



**UNIVERSITY OF KELANIYA – SRI LANKA
FACULTY OF COMPUTING AND TECHNOLOGY**

CSCI 32092 - Data Mining and Warehousing Coursework

Academic Year 2024

REPORT OF THE COURSEWORK

Student no: CS/2019/053

Student Name:M.D.H.M.Gunathilka

SECTION A: DIMENSIONAL MODELING

Do the following tasks for designing the logical data.

I. Identify contexts for measures for sales analytics and make a Bus Matrix, showing all the Possible attributes of identified dimensions.

Contexts for Measures:

1. **Sales**
2. **Marketing Expenditure**
3. **Real-Time Sales**

Dimensions Identified:

1. **Time**
2. **Customer**
3. **Product**
4. **Region**

Possible Attributes of Identified Dimensions:

1. Time Dimension:

- Year
- Quarter
- Month
- Day

2. Customer Dimension:

- CustomerID
- CustomerName
- ContactNumber
- RegionID

3. Product Dimension:

- ProductID
- ProductName
- Price
- Category

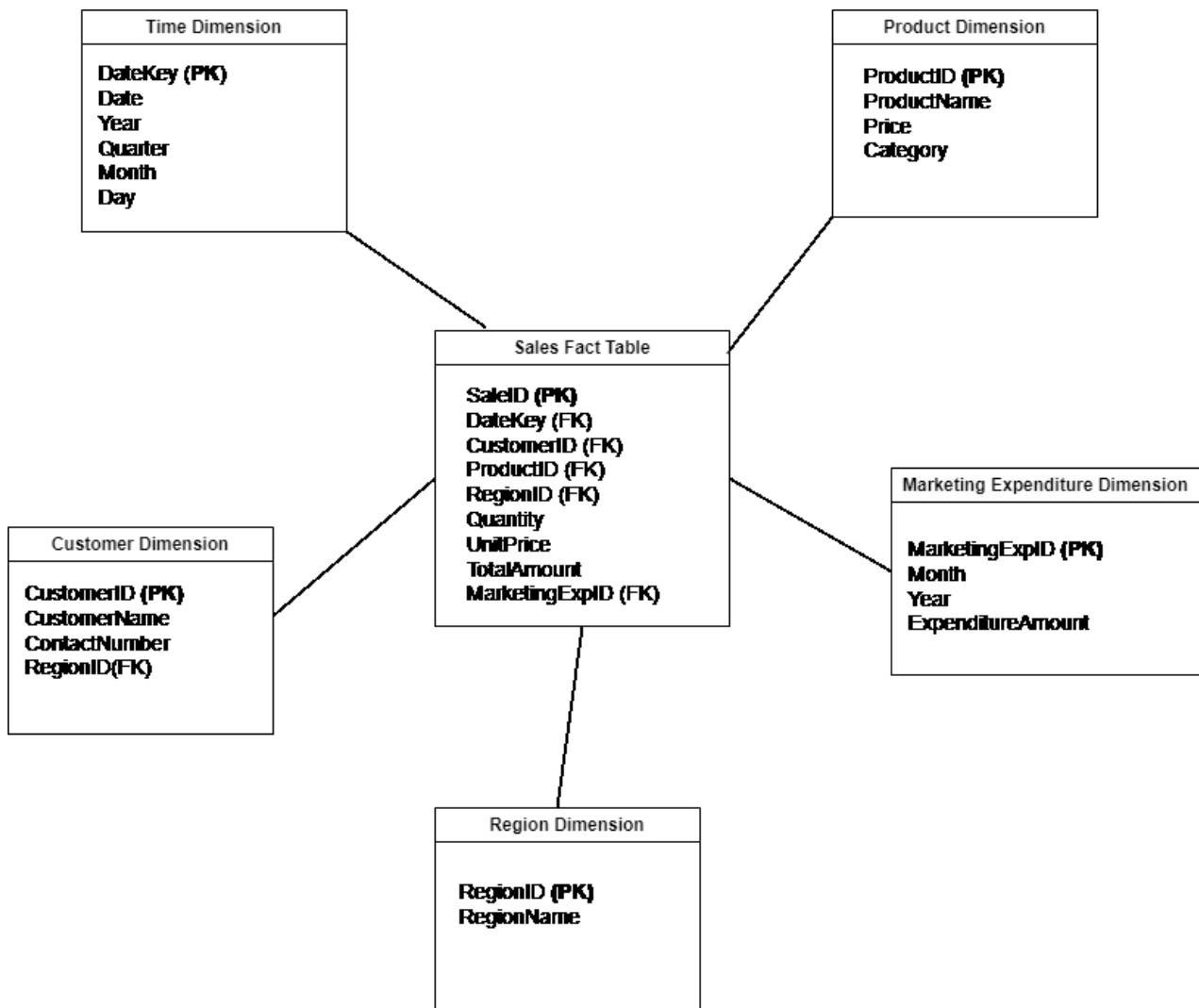
4. Region Dimension:

- RegionID
- RegionName

Bus Matrix

Measure	Time	Customer	Product	Region
Total Sales Amount	✓	✓	✓	✓
Total Quantity Sold	✓	✓	✓	✓
Marketing Expenditure	✓			
Unit Price			✓	
Sales Count	✓	✓	✓	✓

II. Design a star schema for the above scenario. Make sure to consider data coming from all the **3 data sources** when you are deciding 'fields' in your dimensions and fact table



SECTION B: PHYSICAL DW DESIGN AND ETL USING ADF

Assumptions for Completing the POC Using Azure Services

1. Source Database Setup:

- The source database is created using Azure SQL Database according to the provided ER diagram.
- The table structures are created exactly as per the given ER diagram, with appropriate primary and foreign key constraints.

