

**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH  
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN  
KHOA HỆ THÔNG THÔNG TIN**



**BÁO CÁO ĐỒ ÁN THỰC HÀNH MÔN HỌC  
KHO DỮ LIỆU VÀ OLAP**

**ĐỀ TÀI  
XÂY DỰNG KHO DỮ LIỆU  
GLOBAL FASHION RETAIL SALES**

Giảng viên hướng dẫn: ThS. Nguyễn Thị Kim Phụng

Lớp: Kho dữ liệu và OLAP – IS217.Q14

Nhóm sinh viên thực hiện:

1. Trần Đại Hải                          MSSV: 23520420
2. Mai Nguyễn Bảo Duy    MSSV: 23520369

**TP. Hồ Chí Minh, Năm 2025**

## LỜI CẢM ƠN

Kính gửi cô Nguyễn Thị Kim Phụng

Lời đầu thư, nhóm chúng xin được bày tỏ lòng biết ơn chân thành và sâu sắc nhất đến cô và thầy. Trong suốt chặng đường học tập môn “Kho dữ liệu và OLAP” và đặc biệt là quá trình thực hiện đồ án “Xây dựng kho dữ liệu Global Fashion Retail Sales”, cô đã không ngừng dành trọn tâm huyết, sự tận tâm và những chỉ dẫn quý báu để dùi dắt chúng em. Sự hướng dẫn nhiệt tình và chuyên sâu của cô không chỉ đơn thuần là việc truyền đạt kiến thức, mà còn là nguồn cảm hứng lớn lao giúp chúng em khám phá và đào sâu vào những khía cạnh phức tạp của ngành. Nhờ có cô, những khái niệm tưởng chừng khô khan đã trở nên dễ hiểu, những bài toán khó khăn đã được tháo gỡ, từ đó trang bị cho chúng em vốn tri thức vô cùng quý giá và sự tự tin để tự mình vận dụng vào bài báo cáo đồ án một cách hiệu quả nhất. Chúng em xin hứa sẽ luôn ghi nhớ và trân trọng những nền tảng vững chắc mà cô và thầy đã dày công vun đắp.

Bài báo cáo này không chỉ là một sản phẩm học thuật, mà còn là minh chứng cho sự nỗ lực không ngừng nghỉ của cả nhóm trong việc học hỏi và ứng dụng những kiến thức quý báu đó. Dù đây có thể chỉ là một thành quả nhỏ bé, nhưng chúng em hy vọng rằng nó phần nào sẽ thể hiện được sự trân trọng và lòng biết ơn sâu sắc đối với công sức, tâm huyết và sự tận tình mà cô đã dành cho chúng em.

Một lần nữa, nhóm chúng em xin kính chúc cô luôn dồi dào sức khỏe, gặt hái thêm nhiều thành công rực rỡ trên con đường sự nghiệp trồng người, và tràn đầy hạnh phúc trong cuộc sống.

Trân trọng

## **NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN**

....., ngày .... tháng .... năm 2025

**Người nhận xét**

## MỤC LỤC

<b>I CHƯƠNG 1. GIỚI THIỆU TỔNG QUAN VỀ DỮ LIỆU</b>	<b>6</b>
I.1. Phát biểu về dữ liệu	6
I.1.1. Mô tả về dữ liệu	6
I.1.2. Thuộc tính của kho dữ liệu	6
I.1.3. Kho dữ liệu đã xử lý	13
I.2. Xây dựng kho dữ liệu	19
I.2.1. Sơ đồ bông tuyết minh họa	19
I.2.2. Dim_LOCATION	19
I.2.3. Dim_CUSTOMER	19
I.2.4. Dim_STORES	20
I.2.5. Dim_EMPLOYEES	20
I.2.6. Dim_CATEGORIES	21
I.2.7. Dim_SUBCATEGORIES	21
I.2.8. Dim_DISCOUNTS	21
I.2.9. DISCOUNTS_BRIDGE	22
I.2.10. Dim_COLORS	22
I.2.11. Dim_SIZES	22
I.2.12. Dim_PRODUCTS	23
I.2.13. Dim_TRANSACTIONTYPE	23
I.2.14. Dim_PAYMENTMETHOD	24
I.2.15. Dim_DATE	24
I.2.16. Fact_TRANSACTIONS	24
I.3. Nội dung 10 câu hỏi truy vấn	26
<b>II CHƯƠNG 2. TÍCH HỢP DỮ LIỆU VÀO KHO (SSIS)</b>	<b>27</b>
II.1. Chuẩn bị công cụ và kho dữ liệu	27
II.2. Tạo Project SSIS trong Visual Studio 2022	36
II.3. Tạo bảng Dim và bảng Fact	38
II.3.1. Dim_CUSTOMERS	62
II.3.2. Dim_STORES	80

II.3.3. Dim_EMPLOYEES	92
II.3.4. Dim_CATEGORIES	108
II.3.5. Dim_SUBCATEGORIES	121
II.3.6. Dim_DISCOUNTS	131
II.3.7. DISCOUNTS_BRIDGE	177
II.3.8. Dim_COLORS	193
II.3.9. Dim_SIZES	202
II.3.10. Dim_PRODUCTS	219
II.3.11. Dim_TRANSACTIONTYPE	246
II.3.12. Dim_PAYMENTMETHOD	259
II.3.13. Dim_DATE	269
II.3.14. Fact_TRANSACTIONS	280
II.4. Quá trình chạy toàn bộ SSIS	328
II.4.1. Tạo Execute SQL Task	328
II.4.2. Chạy quá trình SSIS	331
<b>III CHƯƠNG 3. KHAI THÁC DỮ LIỆU (SSAS)</b>	<b>332</b>
III.1. Tạo project SSAS trong Visual Studio 2022	332
III.2. Tiến hành thực hiện quá trình SSAS	334
III.2.1. Kết nối với kho dữ liệu	334
III.2.2. Thực hiện câu truy vấn	414
III.2.3. Câu lệnh truy vấn MDX	435
III.2.4. Report	446
<b>IV CHƯƠNG 4: KHAI PHÁ DỮ LIỆU (DATA MINING)</b>	<b>490</b>
IV.1. Ngữ cảnh và đặt vấn đề	490
IV.2. Phương pháp tiếp cận và hướng đi	491
IV.2.1. Lựa chọn thuật toán	491
IV.2.2. Các bước triển khai kỹ thuật (Pipeline)	492
IV.3. Kết quả đạt được	493
IV.3.1. Kết quả từ thuật toán K-means	493
IV.3.2. Kết quả từ thuật toán Hierarchical Clustering	496

# CHƯƠNG 1. GIỚI THIỆU TỔNG QUAN VỀ DỮ LIỆU

## I.1. Phát biểu về dữ liệu

### I.1.1. Mô tả về dữ liệu

- Bộ dữ liệu tổng hợp này mô phỏng dữ liệu giao dịch trong hai năm của một nhà bán lẻ thời trang đa quốc gia, bao gồm:
  - o Hơn 4 triệu hồ sơ bán hàng
  - o 35 cửa hàng tại 7 quốc gia: Hoa Kỳ, Trung Quốc, Đức, Vương quốc Anh, Pháp, Tây Ban Nha, Bồ Đào Nha
- Có 6 tệp csv (Kích thước trung bình: ~ 200MB). Mỗi tệp mô tả dữ liệu cho một khu vực, bao gồm khách hàng, chiết khấu, nhân viên, sản phẩm, cửa hàng và giao dịch.
- Link dataset gốc: [Global Fashion Retail Sales](#)

### I.1.2. Thuộc tính của kho dữ liệu

- File **customers.csv**: nặng khoảng 190.55 MB, chứa 1643306 dòng dữ liệu về khách hàng.

Bảng 1 Bảng thuộc tính file customers.csv

STT	Tên thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	Customer ID	Numerical	Mã số nhận dạng duy nhất cho khách hàng
2	Name	Text	Họ và tên đầy đủ (có thể bao gồm danh hiệu/kính ngữ Mr. hoặc hậu tố chuyên môn)

3	Email	Text	Email ẩn danh với tên miền giả (fake_gmail.com, fake_hotmail.com)
4	Telephone	Text	Số điện thoại có định dạng không nhất quán (hỗn hợp mã quốc gia và phần mở rộng)
5	City	Text	Thành phố
6	Country	Text	Quốc gia
7	Gender	Text	Gender (F, M, D), D = «đa dạng»
8	Date Of Birth	Date	Ngày sinh định dạng theo YYYY-MM-DD
9	Job Title	Text	Nghề nghiệp

- File **discounts.csv**: nặng khoảng 18.27 kB, chứa khoảng 181 dòng dữ liệu về khuyến mãi của cửa hàng.

Bảng 2 Bảng thuộc tính file discounts.csv

STT	Tên thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	Start Date	Date	Ngày giảm giá có hiệu lực, theo định dạng YYYY-MM-DD
2	End Date	Date	Ngày hết hạn giảm giá, theo định dạng YYYY-MM-DD
3	Discount	Numerical	Giá trị thập phân biểu thị tỷ lệ chiết khấu (ví dụ: 0,20 nghĩa là chiết khấu 20%)
4	Description	Text	Mô tả ngắn gọn về chiến dịch giảm giá

5	Category	Text	Các loại sản phẩm được áp dụng giảm giá
6	Sub Category	Text	Danh mục sản phẩm phụ được áp dụng giảm giá

- File **employees.csv**: nặng khoảng 15.17 kB, chứa 404 dòng dữ liệu về nhân viên

Bảng 3 Bảng thuộc tính *employees.csv*

STT	Tên thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	Employee ID	Numerical	Mã số định danh duy nhất cho nhân viên
2	Stores ID	Numerical	Khóa ngoại liên kết đến vị trí ID cửa hàng trong <i>stores.csv</i>
3	Name	Text	Họ và tên đầy đủ định dạng theo [First Name] [Last Name]
4	Position	Text	Vai trò trong hệ thống phân cấp cửa hàng ( <u>Manager</u> giám sát hoạt động, <u>Seller</u> xử lý giao dịch)

- File **products.csv**: nặng khoảng 4.99 MB, chứa 17941 dòng dữ liệu về sản phẩm.

Bảng 4 Bảng thuộc tính file *products.csv*

STT	Tên thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	Product ID	Numerical	Mã số nhận dạng duy nhất cho sản phẩm
2	Category	Text	Phân loại sản phẩm cấp cao (EN) (ví dụ: <u>Feminine</u> , <u>Masculine</u> , <u>Children</u> )

3	Sub Category	Text	Phân loại cụ thể hơn trong danh mục (EN)
4	Description PT	Text	Mô tả sản phẩm bằng tiếng Bồ Đào Nha
5	Description DE	Text	Mô tả sản phẩm bằng tiếng Đức
6	Description FR	Text	Mô tả sản phẩm bằng tiếng Pháp
7	Description ES	Text	Mô tả sản phẩm bằng tiếng Tây Ban Nha
8	Description EN	Text	Mô tả sản phẩm bằng tiếng Anh
9	Description ZH	Text	Mô tả sản phẩm bằng tiếng Trung
10	Color	Text	Màu sản phẩm
11	Size	Text	Kích thước sản phẩm có sẵn
12	Production Cost	Numerical	Chi phí phát sinh để sản xuất Sản phẩm bằng USD

- File **stores.csv**: nặng khoảng 2.19 kB, chứa 35 dòng dữ liệu về cửa hàng

*Bảng 5 Bảng thuộc tính file stores.csv*

STT	Tên thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	Store ID	Numerical	Mã định danh duy nhất cho vị trí cửa hàng
2	Country	Text	Quốc gia nơi cửa hàng tọa lạc
3	City	Text	Thành phố nơi cửa hàng tọa lạc
4	Store Name	Text	Tên dễ đọc theo định dạng Store [City]
5	Number of Employees	Numerical	Tổng số nhân viên được phân công vào cửa hàng
6	ZIP Code	Numerical	Mã bưu chính của vị trí cửa hàng
7	Latitude	Numerical	Vĩ độ địa lý của vị trí cửa hàng
8	Longitude	Numerical	Kinh độ địa lý của vị trí cửa hàng

- File **transactions.csv**: nặng khoảng 805.15 MB, chứa 6416827 dòng dữ liệu về mua hàng

*Bảng 6 Bảng thuộc tính file transactions.csv*

STT	Tên thuộc tính	Kiểu dữ liệu	Ý nghĩa

1	Invoice ID	Text	Mã định danh duy nhất cho mỗi giao dịch, phân biệt bán hàng và trả lại. Mã định danh này theo định dạng: (INV cho bán hàng hoặc RET cho trả lại) + Mã quốc gia + Mã cửa hàng + Bộ đếm tuần tự. Điều này đảm bảo tất cả các mặt hàng từ cùng một giao dịch được nhóm vào cùng một hóa đơn.
2	Line	Numerical	Số thứ tự thể hiện vị trí của sản phẩm trong hóa đơn. Một hóa đơn có thể chứa nhiều mục hàng.
3	Customer ID	Numerical	Mã định danh duy nhất tham chiếu đến khách hàng đã thực hiện giao dịch mua, có trong <u>customers.csv</u>
4	Product ID	Numerical	Mã định danh duy nhất tham chiếu đến sản phẩm đã mua, có trong <u>products.csv</u>
5	Size	Numerical	Kích thước sản phẩm khác nhau (ví dụ: S, M, L, XL). Để trống nếu không áp dụng.
6	Color	Text	Sự thay đổi màu sắc của sản phẩm. Để trống nếu không áp dụng.
7	Unit Price	Numerical	Giá của một đơn vị sản phẩm trước khi áp dụng bất kỳ khoản chiết khấu nào
8	Quantity	Numerical	Số lượng sản phẩm đã mua trong mục hóa đơn này.

9	Date	Date	Ngày và giờ giao dịch theo định dạng YYYY-MM-DD HH:MM:SS (định dạng 24 giờ).
10	Discount	Numerical	Giảm giá áp dụng cho từng mặt hàng, được biểu thị dưới dạng số thập phân (ví dụ: 0.30= giảm giá 30%, 0.00= không giảm giá), tìm thấy danh sách <u>discounts.csv</u>
11	Line Total	Numerical	Tổng chi phí cho từng mặt hàng sau khi áp dụng bất kỳ khoản chiết khấu nào. Được tính như sau: Unit Price × Quantity × (1 - Discount).
12	Store ID	Numerical	Mã định danh duy nhất tham chiếu đến cửa hàng nơi giao dịch diễn ra, có trong <u>stores.csv</u>
13	Employee ID	Numerical	Mã định danh duy nhất tham chiếu đến nhân viên đã xử lý giao dịch, có trong <u>employees.csv</u>
14	Currency	Numerical	Mã tiền tệ ISO gồm ba chữ cái đại diện cho loại tiền tệ được sử dụng cho giao dịch (ví dụ: USD, EUR, CNY, GBP).
15	Currency Symbol	Text	Biểu tượng liên quan đến loại tiền giao dịch (ví dụ: \$, €, £, ¥).
16	SKU	Text	Đơn vị quản lý kho (SKU), mã hàng tồn kho duy nhất kết hợp ID sản phẩm, Kích thước và Màu sắc (ví

			dụ: FESH81-M-PINK= ID sản phẩm 81 + Kích thước M + Màu HỒNG)
17	Transaction Type	Text	Chỉ định giao dịch là <u>Bán hàng</u> hay <u>Trả lại</u> .
18	Payment Method	Text	Phương pháp được sử dụng để hoàn tất giao dịch (ví dụ: <u>Thẻ tín dụng</u> , <u>Tiền mặt</u> ).
19	Invoice Total	Numerical	Chỉ tổng giá trị của giao dịch (ID Hóa đơn). Đây là tổng của tất cả các giá trị Tổng dòng cho cùng một ID Hóa đơn. Giá trị này được lặp lại trên tất cả các mục dòng trong cùng một ID Hóa đơn.

### I.1.3. Kho dữ liệu đã xử lý

- Thuộc tính file **customers.csv**

*Bảng 7 Bảng thuộc tính file customers.csv đã được xử lý*

STT	Tên thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	Customer ID	Numerical	Mã số nhận dạng duy nhất cho khách hàng
2	City	Text	Thành phố
3	Country	Text	Quốc gia
4	Gender	Text	Gender 1. F 2. M

			3. D
5	Date Of Birth	Date	Ngày sinh định dạng theo YYYY-MM-DD
6	Job Title	Text	Nghề nghiệp

- Thuộc tính file **discounts.csv**

Bảng 8 Bảng thuộc tính file discounts.csv đã được xử lý

STT	Tên thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	Start Date	Date	Ngày giảm giá có hiệu lực, theo định dạng YYYY-MM-DD
2	End Date	Date	Ngày hết hạn giảm giá, theo định dạng YYYY-MM-DD
3	Discount	Numerical	Giá trị thập phân biểu thị tỷ lệ chiết khấu (ví dụ: 0,20 nghĩa là chiết khấu 20%)
4	Description	Text	Mô tả ngắn gọn về chiến dịch giảm giá
5	Category	Text	Các loại sản phẩm được áp dụng giảm giá
6	Sub Category	Text	Danh mục sản phẩm phụ được áp dụng giảm giá

- Thuộc tính file **employees.csv**

Bảng 9 Bảng thuộc tính file employees.csv đã được xử lý

STT	Tên thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	Employee ID	Numerical	Mã số định danh duy nhất cho nhân viên

2	Stores ID	Numerical	Khóa ngoại liên kết đến vị trí ID cửa hàng trong <u>stores.csv</u>
3	Position	Text	Vai trò trong hệ thống phân cấp cửa hàng 1. Manager 2. Seller

- Thuộc tính file **products.csv**

Bảng 10 Bảng thuộc tính file products.csv đã được xử lý

STT	Tên thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	Product ID	Numerical	Mã số nhận dạng duy nhất cho sản phẩm
2	Category	Text	Phân loại sản phẩm cấp cao 1. Feminine 2. Masculine 3. Children
3	Sub Category	Text	Phân loại cụ thể hơn trong danh mục (EN)
4	Description EN	Text	Mô tả sản phẩm bằng tiếng Anh
5	Color	Text	Màu sản phẩm
6	Size	Text	Kích thước sản phẩm có sẵn
7	Production Cost	Numerical	Chi phí phát sinh để sản xuất Sản phẩm bằng USD

- Thuộc tính file **stores.csv**

Bảng 11 Bảng thuộc tính file stores.csv đã được xử lý

STT	Tên thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	Store ID	Numerical	Mã định danh duy nhất cho vị trí cửa hàng
2	StoreName	Text	Tên cửa hàng
3	Country	Text	Quốc gia nơi cửa hàng tọa lạc
4	City	Text	Thành phố nơi cửa hàng tọa lạc
5	Number of Employees	Numerical	Tổng số nhân viên được phân công vào cửa hàng

- Thuộc tính file **transactions.csv**

Bảng 12 Bảng thuộc tính file transactions.csv đã được xử lý

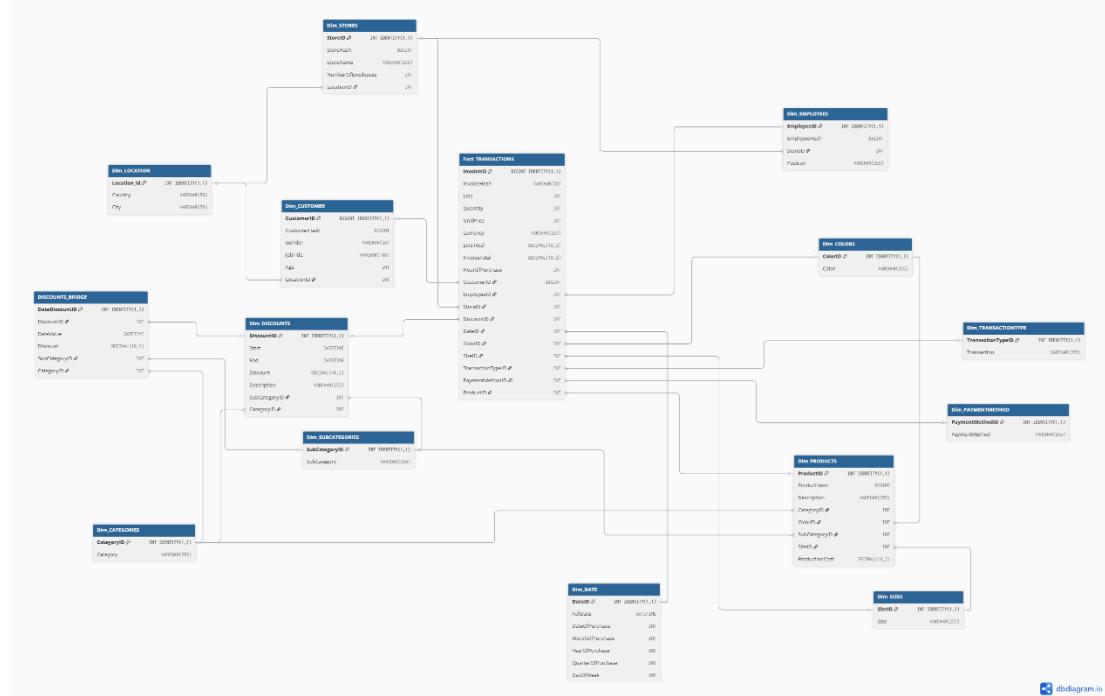
STT	Tên thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	Invoice ID	Text	Mã định danh duy nhất cho mỗi giao dịch, phân biệt bán hàng và trả lại. Mã định danh này theo định dạng: (INV cho bán hàng hoặc RET cho trả lại) + Mã quốc gia + Mã cửa hàng + Bộ đếm tuần tự. Điều này đảm bảo tất cả các mặt hàng từ cùng một giao dịch được nhóm vào cùng một hóa đơn.
2	Line	Numerical	Số thứ tự thể hiện vị trí của sản phẩm trong hóa đơn. Một hóa đơn có thể chứa nhiều mục hàng.

3	Customer ID	Numerical	Mã định danh duy nhất tham chiếu đến khách hàng đã thực hiện giao dịch mua, có trong <u>customers.csv</u>
4	Product ID	Numerical	Mã định danh duy nhất tham chiếu đến sản phẩm đã mua, có trong <u>products.csv</u>
5	Size	Numerical	Kích thước sản phẩm khác nhau (ví dụ: S, M, L, XL). Để trống nếu không áp dụng.
6	Color	Text	Sự thay đổi màu sắc của sản phẩm. Để trống nếu không áp dụng.
7	Unit Price	Numerical	Giá của một đơn vị sản phẩm trước khi áp dụng bất kỳ khoản chiết khấu nào
8	Quantity	Numerical	Số lượng sản phẩm đã mua trong mục hóa đơn này.
9	Date	Date	Ngày và giờ giao dịch theo định dạng YYYY-MM-DD HH:MM:SS (định dạng 24 giờ).
10	Discount	Numerical	Giảm giá áp dụng cho từng mặt hàng, được biểu thị dưới dạng số thập phân (ví dụ: 0.30= giảm giá 30%, 0.00= không giảm giá), tìm thấy danh sách <u>discounts.csv</u>
11	Line Total	Numerical	Tổng chi phí cho từng mặt hàng sau khi áp dụng bất kỳ khoản chiết khấu nào. Được tính như sau: Unit Price × Quantity × (1 - Discount).

12	Store ID	Numerical	Mã định danh duy nhất tham chiếu đến cửa hàng nơi giao dịch diễn ra, có trong <u>stores.csv</u>
13	Employee ID	Numerical	Mã định danh duy nhất tham chiếu đến nhân viên đã xử lý giao dịch, có trong <u>employees.csv</u>
14	Currency	Numerical	Mã tiền tệ ISO gồm ba chữ cái đại diện cho loại tiền tệ được sử dụng cho giao dịch (ví dụ: USD, EUR, CNY, GBP).
15	Transaction Type	Text	Chỉ định giao dịch là 1. Bán hàng 2. Trả lại.
16	Payment Method	Text	Phương pháp được sử dụng để hoàn tất giao dịch 1. Thẻ tín dụng 2. Tiền mặt
17	Invoice Total	Numerical	Chỉ tổng giá trị của giao dịch (ID Hóa đơn). Đây là tổng của tất cả các giá trị Tổng dòng cho cùng một ID Hóa đơn. Giá trị này được lặp lại trên tất cả các mục dòng trong cùng một ID Hóa đơn.

## I.2. Xây dựng kho dữ liệu

### I.2.1. Sơ đồ bông tuyết minh họa



Hình 1 Sơ đồ bông tuyết minh họa kho dữ liệu

### I.2.2. Dim\_LOCATION

Bảng 13 Bảng thuộc tính bảng Dim\_LOCATION

Khóa	Tên thuộc tính	NULL	Kiểu dữ liệu	Mô tả
Khóa chính	location_id		integer	Mã địa điểm
	country		varchar(50)	Quốc gia
	city		varchar(50)	Thành phố

### I.2.3. Dim\_CUSTOMER

Bảng 14 Bảng thuộc tính bảng Dim\_CUSTOMER

<b>Khóa</b>	<b>Tên thuộc tính</b>	<b>NULL</b>	<b>Kiểu dữ liệu</b>	<b>Mô tả</b>
Khóa chính	CustomerID		bigint	Mã khách hàng
	CustomerHash		bigint	Mã khách hàng (ban đầu)
	Gender		varchar(50)	Giới tính
	JobTitle	X	varchar(100)	Nghề nghiệp
	Age		integer	Tuổi khách hàng
Khóa ngoại	LocationID		integer	Mã địa điểm

#### I.2.4. Dim\_STORES

Bảng 15 Bảng thuộc tính bảng Dim\_STORES

<b>Khóa</b>	<b>Tên thuộc tính</b>	<b>NULL</b>	<b>Kiểu dữ liệu</b>	<b>Mô tả</b>
Khóa chính	StoreID		integer	Mã cửa hàng
	StoreHash		bigint	Mã cửa hàng (ban đầu)
	StoreName		varchar(255)	Tên cửa hàng
	NumberOfEmployees		integer	Số nhân viên
Khóa ngoại	LocationID		integer	Mã địa điểm

#### I.2.5. Dim\_EMPLOYEES

Bảng 16 Bảng thuộc tính bảng Dim\_EMPLOYEES

<b>Khóa</b>	<b>Tên thuộc tính</b>	<b>NULL</b>	<b>Kiểu dữ liệu</b>	<b>Mô tả</b>
Khóa chính	EmployeeID		integer	Mã nhân viên
	EmployeeHash		bigint	Mã nhân viên (ban đầu)
	Position		varchar(255)	Vị trí
Khóa ngoại	StoreID		integer	Mã cửa hàng

#### I.2.6. Dim\_CATEGORIES

Bảng 17. Bảng thuộc tính bảng Dim\_CATEGORIES

<b>Khóa</b>	<b>Tên thuộc tính</b>	<b>NULL</b>	<b>Kiểu dữ liệu</b>	<b>Mô tả</b>
Khóa chính	CategoryID		integer	Mã danh mục
	Category		varchar(255)	Tên danh mục

#### I.2.7. Dim\_SUBCATEGORIES

Bảng 18. Bảng thuộc tính bảng Dim\_SUBCATEGORIES

<b>Khóa</b>	<b>Tên thuộc tính</b>	<b>NULL</b>	<b>Kiểu dữ liệu</b>	<b>Mô tả</b>
Khóa chính	SubCategoryID		integer	Mã danh mục phụ
	SubCategory		varchar(255)	Tên danh mục phụ

#### I.2.8. Dim\_DISCOUNTS

Bảng 19. Bảng thuộc tính bảng Dim\_DISCOUNTS

<b>Khóa</b>	<b>Tên thuộc tính</b>	<b>NULL</b>	<b>Kiểu dữ liệu</b>	<b>Mô tả</b>
Khóa chính	DiscountID		integer	Mã giảm giá
	Description		varchar(255)	Mô tả chiến dịch
	Start		datetime	Ngày bắt đầu
	End		datetime	Ngày kết thúc
	Discount		decimal	Giảm giá
Khóa ngoại	SubCategoryID		integer	Mã danh mục phụ
Khóa ngoại	CategoryID		integer	Mã danh mục

### I.2.9. DISCOUNTS\_BRIDGE

Bảng 20. Bảng thuộc tính bảng DISCOUNTS\_BRIDGE

<b>Khóa</b>	<b>Tên thuộc tính</b>	<b>NULL</b>	<b>Kiểu dữ liệu</b>	<b>Mô tả</b>
Khóa chính	DateDiscountID		integer	Mã ngày giảm giá
Khóa ngoại	DiscountID		integer	Mã giảm giá
	DateValue		datetime	Ngày giảm giá
	Discount		decimal	Giảm giá
Khóa ngoại	SubCategoryID		integer	Mã danh mục phụ
Khóa ngoại	CategoryID		integer	Mã danh mục

### I.2.10. Dim\_COLORS

Bảng 21. Bảng thuộc tính bảng Dim\_COLORS

Khóa	Tên thuộc tính	NULL	Kiểu dữ liệu	Mô tả
Khóa chính	ColorID		integer	Mã màu sắc
	Color		varchar(255)	Màu sắc

### I.2.11. Dim\_SIZES

Bảng 22. Bảng thuộc tính bảng Dim\_SIZES

Khóa	Tên thuộc tính	NULL	Kiểu dữ liệu	Mô tả
Khóa chính	SizeID		integer	Mã kích thước
	Size		varchar(255)	Kích thước

### I.2.12. Dim\_PRODUCTS

Bảng 23. Bảng thuộc tính bảng Dim\_PRODUCTS

Khóa	Tên thuộc tính	NULL	Kiểu dữ liệu	Mô tả
Khóa chính	ProductID		integer	Mã sản phẩm
	ProductHash		bigint	Mã sản phẩm (ban đầu)
	Description		varchar(255)	Mô tả sản phẩm theo tiếng Anh
Khóa ngoại	CategoryID		integer	Mã danh mục

Khóa ngoại	ColorID		integer	Mã màu sắc
Khóa ngoại	SubCategoryID		integer	Mã danh mục phụ
Khóa ngoại	SizeID		integer	Mã danh mục
	ProductionCost		decimal	Giá sản phẩm

#### I.2.13. Dim\_TRANSACTIONTYPE

Bảng 24. Bảng thuộc tính bảng Dim\_TRANSACTIONTYPE

Khóa	Tên thuộc tính	NULL	Kiểu dữ liệu	Mô tả
Khóa chính	TransactionTypeID		integer	Mã loại bán hàng
	Transaction		varchar(255)	Loại bán hàng

#### I.2.14. Dim\_PAYMENTMETHOD

Bảng 25. Bảng thuộc tính bảng Dim\_PAYMENTMETHOD

Khóa	Tên thuộc tính	NULL	Kiểu dữ liệu	Mô tả
Khóa chính	PaymentMethodID		integer	Mã phương thức thanh toán
	PaymentMethod		varchar(255)	Phương thức thanh toán

#### I.2.15. Dim\_DATE

Bảng 26. Bảng thuộc tính bảng Dim\_DATE

<b>Khóa</b>	<b>Tên thuộc tính</b>	<b>NULL</b>	<b>Kiểu dữ liệu</b>	<b>Mô tả</b>
Khóa chính	DateID		integer	Mã ngày thanh toán
	FullDate		datetime	Ngày thanh toán đầy đủ
	DateOfPurchase		integer	Ngày thanh toán
	MonthOfPurchase		integer	Tháng thanh toán
	YearOfPurchase		integer	Năm thanh toán
	DayOfWeek		integer	Ngày trong tuần
	QuarterOfPurchase		integer	Quý thanh toán

#### I.2.16. Fact\_TRANSACTIONS

Bảng 27. Bảng thuộc tính bảng Fact\_TRANSACTIONS

<b>Khóa</b>	<b>Tên thuộc tính</b>	<b>NULL</b>	<b>Kiểu dữ liệu</b>	<b>Mô tả</b>
Khóa chính	InvoiceID		bigint	Mã mua hàng
	InvoiceHash		varchar(50)	Mã hash mua hàng
	InvoiceTotal		decimal	Tổng giá trị hóa đơn
	UnitPrice		integer	Giá trị món hàng

	Line		integer	Số thứ tự sản phẩm
	LineTotal		decimal	Tổng giá trị của món hàng này
	Quantity		integer	Số lượng
	HourOfPurchase		integer	Giờ mua hàng
	Currency		varchar(255)	Tiền tệ
Khóa ngoại	CustomerID		bigint	Mã khách hàng
Khóa ngoại	ProductID		integer	Mã sản phẩm
Khóa ngoại	SizeID		integer	Mã kích thước
Khóa ngoại	DiscountID		integer	Mã giảm giá
Khóa ngoại	EmployeeID		integer	Mã nhân viên
Khóa ngoại	StoreID		integer	Mã cửa hàng
Khóa ngoại	DateID		integer	Mã ngày thanh toán
Khóa ngoại	ColorID		integer	Mã màu sắc
Khóa ngoại	PaymentMethodID		integer	Mã phương thức thanh toán
Khóa ngoại	TransactionTypeID		integer	Mã loại bán hàng

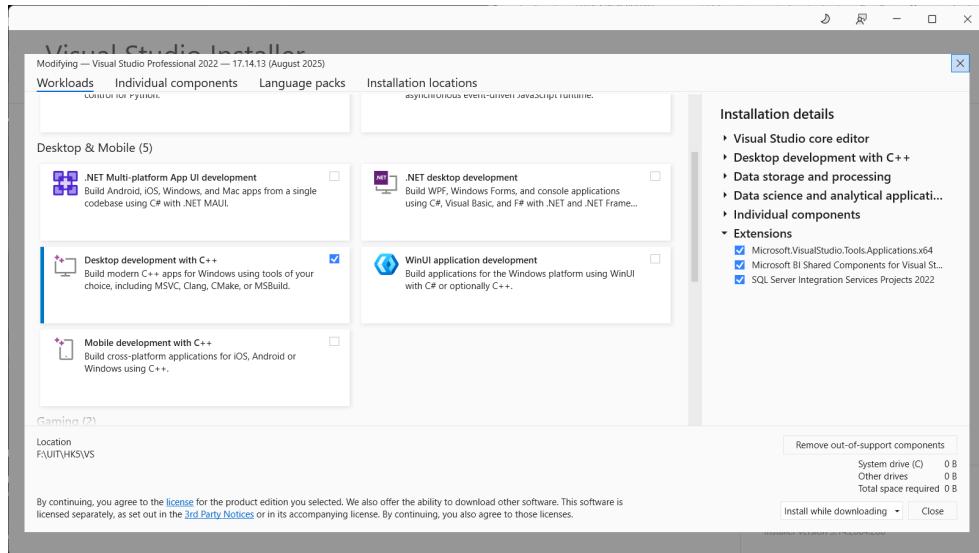
### **I.3. Nội dung 10 câu hỏi truy vấn**

- Doanh số thay đổi như thế nào theo các quý trong năm?
- Doanh số có sự khác biệt giữa các quốc gia không?
- Doanh số ở các quốc gia thay đổi như thế nào qua các năm?
- Doanh số thay đổi như thế nào qua các tháng?
- Các thành phố nào có doanh thu cao nhất?
- Liệu có thể tạo ra các khuyến mãi hữu ích bằng cách xem dữ liệu doanh số của khách hàng nữ?
- Có thể tạo chương trình đặc biệt gì khi xem doanh số một khu vực cụ thể?
- Các danh mục sản phẩm con có sự khác biệt gì qua các năm?
- Các giới tính khác nhau có nhu cầu về sản phẩm trong danh mục con thế nào?
- Khách hàng sử dụng phương thức thanh toán như thế nào?

## CHƯƠNG 2. TÍCH HỢP DỮ LIỆU VÀO KHO (SSIS)

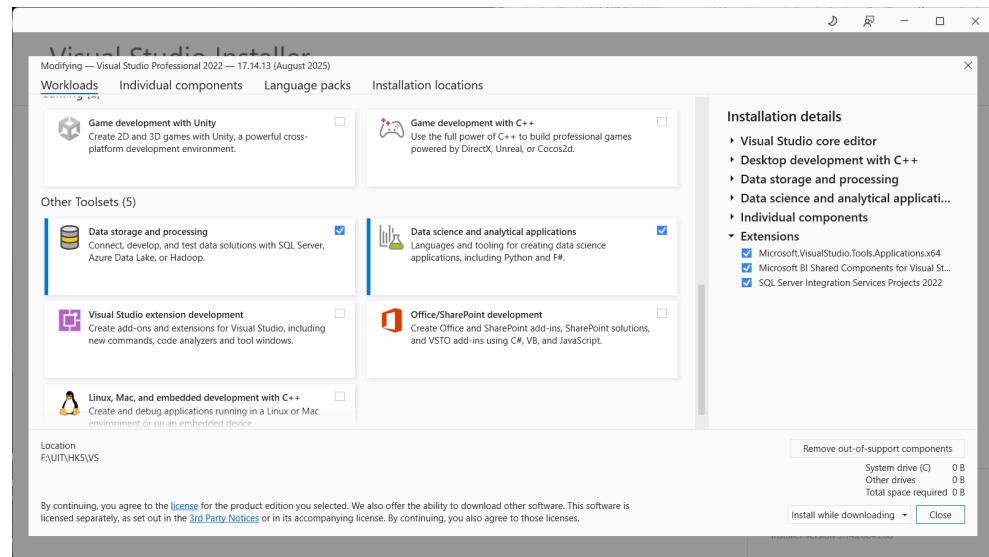
### II.1. Chuẩn bị công cụ và kho dữ liệu

- **Chuẩn bị công cụ:** Download **Visual Studio Community 2022** và tiến hành cài đặt các package cần thiết.
  - o Chọn **Desktop development with C++:**



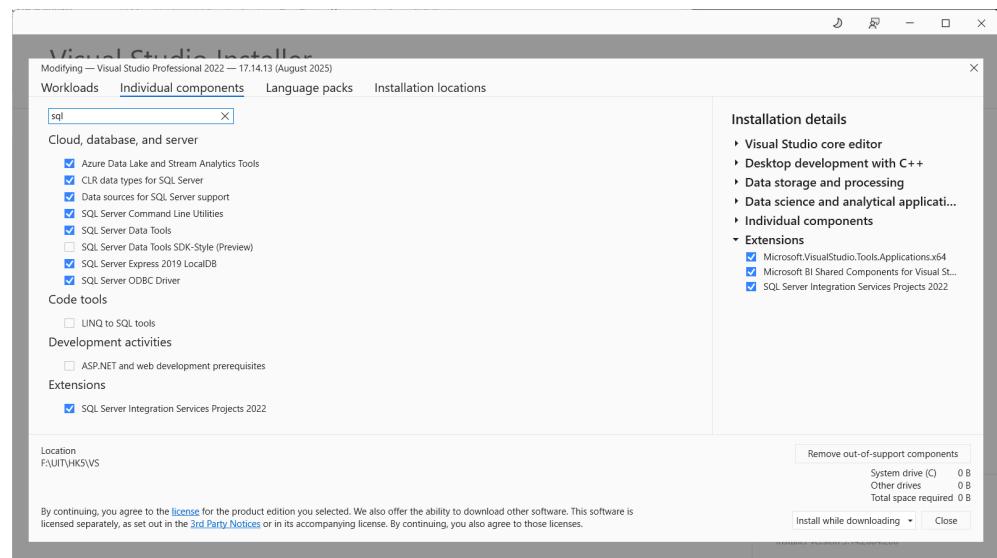
Hình 1. Cài đặt Desktop development C++

o Chọn **Data storage and processing** và **Data science and analytical applications**:



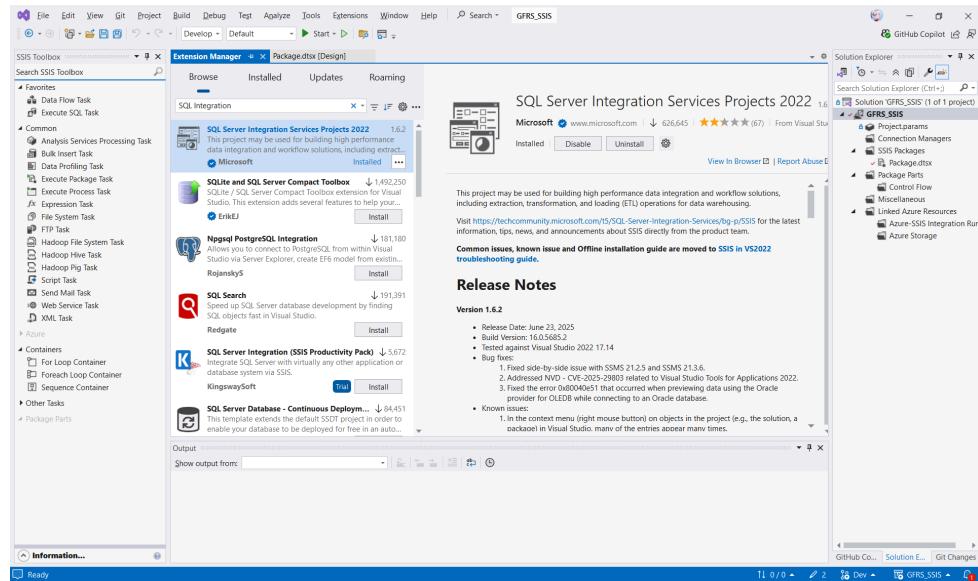
*Hình 2. Chọn các cài đặt khác ở phần Other Toolsets*

o Vào **Individual components**, cài đặt các cài đặt như hình minh họa:



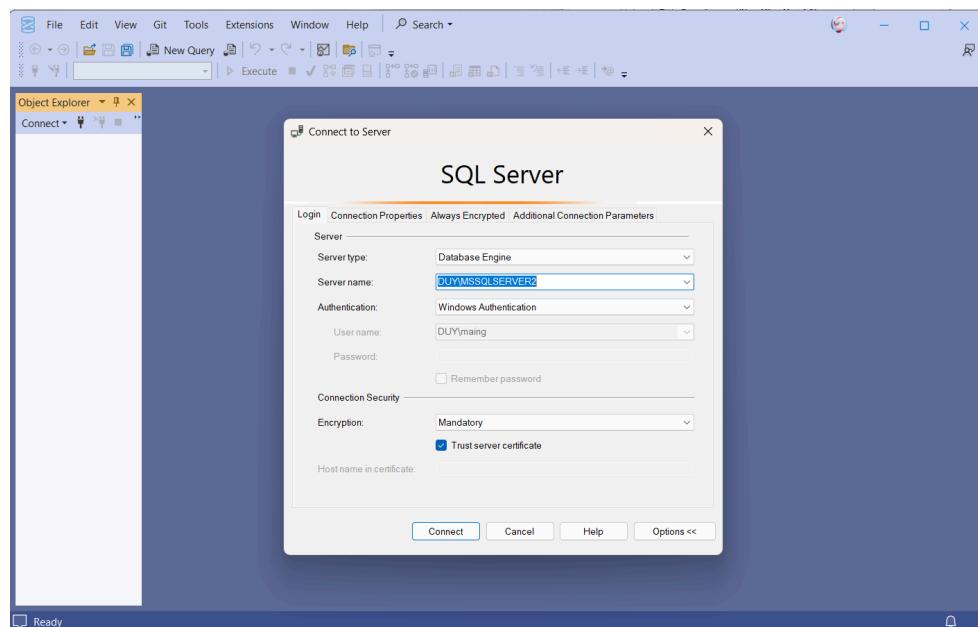
*Hình 3. Chọn các cài đặt khác ở phần Individual components*

- o Vào Extension trên thanh công cụ của Visual Studio 2022, tiến hành tải package SQLServer Integration Services Projects 2022.



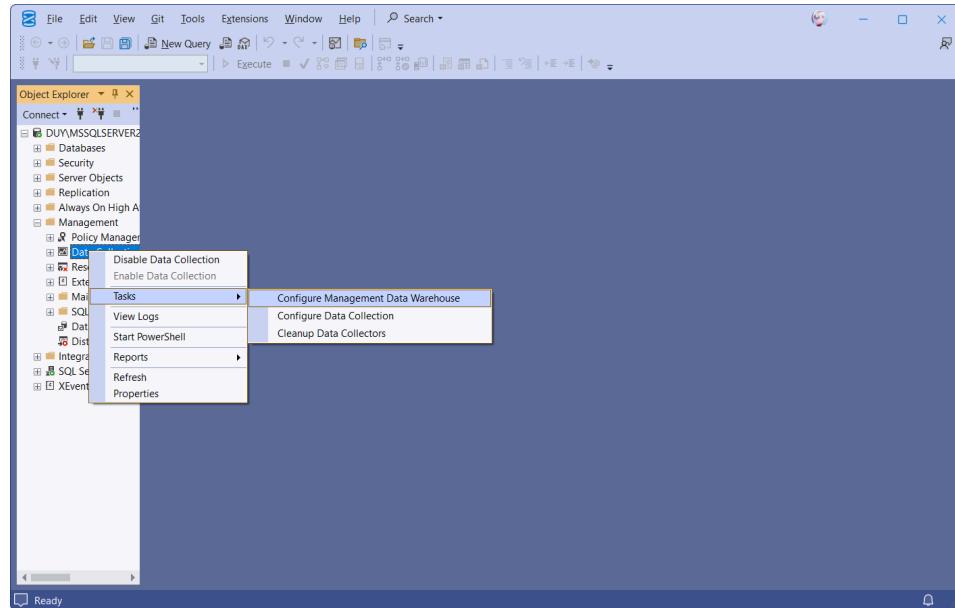
Hình 4. Cài đặt extension cần thiết

- **Kho dữ liệu:** Các bước tạo và cấu hình kho dữ liệu bằng SQL Server 2022
  - o **Bước 1:** Mở Microsoft SQL Server Management Studio kết nối với Server.



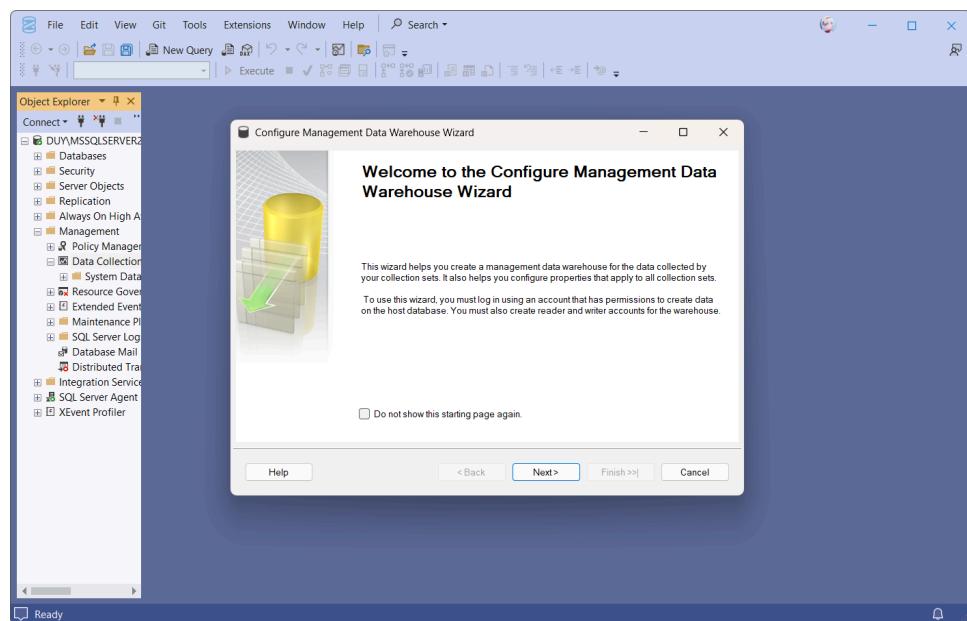
Hình 5. Mở và kết nối vào Microsoft SQL Server Management Studio

- o **Bước 2:** Tạo và cấu hình kho dữ liệu. Vào **Management** → **Data Collection** và nhấp chuột phải rồi chọn **Tasks** → **Configure Management Data Warehouse**



*Hình 6. Tạo và cấu hình kho dữ liệu*

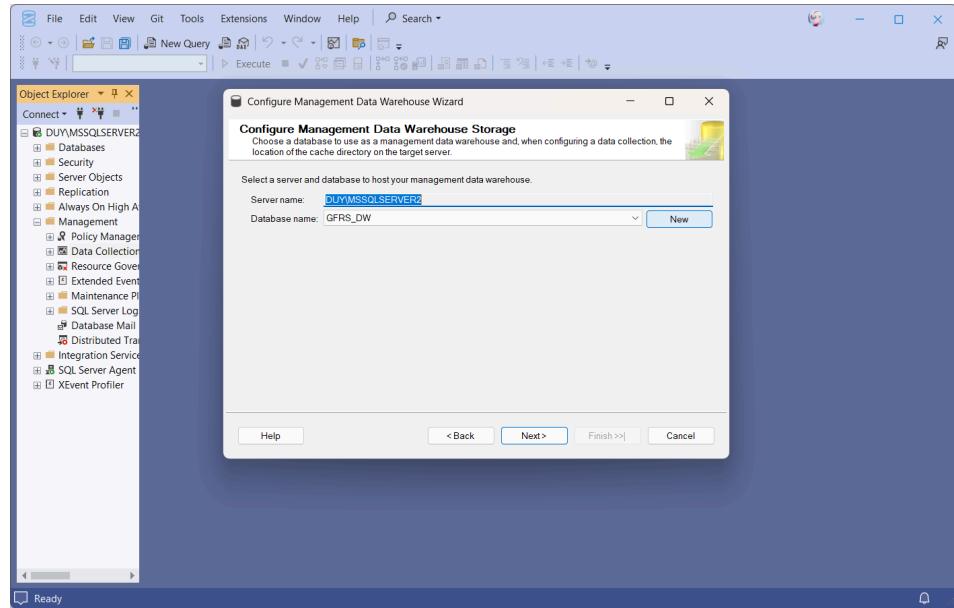
- o **Bước 3:** Nhấn **Next** để tiếp tục, sau đó chọn **New** để tạo kho dữ liệu mới.



*Hình 7. Chọn Next để tiếp tục tạo kho dữ liệu*

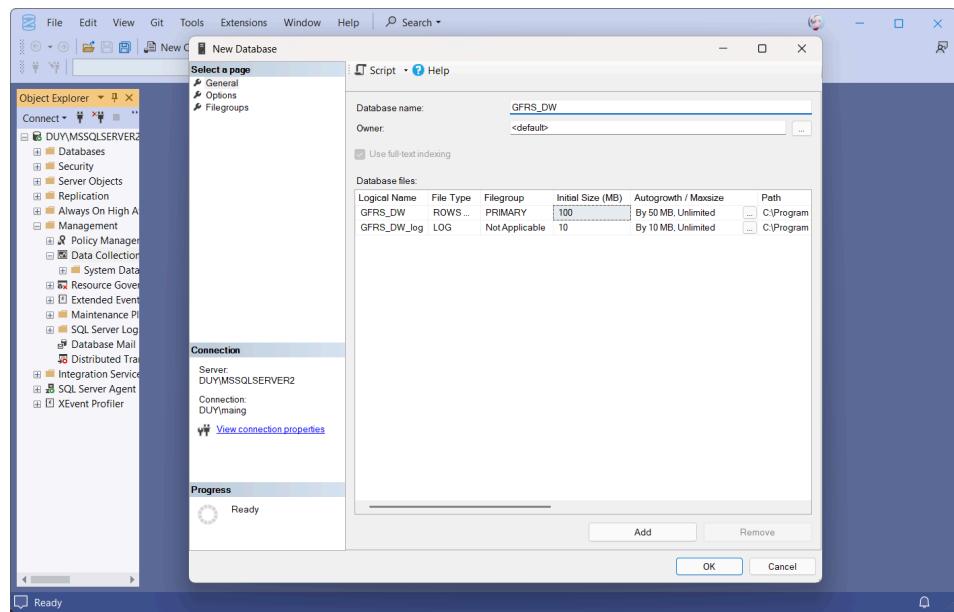


o **Bước 4:** Chọn New để tạo mới kho dữ liệu



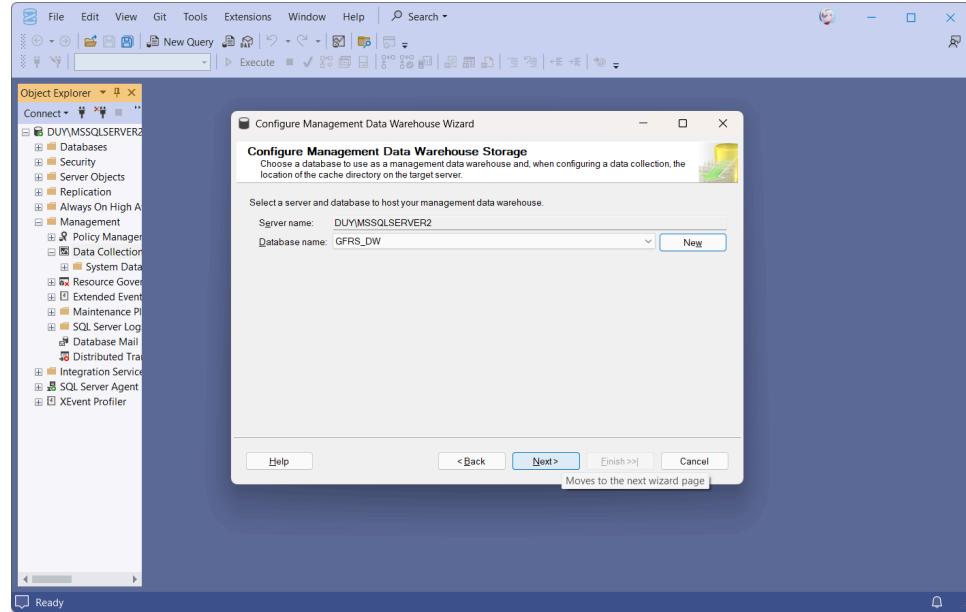
Hình 8. Chọn New để tạo kho dữ liệu mới

o **Bước 5:** Đặt tên cho kho dữ liệu và click **OK**, sau đó tiếp tục nhấn **Next**



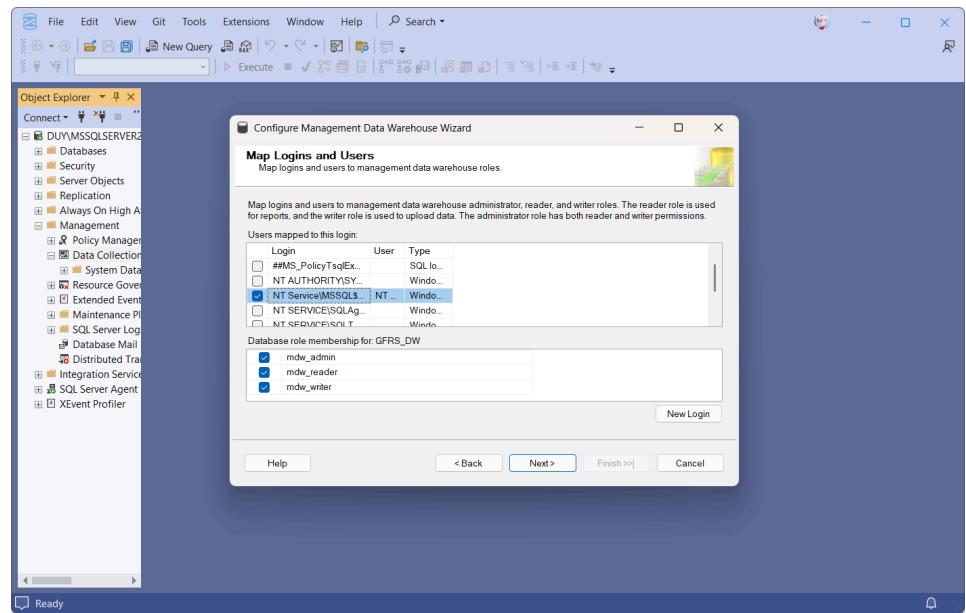
Hình 9. Đặt tên cho kho dữ liệu

o **Bước 6:** Chọn Next để tiếp tục quá trình



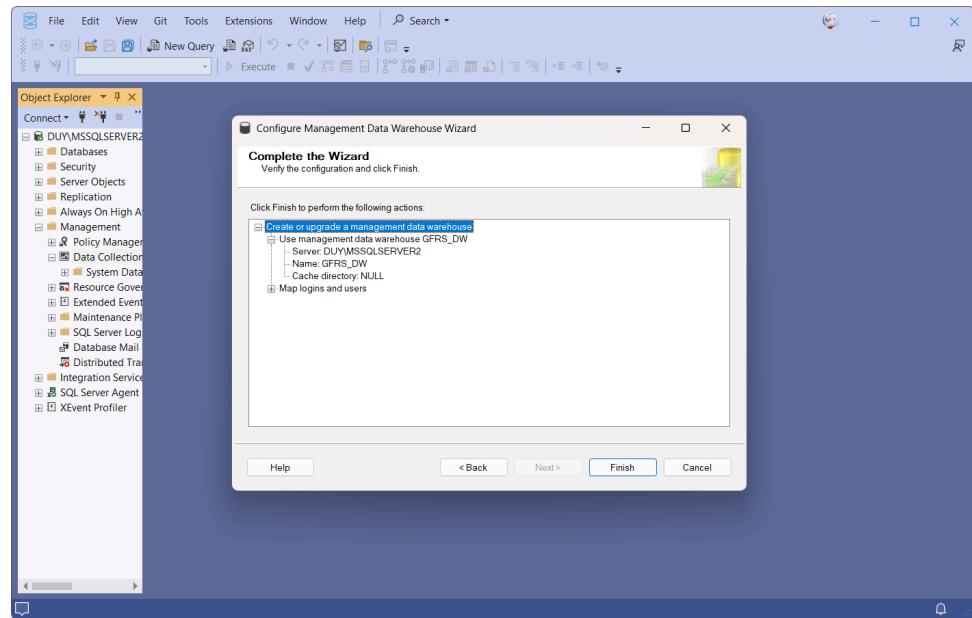
Hình 10. Chọn Next để tiếp tục quá trình

o **Bước 7:** Cài đặt quyền cho Users, chọn Next để tiếp tục



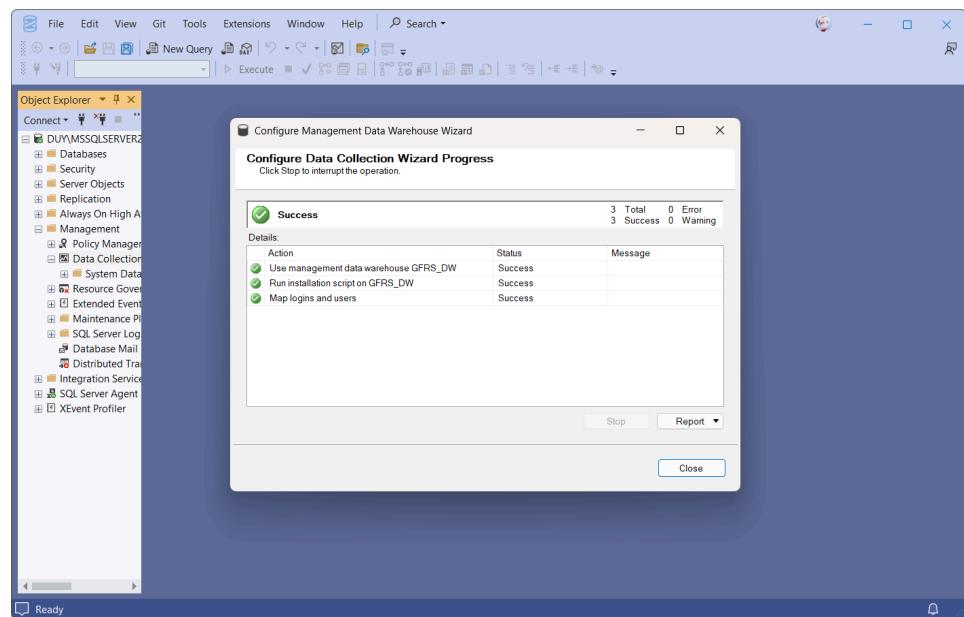
Hình 11. Cài đặt quyền cho Users

o **Bước 8:** Kiểm tra cấu hình kho dữ liệu. Chọn Finish để bắt đầu tạo kho dữ liệu



Hình 12. Kiểm tra cấu hình kho dữ liệu

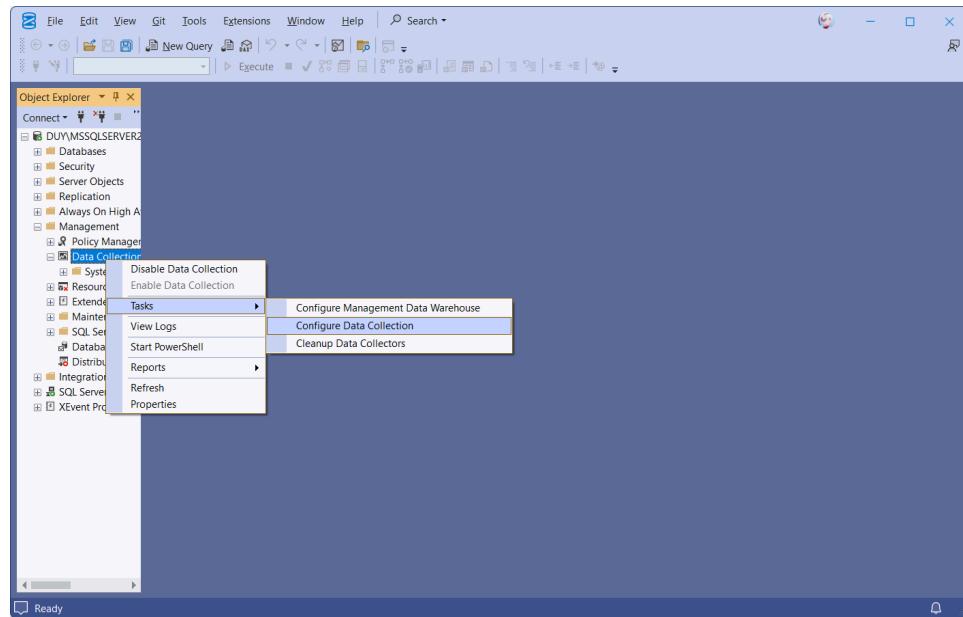
## o Bước 9: Hoàn thành tạo kho dữ liệu.



Hình 13. Hoàn thành tạo kho dữ liệu

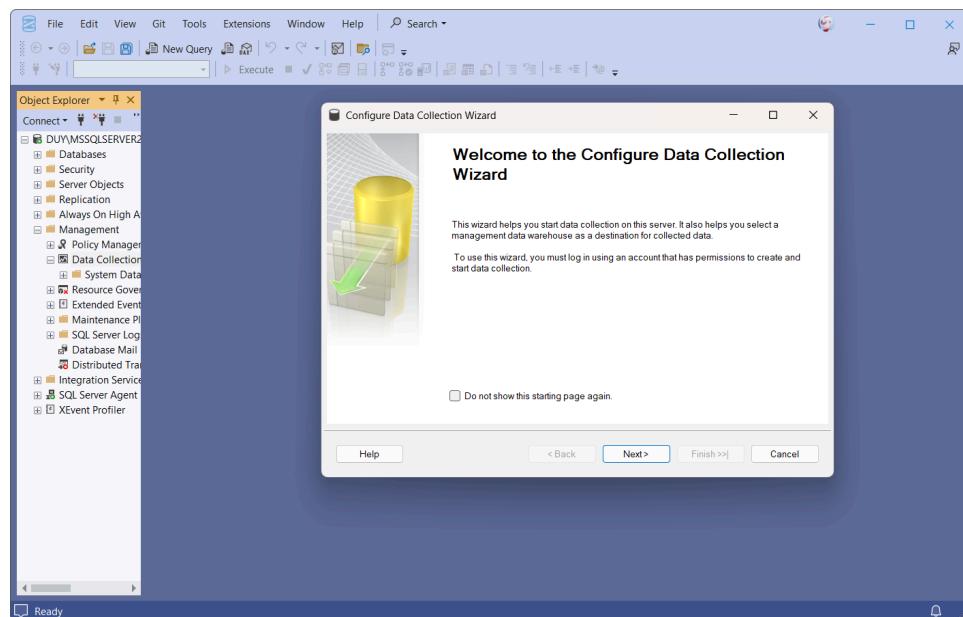
- Cấu hình Data Collection:

- o Bước 1: Vào Management → Data Collection và nhấn chuột phải rồi chọn Tasks → Configure Data Collection.



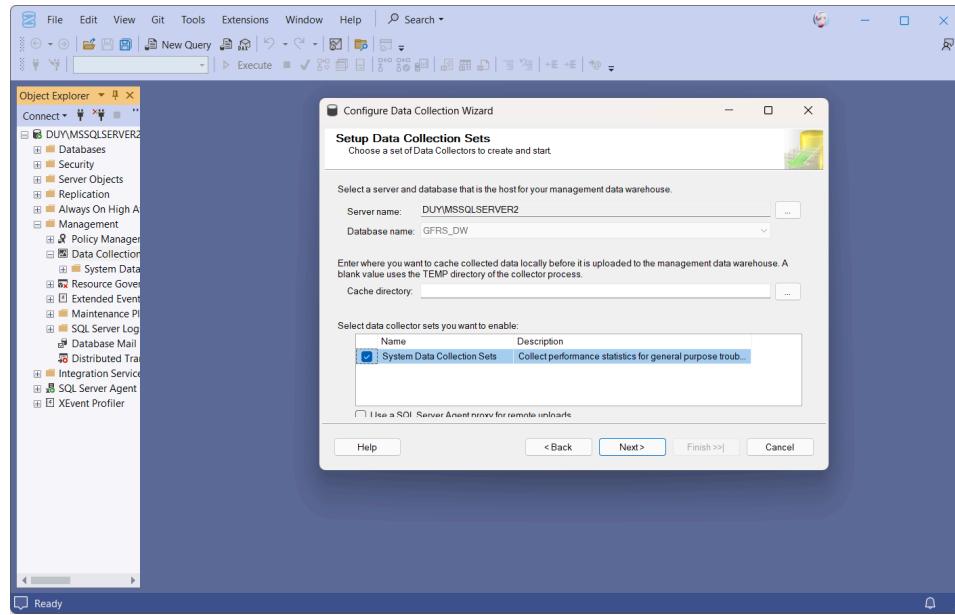
Hình 14. Vào trang cấu hình Data Collection

- o Bước 2: Chọn Next để bắt đầu cấu hình Data Collection.



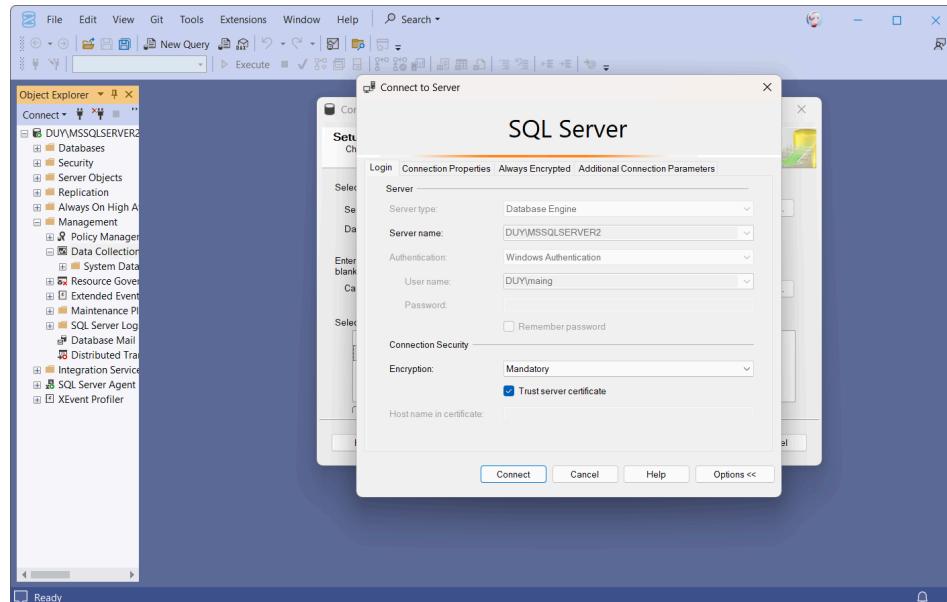
Hình 15. Bắt đầu cấu hình Data Collection

- o **Bước 3:** Ở phần **Select data collector sets you want to enable** chọn **System Data Collection Sets**. Chọn **Next** để tiếp tục.



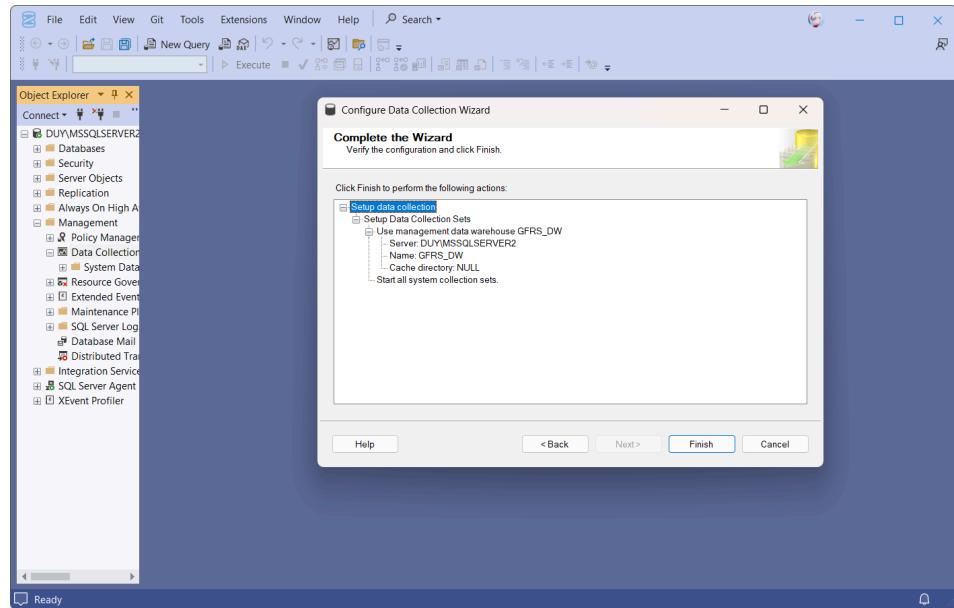
*Hình 16. Chọn System Data Collection Sets*

- o **Bước 4:** Kết nối đến Server. Chọn **Trust server certificate**, chọn **connect** để tiếp tục



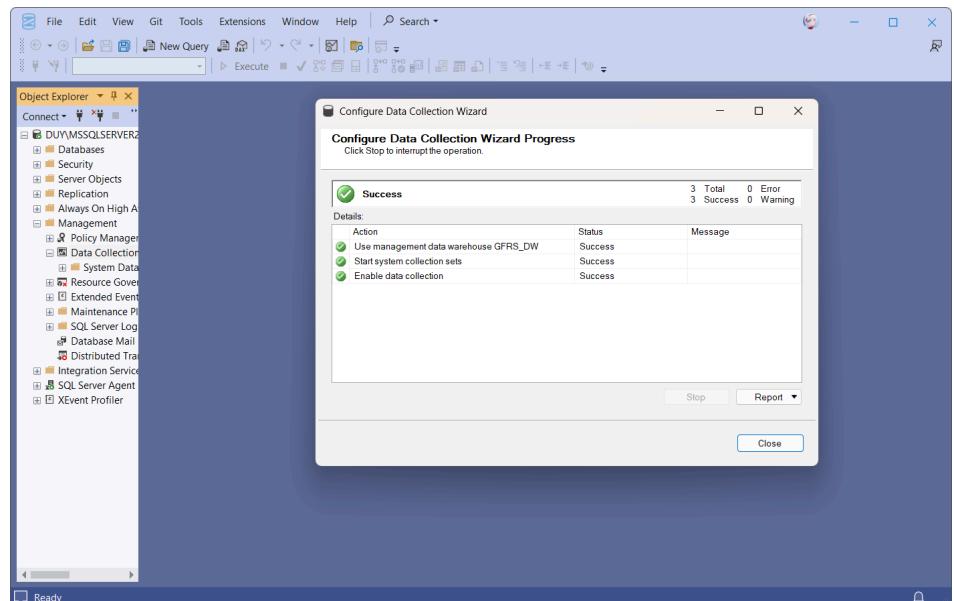
*Hình 17. Kết nối đến server*

- o **Bước 5:** Kiểm tra cấu hình Data Collection



*Hình 18. Kiểm tra cấu hình Data Collection*

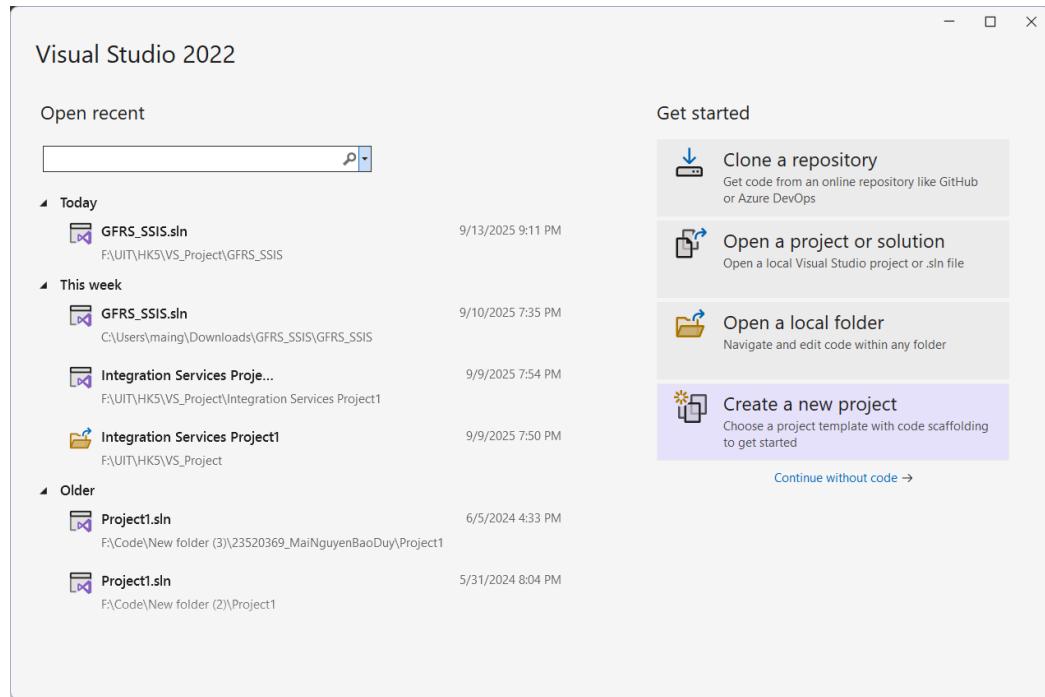
## o Bước 6: Tiến hành cấu hình Data Collection



*Hình 19. Tiến hành cấu hình Data Collection*

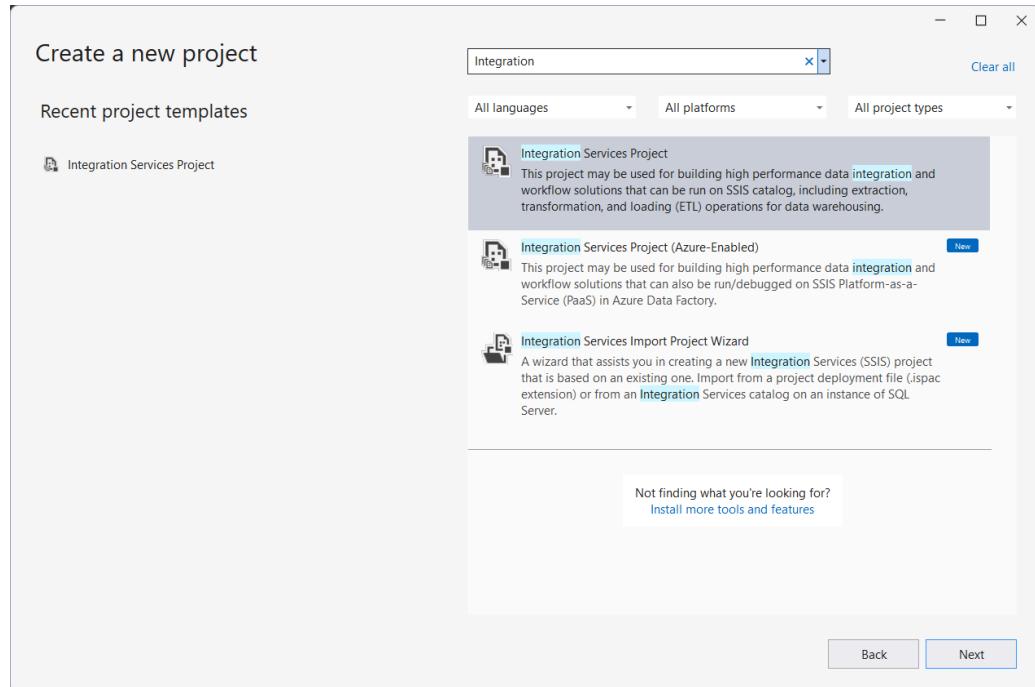
## II.2. Tạo Project SSIS trong Visual Studio 2022

- Bước 1: Mở Visual Studio 2022. Chọn Create a new project



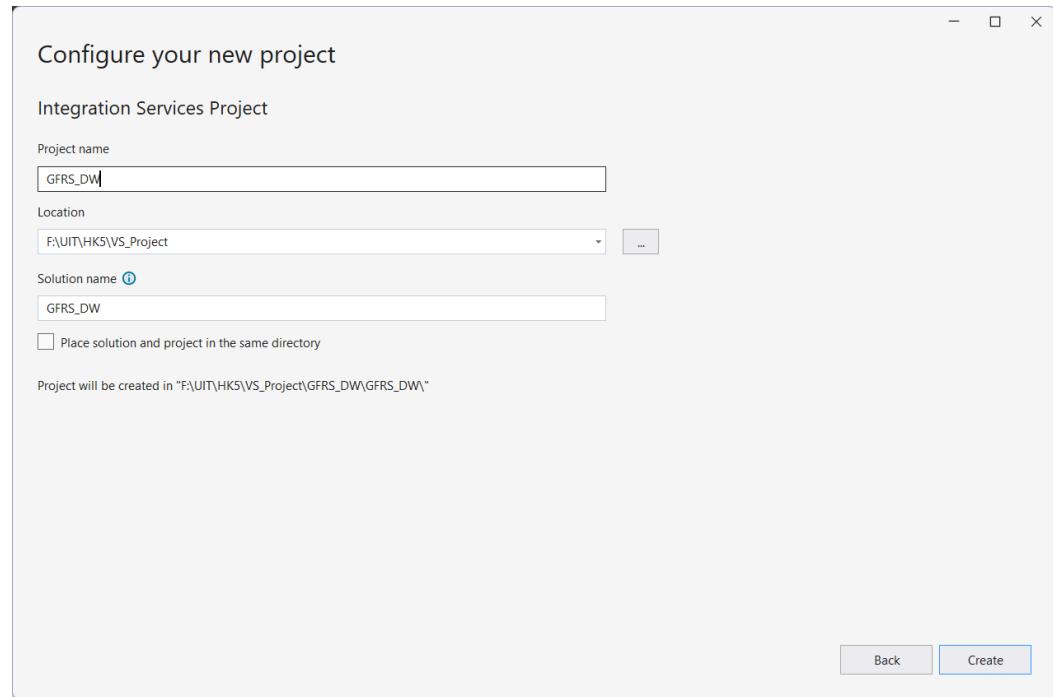
Hình 20. Mở Visual Studio 2022 và tạo project mới

## - Bước 2: Chọn Integration Services Project



Hình 21. Tạo Integration Services Project mới

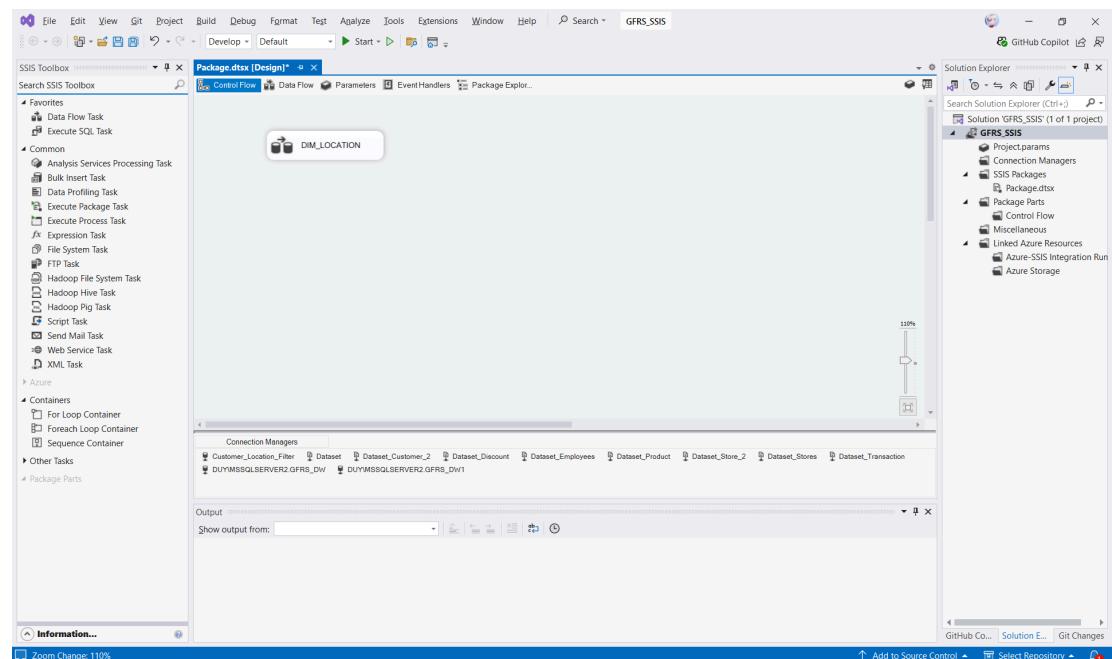
## - Bước 3: Đặt tên Project và chọn Create để tạo



Hình 22. Đặt tên project và tiến hành tạo project mới

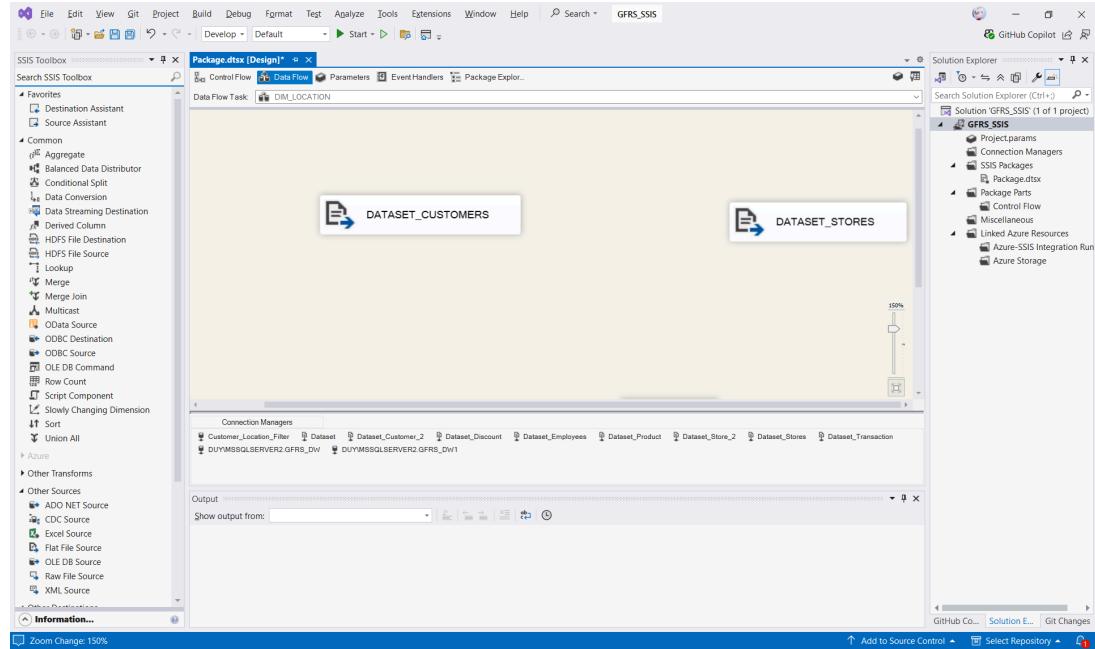
### II.3. Tạo bảng Dim và bảng Fact

- **Bước 1:** Kéo chức năng Data Flow Task trong vùng SSIS Toolbox vào vùng Control Flow và đổi tên thành **Dim\_LOCATION**.



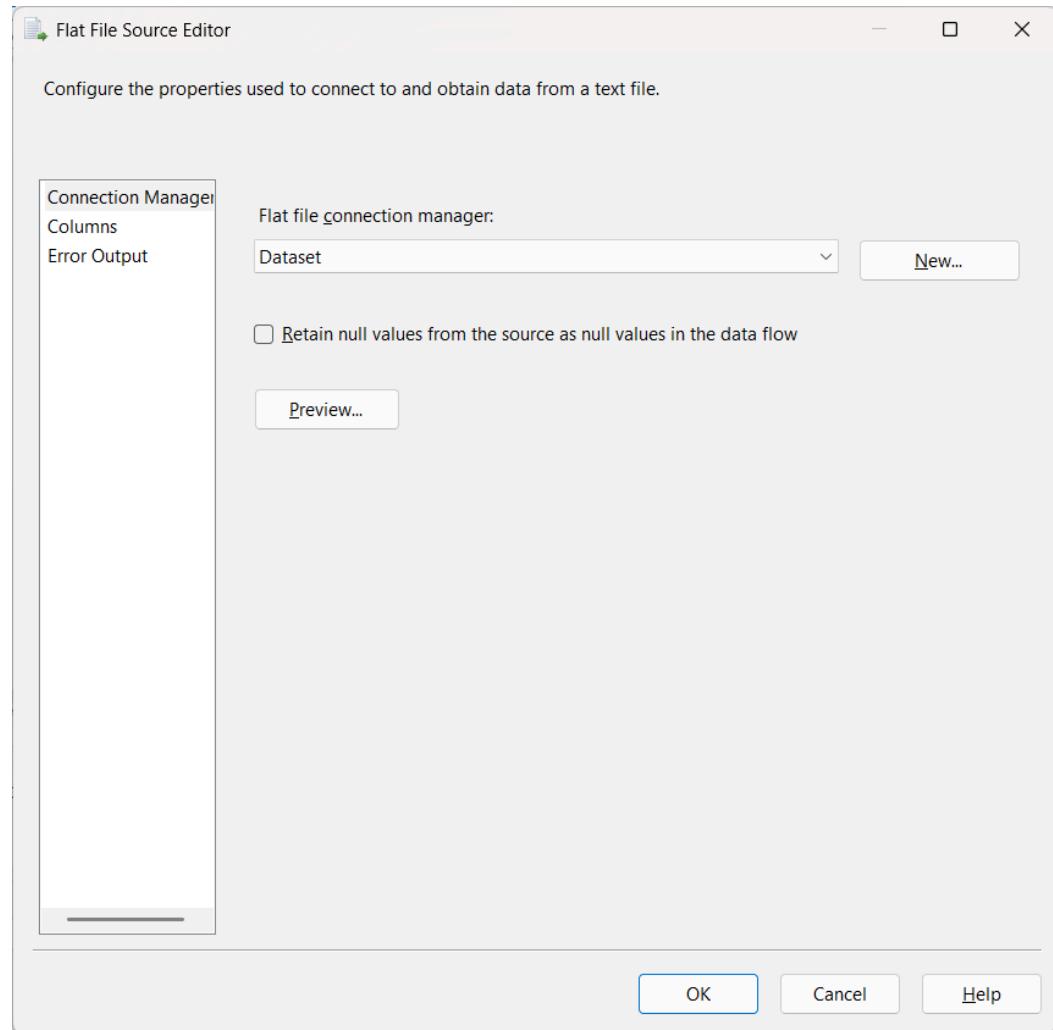
Hình 23. Khởi tạo Data Flow Task Dim\_LOCATION

- **Bước 2:** Nhấn đúp chuột vào **Dim\_LOCATION**. Sau đó kéo chức năng **Flat File Source** 2 lần vào vùng làm việc và đổi tên chúng lần lượt thành **DATASET\_CUSTOMERS** và **DATASET\_STORES**



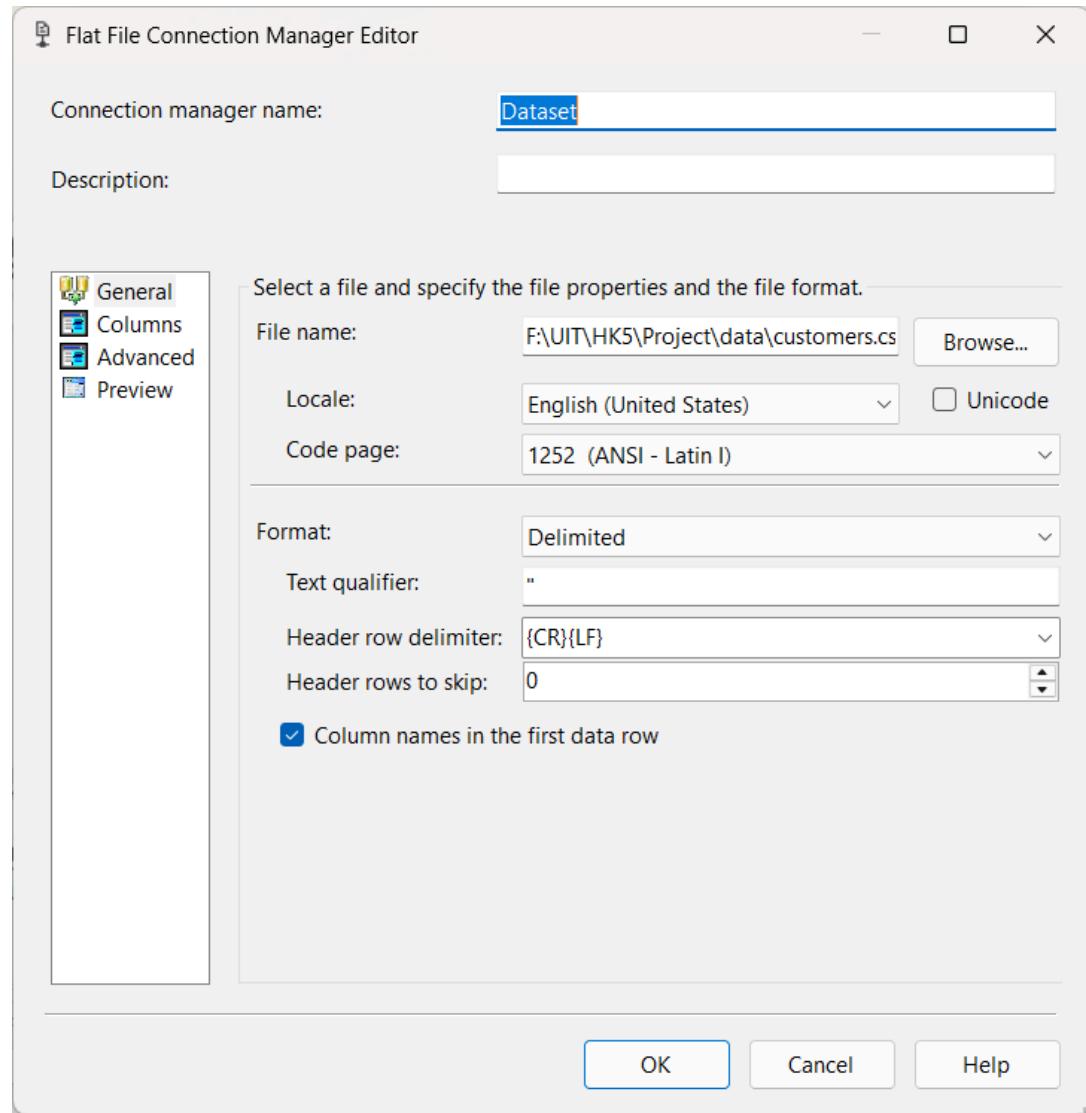
*Hình 24. Tạo Flat File Source **DATASET\_CUSTOMERS** và **DATASET\_STORES***

- **Bước 3:** Nhấn đúp chuột vào **DATASET\_CUSTOMERS**. Màn hình **Flat File Source Editor** hiện ra. Ở tab **Connection Manager** nhán **New**



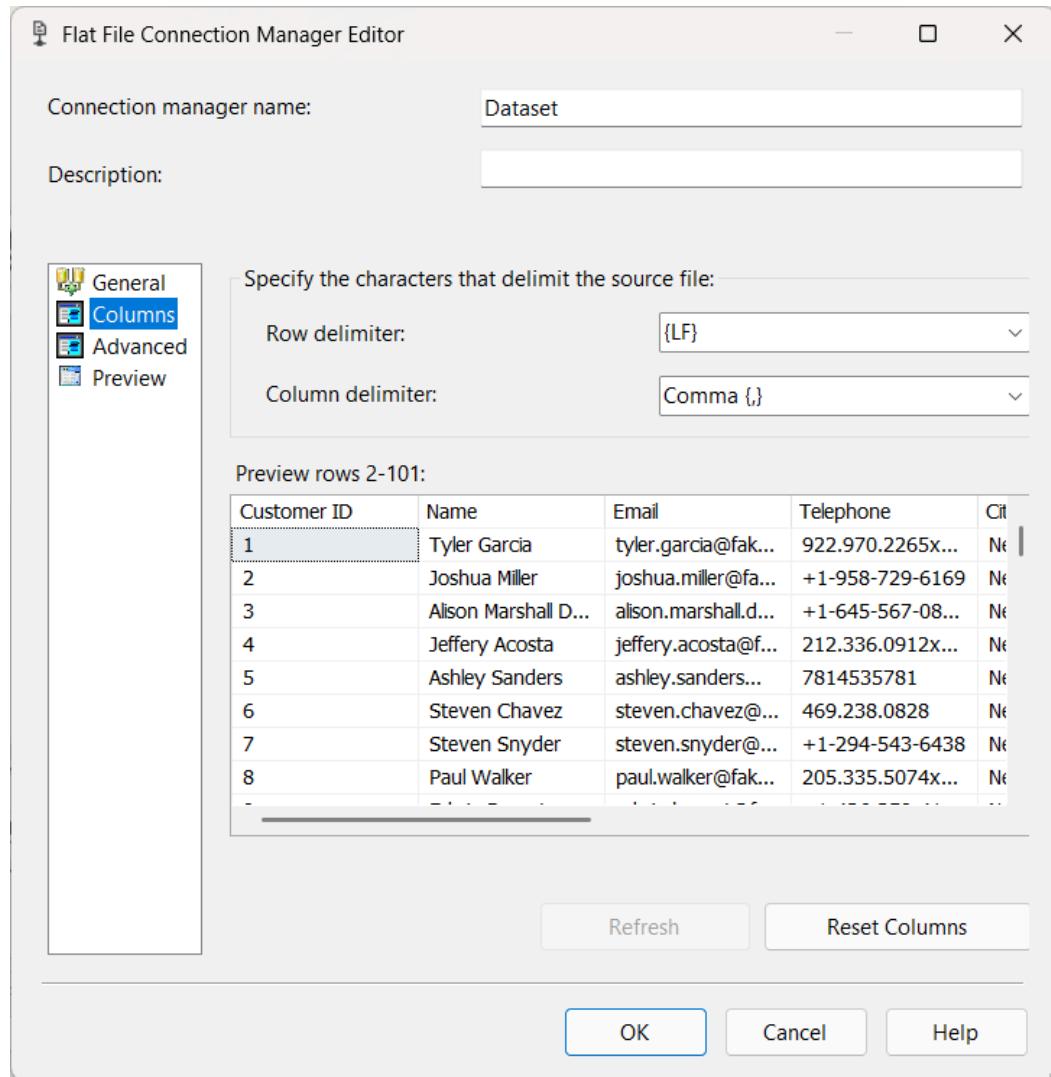
Hình 25. Kết nối nguồn dữ liệu cho DATASET\_CUSTOMERS

- **Bước 4:** Màn hình **Flat File Connection Manager Editor** hiện ra. Ở tab **General**, nhấn **Browse** để tiến hành load file csv **customers.csv** trong bộ dữ liệu **Global Fashion Retail Sales**. Ở mục **Connection manager name**, đặt tên là **Dataset**. Ở mục **Text qualifier**, nhập ký tự dấu nháy kép “



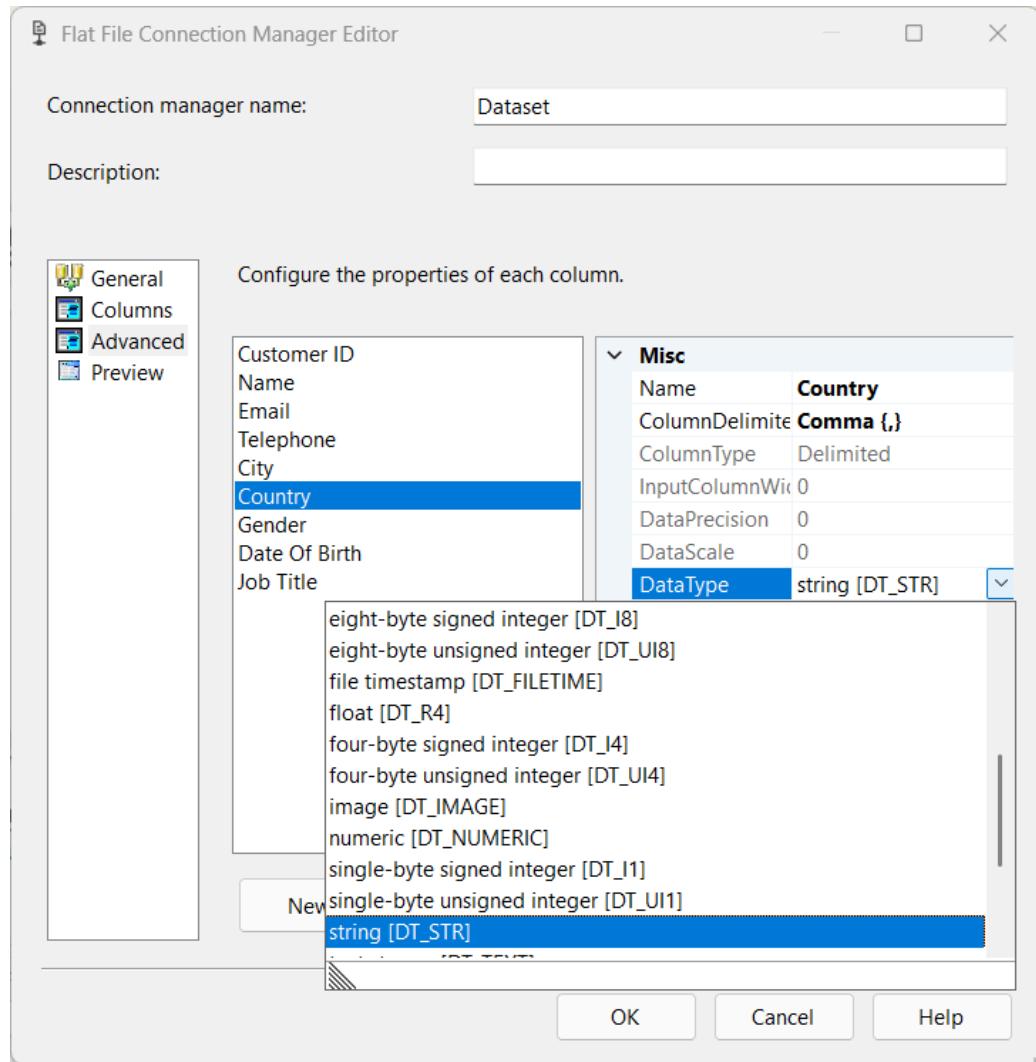
Hình 26. Tạo kết nối đến file customers.csv

- **Bước 5:** Ở tab **Columns**, ta có thể kiểm tra và thấy rằng file csv đã được load thành công



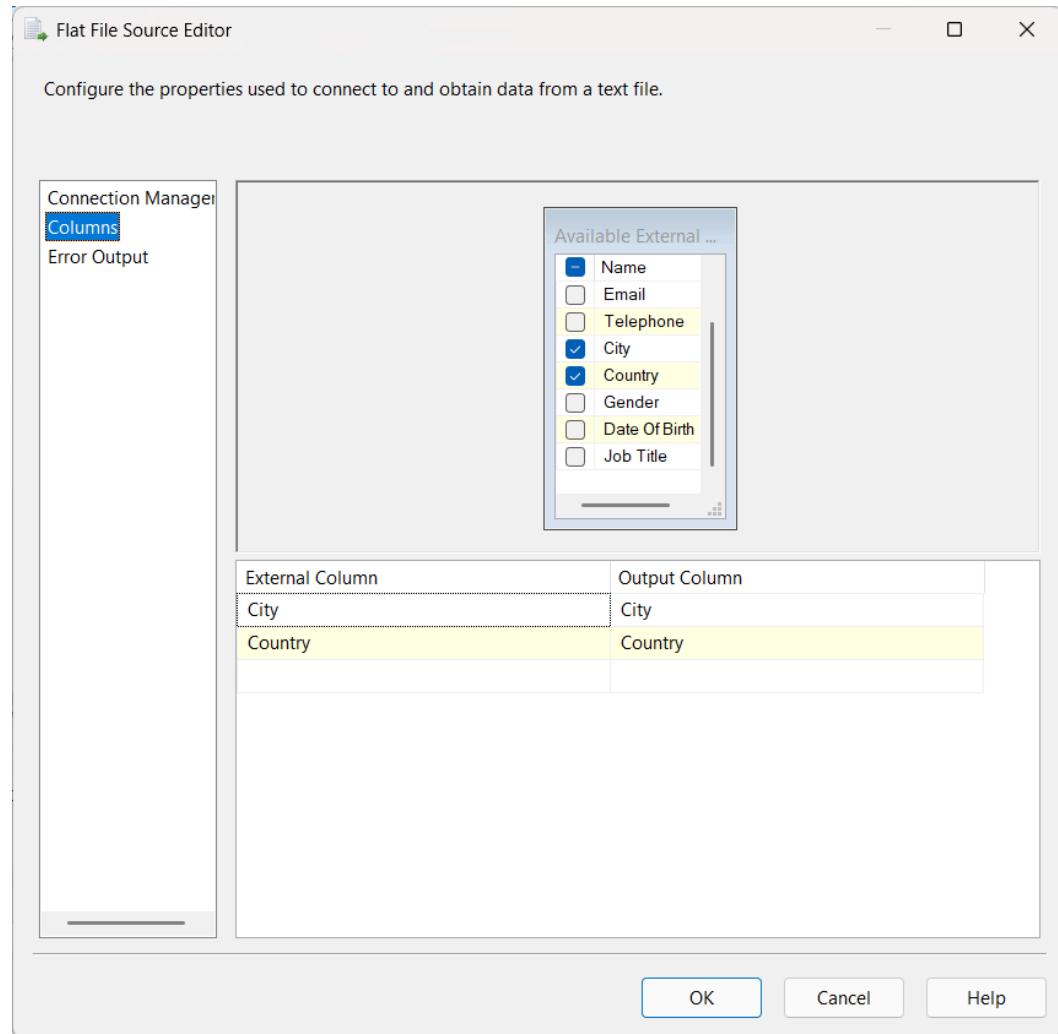
Hình 27. Kiểm tra dữ liệu file customers.csv trước khi nhập vào

- **Bước 6:** Ở tab **Advanced**, tiến hành thay đổi kiểu dữ liệu của các thuộc tính cần sử dụng trong quá trình **SSIS** của file csv tương ứng với kiểu dữ liệu khai báo của thuộc tính đó trong **SQL Server**. Sau khi thay đổi kiểu dữ liệu của các thuộc tính cần sử dụng trong quá trình **SSIS** thì nhấn **OK** và sẽ thoát khỏi màn hình **Flat File Connection Manager Editor**.



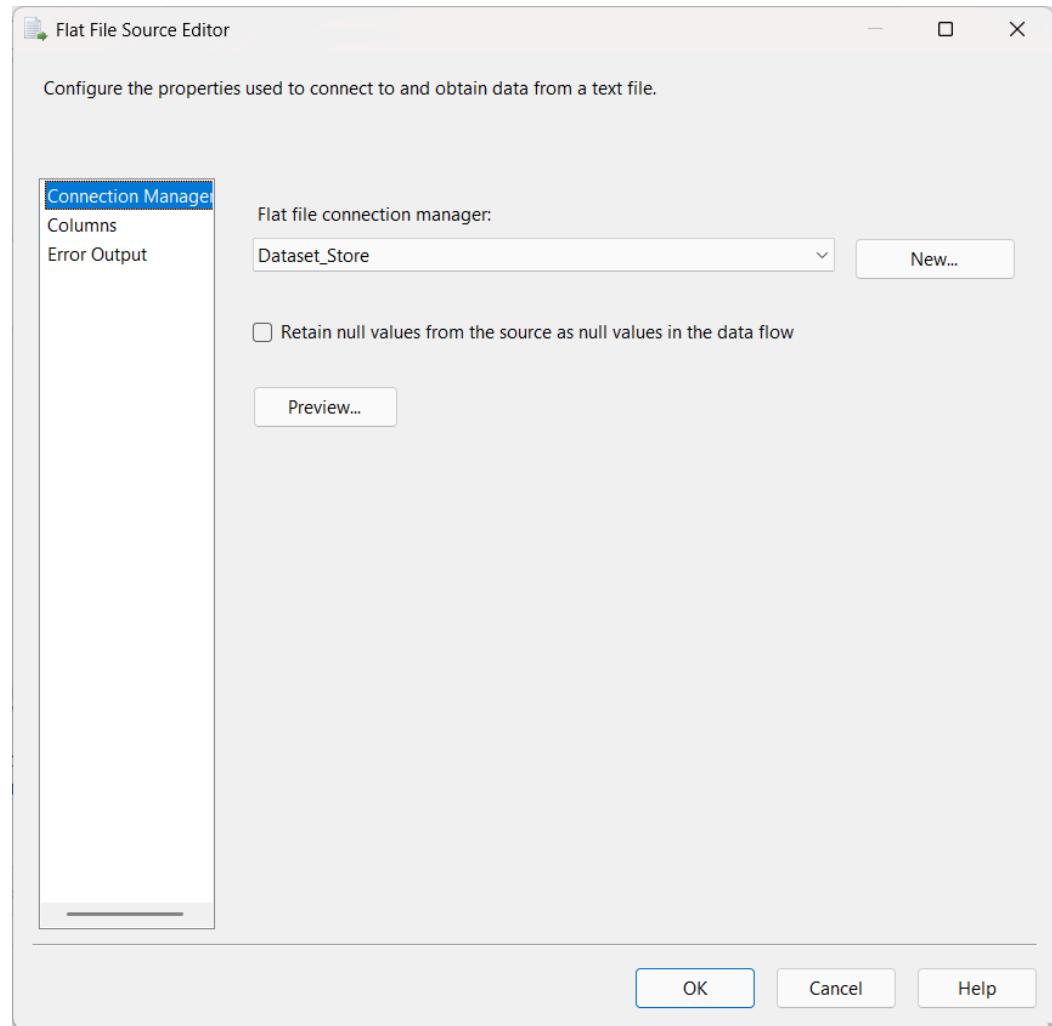
Hình 28. Cài đặt kiểu dữ liệu cho các thuộc tính trong file customers.csv

- **Bước 7:** Ở màn hình **Flat File Source Editor**, qua tab **Columns** và nhấn chọn các thuộc tính cần sử dụng trong quá trình tạo **Dim\_LOCATION** (thao tác này là tùy chọn, nhằm để hiển thị các thuộc tính cần thiết trong quá trình tạo **Dim\_LOCATION** thay vì hiển thị hết tất cả các thuộc tính trong file csv được load vào ở các bước trước đó)



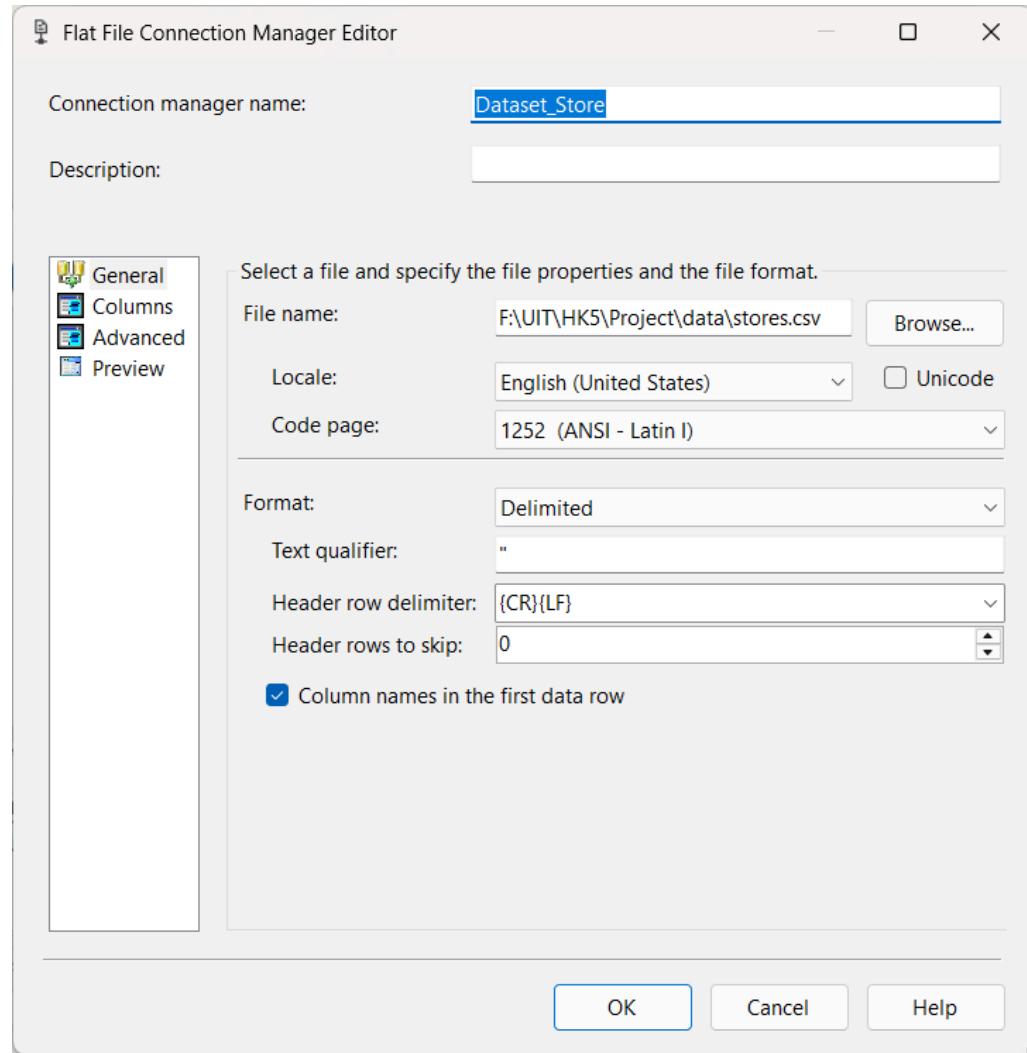
Hình 29. Lựa chọn các thuộc tính cần thiết cho việc tạo Dim\_LOCATION  
từ DATASET\_CUSTOMERS

- **Bước 9:** Nhấn OK để thoát màn hình **Flat File Source Editor**
- **Bước 10:** Nhấn đúp chuột vào **DATASET\_STORES**. Màn hình **Flat File Source Editor** hiện ra. Ở tab **Connection Manager** nhấn **New**



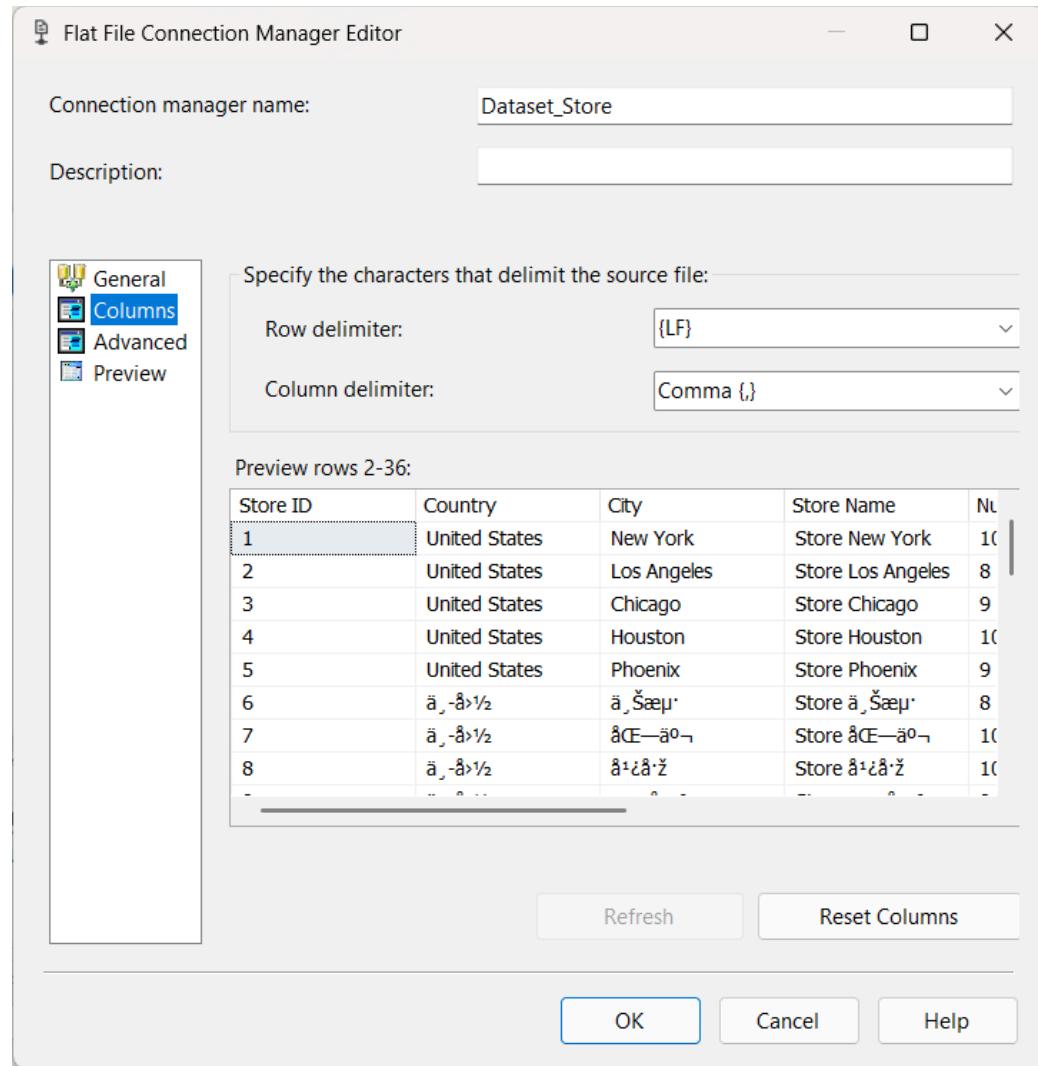
Hình 30. Kết nối với nguồn dữ liệu stores.csv

- **Bước 11:** Màn hình **Flat File Connection Manager Editor** hiện ra. Ở tab **General**, nhấn **Browse** để tiến hành load file csv **stores.csv** trong bộ dữ liệu **Global Fashion Retail Sales**. Ở **Connection manager name**, đặt tên là **DATASET\_STORES**. Ở **Text qualifier**, nhập ký tự dấu nháy kép “



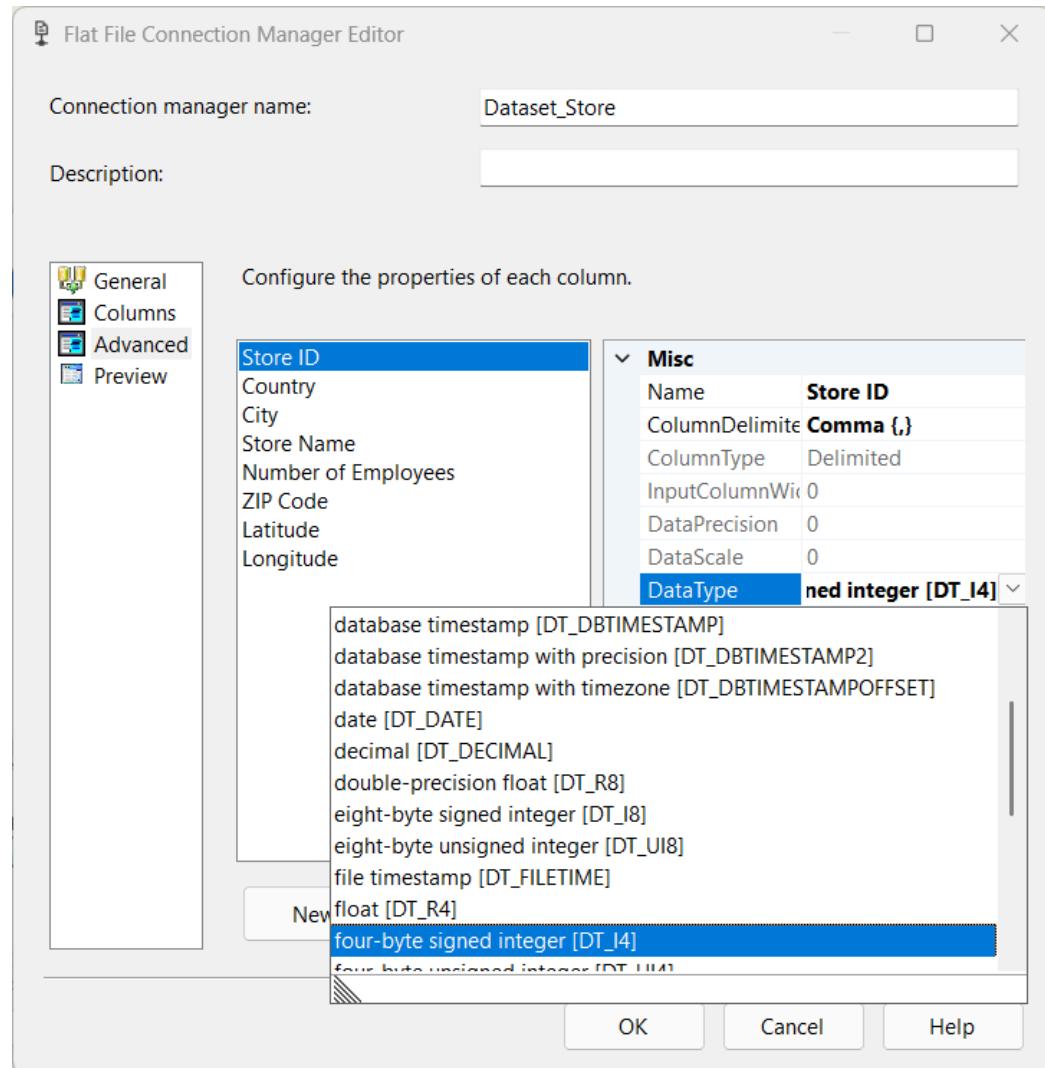
Hình 31. Khởi tạo kết nối với file stores.csv

- **Bước 12:** Ở tab **Columns**, ta kiểm tra và thấy rằng file csv đã được load thành công



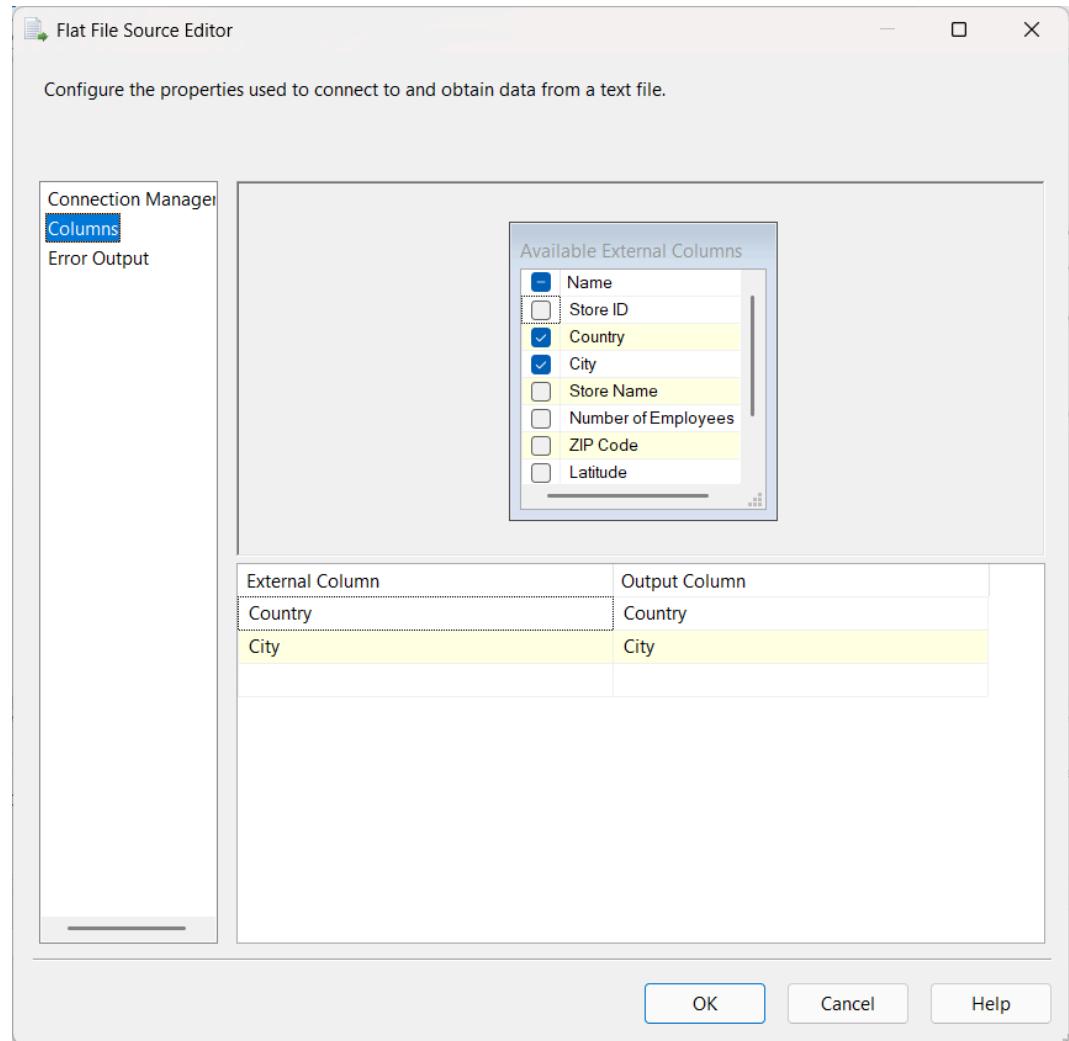
Hình 32. Kiểm tra dữ liệu file stores.csv

- **Bước 13:** Ở tab **Advanced**, tiến hành thay đổi kiểu dữ liệu của các thuộc tính cần sử dụng trong quá trình **SSIS** của file csv tương ứng với kiểu dữ liệu khai báo của thuộc tính đó trong **SQL Server**. Sau khi thay đổi kiểu dữ liệu của các thuộc tính cần sử dụng trong quá trình **SSIS** thì nhấn **OK** và sẽ thoát khỏi màn hình **Flat File Connection Manager Editor**.



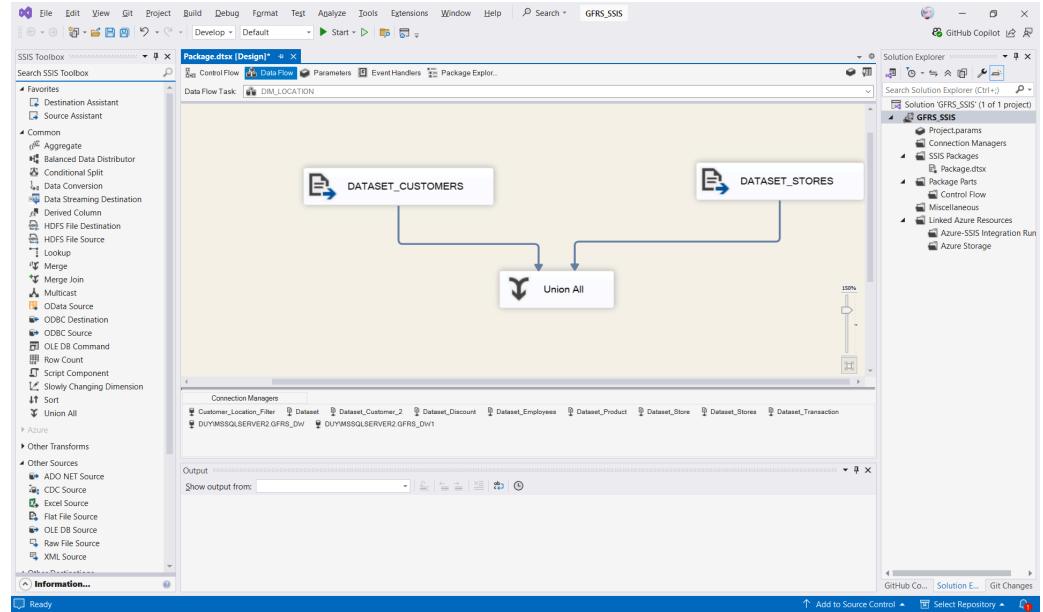
Hình 33. Cài đặt kiểu dữ liệu cho các thuộc tính trong file stores.csv

- **Bước 14:** Ở màn hình **Flat File Source Editor**, qua tab **Columns** và nhấn chọn các thuộc tính cần sử dụng trong quá trình tạo **Dim\_LOCATION**



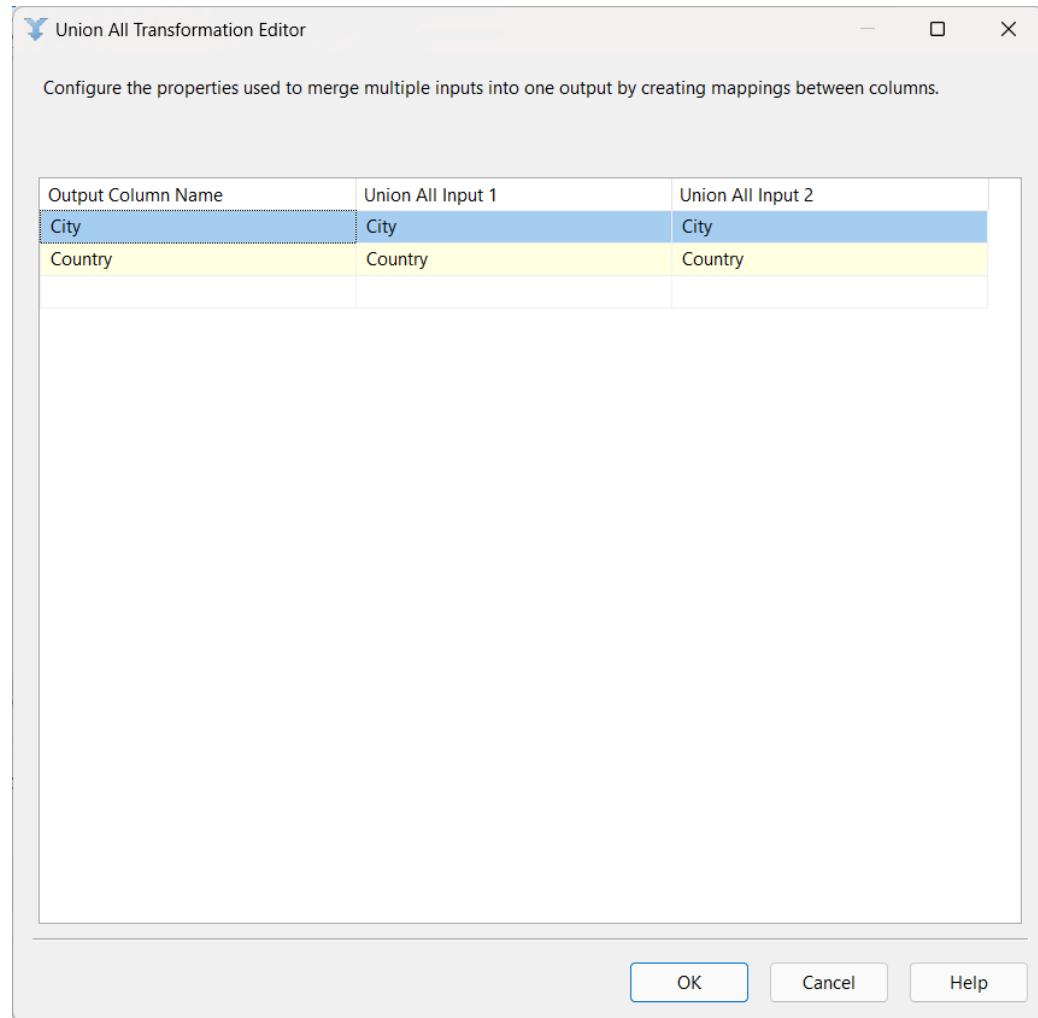
Hình 34. Lựa chọn các thuộc tính cần thiết cho việc tạo Dim\_LOCATION  
từ DATASET\_STORES

- **Bước 15:** Kéo chức năng **Union All** vào cùng **Data Flow** và kéo luồng dữ liệu xanh từ **DATASET\_CUSTOMERS** và **DATASET\_STORES** vào **Union All**



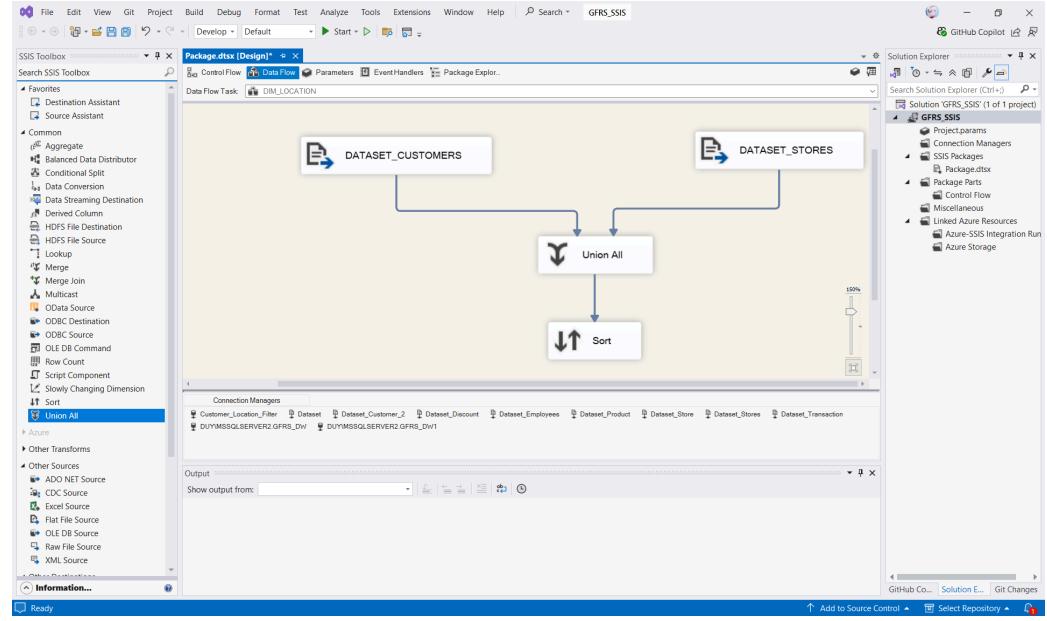
Hình 35. Thiết lập chức năng Union All trong Dim\_LOCATION

- **Bước 16:** Nhấn đúp chuột vào **Union All**, màn hình **Union All Transformation Editor** xuất hiện. Ở **Output Column Name**, đặt tên biến là **City** để lưu trữ giá trị hợp nhất từ **City (Customer)** và **City (Stores)**. Tương tự, đặt tên biến là **Country** để lưu trữ giá trị hợp nhất từ **Country (Customer)** và **Country (Stores)**. Nhấn **OK** và thoát màn hình **Union All Transformation Editor**



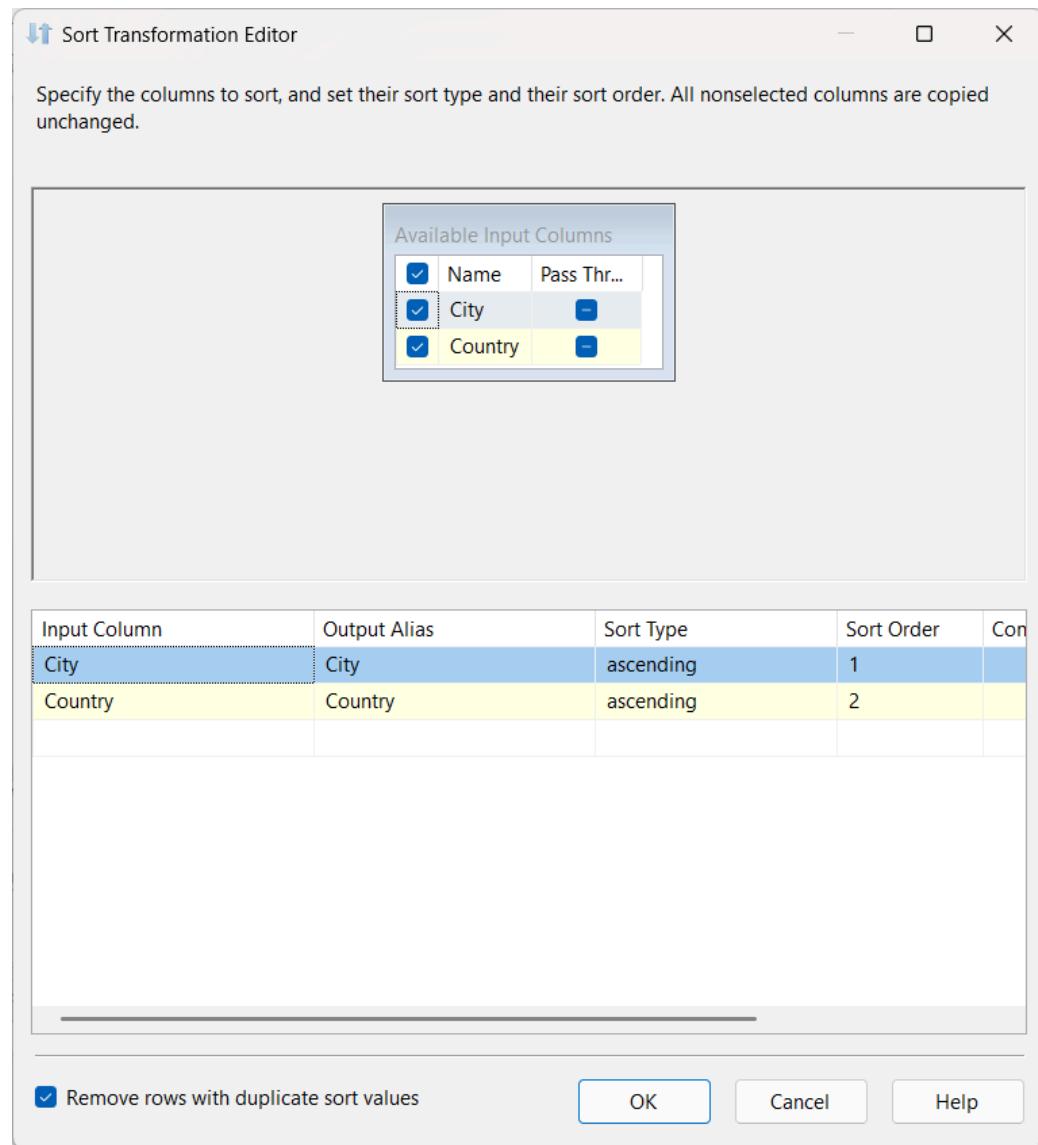
Hình 36. Cài đặt chức năng Union All trong Dim\_LOCATION

- **Bước 16:** Kéo chức năng Sort vào vùng Data Flow và kéo mũi tên luồng dữ liệu từ Union All vào Sort. Mục đích của chức năng Sort là để sắp xếp dữ liệu theo thứ tự mong muốn của người dùng, giúp ích trong việc phân tích và đưa ra quyết định, đồng thời loại bỏ các dòng dữ liệu trùng lặp trước khi đổ dữ liệu vào kho.



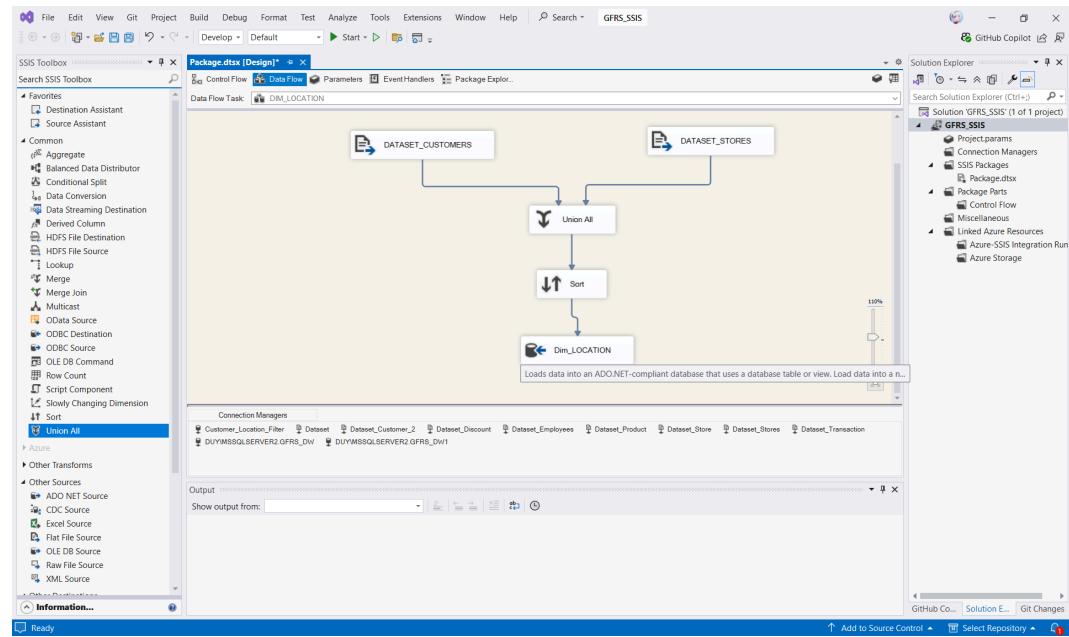
*Hình 37. Thiết lập chức năng Sort trong Dim\_LOCATION*

- **Bước 17:** Nhấn đúp chuột vào **Sort** và màn hình **Sort Transformation Editor** xuất hiện. Tiến hành click vào các thuộc tính muốn sắp xếp và click vào checkbox **Remove rows with duplicate sort values**. Nhấn **OK** và thoát màn hình **Sort Transformation Editor**.



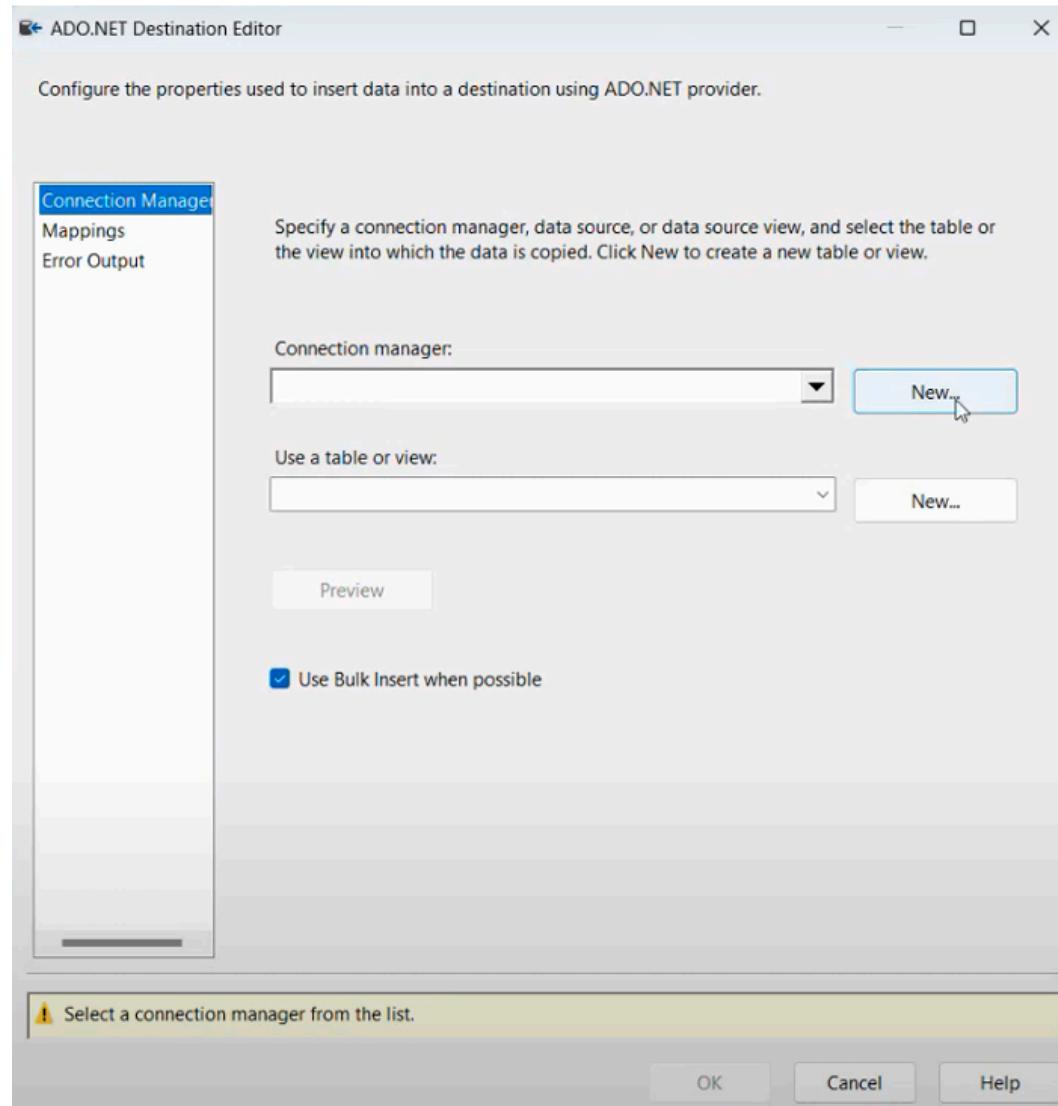
Hình 38. Cài đặt chức năng Sort trong Dim\_LOCATION

- **Bước 18:** Kéo chức năng **ADO NET Destination** vào vùng **Data Flow** và đổi tên thành **Dim\_LOCATION**. Kéo mũi tên luồng dữ liệu từ **Sort** vào **Dim\_LOCATION**.



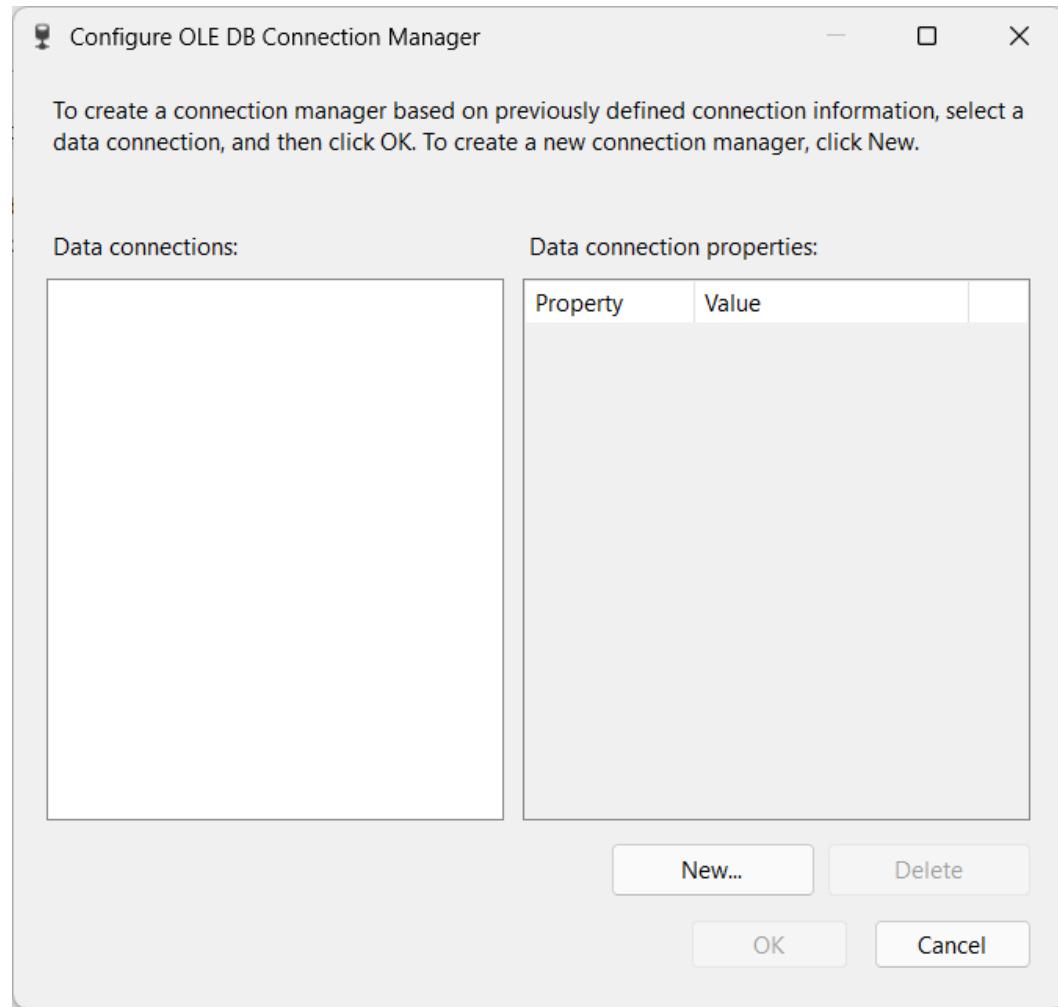
*Hình 39. Cài đặt chức năng ADO.NET Destination trong Dim\_LOCATION*

- **Bước 19:** Nhấn đúp chuột vào **Dim\_LOCATION**, màn hình **ADO.NET Destination Editor** xuất hiện. Ở tab **Connection Manager**, nhấn **New**.



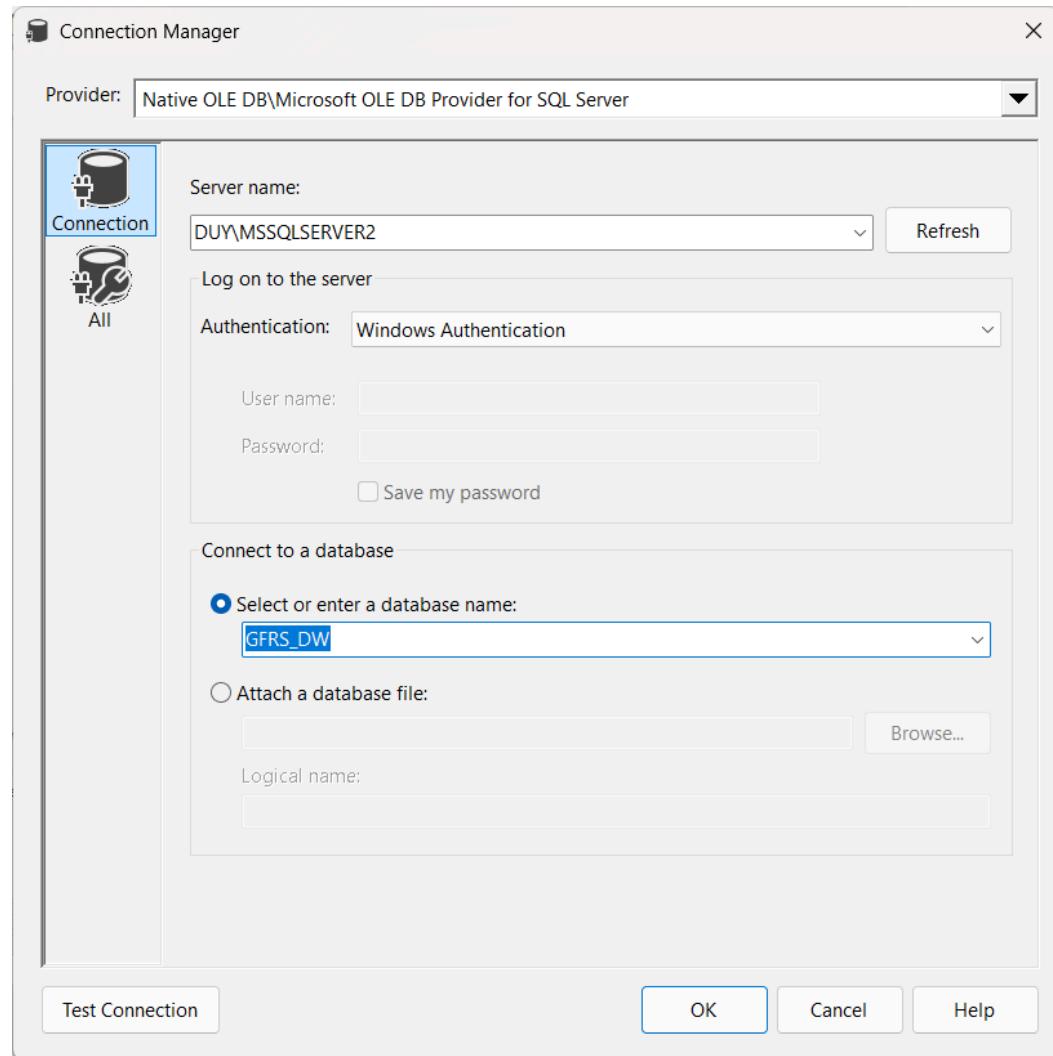
Hình 40. Tạo mới kết nối đến kho dữ liệu

- **Bước 20:** Màn hình **Configure ADO.NET Connection Manager** xuất hiện. Ta chọn **New** để tạo kết nối mới đến kho dữ liệu.



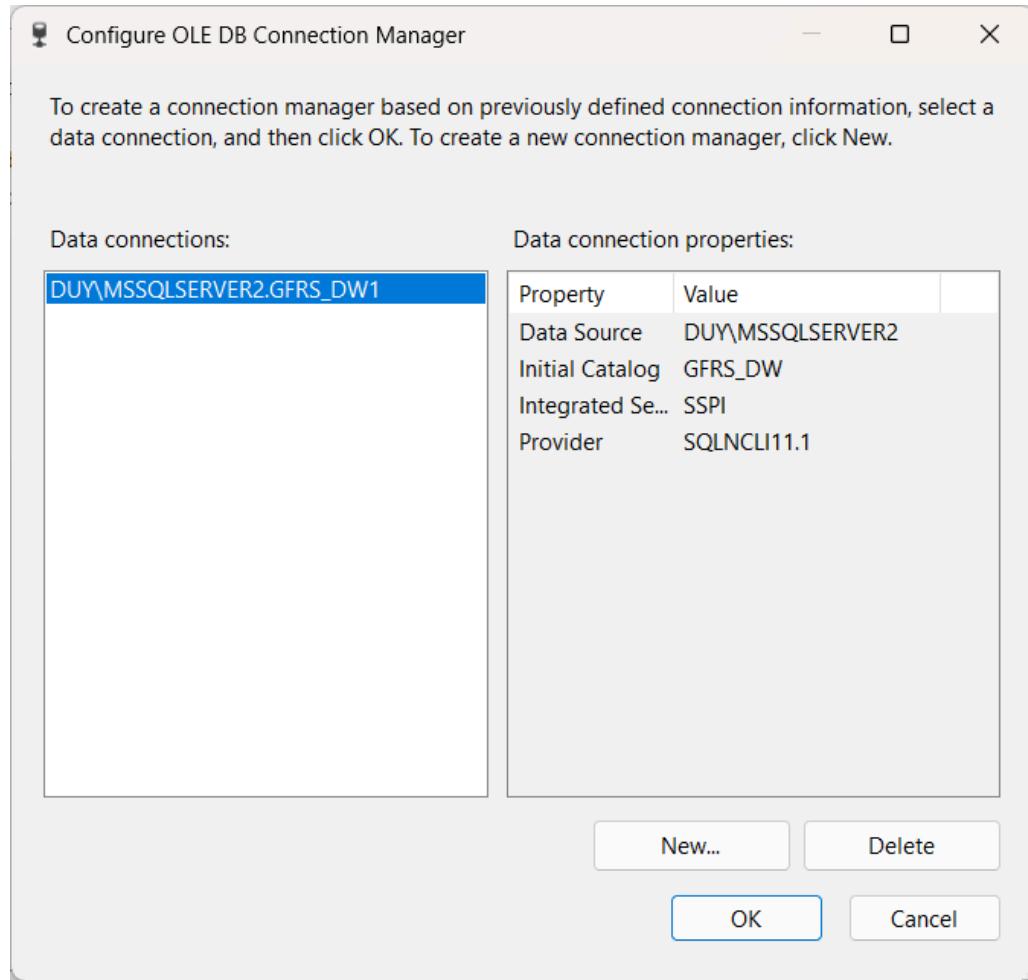
*Hình 41. Thêm kết nối mới đến kho dữ liệu*

- **Bước 21:** Màn hình **Connection Manager** xuất hiện. Ở **Provider** tiến hành chọn **Native OLE DB\Microsoft OLE DB Provider for SQL Server**. Ở **Server name** thì chọn **MSSQLSERVER2**. Chọn **GFRS\_DW** như hình minh họa và nhấn **OK**.



Hình 42. Tạo kết nối đến kho dữ liệu

- **Bước 22:** Chọn kết nối vừa tạo và nhấn **OK** để đóng cửa sổ **Configure OLE DB Connection Manager**



Hình 43. Chọn kết nối vừa tạo

- **Bước 23:** Vào SQL Server 2022, chạy lệnh DDL để tạo Dim\_LOCATION.

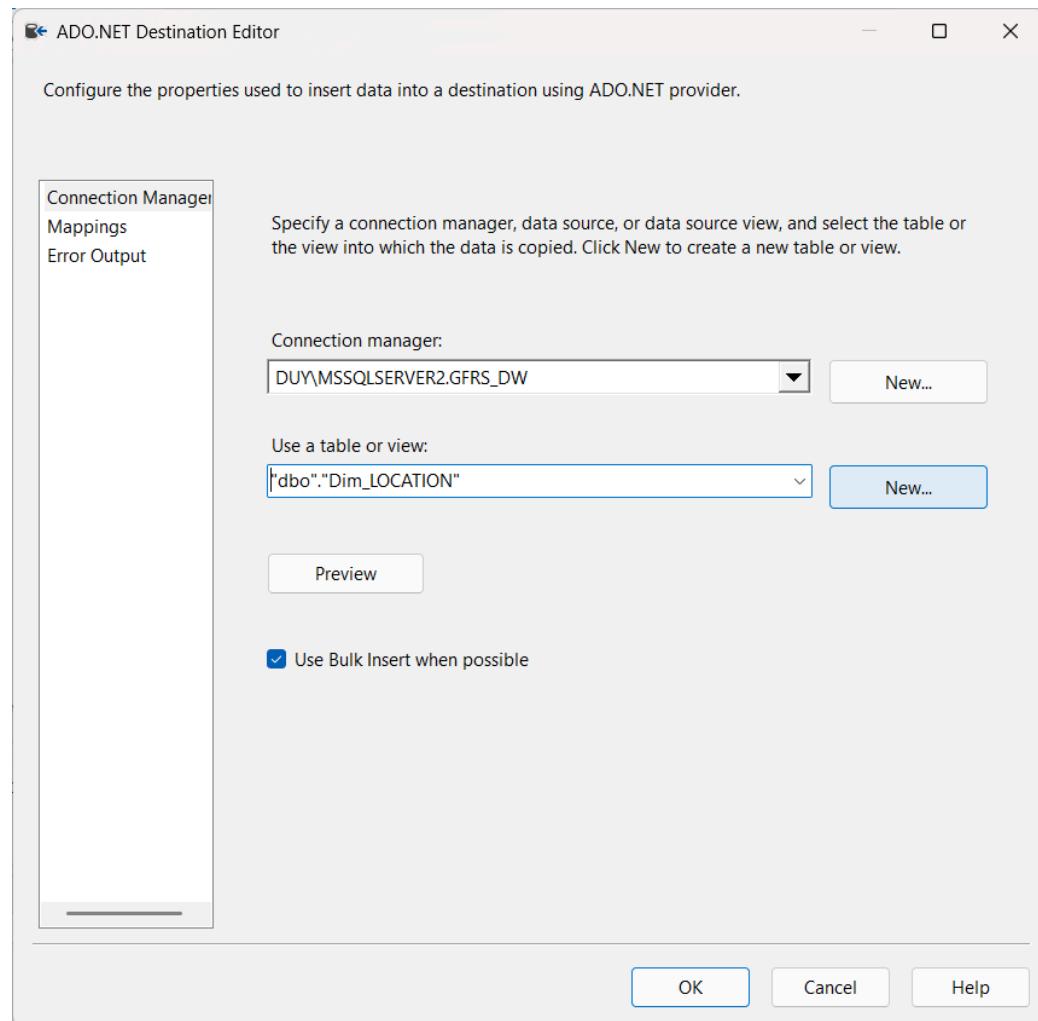
```

File Edit View Query Git Project Tools Extensions Window Help Search
File Edit View Query Git Project Tools Extensions Window Help Search
New Query Execute
Object Explorer DataBases master
DROP TABLE IF EXISTS Dim_LOCATION;
-- Table [DIM_Location]
create table Dim_LOCATION (
    location_id int identity(1, 1) primary key,
    country varchar(50),
    city varchar(50)
);

```

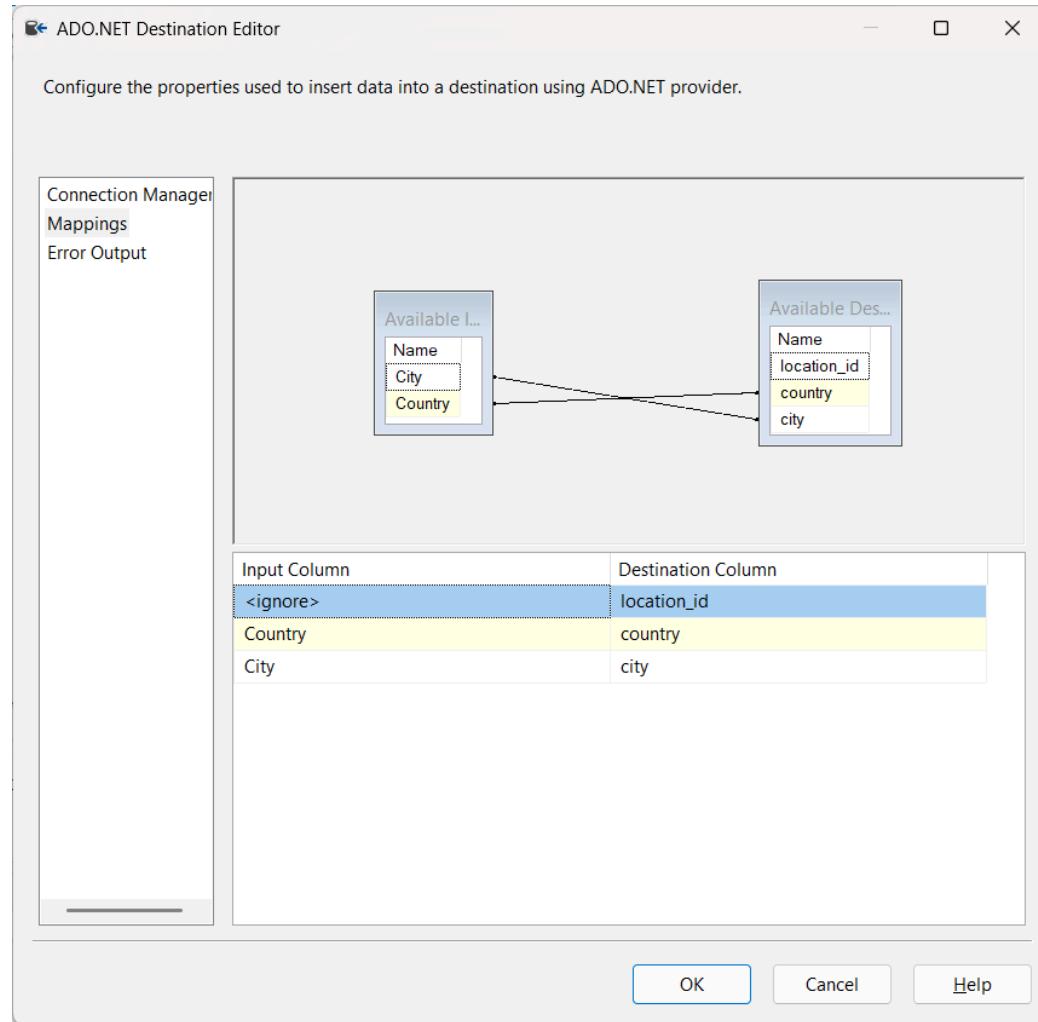
Hình 44. Chạy lệnh DDL tạo bảng Dim\_LOCATION

- **Bước 24:** Ở màn hình ADO.NET Destination Editor, ở Use a table or view tiến hành nhấn chọn “dbo”.“DIM\_LOCATION”, cũng chính là bảng vừa được tạo bằng lệnh DDL ở SQL Server 2022.



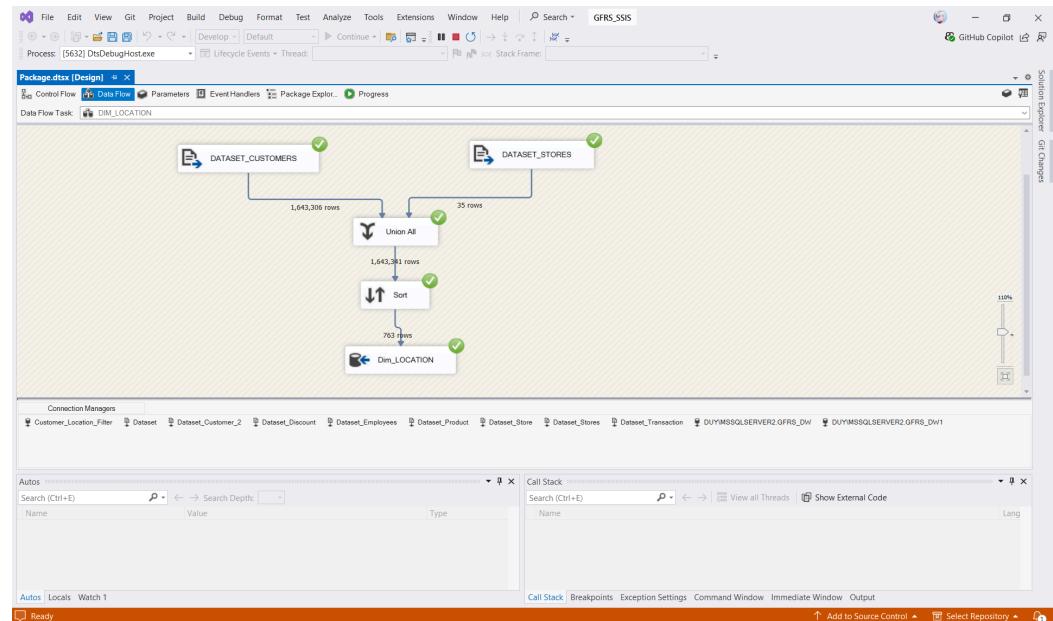
Hình 45. Chọn bảng Dim\_LOCATION đã tạo trong kho dữ liệu

- **Bước 25:** Ở tab **Mapping** của màn hình **ADO.NET Destination Editor** tiến hành chọn các thuộc tính thích hợp ở cột bên trái để ánh xạ với các thuộc tính ở cột bên phải (Các thuộc tính ở cột bên phải là những thuộc tính được tạo bằng lệnh DDL trong **SQL Server 2022**, vì vậy, ta cần phải ánh xạ để đổ dữ liệu vào các thuộc tính này). Sau đó nhấn **OK**.



Hình 46. Cài đặt trước khi đổ dữ liệu vào Dim\_LOCATION

- **Bước 26:** Nhấn Start để kiểm tra quá trình tạo và đổ dữ liệu vào **Dim\_LOCATION** có thành công hay chưa.



*Hình 47. Đỗ dữ liệu vào Dim\_LOCATION thành công*

Kết quả hiển thị cho thấy quá trình diễn ra thành công và có **763 rows** được đỗ vào **Dim\_LOCATION**.

- **Bước 27:** Vào **SQL Server 2022** kiểm tra và thấy dữ liệu đã được đỗ vào **Dim\_LOCATION** thành công.

The screenshot shows the SSMS interface with the following details:

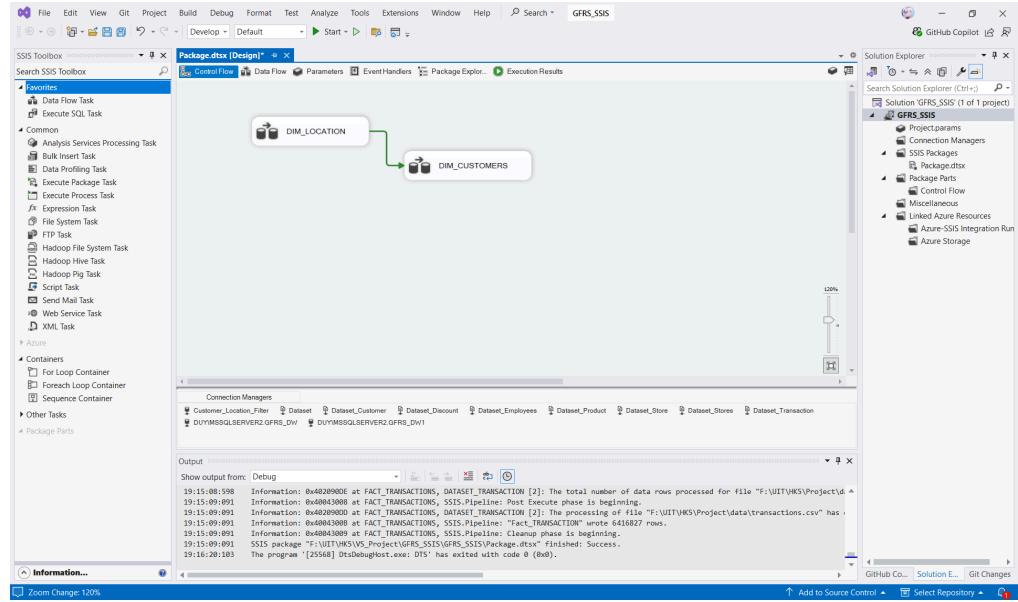
- Object Explorer:** Shows the connection to DUY\MSSQLSERVER2 (SQL Server 16.0.1000.6 - DUY\main) and the GFRS\_DW database.
- SQL Query Editor:** Contains two queries:
  - Query 1 (lines 1-15): Creates the Dim\_LOCATION table with columns location\_id (int identity), country (varchar(50)), and city (varchar(50)). It also includes a select statement to show all data from the table.
  - Query 2 (lines 16-20): Creates the Dim\_CUSTOMER table with columns CustomerID (BIGINT identity), customerHash (bigint), Gender (VARCHAR(50)), and JobTitle (VARCHAR(100)).
- Results Grid:** Displays the data for the Dim\_LOCATION table, showing 763 rows. The columns are location\_id, country, and city.
- Status Bar:** Shows "Query executed successfully" and other session details.

location_id	country	city
82	Espa...	Alcobe...
83	Espa...	Alcoy
84	Portu...	Alequ...
85	Unite...	Alexan...
86	Espa...	Alhaur...
87	Espa...	Alicante
88	France	Allauch
89	Portu...	Almada
90	Deut...	Altona
91	Unite...	Altrinc...
92	Portu...	Amad...
93	Unite...	Anahei...
94	Unite...	Annan...
95	Espa...	Antequ...
96	France	Antibes
97	Unite...	Antrim
98	8,-8½	çŒ-çœ-

Hình 48. Kết quả đỗ dữ liệu vào Dim\_LOCATION

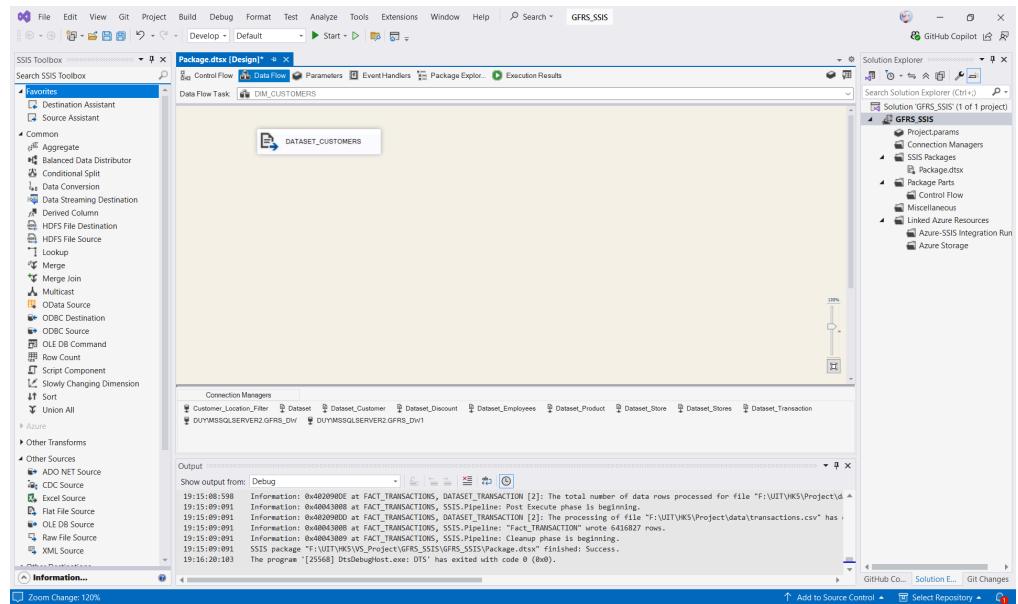
### II.3.1. Dim\_CUSTOMERS

- **Bước 1:** Kéo chức năng **Data Flow Task** trong vùng **SSIS Toolbox** vào vùng **Control Flow** và đổi tên thành **Dim\_CUSTOMERS**.



Hình 49. Khởi tạo Data Flow Task Dim\_CUSTOMER

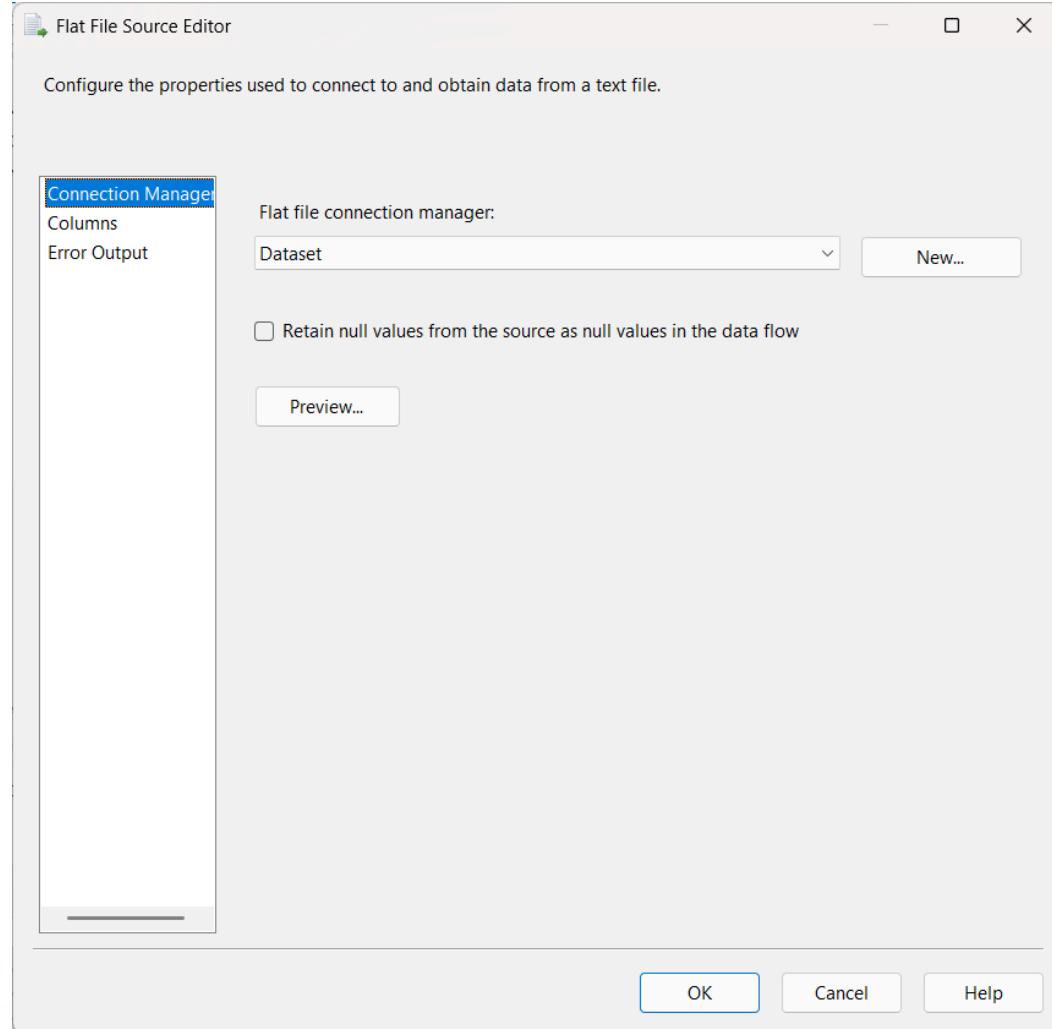
- **Bước 2:** Nhấn đúp chuột vào **Dim\_CUSTOMER**. Sau đó kéo chúc nǎng **Flat File Source** vào vùng làm việc và đổi tên thành **DATASET\_CUSTOMER**



Hình 50. Khởi tạo DATASET\_CUSTOMER trong Dim\_CUSTOMER

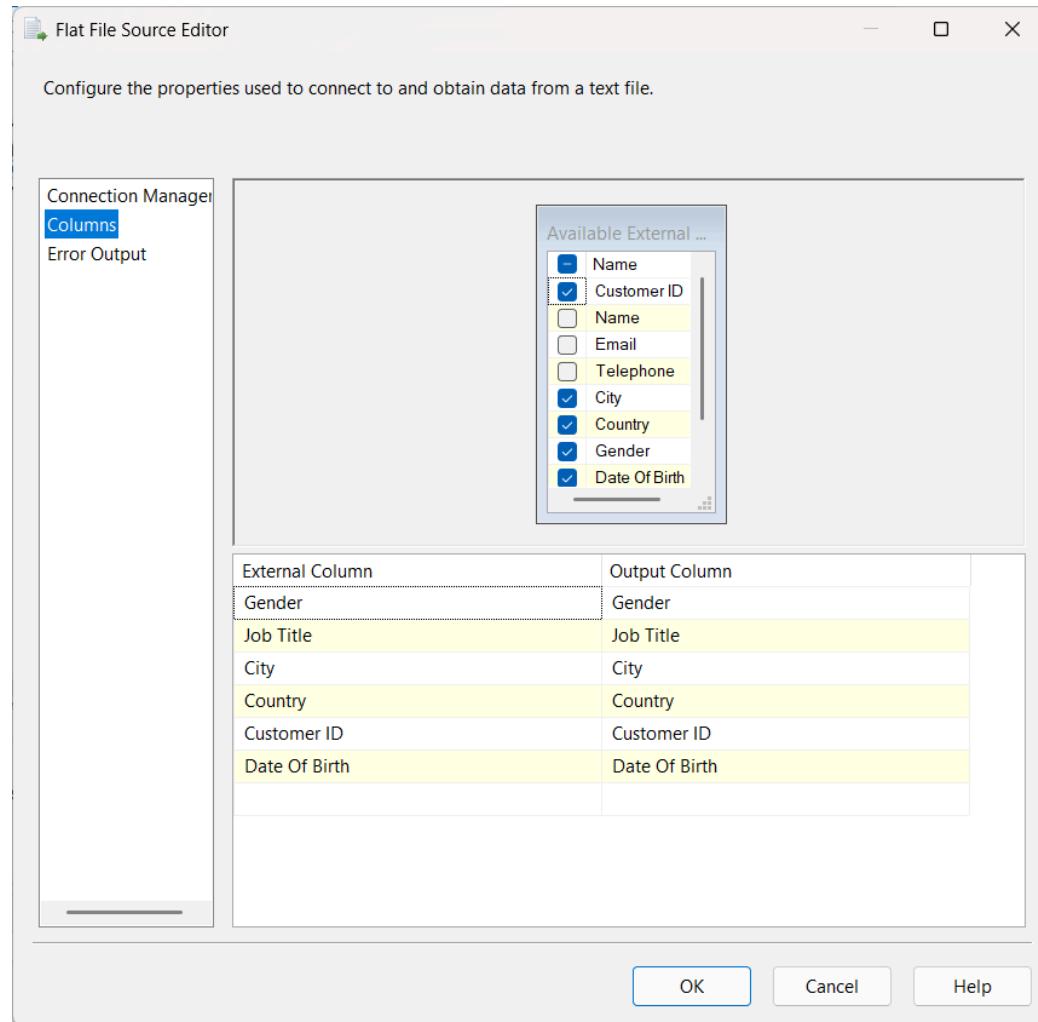
- **Bước 3:** Nhấn đúp chuột vào **DATASET\_CUSTOMER**. Màn hình **Flat File Source Editor** hiện ra. Do quá trình tạo **Dim\_LOCATION**, ở các bước trước đã load thành công file csv và đổi tên thành **Dataset** nên ta có

thể sử dụng lại **Dataset** (nguồn dùng để load file customer.csv đã tạo trước đó) để load dữ liệu mà không cần thiết lập lại giúp tiết kiệm thời gian và giảm nguy cơ gặp lỗi.



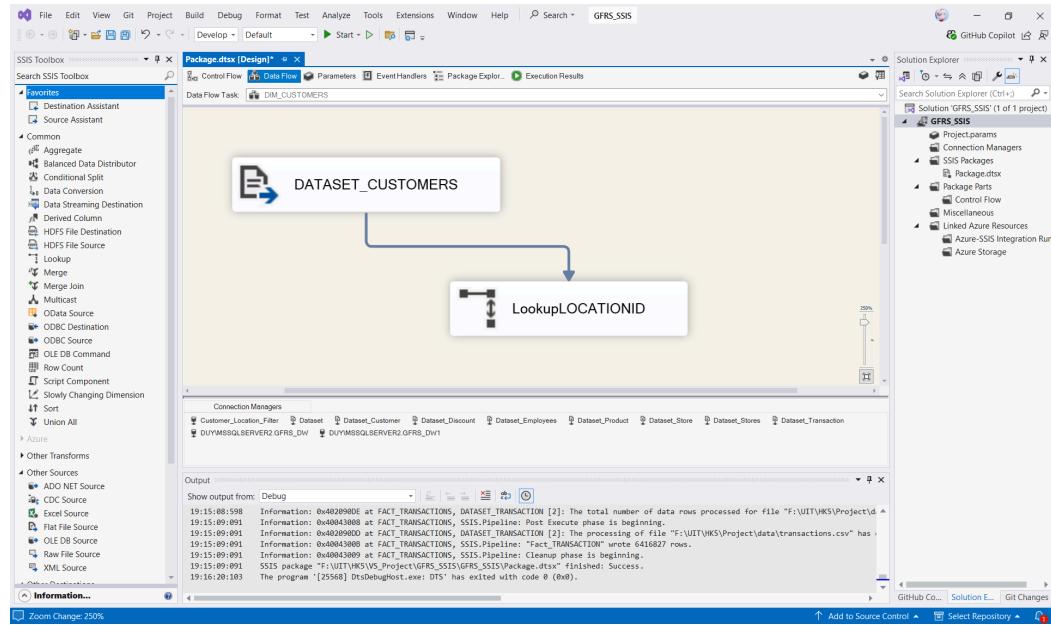
Hình 51. Chọn Dataset đã được thiết lập trước đó

- **Bước 4:** Chuyển sang tab **Columns**, ta có thể lựa chọn các thuộc tính cần thiết cho quá trình nạp dữ liệu cho **Dim\_CUSTOMERS**. Sau đó ta có thể bấm **OK** để đóng cửa sổ **Flat File Source Editor**



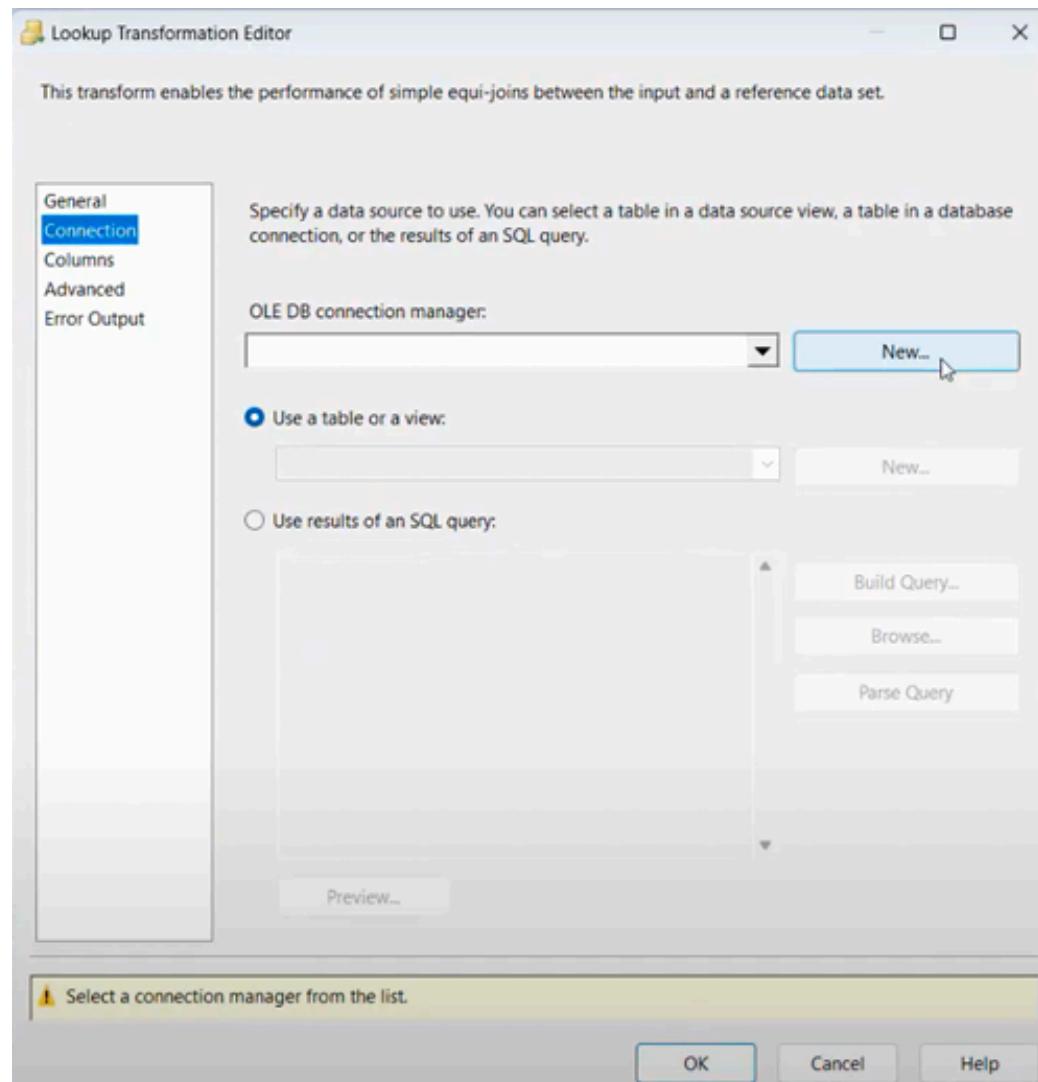
Hình 52. Lựa chọn thuộc tính cần thiết cho quá trình đổ dữ liệu vào  
Dim\_CUSTOMERS

- **Bước 5:** Kéo chức năng **Lookup** vào vùng **Data Flow** và đổi tên thành **LookupLOCATIONID** (dùng để đổi chiểu **LocationID**). Kéo mũi tên luồng dữ liệu từ **DATASET\_CUSTOMERS** vào **LookupLOCATIONID**.



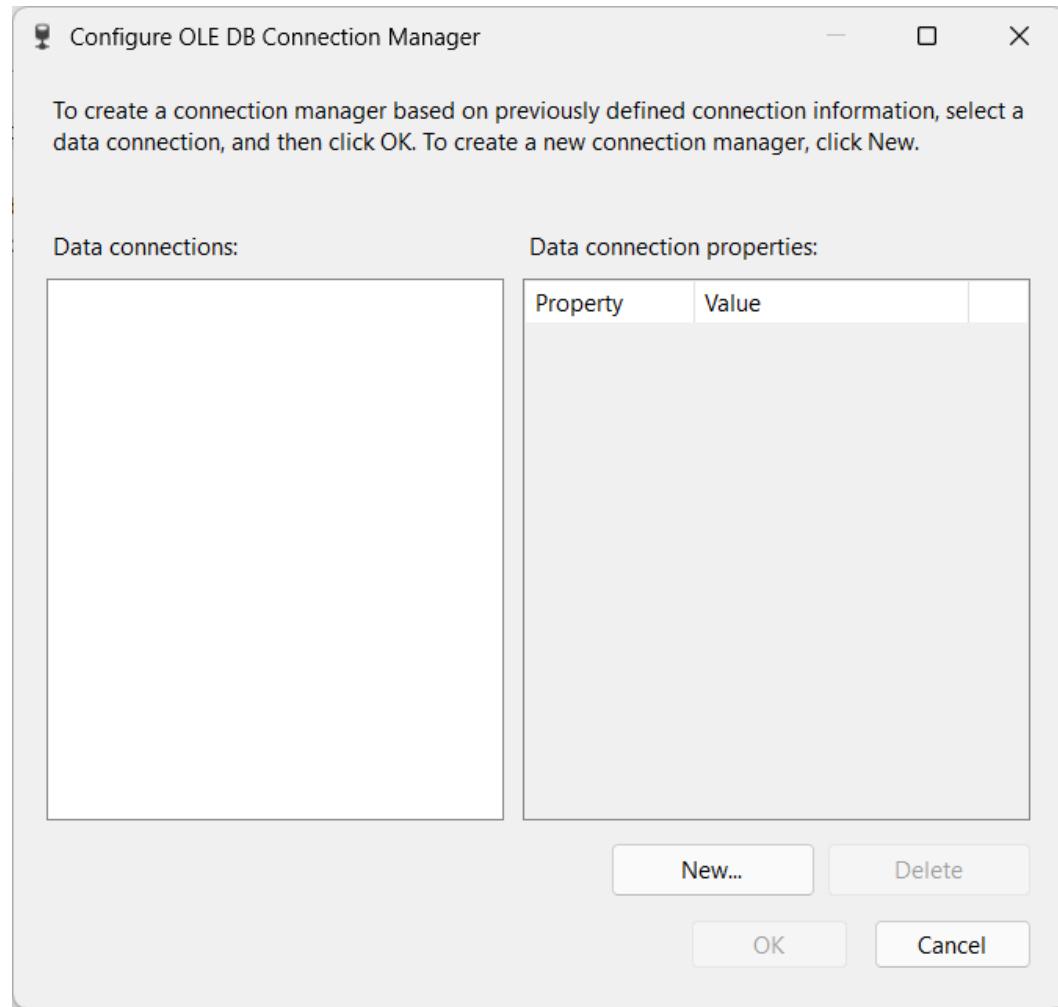
*Hình 53. Khởi tạo chức năng LookupLOCATIONID trong Dim\_CUSTOMERS*

- **Bước 6:** Nhấn đúp chuột vào **LookupLOCATIONID**, màn hình **Lookup Transformation Editor** sẽ hiện lên. Ta chọn **New** để tạo kết nối mới đến kho dữ liệu



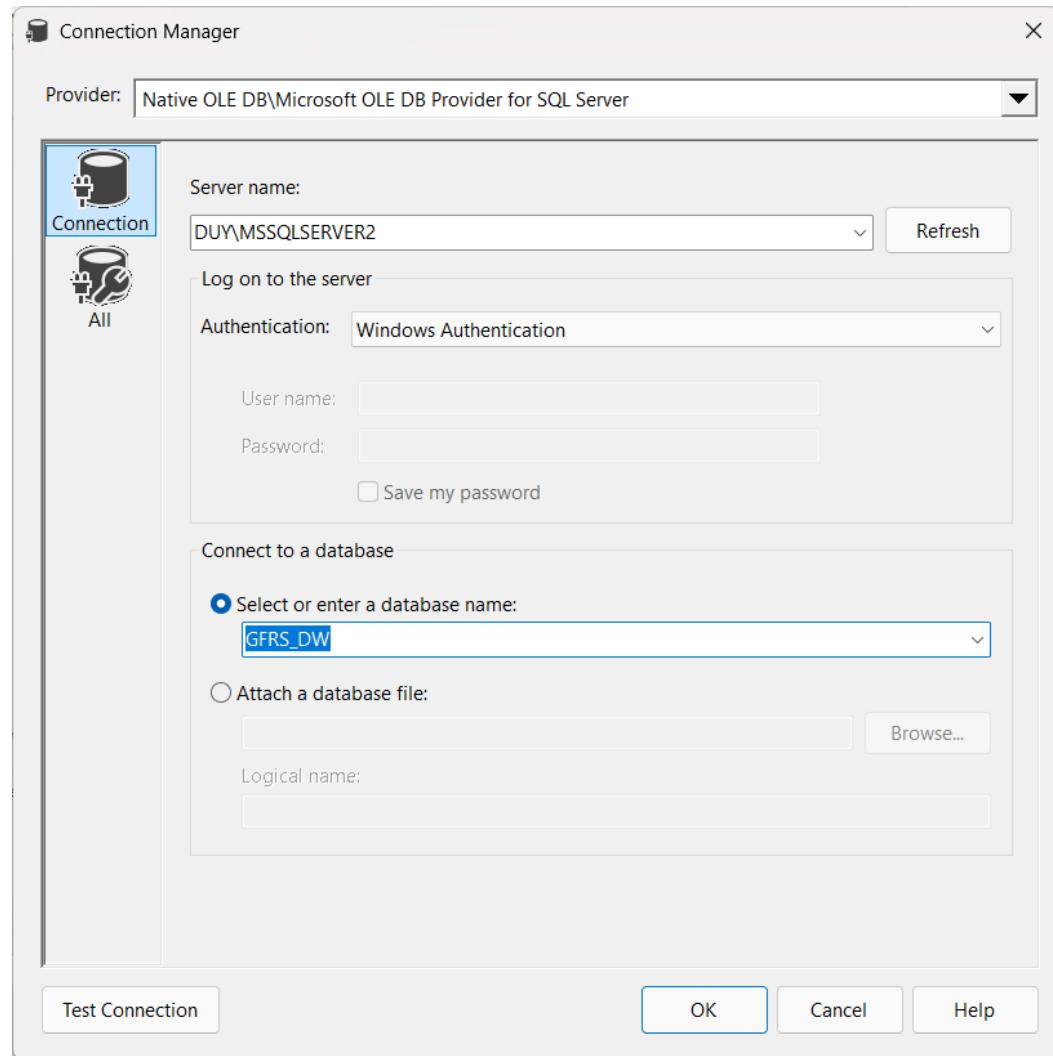
Hình 54. Tạo kết nối đến cơ sở dữ liệu

- **Bước 7:** Màn hình **Configure OLE DB Connection Manager** xuất hiện.  
Nhấn **New** như hình minh họa.



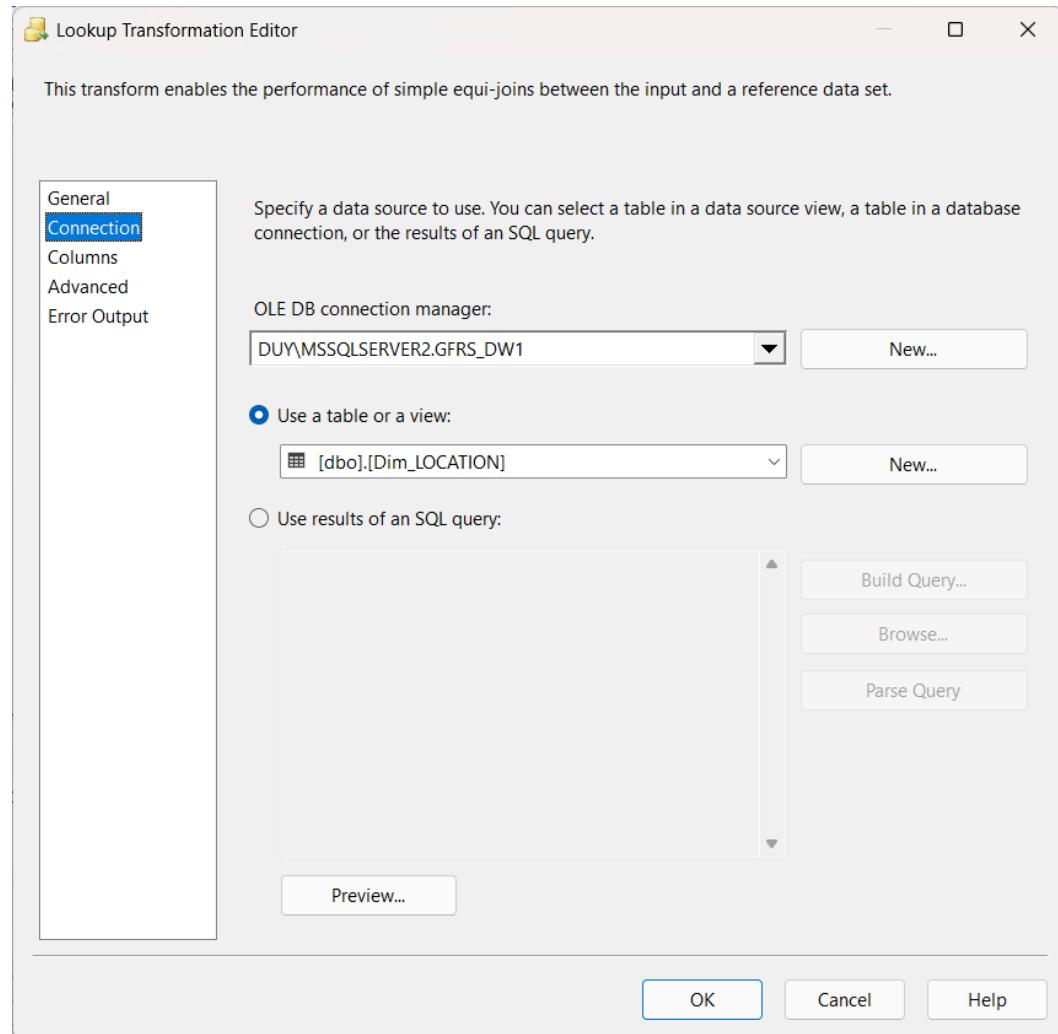
Hình 55. Tạo kết nối lookup đến cơ sở dữ liệu

- **Bước 8:** Màn hình **Connection Manager** xuất hiện. Ở **Provider** tiến hành chọn **Native OLE DB\Microsoft OLE DB Provider for SQL Server**. Ở **Server name** thì chọn **DUY\MSSQLSERBER2**. Chọn **GFRS\_DW** như hình minh họa và nhấn **OK**.



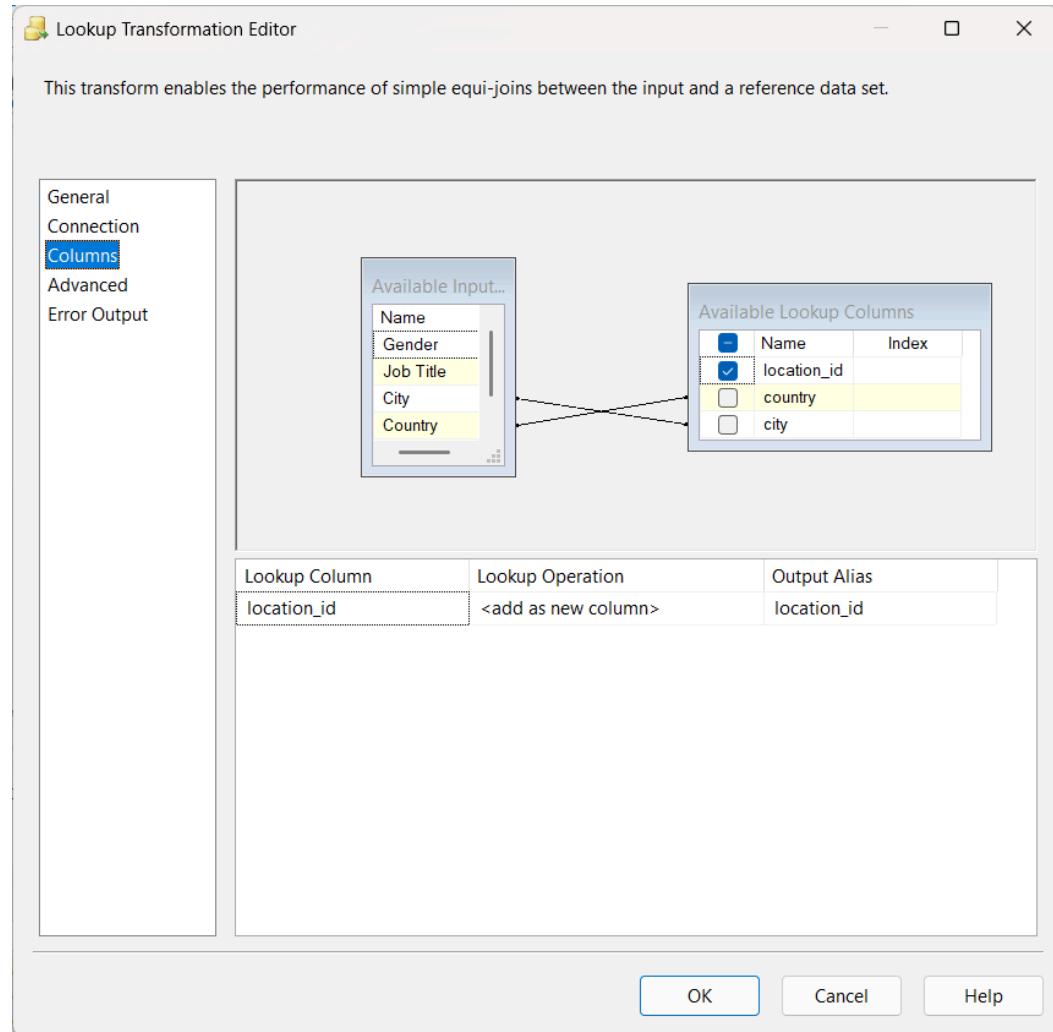
Hình 56. Tạo kết nối đến kho dữ liệu

- **Bước 9:** Ta chọn kết nối đã tạo trước đó **DUY\MSSQLSERVER2.GFRS\_DW1** (do cùng tên với kết nối trước nên hệ thống tự thêm “1” vào sau tên), ở **Use a table or a view**, tiến hành chọn **[dbo].[Dim\_LOCATION]**



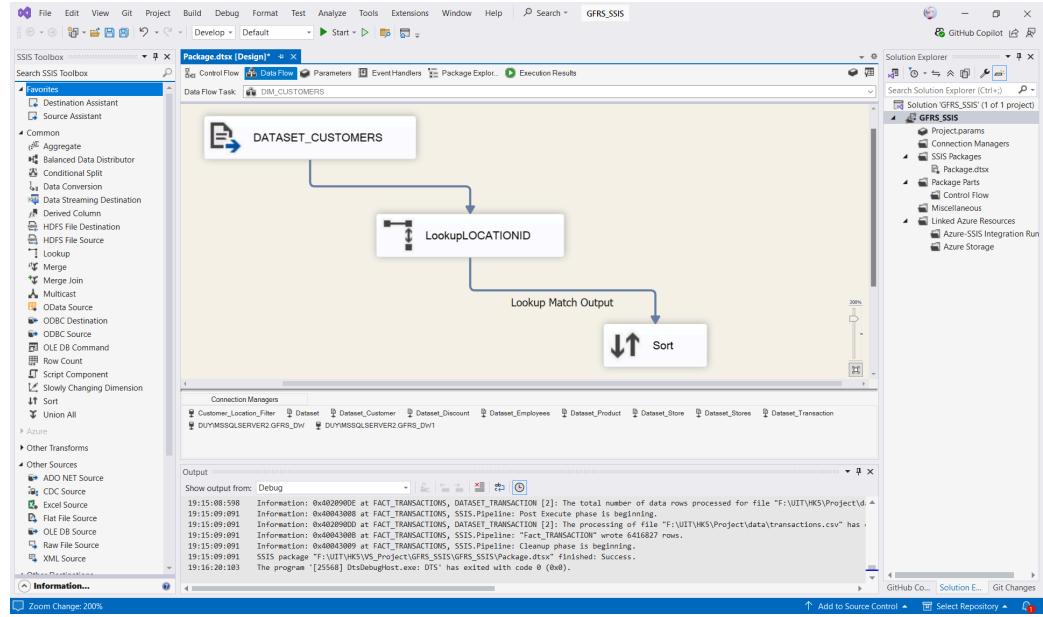
Hình 57. Chọn kết nối và bảng cần lookup

- **Bước 10:** Trong tab **Columns**, ở bảng bên phải click chọn **location\_id**. Ta kéo **City, Country** từ bảng **Available Input Columns** qua tương ứng **city, country** bảng **Available Lookup Columns** (quá trình này giúp hệ thống xác định vùng dữ liệu cần tìm kiếm). Sau đó bấm **OK** để đóng cửa sổ **Lookup Transformation Editor**



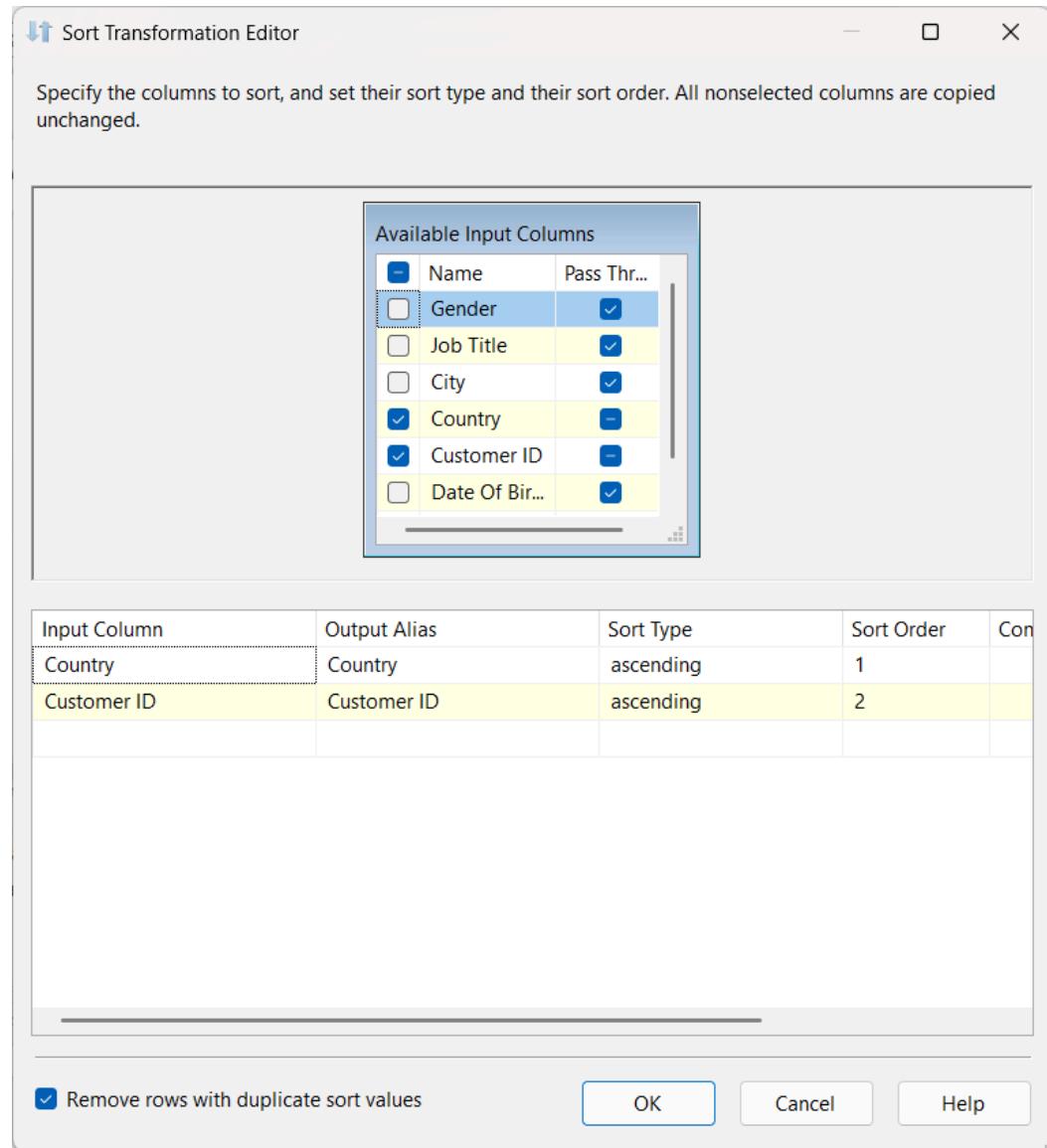
Hình 58. Cài đặt LookupLOCATIONID

- **Bước 11:** Kéo chức năng **Sort** vào vùng **Data Flow** và kéo mũi tên luồng dữ liệu từ **LookupLOCATIONID** vào **Sort**. Mục đích của chức năng **Sort** là để sắp xếp dữ liệu theo thứ tự mong muốn của người dùng, giúp ích trong việc phân tích và đưa ra quyết định, đồng thời loại bỏ các dòng dữ liệu trùng lặp trước khi đổ dữ liệu vào kho. Một hộp thoại sẽ xuất hiện, chọn **Lookup Match Output** và nhấn **OK**.



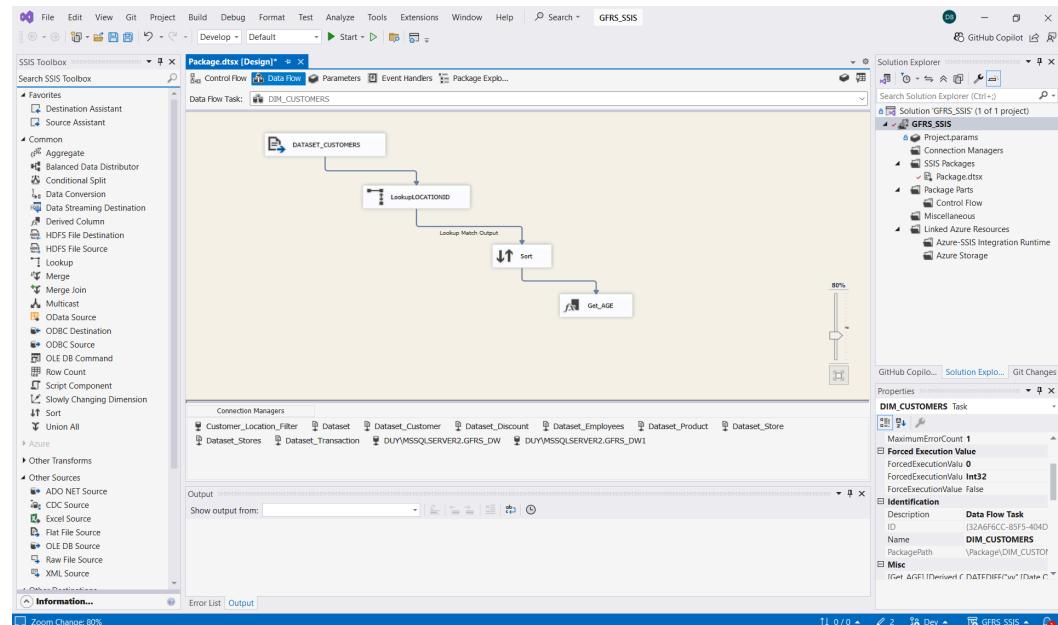
Hình 59. Tạo chức năng Sort trong Dim\_CUSTOMERS

- **Bước 12:** Nhấn đúp chuột vào **Sort** và màn hình **Sort Transformation Editor** xuất hiện. Nhấn **OK** và thoát màn hình **Sort Transformation Editor**.



