1. Data set Description

Online real-time restaurant-reservation services

The selected dataset contains restaurant reservation records from **EZTABLE**, a Taiwan-based online booking platform, covering the **years 2012 to 2014**. It includes information about member bookings, restaurant details, and member profiles. Each record in the training set represents a member's first booking. The data is split across multiple files, including Bookings.csv (bookings), Member.csv (member details), and Restaurant.txt (restaurant attributes), enabling the use of **multiple data sources**. The dataset provides sufficient data to construct dimensional models, build ETL pipelines, and perform business intelligence analysis.

1.Bookings (.csv) -Contains data related to each booking done through the website.

- ➔ booking_id (varchar), member_id (varchar), cdate (datetime), restaurant_id (varchar), datetime (datetime), people (varchar), purpose (varchar), gender (varchar), status varchar, is_required_prepay_satisfied (varchar),

2.Members (.csv) -Details about each member who has made a booking.

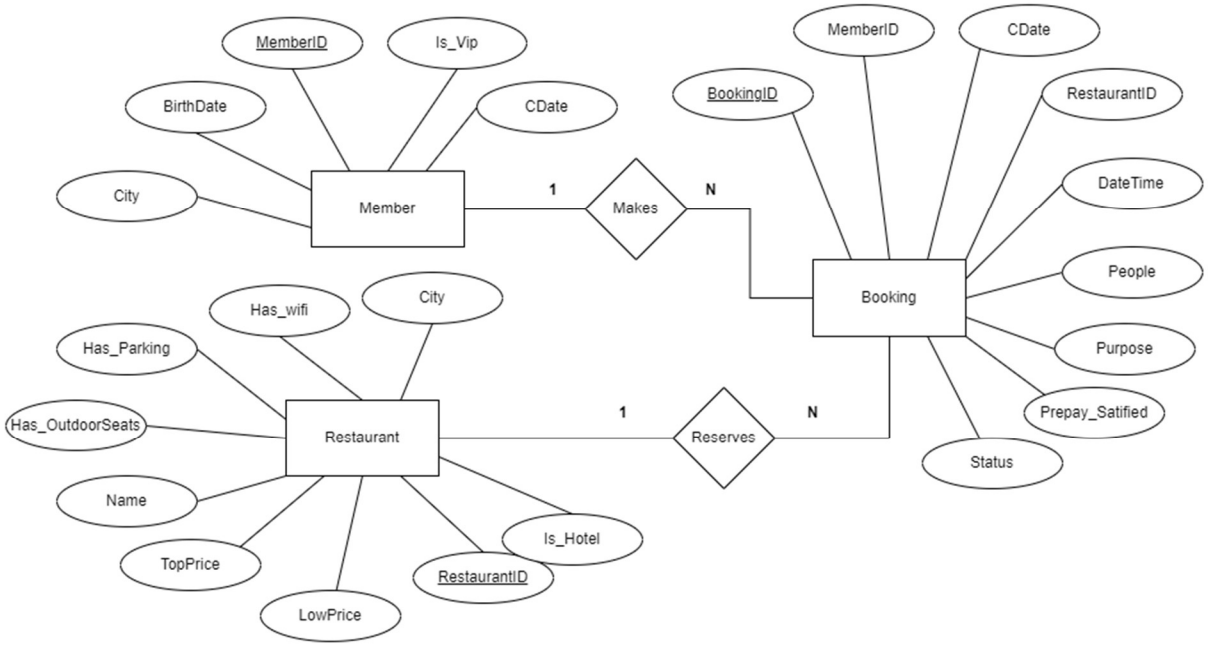
- ➔ id (varchar), is_vip (varchar), gender (varchar), birthdate (date), city (varchar), cdate (datetime)

3.Restaurants (.txt) -This includes all the restaurants in which the site can make bookings to.

- ➔ id(varchar), is_hotel(varchar), city(varchar), cityarea(varchar), name(varchar), abbr(varchar), tel(varchar), parking(int), outdoor_seating(int), wheelchair (int), price1(float), price2(float), lat(int), lng(int),, cdate(varchar)

Data Set Selected - [Online real-time restaurant-reservation services]

ER Diagram



2. Preparation of Data Sources

The dataset has been prepared using three types of data sources to simulate a realistic ETL environment.

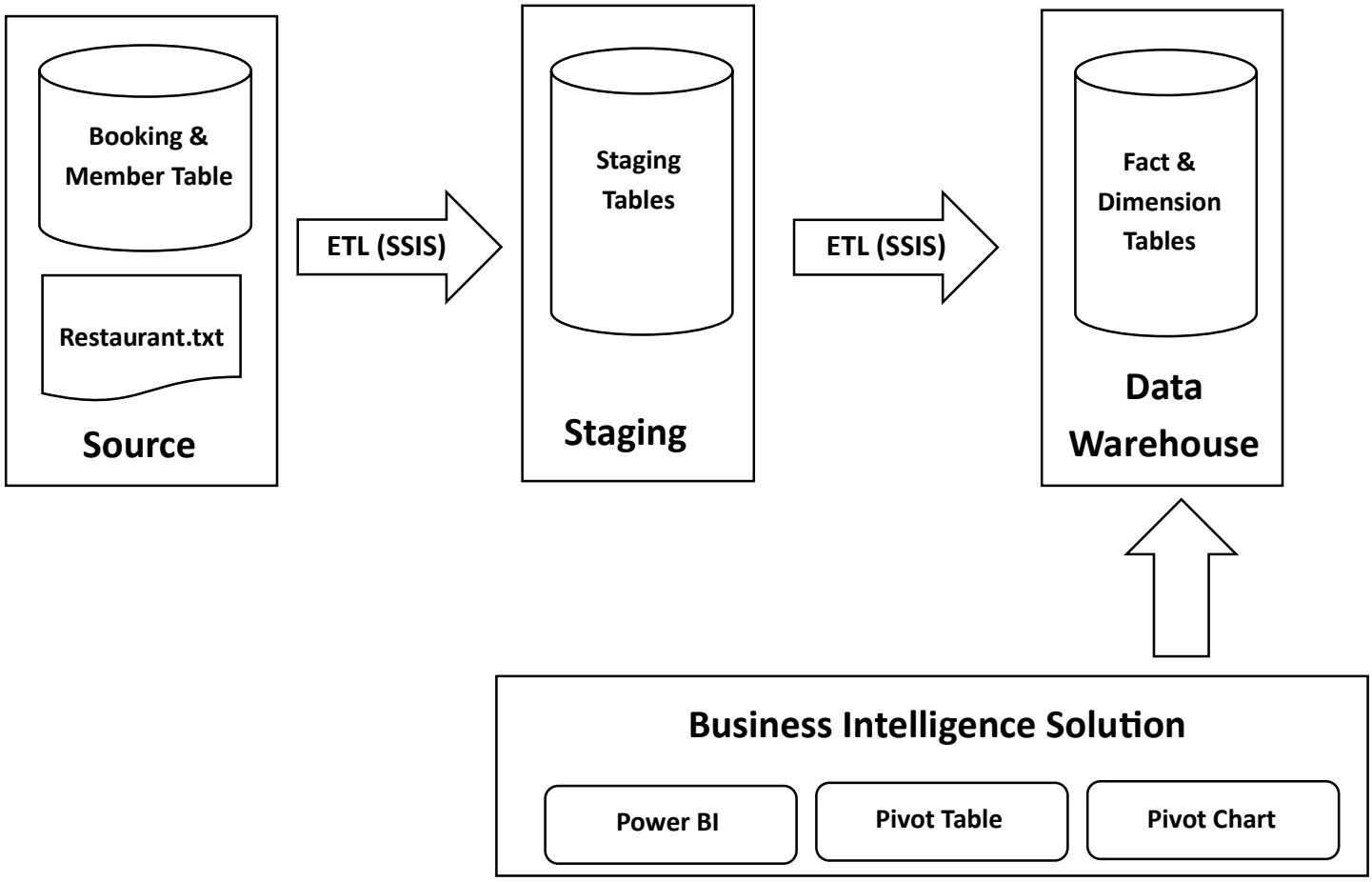
The main booking data from **Booking.csv** was imported into a SQL Server table for structured storage and querying and then forwarded to staging.

