

i. Physical model of the source system



ii. Dimensional Model

a. Business Processes

- **Sales Transactions** individual product sales
- **Customer Activity** monitoring customer engagements
- **Employee Performance** evaluating employee sales metrics

b. Grain

- **FactSales**: Each row represents a single product sale at a specific time.
- **FactCustomerActivity**: Each row represents customer activity (e.g., purchase or visit) for a given date.
- **FactEmployeePerformance**: Each row summarizes an employee's sales over a defined period (e.g., daily or monthly).

c. Fact Tables Types

Fact Table	Type	Description
FactSales	Transactional Fact	Records every sale event
FactCustomerActivity	Snapshot Fact	Captures customer actions per time unit
FactEmployeePerformance	Snapshot Fact	Captures employee performance statistics

d. Dimensions & their Types

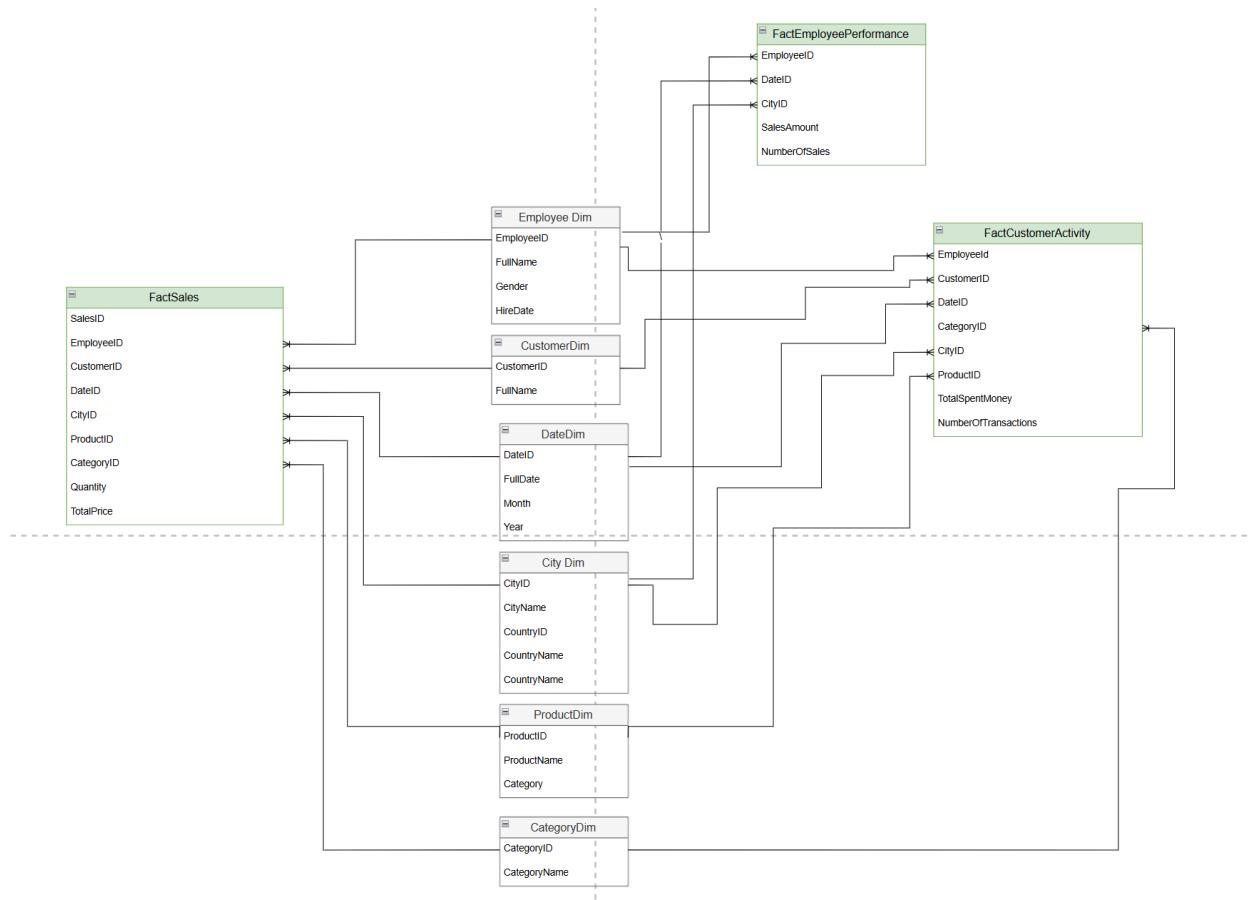
Dimension	Type
DimDate	Conformed, Role-Playing Dimension
DimCustomer	Conformed

	, Slowly Changing Dimension
DimEmployee	Conformed
DimProduct	Conformed , Slowly Changing Dimension
DimLocation	Conformed
DimCategory	Conformed

e. Measures & their Types

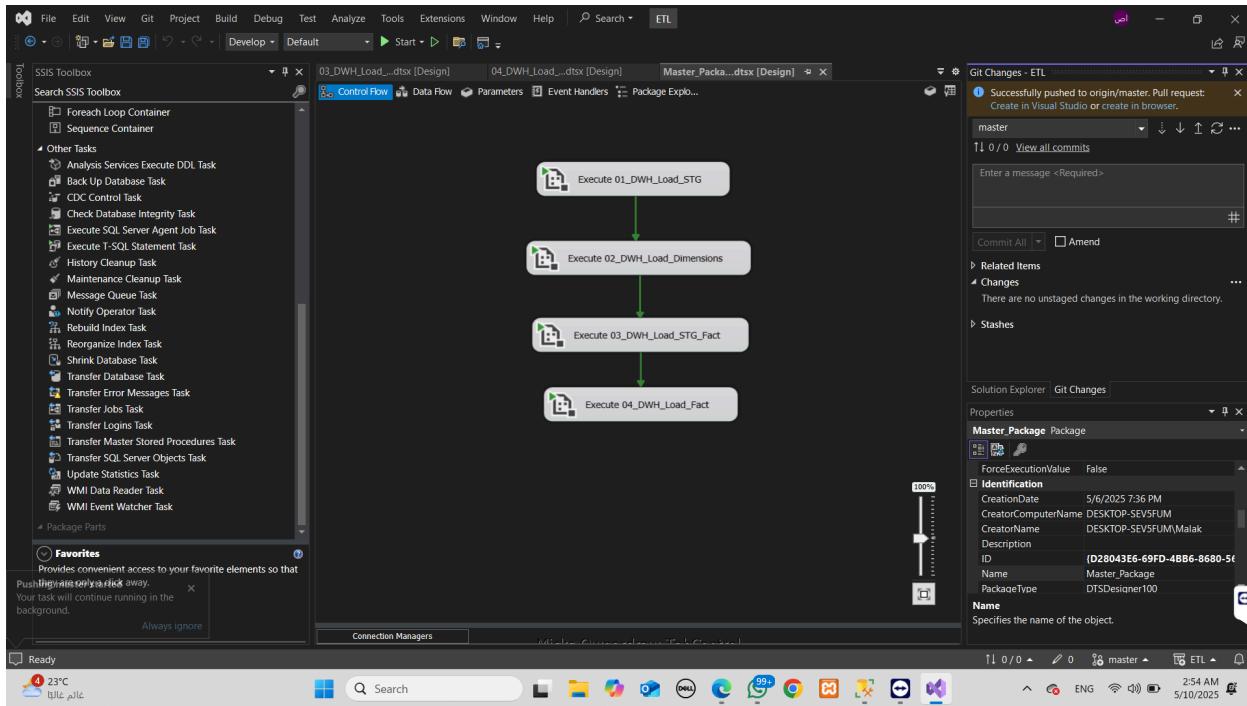
Measure	Type	Fact Table
TotalPrice	Additive	FactSales
Quantity	Additive	FactSales
TotalSpentMoney	Additive	FactCustomerActivity
numberOfTransactions	Additive	FactCustomerActivity
SalesAmount	Additive	FactEmployeePerformance
NumberOfSales	Additive	FactEmployeePerformance