2. Data Insertion:

- The data will be consistent with the defined data types and relationships in the ER diagram.

3. Primary Key and Foreign Key Constraints:

- Each table will have primary keys as defined: `SaleID` for Sales, `CustomerID` for Customers, `ProductID` for Products, and `RegionID` for Regions.
- Foreign key constraints will be enforced to maintain data integrity between tables.

4. Data Types and Sizes:

- Assumptions about the data types and sizes for the table columns will be made if not explicitly specified. For example, `CustomerName` might be assumed as `VARCHAR(100)`, `ContactNumber` as `VARCHAR(15)`, `RegionName` as `VARCHAR(50)`, etc.

5. Date Handling:

- The `Date` column in the Sales table will be assumed to be in the format `YYYY-MM-DD` and will be of data type `DATE`.

6. Azure Data Factory (ADF) Configuration:

- Azure Data Factory will be used to create and configure data flows for ETL (Extract, Transform, Load) processes.
- Linked services and datasets will be configured to connect to the Azure SQL Database.

7. Lookup Transformations in ADF:

- Lookup transformations in ADF will be used to handle foreign key relationships and retrieve dimension keys.
- Multiple lookup transformations can be used within the same data flow to enrich the Sales fact table with dimension keys from the related tables.

8. ETL Process:

- The ETL process will involve extracting data from the source tables, performing necessary transformations (e.g., extracting year and month from the `Date` column), and loading the data into the target fact table and dimension tables.

- Error handling and logging mechanisms will be implemented to monitor the ETL process.

9. Scalability and Performance:

- Assumptions about the scalability and performance of the Azure SQL Database and ADF will be considered, ensuring that the setup can handle the expected volume of data and ETL workload.

10. Security and Access:

- Proper security measures, such as firewall rules and access controls, will be configured to secure the Azure SQL Database and ADF.
- The user executing the ETL processes will have the necessary permissions to read from the source tables and write to the target tables.

11. Testing and Validation:

- After setting up the source database and ETL processes, thorough testing will be conducted to validate the data integrity and correctness of the transformations.
- Sample queries will be run to verify the relationships and data in the populated tables.

Tables in Source Database

Customers Table

The screenshot shows the Microsoft Azure portal's Query editor (preview) for the 'sourcedb1' database. The interface includes a navigation bar with tabs for 'sourcedb1 (geminiserver/sourcedb1)', 'Query editor (preview)', and '...'. Below the navigation bar, there is a message about network settings preventing query issuance. The main area features a 'Query 1' tab with a table of customer data. The table has columns: CustomerID, CustomerName, ContactNumber, and RegionID. The data is as follows:

CustomerID	CustomerName	ContactNumber	RegionID
C100	John Doe	1234567890	1
C101	Jane Smith	1234567891	2
C102	Alice Johnson	1234567892	3
C103	Bob Brown	1234567893	1

At the bottom of the screen, a status bar indicates 'Query succeeded | 0s' and shows the system time as '7:16:15 PM' on '7/5/2024'.

Sales Table

Sourcedb1 (geminiserver/sourcedb1) | Query editor (preview)

Showing limited object explorer here. For full capability please click here to open Azure Data Studio.

SaleID	ProductID	CustomerID	Date	Quantity	TotalAmount
S011	P100	C100	2024-01-01	2	240.00
S012	P101	C101	2024-01-02	1	150.00
S013	P102	C102	2024-01-03	3	270.00
S014	P103	C103	2024-01-04	1	110.00

Products Table

Sourcedb1 (geminiserver/sourcedb1) | Query editor (preview)

Showing limited object explorer here. For full capability please click here to open Azure Data Studio.

ProductID	ProductName	Price	Category
P100	Widget A	120	Electronics
P101	Widget B	150	Electronics
P102	Gadget C	90	Home
P103	Gadget D	110	Home

Regions Table

The screenshot shows the Microsoft Azure Query editor (preview) interface. The left sidebar includes links for Overview, Activity log, Tags, Diagnose and solve problems, and the current Query editor (preview). The main area displays two queries. Query 1 shows data from the 'dbo.Customers' table, and Query 2, which is the active tab, shows data from the 'dbo.Sales' table. The results for Query 2 are displayed as a table:

RegionID	RegionName
1	North
2	South
3	East

At the bottom, a message indicates "Query succeeded | 0s". The status bar at the bottom right shows the date and time as 11/4/2024 10:40 PM.

