

**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 MÔN HỌC
KHO DỮ LIỆU VÀ OLAP**

ĐỀ TÀI:

**XÂY DỰNG KHO DỮ LIỆU VÀ PHÂN TÍCH TRỰC TUYẾN
HOẠT ĐỘNG ĐẶT PHÒNG KHÁCH SẠN**

GVHD: ThS. Nguyễn Thị Kim Phụng

Nhóm sinh viên thực hiện:

- | | |
|----------------------|----------------|
| 1. Nguyễn Hoàng Minh | MSSV: 20521609 |
| 2. Tạ Nhật Minh | MSSV: 20521614 |

TP. HỒ CHÍ MINH, NĂM 2023

NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

....., ngày tháng năm 2023

Người nhận xét

(Ký tên và ghi rõ họ tên)

BẢNG PHÂN CÔNG VÀ ĐÁNH GIÁ THÀNH VIÊN

Table 0. Bảng phân công và đánh giá thành viên

Họ và tên	MSSV	Phân công	Đánh giá
Nguyễn Hoàng Minh	20521609	<ul style="list-style-type: none"> - Trình bày báo cáo <p>Chương 1:</p> <ul style="list-style-type: none"> - Tìm nguồn dữ liệu. - Xây dựng kho dữ liệu. - Viết báo cáo Mục 5. <p>Chương 2:</p> <ul style="list-style-type: none"> - Xây dựng phương pháp đồ dữ liệu. <p>Chương 3:</p> <ul style="list-style-type: none"> - Soạn 15 câu hỏi phân tích OLAP. - SSAS 5 câu cuối. <p>Chương 4:</p> <ul style="list-style-type: none"> - Tiền xử lý dữ liệu huấn luyện mô hình. - Xây dựng và đánh giá các mô hình khai phá dữ liệu. - Gom nhóm và trích xuất đặc điểm của từng nhóm khách hàng. - Viết báo cáo. 	100%
Tạ Nhật Minh	20521614	<p>Chương 1:</p> <ul style="list-style-type: none"> - Tìm hiểu dữ liệu. - Thống kê khám phá dữ liệu. - Tiền xử lý dữ liệu. - Viết báo cáo Mục 1, 2, 3, 4. <p>Chương 2:</p> <ul style="list-style-type: none"> - Thực hiện đồ dữ liệu vào kho. - Quay video Demo đồ dữ liệu. - Viết báo cáo. <p>Chương 3:</p> <ul style="list-style-type: none"> - Tạo Cubes và các Dimensions để phân tích dữ liệu. - SSAS 10 câu đầu. - Vẽ biểu đồ 15 câu truy vấn bằng Power BI và Excel. - Viết báo cáo. <p>Chương 4:</p> <ul style="list-style-type: none"> - Hỗ trợ 	100%

MỤC LỤC

BẢNG PHÂN CÔNG VÀ ĐÁNH GIÁ THÀNH VIÊN.....	3
MỤC LỤC.....	1
DANH MỤC HÌNH ẢNH	5
DANH MỤC BẢNG	16
DANH MỤC TỪ VIẾT TẮT	17
CHƯƠNG 1. GIỚI THIỆU TỔNG QUAN VỀ DỮ LIỆU.....	18
1.1. Giới thiệu nguồn dữ liệu	18
1.2. Danh sách thuộc tính được phân tích.....	18
1.3. Mô tả chi tiết thuộc tính	21
1.4. Khảo sát và tiền xử lý dữ liệu.....	23
1.4.1. Khảo sát dữ liệu	23
1.4.2. Các phương pháp tiền xử lý	25
1.4.3. Kết quả tiền xử lý dữ liệu	26
1.5. Xây dựng kho dữ liệu	26
1.5.1. Khái niệm Dimensional Modeling	26
1.5.2. Xây dựng sơ đồ bông tuyết.....	27
1.5.2.1. Khái niệm lược đồ hình bông tuyết (Snowflake schema).....	27
1.5.2.2. Sơ đồ bông tuyết minh họa	28
1.5.3. Mô tả chi tiết các bảng dữ liệu	28
1.5.3.1. Bảng Fact	28
1.5.3.2. Bảng phụ bậc 1	30
1.5.3.3. Bảng phụ bậc 2	32
CHƯƠNG 2. TÍCH HỢP DỮ LIỆU VÀO KHO (SSIS)	34
2.1. Khái niệm.....	34
2.2. Chuẩn bị công cụ và Data Warehouse	34
2.3. Tạo project SSIS trong Visual Studio 2022	36
2.4. Đổ dữ liệu gốc vào kho	40
2.4.1. Các khái niệm	40
2.4.2. Tiến hành đổ dữ liệu từ nguồn	42
2.5. Tạo các bảng Dim và Bảng Fact.....	46
Các khái niệm	46

2.5.1. <i>Bảng Dim_Hotel_Type</i>	47
2.5.2. <i>Bảng Dim_Reservation_Status</i>	56
2.5.3. <i>Bảng Dim_Deposit_Type</i>	64
2.5.4. <i>Bảng Dim_Distribution_Channel</i>	71
2.5.5. <i>Bảng Dim_Market_Segment</i>	77
2.5.6. <i>Bảng Dim_Customer_Type</i>	84
2.5.7. <i>Bảng Dim_Country</i>	90
2.5.8. <i>Bảng Dim_Customer</i>	96
2.5.9. <i>Bảng Dim_Year</i>	105
2.5.10. <i>Bảng Dim_Quarter</i>	112
2.5.11. <i>Bảng Dim_Month</i>	119
2.5.12. <i>Bảng Dim_Day</i>	124
2.5.13. <i>Bảng Dim_Arrival_Time</i>	131
2.5.14. <i>Bảng Dim_Reservation_Time</i>	142
2.5.15. <i>Bảng Fact</i>	153
2.6. Tạo Execute SQL Task	161
CHƯƠNG 3. PHÂN TÍCH DỮ LIỆU VÀ BÁO CÁO	166
3.1. Nội dung 15 câu truy vấn	166
3.1.1. Báo cáo tình hình kinh doanh	166
3.1.2. Báo cáo và phân tích dữ liệu khách hàng	166
3.2. Phân tích dữ liệu trong kho dữ liệu (SSAS)	167
3.2.1. Tạo project SSAS trong Visual Studio	167
3.2.2. Xác định dữ liệu nguồn (Data sources)	169
3.2.3. Xác định khung nhìn dữ liệu nguồn (Data Source View)	171
3.2.4. Xác định cube và tạo Measures (Cubes)	173
3.2.5. Xác định các thuộc tính của bảng Dimensions	176
3.2.5.1. Định nghĩa các thuộc tính (Attributes) của bảng Dimension ..	176
Bảng Dim_Reservation_Status	176
Bảng Dim_Distribution_Channel	177
Bảng Dim_Hotel_Type	178
Bảng Dim_Customer, Dim_Customer_Type và Dim_Country	179
Bảng Dim_Reservation_Time, Dim_Year, Dim_Quarter, Dim_Month và Dim_Day	180
Bảng Dim_Market_Segment	181
Bảng Dim_Deposit_Type	182

Bảng Dim_Arrival_Time, Dim_Year, Dim_Quarter, Dim_Month và Dim_Day	183
3.2.5.2. Ân các dòng giữ liệu không có giá trị (Unknown).....	184
3.2.6. Deploy và process project SSAS	185
3.2.7. Định nghĩa Named Set	188
3.2.8. Phân tích dữ liệu trên các khối (cube) bằng công cụ SSAS, ngôn ngữ MDX, Power BI và Excel	190
3.2.8.1. Báo cáo tình hình kinh doanh.....	190
1. Thống kê tổng số lượng khách hàng đặt phòng, tổng doanh thu của từng loại khách sạn theo từng quý của năm.	190
2. Thống kê doanh thu theo phân khúc khách hàng (Market_segment) theo từng tháng của năm.....	193
3. Thống kê doanh thu theo kênh phân phối theo từng tháng của năm.197	
4. Thống kê top 10 quốc gia có doanh thu cao nhất trong năm 2015 và 2016, sắp xếp các giá trị theo thứ tự giảm dần.....	201
5. Thống kê tổng thời gian chờ của khách hàng (lead_time) theo từng Quốc gia theo từng tháng của năm.	207
6. Thống kê các quốc gia tạo ra tổng doanh thu lớn hơn 50000.....	211
7. Thống kê tổng thời gian chờ của từng phân khúc thị trường theo từng tháng, quý, năm.	214
8. Thống kê số lượng đặt phòng của khách hàng đến từ Bồ Đào Nha (PRT) thuộc từng phân khúc thị trường theo từng quý của từng năm.	216
9. Thống kê doanh thu theo từng kênh phân phối của từng phân khúc thị trường của từng quốc gia.	218
3.2.8.2. Báo cáo và phân tích dữ liệu khách hàng	220
10. Thống kê có bao nhiêu khách hàng là người lớn, bao nhiêu khách hàng là trẻ em và bao nhiêu khách hàng là em bé trong từng tháng của năm... 220	
11. Thống kê thông tin top 50 khách hàng có chi tiêu cao nhất trong năm trước (2016).....	223
12. Thống kê tên top 5 khách hàng có chi tiêu cao nhất theo từng tháng của năm 2016.....	228
13. Thống kê email top 3 khách hàng có doanh số cao nhất theo quốc gia năm 2016.....	232
14. Thống kê top 10 thông tin khách hàng kèm chi tiêu của họ có thời gian chờ lâu nhất theo từng quốc gia năm 2016.	236
15. Thống kê thông tin và doanh thu khách hàng đặt phòng với hình thức không đặt cọc trước và thuộc phân khúc thị trường đặt phòng trực tiếp..	241
CHƯƠNG 4. KHAI PHÁ DỮ LIỆU	243
4.1. Khởi tạo môi trường khai phá dữ liệu	243
4.2. Khảo sát dữ liệu	244
4.3. Tiền xử lý dữ liệu	246

4.3.1. Xử lý dữ liệu biến rời rạc	246
4.3.2. Xử lý dữ liệu biến liên tục	248
4.4. Lựa chọn các thuộc tính ảnh hưởng đến việc hủy phòng của khách hàng	252
4.5. Tiền xử lý dữ liệu huấn luyện mô hình dự đoán hủy đặt phòng	253
 4.5.1. Chuẩn hóa dữ liệu	253
 4.5.2. Chia dữ liệu huấn luyện	254
4.6. Huấn luyện mô hình	254
 4.6.1. Mô hình machine learning	254
1. Logistic regression	254
2. Decision tree	254
3. Support vector machine	254
4. Random forest	255
5. XGBoost	255
 4.6.2. Mô hình deep learning	255
6. Artificial neural network (ANN)	255
4.7. Đánh giá và so sánh các mô hình	256
4.8. Phân tích các yếu tố ảnh hưởng đến việc hủy đặt phòng và rút ra luật	261
4.9. Gom nhóm khách hàng và đặc trưng của từng nhóm	263
 4.9.1. Lựa chọn số nhóm khách hàng cần gom nhóm	263
 4.9.2. Gom nhóm khách hàng bằng thuật toán K-means và trích xuất đặc trưng của từng nhóm khách hàng	264
DANH MỤC TÀI LIỆU THAM KHẢO	268
PHỤ LỤC	269
Phụ lục 1: Bộ mã lệnh thống kê dữ liệu	269
Phụ lục 2: Bộ mã lệnh tiền xử lý dữ liệu	270
Phụ lục 3: Bộ mã lệnh SQL tạo khóa cho bảng	271

DANH MỤC HÌNH ẢNH

Figure 1. Năm dòng dữ liệu đầu tiên của bộ dữ liệu.....	23
Figure 2. Biểu đồ số lượng đặt phòng và hủy phòng.....	24
Figure 3. Đồ thị thể hiện mối quan hệ giữa dữ liệu bị hủy đặt phòng trong hai loại hình khách sạn trong khoảng thời gian từ khi đặt phòng tới khi nhận được phòng.....	24
Figure 4. Biểu đồ sự tương quan giữa việc thuê phòng vào cuối tuần với hủy đặt phòng tại khách sạn	24
Figure 5. Thống kê số lượng giá trị bị khuyết của từng thuộc tính	25
Figure 6. Kết quả sau khi tiền xử lý dữ liệu	26
Figure 7. Sơ đồ bông tuyết tập dữ liệu hoạt động đặt phòng khách sạn	28
Figure 8. Mở SQL Server và kết nối với Server.....	34
Figure 9. Tạo kho dữ liệu	35
Figure 10. Đặt tên kho dữ liệu	35
Figure 11. Kho dữ liệu đã được tạo.....	35
Figure 12. Tạo Project với Visual Studio 2022	36
Figure 13. Giao diện chọn công cụ SSIS Visual Studio 2022	36
Figure 14. Đặt tên cho Project trong Visual Studio 2022	37
Figure 15. Giao diện làm việc với Project SSIS trong Visual Studio 2022	37
Figure 16. Tạo Connection với Database	38
Figure 17. Thêm kết nối OLEDB.....	38
Figure 18. Thiết lập kết nối	39
Figure 19. Nhập Server name và Database name	39
Figure 20. Hoàn thành kết nối với cơ sở dữ liệu	40
Figure 21. Kết nối thành công với cơ sở dữ liệu	40
Figure 22. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Hotel Booking	42
Figure 23. Nạp dữ liệu	42
Figure 24. Chọn New để khởi tạo nguồn dữ liệu đầu vào	42
Figure 25. Chọn bộ dữ liệu đầu vào	43
Figure 26. Xem trước dữ liệu đầu vào.....	43
Figure 27. Đưa dữ liệu vào Database	44
Figure 28. Đỗ dữ liệu thành công	46
Figure 29. Kiểm tra bảng Hotel_Booking trong SQL Server	46

Figure 30. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Hotel_Type	47
Figure 31. Kéo chức năng OLE DB Source vào màn hình làm việc	47
Figure 32. Chọn Table sẽ lấy dữ liệu.....	48
Figure 33. Chọn các column cần sử dụng	48
Figure 34. Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau	49
Figure 35. Chọn các thuộc tính cần gom nhóm	49
Figure 36. Sử dụng OLE DB Destination để tạo bảng.....	53
Figure 37. Quá trình Mappings dữ liệu	55
Figure 38. Hoàn thành đổ dữ liệu vào HotelType trong kho dữ liệu	55
Figure 39. Kiểm tra bảng Dim_Hotel_Type trong SQL Server.....	56
Figure 40. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Reservation_Status	56
Figure 41. Kéo chức năng OLE DB Source vào màn hình làm việc	56
Figure 42. Chọn Table sẽ lấy dữ liệu.....	57
Figure 43. Chọn các column cần sử dụng	57
Figure 44. Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau	58
Figure 45. Chọn các thuộc tính cần gom nhóm	58
Figure 46. Sử dụng OLE DB Destination để tạo bảng.....	62
Figure 47. Tạo bảng Dim_Reservation_Status.....	62
Figure 48. Quá trình Mappings dữ liệu	63
Figure 49. Hoàn thành đổ dữ liệu vào Dim_Reservation_Status trong kho dữ liệu	63
Figure 50. Kiểm tra bảng Dim_Reservation_Status trong SQL Server.....	64
Figure 51. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Deposit_Type	64
Figure 52. Kéo chức năng OLE DB Source vào màn hình làm việc	64
Figure 53. Chọn Table sẽ lấy dữ liệu.....	65
Figure 54. Chọn các column cần sử dụng	65
Figure 55. Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau	66
Figure 56. Chọn các thuộc tính cần gom nhóm	66
Figure 57. Sử dụng OLE DB Destination để tạo bảng.....	69
Figure 58. Tạo bảng Dim_Deposit_Type.....	69
Figure 59. Quá trình Mappings dữ liệu	70
Figure 60. Hoàn thành đổ dữ liệu vào Dim_Deposit_Type trong kho dữ liệu	70
Figure 61. Kiểm tra bảng Dim_Deposit_Type trong SQL Server	71

Figure 62. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Distribution_Channel	71
Figure 63. Kéo chức năng OLE DB Source vào màn hình làm việc	71
Figure 64. Chọn Table sẽ lấy dữ liệu.....	72
Figure 65. Chọn các column cần sử dụng	72
Figure 66. Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau	73
Figure 67. Chọn các thuộc tính cần gom nhóm	73
Figure 68. Sử dụng OLE DB Destination để tạo bảng.....	75
Figure 69. Tạo bảng Dim_Distribution_Channel.....	76
Figure 70. Quá trình Mappings dữ liệu	76
Figure 71. Hoàn thành đổ dữ liệu vào Dim_Distribution_Channel trong kho dữ liệu ...	76
Figure 72. Kiểm tra bảng Dim_Distribution_Channel trong SQL Server.....	77
Figure 73. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Market_Segment.....	77
Figure 74. Kéo chức năng OLE DB Source vào màn hình làm việc	77
Figure 75. Chọn Table sẽ lấy dữ liệu.....	78
Figure 76. Chọn các column cần sử dụng	78
Figure 77. Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau	79
Figure 78. Chọn các thuộc tính cần gom nhóm	79
Figure 79. Sử dụng OLE DB Destination để tạo bảng	81
Figure 80. Tạo bảng Dim_Market_Segment.....	82
Figure 81. Quá trình Mappings dữ liệu	82
Figure 82. Hoàn thành đổ dữ liệu vào Dim_Market_Segment trong kho dữ liệu	83
Figure 83. Kiểm tra bảng Dim_Market_Segment trong SQL Server.....	83
Figure 84. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Customer_Type	84
Figure 85. Kéo chức năng OLE DB Source vào màn hình làm việc	84
Figure 86. Chọn Table sẽ lấy dữ liệu.....	84
Figure 87. Chọn các column cần sử dụng	85
Figure 88. Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau	85
Figure 89. Chọn các thuộc tính cần gom nhóm	86
Figure 90. Sử dụng OLE DB Destination để tạo bảng	88
Figure 91. Tạo bảng Dim_Customer_Type.....	88
Figure 92. Quá trình Mappings dữ liệu	88
Figure 93. Hoàn thành đổ dữ liệu vào Dim_Customer_Type trong kho dữ liệu.....	89

Figure 94. Kiểm tra bảng Dim_Customer_Type trong SQL Server.....	89
Figure 95. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Country	90
Figure 96. Kéo chức năng OLE DB Source vào màn hình làm việc	90
Figure 97. Chọn Table sẽ lấy dữ liệu.....	90
Figure 98. Chọn các column cần sử dụng	91
Figure 99. Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau	91
Figure 100. Chọn các thuộc tính cần gom nhóm	92
Figure 101. Sử dụng OLE DB Destination để tạo bảng	94
Figure 102. Tạo bảng Dim_Country	95
Figure 103. Quá trình Mappings dữ liệu	95
Figure 104. Hoàn thành đổ dữ liệu vào Dim_Country trong kho dữ liệu.....	96
Figure 105. Kiểm tra bảng Dim_Country trong SQL Server	96
Figure 106. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Customer	97
Figure 107. Kéo chức năng OLE DB Source vào màn hình làm việc	97
Figure 108. Chọn Table sẽ lấy dữ liệu.....	97
Figure 109. Chọn các column cần sử dụng	98
Figure 110. Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau	98
Figure 111. Chọn các thuộc tính cần gom nhóm	99
Figure 112. Sử dụng OLE DB Destination để tạo bảng	103
Figure 113. Tạo bảng Dim_Customer	103
Figure 114. Quá trình Mappings dữ liệu	103
Figure 115. Hoàn thành đổ dữ liệu vào Dim_Customer trong kho dữ liệu.....	104
Figure 116. Kiểm tra bảng Dim_Customer trong SQL Server.....	104
Figure 117. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Year	105
Figure 118. Kéo chức năng OLE DB Source vào màn hình làm việc	105
Figure 119. Chọn Table sẽ lấy dữ liệu.....	105
Figure 120. Chọn các column cần sử dụng	106
Figure 121. Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau	106
Figure 122. Chọn các thuộc tính cần gom nhóm	107
Figure 123. Sử dụng OLE DB Destination để tạo bảng	110
Figure 124. Tạo bảng Dim_Year	110

Figure 125. Quá trình Mappings dữ liệu	111
Figure 126. Hoàn thành đổ dữ liệu vào Dim_Year trong kho dữ liệu.....	111
Figure 127. Kiểm tra bảng Dim_Year trong SQL Server	112
Figure 128. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Quarter	112
Figure 129. Kéo chức năng OLE DB Source vào màn hình làm việc	112
Figure 130. Chọn Table sẽ lấy dữ liệu.....	113
Figure 131. Chọn các column cần sử dụng	113
Figure 132. Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau	114
Figure 133. Chọn các thuộc tính cần gom nhóm	114
Figure 134. Sử dụng OLE DB Destination để tạo bảng	116
Figure 135. Tạo bảng Dim_Quater	117
Figure 136. Quá trình Mappings dữ liệu	117
Figure 137. Hoàn thành đổ dữ liệu vào Dim_Quarter trong kho dữ liệu	118
Figure 138. Kiểm tra bảng Dim_Quarter trong SQL Server	118
Figure 139. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Month	119
Figure 140. Kéo chức năng OLE DB Source vào màn hình làm việc	119
Figure 141. Chọn Table sẽ lấy dữ liệu.....	119
Figure 142. Chọn các column cần sử dụng	120
Figure 143. Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau	120
Figure 144. Chọn các thuộc tính cần gom nhóm	121
Figure 145. Sử dụng OLE DB Destination để tạo bảng	122
Figure 146. Tạo bảng Dim_Month	122
Figure 147. Quá trình Mappings dữ liệu	123
Figure 148. Hoàn thành đổ dữ liệu vào Dim_Month trong kho dữ liệu	123
Figure 149. Kiểm tra bảng Dim_Month trong SQL Server	124
Figure 150. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Day	124
Figure 151. Kéo chức năng OLE DB Source vào màn hình làm việc	124
Figure 152. Chọn Table sẽ lấy dữ liệu.....	125
Figure 153. Chọn các column cần sử dụng	125
Figure 154. Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau	126

Figure 155. Chọn các thuộc tính cần gom nhóm	126
Figure 156. Sử dụng OLE DB Destination để tạo bảng	129
Figure 157. Tạo bảng Dim_Day.....	129
Figure 158. Quá trình Mappings dữ liệu	130
Figure 159. Hoàn thành đổ dữ liệu vào Dim_Day trong kho dữ liệu	130
Figure 160. Kiểm tra bảng Dim_Day trong SQL Server.....	131
Figure 161. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Arrival_Time	131
Figure 162. Kéo chức năng OLE DB Source vào màn hình làm việc	132
Figure 163. Chọn Table sẽ lấy dữ liệu.....	132
Figure 164. Chọn các column cần sử dụng	132
Figure 165. Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau	133
Figure 166. Quá trình Mappings dữ liệu	141
Figure 167. Hoàn thành đổ dữ liệu vào Dim_Arrival_Time trong kho dữ liệu	141
Figure 168. Kiểm tra bảng Dim_Arrival_Time trong SQL Server	142
Figure 169. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Reservation_Time	142
Figure 170. Kéo chức năng OLE DB Source vào màn hình làm việc	143
Figure 171. Chọn Table sẽ lấy dữ liệu.....	143
Figure 172. Chọn các column cần sử dụng	143
Figure 173. Quá trình Mappings dữ liệu	152
Figure 174. Hoàn thành đổ dữ liệu vào Dim_Reservation_Time trong kho dữ liệu ...	152
Figure 175. Kiểm tra bảng Dim_Reservation_Time trong SQL Server	153
Figure 176. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Fact	153
Figure 177. Kéo chức năng OLE DB Source vào màn hình làm việc	153
Figure 178. Chọn Table sẽ lấy dữ liệu.....	154
Figure 179. Chọn các column cần sử dụng	154
Figure 180. Quá trình Mappings dữ liệu	160
Figure 181. Hoàn thành đổ dữ liệu vào Fact trong kho dữ liệu	161
Figure 182. Kiểm tra bảng Fact trong SQL Server	161
Figure 183. Các bảng Dimensions trong Sequence Container.....	164
Figure 184. Các thành phần được kết nối với nhau theo thứ tự.....	165
Figure 185. Hoàn thành đổ dữ liệu vào kho dữ liệu.....	165
Figure 186. Giao diện khởi động Visual Studio 2022	167

Figure 187. Tìm kiếm project template SSAS của Visual Studio	167
Figure 188. Giao diện cấu hình file của SSAS trong Visual Studio.....	168
Figure 189. Giao diện công cụ BI quá trình SSAS.....	168
Figure 190. Tạo một Data Suorce cho quá trình SSAS	169
Figure 191. Chọn kết nối với Data Source đã có sẵn hoặc tạo kết nối mới.....	169
Figure 192. Tạo kết nối mới khi project chạy lần đầu	169
Figure 193. Chọn kết nối với Data Source.....	170
Figure 194. Tùy chọn tài khoản phù hợp	170
Figure 195. Hoàn thành tạo nguồn dữ liệu	171
Figure 196. Tạo Data Source View.....	171
Figure 197. Chọn kho dữ liệu	171
Figure 198. Chọn bảng dữ liệu Fact	172
Figure 199. Chọn các bảng Dimension liên quan đến Fact	172
Figure 200. Chọn các bảng Dim có liên quan đến Dim_Customer, Dim_Arrival_Time và Dim_Reservation_time	172
Figure 201. Các bảng dữ liệu được tạo.....	173
Figure 202. Hoàn tất tạo Data Source View	173
Figure 203. Bắt đầu quá trình tạo Cube	174
Figure 204. Tùy chọn tạo Cube	174
Figure 205. Chọn Fact làm Measure Group cho Cube	174
Figure 206. Tùy chọn các thuộc tính định lượng	175
Figure 207. Chọn các bảng Dimension cho Cube	175
Figure 208. Xem lại Measure Groups và các Dimension.....	175
Figure 209. Hoàn tất quá trình tạo Cube	176
Figure 210. Bảng thuộc tính Dim_Reservation_Status.....	176
Figure 211. Kết quả sau khi kéo thả các thuộc tính.....	177
Figure 212. Bảng thuộc tính Dim_Distribution_Channel.....	177
Figure 213. Kết quả sau khi kéo thả các thuộc tính.....	178
Figure 214. Bảng thuộc tính Dim_Hotel_Type.....	178
Figure 215. Kết quả sau khi kéo thả các thuộc tính.....	179
Figure 216. Bảng thuộc tính Dim_Customer, Dim_Customer_Type và Dim_Country	179
Figure 217. Kết quả sau khi kéo thả các thuộc tính.....	180
Figure 218. Bảng thuộc tính Dim_Reservation_Time, Dim_Year, Dim_Quarter, Dim_Month và Dim_Day.....	180
Figure 219. Kết quả sau khi kéo thả các thuộc tính.....	181

Figure 220. Bảng thuộc tính Dim_Market_Segment.....	181
Figure 221. Kết quả sau khi kéo thả các thuộc tính.....	182
Figure 222. Bảng thuộc tính Dim_Deposit_Type	182
Figure 223. Kết quả sau khi kéo thả các thuộc tính.....	183
Figure 224. Bảng thuộc tính Dim_Arrival_Time, Dim_Year, Dim_Quarter, Dim_Month và Dim_Day.....	183
Figure 225. Kết quả sau khi kéo thả các thuộc tính.....	184
Figure 226. Các bảng Dim của cube	184
Figure 227. Kết quả sau khi chỉnh sửa UnknownMember	185
Figure 228. Bắt đầu Deploy	185
Figure 229. Deploy thành công.....	186
Figure 230. Thực hiện process Cube	186
Figure 231. Chọn Yes để tiếp tục	187
Figure 232. Chọn Run để tiến hành process Cube.....	187
Figure 233. Hoàn tất quá trình process Cube.....	188
Figure 234. Tạo Named Set mới	188
Figure 235. Giao diện của 1 Named Set	189
Figure 236. Hàm Filter để lọc dữ liệu trong Named Set	189
Figure 237. Kết quả SSAS câu 1	190
Figure 238. Kết quả MDX câu 1	191
Figure 239. Kết quả Power BI câu 1	191
Figure 240. Kết quả trực quan Power BI câu 1.....	192
Figure 241. Kết quả Pivot Excel câu 1.....	192
Figure 242. Kết quả Excel Pivot Chart câu 1	193
Figure 243. Kết quả SSAS câu 2	194
Figure 244. Kết quả MDX câu 2	194
Figure 245. Kết quả Power BI câu 2.....	195
Figure 246. Kết quả trực quan Power BI câu 2.....	195
Figure 247. Kết quả Pivot Excel câu 2.....	196
Figure 248. Kết quả Excel Pivot Chart câu 2	196
Figure 249. Kết quả SSAS câu 3.....	197
Figure 250. Kết quả MDX câu 3	198
Figure 251. Kết quả Power BI câu 3.....	198
Figure 252. Kết quả trực quan Power BI câu 3.....	199
Figure 253. Kết quả Pivot Excel câu 3.....	200

Figure 254. Kết quả Excel Pivot Chart câu 3	200
Figure 255. Kéo thả các trường cần dùng để truy vấn	201
Figure 256. Câu truy vấn trong Script Mode sau chỉnh sửa	202
Figure 257. Kết quả SSAS câu 4 năm 2015	202
Figure 258. Kết quả SSAS câu 4 năm 2016	202
Figure 259. Kết quả MDX câu 4 năm 2015	203
Figure 260. Kết quả MDX câu 4 năm 2016	203
Figure 261. Kết quả Power BI câu 4 năm 2015	204
Figure 262. Kết quả Power BI câu 4 năm 2016	204
Figure 263. Kết quả trực quan Power BI câu 4	205
Figure 264. Kết quả Pivot Excel câu 4 năm 2015	206
Figure 265. Kết quả Pivot Excel câu 4 năm 2016	206
Figure 266. Kết quả Excel Pivot Chart câu 4	207
Figure 267. Kết quả SSAS câu 5	208
Figure 268. Kết quả MDX câu 5	208
Figure 269. Kết quả Power BI câu 5	209
Figure 270. Kết quả trực quan Power BI câu 5	210
Figure 271. Kết quả Pivot Excel câu 5	210
Figure 272. Kết quả Excel Pivot Chart câu 5	211
Figure 273. Tạo Named Set câu 6	211
Figure 274. Kéo thả các trường cần dùng để truy vấn	212
Figure 275. Câu truy vấn trong Script Mode sau chỉnh sửa	212
Figure 276. Kết quả SSAS câu 6	212
Figure 277. Kết quả MDX câu 6	213
Figure 278. Kết quả Power BI câu 6	213
Figure 279. Kéo thả các trường cần dùng để truy vấn	214
Figure 280. Kết quả MDX câu 7	215
Figure 281. Kết quả Power BI câu 7	215
Figure 282. Kéo thả các trường cần dùng để truy vấn	216
Figure 283. Kết quả MDX câu 8	217
Figure 284. Kết quả Power BI câu 8	217
Figure 285. Kéo thả các trường cần dùng để truy vấn	218
Figure 286. Kết quả MDX câu 9	219
Figure 287. Kết quả Power BI câu 9	219
Figure 288. Kết quả SSAS câu 10	220

Figure 289. Kết quả MDX câu 10	221
Figure 290. Kết quả Pivot Excel câu 10.....	222
Figure 291. Kết quả trực quan Power BI câu 10.....	222
Figure 292. Kết quả Excel Pivot Chart câu 10.....	223
Figure 293. Kéo thả các trường cần dùng để truy vấn	224
Figure 294. Câu truy vấn trong Script Mode sau chỉnh sửa	224
Figure 295. Kết quả SSAS câu 11.....	224
Figure 296. Kết quả MDX câu 11	225
Figure 297. Kết quả Pivot Excel câu 11.....	226
Figure 298. Kết quả trực quan Power BI câu 11.....	227
Figure 299. Kết quả Excel Pivot Chart câu 11	227
Figure 300. Kéo thả các trường cần dùng để truy vấn	228
Figure 301. Câu truy vấn trong Script Mode sau chỉnh sửa	228
Figure 302. Kết quả SSAS câu 12.....	229
Figure 303. Kết quả MDX câu 12	230
Figure 304. Kết quả Pivot Excel câu 12 (1 tháng)	231
Figure 305. Kết quả trực quan Power BI câu 12.....	231
Figure 306. Kết quả Excel Pivot Chart câu 12	232
Figure 307. Tạo Named Set cho câu 13	232
Figure 308. Kéo thả các trường cần dùng để truy vấn	233
Figure 309. Câu truy vấn trong Script Mode sau chỉnh sửa	233
Figure 310. Kết quả SSAS câu 13.....	233
Figure 311. Kết quả MDX câu 13	234
Figure 312. Kết quả Power BI câu 13 (1 quốc gia).....	235
Figure 313. Kết quả trực quan Power BI câu 13.....	235
Figure 314. Kết quả Excel Pivot Chart câu 13	236
Figure 315. Tạo Named Set [Cau 10] cho câu 14	237
Figure 316. Kéo thả các trường cần dùng để truy vấn	237
Figure 317. Câu truy vấn trong Script Mode sau chỉnh sửa	237
Figure 318. Kết quả SSAS câu 14.....	238
Figure 319. Kết quả MDX câu 14	238
Figure 320. Kết quả Power BI câu 14 (năm 2016).....	239
Figure 321. Kết quả trực quan Power BI câu 14.....	240
Figure 322. Kết quả Excel Pivot Chart câu 14	240
Figure 323. Kéo thả các trường cần dùng để truy vấn	241

Figure 324. Kết quả MDX câu 15	242
Figure 325. Kết quả Power BI câu 15	242
Figure 326. Giao diện môi trường khai phá dữ liệu với Jupiter Notebook	243
Figure 327. Các thư viện sử dụng cho quá trình khai phá.....	243
Figure 328. Thông tin 5 dòng dữ liệu bất kỳ trong bộ dữ liệu	244
Figure 329. Thông tin các thuộc tính dùng để khai phá dữ liệu	245
Figure 330. Biểu đồ số lượng đặt phòng và hủy phòng.....	246
Figure 331. Biểu đồ phân phối của các nhãn category của bộ dữ liệu	248
Figure 332. Biểu đồ dữ liệu numeric trước tiền xử lý	250
Figure 333. Biểu đồ dữ liệu numeric sau tiền xử lý	251
Figure 334. Dữ liệu khi sau khi tiền xử lý (không có thuộc tính is_canceled)	251
Figure 335. Dữ liệu khi sau khi chuẩn hóa	253
Figure 336. Huấn luyện mô hình Logistic regression.....	254
Figure 337. Huấn luyện mô hình Decision tree.....	254
Figure 338. Huấn luyện mô hình Support vector machine.....	254
Figure 339. Huấn luyện mô hình Random forest	255
Figure 340. Huấn luyện mô hình XGBoost	255
Figure 341. Kiến trúc mô hình deep learning.....	256
Figure 342. Huấn luyện kiến trúc trên tập dữ liệu huấn luyện	256
Figure 343. Biểu đồ mức độ các yếu tố ảnh hưởng đến việc hủy phòng của khách hàng	262
Figure 344. Biểu đồ phân phối giữa thời gian chờ và sự hủy phòng	262
Figure 345. Biểu đồ giữa loại thanh toán và sự hủy phòng	263
Figure 346. Biểu đồ lỗi ứng với các nhóm khách hàng được phân chia	264
Figure 347. Biểu đồ trực quan 3 nhóm khách hàng theo 3 yếu tố phân biệt	267

DANH MỤC BẢNG

Table 1. Thuộc tính của tập dữ liệu	18
Table 2. Dictionary thuộc tính Hotel.....	21
Table 3. Dictionary thuộc tính Is_canceled	21
Table 4. Dictionary thuộc tính Market_segment	21
Table 5. Dictionary thuộc tính Meal	22
Table 6. Dictionary thuộc tính Distribution_channel.....	22
Table 7. Dictionary thuộc tính Is_repeated_guest	22
Table 8. Dictionary thuộc tính Deposit_type	22
Table 9. Dictionary thuộc tính Customer_type.....	23
Table 10. Dictionary thuộc tính Reservation_status	23
Table 11. Fact_Hotel_Booking	28
Table 12. Dim_Reservation_Time	30
Table 13. Dim_Arrival_Time	30
Table 14. Dim_Customer.....	31
Table 15. Dim_Hotel_Type	31
Table 16. Dim_Deposit_Type	31
Table 17. Dim_Reservation_Status	32
Table 18. Dim_Distribution_Channel	32
Table 19. Dim_Market_Segment	32
Table 20. Dim_Month	32
Table 21. Dim_Day	33
Table 22. Dim_Year.....	33
Table 23. Dim_Quarter	33
Table 24. Dim_Customer_Type	33
Table 25. Dim_Country.....	33
Table 26. Bảng kết quả dự đoán trên tập dữ liệu kiểm thử	256
Table 27. Bảng kết quả Confusion matrix (tạm dịch: ma trận nhầm lẫn) trên dữ liệu kiểm thử	257
Table 28. Bảng kết quả Classification Report trên tập dữ liệu kiểm thử.....	258
Table 29. Bảng đặc trưng của từng nhóm khách hàng	265

DANH MỤC TỪ VIẾT TẮT

STT	Từ viết tắt	Nội dung
1	EDA	Exploratory Data Analysis
2	Dim	Dimension
3	SSIS	SQL Server Integration Services
4	SSAS	SQL Server Analysis Services
5	SQL	Structured Query Language

CHƯƠNG 1. GIỚI THIỆU TỔNG QUAN VỀ DỮ LIỆU

1.1. Giới thiệu nguồn dữ liệu

Bộ dữ liệu Hotel Booking [1] là dữ liệu được lấy từ bài báo Hotel Booking Demand Datasets [2] trong tạp chí Data in Brief số 22 xuất bản vào tháng 02/2019 được viết bởi Nuno Antonio, Ana Almeida và Luis Nunes.

Bộ dữ liệu gồm các thuộc tính chứa quan sát các hoạt động đặt phòng tại các City Hotel và Resort Hotel trong khoảng thời gian từ **01/07/2015** đến **31/08/2017**.

Bộ dữ liệu bao gồm **119390** dòng dữ liệu với **36** thuộc tính, thu thập các quan sát hoạt động đặt phòng từ city hotel và resort hotel.

Link nguồn dữ liệu: <https://www.kaggle.com/datasets/mojtaba142/hotel-booking-demand>

1.2. Danh sách thuộc tính được phân tích

Table 1. Thuộc tính của tập dữ liệu

STT	Tên thuộc tính	Ý nghĩa	Kiểu dữ liệu
1	Hotel	Kiểu khách sạn.	Text
2	Is_canceled	Hủy đặt phòng.	Number
3	Lead_time	Số ngày từ ngày khách hàng đặt chỗ và được ghi nhận vào hệ thống quản lý của khách sạn (PSM) đến ngày khách hàng đến nhận phòng.	Number
4	Arrival_date_time	Năm của ngày khách hàng đến.	Number
5	Arrival_date_month	Tháng của ngày khách hàng đến gồm 12 giá trị từ tháng 1 đến tháng 12.	Text
6	Arrival_date_week_number	Số thứ tự tuần trong năm của ngày khách hàng đến với giá trị từ 1 đến 53.	Number
7	Arrival_date_of_month	Số ngày trong tháng của ngày khách hàng đến với giá trị từ 1 đến 31.	Number
8	Stays_in_weekend_nights	Số đêm cuối tuần (thứ bảy hoặc chủ nhật) mà khách có lưu trú hoặc đặt phòng tại khách sạn.	Number

9	Stays_in_week_nights	Số đêm trong tuần (từ Thứ Hai đến Thứ Sáu) mà khách đã lưu trú hoặc đặt phòng tại khách sạn (được tính bằng cách đếm số đêm trong tuần).	Number
10	Adults	Số lượng khách hàng là người lớn.	Number
11	Children	Số lượng khách hàng là trẻ em.	Number
12	Babies	Số lượng khách hàng là em bé.	Number
13	Meal	Cho biết các bữa ăn được bao gồm trong giá phòng mà khách hàng đặt.	Text
14	Country	Quốc tịch của khách hàng.	Text
15	Market_segment	Thể hiện các phân khúc khách hàng của khách sạn.	Text
16	Distribution_channel	Để tiếp cận với lượng lớn khách thuê phòng tiềm năng, các khách sạn đều liên kết bán phòng qua nhiều kênh phân phối phòng khách sạn.	Text
17	Is_repeated_guest	Cho biết lượt đặt chỗ có phải từ khách hàng đã lưu trú nhiều lần tại khách sạn hay là lần đầu đến.	Number
18	Previous_cancellations	Số lượng lượt đặt phòng đã bị khách hàng hủy trước lượt đặt phòng hiện tại.	Number
19	Previous_booking_not_canceled	Số lượng lượt nhận phòng thành công của khách hàng trước lượt đặt phòng hiện tại.	Number
20	Reserved_room_type	Mã loại phòng được đại diện bằng các chữ cái.	Text
21	Assigned_room_type	Mã loại phòng được chỉ định cho lượt đặt phòng. Đôi khi loại phòng được chỉ định khác với loại phòng đã đặt vì lý do hoạt động của khách sạn (khách sạn nhận số lượng đặt phòng nhiều hơn số lượng phòng thực tế) hoặc do yêu cầu của khách hàng.	Text
22	Booking_changes	Số lượng sửa đổi đối với lượt đặt phòng kể từ thời điểm đặt	Number

		phòng được nhập trên PMS cho đến thời điểm nhận phòng hoặc lượt đặt phòng bị hủy bỏ.	
23	Deposit_type	Thể hiện hình thức đặt cọc trước.	Text
24	Agent	Thể hiện ID của đại lý du lịch đã đặt chỗ.	Number
25	Company	ID của công ty / tổ chức đã thực hiện đặt phòng hoặc chịu trách nhiệm thanh toán đặt phòng.	Number
26	Days_in_waiting_list	Số ngày lượt đặt chỗ ở trong danh sách chờ trước khi nó được xác nhận với khách hàng.	Number
27	Customer_type	Thể hiện loại khách hàng.	Text
28	Adr	Doanh thu bình quân hằng ngày (Average Daily Rate) được tính bằng cách chia tổng của tất cả các giao dịch lưu trú cho tổng số đêm lưu trú. Số chỗ đậu xe ô tô theo yêu cầu của khách hàng.	Number
29	Required_car_parking_spaces	Số chỗ đậu xe ô tô theo yêu cầu của khách hàng.	Number
30	Total_of_special_requests	Số lượng yêu cầu đặc biệt của khách hàng (ví dụ: có giường đôi hoặc ở tầng cao).	Number
31	Reservation_status	Thể hiện tình trạng đặt phòng.	Text
32	Reservation_status_date	Ngày mà trạng thái cuối cùng của lượt đặt phòng được cập nhật. Thuộc tính này có thể được sử dụng cùng với Reservation_status để biết khi nào đặt phòng bị hủy hoặc khi nào khách hàng đã trả phòng khách sạn.	Date
33	Name	Họ tên của khách hàng.	Text
34	Email	Email của khách hàng.	Text
35	Phone_number	Số điện thoại khách hàng.	Number
36	Credit_card	Số thẻ tín dụng của khách hàng.	Number

1.3. Mô tả chi tiết thuộc tính

Table 2. Dictionary thuộc tính Hotel

Hotel		
STT	Giá trị	Ý nghĩa
1	Resort Hotel	Khách sạn nghỉ dưỡng.
2	City Hotel	Khách sạn thành phố.

Table 3. Dictionary thuộc tính Is_canceled

Is_canceled		
STT	Giá trị	Ý nghĩa
1	0	Không hủy.
2	1	Hủy.

Table 4. Dictionary thuộc tính Market_segment

Market_segment		
STT	Giá trị	Ý nghĩa
1	Offline TA/TO	Travel Agency/ Tour Operator: khách hàng từ các đại lý du lịch và công ty điều hành tour.
2	Online TA	Khách hàng từ các đại lý du lịch nhưng bằng hình thức online trực tuyến.
3	Aviation	Khách hàng đến từ các hãng hàng không.
4	Complementary	Khách hàng được hưởng ưu đãi miễn phí (vd : khi một chuyến bay bị hủy, hãng hàng không có thể sẽ sắp xếp cho khách hàng chỗ nghỉ trong khách sạn và khi check-out khách hàng không phải trả tiền).
5	Corporate	Khách hàng đến từ các công ty, doanh nghiệp.
6	Groups	Khách hàng đến từ các hội nhóm, tổ chức.
7	Direct	Khách hàng đặt phòng trực tiếp tại khách sạn.
8	Undefined	Không xác định.

Table 5. Dictionary thuộc tính Meal

Meal		
STT	Giá trị	Ý nghĩa
1	BB	Bed & Breakfast: chỉ bao gồm bữa sáng.
2	HB	Half Board: Bữa sáng và bữa tối được bao gồm trong giá phòng. Trong một số trường hợp, bạn có thể chọn nhận bữa trưa thay vì bữa sáng - khách sạn sẽ xác nhận điều này khi khách hàng đến.
3	FB	Full Board: được bao gồm bữa sáng, bữa trưa và bữa tối.
4	SC	Self-care: Khách hàng tự phục vụ.

Table 6. Dictionary thuộc tính Distribution_channel

Distribution_channel		
STT	Giá trị	Ý nghĩa
1	TA/TO	Travel Agency/Tour Operator: Kênh phân phối thông qua các đại lý du lịch và công ty điều hành tour.
2	Direct	Kênh phân phối trực tiếp (khách hàng đặt phòng trực tiếp tại khách sạn).
3	Corporate	Kênh phân phối thông qua các công ty, doanh nghiệp.
4	Undefined	Không xác định.

Table 7. Dictionary thuộc tính Is_repeated_guest

Is_repeated_guest		
STT	Giá trị	Ý nghĩa
1	0	Khách hàng lần đầu tới.
2	1	Khách hàng cũ.

Table 8. Dictionary thuộc tính Deposit_type

Deposit_type		
STT	Giá trị	Ý nghĩa
1	No Deposit	Không có tiền cọc trước.
2	Non Refund	Một khoản đặt cọc đã được thực hiện bằng giá trị của tổng chi phí lưu trú.
3	Refundable	Một khoản đặt cọc đã được thực hiện với giá trị dưới tổng chi phí lưu trú.

Table 9. Dictionary thuộc tính Customer_type

Customer_type		
STT	Giá trị	Ý nghĩa
1	Group	Khách hàng đặt phòng theo nhóm.
2	Transient	Khách tạm trú (Khách lưu trú qua đêm không có ý định ở lâu dài).
3	Transient party	Khách tạm trú theo nhóm (Lưu trú qua đêm và không có ý định ở lâu dài).

Table 10. Dictionary thuộc tính Reservation_status

Table 10. Dictionary thuộc tính Reservation_status

Reservation_status		
STT	Giá trị	Ý nghĩa
1	Check-Out	Khách hàng đã Check-in nhưng đã rời đi.
2	No-Show	Khách hàng không Check-in và đã thông báo lý do cho khách sạn.
3	Cancel	Khách hàng hủy lượt đặt phòng.

1.4. Khảo sát và tiền xử lý dữ liệu

1.4.1. Khảo sát dữ liệu

⊕ Kiểm tra bộ dữ liệu:

	hotel	is_canceled	lead_time	arrival_date_year	arrival_date_month	arrival_date_week_number	arrival_date_day_of_month	stays_in_weekend_ni
0	Resort Hotel	0	342	2015	July	27		1
1	Resort Hotel	0	737	2015	July	27		1
2	Resort Hotel	0	7	2015	July	27		1
3	Resort Hotel	0	13	2015	July	27		1
4	Resort Hotel	0	14	2015	July	27		1

5 rows × 36 columns

Figure 1. Năm dòng dữ liệu đầu tiên của bộ dữ liệu

✚ Thống kê mô tả dữ liệu (EDA):

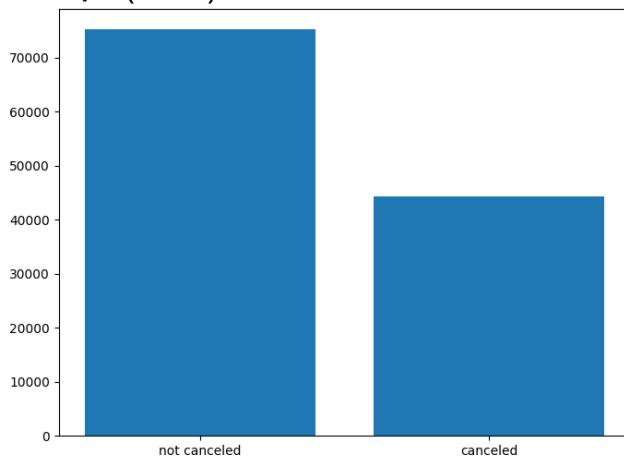


Figure 2. Biểu đồ số lượng đặt phòng và hủy phòng

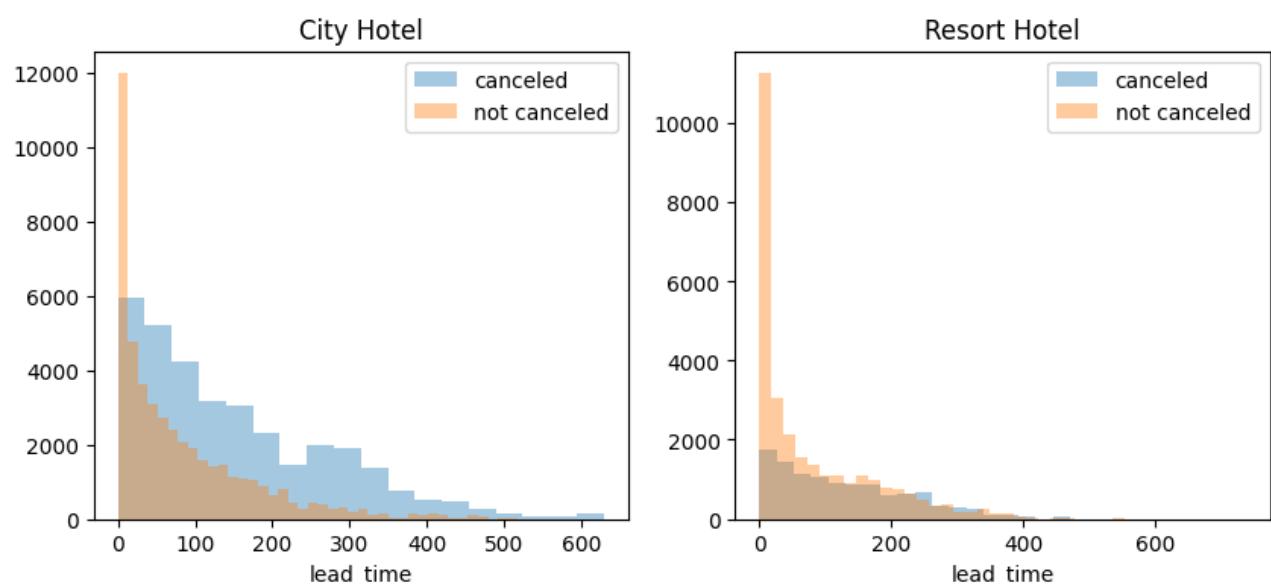


Figure 3. Đồ thị thể hiện mối quan hệ giữa dữ liệu bị hủy đặt phòng trong hai loại hình khách sạn trong khoảng thời gian từ khi đặt phòng tới khi nhận được phòng

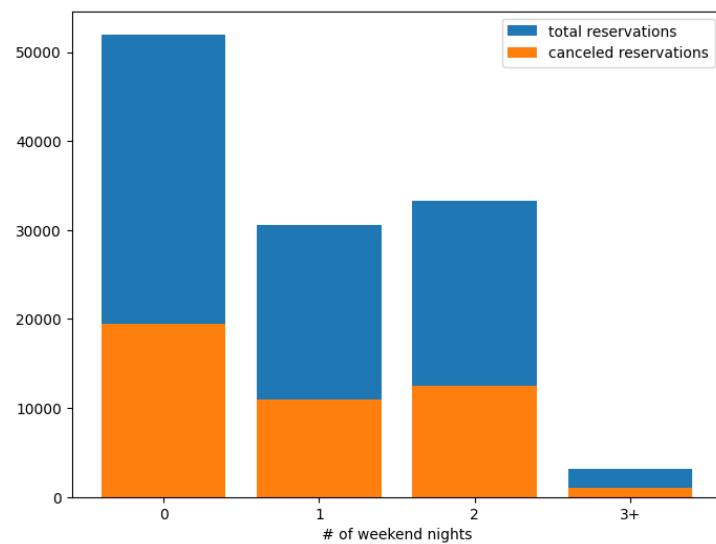


Figure 4. Biểu đồ sự tương quan giữa việc thuê phòng vào cuối tuần với hủy đặt phòng tại khách sạn

hotel	0
is_canceled	0
lead_time	0
arrival_date_year	0
arrival_date_month	0
arrival_date_week_number	0
arrival_date_day_of_month	0
stays_in_weekend_nights	0
stays_in_week_nights	0
adults	0
children	4
babies	0
meal	0
country	488
market_segment	0
distribution_channel	0
is_repeated_guest	0
previous_cancellations	0
previous_bookings_not_canceled	0
reserved_room_type	0
assigned_room_type	0
booking_changes	0
deposit_type	0
agent	16340
company	112593
days_in_waiting_list	0
customer_type	0
adr	0
required_car_parking_spaces	0
total_of_special_requests	0
reservation_status	0
reservation_status_date	0
name	0
email	0
phone-number	0
credit_card	0

Figure 5. Thông kê số lượng giá trị bị khuyết của từng thuộc tính

1.4.2. Các phương pháp tiền xử lý

- Xóa thuộc tính agent và company vì có quá nhiều dữ liệu Null.
- Xóa thuộc tính credit_card vì ẩn nhiều thông tin.
- Xóa các dòng dữ liệu của thuộc tính children có giá trị Null vì các dòng này quá ít nên sẽ không ảnh hưởng nhiều đến việc phân tích.
- Thay thế các dòng dữ liệu Null trong thuộc tính country thành các giá trị có tần suất xuất hiện cao nhất. (giá trị “PTR” xuất hiện nhiều nhất).
- Xóa các dòng dữ liệu có giá trị ‘Undefined’ vì thiếu thông tin.
- Tokenize thuộc tính ‘meal’ từ chữ sang số. (“BB”:1, “FB”:2, “HB”:3, “SC”:4)
- Tokenize thuộc tính “reserved_room_type” và “assigned_room_type” từ chữ sang số. (“A”:1, “B”:2, “C”:3, “D”:4, “E”:5, “F”:6, “G”:7, “H”:8, “I”:9, “K”:10, “L”:11, “P”:12)
- Tokenize tính tháng “arrival_date_month” từ chữ sang số.
- Sửa tên thuộc tính arrival_date_month thành arrival_date_month_name.
- Thêm thuộc tính “id_reservation_status_date” theo định dạng YYYYMMDD của “reservation_status_date”.
- Tạo thuộc tính arrival_date_full dựa vào các thuộc tính khác như arrival_date_year (YYYY), arrival_date_month_number (MM) và arrival_date_day_of_month (DD).
- Tạo các thuộc tính: id_arrival_date, arrival_date_quarter, arrival_date_day_of_week, arrival_date_date_name, arrival_date_day_name_abrev, arrival_date_month_name_abrev, arrival_date_weekday_flag.

1.4.3. Kết quả tiền xử lý dữ liệu

- Dữ liệu gồm có **118216** dòng dữ liệu và **43** cột.

Index: 118216 entries, 0 to 119389													
Data columns (total 43 columns):													
#	Column									Non-Null Count	Dtype		
0	hotel									118216	non-null	object	
1	is_canceled									118216	non-null	int64	
2	lead_time									118216	non-null	int64	
3	arrival_date_year									118216	non-null	int64	
4	arrival_date_month_name									118216	non-null	object	
5	arrival_date_week_number									118216	non-null	int64	

	hotel	is_canceled	lead_time	arrival_date_year	arrival_date_month_name	arrival_date_week_number	arrival_date	stays_in_week_nights	stays_in_week_nights	adults	children	babies	meal_plan
0	Resort Hot	0	342	2015	July	27	1	0	0	2	0	0	1
1	Resort Hot	0	737	2015	July	27	1	0	0	2	0	0	1
2	Resort Hot	0	7	2015	July	27	1	0	1	1	0	0	1
3	Resort Hot	0	13	2015	July	27	1	0	1	1	0	0	1
4	Resort Hot	0	14	2015	July	27	1	0	2	2	0	0	1
5	Resort Hot	0	14	2015	July	27	1	0	2	2	0	0	1
6	Resort Hot	0	0	2015	July	27	1	0	2	2	0	0	1
7	Resort Hot	0	9	2015	July	27	1	0	2	2	0	0	2
8	Resort Hot	1	85	2015	July	27	1	0	3	2	0	0	1
9	Resort Hot	1	75	2015	July	27	1	0	3	2	0	0	3
10	Resort Hot	1	23	2015	July	27	1	0	4	2	0	0	1
11	Resort Hot	0	35	2015	July	27	1	0	4	2	0	0	3
12	Resort Hot	0	68	2015	July	27	1	0	4	2	0	0	1
13	Resort Hot	0	18	2015	July	27	1	0	4	2	1	0	3
14	Resort Hot	0	37	2015	July	27	1	0	4	2	0	0	1
15	Resort Hot	0	68	2015	July	27	1	0	4	2	0	0	1

Figure 6. Kết quả sau khi tiền xử lý dữ liệu

1.5. Xây dựng kho dữ liệu

1.5.1. Khái niệm Dimensional Modeling

Mô hình chiều dữ liệu (*Dimensional modeling*) [3] là một kỹ thuật thiết kế logic thường được sử dụng trong phần thiết kế data warehouse nhằm tìm cách trình bày dữ liệu trong một khuôn khổ tiêu chuẩn trực quan cho phép truy cập hiệu suất cao.

- Bảng Fact:** Chứa các dữ liệu thuộc nhóm định lượng hoặc sự kiện, có tính chất đo lường mà người dùng cần phân tích và được thiết kế dựa trên business process.
- Bảng Dimension:** Chứa các thuộc tính mô tả của bảng fact.

Bảng Dim là bảng dữ liệu tĩnh, bảng Fact là dữ liệu động được nạp bằng các thao tác. Khoa ngoại của Fact được tạo bởi khoa của các bảng Dim. Nghĩa là khoa chính của các bảng Dim chính là khoa ngoại của bảng Fact.

1.5.2. Xây dựng sơ đồ bông tuyết

1.5.2.1. Khái niệm lược đồ hình bông tuyết (Snowflake schema)

Lược đồ hình sao là loại lược đồ Kho dữ liệu đơn giản nhất. Nó được gọi là **lược đồ sao** vì cấu trúc của nó giống như một Ngôi sao. Trong lược đồ hình sao, **tâm** của ngôi sao có thể có một bảng sự kiện (fact) và số lượng bảng chiều (dimension) được liên kết. Nó còn được gọi là *Star Join Schema* và được tối ưu hóa để truy vấn các tập dữ liệu lớn.

Lược đồ hình bông tuyết [4] là một phần cải tiến của Lược đồ hình sao và trong đó một số chiều được phân cấp để thể hiện rõ ràng dạng chuẩn của bảng chiều. Nó được gọi là **bông tuyết** vì sơ đồ của nó giống như một Bông tuyết.

Ưu điểm:

Lược đồ bông tuyết nằm trong cùng một họ với mô hình logic hình sao. Lược đồ hình sao được coi là một trường hợp đặc biệt của lược đồ bông tuyết. Và có những ưu điểm hơn so với lược đồ sao bao gồm:

- Một số công cụ mô hình hóa cơ sở dữ liệu đa chiều (OLAP) được tối ưu hóa cho các lược đồ bông tuyết.
- Đơn giản hóa các thuộc tính dẫn đến sự tiết kiệm, nhưng đánh đổi là sự phức tạp bổ sung trong các truy vấn nguồn.
- Một số chiều được phân cấp để thể hiện rõ ràng dạng chuẩn của bảng chiều.

Nhược điểm:

- Mức chuẩn hóa thuộc tính bổ sung thêm độ phức tạp cho các phép truy vấn nguồn so với lược đồ hình sao.

1.5.2.2. Sơ đồ bong tuyêt minh họa

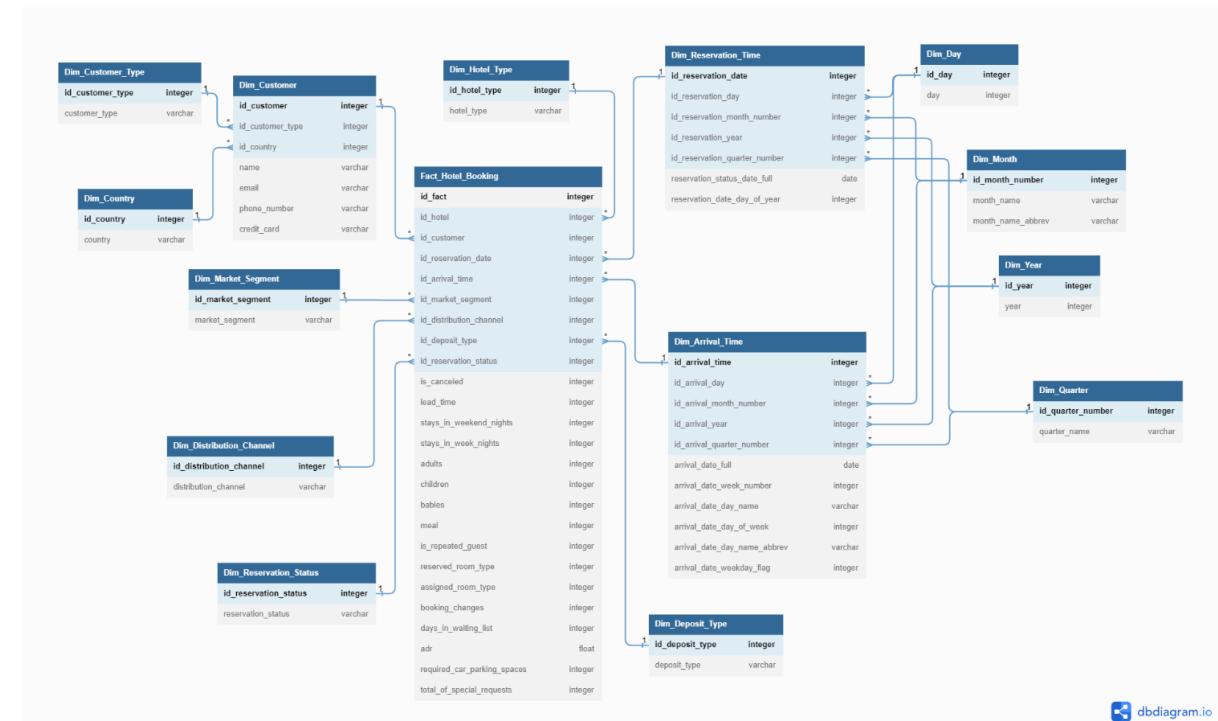


Figure 7. Sơ đồ bong tuyêt tập dữ liệu hoạt động đặt phòng khách sạn

1.5.3. Mô tả chi tiết các bảng dữ liệu

1.5.3.1. Bảng Fact

Table 11. Fact_Hotel_Booking

Ràng buộc	Tên cột	Kiểu dữ liệu	Mô tả	NULL
PK	id_fact	int	Mã ngày khách đến.	NOT
FK	id_hotel	int	Mã của khách sạn.	NOT
FK	id_id_customer	int	Mã khách hàng.	NOT
FK	id_reservation_date	int	Mã ngày đến của khách hàng.	NOT
FK	id_arrival_time	int	Mã phân khúc khách hàng.	NOT
FK	id_market_segment	int	Mã kênh phân phối bán hàng.	NOT
FK	id_distribution_channel	int	Mã hình thức đặt cọc.	NOT
FK	id_deposit_type	int	Mã trạng thái đặt phòng.	NOT
FK	id_reservation_status	int	Mã ngày trang thái đặt phòng.	NOT
	is_canceled	int	Hủy đặt phòng. Có hai giá trị: 0: không hủy. 1: hủy.	

	lead_time	int	Số ngày từ ngày khách hàng đặt chỗ đến ngày khách hàng đến nhận phòng.	
	stays_in_weekend_nights	int	Số đêm cuối tuần (thứ bảy hoặc chủ nhật) mà khách có lưu trú hoặc đặt phòng tại khách sạn.	
	stays_in_week_nights	int	Số đêm trong tuần (từ Thứ Hai đến Thứ Sáu) mà khách đã lưu trú hoặc đặt phòng tại khách sạn (được tính bằng cách đếm số đêm trong tuần).	
	adults	int	Số lượng khách hàng là người lớn.	
	children	int	Số lượng khách hàng là trẻ em.	
	babies	int	Số lượng khách hàng là em bé.	
	meal	int	Cho biết các bữa ăn được bao gồm trong giá phòng mà khách hàng đặt.	
	is_repeated_guest	int	Khách hàng cũ.	
	reserved_room_type	varchar	Mã loại phòng được đại diện bằng các chữ cái.	
	assigned_room_type	int	Mã loại phòng được chỉ định cho lượt đặt phòng.	
	booking_changes	int	Số lượng sửa đổi đối với lượt đặt phòng kể từ thời điểm đặt phòng được nhập trên PMS cho đến thời điểm nhận phòng hoặc lượt đặt phòng bị hủy bỏ.	
	days_in_waiting_list	int	Số ngày lượt đặt chỗ ở trong danh sách chờ trước khi nó được xác nhận với khách hàng.	
	adr	float	Doanh thu bình quân hằng ngày được tính bằng cách chia tổng của tất cả các giao dịch lưu trú cho tổng số đêm lưu trú.	
	required_car_parking_spaces	int	Số chỗ đậu xe ô tô theo yêu cầu của khách hàng.	
	total_of_special_requests	int	Số lượng yêu cầu đặc biệt của khách hàng.	

1.5.3.2. Bảng phụ bậc 1

Table 12. Dim_Reservation_Time

Ràng buộc	Tên cột	Kiểu dữ liệu	Mô tả	NULL
PK	id_reservation_date	int	Mã ngày đặt phòng.	NOT
FK	id_reservation_day	int	Mã ngày.	
FK	id_reservation_month_number	int	Mã tháng.	
FK	id_reservation_year	int	Mã năm.	
FK	id_reservation_quarter_number	int	Mã quý.	
	reservation_status_date_full	date	Ngày đặt phòng.	
	reservation_date_day_of_year	int	Số ngày trong năm (ngày trạng thái đặt phòng).	

Table 13. Dim_Arrival_Time

Ràng buộc	Tên cột	Kiểu dữ liệu	Mô tả	NULL
PK	id_arrival_time	int	Mã ngày khách đến.	NOT
FK	id_arrival_day	int	Mã ngày.	
FK	id_arrival_month_number	int	Mã tháng.	
FK	id_arrival_year	int	Mã năm.	
FK	id_arrival_quarter_number	int	Mã quý.	
	arrival_date_full	date	Ngày khách hàng đến.	
	arrival_date_week_number	int	Số thứ tự tuần trong năm của ngày khách hàng đến với giá trị từ 1 đến 53.	
	arrival_date_day_name	varchar	Ngày trong tuần với giá trị từ thứ hai đến chủ nhật (bằng chữ).	

	arrival_date_day_of_week	int	Ngày trong tuần với giá trị từ thứ hai đến chủ nhật (bằng số).	
	arrival_date_day_name_abbrev	varchar	Ngày trong tuần với giá trị từ thứ hai đến chủ nhật (bằng chữ viết tắt).	
	arrival_date_weekday_flag	int	Ngày trong tuần.	

Table 14. Dim_Customer

Ràng buộc	Tên cột	Kiểu dữ liệu	Mô tả	NULL
PK	id_customer	int	Mã khách hàng.	NOT
FK	id_customer_type	int	Mã loại khách hàng.	
FK	id_country	int	Mã quốc tịch khách hàng.	
	name	varchar	Tên khách hàng.	
	email	varchar	Email của khách hàng.	
	phone_number	varchar	Số điện thoại của khách hàng.	
	credit_card	varchar	Số thẻ tín dụng của khách hàng.	

Table 15. Dim_Hotel_Type

Ràng buộc	Tên cột	Kiểu dữ liệu	Mô tả	NULL
PK	id_hotel_type	int	Mã của khách sạn.	NOT
	hotel_type	varchar	Tên của khách sạn.	

Table 16. Dim_Deposit_Type

Ràng buộc	Tên cột	Kiểu dữ liệu	Mô tả	NULL
PK	id_deposit_type	int	Mã hình thức đặt cọc.	NOT
	deposit_type	varchar	Tên hình thức đặt cọc.	

Table 17. Dim_Reservation_Status

Ràng buộc	Tên cột	Kiểu dữ liệu	Mô tả	NULL
PK	id_reservation_status	int	Mã lượt đặt phòng.	NOT
	reservation_status	varchar	Trạng thái của lượt đặt phòng.	

Table 18. Dim_Distribution_Channel

Ràng buộc	Tên cột	Kiểu dữ liệu	Mô tả	NULL
PK	id_distribution_channel	int	Mã kênh bán hàng.	NOT
	distribution_channel	varchar	Tên kênh bán hàng.	

Table 19. Dim_Market_Segment

Ràng buộc	Tên cột	Kiểu dữ liệu	Mô tả	NULL
PK	id_market_segment	int	Mã phân khúc khách hàng.	NOT
	market_segment	varchar	Tên của các phân khúc khách hàng.	

1.5.3.3. Bảng phụ bậc 2

Table 20. Dim_Month

Ràng buộc	Tên cột	Kiểu dữ liệu	Mô tả	NULL
PK	id_month_number	int	Mã tháng.	NOT
	month_name	varchar	Tên tháng.	
	month_name_abrev	varchar	Tên tháng viết tắt.	

Table 21. Dim_Day

Ràng buộc	Tên cột	Kiểu dữ liệu	Mô tả	NULL
PK	id_day	int	Mã ngày.	NOT
	day	int	Ngày bằng số.	

Table 22. Dim_Year

Ràng buộc	Tên cột	Kiểu dữ liệu	Mô tả	NULL
PK	id_year	int	Mã năm.	NOT
	year	int	Năm bằng số.	

Table 23. Dim_Quarter

Ràng buộc	Tên cột	Kiểu dữ liệu	Mô tả	NULL
PK	id_quarter_number	int	Mã quý.	NOT
	quarter_name	varchar	Tên quý.	

Table 24. Dim_Customer_Type

Ràng buộc	Tên cột	Kiểu dữ liệu	Mô tả	NULL
PK	id_customer_type	int	Mã loại khách hàng.	NOT
	customer_type	varchar	Tên loại khách hàng.	

Table 25. Dim_Country

Ràng buộc	Tên cột	Kiểu dữ liệu	Mô tả	NULL
PK	id_country	int	Mã quốc tịch.	NOT
	country	varchar	Tên quốc tịch.	

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

2.1. Khái niệm

SSIS là tên viết tắt của SQL Server Integration Services, là một trong các thành phần chính của Microsoft SQL Server Enterprise Edition.

SSIS là công cụ dùng để thực hiện các tác vụ quản lý tích hợp dữ liệu (Data Integration) và là thành phần chính trong các ứng dụng của data warehouse như lưu trữ dữ liệu, trích xuất và tải dữ liệu, quản lý dữ liệu,...

2.2. Chuẩn bị công cụ và Data Warehouse

- Các công cụ sử dụng:
 - Microsoft Visual Studio 2022.
 - Microsoft SQL Server Management Studio.
 - SQL Server Data Tools cho phiên bản Visual Studio 2022.
- Các bước tạo kho dữ liệu bằng Microsoft SQL Server Management Studio:
- **Bước 1:** Mở Microsoft SQL Server Management Studio và kết nối với Server.

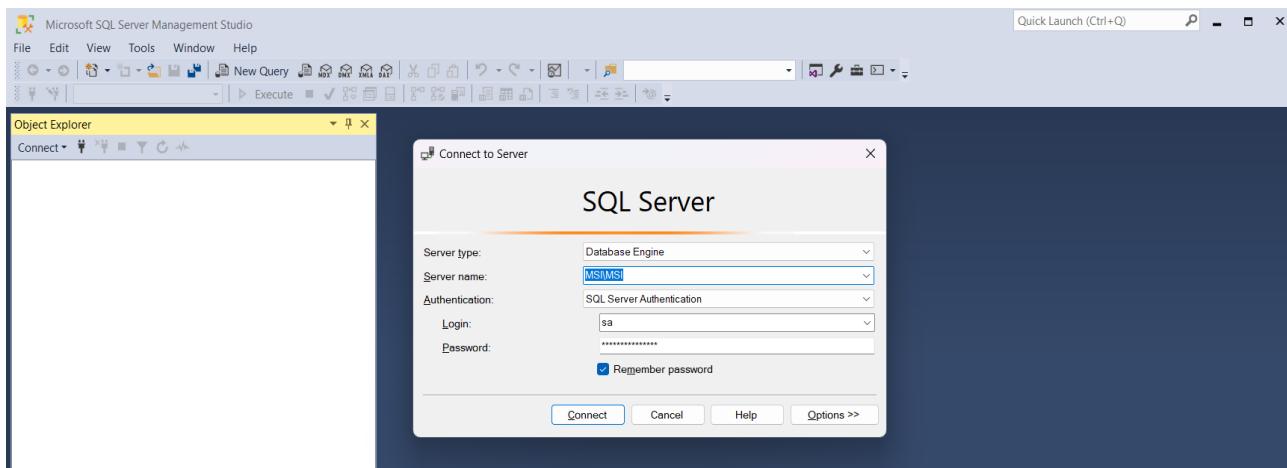


Figure 8. Mở SQL Server và kết nối với Server

- **Bước 2:** Tạo kho dữ liệu DB_HotelBooking, chọn New Database để mở giao diện tạo cơ sở dữ liệu mới.

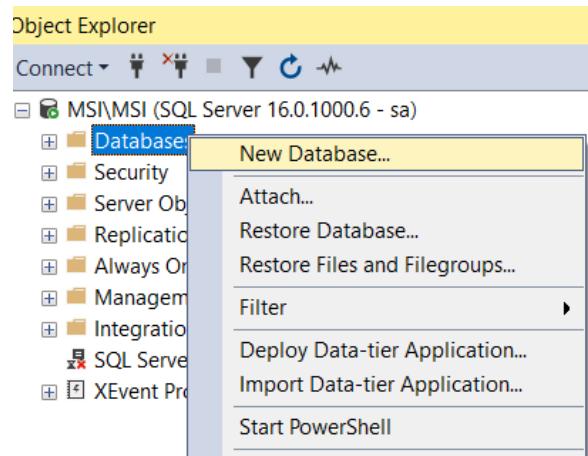


Figure 9. Tạo kho dữ liệu

- **Bước 3:** Nhập tên database mới và nhấn OK để kết thúc quá trình tạo database.

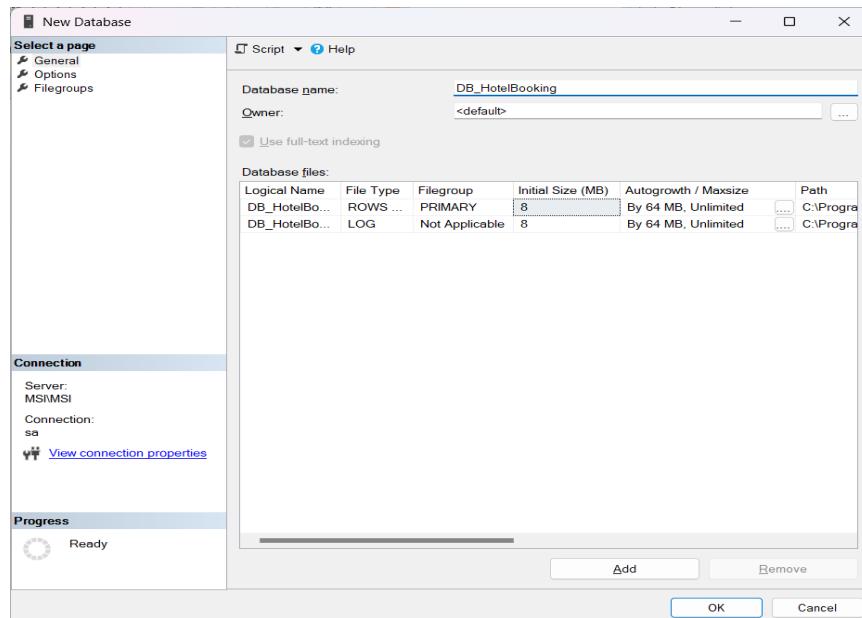


Figure 10. Đặt tên kho dữ liệu

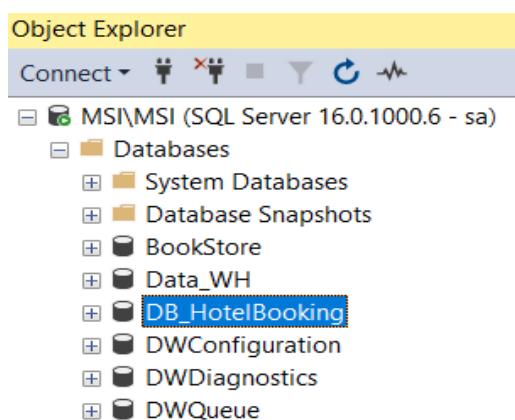


Figure 11. Kho dữ liệu đã được tạo

2.3. Tạo project SSIS trong Visual Studio 2022

- **Bước 1:** Chọn File → New → Project để mở giao diện:

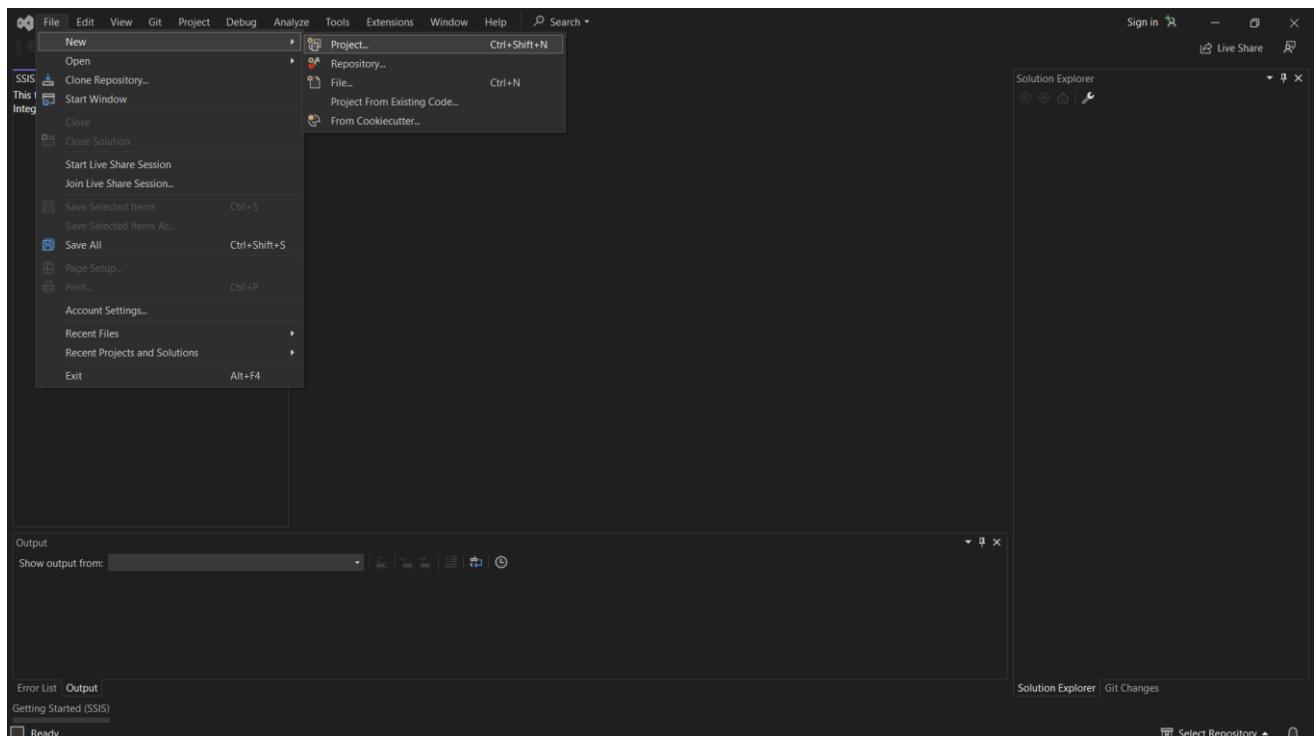


Figure 12. Tạo Project với Visual Studio 2022

- **Bước 2:** Search trên thanh tìm kiếm cụm Integration Services Project. Sau đó, chọn công cụ Integration Services Project chọn đường dẫn để lưu project và đặt tên là HotelBooking_SSIS.

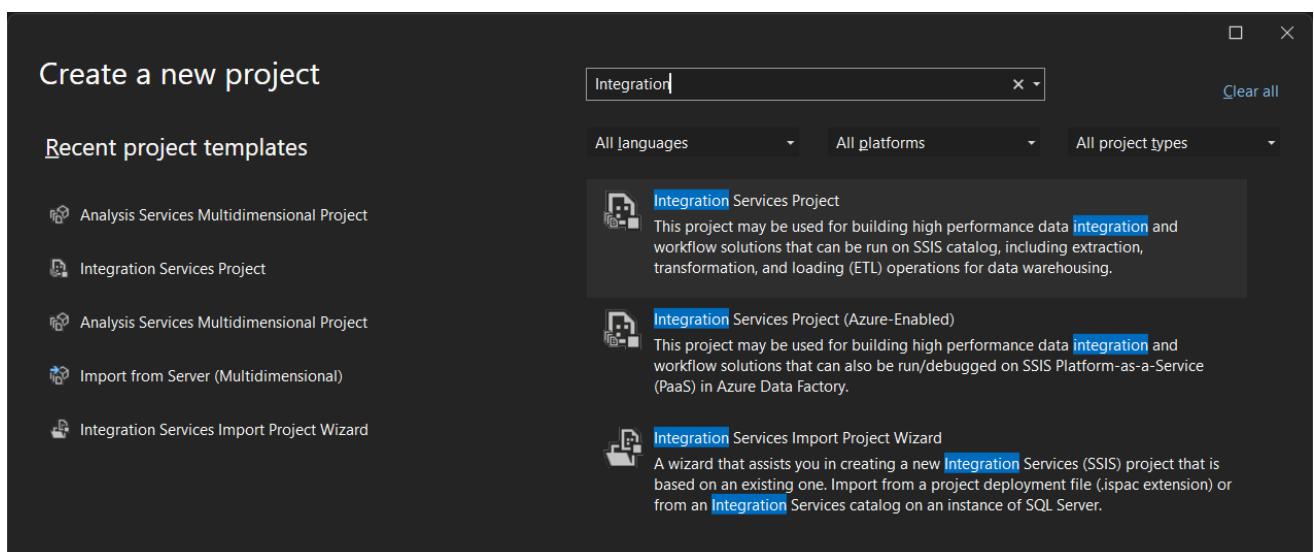


Figure 13. Giao diện chọn công cụ SSIS Visual Studio 2022

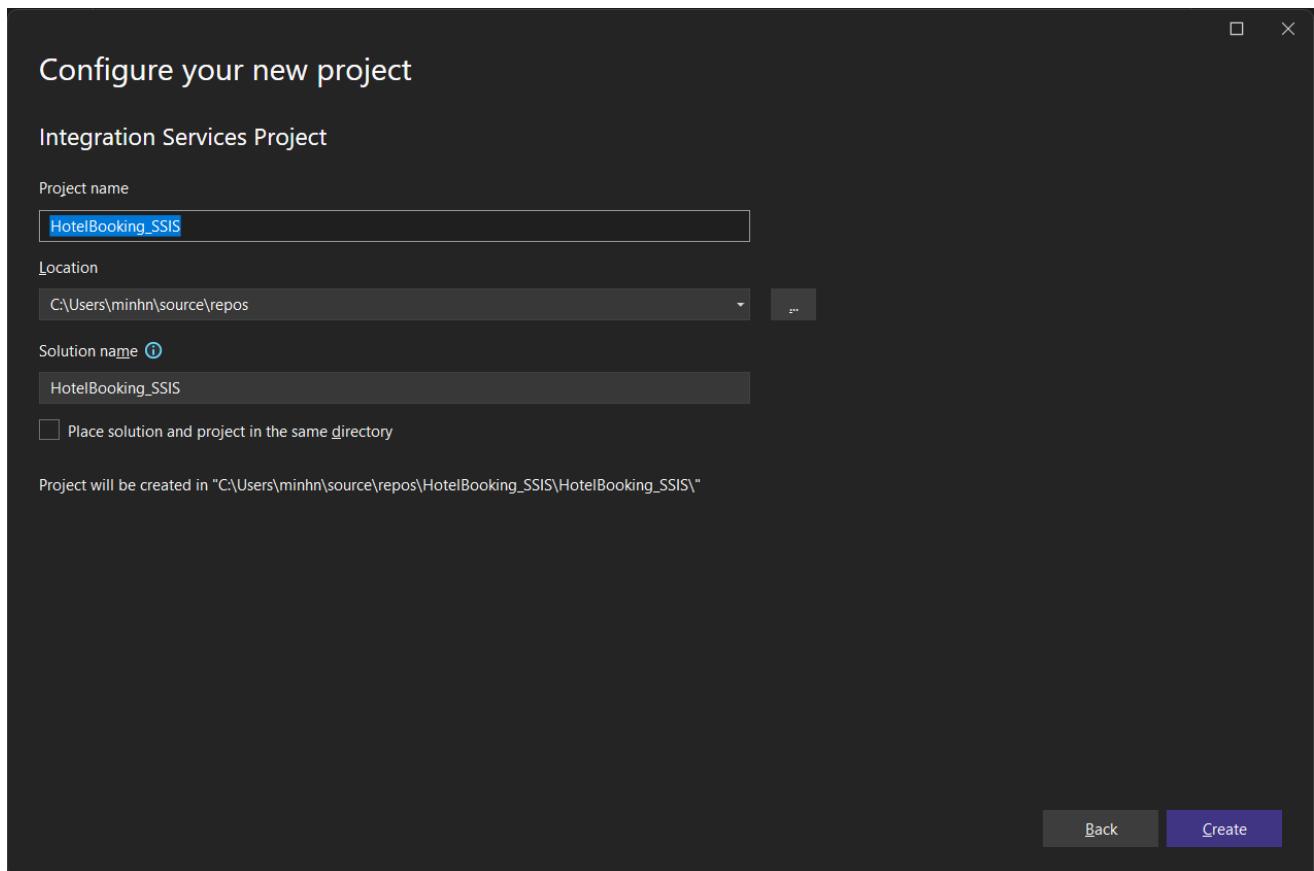


Figure 14. Đặt tên cho Project trong Visual Studio 2022

- **Bước 3:** Sau đó, nhấn OK và giao diện quá trình SSIS của công cụ BI sẽ hiện ra.
- o Các chức năng chính của SSIS nằm ở cột bên trái SSIS ToolBox.

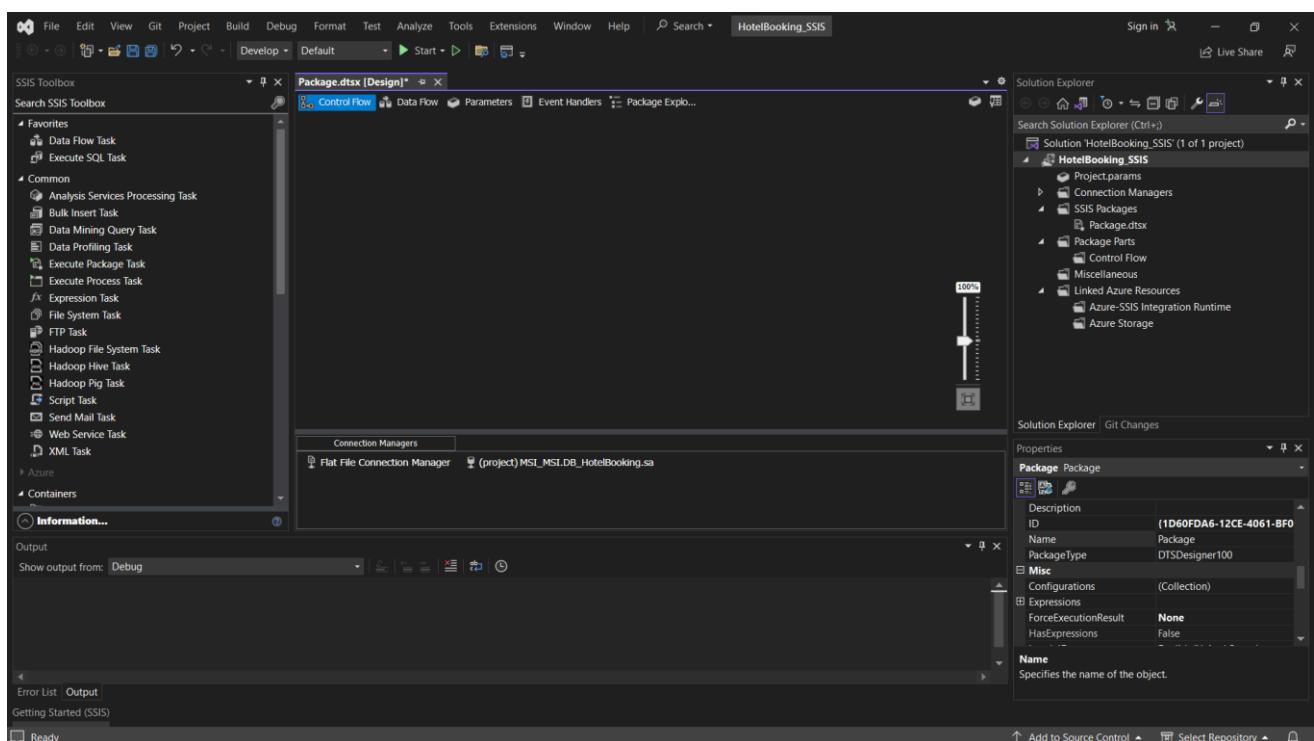


Figure 15. Giao diện làm việc với Project SSIS trong Visual Studio 2022

- **Bước 4:** Tạo kết nối với cơ sở dữ liệu trong SQL Server.
 - o Tại cửa sổ Solution Explorer, chọn Connection Managers → New Connection Manager

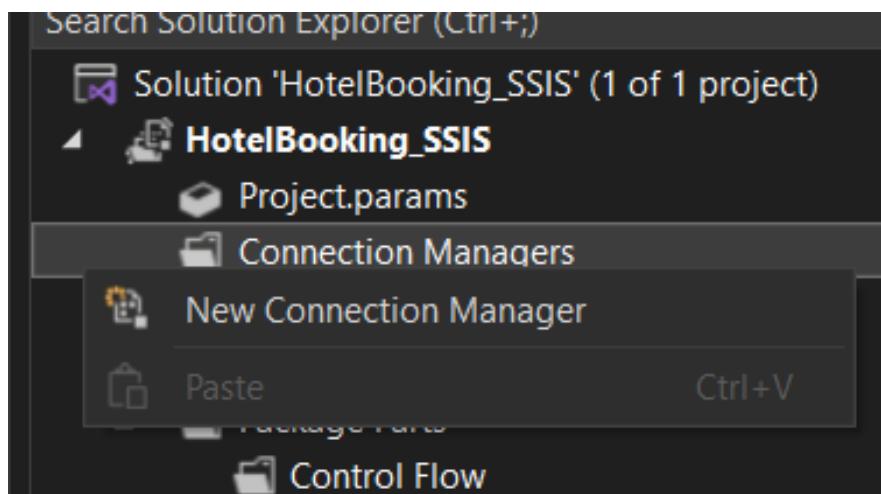


Figure 16. Tạo Connection với Database

- o Chọn OLEDB → Add.

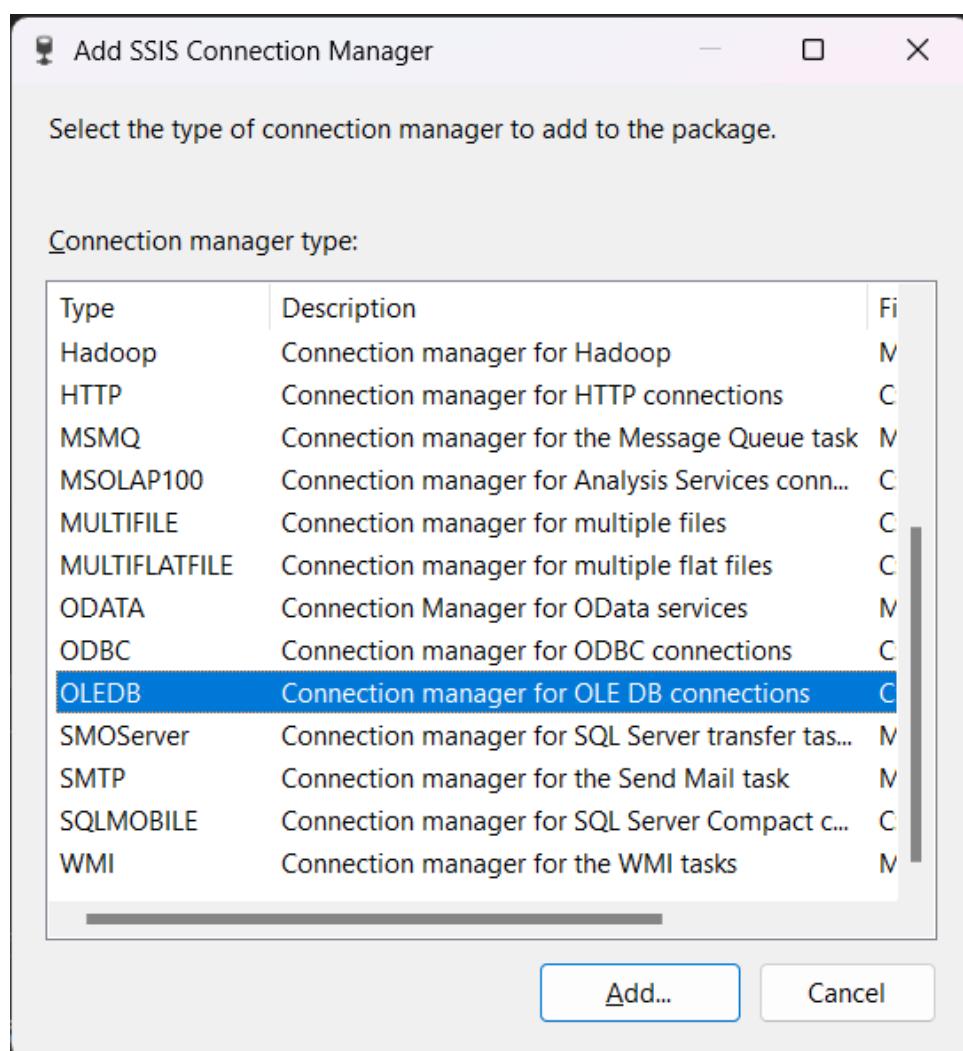


Figure 17. Thêm kết nối OLEDB

- Sau khi xuất hiện giao diện Configure OLE Connection Manager, chọn New:

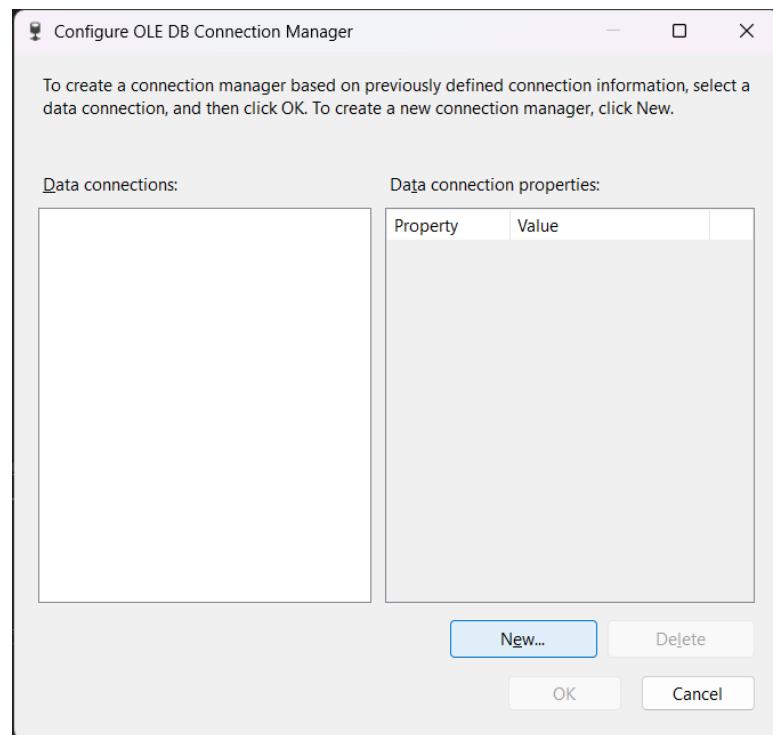


Figure 18. Thiết lập kết nối

- Nhập Server name và chọn Database name được dùng để tách dữ liệu và click OK.

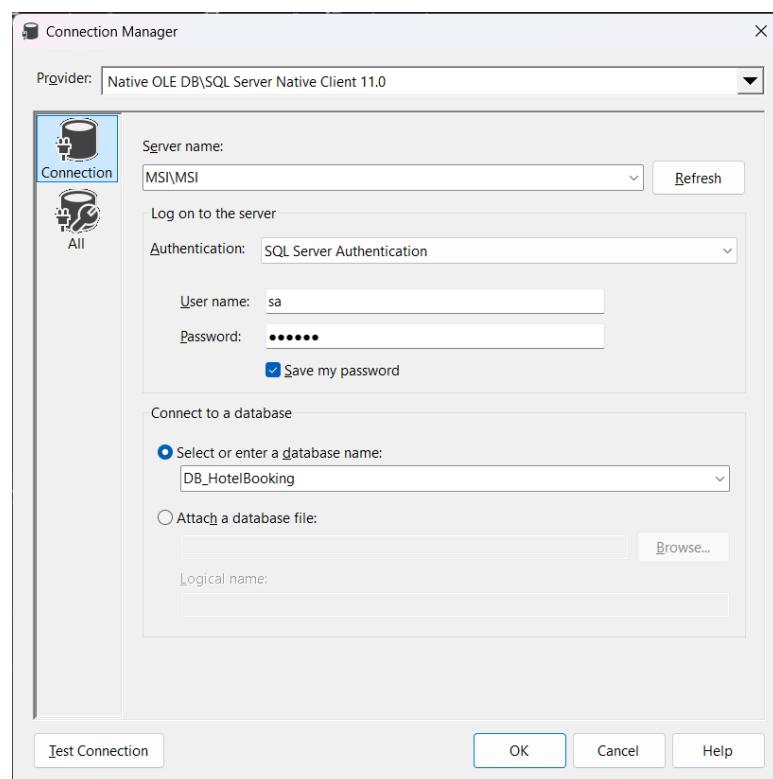


Figure 19. Nhập Server name và Database name

- Tiếp tục nhấn Ok để hoàn thành kết nối với cơ sở dữ liệu.

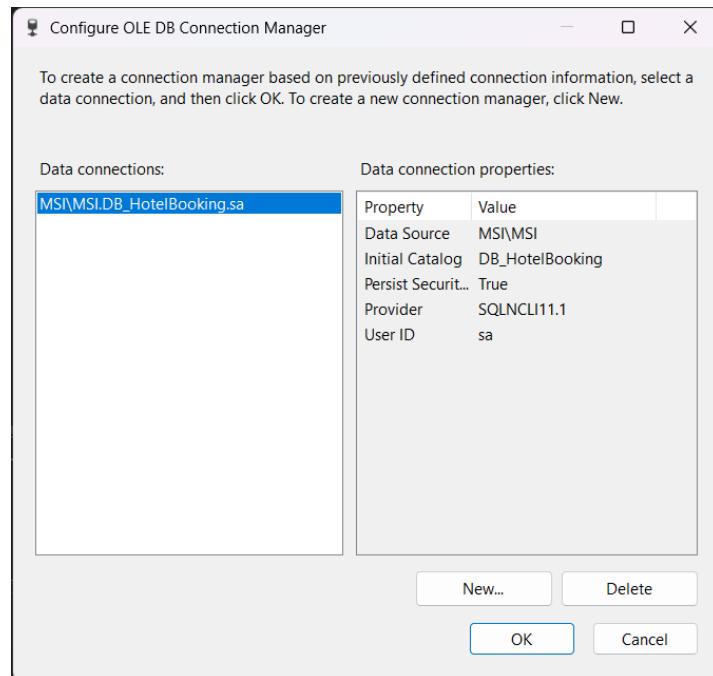


Figure 20. Hoàn thành kết nối với cơ sở dữ liệu

- Tiếp tục tạo kết nối với cơ sở dữ liệu DB_HotelBooking, ta được:

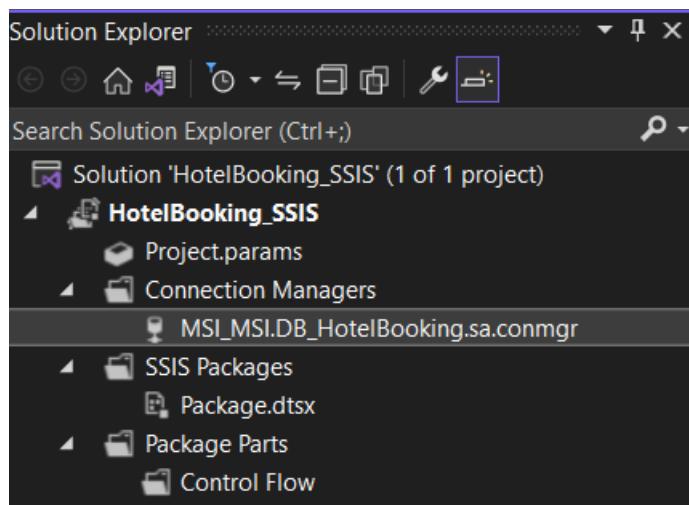


Figure 21. Kết nối thành công với cơ sở dữ liệu

2.4. Đỗ dữ liệu gốc vào kho

2.4.1. Các khái niệm

- **Trong SSIS có hai thành phần chính:**
 - *Data Transformation Run-time engine*: quản lý các control flow và package.
 - *DataFlow engine*: Data Source, Transformations, Destination.
- **Data Flow Task**: là một task chính trong các task của control flow. Nhiệm vụ chính của DataFlow là biến đổi dữ liệu. Có ba thành phần data flow trong SSIS:

- **Sources:** trích xuất dữ liệu từ kho dữ liệu như bảng và view trong cơ sở dữ liệu quan hệ, tệp và Analysis Services databases và là nguồn cung cấp dữ liệu để biến đổi.
 - **Transformations:** thực hiện các tác vụ như cập nhật, tóm tắt, làm sạch, hợp nhất và phân phối dữ liệu, sửa đổi giá trị trong cột, tra cứu giá trị trong bảng, làm sạch dữ liệu và tổng hợp giá trị cột.
 - **Destinations:** ghi dữ liệu từ data flow vào một kho dữ liệu cụ thể hoặc tạo tập dữ liệu trên bộ nhớ. Một data flow có thể có nhiều Destinations tải dữ liệu vào nhiều kho khác nhau.
- **OLE DB Source:** Được sử dụng để trích xuất dữ liệu từ các loại cơ sở dữ liệu quan hệ theo chuẩn OLE-DB bằng cách sử dụng bảng, view hoặc là câu lệnh SQL.
- Trình chỉnh sửa của OLE DB Source cho phép chúng ta tùy chỉnh hình thức truy cập dữ liệu khác nhau nhằm tải dữ liệu như lấy từ bảng, view đã tồn tại hoặc từ một bảng mới hoặc là kết quả của câu lệnh SQL.
 - Sử dụng trình quản lý kết nối OLE DB để kết nối với nguồn dữ liệu mà từ đó nó trích xuất dữ liệu.
- **OLE DB Destination:**
- Được sử dụng để tải dữ liệu vào các cơ sở dữ liệu quan hệ theo chuẩn OLE-DB bằng cách sử dụng bảng, view hoặc là câu lệnh SQL.
 - Trình chỉnh sửa của OLE DB Destination cho phép chúng ta tùy chỉnh hình thức truy cập dữ liệu khác nhau nhằm tải dữ liệu như lấy từ bảng, view đã tồn tại hoặc từ một bảng mới hoặc là kết quả của câu lệnh SQL.
 - Bao gồm các ánh xạ giữa các cột dữ liệu đầu vào và các cột trong nguồn dữ liệu đích.
 - Các dữ liệu của cột các ánh xạ phải tương thích với nhau.
 - Nếu cột dữ liệu đích không cho phép giá trị null thì bắt buộc phải có một cột ở nguồn dữ liệu đầu vào ánh xạ đến cột dữ liệu đích này nếu không sẽ xảy ra lỗi.

2.4.2. Tiến hành đổ dữ liệu từ nguồn

- **Bước 1:** Kéo chức năng Data Flow Task từ cột trái sang màn hình làm việc và đổi tên thành Hotel Booking.

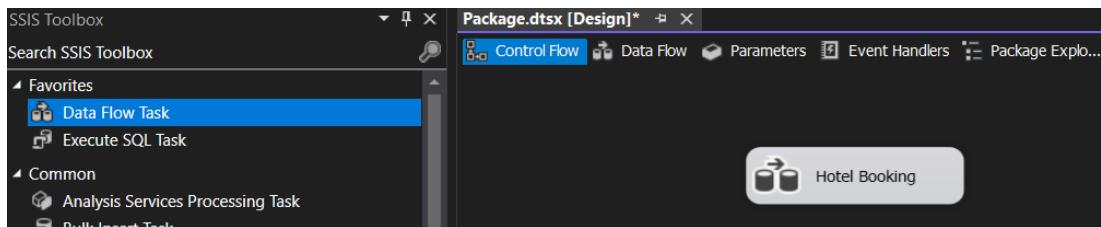


Figure 22. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Hotel Booking

- **Bước 2:** Double click vào Data Cleaning để chuyển từ Control Flow sang Data Flow và chọn chức năng Flat File Source.

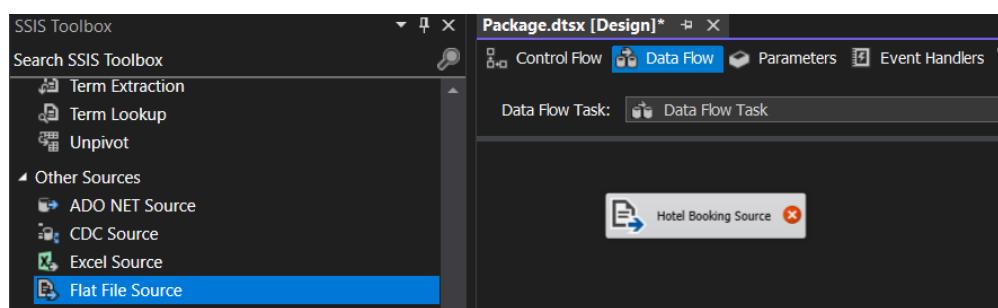


Figure 23. Nạp dữ liệu

- **Bước 3:** Chọn New để tiến hành chọn dữ liệu đổ vào.

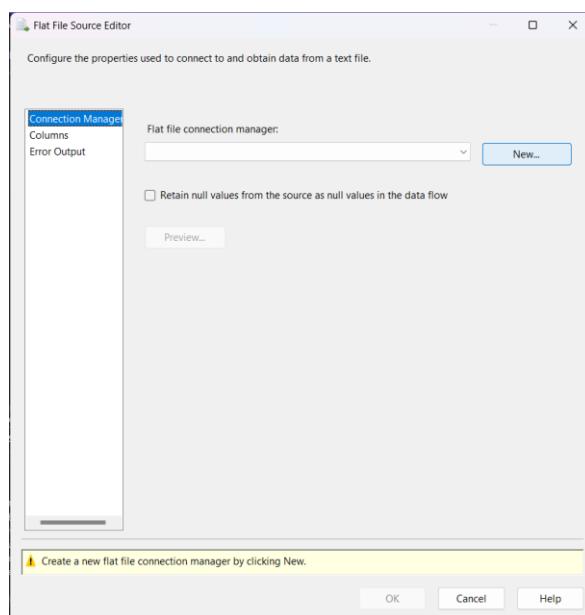


Figure 24. Chọn New để khởi tạo nguồn dữ liệu đầu vào

- **Bước 4:** Chọn đường dẫn của dữ liệu nguồn (Bộ dữ liệu đã được làm sạch trước đó).

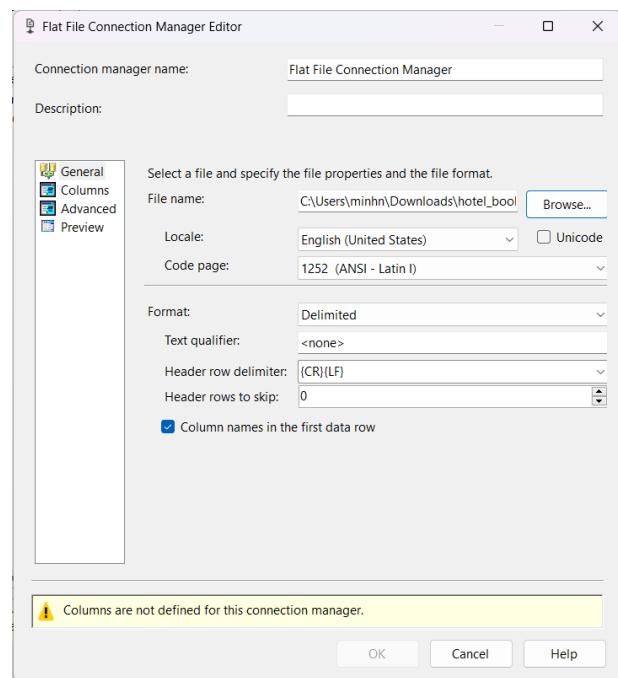


Figure 25. Chọn bộ dữ liệu đầu vào

- **Bước 5:** Kiểm tra các dòng cột của dữ liệu, nhấn OK.

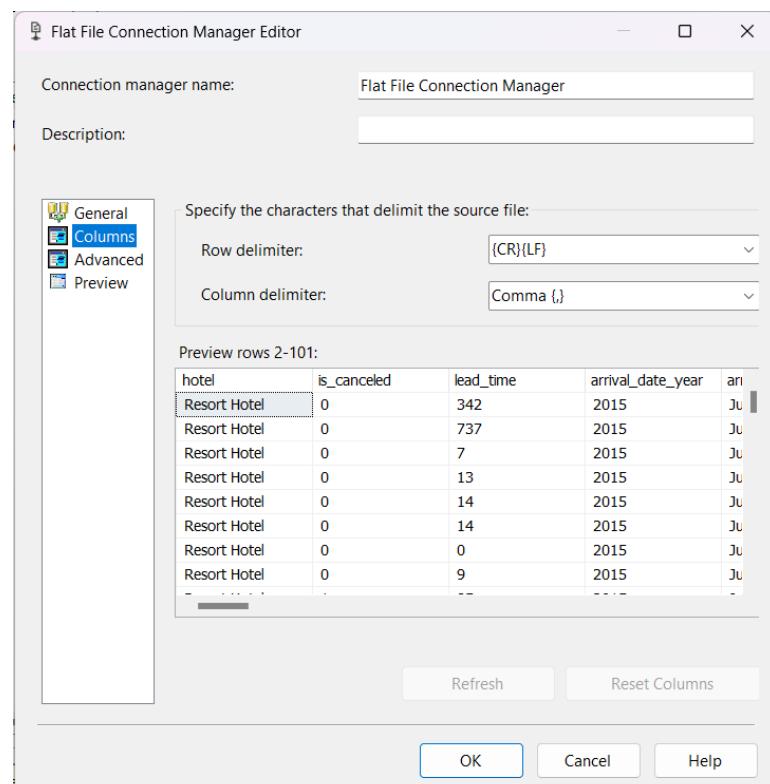


Figure 26. Xem trước dữ liệu đầu vào

- **Bước 6:** Kéo thả công cụ OLE DB Destination và đổi tên thành Hotel Booking Data, kết nối với Hotel Booking Source.

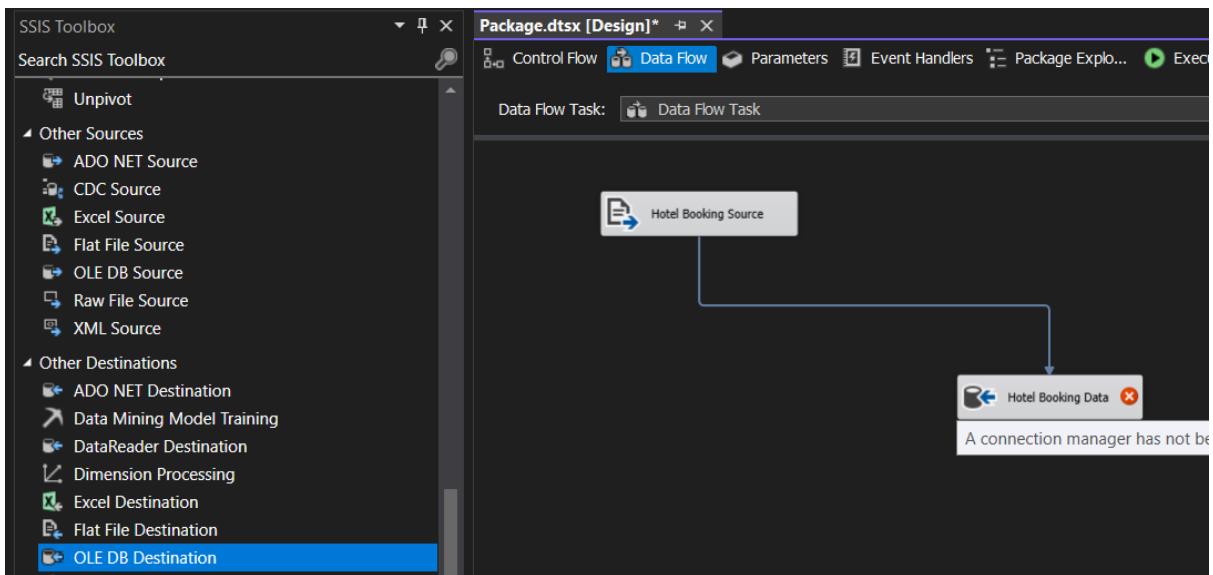
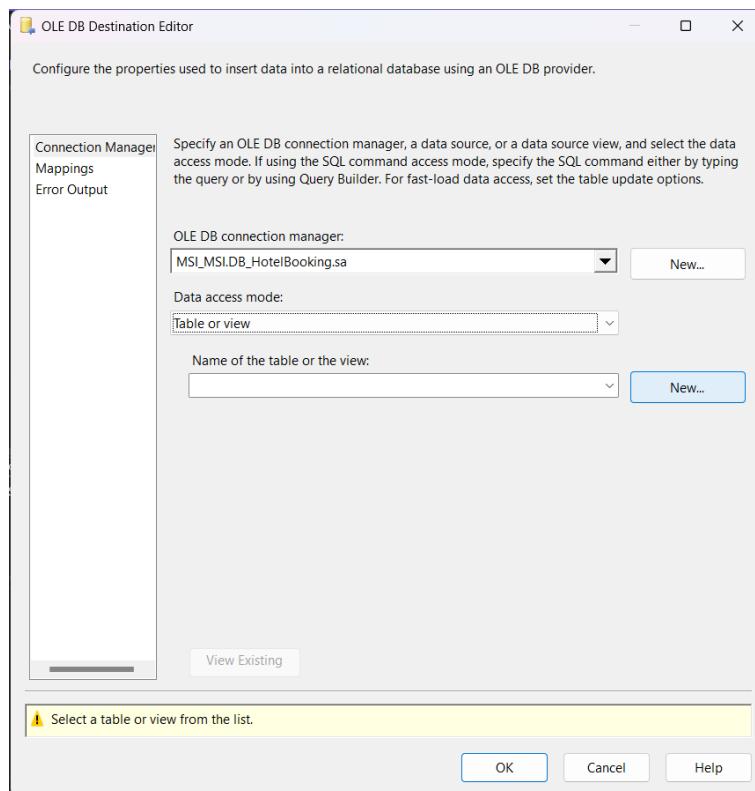


Figure 27. Đưa dữ liệu vào Database

- **Bước 7:** Double Click vào Hotel Booking Data, sau đó chọn New. Sau đó, tùy chỉnh kiểu dữ liệu của các thuộc tính cho phù hợp, nhấn OK.