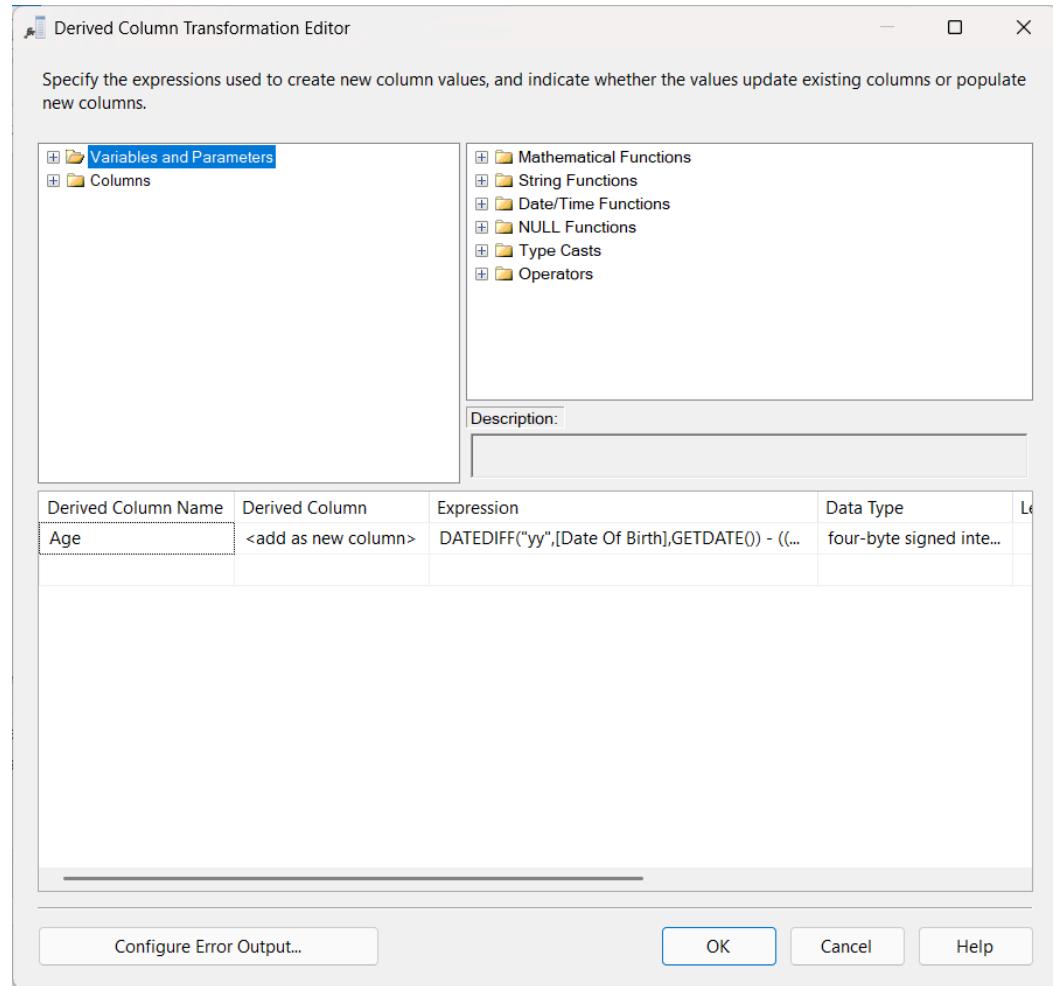
Hình 60. Cài đặt chức năng Sort trong Dim\_CUSTOMERS

- Bước 13: Kéo chức năng **Derived Column** vào vùng **Data Flow** và đổi tên thành **Get\_AGE**. Kéo mũi tên luồng dữ liệu từ **Sort** vào **Get\_AGE**.



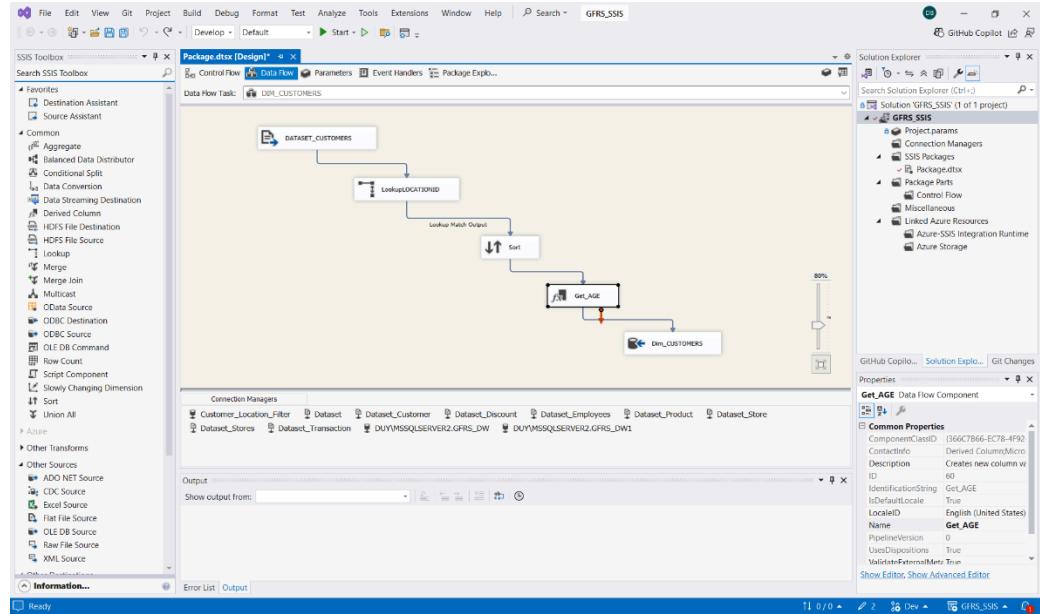
Hình 61. Tạo chức năng Get\_AGE trong Dim\_CUSTOMERS

- **Bước 14:** Nhấn đúp chuột vào **Get\_AGE**, tạo một cột mới có tên là **Age** ở mục **Derived Column Name** và nhập “**DATEDIFF(“yy”, [DateOfBirth], GETDATE() - ((DATEADD(“yy”, DATEDIFF(“yy”, [DateOfBirth], GETDATE()), [DateOfBirth]) > GETDATE() ? 1 : 0)**” để lấy tuổi của khách hàng ở mục **Expression**.



Hình 62. Cài đặt Get\_AGE trong Dim\_CUSTOMERS

- **Bước 15:** Kéo chức năng **ADO NET Destination** vào vùng **Data Flow** và đổi tên thành **Dim\_CUSTOMERS**. Kéo mũi tên luồng dữ liệu từ **Get\_AGE** vào **Dim\_CUSTOMERS**.



Hình 63. Tạo ADO NET Destination trong Dim\_CUSTOMERS

- **Bước 16:** Vào SQL Server 2022, chạy lệnh DDL để tạo Dim\_CUSTOMERS

The screenshot shows the SQL Server Management Studio (SSMS) interface. In the Object Explorer, the database 'GFRS\_DW' is selected. In the center, a query window titled 'SQLQuery1.s...\\maing (73)\*' is open, displaying DDL scripts for creating three dimensions: Dim\_LOCATION, Dim\_CUSTOMER, and Dim\_STORES. The scripts include creating identity columns for primary keys and defining various data types and constraints. The status bar at the bottom indicates the command was executed successfully.

```

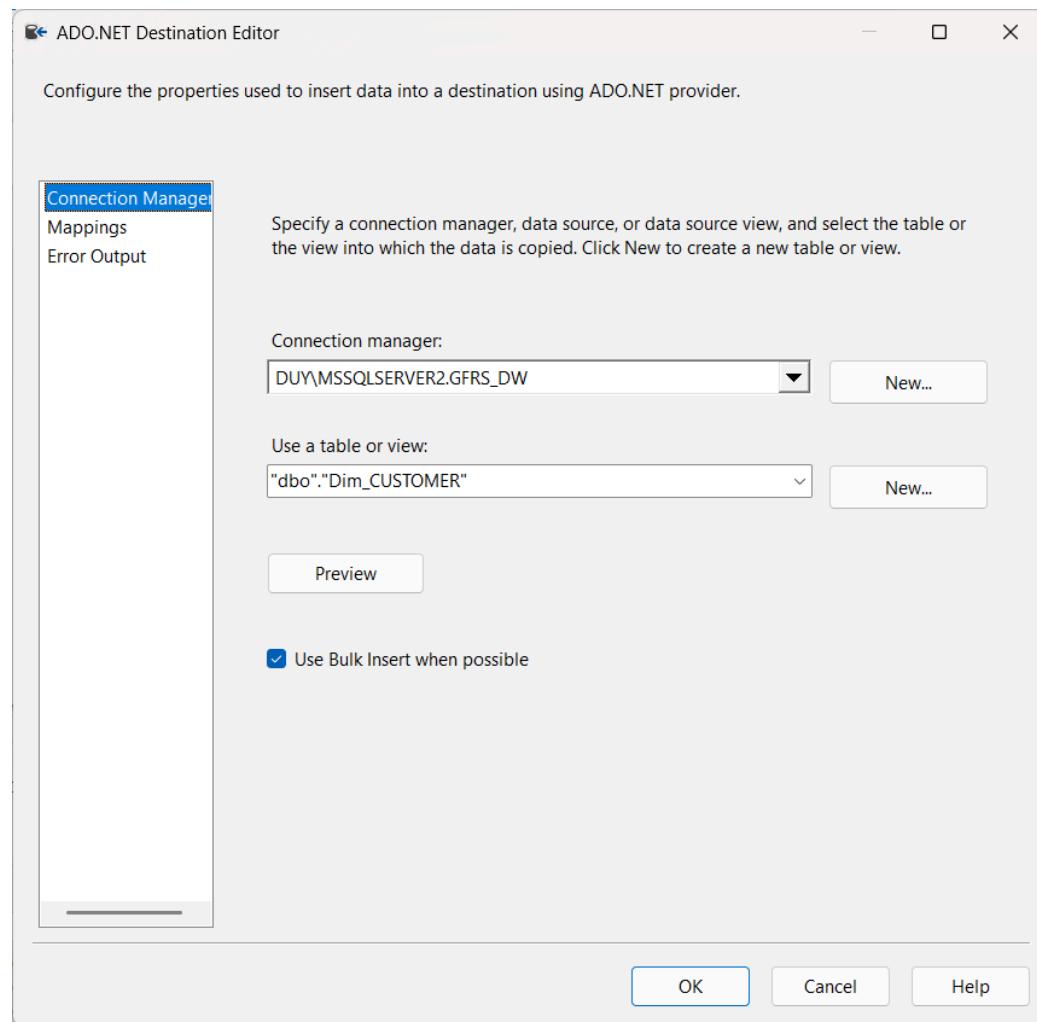
22 DROP TABLE IF EXISTS Dim_TRANSACTIONTYPE;
23 DROP TABLE IF EXISTS Dim_PAYMENTMETHOD;
24
25 CREATE TABLE [Dim_LOCATION] (
26     [Location_ID] INT IDENTITY(1,1) PRIMARY KEY,
27     [Country] VARCHAR(50),
28     [City] VARCHAR(50)
29 )
30 GO
31
32 CREATE TABLE [Dim_CUSTOMER] (
33     [CustomerID] BIGINT IDENTITY(1,1) PRIMARY KEY,
34     [CustomerHash] BIGINT,
35     [Gender] VARCHAR(50),
36     [JobTitle] VARCHAR(100),
37     [Age] INT,
38     [LocationID] INT
39 )
40 GO
41
42 CREATE TABLE [Dim_STORES] (
43     [StoreID] INT IDENTITY(1,1) PRIMARY KEY,
44     [StoreHash] BIGINT,
45     [StoreName] VARCHAR(255),
46     [NumberOfEmployees] INT,
47     [LocationID] INT
48 )
49 GO

```

Hình 64. Chạy lệnh DDL tạo Dim\_CUSTOMERS trong kho dữ liệu

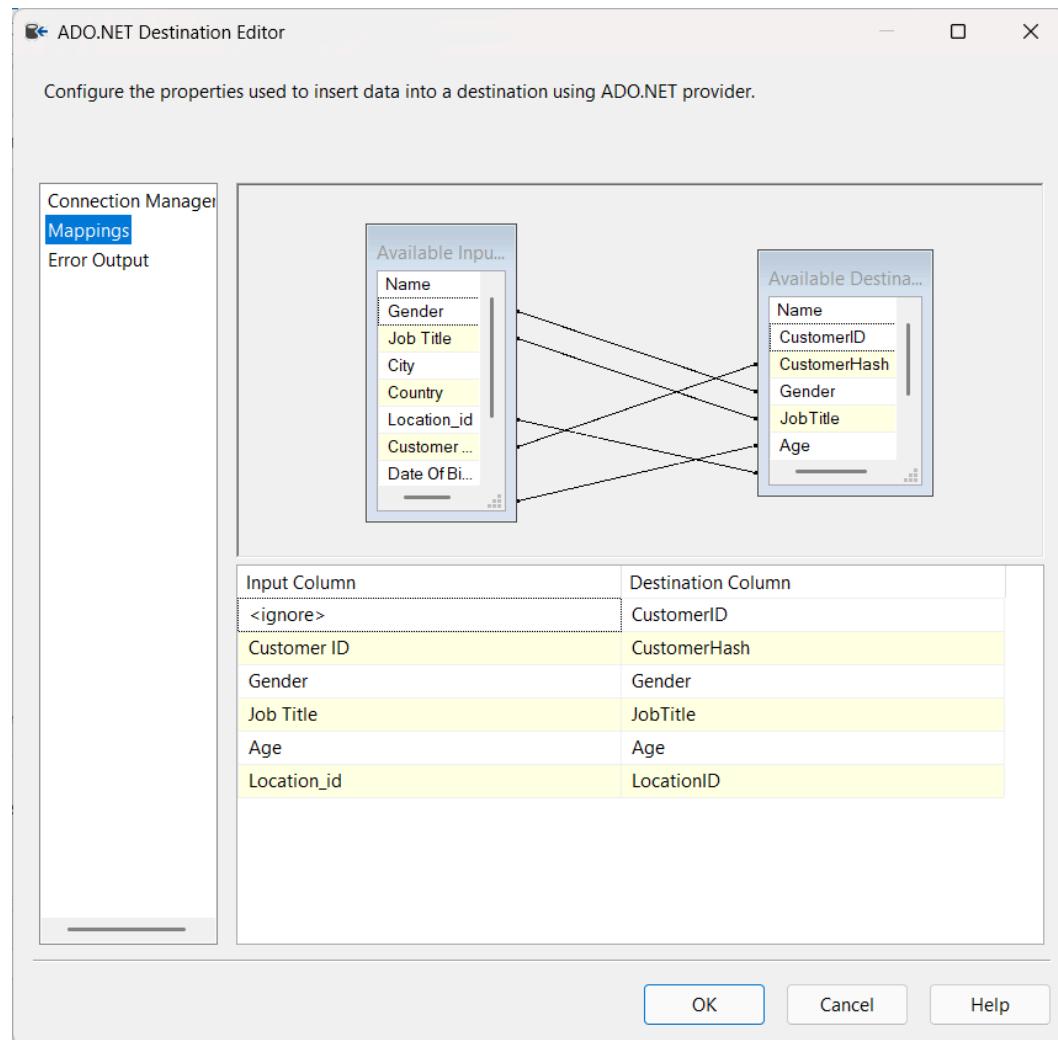
- **Bước 17 :**Nhấn đúp chuột vào Dim\_CUSTOMERS. Trong tab Connection Manager của màn hình ADO.NET Destination Editor, ở Use

**a table or view** tiến hành nhấn chọn “**dbo”.”Dim\_CUSTOMERS**”, cũng chính là bảng vừa được tạo bằng lệnh DDL ở **SQL Server 2022**



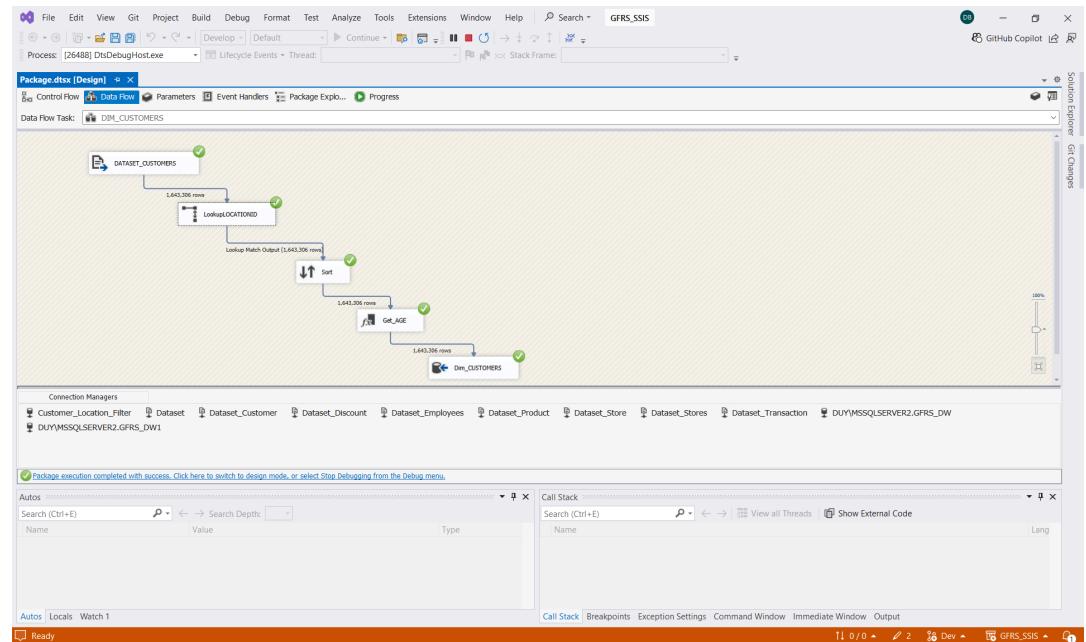
Hình 65. Kết nối với Dim\_CUSTOMERS vừa tạo

- **Bước 18:** Ở tab **Mapping** của màn hình **ADO.NET Destination Editor** tiến hành chọn các thuộc tính thích hợp ở cột bên trái để ánh xạ với các thuộc tính ở cột bên phải.



Hình 66. Cài đặt ánh xạ cho chức năng ADO NET Destination

- **Bước 19:** Nhấn Start để kiểm tra quá trình tạo và đổ dữ liệu vào **Dim\_CUSTOMERS** có thành công hay chưa.



*Hình 67. Đỗ dữ liệu thành công vào Dim\_CUSTOMERS*

Kết quả hiển thị cho thấy quá trình diễn ra thành công và có **1,643,306 rows** được đỗ vào **DIM\_CUSTOMERS**.

- **Bước 20:** Vào **SQL Server 2022** kiểm tra và thấy dữ liệu đã được đỗ vào **Dim\_CUSTOMERS** thành công.

```

22 DROP TABLE IF EXISTS Dim_TRANSACTIONTYPE;
23 DROP TABLE IF EXISTS Dim_PAYMENTMETHOD;
24
25 CREATE TABLE [dim_LOCATION] (
26     [Location_id] INT IDENTITY(1,1) PRIMARY KEY,
27     [Country] VARCHAR(50),
28     [City] VARCHAR(50)
29 )
29 GO
30
31 CREATE TABLE [Dim_CUSTOMER] (
32     [CustomerID] BIGINT IDENTITY(1,1) PRIMARY KEY,
33     [CustomerHash] BIGINT,
34     [Gender] VARCHAR(50),
35     [JobTitle] VARCHAR(100),
36     [Age] INT,
37     [LocationID] INT
38 )
38 GO
39
40 SELECT * FROM Dim_CUSTOMER;
41
42 CREATE TABLE [dim_STORES] (

```

No issues found

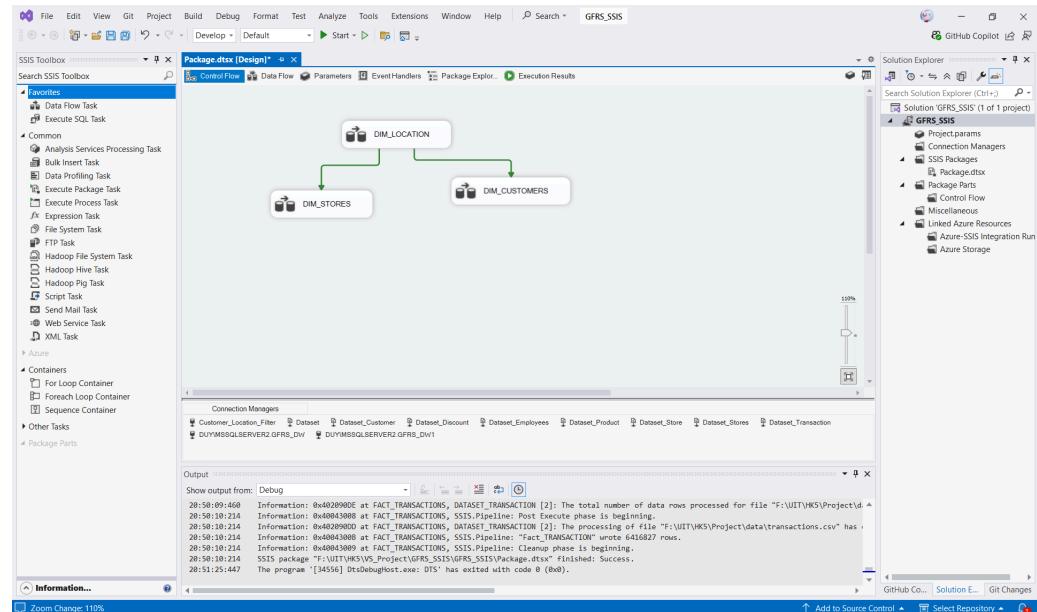
CustomerID	CustomerHash	Gender	JobTitle	Age	LocationID
1	354451	F		9	6
2	354452	F	Financial controller	75	6
3	354453	F		21	6
4	354454	M		18	6
5	354455	M		19	6
6	354456	M		23	6
7	354457	F	Location manager	37	6
8	354458	F		20	6
9	354459	M	English as a forei...	41	6
10	354460	F		20	6
11	354461	F		19	6
12	354462	F		21	6

Executing query... DUY\SQLSERVER2 (16.0 RTM) DUY\maing (73) GFRS\_DW 00:00:03 | 0 rows

Hình 68. Dữ liệu đã được đổ vào Dim\_CUSTOMERS

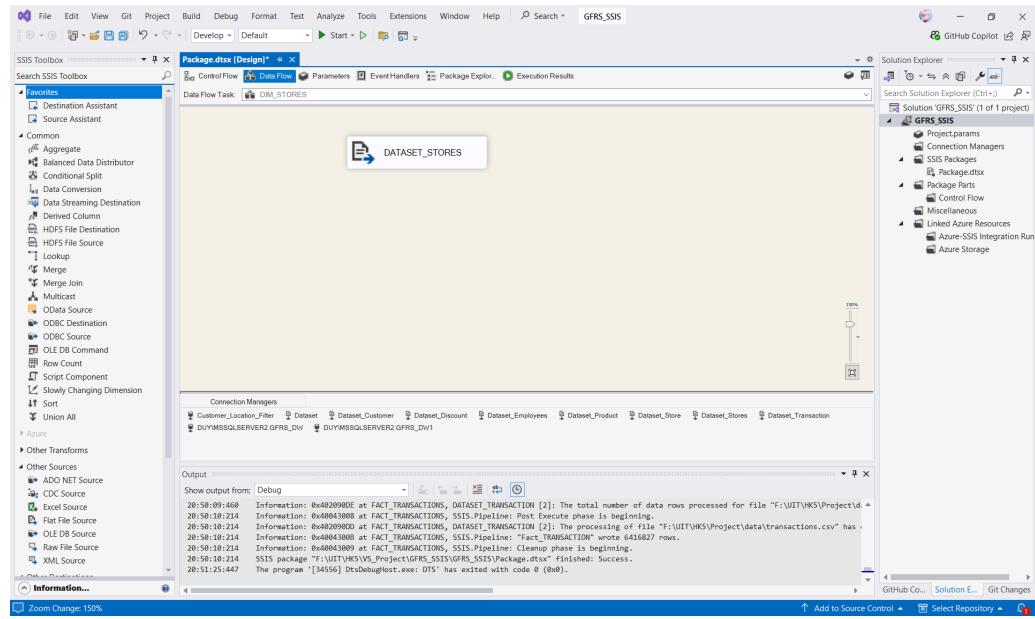
### II.3.2. Dim\_STORES

- **Bước 1:** Kéo chức năng Data Flow Task trong vùng SSIS Toolbox vào vùng Control Flow và đổi tên thành Dim\_STORES.



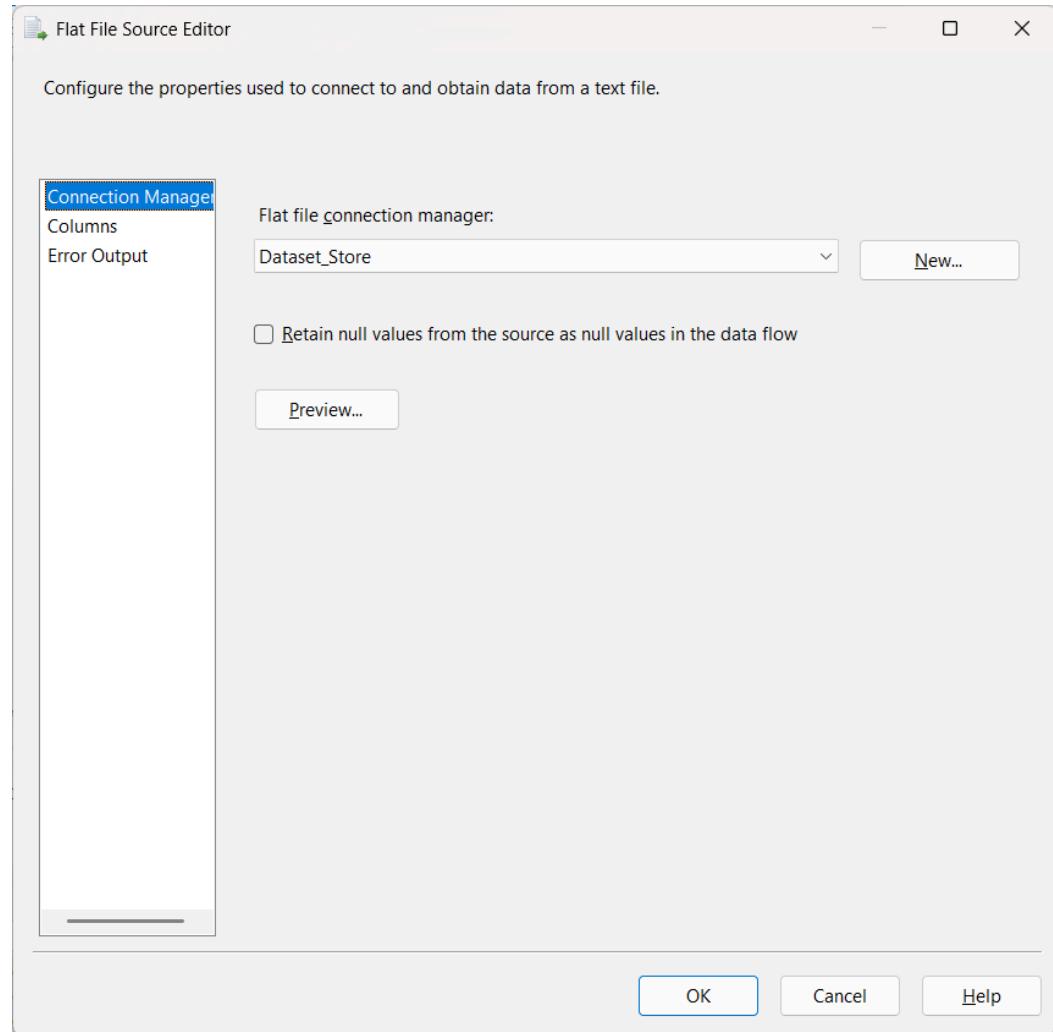
Hình 69. Tạo Data Flow Task Dim\_STORES

- **Bước 2:** Nhấn đúp chuột vào **Dim\_STORES**. Sau đó kéo chức năng **Flat File Source** vào vùng làm việc và đổi tên thành **DATASET\_STORES**



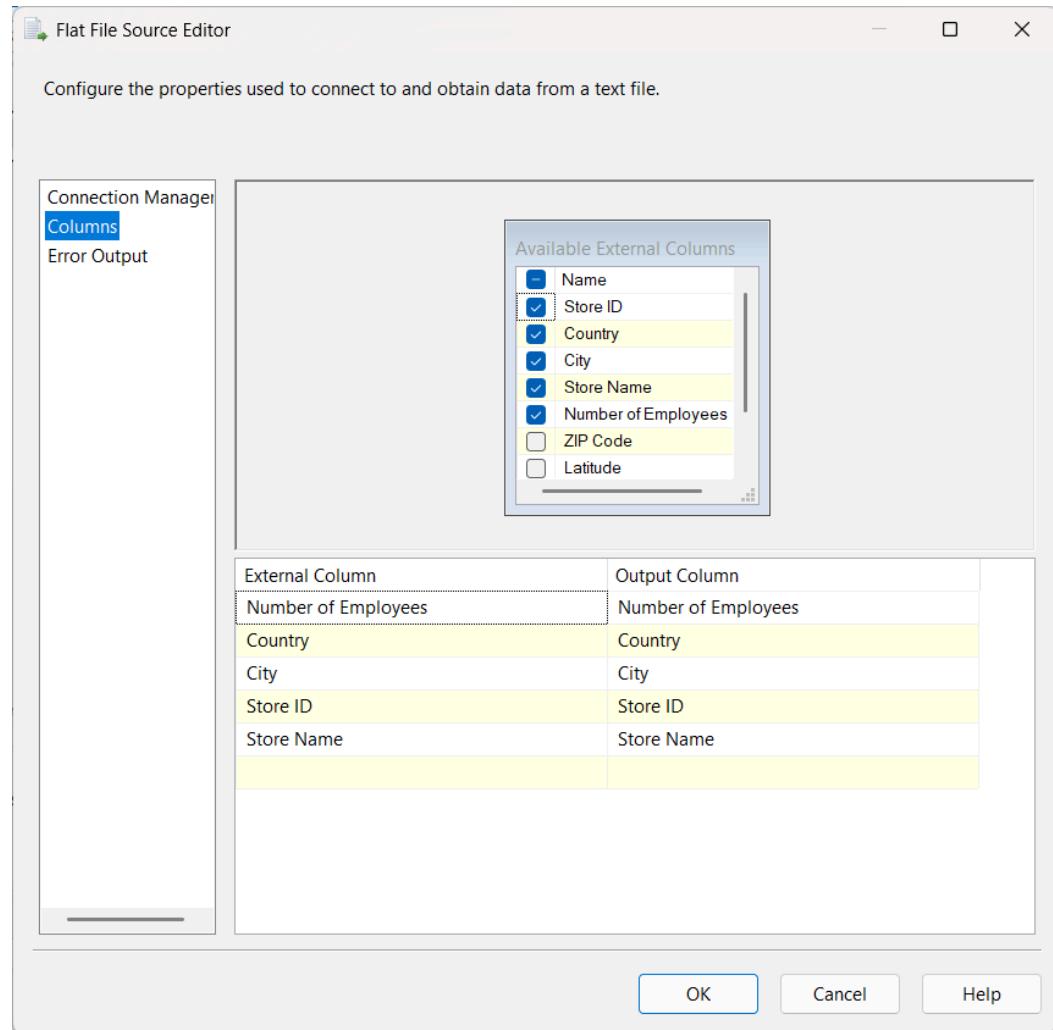
Hình 70. Tạo DATASET\_STORES trong Dim\_STORES

- **Bước 3:** Nhấn đúp chuột vào **DATASET\_STORES**. Màn hình **Flat File Source Editor** hiện ra. Do quá trình tạo **Dim\_LOCATION**, ở các bước trước đã load thành công file csv và đổi tên thành **Dataset\_Store** nên ta có thể sử dụng lại **Dataset\_Store** để load dữ liệu mà không cần thiết lập lại giúp tiết kiệm thời gian và giảm nguy cơ gặp lỗi.



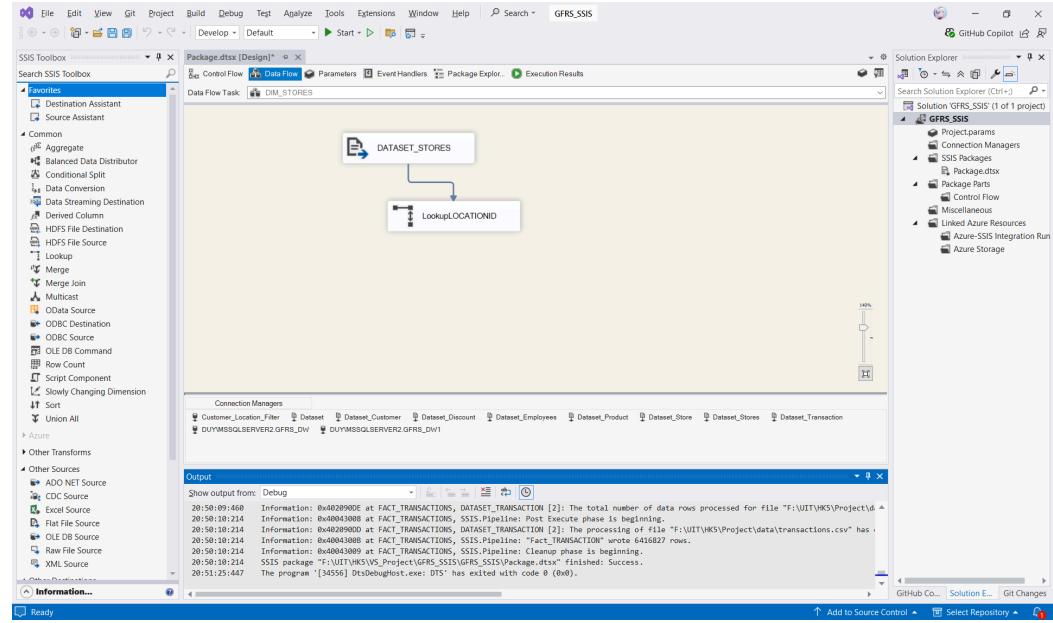
Hình 71. Cài đặt nguồn dữ liệu cho DATASET\_STORES

- **Bước 4:** Chuyển sang tab **Columns**, ta có thể lựa chọn các thuộc tính cần thiết cho quá trình nạp dữ liệu cho **Dim\_STORES**. Sau đó ta có thể bấm **OK** để đóng cửa sổ **Flat File Source Editor**



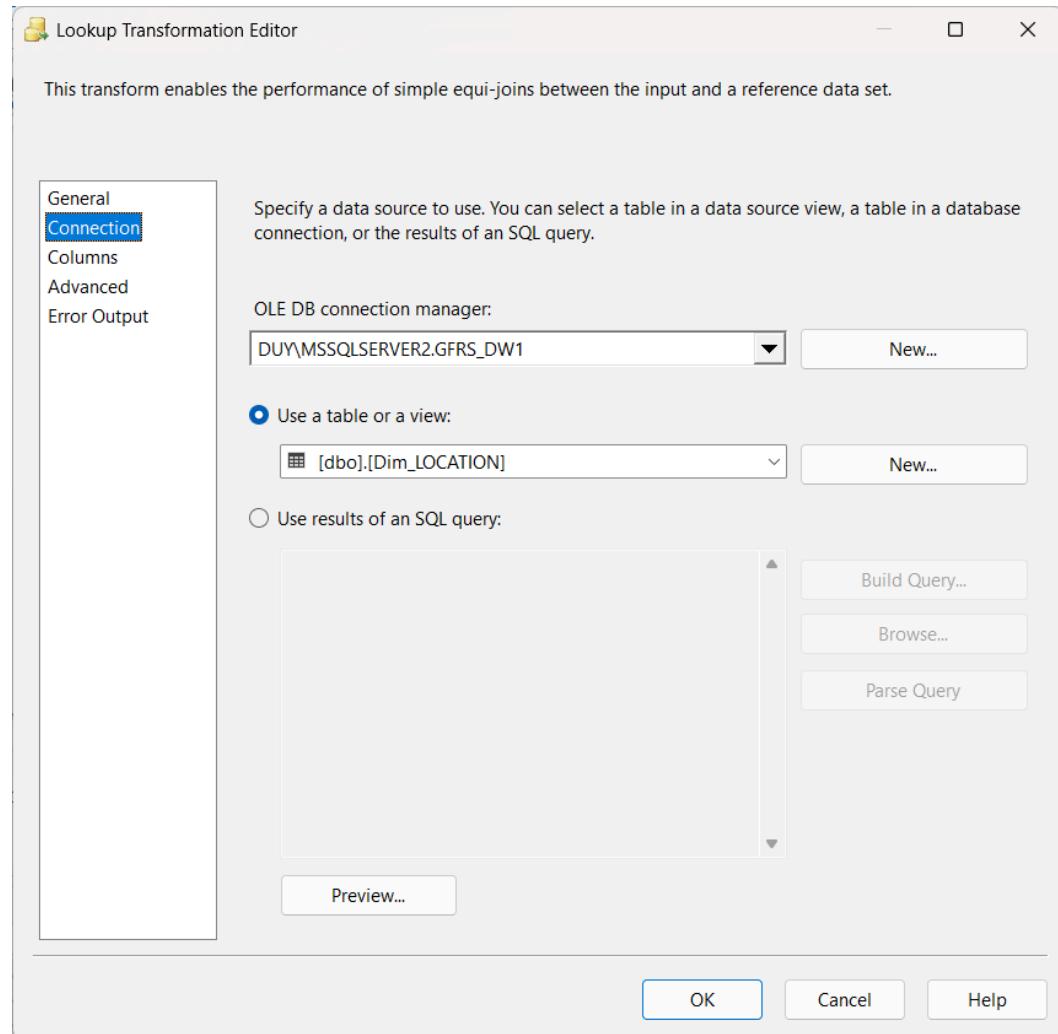
Hình 72. Lựa chọn thuộc tính cần thiết cho quá trình tạo Dim\_STORES

- **Bước 5:** Kéo chức năng **Lookup** vào vùng **Data Flow** và đổi tên thành **LookupLOCATIONID**. Kéo mũi tên luồng dữ liệu từ **DATASET\_STORES** vào **LookupLOCATIONID**.



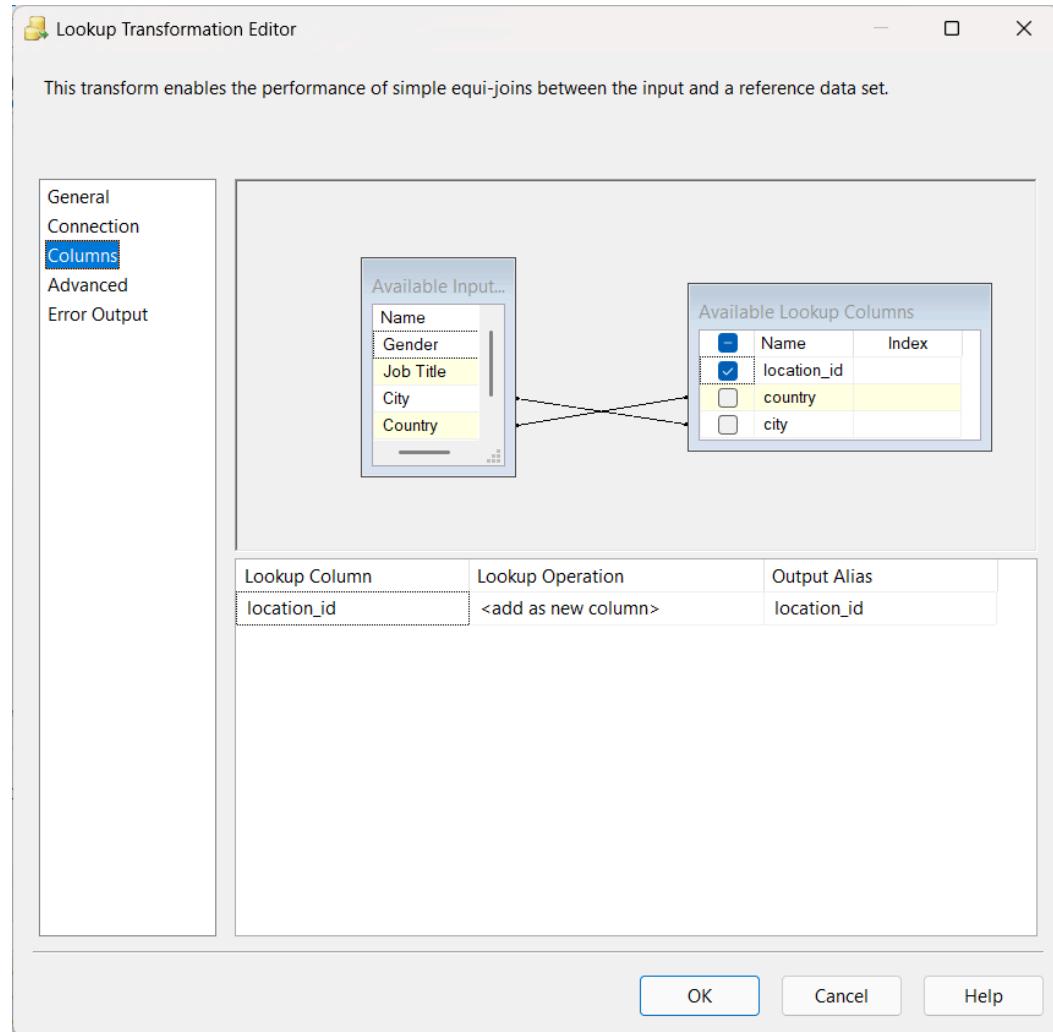
Hình 73. Tạo chức năng LookupLOCATIONID trong Dim\_STORES

- **Bước 6:** Nhấn đúp chuột vào **LookupLOCATIONID**, màn hình **Lookup Transformation Editor** xuất hiện. Do chúng ta đã thiết lập **OLE DB connection manager** ở bước trước, nên ta có thể sử dụng lại. Ở phần **Use a table or a view**, tiến hành chọn **[dbo].[Dim\_LOCATION]**.



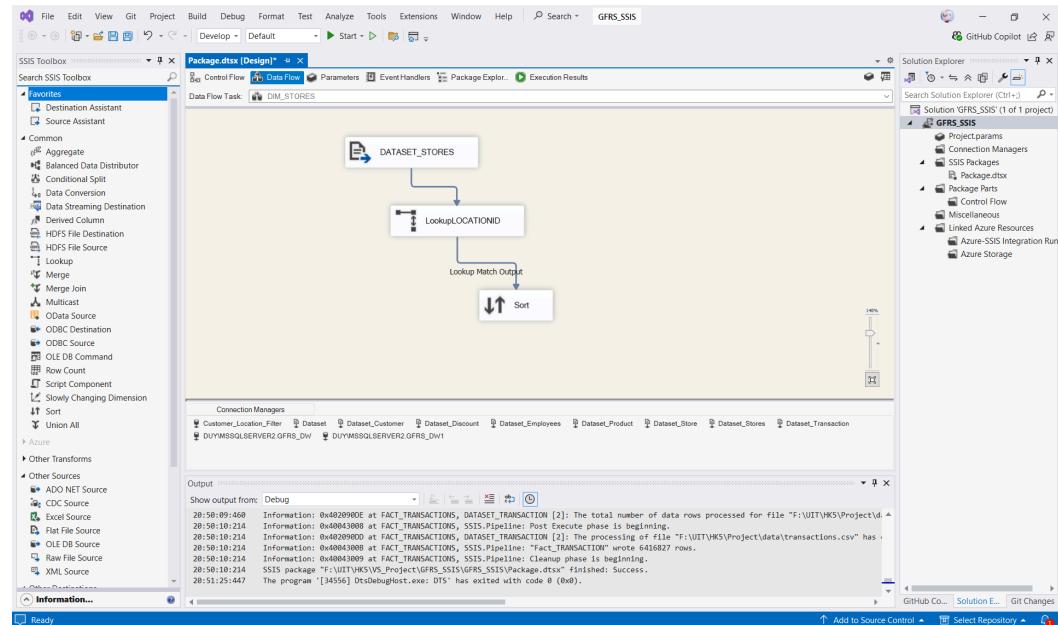
Hình 74. Cài đặt nguồn dữ liệu cho LookupLOCATIONID

- **Bước 7:** Trong tab **Columns**, ở bảng bên phải click chọn **location\_id**. Ta kéo **City**, **Country** từ bảng **Available Input Columns** qua tương ứng **city**, **country** bảng **Available Lookup Columns** (quá trình này giúp hệ thống xác định vùng dữ liệu cần tìm kiếm). Sau đó bấm **OK** để đóng cửa sổ **Lookup Transformation Editor**



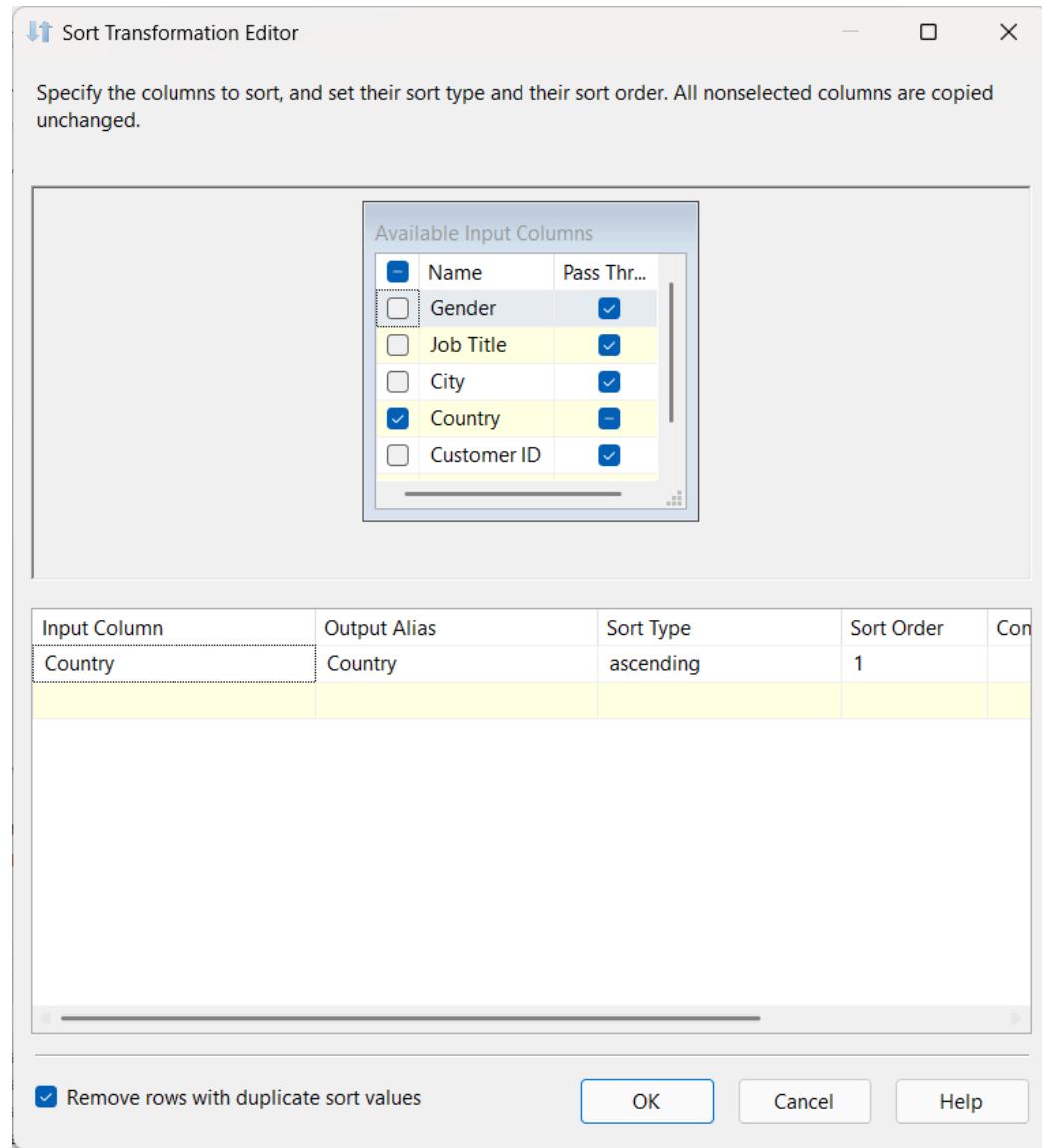
Hình 75. Cài đặt LookupLOCATIONID trong Dim\_STORES

- **Bước 8:** Kéo chức năng **Sort** vào vùng **Data Flow** và kéo mũi tên luồng dữ liệu từ **LookupLOCATIONID** vào **Sort**. Mục đích của chức năng **Sort** là để sắp xếp dữ liệu theo thứ tự mong muốn của người dùng, giúp ích trong việc phân tích và đưa ra quyết định, đồng thời loại bỏ các dòng dữ liệu trùng lặp trước khi đổ dữ liệu vào kho. Một hộp thoại sẽ xuất hiện, chọn **Lookup Match Output** và nhấn **OK**.



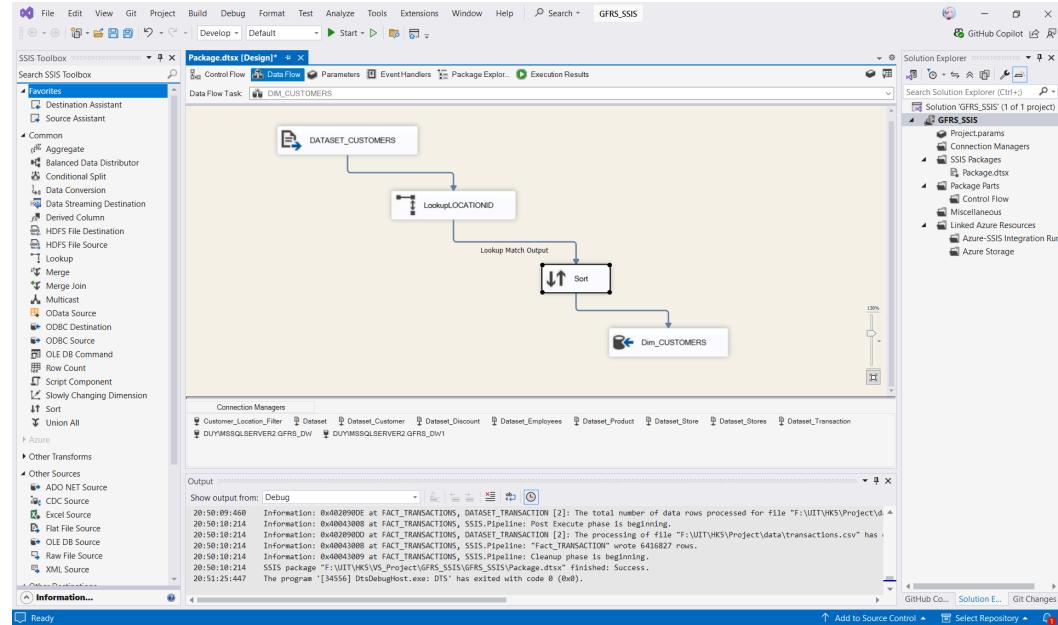
Hình 76. Tạo chức năng Sort trong Dim\_STORES

- **Bước 9:** Nhấn đúp chuột vào **Sort** và màn hình **Sort Transformation Editor** xuất hiện. Tiến hành click vào các thuộc tính muốn sắp xếp và click vào checkbox **Remove rows with duplicate sort values**. Nhấn **OK** và thoát màn hình **Sort Transformation Editor**.



Hình 77. Cài đặt chức năng Sort trong Dim\_STORES

- **Bước 10:** Kéo chức năng **ADO NET Destination** vào vùng **Data Flow** và đổi tên thành **Dim\_STORES**. Kéo mũi tên luồng dữ liệu từ **Sort** vào **Dim\_STORES**.



Hình 78. Tạo ADO NET Destination Dim\_STORES trong Dim\_STORES

- **Bước 11:** Vào SQL Server 2022, chạy lệnh DDL để tạo Dim\_STORES

The screenshot shows the Microsoft SQL Server Management Studio (SSMS) interface. The Object Explorer on the left shows a connection to 'D:\MSSQLSERVER2 (SQL Server 16)' with several databases listed. The 'Solution1' solution is open in the center, displaying a query window titled 'SQLQuery1.s...\\maing (73)\*'. The query contains two CREATE TABLE statements for 'Dim\_STORES' and 'Dim\_EMPLOYEES', both with primary keys and various columns like StoreID, StoreHash, and StoreName. Below the queries is a results grid titled 'Customer' with columns CustomerID, CustomerHash, Gender, JobTitle, Age, and LocationID. The results show 12 rows of data. At the bottom, a status bar indicates the session is 'D:\MSSQLSERVER2 (16.0 RTM)' and the query was executed successfully at '00:00:05'.

```
37
38
39
40
41    [LocationID] INT
42
43    GO
44
45    CREATE TABLE [Dim_STORES] (
46        [StoreID] INT IDENTITY(1, 1) PRIMARY KEY,
47        [StoreHash] BIGINT,
48        [StoreName] VARCHAR(255),
49        [NumberOfEmployees] INT,
50        [LocationID] INT
51
52    GO
53
54    CREATE TABLE [Dim_EMPLOYEES] (
55        [EmployeeID] INT IDENTITY(1, 1) PRIMARY KEY,
56        [EmployeeHash] BIGINT,
57        [StoreID] INT,
58        [Position] VARCHAR(255)
59    )
60    GO
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
```

CustomerID	CustomerHash	Gender	JobTitle	Age	LocationID
1	354451	F		19	6
2	354452	F	Financial controller	75	6
3	354453	F		21	6
4	354454	M		18	6
5	354455	M		19	6
6	354456	M		23	6
7	354457	F	Location manager	20	6
8	354458	F		20	6
9	354459	M	English as a fore...	41	6
10	354460	F		20	6
11	354461	F		19	6
12	354462	F		21	6

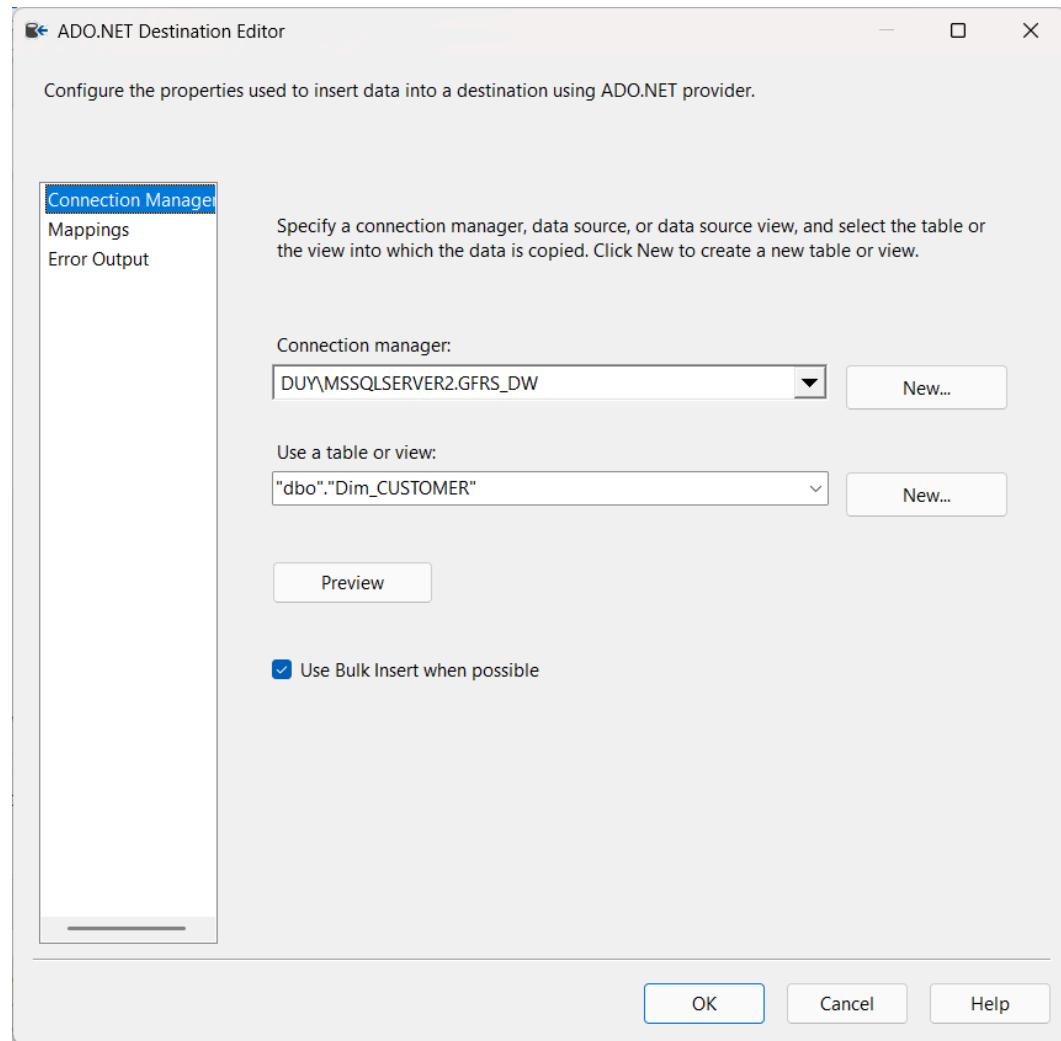
Query executed successfully.

D:\MSSQLSERVER2 (16.0 RTM) D:\maing (73) GFRS\_DW 00:00:05 1,643,306 rows

Hình 79. Chạy lệnh DDL tạo Dim STORES

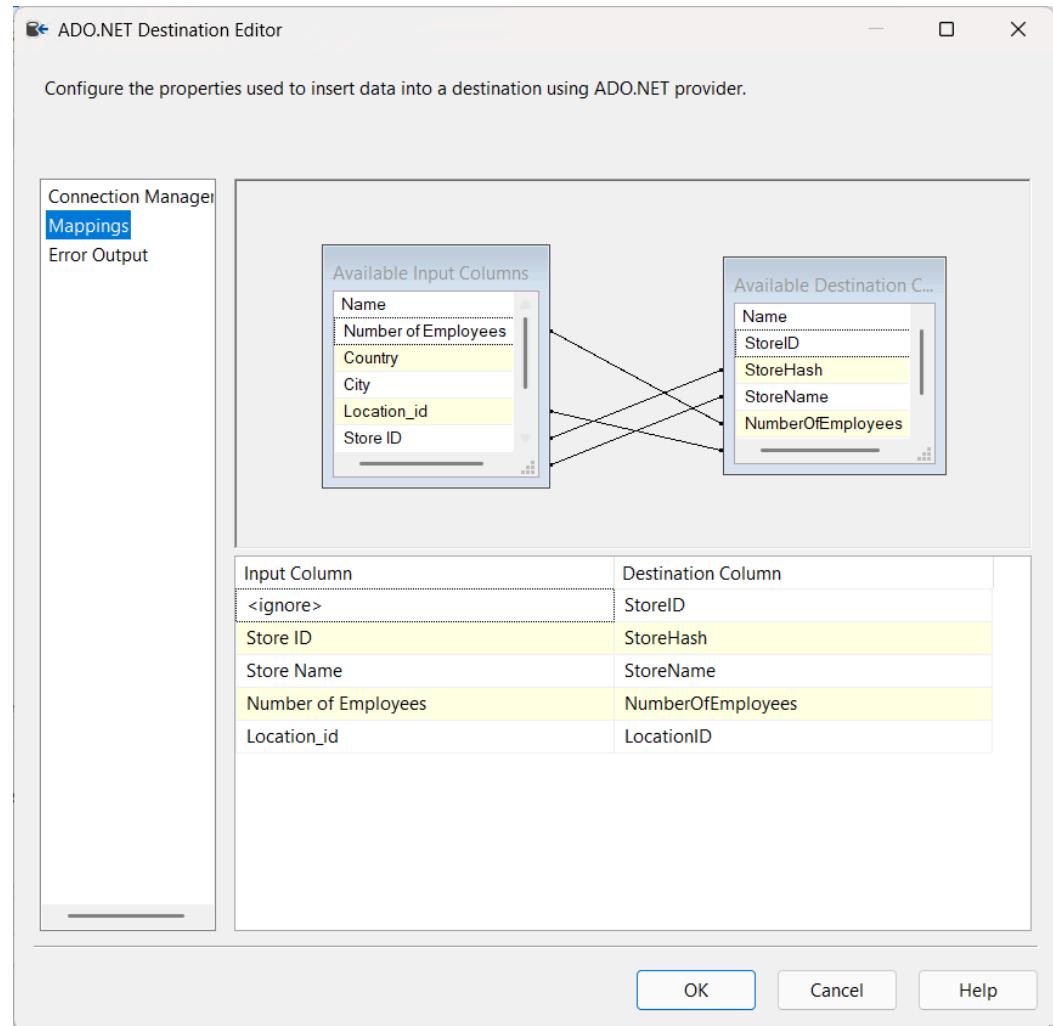
- **Bước 12** :Nhấn đúp chuột vào Dim\_STORES. Trong tab **Connection Manager** của màn hình **ADO.NET Destination Editor**, ở **Use a table or view**:

view tiến hành nhấn chọn “**dbo”.”Dim\_STORES**”, cũng chính là bảng vừa được tạo bằng lệnh **DDL** ở **SQL Server 2022**



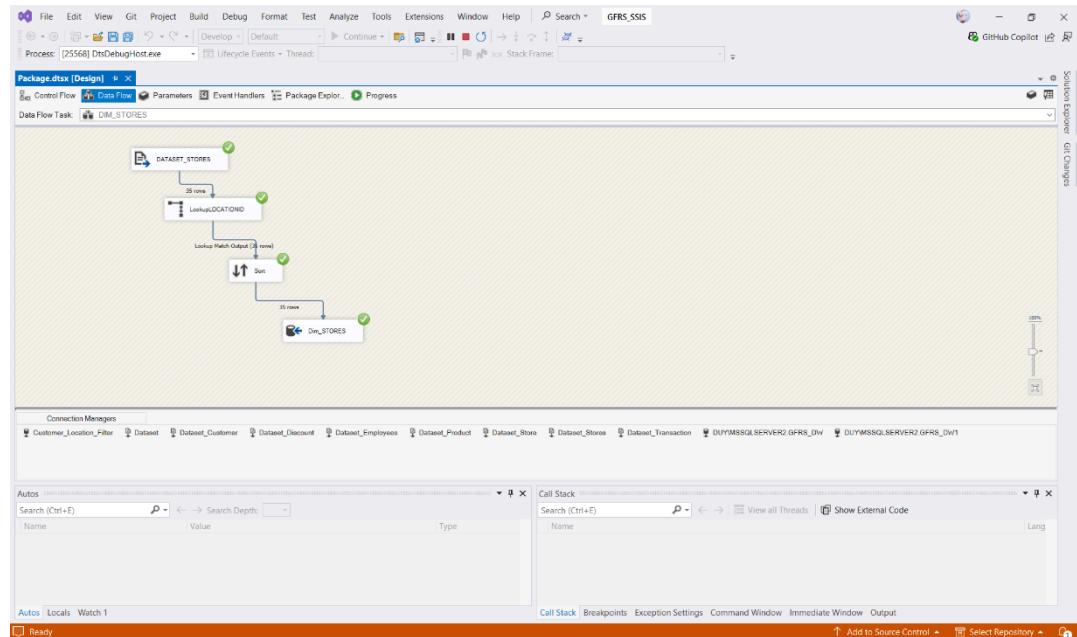
Hình 80. Chọn đích đến cho Dim\_STORES

- **Bước 13:** Ở tab **Mapping** của màn hình **ADO.NET Destination Editor** tiến hành chọn các thuộc tính thích hợp ở cột bên trái để ánh xạ với các thuộc tính ở cột bên phải.



Hình 81. Cài đặt ADO NET Destination Dim\_STORES

- **Bước 14:** Nhấn Start để kiểm tra quá trình tạo và đổ dữ liệu vào Dim\_STORES có thành công hay chưa.



Hình 82. Đỗ dữ liệu vào Dim\_STORES

Kết quả hiển thị cho thấy quá trình diễn ra thành công và có **35 rows** được đỗ vào **DIM\_STORES**.

- **Bước 15:** Vào SQL Server 2022 kiểm tra và thấy dữ liệu đã được đỗ vào **Dim\_STORES** thành công.

```

File Edit View Query Git Project Tools Extensions Window Help Search Solution1
Object Explorer Connect Databases System Databases OLAP OLAP_CK_HK2_2023_2024 OLAP_BT1_DW OLAP_CK2_HK2_2023_2024 OLAP_CK_HK2_2023_2024 Security Server Objects Replication Always On High Availability Management Integration Services Catalogs SQL Server Agent XEvent Profiler
SQLQuery1...maing (73)* DB_New...sq...maing (108) Untitled (1)...UYmaing (75)
[LocationID] INT
)
GO
CREATE TABLE [Dim_STORES] (
    [StoreID] INT IDENTITY(1,1) PRIMARY KEY,
    [StoreHash] BIGINT,
    [StoreName] VARCHAR(255),
    [NumberOfEmployees] INT,
    [LocationID] INT
)
GO
Select * FROM Dim_STORES;
CREATE TABLE [Dim_EMPLOYEES] (
    [EmployeeID] INT IDENTITY(1,1) PRIMARY KEY,
    [EmployeeHash] BIGINT,
    [StoreID] INT,
    [Position] VARCHAR(255)
)

```

Results

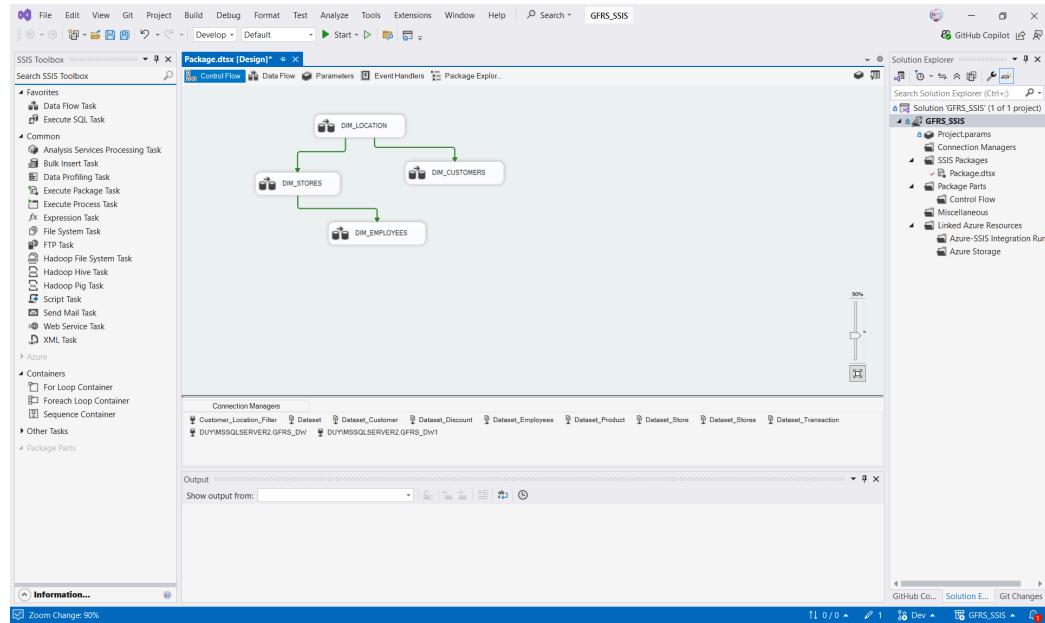
StoreID	StoreHash	StoreName	NumberOfEmployees	LocationID
1	1	Store New York	10	511
2	2	Store Los Angeles	8	435
3	3	Store Chicago	9	210
4	4	Store Houston	10	364
5	5	Store Phoenix	9	563
6	6	Store a_Sea	8	6
7	7	Store 12-12	10	98
8	8	Store a_12	10	38
9	9	Store a_12@	9	60
10	10	Store a_12#	10	269
11	11	Store Berlin	9	139
12	12	Store Hamburg	10	350

Query executed successfully.

Hình 83. Dữ liệu được đỗ thành công vào Dim\_STORES

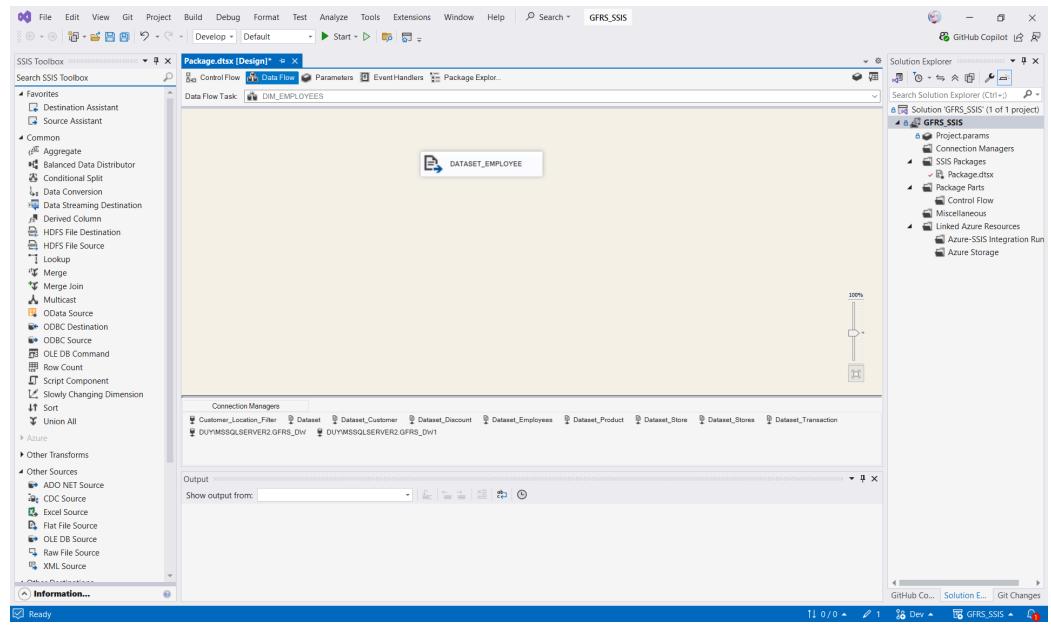
### II.3.3. Dim\_EMPLOYEES

- **Bước 1:** Kéo chức năng Data Flow Task trong vùng SSIS Toolbox vào vùng Control Flow và đổi tên thành **Dim\_EMPLOYEES**.



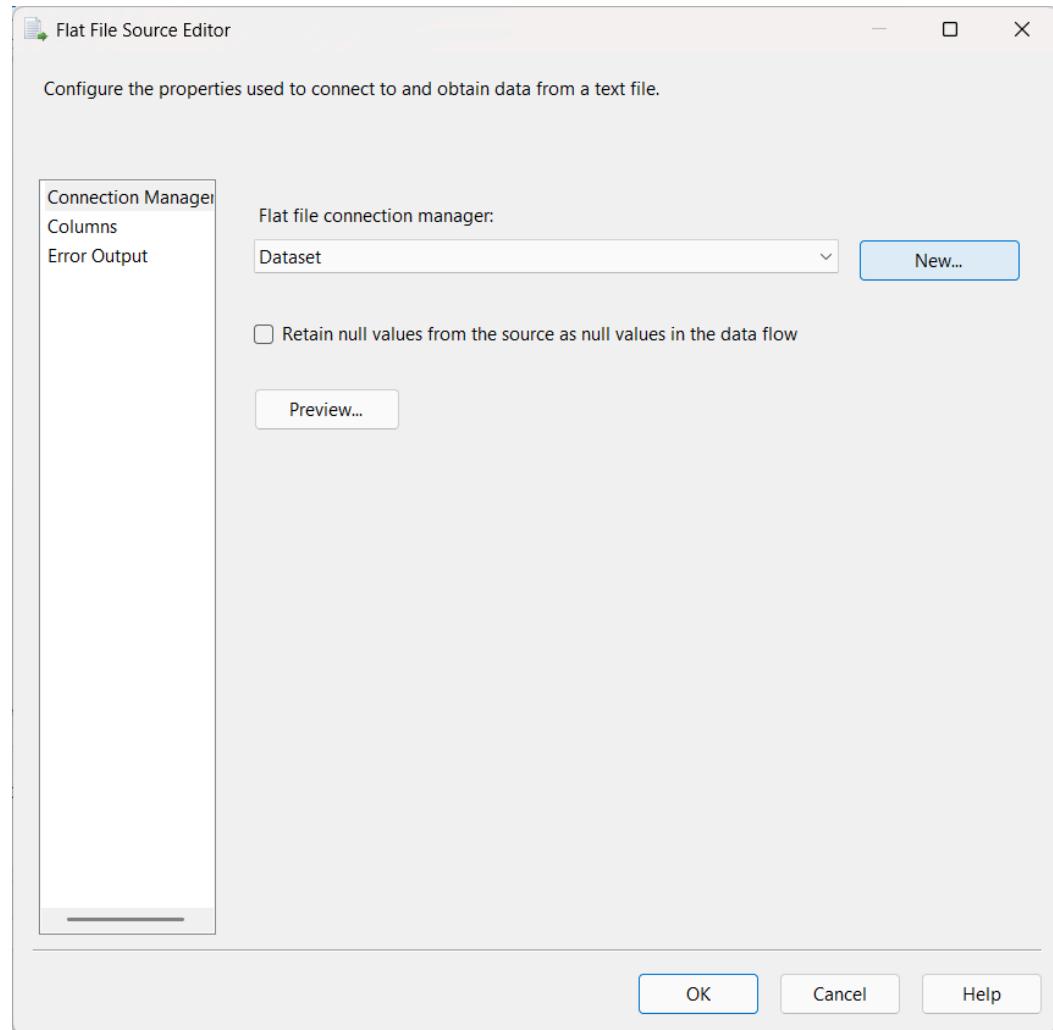
Hình 84. Tạo chức năng Dim\_EMPLOYEES

- **Bước 2:** Nhấn đúp chuột vào **Dim\_EMPLOYEES**. Sau đó kéo chức năng Flat File Source vào vùng làm việc và đổi tên thành **DATASET\_EMPLOYEE**



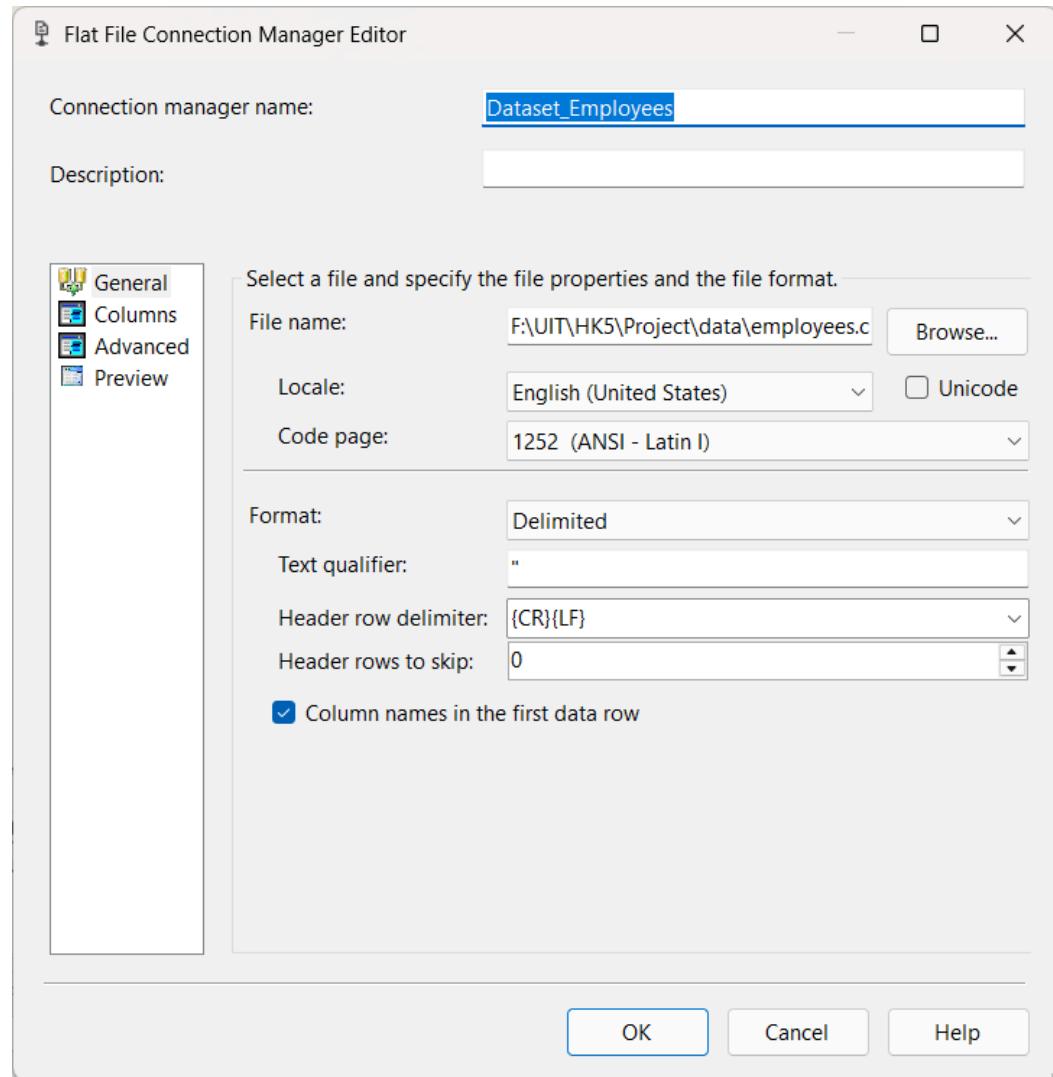
*Hình 85. Tạo DATASET\_EMPLOYEE trong Dim\_EMPLOYEES*

- **Bước 3:** Nhấn đúp chuột vào **DATASET\_EMPLOYEE**. Màn hình **Flat File Source Editor** hiện ra. Ở tab **Connection Manager** nhấn **New**



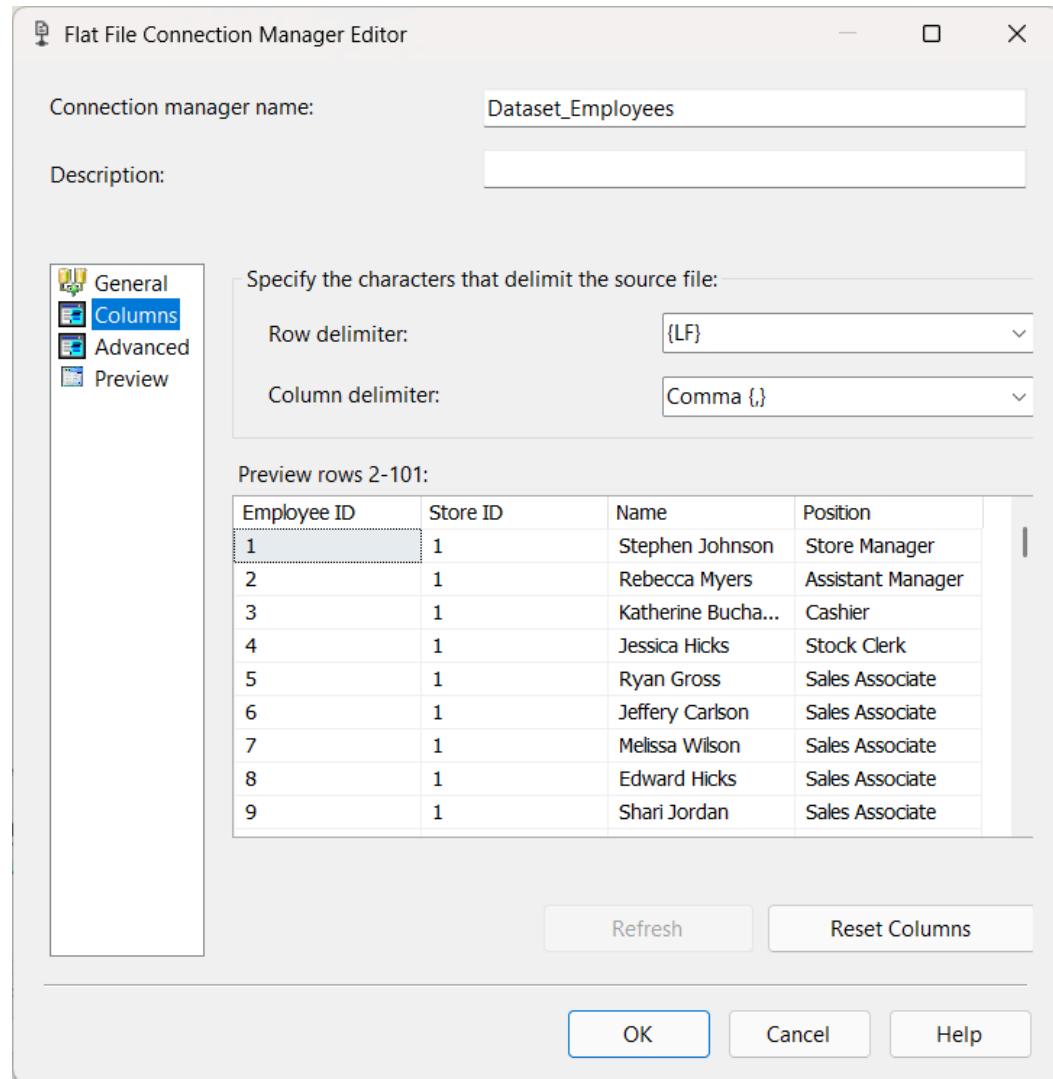
Hình 86. Tạo nguồn dữ liệu mới cho DATASET\_EMPLOYEE

- **Bước 4:** Màn hình **Flat File Connection Manager Editor** hiện ra. Ở tab **General**, nhấn **Browse** để tiến hành load file csv **employees.csv** trong bộ dữ liệu **Global Fashion Retail Sales**. Ở mục **Connection manager name**, đặt tên là **Dataset\_Employees**. Ở mục **Text qualifier**, nhập ký tự dấu nháy kép “



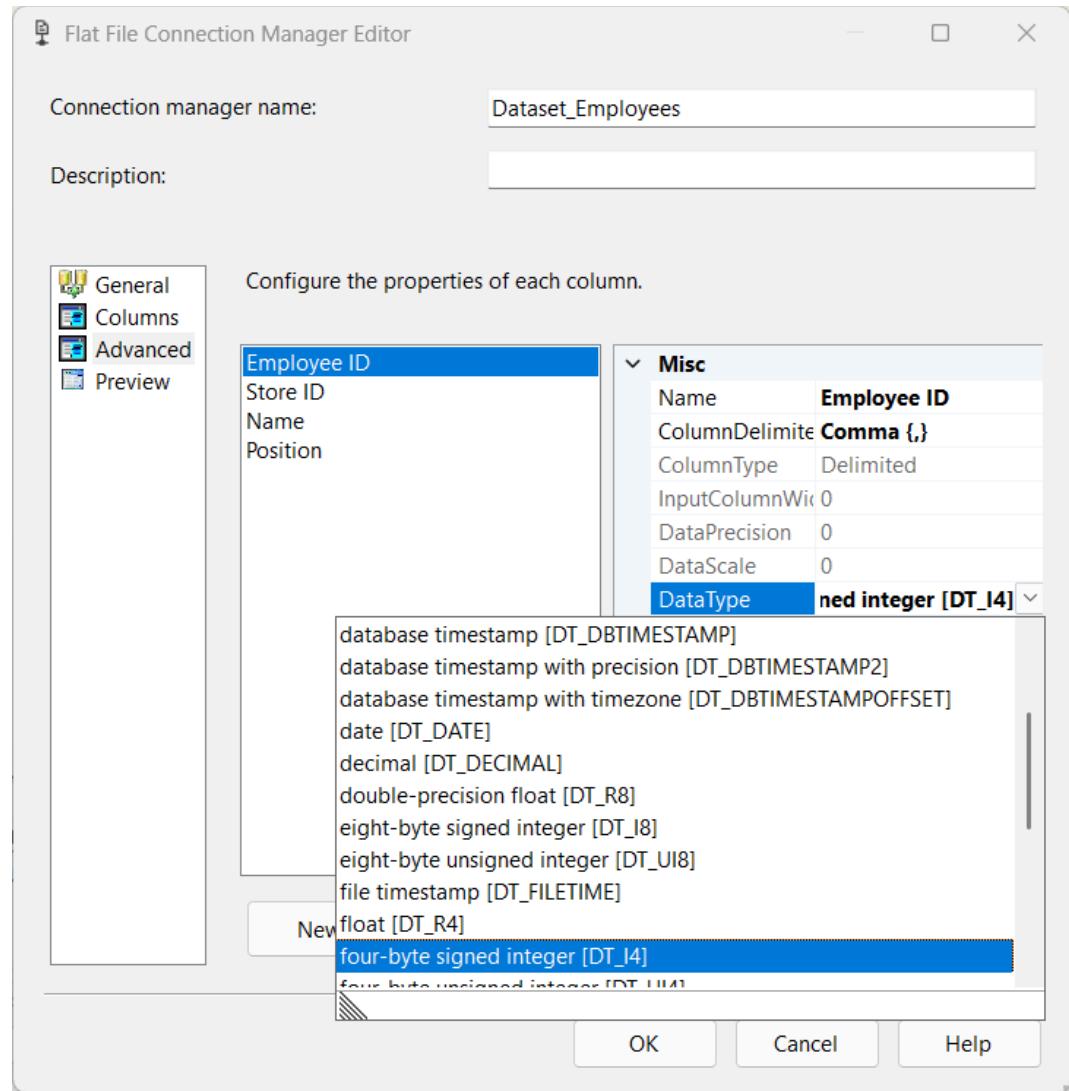
Hình 87. Tạo nguồn dữ liệu Dataset\_Employees mới

- **Bước 5:** Ở tab **Columns**, ta có thể kiểm tra và thấy rằng file csv đã được load thành công



Hình 88. Kiểm tra dữ liệu Dataset\_Employee trước khi nạp

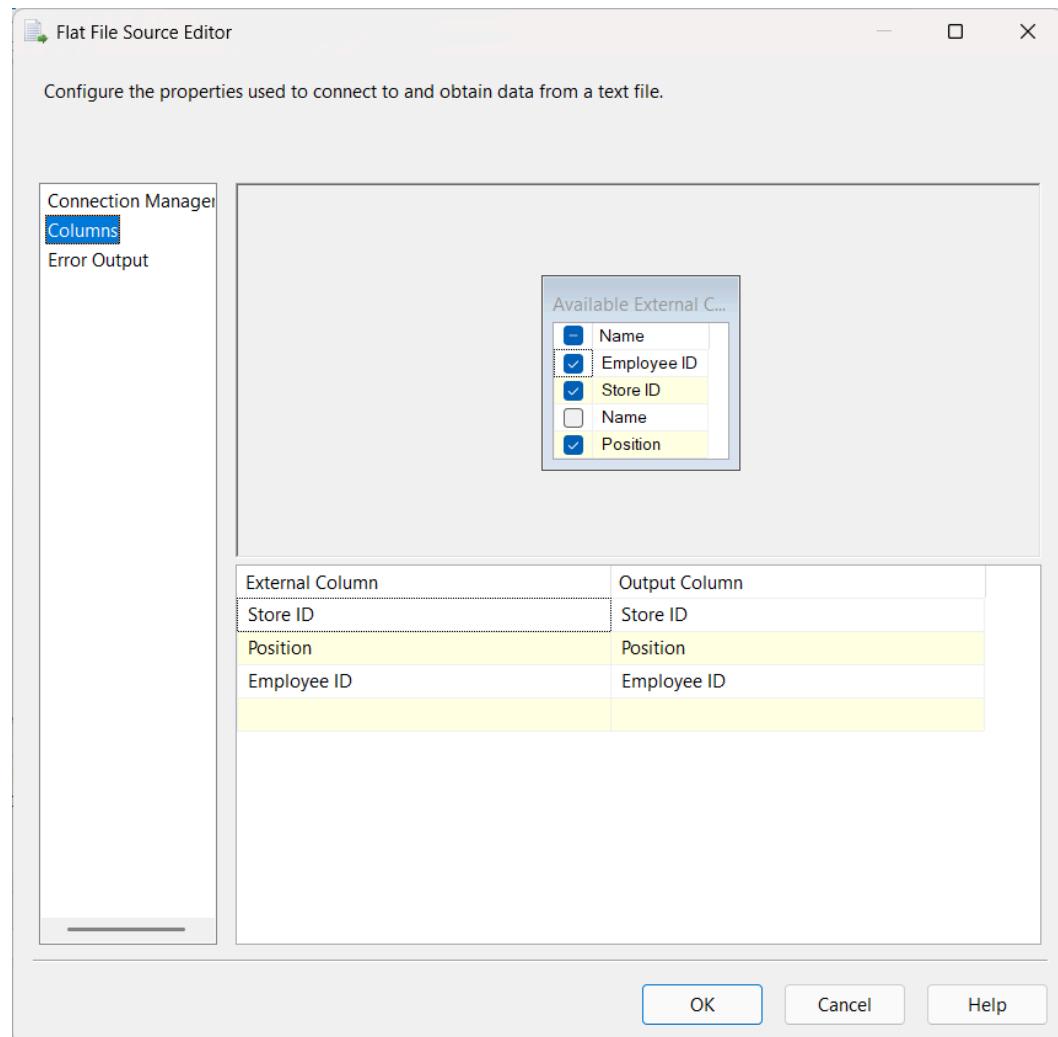
- **Bước 6:** Ở tab **Advanced**, tiến hành thay đổi kiểu dữ liệu của các thuộc tính cần sử dụng trong quá trình **SSIS** của file csv tương ứng với kiểu dữ liệu khai báo của thuộc tính đó trong **SQL Server**. Sau khi thay đổi kiểu dữ liệu của các thuộc tính cần sử dụng trong quá trình **SSIS** thì nhấn **OK** và sẽ thoát khỏi màn hình **Flat File Connection Manager Editor**.



Hình 89. Chọn kiểu dữ liệu phù hợp cho các thuộc tính cho

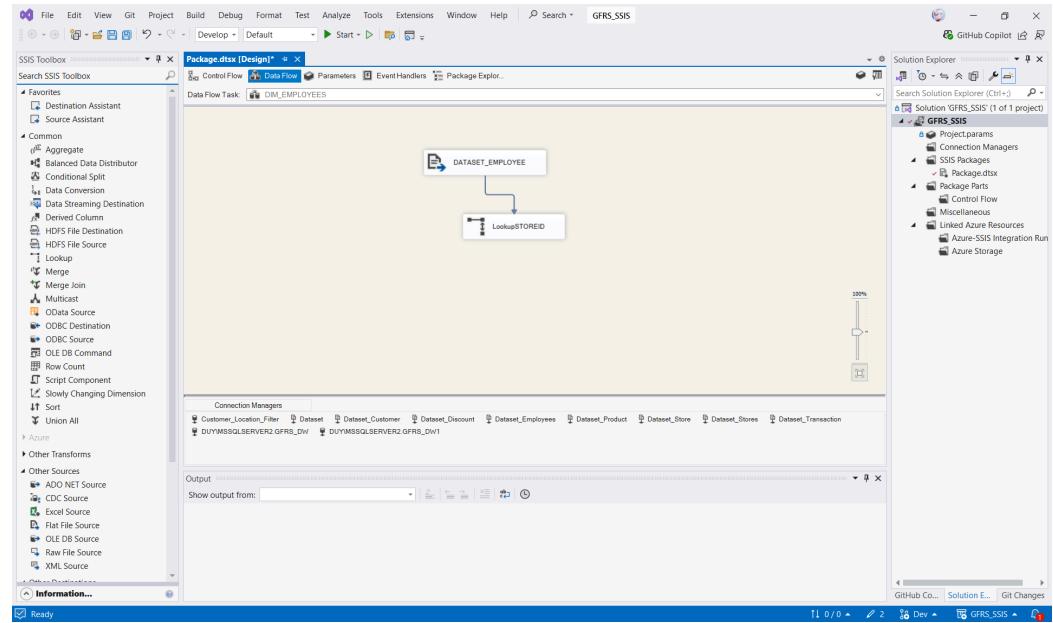
*Dataset\_Employee*

- **Bước 7:** Ở màn hình **Flat File Source Editor**, qua tab **Columns** và nhấn chọn các thuộc tính cần sử dụng trong quá trình tạo **Dim\_EMPLOYEES**



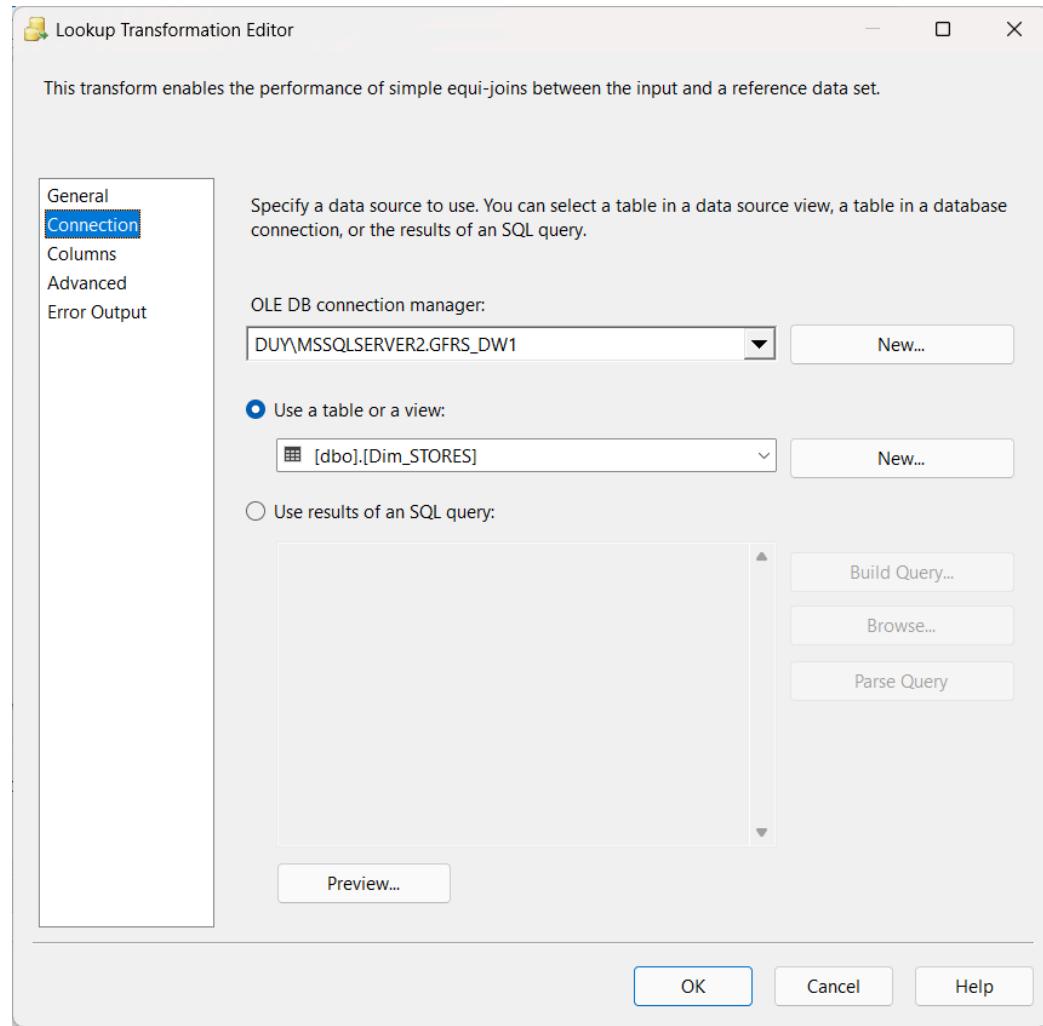
Hình 90. Chọn các thuộc tính cần thiết cho quá trình tạo  
Dim\_EMPLOYEES

- **Bước 8:** Nhấn **OK** để thoát màn hình **Flat File Source Editor**
- **Bước 9:** Kéo chức năng **Lookup** vào vùng **Data Flow** và đổi tên thành **LookupSTOREID**. Kéo mũi tên luồng dữ liệu từ **DATASET\_EMPLOYEE** vào **LookupSTOREID**.



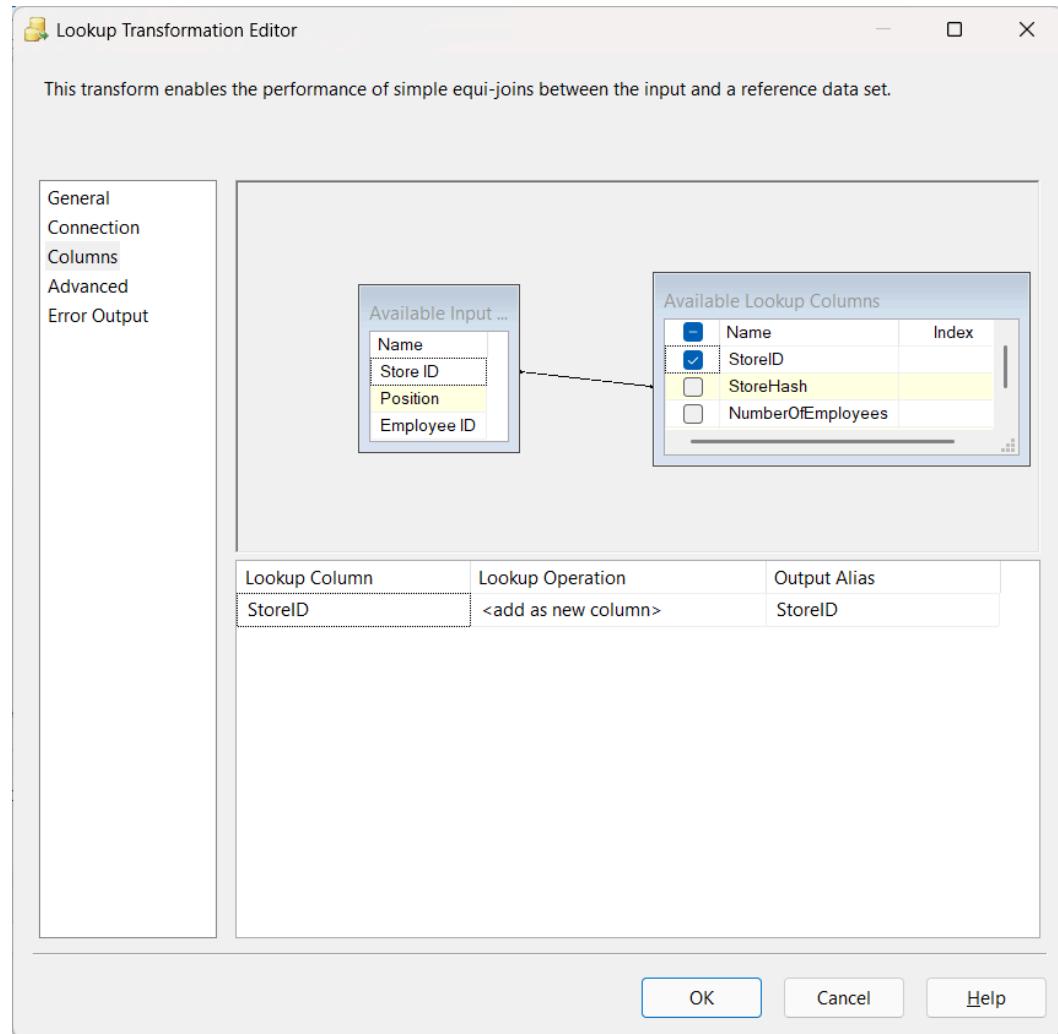
*Hình 91. Tạo chức năng LookupSTOREID trong Dim\_EMPLOYEES*

- **Bước 10:** Nhấn đúp chuột vào **LookupSTOREID**, màn hình **Lookup Transformation Editor** xuất hiện. Do chúng ta đã thiết lập **OLE DB connection manager** ở bước trước, nên ta có thể sử dụng lại. Ở phần **Use a table or a view**, tiến hành chọn **[dbo].[Dim\_STORES]**.



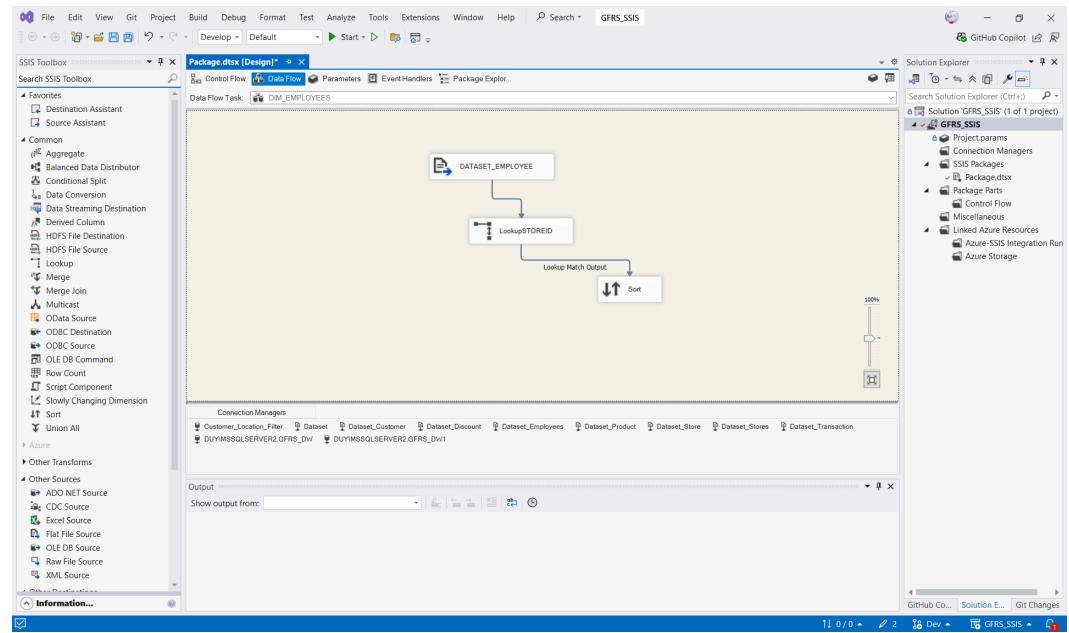
Hình 92. Lựa chọn nguồn dữ liệu cho LookupSTOREID

- **Bước 11:** Trong tab **Columns**, ở bảng bên phải click chọn **StoreID**. Ta kéo **StoreID** từ bảng **Available Input Columns** qua tương ứng **StoreHash** bảng **Available Lookup Columns** (quá trình này giúp hệ thống xác định vùng dữ liệu cần tìm kiếm). Sau đó bấm **OK** để đóng cửa sổ **Lookup Transformation Editor**



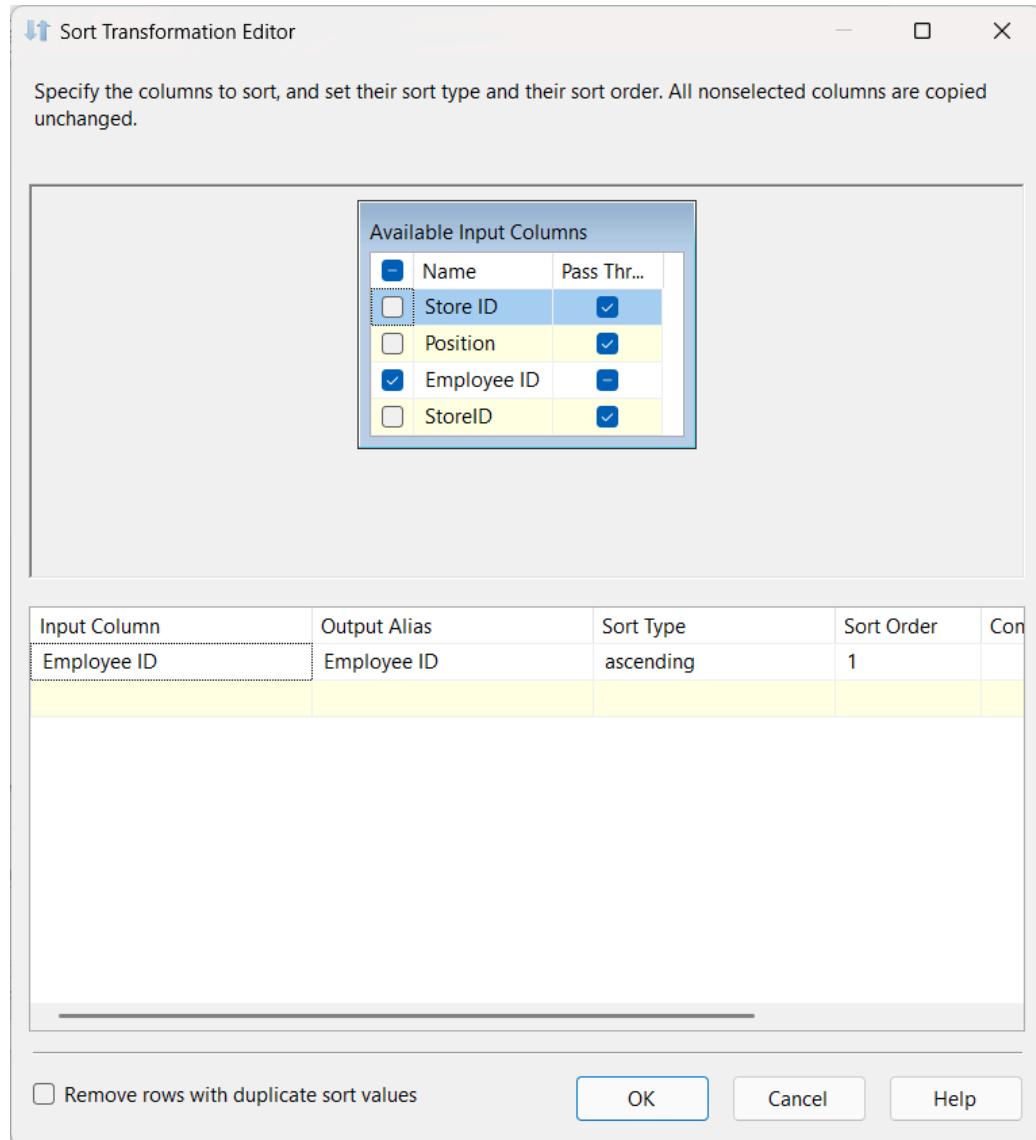
Hình 93. Cài đặt LookupSTOREID

- **Bước 12:** Kéo chức năng Sort vào vùng Data Flow và kéo mũi tên luồng dữ liệu từ LookupSTOREID vào Sort. Một hộp thoại sẽ xuất hiện, chọn **Lookup Match Output** và nhấn **OK**.



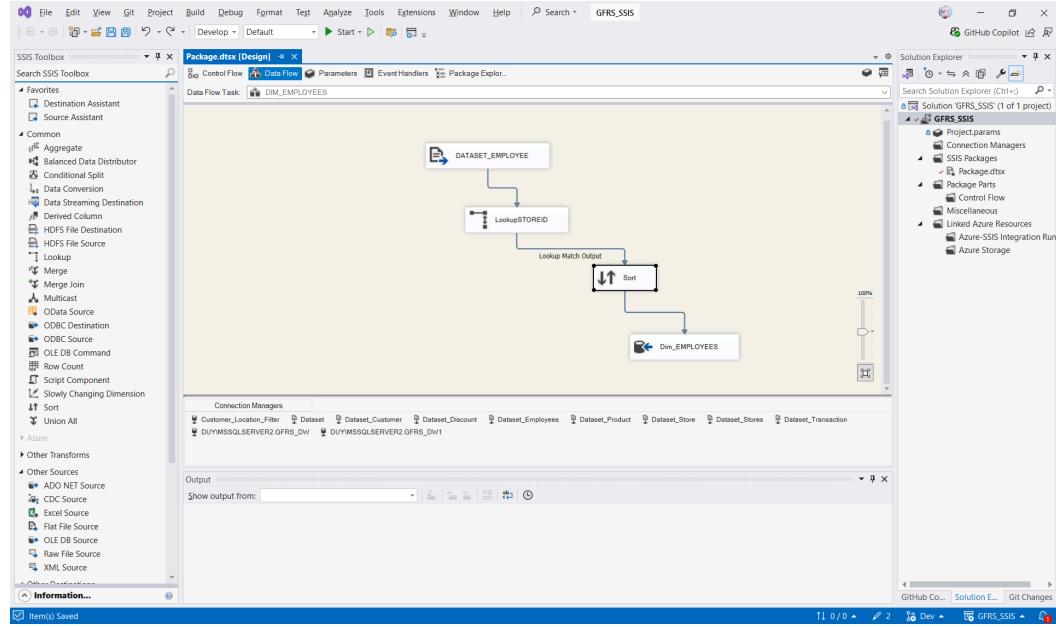
*Hình 94. Tạo chức năng Sort trong Dim\_EMPLOYEES*

- **Bước 13:** Nhấn đúp chuột vào **Sort** và màn hình **Sort Transformation Editor** xuất hiện. Nhấn **OK** và thoát màn hình **Sort Transformation Editor**.



Hình 95. Cài đặt chức năng Sort trong Dim\_EMPLOYEES

- **Bước 14:** Kéo chức năng ADO NET Destination vào vùng Data Flow và đổi tên thành **Dim\_EMPLOYEES**. Kéo mũi tên luồng dữ liệu từ **Sort** vào **Dim\_EMPLOYEES**.



Hình 96. Tạo ADO NET Destination Dim\_EMPLOYEES

- **Bước 15:** Vào SQL Server 2022, chạy lệnh DDL để tạo Dim\_EMPLOYEES

```

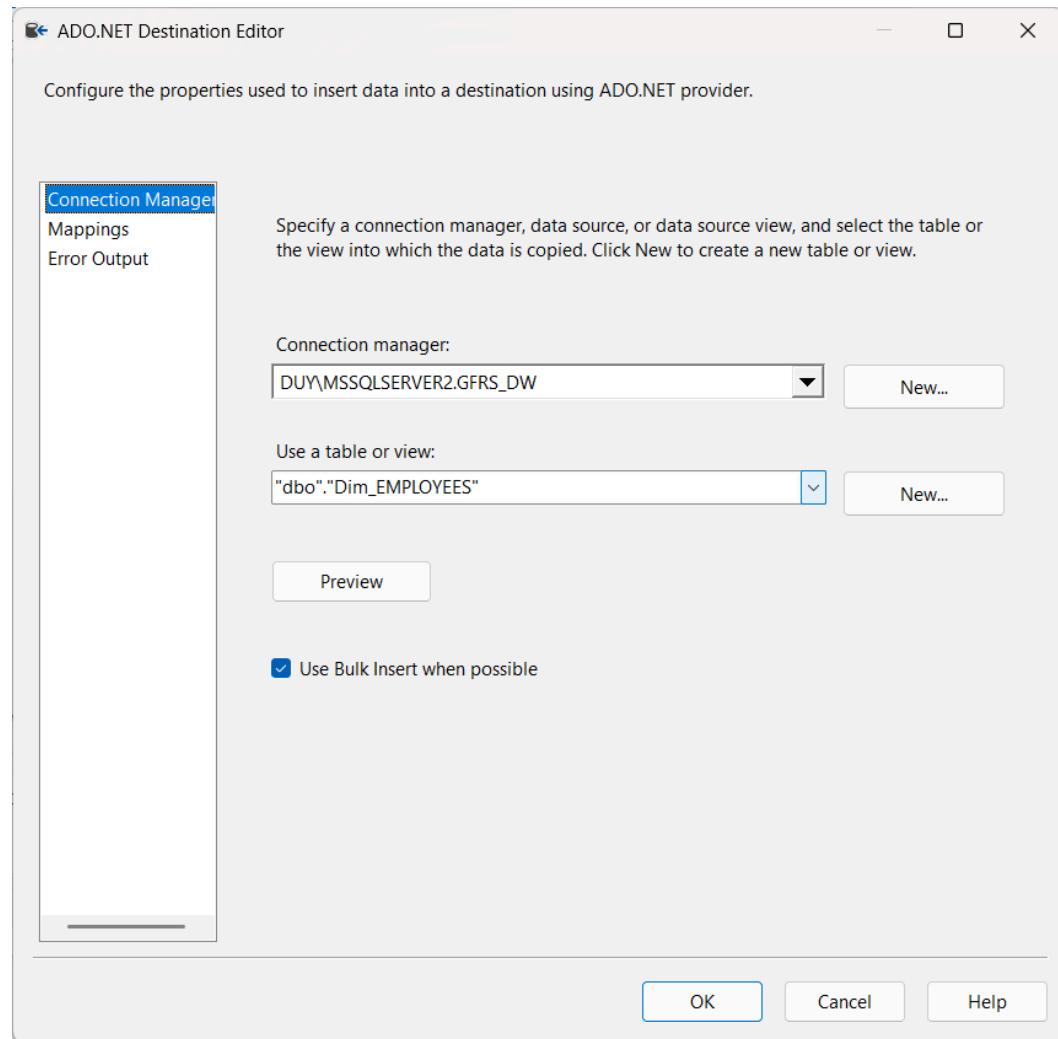
22     StoreID int identity(1,1) primary key,
23     StoreHash int,
24     NumberOfEmployees int,
25     LocationID int
26 );
27
28 drop table if EXISTS Dim_EMPLOYEES
29 CREATE TABLE Dim_EMPLOYEES (
30     [EmployeeID] integer identity(1,1) PRIMARY KEY,
31     [EmployeeHash] integer,
32     [StoreID] bigint,
33     [Position] varchar(255)
34 )
35
36 drop table if exists Dim_CATEGORIES
37 CREATE TABLE [Dim_CATEGORIES] (
38     [CategoryID] integer identity(1,1) PRIMARY KEY,
39     [Category] varchar(255)
40 )
41
42 drop table if exists Dim_SUBCATEGORIES
43 CREATE TABLE [Dim_SUBCATEGORIES] (
44     [SubCategoryID] integer identity(1,1) PRIMARY KEY,
45     [SubCategory] varchar(255)
46 )
47

```

Hình 97. Chạy lệnh DDL tạo Dim\_EMPLOYEES

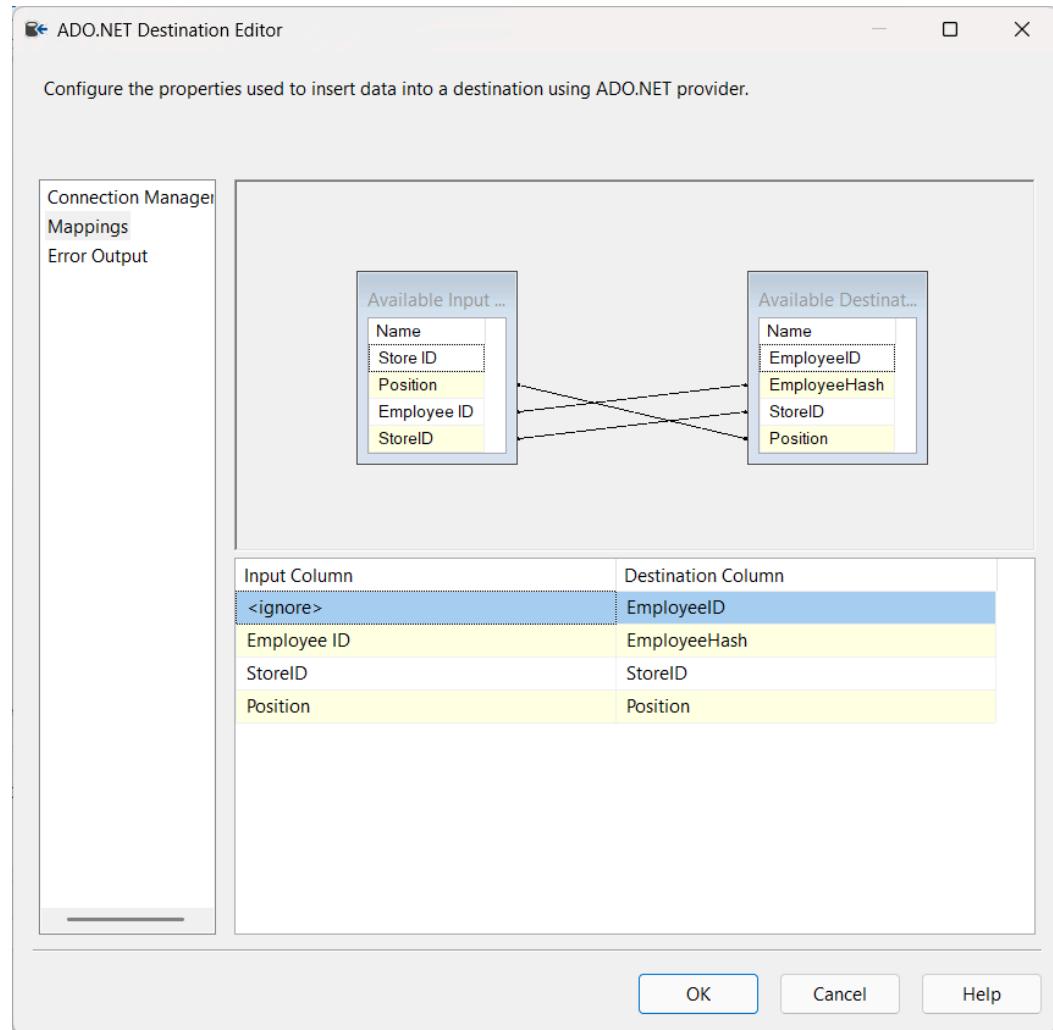
- **Bước 16:** Nhấn đúp chuột vào Dim\_EMPLOYEES. Trong tab Connection Manager của màn hình ADO.NET Destination Editor, ở Use a table or

view tiến hành nhân chọn “**dbo”.”Dim\_EMPLOYEES**”, cũng chính là bảng vừa được tạo bằng lệnh **DDL** ở **SQL Server 2022**



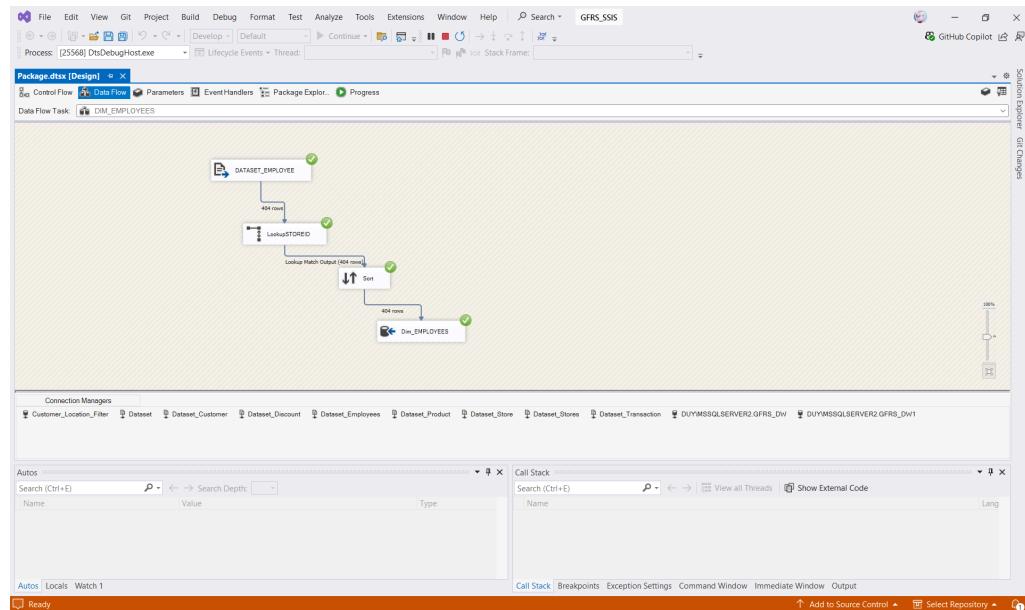
Hình 98. Kết nối đích đến cho Dim\_EMPLOYEES

- **Bước 17:** Ở tab **Mapping** của màn hình **ADO.NET Destination Editor** tiến hành chọn các thuộc tính thích hợp ở cột bên trái để ánh xạ với các thuộc tính ở cột bên phải.



Hình 99. Cài đặt ADO NET Destination Dim\_EMPLOYEES

- **Bước 18:** Nhấn Start để kiểm tra quá trình tạo và đổ dữ liệu vào Dim\_EMPLOYEES có thành công hay chưa.



Hình 100. Đỗ dữ liệu vào Dim\_EMPLOYEES

Kết quả hiển thị cho thấy quá trình diễn ra thành công và có **404 rows** được đỗ vào **DIM\_STORES**.

- **Bước 19:** Vào SQL Server 2022 kiểm tra và thấy dữ liệu đã được đỗ vào Dim\_EMPLOYEES thành công.

```

File Edit View Query Git Project Tools Extensions Window Help Search Solution1
DUY\maing (85) Database.sql ...U\maing (51)
28 drop table if EXISTS Dim_EMPLOYEES
29
30 CREATE TABLE Dim_EMPLOYEES (
31     [EmployeeID] integer identity(1,1) PRIMARY KEY,
32     [EmployeeHash] integer,
33     [StoreID] bigint,
34     [Position] varchar(255)
35 )
36 select * from Dim_EMPLOYEES;
37
38 drop table if exists Dim_CATEGORIES
39 CREATE TABLE [Dim_CATEGORIES] (
40     [CategoryID] integer identity(1,1) PRIMARY KEY,
        [Category] varchar(255)

Results Messages
EmployeeID EmployeeHash StoreID Position
1 1 1 Store Manager
2 2 2 1 Assistant Manager
3 3 3 1 Cashier
4 4 4 1 Stock Clerk
5 5 5 1 Sales Associate
6 6 6 1 Sales Associate
7 7 7 1 Sales Associate
8 8 8 1 Sales Associate
9 9 9 1 Sales Associate
10 10 10 1 Sales Associate
11 11 11 1 Sales Associate
12 12 12 1 Sales Associate
13 13 13 1 Sales Associate
14 14 14 2 Store Manager

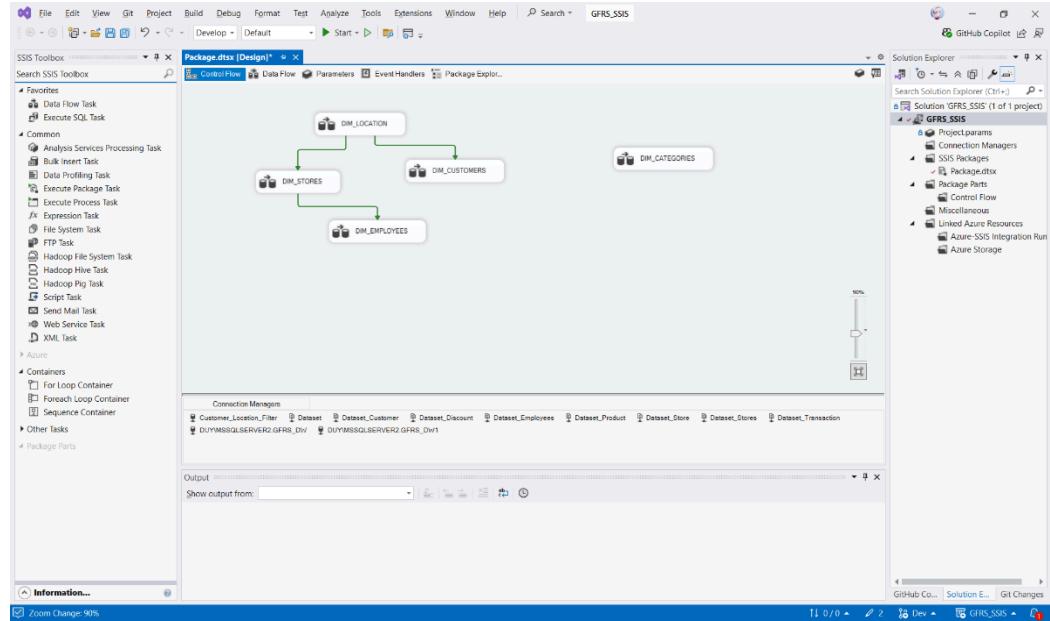
Query executed successfully.

```

Hình 101. Dữ liệu được đỗ thành công vào Dim\_EMPLOYEES

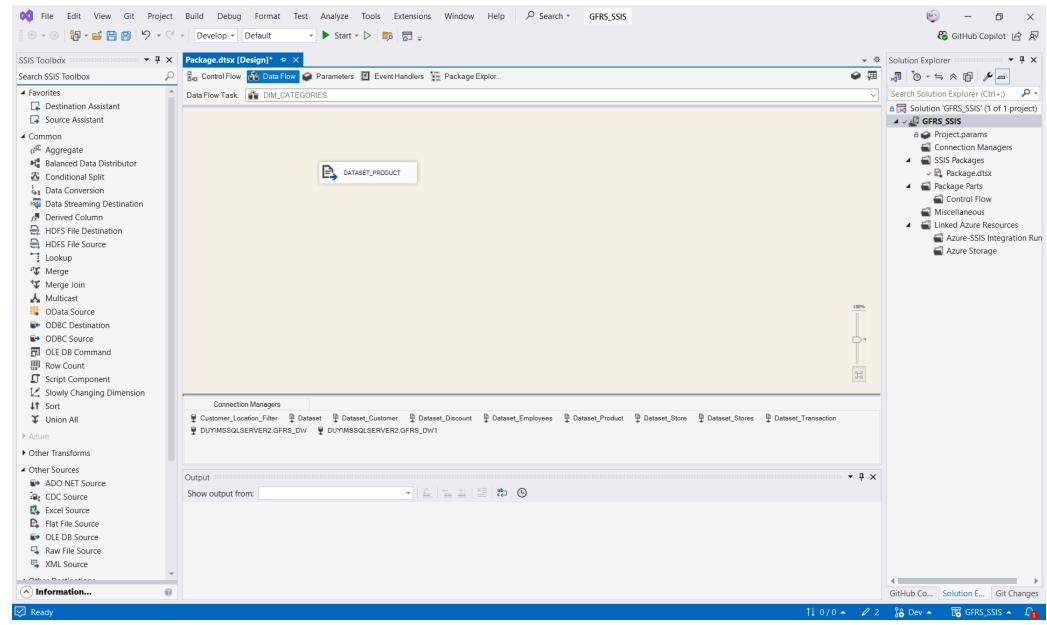
#### II.3.4. Dim\_CATEGORIES

- **Bước 1:** Kéo chức năng Data Flow Task trong vùng SSIS Toolbox vào vùng Control Flow và đổi tên thành **Dim\_CATEGORIES**.



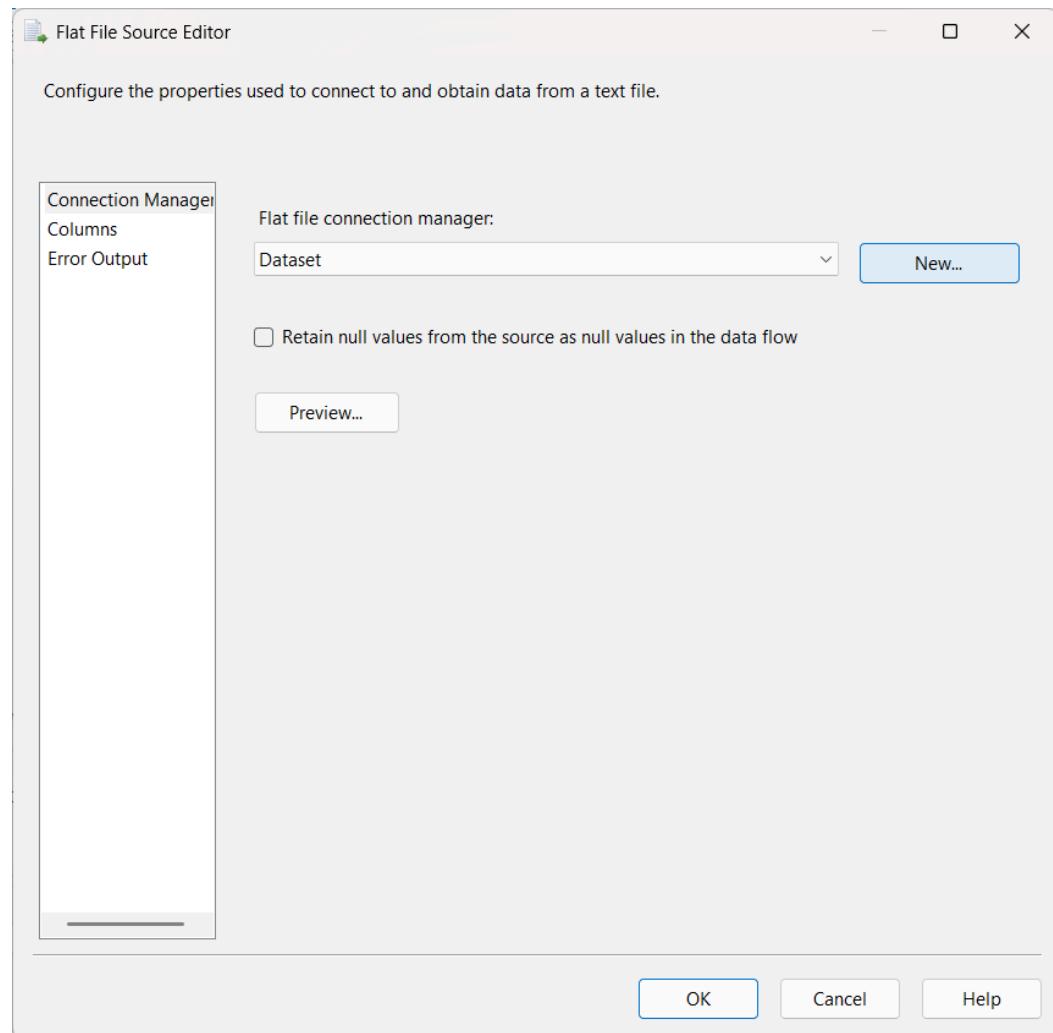
Hình 102. Tạo chức năng Dim\_CATEGORIES

- **Bước 2:** Nhấn đúp chuột vào **Dim\_CATEGORIES**. Sau đó kéo chức năng **Flat File Source** vào vùng làm việc và đổi tên thành **DATASET\_PRODUCT**



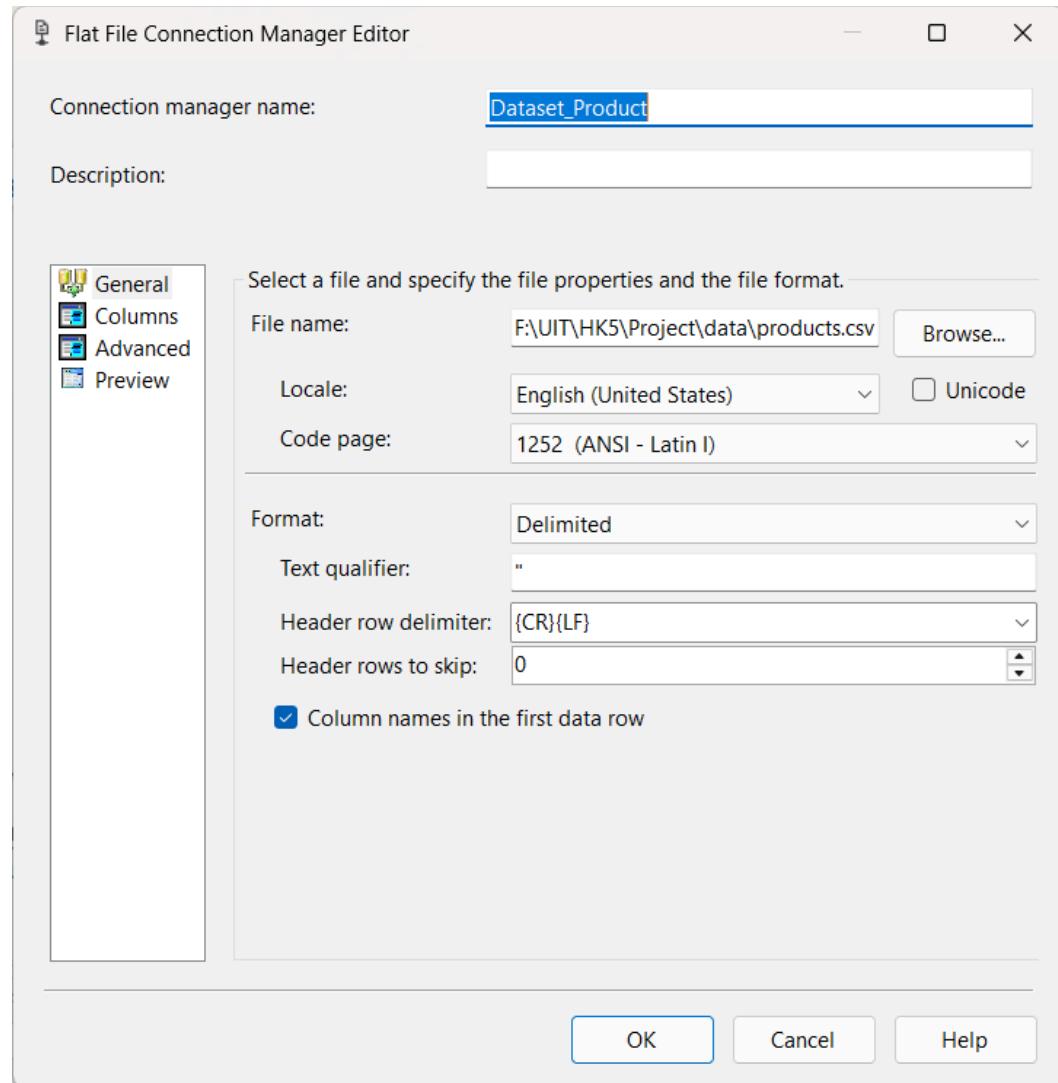
*Hình 103. Tạo DATASET\_PRODUCT trong Dim\_CATEGORIES*

- **Bước 3:** Nhấn đúp chuột vào DATASET\_PRODUCT. Màn hình **Flat File Source Editor** hiện ra. Ở tab **Connection Manager** nhấn **New**



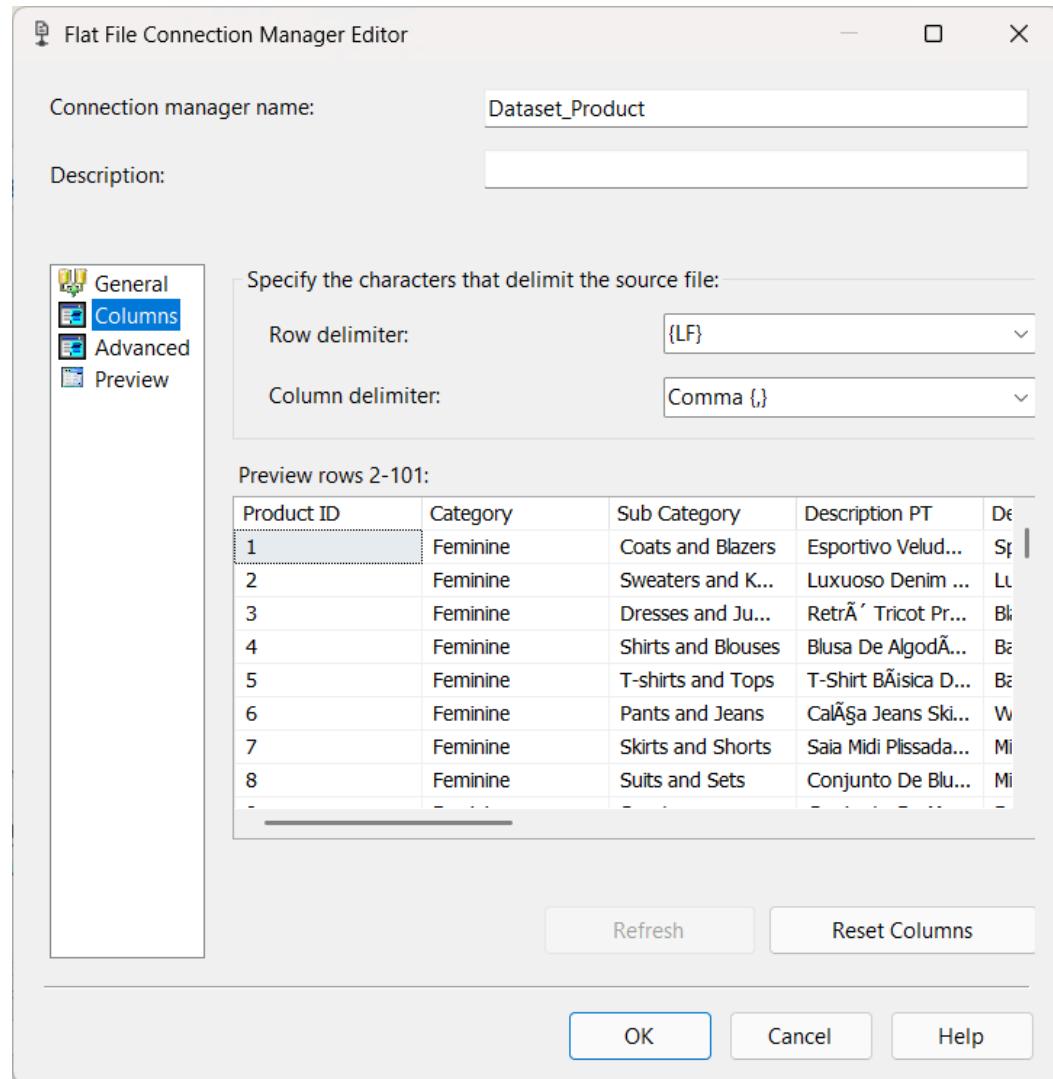
*Hình 104. Tạo nguồn dữ liệu cho DATASET\_PRODUCT trong Dim\_CATEGORIES*

- **Bước 4:** Màn hình **Flat File Connection Manager Editor** hiện ra. Ở tab **General**, nhấn **Browse** để tiến hành load file csv **products.csv** trong bộ dữ liệu **Global Fashion Retail Sales**. Ở mục **Connection manager name**, đặt tên là **Dataset\_Product**. Ở mục **Text qualifier**, nhập ký tự dấu nháy kép “



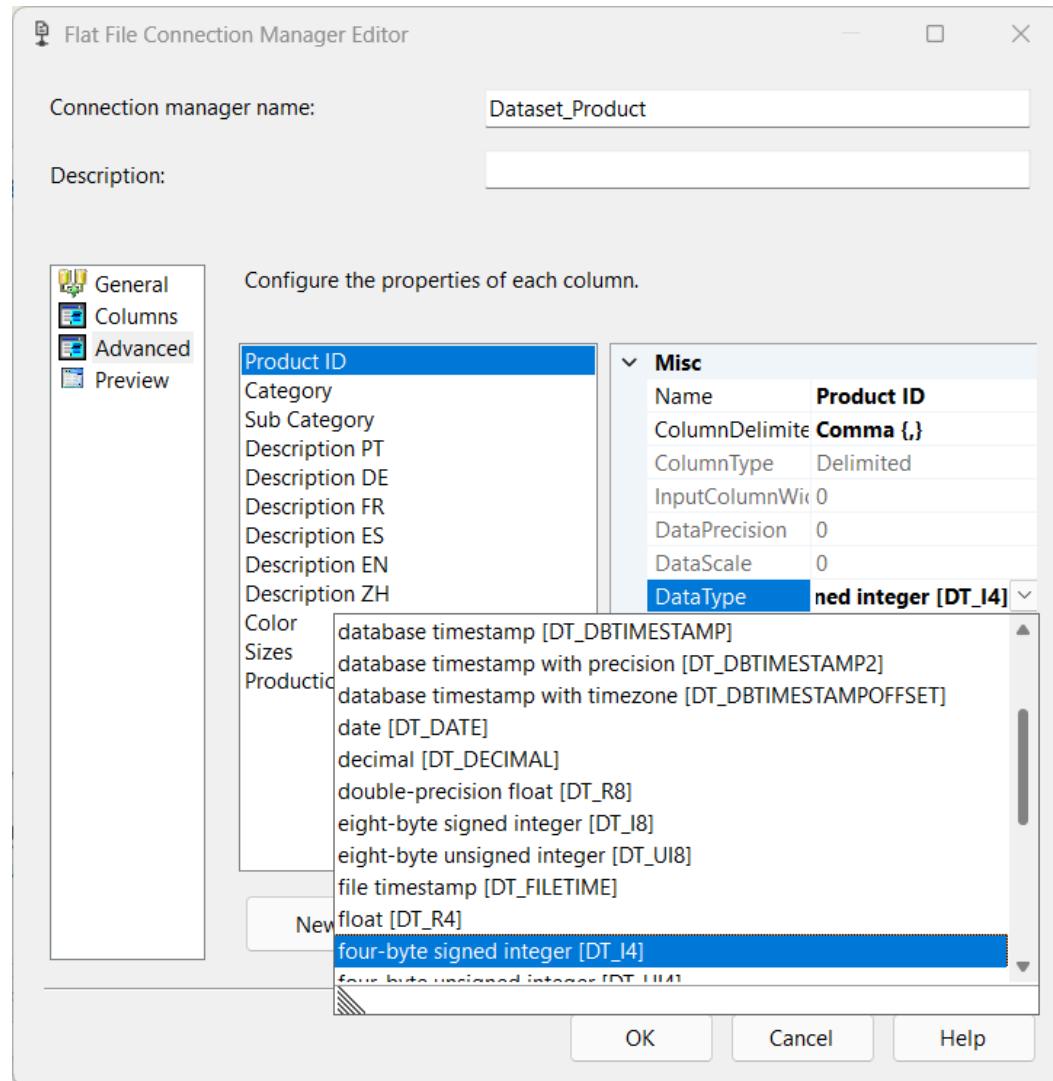
Hình 105. Tạo Dataset\_Product

- **Bước 5:** Ở tab **Columns**, ta có thể kiểm tra và thấy rằng file csv đã được load thành công



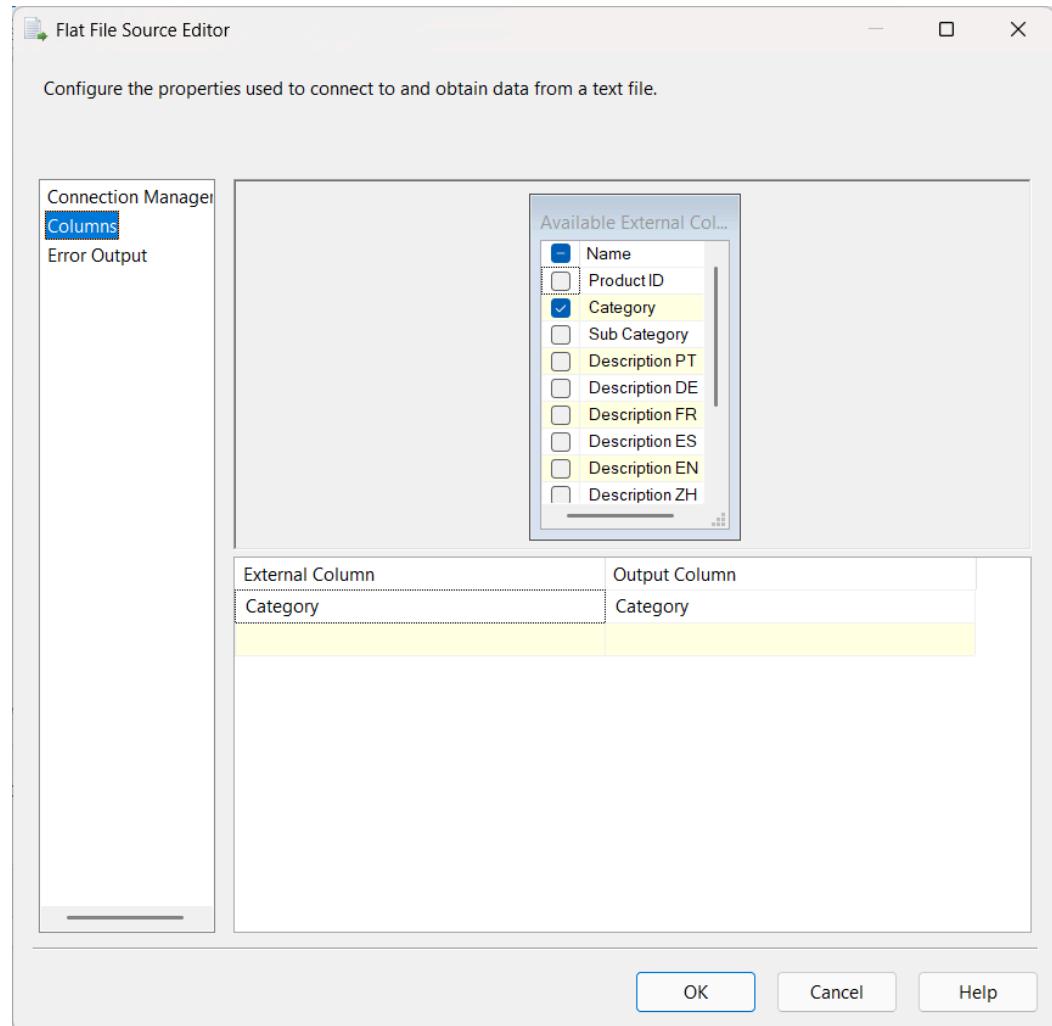
Hình 106. Kiểm tra dữ liệu Dataset\_Product trước khi nạp

- **Bước 6:** Ở tab **Advanced**, tiến hành thay đổi kiểu dữ liệu của các thuộc tính cần sử dụng trong quá trình **SSIS** của file csv tương ứng với kiểu dữ liệu khai báo của thuộc tính đó trong **SQL Server**. Sau khi thay đổi kiểu dữ liệu của các thuộc tính cần sử dụng trong quá trình **SSIS** thì nhấn **OK** và sẽ thoát khỏi màn hình **Flat File Connection Manager Editor**.



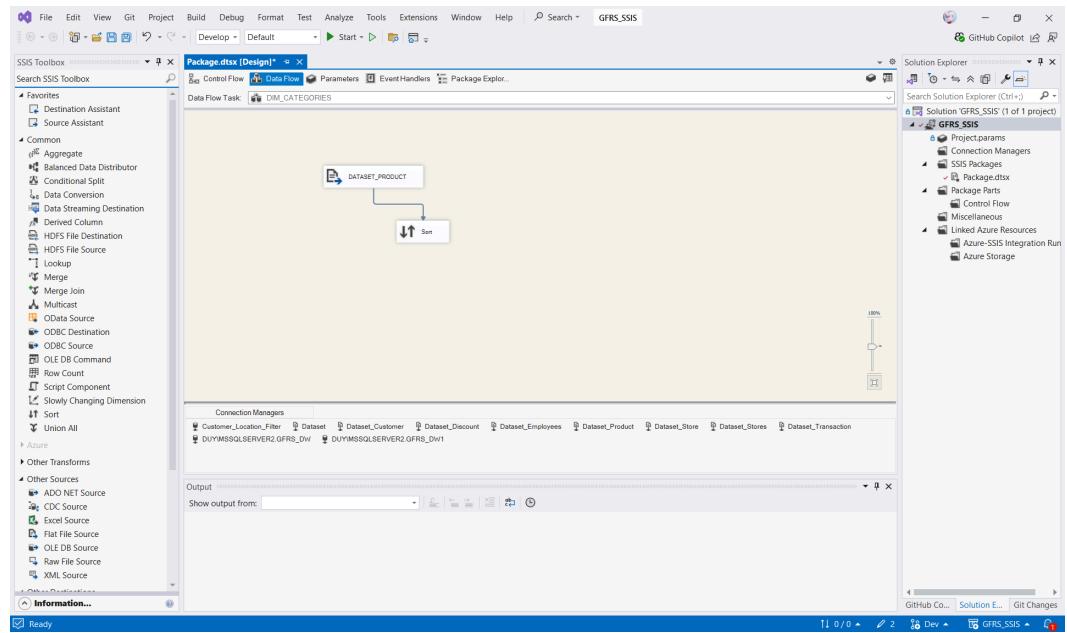
Hình 107. Lựa chọn kiểu dữ liệu phù hợp cho các thuộc tính trong Dataset\_Product

- **Bước 7:** Ở màn hình **Flat File Source Editor**, qua tab **Columns** và nhấn chọn các thuộc tính cần sử dụng trong quá trình tạo **Dim\_CATEGORIES**



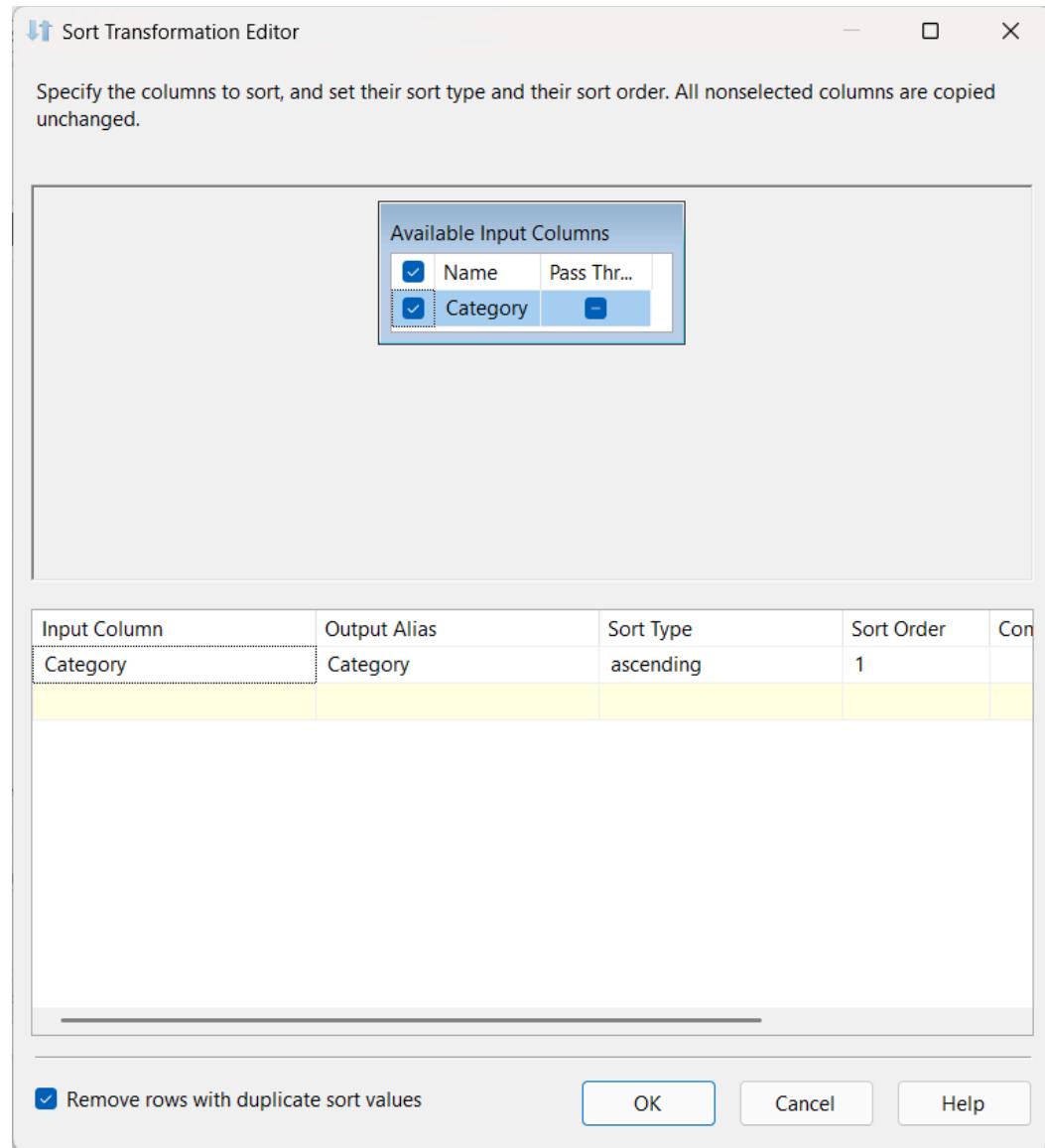
Hình 108. Lựa chọn thuộc tính cần thiết trong quá trình tạo  
*Dim\_CATEGORIES*

- **Bước 8:** Nhấn OK để thoát màn hình **Flat File Source Editor**
- **Bước 9:** Kéo chức năng **Sort** vào vùng **Data Flow** và kéo mũi tên luồng dữ liệu từ **DATASET\_PRODUCT** vào **Sort**.



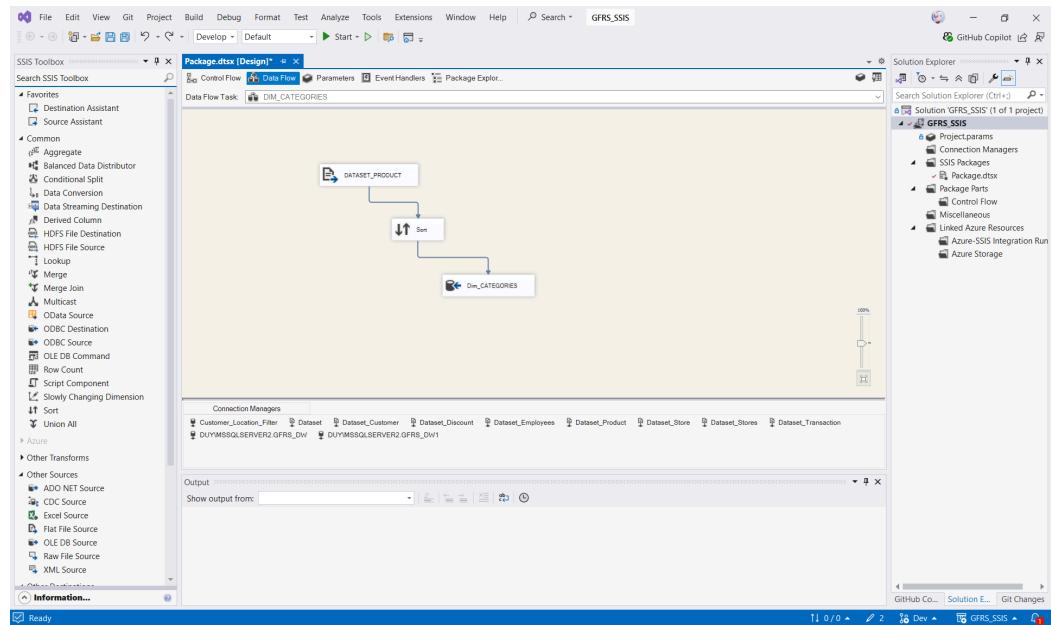
*Hình 109. Tạo chức năng Sort trong Dim\_CATEGORIES*

- **Bước 10:** Nhấn đúp chuột vào **Sort** và màn hình **Sort Transformation Editor** xuất hiện. Chọn **Remove rows with duplicate sort values** để lọc các dữ liệu trùng lặp. Nhấn **OK** và thoát màn hình **Sort Transformation Editor**.



Hình 110. Cài đặt chức năng Sort trong Dim\_CATEGORIES

- **Bước 11:** Kéo chức năng ADO NET Destination vào vùng Data Flow và đổi tên thành **Dim\_CATEGORIES**. Kéo mũi tên luồng dữ liệu từ **Sort** vào **Dim\_CATEGORIES**.



Hình 111. Tạo chức năng ADO.NET Destination Dim\_CATEGORIES

- **Bước 12:** Vào SQL Server 2022, chạy lệnh DDL để tạo Dim\_CATEGORIES

```

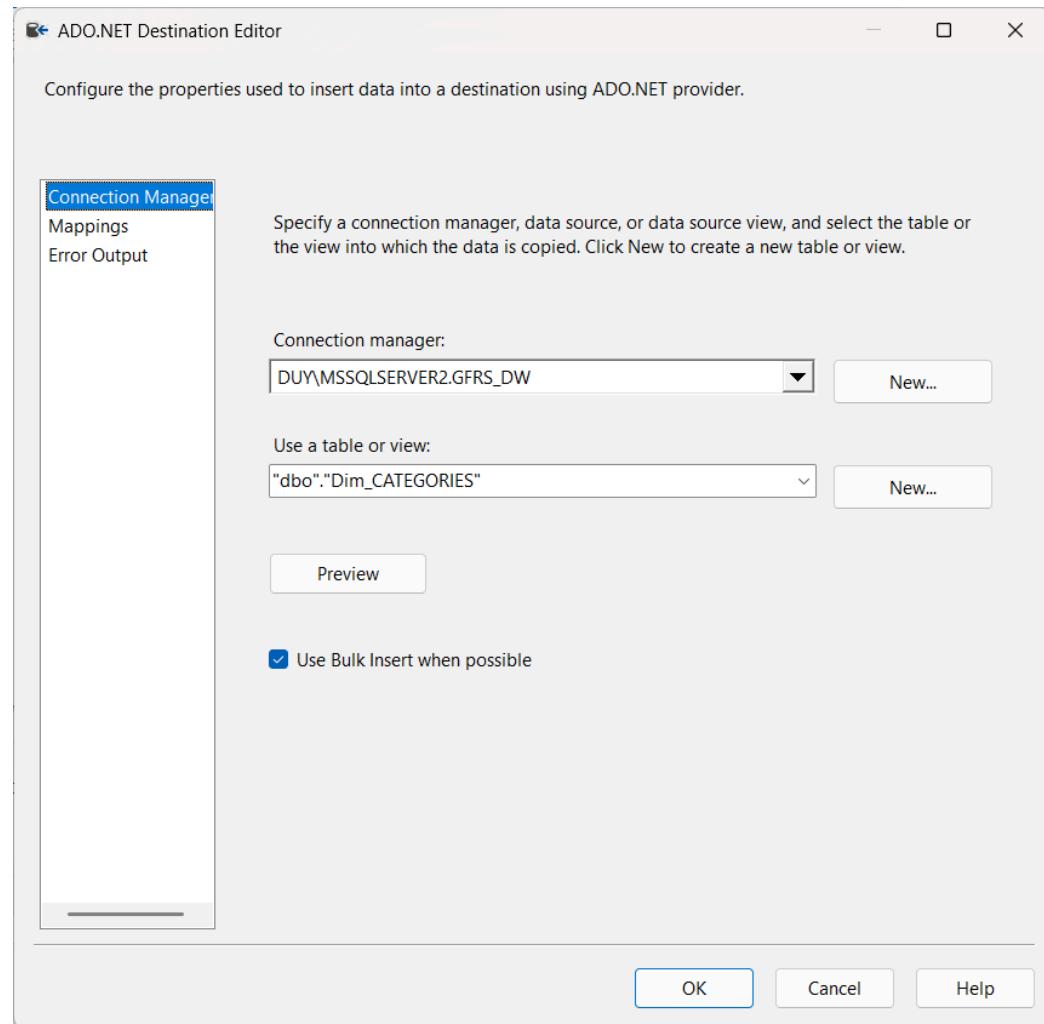
File Edit View Query Git Project Tools Window Help | Search Solution1
File Edit View Query Git Project Tools Extensions Window Help | Search
GFRS_DW | Execute | New Query | Database | Object Explorer | Results | Messages
Object Explorer | SQLQuery1... | maing (85)* | DataBase.sql... | maing (51)
31   [EmployeeHash] integer,
32   [StoreID] bigint,
33   [Position] varchar(255)
34 )
35 select * from Dim_EMPLOYEES;
36
37 drop table if exists Dim_CATEGORIES
38 CREATE TABLE [Dim_CATEGORIES] (
39   [CategoryID] integer identity(1,1) PRIMARY KEY,
40   [Category] varchar(255)
41 )
42
43 drop table if exists Dim_SUBCATEGORIES
44
45
Results | Messages
1 EmployeeID EmployeeHash StoreID Position
2 1 1 1 Store Manager
3 2 2 1 Assistant Manager
4 3 3 1 Cashier
5 4 4 1 Stock Clerk
6 5 5 1 Sales Associate
7 6 6 1 Sales Associate
8 7 7 1 Sales Associate
9 8 8 1 Sales Associate
10 9 9 1 Sales Associate
11 10 10 1 Sales Associate
12 11 11 1 Sales Associate
13 12 12 1 Sales Associate
14 13 13 1 Sales Associate
15 14 14 2 Store Manager
Ln: 38 Ch: 1 TABS CRLF
Query executed successfully.
DUY\MSQLSERVER2 (16.0 RTM) | DUY\maing (85) | GFRS_DW | 00:00:00 | 404 rows
Ready

```

Hình 112. Chạy lệnh DDL tạo Dim\_CATEGORIES

- **Bước 13 :**Nhấn đúp chuột vào Dim\_CATEGORIES. Trong tab Connection Manager của màn hình ADO.NET Destination Editor, ở Use

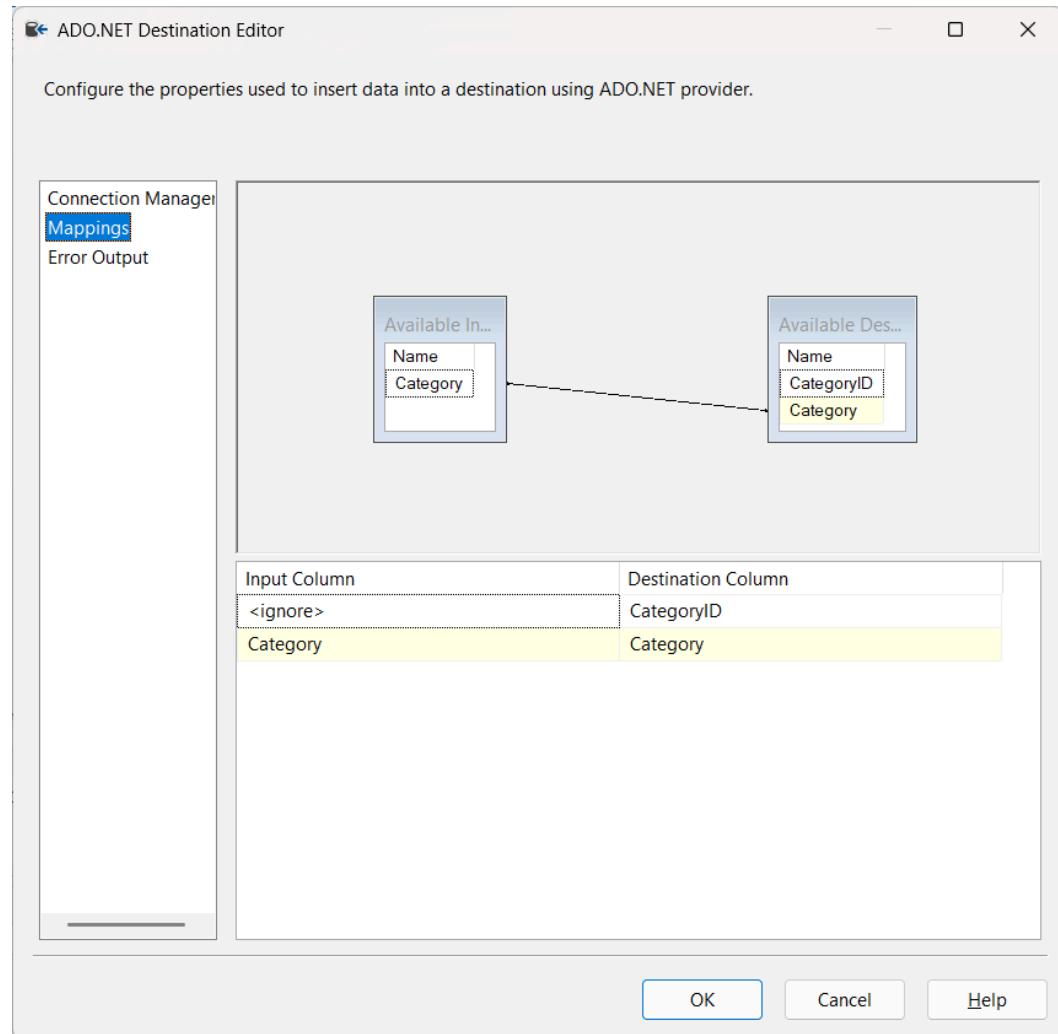
a table or view tiến hành nhấn chọn “dbo”.”Dim\_CATEGORIES”, cũng chính là bảng vừa được tạo bằng lệnh DDL ở SQL Server 2022



Hình 113. Lựa chọn đích đến cho ADO NET Destination

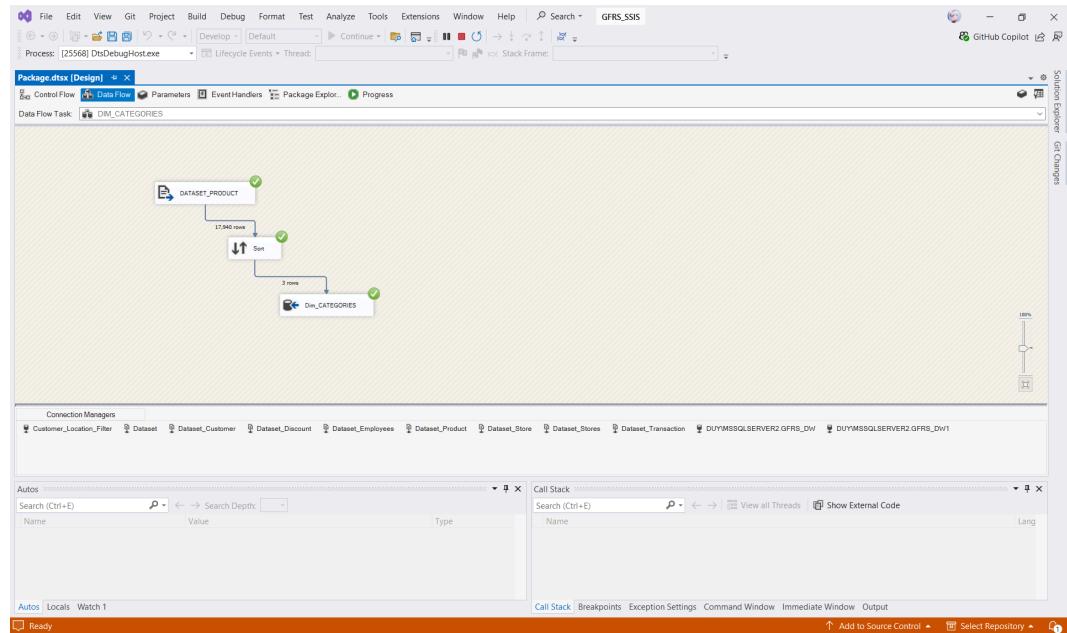
#### Dim\_CATEGORIES

- **Bước 14:** Ở tab **Mapping** của màn hình **ADO.NET Destination Editor** tiến hành chọn các thuộc tính thích hợp ở cột bên trái để ánh xạ với các thuộc tính ở cột bên phải.



Hình 114. Cài đặt ADO NET Destination Dim\_CATEGORIES

- **Bước 15:** Nhấn Start để kiểm tra quá trình tạo và đổ dữ liệu vào Dim\_CATEGORIES có thành công hay chưa.



Hình 115. Đồ dữ liệu vào Dim\_CATEGORIES

Kết quả hiển thị cho thấy quá trình diễn ra thành công và có **3 rows** được đổ vào **DIM\_CATEGORIES**.

- **Bước 16:** Vào SQL Server 2022 kiểm tra và thấy dữ liệu đã được đổ vào Dim\_CATEGORIES thành công.

The screenshot shows the SQL Server Management Studio (SSMS) interface. In the Object Explorer, the database 'GFRS\_DW' is selected. In the main pane, a query window titled 'SQLQuery1.s... \maing (85)' is open, displaying T-SQL code for creating two tables: 'Dim\_CATEGORIES' and 'Dim\_SUBCATEGORIES'. The code includes dropping existing tables if they exist, creating the tables with their respective columns (CategoryID and Category for Dim\_CATEGORIES, and SubCategoryID and SubCategory for Dim\_SUBCATEGORIES), and selecting all data from the newly created Dim\_CATEGORIES table. The results pane shows the data inserted into the Dim\_CATEGORIES table:

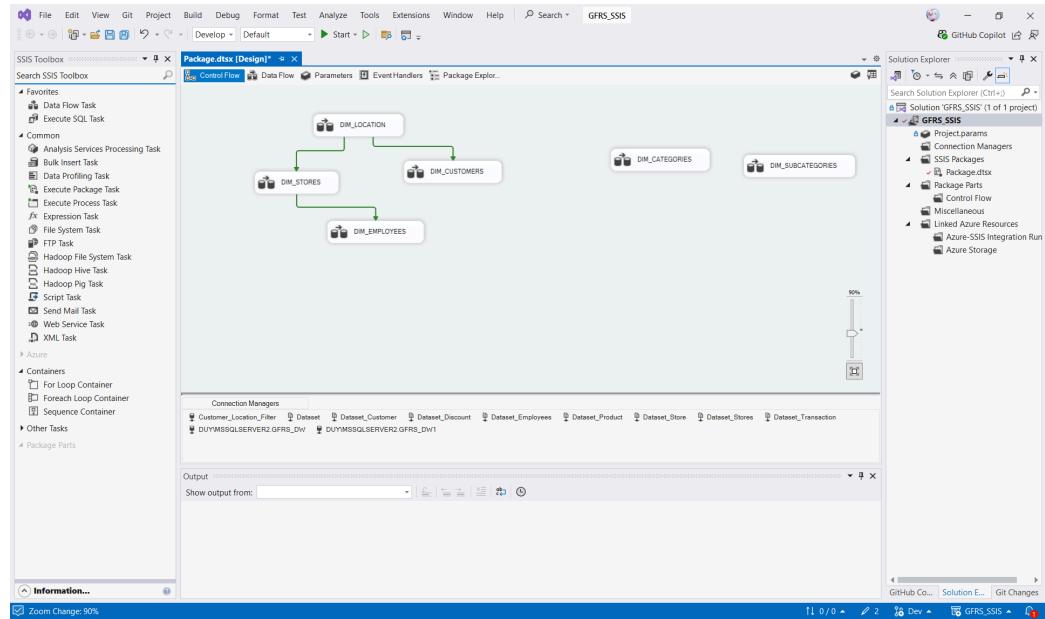
CategoryID	Category
1	Children
2	Feminine
3	Masculine

A message at the bottom of the results pane states 'Query executed successfully.'

Hình 116. Dữ liệu được đổ thành công vào Dim\_CATEGORIES

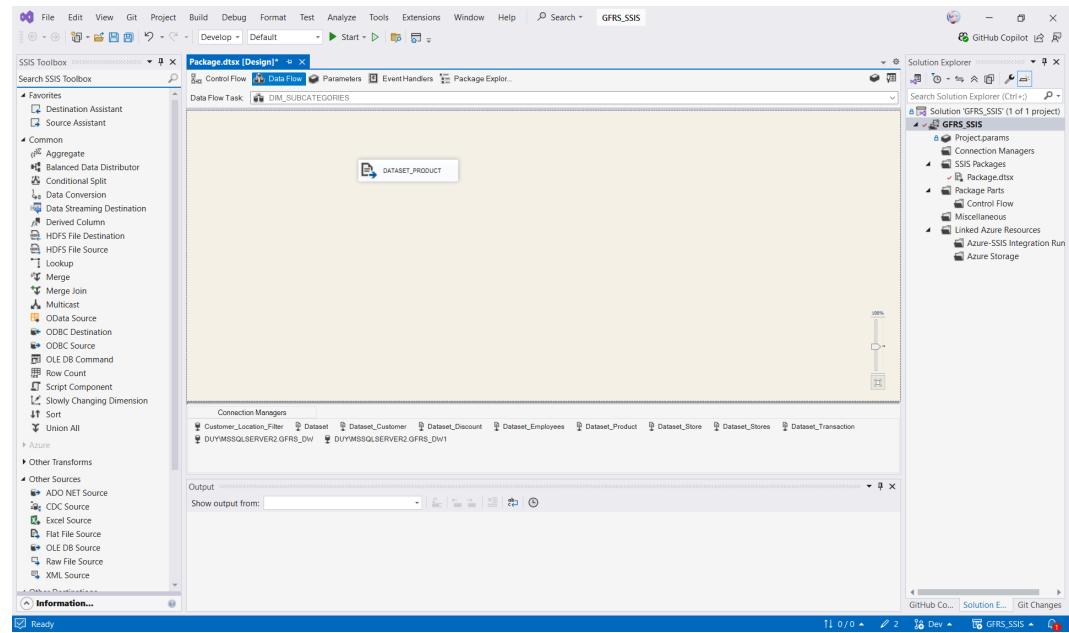
### II.3.5. Dim\_SUBCATEGORIES

- **Bước 1:** Kéo chức năng Data Flow Task trong vùng SSIS Toolbox vào vùng Control Flow và đổi tên thành **Dim\_SUBCATEGORIES**.



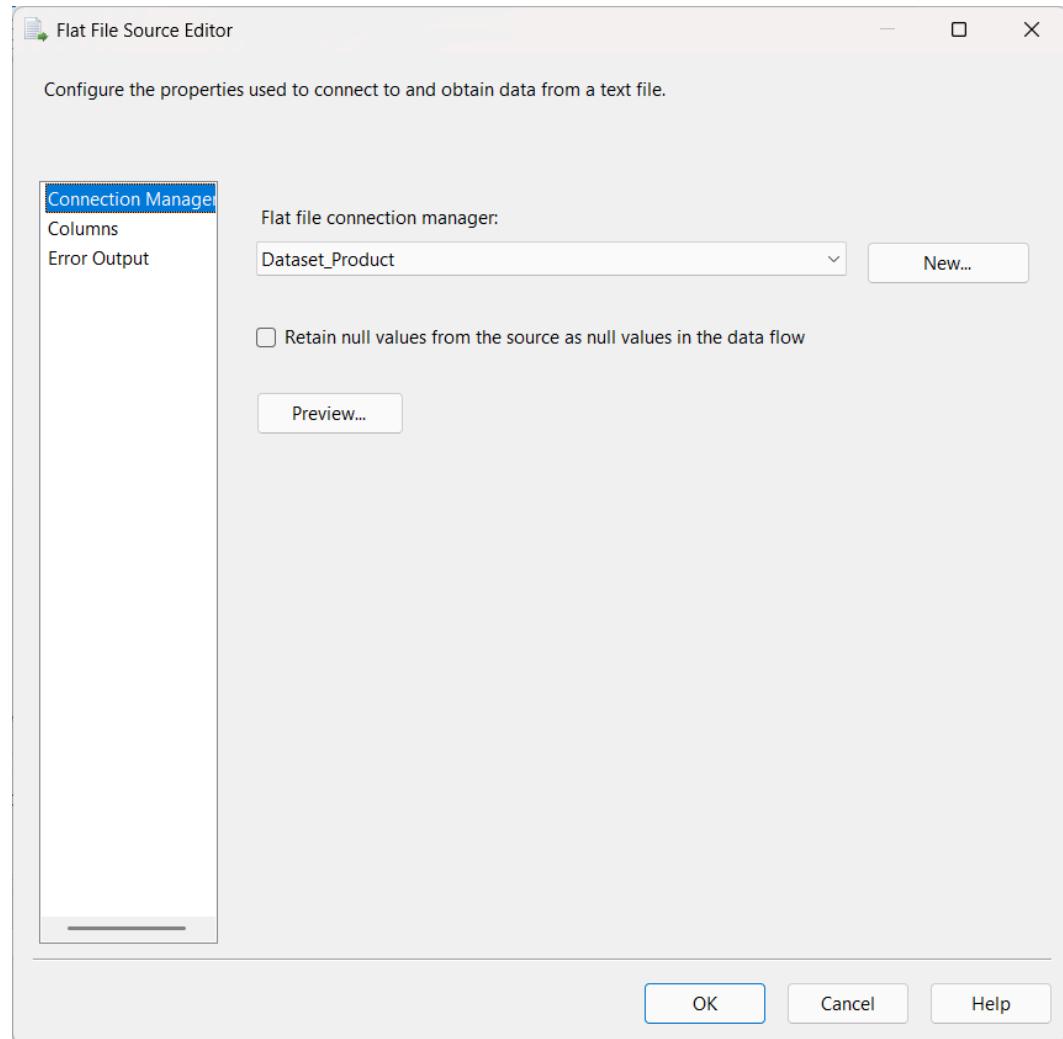
Hình 117. Tạo chức năng Dim\_SUBCATEGORIES

- **Bước 2:** Nhấn đúp chuột vào **Dim\_SUBCATEGORIES**. Sau đó kéo chức năng **Flat File Source** vào vùng làm việc và đổi tên thành **DATASET\_PRODUCT**



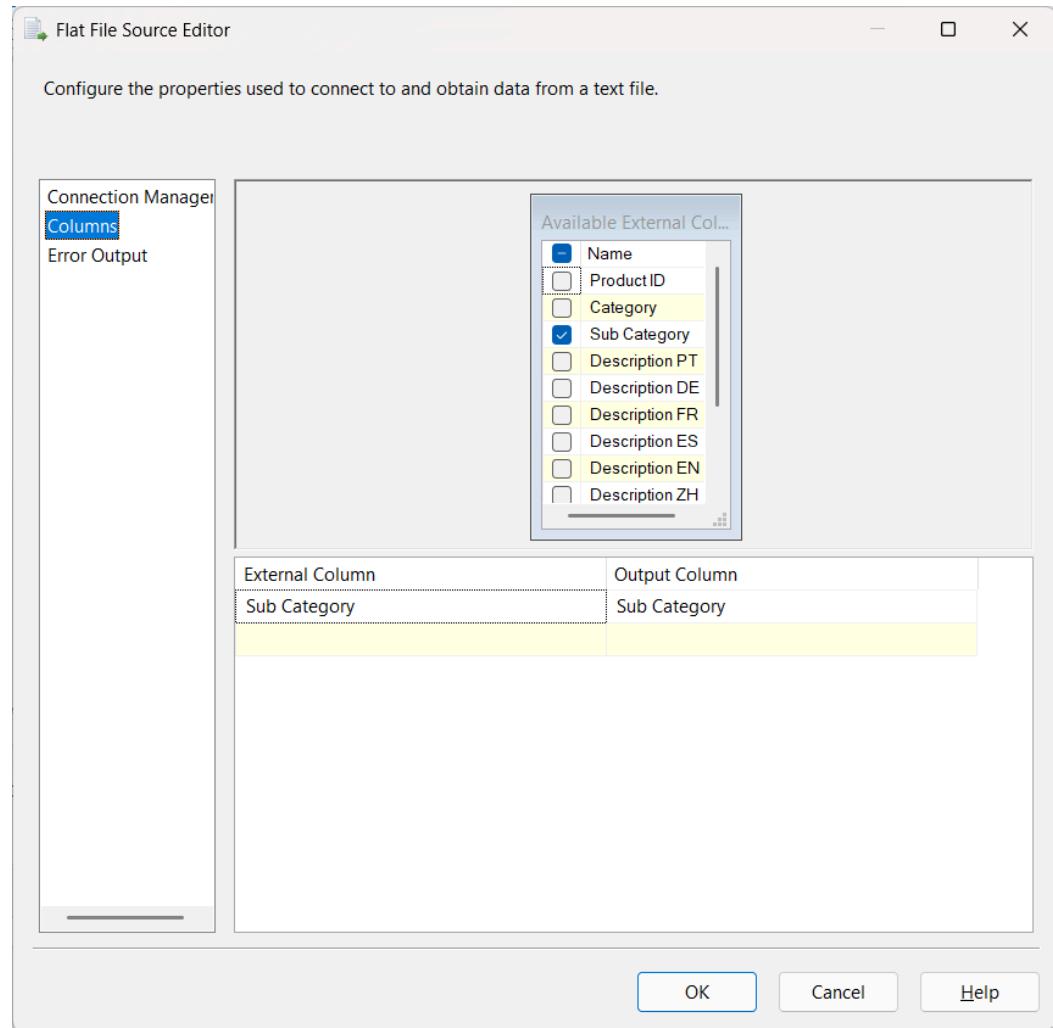
*Hình 118. Tạo DATASET\_PRODUCT trong Dim\_SUBCATEGORIES*

- **Bước 3:** Nhấn đúp chuột vào **DATASET\_PRODUCT**. Màn hình **Flat File Source Editor** hiện ra. Ở tab **Connection Manager** ta chọn **Dataset\_Product** đã được thiết lập ở phần trước



Hình 119. Chọn nguồn dữ liệu cho quá trình Dim\_SUBCATEGORIES

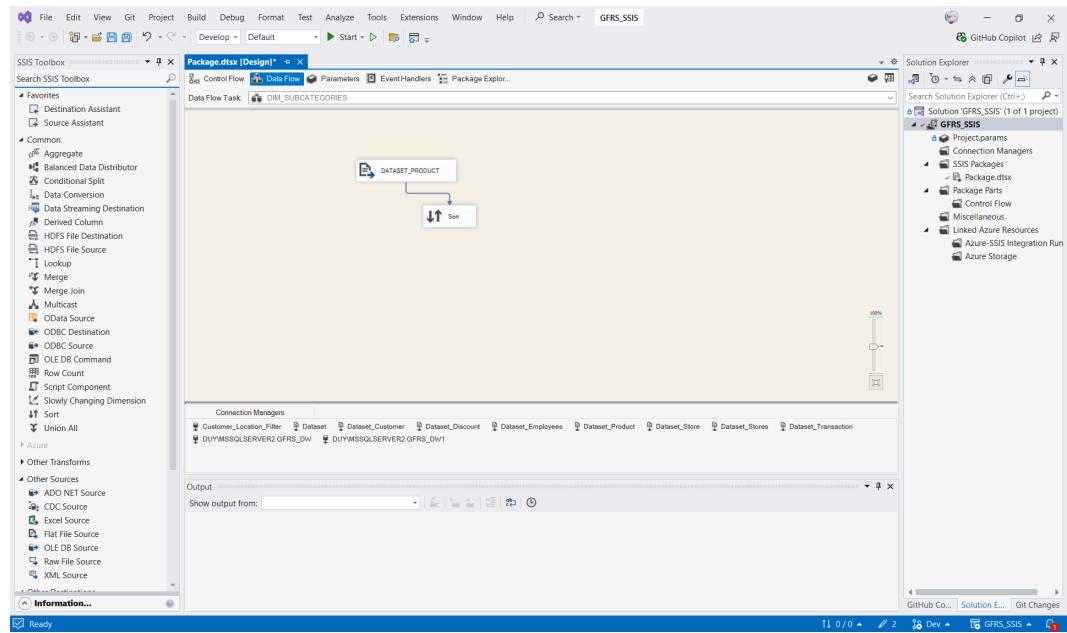
- **Bước 4:** Ở màn hình **Flat File Source Editor**, qua tab **Columns** và nhấn chọn các thuộc tính cần sử dụng trong quá trình tạo **Dim\_SUBCATEGORIES**



Hình 120. Lựa chọn thuộc tính cần thiết cho quá trình tạo

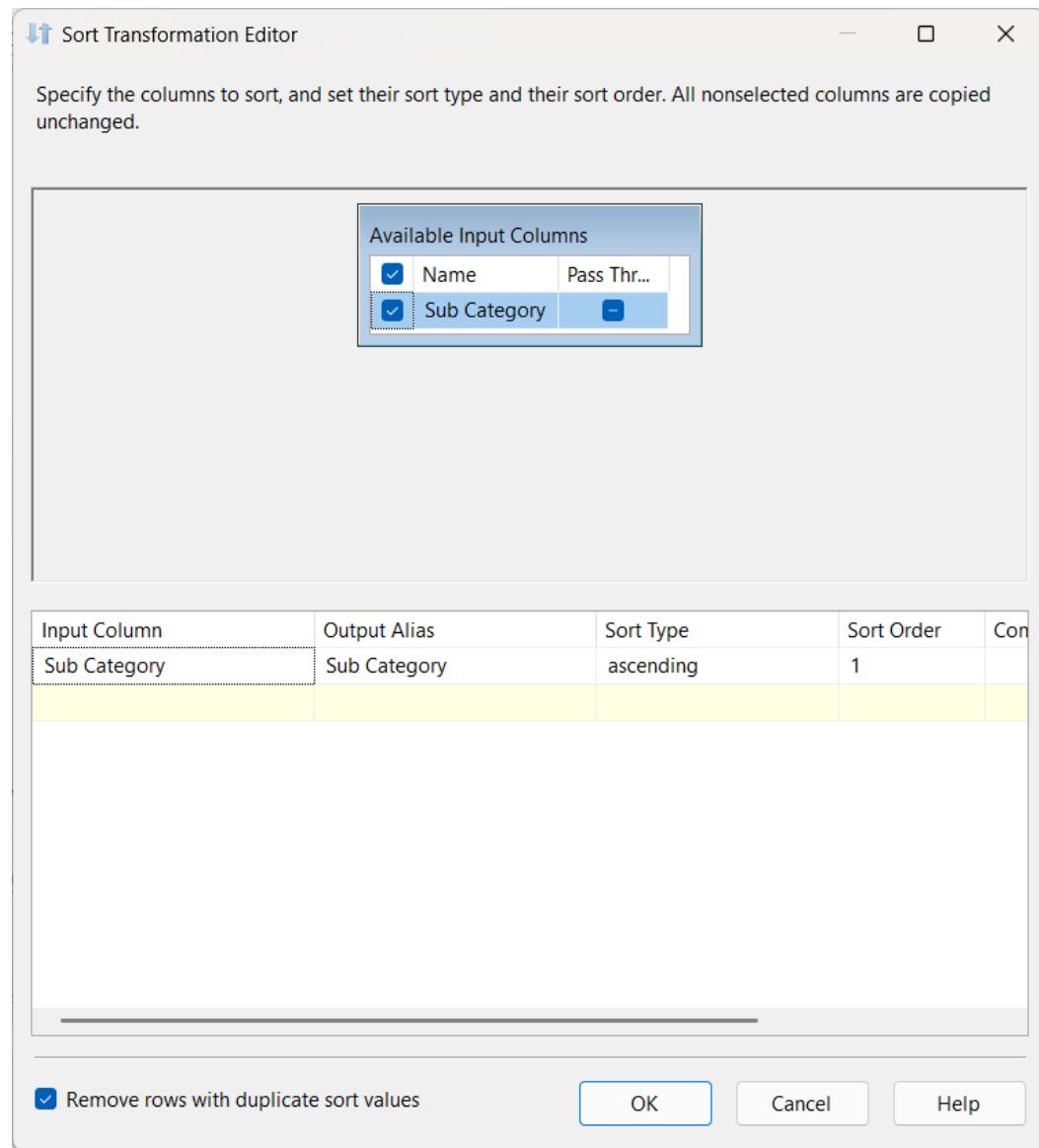
*Dim\_SUBCATEGORIES*

- **Bước 5:** Nhấn OK để thoát màn hình **Flat File Source Editor**
- **Bước 6:** Kéo chức năng **Sort** vào vùng **Data Flow** và kéo mũi tên luồng dữ liệu từ **DATASET\_PRODUCT** vào **Sort**.



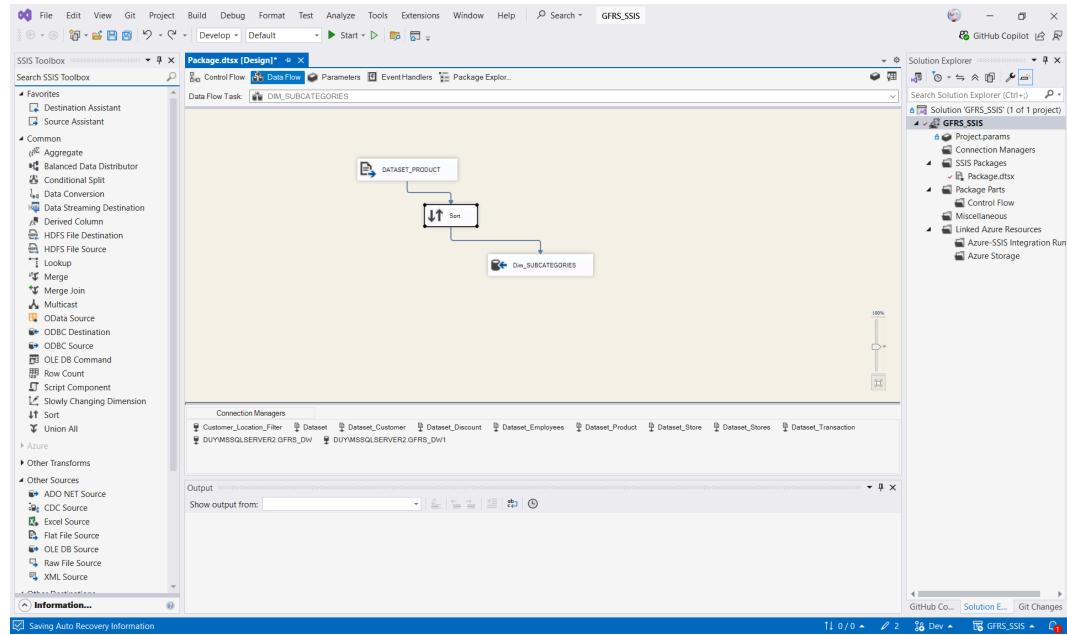
*Hình 121. Tạo chức năng Sort trong Dim\_SUBCATEGORIES*

- **Bước 7:** Nhấn đúp chuột vào **Sort** và màn hình **Sort Transformation Editor** xuất hiện. Chọn **Remove rows with duplicate sort values** để lọc các dữ liệu trùng lặp. Nhấn **OK** và thoát màn hình **Sort Transformation Editor**.



Hình 122. Cài đặt chức năng Sort trong Dim\_SUBCATEGORIES

- **Bước 8:** Kéo chức năng **ADO NET Destination** vào vùng **Data Flow** và đổi tên thành **Dim\_SUBCATEGORIES**. Kéo mũi tên luồng dữ liệu từ **Sort** vào **Dim\_SUBCATEGORIES**.



Hình 123. Tạo ADO.NET Destination trong Dim\_SUBCATEGORIES

- **Bước 9:** Vào SQL Server 2022, chạy lệnh DDL để tạo Dim\_SUBCATEGORIES

```

40 [Category] varchar(255)
41 )
42
43 select * from Dim_CATEGORIES;
44
45 drop table if exists Dim_SUBCATEGORIES
46 CREATE TABLE [Dim_SUBCATEGORIES] (
47     [SubCategoryID] integer identity(1,1) PRIMARY KEY,
48     [SubCategory] varchar(255)
49 )
50
51 drop table if exists Dim_DISCOUNTS
52 CREATE TABLE [Dim_DISCOUNTS] (

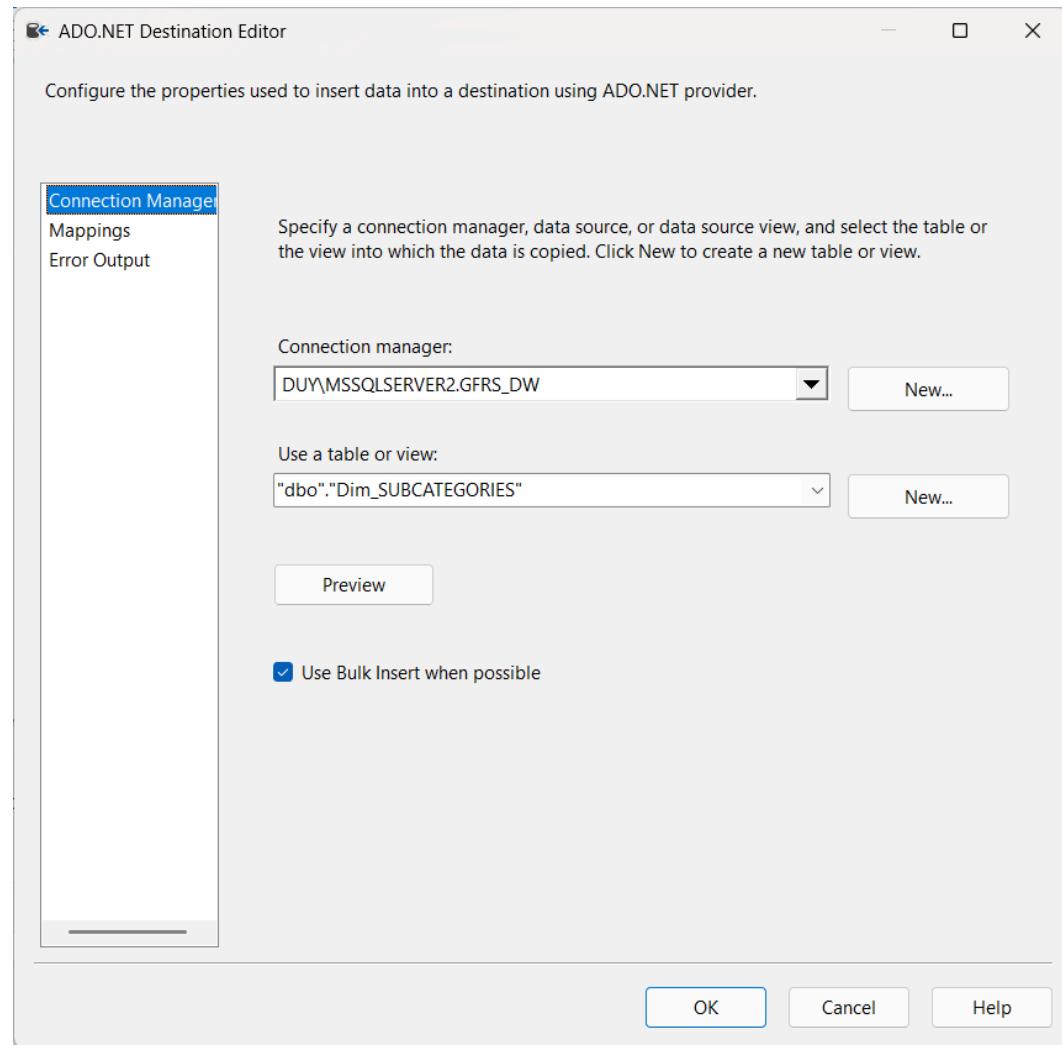
```

CategoryID	Category
1	Children
2	Feminine
3	Masculine

Hình 124. Chạy lệnh DDL tạo Dim\_SUBCATEGORIES

- **Bước 10 :**Nhấn đúp chuột vào Dim\_SUBCATEGORIES. Trong tab Connection Manager của màn hình ADO.NET Destination Editor, ở Use

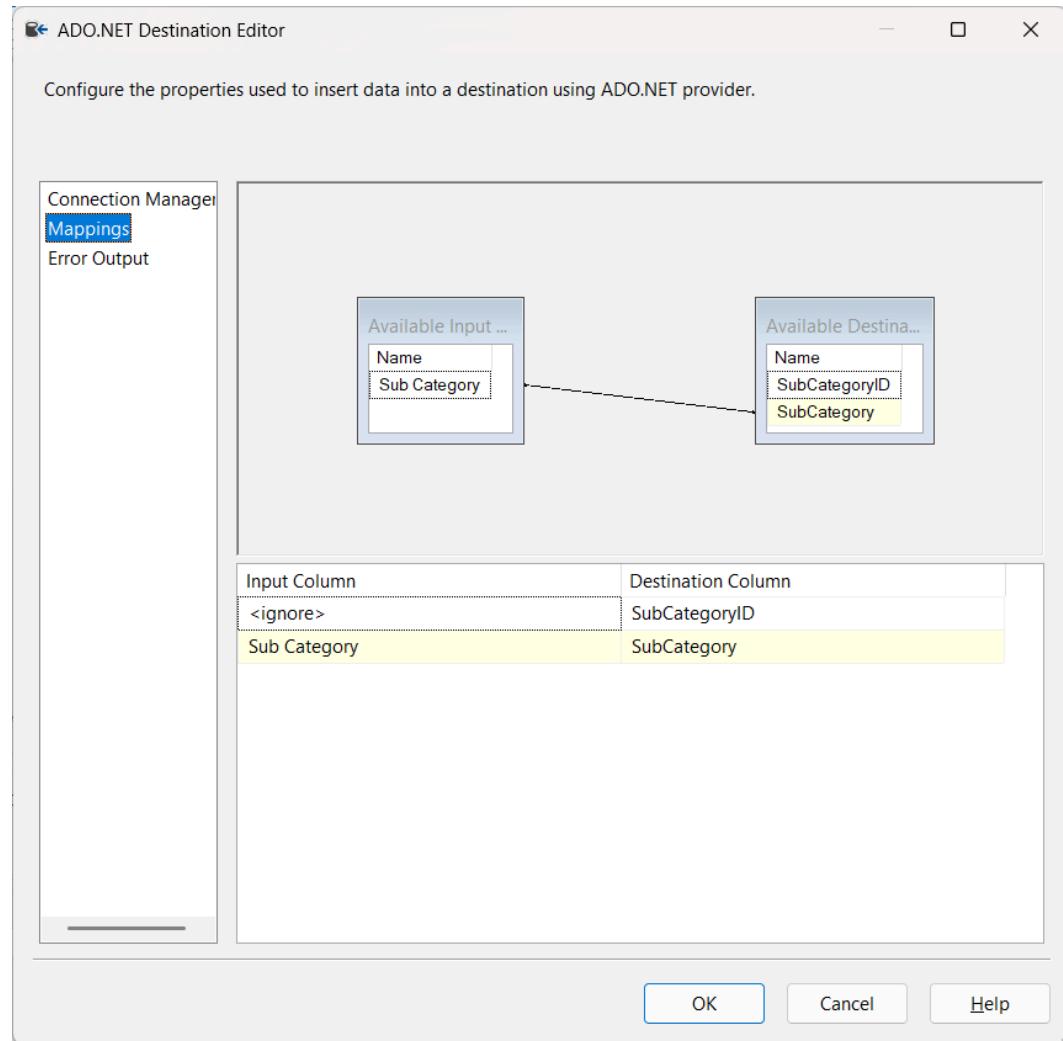
a table or view tiến hành nhấn chọn “**dbo”.”Dim\_SUBCATEGORIES**”,  
cũng chính là bảng vừa được tạo bằng lệnh DDL ở **SQL Server 2022**



Hình 125. Lựa chọn đích đến cho ADO NET Destination

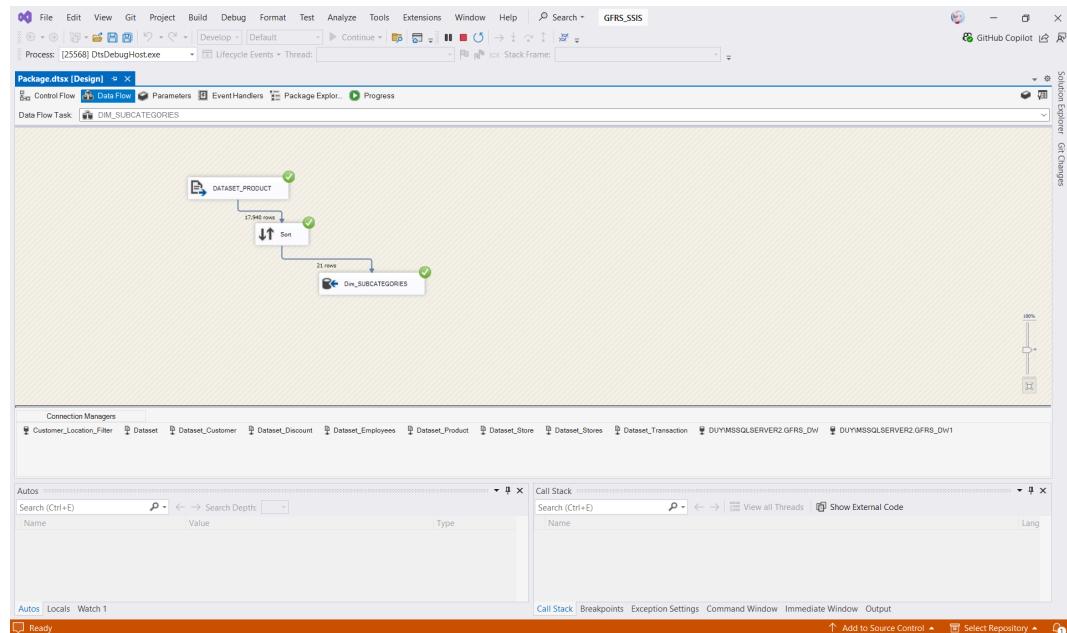
#### *Dim\_SUBCATEGORIES*

- **Bước 11:** Ở tab **Mapping** của màn hình **ADO.NET Destination Editor** tiến hành chọn các thuộc tính thích hợp ở cột bên trái để ánh xạ với các thuộc tính ở cột bên phải.



Hình 126. Cài đặt ADO NET Destination Dim\_SUBCATEGORIES

- **Bước 12:** Nhấn Start để kiểm tra quá trình tạo và đổ dữ liệu vào Dim\_SUBCATEGORIES có thành công hay chưa.



Hình 127. Đỗ dữ liệu vào Dim\_SUBCATEGORIES

Kết quả hiển thị cho thấy quá trình diễn ra thành công và có **21 rows** được đỗ vào **DIM\_SUBCATEGORIES**.

- **Bước 13:**Vào SQL Server 2022 kiểm tra và thấy dữ liệu đã được đỗ vào **Dim\_SUBCATEGORIES** thành công.

The screenshot shows the SQL Server Management Studio (SSMS). A query window titled "GFRS\_DW" contains the following T-SQL code:

```

[Category] varchar(255)
)
select * from Dim_CATEGORIES;

drop table if exists Dim_SUBCATEGORIES
CREATE TABLE [Dim_SUBCATEGORIES] (
    [SubCategoryID] integer identity(1,1) PRIMARY KEY,
    [SubCategory] varchar(255)
)
Select * from Dim_SUBCATEGORIES;
drop table if exists Dim_DISCOUNTS
CREATE TABLE [Dim_DISCOUNTS] (

```

The results pane shows the data inserted into the Dim\_SUBCATEGORIES table:

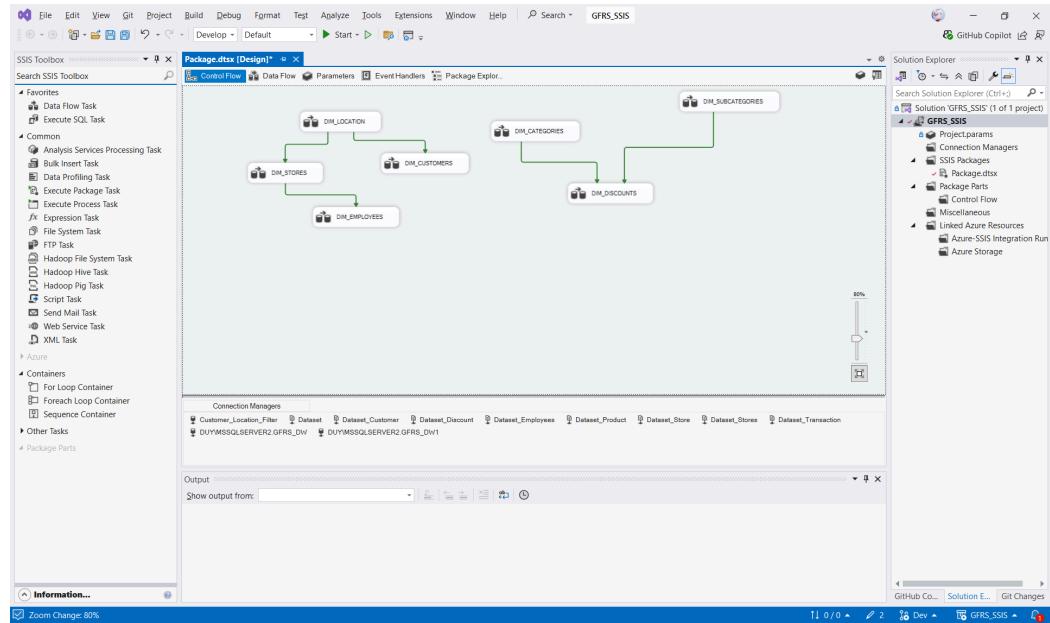
SubCategoryID	SubCategory
1	Accessories
2	Baby (0-12 months)
3	Coats
4	Coats and Blazers
5	Dresses and Jumpsuits
6	Girl and Boy (1-5 years, 6-14 years)
7	Lingerie and Pajamas
8	Pajamas
9	Pants and Jeans
10	Shirts
11	Shirts and Blouses
12	Skirts and Shorts
13	Sportswear
14	Suits and Blazers

A status bar at the bottom of the screen indicates: "Query executed successfully." and "DUY\SQLSERVER2 (16.0 RTM) | DUY\maing (85) | GFRS\_DW | 00:00:00 | 21 rows".

Hình 128. Dữ liệu được đỗ thành công vào Dim\_SUBCATEGORIES

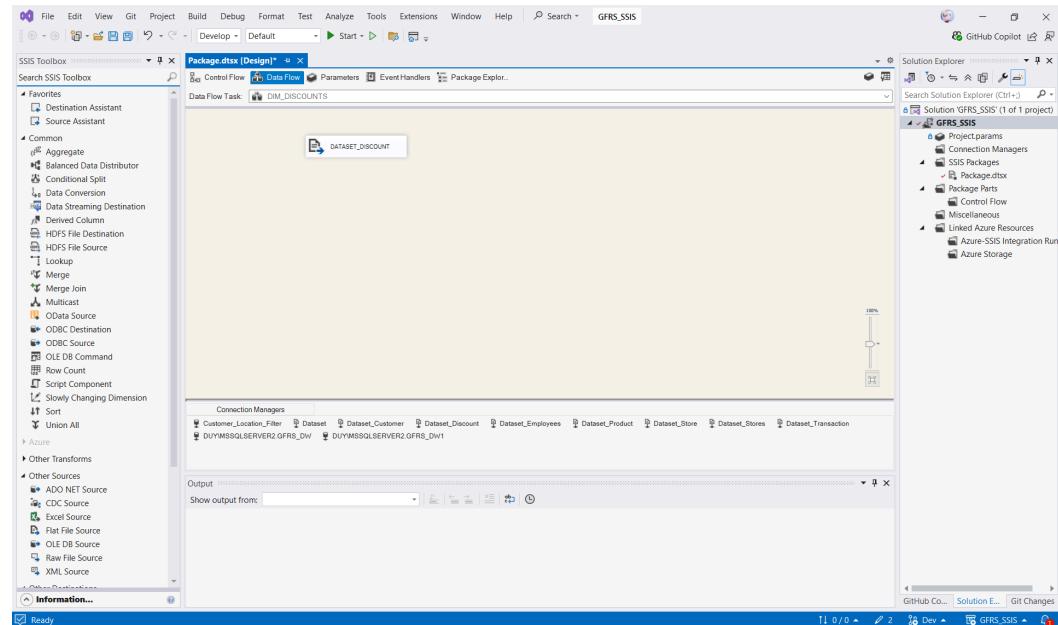
### II.3.6. Dim\_DISCOUNTS

- **Bước 1:** Kéo chức năng Data Flow Task trong vùng SSIS Toolbox vào vùng Control Flow và đổi tên thành **Dim\_DISCOUNTS**.