The screenshot shows the Microsoft Azure Query editor interface. The left sidebar displays the database schema with tables like Customer_Dimension, Marketing_Expenditure_Dimension, Product_Dimension, Region_Dimension, and Sales_Fact_Table. The main area shows a query script:

```
33    ExpenditureAmount INT  
34 );  
35  
36 CREATE TABLE Sales_Fact_Table (  
37     SalesID INT PRIMARY KEY,  
38     DateKey INT,  
39     CustomerID INT,  
40     ProductID INT,  
41     RegionID INT,  
42     Quantity INT,  
43     UnitPrice INT,  
44     MarketingExpID INT,  
45     TotalAmount DECIMAL(10, 2),  
46     CONSTRAINT fk_DateKey FOREIGN KEY (DateKey) REFERENCES Time_Dimension(DateKey),  
47     CONSTRAINT fk_ProductID FOREIGN KEY (ProductID) REFERENCES Product_Dimension(ProductID),  
48     CONSTRAINT fk_CustomerID FOREIGN KEY (CustomerID) REFERENCES Customer_Dimension(CustomerID),  
49     CONSTRAINT fk_RegionID FOREIGN KEY (RegionID) REFERENCES Region_Dimension(RegionID),  
50     CONSTRAINT fk_MarketingExpID FOREIGN KEY (MarketingExpID) REFERENCES Marketing_Expenditure_Dimension(Marketin  
51 );  
52 DROP TABLE Customer_Dimension;  
53
```

The status bar at the bottom indicates "Query succeeded | 0s".

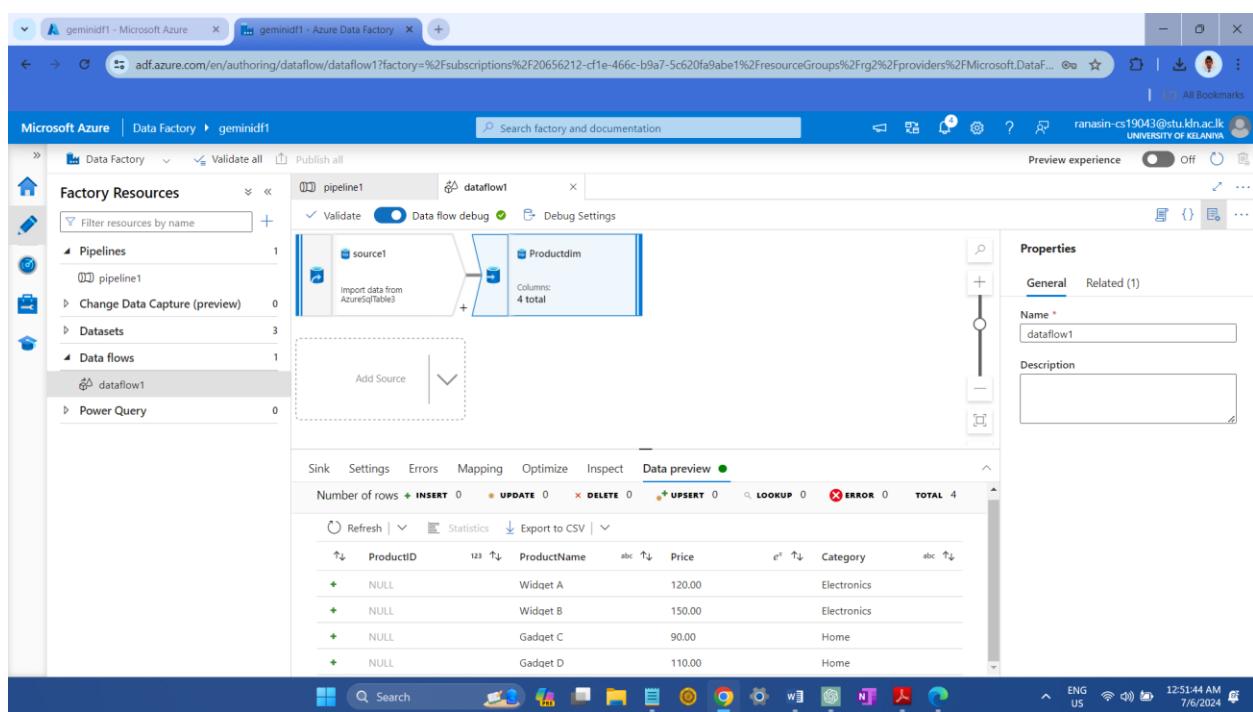
targetdb1 (geminiserver/targetdb1) | Query editor (preview)

```

33    ExpenditureAmount INT
34  );
35
36 CREATE TABLE Sales_Fact_Table (
37   SalesID INT PRIMARY KEY,
38   DateKey INT,
39   CustomerID INT,
40   ProductID INT,
41   RegionID INT,
42   Quantity INT,
43   UnitPrice INT,
44   MarketingExpID INT,
45   TotalAmount DECIMAL(10, 2),
46   CONSTRAINT fk_datekey FOREIGN KEY (DateKey) REFERENCES Time_Dimension(DateKey),
47   CONSTRAINT fk_ProductID FOREIGN KEY (ProductID) REFERENCES Product_Dimension(ProductID),
48   CONSTRAINT fk_CustomerID FOREIGN KEY (CustomerID) REFERENCES Customer_Dimension(CustomerID),
49   CONSTRAINT fk_RegionID FOREIGN KEY (RegionID) REFERENCES Region_Dimension(RegionID),
50   CONSTRAINT fk_MarketingExpID FOREIGN KEY (MarketingExpID) REFERENCES Marketing_Expenditure_Dimension(MarketingExpID)
51 );
52 DROP TABLE Customer_Dimension;
53

```

Query succeeded | 0s



The screenshot shows the Microsoft Azure Data Factory pipeline editor. On the left, the 'Factory Resources' sidebar lists 'Pipelines' (pipeline1), 'Datasets' (8), 'Data flows' (2), and 'Power Query' (0). The main workspace displays 'pipeline1' with a single 'Copy data' activity named 'Copy data1'. Below the activities, the 'Output' tab shows a table of pipeline run details:

Activity name	Activity status	Activity type	Run start	Duration	Integration runtime
Copy data1	Succeeded	Copy data	7/6/2024, 1:36:06 AM	16s	AutoResolveIntegration

At the bottom right, the status bar shows '1:37:22 AM 7/6/2024'.