The member information from **Member.csv** was also imported into the SQL server and was chosen as the slowly changing dimension and also forwarded to staging.

The restaurant data from **Restaurant.txt** was kept introducing a different data source format and bought straight to the staging through the Staging process

This separation allows for a richer ETL process, involving multiple formats and transformations. Each source contains unique keys (member_id, restaurant_id,booking_id) enabling the integration of data across sources. These formats support various extraction methods within SSIS.

3. Architectural Design



1.Source - For the given dataset, two of the data sources are first loaded into the SQL server to observe the data. The other file for restaurant is directly taken as a flat file. The source database name is DWBI_Restaurant_Source.

2.Staging - Acts as a temporary storage facility to make sure all data sources required for the warehouse are bought. The staging database name is DWBI_Restaurant_Staging.

3.Warehouse - A data warehouse is a large collection of business data that is used to enhance internal decision-making. In this scenario the database DWBI_Restaurant_DW is used as the data warehouse. The said data warehouse comprises of 4 dimensional tables and 1 fact table

4.BI Solution - Integrates Power BI and Excel to analyze data from deployed SSAS Cube made of the Data Warehouse tables. Power BI provides interactive reports and dashboards, while Excel enables detailed ad-hoc analysis using pivot tables and charts. These reports ensure real-time insights and easy access for decision-making

5.ETL - Extraction, transformation and Loading of data is done through an Visual studio integration project(SSIS), by connecting all above mentioned databases and files sources.

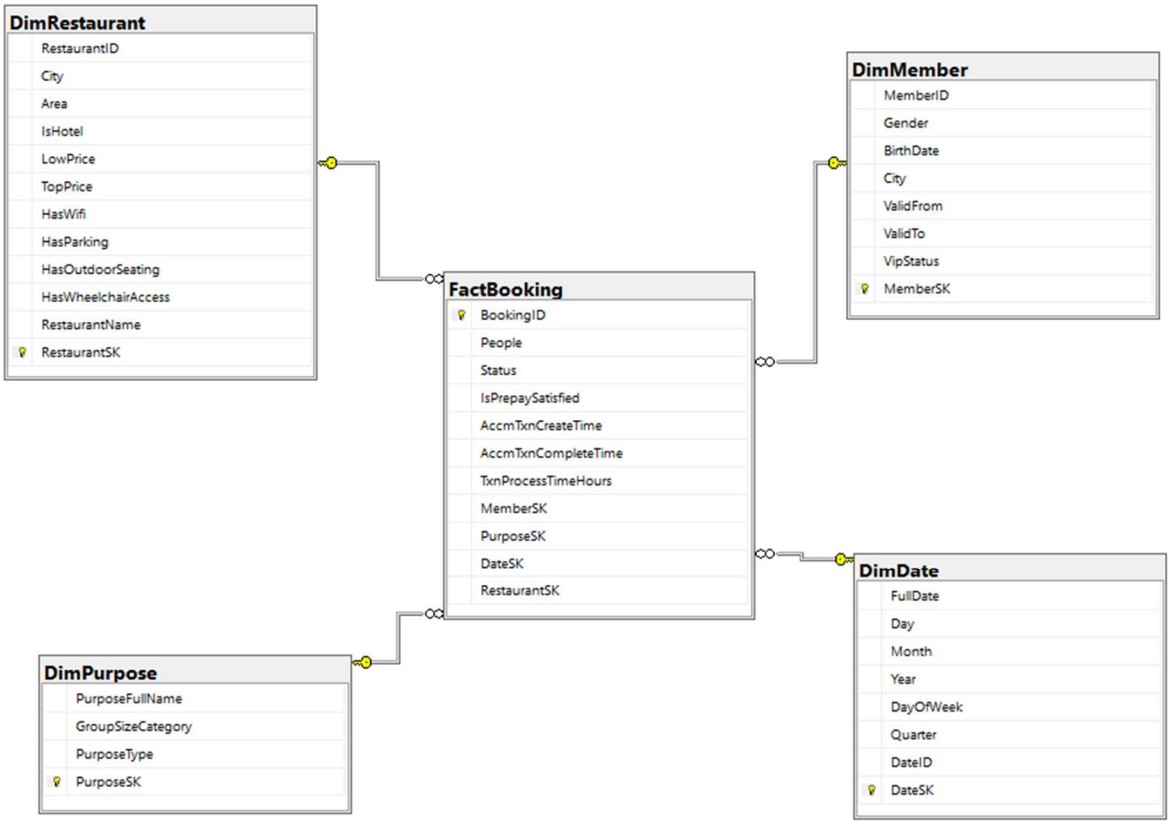
4. Data Warehouse Design

The data warehouse follows a **Star Schema** structure with the central **FactBooking** table storing detailed booking transactions. It connects to four dimensions: **DimMember, DimRestaurant, DimDate**, and the newly added **DimPurpose**. DimMember is designed as a **Slowly Changing Dimension** (Type 2) to track changes in member demographics. DimRestaurant captures restaurant location and feature attributes. DimDate enables flexible time-based analysis, and DimPurpose stores the reason for the booking along with a new derived category called GroupSizeCategory based on the number of people in the party. This model supports in-depth business intelligence reporting, such as analyzing large group bookings for birthdays, or tracking member behavior across cities and time.

Assumptions Made:

1. MemberID, RestaurantID, and BookingID are stored as VARCHAR to match anonymized IDs.
2. DimMember assumes Type 2 SCD tracking using ValidFrom, ValidTo, and IsCurrent.
3. HasWifi, HasParking, etc., are binary features stored as BIT.
4. DateID in DimDate follows a YYYYMMDD HHMMSS000 format.
5. Only successful bookings (excluding test/invalid ones) are loaded into FactBooking.

Relational Diagram Design



- **Schema Used – Star Schema**

  A Star schema simplifies queries, improves performance, and makes reporting faster by structuring data into **one central fact table** connected to **multiple dimension tables**.

- **Hierarchies used – For Date and Restaurant**

  The date is broken down into hierarchies starting from day to month and then the year. Since the same restaurant exists in different cities there exists a hierarchy from the restaurant name to the city to the Area.

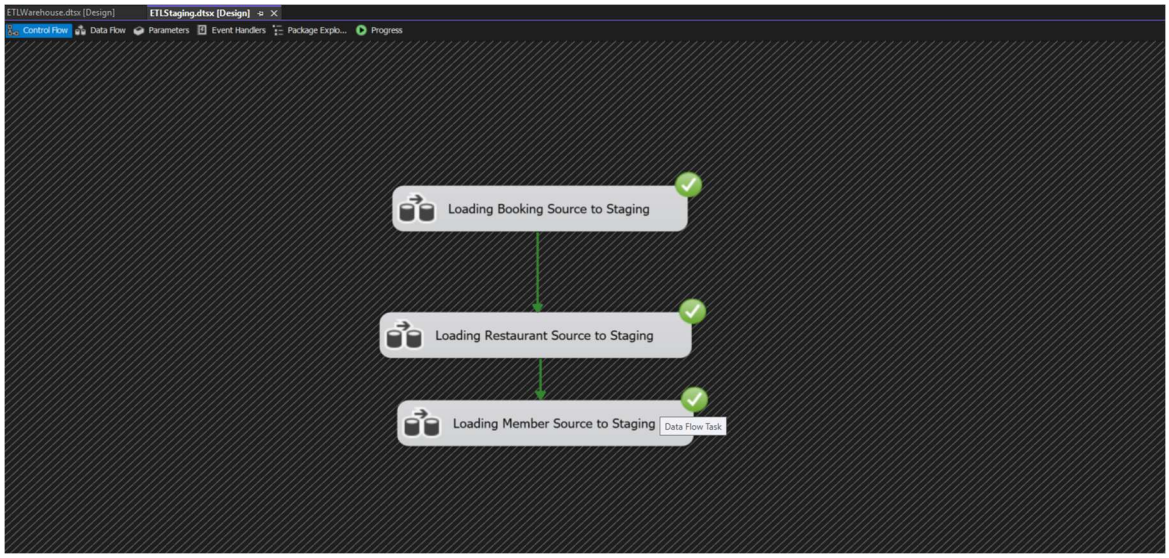
- **Slowly Changing Dimension - Member**