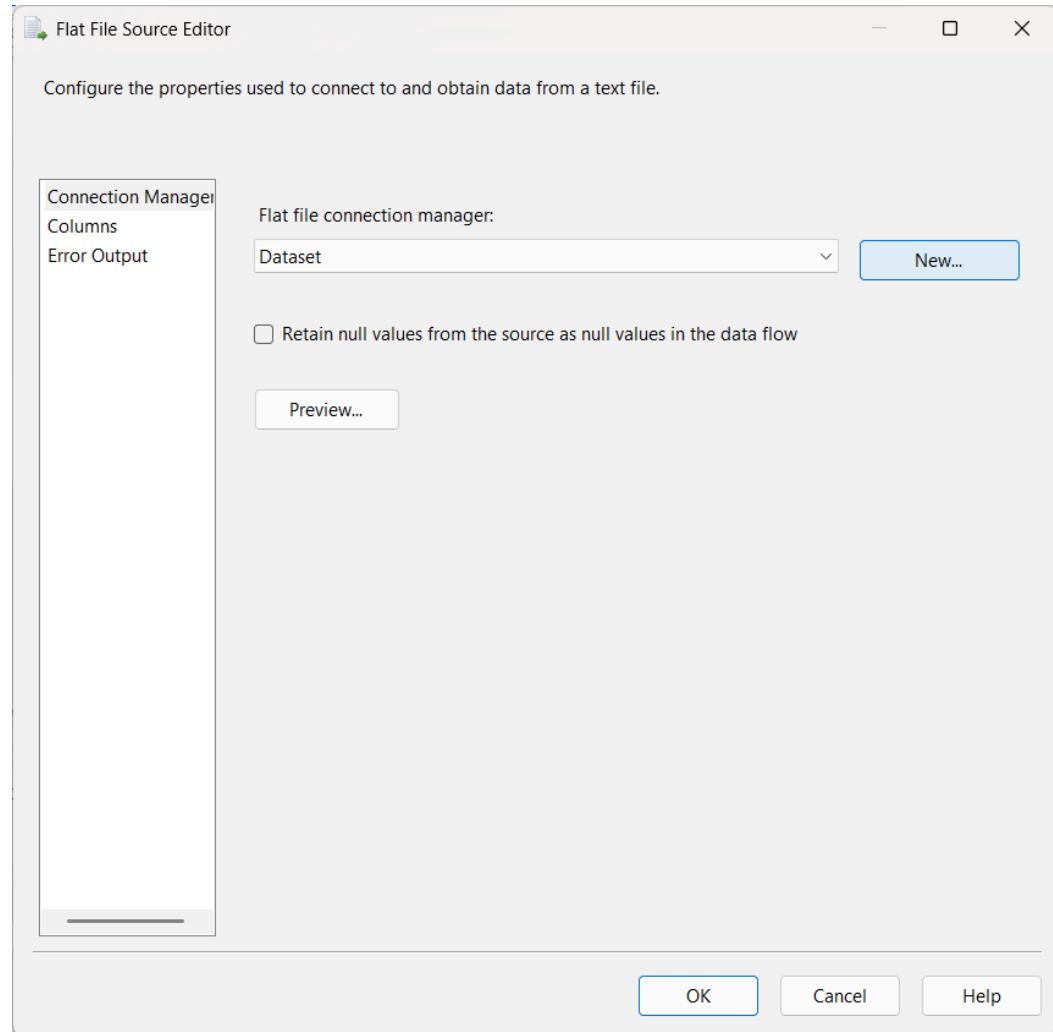
Hình 129. Tạo chức năng Dim\_DISCOUNTS

- **Bước 2:** Nhấn đúp chuột vào **Dim\_DISCOUNTS**. Sau đó kéo chức năng Flat File Source vào vùng làm việc và đổi tên thành **DATASET\_DISCOUNT**



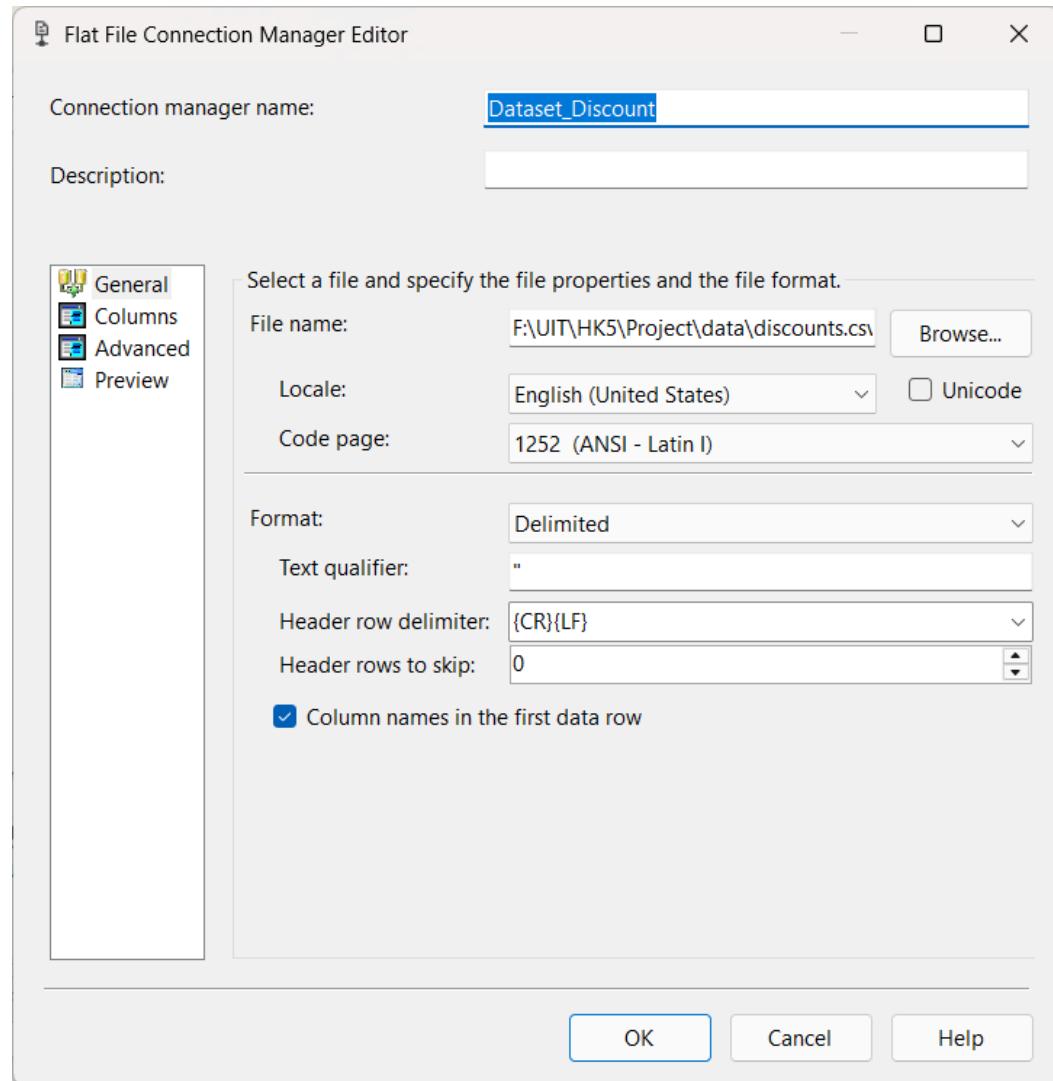
*Hình 130. Tạo chức năng DATASET\_DISCOUNT*

- **Bước 3:** Nhấn đúp chuột vào DATASET\_DISCOUNT. Màn hình Flat File Source Editor hiện ra. Ở tab Connection Manager nhấn New



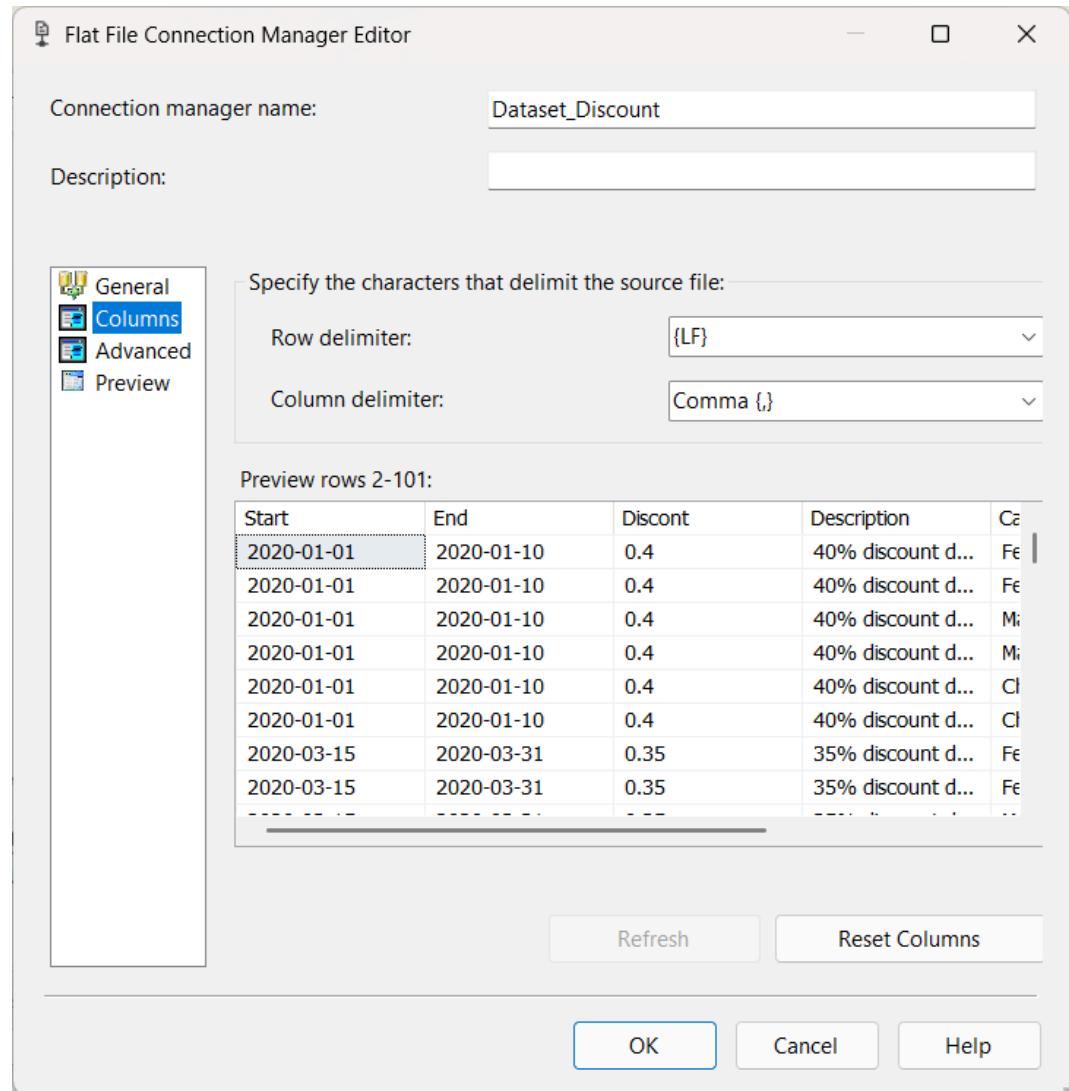
Hình 131. Tạo nguồn dữ liệu cho DATASET\_DISCOUNT

- **Bước 4:** Màn hình **Flat File Connection Manager Editor** hiện ra. Ở tab **General**, nhấn **Browse** để tiến hành load file csv **employees.csv** trong File **Global Fashion Retail Sales**. Ở mục **Connection manager name**, đặt tên là **Dataset\_Discount**. Ở mục **Text qualifier**, nhập ký tự dấu nháy kép “



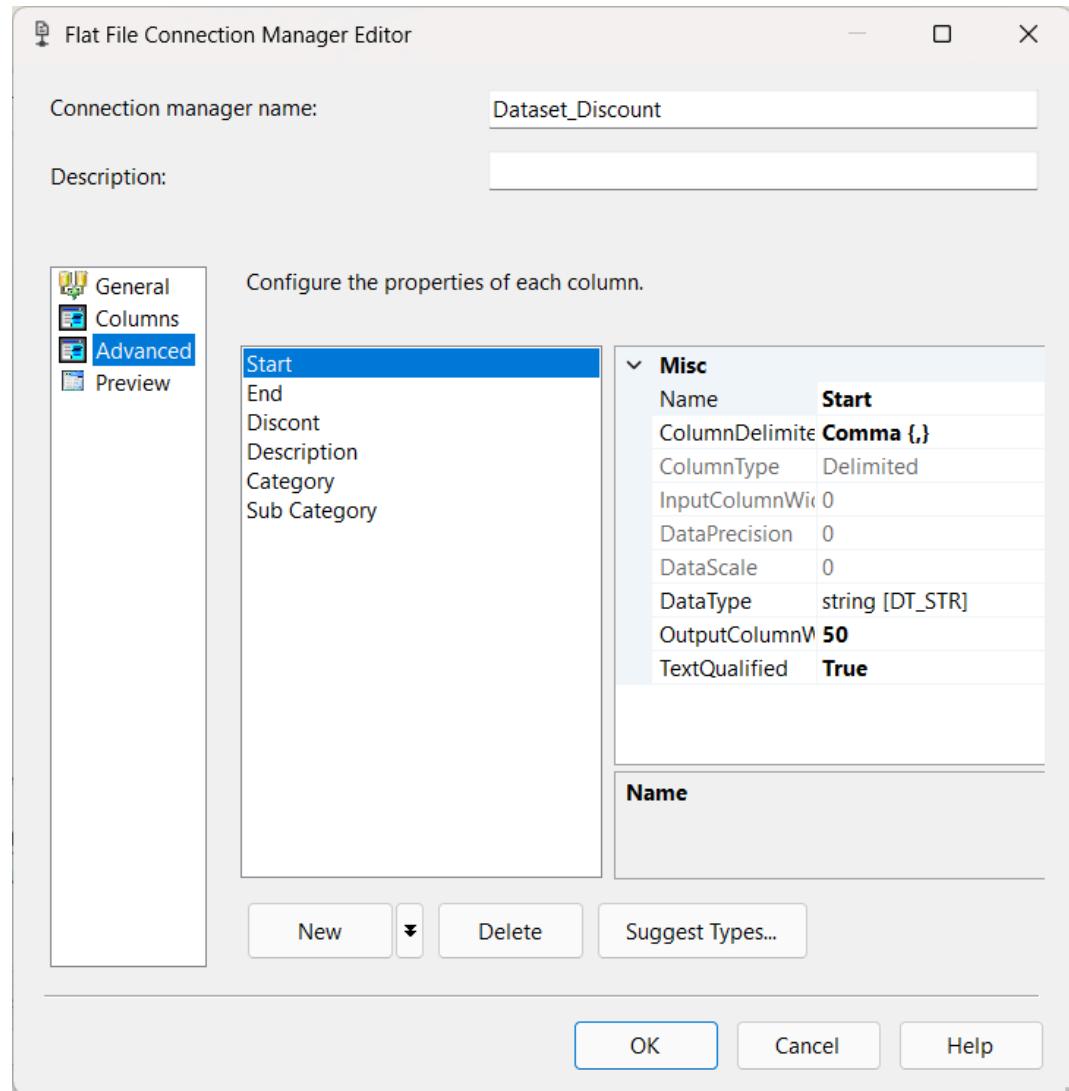
Hình 132. Tạo nguồn dữ liệu Dataset\_Discount

- **Bước 5:** Ở tab **Columns**, ta có thể kiểm tra và thấy rằng file csv đã được load thành công



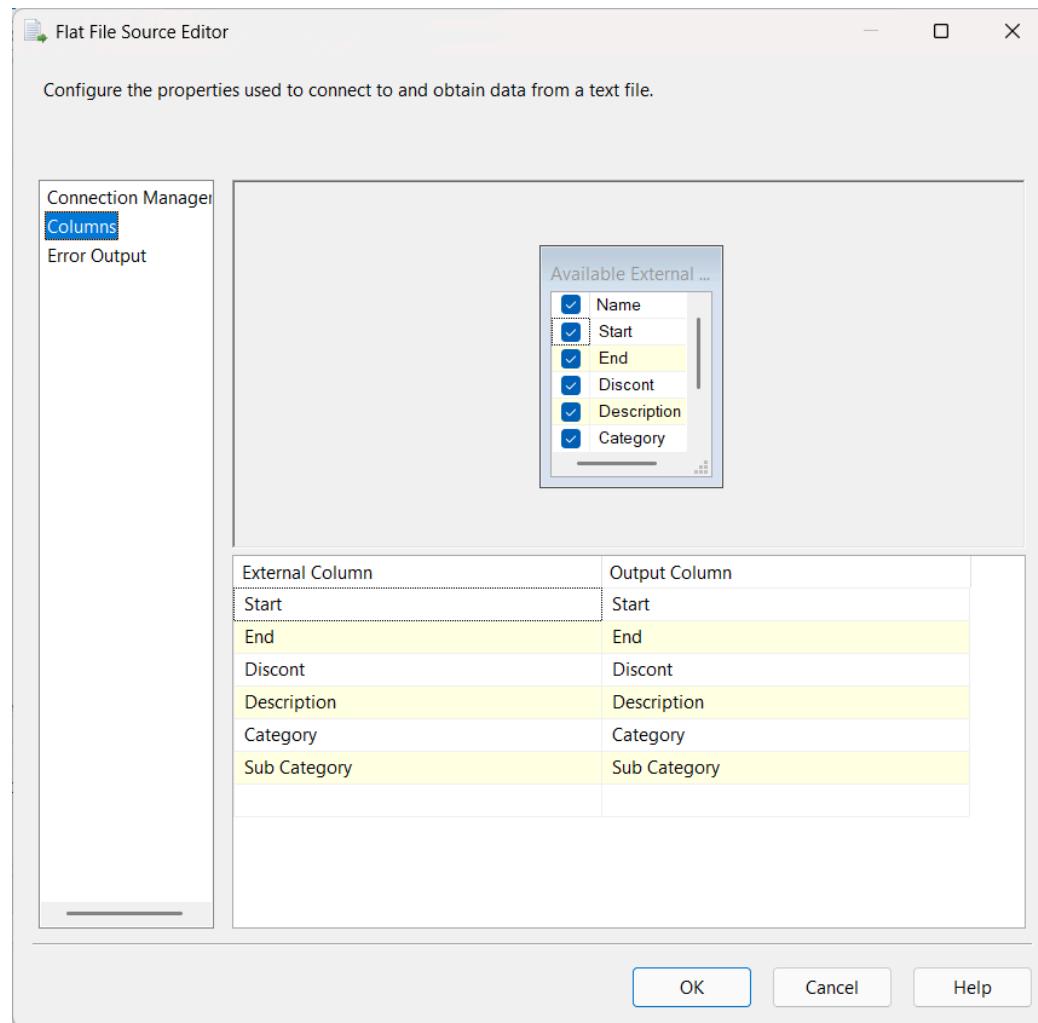
Hình 133. Kiểm tra dữ liệu Dataset\_Discount trước khi nạp

- **Bước 6:** Ở tab **Advanced**, tiến hành thay đổi kiểu dữ liệu của các thuộc tính cần sử dụng trong quá trình **SSIS** của file csv tương ứng với kiểu dữ liệu khai báo của thuộc tính đó trong **SQL Server**. Sau khi thay đổi kiểu dữ liệu của các thuộc tính cần sử dụng trong quá trình **SSIS** thì nhấn **OK** và sẽ thoát khỏi màn hình **Flat File Connection Manager Editor**.



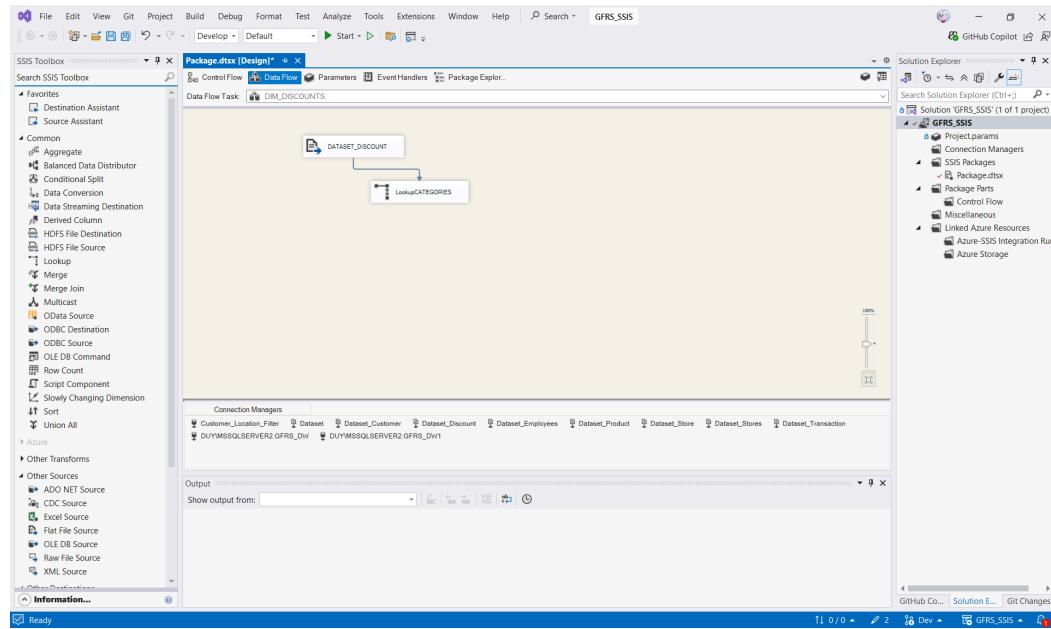
Hình 134. Lựa chọn kiểu dữ liệu phù hợp cho các thuộc tính trong Dataset\_Discount

- **Bước 7:** Ở màn hình **Flat File Source Editor**, qua tab **Columns** và nhấn chọn các thuộc tính cần sử dụng trong quá trình tạo **Dim\_DISCOUNT**



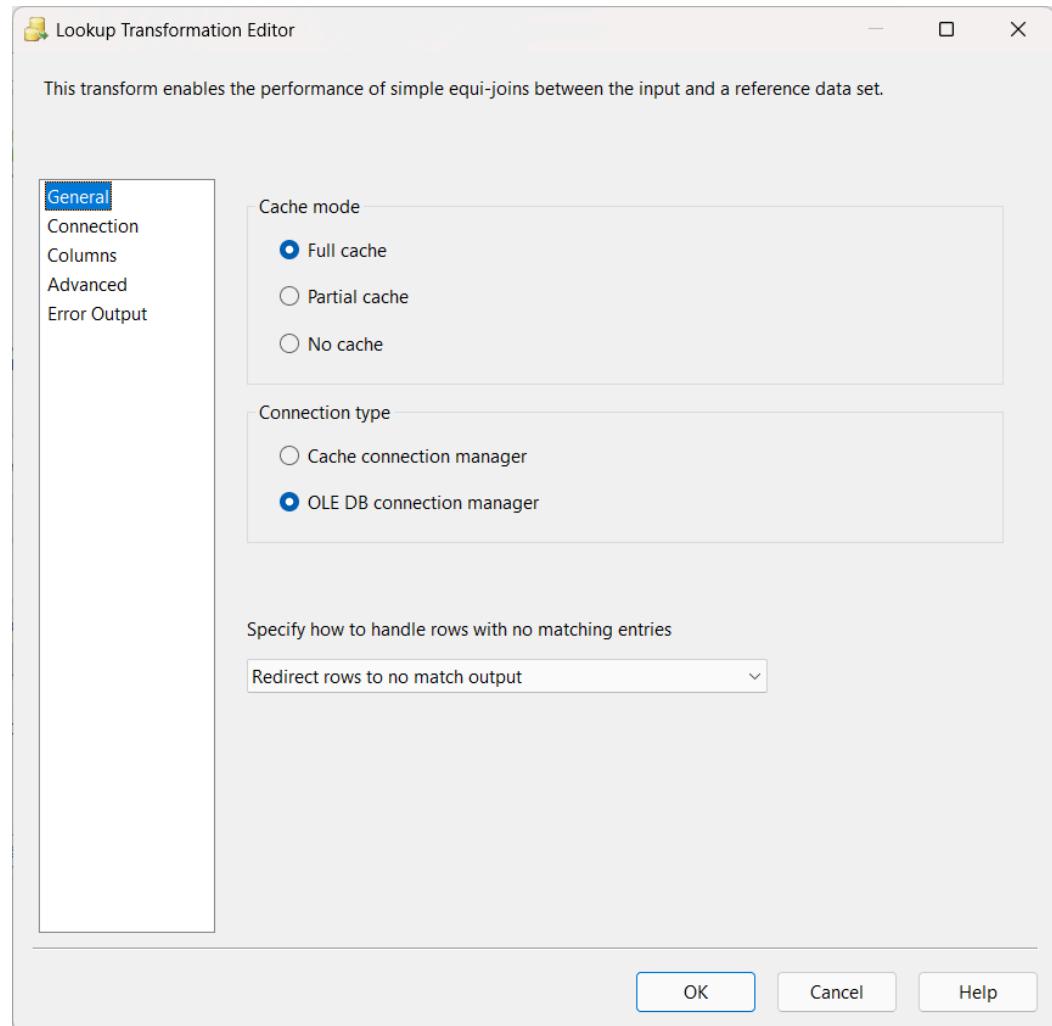
Hình 135. Lựa chọn thuộc tính cần thiết cho quá trình tạo  
Dim\_DISCOUNTS

- **Bước 8:** Nhấn OK để thoát màn hình **Flat File Source Editor**
- **Bước 9:** Kéo chức năng **Lookup** vào vùng **Data Flow** và đổi tên thành **LookupCATEGORIES**. Kéo mũi tên luồng dữ liệu từ **DATASET\_DISCOUNT** vào **LookupCATEGORIES**.



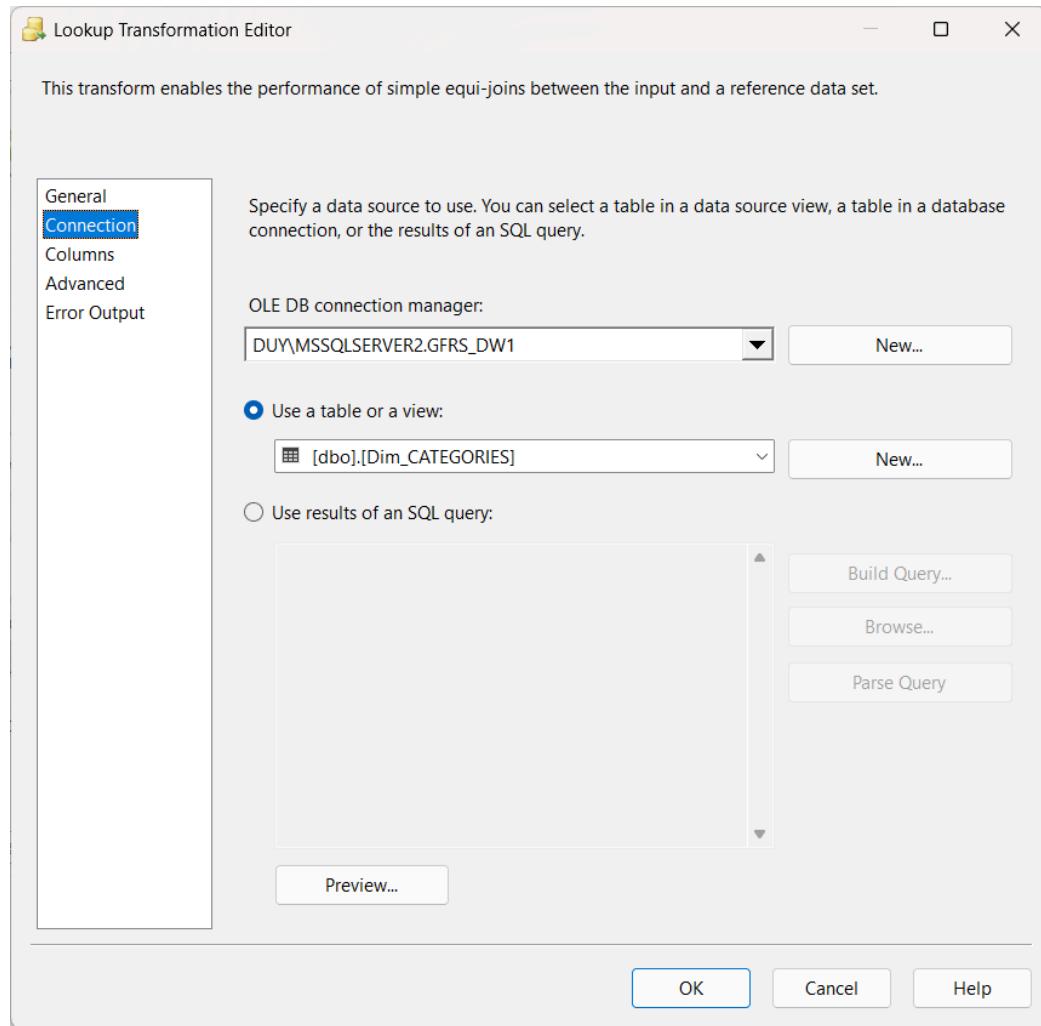
Hình 136. Tạo LookupCATEGORIES trong Dim\_DISCOUNTS

**Bước 10:** Nhấn đúp chuột vào **LookupCATEGORIES**, màn hình **Lookup Transformation Editor** xuất hiện. Ở tab **General**, phần **Specify how to handle rows with no matching entries** ta chọn **Redirect rows to no match output** (giúp pipeline không bị fail, và cho phép ta tự định nghĩa cách xử lý dữ liệu không khớp) để phân luồng Discount chỉ áp dụng cho những sản phẩm nhất định hay cho tất cả các sản phẩm (phần Category và Sub Category của dữ liệu sẽ được bỏ trống).



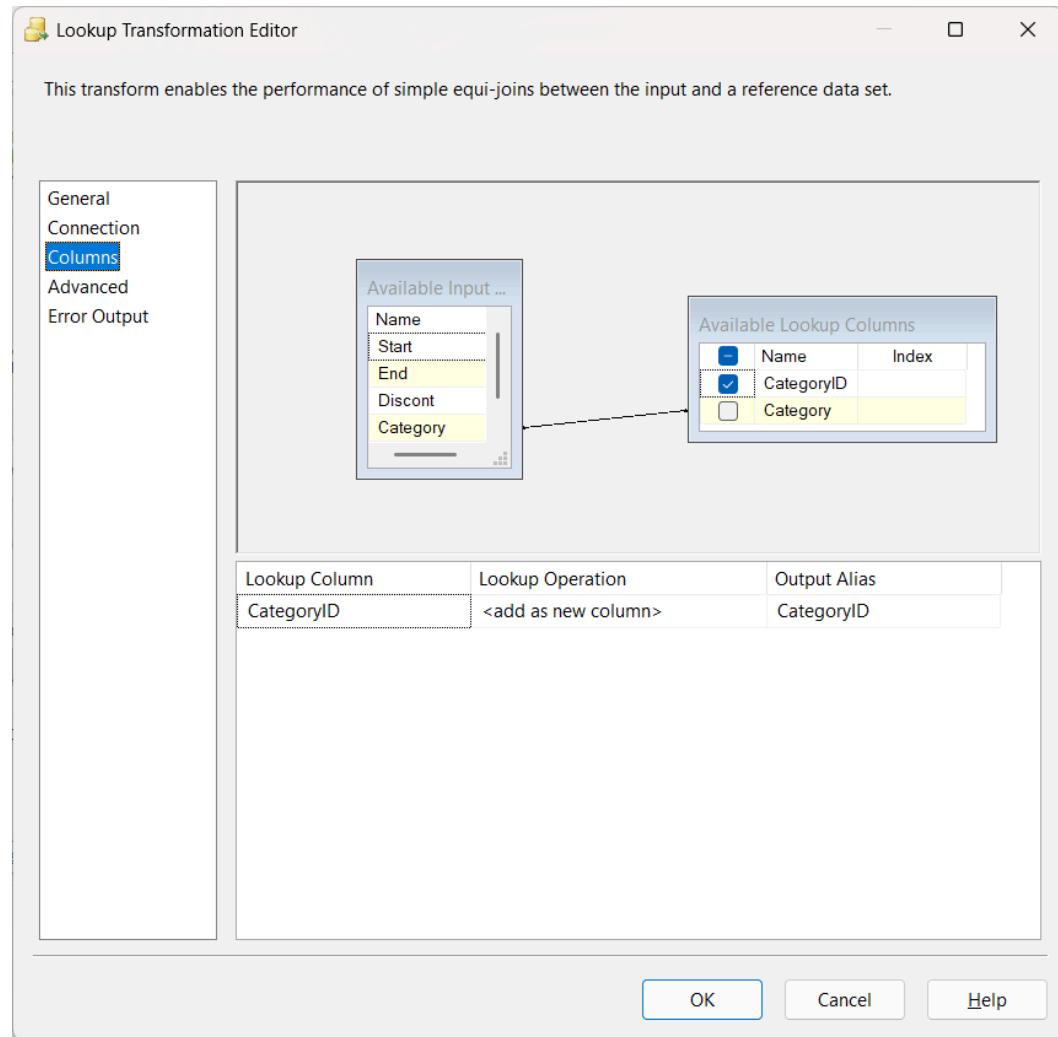
Hình 137. Cài đặt LookupCATEGORIES ở tab General

- **Bước 11:** Ở tab Connection, phần **Use a table or a view**, tiến hành chọn **[dbo].[Dim\_CATEGORIES]** (do chúng ta đã thiết lập **OLE DB connection manager** ở bước trước, nên ta có thể sử dụng lại)



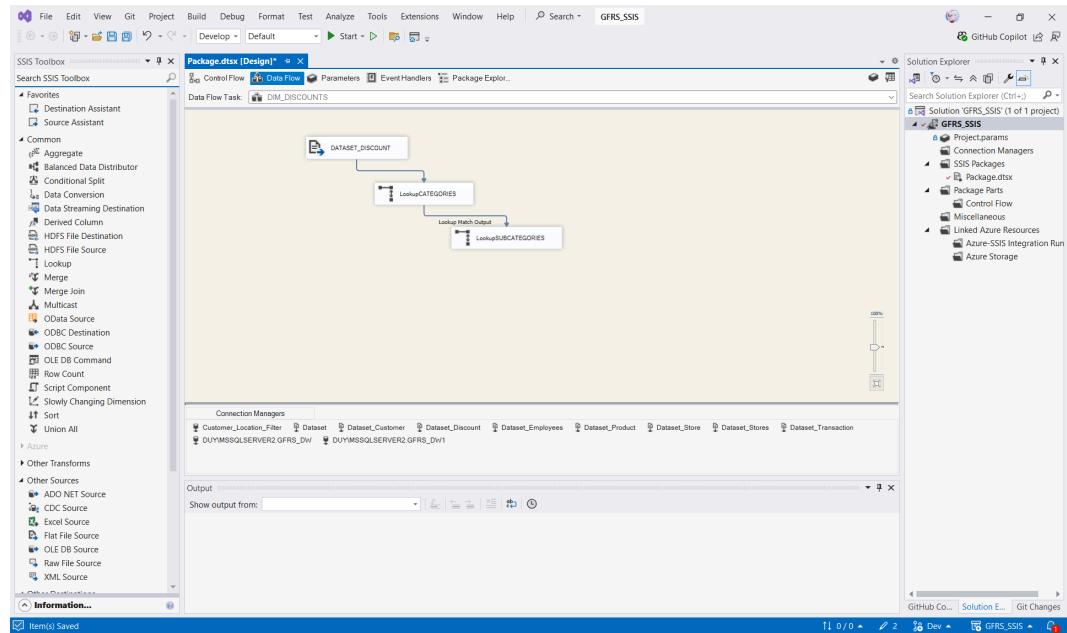
Hình 138. Kết nối nguồn dữ liệu cho LookupCATEGORIES

- **Bước 12:** Trong tab **Columns**, ở bảng bên phải click chọn **CategoryID**. Ta kéo **Category** từ bảng **Available Input Columns** qua tương ứng **Category** bảng **Available Lookup Columns** (quá trình này giúp hệ thống xác định vùng dữ liệu cần tìm kiếm). Sau đó bấm **OK** để đóng cửa sổ **Lookup Transformation Editor**



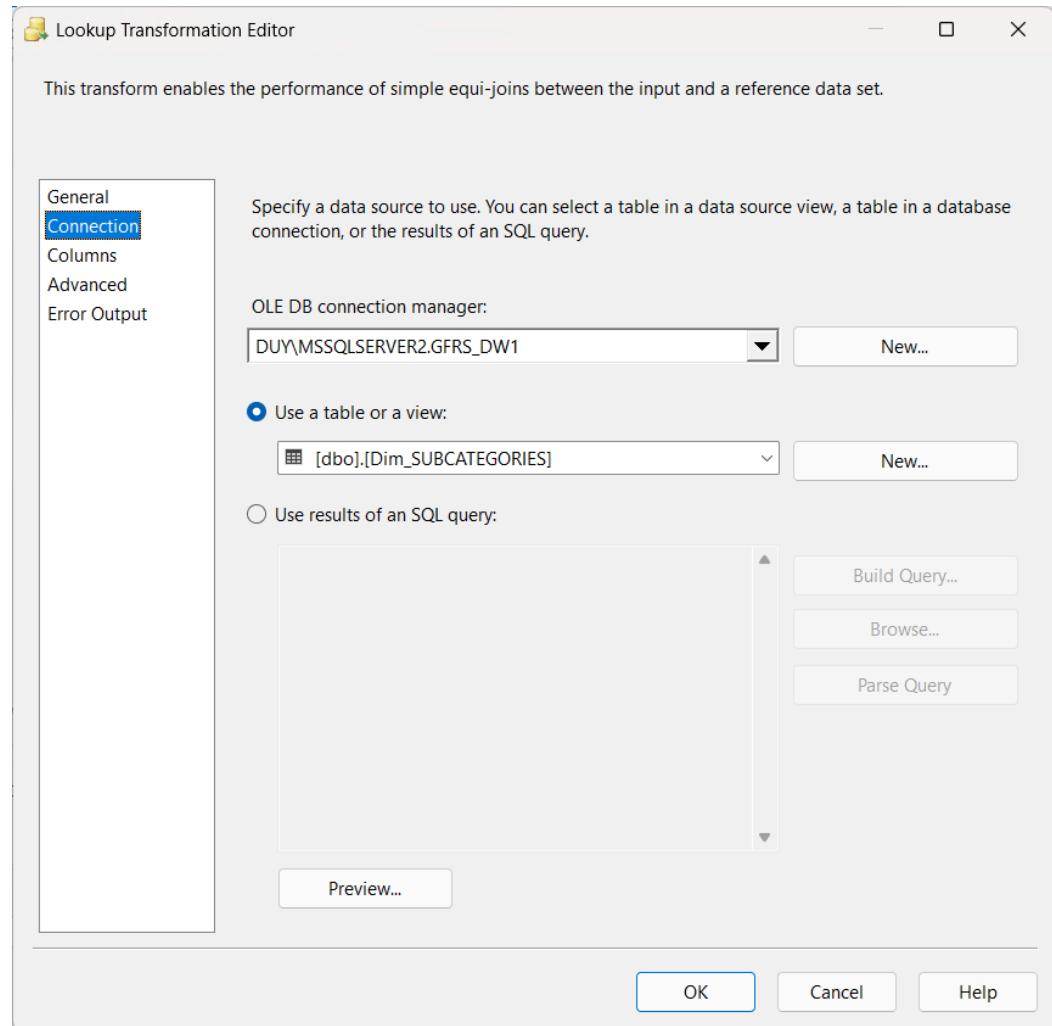
Hình 139. Cài đặt LoopkupCATEGORIES trong Dim\_DISCOUNTS

- **Bước 13:** Kéo chức năng **Lookup** vào vùng **Data Flow** và đổi tên thành **LookupSUBCATEGORIES**. Kéo mũi tên luồng dữ liệu từ **LookupCATEGORIES** vào **LookupSUBCATEGORIES**. Một hộp thoại sẽ xuất hiện, chọn **Lookup Match Output** và nhấn **OK**.



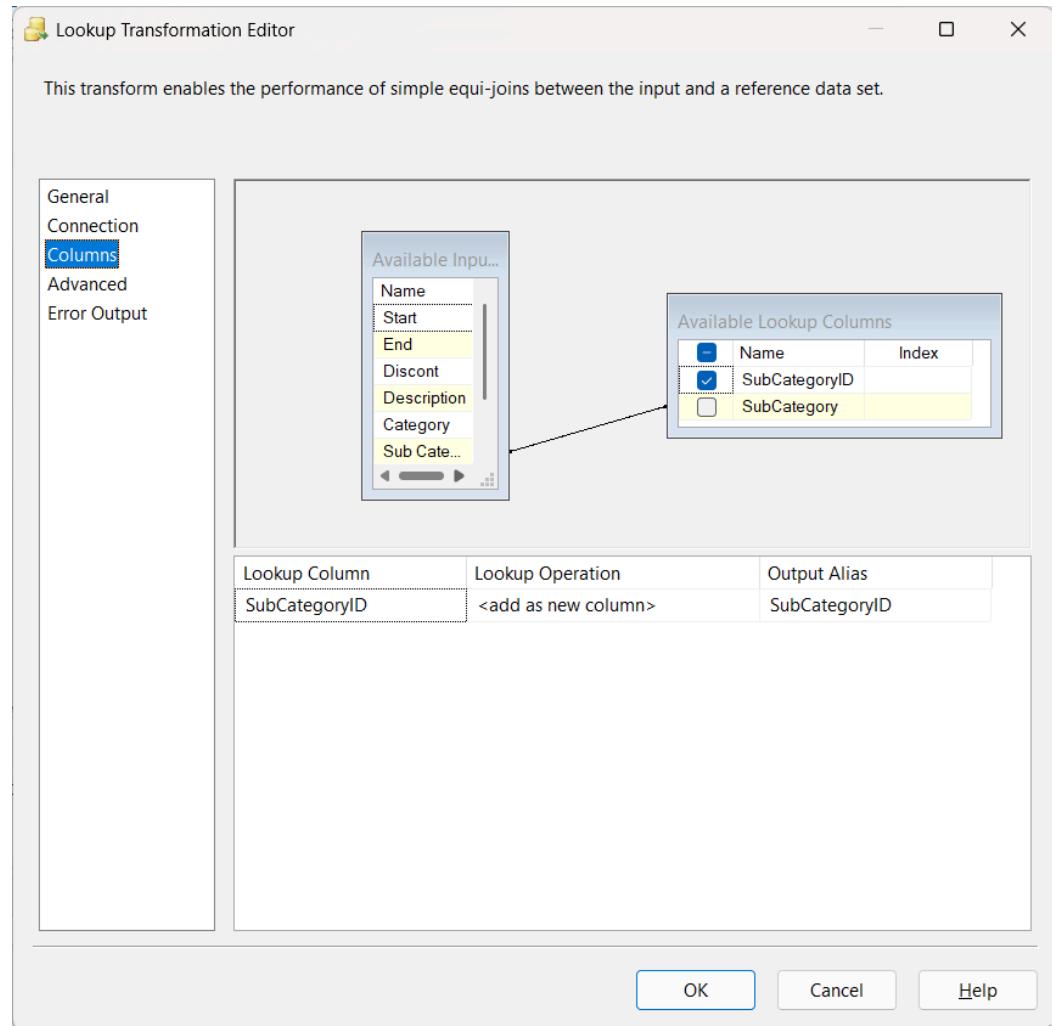
*Hình 140. Tạo chức năng LookupSUBCATEGORIES trong Dim\_DISCOUNTS*

- **Bước 14:** Nhấn đúp chuột vào **LookupSUBCATEGORIES**, màn hình **Lookup Transformation Editor** xuất hiện. Do chúng ta đã thiết lập **OLE DB connection manager** ở bước trước, nên ta có thể sử dụng lại. Ở phần **Use a table or a view**, tiến hành chọn **[dbo].[Dim\_SUBCATEGORIES]**.



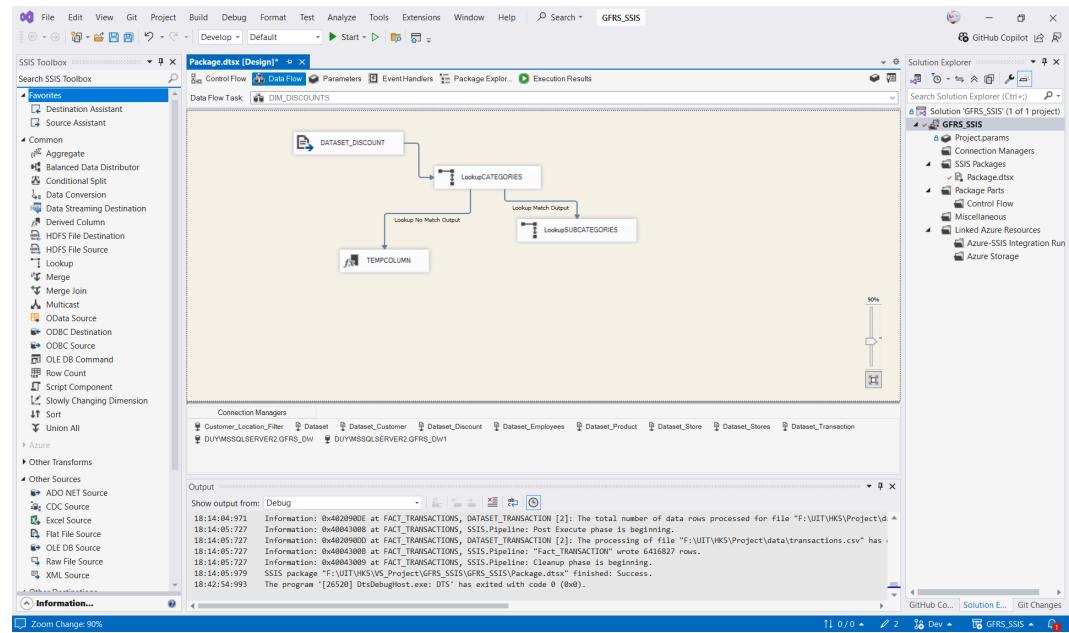
Hình 141. Kết nối nguồn dữ liệu cho LookupSUBCATEGORIES

- **Bước 15:** Trong tab **Columns**, ở bảng bên phải click chọn **SubCategoryID**. Ta kéo **Sub Category** từ bảng **Available Input Columns** qua tương ứng **SubCategory** bảng **Available Lookup Columns** (quá trình này giúp hệ thống xác định vùng dữ liệu cần tìm kiếm). Sau đó bấm **OK** để đóng cửa sổ **Lookup Transformation Editor**



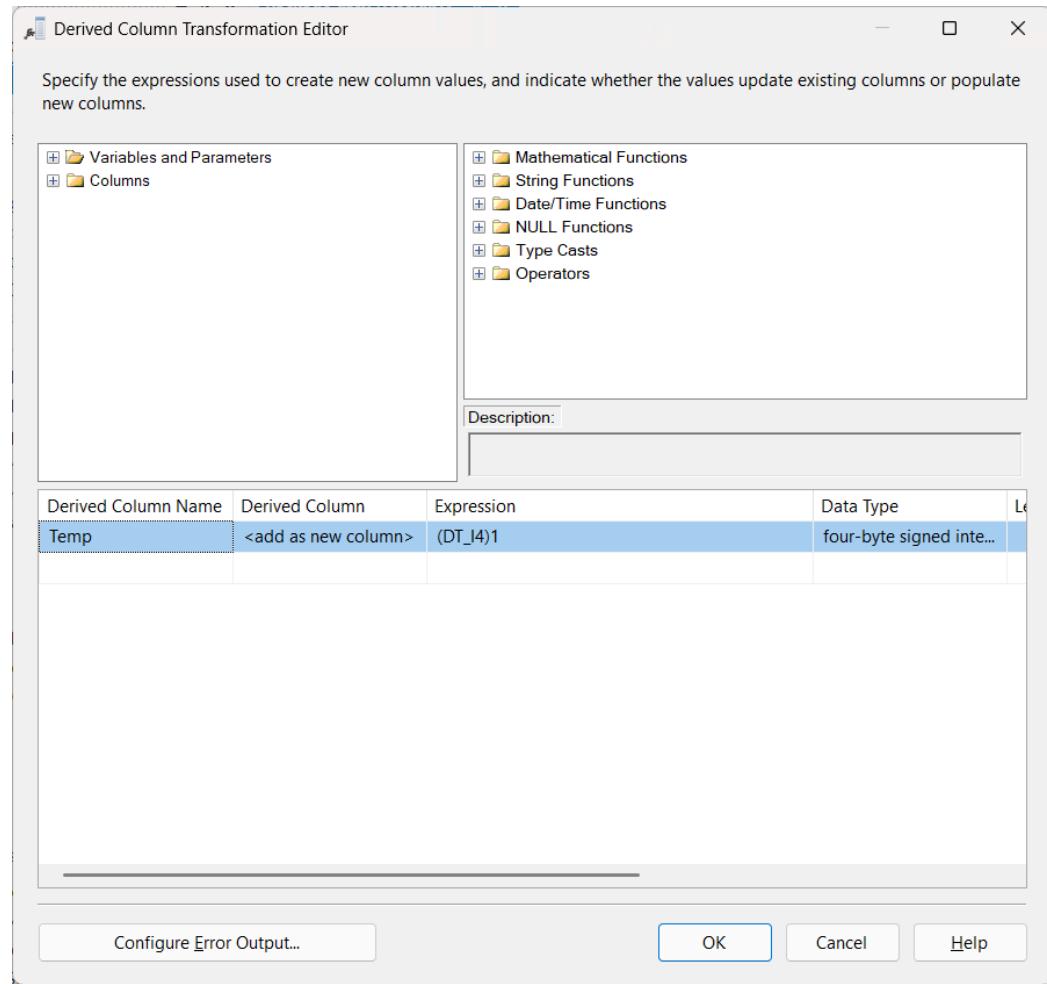
Hình 142. Cài đặt *LookupSUBCATEGORIES* trong *Dim\_DISCOUNTS*

- **Bước 16:** Ta kéo chức năng **Derived Column** (giúp biến đổi, chuẩn hóa hoặc tạo thêm cột mới từ dữ liệu gốc trước khi nạp vào đích) vào vùng **Data Flow** và đổi tên thành **TEMPCOLUMN**. Kéo luồng dữ liệu từ **LookupCATEGORIES** vào **TEMPCOLUMN** (hệ thống tự đặt luồng này là **Lookup No Match Output**) để tạo một cột giả



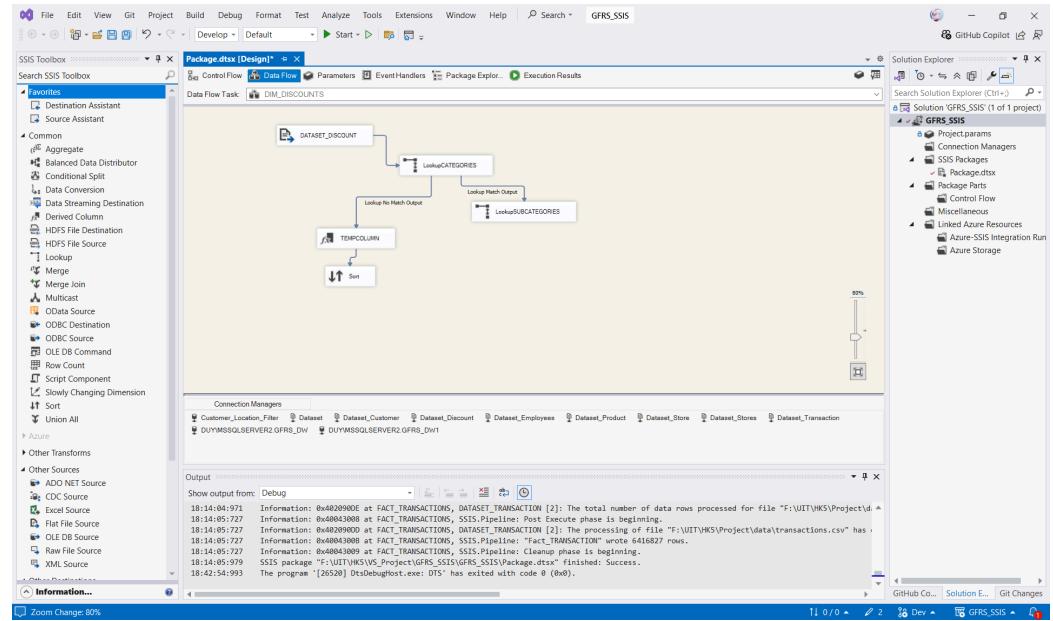
Hình 143. Tạo chức năng Derived Column trong Dim\_DISCOUNTS

- **Bước 17:** Nhấn đúp chuột vào **TEMPCOLUMN** của sổ **Derived Column Transformation Editor** sẽ được hiện lên. Ở ô **Expression** ta gõ (**DT\_I4**) 1 và đổi tên cột ở phần **Derived Column Name** thành Temp để tạo một cột mới với kiểu dữ liệu là **four-byte signed integer** với giá trị mặc định là 1. Nhấn **OK** để thoát cửa sổ **Derived Column Transformation Editor**



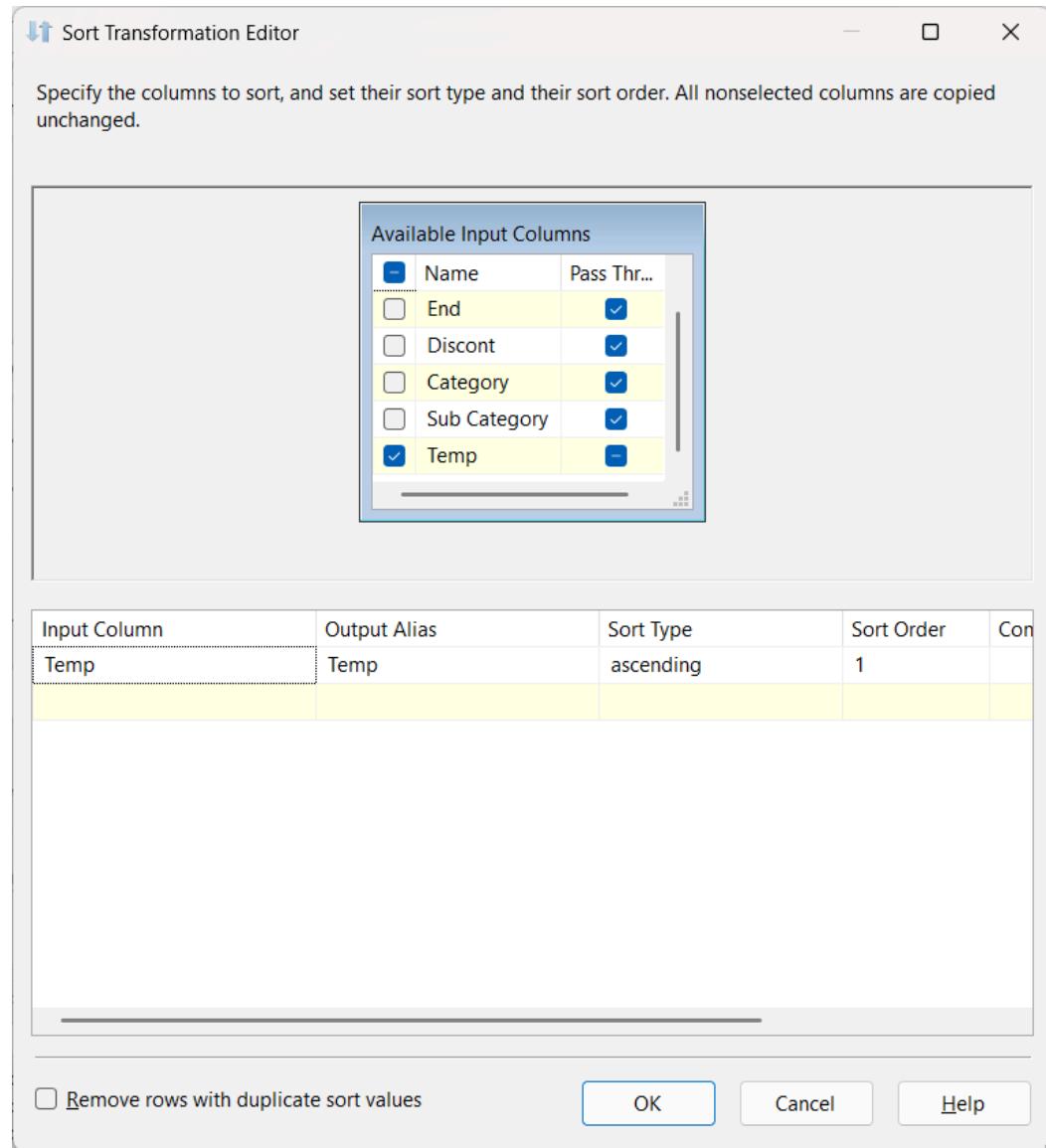
Hình 144. Cài đặt TEMP COLUMN trong Dim\_DISCOUNTS

- **Bước 18:** Kéo chức năng Sort vào vùng Data Flow và kéo mũi tên luồng dữ liệu từ TEMPCOLUMN vào Sort.



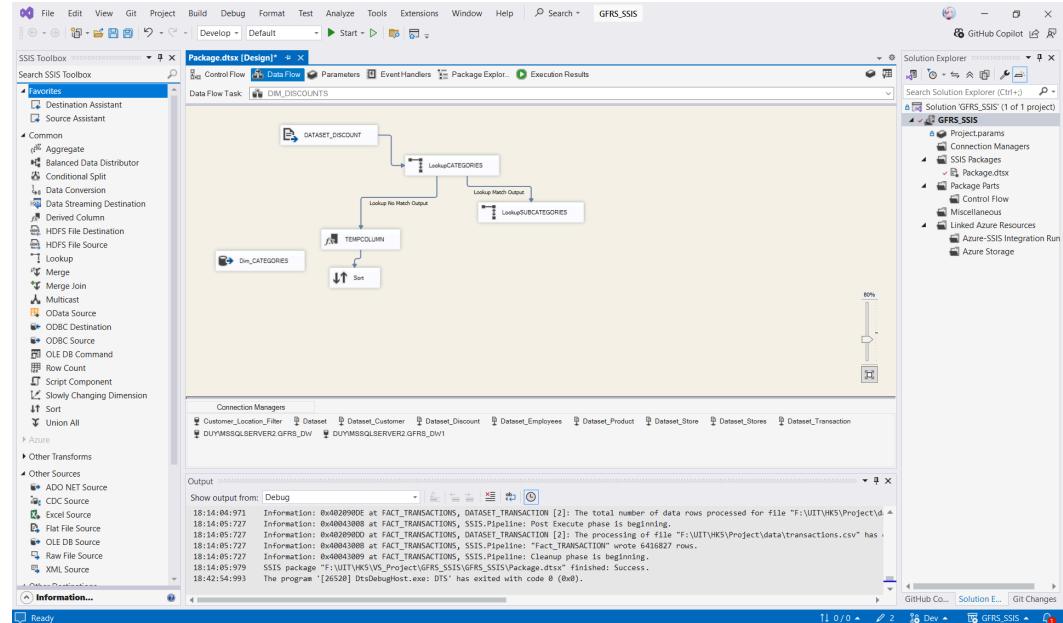
*Hình 145. Tạo chìc năng Sort trong Dim\_DISCOUNTS*

- **Bước 19:** Nhấn đúp chuột vào **Sort** và màn hình **Sort Transformation Editor** xuất hiện. Chọn cột **Temp**. Nhấn **OK** và thoát màn hình **Sort Transformation Editor**.



Hình 146. Cài đặt Sort trong Dim\_DISCOUNTS

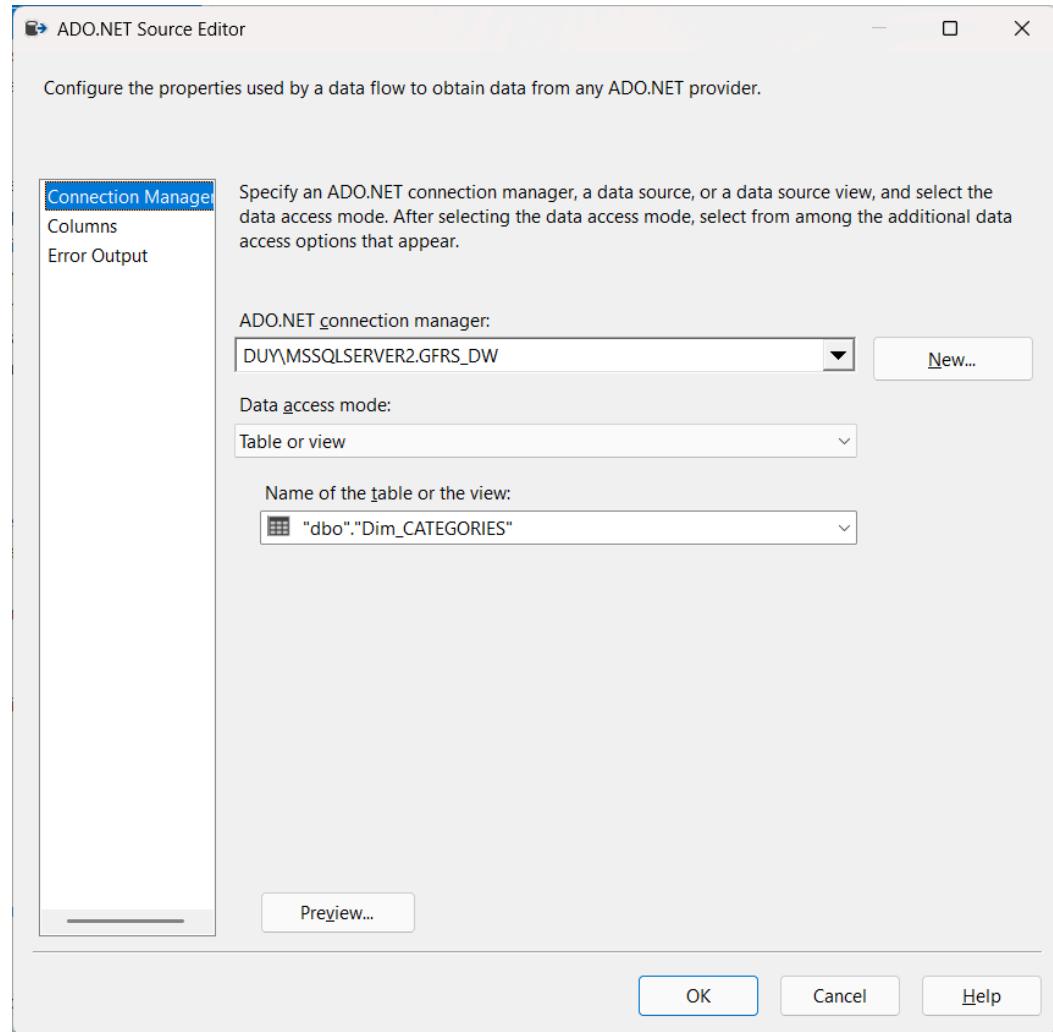
- **Bước 20:** Kéo chức năng ADO NET Source vào vùng Data Flow và đổi tên thành **Dim\_CATEGORIES** (lấy các danh mục sản phẩm để nối với các Discount trong Category)



Hình 147. Tạo ADO.NET Source Dim\_CATEGORIES trong

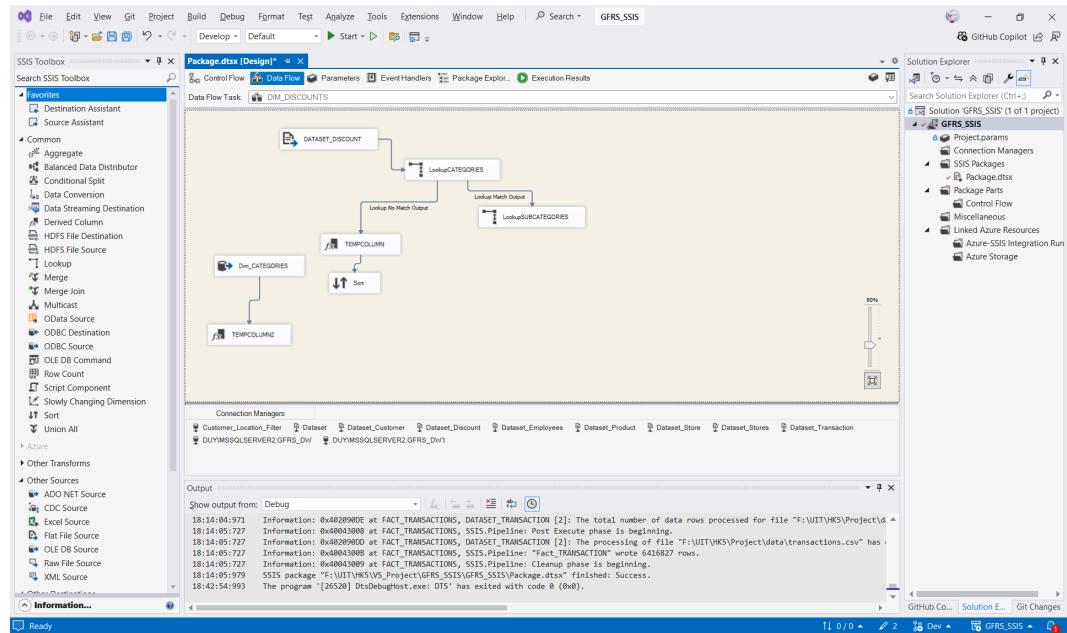
### Dim\_Discounts

- **Bước 21:** Nhấn đúp chuột vào **Dim\_CATEGORIES** của sổ **ADO.NET Source Editor** sẽ hiện lên, ở phần Name of the table or the view ta chọn “**dbo”.”Dim\_CATEGORIES**”. Nhấn OK để đóng cửa sổ **ADO.NET Source Editor**



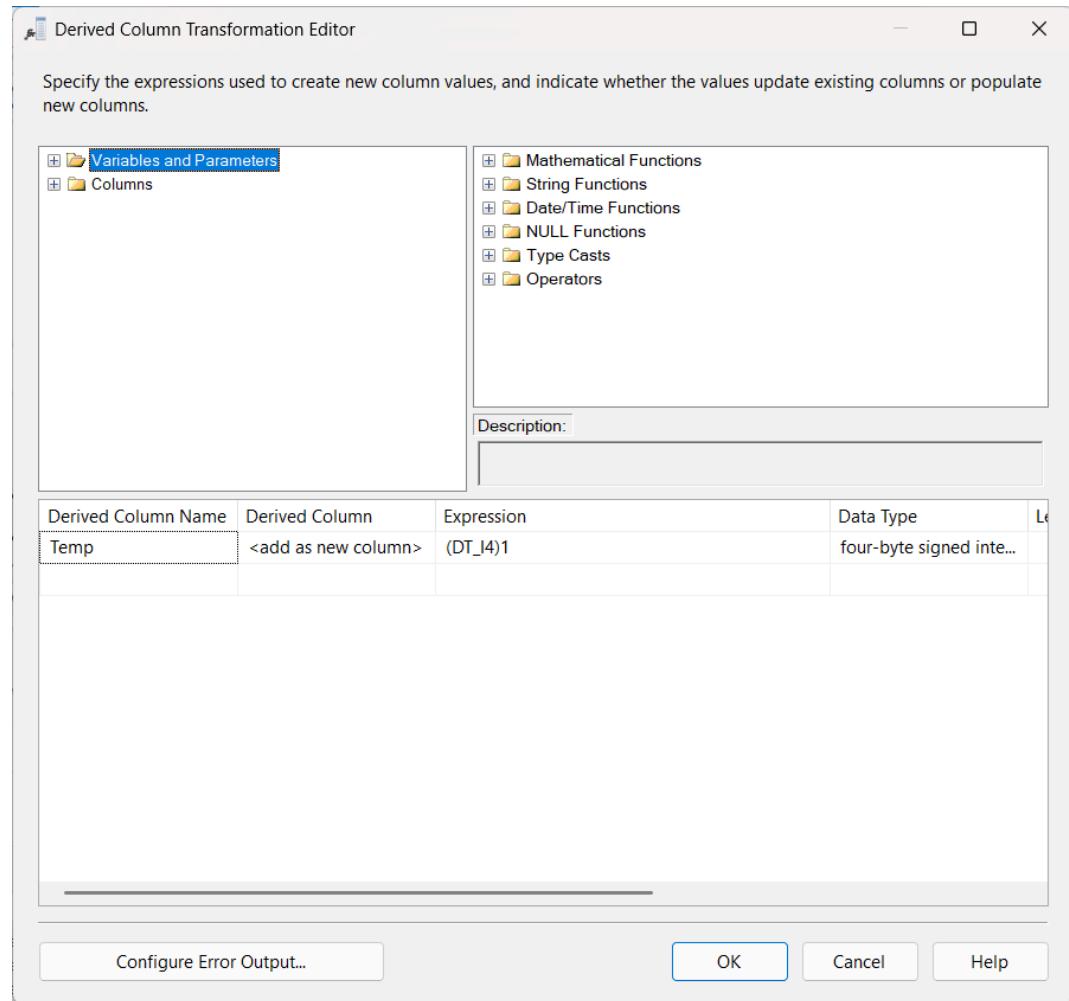
Hình 148. Chọn nguồn dữ liệu cho ADO NET Source Dim\_CATEGORIES

- **Bước 22:** Ta kéo chức năng **Derived Column** vào vùng **Data Flow** và đổi tên thành **TEMPCOLUMN2**. Kéo luồng dữ liệu từ **Dim\_CATEGORIES** vào **TEMPCOLUMN2** để tạo một cột giả (sau đó cho **Sort** cột này ở cả 2 nguồn dữ liệu để **Merge Join** nhận diện rằng nguồn dữ liệu nay đã được sắp xếp giúp quá trình **Merge Join** thuận tiện hơn)



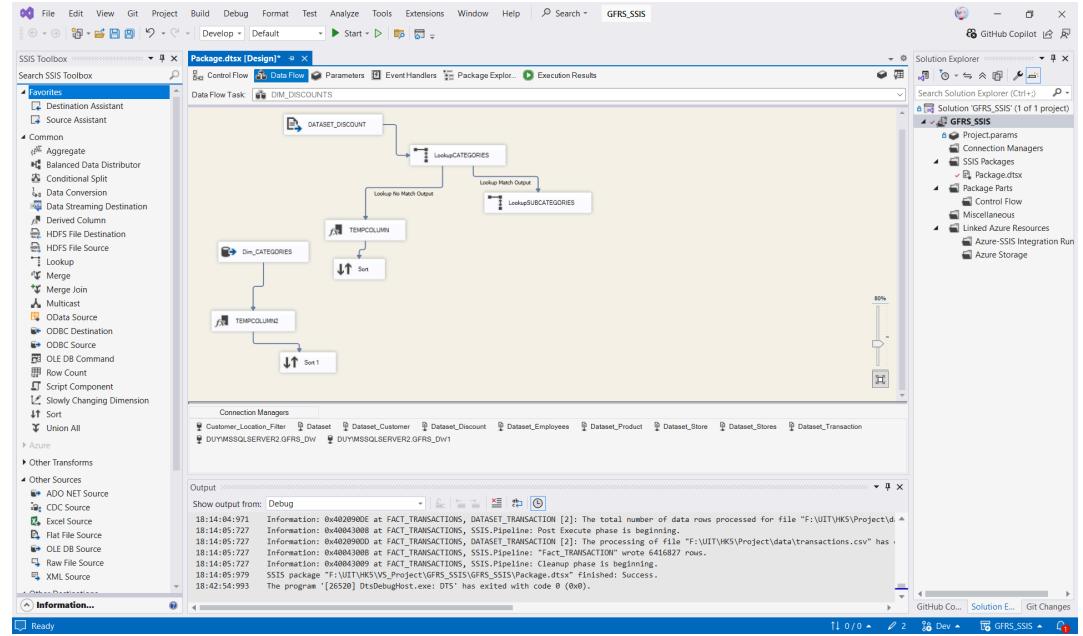
Hình 149. Tạo chức năng Derived Column thứ 2 trong Dim\_DISCOUNTS

- **Bước 23:** Nhấn đúp chuột vào **TEMPCOLUMN2** của sổ **Derived Column Transformation Editor** sẽ được hiện lên. Ở ô **Expression** ta gõ **(DT\_I4) 1** và đổi tên cột ở phần **Derived Column Name** thành **Temp** để tạo một cột mới với kiểu dữ liệu là **four-byte signed integer** với giá trị mặc định là 1. Nhấn **OK** và thoát cửa sổ **Derived Column Transformation Editor**



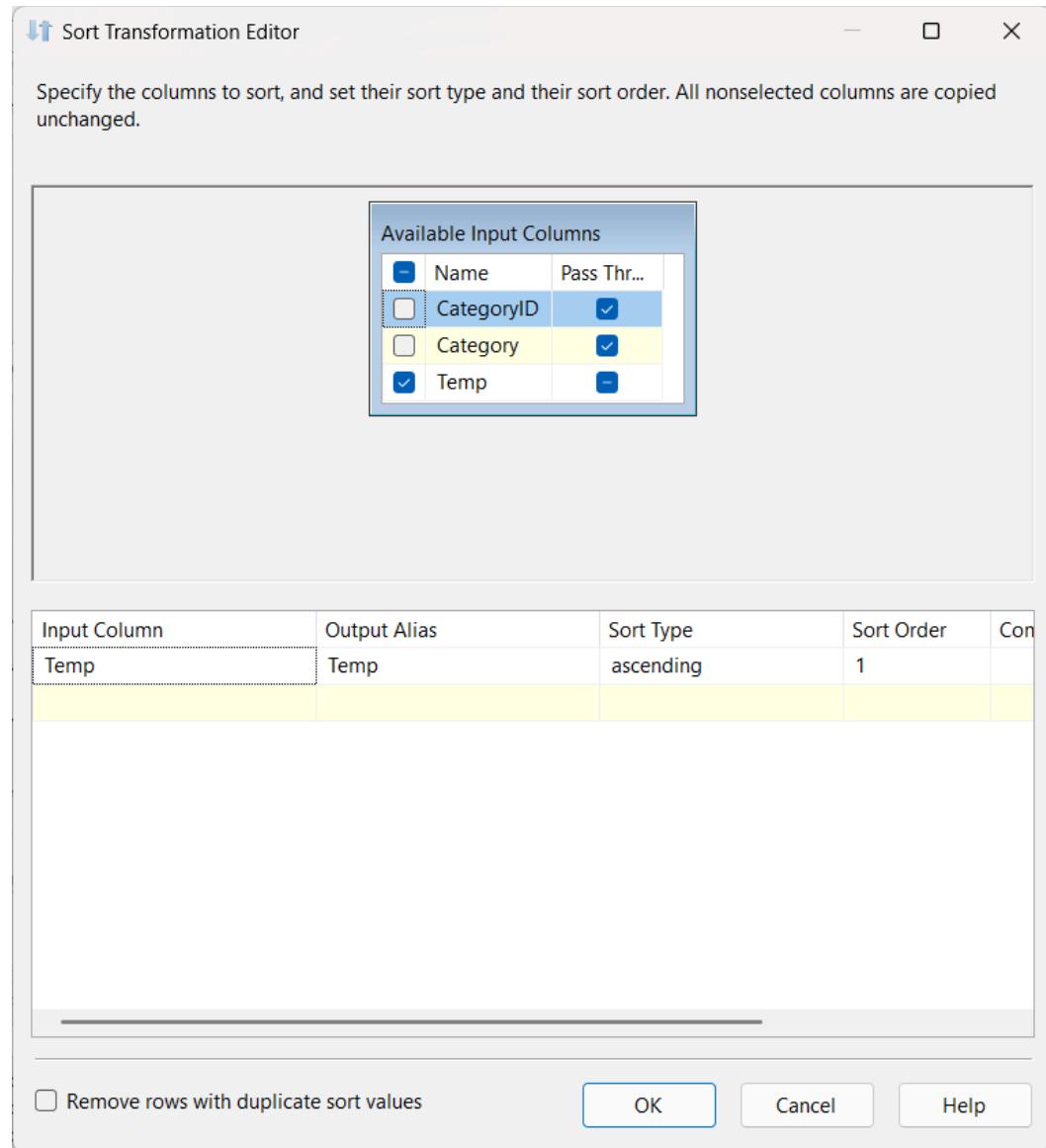
Hình 150. Cài đặt TEMPCOLUMN2

- **Bước 24:** Kéo chức năng Sort vào vùng Data Flow và kéo mũi tên luồng dữ liệu từ TEMPCOLUMN2 vào Sort 1.



Hình 151. Tạo chức năng Sort thứ 2 trong Dim\_DISCOUNTS

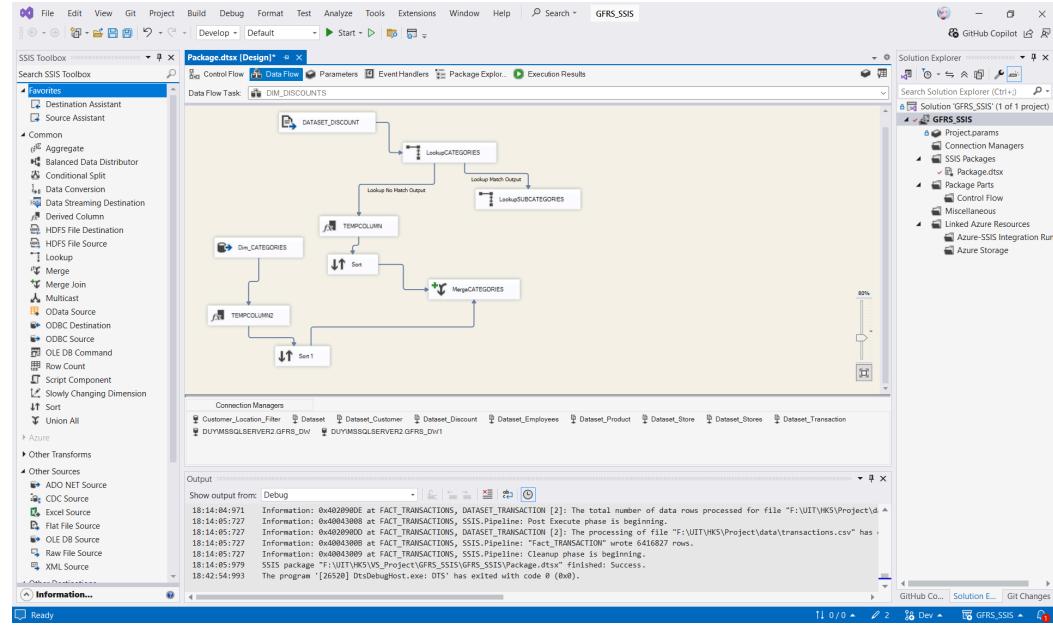
- **Bước 25:** Nhấn đúp chuột vào **Sort 1** và màn hình **Sort Transformation Editor** xuất hiện. Chọn cột **Temp**. Nhấn **OK** và thoát màn hình **Sort Transformation Editor**.



Hình 152. Cài đặt Sort 1 trong Dim\_DISCOUNTS

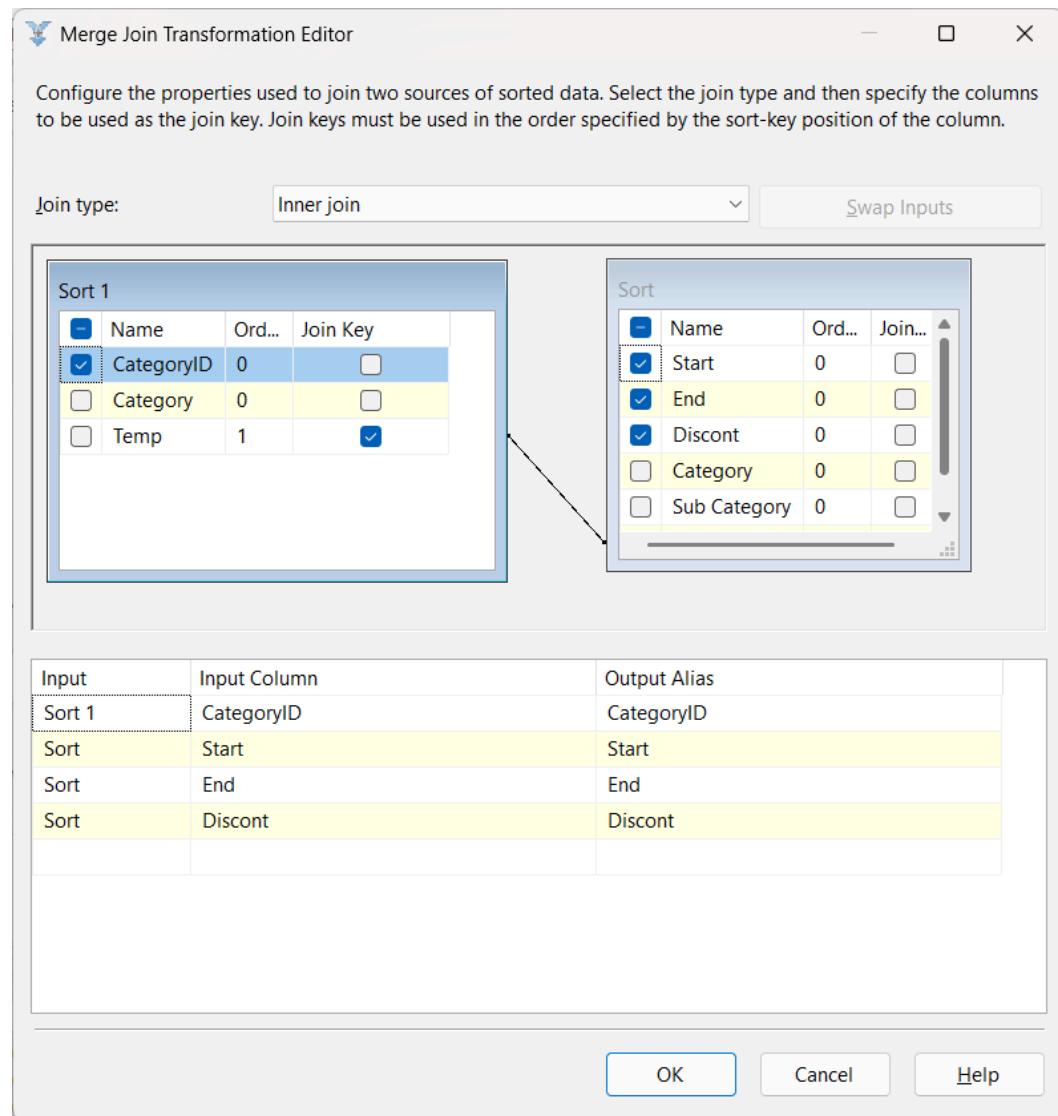
- **Bước 26:** Kéo chức năng **Merge Join** vào vùng **Data Flow** và đổi tên thành **MergeCATEGORIES**. Sau đó kéo mũi tên luồng dữ liệu từ **Sort** vào **MergeCATEGORIES**. Một cửa sổ sẽ hiện lên, ở phần **Input** ta chọn **Merge Join Left Input**. Sau đó kéo luồng dữ liệu từ **Sort 1** vào

## MergeCATEGORIES.



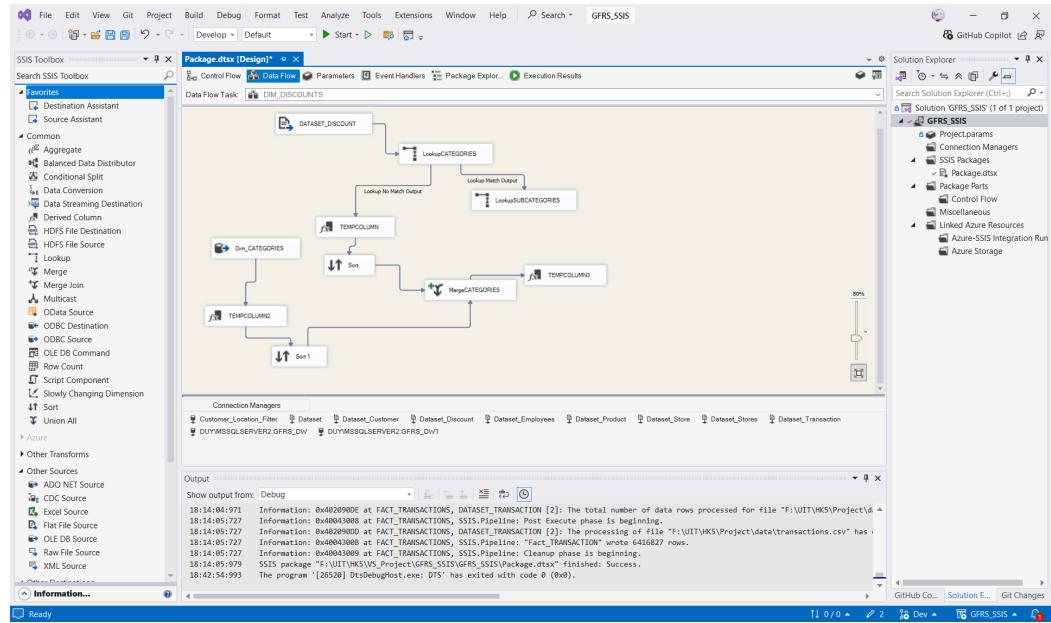
Hình 153. Tạo chức năng Merge Join trong Dim\_DISCOUNTS

- **Bước 27:** Nhấn đúp chuột vào **MergeCATEGORIES**, cửa sổ **Merger Join Transformation Editor** sẽ hiện lên. Phần **Join type** chọn **Inner join**. Ta chọn check box **Join Key** ở sau 2 cột giả Temp ở **Sort** và **Sort 1**. Tiếp đó chọn các phần tử muôn kết hợp. Nhập **OK** và kết thúc cửa sổ **Merger Join Transformation Editor**.



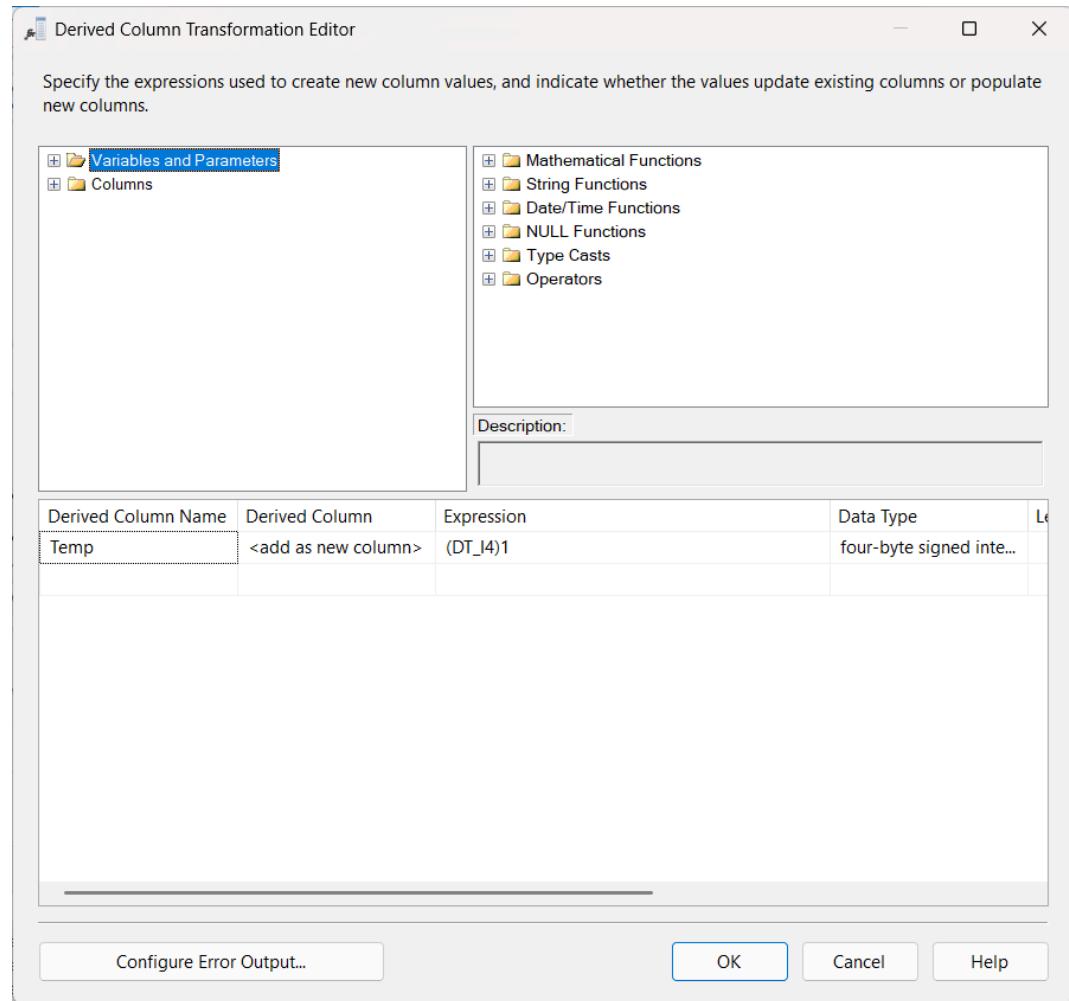
Hình 154. Cài đặt MergeCATEGORIES

- **Bước 28:** Ta kéo chức năng **Derived Column** vào vùng **Data Flow** và đổi tên thành **TEMPCOLUMN3**. Kéo luồng dữ liệu từ **MergeCATEGORIES** vào **TEMPCOLUMN3** để tạo một cột giả.



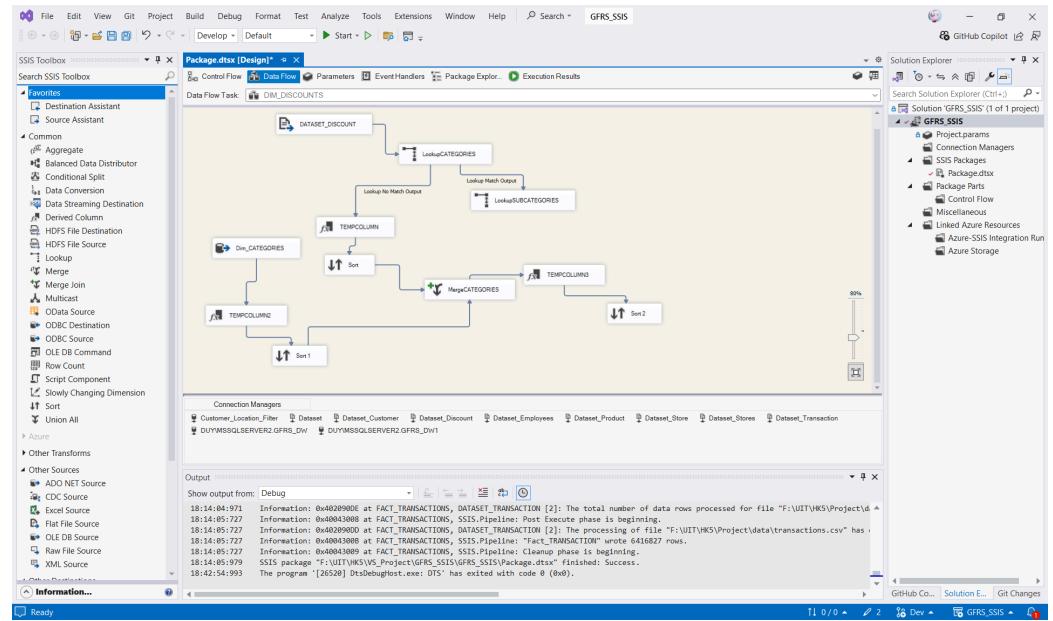
Hình 155. Tạo Derived Column thứ 3 trong Dim\_DISCOUNTS

- **Bước 29:** Nhấn đúp chuột vào **TEMPCOLUMN3** của sổ **Derived Column Transformation Editor** sẽ được hiện lên. Ở ô **Expression** ta gõ **(DT\_I4) 1** và đổi tên cột ở phần **Derived Column Name** thành **Temp** để tạo một cột mới với kiểu dữ liệu là **four-byte signed integer** với giá trị mặc định là 1. Nhấn **OK** và thoát cửa sổ **Derived Column Transformation Editor**



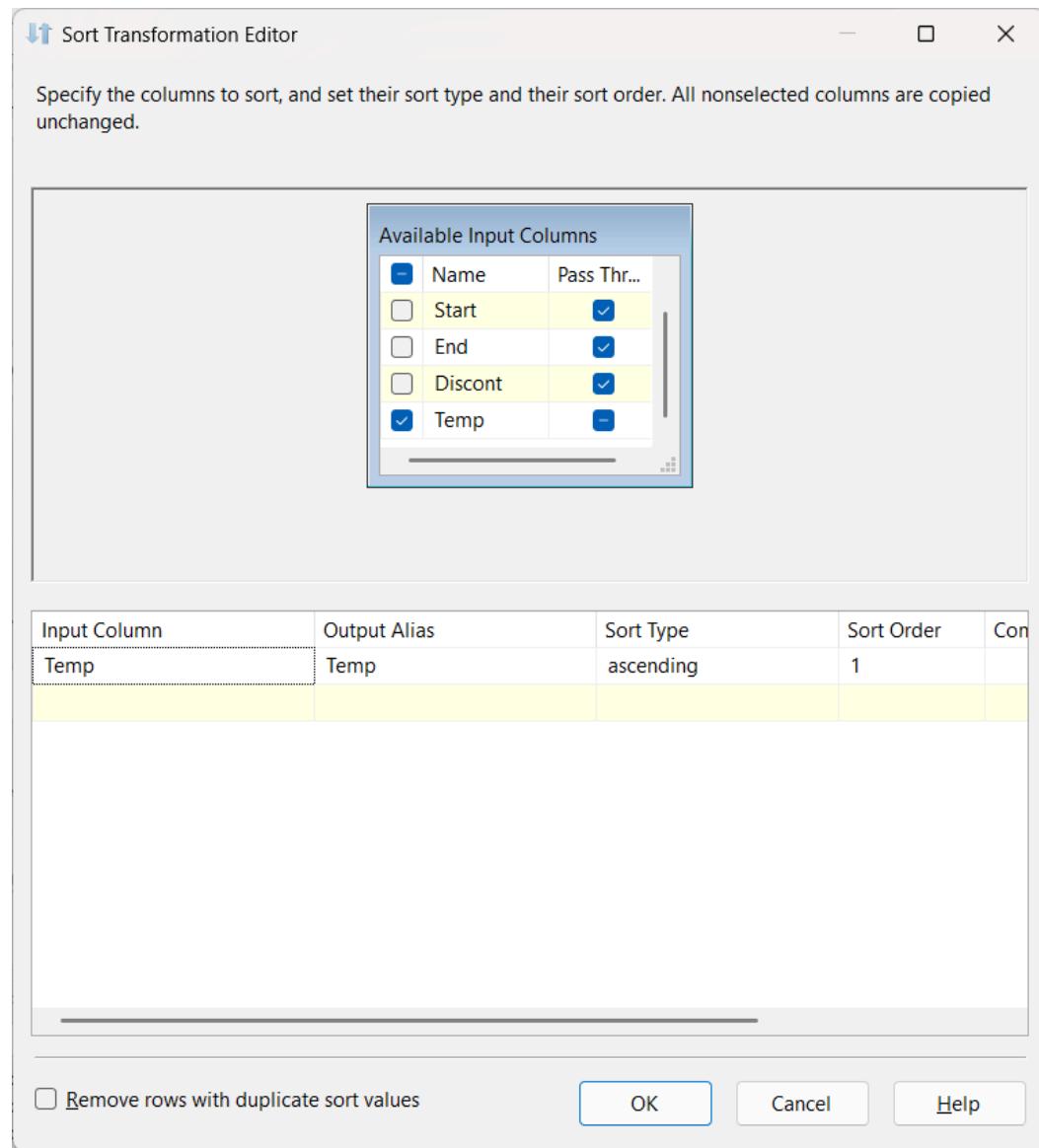
Hình 156. Cài đặt TEMPCOLUMN3

- **Bước 30:** Kéo chức năng Sort vào vùng Data Flow và kéo mũi tên luồng dữ liệu từ TEMPCOLUMN3 vào Sort 2.



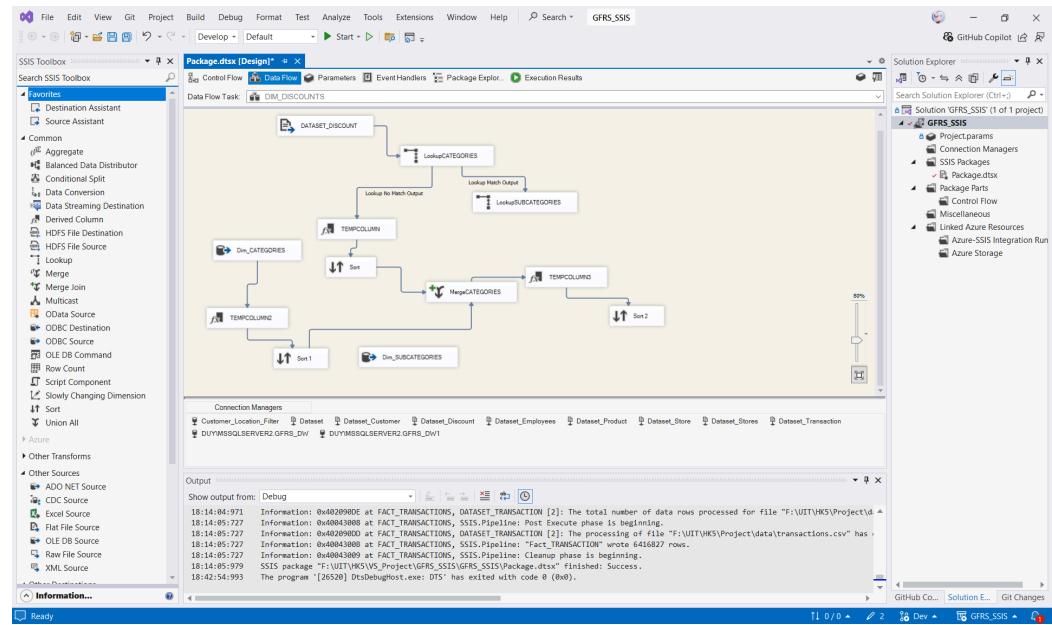
Hình 157. Tạo chức năng Sort thứ 3 trong Dim\_DISCOUNTS

- **Bước 31:** Nhấn đúp chuột vào **Sort** và màn hình **Sort Transformation Editor** xuất hiện. Chọn cột **Temp**. Nhấn **OK** và thoát màn hình **Sort Transformation Editor**.



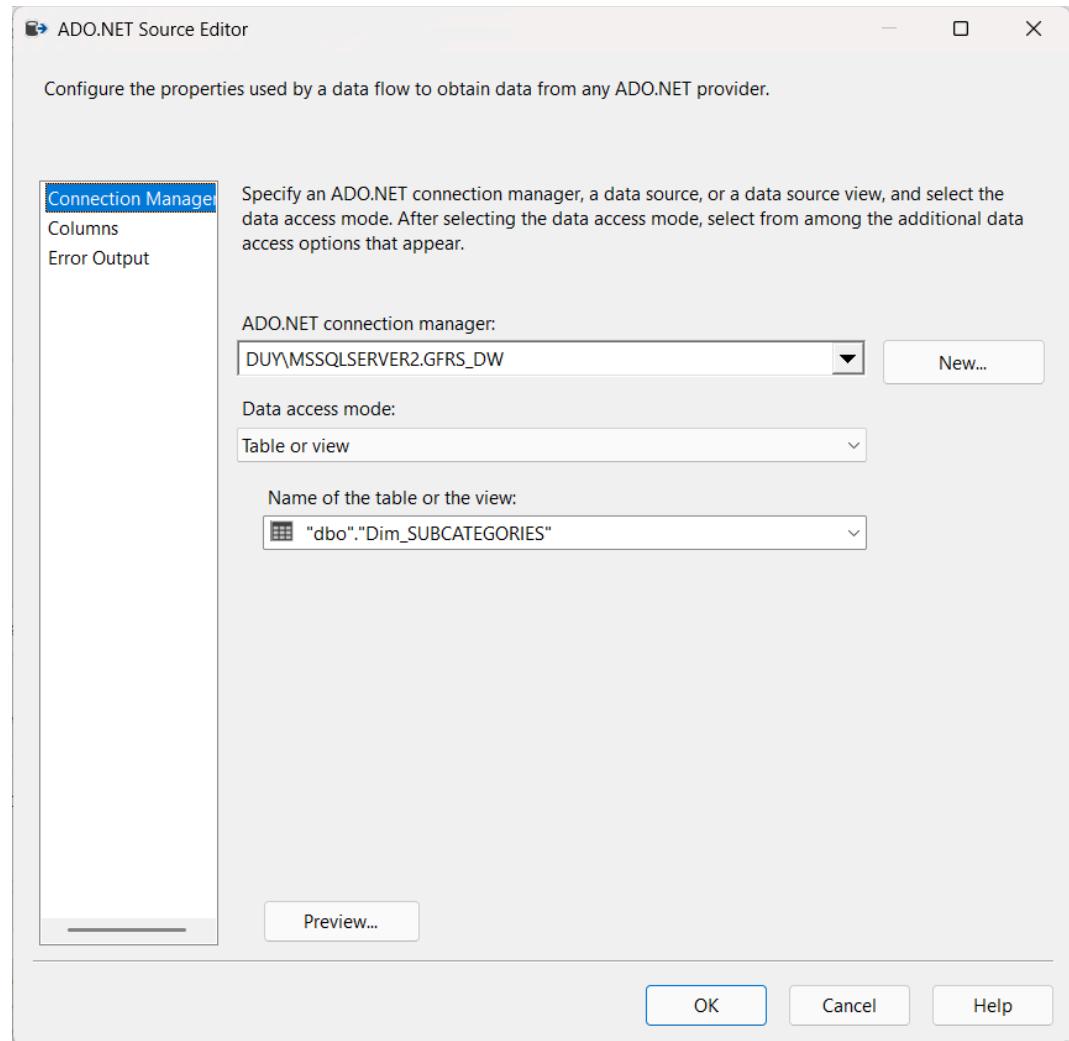
Hình 158. Cài đặt Sort 2

- **Bước 32:** Kéo chức năng ADO NET Source vào vùng Data Flow và đổi tên thành **Dim\_SUBCATEGORIES**.



Hình 159. Tạo ADO NET Source Dim\_SUBCATEGORIES trong Dim\_DISCOUNTS

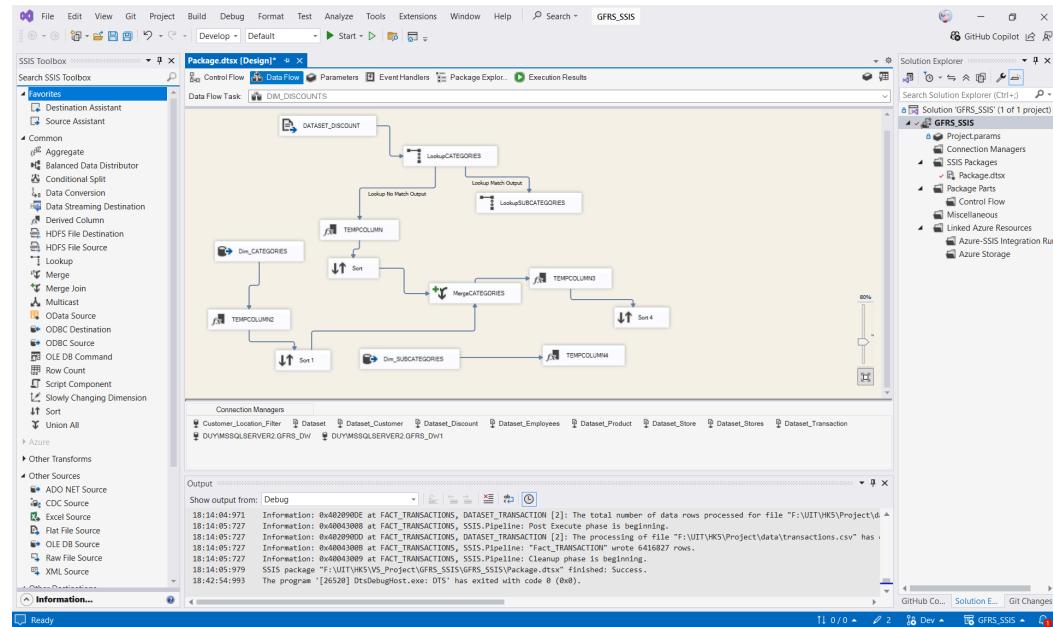
- **Bước 33:** Nhấn đúp chuột vào **Dim\_SUBCATEGORIES** cửa sổ **ADO.NET Source Editor** sẽ hiện lên, ở phần **Name of the table or the view** ta chọn “**dbo**.””**Dim\_SUBCATEGORIES**”. Nhấn **OK** để đóng cửa sổ **ADO.NET Source Editor**



Hình 160. Cài đặt nguồn dữ liệu cho ADO NET Source

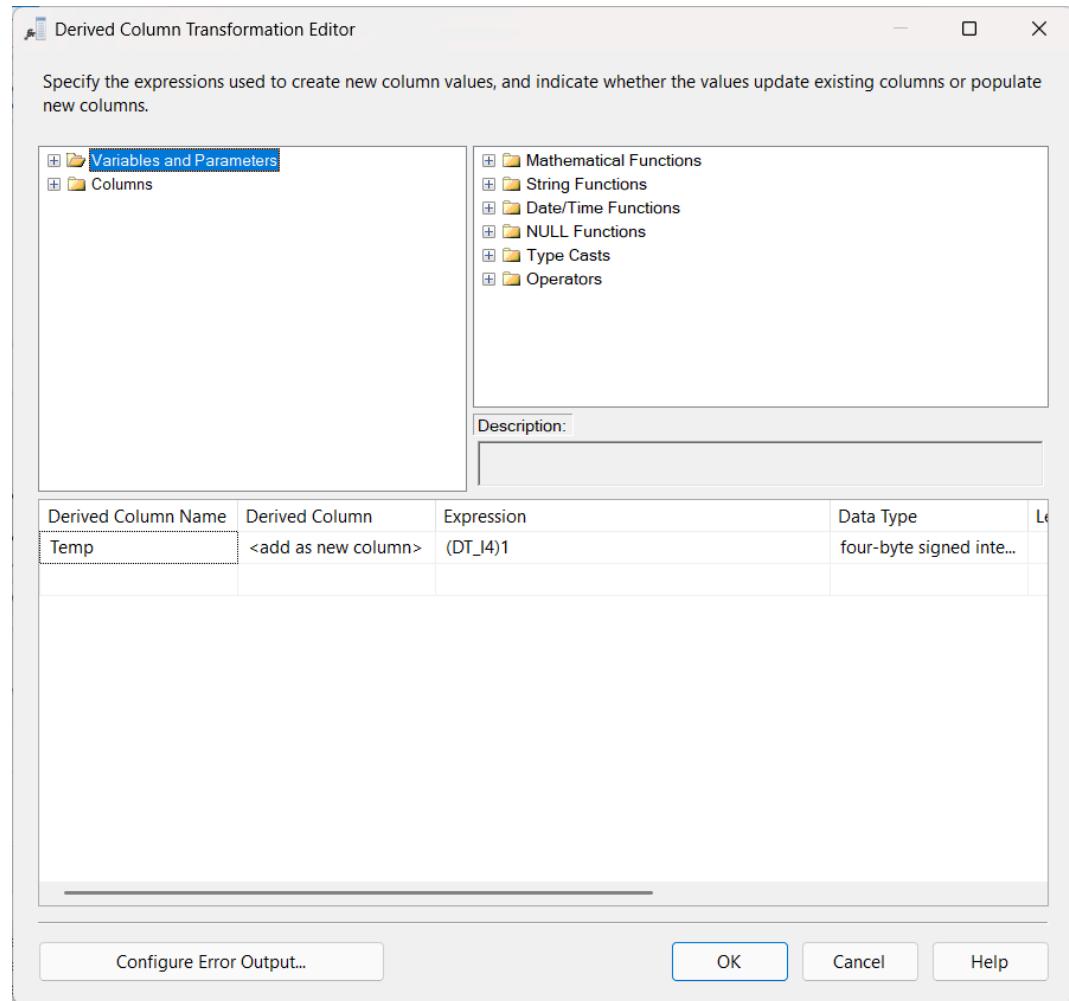
#### Dim\_SUBCATEGORIES

- **Bước 34:** Ta kéo chức năng **Derived Column** vào vùng **Data Flow** và đổi tên thành **TEMPCOLUMN4**. Kéo luồng dữ liệu từ **Dim\_SUBCATEGORIES** vào **TEMPCOLUMN4** để tạo một cột giả.



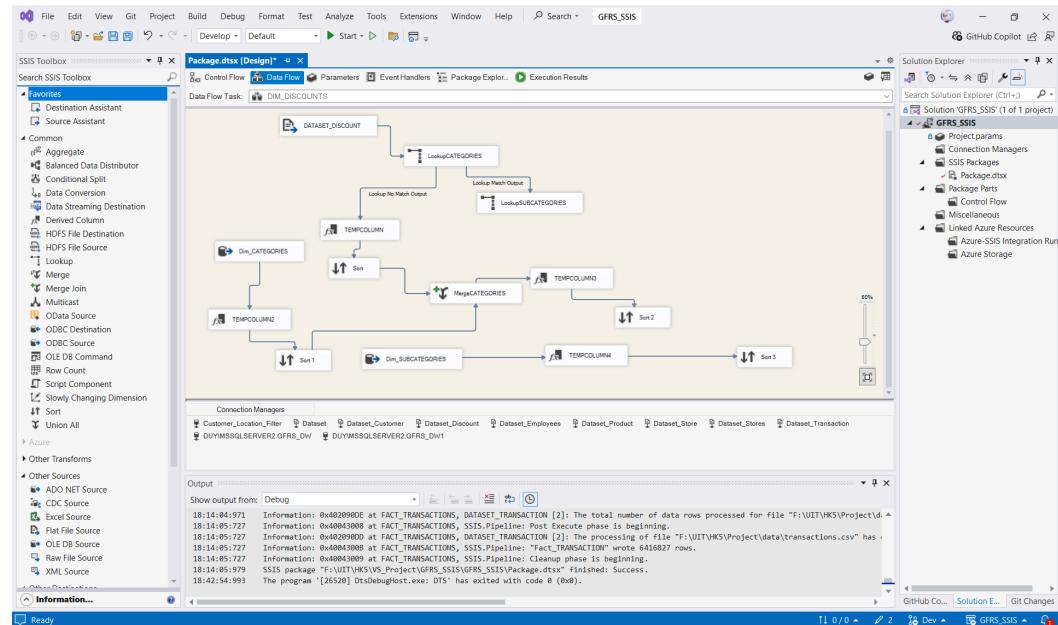
Hình 161. Tạo chức năng Derived Column thứ 4 trong Dim\_DISCOUNTS

- **Bước 35:** Nhấn đúp chuột vào **TEMPCOLUMN4** của sổ **Derived Column Transformation Editor** sẽ được hiện lên. Ở ô **Expression** ta gõ **(DT\_I4) 1** và đổi tên cột ở phần **Derived Column Name** thành **Temp** để tạo một cột mới với kiểu dữ liệu là **four-byte signed integer** với giá trị mặc định là 1. Nhấn **OK** và thoát cửa sổ **Derived Column Transformation Editor**



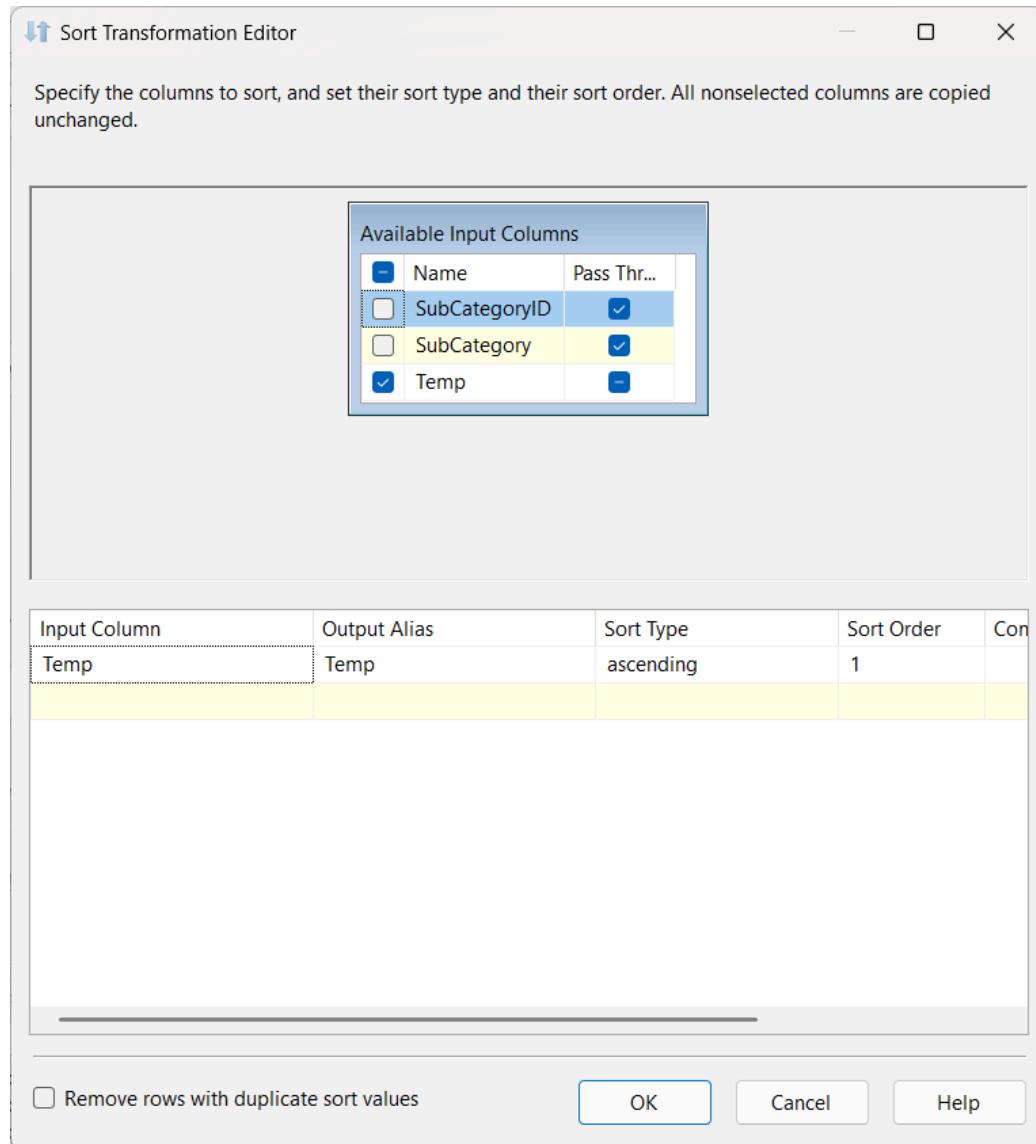
Hình 162. Cài đặt TEMPCOLUMN4

- **Bước 36:** Kéo chức năng Sort vào vùng Data Flow và kéo mũi tên luồng dữ liệu từ TEMPCOLUMN2 vào Sort 3.



Hình 163. Tạo chức năng Sort thứ 4 trong Dim\_DISCOUNTS

- **Bước 37:** Nhấn đúp chuột vào **Sort 3** và màn hình **Sort Transformation Editor** xuất hiện. Chọn cột **Temp**. Nhấn **OK** và thoát màn hình **Sort Transformation Editor**.

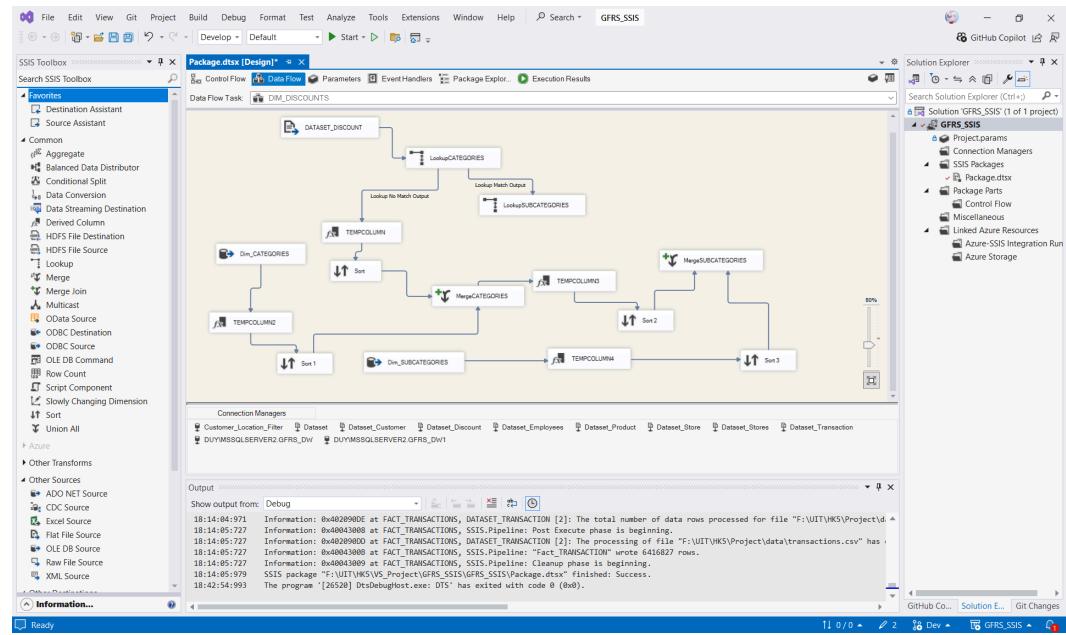


Hình 164. Cài đặt Sort 3

- **Bước 38:** Kéo chức năng **Merge Join** vào vùng **Data Flow** và đổi tên thành **MergeSUBCATEGORIES**. Sau đó kéo mũi tên luồng dữ liệu từ **Sort2** vào **MergeSUBCATEGORIES**. Một cửa sổ sẽ hiện lên, ở phần **Input** ta chọn **Merge Join Left Input**. Sau đó kéo luồng dữ liệu từ **Sort 3**

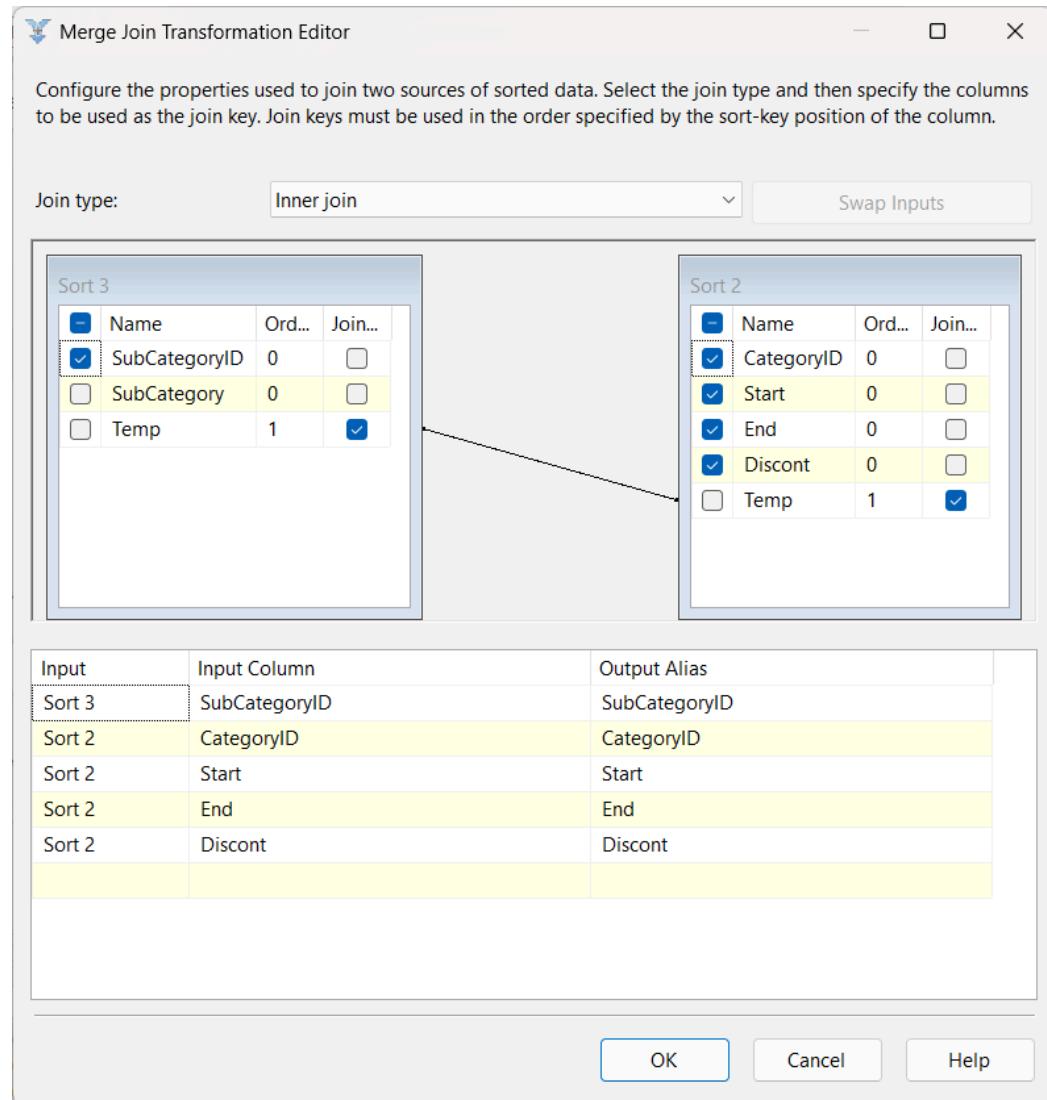
vào

## MergeSUBCATEGORIES.



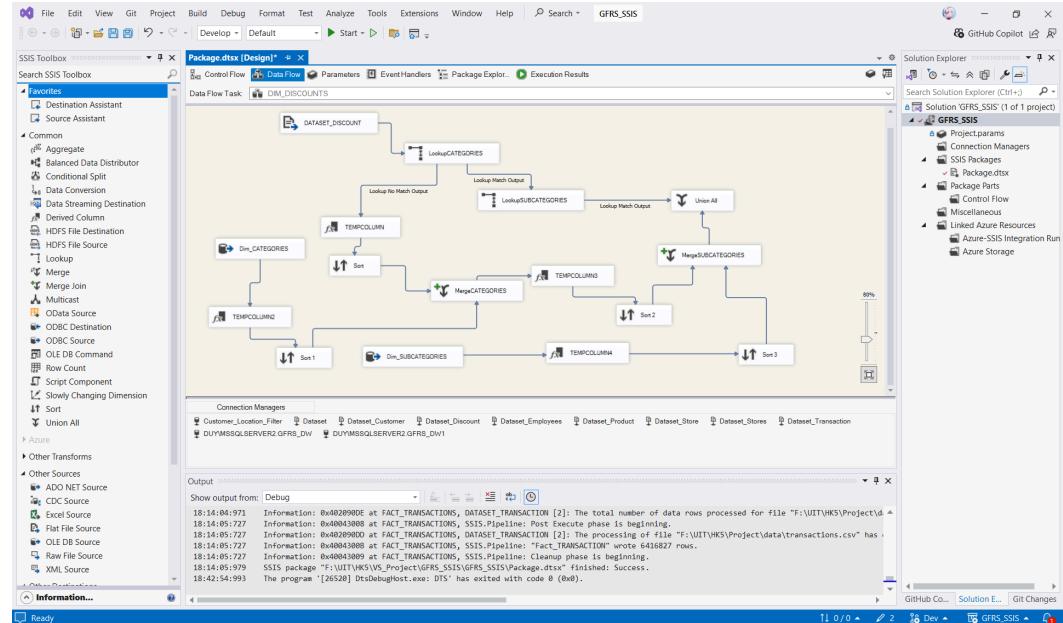
Hình 165. Tạo chức năng Merge Join thứ 2 trong Dim\_DISCOUNTS

- **Bước 39:** Nhấn đúp chuột vào **MergeSUBCATEGORIES**, cửa sổ **Merger Join Transformation Editor** sẽ hiện lên. Phần **Join type** chọn **Inner join**. Ta chọn check box **Join Key** ở sau 2 cột giả **Temp** ở **Sort2** và **Sort 3**. Tiếp đó chọn các phần tử muốn kết hợp. Nhập **OK** và kết thúc cửa sổ **Merger Join Transformation Editor**.



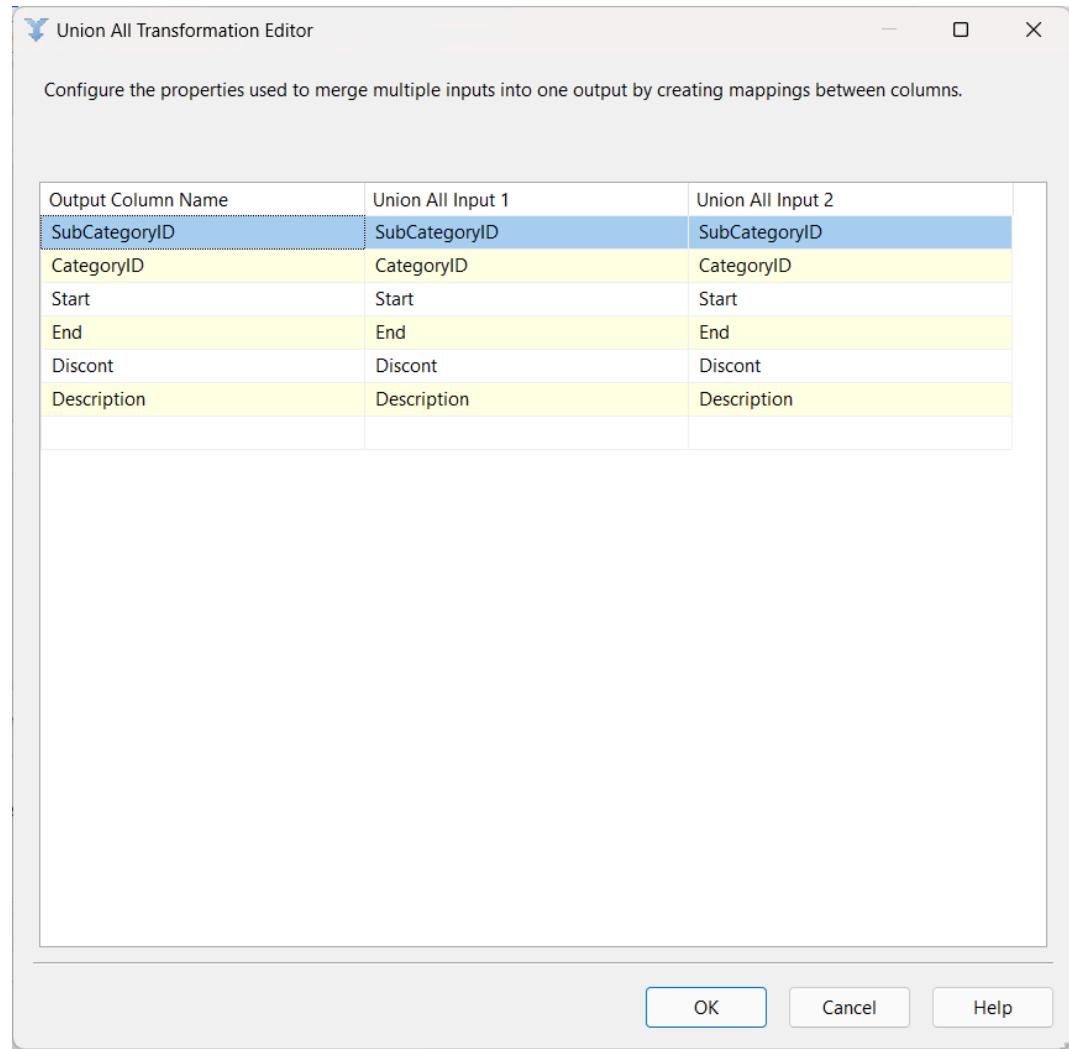
Hình 166. Cài đặt MergeSUBCATEGORIES

- **Bước 40:** Kéo chức năng **Union All** (một transformation dùng để gộp nhiều luồng dữ liệu thành một luồng duy nhất) vào cùng **Data Flow**. Kéo luồng dữ liệu từ **LookupSUBCATEGORIES** vào **Union All**. Một cửa sổ hiện lên, ta chọn **Lookup Match Output**. Sau đó, kéo luồng dữ liệu từ **MergeSUBCATEGORIES** vào **Union All**.



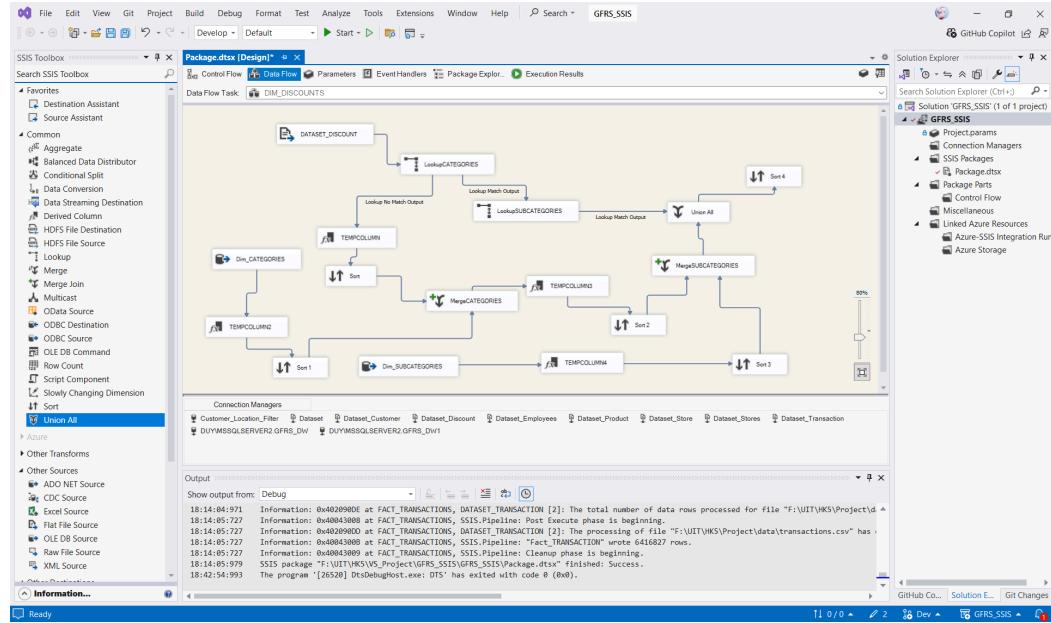
Hình 167. Tạo chức năng Union All trong Dim\_DISCOUNTS

- **Bước 41:** Nhấn đúp chuột vào **Union All**, cửa sổ **Union All Transformation Editor** hiện lên. Ta chọn các cột tương ứng nhau ở 2 Input. Sau đó bấm **OK** và thoát khỏi màn hình **Union All Transformation Editor**.



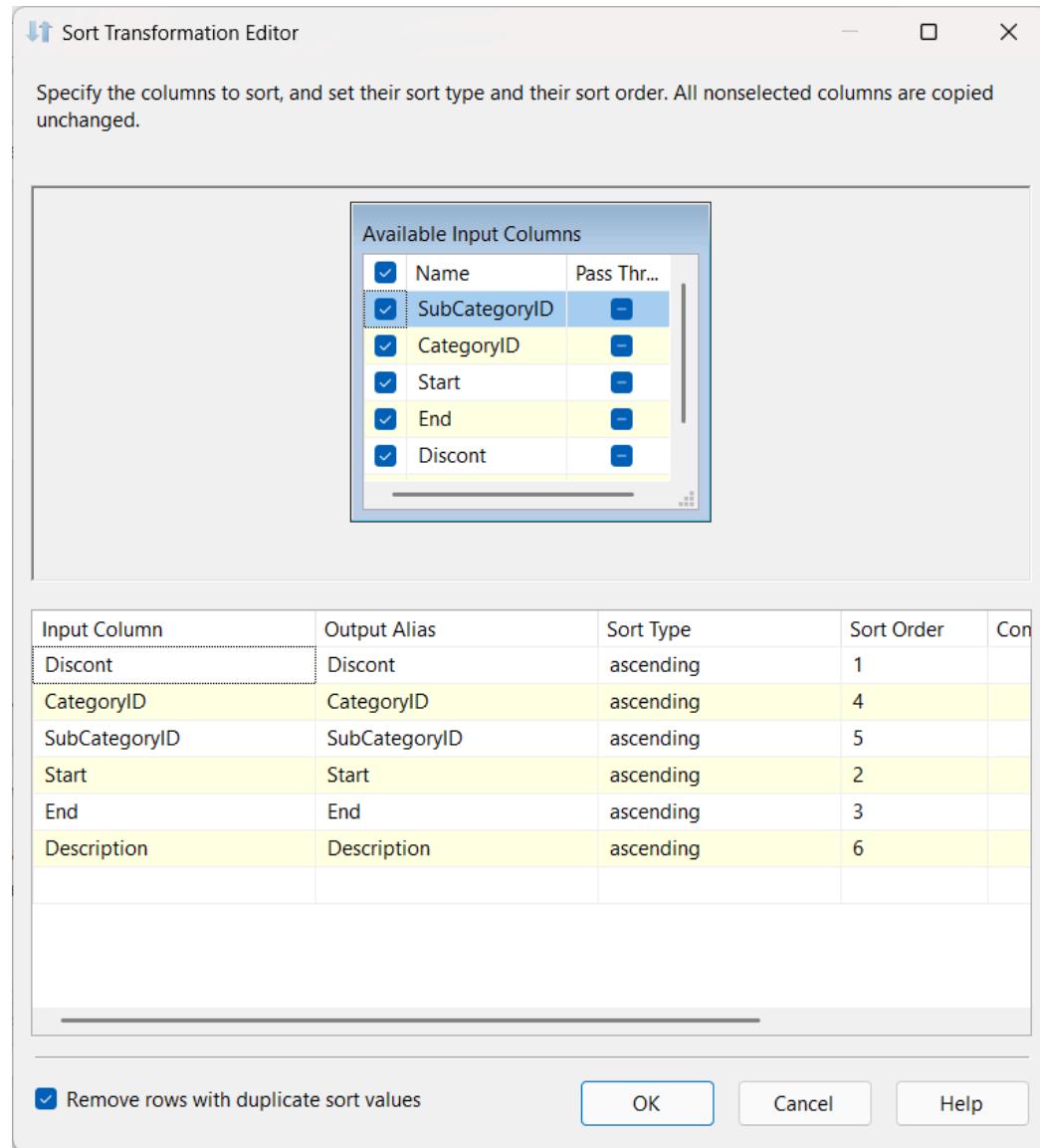
Hình 168. Cài đặt Union All trong Dim\_DISCOUNTS

- **Bước 42:** Kéo chức năng Sort vào vùng Data Flow và kéo mũi tên luồng dữ liệu từ Union All vào Sort 4.



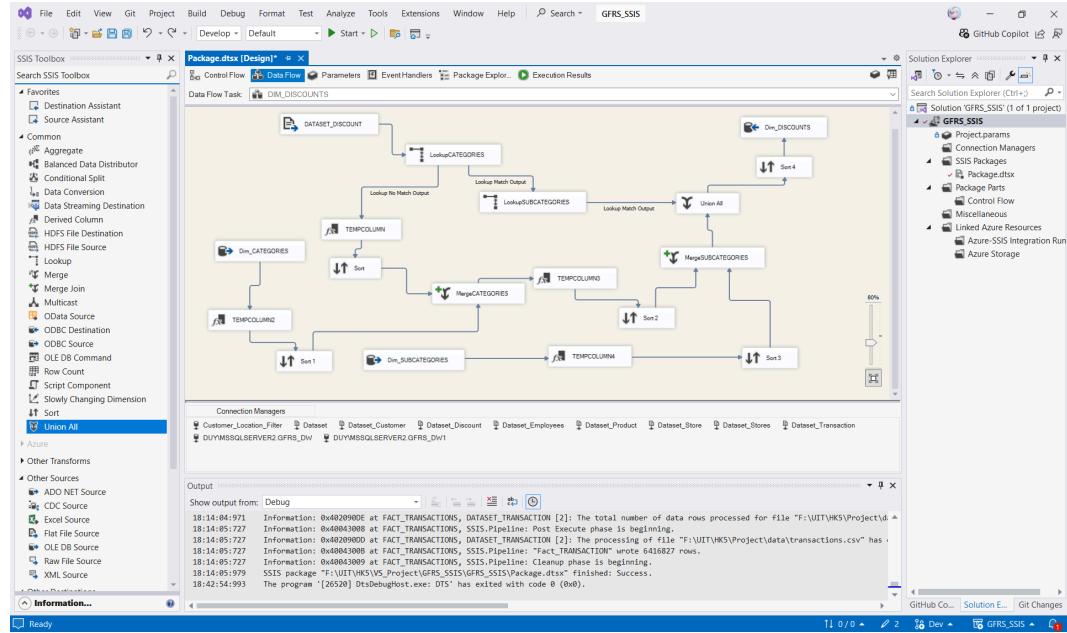
Hình 169. Tạo chức năng Sort thứ 5 trong Dim\_DISCOUNTS

- **Bước 43:** Nhấn đúp chuột vào **Sort 4** và màn hình **Sort Transformation Editor** xuất hiện. Chọn các cột cần sắp xếp và chọn **Remove rows with duplicate sort values**. Nhấn **OK** và kết thúc cửa sổ **Sort Transformation Editor**



Hình 170. Cài đặt Sort 4 trong Dim\_DISCOUNTS

- **Bước 44:** Kéo chức năng **ADO NET Destination** vào vùng **Data Flow** và đổi tên thành **Dim\_DISCOUNTS**. Kéo mũi tên luồng dữ liệu từ **Sort 4** vào **Dim\_DISCOUNTS**.



Hình 171. Tạo chức năng ADO NET Destination Dim\_DISCOUNTS

- **Bước 45:** Vào SQL Server 2022, chạy lệnh DDL để tạo **Dim\_DISCOUNTS** và đồng thời tạo thêm dữ liệu với **Discount = 0.0** tương ứng trung cho việc không có giảm giá cho giao dịch bán hàng.

```

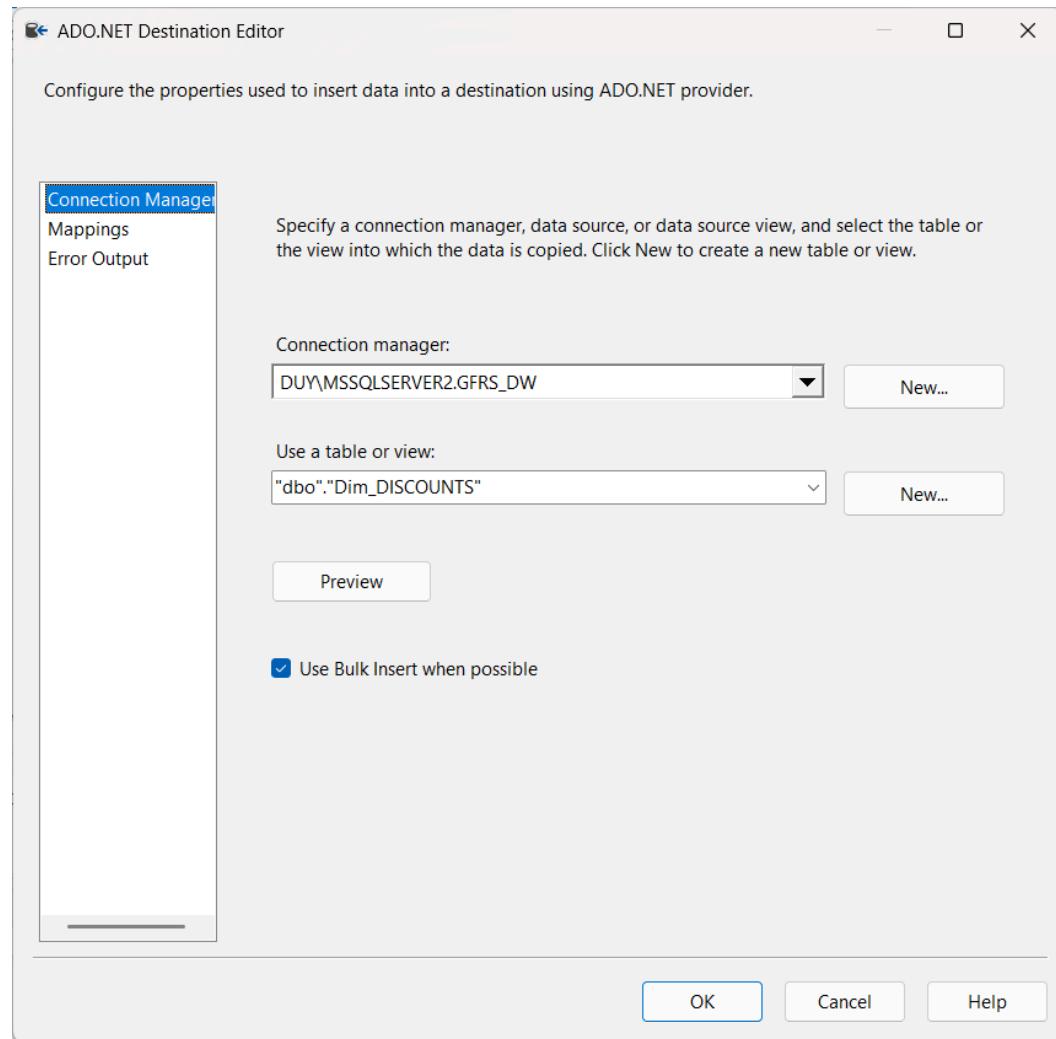
CREATE TABLE [Dim_DISCOUNTS] (
    [DiscountID] INT IDENTITY(1,1) PRIMARY KEY,
    [Start] DATETIME,
    [End] DATETIME,
    [Discount] DECIMAL(10,2),
    [Description] VARCHAR(255),
    [SubCategoryID] INT,
    [CategoryID] INT
)
GO
-- Insert default row as in your original script (optional)
INSERT INTO Dim_DISCOUNTS([Start], [End], [Discount], [SubCategoryID], [CategoryID])
VALUES (NULL, NULL, 0.0, NULL, NULL);

CREATE TABLE [DISCOUNTS_BRIDGE] (
)

```

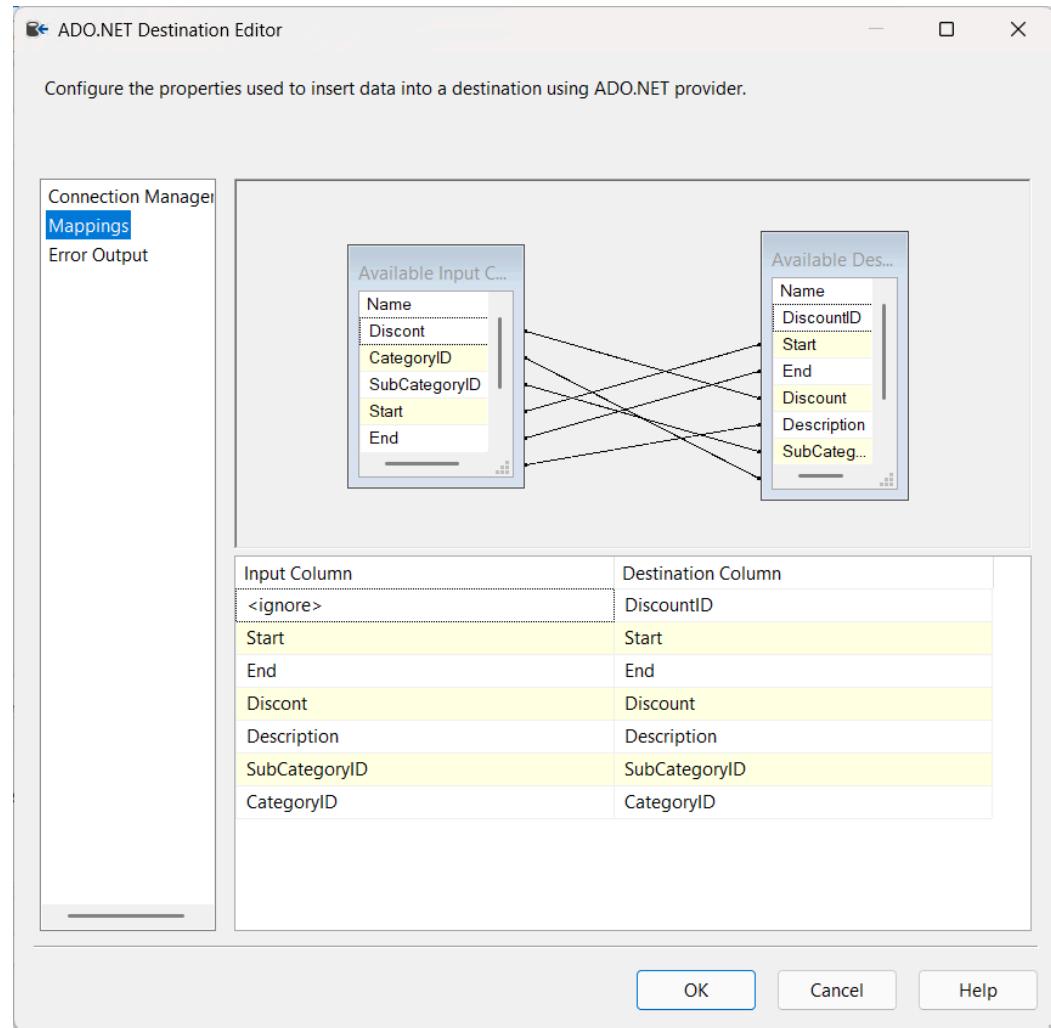
Hình 172. Chạy lệnh DDL tạo Dim\_DISCOUNTS

- **Bước 46:** Nhấn đúp chuột vào **Dim\_DISCOUNTS**. Trong tab **Connection Manager** của màn hình **ADO.NET Destination Editor**, ở **Use a table or view** tiến hành nhấp chọn “**dbo”.”Dim\_DISCOUNTS**”, cũng chính là bảng vừa được tạo bằng lệnh DDL ở **SQL Server 2022**



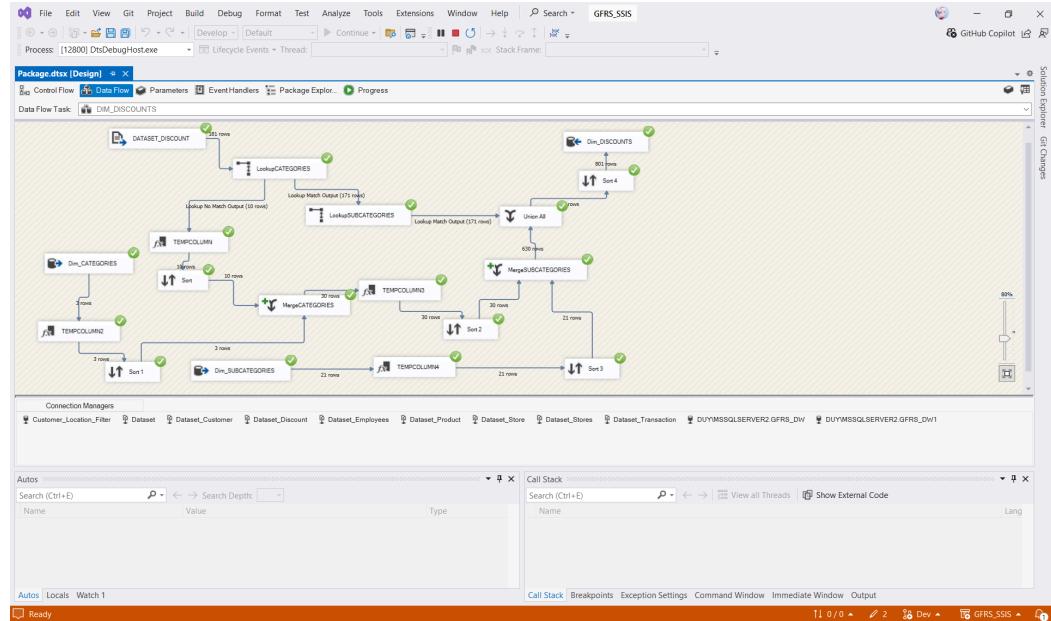
*Hình 173. Kết nối đích đến cho ADO.NET Destination Dim\_DISCOUNTS*

- **Bước 47:** Ở tab **Mapping** của màn hình **ADO.NET Destination Editor** tiến hành chọn các thuộc tính thích hợp ở cột bên trái để ánh xạ với các thuộc tính ở cột bên phải.



Hình 174. Cài đặt ADO NET Destination Dim\_DISCOUNTS

- **Bước 48:** Nhấn Start để kiểm tra quá trình tạo và đổ dữ liệu vào Dim\_DISCOUNTS có thành công hay chưa.



Hình 175. Đỗ dữ liệu vào Dim\_DISCOUNTS

Kết quả hiển thị cho thấy quá trình diễn ra thành công và có **801 rows** được đỗ vào **DIM\_DISCOUNTS**.

- **Bước 49:** Vào SQL Server 2022 kiểm tra và thấy dữ liệu đã được đỗ vào Dim\_DISCOUNTS thành công.

```

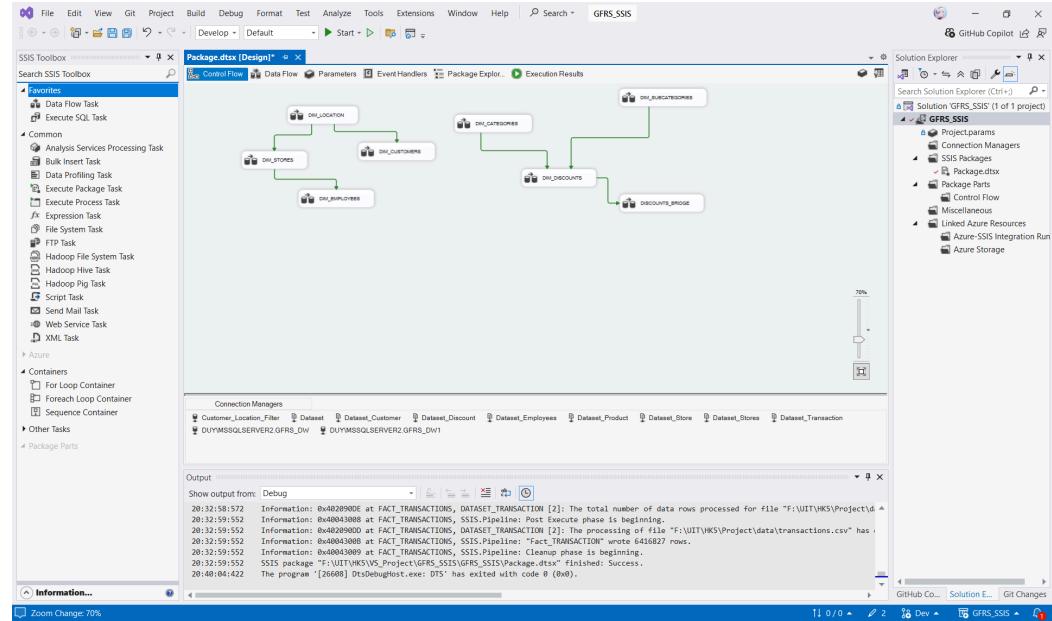
File Edit View Query Git Project Tools Extensions Window Help Solution1 Sign in
File Edit View Query Git Project Tools Extensions Window Help Solution1 Sign in
Object Explorer
Connect > DUYMSSQLSERVER2 (SQL Server 16... 73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
No issues found
Results Messages
DiscountID Start End Discount Description SubCategoryID CategoryID
1 1 2020-10-01 00:00:00.000 0.20 20% discount during our Autumn Essentials Sale 1 1
2 2 2020-10-01 00:00:00.000 0.20 20% discount during our Autumn Essentials Sale 16 1
3 3 2020-10-01 00:00:00.000 0.20 20% discount during our Autumn Essentials Sale 16 1
4 4 2020-10-01 00:00:00.000 0.20 20% discount during our Autumn Essentials Sale 13 2
5 5 2020-10-01 00:00:00.000 0.20 20% discount during our Autumn Essentials Sale 17 2
6 6 2020-10-01 00:00:00.000 0.20 20% discount during our Autumn Essentials Sale 13 3
7 7 2020-10-01 00:00:00.000 0.20 20% discount during our Autumn Essentials Sale 18 3
8 8 2021-10-01 00:00:00.000 0.20 20% discount during our Autumn Essentials Sale 1 1
9 9 2021-10-01 00:00:00.000 0.20 20% discount during our Autumn Essentials Sale 16 1
10 10 2021-10-01 00:00:00.000 0.20 20% discount during our Autumn Essentials Sale 13 2
11 11 2021-10-01 00:00:00.000 0.20 20% discount during our Autumn Essentials Sale 17 2
12 12 2021-10-01 00:00:00.000 0.20 20% discount during our Autumn Essentials Sale 13 3

```

Hình 176. Dữ liệu được đỗ thành công vào Dim\_DISCOUNTS

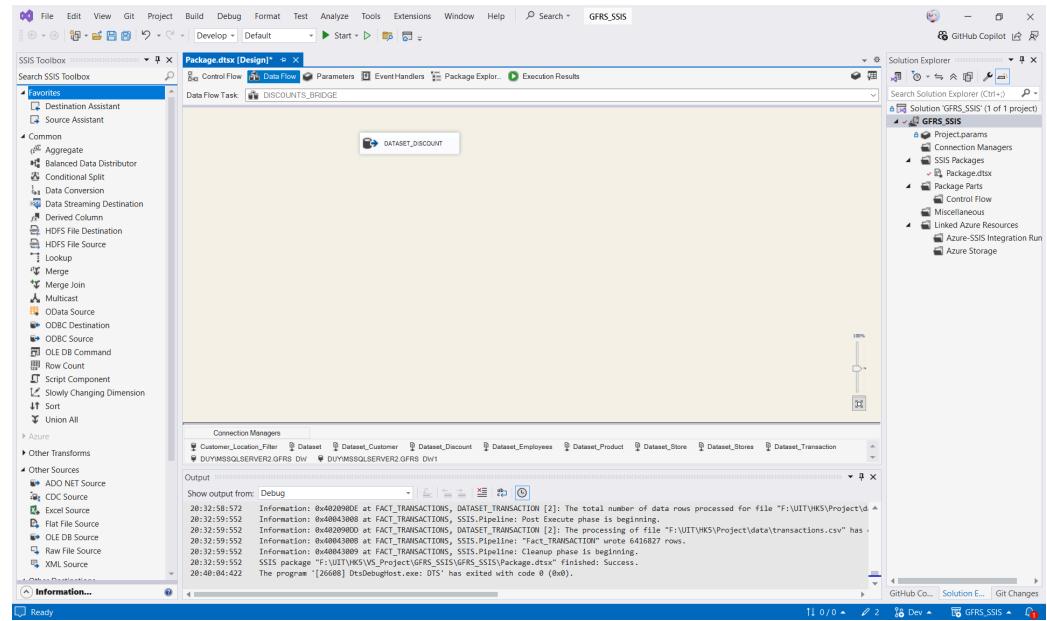
## II.3.7. DISCOUNTS\_BRIDGE

- **Bước 1:** Kéo chức năng Data Flow Task trong vùng SSIS Toolbox vào vùng Control Flow và đổi tên thành **DISCOUNTS\_BRIDGE**.



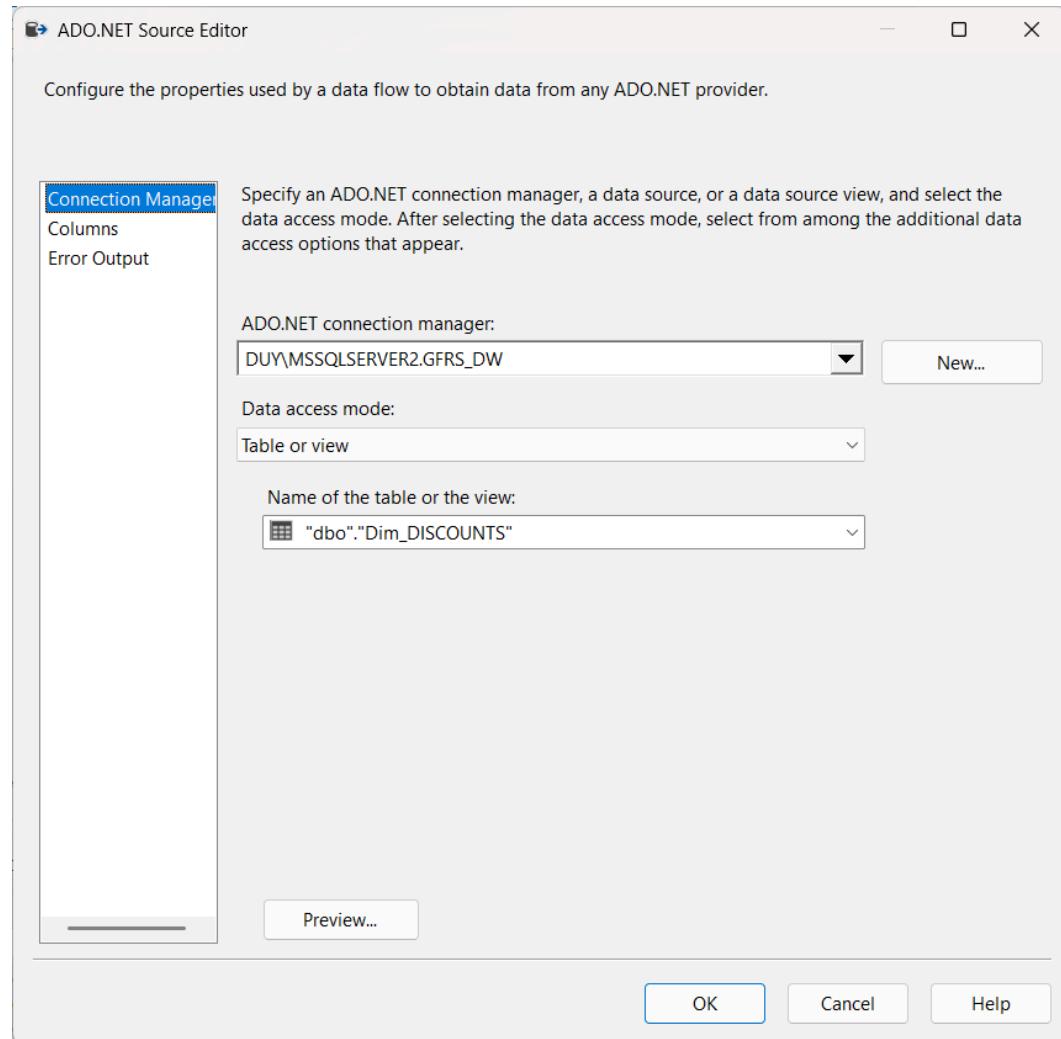
Hình 177. Tạo **DISCOUNTS\_BRIDGE** trong Control Flow

- **Bước 2:** Nhấn đúp chuột vào **DISCOUNTS\_BRIDGE**. Sau đó kéo chức năng **Flat File Source** vào vùng làm việc và đổi tên thành **DATASET\_DISCOUNT**



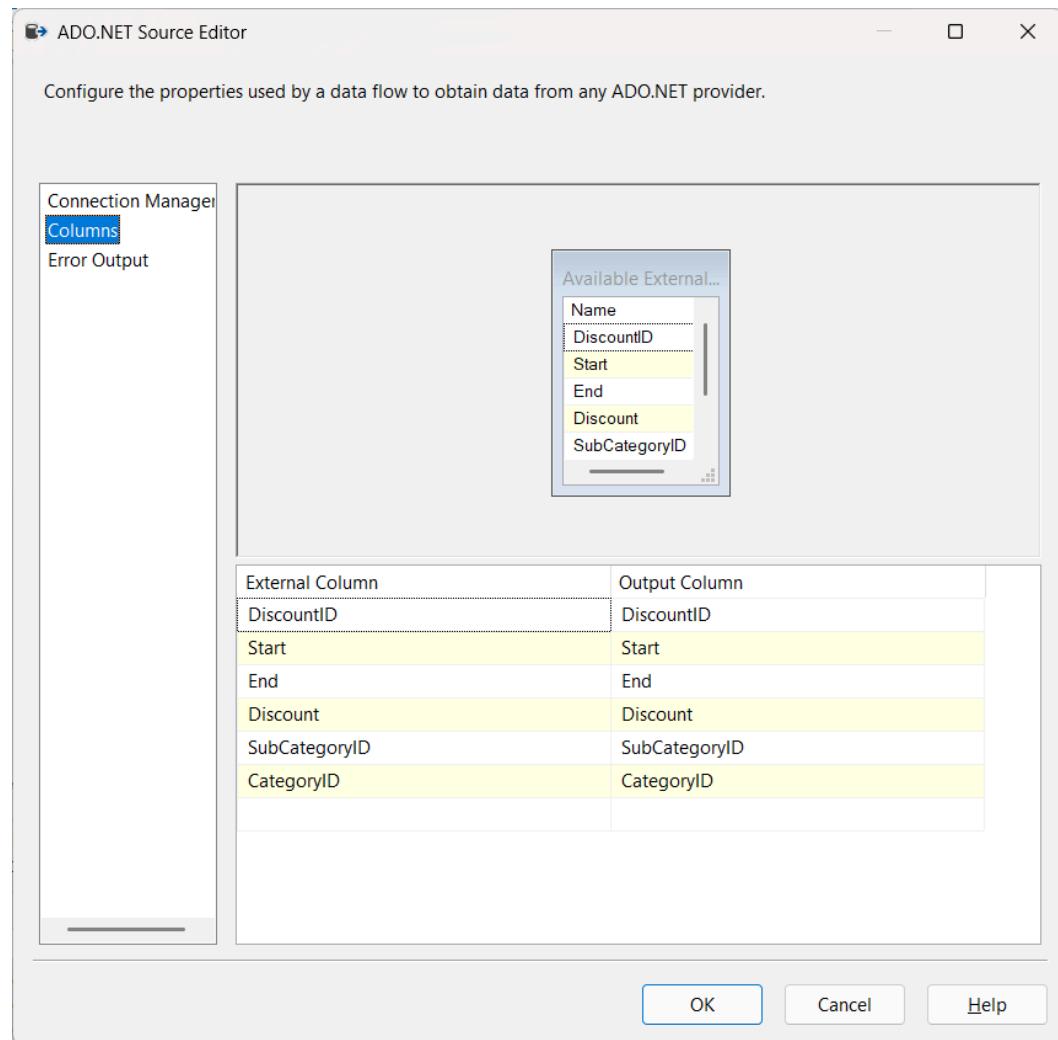
Hình 178. Tạo DATASET\_DISCOUNT trong DISCOUNTS\_BRIDGE

- **Bước 3:** Nhấn đúp chuột vào DATASET\_DISCOUNT. Màn hình Flat File Source Editor hiện ra. Do quá trình tạo Dim\_DISCOUNTS, ở các bước trước đã load thành công file csv và đổi tên thành Dataset\_Discount nên ta có thể sử dụng lại Dataset\_Discount để load dữ liệu mà không cần thiết lập lại giúp tiết kiệm thời gian và giảm nguy cơ gặp lỗi.



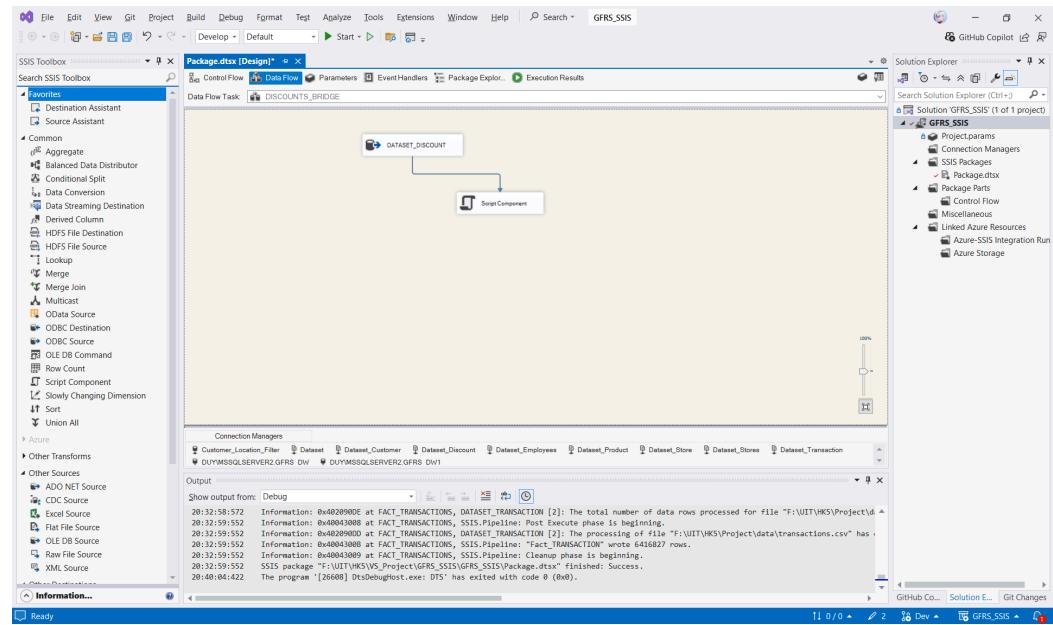
Hình 179. Kết nối nguồn dữ liệu cho DISCOUNTS\_BRIDGE

- **Bước 4:** Chuyển sang tab **Columns**, ta có thể lựa chọn các thuộc tính cần thiết cho quá trình nạp dữ liệu cho **DISCOUNTS\_BRIDGE**. Sau đó ta có thể bấm **OK** để đóng cửa sổ **Flat File Source Editor**



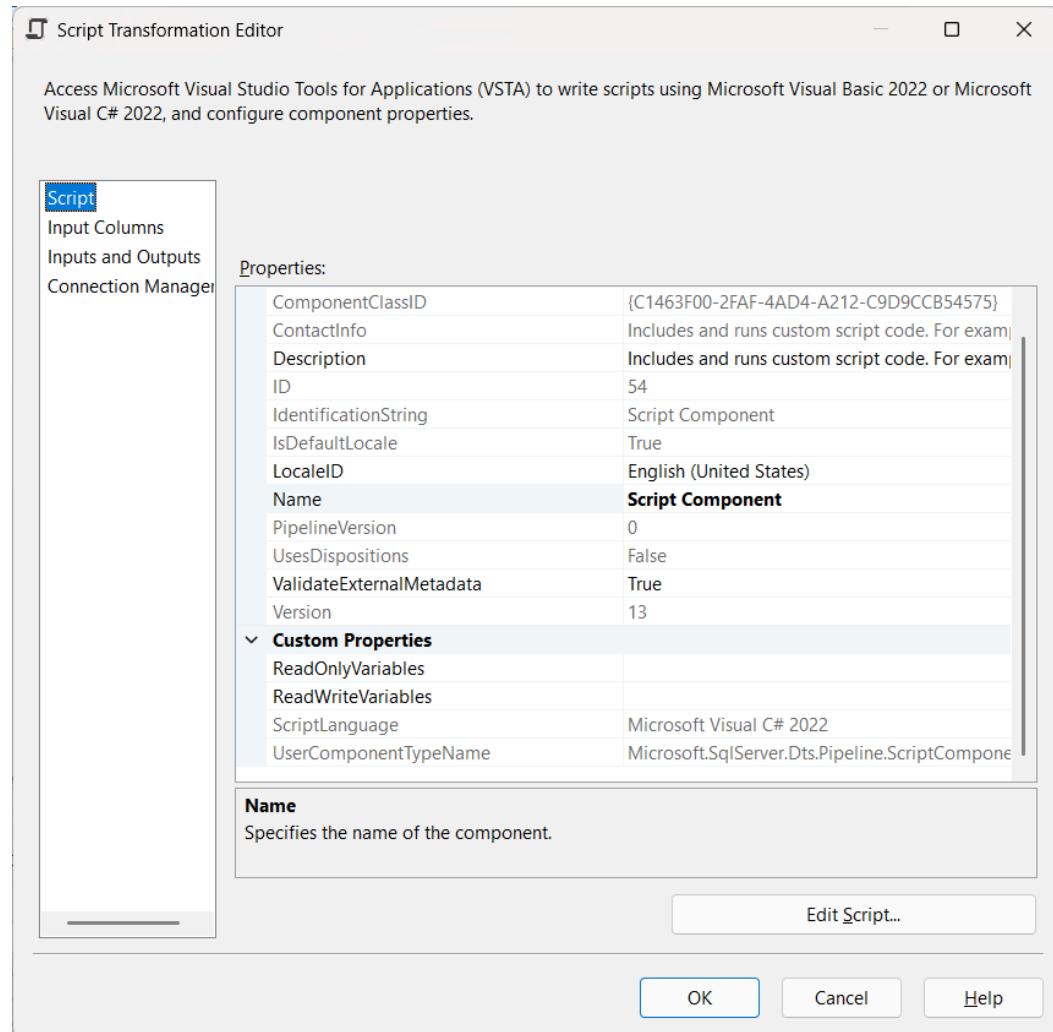
Hình 180. Lựa chọn thuộc tích cản thiết cho quá trình tạo  
DISCOUNTS\_BRIDGE

- **Bước 5:** Kéo Script Component từ cùng SSIS Toolbox vào vùng Data Flow. Sau đó kéo luồng dữ liệu từ **DATASET\_DISCOUNT** vào Script Component.



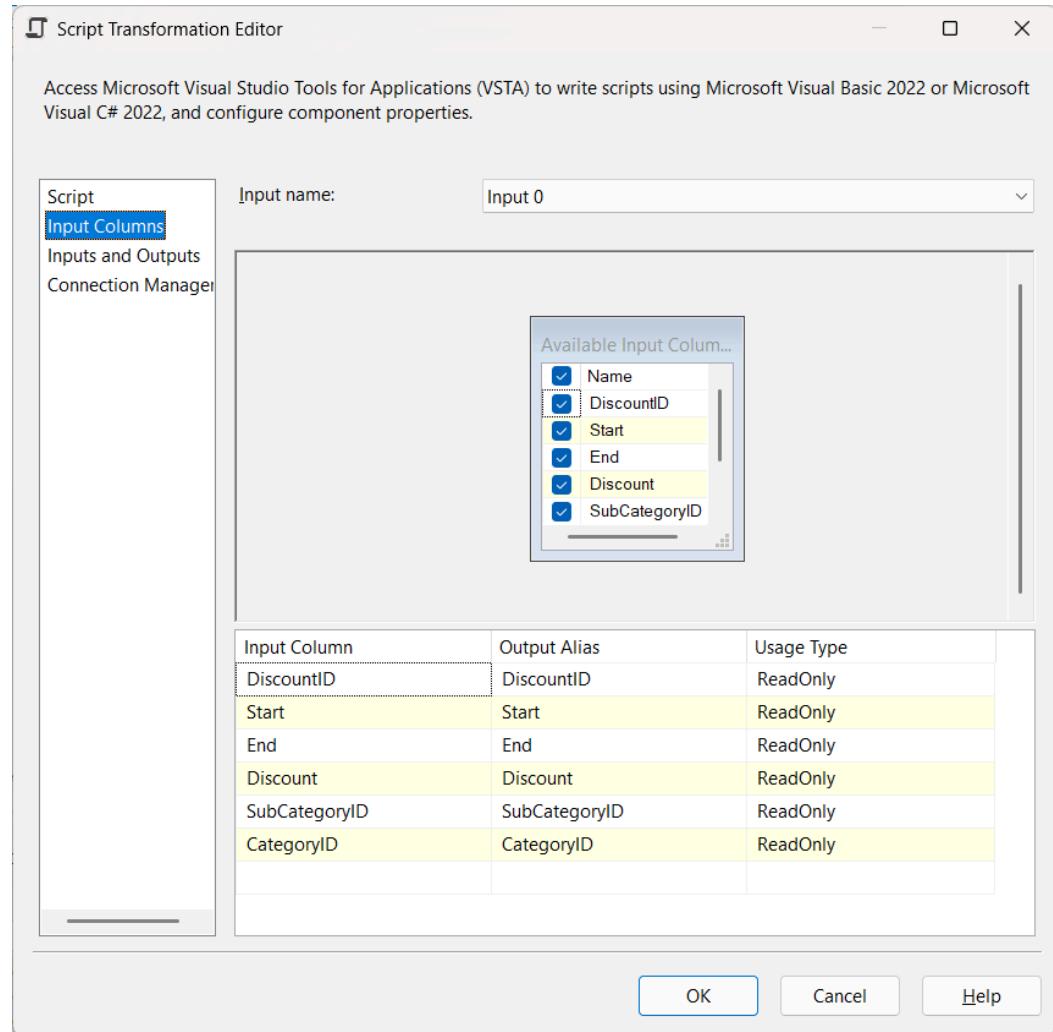
*Hình 181. Tạo chức năng Script Component trong DISCOUNTS\_BRIDGE*

- **Bước 6:** Nhấn đúp chuột vào **Script Component**, cửa sổ **Script Transformation Editor** hiện lên.



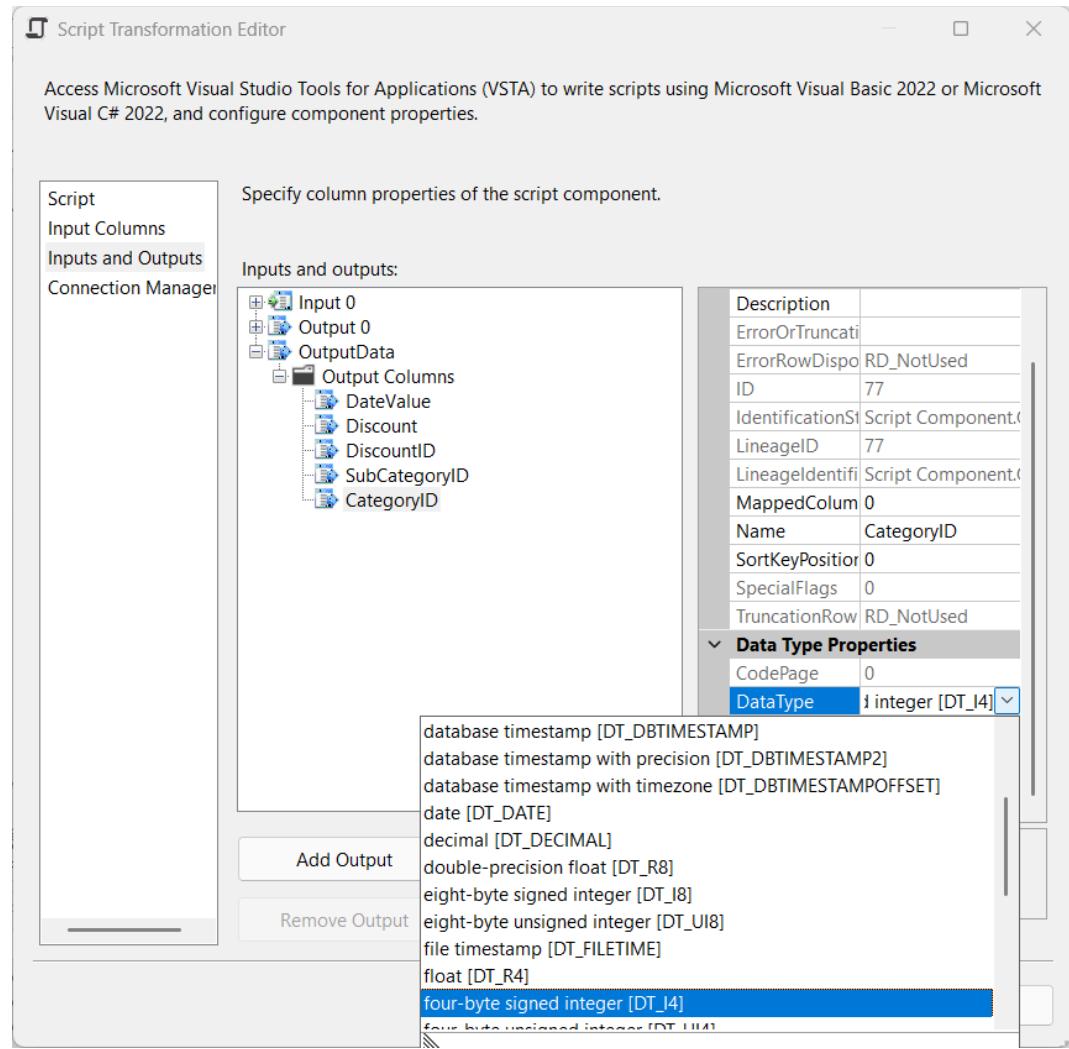
Hình 182. Kiểm tra cấu hình Script Component

- **Bước 7:** Chuyển sang tab **Input Columns**, ta sẽ chọn các cột cần thiết để xử lý chuyển đổi dữ liệu gồm ngày bắt đầu (**Start**) đến ngày kết thúc (**End**) sang các dữ liệu **DateValue** (với **DateValue >= Start** và **DateValue <= End**)



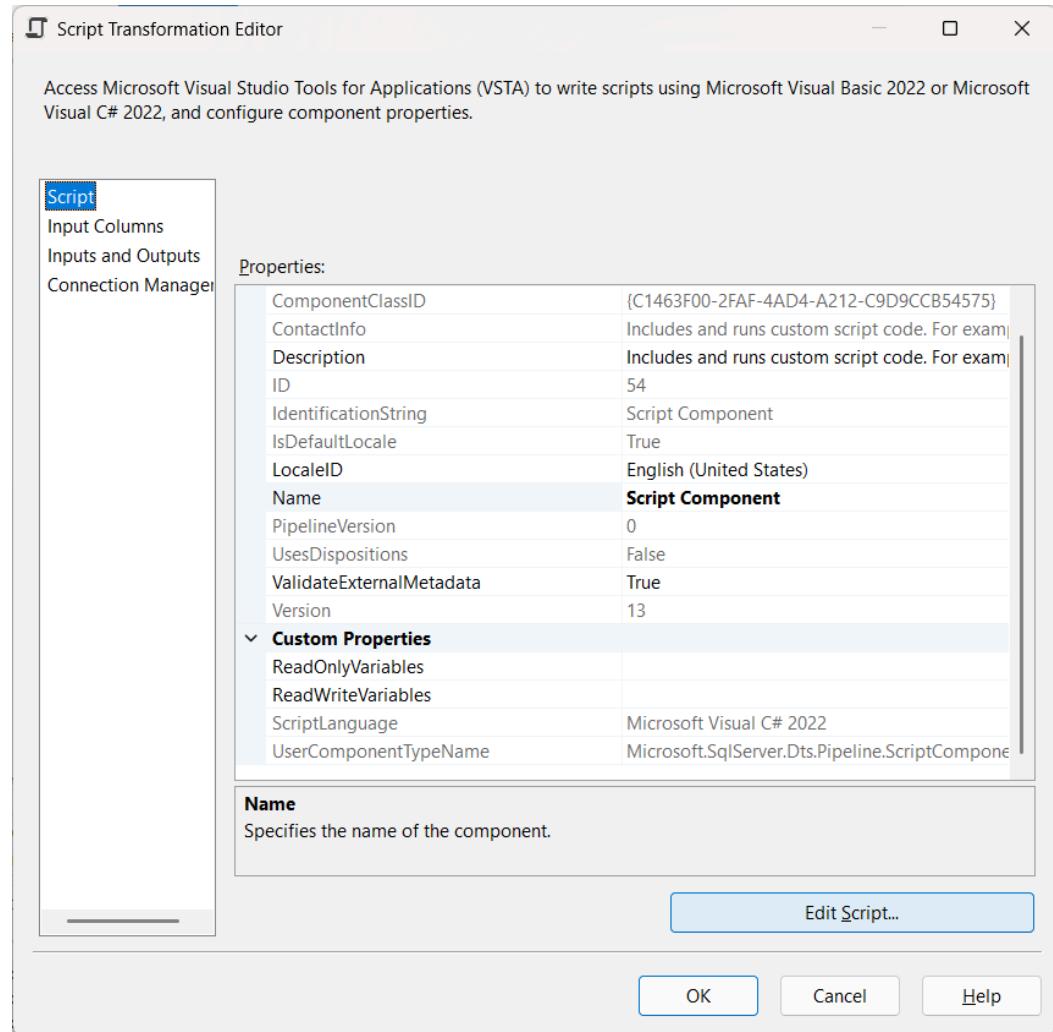
Hình 183. Lựa chọn các thuộc tính đầu vào cho Script Component

- **Bước 8:** Chuyển sang tab **Inputs and Outputs**, ta chọn **Add Output**, và đổi **Output** vừa thêm vào thành **OutputData**. Nhấn đúp chuột **OutputData**, nháy chuột vào **Output Columns**, chọn **Add Column** để thêm các cột khi xuất ra (**DateValue**: datetime, **Discount**: decimal, **DiscountID**: integer, **SubCategoryID**: integer, **CategoryID**: integer)



Hình 184. Tạo thuộc tính đầu ra cho Script Component

- **Bước 9:** Chuyển sang tab **Script**, chọn **Edit Script..**



Hình 185. Chọn Edit Script để tiến hành sửa đổi

- **Bước 10:** Một cửa sổ mới sẽ hiện lên, ta sửa đổi mã lệnh (C#) lại như sau

```

    public override void Input0_ProcessInputRow(Input0Buffer Row)
    {
        decimal discount = Row.Discount;
        if (discount == 0.0m || Row.Discount_IsNull)
        {
            // B7 qua ntu khuy?n m?i = 0
            return;
        }
        DateTime startDate = Row.Start;
        DateTime endDate = Row.End;

        // L?p t?ng ngay trong khong?
        for (DateTime d = startDate; d <= endDate; d = d.AddDays(1))
        {
            OutputDataBuffer.AddRow();
            OutputDataBuffer.DateValue = d;
            OutputDataBuffer.CategoryID = Row.CategoryID;
            OutputDataBuffer.SubCategoryID = Row.SubCategoryID;
            OutputDataBuffer.Discount = Row.Discount;
            OutputDataBuffer.DiscountID = Row.DiscountID;
        }
    }

    public override void CreateNewOutputRows()
    {
        /*
    
```

Hình 186. Lệnh C# để cân sửa đổi trong Script Component

### - Bước 11: Tiến hành build phần code đã sửa đổi

```

    public override void Input0_ProcessInputRow(Input0Buffer Row)
    {
        decimal discount = Row.Discount;
        if (discount == 0.0m || Row.Discount_IsNull)
        {
            // B7 qua ntu khuy?n m?i = 0
            return;
        }
        DateTime startDate = Row.Start;
        DateTime endDate = Row.End;

        // L?p t?ng ngay trong khong?
        for (DateTime d = startDate; d <= endDate; d = d.AddDays(1))
        {
            OutputDataBuffer.AddRow();
            OutputDataBuffer.DateValue = d;
            OutputDataBuffer.CategoryID = Row.CategoryID;
            OutputDataBuffer.SubCategoryID = Row.SubCategoryID;
            OutputDataBuffer.Discount = Row.Discount;
            OutputDataBuffer.DiscountID = Row.DiscountID;
        }
    }

    public override void CreateNewOutputRows()
    {
        /*
    
```

Hình 187. Tiến hành build script vừa sửa đổi

Kết quả khi build thành công

```

    This tool window can only be used by a SQL Server Integration Services package document.

main.cs  x  SC_e596f0f82740443ebd1ccfde591efc80  Input0_ProcessingRow(Input0Buffer Row)
110     /// Example of writing a value to a column in the row:
111     /// Row.ZipCode = ZipCode
112     /// <summary>
113     /// <param name="Row">The row that is currently passing through the component</param>
114     </summary>
115     public override void Input0_ProcessingRow(Input0Buffer Row)
116     {
117         decimal discount = Row.Discount;
118         if (discount == 0.0e+0 || Row.Discount_IsNull)
119         {
120             // B7 qua n?u khuy?n m?i = 0
121             return;
122         }
123         DateTime startDate = Row.Start;
124         DateTime endDate = Row.End;
125
126         // L?p t?m p?y trong kh?m
127         for (DateTime d = startDate; d <= endDate; d = d.AddDays(1))
128         {
129             OutputDataBuffer.AddRow();
130             OutputDataBuffer.DateValue = d;
131             OutputDataBuffer.CategoryID = Row.CategoryID;
132             OutputDataBuffer.SubCategoryID = Row.SubCategoryID;
133             OutputDataBuffer.Discount = Row.Discount;
134             OutputDataBuffer.DiscountID = Row.DiscountID;
135         }
136     }
137
138     public override void CreateNewOutputRows()
139     {
140     }

```

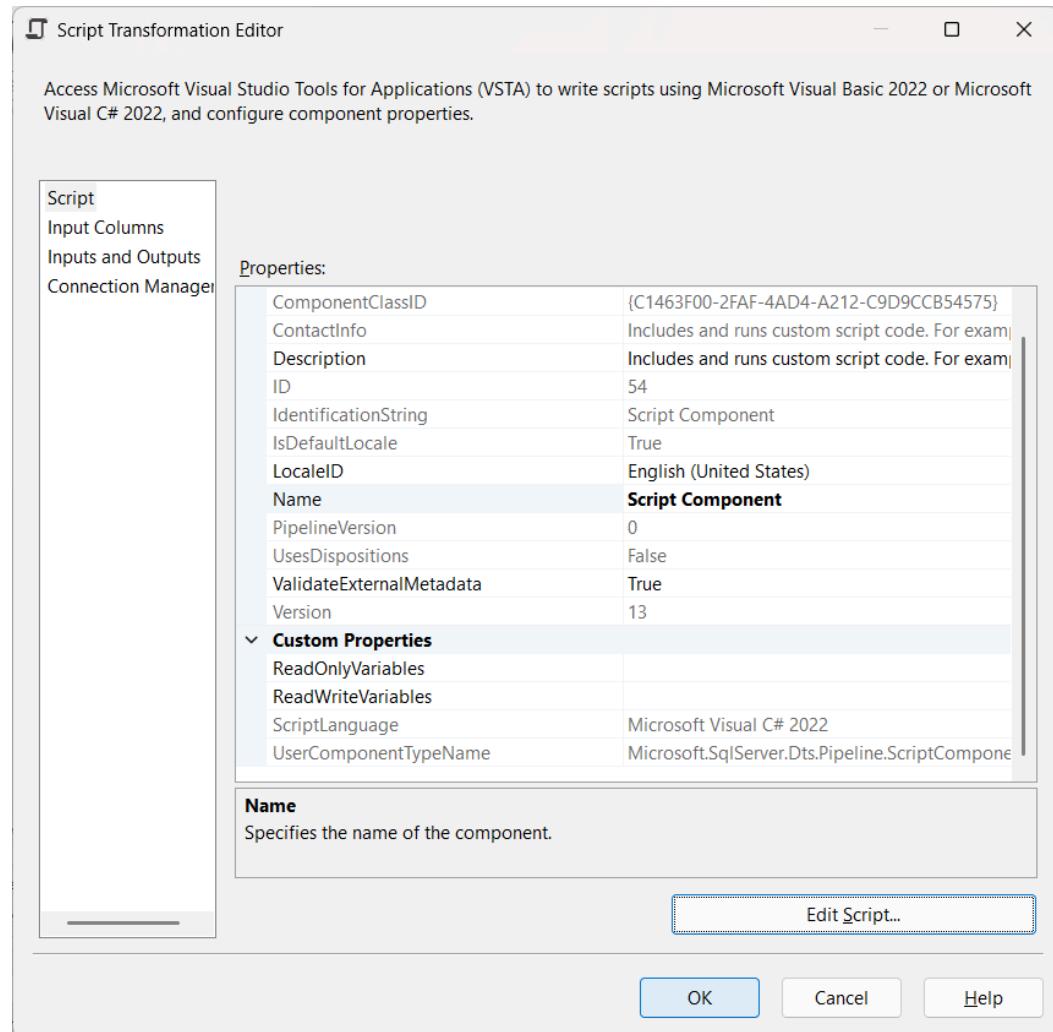
Output

Show output from: Build  
Build started: Project: SC\_e596f0f82740443ebd1ccfde591efc80  
Build succeeded

Build succeeded

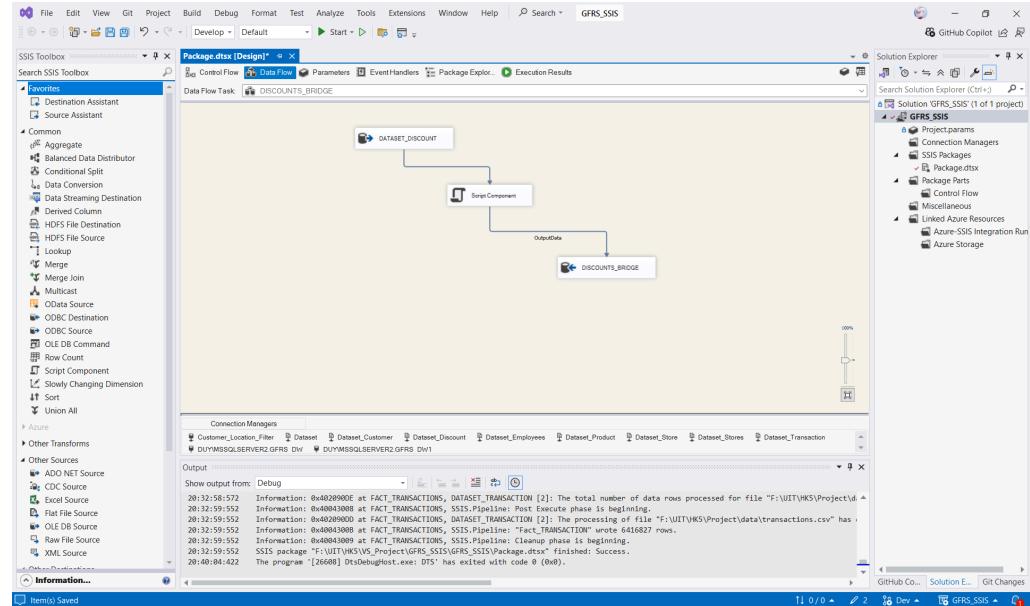
*Hình 188. Script đã được build thành công*

- **Bước 12:** Đóng cửa sổ code . Chọn **OK** và thoát cửa sổ **Script Transformation Editor**



Hình 189. Chọn OK để thoát cửa sổ Script Transformation Editor

- **Bước 13:** Kéo **ADO NET Destination** từ **SSIS Toolbox** vào **vùng Data Flow** và đổi tên thành **DISCOUNTS\_BRIDGE**. Kéo luồng dữ liệu từ **Script Component** vào **DISCOUNTS\_BRIDGE**. Một cửa sổ hiện lên, ở phần **Output**, ta chọn **OutputData**.



Hình 190. Tạo ADO.NET Destination DISCOUNTS\_BRIDGE

- **Bước 14:** Vào SQL Server 2022, chạy lệnh DDL để tạo DISCOUNTS\_BRIDGE.

```

-- Insert default row as in your original script (optional)
INSERT INTO Dim_Discounts([Start], [End], [Discount], [SubCategoryID], [CategoryID])
VALUES (NULL, NULL, 0.0, NULL, NULL);
SELECT * FROM Dim_Discounts;

CREATE TABLE [DISCOUNTS_BRIDGE] (
    [DateDiscountID] INT IDENTITY(1,1) PRIMARY KEY,
    [DiscountID] INT,
    [DateValue] DATETIME,
    [Discount] DECIMAL(10,2),
    [SubCategoryID] INT,
    [CategoryID] INT
)
GO

CREATE TABLE [dim_COLORS] (
    [ColorID] INT IDENTITY(1,1) PRIMARY KEY,
    [Color] VARCHAR(255)
)
GO

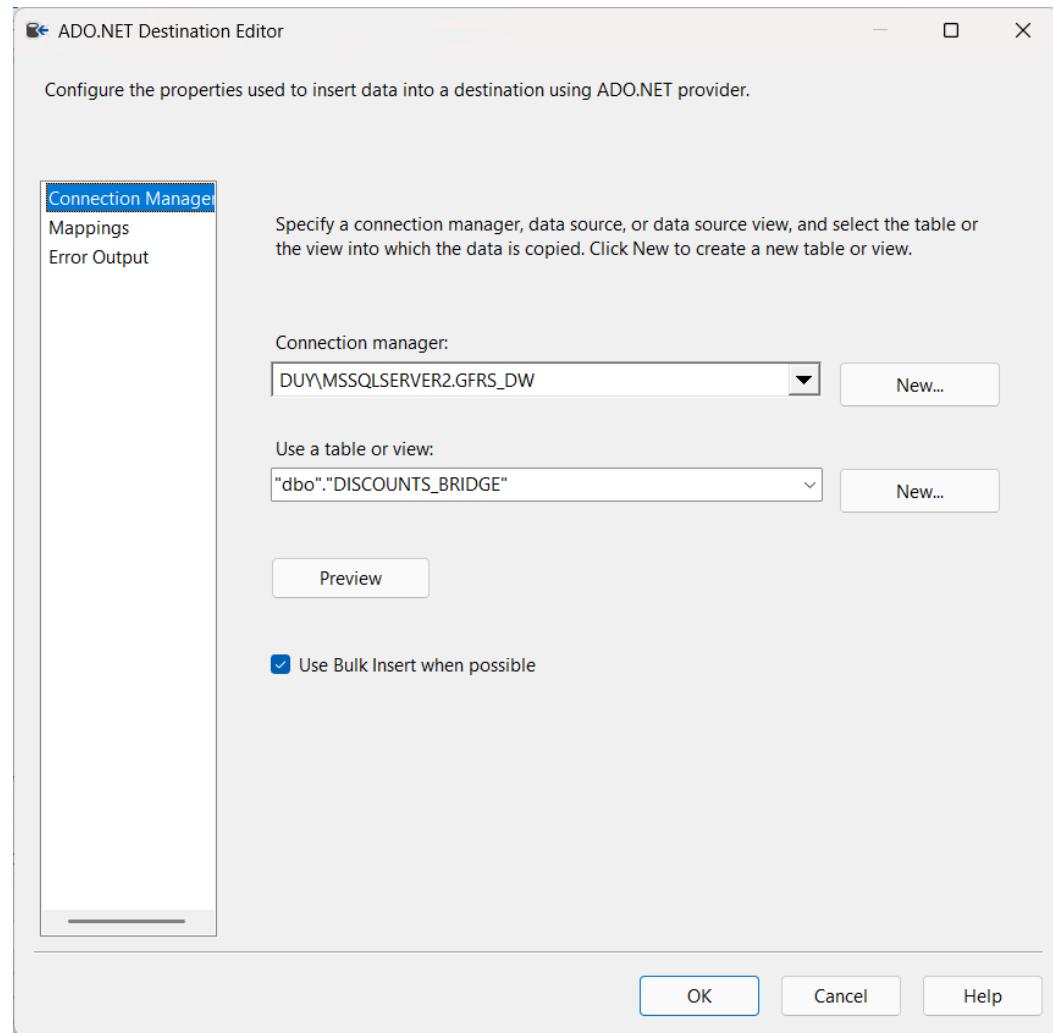
```

DiscountID	Start	End	Discount	Description	SubCategoryID	CategoryID
1	NULL	NULL	0.00	NULL	NULL	NULL
2	2020-10-01 00:00:00.000	2020-10-00 00:00:00.000	0.20	20% discount during our Autumn Essentials Sale	1	1
3	2020-10-01 00:00:00.000	2020-10-00 00:00:00.000	0.20	20% discount during our Autumn Essentials Sale	16	1
4	2020-10-01 00:00:00.000	2020-10-00 00:00:00.000	0.20	20% discount during our Autumn Essentials Sale	13	2
5	2020-10-01 00:00:00.000	2020-10-00 00:00:00.000	0.20	20% discount during our Autumn Essentials Sale	17	2
6	2020-10-01 00:00:00.000	2020-10-00 00:00:00.000	0.20	20% discount during our Autumn Essentials Sale	13	3
7	2020-10-01 00:00:00.000	2020-10-00 00:00:00.000	0.20	20% discount during our Autumn Essentials Sale	18	3
8	2020-10-01 00:00:00.000	2021-10-00 00:00:00.000	0.20	20% discount during our Autumn Essentials Sale	1	1
9	2020-10-01 00:00:00.000	2021-10-00 00:00:00.000	0.20	20% discount during our Autumn Essentials Sale	16	1
10	2020-10-01 00:00:00.000	2021-10-00 00:00:00.000	0.20	20% discount during our Autumn Essentials Sale	13	2
11	2020-10-01 00:00:00.000	2021-10-00 00:00:00.000	0.20	20% discount during our Autumn Essentials Sale	17	2
12	2020-10-01 00:00:00.000	2021-10-00 00:00:00.000	0.20	20% discount during our Autumn Essentials Sale	13	3

Hình 191. Chạy lệnh DDL tạo DISCOUNTS\_BRIDGE trong kho dữ liệu

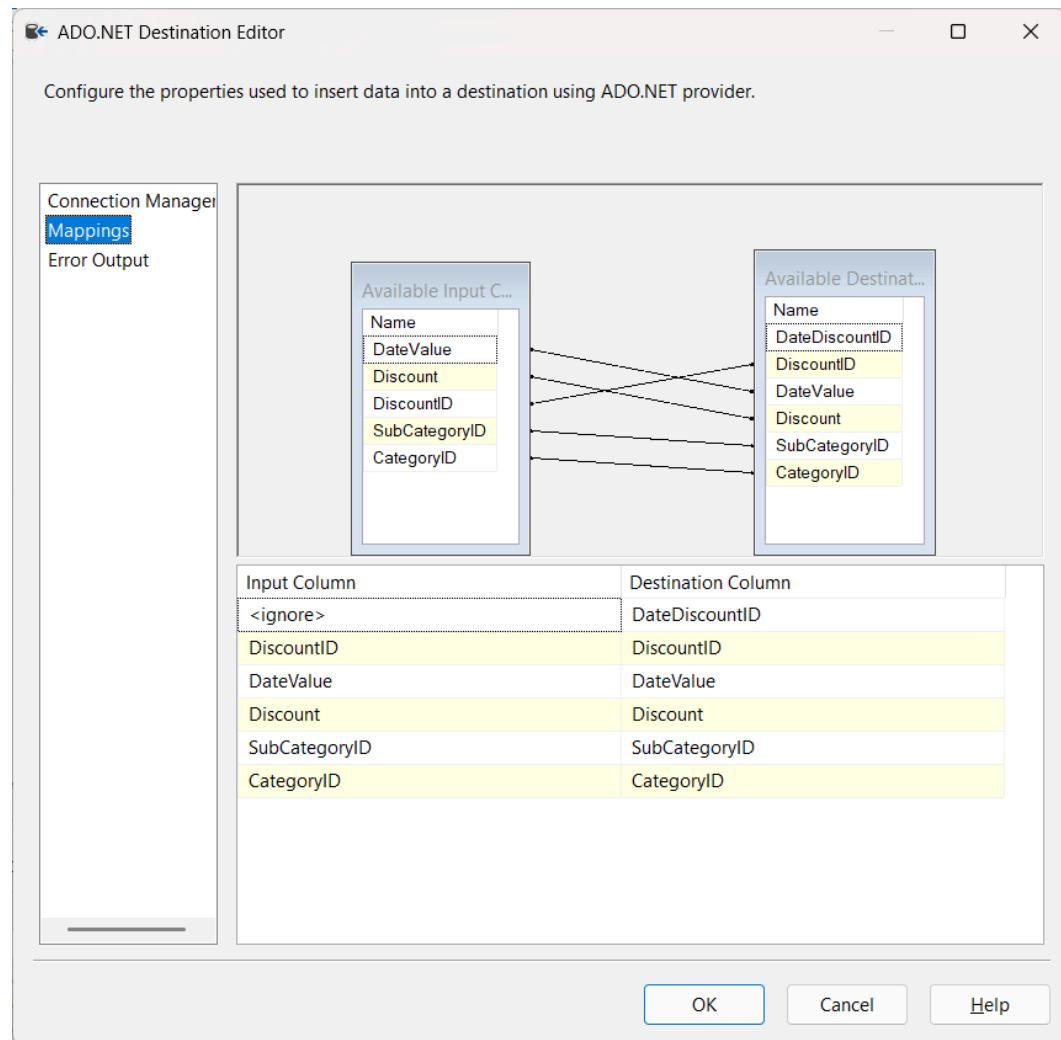
- **Bước 15 :**Nhấn đúp chuột vào DISCOUNTS\_BRIDGE.. Trong tab Connection Manager của màn hình ADO.NET Destination Editor, ở Use

a table or view tiến hành nhấn chọn “**dbo”.” DISCOUNTS\_BRIDGE.”**, cũng chính là bảng vừa được tạo bằng lệnh DDL ở **SQL Server 2022**



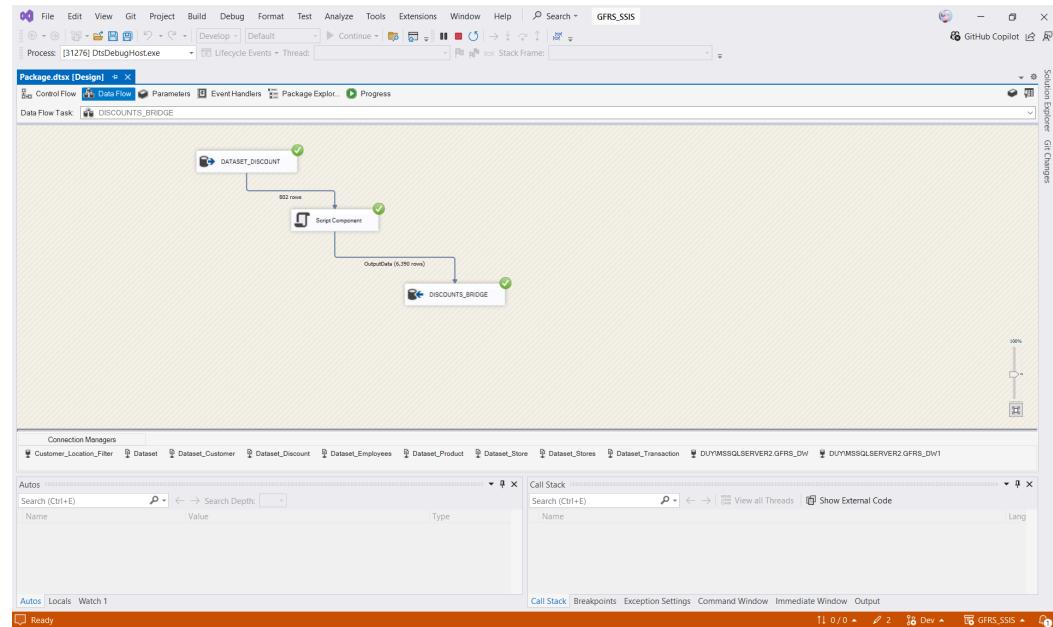
Hình 192. Kết nối đích đến vừa tạo cho ADO NET Destination

- **Bước 16:** Ở tab **Mapping** của màn hình **ADO.NET Destination Editor** tiến hành chọn các thuộc tính thích hợp ở cột bên trái để ánh xạ với các thuộc tính ở cột bên phải.



Hình 193. Cài đặt ADO.NET Destination DISCOUNTS\_BRIDGE

- **Bước 17:** Nhấn **Start** để kiểm tra quá trình tạo và đổ dữ liệu vào **DISCOUNTS\_BRIDGE** có thành công hay chưa.



Hình 194. Đỗ dữ liệu vào DISCOUNTS\_BRIDGE

Kết quả hiển thị cho thấy quá trình diễn ra thành công và có **6,390 rows** được đỗ vào **DISCOUNTS\_BRIDGE**.

- **Bước 18:** Vào SQL Server 2022 kiểm tra và thấy dữ liệu đã được đỗ vào **DISCOUNTS\_BRIDGE** thành công.

The screenshot shows the SQL Server Management Studio (SSMS) interface. The top menu bar includes File, Edit, View, Query, Git, Project, Tools, Extensions, Window, Help, and Search. The Object Explorer on the left shows the database structure for 'GFRS\_DW'. A query window titled 'SQLQuery1.s...\\maing (81)' is open, displaying the following T-SQL code:

```

55  INSERT INTO Dim_Discounts([Start], [End], [Discount], [SubCategoryID], [CategoryID])
56  VALUES (NULL, NULL, 0.0, NULL, NULL);
57
58  drop table if exists [DISCOUNTS_BRIDGE];
59  CREATE TABLE [DISCOUNTS_BRIDGE] (
60      [DateDiscountID] integer identity(1,1) PRIMARY KEY,
61      [DiscountID] integer,
62      [DateValue] datetime,
63      [Discount] decimal(10,2),
64      [SubCategoryID] integer,
65      [CategoryID] integer
66  )
67  select * from [DISCOUNTS_BRIDGE];
68  --select * from Dim_Discounts;
69  --select * from Fact_TRANSACTIONS;

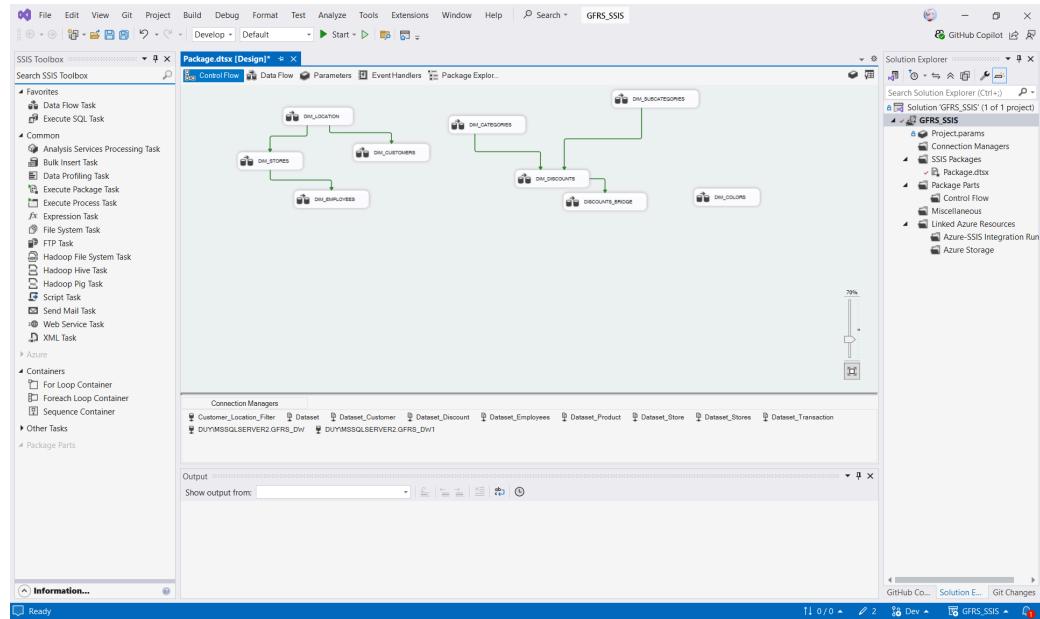
```

The results pane below the code shows a table with 11 rows of data, corresponding to the 6,390 rows inserted. The status bar at the bottom indicates 'Query executed successfully.' and other session details.

