/12/2099 to the last_load_date because I want to make sure that there is no problem will happen in the future.
for example : there is any problem in the SQL Get Last Load Date and the last_load_date instead to take the time I run the package, it takes the default value for it (31/12/2099) so when the condition that I created about the ModifiedDate when it is (modifieddate >= last_load_date and modifieddate < start_time),it will give me an empty package or will not add new rows so I understand from this that there is a problem in the package and search to solve it.