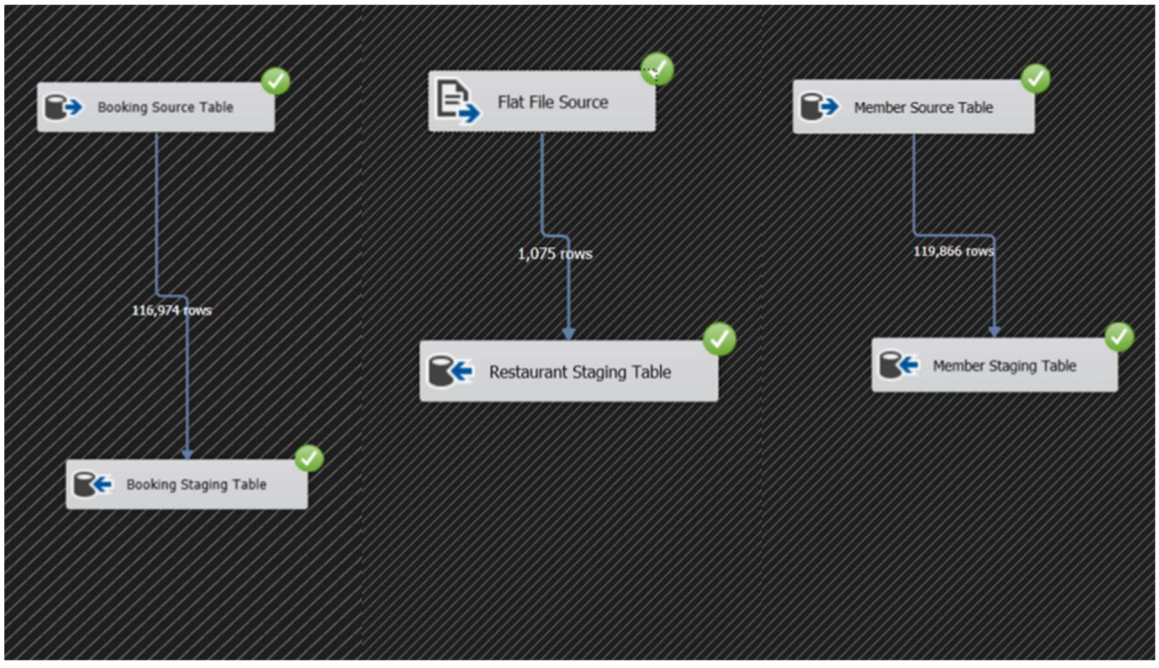
  The Member dimension was taken as the slowly changing dimension due to values such as City as well as VIP status which can change over time. We chose Type 2 SCD, so that a new record is created if a change is made to one of those attributes.

5 ETL Development

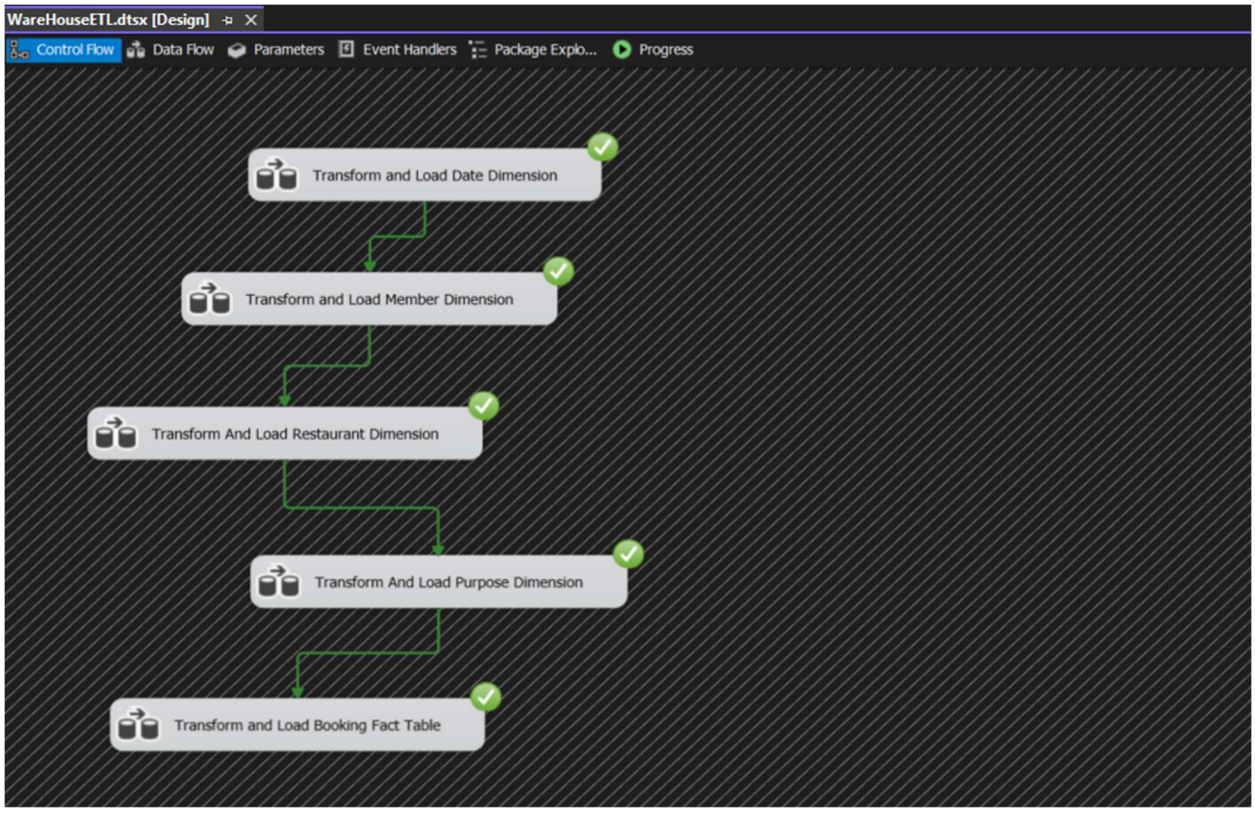
5.1 Extract Data from files to Source -Select the flat files and import the to the Source database and then create a Staging Database as well.

