

# DWBI- Practical 1

## 1. Introduction to the Worksheet

In this practical session, you will be given a hands-on understanding on how-to setup the environment to continue with the data warehouse implementation.

Standard MSDN SSIS Tutorials are available at:

<https://docs.microsoft.com/en-us/sql/integration-services/ssis-how-to-create-an-etl-package>

Standard MSDN tutorials are based on “Adventure Works 2012” database and data warehouse. They can be downloaded in the below link:

<http://msftdbprodsamples.codeplex.com/releases/view/55330>

How to attach above downloaded data files into SQL Server instance via SQL Server Management Studio IDE is explained in the following links:

<https://www.mssqltips.com/sqlservertip/2638/attach-sample-database--adventureworks-in-sql-server2012/>

[https://www.youtube.com/watch?v=x\\_JDwpgm9Po](https://www.youtube.com/watch?v=x_JDwpgm9Po)

**Try out MSDN tutorials on your own upon completion of these practical sessions on Data Warehouse Creation and ETL tasks with SSIS.**

To continue with this tutorial, let's now setup the environment.

## 2. Setting-up the Environment

Following tools are required to perform the tasks ahead.

- SQL Server Management Studio
- SQL Server Data Tools
- SQL Server 2016 or higher
- Microsoft Office (Excel/ Word)

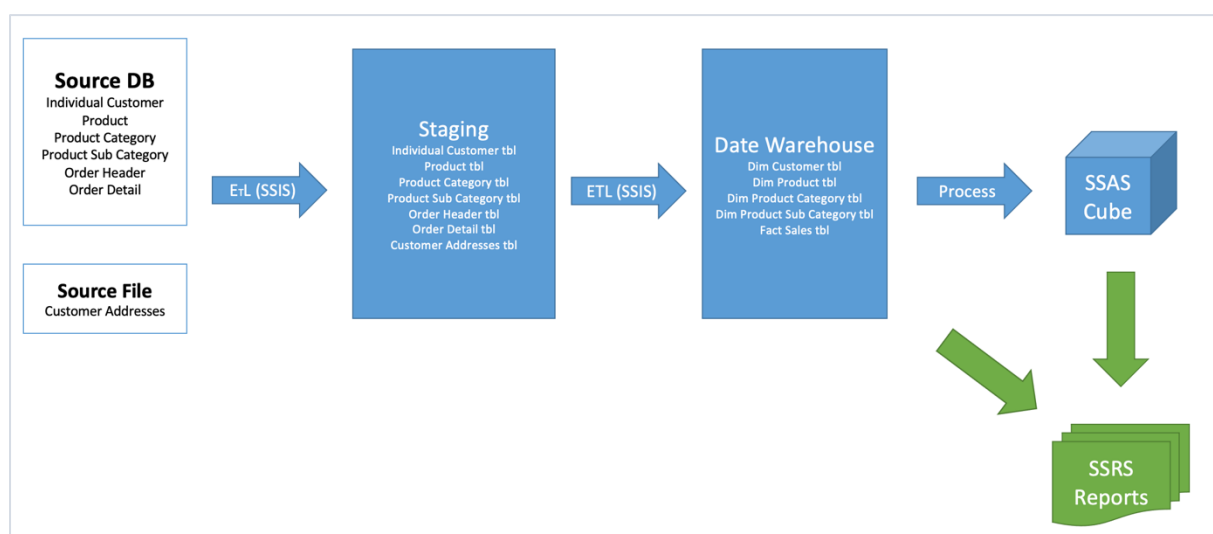
Below database backups are provided to you. Restore them before starting the tasks in the worksheet.

- SLIIT\_RetailSourceDB.bak -> This database backup contains the source system data.
- CustomerAddress.txt -> This file contains all customer address information.

### 3. Scenario

This data is about ABC Retail Store. They sell goods at different locations. However, due to their business requirements, their customer address information is stored in a separate text file. These addresses are updated when customers inform about their location changes.

A set of new requirements have come up to analyse the performance of their business from different perspectives. Due to these management requirements, it was decided to create a data warehouse & an analytical platform to analyse/measure their performance.