Hình 195. Dữ liệu được đỗ thành công vào DISCOUNTS\_BRIDGE

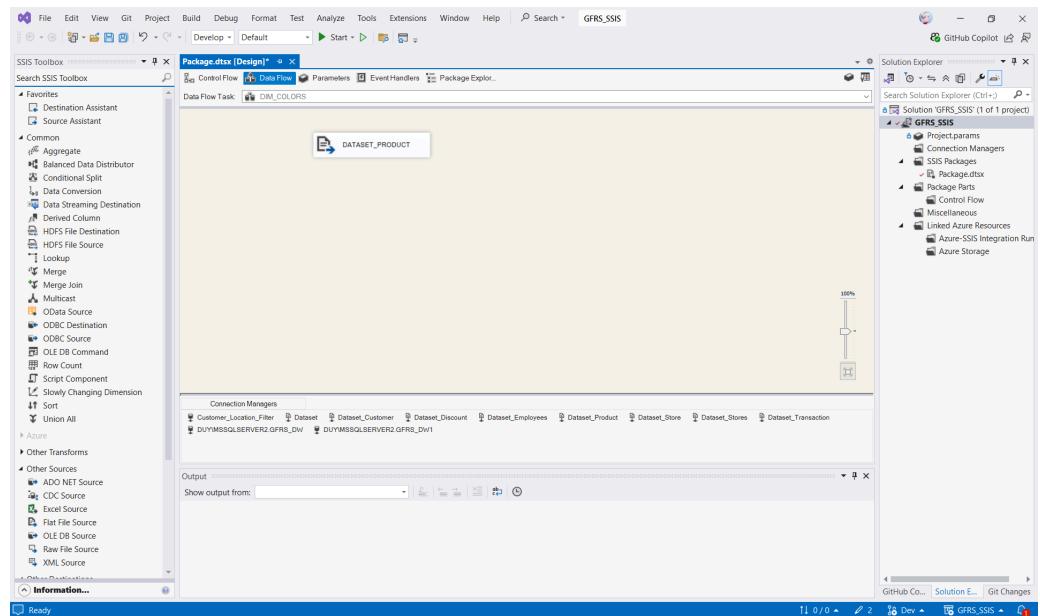
### II.3.8. Dim\_COLORS

- **Bước 1:** Kéo chức năng Data Flow Task trong vùng SSIS Toolbox vào vùng Control Flow và đổi tên thành **Dim\_COLORS**



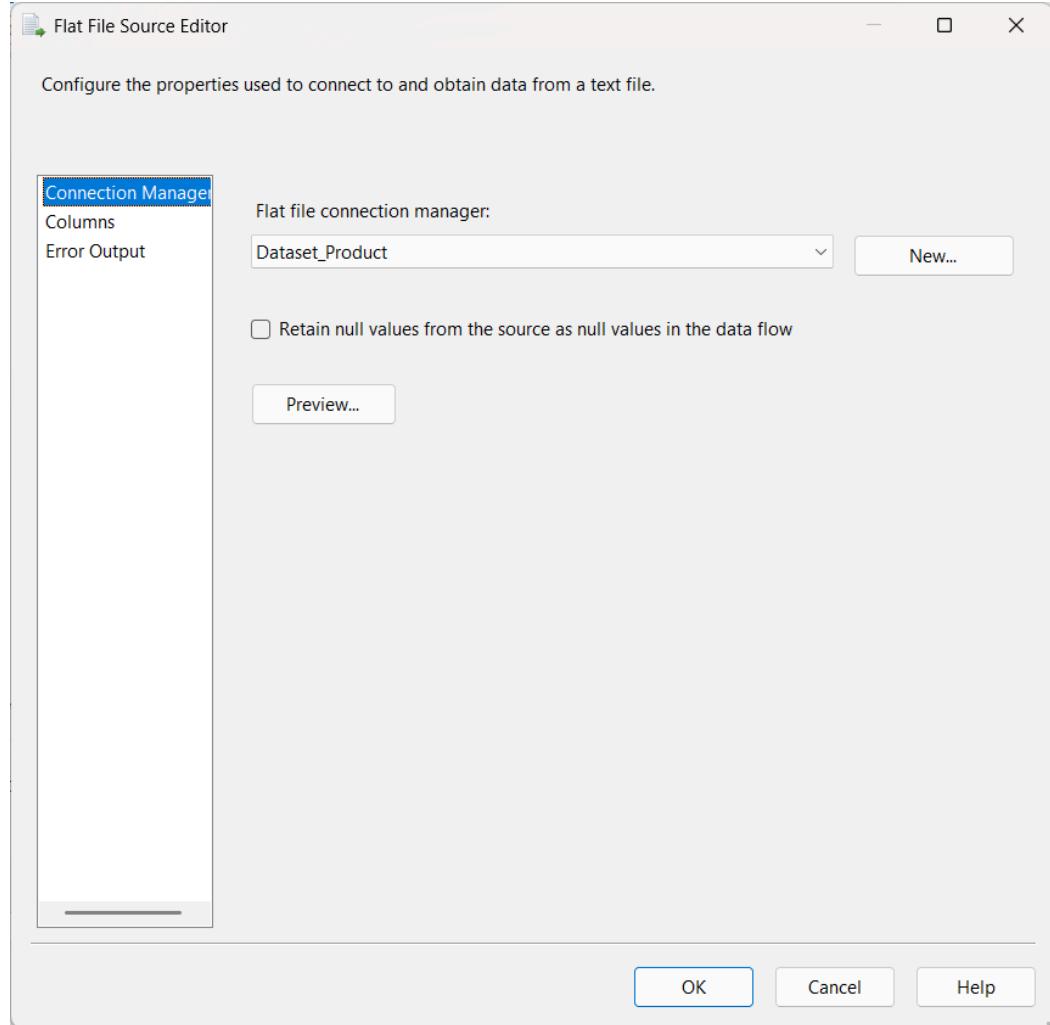
Hình 196. Tạo Dim\_COLORS trong Control Flow

- **Bước 2:** Nhấn đúp chuột vào **Dim\_COLORS**. Sau đó kéo chức năng Flat File Source vào vùng làm việc và đổi tên thành **DATASET\_PRODUCT**



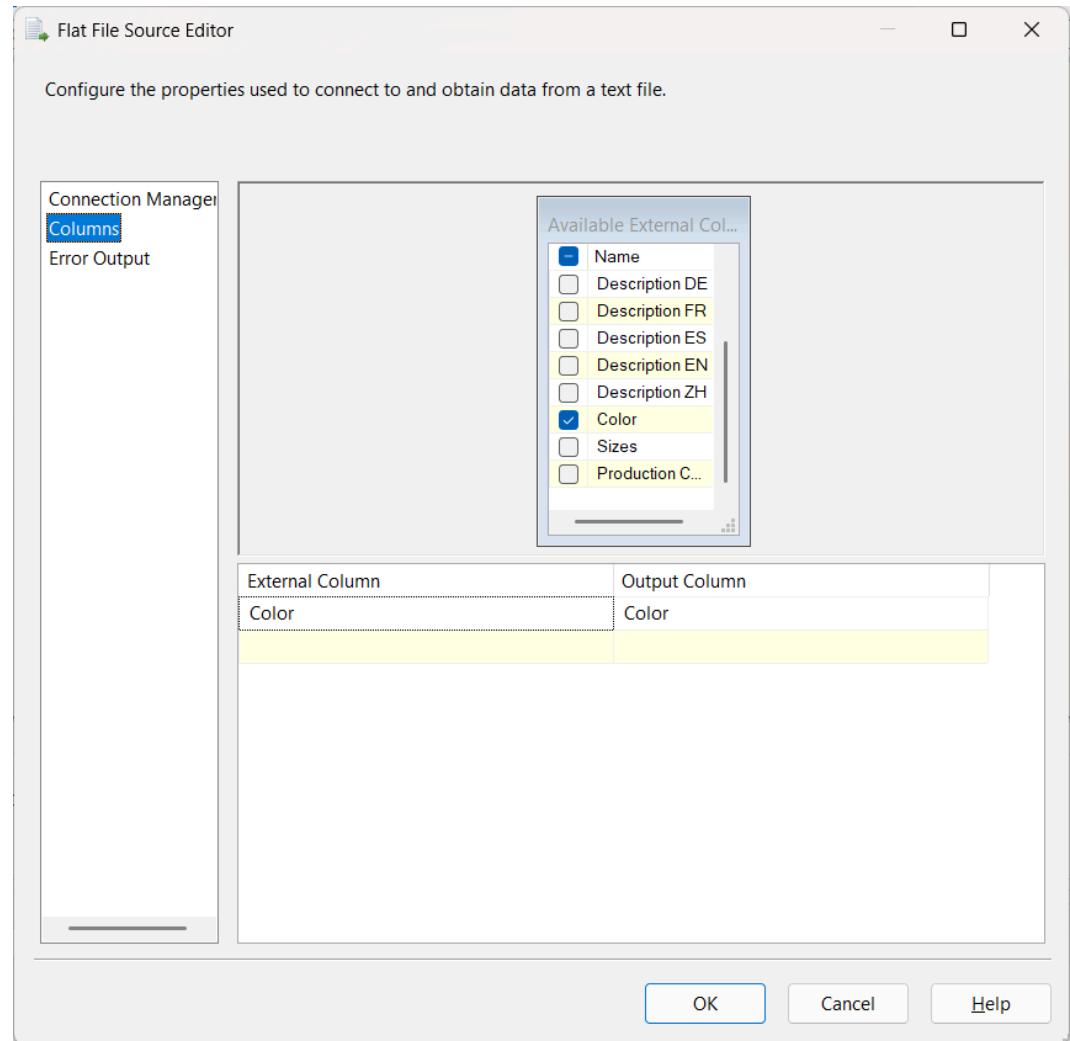
Hình 197. Tạo DATASET\_PRODUCT trong Dim\_COLORS

- **Bước 3:** Nhấn đúp chuột vào **DATASET\_PRODUCT**. Màn hình **Flat File Source Editor** hiện ra. Ở tab **Connection Manager** ta chọn **Dataset\_Product** đã được thiết lập ở phần trước



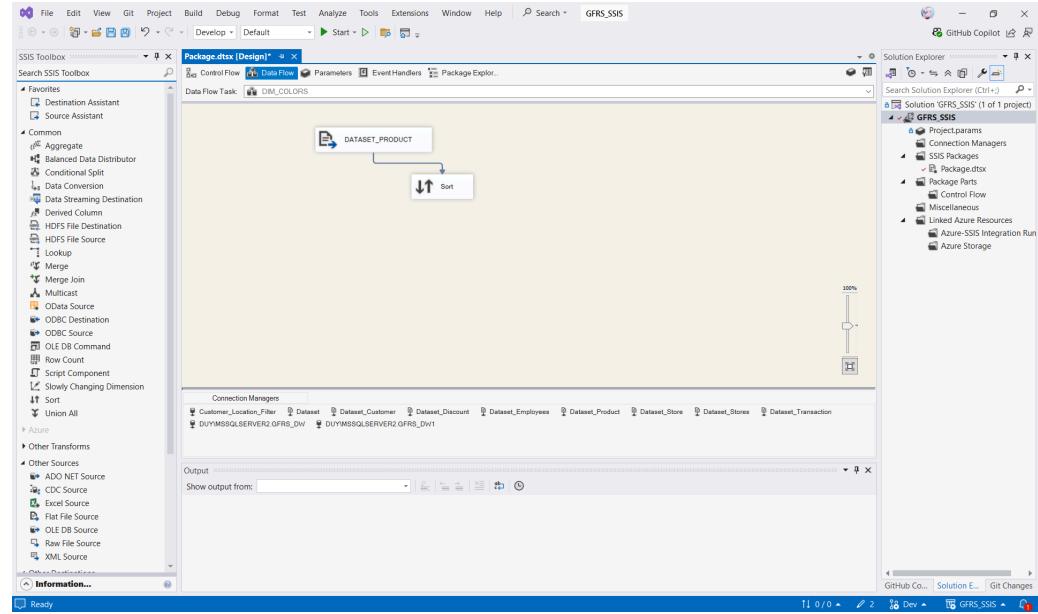
Hình 198. Chọn nguồn dữ liệu cho DATASET\_PRODUCT

- **Bước 4:** Ở tab **Columns** ta chọn cột **Color** để tiến hành dữ liệu vào **Dim\_COLORS**. Sau đó, nhấn **OK** và kết thúc cửa sổ **Flat File Source Editor**



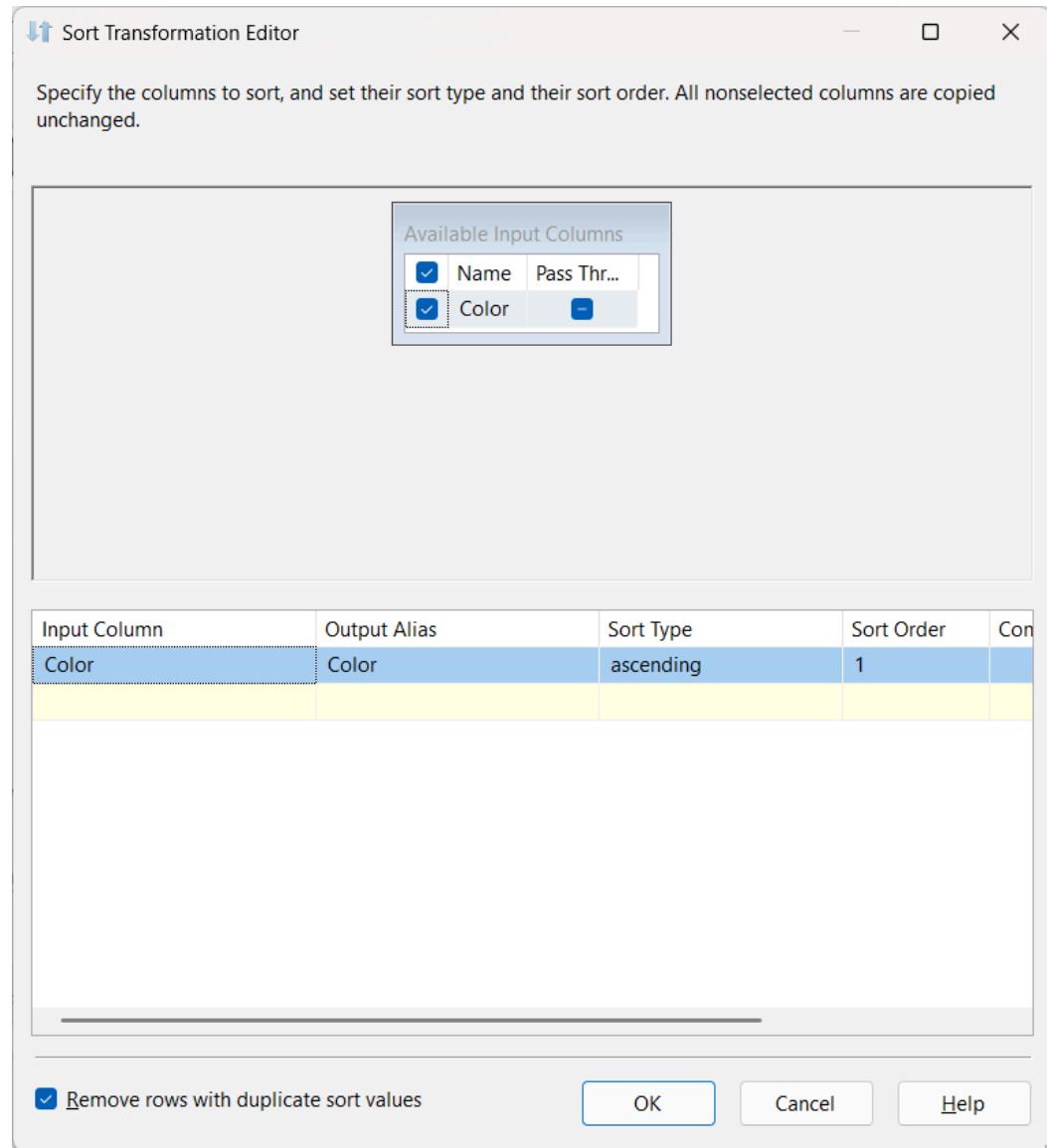
Hình 199. Lựa chọn thuộc tính cần thiết cho Dim\_COLORS

- **Bước 5:** Kéo chức năng Sort từ vùng SSIS Toolbox vào cùng Data Flow. Kéo luồng dữ liệu từ DATASET\_PRODUCT vào Sort.



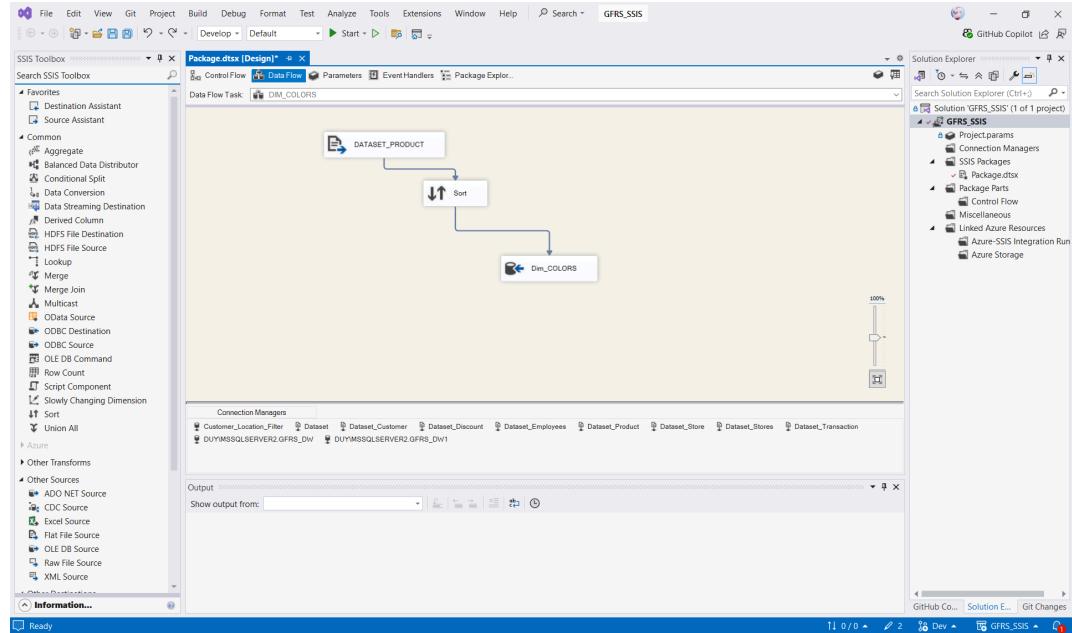
Hình 200. Tạo chức năng Sort trong Dim\_COLORS

- **Bước 6:** Nhấn đúp chuột vào **Sort**, cửa sổ **Sort Transformation Editor** sẽ hiện lên. Ta chọn cột **Color** trên cửa sổ **Available Input Columns**. Sau đó chọn **Remove rows with duplicate sort values** để lọc những màu sắc trùng nhau. Sau đó nhấn **OK** và kết thúc cửa sổ **Sort Transformation Editor**.



Hình 201. Cài đặt chức năng Sort trong Dim\_COLORS

- **Bước 7:** Kéo ADO NET Destination từ SSIS Toolbox vào vùng Data Flow và đổi tên thành **Dim\_COLORS**. Kéo luồng dữ liệu từ Sort vào **Dim\_COLORS**.



Hình 202. Tạo chức năng ADO.NET Destination Dim\_COLORS

- **Bước 8:** Vào SQL Server 2022, chạy lệnh DDL để tạo Dim\_COLORS.

```

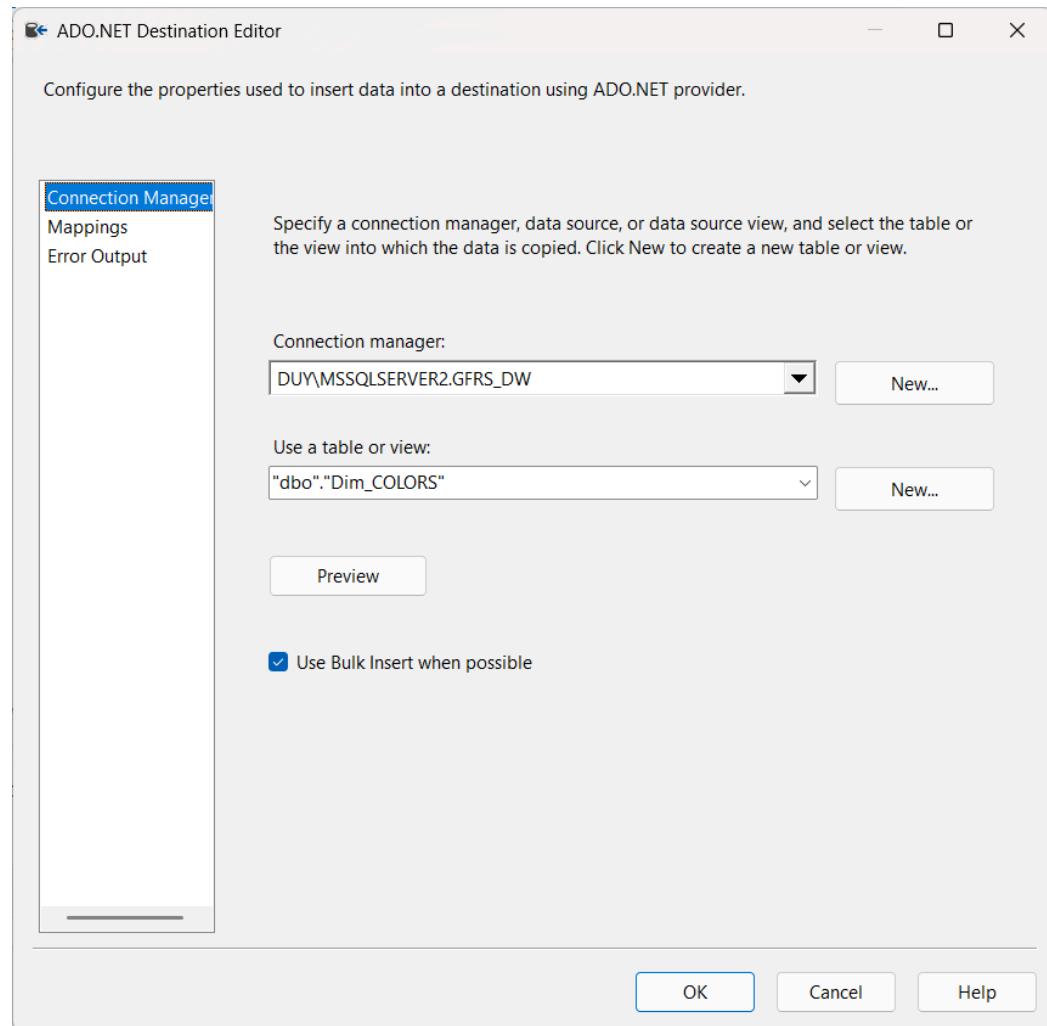
65      [CategoryID] integer
66
67      --select * from [DISCOUNTS_BRIDGE];
68      --select * from Dim_DISCOUNTS;
69      --select * from Fact_TRANSACTIONS;
70      drop table if exists Dim_COLORS;
71      CREATE TABLE [Dim_COLORS] (
72          [ColorID] integer identity(1,1) PRIMARY KEY,
73          [Color] varchar(255)
74      )
75
76      drop table if exists Dim_SIZES;
77      CREATE TABLE [Dim_SIZES] (
78          [SizeID] integer identity(1,1) PRIMARY KEY,
79          [Size] varchar(255)
80      )
81
82      drop table if exists Dim_PRODUCTS;
83      CREATE TABLE Dim_PRODUCTS (
84          [ProductID] integer identity(1,1) PRIMARY KEY,
85          ProductHash integer,
86          [CategoryID] integer,
87          [ColorID] integer,
88          [SubCategoryID] integer,
89          [SizeID] integer,
90          [ProductionCost] decimal

```

Hình 203. Chạy lệnh DDL tạo Dim\_COLORS trong kho dữ liệu

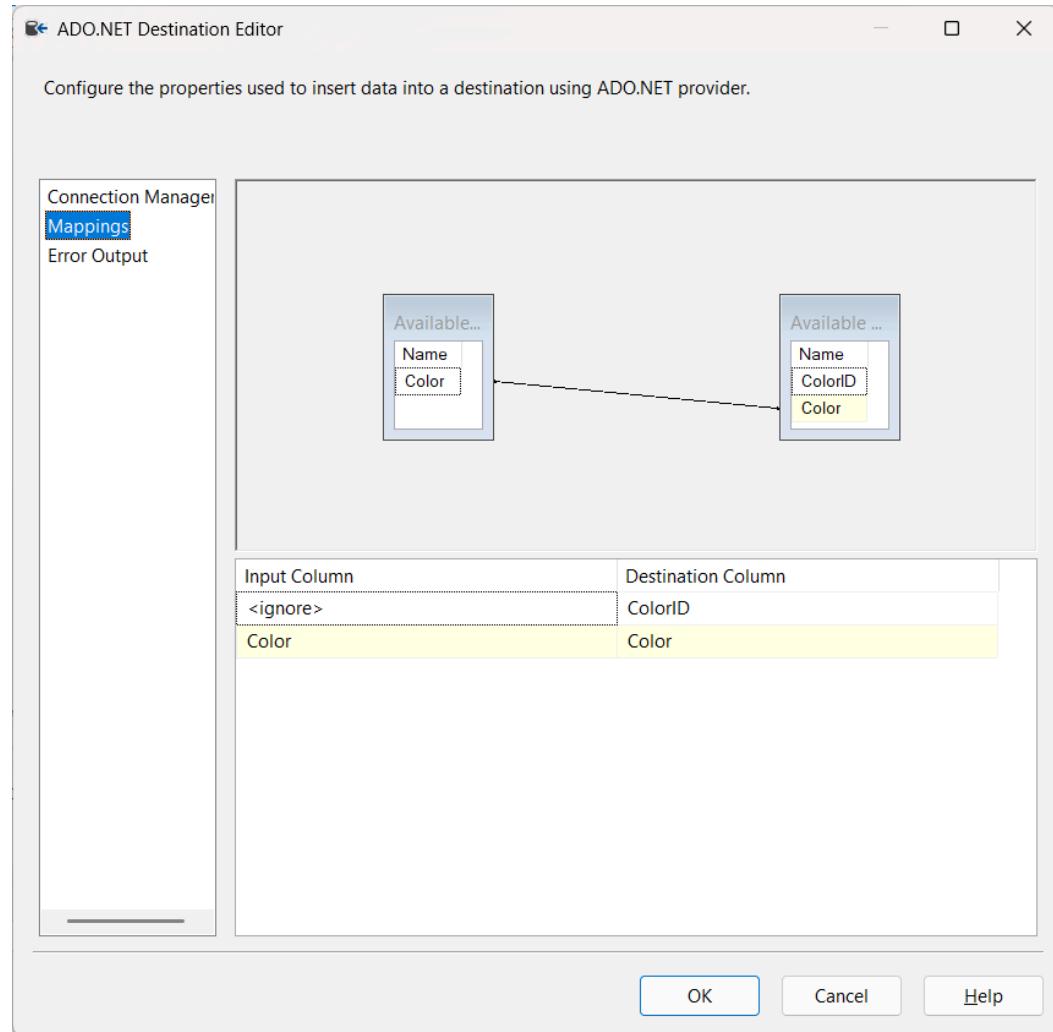
- **Bước 9:** Nhấn đúp chuột vào Dim\_COLORS.. Trong tab Connection Manager của màn hình ADO.NET Destination Editor, ở Use a table or

view tiến hành nhấn chọn “**dbo”.” Dim\_COLORS.”**, cũng chính là bảng vừa được tạo bằng lệnh **DDL** ở **SQL Server 2022**



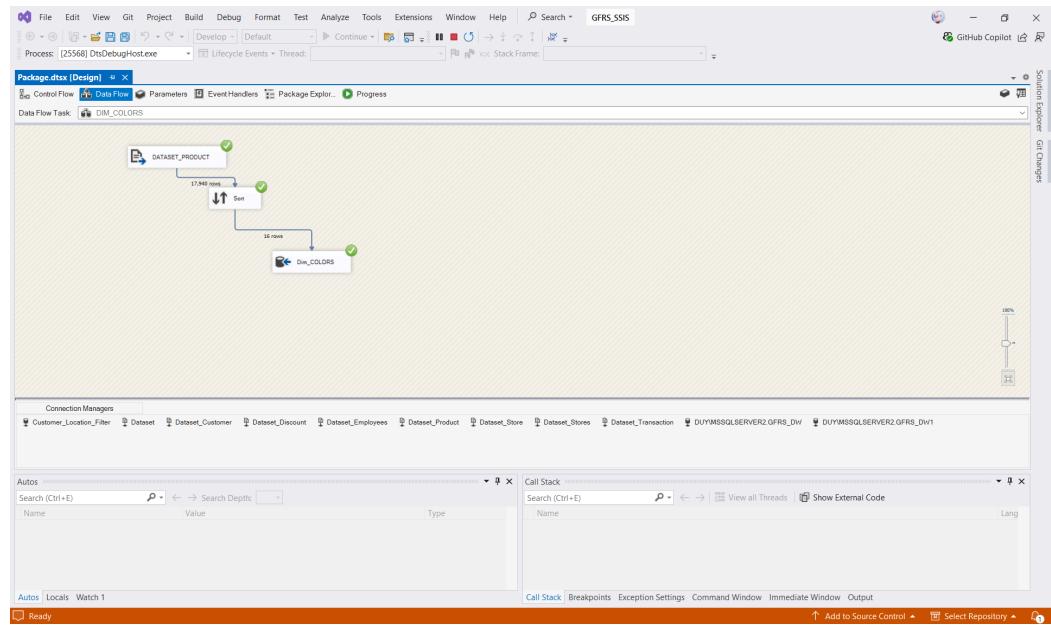
Hình 204. Kết nối đích đến cho ADO NET Destination Dim\_COLORS

- **Bước 10:** Ở tab **Mapping** của màn hình **ADO.NET Destination Editor** tiến hành chọn các thuộc tính thích hợp ở cột bên trái để ánh xạ với các thuộc tính ở cột bên phải.



Hình 205. Cài đặt ADO NET Destination Dim\_COLORS

- **Bước 11:** Nhấn Start để kiểm tra quá trình tạo và đỗ dữ liệu vào Dim\_COLORS có thành công hay chưa.



Hình 206. Đổ dữ liệu vào Dim\_COLORS

Kết quả hiển thị cho thấy quá trình diễn ra thành công và có **16 rows** được đổ vào **Dim\_COLORS**.

- **Bước 12:** Vào SQL Server 2022 kiểm tra và thấy dữ liệu đã được đổ vào Dim\_COLORS thành công.

The screenshot shows the SQL Server Management Studio (SSMS). In the Object Explorer, the connection is to 'GFRS\_DW'. In the center, a query window displays the creation of the 'Dim\_COLORS' table and a select statement to verify its contents. The results pane shows the following data:

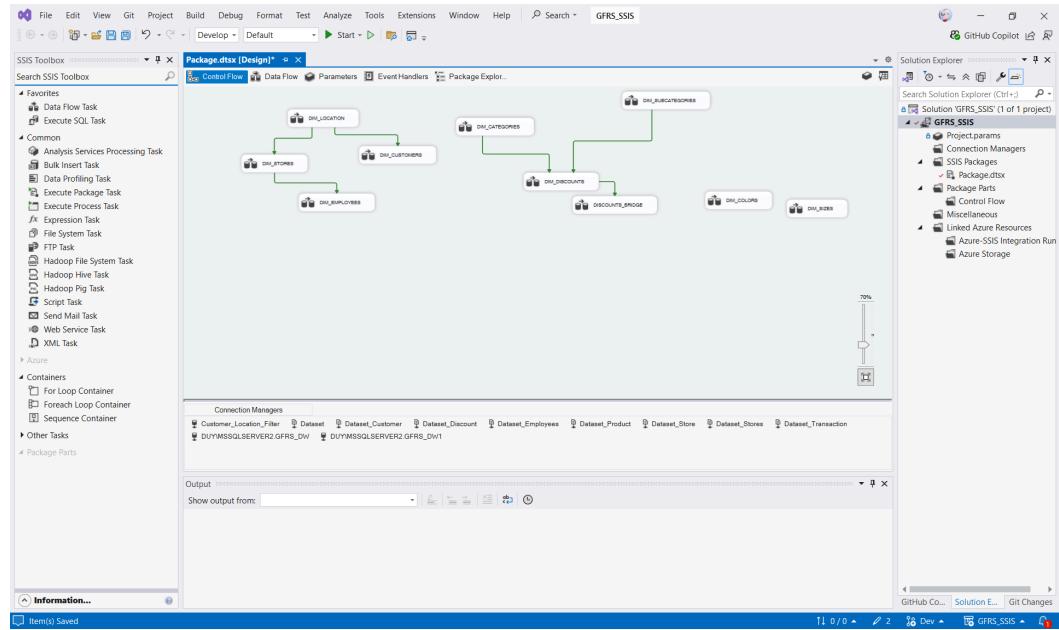
ColorID	Color
1	BEIGE
2	BLACK
3	BLUE
5	BURGUNDY
6	GOLD
7	GREEN
8	LILAC
9	MUSTARD
10	NEUTRAL
11	PINK
12	RED
13	SILVER
14	TURQUOISE

At the bottom, a message indicates 'Query executed successfully.'

Hình 207. Dữ liệu được đổ thành công vào Dim\_COLORS

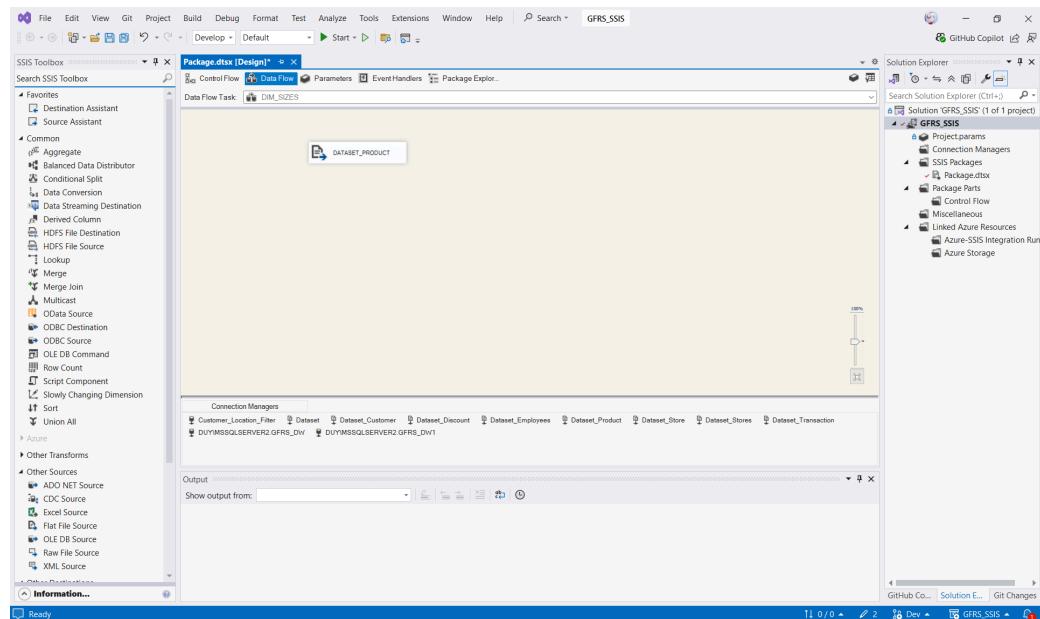
### II.3.9. Dim\_SIZES

- **Bước 1:** Kéo chức năng Data Flow Task trong vùng SSIS Toolbox vào vùng Control Flow và đổi tên thành Dim\_SIZES



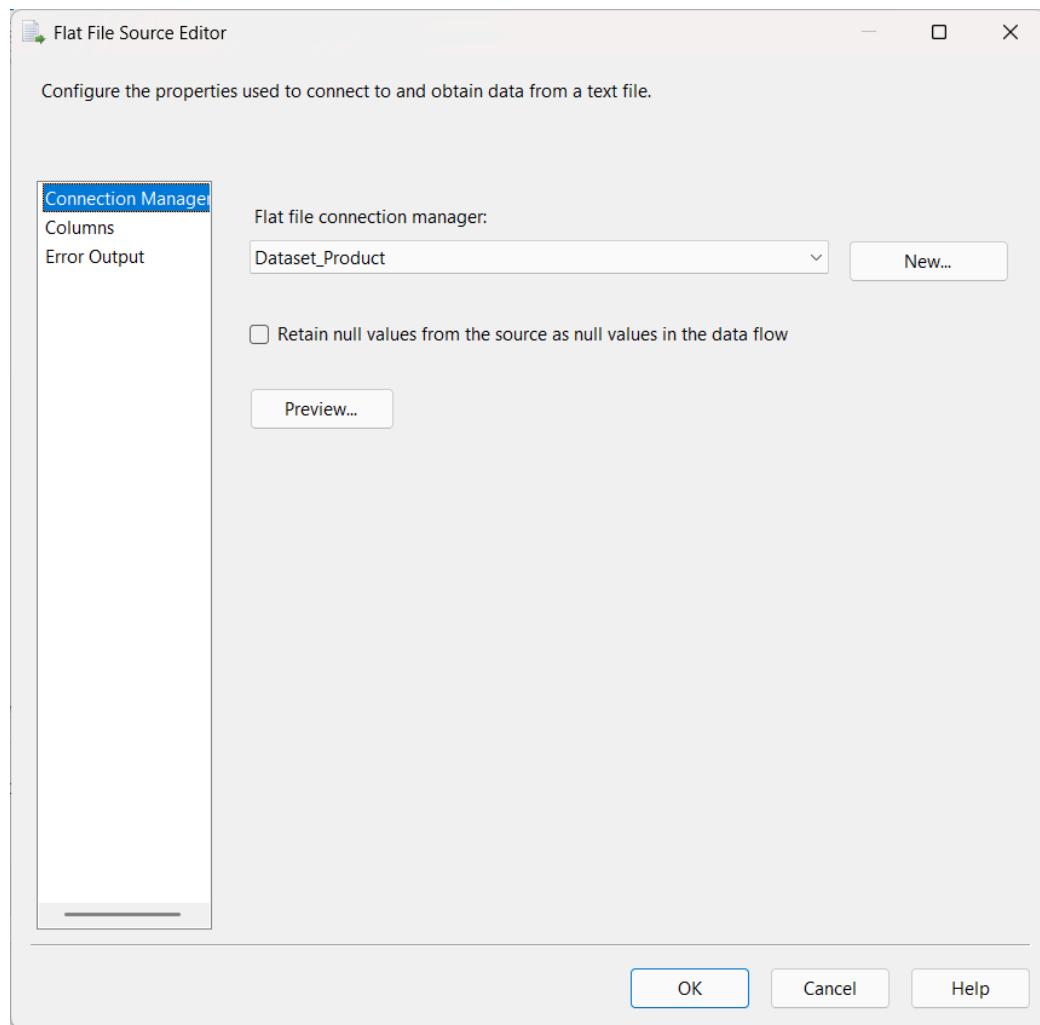
Hình 208. Tạo Dim\_SIZES trong Control Flow

- **Bước 2:** Nhấn đúp chuột vào Dim\_SIZES. Sau đó kéo chức năng Flat File Source vào vùng làm việc và đổi tên thành DATASET\_PRODUCT



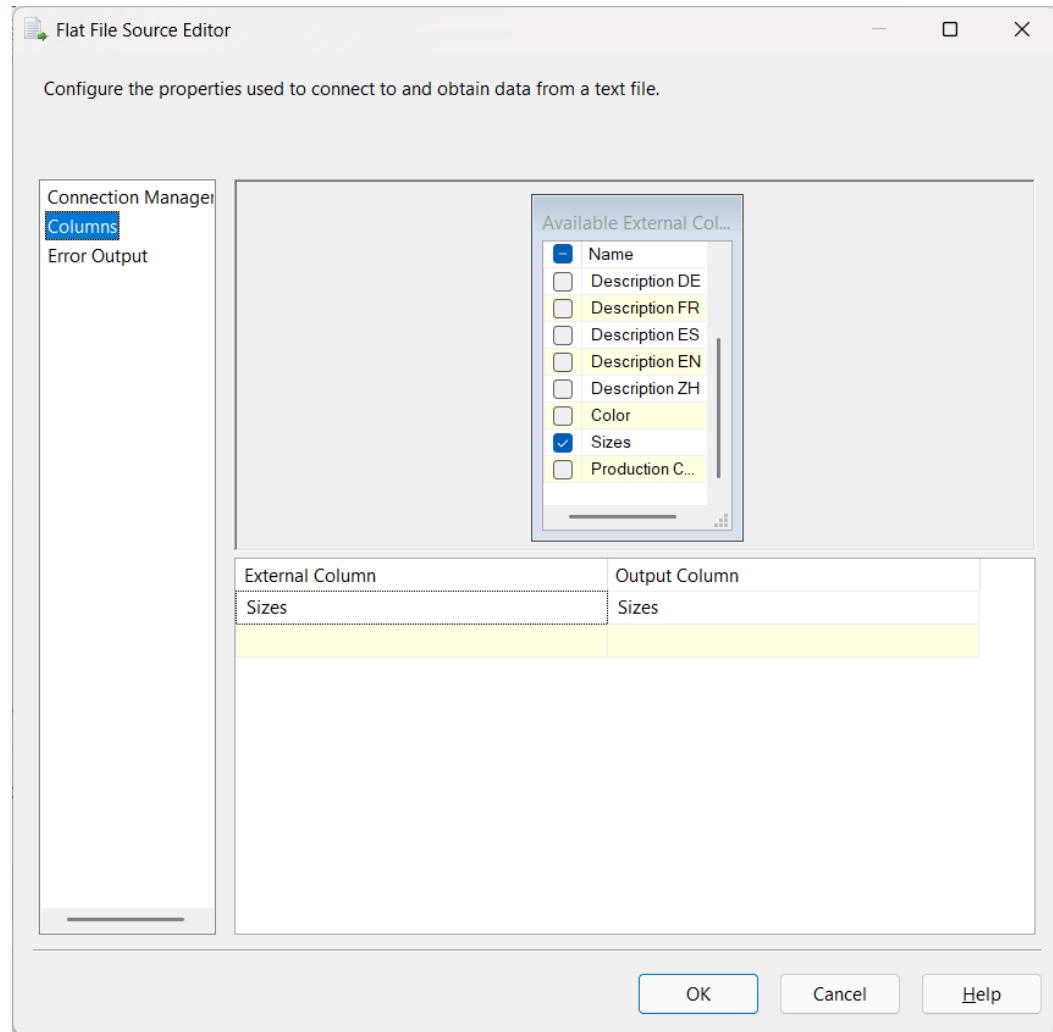
Hình 209. Tạo DATASET\_PRODUCT trong Dim\_SIZES

- **Bước 3:** Nhấn đúp chuột vào **DATASET\_PRODUCT**. Màn hình **Flat File Source Editor** hiện ra. Ở các bước trước đã load thành công file csv và đổi tên thành **Dataset\_Discount** nên ta có thể sử dụng lại **Dataset\_Discount** để load dữ liệu mà không cần thiết lập lại giúp tiết kiệm thời gian và giảm nguy cơ gặp lỗi.



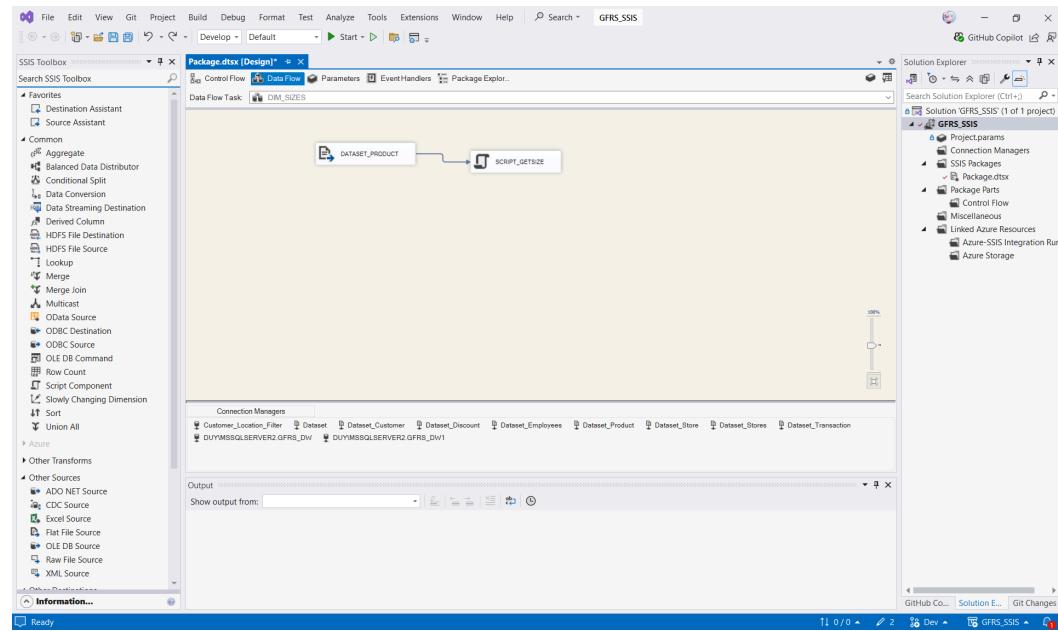
*Hình 210. Chọn nguồn dữ liệu cho DATASET\_PRODUCT*

- **Bước 4:** Chuyển sang tab **Columns**, ta có thể lựa chọn các thuộc tính cần thiết cho quá trình nạp dữ liệu cho **Dim\_SIZES**. Sau đó ta có thể bấm **OK** để đóng cửa sổ **Flat File Source Editor**



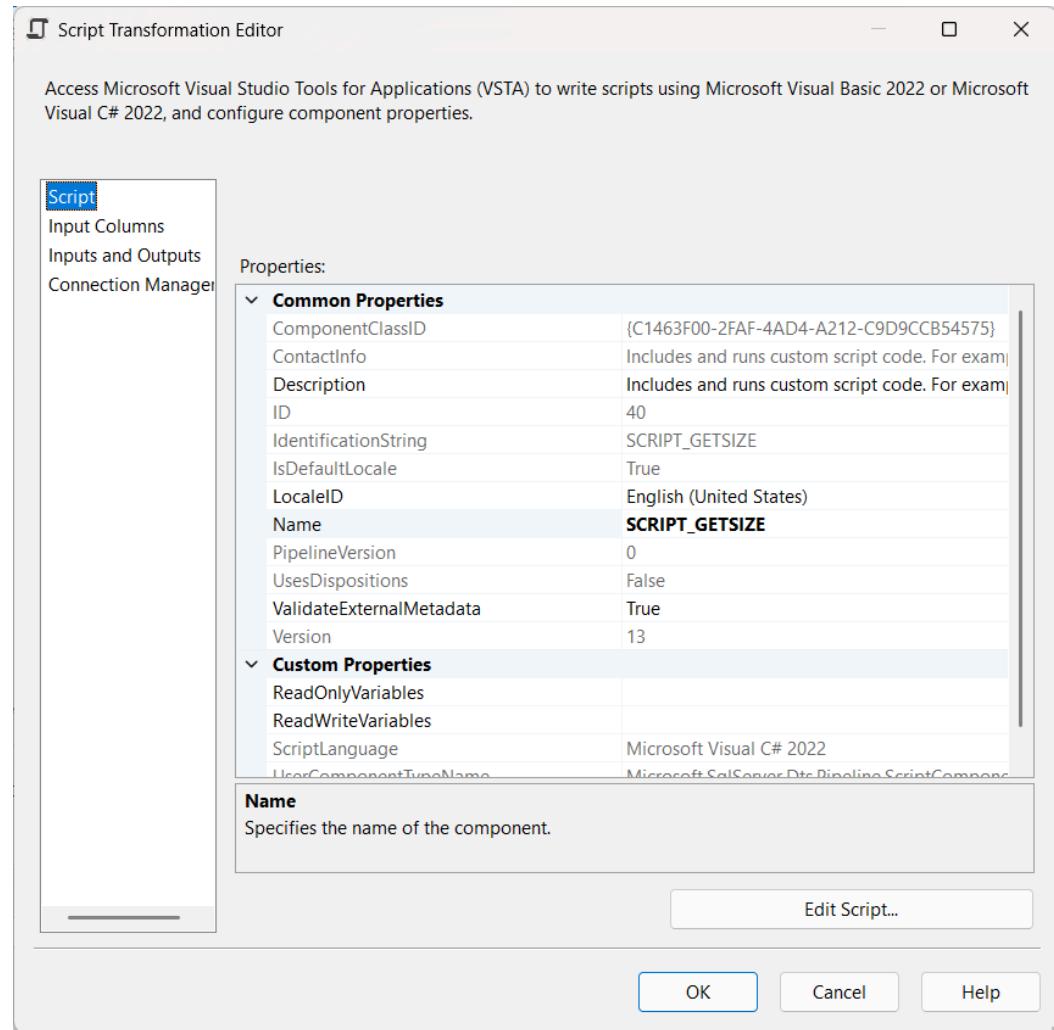
Hình 211. Chọn thuộc tính cần thiết cho quá trình Dim\_SIZES

- **Bước 5:** Kéo **Script Component** từ cùng **SSIS Toolbox** vào vùng **Data Flow** và đổi tên thành **SCRIPT\_GETSIZE**. Sau đó kéo luồng dữ liệu từ **DATASET\_PRODUCT** vào **SCRIPT\_GETSIZE**. (Vì dữ liệu thuộc dạng “34|35|36” nên ta cần tách riêng chúng ra thành những rows riêng biệt)



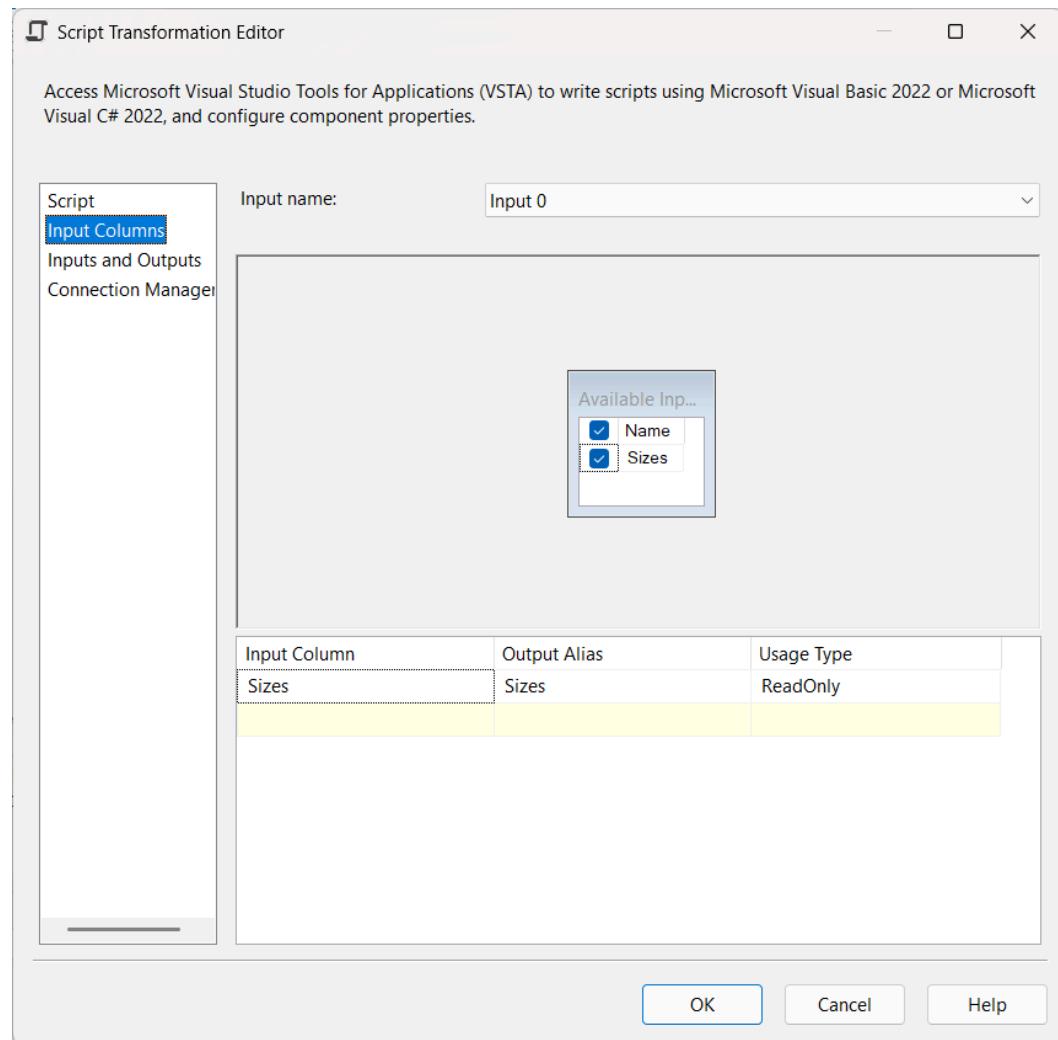
Hình 212. Tạo chức năng Script Component trong Dim\_SIZE

- **Bước 6:** Nhấn đúp chuột vào **SCRIPT\_GETSIZE**, cửa sổ **Script Transformation Editor** hiện lên.



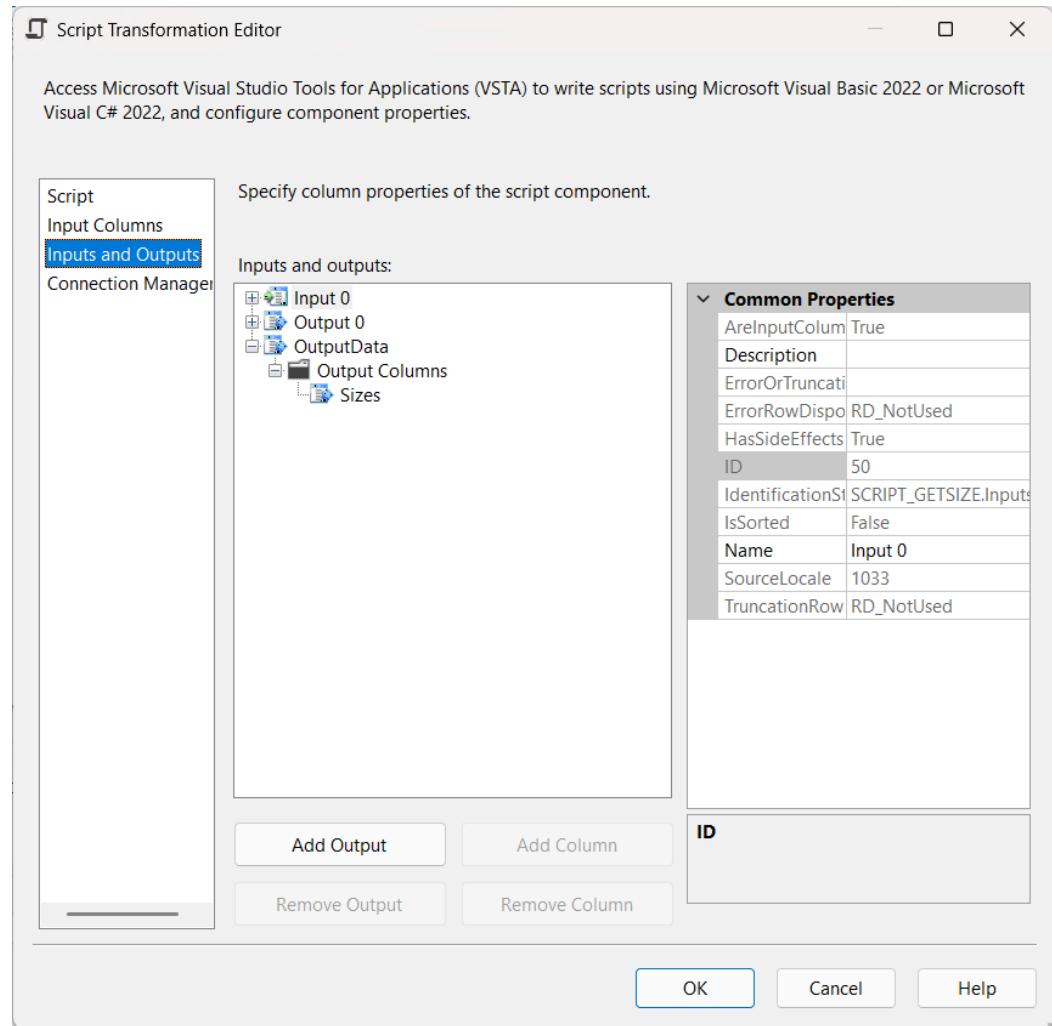
Hình 213. Kiểm tra cấu hình `SCRIPT_GETSIZE`

- **Bước 7:** Chuyển sang tab **Input Columns**, ta sẽ chọn cột **Sizes**



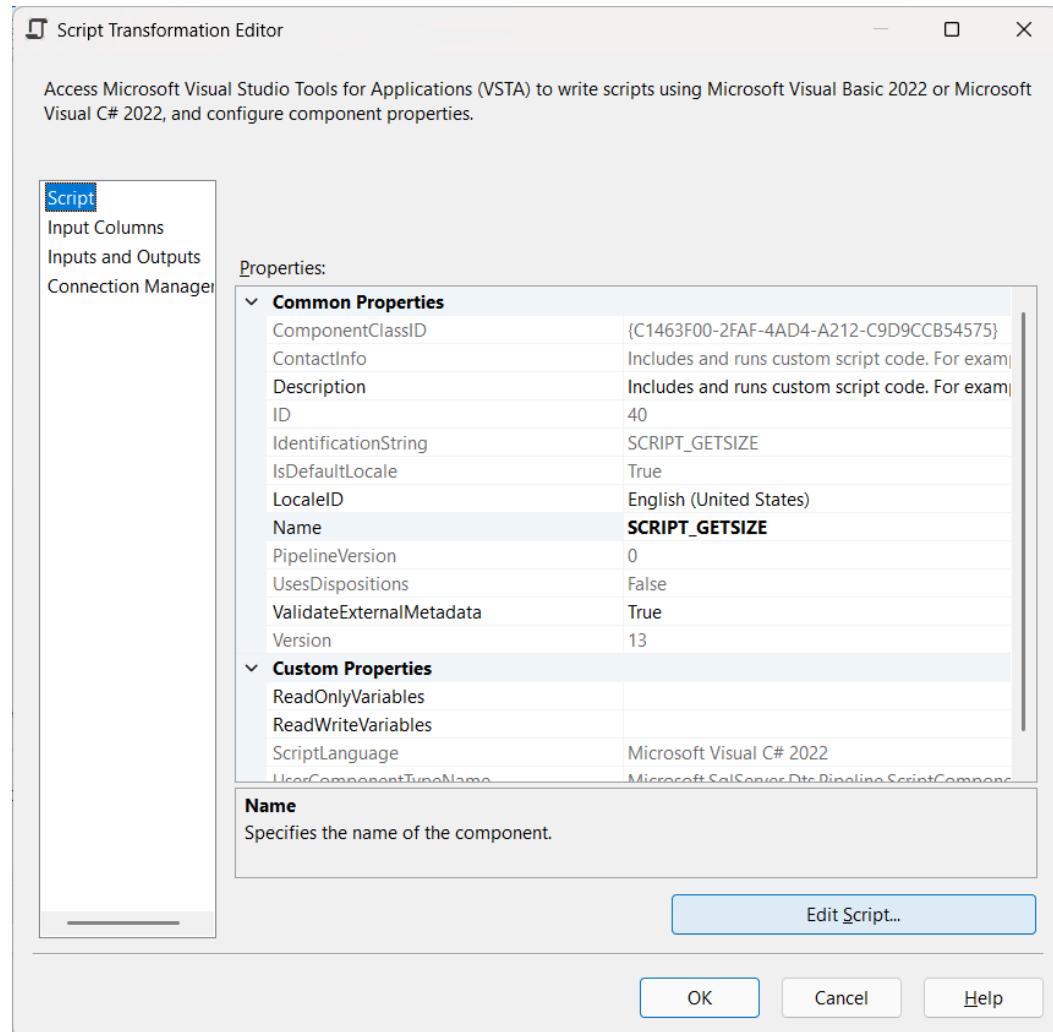
Hình 214. Chọn thuộc tính đầu vào cho `SCRIPT_GETSIZE`

- **Bước 8:** Chuyển sang tab **Inputs and Outputs**, ta chọn **Add Output**, và đổi **Output** vừa thêm vào thành **OutputData**. Nhấn đúp chuột **OutputData**, nháy chuột vào **Output Columns**, chọn **Add Column** để thêm các cột khi xuất ra (**Sizes: String**)



Hình 215. Cài đặt thuộc tính đầu ra cho SCRIPT\_GETSIZE

- **Bước 9:** Chuyển sang tab **Script**, chọn **Edit Script..**



Hình 216. Chọn Edit Script để sửa đổi câu lệnh

- **Bước 10:** Một cửa sổ mới sẽ hiện lên, ta sửa đổi mã lệnh (C#) lại như sau

```

108     /// <summary>
109     /// This method is called once for every row that passes through the component from Input0.
110     /// Example of reading a value from a column in the the row:
111     /// string zipcode = Row.ZipCode
112     /// Row.ZipCode = zipCode
113     /// </summary>
114     /// <param name="Row">The row that is currently passing through the component</param>
115
116     public override void Input0_ProcessInputRow(Input0Buffer Row)
117     {
118         if (!Row.Sizes_IsNull)
119         {
120             string[] sizes = Row.Sizes.Split(',');
121
122             foreach (string s in sizes)
123             {
124                 OutputDataBuffer.AddRow();
125                 OutputDataBuffer.Sizes = s.Trim();
126             }
127         }
128     }
129
130 }
131

```

Hình 217. Sửa đổi câu lệnh C# để lấy kích cỡ

- **Bước 11:** Tiến hành build phần code đã sửa đổi

```

108     /// <summary>
109     /// This method is called once for every row that passes through the component from Input0.
110     /// Example of reading a value from a column in the the row:
111     /// string zipcode = Row.ZipCode
112     /// Row.ZipCode = zipCode
113     /// </summary>
114     /// <param name="Row">The row that is currently passing through the component</param>
115
116     public override void Input0_ProcessInputRow(Input0Buffer Row)
117     {
118         if (!Row.Sizes_IsNull)
119         {
120             string[] sizes = Row.Sizes.Split(',');
121
122             foreach (string s in sizes)
123             {
124                 OutputDataBuffer.AddRow();
125                 OutputDataBuffer.Sizes = s.Trim();
126             }
127         }
128     }
129
130 }
131

```

Hình 218. Tiến hành build code vừa sửa đổi

Kết quả khi build thành công

```

    This tool window can only be used by a SQL Server Integration Services package document.

    main.cs
    SC_42e9addee48146e48019e0dc2d05469b
    ScriptMain
    Input0_ProcessInputRow(Input0Buffer Row)

    184     /// <summary>
    185     /// This method is called once for every row that passes through the component from Input0.
    186     ///
    187     /// Example of reading a value from a column in the the row:
    188     /// string zipCode = Row.ZipCode
    189     ///
    190     /// Example of writing a value to a column in the row:
    191     /// Row.ZipCode = zipCode
    192     ///
    193     /// </summary>
    194     /// <param name="Row">The row that is currently passing through the component</param>
    195
    196     public override void Input0_ProcessInputRow(Input0Buffer Row)
    197     {
    198         if (!Row.Sizes.IsNull)
    199         {
    200             string[] sizes = Row.Sizes.Split('|');
    201
    202             foreach (string s in sizes)
    203             {
    204                 OutputDataBuffer.AddRow();
    205                 OutputDataBuffer.Sizes = s.Trim();
    206             }
    207         }
    208     }
    209
    210 }

    2 references:
    116
    117
    118
    119
    120
    121
    122
    123
    124
    125
    126
    127
    128
    129
    130
    131

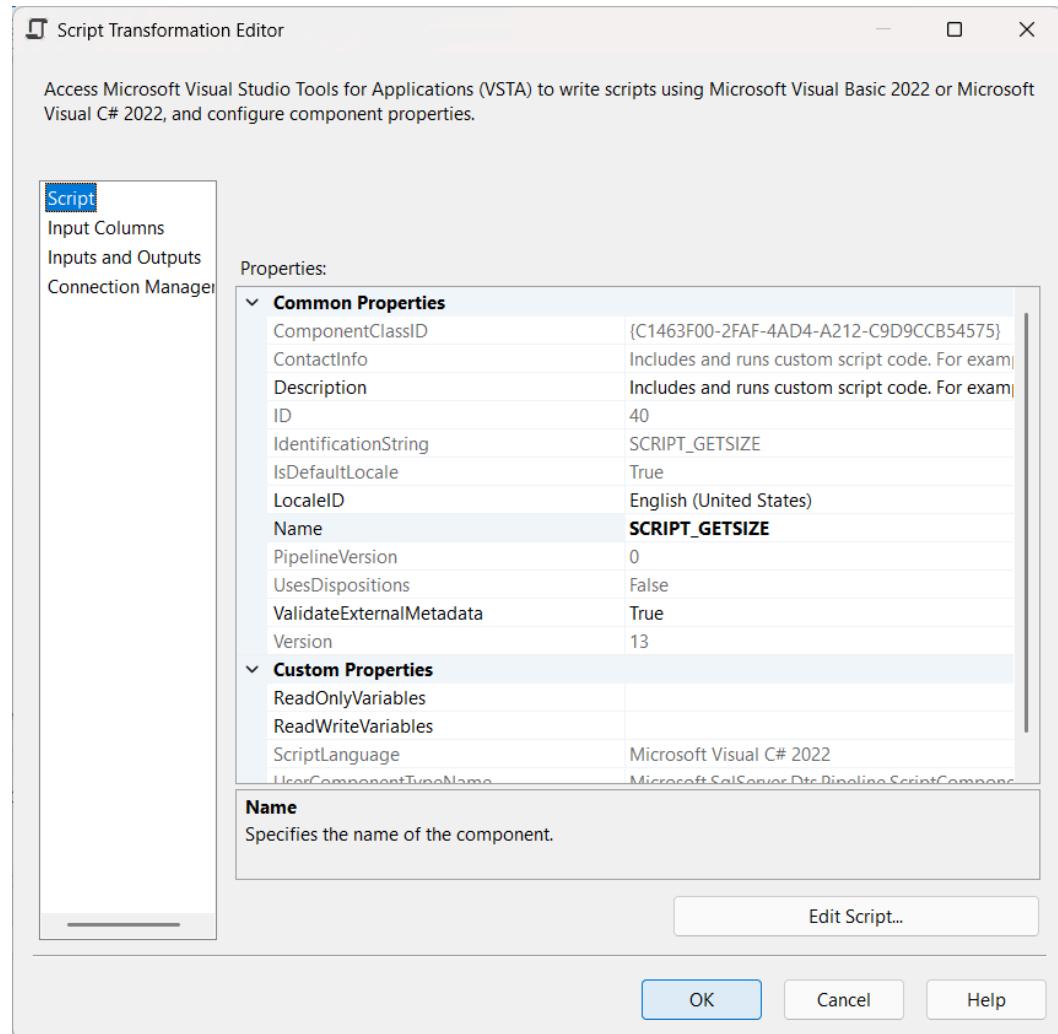
```

Output

Show output from: Build  
Build started: Project: SC\_42e9addee48146e48019e0dc2d05469b  
Build succeeded

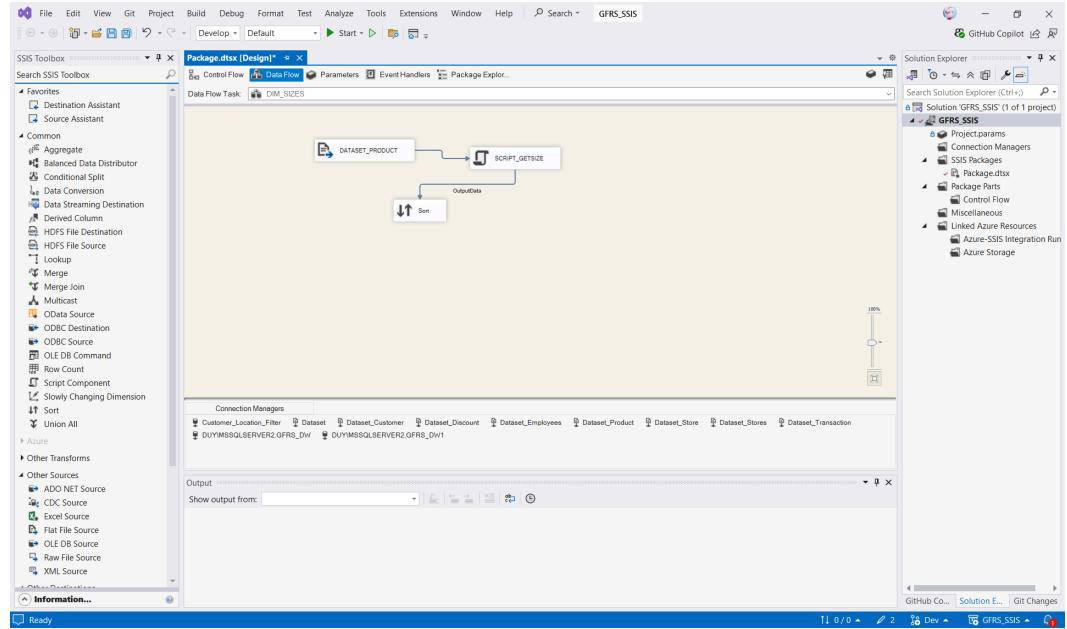
Hình 219. Buil thành công SCRIPT\_GETSIZE

- **Bước 12:** Đóng cửa sổ code . Chọn **OK** và thoát cửa sổ **Script Transformation Editor**



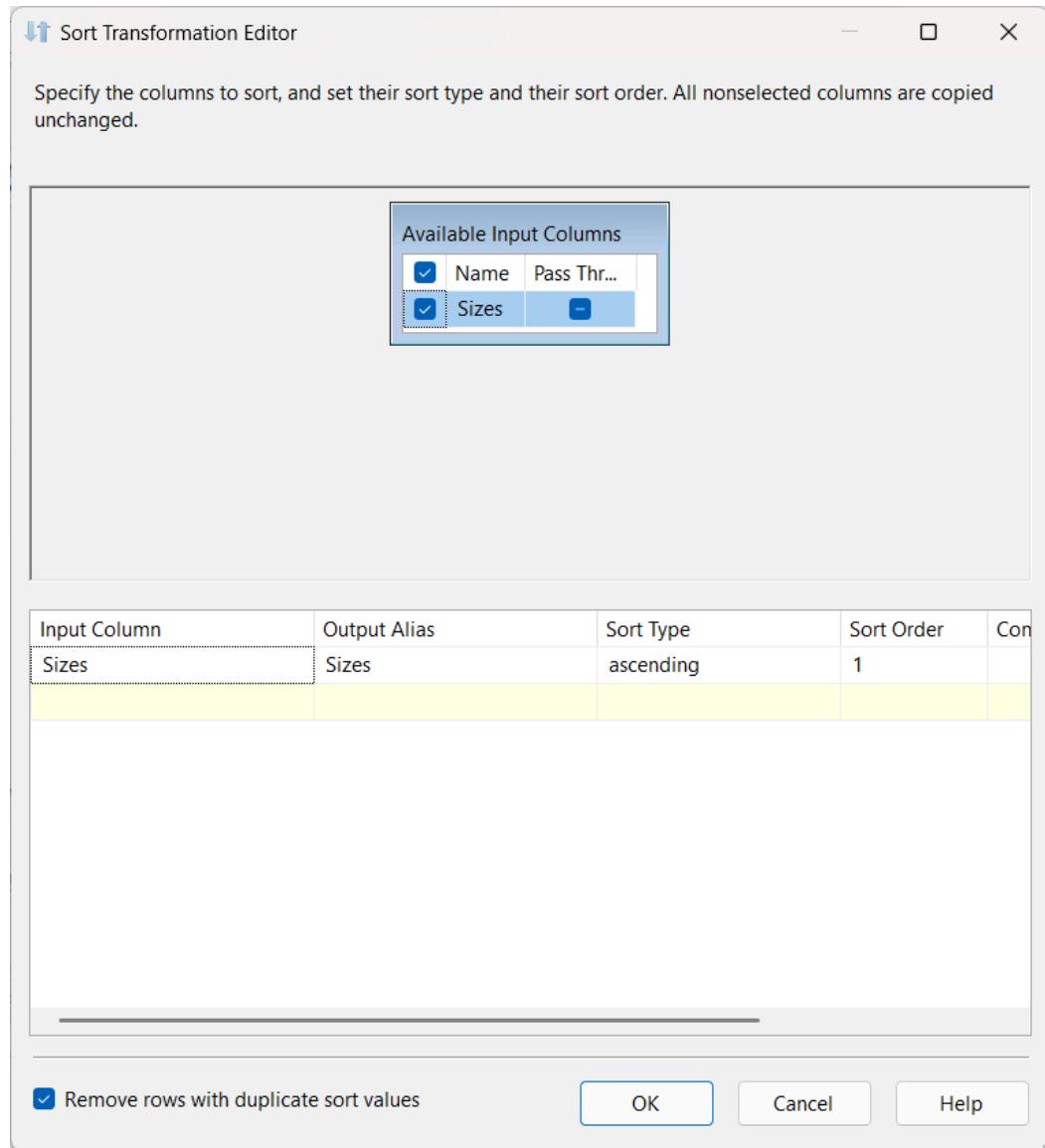
Hình 220. Chọn OK để kết thúc quá trình chỉnh sửa SCRIPT\_GETSIZE

- **Bước 13:** Kéo chức năng **Sort** từ **SSIS Toolbox** vào vùng **Data Flow**. Một cửa sổ hiện lên, ở phần **Output**, ta chọn **OutputData**.



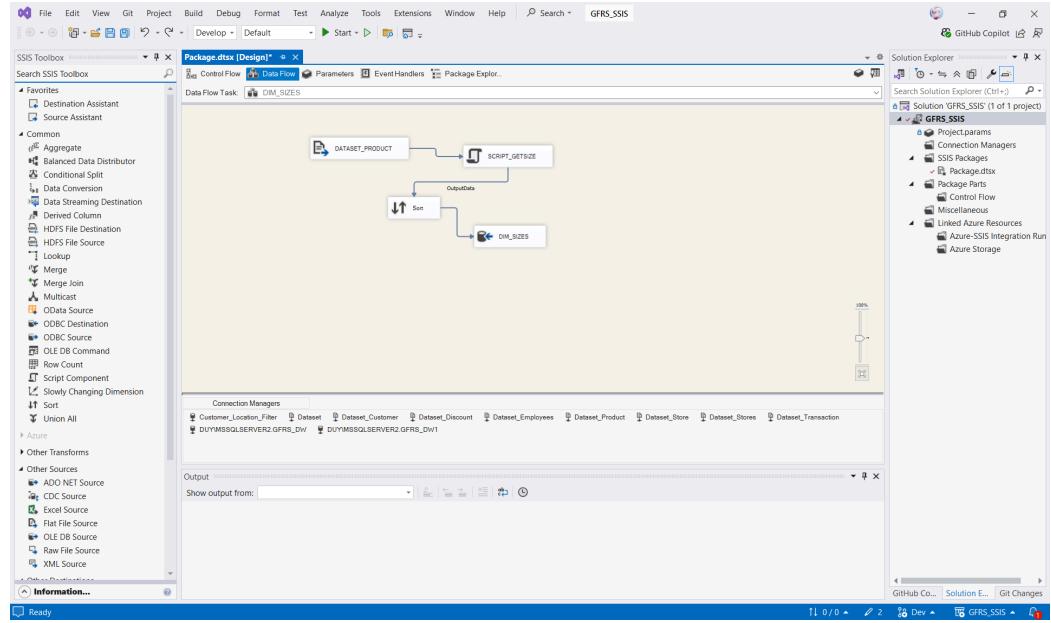
Hình 221. Tạo chức năng Sort trong Dim\_SIZES

- **Bước 14:** Nhấn đúp chuột vào **Sort**, cửa sổ **Sort Transformation Editor** sẽ hiện lên. Ta chọn cột **Sizes** trên cửa sổ **Available Input Columns**. Sau đó chọn **Remove rows with duplicate sort values** để lọc những kích thước trùng nhau. Sau đó nhấn **OK** và kết thúc cửa sổ **Sort Transformation Editor**.



Hình 222. Cài đặt Sort trong Dim\_SIZES

- **Bước 15:** Kéo ADO NET Destination từ SSIS Toolbox vào vùng Data Flow và đổi tên thành Dim\_SIZES. Kéo luồng dữ liệu từ Script Component vào Dim\_SIZES.



Hình 223. Tạo ADO.NET Destination Dim\_SIZES

- **Bước 16:** Vào SQL Server 2022, chạy lệnh DDL để tạo Dim\_SIZES.

```

CREATE TABLE [Dim_COLORS] (
    [ColorID] integer identity(1,1) PRIMARY KEY,
    [Color] varchar(255)
)
select * from Dim_COLORS;

drop table if exists Dim_SIZES;
CREATE TABLE [Dim_SIZES] (
    [SizeID] integer identity(1,1) PRIMARY KEY,
    [Size] varchar(255)
)

drop table if exists Dim_PRODUCTS;

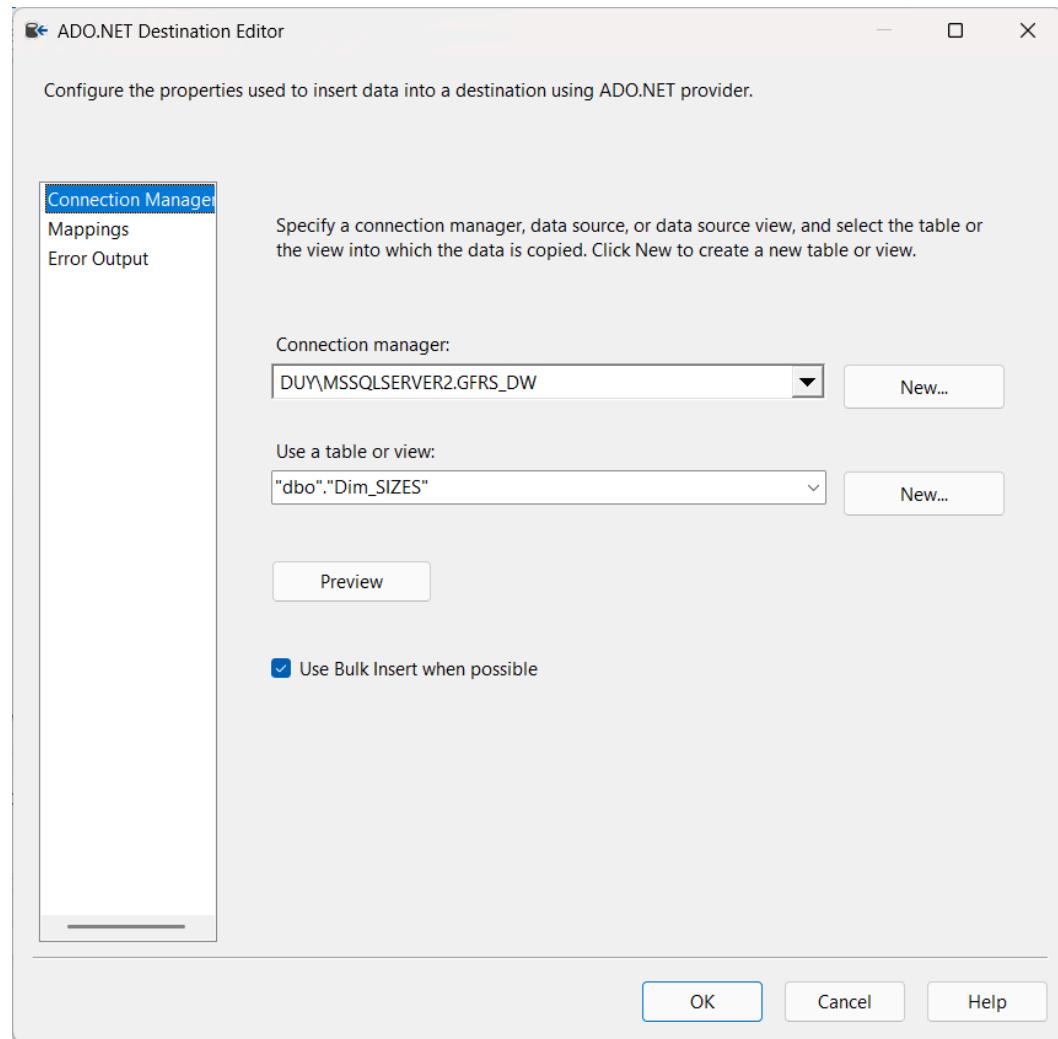
```

ColorID	Color
7	GREEN
8	LILAC
9	MUSTARD
10	NEUTRAL
11	PINK
12	RED
13	SILVER
14	TURQUOISE
15	WHITE
16	YELLOW
17	
18	BEIGE
19	BLACK
20	BLUE

Hình 224. Chạy lệnh DDL tạo Dim\_SIZES trong kho dữ liệu

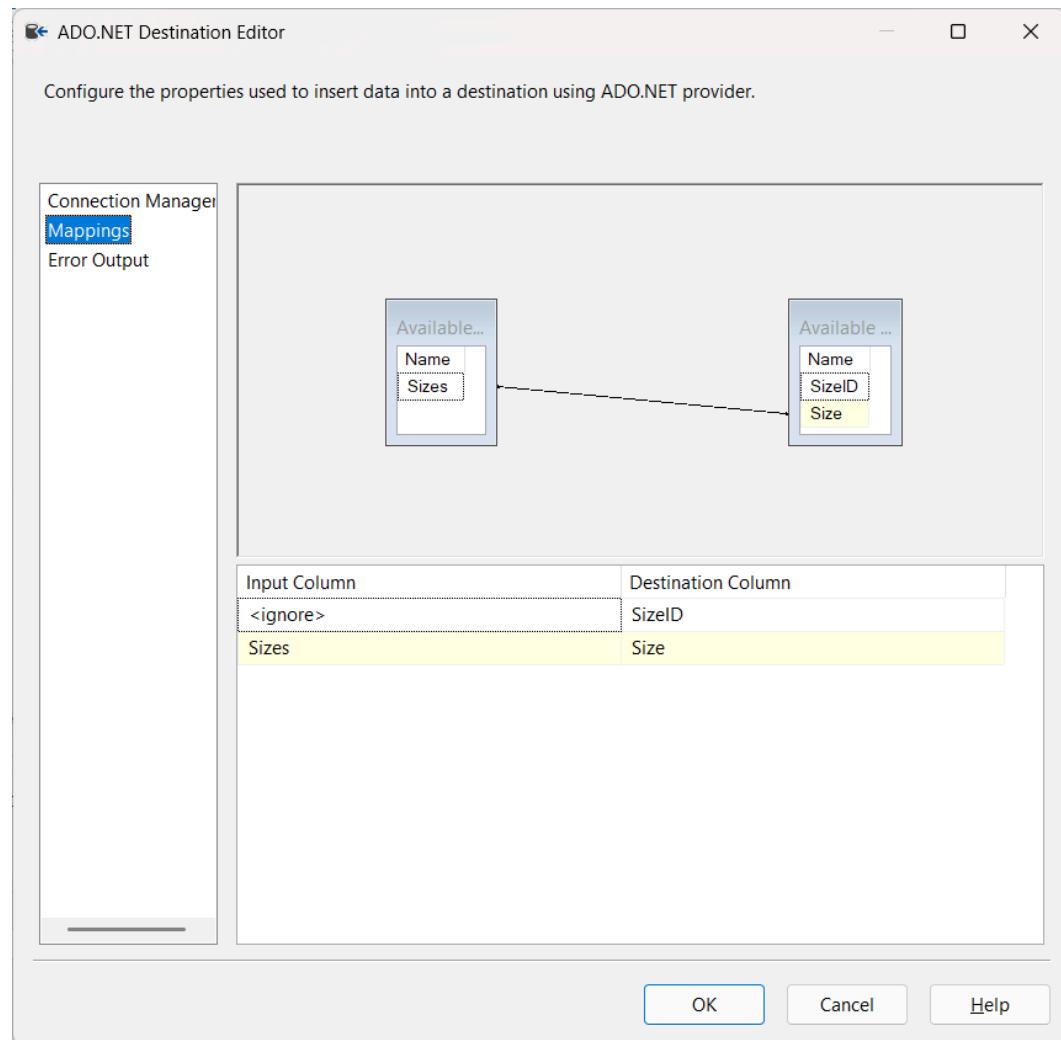
- **Bước 17:** Nhấn đúp chuột vào Dim\_SIZES. Trong tab Connection Manager của màn hình ADO.NET Destination Editor, ở Use a table or

view tiến hành nhấn chọn “**dbo”.” Dim\_SIZES**”, cũng chính là bảng vừa được tạo bằng lệnh **DDL** ở **SQL Server 2022**



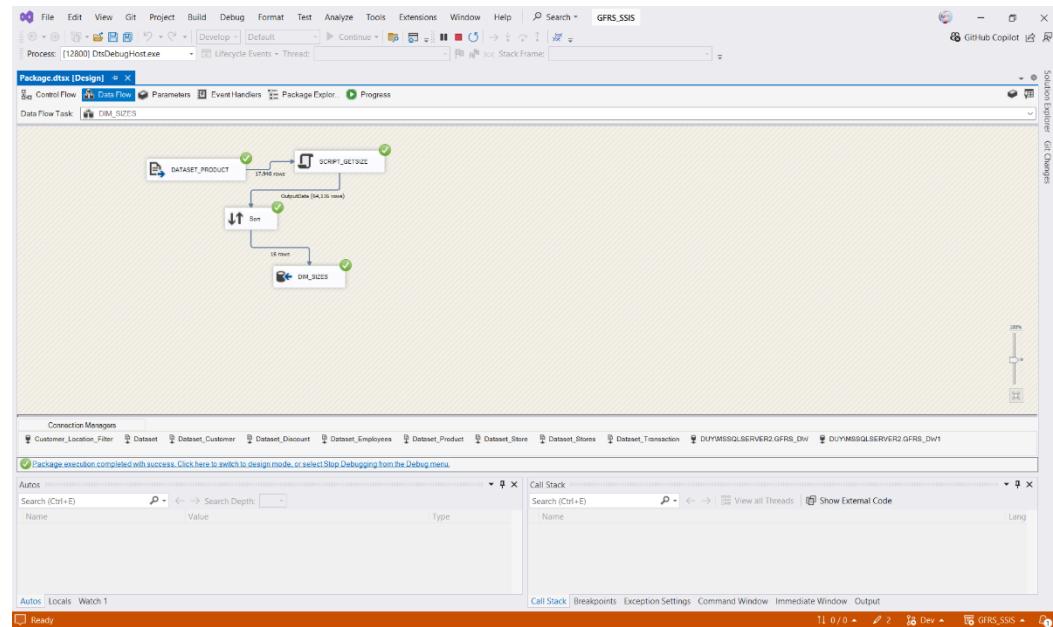
Hình 225. Cài đặt ADO NET Destination Dim\_SIZES

- **Bước 18:** Ở tab **Mapping** của màn hình **ADO.NET Destination Editor** tiến hành chọn các thuộc tính thích hợp ở cột bên trái để ánh xạ với các thuộc tính ở cột bên phải.



Hình 226. Cài đặt ADO.NET Destination

- **Bước 19:** Nhấn Start để kiểm tra quá trình tạo và đổ dữ liệu vào **Dim\_SIZES** có thành công hay chưa.



Hình 227. Đỗ dữ liệu vào Dim\_SIZES

Kết quả hiển thị cho thấy quá trình diễn ra thành công và có **16 rows** được đỗ vào **Dim\_SIZES**.

- **Bước 20:** Vào SQL Server 2022 kiểm tra và thấy dữ liệu đã được đỗ vào Dim\_SIZES thành công.

The screenshot shows the SQL Server Management Studio (SSMS) interface. In the Object Explorer, the database 'GFRS\_DW' is selected. In the main pane, a query window titled 'SQLQuery1...\\maing (72)\*' contains the following SQL code:

```

74 --select * from Dim_COLORS;
75
76 drop table if exists Dim_SIZES;
77 CREATE TABLE [Dim_SIZES] (
78     [SizeID] integer identity(1,1) PRIMARY KEY,
79     [Size] varchar(255)
80 )
81
82 select * from Dim_SIZES;
83 drop table if exists Dim_PRODUCTS;
84 CREATE TABLE Dim_PRODUCTS (
85     [ProductID] integer identity(1,1) PRIMARY KEY,
86

```

The results pane shows the data for the Dim\_SIZES table:

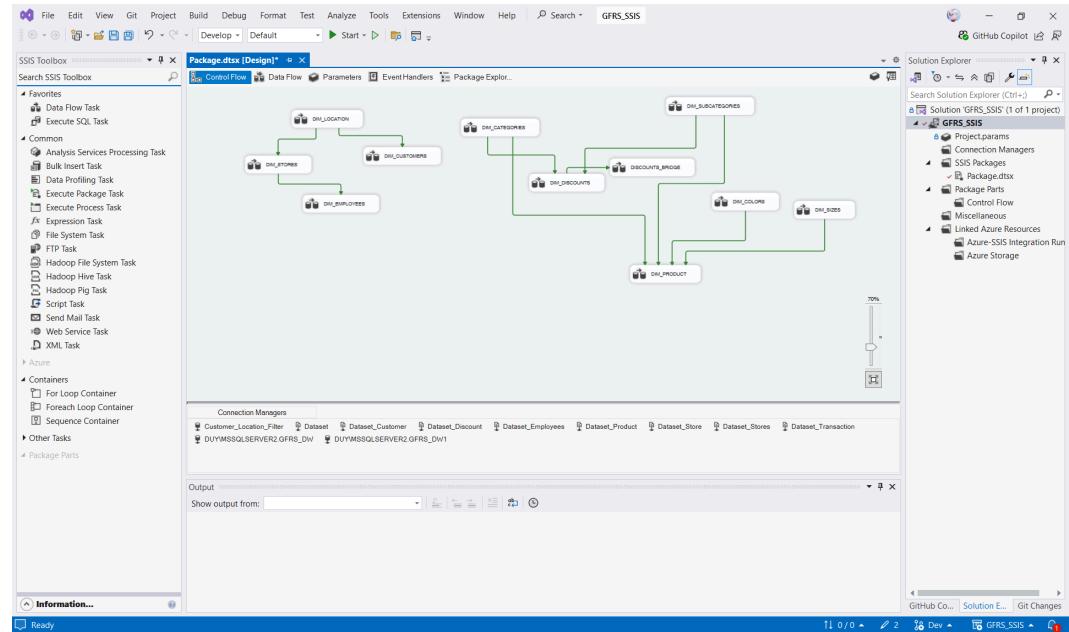
SizeID	Size
1	1
2	36
3	38
4	40
5	42
6	44
7	46
8	48
9	G
10	GG
11	L
12	M
13	P
14	S

The status bar at the bottom indicates 'Query executed successfully.'

Hình 228. Dữ liệu được đỗ thành công vào Dim\_SIZES

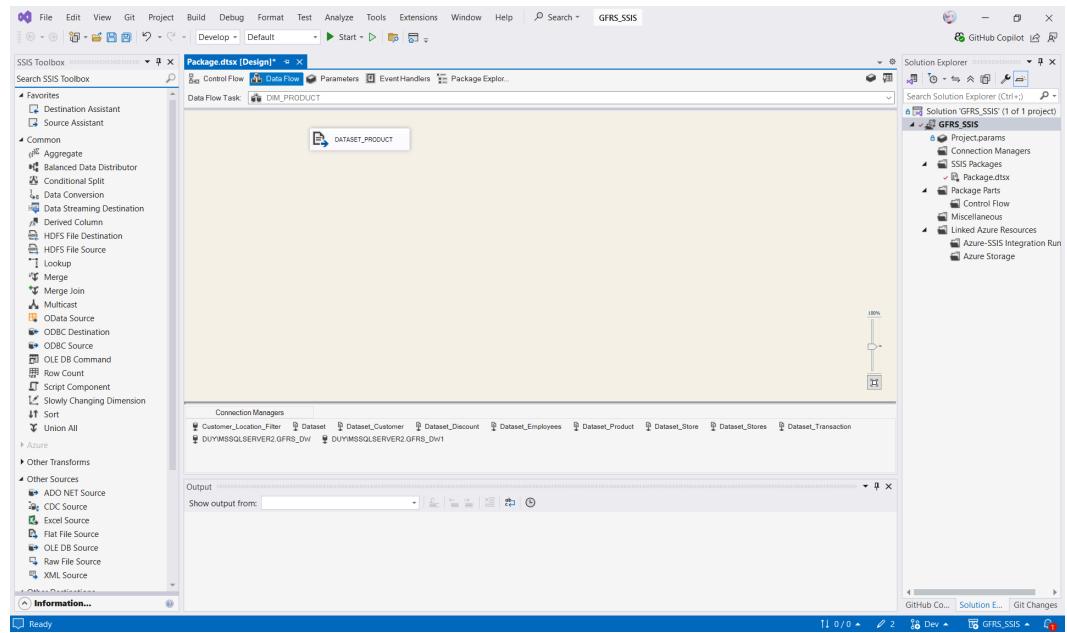
### II.3.10. Dim\_PRODUCTS

- **Bước 1:** Kéo chức năng Data Flow Task trong vùng SSIS Toolbox vào vùng Control Flow và đổi tên thành **Dim\_PRODUCTS**



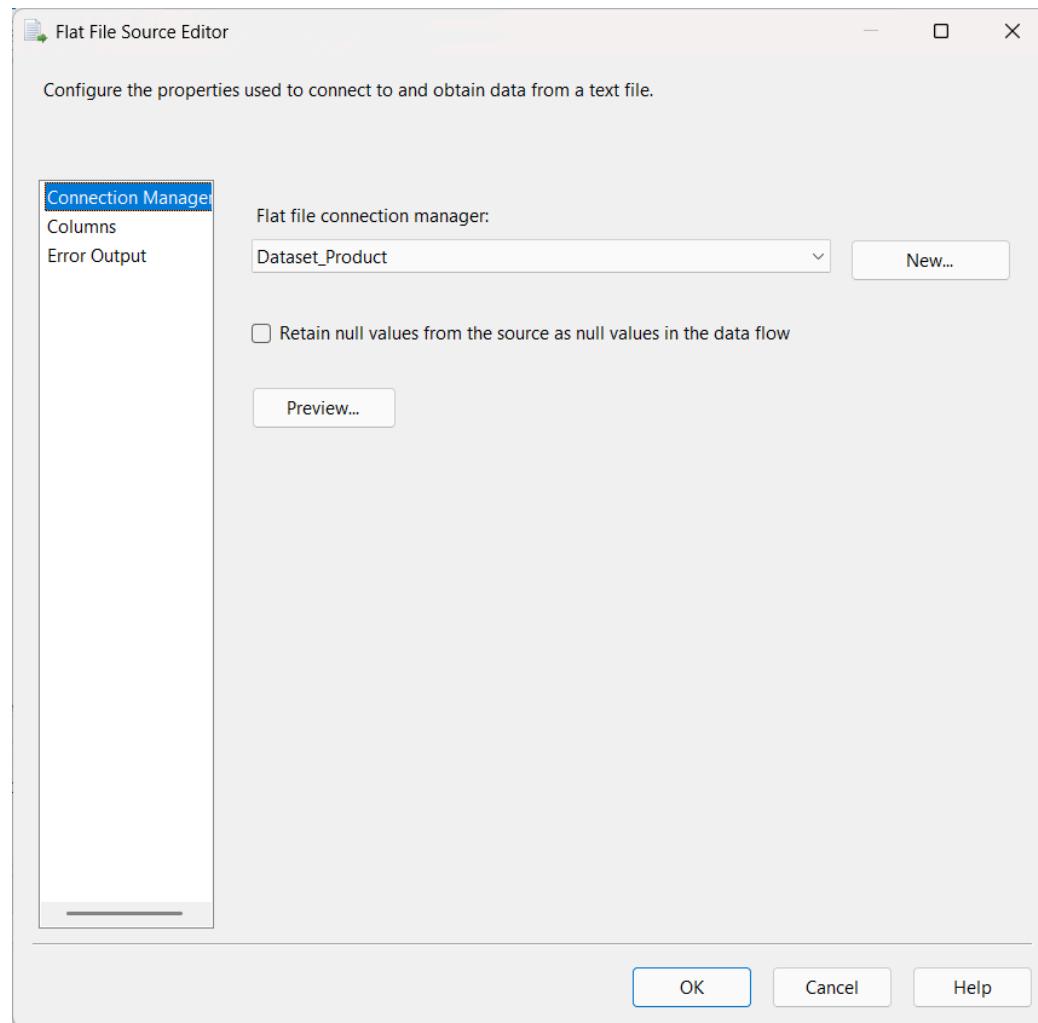
Hình 229. Tạo chức năng Dim\_PRODUCTS trong Control Flow

- **Bước 2:** Nhấn đúp chuột vào **Dim\_PRODUCTS**. Sau đó kéo chức năng **Flat File Source** vào vùng làm việc và đổi tên thành **DATASET\_PRODUCT**



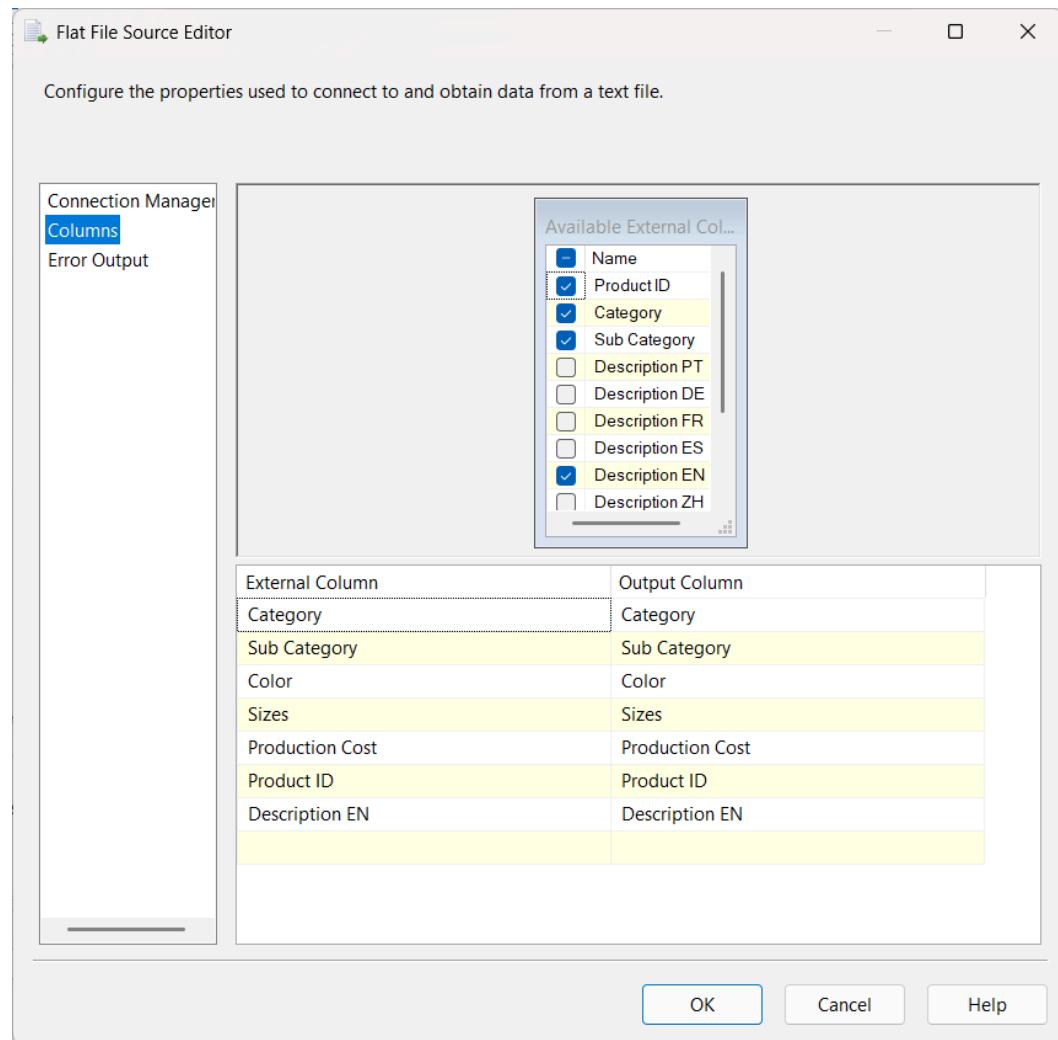
Hình 230. Tạo DATASET\_PRODUCT trong Dim\_PRODUCTS

- **Bước 3:** Nhấn đúp chuột vào **DATASET\_PRODUCT**. Màn hình **Flat File Source Editor** hiện ra. Ở tab **Connection Manager** ta chọn **Dataset\_Product** đã được thiết lập ở phần trước



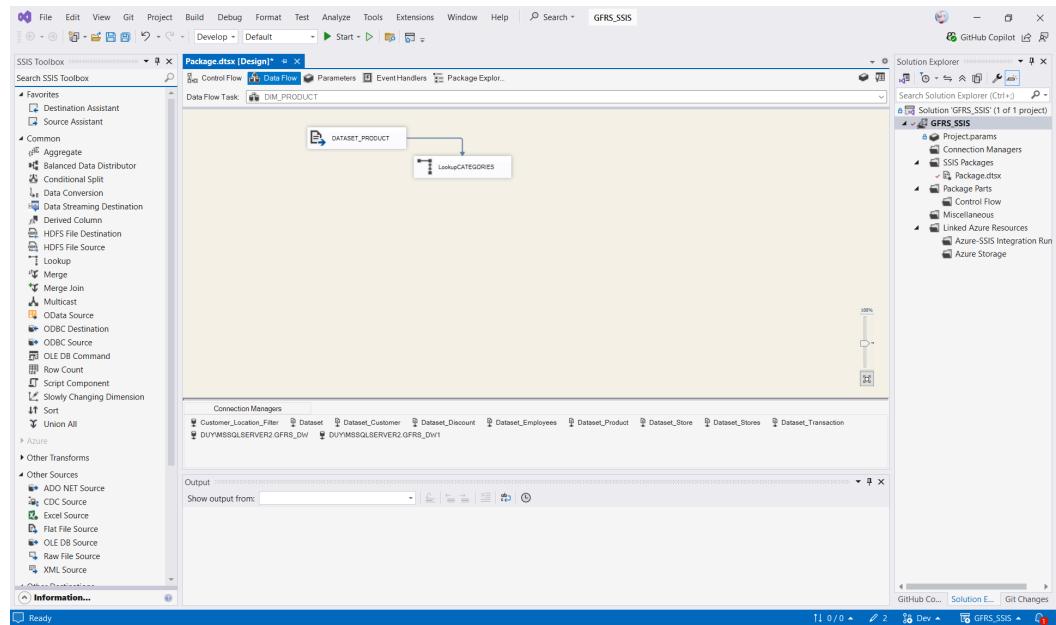
*Hình 231. Kết nối nguồn dữ liệu cho DATASET\_PRODUCT*

- **Bước 4:** Chuyển sang tab **Columns**, ta có thể lựa chọn các thuộc tính cần thiết cho quá trình nạp dữ liệu cho **Dim\_PRODUCT**. Sau đó ta có thể bấm **OK** để đóng cửa sổ **Flat File Source Editor**



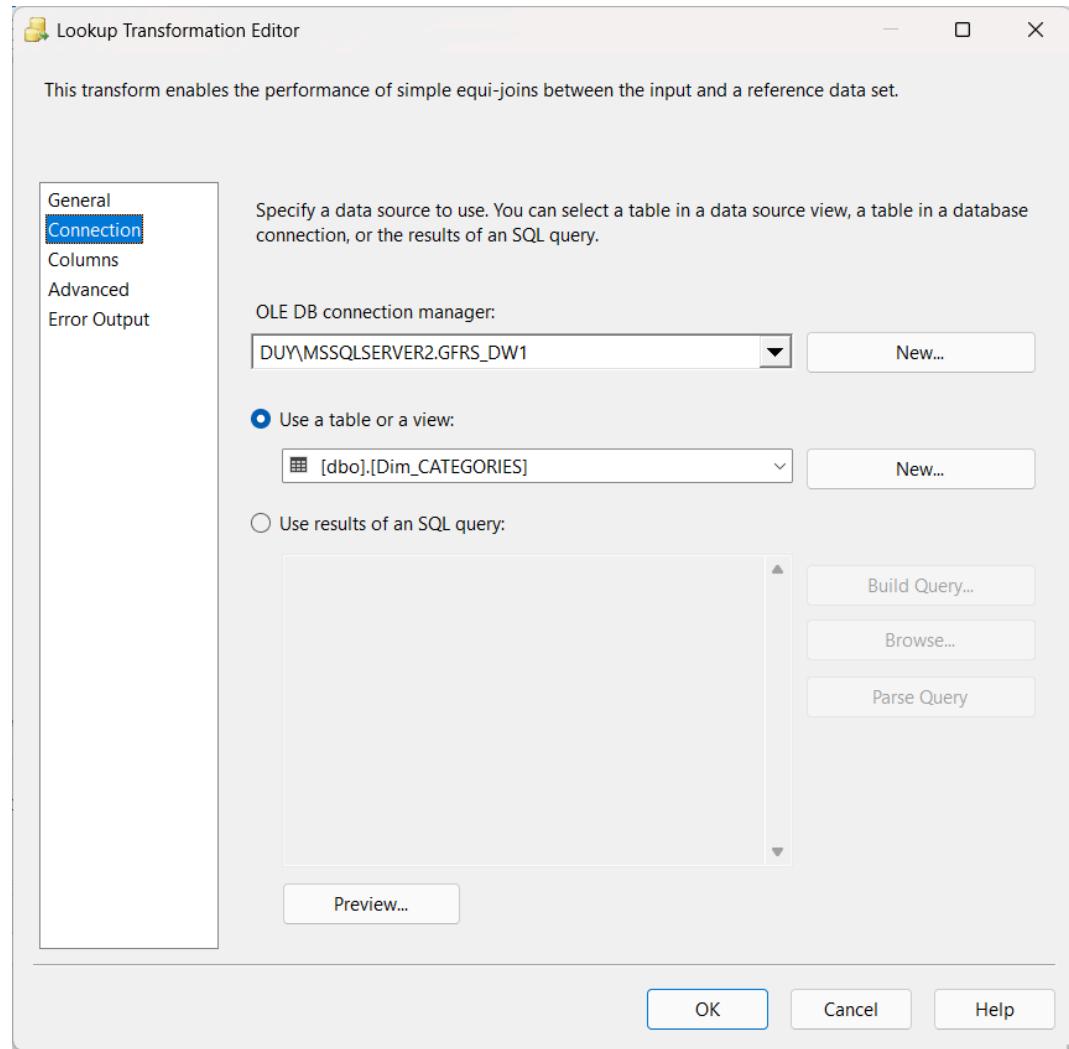
Hình 232. Chọn các thuộc tính cần thiết cho quá trình tạo  
Dim\_PRODUCTS

- **Bước 5:** Kéo chức năng **Lookup** vào vùng **Data Flow** và đổi tên thành **LookupCATEGORIES**. Kéo mũi tên luồng dữ liệu từ **DATASET\_PRODUCT** vào **LookupCATEGORIES**.



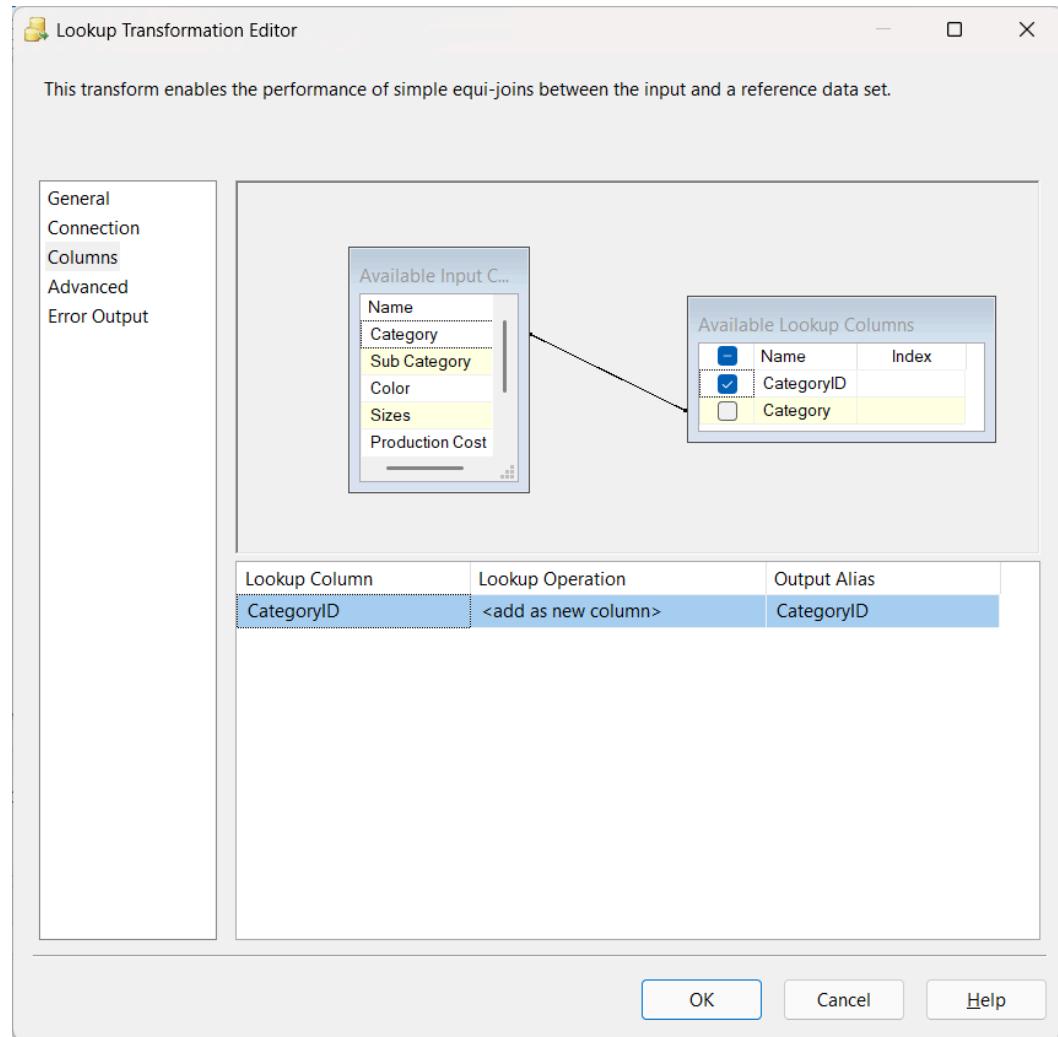
Hình 233. Tạo chức năng LookupCATEGORIES trong Dim\_DISCOUNTS

- **Bước 6:** Nhấn đúp chuột vào **LookupCATEGORIES**, màn hình **Lookup Transformation Editor** xuất hiện. Do chúng ta đã thiết lập **OLE DB connection manager** ở bước trước, nên ta có thể sử dụng lại. Ở phần **Use a table or a view**, tiến hành chọn **[dbo].[Dim\_CATEGORIES]**.



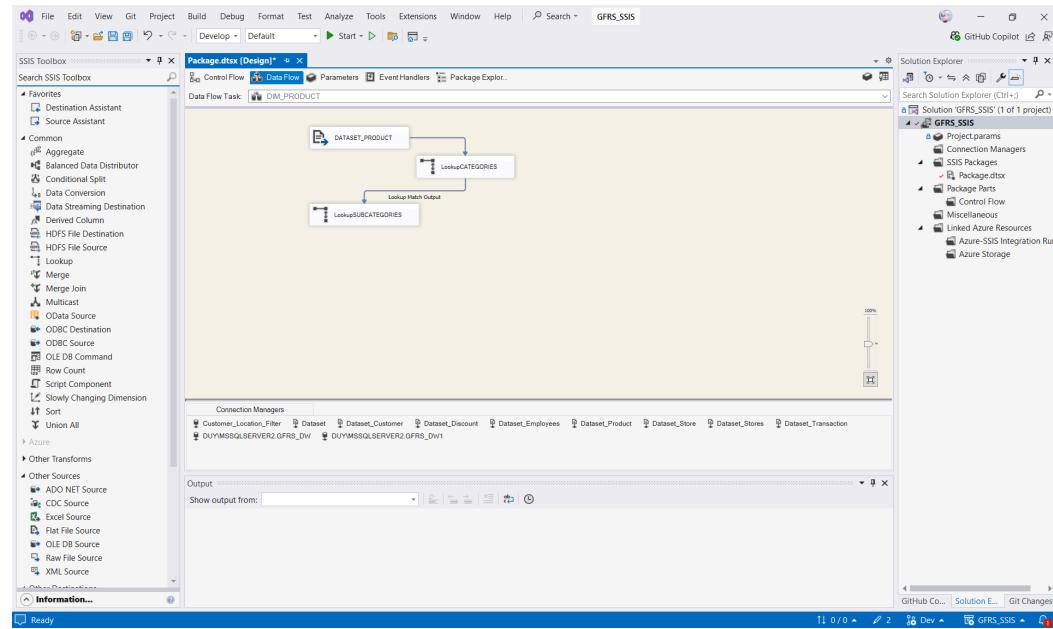
Hình 234. Kết nối nguồn dữ liệu cho Lookup CATEGORIES

- **Bước 7:** Trong tab **Columns**, ở bảng bên phải click chọn **CategoryID**. Ta kéo **Category** từ bảng **Available Input Columns** qua tương ứng **Category** bảng **Available Lookup Columns** (quá trình này giúp hệ thống xác định vùng dữ liệu cần tìm kiếm). Sau đó bấm **OK** để đóng cửa sổ **Lookup Transformation Editor**



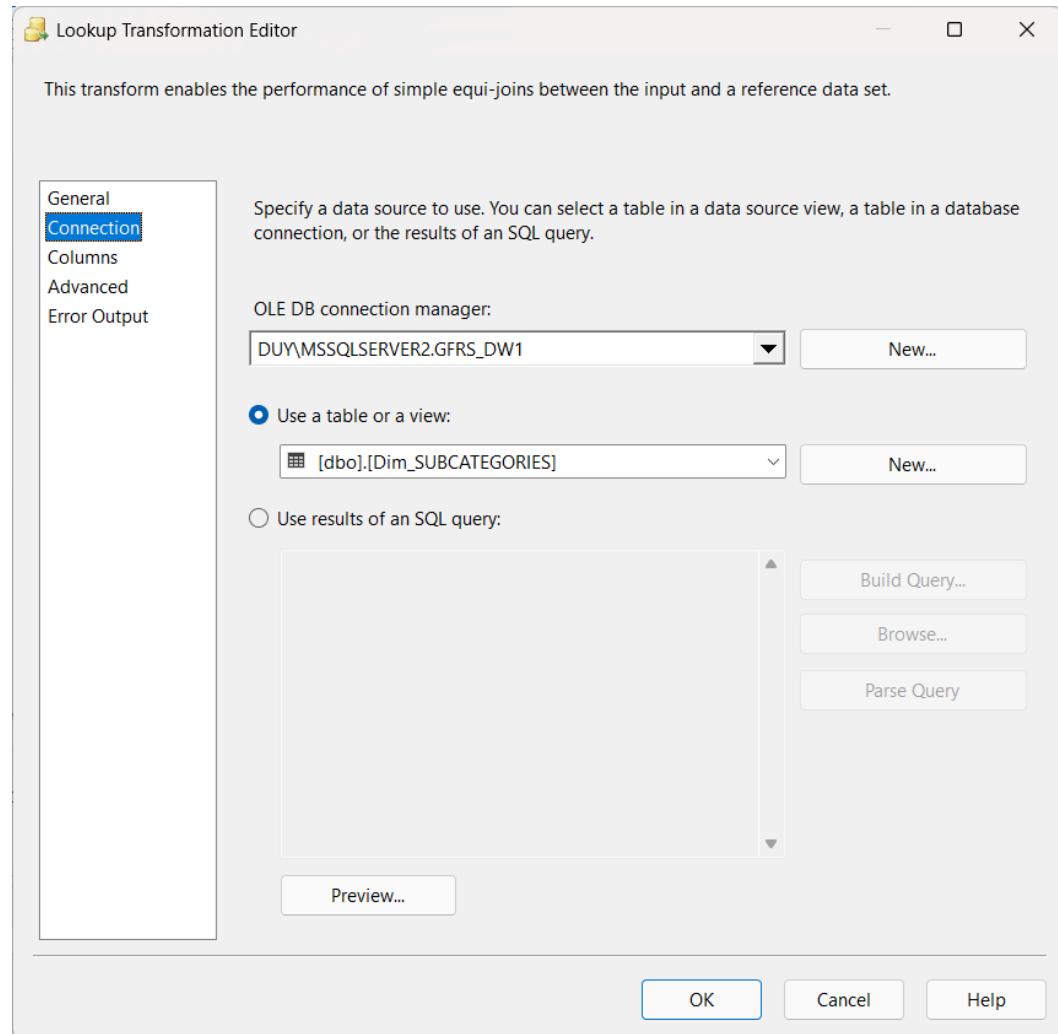
Hình 235. Cài đặt chức năng **LookupCATEGORIES**

- **Bước 8:** Kéo chức năng **Lookup** vào vùng **Data Flow** và đổi tên thành **LookupSUBCATEGORIES**. Kéo mũi tên luồng dữ liệu từ **LookupCATEGORIES** vào **LookupSUBCATEGORIES**. Một hộp thoại sẽ xuất hiện, chọn **Lookup Match Output** và nhấn **OK**.



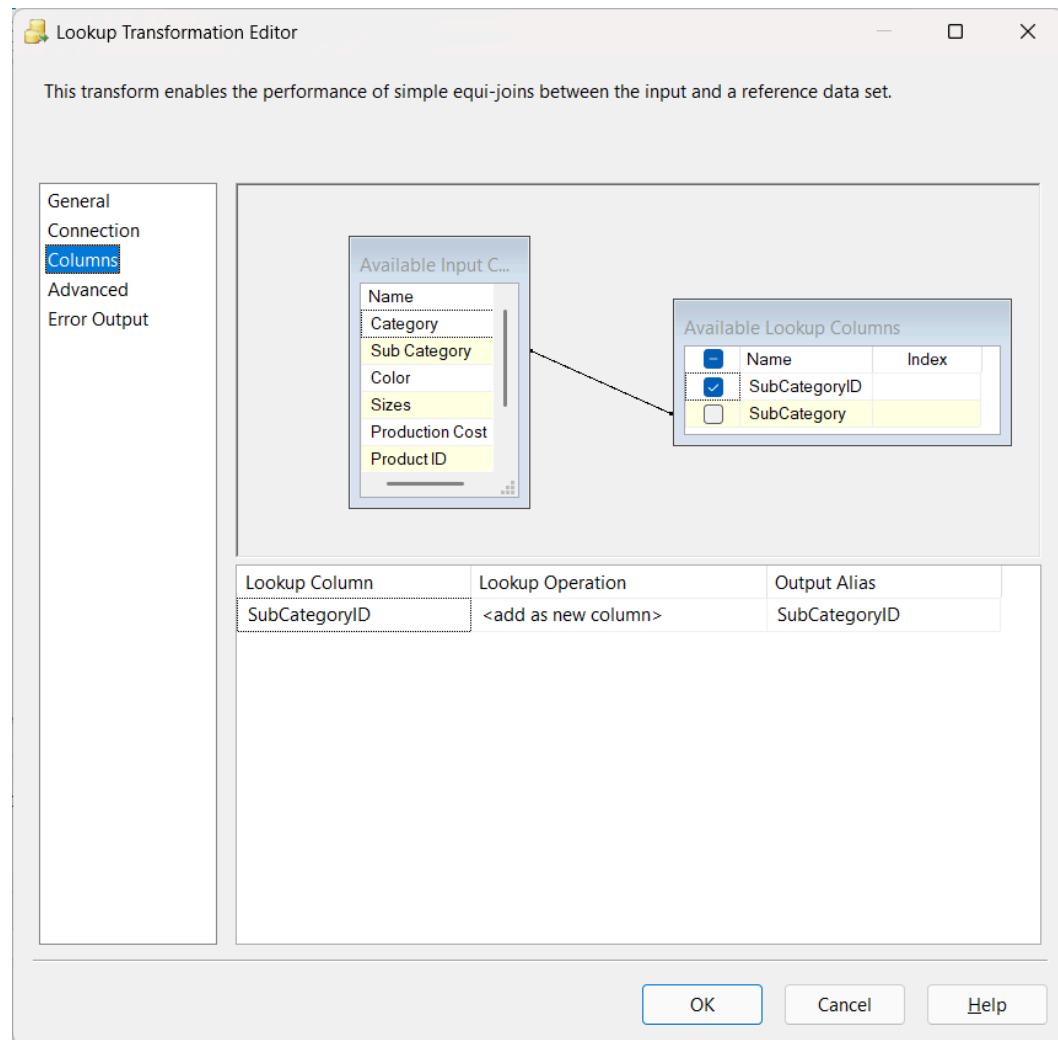
*Hình 236. Tạo chức năng LookupSUBCATEGORIES trong Dim\_DISCOUNTS*

- **Bước 9:** Nhấn đúp chuột vào **LookupSUBCATEGORIES**, màn hình **Lookup Transformation Editor** xuất hiện. Do chúng ta đã thiết lập **OLE DB connection manager** ở bước trước, nên ta có thể sử dụng lại. Ở phần **Use a table or a view**, tiến hành chọn **[dbo].[Dim\_SUBCATEGORIES]**.



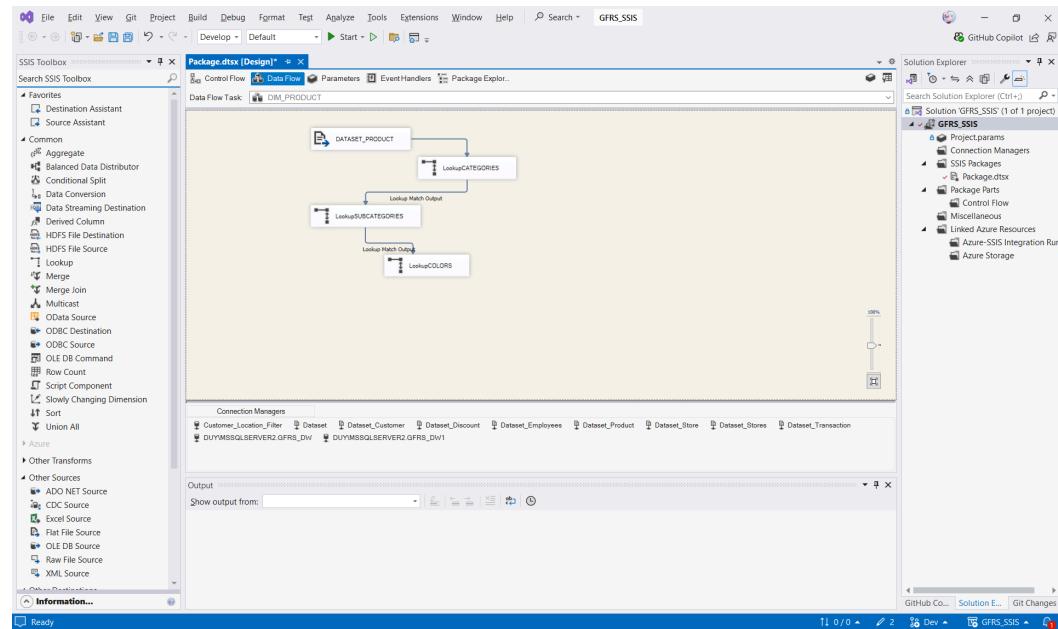
Hình 237. Kết nối nguồn dữ liệu cho chức năng LookupSUBCATEGORIES

- **Bước 10:** Trong tab **Columns**, ở bảng bên phải click chọn **SubCategoryID**. Ta kéo **Sub Category** từ bảng **Available Input Columns** qua tương ứng **SubCategory** bảng **Available Lookup Columns** (quá trình này giúp hệ thống xác định vùng dữ liệu cần tìm kiếm). Sau đó bấm **OK** để đóng cửa sổ **Lookup Transformation Editor**



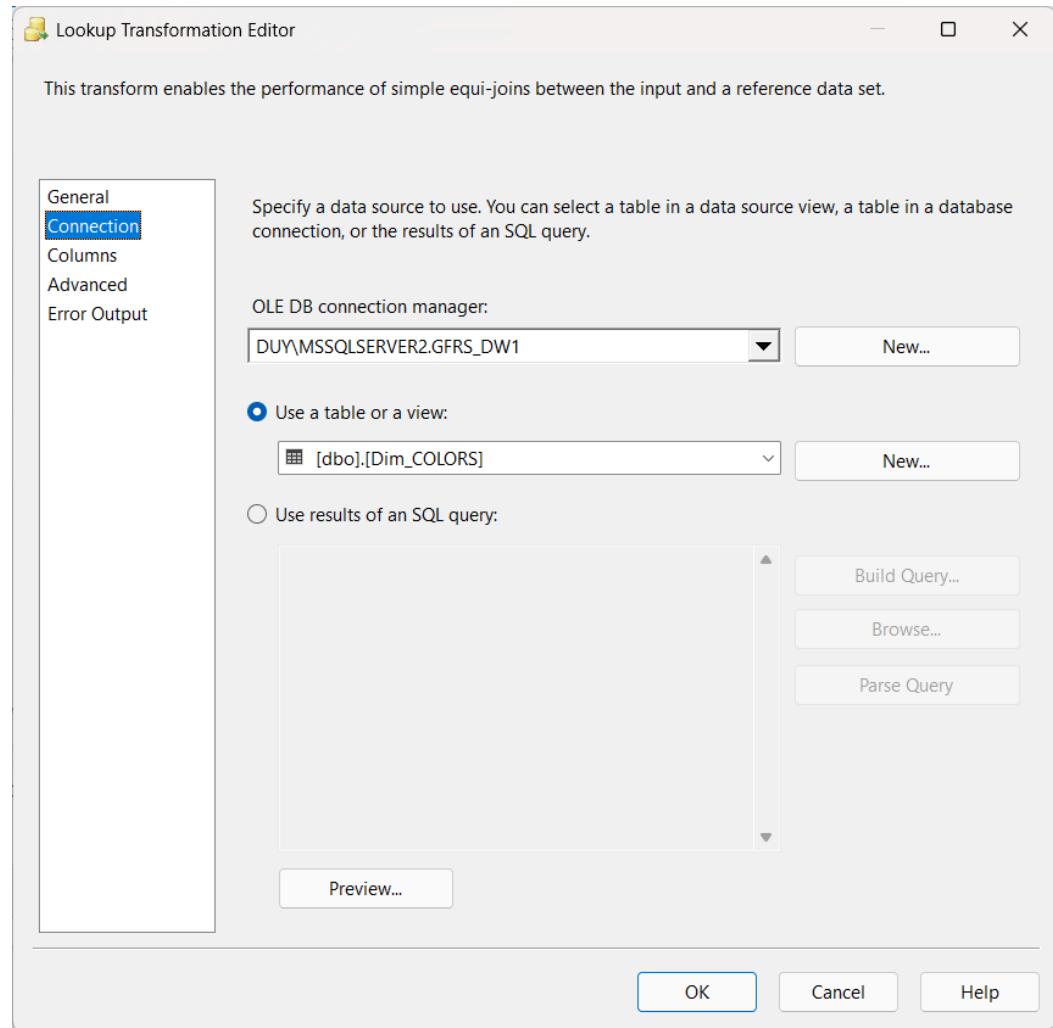
Hình 238. Cài đặt chức năng **LookupSUBCATEGORIES**

- **Bước 11:** Kéo chức năng **Lookup** vào vùng **Data Flow** và đổi tên thành **LookupCOLORS**. Kéo mũi tên luồng dữ liệu từ **LookupSUBCATEGORIES** vào **LookupCOLORS**. Một hộp thoại sẽ xuất hiện, chọn **Lookup Match Output** và nhấn **OK**.



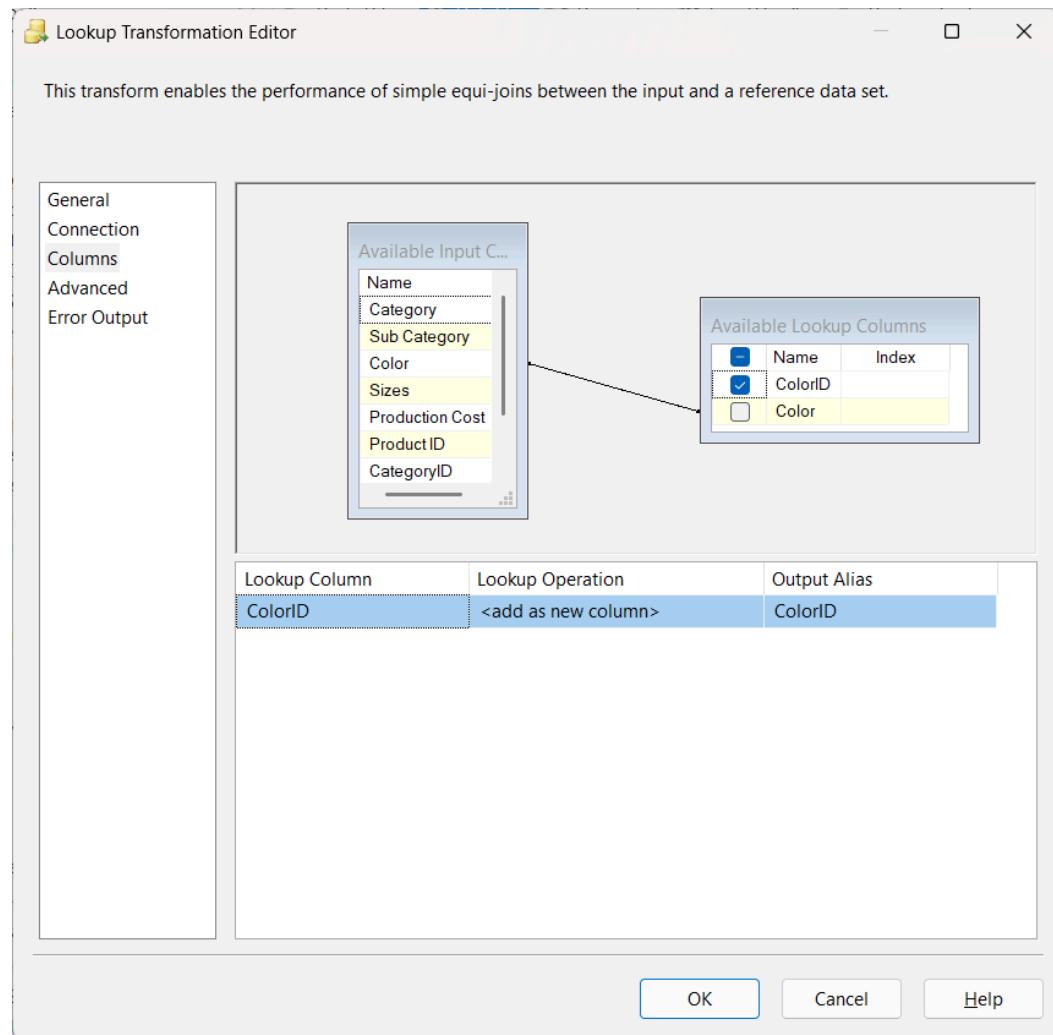
Hình 239. Tạo chức năng LookupCOLORS trong Dim\_DISCOUNTS

- **Bước 12:** Nhấn đúp chuột vào **LookupCOLORS**, màn hình **Lookup Transformation Editor** xuất hiện. Do chúng ta đã thiết lập **OLE DB connection manager** ở bước trước, nên ta có thể sử dụng lại. Ở phần **Use a table or a view**, tiến hành chọn **[dbo].[Dim\_COLORS]**.



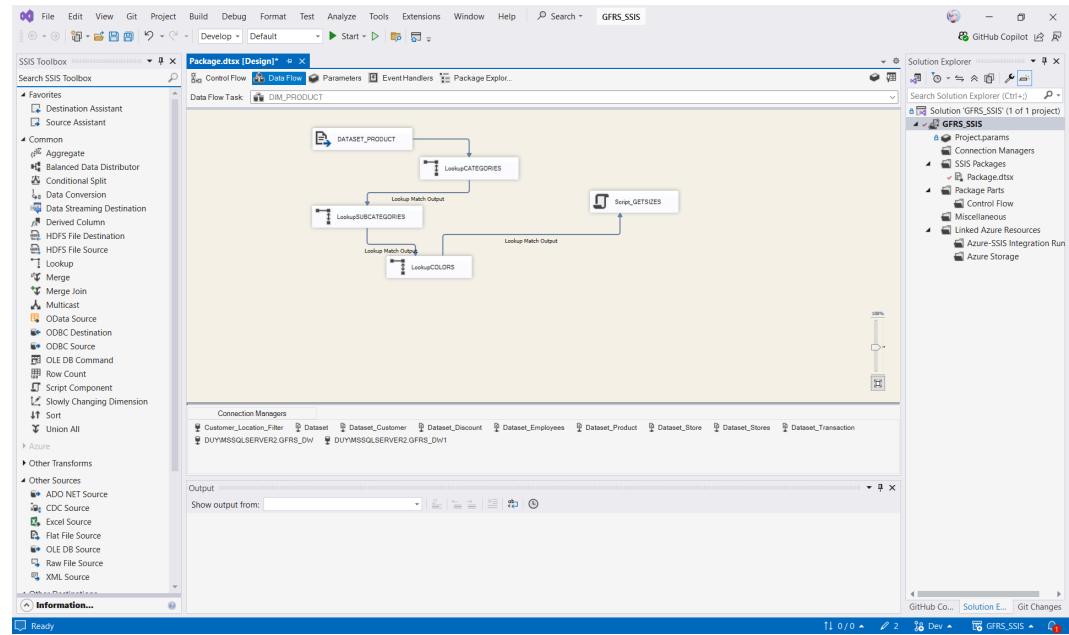
Hình 240. Kết nối nguồn dữ liệu cho chức năng LookupCOLORS

- **Bước 13:** Trong tab **Columns**, ở bảng bên phải click chọn **ColorID**. Ta kéo **Color** từ bảng **Available Input Columns** qua tương ứng **Color** bảng **Available Lookup Columns** (quá trình này giúp hệ thống xác định vùng dữ liệu cần tìm kiếm). Sau đó bấm **OK** để đóng cửa sổ **Lookup Transformation Editor**



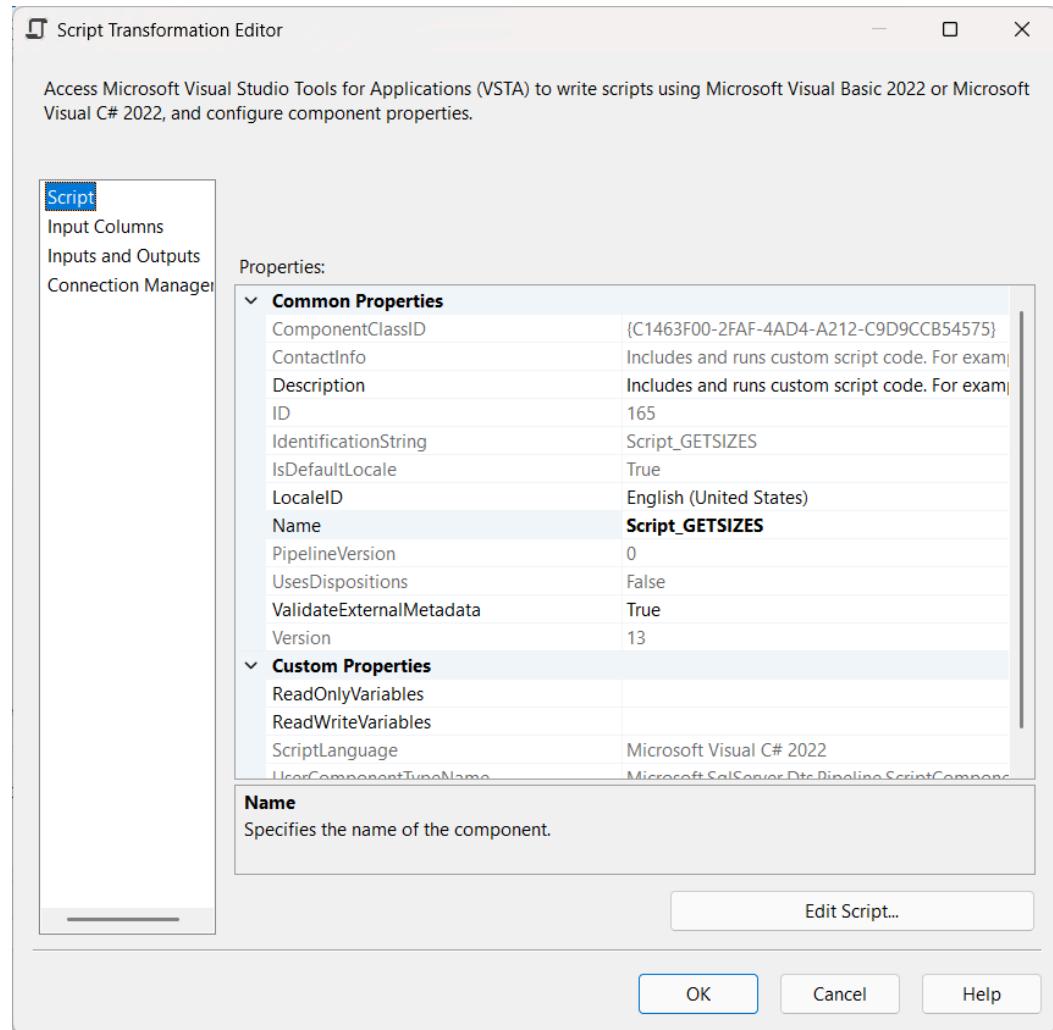
Hình 241. Cài đặt chức năng LookupCOLORS

- **Bước 14:** Kéo Script Component từ cùng SSIS Toolbox vào vùng Data Flow và đổi tên thành Script\_GETSIZES. Sau đó kéo luồng dữ liệu từ LookupCOLORS vào Script\_GETSIZES. Một cửa sổ hiện lên, ta chọn **Lookup Match Output**.



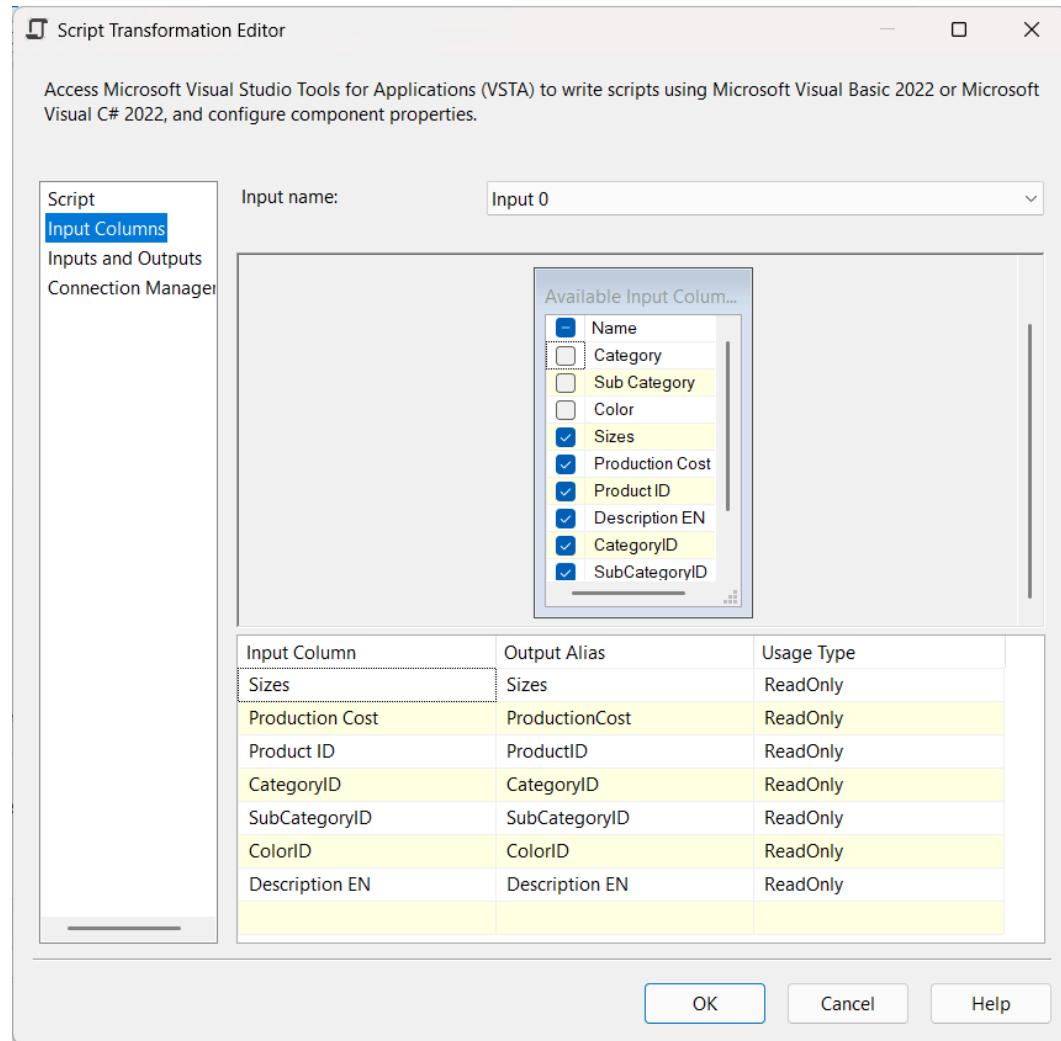
Hình 242. Tạo chức năng Script Component trong Dim\_DISCOUNTS

- **Bước 15:** Nhấn đúp chuột vào Script\_GETSIZES, cửa sổ **Script Transformation Editor** hiện lên.



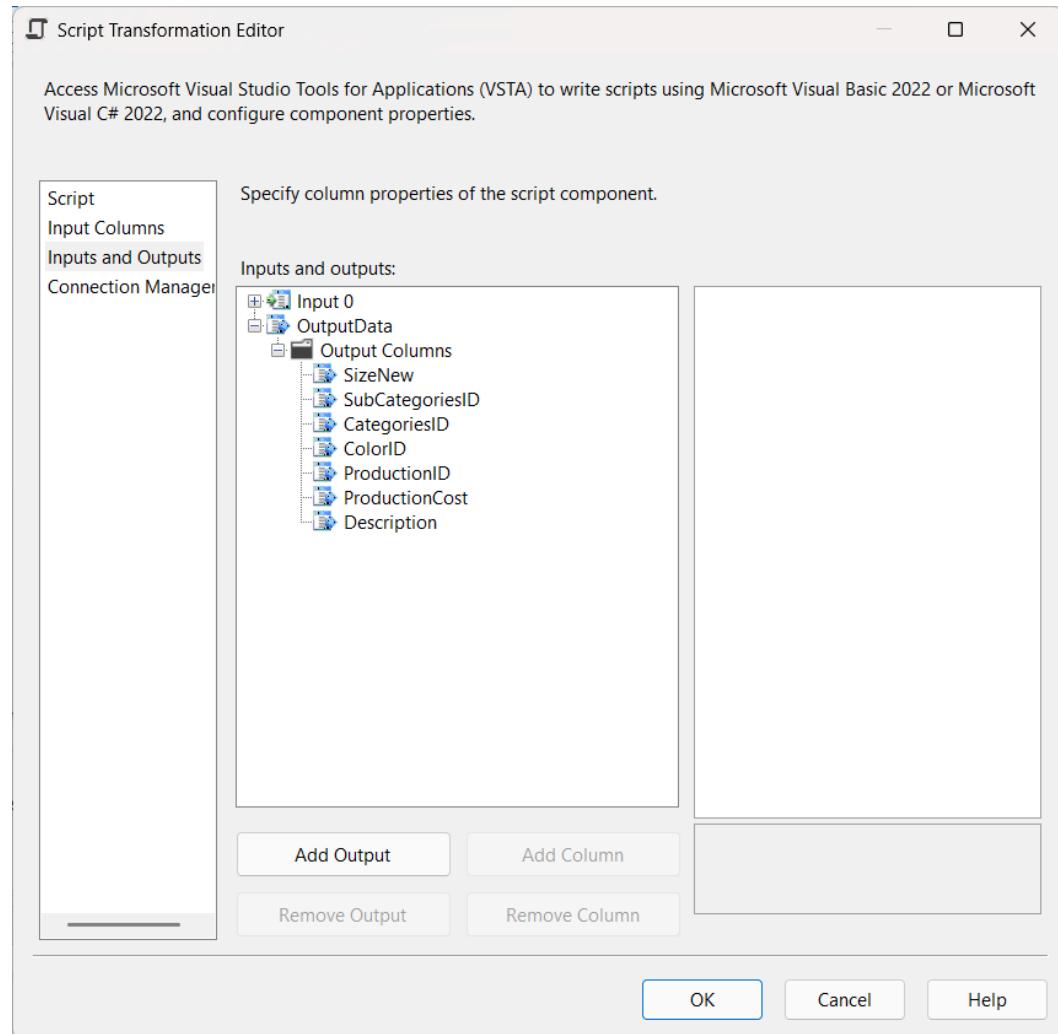
Hình 243. Kiểm tra cấu hình chức năng Script\_GETSIZES

- **Bước 16:** Chuyển sang tab **Input Columns**, ta sẽ chọn các cột cần thiết



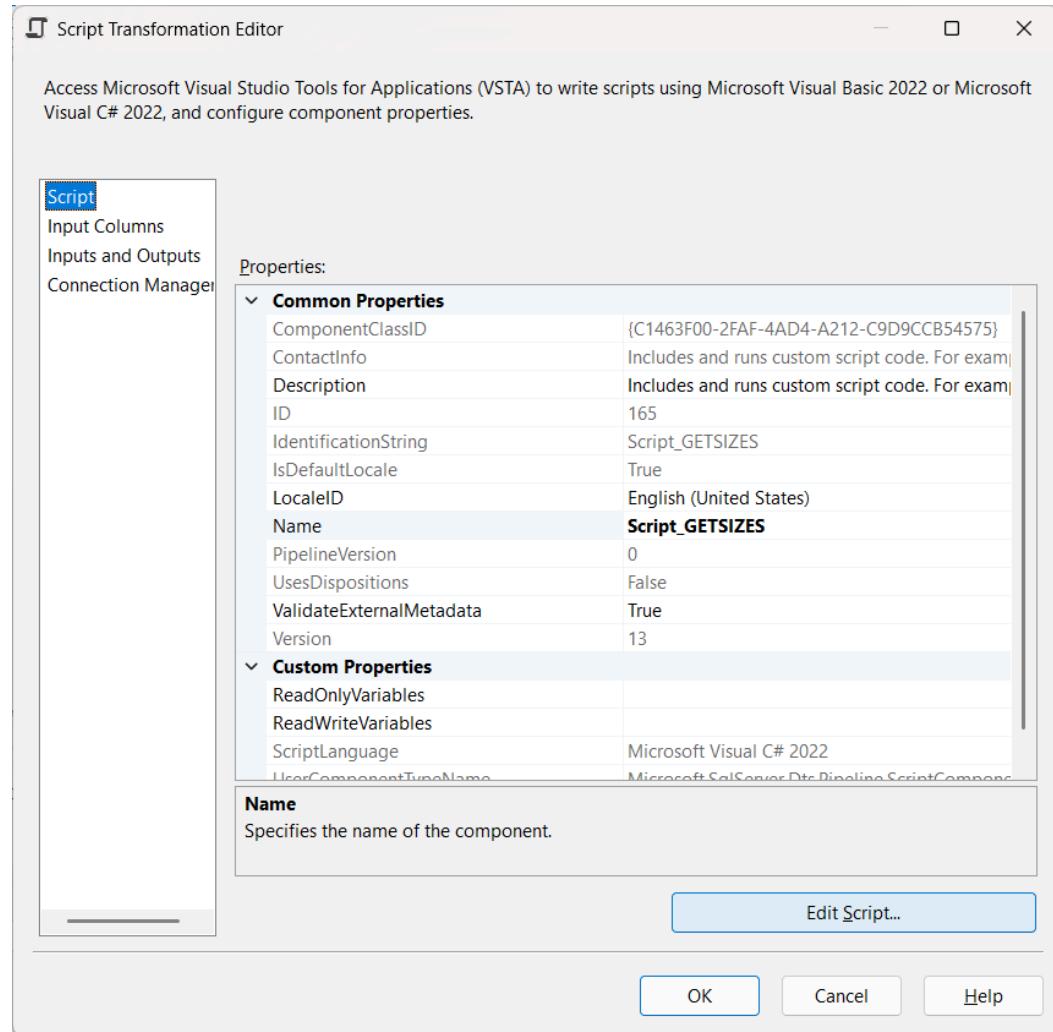
Hình 244. Cài đặt nguồn dữ liệu đầu vào cho chức năng Script\_GETSIZES

- **Bước 17:** Chuyển sang tab **Inputs and Outputs**, ta chọn **Add Output**, và đổi **Output** vừa thêm vào thành **OutputData**. Nhấn đúp chuột **OutputData**, nháy chuột vào **Output Columns**, chọn **Add Column** để thêm các cột khi xuất ra.



Hình 245. Cài đặt dữ liệu đầu ra cho chức năng Script\_GETSIZES

- **Bước 18:** Chuyển sang tab **Script**, chọn **Edit Script..**.



Hình 246. Chọn Edit Script để sửa đổi câu lệnh

- **Bước 19:** Một cửa sổ mới sẽ hiện lên, ta sửa đổi mã lệnh (C#) lại như sau

The screenshot shows the Microsoft Visual Studio IDE interface. The top menu bar includes File, Edit, View, Git, Project, Build, Debug, Test, Analyze, Tools, Extensions, Window, Help, and Search. The title bar displays "VstaProjects". The left sidebar has "SSIS Toolbox" and "Toolbox" tabs, with "Toolbox" currently selected. The main code editor window shows C# code for a Script Task named "ScriptMain". The code is as follows:

```
    /// Example of writing a value to a column in the row;
    /// Row.ZipCode = zipCode
    /// <summary>
    /// <param name="Row">The row that is currently passing through the component</param>
    /// </summary>
    public override void Input0_ProcessInputRow(Input0Buffer Row)
    {
        string[] sizes = Row.Sizes.Split('|');
        foreach (string s in sizes)
        {
            OutputDataBuffer.AddRow();
            OutputDataBuffer.ColorID = Row.ColorID;
            OutputDataBuffer.DescriptionEN = Row.DescriptionEN;
            OutputDataBuffer.ProductionCast = Row.ProductionCast;
            OutputDataBuffer.ProductionID = Row.ProductionID;
            OutputDataBuffer.CategoryID = Row.CategoryID;
            OutputDataBuffer.SubCategoryID = Row.SubCategoryID;
        }
    }
}
```

The Solution Explorer on the right shows a project named "SSIS-Project1" with files like "ScriptComponent.cs", "BufferWrapper.cs", and "ComponentWrapper.cs". The bottom status bar indicates "Ready" and shows navigation keys like Lnc 1, Chc 1, SPC, LF, and a GitHub Copilot status.

Hình 247. Sửa đổi code C# để lấy kích thước

- **Bước 20:** Tiến hành **build** phần **code** đã sửa đổi

The screenshot shows the Microsoft Visual Studio IDE interface. The code editor displays C# code for a component named `SC_e1f7e4a979f74d63aafb1dbfa6161e70`. The code includes methods for `PreExecute` and `PostExecute`, both of which contain comments about their purpose and usage. The Solution Explorer on the right shows the project structure with files like `main.cs` and `ComponentWrappers.cs`. The GitHub Copilot logo is visible in the top right corner.

```
Run Code Analysis on Solution Alt+I11
Build SC_e1f7e4a979f74d63aafb1dbfa6161e70 Ctrl+B
Rebuild SC_e1f7e4a979f74d63aafb1dbfa6161e70
Clean SC_e1f7e4a979f74d63aafb1dbfa6161e70
Run Code Analysis on SC_e1f7e4a979f74d63aafb1dbfa6161e70

Configuration Manager... 16    /// <summary>
17    /// This is the class to which to add your code. Do not change the name, attributes, or parent
18    /// of this class.
19    /// </summary>
20    [Microsoft.SqlServer.Dts.Pipeline.SSIScriptComponentEntryPointAttribute]
21
22    public class ScriptMain : UserComponent
23    {
24        // Help: Using Integration Services variables and parameters
25        // Help: Using Integration Services Connection Managers
26        // Help: Firing Integration Services Events
27
28        /// <summary>
29        /// This method is called once, before rows begin to be processed in the data flow.
30        /// You can remove this method if you don't need to do anything here.
31        /// </summary>
32
33        public override void PreExecute()
34        {
35            base.PreExecute();
36            /* Add your code here
37            */
38        }
39
40        /// <summary>
41        /// This method is called after all the rows have passed through this component.
42        /// You can delete this method if you don't need to do anything here.
43        /// </summary>
44
45    }
46
47    // No issues found
48
49
50 %
```

Hình 248. Tiến hành build code vừa sửa đổi

Kết quả khi build thành công

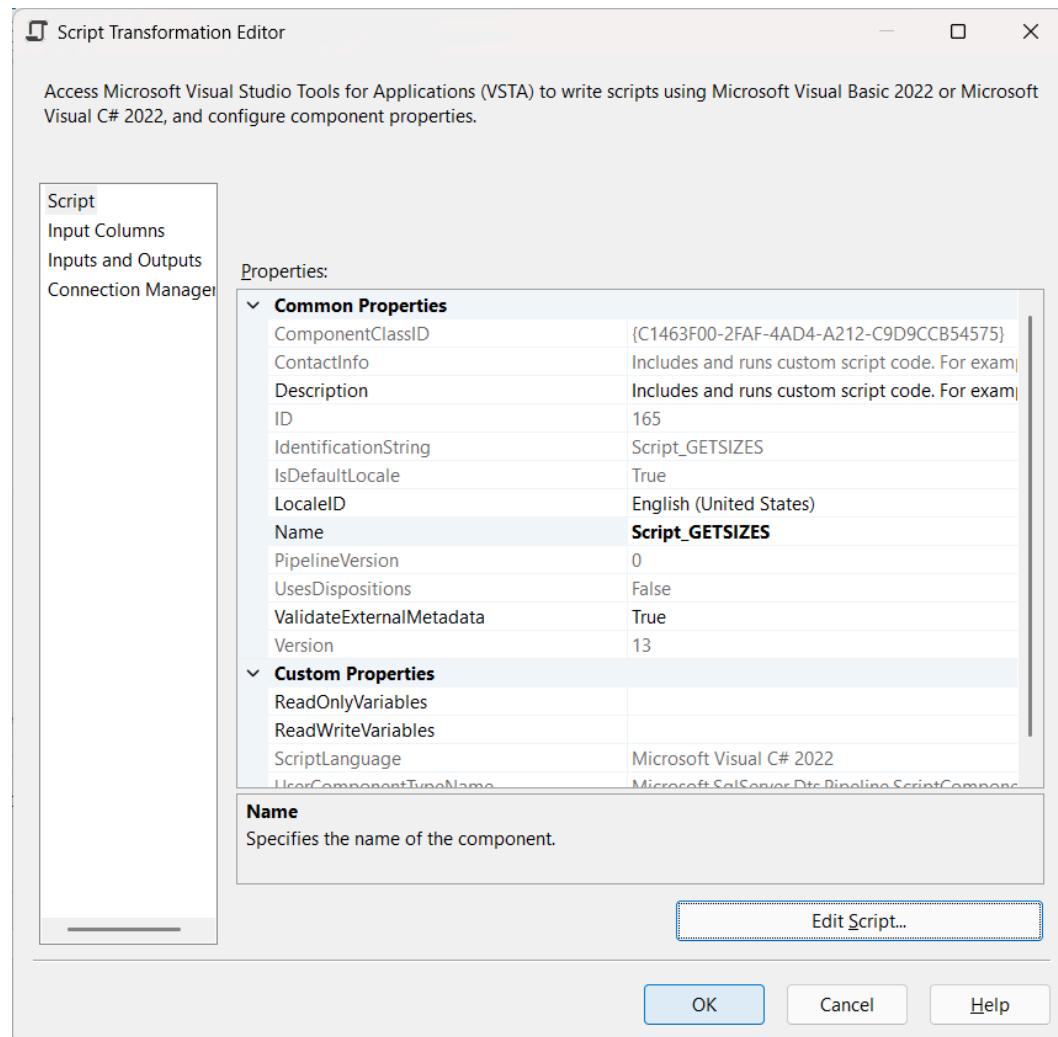
```
main.cs  X SC_e7117e4a79f74d63aafb1dbfa51fe70  ScriptMain  PreExecute()
This tool window can only be used by a SQL Server Integration Services package document.

1  > Help: Introduction to the Script Component
2  8  > Namespaces
3  15
4  16  </summary>
5  17  // This is the class to which to add your code. Do not change the name, attributes, or parent
6  18  // of this class.
7  19  // 
8  20  [Microsoft.SqlServer.Dts.Pipeline.SSIScriptComponentEntryPointAttribute]
9  21  <references>
10  22  <public class ScriptMain : UserComponent
11  23  {
12  24  <summary>
13  25  // Help: Using Integration Services variables and parameters
14  26  // Help: Using Integration Services Connection Manager
15  27  // Help: Firing Integration Services Events
16  28
17  29  /// <summary>
18  30  /// This method is called once, before rows begin to be processed in the data flow.
19  31  /// You can remove this method if you don't need to do anything here.
20  32  /// </summary>
21  33
22  34  public override void PreExecute()
23  35  {
24  36  base.PreExecute();
25  37  // * Add your code here
26  38  //}
27  39
28  40  /// <summary>
29  41  /// This method is called after all the rows have passed through this component.
30  42  /// You can delete this method if you don't need to do anything here.
31  43  /// </summary>
32  44
33  45
34  46
35  47
36  48
37  49
38  50
39  51
40  52
41  53
42  54
43  55
44  56
45  57
46  58
47  59
48  60
49  61
50  62
51  63
52  64
53  65
54  66
55  67
56  68
57  69
58  70
59  71
60  72
61  73
62  74
63  75
64  76
65  77
66  78
67  79
68  80
69  81
70  82
71  83
72  84
73  85
74  86
75  87
76  88
77  89
78  90
79  91
80  92
81  93
82  94
83  95
84  96
85
86
87
88
89
90
91
92
93
94
95
96
No issues found
```

Output  
Show output from: Build  
Build started: Project: SC\_e7117e4a79f74d63aafb1dbfa51fe70  
Build succeeded

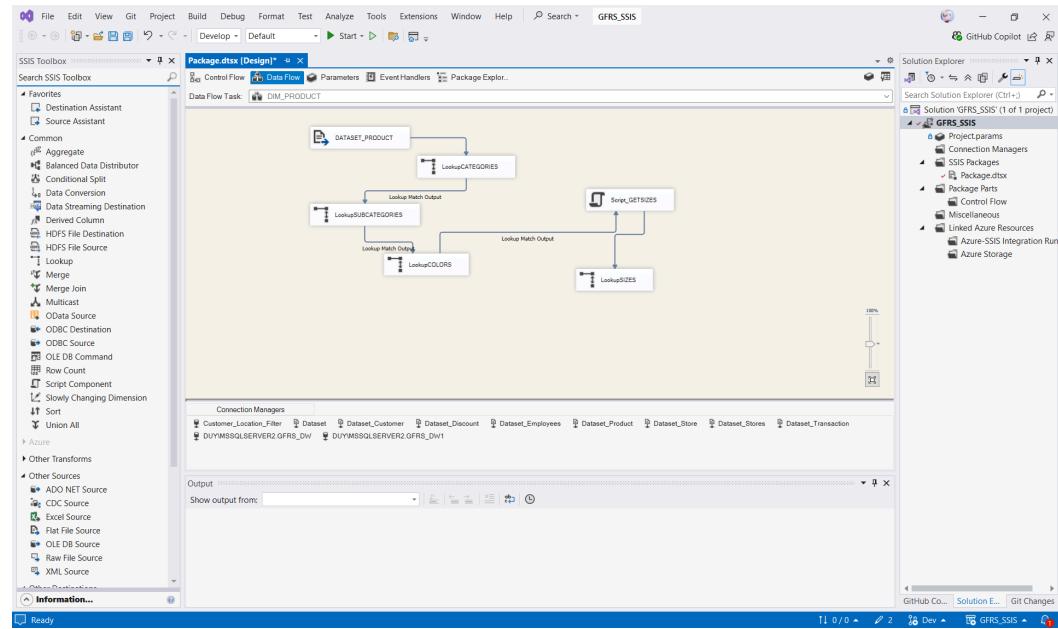
Hình 249. Code được build thành công

- **Bước 21:** Đóng cửa sổ code . Chọn **OK** và thoát cửa sổ **Script Transformation Editor**



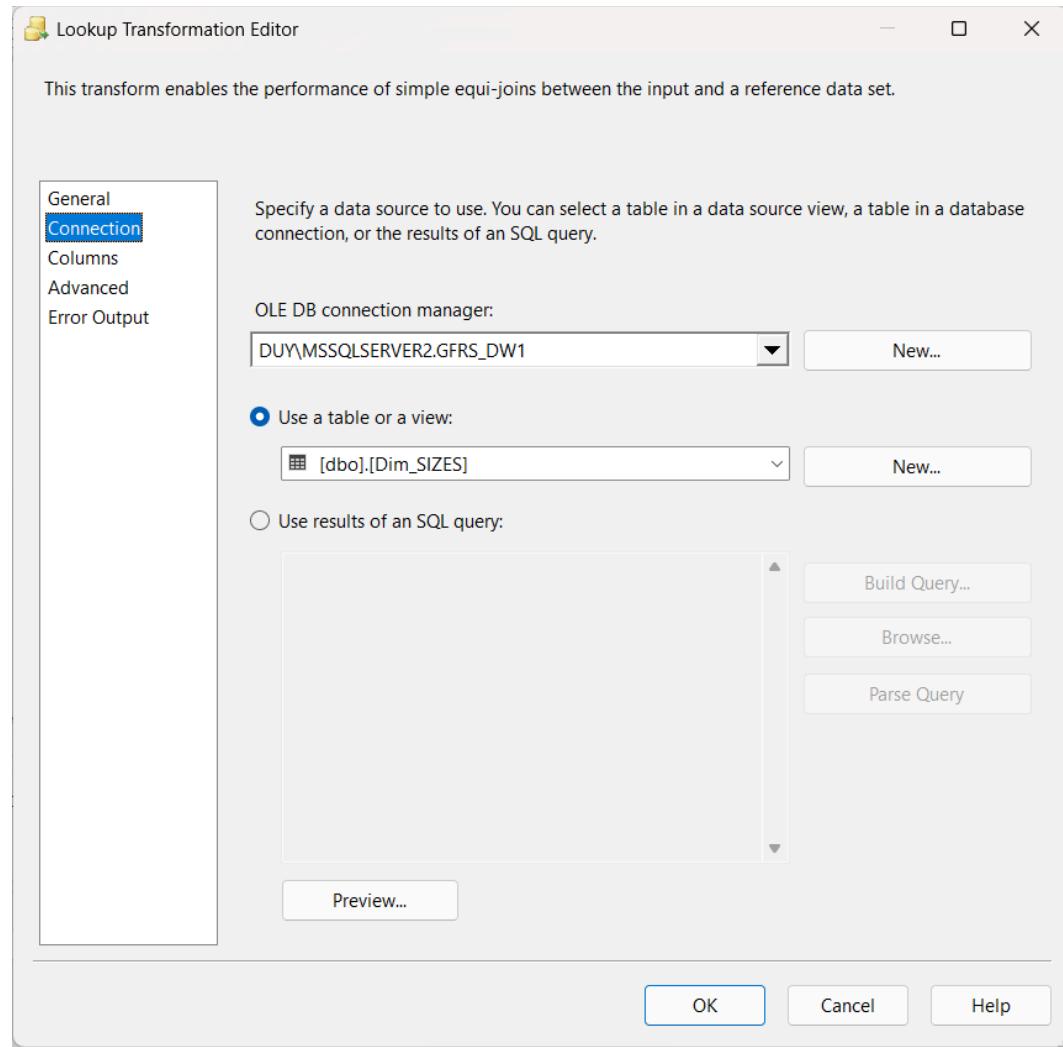
Hình 250. chọn OK để thoát chúc năng Script Transformation Editor

- **Bước 22:** Kéo chúc năng **Lookup** vào vùng **Data Flow** và đổi tên thành **LookupSIZES**. Kéo mũi tên luồng dữ liệu từ **Script\_GETSIZES** vào **LookupSIZES**. Một hộp thoại sẽ xuất hiện, chọn **OutputData** và nhấn **OK**.



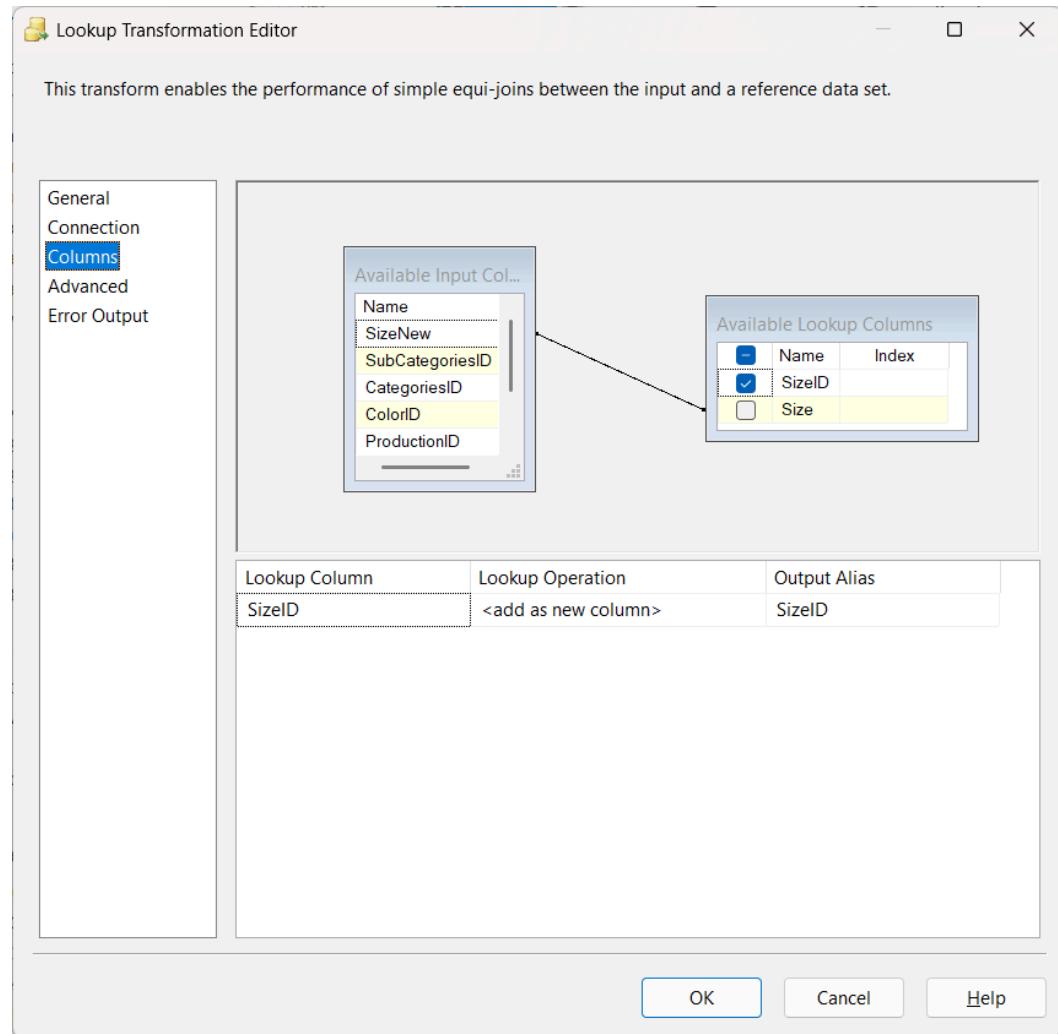
Hình 251. Tạo chức năng LookupSIZES trong Dim\_DISCOUNTS

- **Bước 23:** Nhấn đúp chuột vào **LookupSIZES**, màn hình **Lookup Transformation Editor** xuất hiện. Do chúng ta đã thiết lập **OLE DB connection manager** ở bước trước, nên ta có thể sử dụng lại. Ở phần **Use a table or a view**, tiến hành chọn **[dbo].[Dim\_SIZES]**.



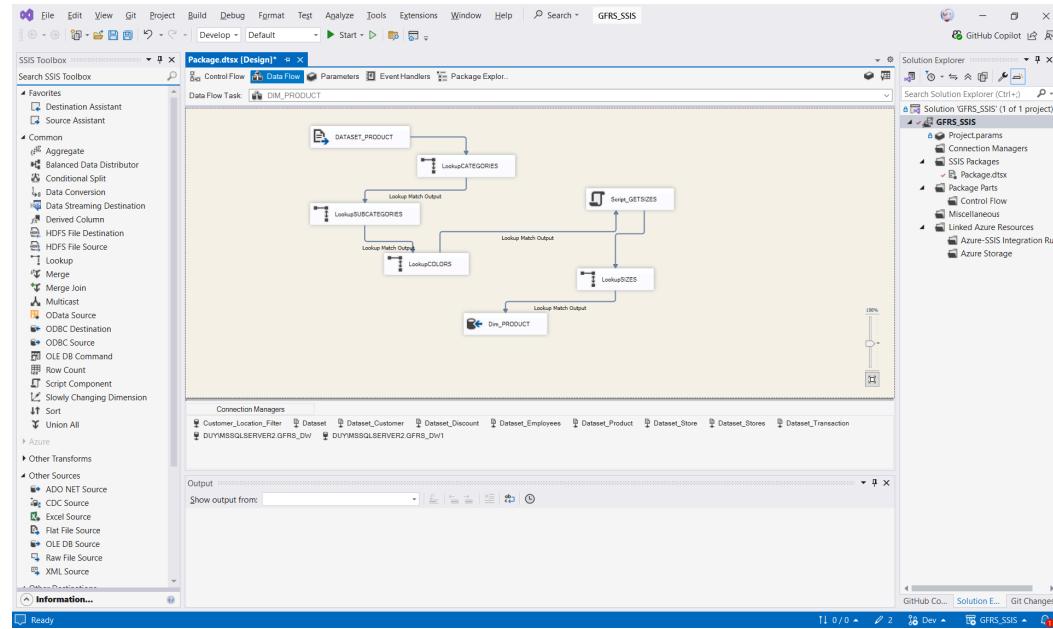
Hình 252. Kết nối nguồn dữ liệu cho chức năng LookupSIZEs

- **Bước 24:** Trong tab **Columns**, ở bảng bên phải click chọn **SizeID**. Ta kéo **SizeNew** từ bảng **Available Input Columns** qua tương ứng **Size** bảng **Available Lookup Columns** (quá trình này giúp hệ thống xác định vùng dữ liệu cần tìm kiếm). Sau đó bấm **OK** để đóng cửa sổ **Lookup Transformation Editor**



Hình 253. Cài đặt chức năng LookupSIZES

- **Bước 25:** Kéo chức năng **ADO NET Destination** vào vùng **Data Flow** và đổi tên thành **Dim\_PRODUCTS**. Kéo mũi tên luồng dữ liệu từ **LookupSIZES** vào **Dim\_PRODUCTS**. Một cửa sổ hiện lên, ta chọn **Lookup Match Output** và nhấn **OK**



Hình 254. Tạo chức năng ADO NET Destination Dim\_PRODUCTS

- Bước 26: Vào SQL Server 2022, chạy lệnh DDL để tạo Dim\_PRODUCTS

The screenshot shows the Microsoft SQL Server Management Studio (SSMS) interface. The Object Explorer on the left lists the database structure, including databases like GFRS\_DW and DW\_DLCK\_HK2\_2023\_2024. The main area contains two query windows:

- Query1.s...\\maing (73)\***: Contains T-SQL code for creating two tables: Dim\_SIZES and Dim\_PRODUCTS. The Dim\_PRODUCTS table includes columns for ProductID, ProductHash, Description, CategoryID, ColorID, SubCategoryID, SizeID, and ProductionCost.
- DB\_New - 1.sql - \\maing (108)**: Contains a single GO statement.

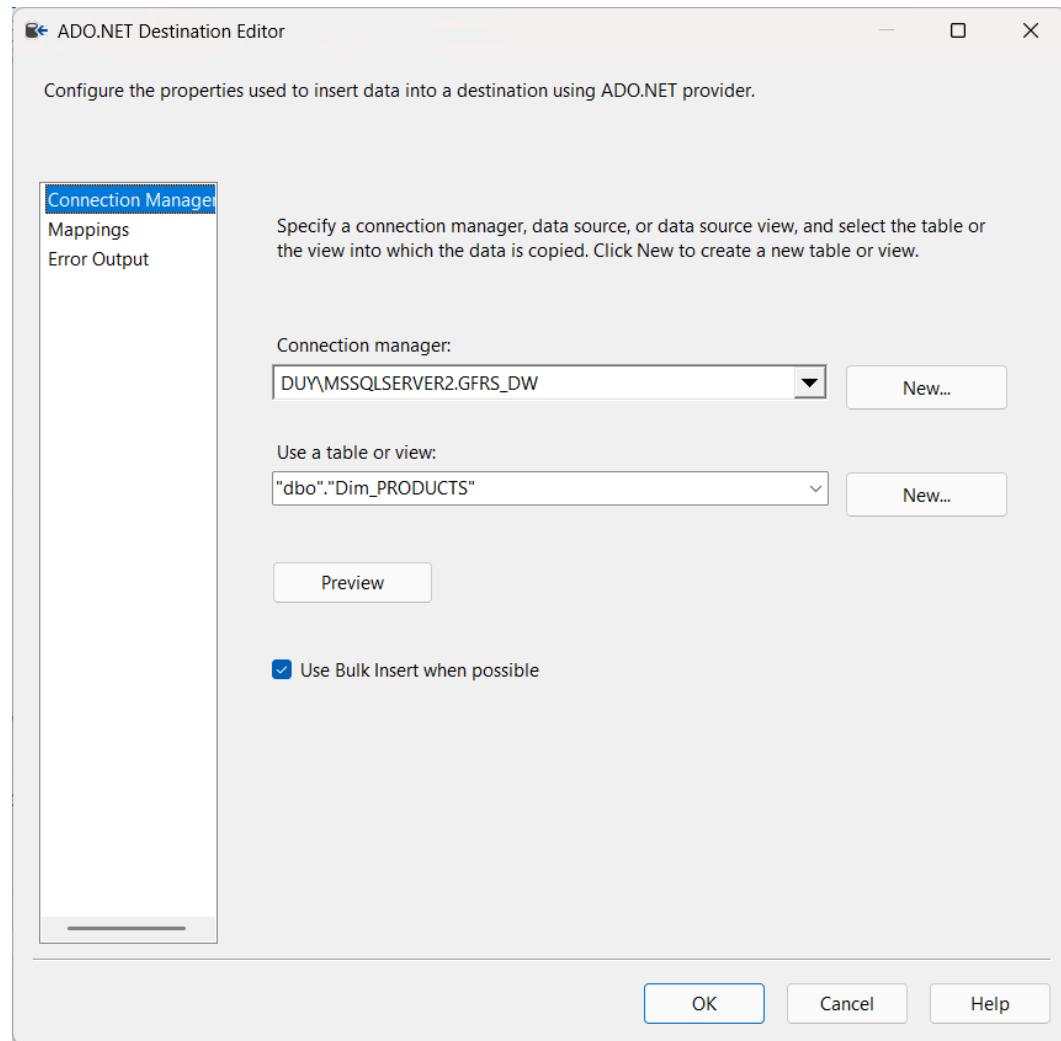
The results pane at the bottom shows a table with 12 rows of data from the Dim\_PRODUCTS table, with columns for ProductID, Start, End, Discount, Description, SubCategoryID, and CategoryID. The status bar at the bottom right indicates the execution was successful with 802 rows affected.

DiscountID	Start	End	Discount	Description	SubCategoryID	CategoryID
1	NULL	NULL	0.00	NULL	NULL	NULL
2	2020-10-01 00:00:00.000	2020-10-10 00:00:00.000	0.20	20% discount during our Autumn Essentials Sale	1	1
3	2020-10-01 00:00:00.000	2020-10-10 00:00:00.000	0.20	20% discount during our Autumn Essentials Sale	16	1
4	2020-10-01 00:00:00.000	2020-10-10 00:00:00.000	0.20	20% discount during our Autumn Essentials Sale	13	2
5	2020-10-01 00:00:00.000	2020-10-10 00:00:00.000	0.20	20% discount during our Autumn Essentials Sale	17	2
6	2020-10-01 00:00:00.000	2020-10-10 00:00:00.000	0.20	20% discount during our Autumn Essentials Sale	13	3
7	2020-10-01 00:00:00.000	2020-10-10 00:00:00.000	0.20	20% discount during our Autumn Essentials Sale	18	3
8	2021-10-01 00:00:00.000	2021-10-10 00:00:00.000	0.20	20% discount during our Autumn Essentials Sale	1	1
9	2021-10-01 00:00:00.000	2021-10-10 00:00:00.000	0.20	20% discount during our Autumn Essentials Sale	16	1
10	2021-10-01 00:00:00.000	2021-10-10 00:00:00.000	0.20	20% discount during our Autumn Essentials Sale	13	2
11	2021-10-01 00:00:00.000	2021-10-10 00:00:00.000	0.20	20% discount during our Autumn Essentials Sale	17	2
12	2021-10-01 00:00:00.000	2021-10-10 00:00:00.000	0.20	20% discount during our Autumn Essentials Sale	13	3

Hình 255. Chạy lệnh DDL tạo Dim PRODUCTS trong kho dữ liệu

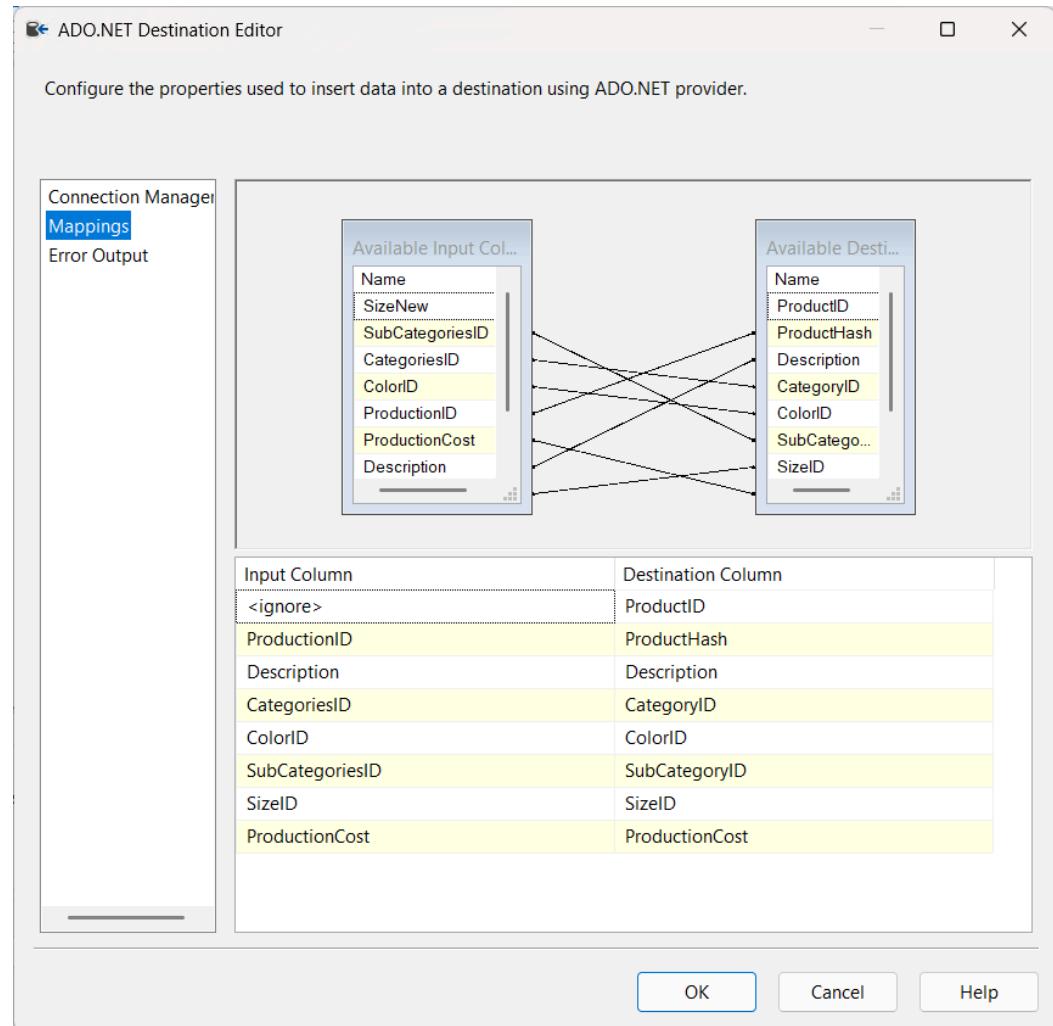
- **Bước 27** :Nhấn đúp chuột vào **Dim\_DISCOUNTS**. Trong tab **Connection Manager** của màn hình **ADO.NET Destination Editor**, ở **Use a table or view**

view tiến hành nhấn chọn “**dbo”.”Dim\_PRODUCTS**”, cũng chính là bảng vừa được tạo bằng lệnh **DDL** ở **SQL Server 2022**



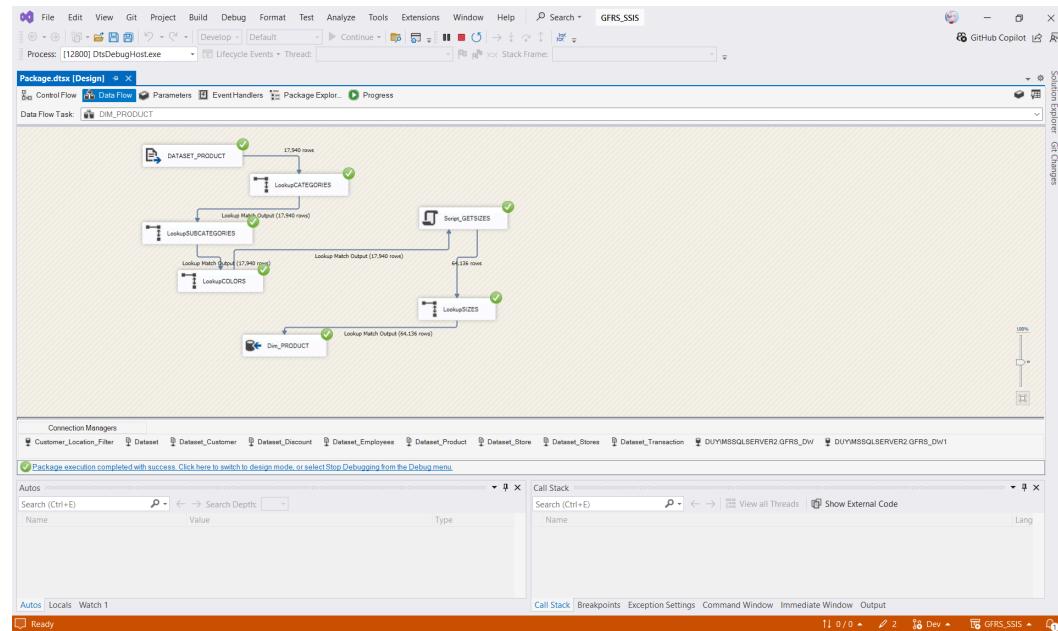
Hình 256. Kết nối đích đến cho ADO NET Destination Dim\_PRODUCTS

- **Bước 28:** Ở tab **Mapping** của màn hình **ADO.NET Destination Editor** tiến hành chọn các thuộc tính thích hợp ở cột bên trái để ánh xạ với các thuộc tính ở cột bên phải.



Hình 257. Cài đặt ADO NET Destination Dim\_PRODUCTS

- **Bước 28:** Nhấn Start để kiểm tra quá trình tạo và đổ dữ liệu vào Dim\_PRODUCTS có thành công hay chưa.



Hình 258. Đồ dữ liệu vào Dim\_PRODUCTS

Kết quả hiển thị cho thấy quá trình diễn ra thành công và có **64,136 rows** được đỗ vào **DIM\_PRODUCTS**.

- **Bước 29:** Vào SQL Server 2022 kiểm tra và thấy dữ liệu đã được đổ vào Dim\_PRODUCTS thành công.

The screenshot shows the Microsoft SQL Server Management Studio (SSMS) interface. The Object Explorer on the left shows the database structure for 'DUY1MSSQLSERVER2' (SQL Server 16). The 'Databases' node is expanded, showing 'System Databases', 'Database Snapshots', and several user databases including 'DW\_DLCK\_HK2\_2023\_2024', 'GFRS\_DW', 'OLAP\_BT1\_DW', 'OLAP\_CK\_2\_HK2\_2023\_2024', and 'OLAP\_CK\_HK2\_2023\_2024'. The 'Security' node is also expanded. The 'Solution Explorer' tab at the top right shows 'Solution1'.

The main window displays a T-SQL script for creating dimension tables:

```
112 [Description] VARCHAR(255),
113 [CategoryID] INT,
114 [ColorID] INT,
115 [SubCategoryID] INT,
116 [SizeID] INT,
117 [ProductionCost] DECIMAL(18, 2)
118 )
119 GO
120 Select * from Dim_PRODUCTS;
121
122
123 CREATE TABLE [Dim_TRANSACTIONTYPE] (
124     [TransactionTypeID] INT IDENTITY(1,1) PRIMARY KEY,
125     [Transaction] VARCHAR(255)
126 )
127 GO
128
129 CREATE TABLE [Dim_PAYMENTMETHOD] (
130     [PaymentMethodID] INT IDENTITY(1,1) PRIMARY KEY,
131     [PaymentMethod] VARCHAR(255)
132 )
133 GO
```

The status bar at the bottom indicates '19 %' completion, 'No issues found', and 'Ln: 120 Ch: 1 SPC LF'.

The results pane shows the data for the 'Dim\_PRODUCTS' table:

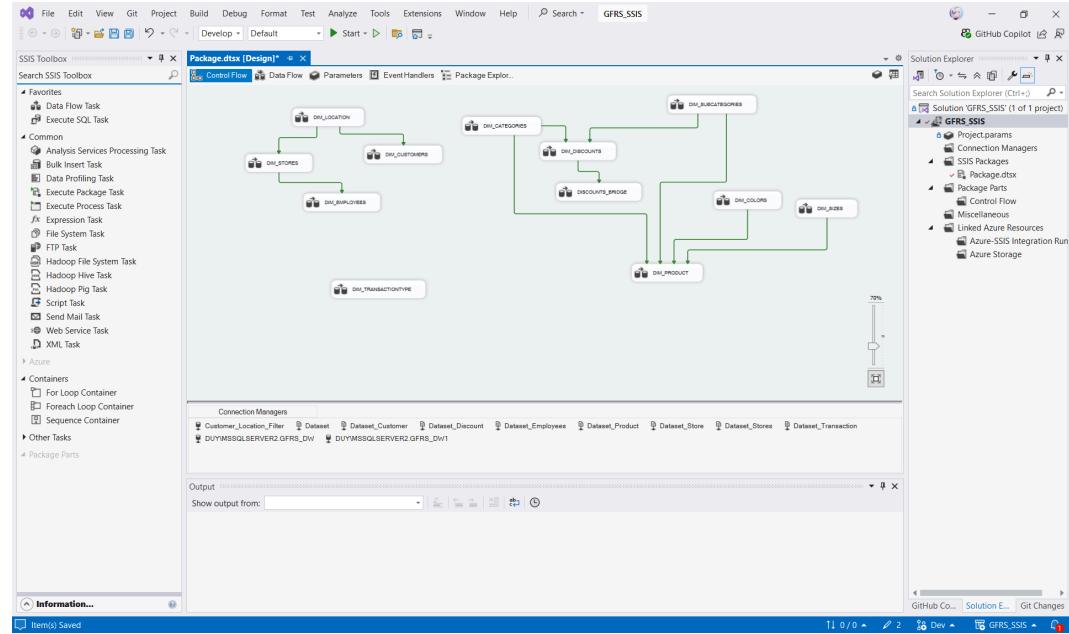
ProductID	ProductName	Description	CategoryID	ColorID	SubCategoryID	SizeID	ProductionCost
1	1	Sports Velour Sports With Buttons	2	1	4	14	11.00
2	2	Sports Velour Sports With Buttons	2	1	4	12	11.00
3	3	Sports Velour Sports With Buttons	2	1	4	11	11.00
4	4	Sports Velour Sports With Buttons	2	1	4	15	11.00
5	5	Luxurious Pink Denim With Buttons	2	11	17	14	20.00
6	6	Luxurious Pink Denim With Buttons	2	11	17	12	20.00
7	7	Luxurious Pink Denim With Buttons	2	11	17	11	20.00
8	8	Luxurious Pink Denim With Buttons	2	11	17	15	20.00
9	9	Black Tricot Printed Tricot	2	3	5	14	26.00
10	10	Black Tricot Printed Tricot	2	3	5	12	26.00
11	11	Black Tricot Printed Tricot	2	3	5	11	26.00
12	12	Black Tricot Printed Tricot	2	3	5	15	26.00

The status bar at the bottom right indicates 'Query executed successfully.' and the connection details: 'D:\MSSQLSERVER2 (16.0 RTM) DUY1\maing (73) GFRS\_DW 000.0000 64136 rows'.

Hình 259. Dữ liệu được đổ thành công vào Dim PRODUCTS

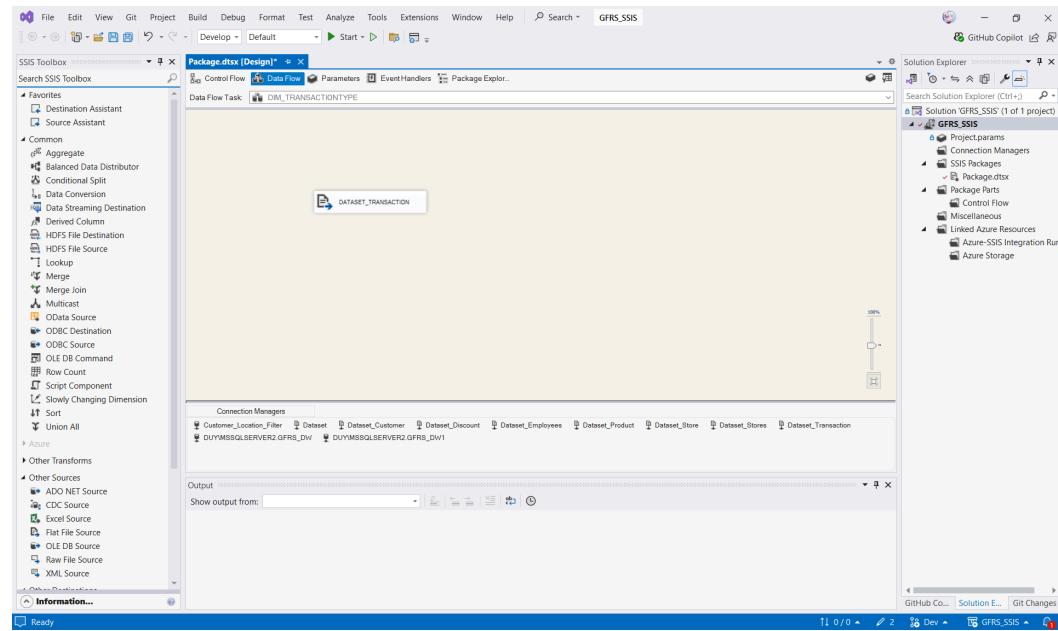
### II.3.11. Dim\_TRANSACTIONTYPE

- **Bước 1:** Kéo chức năng Data Flow Task trong vùng SSIS Toolbox vào vùng Control Flow và đổi tên thành **Dim\_TRANSACTIONTYPE**.



Hình 260. Tạo chức năng Dim\_TRANSACTIONTYPE trong Control Flow

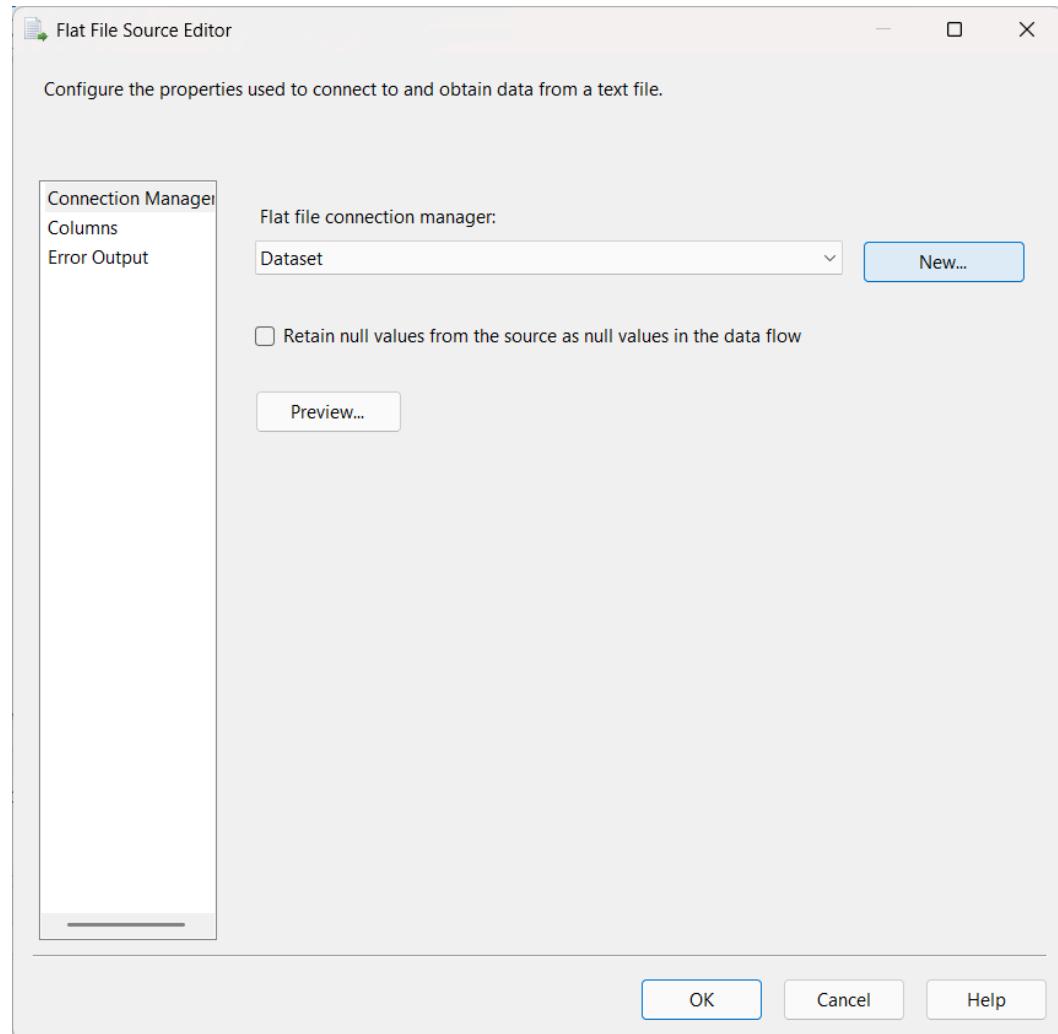
- **Bước 2:** Nhấn đúp chuột vào **Dim\_TRANSACTIONTYPE**. Sau đó kéo chức năng **Flat File Source** vào vùng làm việc và đổi tên thành **DATASET\_TRANSACTION**



*Hình 261. Tạo DATASET\_TRANSACTION trong*

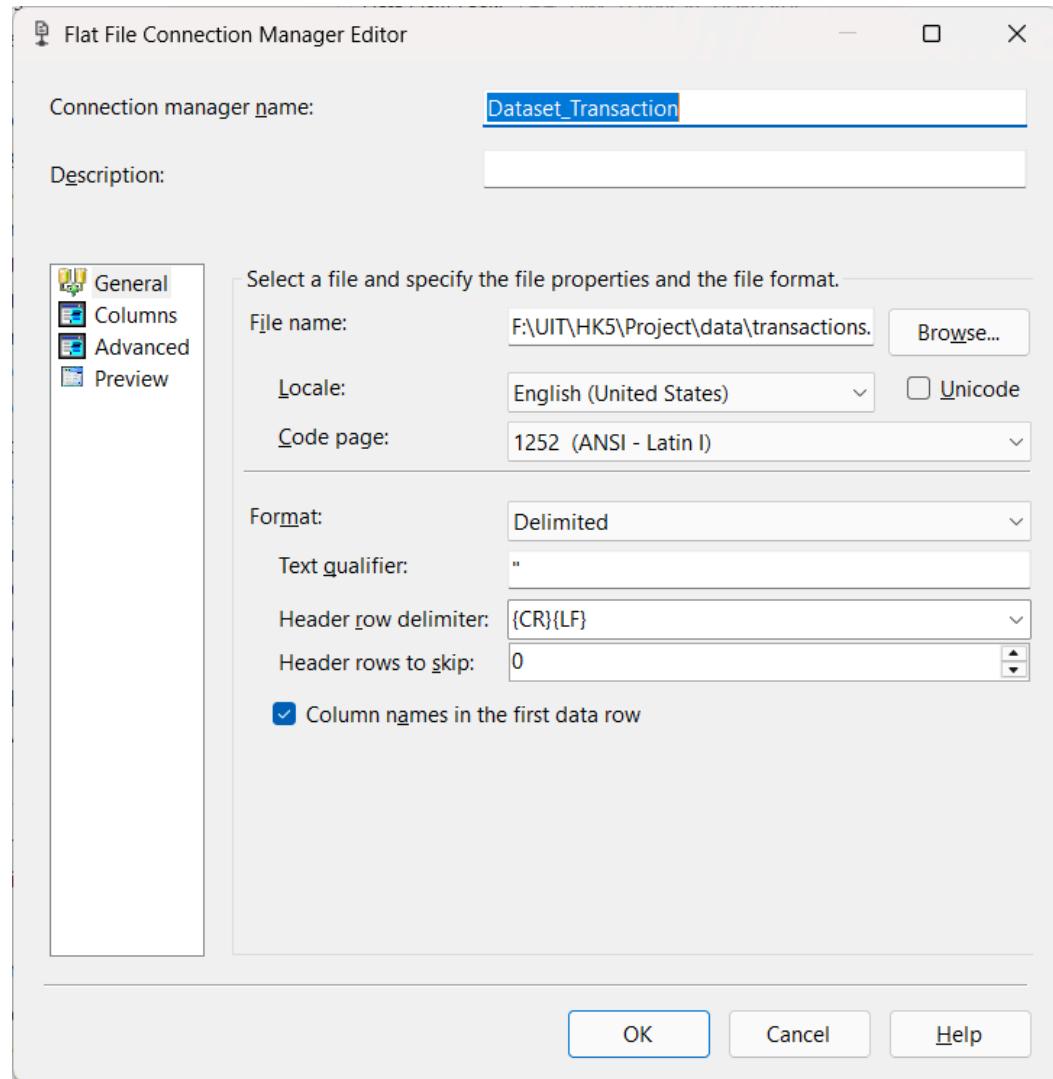
*Dim\_TRANSACTIONTYPE*

- **Bước 3:** Nhấn đúp chuột vào **DATASET\_TRANSACTION**. Màn hình **Flat File Source Editor** hiện ra. Ở tab **Connection Manager** ta chọn **New**.



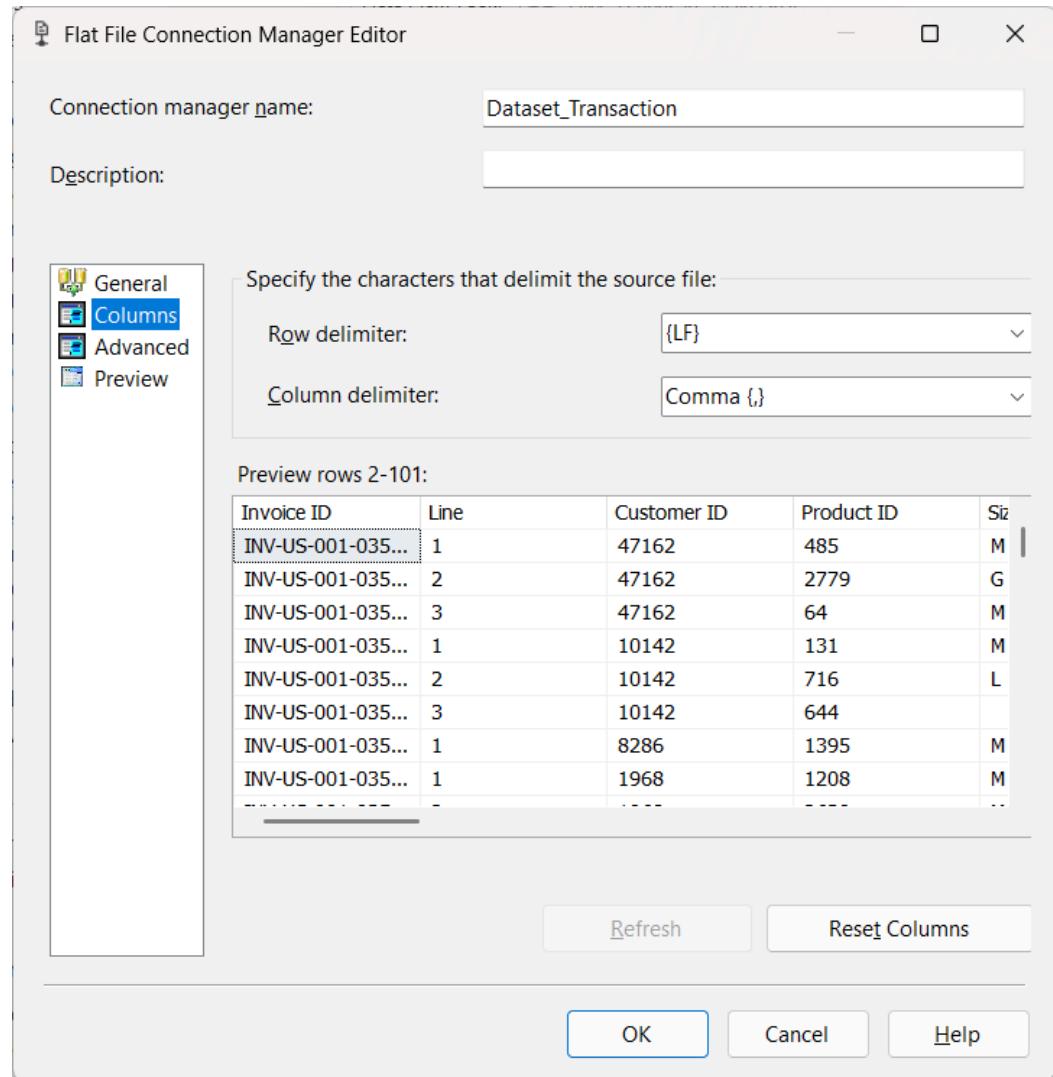
Hình 262. Chọn New để tạo nguồn dữ liệu mới cho  
DATASET\_TRANSACTION

- **Bước 4:** Cửa sổ **Flat File Connection Manager Editor** hiện lên. Ở phần **Connection manager name** ta đổi tên thành **Dataset\_Transaction**. Tiếp đó nhấn **Browse** để chọn file **transactions.csv**. **Text qualifier** để là “



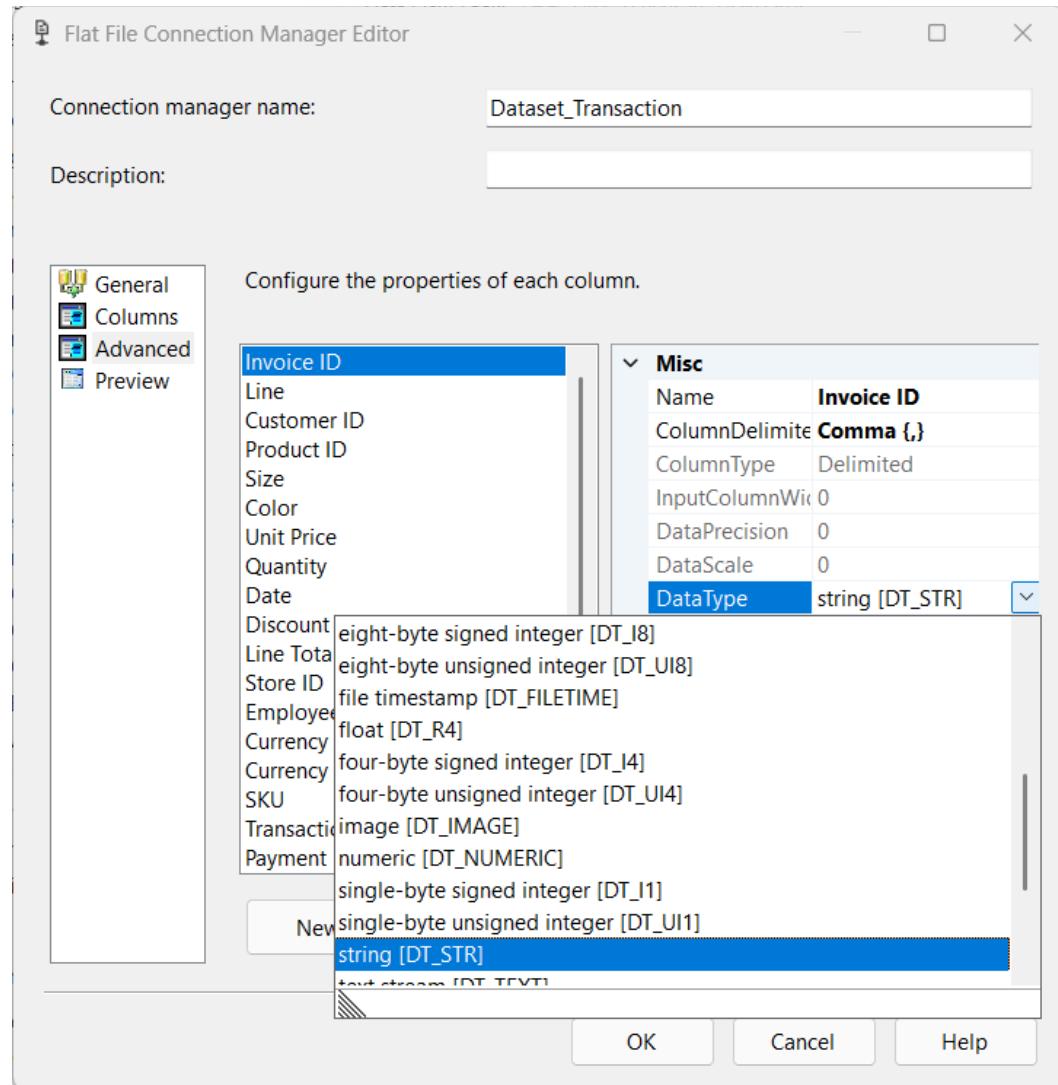
Hình 263. Cấu hình nguồn dữ liệu mới cho DATASET\_TRANSACTION

- **Bước 5:** Chuyển sang tab **Columns**, ta có thể xem dữ liệu sẽ được đổ vào



Hình 264. Kiểm tra nguồn dữ liệu đầu vào file transactions.csv

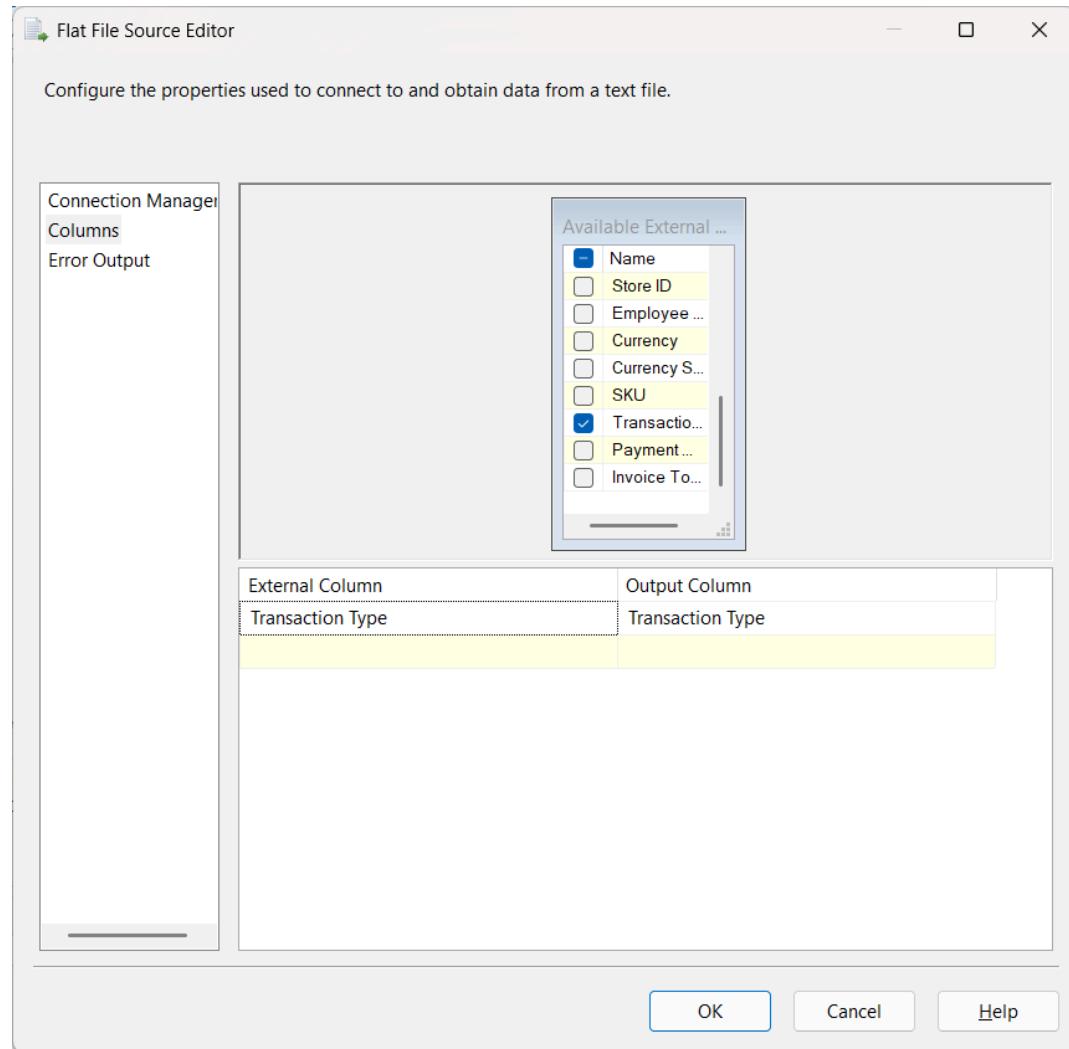
- **Bước 6:** Chuyển sang tab **Advanced**, ta sẽ chọn kiểu dữ liệu phù hợp trước khi nạp. Sau đó chọn **OK** và kết thúc cửa sổ **Flat File Connection Manager Editor**.



Hình 265. Cài đặt kiểu dữ liệu cho các thuộc tính đầu vào

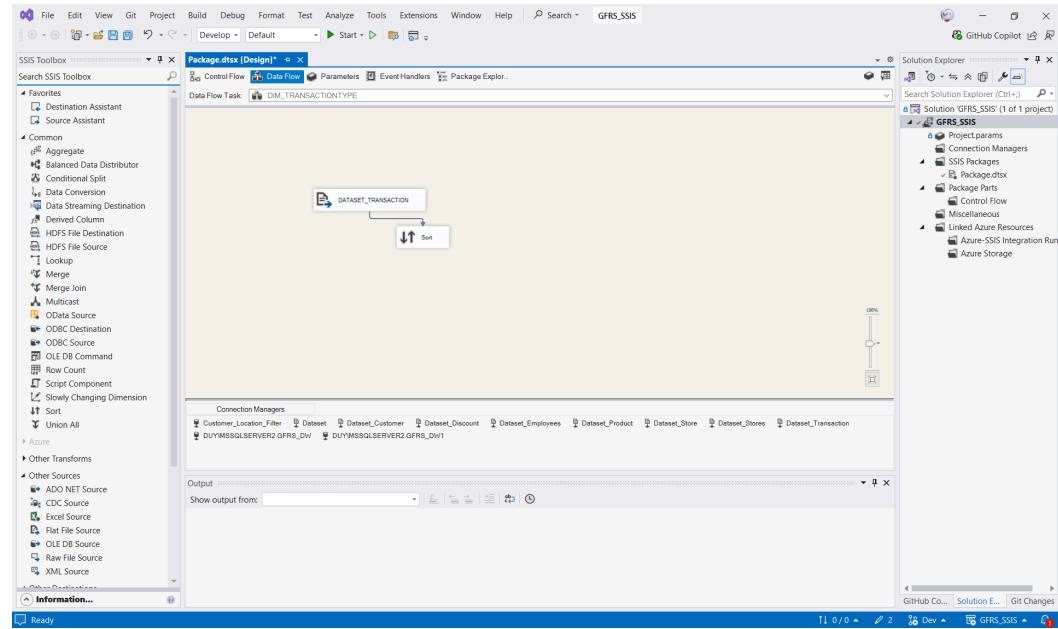
DATASET\_TRANSACTION

- **Bước 7:** Ở tab **Columns** ta chọn cột **Color** để tiến hành dữ liệu vào **Dim\_TRANSACTIONTYPE**. Sau đó, nhấn **OK** và kết thúc cửa sổ **Flat File Source Editor**



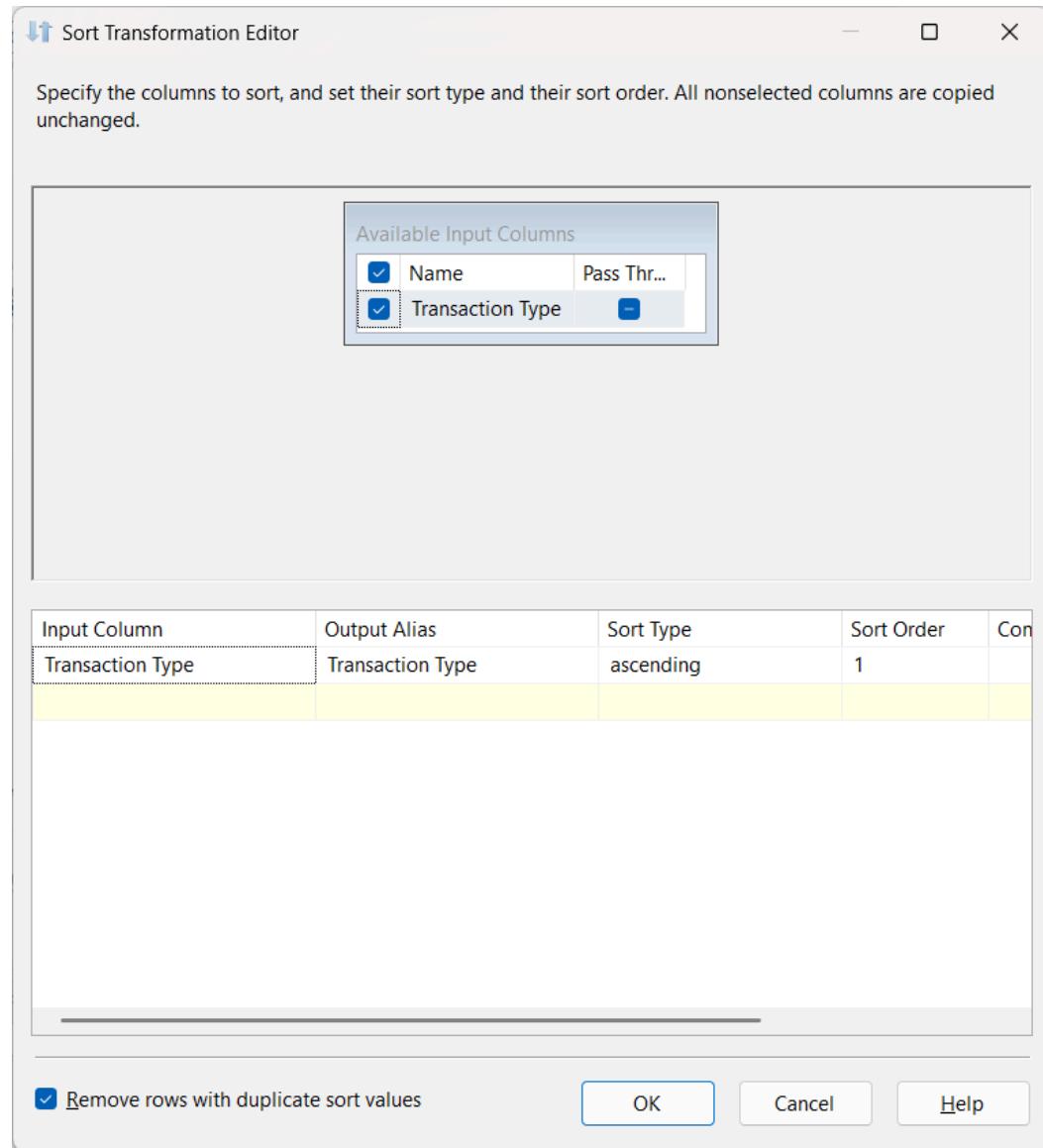
Hình 266. Chọn các thuộc tính cần thiết cho quá trình tạo  
*Dim\_TRANSACTIONTYPE*

- **Bước 8:** Kéo chức năng Sort từ vùng **SSIS Toolbox** vào cùng **Data Flow**. Kéo luồng dữ liệu từ **DATASET\_TRANSACTION** vào Sort.