5.2 Extract Data from Source to Staging tables

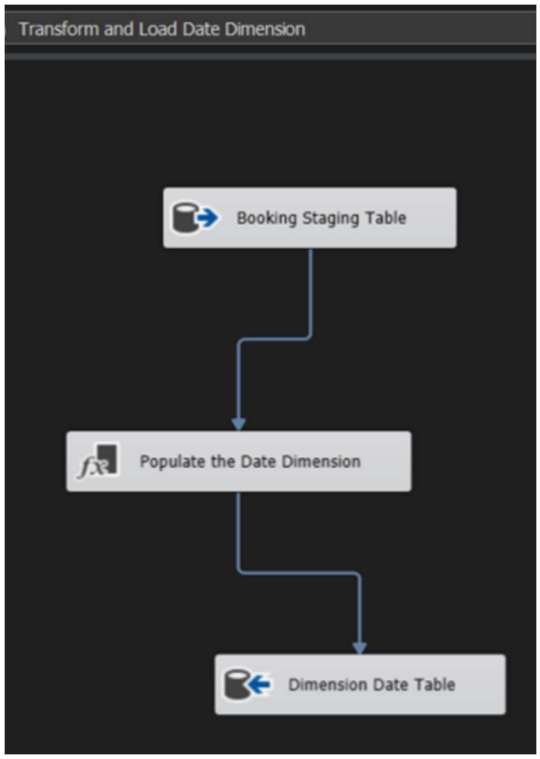




5.3 Extract Data from Staging to Warehouse



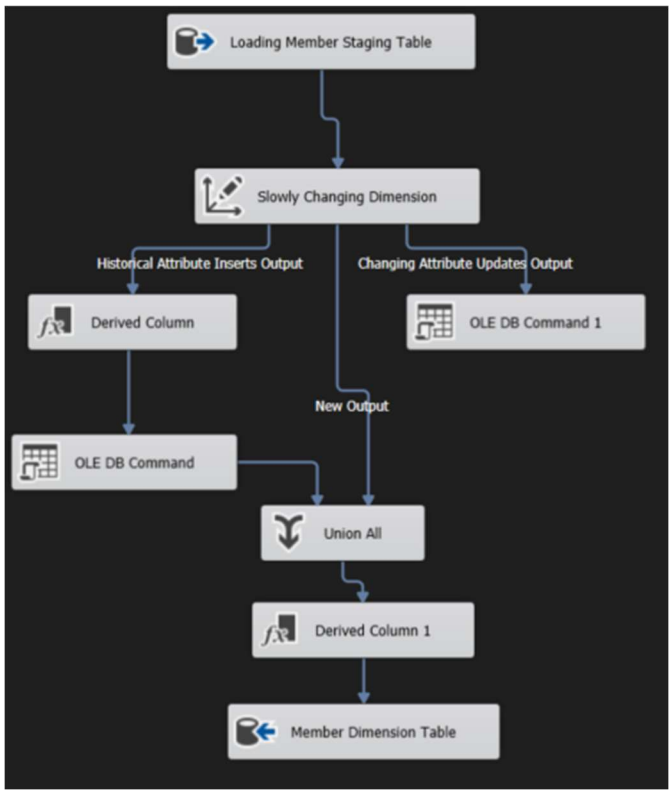
5.3.1 Creating a Date Dimension Table and Loading it




```
CREATE TABLE [dbo].[DimDate] (
    [FullDate] DATETIME NULL,
    [Day] TINYINT NULL,
    [Month] TINYINT NULL,
    [Year] SMALLINT NULL,
    [DayOfWeek] VARCHAR(10) NULL,
    [Quarter] TINYINT NULL,
    [DateID] DATETIME NOT NULL,
    [DateSK] INT IDENTITY(1,1) NOT NULL,
    CONSTRAINT [PK_DimDate_DateSK] PRIMARY KEY
    CLUSTERED ([DateSK] ASC)
) ON [PRIMARY];
GO
```

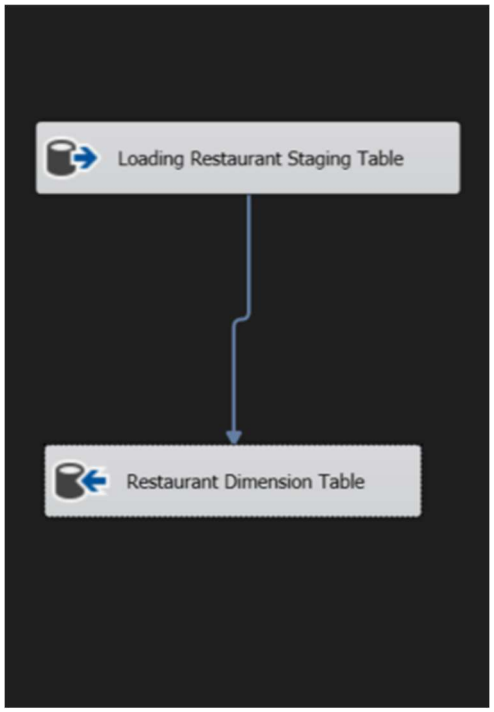
5.3.2 Creating a Member Dimension Table and Loading it

(Chosen as the Slowly Changing Dimension)



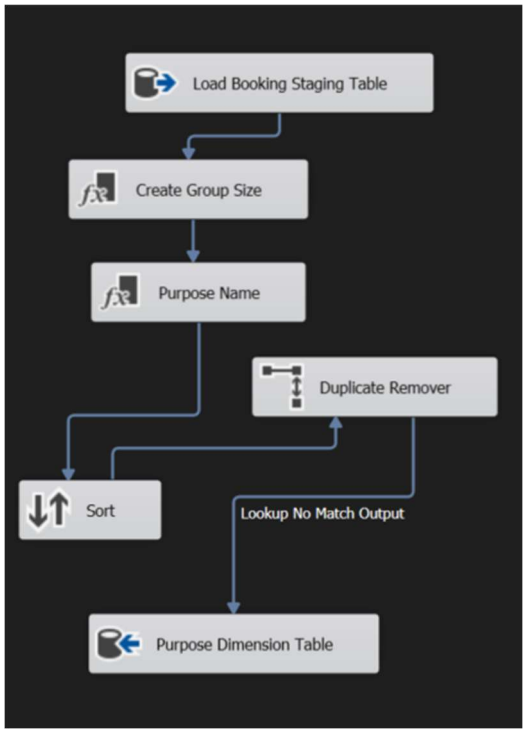

```sql
CREATE TABLE [dbo].[DimMember] (
    [MemberID] VARCHAR(50) NOT NULL,
    [Gender] VARCHAR(50) NULL,
    [BirthDate] VARCHAR(50) NULL,
    [City] VARCHAR(100) NULL,
    [ValidFrom] DATETIME NULL,
    [ValidTo] DATETIME NULL,
    [VipStatus] VARCHAR(50) NULL,
    [MemberSK] INT IDENTITY(1,1) NOT NULL,
    CONSTRAINT [PK_DimMember_MemberSK] PRIMARY KEY
    CLUSTERED ([MemberSK] ASC)
) ON [PRIMARY];
GO
```

5.3.3 Creating a Restaurant Dimension Table and Loading it




```sql
CREATE TABLE [dbo].[DimRestaurant] (
    [RestaurantID] VARCHAR(50) NOT NULL,
    [City] VARCHAR(100) NULL,
    [Area] VARCHAR(100) NULL,
    [IsHotel] BIT NULL,
    [LowPrice] INT NULL,
    [TopPrice] INT NULL,
    [HasWifi] BIT NULL,
    [HasParking] BIT NULL,
    [HasOutdoorSeating] BIT NULL,
    [HasWheelchairAccess] BIT NULL,
    [RestaurantName] VARCHAR(50) NULL,
    [RestaurantSK] INT IDENTITY(1,1) NOT NULL,
    CONSTRAINT [PK_DimRestaurant] PRIMARY KEY
    CLUSTERED ([RestaurantSK] ASC)
) ON [PRIMARY];
GO
```

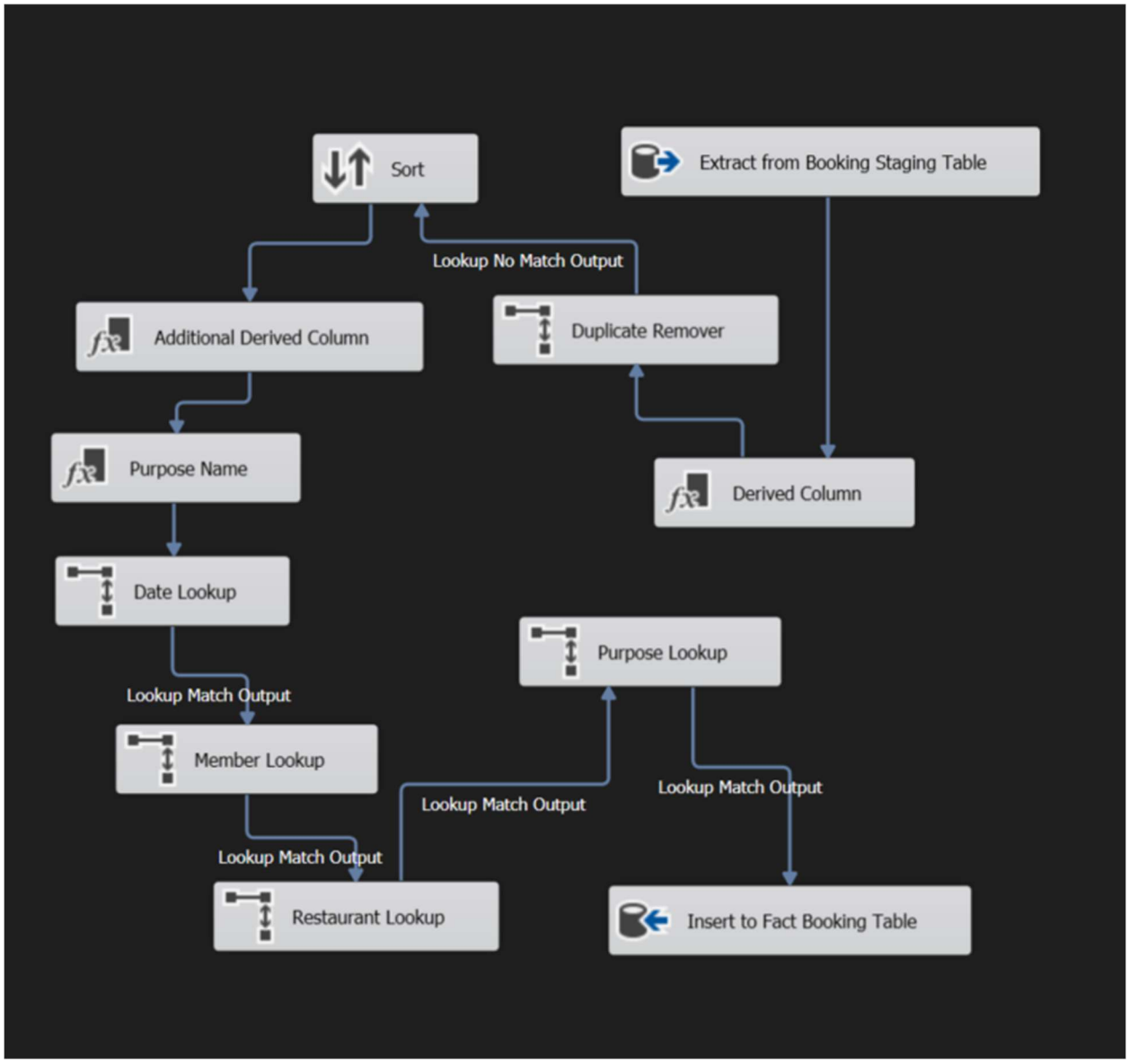
5.3.4 Creating a Purpose Dimension Table and Loading it