### 4. Introduction to the Environment

As the above diagram depicts, main data sources are relational tables in a source database and a flat file. As the first step, data needs to be extracted from the source systems (database and flat file) and loaded in the '**Staging**' area, which is an intermediate storage between source systems and target, data warehouse. Data model/table structures of the '**Staging**' database is very much closer the structure of the sources. Main difference is all the source tables/files are available in a single database server. Next step is to extract data from '**Staging**'

database tables, transform data and load them into corresponding tables in dimensional model created in the data warehouse. This step will contain considerable transformation as the data warehouse tables are based on a dimensional model.

## 5. Setting-up the Data Sources

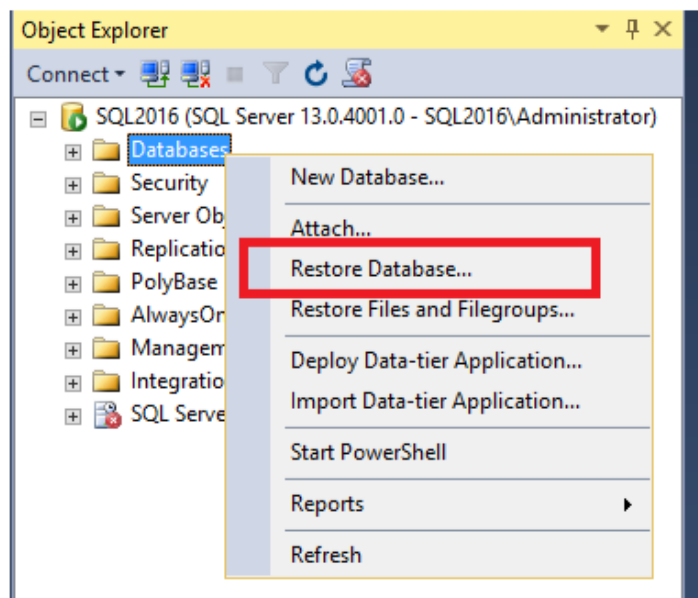
For this tutorial, the source data is a set of historical retail data contained in the database, '**SLIIT\_RetailSourceDB**', and customer address data saved in '**CustomerAddress.txt**' file.

### Setting-up source database

A backup of the '**SLIIT\_RetailSourceDB**' database is provided: '**SLIIT\_RetailSourceDB.bak**' file.

In order to restore the backup file:

- Connect to the SQL Server instance.
- Right click **Databases**.
- Select **Restore Databases**.
- Then follow the steps in the wizard.



### Setting-up source text file

Now, create a new folder in your C drive named '**SLIIT Data Warehouse Solution Tutorial**'.

Copy the '**CustomerAddress.txt**' file to the newly created folder. This will be treated as the source location for the addresses file of the customers.

## 6. Verify the Environment

1. Ensure you can connect to the SQL Server instance.
2. Ensure the restored database is available.
3. Ensure the '**CustomerAddress.txt**' text file is copied to the folder.
4. In SQL Server Management Studio, verify whether the below tables are available in the restored database.
  - IndividualCustomer
  - Product
  - ProductCategory
  - ProductSubCategory
  - SalesOrderDetail
  - SalesOrderHeader
5. Ensure the tables have data using the below query.

```
use SLIIT_RetailSourceDB;  
select * from dbo.IndividualCustomer;
```