f. Final Star/Galaxy Schema

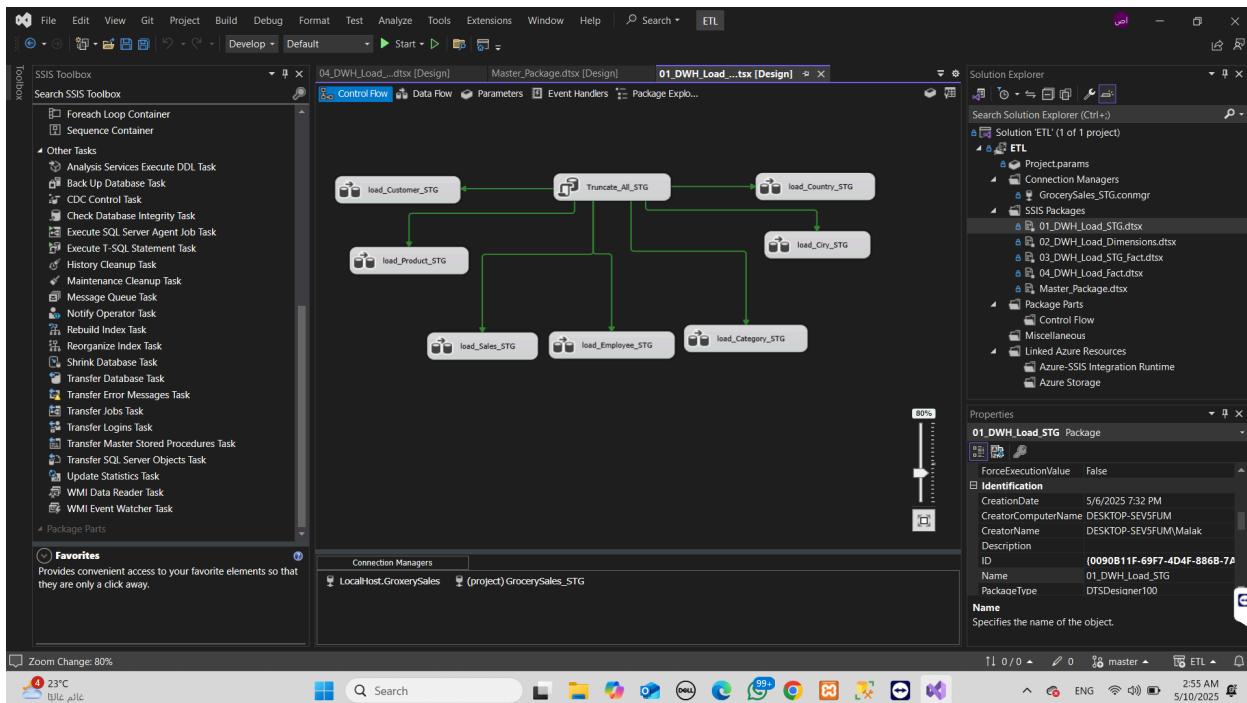


iii. Data Flow & Control Flow Screenshots

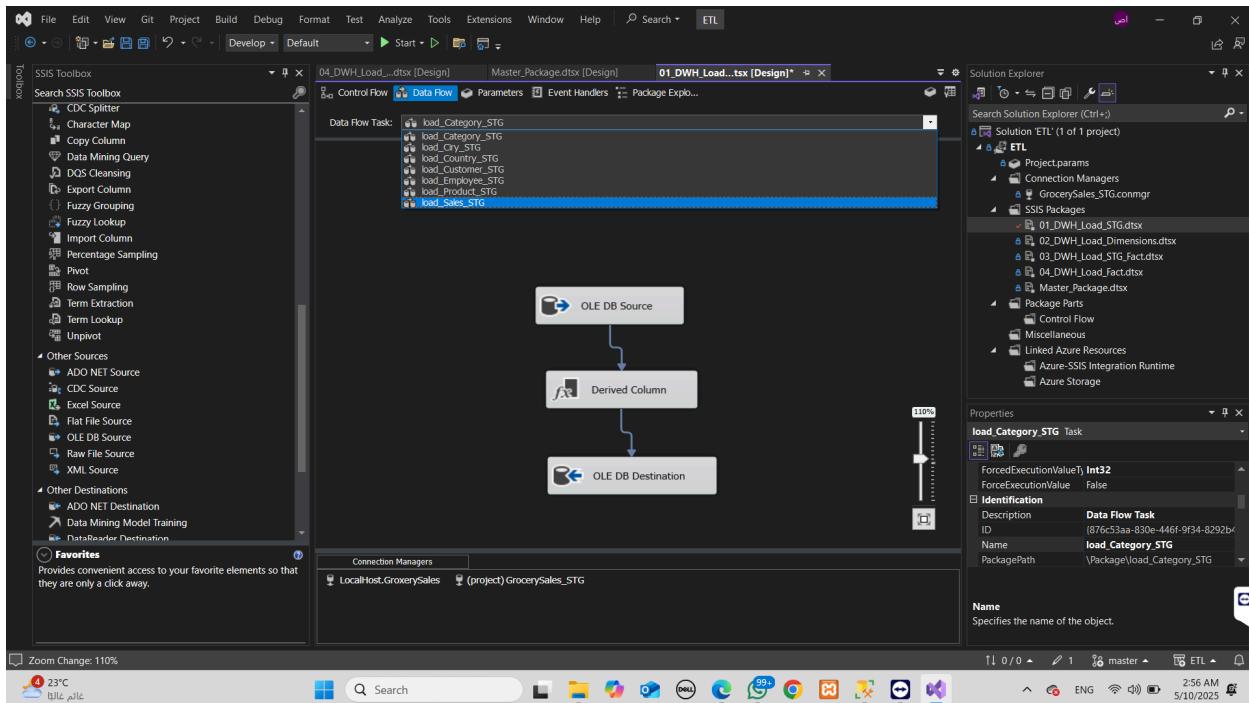
1. Master Package Control Flow



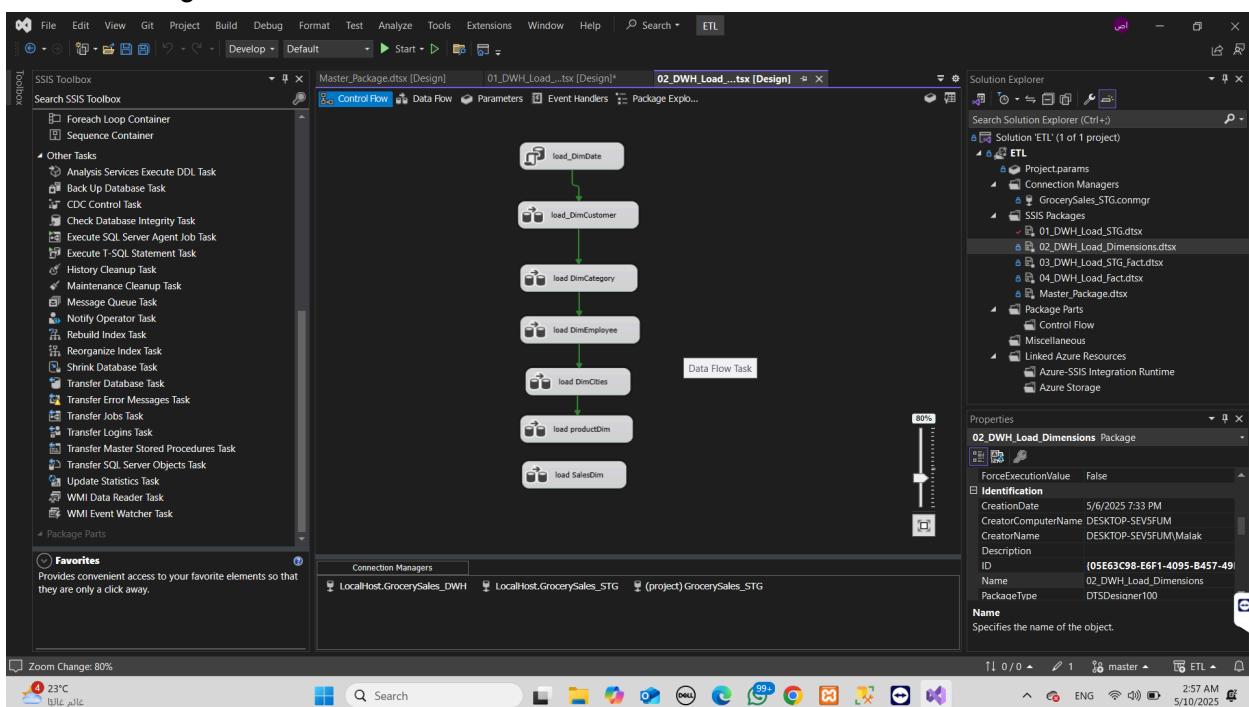
2. Package 1 Control Flow: Load tables to STG



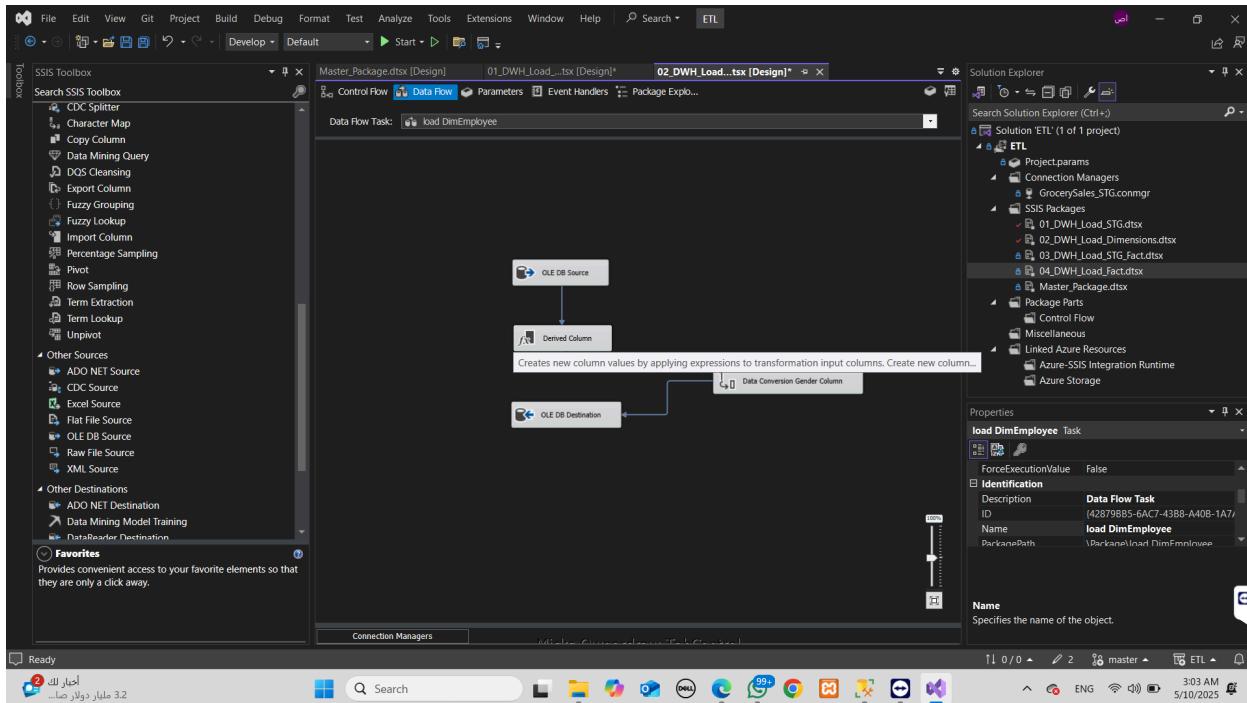
3. Package 1 Data Flow: Loading tables to STG (this flow applies for all tables - Derived Column for 'last extract date and source name' calculations)



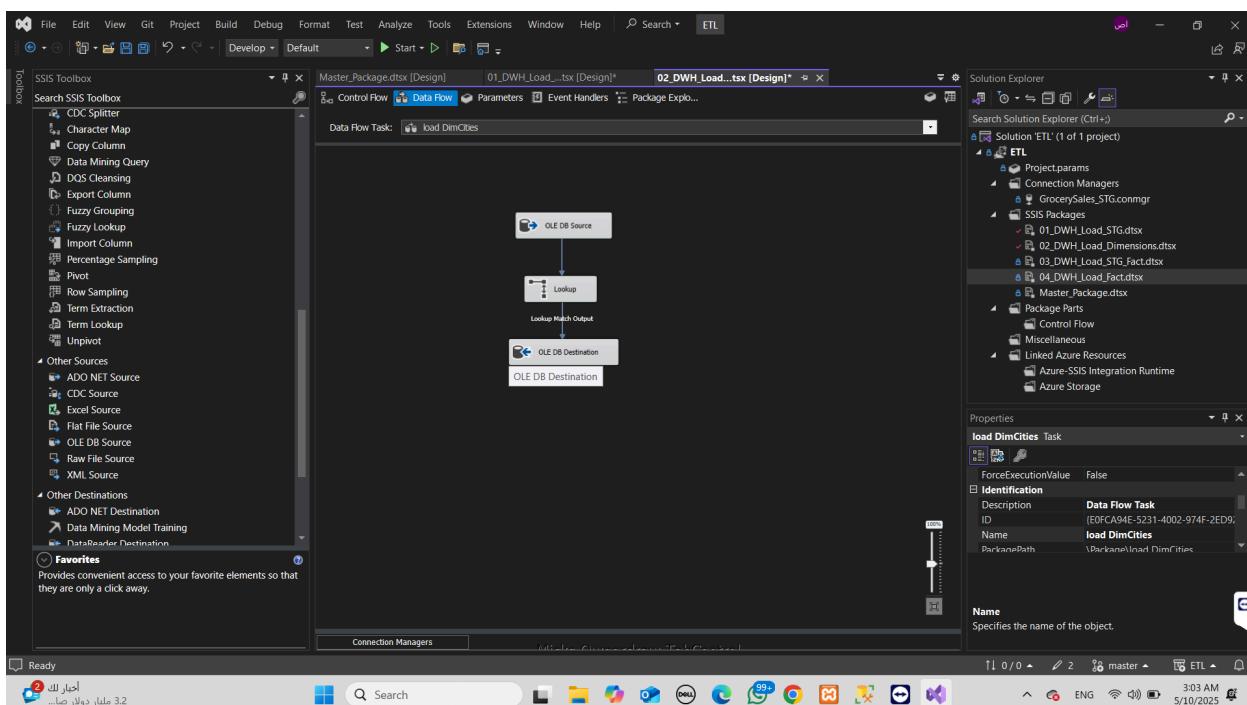
4. Package 2 Control Flow: Load STG Dimensions to DWH