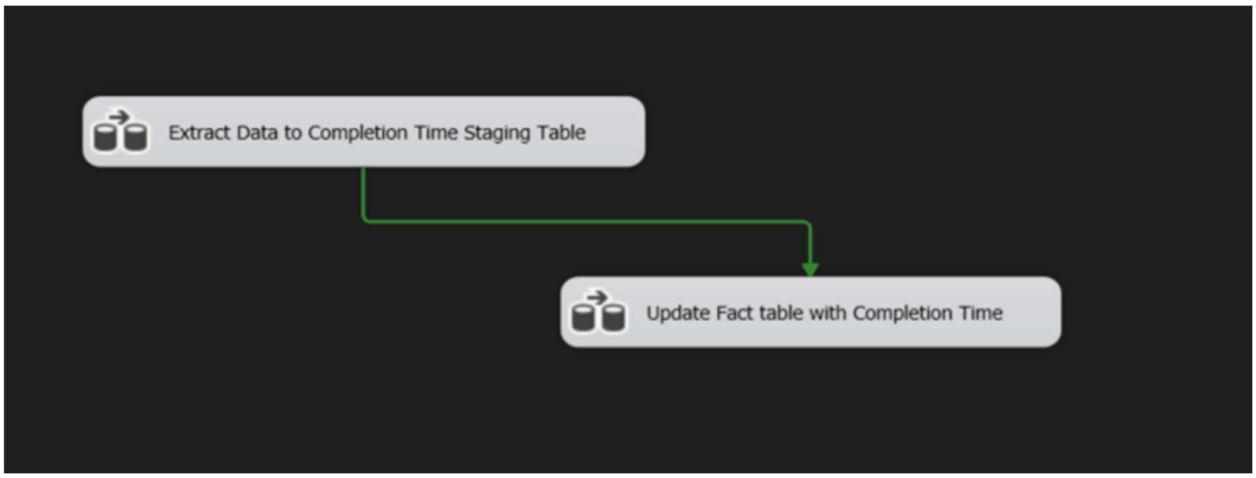

```
CREATE TABLE [dbo].[DimPurpose] (
    [PurposeFullName] VARCHAR(50) NULL,
    [GroupSizeCategory] VARCHAR(50) NULL,
    [PurposeType] VARCHAR(50) NOT NULL,
    [PurposeSK] INT IDENTITY(1,1) NOT NULL,
    CONSTRAINT [PK_DimPurpose_PurposeSK] PRIMARY KEY
    CLUSTERED ([PurposeSK] ASC)
) ON [PRIMARY];
GO
```

5.4 Creating a Fact Booking Table and Loading it by looking up the Id on each dimension and then referencing the equivalent SK on each table

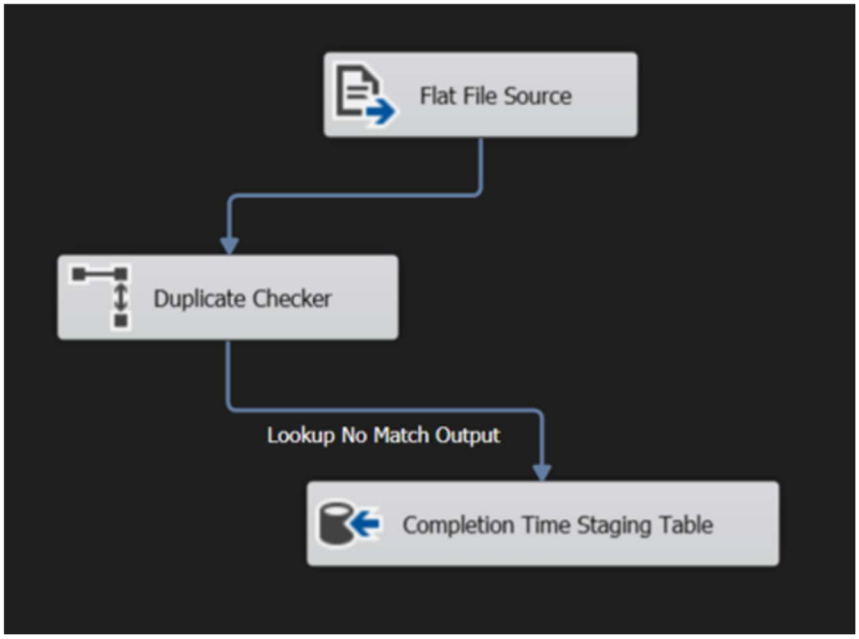
```
CREATE TABLE [dbo].[FactBooking] (
    [BookingID] INT NOT NULL,
    [People] TINYINT NULL,
    [Status] VARCHAR(50) NULL,
    [IsPrepaySatisfied] BIT NULL,
    [AccmTxnCreateTime] DATETIME NULL ,
    [AccmTxnCompleteTime] DATETIME NULL,
    [TxnProcessTimeHours] INT NULL,
    [MemberSK] INT NULL,
    [PurposeSK] INT NULL,
    [DateSK] INT NULL,
    [RestaurantSK] INT NULL,
    CONSTRAINT [PK_FactBooking] PRIMARY KEY CLUSTERED ([BookingID] ASC)
) ON [PRIMARY];
GO

ALTER TABLE [dbo].[FactBooking]
    WITH CHECK ADD
    CONSTRAINT [FK_FactBooking_DimDateSK] FOREIGN KEY ([DateSK]) REFERENCES [dbo].[DimDate] ([DateSK]),
    CONSTRAINT [FK_FactBooking_DimMember] FOREIGN KEY ([MemberSK]) REFERENCES [dbo].[DimMember] ([MemberSK]),
    CONSTRAINT [FK_FactBooking_DimRestaurant] FOREIGN KEY ([RestaurantSK]) REFERENCES [dbo].[DimRestaurant] ([RestaurantSK]),
    CONSTRAINT [FK_FactBooking_PurposeSK] FOREIGN KEY ([PurposeSK]) REFERENCES [dbo].[DimPurpose] ([PurposeSK]);
GO

ALTER TABLE [dbo].[FactBooking] CHECK CONSTRAINT [FK_FactBooking_DimDateSK];
ALTER TABLE [dbo].[FactBooking] CHECK CONSTRAINT [FK_FactBooking_DimMember];
ALTER TABLE [dbo].[FactBooking] CHECK CONSTRAINT [FK_FactBooking_DimRestaurant];
ALTER TABLE [dbo].[FactBooking] CHECK CONSTRAINT [FK_FactBooking_PurposeSK];
GO
```