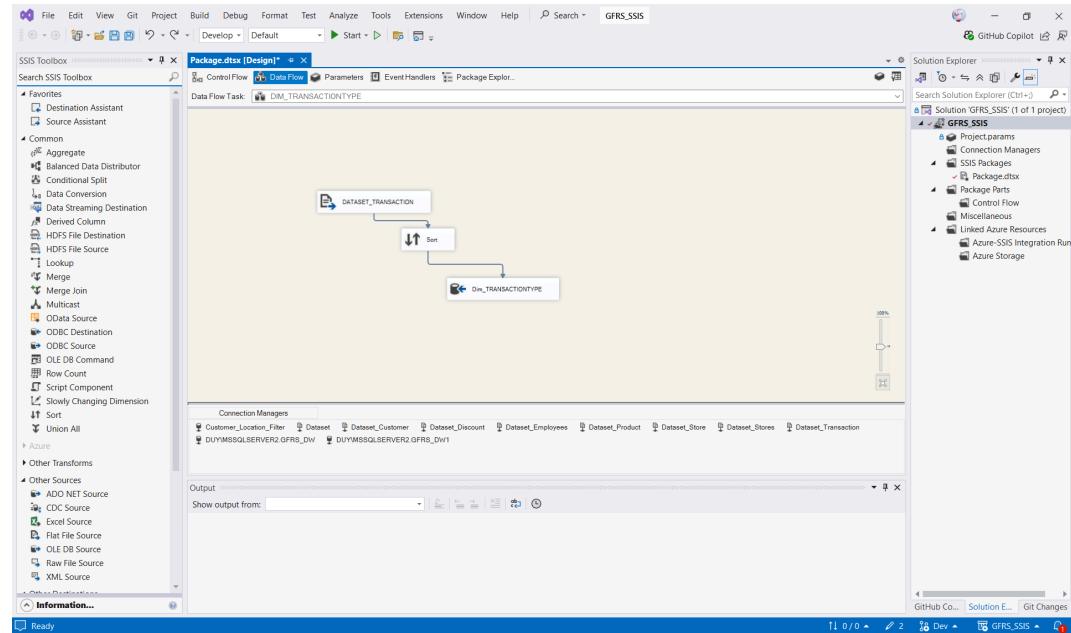
Hình 267. Tạo chức năng Sort trong Dim\_TRANSACTIONTYPE

- **Bước 10:** Nhấn đúp chuột vào Sort, cửa sổ Sort Transformation Editor sẽ hiện lên. Ta chọn cột Transaction Type trên cửa sổ Available Input Columns. Sau đó chọn Remove rows with duplicate sort values để lọc những bản ghi trùng lặp. Sau đó nhấn OK và kết thúc cửa sổ Sort Transformation Editor.



Hình 268. Cài đặt chức năng Sort trong Dim\_TRANSACTIONTYPE

- **Bước 11:** Kéo ADO NET Destination từ SSIS Toolbox vào vùng Data Flow và đổi tên thành Dim\_TRANSACTIONTYPE. Kéo luồng dữ liệu từ Sort vào Dim\_TRANSACTIONTYPE.



Hình 269. Tạo ADO NET Destination Dim\_TRANSACTIONTYPE

- **Bước 12:** Vào SQL Server 2022, chạy lệnh DDL để tạo Dim\_TRANSACTIONTYPE.

```

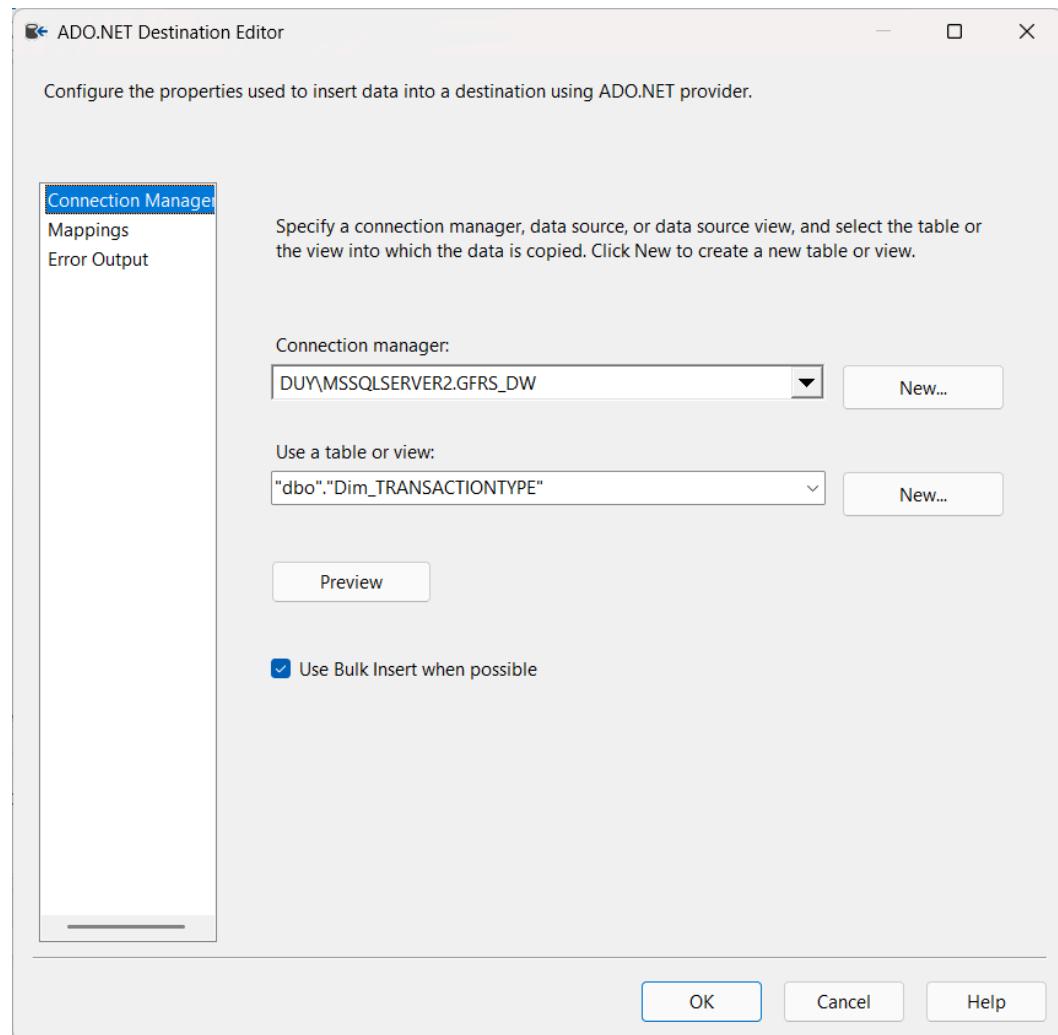
93
94
95
96
97
98
99
100
101
102
103
104
105
121 %  No issues found 4  Ln: 98 Ch: 1 TABS CRLF
Results Messages
ProductID ProductHash CategoryID ColorID SubCategoryID SizeID ProductionCost
1 1 2 1 4 14 11
2 2 1 2 1 4 12 11
3 3 1 2 1 4 11 11
4 4 1 2 1 4 15 11
5 5 2 2 11 17 14 20
6 6 2 2 11 17 12 20
7 7 2 2 11 17 11 20
8 8 2 2 11 17 15 20
9 9 3 2 3 5 14 26
10 10 3 2 3 5 12 26
11 11 3 2 3 5 11 26
12 12 3 2 3 5 15 26
13 13 4 2 1 11 14 28
14 14 4 2 1 11 12 28

```

Query executed successfully.

Hình 270. Chạy lệnh DDL tạo Dim\_TRANSACTIONTYPE trong kho dữ liệu

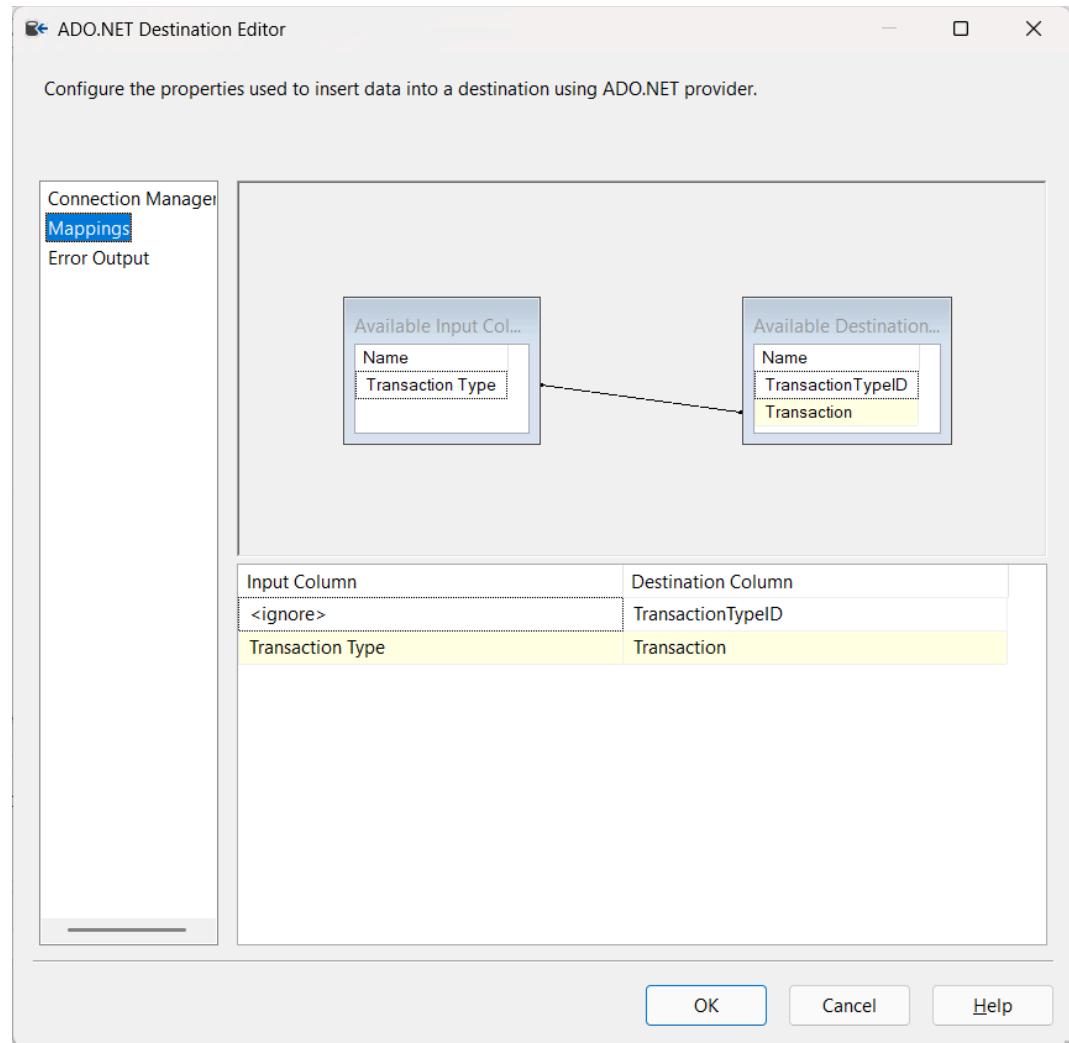
- **Bước 13:** Nhấn đúp chuột vào **Dim\_TRANSACTIONTYPE**. Trong tab **Connection Manager** của màn hình **ADO.NET Destination Editor**, ở **Use a table or view** tiến hành nhấn chọn “**dbo”**.” **Dim\_TRANSACTIONTYPE**, cũng chính là bảng vừa được tạo bằng lệnh **DDL** ở **SQL Server 2022**



*Hình 271. Kết nối đích đến cho ADO.NET Destination*

*Dim\_TRANSACTIONTYPE*

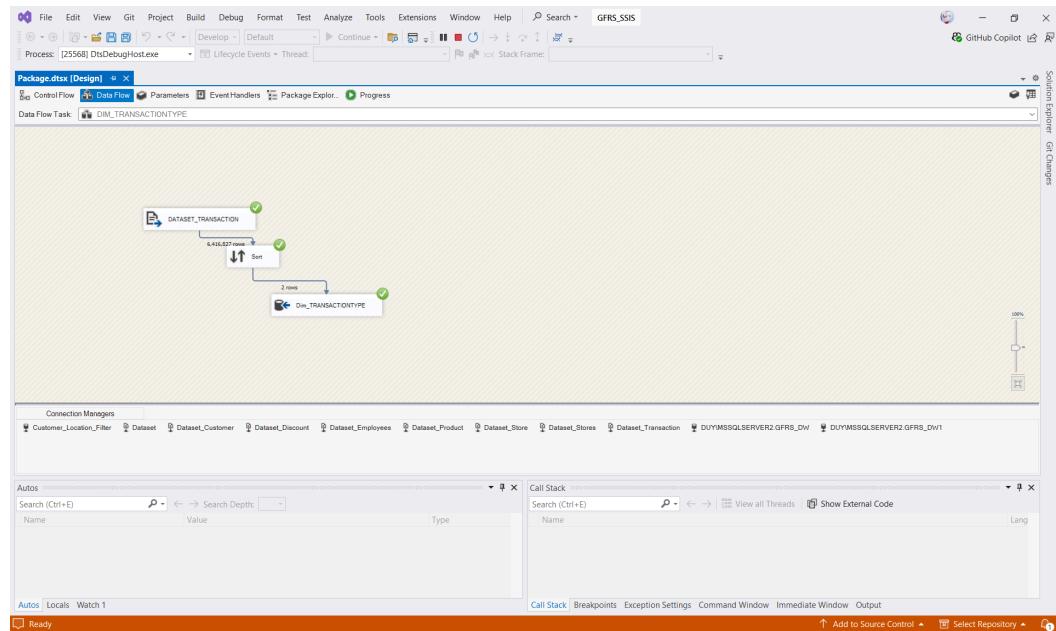
- **Bước 14:** Ở tab **Mapping** của màn hình **ADO.NET Destination Editor** tiến hành chọn các thuộc tính thích hợp ở cột bên trái để ánh xạ với các thuộc tính ở cột bên phải.



Hình 272. Cài đặt chức năng ADO NET Destination

Dim\_TRANSACTIONTYPE

- **Bước 15:** Nhấn Start để kiểm tra quá trình tạo và đổ dữ liệu vào **Dim\_TRANSACTIONTYPE** có thành công hay chưa.



Hình 273. Đỗ dữ liệu vào Dim\_TRANSACTIONTYPE

Kết quả hiển thị cho thấy quá trình diễn ra thành công và có **2 rows** được đỗ vào **Dim\_TRANSACTIONTYPE**.

- **Bước 16:** Vào **SQL Server 2022** kiểm tra và thấy dữ liệu đã được đỗ vào **Dim\_TRANSACTIONTYPE** thành công.

```

File Edit View Query Git Project Tools Extensions Window Help Search Solution1
File Edit View Query Git Project Tools Extensions Window Help Search Solution1
New Query Execute
Object Explorer Database Snapshots Security Server Objects Replication Management Integration Services SQL Server Agent XEvent Profiler
Databases System Databases GFRS_DW
93
94
95
96
97
98
99
100
101
102
103
104
105
-- select * from Dim_PRODUCTS;

drop table if exists Dim_TRANSACTIONTYPE;
CREATE TABLE [Dim_TRANSACTIONTYPE] (
    [TransactionTypeID] integer identity(1,1) PRIMARY KEY,
    [Transaction] varchar(255)
);

select * from Dim_TRANSACTIONTYPE;
drop table if exists Dim_PAYMENTMETHOD;
CREATE TABLE [Dim_PAYMENTMETHOD] (

```

TransactionTypeID	Transaction
1	Return
2	Sale

No issues found

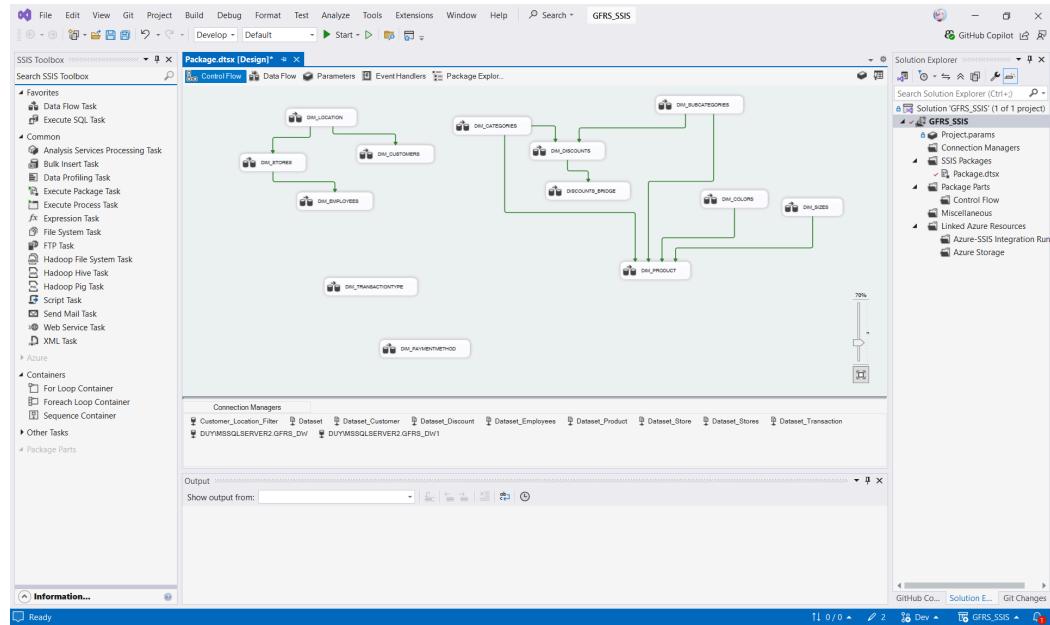
Results Messages

Query executed successfully.

Hình 274. Dữ liệu được đỗ thành công vào Dim\_TRANSACTIONTYPE

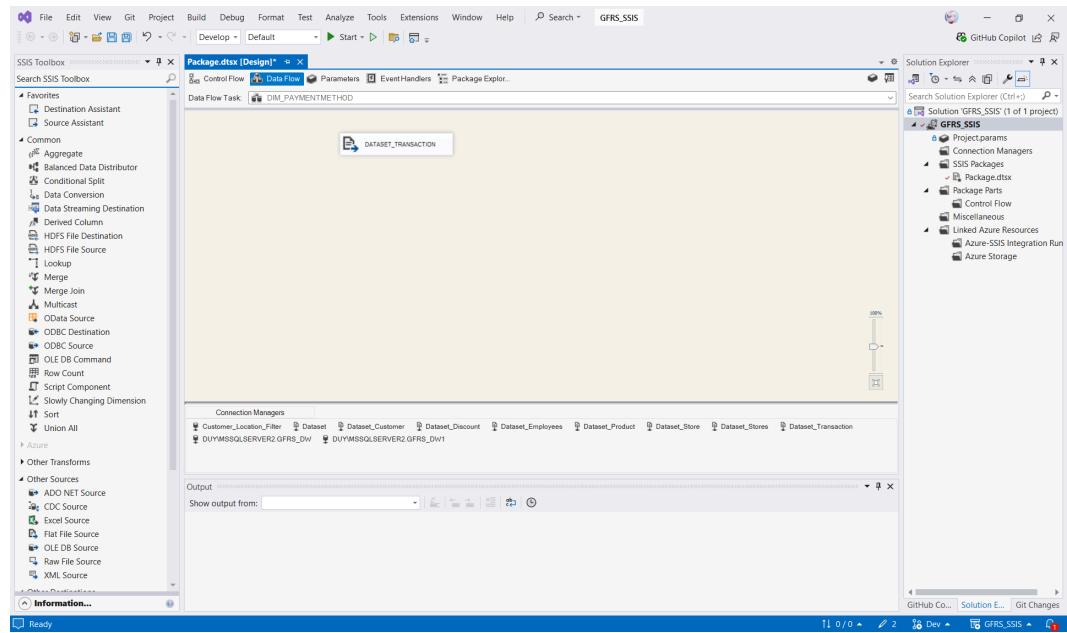
### II.3.12. Dim\_PAYMENTMETHOD

- **Bước 1:** Kéo chức năng Data Flow Task trong vùng SSIS Toolbox vào vùng Control Flow và đổi tên thành **Dim\_PAYMENTMETHOD**



Hình 275. Tạo chức năng Dim\_PAYMENTMETHOD trong Control Flow

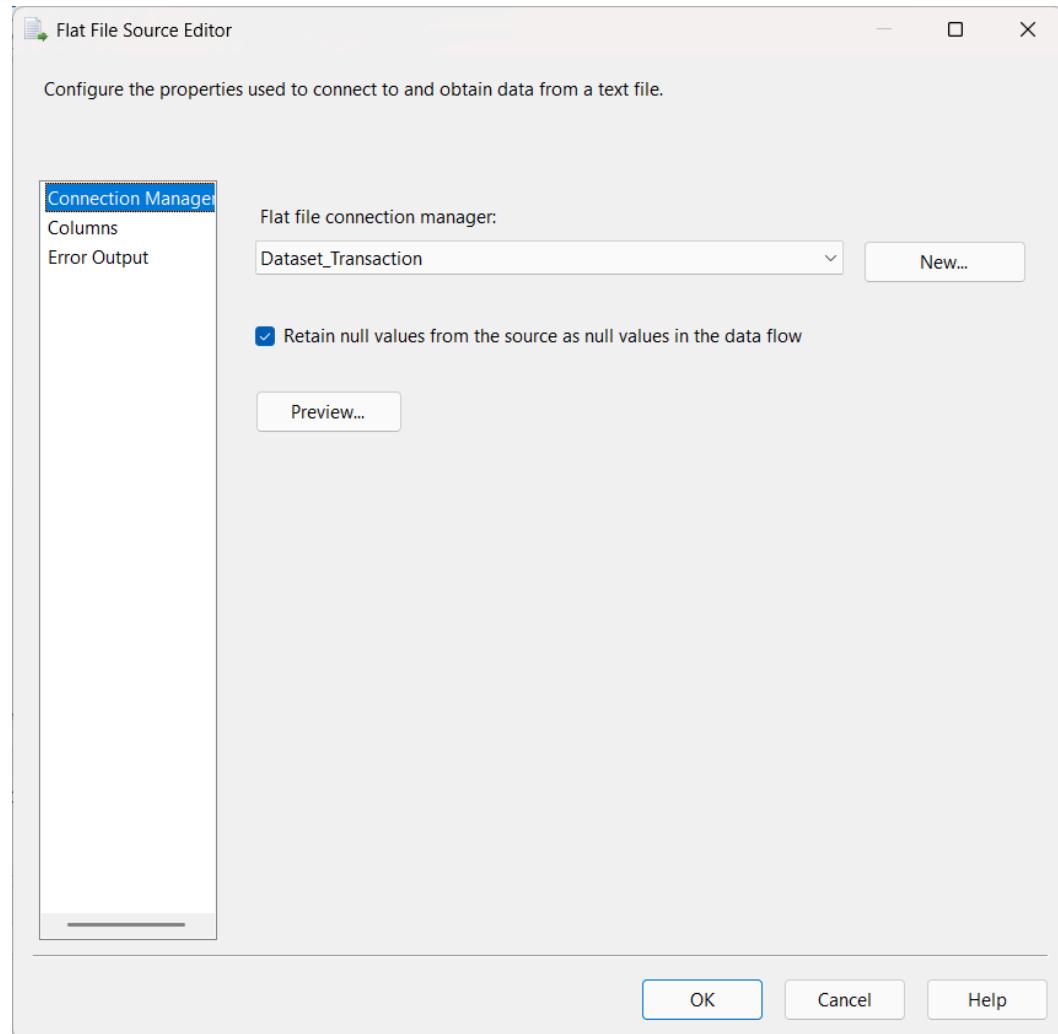
- **Bước 2:** Nhấn đúp chuột vào **Dim\_PAYMENTMETHOD**. Sau đó kéo chức năng **Flat File Source** vào vùng làm việc và đổi tên thành **DATASET\_TRANSACTION**



*Hình 276. Tạo chức năng DATASET\_TRANSACTION trong*

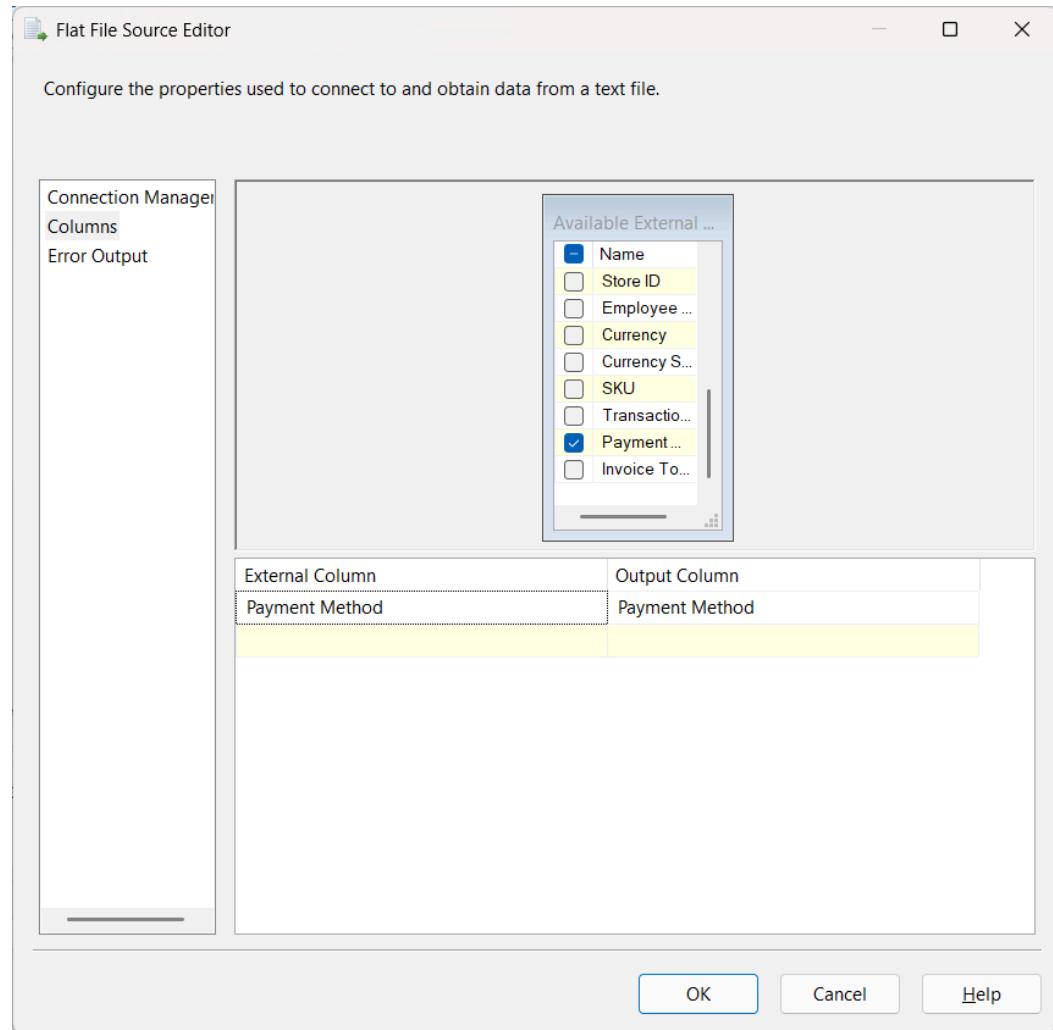
*Dim\_PAYMENTMETHOD*

- **Bước 3:** Nhấn đúp chuột vào **DATASET\_TRANSACTION**. Màn hình **Flat File Source Editor** hiện ra. Ở tab **Connection Manager** ta chọn **Dataset\_Transaction** đã được thiết lập ở phần trước



Hình 277. Kết nối nguồn dữ liệu cho DATASET\_TRANSACTION

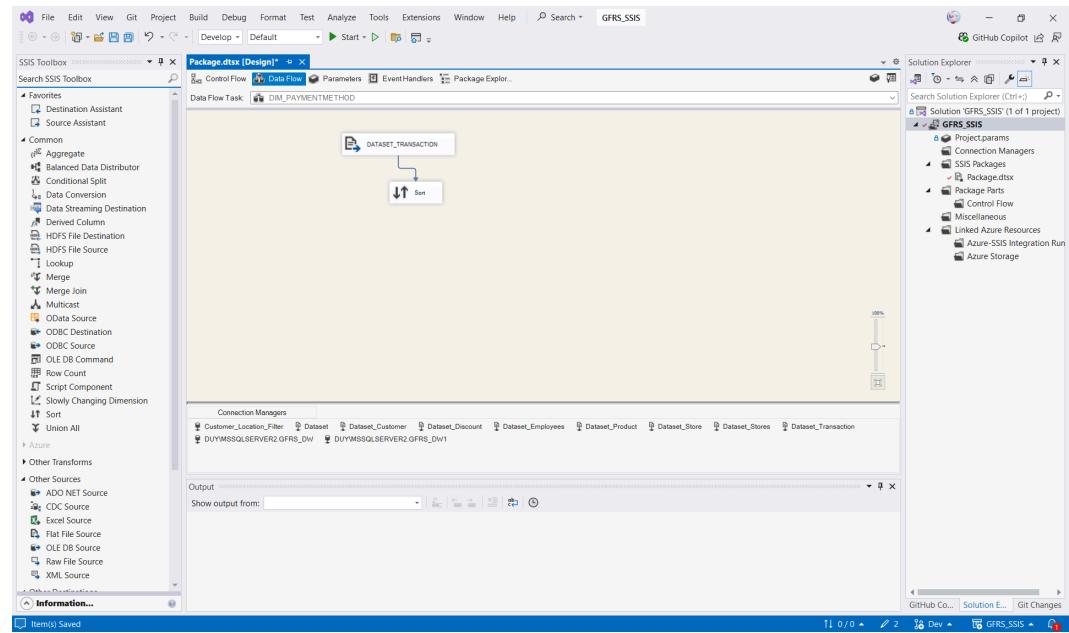
- **Bước 4:** Ở tab **Columns** ta chọn cột **Color** để tiến hành dữ liệu vào **Dim\_PAYMENTMETHOD**. Sau đó, nhấn **OK** và kết thúc cửa sổ **Flat File Source Editor**



Hình 278. Chọn thuộc tính cần thiết cho quá trình tạo

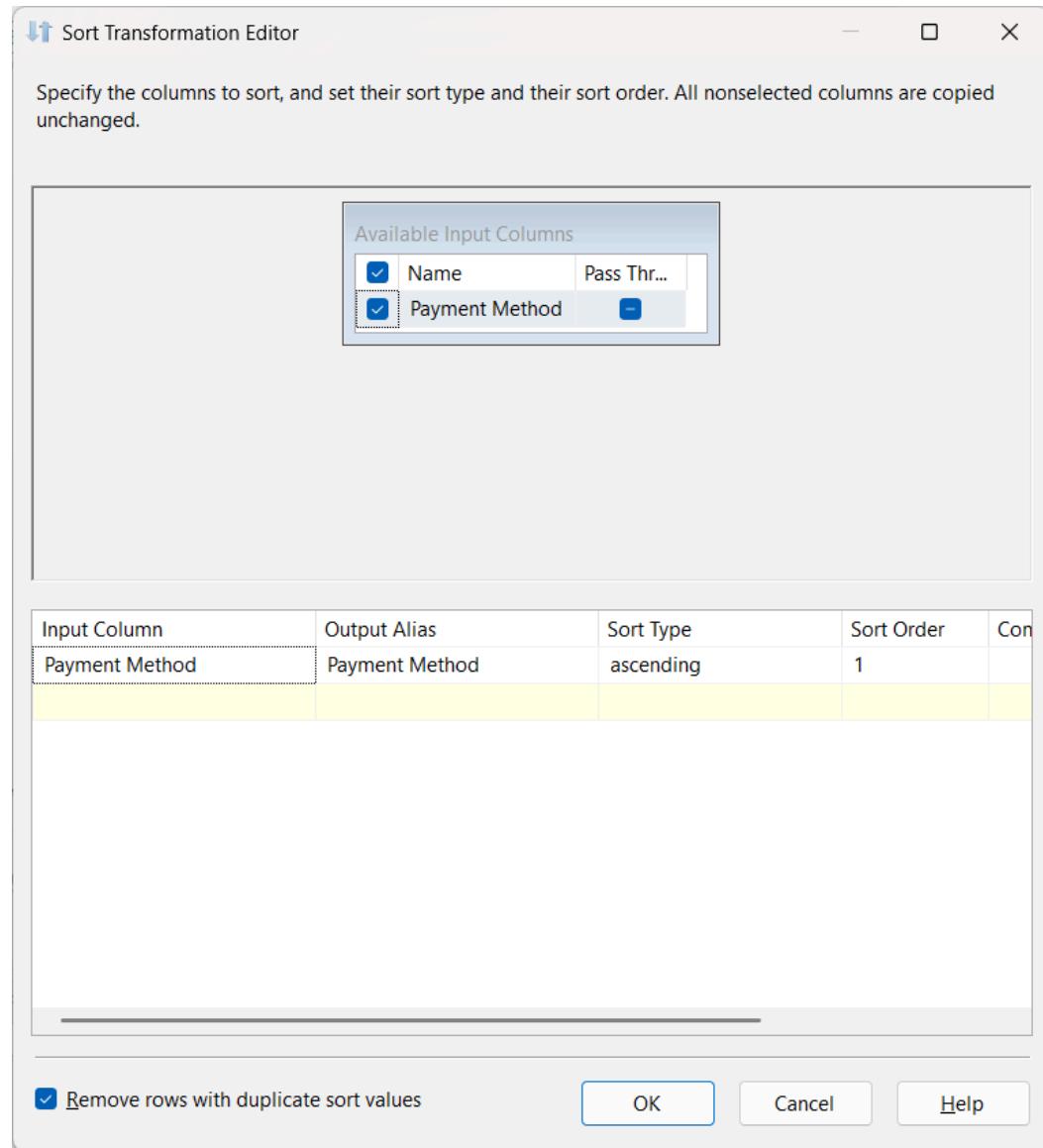
*Dim \_PAYMENTMETHOD*

- **Bước 5:** Kéo chức năng Sort từ vùng SSIS Toolbox vào cùng Data Flow. Kéo luồng dữ liệu từ **DATASET\_TRANSACTION** vào Sort.



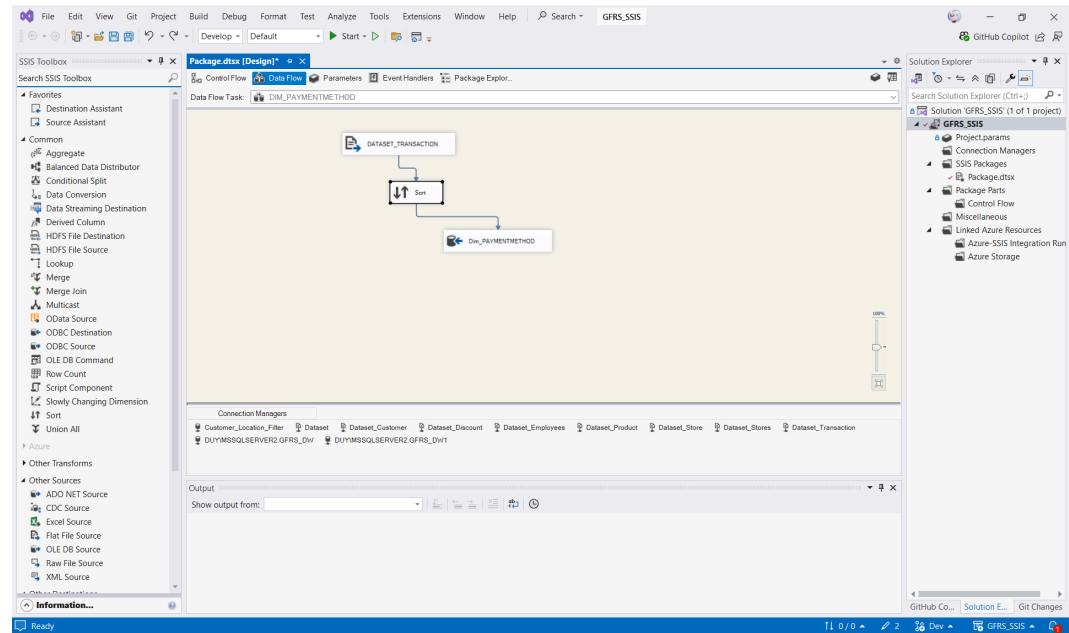
Hình 279. Tạo chức năng Sort trong Dim\_PAYMENTMETHOD

- **Bước 6:** Nhấn đúp chuột vào **Sort**, cửa sổ **Sort Transformation Editor** sẽ hiện lên. Ta chọn cột **Payment Method** trên cửa sổ **Available Input Columns**. Sau đó chọn **Remove rows with duplicate sort values** để lọc những dữ liệu trùng lặp. Sau đó nhấn **OK** và kết thúc cửa sổ **Sort Transformation Editor**.



Hình 280. Cài đặt chức năng Sort trong Dim\_PAYMENTMETHOD

- **Bước 7:** Kéo **ADO NET Destination** từ **SSIS Toolbox** vào **vùng Data Flow** và đổi tên thành **Dim\_DATE**. Kéo luồng dữ liệu từ **Sort** vào **Dim\_DATE**.



Hình 281. Tạo chức năng ADO.NET Destination Dim\_PAYMENTMETHOD

- **Bước 8:** Vào SQL Server 2022, chạy lệnh DDL để tạo Dim\_PAYMENTMETHOD.

```

99    [TransactionTypeID] integer identity(1,1) PRIMARY KEY,
100   [Transaction] varchar(255)
101
102
103
104
105   CREATE TABLE [Dim_PAYMENTMETHOD] (
106     [PaymentMethodID] integer identity(1,1) PRIMARY KEY,
107     [PaymentMethod] varchar(255)
108
109
110
111

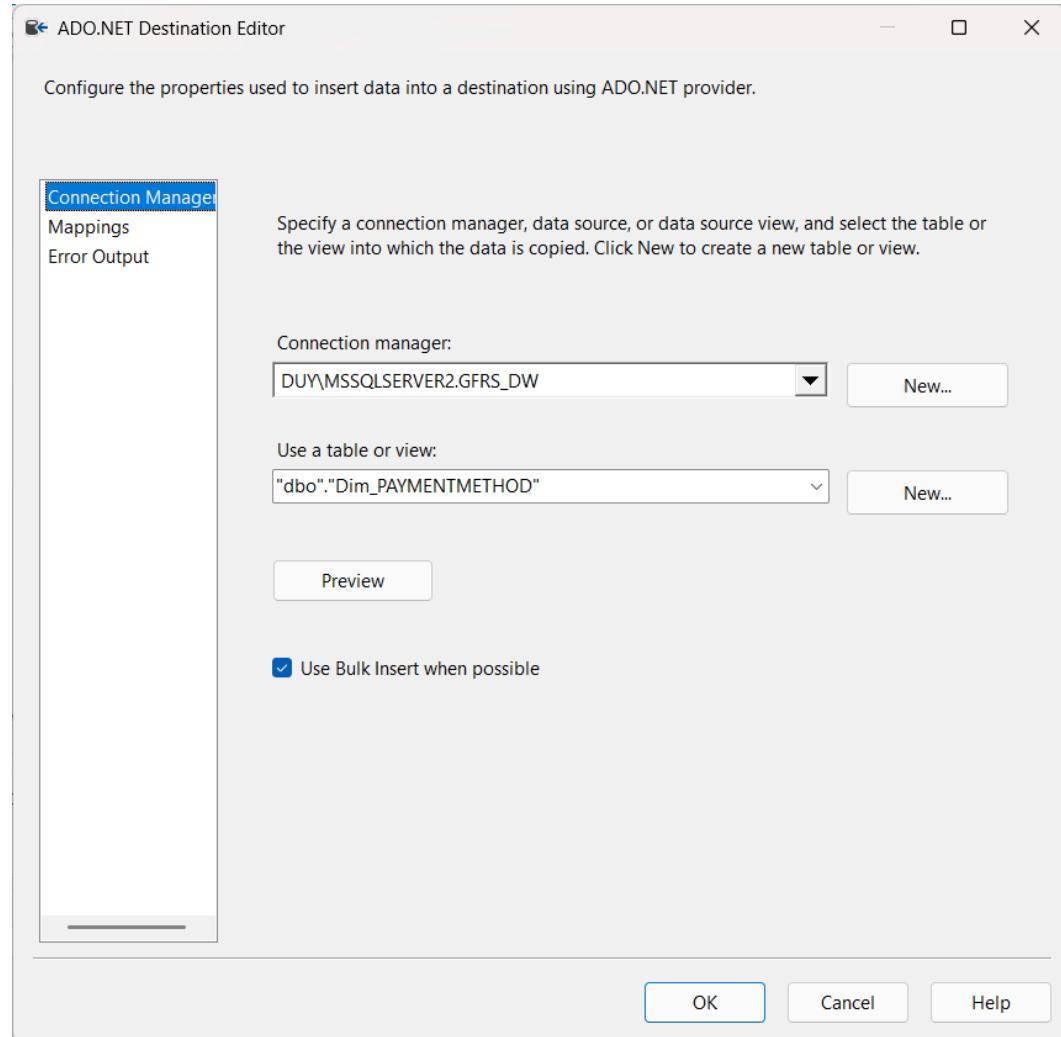
```

TransactionTypeID	Transaction
1	Return
2	Sale

Hình 282. Chạy lệnh DDL tạo Dim\_PAYMENTMETHOD trong kho dữ liệu

- **Bước 9:** Nhấn đúp chuột vào Dim\_PAYMENTMETHOD. Trong tab Connection Manager của màn hình ADO.NET Destination Editor, ở Use

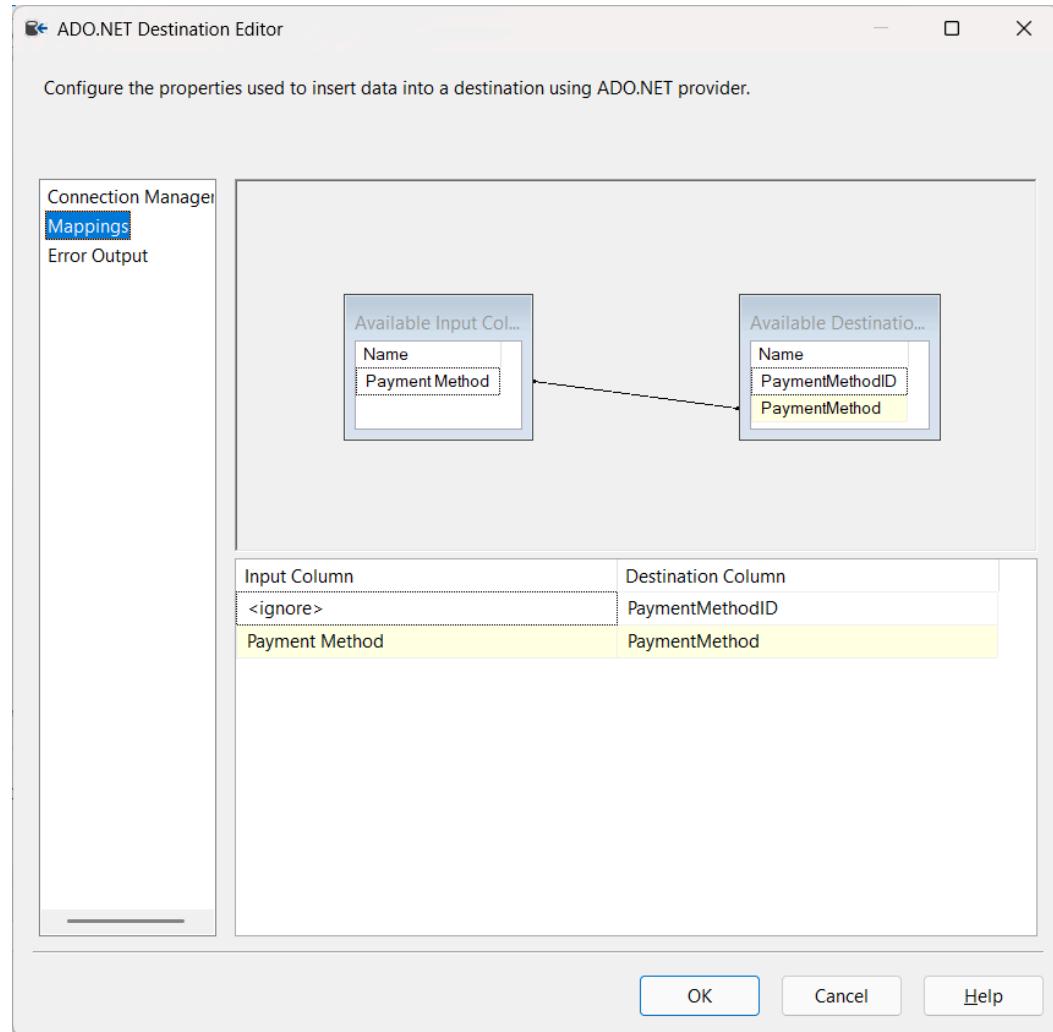
a table or view tiến hành nhấn chọn “dbo.”” Dim\_PAYMENTMETHOD”, cũng chính là bảng vừa được tạo bằng lệnh DDL ở SQL Server 2022



Hình 283. Cài đặt nguồn dữ liệu cho chức năng ADO NET Destination

#### Dim\_PAYMENTMETHOD

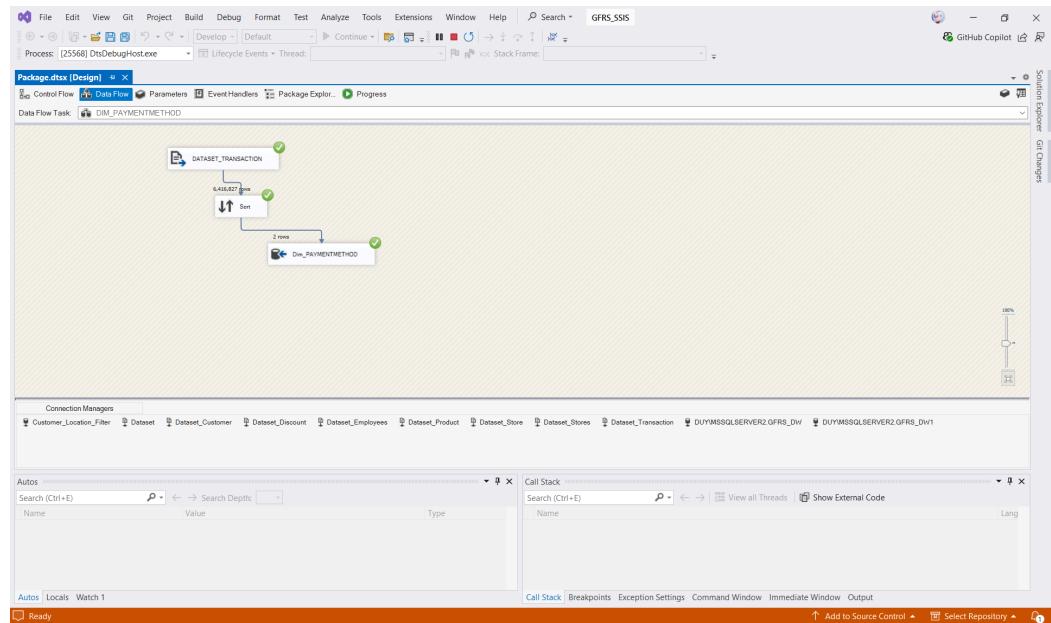
- **Bước 10:** Ở tab **Mapping** của màn hình **ADO.NET Destination Editor** tiến hành chọn các thuộc tính thích hợp ở cột bên trái để ánh xạ với các thuộc tính ở cột bên phải.



Hình 284. Cài đặt chức năng ADO NET Destination

Dim \_PAYMENTMETHOD

- **Bước 11:** Nhấn Start để kiểm tra quá trình tạo và đổ dữ liệu vào Dim \_PAYMENTMETHOD có thành công hay chưa.



Hình 285. Đỗ dữ liệu vào Dim\_PAYMENTMETHOD

Kết quả hiển thị cho thấy quá trình diễn ra thành công và có **2 rows** được đỗ vào **Dim\_PAYMENTMETHOD**.

- **Bước 12:** Vào SQL Server 2022 kiểm tra và thấy dữ liệu đã được đỗ vào Dim\_PAYMENTMETHOD thành công.

The screenshot shows the SQL Server Management Studio (SSMS). The Object Explorer on the left shows the database structure under 'DUY\MYSQLSERVER2'. The central pane displays a script for creating the Dim\_PAYMENTMETHOD table and inserting data. The script is as follows:

```

99
100
101
102
103
104
105
106
107
108
109
110
111
    [TransactionTypeID] integer identity(1,1) PRIMARY KEY,
    [Transaction] varchar(255)
);

--select * from Dim_TRANSACTIONTYPE;
drop table if exists Dim_PAYMENTMETHOD;
CREATE TABLE [Dim_PAYMENTMETHOD] (
    [PaymentMethodID] integer identity(1,1) PRIMARY KEY,
    [PaymentMethod] varchar(255)
);
select * from Dim_PAYMENTMETHOD;

drop table if exists Fact_TRANSACTIONS;

```

The 'Results' tab shows the output of the 'select \* from Dim\_PAYMENTMETHOD;' command, displaying two rows of data:

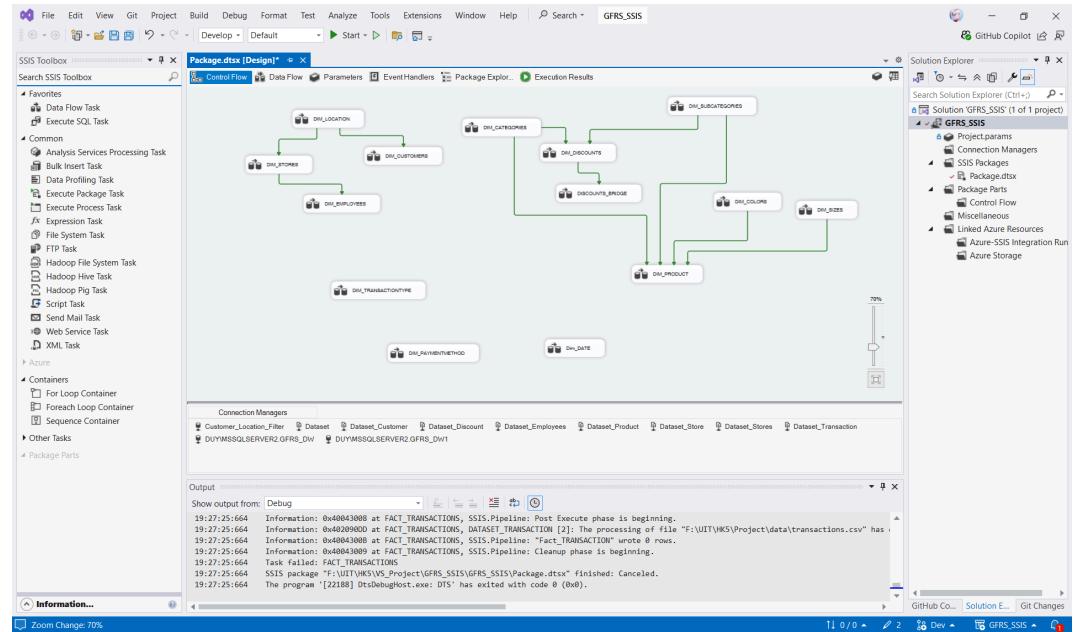
PaymentMethodID	PaymentMethod
1	Cash
2	Credit Card

The status bar at the bottom indicates 'Query executed successfully.' and shows connection details: 'DUY\MYSQLSERVER2 (16.0 RTM) | DUY\maing (72) | GFRS\_DW | 00:00:00 | 2 rows'.

Hình 286. Dữ liệu được đỗ thành công vào Dim\_PAYMENTMETHOD

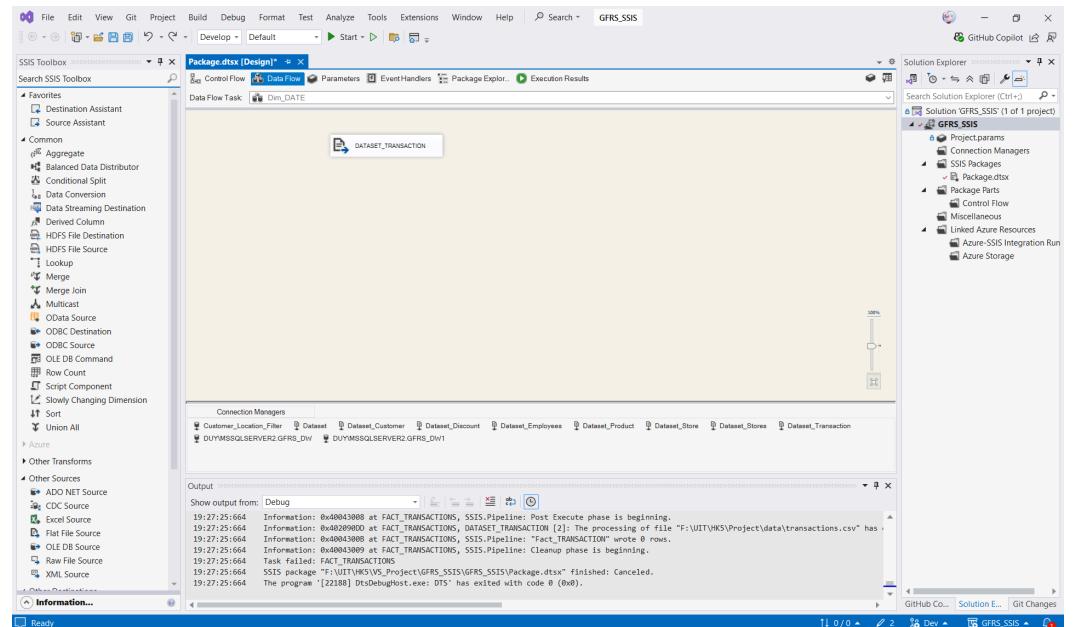
### II.3.13. Dim\_DATE

- Bước 1: Kéo chức năng Data Flow Task trong vùng SSIS Toolbox vào vùng Control Flow và đổi tên thành Dim\_DATE



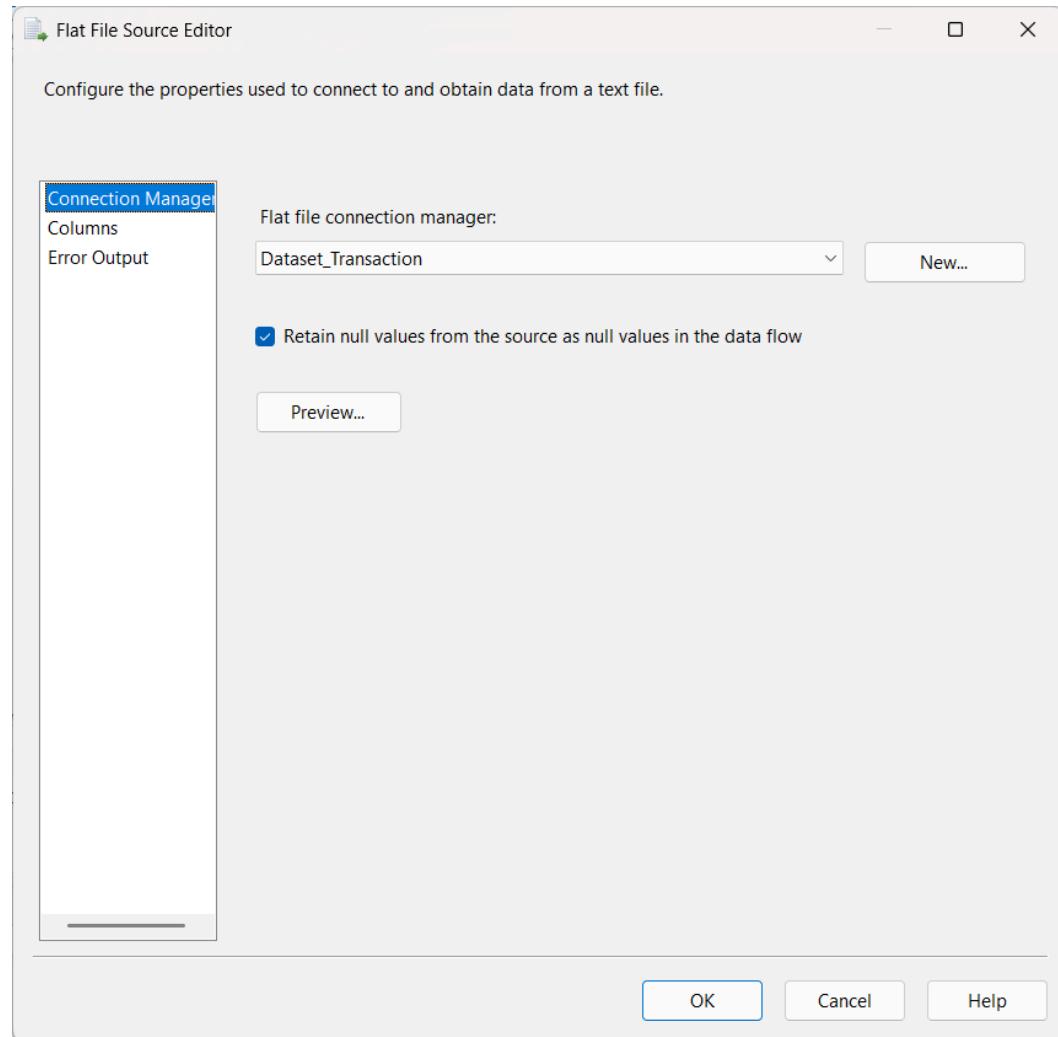
Hình 287. Tạo Dim\_DATE trong vùng Control Flow

- Bước 2: Nhấn đúp chuột vào Dim\_DATE. Sau đó kéo chức năng Flat File Source vào vùng làm việc và đổi tên thành DATASET\_TRANSACTION



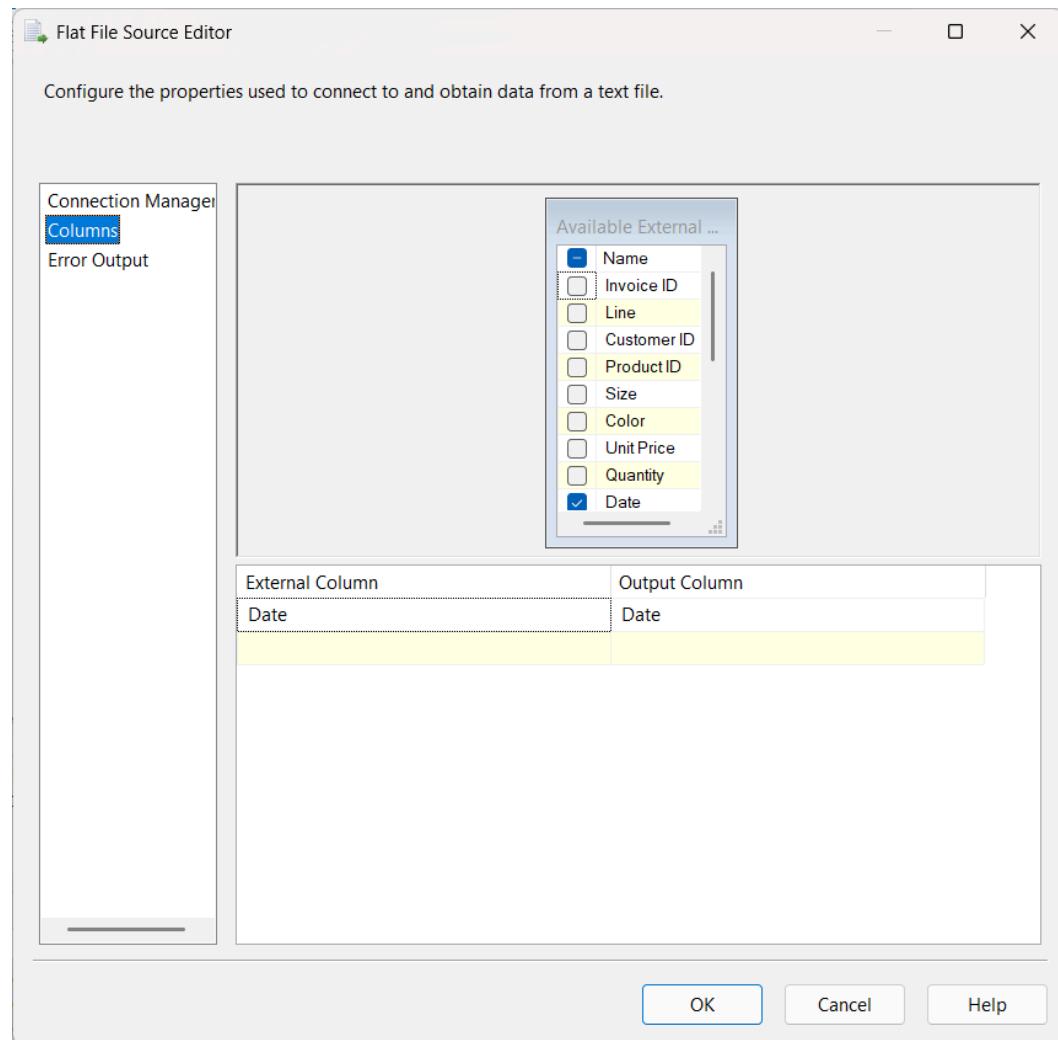
Hình 288. Tạo chức năng DATASET\_TRANSACTION trong Dim\_DATE

- **Bước 3:** Nhấn đúp chuột vào **DATASET\_TRANSACTION**. Màn hình **Flat File Source Editor** hiện ra. Ở tab **Connection Manager** ta chọn **Dataset\_Transaction** đã được thiết lập ở phần trước



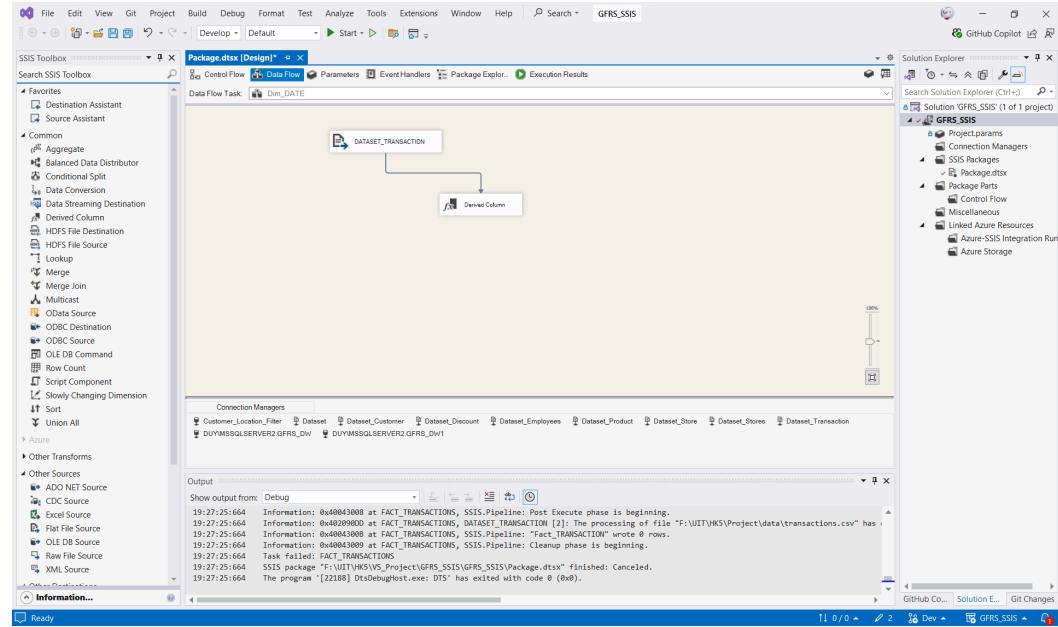
Hình 289. Kết nối nguồn dữ liệu cho DATASET\_TRANSACTION

- **Bước 4:** Ở tab **Columns** ta chọn cột **Date** để tiến hành nạp dữ liệu vào **Dim\_DATE**. Sau đó, nhấn **OK** và kết thúc cửa sổ **Flat File Source Editor**



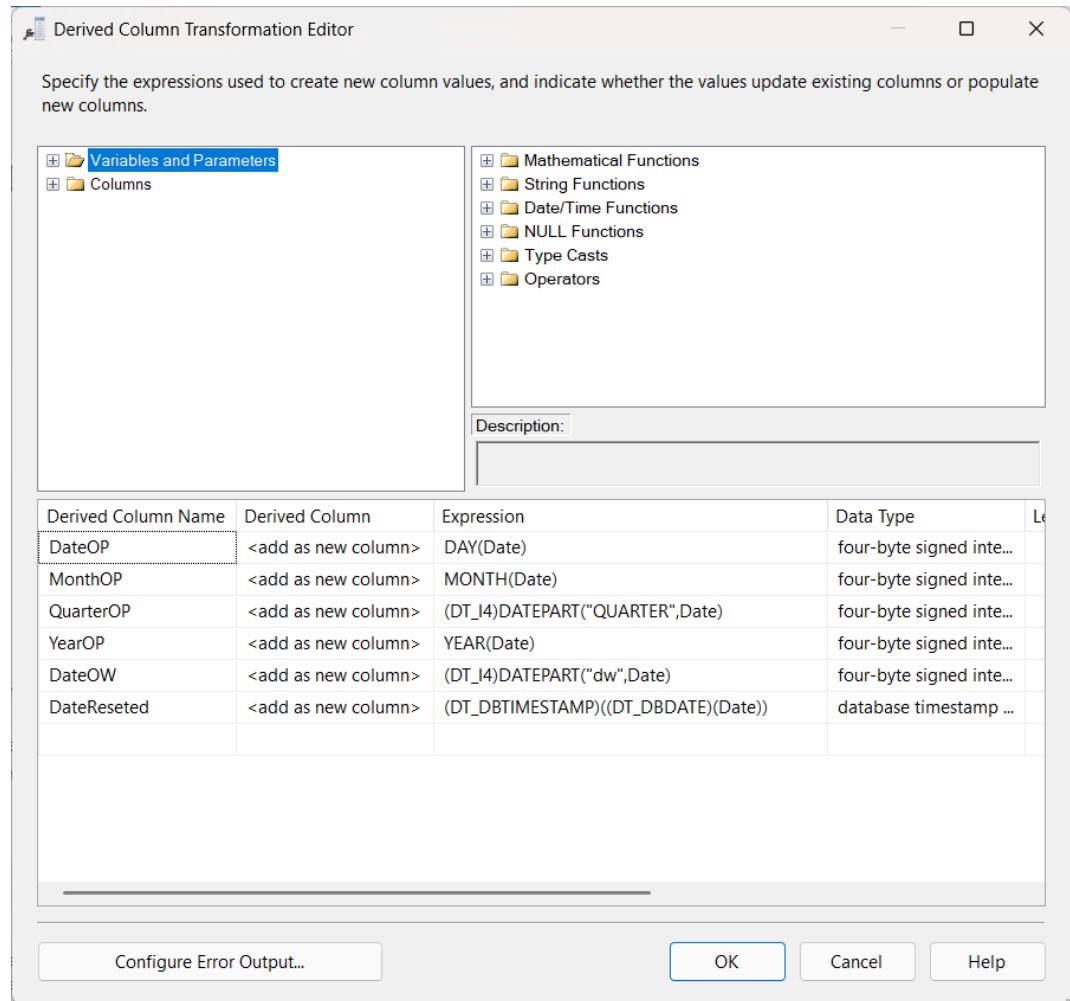
Hình 290. Chọn các thuộc tính cần thiết cho quá trình tạo Dim\_DATE

- **Bước 5:** Kéo chức năng **Derived Column** từ vùng **SSIS Toolbox** vào cùng **Data Flow**. Kéo luồng dữ liệu từ **DATASET\_TRANSACTION** vào **Derived Column**.



Hình 291. Tạo chức năng Derived Column trong Dim\_DATE

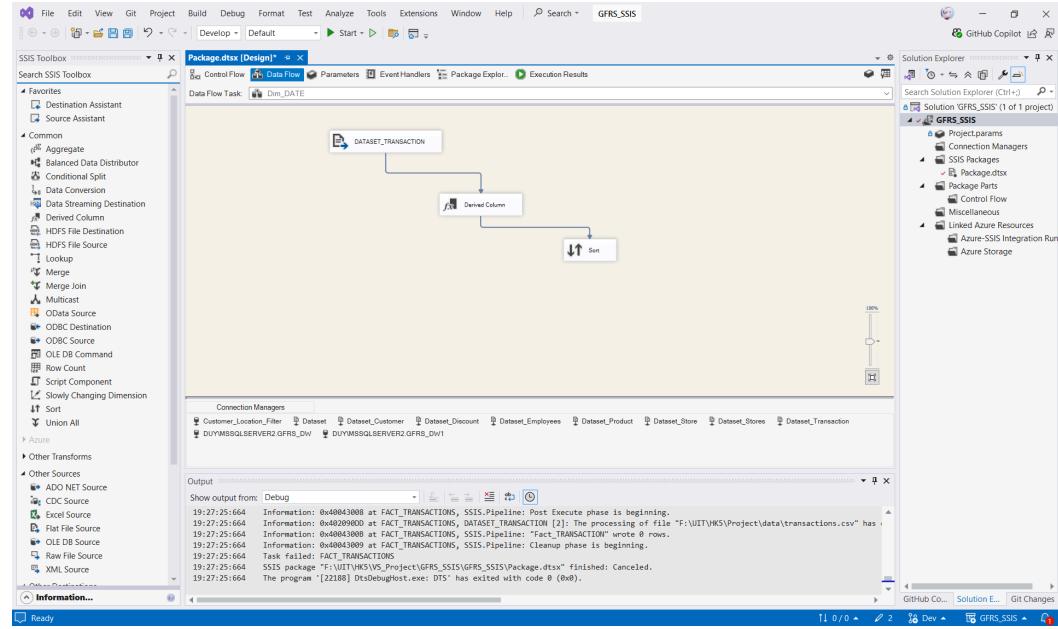
- **Bước 6:** Nhấn đúp chuột vào **Derived Column**, một cửa sổ **Derived Column Transformation Editor** sẽ hiện lên. Ta tiến hành tạo các cột mới với các **Expression** như sau:
  - o **DateOP:** DAY(Date)
  - o **MonthOP:** MONTH(Date)
  - o **YearOP:** YEAR(Date)
  - o **QuarterOP:** (DT\_I4)DATEPART("QUARTER",Date)
  - o **DateReseted:** (DT\_DBTIMESTAMP)((DT\_DBDATE)(Date))
  - o **DateOW:** (DT\_I4)DATEPART("dw",Date)



Hình 292. Cài đặt chức năng Derived Column trong Dim\_DATE

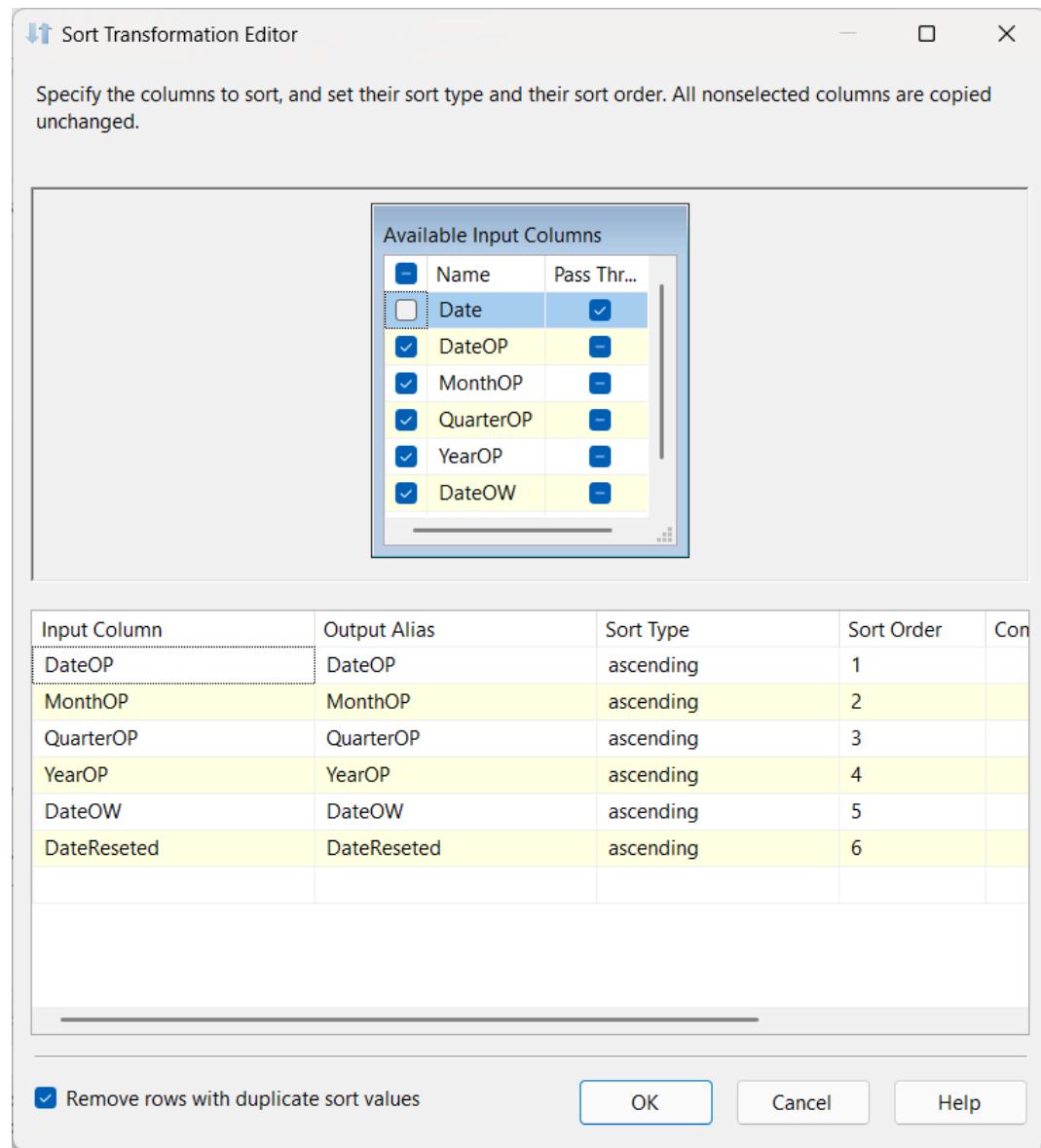
Sau đó nhấn **OK** và kết thúc cửa sổ **Derived Column Transformation Editor**

- **Bước 7:** Kéo chức năng **Sort** từ vùng **SSIS Toolbox** vào cùng **Data Flow**. Kéo luồng dữ liệu từ **Derived Column** vào **Sort**.



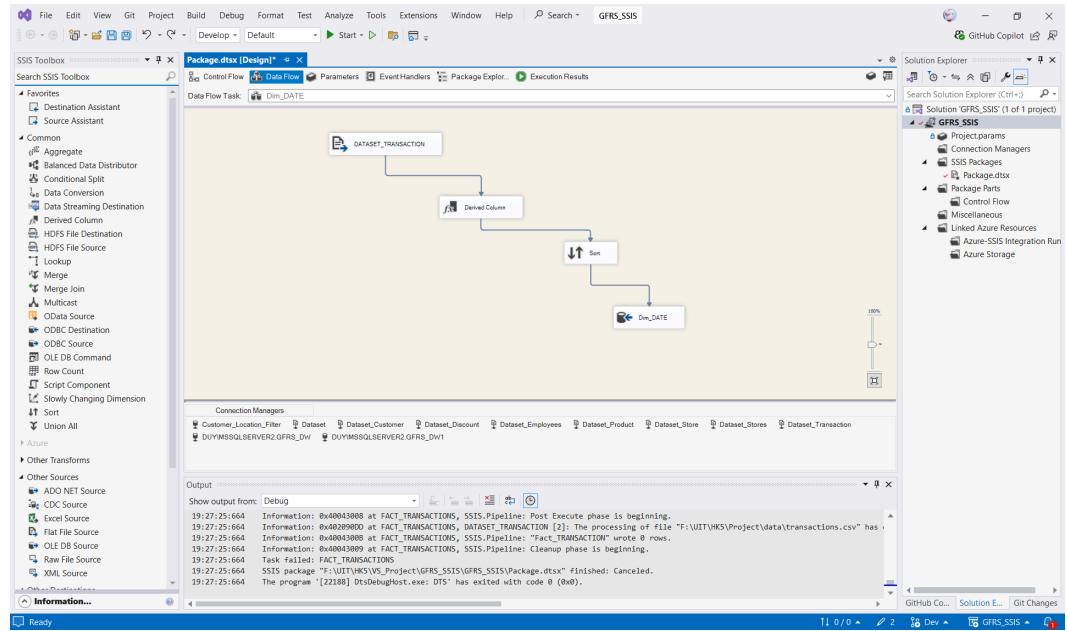
Hình 293. Tạo chức năng Sort trong Dim\_DATE

- **Bước 8:** Nhấn đúp chuột vào **Sort**, cửa sổ **Sort Transformation Editor** sẽ hiện lên. Ta chọn các cột trên cửa sổ **Available Input Columns**. Sau đó chọn **Remove rows with duplicate sort values** để lọc những dữ liệu trùng lặp. Sau đó nhấn **OK** và kết thúc cửa sổ **Sort Transformation Editor**.



Hình 294. Cài đặt chức năng Sort trong Dim\_DATE

- **Bước 9:** Kéo **ADO NET Destination** từ **SSIS Toolbox** vào **vùng Data Flow** và đổi tên thành **Dim\_DATE**. Kéo luồng dữ liệu từ **Sort** vào **Dim\_DATE**.



Hình 295. Tạo chức năng ADO.NET Destination Dim\_DATE

- **Bước 10:** Vào SQL Server 2022, chạy lệnh DDL để tạo Dim\_DATE.

```

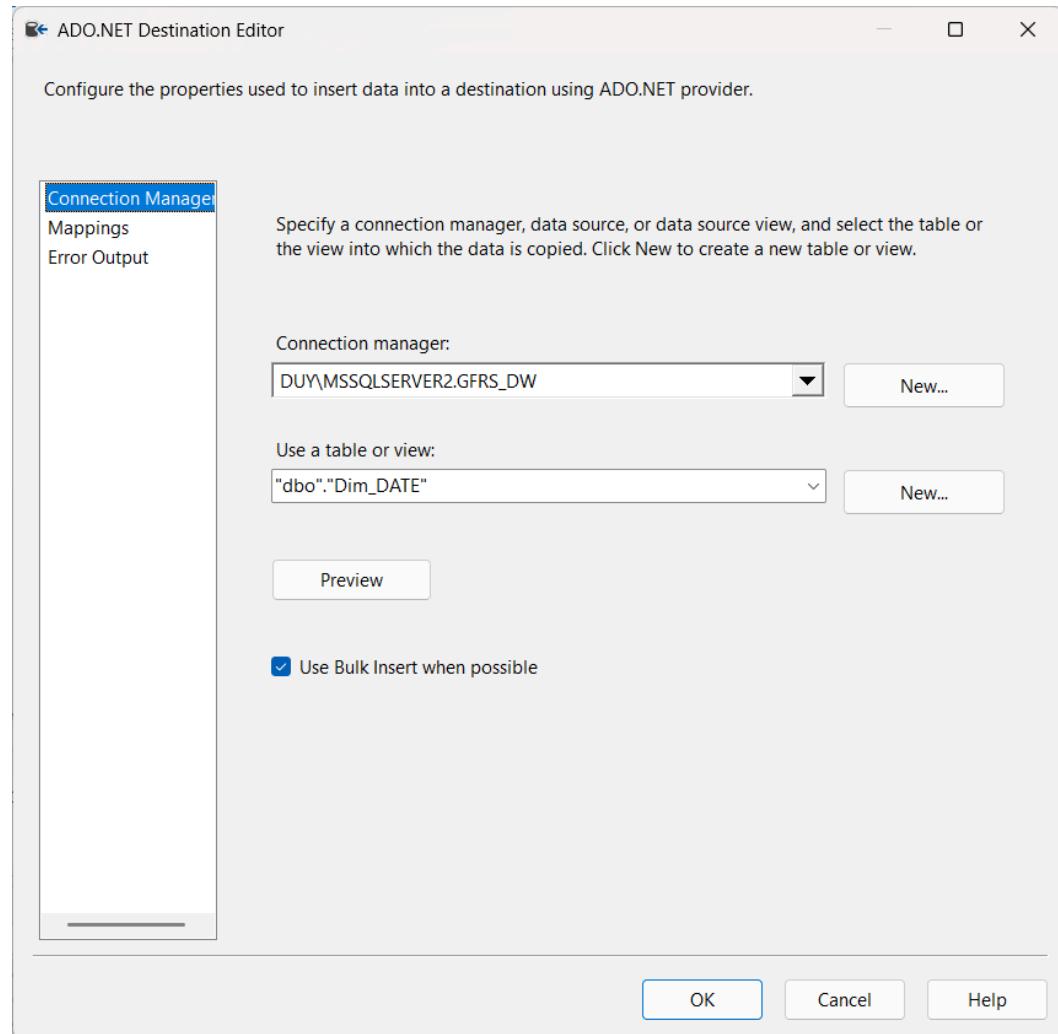
-- Date dimension
CREATE TABLE Dim_DATE (
    [DateID] INT IDENTITY(1,1) PRIMARY KEY,
    [FullDate] DATETIME,
    [DateOfPurchase] INT,
    [MonthOfPurchase] INT,
    [YearOfPurchase] INT,
    [QuarterOfPurchase] INT,
    [DayOfWeek] INT
);

TRUNCATE TABLE Dim_DATE;
-- Fact table Transactions
CREATE TABLE Fact_TRANSACTIONS (
    [InvoiceID] BIGINT IDENTITY(1,1) PRIMARY KEY,
    [InvoiceHash] VARCHAR(50),
    [Line] INT,
);

```

Hình 296. Chạy lệnh DDL tạo Dim\_DATE trong kho dữ liệu

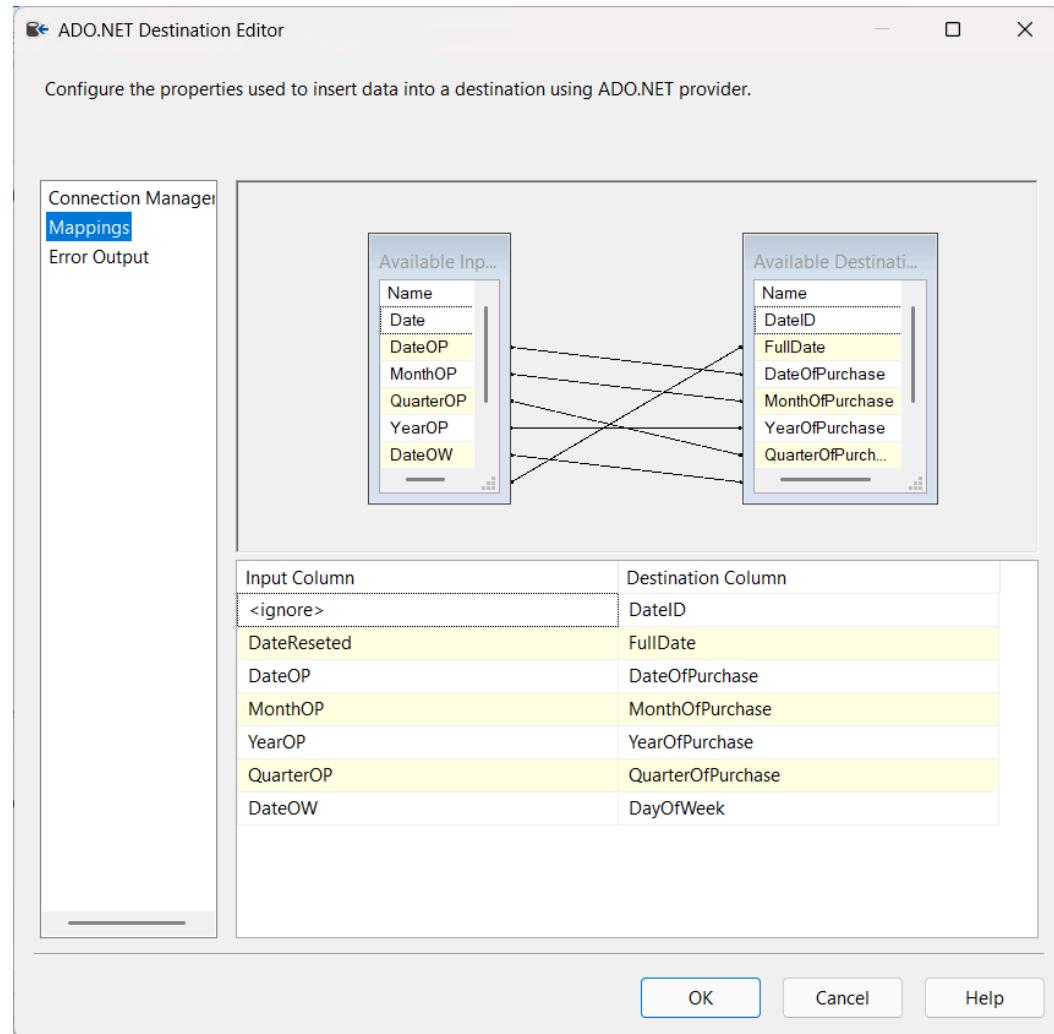
- **Bước 11:** Nhấn đúp chuột vào Dim\_DATE. Trong tab Connection Manager của màn hình ADO.NET Destination Editor, ở Use a table or view tiến hành nhấp chọn “dbo.” Dim\_DATE”, cũng chính là bảng vừa được tạo bằng lệnh DDL ở SQL Server 2022



Hình 297. Kết nối đích đến cho chức năng ADO.NET Destination

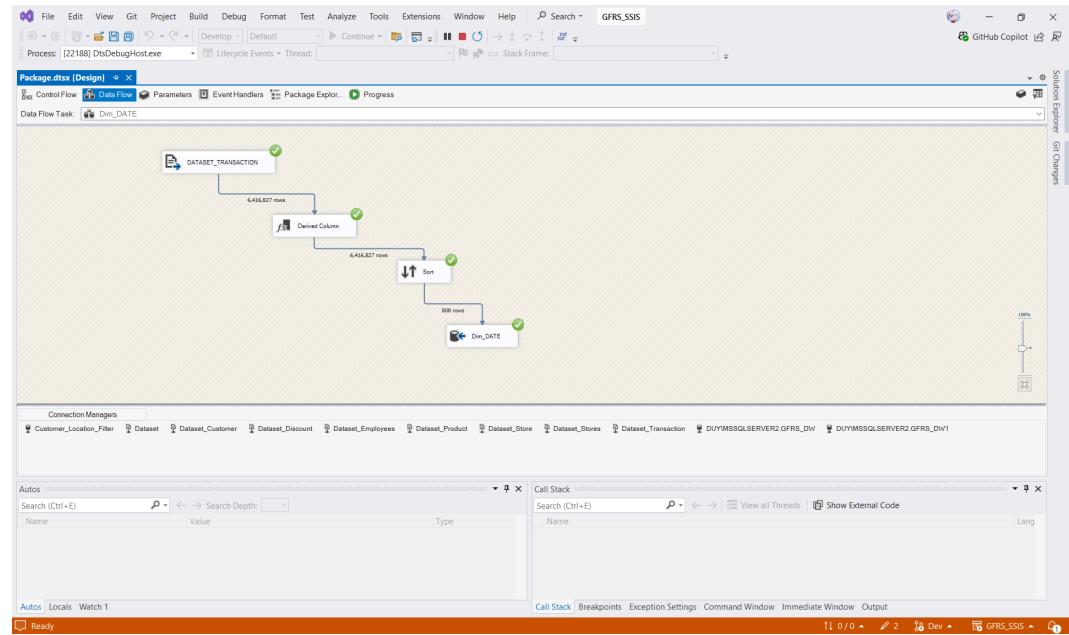
*Dim\_DATE*

- **Bước 12:** Ở tab **Mapping** của màn hình **ADO.NET Destination Editor** tiến hành chọn các thuộc tính thích hợp ở cột bên trái để ánh xạ với các thuộc tính ở cột bên phải.



Hình 298. Cài đặt chức năng ADO NET Destination Dim\_DATE

- **Bước 13:** Nhấn Start để kiểm tra quá trình tạo và đồ dữ liệu vào Dim\_DATE có thành công hay chưa.



Hình 299. Đỗ dữ liệu vào Dim\_DATE

Kết quả hiển thị cho thấy quá trình diễn ra thành công và có **808 rows** được đỗ vào **Dim\_DATE**.

- **Bước 14:** Vào SQL Server 2022 kiểm tra và thấy dữ liệu đã được đỗ vào Dim\_DATE thành công.

The screenshot shows the SSMS interface with a query window running against the 'GFRS\_DW' database. The query is:

```
SELECT * FROM Dim_DATE
```

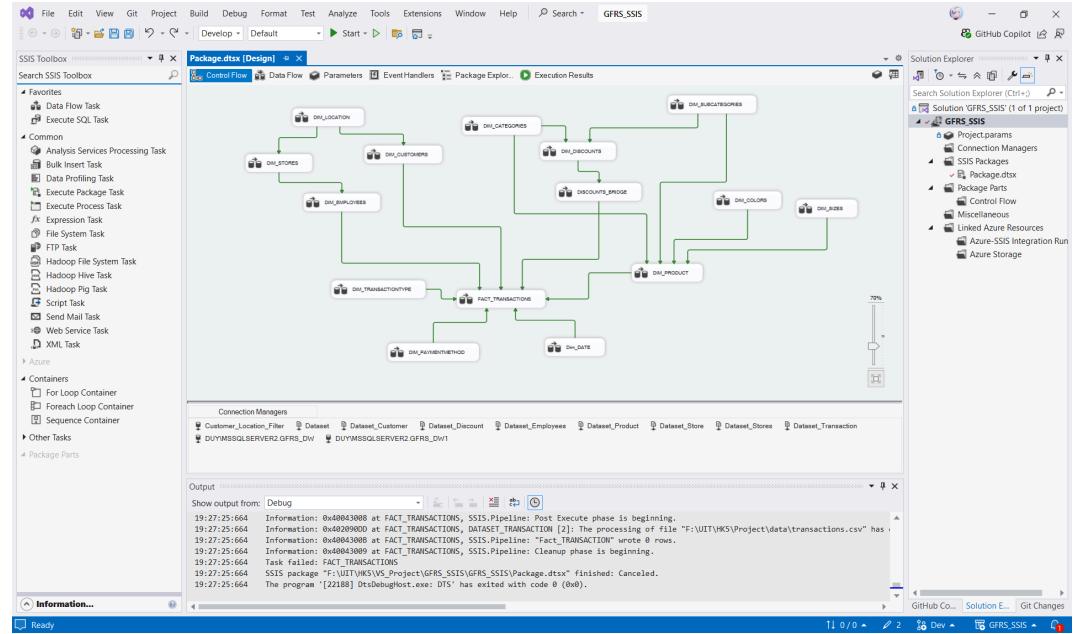
The results grid displays 1,616 rows of data, showing columns such as DateID, FullDate, DateOffPurchase, MonthOffPurchase, YearOffPurchase, QuarterOffPurchase, and DayOfWeek.

DateID	FullDate	DateOffPurchase	MonthOffPurchase	YearOffPurchase	QuarterOffPurchase	DayOfWeek
13..1381	2023-04-22 00:00:00	22	4	2023	2	7
13..1382	2023-04-22 00:00:00	23	4	2024	2	2
13..1383	2023-04-22 00:00:00	23	5	2023	2	2
13..1384	2024-05-22 00:00:00	22	5	2024	2	4
13..1385	2023-06-22 00:00:00	22	6	2023	2	5
13..1386	2024-06-22 00:00:00	22	6	2024	2	7
13..1387	2023-07-22 00:00:00	22	7	2023	3	2
13..1388	2023-07-22 00:00:00	22	7	2024	3	2
13..1389	2023-08-22 00:00:00	22	8	2023	3	3
13..1390	2023-08-22 00:00:00	22	8	2024	3	5
13..1391	2023-08-22 00:00:00	22	9	2023	3	6
13..1392	2023-08-22 00:00:00	22	9	2024	3	1
13..1393	2023-09-22 00:00:00	22	10	2023	4	1
13..1394	2024-09-22 00:00:00	22	10	2024	4	3
13..1395	2023-10-22 00:00:00	22	11	2023	4	4
13..1396	2024-10-22 00:00:00	22	11	2024	4	4
13..1397	2023-11-22 00:00:00	22	12	2023	4	6
13..1398	2024-11-22 00:00:00	22	12	2024	4	1
13..1399	2023-12-23 00:00:00	23	1	2023	1	2
14..1400	2024-12-23 00:00:00	23	1	2024	1	3

Hình 300. Dữ liệu được đỗ thành công vào Dim\_DATE

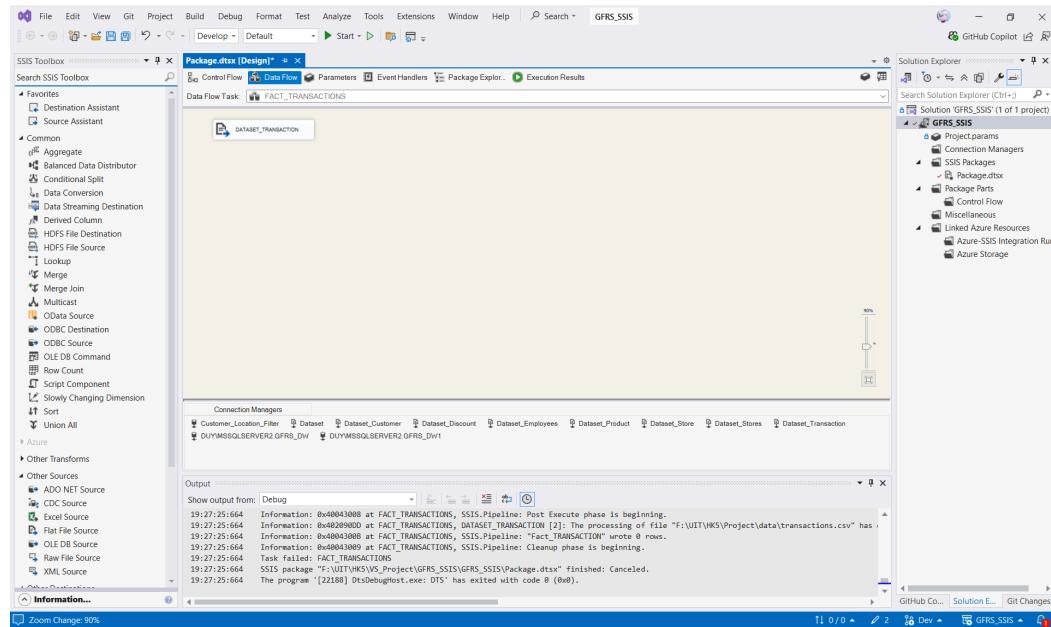
### II.3.14. Fact\_TRANSACTIONS

- **Bước 1:** Kéo chức năng Data Flow Task trong vùng SSIS Toolbox vào vùng Control Flow và đổi tên thành **Fact\_TRANSACTION**



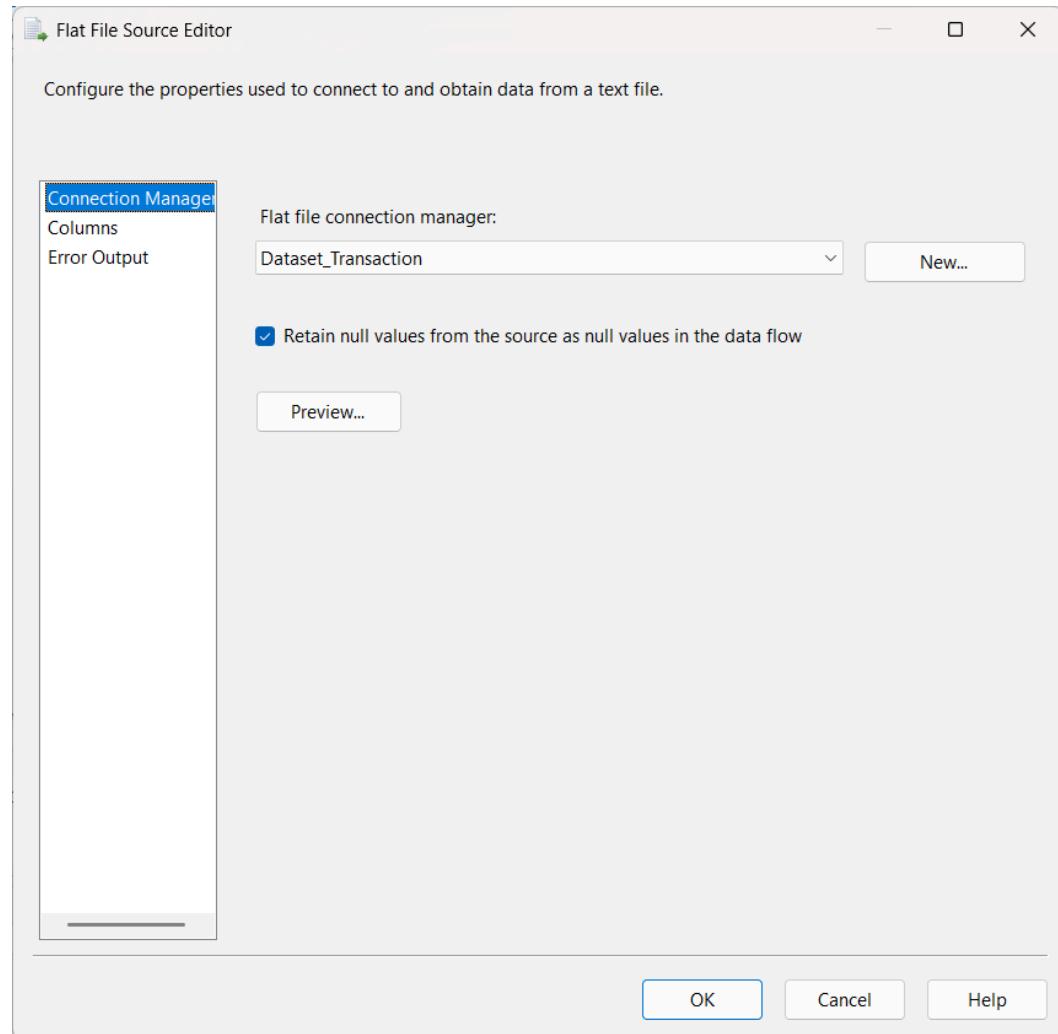
Hình 301. Tạo Fact\_TRANSACTION trong Control Flow

- **Bước 2:** Nhấn đúp chuột vào **Fact\_TRANSACTION**. Sau đó kéo chức năng **Flat File Source** vào vùng làm việc và đổi tên thành **DATASET\_TRANSACTION**



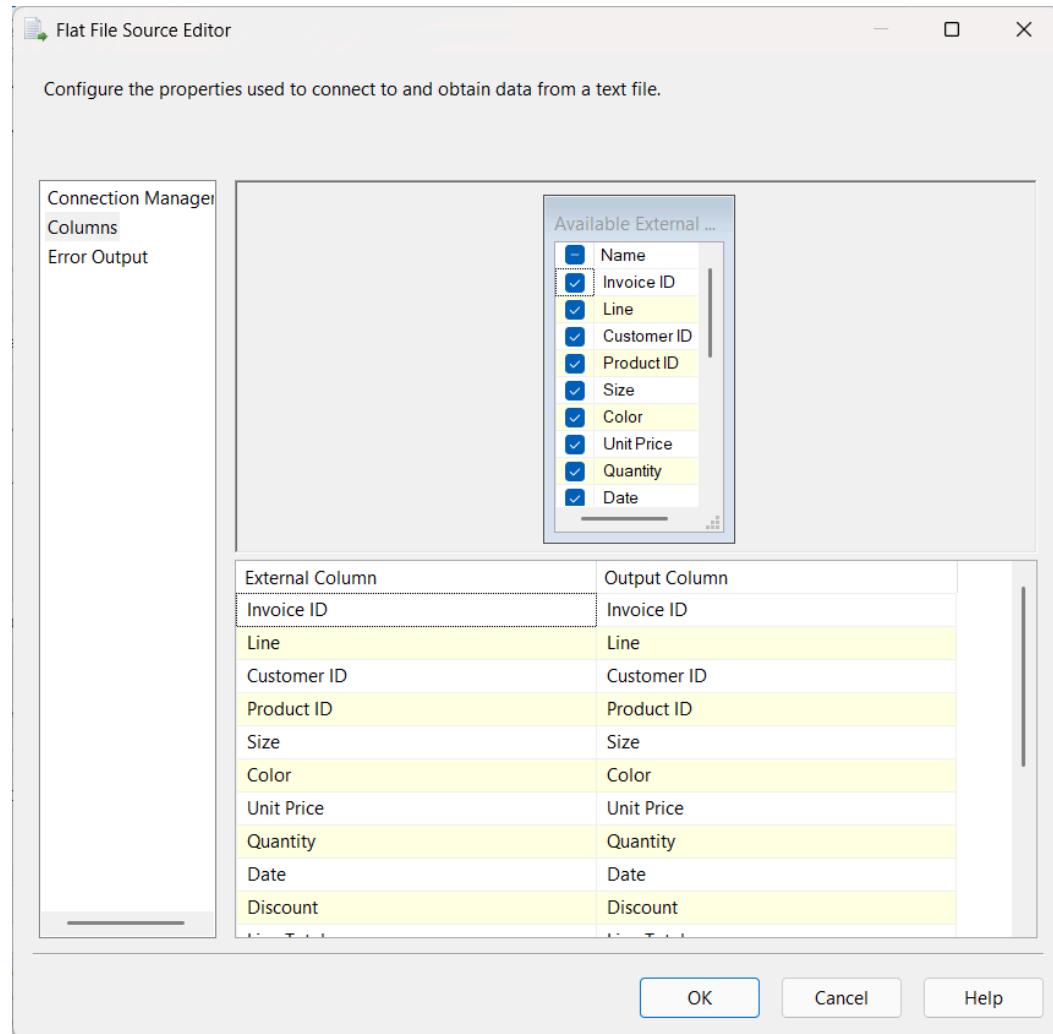
*Hình 302. Tạo DATASET\_TRANSACTION trong Fact\_TRANSACTIONS*

- **Bước 3:** Nhấn đúp chuột vào **DATASET\_TRANSACTION**. Màn hình **Flat File Source Editor** hiện ra. Ở tab **Connection Manager** ta chọn **Dataset\_Transaction** đã được thiết lập ở phần trước



Hình 303. Kết nối nguồn dữ liệu cho DATASET\_TRANSACTION

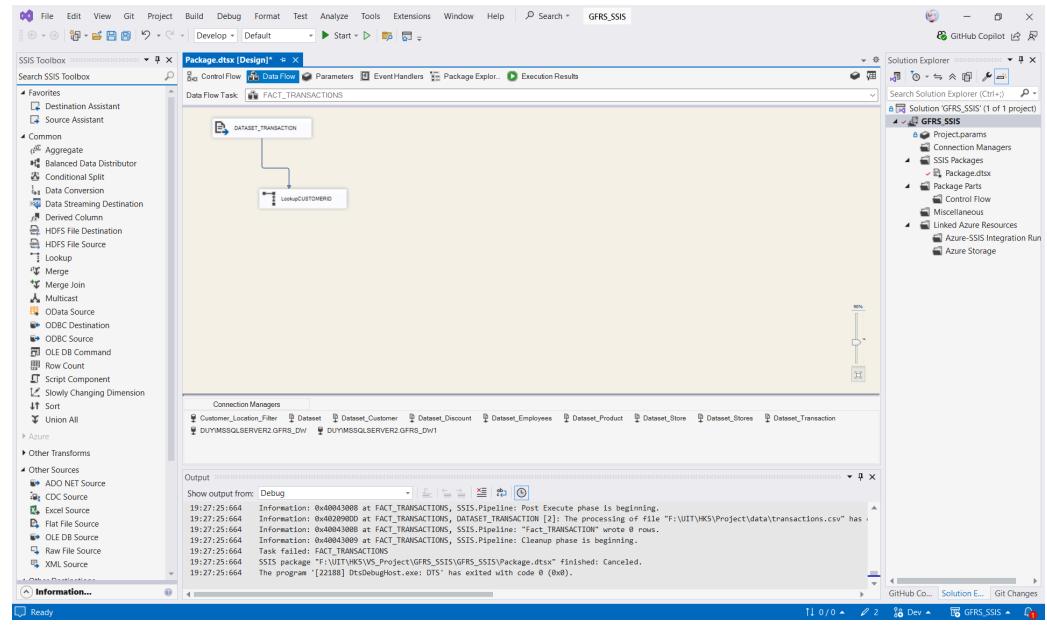
- **Bước 4:** Ở tab **Columns** ta chọn cột **Date** để tiến hành nạp dữ liệu vào **Fact\_TRANSACTIONS**. Sau đó, nhấn **OK** và kết thúc cửa sổ **Flat File Source Editor**



Hình 304. Chọn thuộc tính cần thiết cho quá trình tạo

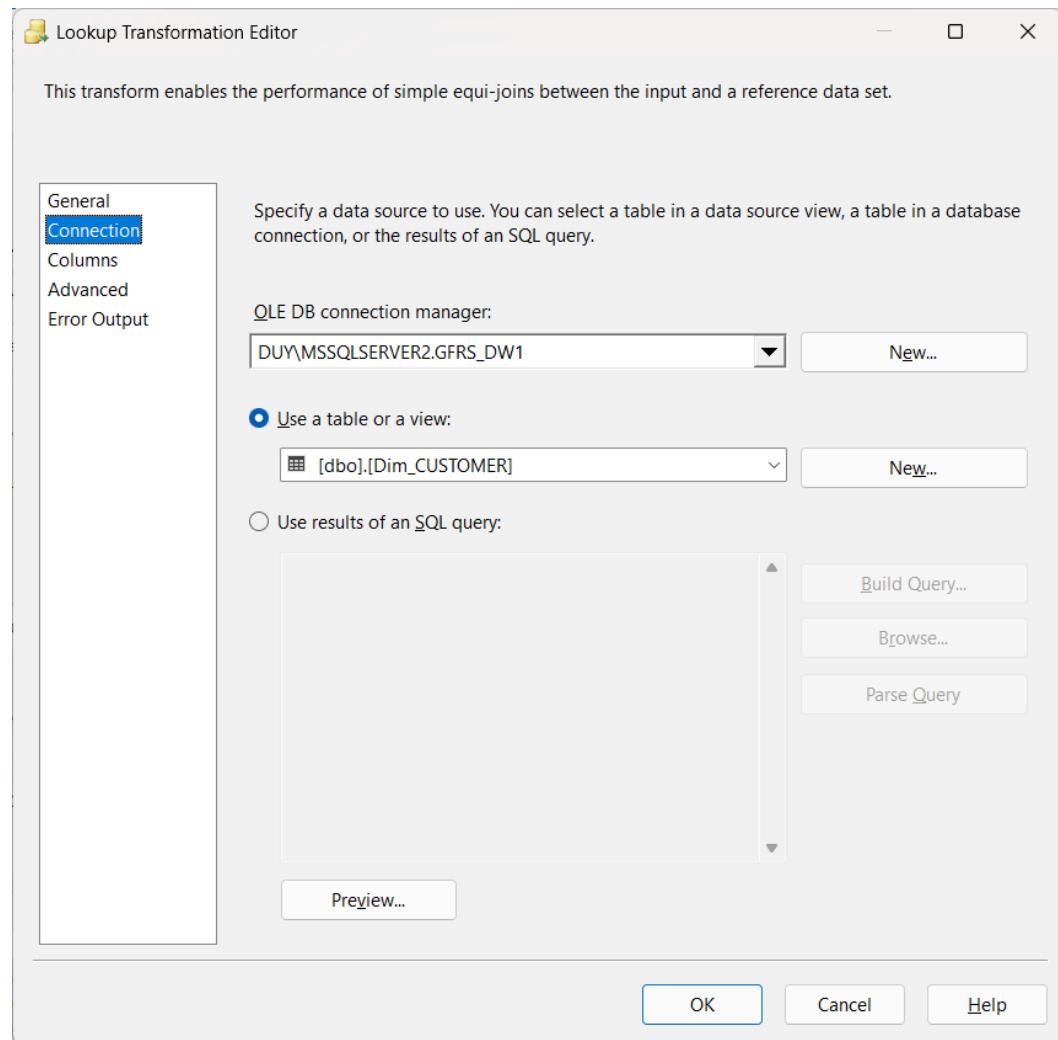
*Fact\_TRANSACTION*

- **Bước 5:** Kéo chức năng **Lookup** vào vùng **Data Flow** và đổi tên thành **LookupCUSTOMERID**. Kéo mũi tên luồng dữ liệu từ **DATESET\_TRANSACTION** vào **LookupCUSTOMERID**.



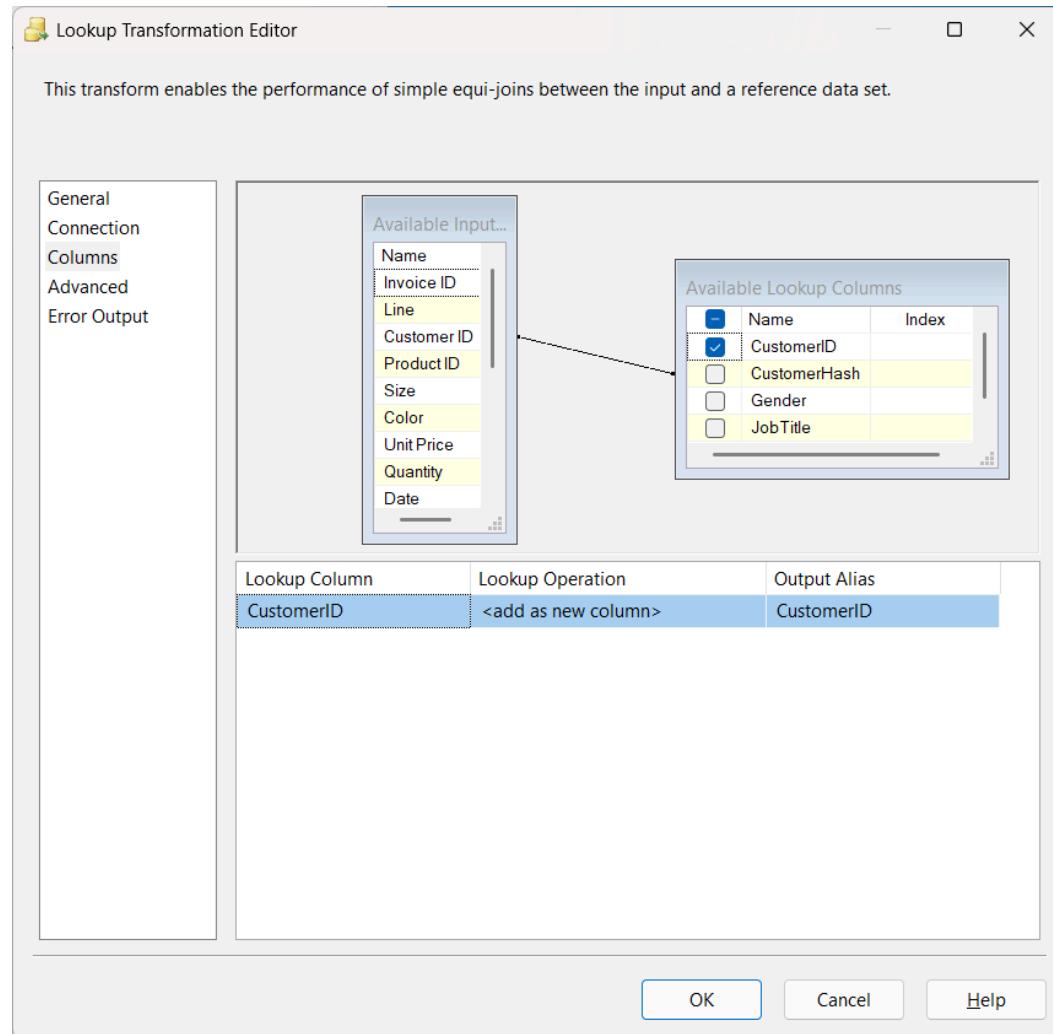
*Hình 305. Tạo chức năng LookupCUSTOMERID trong Fact\_TRANSACTIONS*

- **Bước 6:** Nhấn đúp chuột vào **LookupCUSTOMERID**, màn hình **Lookup Transformation Editor** xuất hiện. Do chúng ta đã thiết lập **OLE DB connection manager** ở bước trước, nên ta có thể sử dụng lại. Ở phần **Use a table or a view**, tiến hành chọn **[dbo].[Dim\_CUSTOMER]**.



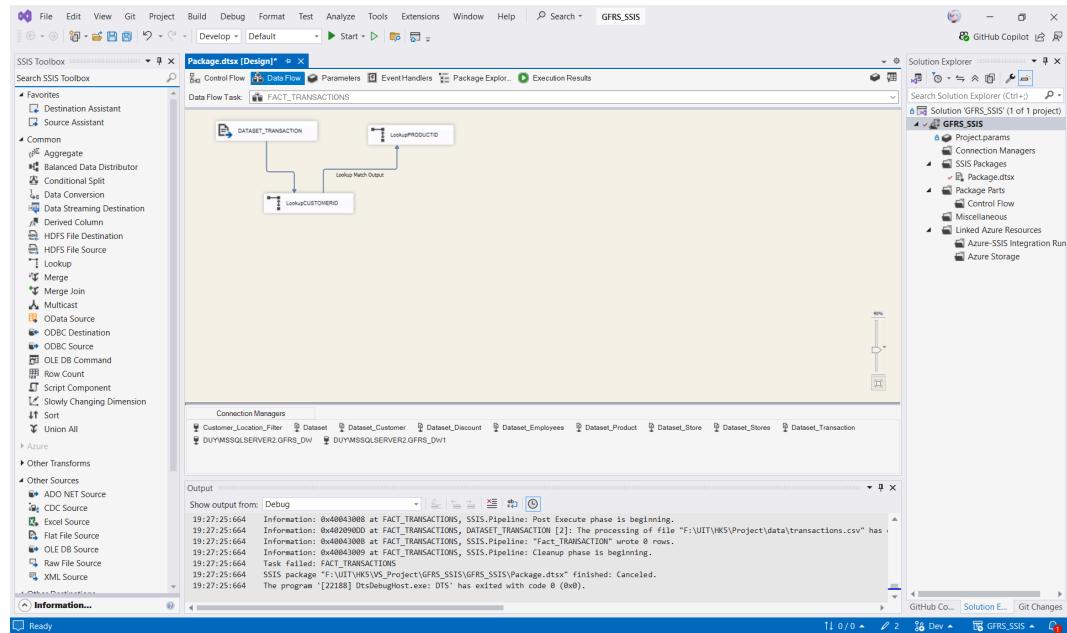
Hình 306. Kết nối nguồn dữ liệu cho LookupCUSTOMERID

- **Bước 7:** Trong tab **Columns**, ở bảng bên phải click chọn **CustomerID**. Ta kéo **Customer ID** từ bảng **Available Input Columns** qua tương ứng **CustomerHash** bảng **Available Lookup Columns** (quá trình này giúp hệ thống xác định vùng dữ liệu cần tìm kiếm). Sau đó bấm **OK** để đóng cửa sổ **Lookup Transformation Editor**



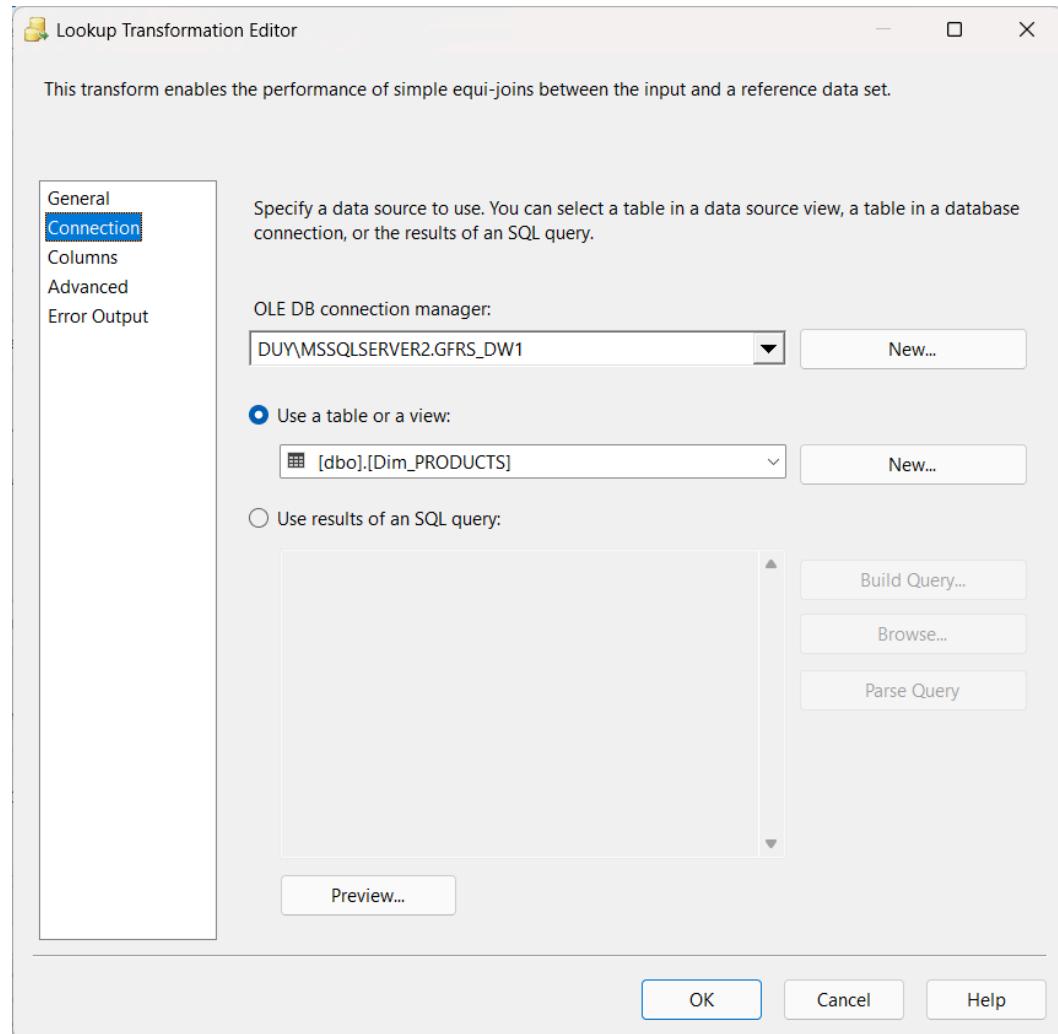
Hình 307. Cài đặt **LookupCUSTOMERID**

- **Bước 8:** Kéo chức năng **Lookup** vào vùng **Data Flow** và đổi tên thành **LookupPRODUCTID**. Kéo mũi tên luồng dữ liệu từ **LookupCUSTOMERID** vào **LookupPRODUCTID**. Một hộp thoại sẽ xuất hiện, chọn **Lookup Match Output** và nhấn **OK**.



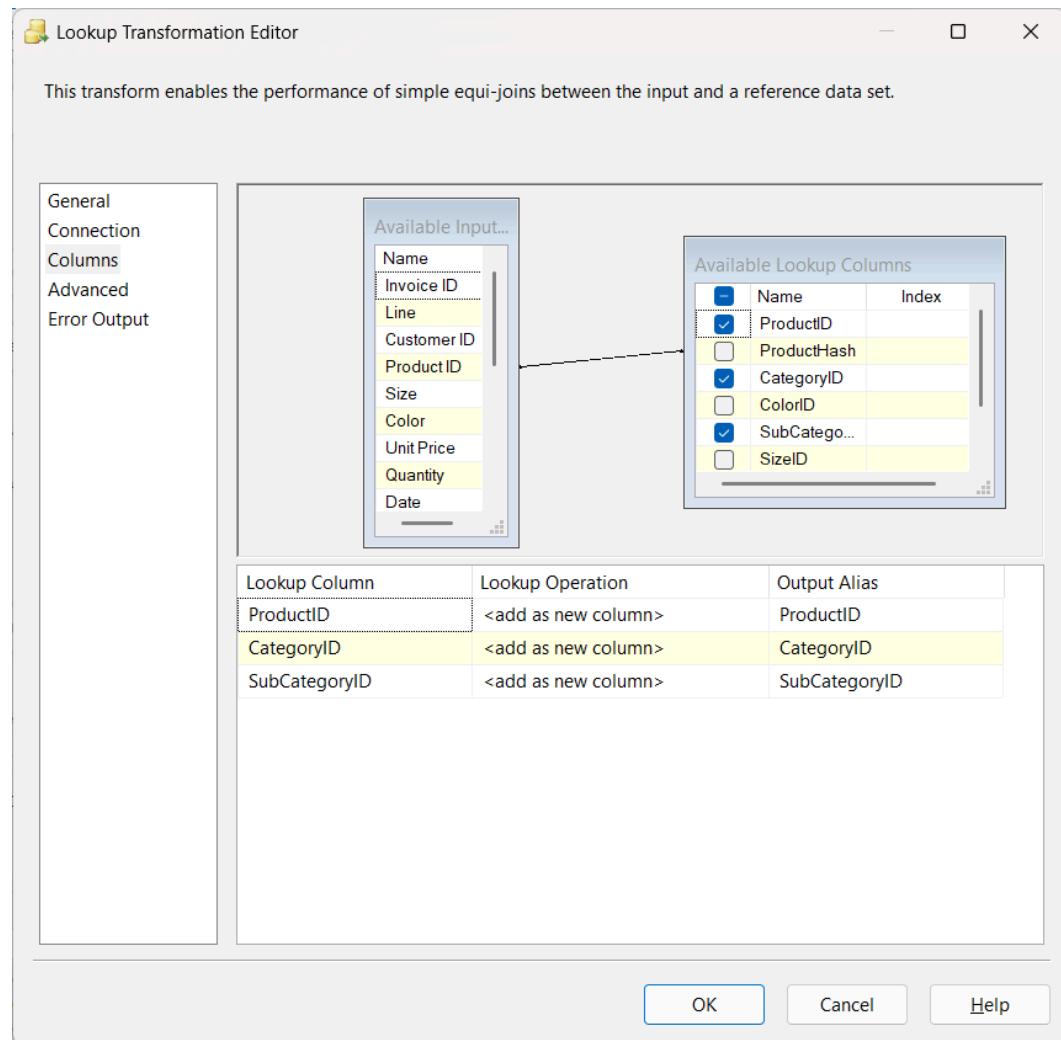
*Hình 308. Tạo chức năng LookupPRODUCTID trong Fact\_TRANSACTIONS*

- **Bước 9:** Nhấn đúp chuột vào **LookupPRODUCTID**, màn hình **Lookup Transformation Editor** xuất hiện. Do chúng ta đã thiết lập **OLE DB connection manager** ở bước trước, nên ta có thể sử dụng lại. Ở phần **Use a table or a view**, tiến hành chọn **[dbo].[Dim\_PRODUCTS]**.



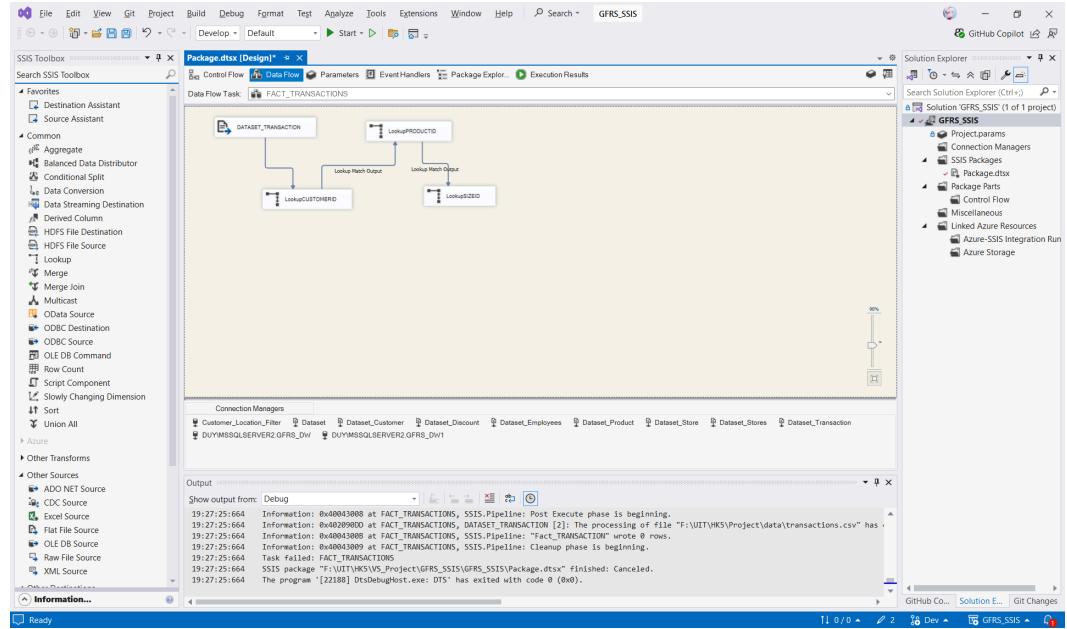
Hình 309. Cài đặt nguồn dữ liệu cho chức năng LookupPRODUCTID

- **Bước 10:** Trong tab **Columns**, ở bảng bên phải click chọn **ProductID**, **CategoryID**, **SubCategoryID**. Ta kéo **Product ID** từ bảng **Available Input Columns** qua tương ứng **ProductHash** bảng **Available Lookup Columns** (quá trình này giúp hệ thống xác định vùng dữ liệu cần tìm kiếm). Sau đó bấm **OK** để đóng cửa sổ **Lookup Transformation Editor**



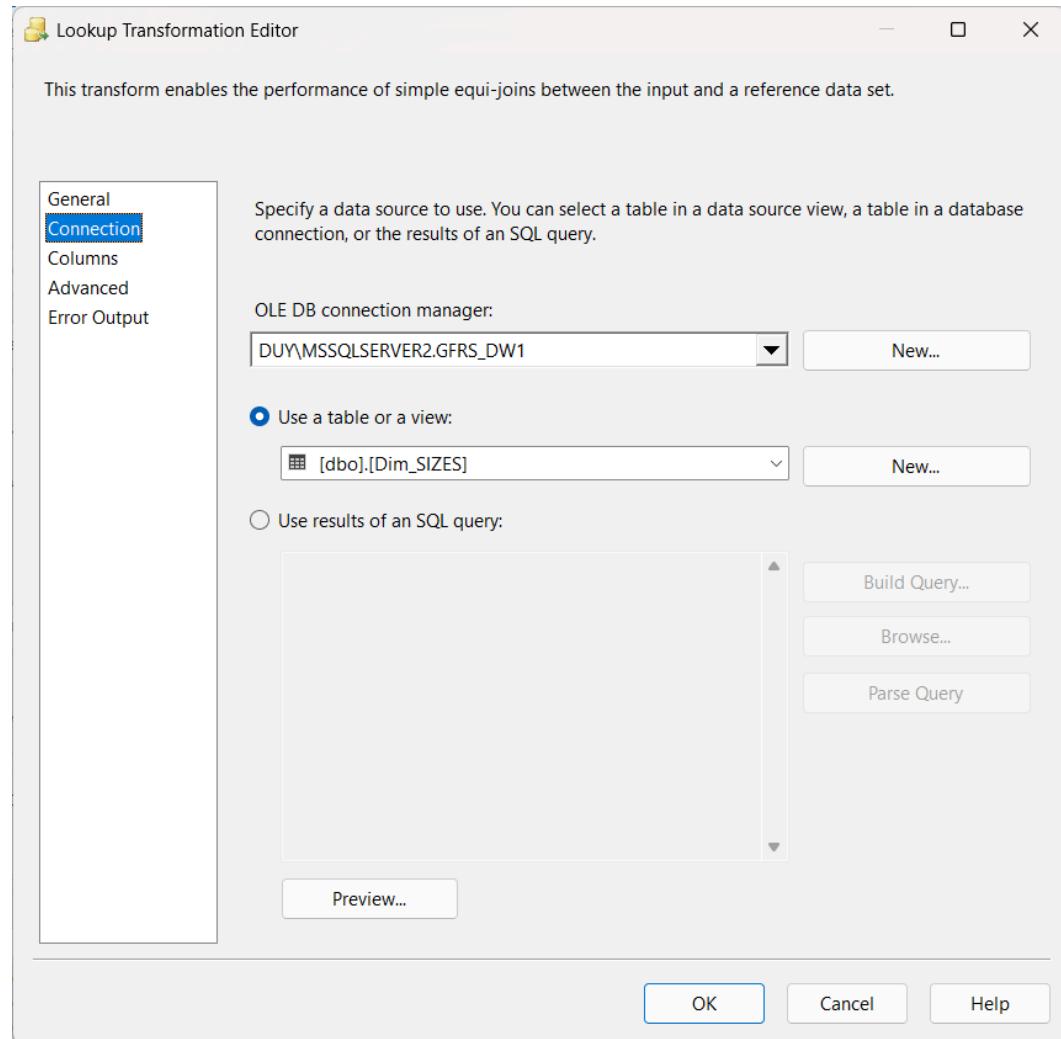
Hình 310. Cài đặt chức năng LookupPRODUCTID

- **Bước 11:** Kéo chức năng **Lookup** vào vùng **Data Flow** và đổi tên thành **LookupSIZEID**. Kéo mũi tên luồng dữ liệu từ **LookupPRODUCTID** vào **LookupSIZEID**. Một hộp thoại sẽ xuất hiện, chọn **Lookup Match Output** và nhấn **OK**.



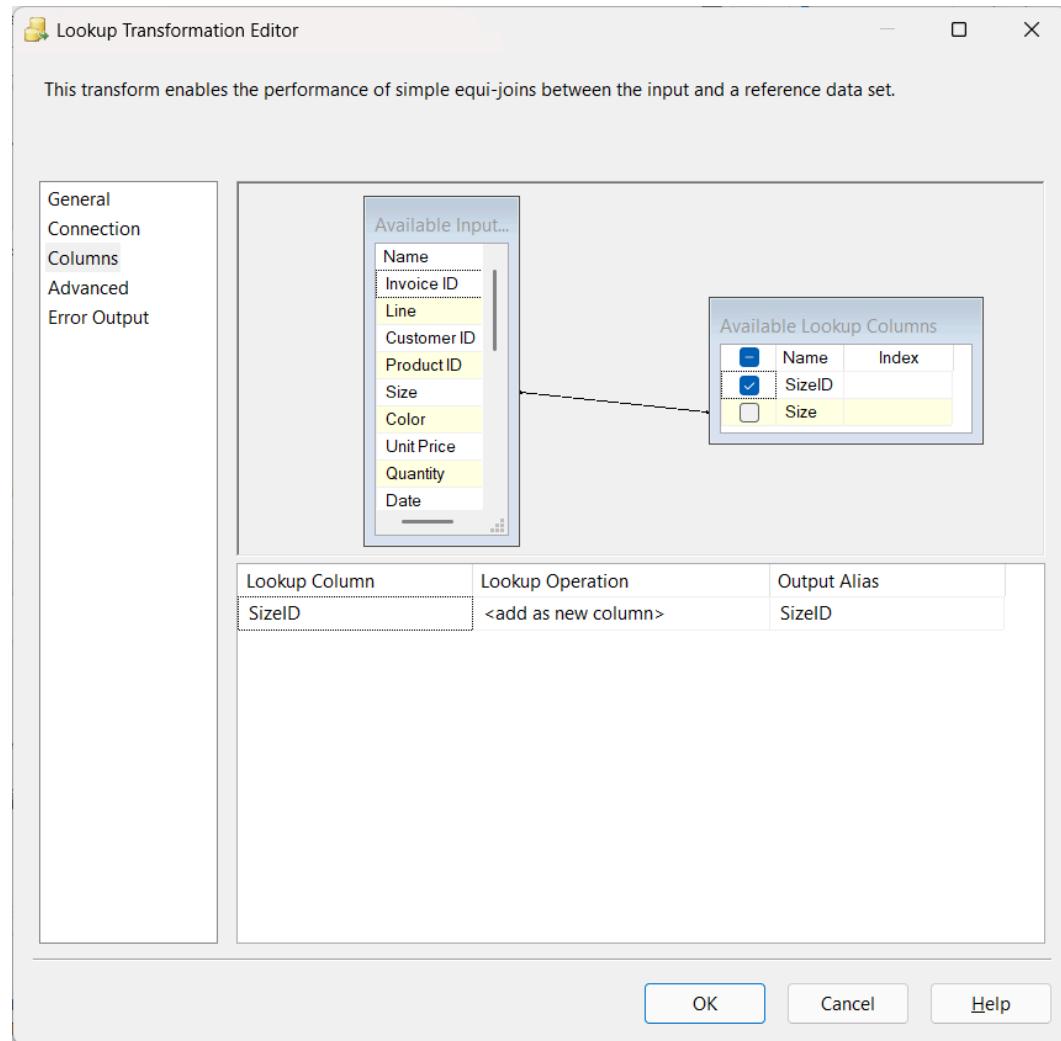
Hình 311. Tạo chức năng LookupSIZEID

- **Bước 12:** Nhấn đúp chuột vào **LookupSIZEID**, màn hình **Lookup Transformation Editor** xuất hiện. Do chúng ta đã thiết lập **OLE DB connection manager** ở bước trước, nên ta có thể sử dụng lại. Ở phần **Use a table or a view**, tiến hành chọn **[dbo].[Dim\_SIZES]**.



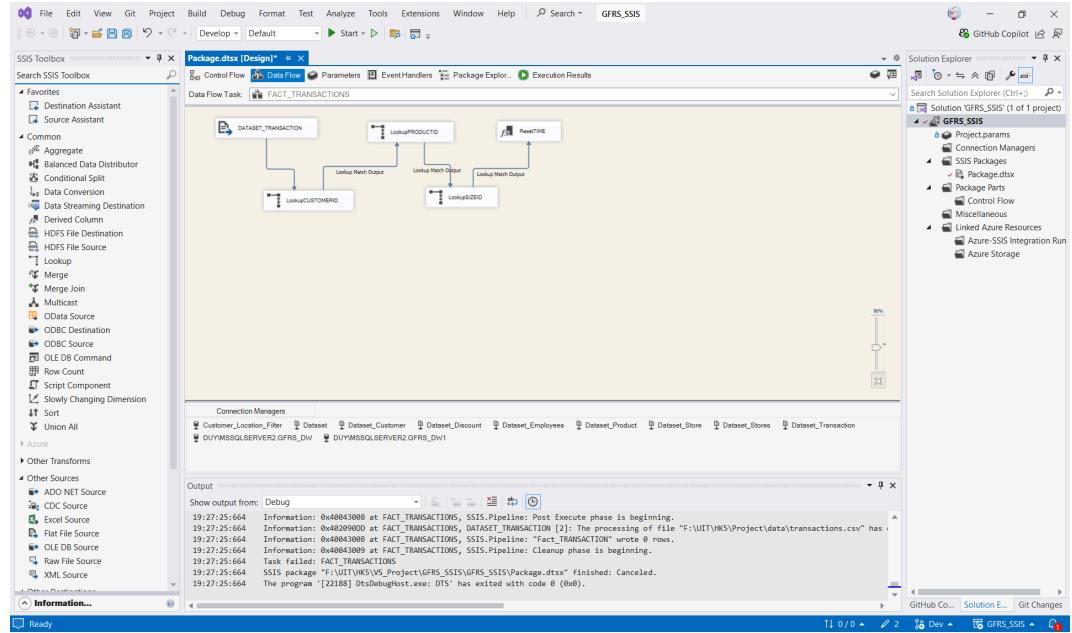
Hình 312. Cài đặt nguồn dữ liệu cho LookupSIZEID

- **Bước 13:** Trong tab **Columns**, ở bảng bên phải click chọn **SizeID**. Ta kéo **Size** từ bảng **Available Input Columns** qua tương ứng **Size** bảng **Available Lookup Columns** (quá trình này giúp hệ thống xác định vùng dữ liệu cần tìm kiếm). Sau đó bấm **OK** để đóng cửa sổ **Lookup Transformation Editor**.



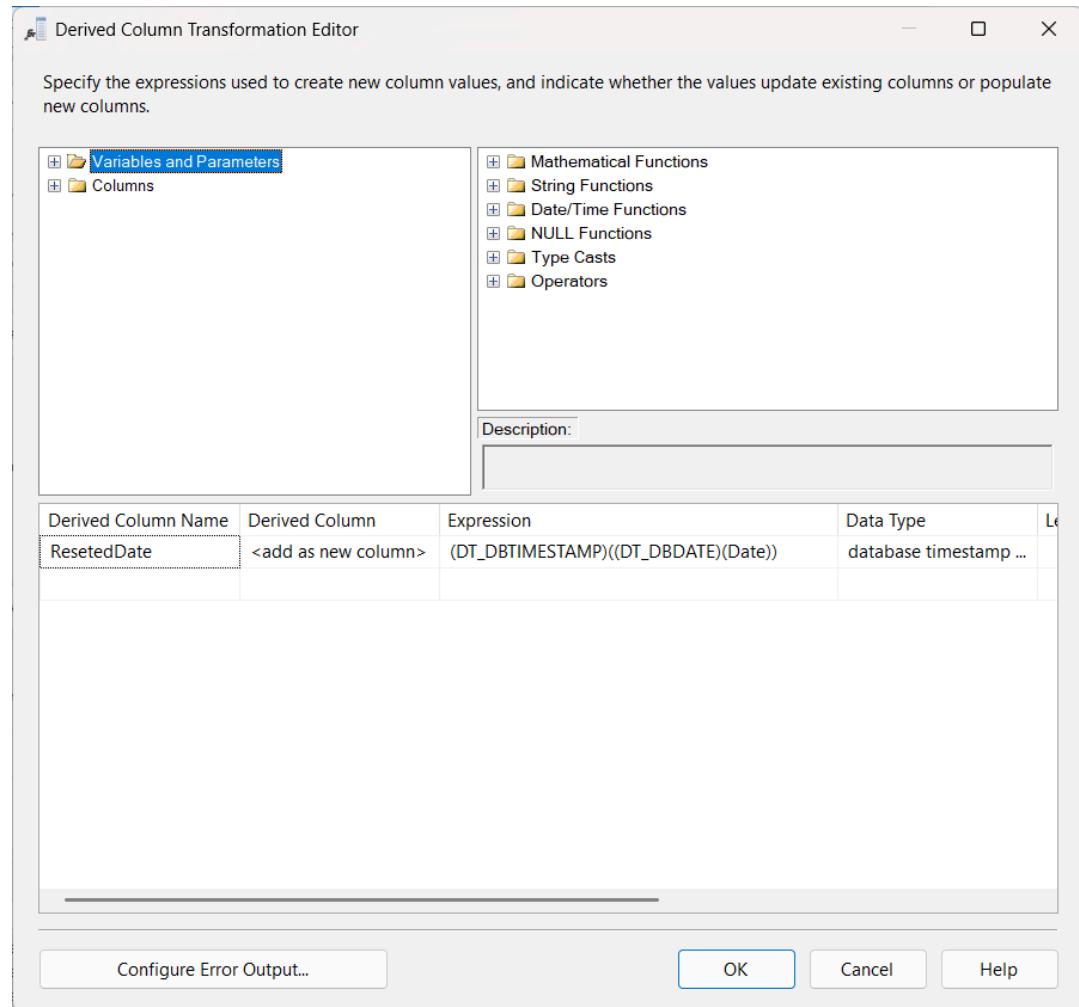
Hình 313. Cài đặt chức năng LookupSIZEID

- **Bước 14:** Kéo chức năng **Derived Column** vào vùng **Data Flow** và đổi tên thành **ResetTIME** (đổi ngày có giờ phút giây cũ thẻ sang ngày có giờ phút giây bằng 0). Kéo mũi tên luồng dữ liệu từ **LookupSIZEID** vào **ResetTIME**. Một hộp thoại sẽ xuất hiện, chọn **Lookup Match Output** và nhấn **OK**.



Hình 314. Tạo chức năng Derived Column trong Fact\_TRANSACTIONS

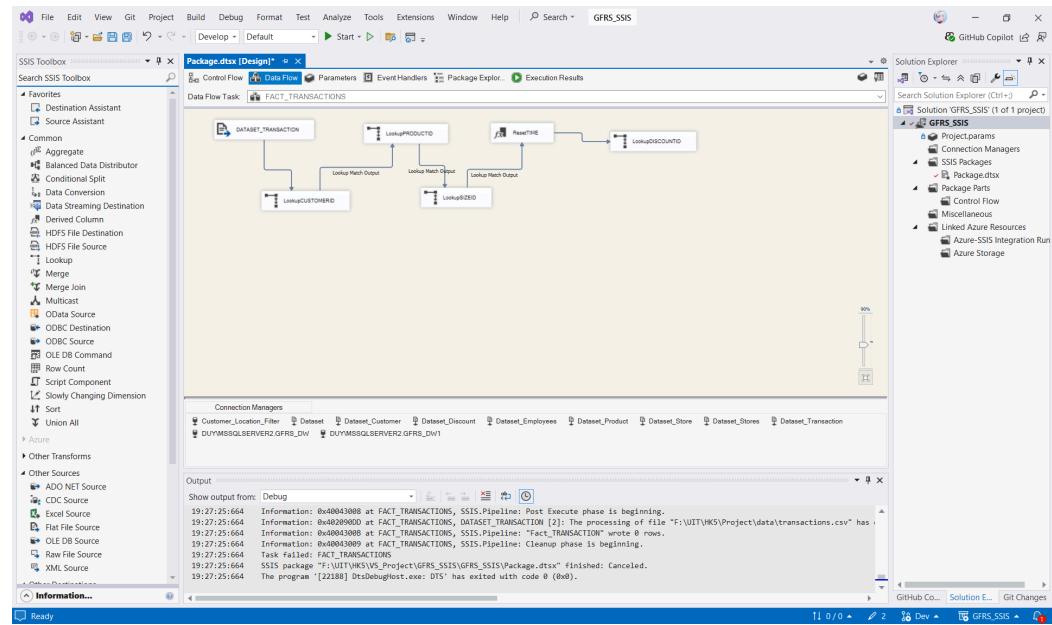
- **Bước 15:** Nhấn đúp chuột vào **ResetTIME**, một cửa sổ **Derived Column Transformation Editor** sẽ hiện lên. Ta tiến hành tạo các cột **ResetedDATE** mới với **Expression** là **(DT\_DBTIMESTAMP)((DT\_DBDATE)(Date))**



*Hình 315. Cài đặt chức năng ResetTIME*

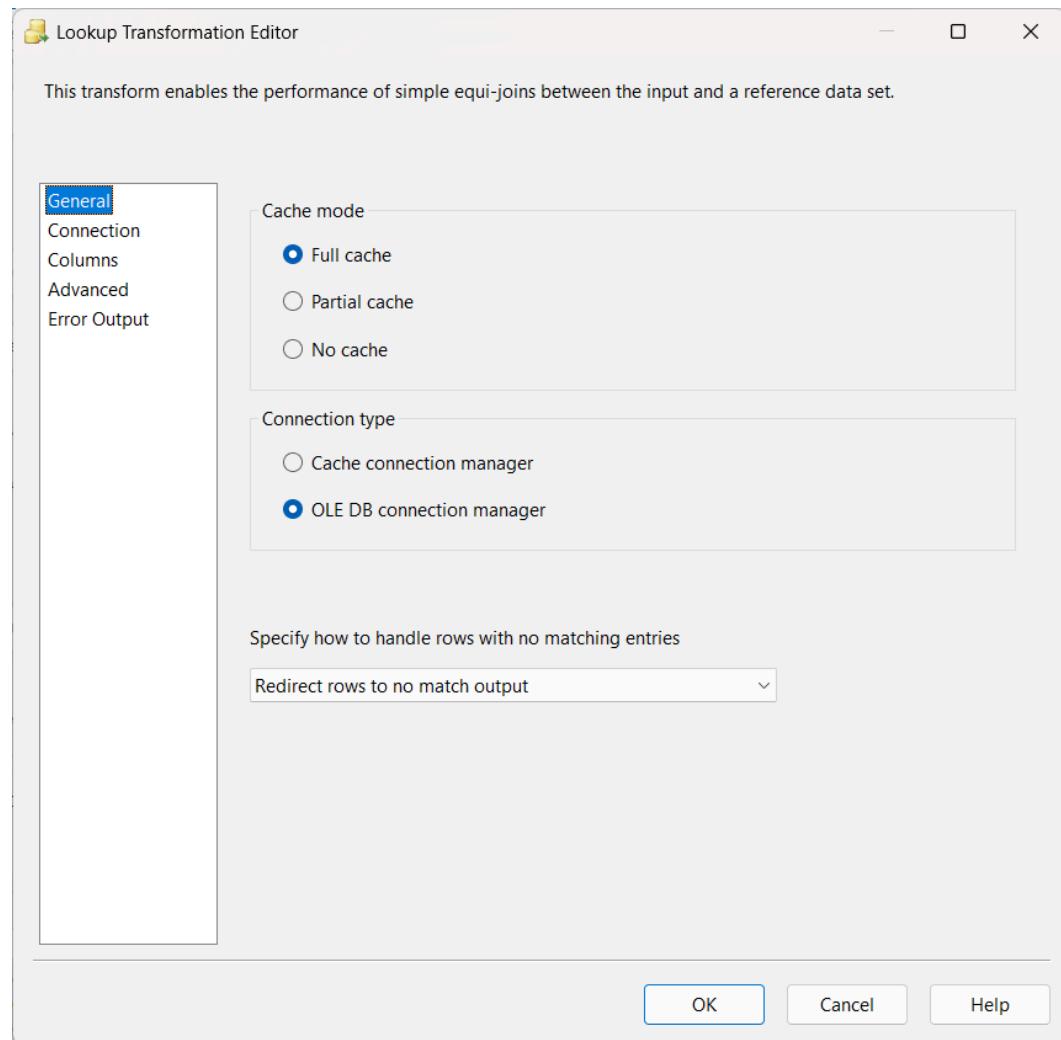
Sau đó nhấn **OK** và kết thúc cửa sổ **Derived Column Transformation Editor**

- **Bước 16:** Kéo chức năng **Lookup** vào vùng **Data Flow** và đổi tên thành **LookupDISCOUNTID**. Kéo mũi tên luồng dữ liệu từ **ResetTIME** vào **LookupDISCOUNTID**.



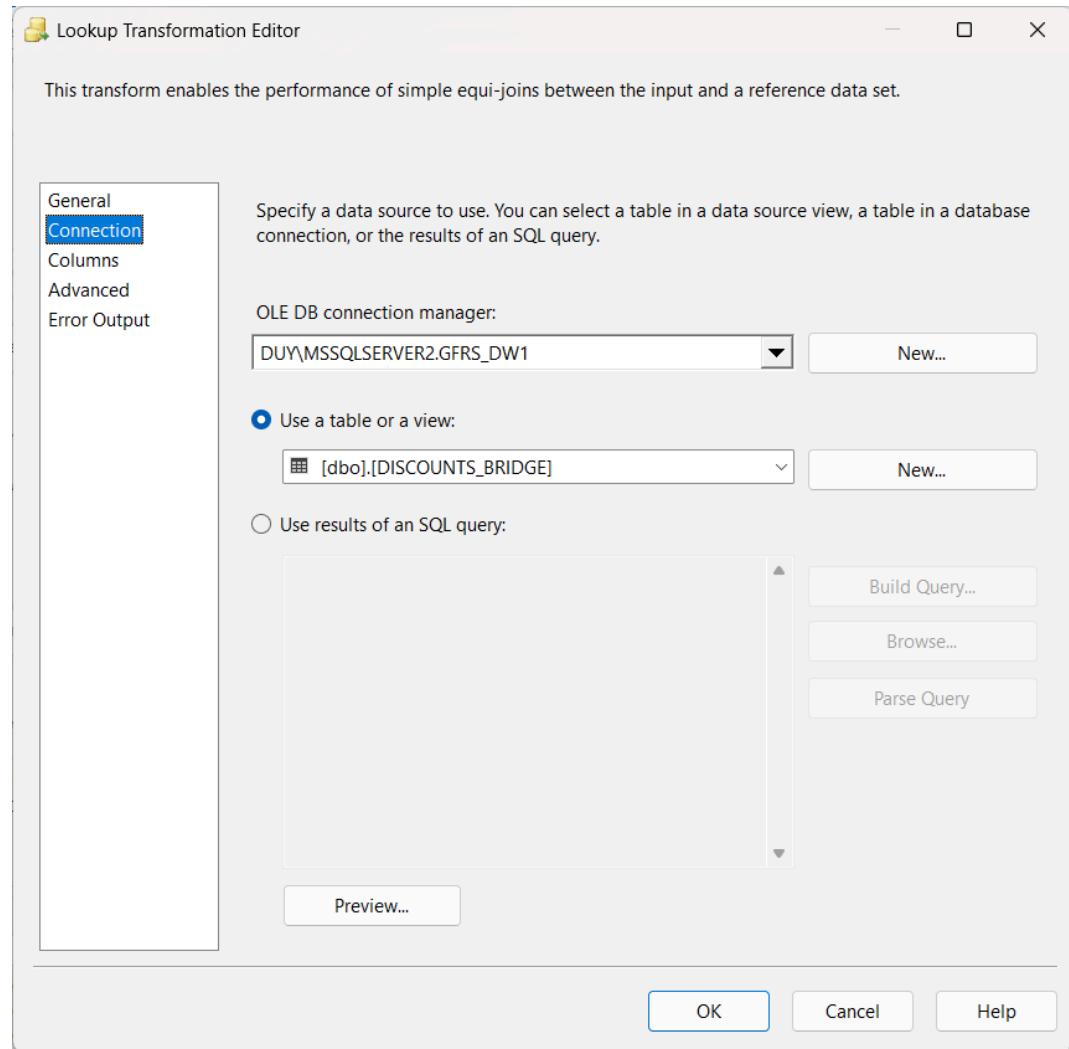
Hình 316. Tạo chức năng LookupDISCOUNTID trong Fact\_TRANSACTIONS

- **Bước 17:** Nhấn đúp chuột vào **LookupDISCOUNTID**, màn hình **Lookup Transformation Editor** xuất hiện. Ở tab **General**, phần **Specify how to handle rows with no matching entries** ta chọn **Redirect rows to no match output**.



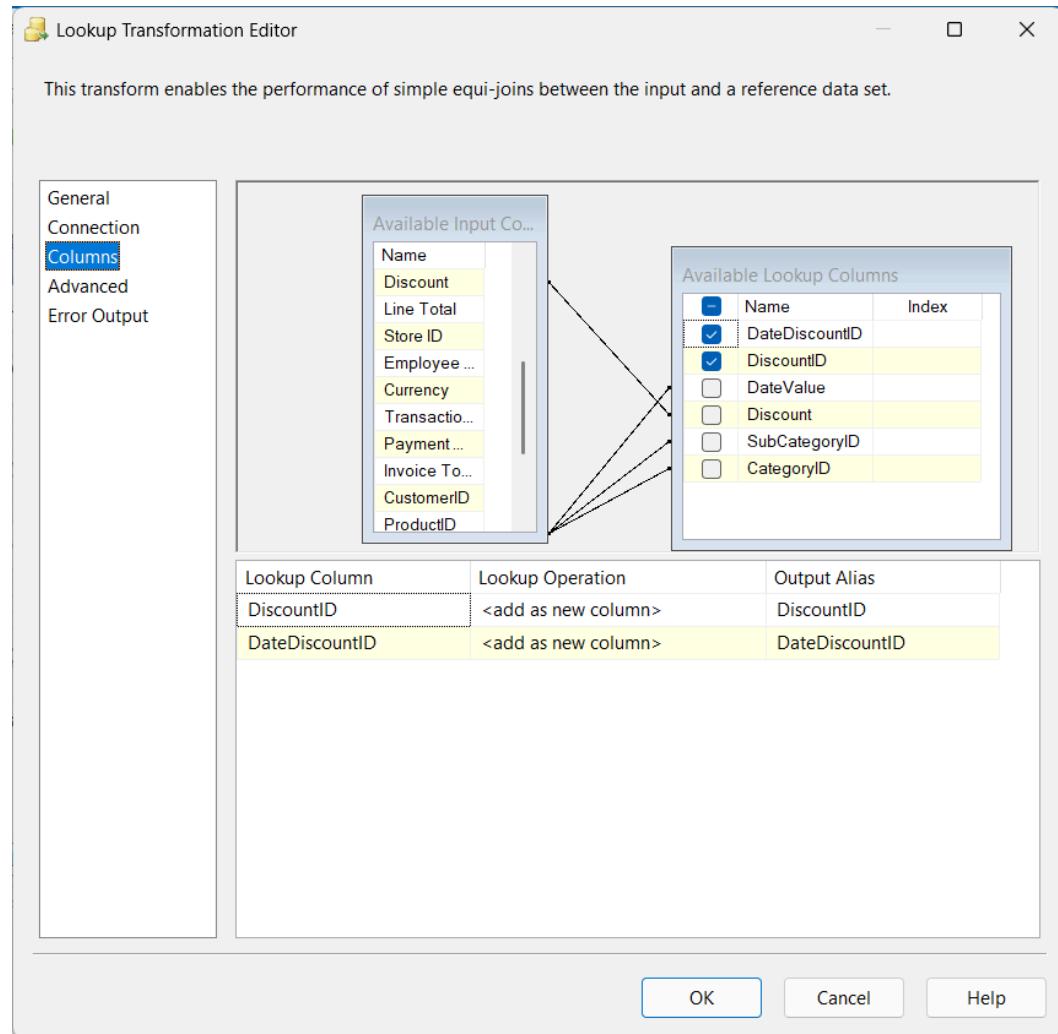
Hình 317. Cài đặt LookupDISCOUNTID ở tab General

- **Bước 18:** Chuyển sang tab connection, do chúng ta đã thiết lập **OLE DB connection manager** ở bước trước, nên ta có thể sử dụng lại. Ở phần **Use a table or a view**, tiến hành chọn **[dbo].[DISCOUNT\_BRIDGE]**.



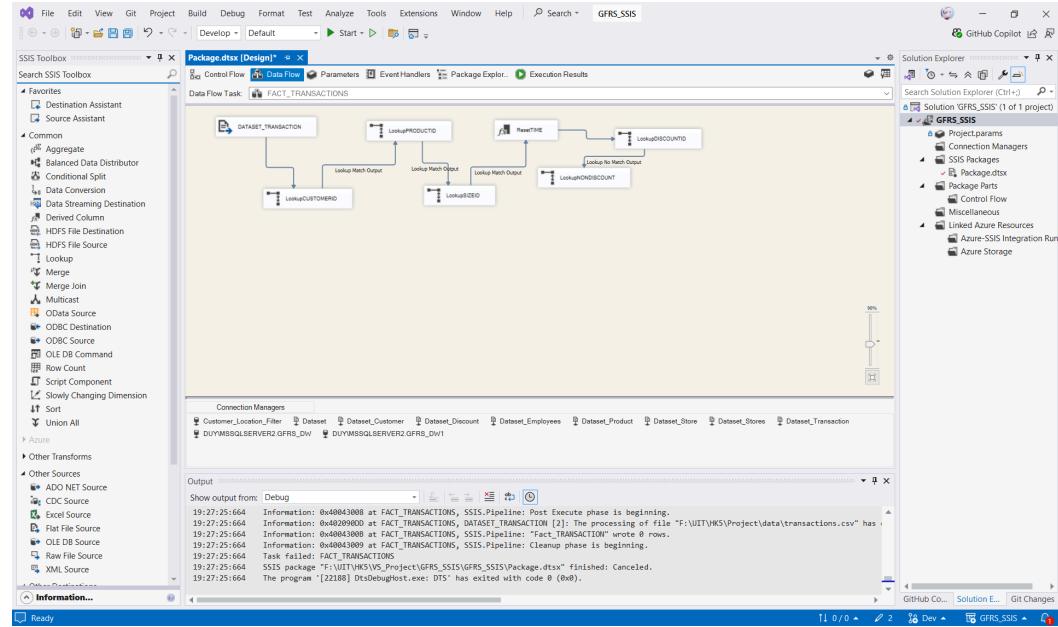
Hình 318. Kết nối nguồn dữ liệu cho LookupDISCOUNTID

- **Bước 19:** Trong tab **Columns**, ở bảng bên phải click chọn **DiscountID**. Ta kéo **Discount**, **DateValue**, **Discount**, **SubcategoryID**, **CategoryID** từ bảng **Available Input Columns** qua tương ứng **Discount**, **DateValue**, **Discount**, **SubcategoryID**, **CategoryID** bảng **Available Lookup Columns** (quá trình này giúp hệ thống xác định vùng dữ liệu cần tìm kiếm). Sau đó bấm **OK** để đóng cửa sổ **Lookup Transformation Editor**.



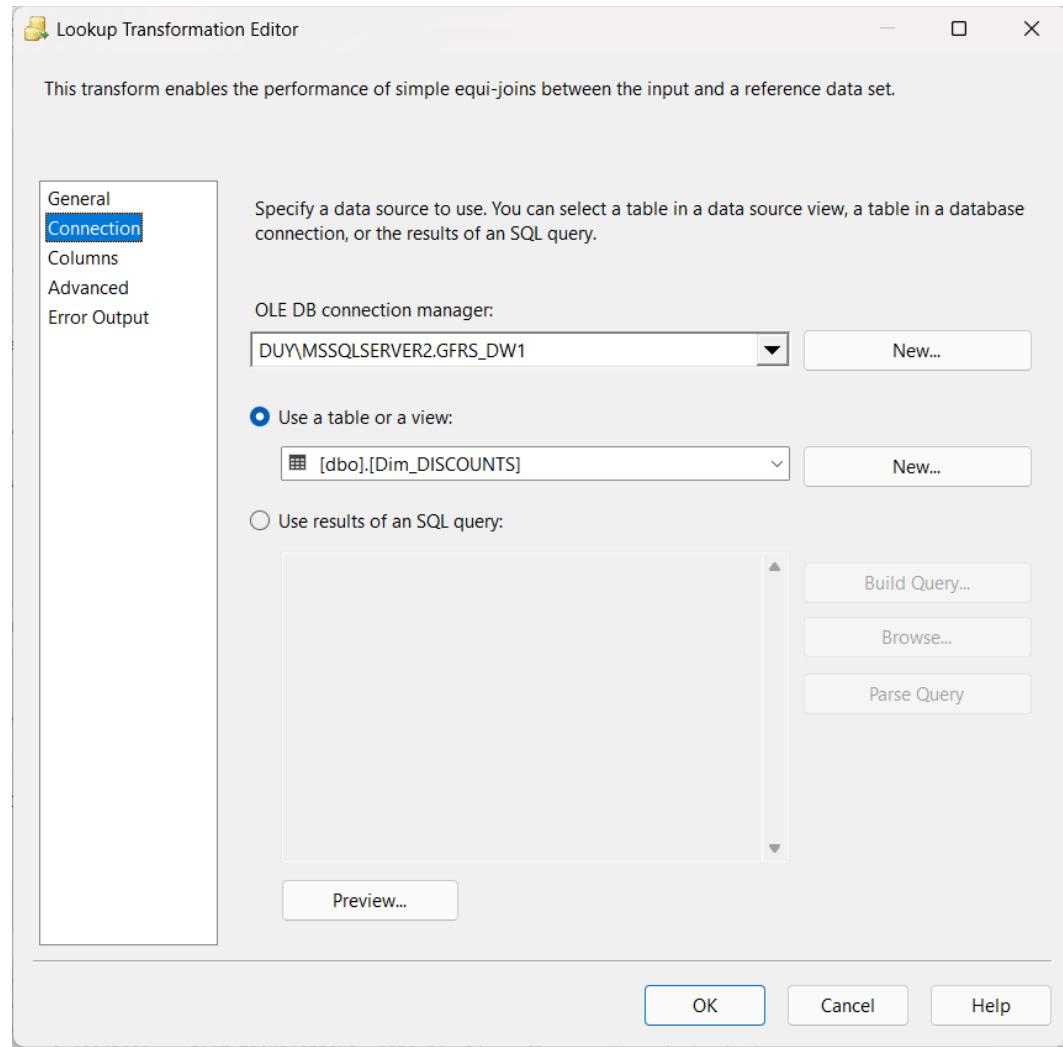
Hình 319. Cài đặt chức năng LookupDISCOUNTID

- **Bước 20:** Kéo chức năng **Lookup** vào vùng **Data Flow** và đổi tên thành **LookupNONDISCOUNT** (gắn mã cho các giao dịch không có mã giảm giá). Kéo mũi tên luồng dữ liệu từ **LookupDISCOUNTID** vào **LookupNONDISCOUNT**. Một hộp thoại sẽ xuất hiện, chọn **Lookup No Match Output** và nhấn **OK**.



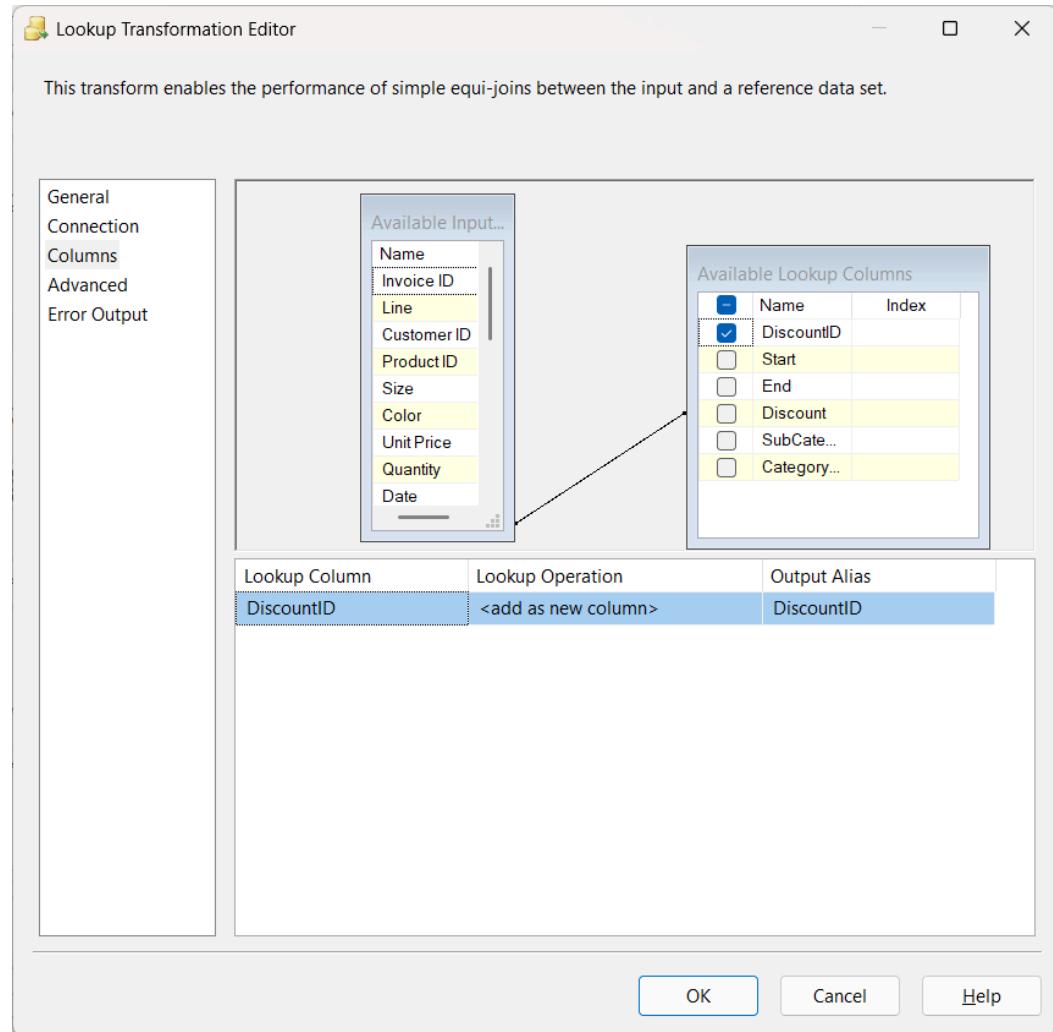
Hình 320. Tạo chức năng **LookupNODISCOUNT**

- **Bước 21:** Nhấn đúp chuột vào **LookupNODISCOUNT**, màn hình **Lookup Transformation Editor** xuất hiện. Do chúng ta đã thiết lập **OLE DB connection manager** ở bước trước, nên ta có thể sử dụng lại. Ở phần **Use a table or a view**, tiến hành chọn **[dbo].[Dim\_DISCOUNTS]**.



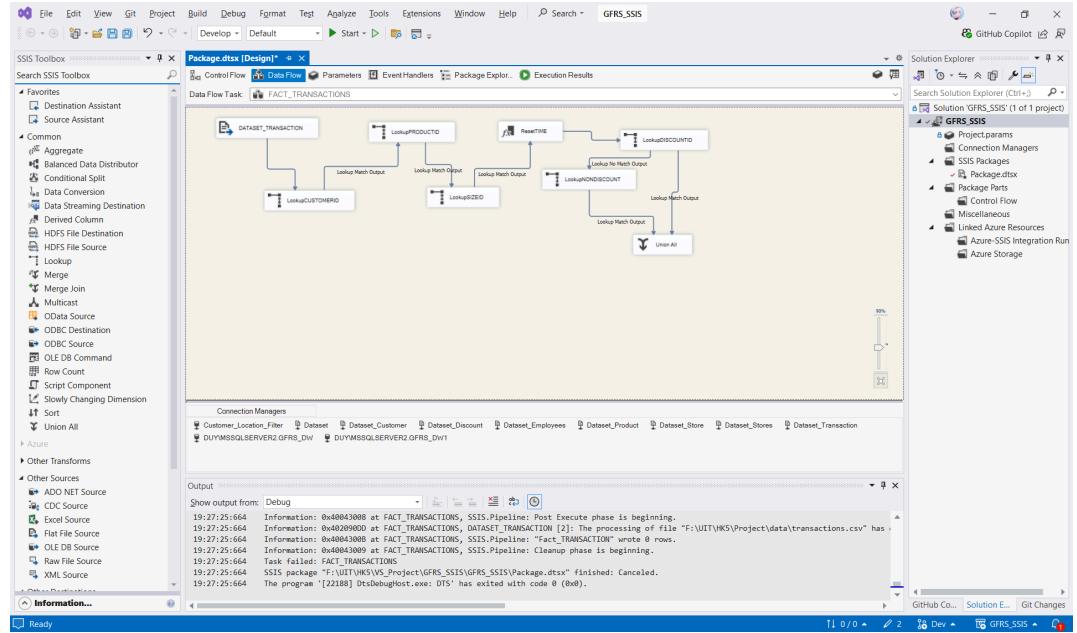
Hình 321. Kết nối nguồn dữ liệu cho Lookup NONDISCOUNT

- **Bước 22:** Trong tab **Columns**, ở bảng bên phải click chọn **DiscountID**. Ta kéo **Discount** từ bảng **Available Input Columns** qua tương ứng **Discount** bảng **Available Lookup Columns** (quá trình này giúp hệ thống xác định vùng dữ liệu cần tìm kiếm). Sau đó bấm **OK** để đóng cửa sổ **Lookup Transformation Editor**.



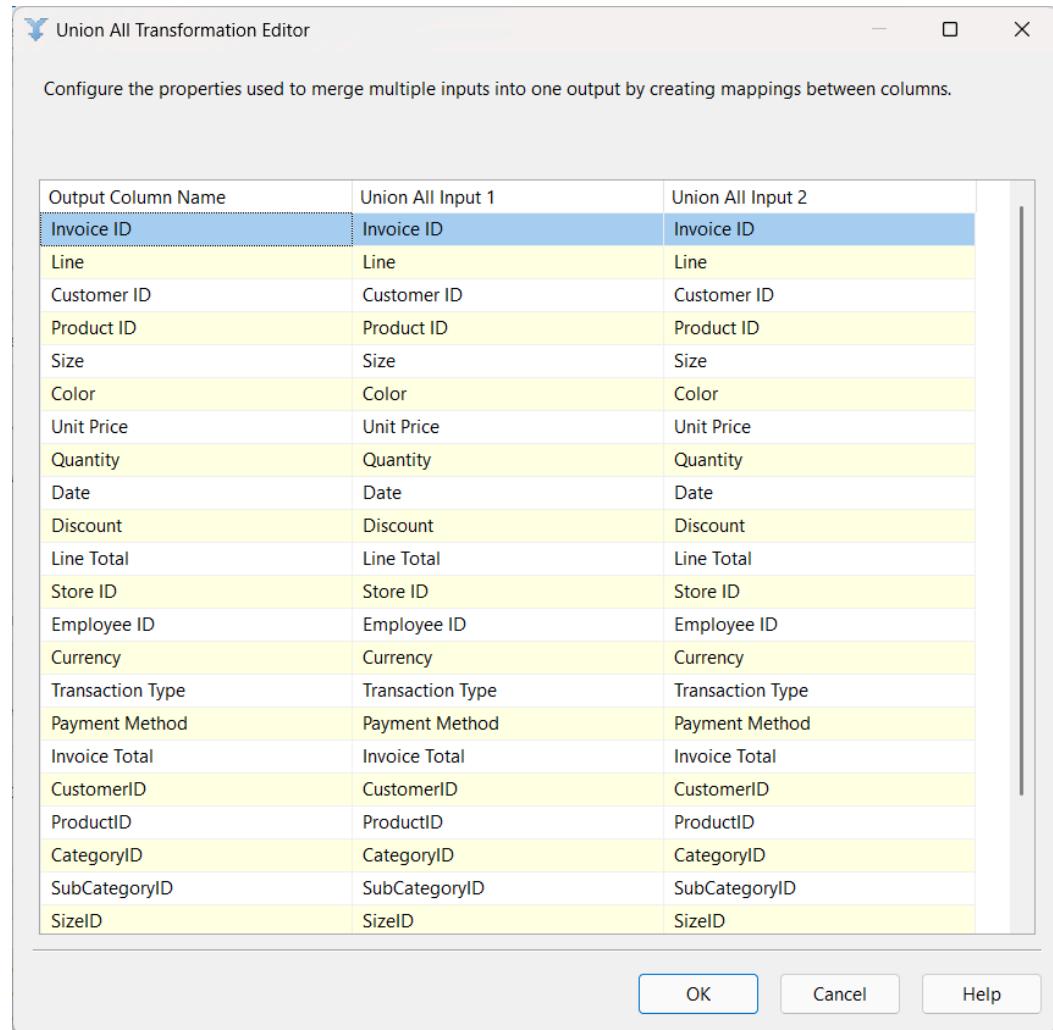
Hình 322. Cài đặt chức năng LookupNONDISCOUNT

- **Bước 23:** Kéo chức năng **Union All** từ vùng **SSIS Toolbox** vào **Data Flow**. Tiếp đó ta lần lượt luồng dữ liệu từ **LookupNONDISCOUNT** và **LookupDISCOUNTID** vào **Union All**. Khi kéo **LookupNONDISCOUNT** sẽ xuất hiện 1 hộp thoại ta chọn **Lookup Match Output**



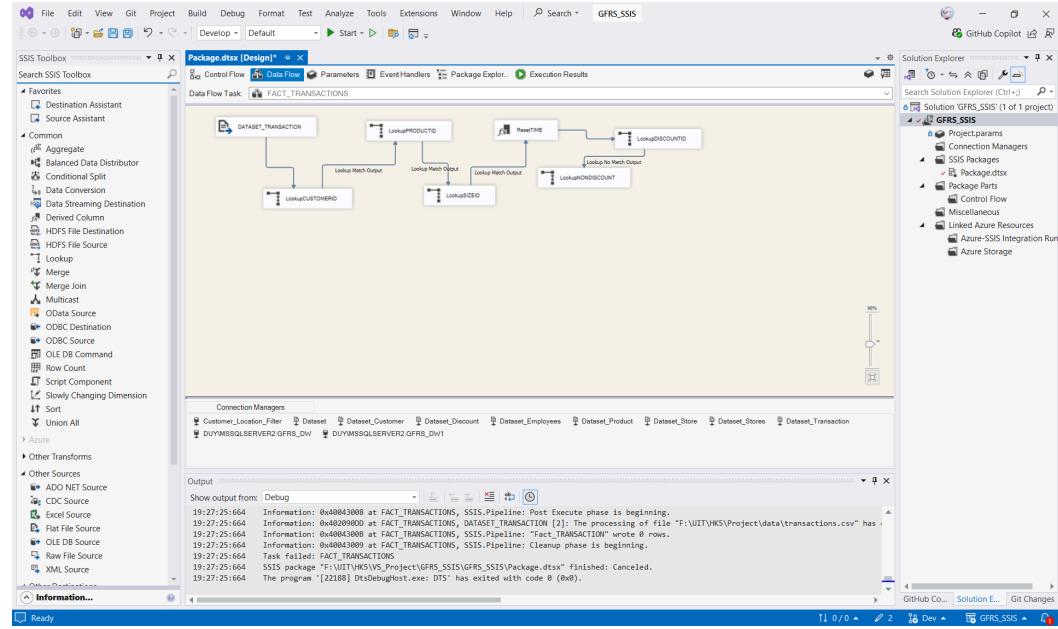
Hình 323. Tạo chức năng Union All trong Fact\_TRANSACTIONS

- **Bước 24:** Nhấn đúp chuột vào **Union All**, ta kiểm tra xem các cột cần nối đã trùng khớp với nhau hay chưa. Sau đó nhấn **OK** để đóng cửa sổ **Union All Transformation Editor**



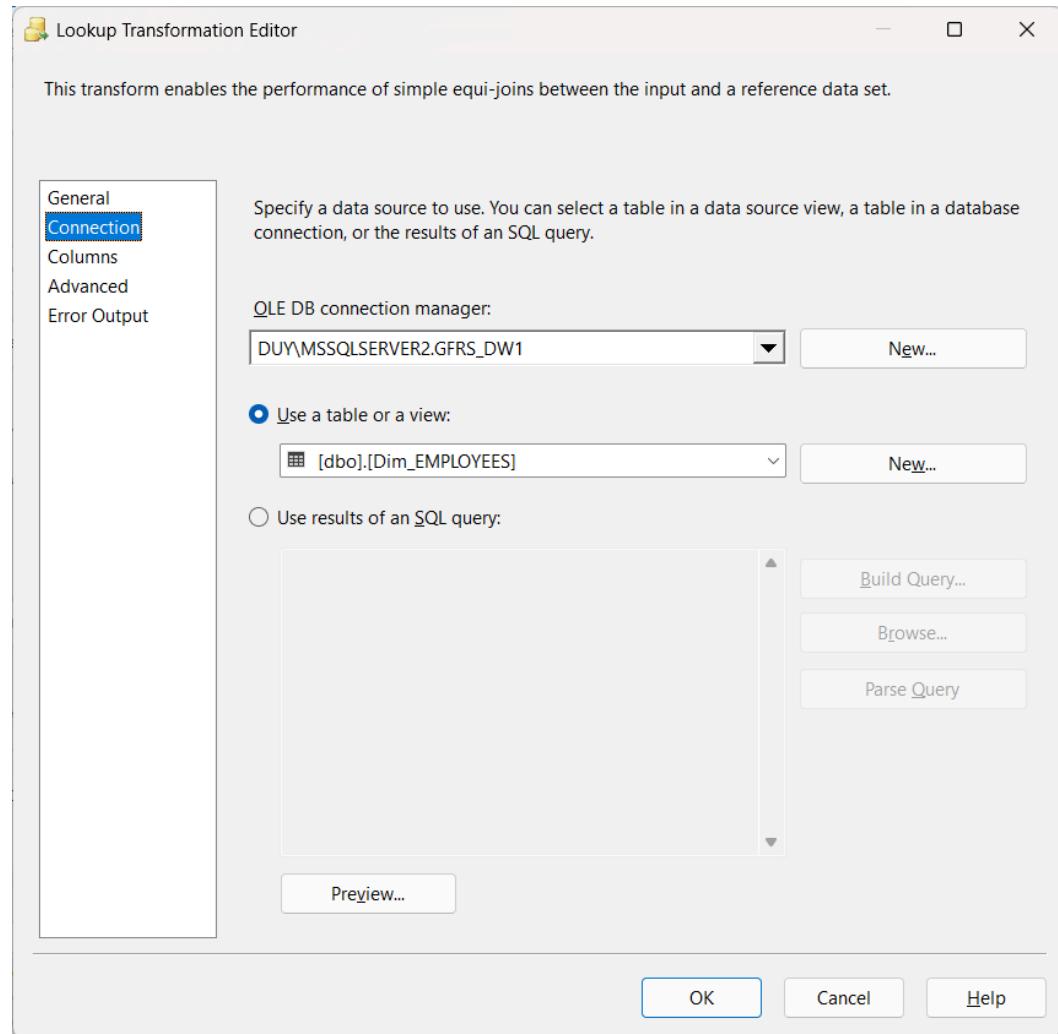
Hình 324. Cài đặt chức năng Union All trong Fact\_TRANSACTIONS

- **Bước 25:** Kéo chức năng Lookup vào vùng Data Flow và đổi tên thành **LookupEMPLOYEEID**. Kéo mũi tên luồng dữ liệu từ **Union All** vào **LookupEMPLOYEEID**.



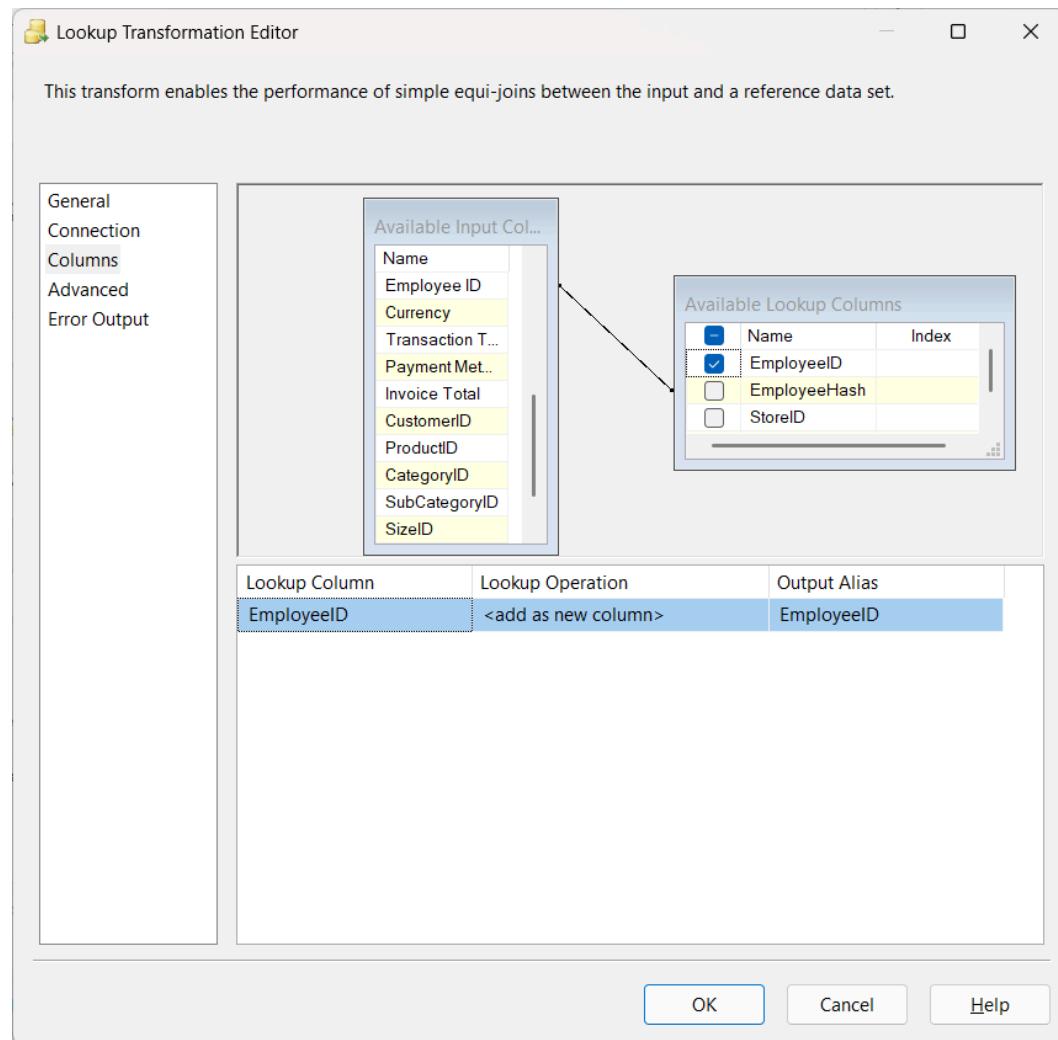
Hình 325. Tạo chức năng LookupEMPLOYEEID

- **Bước 26:** Nhấn đúp chuột vào **LookupNODISCOUNT**, màn hình **Lookup Transformation Editor** xuất hiện. Do chúng ta đã thiết lập **OLE DB connection manager** ở bước trước, nên ta có thể sử dụng lại. Ở phần **Use a table or a view**, tiến hành chọn **[dbo].[Dim\_EMPLOYEES]**.



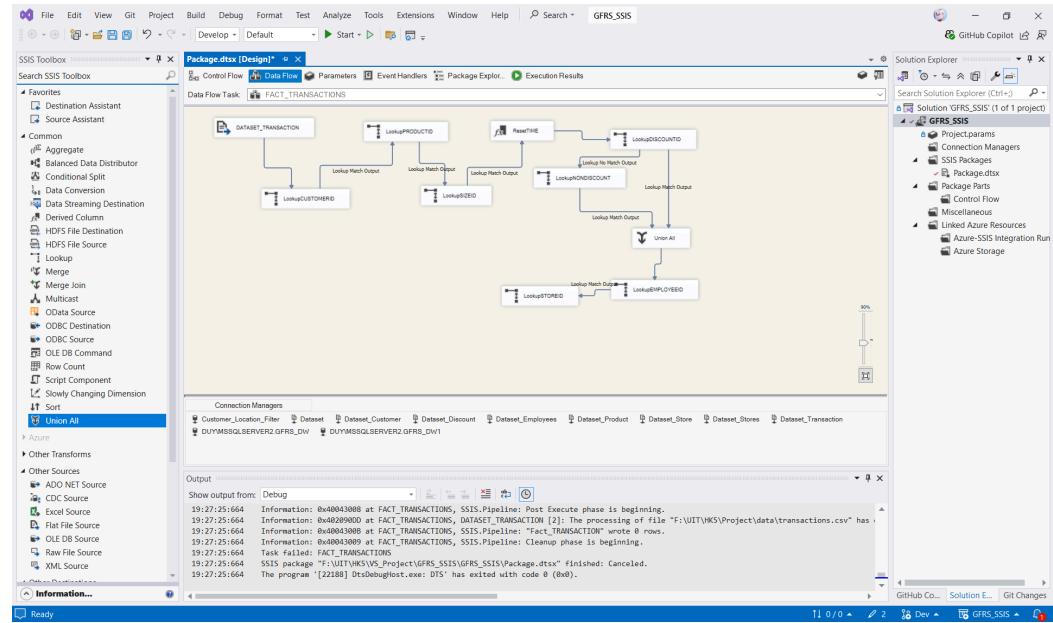
Hình 326. Kết nối nguồn dữ liệu cho chức năng LookupEMPLOYEEID

- **Bước 27:** Trong tab **Columns**, ở bảng bên phải click chọn **EmployeeID**. Ta kéo **Employee ID** từ bảng **Available Input Columns** qua tương ứng **EmployeeHash** bảng **Available Lookup Columns** (quá trình này giúp hệ thống xác định vùng dữ liệu cần tìm kiếm). Sau đó bấm **OK** để đóng cửa sổ **Lookup Transformation Editor**.



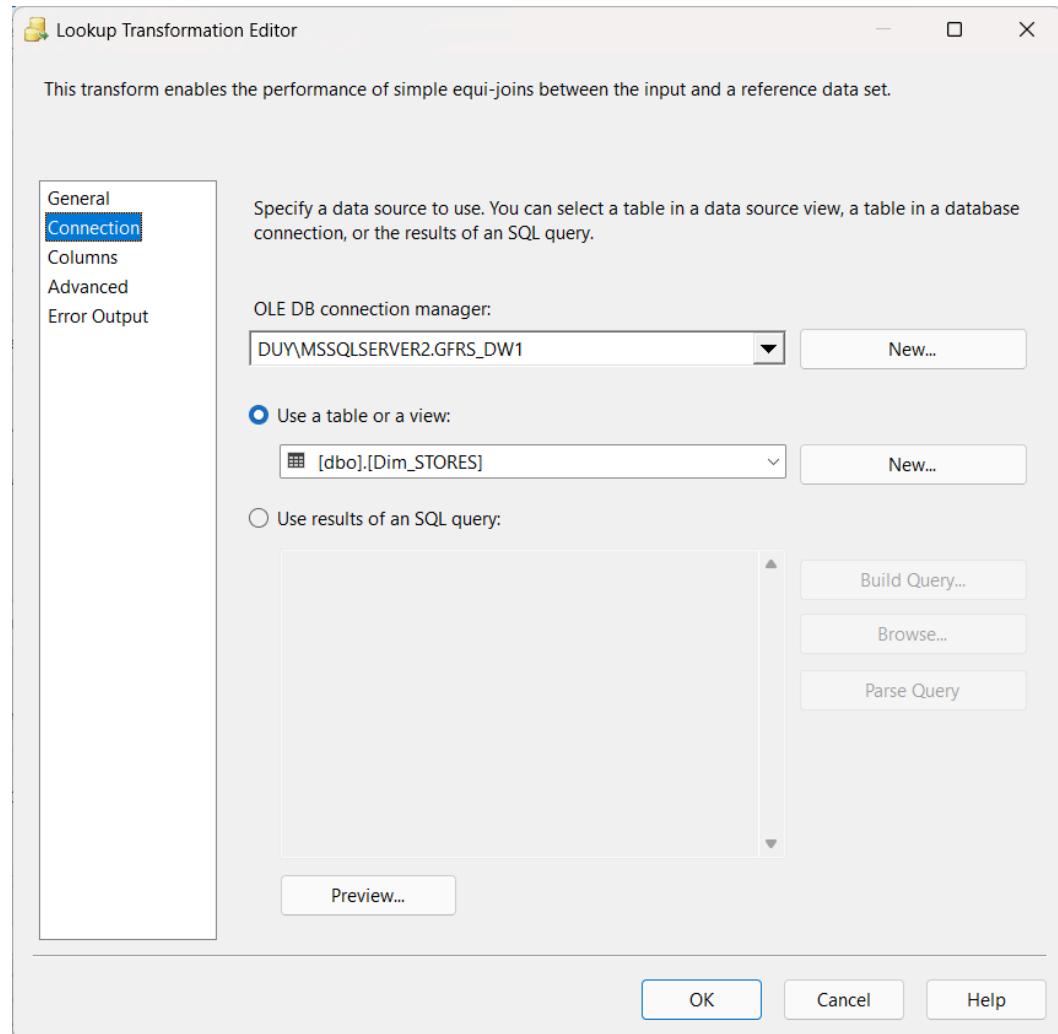
Hình 327. Cài đặt chức năng LookupEMPLOYEEID trong  
Fact\_TRANSACTIONS

- **Bước 28:** Kéo chức năng **Lookup** vào vùng **Data Flow** và đổi tên thành **LookupSTOREID**. Kéo mũi tên luồng dữ liệu từ **LookupEMPLOYEEID** vào **LookupSTOREID**. Một hộp thoại sẽ xuất hiện, chọn **Lookup Match Output** và nhấn **OK**.



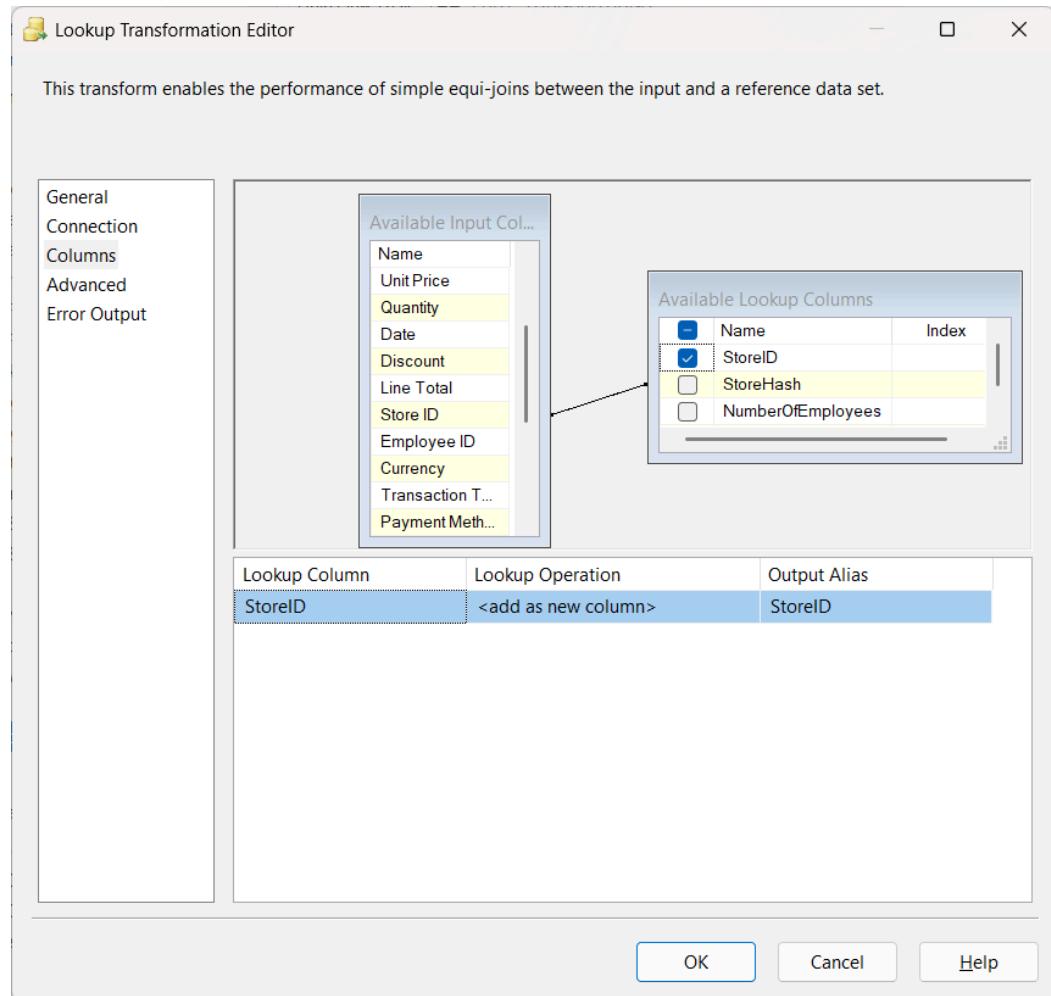
Hình 328. Tạo chức năng LookupSTOREID trong Fact\_TRANSACTIONS

- **Bước 29:** Nhấn đúp chuột vào **LookupSTOREID**, màn hình **Lookup Transformation Editor** xuất hiện. Do chúng ta đã thiết lập **OLE DB connection manager** ở bước trước, nên ta có thể sử dụng lại. Ở phần **Use a table or a view**, tiến hành chọn **[dbo].[Dim\_STORES]**.



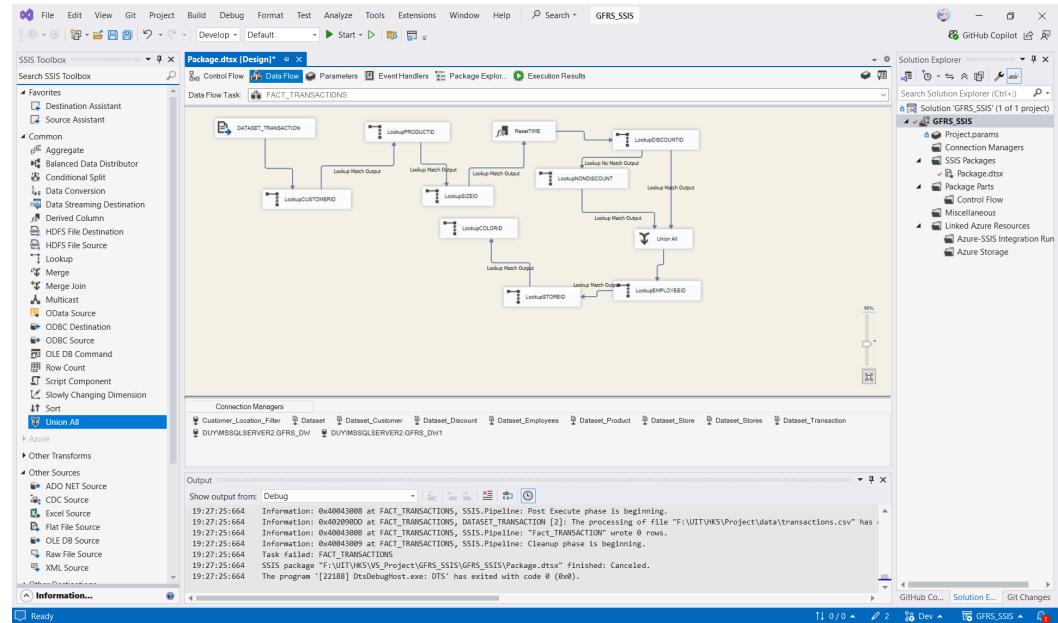
Hình 329. Kết nối nguồn dữ liệu cho chức năng LookupSTOREID

- **Bước 30:** Trong tab **Columns**, ở bảng bên phải click chọn **StoreID**. Ta kéo **Store ID** từ bảng **Available Input Columns** qua tương ứng **StoreHash** bảng **Available Lookup Columns** (quá trình này giúp hệ thống xác định vùng dữ liệu cần tìm kiếm). Sau đó bấm **OK** để đóng cửa sổ **Lookup Transformation Editor**.



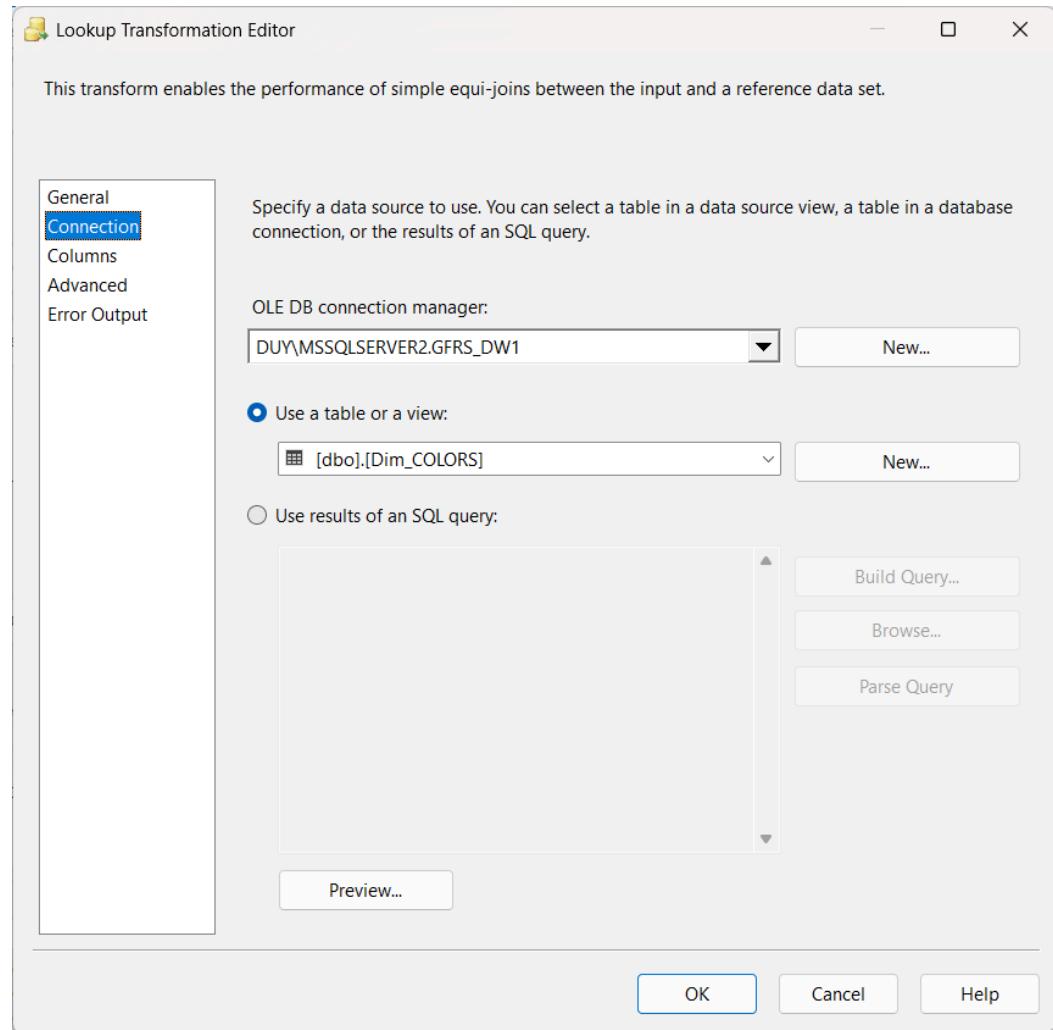
Hình 330. Cài đặt chức năng LookupSTOREID trong  
Fact\_TRANSACTION

- **Bước 31:** Kéo chức năng **Lookup** vào vùng **Data Flow** và đổi tên thành **LookupCOLORID**. Kéo mũi tên luồng dữ liệu từ **LookupSTOREID** vào **LookupCOLORID**. Một hộp thoại sẽ xuất hiện, chọn **Lookup Match Output** và nhấn **OK**.



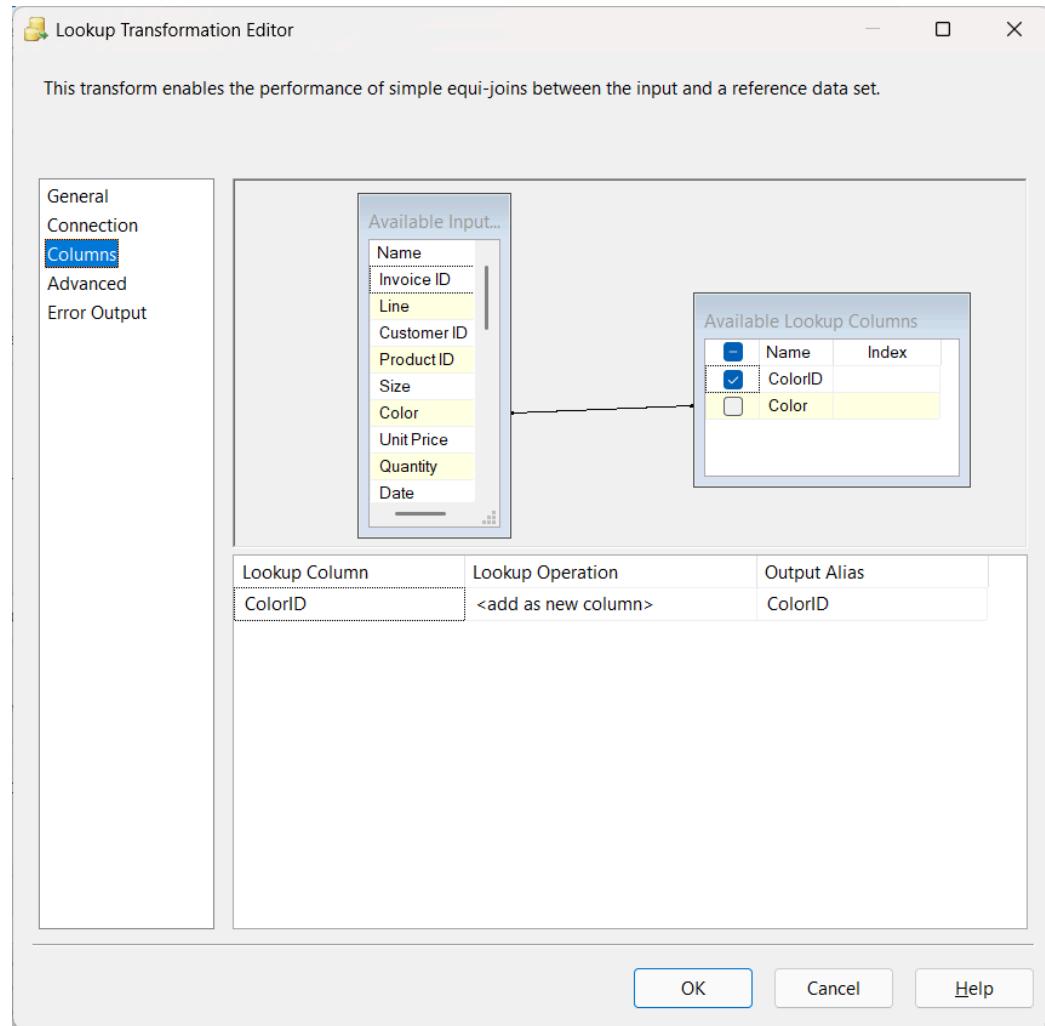
Hình 331. Tạo chức năng LookupCOLORID trong Fact\_TRANSACTIONS

- **Bước 32:** Nhấn đúp chuột vào **LookupCOLORID**, màn hình **Lookup Transformation Editor** xuất hiện. Do chúng ta đã thiết lập **OLE DB connection manager** ở bước trước, nên ta có thể sử dụng lại. Ở phần **Use a table or a view**, tiến hành chọn **[dbo].[Dim\_COLORS]**.



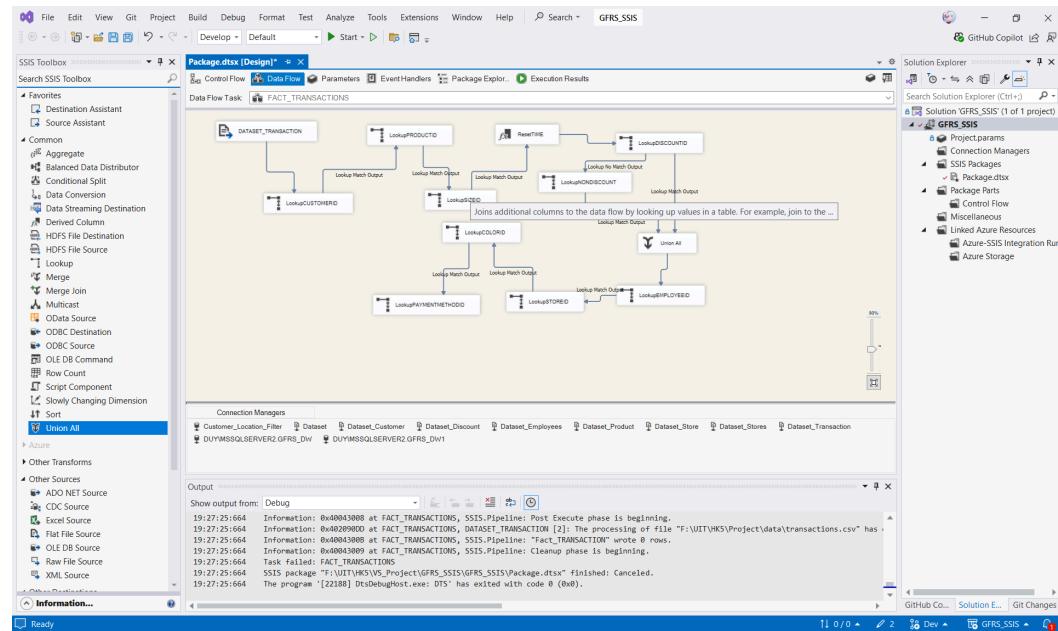
Hình 332. Kết nối nguồn dữ liệu cho chức năng LookupCOLORID

- **Bước 33:** Trong tab **Columns**, ở bảng bên phải click chọn **ColorID**. Ta kéo **Color** từ bảng **Available Input Columns** qua tương ứng **Color** bảng **Available Lookup Columns** (quá trình này giúp hệ thống xác định vùng dữ liệu cần tìm kiếm). Sau đó bấm **OK** để đóng cửa sổ **Lookup Transformation Editor**.



Hình 333. Cài đặt chức năng LookupCOLORID trong  
Fact\_TRANSACTIONS

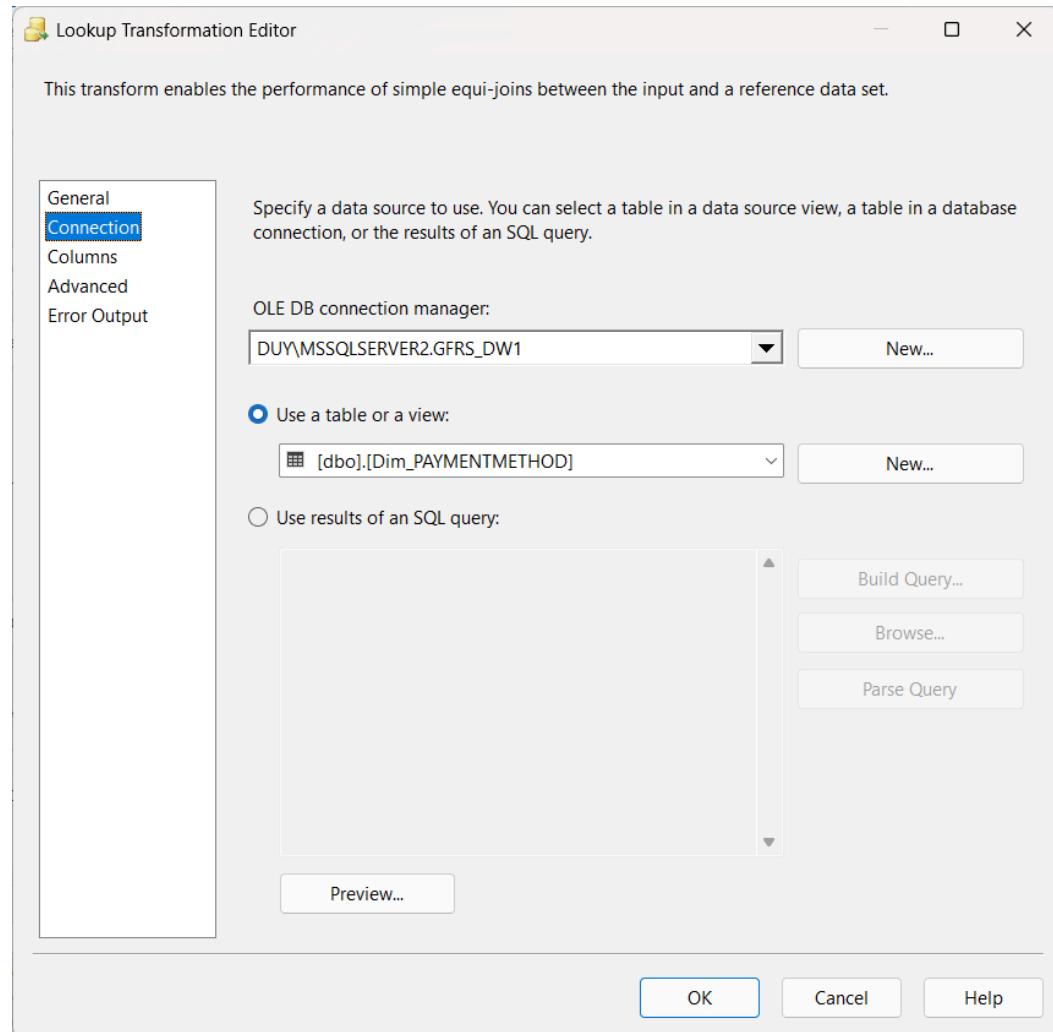
- **Bước 34:** Kéo chức năng Lookup vào vùng Data Flow và đổi tên thành **LookupPAYMENTMETHODID**. Kéo mũi tên luồng dữ liệu từ **LookupCOLORID** vào **LookupPAYMENTMETHODID**. Một hộp thoại sẽ xuất hiện, chọn **Lookup Match Output** và nhấn **OK**.