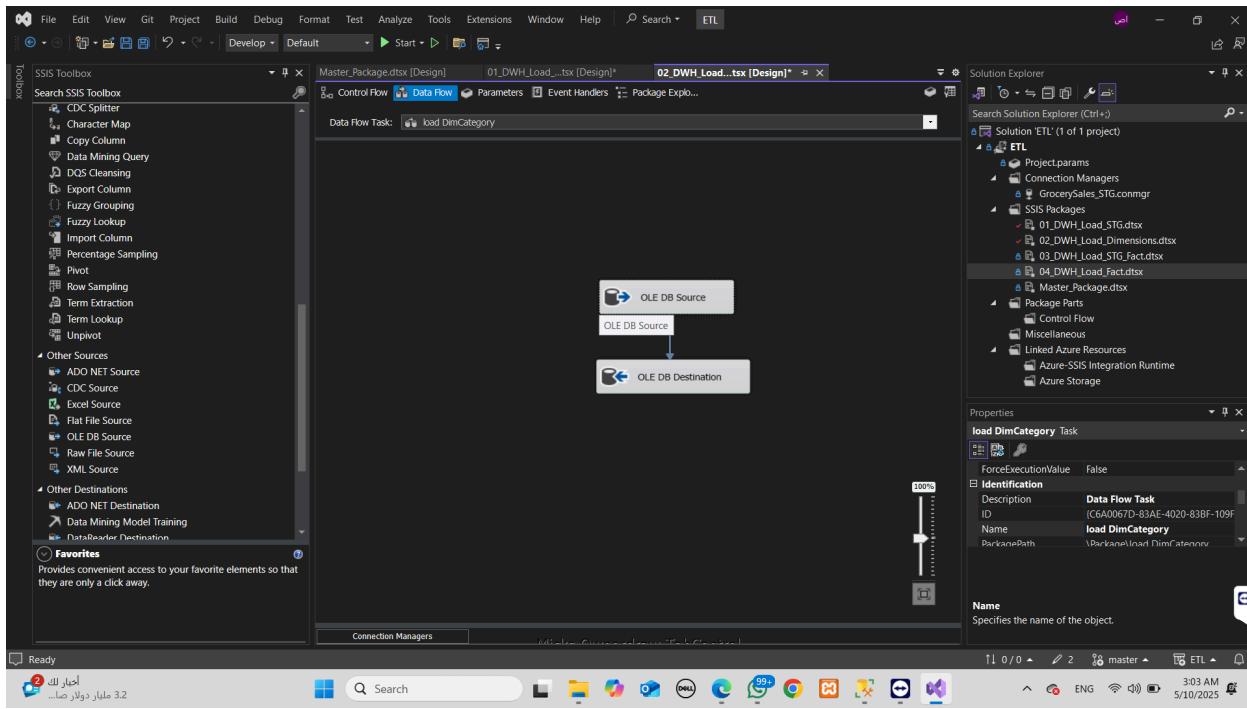
5. Package 2 Data Flow: Load STG Dimensions to DWH



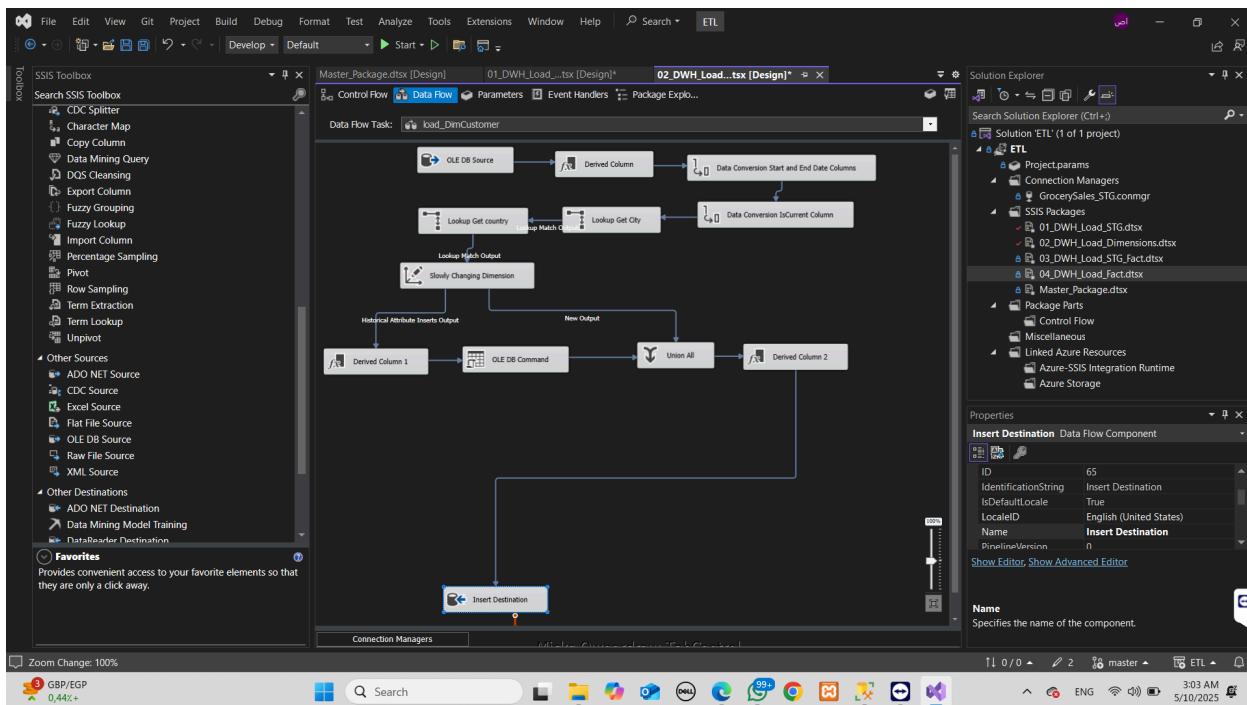
Load DimEmployee



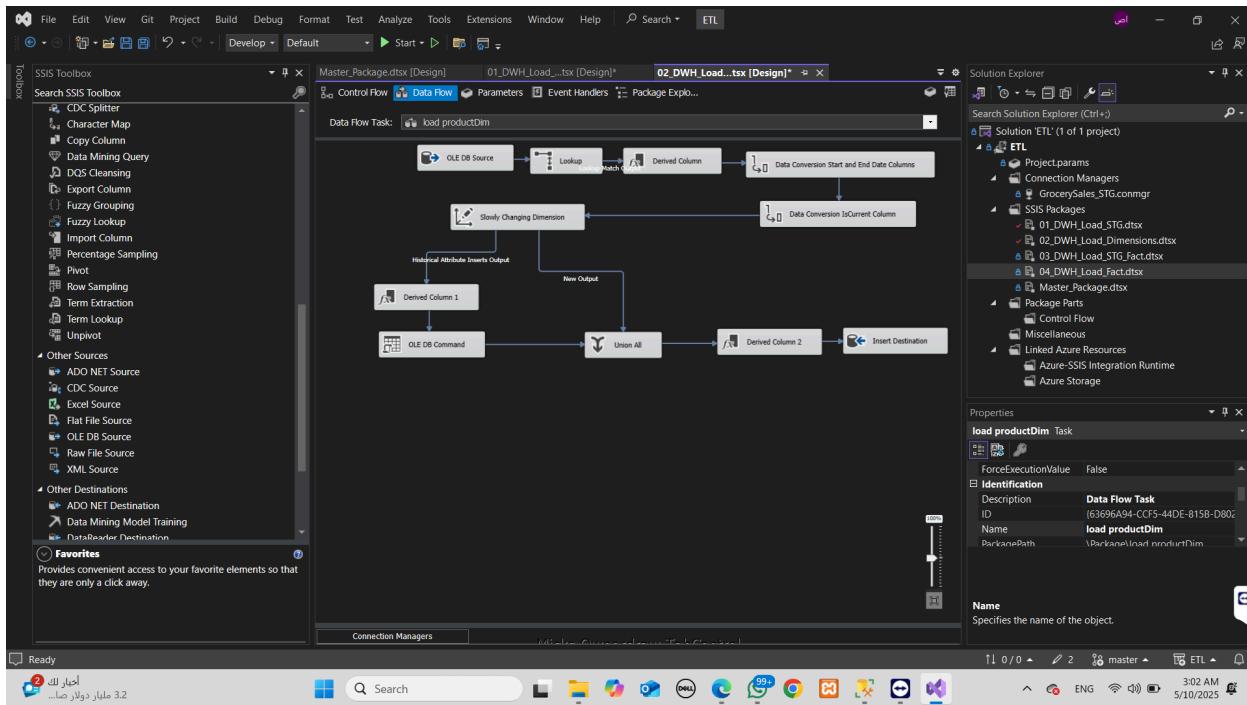
Load DimCities



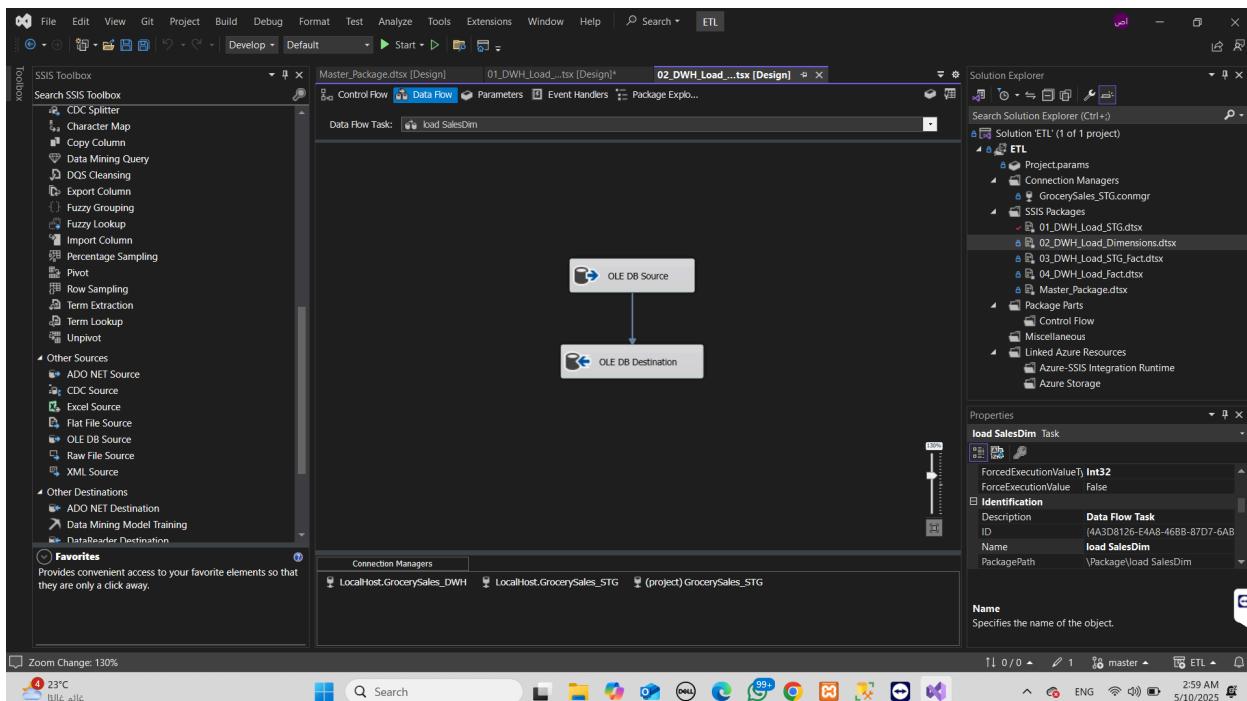
Load DimSales



Load DimCustomers (Slowly Changing Dimension)