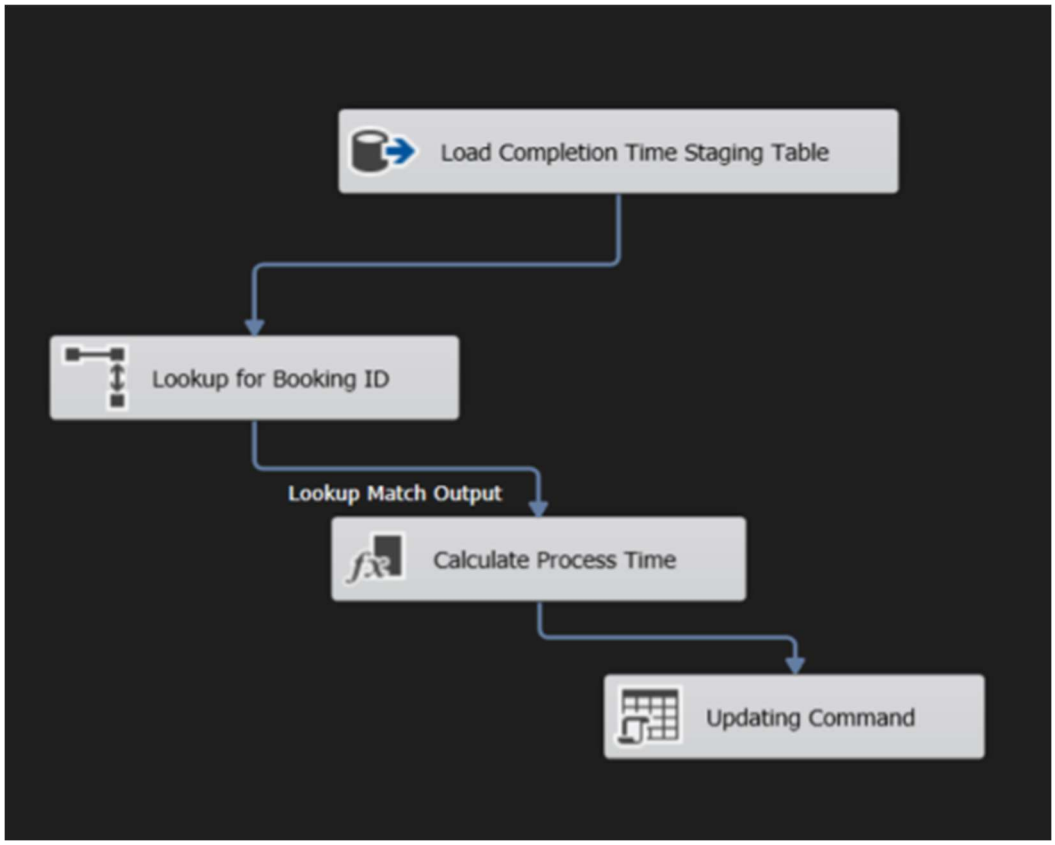
5.5 For the completion-time table, as csv file was created and imported to the staging tables. This was referenced to update Completing Time and Processing time in hours in the Fact Table.



5.5.1 Extracting from csv Flat file to Staging Database



5.5.2 Referencing the Booking ID and Updating the Complete-time and Process-time-hours Columns.



6. The **ETL process is complete**.

6.1. Fact Booking Table

```
SELECT TOP (1000) [FullDate]
      ,[Day]
      ,[Month]
      ,[Year]
      ,[DayOfWeek]
      ,[Quarter]
      ,[DateID]
      ,[DateSK]
  FROM [DWBI_Restaurant_DW].[dbo].[DimDate]
```

100 %

Results | Messages

	FullDate	Day	Month	Year	DayOfWeek	Quarter	DateID	DateSK
1	2013-06-12 19:00:00.000	12	6	2013	4	2	2013-06-12 19:00:00.000	1
2	2014-06-09 11:30:00.000	9	6	2014	2	2	2014-06-09 11:30:00.000	2
3	2013-10-17 17:30:00.000	17	10	2013	5	4	2013-10-17 17:30:00.000	3
4	2013-09-15 19:30:00.000	15	9	2013	1	3	2013-09-15 19:30:00.000	4
5	2013-06-16 12:30:00.000	16	6	2013	1	2	2013-06-16 12:30:00.000	5
6	2013-02-20 14:30:00.000	20	2	2013	4	1	2013-02-20 14:30:00.000	6
7	2013-07-31 14:00:00.000	31	7	2013	4	3	2013-07-31 14:00:00.000	7
8	2012-01-26 12:00:00.000	26	1	2012	5	1	2012-01-26 12:00:00.000	8
9	2014-08-20 13:30:00.000	20	8	2014	4	3	2014-08-20 13:30:00.000	9
10	2013-01-14 11:00:00.000	14	1	2013	2	1	2013-01-14 11:00:00.000	10
11	2013-11-17 20:00:00.000	17	11	2013	1	4	2013-11-17 20:00:00.000	11
12	2013-05-18 18:00:00.000	18	5	2013	7	2	2013-05-18 18:00:00.000	12
13	2013-05-24 12:30:00.000	24	5	2013	6	2	2013-05-24 12:30:00.000	13
14	2013-08-05 19:30:00.000	5	8	2013	2	3	2013-08-05 19:30:00.000	14
15	2014-07-20 11:30:00.000	20	7	2014	1	3	2014-07-20 11:30:00.000	15

Query executed successfully. | ASUSCREATOR (16.0 RTM) | ASUSCREATOR\janin (65) | DWBI_Restaurant_DW | 00:00:00 | 1,000 rows

6.2. Dimension Date

SQLQuery6.sql - AS...REATOR\janin (60))

```
SELECT TOP (1000) [BookingID]
      ,[People]
      ,[Status]
      ,[IsPrepaySatisfied]
      ,[AccmTxnCreateTime]
      ,[AccmTxnCompleteTime]
      ,[TxnProcessTimeHours]
      ,[MemberSK]
      ,[PurposeSK]
      ,[DateSK]
      ,[RestaurantSK]
  FROM [DWBI_Restaurant_DW].[dbo].[FactBooking]
```