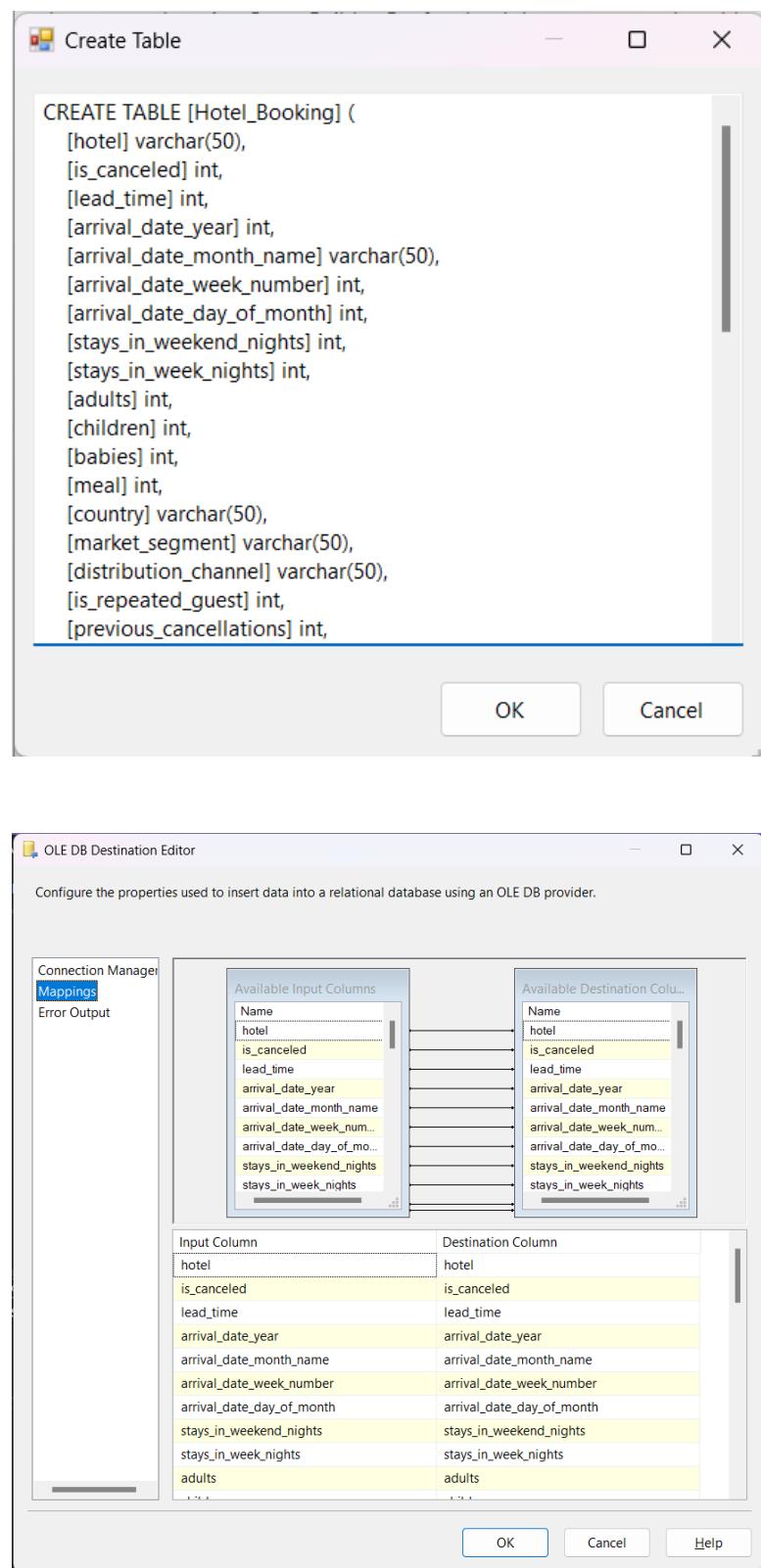


Figure 7-8-9. Quá trình khởi tạo để đổ dữ liệu vào Database

- **Bước 8:** Chọn Start để bắt đầu quá trình đổ dữ liệu. Khi đổ thành công, ta có kết quả như sau:

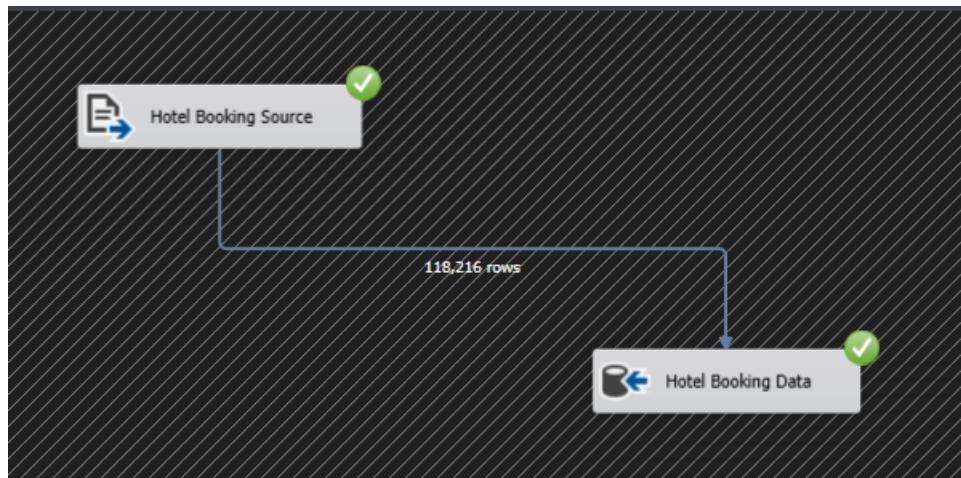


Figure 28. Đổ dữ liệu thành công

- **Bước 9:** Kiểm tra bảng DataNull và DataClean trên SQL Server.

hotel	is_canceled	lead_time	arrival_date_year	arrival_date_month_name	arrival_date_week_number	arrival_date_day_of_month	stays_in_weekend_nights	stays_in_week_nights	a
1 Resort Hotel	0	342	2015	July	27	1	0	0	2
2 Resort Hotel	0	737	2015	July	27	1	0	0	2
3 Resort Hotel	0	7	2015	July	27	1	0	1	1
4 Resort Hotel	0	13	2015	July	27	1	0	1	1
5 Resort Hotel	0	14	2015	July	27	1	0	2	2
6 Resort Hotel	0	14	2015	July	27	1	0	2	2
7 Resort Hotel	0	0	2015	July	27	1	0	2	2
8 Resort Hotel	0	9	2015	July	27	1	0	2	2
9 Resort Hotel	1	85	2015	July	27	1	0	3	2
10 Resort Hotel	1	75	2015	July	27	1	0	3	2
11 Resort Hotel	1	23	2015	July	27	1	0	4	2
12 Resort Hotel	0	35	2015	July	27	1	0	4	2
13 Resort Hotel	0	68	2015	July	27	1	0	4	2
14 Resort Hotel	0	18	2015	July	27	1	0	4	2

Figure 29. Kiểm tra bảng Hotel_Booking trong SQL Server

2.5. Tạo các bảng Dim và Bảng Fact

Các khái niệm

Sort: Sắp xếp dữ liệu đầu vào theo thứ tự tăng dần hoặc giảm dần và sao chép dữ liệu đã sắp xếp vào dữ liệu đầu ra. Ngoài ra, sort cũng loại bỏ các hàng trùng lặp như một phần của cách sắp xếp của nó.

Script component: Script có thể được sử dụng như một source, một transformation, hoặc một destination. Thành phần này hỗ trợ một đầu vào và nhiều đầu ra:

- Nếu sử dụng như một source thì Script component hỗ trợ nhiều đầu ra.
- Nếu được sử dụng như transformation thì Script component hỗ trợ một đầu vào và nhiều đầu ra.
- Nếu sử dụng như destination thì Script component hỗ trợ một đầu vào.

Aggregate: Được sử dụng Để thực hiện các phép tính tổng hợp trên các nhóm dữ liệu như Group by, Sum, Average, Count, Count Distinct, Minimum, Maximum,...

Lookup: Thực hiện tra cứu bằng cách nối dữ liệu trong các cột dữ liệu đầu vào với các cột trong tập dữ liệu tham chiếu thường để truy cập thông tin bổ sung trong bảng có liên quan dựa trên các giá trị trong các cột chung

2.5.1. *Bảng Dim_Hotel_Type*

- **Bước 1:** Kéo chức năng Data Flow Task từ cột trái sang màn hình làm việc và đổi tên thành Dim_Hotel_Type.

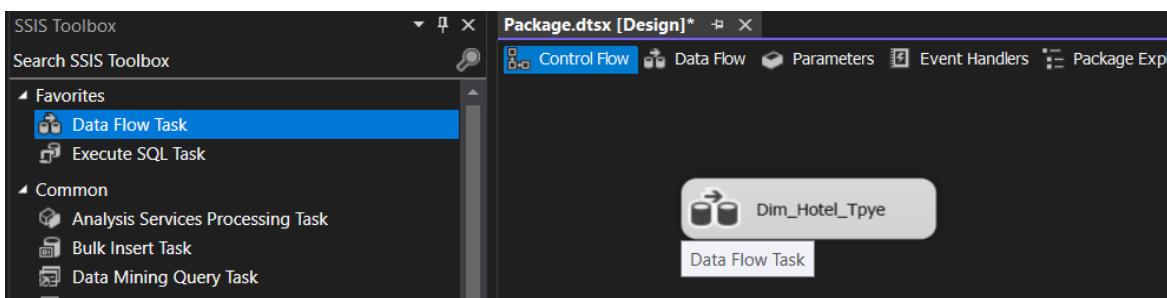


Figure 30. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Hotel_Type

- **Bước 2:** Nhấn double vào Data Flow Task vừa tạo và tìm kiếm chức năng OLE DB Source, kéo vào màn hình làm việc.

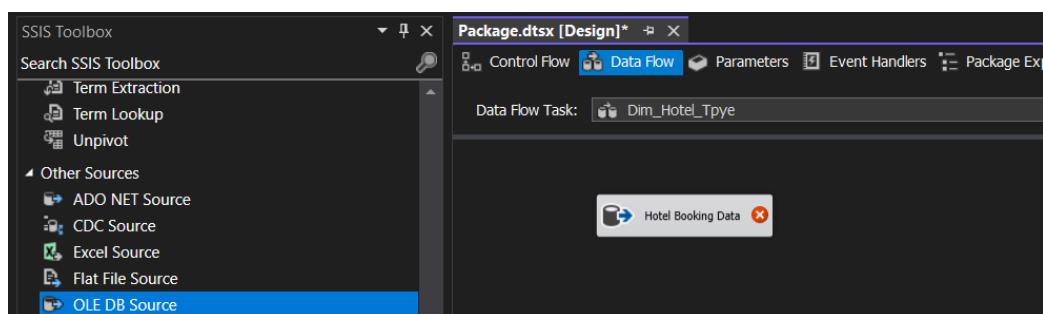


Figure 31. Kéo chức năng OLE DB Source vào màn hình làm việc

- **Bước 3:** Click double vào OLE DB Source và dẫn đến DB_HotelBooking. Sau đó chọn Name of the table or the view là [dbo].Hotel_Booking.

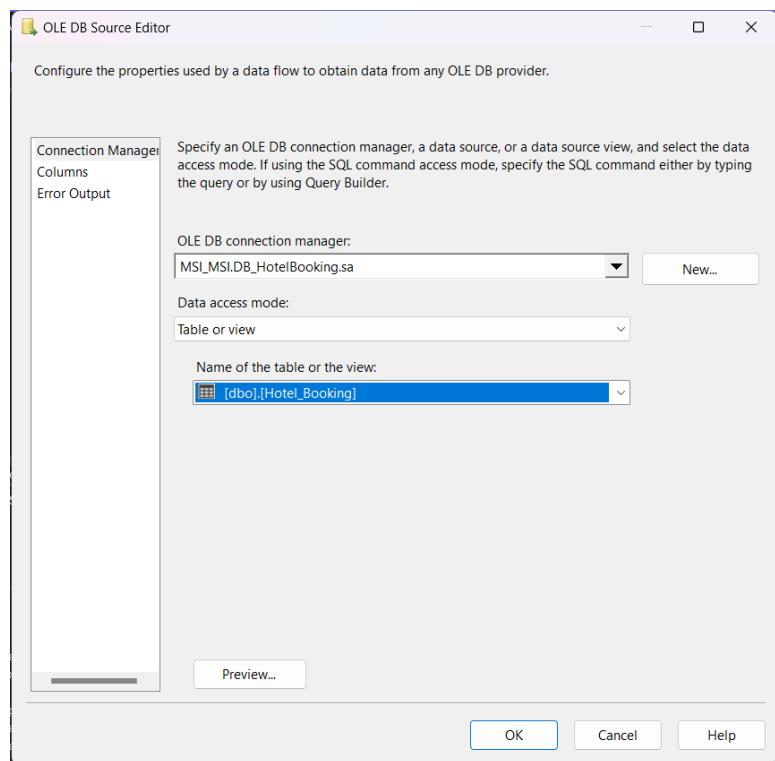


Figure 32. Chọn Table sẽ lấy dữ liệu

- **Bước 4:** Chọn Columns để lựa chọn các cột cần sử dụng và nhấn OK.

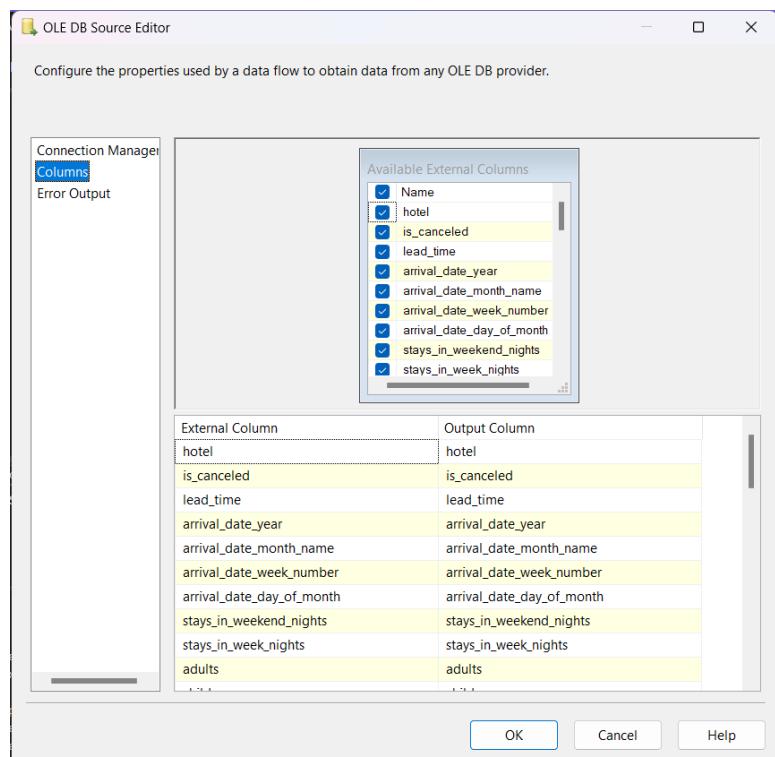


Figure 33. Chọn các column cần sử dụng

- **Bước 5:** Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau lên thuộc tính sử dụng là hotel.

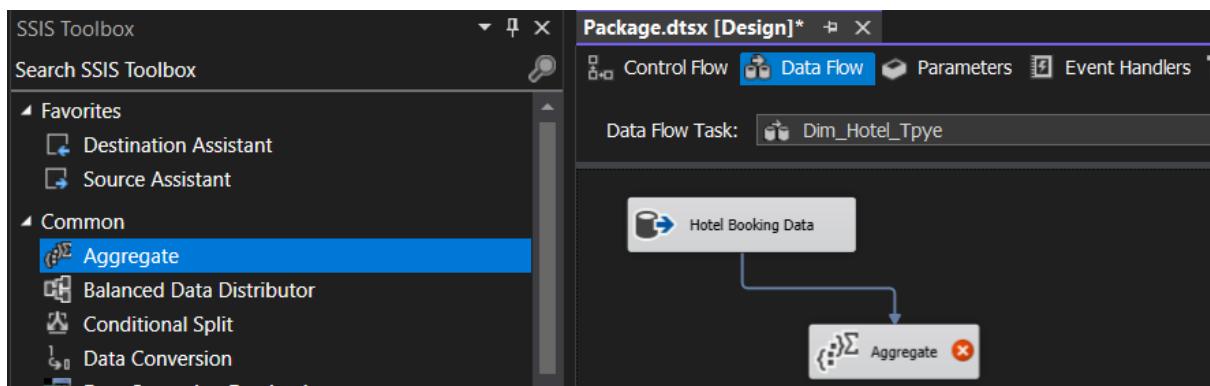


Figure 34. Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau

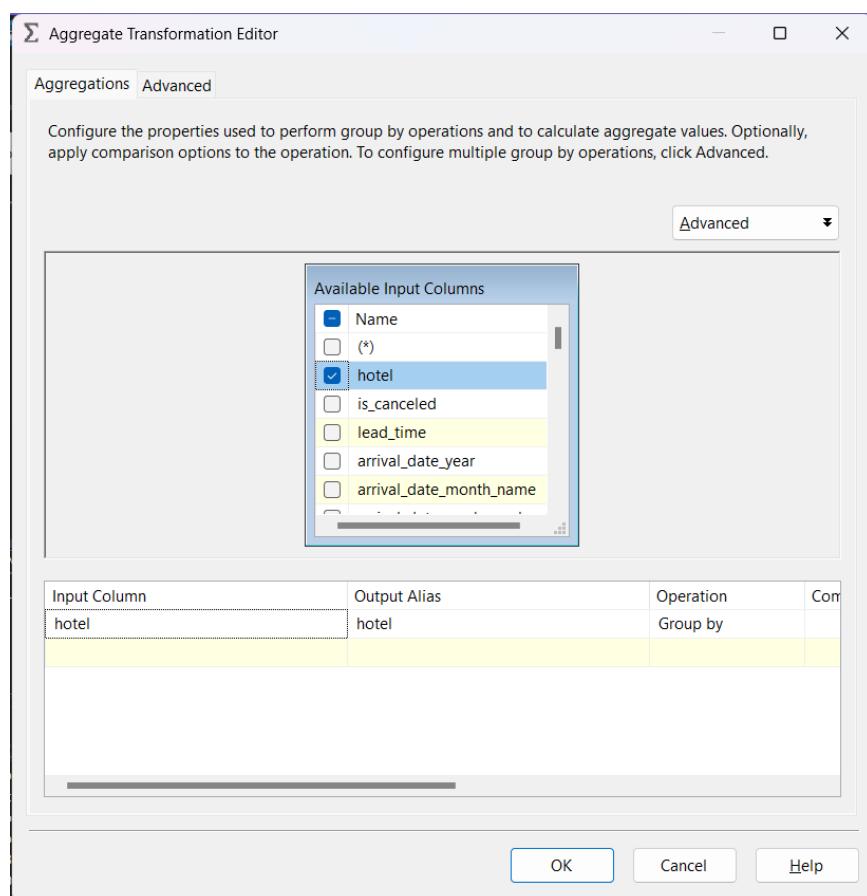


Figure 35. Chọn các thuộc tính cần gom nhóm

- **Bước 6:** Dùng công cụ Sort để sắp xếp các giá trị theo chiều tăng dần.

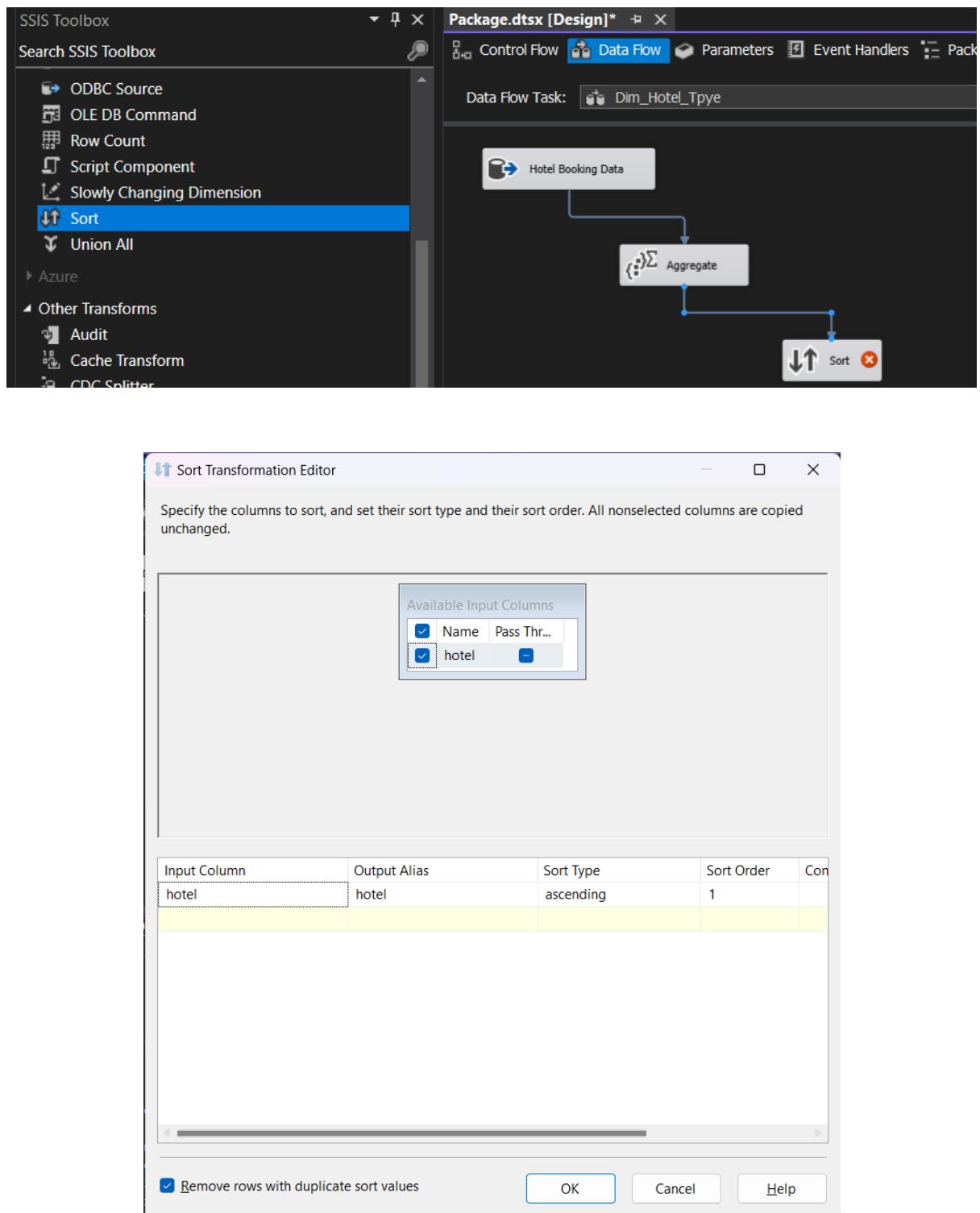
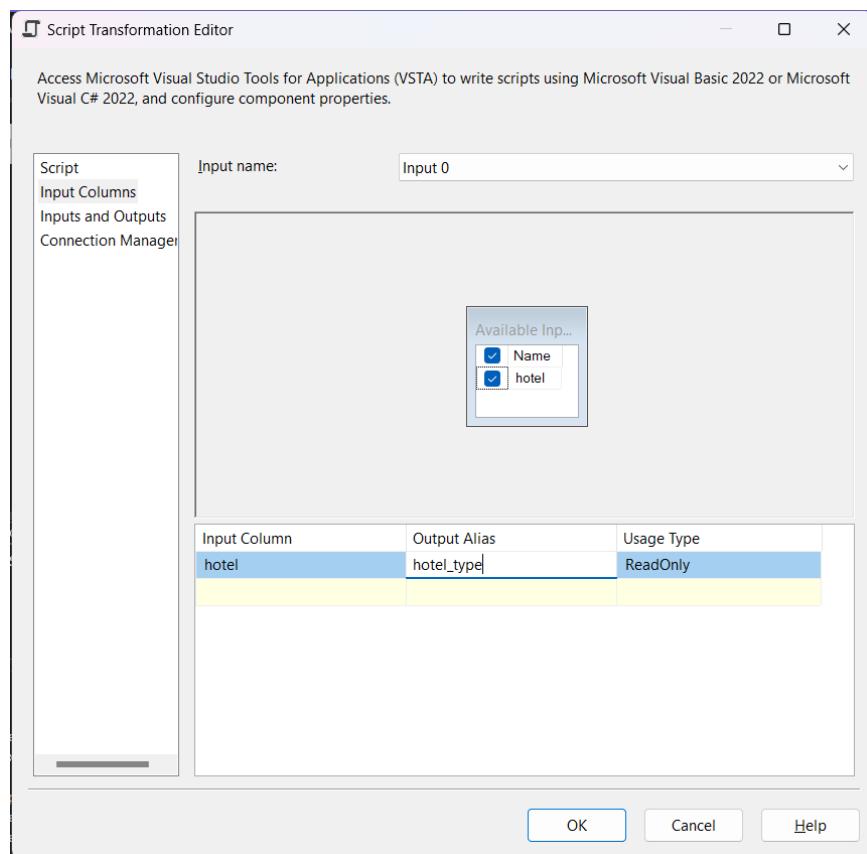
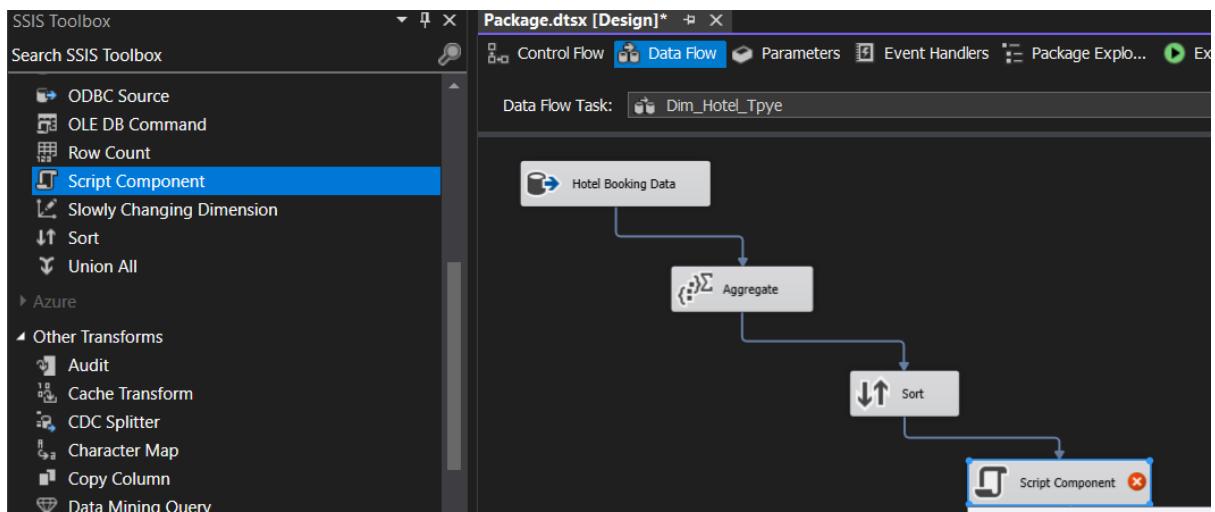
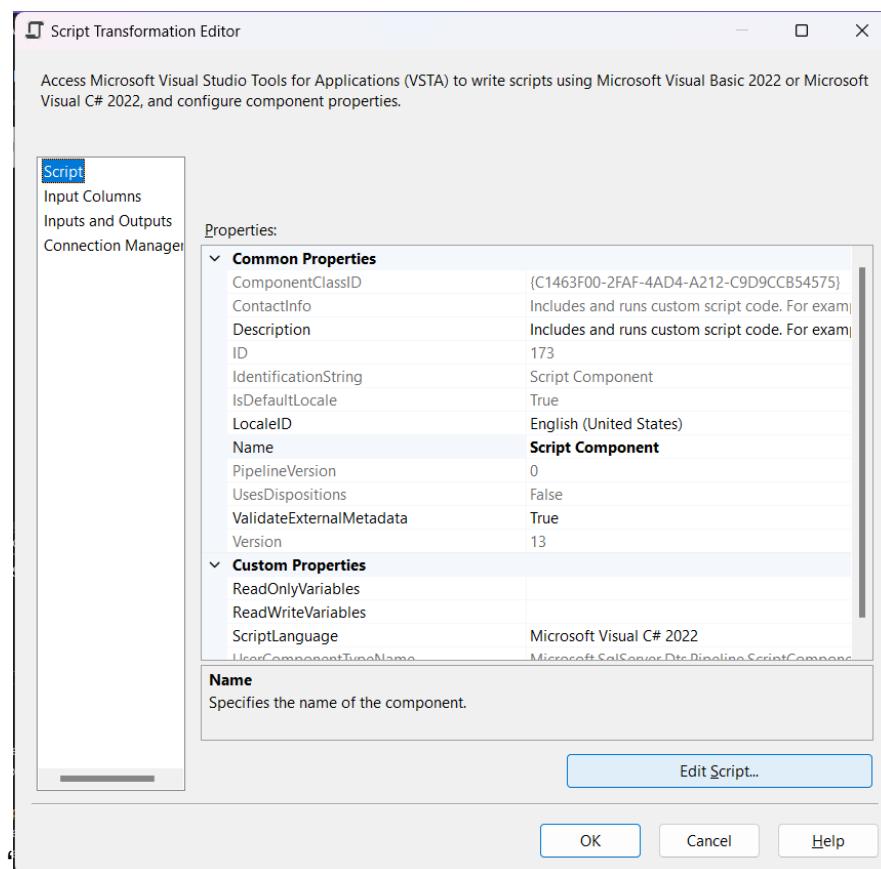
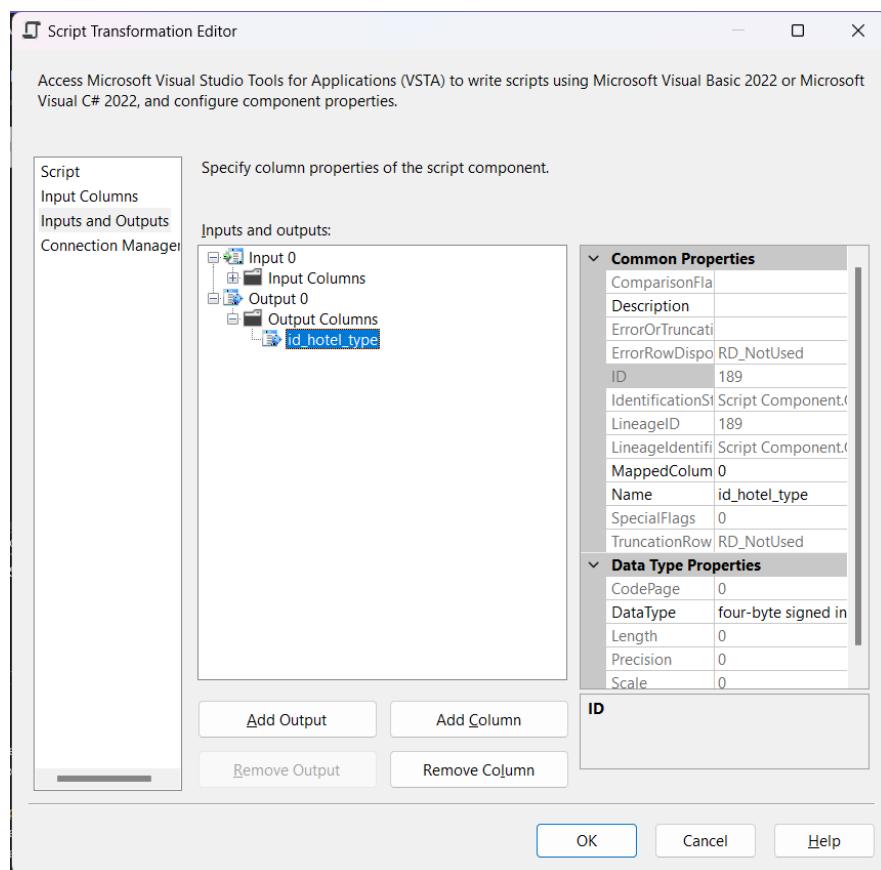


Figure 7-8. Sắp xếp các giá trị theo chiều tăng dần

- **Bước 7:** Kéo thả công cụ Script Component để tạo khóa chính id_hotel_type và đổi tên thuộc tính từ hotel sang hotel_type.





```

public class ScriptMain : UserComponent
{
    Help: Using Integration Services variables and parameters
    Help: Using Integration Services Connection Managers
    Help: Firing Integration Services Events
    int count = 1;
    /// <summary>
    /// This method is called once, before rows begin to be processed in the d
    ///
    /// You can remove this method if you don't need to do anything here.
    /// </summary>
    /// <param name="Row">The row that is currently passing through
    public override void Input0_ProcessInputRow(Input0Buffer Row)
    {
        Row.idhoteltypes = count;
        count++;
    }
}

```

Figure 9-10-11-12-13-14. Sử dụng Script Component để tạo khóa chính id_hotel_type và đổi tên thuộc tính từ hotel sang hotel_type

- **Bước 8:** Chọn công cụ OLE DB Destination để tiến hành tạo bảng trong cơ sở dữ liệu DB_HotelBooking với tên gọi là Dim_Hotel_Type.

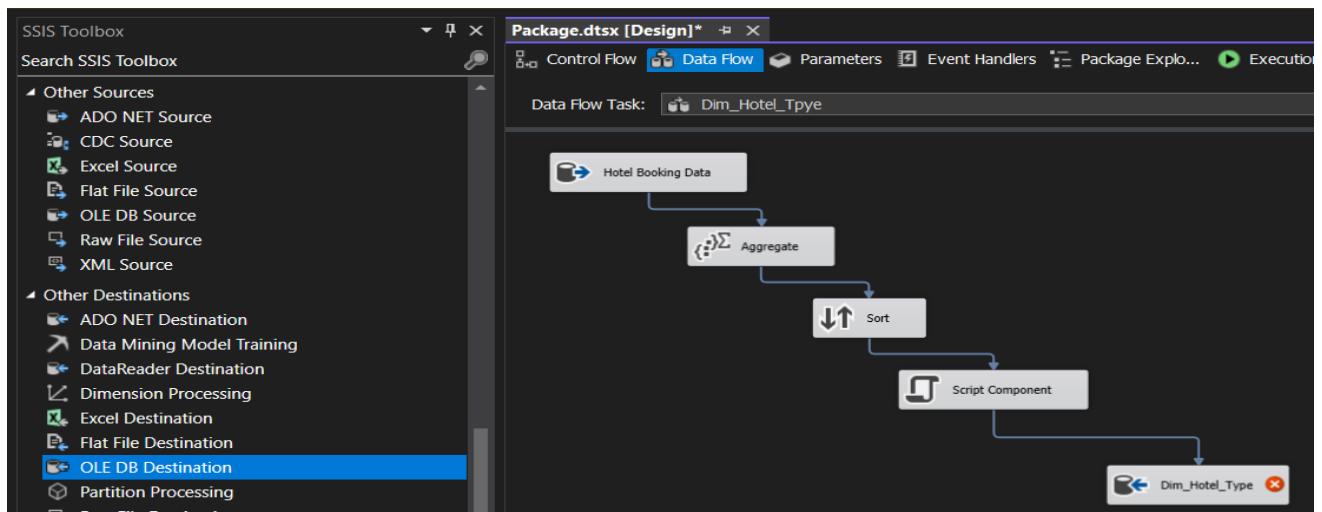


Figure 36. Sử dụng OLE DB Destination để tạo bảng

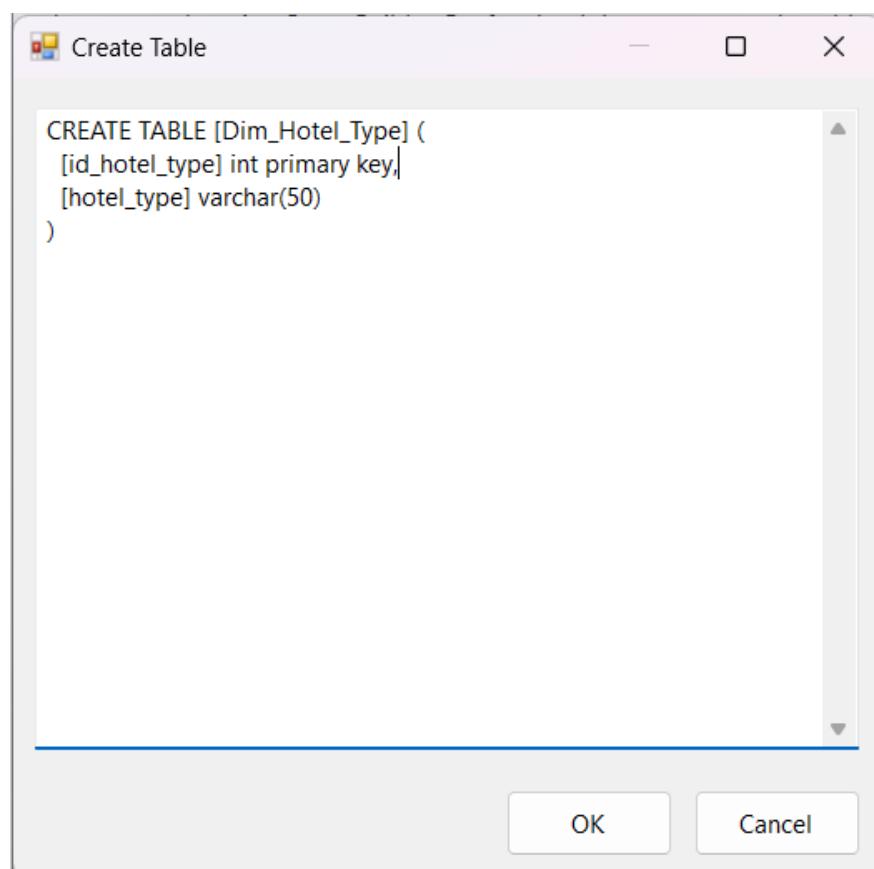
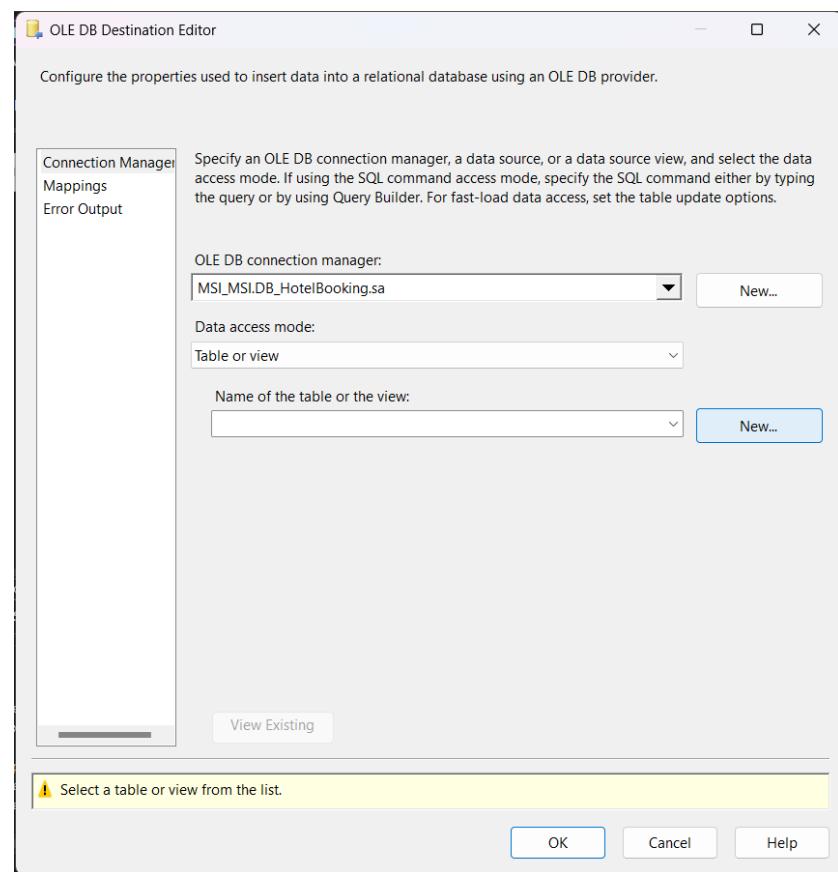


Figure 16-17. Tạo bảng Dim_Hotel_Type

- **Bước 9:** Nhấn Mappings để kiểm tra kết nối và nhấn Ok để hoàn tất quá trình tạo bảng.

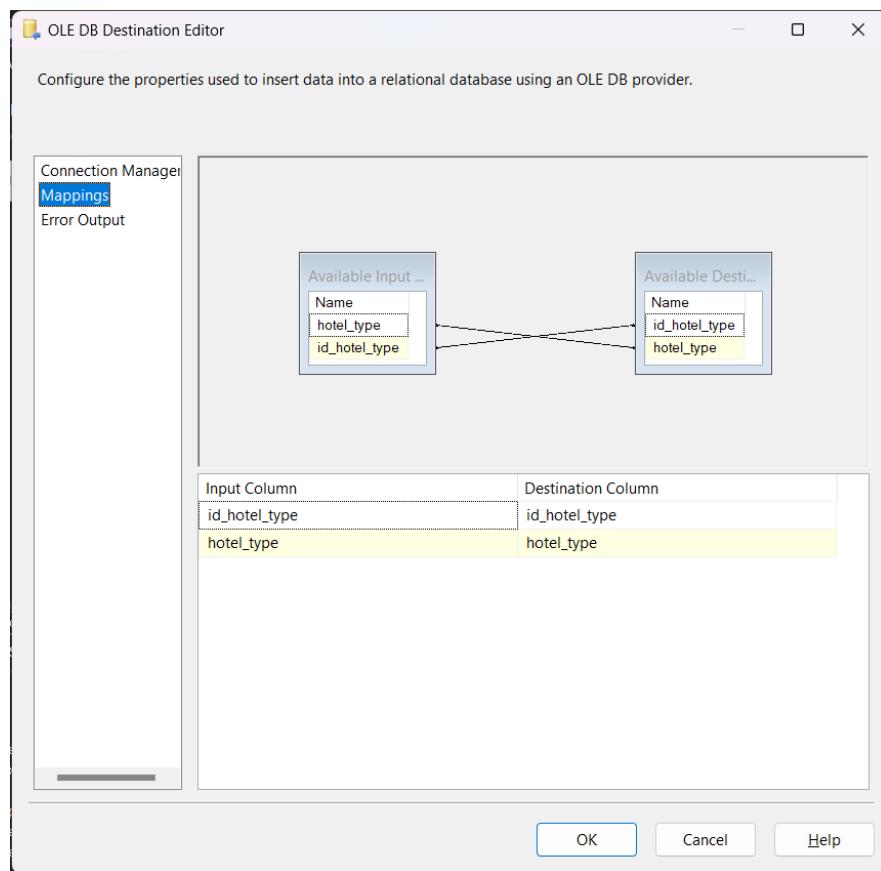


Figure 37. Quá trình Mappings dữ liệu

- **Bước 10:** Chọn Start để bắt đầu quá trình đổ dữ liệu. Khi đổ thành công, ta có kết quả như sau:

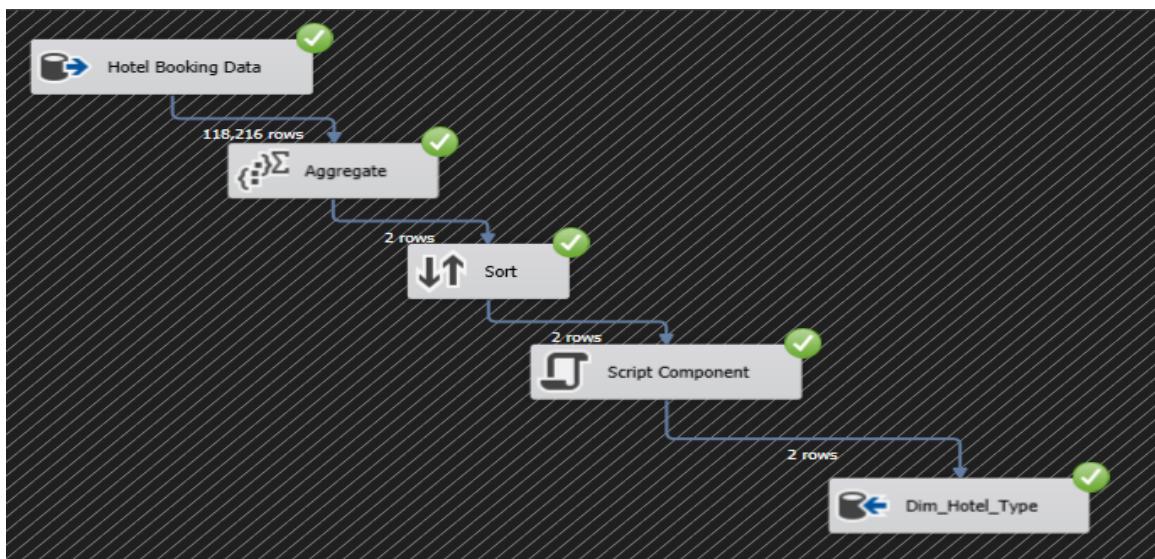


Figure 38. Hoàn thành đổ dữ liệu vào HotelType trong kho dữ liệu

- **Bước 11:** Kiểm tra bảng Dim_Hotel_Type trên SQL Server.

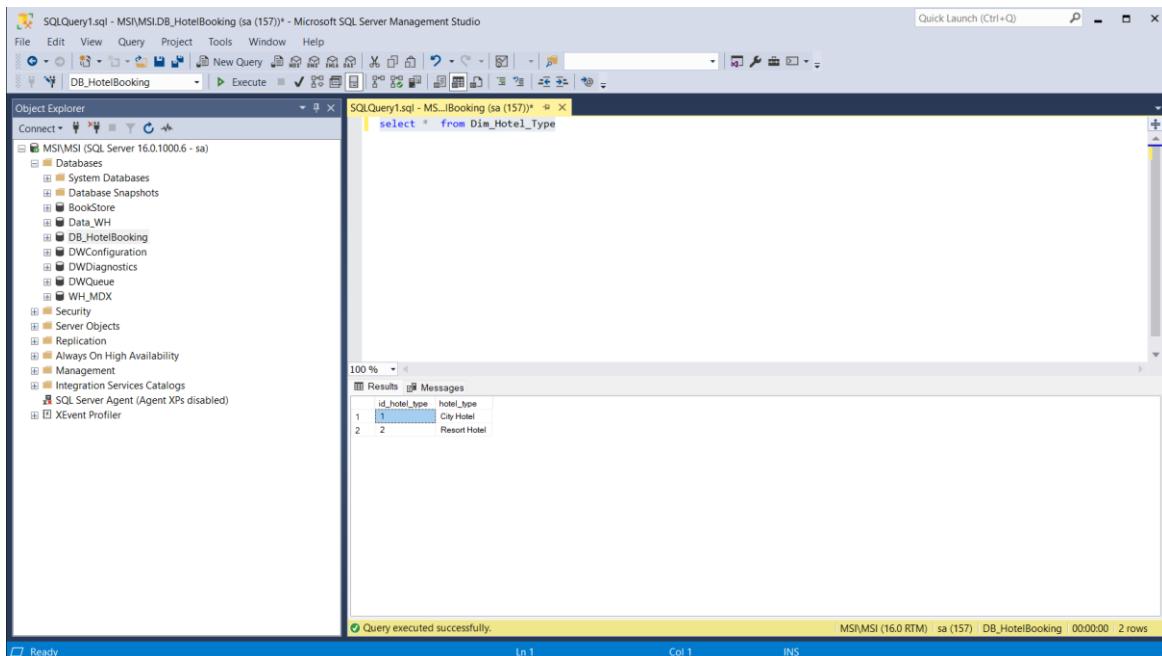


Figure 39. Kiểm tra bảng Dim_Hotel_Type trong SQL Server

2.5.2. Bảng Dim_Reservation_Status

- **Bước 1:** Kéo chức năng Data Flow Task từ cột trái sang màn hình làm việc và đổi tên thành Dim_Reservation_Status.

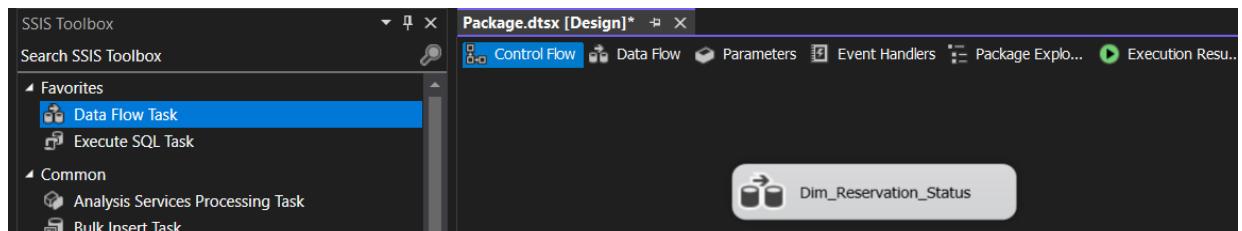


Figure 40. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Reservation_Status

- **Bước 2:** Nhấn double vào Data Flow Task vừa tạo và tìm kiếm chức năng OLE DB Source, kéo vào màn hình làm việc

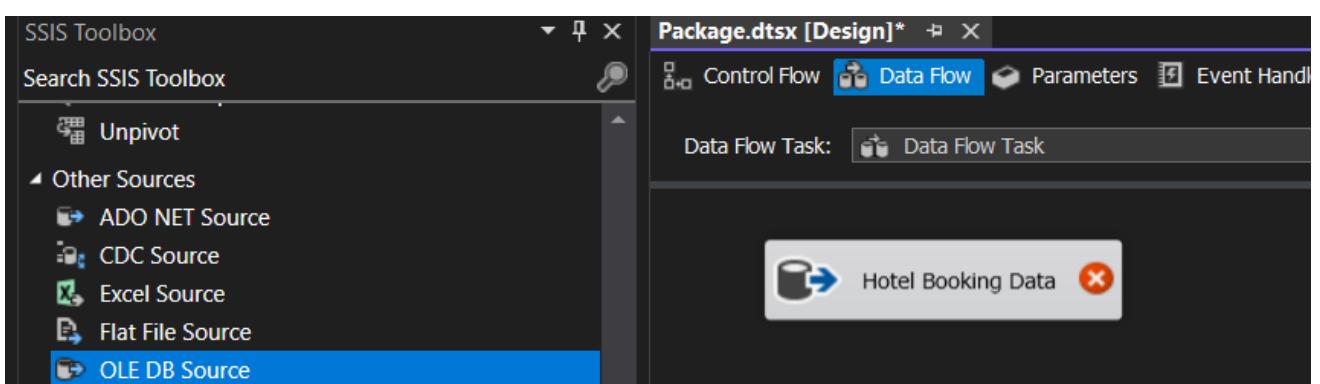


Figure 41. Kéo chức năng OLE DB Source vào màn hình làm việc

- **Bước 3:** Click double vào OLE DB Source và dẫn đến DB_HotelBooking. Sau đó chọn Name of the table or the view là [dbo].Hotel_Booking.

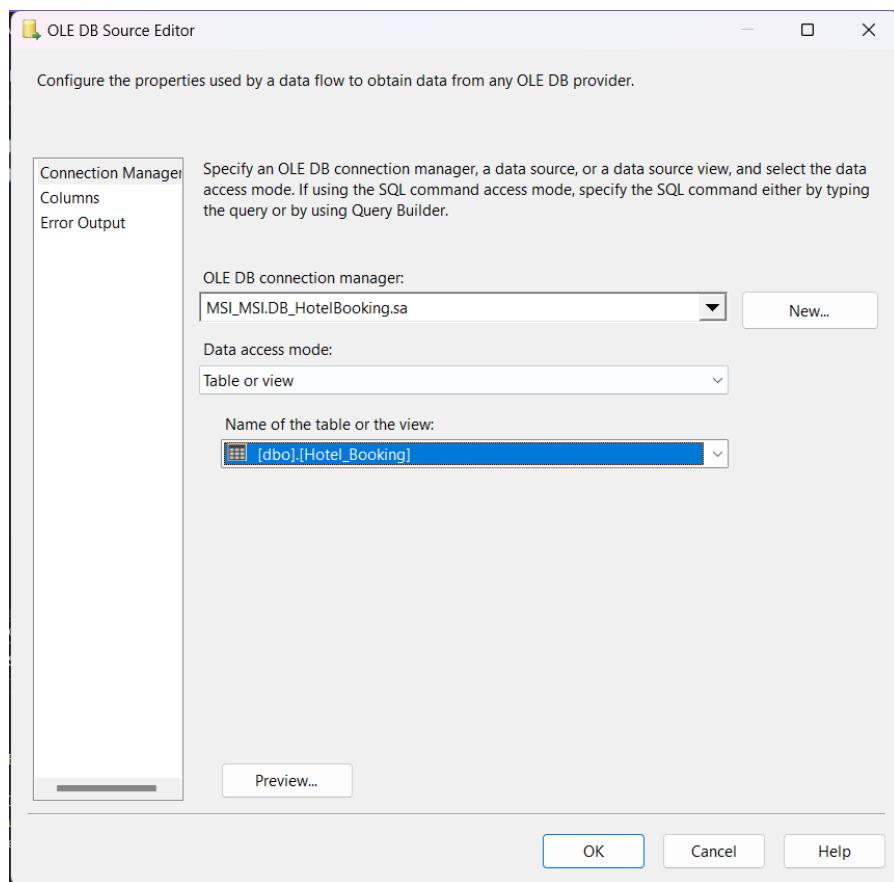


Figure 42. Chọn Table sẽ lấy dữ liệu

- **Bước 4:** Chọn Columns để lựa chọn các cột cần sử dụng và nhấn OK

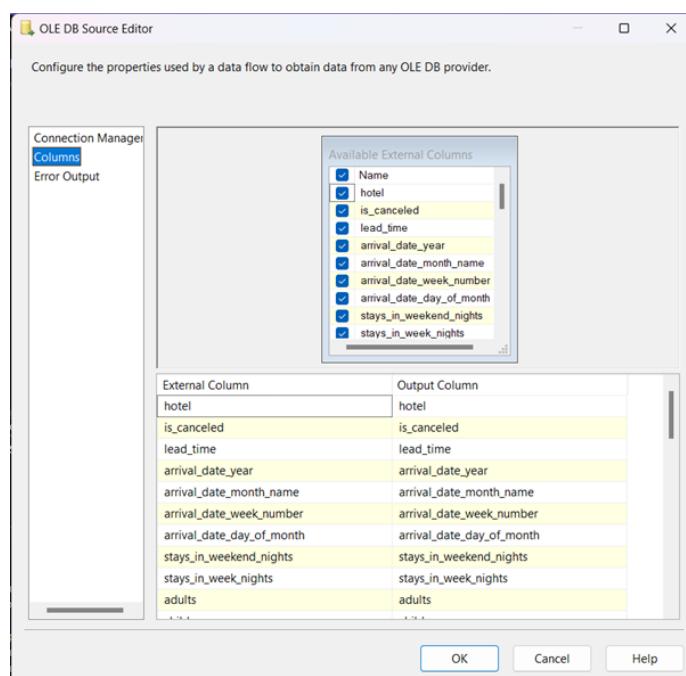


Figure 43. Chọn các column cần sử dụng

- **Bước 5:** Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau lên thuộc tính sử dụng là reservation_status.

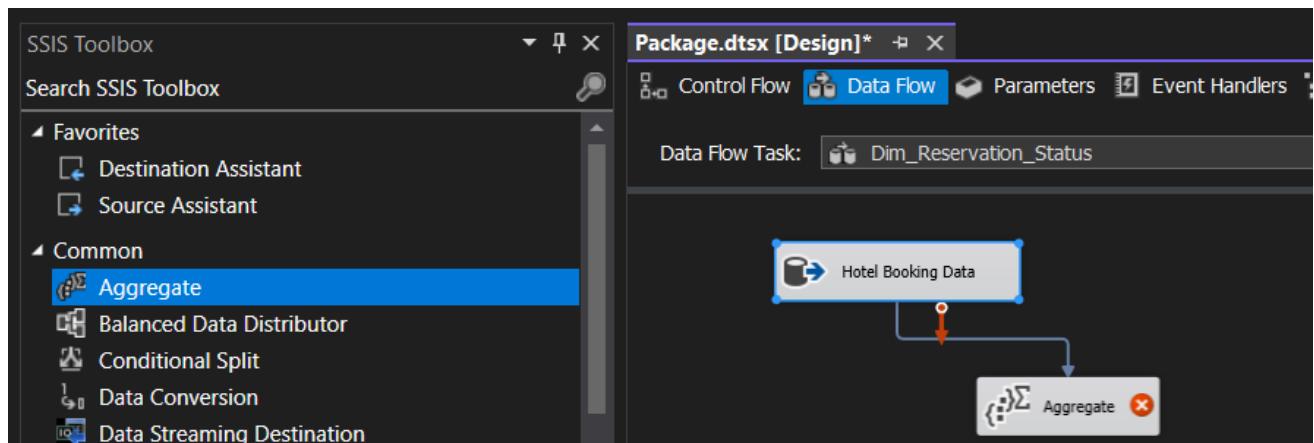


Figure 44. Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau

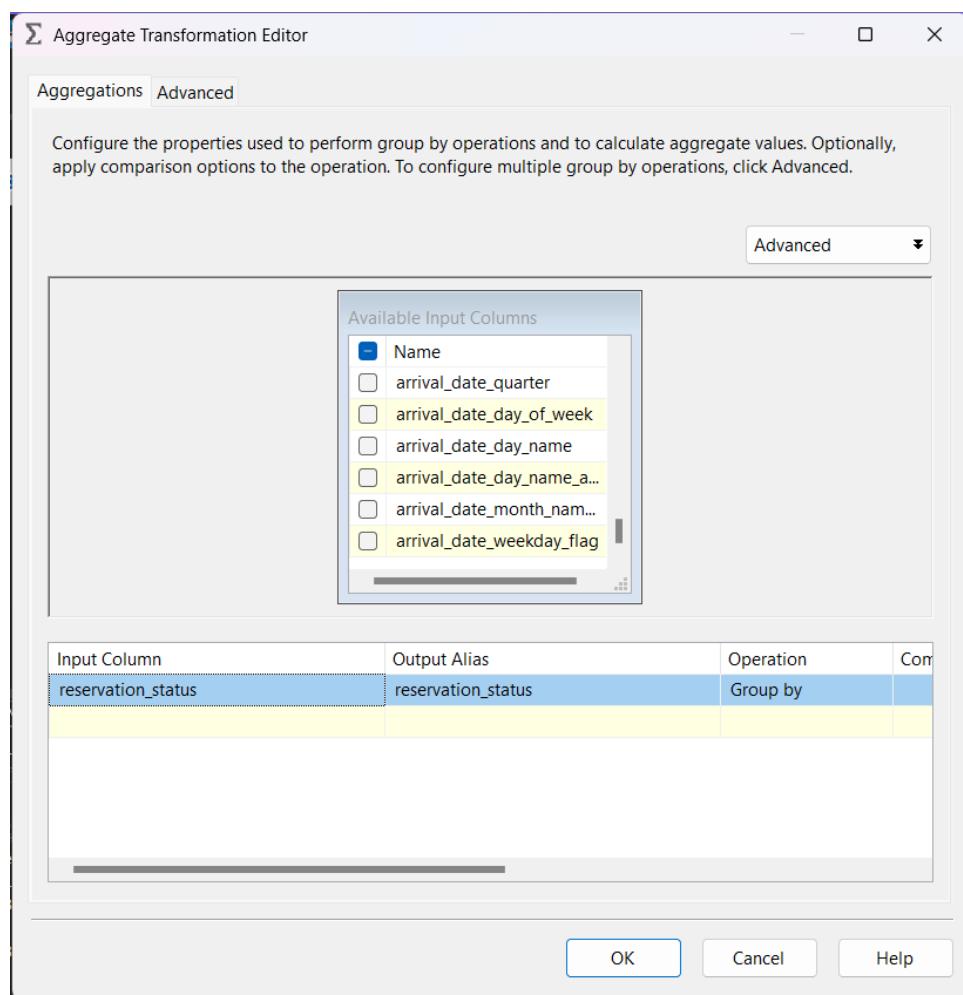


Figure 45. Chọn các thuộc tính cần gom nhóm

- **Bước 6:** Dùng công cụ Sort để sắp xếp các giá trị theo chiều tăng dần.

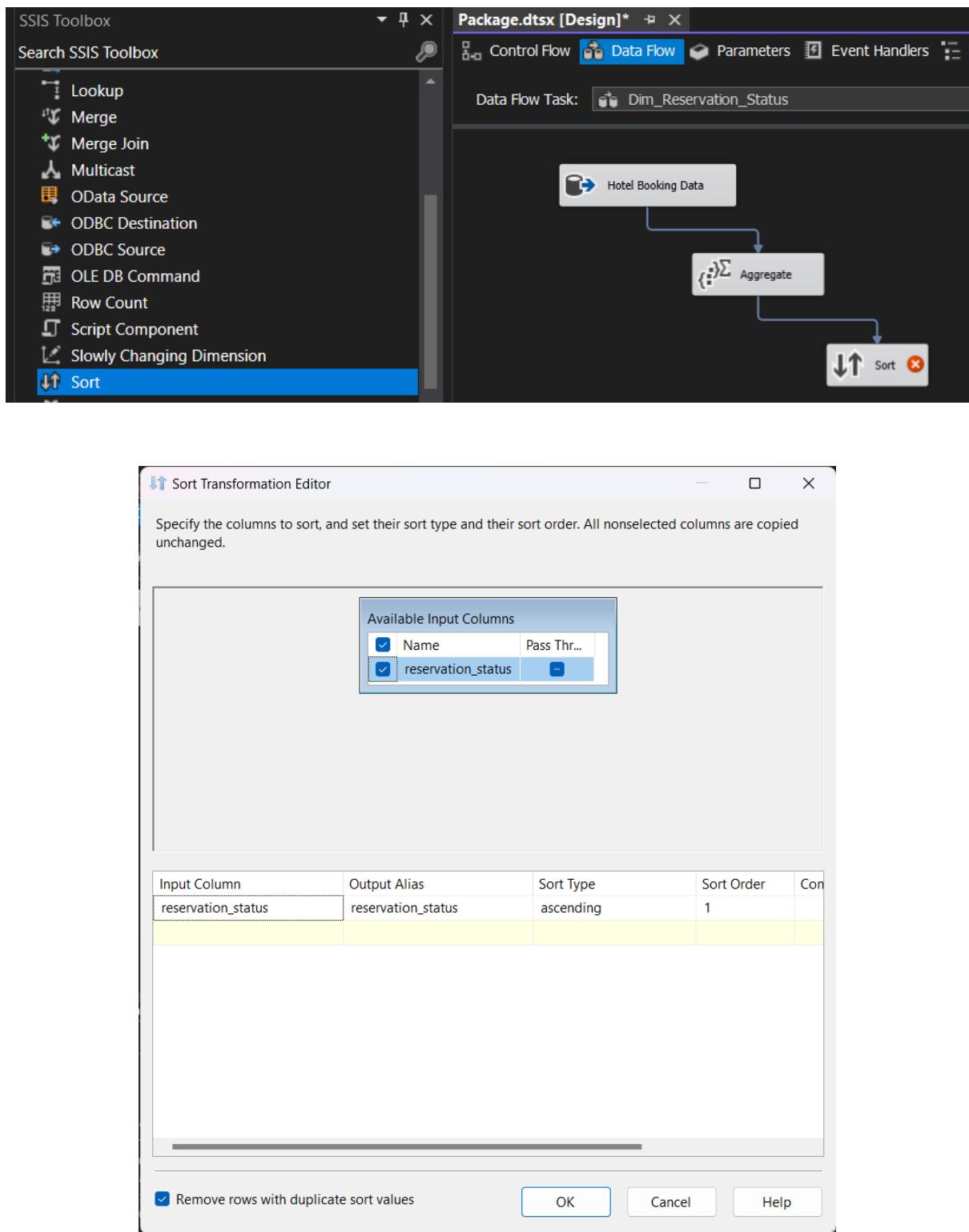
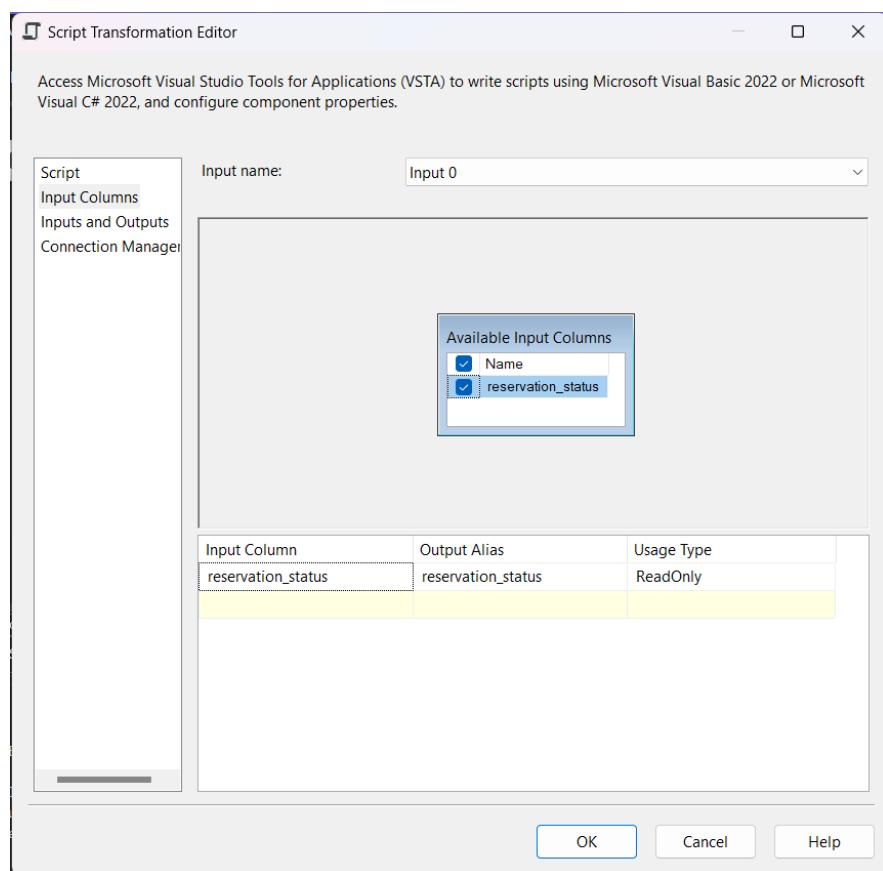
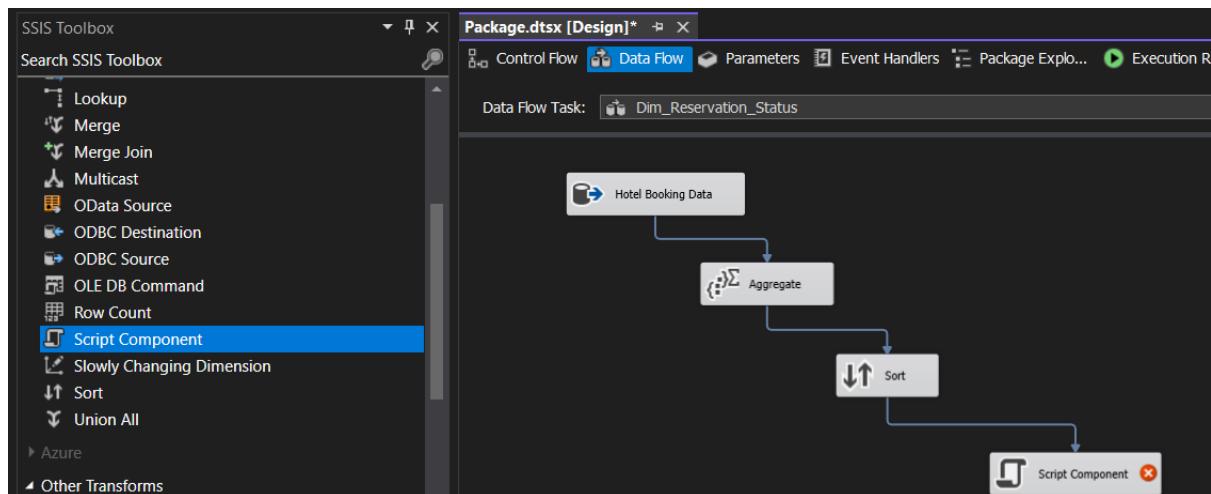
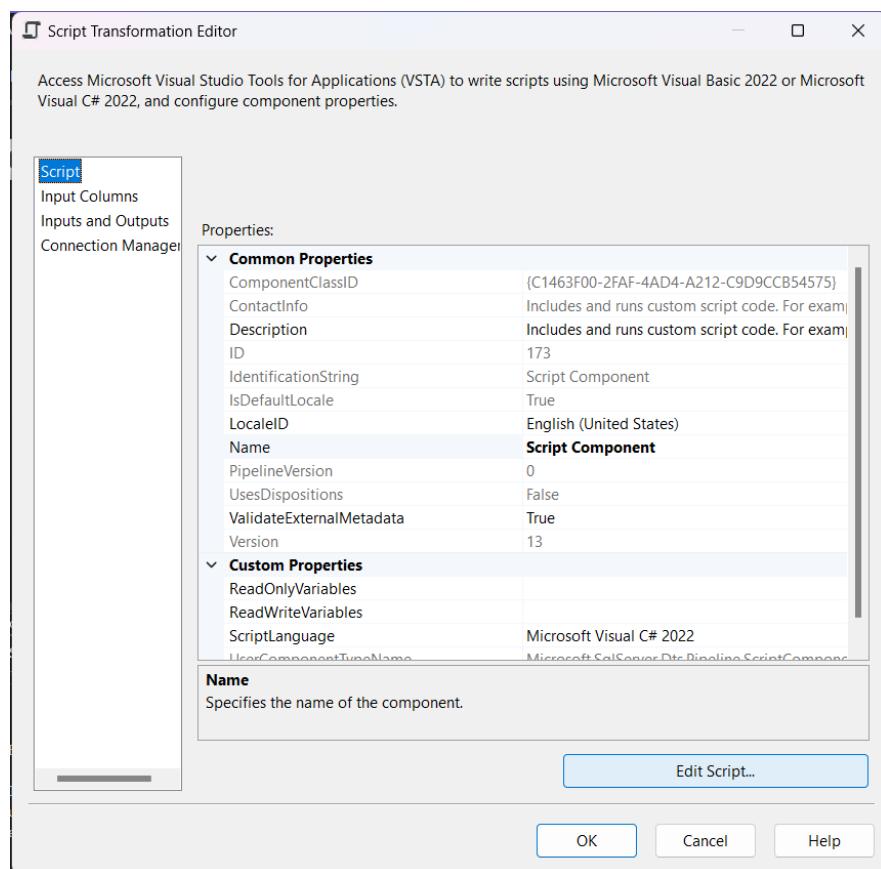
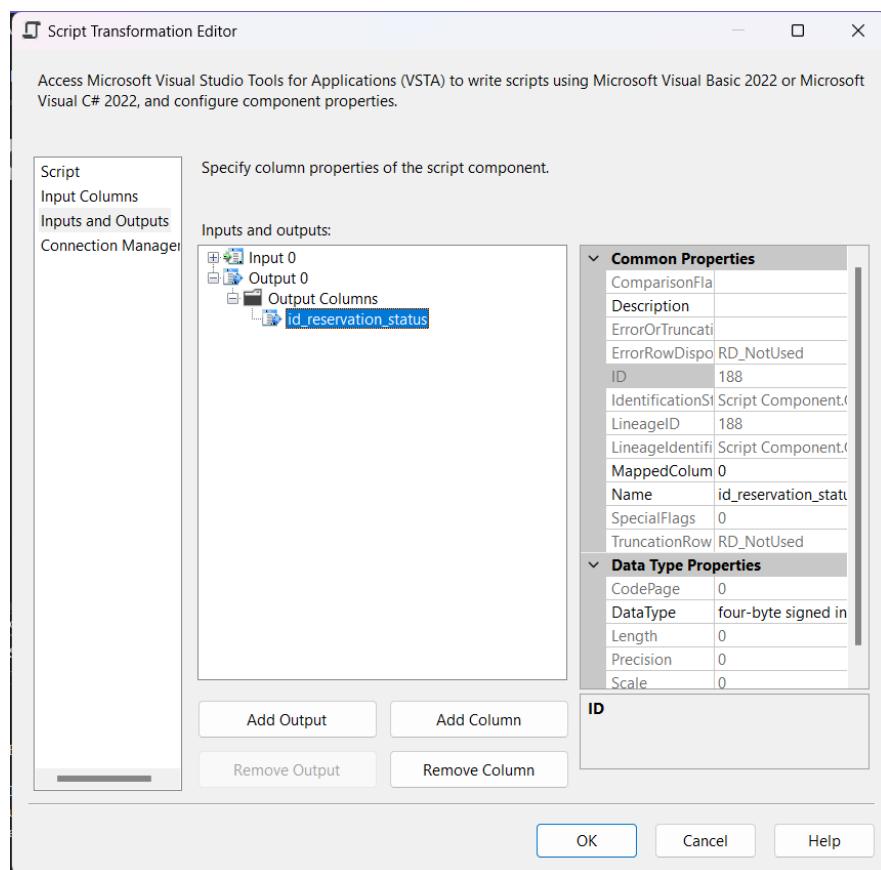


Figure 7-8. Sắp xếp các giá trị theo chiều tăng dần

- **Bước 7:** Kéo thả công cụ Script Component để tạo khóa chính id_reservation_status.





```

public class ScriptMain : UserComponent
{
    Help: Using Integration Services variables and parameters
    Help: Using Integration Services Connection Managers
    Help: Firing Integration Services Events
    int count = 1;
    /// <summary>
    /// This method is called once, before rows begin to be processed
    ///
    /// You can remove this method if you don't need to do anything here
    /// </summary>
    0 references

    public override void OnPreExecute()
    {
    }

    public override void Input0_ProcessInputRow(Input0Buffer Row)
    {
        Row.idreservationstatus = count;
        count++;
    }
}

```

Figure 9-10-11-12-13-14. Sử dụng Script Component để tạo khóa chính id_reservation_status

- **Bước 8:** Chọn công cụ OLE DB Destination để tiến hành tạo bảng trong cơ sở dữ liệu DB_HotelBooking với tên gọi là Dim_Reservation_Status.

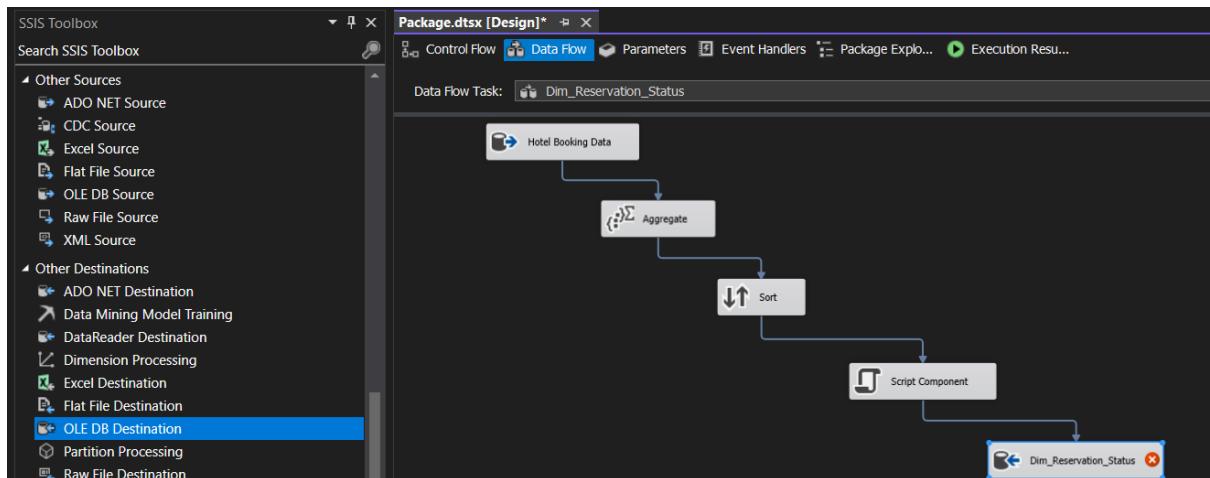


Figure 46. Sử dụng OLE DB Destination để tạo bảng

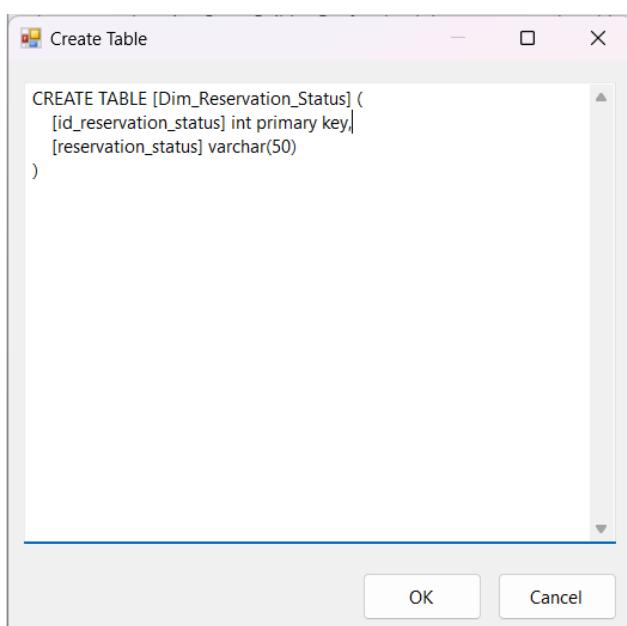


Figure 47. Tạo bảng Dim_Reservation_Status

- **Bước 9:** Nhấn Mappings để kiểm tra kết nối và nhấn Ok để hoàn tất quá trình tạo bảng.

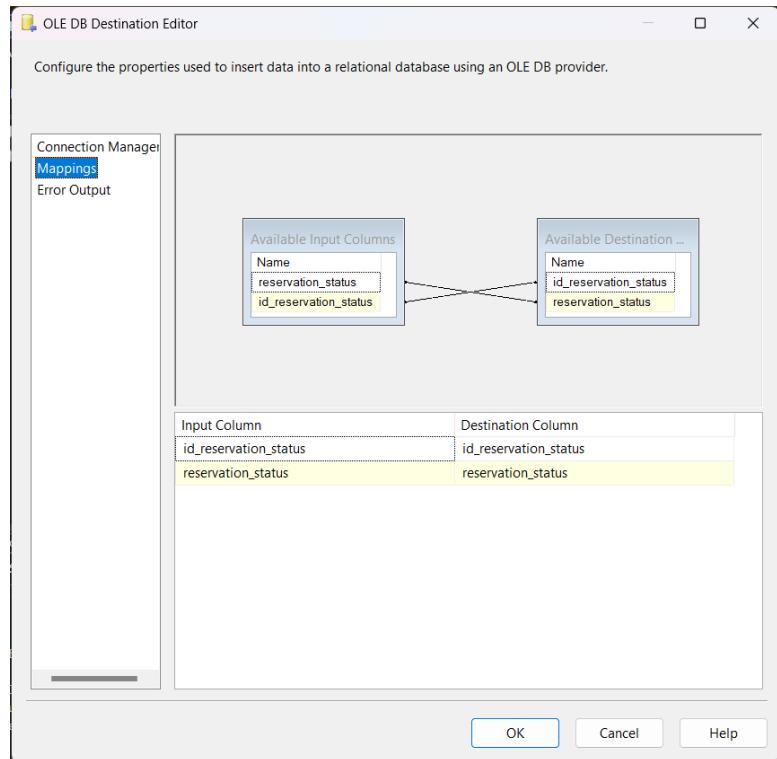


Figure 48. Quá trình Mappings dữ liệu

- **Bước 10:** Chọn Start để bắt đầu quá trình đổ dữ liệu. Khi đổ thành công, ta có kết quả như sau:

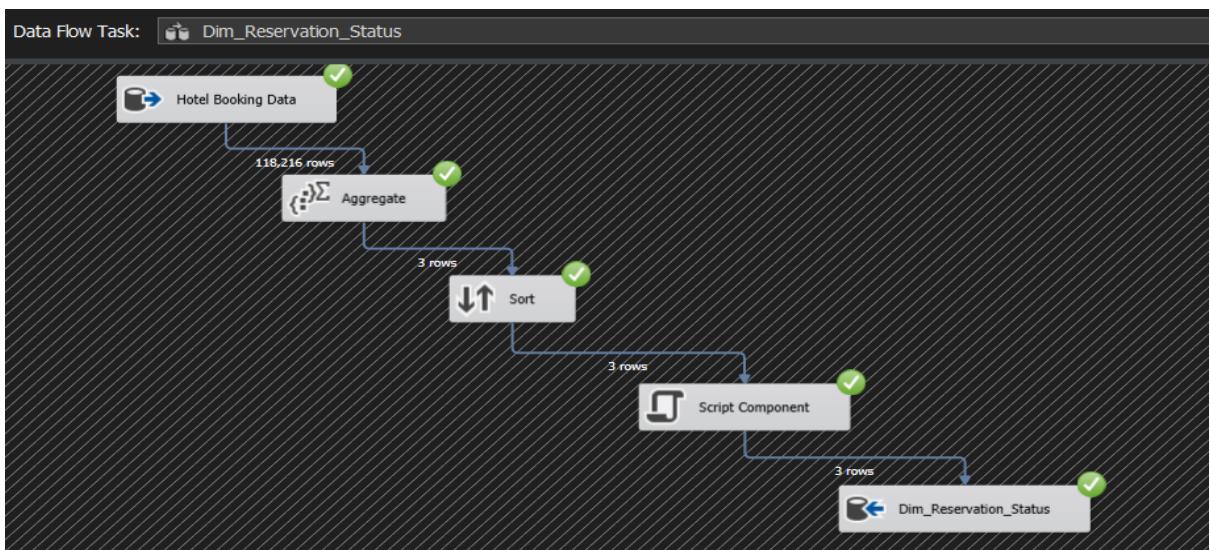


Figure 49. Hoàn thành đổ dữ liệu vào Dim_Reservation_Status trong kho dữ liệu

- **Bước 11:** Kiểm tra bảng Dim_Reservation_Status trên SQL Server.

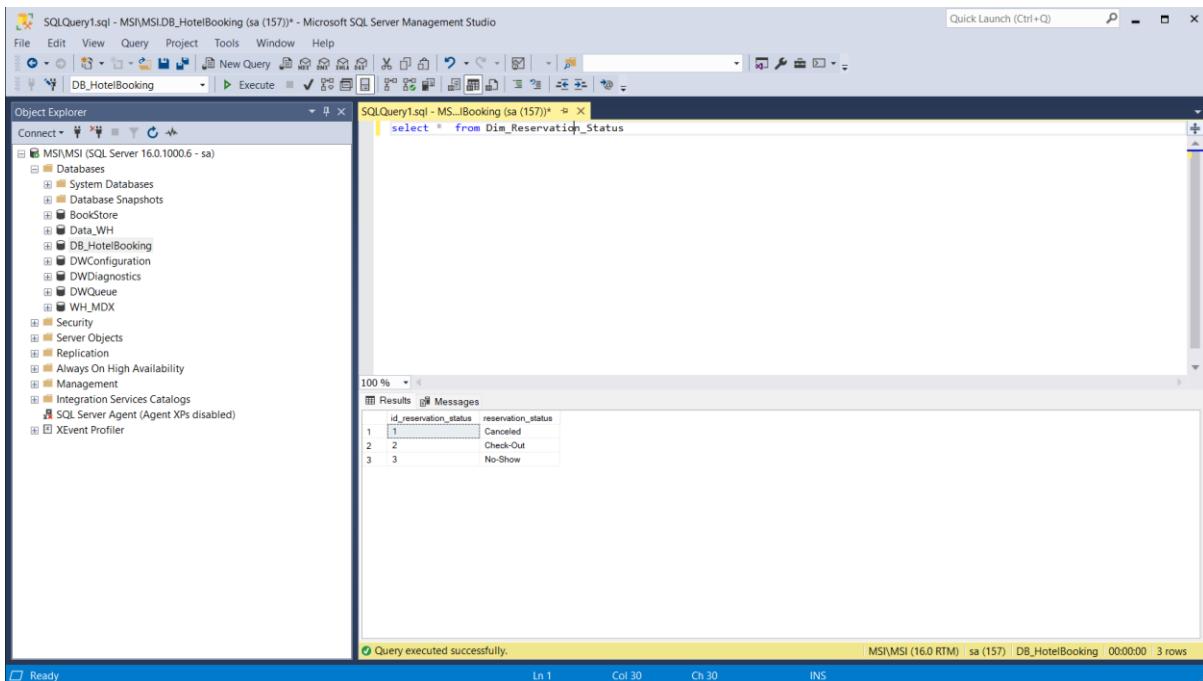


Figure 50. Kiểm tra bảng Dim_Reservation_Status trong SQL Server

2.5.3. Bảng Dim_Deposit_Type

- **Bước 1:** Kéo chức năng Data Flow Task từ cột trái sang màn hình làm việc và đổi tên thành DimDistributionChannel.

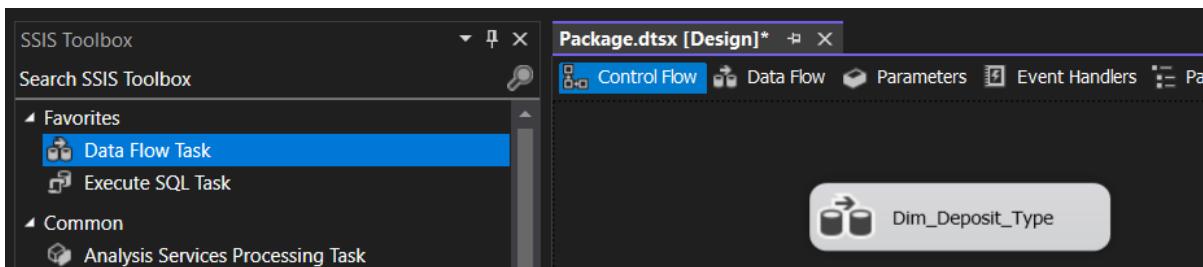


Figure 51. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Deposit_Type

- **Bước 2:** Nhấn double vào Data Flow Task vừa tạo và tìm kiếm chức năng OLE DB Source, kéo vào màn hình làm việc.

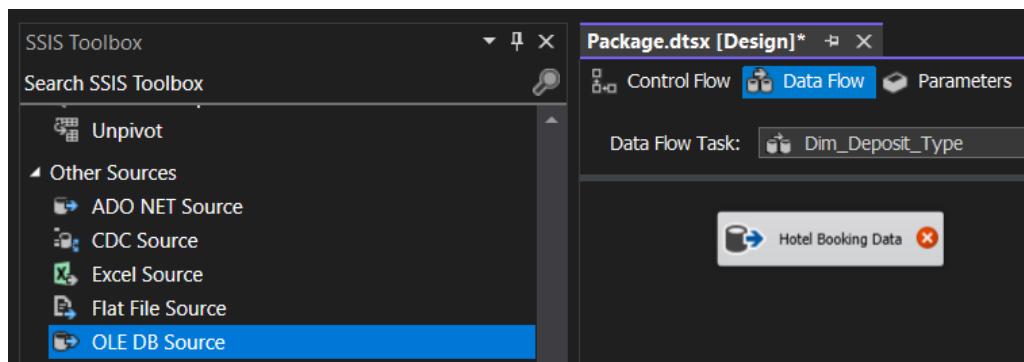


Figure 52. Kéo chức năng OLE DB Source vào màn hình làm việc

- **Bước 3:** Click double vào OLE DB Source và dẫn đến DB_HotelBooking. Sau đó chọn Name of the table or the view là [dbo].Hotel_Booking.

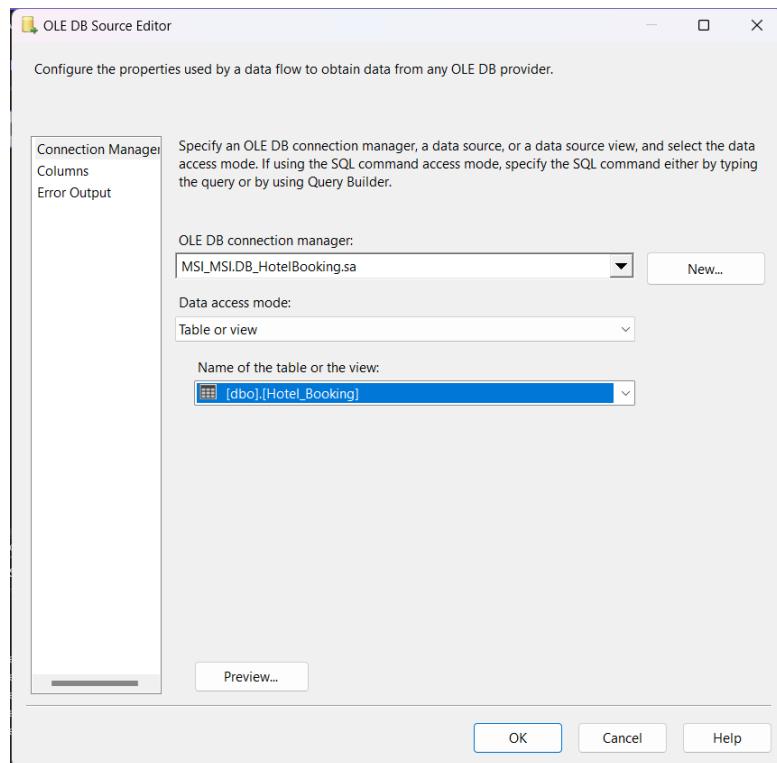


Figure 53. Chọn Table sẽ lấy dữ liệu

- **Bước 4:** Chọn Columns để lựa chọn các cột cần sử dụng và nhấn OK.

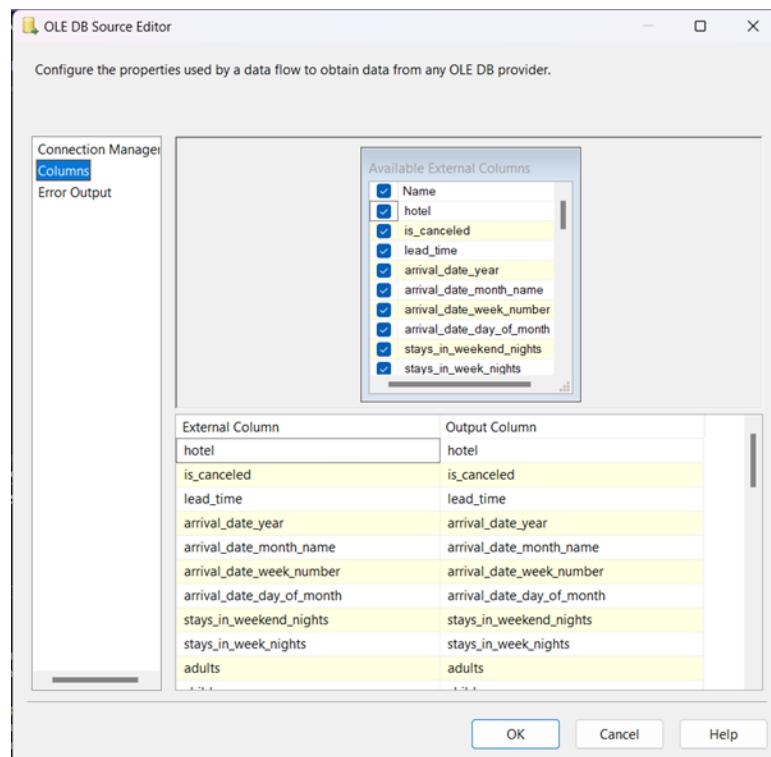


Figure 54. Chọn các column cần sử dụng

- **Bước 5:** Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau lên thuộc tính sử dụng là deposit_type.

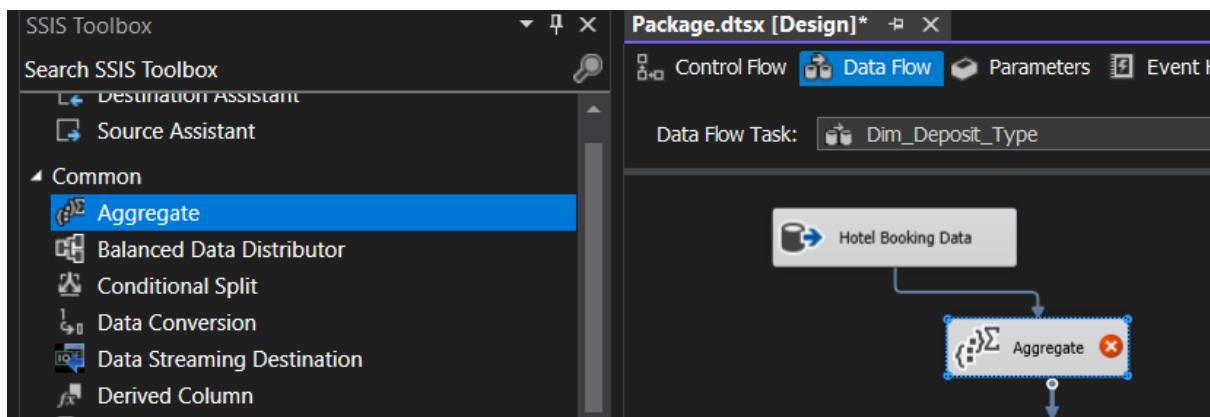


Figure 55. Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau

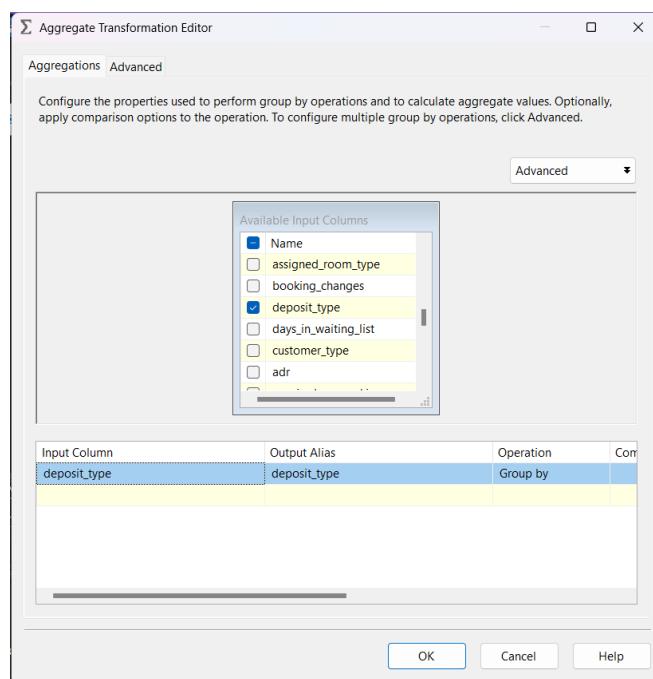
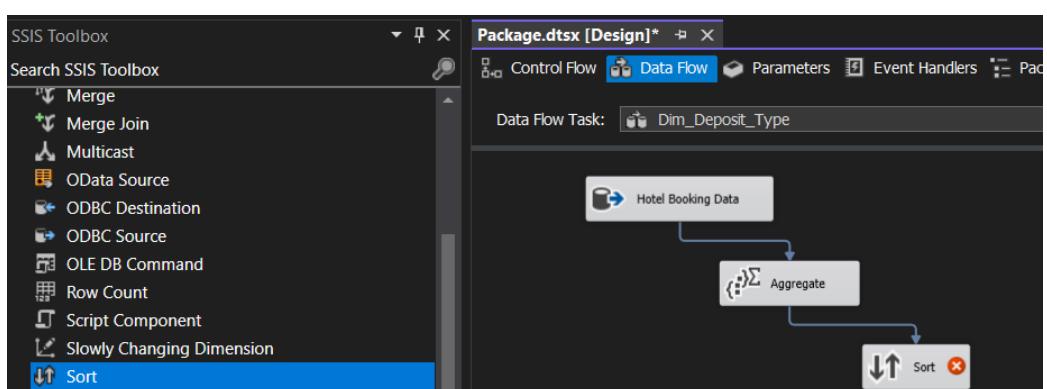


Figure 56. Chọn các thuộc tính cần gom nhóm

- **Bước 6:** Dùng công cụ Sort để sắp xếp các giá trị theo chiều tăng dần.



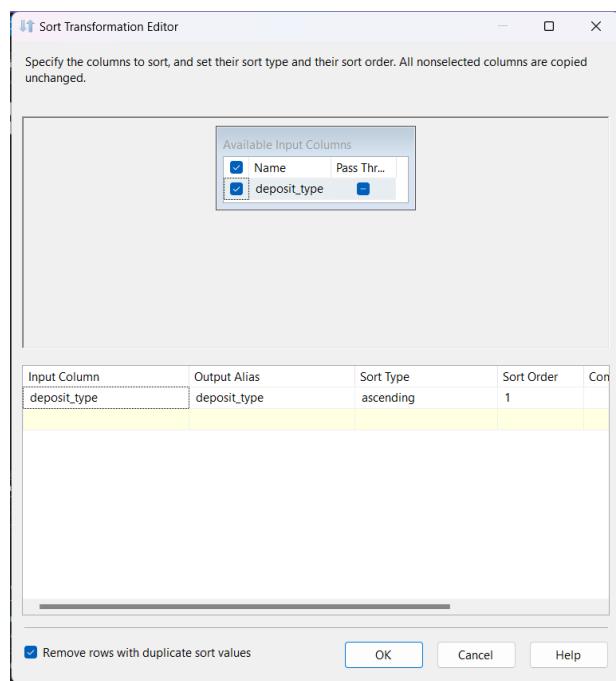
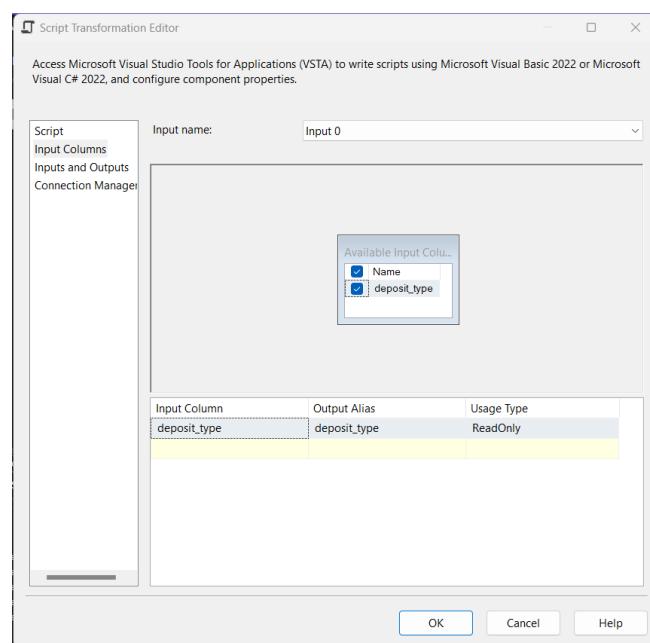
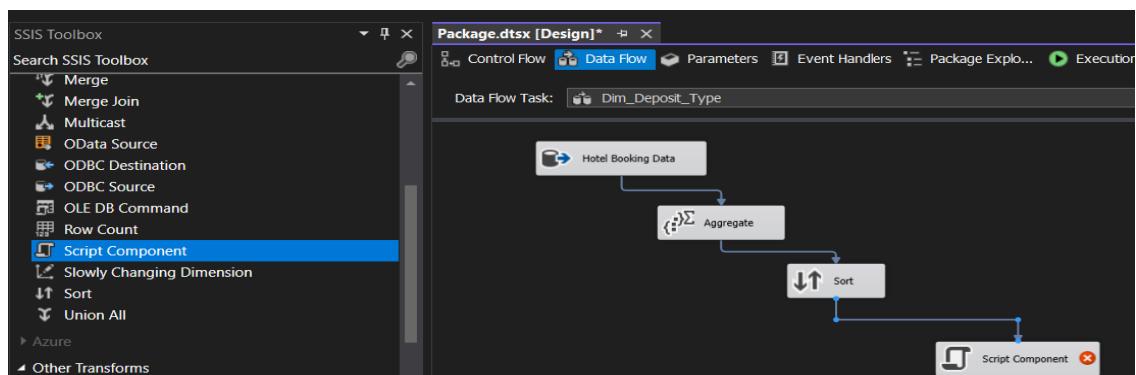
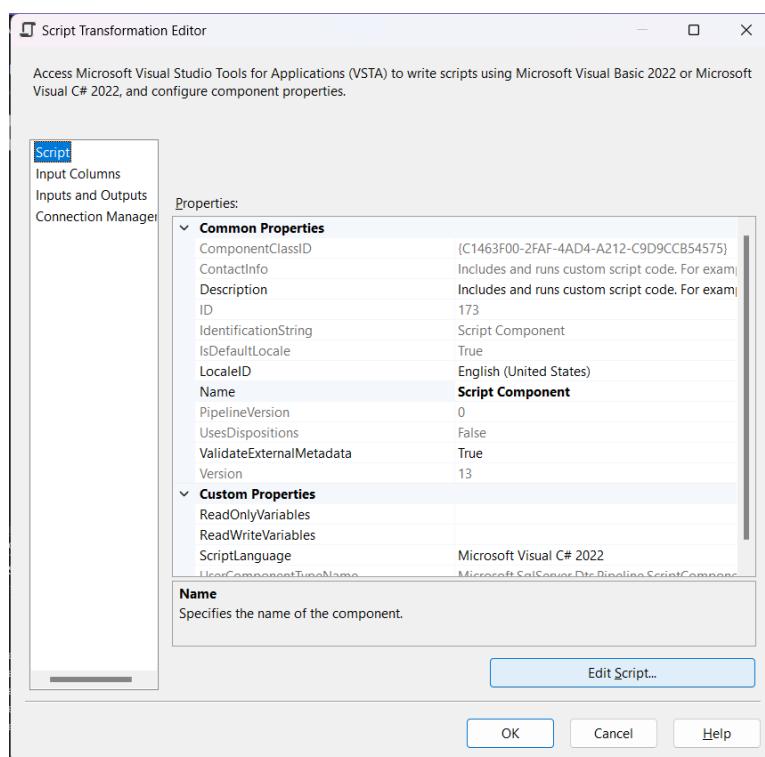
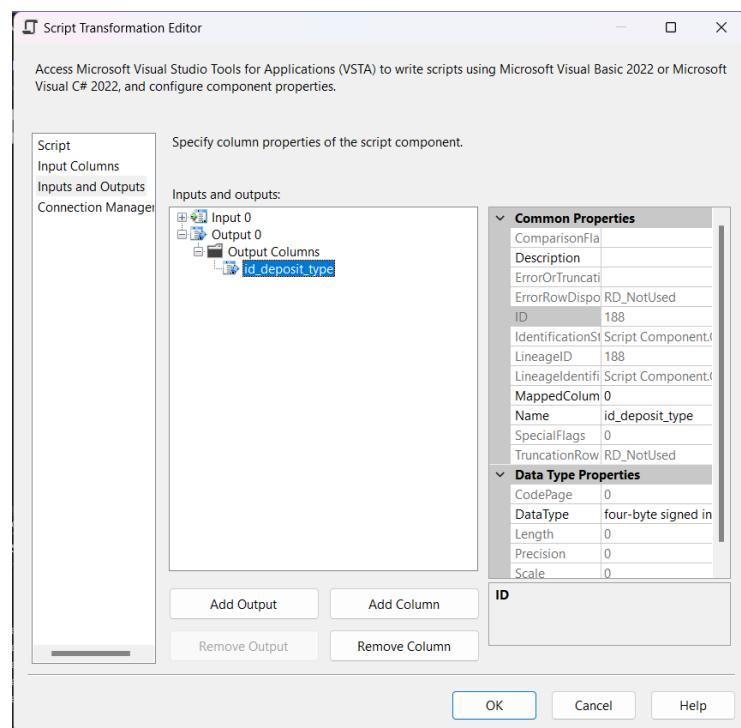


Figure 7-8. Sắp xếp các giá trị theo chiều tăng dần

- **Bước 7:** Kéo thả công cụ Script Component để tạo khóa chính id_deposit_type.



IS217.N21 – Kho dữ liệu và phân tích hoạt động đặt phòng khách sạn



```
/// // partial name = Row -> the row that is currently passing through a component
public override void Input0_ProcessInputRow(Input0Buffer Row)
{
    Row.iddeposittype = count;
    count++;
}
```

Figure 9-10-11-12-13. Sử dụng Script Component để tạo khóa chính id_deposit_type

- **Bước 8:** Chọn công cụ OLE DB Destination để tiến hành tạo bảng trong cơ sở dữ liệu DB_HotelBooking với tên gọi là Dim_Deposit_Type.

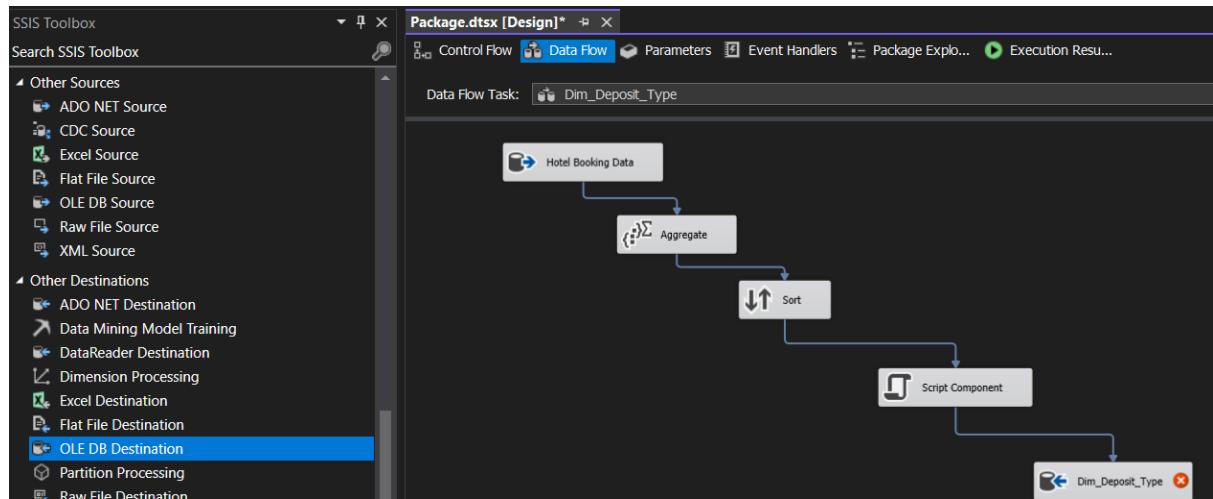


Figure 57. Sử dụng OLE DB Destination để tạo bảng

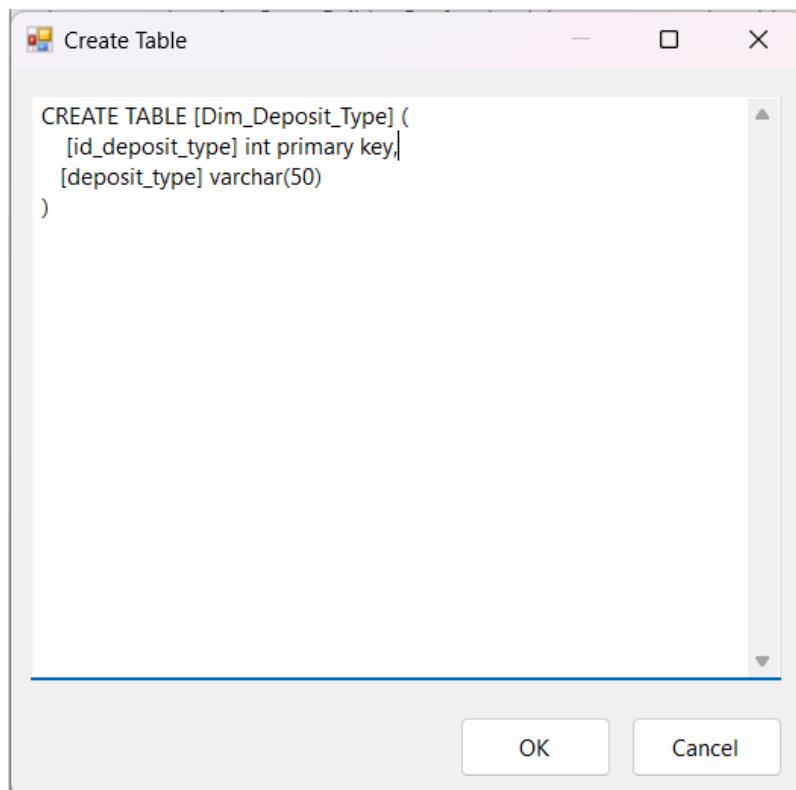


Figure 58. Tạo bảng Dim_Deposit_Type

- **Bước 9:** Nhấn Mappings để kiểm tra kết nối và nhấn Ok để hoàn tất quá trình tạo bảng.

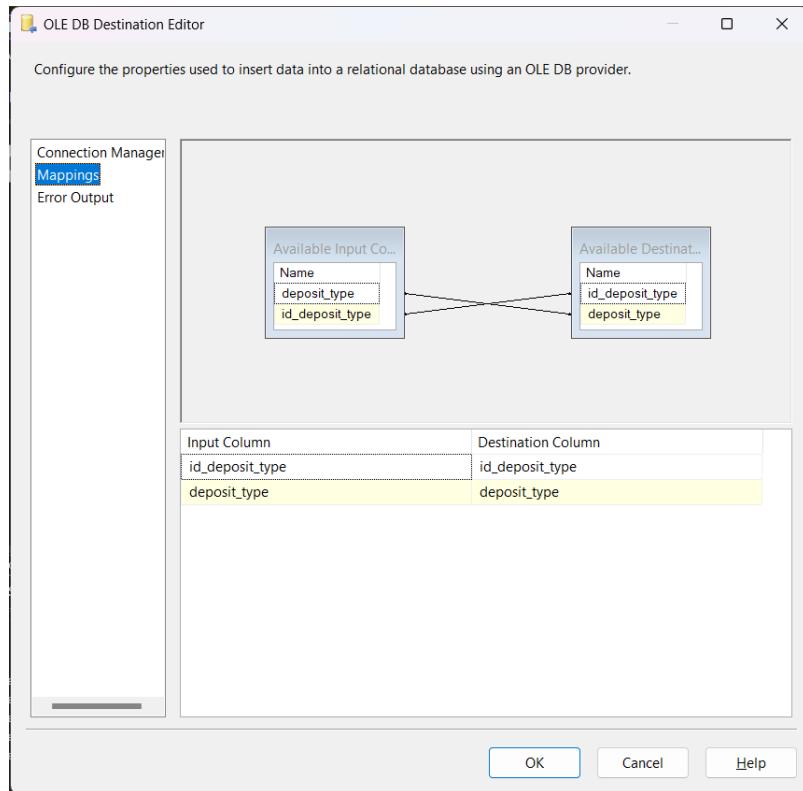


Figure 59. Quá trình Mappings dữ liệu

- **Bước 10:** Chọn Start để bắt đầu quá trình đổ dữ liệu. Khi đổ thành công, ta có kết quả như sau:

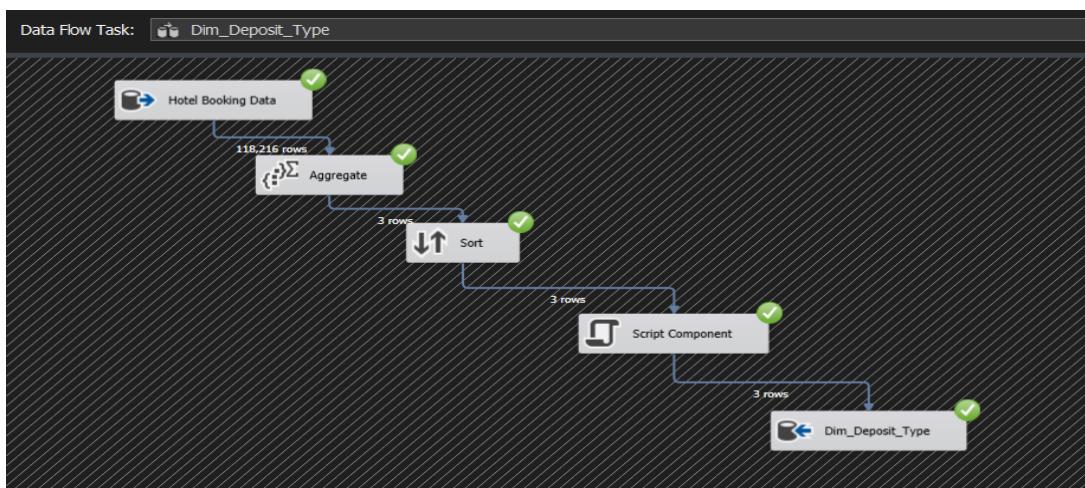


Figure 60. Hoàn thành đổ dữ liệu vào Dim_Deposit_Type trong kho dữ liệu

- **Bước 11:** Kiểm tra bảng Dim_Deposit_Type trên SQL Server.

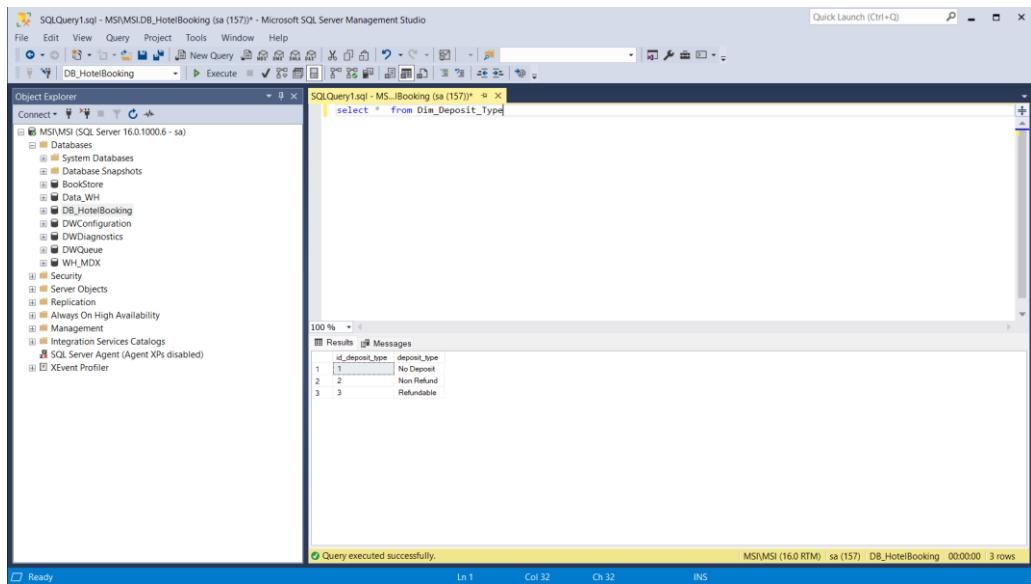


Figure 61. Kiểm tra bảng Dim_Deposit_Type trong SQL Server

2.5.4. Bảng Dim_Distribution_Channel

- **Bước 1:** Kéo chức năng Data Flow Task từ cột trái sang màn hình làm việc và đổi tên thành DimDistributionChannel.

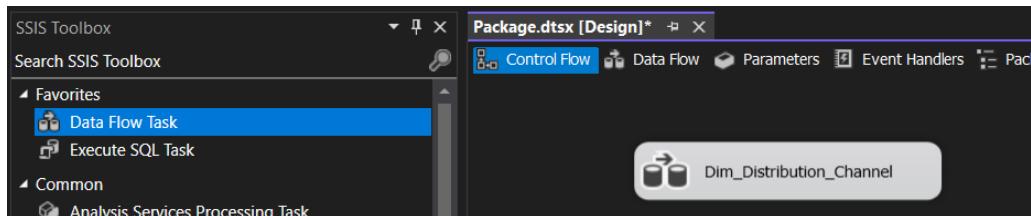


Figure 62. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Distribution_Channel

- **Bước 2:** Nhấn double vào Data Flow Task vừa tạo và tìm kiếm chức năng OLE DB Source, kéo vào màn hình làm việc.

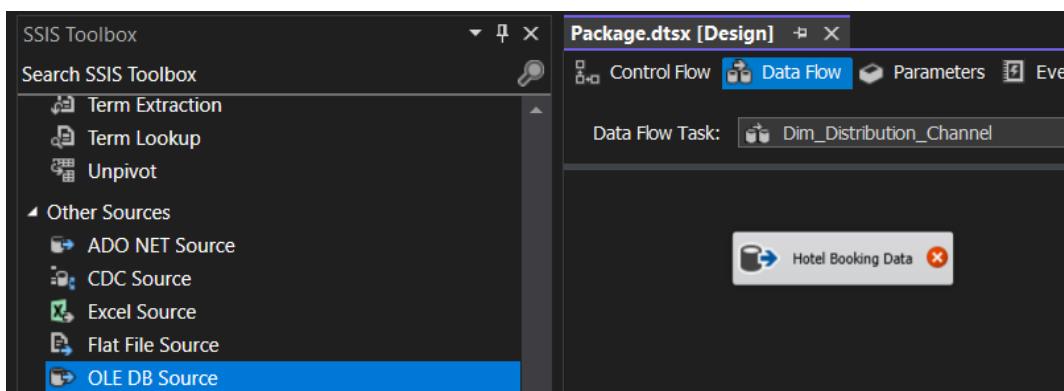


Figure 63. Kéo chức năng OLE DB Source vào màn hình làm việc

- **Bước 3:** Click double vào OLE DB Source và dẫn đến DB_HotelBooking. Sau đó chọn Name of the table or the view là [dbo].Hotel_Booking.