Hình 334. Tạo chức năng LookupPAYMENTMETHODID trong

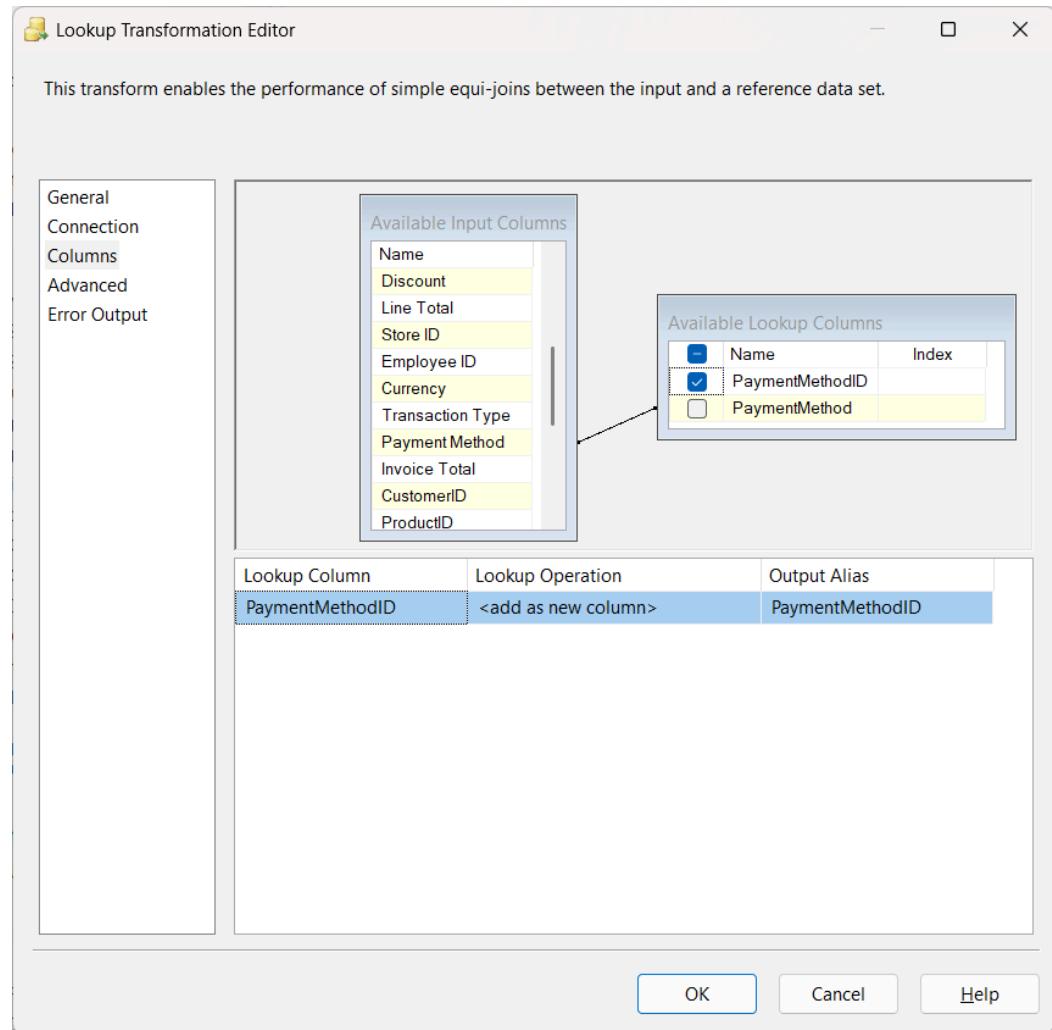
## *Fact TRANSACTIONS*

- **Bước 35:** Nhấn đúp chuột vào **LookupPAYMENTMETHODID**, màn hình **Lookup Transformation Editor** xuất hiện. Do chúng ta đã thiết lập **OLE DB connection manager** ở bước trước, nên ta có thể sử dụng lại. Ở phần **Use a table or a view**, tiến hành chọn **[dbo].[Dim\_PAYMENTMETHOD]**.



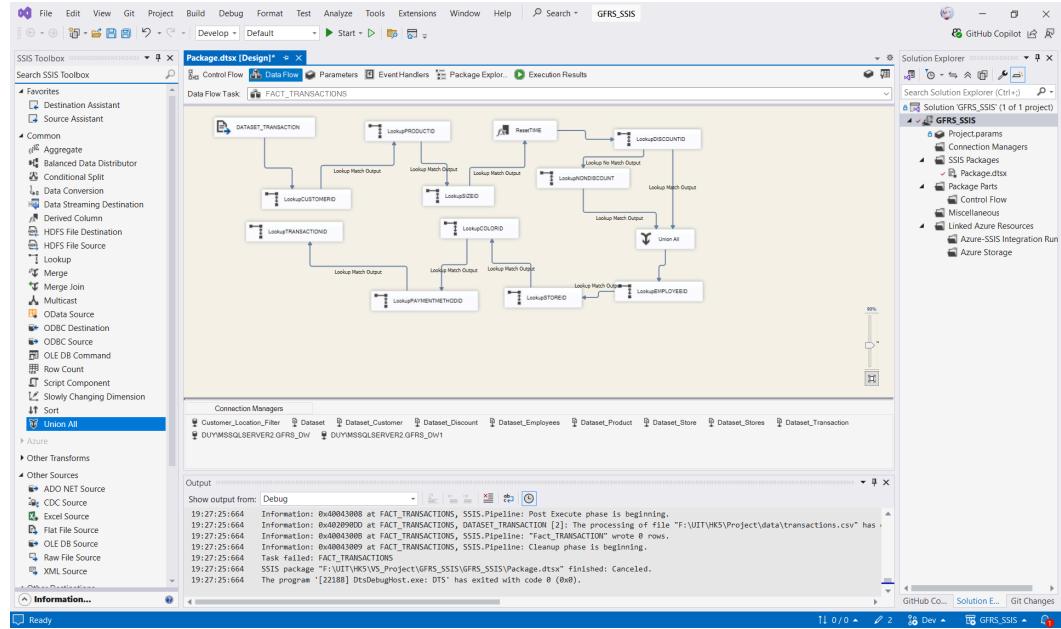
Hình 335. Kết nối nguồn dữ liệu cho chức năng  
*LookupPAYMENTMETHODID*

- **Bước 36:** Trong tab **Columns**, ở bảng bên phải click chọn **PaymentMethodID**. Ta kéo **Payment Method** từ bảng **Available Input Columns** qua tương ứng **PaymentMethod** bảng **Available Lookup Columns** (quá trình này giúp hệ thống xác định vùng dữ liệu cần tìm kiếm). Sau đó bấm **OK** để đóng cửa sổ **Lookup Transformation Editor**.



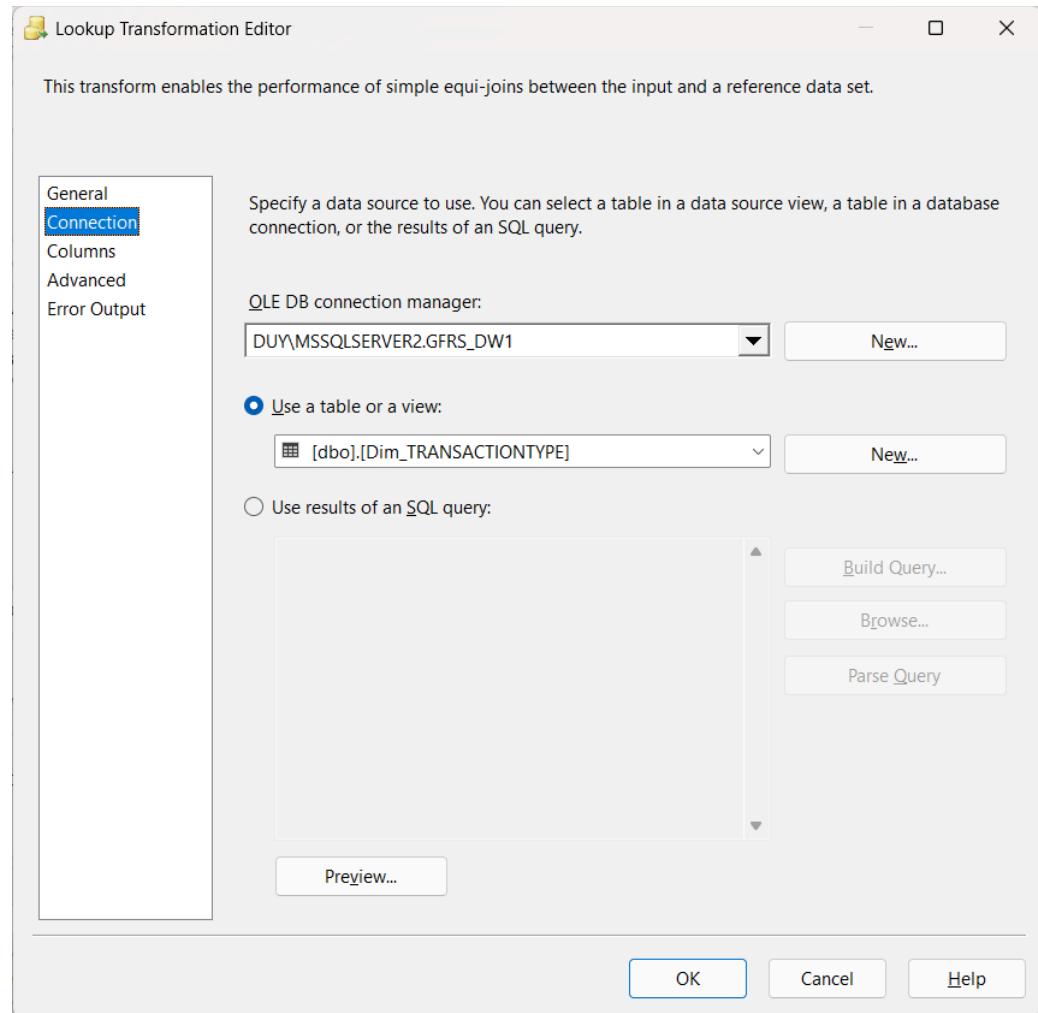
*Hình 336. Cài đặt chức năng LookupPAYMENTMETHODID trong Fact\_TRANSACTION*

- **Bước 37:** Kéo chức năng **Lookup** vào vùng **Data Flow** và đổi tên thành **LookupTRANSACTIONID** (lấy các mã loại mua sắm phù hợp). Kéo mũi tên luồng dữ liệu từ **LookupPAYMENTMETHODID** vào **LookupTRANSACTIONID**. Một hộp thoại sẽ xuất hiện, chọn **Lookup Match Output** và nhấn **OK**.



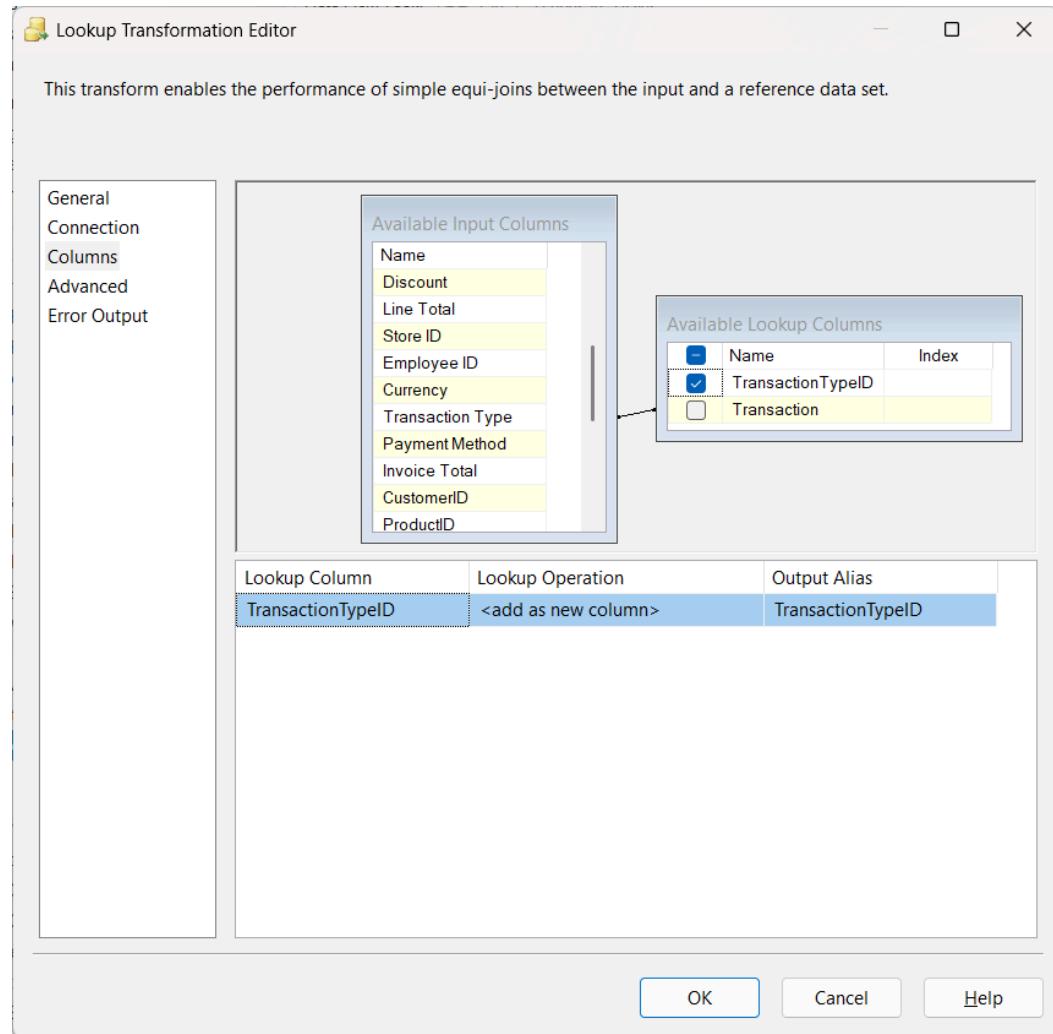
Hình 337. Tạo chức năng **LookupTRANSACTIONID** trong  
**Fact\_TRANSACTIONS**

- **Bước 38:** Nhấn đúp chuột vào **LookupTRANSACTIONID**, màn hình **Lookup Transformation Editor** xuất hiện. Do chúng ta đã thiết lập **OLE DB connection manager** ở bước trước, nên ta có thể sử dụng lại. Ở phần **Use a table or a view**, tiến hành chọn **[dbo].[Dim\_TRANSACTIONTYPE]**.



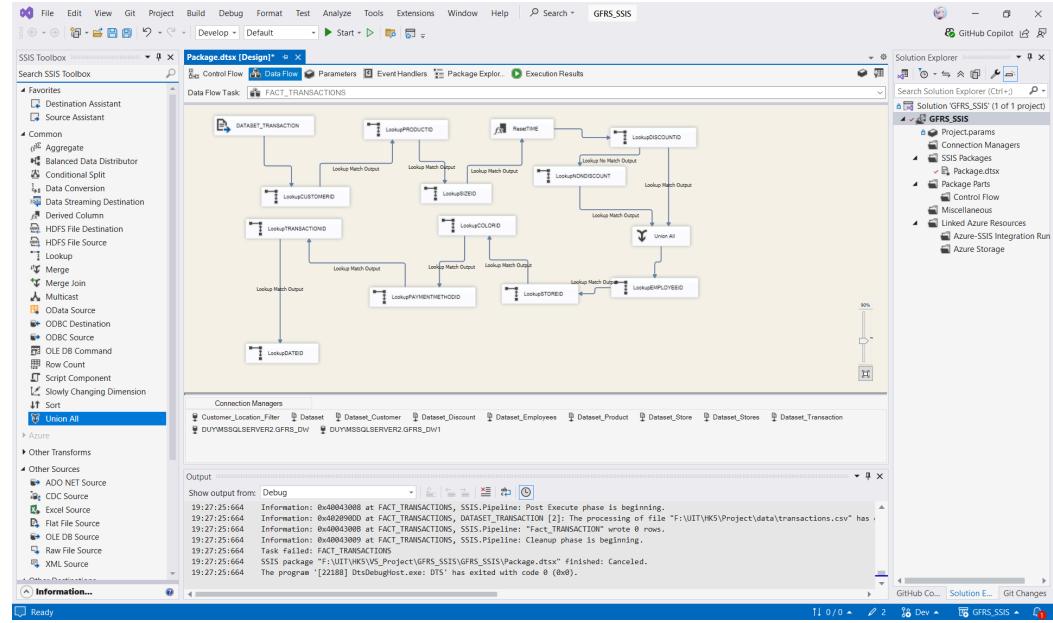
Hình 338. Kết nối nguồn dữ liệu cho chức năng LookupTRANSACTIONID

- **Bước 39:** Trong tab **Columns**, ở bảng bên phải click chọn **TransactionTypeID**. Ta kéo **Transaction Type** từ bảng **Available Input Columns** qua tương ứng **Transaction** bảng **Available Lookup Columns** (quá trình này giúp hệ thống xác định vùng dữ liệu cần tìm kiếm). Sau đó bấm **OK** để đóng cửa sổ **Lookup Transformation Editor**.



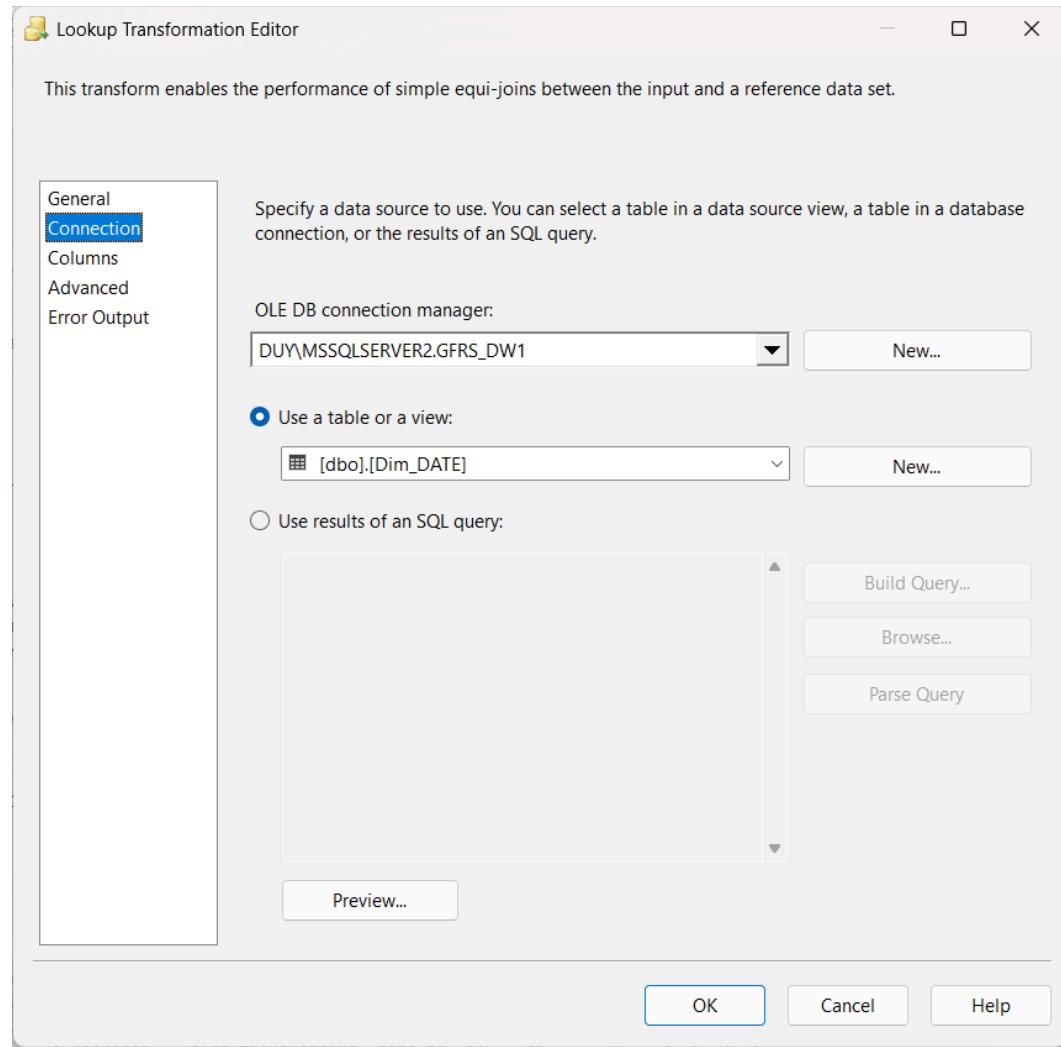
Hình 339. Cài đặt chức năng **LookupTRANSACTIONID** trong  
*Fact\_TRANSACTIONS*

- **Bước 40:** Kéo chức năng **Lookup** vào vùng **Data Flow** và đổi tên thành **LookupDATEID**. Kéo mũi tên luồng dữ liệu từ **LookupTRANSACTIONID** vào **LookupDATEID**. Một hộp thoại sẽ xuất hiện, chọn **Lookup Match Output** và nhấn **OK**.



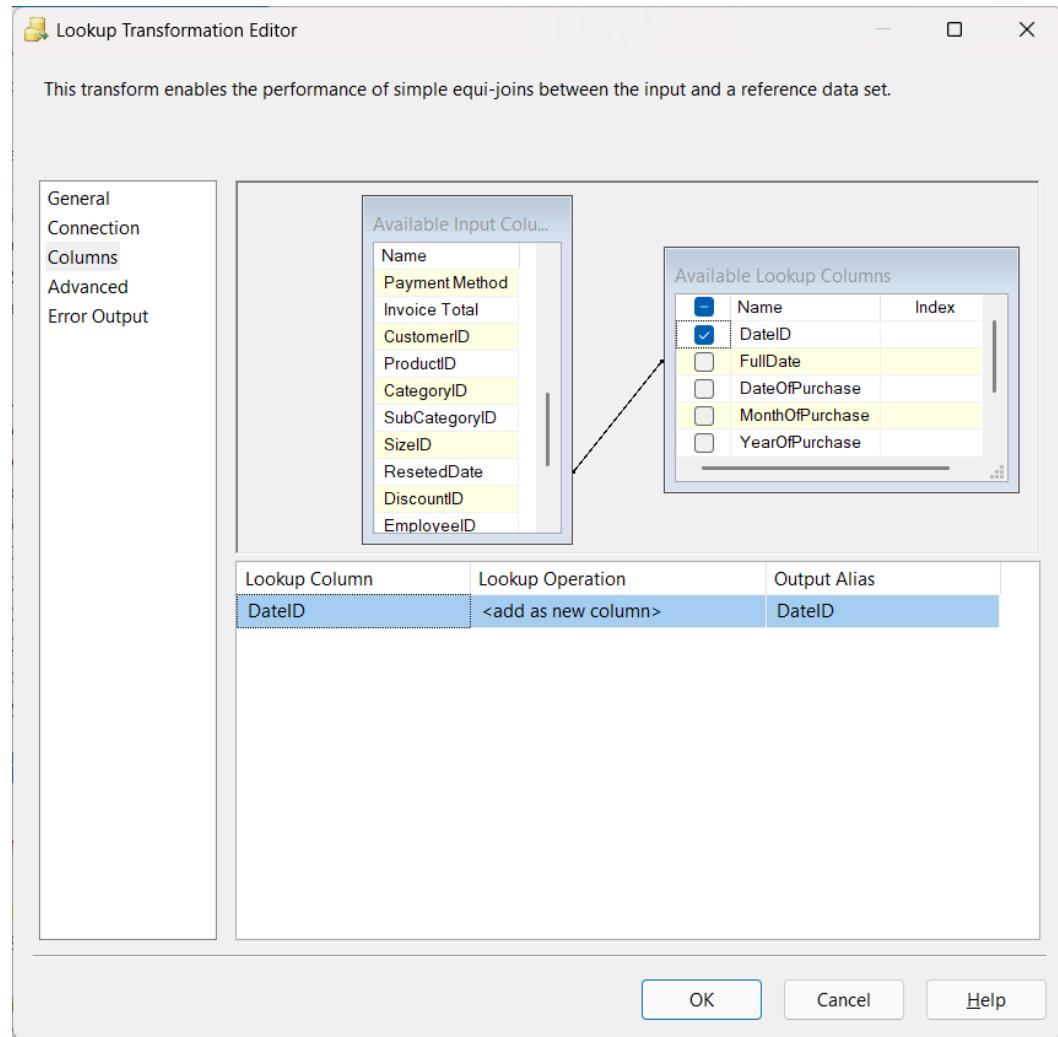
Hình 340. Tạo chinc năng **LookupDATEID** trong **Fact\_TRANSACTIONS**

- **Bước 41:** Nhán đúp chuột vào **LookupDATEID**, màn hình **Lookup Transformation Editor** xuất hiện. Do chúng ta đã thiết lập **OLE DB connection manager** ở bước trước, nên ta có thể sử dụng lại. Ở phần **Use a table or a view**, tiến hành chọn **[dbo].[Dim\_DATE]**.



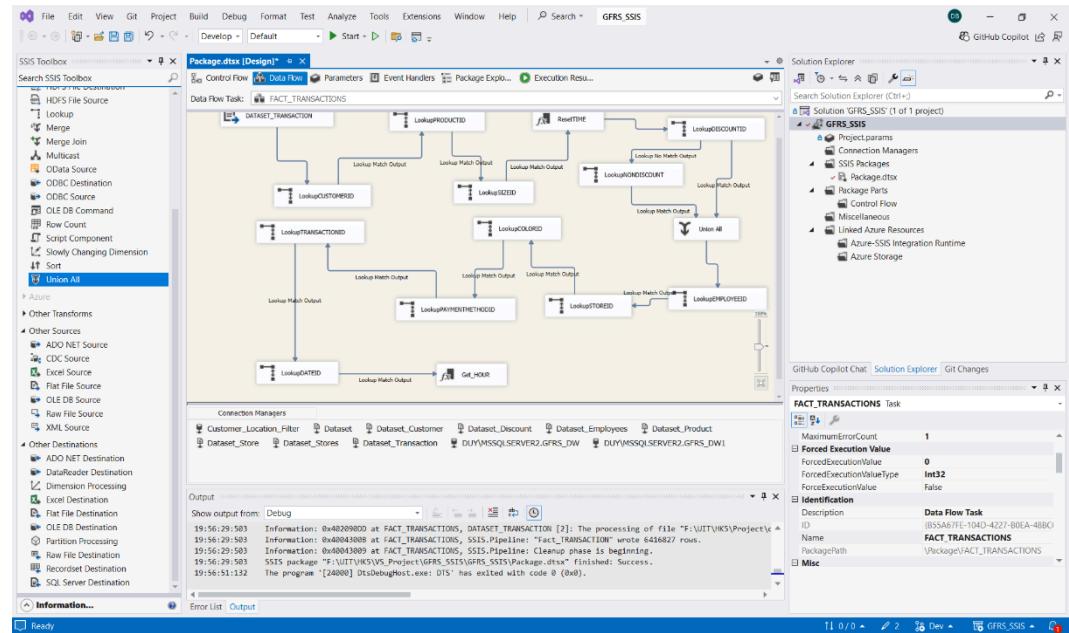
Hình 341. Kết nối nguồn dữ liệu cho chức năng LookupDATEID

- **Bước 42:** Trong tab **Columns**, ở bảng bên phải click chọn **DateID**. Ta kéo **ResetedDate** từ bảng **Available Input Columns** qua tương ứng **Full Date** bảng **Available Lookup Columns** (quá trình này giúp hệ thống xác định vùng dữ liệu cần tìm kiếm). Sau đó bấm **OK** để đóng cửa sổ **Lookup Transformation Editor**.



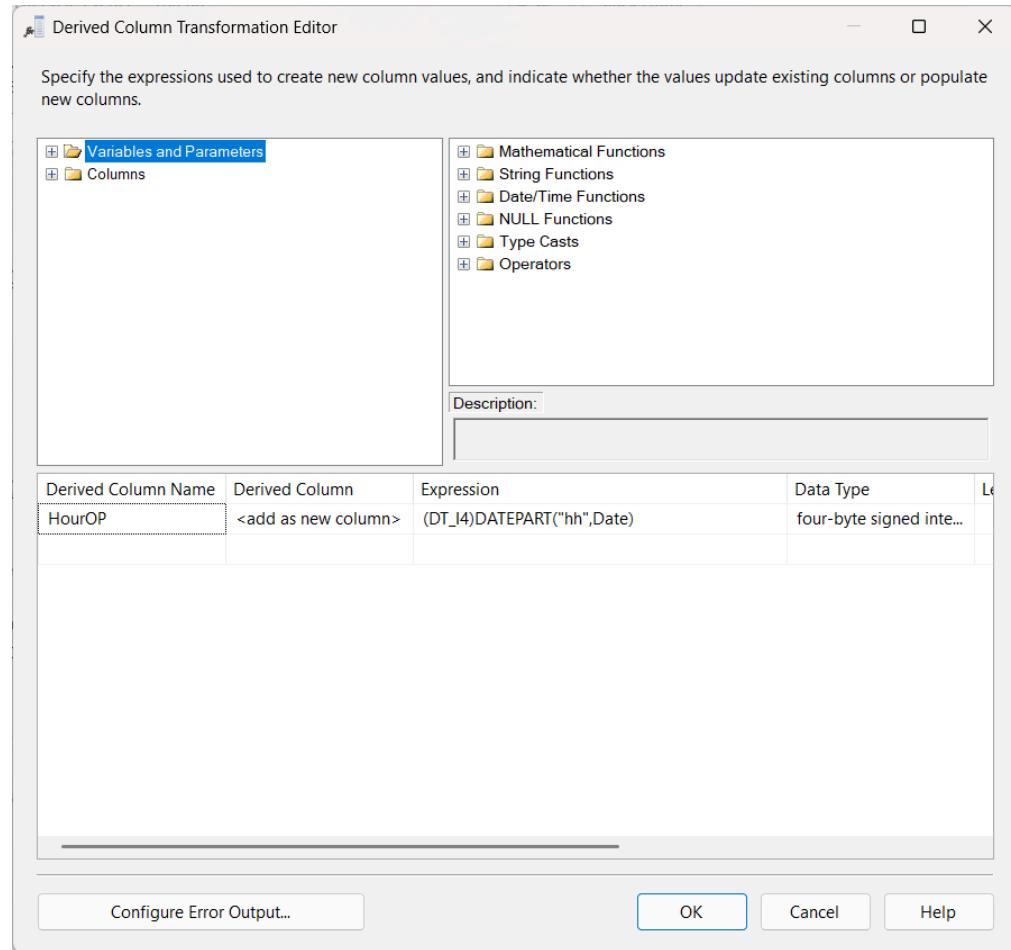
Hình 342. Cài đặt chức năng *LookupDATEID* trong *Fact\_TRANSACTIONS*

- **Bước 43:** Kéo **Derived Column** từ **SSIS Toolbox** vào vùng **Data Flow** sau đó đổi tên lại thành **Get\_HOUR**. Kéo luồng dữ liệu từ **LookupDATEID** vào **Get\_HOUR**



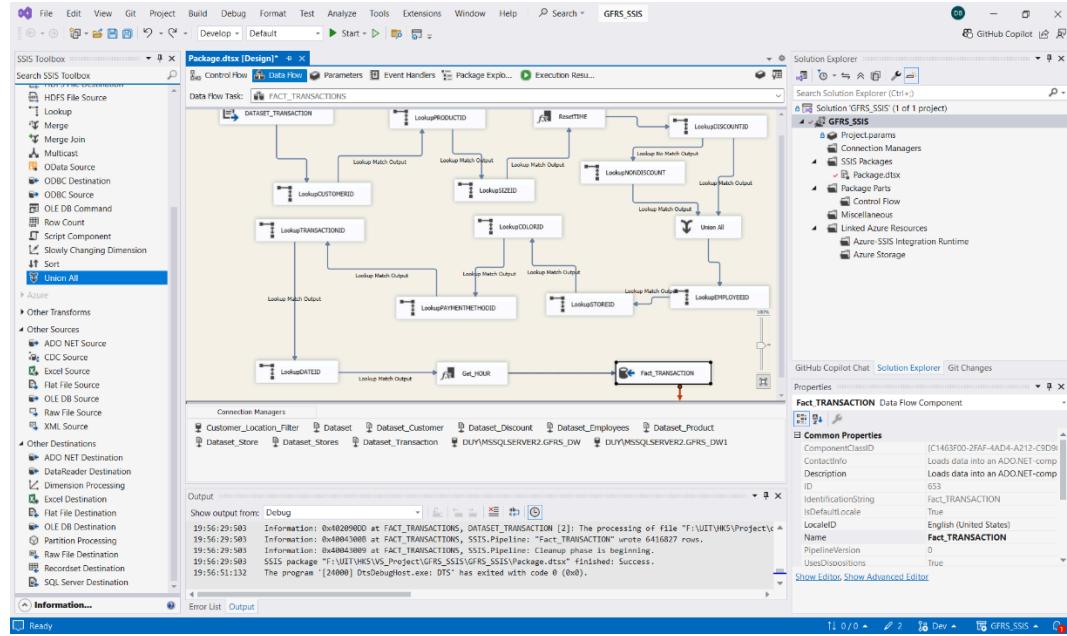
Hình 343. Tạo chức năng Get\_HOUR trong Fact\_TRANSACTIONS

- **Bước 44:** Khi này cửa sổ **Derived Column Transformation Editor** sẽ hiện lên, ta tạo một cột mới **HourOP** với **Expression** là **(DT\_I4)DATEPART("hh", Date)**. Sau đó chọn **OK**.



Hình 344. Tùy chỉnh Get\_HOUR trong Fact\_TRANSACTIONS

- **Bước 45:** Kéo **ADO NET Destination** từ **SSIS Toolbox** vào **vùng Data Flow** và đổi tên thành **Fact\_TRANSACTIONS**. Kéo luồng dữ liệu từ **Get\_HOUR** vào **Fact\_TRANSACTIONS**.



Hình 345. Tạo chức năng ADO.NET Destination Fact\_Transactions

- Bước 46: Vào SQL Server 2022, chạy lệnh DDL để tạo Fact\_Transactions.

```

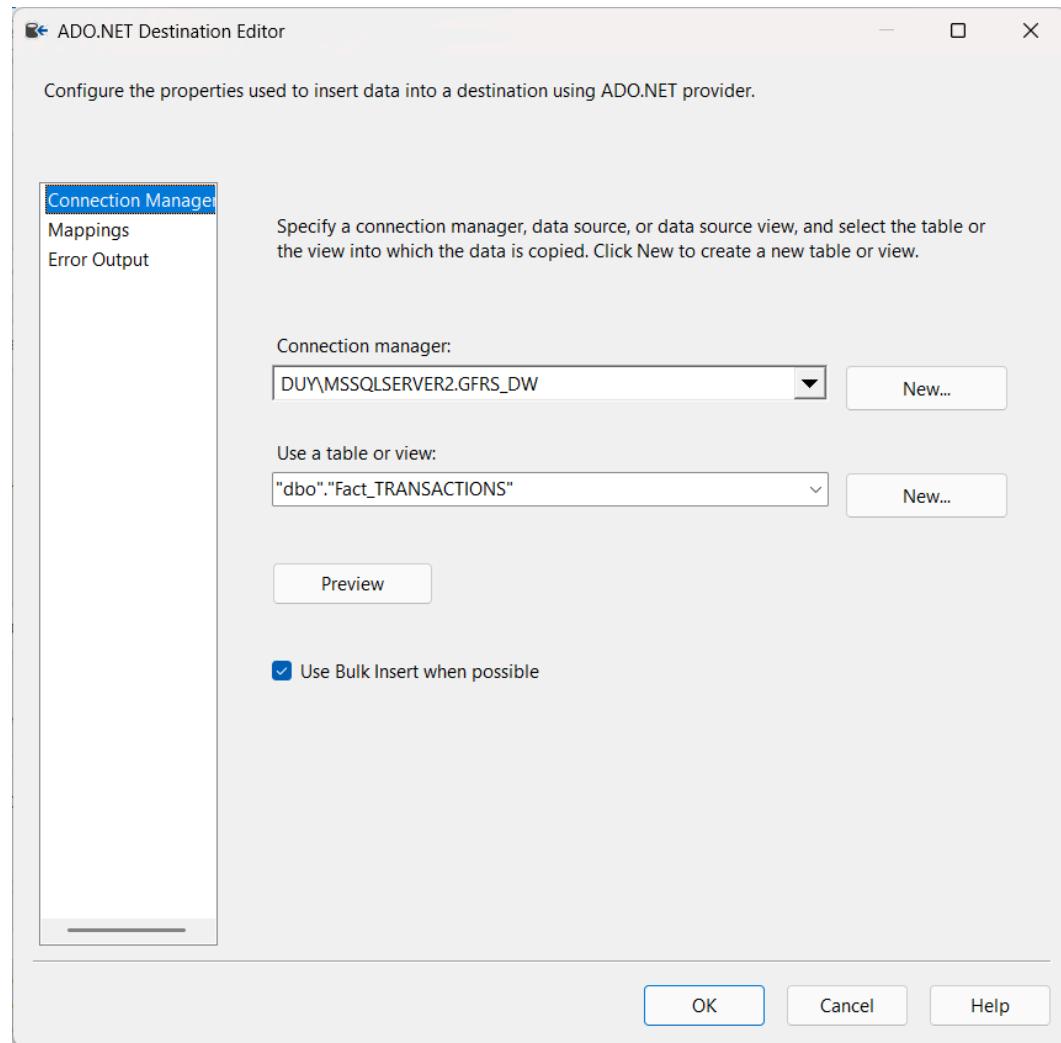
CREATE TABLE [Fact_Transactions] (
    [InvoiceID] BIGINT IDENTITY(1,1) PRIMARY KEY,
    [InvoiceHash] VARCHAR(50),
    [Line] INT,
    [Quantity] INT,
    [UnitPrice] INT,
    [Currency] VARCHAR(255),
    [LineTotal] DECIMAL(18, 2),
    [InvoiceTotal] DECIMAL(18, 2),
    [HourOfPurchase] INT,
    [CustomerID] BIGINT,
    [EmployeeID] INT,
    [StoreID] INT,
    [DiscountID] INT,
    [DateID] INT,
    [ColorID] INT,
    [SizeID] INT,
    [TransactionTypeID] INT,
    [PaymentMethodID] INT,
    [ProductID] INT
)
GO
-- Add Foreign Key constraints
ALTER TABLE Dim_Customer
ADD CONSTRAINT FK_Customer_Location FOREIGN KEY (LocationID) REFERENCES Dim_Location(Location_id);

```

Hình 346. Chạy lệnh DDL tạo Fact\_Transactions trong kho dữ liệu

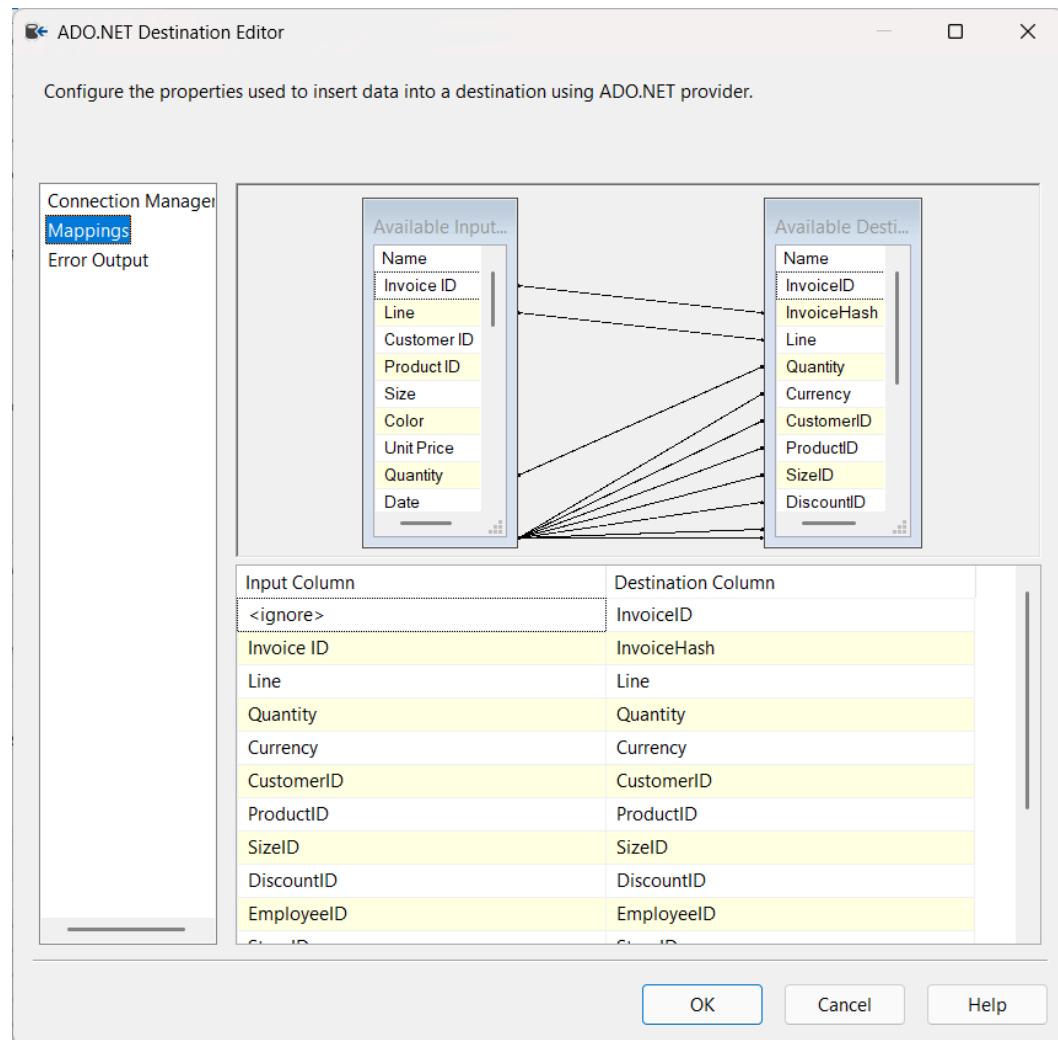
- Bước 47: Nhấn đúp chuột vào Fact\_Transactions. Trong tab Connection Manager của màn hình ADO.NET Destination Editor, ở Use

a table or view tiến hành nhấn chọn “**dbo**”.”**Fact\_TRANSACTIONS**”, cũng chính là bảng vừa được tạo bằng lệnh **DDL** ở **SQL Server 2022**



Hình 347. Kết nối đích đến cho chức năng ADO NET Destination

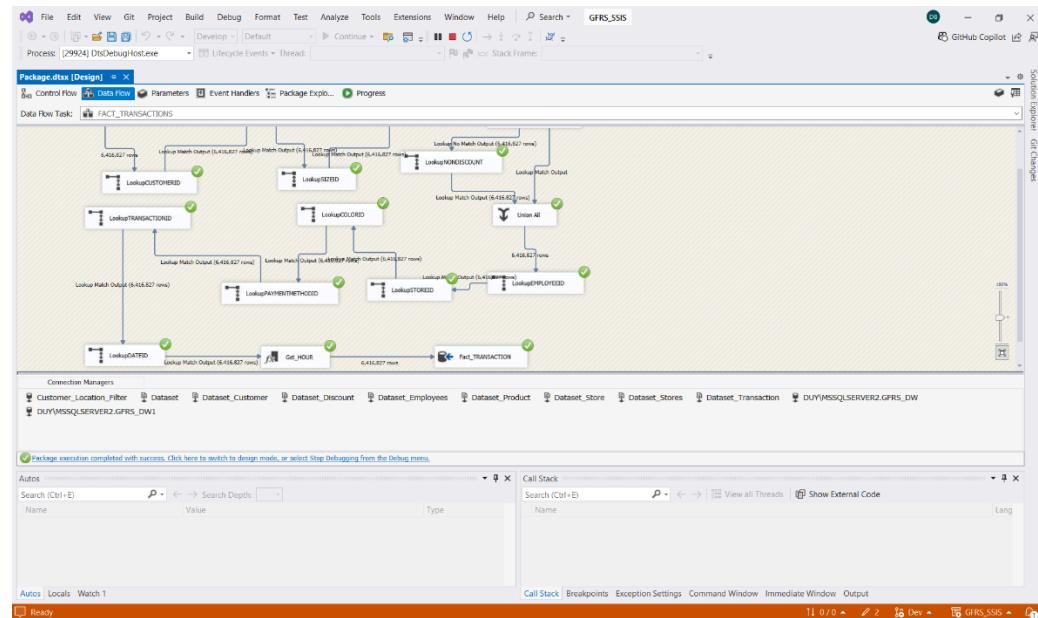
- **Bước 48:** Ở tab **Mapping** của màn hình **ADO.NET Destination Editor** tiến hành chọn các thuộc tính thích hợp ở cột bên trái để ánh xạ với các thuộc tính ở cột bên phải.



Hình 348. Cài đặt chức năng ADO NET Destination

#### Fact\_TRANSACTIONS

- **Bước 49:** Nhấn Start để kiểm tra quá trình tạo và đổ dữ liệu vào **Fact\_Transactions** có thành công hay chưa.



Hình 349. Đỗ dữ liệu vào Fact\_TRANSACTION

Kết quả hiển thị cho thấy quá trình diễn ra thành công và có **6,416,827 rows** được đỗ vào Fact\_TRANSACTION.

- Bước 50:** Vào SQL Server 2022 kiểm tra và thấy dữ liệu đã được đỗ vào Fact\_TRANSACTION thành công.

```

ALTER TABLE Fact_TRANSACTIONS
ADD CONSTRAINT FK_Transaction_Payment FOREIGN KEY (PaymentMethodID) REFERENCES Dim_PAYMENTMETHOD(PaymentMethodID);
ALTER TABLE Fact_TRANSACTIONS
ADD CONSTRAINT FK_Transaction_Type FOREIGN KEY (TransactionTypeID) REFERENCES Dim_TRANSACTIONTYPE(TransactionTypeID);

TRUNCATE TABLE Fact_TRANSACTIONS;
SELECT * FROM Dim_DATE;
SELECT * FROM Fact_TRANSACTIONS
  
```

InvoiceID	Line	Quantity	UnitPrice	Currency	LineTotal	InvoiceTotal	CustomerID	ProductID	SizeID	DiscountID	EmployeeID	ShipID	DateID	ColorID	PaymentMethodID	TransactionID
1		1	80	USD	80	15	1365497	1728	12	7	1	809	1	1	2	
2	2	1	32	USD	19	128	15	1365497	9923	9	102	7	1	809	1	1
3	3	1	46	USD	27	128	15	1365497	231	12	102	7	1	809	1	2
4	4	1	70	USD	70	20	1415711	2545	13	102	6	1	809	4	1	
5	5	1	26	USD	26	77	20	1415711	2545	11	1	6	1	809	15	2
6	6	1	9	USD	9	77	20	1415711	2293	1	6	1	809	1	1	
7	7	1	96	USD	96	95	20	1380440	4977	12	1	6	1	809	1	2
8	8	1	75	USD	75	9	20	1403233	949	12	102	9	1	809	12	2
9	9	1	46	USD	27	72	8	1403233	9623	12	102	9	1	809	15	2
10	10	1	32	USD	19	19	17	1361968	6499	12	102	12	1	809	1	2

Hình 350. Dữ liệu được đỗ thành công vào Fact\_TRANSACTION

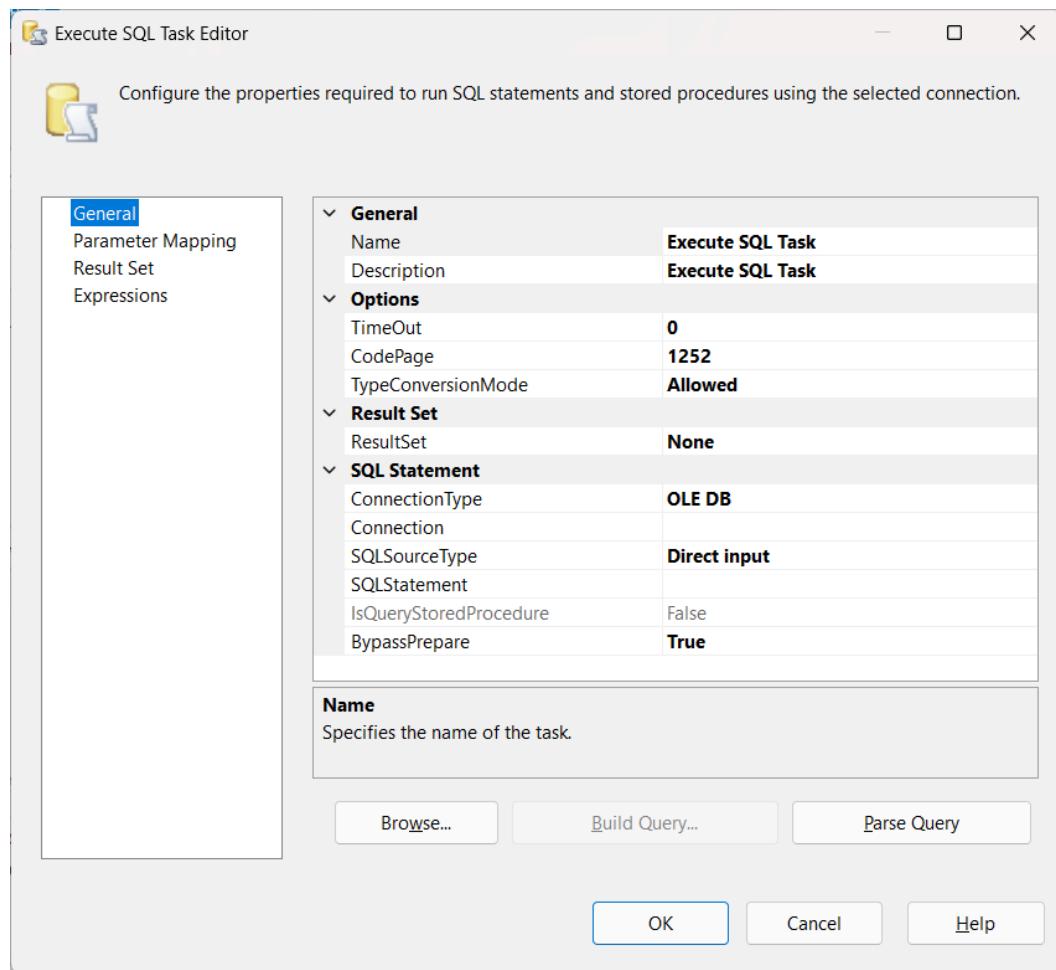
## II.4. Quá trình chạy toàn bộ SSIS

Để tối ưu quá trình SSIS được diễn ra nhanh chóng hơn, ta có thể thực hiện như sau:

### II.4.1. Tạo Execute SQL Task

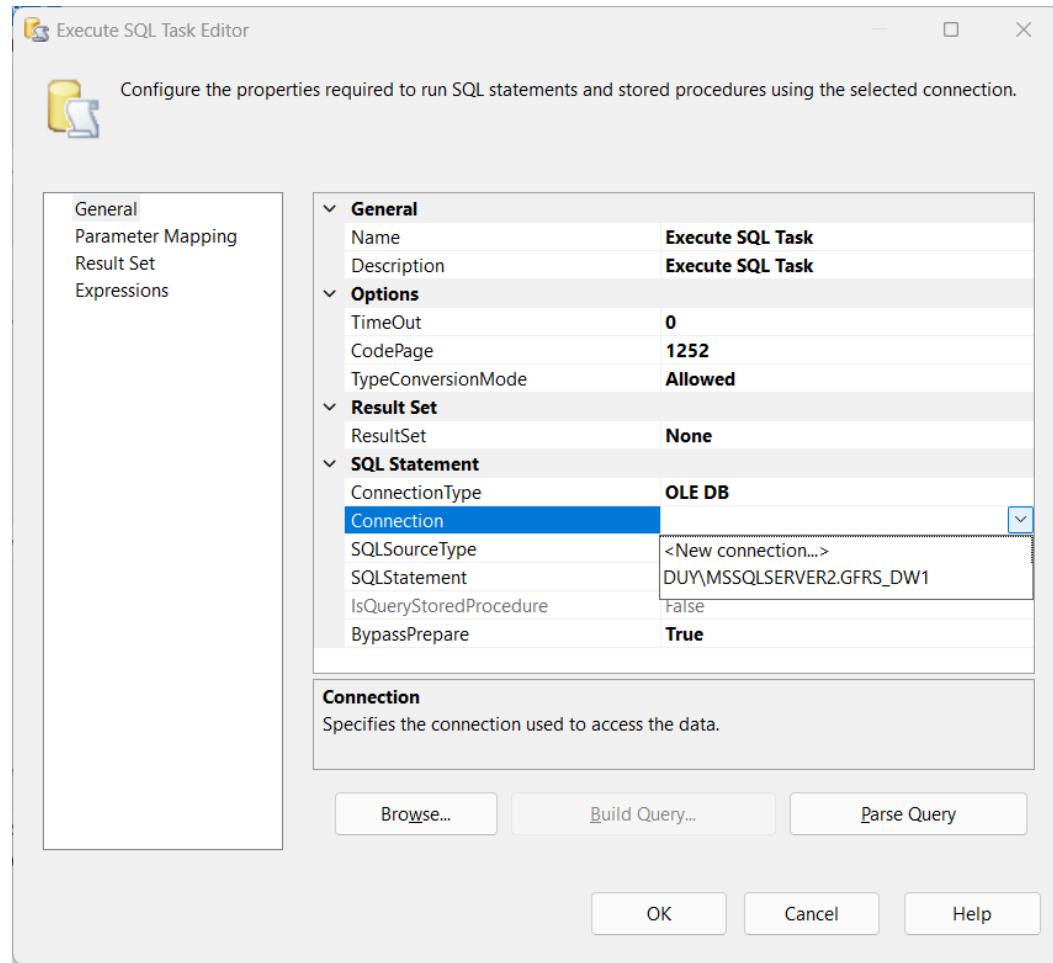
Chức năng này có nhiệm vụ tạo kho dữ liệu mà không cần thực hiện thủ công bằng công cụ SQL Server Management Studio (SSMS). Để tạo, ta thực hiện như sau:

- **Bước 1:** Ta kéo chức năng **Execute SQL Task** từ **SSIS Toolbox** và đổi tên lại thành **Execute DATAWAREHOUSE**. Nhấn đúp chuột để mở, cửa sổ **Execute SQL Task Editor** sẽ hiện lên



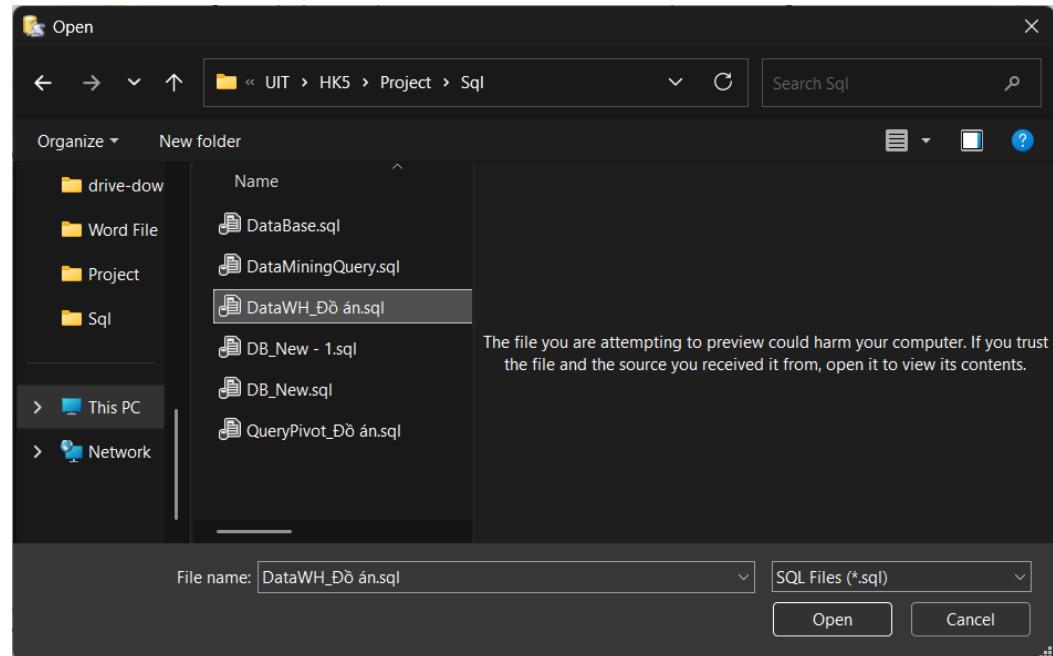
Hình 351. Cửa sổ Execute SQL Task Editor

- **Bước 2:** Ở mục **Connection**, ta có thể chọn <New connection...> để tạo một kết nối đến Kho dữ liệu. Hoặc sử dụng một kết nối có sẵn (nếu làm phần tạo bảng DIM và FACT trước)

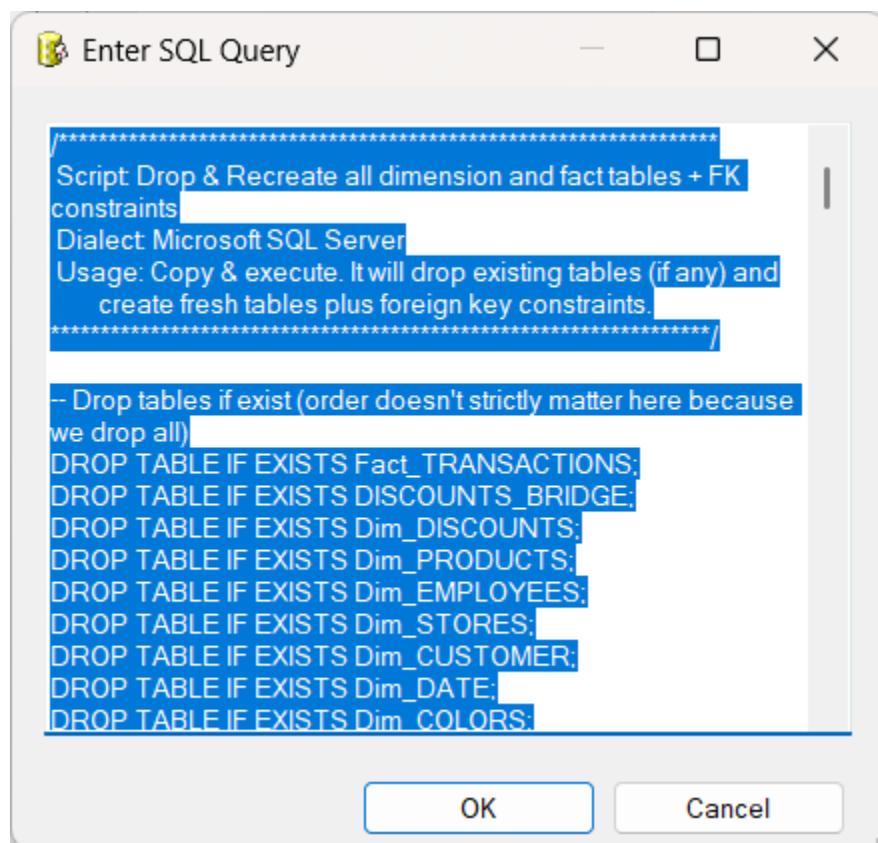


Hình 352. Tạo kết nối đến kho dữ liệu

- **Bước 3:** Ta có thể chọn **Browse** để chọn tệp SQL hoặc nhấn (...) bên cạnh **SQLStatement** để nhập code SQL tạo bảng cho kho dữ liệu.



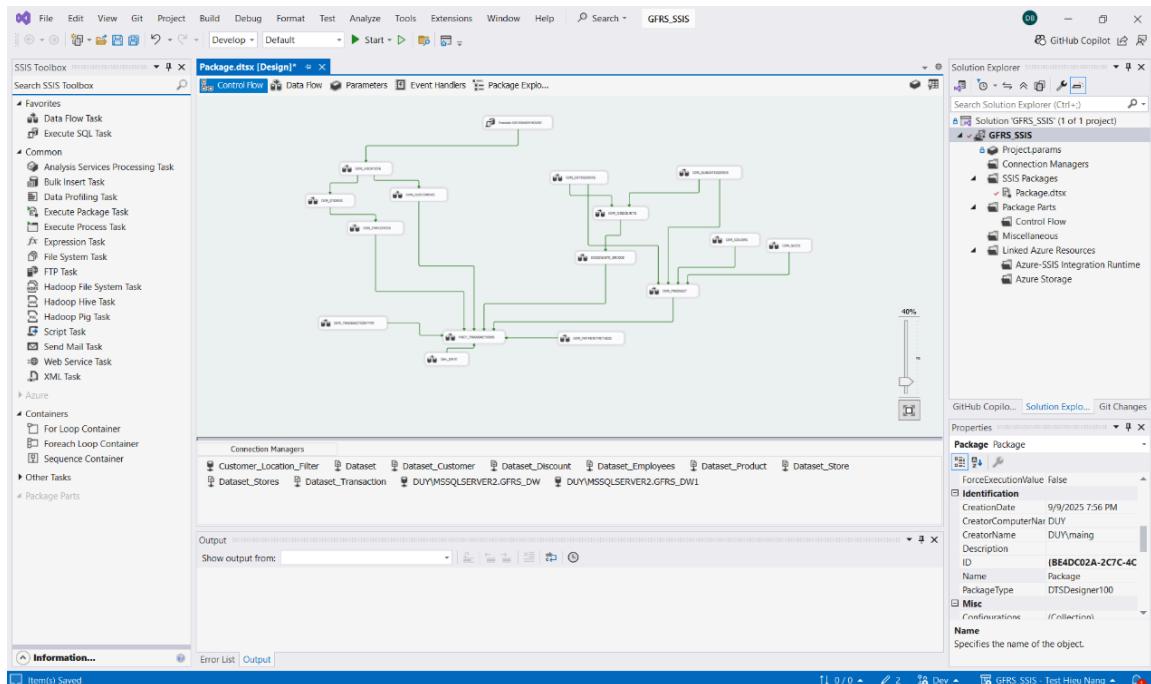
Hình 353. Chọn file SQL tạo bảng cho kho dữ liệu



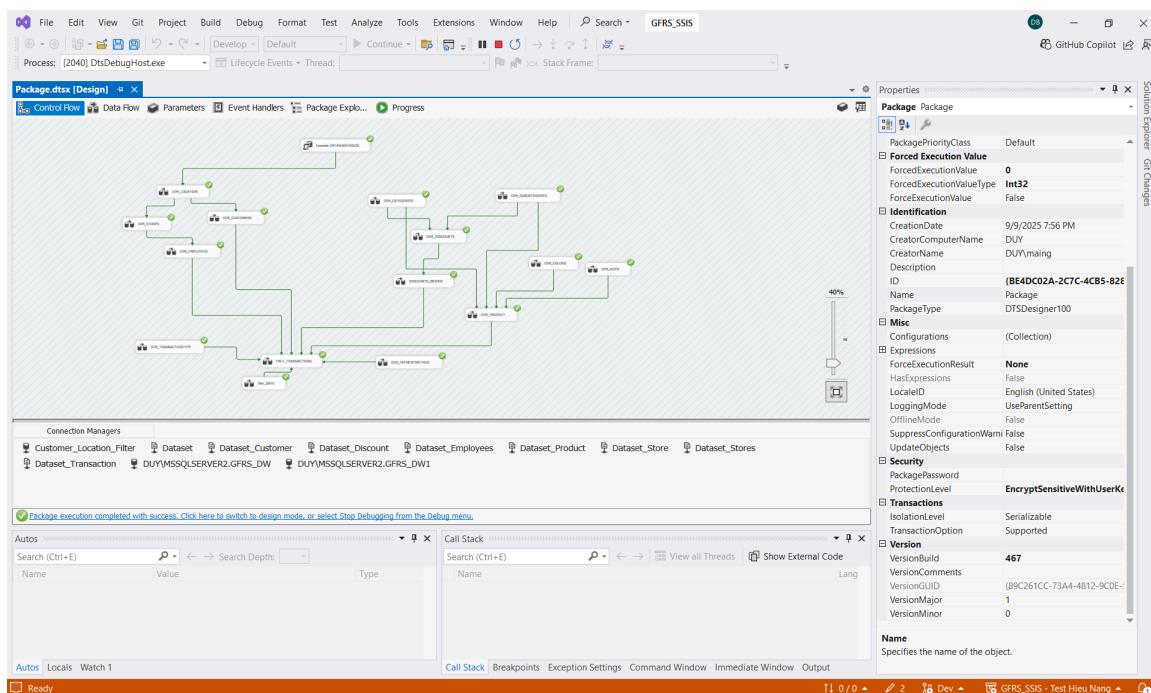
Hình 354. Thêm code SQL bằng SQLStatement

- **Bước 4:** Sau đó chọn OK và kết thúc.

## II.4.2. Chạy quá trình SSIS



Hình 355. Trình tự thực hiện quá trình SSIS

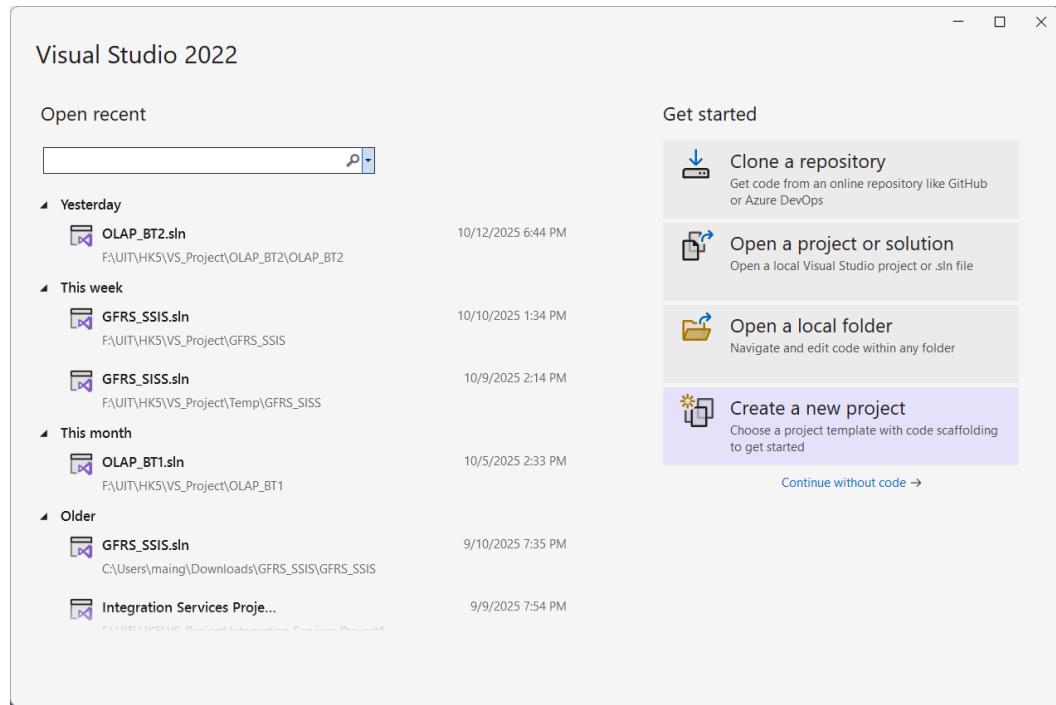


Hình 356. Quá trình chạy SSIS đã thành công

## CHƯƠNG 3. KHAI THÁC DỮ LIỆU (SSAS)

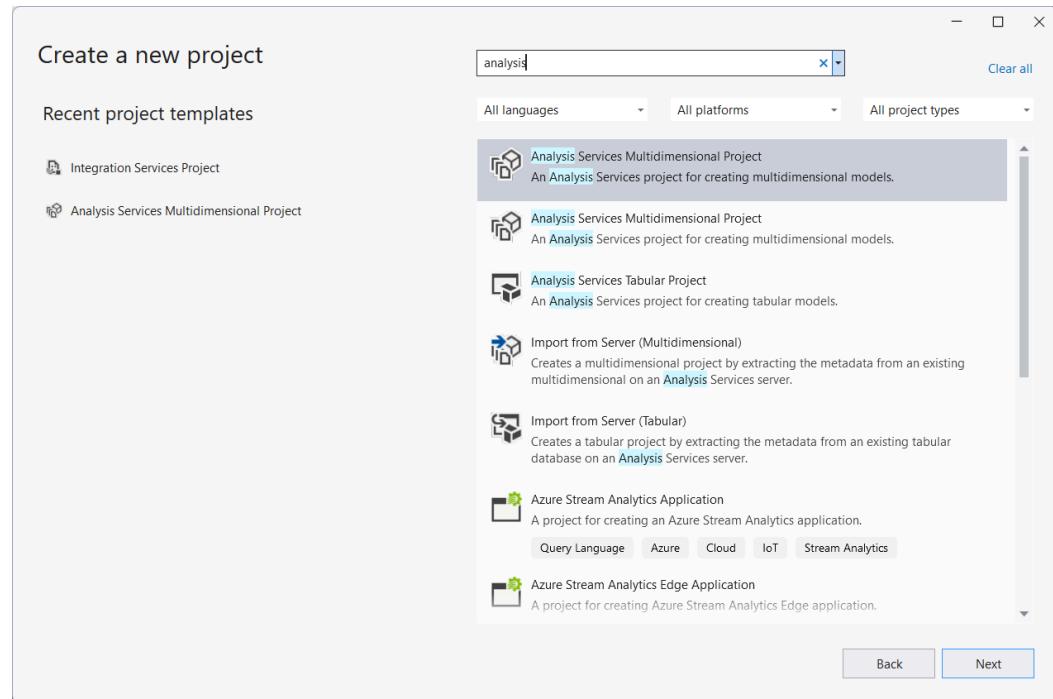
### III.1. Tạo project SSAS trong Visual Studio 2022

- **Bước 1:** Mở phần mềm **Visual Studio 2022**. Sau đó chọn **Create a new project**



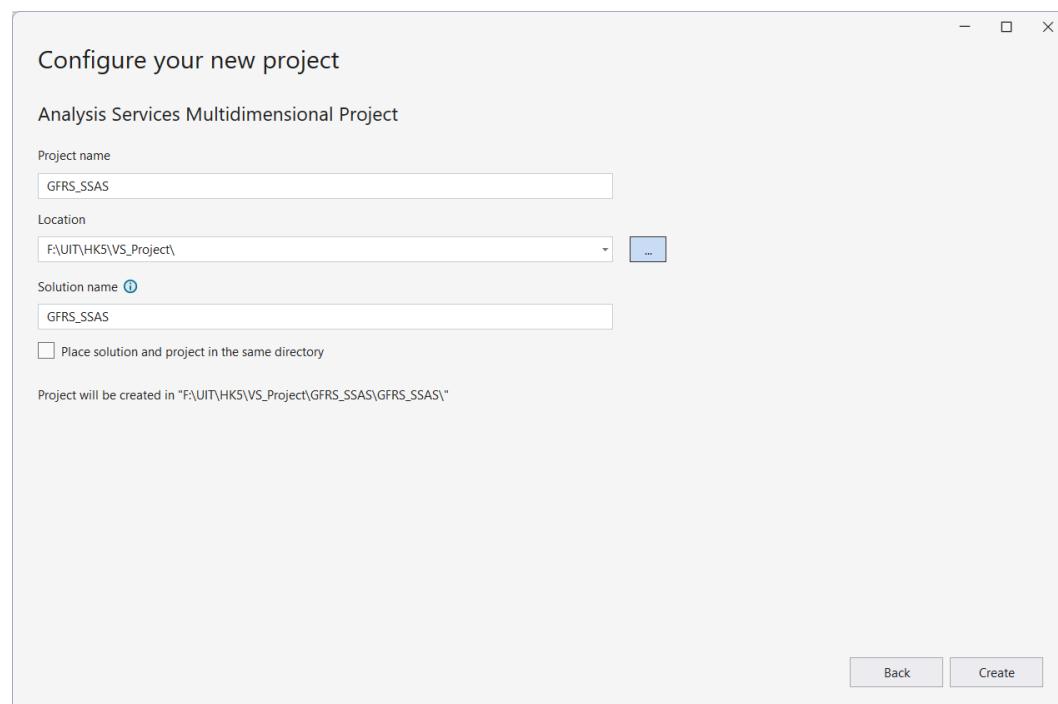
Hình 357. Tạo project SSAS mới trong Visual Studio 2022

- **Bước 2:** Chọn **Analysis Services Multidimensional Project**. Sau đó chọn **Next**.



Hình 358. Chọn Analysis Services Multidimensional Project

- **Bước 3:** Đặt tên và chọn nơi lưu trữ cho project SSAS. Sau đó chọn **Create**



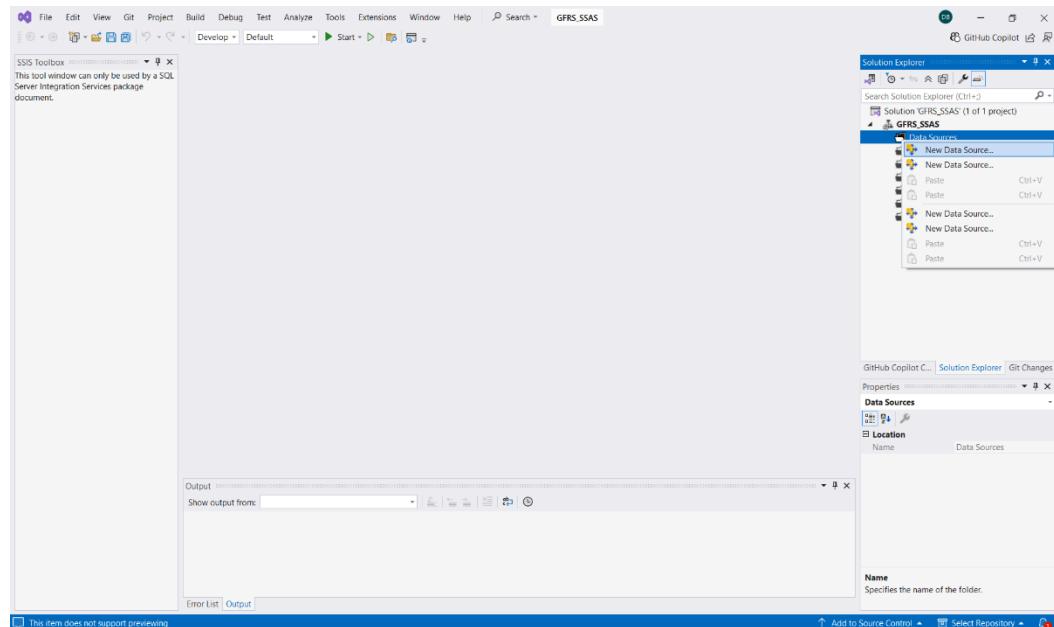
Hình 359. Đặt tên và tạo project SSAS

## III.2. Tiến hành thực hiện quá trình SSAS

### III.2.1. Kết nối với kho dữ liệu

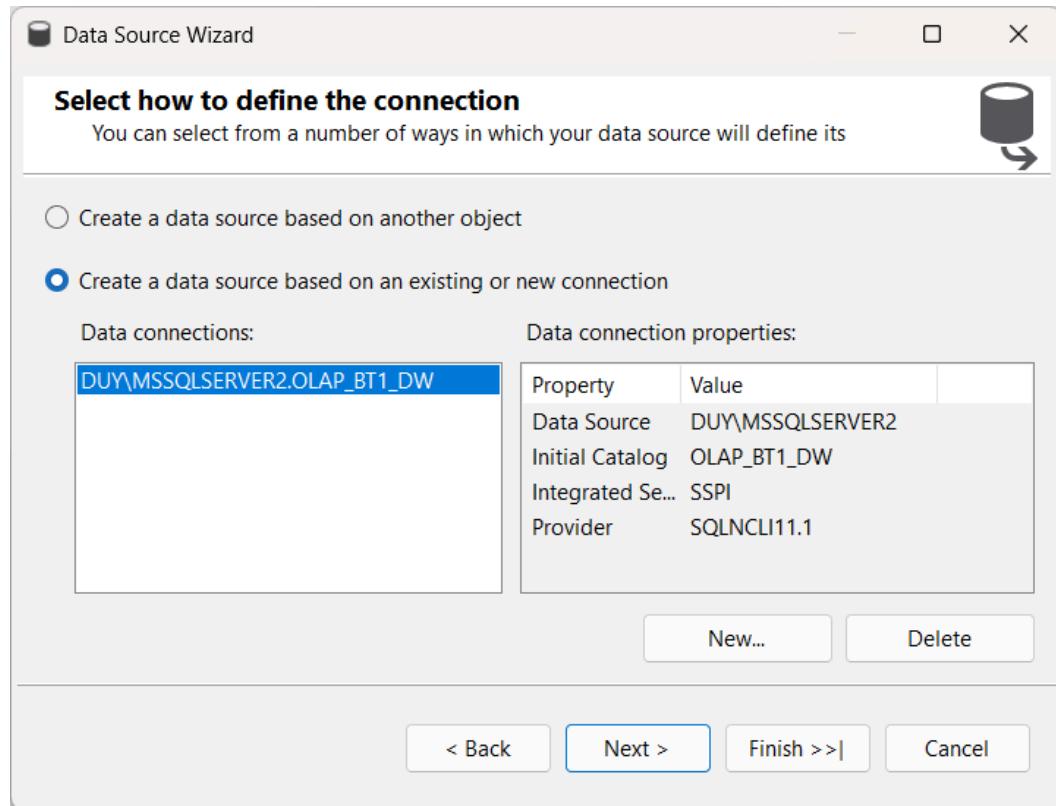
#### III.2.1.1. Tạo New Data Source

- **Bước 1:** Ở phần Solution Explorer, ở mục Data Sources của dự án **GFRS\_SSAS**, ta nhấn chuột phải và chọn **New Data Source..**



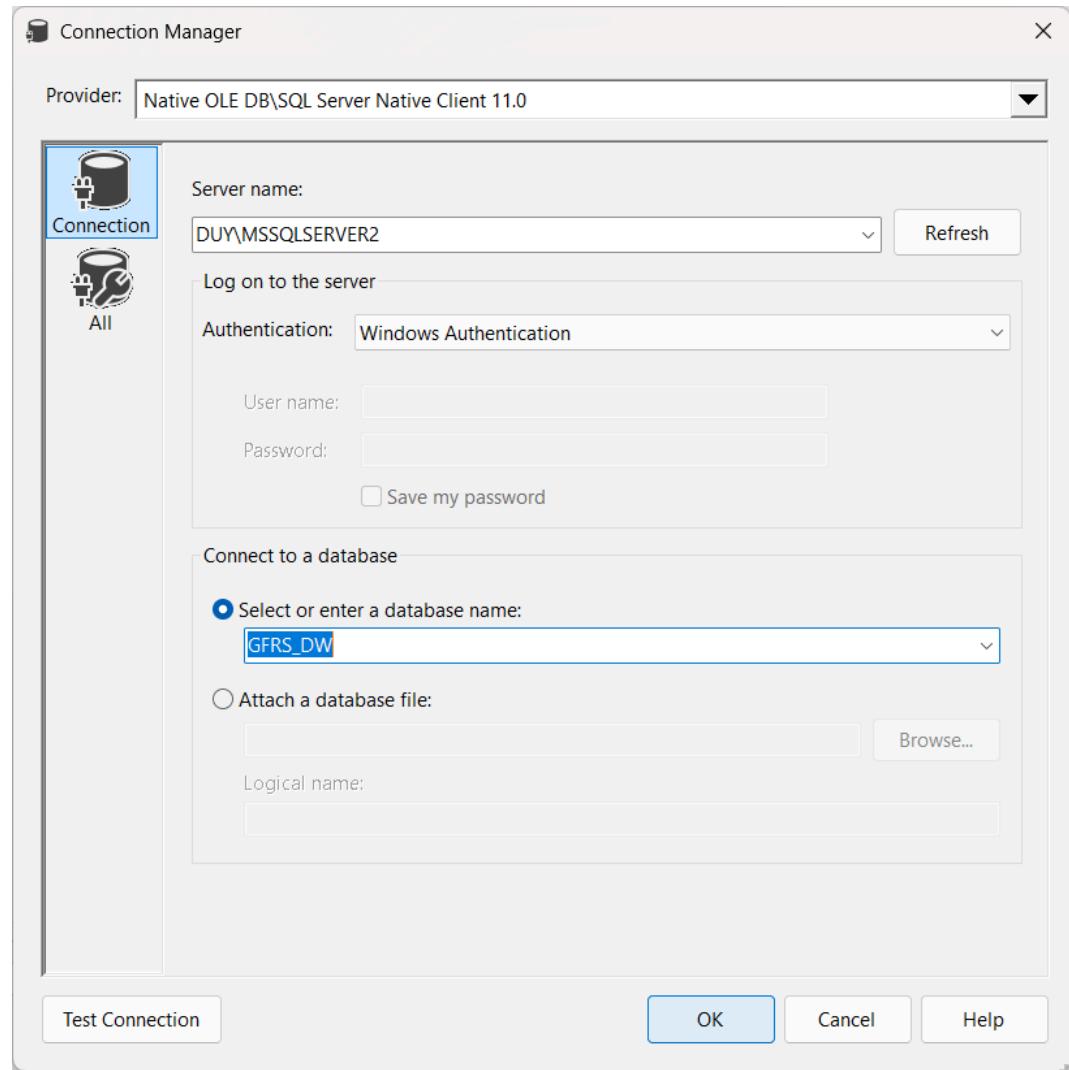
Hình 360. Tạo New Data Source mới

- **Bước 2:** Cửa sổ **Data Source Wizard** sẽ hiện lên, ta chọn **Create a data source based on an existing or new connection**, sau đó chọn **New** để tạo kết nối mới.



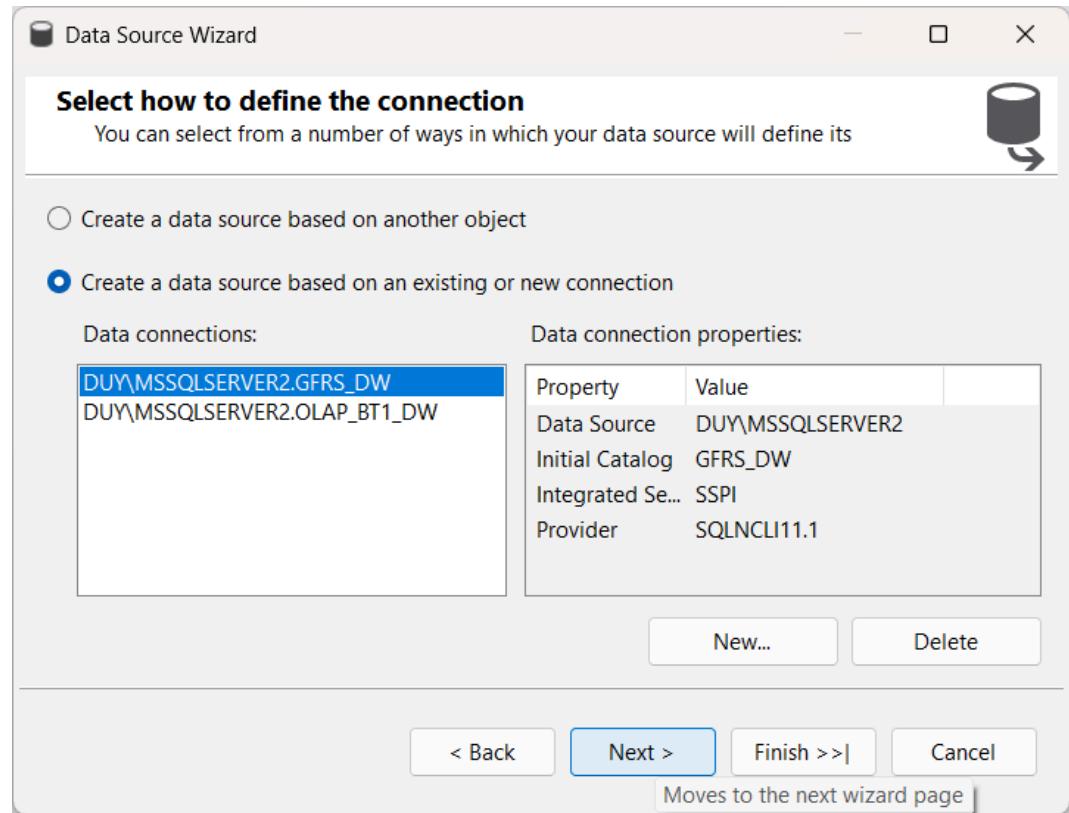
Hình 361. Tạo kết nối đến kho dữ liệu

- **Bước 3:** Cửa sổ **Connection Manager** sẽ hiện lên, ta chọn đúng tên server (**DUY\MSSQLSERVER2**) và kho dữ liệu (**GFRS\_DW**) đã tạo ở phần SSIS. Sau đó chọn **OK**.



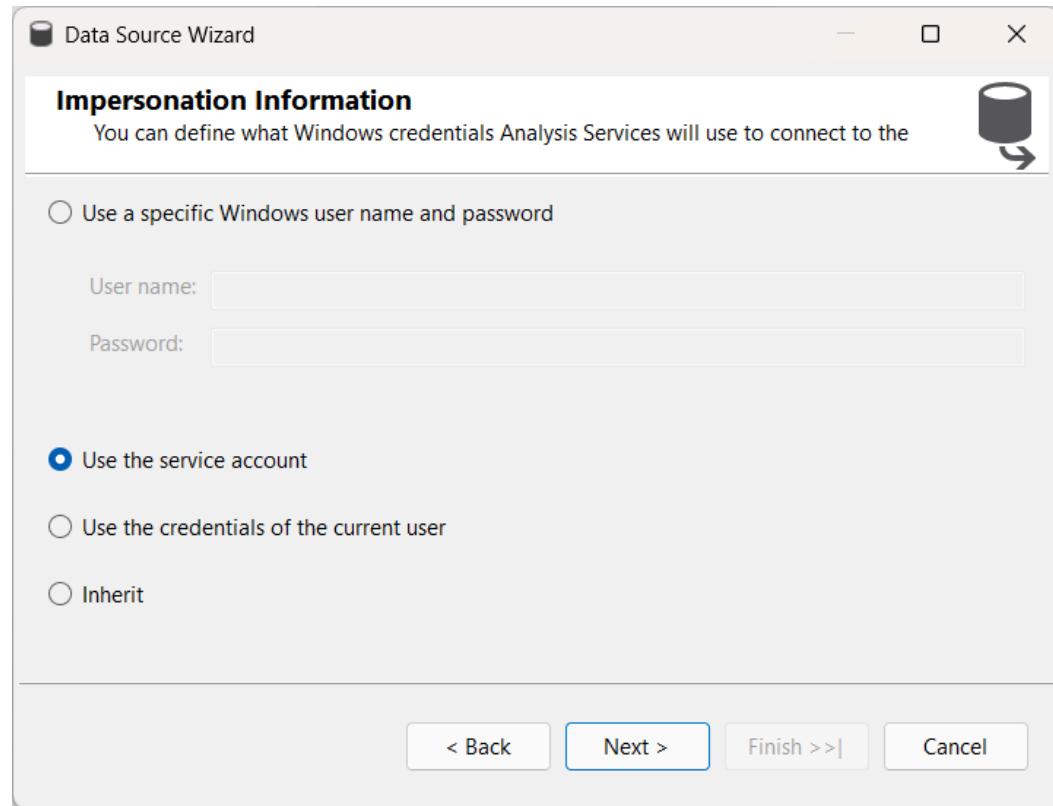
Hình 362. *Chỉnh sửa kết nối đến kho dữ liệu*

- **Bước 4:** Chọn **Data connections** ta vừa tạo, sau đó chọn **Next**.



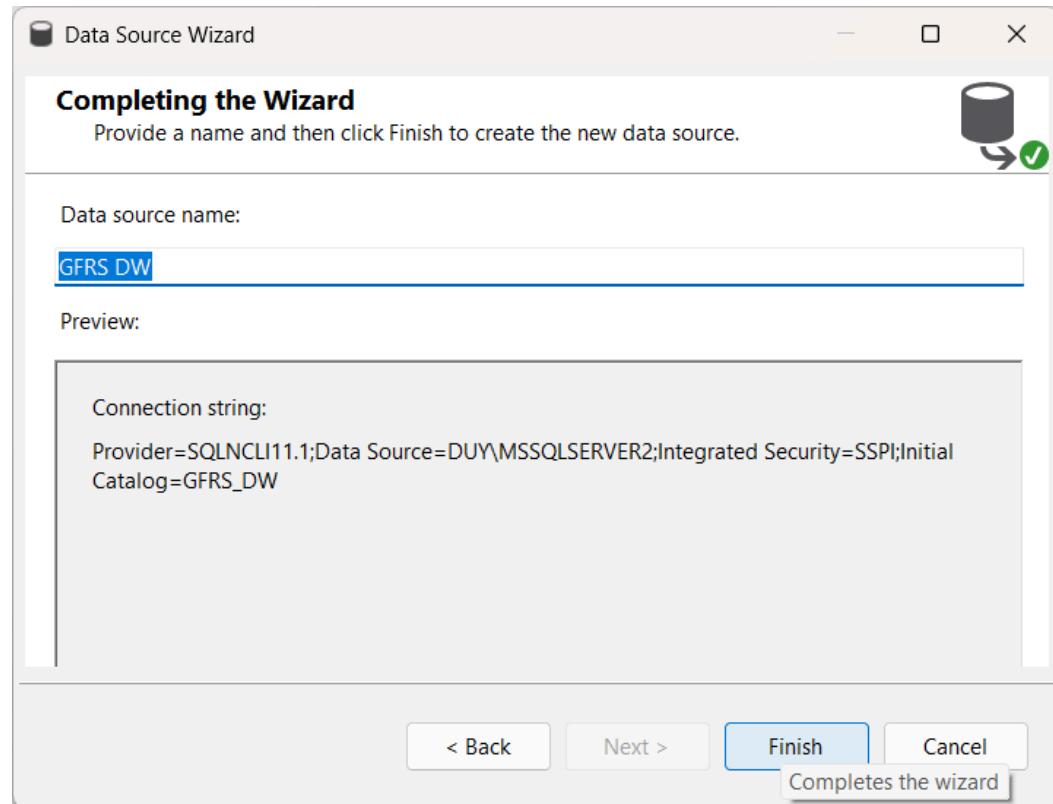
Hình 363. Chọn Data connections vừa tạo và tiếp tục

- **Bước 5:** Chọn **Use the service account**, sau đó chọn **Next**.



Hình 364. Chọn Use the service account và tiếp tục

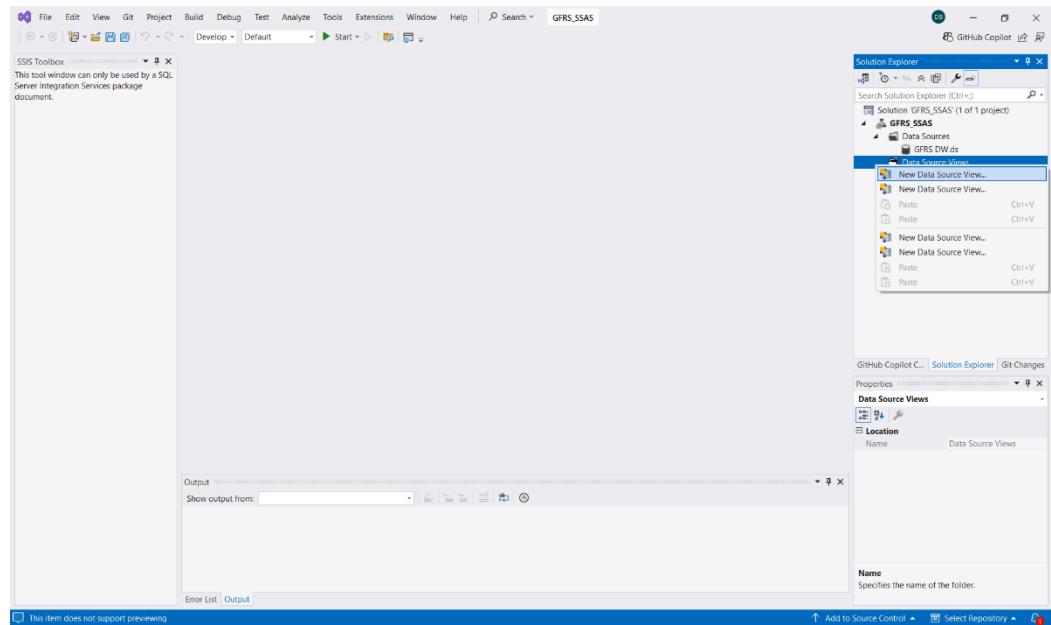
- **Bước 6:** Đặt tên cho **Data Source**, sau đó chọn **Finish**.



Hình 365. Đặt tên và kết thúc quá trình tạo kết nối

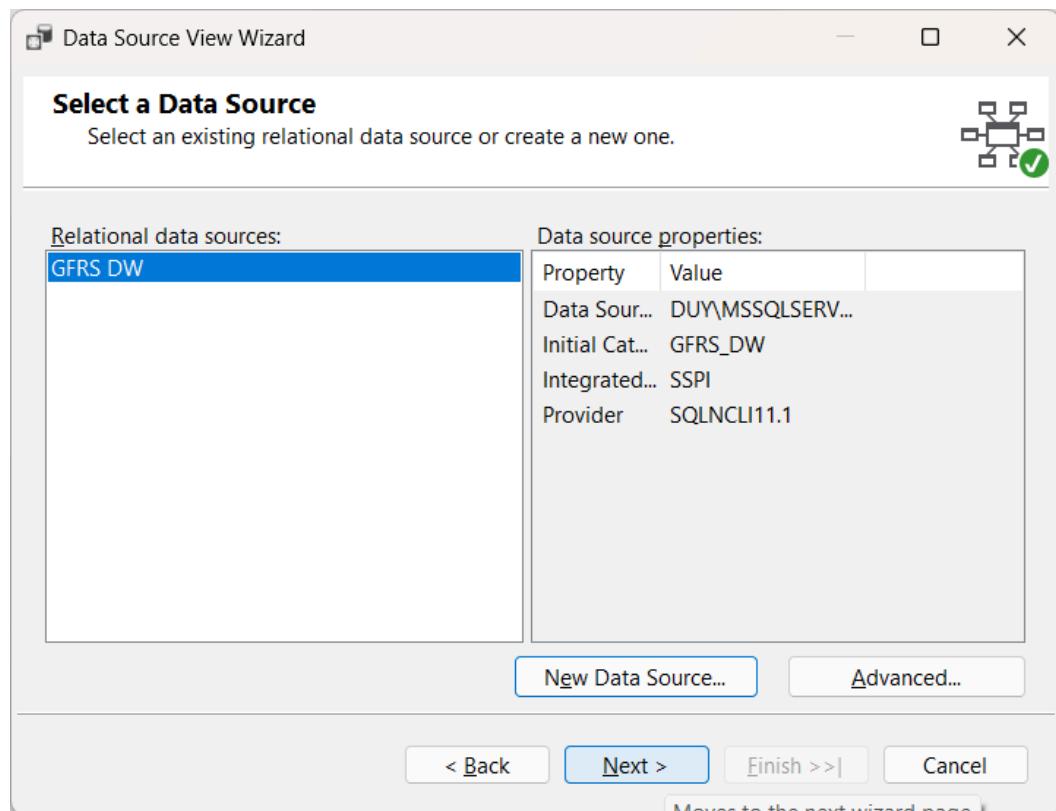
### III.2.1.2. Tạo Data Source View

- **Bước 1:** Ở mục **Data Source Views**, nhấn chuột phải sau đó chọn **New Data Source View...**



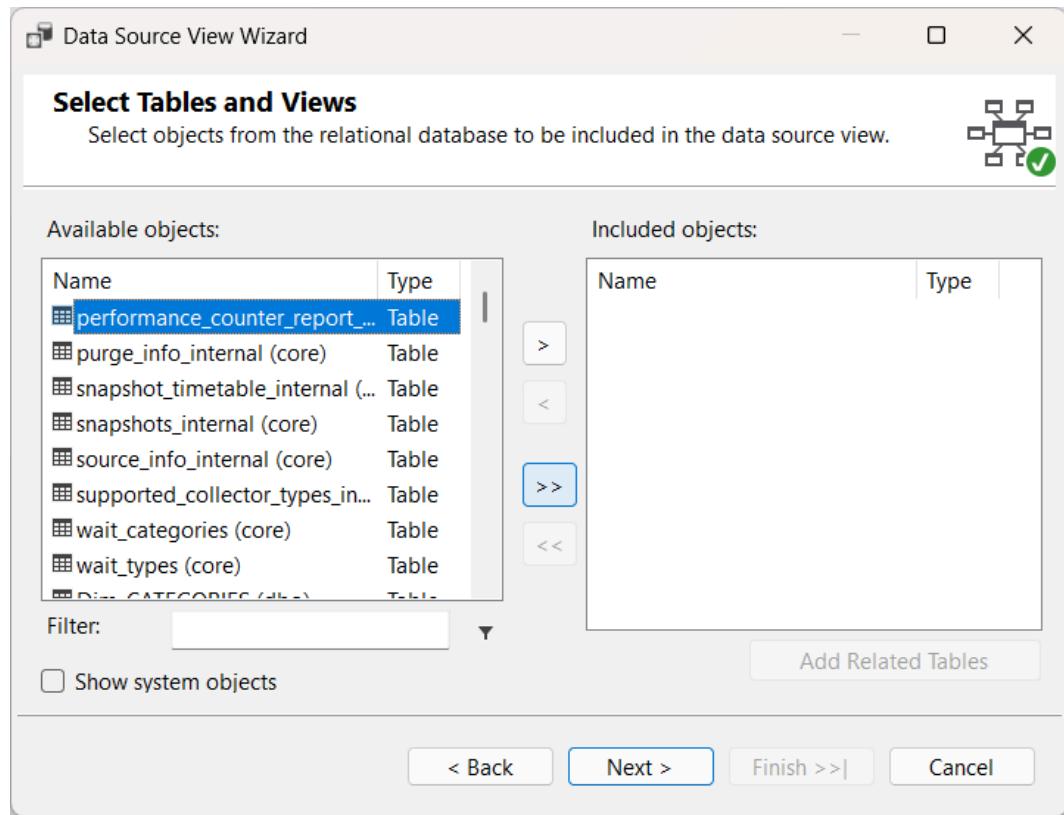
Hình 366. Tạo Data Source View mới

- **Bước 2:** Cửa sổ **Data Source View Wizard** hiện lên, ta chọn **GFRS\_DW** sau đó chọn **Next**.

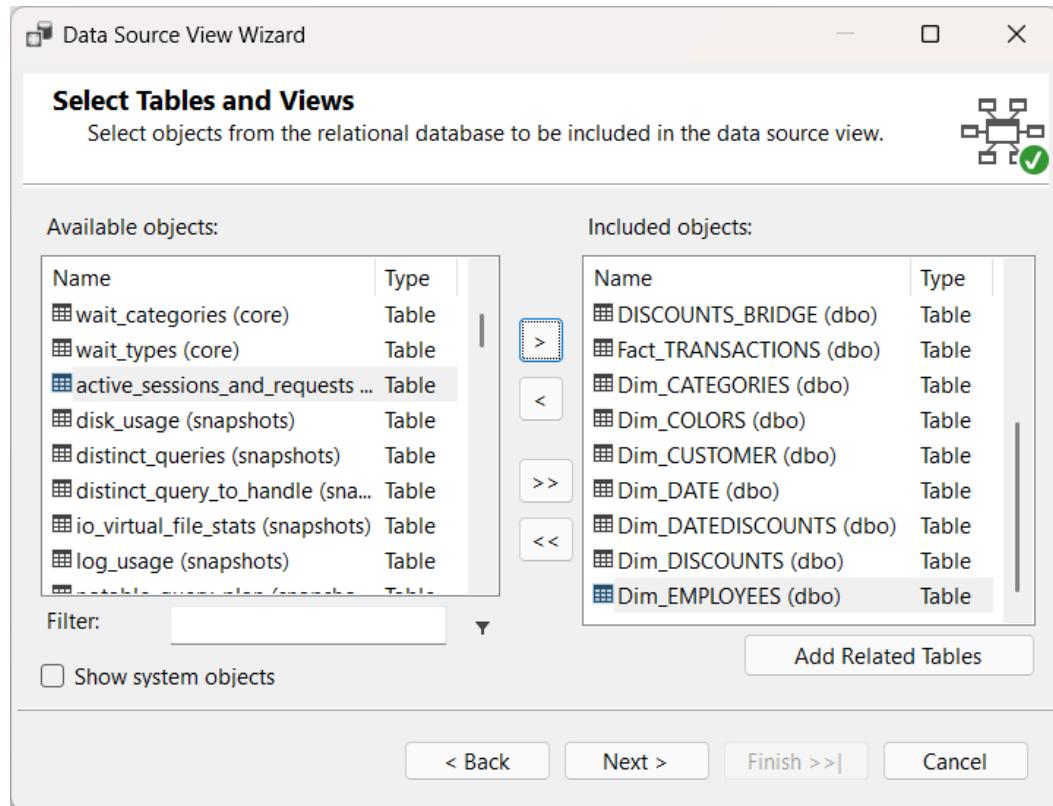


Hình 367. Lựa chọn Data Source

- **Bước 3:** Ta chọn các bảng đã tạo ở phần SSIS từ khung **Available objects** sau đó chọn > để đẩy sang **Included objects** (Ta có thể chọn nhiều bảng bằng cách giữ **Shift** sau đó chọn >> để đẩy nhiều bảng). Sau đó chọn **Next** để tiếp tục.

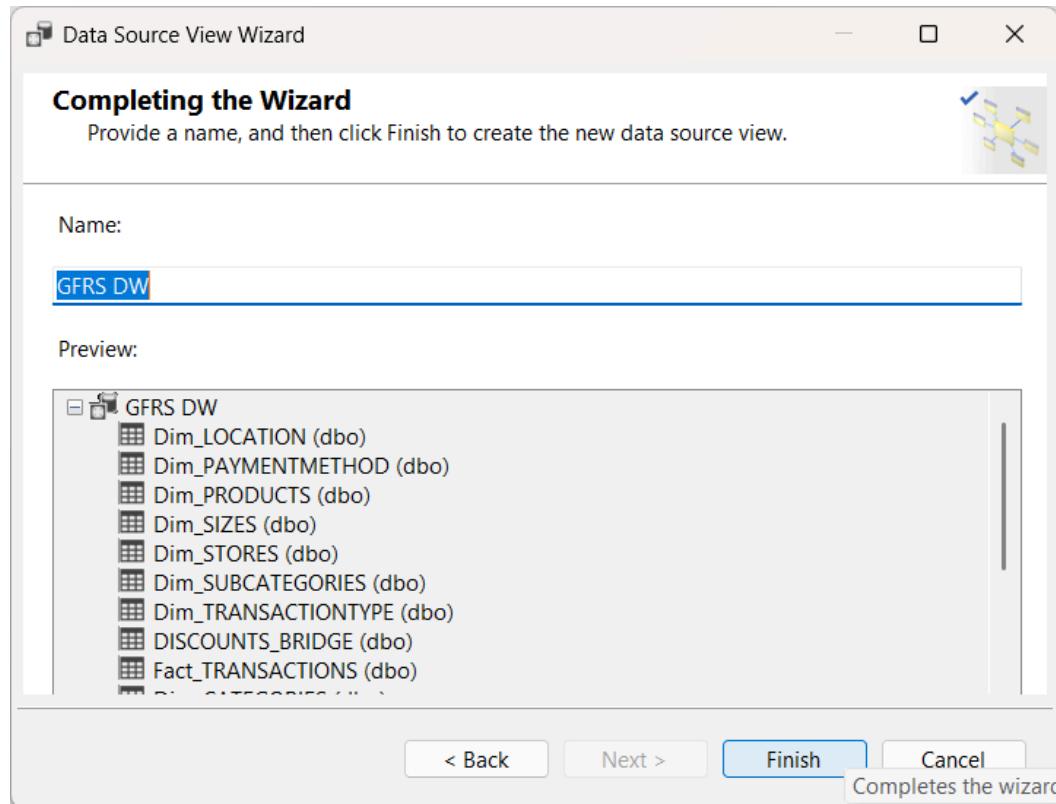


Hình 368. Chọn các bảng cần thiết cho Data Source View

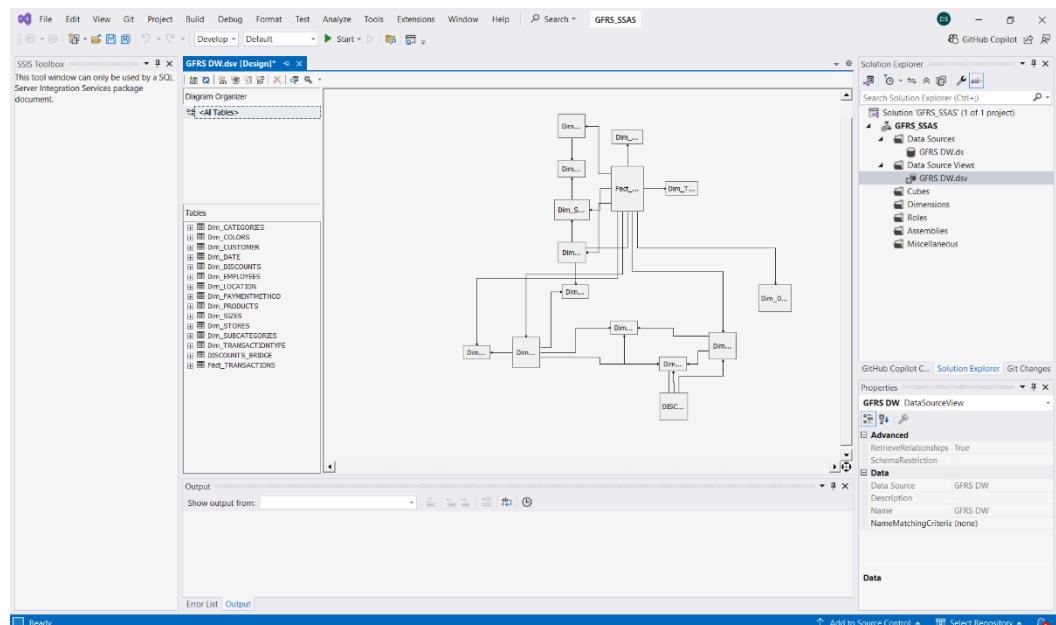


Hình 369. Các bảng cần thiết cho Data Source View

- **Bước 4:** Đặt tên sau đó chọn **Finish** để kết thúc.



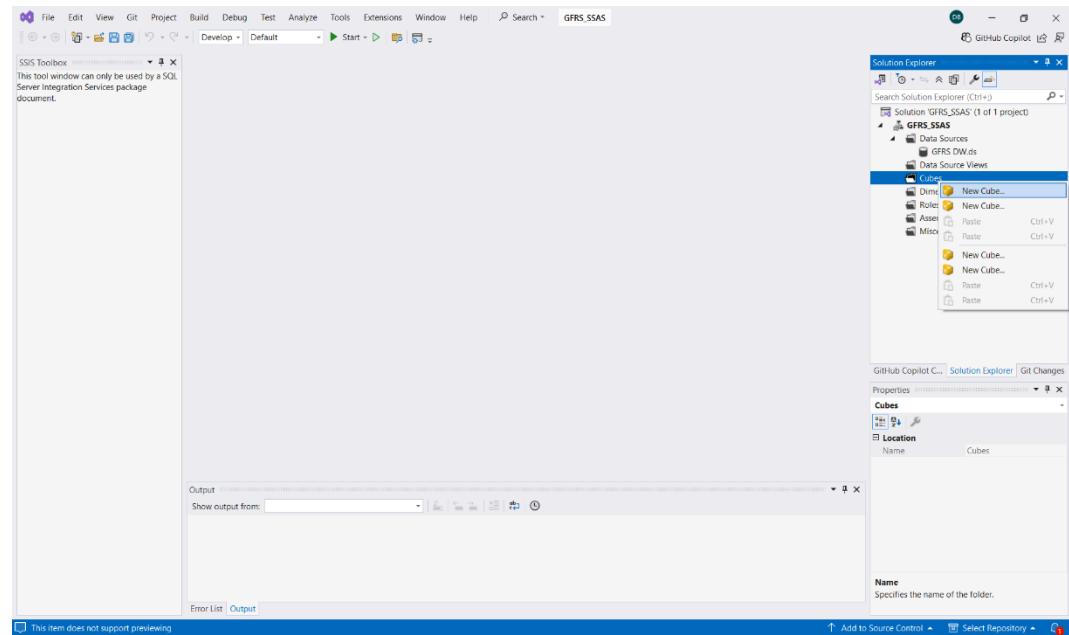
Hình 370. Đặt tên cho Data Source View sau đó kết thúc



Hình 371. Data Source View mới được thêm vào

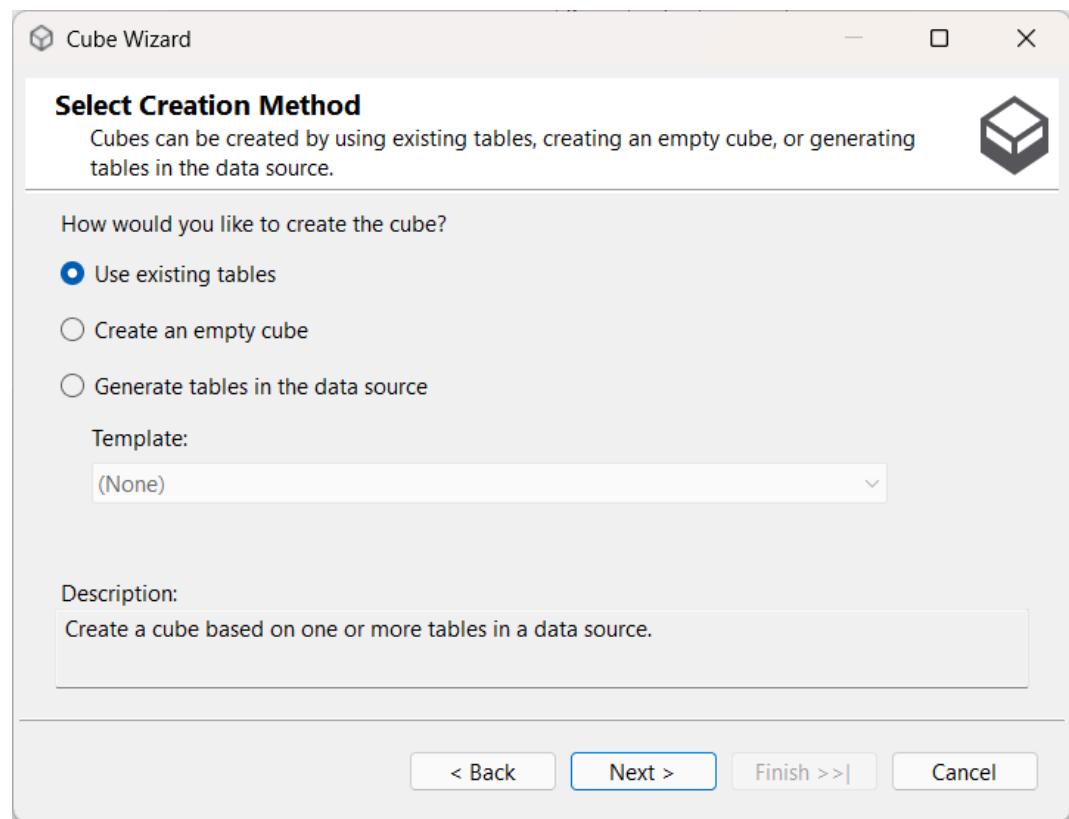
### III.2.1.3. Tạo Cube

- **Bước 1:** Ở mục Cubes, chọn New Cube...



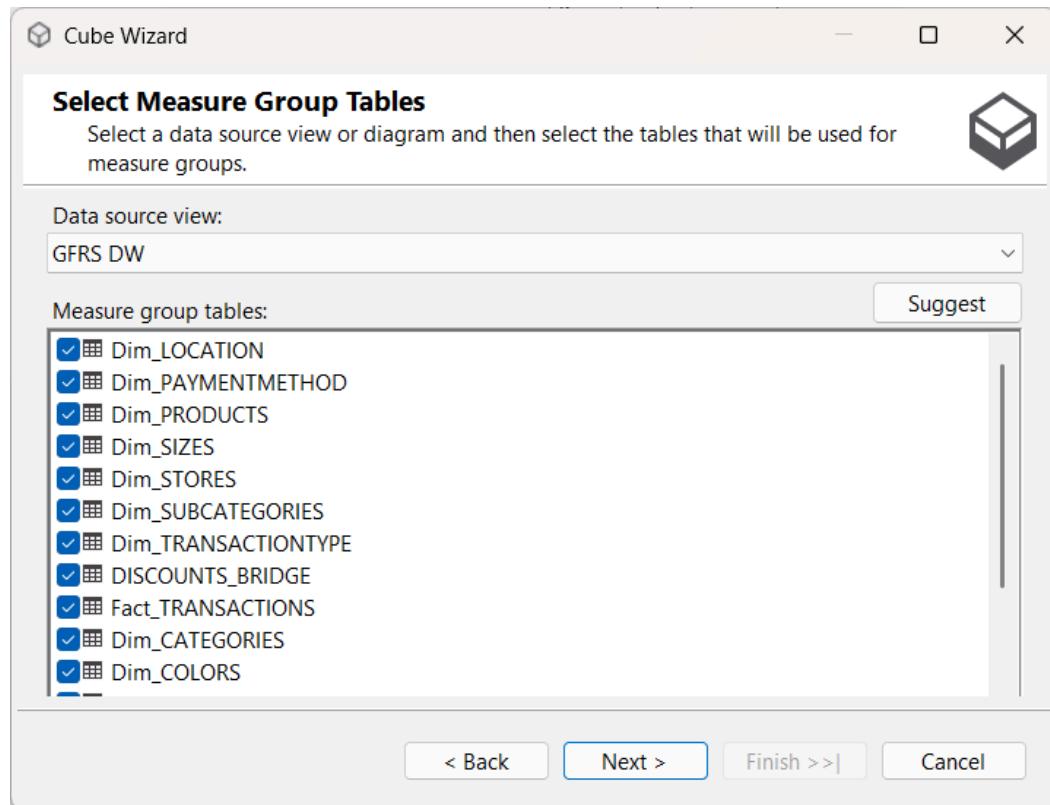
Hình 372. Tạo cube mới

- **Bước 2:** Cửa sổ **Cube Wizard** hiện lên, ta chọn **Use existing tables**, sau đó chọn **Next**.



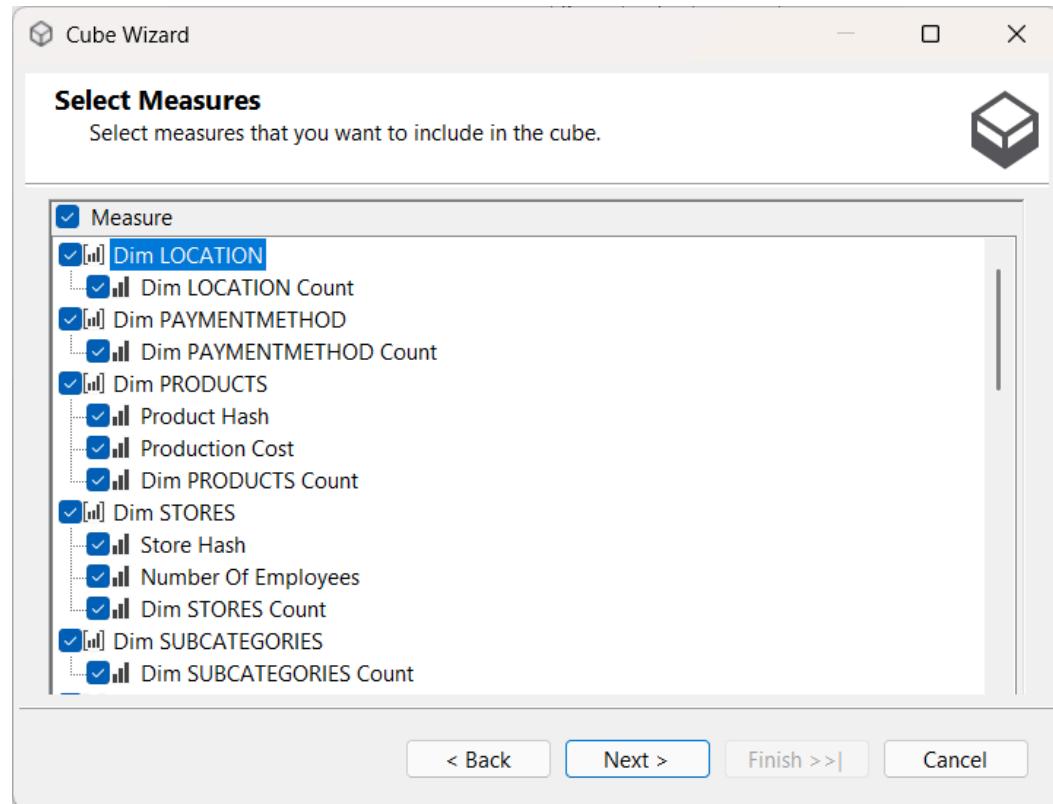
Hình 373. Lựa chọn phương thức cho Cube

- **Bước 3:** Chọn các Measure group tables cần thiết để đưa vào **Cube**, sau đó chọn **Next**.



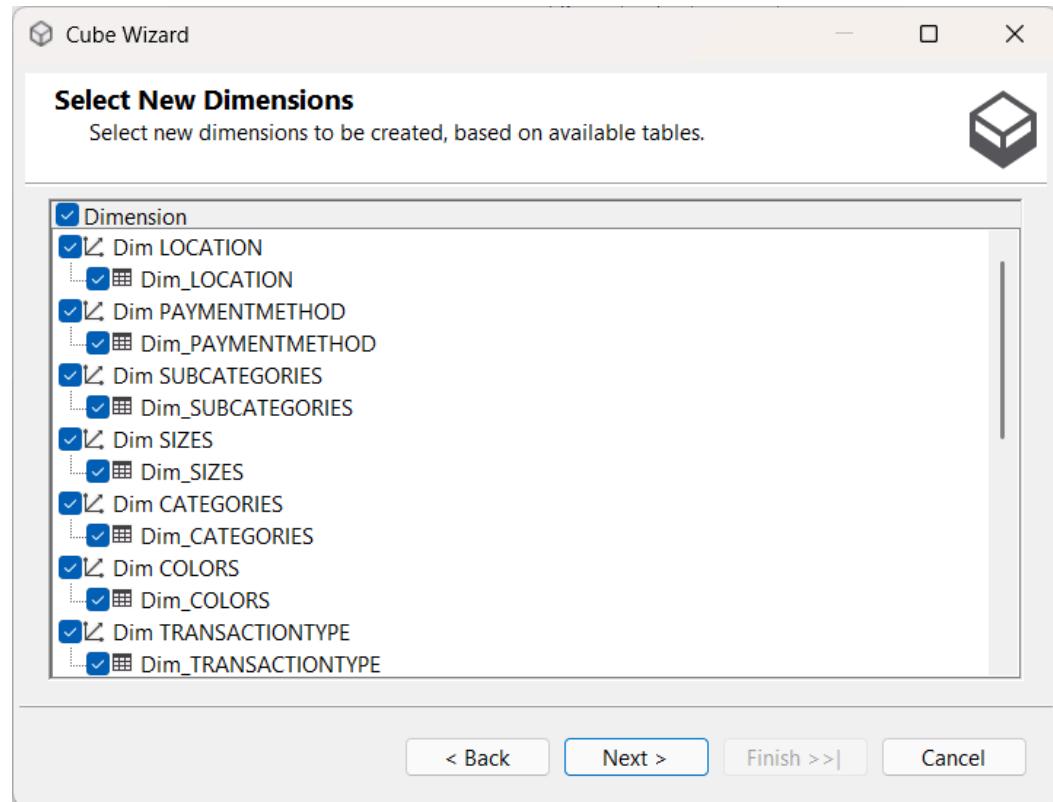
Hình 374. Chọn các Measure group tables cần thiết

- **Bước 4:** Lựa chọn các **Measure** cần thiết, sau đó chọn **Next** để tiếp tục.



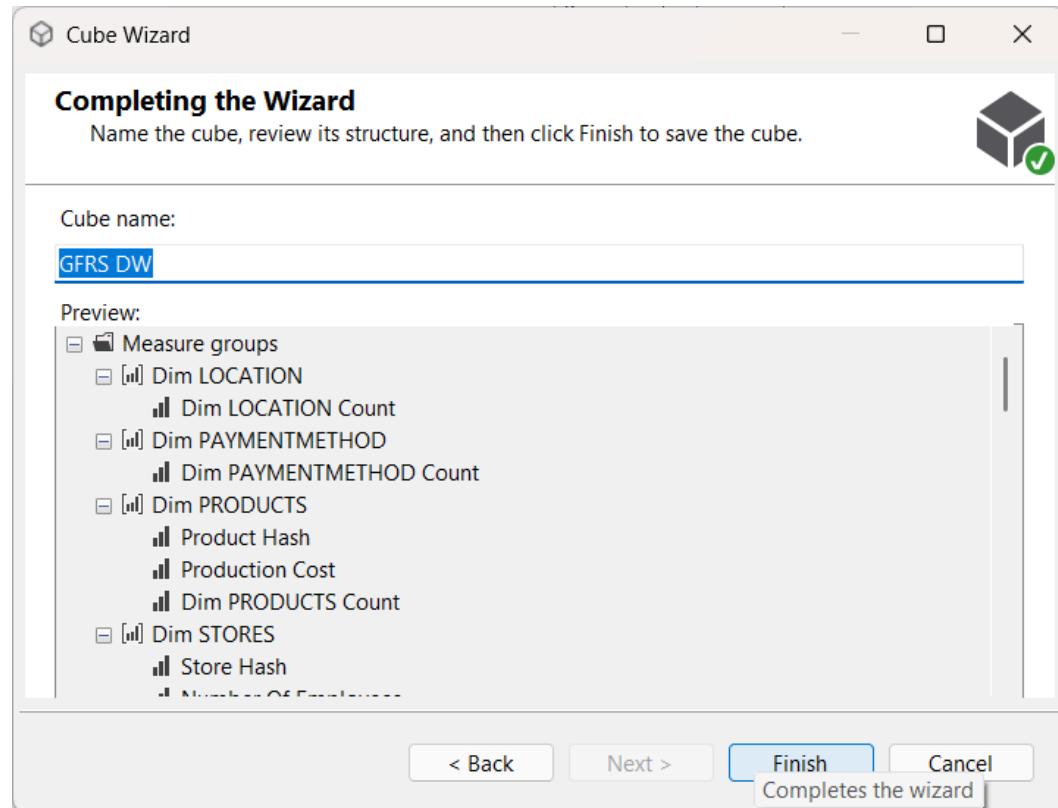
Hình 375. Chọn các Measure phù hợp cho Cube

- **Bước 15:** Chọn các **Dimension** phù hợp cho **Cube**, sau đó chọn **Next**.

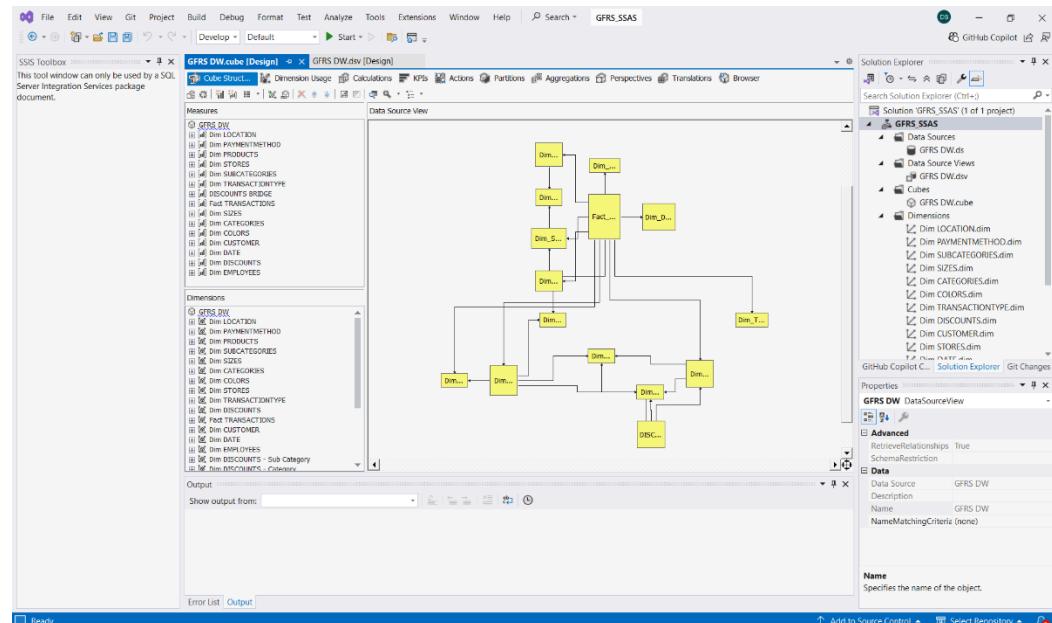


Hình 376. Chọn các Dimension cần thiết cho Cube

- **Bước 16:** Sau đó kiểm tra lại dữ liệu cho **Cube**, chọn **Finish** để kết thúc.



Hình 377. Kiểm tra và tạo Cube mới

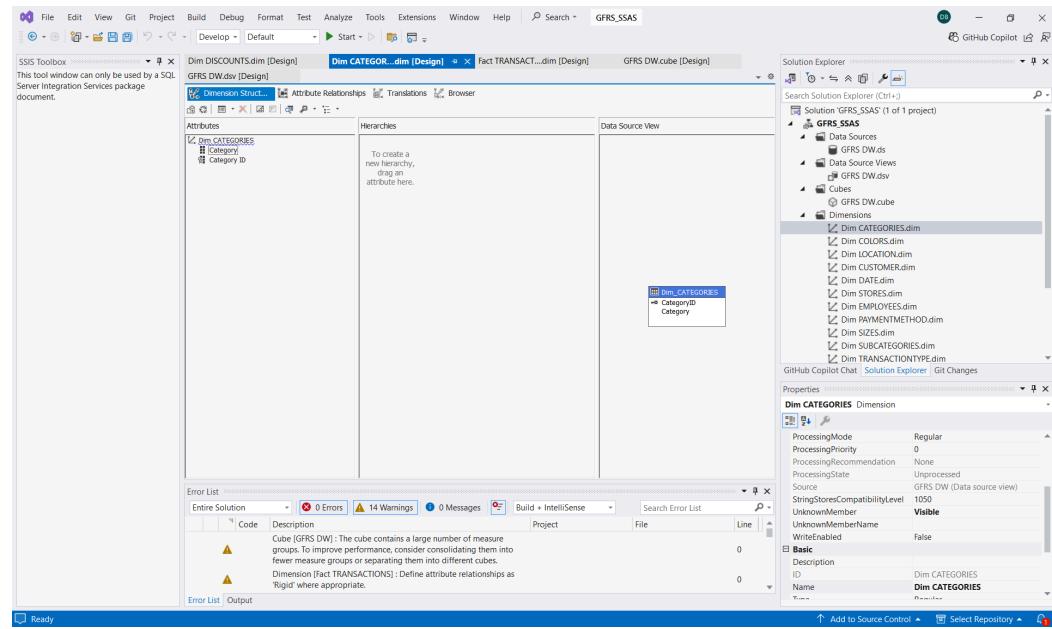


Hình 378. Tạo Cube thành công

### III.2.1.4. Cài đặt các Dimensions và phân cấp dữ liệu

#### III.2.1.4.1. Cài đặt Dim\_CATEGORIES

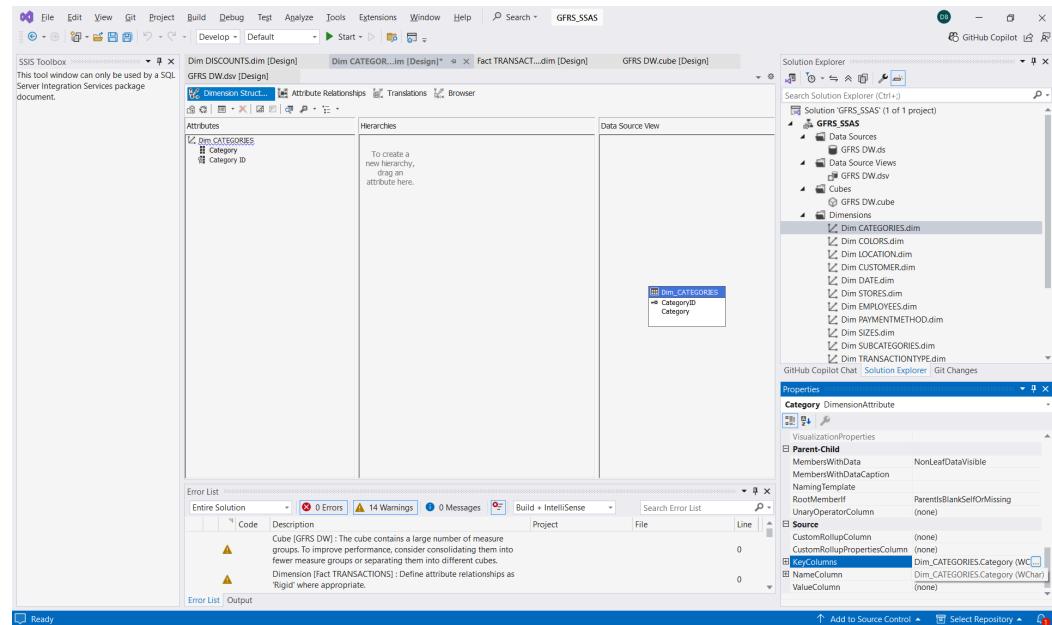
- **Bước 1:** Kéo thuộc tính Category từ khung Data Source View sang khung Attributes.



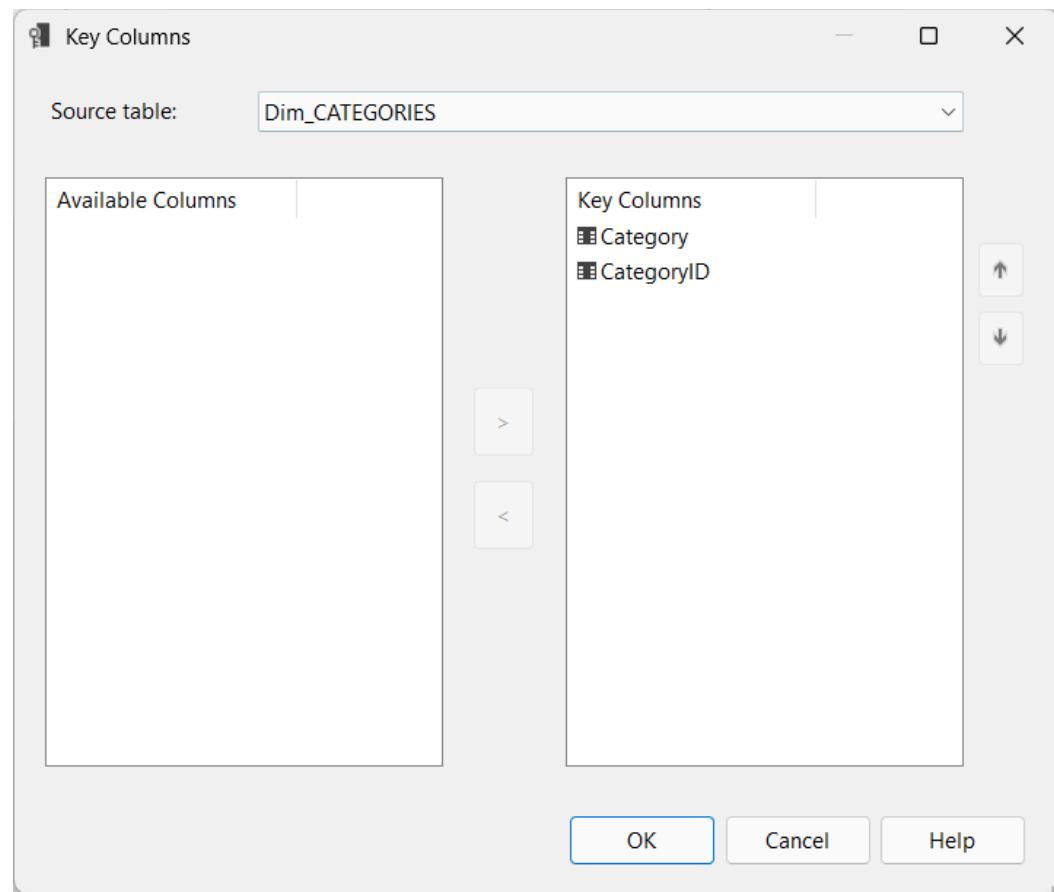
Hình 379. Kéo thuộc tính Category sang khung Attributes

- **Bước 2:** Nhấn chuột vào **Category** trên khung **Attributes**. Ở phần **Properties** ta kéo xuống phần **KeyColumns**, nhấn chuột vào (...) bên phải. Cửa sổ **Key Columns** hiện lên, ta chọn thuộc tính **CategoryID** từ khung **Available Columns**, sau đó nhấn > để đẩy sang khung **Key Columns**, nhấn **OK** để hoàn tất. Tiếp đó, ta nhấn chuột vào (...) ở **NameColumn**, cửa sổ

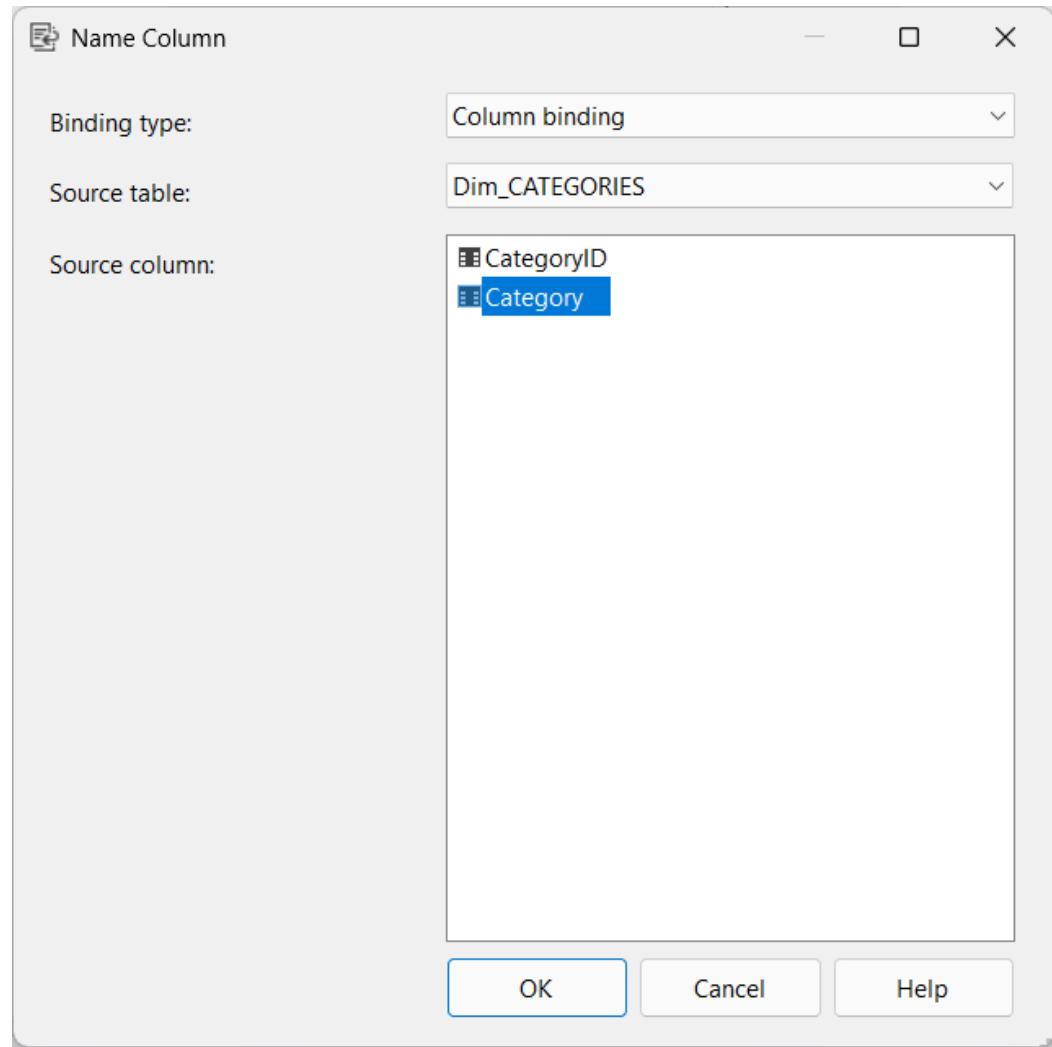
**Name Column** hiện lên, ta chọn **Category** sau đó nhấn **OK**.



Hình 380. Thêm thuộc tính vào KeyColumns cho Category



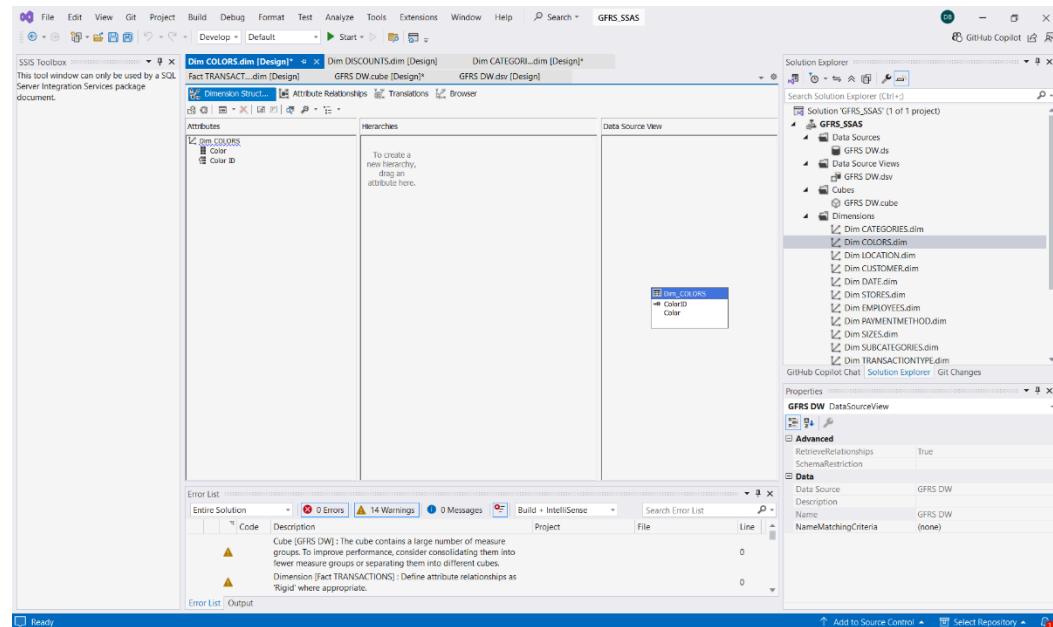
Hình 381. Thêm thuộc tính CategoryID vào Key Columns



Hình 382. Chọn tên cho Name Column cho thuộc tính Category

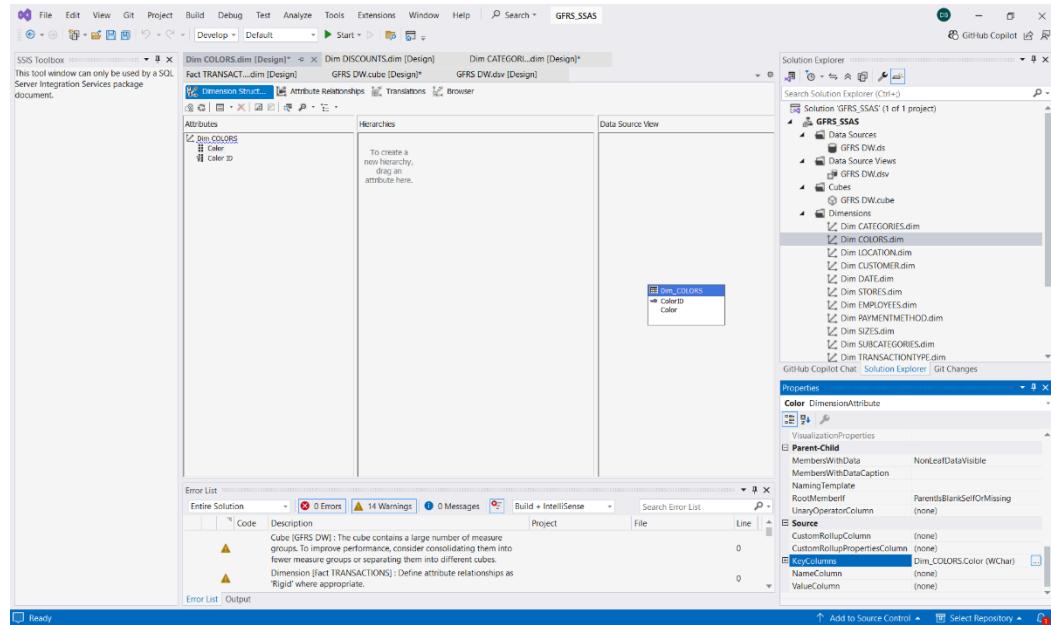
#### III.2.1.4.2. Cài đặt Dim\_COLORS

- **Bước 1:** Kéo thuộc tính **Color** từ khung **Data Source View** sang khung **Attributes**.

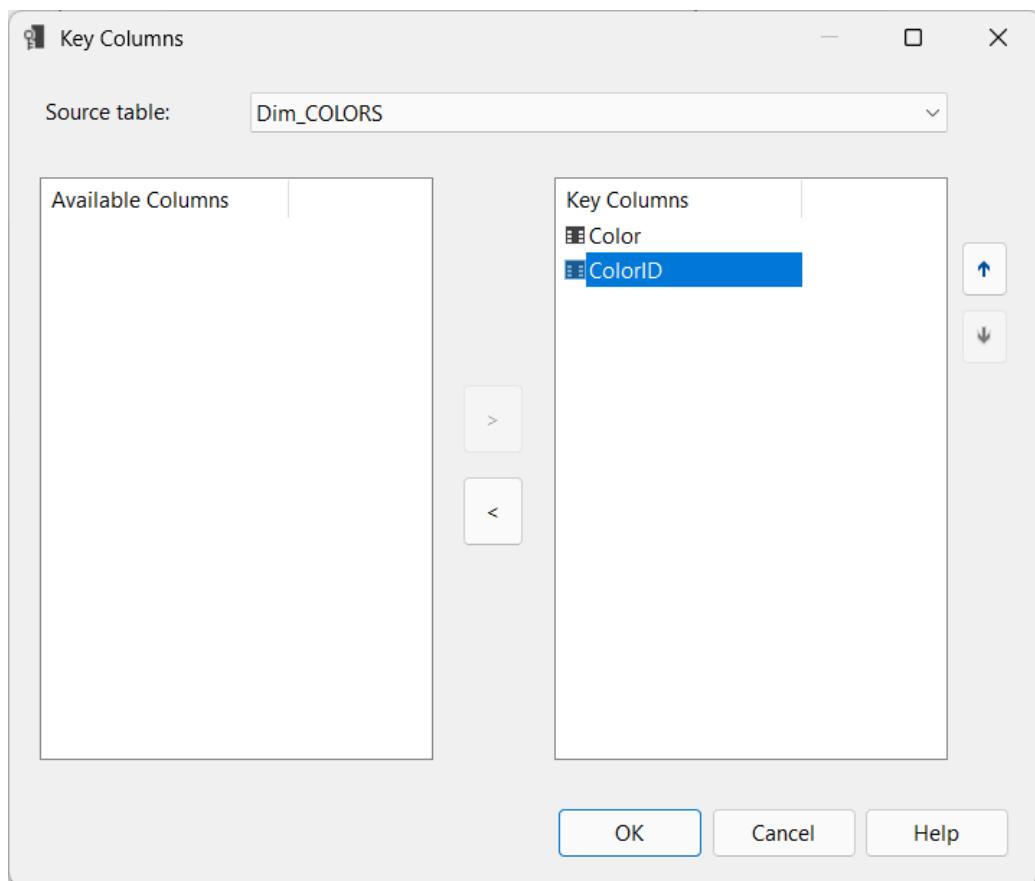


Hình 383. Kéo thuộc tính Color sang khung Attributes

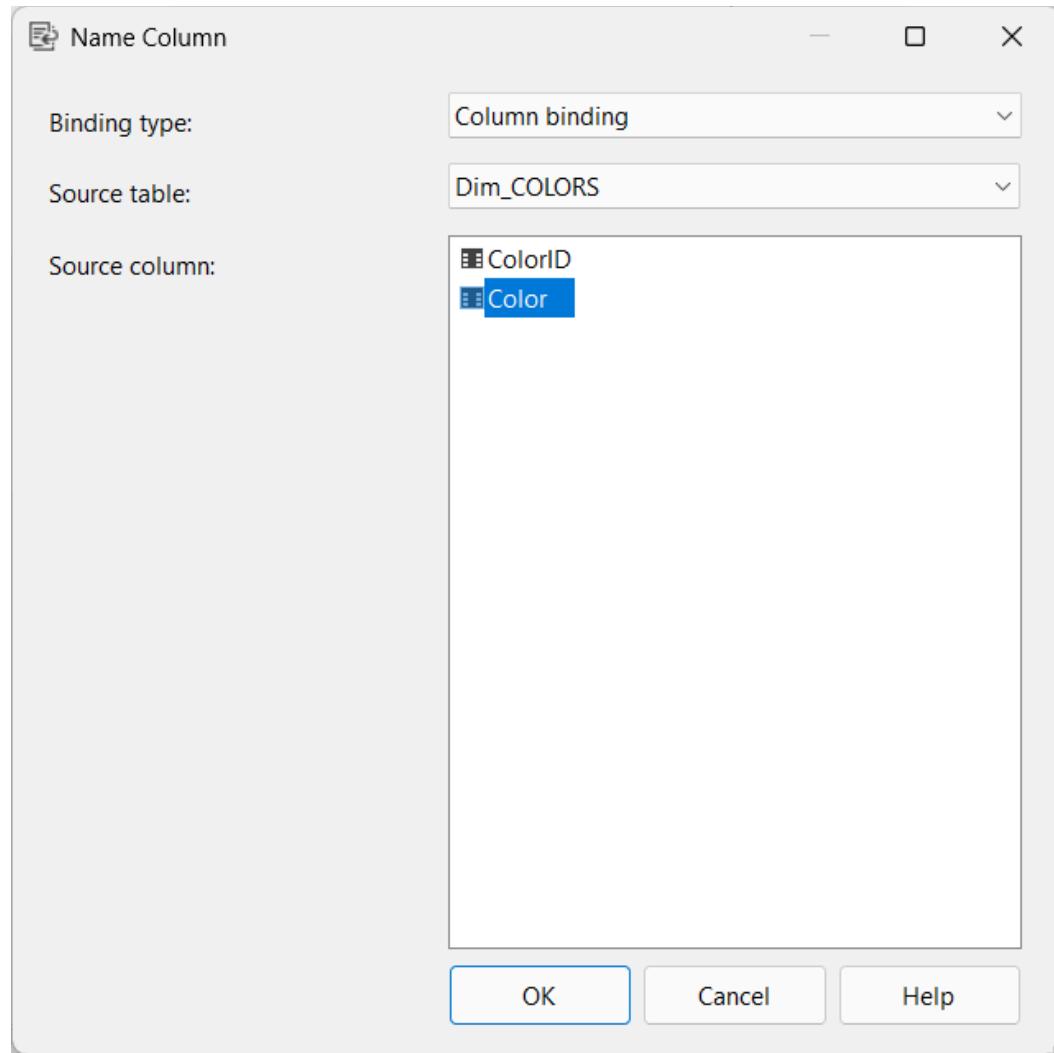
- **Bước 2:** Nhấn chuột vào **Color** trên khung **Attributes**. Ở phần **Properties** ta kéo xuống phần **KeyColumns**, nhấn chuột vào (...) bên phải. Cửa sổ **Key Columns** hiện lên, ta chọn thuộc tính **ColorID** từ khung **Available Columns**, sau đó nhấn > đè đẩy sang khung **Key Columns**, nhấn **OK** để hoàn tất. Tiếp đó, ta nhấn chuột vào (...) ở **NameColumn**, cửa sổ **Name Column** hiện lên, ta chọn **Color** sau đó nhấn **OK**.



Hình 384. Thêm thuộc tính vào KeyColumns cho Color



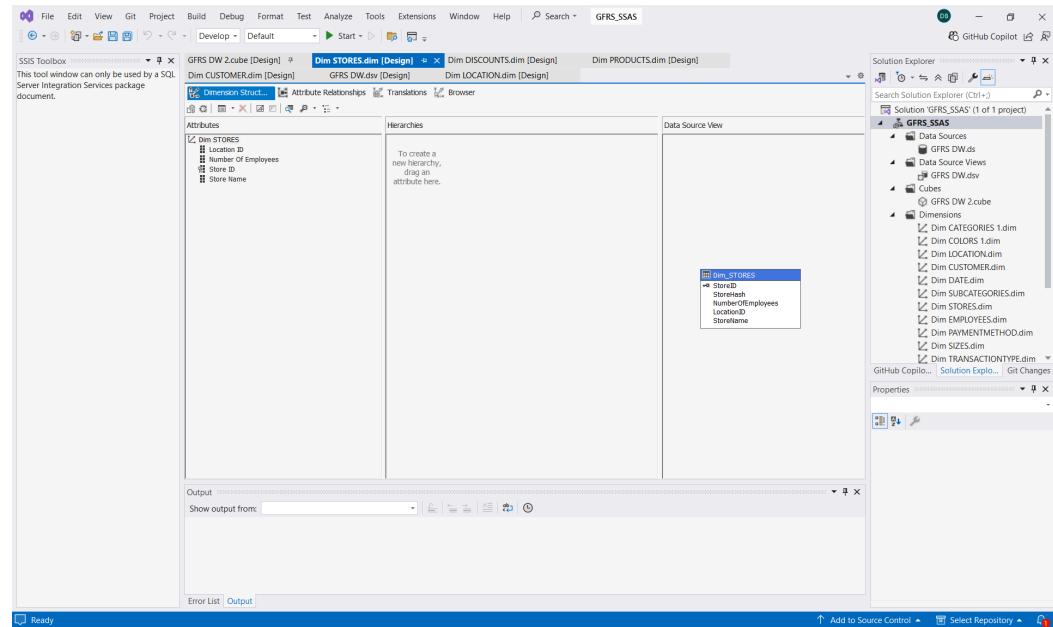
Hình 385. Thêm thuộc tính ColorID vào Key Columns



Hình 386. Chọn tên cho Name Column cho thuộc tính Color

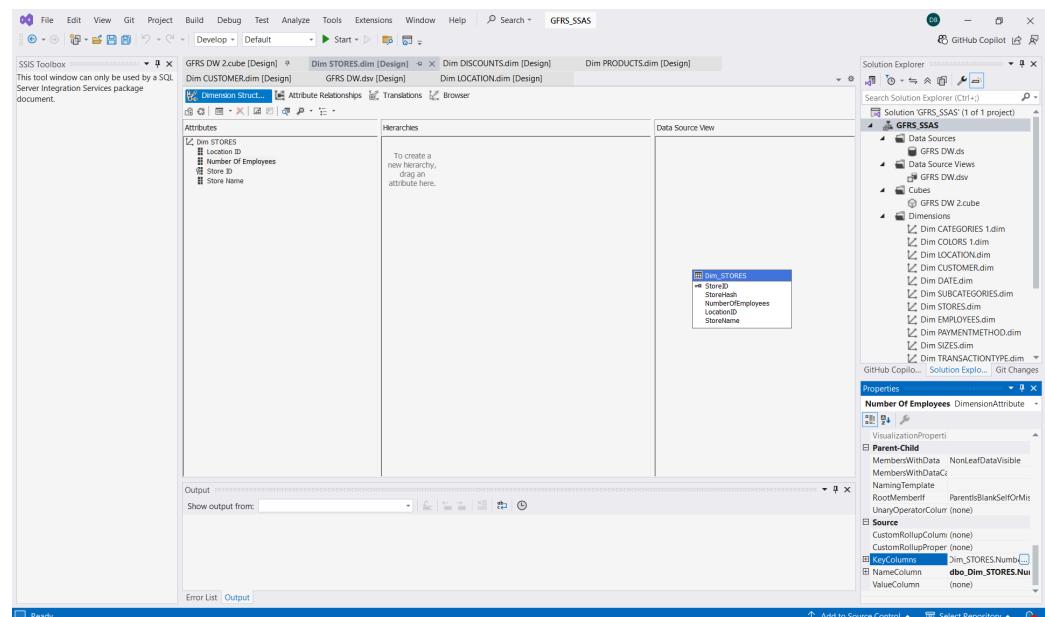
#### **III.2.1.4.3. Cài đặt Dim\_STORES**

- **Bước 1:** Kéo thuộc tính **NumberOfEmployees** và **StoreName** từ khung **Data Source View** sang khung **Attributes**.

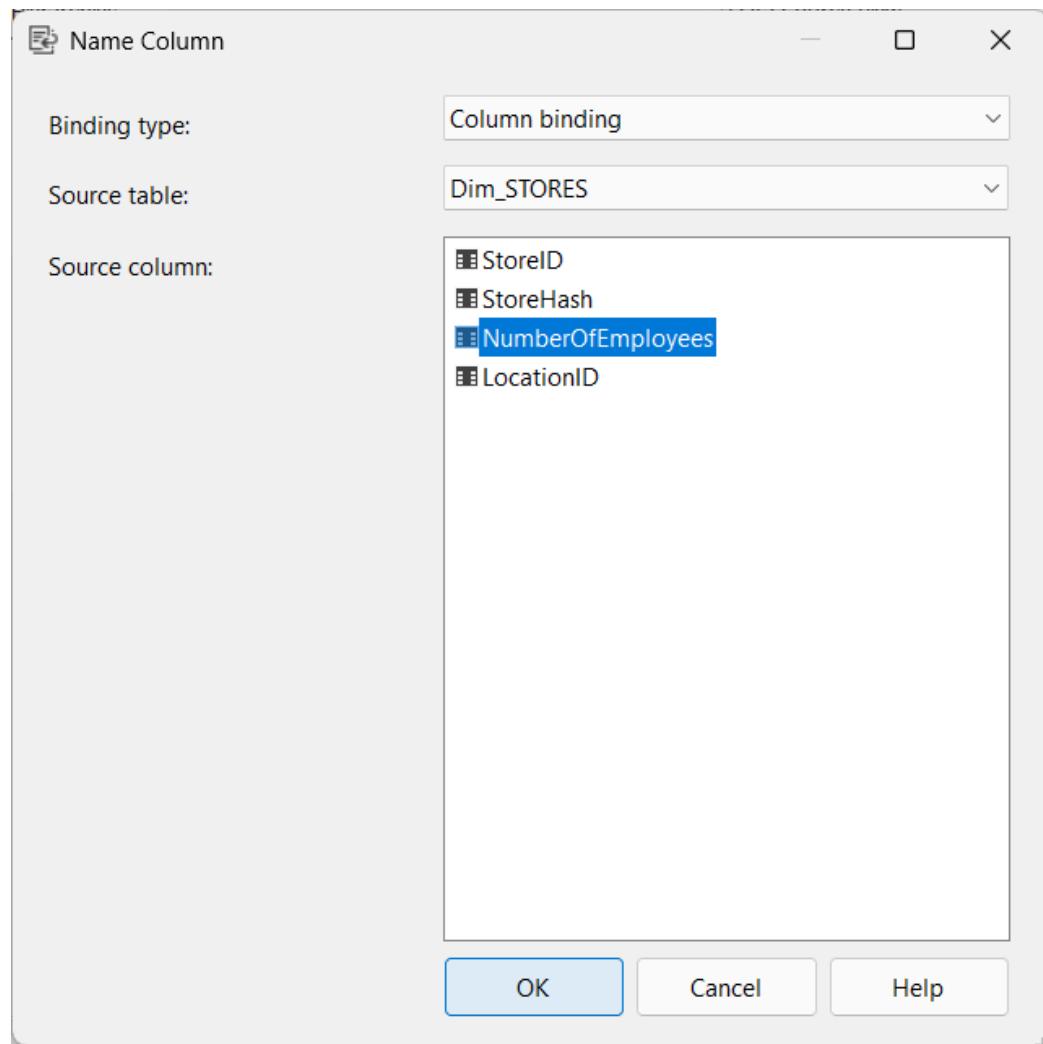


Hình 387. Kéo thuộc tính *NumberOfEmployees* và *Store Name* sang khung *Attributes*

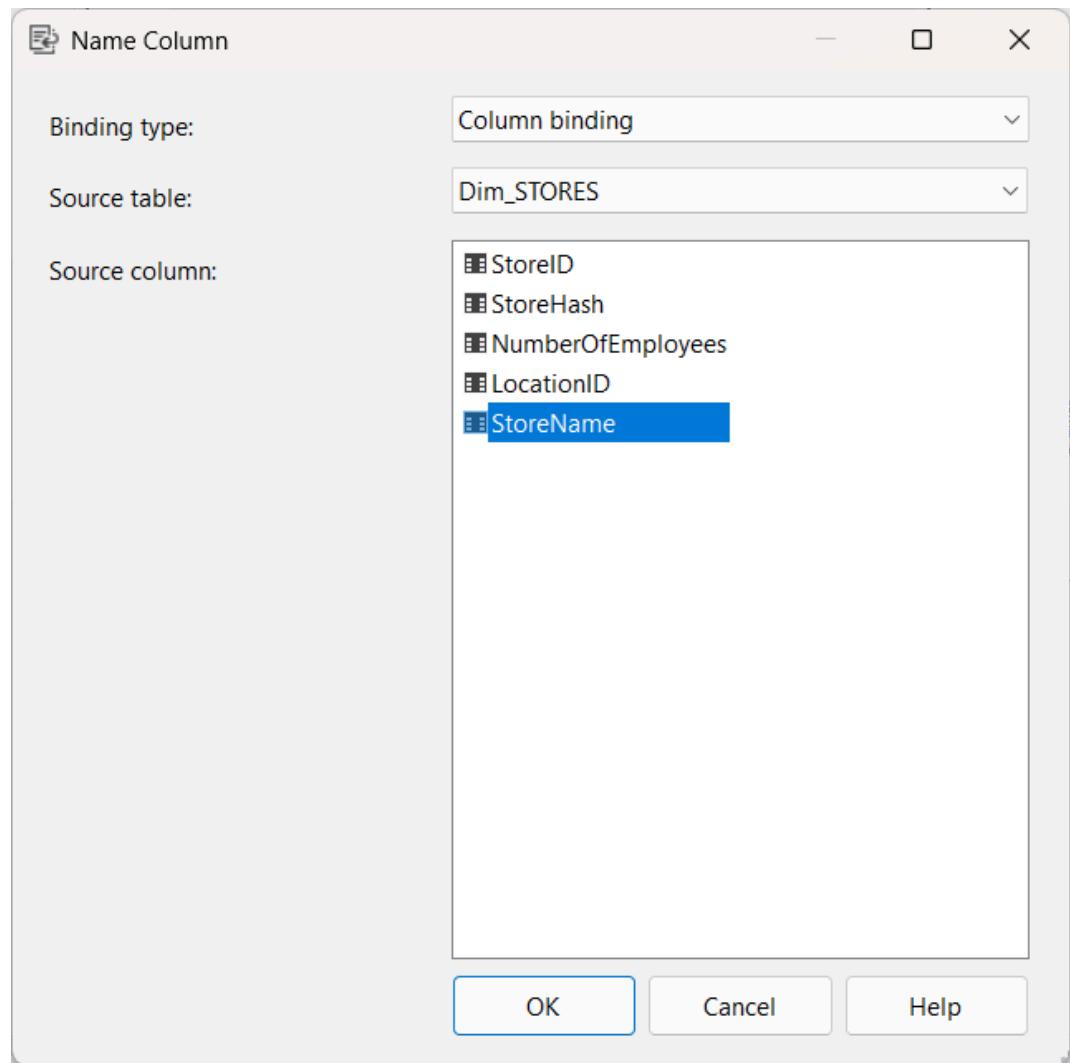
- **Bước 2:** Nhấn chuột vào **NumberOfEmployees** trên khung **Attributes**. Ở phần **Properties** ta kéo xuống phần **NameColumn** ta nhấn (...), cửa sổ **Name Column** hiện lên, ta chọn **NumberOfEmployees** sau đó nhấn **OK**. Làm tương tự với **Age** với **Name Column** là **Age**



Hình 388. Đổi *Name Column* cho *NumberOfEmployees*



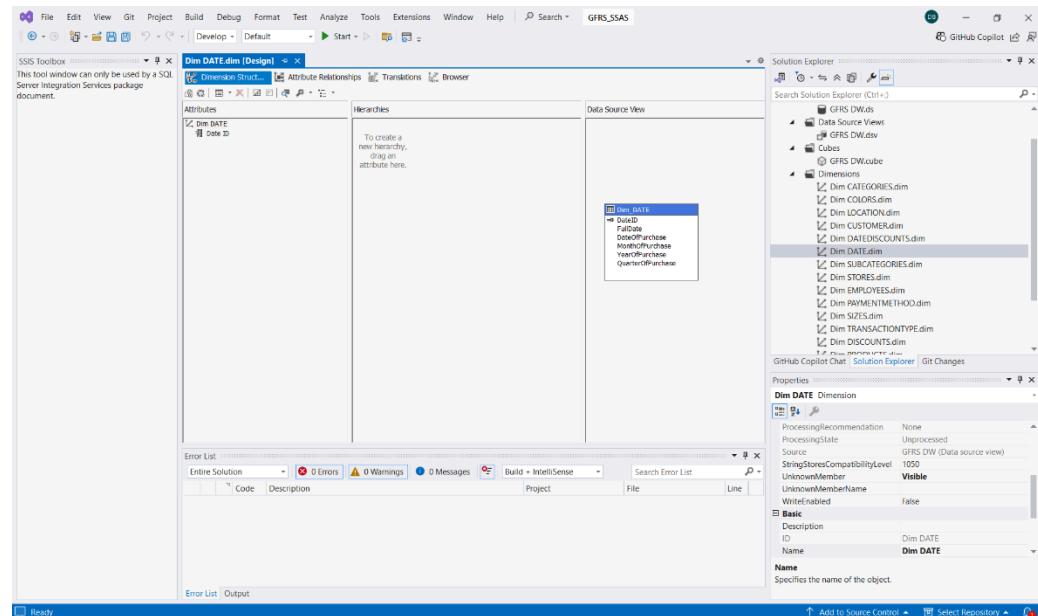
Hình 389. Chọn tên cho Name Column cho thuộc tính *NumberOfEmployees*



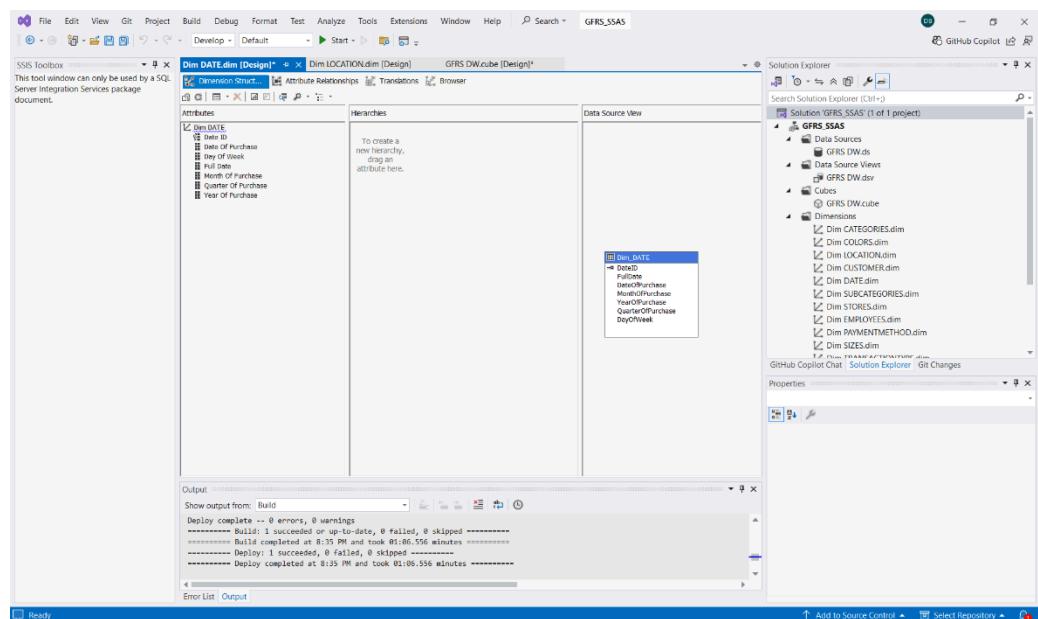
Hình 390. Chọn tên cho Name Column cho thuộc tính StoreName

#### **III.2.1.4.4. Cài đặt và phân cấp dữ liệu Dim\_DATE**

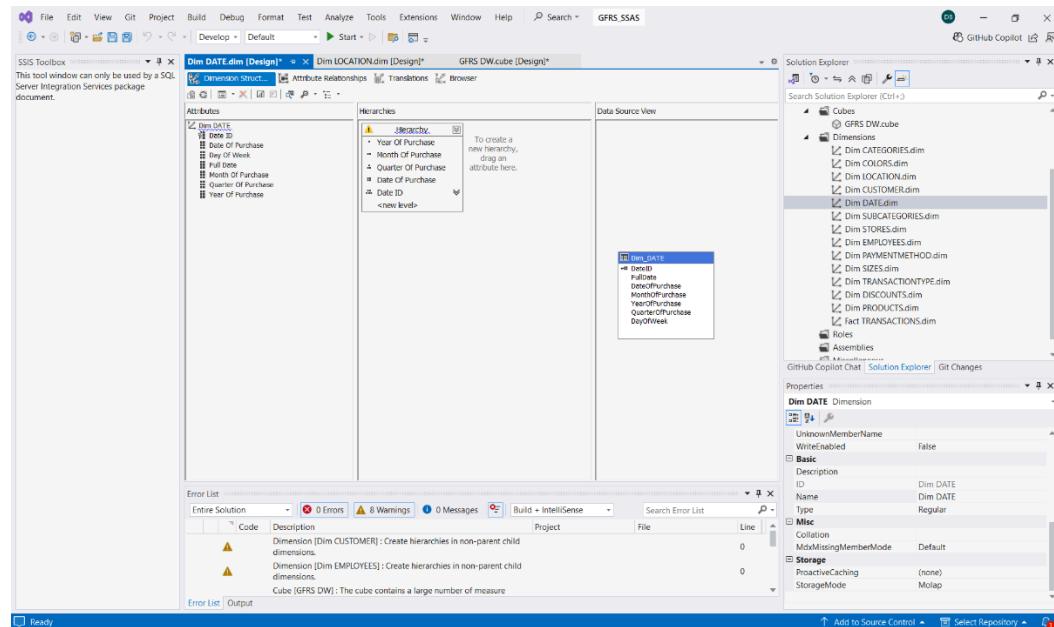
- **Bước 1:** Mở **Dim Date**, vào tab **Dimension Structure**. Ta kéo các cột dữ liệu từ bảng **Data Source View** sang bảng **Attributes**, sau đó kéo các cột này từ **Attributes** vào **Hierachies**. Sau đó sắp xếp theo thứ tự từ trên xuống: **YearOfPurchase**, **QuarterOfPurchase**, **MonthOfPurchase**, **DateID**



Hình 391. Chọn Dim DATE

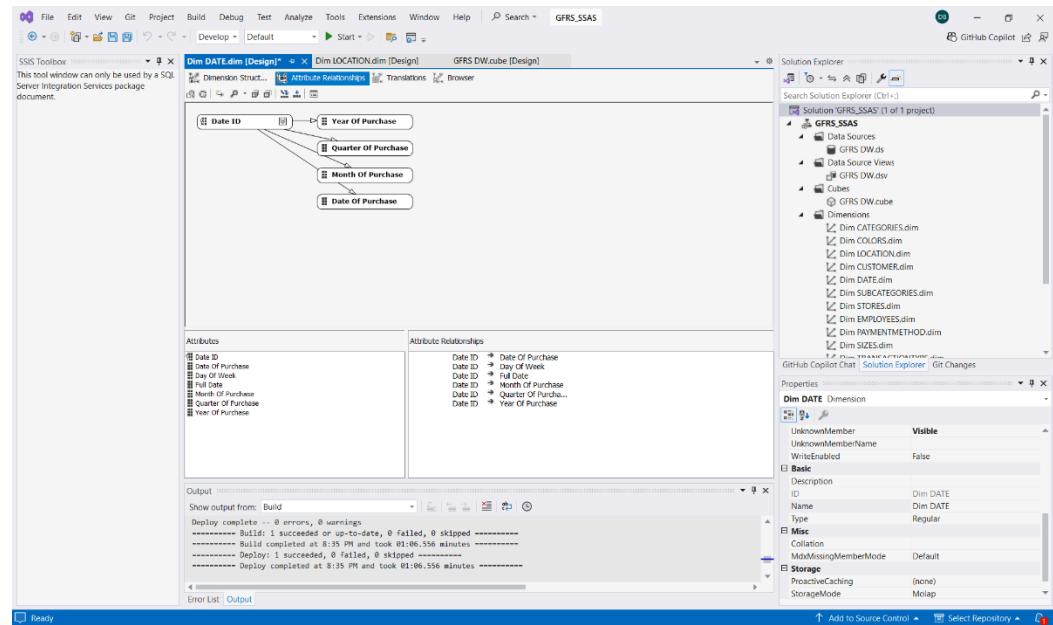


Hình 392. Kéo các thuộc tính của Dim Date vào Attributes

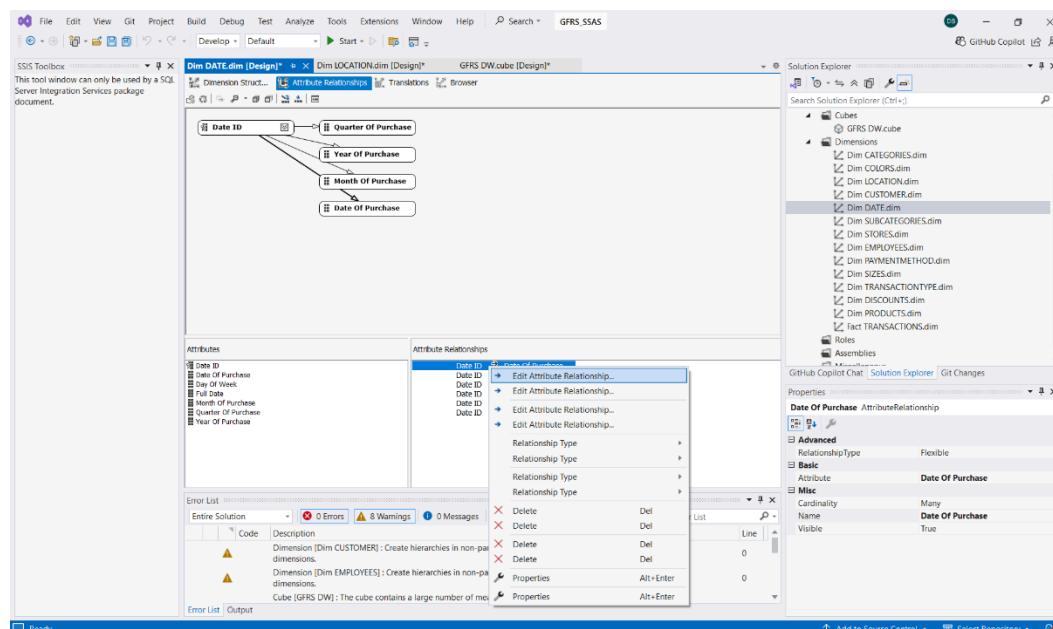


Hình 393. Phân cấp dữ liệu Dim DATE theo thứ tự

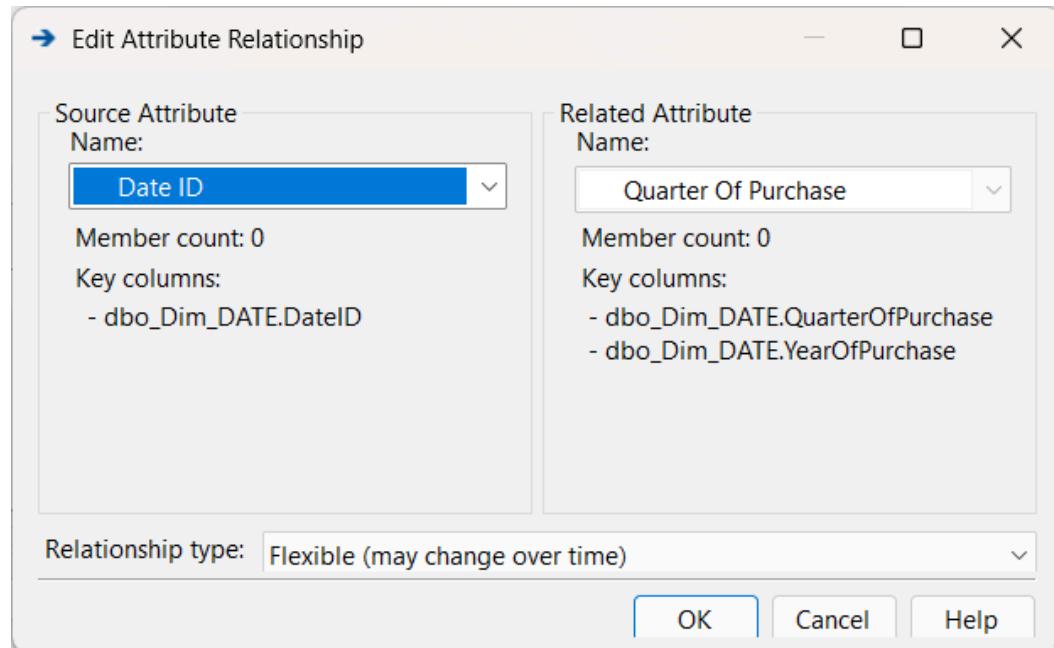
- **Bước 2:** Chuyển sang tab **Attribute Relationship**, nhấp chuột phải vào thuộc tính trong khung **Attribute Relationship** chọn **Edit Attribute Relationship....**. Cửa sổ **Edit Attribute Relationship** sẽ hiện lên, ta đổi **Name** phần **Source Attribute** :
  - o **Thuộc tính Month Of Purchase:** Date ID thành Date Of Purchase
  - o **Thuộc tính Quarter Of Purchase:** Date ID thành Month Of Purchase
  - o **Thuộc tính Year Of Purchase:** Date ID thành Quarter Of Purchase
 Sau đó chọn **OK**.



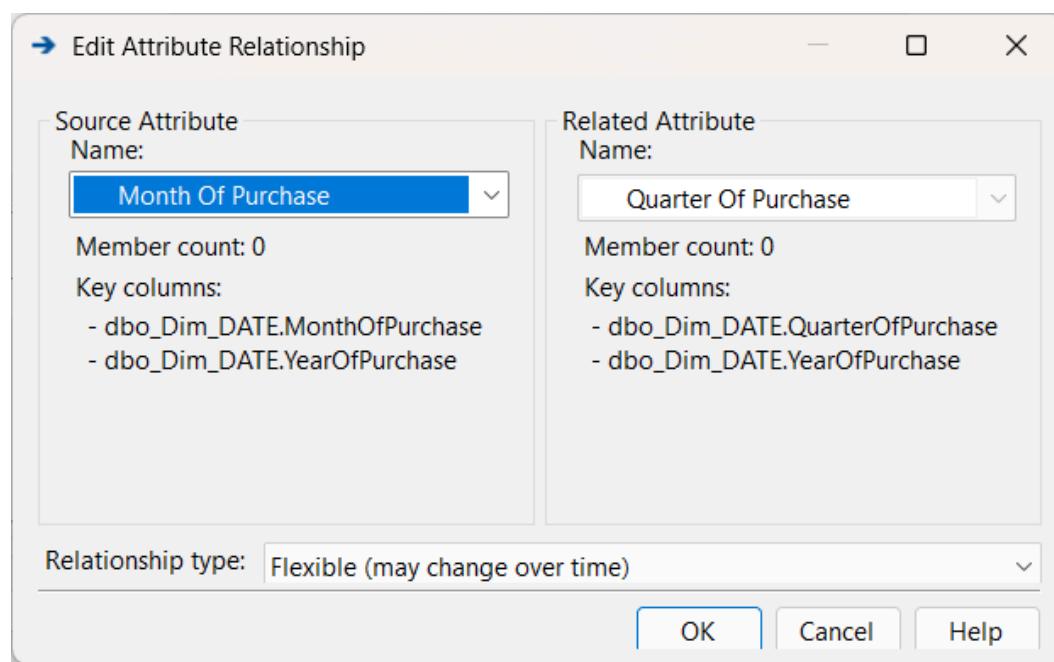
Hình 394. Chuyển sang tab Attribute Relationship của Dim DATE



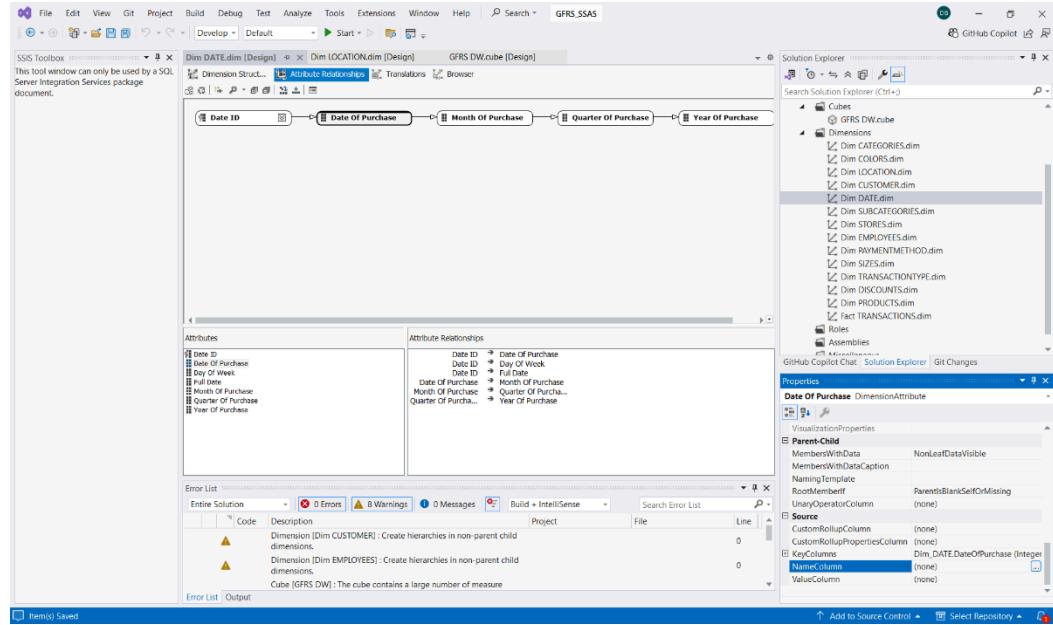
Hình 395. Điều chỉnh thuộc tính của Quarter Of Purchase



Hình 396. Cửa sổ Edit Relationship



Hình 397. Cửa sổ Edit Attribute Relationship sau khi chỉnh sửa



Hình 398. Attribute Relationship sau khi sửa đổi

- **Bước 3:** Ta chọn vào một cột sau đó kéo phần **Properties** xuống **Source**. Ở mục **KeyColumns** chọn (...). Cửa sổ **Key Columns** sẽ hiện lên, ta chọn cột bên khung **Available Columns** sau đó ấn > để đẩy sang **Key Columns**, cuối cùng nhấn **OK** để hoàn thành. Tiếp đó ta ấn (...) ở mục **NameColumn**, cửa sổ **NameColumn** sẽ hiện lên, ta chọn tên cột sau đó ấn **OK**. Các **KeyColumns** và **NameColumn** như sau:

#### o **DateOfWeek:**

- KeyColumns: DateOfWeek
- NameColumn: DateOfWeek

#### o **DateOfPurchase**

- KeyColumns: DateOfPurchase, DateID
- NameColumn: DateOfPurchase

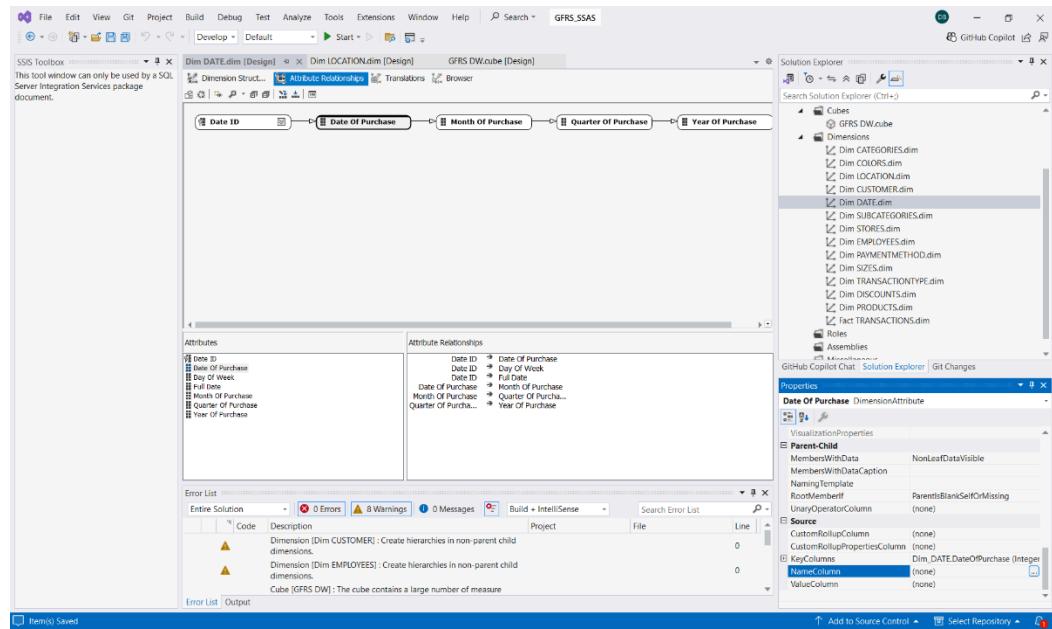
#### o **MonthOfPurchase**

- KeyColumns: MonthOfPurchase, YearOfPurchase
- NameColumn: MonthOfPurchase

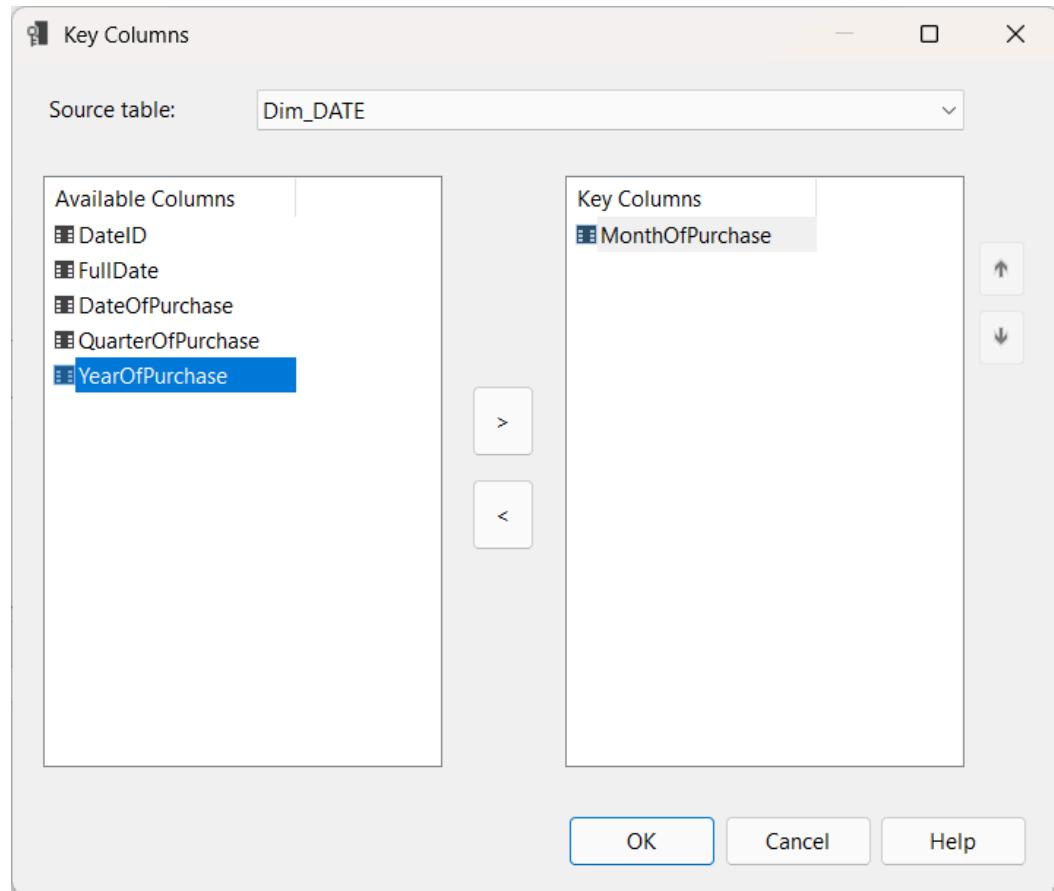
#### o **QuarterOfPurchase**

- KeyColumns: QuarterOfPurchase, YearOfPurchase

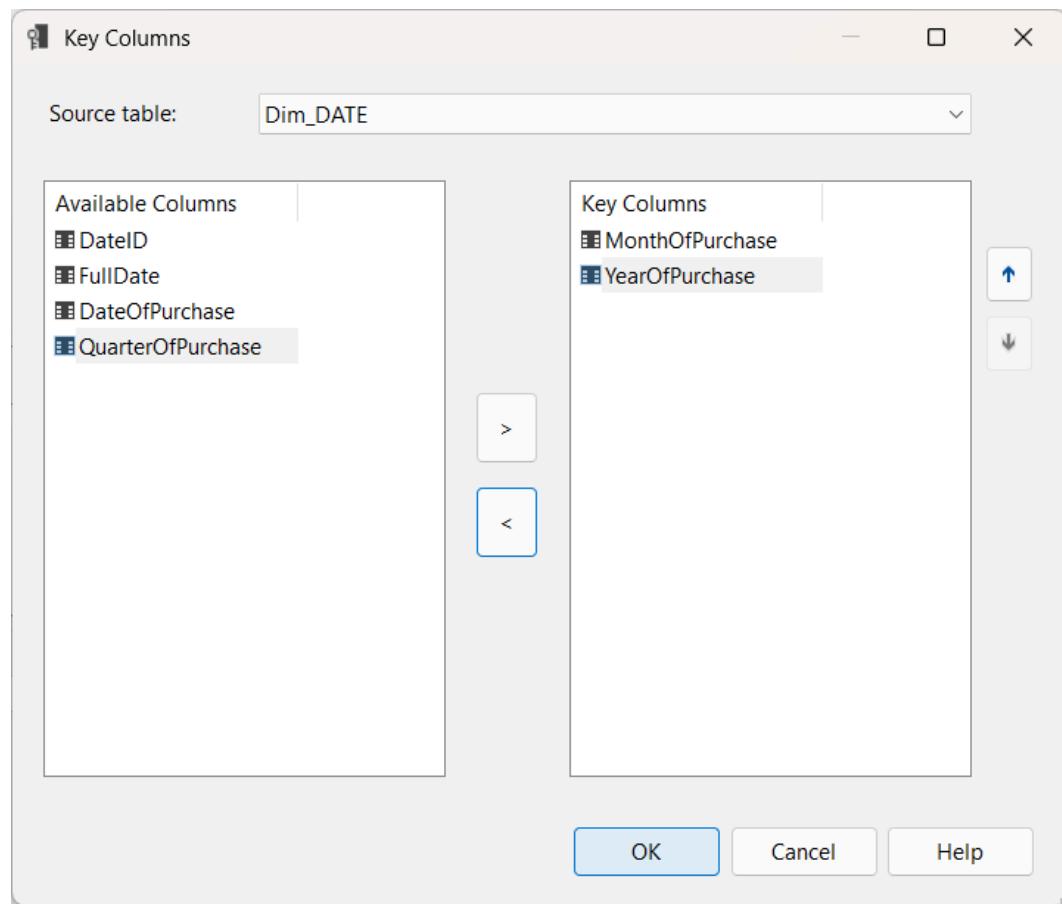
- NameColumn: QuarterOfPurchase



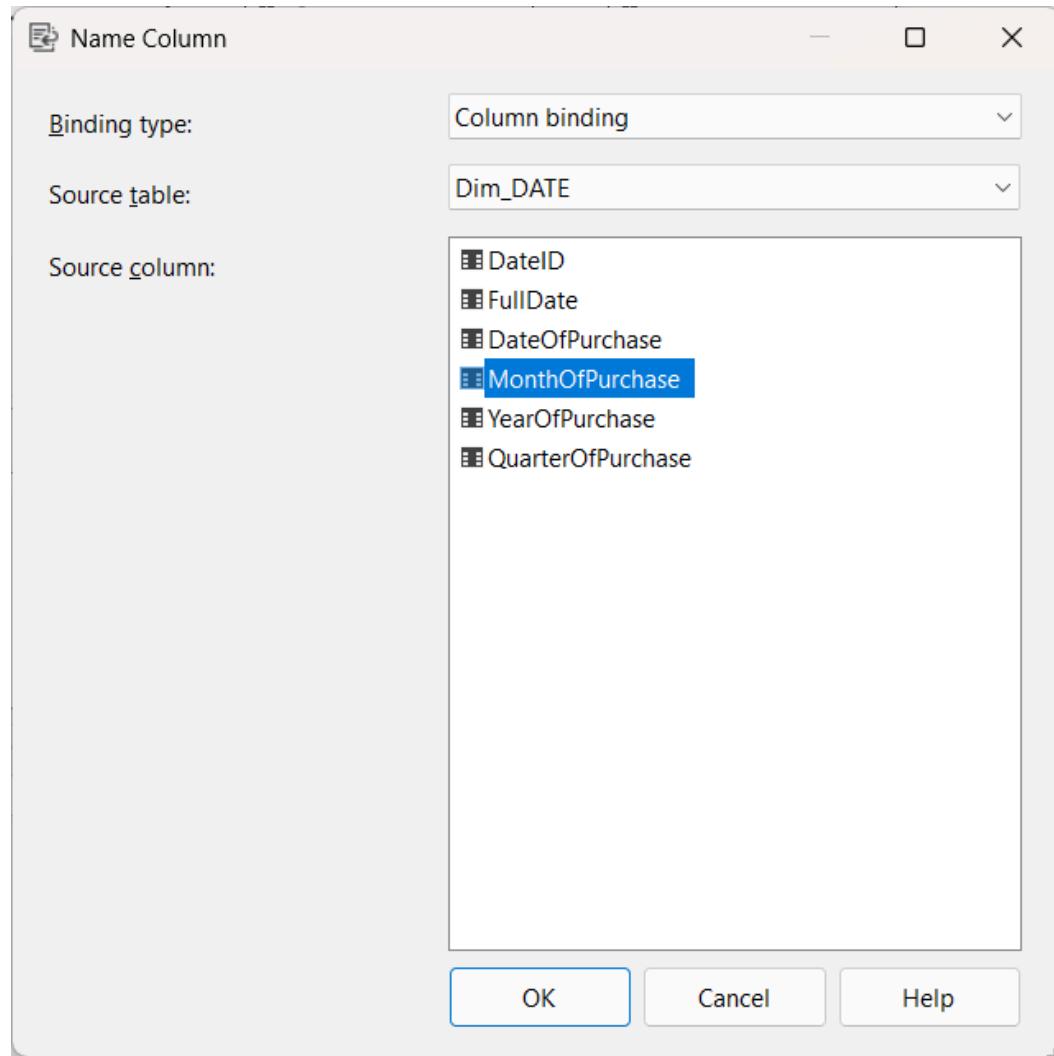
Hình 399. Mở KeyColumns của MonthOfPurchase



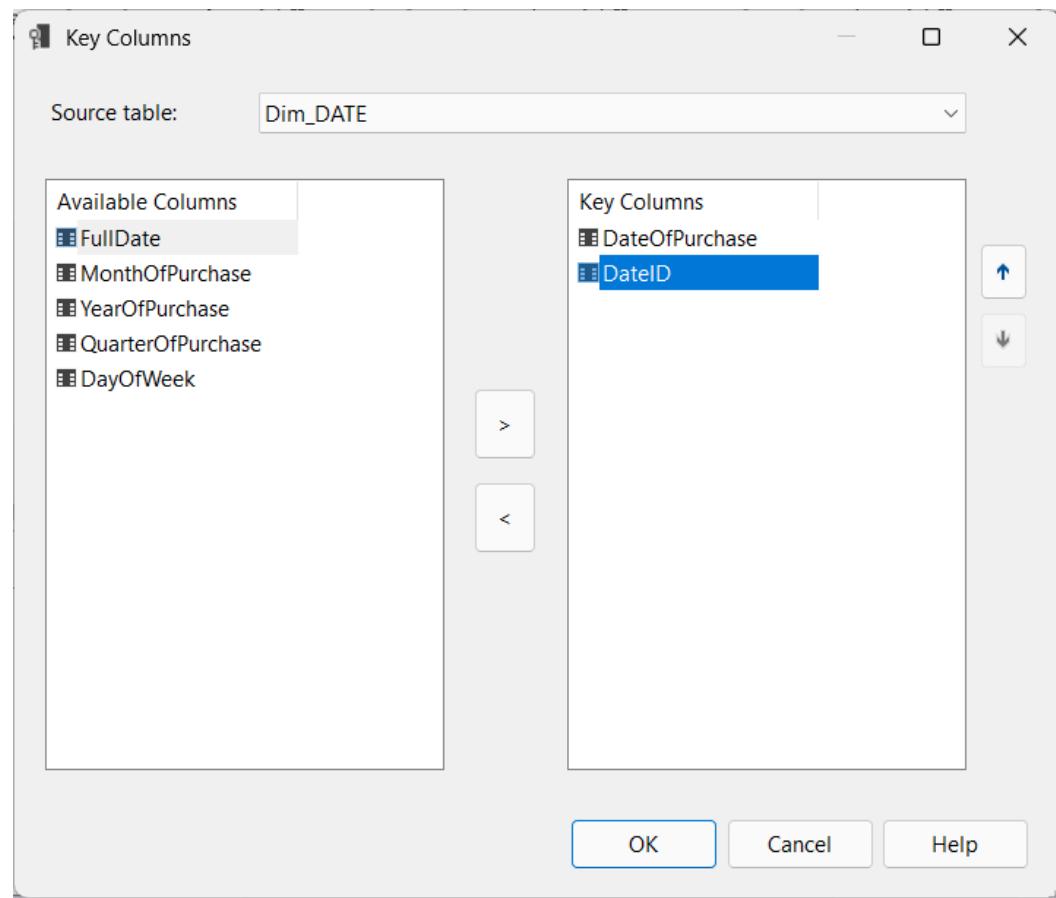
Hình 400. Chọn thuộc tính YearOfPurchase muốn thêm vào Key Columns



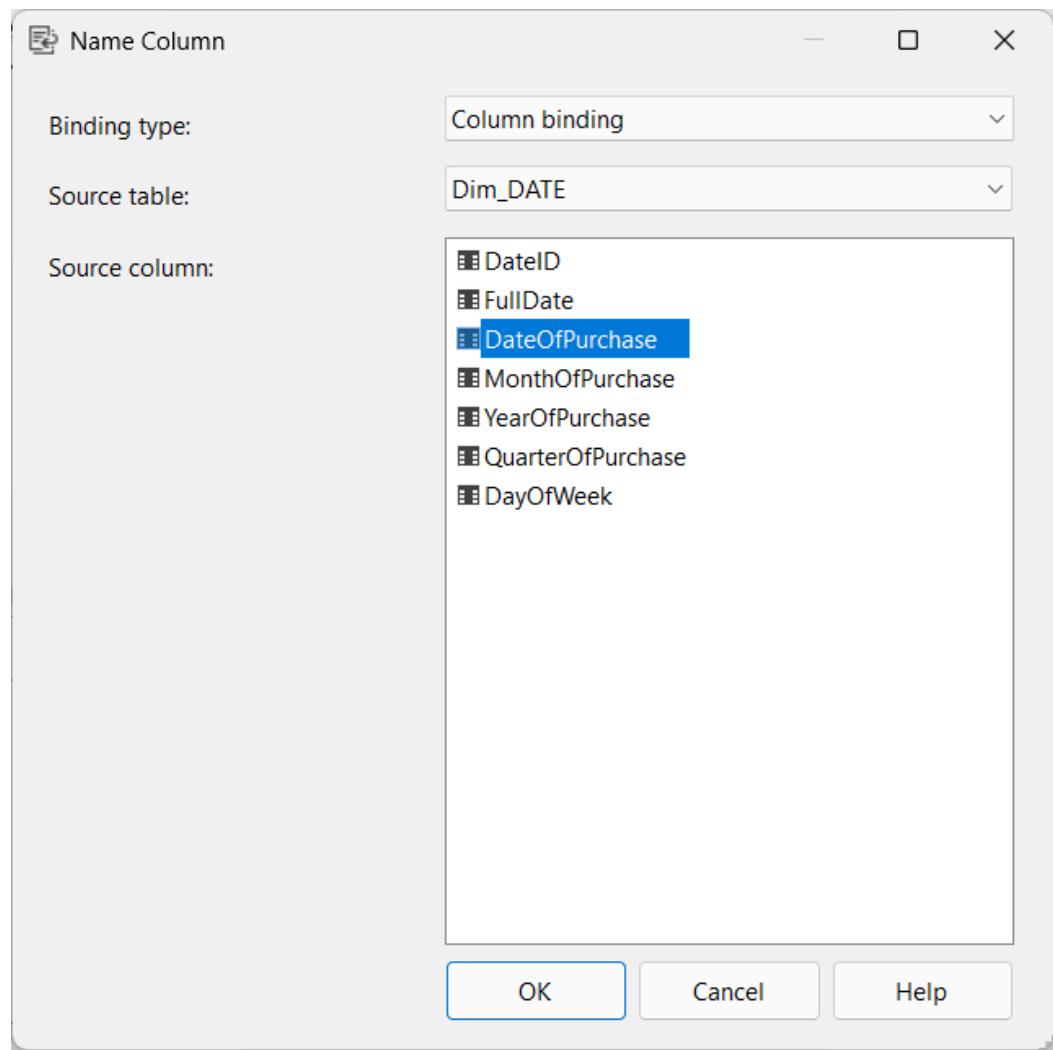
Hình 401. Thêm KeyColumns từ Available Columns cho MonthOfPurchase



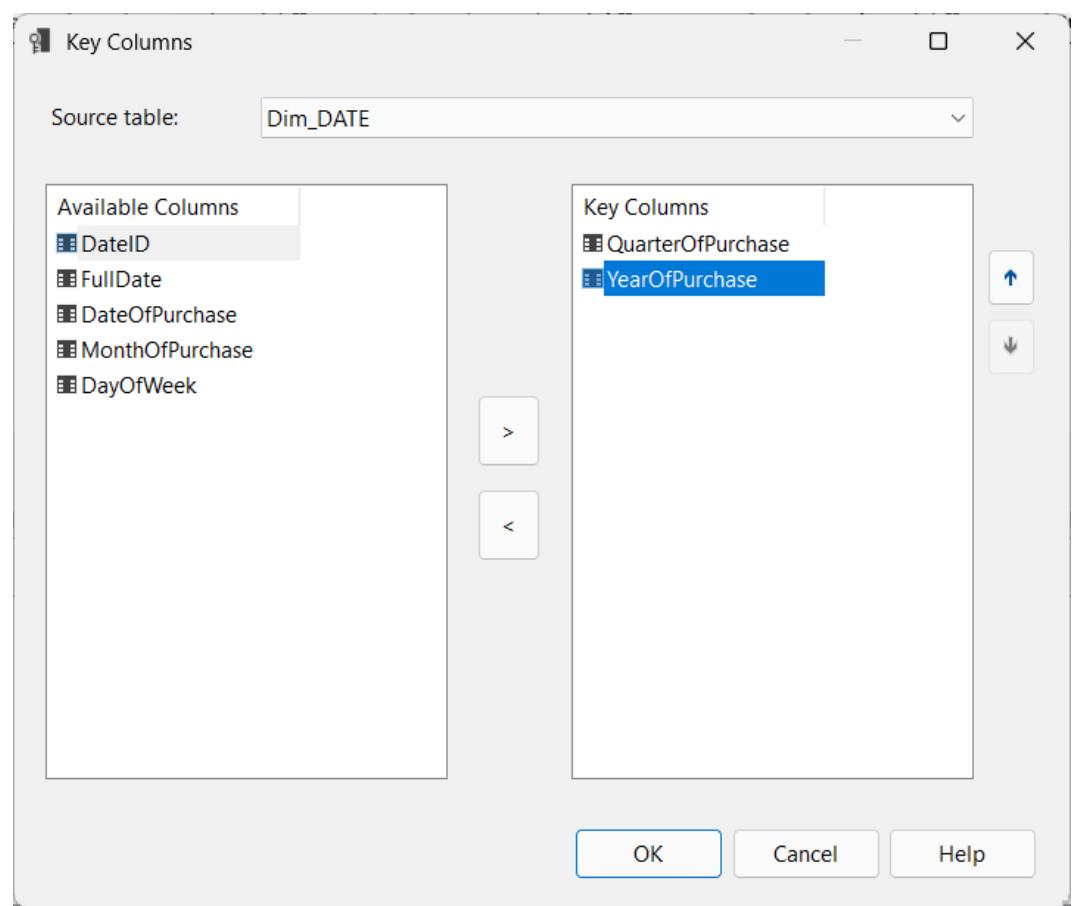
Hình 402. Chọn Name Column cho MonthOfPurchase



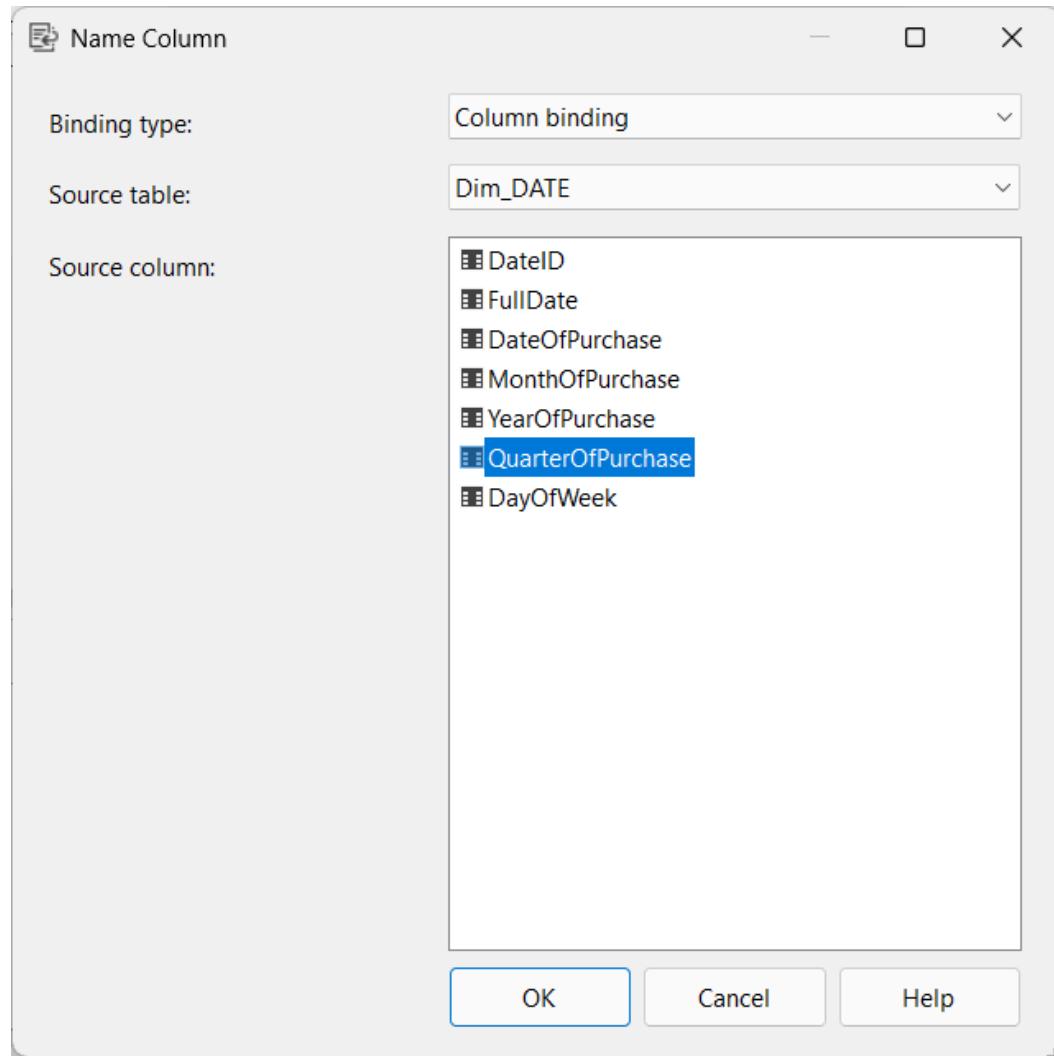
Hình 403. Thêm thuộc tính DateID vào Key Columns



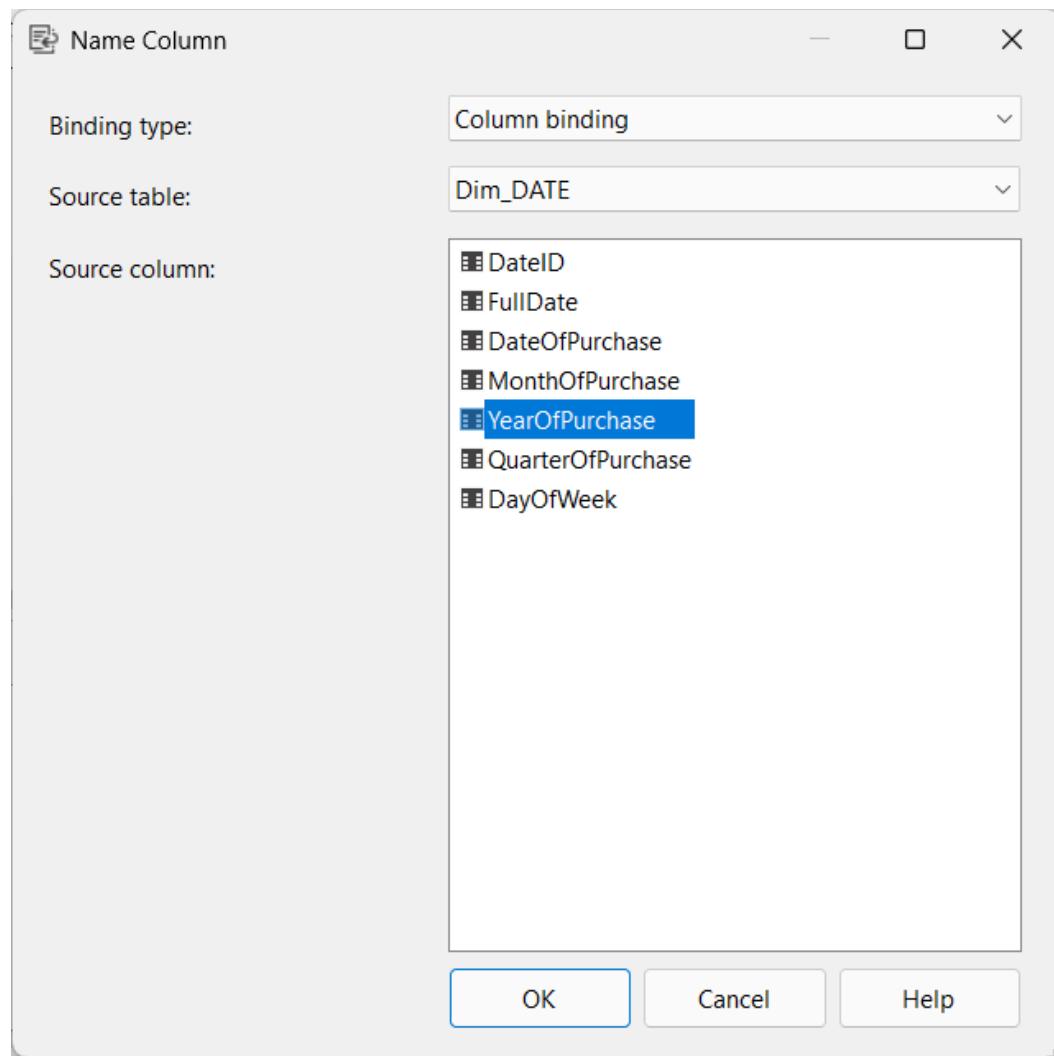
Hình 404. Chọn tên cho Name Column cho thuộc tính DateOfPurchase



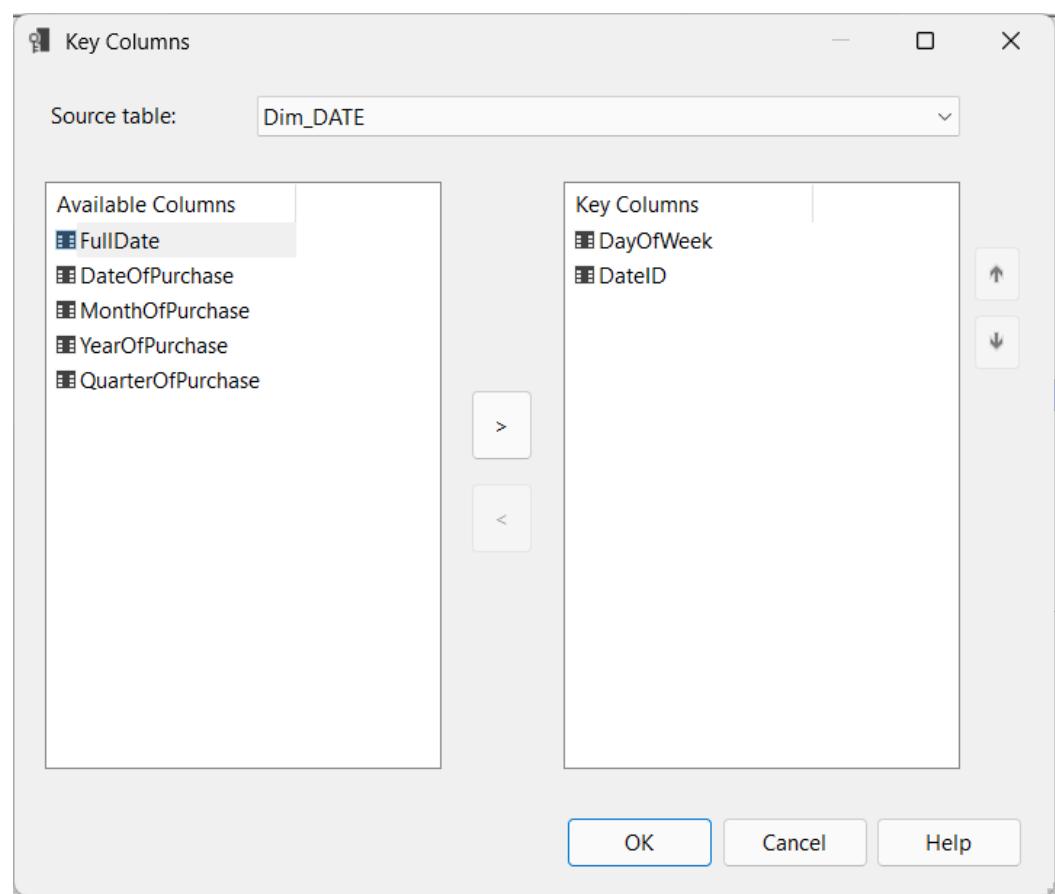
Hình 405. Thêm thuộc tính YearOfPurchase vào Key Columns



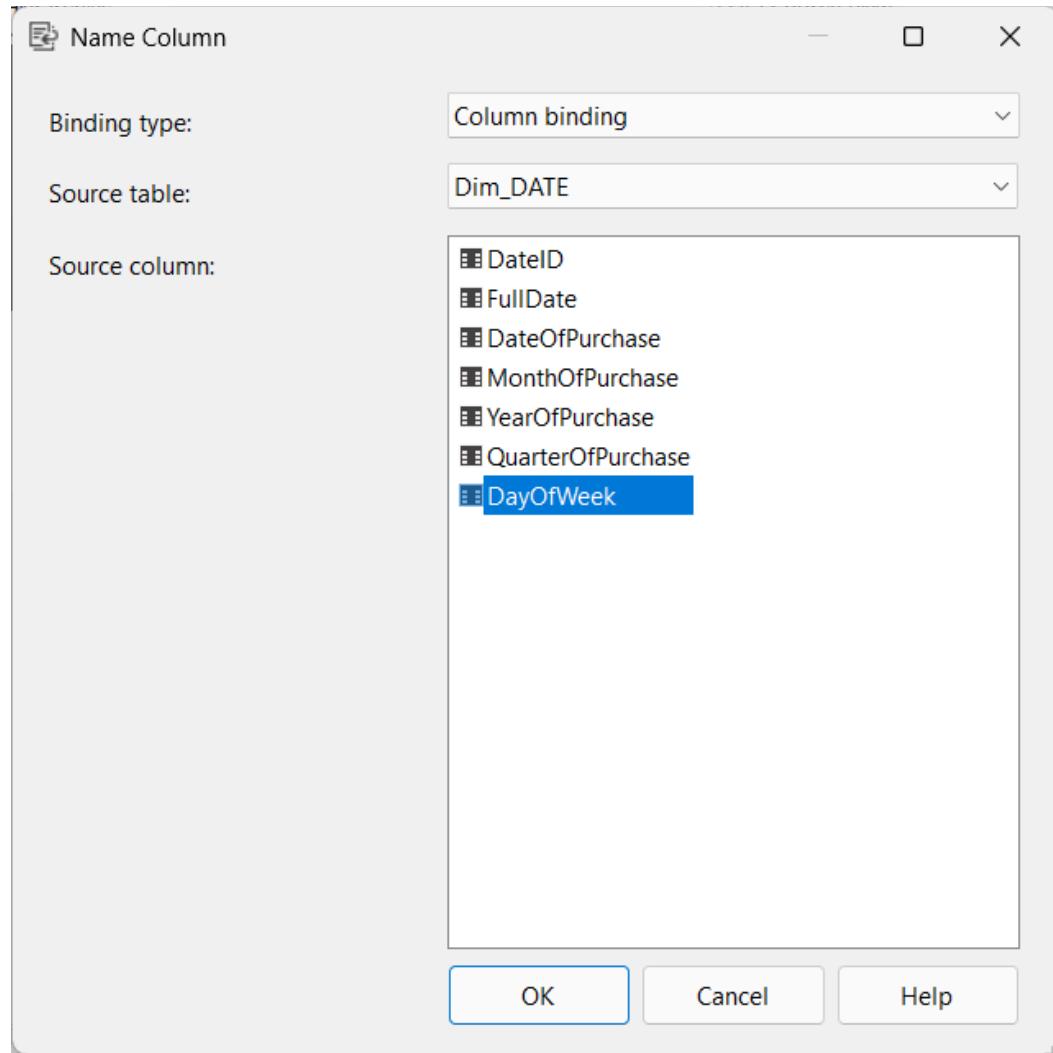
Hình 406. Chọn tên cho Name Column cho thuộc tính QuarterOfPurchase



Hình 407. Chọn tên cho Name Column cho thuộc tính YearOfPurchase



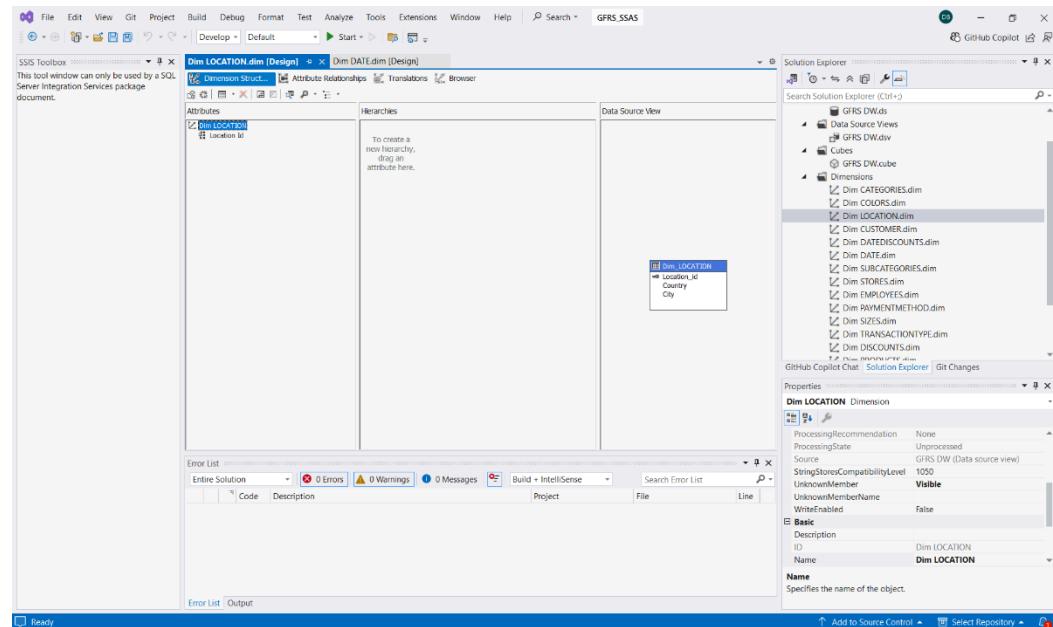
Hình 408. Thêm thuộc tính DateID vào Key Columns



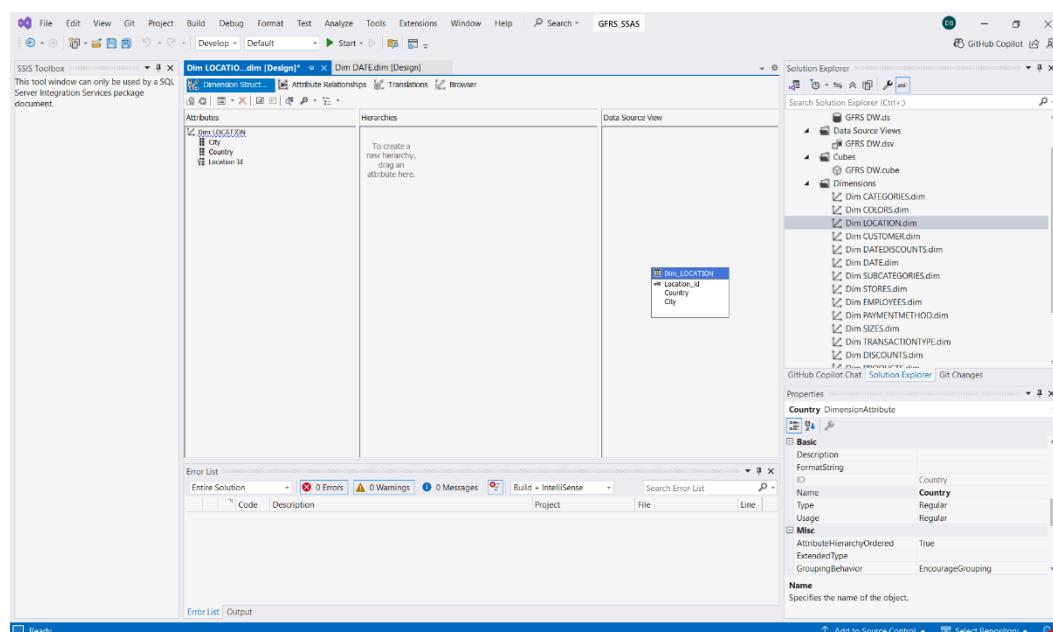
Hình 409. Chọn tên cho Name Column cho thuộc tính DateOfWeek

#### **III.2.1.4.5. Cài đặt và phân cấp dữ liệu cho Dim\_LOCATION**

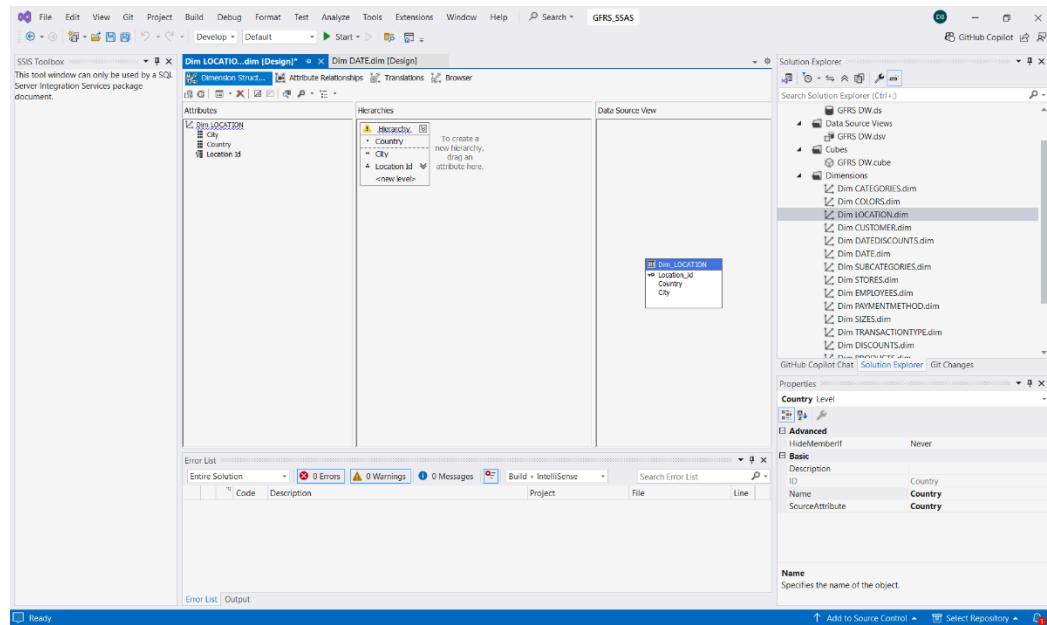
- **Bước 1:** Mở **Dim\_LOCATION**, vào tab **Dimension Structure**. Ta kéo các cột dữ liệu từ bảng **Data Source View** sang bảng **Attributes**, sau đó kéo các cột này từ **Attributes** vào **Hierachies**. Sau đó sắp xếp theo thứ tự từ trên xuống: **COUNTRY, CITY, LOCATIONID**



Hình 410. Cửa sổ Dimension Structure của Dim\_LOCATION

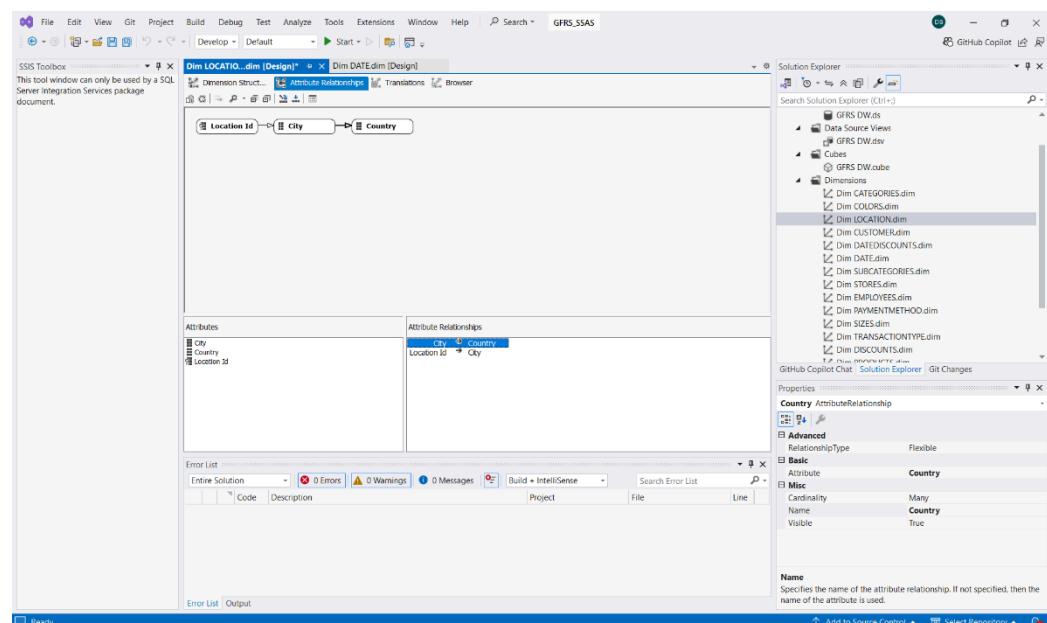


Hình 411. Kéo dữ liệu qua Attributes



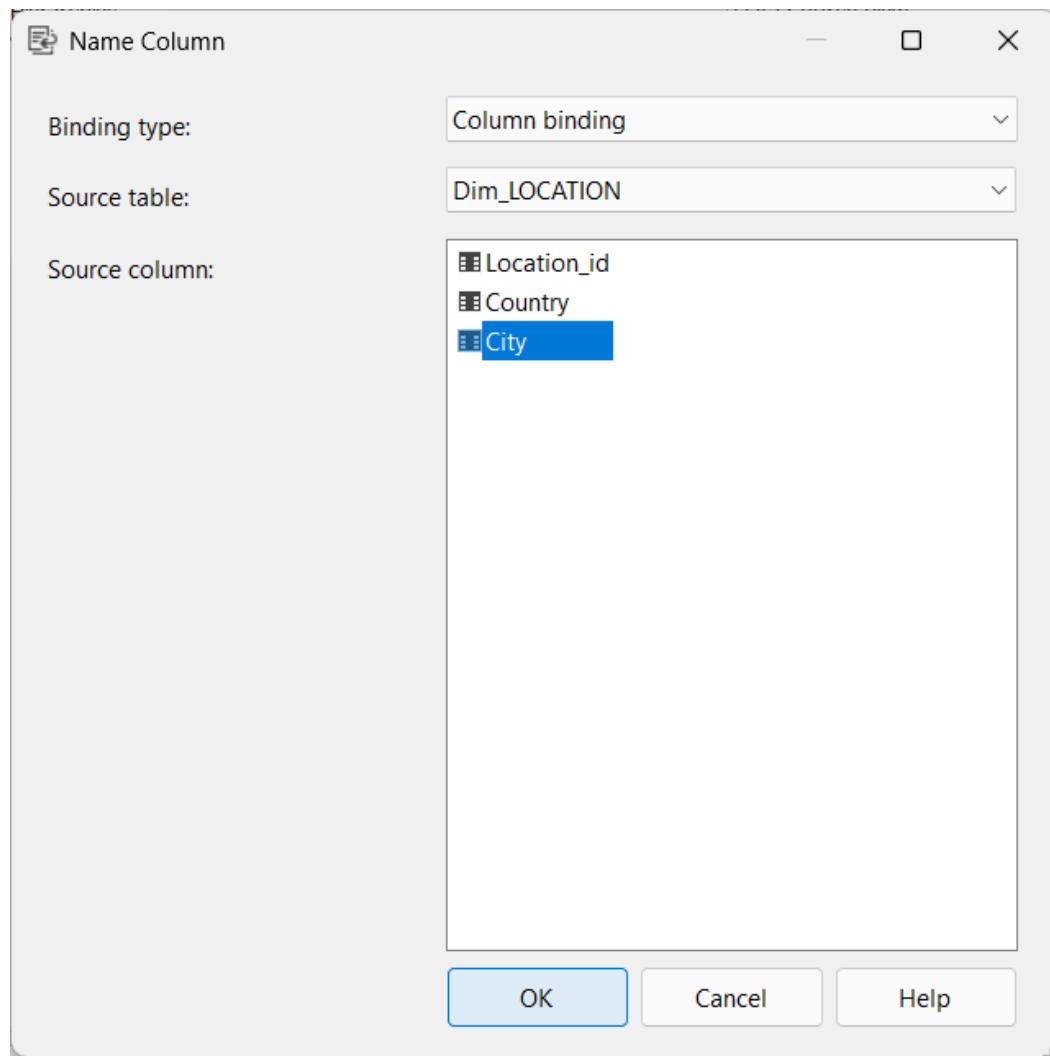
Hình 412. Phân cấp dữ liệu theo thứ tự

- **Bước 2:** Chuyển sang tab **Attribute Relationship**, nhấp chuột phải vào thuộc tính trong khung **Attribute Relationship** chọn **Edit Attribute Relationship...** của **Location ID**  **Country**. Cửa sổ **Edit Attribute Relationship** sẽ hiện lên, ta đổi **Name** phần **Source Attribute** thành **City**. Sau đó chọn **OK**.