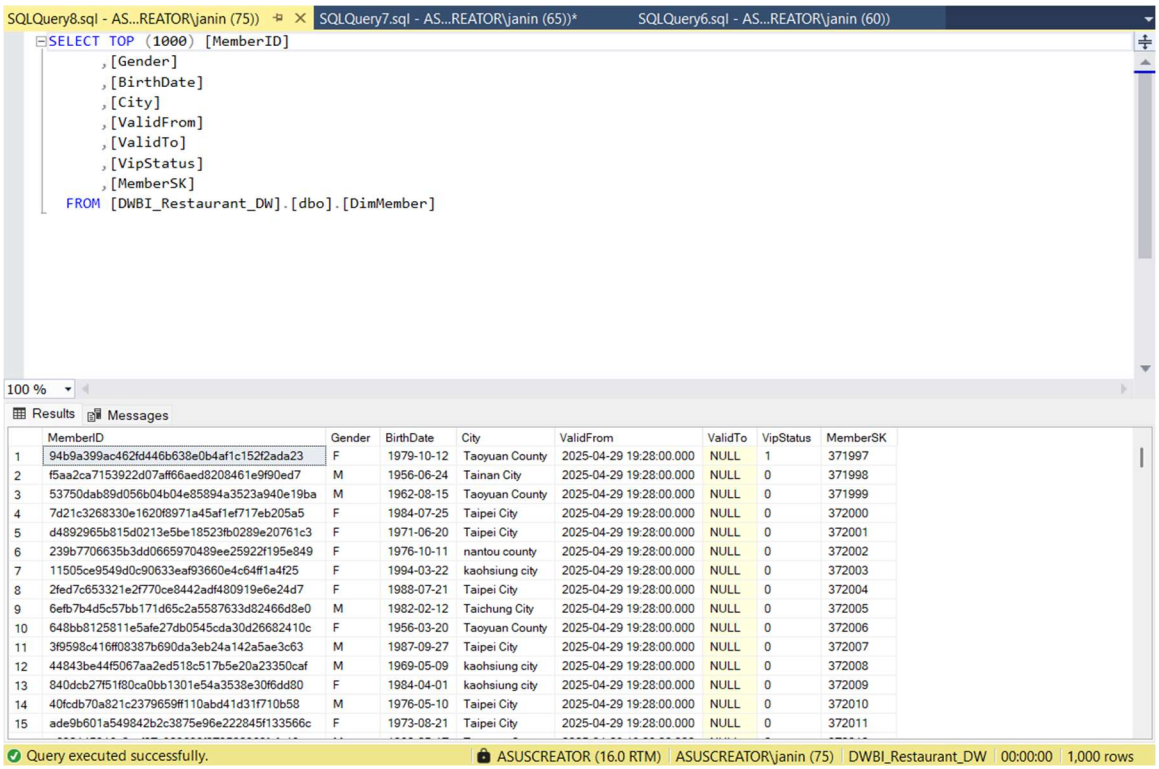
100 %

Results | Messages

	BookingID	People	Status	IsPrepaySatisfied	AccmTxnCreateTime	AccmTxnCompleteTime	TxnProcessTimeHours	MemberSK	PurposeSK	DateSK	RestaurantSK
1	1	5	ok	1	2025-04-29 20:09:00.000	2025-04-30 05:10:37.000	9	428514	91	13023	4558
2	7	9	ok	1	2025-04-29 20:09:00.000	2025-05-04 23:32:38.000	123	423468	56	14906	4545
3	8	2	ok	1	2025-04-29 20:09:00.000	2025-04-30 20:36:28.000	24	479873	168	7209	4565
4	12	2	ok	1	2025-04-29 20:09:00.000	2025-04-30 13:47:27.000	17	455305	30	15262	5012
5	13	4	ok	1	2025-04-29 20:09:00.000	2025-05-01 00:20:50.000	28	464841	33	14244	5182
6	14	7	canceled	1	2025-04-29 20:09:00.000	2025-05-04 18:45:13.000	118	471082	78	3387	4487
7	15	3	canceled	1	2025-04-29 20:09:00.000	2025-05-03 11:15:39.000	87	476961	153	17892	5139
8	16	2	ok	1	2025-04-29 20:09:00.000	2025-05-04 13:22:30.000	113	425476	79	1166	4578
9	18	2	ok	1	2025-04-29 20:09:00.000	2025-05-03 08:54:09.000	84	452538	57	9417	4524
10	19	2	ok	1	2025-04-29 20:09:00.000	2025-04-30 04:03:08.000	8	454591	79	10219	4331
11	20	5	ok	1	2025-04-29 20:09:00.000	2025-05-02 16:38:01.000	68	427381	57	11548	4584
12	22	2	ok	1	2025-04-29 20:09:00.000	2025-05-04 23:12:17.000	123	447399	165	11231	4688
13	23	4	ok	1	2025-04-29 20:09:00.000	2025-04-29 06:50:07.000	-14	429691	97	16002	5147
14	24	3	new	1	2025-04-29 20:09:00.000	2025-05-03 17:35:09.000	93	476803	45	11820	5028
15	26	4	new	1	2025-04-29 20:09:00.000	2025-05-01 14:54:51.000	42	438853	15	16860	4970

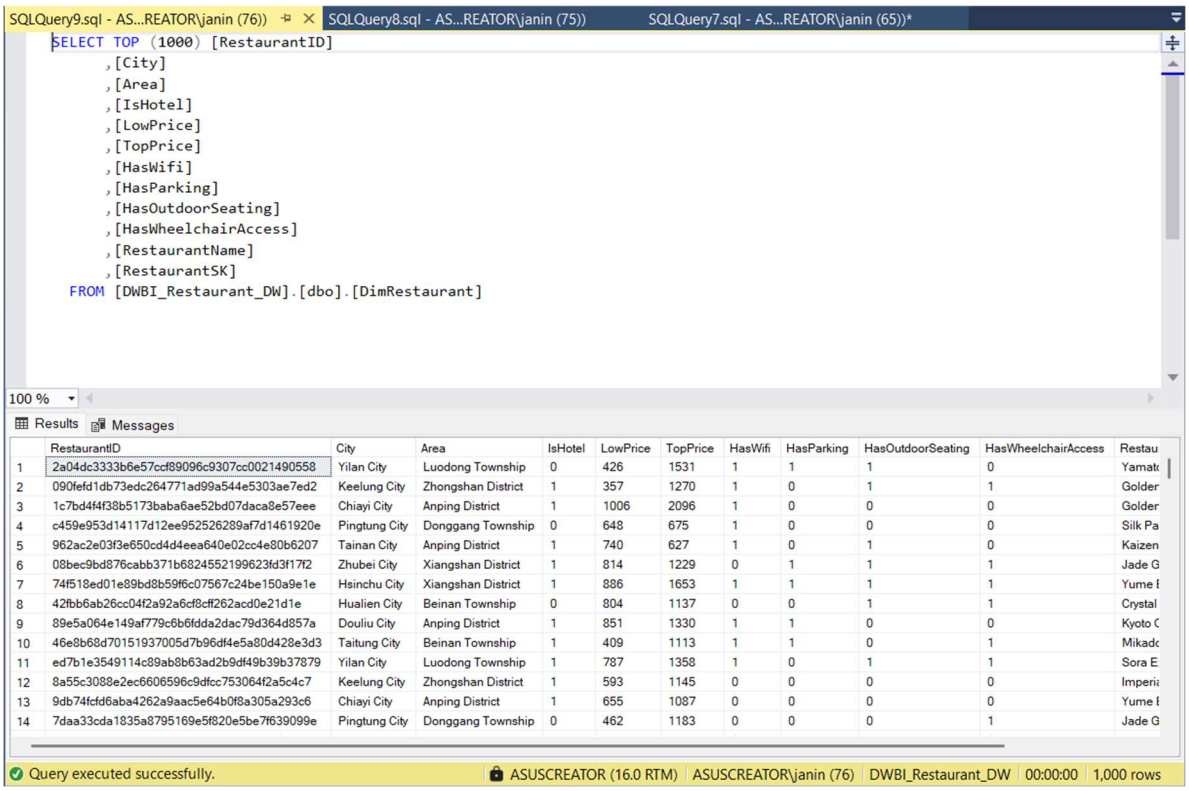
Query executed successfully. | ASUSCREATOR (16.0 RTM) | ASUSCREATOR\janin (60) | DWBI_Restaurant_DW | 00:00:00 | 1,000 rows