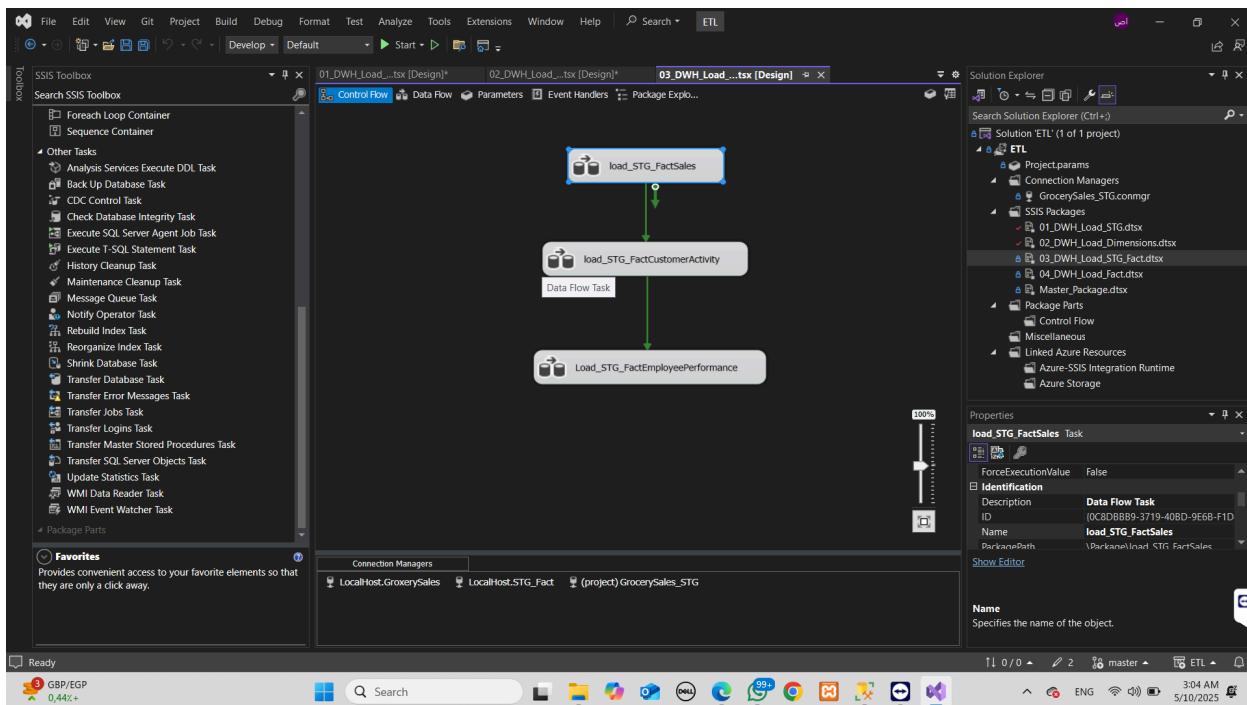


Load DimProducts (Slowly Changing Dimension)

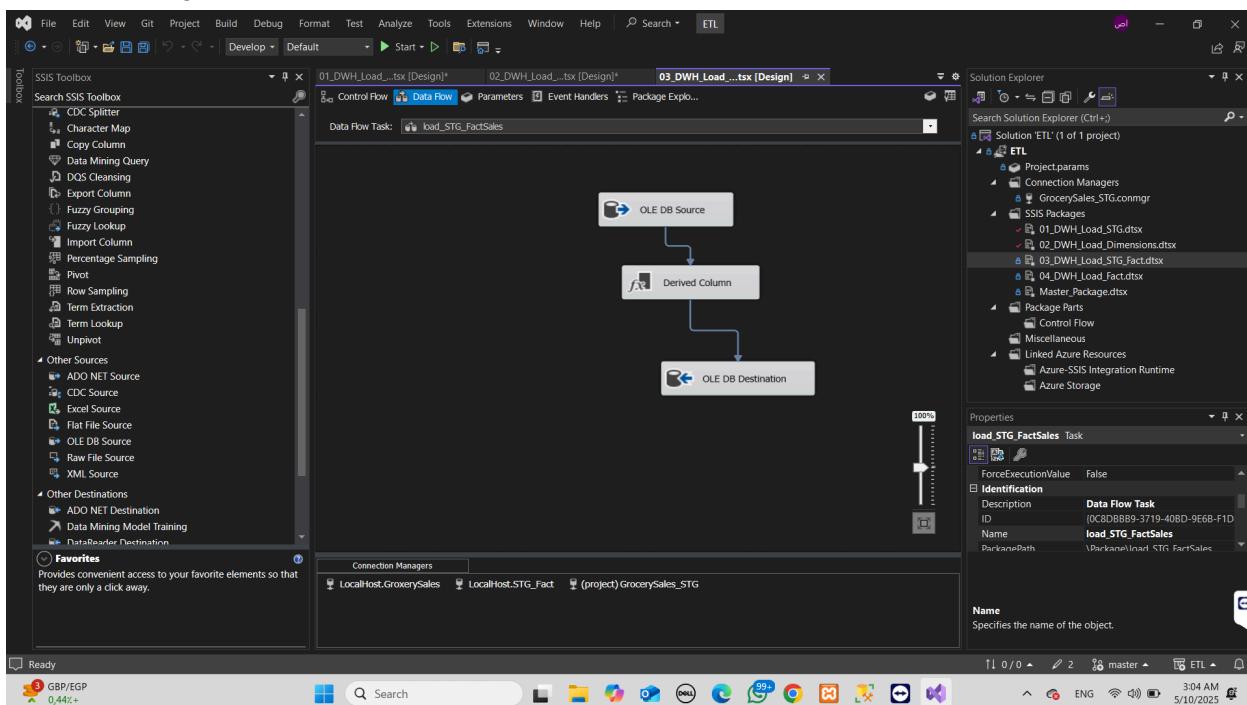


Load DimCategories

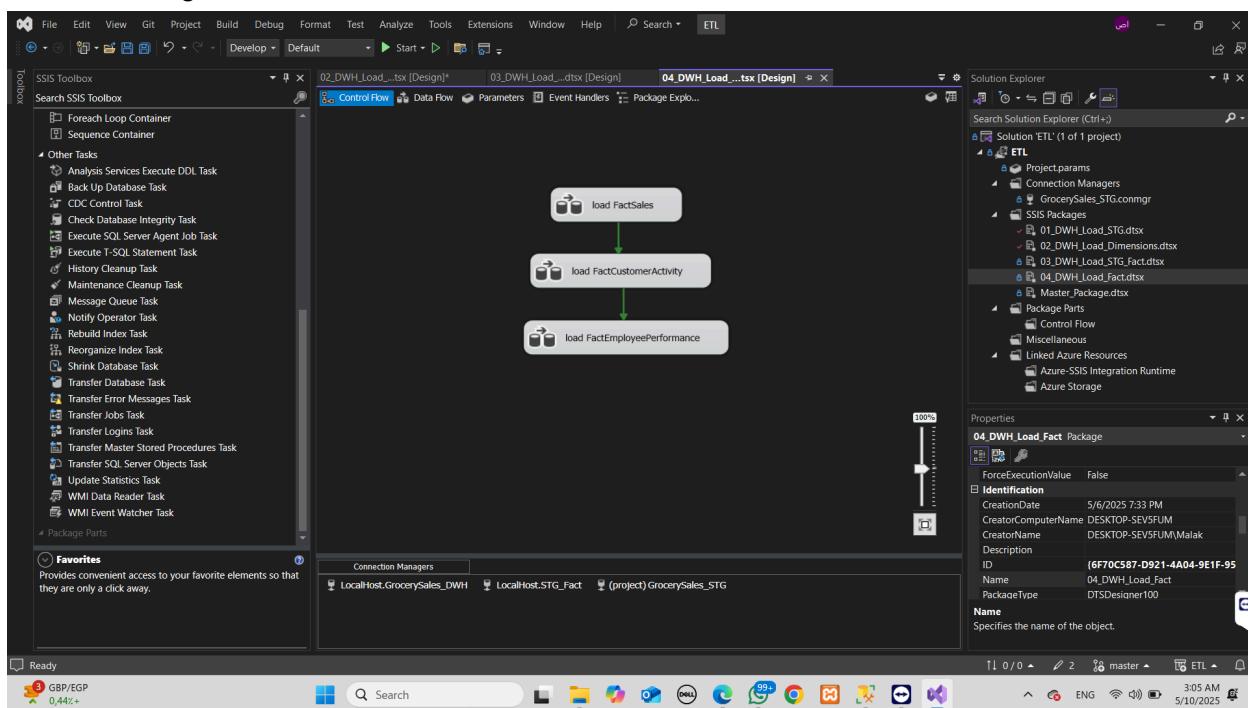
6. Package 3 Control Flow : Load Fact Tables to STG



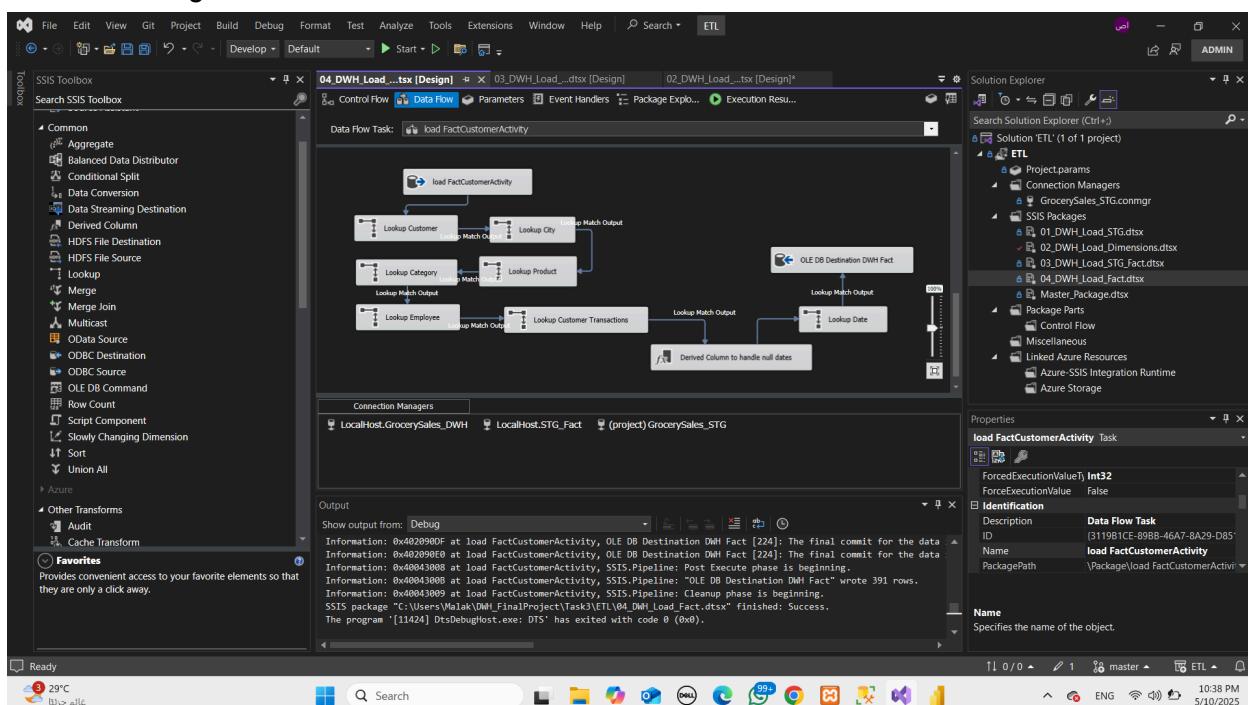
7. Package 3 Data Flow : Load Fact Tables to STG (all 3 Fact Tables same DataFlow)



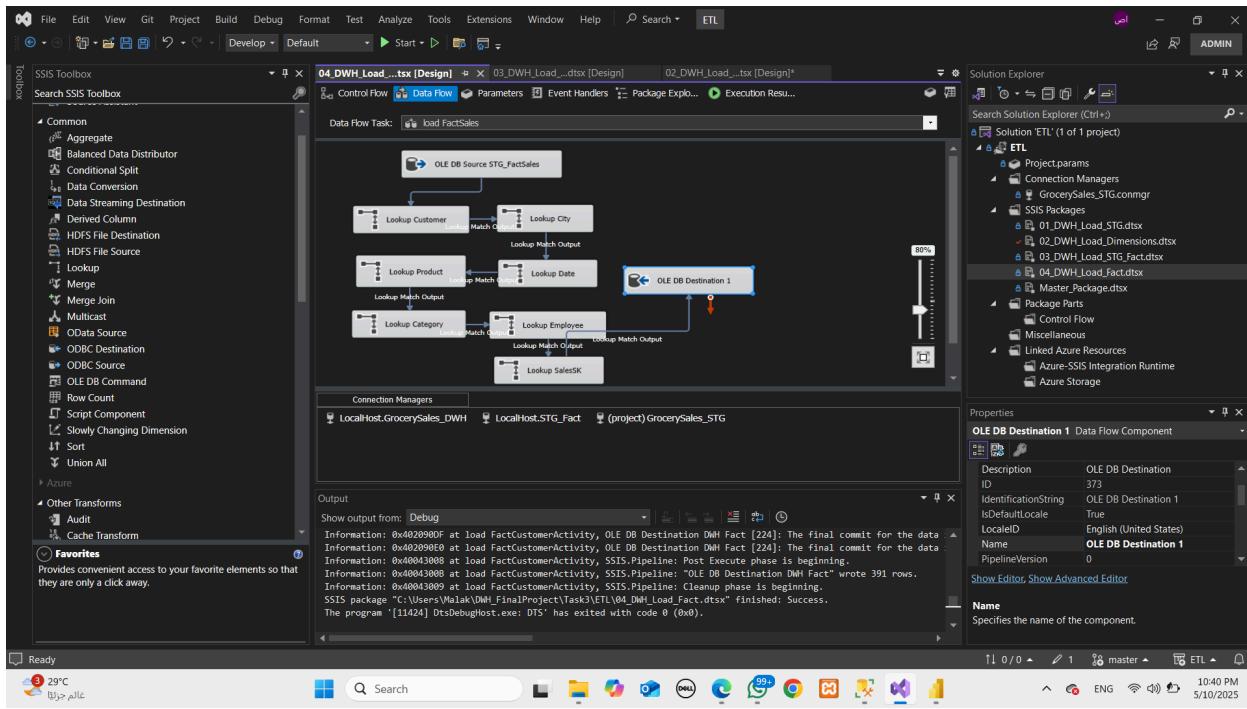
8. Package 4 Control Flow : Load Fact to DWH