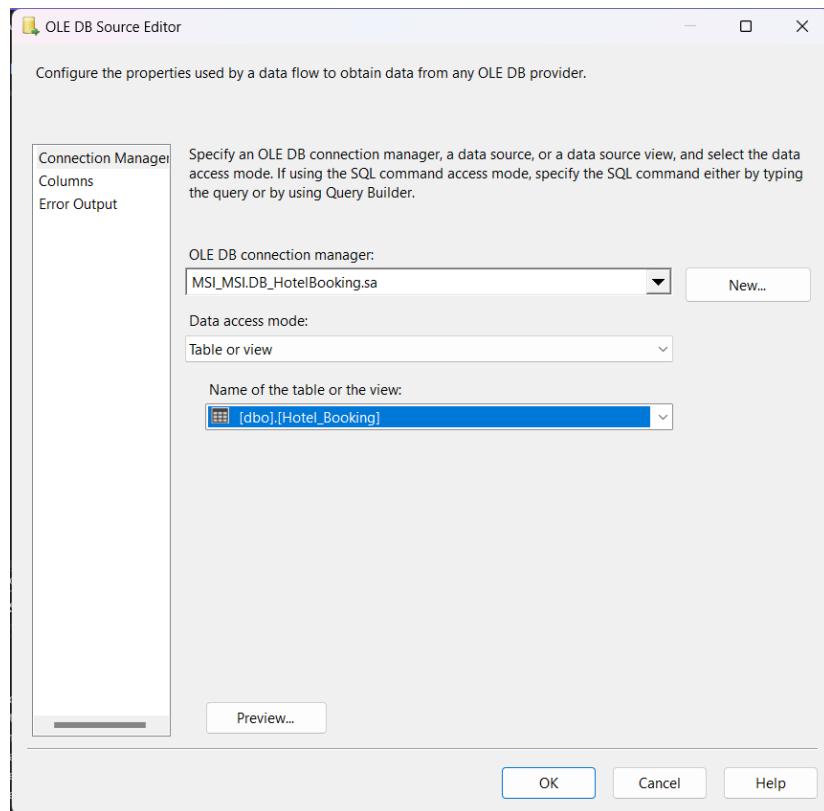


Figure 64. Chọn Table sẽ lấy dữ liệu

- **Bước 4:** Chọn Columns để lựa chọn các cột cần sử dụng và nhấn OK.

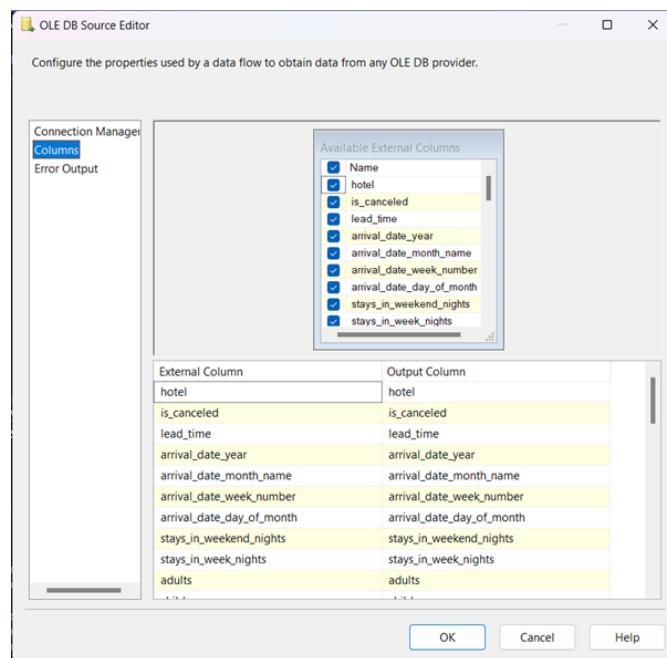


Figure 65. Chọn các column cần sử dụng

- **Bước 5:** Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau lên thuộc tính sử dụng là distribution_channel.

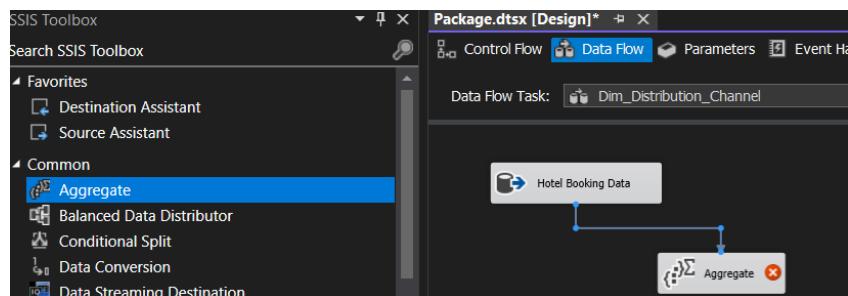


Figure 66. Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau

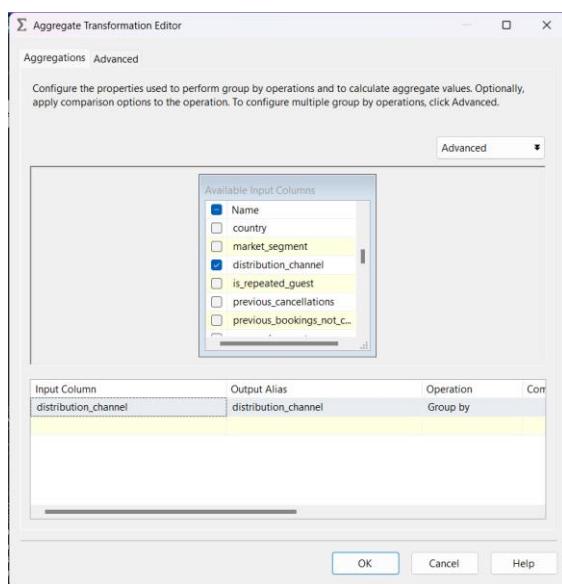
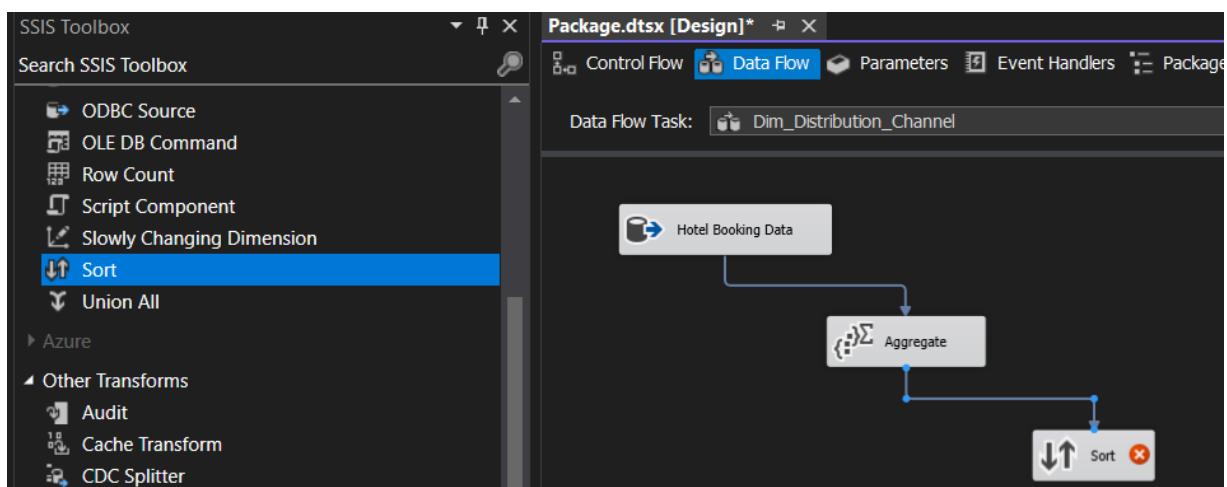


Figure 67. Chọn các thuộc tính cần gom nhóm

- **Bước 6:** Dùng công cụ Sort để sắp xếp các giá trị theo chiều tăng dần.



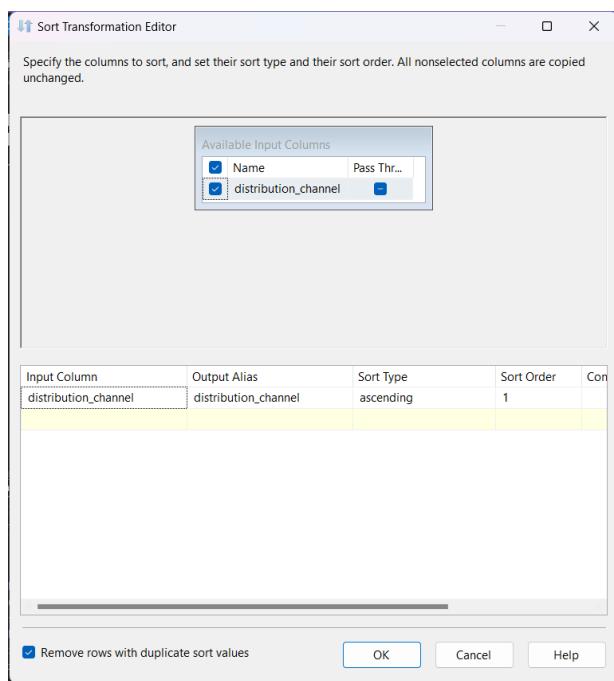
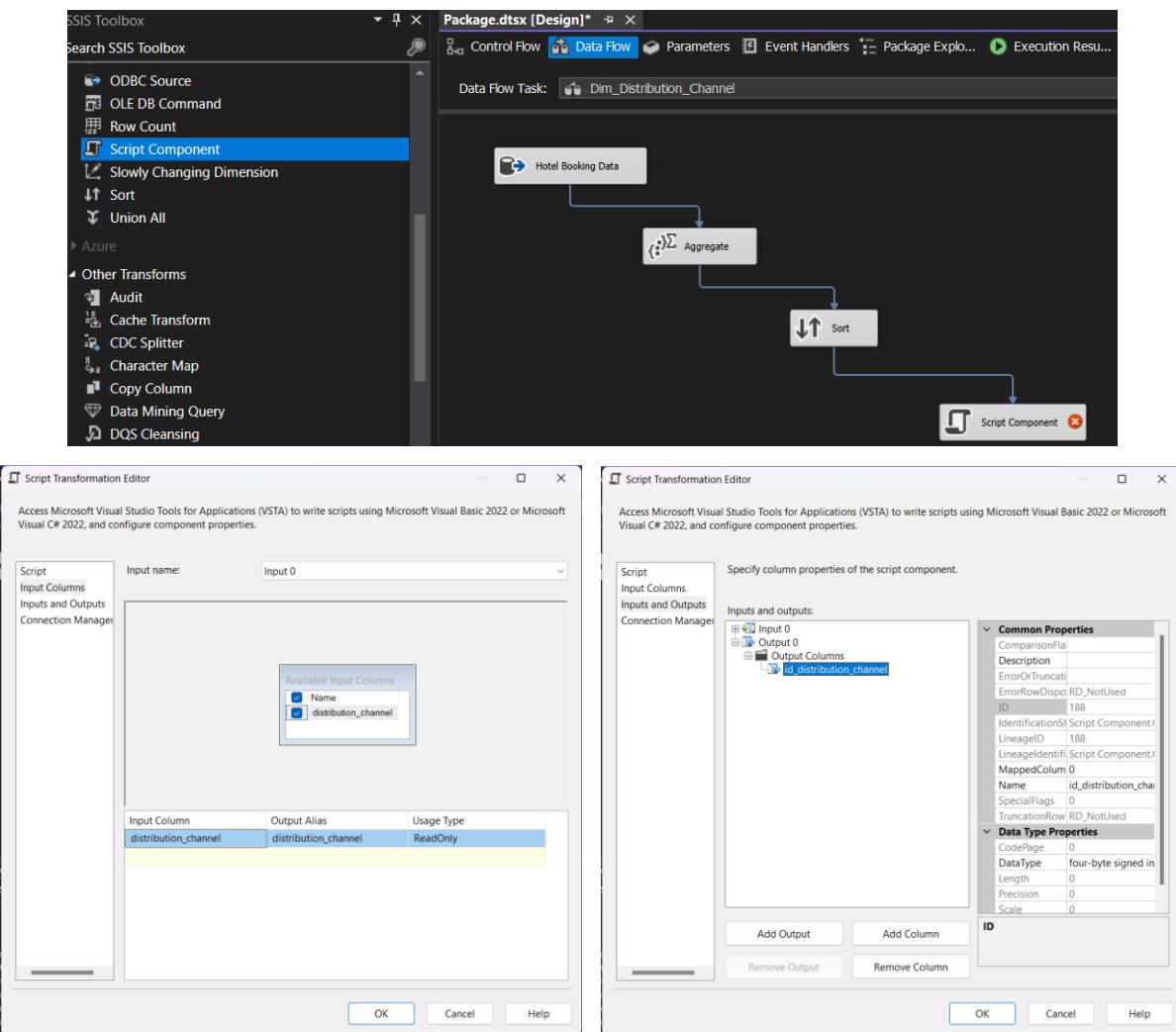


Figure 7-8. Sắp xếp các giá trị theo chiều tăng dần

- **Bước 7:** Kéo thả công cụ Script Component để tạo khóa chính id_distribution_channel.



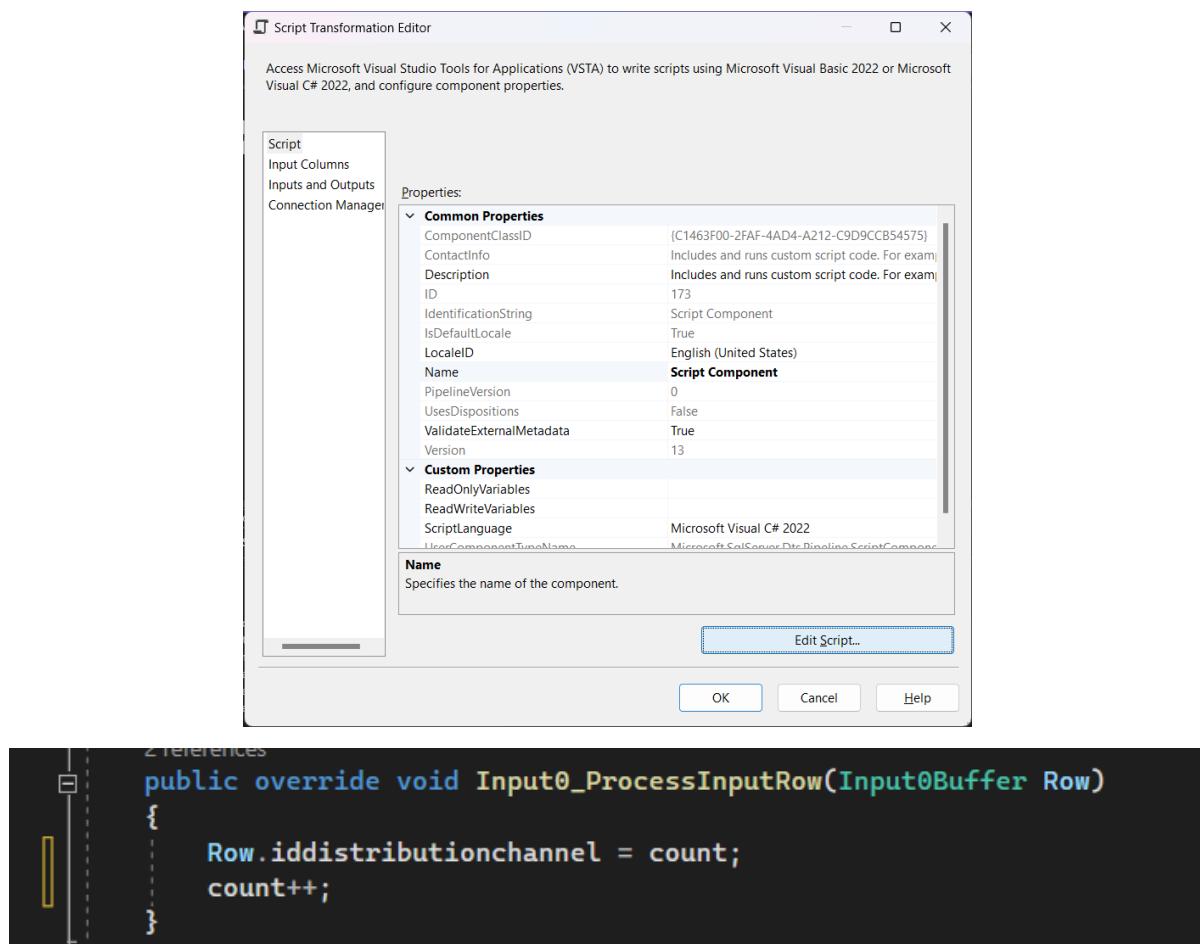


Figure 9-10-11-12-13. Sử dụng Script Component để tạo khóa chính id_distribution_channel

- **Bước 8:** Chọn công cụ OLE DB Destination để tiến hành tạo bảng trong cơ sở dữ liệu DB_HotelBooking với tên gọi là Dim_Distribution_Channel.

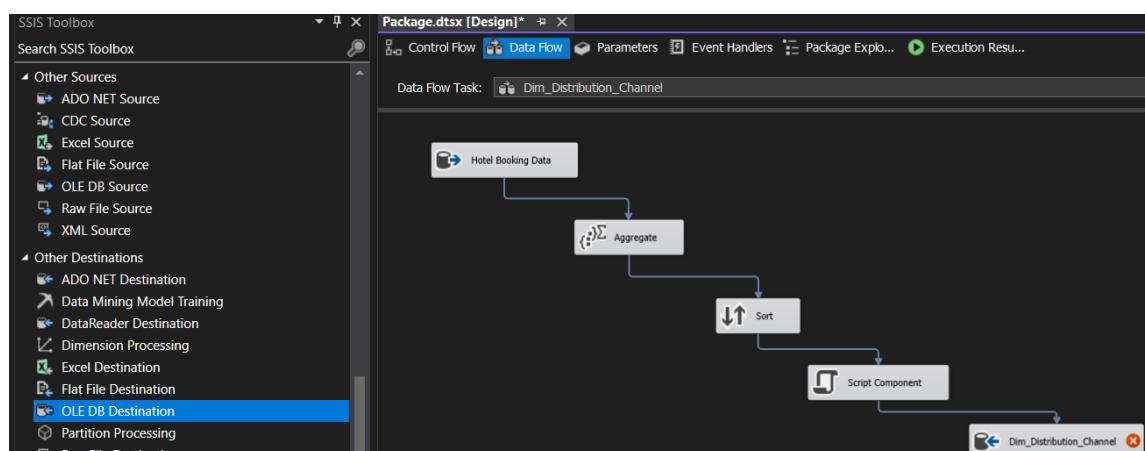


Figure 68. Sử dụng OLE DB Destination để tạo bảng

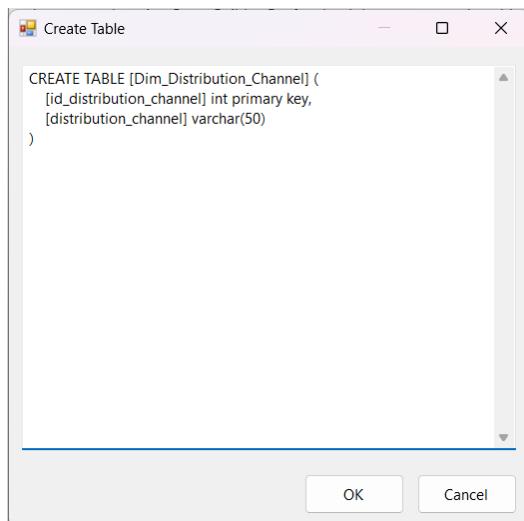


Figure 69. Tạo bảng Dim_Distribution_Channel

- **Bước 9:** Nhấn Mappings để kiểm tra kết nối → Ok để hoàn tất quá trình tạo bảng.

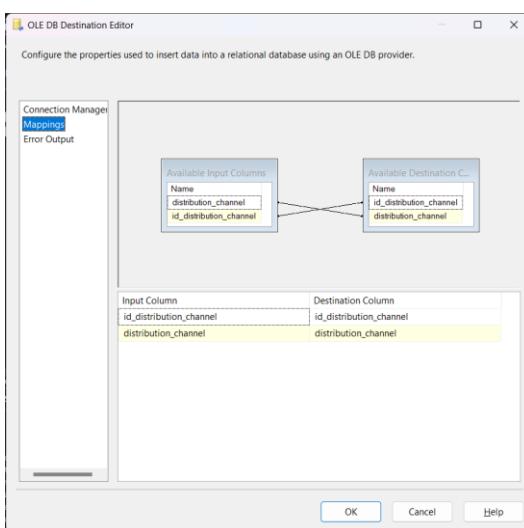


Figure 70. Quá trình Mappings dữ liệu

- **Bước 10:** Chọn Start để bắt đầu quá trình đổ dữ liệu. Khi đổ thành công, ta có kết quả như sau:

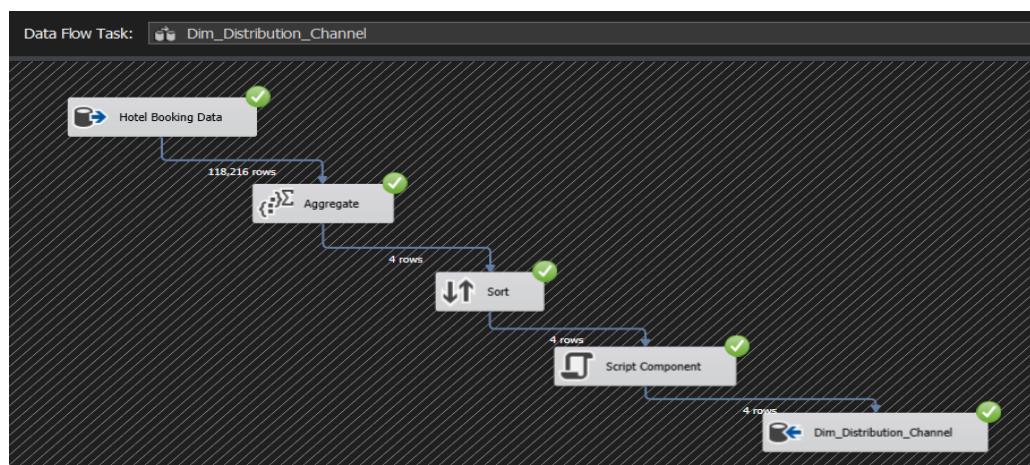


Figure 71. Hoàn thành đổ dữ liệu vào Dim_Distribution_Channel trong kho dữ liệu

- **Bước 11:** Kiểm tra bảng Dim_Distribution_Channel trên SQL Server.

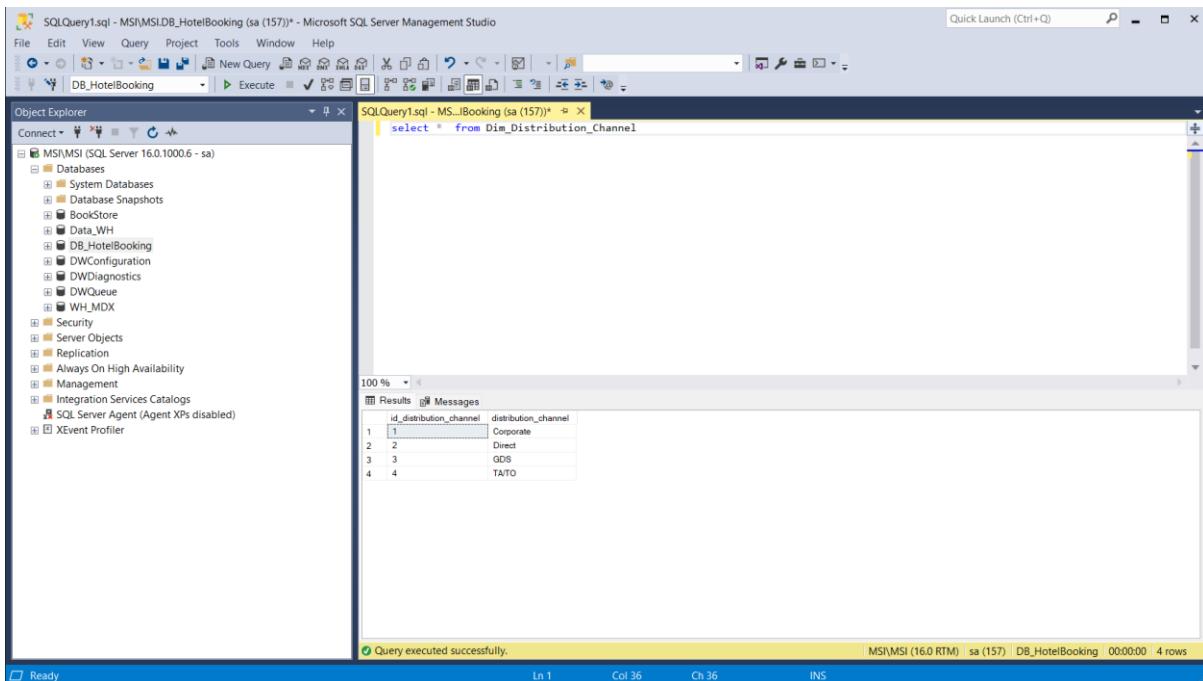


Figure 72. Kiểm tra bảng Dim_Distribution_Channel trong SQL Server

2.5.5. Bảng Dim_Market_Segment

- **Bước 1:** Kéo chức năng Data Flow Task từ cột trái sang màn hình làm việc và đổi tên thành Dim_Market_Segment.

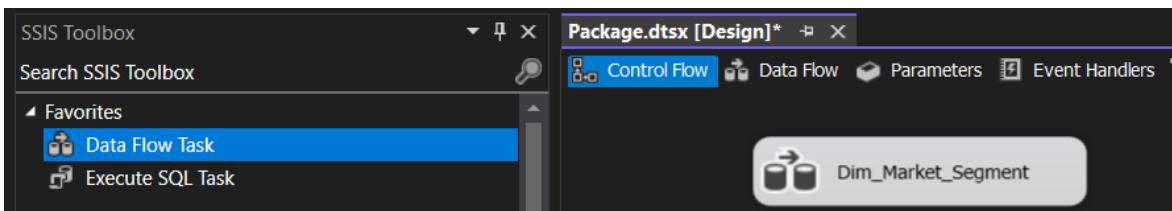


Figure 73. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Market_Segment

- **Bước 2:** Nhấn double vào Data Flow Task vừa tạo và tìm kiếm chức năng OLE DB Source, kéo vào màn hình làm việc

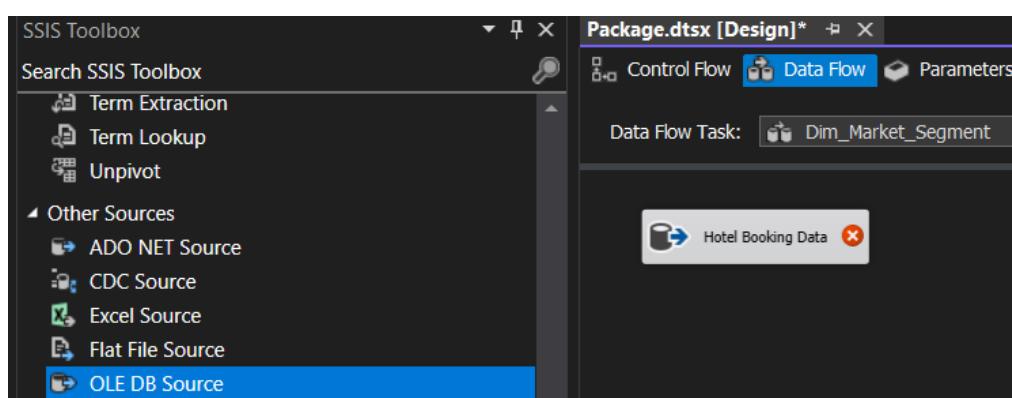


Figure 74. Kéo chức năng OLE DB Source vào màn hình làm việc

- **Bước 3:** Click double vào OLE DB Source và dẫn đến DB_HotelBooking. Sau đó chọn Name of the table or the view là [dbo].Hotel_Booking.

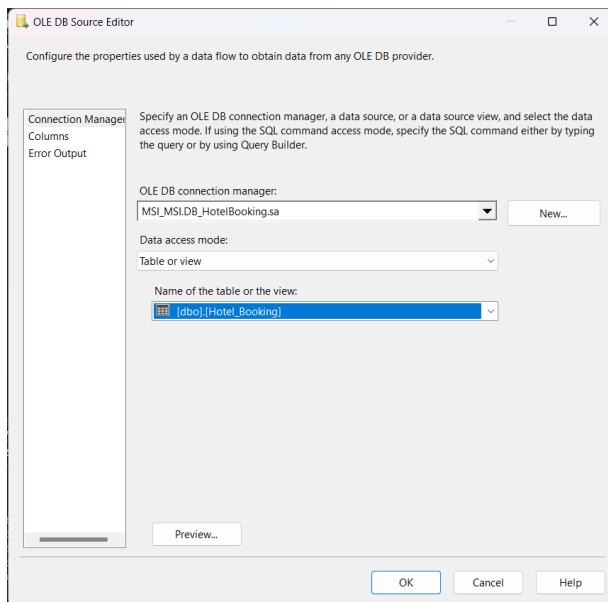


Figure 75. Chọn Table sẽ lấy dữ liệu

- **Bước 4:** Chọn Columns để lựa chọn các cột cần sử dụng và nhấn OK

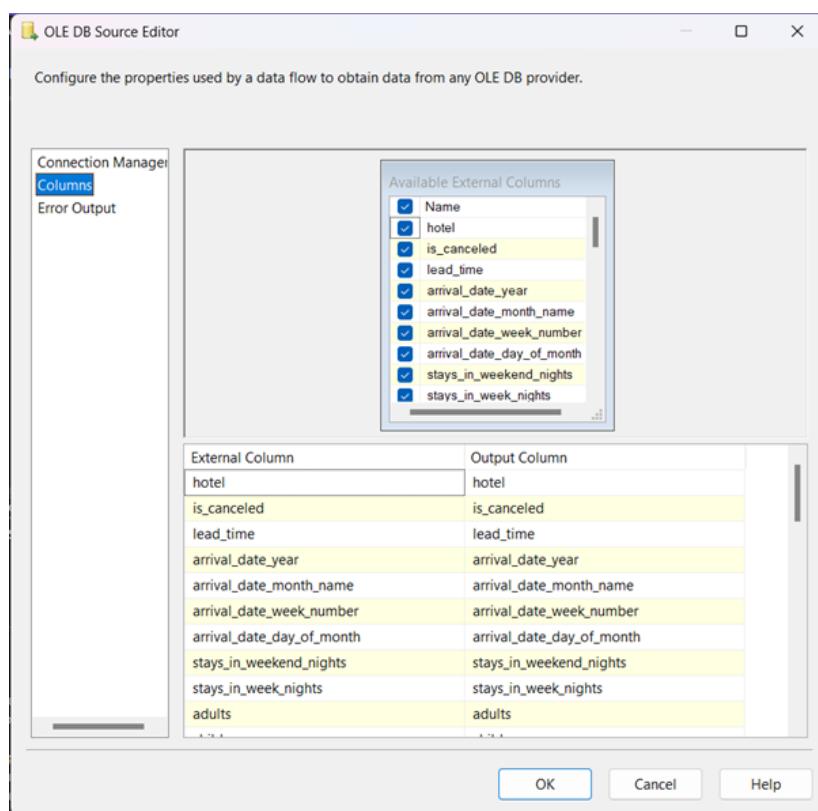


Figure 76. Chọn các column cần sử dụng

- **Bước 5:** Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau lên thuộc tính sử dụng là market_segment.

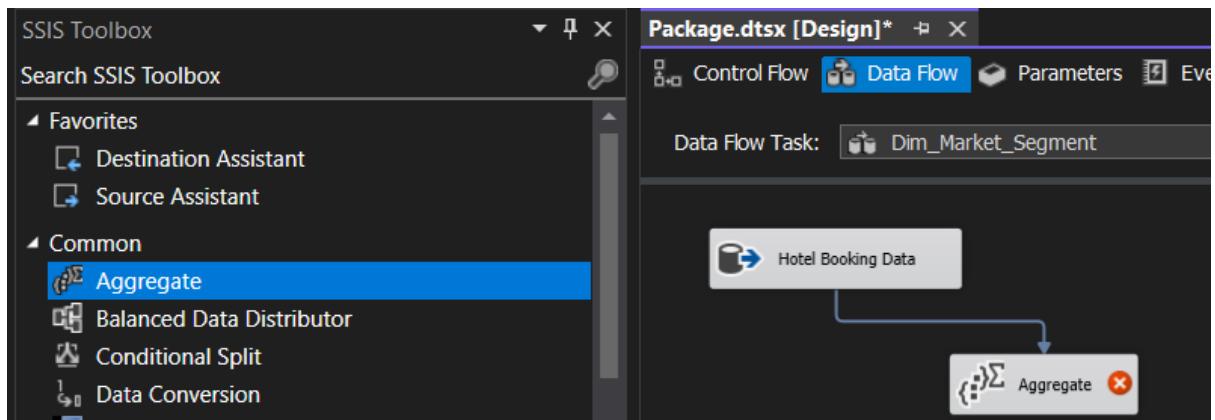


Figure 77. Dùng công cụ Aggregate để thực hiện lọc các thuộc tính trùng nhau

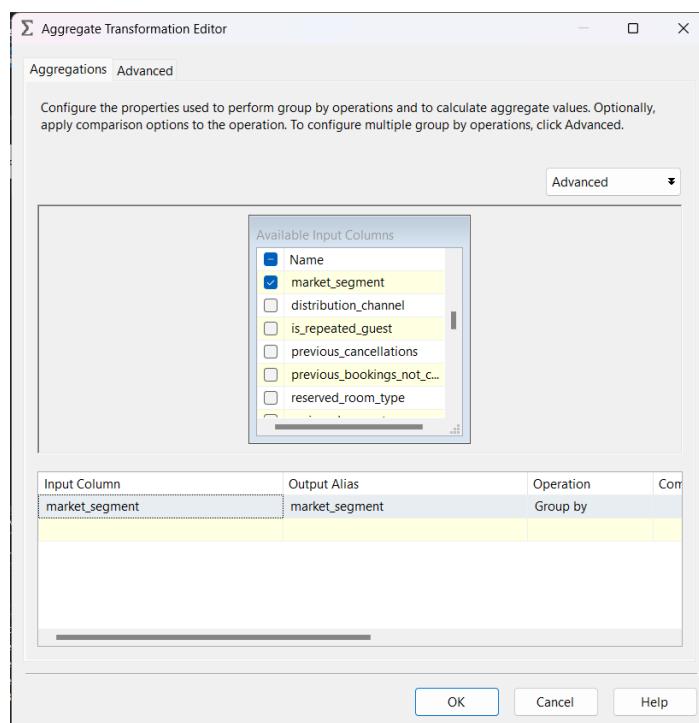
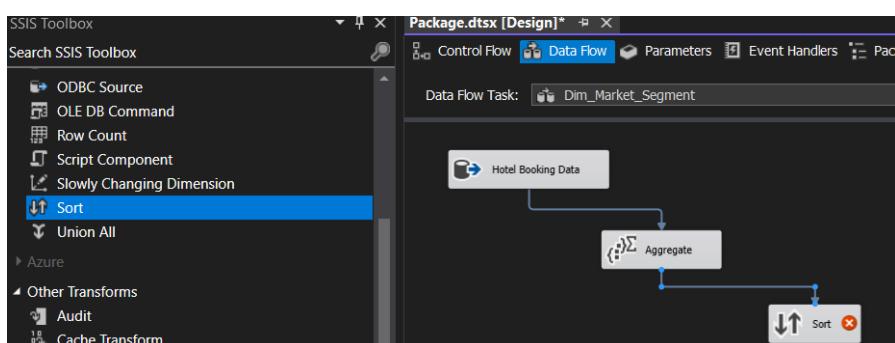


Figure 78. Chọn các thuộc tính cần gom nhóm

- **Bước 6:** Dùng công cụ Sort để sắp xếp các giá trị theo chiều tăng dần.



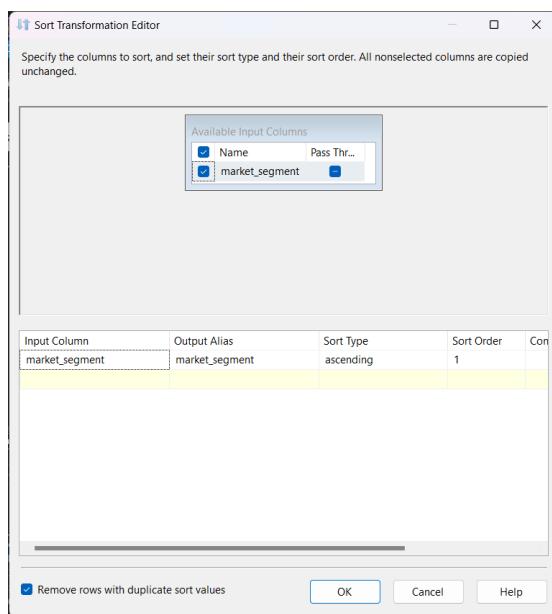
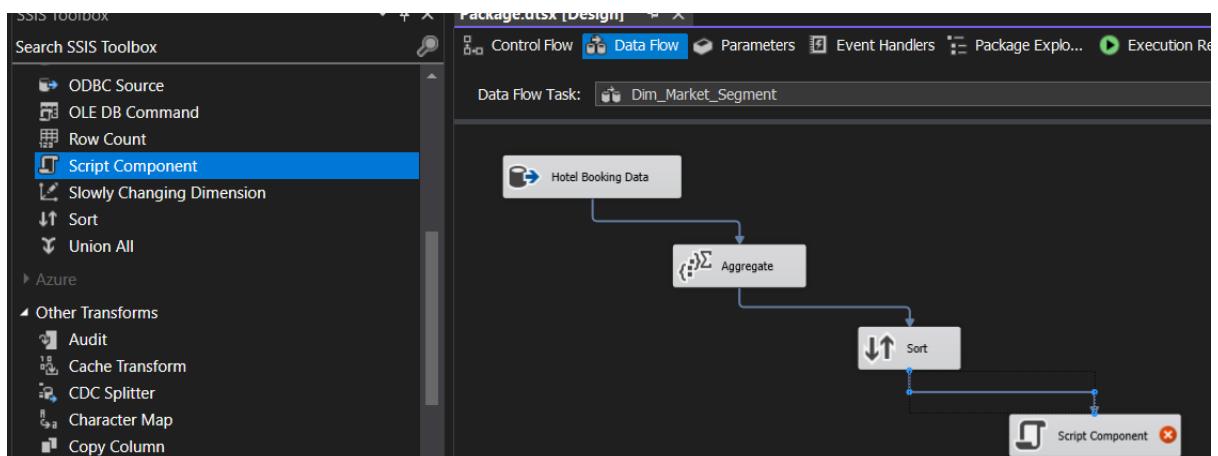


Figure 7-8. Sắp xếp các giá trị theo chiều tăng dần

- **Bước 7:** Kéo thả công cụ Script Component để tạo khóa chính id_market_segment.



Common Properties	
ComparisonFlag	
Description	
ErrorOnTruncation	
ErrorRowDisposition	
ID	188
IdentificationString	Script Component
LineageID	188
LineageIdentifier	Script Component
MappedColumn	0
Name	id_market_segment
SpecialFlags	0
TruncationRowDisposition	RD_NotUsed

Data Type Properties	
CodePage	0
DataType	four-byte signed
Length	0
Precision	0
Scale	0

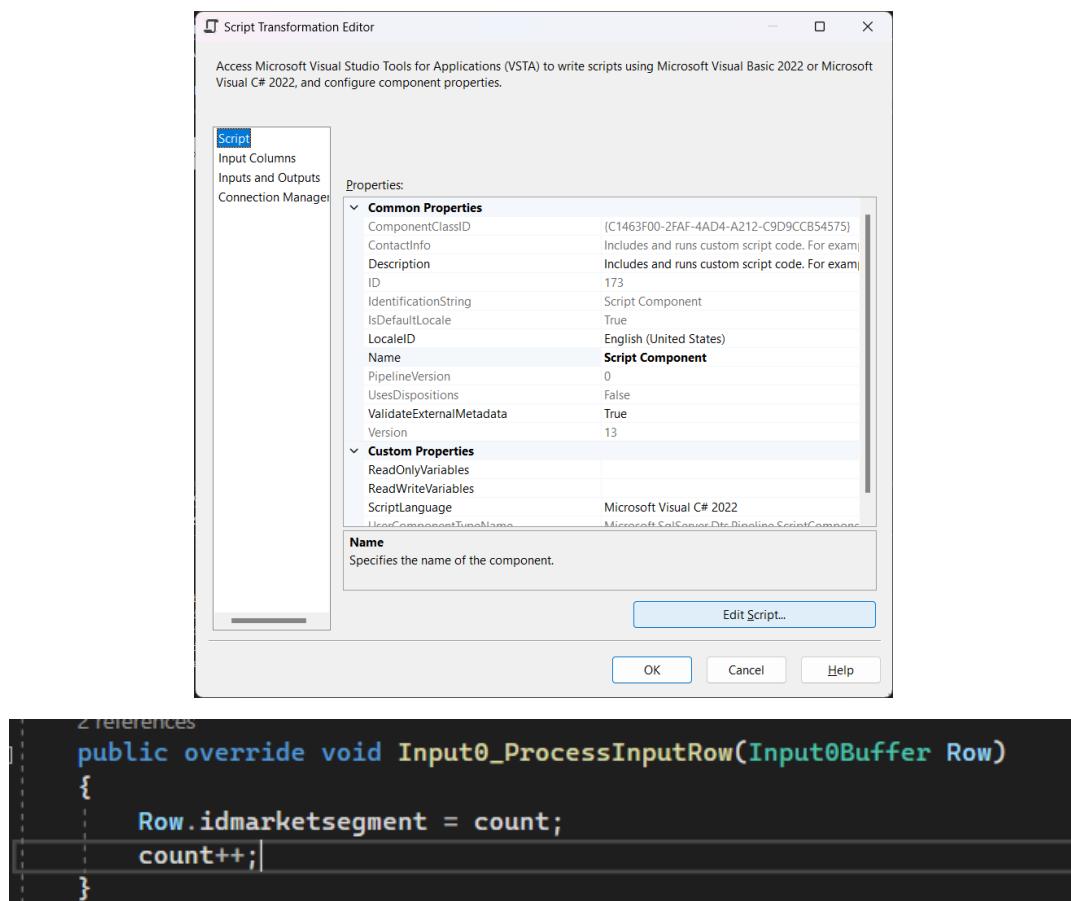


Figure 9-10-11-12-13. Sử dụng Script Component để tạo khóa chính id_market_segment

- **Bước 8:** Chọn công cụ OLE DB Destination để tiến hành tạo bảng trong cơ sở dữ liệu DB_HotelBooking với tên gọi là Dim_Market_Segment.

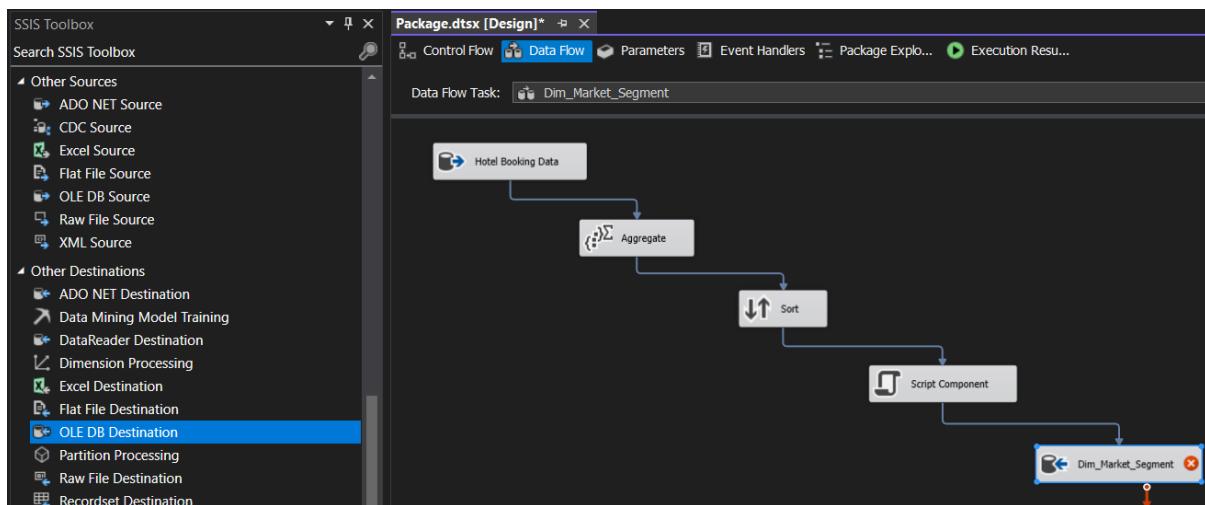


Figure 79. Sử dụng OLE DB Destination để tạo bảng

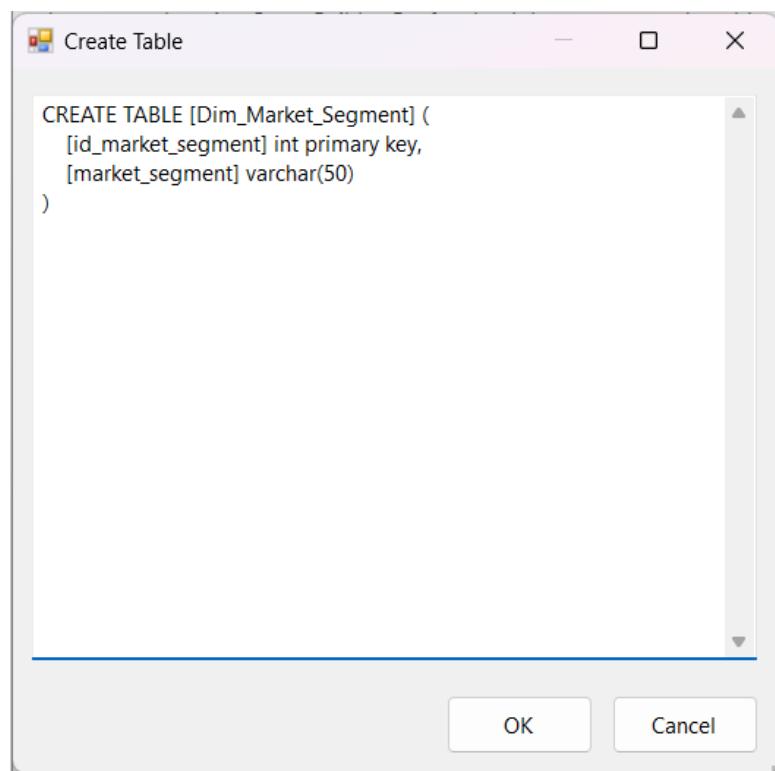


Figure 80. Tạo bảng Dim_Market_Segment

- **Bước 9:** Nhấn Mappings để kiểm tra kết nối và nhấn Ok để hoàn tất quá trình tạo bảng.

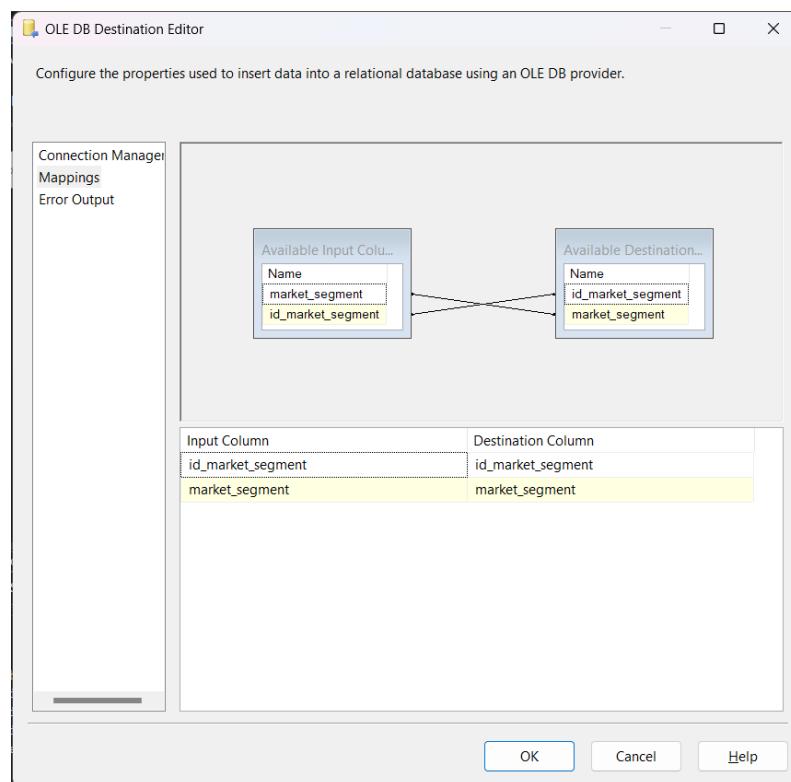


Figure 81. Quá trình Mappings dữ liệu

- **Bước 10:** Chọn Start để bắt đầu quá trình đổ dữ liệu. Khi đổ thành công, ta có kết quả như sau

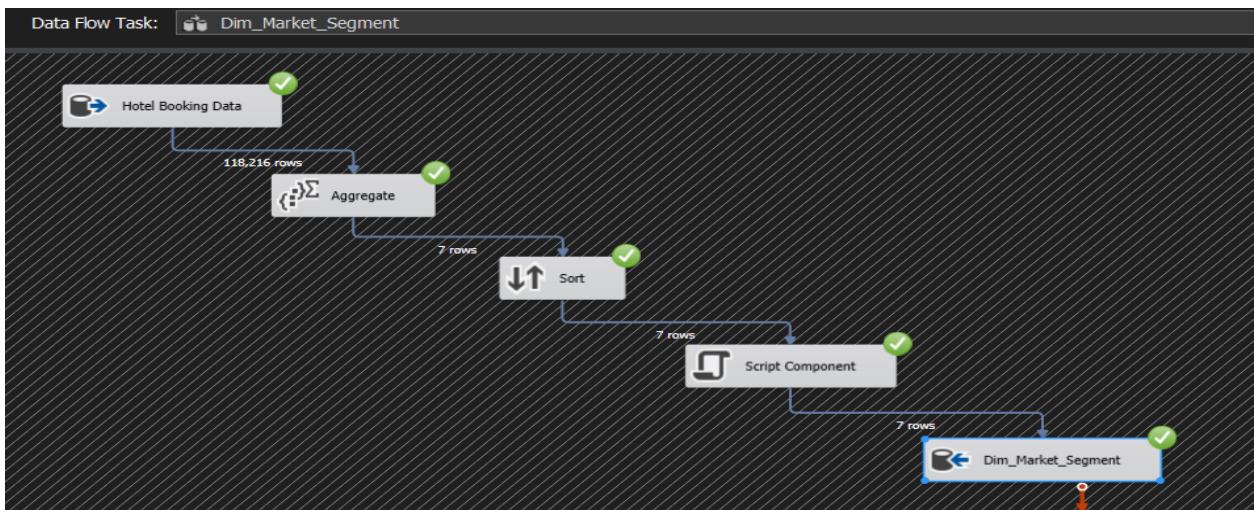


Figure 82. Hoàn thành đổ dữ liệu vào Dim_Market_Segment trong kho dữ liệu

- **Bước 11:** Kiểm tra bảng Dim_Market_Segment trên SQL Server.

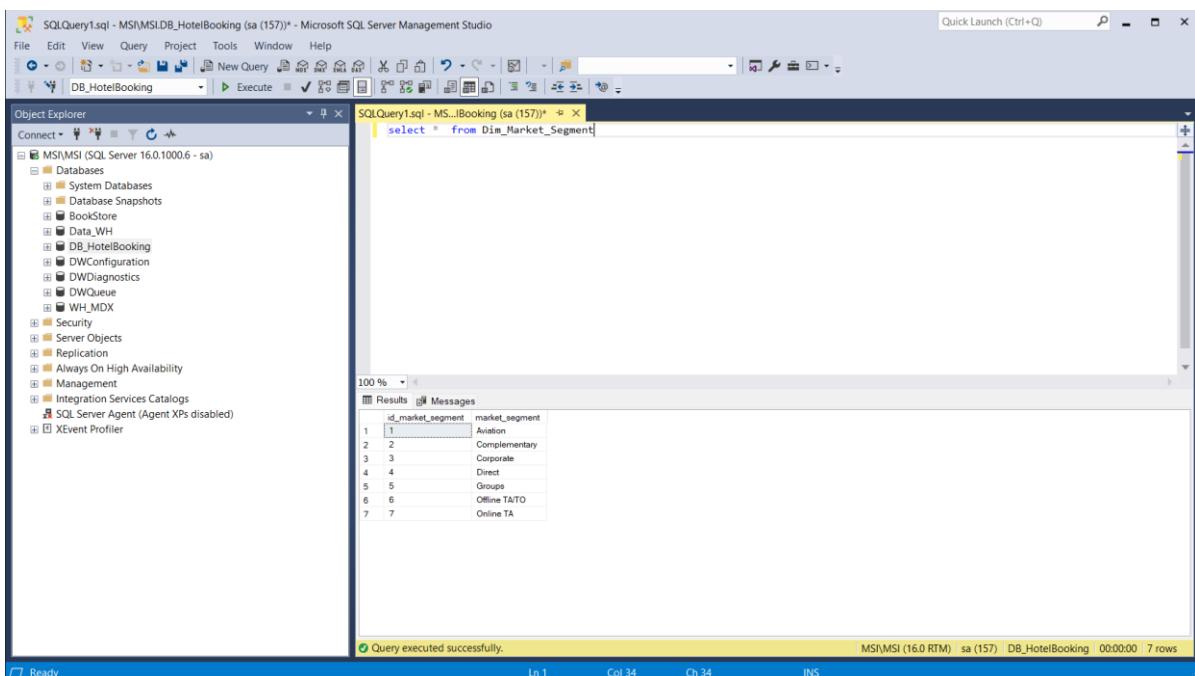


Figure 83. Kiểm tra bảng Dim_Market_Segment trong SQL Server

2.5.6. Bảng Dim_Customer_Type

- **Bước 1:** Kéo chức năng Data Flow Task từ cột trái sang màn hình làm việc và đổi tên thành Dim_Customer_Type.

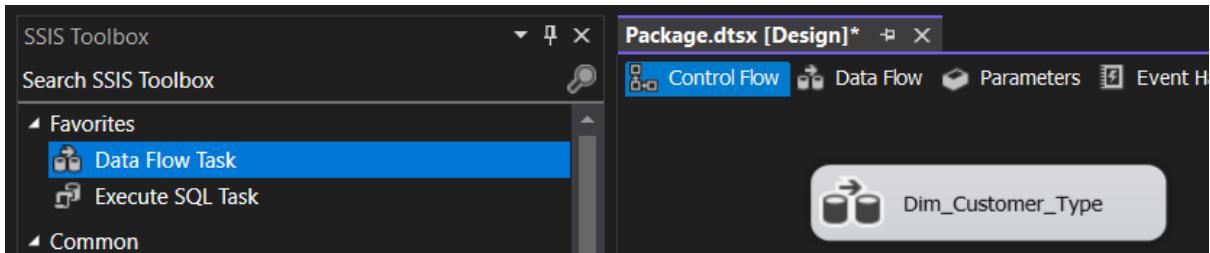


Figure 84. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Customer_Type

- **Bước 2:** Nhấn double vào Data Flow Task vừa tạo và tìm kiếm chức năng OLE DB Source, kéo vào màn hình làm việc.

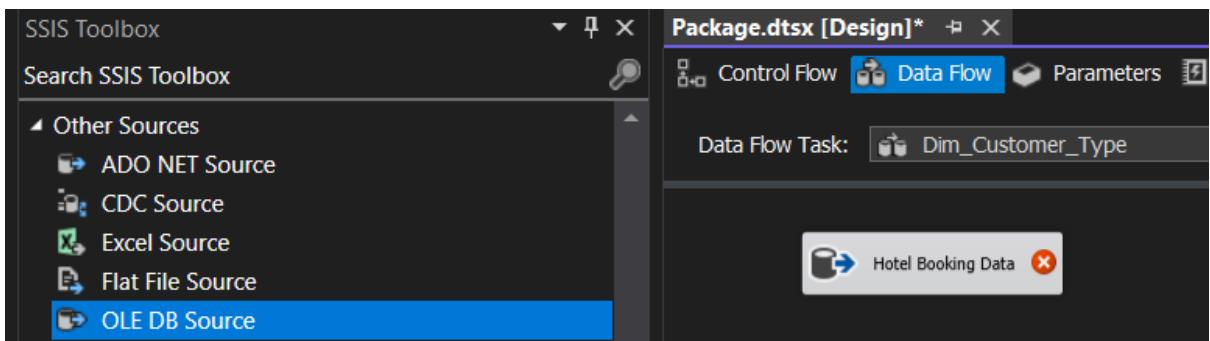


Figure 85. Kéo chức năng OLE DB Source vào màn hình làm việc

- **Bước 3:** Click double vào OLE DB Source và dẫn đến DB_HotelBooking. Sau đó chọn Name of the table or the view là [dbo].Hotel_Booking.

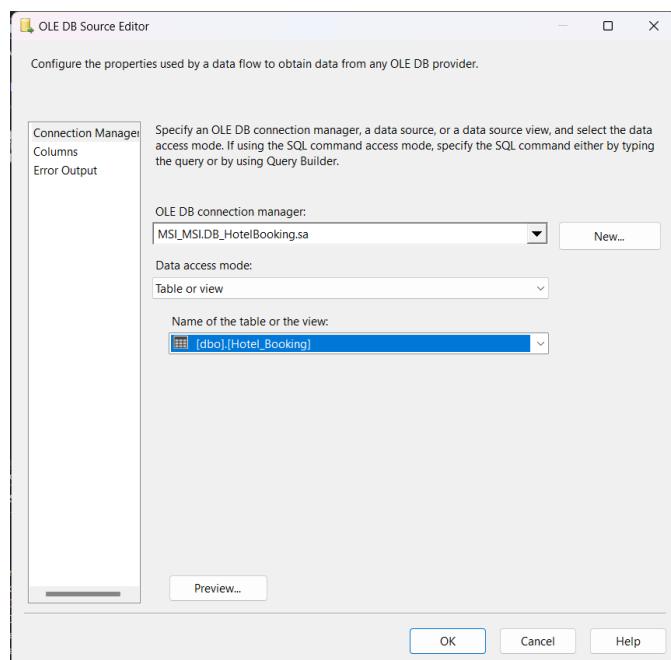


Figure 86. Chọn Table sẽ lấy dữ liệu

- **Bước 4:** Chọn Columns để lựa chọn các cột cần sử dụng và nhấn OK.

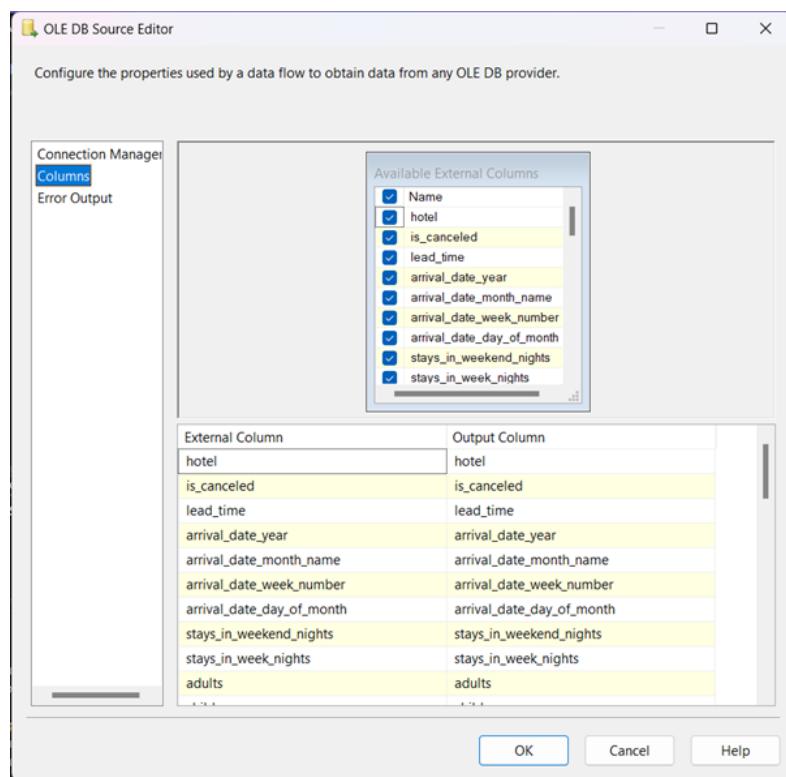


Figure 87. Chọn các column cần sử dụng

- **Bước 5:** Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau lên thuộc tính sử dụng là customer_type.

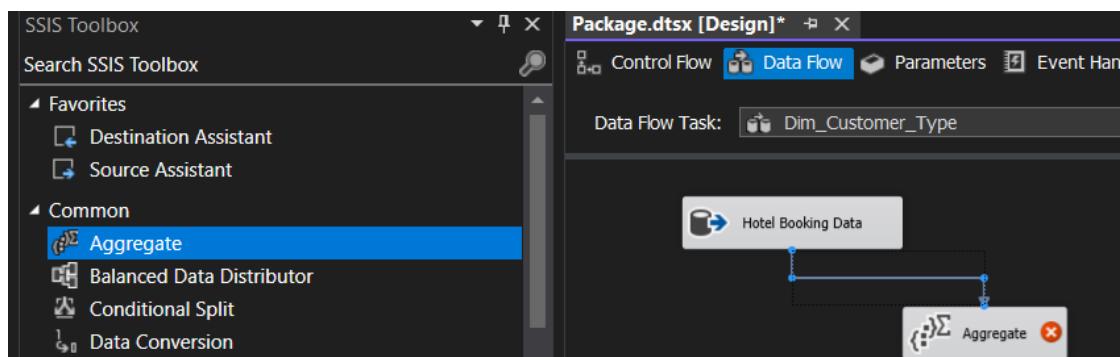


Figure 88. Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau

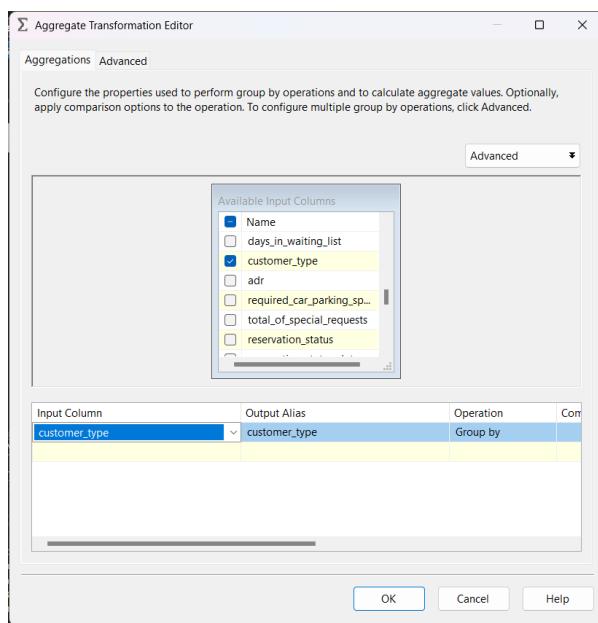


Figure 89. Chọn các thuộc tính cần gom nhóm

- **Bước 6:** Dùng công cụ Sort để sắp xếp các giá trị theo chiều tăng dần.

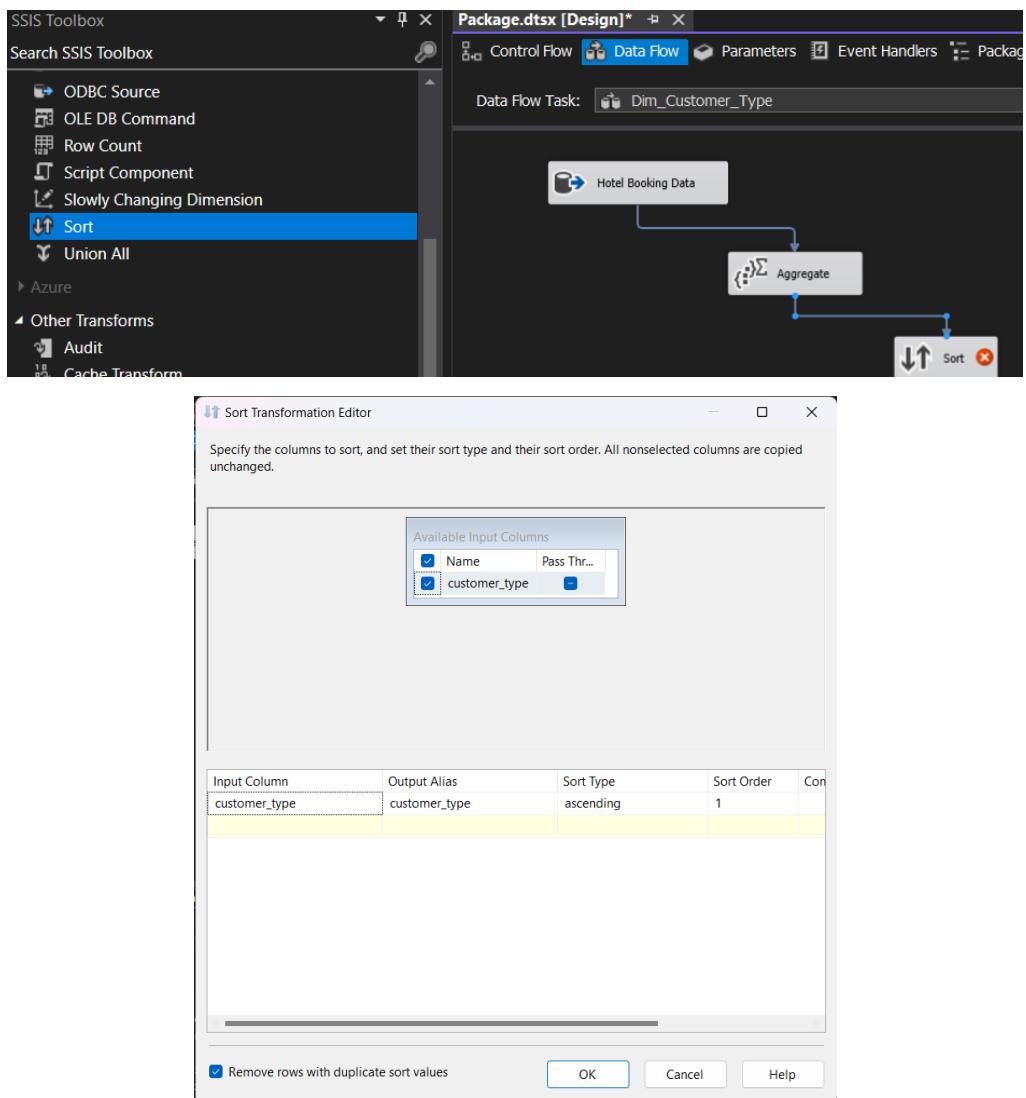


Figure 7-8. Sắp xếp các giá trị theo chiều tăng dần

- **Bước 7:** Kéo thả công cụ Script Component để tạo khóa chính id_customer_type.

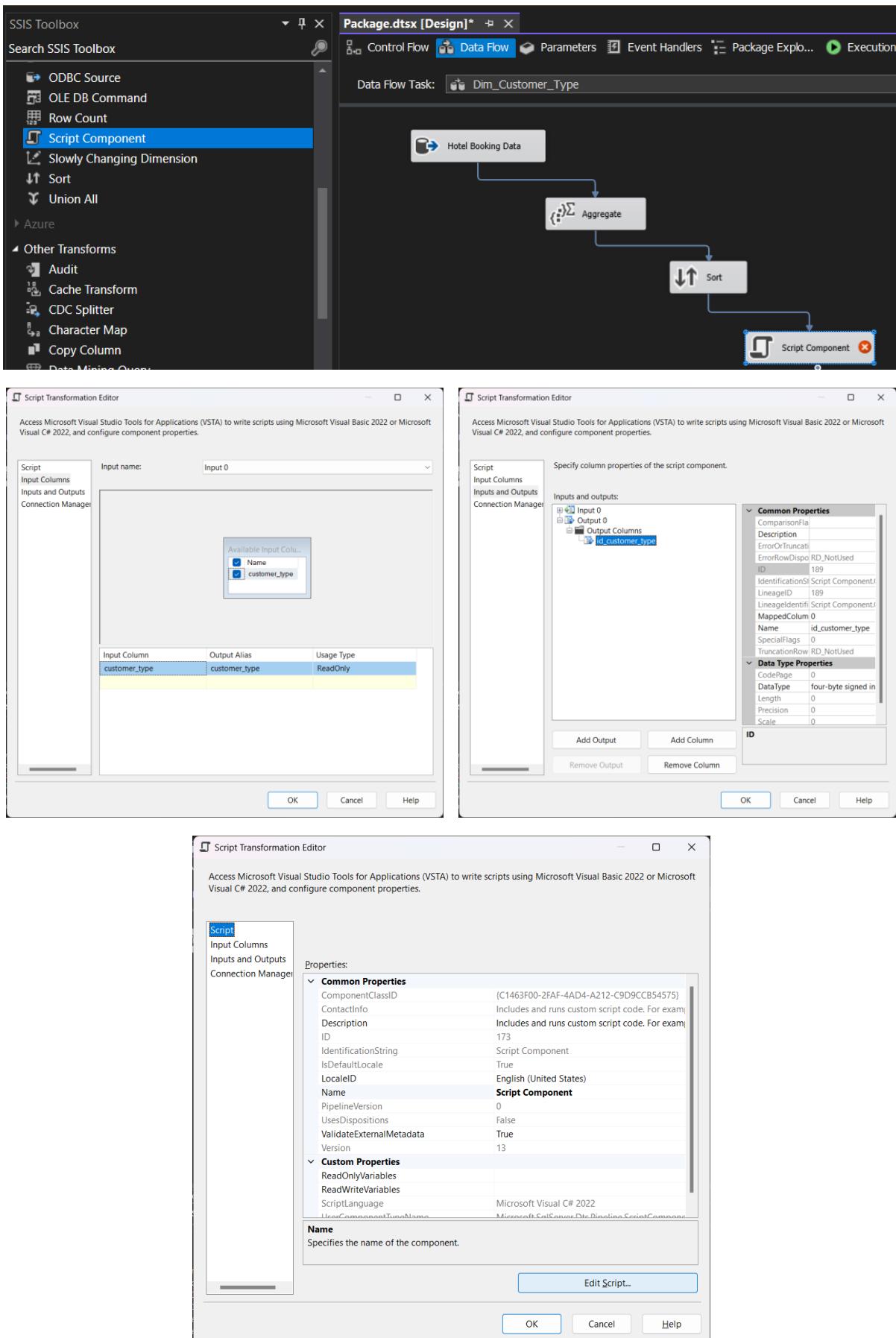


Figure 9-10-11-12-13. Sử dụng Script Component để tạo khóa chính id_customer_type

- **Bước 8:** Chọn công cụ OLE DB Destination để tiến hành tạo bảng trong cơ sở dữ liệu DB_HotelBooking với tên gọi là Dim_Customer_Type.

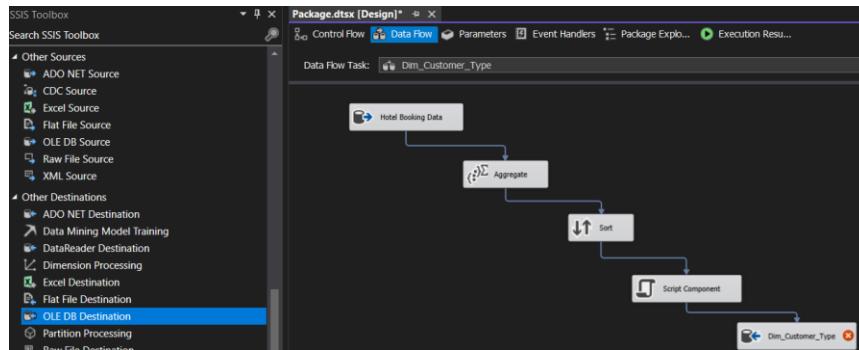


Figure 90. Sử dụng OLE DB Destination để tạo bảng

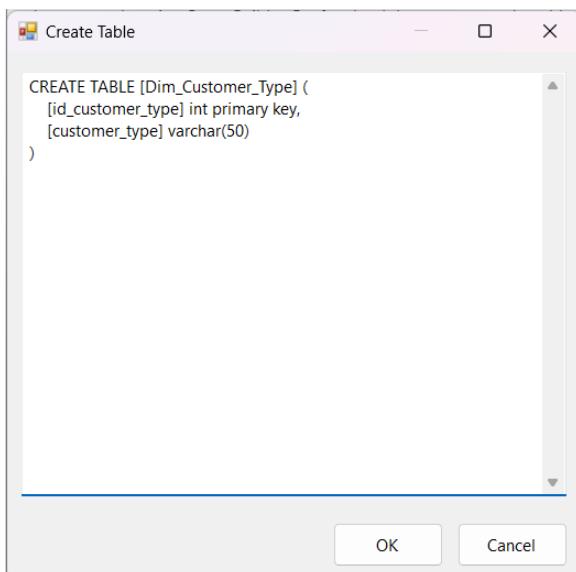


Figure 91. Tạo bảng Dim_Customer_Type

- **Bước 9:** Nhấn Mappings để kiểm tra kết nối và nhấn Ok để hoàn tất quá trình tạo bảng.

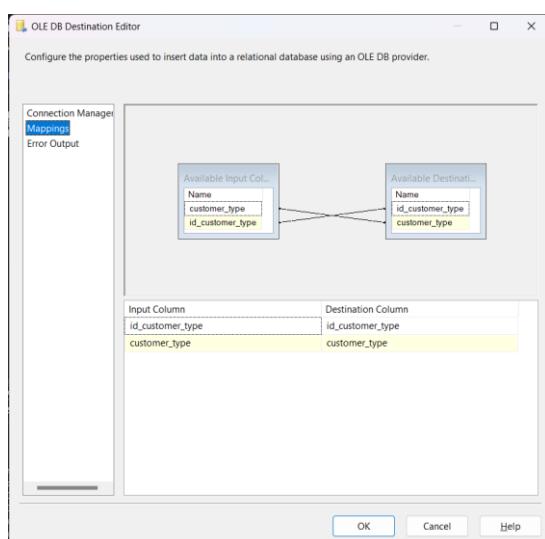


Figure 92. Quá trình Mappings dữ liệu

- **Bước 10:** Chọn Start để bắt đầu quá trình đổ dữ liệu. Khi đổ thành công, ta có kết quả như sau:

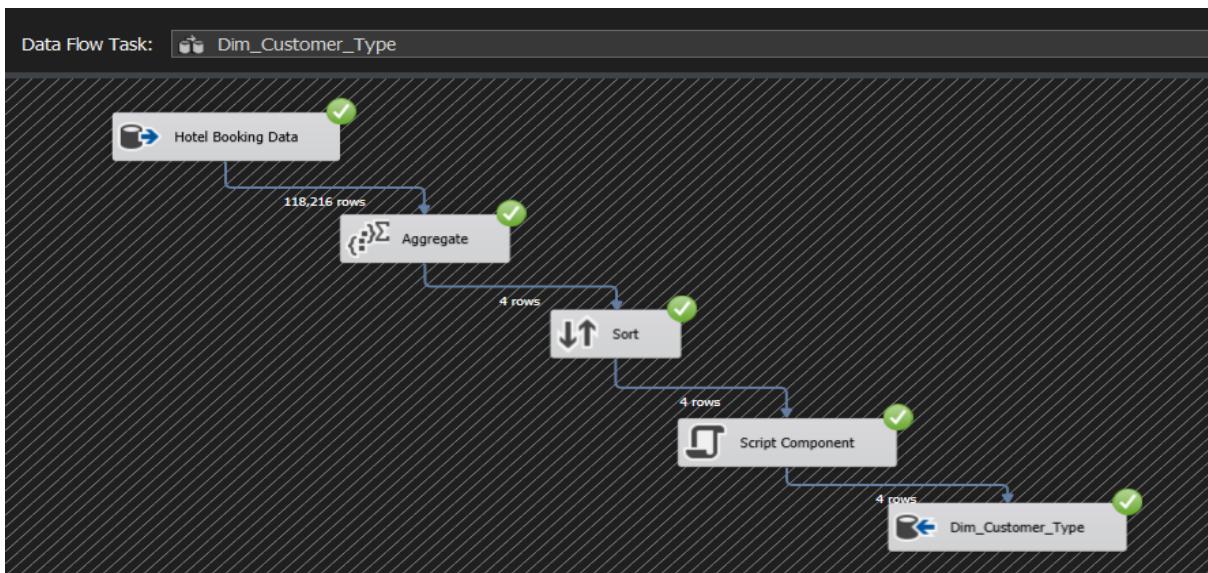


Figure 93. Hoàn thành đổ dữ liệu vào Dim_Customer_Type trong kho dữ liệu

- **Bước 11:** Kiểm tra bảng Dim_Customer_Type trên SQL Server.

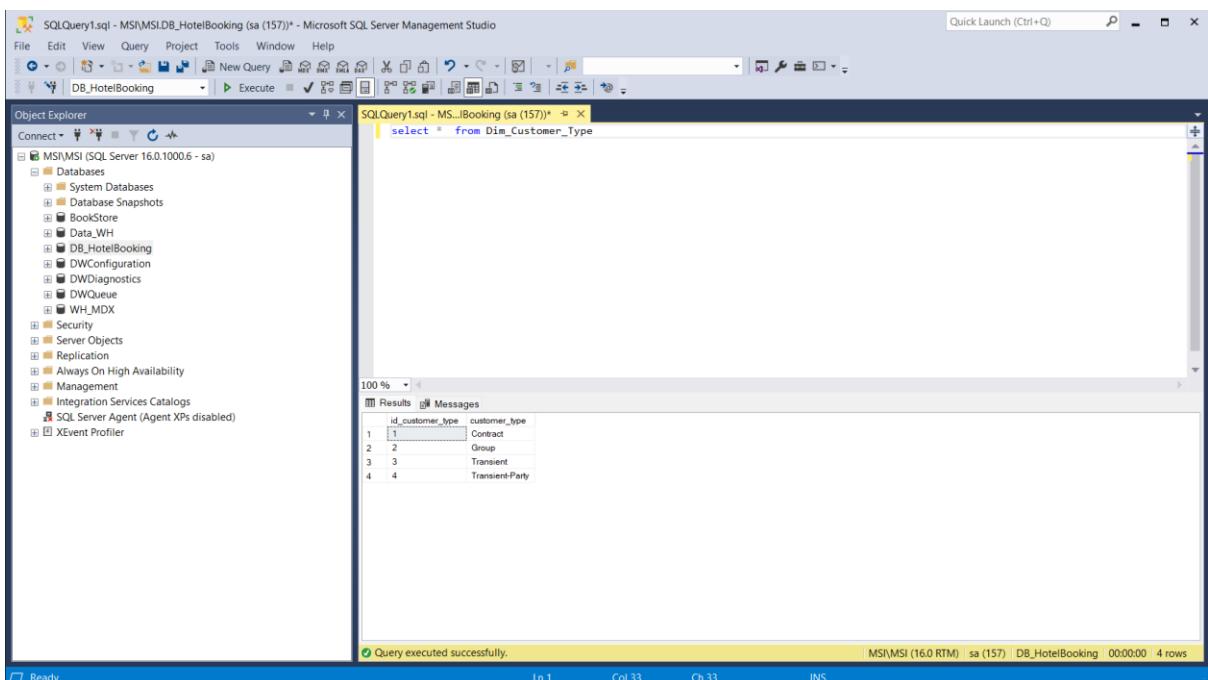


Figure 94. Kiểm tra bảng Dim_Customer_Type trong SQL Server

2.5.7. Bảng Dim_Country

- **Bước 1:** Kéo chức năng Data Flow Task từ cột trái sang màn hình làm việc và đổi tên thành Dim_Country.

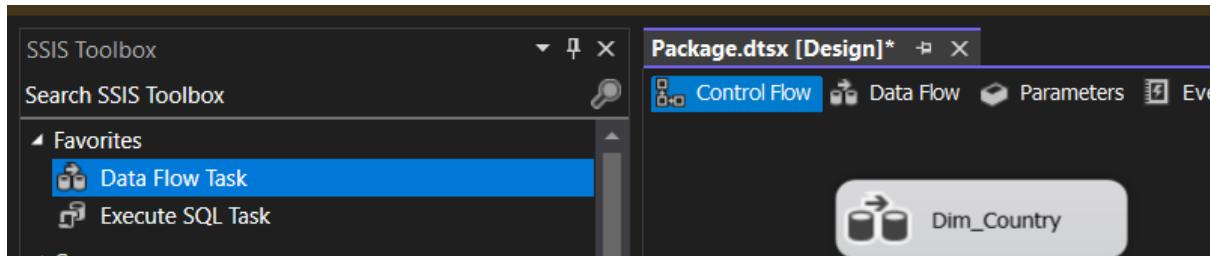


Figure 95. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Country

- **Bước 2:** Nhấn double vào Data Flow Task vừa tạo và tìm kiếm chức năng OLE DB Source, kéo vào màn hình làm việc

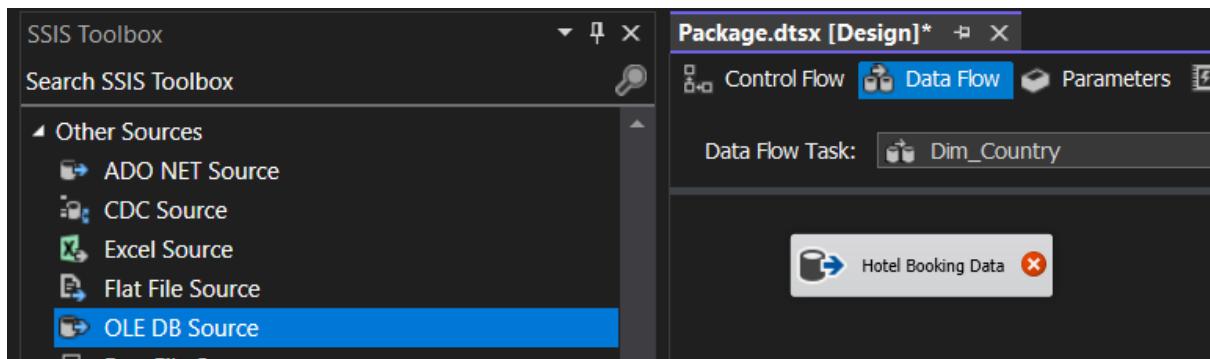


Figure 96. Kéo chức năng OLE DB Source vào màn hình làm việc

- **Bước 3:** Click double vào OLE DB Source và dẫn đến DB_HotelBooking. Sau đó chọn Name of the table or the view là [dbo].Hotel_Booking.

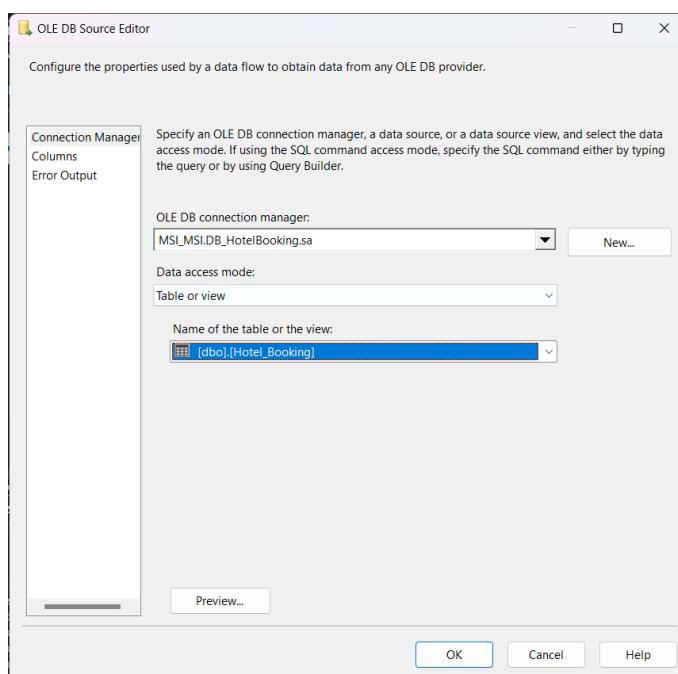


Figure 97. Chọn Table sẽ lấy dữ liệu

- **Bước 4:** Chọn Columns để lựa chọn các cột cần sử dụng và nhấn OK

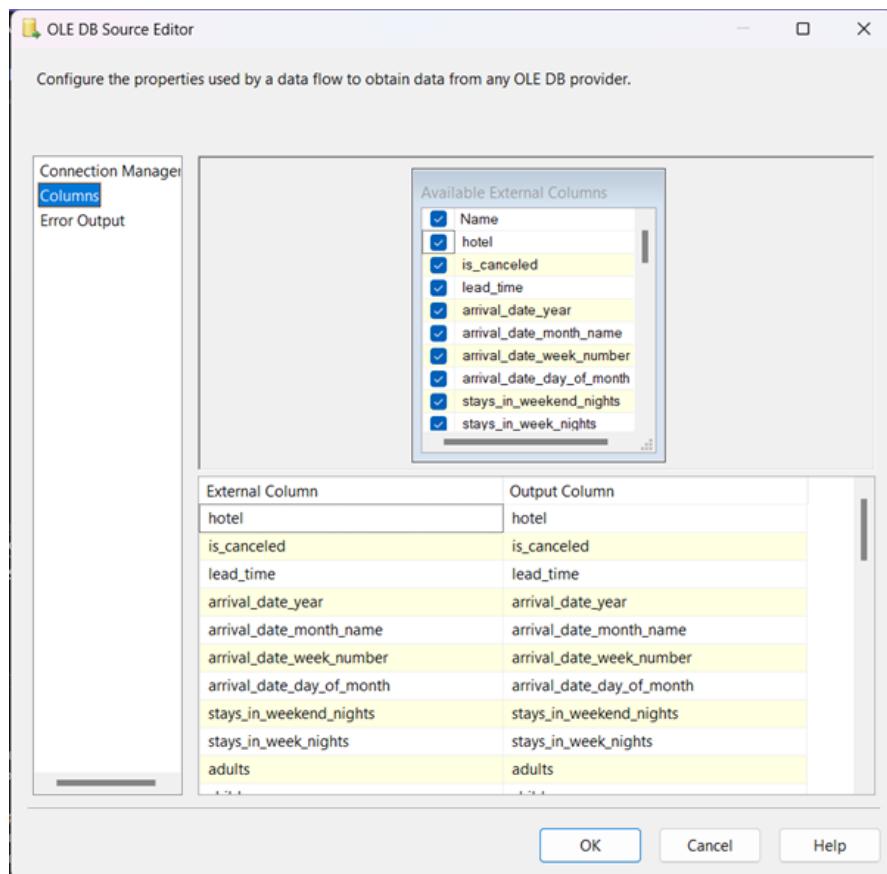


Figure 98. Chọn các column cần sử dụng

- **Bước 5:** Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau lên thuộc tính sử dụng là country.

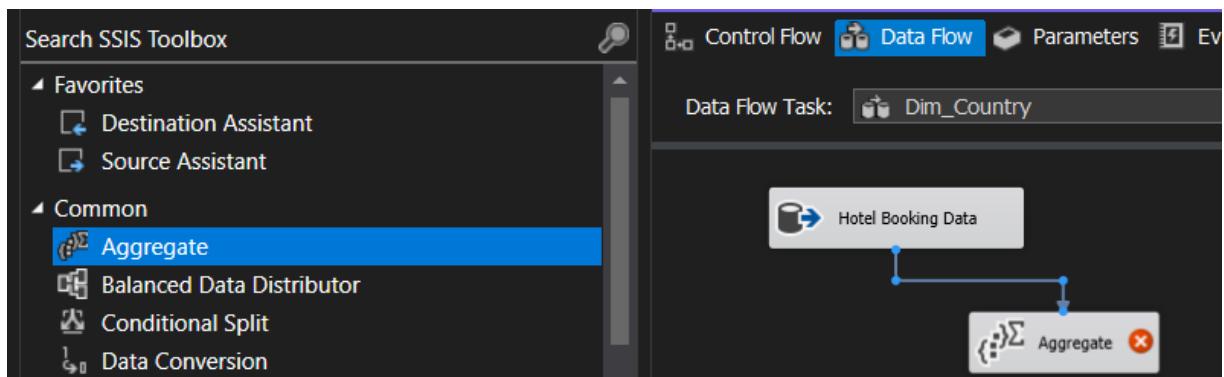


Figure 99. Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau

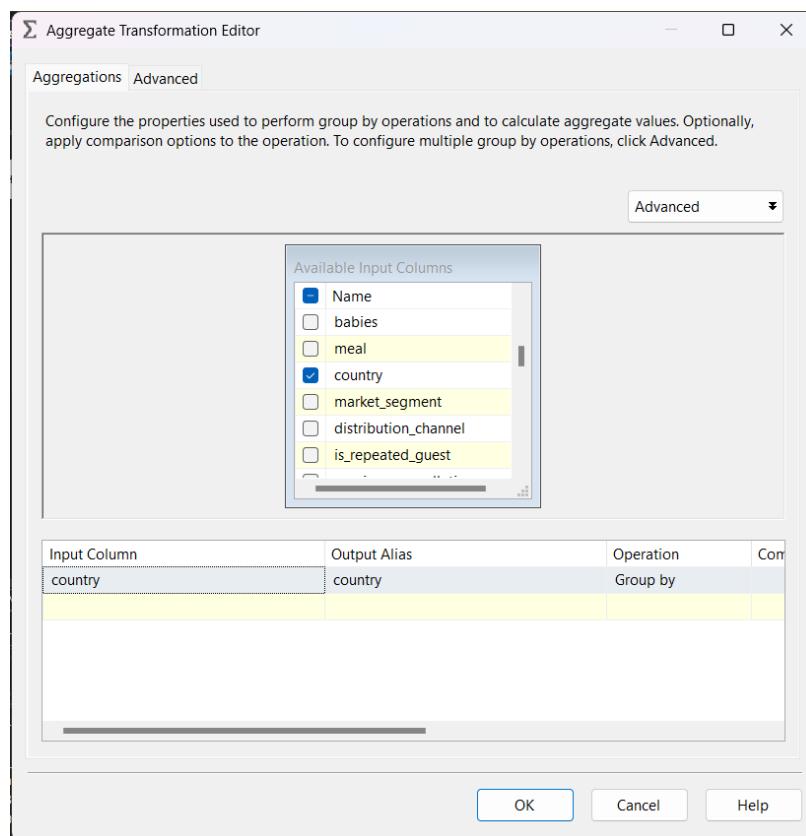
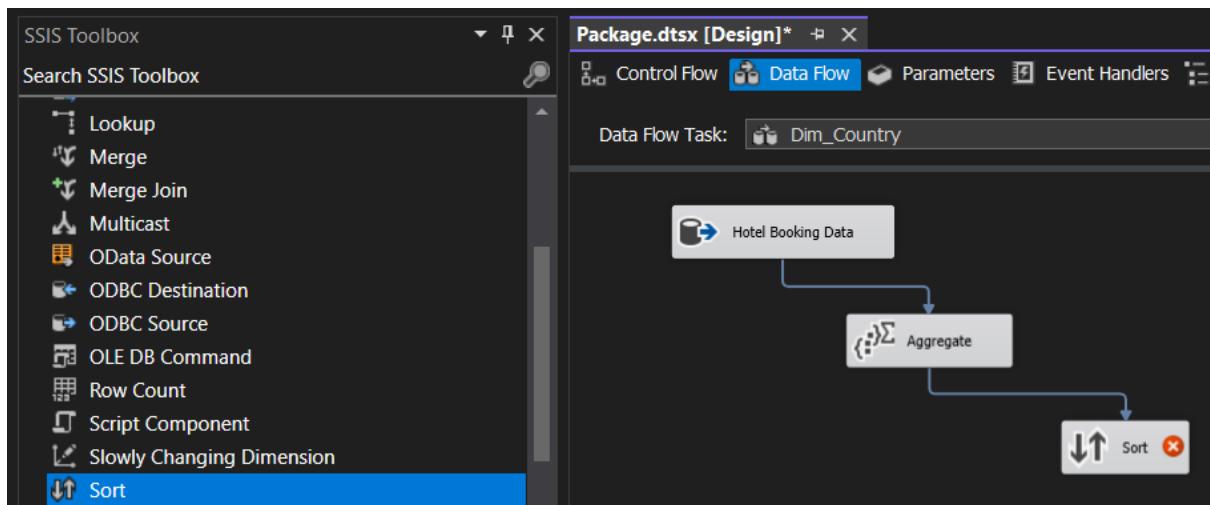


Figure 100. Chọn các thuộc tính cần gom nhóm

- **Bước 6:** Dùng công cụ Sort để sắp xếp các giá trị theo chiều tăng dần.



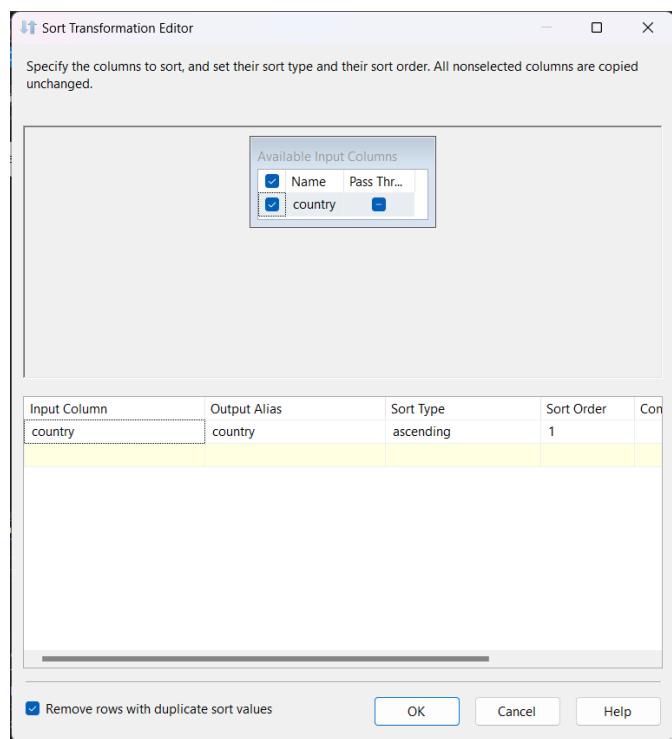
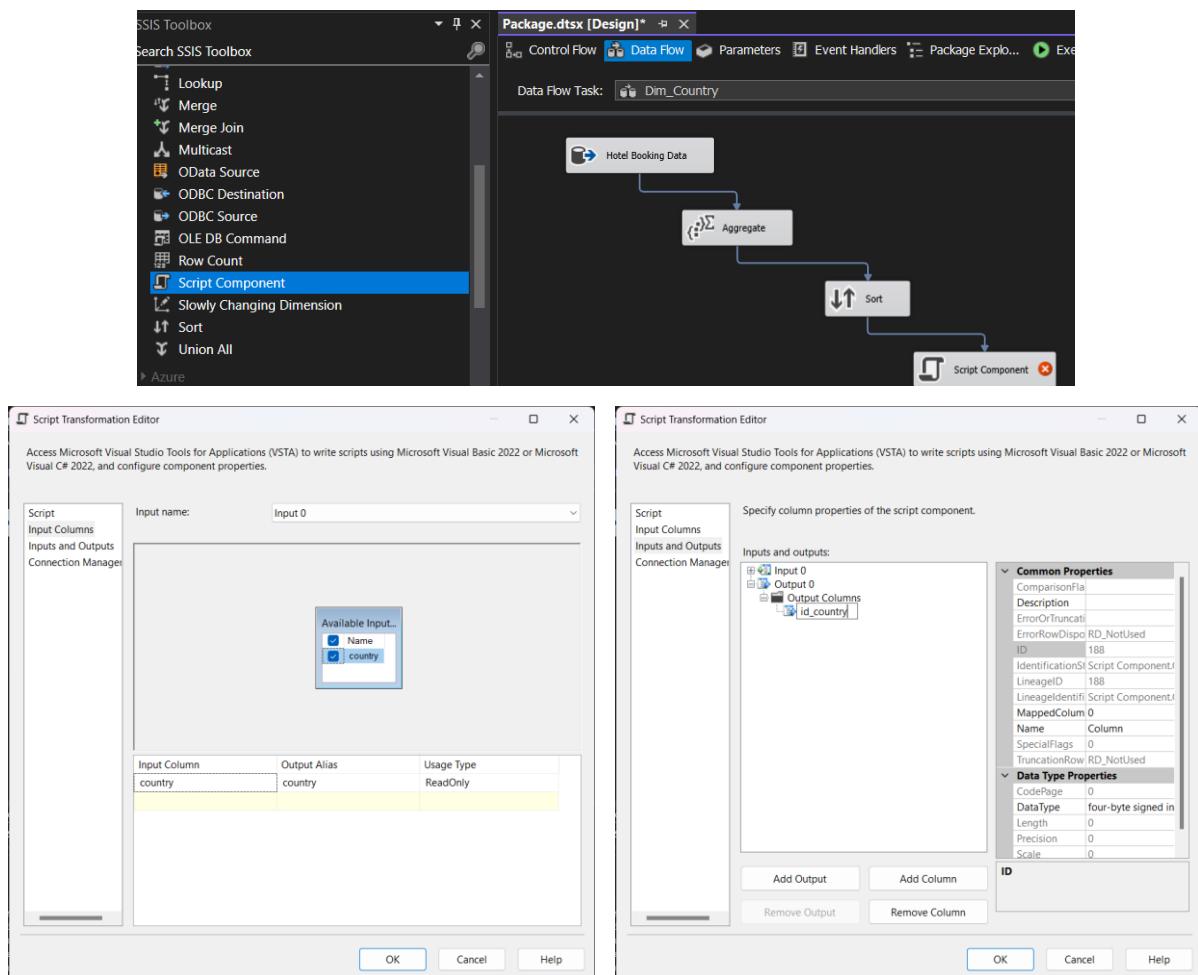
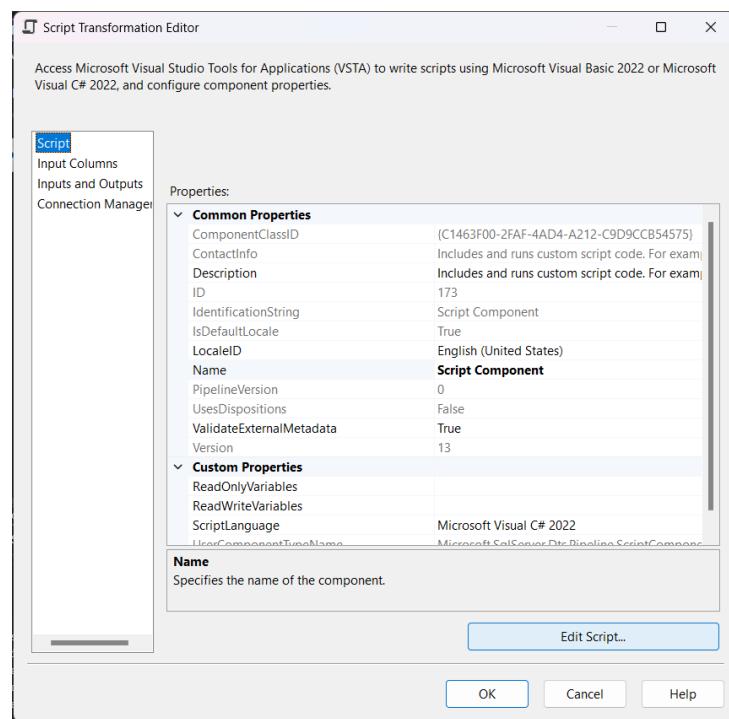


Figure 7-8. Sắp xếp các giá trị theo chiều tăng dần

- **Bước 7:** Kéo thả công cụ Script Component để tạo khóa chính id_country.





```
2 references
public override void Input0_ProcessInputRow(Input0Buffer Row)
{
    Row.idcountry = count;
    count++;
}
```

Figure 9-10-11-12-13. Sử dụng Script Component để tạo khóa chính id_country

- **Bước 8:** Chọn công cụ OLE DB Destination để tiến hành tạo bảng trong cơ sở dữ liệu DB_HotelBooking với tên gọi là Dim_Country.

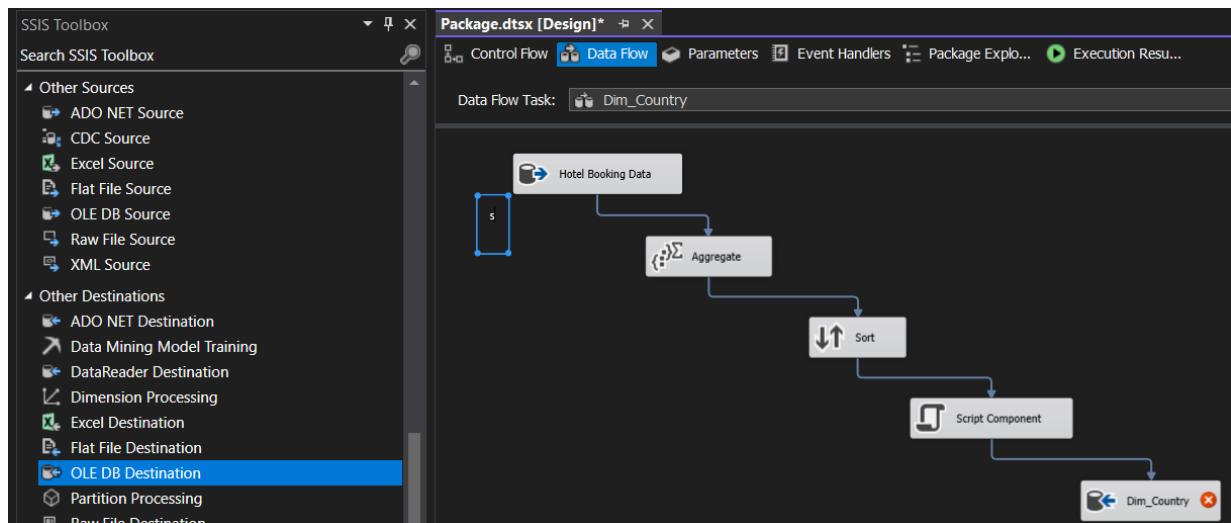


Figure 101. Sử dụng OLE DB Destination để tạo bảng

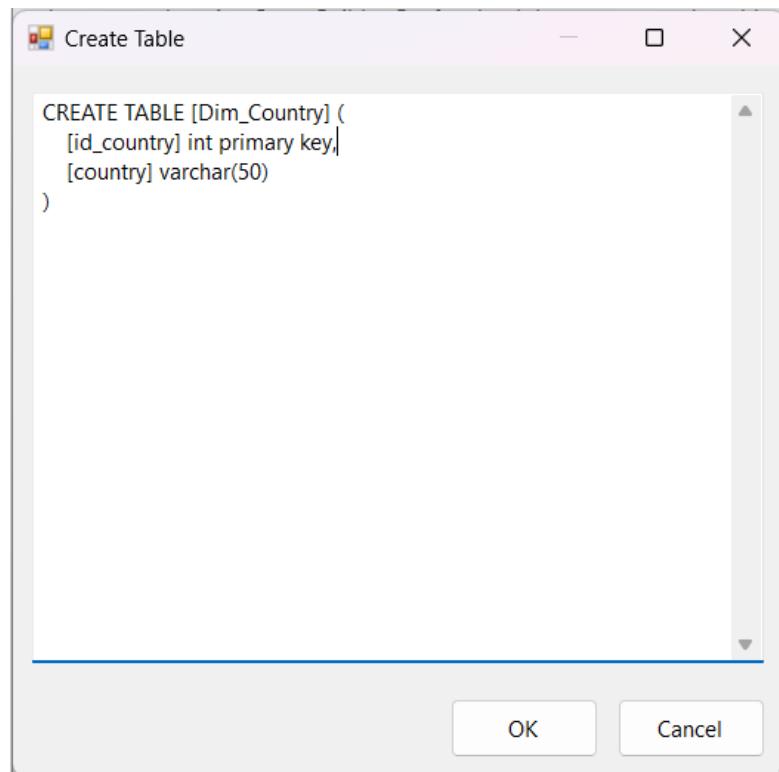


Figure 102. Tạo bảng Dim_Country

- **Bước 9:** Nhấn Mappings để kiểm tra kết nối và nhấn Ok để hoàn tất quá trình tạo bảng.

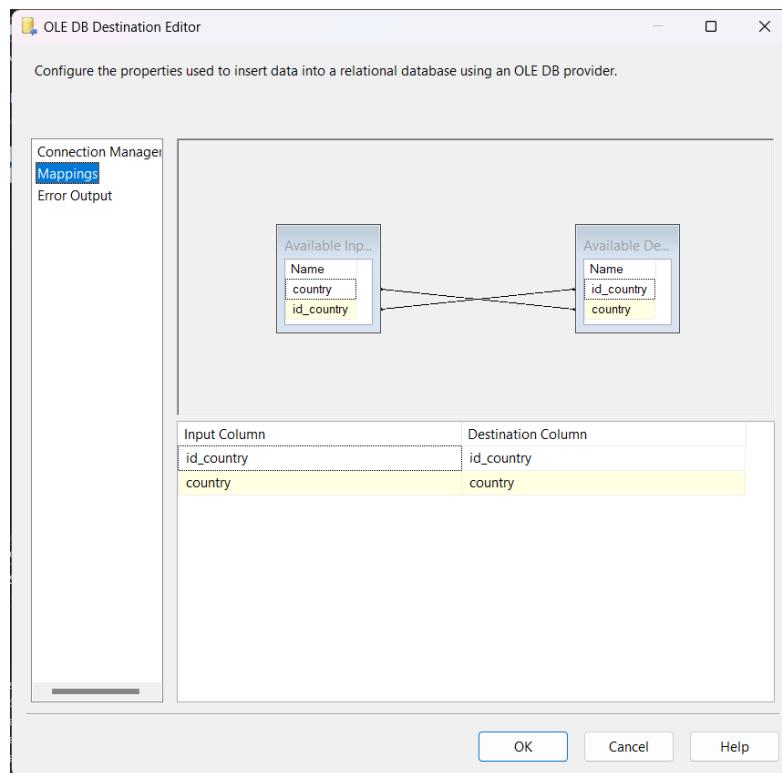


Figure 103. Quá trình Mappings dữ liệu

- **Bước 10:** Chọn Start để bắt đầu quá trình đổ dữ liệu. Khi đổ thành công, ta có kết quả như sau:

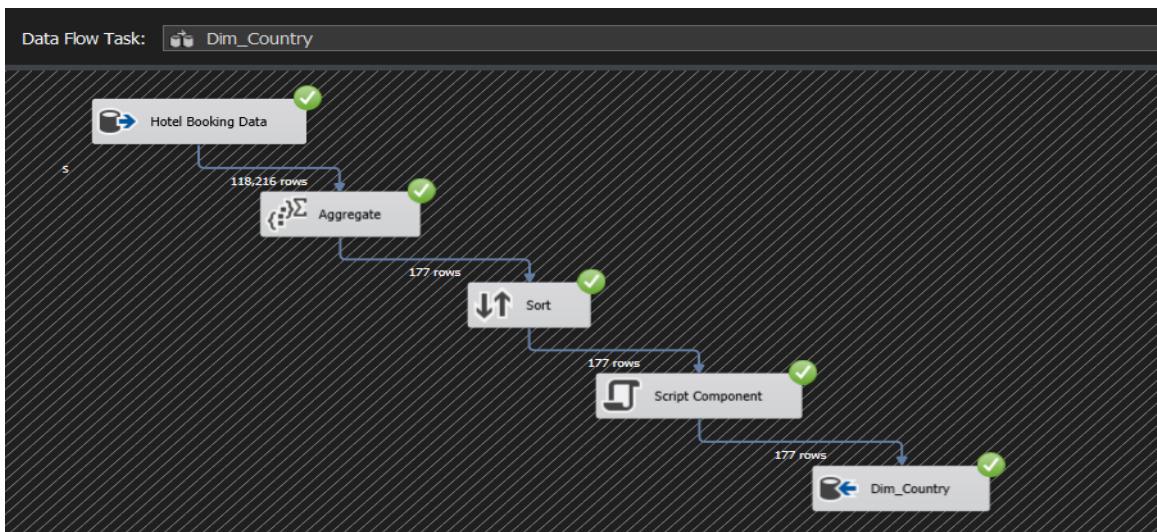


Figure 104. Hoàn thành đổ dữ liệu vào Dim_Country trong kho dữ liệu

- **Bước 11:** Kiểm tra bảng Dim_Country trên SQL Server.

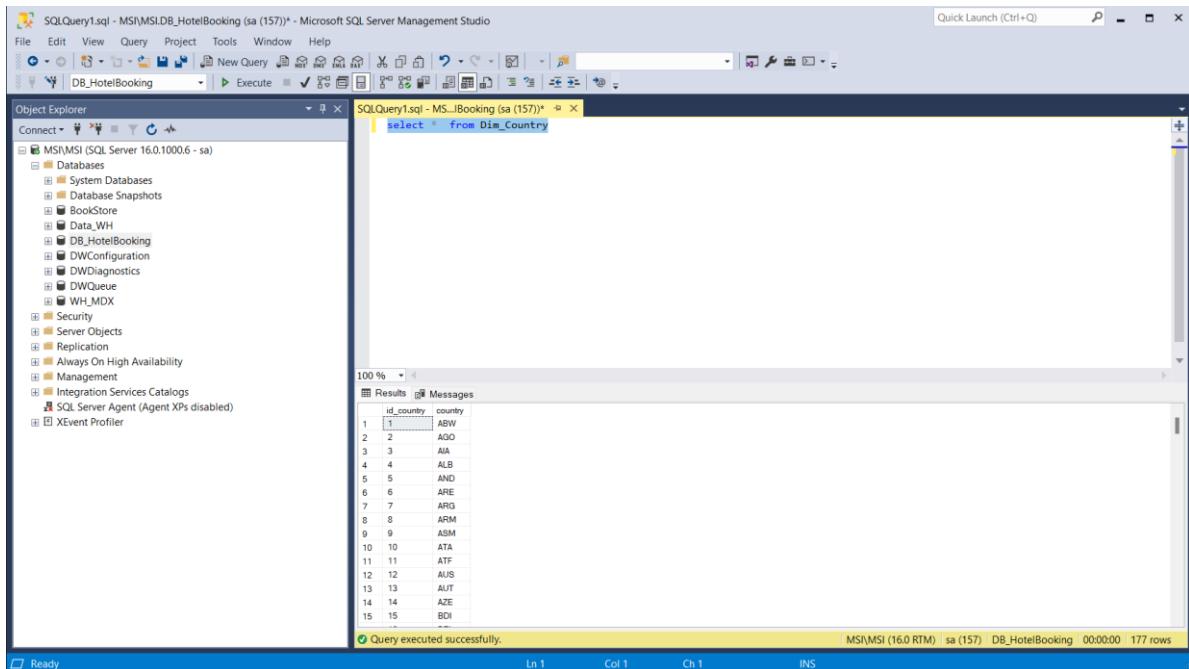


Figure 105. Kiểm tra bảng Dim_Country trong SQL Server

2.5.8. Bảng Dim_Customer

- **Bước 1:** Kéo chức năng Data Flow Task từ cột trái sang màn hình làm việc và đổi tên thành Dim_Customer và tạo đường liên kết từ Dim_Customer_Type và Dim_Country đến Dim_Customer để thực hiện theo thứ tự lần lượt là Dim_Customer_Type, Dim_Country và Dim_Customer.

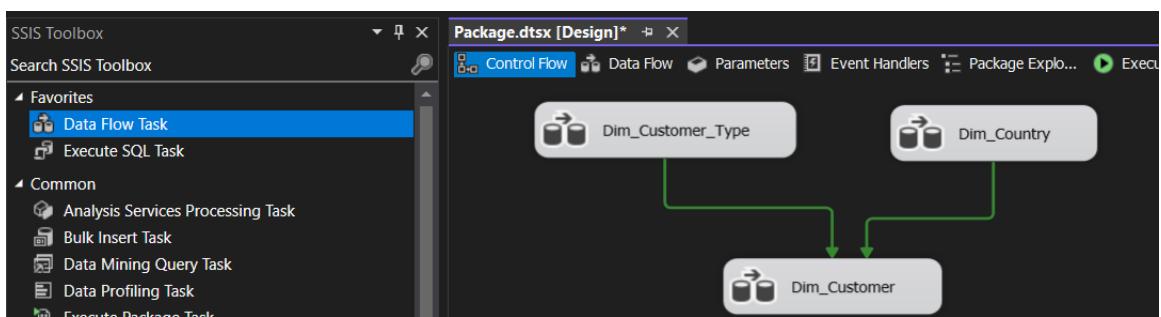


Figure 106. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Customer

- **Bước 2:** Nhấn double vào Data Flow Task vừa tạo và tìm kiếm chức năng OLE DB Source, kéo vào màn hình làm việc.

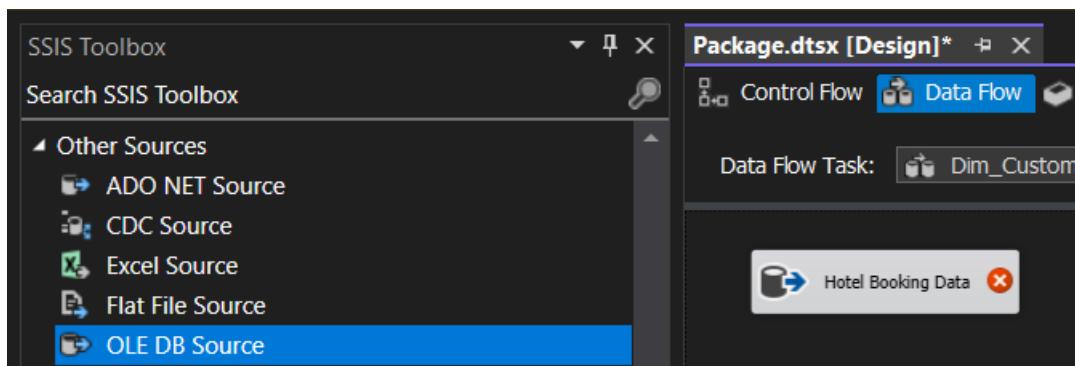


Figure 107. Kéo chức năng OLE DB Source vào màn hình làm việc

- **Bước 3:** Click double vào OLE DB Source và dẫn đến DB_HotelBooking. Sau đó chọn Name of the table or the view là [dbo].Hotel_Booking.