6.3. Dimension Member



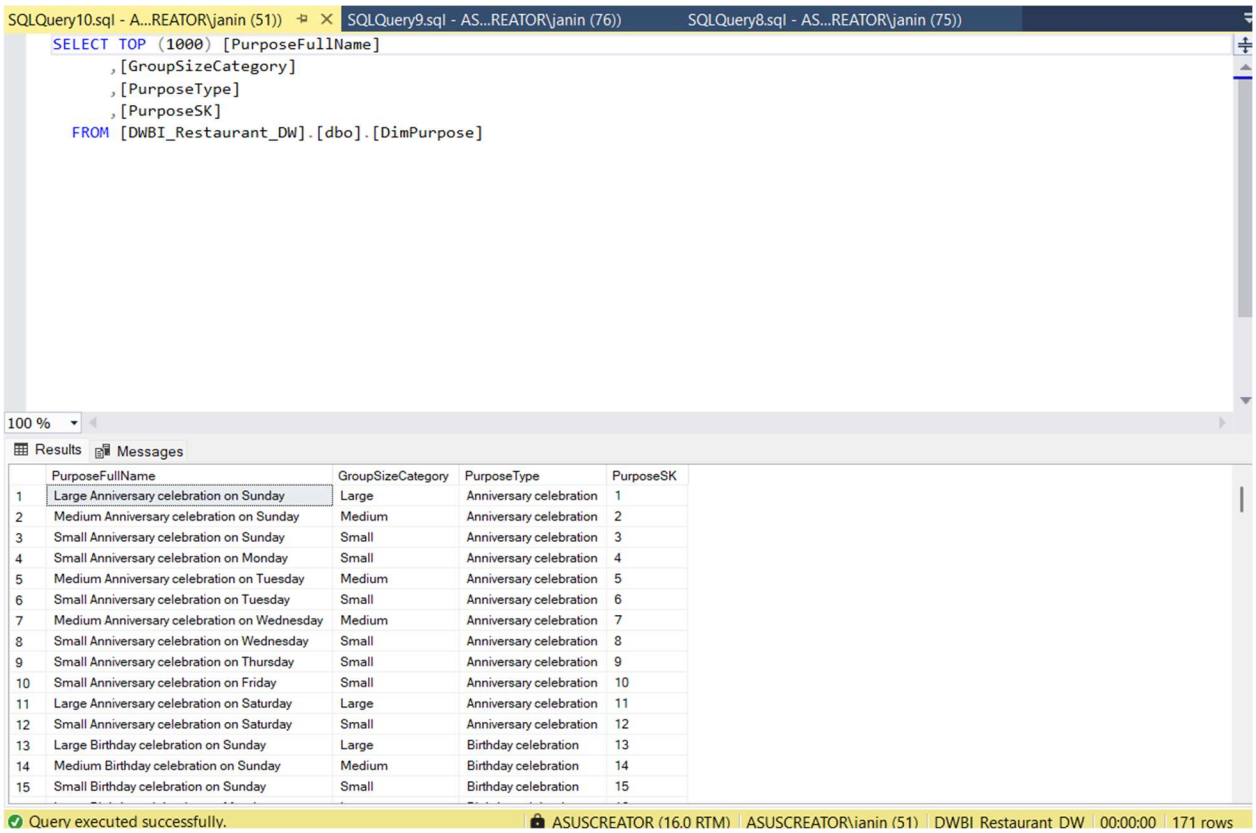
	MemberID	Gender	BirthDate	City	ValidFrom	ValidTo	VipStatus	MemberSK
1	94b9a399ac462fd446b638e0b4af1c152f2ada23	F	1979-10-12	Taoyuan County	2025-04-29 19:28:00.000	NULL	1	371997
2	f5aa2ca7153922d07aff66aed8208461e9f90ed7	M	1956-06-24	Tainan City	2025-04-29 19:28:00.000	NULL	0	371998
3	53750dab89d056b04b04e85894a3523a940e19ba	M	1962-08-15	Taoyuan County	2025-04-29 19:28:00.000	NULL	0	371999
4	7d21c3268330e1620f8971a45af1ef717eb205a5	F	1984-07-25	Taipei City	2025-04-29 19:28:00.000	NULL	0	372000
5	d4892965b815d0213e5be18523fb0289e20761c3	F	1971-06-20	Taipei City	2025-04-29 19:28:00.000	NULL	0	372001
6	239b7706635b3dd0665970489ee25922f195e849	F	1976-10-11	nantou county	2025-04-29 19:28:00.000	NULL	0	372002
7	11505ce9549d0c90633eaf93660e4c64ff1a4f25	F	1994-03-22	kaohsiung city	2025-04-29 19:28:00.000	NULL	0	372003
8	2fed7c653321e2f770ce8442adf480919e6e24d7	F	1988-07-21	Taipei City	2025-04-29 19:28:00.000	NULL	0	372004
9	6efb7b4d5c57bb171d65c2a5587633d82466d8e0	M	1982-02-12	Taichung City	2025-04-29 19:28:00.000	NULL	0	372005
10	648bb8125811e5afe27db0545cda30d26682410c	F	1956-03-20	Taoyuan County	2025-04-29 19:28:00.000	NULL	0	372006
11	3f9598c416ff08387b690da3eb24a142a5ae3c63	M	1987-09-27	Taipei City	2025-04-29 19:28:00.000	NULL	0	372007
12	44843be44f5067aa2ed518c517b5e20a23350caf	M	1969-05-09	kaohsiung city	2025-04-29 19:28:00.000	NULL	0	372008
13	840dcb27f51f80ca0bb1301e54a3538e30f6dd80	F	1984-04-01	kaohsiung city	2025-04-29 19:28:00.000	NULL	0	372009
14	40fcdb70a821c2379659ff110abd41d31f710b58	M	1976-05-10	Taipei City	2025-04-29 19:28:00.000	NULL	0	372010
15	ade9b601a549842b2c3875e96e222845f133566c	F	1973-08-21	Taipei City	2025-04-29 19:28:00.000	NULL	0	372011

6.4. Dimension Restaurant



	RestaurantID	City	Area	IsHotel	LowPrice	TopPrice	HasWifi	HasParking	HasOutdoorSeating	HasWheelchairAccess	Restau
1	2a04dc3333b6e57ccf89096c9307cc0021490558	Yilan City	Luodong Township	0	426	1531	1	1	1	0	Yamat
2	090fefd1db73edc264771ad99a544e5303ae7ed2	Keelung City	Zhongshan District	1	357	1270	1	0	1	1	Golder
3	1c7bd4f4f38b5173baba6ae52bd07daca8e57eee	Chiayi City	Anping District	1	1006	2096	1	0	0	0	Golder
4	c459e953d14117d12ee952526289af7d1461920e	Pingtung City	Donggang Township	0	648	675	1	0	0	0	Silk Pa
5	962ac2e03f3e650cd4d4eea640e02cc4e80b6207	Tainan City	Anping District	1	740	627	1	0	1	0	Kaizen
6	08bec9bd876cabb371b6824552199623fd3f17f2	Zhubei City	Xiangshan District	1	814	1229	0	1	1	1	Jade G
7	74f518ed01e89bd8b59f6c07567c24be150a9e1e	Hsinchu City	Xiangshan District	1	886	1653	1	1	1	1	Yume E
8	42fbb6ab26cc04f2a92a6cf8cff262acd0e21d1e	Hualien City	Beinan Township	0	804	1137	0	0	1	1	Crystal
9	89e5a064e149af779c6b6fdda2dac79d364d857a	Douliu City	Anping District	1	851	1330	1	1	0	0	Kyoto C
10	46e8b68d70151937005d7b96df4e5a80d428e3d3	Taitung City	Beinan Township	1	409	1113	1	1	0	1	Mikado
11	ed7b1e3549114c89ab8b63ad2b9df49b39b37879	Yilan City	Luodong Township	1	787	1358	1	0	1	1	Sora E
12	8a55c3088e2ec6606596c9dfcc753064f2a5c4c7	Keelung City	Zhongshan District	1	593	1145	0	0	0	0	Imperia
13	9db74fcfd6aba4262a9aac5e64b0f8a305a293c6	Chiayi City	Anping District	1	655	1087	0	0	0	0	Yume E
14	7daa33cda1835a8795169e5f820e5be7f639099e	Pingtung City	Donggang Township	0	462	1183	0	0	0	1	Jade G

6.5. Dimension Purpose



	PurposeFullName	GroupSizeCategory	PurposeType	PurposeSK
1	Large Anniversary celebration on Sunday	Large	Anniversary celebration	1
2	Medium Anniversary celebration on Sunday	Medium	Anniversary celebration	2
3	Small Anniversary celebration on Sunday	Small	Anniversary celebration	3
4	Small Anniversary celebration on Monday	Small	Anniversary celebration	4
5	Medium Anniversary celebration on Tuesday	Medium	Anniversary celebration	5
6	Small Anniversary celebration on Tuesday	Small	Anniversary celebration	6
7	Medium Anniversary celebration on Wednesday	Medium	Anniversary celebration	7
8	Small Anniversary celebration on Wednesday	Small	Anniversary celebration	8
9	Small Anniversary celebration on Thursday	Small	Anniversary celebration	9
10	Small Anniversary celebration on Friday	Small	Anniversary celebration	10
11	Large Anniversary celebration on Saturday	Large	Anniversary celebration	11
12	Small Anniversary celebration on Saturday	Small	Anniversary celebration	12
13	Large Birthday celebration on Sunday	Large	Birthday celebration	13
14	Medium Birthday celebration on Sunday	Medium	Birthday celebration	14
15	Small Birthday celebration on Sunday	Small	Birthday celebration	15

7. With all tables properly structured and interconnected using **foreign keys** to the **FactBooking** table, ensuring a well-defined relational model. Each dimension—**DimMember, DimPurpose, DimDate, and DimRestaurant**—has its own **surrogate key (SK)**, facilitating joins and efficient data retrieval. These relationships establish a **proper data warehouse schema**, enabling seamless reporting, analysis, and decision-making within the BI solution.

- THE END -