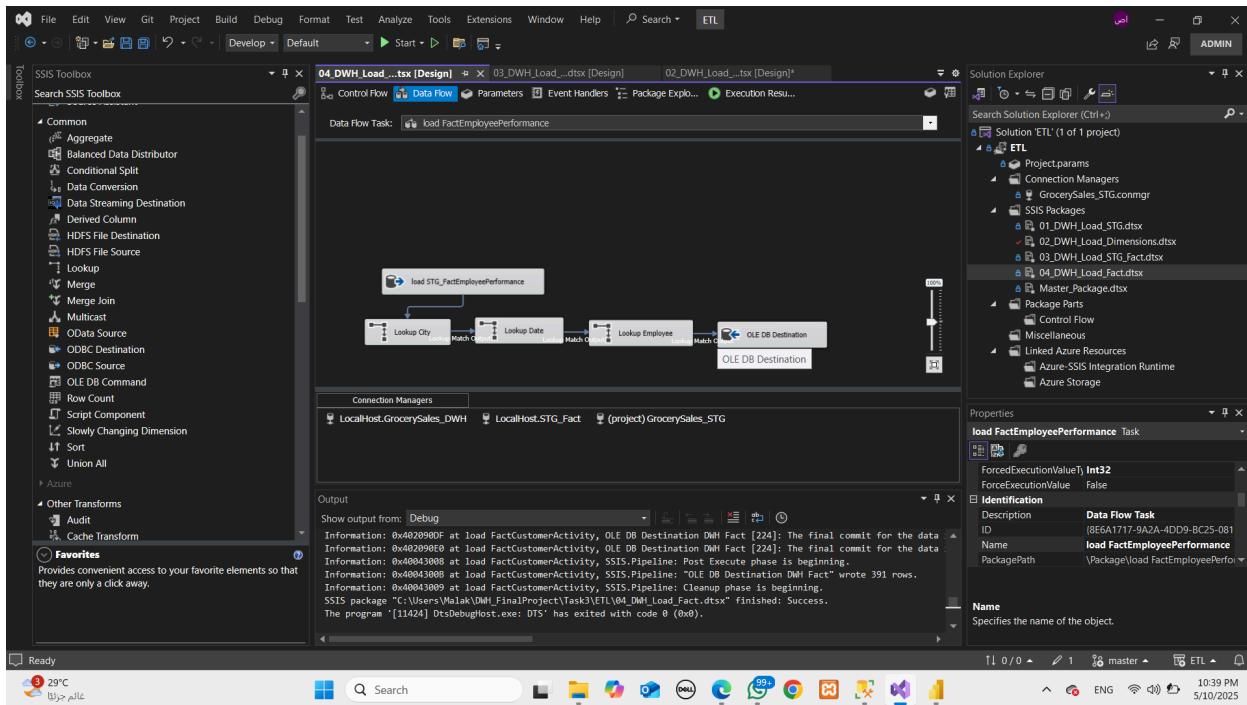
9. Package 4 Data Flow : Load Fact to DWH



Load FactCustomerActivity



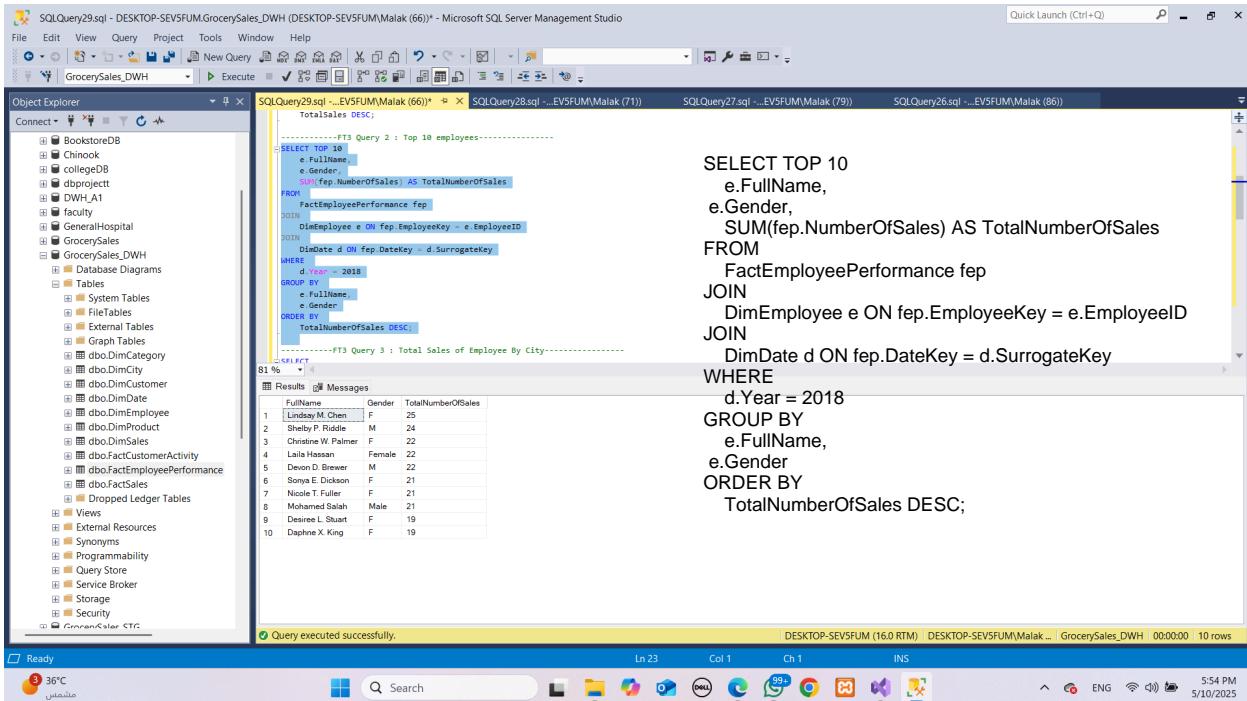
Load FactSales



Load FactEmployeePerformance

iv. Queries Screenshots

— FT(1) . FactEmployeePerformance Queries —



SQLQuery29.sql - DESKTOP-SEVSFUM.GrocerySales_DWH (DESKTOP-SEVSFUM\Malak (66)) - Microsoft SQL Server Management Studio

```

SELECT TOP 10
    e.FullName,
    e.Gender,
    SUM(fep.NumberOfSales) AS TotalNumberOfSales
FROM
    FactEmployeePerformance fep
JOIN
    DimEmployee e ON fep.EmployeeKey = e.EmployeeID
JOIN
    DimDate d ON fep.DateKey = d.SurrogateKey
WHERE
    d.Year = 2018
GROUP BY
    e.FullName,
    e.Gender
ORDER BY
    TotalNumberOfSales DESC;

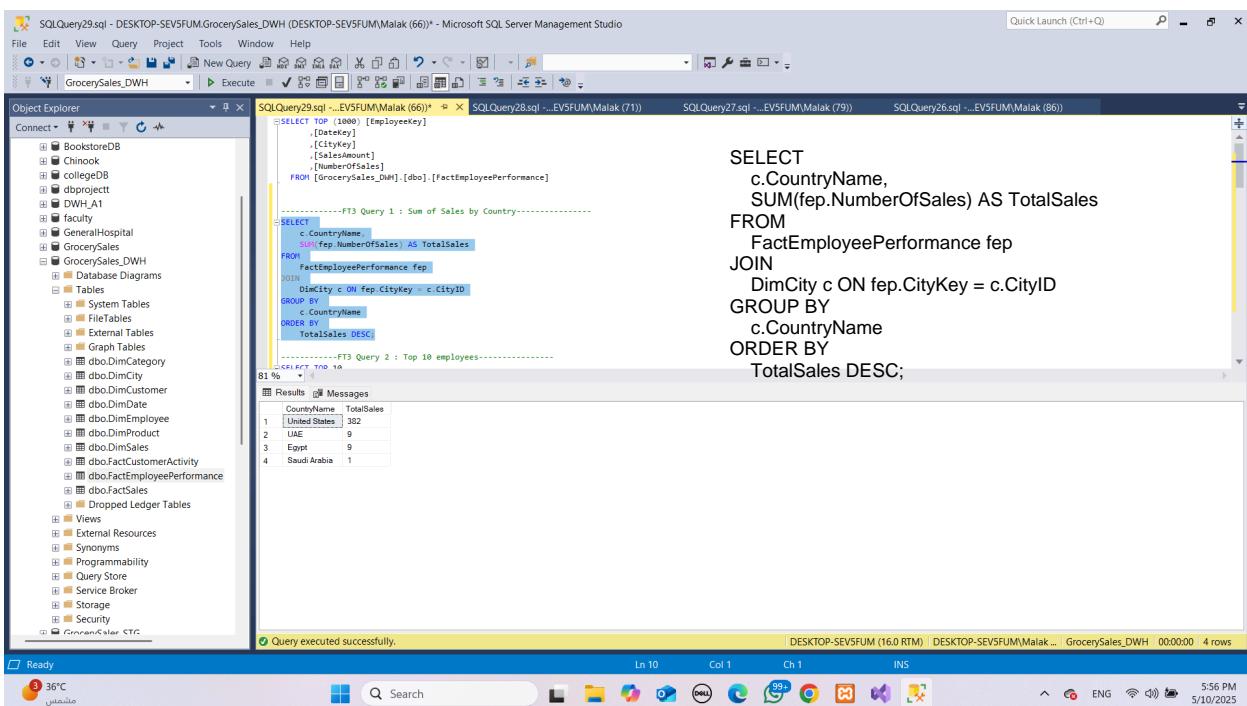
```

Results

	FullName	Gender	TotalNumberOfSales
1	Lindsay M. Chen	F	25
2	Shelby P. Riddle	M	24
3	Christine W. Palmer	F	22
4	Laila H. Hartman	Female	22
5	Dorothy D. Brewster	M	22
6	Sonja E. Dickson	F	21
7	Nicole T. Fuller	F	21
8	Mohamed Salah	Male	21
9	Desiree L. Stuart	F	19
10	Daphne L. King	F	19

Query executed successfully.

Query 1 : Top 10 Employees in Performance



SQLQuery29.sql - DESKTOP-SEVSFUM.GrocerySales_DWH (DESKTOP-SEVSFUM\Malak (66)) - Microsoft SQL Server Management Studio

```

SELECT TOP (1000) [EmployeeKey]
    ,c.CityKey
    ,[CityName]
    ,[SalesAmount]
    ,[NumberOfSales]
FROM [GrocerySales_DWH].[dbo].[FactEmployeePerformance]

```

Results

	EmployeeKey	CityKey	CityName	SalesAmount	NumberOfSales
1	1	1	United States	382	1
2	2	2	UAE	9	1
3	3	3	Egypt	9	1
4	4	4	Saudi Arabia	1	1

Query executed successfully.