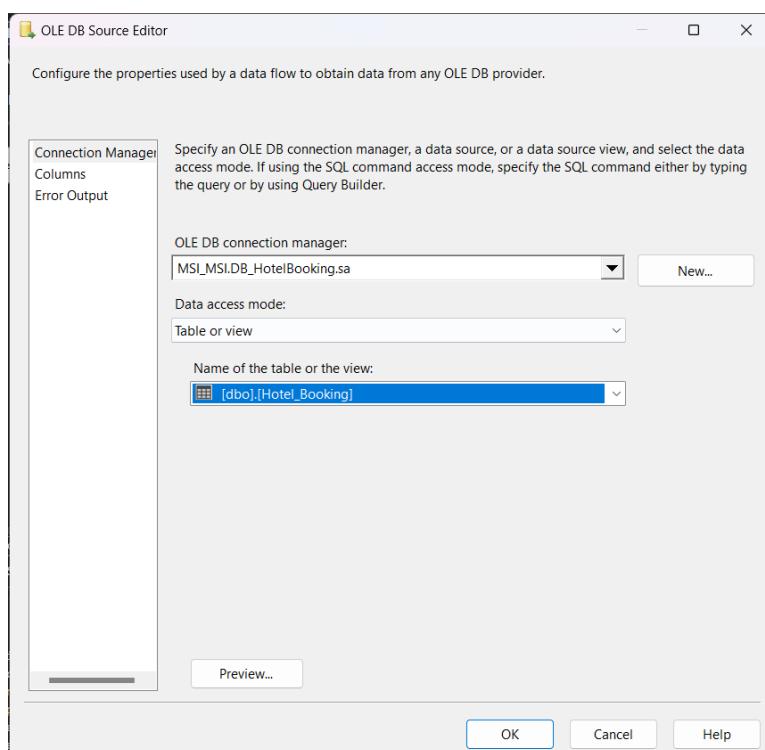


Figure 108. Chọn Table sẽ lấy dữ liệu

- **Bước 4:** Chọn Columns để lựa chọn các cột cần sử dụng và nhấn OK

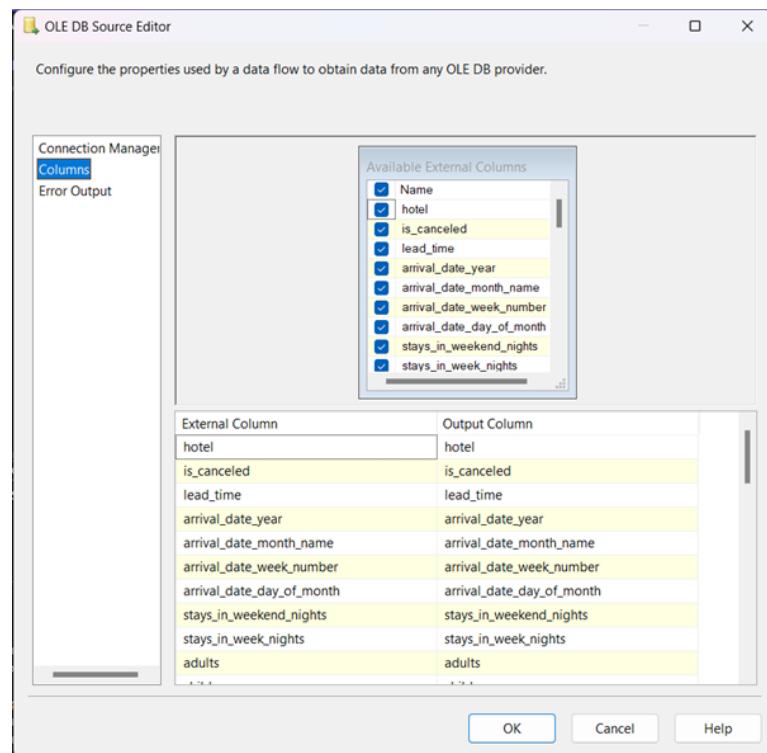


Figure 109. Chọn các column cần sử dụng

- **Bước 5:** Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau lên thuộc tính sử dụng là email, phone_number, name, country, customer_type.

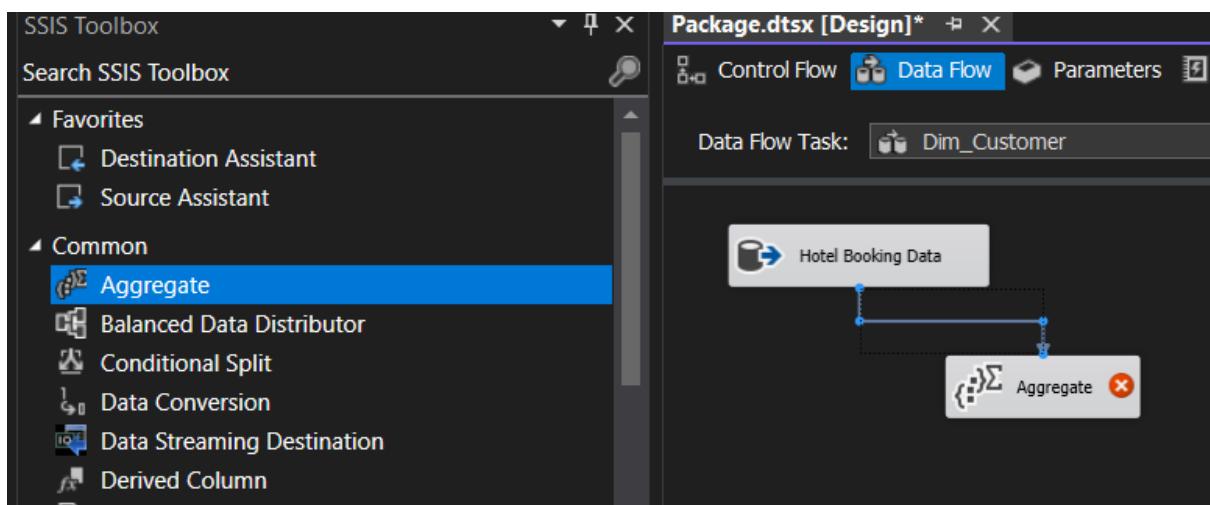


Figure 110. Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau

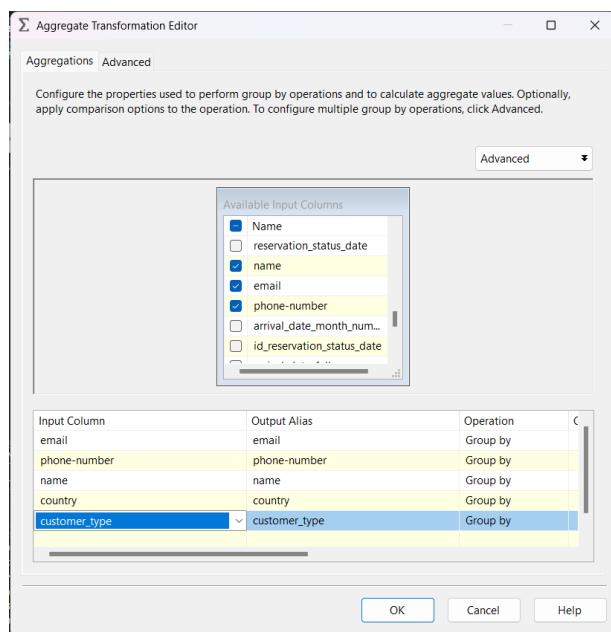


Figure 111. Chọn các thuộc tính cần gom nhóm

- **Bước 6:** Dùng công cụ Sort để sắp xếp các giá trị theo chiều tăng dần.

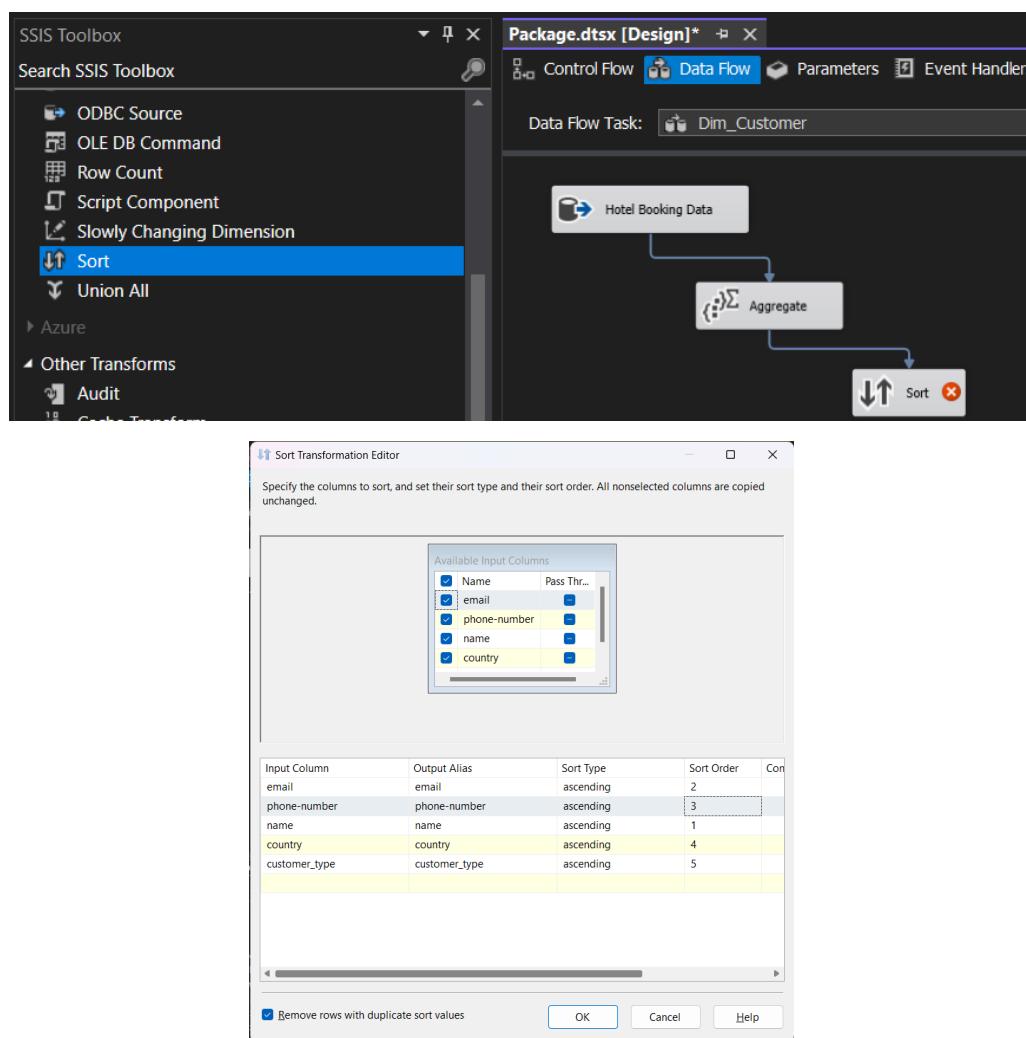


Figure 7-8. Sắp xếp các giá trị theo chiều tăng dần

- **Bước 7:** Tìm kiếm công cụ Lookup trên SSIS Toolbox và đổi tên thành Customer_Type_Lookup để tạo khóa ngoại id_customer_type đến bảng Dim_Customer.

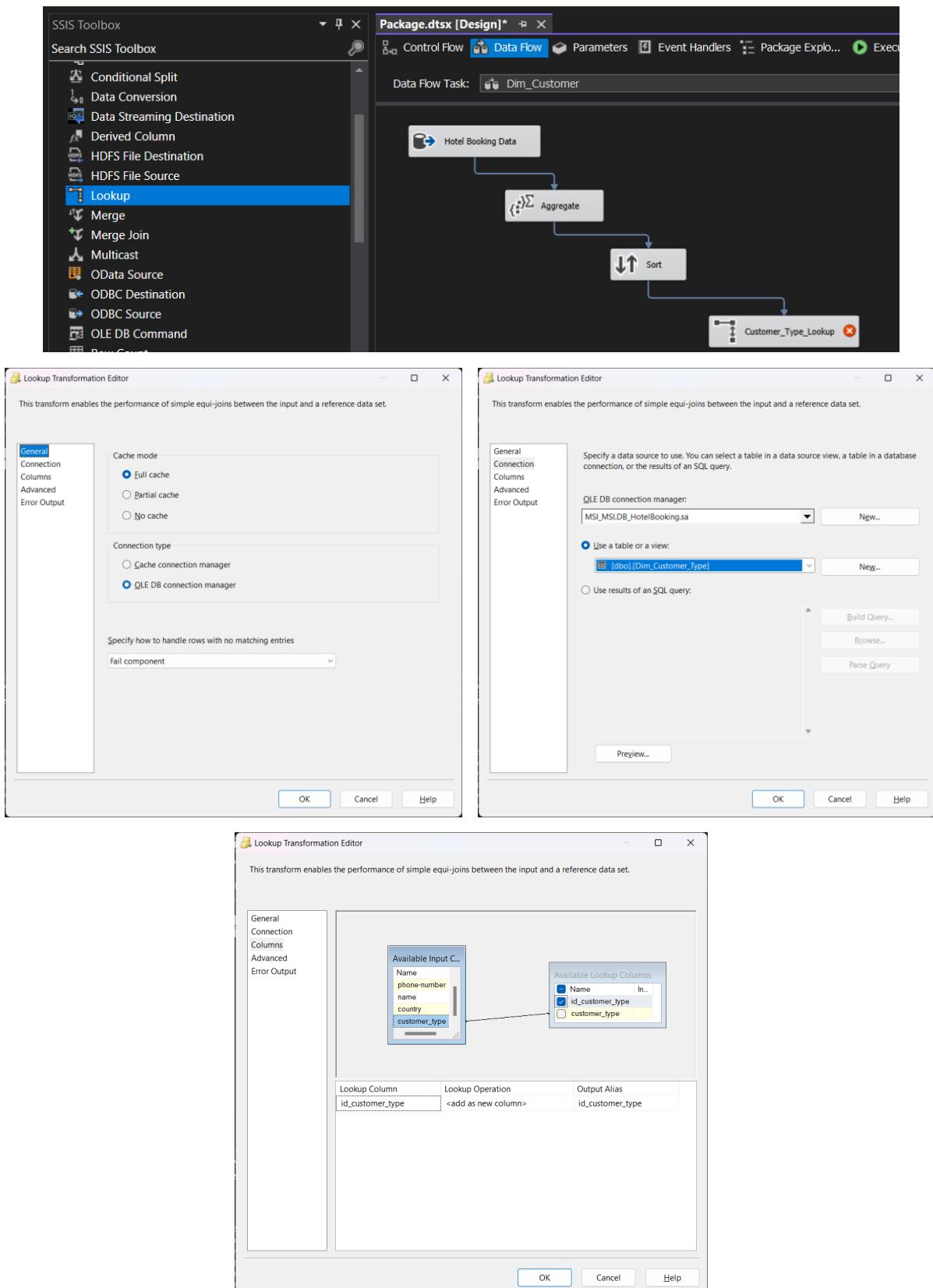


Figure 9-10-11-12. Tạo khóa ngoại id_customer_type đến bảng Dim_Customer

- **Bước 8:** Tìm kiếm công cụ Lookup trên SSIS Toolbox và đổi tên thành Country_Lookup để tạo khóa ngoại id_country đến bảng Dim_Customer.

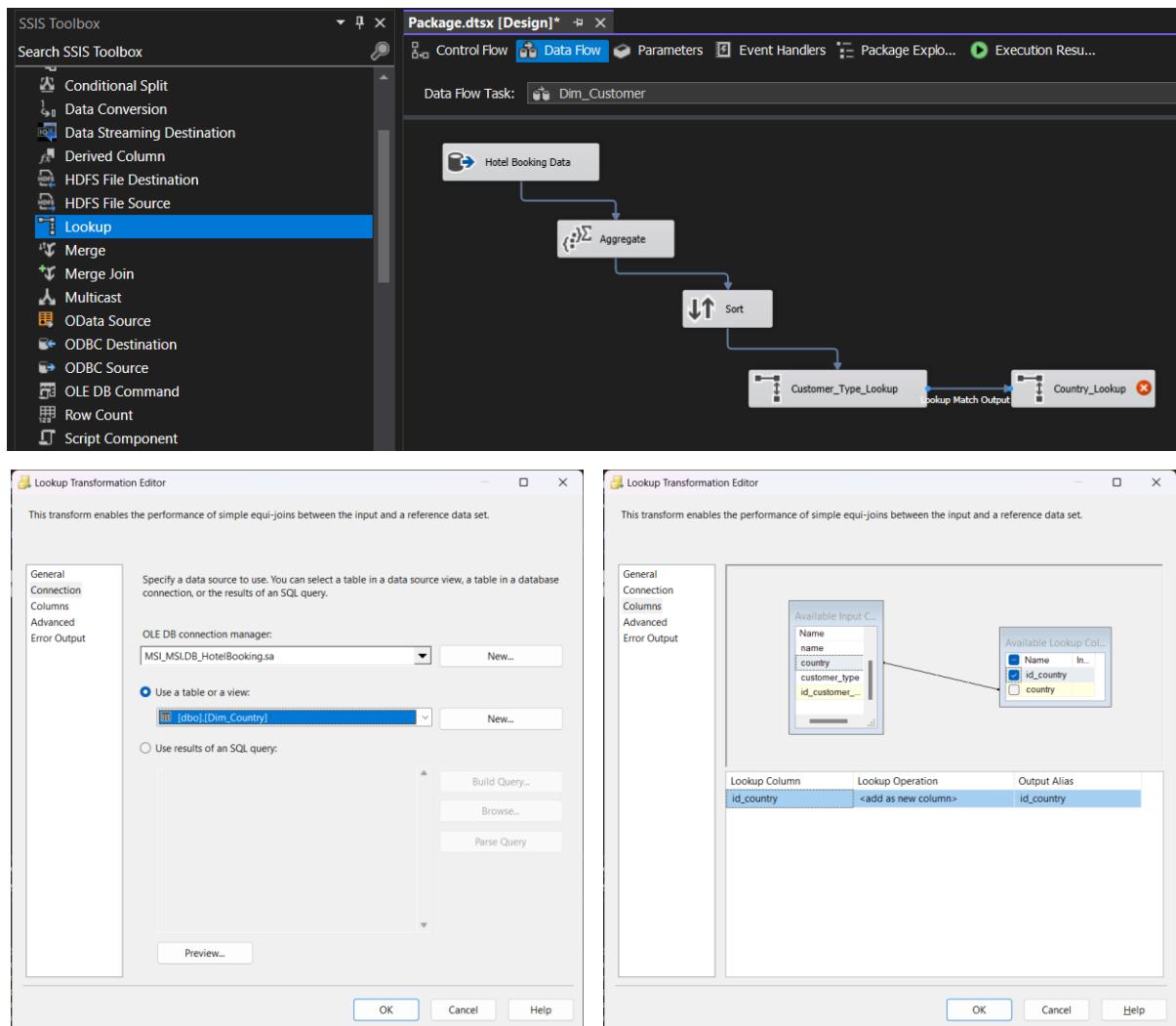
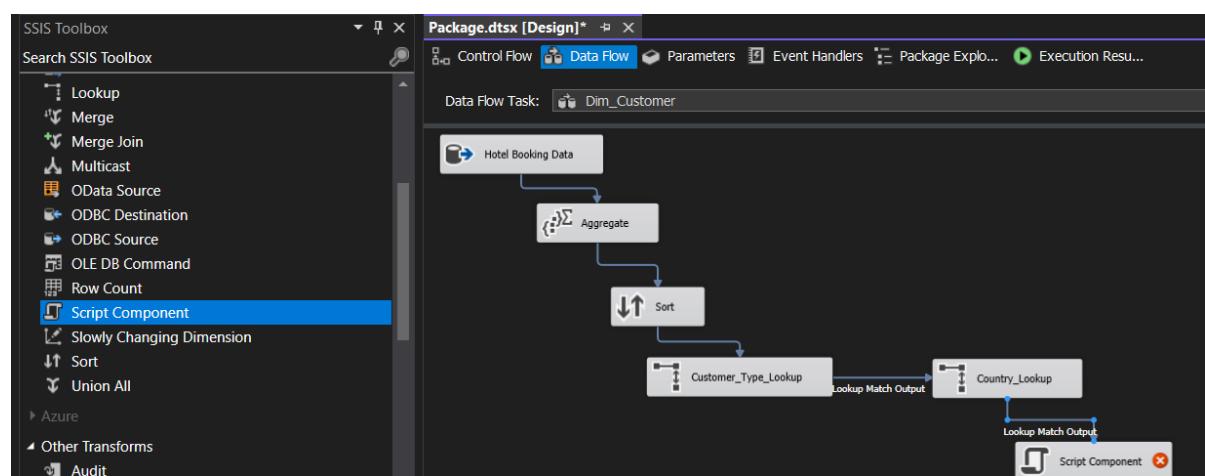


Figure 13-14-15. Tạo khóa ngoại id_country đến bảng Dim_Customer

- **Bước 9:** Kéo thả công cụ Script Component để tạo khóa chính id_customer.



IS217.N21 – Kho dữ liệu và phân tích hoạt động đặt phòng khách sạn

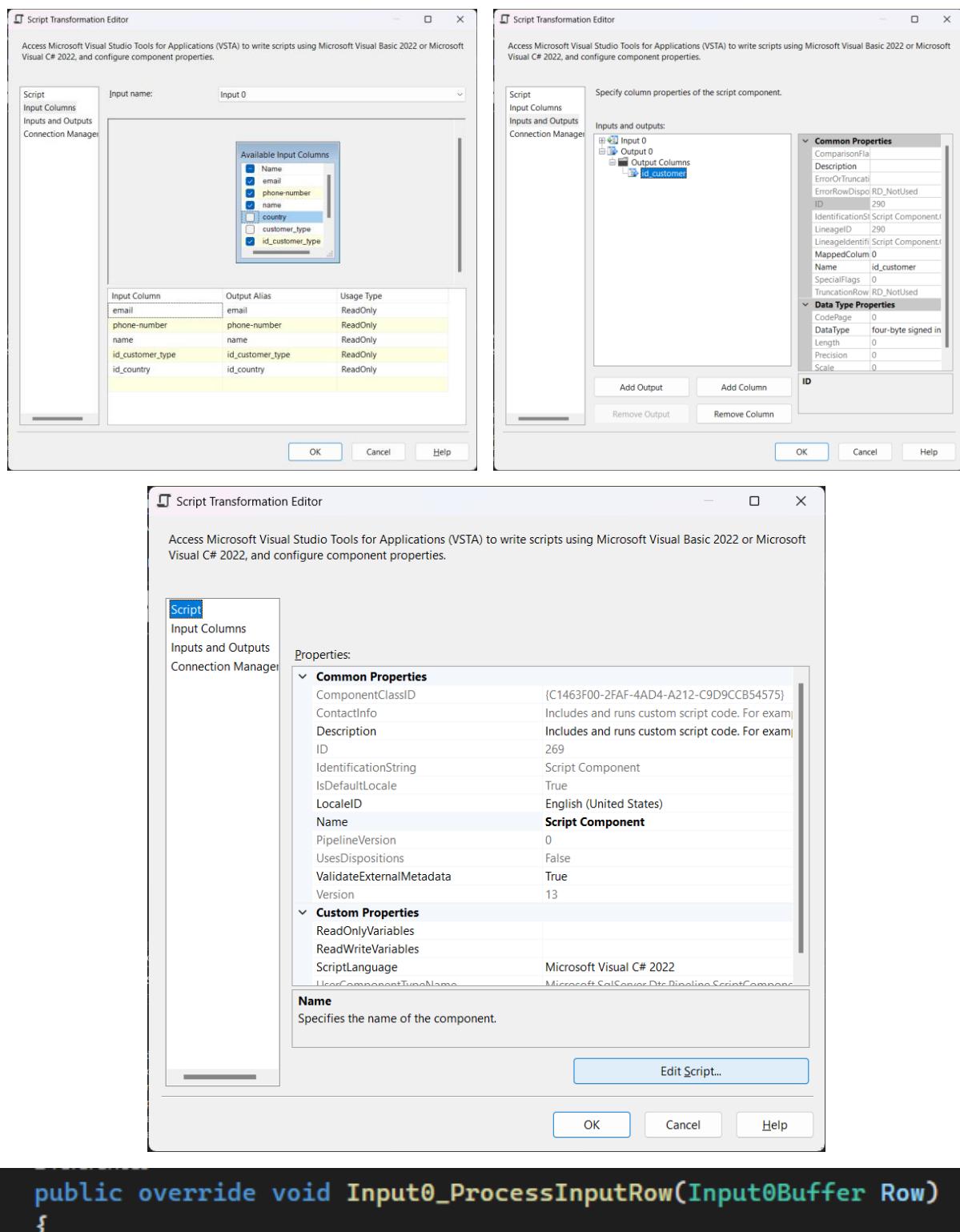


Figure 16-17-18-19-20. Sử dụng Script Component để tạo khóa chính id_customer

- **Bước 10:** Chọn công cụ OLE DB Destination để tiến hành tạo bảng trong cơ sở dữ liệu DB_HotelBooking với tên gọi là Dim_Customer.

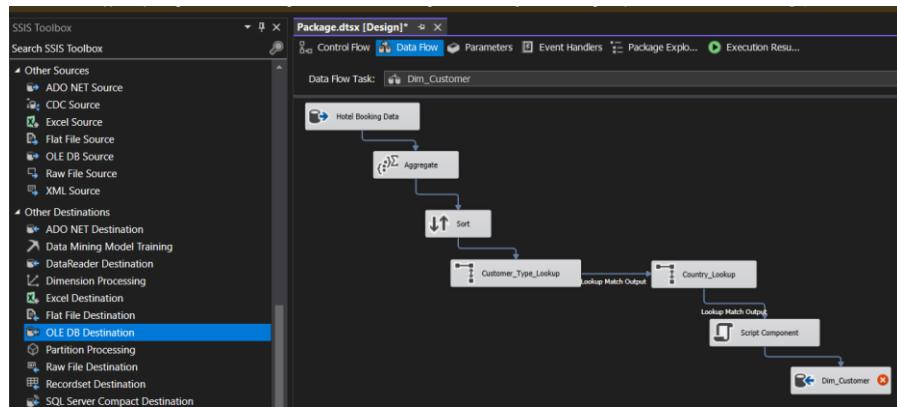


Figure 112. Sử dụng OLE DB Destination để tạo bảng

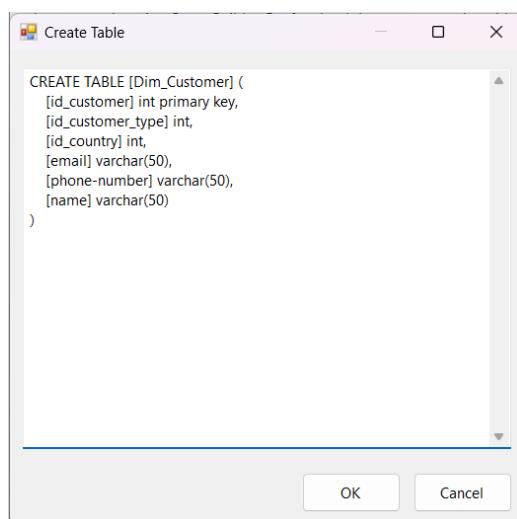


Figure 113. Tạo bảng Dim_Customer

- **Bước 11:** Nhấn Mappings để kiểm tra kết nối và nhấn Ok để hoàn tất quá trình tạo bảng.

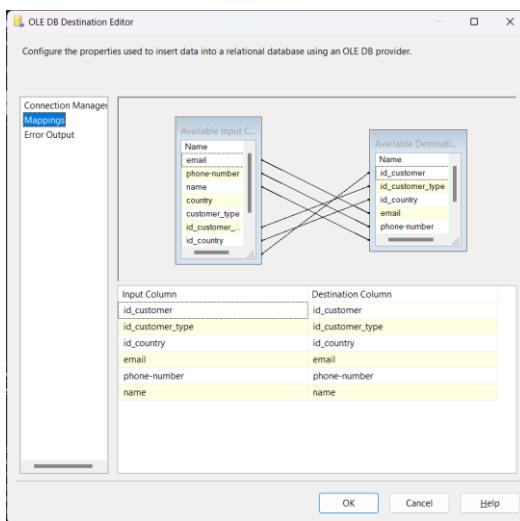


Figure 114. Quá trình Mappings dữ liệu

- **Bước 12:** Chọn Start để bắt đầu quá trình đổ dữ liệu. Khi đổ thành công, ta có kết quả như sau:

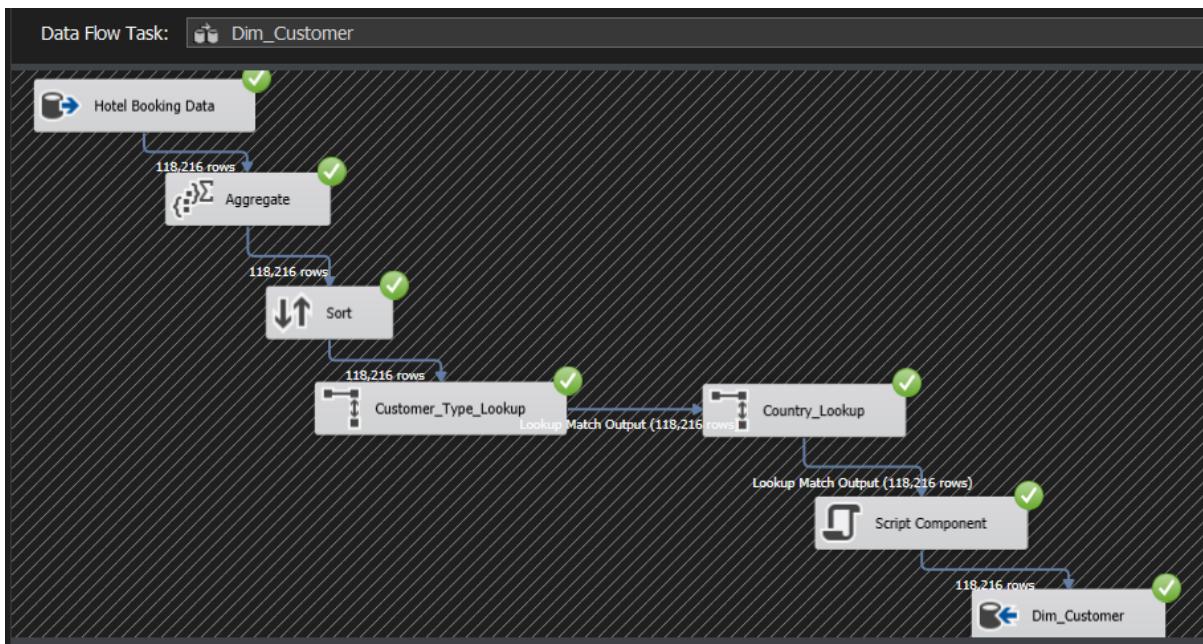


Figure 115. Hoàn thành đổ dữ liệu vào Dim_Customer trong kho dữ liệu

- **Bước 13:** Kiểm tra bảng Dim_Customer trên SQL Server.

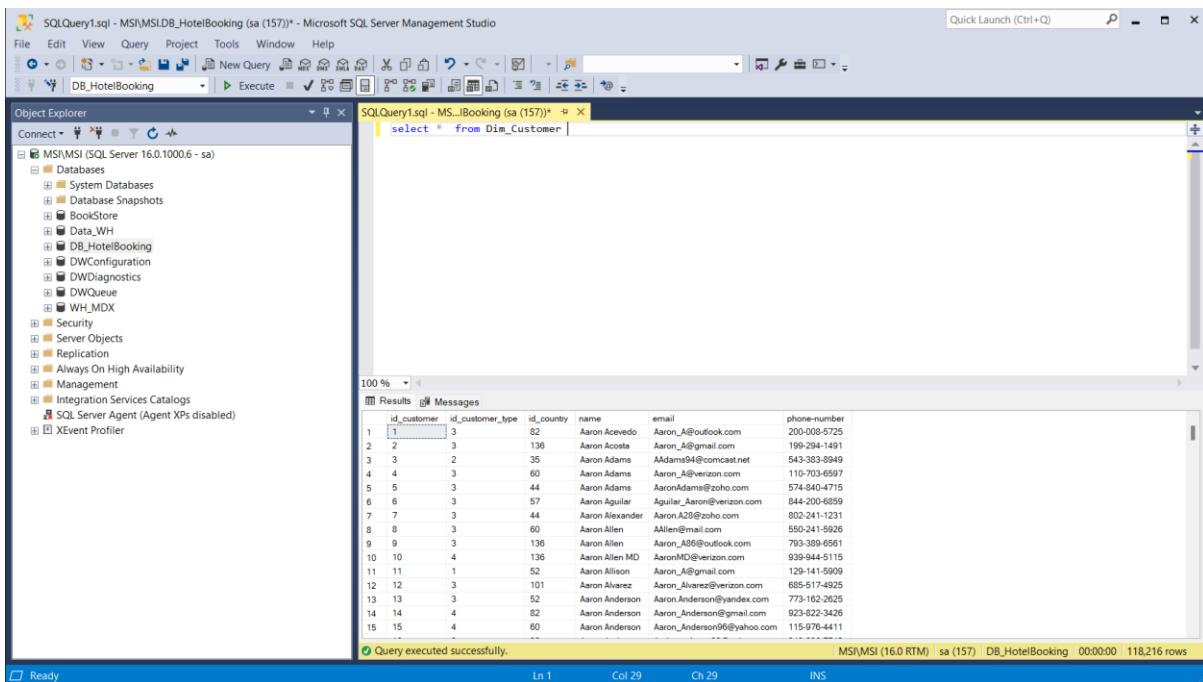


Figure 116. Kiểm tra bảng Dim_Customer trong SQL Server

2.5.9. Bảng Dim_Year

- **Bước 1:** Kéo chức năng Data Flow Task từ cột trái sang màn hình làm việc và đổi tên thành Dim_Year.

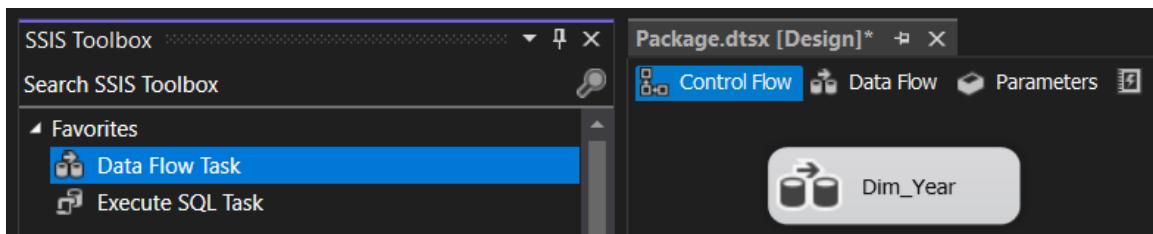


Figure 117. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Year

- **Bước 2:** Nhấn double vào Data Flow Task vừa tạo và tìm kiếm chức năng OLE DB Source, kéo vào màn hình làm việc

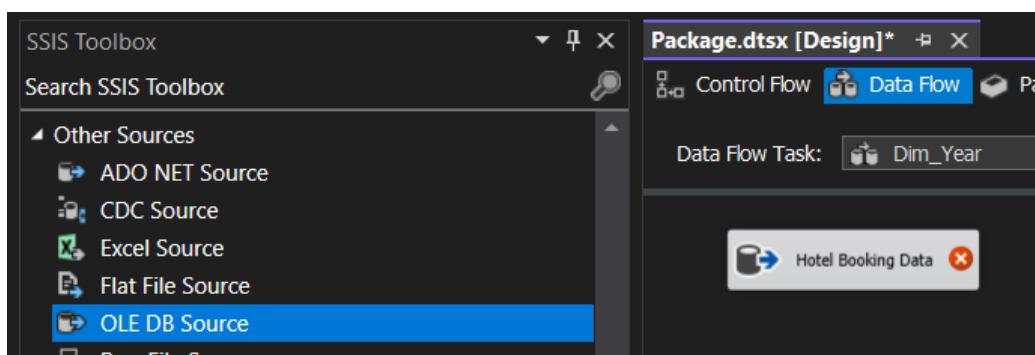


Figure 118. Kéo chức năng OLE DB Source vào màn hình làm việc 1

- **Bước 3:** Click double vào OLE DB Source và dẫn đến DB_HotelBooking. Sau đó chọn Name of the table or the view là [dbo].Hotel_Booking.

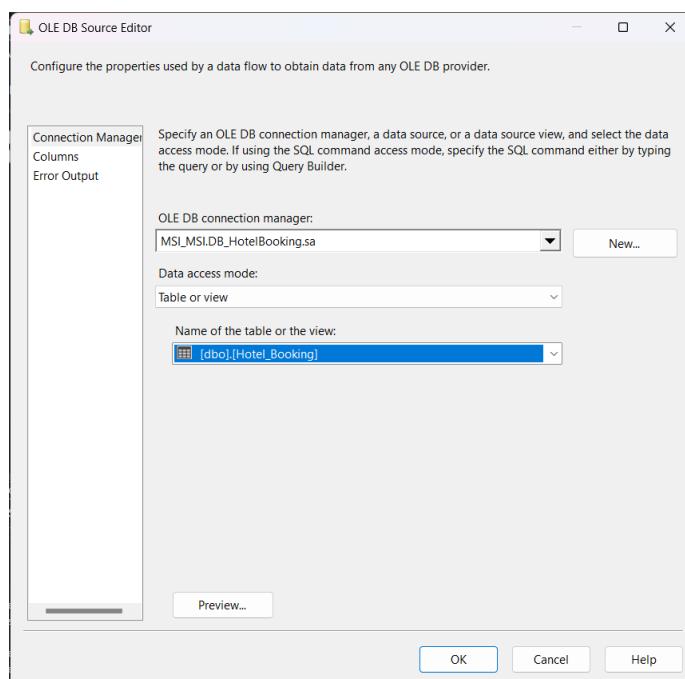


Figure 119. Chọn Table sẽ lấy dữ liệu

- **Bước 4:** Chọn Columns để lựa chọn các cột cần sử dụng và nhấn OK

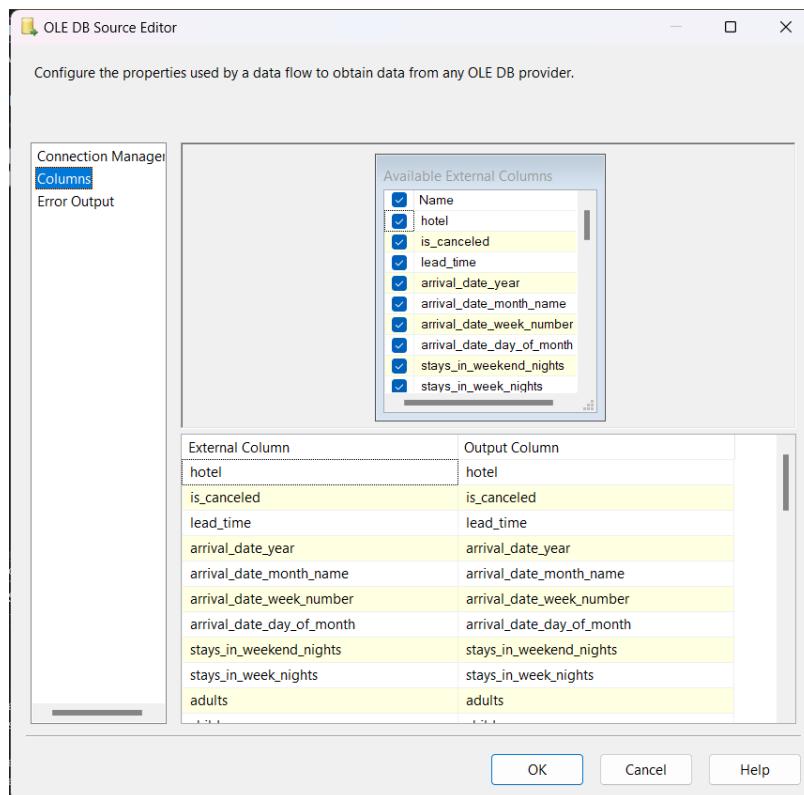


Figure 120. Chọn các column cần sử dụng

- **Bước 5:** Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau lên thuộc tính sử dụng là reservation_status_date.

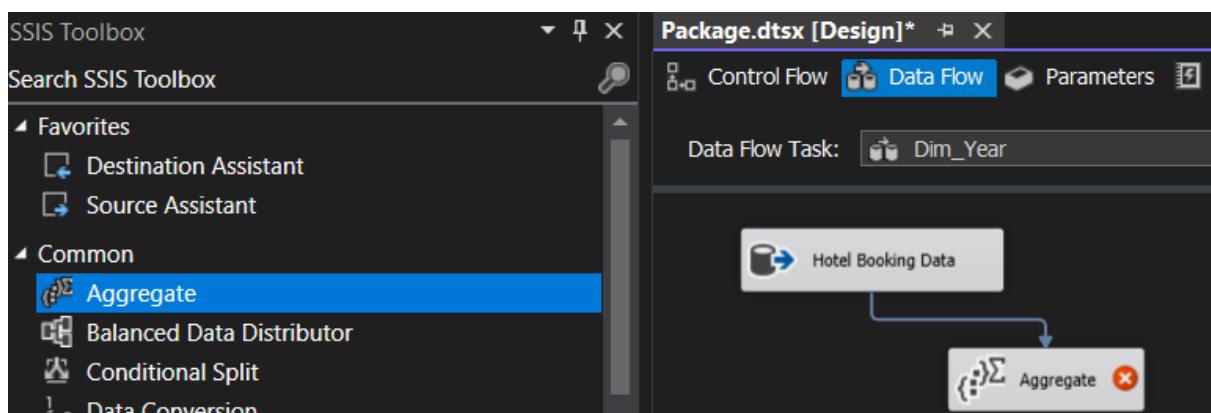


Figure 121. Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau

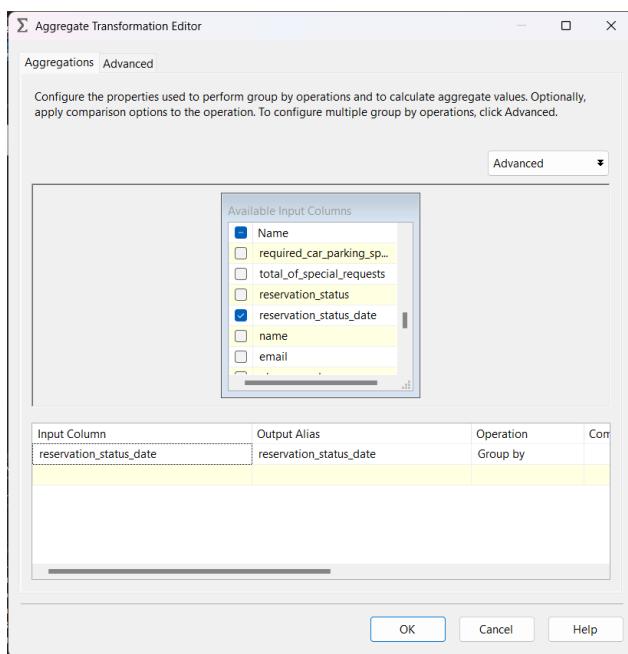


Figure 122. Chọn các thuộc tính cần gom nhóm

- **Bước 6:** Dùng công cụ Derived Column để lấy được thuộc tính year từ reservation_status_date.

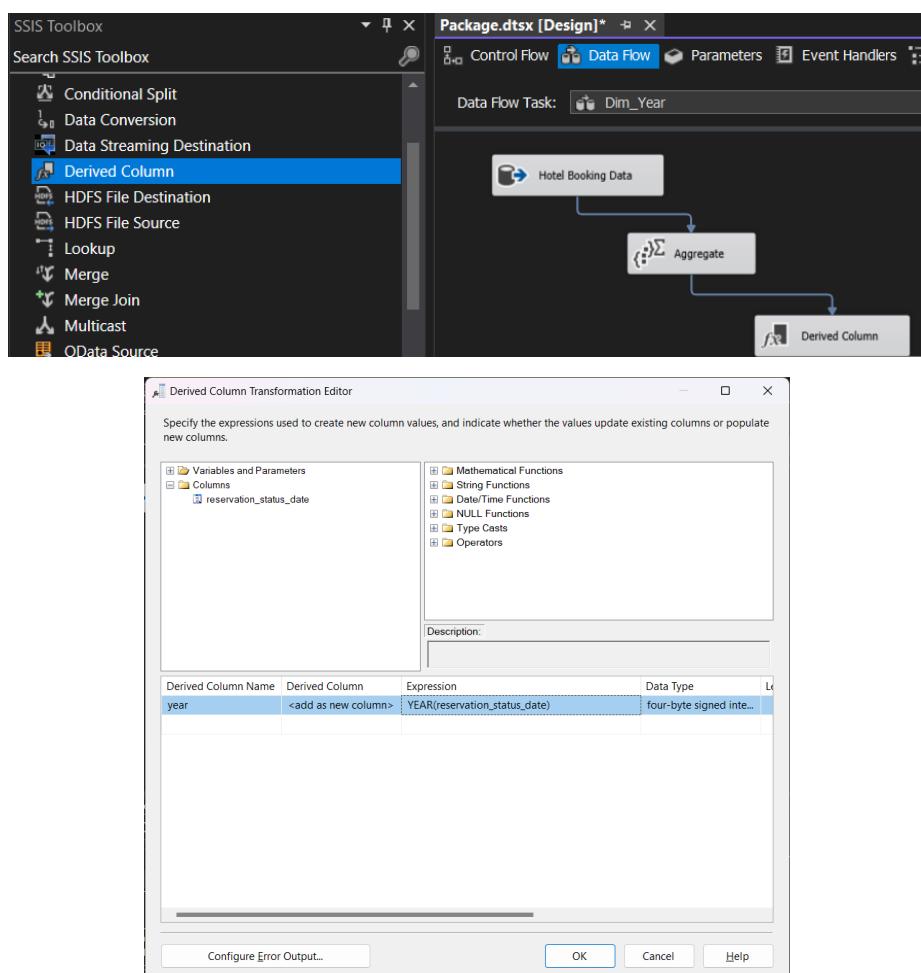


Figure 7-8. Tạo thuộc tính mới year

- **Bước 7:** Dùng công cụ Sort để sắp xếp các giá trị theo chiều tăng dần

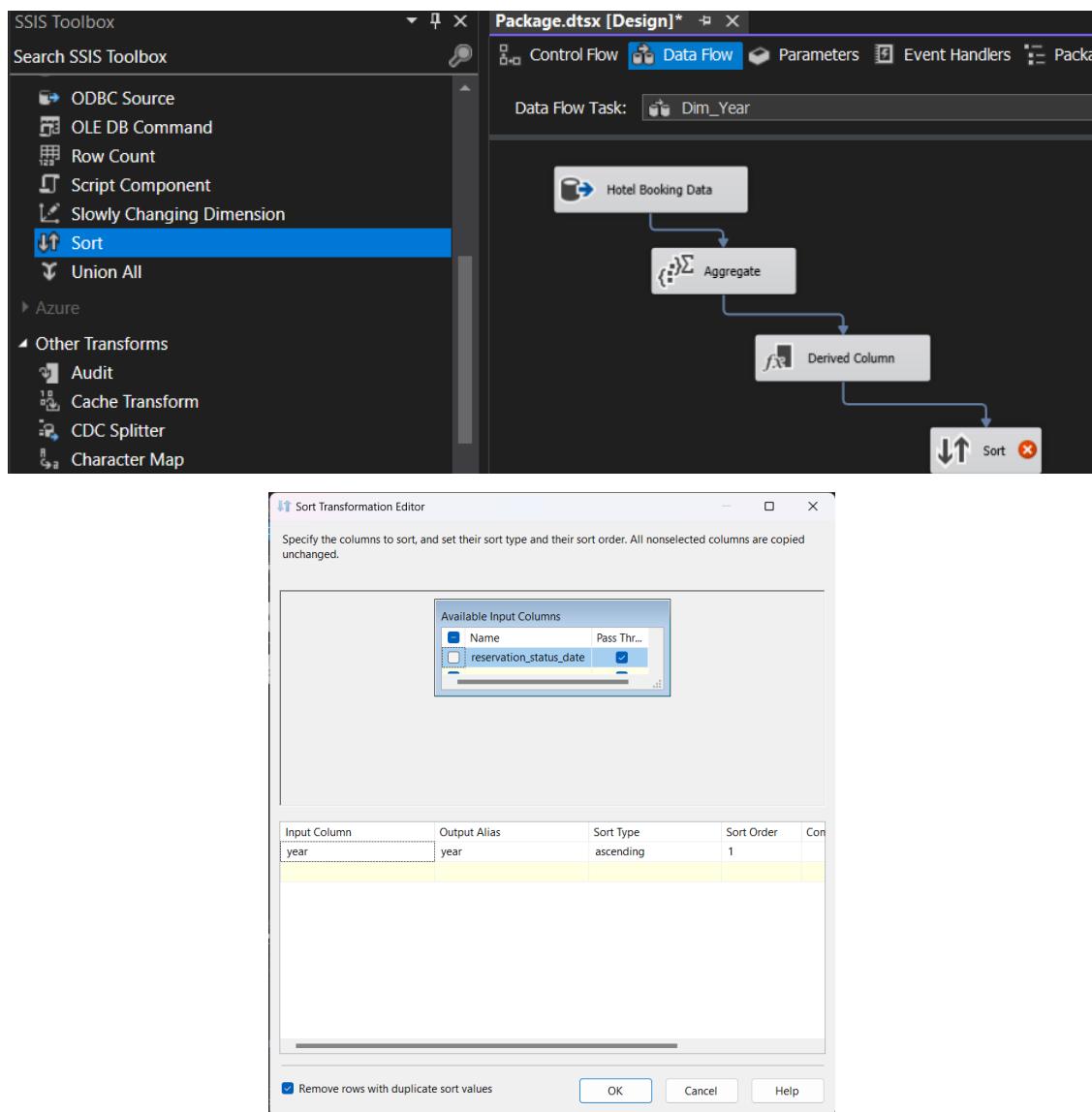
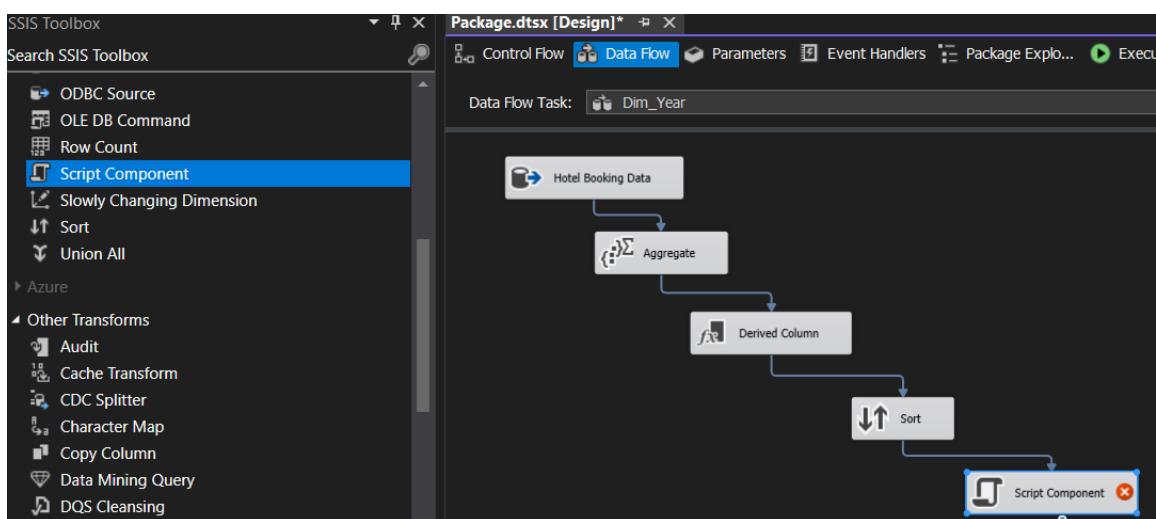


Figure 9-10. Sắp xếp các giá trị theo chiều tăng dần

- **Bước 8:** Kéo thả công cụ Script Component để tạo khóa chính id_year.



IS217.N21 – Kho dữ liệu và phân tích hoạt động đặt phòng khách sạn

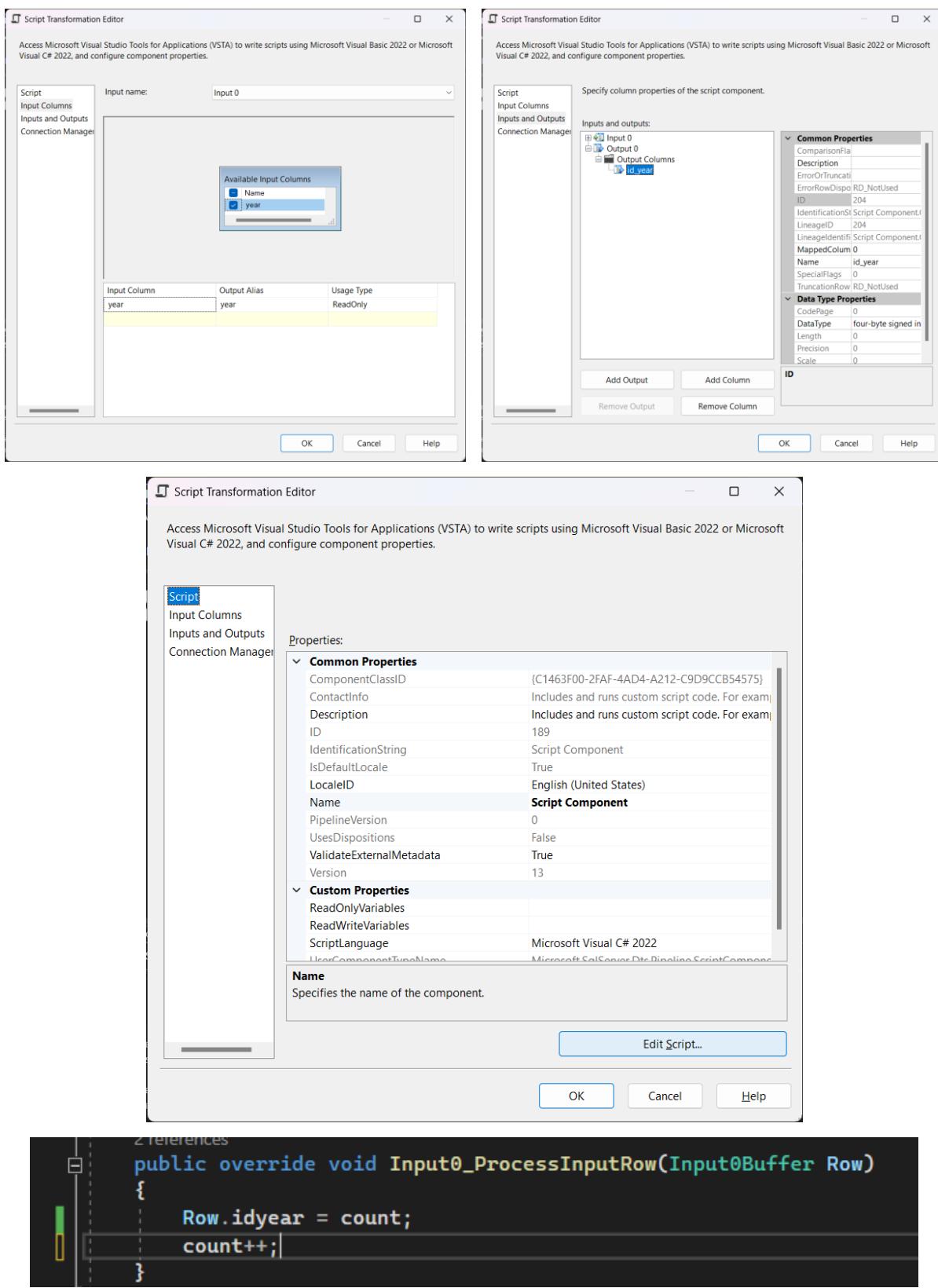


Figure 11-12-13-14-15. Sử dụng Script Component để tạo khóa chính id_year

- **Bước 9:** Chọn công cụ OLE DB Destination để tiến hành tạo bảng trong cơ sở dữ liệu DB_HotelBooking với tên gọi là Dim_Year.

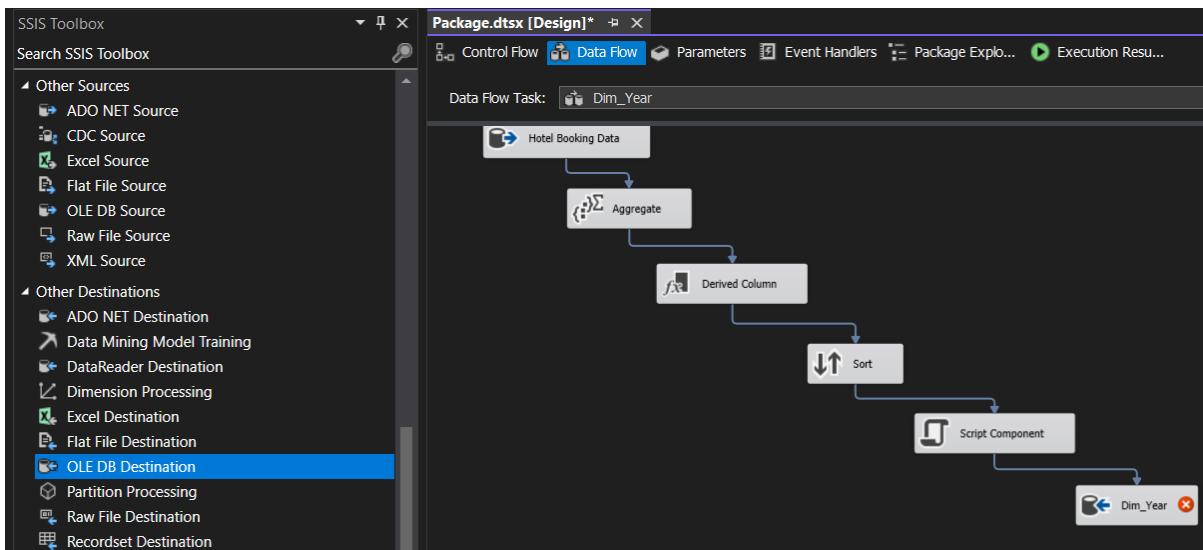


Figure 123. Sử dụng OLE DB Destination để tạo bảng

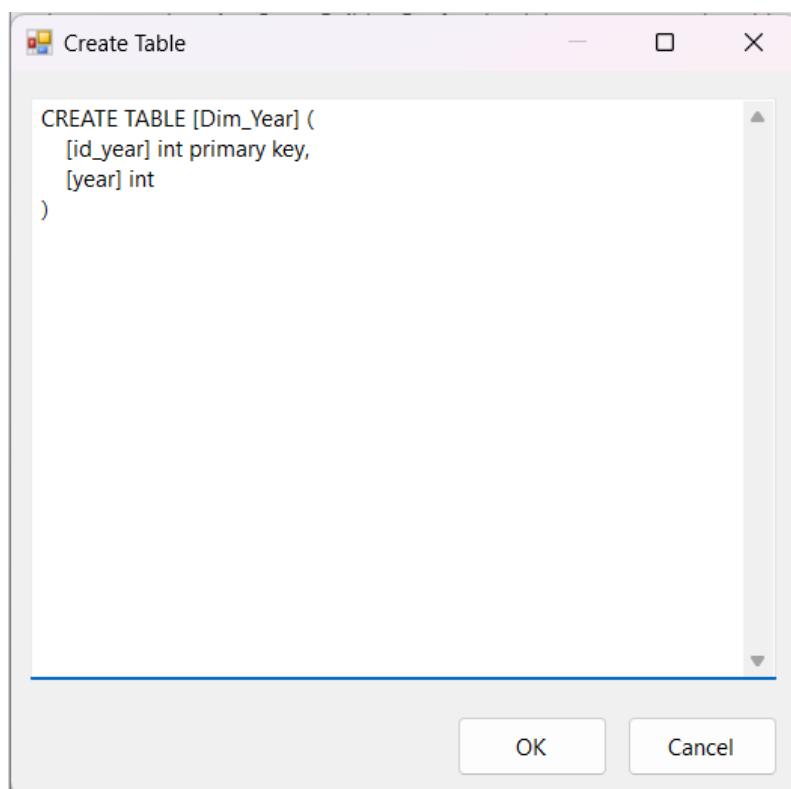


Figure 124. Tạo bảng Dim_Year

- **Bước 10:** Nhấn Mappings để kiểm tra kết nối và nhấn Ok để hoàn tất quá trình tạo bảng.

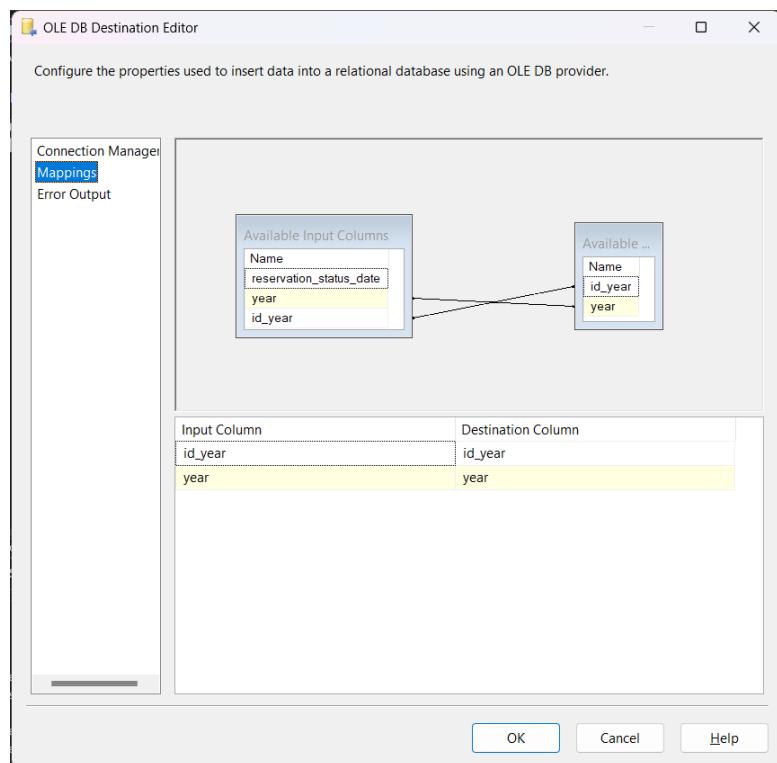


Figure 125. Quá trình Mappings dữ liệu

- **Bước 11:** Chọn Start để bắt đầu quá trình đổ dữ liệu. Khi đổ thành công, ta có kết quả như sau:

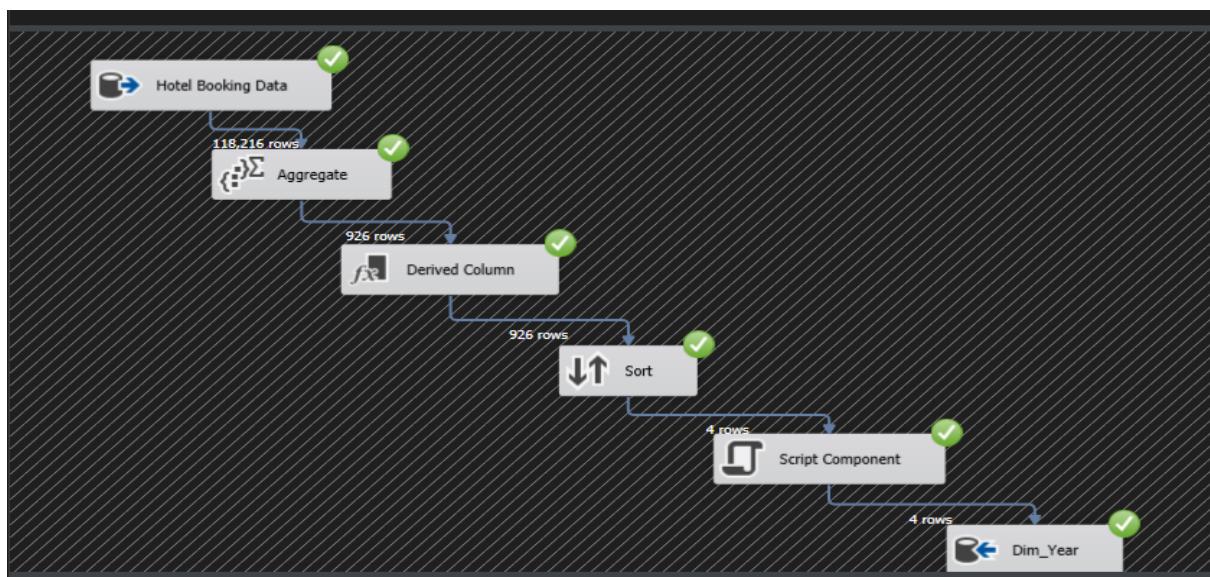


Figure 126. Hoàn thành đổ dữ liệu vào Dim_Year trong kho dữ liệu

- **Bước 12:** Kiểm tra bảng Dim_Year trên SQL Server.

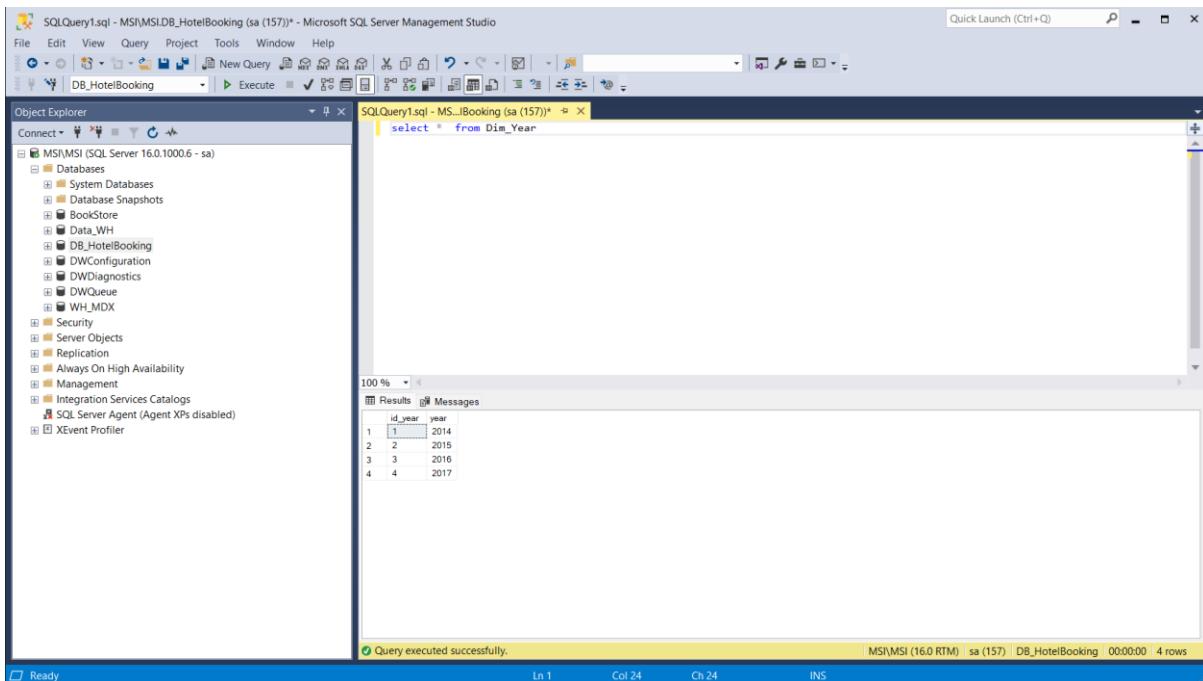


Figure 127. Kiểm tra bảng Dim_Year trong SQL Server

2.5.10. Bảng Dim_Quarter

- **Bước 1:** Kéo chức năng Data Flow Task từ cột trái sang màn hình làm việc và đổi tên thành Dim_Quarter.

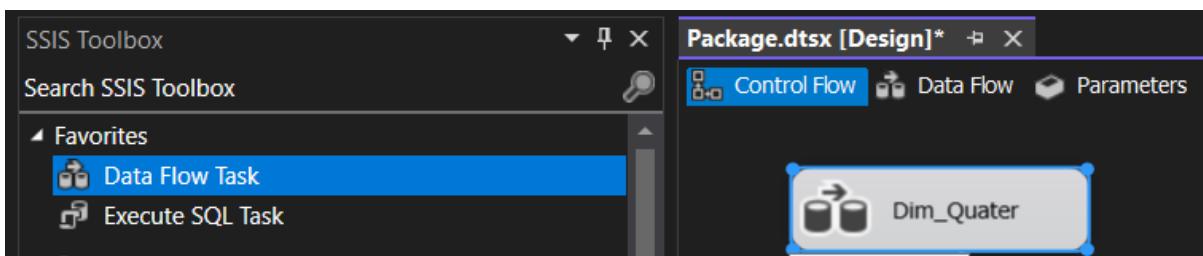


Figure 128. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Quarter

- **Bước 2:** Nhấn double vào Data Flow Task vừa tạo và tìm kiếm chức năng OLE DB Source, kéo vào màn hình làm việc.

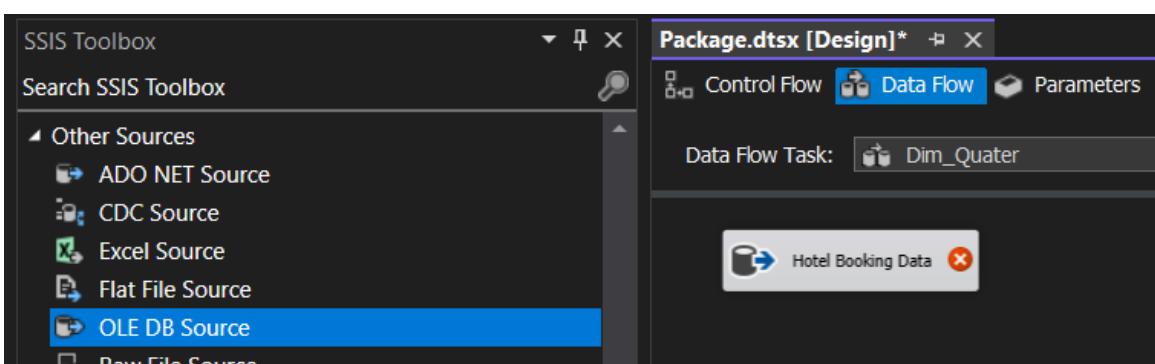


Figure 129. Kéo chức năng OLE DB Source vào màn hình làm việc

- **Bước 3:** Click double vào OLE DB Source và dẫn đến DB_HotelBooking. Sau đó chọn Name of the table or the view là [dbo].Hotel_Booking.

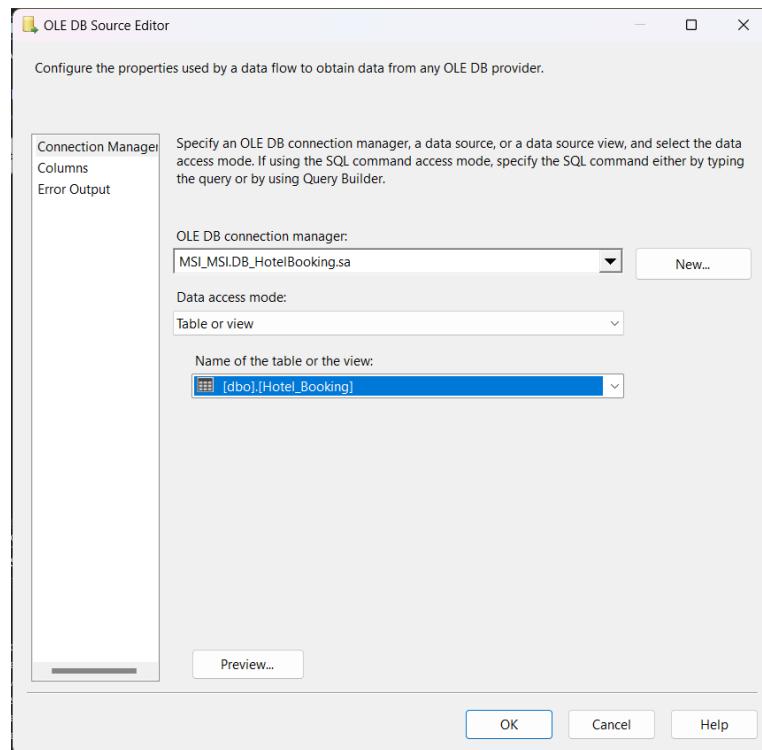


Figure 130. Chọn Table sẽ lấy dữ liệu

- **Bước 4:** Chọn Columns để lựa chọn các cột cần sử dụng và nhấn OK.

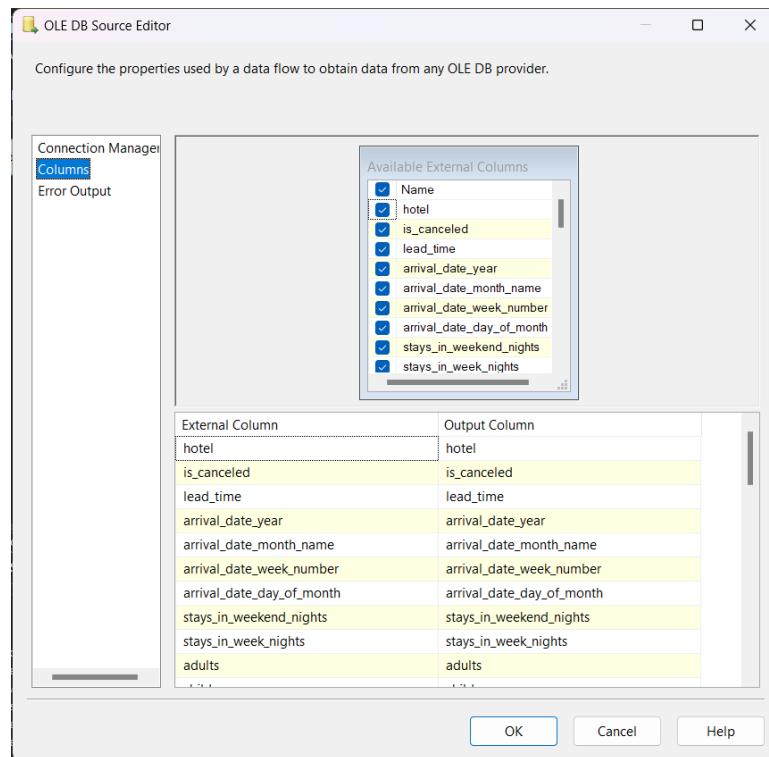


Figure 131. Chọn các column cần sử dụng

- **Bước 5:** Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau lên thuộc tính sử dụng là arrival_date_quarter.

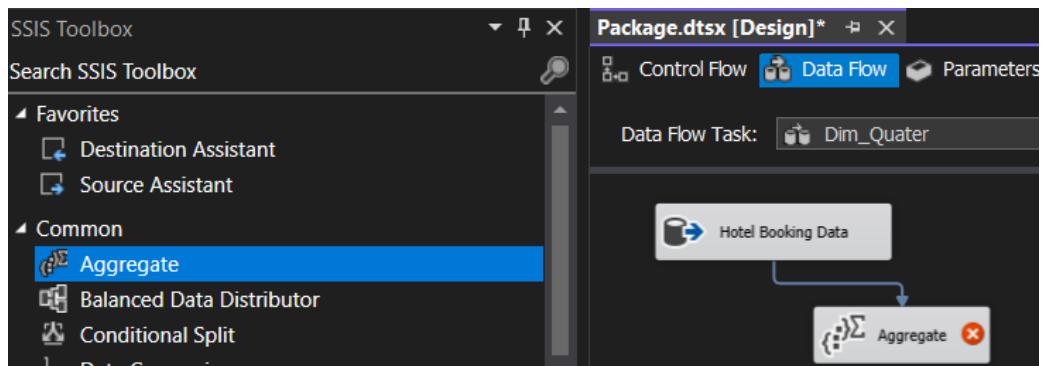


Figure 132. Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau

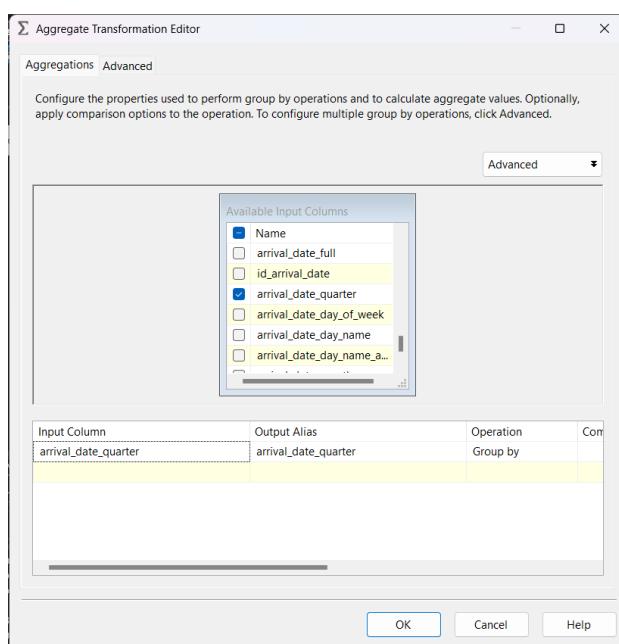
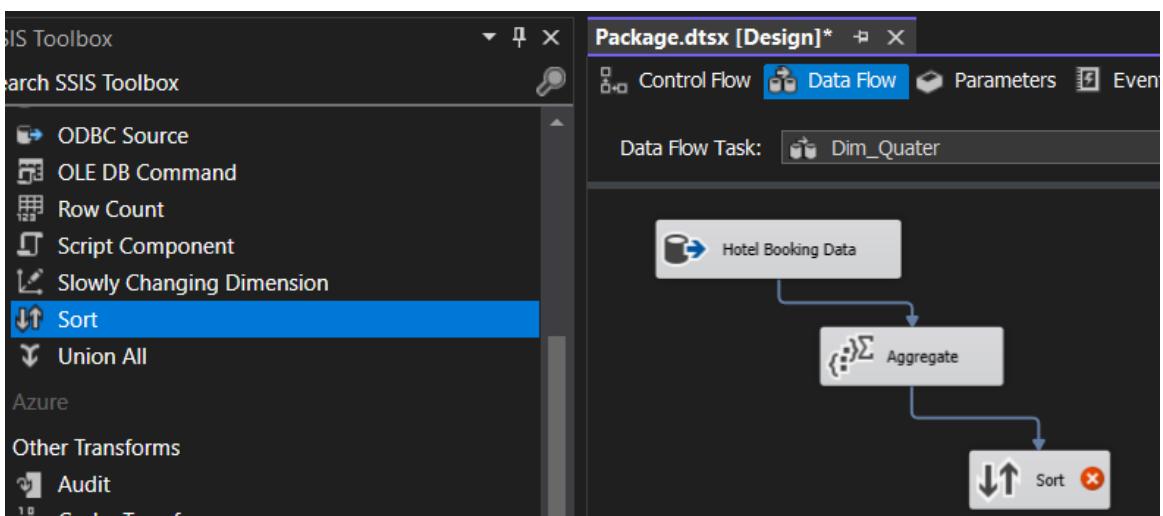


Figure 133. Chọn các thuộc tính cần gom nhóm

- **Bước 6:** Dùng công cụ Sort để sắp xếp các giá trị theo chiều tăng dần.



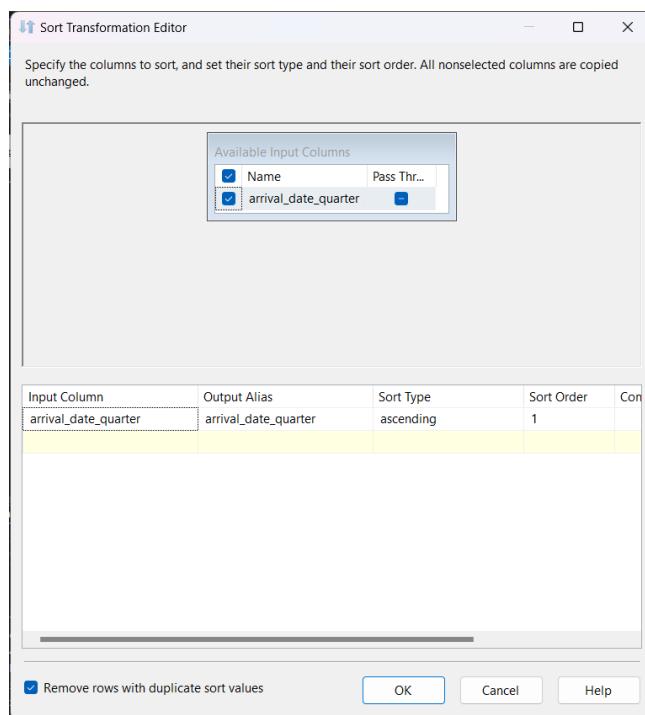
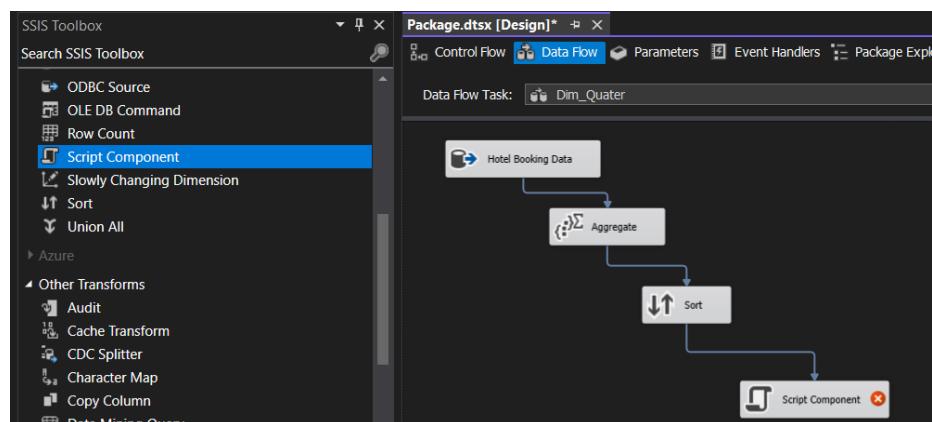


Figure 7-8. Sắp xếp các giá trị theo chiều tăng dần

- **Bước 7:** Kéo thả công cụ Script Component để tạo khóa chính id_quarter.



Common Properties	
ComparisonType	
Description	
ErrorOrTruncate	
ErrorRowDisposition	
ID	193
IdentificationString	Script Component
LineageID	193
LineageIdentifier	Script Component
MappedColumn	0
Name	id_quarter
SpecialFlags	0
TruncationRowDisposition	NotUsed

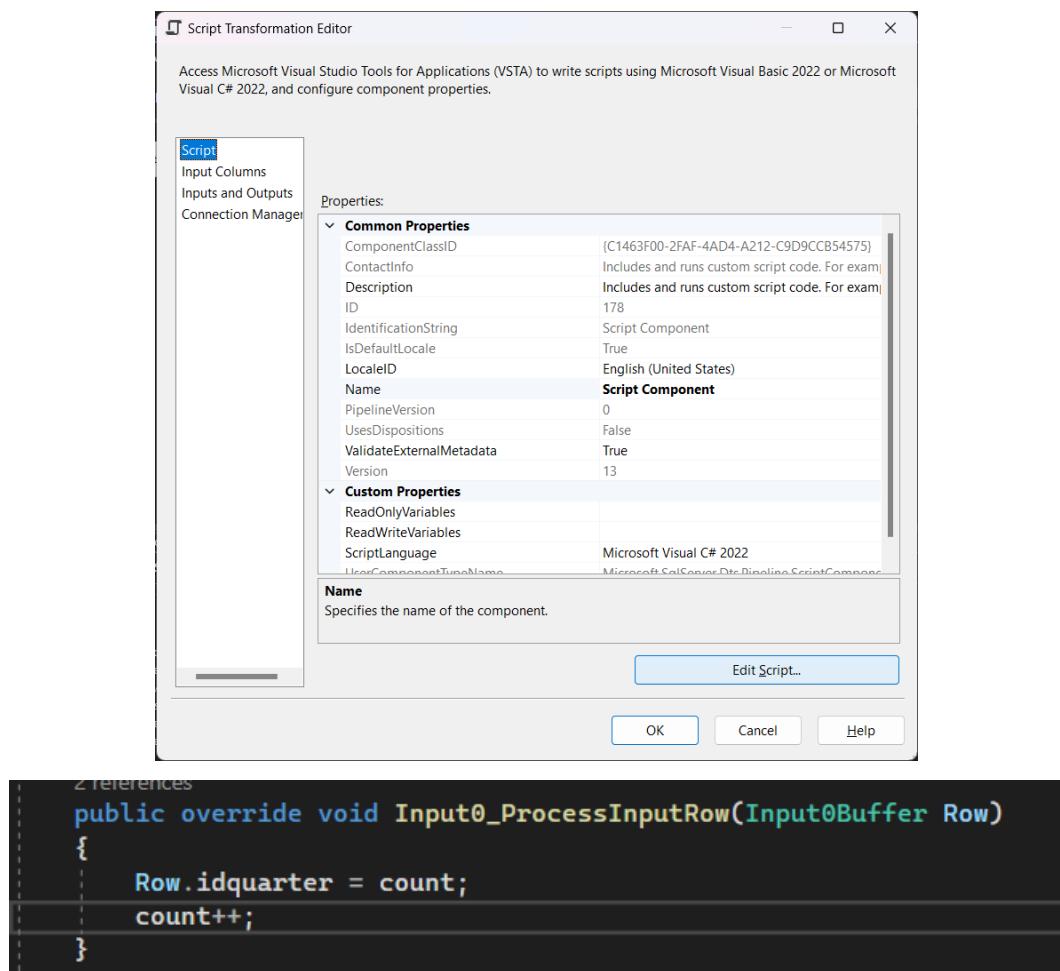


Figure 9-10-11-12-13. Sử dụng Script Component để tạo khóa chính id_quarter

- **Bước 8:** Chọn công cụ OLE DB Destination để tiến hành tạo bảng trong cơ sở dữ liệu DB_HotelBooking với tên gọi là Dim_Quarter.

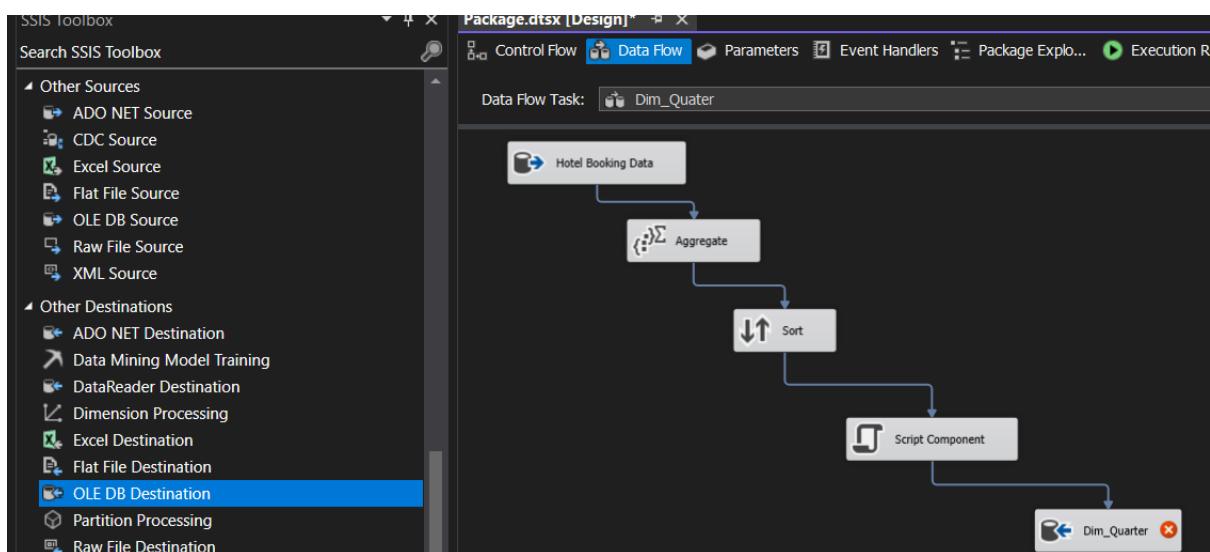


Figure 134. Sử dụng OLE DB Destination để tạo bảng

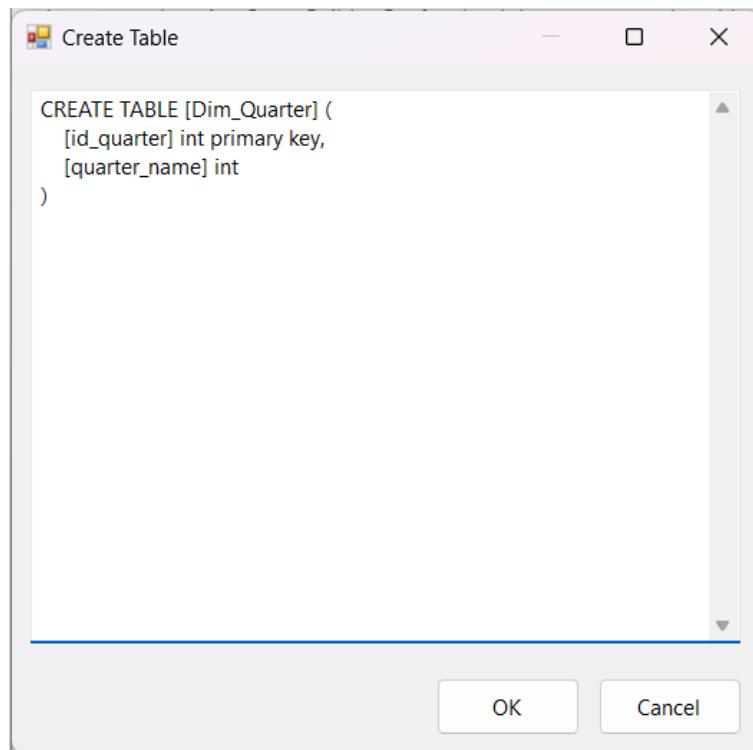


Figure 135. Tạo bảng Dim_Quater

- **Bước 9:** Nhấn Mappings để kiểm tra kết nối và nhấn Ok để hoàn tất quá trình tạo bảng.

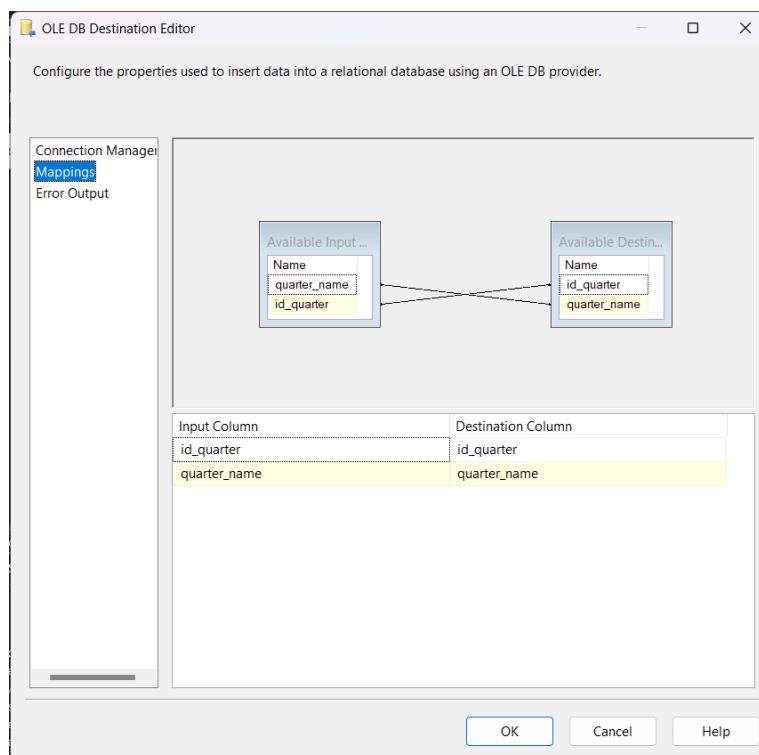


Figure 136. Quá trình Mappings dữ liệu

- **Bước 10:** Chọn Start để bắt đầu quá trình đổ dữ liệu. Khi đổ thành công, ta có kết quả như sau:

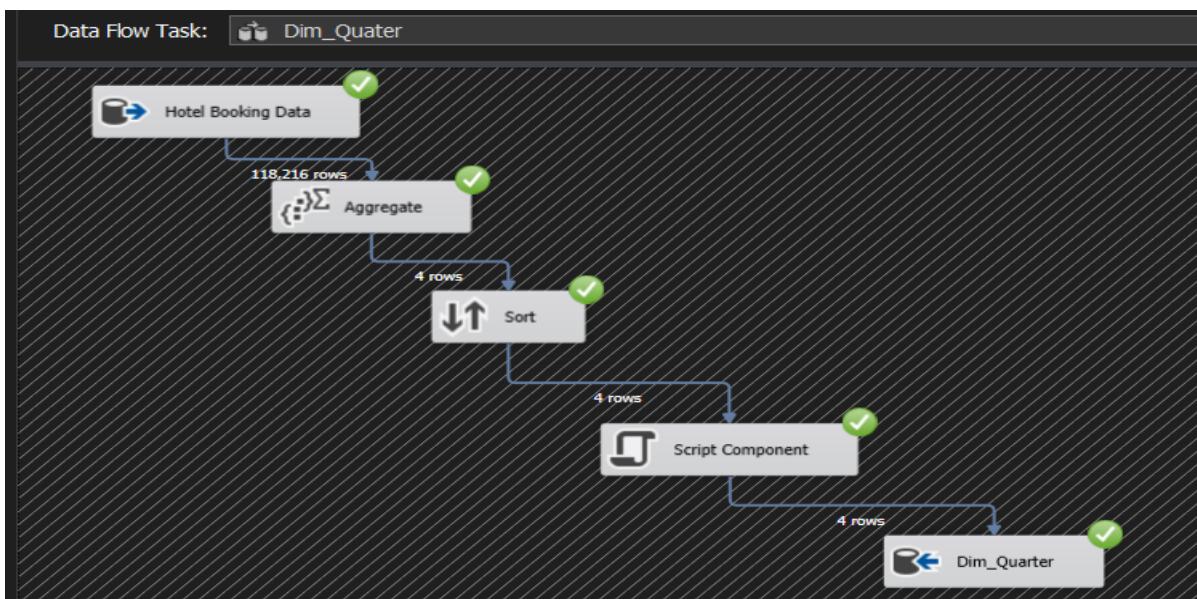


Figure 137. Hoàn thành đổ dữ liệu vào Dim_Quarter trong kho dữ liệu

- **Bước 11:** Kiểm tra bảng Dim_Quarter trên SQL Server.

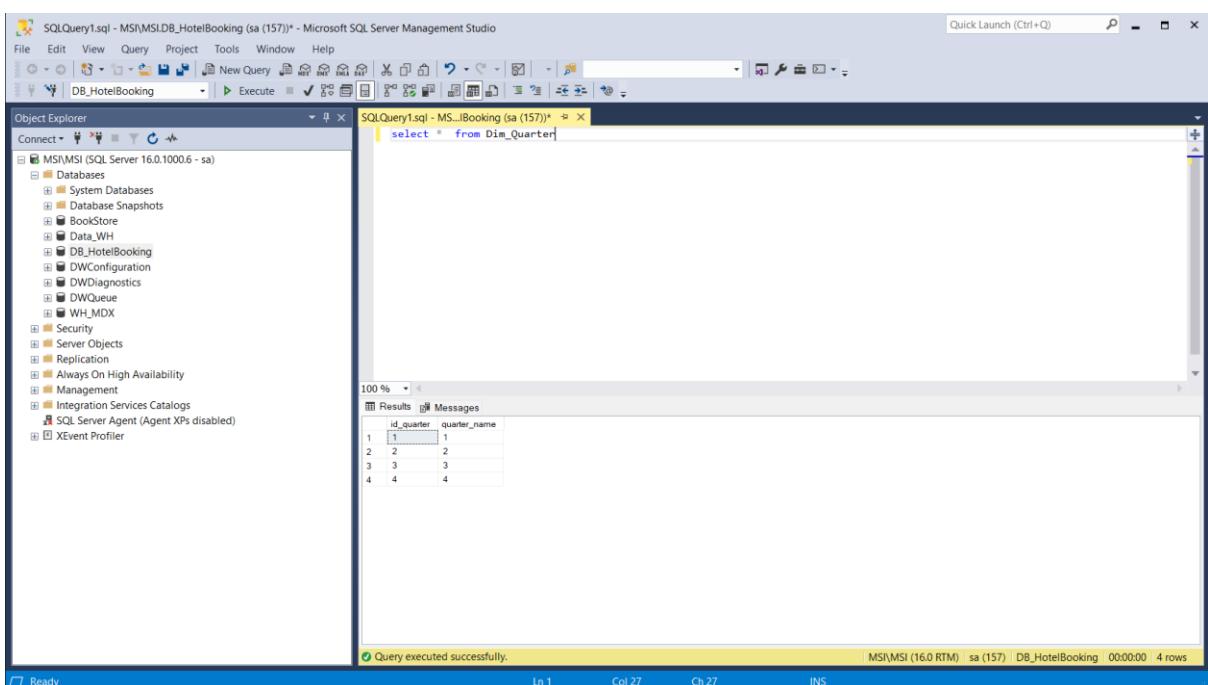


Figure 138. Kiểm tra bảng Dim_Quarter trong SQL Server

2.5.11. Bảng Dim_Month

- **Bước 1:** Kéo chức năng Data Flow Task từ cột trái sang màn hình làm việc và đổi tên thành Dim_Month.

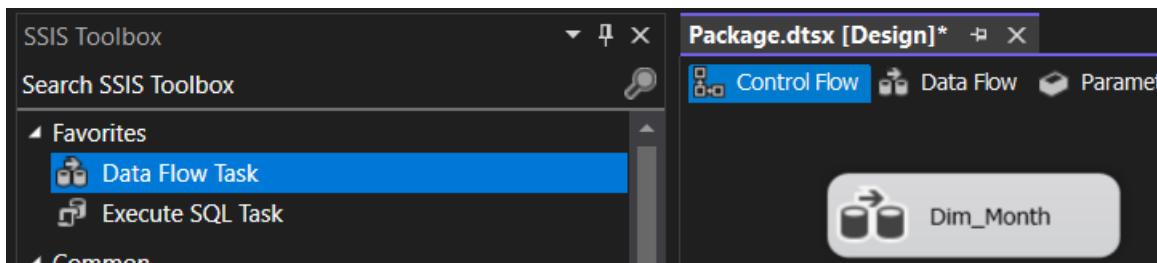


Figure 139. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Month

- **Bước 2:** Nhấn double vào Data Flow Task vừa tạo và tìm kiếm chức năng OLE DB Source, kéo vào màn hình làm việc.

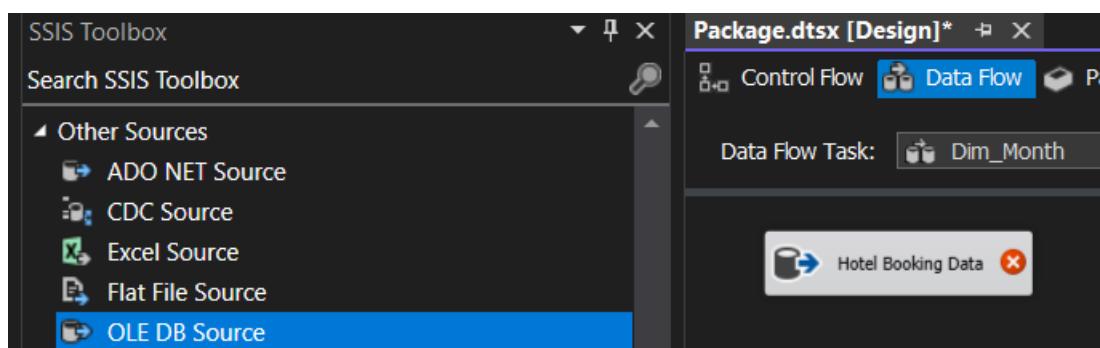


Figure 140. Kéo chức năng OLE DB Source vào màn hình làm việc

- **Bước 3:** Click double vào OLE DB Source và dẫn đến DB_HotelBooking. Sau đó chọn Name of the table or the view là [dbo].Hotel_Booking.

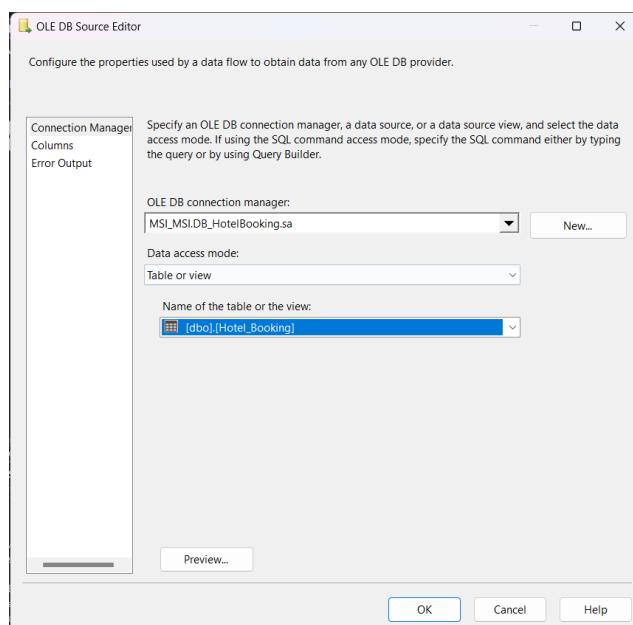


Figure 141. Chọn Table sẽ lấy dữ liệu

- **Bước 4:** Chọn Columns để lựa chọn các cột cần sử dụng và nhấn OK.

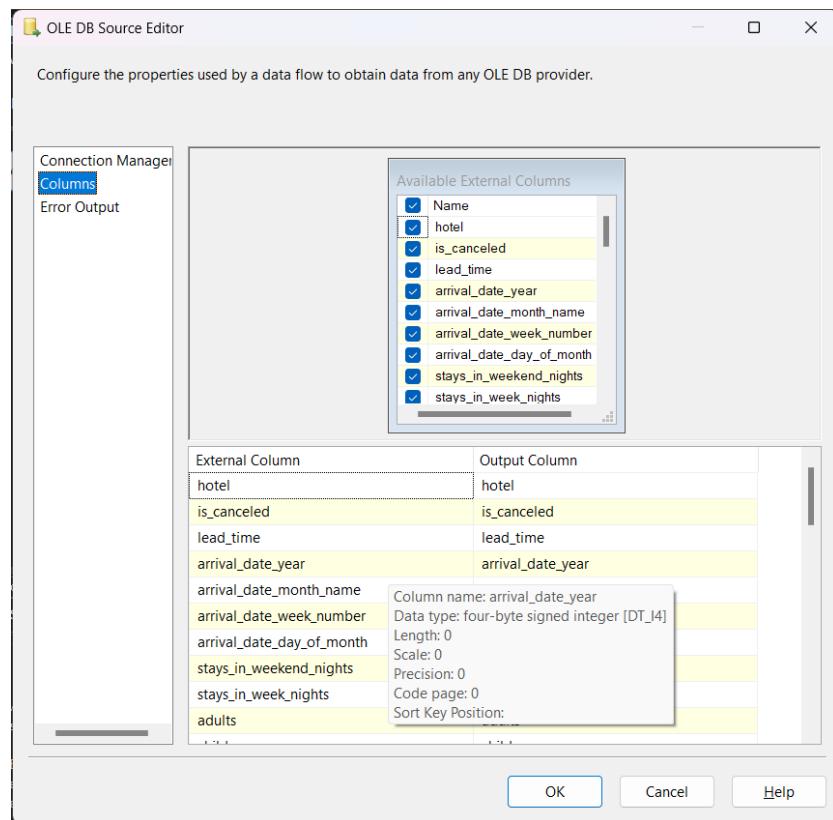


Figure 142. Chọn các column cần sử dụng

- **Bước 5:** Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau lên thuộc tính sử dụng là arrival_date_month_number, arrival_date_month_name, arrival_date_month_name_abbrev.

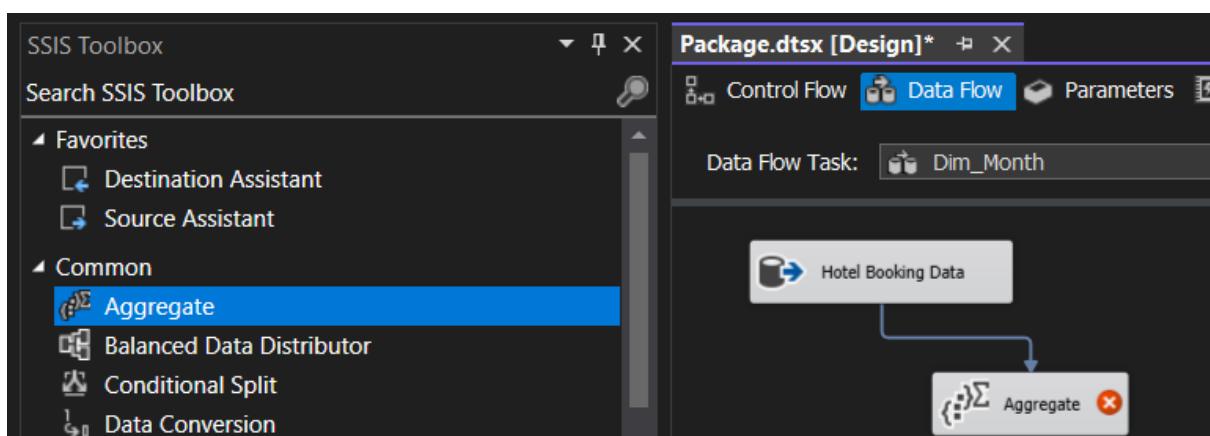


Figure 143. Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau

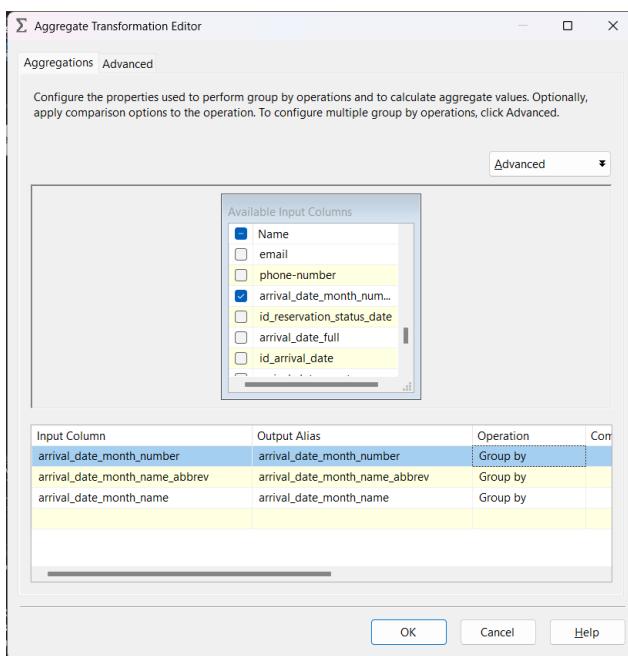


Figure 144. Chọn các thuộc tính cần gom nhóm

- **Bước 6:** Dùng công cụ Sort để sắp xếp các giá trị theo chiều tăng dần

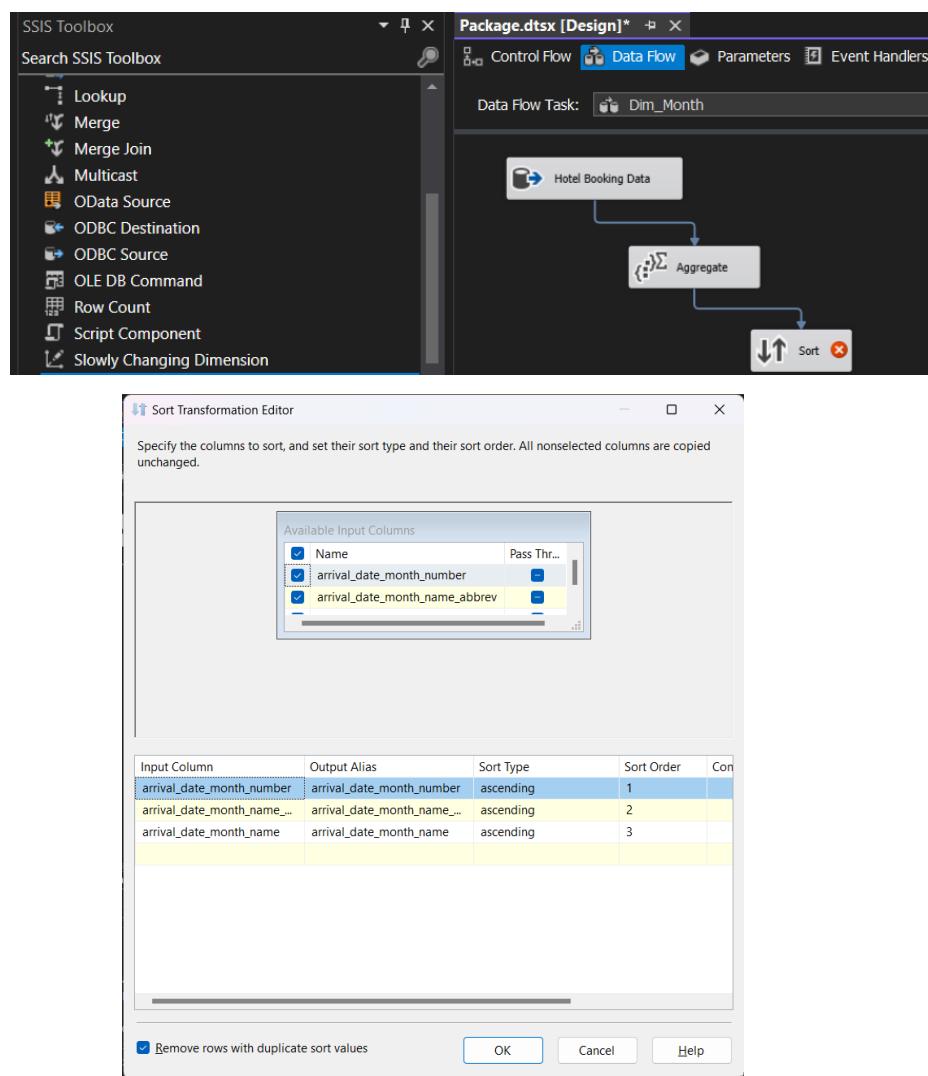


Figure 7-8. Sắp xếp các giá trị theo chiều tăng dần

- **Bước 7:** Chọn công cụ OLE DB Destination để tiến hành tạo bảng trong cơ sở dữ liệu DB_HotelBooking với tên gọi là Dim_Month.

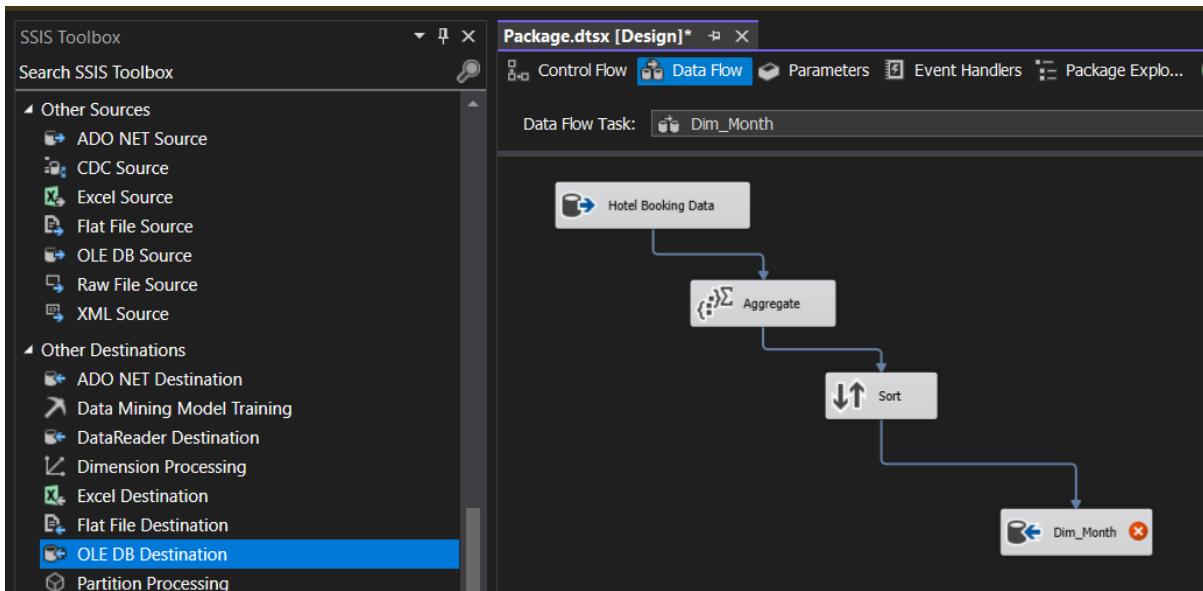


Figure 145. Sử dụng OLE DB Destination để tạo bảng

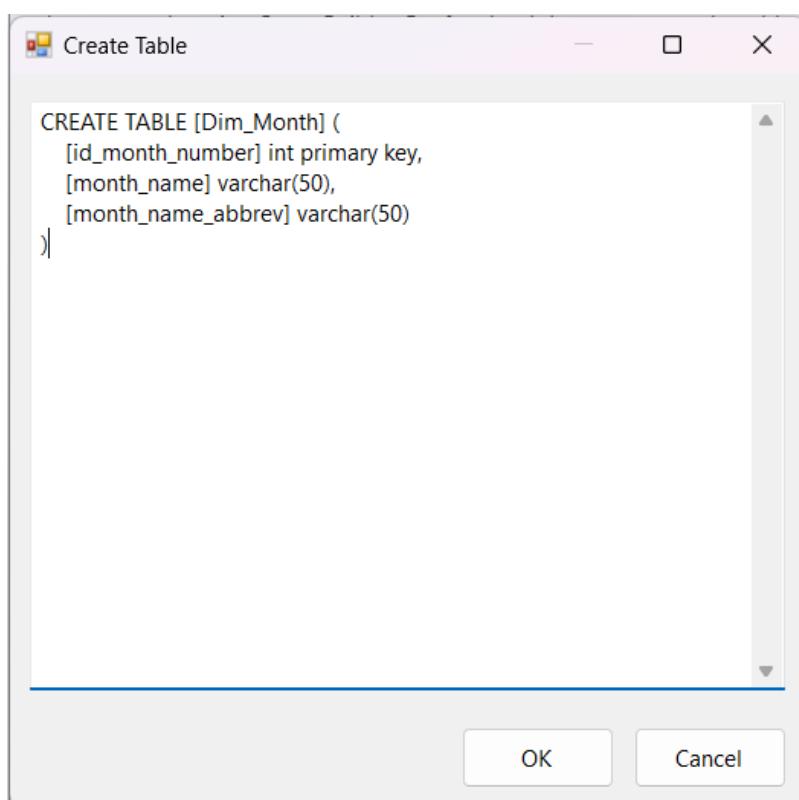


Figure 146. Tạo bảng Dim_Month

- **Bước 8:** Nhấn Mappings để kiểm tra kết nối và nhấn Ok để hoàn tất quá trình tạo bảng.

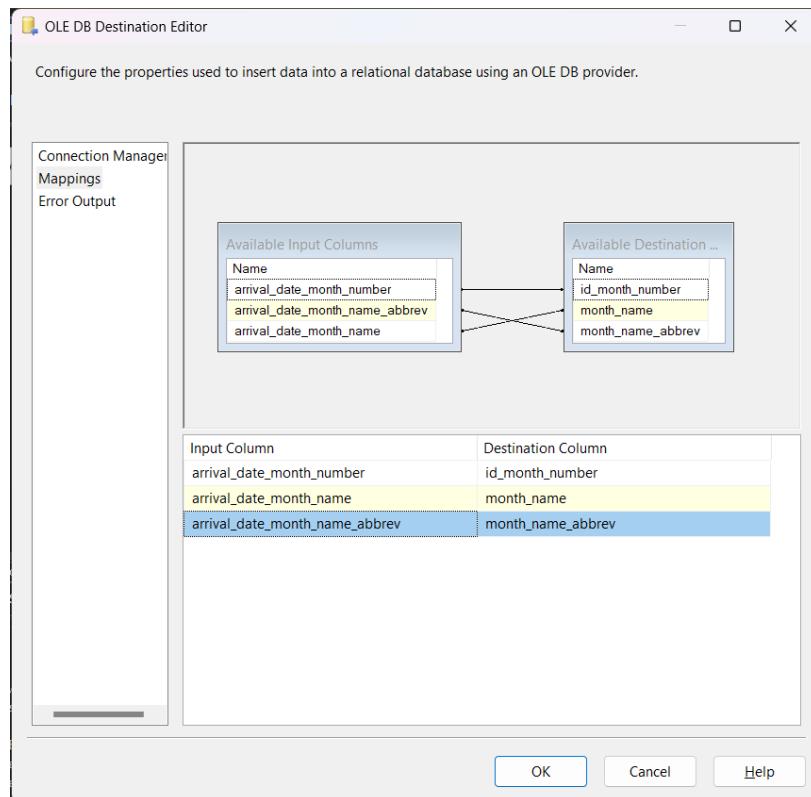


Figure 147. Quá trình Mappings dữ liệu

- **Bước 9:** Chọn Start để bắt đầu quá trình đổ dữ liệu. Khi đổ thành công, ta có kết quả như sau:

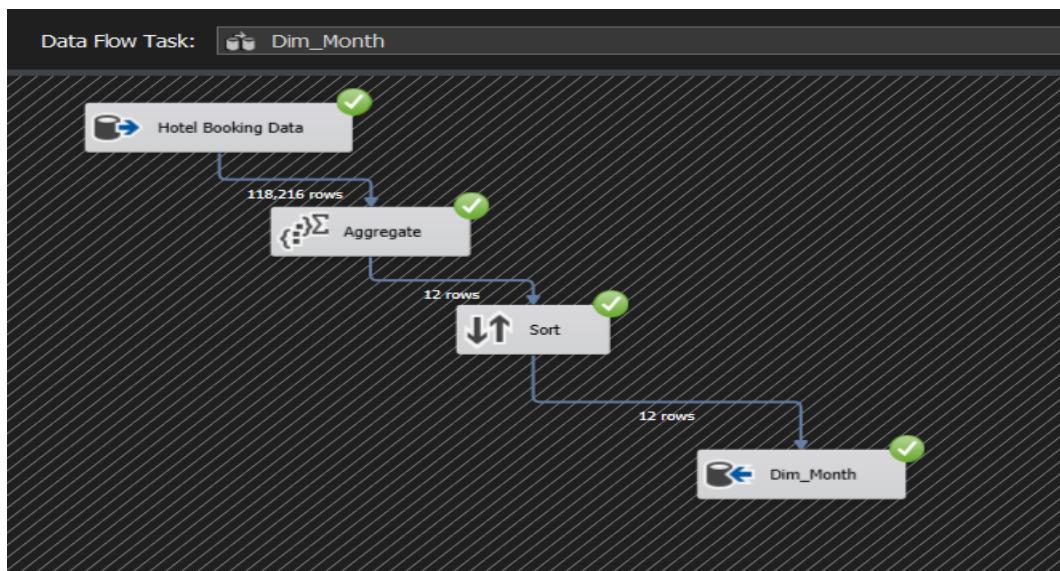


Figure 148. Hoàn thành đổ dữ liệu vào Dim_Month trong kho dữ liệu

- **Bước 10:** Kiểm tra bảng Dim_Month trên SQL Server.