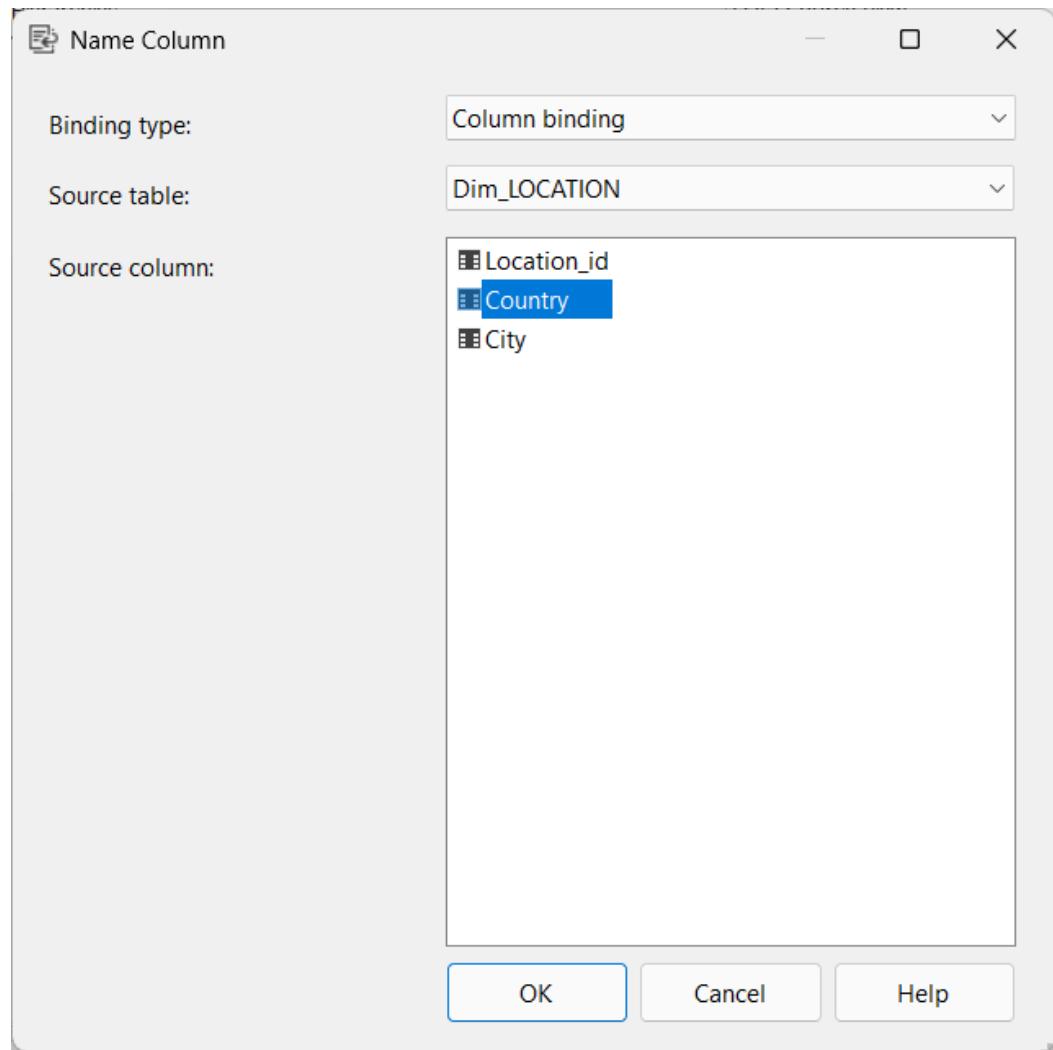


Hình 413. Dim\_LOCATION sau khi phân cấp

- **Bước 3:** Nhấn chuột vào **City** trên khung **Attributes**. Ở phần **Properties** ta kéo xuống phần **NameColumn** nhấn chuột vào (...), cửa sổ **Name Column** hiện lên, ta chọn **City** sau đó nhấn **OK**. Làm tương tự với **LocationID** và **Country** với **NameColumn** chính là tên thuộc tính.



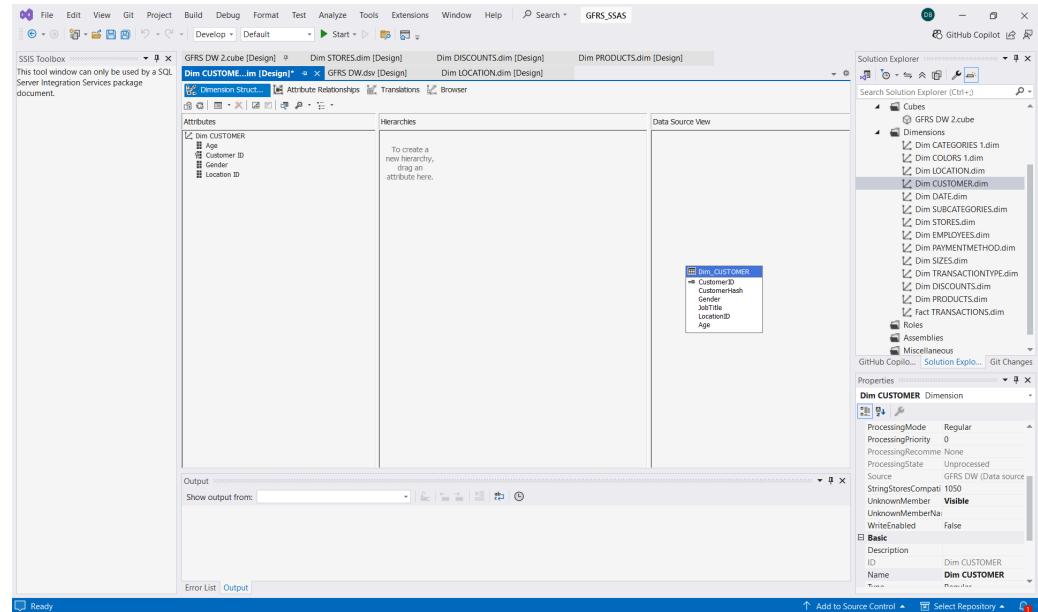
Hình 414. Chọn tên cho Name Column cho thuộc tính City



Hình 415. Chọn tên cho Name Column cho thuộc tính Country

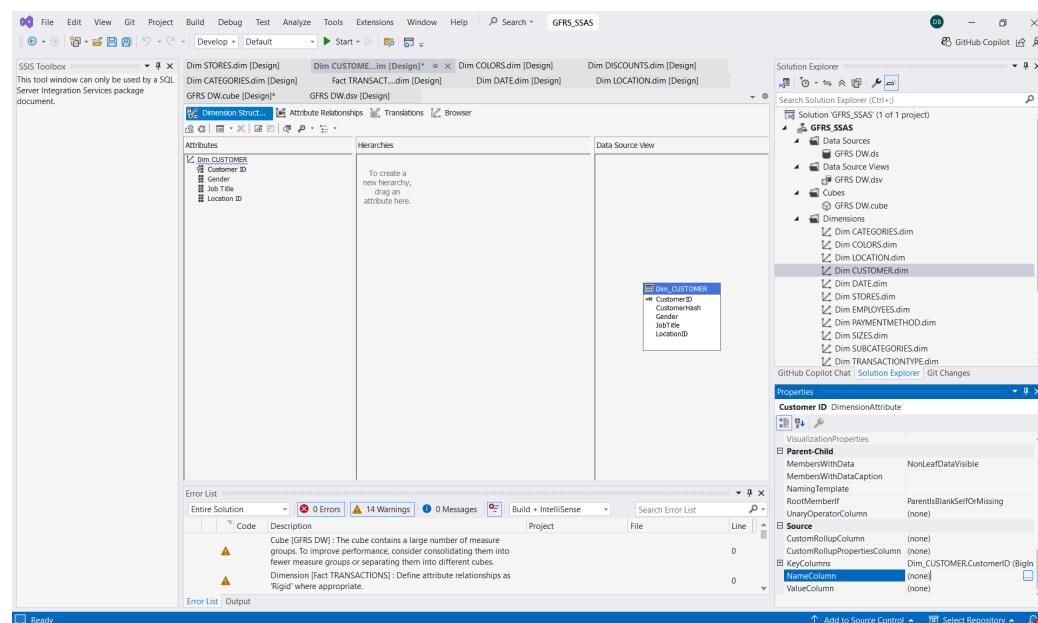
#### III.2.1.4.6. Cài đặt Dim\_CUSTOMERS

- **Bước 1:** Kéo thuộc tính **Gender, Age** từ khung **Data Source View** sang khung **Attributes**.

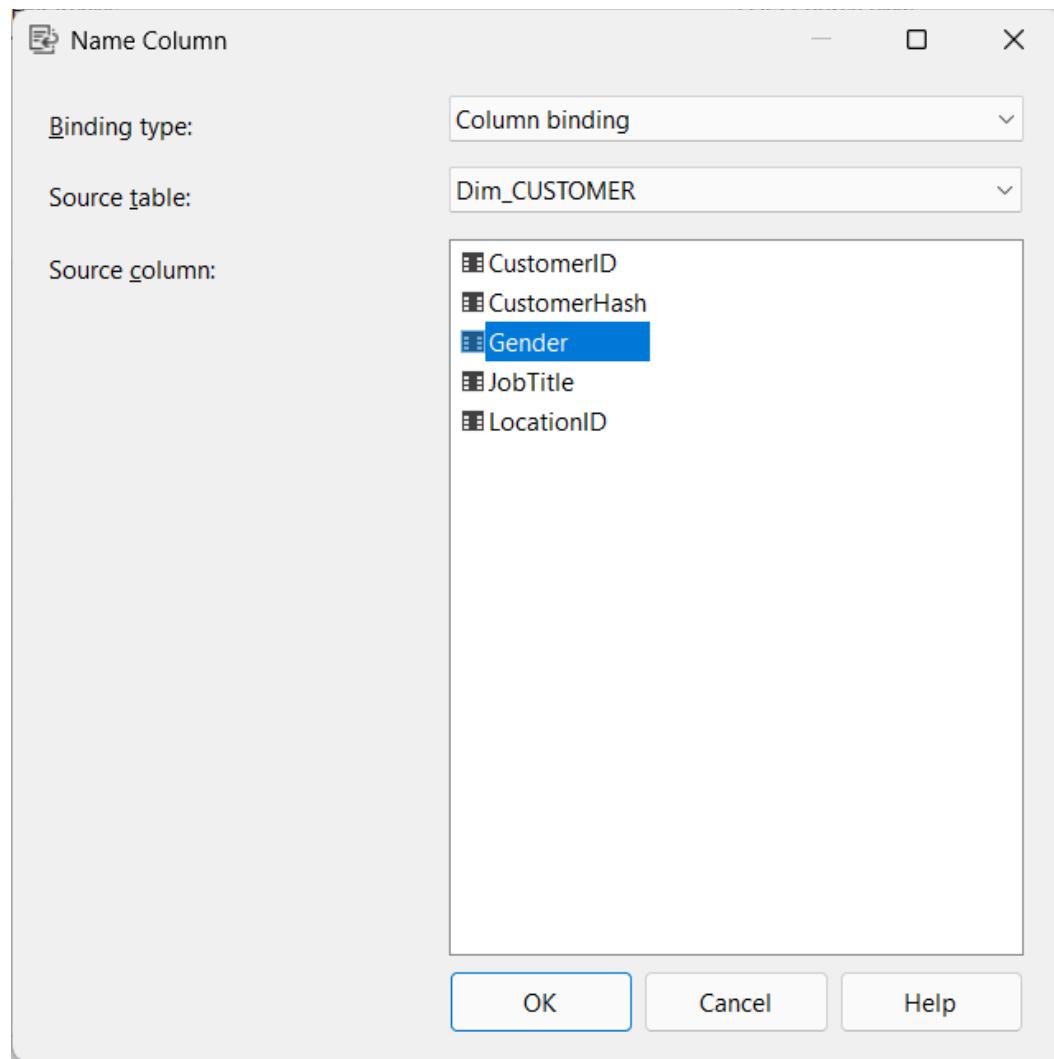


Hình 416. Kéo thuộc tính Gender, JobTitle, Age sang khung Attributes

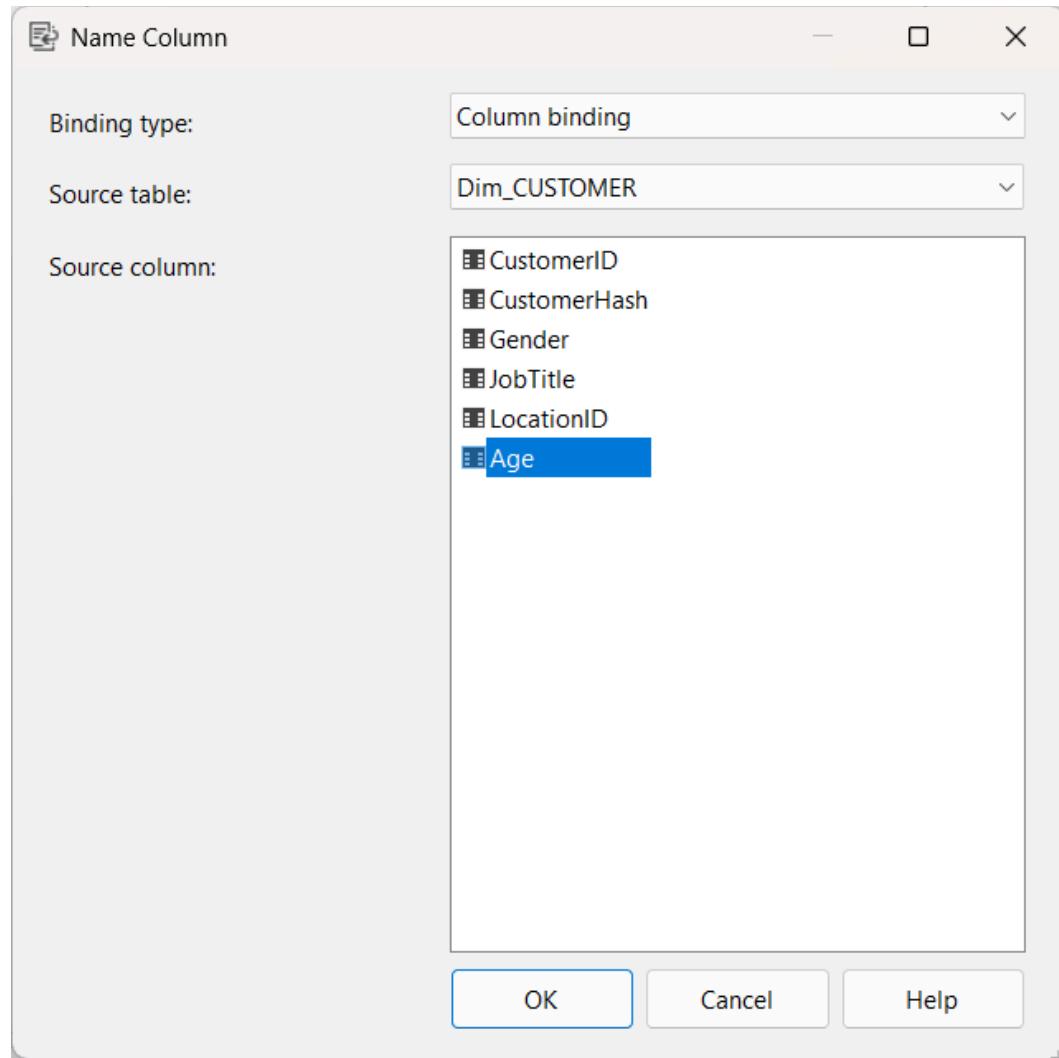
- **Bước 2:** Nhấn chuột vào **Gender** trên khung **Attributes**. Ở phần **Properties** ta kéo xuống phần **NameColumn** nhấn chuột vào (...), cửa sổ **Name Column** hiện lên, ta chọn **Gender** sau đó nhấn **OK**. Làm tương tự với các thuộc tính **Age** với **NameColumn** tương ứng.



Hình 417. Thêm NameColumn cho thuộc tính Gender



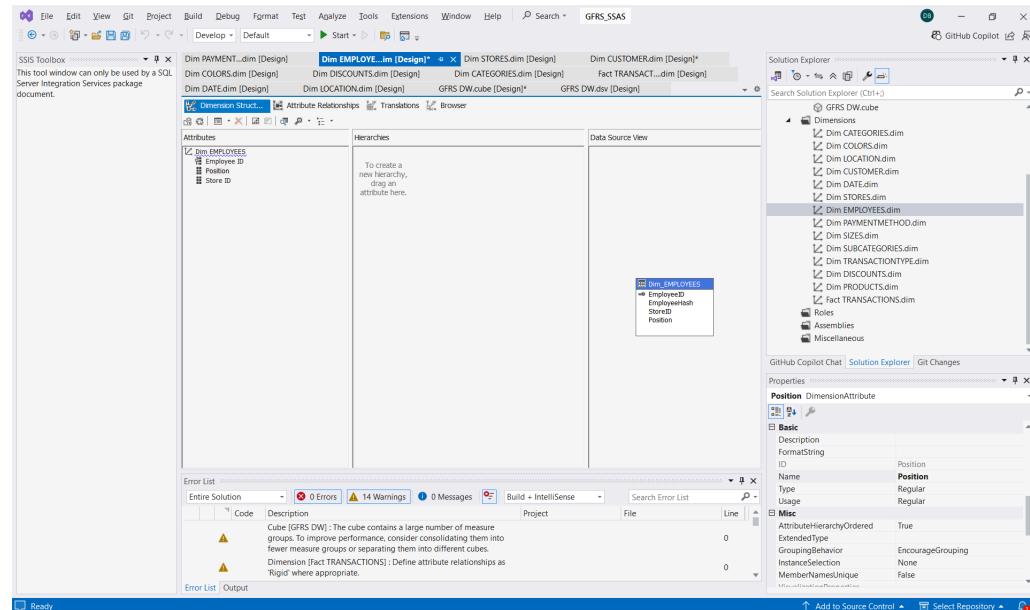
Hình 418. Chọn tên cho Name Column cho thuộc tính Gender



Hình 419. Chọn tên cho Name Column cho thuộc tính Age

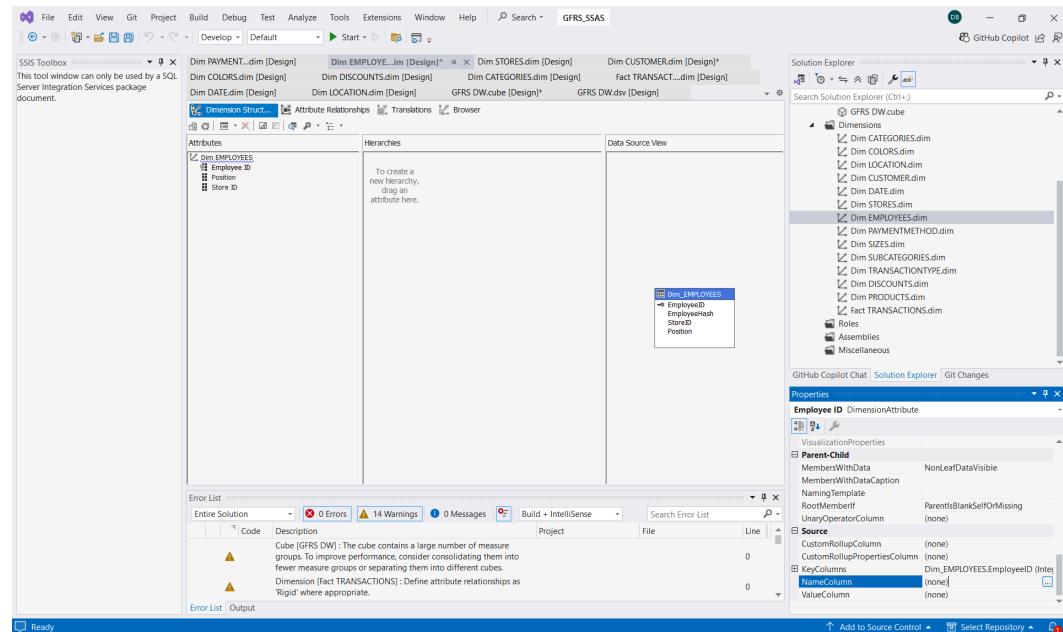
#### **III.2.1.4.7. Cài đặt Dim\_EMPLOYEES**

- **Bước 1:** Kéo thuộc tính **Position** từ khung **Data Source View** sang khung **Attributes**.

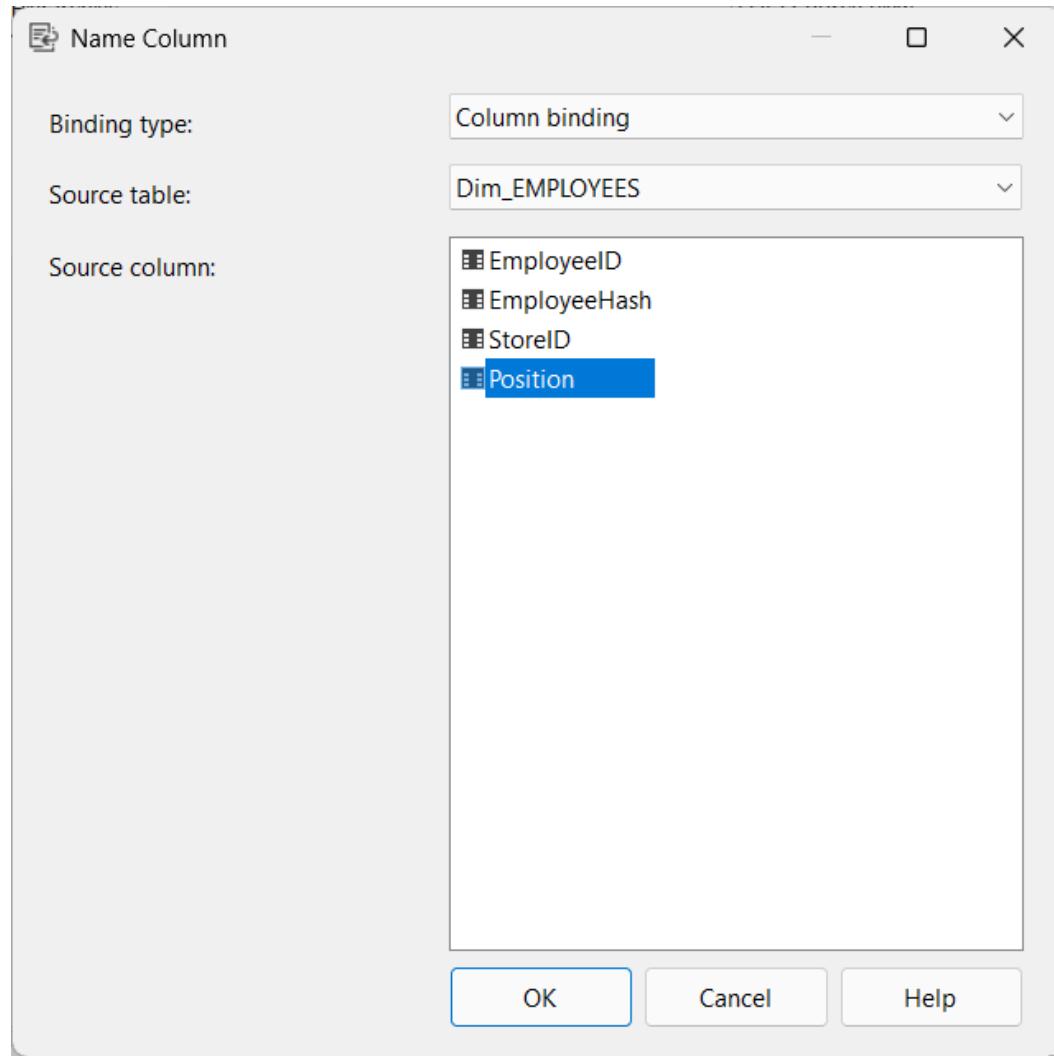


Hình 420. Kéo thuộc tính Position sang khung Attributes

- **Bước 2:** Nhấn chuột vào **Position** trên khung **Attributes**. Ở phần **Properties** ta kéo xuống phần **NameColumn** ta nhấp chuột vào (...), cửa sổ **Name Column** hiện lên, ta chọn **Position** sau đó nhấn **OK**.



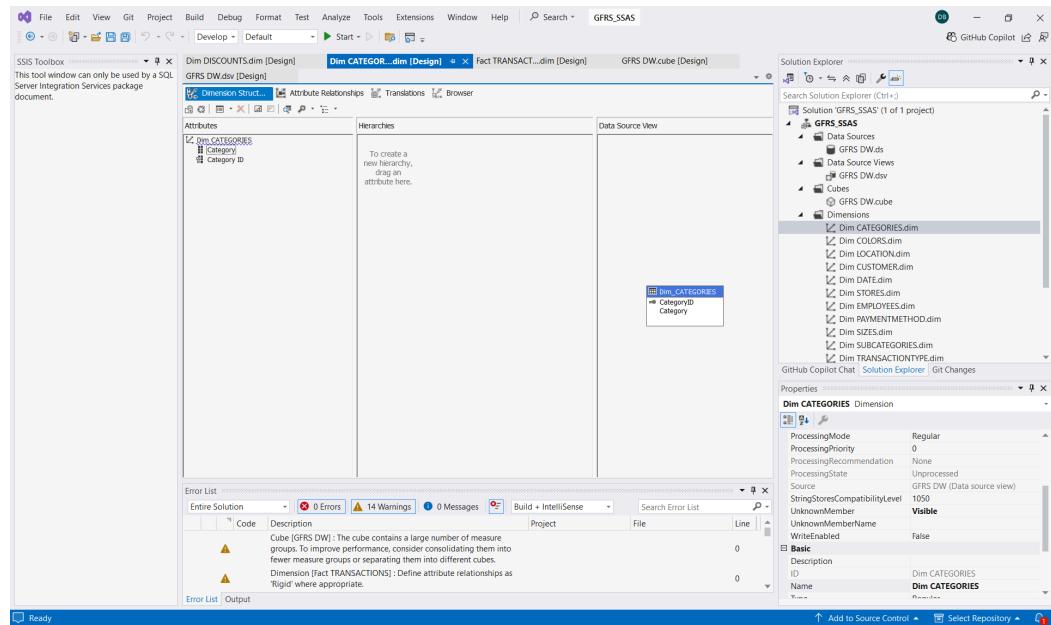
Hình 421. Thêm NameColumn cho thuộc tính Position



Hình 422. Chọn tên cho Name Column cho thuộc tính Position

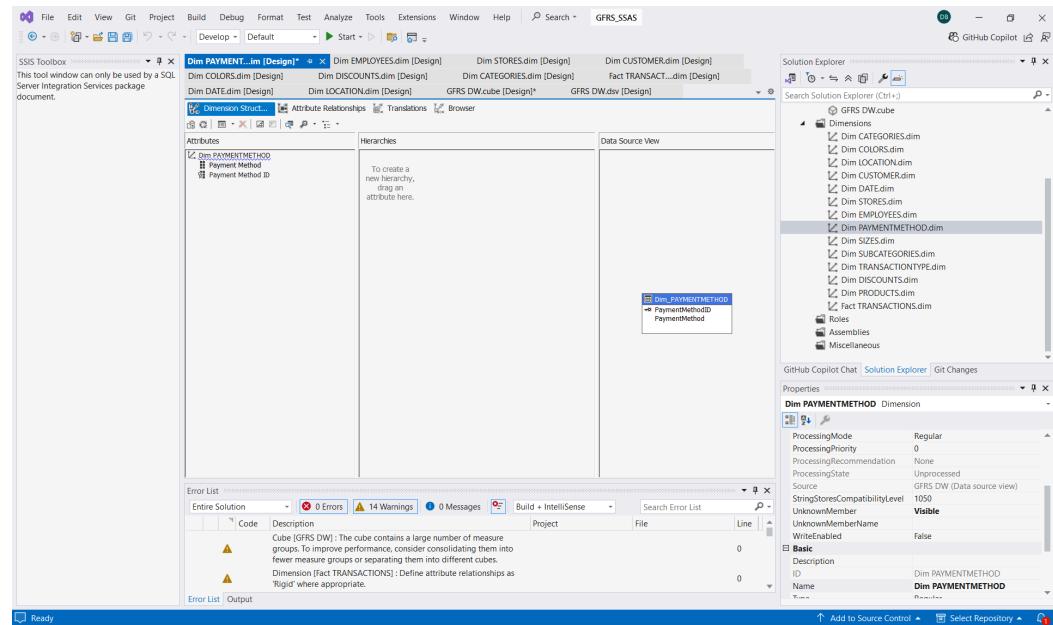
#### ***III.2.1.4.8. Cài đặt Dim\_PAYMENTMETHOD***

- **Bước 1:** Kéo thuộc tính PaymentMethod từ khung **Data Source View** sang khung **Attributes**.

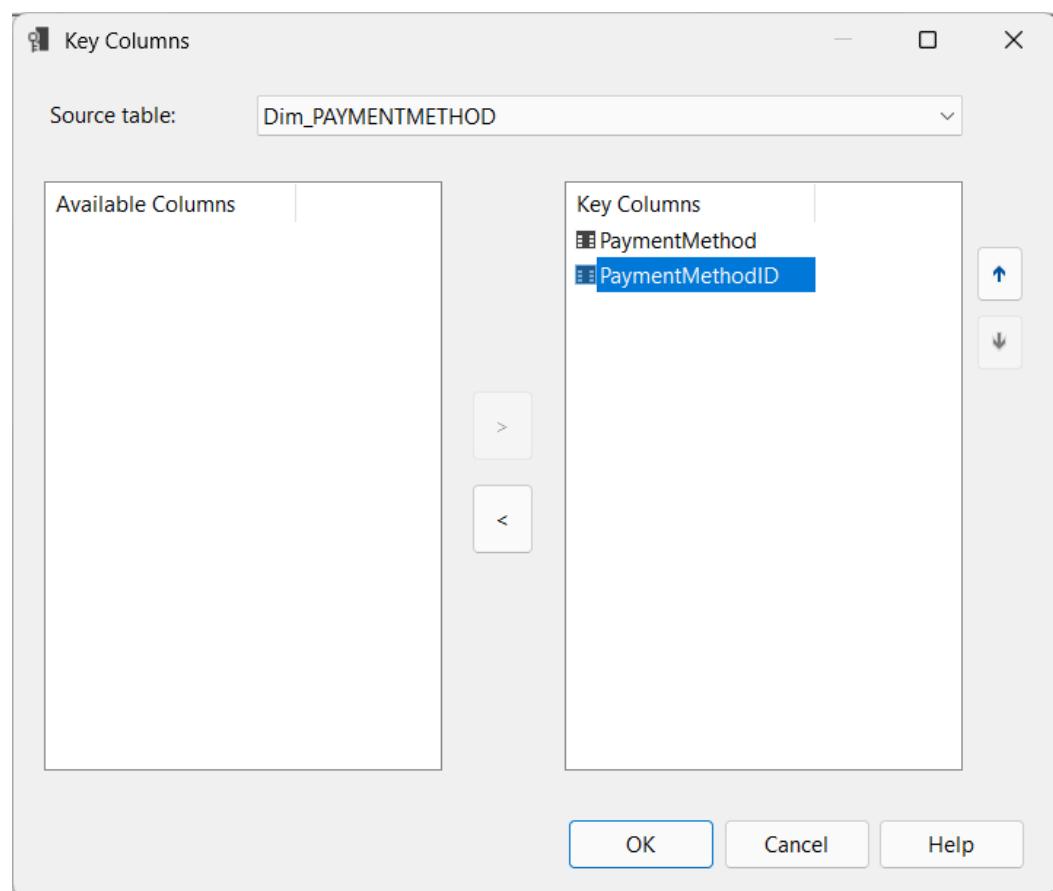


Hình 423. Kéo thuộc tính PaymentMethod sang khung Attributes

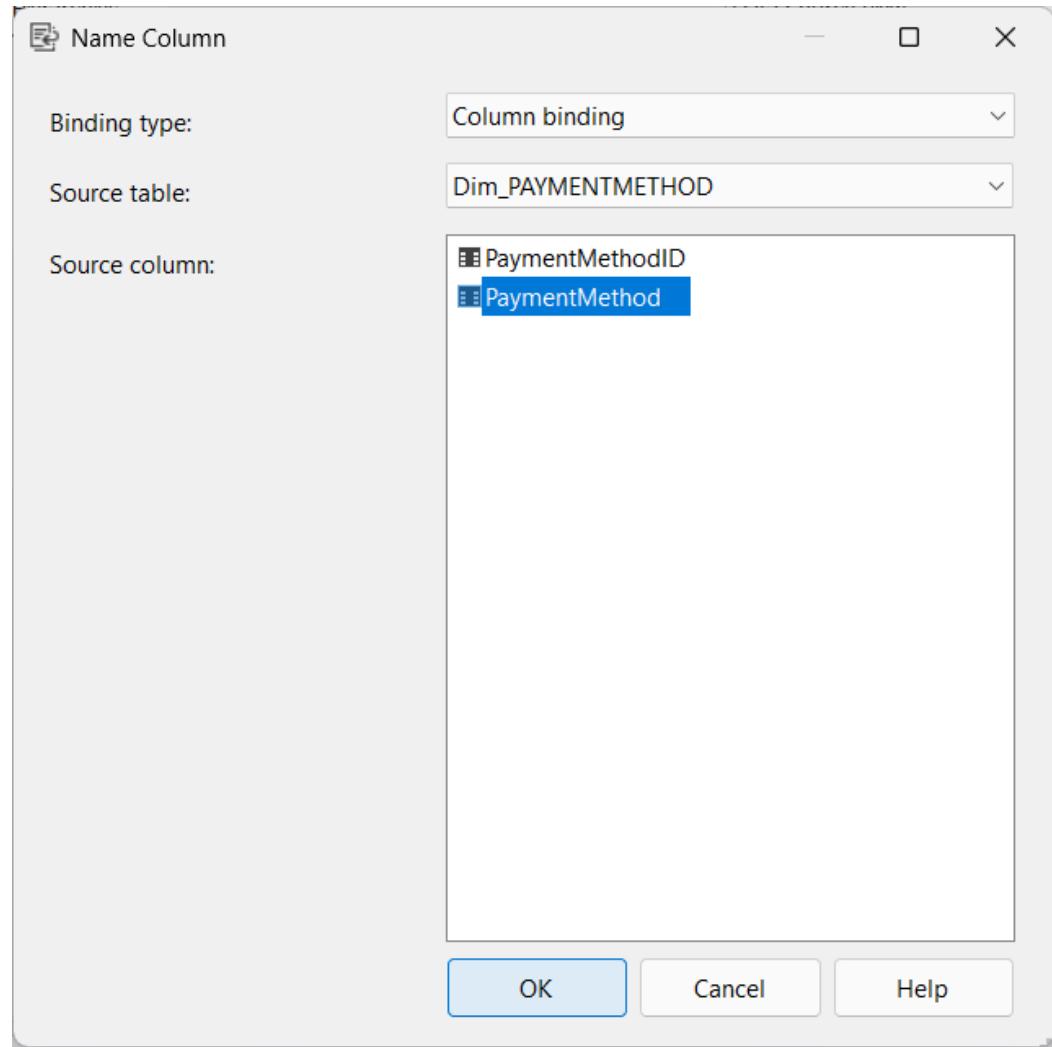
- **Bước 2:** Nhấn chuột vào **PaymentMethod** trên khung **Attributes**. Ở phần **Properties** ta kéo xuống phần **KeyColumns**, nhấn chuột vào (...) bên phải. Cửa sổ **Key Columns** hiện lên, ta chọn thuộc tính **PaymentMethodID** từ khung **Available Columns**, sau đó nhấn > đè đầy sang khung **Key Columns**, nhấn **OK** để hoàn tất. Tiếp đó, ta nhấn chuột vào (...) ở **NameColumn**, cửa sổ **Name Column** hiện lên, ta chọn **PaymentMethod** sau đó nhấn **OK**.



Hình 424. Thêm thuộc tính vào KeyColumns cho PaymentMethod



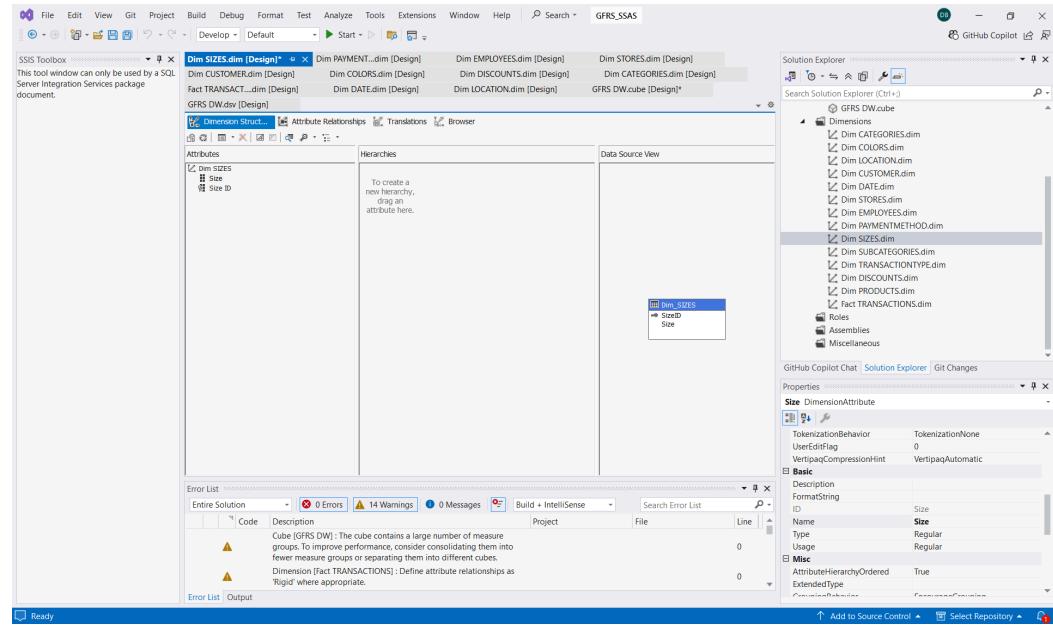
Hình 425. Thêm thuộc tính PaymentMethodID vào Key Columns



Hình 426. Chọn tên cho Name Column cho thuộc tính PaymentMethod

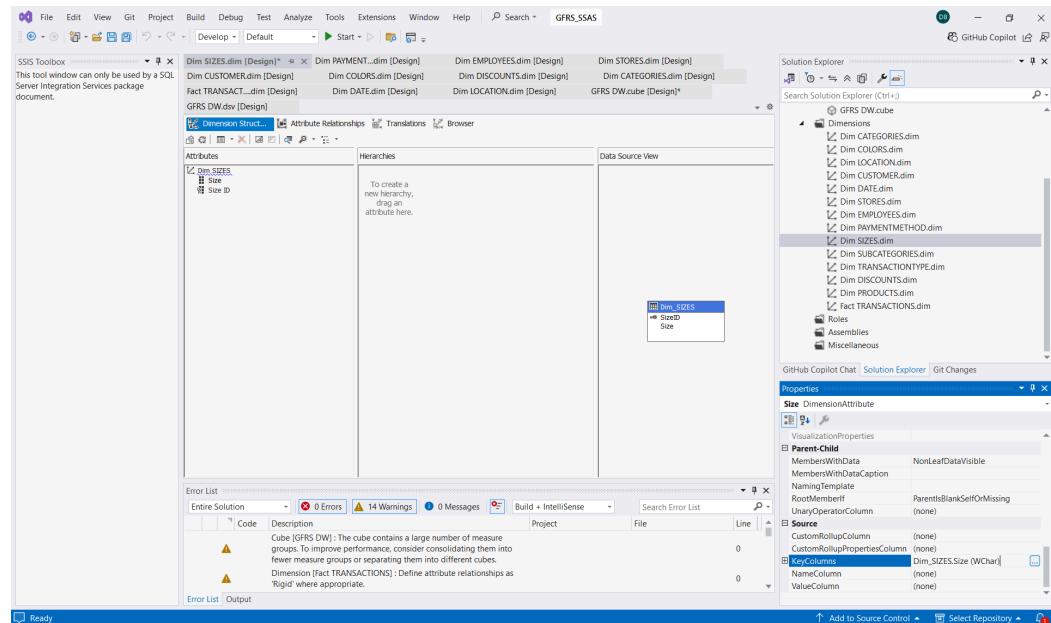
#### III.2.1.4.9. Cài đặt Dim\_SIZES

- **Bước 1:** Kéo thuộc tính Size từ khung **Data Source View** sang khung **Attributes**.

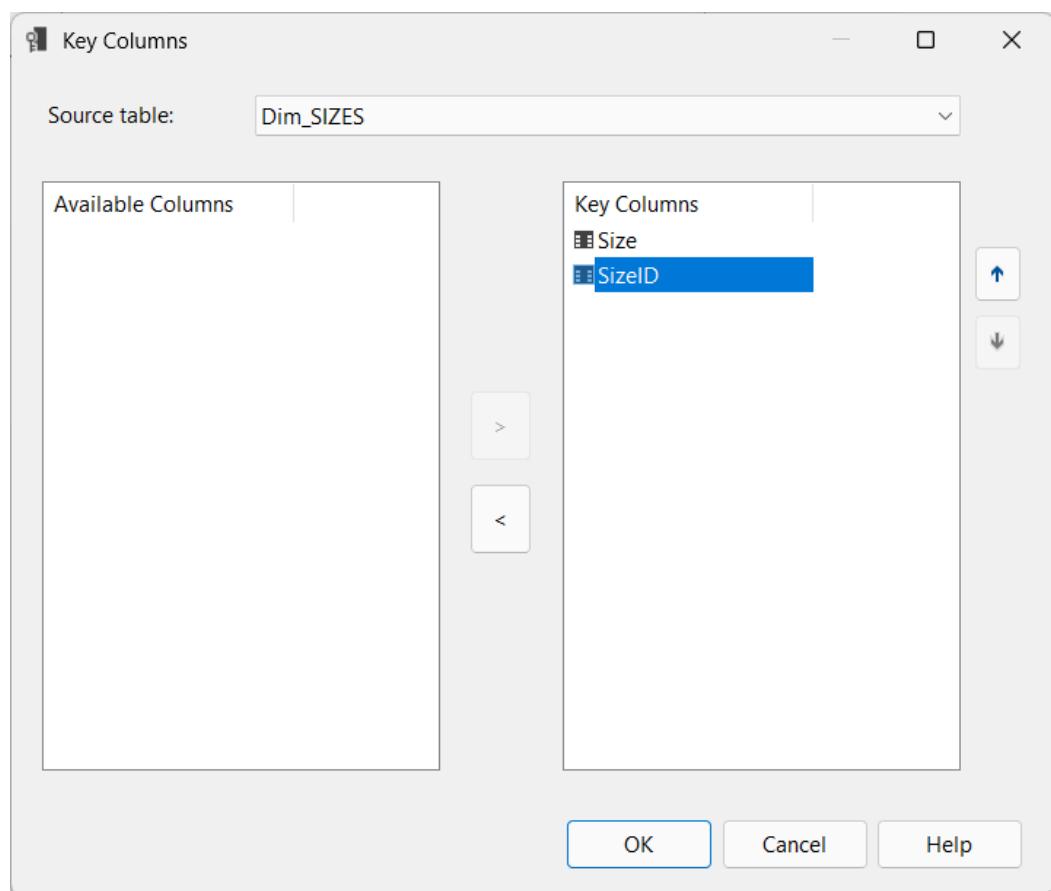


Hình 427. Kéo thuộc tính Size sang khung Attributes

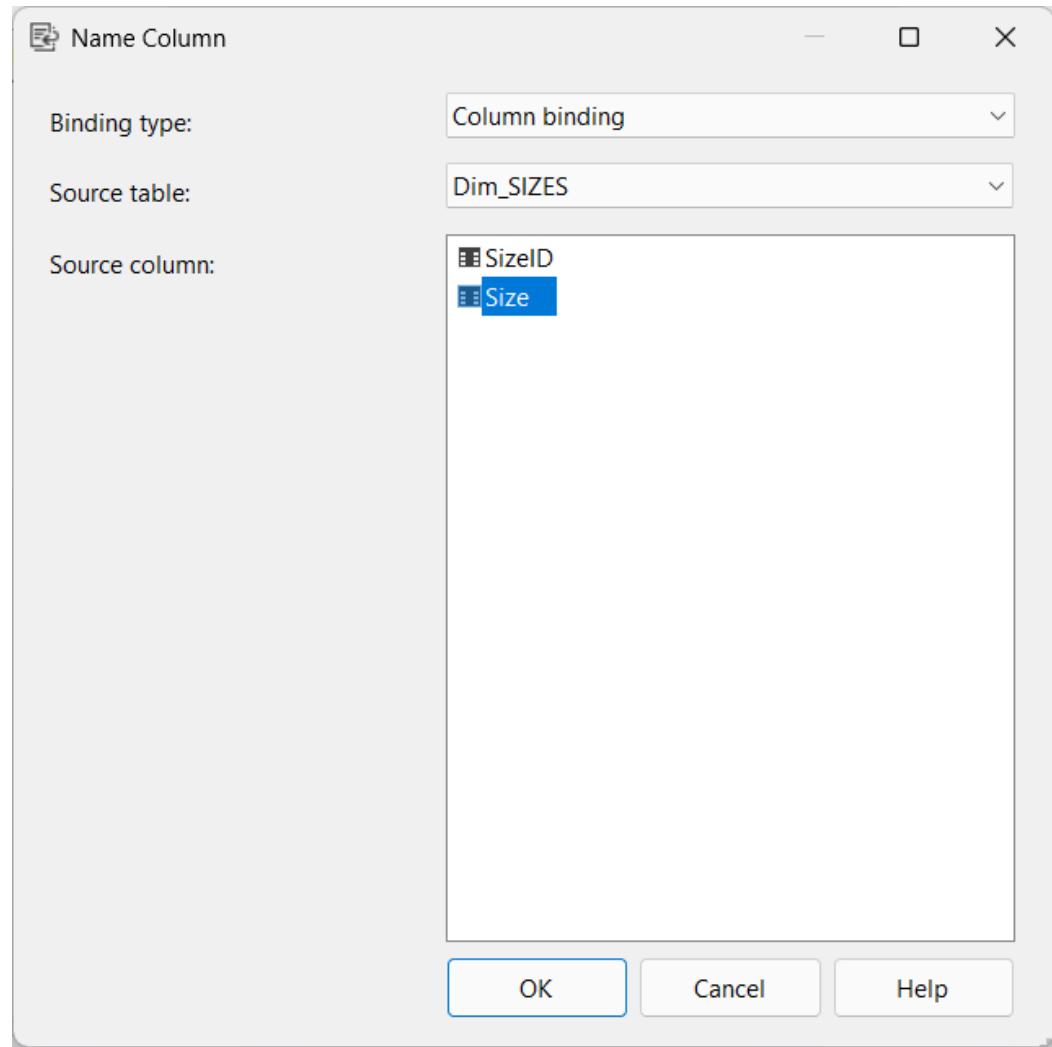
- **Bước 2:** Nhấn chuột vào **Size** trên khung **Attributes**. Ở phần **Properties** ta kéo xuống phần **KeyColumns**, nhấn chuột vào (...) bên phải. Cửa sổ **Key Columns** hiện lên, ta chọn thuộc tính **SizeID** từ khung **Available Columns**, sau đó nhấn > đè đẩy sang khung **Key Columns**, nhấn **OK** để hoàn tất. Tiếp đó, ta nhấn chuột vào (...) ở **NameColumn**, cửa sổ **Name Column** hiện lên, ta chọn **Size** sau đó nhấn **OK**.



Hình 428. Thêm thuộc tính vào KeyColumns cho Size



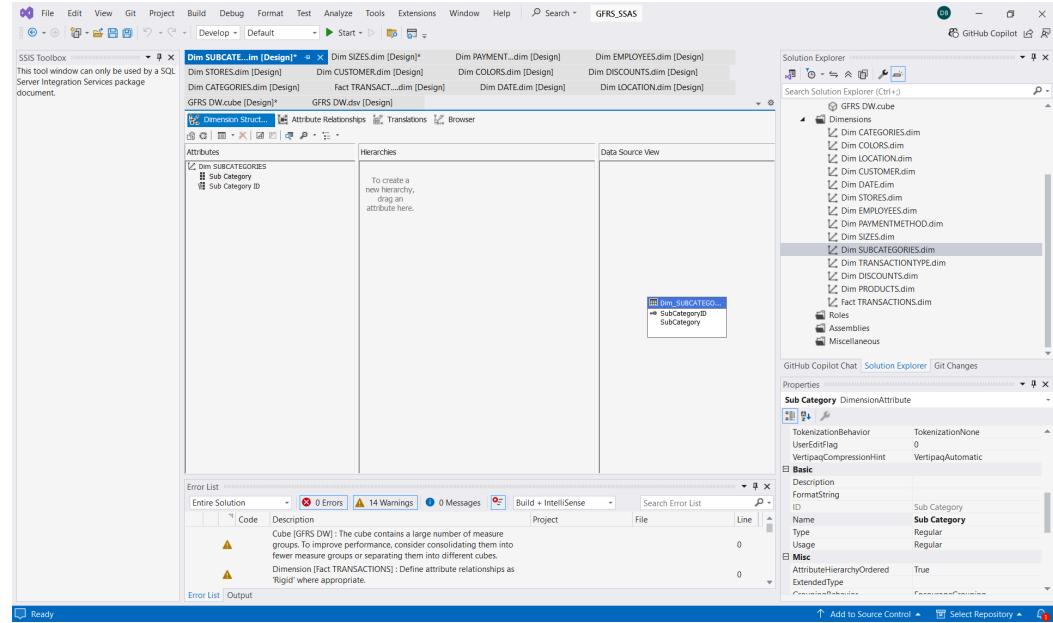
Hình 429. Thêm thuộc tính SizeID vào Key Columns



Hình 430. Chọn tên cho Name Column cho thuộc tính Size

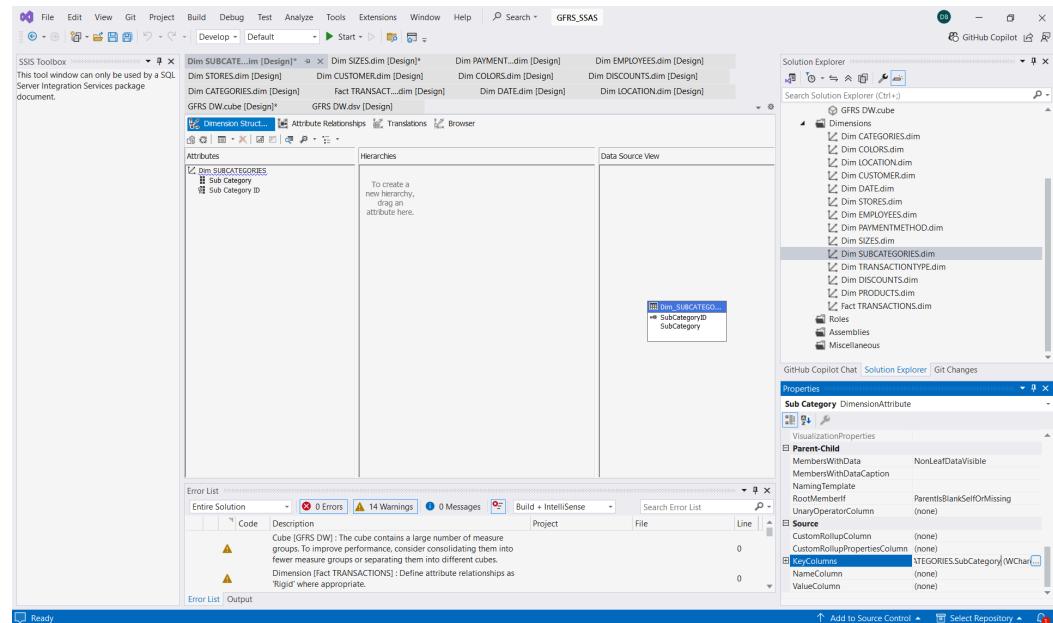
#### III.2.1.4.10. Cài đặt Dim\_SUBCATEGORIES

- **Bước 1:** Kéo thuộc tính **SubCategory** từ khung **Data Source View** sang khung **Attributes**.

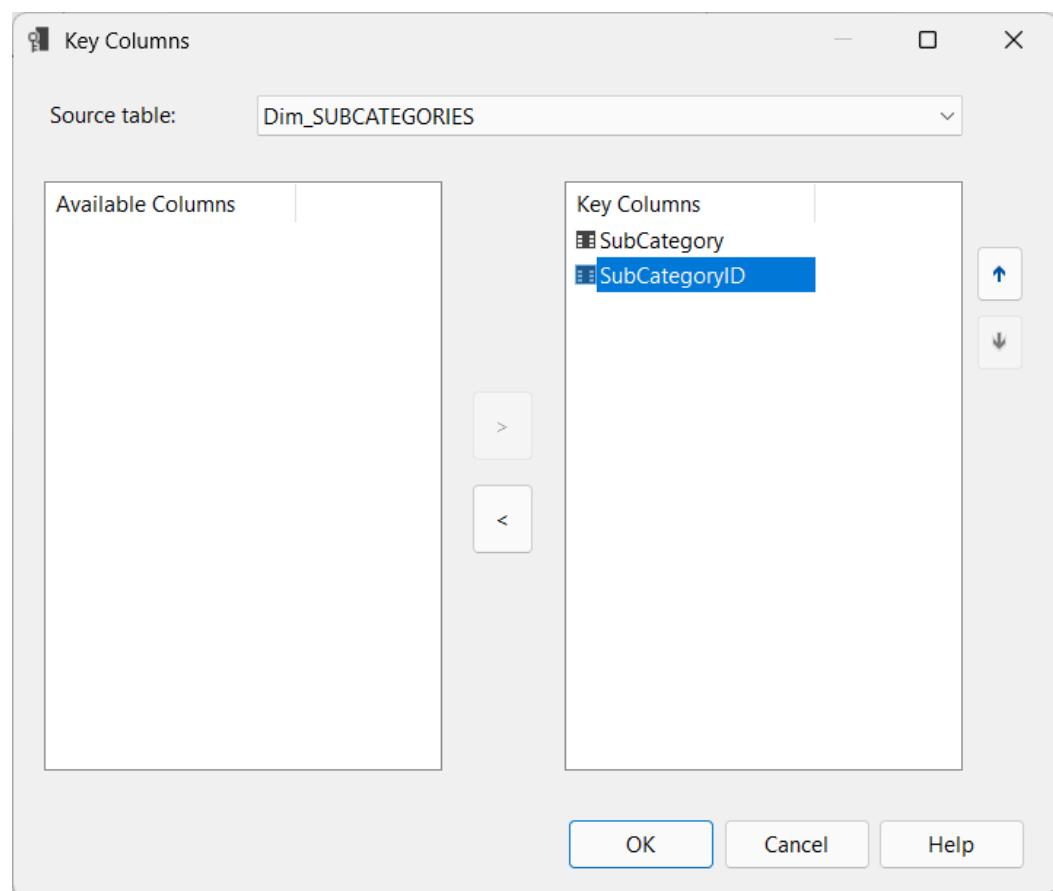


Hình 431. Kéo thuộc tính SubCategory sang khung Attributes

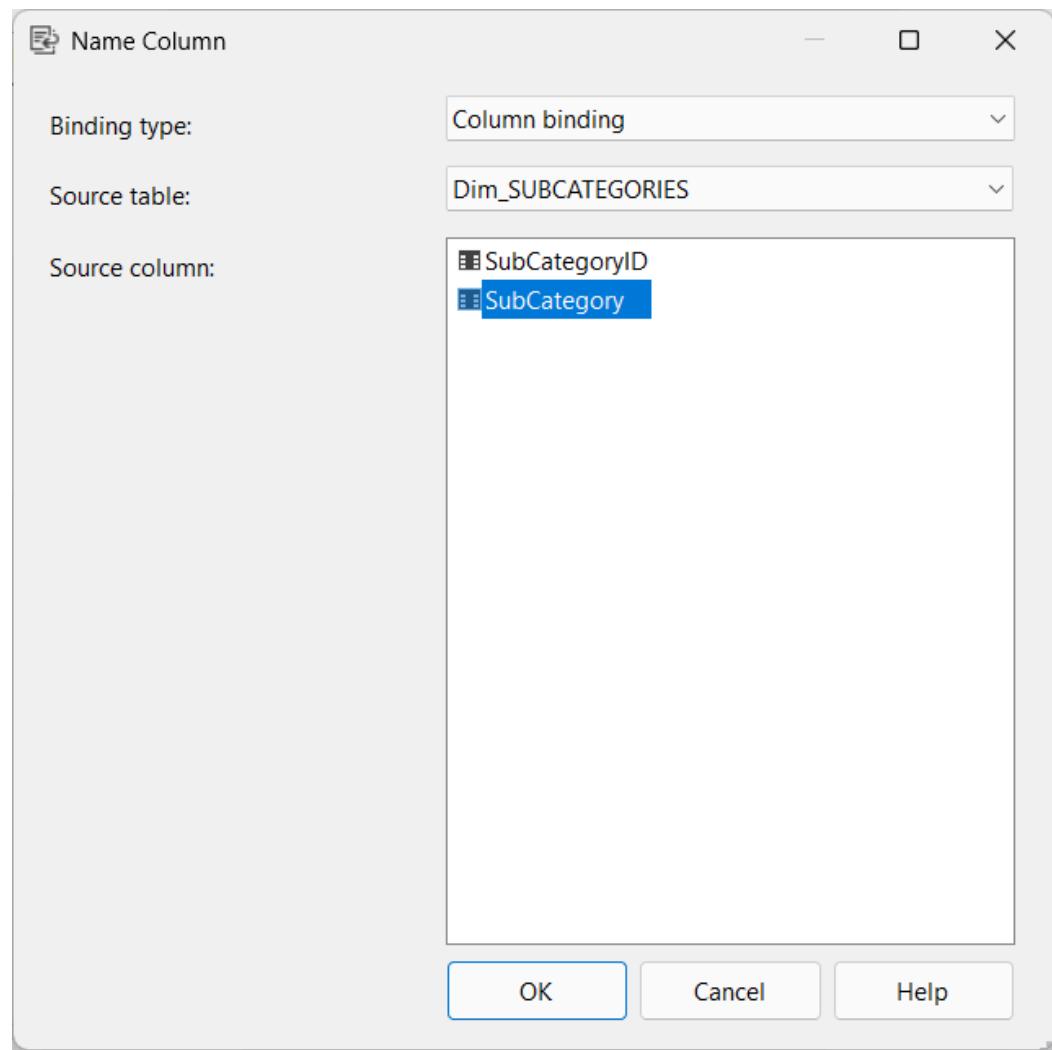
- **Bước 2:** Nhấn chuột vào **SubCategory** trên khung **Attributes**. Ở phần **Properties** ta kéo xuống phần **KeyColumns**, nhấn chuột vào (...) bên phải. Cửa sổ **Key Columns** hiện lên, ta chọn thuộc tính **SubCategoryID** từ khung **Available Columns**, sau đó nhấn > để đẩy sang khung **Key Columns**, nhấn **OK** để hoàn tất. Tiếp đó, ta nhấn chuột vào (...) ở **NameColumn**, cửa sổ **Name Column** hiện lên, ta chọn **SubCategory** sau đó nhấn **OK**.



Hình 432. Thêm thuộc tính vào KeyColumns cho SubCategory



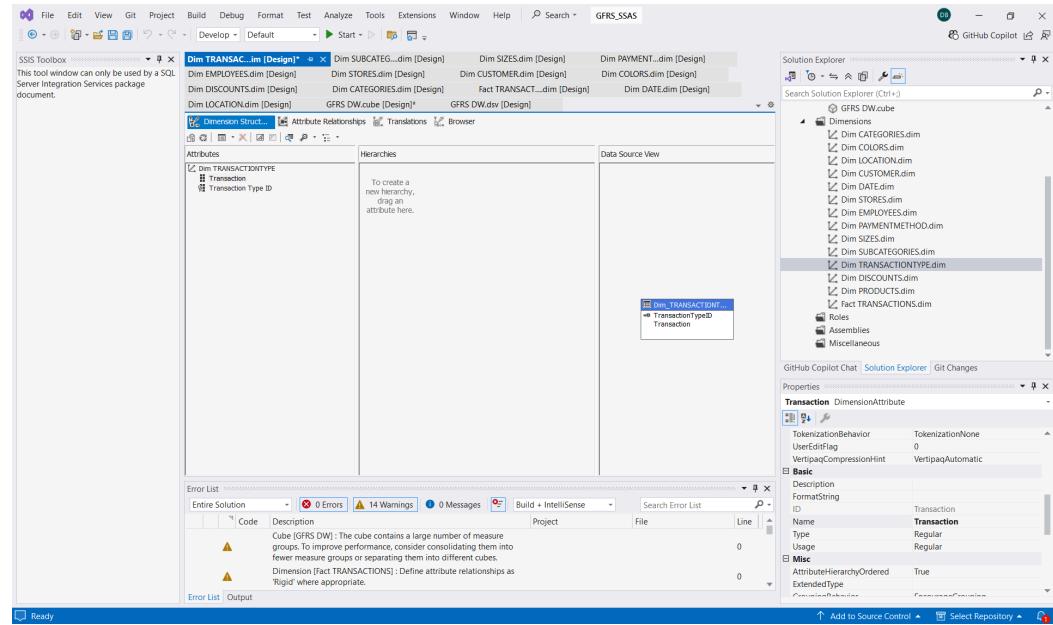
Hình 433. Thêm thuộc tính SubCategoryID vào SubCategory



Hình 434. Chọn tên cho Name Column cho thuộc tính SubCategory

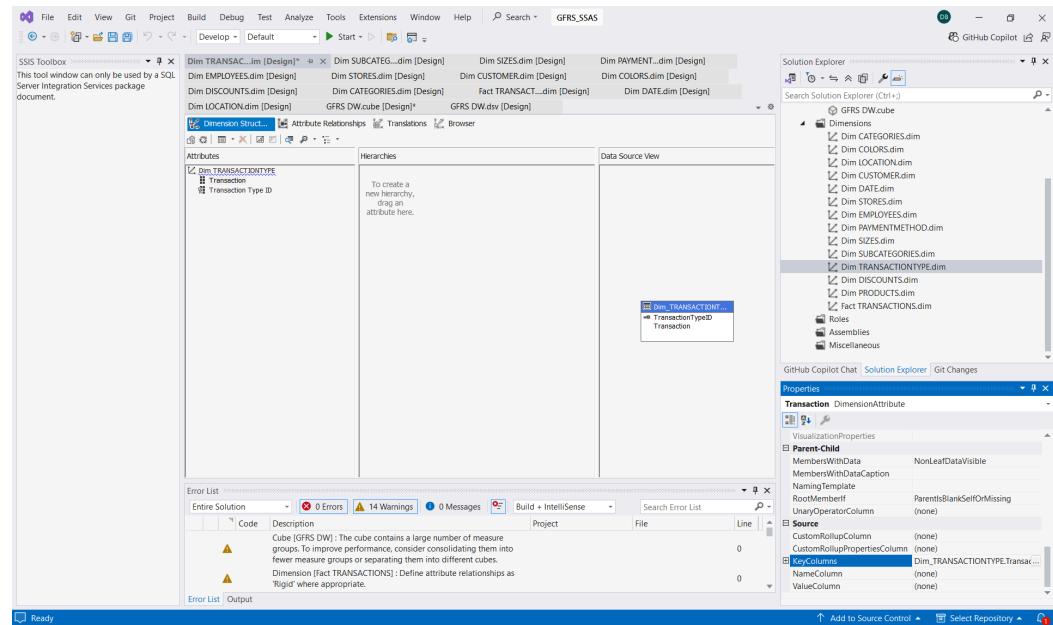
#### ***III.2.1.4.11. Cài đặt Dim\_TRANSACTIONTYPE***

- **Bước 1:** Kéo thuộc tính **Transaction** từ khung **Data Source View** sang khung **Attributes**.

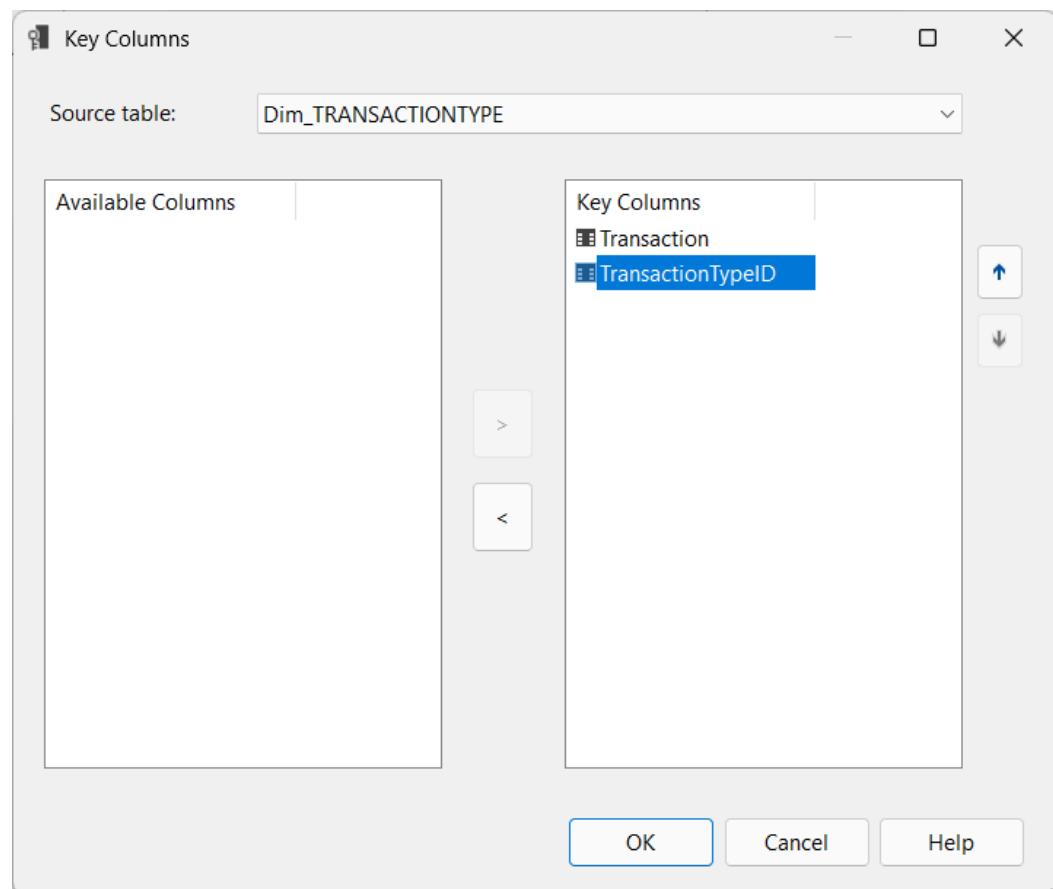


Hình 435. Kéo thuộc tính Transaction sang khung Attributes

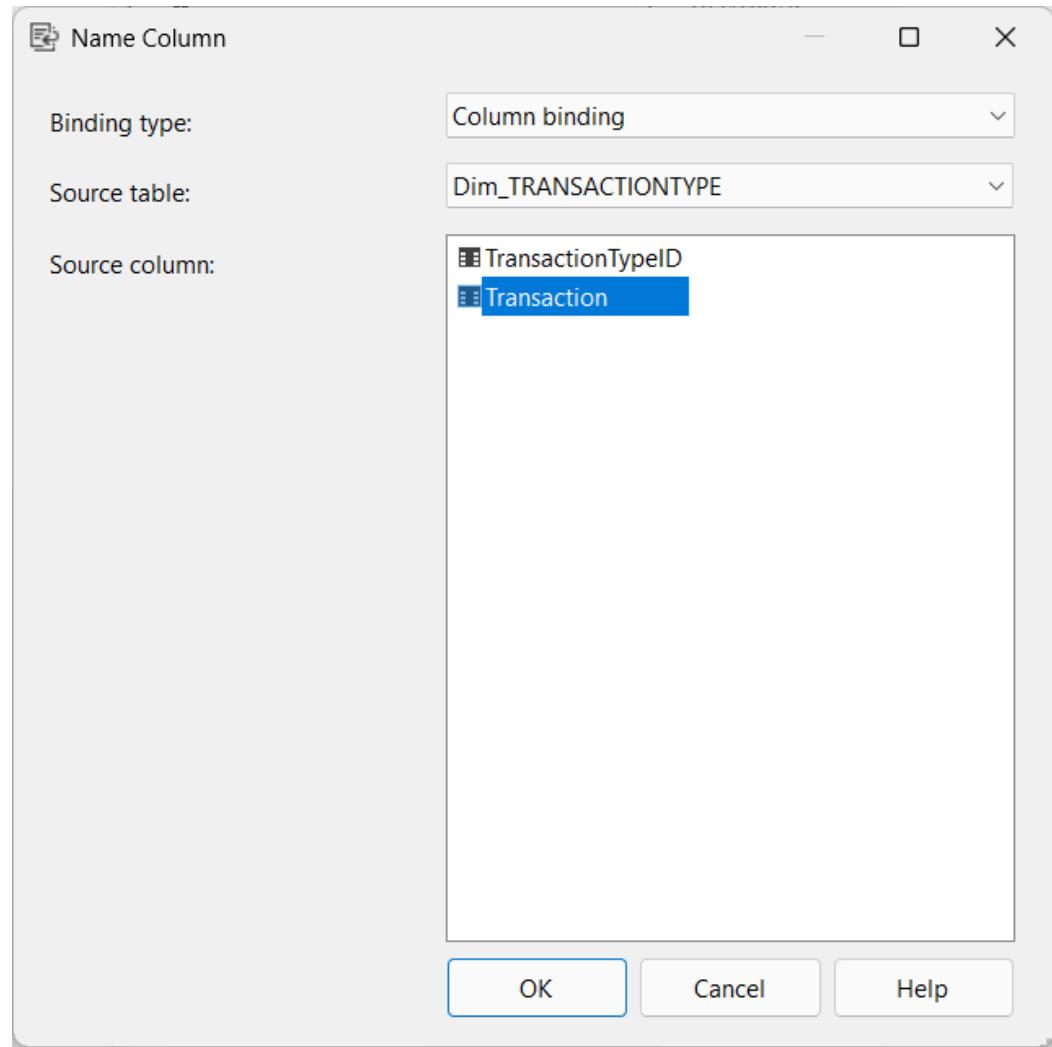
- **Bước 2:** Nhấn chuột vào **Transaction** trên khung **Attributes**. Ở phần **Properties** ta kéo xuống phần **KeyColumns**, nhấn chuột vào (...) bên phải. Cửa sổ **Key Columns** hiện lên, ta chọn thuộc tính **TransactionTypeID** từ khung **Available Columns**, sau đó nhấn > đè đầy sang khung **Key Columns**, nhấn **OK** để hoàn tất. Tiếp đó, ta nhấn chuột vào (...) ở **NameColumn**, cửa sổ **Name Column** hiện lên, ta chọn **Transaction** sau đó nhấn **OK**.



Hình 436. Thêm thuộc tính vào TransactionTypeID cho Transaction



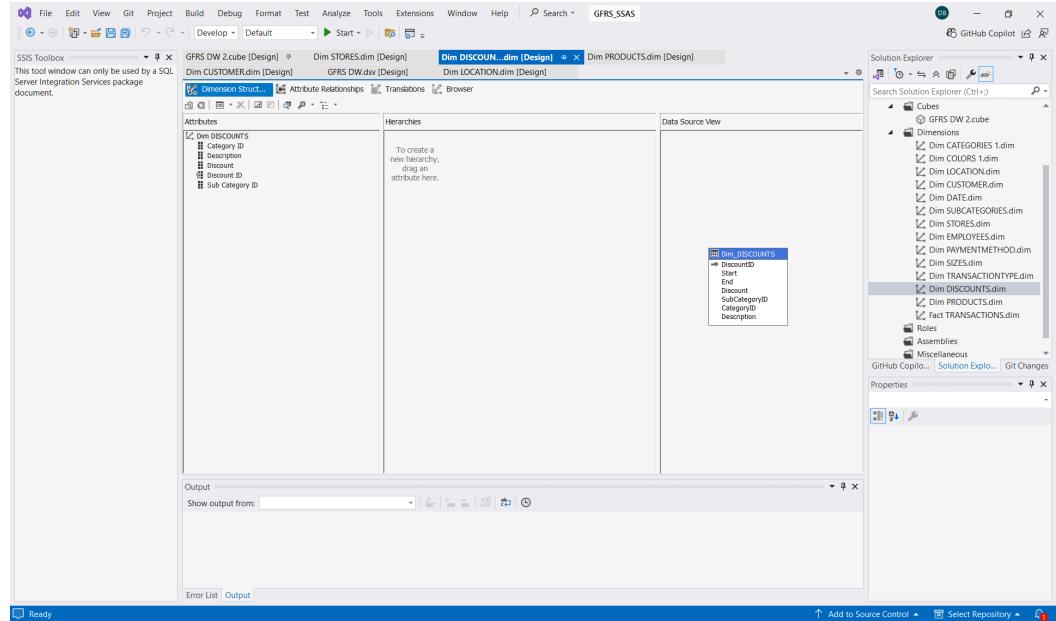
Hình 437. Thêm thuộc tính TransactionTypeID vào Key Columns



Hình 438. Chọn tên cho Name Column cho thuộc tính TransactionType

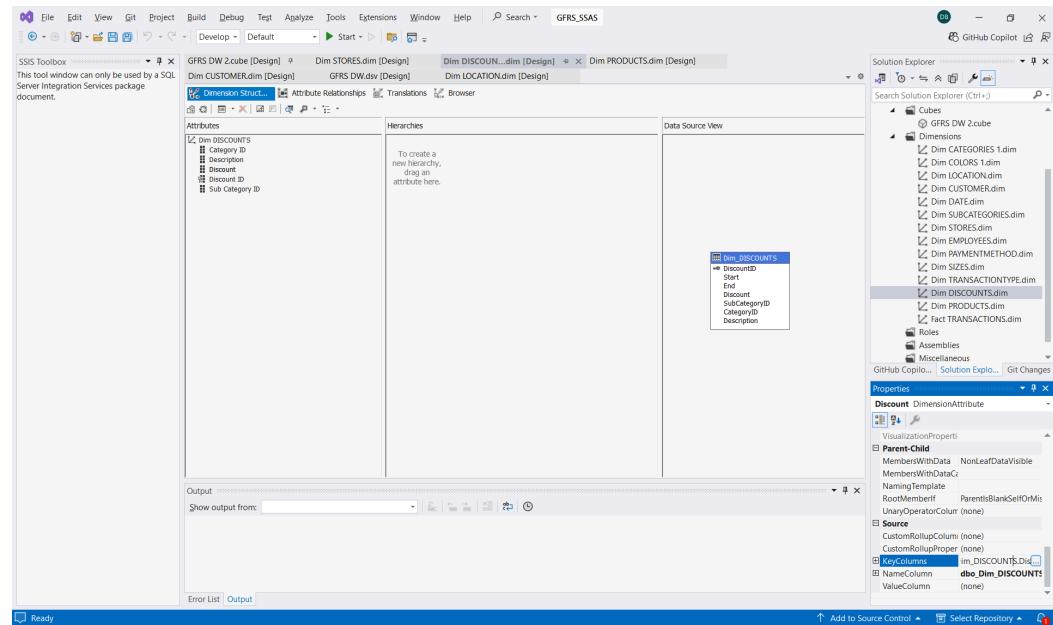
#### **III.2.1.4.12. Cài đặt Dim\_DISCOUNTS**

- **Bước 1:** Kéo thuộc tính **Discount**, **Description** từ khung **Data Source View** sang khung **Attributes**.

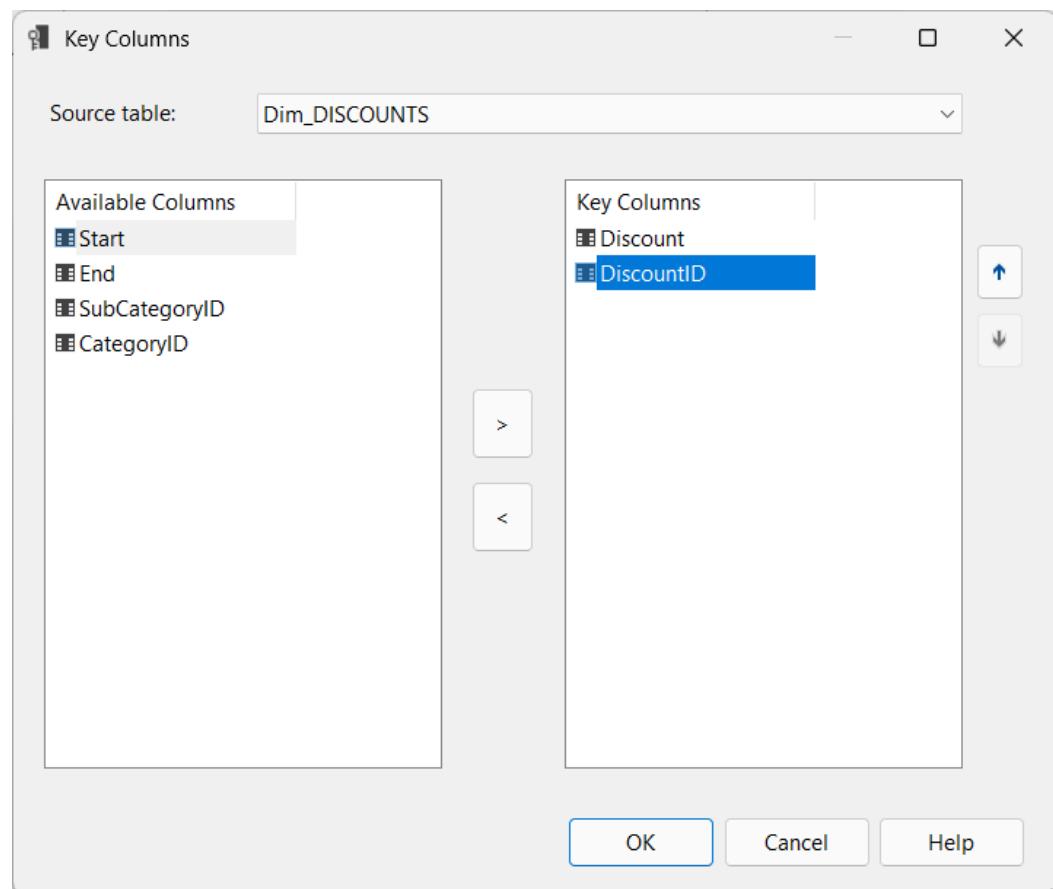


Hình 439. Kéo thuộc tính **Discount, Description** sang khung **Attributes**

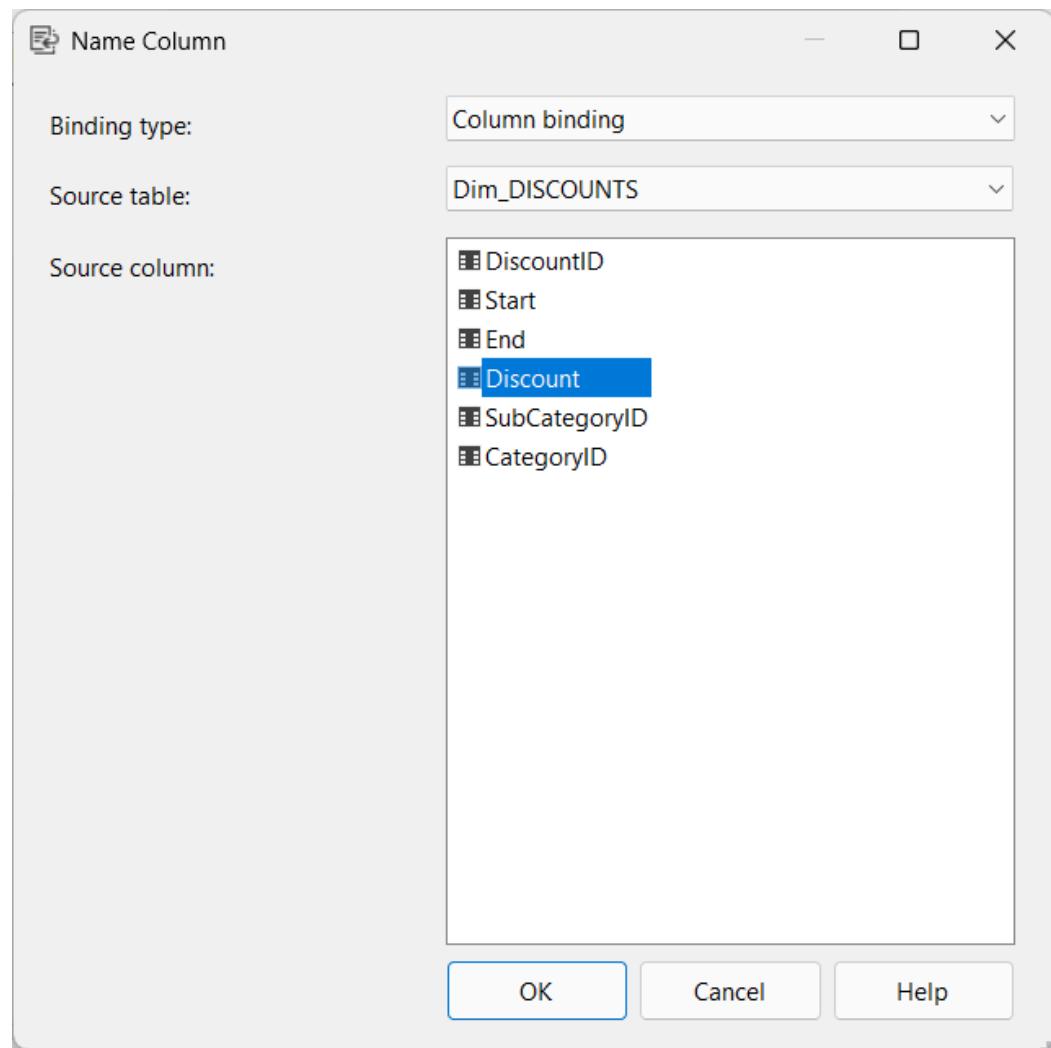
- **Bước 2:** Nhấn chuột vào **Discount** trên khung **Attributes**. Ở phần **Properties** ta kéo xuống phần **KeyColumns**, nhấn chuột vào (...) bên phải. Cửa sổ **Key Columns** hiện lên, ta chọn thuộc tính **DiscountID** từ khung **Available Columns**, sau đó nhấn > để đẩy sang khung **Key Columns**, nhấn **OK** để hoàn tất. Tiếp đó, ta nhấn chuột vào (...) ở **NameColumn**, cửa sổ **Name Column** hiện lên, ta chọn **Discount** sau đó nhấn **OK**. Làm tương tự với **Description** với **Key Columns** không đổi, **NameColumn** tương ứng.



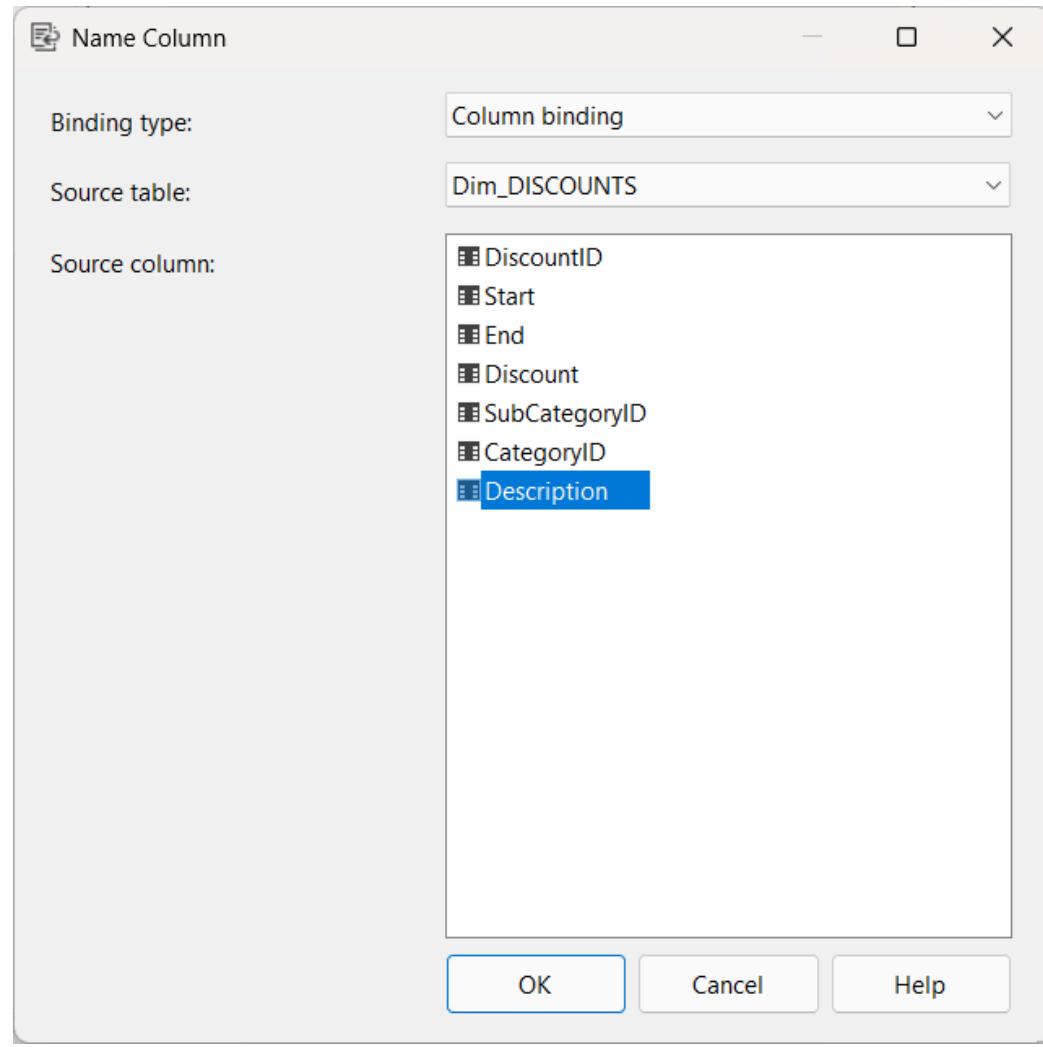
Hình 440. Thêm thuộc tính vào *DiscountID* cho *Discount*



Hình 441. Thêm thuộc tính *DiscountID* vào *Key Columns*



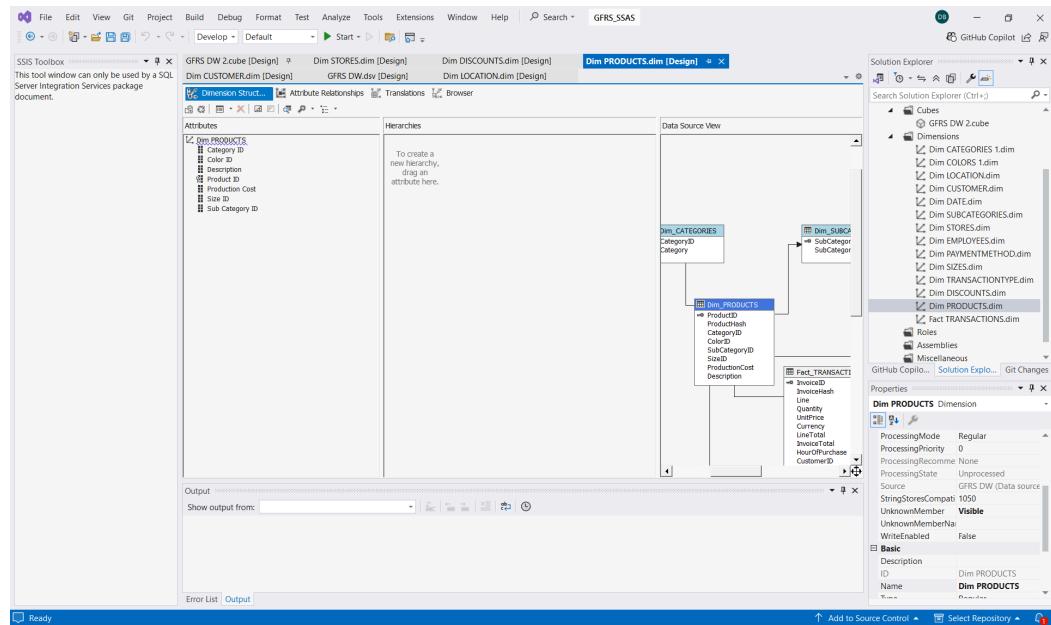
Hình 442. Chọn tên cho Name Column cho thuộc tính Discount



Hình 443. Chọn tên cho Name Column cho thuộc tính Description

#### **III.2.1.4.13. Cài đặt Dim\_PRODUCTS**

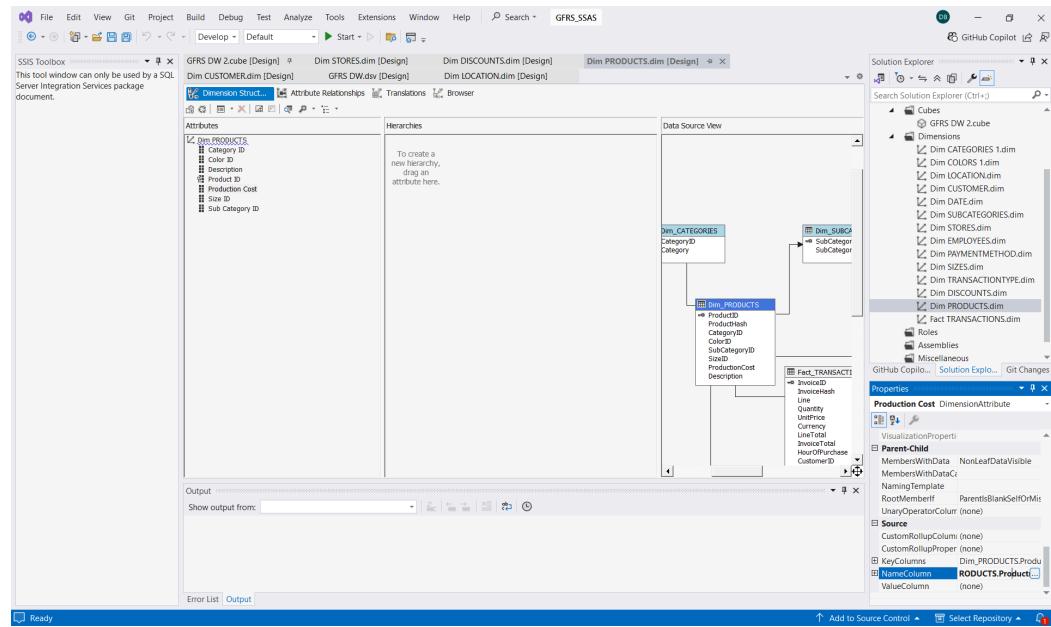
- **Bước 1:** Kéo thuộc tính **Production Cost**, **Description** từ khung **Data Source View** sang khung **Attributes**.



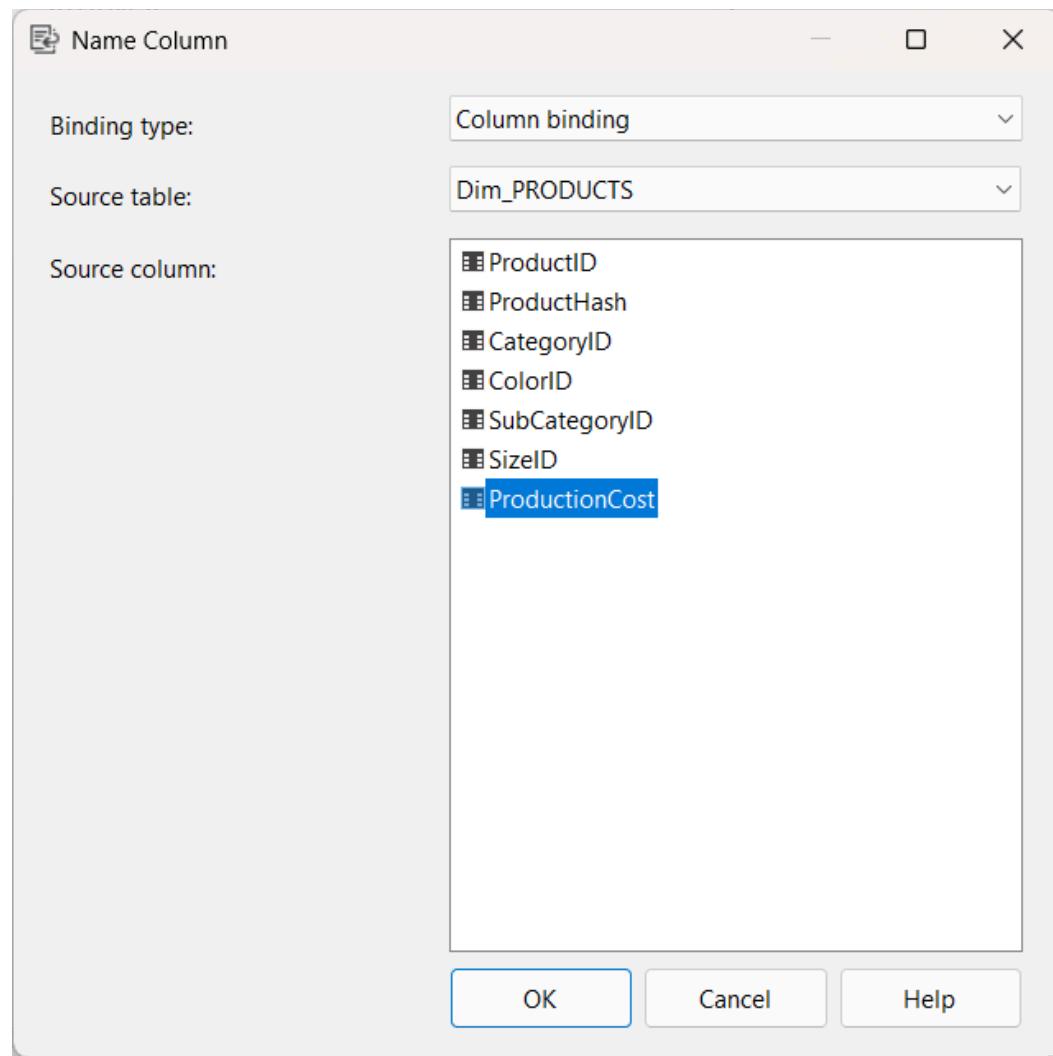
Hình 444. Kéo thuộc tính Production Cost, Description sang khung Attributes

#### Attributes

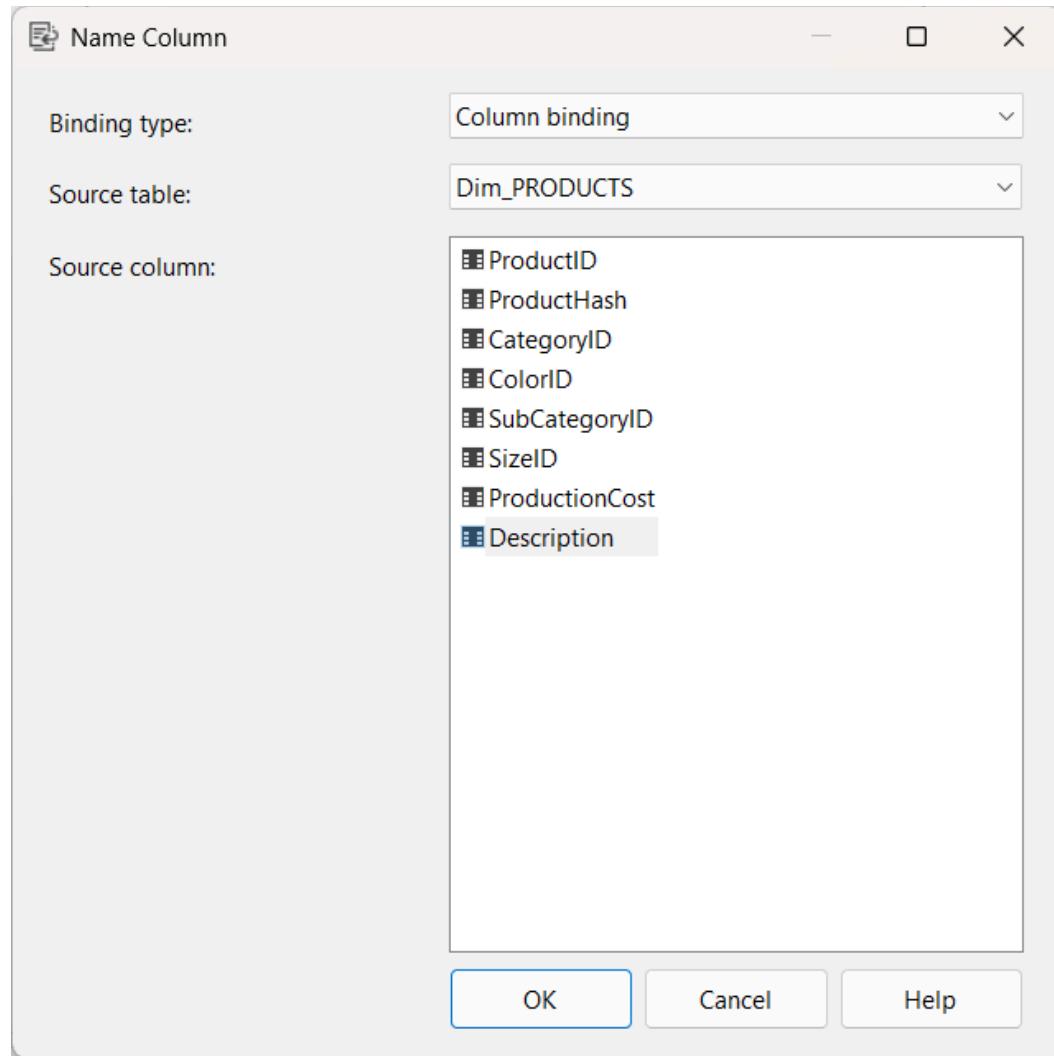
- **Bước 2:** Nhấn chuột vào **Production Cost** trên khung **Attributes**. Ở phần **Properties** ta kéo xuống phần **NameColumn** nhấn chuột vào (...), cửa sổ **Name Column** hiện lên, ta chọn **Production Cost** sau đó nhấn **OK**. Làm tương tự với **Description** với **KeyColumns** không đổi, **NameColumn** tương ứng.



Hình 445. Thêm thuộc tính vào NameColumn cho ProductionCost



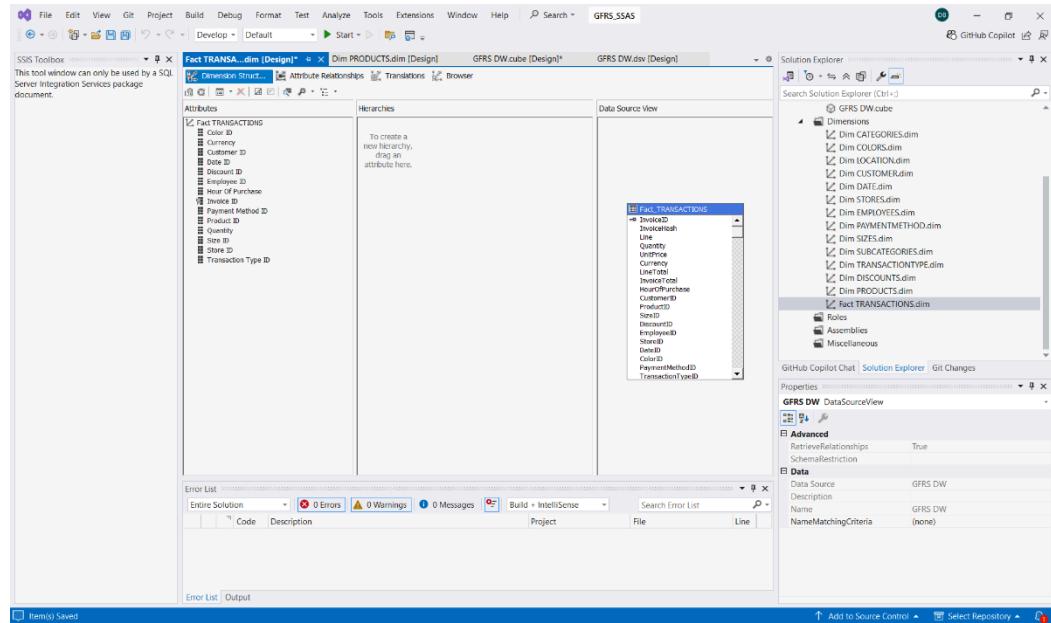
Hình 446. Chọn tên cho Name Column cho thuộc tính ProductionCost



Hình 447. Chọn tên cho Name Column cho thuộc tính Description

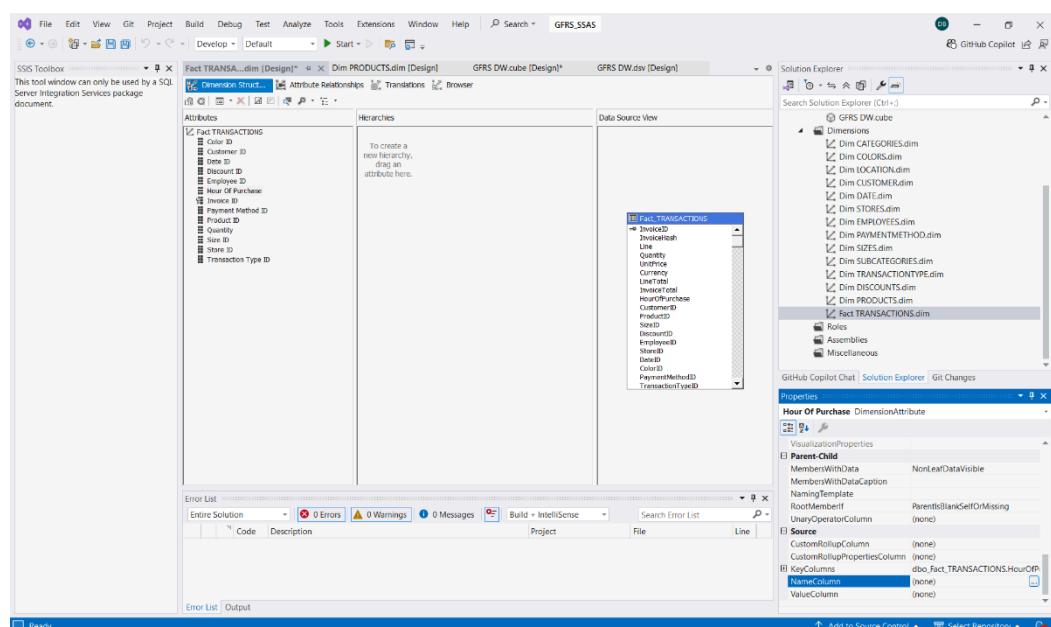
#### ***III.2.1.4.14. Cài đặt Fact\_TRANSACTION***

- **Bước 1:** Kéo thuộc tính **HourOfPurchase**, **Quantity** từ khung **Data Source View** sang khung **Attributes**.

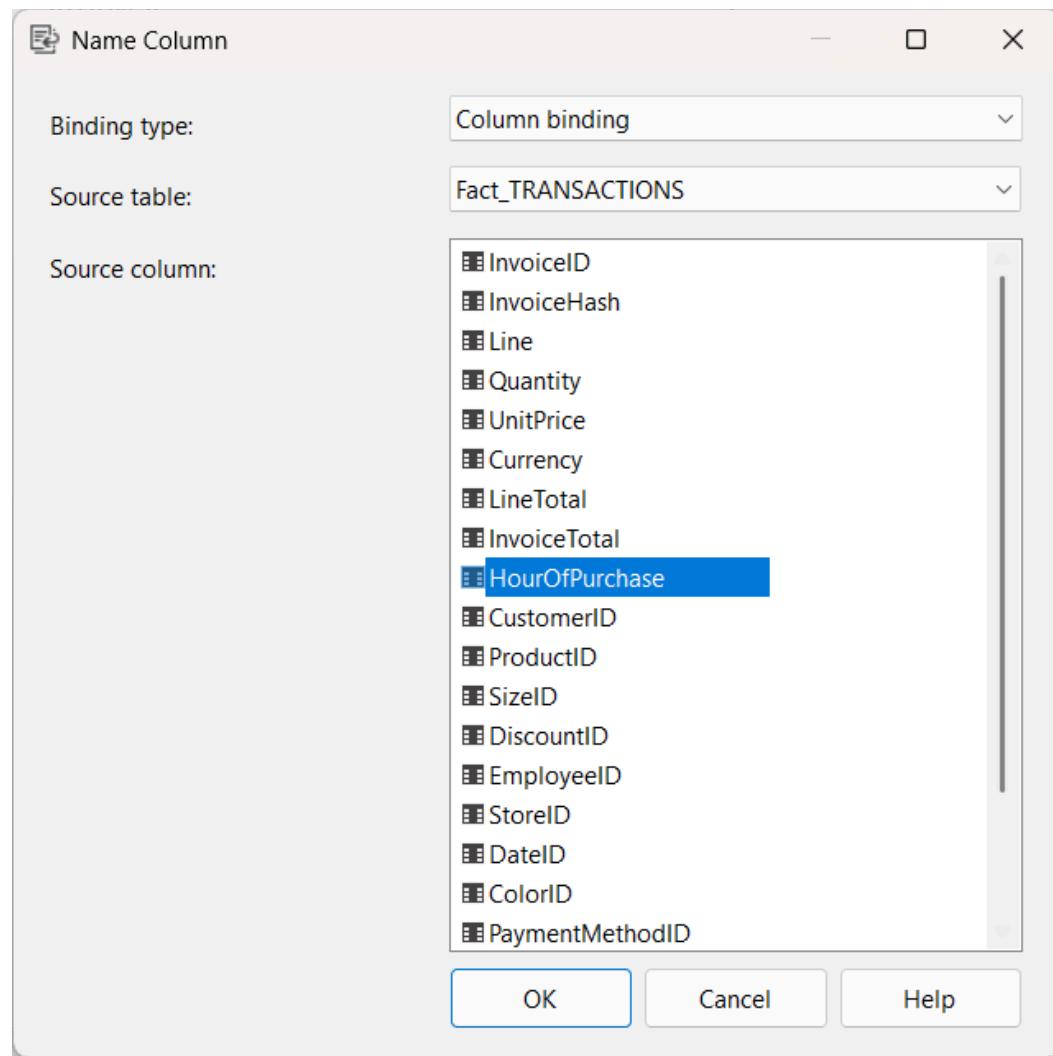


Hình 448. Kéo thuộc tính HourOfPurchase, Quantity sang khung Attributes

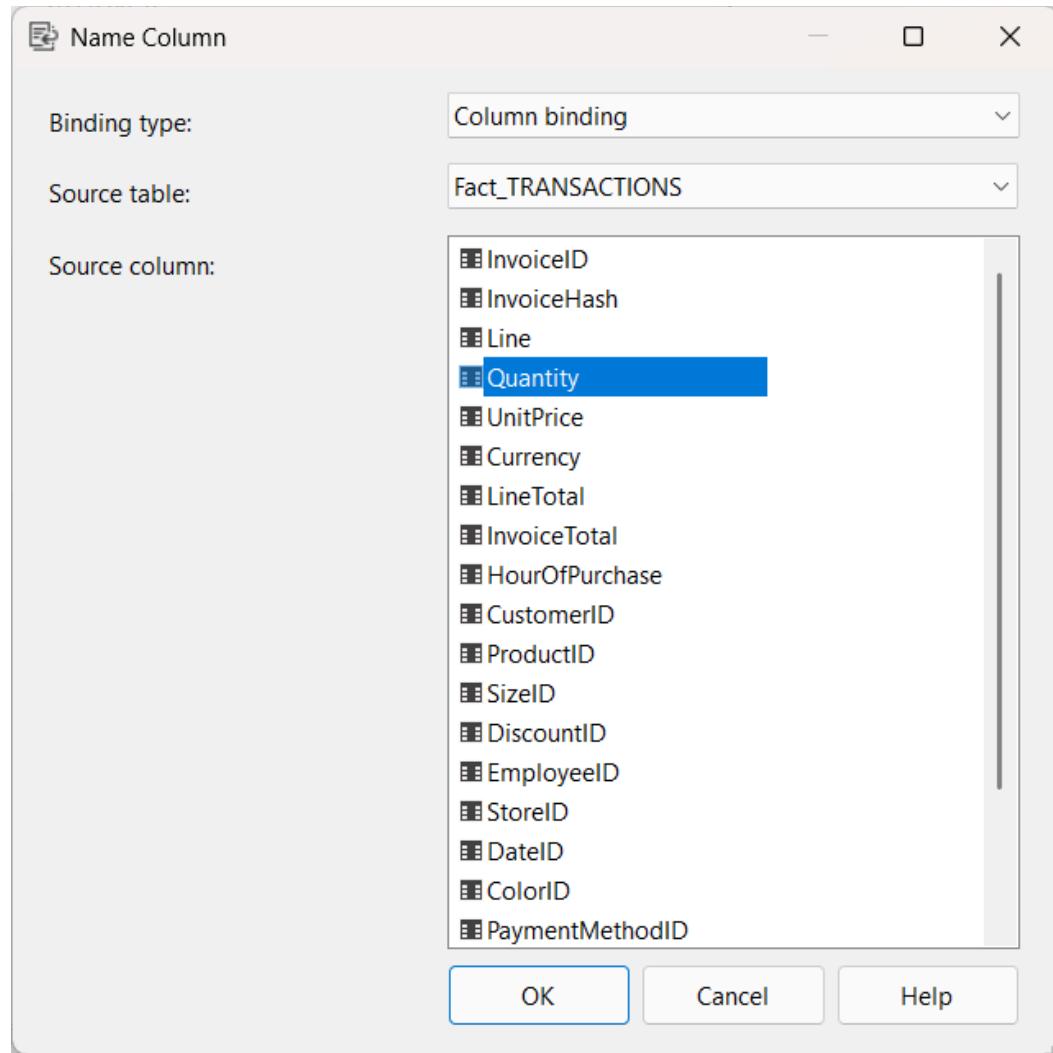
- **Bước 2:** Nhấn chuột vào **HourOfPurchase** trên khung **Attributes**. Ở phần **Properties** ta kéo xuống phần **NameColumn** nhấn chuột vào (...), cửa sổ **Name Column** hiện lên, ta chọn **Production Cost** sau đó nhấn **OK**. Làm tương tự với **Quantity** với **KeyColumns** không đổi, **NameColumn** tương ứng.



Hình 449. Thêm thuộc tính vào NameColumn cho HourOfPurchase



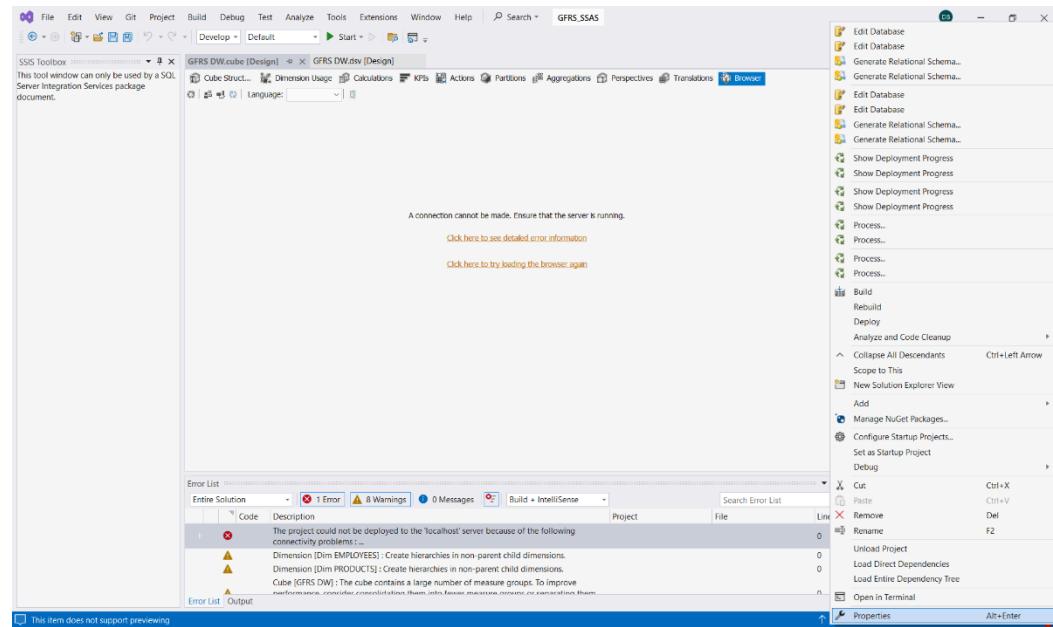
Hình 450. Chọn tên cho Name Column cho thuộc tính HourOfPurchase



Hình 451. Chọn tên cho Name Column cho thuộc tính Quantity

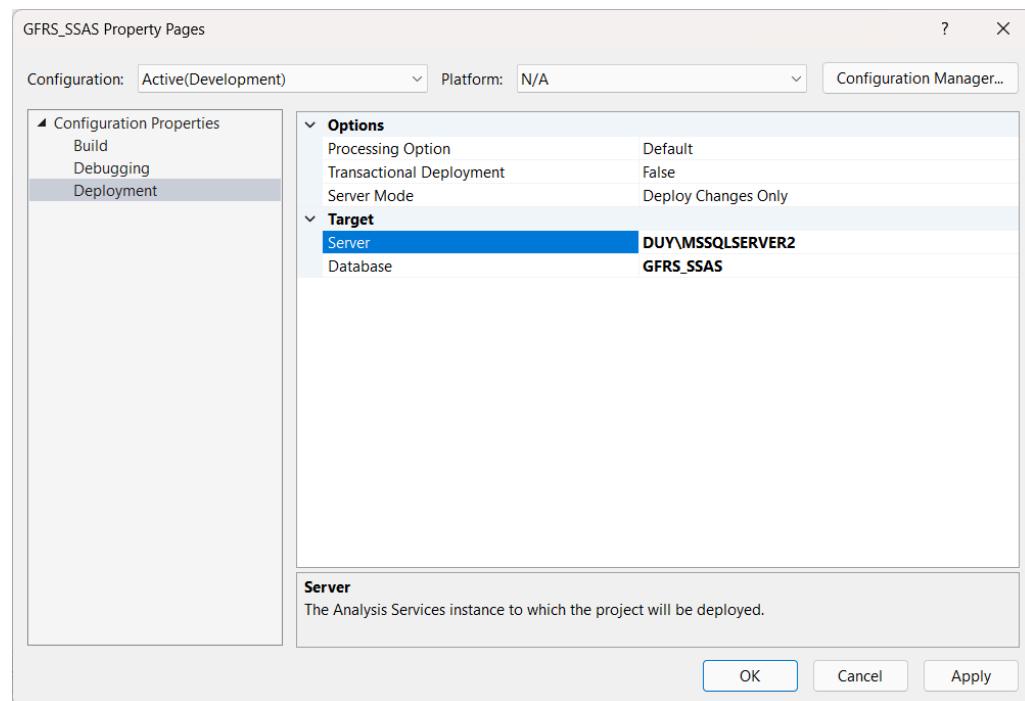
### III.2.1.5. Tiến hành Process Cube

- **Bước 1:** Tiến hành chạy Cube. Ta nhấp chuột phải vào **GFRS\_SSAS** vừa tạo, sau đó chọn **Properties**.



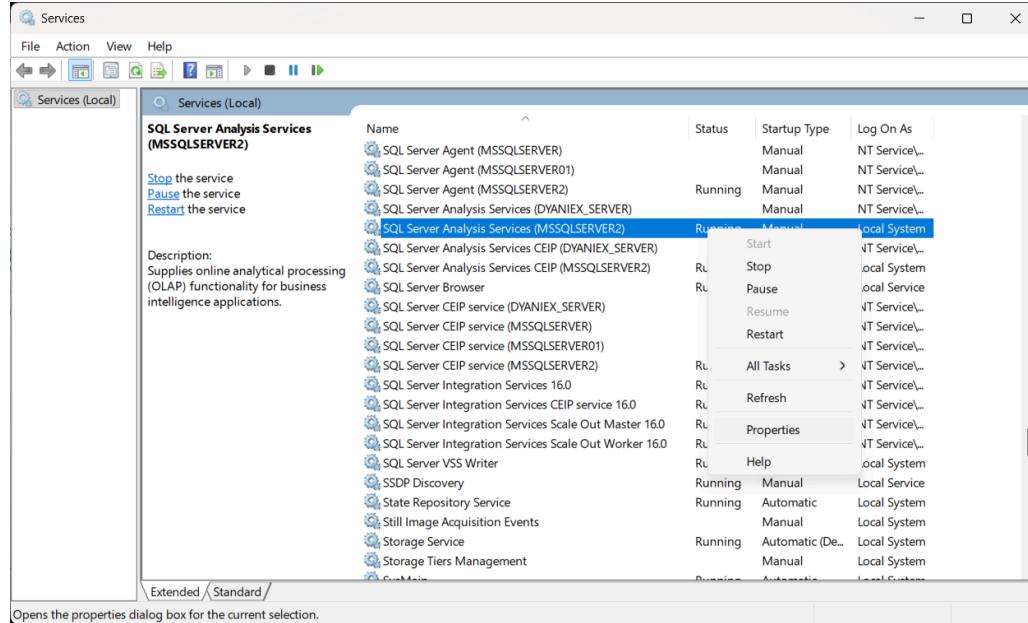
Hình 452. Cài đặt trước khi chạy Cube

- **Bước 2:** Cửa sổ Property Page sẽ hiện lên, chuyển đến mục Deployment, ta đổi tên Server (**DUY\MSSQLSERVER2**) và Database (**GFRS\_SSAS**) đúng với tên đã tạo trước đó. Sau đó chọn **OK**.



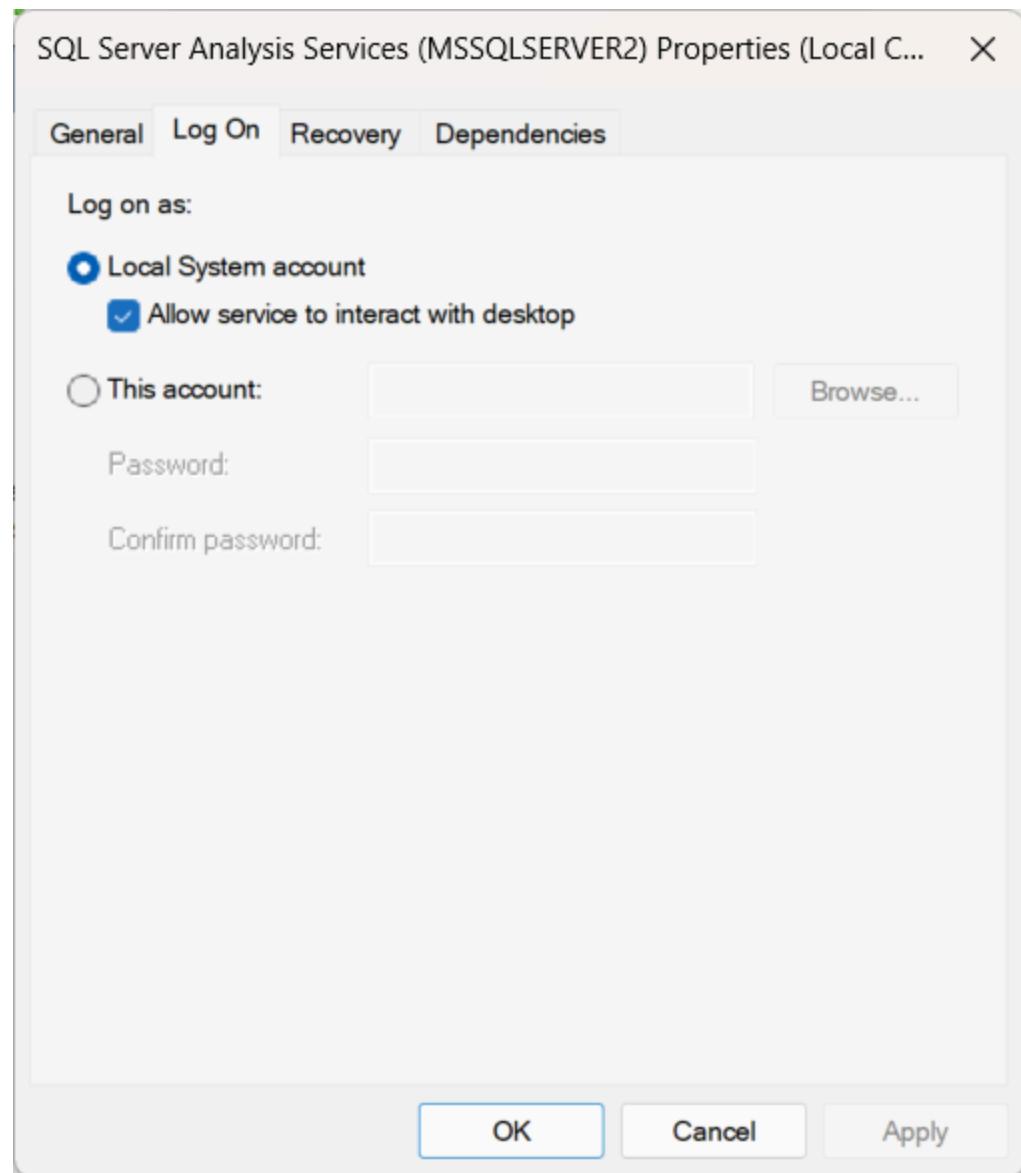
Hình 453. Cài đặt tên server và database

- **Bước 3:** Mở phần Services của thiết bị. Lần lượt tìm đến **SQL Server Analysis Services (MSSQLSERVER2)** và **SQL Server Analysis Services CEIP (MSSQLSERVER2)**, nhấp chuột phải và chọn Properties.



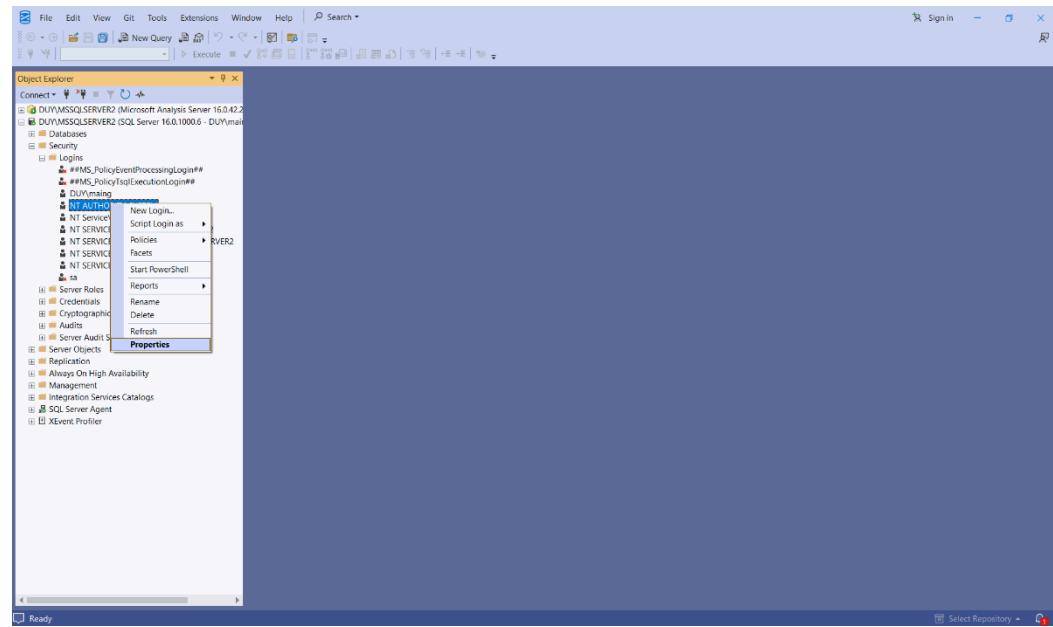
Hình 454. Cài đặt SQL Server Analysis Services

- **Bước 4:** Cửa sổ Properties hiện lên, chuyển qua tab Log On chọn Local System account, sau đó tick vào ô Allow service to interact with desktop. Sau đó chọn Ok để kết thúc. Làm tương tự với **SQL Server Analysis Services CEIP (MSSQLSERVER2)**. Sau đó Restart lại 2 services này



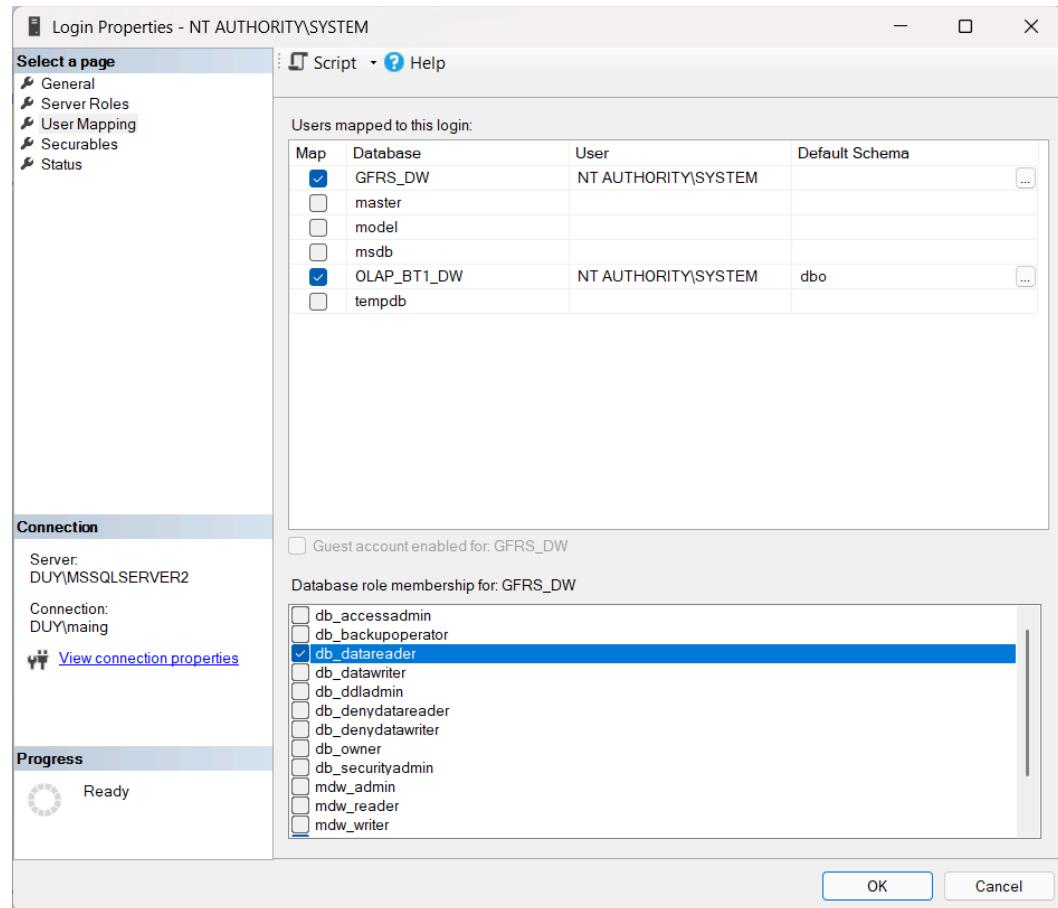
Hình 455. Cài đặt Properties ở phần Log On

- **Bước 5:** Mở **SQL Server Management Studio 21**, đăng nhập vào **GFRS\_DW**. Ở mục **Security**, chọn mục **Logins**, sau đó nhấn chuột phải vào **NT AUTHORITY\SYSTEM**, chọn **Properties**.



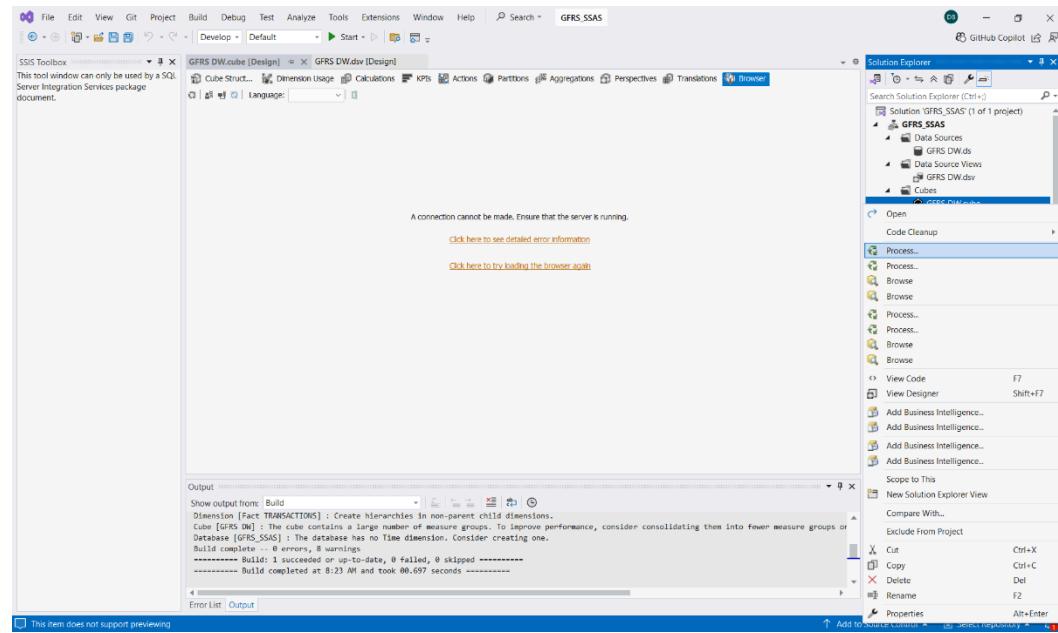
Hình 456. Mở Properties của NT AUTHORITY\SYSTEM

- **Bước 6:** Chuyển sang tab **User Mapping**, chọn Database **GFRS\_DW**, đánh dấu vào ô **db\_datareader** ở khung **Database role membership for:**, sau đó chọn **OK** để kết thúc

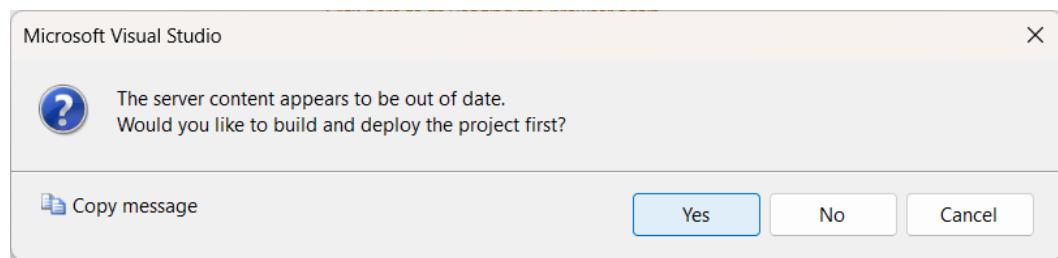


Hình 457. Cài đặt role membership

- **Bước 7:** Trở lại với project SSAS trong **Visual Studio 2022**. Nhấp phải vào **Cube** vừa tạo, chọn **Process**. Một cửa sổ thông báo có thể sẽ hiện lên, ta chọn **Yes** để tiếp tục.

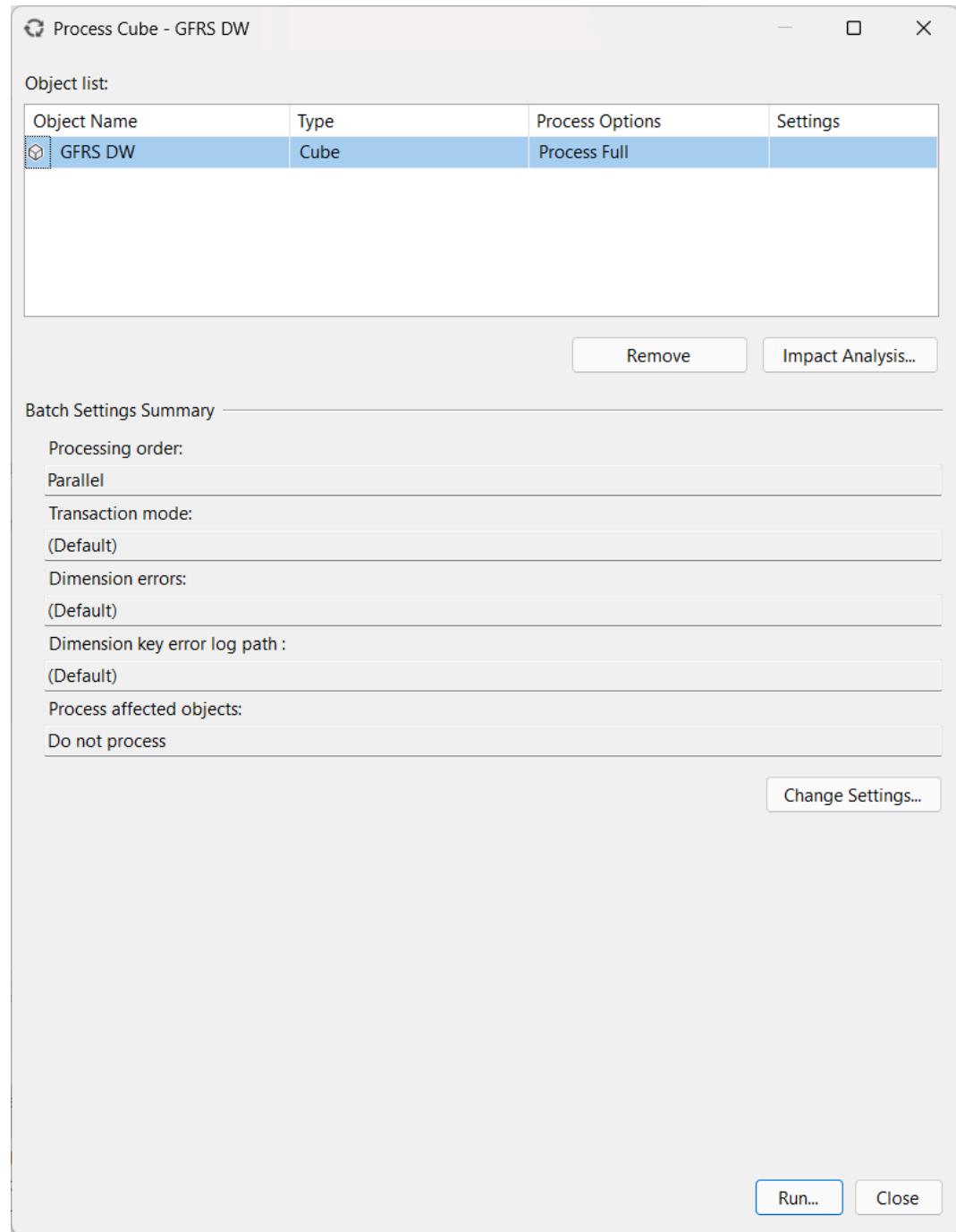


Hình 458. Process Cube vừa mới tạo

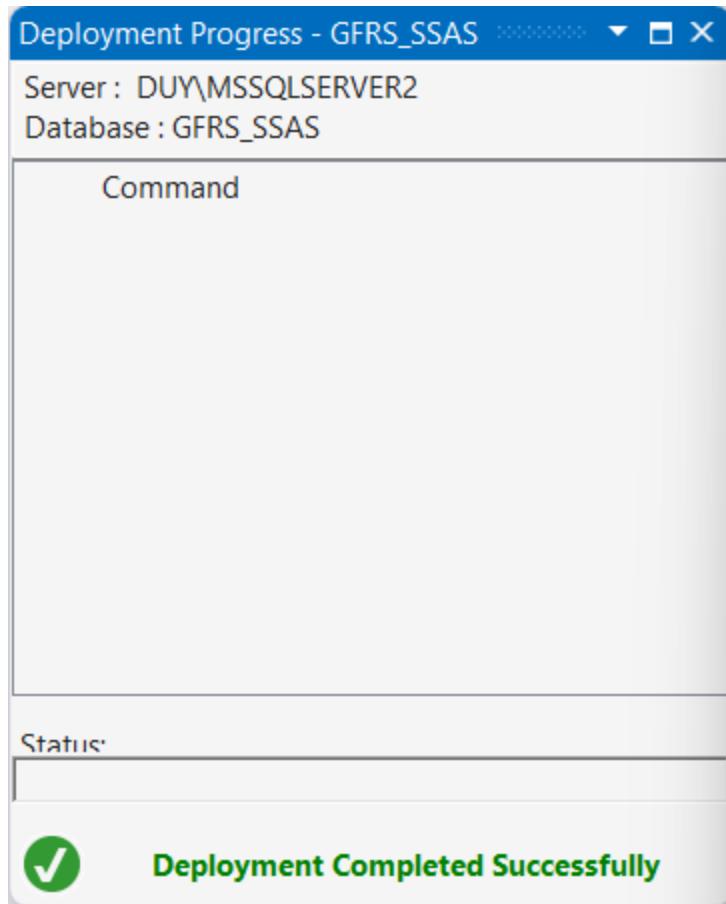


Hình 459. Chọn Yes để tiếp tục quá trình

- **Bước 8:** Cửa sổ **Process Cube** sẽ hiện lên, chọn **Run..** để chạy.

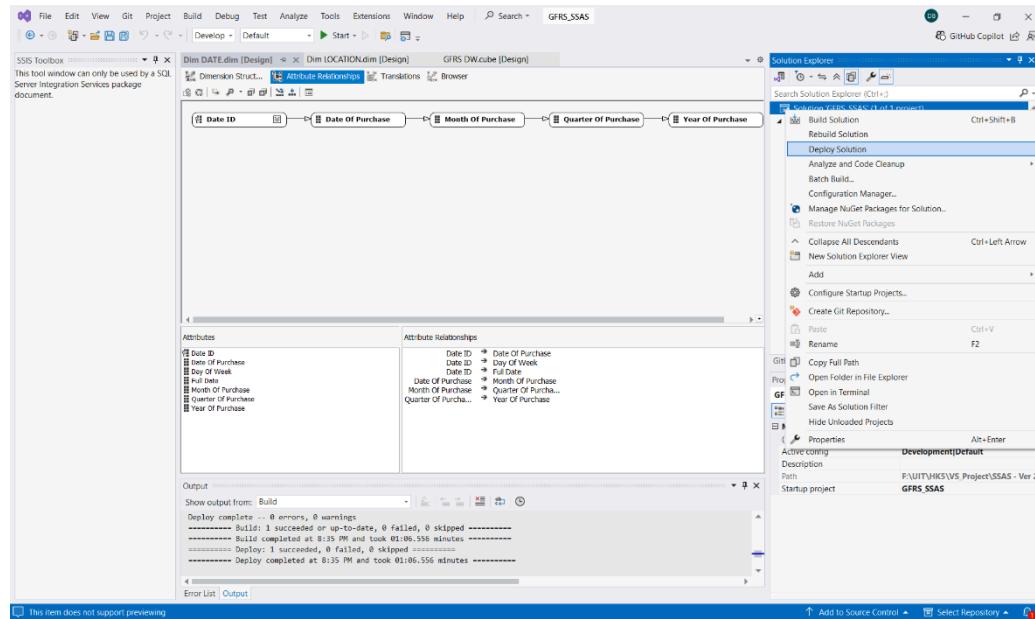


Hình 460. Cửa sổ Process Cube

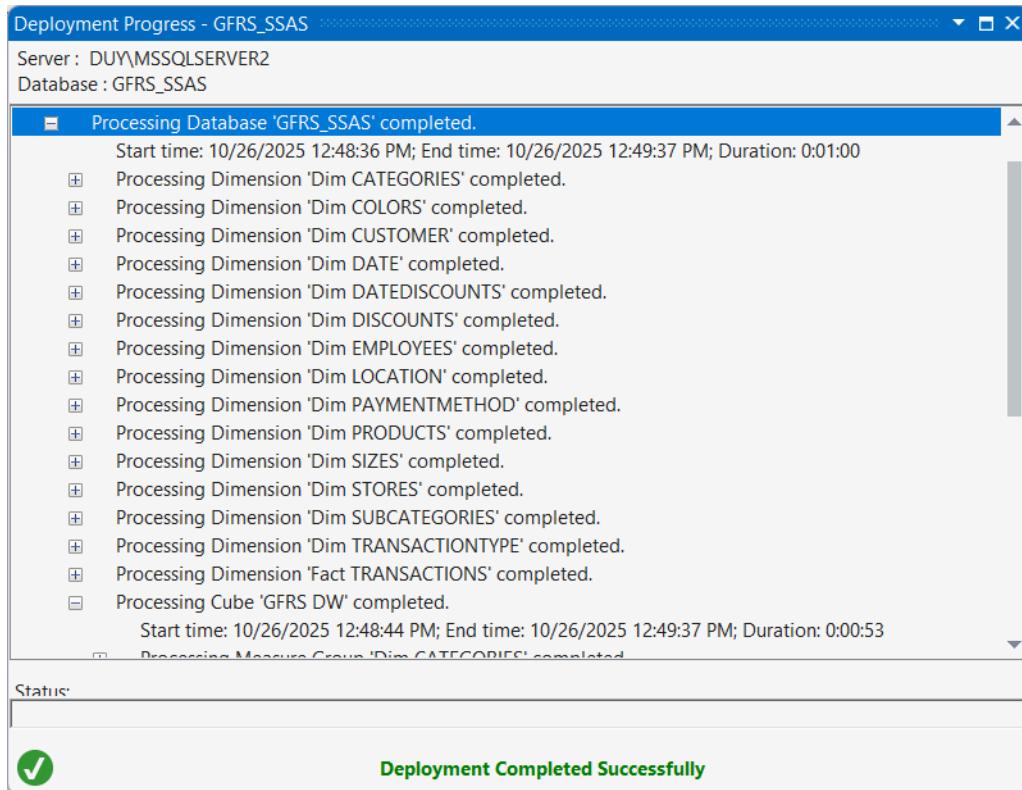


Hình 461. Cửa sổ thông báo Process Cube thành công

### III.2.1.6. Deploy Solution: Nhấp chuột phải vào Solution chọn Deploy Solution



Hình 462. Chọn Deploy Solution để chạy Solution

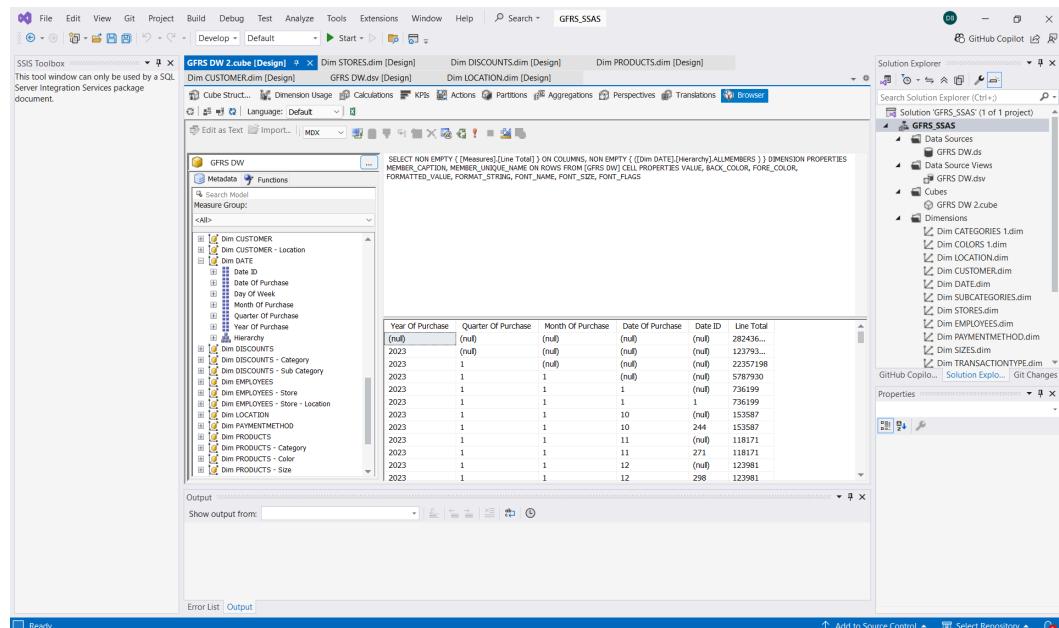


Hình 463. Solution được Deploy thành công

### III.2.2. Thực hiện câu truy vấn

#### III.2.2.1. Doanh số roll up theo năm

- **Mục tiêu:** hiển thị doanh số được roll up theo năm
- **Ý nghĩa:** **tổng hợp và cộng gộp** toàn bộ dữ liệu doanh thu chi tiết (từ cấp độ thấp như ngày, tháng) lên cấp độ cao hơn là **Năm**. Thao tác này giúp chuyển đổi dữ liệu thô thành các con số tổng kết để quan sát được bức tranh toàn cảnh và xu hướng dài hạn của dữ liệu.
- **Công cụ thực hiện:** SSAS trong Visual Studio 2022
- **Thực hiện:** Kéo thuộc tính **Hierarchy** từ **Dim DATE** vào khung truy vấn. Tiếp đó kéo **Measure Line Total** ở mục **Fact TRANSACTIONS** vào khung. Sau đó, trên thành công cụ ta tắt chế độ **Design Mode**, khi này câu lệnh MDX kéo thả của ta vừa thực sẽ xuất hiện, ở đây phần (**[Dim DATE].[Hierarchy].[Date ID].ALLMEMBERS**) ta sẽ xóa **[Date ID]** để thực hiện câu truy vấn Roll UP. Cuối cùng chọn **Click to execute the query** để thực hiện câu truy vấn.

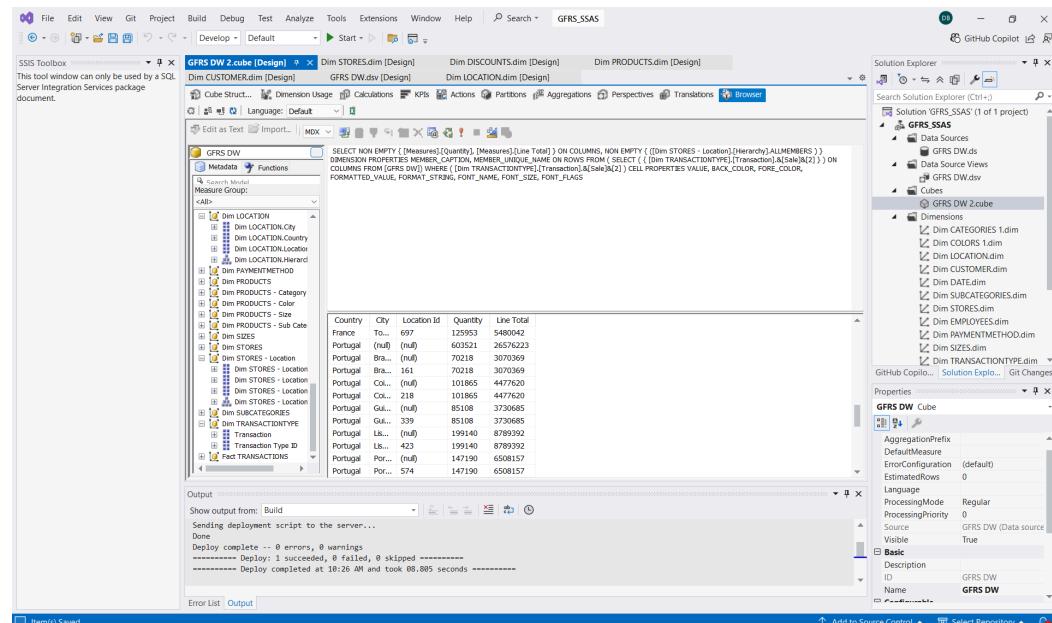


Hình 464. Câu truy vấn hiển thị doanh số roll up theo năm

#### III.2.2.2. Doanh số và số lượng sản phẩm roll up theo Quốc gia

- **Mục tiêu:** Hiển thị doanh số roll up theo từng quốc gia

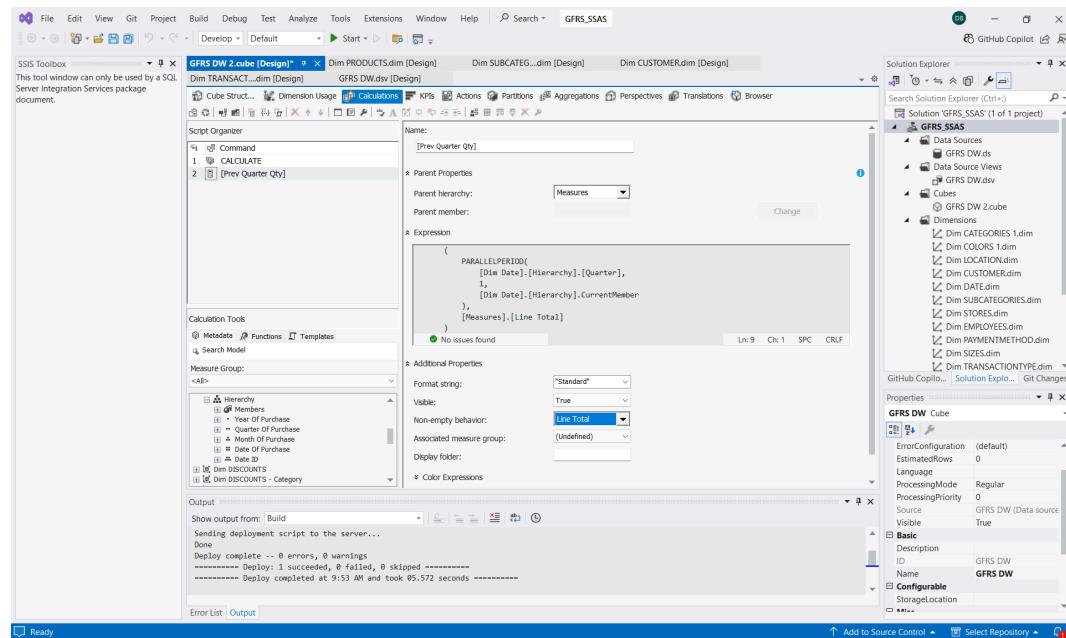
- **Ý nghĩa:** **tổng hợp** toàn bộ dữ liệu bán hàng để tính ra **tổng doanh thu** và **tổng số lượng sản phẩm** bán được tại từng **Quốc gia** cụ thể. Thao tác này giúp đánh giá quy mô và hiệu suất kinh doanh của từng thị trường quốc gia mà không cần đi sâu vào chi tiết vùng miền hay thành phố.
- **Công cụ thực hiện:** SSAS trong Visual Studio 2022
- **Thực hiện:** Kéo thuộc tính **Hierarchy** từ **Dim STORES - Location** vào khung truy vấn. Tiếp đó kéo **Measure Line Total, Quantity** ở mục **Fact TRANSACTIONS** vào khung. Ké tiếp, ta kéo **Transaction** từ **Dim TRANSACTIONTYPE** lên khung bộ lọc và chỉnh **Filter Expression** thành **Sale**. Sau đó, trên thành công cụ ta tắt chế độ **Design Mode**, khi này câu lệnh MDX kéo thả của ta vừa thực sẽ xuất hiện, ở đây phần ([**Dim STORES - Location**].[**Hierarchy**].[**Location Id**].**ALLMEMBERS**) ta sẽ xóa **[Location ID]** để thực hiện câu truy vấn Roll UP. Cuối cùng chọn **Click to execute the query** để thực hiện câu truy vấn.



*Hình 465. Câu truy vấn hiển thị doanh số và số lượng sản phẩm roll up theo quốc gia*

### III.2.2.3. Phân tích doanh số và tốc độ tăng trưởng theo các quý

- **Mục tiêu:** Hiển thị doanh số từng quý và tốc độ tăng trưởng của quý đó so với quý trước.
- **Ý nghĩa:** theo dõi biến động hiệu suất kinh doanh theo chu kỳ 3 tháng (Quý) để nhận diện tính mùa vụ và xu hướng ngắn hạn. Cột so sánh với quý trước đóng vai trò đo lường **đà tăng trưởng**, giúp xác định xem doanh nghiệp đang tăng tốc hay suy giảm so với giai đoạn liền kề.
- **Công cụ thực hiện:** SSAS trong Visual Studio 2022
- **Chuẩn bị:** Để thực hiện câu truy vấn này ta cần tạo **Measure Prev Quarter Qty** để tính doanh số của quý trước đó so với quý hiện tại, và **Measure Growth Rate** để tính tốc độ tăng trưởng. Vào tab **Calculations** của cube, ở ô **Script Organizer**, nhấn chuột phải chọn **New Calculated Member** và thực hiện thêm 2 Measure sau:



Hình 466. Tao Prev Quarter Qty mới

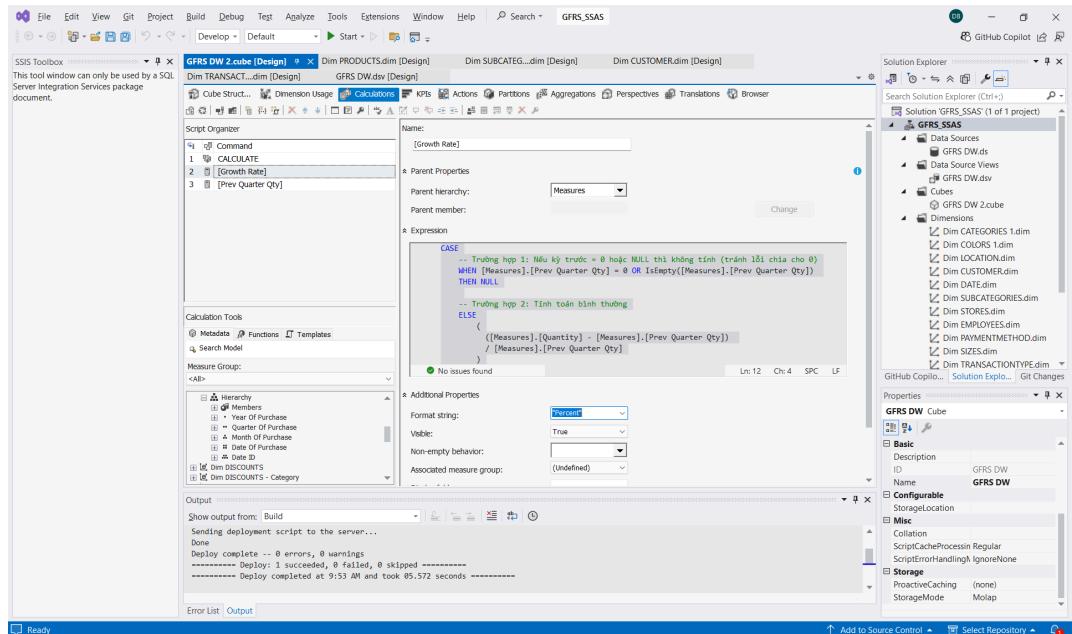
Bảng 28. Expression cho Prev Quarter Qty

$PARALLELPERIOD($
-------------------

```

[Dim DATE].[Hierarchy].[Quarter Of Purchase],
1,
[Dim Date].[Hierarchy].CurrentMember
),
[Measures].[Line Total]
)

```



Hình 467. Tạo Growth Rate mới

Bảng 29. Expression cho Growth Rate

(CASE	
-- Kiểm tra mẫu số bằng 0 hoặc Null	
WHEN [Measures].[Prev Quarter Qty] = 0 OR	
IsEmpty([Measures].[Prev Quarter Qty])	
THEN NULL	
-- Tính toán	
ELSE	

```

(
    ([Measures].[Line Total] - [Measures].[Prev Quarter Qty])
    / [Measures].[Prev Quarter Qty]
)
END)

```

- **Thực hiện:** Kéo thuộc tính **Year Of Purchase**, **Quarter Of Purchase** vào khung truy vấn. Tiếp đó kéo **Measure Line Total**, **Growth Rate** vào khung. Sau đó chọn **Click to execute the query** để thực hiện câu truy vấn.

The screenshot shows the SSAS ToolBox interface with the following details:

- Solution Explorer:** Shows the project structure under "GFRS\_SSAS".
- Toolbox:** Displays various MDX objects like Dimension, Hierarchy, Operator, Filter Expression, and Parameter.
- Query Results Grid:** Shows data for "Year Of Purchase" and "Quarter Of Purchase" with columns for "Line Total" and "Growth Rate".
- Properties Window:** Shows properties for the "GFRS DW Cube" including "Source: GFRS DW (Data source)" and "Visible: True".
- Output Window:** Displays the command history and completion status.

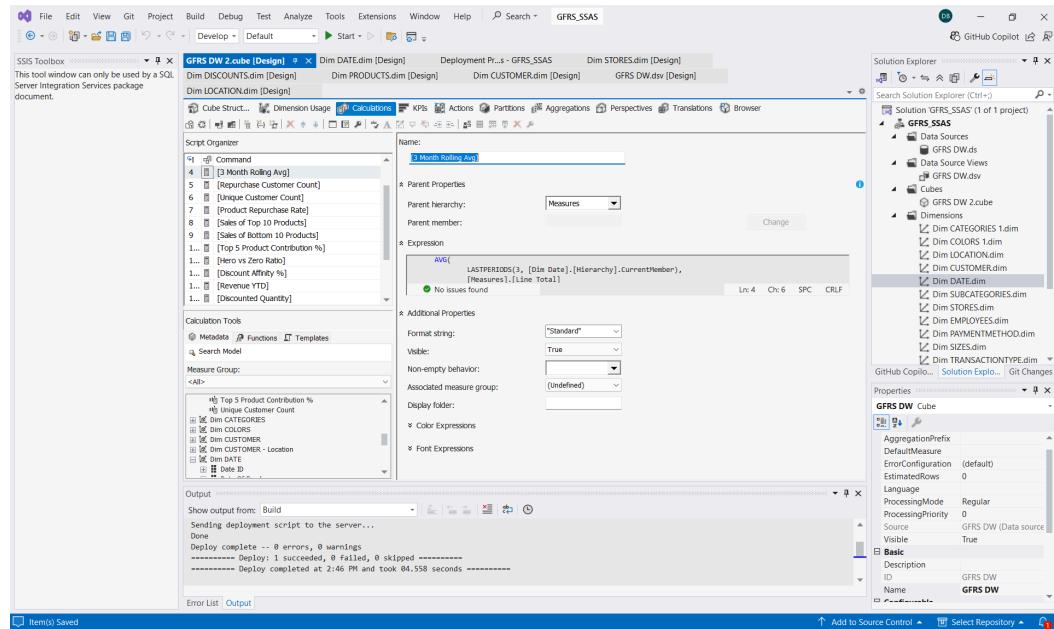
Year Of Purchase	Quarter Of Purchase	Line Total	Growth Rate
2023	1	22357198	(null)
2023	2	27295652	0.020960...
2023	3	27225652	-0.002370...
2023	4	46914678	0.7231785...
2024	1	26050773	0.4447202...
2024	2	29895928	0.1476023...
2024	3	31147512	0.0418646...
2024	4	52729965	0.6929109...
2025	1	18818983	0.6431064...
Unknown	Unknown	(null)	-1

Hình 468. Câu truy vấn hiển thị doanh số theo quý và tốc độ tăng trưởng của từng quý

#### III.2.2.4. Phân tích doanh số theo tháng

- **Mục tiêu:** Hiển thị doanh số theo tháng cùng với Đường trung bình động (Moving Average) và Doanh thu lũy kế từ đầu năm (YTD Revenue)
- **Ý nghĩa:** kết hợp quan sát xu hướng chủ đạo (đã loại bỏ các biến động nhiễu nhờ đường trung bình động) và theo dõi tiến độ tích lũy tài chính từ đầu năm đến thời điểm hiện tại. Việc này giúp đánh giá đồng thời đà tăng trưởng thực tế và khả năng hoàn thành mục tiêu tổng của cả năm.

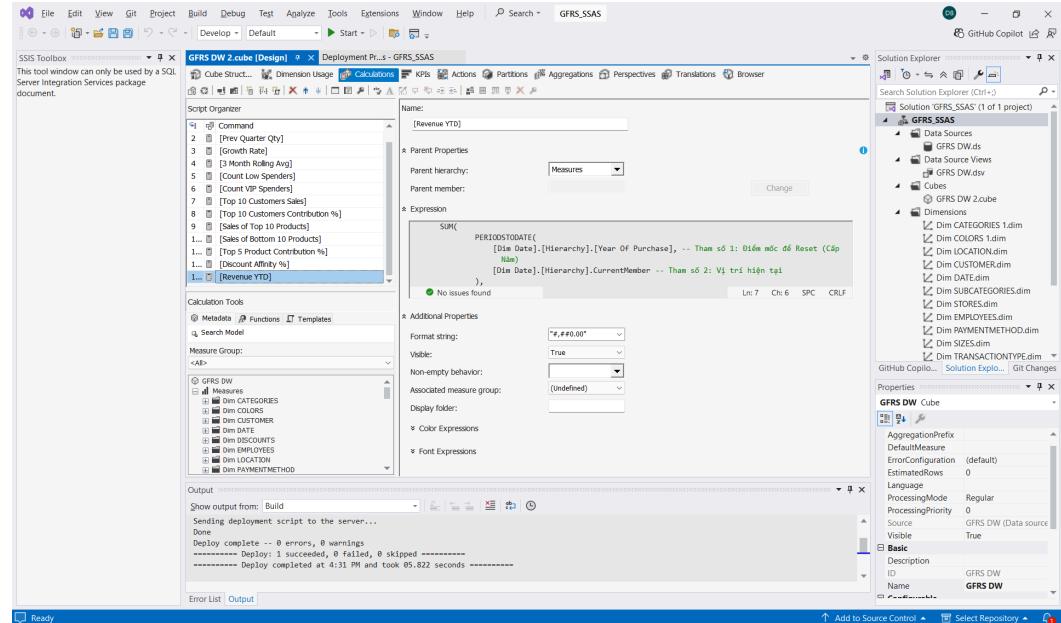
- **Công cụ thực hiện: SSAS trong Visual Studio 2022**
- **Chuẩn bị:** Để thực hiện câu truy vấn này ta cần tạo **Measure 3 Month Rolling Avg** để tính doanh số trung bình của 3 tháng gần nhất, và **Measure Revenue YTD** để doanh số từ đầu năm đến tháng hiện tại. Vào tab **Calculations** của cube, ở ô **Script Organizer**, nhấn chuột phải chọn **New Calculated Member** và thực hiện thêm 2 Measure sau:



Hình 469. Tạo Measure 3 Month Rolling Avg mới

Bảng 30. Expression Measure 3 Month Rolling Avg

$\text{AVG}($ $\text{LASTPERIODS}(3, [\text{Dim Date}].[\text{Hierarchy}].\text{CurrentMember}),$ $[\text{Measures}].[\text{Line Total}]$ $)$
--



Hình 470. Tạo Revenue YTD mới

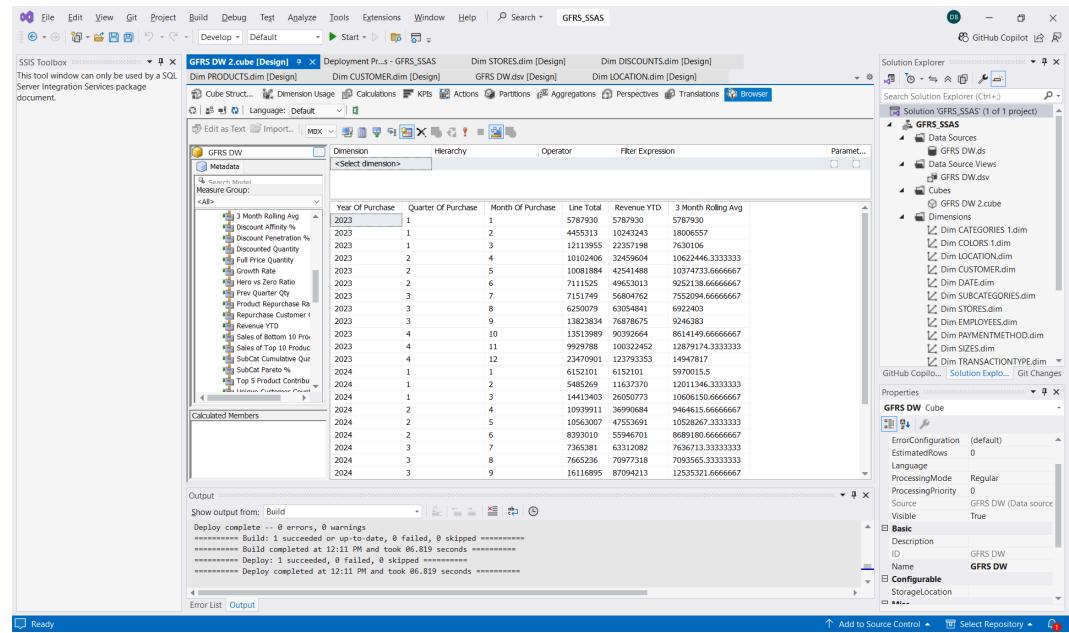
Bảng 31. Expression YTD Revenue

```

SUM(
    PERIODSTODATE(
        [Dim Date].[Hierarchy].[Year Of Purchase],
        [Dim Date].[Hierarchy].CurrentMember
    ),
    [Measures].[Line Total]
)

```

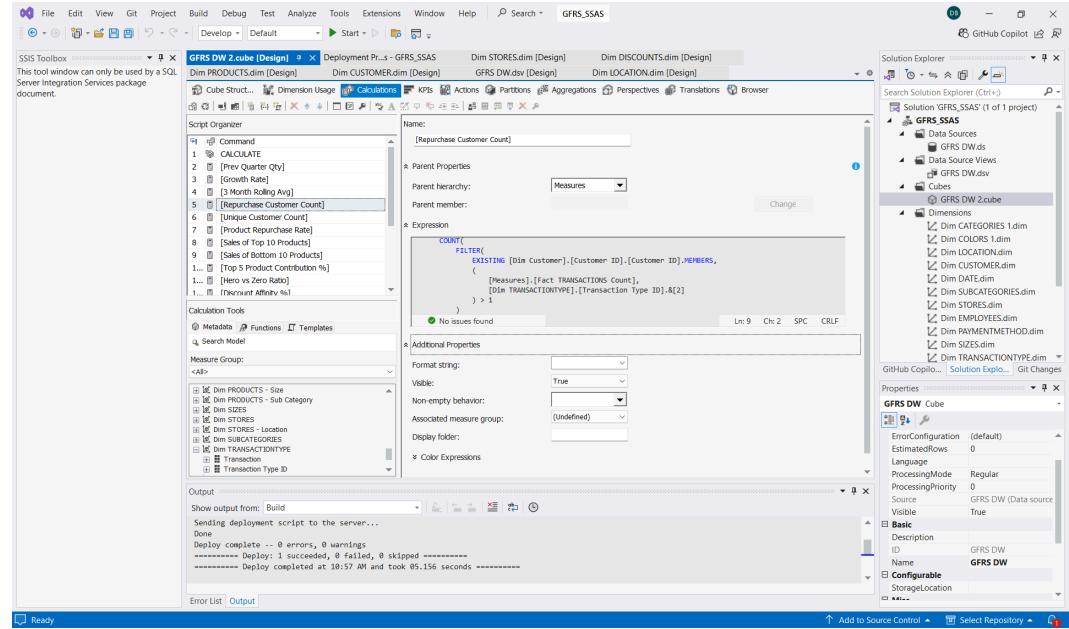
- **Thực hiện:** Kéo thuộc tính **Month Of Purchase**, **Quarter Of Purchase**, **Year Of Purchase** vào khung truy vấn. Tiếp đó kéo **Measure Line Total** ở mục **Fact TRANSACTIONS** vào khung. Kế tiếp, kéo **Revenue YTD**, **3 Month Rolling Avg**. Sau đó chọn **Click to execute the query** để thực hiện câu truy vấn.



Hình 471. Câu truy vấn hiển thị Doanh số theo tháng

### III.2.2.5. Phân tích sự yêu thích của khách hàng đối với các danh mục sản phẩm con vào năm 2024

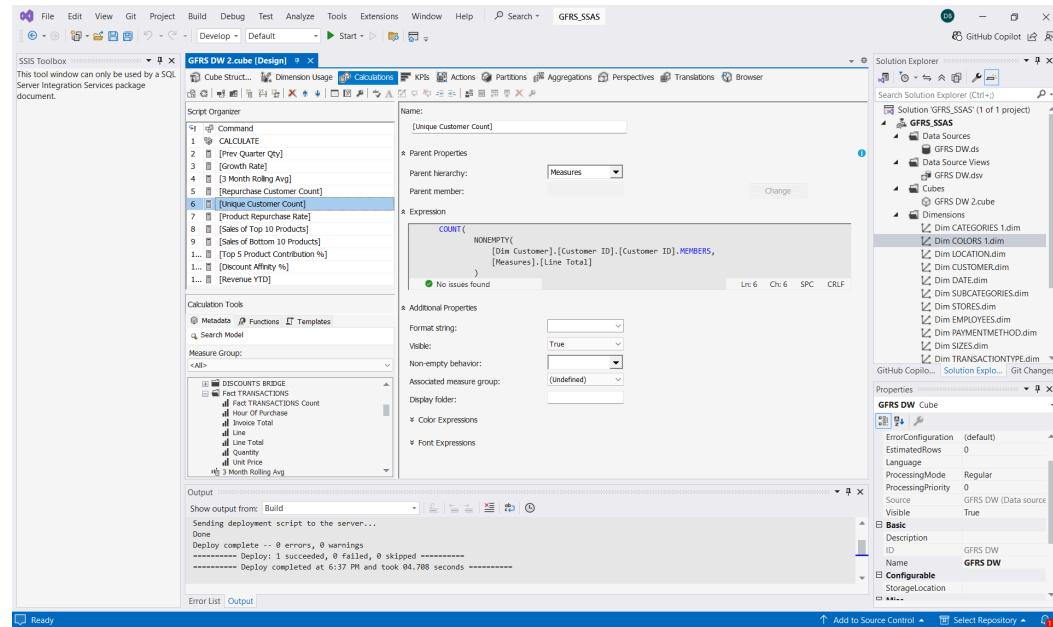
- **Mục tiêu:** Hiển thị doanh số các loại sản phẩm theo danh mục sản phẩm phụ, số giao dịch duy nhất, số giao dịch mua lại để tính sự yêu thích của khách hàng
- **Ý nghĩa:** Hiểu được mức độ trung thành và xu hướng quay lại của khách hàng đối với các danh mục sản phẩm phụ để từ đó tập trung tối ưu về hàng hóa hoặc thiết kế các chiến dịch khuyến mãi riêng cho từng danh mục sản phẩm.
- **Công cụ thực hiện:** SSAS trong Visual Studio 2022
- **Chuẩn bị:** Để thực hiện câu truy vấn này ta cần tạo **Measure Unique Customer Count** để tính tổng số khách hàng đã mua, **Measure Repurchase Customer Count** để đếm số khách hàng mua lại, và **Measure Product Repurchase Rate** để tính tỷ lệ yêu thích. Vào tab **Calculations** của cube, ở ô **Script Organizer**, nhấn chuột phải chọn **New Calculated Member** và thực hiện thêm 3 Measure sau:



Hình 472. Tạo Measure Repurchase Customer Count

Bảng 32. Expression Measure Repurchase Customer Count

```
COUNT(
    FILTER(
        EXISTING [Dim Customer].[Customer ID].[Customer ID].MEMBERS,
        (
            [Measures].[Fact TRANSACTIONS Count],
            [Dim TRANSACTIONTYPE].[TransactionTypeID].&[2]
        ) > 1
    )
)
```



Hình 473. Tạo Measure Unique Customer Count

Bảng 33. Expression Measure Unique Customer Count

COUNT(

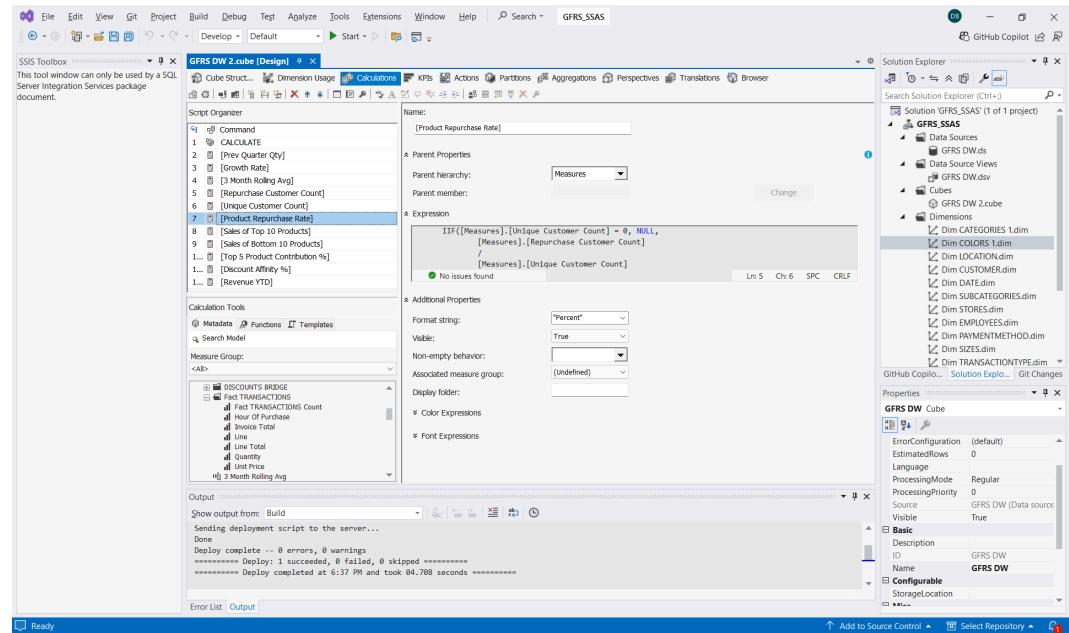
NONEMPTY(

[Dim Customer].[Customer ID].[Customer ID].MEMBERS,

[Measures].[Line Total]

)

)

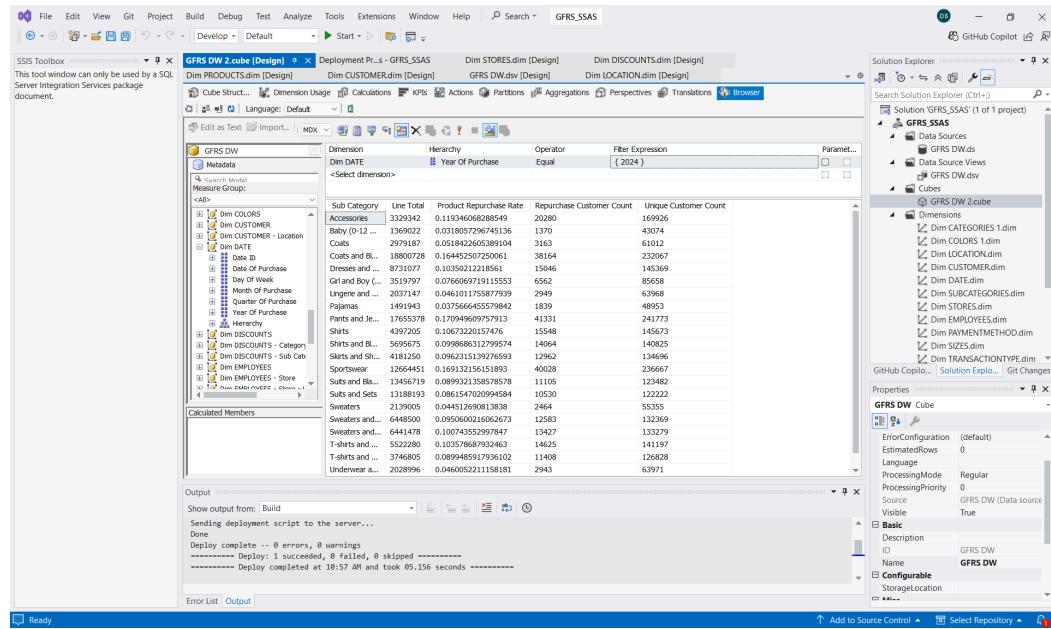


Hình 474. Tạo Measure Product Repurchase Rate

Bảng 34. Expression Measure Product Repurchase Rate

$\text{IIF}([\text{Measures}].[\text{Unique Customer Count}] = 0, \text{NULL},$ $[\text{Measures}].[\text{Repurchase Customer Count}]$ $/$ $[\text{Measures}].[\text{Unique Customer Count}]$ $ $
---

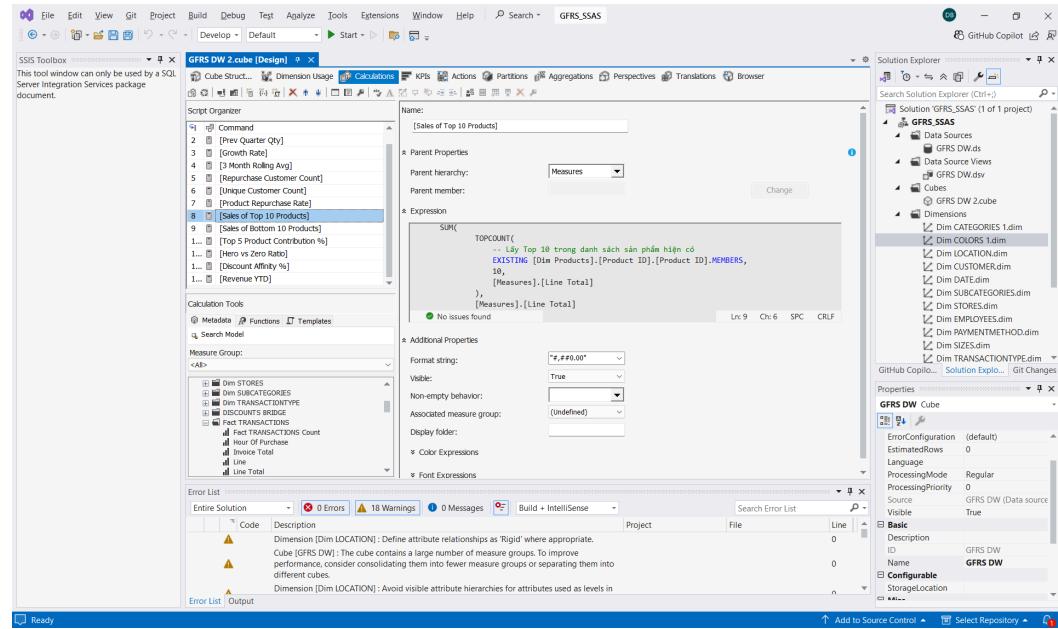
- **Thực hiện:** Kéo thuộc tính SubCategory từ Dim STORES – Sub Category vào khung truy vấn. Tiếp đó kéo Measure Line Total, Product Repurchase Rate, Repurchase Customer Count, Unique Customer Count ở mục Fact TRANSACTIONS vào khung. Ta kéo Year Of Purchase lên khung tham số bên trên, ở mục Filter Expression chọn 2024. Sau đó chọn Click to execute the query để thực hiện câu truy vấn.



Hình 475. Câu truy vấn hiển thị sự yêu thích của khách hàng đối với các danh mục sản phẩm con vào năm 2024

### III.2.2.6. So sánh sự bình đẳng trong các danh mục sản phẩm.

- **Mục tiêu:** Hiển thị doanh số của 10 sản phẩm có doanh số cao nhất, doanh số 10 sản phẩm thấp nhất, và tỷ lệ chênh lệch giữa chúng
- **Ý nghĩa:** làm rõ **sự phân hóa hiệu suất** cực đại trong danh mục hàng hóa, giúp nhận diện mức độ phụ thuộc vào nhóm sản phẩm chủ lực (“ngôi sao”) và gánh nặng từ nhóm kém hiệu quả. Tỷ lệ chênh lệch đóng vai trò là thước đo sự **mất cân đối**, giúp doanh nghiệp quyết định nên loại bỏ mã hàng yếu kém hay tối ưu hóa nguồn lực cho nhóm dẫn đầu.
- **Công cụ thực hiện:** SSAS trong Visual Studio 2022
- **Chuẩn bị:** Để thực hiện câu truy vấn này ta cần tạo **Measure Sales of Top 10 Products** để tính tổng doanh thu 10 sản phẩm bán chạy nhất, **Measure Sales of Bottom 10 Products** để tính tổng doanh thu 10 sản phẩm bán chậm nhất, và **Measure Hero vs Zero Ratio** để tính tỷ lệ chênh lệch giữa nhóm bán chạy và bán chậm. Vào tab **Calculations** của cube, ở ô **Script Organizer**, nhấn chuột phải chọn **New Calculated Member** và thực hiện thêm 3 Measure sau:



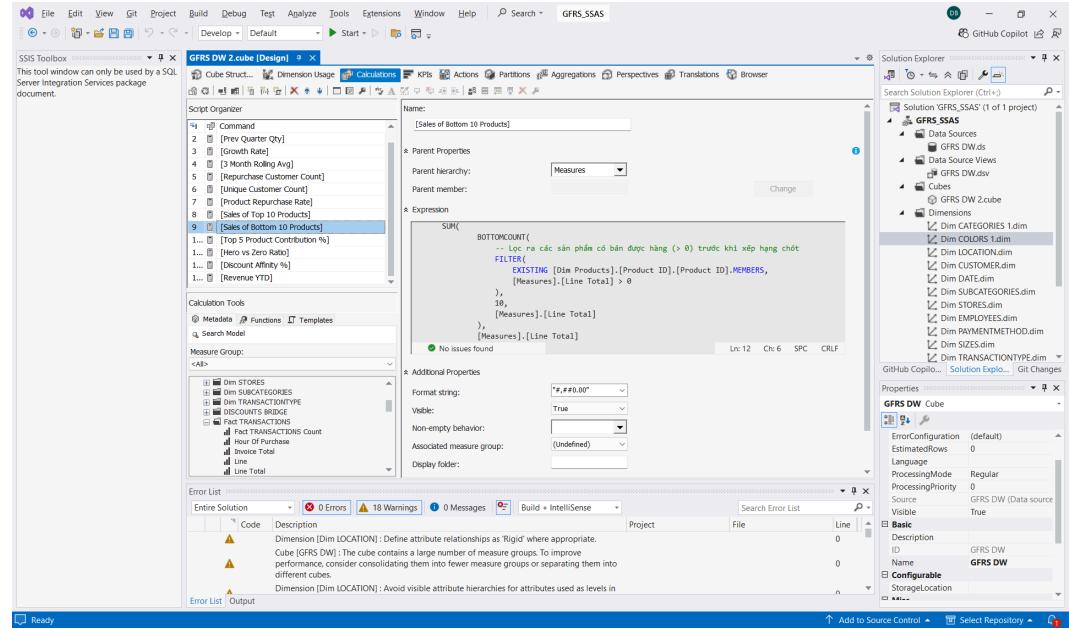
Hình 476. Tạo Measure Sales of Top 10 Products

Bảng 35. Expression Measure Sales of Top 10 Products

```

SUM(
    TOPCOUNT(
        EXISTING [Dim Products].[Product ID].[Product ID].MEMBERS,
        10,
        [Measures].[Line Total]
    ),
    [Measures].[Line Total]
)

```



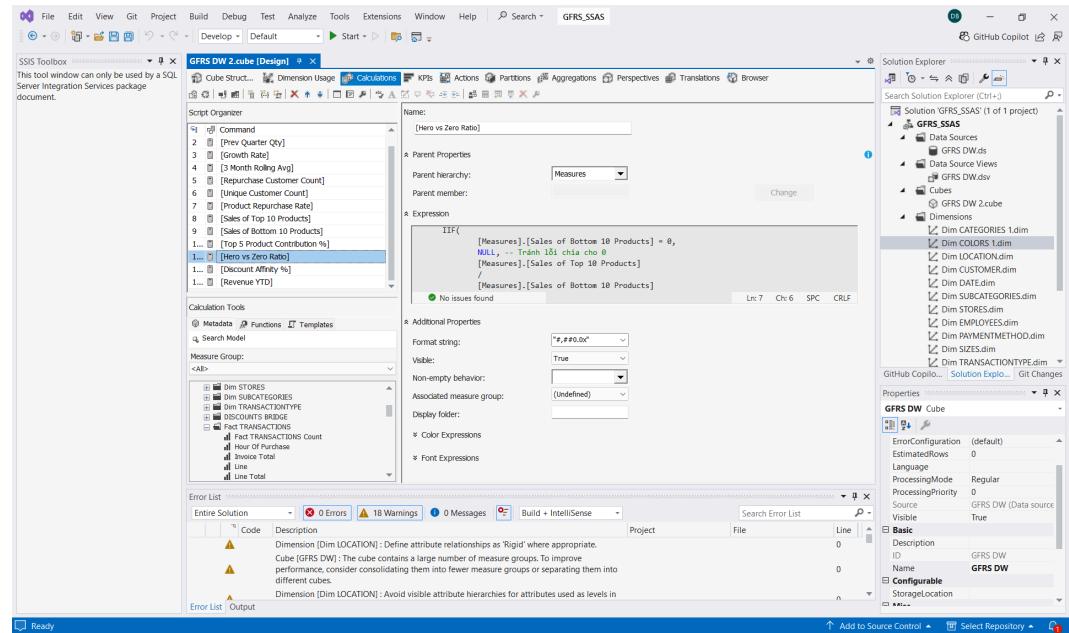
Hình 477. Tạo Measure Sales of Bottom 10 Products

Bảng 36. Expression Measure Sales of Bottom 10 Products

```

SUM(
    BOTTOMCOUNT(
        FILTER(
            EXISTING [Dim Products].[Product ID].MEMBERS,
            [Measures].[Line Total] > 0
        ), 10,
        [Measures].[Line Total]
    ),
    [Measures].[Line Total]
)

```

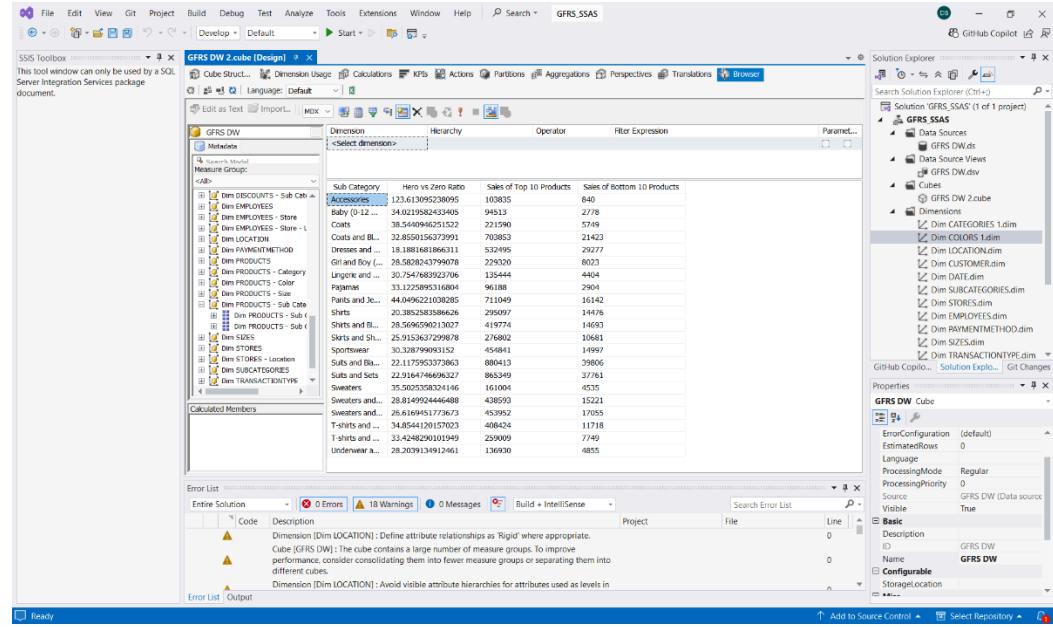


Hình 478. Tạo Measure Hero vs Zero Ratio

Bảng 37. Expression Measure Hero vs Zero Ratio

```
IIF(
    [Measures].[Sales of Bottom 10 Products] = 0,
    NULL,
    [Measures].[Sales of Top 10 Products]
    /
    [Measures].[Sales of Bottom 10 Products]
)
```

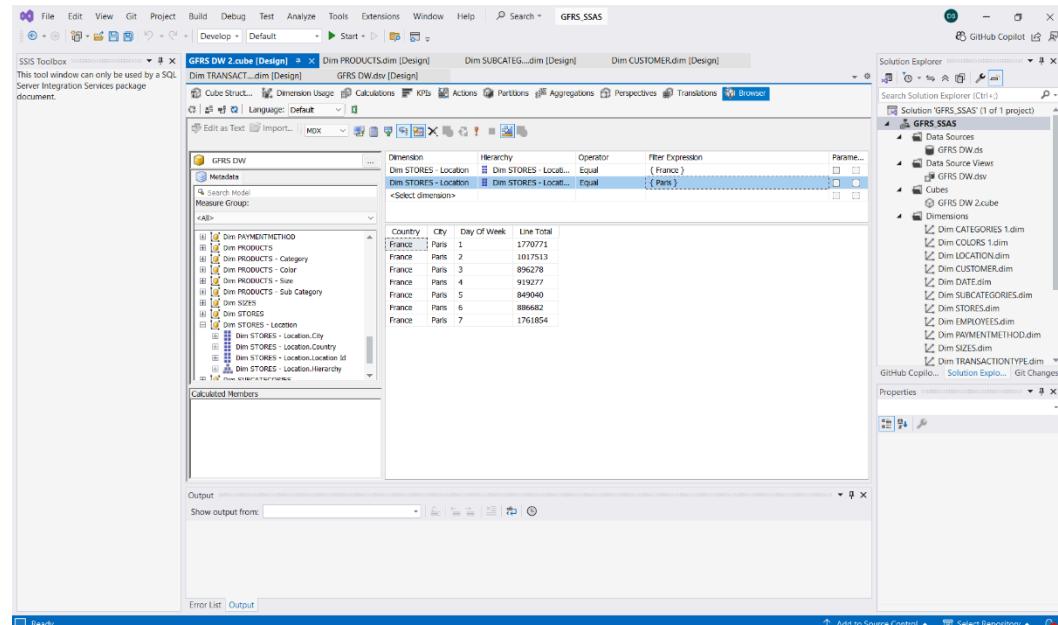
- **Thực hiện:** Kéo thuộc tính **Sub Category** từ **Dim PRODUCT – Sub Category** vào khung truy vấn. Tiếp đó kéo **Measure Hero vs Zero Ratio. Sales of Top 10 Products, Sales of Bottom 10 Products** ở mục **Fact TRANSACTIONS** vào khung. Sau đó chọn **Click to execute the query** để thực hiện câu truy vấn.