The screenshot shows the Microsoft Azure Data Factory pipeline editor. On the left, the 'Factory Resources' sidebar lists 'Pipelines' (4) including pipeline1, pipeline2, pipeline3, and pipeline4; 'Datasets' (23); 'Data flows' (3); and 'Power Query' (0). The main workspace displays 'pipeline3' with a single 'Copy data' activity named 'Copy data1'. To the right, the 'Properties' panel is open for 'pipeline3', showing 'General' tab with 'Name' set to 'pipeline3' and 'Description' empty. The 'Output' tab shows a table of pipeline run details:

Activity name	Activity status	Activity type	Run start	Duration	Integration runtime
Copy data1	Succeeded	Copy data	7/6/2024, 8:56:42 PM	16s	AutoResolveIntegration

At the bottom right, the status bar shows '9:57:41 PM 7/6/2024'.

V. Use Azure Data Factory to perform ETL from 3 data sources to the target data warehouse. Use necessary schema modifiers (i.e., derived column (to calculate the total sales by unit price x quantity), lookup, join (to join the real-7me sales data in JSON with the historical sales data in Azure target SQL database), union, etc.)

The screenshot shows the Microsoft Azure Data Factory interface for the factory 'geminidf1'. On the left, the 'Factory Resources' sidebar lists 'Pipelines' (pipeline1, pipeline2, pipeline3, pipeline4, pipeline5) and 'Datasets' (AzureSqlTable1 to AzureSqlTable9). The main workspace displays 'Activities' for pipeline5, specifically a 'Data flow' activity named 'Data flow1'. The 'Properties' pane on the right shows the pipeline's name as 'pipeline5' and its status as 'Succeeded'. Below the activities, a table provides details about the most recent run, including the Pipeline run ID (883417c8-5b3e-406e-bdc8-c0672472f0a4), Pipeline status (Succeeded), and Run start (7/7/2024, 7:09:21 AM).

This screenshot shows the Microsoft Azure Data Factory interface for the factory 'geminidf1'. The 'Factory Resources' sidebar lists 'Pipelines' (pipeline1, pipeline2, pipeline3, pipeline4, pipeline5) and 'Datasets' (AzureSqlTable1 to AzureSqlTable9). The main workspace displays the 'Activities' for pipeline5, specifically a complex 'Data flow' activity named 'dataflow4'. This data flow consists of multiple stages: source1 (TimeSource), lookup2, sink1; source1 (CustomerSource), lookup3; source1 (ProductSource); TimeSource; and CustomerSource. The 'Properties' pane on the right shows the pipeline's name as 'dataflow4'.