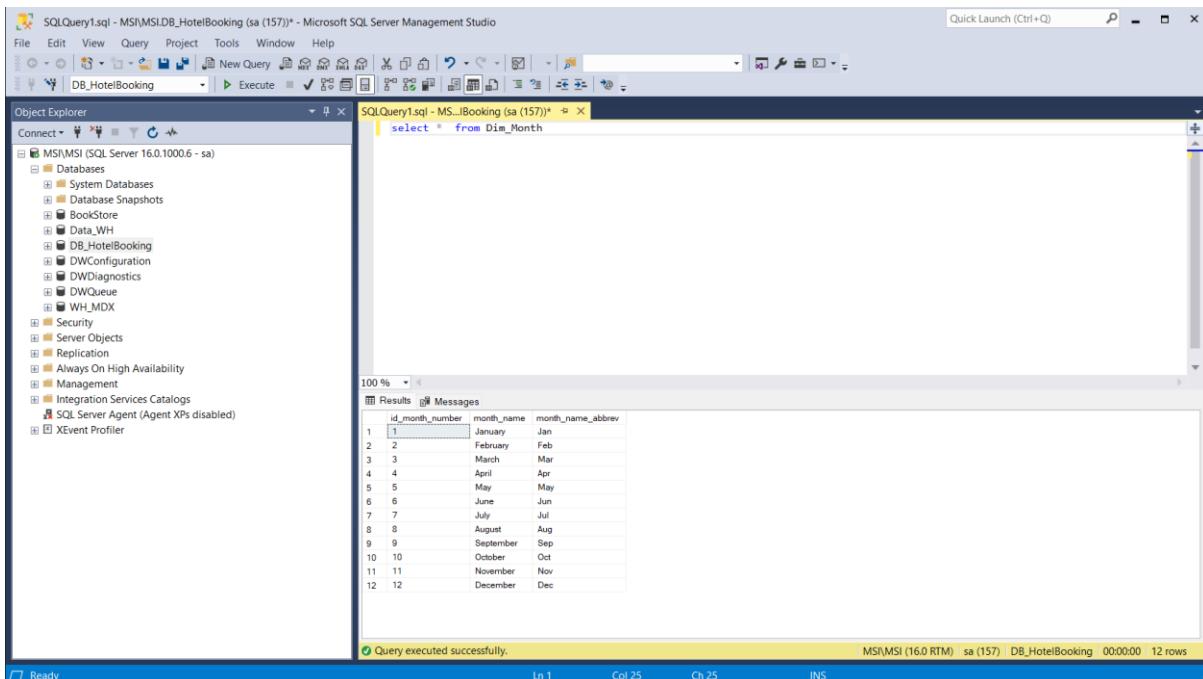


Figure 149. Kiểm tra bảng Dim_Month trong SQL Server

2.5.12. Bảng Dim_Day

- **Bước 1:** Kéo chức năng Data Flow Task từ cột trái sang màn hình làm việc và đổi tên thành Dim_Day.

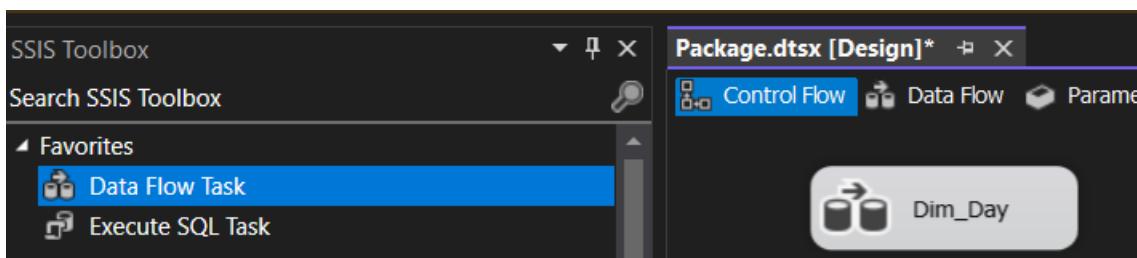


Figure 150. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Day

- **Bước 2:** Nhấn double vào Data Flow Task vừa tạo và tìm kiếm chức năng OLE DB Source, kéo vào màn hình làm việc.

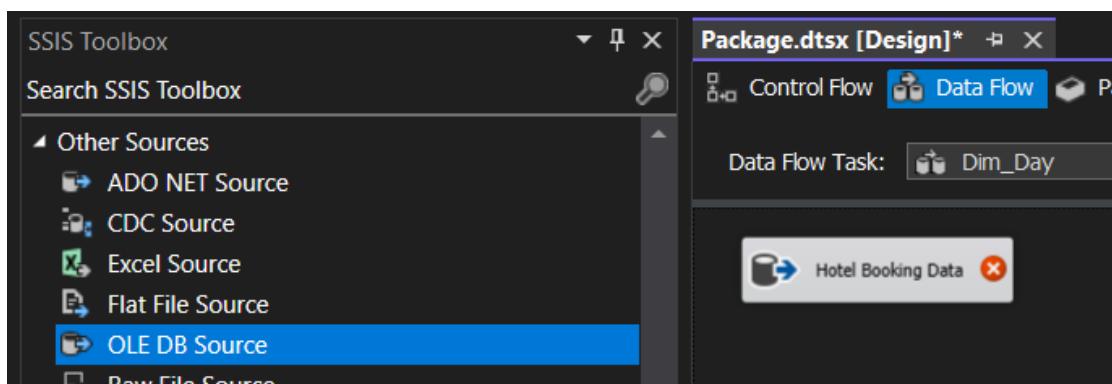


Figure 151. Kéo chức năng OLE DB Source vào màn hình làm việc

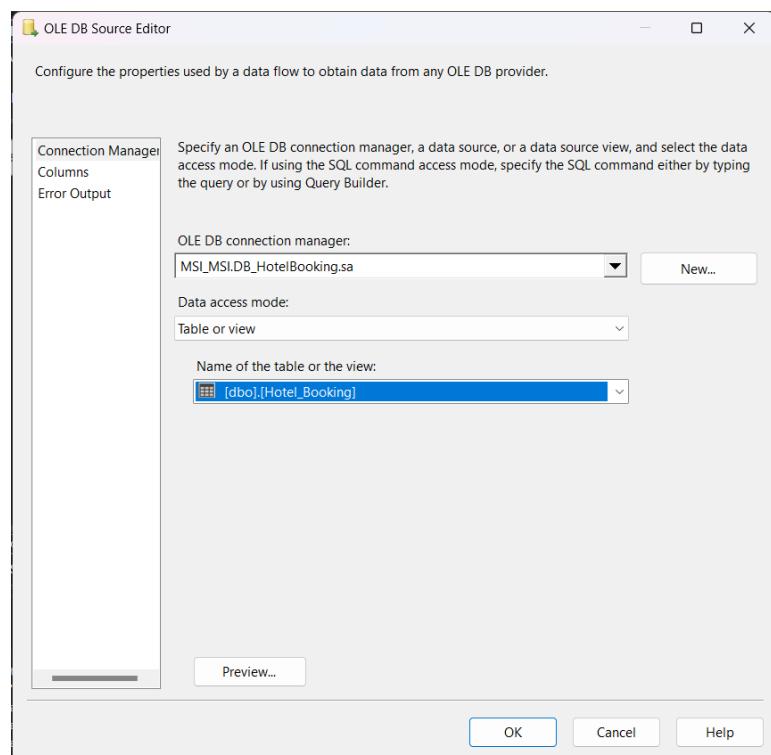


Figure 152. Chọn Table sẽ lấy dữ liệu

- **Bước 3:** Chọn Columns để lựa chọn các cột cần sử dụng và nhấn OK

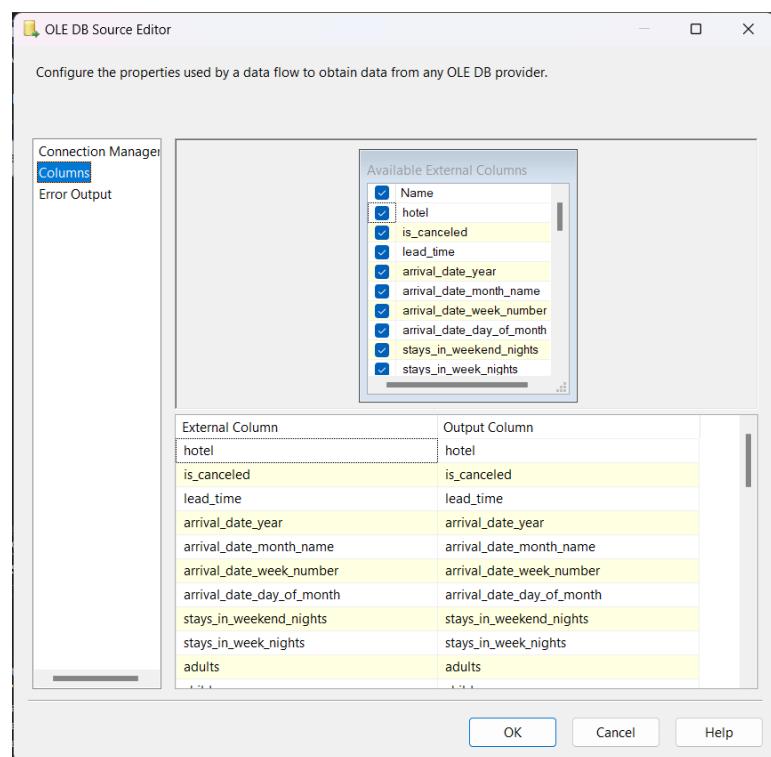


Figure 153. Chọn các column cần sử dụng

- **Bước 4:** Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau lên thuộc tính sử dụng là arrival_date_day_of_month.

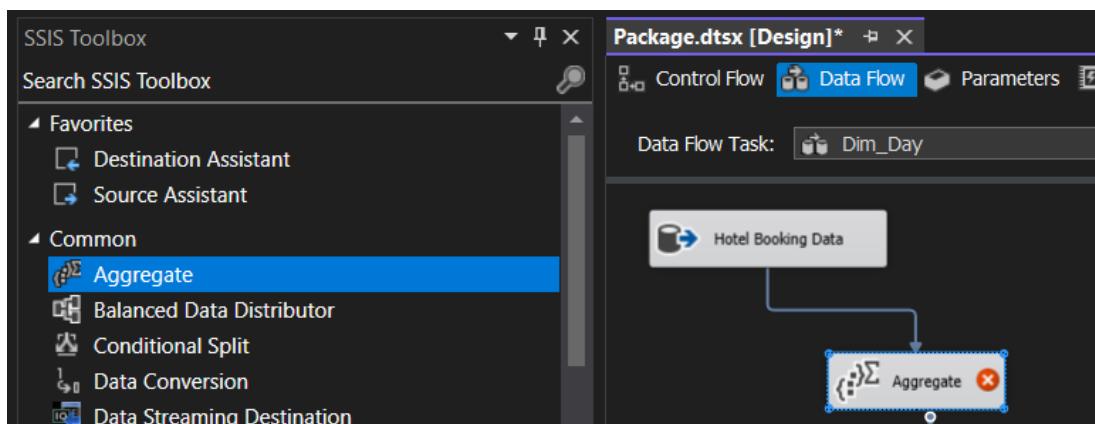


Figure 154. Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau

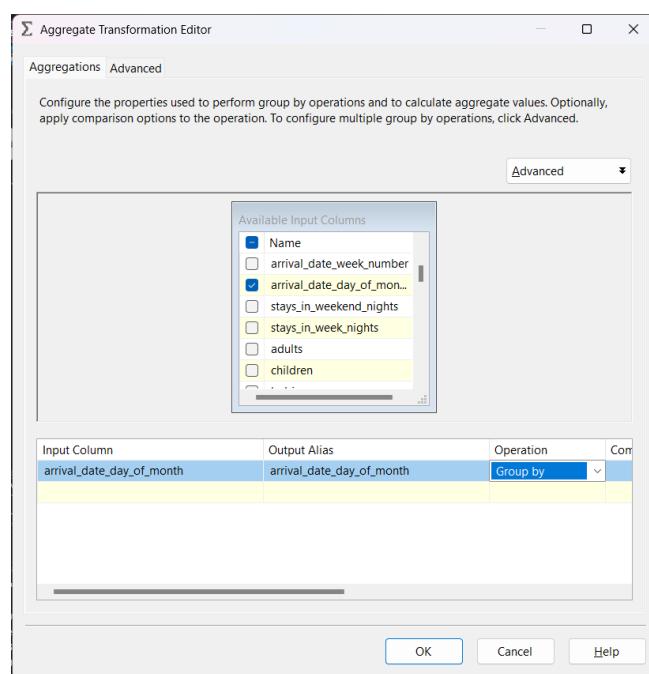
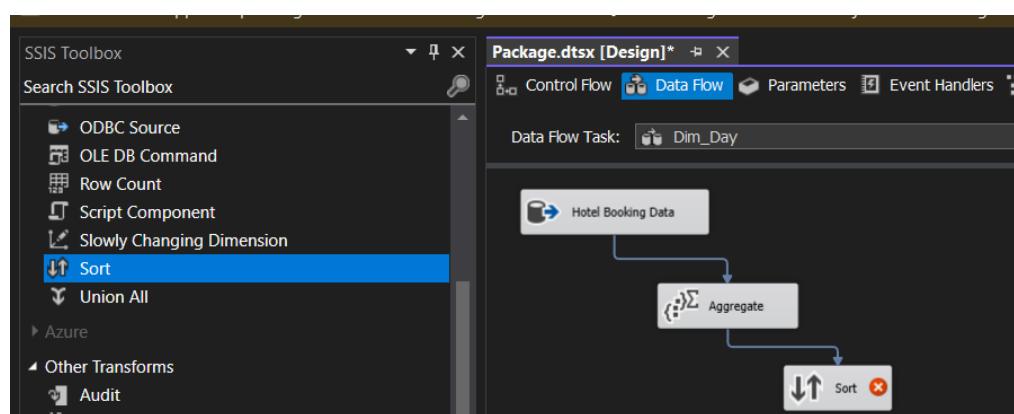


Figure 155. Chọn các thuộc tính cần gom nhóm

- **Bước 5:** Dùng công cụ Sort để sắp xếp các giá trị theo chiều tăng dần.



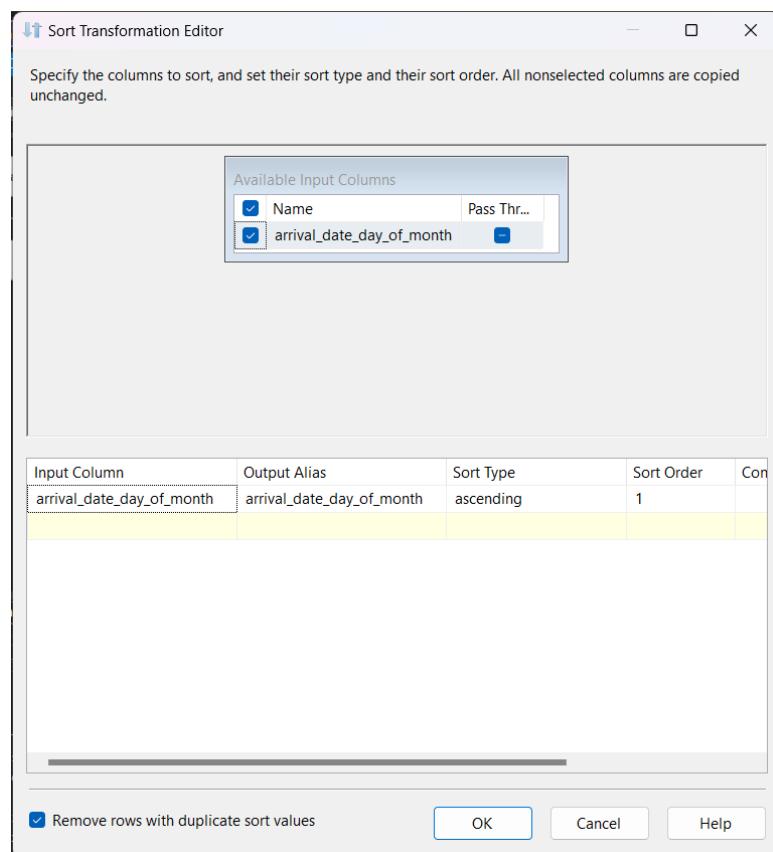
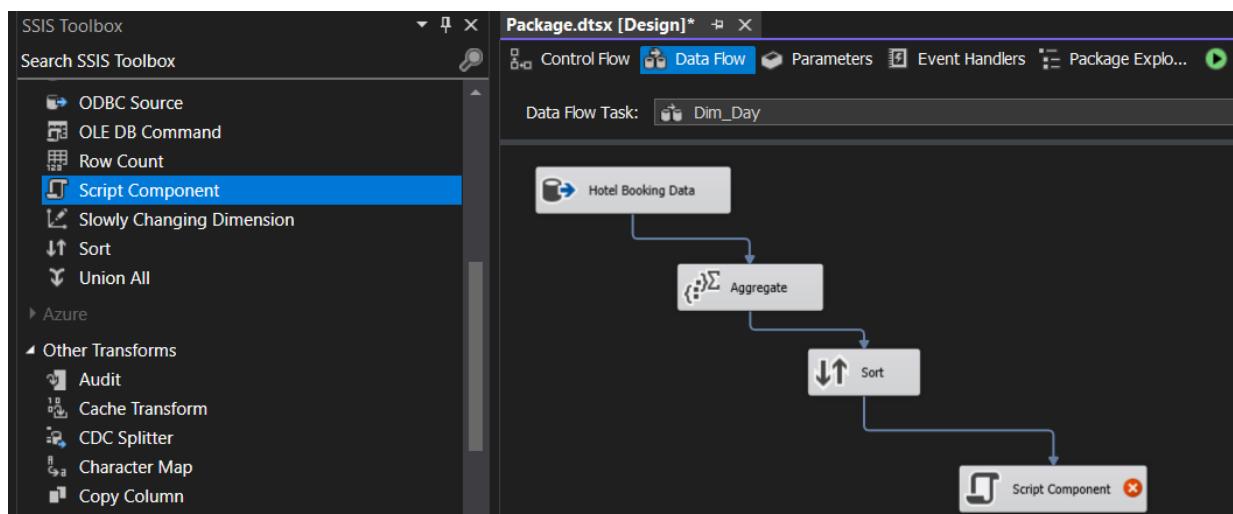
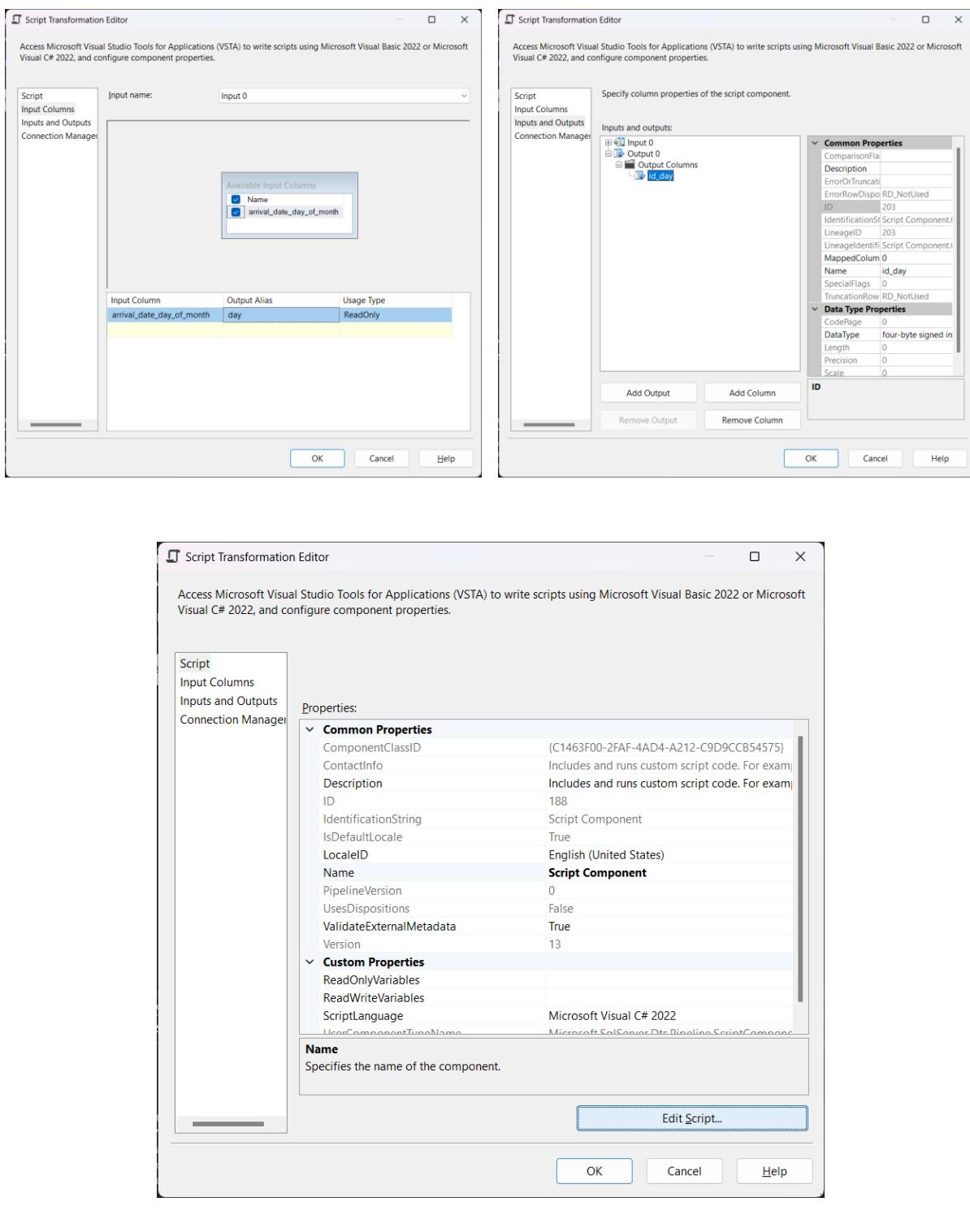


Figure 7-8. Sắp xếp các giá trị theo chiều tăng dần

- **Bước 6:** Kéo thả công cụ Script Component để tạo khóa chính id_day.



IS217.N21 – Kho dữ liệu và phân tích hoạt động đặt phòng khách sạn



```

    public override void Input0_ProcessInputRow(Input0Buffer Row)
{
    Row.idday = count;
    count++;
}
  
```

Figure 9-10-11-12-13. Sử dụng Script Component để tạo khóa chính id_day

- **Bước 7:** Chọn công cụ OLE DB Destination để tiến hành tạo bảng trong cơ sở dữ liệu DB_HotelBooking với tên gọi là Dim_Day.

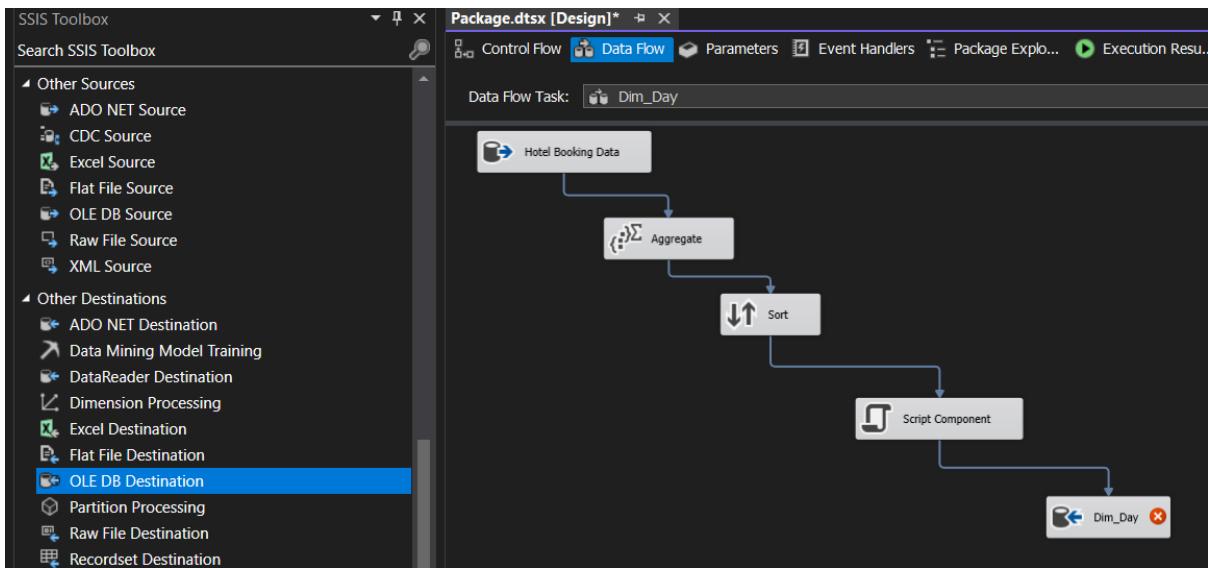


Figure 156. Sử dụng OLE DB Destination để tạo bảng

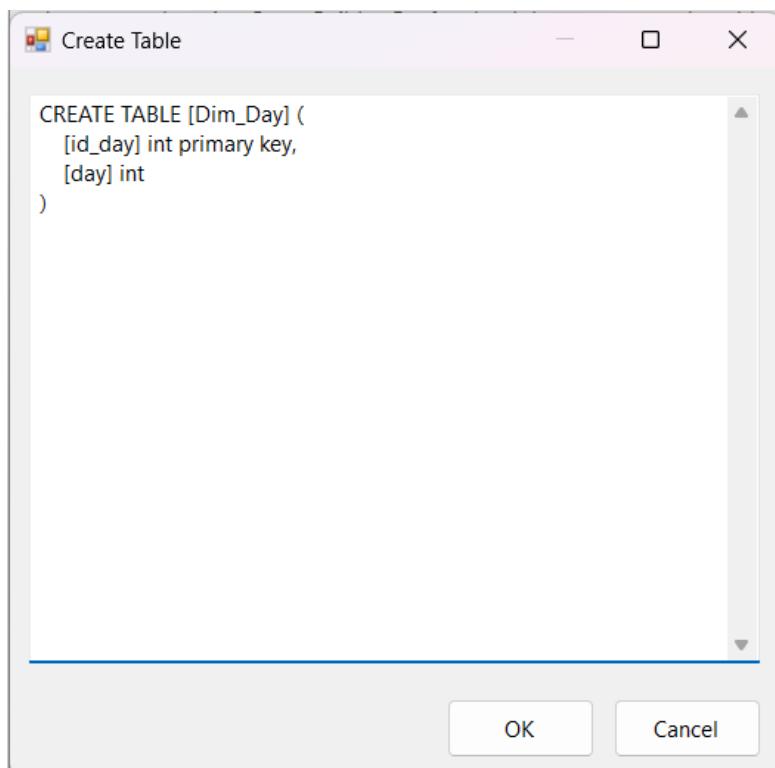


Figure 157. Tạo bảng Dim_Day

- **Bước 8:** Nhấn Mappings để kiểm tra kết nối và nhấn Ok để hoàn tất quá trình tạo bảng.

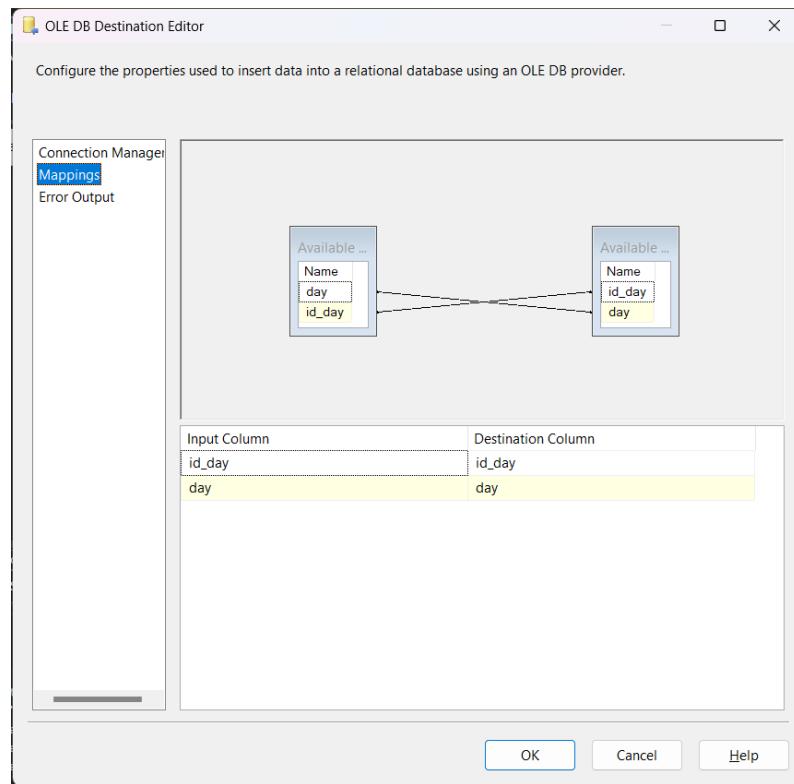


Figure 158. Quá trình Mappings dữ liệu

- **Bước 9:** Chọn Start để bắt đầu quá trình đổ dữ liệu. Khi đổ thành công, ta có kết quả như sau:

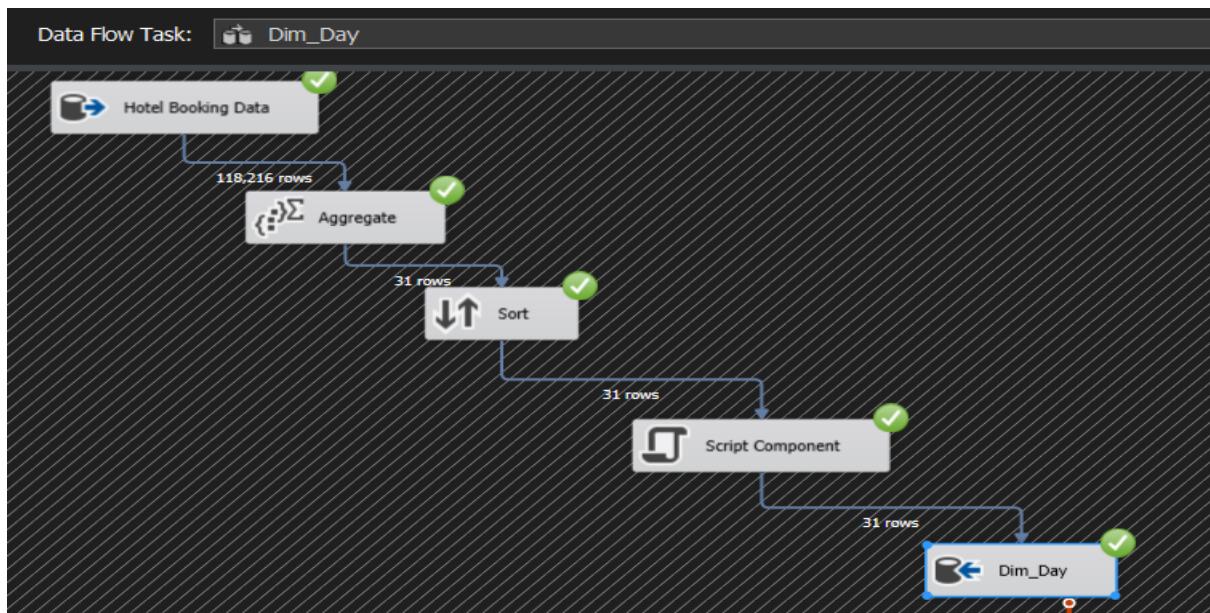


Figure 159. Hoàn thành đổ dữ liệu vào Dim_Day trong kho dữ liệu

- **Bước 10:** Kiểm tra bảng Dim_Day trên SQL Server.

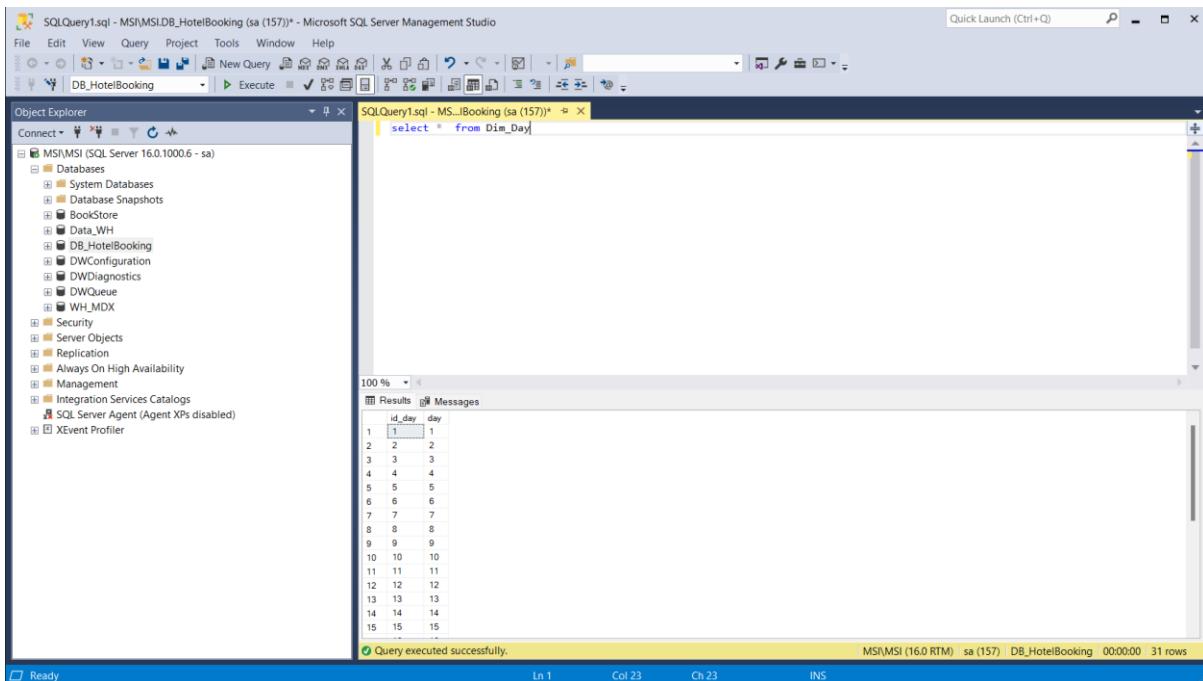


Figure 160. Kiểm tra bảng Dim_Day trong SQL Server

2.5.13. Bảng Dim_Arrival_Time

- **Bước 1:** Kéo chức năng Data Flow Task từ cột trái sang màn hình làm việc và đổi tên thành Dim_Arrival_Time và tạo đường liên kết từ Dim_Year, Dim_Quarter, Dim_Month và Dim_Day đến Dim_Arrival_Time.

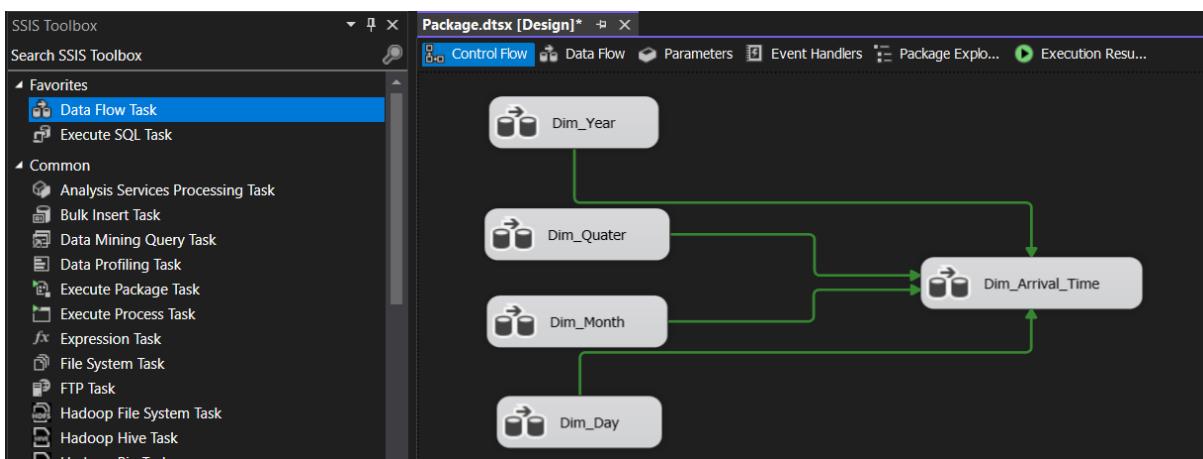


Figure 161. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Arrival_Time

- **Bước 2:** Nhấn double vào Data Flow Task vừa tạo và tìm kiếm chức năng OLE DB Source, kéo vào màn hình làm việc

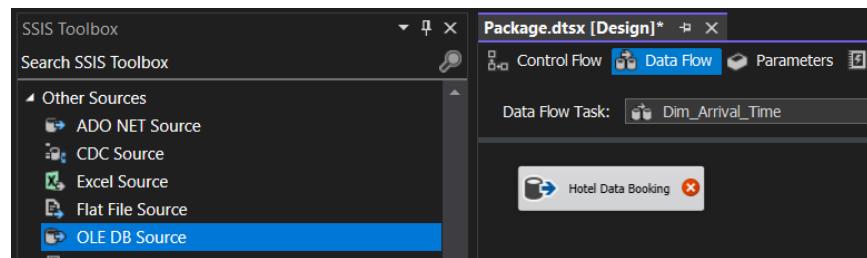


Figure 162. Kéo chức năng OLE DB Source vào màn hình làm việc

- **Bước 3:** Click double vào OLE DB Source và dẫn đến DB_HotelBooking. Sau đó chọn Name of the table or the view là [dbo].Hotel_Booking.

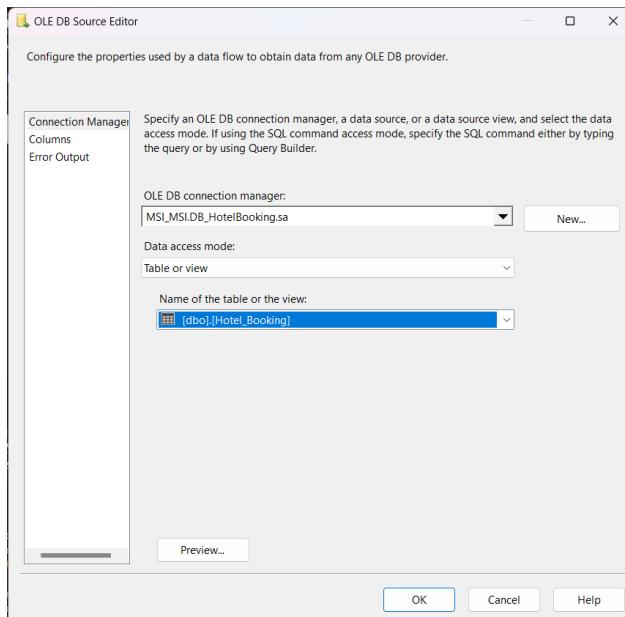


Figure 163. Chọn Table sẽ lấy dữ liệu

- **Bước 4:** Chọn Columns để lựa chọn các cột cần sử dụng và nhấn OK.

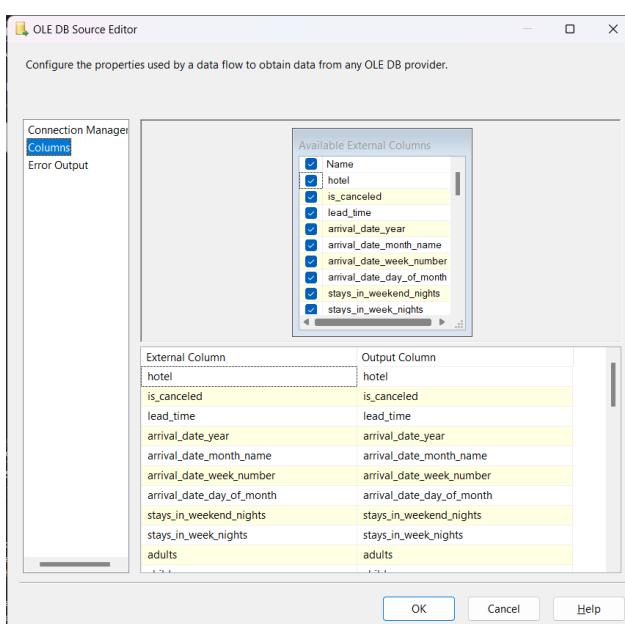


Figure 164. Chọn các column cần sử dụng

- **Bước 5:** Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau lên thuộc tính sử dụng là id_arrival_date, arrival_date_full, arrival_date_year, arrival_date_quarter, arrival_date_month_number, arrival_date_day_of_week, arrival_week_number, arrival_date_day_of_month, arrival_date_month_name, arrival_date_month_name_abbrev, arrival_date_day_name, arrival_date_day_name_abbrev, arrival_date_weekday_flag.

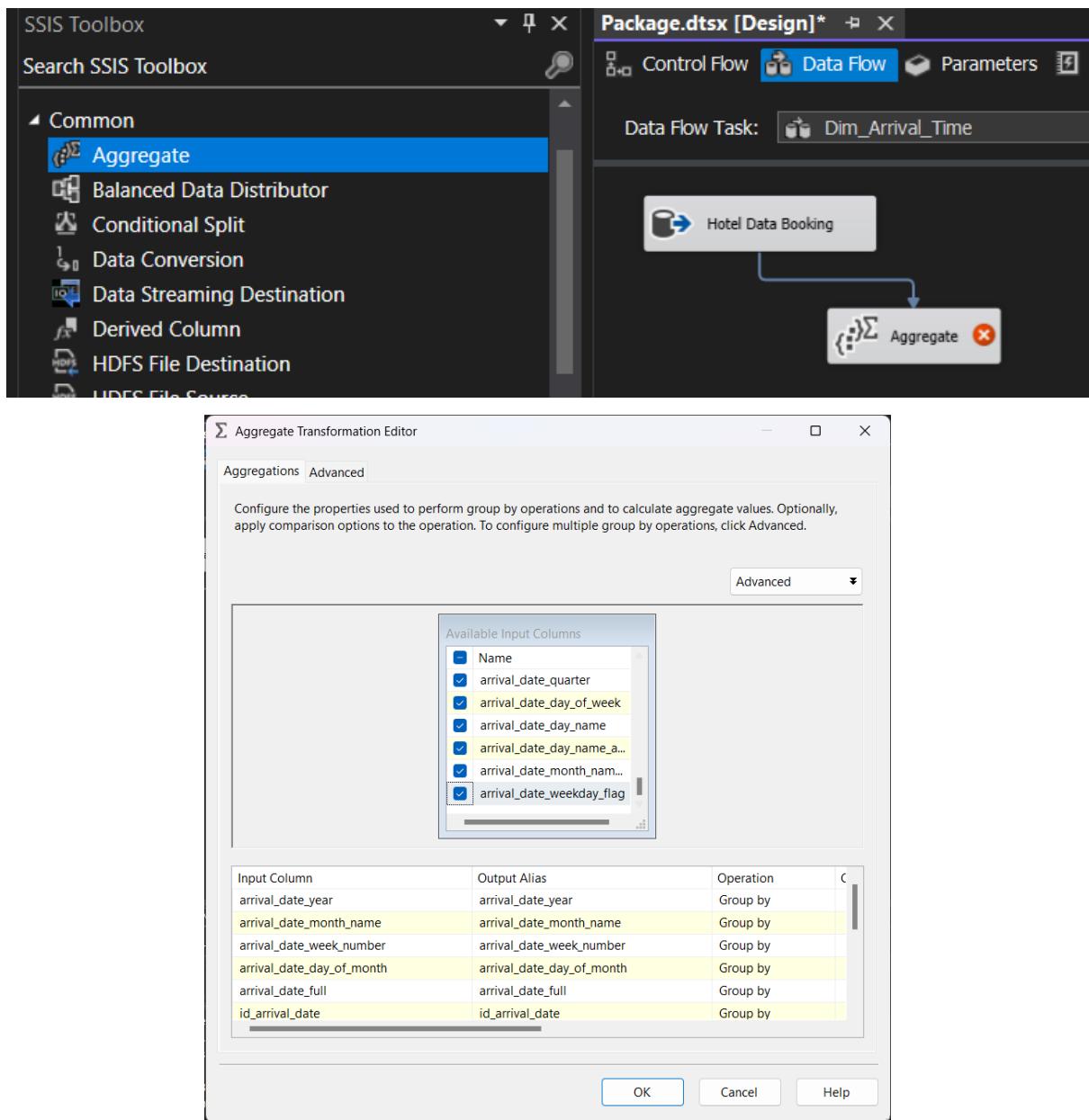


Figure 165. Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau

- **Bước 6:** Dùng công cụ Sort để sắp xếp các giá trị theo chiều tăng dần.

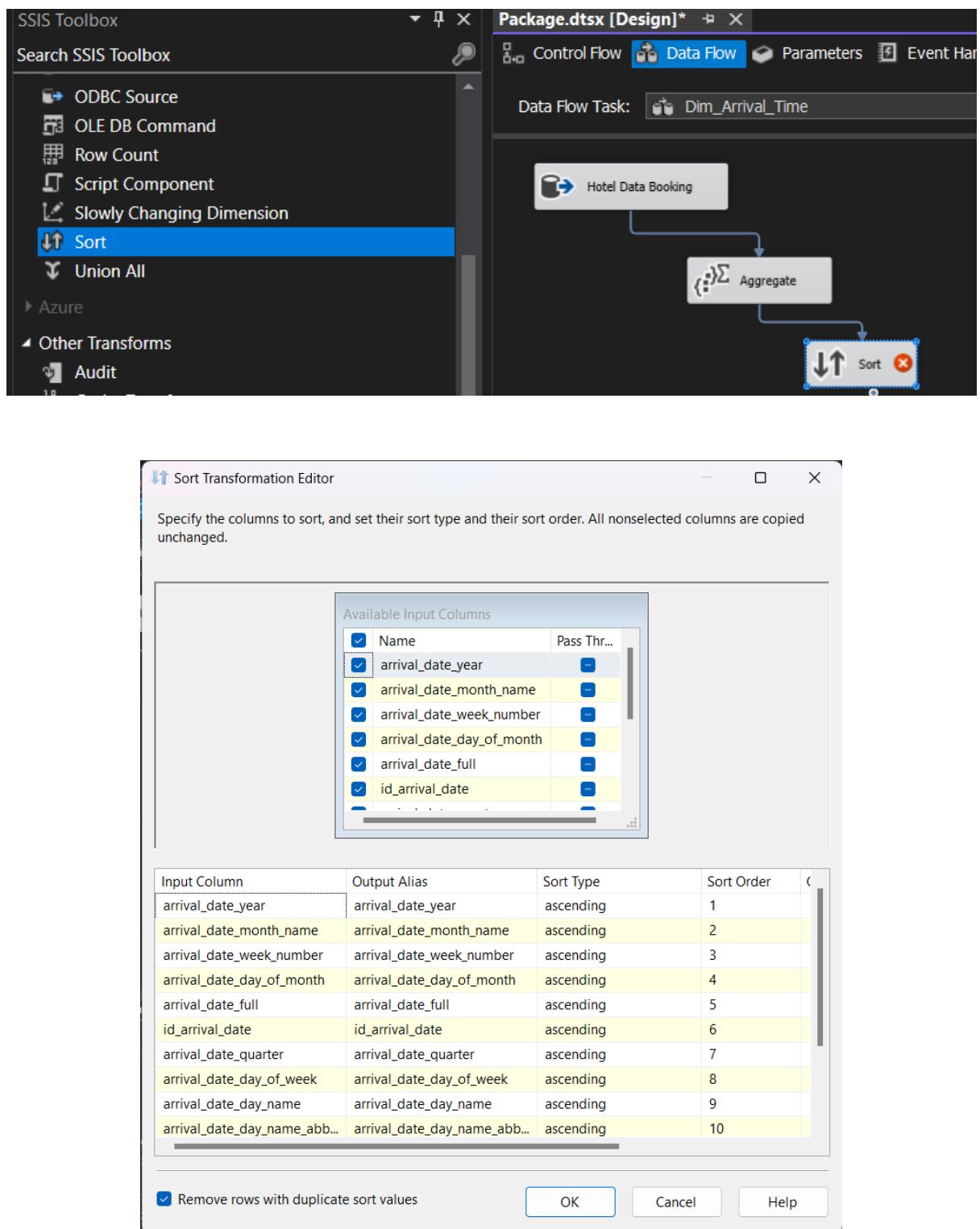


Figure 6-7. Sắp xếp các giá trị theo chiều tăng dần

- **Bước 7:** Tìm kiếm công cụ Lookup trên SSIS Toolbox và đổi tên thành Yaer_Lookup để tạo khóa ngoại id_arrival_year đến bảng Dim_Arrival_Time.

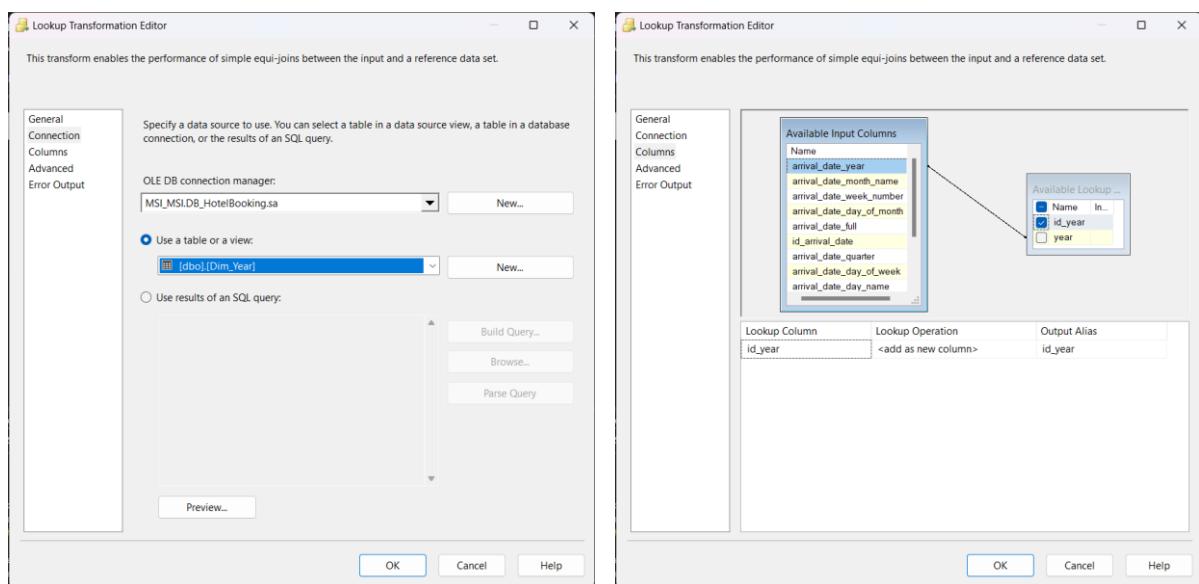
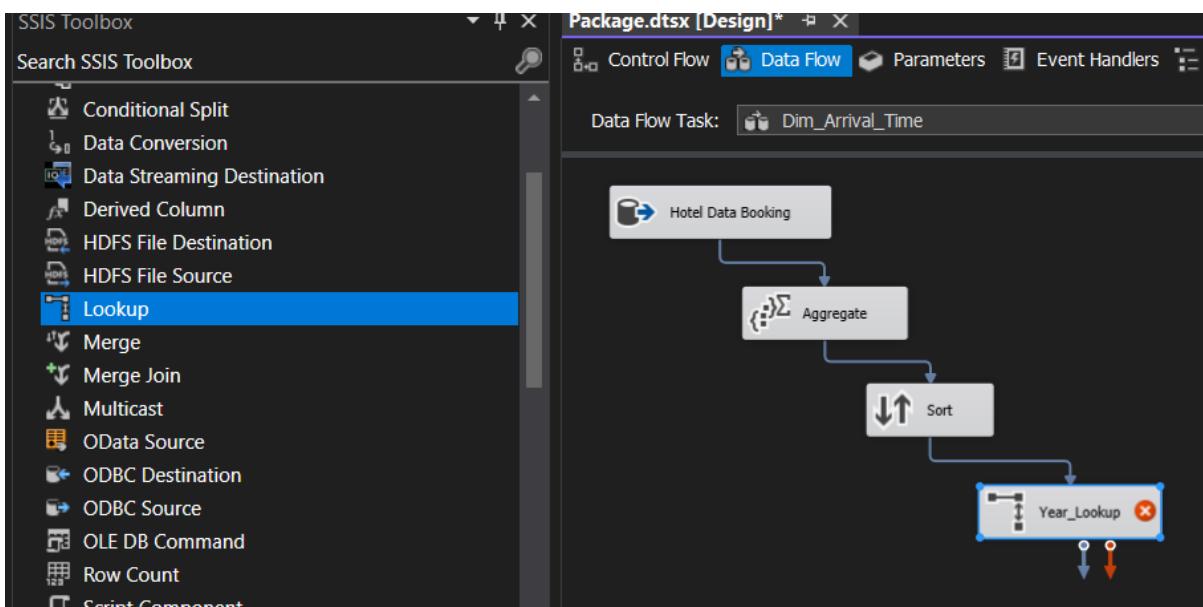


Figure 8-9-10. Tạo khóa ngoại id_arrival_year đến bảng Dim_Arrival_Time

- **Bước 8:** Tìm kiếm công cụ Lookup trên SSIS Toolbox và đổi tên thành Quarter_Lookup để tạo khóa ngoại id_arrival_quarter đến bảng Dim_Arrival_Time.

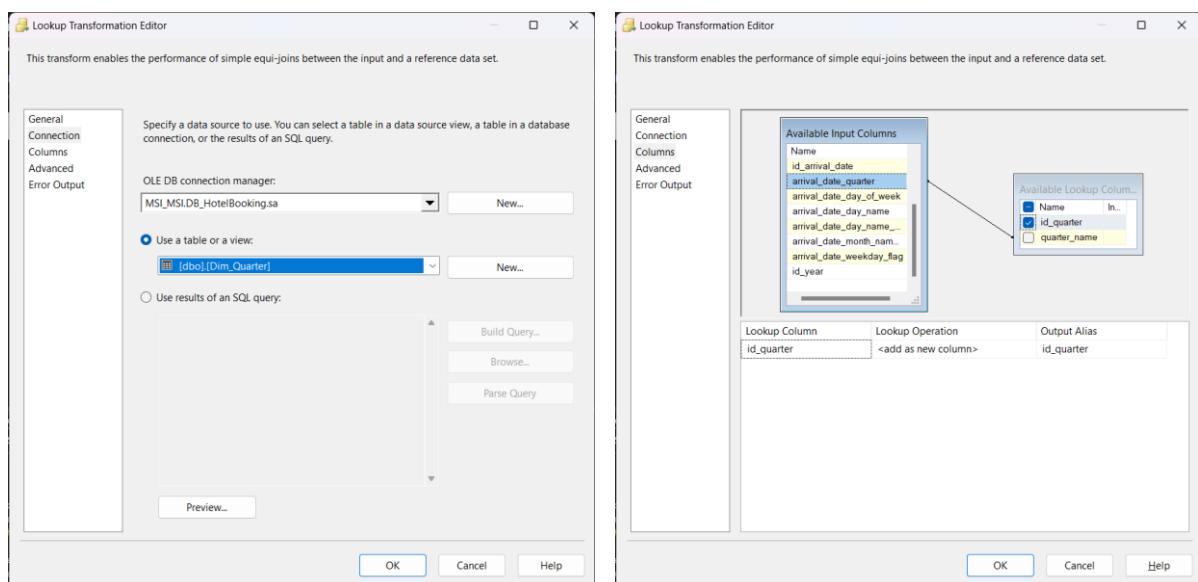
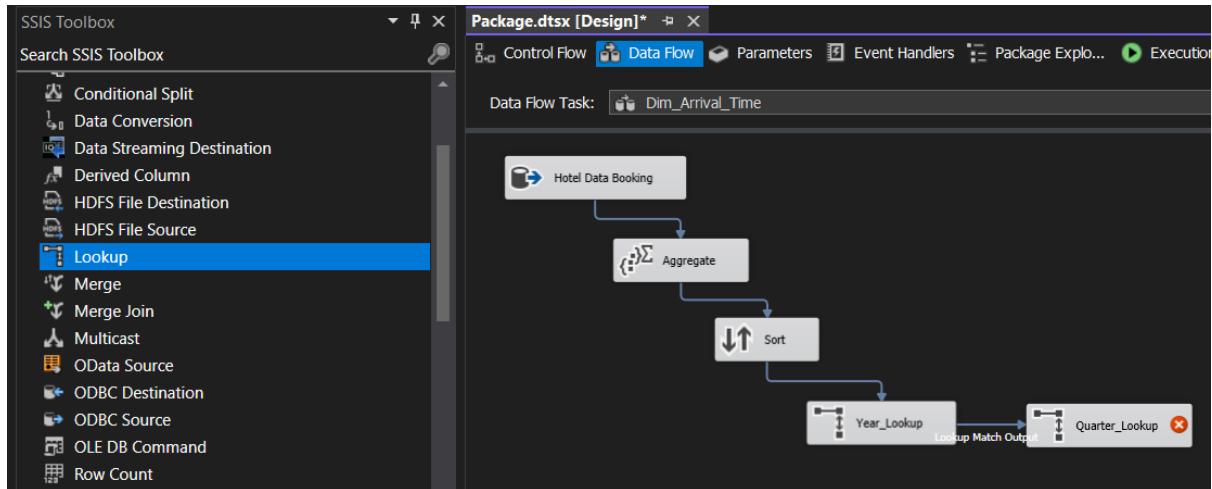


Figure 11-12-13. Tạo khóa ngoại id_arrival_quarter đến bảng Dim_Arrival_Time

- **Bước 9:** Tìm kiếm công cụ Lookup trên SSIS Toolbox và đổi tên thành Month_Lookup để tạo khóa ngoại id_arrival_month đến bảng Dim_Arrival_Time.

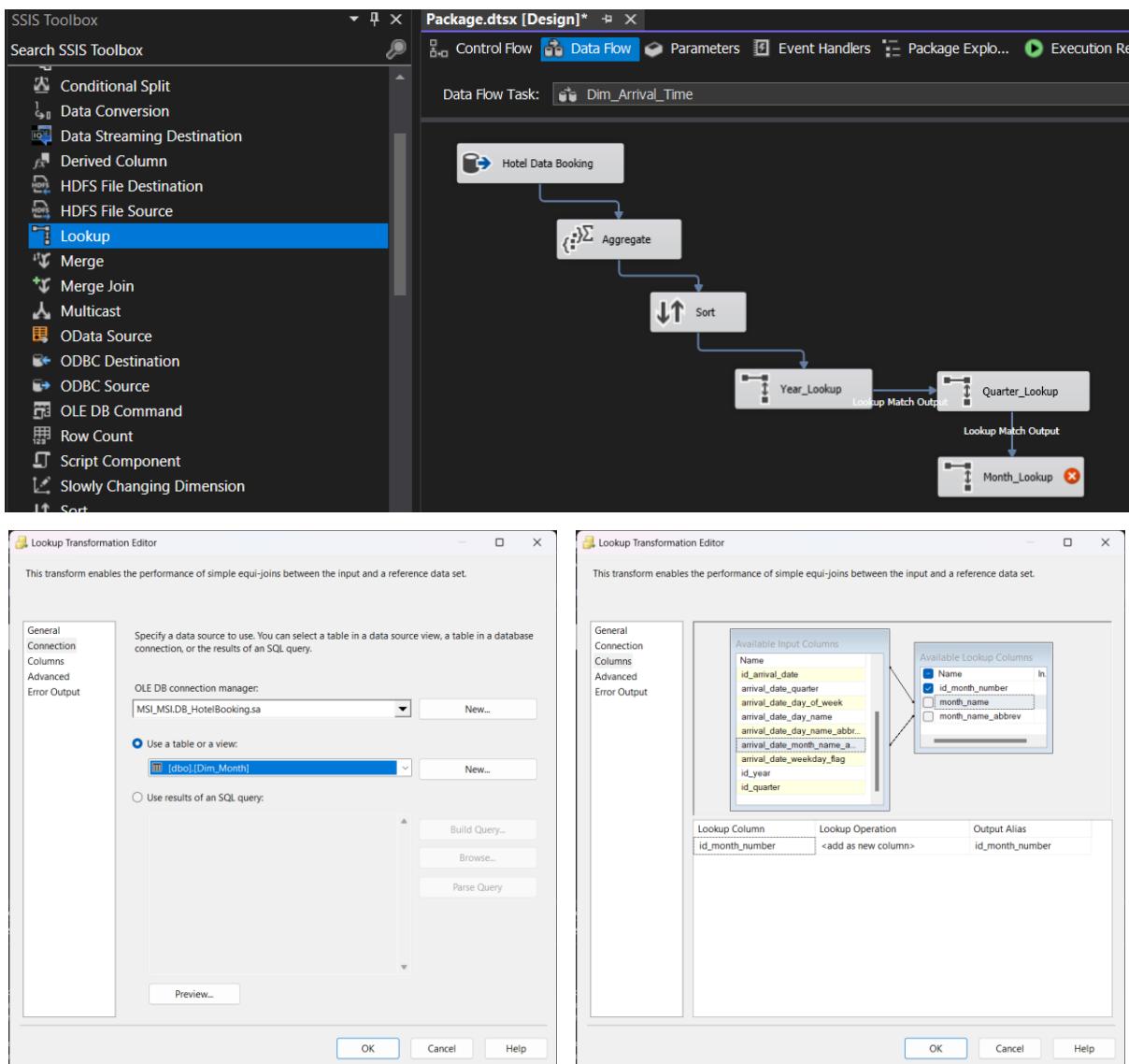


Figure 14-15-16. Tạo khóa ngoại id_arrival_month đến bảng Dim_Arrival_Time

- **Bước 10:** Tìm kiếm công cụ Lookup trên SSIS Toolbox và đổi tên thành Day_Lookup để tạo khóa ngoại id_arrival_day đến bảng Dim_Arrival_Time.

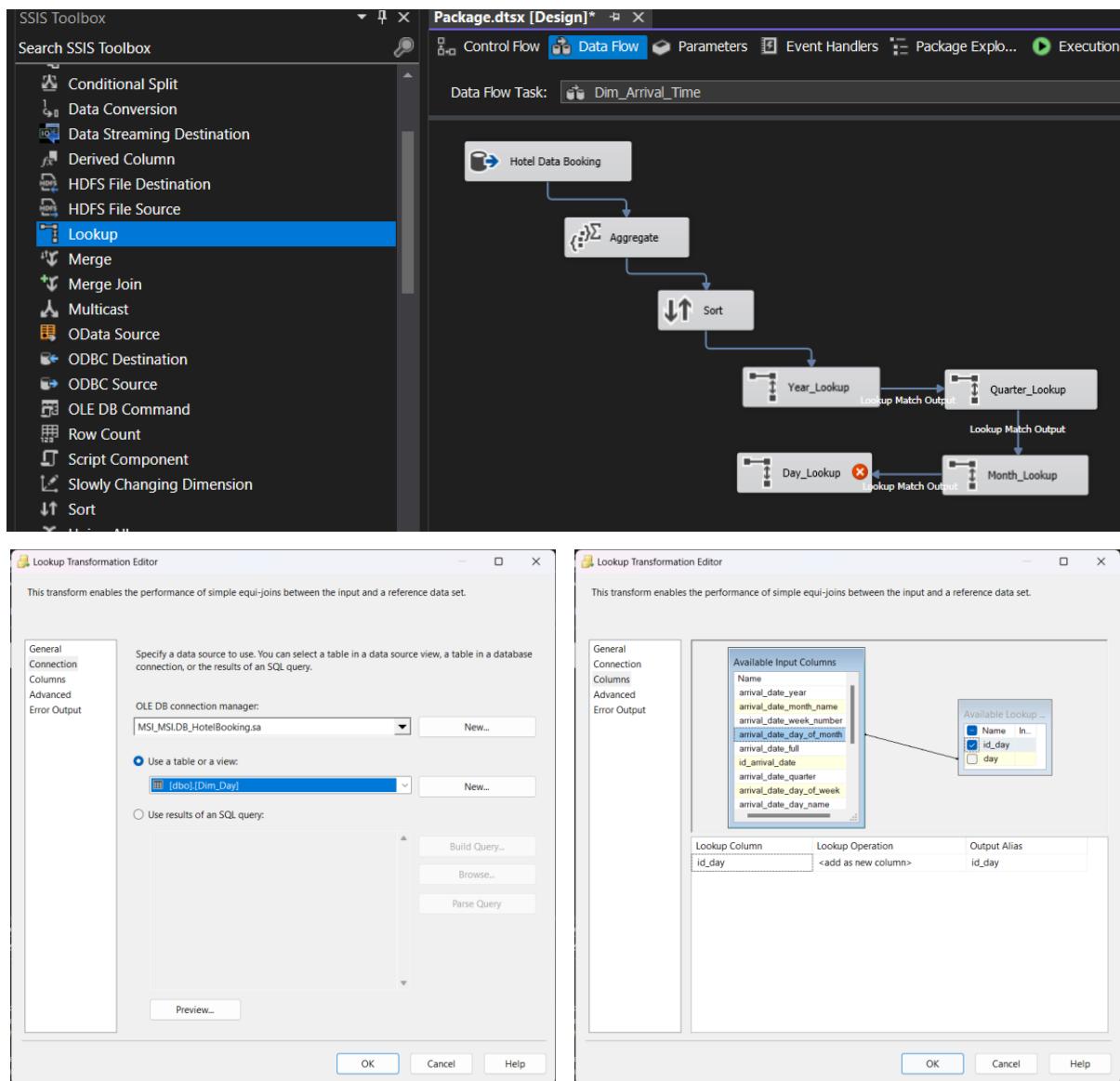
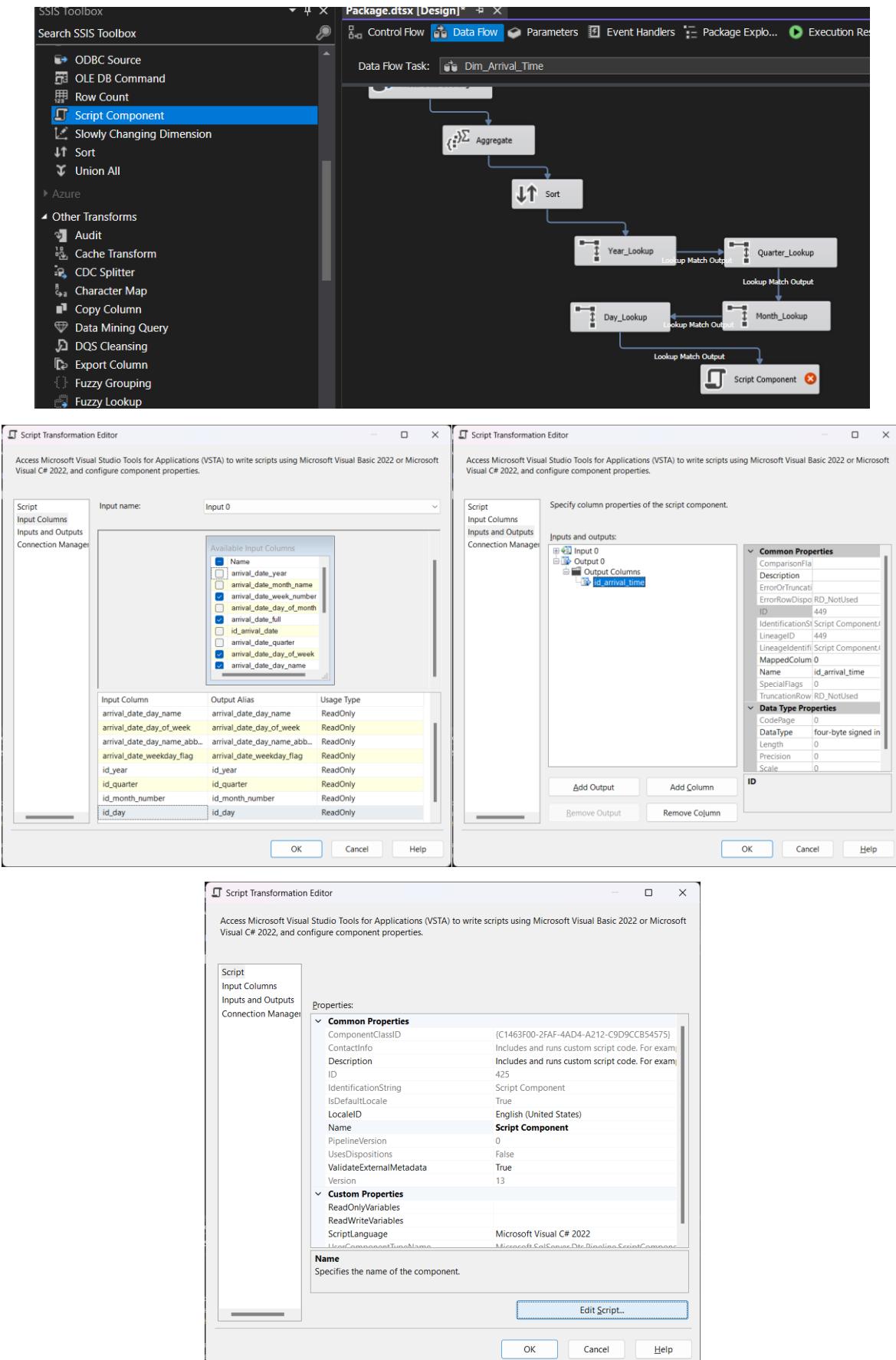
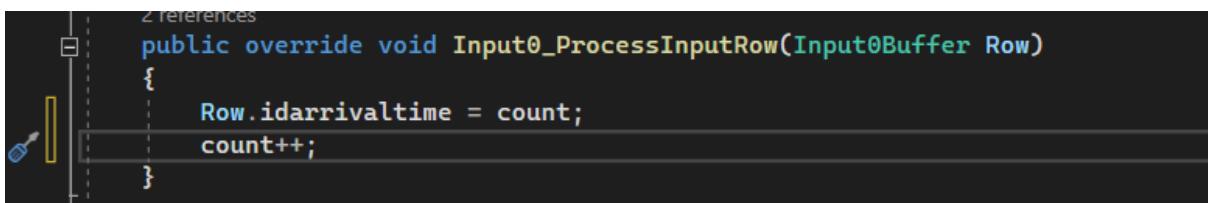


Figure 17-18-19. Tạo khóa ngoại id_arrival_day đến bảng Dim_Arrival_Time

- **Bước 11:** Kéo thả công cụ Script Component để tạo khóa chính id_arrival_time.





```

    2 references
    public override void Input0_ProcessInputRow(Input0Buffer Row)
    {
        Row.idarrivaltime = count;
        count++;
    }

```

Figure 20-21-22-23-24. Sử dụng Script Component để tạo khóa chính id_arrival_time

- **Bước 12:** Chọn công cụ OLE DB Destination để tiến hành tạo bảng trong cơ sở dữ liệu DB_HotelBooking với tên gọi là Dim_Arrival_Time.

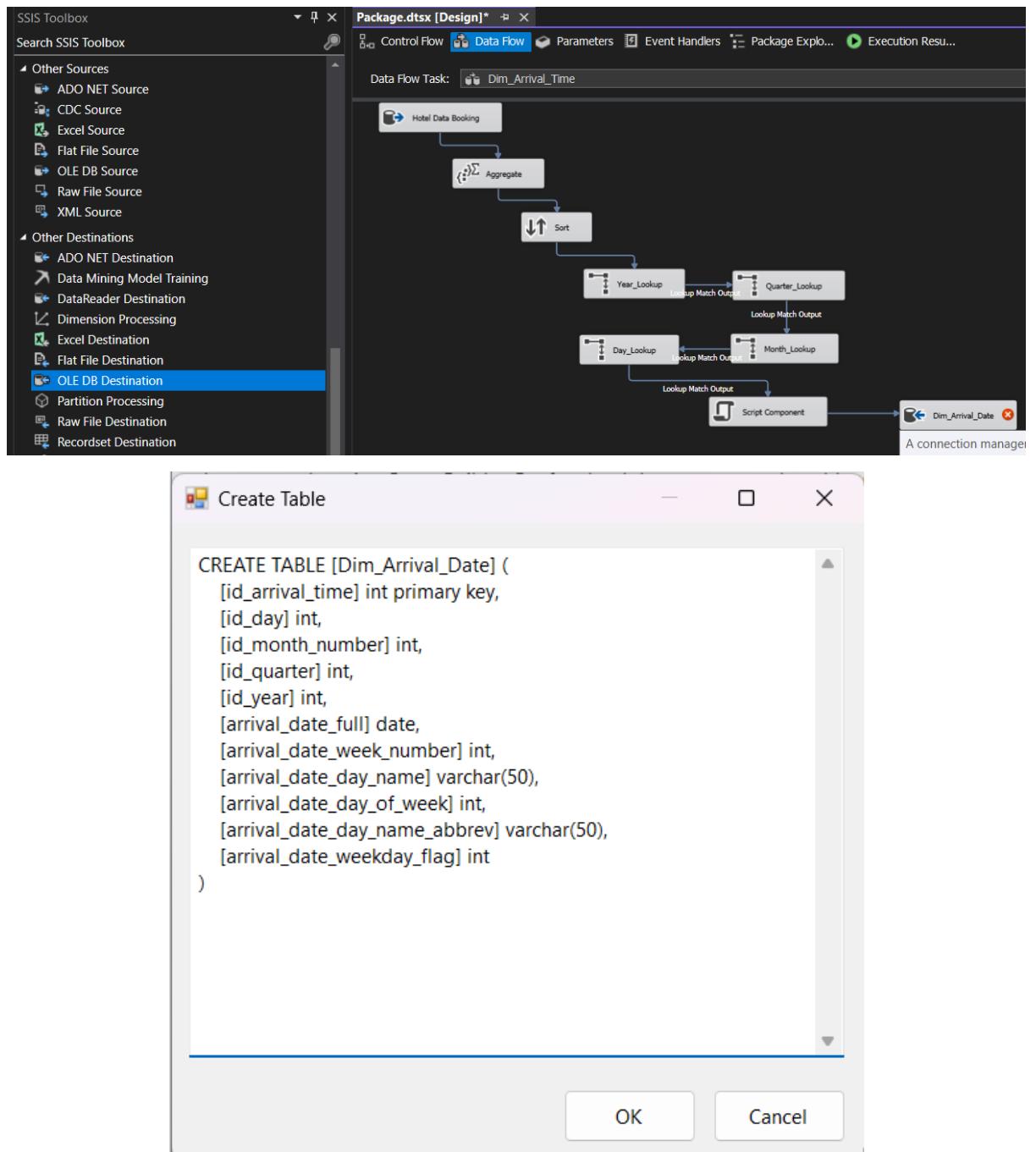


Figure 25-26. Tạo bảng Dim_Arrival_Time

- **Bước 13:** Nhấn Mappings để kiểm tra kết nối và nhấn Ok để hoàn tất quá trình tạo bảng.

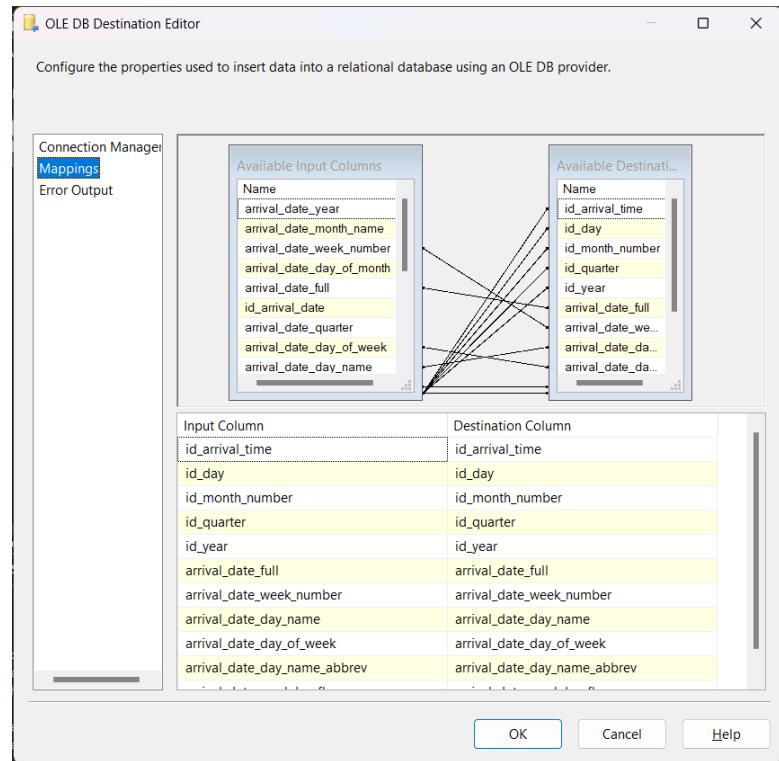


Figure 166. Quá trình Mappings dữ liệu

- **Bước 14:** Chọn Start để bắt đầu quá trình đổ dữ liệu. Khi đổ thành công, ta có kết quả như sau:

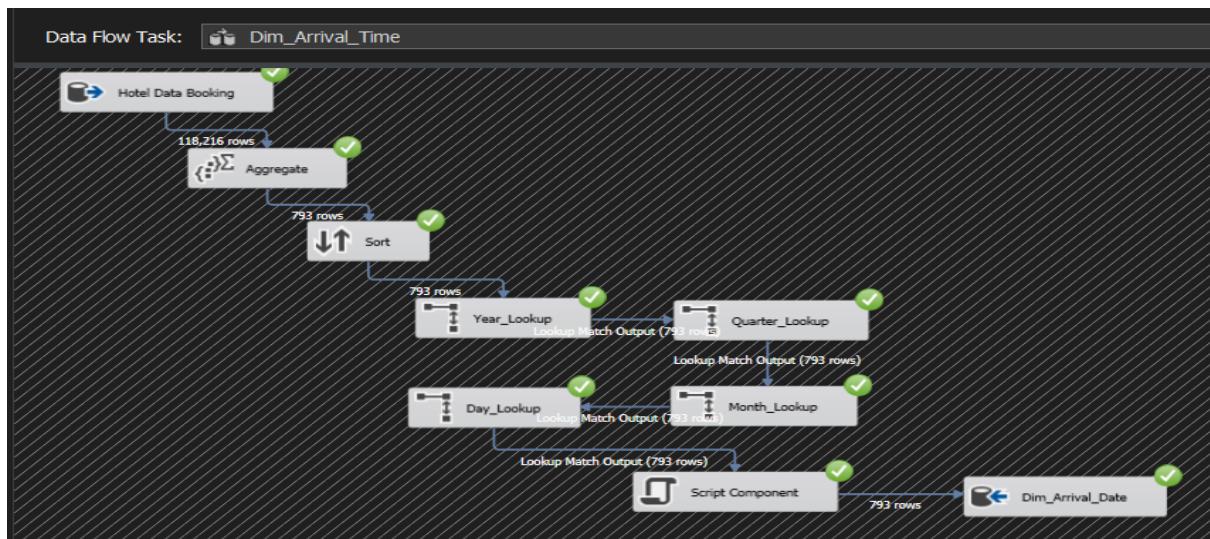


Figure 167. Hoàn thành đổ dữ liệu vào Dim_Arrival_Time trong kho dữ liệu

- **Bước 15:** Kiểm tra bảng Dim_Arrival_Time trên SQL Server.

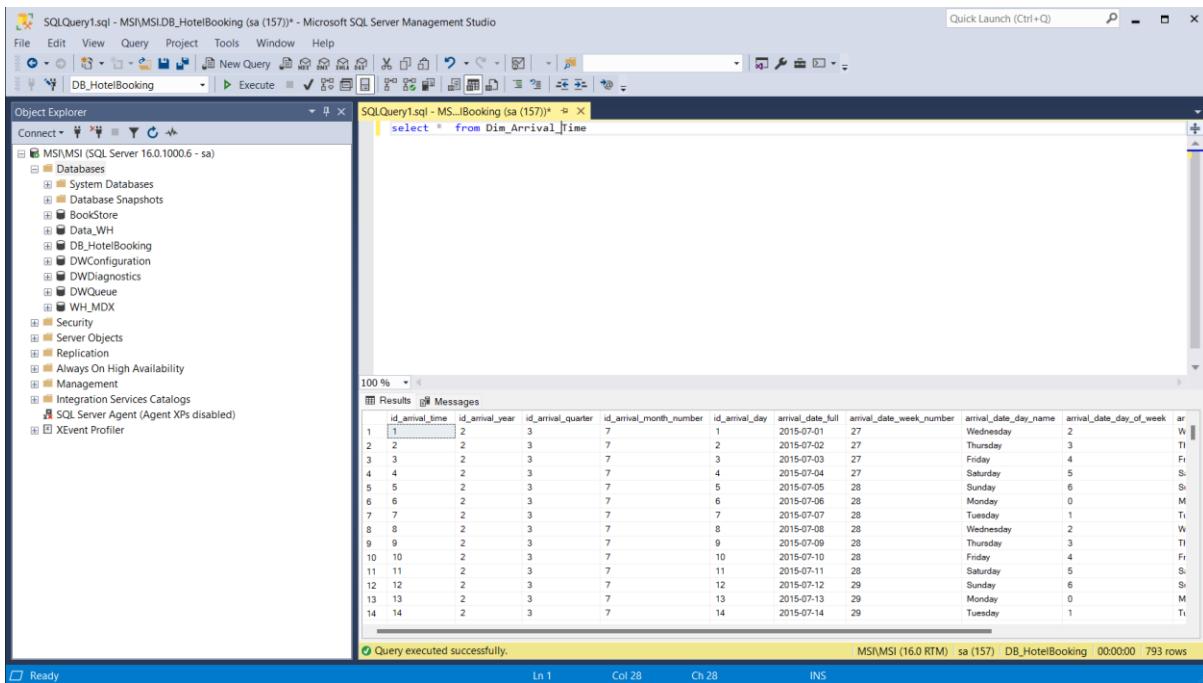


Figure 168. Kiểm tra bảng Dim_Arrival_Time trong SQL Server

2.5.14. Bảng Dim_Reservation_Time

- **Bước 1:** Kéo chức năng Data Flow Task từ cột trái sang màn hình làm việc và đổi tên thành Dim_Reservation_Time và tạo đường liên kết từ Dim_Year, Dim_Quarter, Dim_Month, Dim_Day đến Dim_Reservation_Time.

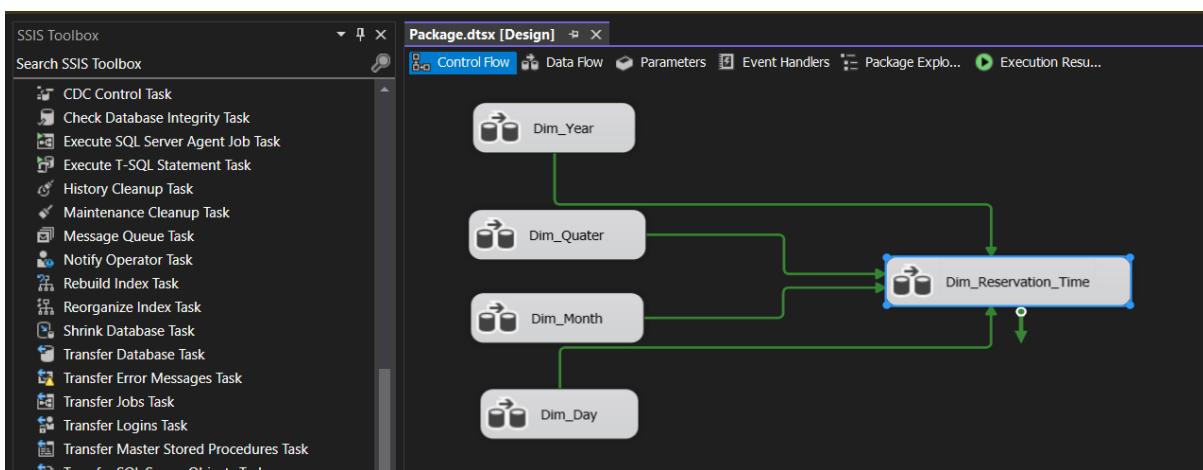


Figure 169. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Dim_Reservation_Time

- **Bước 2:** Nhấn double vào Data Flow Task vừa tạo và tìm kiếm chức năng OLE DB Source, kéo vào màn hình làm việc

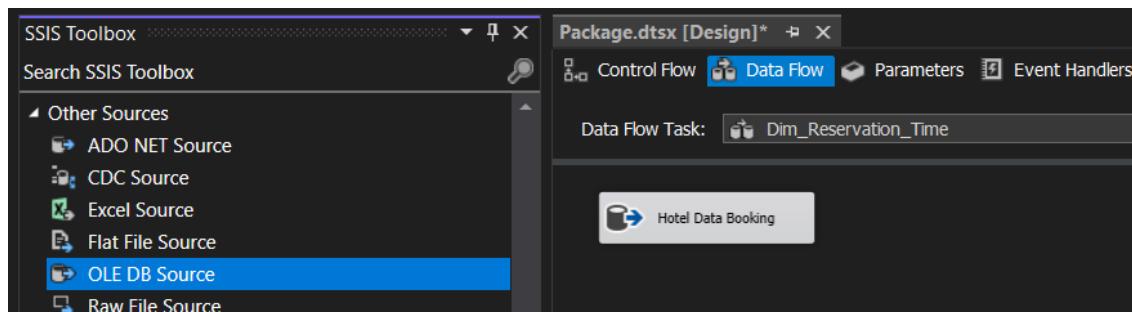


Figure 170. Kéo chức năng OLE DB Source vào màn hình làm việc

- **Bước 3:** Click double vào OLE DB Source và dẫn đến DB_HotelBooking. Sau đó chọn Name of the table or the view là [dbo].Hotel_Booking.

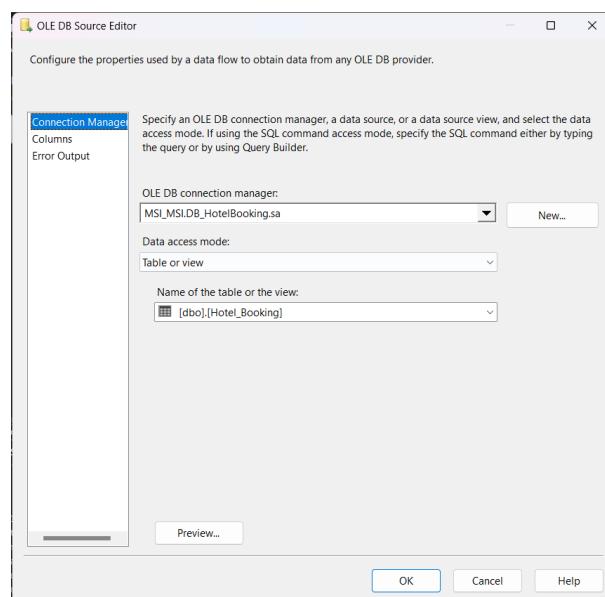


Figure 171. Chọn Table sẽ lấy dữ liệu

- **Bước 4:** Chọn Columns để lựa chọn các cột cần sử dụng và nhấn OK.

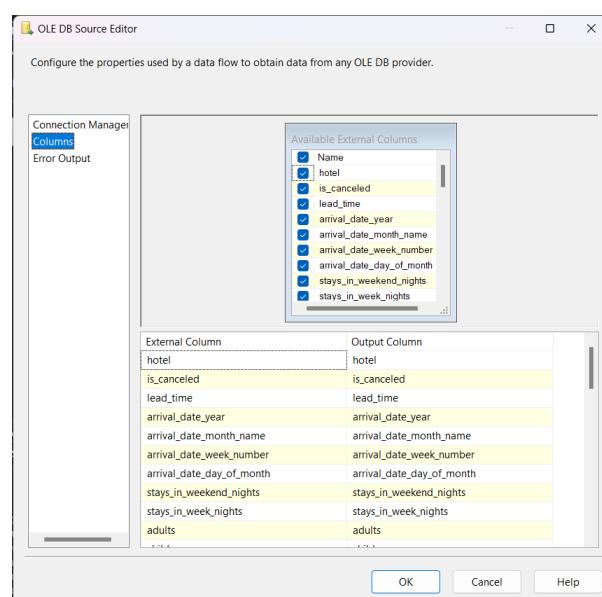


Figure 172. Chọn các column cần sử dụng

- **Bước 5:** Dùng công cụ Aggregate dùng để thực hiện lọc các thuộc tính trùng nhau lên thuộc tính sử dụng là id_reservation_status_date, id_reservation_status_date.

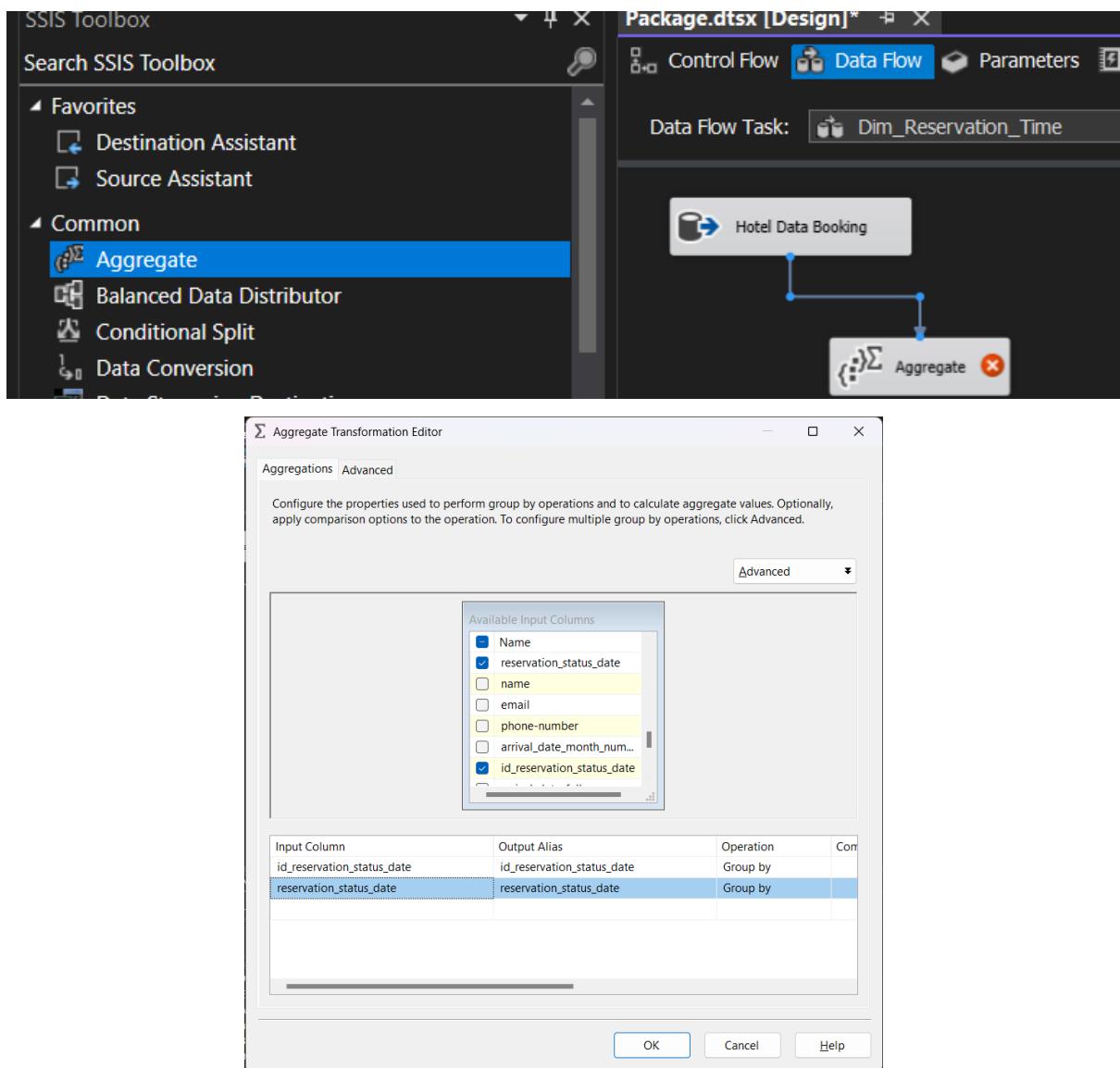
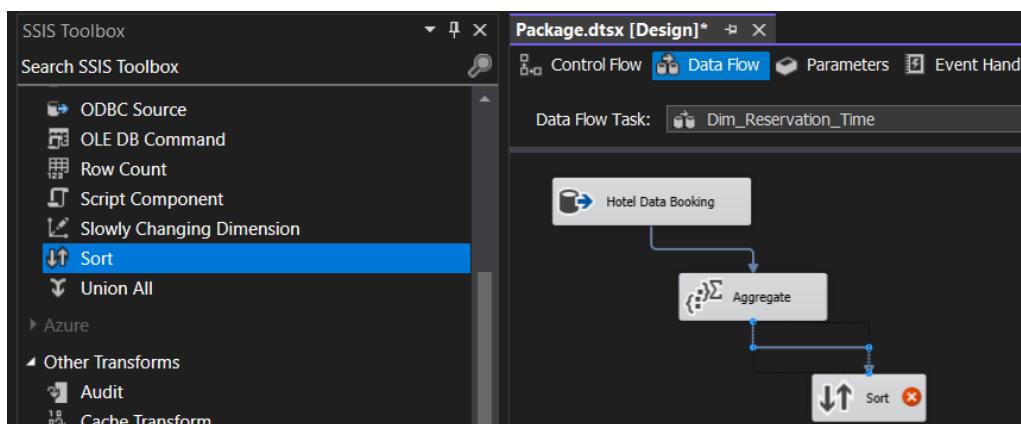


Figure 5-6. Dùng công cụ Aggregate dùng để lọc các thuộc tính trùng nhau

- **Bước 6:** Dùng công cụ Sort để sắp xếp các giá trị theo chiều tăng dần.



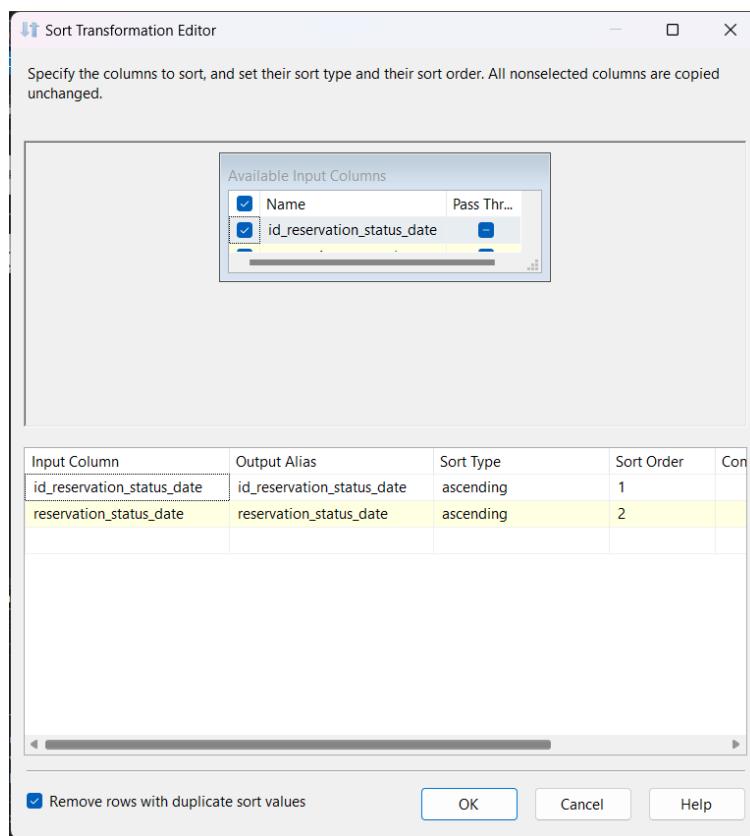
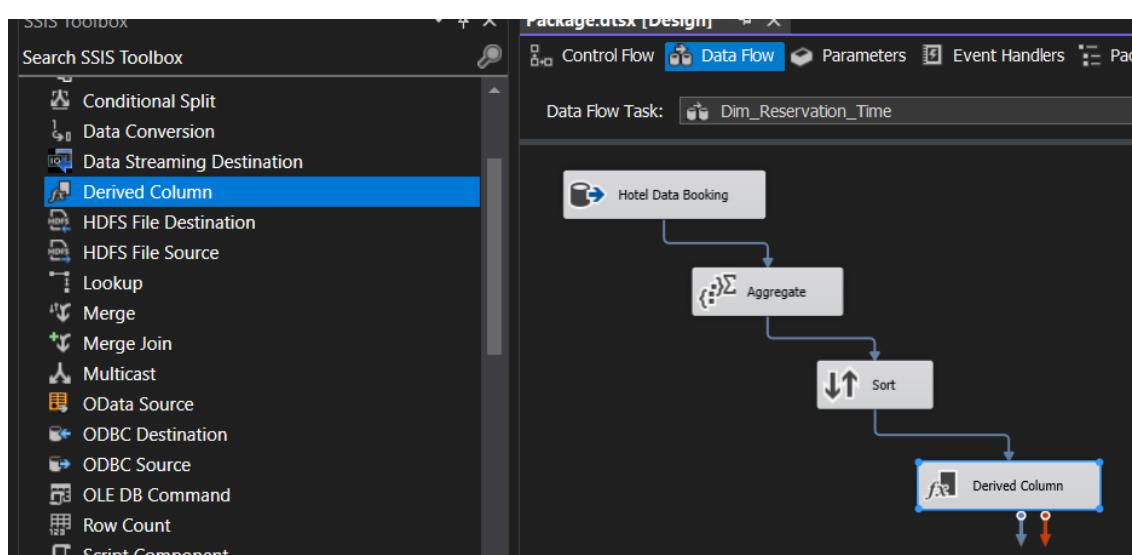


Figure 7-8. Sắp xếp các giá trị theo chiều tăng dần

- **Bước 7:** Sử dụng công cụ Derived Column, để lấy các thuộc tính reservation_date_day, reservation_date_month, reservation_date_quarter, reservation_date_year, reservation_date_day, reservation_date_day_of_year.



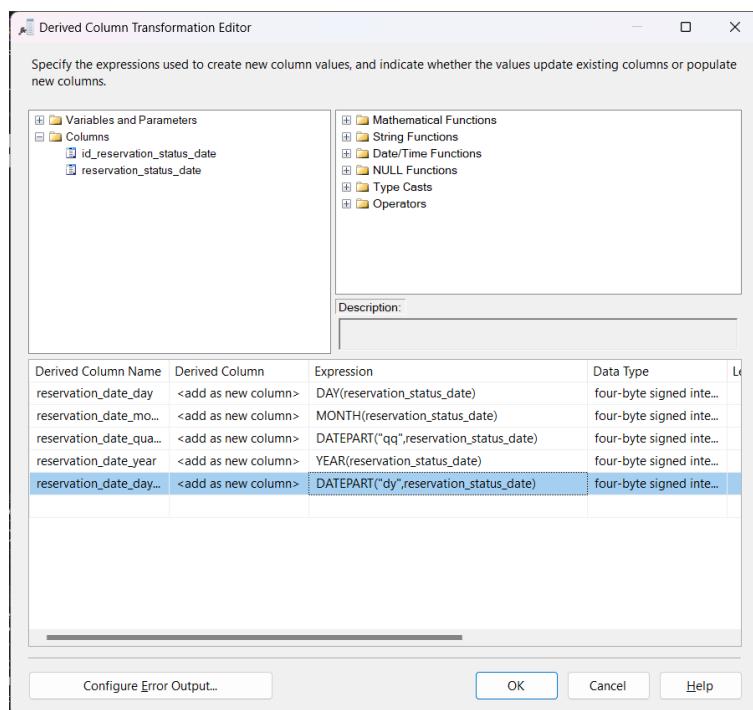
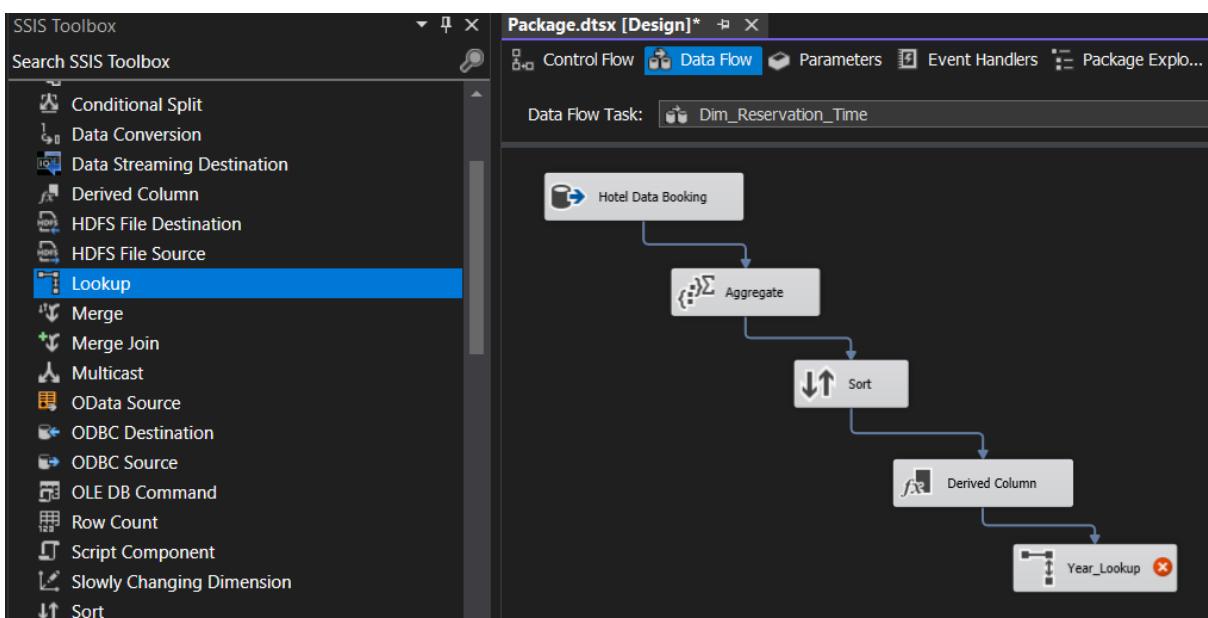


Figure 9-10. Tách reservation_status_date ra thành 5 thuộc tính nhỏ

- Bước 8:** Tìm kiếm công cụ Lookup trên SSIS Toolbox và đổi tên thành Year_Lookup để tạo khóa ngoại id_reservation_year đến bảng Dim_Reservation_Time.



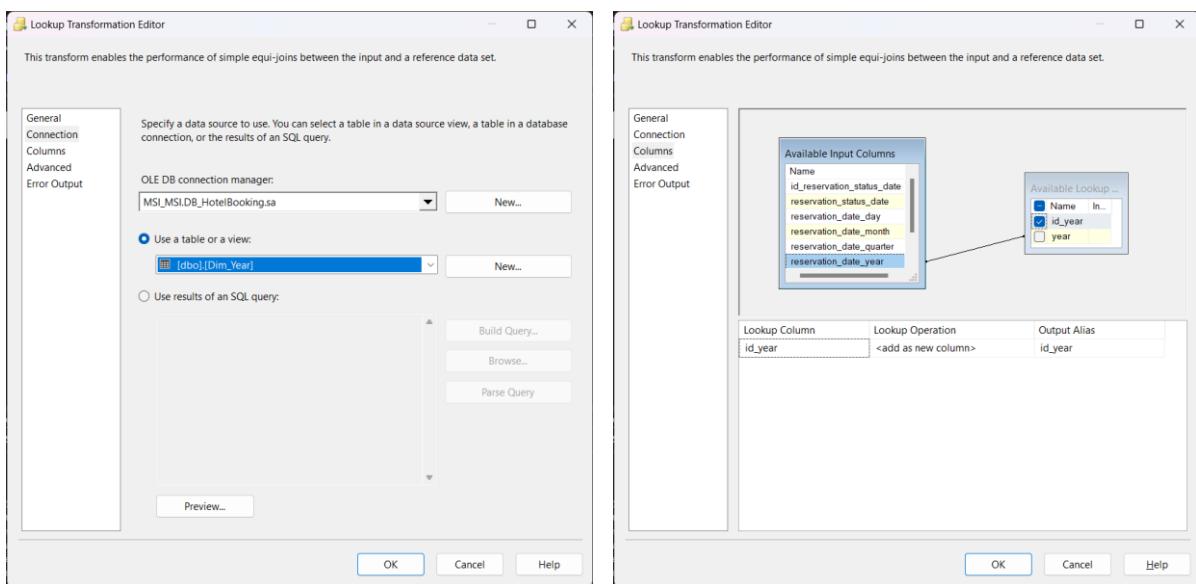


Figure 11-12-13. Tạo khóa ngoại id_reservation_year đến bảng Dim_Reservation_Time

- **Bước 9:** Tìm kiếm công cụ Lookup trên SSIS Toolbox và đổi tên thành Quarter_Lookup để tạo khóa ngoại id_reservation_quarter đến bảng Dim_Reservation_Time.

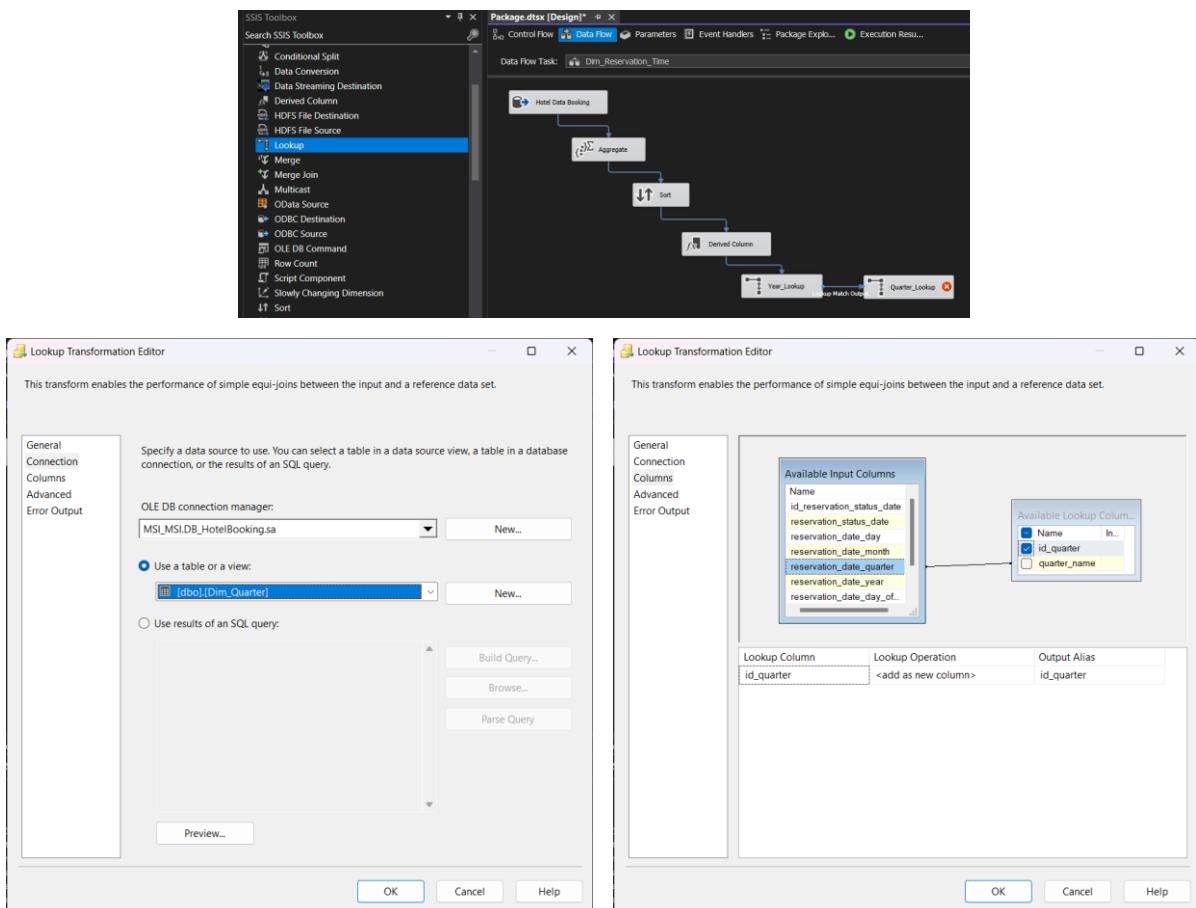


Figure 14-15-16. Tạo khóa ngoại id_reservation_quarter đến bảng Dim_Reservation_Time

- **Bước 10:** Tìm kiếm công cụ Lookup trên SSIS Toolbox và đổi tên thành Month_Lookup để tạo khóa ngoại id_reservation_month đến bảng Dim_Reservation_Time.

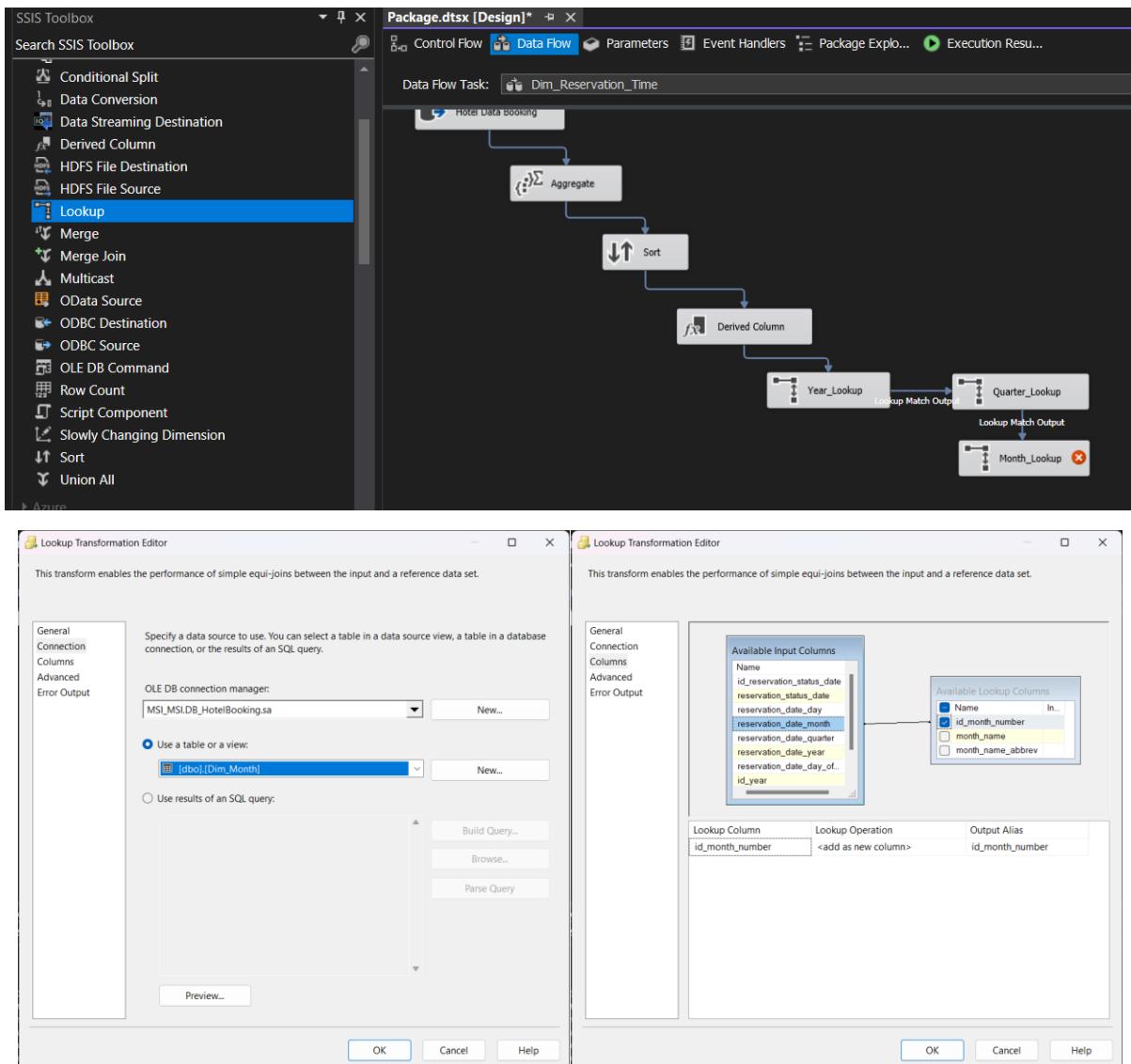


Figure 17-18-19. Tạo khóa ngoại id_reservation_month đến bảng Dim_Reservation_Time

- **Bước 11:** Tìm kiếm công cụ Lookup trên SSIS Toolbox và đổi tên thành Day_Lookup để tạo khóa ngoại id_reservation_day đến bảng Dim_Reservation_Time.