6. Change the table names in the query to view data of other tables.

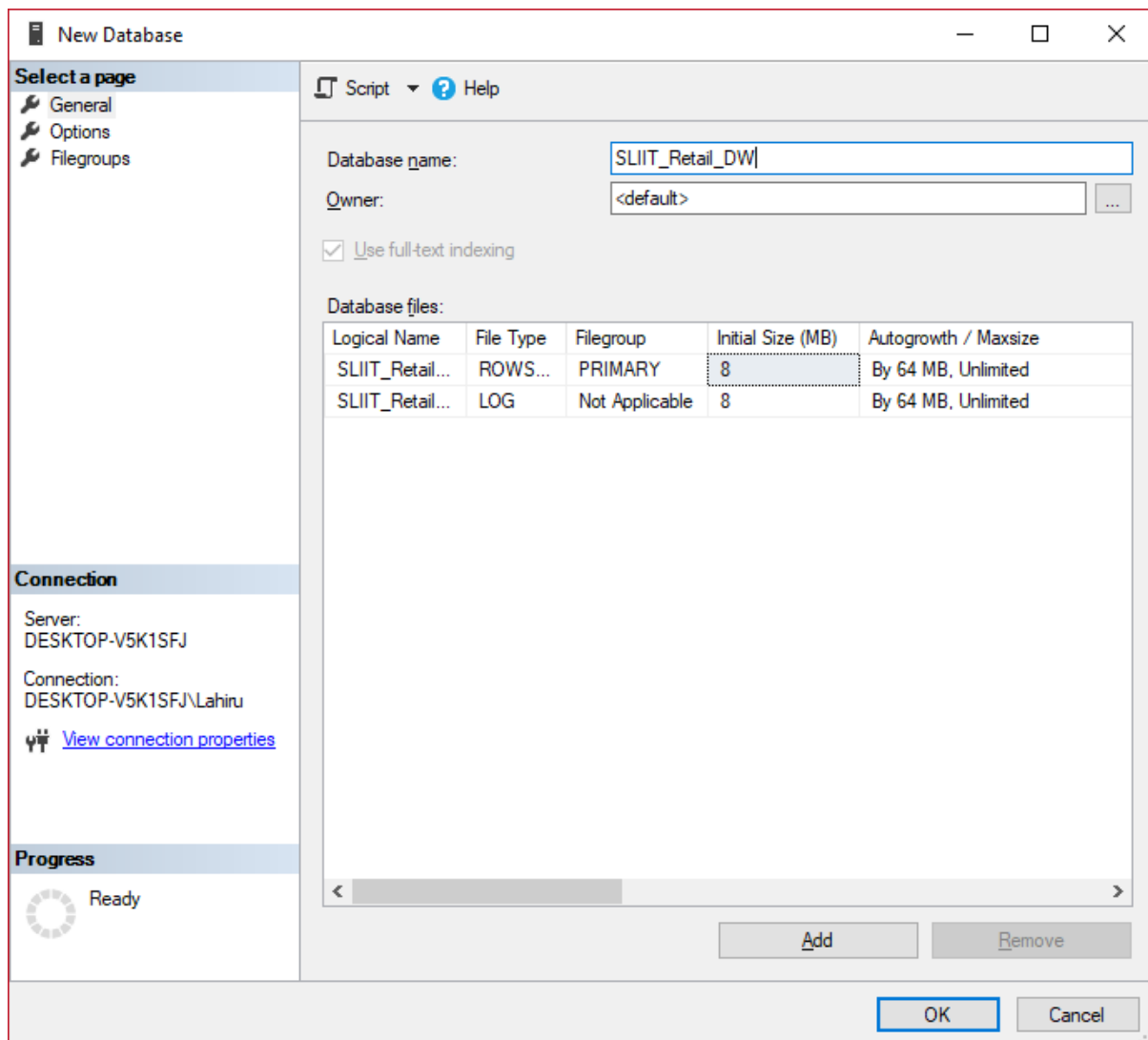
## 7. Creating Staging and Data Warehouse Databases

Now, create the intermediate destination database of the first step, by following below steps:

- Right click on **Databases**.
- Select **New Database**.
- Provide the database name as '**SLIIT\_Retail\_Staging**'.
- Click **OK** to complete the database creation.

Follow same steps to create the database for data warehouse.

- Right click on **Databases**.
- Select **New Database**.
- Provide the database name as '**SLIIT\_Retail\_DW**'.
- Click **OK** to complete the database creation.



Note that we did not create any tables in the '**Staging**' database or in the data warehouse but only the database.

## 8. Understanding the Documentation

In order to understand the source dataset, refer the excel file named '**Data Descriptions**'.

In the excel file:

- Tab named '**Source Details**' describes the dataset from table level.
- Tab named '**Source Table Details**' describes each and every column of the dataset.

## 9. Creating Mapping Documents

Before going ahead with the data warehouse design (essentially to identify the set of tables required), it is required to understand the business requirements and determine the dimensional model for the data warehouse.