Query 2 : Total Employees Performance across Countries

FactEmployeePerformanceQueries.sql - DESKTOP-SEVSFUM.GrocerySales_DWH (DESKTOP-SEVSFUM\Malak (66)) - Microsoft SQL Server Management Studio

```
--FT3 Query 3 : Total Sales of Employee By City-----
SELECT
    e.FullName,
    c.CityName,
    d.Year,
    d.Month,
    SUM(fep.NumberOfSales) AS TotalNumberOfSales
FROM
    FactEmployeePerformance fep
JOIN
    DimEmployee e ON fep.EmployeeKey = e.EmployeeID
JOIN
    DimDate d ON fep.DateKey = d.SurrogateKey
JOIN
    DimCity c ON fep.CityKey = c.CityID
WHERE
    d.Year = 2018
GROUP BY
    e.FullName,
    c.CityName

```

```
--FT3 Query 3 : Total Sales of Employee By City-----
SELECT
    e.FullName,
    c.CityName,
    d.Year,
    d.Month,
    SUM(fep.NumberOfSales) AS TotalNumberOfSales
FROM
    FactEmployeePerformance fep
JOIN
    DimEmployee e ON fep.EmployeeKey = e.EmployeeID
JOIN
    DimDate d ON fep.DateKey = d.SurrogateKey
JOIN
    DimCity c ON fep.CityKey = c.CityID
WHERE
    d.Year = 2018
GROUP BY
    e.FullName,
    c.CityName,
    d.Year,
    d.Month
ORDER BY
    e.FullName,
    d.Month;
```

Query executed successfully.

Query 3 : Total Employee Sales Performance in Cities for each Month

FactEmployeePerformanceQueries.sql - DESKTOP-SEVSFUM.GrocerySales_DWH (DESKTOP-SEVSFUM\Malak (66)) - Microsoft SQL Server Management Studio

```
--FT3 Query 4 : Monthly Sales Performance by Employee-----
SELECT
    ed.FullName,
    dd.Month,
    dd.Year,
    SUM(fep.SalesAmount) AS MonthlySales
FROM
    FactEmployeePerformance fep
JOIN
    DimEmployee ed ON fep.EmployeeKey = ed.EmployeeID
JOIN
    DimDate dd ON fep.DateKey = dd.SurrogateKey
GROUP BY
    ed.FullName, dd.Month, dd.Year
ORDER BY
    dd.Year, dd.Month, MonthlySales DESC;
```

```
--FT3 Query 4 : Monthly Sales Performance by Employee-----
SELECT
    ed.FullName,
    dd.Month,
    dd.Year,
    SUM(fep.SalesAmount) AS MonthlySales
FROM
    FactEmployeePerformance fep
JOIN
    DimEmployee ed ON fep.EmployeeKey = ed.EmployeeID
JOIN
    DimDate dd ON fep.DateKey = dd.SurrogateKey
GROUP BY
    ed.FullName, dd.Month, dd.Year
ORDER BY
    dd.Year, dd.Month, MonthlySales DESC;
```

Query executed successfully.

Query 4 : Monthly Employees Sales Performance

— FT(2) . FactSales Queries —

The screenshot shows the Microsoft SQL Server Management Studio interface with two queries open in tabs:

- Query 1 (Tab 1):** This query retrieves the total quantity sold for each category. It uses a subquery to get the top 100 sales and then joins it with the DimCategory table to sum the quantities.
- Query 2 (Tab 2):** This query retrieves the total quantity sold by each employee in each city for March 2018. It joins FactSales with DimCity and DimEmployee tables, filters for the month and year, and groups by city and employee.

```

-- Query 1: Total Quantity for Categories
SELECT
    cat.CategoryName,
    SUM(fs.Quantity) AS Quantity
FROM
    FactSales fs
JOIN
    DimCategory cat ON fs.CategoryKey = cat.CategoryID
GROUP BY
    cat.CategoryName
ORDER BY
    Quantity DESC;

-- Query 2: Total Quantity Sold by Employee per City
SELECT
    c.CityName,
    e.FullName,
    d.FullDate,
    SUM(fs.Quantity) AS TotalQuantitySold
FROM
    FactSales fs
JOIN
    DimCity c ON fs.CityKey = c.CityID
JOIN
    DimDate d ON fs.DateKey = d.SurrogateKey
JOIN
    DimEmployee e ON fs.EmployeeKey = e.EmployeeID
WHERE
    d.Month = 3 AND d.Year = 2018
GROUP BY
    c.CityName,
    e.FullName;
  
```

Query 1 : Total Quantity for Categories

The screenshot shows the results of Query 2, which lists employees and their total quantity sold in specific cities during March 2018. The results are ordered by TotalQuantitySold in descending order.

CityName	FullName	FullDate	TotalQuantitySold
Albuquerque	Sonya E. Dickson	2018-03-03	1
Anchorage	Desiree L. Stuart	2018-03-20	1
Anchorage	Chadwick U. Walton	2018-03-30	1
Arlington	Holly E. Collins	2018-03-31	1
Aurora	Shelby P. Riddle	2018-03-22	1
Birmingham	Daphne X. King	2018-03-21	1
Birmingham	Barbara G. Hart	2018-03-19	1
Buffalo	Lindsay M. Chen	2018-03-04	1
Buffalo	Seeth D. Franco	2018-03-17	1
Cairo	Katina Y. Marks	2018-03-10	1
Cairo	Katina Y. Marks	2018-03-13	1
Charlotte	Lindsay M. Chen	2018-03-20	1
Charlotte	Devon D. Brewster	2018-03-29	1
Cincinnati	Shelby P. Riddle	2018-03-07	1
Columbus	Devon D. Brewster	2018-03-27	1
Dayton	Katina Y. Marks	2018-03-10	1
Dayton	Katina Y. Marks	2018-03-13	1
Dubai	Lindsay M. Chen	2018-03-04	1

Query 2 : Total Quantity sold by each Employee in each City with specific Dates

```
SELECT
    c.FullName,
    c.Country,
    cat.CategoryName,
    p.ProductName,
    SUM(fs.Quantity) AS TotalQuantity
FROM
    FactSales fs
JOIN
    DimCity city ON fs.CityKey = city.CityID
JOIN
    DimDate d ON fs.DateKey = d.SurrogateKey
JOIN
    DimProduct p ON fs.ProductKey = p.ProductID
JOIN
    DimCategory cat ON fs.CategoryKey = cat.CategoryID
```

```
SELECT
    c.FullName,
    c.Country,
    cat.CategoryName,
    p.ProductName,
    SUM(fs.Quantity) AS TotalQuantity
FROM
    FactSales fs
JOIN
    DimCity city ON fs.CityKey = city.CityID
JOIN
    DimDate d ON fs.DateKey = d.SurrogateKey
WHERE
    d.Month IN (1, 2, 3, 4) AND d.Year = 2018
GROUP BY
    cat.CategoryName,
    p.ProductName,
    c.FullName,
    c.Country
ORDER BY
    cat.CategoryName,
```

Query executed successfully.

Query 3 : Total Quantity with Grouping Customers, Countries, Categories and Products

```
SELECT
    c.Country,
    d.FullDate,
    SUM(fs.Quantity) AS TotalQuantity
FROM
    FactSales fs
JOIN
    DimCity city ON fs.CityKey = city.CityID
JOIN
    DimDate d ON fs.DateKey = d.SurrogateKey
JOIN
    DimCustomer c ON fs.CustomerKey = c.CustomerID
WHERE
    d.Month IN (1, 2, 3, 4) AND d.Year = 2018
GROUP BY
    c.Country,
    d.FullDate
ORDER BY
    c.Country,
```

```
SELECT
    c.Country,
    d.FullDate,
    SUM(fs.Quantity) AS TotalQuantity
FROM
    FactSales fs
JOIN
    DimCity city ON fs.CityKey = city.CityID
JOIN
    DimDate d ON fs.DateKey = d.SurrogateKey
JOIN
    DimCustomer c ON fs.CustomerKey = c.CustomerID
WHERE
    d.Month IN (1, 2, 3, 4) AND d.Year = 2018
GROUP BY
    c.Country,
    d.FullDate
ORDER BY
    c.Country,
    d.FullDate;
```

Query executed successfully.

Query 4 : Total Quantity sold for Countries Daily

— FT(3) . FactCustomerActivity Queries —

SQLQuery31.sql - DESKTOP-SEVFUM.GrocerySales_DWH (DESKTOP-SEVFUM\Malak (87)) - Microsoft SQL Server Management Studio

```

SELECT
    d.Month,
    p.ProductName,
    SUM(fca.NumberOfTransactions) AS TotalTransactions
FROM
    FactCustomerActivity fca
JOIN
    DimProduct p ON fca.ProductKey = p.ProductID
JOIN
    DimDate d ON fca.DateKey = d.SurrogateKey
GROUP BY
    d.Month, p.ProductName
ORDER BY
    d.Month, TotalTransactions DESC;

```

Results

Month	ProductName	TotalTransactions
1	Milk - 2%	3
1	Bay Leaf	2
1	Wine - White, Colaba Crash	2
1	Wine - Red, Schneiders And Schyl	1
1	Wine - Woodham Estate Bin 777	1
1	Yogurt - Blueberry, 175 Gr	1
1	Wine - Cotes Hermitage E	1
1	Wine - Red, Harrow Estates, Cab	1
1	Wine - Two Oceans Cabernet	1
1	Wine - Toasted Head	1
1	Wine - Vidal Lewine Magnotta	1
1	Tart Shells - Sweet, 4	1
1	Tea - Decaf Lipton	1
1	Tea - Herbal Sweet Dreams	1
1	Tea - Jasmine Green	1
1	Tomatoes Tear Drop	1
1	Unlabelled - Dried	1

Query executed successfully.

Query 1 : Monthly Product Popularity

SQLQuery31.sql - DESKTOP-SEVFUM.GrocerySales_DWH (DESKTOP-SEVFUM\Malak (87)) - Microsoft SQL Server Management Studio

```

SELECT TOP 5
    city.CityName,
    SUM(fca.TotalSpent) AS TotalCustomerSpent
FROM
    FactCustomerActivity fca
JOIN
    DimCity city ON fca.CityKey = city.CityID
JOIN
    DimDate d ON fca.DateKey = d.SurrogateKey
WHERE
    d.Year = 2018
GROUP BY
    city.CityName
ORDER BY
    TotalCustomerSpent DESC;

```

Results

CityName	TotalCustomerSpent
Chicago	200.00
Riyadh	200.00
Cairo	150.00
Dayton	150.00
Buffalo	75.00

Query executed successfully.

Query 2 : Top 5 Cities with Highest Customer Spending in 2018

SQLQuery31.sql - DESKTOP-SEVFUM.GrocerySales_DWH (DESKTOP-SEVFUM\Malak (87)) - Microsoft SQL Server Management Studio

```

SELECT
    c.FullName AS CustomerName,
    cat.CategoryName,
    d.FullDate,
    SUM(fca.NumberOfTransactions) AS TotalTransactions
FROM
    FactCustomerActivity fca
JOIN
    DimCustomer c ON fca.CustomerKey = c.CustomerID
JOIN
    DimCategory cat ON fca.CategoryKey = cat.CategoryID
JOIN
    DimDate d ON fca.DateKey = d.SurrogateKey
WHERE
    d.Year = 2018 AND d.Quarter = 1
GROUP BY
    c.FullName, cat.CategoryName, d.FullDate
ORDER BY
    cat.CategoryName,
    TotalTransactions DESC;

```

Results Messages

CustomerName	CategoryName	FullDate	TotalTransactions
Andrew S. Smith	Bakery	2018-01-04	1
Angela A. Burns	Bakery	2018-01-08	1
Arlene K. Jewelry	Bakery	2018-01-04	1
Audrey O. Parks	Bakery	2018-03-21	1
Billie D. Fischer	Bakery	2018-01-28	1
Bob Smith	Bakery	2018-01-19	1
Bobbie B. Simon	Bakery	2018-01-16	1
Charity T. Odorn	Bakery	2018-01-19	1
Cody H. Dunn	Bakery	2018-01-15	1
Darrick H. Kirby	Bakery	2018-01-16	1
Edward B. Ferrell	Bakery	2018-03-21	1
Grace M. Vincent	Bakery	2018-03-04	1
Gwendolyn B. Love	Bakery	2018-03-02	1
Holly C. Wong	Bakery	2018-03-27	1
Jessie L. Humphrey	Bakery	2018-03-26	1
Jacob S. Webster	Bakery	2018-02-11	1
Karen Y. Parker	Bakery	2018-03-11	1

Query executed successfully.