VI. Assuming analysis needs to be done in the period 2024/1/1 – 2024/5/24, populate your date dimension manually using SQL query or by performing ETL using a source CSV file.

```

10 );
11 -- Insert date values from 2024/01/01 to 2024/05/24
12 DECLARE @startDate DATE = '2024-01-01';
13 DECLARE @endDate DATE = '2024-05-24';
14
15 ;WITH DateRange AS (
16     SELECT @startDate AS Date
17     UNION ALL
18     SELECT DATEADD(DAY, 1, Date)
19     FROM DateRange
20     WHERE Date < @endDate
21 )
22 INSERT INTO Time_Dimension (DateKey, Date, Year, Quarter, Month, Day)
23 SELECT
24     YEAR(Date) = 10000 + MONTH(Date) * 100 + DAY(Date) AS DateKey,
25     Date,
26     YEAR(Date) AS Year,
27     DATEPART(QUARTER, Date) AS Quarter,
28     MONTH(Date) AS Month,
29     DAY(Date) AS Day
30  FROM

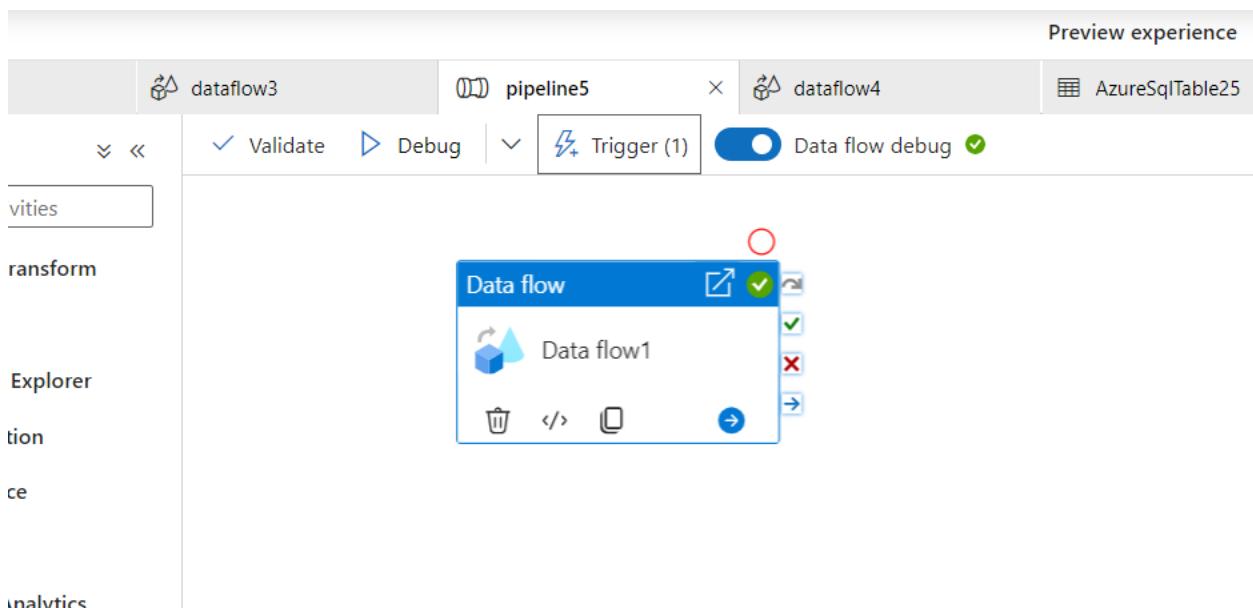
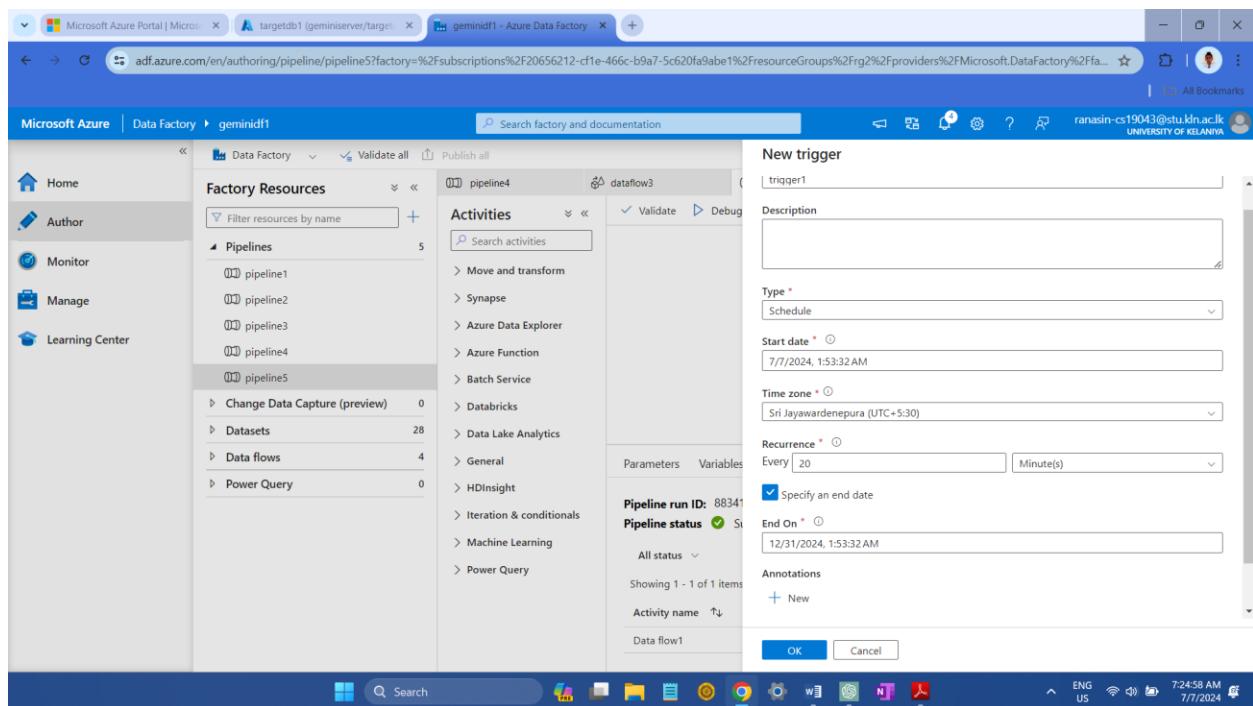
```

Query succeeded | 0s

DateKey	Year	Quarter	Month	Day	Date
20240101	2024	1	1	1	2024-01-01
20240102	2024	1	1	2	2024-01-02
20240103	2024	1	1	3	2024-01-03
20240104	2024	1	1	4	2024-01-04
20240105	2024	1	1	5	2024-01-05
20240106	2024	1	1	6	2024-01-06
20240107	2024	1	1	7	2024-01-07
20240108	2024	1	1	8	2024-01-08
20240109	2024	1	1	9	2024-01-09