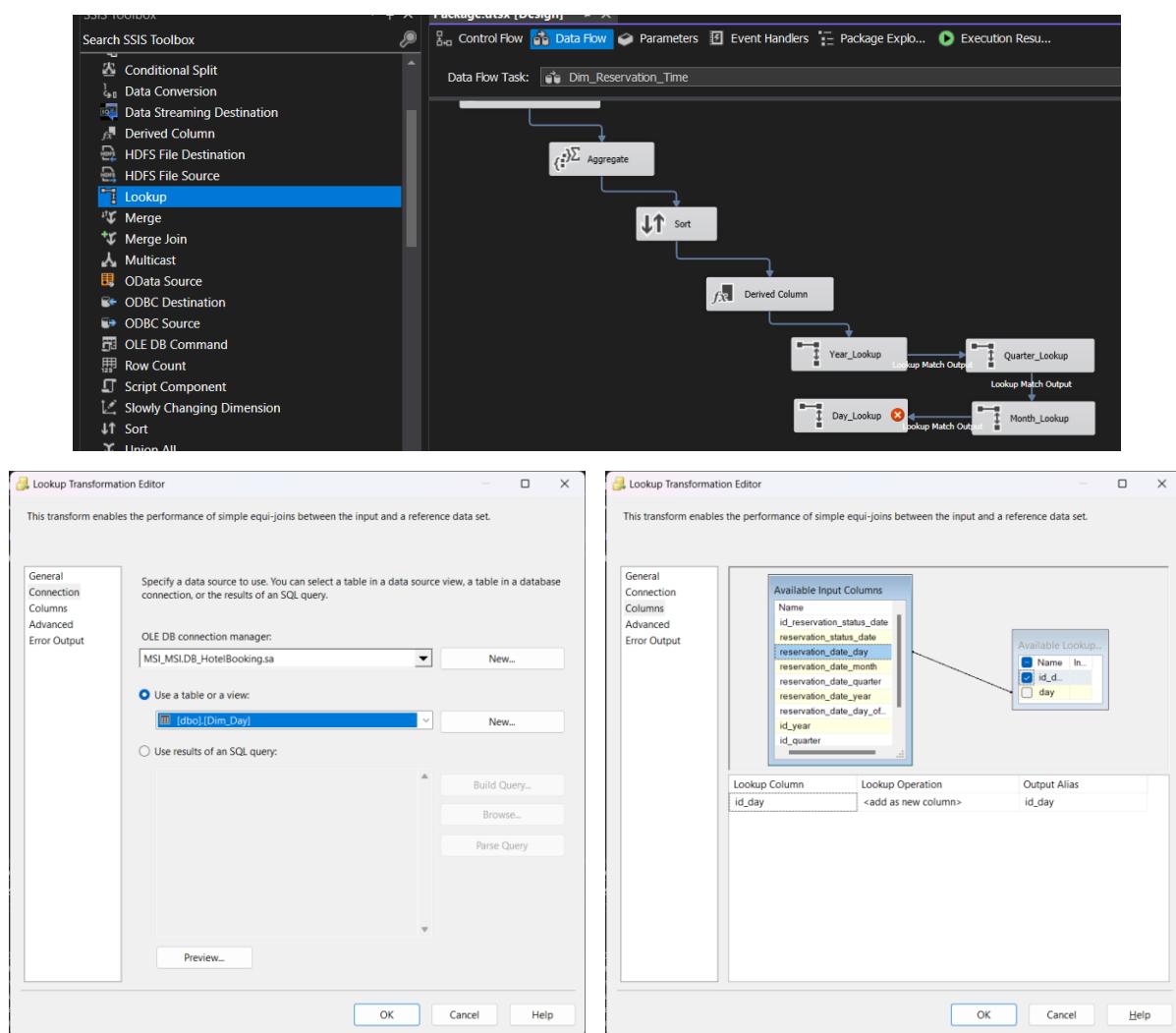
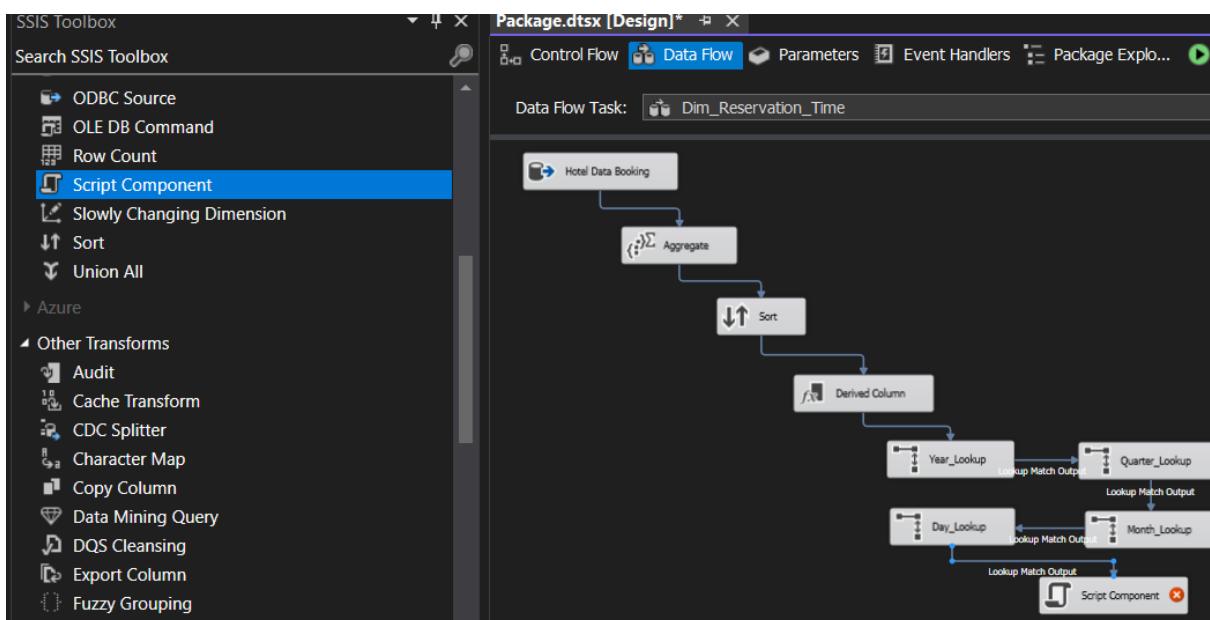


Figure 20-21-22. Tạo khóa ngoại id_reservation_day đến bảng Dim_Reservation_Time

- **Bước 12:** Kéo thả công cụ Script Component để tạo khóa chính id_reservation_time.



IS217.N21 – Kho dữ liệu và phân tích hoạt động đặt phòng khách sạn

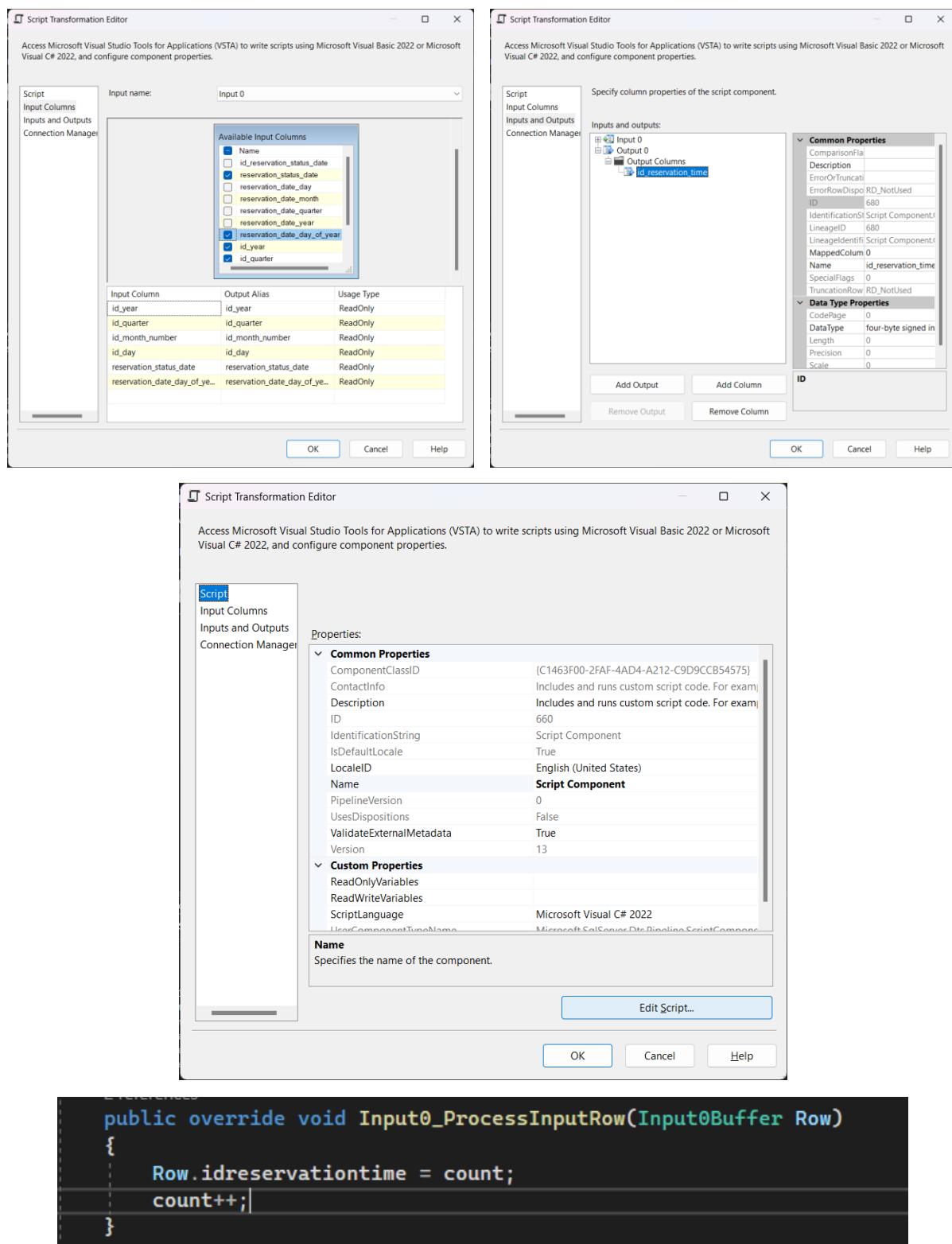


Figure 23-24-25-26-27. Sử dụng Script Component để tạo khóa chính id_reservation_time

- **Bước 13:** Chọn công cụ OLE DB Destination để tiến hành tạo bảng trong cơ sở dữ liệu DB_HotelBooking với tên gọi là Dim_Reservation_Time.

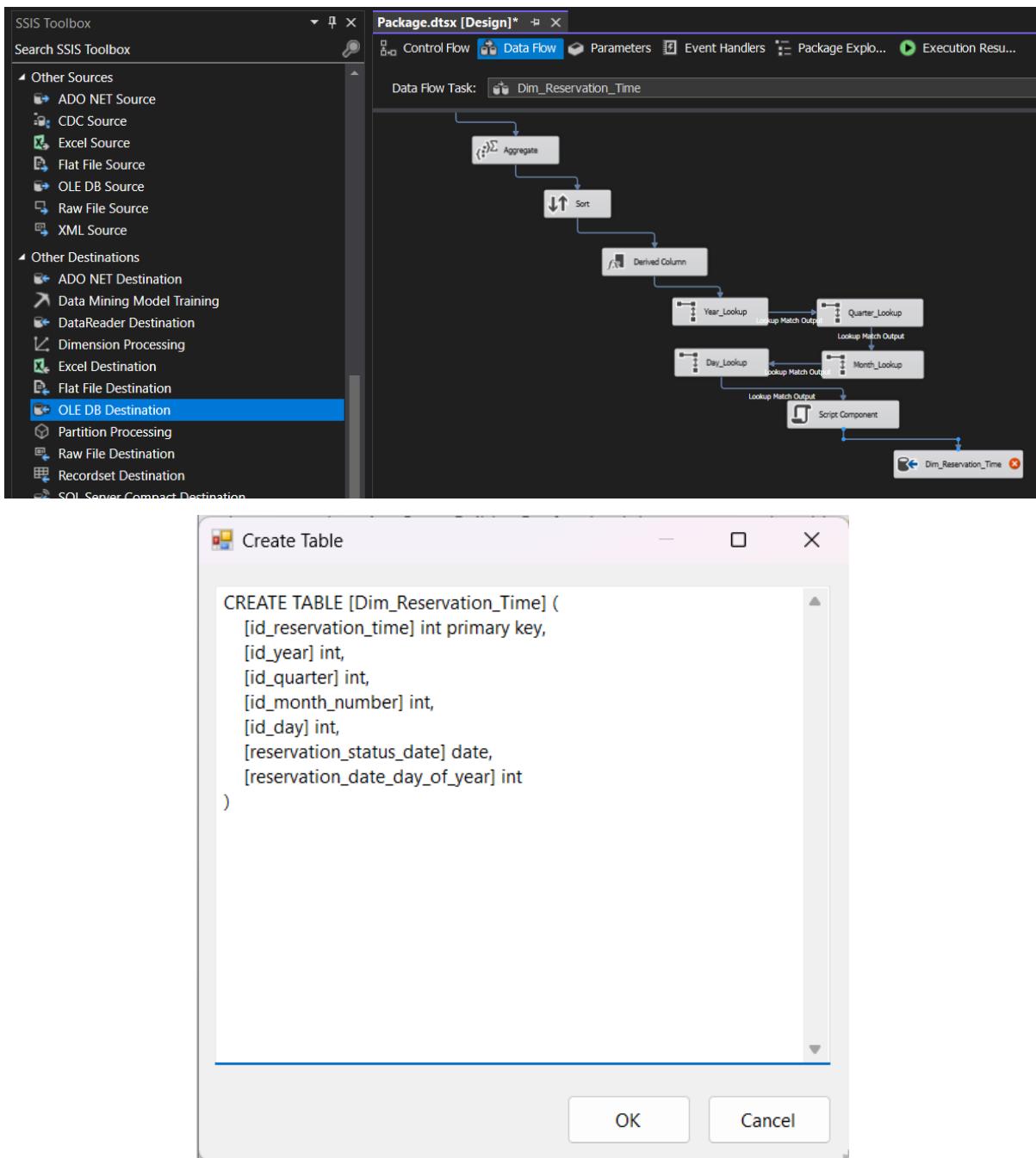


Figure 28-29. Tạo bảng Dim_Reservation_Time

- **Bước 14:** Nhấn Mappings để kiểm tra kết nối và nhấn Ok để hoàn tất quá trình tạo bảng.

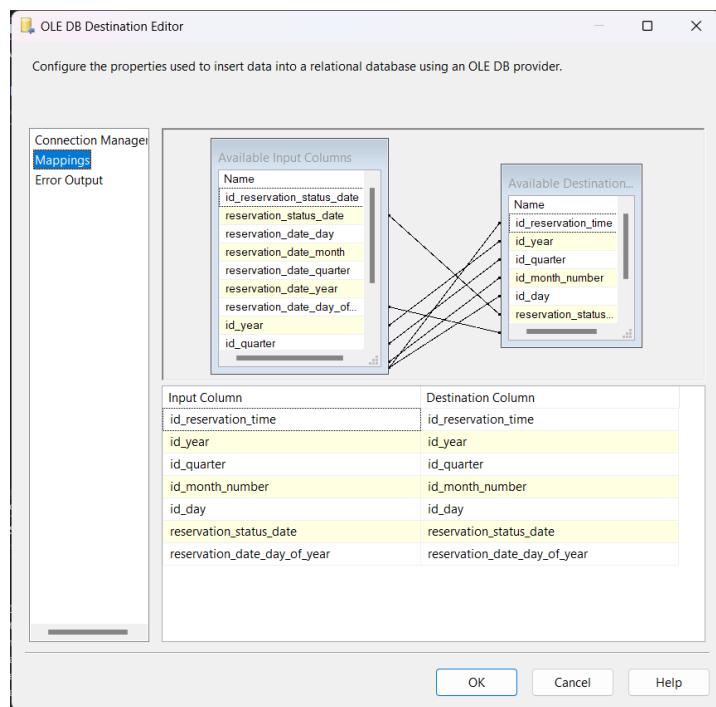


Figure 173. Quá trình Mappings dữ liệu

- **Bước 15:** Chọn Start để bắt đầu quá trình đổ dữ liệu. Khi đổ thành công, ta có kết quả như sau:

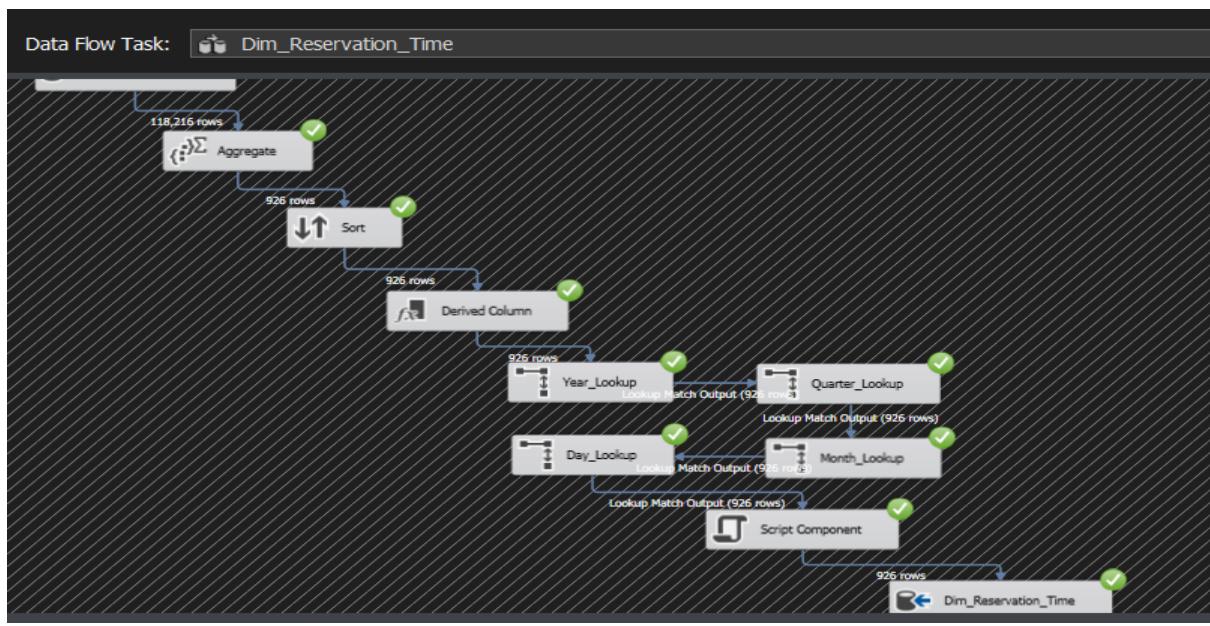


Figure 174. Hoàn thành đổ dữ liệu vào Dim_Reservation_Time trong kho dữ liệu

- **Bước 16:** Kiểm tra bảng Dim_Reservation_Time trên SQL Server.

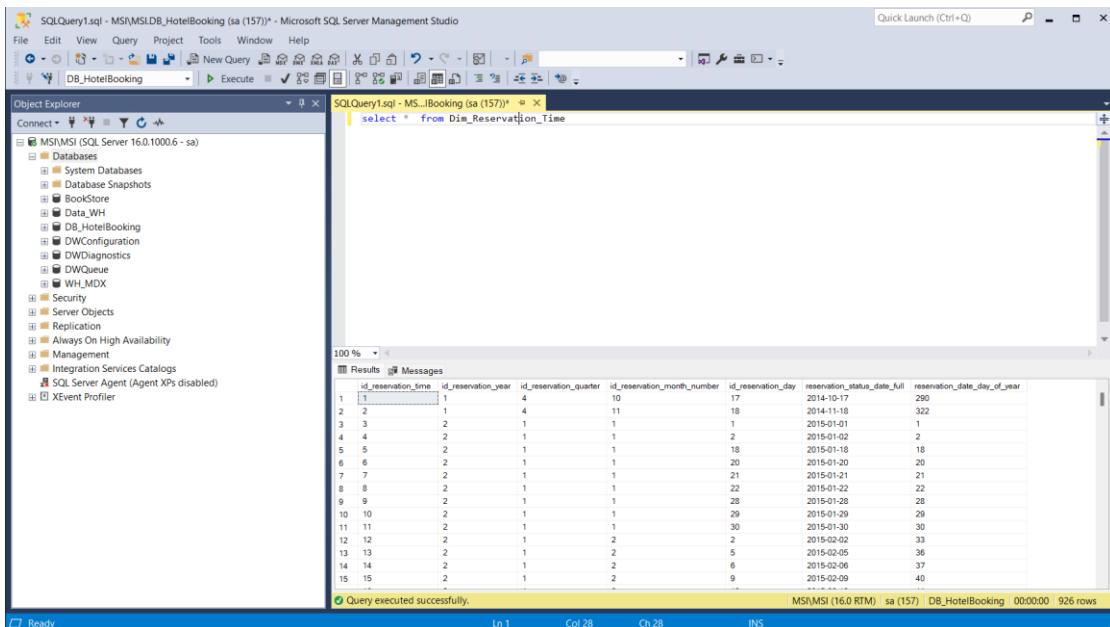


Figure 175. Kiểm tra bảng Dim_Reservation_Time trong SQL Server

2.5.15. Bảng Fact

- **Bước 1:** Kéo chức năng Data Flow Task từ cột trái sang màn hình làm việc và đổi tên thành Create Fact Table.

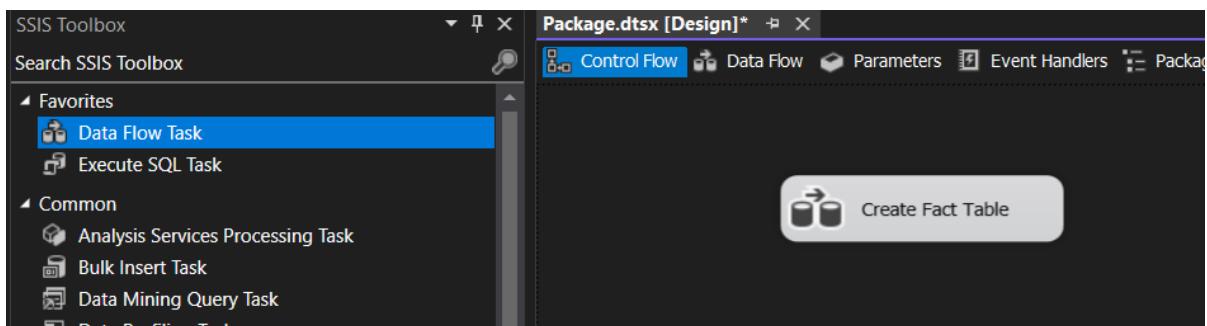


Figure 176. Kéo Data Flow Task vào màn hình làm việc và đổi tên thành Fact

- **Bước 2:** Nhấn double vào Data Flow Task vừa tạo và tìm kiếm chức năng OLE DB Source, kéo vào màn hình làm việc.

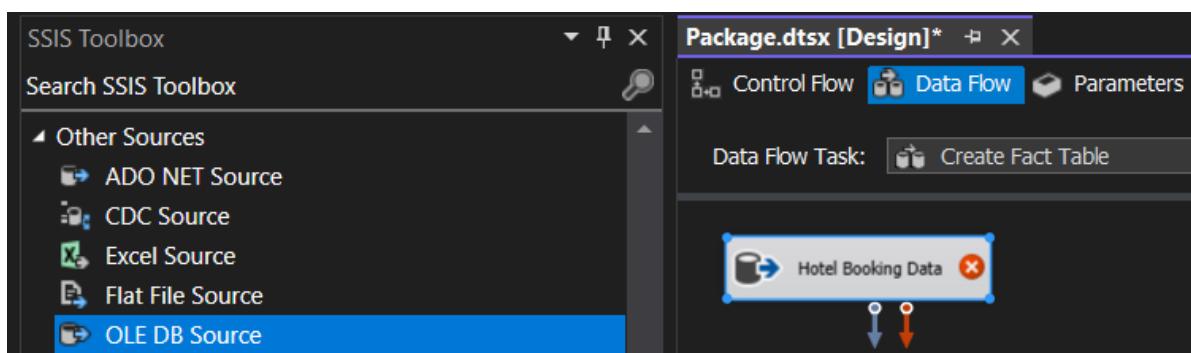


Figure 177. Kéo chức năng OLE DB Source vào màn hình làm việc

- **Bước 3:** Click double vào OLE DB Source và dẫn đến DB_HotelBooking. Sau đó chọn Name of the table or the view là [dbo].Hotel_Booking.

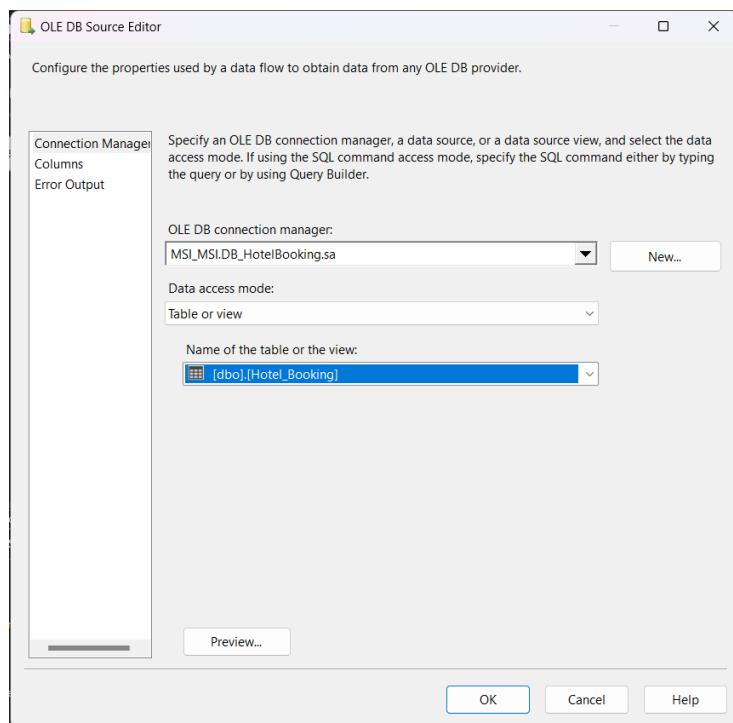


Figure 178. Chọn Table sẽ lấy dữ liệu

- **Bước 4:** Chọn Columns để lựa chọn các cột cần sử dụng và nhấn OK.

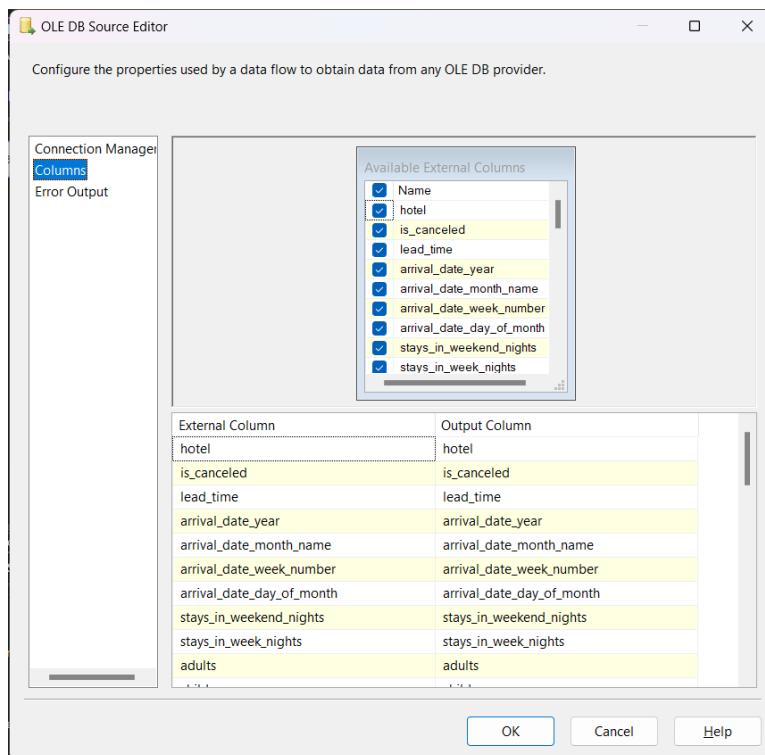


Figure 179. Chọn các column cần sử dụng

- **Bước 5:** Tìm kiếm công cụ Lookup trên SSIS Toolbox và tiến hành tạo các khóa ngoại từ các bảng Dim đến bảng Fact.

IS217.N21 – Kho dữ liệu và phân tích hoạt động đặt phòng khách sạn

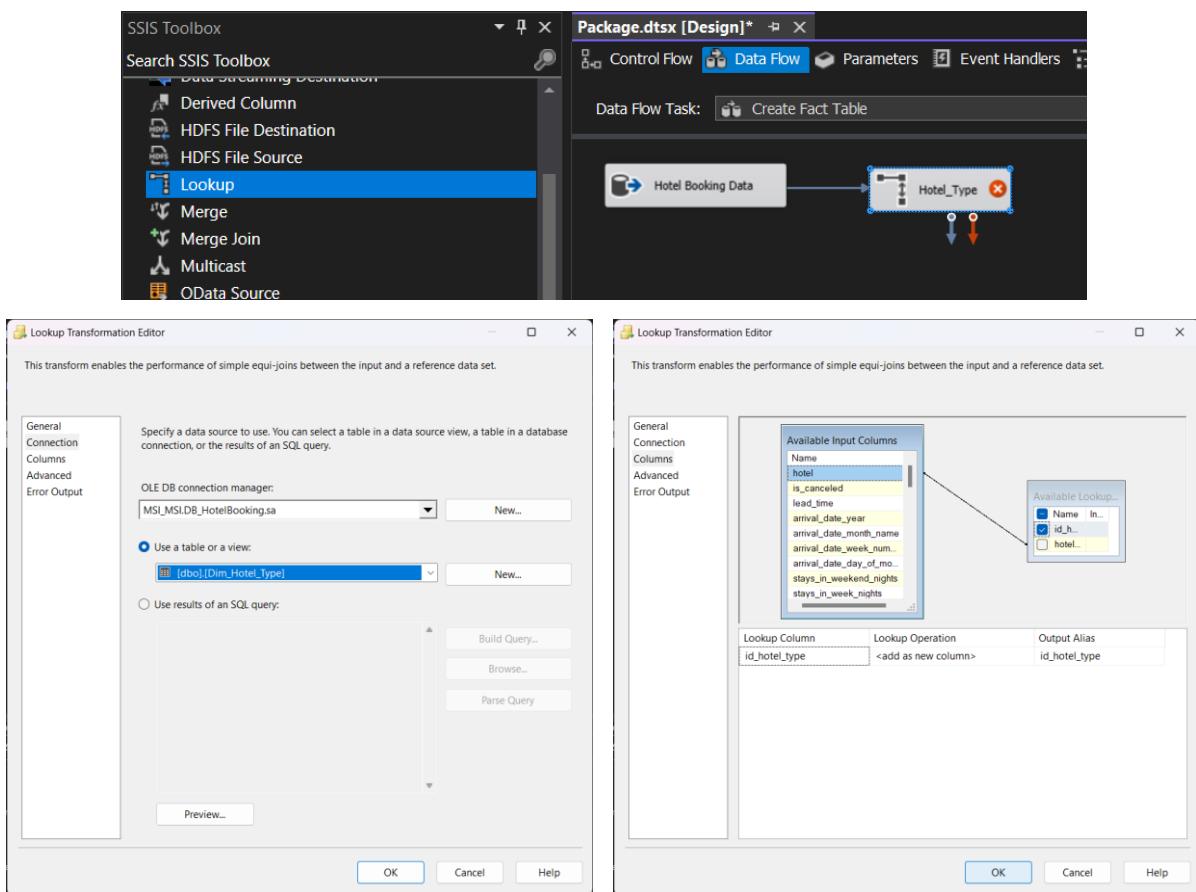


Figure 5-6-7. Tạo khóa ngoại id_hotel_type đến bảng Fact

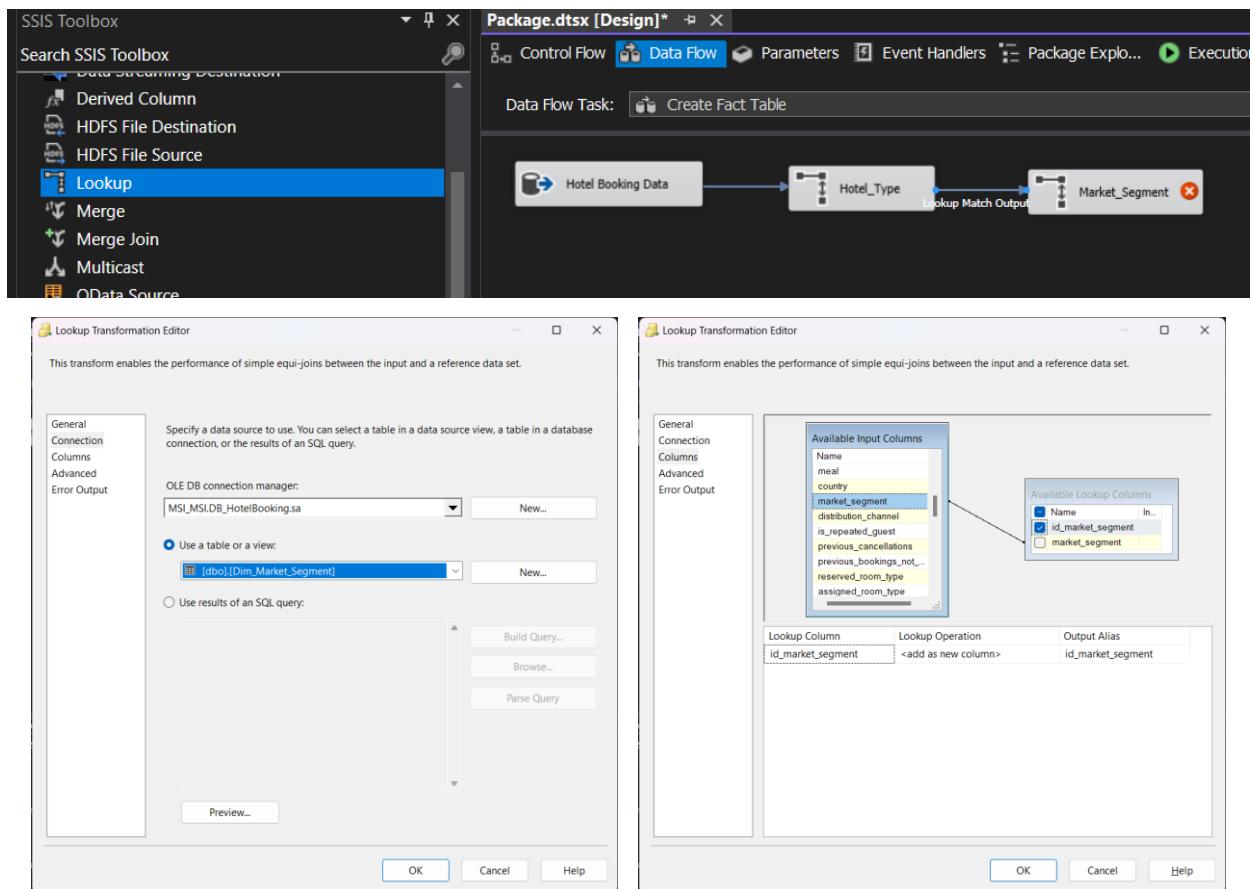


Figure 8-9-10. Tạo khóa ngoại id_market_segment đến bảng Fact

IS217.N21 – Kho dữ liệu và phân tích hoạt động đặt phòng khách sạn

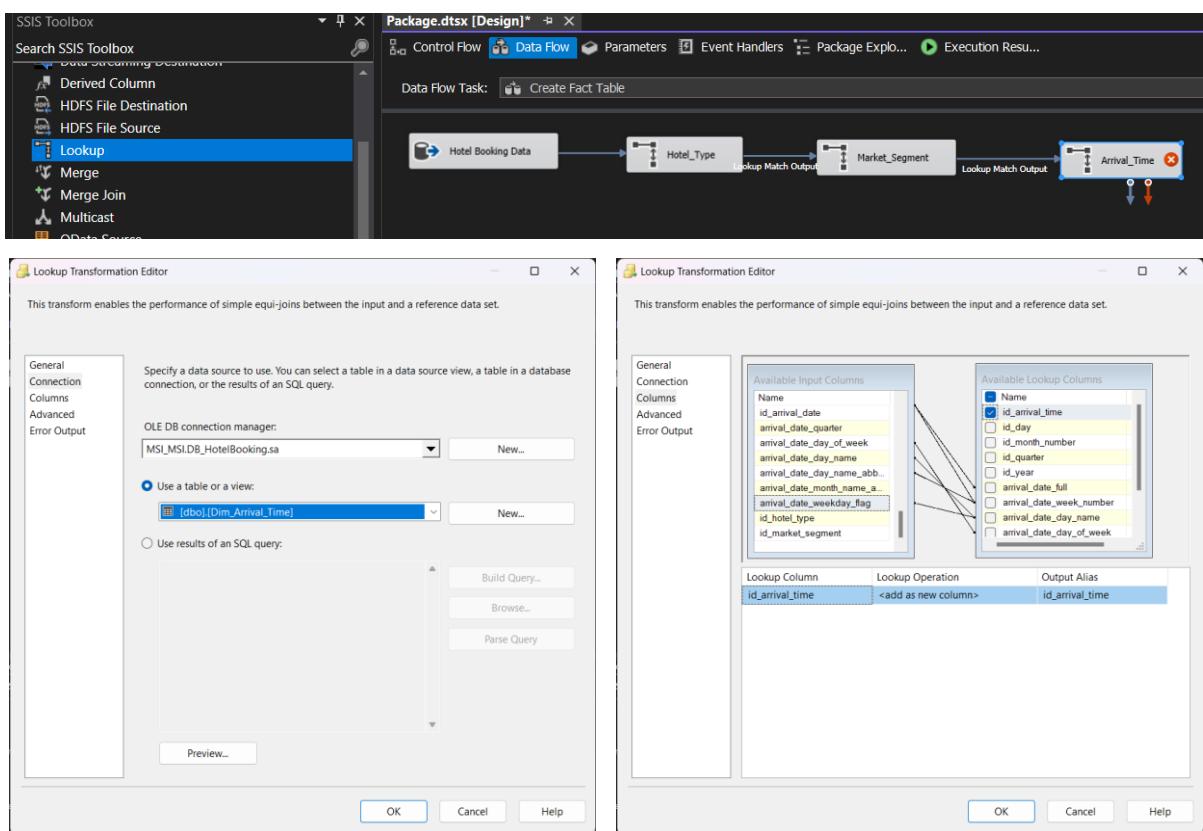


Figure 11-12-13. Tạo khóa ngoại id_arrival_time đến bảng Fact

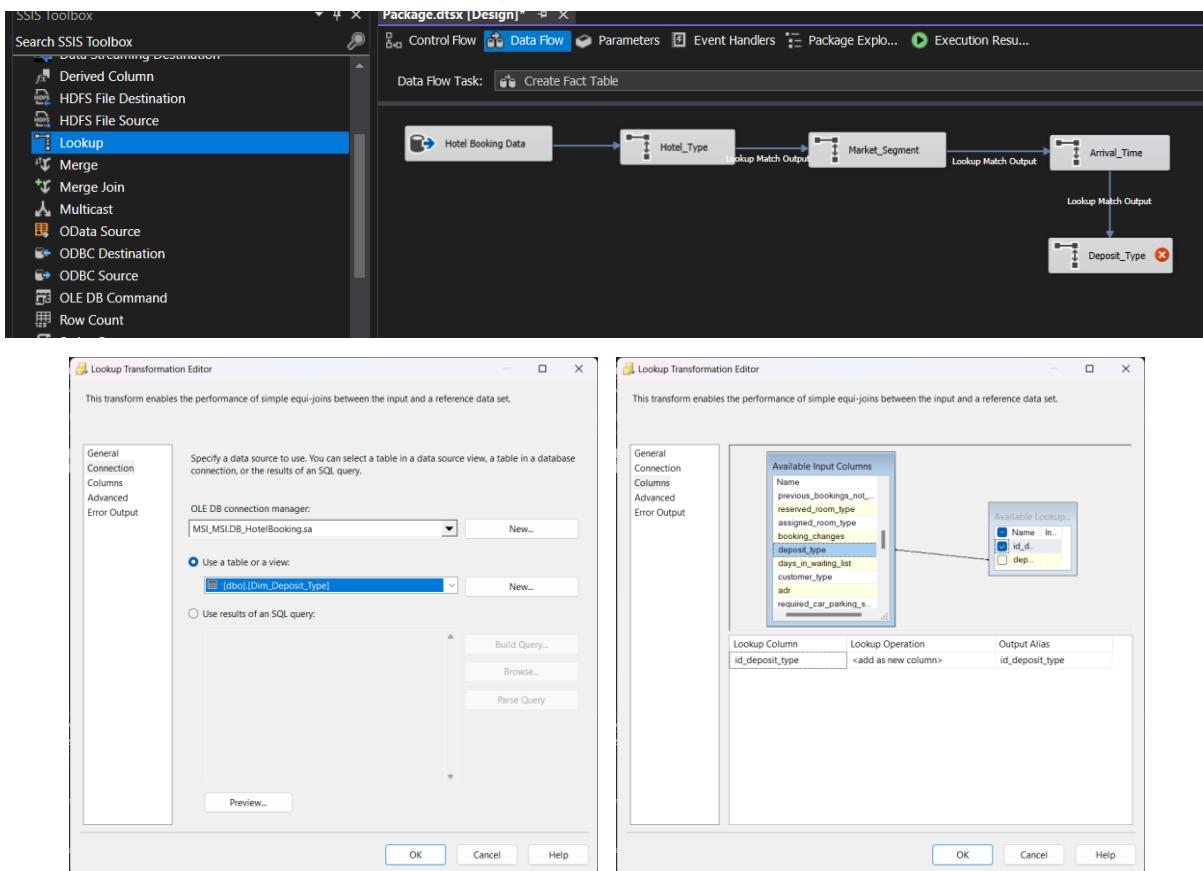


Figure 14-15-16. Tạo khóa ngoại id_deposit_type đến bảng Fact

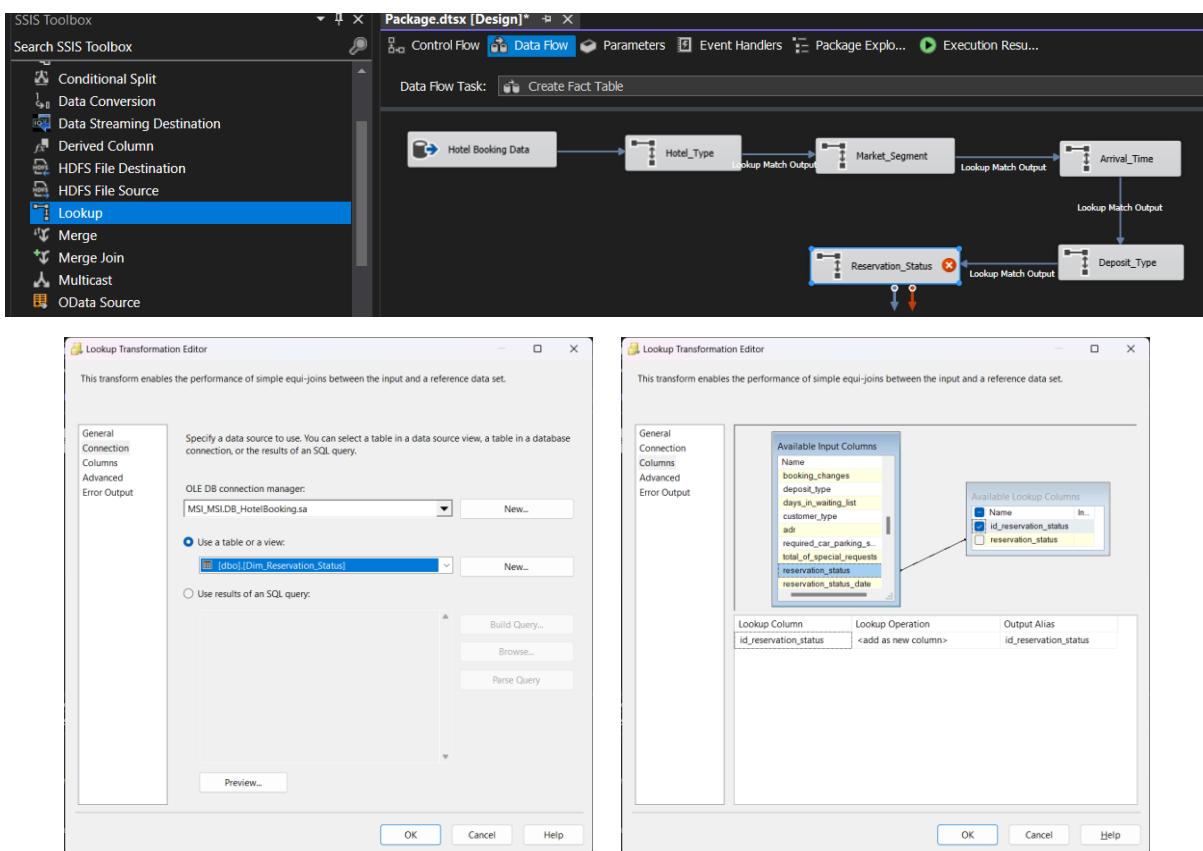


Figure 17-18-19. Tạo khóa ngoại id_reservation_status đến bảng Fact

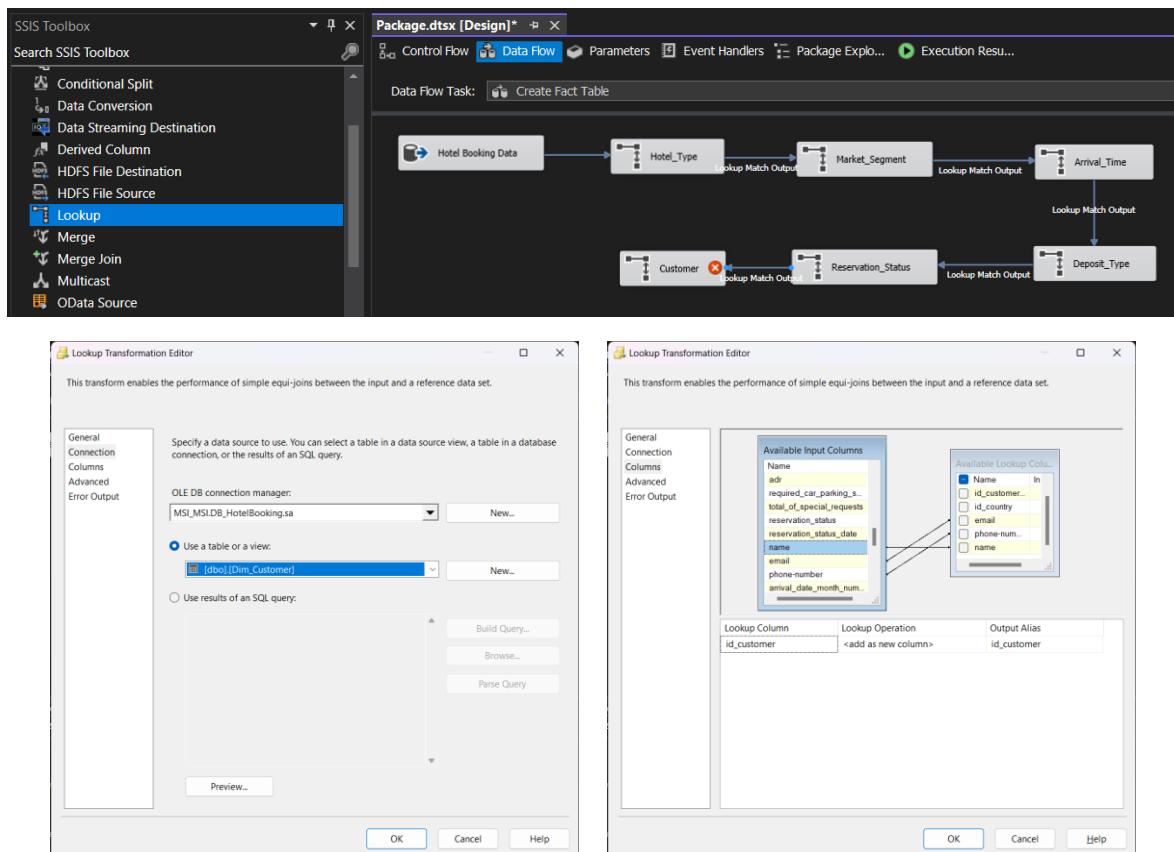


Figure 20-21-22. Tạo khóa ngoại id_customer đến bảng Fact

IS217.N21 – Kho dữ liệu và phân tích hoạt động đặt phòng khách sạn

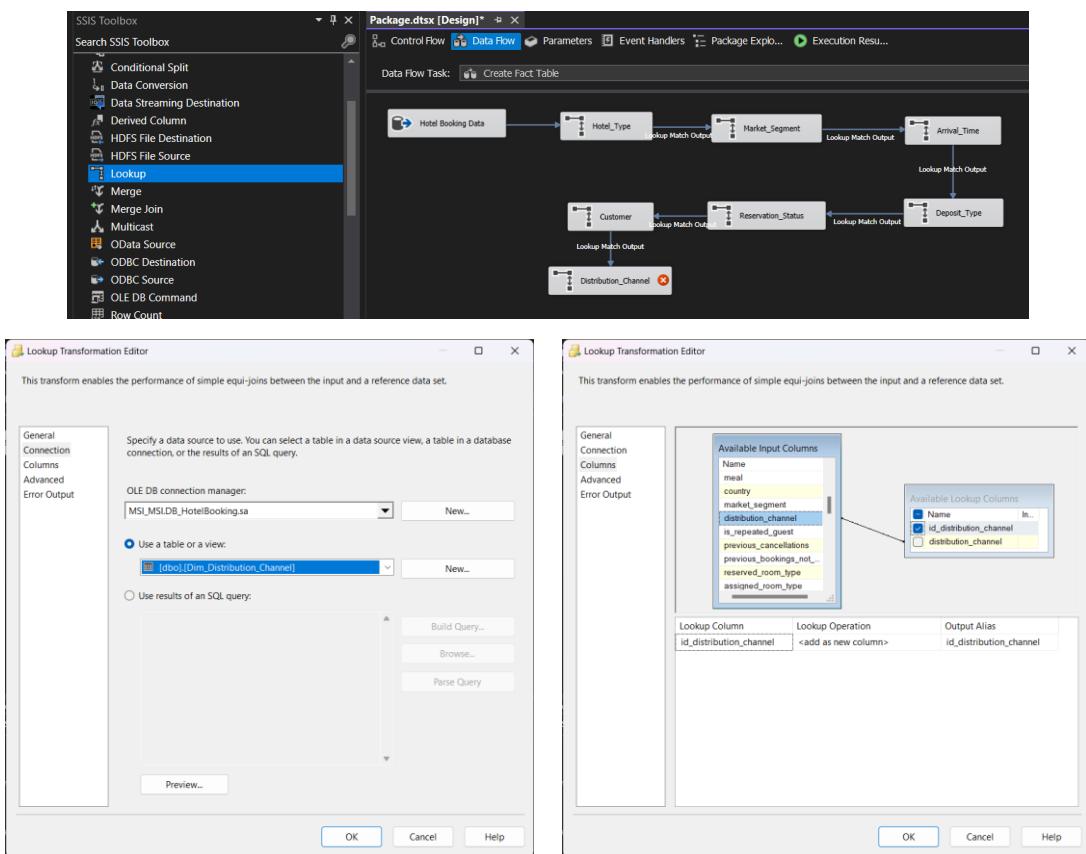


Figure 23-24-25. Tạo khóa ngoại id_distrinbution_channel đến bảng Fact

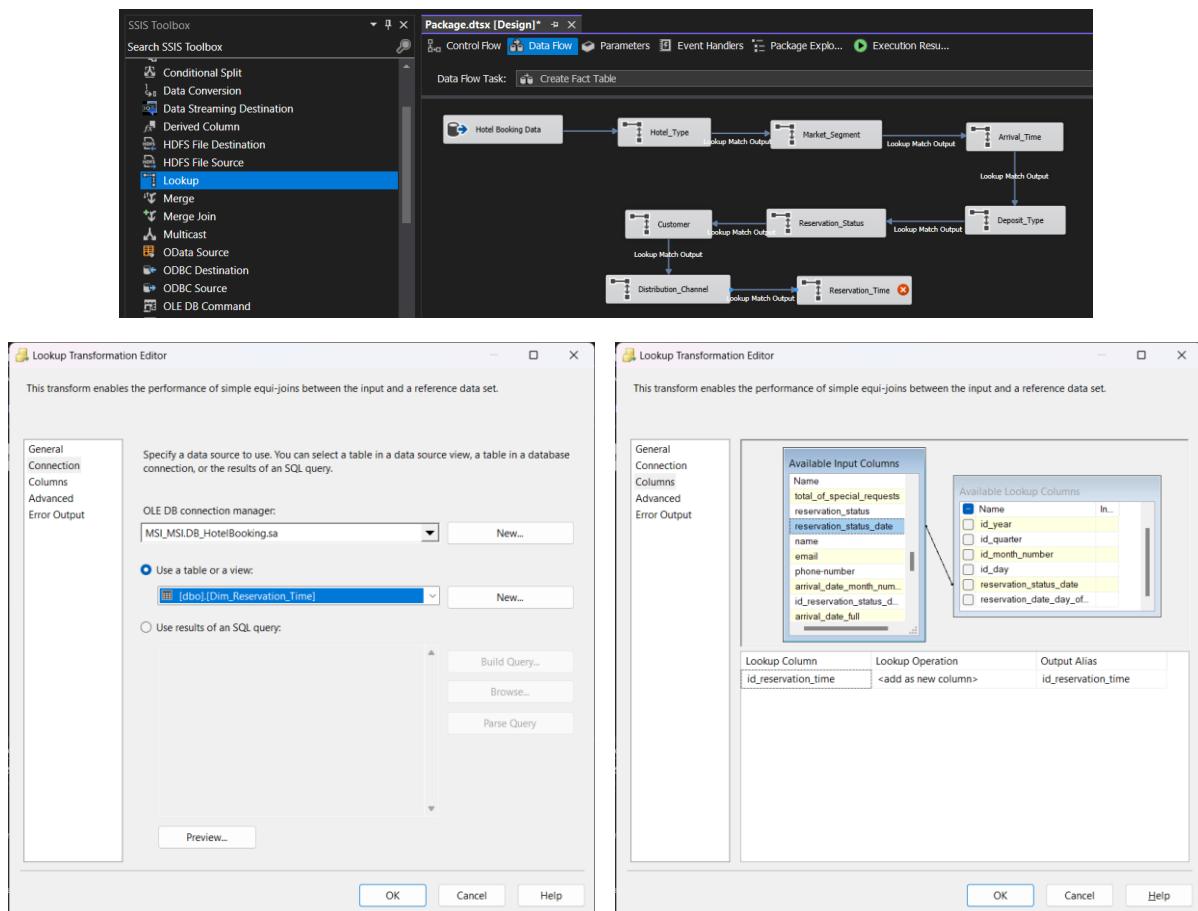


Figure 26-27-28. Tạo khóa ngoại id_reservation_time đến bảng Fact

- **Bước 6:** Kéo thả công cụ Script Component để tạo khóa chính id_fact.

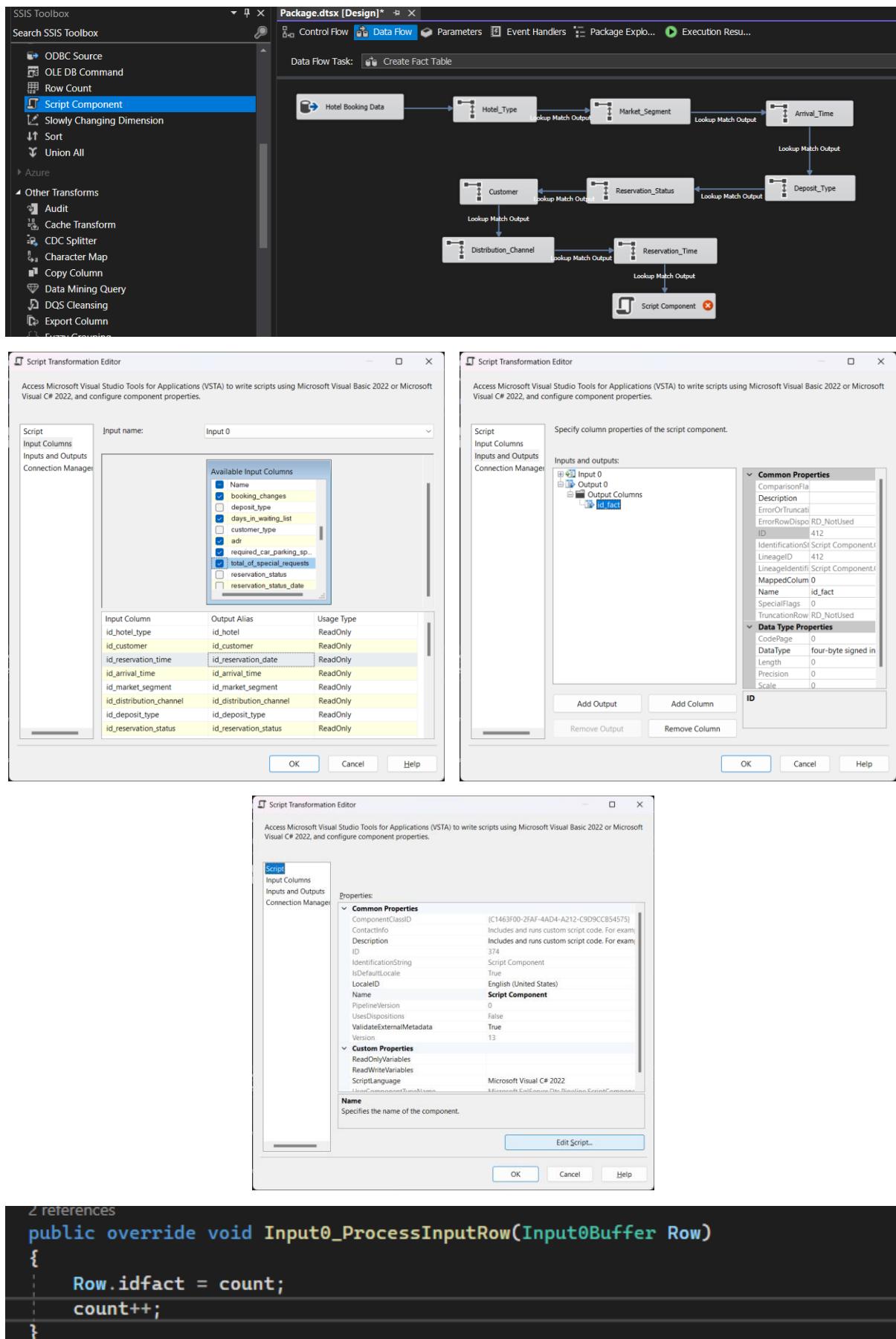


Figure 29-30-31-32-33. Sử dụng Script Component để tạo khóa chính id fact

- **Bước 7:** Chọn công cụ OLE DB Destination để tiến hành tạo bảng trong cơ sở dữ liệu DB_HotelBooking với tên gọi là Fact.

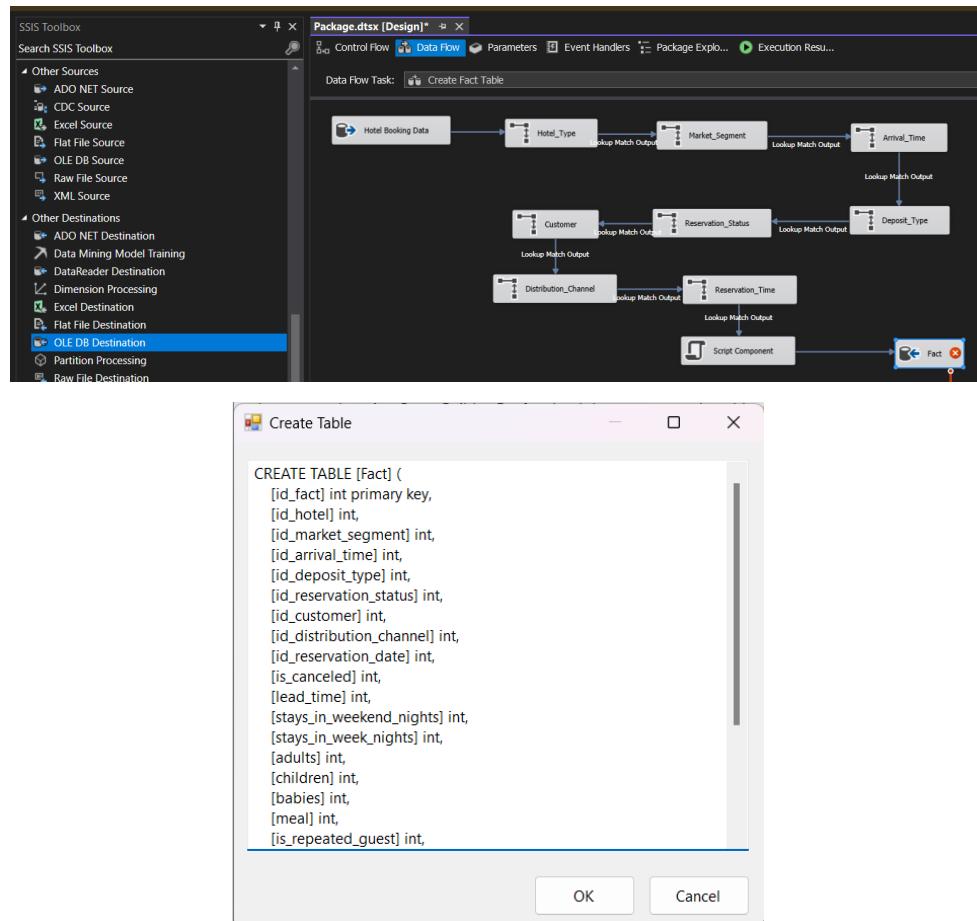


Figure 34-35. Tạo bảng Fact

- **Bước 8:** Nhấn Mappings để kiểm tra kết nối và nhấn Ok để hoàn tất quá trình tạo bảng.

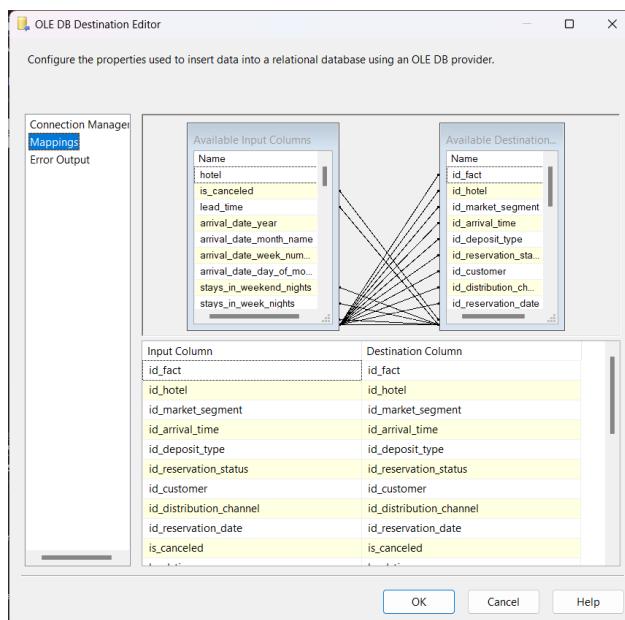


Figure 180. Quá trình Mappings dữ liệu

- **Bước 9:** Chọn Start để bắt đầu quá trình đổ dữ liệu. Khi đổ thành công, ta có kết quả như sau:

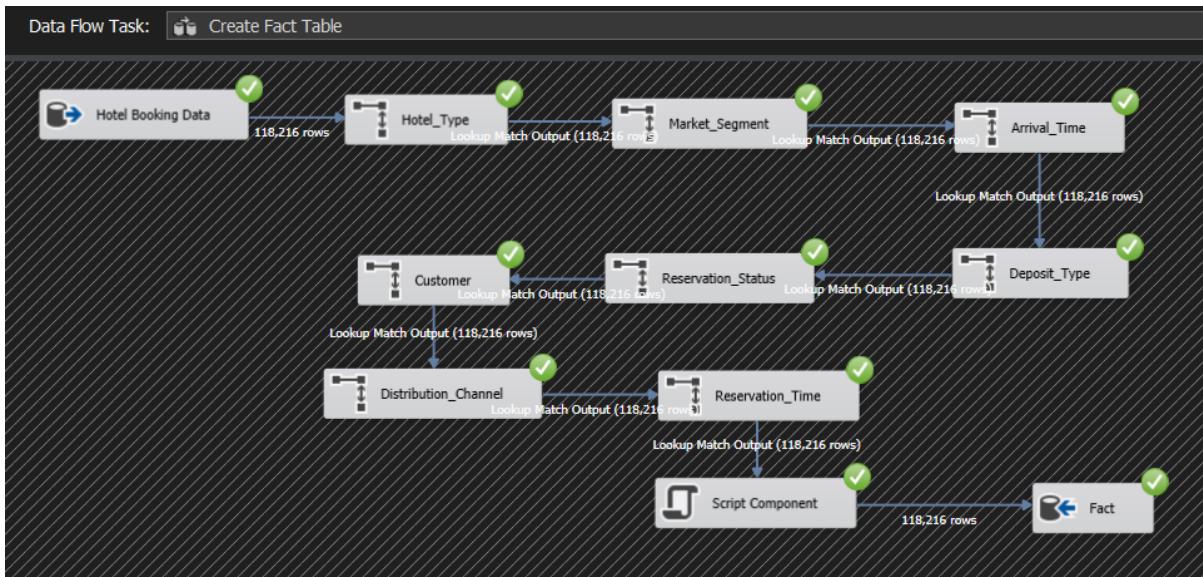


Figure 181. Hoàn thành đổ dữ liệu vào Fact trong kho dữ liệu

- **Bước 10:** Kiểm tra bảng Fact trên SQL Server.

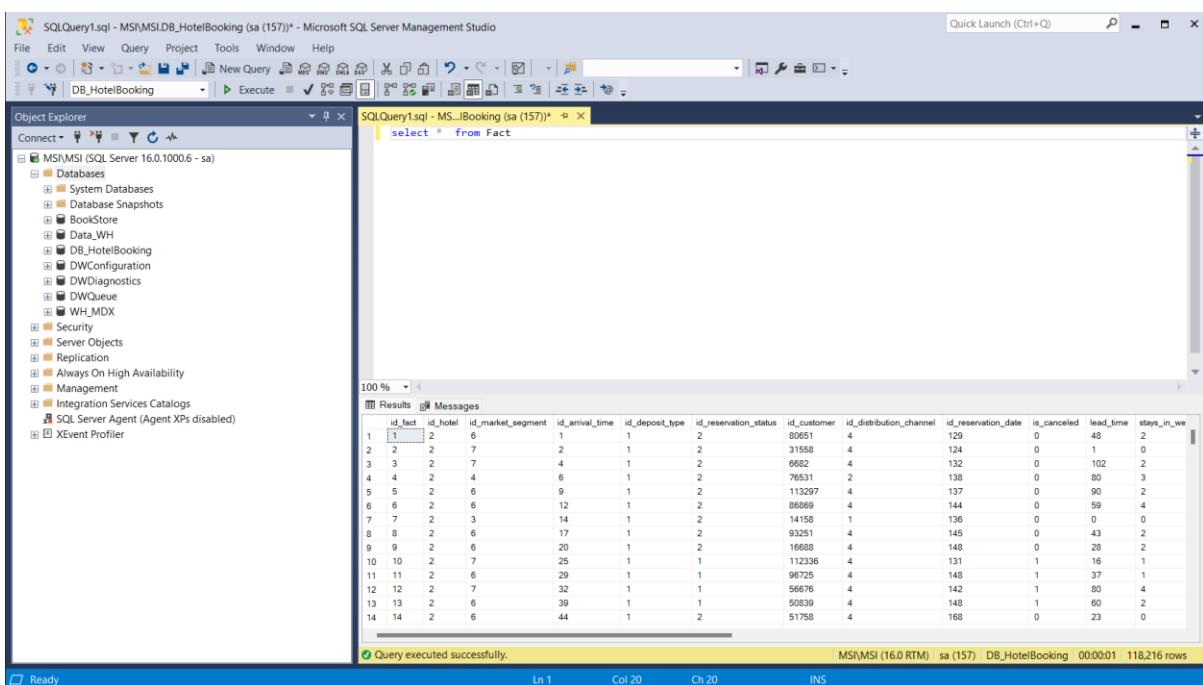


Figure 182. Kiểm tra bảng Fact trong SQL Server

2.6. Tạo Execute SQL Task

Execute SQL Task: Thực thi chạy các câu lệnh SQL hoặc các procedure được lưu trữ từ package. Tác vụ có thể chứa lệnh SQL đơn hoặc nhiều câu lệnh SQL chạy tuần tự.

- **Bước 1:** Kéo thả các Execute SQL Task và lần lượt đặt tên là Delete Hotel Booking Table, Drop Foreign Key, Delete Dim and Fact Table, Create Foreign Key. Bổ sung các câu lệnh SQL để ngăn ngừa việc trùng khi đổ dữ liệu vào những lần sau và tạo kết nối giữa các bảng Dimension và bảng Fact.

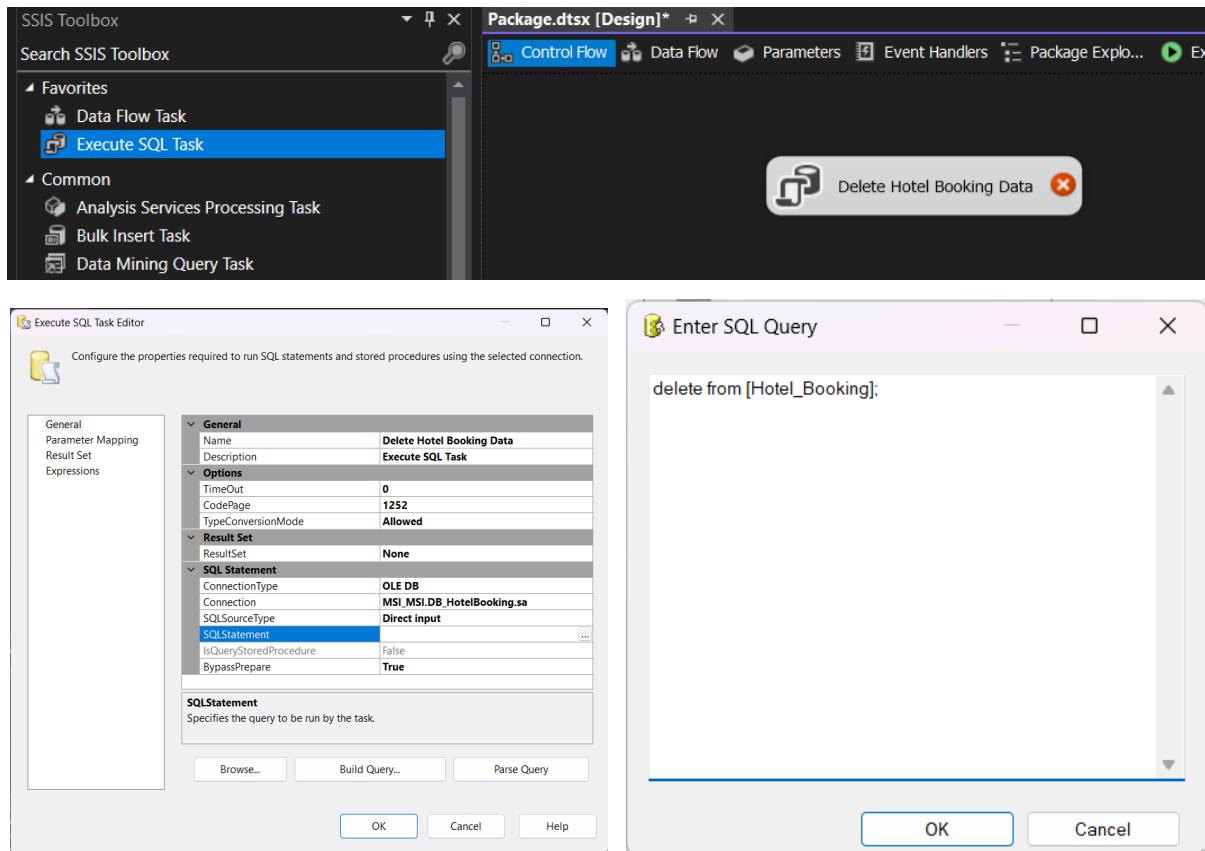
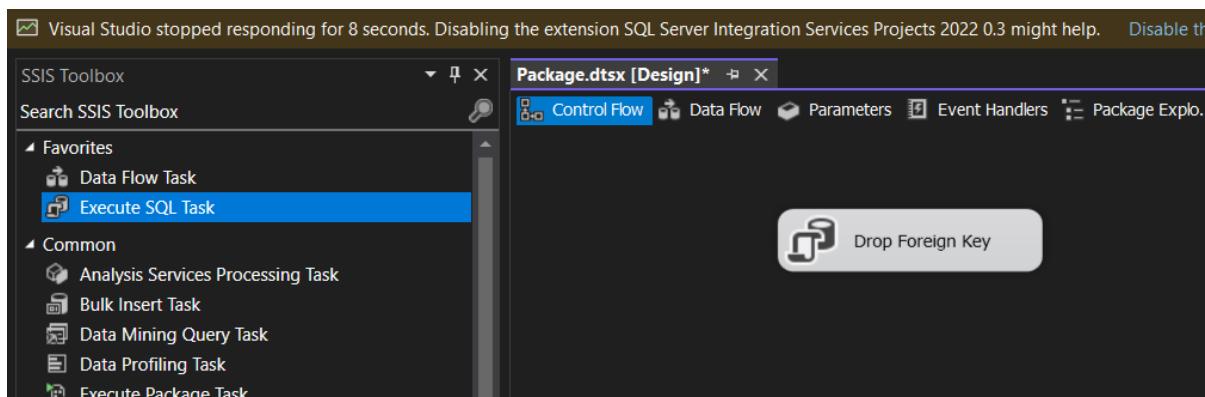


Figure 1-2-3. Viết câu lệnh loại bỏ trùng sau mỗi lần đổ dữ liệu của bảng Hotel_Booking



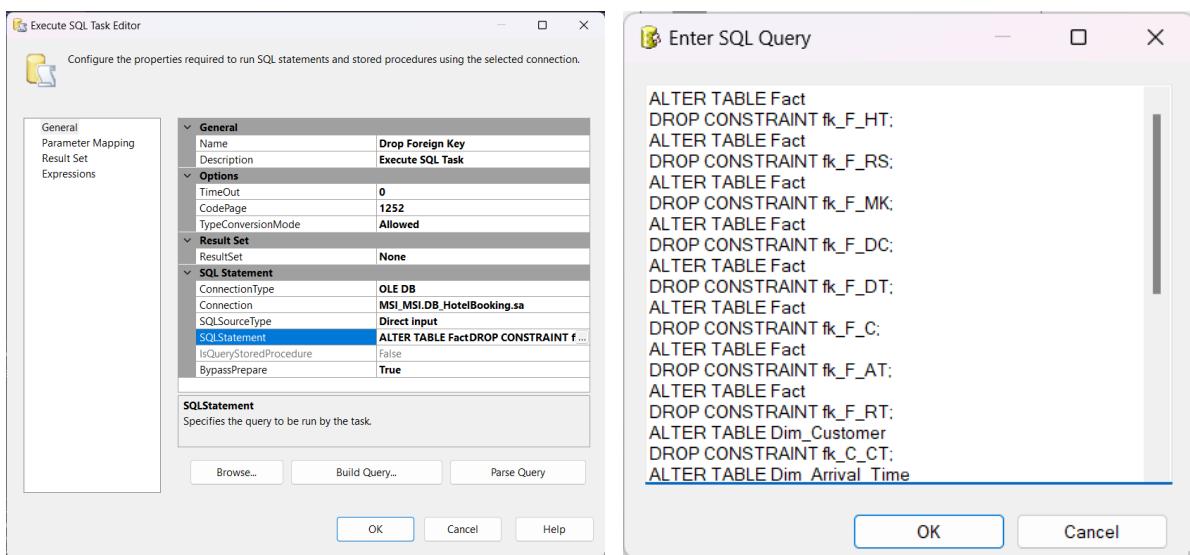


Figure 4-5-6. Viết câu lệnh loại bỏ tất cả các khóa ngoại

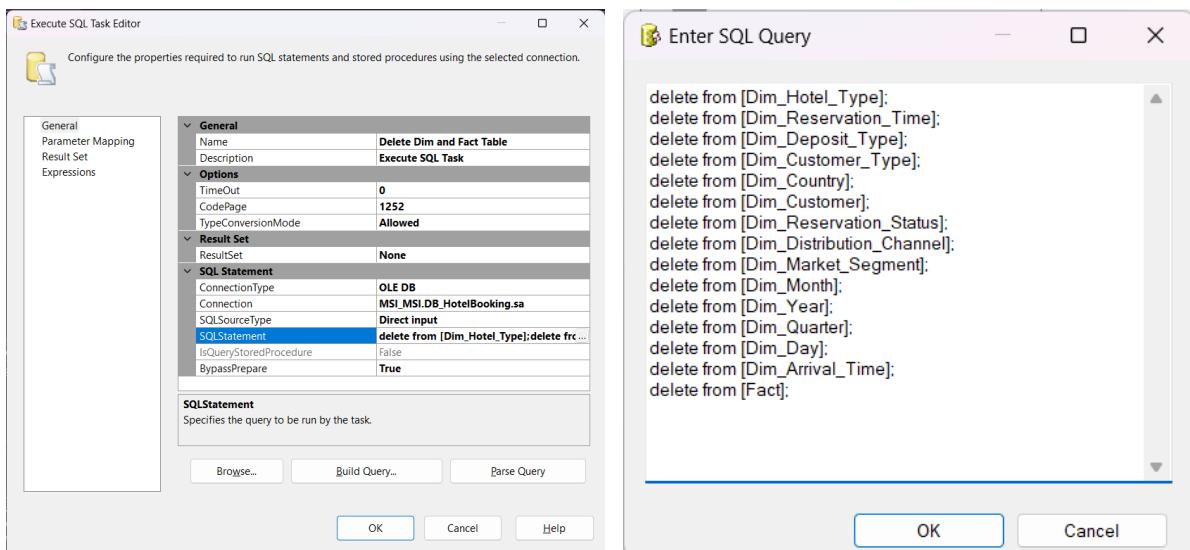
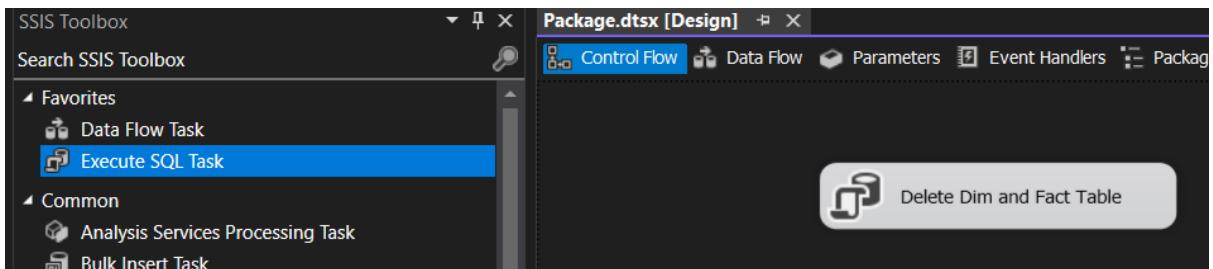


Figure 7-8-9. Viết câu lệnh loại bỏ trùng sau mỗi lần đổ dữ liệu của bảng Dimension and Fact

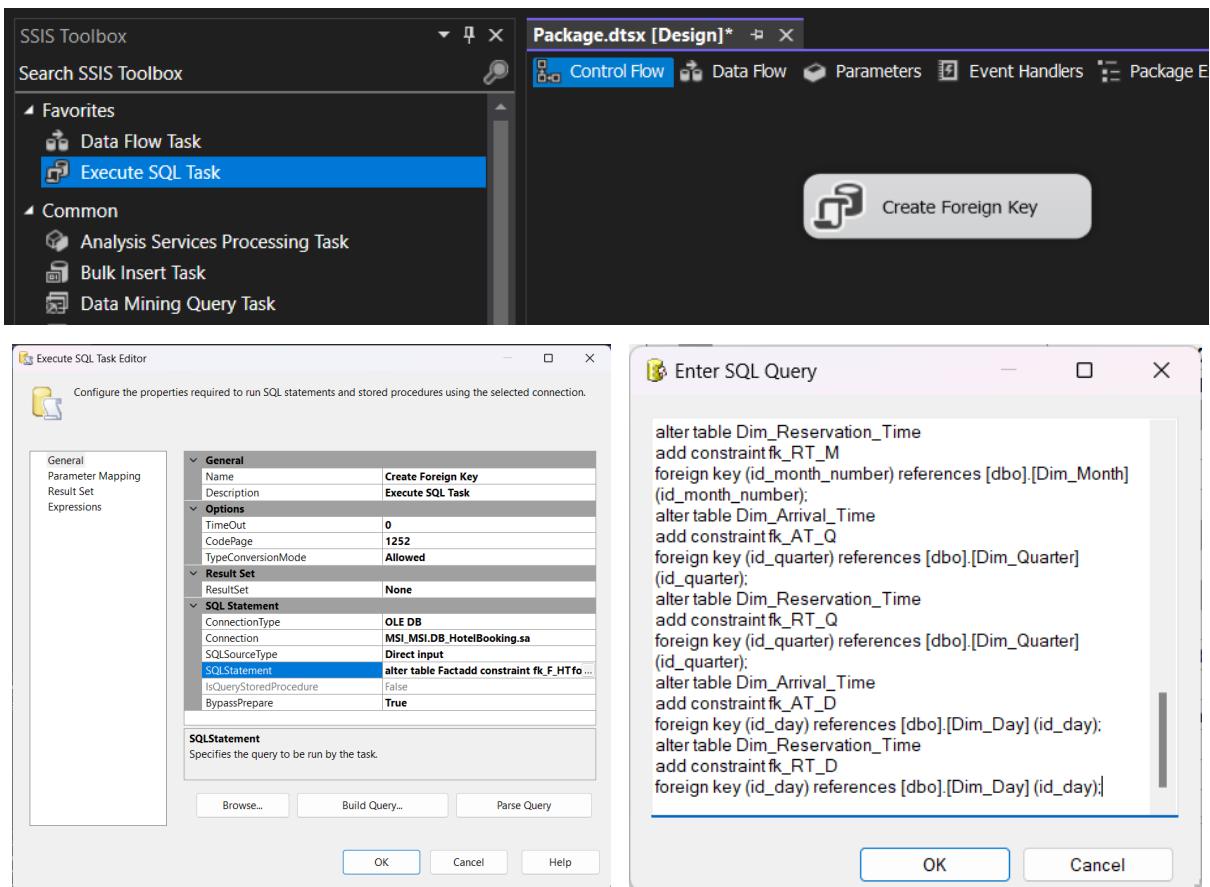


Figure 10-11-12. Viết câu lệnh tạo tất cả các khóa ngoại cần thiết

- **Bước 2:** Chọn Sequence Container và chuyển các Dimension tables vào trong đó.

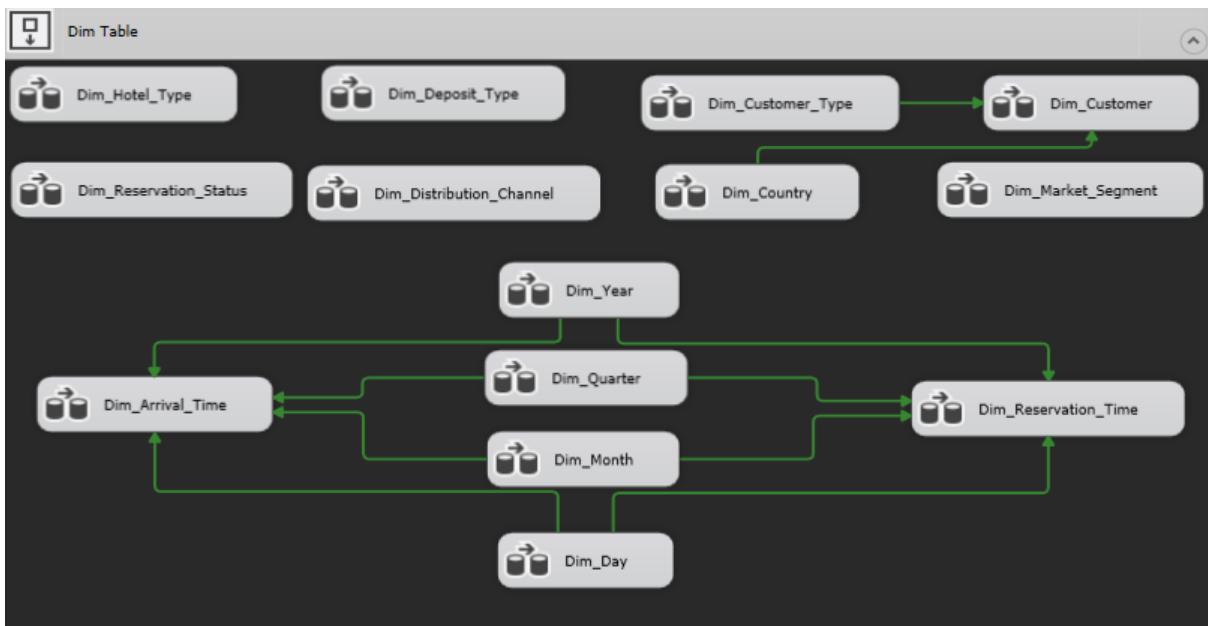


Figure 183. Các bảng Dimensions trong Sequence Container

- **Bước 3:** Tạo đường kết nối giữa các thành phần với nhau theo thứ tự

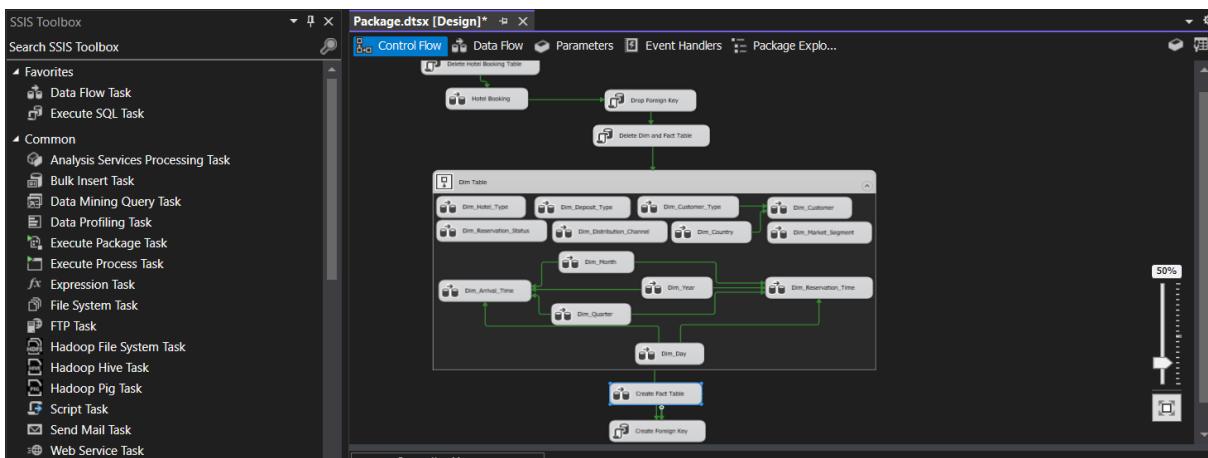


Figure 184. Các thành phần được kết nối với nhau theo thứ tự

- **Bước 4:** Chọn Start để bắt đầu quá trình đổ dữ liệu. Khi đổ thành công, ta có kết quả như sau:

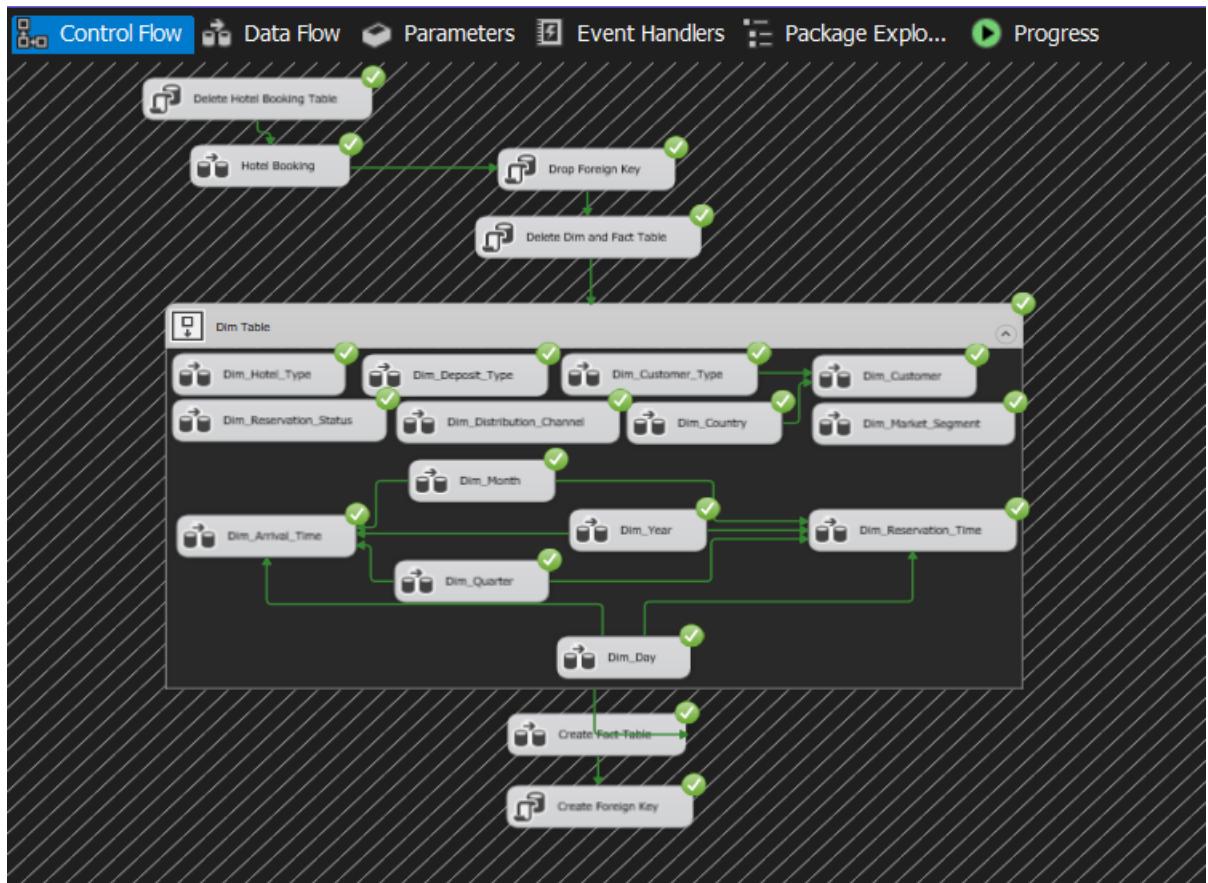


Figure 185. Hoàn thành đổ dữ liệu vào kho dữ liệu

CHƯƠNG 3. PHÂN TÍCH DỮ LIỆU VÀ BÁO CÁO

3.1. Nội dung 15 câu truy vấn

3.1.1. Báo cáo tình hình kinh doanh

- Thống kê tổng số lượng khách hàng đặt phòng, tổng doanh thu của từng loại khách sạn theo từng quý của năm.
- Thống kê doanh thu theo phân khúc khách hàng (Market_segment) theo từng tháng của năm.
- Thống kê doanh thu theo kênh phân phối theo từng tháng của năm.
- Thống kê top 10 quốc gia có doanh thu cao nhất trong năm 2015 và 2016, sắp xếp các giá trị theo thứ tự giảm dần.
- Thống kê tổng thời gian chờ của khách hàng (lead_time) theo từng Quốc gia theo từng tháng của năm.
- Thống kê các quốc gia tạo ra tổng doanh thu lớn hơn 50000.
- Thống kê tổng thời gian chờ của từng phân khúc thị trường theo từng tháng, quý, năm.
- Thống kê số lượng đặt phòng của khách hàng đến từ Bồ Đào Nha (PRT) thuộc từng phân khúc thị trường theo từng quý của từng năm
- Thống kê doanh thu theo từng kênh phân phối của từng phân khúc thị trường của từng quốc gia.

3.1.2. Báo cáo và phân tích dữ liệu khách hàng

- Thống kê có bao nhiêu khách hàng là người lớn, bao nhiêu khách hàng là trẻ em và bao nhiêu khách hàng là em bé trong từng tháng của năm.
- Thống kê thông tin top 50 khách hàng có doanh số cao nhất trong năm trước.
- Thống kê tên top 5 khách hàng có chi tiêu cao nhất theo từng tháng của năm 2016.
- Thống kê email top 3 khách hàng có doanh số cao nhất theo quốc gia và theo năm.
- Thống kê top 10 thông tin khách hàng kèm chi tiêu của họ có thời gian chờ lâu nhất theo từng quốc gia theo năm 2016.
- Thống kê thông tin và doanh thu khách hàng đặt phòng với hình thức không đặt cọc trước và thuộc phân khúc thị trường đặt phòng trực tiếp.

3.2. Phân tích dữ liệu trong kho dữ liệu (SSAS)

3.2.1. Tạo project SSAS trong Visual Studio

- **Bước 1:** Mở phần mềm Visual Studio → Create a new project.

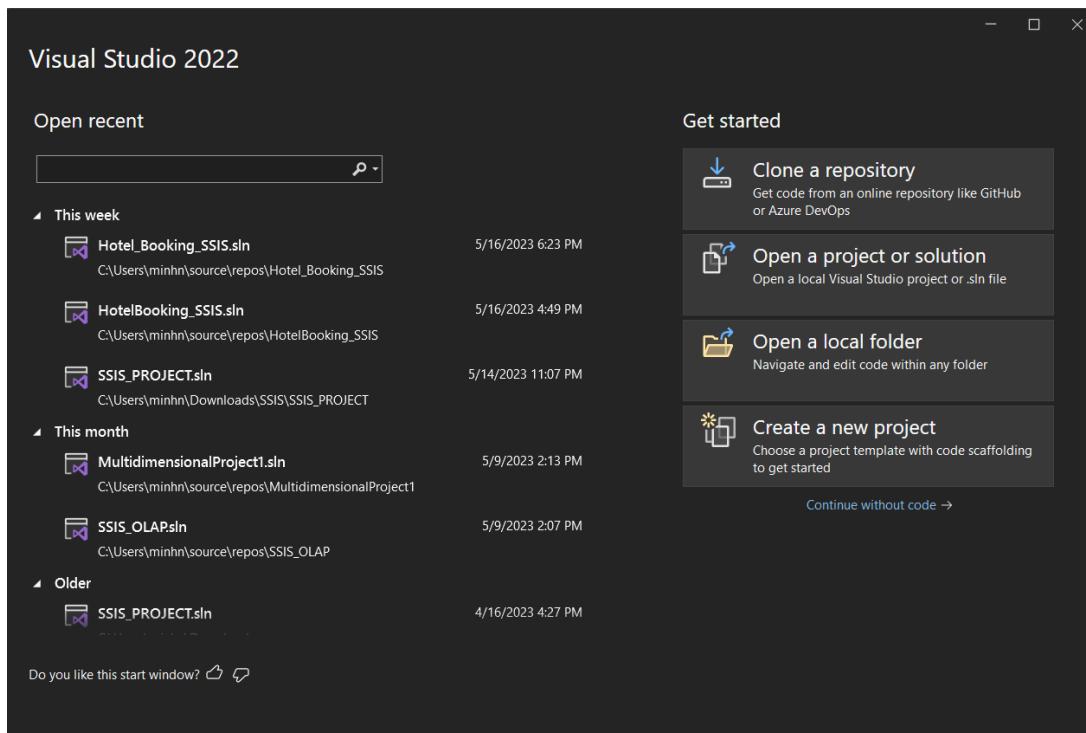


Figure 186. Giao diện khởi động Visual Studio 2022

- **Bước 2:** Tìm project template Analysis Services Multidimensional project → Next.

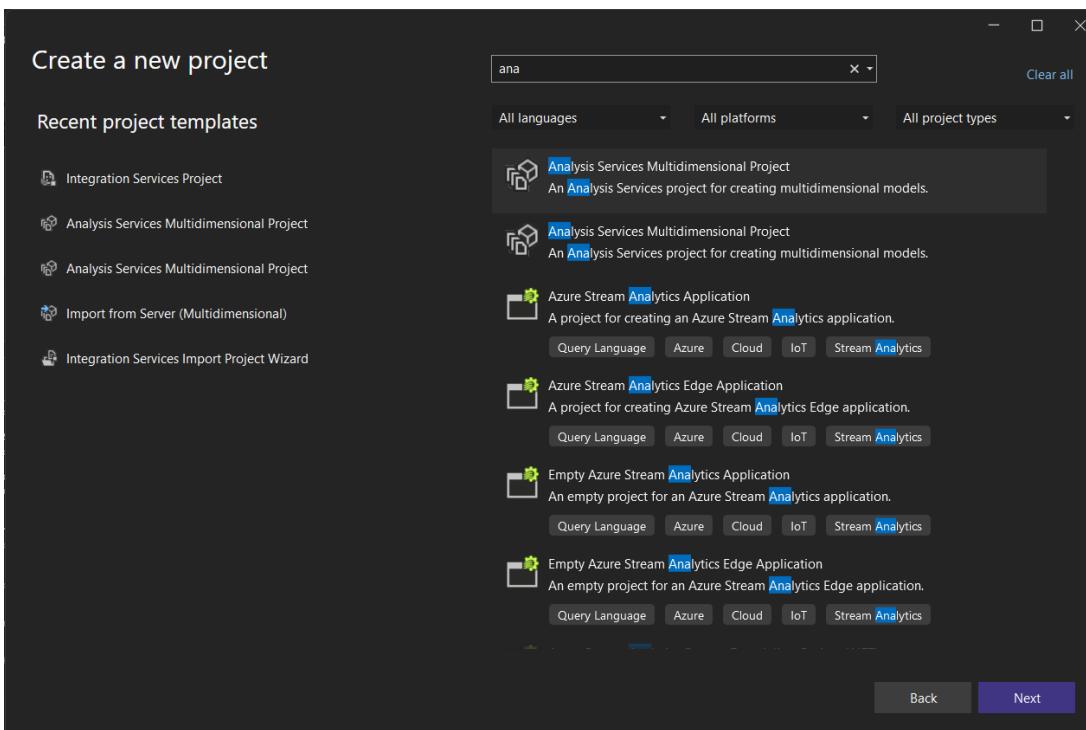


Figure 187. Tìm kiếm project template SSAS của Visual Studio

- **Bước 3:** Chọn đường dẫn để lưu project, đặt tên và nhấn OK.

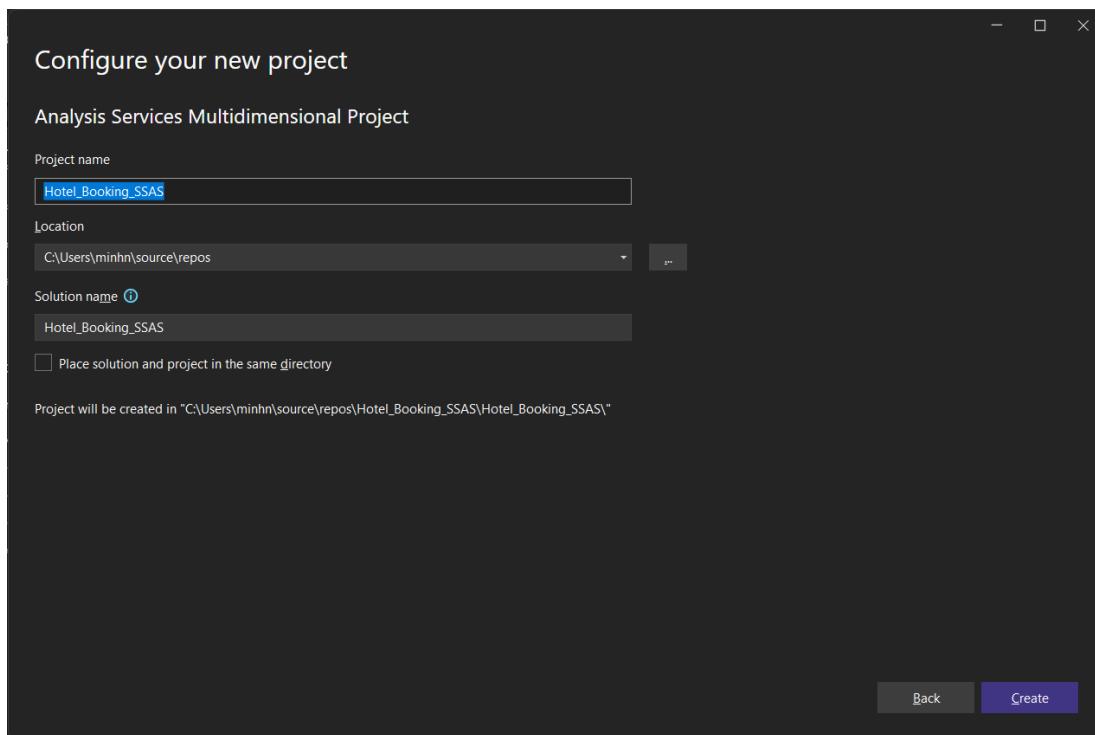


Figure 188. Giao diện cấu hình file của SSAS trong Visual Studio

- **Bước 4:** Giao diện quá trình SSAS của công cụ BI hiện ra.

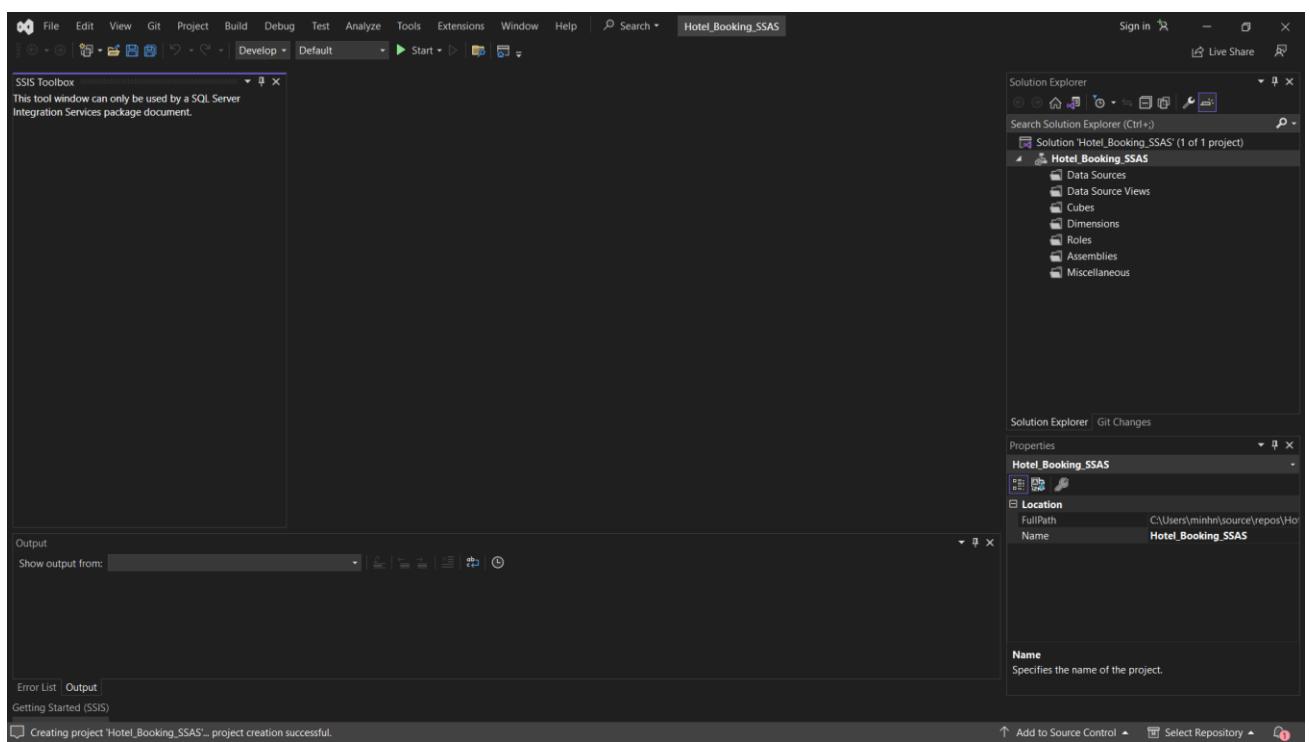


Figure 189. Giao diện công cụ BI quá trình SSAS

3.2.2. Xác định dữ liệu nguồn (Data sources)

- **Bước 1:** Bên gốc phải phần Solution Explorer nhấn chuột phải vào Data Source và chọn New Data Source.

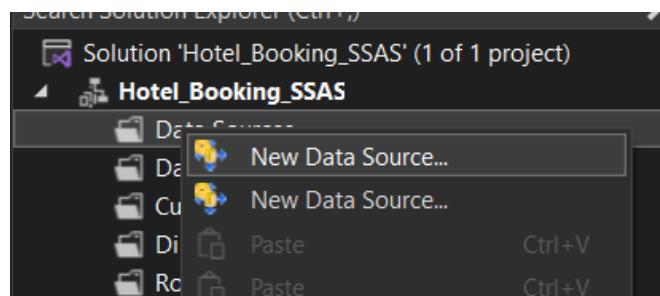


Figure 190. Tạo một Data Suorce cho quá trình SSAS

- **Bước 2:** Chọn kết nối đã tạo từ quá trình SSIS rồi tiếp tục nhấn Next.

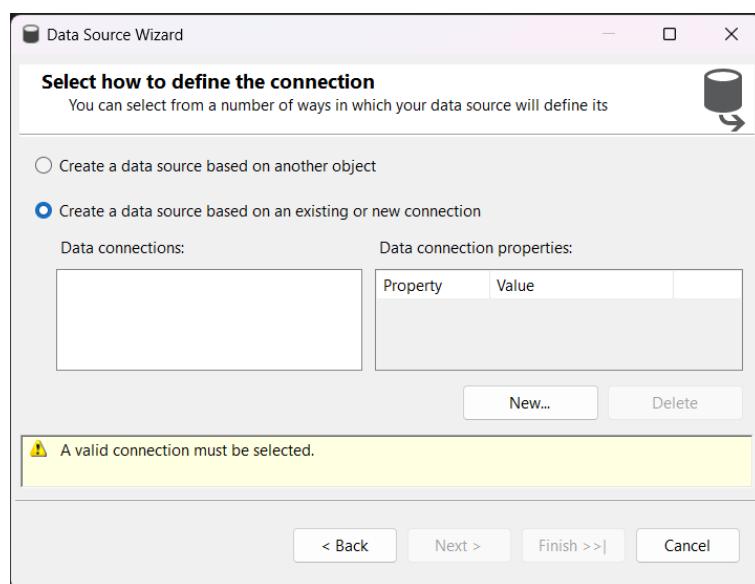


Figure 191. Chọn kết nối với Data Source đã có sẵn hoặc tạo kết nối mới

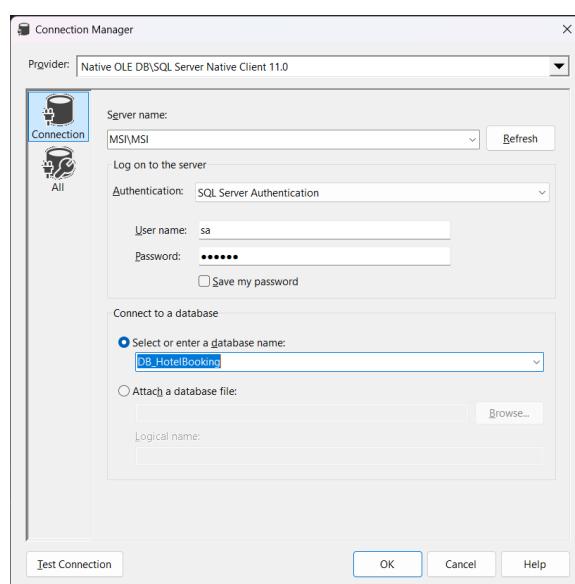


Figure 192. Tạo kết nối mới khi project chạy lần đầu

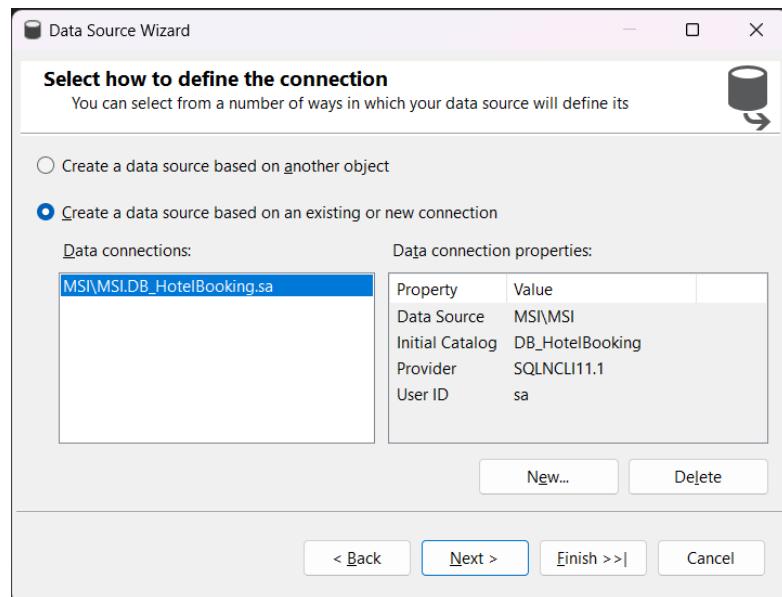


Figure 193. Chọn kết nối với Data Source

- **Bước 3:** Tại trang Impersonation Information, định nghĩa security credentials cho Analysis Services để kết nối tới data source. Chọn Use a specific Windows user name and password. Sau đó điền User name và Password. Nhấn Next để tiếp tục.

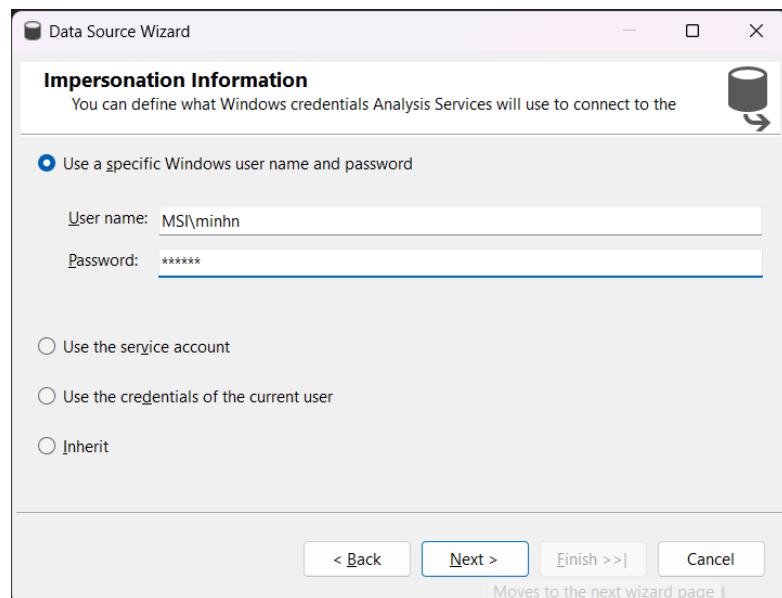


Figure 194. Tùy chọn tài khoản phù hợp

- **Bước 4:** Nhấn Finish để kết thúc quá trình xác định dữ liệu nguồn.

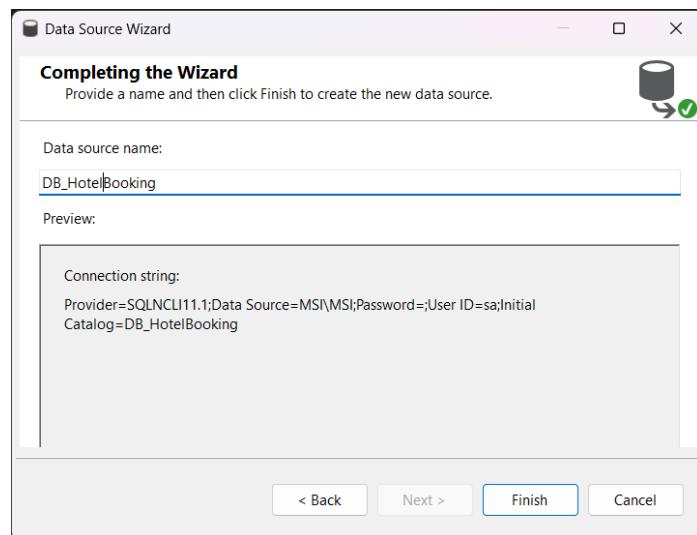


Figure 195. Hoàn thành tạo nguồn dữ liệu

3.2.3. Xác định khung nhìn dữ liệu nguồn (Data Source View)

- **Bước 1:** Bên gốc phải phần Solution Explorer nhấn chuột phải vào Data Source View và chọn New Data Source View.

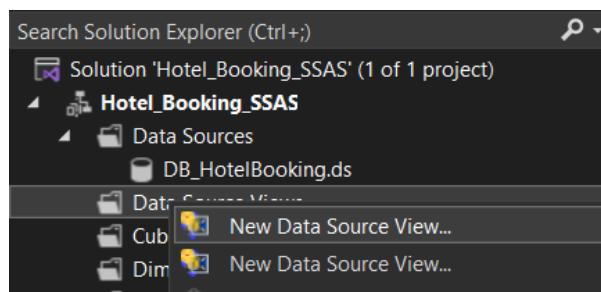


Figure 196. Tạo Data Source View

- **Bước 2:** Chọn kho dữ liệu DB_HotelBooking rồi nhấn Next để tiếp tục.

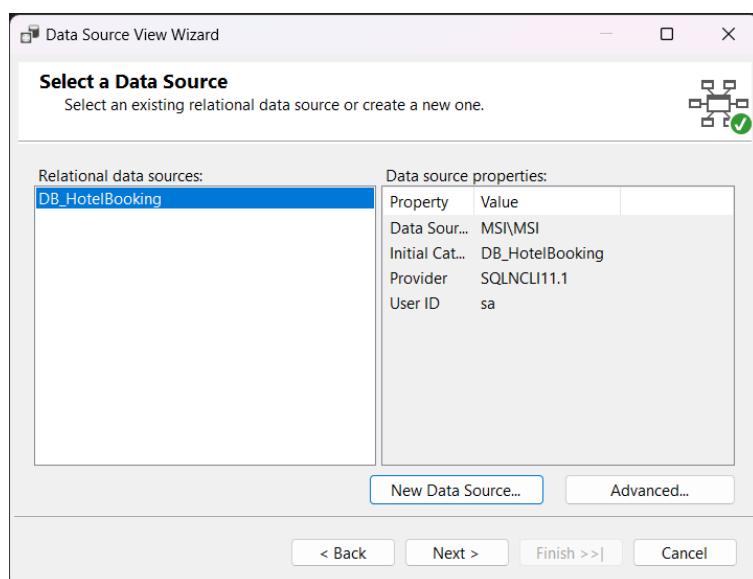


Figure 197. Chọn kho dữ liệu

- **Bước 3:** Chọn bảng FACT và DIM cho quá trình phân tích sau đó nhấn Next.

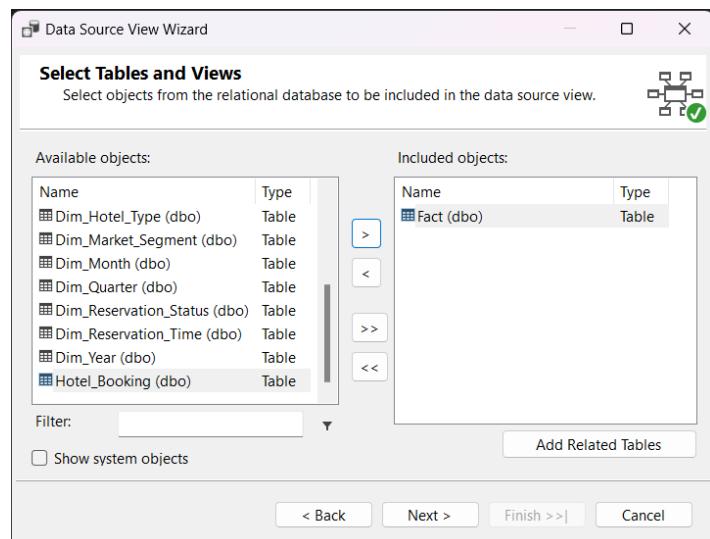


Figure 198. Chọn bảng dữ liệu Fact

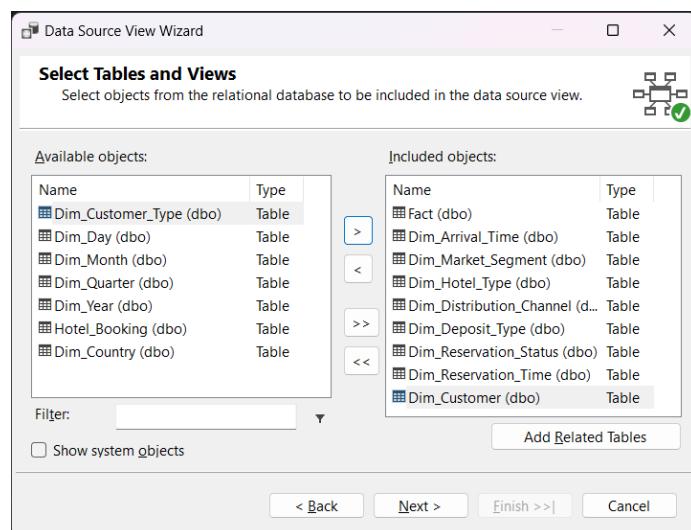


Figure 199. Chọn các bảng Dimension liên quan đến Fact

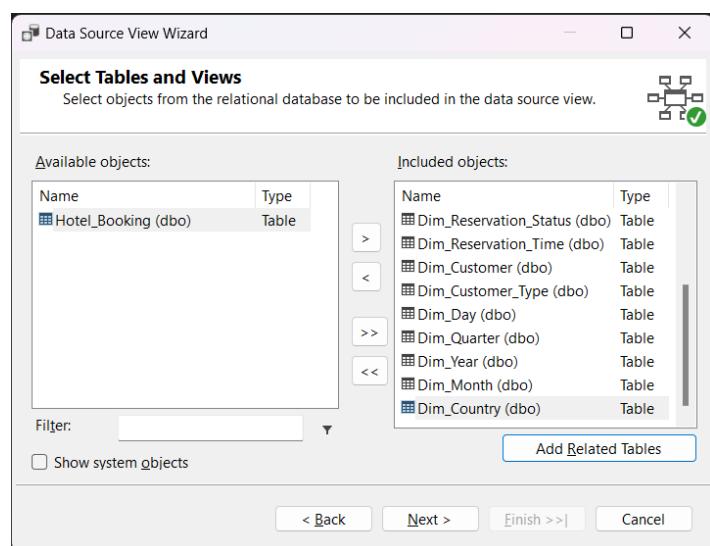


Figure 200. Chọn các bảng Dim có liên quan đến Dim_Customer, Dim_Arrival_Time và Dim_Reservation_time

- **Bước 4:** Nhấn Next.