Query 3 : Most Active Customers by Categories in 2018 first Quarter

SQLQuery31.sql - DESKTOP-SEVFUM.GrocerySales_DWH (DESKTOP-SEVFUM\Malak (87)) - Microsoft SQL Server Management Studio

```

SELECT
    pd.ProductName,
    SUM(fca.NumberOfTransactions) AS TotalTransactions
FROM
    FactCustomerActivity fca
JOIN
    DimProduct pd ON fca.ProductKey = pd.ProductID
GROUP BY
    pd.ProductName
ORDER BY
    pd.ProductName,
    TotalTransactions DESC;

```

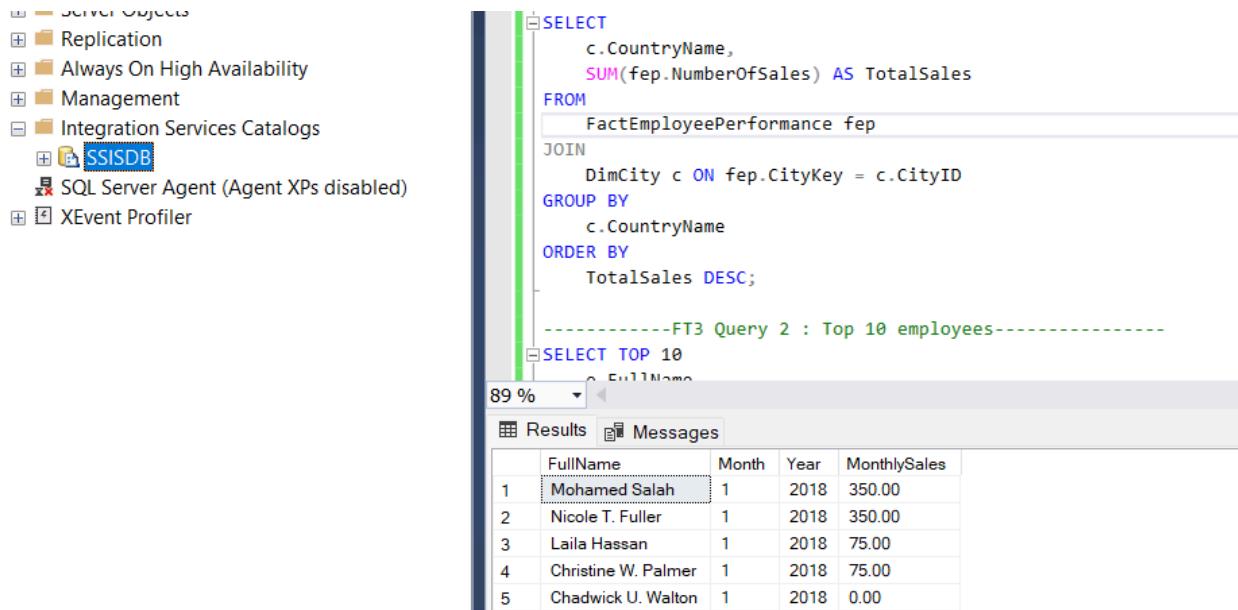
Results Messages

ProductName	TotalTransactions
Bread Crumbs - Japanese Style	5
Cake - Box Window 10x10x2.5	4
Cookies - Assorted	4
Dc - Frozen Momji	4
Milk - 2%	4
Lettuce - Spring Mix	3
Ketchup - Tomato	3
Flour - Pasty	3
Guinea Fowl	3
Catfish Head	3
Bay Leaf	3
Popcorn - Paprika, Hungarian	3
Placemat - Scallops, White	3
Pork - Neck And Feet Attached	3
Sword Pick Asst	3
Tart Shells - Sweet	3
Meat - Assorted, Unknown	2

Query executed successfully.

Query 4 : Most Products bought by Customers

v. The deployed packages in SSIS with their schedule Screenshots



The screenshot shows the SQL Server Management Studio (SSMS) interface. On the left, the Object Explorer displays various database objects like Replication, Always On High Availability, Management, Integration Services Catalogs, and the SSISDB catalog. In the center, a query window contains the following T-SQL code:

```

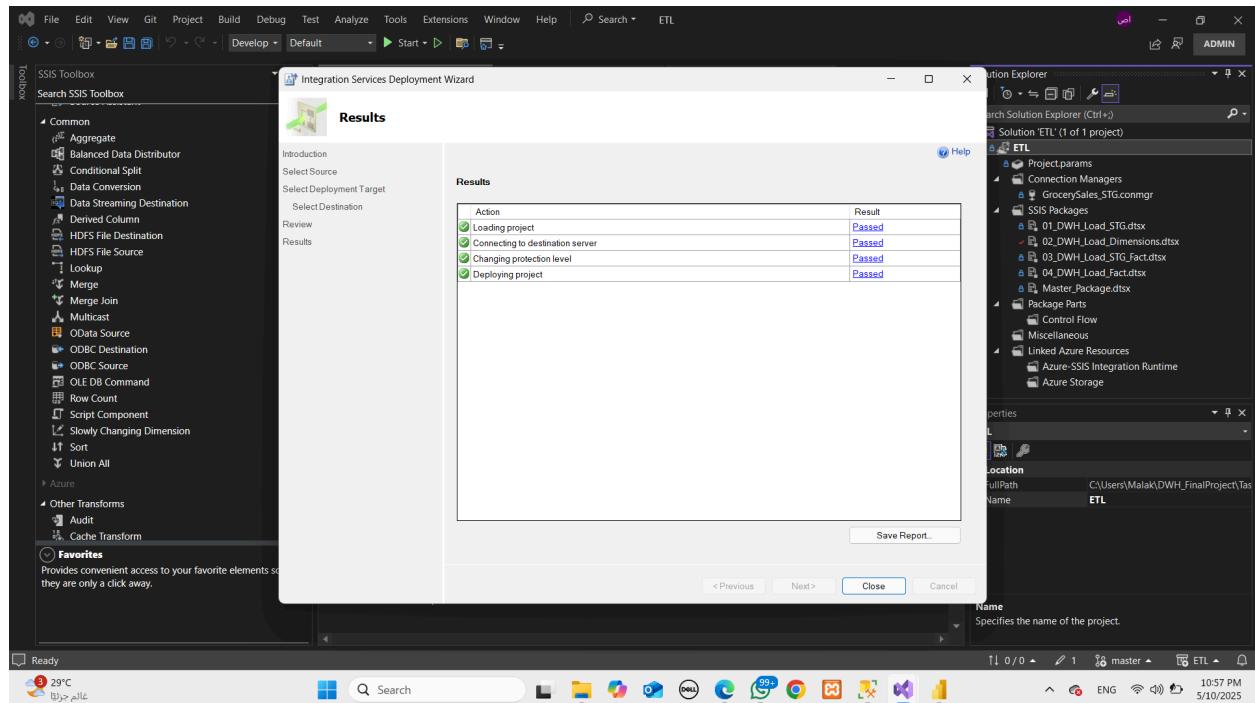
SELECT
    c.CountryName,
    SUM(fep.NumberOfSales) AS TotalSales
FROM
    FactEmployeePerformance fep
JOIN
    DimCity c ON fep.CityKey = c.CityID
GROUP BY
    c.CountryName
ORDER BY
    TotalSales DESC;
-----FT3 Query 2 : Top 10 employees-----
SELECT TOP 10
    FullName

```

Below the query window, the Results tab shows a table with the following data:

	FullName	Month	Year	MonthlySales
1	Mohamed Salah	1	2018	350.00
2	Nicole T. Fuller	1	2018	350.00
3	Laila Hassan	1	2018	75.00
4	Christine W. Palmer	1	2018	75.00
5	Chadwick U. Walton	1	2018	0.00

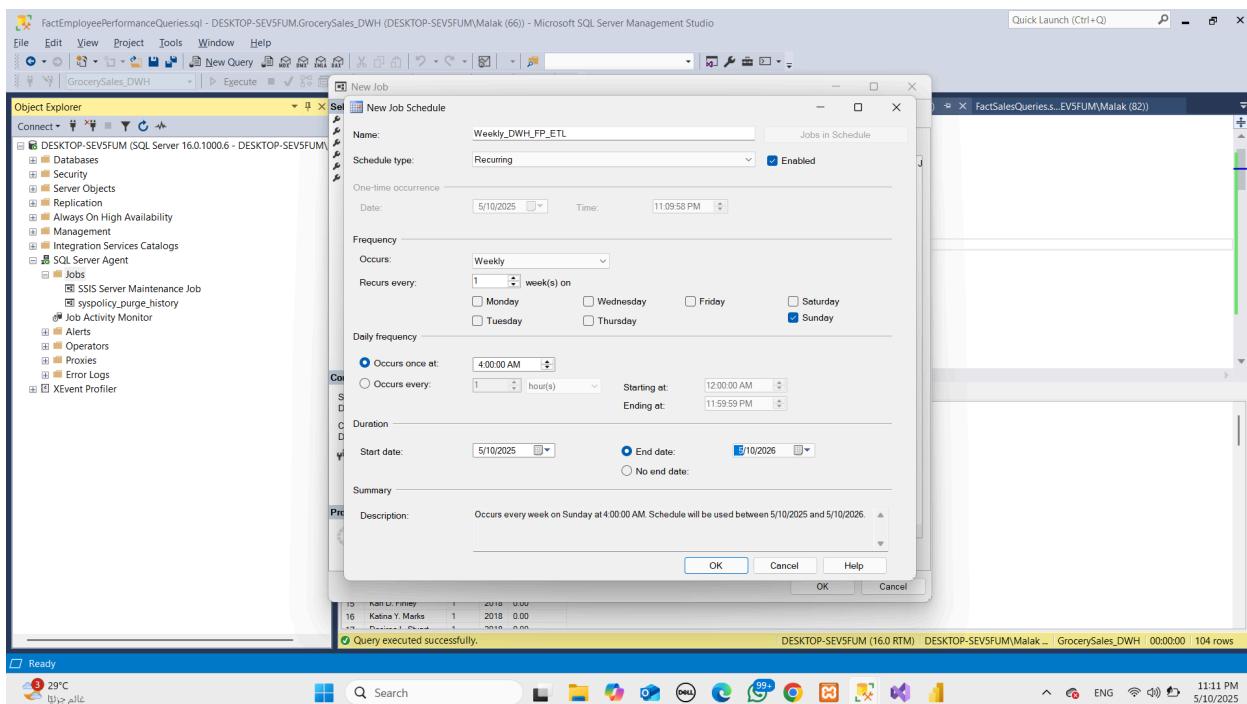
1. Created an ETL file in SSISDB in SSMS