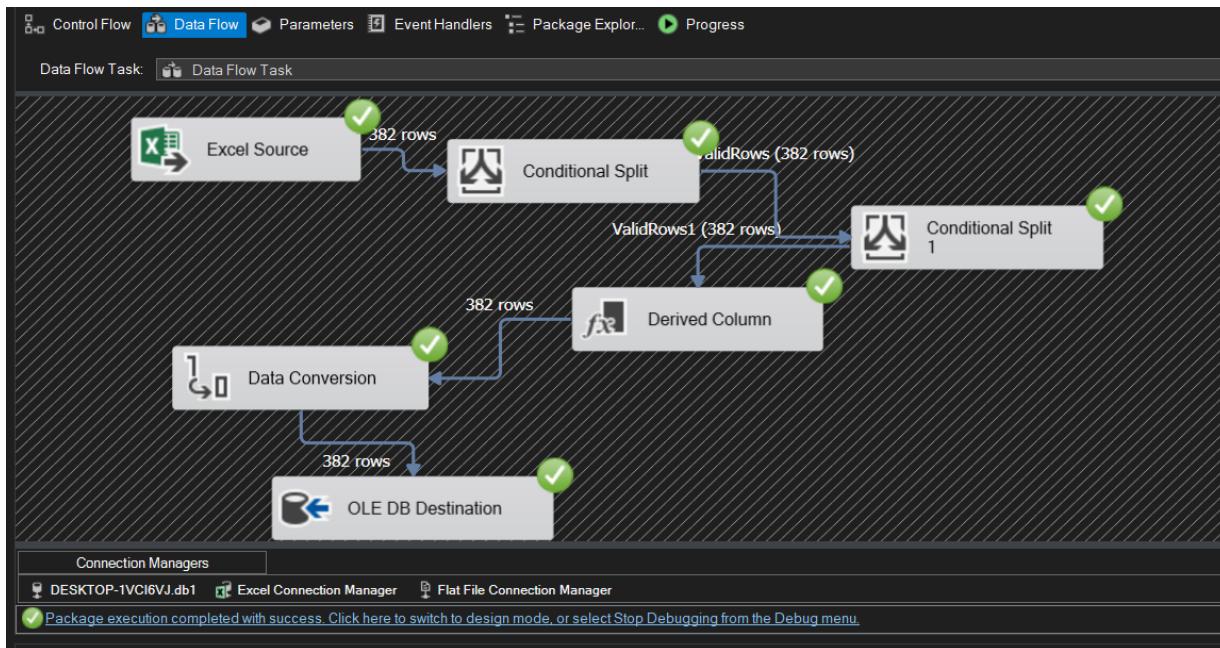
Query succeeded | 0s

VII. Create a trigger in Azure Data Factory, to run the ETL pipeline every 20 minutes with the end date 2024/12/31.



SECTION C: SSIS

Using SSIS, clean the data by removing any rows where 'Electorate' is less than 'ValidVotes' as this may indicate a data entry error. Then, calculate the percentage of votes that were for 'Remain' and 'Leave' in each area. (Hint: Get the output to a text file using the flat file destination task in the SSIS toolbox)



SQL Server Management Studio (SSMS) interface showing a query results grid and a query editor window.

Object Explorer:

- Connected to DESKTOP-1VC16VJ (SQL Server 16.0.1000.6 - sa)
- Databases: master, msdb, tempdb, db1
- Tables: dbo.Modified_Voting_figures
- Views: dbo.Winning_Side
- Other: Security, Replication, Always On High Availability, Management, Integration Services Catalogs

Query Editor:

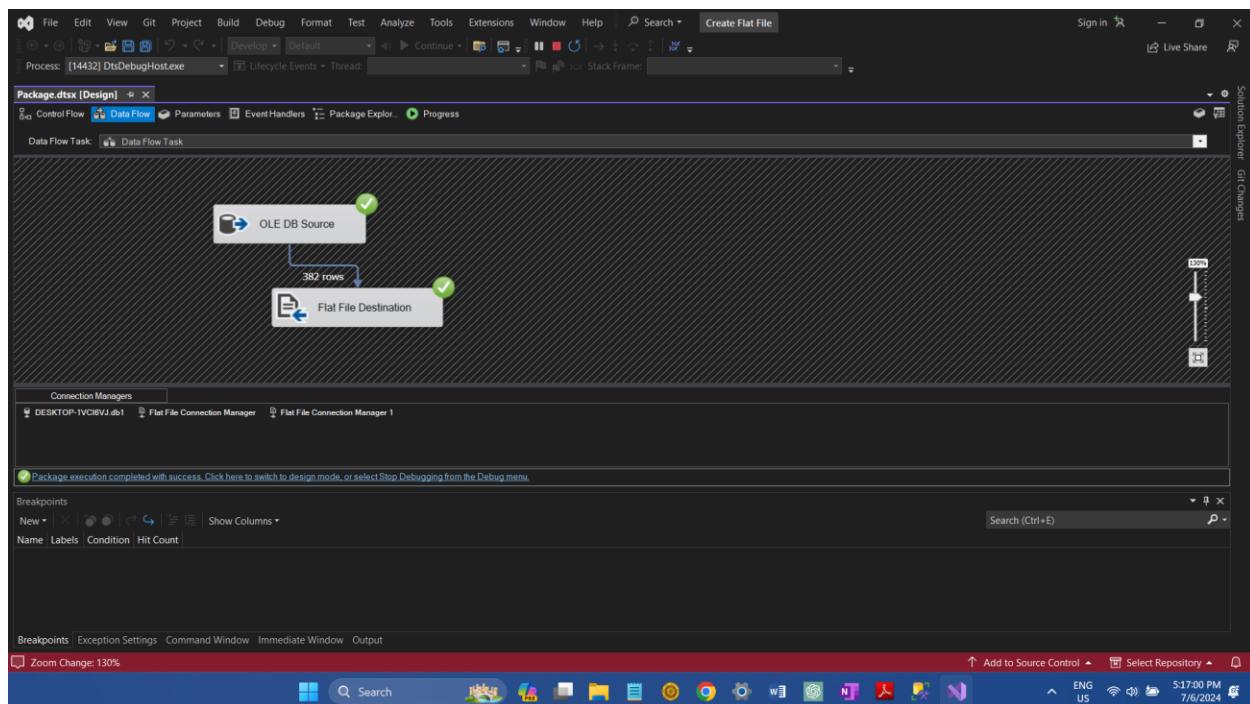
```

SELECT TOP (1000) [Region]
      ,[Area]
      ,[Electorate]
      ,[ValidVotes]
      ,[Remain]
      ,[Leave]
      ,[RemainPercentage]
      ,[LeavePercentage]
  FROM [db1].[dbo].[Modified_Voting_figures]
  