Based on the dataset:

- Identify the dimension tables.
- Identify the fact table(s).
  - Identify the derived measures.

[Read more on different ways of creating mapping documents](#)

**Based on your understanding, Prepare the mapping sheet to support the dimensions & facts you identified. Update the 'ETL Mapping Document' tab of the 'Data Descriptions' Excel file.**

**Note:** These should ultimately meet the end user requirements.

## 10. Creating Data Warehouse Tables

Before creating the fact & dimension tables, start by creating the Date Dimension.

Use the code in the file '**DateMaster.sql**' file provided to you as a resource.

- Open the file in **SSMS**.
- Change the database to '**SLIIT\_Retail\_DW**'.
- Paste the '**DateMaster.sql**' code in query window.
- Click **Execute** to generate the date dimension.

Based on the excel mapping document, create identified dimension and fact tables in the data warehouse by following below steps:

- Open SQL Server Management Studio & connect to the server.
- Expand **Databases**.
- Expand '**SLIIT\_Retail\_DW**' database.
- Right click on **Tables** → select **New** → **New Table**.
- In the table designer view, provide the column names & data types.
- One example is given below for the '**Customer**' dimension.
- For each column, simply describe what the column is under **Description** property.

DESKTOP-V5K1SFJ.S...I\_DW - dbo.Table\_1\* X

Column Name	Data Type	Allow Nulls
CustomerID	int	<input type="checkbox"/>
AlternateCustomerID	int	<input checked="" type="checkbox"/>
Title	varchar(8)	<input type="checkbox"/>
FirstName	varchar(50)	<input checked="" type="checkbox"/>
MiddleName	varchar(50)	<input type="checkbox"/>
LastName	varchar(50)	<input checked="" type="checkbox"/>
Gender	varchar(6)	<input type="checkbox"/>
PhoneNumber	varchar(25)	<input checked="" type="checkbox"/>
EmailAddress	varchar(50)	<input checked="" type="checkbox"/>
EmailPromotion	int	<input checked="" type="checkbox"/>
AddressType	varchar(50)	<input checked="" type="checkbox"/>
AddressLine1	varchar(50)	<input checked="" type="checkbox"/>
AddressLine2	varchar(50)	<input checked="" type="checkbox"/>

Column Properties

(Is Identity)	Yes
Identity Increment	1
Identity Seed	1
Indexable	Yes
Is Columnset	No
Is Sparse	No

Identity Specification

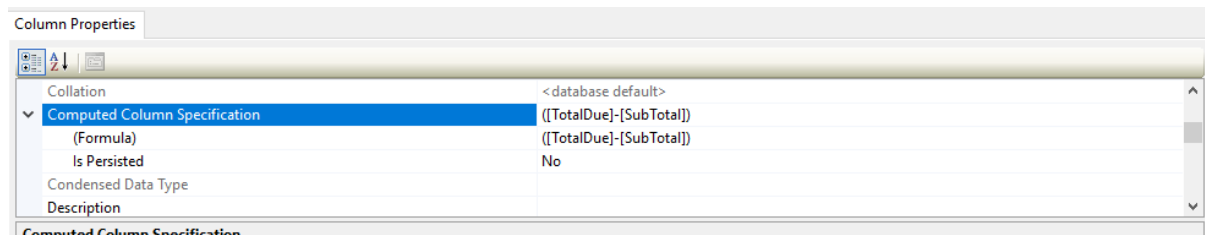
Once done, press **CTRL + S** to save the table. In the popup, provide the table name.

For dimensions, name the table using '**Dim<Table Name>**' format.

For Facts, name the table using '**Fact<Table Name>**' format.

When designing the main fact table, follow the below process in adding the computed fields.

1. Add the column in the table designer.
2. Under **Column Properties** of the computed field, add the formula provided in the '**Data Descriptions**' excel file to the **(Formula)** property that resides under **Computed Column Specification** section.



By now you have setup the environment & have created the data warehouse tables. In the next tutorial, we will use SSIS to develop a ETL process to load the data to the staging database and from there to the data warehouse tables we designed.

**Note:** We did not create 'Staging' database tables. We will create staging database tables when creating the SSIS package the load data from source tables and flat file to staging database as required.

---