Hình 479. Câu truy vấn hiển thị sự chênh lệch trong các danh mục sản phẩm

### III.2.2.7. Các cửa hàng thuộc thành phố Paris ở Pháp theo ngày trong tuần

- **Mục tiêu:** Hiển thị doanh số các cửa hàng ở Paris ở pháp từng ngày trong tuần
- **Ý nghĩa:** So sánh hiệu suất cửa hàng ở Paris Pháp để điều phối, hỗ trợ vào những ngày có nhu cầu nhân sự cao.
- **Công cụ thực hiện:** SSAS trong Visual Studio 2022
- **Thực hiện:** Kéo thuộc tính **Country, City, Day Of Week** vào khung truy vấn. Tiếp đó kéo **Measure Line Total** ở mục **Fact TRANSACTIONS** vào khung. Ta kéo **Country** lên khung tham số bên trên, ở mục **Filter Expression** chọn **France** và kéo **City** sang, sau đó chỉnh **Filter Expression** thành **Paris**. Sau đó chọn **Click to execute the query** để thực hiện câu truy vấn.



Hình 480. Câu truy vấn hiển thị doanh số các cửa hàng ở Paris France theo các ngày trong tuần

### III.2.2.8. Doanh số theo SubCategory theo từng Năm

- **Mục tiêu:** Hiển thị doanh số các SubCategory của từng Năm
- **Ý nghĩa:**
  - o Báo cáo dạng bảng xoay → rất hữu ích cho SSAS / Power BI.
  - o Giúp tìm Category nào bán mạnh theo năm.
- **Công cụ thực hiện:** SQL Server Management Studio 21
- **Câu truy vấn:**

Bảng 38. Câu lệnh truy vấn hiển thị doanh số các SubCategory theo tháng

```

SELECT *
FROM (
    SELECT
        d.YearOfPurchase,
        c.SubCategory,
        SUM(f.LineTotal) AS Revenue
    FROM Fact_TRANSACTIONS f
)
```

```

JOIN Dim_DATE d ON f.DateID = d.DateID
JOIN Dim_PRODUCTS p ON f.ProductID = p.ProductID
JOIN Dim_SUBCATEGORIES c ON p.SubCategoryID = c.SubCategoryID
GROUP BY c.SubCategory, d.YearOfPurchase
) AS SourceData
PIVOT (
    SUM(Revenue)
    FOR YearOfPurchase IN ([2023],[2024],[2025])
) AS PivotTable
ORDER BY SubCategory;

```

```

File Edit View Query Git Project Tools Window Help Search Solution1
File Edit View Query Git Project Tools Window Help Search Solution1
Object Explorer
Connect D:\GRS\GFS\SSAS [GRS.DW]
Databases GRP_BITZ OLAP_CK_H2_2023_2024 Assemblies Management
Query.sql -> [Name: (TUV\maing)]
SELECT *
FROM (
    SELECT
        d.YearOfPurchase,
        c.SubCategory,
        SUM(f.LineTotal) AS Revenue
    FROM Fact.TRANSACTIONS f
    JOIN Dim_DATE d ON f.DateID = d.DateID
    JOIN Dim_PRODUCTS p ON f.ProductID = p.ProductID
    JOIN Dim_SUBCATEGORIES c ON p.SubCategoryID = c.SubCategoryID
    GROUP BY c.SubCategory, d.YearOfPurchase
) AS SourceData
PIVOT (
    SUM(Revenue)
    FOR YearOfPurchase IN ([2023], [2024], [2025])
) AS PivotTable
ORDER BY SubCategory;

```

SubCategory	2023	2024	2025
Accessories	289748	3320342	416264
Baby (0-12 months)	115002	136022	177002
Books	21400	21400	21400
Clothing	16883929	1880728	2737414
Dresses and Jumpsuits	775936	871977	1163749
Girl and Boy (1-5 years, 6-12 years)	292283	3519797	493631
Lingerie and Panties	105571	239147	27280
Plates	129600	148100	170000
Pants and Jeans	1584318	17853578	2265401
Shirts	383456	4397205	598116
Shirts and Blouses	5046921	5995475	788743
Socks and Gloves	388300	449100	51111
SourceData	1112793	12964451	1569712
Gulls and Diapers	117203	1345719	1527510
Shirt and Suits	1176329	1318193	1779205
Sweaters	1041418	214000	322443
T-Shirts and Knitwear	571425	644500	764444
Sweaters and Sweathirts	579256	644178	891401
T-Shirts and Polos	479444	5522200	736527
T-Shirts and Tops	3298030	3746809	479721

Query executed successfully.

Hình 481. Câu lệnh hiển thị doanh số các SubCategory theo năm

### III.2.2.9. Phân tích sở thích mua hàng theo Giới tính và SubCategory

- **Mục tiêu:** Hiển thị doanh số sản phẩm theo giới tính và loại sản phẩm
- **Ý nghĩa:** phân tích được thị trường từ đó chủ động được nguồn hàng, đồng thời tạo thêm các khuyến mãi hấp dẫn.

- Công cụ thực hiện: SQL Server Management Studio 21

- Câu lệnh truy vấn

Bảng 39. Câu lệnh truy vấn số lượng sản phẩm mua theo giới tính của từng loại sản phẩm

```
SELECT  
    SubCategory,  
    ISNULL([F], 0) AS Female,  
    ISNULL([M], 0) AS Male,  
    ISNULL([D], 0) AS Diverse  
FROM (  
    SELECT c.Gender, cat.SubCategory, f.Quantity  
    FROM Fact_TRANSACTIONS AS f  
    JOIN Dim_CUSTOMER AS c ON f.CustomerID = c.CustomerID  
    JOIN Dim_PRODUCTS AS p ON f.ProductID = p.ProductID  
    JOIN Dim_SUBCATEGORIES AS cat ON p.SubCategoryID =  
        cat.SubCategoryID  
    WHERE f.TransactionTypeID = 2  
) AS SourceData  
PIVOT (  
    SUM(Quantity)  
    FOR Gender IN ([F], [M], [D])  
) AS PivotTable  
ORDER BY SubCategory;
```

The screenshot shows a SQL Server Management Studio window with the following details:

- Object Explorer:** Shows the database structure for 'DUILY\MSQLSERVER2' (SQL Server 16). It includes databases like 'DUILY\maing\_2023\_2024', 'GRFS\_DW', and 'OLAP\_B11.DW'. There are also security and integration services.
- SQL Query Editor:** Contains the following T-SQL code:
 

```

SELECT
    SubCategory,
    ISNULL([F], 0) AS Female,
    ISNULL([M], 0) AS Male,
    ISNULL([D], 0) AS Diverse -- Giả sử D là mã cho giới tính khác

FROM (
    SELECT
        c.Gender,
        cat.SubCategory,
        f.Quantity
    FROM Fact_Transactions AS f
    JOIN Dim_Customer AS c ON f.CustomerID = c.CustomerID
    JOIN Dim_Products AS p ON f.ProductID = p.ProductID
    JOIN Dim_SubCategories AS cat ON p.SubCategoryID = cat.SubCategoryID
    WHERE f.TransactionTypeID = 2
) AS SourceData
PIVOT (
    SUM(Quantity)
) BY SubCategory
      
```
- Results Grid:** Displays the output of the query. The columns are 'SubCategory', 'Female', 'Male', and 'Diverse'. The data shows various product categories and their sales volume for different genders. For example, Baby (0-12 months) has 295 female sales, 52 male sales, and 0 diverse sales.
- Status Bar:** Shows 'Query executed successfully.' and other system information.

Hình 482. Câu lệnh truy vấn hiển thị số sản phẩm của

### III.2.2.10. Doanh số của các loại hình thanh toán ở các thành phố

- **Mục tiêu:** Hiển thị doanh số thanh toán bằng tiền mặt và thẻ ở từng thành phố
- **Ý nghĩa:**
  - o Chủ động nguồn tiền mặt ở các khu vực dùng nhiều tiền mặt
  - o Chủ động các thiết bị thanh toán trực tuyến để đáp ứng nhu cầu khách hàng
- **Công cụ thực hiện: SQL Server Management Studio 21**
- **Câu lệnh truy vấn:**

Bảng 40. Câu lệnh truy vấn hiển thị doanh số thanh toán bằng thẻ và tiền mặt ở các thành phố

```

SELECT
    City, ISNULL([Cash], 0) AS PaidByCash,
    ISNULL([Credit Card], 0) AS PaidByCreditCard
FROM (
    
```

```

SELECT l.City, pm.PaymentMethod, f.LineTotal
FROM Fact_TRANSACTIONS AS f
JOIN Dim_STORES AS s ON f.StoreID = s.StoreID
JOIN Dim_LOCATION AS l ON s.LocationID = l.Location_id
JOIN Dim_PAYMENTMETHOD AS pm ON f.PaymentMethodID =
pm.PaymentMethodID
) AS SourceData
PIVOT (
    SUM(LineTotal)
    FOR PaymentMethod IN ([Cash], [Credit Card])
) AS PivotTable
ORDER BY City;

```

The screenshot shows the SSMS interface with the following details:

- Object Explorer:** Shows the database structure, including Databases (GRFS\_SSAS, OLAP\_BT2), Management (DUMY\_MSSQLSERVER2), and System Databases.
- Query Editor:** Contains the T-SQL query provided in the code block.
- Results Pane:** Displays the execution results in a table format:

City	PaidByCash	PaidByCreditCard
1 London	2783697	11160158
2 São Paulo	3703222	14809398
3 Auckland	3677486	14473818
4 Newark	3006800	12104682
5 Barcelona	7383777	2961633
6 Berlin	2348196	9530499
7 Birmingham	959872	3544153
8 Bengaluru	591238	23069587
9 Bristol	951545	3738030
10 Chicago	2124145	8481064
11 Copenhagen	587493	3536116
12 Dublin	1901214	7773290
13 Franklin Park	671082	2632968
14 Glasgow	508951	2299686
15 Guanajuato	857854	2832568

Hình 483. Câu lệnh truy vấn hiển thị Doanh số trả bằng tiền mặt hoặc bằng thẻ của từng thành phố

### III.2.3. Câu lệnh truy vấn MDX

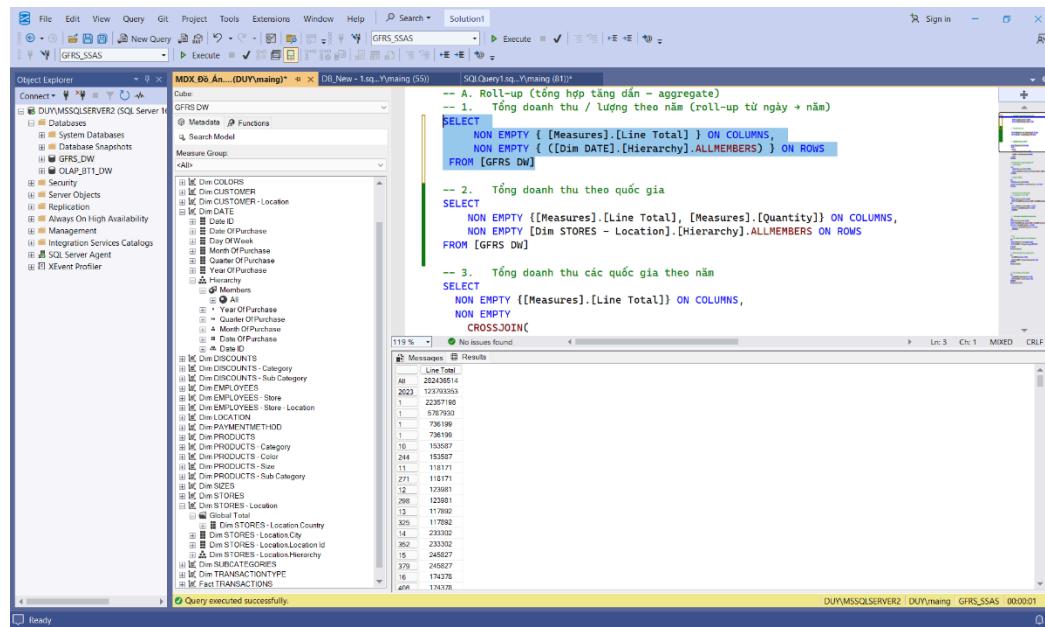
#### III.2.3.1. Tổng Doanh số roll up theo năm

- Câu lệnh truy vấn:

Bảng 41. Câu lệnh truy vấn MDX thứ 1

```
SELECT  
    NON EMPTY { [Measures].[Line Total] } ON COLUMNS,  
    NON EMPTY { ([Dim DATE].[Hierarchy].ALLMEMBERS) }  
ON ROWS  
FROM [GFRS DW]
```

- Thực hiện:



Hình 484. Kết quả thực hiện câu lệnh truy vấn MDX thứ 1

#### III.2.3.2. Tổng Doanh số và số lượng sản phẩm roll up theo quốc gia

- Câu lệnh truy vấn:

Bảng 42. Câu lệnh truy vấn MDX thứ 2

```
SELECT  
    NON EMPTY {[Measures].[Line Total]} ON COLUMNS,
```

```

NON EMPTY {[Dim STORES -
Location].[Hierarchy].ALLMEMBERS} ON ROWS
FROM [GFRS DW]
WHERE ([Dim TRANSACTIONTYPE].[Transaction Type ID].&[2]);

```

- **Thực hiện:**

```

File Edit View Query Git Project Tools Window Help Search Solution1
DUY\maing Databases System Databases Database Snapshots DW_GRS_CK_HQ_2023_2024 DW_GRS_CK_HQ_2023_2024 DW_GRS_CK_HQ_2023_2024 DW_GRS_CK_HQ_2023_2024 Security Server Objects Replication Always On High Availability Management Integration Services Catalogs SQL Server Agent XEvent Profiler
MDX_D9_An... (DUY\maing)* SQLQuery1ao...Yuming (76)
Object Explorer
Connect GRS DW
Metadata Functions
Search Model
Measure Group
Cube
GRS DW
Measures
KPIs
Dim CATEGORIES
Dim COLORS
Dim CUSTOMER
Dim CUSTOMER - Location
Dim DATE
Dim DISCOUNTS
Dim DISCOUNTS - Category
Dim DISCOUNTS - Sub-Category
Dim EMPLOYEES
Dim EMPLOYEES - Store
Dim EMPLOYEES - Store - Location
Dim LOCATION
Dim PAYMENTMETHOD
Dim PRODUCTS
Dim PRODUCTS - Category
Dim PRODUCTS - Color
Dim PRODUCTS - Size
Dim PRODUCTS - Sub-Category
Dim SIZES
Dim STORES
Dim STORES - Location
Dim SURCATEGORIES
Dim TRANSACTIONTYPE
Dim TRANSACTIONTYPE - Transaction Type ID
Dim Fact TRANSACTIONS
-- 1. Tổng doanh thu theo năm (Roll-up từ ngày + năm)
SELECT
NON EMPTY { [Measures].[Line Total] } ON COLUMNS,
NON EMPTY { ([Dim DATE].[Hierarchy].ALLMEMBERS) } ON ROWS
FROM [GRS DW];
-- 2. Tổng doanh thu theo quốc gia
SELECT
NON EMPTY {[Measures].[Line Total], [Measures].[Quantity]} ON COLUMNS,
NON EMPTY {[Dim STORES - Location].[Hierarchy].ALLMEMBERS ON ROWS}
FROM [GRS DW];
-- 3. Doanh số , số lượng sản phẩm và tốc độ tăng trưởng của từng quý
SELECT
NON EMPTY { [Measures].[Line Total], [Measures].[Quantity], [Measures].[Growth Rate] }
NON EMPTY { ([Dim DATE].[Year Of Purchase].ALLMEMBERS * [Dim DATE].[Quarter Of Purchase].ALLMEMBERS) } ON COLUMNS
FROM [GRS DW]
Results
Line Total Quantity
All 29135608 6656234
A.Séc 79847137 1035960
A.Séc 19595139 369454
6 19585112 369454
*+* 19154112 369454
38 16077876 320597
** 16077876 320597
60 16077876 320597
*+* 14774010 267488
16 17678945 267488
41* 10250591 267591
209 10250591 267591
34142592 781572
Deutschland 0 0
Berlin 12437092 781572
139 12437092 781572
Frankfurt am Main 3494249 79029
305 3494249 79029
Hamburg 7496324 160491
365 7893237 160491
xMn 5662746 125748

```

Hình 485. Kết quả thực hiện câu lệnh truy vấn MDX thứ 2

### III.2.3.3. Doanh số và tốc độ tăng trưởng của từng quý

- **Câu lệnh truy vấn:**

Bảng 43. Câu lệnh truy vấn MDX thứ 3

```

SELECT
NON EMPTY { [Measures].[Line Total], [Measures].[Growth Rate] }
ON COLUMNS,
NON EMPTY { ([Dim DATE].[Year Of Purchase].[Year Of
Purchase].ALLMEMBERS * [Dim DATE].[Quarter Of
Purchase].[Quarter Of Purchase].ALLMEMBERS) } ON ROWS
FROM [GRS DW]

```

- **Thực hiện:**

The screenshot shows the Microsoft SQL Server Management Studio (SSMS) interface. The title bar indicates the connection is to 'GRS\_SSAS' and the query type is 'MDX'. The Object Explorer on the left shows the database structure, including 'GRS DW' and various dimensions like 'Dim STORES', 'Dim EMPLOYEES', and 'Dim TRANSACTIONS'. The main pane displays an MDX query:

```
-- 2. Tông doanh thu theo quốc gia  
SELECT  
    NON EMPTY {[Measures].[Line Total], [Measures].[Quantity]} ON COLUMNS,  
    NON EMPTY [Dim STORES - Location].[Hierarchy].ALLMEMBERS ON ROWS  
FROM [GRS DW]  
WHERE ( [Dim TRANSACTIONTYPE].[Transaction Type ID], &[2] );  
  
-- 3. Doanh số và tốc độ tăng trưởng của từng quý  
SELECT  
    NON EMPTY { [Measures].[Line Total], [Measures].[Growth Rate] } ON COLUMNS,  
    NON EMPTY { ([Dim DATE].[Year Of Purchase].[Year Of Purchase].ALLMEMBERS * [Dim DATE].[Year Of Purchase]) } ON ROWS  
FROM [GRS DW];  
  
-- 4. Doanh số theo tháng  
SELECT  
    NON EMPTY { [Measures].[Line Total], [Measures].[3 Month Rolling Avg], [Measures].[Revenue] } ON COLUMNS,  
    NON EMPTY { ([Dim DATE].[Year Of Purchase].[Year Of Purchase].ALLMEMBERS * [Dim DATE].[Year Of Purchase]) } ON ROWS  
FROM [GRS DW];
```

The results grid shows the following data:

	Line Total	Growth Rate
2023	1 22297199 (null)	22.09%
2023	2 2795815 22.09%	-0.23%
2023	3 2725862 -0.23%	72.32%
2023	4 4691467 72.32%	-0.47%
2024	1 2609374 72.32%	14.56%
2024	2 20898529 14.56%	4.19%
2024	3 31147512 4.19%	69.99%
2024	4 52729965 69.99%	-64.31%
Unknown	Unknown (null)	-100.00%

At the bottom, a message bar says 'Query executed successfully.'

Hình 486. Kết quả thực hiện câu lệnh truy vấn MDX thứ 3

### **III.2.3.4. Doanh số theo tháng**

#### - Câu lệnh truy vấn:

Bảng 44. Câu lệnh truy vấn MDX thứ 4

```
SELECT  
    NON EMPTY { [Measures].[Line Total], [Measures].[3 Month  
Rolling Avg],[Measures].[Revenue YTD] } ON COLUMNS,  
    NON EMPTY { ([Dim DATE].[Year Of Purchase].[Year Of  
Purchase].ALLMEMBERS * [Dim DATE].[Month Of  
Purchase].[Month Of Purchase].ALLMEMBERS ) } ON ROWS  
FROM [GFRS DW]
```

- #### - Thực hiện:

```

-- 3. Doanh số và tốc độ tăng trưởng của từng quý
SELECT
    NON EMPTY { [Measures].[Line Total], [Measures].[Growth Rate] } ON COLUMNS,
    NON EMPTY { [Dim DATE].[Year Of Purchase].[Year Of Purchase].ALLMEMBERS * [Dim DATE].[Year Of Purchase].ALLMEMBERS }
    FROM [GFRS DW]

-- 4. Doanh số theo tháng
SELECT
    NON EMPTY { [Measures].[Line Total], [Measures].[3 Month Rolling Avg], [Measures].[Revenue YTD] } ON COLUMNS,
    NON EMPTY { {[Dim DATE].[Year Of Purchase].[Year Of Purchase].ALLMEMBERS * [Dim DATE].[Year Of Purchase].ALLMEMBERS} } ON ROWS
    FROM [GFRS DW]

-- 5. Sự yêu thích của khách hàng năm 2024
SELECT
    NON EMPTY { {[Measures].[Line Total], [Measures].[3 Month Rolling Avg], [Measures].[Revenue YTD]} } ON COLUMNS,
    NON EMPTY { {[Dim PRODUCTS - Sub Category].[Sub Category].ALLMEMBERS} } ON ROWS
    FROM [GFRS DW]

```

Messages Results

	Line Total	3 Month Rolling Avg.	Revenue YTD
2023 1	5797930	5,797,930.00	5,797,930.00
2023 10	13013890	11,165,967.33	90,392,664.00
2023 11	96297930	10,567,050.00	105,264,050.00
2023 12	23477901	13,823,228.00	123,793,253.00
2023 2	4460013	8,171,671.90	10,243,243.00
2023 3	1211955	7,452,399.33	22,357,196.00
2023 4	10102404	8,155,508.33	32,200,508.00
2023 5	10001044	8,760,100.67	42,541,480.00
2023 6	7111525	8,086,050.00	49,653,013.00
2023 7	7151740	8,115,052.67	56,804,780.00
2023 8	6250070	8,837,784.33	63,054,841.00
2023 9	16102080	7,680,500.00	76,780,500.00
2023 10	61231241	13,136,263.33	8,152,013.00
2024 10	19133746	12,991,959.00	102,207,959.00
2024 11	11442782	14,251,134.33	113,736,721.00
2024 12	20693457	17,571,653.00	138,024,178.00
2024 2	6468000	11,938,500.00	11,938,500.00
2024 3	14413403	8,653,591.00	26,056,773.00
2024 4	10939911	9,327,952.67	36,990,694.00
2024 6	10543007	11,972,107.00	47,553,691.00

Hình 487. Kết quả thực hiện câu lệnh truy vấn MDX thứ 4

### III.2.3.5. Phân tích sự yêu thích của khách hàng đối với các danh mục

#### sản phẩm con vào năm 2024

##### - Câu lệnh truy vấn:

Bảng 45. Câu lệnh truy vấn MDX thứ 5

```

SELECT
    NON EMPTY { [Measures].[Line Total],[Measures].[Product
Repurchase Rate],[Measures].[Repurchase Customer
Count],[Measures].[Unique Customer Count] } ON COLUMNS,
    NON EMPTY { {[Dim PRODUCTS - Sub Category].[Sub
Category].ALLMEMBERS} } ON ROWS
    FROM [GFRS DW]
    WHERE ( [Dim DATE].[Year Of Purchase].&[2024] );

```

##### - Thực hiện

The screenshot shows the Microsoft SQL Server Management Studio (SSMS) interface. The title bar reads "File Edit View Query Project Tools Extensions Window Help Search Solution1". The left pane is the "Object Explorer" showing database structures for "DUMMYSQLSERVER2" and "GFRS\_SSAS". The right pane is the "MDX Query" editor with the following query:

```
-- 3. Doanh số , số lượng sản phẩm và tốc độ tăng trưởng của từng quý
SELECT
NON EMPTY { [Measures].[Line Total], [Measures].[Quantity], [Measures].[Growth Rate] }
NON EMPTY { {[Dim DATE].[Year Of Purchase].<Year Of Purchase>.ALLMEMBERS * [Dim DATE].[Year Of Purchase].ALLMEMBERS} }
FROM [GFRS DW]

-- 4. Doanh số theo tháng
SELECT
NON EMPTY { [Measures].[Line Total], [Measures].[3 Month Rolling Avg], [Measures].[Revenue] }
NON EMPTY { {[Dim DATE].[Year Of Purchase].<Year Of Purchase>.ALLMEMBERS * [Dim DATE].[Year Of Purchase].ALLMEMBERS} }
FROM [GFRS DW]

-- 5. Số yêu thích của Khách hàng năm 2024
SELECT
NON EMPTY { [Measures].[Line Total], [Measures].[Product Repurchase Rate], [Measures].[Revenue] }
NON EMPTY { {[Dim PRODUCTS - Sub Category].[Sub Category].ALLMEMBERS} ON ROWS
FROM [GFRS DW]
WHERE ([Dim DATE].[Year Of Purchase].<2024>) ;

-- C. Slice & Dice (cắt lát, bóc lọc theo điều kiện)
-- 1. Lọc chỉ lấy Gender = 'F' và Year = 2024
SELECT
NON EMPTY { [Measures].[Line Total] } ON COLUMNS,
```

The bottom status bar indicates "Query executed successfully." and shows connection details: "DUMMYSQLSERVER2 DUMMaing GFRS\_SSAS 000156".

Hình 488. Kết quả thực hiện câu lệnh truy vấn MDX thứ 5

### **III.2.3.6. Sự chênh lệch trong các danh mục sản phẩm**

#### - Câu lệnh truy vấn:

Bảng 46. Câu lệnh truy vấn MDX thứ 6

```
SELECT  
    NON EMPTY { [Measures].[Hero vs Zero Ratio], [Measures].[Sales  
of Top 10 Products], [Measures].[Sales of Bottom 10 Products] } ON  
COLUMNS,  
    NON EMPTY { ([Dim PRODUCTS - Sub Category].[Sub  
Category].[Sub Category].ALLMEMBERS ) } ON ROWS  
FROM [GFRS DW];
```

- #### - Thực hiện

The screenshot shows the Microsoft SQL Server Management Studio (SSMS) interface. In the Object Explorer, the connection is set to 'DUY\MSQLSERVER2 (Microsoft Analysis Services)'. The 'MDX 06\_Ánh... (DUY\maing)\*' query window is active, displaying three MDX queries labeled 5, 6, and 7. The results pane shows a table with columns: Hero vs Zero Ratio, Sales of Top 10 Products, and Sales of Bottom 10 Products. The table contains data for various product categories. The status bar at the bottom indicates 'Query executed successfully.'

```

-- 5. Sự yêu thích của Khách hàng năm 2024
SELECT
NON EMPTY {[Measures].[Line Total],[Measures].[Product Repurchase Rate],[Measures].[Repurchase Rate]} ON COLUMNS
NON EMPTY {[Dim PRODUCTS - Sub Category].[Sub Category].ALLMEMBERS} ON ROWS
FROM [GFRS DW]
WHERE {[Dim DATE].[Year Of Purchase].&[2024]};

-- 6. Sự chênh lệch trong các danh mục sản phẩm
SELECT
NON EMPTY {[Measures].[Hero vs Zero Ratio],[Measures].[Sales of Top 10 Products],[Measures].[Sales of Bottom 10 Products]} ON COLUMNS
NON EMPTY {[Dim PRODUCTS - Sub Category].[Sub Category].ALLMEMBERS} ON ROWS
FROM [GFRS DW];

-- 7. Hiển thị doanh thu ở Allaua Pháp theo các ngày trong tuần
SELECT
NON EMPTY {[Measures].[Line Total]} ON COLUMNS,
NON EMPTY {[Dim LOCATION].[Country].[Country].ALLMEMBERS * [Dim LOCATION].[City].[City]} ON COLUMNS
FROM ( SELECT ( {[Dim LOCATION].[City].&[Paris]} ) ) ON COLUMNS
FROM [GFRS DW];

```

	Hero vs Zero Ratio	Sales of Top 10 Products	Sales of Bottom 10 Products
Accessories	123.6x	103,835.00	\$40.00
Baby (0-12 months)	34.0x	94,513.00	2,770.00
Body Care	35.8x	24,590.00	5,760.00
Clothing	23.2x	70,300.00	2,430.00
Dresses and Jumpsuits	18.2x	53,406.00	29,777.00
Girl and Boy (1-5 years, 6-14 years)	28.6x	229,320.00	8,023.00
Lingerie and Pajamas	30.8x	135,444.00	4,491.00
Pearmies	23.1x	98,181.00	2,640.00
Playwear and Jeans	44.0x	71,549.00	18,141.00
Shirts	20.4x	295,097.00	14,478.00
Shirts and Blouses	28.6x	419,774.00	14,693.00
Shorts and Shorts	25.9x	278,802.00	10,681.00

Hình 489. Kết quả thực hiện câu lệnh truy vấn MDX thứ 6

### III.2.3.7. Hiển thị Doanh số các cửa hàng ở Paris Pháp theo các ngày trong tuần

#### - Câu lệnh truy vấn:

Bảng 47. Câu lệnh truy vấn MDX thứ 7

```

SELECT
NON EMPTY {[Measures].[Line Total]} ON COLUMNS,
NON EMPTY {[Dim
LOCATION].[Country].[Country].ALLMEMBERS * [Dim
LOCATION].[City].[City].ALLMEMBERS * [Dim DATE].[Day Of
Week].[Day Of Week].ALLMEMBERS } ON ROWS
FROM ( SELECT ( {[Dim LOCATION].[City].&[Paris]} ) ) ON
COLUMNS
FROM ( SELECT ( {[Dim LOCATION].[Country].&[France]} ) )
ON COLUMNS
FROM [GFRS DW])

```

#### - Thực hiện:

```

NON EMPTY { {[Dim LOCATION].[Country].[Country].ALLMEMBERS * [Dim CUSTOMER].[Gender].[]} ON COLUMNS
SELECT {[Measures].[Line Total]} ON COLUMNS
FROM ( SELECT {[Dim LOCATION].[City].&[Paris]} ) ON COLUMNS
FROM [GFRS DW])

-- 2. Hiển thị doanh thu ở Allauch Pháp theo các ngày trong tuần
SELECT
NON EMPTY { {[Measures].[Line Total]} } ON COLUMNS,
NON EMPTY { {[Dim LOCATION].[Country].[Country].ALLMEMBERS * [Dim LOCATION].[City].[City].[]} } ON COLUMNS
FROM ( SELECT ( {[Dim LOCATION].[Country].&[France]} ) ) ON COLUMNS
FROM ( SELECT ( {[Dim LOCATION].[Country].&[France]} ) ) ON COLUMNS
FROM [GFRS DW]

-- D. Pivot
-- 1. Pivot: Categories on columns, Years on rows (hoặc ngược lại)

```

Query executed successfully.

	Line Total		
France	Paris	1	6333753
France	Paris	2	3393205
France	Paris	3	3373912
France	Paris	4	31432094
France	Paris	5	30571104
France	Paris	6	31034509
France	Paris	7	60920837

Hình 490. Kết quả thực hiện câu lệnh truy vấn MDX thứ 7

### III.2.3.8. Hiển thị Doanh số các doanh mục sản phẩm theo năm

- Câu lệnh truy vấn:

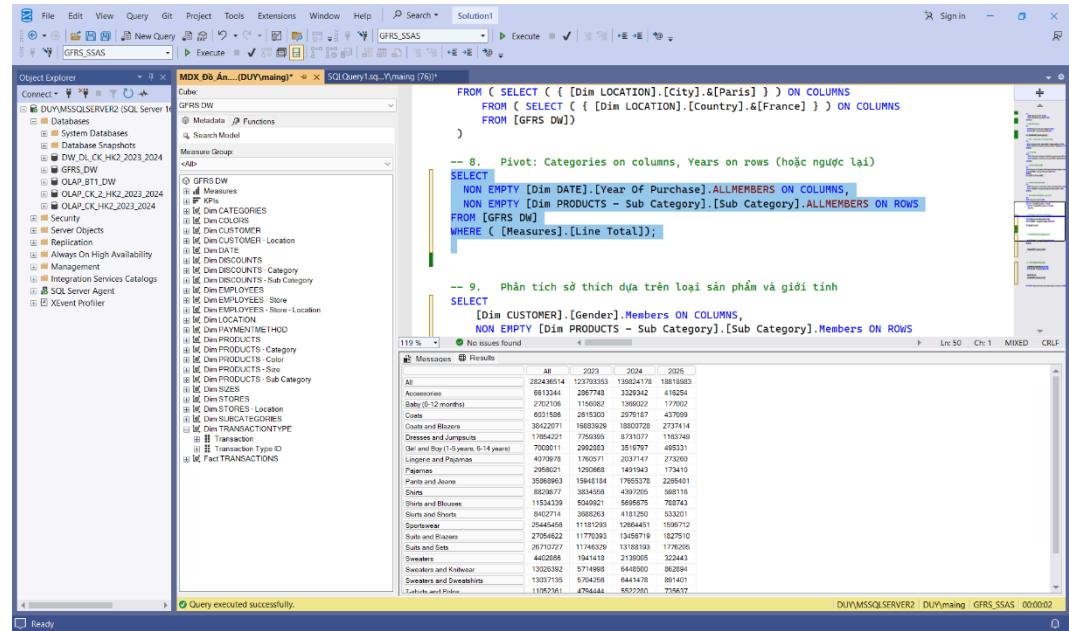
Bảng 48. Câu lệnh truy vấn MDX thứ 8

```

SELECT
    NON EMPTY [Dim DATE].[Year Of Purchase].ALLMEMBERS ON
    COLUMNS,
    NON EMPTY [Dim PRODUCTS - Sub Category].[Sub
    Category].ALLMEMBERS ON ROWS
    FROM [GFRS DW]
    WHERE ( [Measures].[Line Total]);

```

- Thực hiện:



Hình 491. Kết quả thực hiện câu lệnh truy vấn MDX thứ 8

### III.2.3.9. Doanh số dựa trên giới tính và loại sản phẩm

#### - Câu lệnh thực hiện

Bảng 49. Câu lệnh truy vấn MDX thứ 9

```

SELECT
[Dim CUSTOMER].[Gender].Members ON COLUMNS,
NON EMPTY [Dim PRODUCTS - Sub Category].[Sub
Category].Members ON ROWS
FROM [GFRS DW]
WHERE (
[Measures].[Quantity],
[Dim TRANSACTIONTYPE].[Transaction Type ID].&[2]
);
  
```

#### - Thực hiện:

```

File Edit View Query Git Project Tools Extensions Window Help Search Solution
GFRS_SSAS Execute ✓
Object Explorer
Connect → GFRS_SSAS
D:\DUV\MSQLSERVER2 (Microsoft Analysis Services)
GFRS_SSAS
GFRS_BI2
OLAP_CK_HK2_2023_2024
Assemblies
Management
MDX Bå An... (DUV\maing) ×
Cube: GFRS DW
Measure Group: All
-- 9. Doanh số dựa trên loại sản phẩm và giới tính
SELECT
    [Dim CUSTOMER].[Gender].Members ON COLUMNS,
    NON EMPTY [Dim PRODUCTS - Sub Category].[Sub Category].Members ON ROWS
FROM [GFRS DW]
WHERE (
    [Measures].[Line Total]
);

-- 10. Pivot với nhiều measures (Sales, Quantity)

```

All	Sales	Quantity	M	Unknown
All	282430514	179308	17952422	(null)
Accessories	6813344	4862	198614	2409868
Boys (0-12 months)	2702106	1301	1717004	92721
Clothes	28422671	1303	198614	2301009
Girls and Babies	36422671	24587	24450410	13019034
Dresses and Jumpsuits	17964221	15326	1135216	6286979
Girl and Boy (1-5 years, 6-14 years)	7008111	9788	4448183	2961410
Lingerie and Pyjamas	4970500	2725	2950385	1421295
Apparel	282430514	179308	17952422	1080000
Pants and Jeans	35069593	21324	2201297	13048572
Shirts	8128677	5560	5529966	3297313
Shirts and Blouses	11354339	7313	7415006	4111502
Shirts and Shorts	11354339	542	5421500	2370000
Apparel	25445459	17326	16138007	6286211
Suits and Blazers	27954622	10043	1690438	10132241
Suits and Sets	26710727	14044	17202291	9483402
Swimwear	4402656	239	26010995	1600231
Sweatshirts and Hoodies	13037135	7371	8304484	4800000
Sweaters and Sweating	13037135	7371	8151242	4078522
T-shirts and Prints	11952361	7183	6827967	4123101

Hình 492. Kết quả thực hiện câu lệnh truy vấn MDX thứ 9

### III.2.3.10. Hiển thị doanh số các phương thức thanh toán ở các thành phố

#### - Câu lệnh truy vấn:

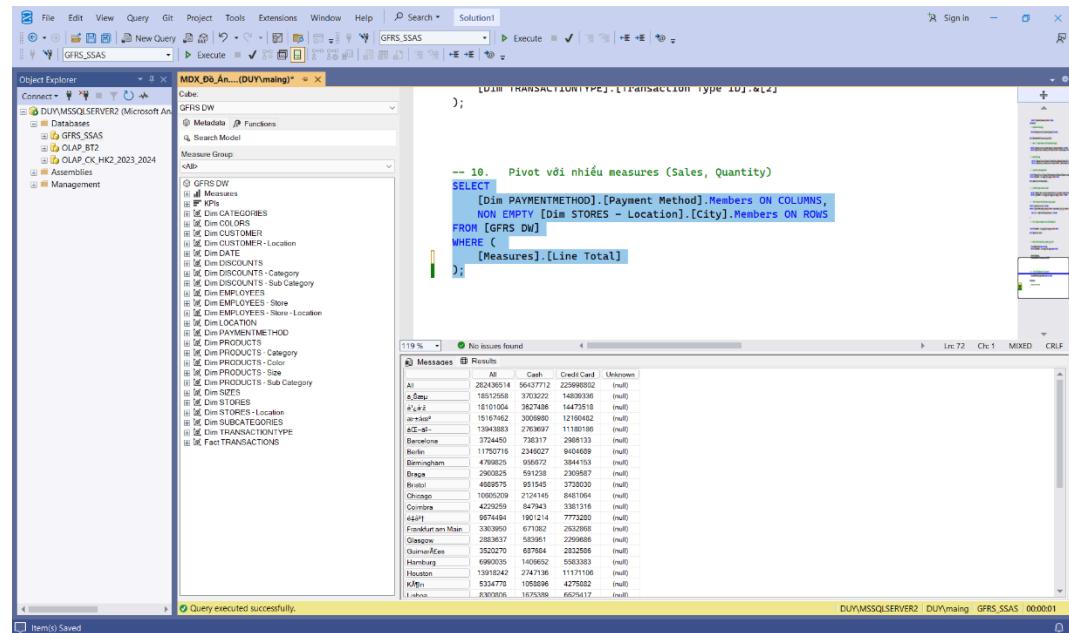
Bảng 50. Câu lệnh truy vấn MDX thứ 10

```

SELECT
    [Dim PAYMENTMETHOD].[Payment Method].Members ON
    COLUMNS,
    NON EMPTY [Dim STORES - Location].[City].Members ON
    ROWS
FROM [GFRS DW]
WHERE ( [Measures].[Line Total]);

```

#### - Thực hiện



Hình 493. Kết quả thực hiện câu lệnh truy vấn MDX thứ 10

### III.2.3.11. Top 1 danh mục phụ sản phẩm bán chạy nhất theo quý năm 2023 (yêu cầu bổ sung)

- Câu lệnh truy vấn:

```

SELECT
    ([Measures].[Quantity],[Dim
TRANSACTIONTYPE].[Transaction].&[Sale]&[2]) ON COLUMNS,
    GENERATE(
        [Dim DATE].[Hierarchy].[2023].CHILDREN,
        CROSSJOIN(
            {[Dim DATE].[Hierarchy].CURRENTMEMBER},
            TOPCOUNT(
                [Dim PRODUCTS - Sub Category].[Sub Category].CHILDREN,
                1,
                ([Measures].[Quantity],[Dim
TRANSACTIONTYPE].[Transaction].&[Sale]&[2])
            )
        )
    ) ON ROWS
    FROM [GFRS DW]
```

- Thực hiện:

```

[Measures].[Line Total]
);

-- Yêu cầu bổ sung
-- 1. Lấy top 1 danh mục sản phẩm phụ bán chạy nhất theo từng quý năm 2023
SELECT [Measures].[Quantity],[Dim TRANSACTIONTYPE].[Transaction].&[Sale]&[2]) ON COLUMNS,
GENERATE (
[Dim DATE].[Hierarchy].[2023]).CHILDREN,
CROSSJOIN(
{[Dim DATE].[Hierarchy].CURRENTMEMBER},
TOPCOUNT(
[Dim PRODUCTS - Sub Category].[Sub Category].CHILDREN,
1,
([Measures].[Quantity],[Dim TRANSACTIONTYPE].[Transaction].&[Sale]&[2]))
)
) ON ROWS
FROM [GFRS DW]

-- 2. Lấy top 3 danh mục sản phẩm có doanh số nhất vào năm 2023 và 2024
SELECT [Measures].[Line Total] ON COLUMNS,
TOPCOUNT(
[Measures].[Line Total]
[Dim PRODUCTS - Sub Category].[Sub Category].CHILDREN,
3,
[Measures].[Line Total]
) ON ROWS
FROM [GFRS DW]
WHERE {[Dim DATE].[Hierarchy].[Year Of Purchase].&[2023],[Dim DATE].[Hierarchy].[Year Of Purchase].&[2024]}

```

Messages Results

Category	Value	Count
Coats and Blazers	\$8332	1
Pants and Jeans	\$8937	2
Shirts	\$6768	3
Sweatshirts	\$13100	

Hình 494. Kết quả thực hiện câu lệnh truy vấn MDX thứ 11

### III.2.3.12. Top 3 danh mục phụ sản phẩm có doanh số cao nhất trong năm 2023, 2024

- Câu lệnh truy vấn:

```

SELECT [Measures].[Line Total] ON COLUMNS,
TOPCOUNT(
[Dim PRODUCTS - Sub Category].[Sub Category].CHILDREN,
3,
[Measures].[Line Total]
) ON ROWS
FROM [GFRS DW]
WHERE {[Dim DATE].[Hierarchy].[Year Of Purchase].&[2023],[Dim DATE].[Hierarchy].[Year Of Purchase].&[2024]}

```

- Thực hiện:

```

SELECT [Measures].[Line Total] ON COLUMNS,
TOPCOUNT(
    [dim PRODUCTS - Sub Category].[Sub Category].CHILDREN,
    3,
    ([Measures].[Quantity],[Dim TRANSACTIONTYPE].[Transaction]&[Sale]&[2])
)
) ON ROWS
FROM [GFRS DW]
-- 2. Lấy top 3 danh mục sản phẩm có doanh số nhất vào năm 2023 và 2024
SELECT [Measures].[Line Total] ON COLUMNS,
TOPCOUNT(
    [dim PRODUCTS - Sub Category].[Sub Category].CHILDREN,
    3,
    [Measures].[Line Total]
)
) ON ROWS
FROM [GFRS DW]
WHERE {[Dim DATE].[Hierarchy].[Year Of Purchase]&[2023], [Dim DATE].[Hierarchy].[Year Of Purchase]&[2024]}

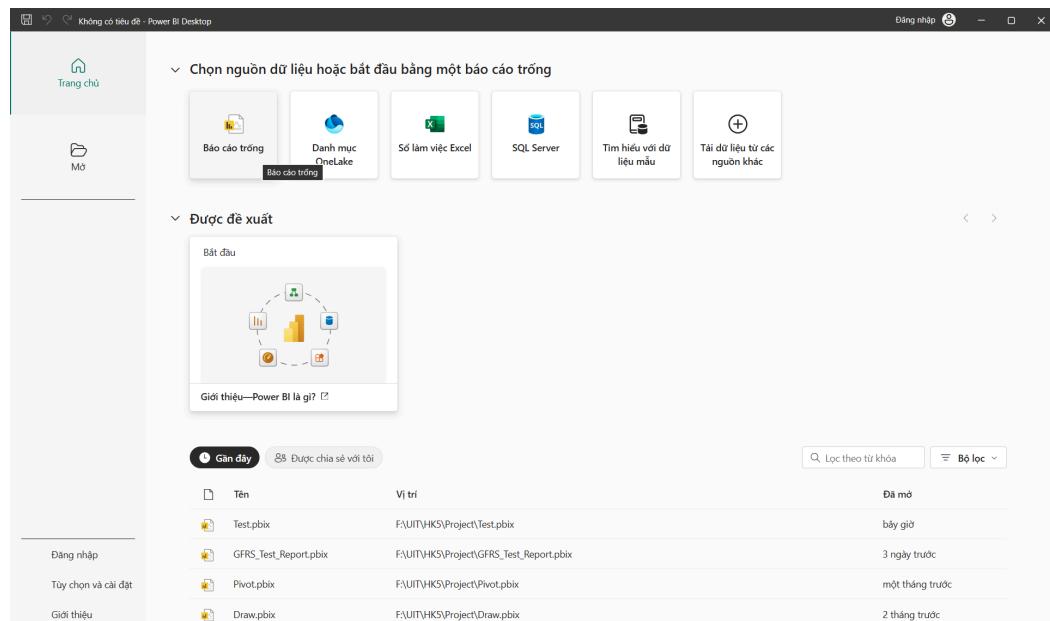
```

Hình 495. Kết quả thực hiện câu lệnh truy vấn MDX thứ 12

### III.2.4. Report

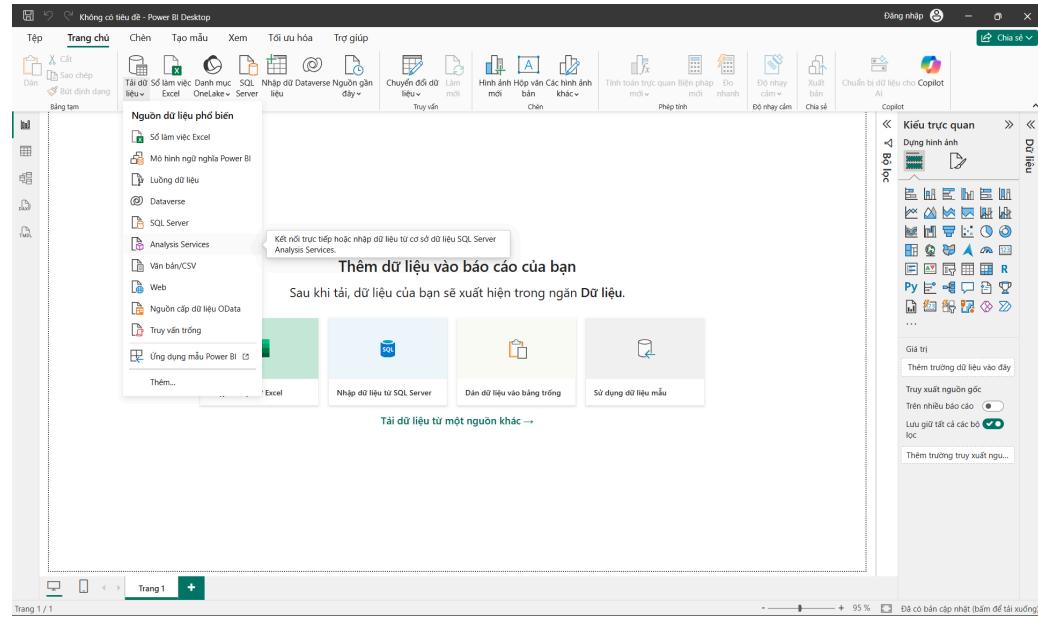
#### III.2.4.1. Tạo và kết nối với Analysis Services

- **Bước 1:** Mở phần mềm Power BI, sau đó chọn **Báo cáo trống**



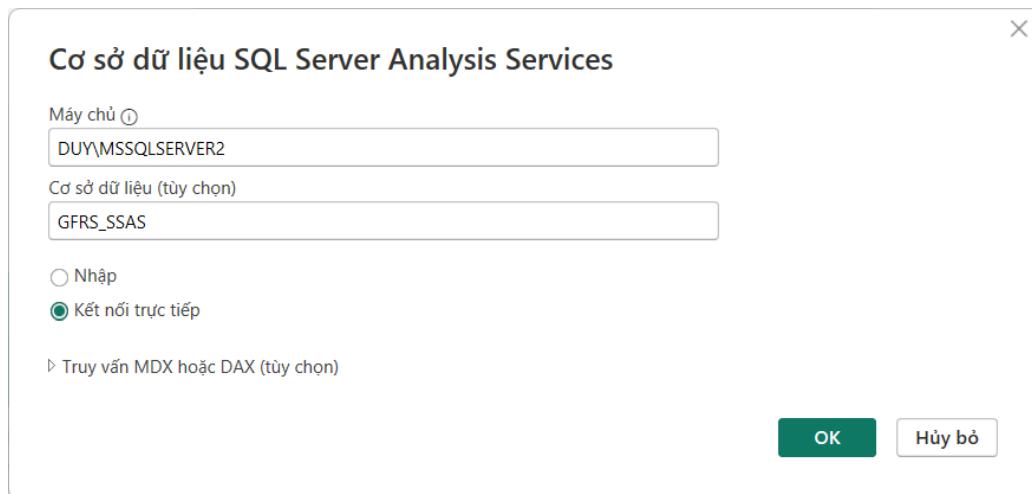
Hình 496. Tạo Báo cáo mới trong Power BI

- **Bước 2:** Ở Trang chủ, ta chọn **Tải dữ liệu**  Analysis Services



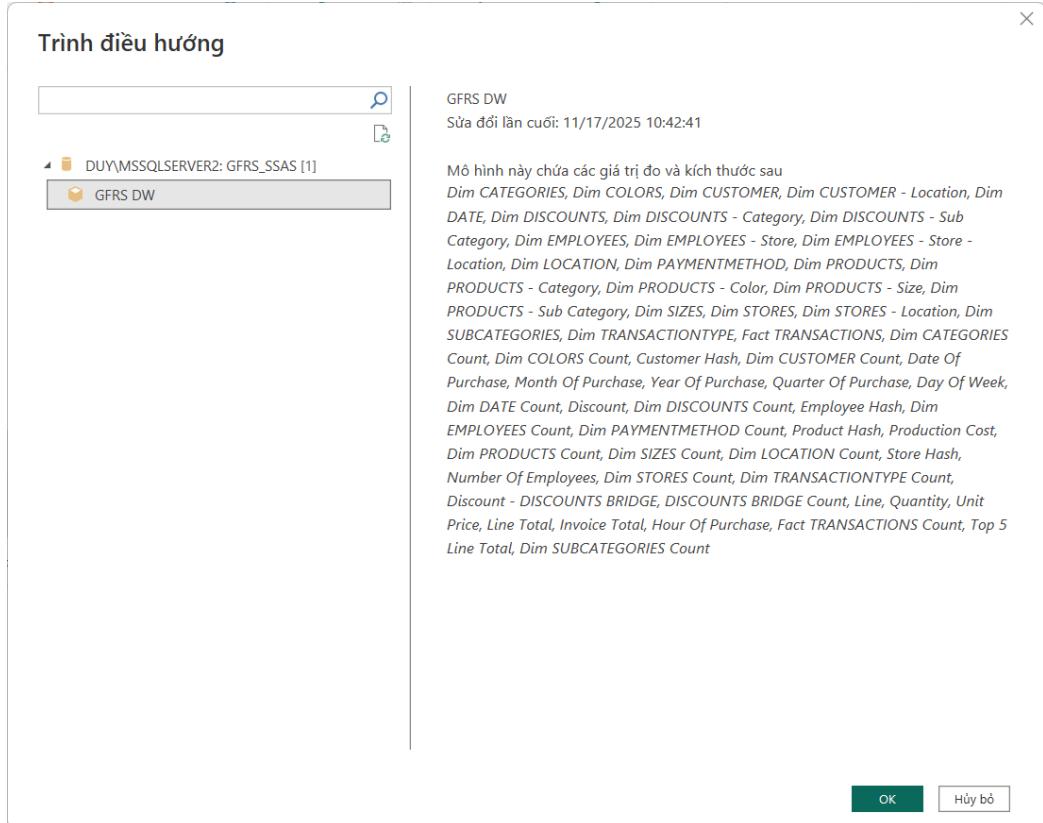
Hình 497. Chọn chức năng Analysis Services

- **Bước 3:** Cửa sổ Cơ sở dữ liệu SQL Server Analysis Services sẽ hiện lên, ta nhập đúng tên máy chủ (**DUY\MSSQLSERVER2**) và tên cơ sở dữ liệu (**GFRS\_SSAS**), sau đó nhấn **OK** để tiếp tục.



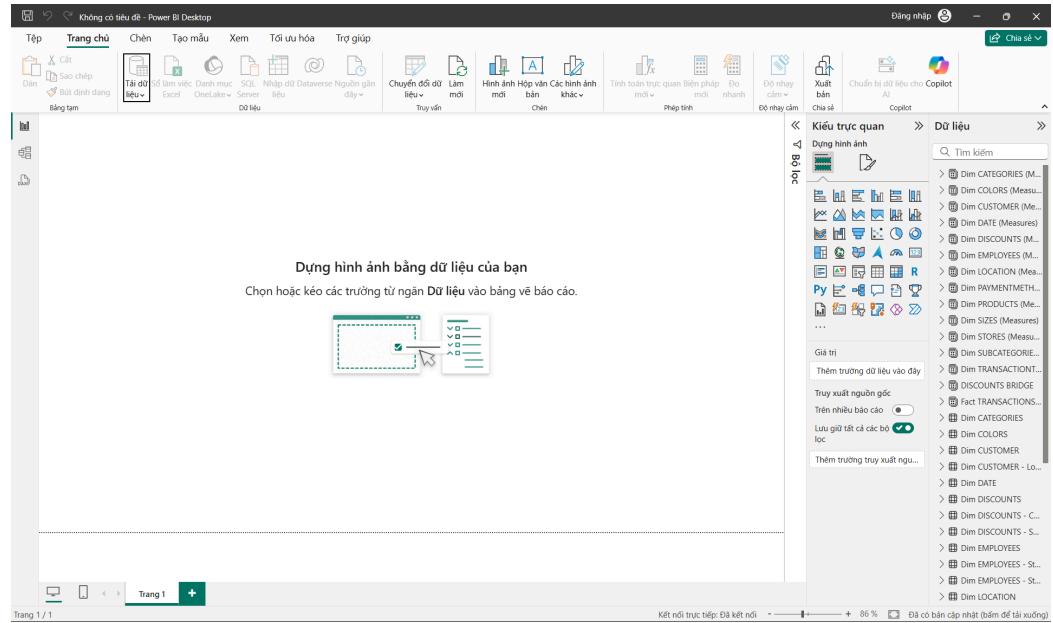
Hình 498. Nhập tên máy chủ và cơ sở dữ liệu cho báo cáo

- **Bước 4:** Cửa sổ Trình điều hướng sẽ hiện lên, ta chọn đúng **Data Warehouse (GFRS\_DW)** đã tạo, sau đó nhấn **Ok** để tiếp tục.



Hình 499. Chọn kho dữ liệu trong Trình điều hướng

Sau khi kết nối thành công, ở cửa sổ làm việc sẽ xuất hiện một **Tab Dữ liệu**, tab này chứa các **Measures**, các **Dim**, **Fact** của chúng ta đã tạo ở phần SSAS.

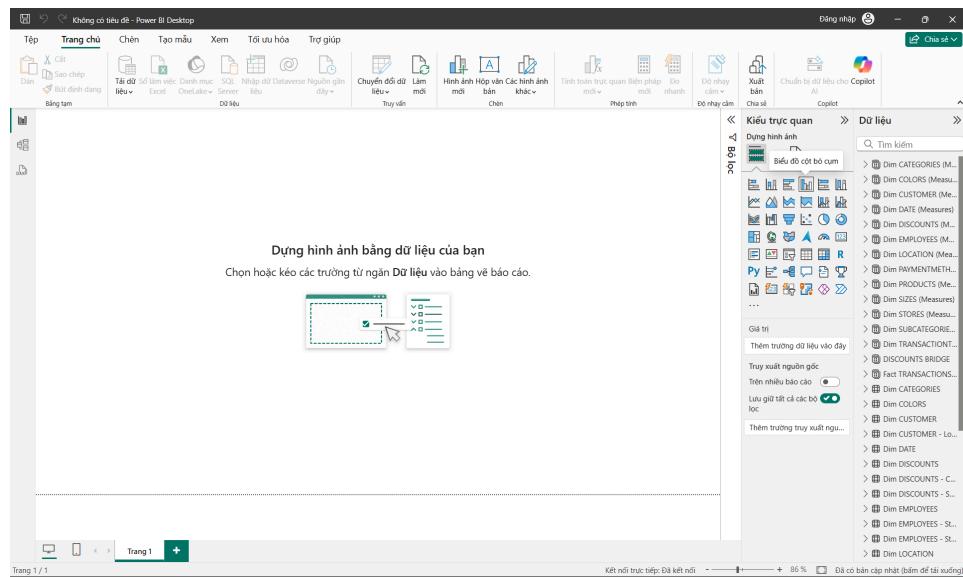


Hình 500. Cửa sổ làm việc sau khi kết nối thành công

### III.2.4.2. Tạo report trong Power BI

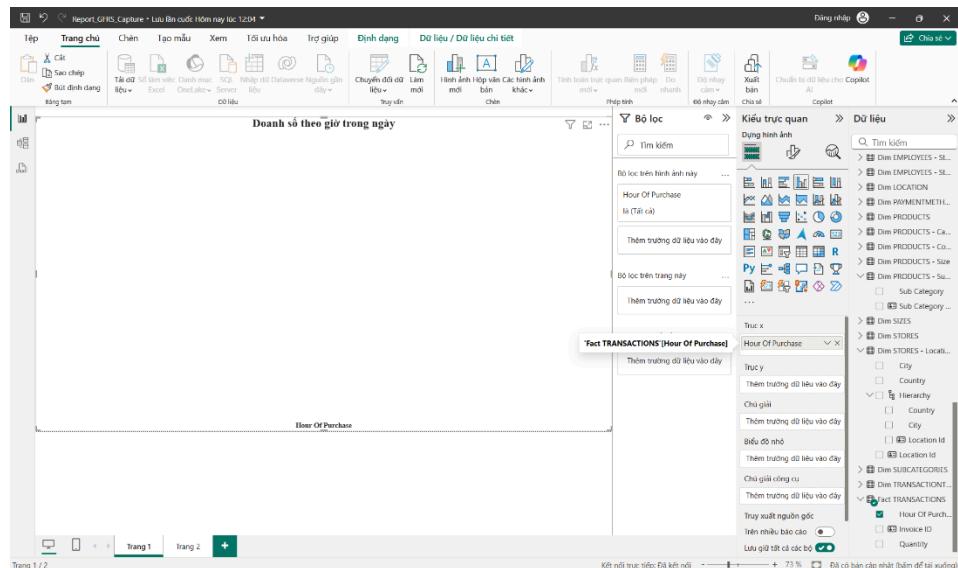
#### III.2.4.2.1. Báo biểu doanh số bán các giờ trong ngày (Group by + Chart)

- **Mục tiêu:** Hiển thị doanh số các giờ trong ngày từ lúc mở cửa đến lúc đóng cửa
- **Ý nghĩa:** nhận diện khung giờ “vàng” (cao điểm) và thấp điểm trong ngày để tối ưu hóa việc sắp xếp nhân sự theo ca kíp và điều phối hoạt động vận hành (như nhập hàng, sơ chế) phù hợp với nhịp độ khách hàng thực tế
- **Thực hiện**
  - o **Bước 1:** Ở tab **Kiểu trực quan**, ta chọn **Biểu đồ cột bó cụm**



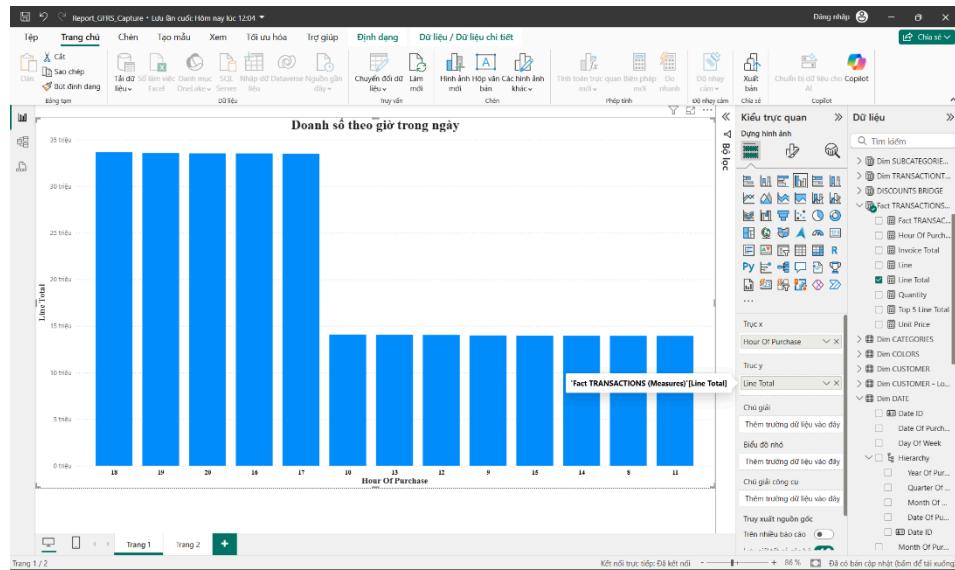
*Hình 501. Chọn biểu đồ cho báo cáo 1*

**o Bước 2:** Tiếp đó ta kéo Hour Of Purchase từ Fact TRANSACTIONS ở tab **Dữ liệu** sang **Trục X** của biểu đồ



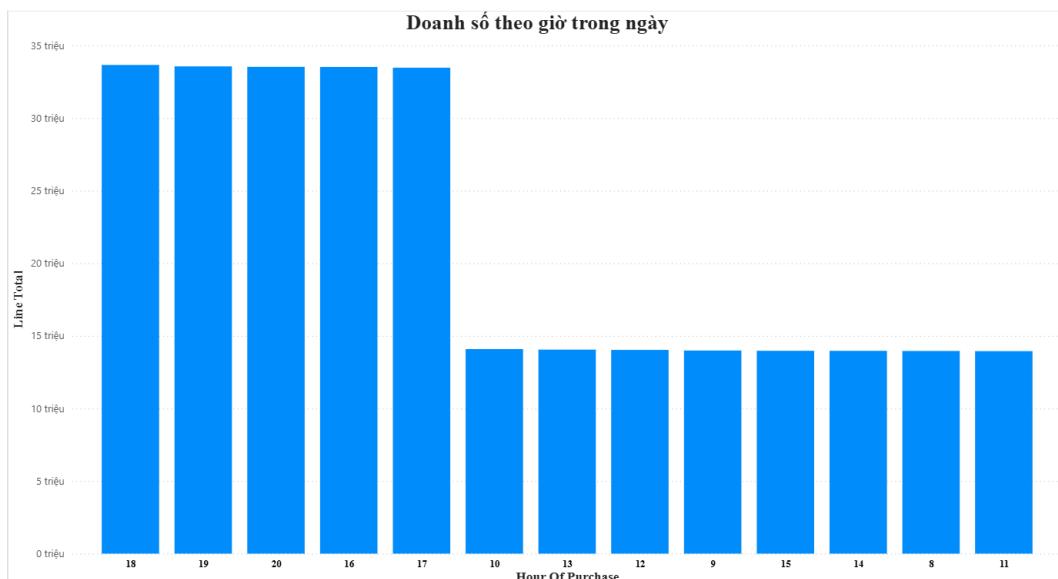
*Hình 502. Chọn trường cho Trục X báo cáo 1*

**o Bước 3:** Sau cùng, ta kéo Measure Line Total từ Fact TRANSACTIONS vào **Trục Y**



Hình 503. Chọn Trường cho Trục Y báo biểu 1

- Kết quả:



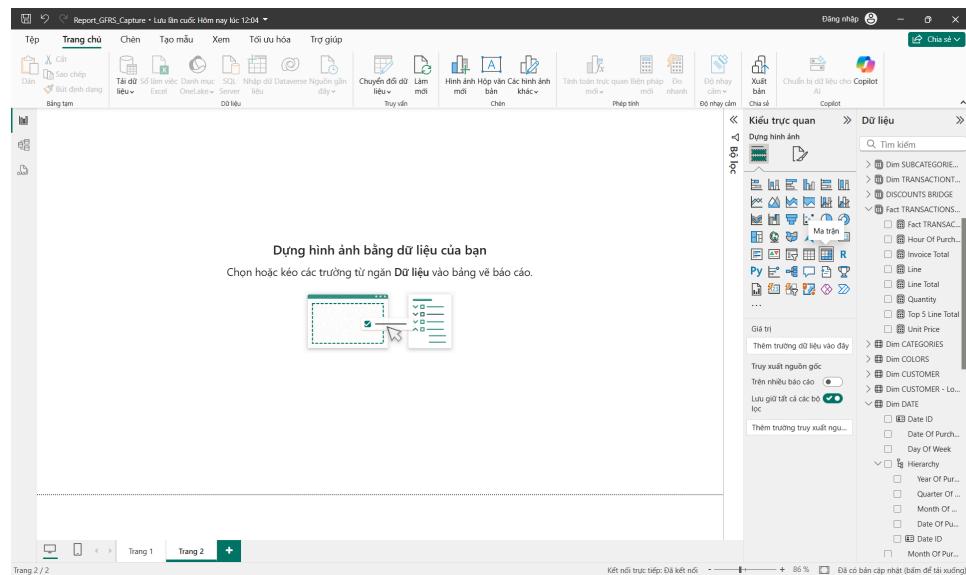
Hình 504. Doanh số theo giờ trong ngày

- **Nhận xét:** Dữ liệu ghi nhận doanh số **tăng vọt** trong khung giờ từ 17h đến 20h. Đây là cơ sở quan trọng để **tái cơ cấu lịch trực**, tập trung nguồn lực tối đa vào khung giờ vàng này nhằm **tối ưu hóa trải nghiệm khách hàng**.

### **III.2.4.2.2. Báo biểu doanh số các danh mục phụ theo từng quý**

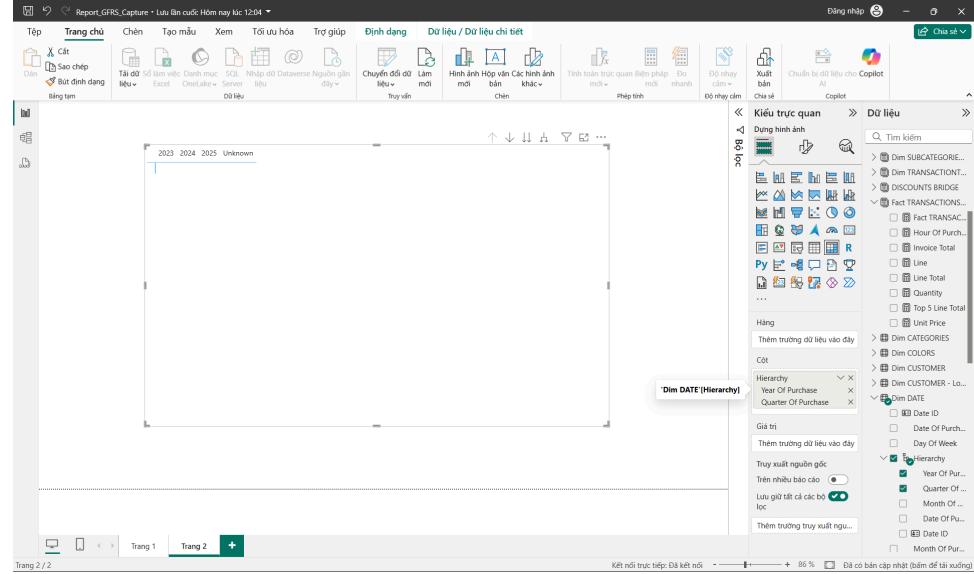
- **Mục tiêu:** Hiển thị doanh số các danh mục phụ theo từng quý từ năm 2023 đến 2025
- **Ý nghĩa:** theo dõi diễn biến kinh doanh của từng nhóm hàng nhỏ qua các mùa trong chu kỳ 3 năm để nhận diện **tính mùa vụ đặc thù** (sản phẩm nào bán chạy vào thời điểm nào). Điều này là cơ sở để **dự báo nhu cầu** và điều chỉnh kế hoạch nhập hàng hoặc tung khuyến mãi “đón đầu” xu hướng theo từng giai đoạn.
- **Thực hiện:**

#### **o Bước 1: Chọn Ma trận trên tab Kiểu trực quan**



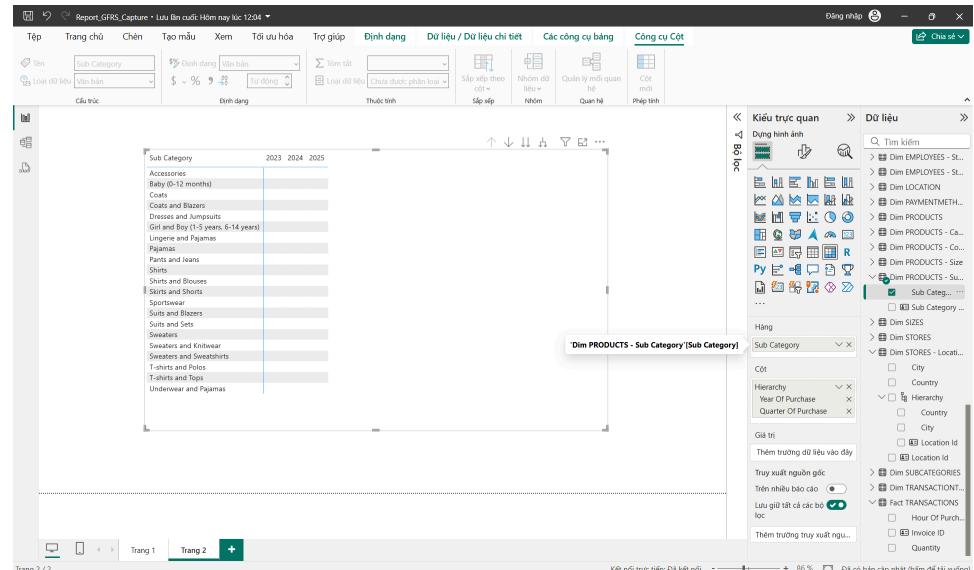
*Hình 505. Chọn kiểu cho báo biểu 2*

#### **o Bước 2: Kéo Hierarchy trong Dim DATE vào Cột, xóa trường Date ID, Date Of Purchase, Month Of Purchase**



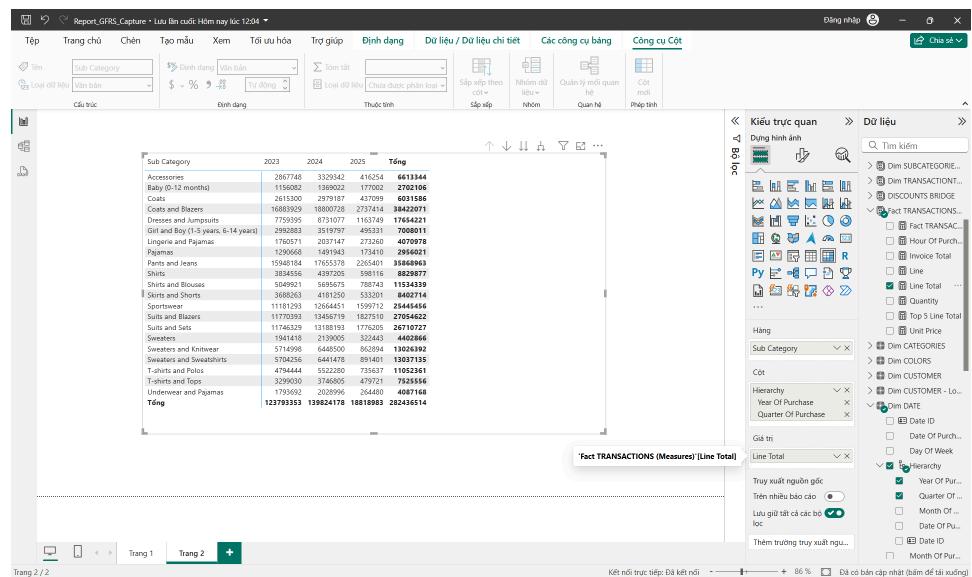
Hình 506. Chọn cột cho báo biểu 2

- o Bước 3: Kéo Sub Category từ Dim PRODUCT – Sub CATEGORY vào Hàng.



Hình 507. Chon hàng cho báo biểu 2

- o Bước 4: Sau cùng, ta kéo Measure Line Total từ Fact TRANSACTIONS vào Giá trị



Hình 508. Chọn giá trị cho báo biểu 2

### - Kết quả:

Year Of Purchase Sub Category	2023				2024				2025				Tổng
	1	2	3	4	1	2	3	4	1	2	3	4	
Accessories	467608	667187	588161	1144792	<b>2867748</b>	538535	745952	698806	1346049	<b>3329342</b>	416254	<b>416254</b>	<b>6613344</b>
Baby (0-12 months)	199833	255025	256965	44259	<b>1156082</b>	237110	288423	307260	536229	<b>1369022</b>	177002	<b>177002</b>	<b>2702106</b>
Coats	485024	478949	838145	813182	<b>2615300</b>	547127	529671	976278	926111	<b>2979187</b>	437099	<b>437099</b>	<b>6031586</b>
Coats and Blazers	3041300	3593252	3662144	6587233	<b>16883929</b>	3521499	3884181	4123656	7271392	<b>18800728</b>	2737414	<b>2737414</b>	<b>38422071</b>
Dresses and Jumpsuits	1605505	1800281	1659599	2694010	<b>7759395</b>	1835392	1967137	1908667	3019881	<b>8731077</b>	1163749	<b>1163749</b>	<b>17654221</b>
Girl and Boy (1-5 years, 6-14 years)	1014821	574063	707045	696954	<b>2992883</b>	1251171	633128	844764	790734	<b>3519797</b>	495331	<b>495331</b>	<b>7008011</b>
Lingerie and Pajamas	300638	399415	386759	673759	<b>1760571</b>	355913	440451	456483	784059	<b>2037147</b>	273260	<b>273260</b>	<b>4070978</b>
Pajamas	204736	392455	250800	443397	<b>1290668</b>	237019	440387	303155	511382	<b>1491943</b>	173410	<b>173410</b>	<b>2956021</b>
Pants and Jeans	2623858	3766958	3702028	5855340	<b>15948184</b>	3009180	4047973	4114579	6483646	<b>17655378</b>	2265401	<b>2265401</b>	<b>35868963</b>
Shirts	801296	777070	912262	1343928	<b>3834556</b>	932502	860459	1071846	1532396	<b>4397205</b>	598116	<b>598116</b>	<b>8829877</b>
Shirts and Blouses	1030488	1043002	1135891	1840540	<b>5049921</b>	1230505	1124284	1267715	2073171	<b>5695675</b>	788743	<b>788743</b>	<b>11534339</b>
Skirts and Shorts	605066	895235	837595	1350367	<b>3688263</b>	697178	964011	966261	1553800	<b>4181250</b>	533201	<b>533201</b>	<b>8402714</b>
Sportswear	1801537	2381219	2340395	4655442	<b>11181293</b>	2144204	2635797	2679452	5204996	<b>12664451</b>	1599712	<b>1599712</b>	<b>25454546</b>
Suits and Blazers	2029466	2593848	2584331	562748	<b>11770393</b>	2368995	2903484	3018071	5166169	<b>13456719</b>	1827510	<b>1827510</b>	<b>25454622</b>
Suits and Sets	1977755	2577856	2595575	4595411	<b>11746329</b>	2326785	2813830	2926061	5121517	<b>13188193</b>	1776205	<b>1776205</b>	<b>26710727</b>
Sweaters	356977	359527	343736	884778	<b>1941418</b>	391003	386257	388196	973549	<b>2139005</b>	322443	<b>322443</b>	<b>4402866</b>
Sweaters and Knitwear	969102	1180263	1172233	2373400	<b>5714998</b>	1127253	1308752	1354666	2657829	<b>6448500</b>	862894	<b>862894</b>	<b>13026392</b>
Sweaters and Sweatshirts	1000501	1172902	1181747	2349106	<b>5704256</b>	1155726	1301114	1368402	2616238	<b>6441478</b>	891401	<b>891401</b>	<b>13037135</b>
T-shirts and Polos	960450	1136865	962537	1732772	<b>4794444</b>	1174438	1308549	1108587	1930706	<b>5522280</b>	735637	<b>735637</b>	<b>11052361</b>
T-shirts and Tops	549883	854129	709454	1185564	<b>3299030</b>	612934	870152	811373	1452346	<b>3746805</b>	479721	<b>479721</b>	<b>7525556</b>
Underwear and Pajamas	311354	398094	396546	687696	<b>1793692</b>	356303	441936	453234	775725	<b>2028996</b>	264480	<b>264480</b>	<b>4087168</b>
<b>Tổng</b>	<b>22357198</b>	<b>27295815</b>	<b>27225662</b>	<b>46914678</b>	<b>123793353</b>	<b>26050773</b>	<b>29895928</b>	<b>311747512</b>	<b>52729963</b>	<b>139824178</b>	<b>18818983</b>	<b>18818983</b>	<b>282436514</b>

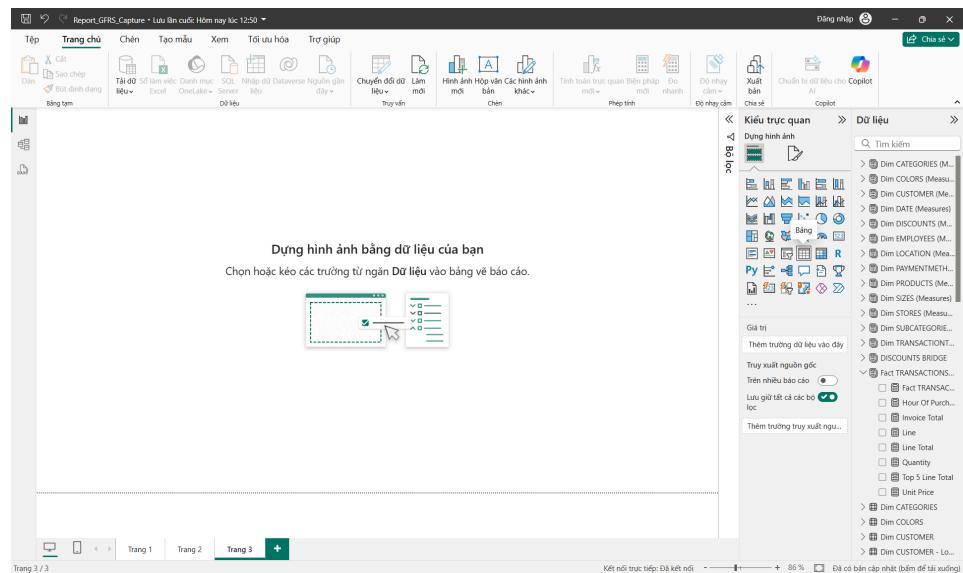
Hình 509. Doanh số theo các quý từ năm 2023 đến 2025

- Nhận xét:** Hoạt động kinh doanh mang tính **chu kỳ mùa vụ đậm nét**. Biểu đồ Doanh số có hình thái **đốc lèn**, khởi đầu thấp vào đầu năm và **đạt đỉnh cực đại vào Quý 4** (gấp 2 lần Quý 1). Điều này khẳng định Quý 4 là ‘thời điểm vàng’, quyết định 40-50% thành bại của cả năm tài chính, đòi hỏi sự tập trung nguồn lực tuyệt đối về hàng hóa và nhân sự cho giai đoạn này.

### **III.2.4.2.3. Báo biểu về Doanh số theo Danh mục con (Sub Category) và Nhóm khách hàng (Category) (Table)**

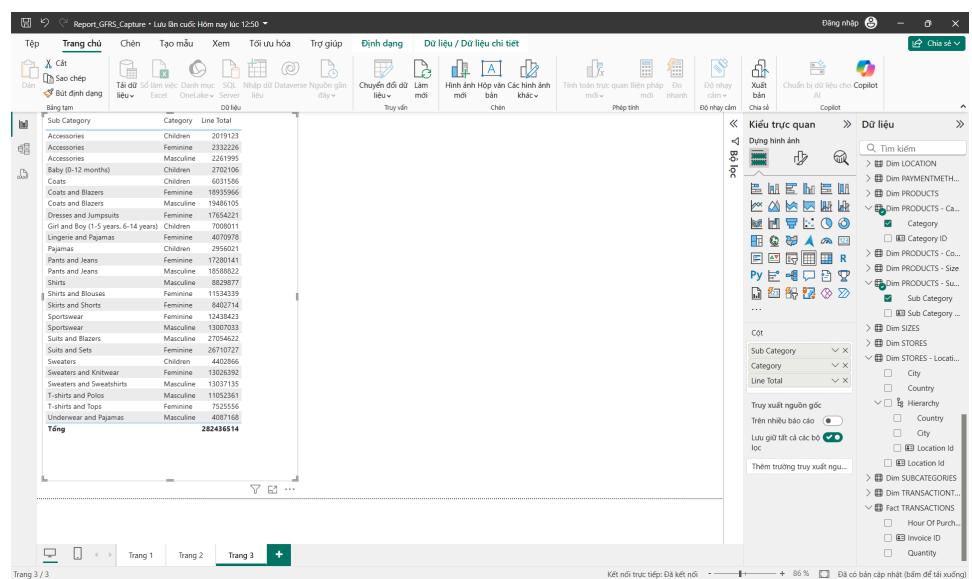
- **Mục tiêu:** Hiển thị doanh số theo từng danh mục con và nhóm khách hàng cụ thể.
- **Ý nghĩa:** xác định **thị hiếu tiêu dùng** riêng biệt của từng phân khúc khách hàng đối với các dòng sản phẩm. Điều này là cơ sở quan trọng để xây dựng chiến lược **cá nhân hóa** (như bán chéo/cross-selling sản phẩm phù hợp) nhằm tối đa hóa giá trị khai thác trên từng tập khách hàng.
- **Thực hiện**

#### **o Bước 1: Chọn Bảng ở phần Kiểu trực quan**



Hình 510. Chọn kiểu cho báo biểu 3

#### **o Bước 2: Chọn các trường Measure Line Total từ Fact TRANSACTION, Category từ Dim PRODUCTS – Category, Sub Category từ Dim PRODUCTS – Sub Category**



Hình 511. Chọn các trường cho báo biểu 3

- Kết quả:

Sub Category	Category	Line Total
Accessories	Children	2019123
Accessories	Feminine	2332226
Accessories	Masculine	2261995
Baby (0-12 months)	Children	2702106
Coats	Children	6031586
Coats and Blazers	Feminine	18935966
Coats and Blazers	Masculine	19486105
Dresses and Jumpsuits	Feminine	17654221
Girl and Boy (1-5 years, 6-14 years)	Children	7008011
Lingerie and Pajamas	Feminine	4070978
Pajamas	Children	2956021
Pants and Jeans	Feminine	17280141
Pants and Jeans	Masculine	18588822
Shirts	Masculine	8829877
Shirts and Blouses	Feminine	11534339
Skirts and Shorts	Feminine	8402714
Sportswear	Feminine	12438423
Sportswear	Masculine	13007033
Suits and Blazers	Masculine	27054622
Suits and Sets	Feminine	26710727
Sweaters	Children	4402866
Sweaters and Knitwear	Feminine	13026392
Sweaters and Sweatshirts	Masculine	13037135
T-shirts and Polos	Masculine	11052361
T-shirts and Tops	Feminine	7525556
Underwear and Pajamas	Masculine	4087168
<b>Tổng</b>		<b>282436514</b>

Hình 512. Doanh số theo doanh mục con và nhóm khách hàng

- Nhận xét:

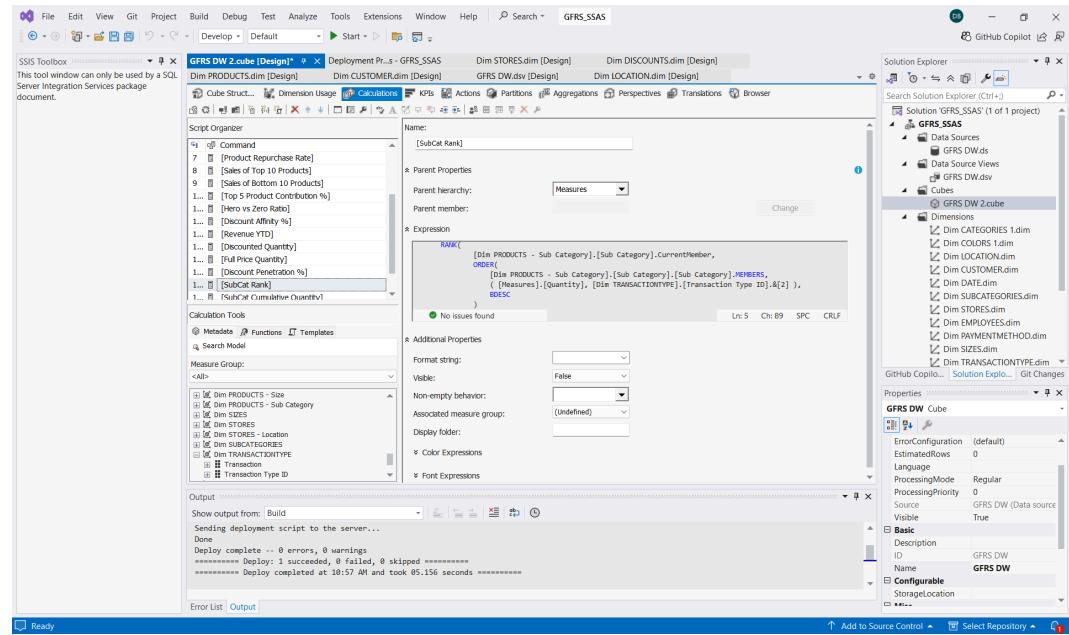
- o **Về cơ cấu sản phẩm:** Số liệu cho thấy dòng sản phẩm **Suit, Blazer và đồ âu (Suits & Sets)** đang là động lực tăng trưởng chính, chiếm tỷ trọng Doanh số cao nhất ở cả hai nhóm khách hàng Nam và Nữ.

Điều này khẳng định vị thế của thương hiệu trong phân khúc **thời trang công sở và sự kiện**.

- o **Về đối tượng khách hàng:** Doanh nghiệp duy trì được cơ cấu Doanh số **cân bằng lý tưởng** giữa nhóm khách hàng Nam (Masculine) và Nữ (Feminine). Sự chênh lệch doanh số giữa hai nhóm này ở các hạng mục chính (Quần, Áo khoác, Đồ thể thao) là không đáng kể, cho thấy độ phủ thương hiệu tốt đều cho cả hai giới. Ngược lại, mảng Trẻ em (Children) vẫn còn nhiều dư địa nhưng hiện tại đóng góp chưa tương xứng với tiềm năng

#### ***III.2.4.2.4. Báo biểu về Pareto cho Sub Category.***

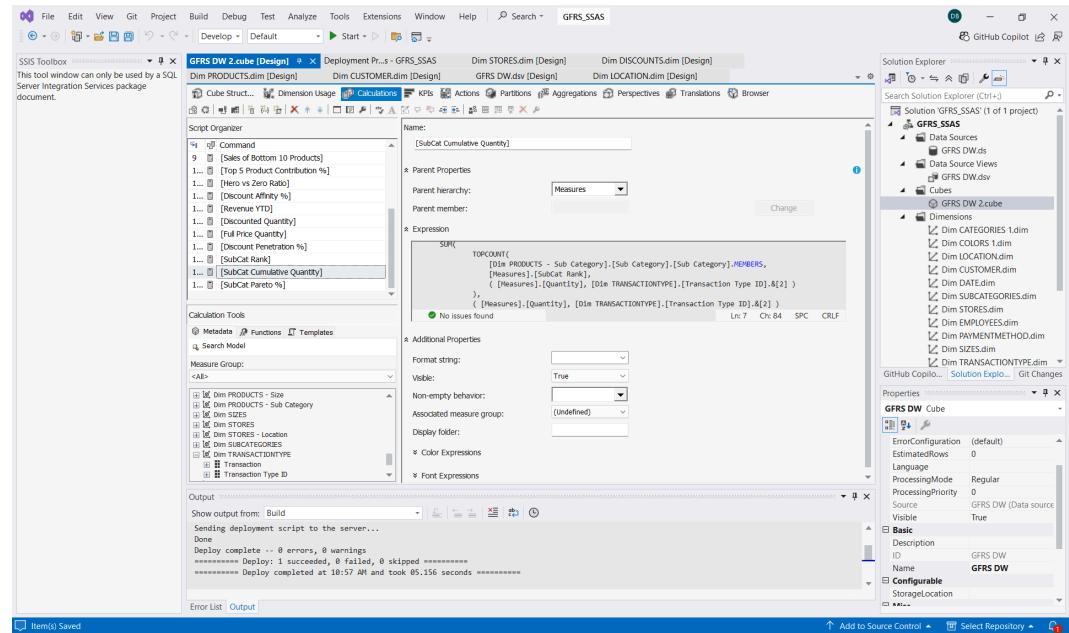
- **Mục tiêu:** Giải thích cho câu hỏi “Có phải 20% mã sản phẩm đang gánh 80% doanh số của cửa hàng không?”
- **Ý nghĩa:** áp dụng nguyên lý 80/20 để phân loại và nhận diện nhóm **sản phẩm cốt lõi** đóng góp phần lớn vào kết quả kinh doanh, giúp tối ưu hóa nguồn lực quản lý kho và ưu tiên vốn cho các mặt hàng bán chạy. Đồng thời, phân tích này giúp đánh giá **rủi ro phụ thuộc** nguồn thu, xác định xem doanh nghiệp có đang “bỏ trung vào quá ít giờ” hay không.
- **Chuẩn bị:** Để thực hiện được báo biểu này ta cần tạo **Measure SubCat Rank** để xếp hạng các Sub Category theo số lượng bán được, **Measure SubCat Cumulative Quantity** để cộng dồn số lượng từ SubCategory hạng 1 đến hạng hiện tại, và **Measure SubCat Pareto %** để tổng tất cả Sub Category. Vào tab **Calculations** của cube, ở ô **Script Organizer**, nhấn chuột phải chọn **New Calculated Member** và thực hiện thêm 3 Measure sau:



Hình 513. Tạo SubCat Rank mới

Bảng 51. Expression SubCat Rank

<pre>RANK(     [Dim PRODUCTS - Sub Category].[Sub Category].CurrentMember,     ORDER(         [Dim PRODUCTS - Sub Category].[Sub Category].[Sub Category].MEMBERS,         ( [Measures].[Quantity], [Dim             TRANSACTIONTYPE].[Transaction Type ID].&amp;[2] ),         BDESC     ) )</pre>
---



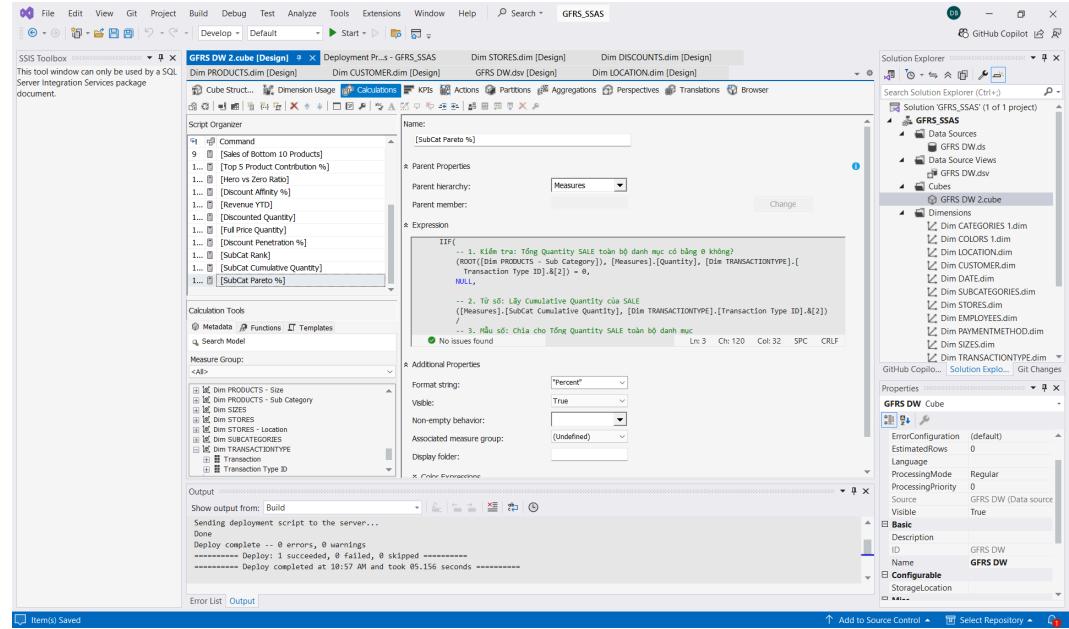
Hình 514. Tạo SubCat Cumulative Quantity mới

Bảng 52. Expression SubCat Cumulative Quantity

```

SUM(
    TOPCOUNT(
        [Dim PRODUCTS - Sub Category].[Sub Category].[Sub
Category].MEMBERS,
        [Measures].[SubCat Rank],
        ( [Measures].[Quantity], [Dim
TRANSACTIONTYPE].[Transaction Type ID].&[2] )
    ),
    ( [Measures].[Quantity], [Dim TRANSACTIONTYPE].[Transaction
Type ID].&[2] )
)

```



Hình 515. Tạo SubCat Pareto % mới

Bảng 53. Expression SubCat Pareto %

IIF(

-- 1. Kiểm tra: Tổng Quantity SALE toàn bộ danh mục có bằng 0 không?

(ROOT([Dim PRODUCTS - Sub Category]), [Measures].[Quantity],  
[Dim TRANSACTIONTYPE].[Transaction Type ID].&[2]) = 0,

NULL,

-- 2. Tỷ số: Lấy Cumulative Quantity của SALE

([Measures].[SubCat Cumulative Quantity], [Dim  
TRANSACTIONTYPE].[Transaction Type ID].&[2])

/

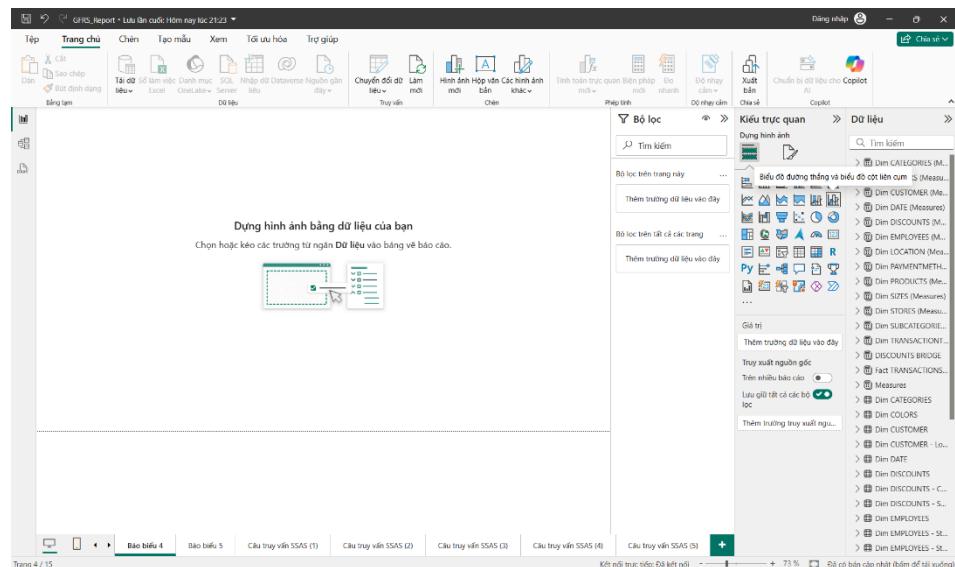
-- 3. Mẫu số: Chia cho Tổng Quantity SALE toàn bộ danh mục

(ROOT([Dim PRODUCTS - Sub Category]), [Measures].[Quantity],  
[Dim TRANSACTIONTYPE].[Transaction Type ID].&[2])

)

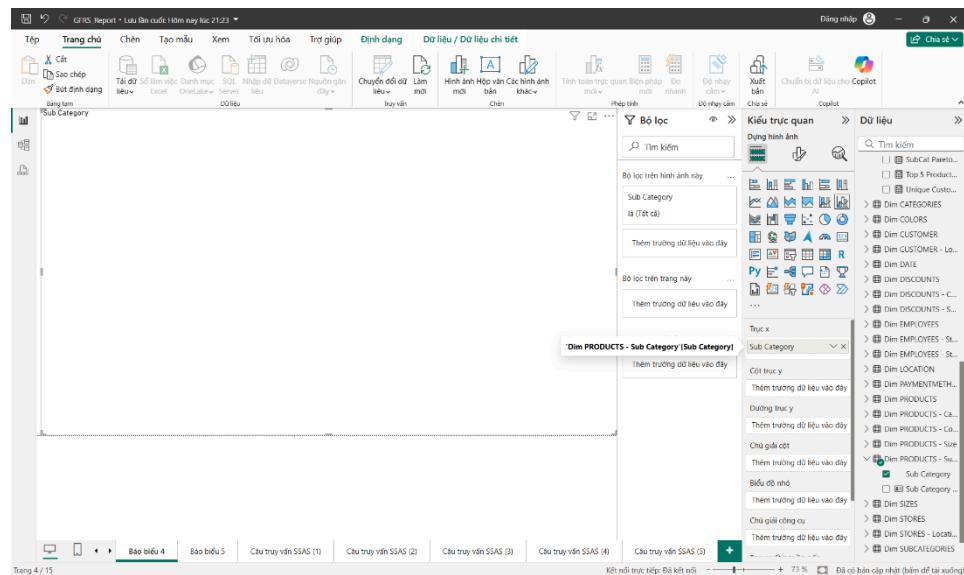
- **Thực hiện:**

- o **Bước 1: Chọn Biểu đồ đường thẳng và biểu đồ cột liên cụm từ Kiểu trực quan**



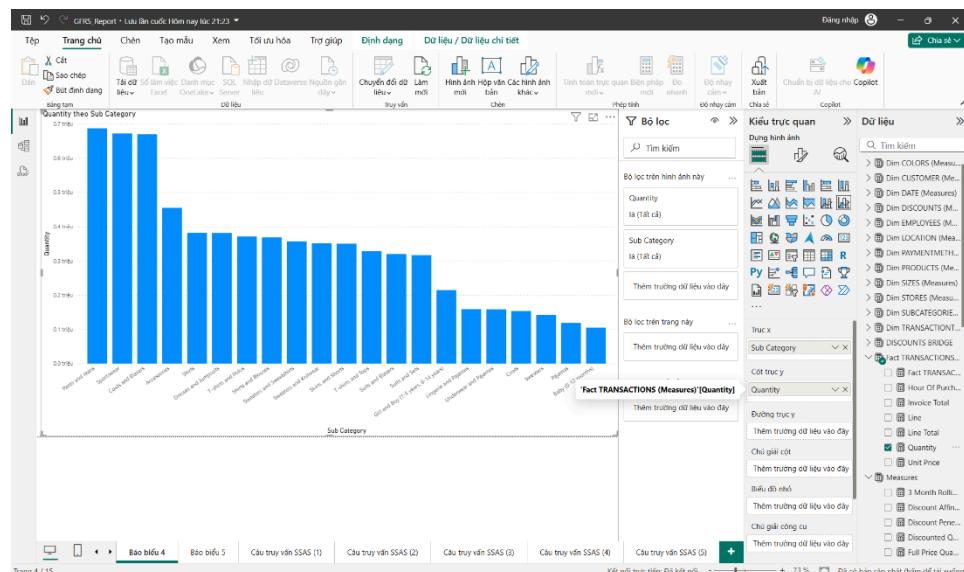
Hình 516. Chọn kiểu cho báo biểu 4

- o **Bước 2: Kéo Sub Category từ Dim PRODUCTS – Sub Category vào Trục X**



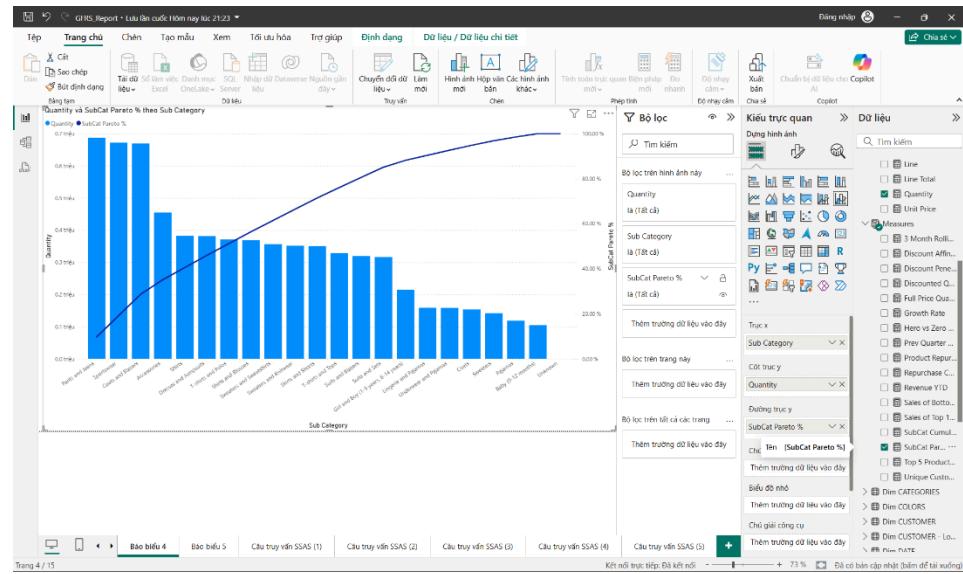
Hình 517. Chọn trường cho Trục X báo biểu 4

### o Bước 3: Kéo Measure Quantity từ Fact TRANSACTION vào Trục Y



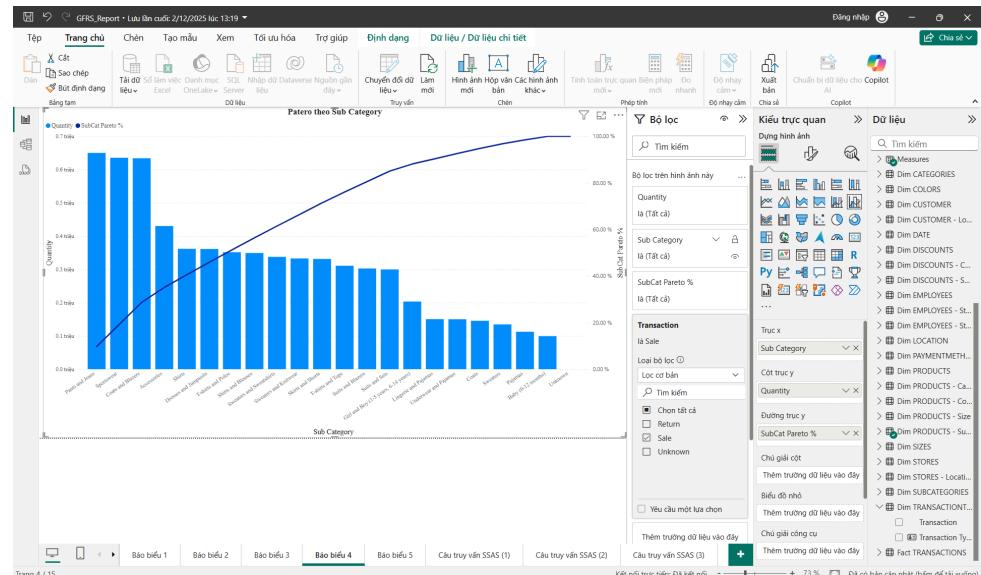
Hình 518. Chọn trường cho Trục Y báo biểu 4

### o Bước 4: Kéo Measure SubCat Pareto % vào Đường trực Y



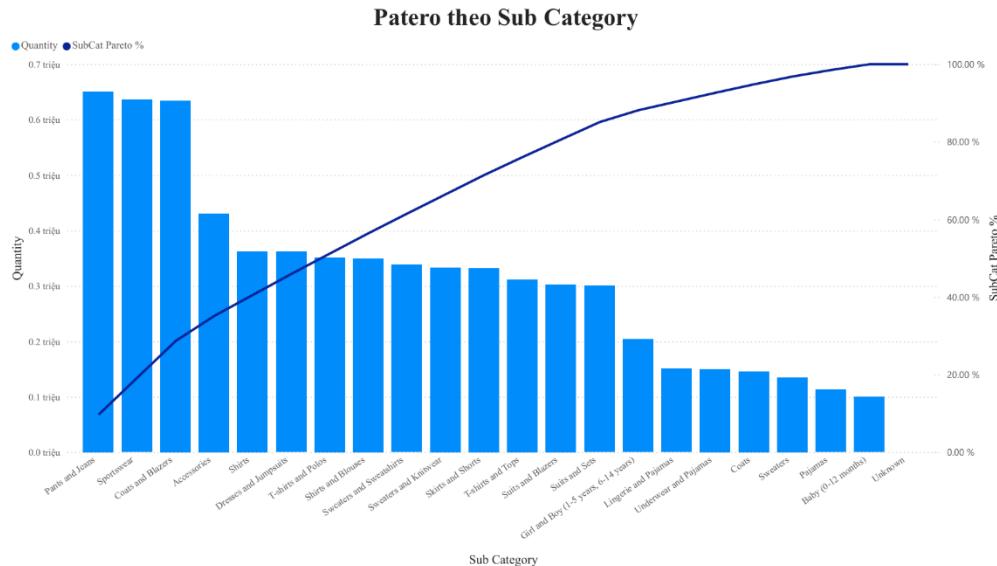
Hình 519. Chọn trường cho Đường trực Y báo biểu 4

## o Bước 5: Kéo Transaction từ Dim TRANSACTIONTYPE vào bộ lọc và chọn Sale



Hình 520. Cài đặt bộ lock cho báo biểu 4

- Kết quả:



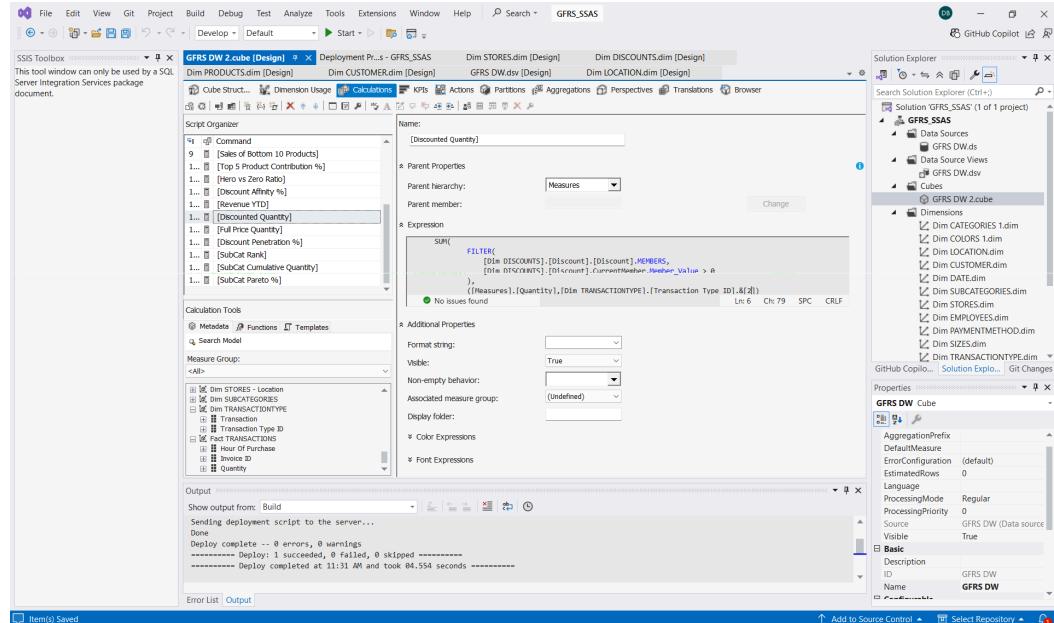
Hình 521. Doanh số các số Sub Category và đường Pareto

- **Nhận xét:** Cấu trúc doanh số khá bền vững, doanh thu phân bố đều. Cần gom khoảng 12-14 nhóm hàng mới đạt 80% tổng lượng (ít rủi ro phụ thuộc nhưng quản lý kho cực hơn). Các loại sản phẩm có thể hỗ trợ giúp đỡ doanh số bền vững

#### **III.2.4.2.5. Báo biểu Phân tích Độ nhạy cảm về Giá & Hiệu quả Khuyến mãi**

- **Mục tiêu:** Trả lời câu hỏi: “Doanh số đến từ khách hàng mua đúng giá hay đến từ khách hàng chỉ săn khuyến mãi?”
- **Ý nghĩa:** đánh giá **chất lượng doanh thu** và sức mạnh thực sự của thương hiệu bằng cách đo lường mức độ phụ thuộc vào các chương trình kích cầu. Phân tích này giúp nhận diện xem doanh nghiệp đang tăng trưởng bền vững hay đang bị **xói mòn biên lợi nhuận** do phải liên tục giảm giá để duy trì doanh số.
- **Chuẩn bị:** Để thực hiện được báo biểu này ta cần tạo **Measure Discounted Quantity** để lấy số lượng bán hàng có giảm giá, **Measure Full Price Quantity** để lấy số lượng bán hàng đúng giá, và **Measure Discount**

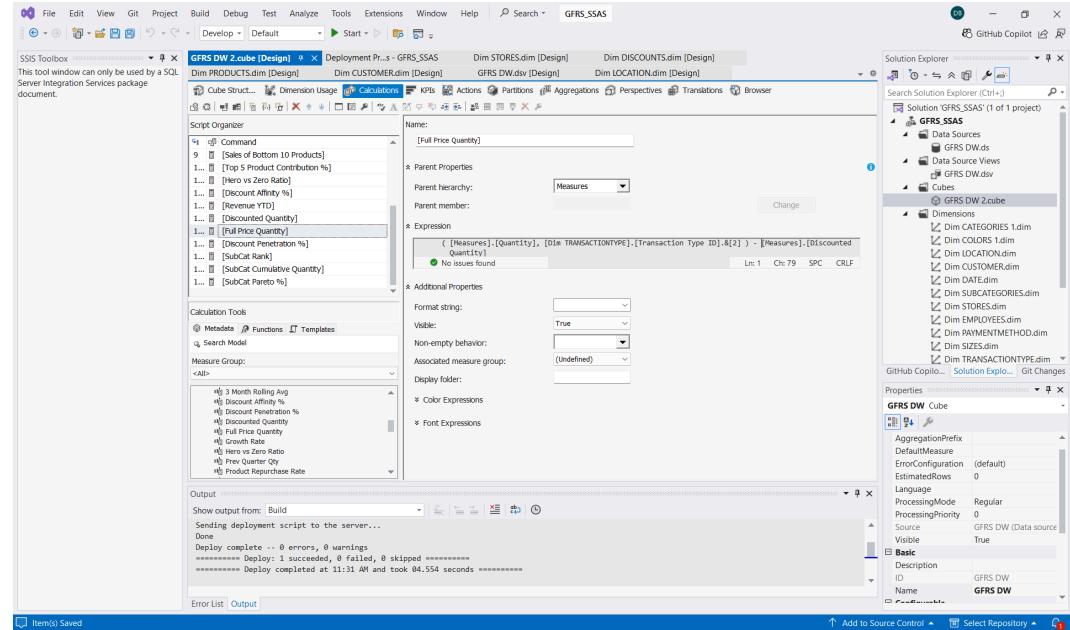
**Penetration %** để tính tỷ lệ bán hàng có giảm giá. Vào tab **Calculations** của cube, ở ô **Script Organizer**, nhấn chuột phải chọn **New Calculated Member** và thực hiện thêm 3 Measure sau:



Hình 522. Tạo Measure Discounted Quantity

Bảng 54. Expression Measure Discounted Quantity

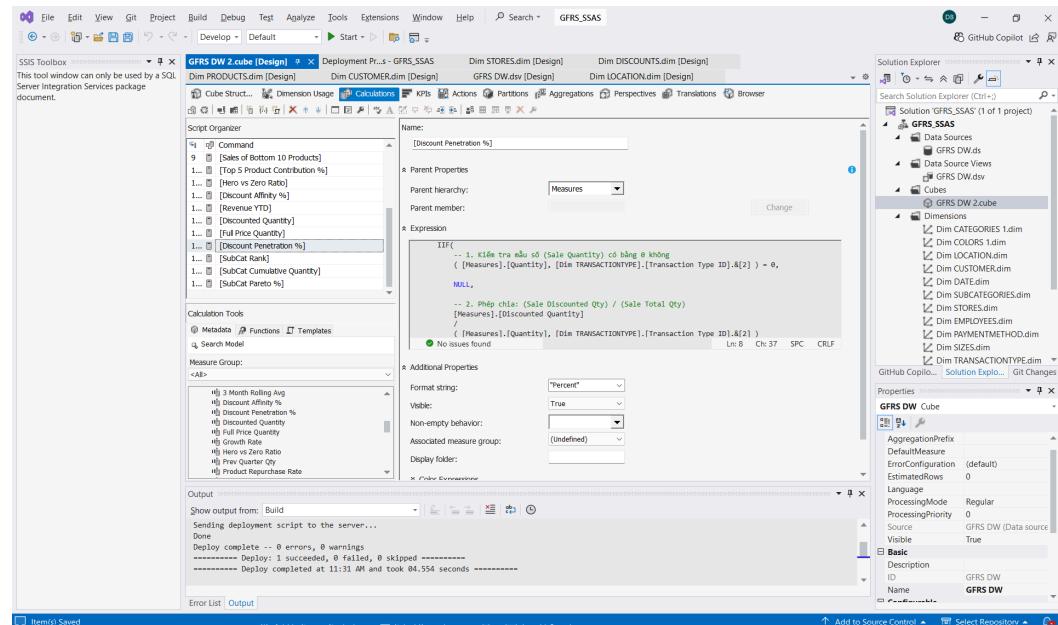
$\text{SUM}($  $\text{FILTER}($  $[Dim DISCOUNTS].[Discount].[Discount].MEMBERS,$  $[Dim DISCOUNTS].[Discount].CurrentMember.Member\_Value$ $> 0$  $),$  $([Measures].[Quantity],[Dim TRANSACTIONTYPE].[Transaction$ $Type ID].&[2])$
---



Hình 523. Tạo Measure Full Price Quantity

Bảng 55. Expression Measure Full Price Quantity

( [Measures].[Quantity], [Dim TRANSACTIONTYPE].[Transaction Type ID].&[2] ) - [Measures].[Discounted Quantity]
--



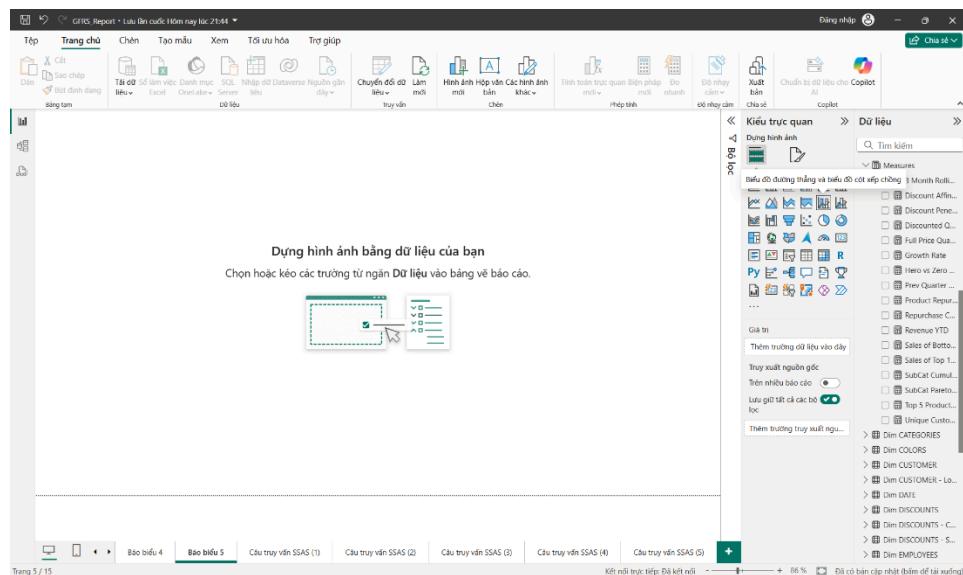
Hình 524. Tạo Measure Discount Penetration %

Bảng 56. Expression Measure Discount Penetration %

$\text{IIF}(\text{-- 1. Kiểm tra mẫu số (Sale Quantity) có bằng 0 không}$ $(\text{[Measures].[Quantity]}, \text{[Dim TRANSACTIONTYPE].[Transaction Type ID].&[2]}) = 0,$ $\text{NULL},$  $\text{-- 2. Phép chia: (Sale Discounted Qty) / (Sale Total Qty)}$ $[\text{Measures}].[\text{Discounted Quantity}]$ $/$ $([\text{Measures}].[\text{Quantity}], \text{[Dim TRANSACTIONTYPE].[Transaction Type ID].&[2]})$
---

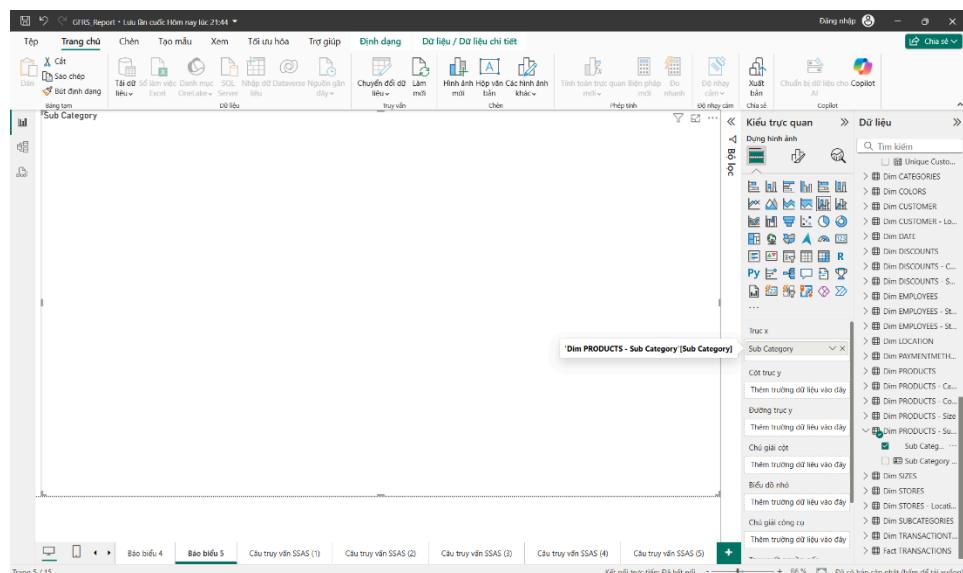
- **Thực hiện:**

- o **Bước 1: Chọn Biểu đồ đường thẳng và biểu đồ cột xếp chồng từ Kiểu trực quan**



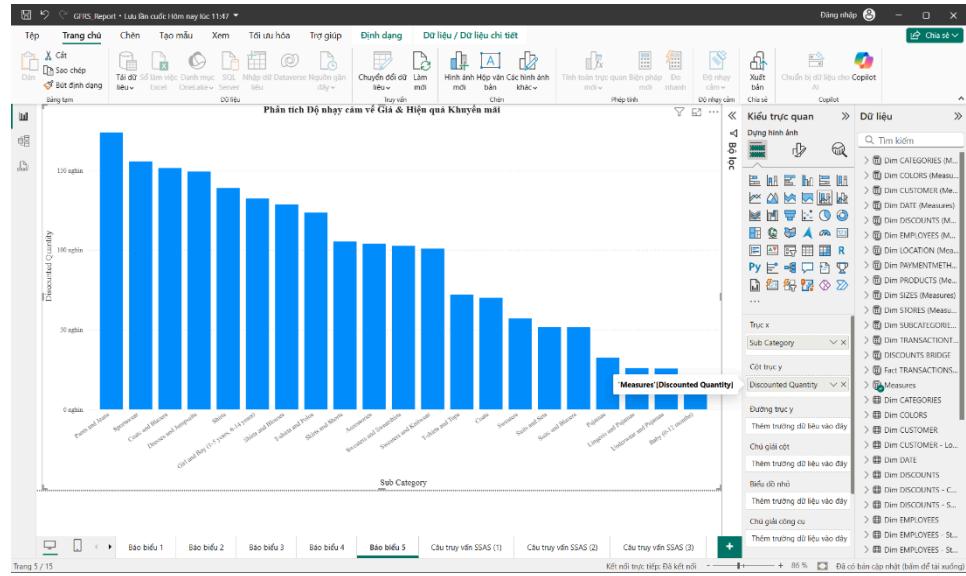
Hình 525. Chọn kiểu cho báo biểu 5

- o **Bước 2: Kéo Sub Category từ Dim PRODUCTS – Sub Category sang Trục x.**



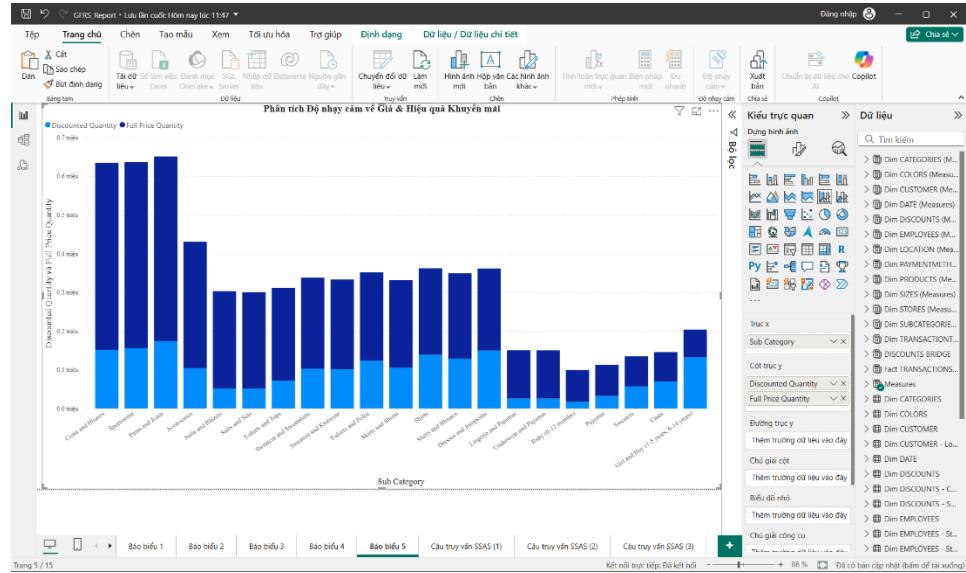
Hình 526. Chọn trường cho Trục x báo biểu 5

### o Bước 3: Kéo Discounted Quantity từ Measures sang Cột trục y



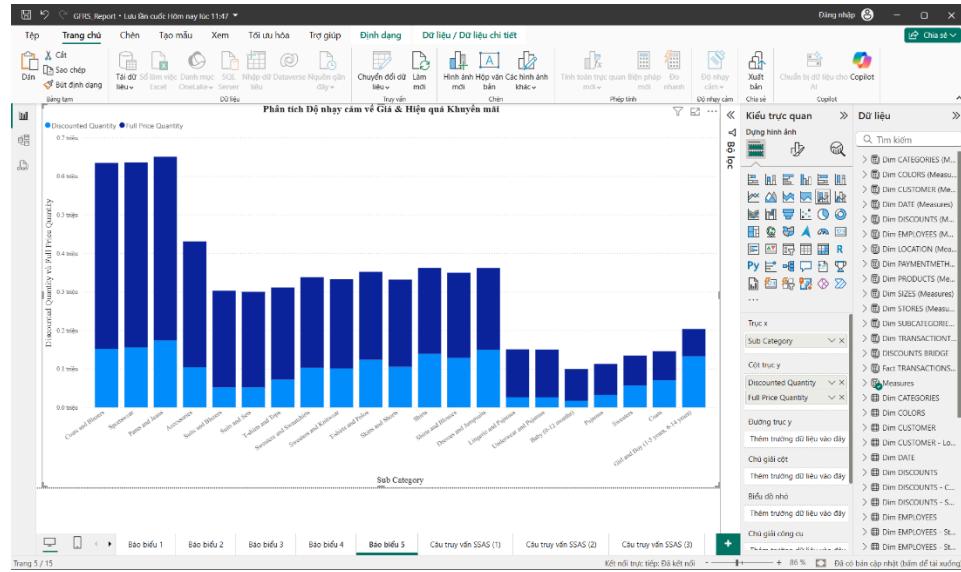
Hình 527. Thêm trường thứ 1 vào Cột trục y báo biểu 5

### o Bước 4: Kéo Full Price Quantity từ Measures trong Dim PRODUCTS – Sub Category sang Cột trục y.



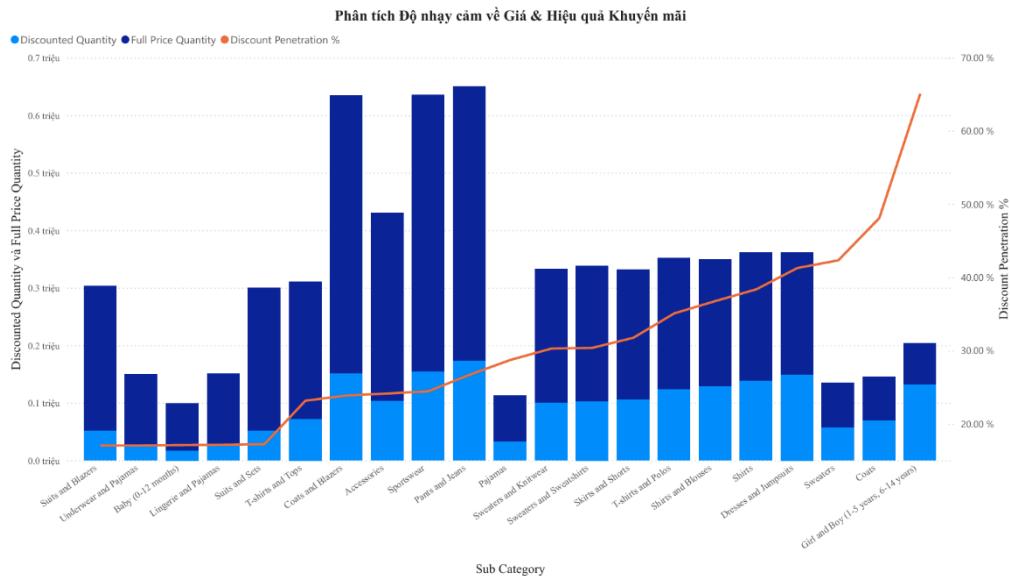
Hình 528. Thêm trường thứ 2 vào Cột trục y báo biểu 5

### o Bước 5: Kéo Measure Penetration % từ Measures sang Đường trục y



Hình 529. Chọn trường cho Đường trực y báo biểu 5

### - Kết quả:



Hình 530. Phân tích Độ nhạy cảm về Giá & Hiệu quả Khuyến mãi

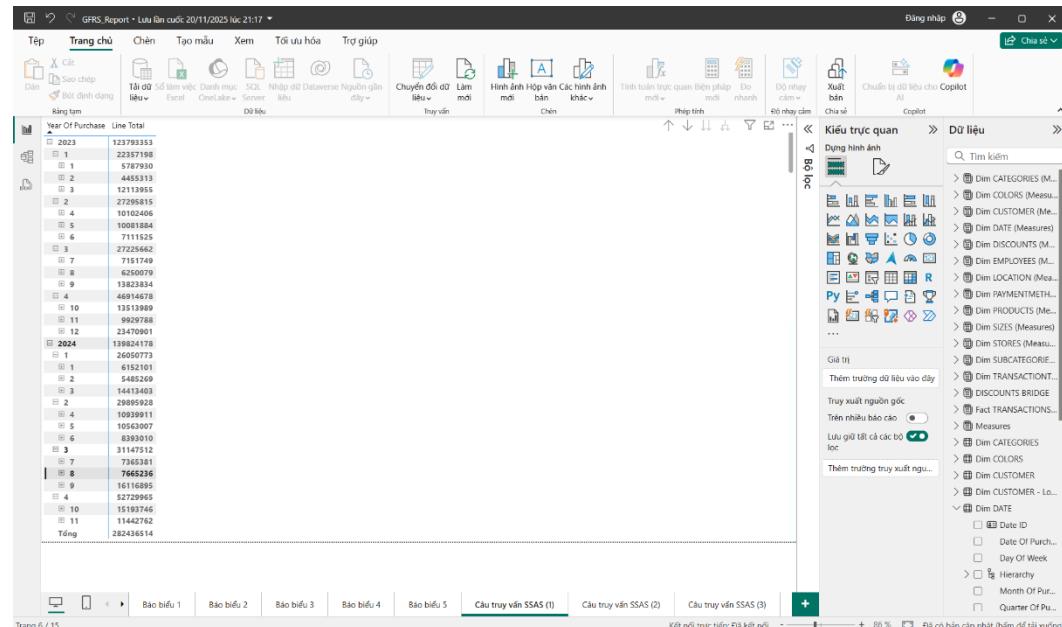
- **Nhận xét:** dữ liệu phân hóa thành 4 nhóm, nhóm an toàn (Suits And Blazers, Underwear and Pajamas,...) có tỷ lệ mua có giảm giá thấp, chứng tỏ không phụ thuộc nhiều; nhóm cần quan sát (Coats and Blazers, Accessories,...), nhóm này có doanh số cao, tỷ lệ mua có giảm giá cũng

khá cao, tuy nhiên do doanh số cao, nên khả năng khách hàng mua nhiều hơn khi có khuyến mãi là có, cần quan sát nhiều hơn; nhóm nguy hiểm (Sweater and Knitwear, Sweater and Sweatshirt,..) nhóm này có doanh số khá cao, tỷ lệ mua có giảm giá cao hơn mức trung bình, điều này chứng tỏ nhóm này phụ thuộc nhiều vào giảm giá; nhóm rất nguy hiểm (Sweaters, Coats,..), nhóm này có tỷ lệ mua khi có khuyến mãi cao nhất, điều này có nghĩa các sản phẩm này chỉ có thể bán được khi có khuyến mãi.

### III.2.4.3. Thực hiện lại các câu truy vấn SSAS trong Power BI

#### III.2.4.3.1. Báo biểu Doanh số roll up theo năm

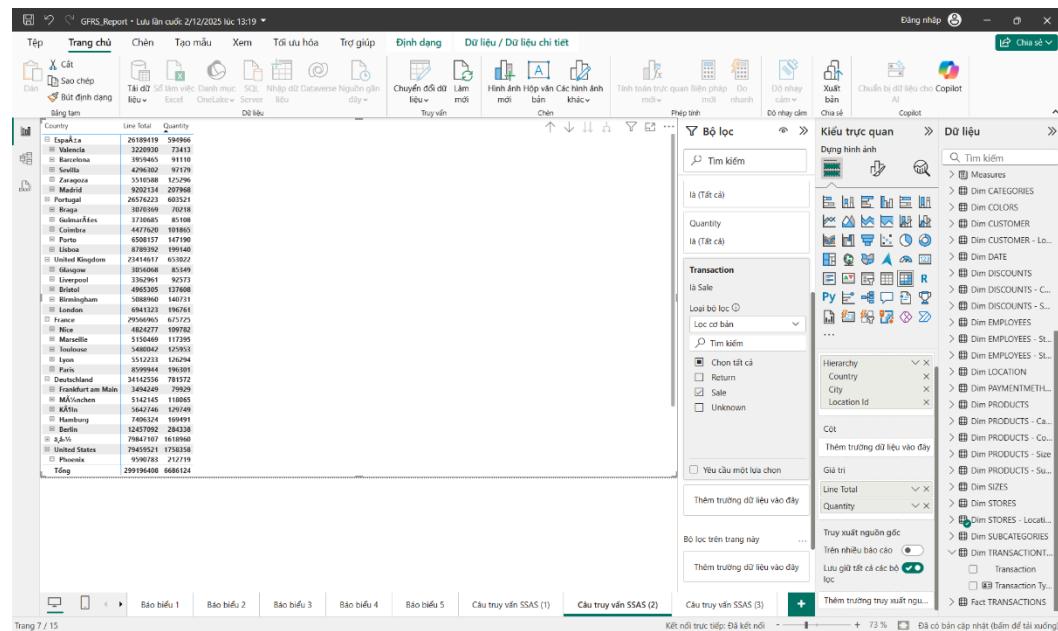
- **Thực hiện:** Chọn Ma trận để minh họa câu truy vấn này, kéo Hierarchy từ Dim DATE vào Trục X, sau đó kéo Measure Line Total từ Fact TRANSACTIONS vào Trục Y.
- **Kết quả:**



Hình 531. Câu truy vấn phản SSAS trong Power BI 1

### III.2.4.3.2. Báo biểu Doanh số và số lượng sản phẩm roll up theo quốc gia

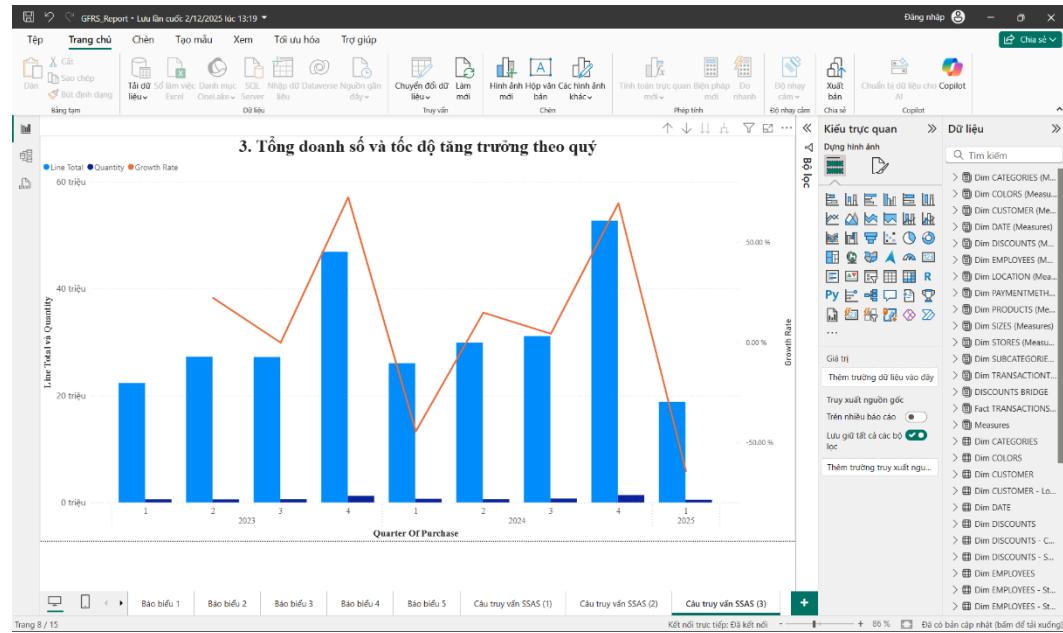
- **Thực hiện:** Chọn Ma trận để minh họa truy vấn này, kéo Hierarchy từ Dim STORES – Location vào Trục X, sau đó kéo Measure Line Total và Quantity từ Fact TRANSACTIONS vào Trục Y. Sau cùng, kéo Transaction từ Dim TRANSACTIONTYPE vào bộ lọc và chỉnh thành Sale.
- **Kết quả:**



Hình 532. Câu truy vấn phần SSAS trong Power BI 2

### III.2.4.3.3. Báo biểu Doanh số và tốc độ tăng trưởng theo từng quý

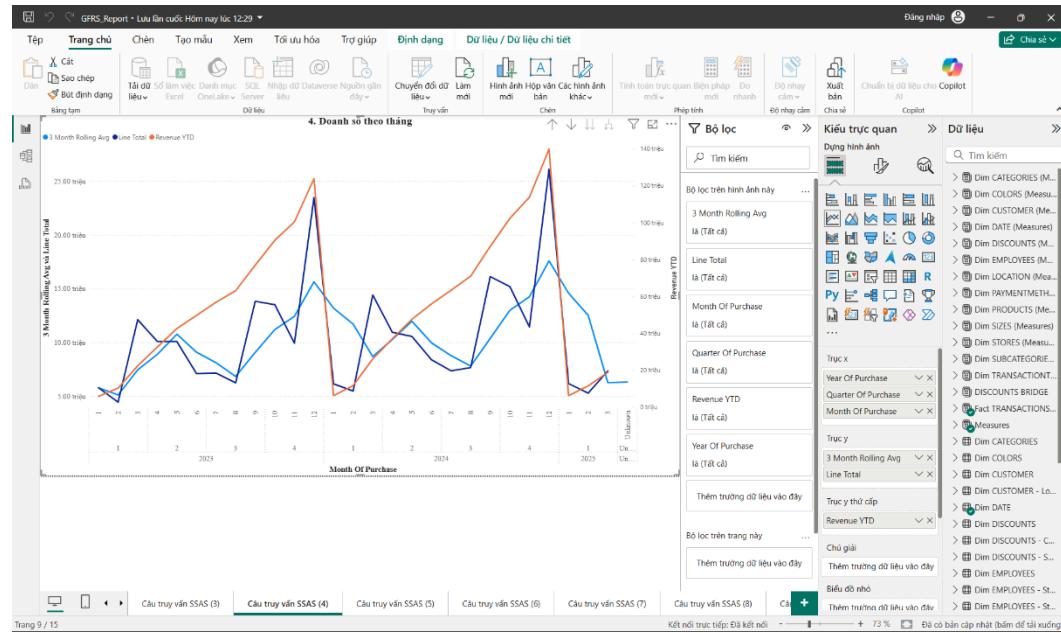
- **Thực hiện:** Chọn Biểu đồ đường thẳng và biểu đồ cột liên cụm để minh họa truy vấn này, kéo Quarter Of Purchase và Year Of Purchase từ Dim DATE vào Trục X, sau đó kéo Measure Line Total và Quantity từ Fact TRANSACTIONS vào Trục Y, kéo Growth Rate vào Đường trực y
- **Kết quả:**



Hình 533. Câu truy vấn phản SSAS trong Power BI 3

#### **III.2.4.3.4. Báo biểu doanh số theo tháng**

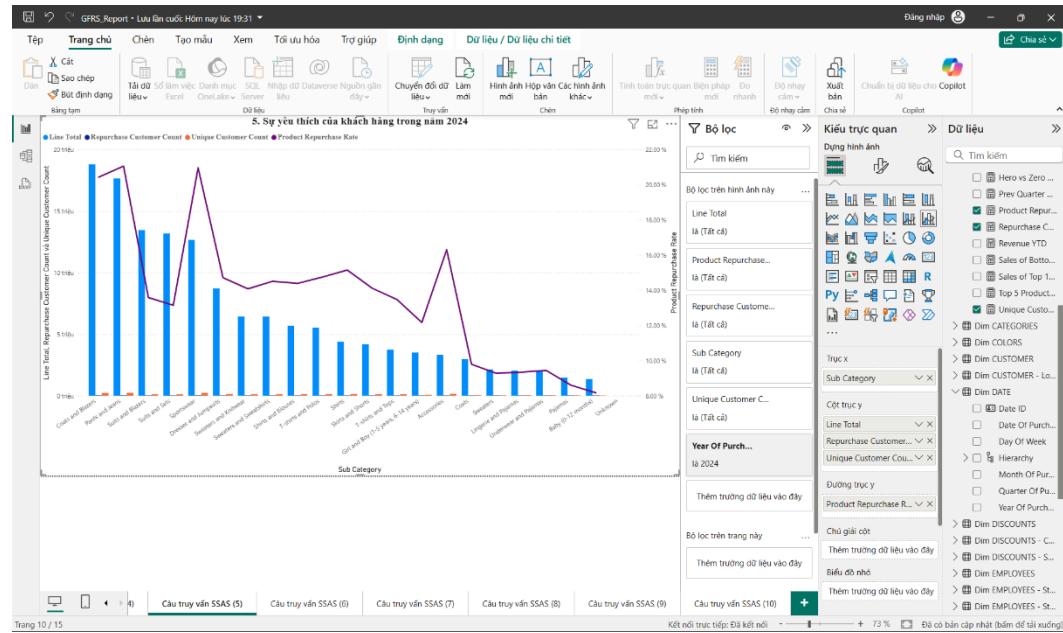
- **Thực hiện:** Chọn **Biểu đồ đường** để minh họa câu truy vấn này, kéo **Year Of Purchase**, **Quarter Of Purchase**, **Month Of Purchase** từ **Dim DATE** vào **Trục X**, sau đó kéo **Measure Line Total** từ **Fact TRANSACTIONS** vào **Trục Y**.
- **Kết quả:**



Hình 534. Câu truy vấn phản SSAS trong Power BI 4

### III.2.4.3.5. Báo biểu sự yêu thích của khách hàng năm 2024

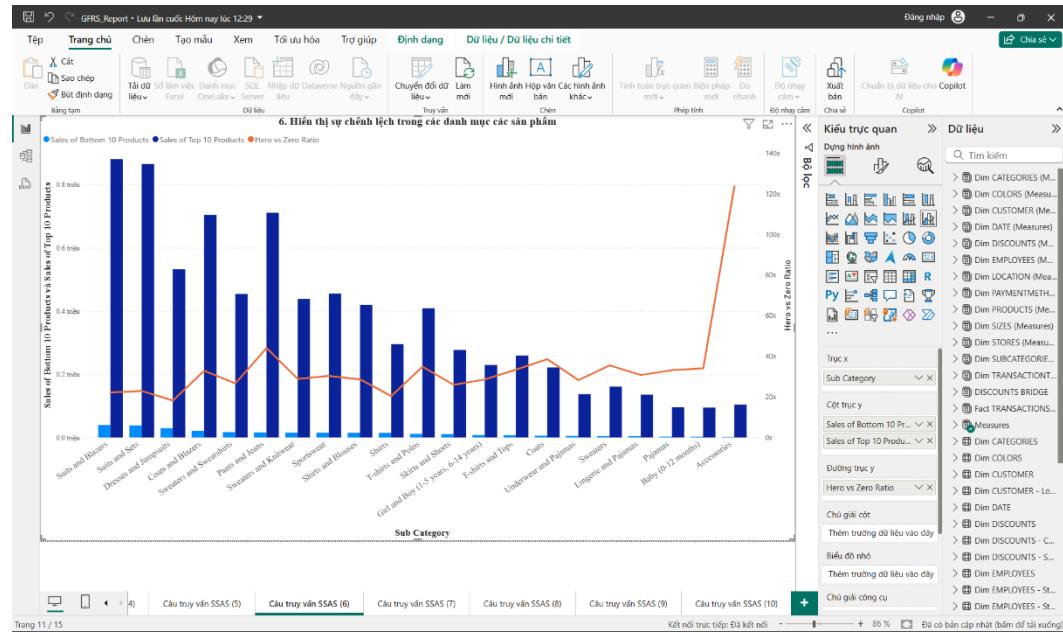
- **Thực hiện:** Chọn **Biểu đồ đường thẳng và biểu đồ cột liên cụm** để minh họa câu truy vấn này, kéo **Sub Category** từ **Dim PRODUCT – Sub Category** vào **Trục X**, sau đó kéo **Measure Line Total, Repurchase Customer Count, Unique Customer Count** từ **Fact TRANSACTIONS** vào **Cột trực y**, sau cùng kéo **Product Repurchase Rate** vào **Đường trực y**. Sau đó kéo **Year Of Purchase** từ **Dim DATE** vào bộ lọc, chỉnh chỉ lấy năm **2024**.
- **Kết quả:**



Hình 535. Câu truy vấn phàn SSAS trong Power BI 5

### III.2.4.3.6. Sự chênh lệch trong danh mục con các sản phẩm

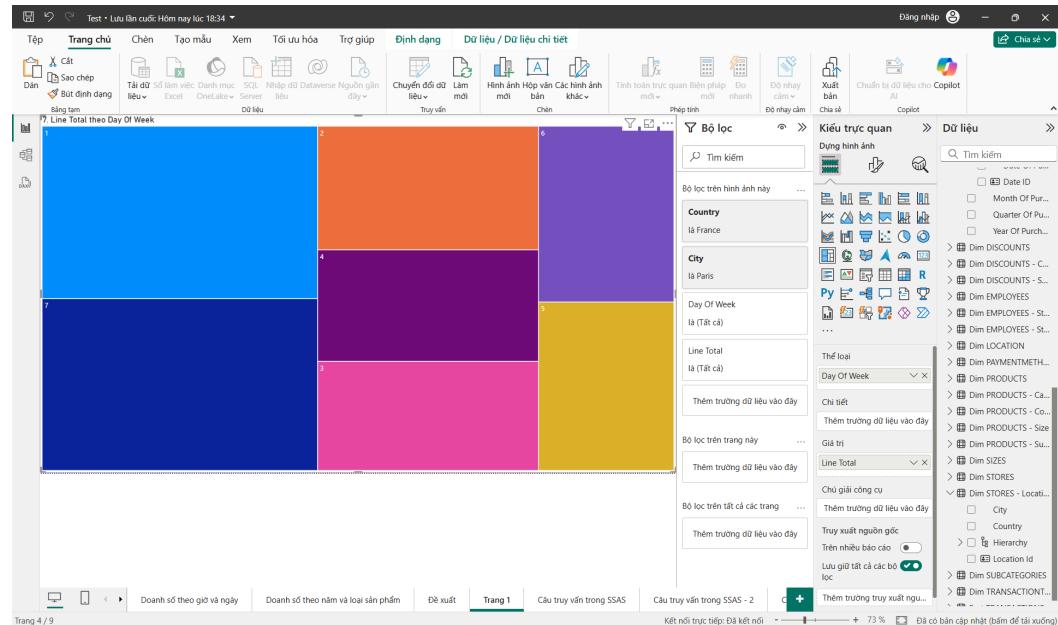
- **Thực hiện:** Chọn **Biểu đồ đường thẳng và biểu đồ cột liê cụm** để minh họa truy vấn này, kéo **Sub Category** từ **Dim PRODUCT – Sub Category** vào **Trục X**, sau đó kéo **Measure Sales of Bottom 10 Products, Sales of Top 10 Products** vào **Trục Y**, cuối cùng ta kéo **Hero vs Zero** vào **Đường trực y**.
- **Kết quả:**



Hình 536. Câu truy vấn phàn SSAS trong Power BI 6

### III.2.4.3.7. Báo biểu doanh số các ngày trong tuần ở Paris Pháp

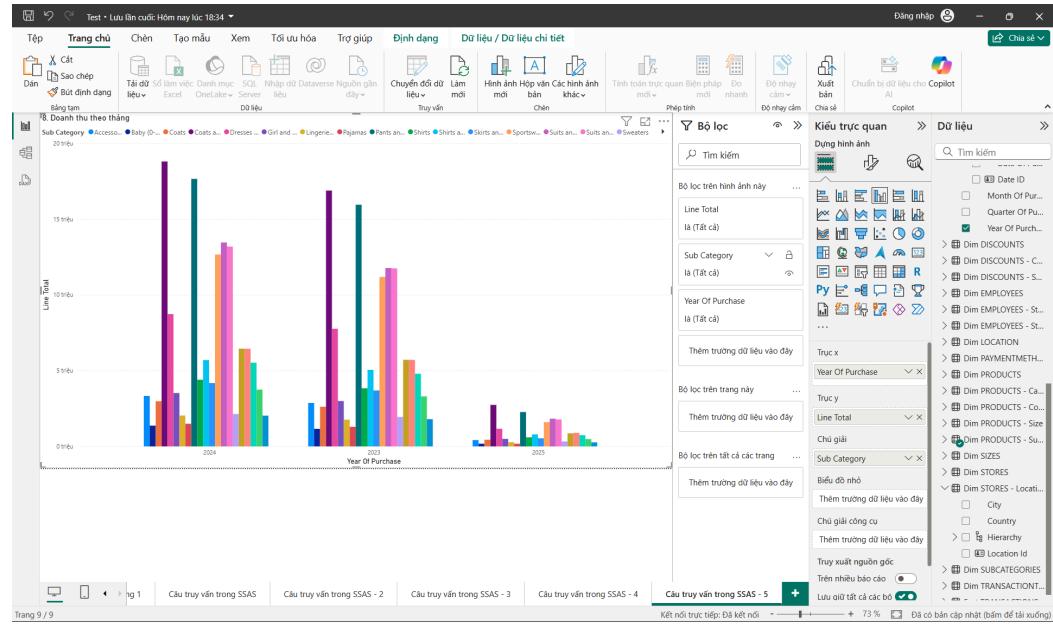
- **Thực hiện:** Chọn **Biểu đồ cây** để minh họa cho câu truy vấn này. Đầu tiên kéo **Day Of Week** từ **Dim DATE** vào **Thẻ loại**. Sau đó kéo **Measure Line Total** từ **Fact TRANSACTIONS** vào **Giá trị**. Sau cùng, ta kéo **Country**, **City** từ **Dim STORES – Location** vào tab **Bộ lọc**, chỉnh **Country** thành **France**, và **City** thành **Paris**
- **Kết quả:**



Hình 537. Câu truy vấn phản SSAS trong Power BI 7

### III.2.4.3.8. Báo biểu doanh số các danh mục con qua các năm

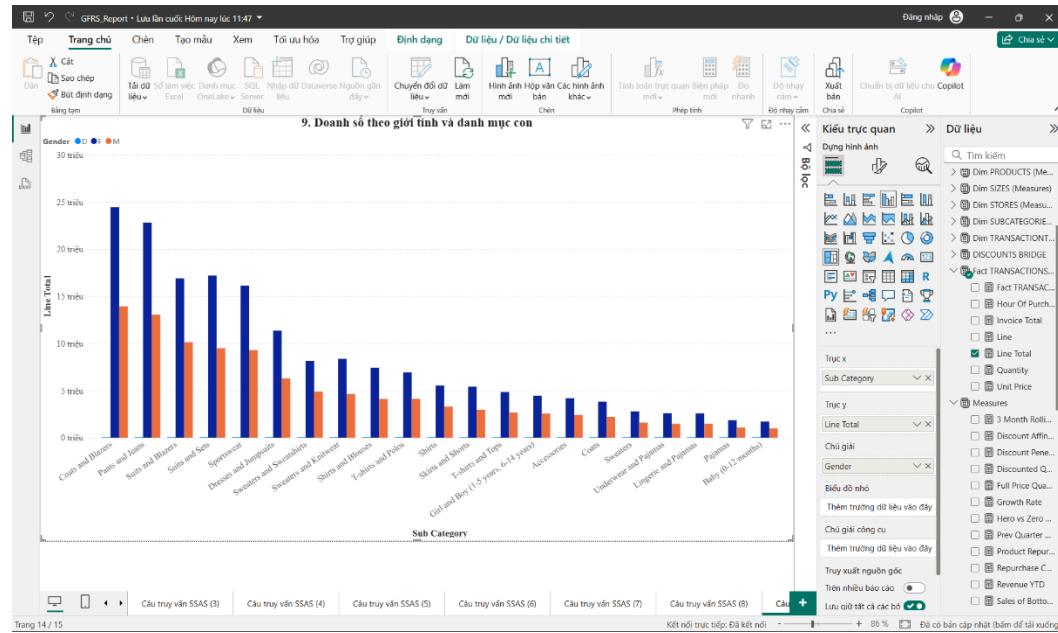
- **Thực hiện:** Chọn **Biểu đồ cột bó cụm** để minh họa truy vấn này, kéo **Year Of Purchase** từ **Dim DATE** vào **Trục X**, sau đó kéo **Measure Line Total** từ **Fact TRANSACTIONS** vào **Trục Y**, cuối cùng ta Kéo **Sub Category** từ **Dim PRODUCTS – Sub Category** vào **Chú giải**.
- **Kết quả:**



Hình 538. Câu truy vấn phần SSAS trong Power BI 8

### III.2.4.3.9. Báo biểu doanh thu theo danh mục con và giới tính

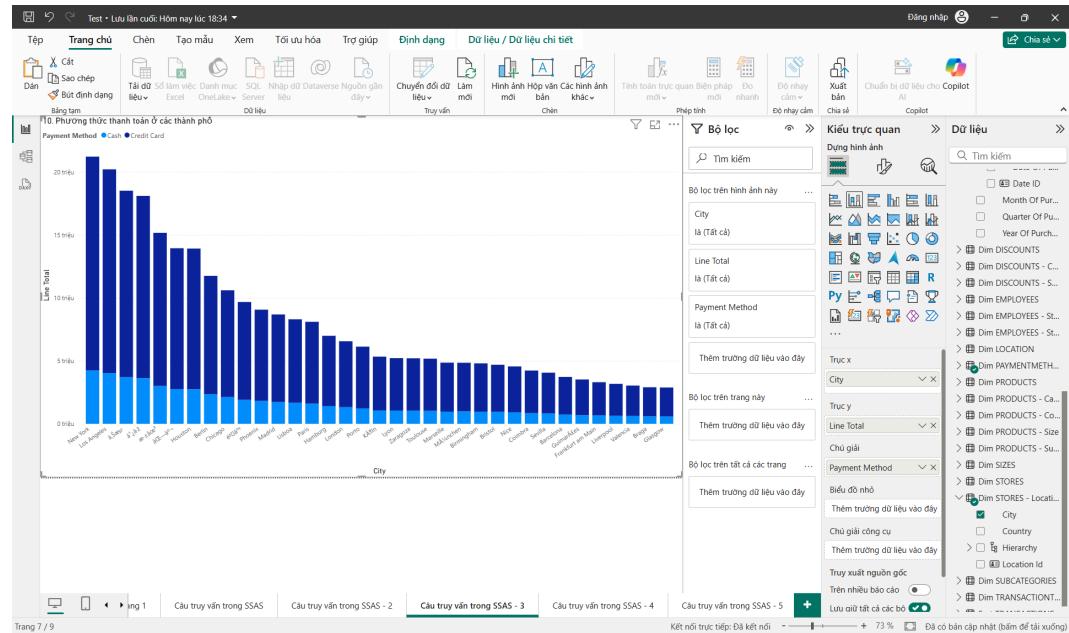
- **Thực hiện:** Chọn **Biểu đồ cột bô cụm** để minh họa truy vấn này, kéo **Sub Category** từ **Dim PRODUCTS – Sub Category** vào **Trục X**, sau đó kéo **Measure Line Total** từ **Fact TRANSACTIONS** vào **Trục Y**, cuối cùng ta Kéo **Gender** từ **Dim CUSTOMERS** vào **Chú giải**.
- **Kết quả:**



Hình 539. Câu truy vấn phàn SSAS trong Power BI 9

### III.2.4.3.10. Báo biểu doanh số theo phương thức thanh toán ở các thành phố

- **Thực hiện:** Chọn **Biểu đồ cột bô cụm** để minh họa truy vấn này, kéo **City** từ **Dim STORES – Location** vào **Trục X**, sau đó kéo **Measure Line Total** từ **Fact TRANSACTIONS** vào **Trục Y**, cuối cùng ta Kéo **Payment Method** từ **Dim PAYMENTMETHOD** vào **Chú giải**.
- **Kết quả:**

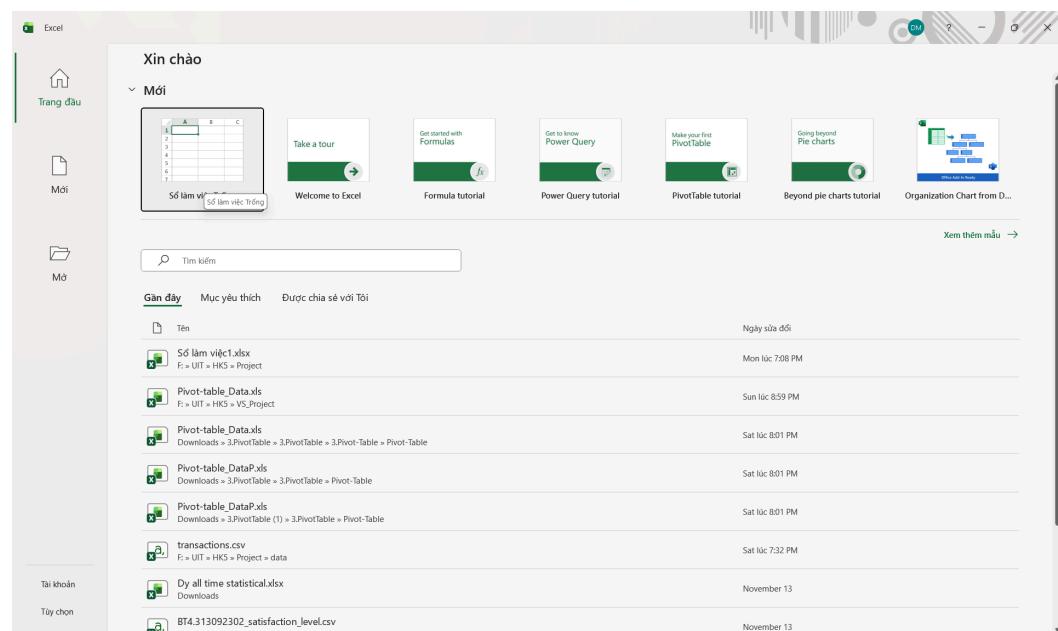


Hình 540. Câu truy vấn phân SSAS trong Power BI 10

### III.2.4.4. Thực hiện lại các câu truy vấn SSAS trong Excel

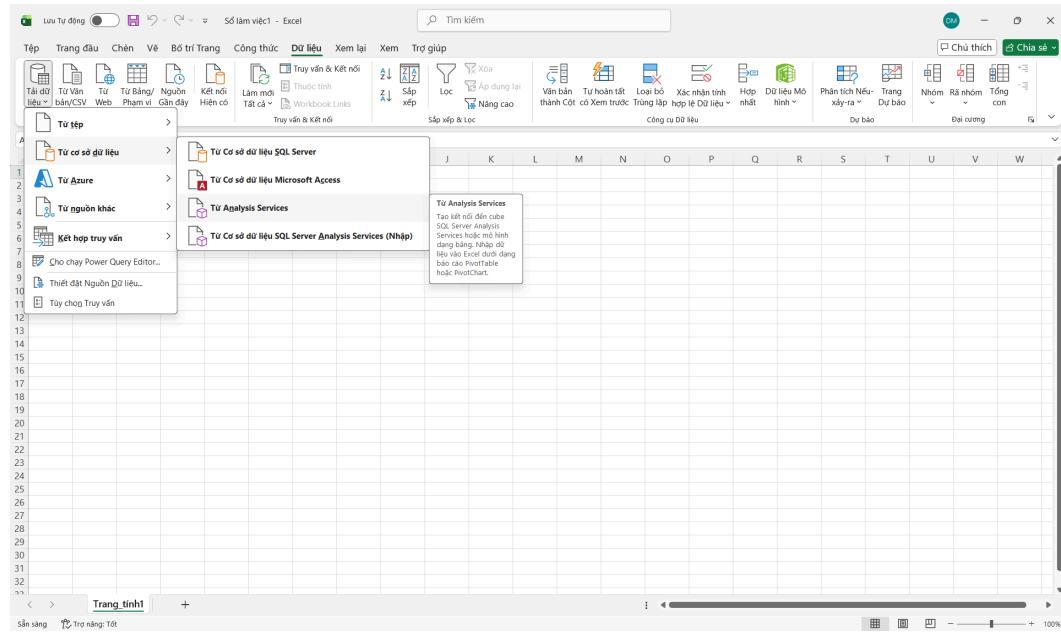
#### III.2.4.4.1. Kết nối Excel với Analysis Services

- **Bước 1:** Mở phần mềm Excel, ta chọn **Sổ làm việc Trống**



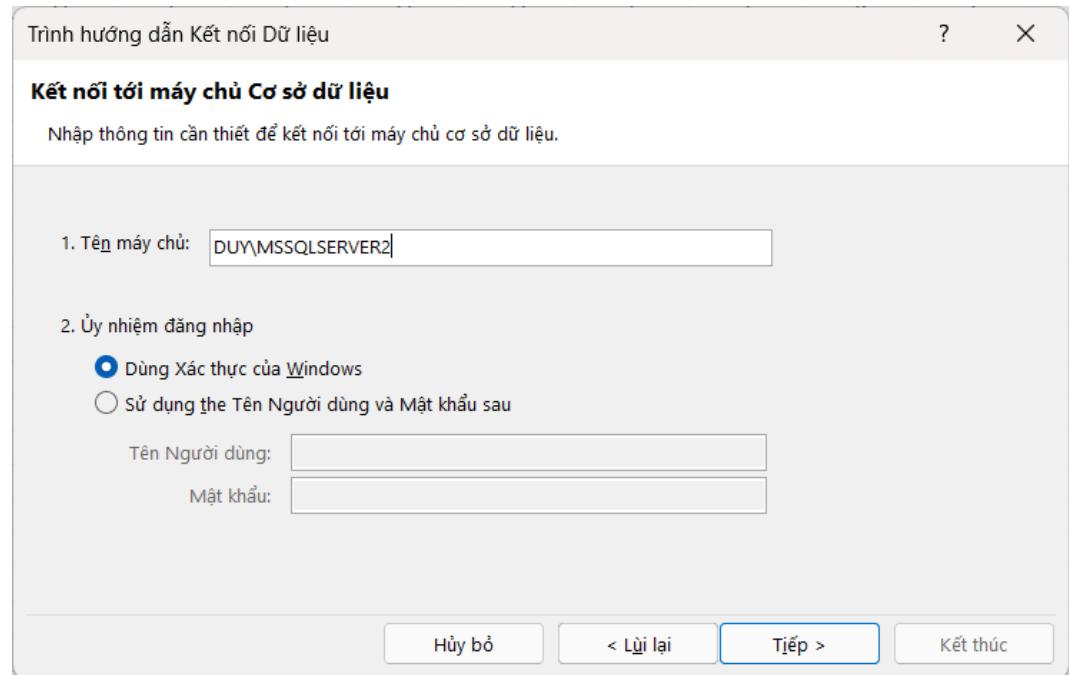
Hình 541. Tạo Sổ làm việc Trống mới

- **Bước 2:** Ta chuyển sang tab **Dữ liệu**, chọn **Tải dữ liệu**  **Từ cơ sở dữ liệu**  
 **Từ Analysis Services**



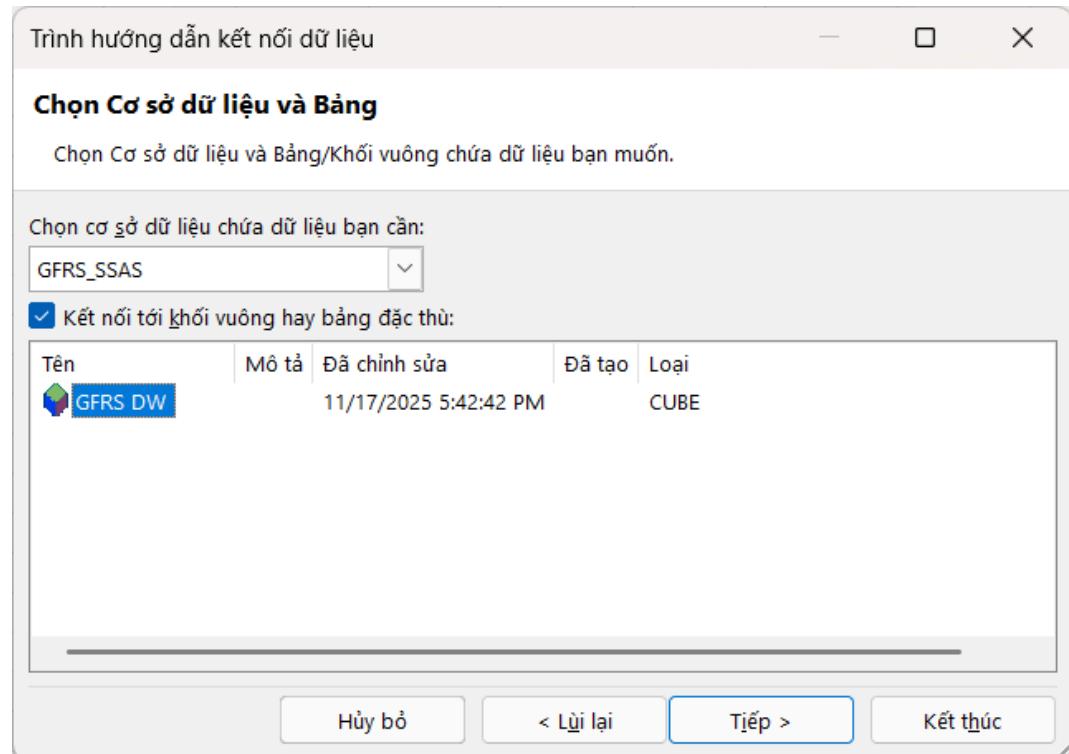
Hình 542. Chọn kết nối với Analysis Services

- **Bước 3:** Cửa sổ **Trình hướng dẫn Kết nối Dữ liệu**, nhập **Tên máy chủ (DUY\MSSQLSERVER2)**, sau đó chọn **Tiếp**



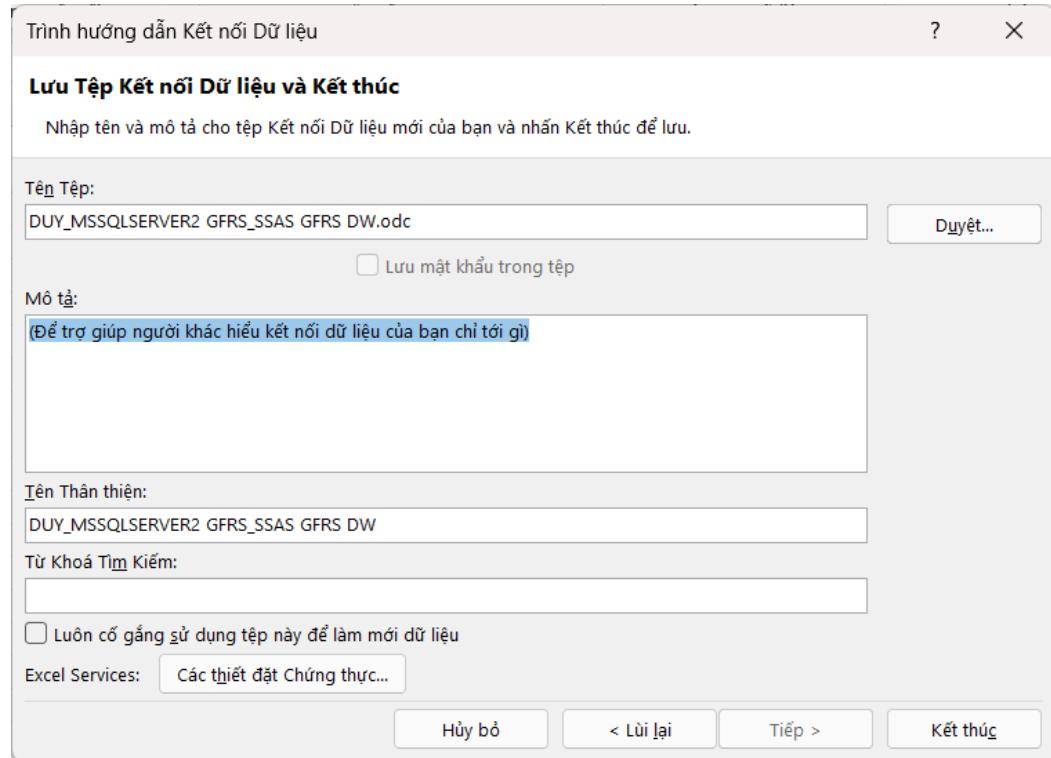
Hình 543. Chọn tên máy chủ phù hợp để kết nối

- **Bước 4:** Chọn kho dữ liệu chứa dữ liệu bạn cần là **GFRS\_SSAS**, sau đó chọn **GFRS\_DW**, cuối cùng chọn **Tiếp**



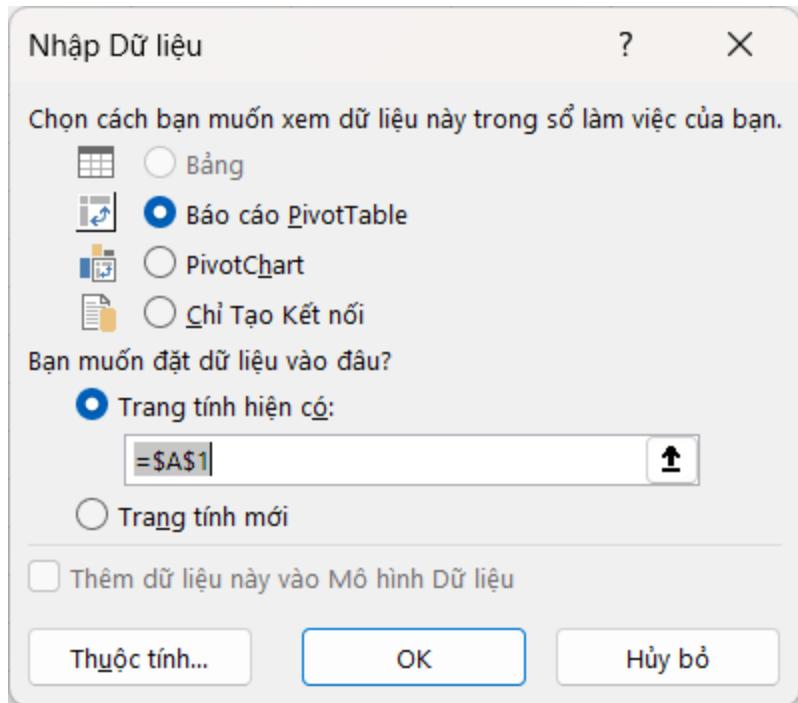
Hình 544. Chọn kho dữ liệu GFRS DW

- **Bước 5:** Ta đặt tên tệp và chọn nơi lưu trữ, cuối cùng bấm **Kết thúc** để hoàn thành quá trình kết nối.



Hình 545. Lưu tệp Kết nối Dữ liệu và Kết thúc

- **Bước 6:** Màn hình **Nhập Dữ liệu** sẽ hiện lê, ta chọn **Báo cáo PivotTable** và chọn **Trang tính hiện có**. Cuối cùng **OK** để hoàn thành



Hình 546. Nhập dữ liệu vào Excel

Hình 547. Màn hình Excel sau khi hoàn thành quá trình kết nối với Analysis Services

### **III.2.4.4.2. Thực hiện các câu truy vấn**

#### **III.2.4.4.2.1 Doanh số theo roll up theo năm**

- **Thực hiện:** Kéo Hierarchy từ Dim DATE vào Hàng, sau đó kéo Measure Line Total từ Fact TRANSACTIONS vào Giá trị.
- **Kết quả:**

	A	B	C
1	A	Nhân Hàng	
2		1. Doanh số roll up theo năm	
3		Nhân Hàng - Line Total	
4		2023	
5		1	
6		1	
7		1	
8		1	736199
9		10	153587
10		11	118171
11		12	123981
12		13	117892
13		14	233302
14		15	245827
15		16	174378
16		17	122995
17		18	119331
18		19	117365
19		2	171447
20		20	122857
21		21	235083
22		22	310265
23		23	117657
24		24	120175

Hình 548. Câu truy vấn phản SSAS trong Excel 1

#### **III.2.4.4.2.2 Doanh số và số lượng sản phẩm roll up theo quốc gia**

- **Thực hiện:** Kéo Hierarchy từ Dim STORES – Location vào Hàng, sau đó kéo Measure Line Total từ Fact TRANSACTIONS vào Giá trị. Tiếp đó kéo Transaction từ Dim TRANSACTIONTYPE vào bộ lọc, sau đó chỉnh thành Sale
- **Kết quả:**

2. Doanh số roll up theo quốc gia

	Transaction	Sale	
1	Nhân Hàng	Line Total	Quantity
2	Transaction	79847107	1618960
3	Đức		
4	Berlin	12457092	284338
5	Frankfurt am Main	3494249	79929
6	Hamburg	7406324	169491
7	Köln	5642746	129749
8	München	5142145	118065
9	Espanha	26189419	594966
10	France		
11	Lyon	5512233	126294
12	Marseille	5150469	117395
13	Nice	4824277	109782
14	Paris	8599944	196301
15	Toulouse	5480042	125953
16	Portugal	26576223	603521
17	United Kingdom	23414617	653022
18	United States	79459521	1758358
19	Tổng Cuối	2.99E+08	6686124

Hình 549. Câu truy vấn phần SSAS trong Excel 2

### III.2.4.4.2.3 Doanh số và tốc độ tăng trưởng theo các quý

- **Thực hiện:** Kéo Quarter Of Purchase và Year Of Purchase từ Dim DATE vào Hàng, sau đó kéo Measure Line Total từ Fact TRANSACTIONS vào Giá trị. Kéo Growth Rate từ Giá trị vào Cột.
- **Kết quả:**

3. Doanh số và tốc độ tăng trưởng từng quý

	Year Of Purchase	Quarter Of Purchase	Line Total	Growth Rate
1	2023	1	22357198	22.09%
2	2023	2	27295815	-0.26%
3	2023	3	27225662	72.32%
4	2023	4	46914678	-44.47%
5	2024	1	26050773	14.76%
6	2024	2	31147512	4.19%
7	2024	3	52729965	69.29%
8	2024	4	18818983	-64.31%
9	Tổng Cuối		2.82E+08	-100.00%

Hình 550. Câu truy vấn phần SSAS trong Excel 3

### **III.2.4.4.2.4 Doanh số theo tháng**

- **Thực hiện:** kéo Year Of Purchase ,Quarter Of Purchase, Month Of Purchase từ Dim DATE vào Hàng, sau đó kéo Measure Line Total, 3 Month Rolling Avg, Revenue YTD từ Fact TRANSACTIONS vào Giá trị.
- **Kết quả:**

The screenshot shows a Microsoft Excel spreadsheet titled "GFRS\_Pivot.xlsx". A PivotTable is being constructed in the range M3:Q24. The PivotTable Fields pane on the right side lists various dimensions and measures. Dimensions include "Dim CATEGORIES", "Dim COLORS", "Dim CUSTOMER", and "Dim DATE". Measures include "Line Total", "3 Month Rolling Avg", and "Revenue YTD". The main table area displays monthly sales data for different categories and years, with some cells showing formulas like =SUM or =AVERAGE.

4. Doanh số theo tháng			
Nhà Hàng	Line Total	3 Month Rolling Avg	Revenue YTD
2023			
1	5787930	5787930	5,787,930.00
2	4455313	5121621.5	10,243,243.00
3	12113955	7452399.333	22,357,198.00
2024			
1	10102406	8890558	32,459,604.00
2	10081884	10766081.67	42,541,488.00
3	7111525	9098605	49,653,013.00
4	7151749	8115052.667	56,804,762.00
5	6250079	6837784.333	63,054,841.00
6	13823834	9075220.667	76,878,675.00
10	13513989	11195967.33	90,392,664.00
11	9920788	12422537	100,322,452.00
12	23470901	15638226	123,793,353.00
2024			
1	6152101	13184263.33	6,152,101.00
2	5485269	11702757	11,637,370.00

Hình 551. Câu truy vấn phân SSAS trong Excel 4

### **III.2.4.4.2.5 Sự yêu thích của khách hàng năm 2024**

- **Thực hiện:** kéo Sub Category từ Dim PRODUCTS – Sub Category vào Hàng, sau đó kéo Measure Line Total, Repurchase Customer Count, Unique Customer Count, Repurchase Customer Rate từ Fact TRANSACTIONS vào Giá trị. Sau đó kéo Year Of Purchase từ Dim DATE vào Bộ lọc, chỉnh lại thành 2024.
- **Kết quả:**

5. Sự yêu thích của khách hàng năm 2024

	R	S	T	U	V
1					
2	Hierarchy	2024			
3					
4	Nhãn Hàng	Unique Customer Count	Repurchase Customer Count	Product Repurchase Rate	Line Total
5	Accessories	169296	20280	11.93%	3329342
6	Baby (0-12 months)	43074	1370	3.18%	1369022
7	Coats	61012	3163	5.18%	2979187
8	Coats and Blazers	232067	38164	16.45%	18800728
9	Dresses and Jumpsuits	145369	15046	10.35%	8731077
10	Girl and Boy (1-5 years, 6-14 years)	85658	6562	7.66%	3519797
11	Lingerie and Pajamas	63968	2949	4.61%	2037147
12	Pajamas	48953	1839	3.76%	1491943
13	Pants and Jeans	241773	41331	17.09%	17655378
14	Slirts	145673	15548	10.67%	4397205
15	Shirts and Blouses	140825	14064	9.99%	5695675
16	Skirts and Shorts	134696	12962	9.62%	4181250
17	Sportswear	236667	40028	16.91%	12664451
18	Suits and Blazers	123482	11105	8.99%	13456719
19	Suits and Sets	122222	10530	8.62%	13188193
20	Sweaters	55355	2464	4.45%	2139005
21	Sweaters and Knitwear	132369	12583	9.51%	6448500
22	Sweaters and Sweatshirts	133279	13427	10.07%	6441478
23	T-shirts and Polos	141197	14625	10.36%	5522280
24	T-shirts and Tops	126828	11408	8.99%	3746805

Hình 552. Câu truy vấn phần SSAS trong Excel 5

### III.2.4.4.2.6 Sự chênh lệch trong các danh mục con sản phẩm

- **Thực hiện:** kéo Sub Category từ Dim PRODUCTS – Sub Category vào Hàng, sau đó kéo Measure Sales of Top 10 Products, Sales of Top Bottom 10 Products, Hero vs Zero Ratio vào Giá trị.
- **Kết quả:**

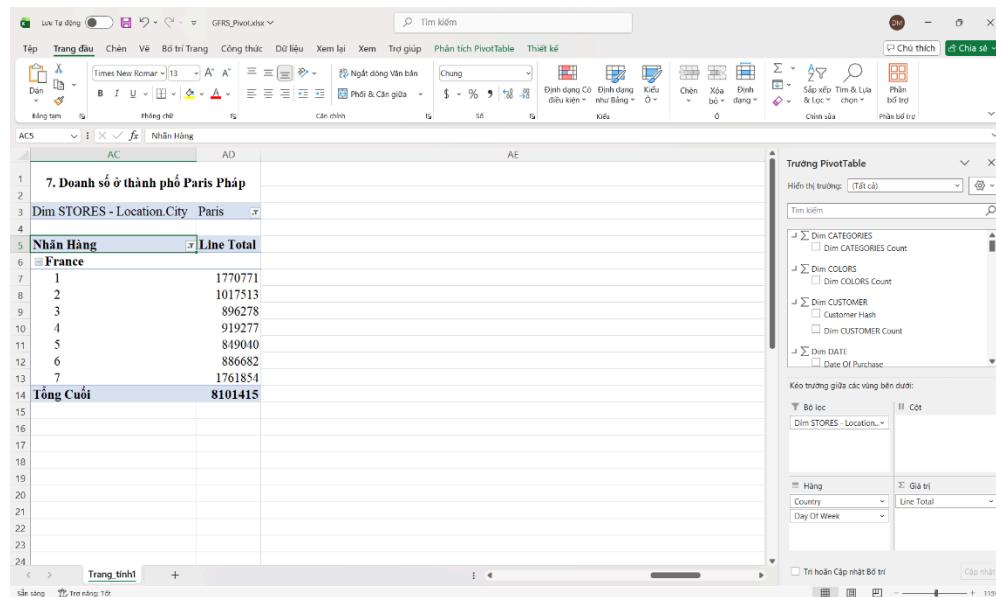
6. Sự chênh lệch trong các nhóm sản phẩm

	X	Y	Z	AA	AB
1					
2					
3	Nhãn Hàng	Sales of Bottom 10 Products	Sales of Top 10 Products	Hero vs Zero Ratio	
4	Accessories	840.00	103,835.00	123,6130952	
5	Baby (0-12 months)	2,778.00	94,513.00	34,02195824	
6	Coats	5,749.00	221,590.00	38,54409463	
7	Coats and Blazers	21,423.00	703,853.00	32,85501564	
8	Dresses and Jumpsuits	29,277.00	532,495.00	18,18816819	
9	Girl and Boy (1-5 years, 6-14 years)	8,023.00	229,320.00	28,58282438	
10	Lingerie and Pajamas	4,404.00	135,444.00	30,75476839	
11	Pajamas	2,904.00	96,188.00	33,12258953	
12	Pants and Jeans	16,142.00	711,049.00	44,0496221	
13	Slirts	14,476.00	295,097.00	20,38525836	
14	Shirts and Blouses	14,693.00	419,774.00	28,586965902	
15	Skirts and Shorts	10,681.00	276,802.00	25,91536373	
16	Sportswear	14,997.00	454,841.00	30,32879909	
17	Suits and Blazers	39,806.00	880,413.00	22,11759534	
18	Suits and Sets	37,761.00	865,349.00	22,91647467	
19	Sweaters	4,535.00	161,004.00	35,50253583	
20	Sweaters and Knitwear	15,221.00	438,593.00	28,81499244	
21	Sweaters and Sweatshirts	17,055.00	453,952.00	26,61694518	
22	T-shirts and Polos	11,718.00	408,424.00	34,85441202	
23	T-shirts and Tops	7,749.00	259,009.00	33,42482901	
24	Underwear and Pajamas	4,855.00	136,930.00	28,20391349	

Hình 553. Câu truy vấn phần SSAS trong Excel 6

### **III.2.4.4.2.7 Doanh số các ngày trong tuần ở Paris Pháp**

- **Thực hiện:** Đầu tiên kéo Day Of Week từ Dim DATE vào Hàng. Sau đó kéo Measure Line Total từ Fact TRANSACTIONS vào Giá trị. Sau cùng, ta kéo Country, City từ Dim STORES – Location vào tab Bộ lọc, chỉnh Country thành France, và City thành Paris
- **Kết quả:**



Hình 554. Câu truy vấn phản SSAS trong Excel 7

### **III.2.4.4.2.8 Doanh số các danh mục con qua các năm**

- **Thực hiện:** kéo Year Of Purchase từ Dim DATE vào Cột, sau đó kéo Measure Line Total từ Fact TRANSACTIONS vào Giá trị, cuối cùng ta Kéo Sub Category từ Dim PRODUCTS – Sub Category vào Hàng.
- **Kết quả:**

Hình 555. Câu truy vấn phản SSAS trong Excel 8

### III.2.4.4.2.9 Doanh số các danh mục con theo giới tính

- Thực hiện:** Kéo Sub Category từ Dim PRODUCTS – Sub Category vào Hàng, sau đó kéo Measure Line Total từ Fact TRANSACTIONS vào Giá trị, cuối cùng ta kéo Gender từ Dim CUSTOMERS vào Cột.
- Kết quả:**

Hình 556. Câu truy vấn phản SSAS trong Excel 9

### III.2.4.4.2.10 Doanh số theo các phương thức thanh toán ở các thành phố

- **Thực hiện:** Kéo City từ Dim STORES – Location vào Hàng, sau đó kéo Measure Line Total từ Fact TRANSACTIONS vào Giá trị, cuối cùng ta Kéo Payment Method từ Dim PAYMENTMETHOD vào Cột.
- **Kết quả:**

Hình 557. Câu truy vấn phân SSAS trong Excel 10

## CHƯƠNG 4: KHAI PHÁ DỮ LIỆU (DATA MINING)

### IV.1. Ngữ cảnh và đặt vấn đề

**Mục tiêu:** Chuyển đổi dữ liệu giao dịch thành sự thấu hiểu hành vi khách hàng (Customer Insight).

**Nội dung:** Trong bối cảnh dữ liệu giao dịch (Transaction Logs) của doanh nghiệp ngày càng lớn, việc nhìn vào các con số doanh thu tổng hợp là không đủ để đưa ra các chiến lược kinh doanh chính xác. Dữ liệu đầu vào của hệ thống hiện tại là hàng triệu dòng

log ghi nhận lịch sử mua hàng, bao gồm: thời gian mua, số lượng, giá trị đơn hàng và mã khách hàng.

Tuy nhiên, dữ liệu này ở dạng thô rất rời rạc. Chúng ta không biết được ai là khách hàng trung thành, ai sắp rời bỏ dịch vụ, hay ai chỉ săn khuyến mãi. Do đó, yêu cầu đặt ra là phải áp dụng các kỹ thuật Khai phá dữ liệu (Data Mining), cụ thể là bài toán **Phân cụm khách hàng (Customer Segmentation)**.

Việc phân cụm sẽ giúp doanh nghiệp nhóm các khách hàng có hành vi tương đồng lại với nhau, từ đó cá nhân hóa các chiến dịch Marketing, tối ưu hóa nguồn lực chăm sóc khách hàng và tối đa hóa lợi nhuận theo nguyên lý Pareto (20% khách hàng mang lại 80% doanh thu).

## **IV.2. Phương pháp tiếp cận và hướng đi**

Để giải quyết bài toán này, tôi đề xuất quy trình triển khai theo mô hình tiêu chuẩn **CRISP-DM** (hoặc giản lược hóa) với các bước cụ thể sau:

### **IV.2.1. Lựa chọn thuật toán**

Tôi sẽ sử dụng phương pháp so sánh giữa hai thuật toán gom cụm phổ biến để tìm ra kết quả tối ưu nhất:

#### **IV.2.1.1. K-Means (MiniBatchKMeans):**

- o Lý do chọn: Đây là thuật toán phổ biến nhất, tốc độ xử lý nhanh và phù hợp với tập dữ liệu lớn (như trong đồ án này là hơn 1.2 triệu dòng). K-Means phân chia dữ liệu thành \$K\$ nhóm rời rạc dựa trên khoảng cách Euclid.

#### **IV.2.1.2. Hierarchical Clustering (Phân cụm phân cấp)**

- o Lý do chọn: Khác với K-Means cần xác định trước số cụm K, thuật toán này xây dựng một cấu trúc phân cấp (dendrogram). Nó giúp chúng ta có cái nhìn trực quan về cách các nhóm dữ liệu gộp lại với nhau.
- o Lưu ý kỹ thuật: Do dữ liệu lớn, Hierarchical Clustering tốn rất nhiều bộ nhớ.

#### **IV.2.2.Các bước triển khai kỹ thuật (Pipeline)**

##### **Bước 1: Tiền xử lý & Làm sạch dữ liệu (Data Cleaning):**

- o Xử lý các giá trị rỗng (Null values) và loại bỏ các giao dịch trùng lặp.
- o Lọc bỏ các giao dịch bị hủy hoặc trả hàng (nếu có Quantity < 0 hoặc Monetary < 0).
- o Chuyển đổi kiểu dữ liệu thời gian (Date) để tính toán khoảng cách ngày.

##### **Bước 2: Rút trích đặc trưng (Feature Engineering - RFM+):**

- o Từ dữ liệu giao dịch thô, chuyển đổi thành dữ liệu cấp độ Khách hàng (Customer Level) dựa trên mô hình **RFM**:
  - **Recency (R)**: Số ngày kể từ lần mua cuối cùng.
  - **Frequency (F)**: Tổng số lần mua hàng.
  - **Monetary (M)**: Tổng số tiền đã chi tiêu.
- o *Mở rộng*: Tính thêm các chỉ số phụ trợ như **AOV** (Giá trị trung bình đơn hàng) và **AvgBasketSize** (Kích thước giỏ hàng trung bình) để làm rõ hơn hành vi.

##### **Bước 3: Xử lý ngoại lai (Outlier Handling):**

- o Dữ liệu tài chính thường có phân phối lệch (skewed). Cần áp dụng kỹ thuật Capping (ví dụ: giới hạn ở phân vị 99.9%) để tránh việc các “siêu khách hàng” làm lệch mô hình quá xa.
- o Áp dụng biến đổi log (Log Transformation) cho các biến có độ lệch cao như Monetary.

##### **Bước 4: Chuẩn hóa dữ liệu (Normalization):**

- o Sử dụng **StandardScaler** hoặc **RobustScaler** để đưa các đặc trưng về cùng một miền giá trị, đảm bảo thuật toán gom cụm không bị thiên vị bởi các biến có giá trị lớn (như Doanh thu) so với biến có giá trị nhỏ (như Tần suất).

##### **Bước 5: Xác định số cụm tối ưu:**

- o Chạy thuật toán với số cụm k từ 3 đến 8.

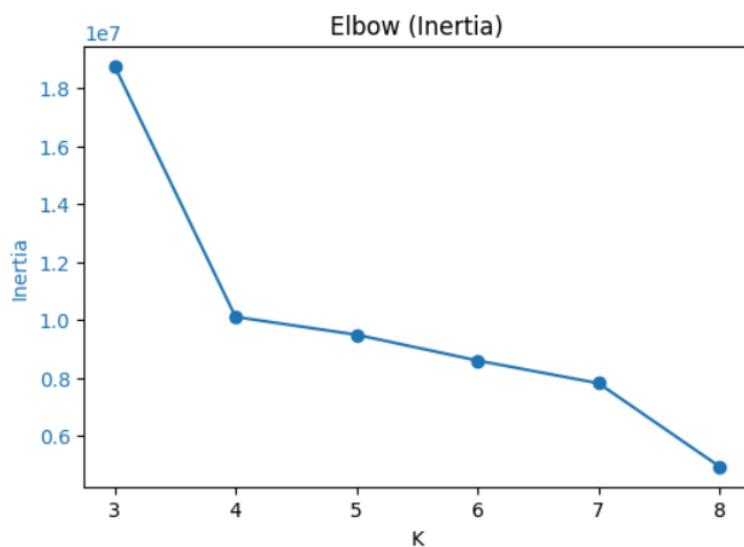
- o Sử dụng phương pháp **Elbow (Khuỷu tay)** hoặc **Silhouette** dựa trên chỉ số Inertia để chọn điểm gãy tối ưu.

### IV.3. Kết quả đạt được

#### IV.3.1. Kết quả từ thuật toán K-means

Đây là thuật toán chính được sử dụng để phân loại toàn bộ 1.2 triệu giao dịch.

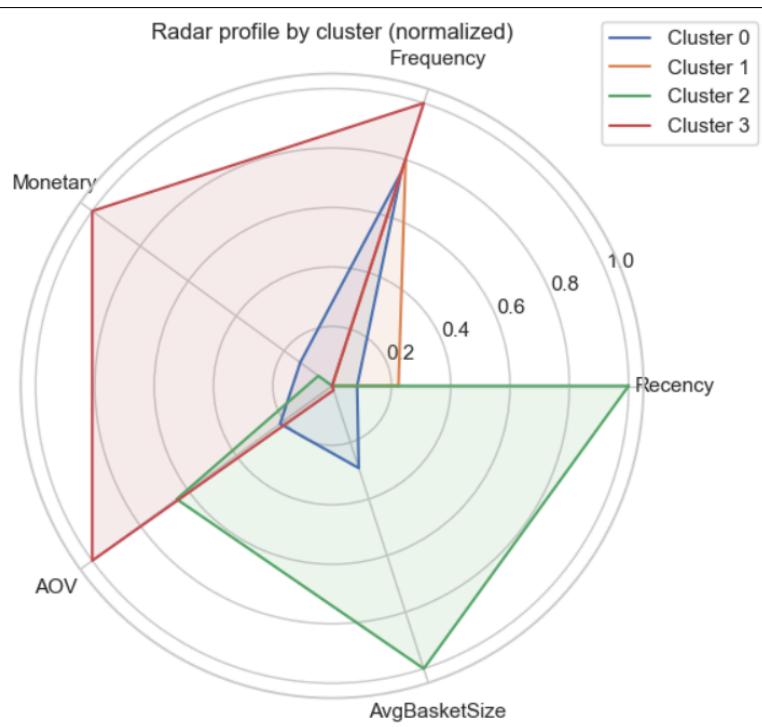
##### IV.3.1.1. Xác định số cụm (K):



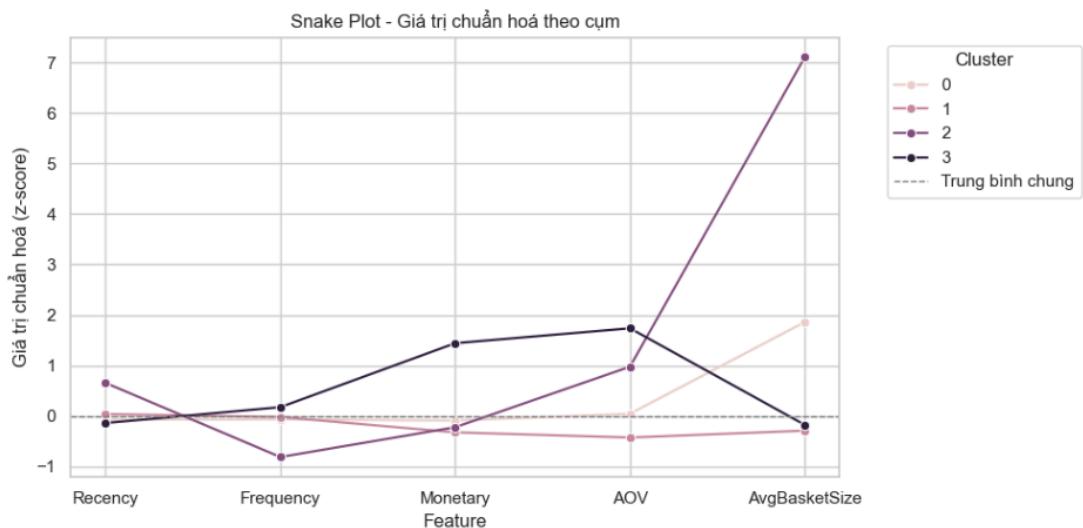
Hình 545. Biểu đồ Elbow xác định số cụm của K-means

- **Nhận xét:** Biểu đồ cho thấy độ lỗi giảm mạnh và bắt đầu bão hòa tại điểm K=4 (khuỷu tay), gợi ý đây là số lượng cụm tối ưu.

#### IV.3.1.2. Phân tích đặc điểm các cụm (Profiling):



Hình 558. Biểu đồ phân tích đặc điểm cụm thứ 1



Hình 559. Biểu đồ phân tích đặc điểm cụm thứ 2

Cluster	Recency	Frequency	Monetary	AOV	AvgBasketSize
0	203.05	4.69	469.47	119.99	1.57
1	224.57	4.90	209.80	44.49	1.03
2	344.42	1.32	321.91	269.94	2.91
3	190.01	5.78	2162.31	392.36	1.05

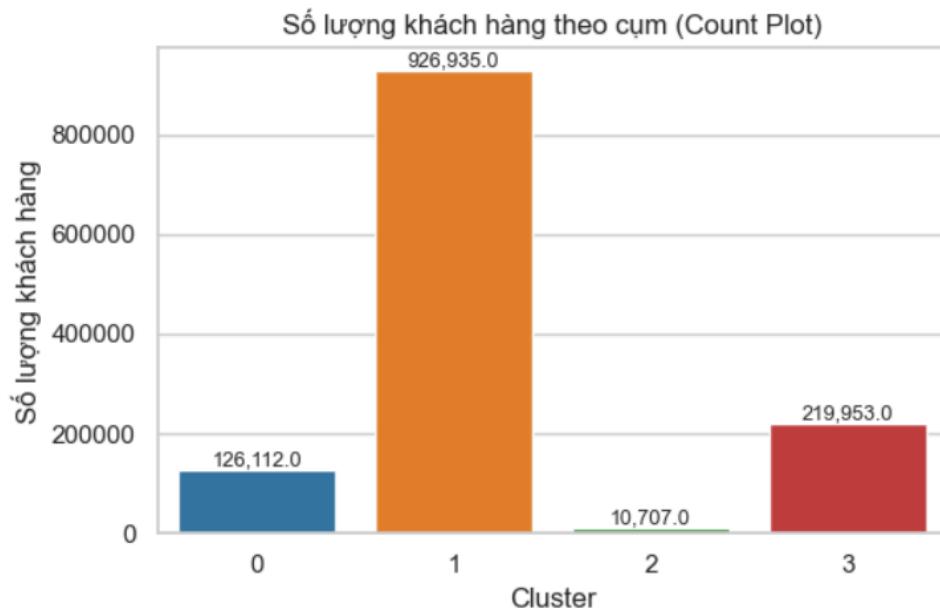
Hình 560. Biểu đồ phân tích đặc điểm cụm thứ 3

#### Mô tả chi tiết:

- **Cluster 3: Khách hàng Tốt Nhất (Best Customers)**
  - Thẻ mạnh: Tần suất (Frequency) cao nhất, Monetary cao nhất, và AOV cao nhất. Họ có Recency thấp nhất, nghĩa là họ mua hàng gần đây nhất.
  - Hành vi: Mua hàng thường xuyên, chi tiêu nhiều nhất, và giá trị mỗi giao dịch cao nhất. Đây là nhóm VIP đóng góp doanh thu lớn nhất.
- **Cluster 0: Khách hàng Tiềm Năng/Mới**
  - Thẻ mạnh: Các chỉ số đều ổn định và trên mức trung bình chung. Recency, Frequency, Monetary đều có Z-score gần 0 hoặc hơi dương.
  - Hành vi: Họ có hành vi mua sắm khá cân bằng, không quá nổi trội nhưng cũng không quá kém. Họ có Monetary và AOV ở mức khá.
- **Cluster 1: Khách hàng Tần Suất Cao/Giá Trị Thấp (High-Frequency/Low-Value)**
  - Thẻ mạnh: Số lượng khách hàng lớn nhất và Frequency cao (Z-score trên 0).
  - Hạn chế: Monetary, AOV, và AvgBasketSize thấp nhất (Z-score âm).
  - Hành vi: Mua hàng thường xuyên nhưng mỗi lần mua chỉ chi tiêu rất ít.
- **Cluster 2: Khách hàng Đơn Lớn/Ngủ Đông (Big Spenders/At-Risk)**

- o Thể mạnh: AvgBasketSize và AOV RÂT cao (cao nhất trong các cụm).
- o Hạn chế: Recency cao nhất (đã lâu không mua hàng) và Frequency thấp nhất (ít mua hàng). Đây cũng là nhóm ít khách hàng nhất và đóng góp doanh thu thấp nhất về tổng thể.
- o Hành vi: Họ từng mua những đơn hàng rất lớn nhưng hiện tại đã ngừng mua hàng hoặc mua rất thưa thớt (nguy cơ mất khách hàng).

#### IV.3.1.3. Quy mô các nhóm:



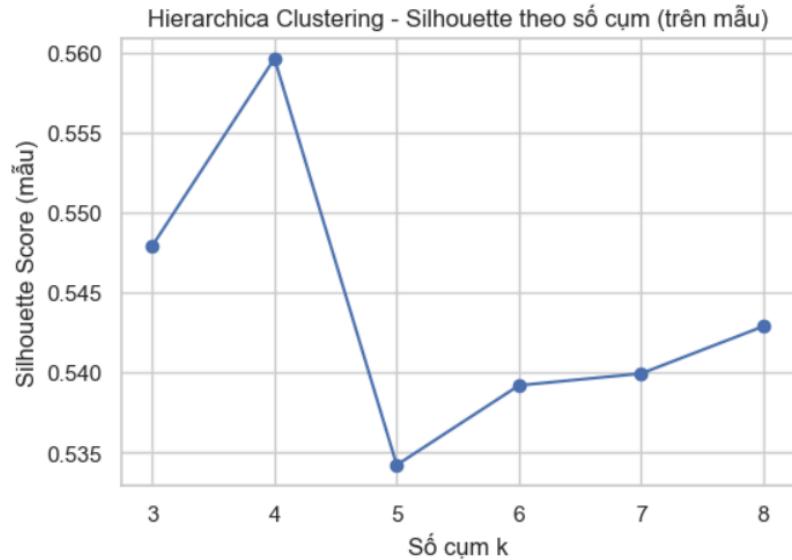
*Hình 561. Biểu đồ quy mô các nhóm*

- **Nhận xét:** Nhóm khách hàng phổ thông chiếm tỷ trọng lớn nhất, trong khi nhóm VIP số lượng ít nhưng đóng góp doanh thu cao.

#### IV.3.2. Kết quả từ thuật toán Hierarchical Clustering

Đây là thuật toán sử dụng phân cụm phân cấp

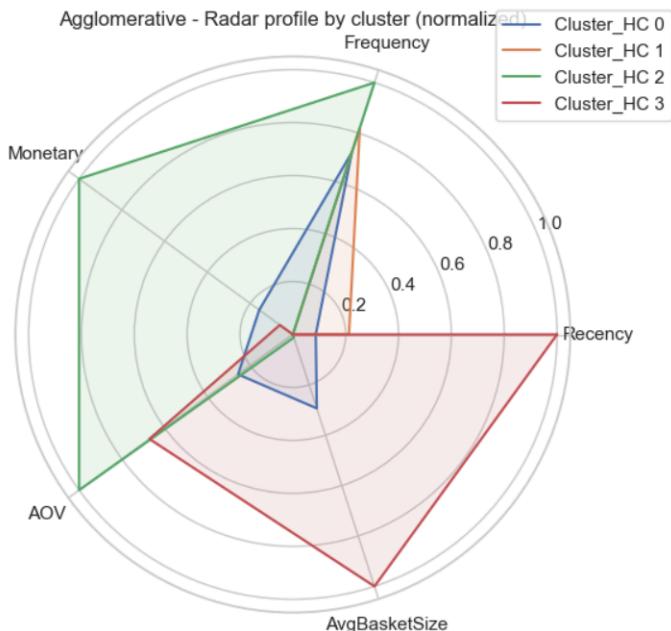
#### IV.3.2.1. Xác định số cụm (K)



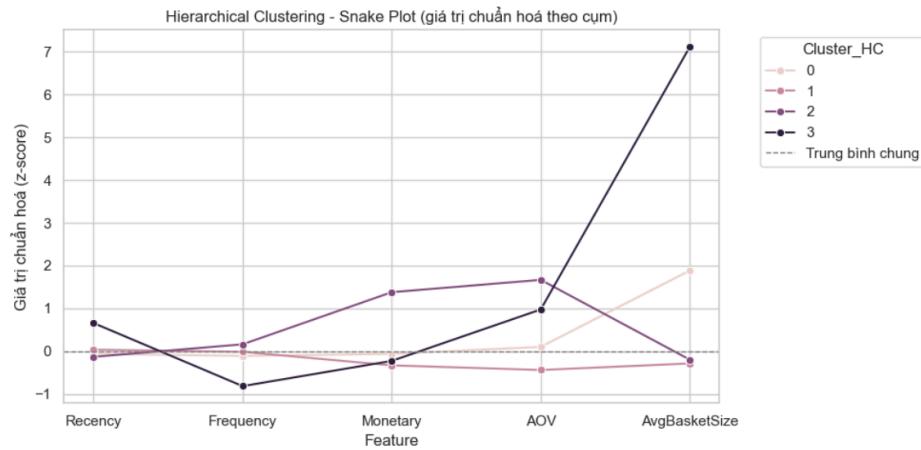
Hình 562. Biểu đồ số cụm

- **Nhận xét:** Biểu đồ cho thấy tại K=4 cho ngưỡng điểm Silhouette cao nhất và gần bằng 1

#### IV.3.2.2. Phân tích đặc điểm các cụm (Profiling)



Hình 563. Biểu đồ phân tích đặc điểm cụm thứ 1



Hình 564. Biểu đồ phân tích đặc điểm cụm thứ 2

	Recency	Frequency	Monetary	AOV	AvgBasketSize
Cluster_HC					
0	204.95	4.49	500.65	129.39	1.58
1	224.23	4.93	203.04	42.56	1.03
2	191.63	5.72	2094.66	381.36	1.05
3	345.04	1.31	318.30	269.94	2.92

Hình 565. Biểu đồ phân tích đặc điểm cụm thứ 3

#### Mô tả chi tiết:

- **Cluster HC-3: Khách hàng Trụ Cột (The Pillars/VIP)**
  - Thẻ mạnh: Đây là nhóm có chỉ số Monetary (Doanh thu) và Frequency (Tần suất) vượt trội hoàn toàn so với các nhóm còn lại. Recency rất thấp (mua hàng gần đây).
  - Hành vi: Đây là những khách hàng trung thành nhất, có sức mua lớn nhất và thường xuyên quay lại. Nhóm này chiếm tỷ trọng nhỏ về số lượng nhưng đóng góp phần lớn doanh thu (Revenue Share cao nhất). Họ là “xương sống” của doanh nghiệp.
- **Cluster HC-0: Khách hàng Tiềm Năng (The Incubator/Potential)**
  - Thẻ mạnh: Các chỉ số RFM và AOV dao động quanh mức trung bình (Z-score xấp xỉ 0). Không có chỉ số nào quá tệ nhưng cũng chưa có sự bứt phá.

- o Hành vi: Nhóm này mua sắm ở mức độ vừa phải, ổn định. Họ là nhóm “an toàn” và là nguồn dự trữ để chuyển đổi lên thành khách hàng VIP nếu có tác động marketing phù hợp.

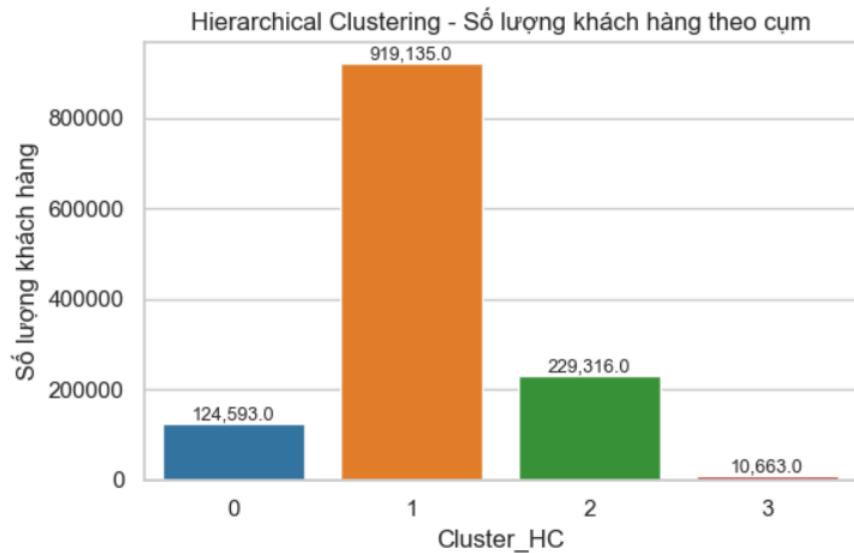
**•Cluster HC-1: Khách hàng Phổ Thông/Giá Trị Thấp (The Volume Drivers/Mass)**

- o Thé mạnh: Chiếm số lượng đông đảo nhất trong tập dữ liệu. Tần suất mua hàng (Frequency) khá tốt (cao thứ 2, chỉ sau VIP).
- o Hạn chế: Giá trị đơn hàng (AOV) và Kích thước giỏ hàng (Basket Size) thấp nhất.
- o Hành vi: Mua lặt nhặt, thường mua các sản phẩm giá trị thấp hoặc chỉ mua khi có nhu cầu thiết yếu/khuyến mãi. Dù doanh thu trên mỗi đầu người thấp, nhưng họ giúp duy trì lượng traffic và vận hành của hệ thống.

**•Cluster HC-2: Khách hàng Cá Biệt/Ngủ Đông (The Anomalies/One-off Big Spenders)**

- o Thé mạnh: Chỉ số AvgBasketSize và AOV cao đột biến (cao nhất trong tất cả các cụm, thậm chí cao hơn VIP).
- o Hạn chế: Recency rất cao (đã rất lâu không quay lại) và Frequency cực thấp (thường chỉ mua 1-2 lần).
- o Hành vi: Có hành vi mua sắm khác thường: Mua một đơn hàng giá trị cực lớn (có thể là mua sỉ, mua dự án hoặc mua thiết bị đắt tiền) nhưng sau đó không quay lại nữa. Nguy cơ rời bỏ (Churn) của nhóm này là cao nhất.

#### IV.3.2.3. Quy mô các nhóm



Hình 566. Biểu đồ quy mô các nhóm

- **Nhận xét:** Tương tự K-Means, ta vẫn thấy sự tập trung doanh thu vào một nhóm nhỏ khách hàng VIP (nguyên lý Pareto).

## **THU MỤC DỰ ÁN**

Link Drive thư mục dự án: [Google Drive](#)