```

Results Grid:

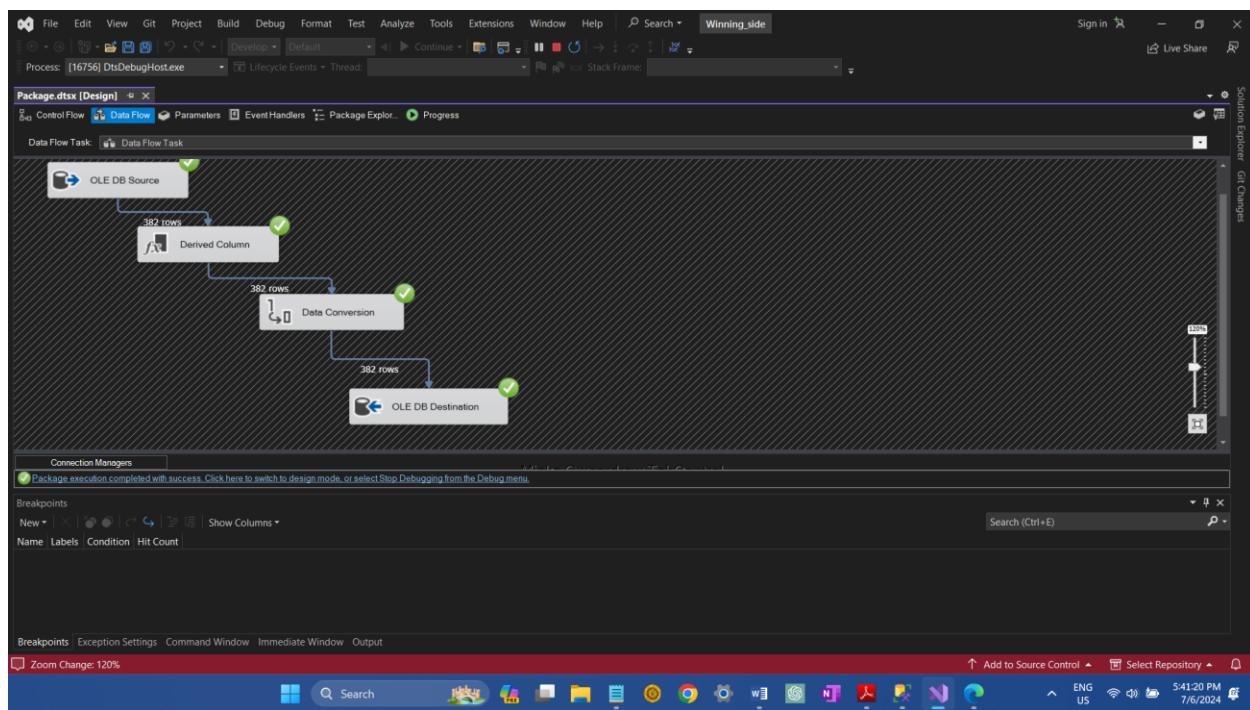
	Region	Area	Electorate	ValidVotes	Remain	Leave	RemainPercentage	LeavePercentage
1	East	Peterborough	128905	87396	34176	53216	39.10	60.89
2	East	Luton	127812	87051	35797	52773	43.45	56.54
3	East	Southend-on-Sea	128805	83870	30348	64527	37.91	62.08
4	East	Thurrock	109897	79916	22151	57785	27.71	72.28
5	East	Bedford	119530	86066	41497	44569	48.21	51.78
6	East	Central Bedfordshire	204004	158004	69670	89134	43.87	56.12
7	East	Cambridge	80108	57799	42682	15117	73.84	26.15
8	East	East Cambridgeshire	62435	48098	23599	24487	49.07	50.92
9	East	Fenland	71447	52626	15055	37571	28.60	71.39
10	East	Huntingdonshire	128486	99927	45729	54195	45.76	54.23
11	East	South Cambridgeshire	114830	93180	56128	37061	60.23	39.76
12	East	Bassilione	132771	97996	30748	67251	31.37	68.62
13	East	Brantree	112562	86239	33523	52713	38.87	61.12
14	East	Brentwood	58780	10071	27627	40354	59.15	40.85
15	East	Carterton	65860	81845	14154	51706	23.30	76.69
16	East	Chelmsford	129971	107074	47645	63249	47.17	52.82
17	East	Colchester	127520	65719	44414	51205	46.40	53.59
18	East	Epping Forest	100016	76852	28676	48176	37.31	62.68
19	East	Harlow	59124	43469	13887	29602	31.90	68.09
20	East	Maldon	49973	38831	14529	24302	37.41	62.58
21	East	Rochford	66589	52447	17510	34937	33.38	66.61
22	East	Tendring	111167	82657	25210	57447	30.49	69.50
23	East	Uttlesford	64735	51943	25619	26324	49.32	50.67
24	East	Bromsgrove	68897	50872	17166	33706	33.74	66.25
25	East	Dacorum	108965	86244	42542	43702	49.32	50.67
26	East	Hemel Hempstead	73096	56125	27583	78532	49.16	50.83