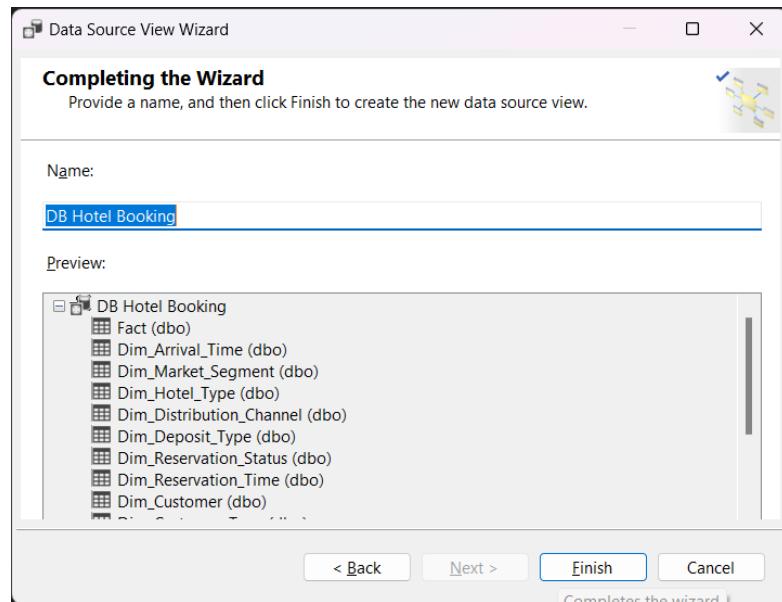


Figure 201. Các bảng dữ liệu được tạo

- **Bước 5:** Nhấn Finish để kết thúc quá trình tạo Data Source View và kết quả sẽ được hiển thị như hình.

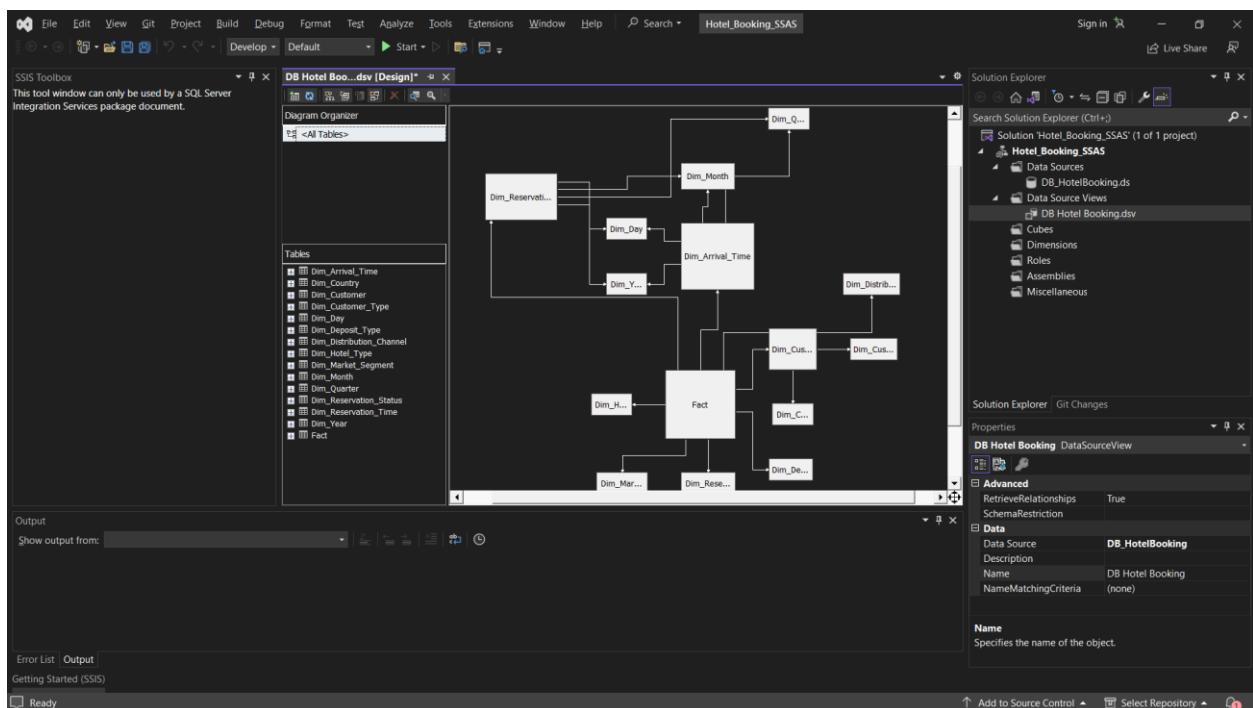


Figure 202. Hoàn tất tạo Data Source View

3.2.4. Xác định cube và tạo Measures (Cubes)

- **Bước 1:** Bên gốc phải phần Solution Explorer nhấp chuột phải vào Cube và chọn New Cube.

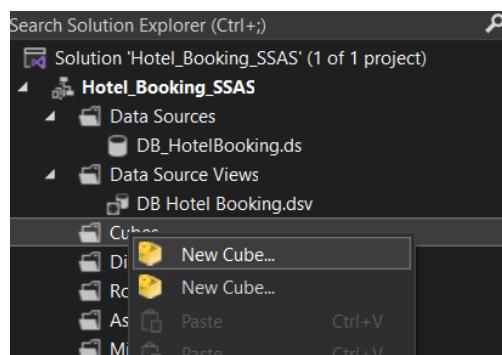


Figure 203. Bắt đầu quá trình tạo Cube

- **Bước 2:** Trong cửa sổ Select Createion Method chọn use existing tables sau đó nhấn Next.

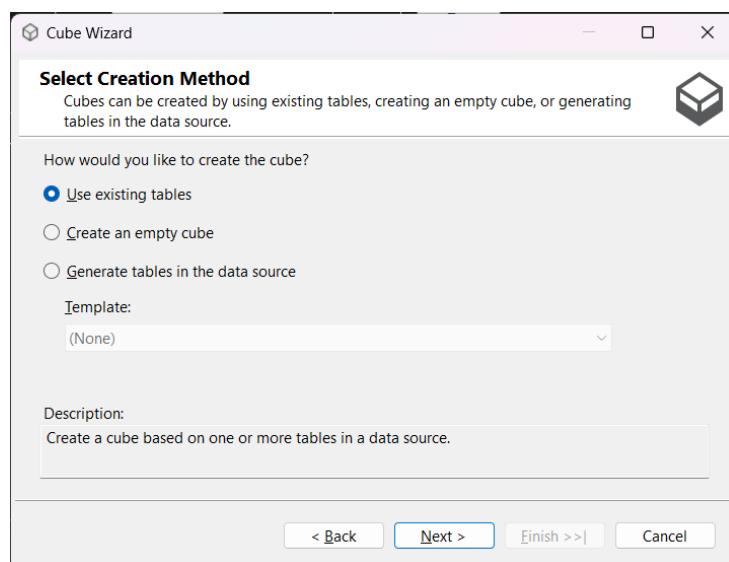


Figure 204. Tùy chọn tạo Cube

- **Bước 3:** Chọn bảng Fact chứa các thuộc tính độ đo để phân chia các measure group.

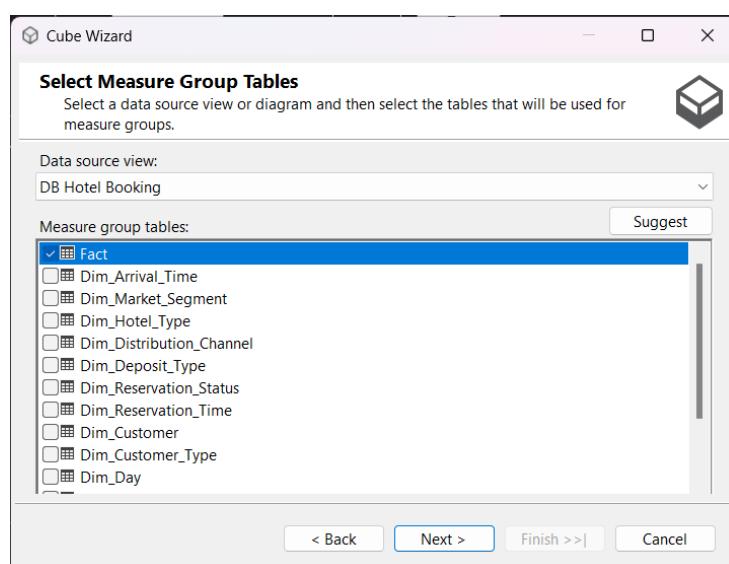


Figure 205. Chọn Fact làm Measure Group cho Cube

- **Bước 4:** Chọn những thuộc tính định lượng để xuất. Nhấn Next để tiếp tục.

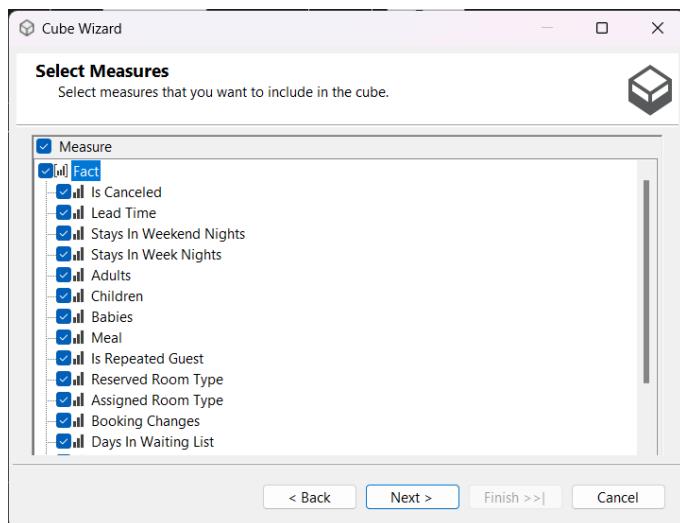


Figure 206. Tùy chọn các thuộc tính định lượng

- **Bước 5:** Chọn danh sách Dimension nhấn Next để tiếp tục.

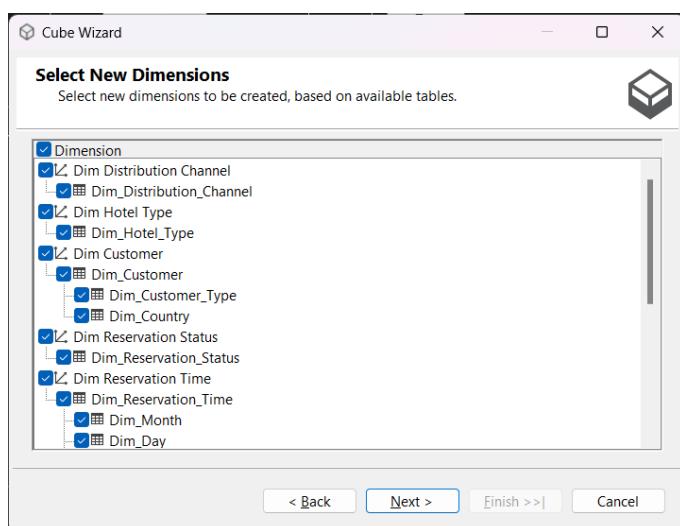


Figure 207. Chọn các bảng Dimension cho Cube

- **Bước 6:** Nhấn Finish để hoàn thành quá trình tạo cube.

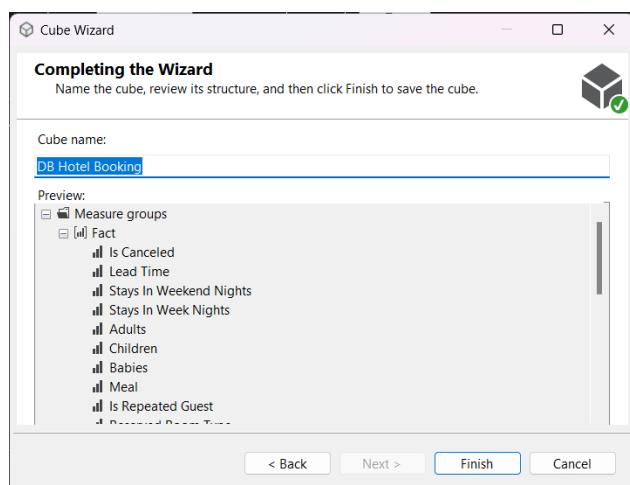


Figure 208. Xem lại Measure Groups và các Dimension

- **Bước 7:** Nhấn Finish để kết thúc quá trình tạo Cube và kết quả sẽ được hiển thị như hình.

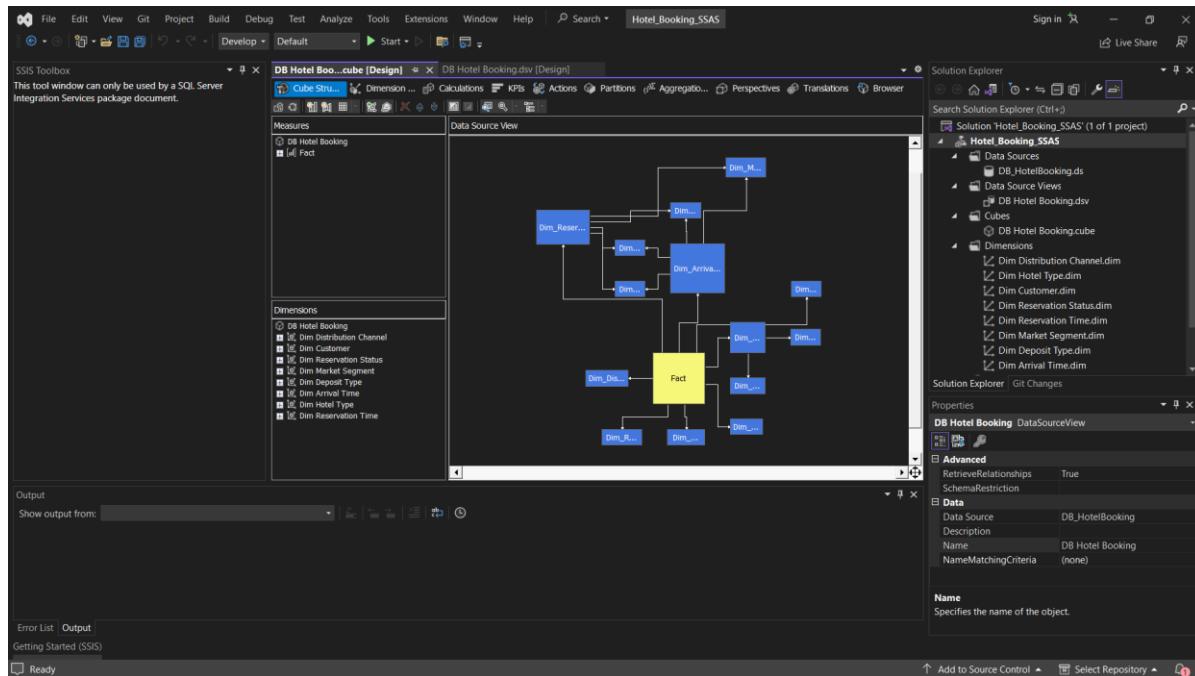


Figure 209. Hoàn tất quá trình tạo Cube

3.2.5. Xác định các thuộc tính của bảng Dimensions

3.2.5.1. Định nghĩa các thuộc tính (Attributes) của bảng Dimension

Bảng Dim_Reservation_Status

- **Bước 1:** Ở Solution Explorer, double-click vào Dim Reservation Status.dim để tiến hành thêm các thuộc tính.

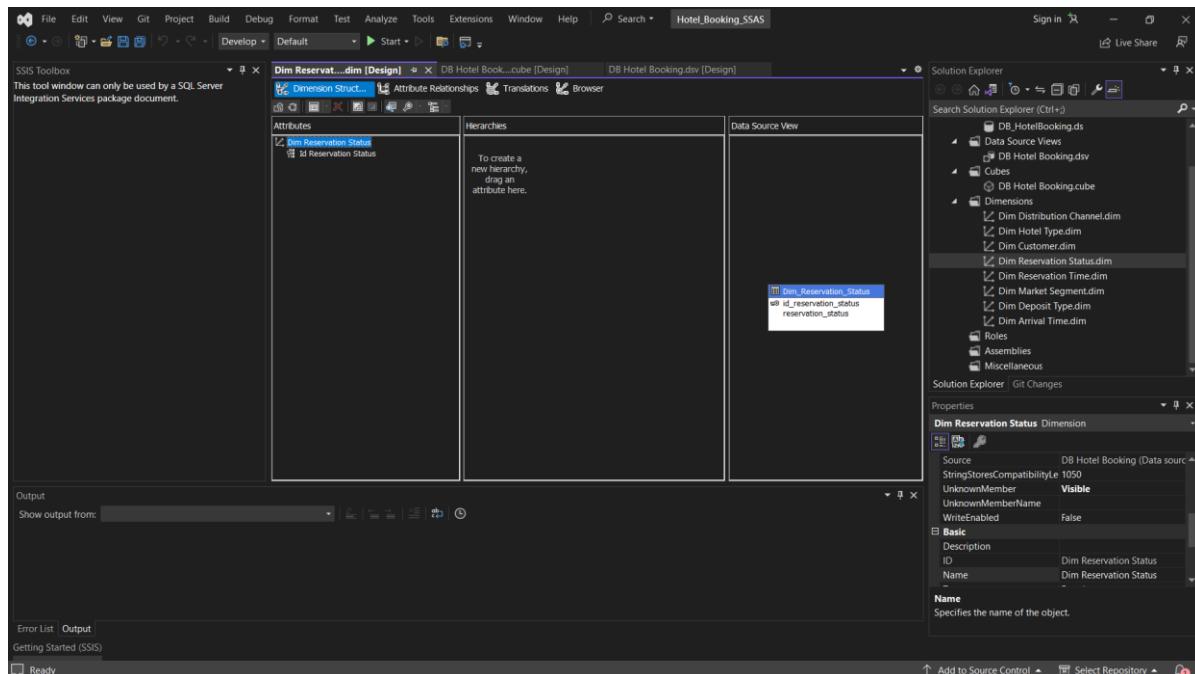


Figure 210. Bảng thuộc tính Dim_Reservation_Status

- **Bước 2:** Kéo các thuộc tính từ Data Source View sang Attributes.

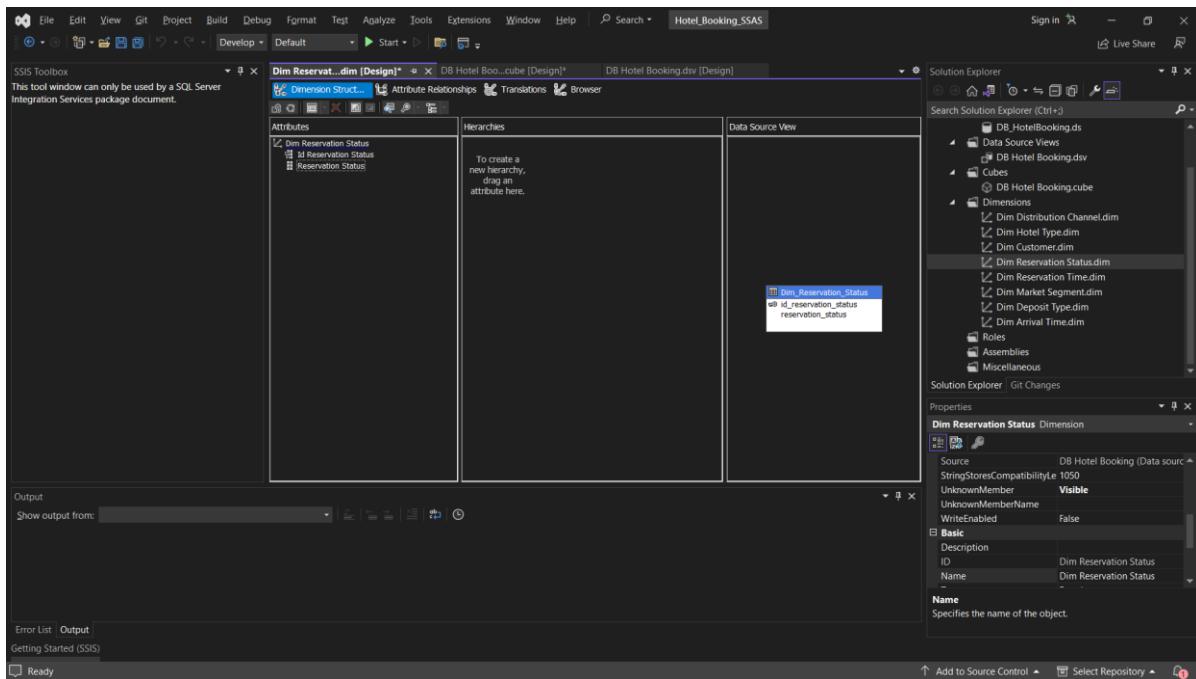


Figure 211. Kết quả sau khi kéo thả các thuộc tính

Bảng Dim_Distribution_Channel

- **Bước 1:** Ở Solution Explorer, double-click vào Dim Distribution Channel.dim để tiến hành thêm các thuộc tính.

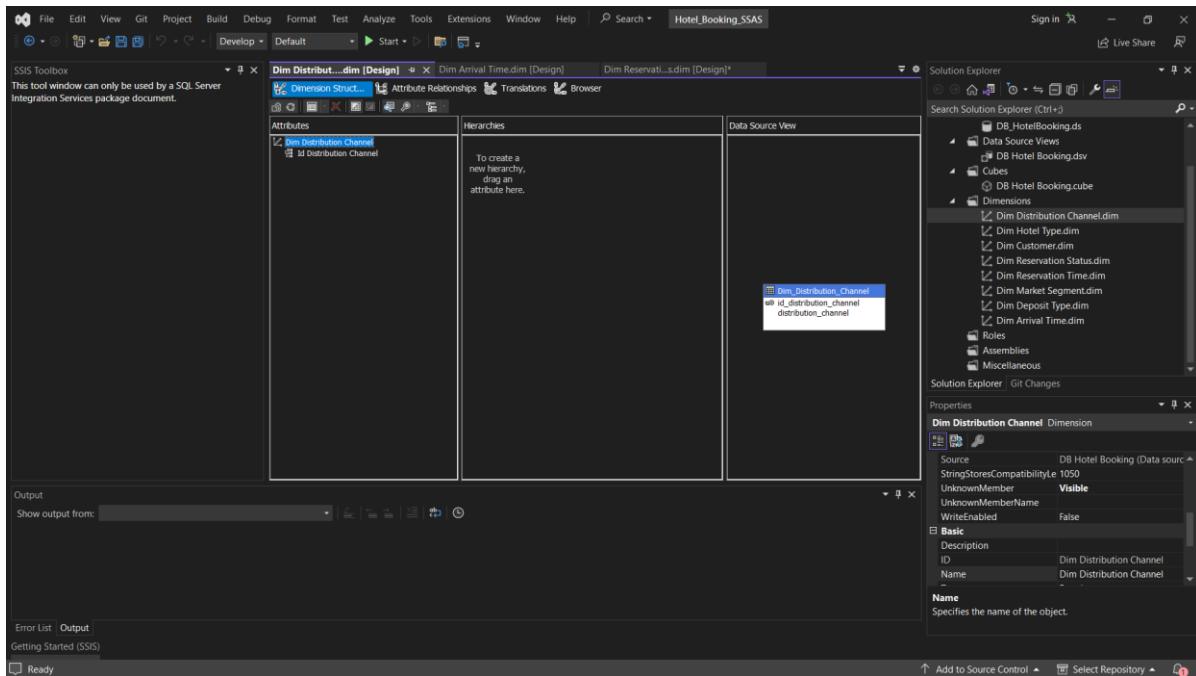


Figure 212. Bảng thuộc tính Dim_Distribution_Channel

- **Bước 2:** Kéo các thuộc tính sử dụng từ Data Source View sang Attributes.

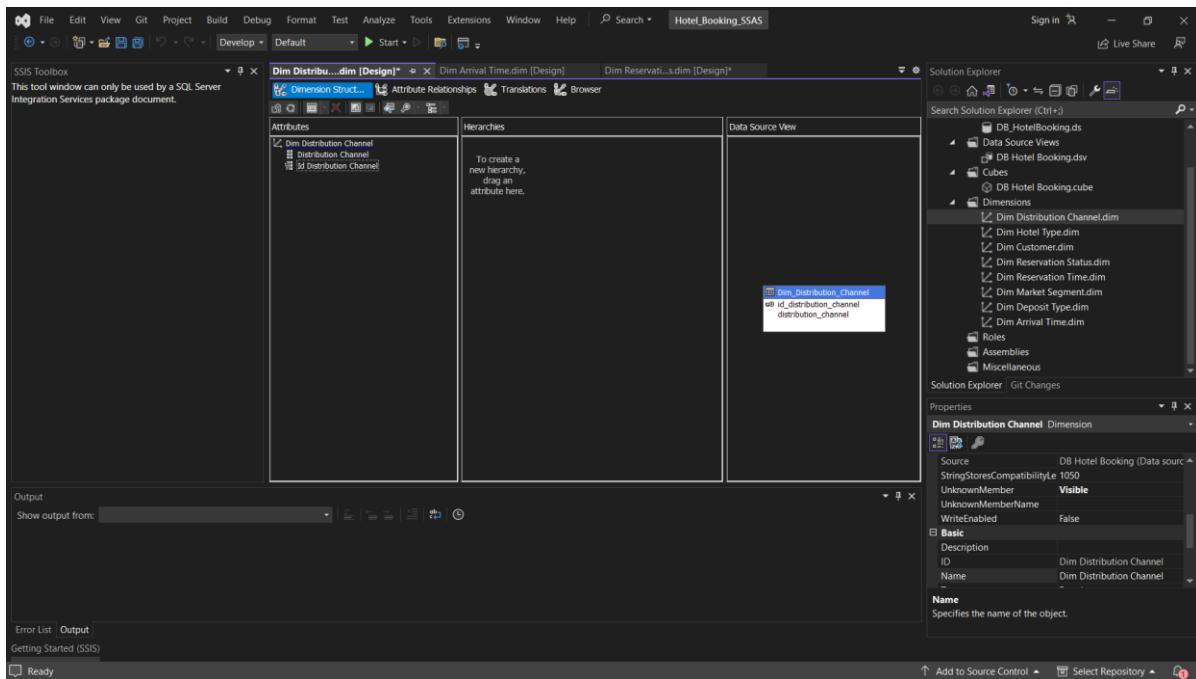


Figure 213. Kết quả sau khi kéo thả các thuộc tính

Bảng Dim_Hotel_Type

- **Bước 1:** Ở Solution Explorer, double-click vào Dim Hotel Type.dim để tiến hành thêm các thuộc tính.

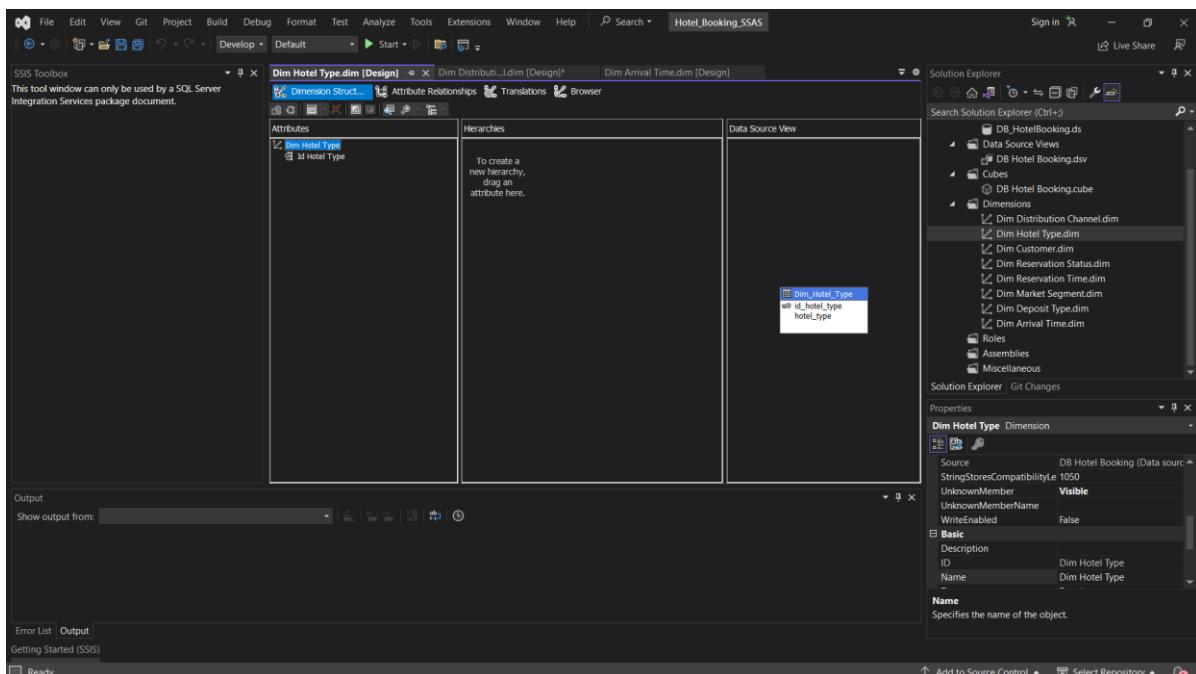


Figure 214. Bảng thuộc tính Dim_Hotel_Type

- **Bước 2:** Kéo các thuộc tính sử dụng từ Data Source View sang Attributes.

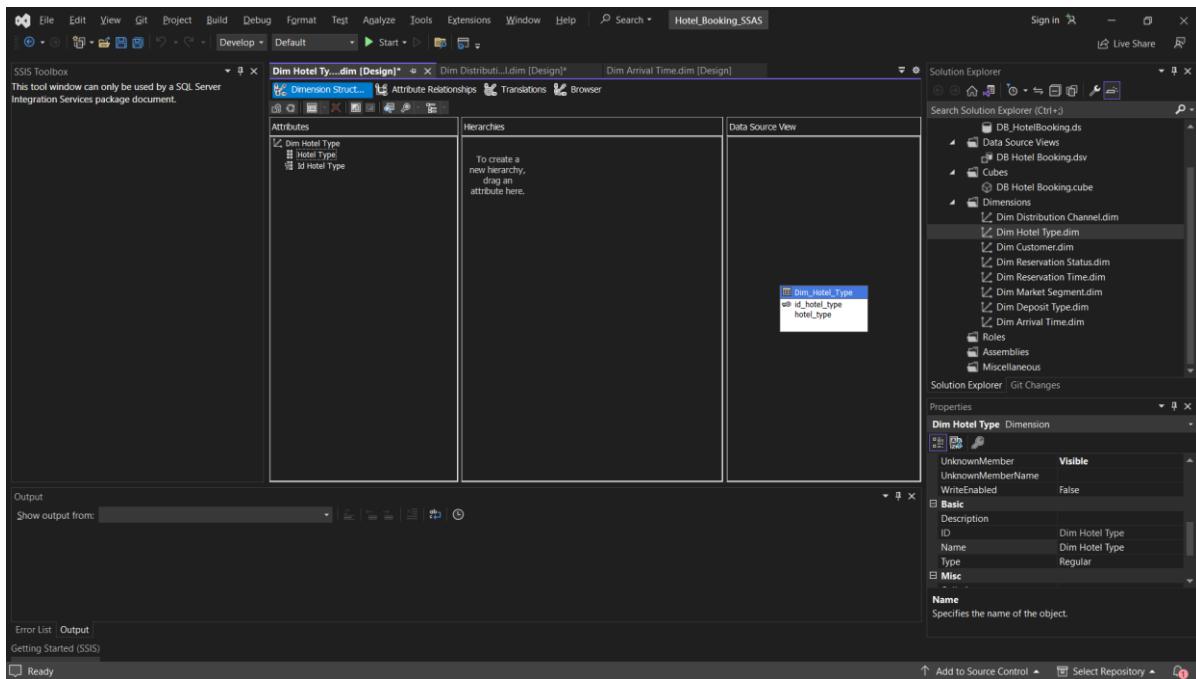


Figure 215. Kết quả sau khi kéo thả các thuộc tính

Bảng Dim_Customer, Dim_Customer_Type và Dim_Country

- **Bước 1:** Ở Solution Explorer, double-click vào Dim Customer.dim để tiến hành thêm các thuộc tính.

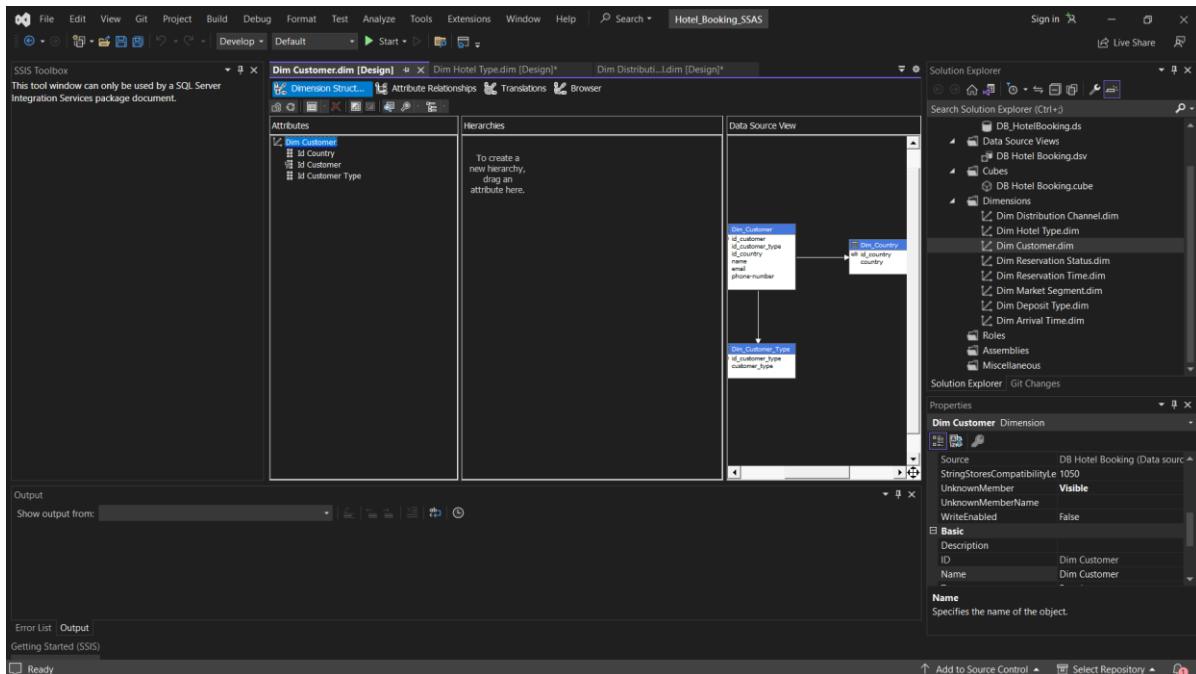


Figure 216. Bảng thuộc tính Dim_Customer, Dim_Customer_Type và Dim_Country

- **Bước 2:** Kéo các thuộc tính sử dụng từ Data Source View sang Attributes.

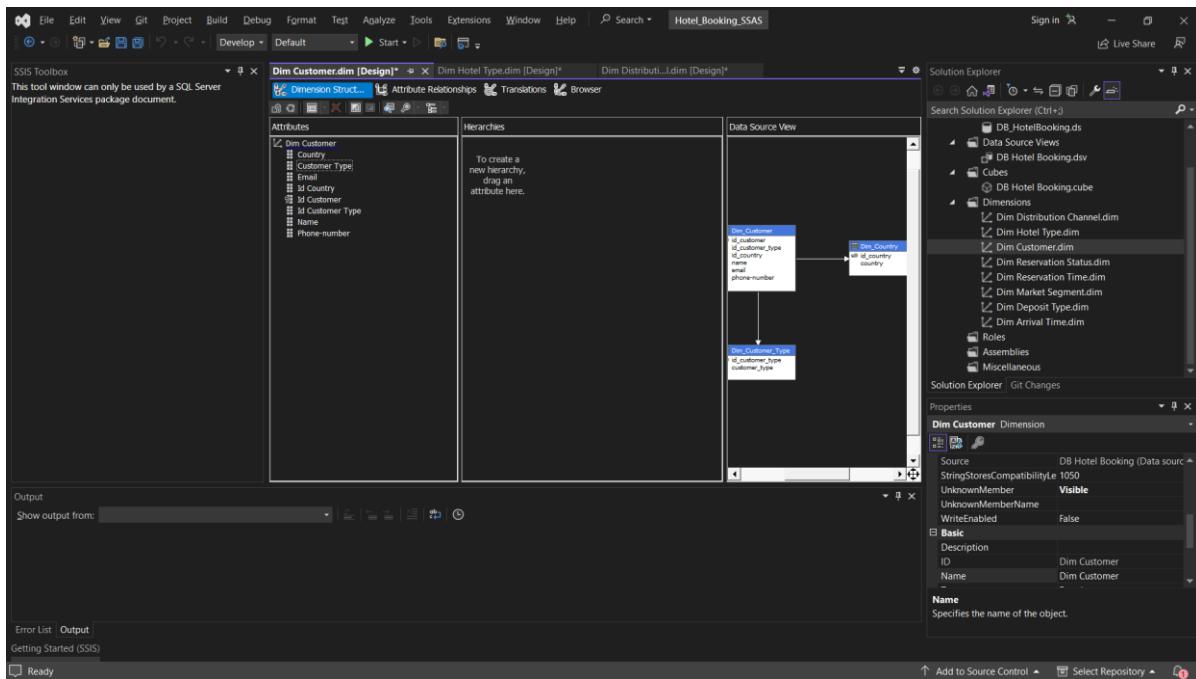


Figure 217. Kết quả sau khi kéo thả các thuộc tính

Bảng Dim_Reservation_Time, Dim_Year, Dim_Quarter, Dim_Month và Dim_Day

- **Bước 1:** Ở Solution Explorer, double-click vào Dim Reservation Time.dim để tiến hành thêm các thuộc tính.

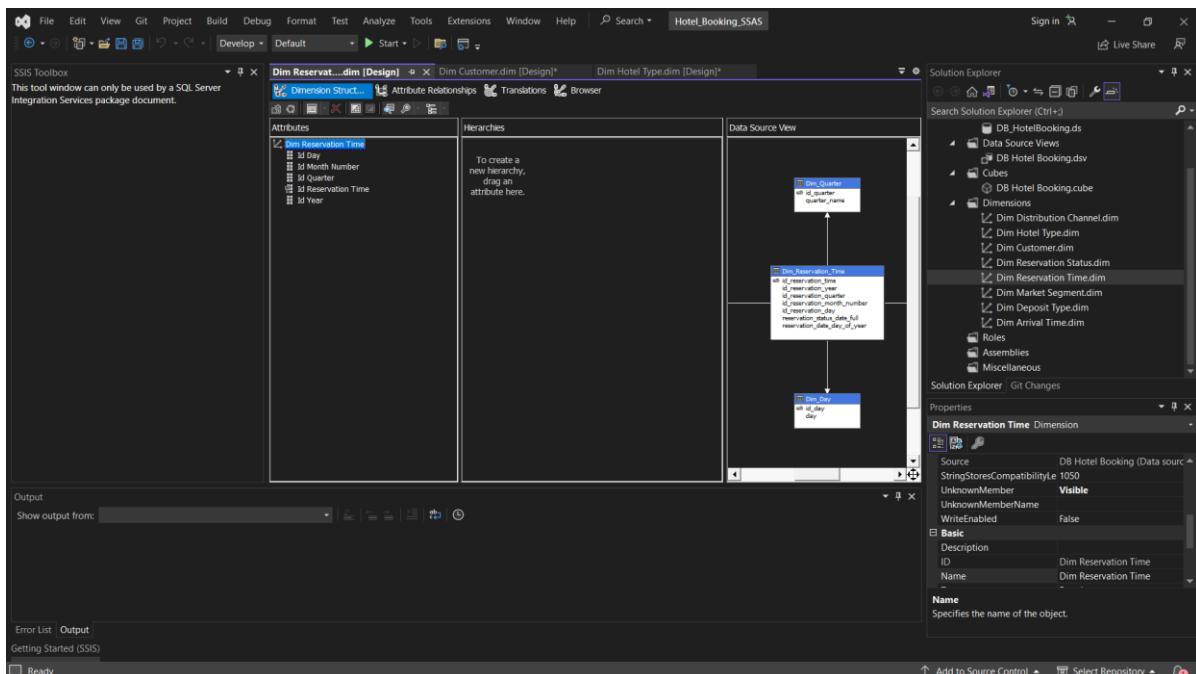


Figure 218. Bảng thuộc tính Dim_Reservation_Time, Dim_Year, Dim_Quarter, Dim_Month và Dim_Day

- **Bước 2:** Kéo các thuộc tính sử dụng từ Data Source View sang Attributes.

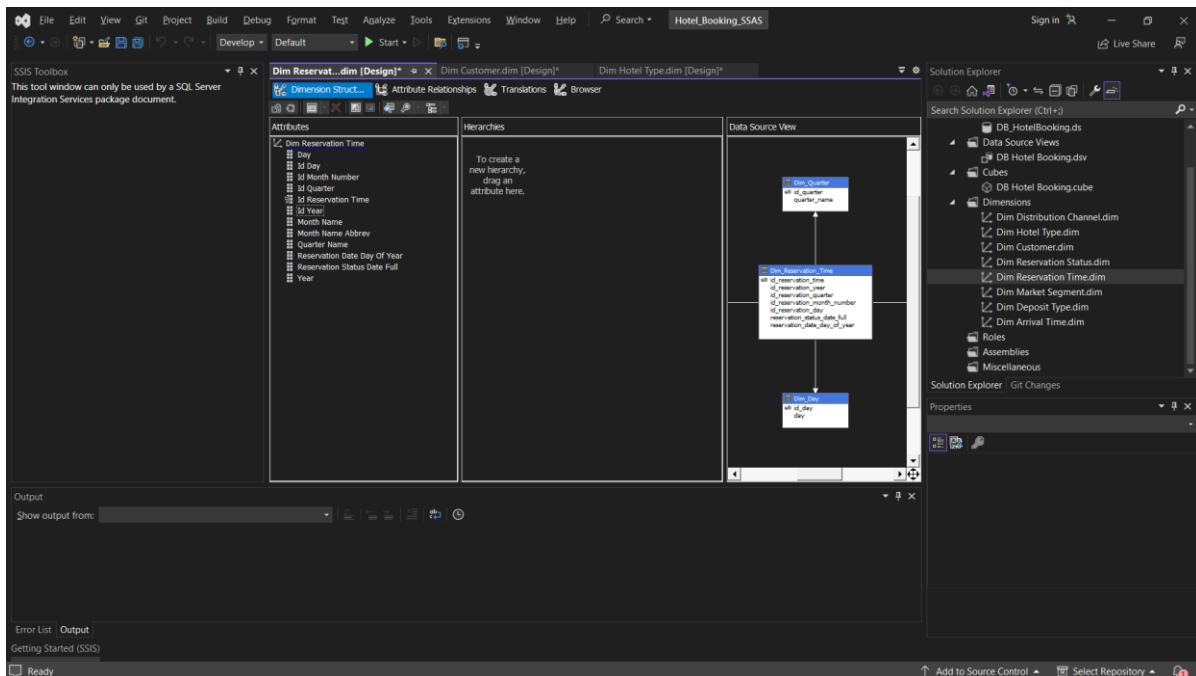


Figure 219. Kết quả sau khi kéo thả các thuộc tính

Bảng Dim_Market_Segment

- **Bước 1:** Ở Solution Explorer, double-click vào Dim Market Segment.dim để tiến hành thêm các thuộc tính.

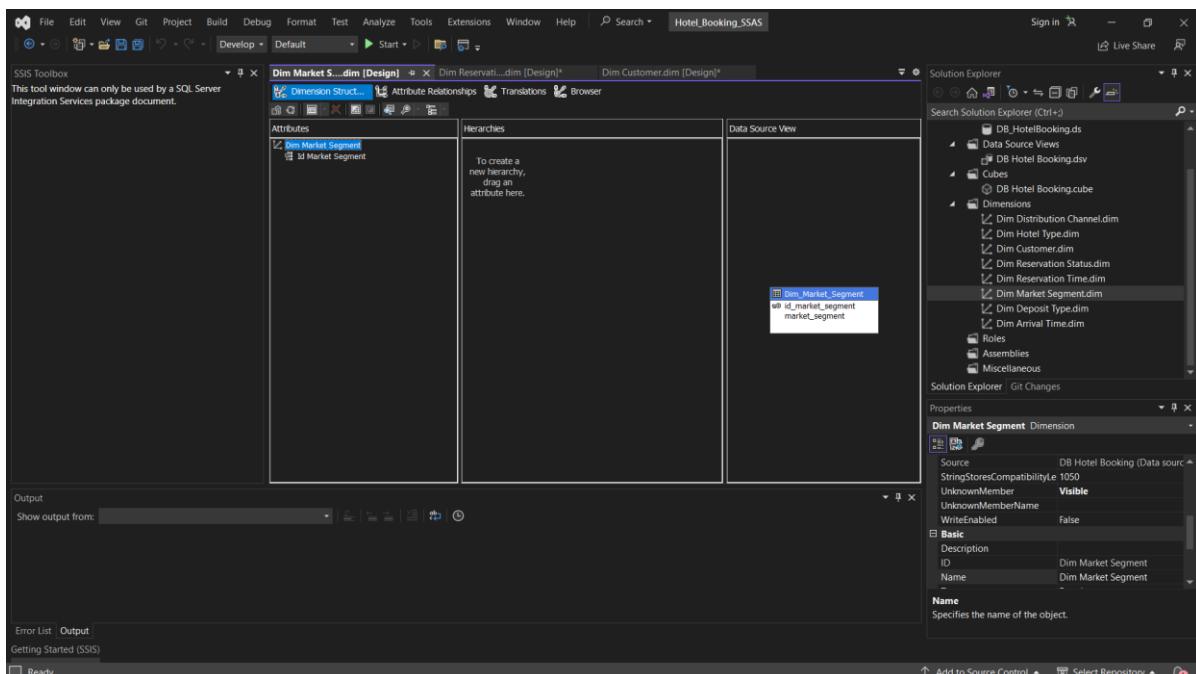


Figure 220. Bảng thuộc tính Dim_Market_Segment

- **Bước 2:** Kéo các thuộc tính sử dụng từ Data Source View sang Attributes.

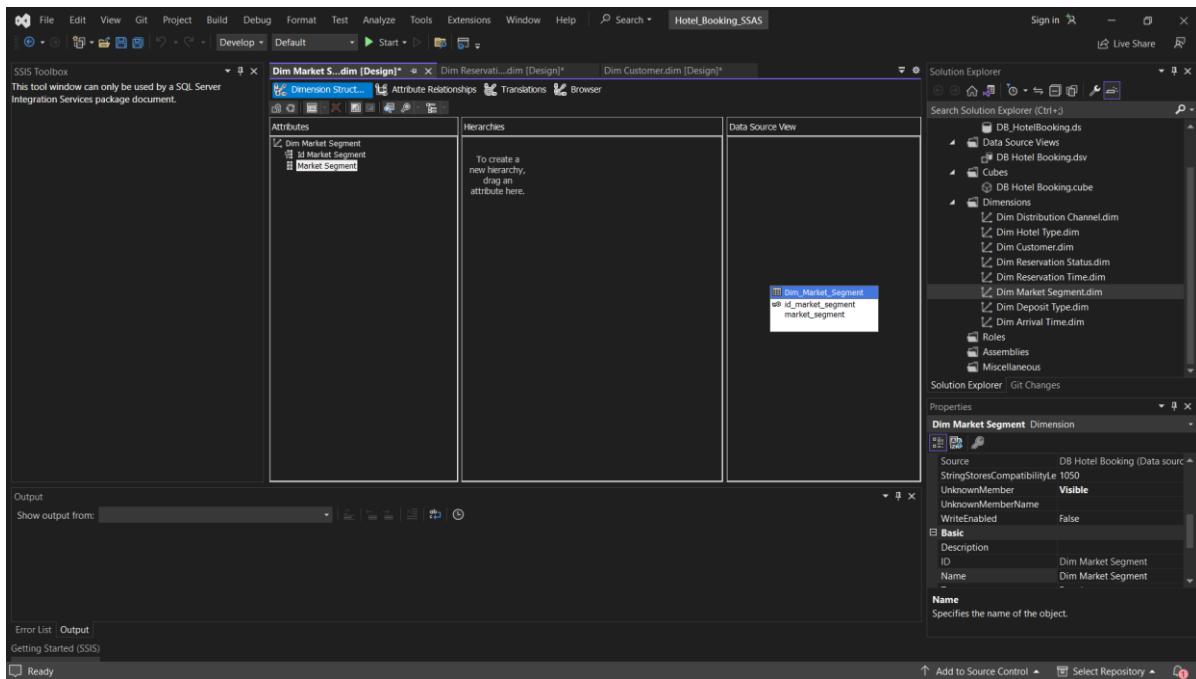


Figure 221. Kết quả sau khi kéo thả các thuộc tính

Bảng Dim_Deposit_Type

- **Bước 1:** Ở Solution Explorer, double-click vào Dim Deposit Type.dim để tiến hành thêm các thuộc tính.

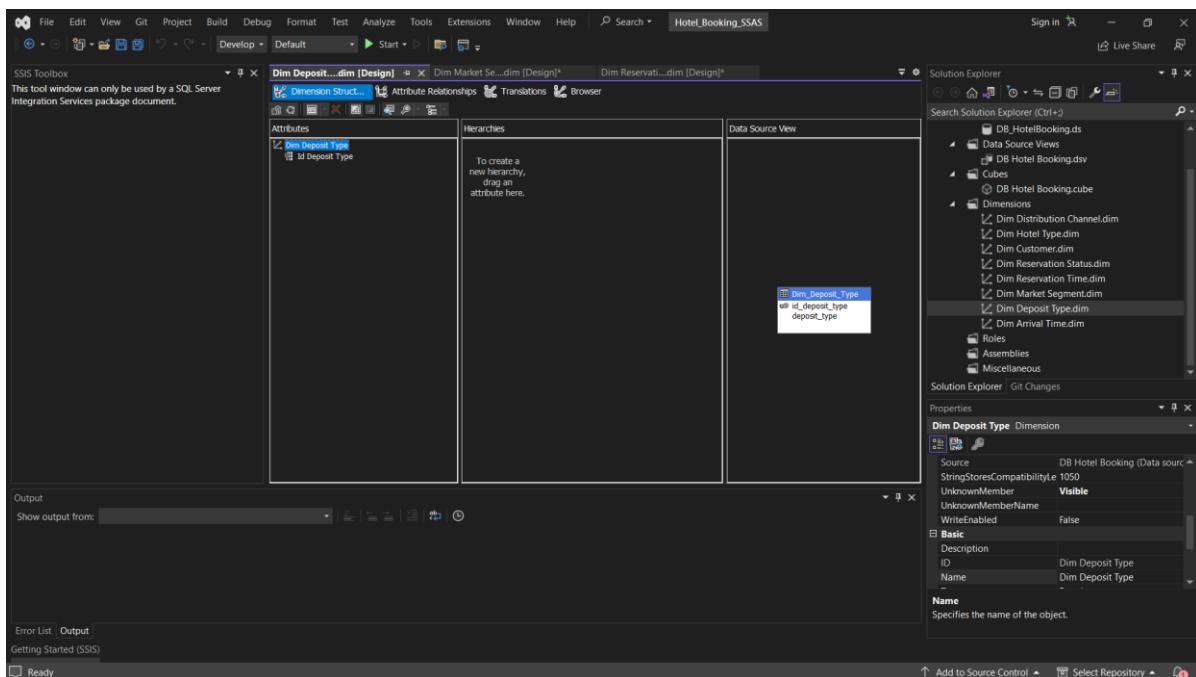


Figure 222. Bảng thuộc tính Dim_Deposit_Type

- **Bước 2:** Kéo các thuộc tính sử dụng từ Data Source View sang Attributes.

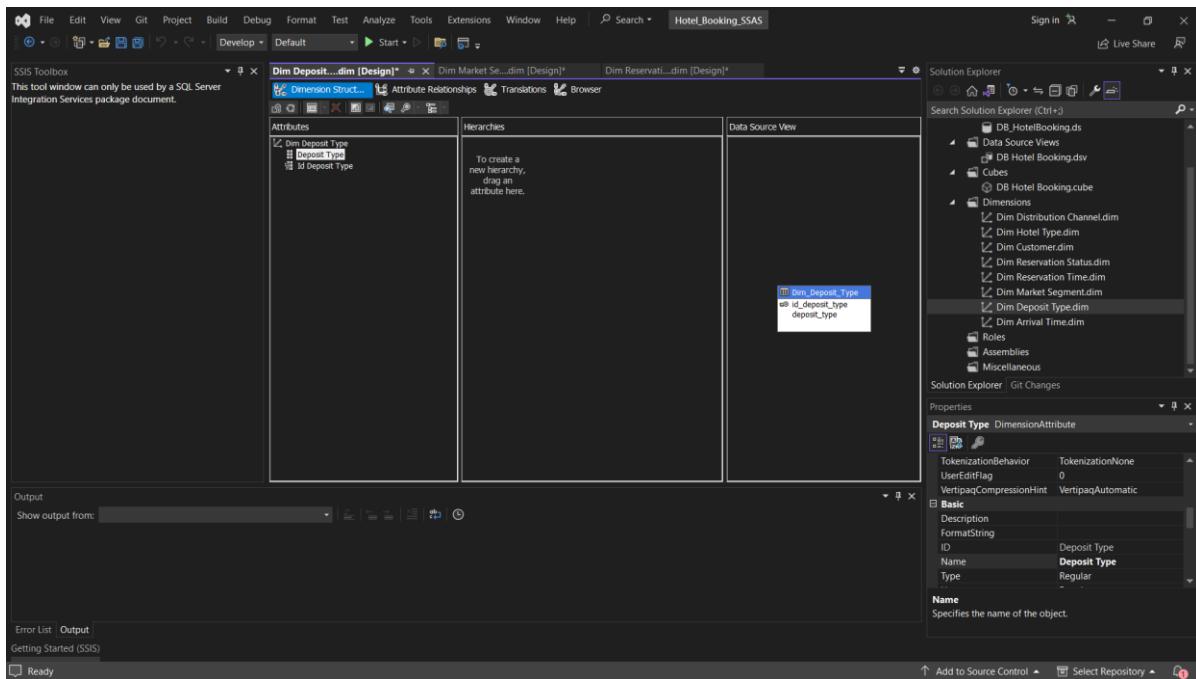


Figure 223. Kết quả sau khi kéo thả các thuộc tính

Bảng Dim_Arrival_Time, Dim_Year, Dim_Quarter, Dim_Month và Dim_Day

- **Bước 1:** Ở Solution Explorer, double-click vào Dim Arrival Time.dim để tiến hành thêm các thuộc tính.

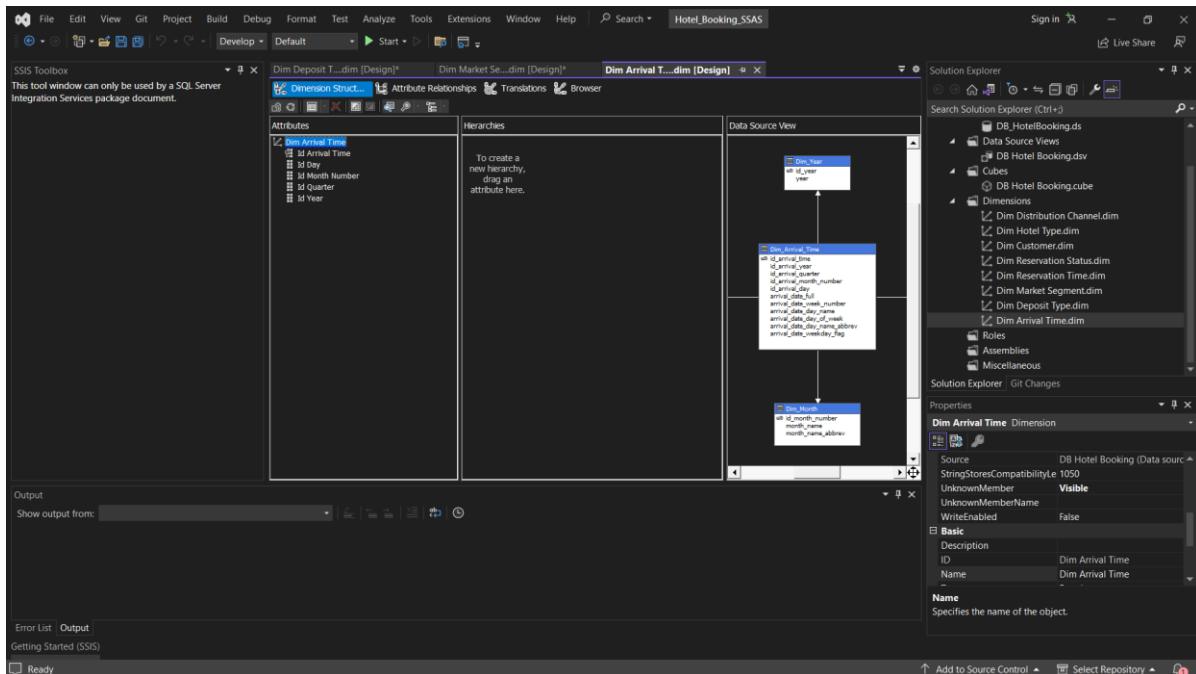


Figure 224. Bảng thuộc tính Dim_Arrival_Time, Dim_Year, Dim_Quarter, Dim_Month và Dim_Day

- **Bước 2:** Kéo các thuộc tính sử dụng từ Data Source View sang Attributes.

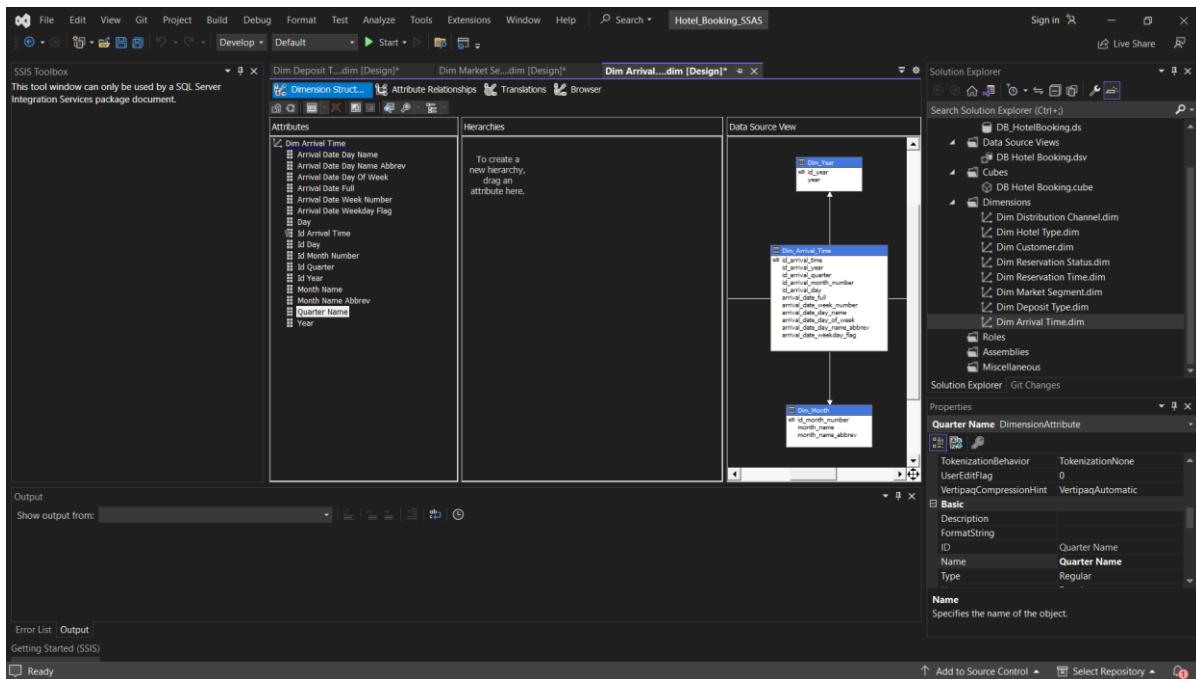


Figure 225. Kết quả sau khi kéo thả các thuộc tính

3.2.5.2. Án các dòng giữ liệu không có giá trị (Unknown)

- **Bước 1:** Tại Dimensions, double click các bảng Dim: Distribution Channel, Hotel Type, Customer, Reservation Status, Market Segment, Deposit Type.

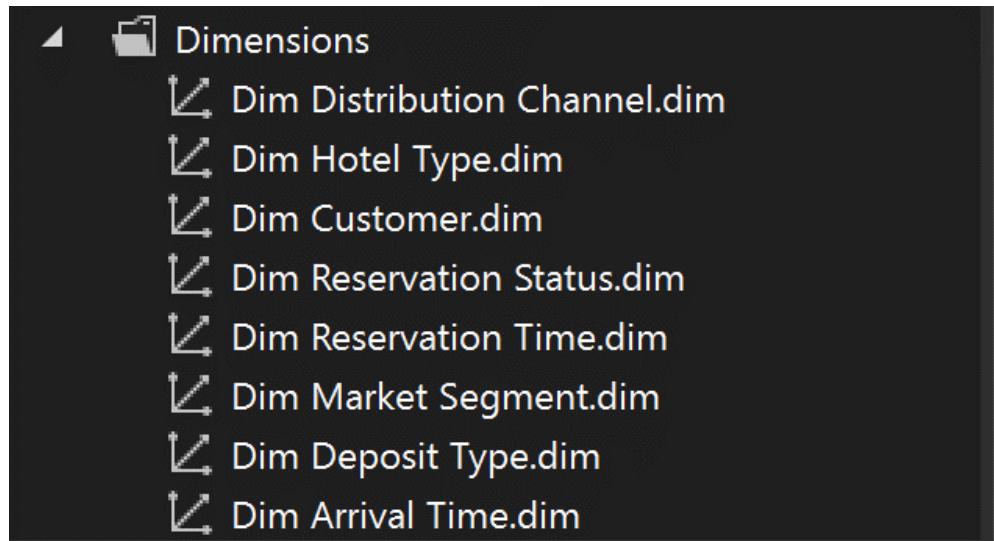


Figure 226. Các bảng Dim của cube

- **Bước 2:** Tại cửa sổ Properties của các bảng, thay đổi thông tin UnknownMember từ **Visible** thành **Hidden**.

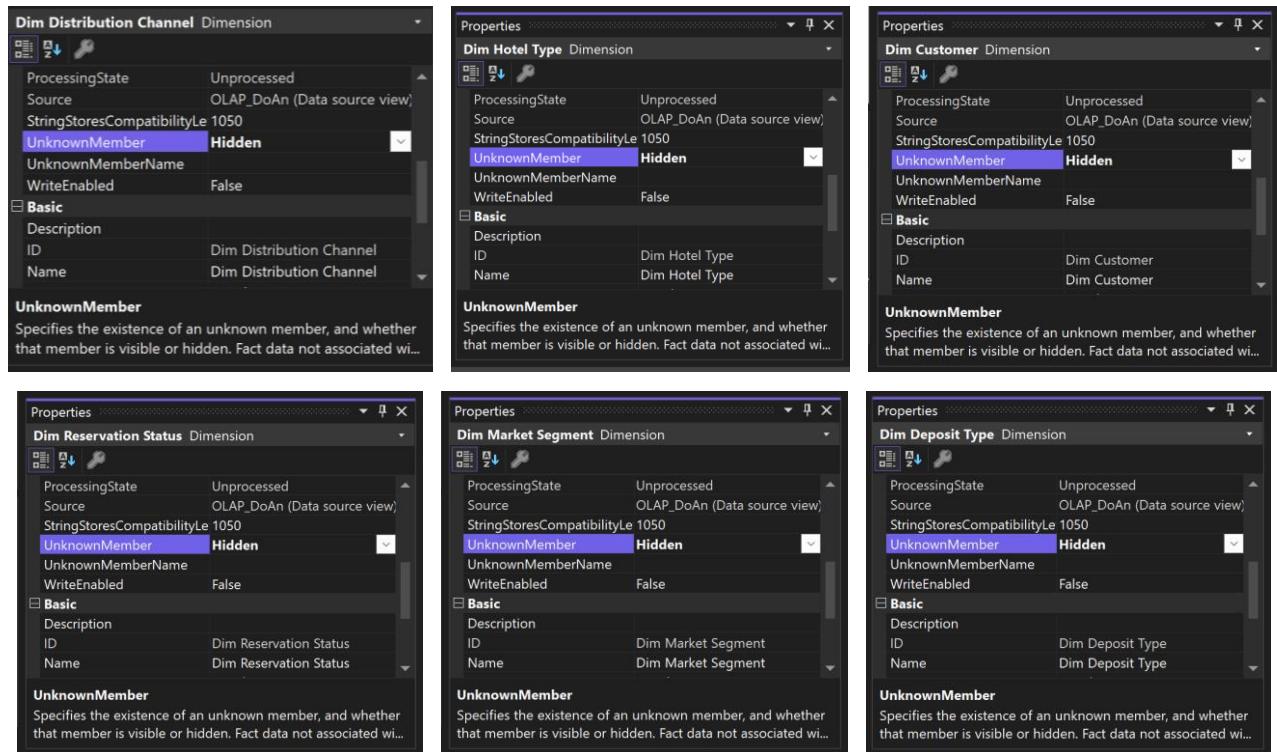


Figure 227. Kết quả sau khi chỉnh sửa UnknownMember

3.2.6. Deploy và process project SSAS

- **Bước 1:** Tại Solution Explorer, chuột phải vào Hotel_Booking_SSAS chọn Deploy.

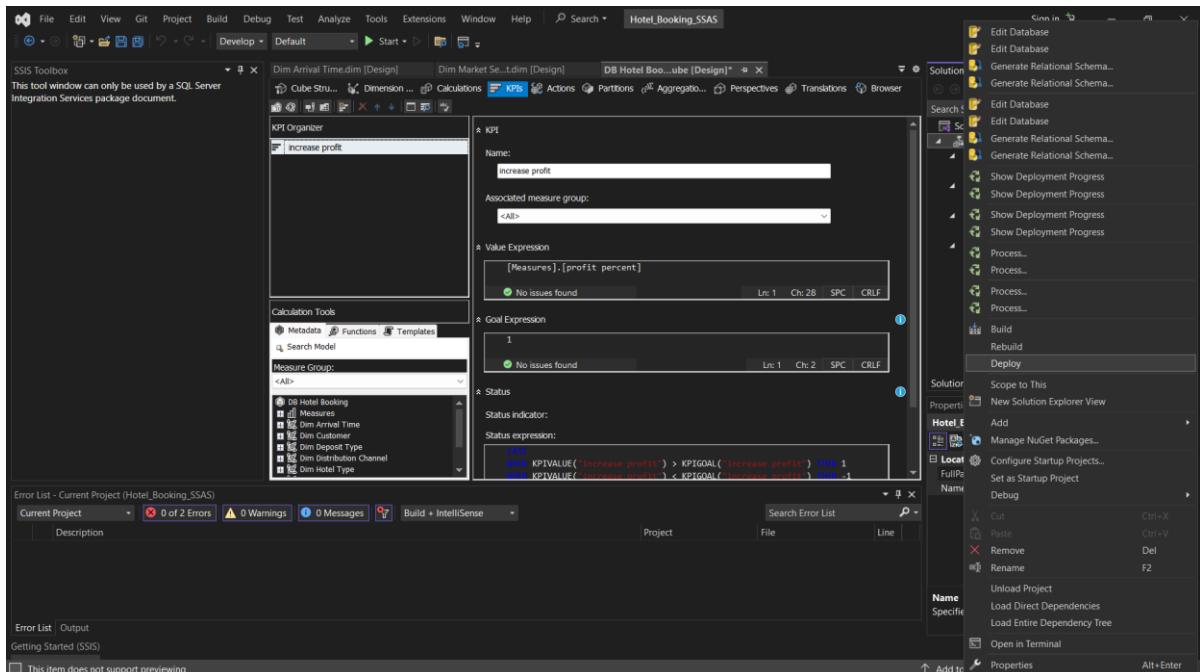


Figure 228. Bắt đầu Deploy

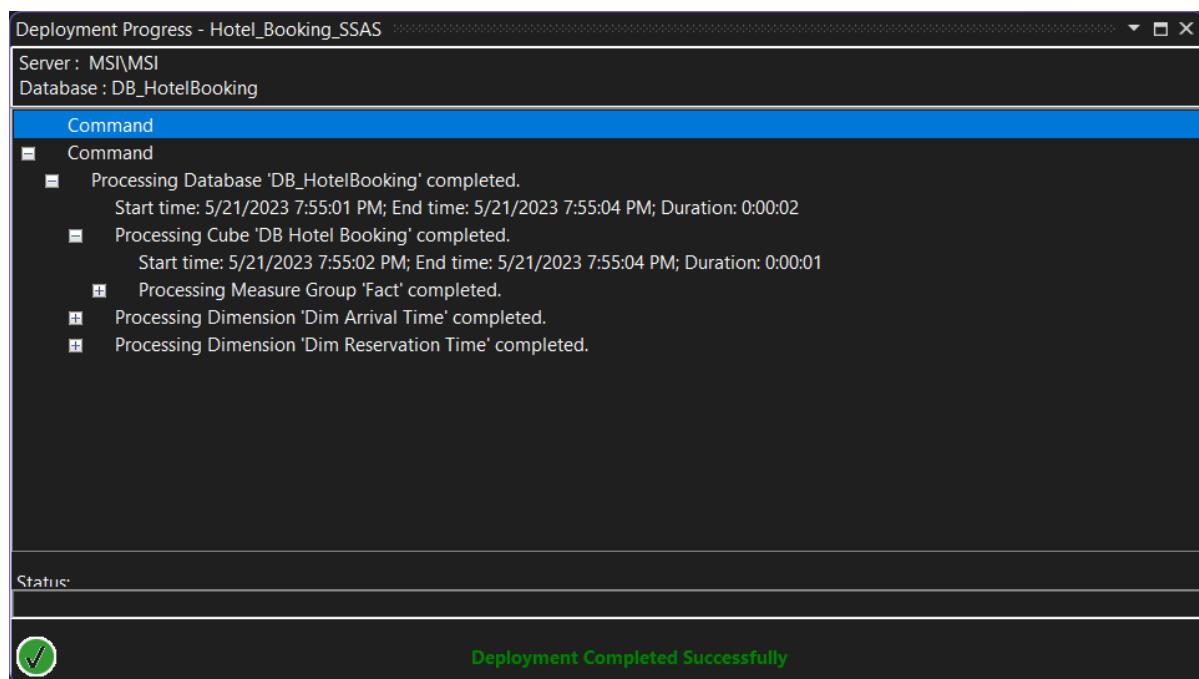


Figure 229. Deploy thành công

- **Bước 2:** Tại Cubes, chuột phải vào DB Hotel Booking.cube chọn Process.

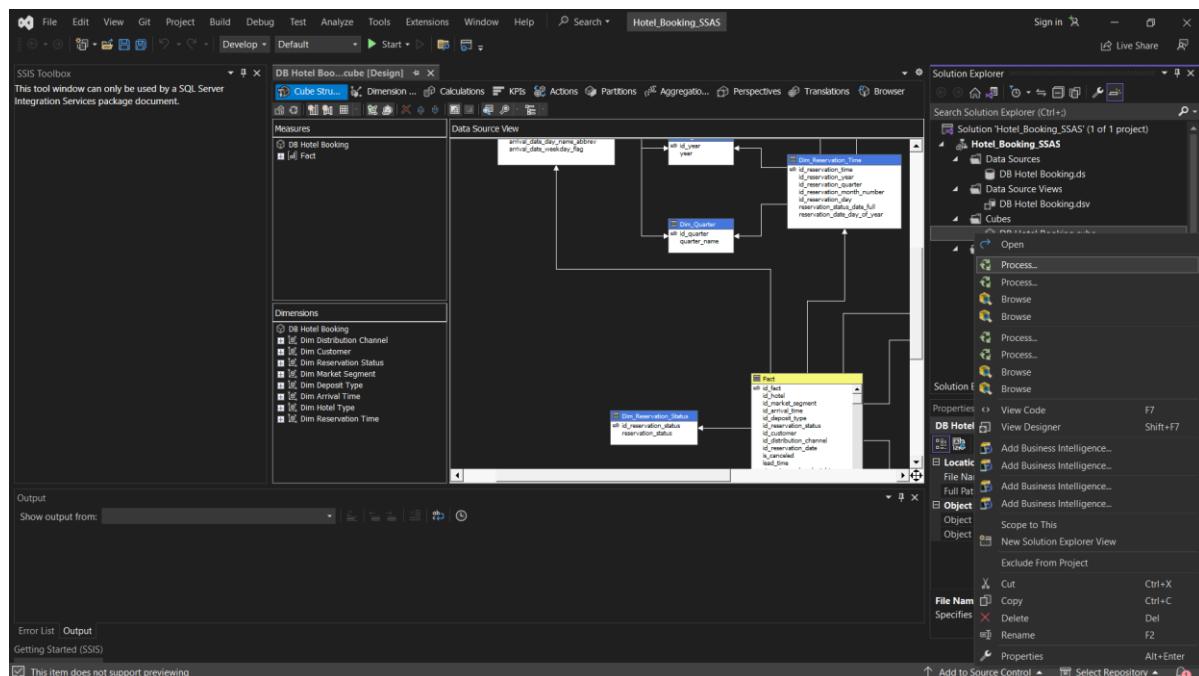


Figure 230. Thực hiện process Cube

- **Bước 3:** Nhấn Yes để tiếp tục. Sau đó chọn Run để bắt đầu thực hiện.

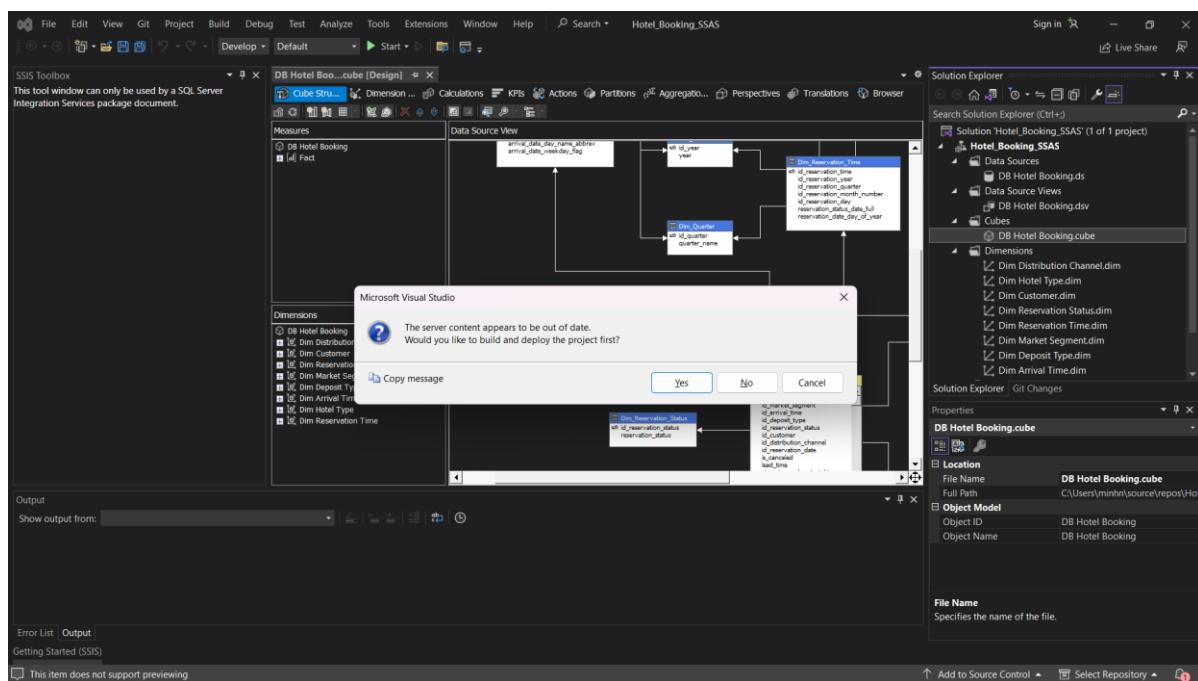


Figure 231. Chọn Yes để tiếp tục

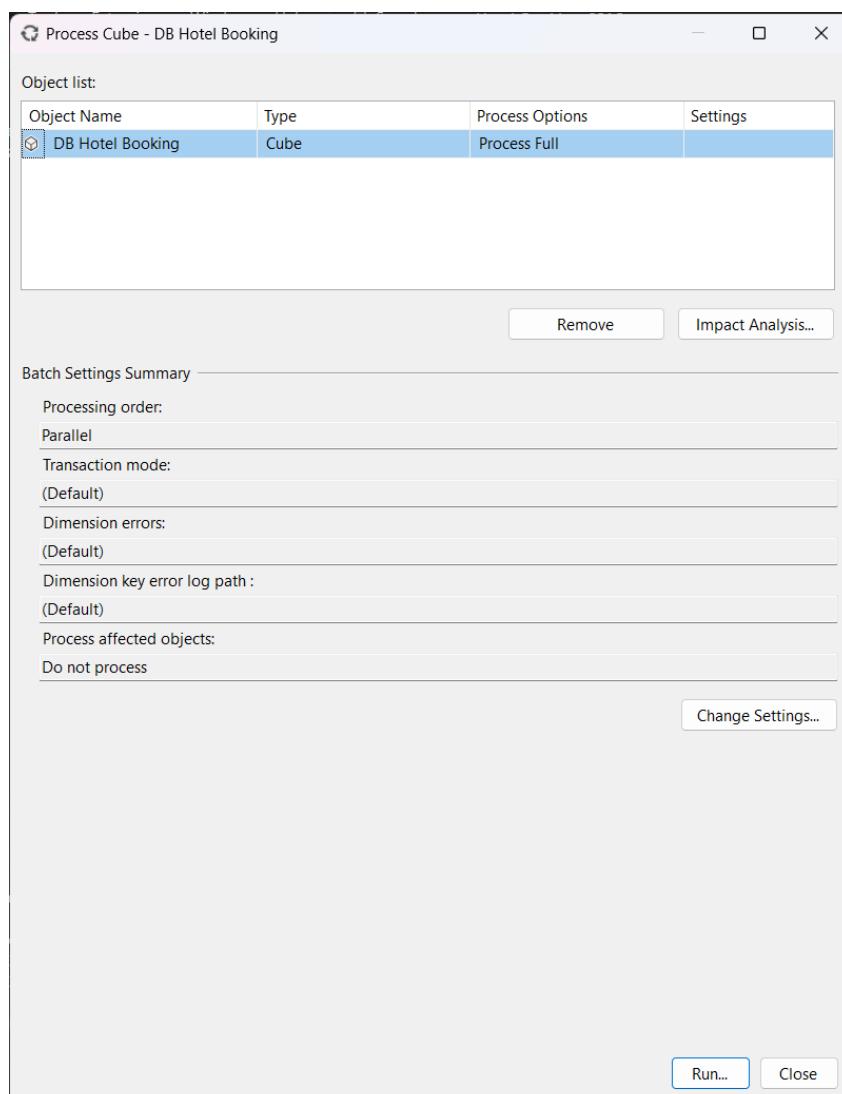


Figure 232. Chọn Run để tiến hành process Cube

- **Bước 4:** Sau khi chạy thành công ấn Close để hoàn tất.

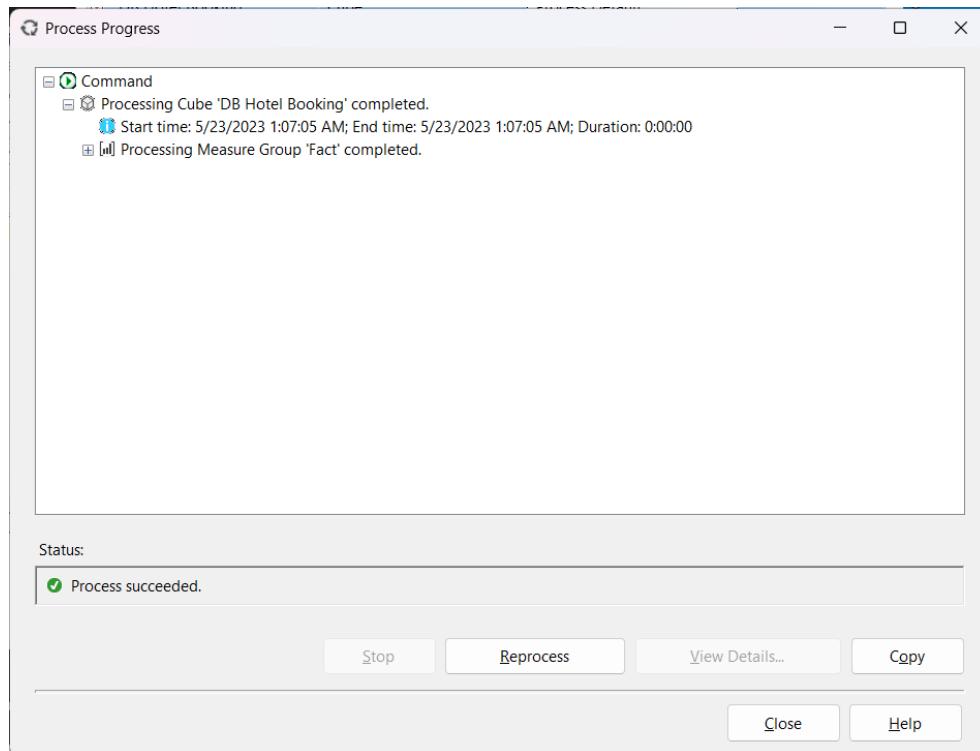


Figure 233. Hoàn tất quá trình process Cube

3.2.7. Định nghĩa Named Set

Khái niệm:

- Named Set là một thể hiện của ngôn ngữ MDX cái mà trả về một cái tập các thành viên trong một chiều.
- Named Set giống như những operator tự lọc trước khi truy vấn MDX.
- **Bước 1:** Tạo mới Named Set bằng cách click chuột phải chọn New Named Set.

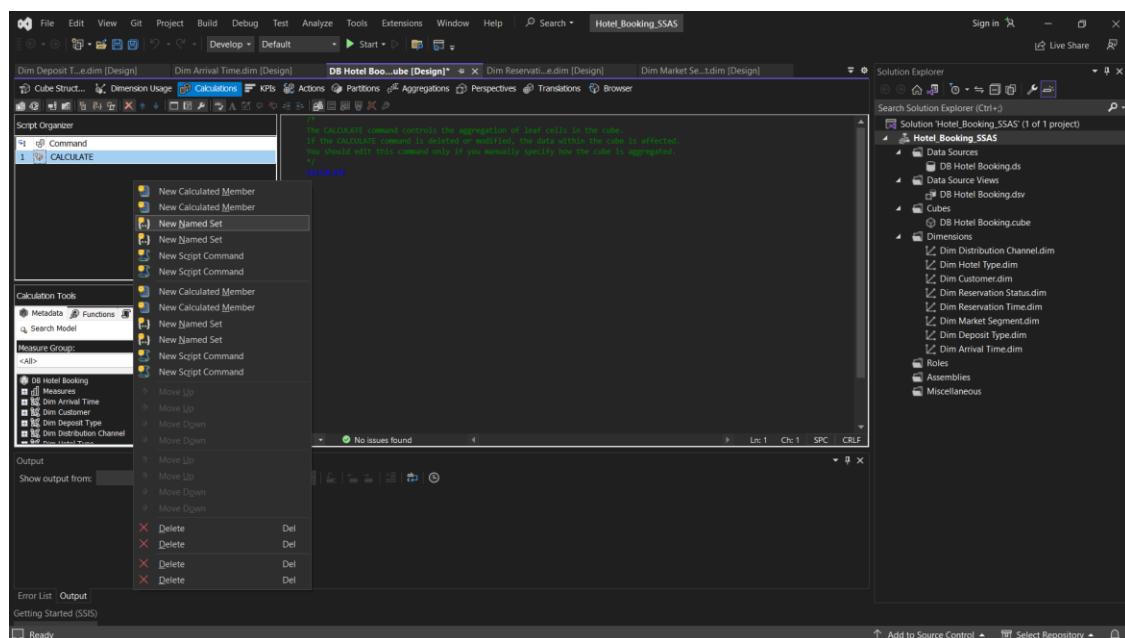


Figure 234. Tạo Named Set mới

- **Bước 2:** Thành phần trong Named Set.

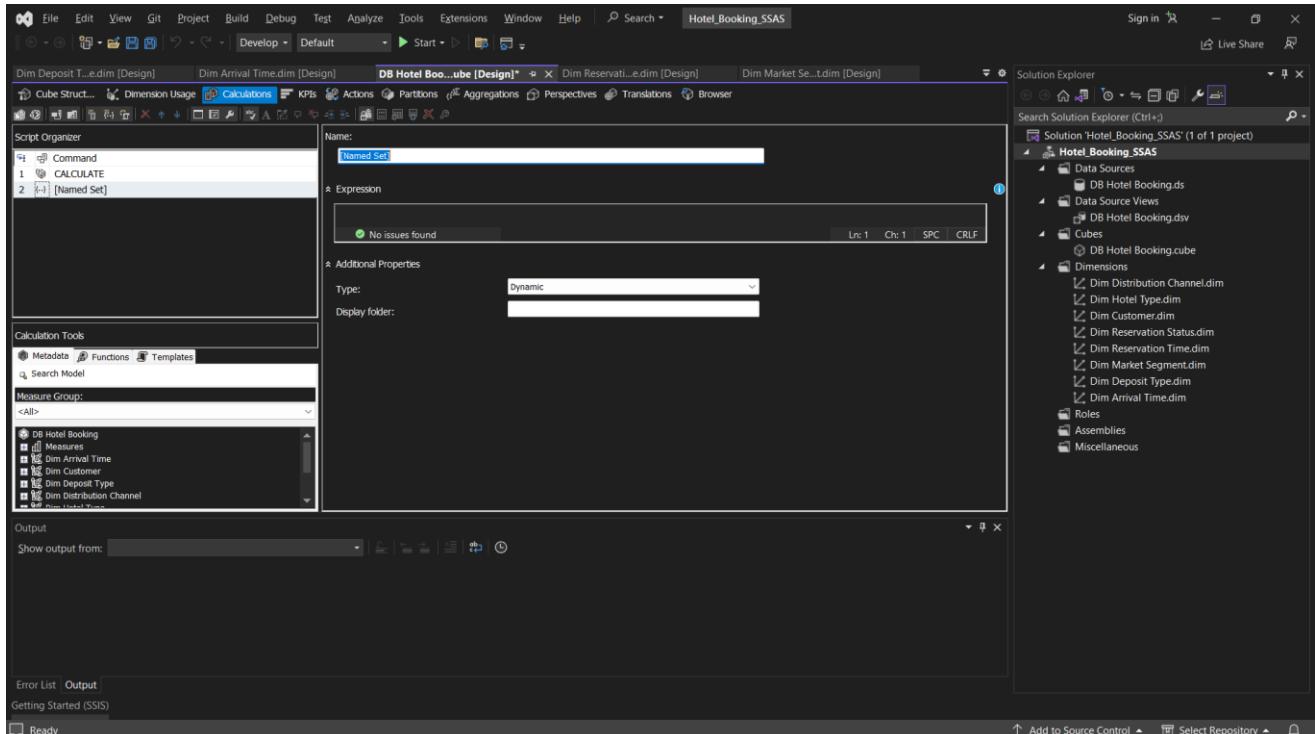


Figure 235. Giao diện của 1 Named Set

- **Bước 3.1:** Name: Chứa tên của Named Set.
- **Bước 3.2:** Expression: Chứa các hàm trong ngôn ngữ MDX.
- **Bước 3.3:** Ví dụ: quốc gia tạo ra tổng doanh thu lớn hơn 50000.

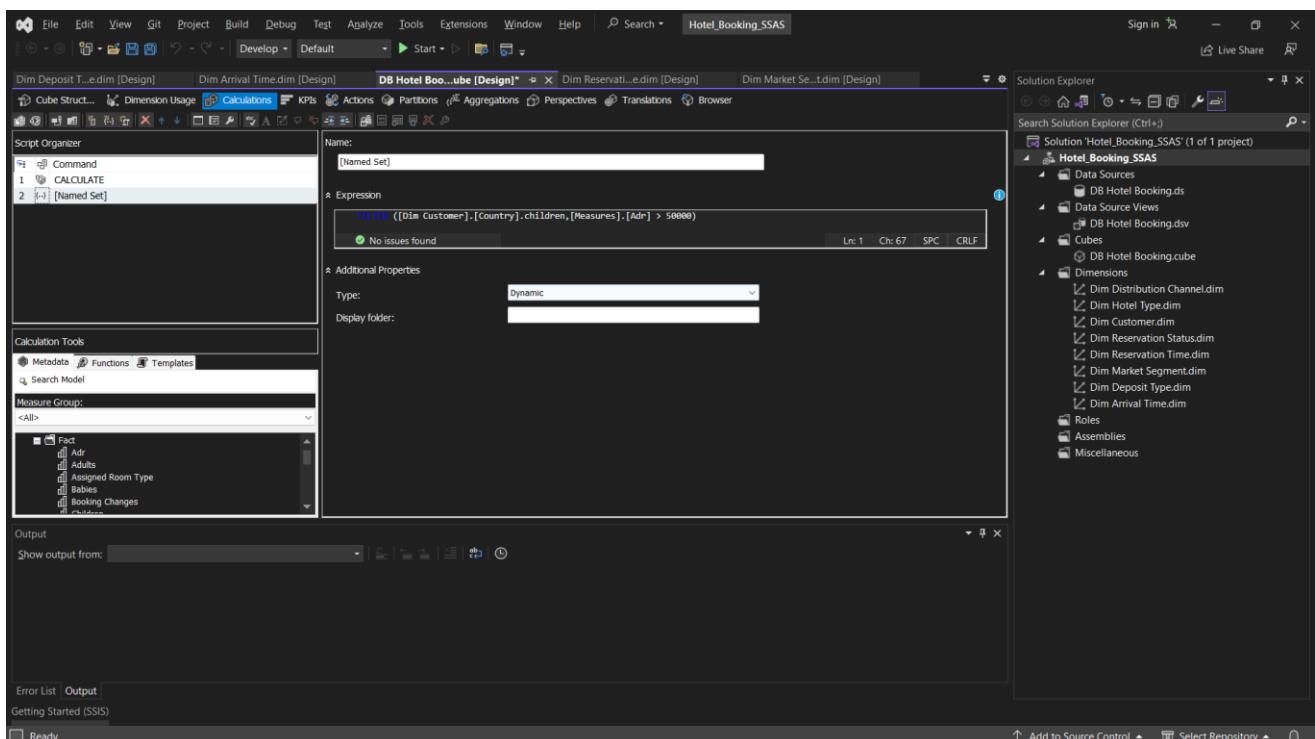


Figure 236. Hàm Filter để lọc dữ liệu trong Named Set

3.2.8. Phân tích dữ liệu trên các khối (cube) bằng công cụ SSAS, ngôn ngữ MDX, Power BI và Excel

3.2.8.1. Báo cáo tình hình kinh doanh

- Thống kê tổng số lượng khách hàng đặt phòng, tổng doanh thu của từng loại khách sạn theo từng quý của năm.

Ý nghĩa câu truy vấn: Cho phép người quản lý có cái nhìn tổng quan tình hình kinh doanh của khách sạn theo từng quý trong năm từ đó đưa ra các chiến lược kinh doanh phù hợp với những quý có doanh thu ổn định và vượt trội.

- Công cụ SSAS trên các khối Cube

Kéo thuộc tính Adr và Fact Count từ Fact sang khung truy vấn → Kéo thuộc tính Quarter Name và Year từ Dim Arrival Time sang khung truy vấn → Chọn Click to execute the query.

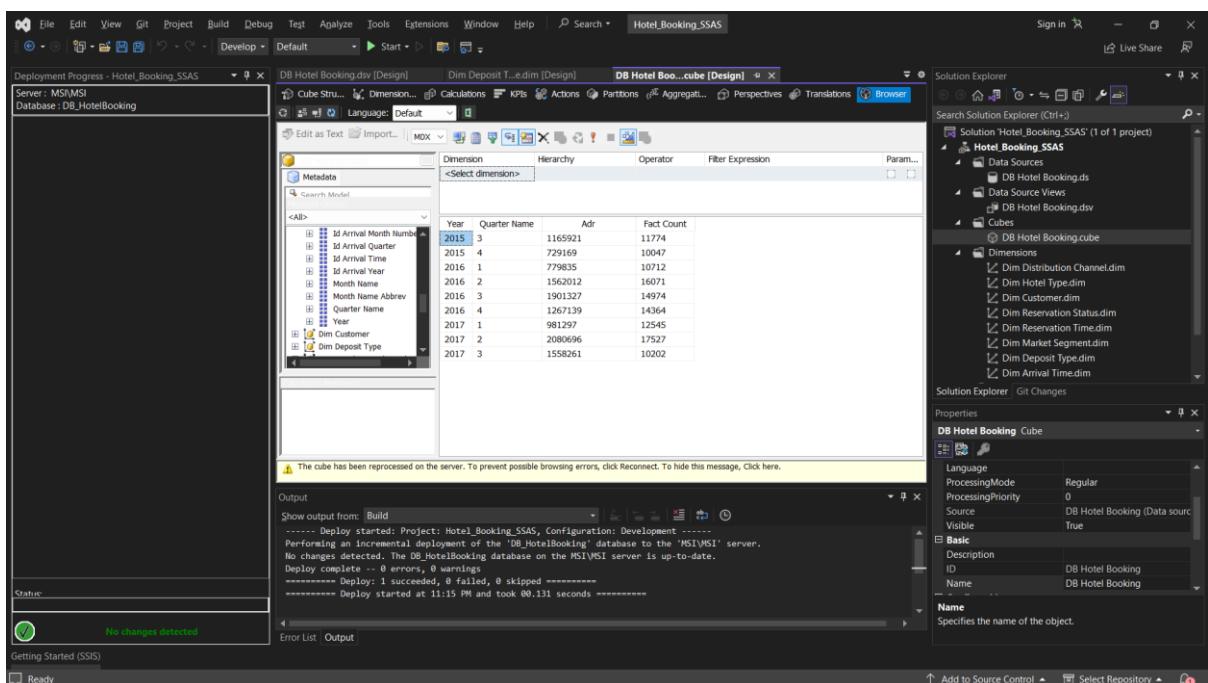


Figure 237. Kết quả SSAS câu 1

- Ngôn ngữ MDX trên các khối Cube

Query:

```
SELECT {[Measures].[Adr], [Measures].[Fact Count]} ON ROWS,
NON EMPTY {[Dim Arrival Time].[Year].children* [Dim Arrival Time].[Quarter Name].children} ON COLUMNS
FROM [DB Hotel Booking]
```

Kết quả:

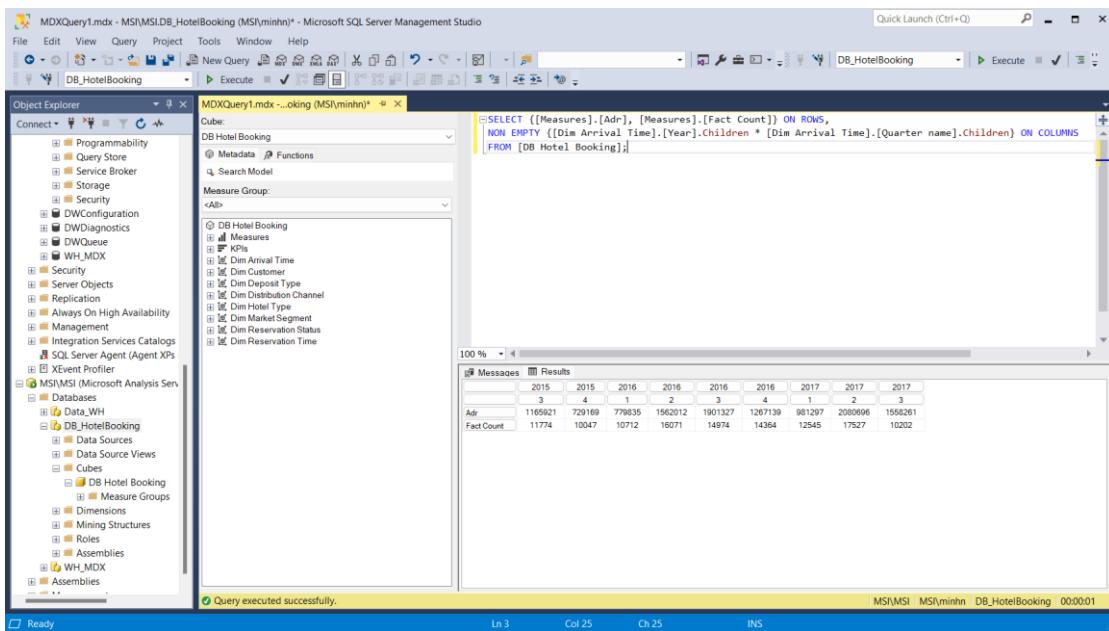


Figure 238. Kết quả MDX câu 1

- Công cụ Power BI

Kéo thả thuộc tính year từ Dim_Year vào Columns vào trường Columns → Kéo thả thuộc tính quarter từ Dim_Quarter vào Columns vào trường Rows → Kéo thả thuộc tính adr và id_fact từ bảng Fact vào Values.

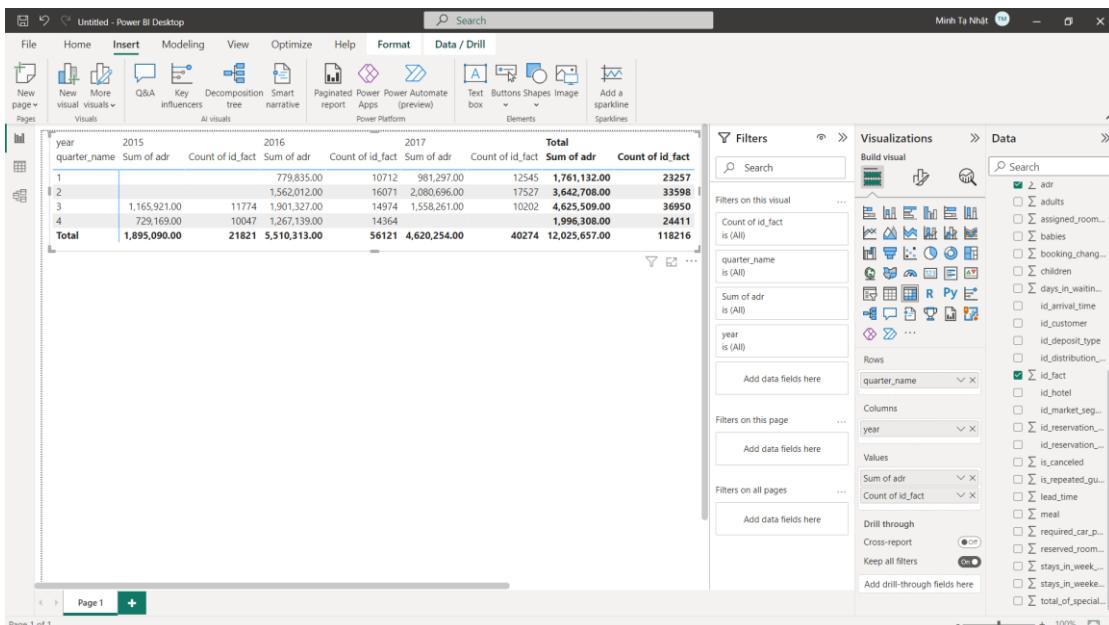


Figure 239. Kết quả Power BI câu 1

- BI Chart

Chọn kiểu Clustered Column Chart. Kéo thả thuộc tính year từ Dim_Year vào X-axis và quarter_name từ Dim_Quarter vào Legend → Kéo thả thuộc tính adr từ

bảng Fact vào Y-axis để có được biểu đồ doanh thu và id_fact từ bảng Fact vào Y-axis để có biểu đồ về lượt đặt phòng.

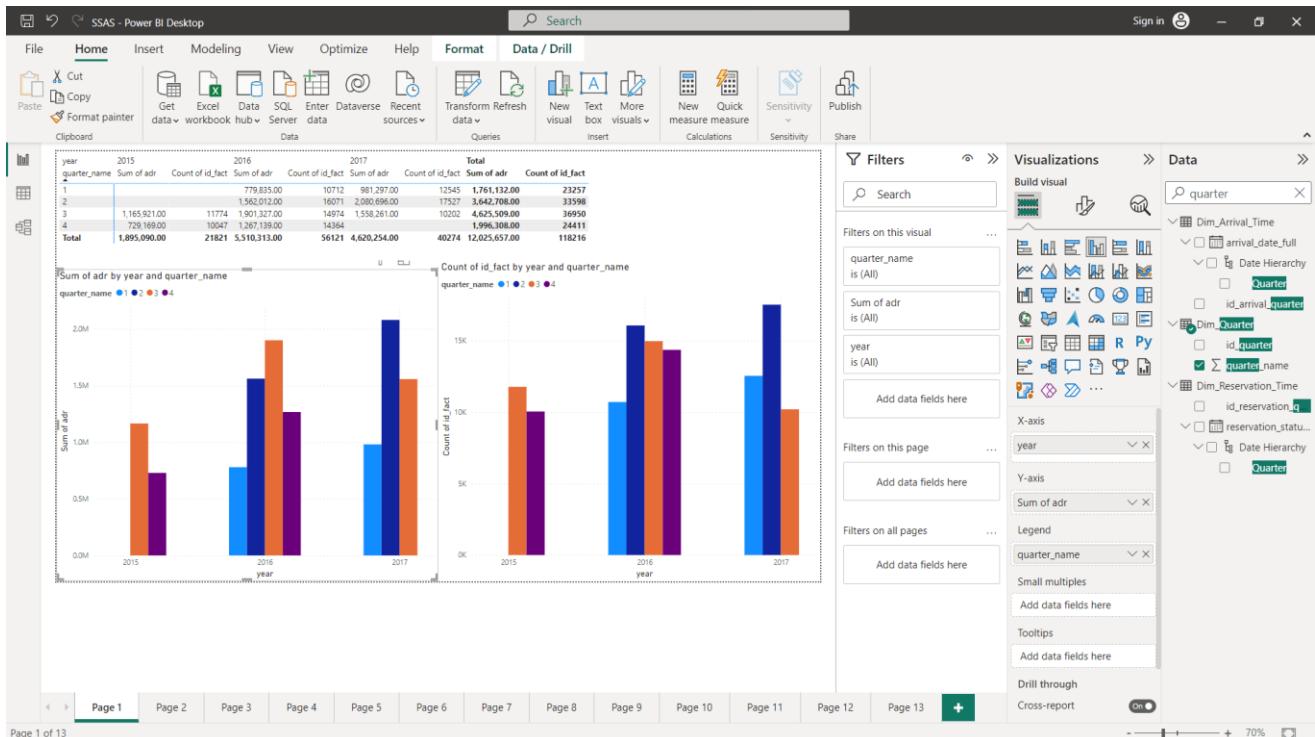


Figure 240. Kết quả trực quan Power BI câu 1

- Công cụ Excel

Kéo thả thuộc tính year từ Dim_Year vào Columns và quarter_name từ Dim_Quarter vào Rows → Kéo thả thuộc tính adr và id_fact từ bảng Fact vào Values.

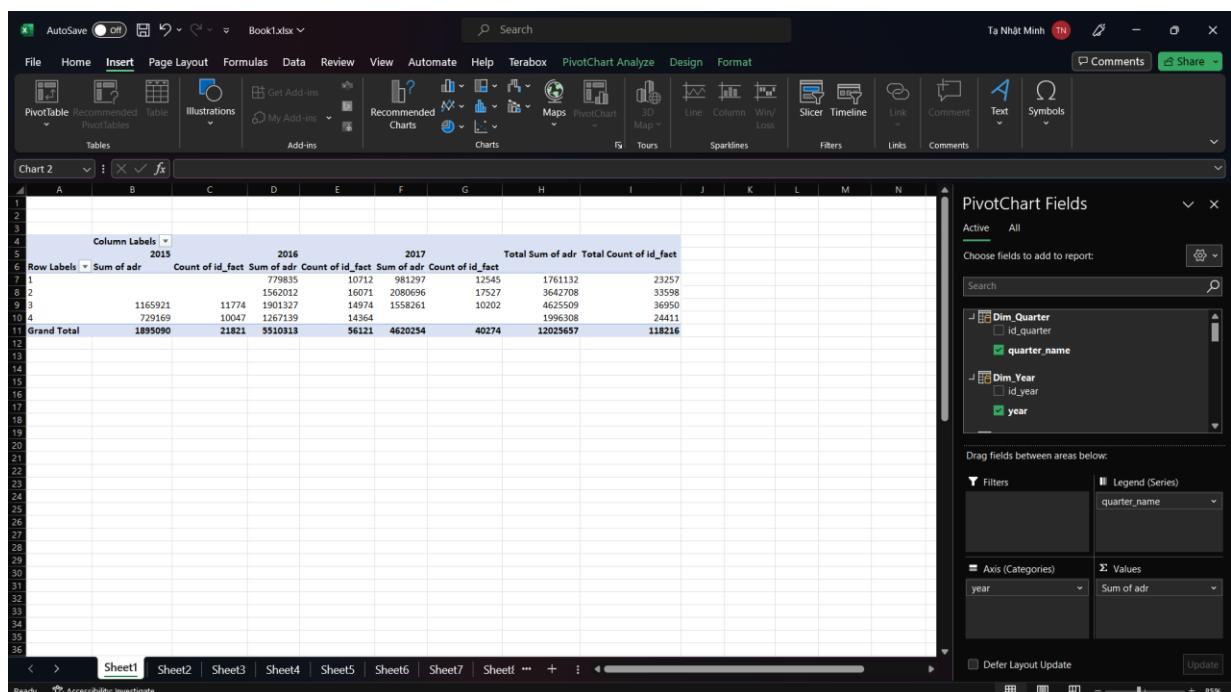


Figure 241. Kết quả Pivot Excel câu 1

- Pivot Chart Excel

Kéo thả thuộc tính year từ Dim_Year vào Axis và quarter_name từ Dim_Quarter vào Legend → Kéo thả thuộc tính adr từ bảng Fact vào Values để có được biểu đồ doanh thu và id_fact từ bảng Fact vào Values để có biểu đồ lượt đặt phòng.

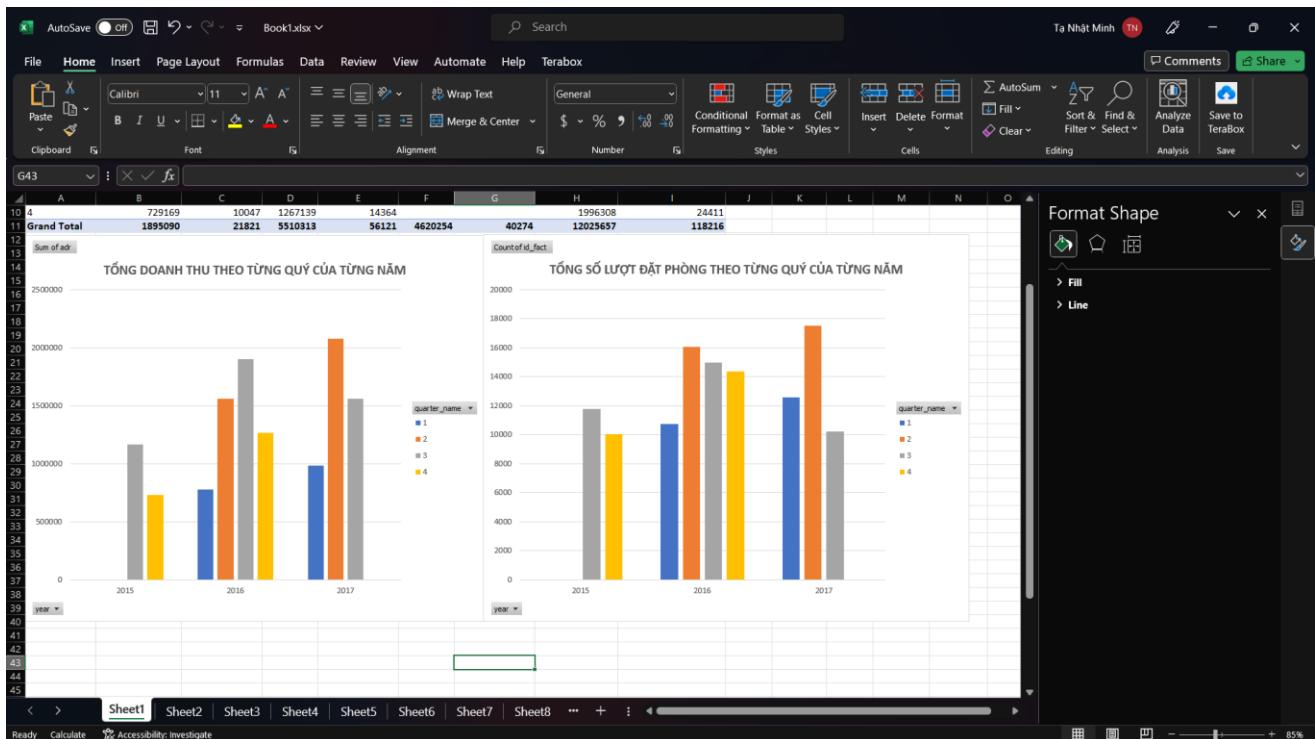


Figure 242. Kết quả Excel Pivot Chart câu 1

- Thống kê doanh thu theo phân khúc khách hàng (Market_segment) theo từng tháng của năm.

Ý nghĩa câu truy vấn: Cho người quản lý biết được khách hàng đến từ nhóm nào là mang lại lợi nhuận nhiều nhất.

- **Công cụ SSAS trên các khối Cube**

Kéo thả Year và Month Name từ Dim Arrival Time sang khung truy vấn → Kéo thả Market Segment từ Dim Market Segment sang khung truy vấn → Kéo thả Adr từ Fact sang khung truy vấn → Chọn click to execute the query.

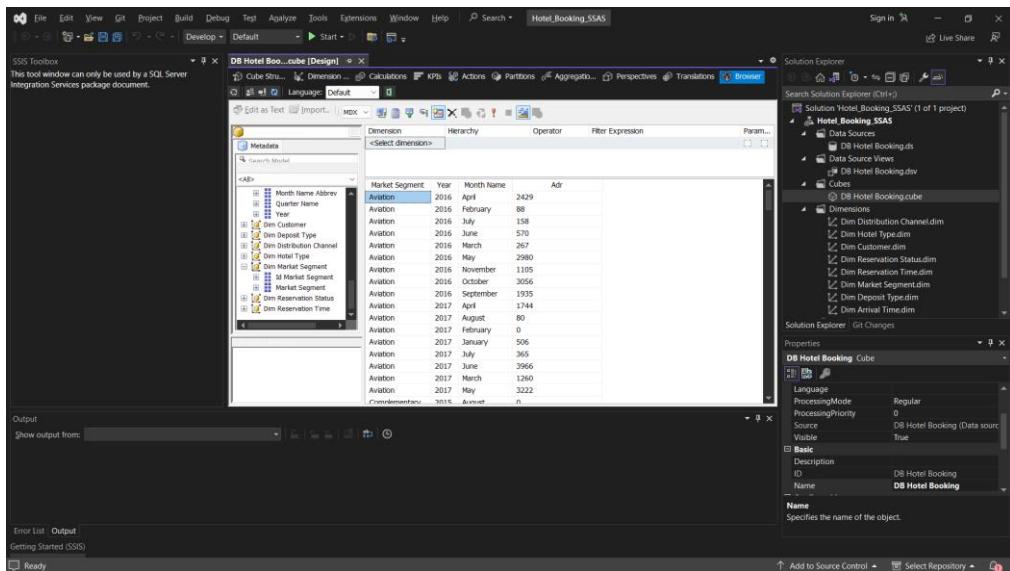


Figure 243. Kết quả SSAS câu 2

- **Ngôn ngữ MDX trên các khối Cube**

Query:

```
SELECT {[Measures].[Adr]} ON ROWS,
NON EMPTY {[Dim Market Segment].[Market Segment].children*
[Dim Arrival Time].[Year].children*
[Dim Arrival Time].[Month Name].children
} ON COLUMNS
FROM [DB Hotel Booking]
```

Kết quả:

Market Segment	Year	Month Name	Adr
Aviation	2016	April	2429
Aviation	2016	February	88
Aviation	2016	July	158
Aviation	2016	June	570
Aviation	2016	March	287
Aviation	2016	May	2980
Aviation	2016	November	1105
Aviation	2016	October	3056
Aviation	2016	September	1935
Aviation	2017	April	1744
Aviation	2017	August	80
Aviation	2017	January	0
Aviation	2017	January	506
Aviation	2017	July	365
Aviation	2017	June	3966
Aviation	2017	March	1280
Aviation	2017	May	3222
Aviation	2015	Aviation	n

Figure 244. Kết quả MDX câu 2

- Công cụ Power BI

Kéo thả year từ bảng Dim_Year và month_name từ bảng Dim_Month vào trường Rows → Kéo thả market_segment từ Dim_Market_Segment vào trường Columns → Kéo thả adr từ bảng Fact vào trường Values.

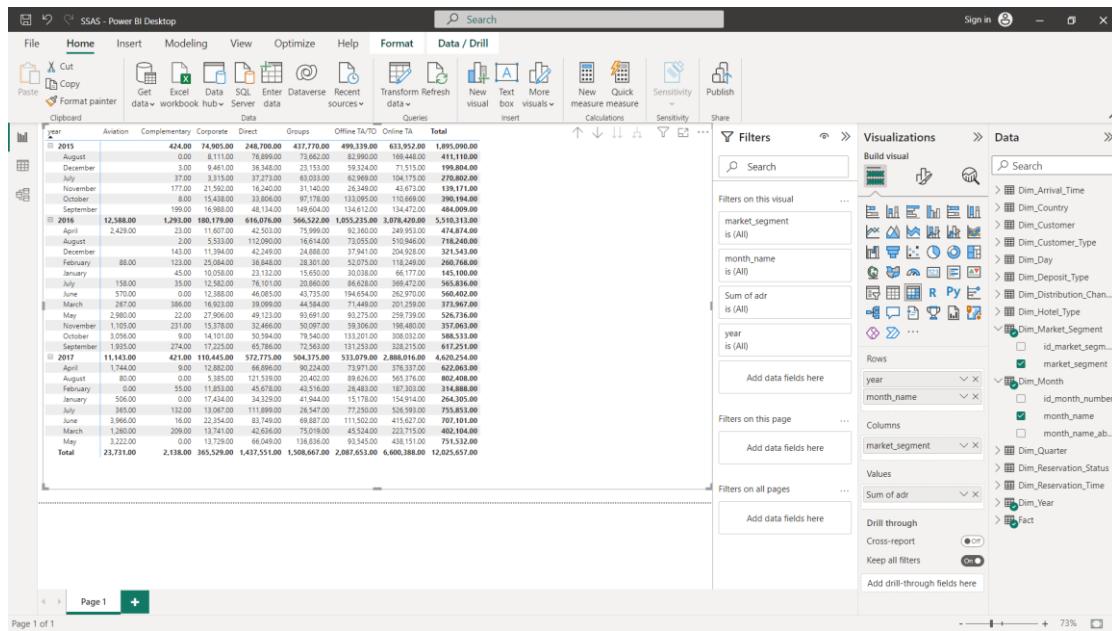


Figure 245. Kết quả Power BI câu 2

- BI Chart

Chọn kiểu Clustered Column Chart. Kéo thả thuộc tính year từ Dim_Year và market_segment từ Dim_Market_Segment vào X-axis và month_name từ Dim_Month vào Legend → Kéo thả thuộc tính adr từ bảng Fact vào Y-axis.