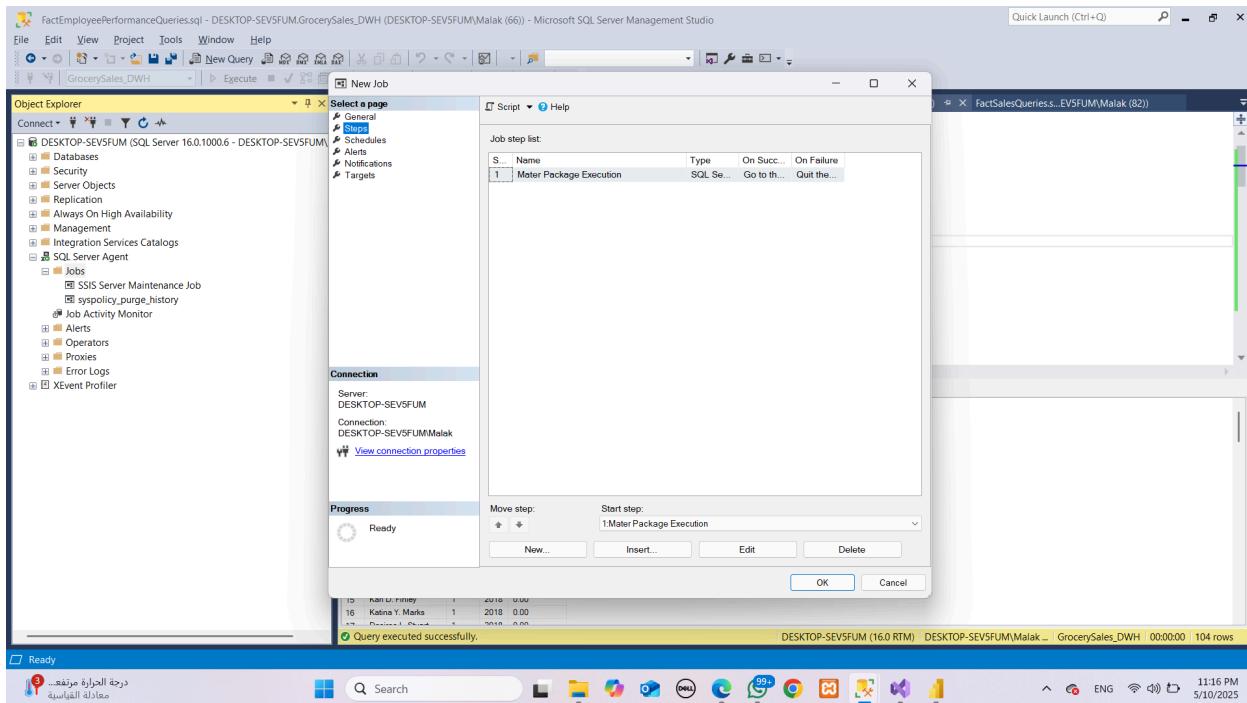
2. Deployed Project from Solution Explorer in MVS

The screenshot shows the Microsoft SQL Server Management Studio (SSMS) interface. In the Object Explorer, there is a node for 'GrocerySales_DWH' under 'Integration Services Catalogs'. The 'SSISDB' database contains several packages: '01_DWH_Load_STG.dtsx', '02_DWH_Load_Dimensions.dtsx', '03_DWH_Load_STG_Facts.dtsx', '04_DWH_Load_Fact.dtsx', and 'Master_Package.dtsx'. A query window titled 'FactCustomerActiv... VSFUM\Malak (87)' is open, displaying two queries. The first query, 'FT3 Query 1 : Sum of Sales by Country', selects the country name and sum of sales from the 'FactEmployeePerformance' table. The second query, 'FT3 Query 2 : Top 10 employees', selects the top 10 employees from the same table. The results grid shows 104 rows of employee data.

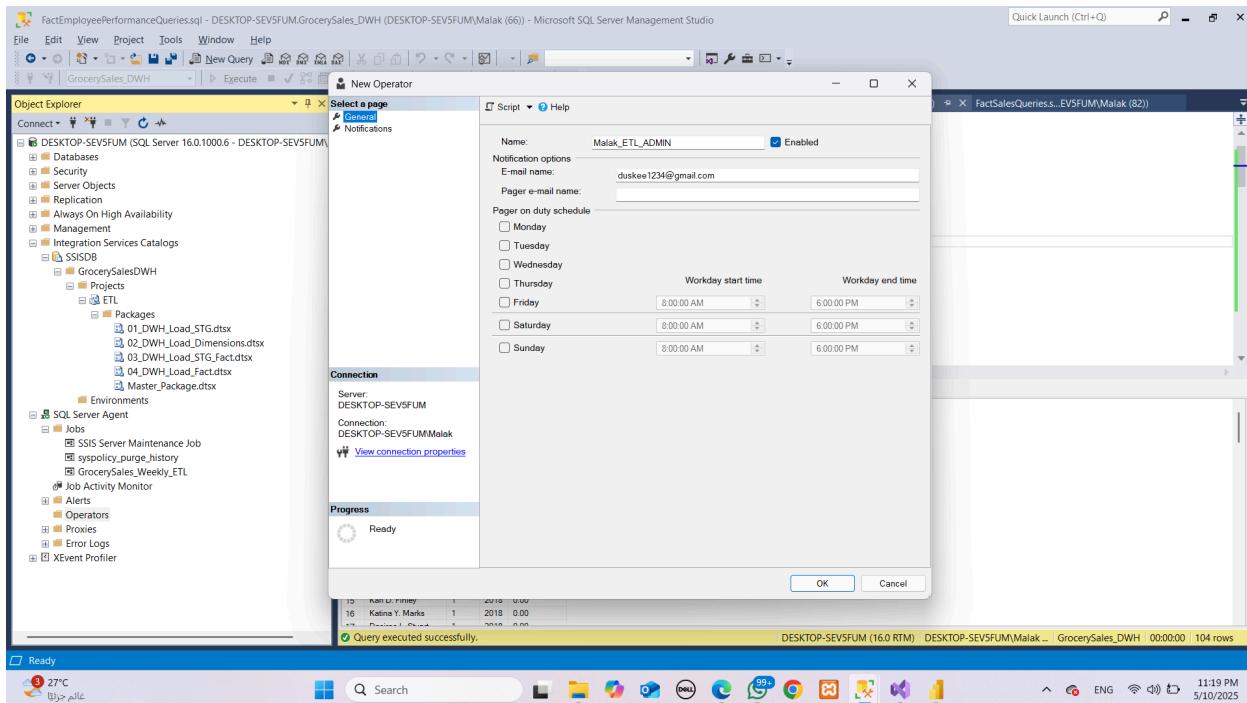
3. Packages Found Deployed in Integration Services in SSMS



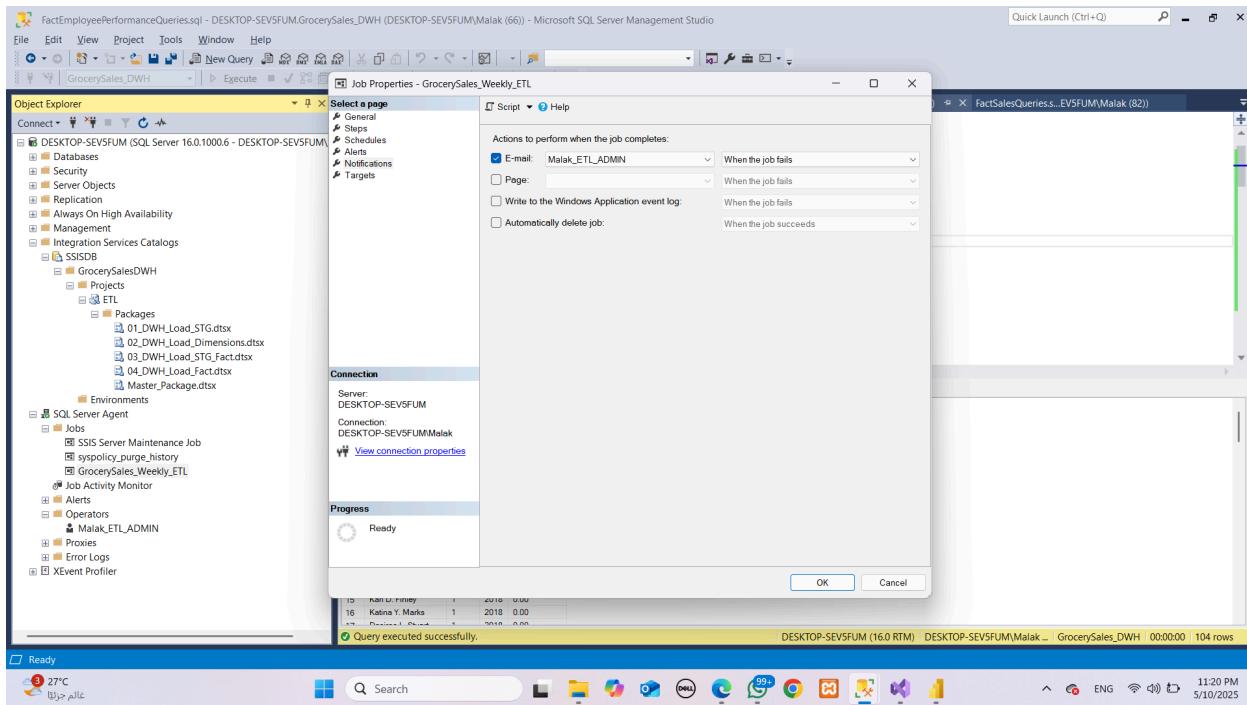
4. Scheduled New Job from SQL Server Agent



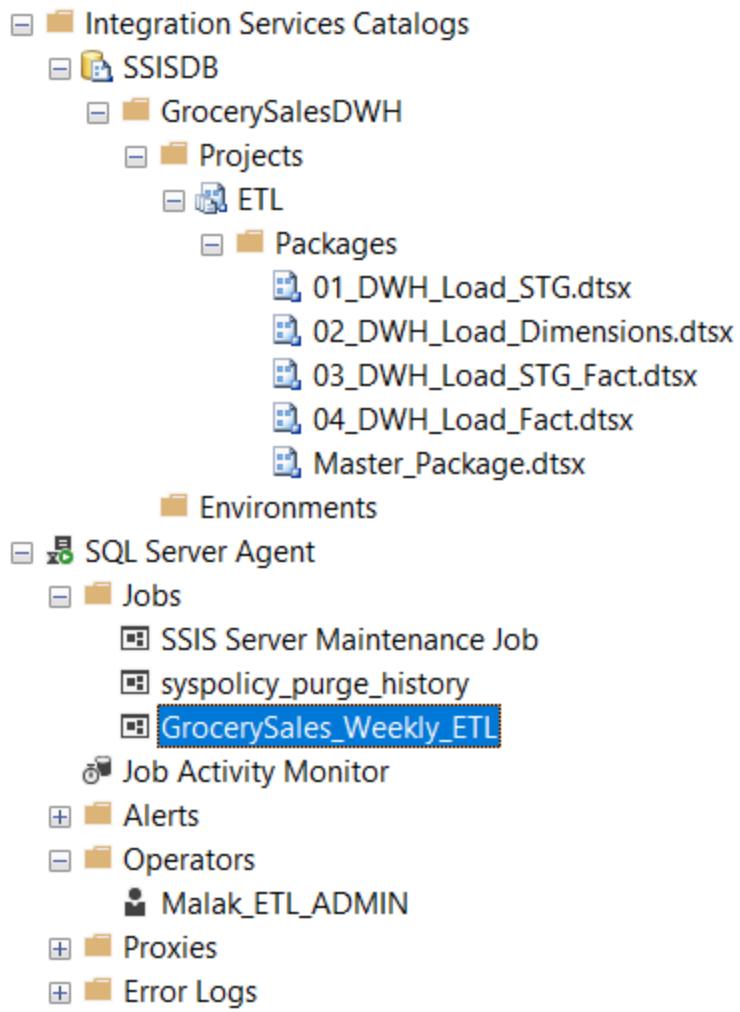
5. Assign Sequence Steps (we already deployed a master package containing the sequence of package execution so we only will schedule the master package)



6. Add an Operator inside SQL Server Agent with my email and named "Malak_ETL_Admin"



7. Add operator in job “GrocerySales_Weekly_ETL” to get email notification on job



8. Final Deployed Packages and Weekly Job with E-mail Notification

vi. Power BI Dashboard