Query executed successfully.



Flat File Brexit_Voters_Figure_TEXT.txt is attached

X. Create an SSIS package that aggregates the total number of votes (Remain and Leave) per region and calculates the total electorate per region. Also, determine which side won in each region based on the total votes. (Hint: Get the output to a text file using a flat file destination task in the SSIS toolbox)



SQlQuery3.sql - DESKTOP-1VC16VJ.db1 (sa (57)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

New Query Execute Quick Launch (Ctrl+Q)

Object Explorer

Connect DESKTOP-1VC16VJ (SQL Server 16.0.1000.6 - sa)

Databases System Database Database Snapshots db1 Tables System Tables FileTables External Tables Graph Tables cbo.Modified_Voting_figures Columns Keys Constraints Triggers Indexes Statistics cbo.Winning_Side Dropped Ledger Tables Views External Resources Synonyms Programmability Query Store Service Broker Storage Security my_first_database Security Server Objects Replication Always On Availability Group Management Integration Services Catalog

SQLQuery3.sql - DES...C16VJ.db1 (sa (59))

```
SELECT TOP (1000) [Region]
    ,[Electorate]
    ,[ValidVotes]
    ,[Remain]
    ,[Leave]
    ,[Total_Electorate]
    ,[Winning_Side]
FROM [db1].[dbo].[Winning_Side]

DELETE FROM Winning_Side;
SELECT * FROM Winning_Side;
```

Results Messages

Region	Area	Electorate	ValidVotes	Remain	Leave	Total_Electorate	Winning_Side
1	East	Peterborough	120892	87982	34178	53215	87982
2	East	Brent	127612	64481	36708	47715	84481
3	East	Southend-on-Sea	128956	93870	39249	64523	93870
4	East	Thurrock	109897	79916	22151	57765	79916
5	East	Bedford	119530	60686	41497	44599	60686
6	East	Central Bedfordshire	204004	158804	69670	89134	158804
7	Cambridge		80108	57799	42682	15117	57799
8	East	East Cambridgeshire	62435	48086	23599	24487	48086
9	East	Fenland	71447	52626	37571	52626	Leave
10	East	Huntingdonshire	128486	99227	45729	54191	99227
11	East	South Cambridgeshire	114830	93189	56128	37061	93189
12	East	Basildon	132771	97999	30748	67251	97999
13	East	Braintree	112562	86073	35370	51713	86073
14	East	Brentwood	58777	56704	10577	78262	Leave
15	East	Castle Point	68860	51045	14154	37691	51045
16	East	Chelmsford	129971	100794	47545	83249	100794
17	East	Colchester	127520	95719	44414	51305	95719
18	East	Epping Forest	100016	76852	20876	48176	76852
19	East	Harrow	59124	43469	13887	29602	43469
20	East	Maldon	49073	38831	14529	24302	38831
21	East	Rochford	66589	52447	17510	34937	52447
22	East	Tendring	111167	83667	36310	83667	83667

Query executed successfully.

DESKTOP-1VC16VJ (16.0 RTM) | sa (57) | db1 | 00:00:00 | 382 rows

Ready

File Edit View Git Project Build Debug Format Test Analyze Tools Extensions Window Help Search Create Flat File

Process: [14432] DtsDebugHost.exe Lifecycle Events Thread: Stack Frame: Live Share

Sign in ENG US 5:41:00 PM 7/6/2024

Package dttx [Design] Control Flow Data Flow Parameters EventHandlers Package Explorer Progress

Data Flow Task

OLE DB Source → Flat File Destination

OLE DB Source → Flat File Connection Manager

Flat File Connection Manager

Package execution completed with success. Click here to switch to design mode, or select Stop Debugging from the Debug menu.

Breakpoints New | X | ○ | ⌂ | ⌂ | Show Columns *

Name Labels Condition Hit Count

Search (Ctrl+E)

Breakpoints Exception Settings Command Window Immediate Window Output

Zoom Change: 130%

Add to Source Control Select Repository

ENG US 5:17:00 PM 7/6/2024

Flat File [Winning_Text.txt](#) is attached.