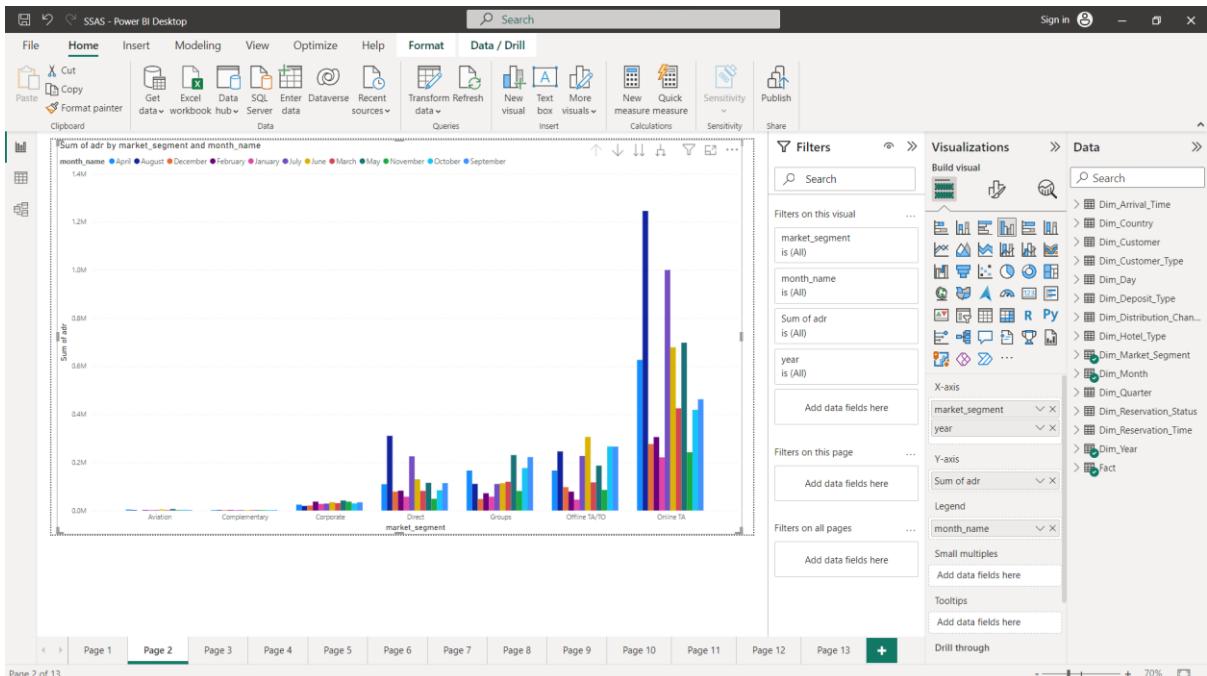


Figure 246. Kết quả trực quan Power BI câu 2

- Công cụ Excel

Kéo thả year từ bảng Dim_Year và month_name từ bảng Dim_Month vào trường Rows → Kéo thả market_segment từ Dim_Market_Segment vào trường Columns → Kéo thả adr từ bảng Fact vào trường Values.

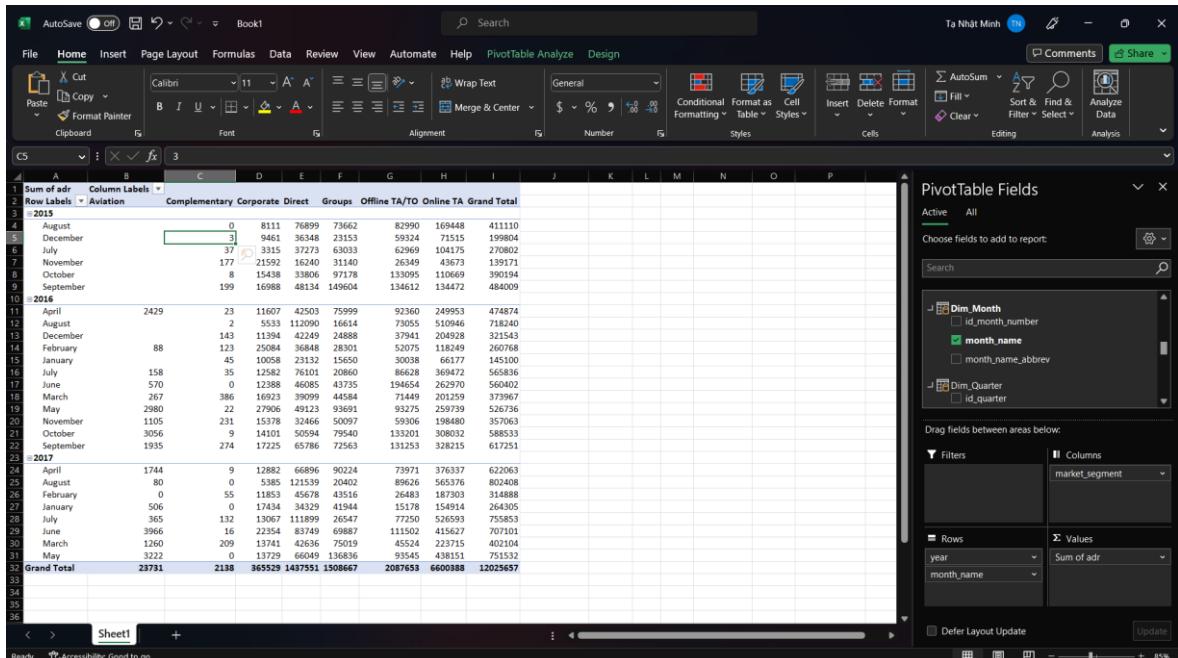


Figure 247. Kết quả Pivot Excel câu 2

- Pivot Chart Excel

Kéo thả thuộc tính year từ Dim_Year vào Filters và month_name từ Dim_Month vào Legend → Kéo thả market_segment từ Dim_Market_Segment vào Axis → Kéo thả thuộc tính adr từ bảng Fact vào Values.

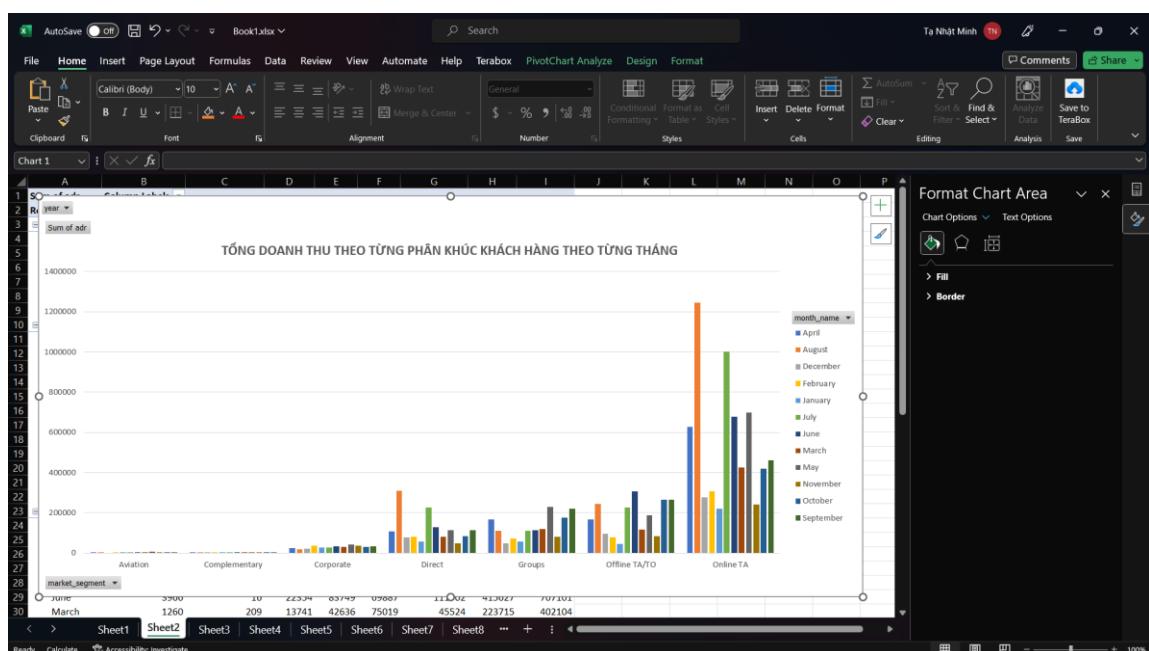


Figure 248. Kết quả Excel Pivot Chart câu 2

3. Thống kê doanh thu theo kênh phân phối theo từng tháng của năm.

Ý nghĩa câu truy vấn: Cho phép người quản lý biết được kênh phân phối nào là mang lại lợi nhuận nhiều nhất.

- Công cụ SSAS trên các khối Cube

Kéo thả Year và Month Name từ Dim Arrival Time sang khung truy vấn → Kéo thả Distribution Channel từ Dim Distribution Channel sang khung truy vấn → Kéo thả Adr từ Fact sang khung truy vấn → Chọn click to execute the query.

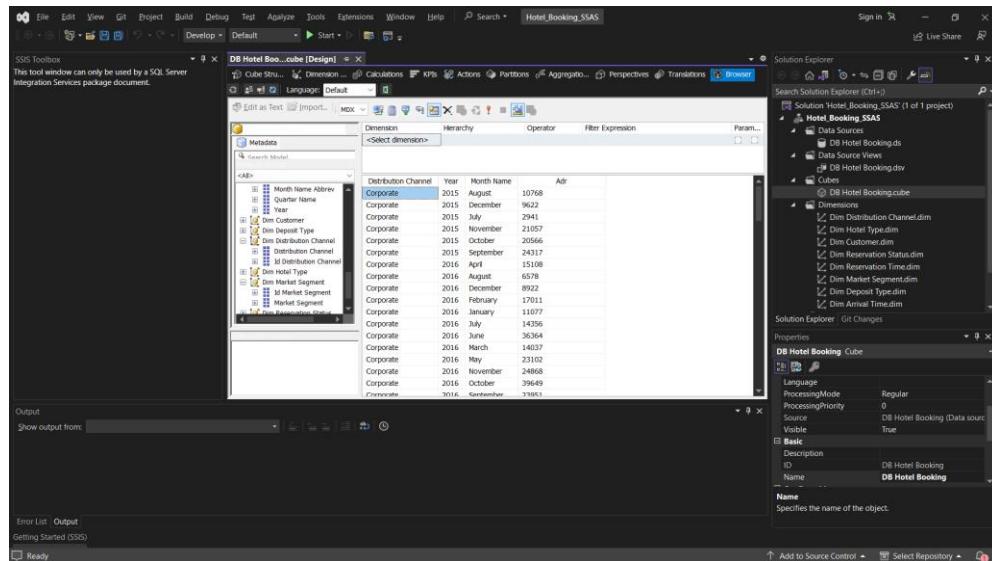


Figure 249. Kết quả SSAS câu 3

- Ngôn ngữ MDX trên các khối Cube

Query:

```

SELECT {[Measures].[Adr]} ON ROWS,
NON EMPTY {[Dim Distribution Channel].[Distribution Channel].children*}
[Dim Arrival Time].[Year].children*
[Dim Arrival Time].[Month Name].children
} ON COLUMNS
FROM [DB Hotel Booking]

```

Kết quả:

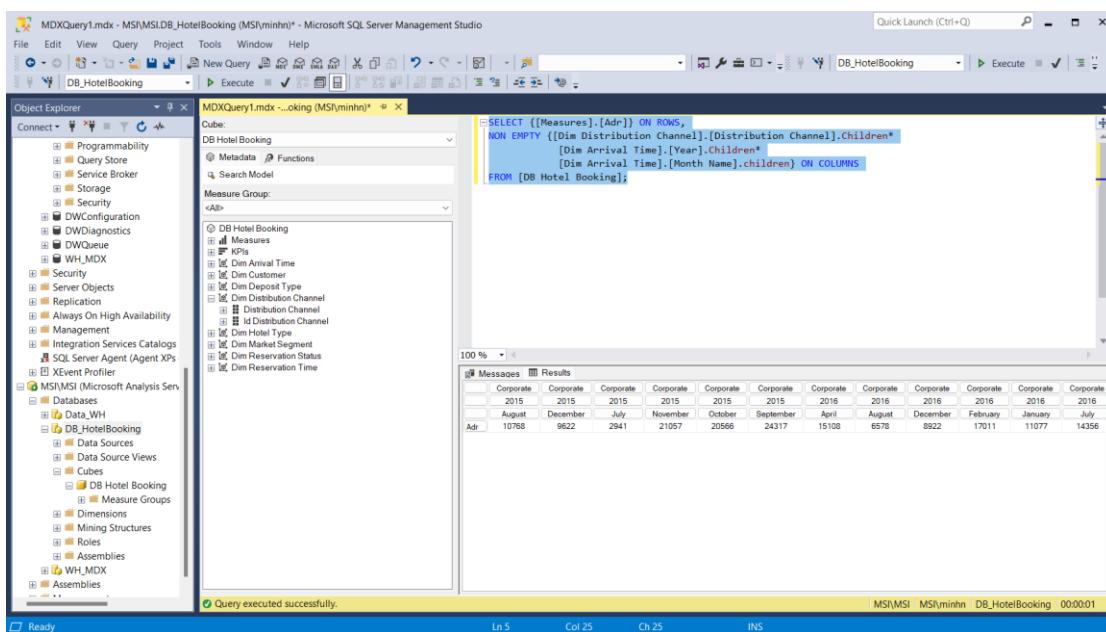


Figure 250. Kết quả MDX câu 3

- Công cụ Power BI

Kéo thả year từ bảng Dim_Year và month_name từ bảng Dim_Month vào trường Rows → Kéo thả distribution_channel từ Dim_Distribution_Channel vào trường Columns → Kéo thả adr từ bảng Fact vào trường Values.

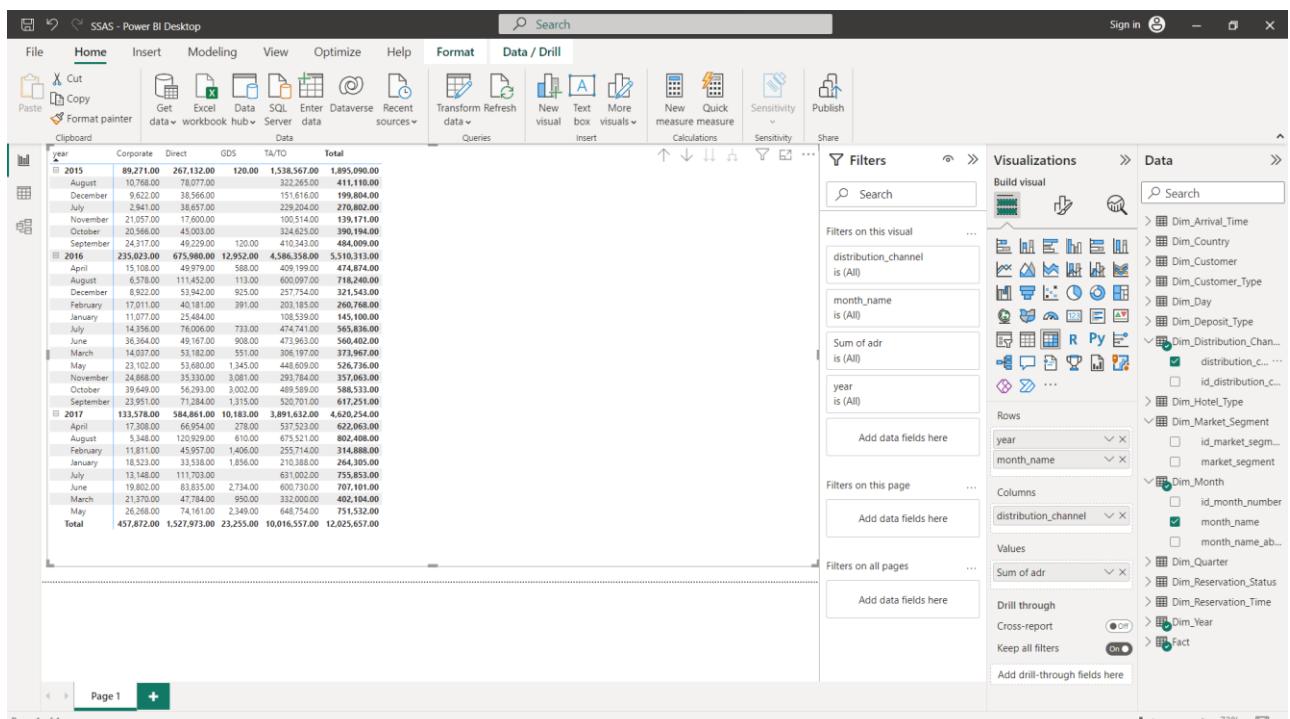


Figure 251. Kết quả Power BI câu 3

- BI Chart

Chọn kiểu Clustered Column Chart. Kéo thả thuộc tính *month_name* từ *Dim_Month* và *distribution_channel* từ *Dim_Distribution_Channel* vào X-axis và *year* từ *Dim_Year* vào Legend → Kéo thả thuộc tính *adr* từ bảng Fact vào Y-axis.

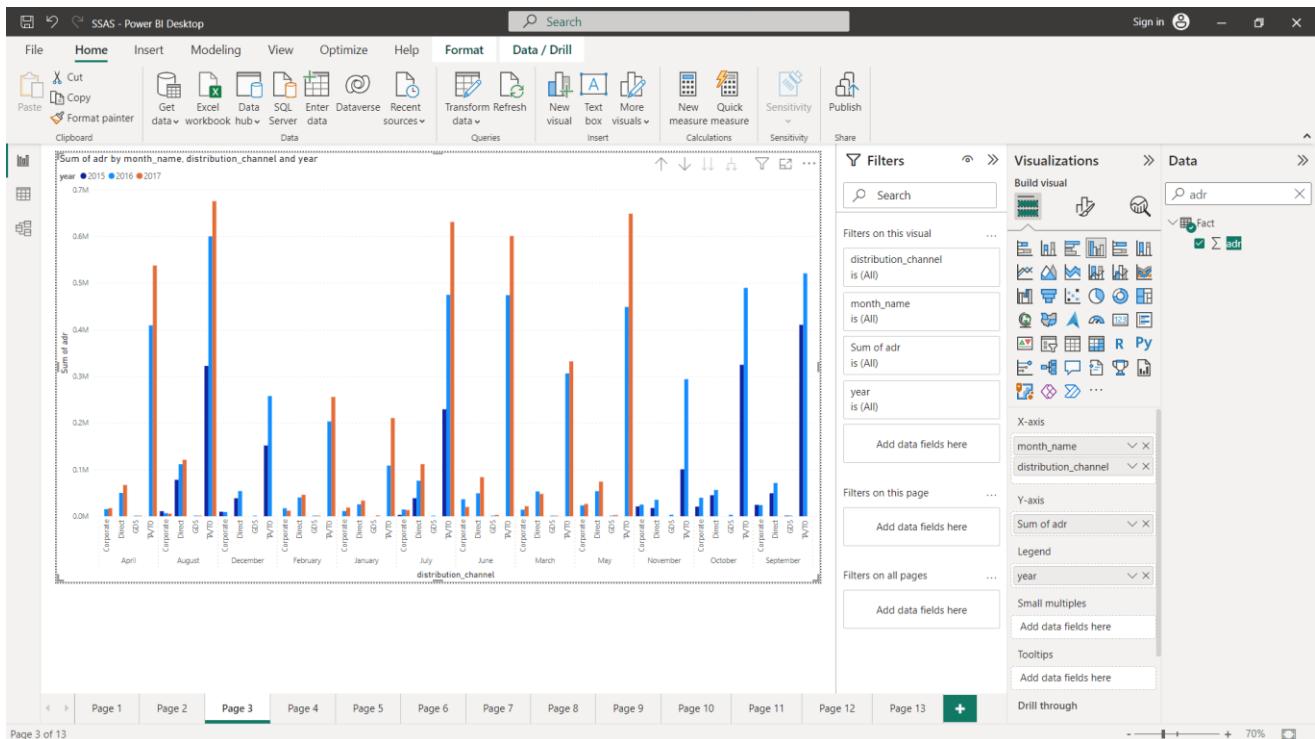


Figure 252. Kết quả trực quan Power BI câu 3

- Công cụ Excel

Kéo thả *year* từ bảng *Dim_Year* và *month_name* từ bảng *Dim_Month* vào trường Rows → Kéo thả *distribution_channel* từ *Dim_Distribution_Channel* vào trường Columns → Kéo thả *adr* từ bảng Fact vào trường Values.

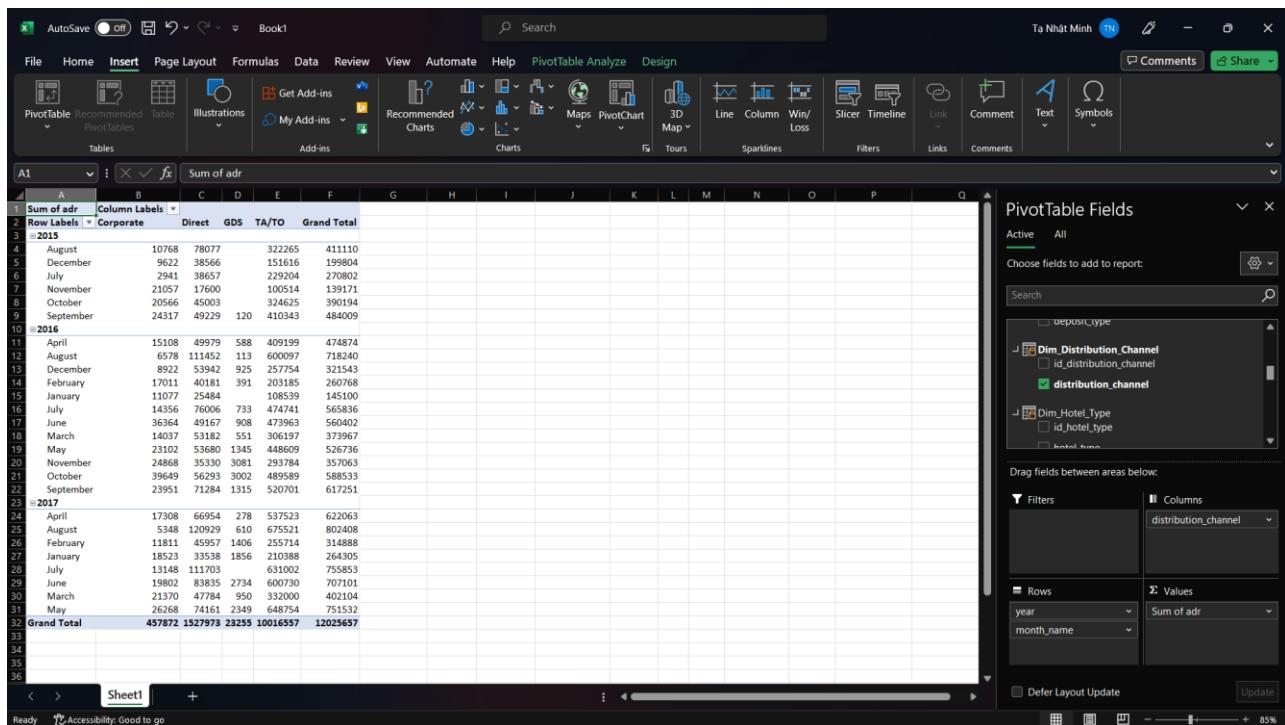
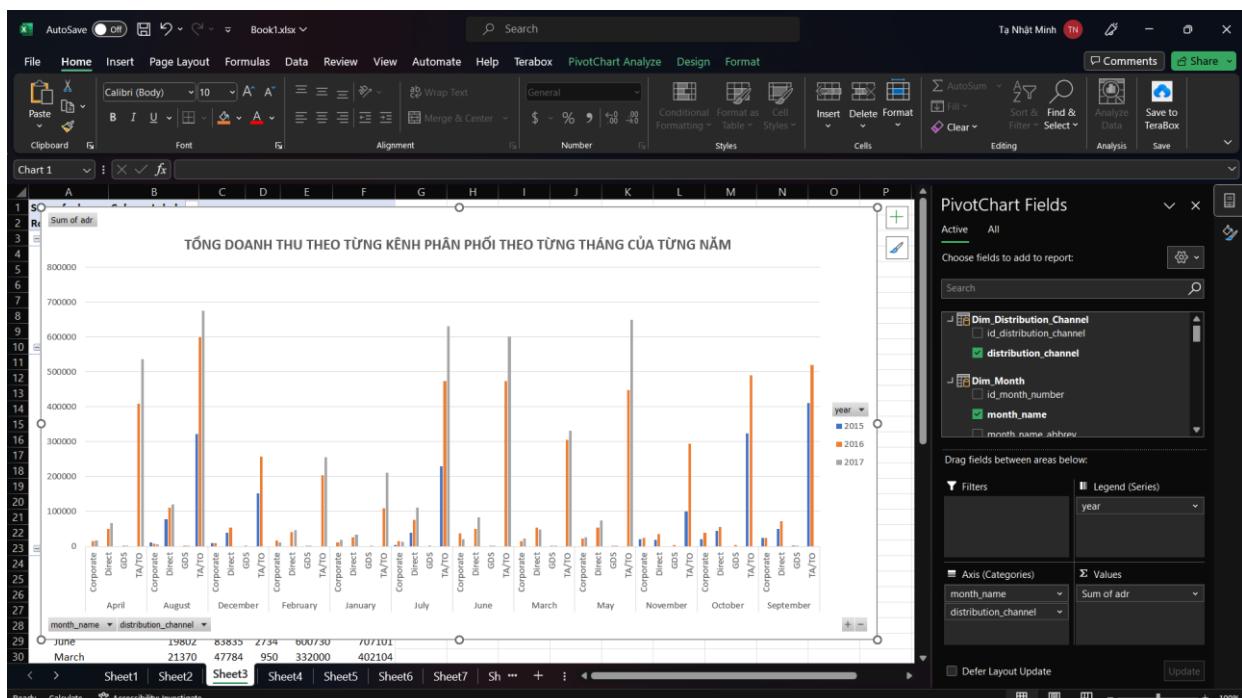


Figure 253. Kết quả Pivot Excel câu 3

- Pivot Chart Excel

Kéo thả thuộc tính year từ Dim_Year vào Legend và month_name từ Dim_Month vào Axis → Kéo thả thuộc tính adr từ bảng Fact vào Value.



4. Thống kê top 10 quốc gia có doanh thu cao nhất trong năm 2015 và 2016, sắp xếp các giá trị theo thứ tự giảm dần.

Ý nghĩa câu truy vấn: Cho phép người quản lý thống kê lượng doanh thu lớn tập trung ở những khách hàng thuộc những quốc gia nào, từ đó có chính sách đầu tư phù hợp cho khách sạn.

- **Công cụ SSAS trên các khối Cube**

Tạo Named Set có tên là [Cau 4] dùng để lấy top 10 quốc gia đem đến doanh thu cao nhất cho khách sạn.



Kéo thả Country từ Dim Customer vào vùng truy vấn → Kéo thả Adr từ Measure vào vùng truy vấn → Kéo thả Year từ Dim Arrival Time vào phần Filter, tại Filter Expression chọn 2015 để truy vấn cho năm 2015 và chọn 2016 để truy vấn cho năm 2016 → Chọn biểu tượng Design Mode để chuyển sang script mode. Chỉnh sửa câu truy vấn ban đầu → Chọn click to execute the query.

Country	Adr
AGO	5338
ALB	172
ARG	1301
ARM	49
AUS	2462
AUT	7956
AZE	225
BEL	20540
BGR	362
BLR	679
BRA	14448
BWA	118
CHE	13980
CHL	566
CHN	4503
CMR	354
CN	11029
COT	788

Figure 255. Kéo thả các trường cần dùng để truy vấn

```

SELECT NON EMPTY { [Measures].[Adr] } ON COLUMNS,
NON EMPTY { [Cau 4] } ON ROWS
FROM [DB Hotel Booking]
WHERE [Dim Arrival Time].[Year].&[2015]
CELL PROPERTIES VALUE,
BACK_COLOR,
FORE_COLOR,
FORMATTED_VALUE,
FORMAT_STRING,
FONT_NAME,
FONT_SIZE,
FONT_FLAGS

```

Figure 256. Câu truy vấn trong Script Mode sau chỉnh sửa

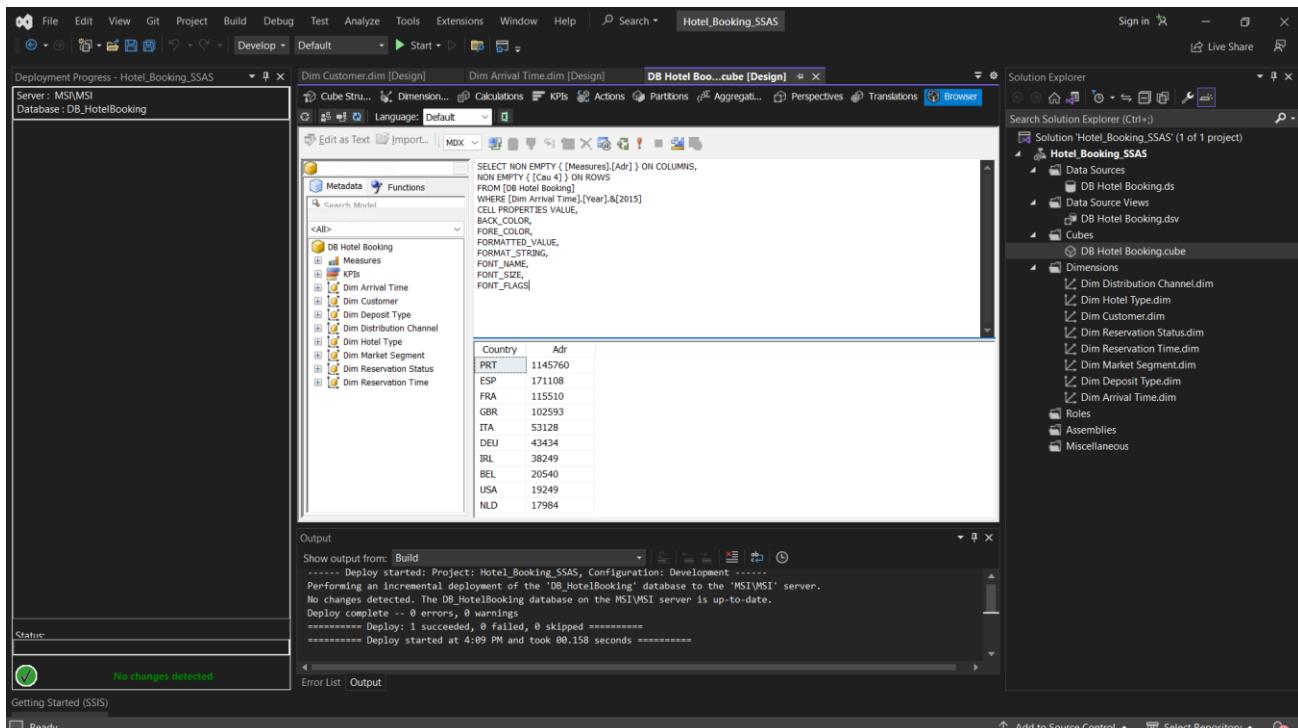


Figure 257. Kết quả SSAS câu 4 năm 2015

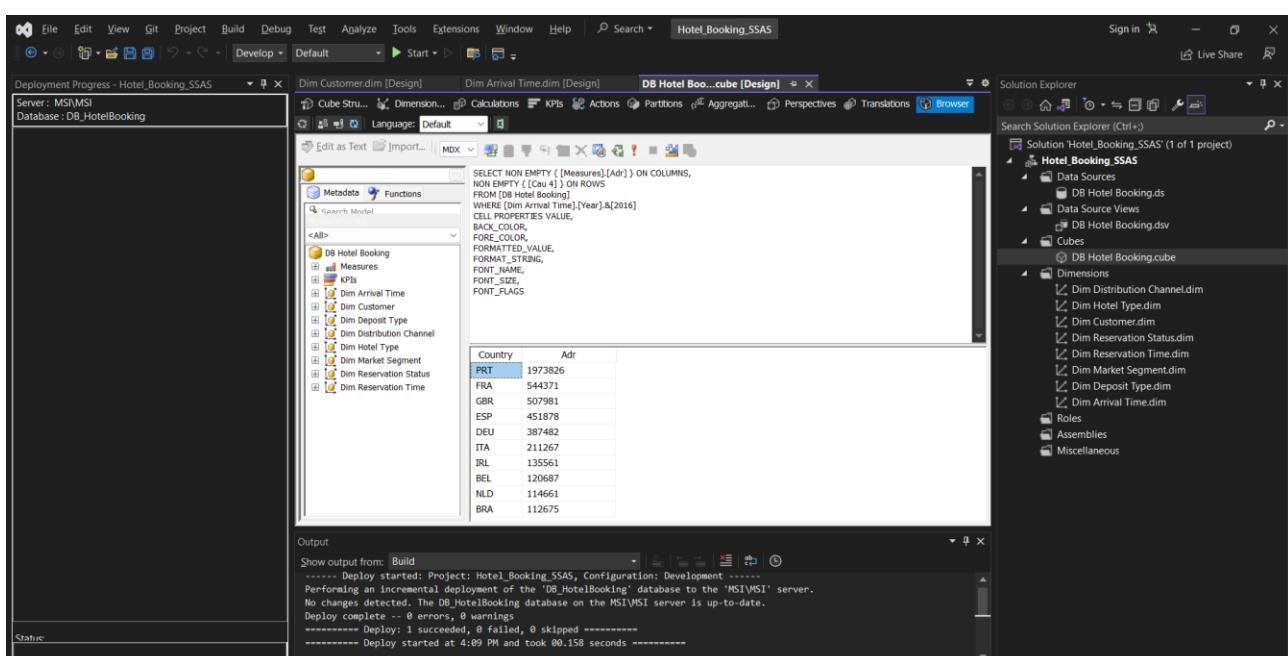


Figure 258. Kết quả SSAS câu 4 năm 2016

- **Ngôn ngữ MDX trên các khối Cube**

Query:

```
SELECT {[Measures].[Adr]} ON COLUMNS,
NON EMPTY {GENERATE(
    [Dim Customer].[Country].MEMBERS,
    TOPCOUNT([Dim Customer].[Country].children, 10, [Measures].[Adr])
)} ON ROWS
FROM [DB Hotel Booking]
WHERE [Dim Arrival Time].[Year].&[2016]
```

Kết quả:

Adr	Count
PRT	1145760
ESP	171108
FRA	115510
GBR	102953
ITA	53126
DEU	43434
IRL	32349
BEL	20540
USA	19249
NLD	17984

Figure 259. Kết quả MDX câu 4 năm 2015

Adr	Count
PRT	1973826
FRA	544371
GBR	507981
ESP	451878
DEU	387482
ITA	211267
IRL	135561
BEL	120687
NLD	114661
BRA	112675

Figure 260. Kết quả MDX câu 4 năm 2016

- Công cụ Power BI

Kéo thả year từ bảng Dim_Year vào trường Columns → Kéo thả country từ Dim_Country vào trường Rows → Kéo thả adr từ bảng Fact vào trường Values → Tại Filters year, chọn filter type là Basic Filtering, chọn giá trị 2015 để truy vấn top 10 năm 2015, và chọn 2016 để truy vấn top 10 của năm 2016 → Tại Filters country, chọn filter type Top N, set up số giá trị cần lấy top là 10, kéo thả adr vào ô By value → Sort giá trị tại cột Total theo thứ tự giảm dần.

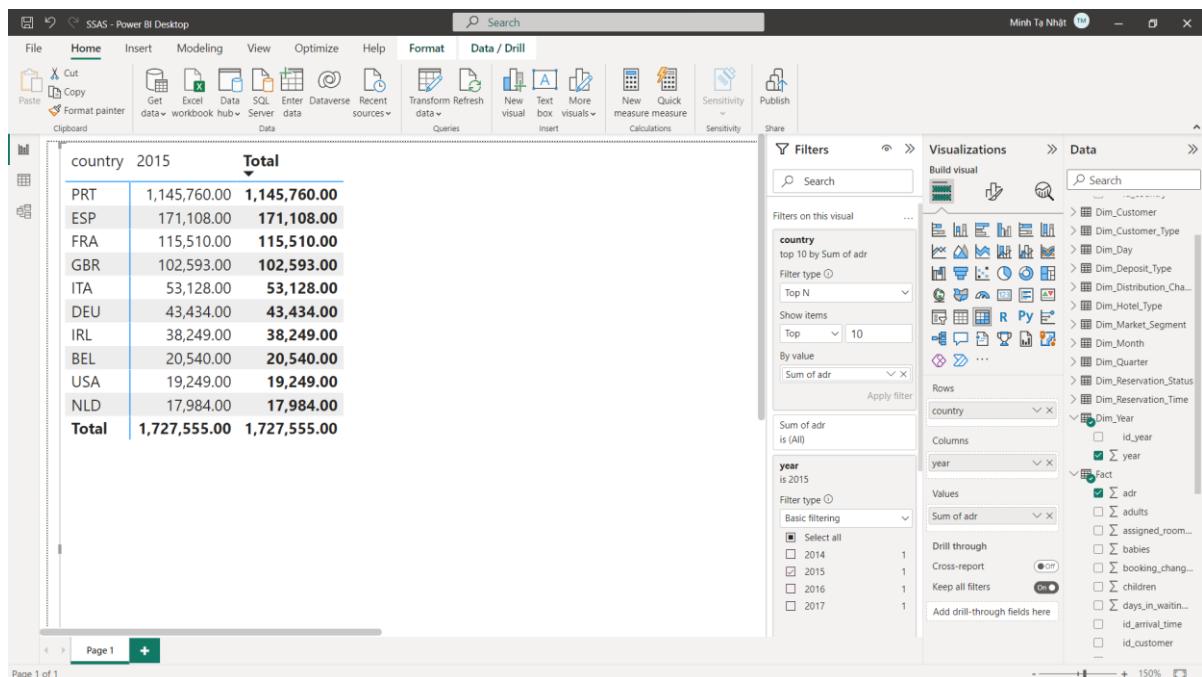


Figure 261. Kết quả Power BI câu 4 năm 2015

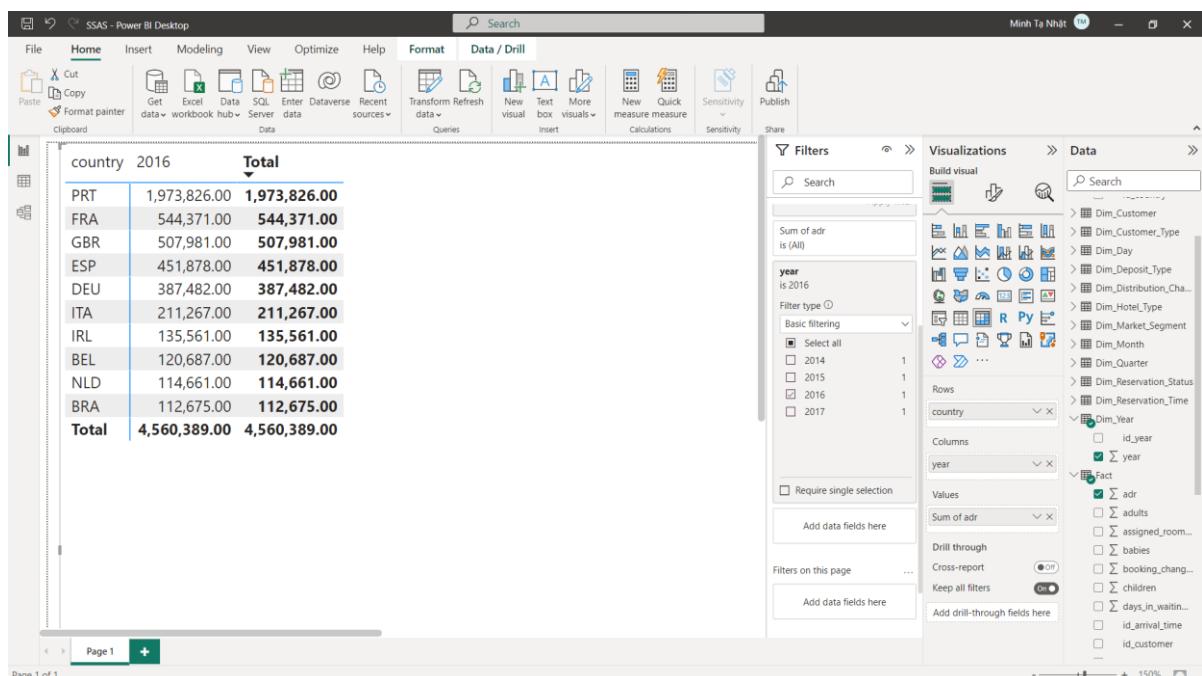


Figure 262. Kết quả Power BI câu 4 năm 2016

- BI Chart

Chọn kiểu Clustered Column Chart. Kéo thả thuộc tính year từ Dim_Year vào Legend và country từ Dim_Country vào X-axis → Kéo thả thuộc tính adr từ bảng Fact vào Y-axis. Tại filter chọn Basic Filtering vào chọn lần lượt giá trị 2015 và 2016 để thu được 2 biểu đồ tương ứng của 2 năm.

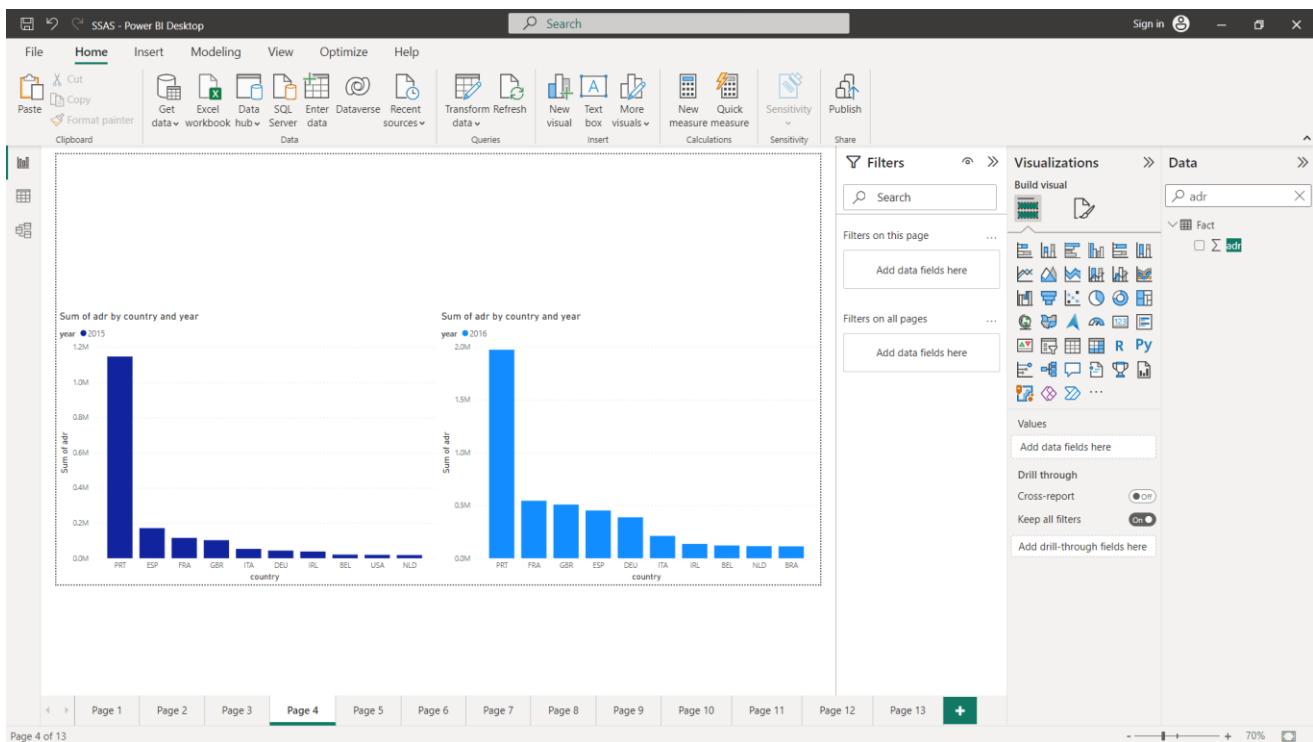


Figure 263. Kết quả trực quan Power BI câu 4

- Công cụ Excel

Kéo thả year từ bảng Dim_Year vào trường Filters → Kéo thả country từ Dim_Country vào trường Rows → Kéo thả adr từ bảng Fact vào trường Values → Tại Filter year, chọn giá trị 2015 để truy vấn top 10 năm 2015, và chọn 2016 để truy vấn top 10 của năm 2016. Tại Row Labels của country, chọn Filter Top 10 → Sort giá trị tại cột Sum of adr theo thứ tự giảm dần.

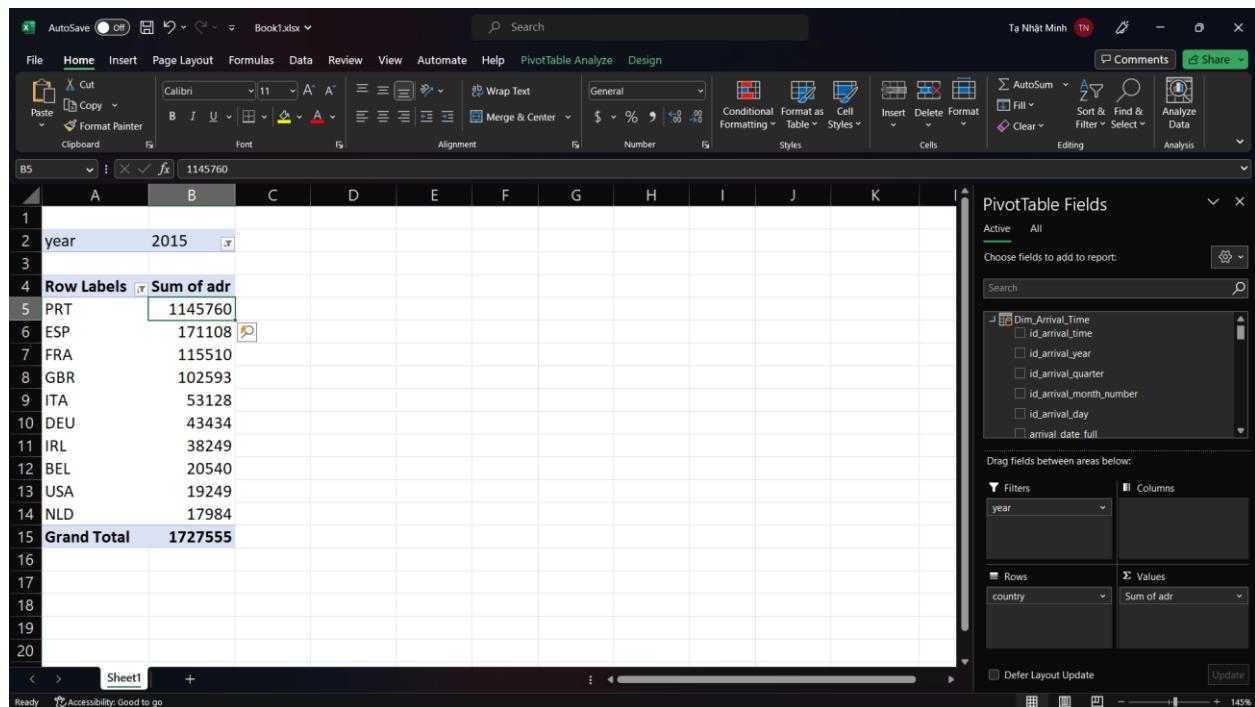


Figure 264. Kết quả Pivot Excel câu 4 năm 2015

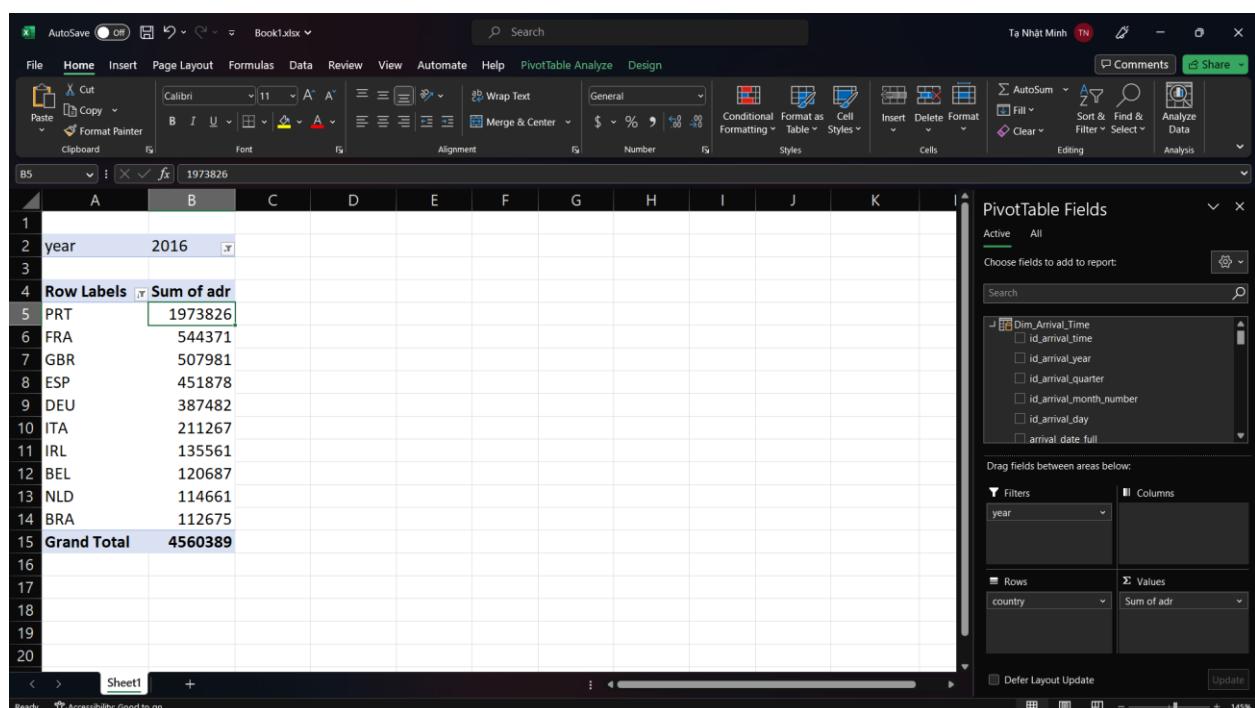


Figure 265. Kết quả Pivot Excel câu 4 năm 2016

- Pivot Chart Excel

Kéo thả thuộc tính *year* từ *Dim_Year* vào *Filters* và *country* từ *Dim_Country* vào *Legend* → Kéo thả thuộc tính *adr* từ bảng *Fact* vào *Value*. Tại *filter year* chọn lần lượt 2015 và 2016 để có được 2 biểu đồ top 10 của năm 2015 và 2016.

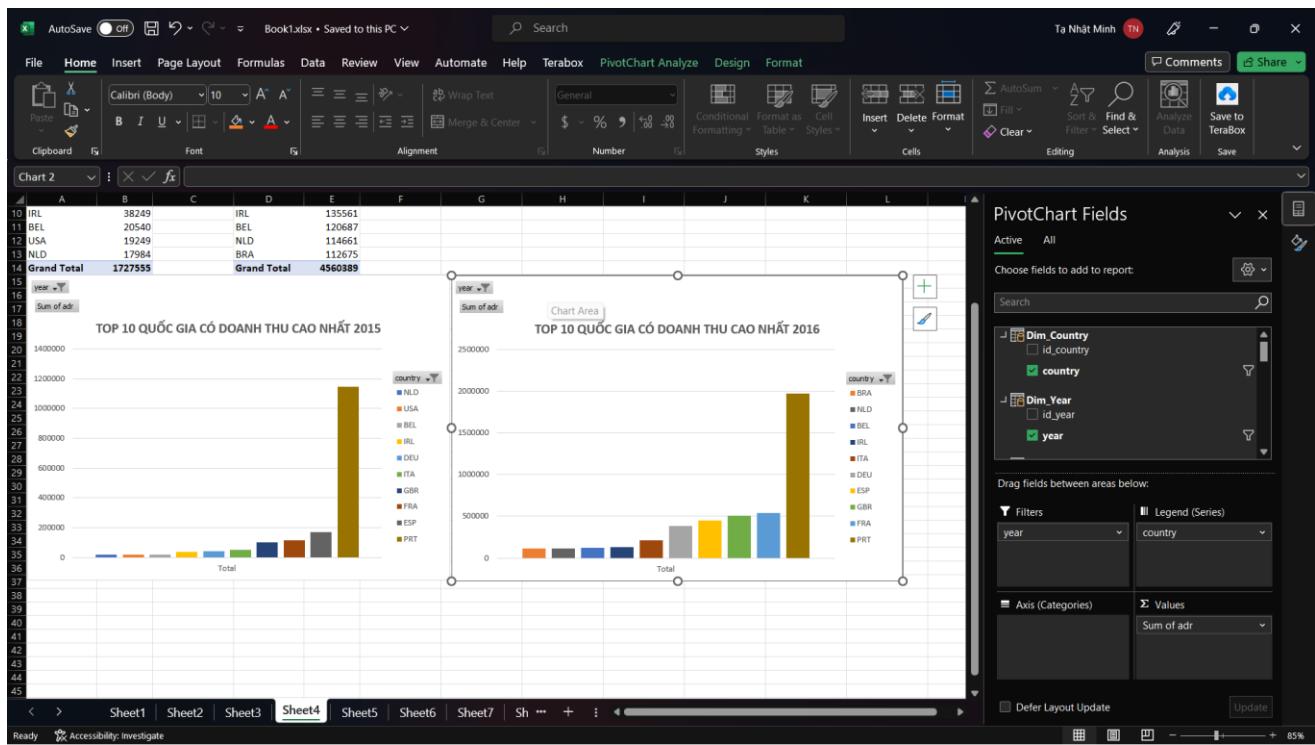


Figure 266. Kết quả Excel Pivot Chart câu 4

5. Thống kê tổng thời gian chờ của khách hàng (lead_time) theo từng Quốc gia theo từng tháng của năm.

Ý nghĩa câu truy vấn: Thu thập thông tin về thời gian chờ để phục vụ cho quá trình phân tích tại sao tháng đó lại có lead_time cao, từ đó suy ra được khoảng thời gian mà khách hàng đến khách sạn nhiều và đưa ra kế hoạch kinh doanh phù hợp.

- Công cụ SSAS trên các khối Cube

Kéo thả Year và Month Name từ Dim Arrival Time sang khung truy vấn → Kéo thả Country từ Dim Customer sang khung truy vấn → Kéo thả Lead Time từ Measure Fact sang khung truy vấn → Chọn click to execute the query.

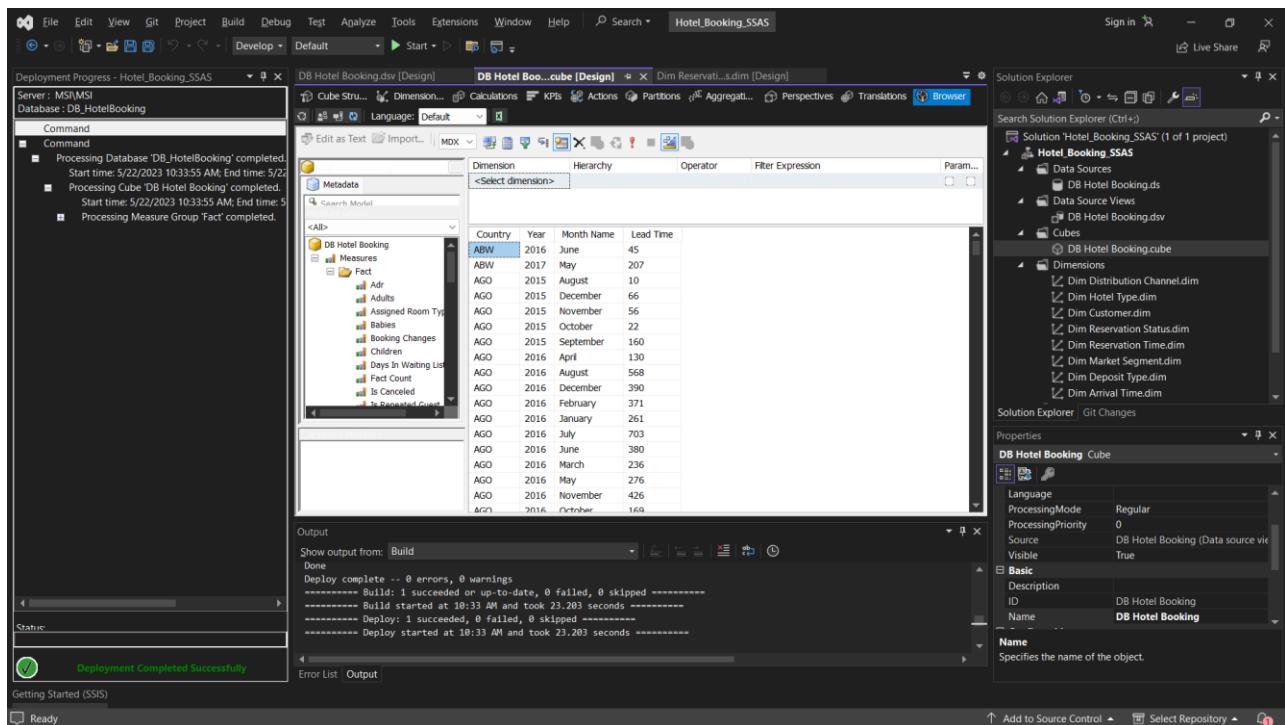


Figure 267. Kết quả SSAS câu 5

- Ngôn ngữ MDX trên các khối Cube

Query:

```
SELECT {[Measures].[Lead Time]} ON ROWS,
NON EMPTY {[Dim Arrival Time].[Year].children* [Dim Arrival Time].[Month Name].children} ON COLUMNS
FROM [DB Hotel Booking]
```

Kết quả:

Lead Time	2015	2015	2015	2015	2015	2015	2016	2016	2016	2016	2016	2016	2016	2016	2016
Lead Time	386525	140830	349584	109308	507567	629729	457985	612452	339345	137542	73078	559340	624912	267584	628092

Figure 268. Kết quả MDX câu 5

- Công cụ Power BI

Kéo thả year từ bảng Dim_Year và country từ Dim_Country vào trường Rows
 → Kéo thả month_name từ bảng Dim_Month vào trường Columns → Kéo thả lead_time từ bảng Fact vào trường Values.

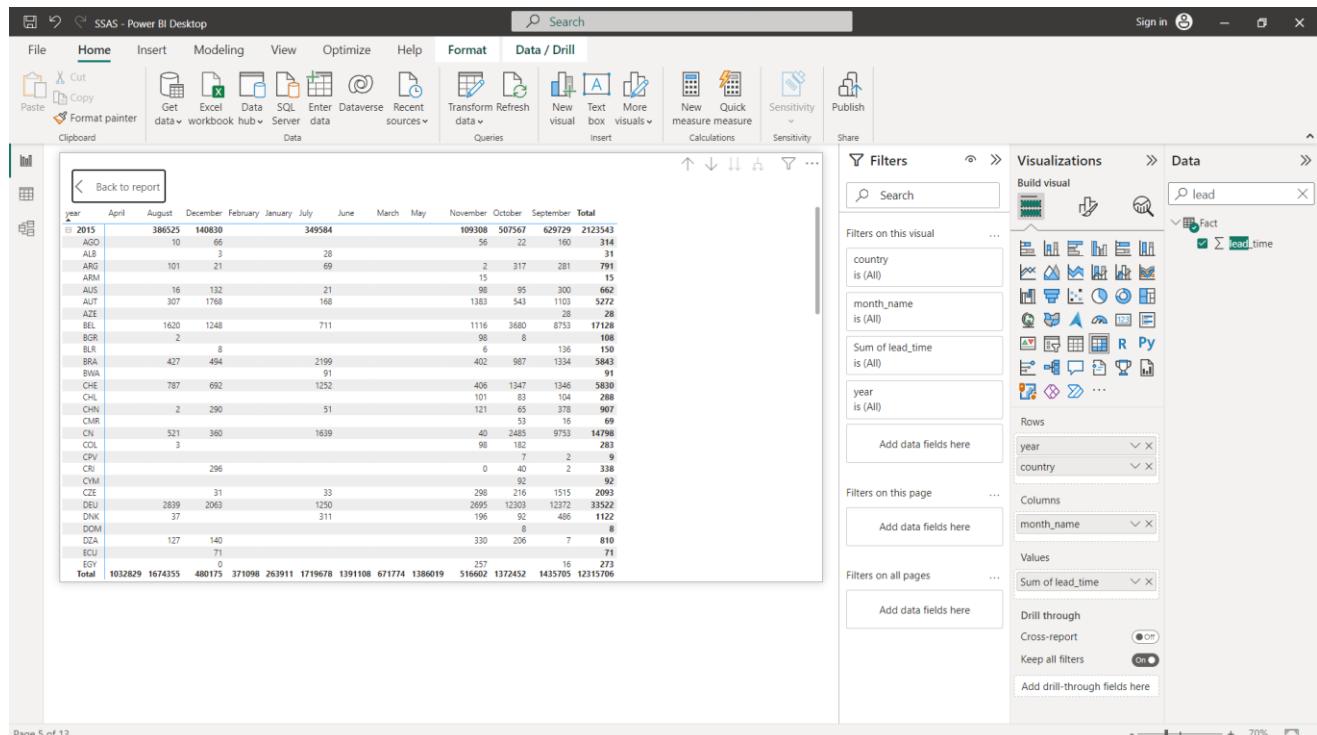


Figure 269. Kết quả Power BI câu 5

- BI Chart

Chọn kiểu Line Chart. Kéo thả thuộc tính year từ Dim_Year và month_name từ Dim_Month vào X-axis và country từ Dim_Country vào Legend → Kéo thả thuộc tính lead_time từ bảng Fact vào Y-axis.

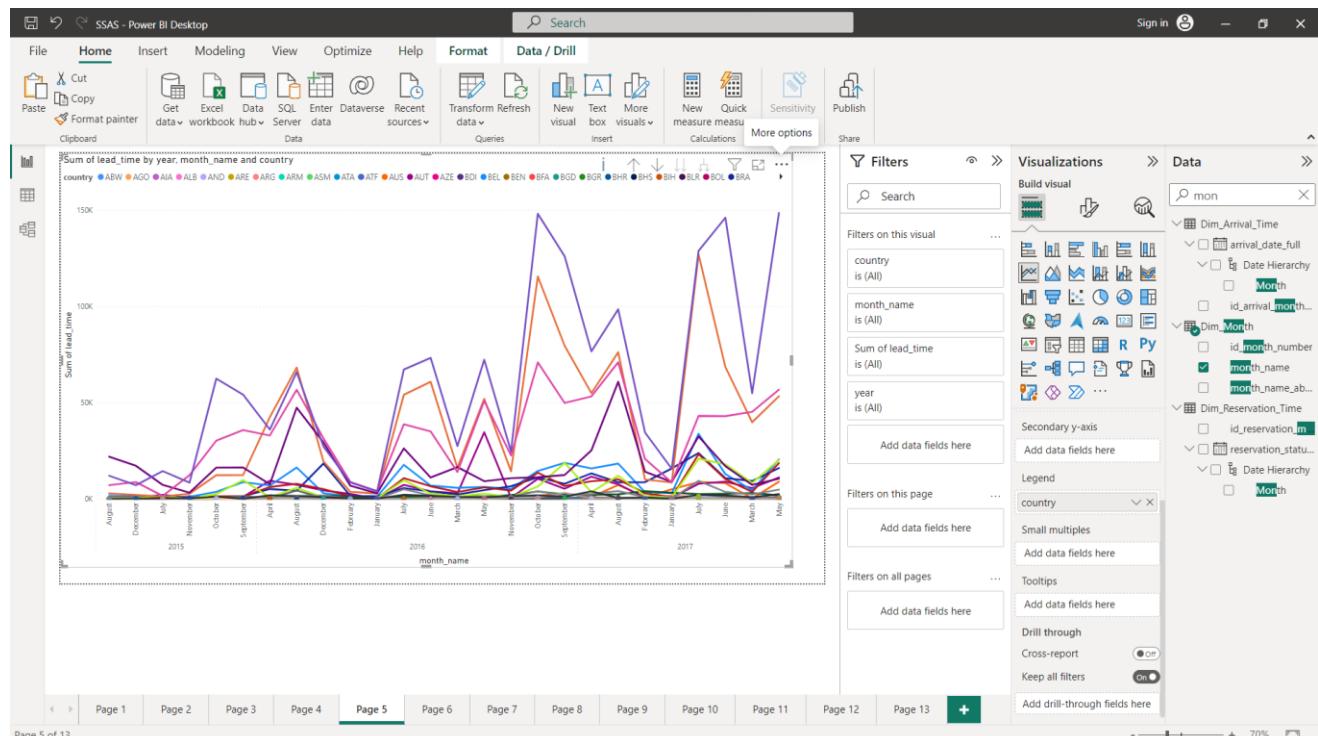


Figure 270. Kết quả trực quan Power BI câu 5

- Công cụ Excel

Kéo thả year từ bảng Dim_Year và month_name từ bảng Dim_Month vào trường Columns → Kéo thả country từ Dim_Country vào trường Rows → Kéo thả lead_time từ bảng Fact vào trường Values.

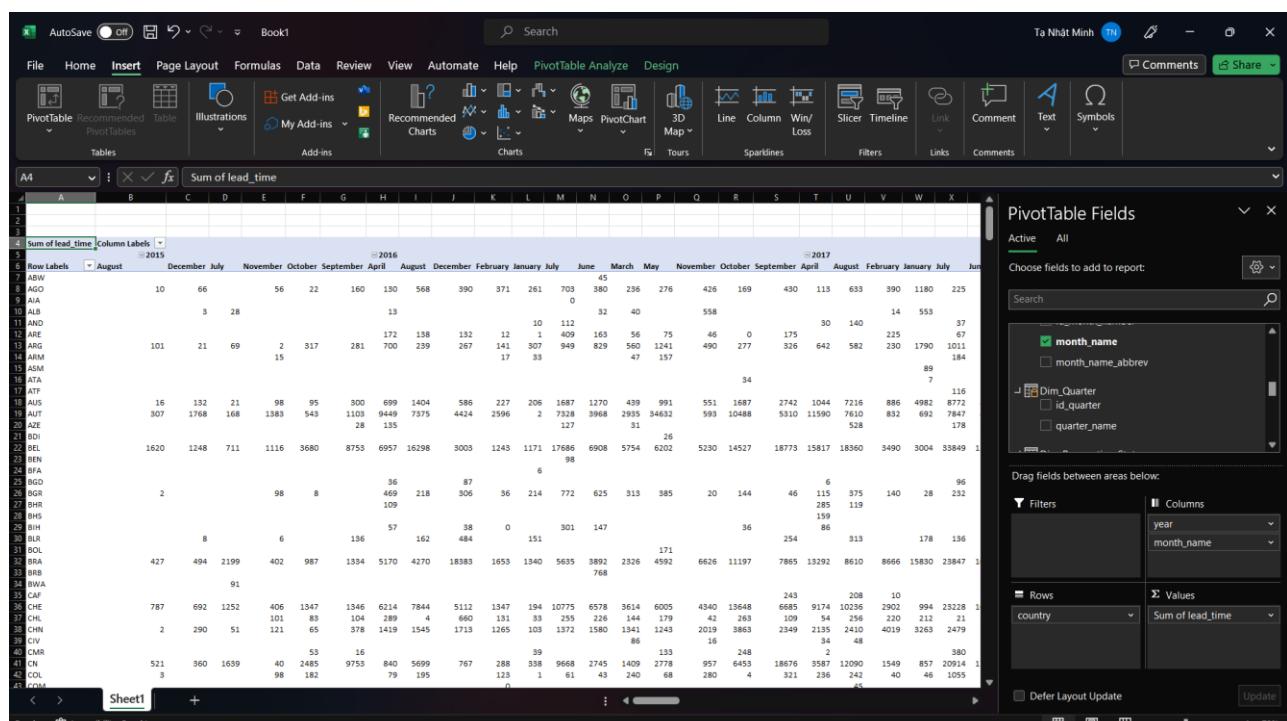


Figure 271. Kết quả Pivot Excel câu 5

- **Pivot Chart Excel**

Kéo thả thuộc tính *year* từ *Dim_Year* và *month_name* từ *Dim_Month* vào Axis và *country* từ *Dim_Country* vào Legend → Kéo thả thuộc tính *lead_time* từ bảng Fact vào Values.

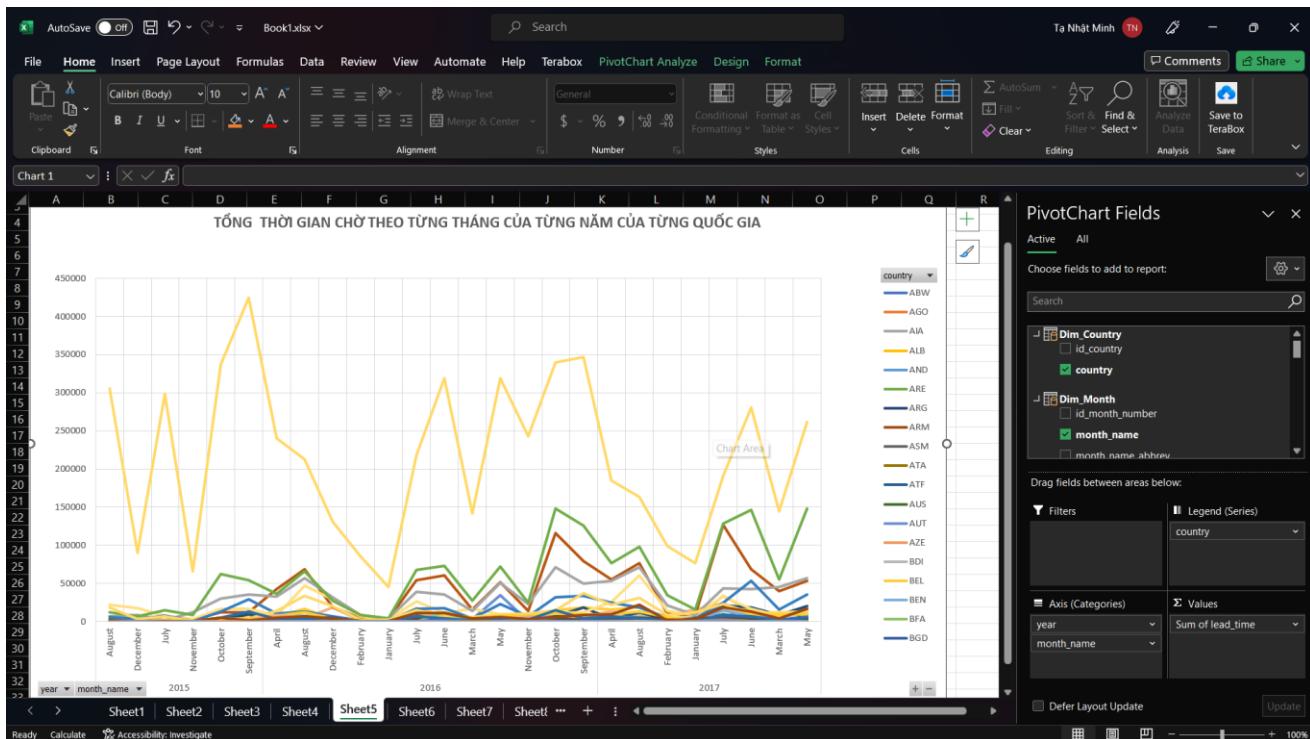


Figure 272. Kết quả Excel Pivot Chart câu 5

6. Thống kê các quốc gia tạo ra tổng doanh thu lớn hơn 50000

Ý nghĩa câu truy vấn:

Thống kê các quốc gia tạo ra doanh thu lớn nhằm đánh giá tiềm năng kinh doanh của quốc gia đó.

- **Công cụ SSAS trên các khối Cube**

Tạo Named Set có tên là [Câu 11] dùng để lấy các quốc gia có tổng doanh thu lớn hơn 50000.

Name:

Expression

```
FILTER([Dim Customer].[Country].children,
      [Measures].[Adr] > 50000)
```

✓ No issues found

Ln: 2 Ch: 34 SPC

Figure 273. Tạo Named Set câu 6

Kéo thả Country từ Dim Customer vào vùng truy vấn → Kéo thả Adr từ Measure vào vùng truy vấn. Chọn biểu tượng Design Mode để chuyển sang script mode. Chỉnh sửa câu truy vấn ban đầu → Chọn click to execute the query.

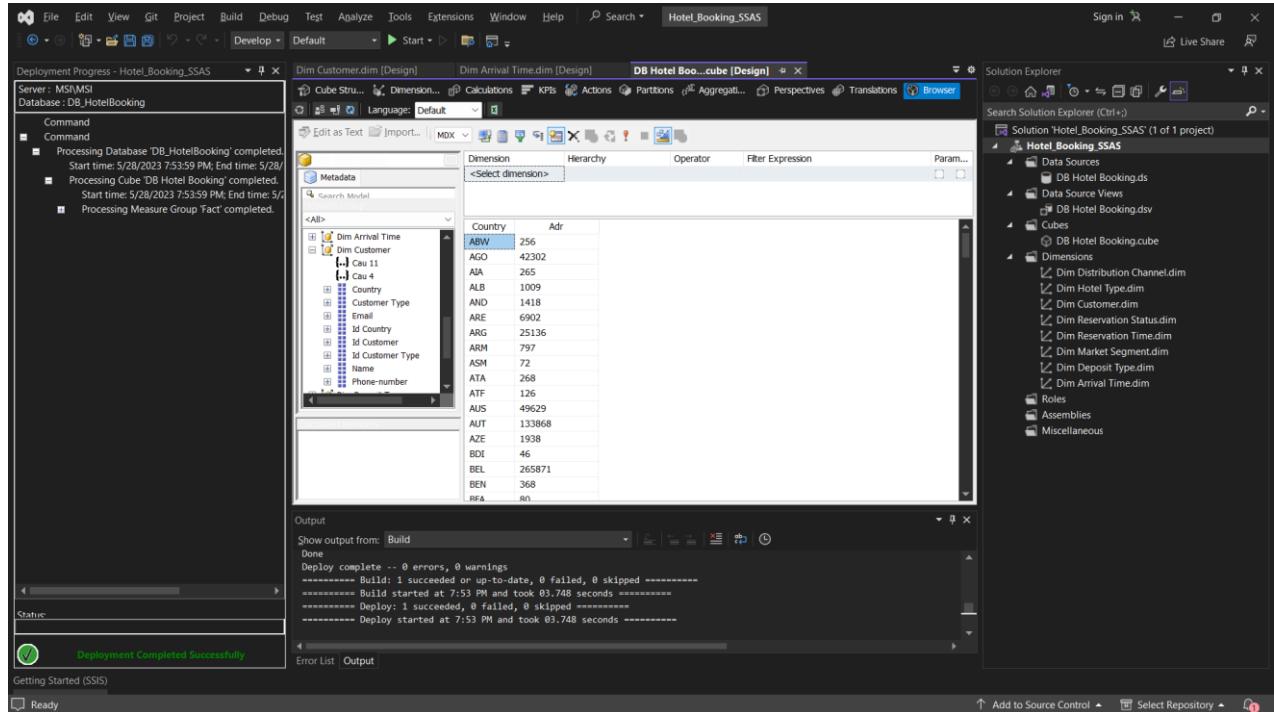


Figure 274. Kéo thả các trường cần dùng để truy vấn

```
SELECT NON EMPTY { [Measures].[Adr] } ON COLUMNS,
NON EMPTY { [Cau 11] } ON ROWS
FROM [DB Hotel Booking]
```

Figure 275. Câu truy vấn trong Script Mode sau chỉnh sửa

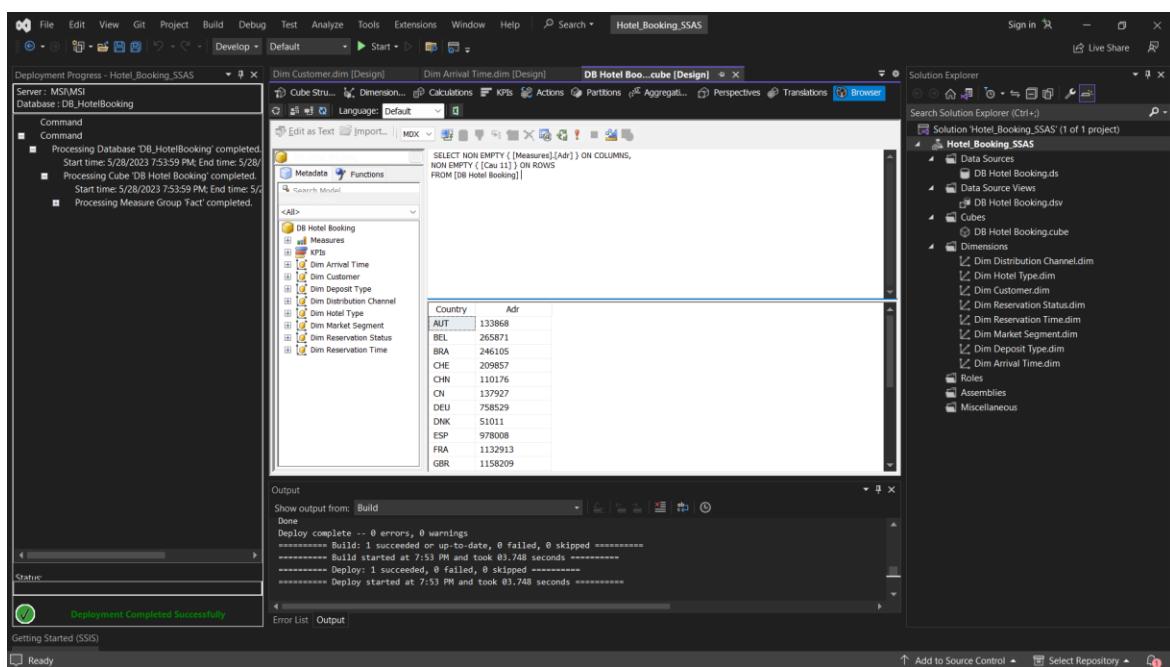


Figure 276. Kết quả SSAS câu 6

- **Ngôn ngữ MDX trên các khối Cube**

Query:

```
SELECT NON EMPTY {[Measures].[Adr]} ON COLUMNS,
NON EMPTY {[Dim Customer].[Country].children}
HAVING [Measures].[Adr] > 50000 ON ROWS
FROM [DB Hotel Booking]
```

Kết quả:

Country	Sum of adr
AUT	133688
BEL	266571
BRA	246105
CHE	209857
CHN	110176
CN	137927
DEU	758529
DNK	51011
ESP	978008
FRA	1132913
GBR	1000000
IRL	330217
ISR	73904
ITA	427925
NLD	226801
NOR	73638
POL	98201
PRT	4426148

Figure 277. Kết quả MDX câu 6

- **Công cụ Power BI**

Kéo country từ Dim_Country vào trường Columns → Kéo thả adr từ Fact vào Values → Tại Filters Sum of adr, chọn is greater than với tham số là 50000 → Chọn Apply filter.

country	Sum of adr
PRT	4,426,148.00
GBR	1,158,209.00
FRA	1,132,913.00
ESP	978,008.00
DEU	758,529.00
ITA	427,925.00
IRL	330,217.00
BEL	265,871.00
USA	257,103.00
BRA	246,105.00
NLD	226,801.00
CHE	209,857.00
CN	137,927.00
AUT	133,868.00
SWE	111,833.00
CHN	110,176.00
POL	98,201.00
RUS	74,865.00
ISR	73,904.00
NOR	73,638.00
ROU	57,284.00
DNK	51,011.00
Total	11,340,393.00

Figure 278. Kết quả Power BI câu 6

7. Thống kê tổng thời gian chờ của từng phân khúc thị trường theo từng tháng, quý, năm.

Ý nghĩa câu truy vấn:

Dựa vào thống kê thời gian chờ này khách sạn có thể có những dự trữ, sắp xếp các hoạt động đặt phòng một cách tối ưu nhất.

- Công cụ SSAS trên các khối Cube

Kéo thả Year, Quarter Name, Month Name từ Dim Arrival Time và Market Segment từ Dim Market Segment vào vùng truy vấn → Kéo thả Lead Time từ Measure vào vùng truy vấn. Chọn click to execute the query.

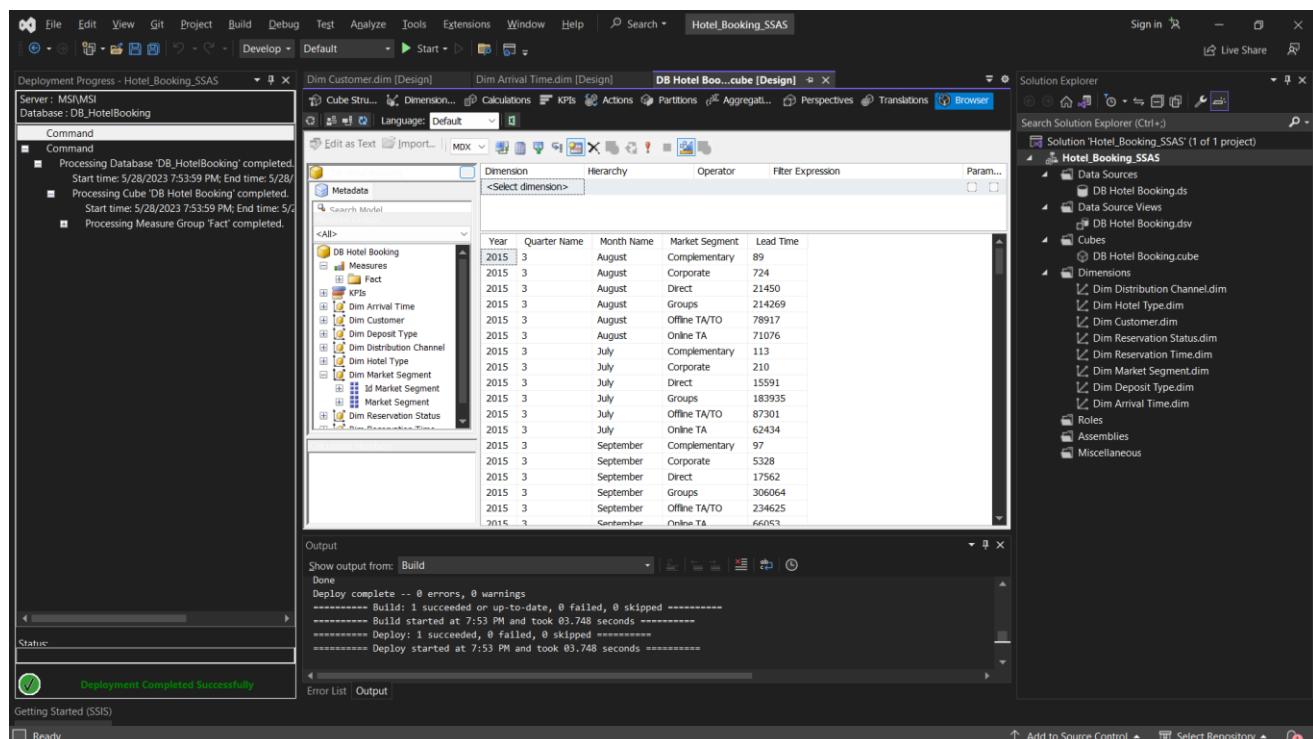


Figure 279. Kéo thả các trường cần dùng để truy vấn

- Ngôn ngữ MDX trên các khối Cube

Query:

```
select non empty {[Dim Market Segment].[Market Segment].children*[Measures].[Lead Time]} on columns,
non empty {[Dim Arrival Time].[Year].children*
[Dim Arrival Time].[Quarter Name].children*[Dim Arrival Time].[Month Name].children}
on rows
from [DB Hotel Booking]
```

Kết quả:

The screenshot shows the Microsoft SQL Server Management Studio interface. In the Object Explorer, there is a connection to 'MSI\MSI (Microsoft Analysis Server)' containing databases like 'Data_WH', 'DB_HotelBooking', and 'WH_MDX'. The 'MDXQuery2.mdx' query window displays an MDX query and its results. The query retrieves data from the 'DB Hotel Booking' cube, specifically focusing on market segments, arrival times, years, quarters, and months. The results grid shows various metrics such as Lead Time, Arrival Date, and Total Requests for specific dates in 2015 and 2016.

Lead Time	Arrival Date	Total Requests
August	2015	1084
July	2015	299
September	2015	113
December	2015	785
November	2015	163
October	2015	285
February	2016	14
January	2016	0
March	2016	123
April	2016	144
June	2016	153
May	2016	227
August	2016	0
July	2016	123
September	2016	91
December	2016	1153
January	2017	3084
March	2017	41

Figure 280. Kết quả MDX câu 7

- Công cụ Power BI

Kéo market_segment từ Dim_Market_Segment vào trường Columns → Kéo thả lead_time từ Fact vào Values → Kéo thả year từ Dim_Year, quarter_name từ Dim_Quarter và month_name từ Dim_Month vào trường Rows.

The screenshot shows the Power BI Desktop interface. A data grid displays a table with columns for year, Aviation, Complementary, Corporate, Direct, Groups, Offline TA/TO, Online TA, and Total. The rows are grouped by year (2015, 2016) and month (e.g., August, September). To the right of the grid, the 'Visualizations' pane is open, showing various chart and report options. The 'Data' pane on the far right lists all available fields from the data source, including market_segment, lead_time, year, id_quarter, and month_name.

Figure 281. Kết quả Power BI câu 7

8. Thống kê số lượng đặt phòng của khách hàng đến từ Bồ Đào Nha (PRT) thuộc từng phân khúc thị trường theo từng quý của từng năm.

Ý nghĩa câu truy vấn:

Phân tích phân khúc thị trường nào từ Bồ Đào Nha được khách hàng ưa chuộng để tập trung phát triển.

- Công cụ SSAS trên các khối Cube

Kéo thả Year, Quarter Name từ Dim Arrival Time và Market Segment từ Dim Market Segment vào vùng truy vấn → Kéo thả Country từ Dim Customer vào bộ lọc với giá trị lọc là ‘PRT’ → Kéo thả Adr và Fact Count từ Measure vào vùng truy vấn. Chọn click to execute the query.

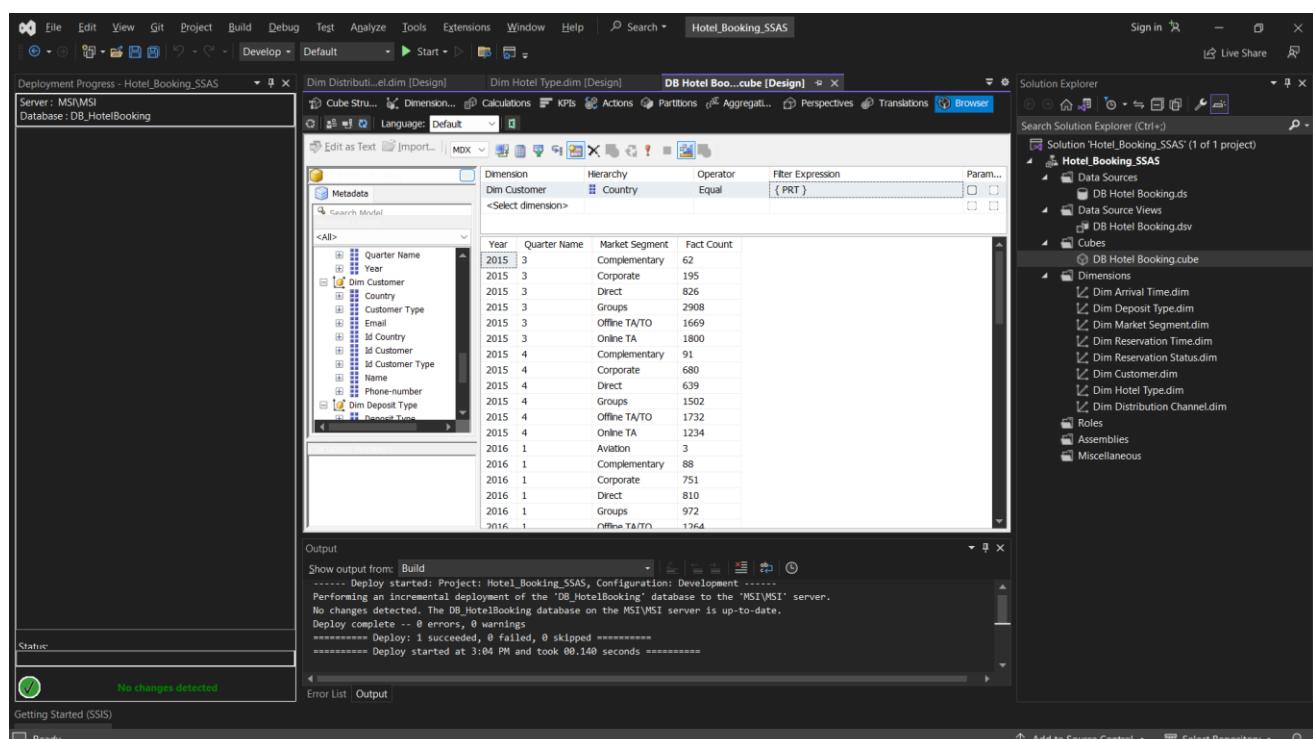


Figure 282. Kéo thả các trường cần dùng để truy vấn

- Ngôn ngữ MDX trên các khối Cube

Query:

```

SELECT NON EMPTY { [Measures].[Fact Count] } ON COLUMNS,
NON EMPTY { ([Dim Arrival Time].[Year].children *
[Dim Arrival Time].[Quarter Name].children *
[Dim Market Segment].[Market Segment].children ) } ON ROWS
FROM [DB Hotel Booking]
WHERE ( [Dim Customer].[Country].&[PRT] )

```

Kết quả:

IS217.N21 – Kho dữ liệu và phân tích hoạt động đặt phòng khách sạn

The screenshot shows the SSMS interface with the following details:

- File Bar:** File, Edit, View, Query, Project, Tools, Window, Help.
- Toolbar:** Standard toolbar with various icons for file operations, search, and execution.
- Object Explorer:** Shows the connection to MSIMSI (Microsoft Analysis Server) and the DB_HotelBooking cube.
- MDX Query Editor:** Displays the MDX query:


```
SELECT NON EMPTY { [Measures].[Fact Count] } ON COLUMNS,
      NON EMPTY { ([Dim Arrival Time].[Year].children *
      [Dim Arrival Time].[Quarter Name].children *
      [Dim Market Segment].[Market Segment].children ) } ON ROWS
      FROM [DB Hotel Booking]
      WHERE { [Dim Customer].[Country].&[PRT] }
```
- Results Grid:** A table titled "Fact Count" showing the following data:

Year	Quarter	Market Segment	Fact Count
2015	3	Complementary	62
2015	3	Corporate	195
2015	3	Direct	826
2015	3	Groups	2908
2015	3	Offline TATO	1669
2015	3	Online TA	1800
2015	4	Complementary	91
2015	4	Corporate	600
2015	4	Direct	639
2015	4	Groups	1502
2015	4	Offline TATO	1732
2015	4	Online TA	1234
2016	1	Avalon	3
2016	1	Complementary	88
2016	1	Corporate	751
2016	1	Direct	810
2016	1	Groups	972
2016	1	Offline TATO	1264
- Status Bar:** Query executed successfully, MSI\MSI MSI\minhn DB_HotelBooking 00:00:01.

Figure 283. Kết quả MDX câu 8

- Công cụ Power BI

Kéo `market_segment` từ `Dim_Maret_Segment` và `quarter_name` từ `Dim_Quarter` vào trường Rows → Kéo thả `id_fact (Count)` từ `Fact` vào Values → Kéo thả `year` từ `Dim_Year`, `country` từ `Dim_Country` vào trường Columns. Tại bộ lọc `country` chọn giá trị lọc là 'PRT'.

The screenshot shows the Power BI Desktop interface with the following details:

- File Bar:** File, Home, Insert, Modeling, View, Optimize, Help.
- Toolbar:** Standard toolbar with various icons for file operations, search, and execution.
- Report View:** A matrix visual showing fact counts for different market segments and years. The matrix has columns for 2015, 2016, 2017, and Total. The rows are grouped by market segment (Aviation, Complementary, Corporate, Direct, Groups, Offline TA/TO).
- Filter Pane:** Located on the right side, it shows the following filters applied:
 - Count of id_fact is (All)
 - country is PRT
 - market_segment is (All)
 - quarter_name is (All)
 - year is (All)
- Page Footer:** Page 12 of 12.

Figure 284. Kết quả Power BI câu 8

9. Thống kê doanh thu theo từng kênh phân phối của từng phân khúc thị trường của từng quốc gia.

Ý nghĩa câu truy vấn:

Phân tích sự khác biệt giữa các phân khúc thị trường và các kênh phân phối tại các quốc gia khác nhau, đưa ra các giải pháp để tăng doanh thu cho khách sạn tại từng quốc gia.

- **Công cụ SSAS trên các khối Cube**

Kéo thả Country từ Dim Country, Market Segment từ Dim Market Segment và Distribution Channel từ Dim Distribution Channel vào vùng truy vấn → Kéo thả Adr từ Measure vào vùng truy vấn. Chọn click to execute the query.

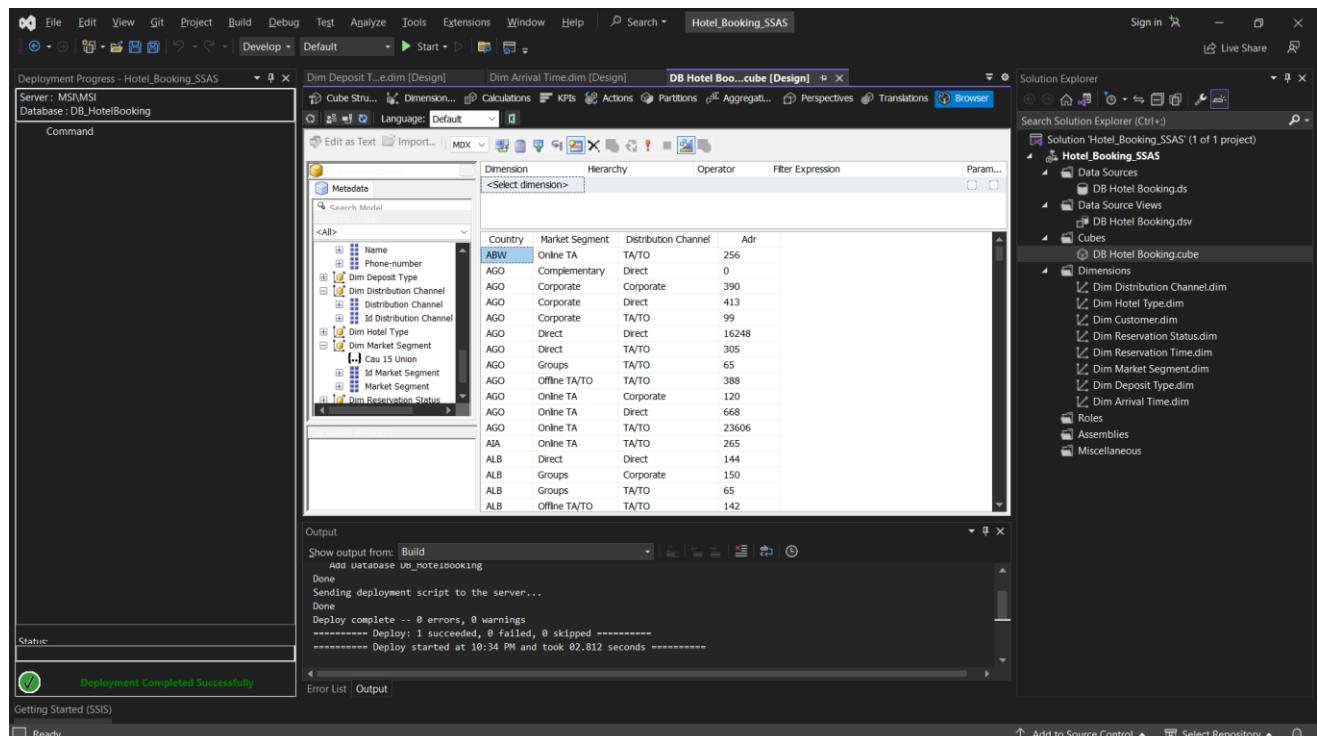


Figure 285. Kéo thả các trường cần dùng để truy vấn

- **Ngôn ngữ MDX trên các khối Cube**

Query:

```

SELECT NON EMPTY { [Measures].[Adr] } ON COLUMNS,
NON EMPTY { ([Dim Customer].[Country].children *
[Dim Market Segment].[Market Segment].children *
[Dim Distribution Channel].[Distribution Channel].children ) } ON ROWS
FROM [DB Hotel Booking]
    
```

Kết quả:

IS217.N21 – Kho dữ liệu và phân tích hoạt động đặt phòng khách sạn

```

SELECT NON EMPTY { [Measures].[Adr] } ON COLUMNS,
NON EMPTY { {[Dim Customer].[Country].children *
[Dim Market Segment].[Market Segment].children *
[Dim Distribution Channel].[Distribution Channel].children } } ON ROWS
FROM [DB Hotel Booking]
  
```

ADR	Country	Count
ABW	Online TA	256
AGO	Complementary	0
AGO	Corporate	390
AGO	Corporate	413
AGO	TATO	99
AGO	Direct	16248
AGO	TATO	305
AGO	Groups	65
AGO	Offline TA/TO	388
AGO	Online TA	120
AGO	Online TA	668
AGO	TATO	23606
AIA	Online TA	265
ALB	Direct	144
ALB	Groups	150
ALB	TATO	65
ALB	Offline TA/TO	142
ALB	Online TA	508

Figure 286. Kết quả MDX câu 9

- Công cụ Power BI

Kéo market_segment từ Dim_Market_Segment và distribution_channel từ Dim_Distribution_Channel vào trường Rows → Kéo thả adr (Count) từ Fact vào Values → Kéo thả country từ Dim_Country vào trường Columns.

Figure 287. Kết quả Power BI câu 9

3.2.8.2. Báo cáo và phân tích dữ liệu khách hàng

10. Thống kê có bao nhiêu khách hàng là người lớn, bao nhiêu khách hàng là trẻ em và bao nhiêu khách hàng là em bé trong từng tháng của năm.

Ý nghĩa câu truy vấn: Cho biết số lượng của từng nhóm khách hàng vào các thời điểm trong năm. Từ đó giúp có thể khách sạn đưa ra các kế hoạch kinh doanh phù hợp đối với từng nhóm khách hàng.

- Công cụ SSAS trên các khối Cube

Kéo thả các Attributes: Year, Month Name trong bảng Arrival_Time, độ đo Adults, Children, Babies trong bảng Measures từ Measure Group sang cửa sổ lọc dữ liệu và cửa sổ thực thi. Ta thu được kết quả câu truy vấn.

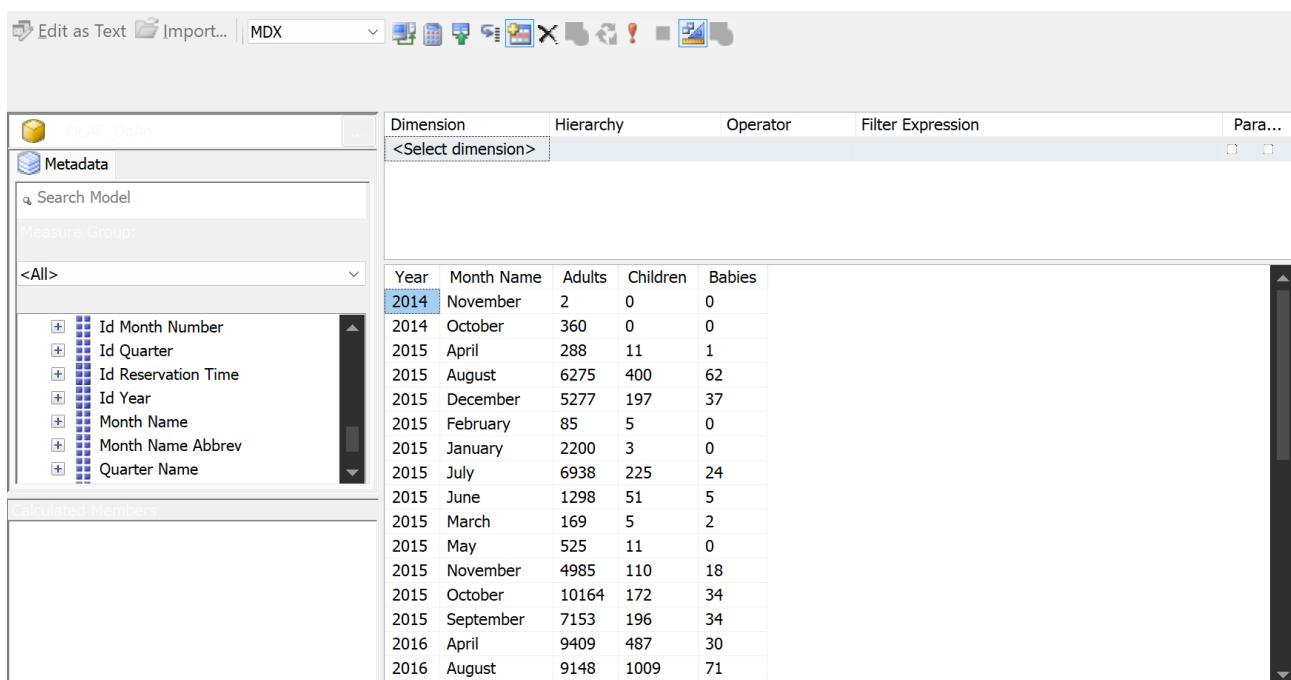


Figure 288. Kết quả SSAS câu 10

- Ngôn ngữ MDX trên các khối Cube

Query:

```

SELECT
NON EMPTY { [Measures].[Adults], [Measures].[Children], [Measures].[Babies] } ON COLUMNS,
NON EMPTY {[Dim Reservation Time].[Year].[Year].ALLMEMBERS
            * [Dim Reservation Time].[Month Name].[Month Name].ALLMEMBERS } ON ROWS
FROM [OLAP_DoAn]
  
```

Kết quả:

```

SELECT
NON EMPTY { [Measures].[Adults], [Measures].[Children], [Measures].[Babies] } ON COLUMNS,
NON EMPTY {[Dim Reservation Time].[Year].ALLMEMBERS
* [Dim Reservation Time].[Month Name].[Month Name].ALLMEMBERS } ON ROWS
FROM [OLAP_DoAn]

```

		Adults	Children	Babies
2014	November	2	0	0
2014	October	360	0	0
2015	April	288	11	1
2015	August	6275	400	62
2015	September	5777	197	37
2015	February	85	5	0
2015	January	2200	3	0
2015	July	6938	225	24
2015	June	1298	51	5
2015	March	169	5	2
2015	May	525	11	0
2015	November	4985	110	18
2015	October	10154	172	34
2015	September	7153	196	34
2016	April	9409	487	30
2016	August	9148	1099	71
2016	December	7744	455	21
2016	February	8096	452	40
2016	January	7689	261	33
2016	July	8852	884	40

Figure 289. Kết quả MDX câu 10

- Giải thích câu truy vấn MDX:

- Hàm Non Empty được sử dụng để loại bỏ các dữ liệu null.
- Toán tử “*” có chức năng như hàm CrossJoin, được dùng để kết các thuộc tính của các bảng chiều.

- **Công cụ Power BI**

Kéo thả year từ bảng Dim_Year và month_name từ bảng Dim_Month vào trường Rows → Kéo thả adults, babies, children từ Fact vào trường Values.

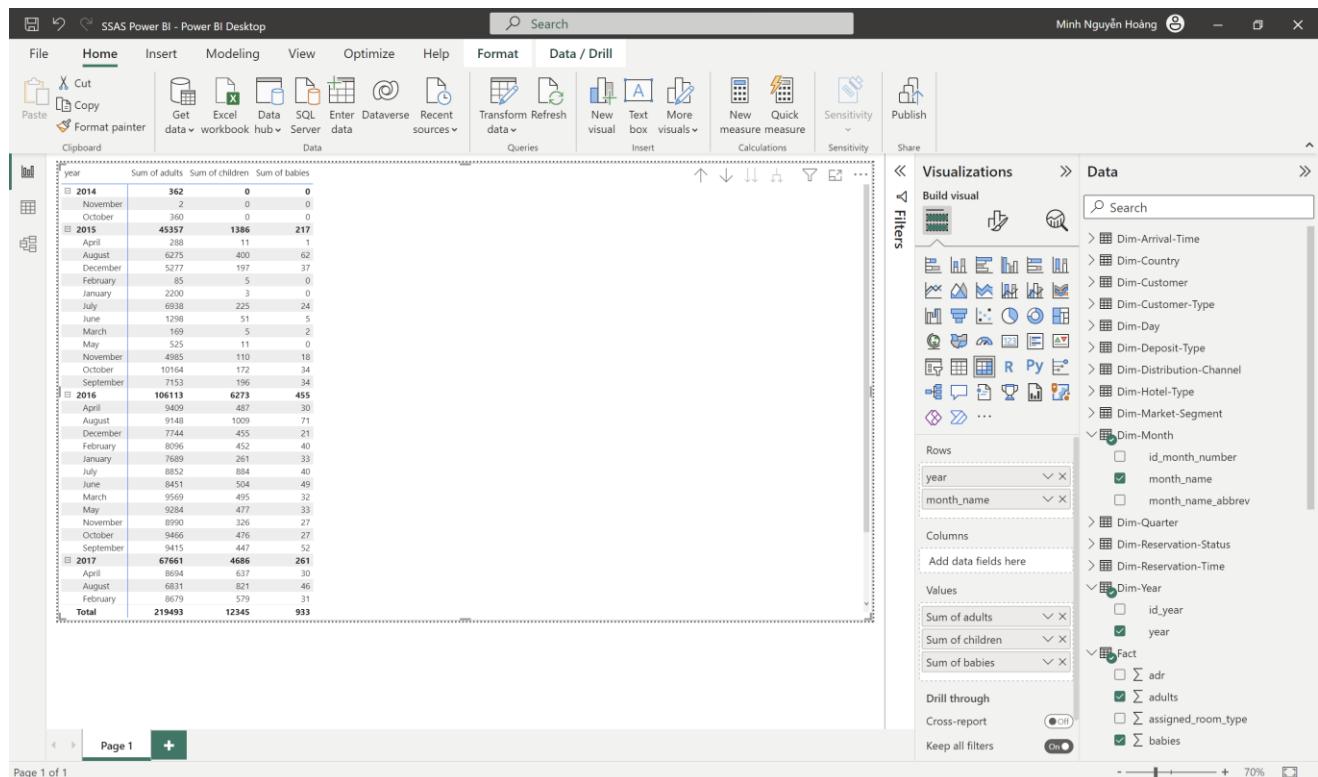


Figure 290. Kết quả Pivot Excel câu 10

- BI Chart

Chọn kiểu Stacked Area Chart. Kéo thả thuộc tính year từ Dim_Year và month_name từ Dim_Month vào X-axis → Kéo thả thuộc tính adults, childrens, babies từ bảng Fact vào Y-axis.

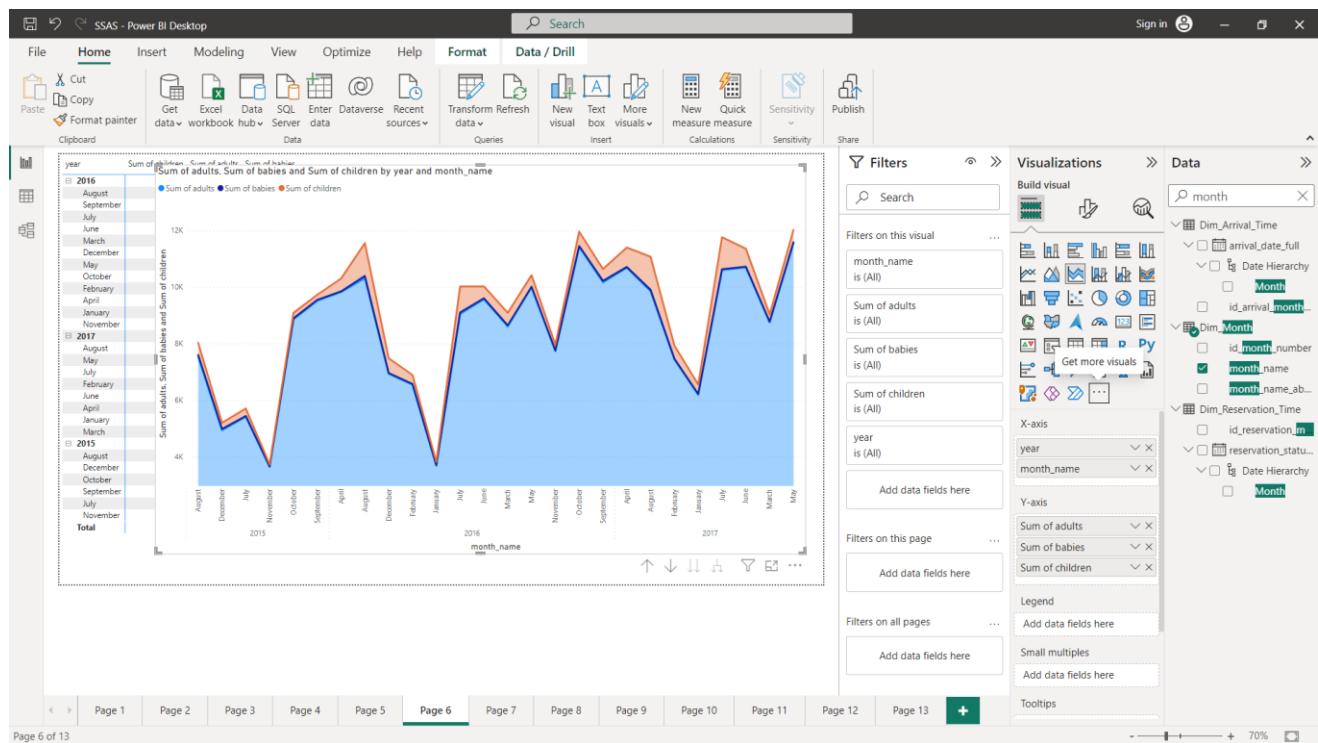


Figure 291. Kết quả trực quan Power BI câu 10

- **Pivot Chart Excel**

Kéo thả thuộc tính year từ Dim_Year và month_name từ Dim_Month vào Axis
 → Kéo thả thuộc tính adults, childrens và babies từ bảng Fact vào Values.

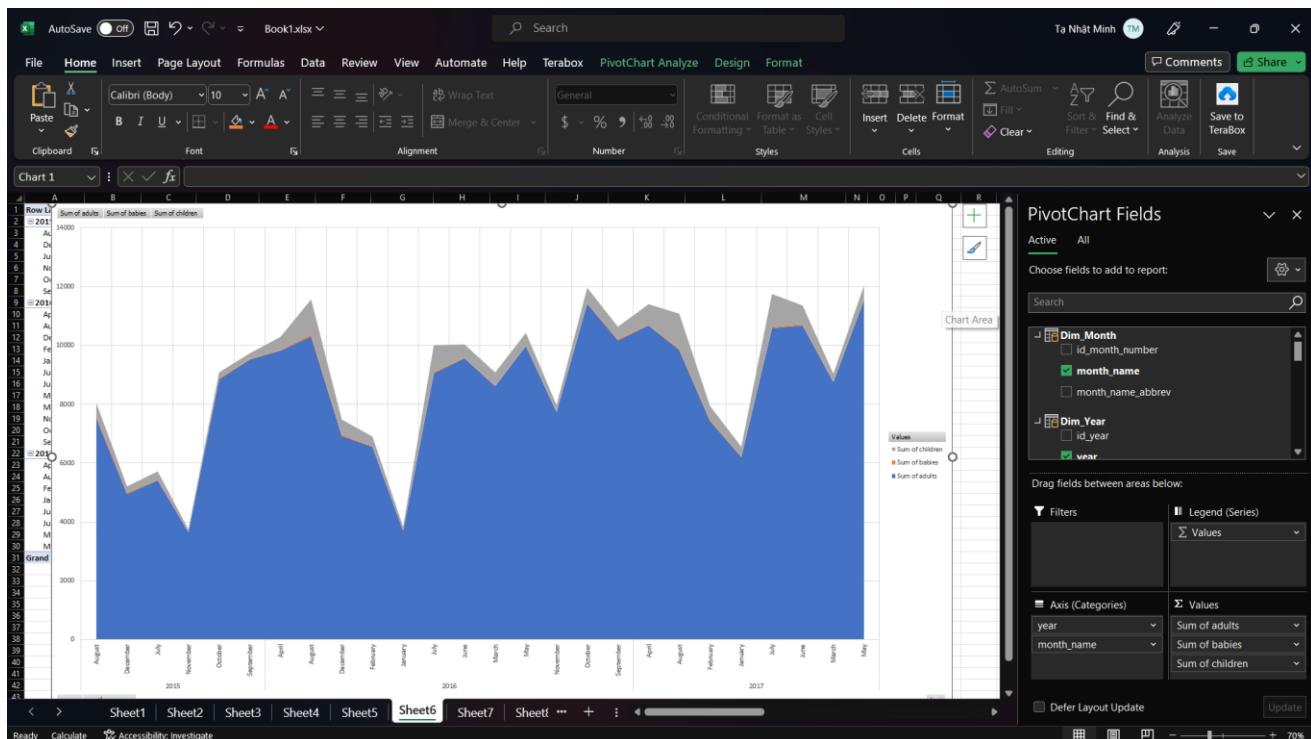


Figure 292. Kết quả Excel Pivot Chart câu 10

11. Thống kê thông tin top 50 khách hàng có chi tiêu cao nhất trong năm trước (2016).

Ý nghĩa câu truy vấn: Cho biết các khách hàng tiềm năng của khách sạn từ đó đưa ra các chương trình khuyến mãi đặc biệt nhằm tri ân khách hàng.

- **Công cụ SSAS trên các khối Cube**

Kéo thả Name, Country, Phone-number, Email, Customer Type từ Dim Customer và Year từ Dim Arrival Time vào vùng truy vấn → Kéo thả Adr từ Measure vào vùng truy vấn → Kéo thả Year từ Dim Arrival Time vào phần Filter, tại Filter Expression chọn 2016 để truy vấn cho năm 2016 → Chọn biểu tượng Design Mode



để chuyển sang script mode. Chỉnh sửa câu truy vấn ban đầu → Chọn click to execute the query.

The screenshot shows the SSAS MDX Editor interface. On the left, there's a navigation pane with 'Metadata' and 'Search Model' sections, and a tree view of dimensions like 'Dim Customer'. A 'Calculated Members' section is also present. At the top, there's a toolbar with various icons. In the center, there's a table with columns: Dimension, Hierarchy, Operator, Filter Expression, and Parameters. The 'Filter Expression' field contains '{ 2016 }'. Below this is a large table grid showing customer data. The columns are: Name, Country, Phone-number, Email, Customer Type, Year, and Adr. The data includes rows for Aaron Acevedo through Aaron Bover.

Dimension	Hierarchy	Operator	Filter Expression	Parameters
Dim Arrival Time	Year	Equal	{ 2016 }	
<Select dimension>				

Name	Country	Phone-number	Email	Customer Type	Year	Adr
Aaron Acevedo	ITA	200-008-5725	Aaron_A@outlook.com	Transient	2016	152
Aaron Acosta	PRT	199-294-1491	Aaron_A@gmail.com	Transient	2016	115
Aaron Adams	CN	543-383-8949	AAdams94@comcast.net	Group	2016	46
Aaron Adams	DEU	574-840-4715	AaronAdams@zoho.com	Transient	2016	78
Aaron Alexander	DEU	802-241-1231	Aaron_A28@zoho.com	Transient	2016	171
Aaron Allen	PRT	793-389-6561	Aaron_A86@outlook.com	Transient	2016	62
Aaron Andrews	CHN	349-236-7542	Andrews.Aaron33@aol.com	Transient	2016	91
Aaron Austin	PRT	348-629-6654	AAustin@aol.com	Transient	2016	0
Aaron Bailey	ESP	488-308-4593	Aaron.Bailey@comcast.net	Transient	2016	160
Aaron Bailey	PRT	602-442-2678	Aaron.Bailey@aol.com	Transient	2016	215
Aaron Barker	PRT	790-077-7919	AaronBarker@aol.com	Transient	2016	75
Aaron Blake	PRT	742-319-2034	Blake_Aaron@zoho.com	Transient-Party	2016	105
Aaron Bowen	GBR	895-181-2623	Aaron_B@zoho.com	Transient	2016	82
Aaron Bowen	PRT	644-504-5841	Aaron_B50@yahoo.com	Transient-Party	2016	112
Aaron Boyd	PRT	748-265-2150	Aaron_Boyd91@mail.com	Transient	2016	62
Aaron Bover	FRA	665-222-4861	Bover_Aaron@mail.com	Transient	2016	87

Figure 293. Kéo thả các trường cần dùng để truy vấn

```

SELECT NON EMPTY { [Measures].[Adr] } ON COLUMNS, NON EMPTY { TopCount
    ((([Dim Customer].[Name].[Name]).ALLMEMBERS * [Dim Customer].[Country].[Country].ALLMEMBERS * [Dim Customer].[Phone-number].[Phone-number].ALLMEMBERS * [Dim Customer].[Email].[Email].ALLMEMBERS * [Dim Customer].[Customer Type].[Customer Type].ALLMEMBERS * [Dim Arrival Time].[Year].[Year].ALLMEMBERS)
    , 50
    ,[Measures].[Adr]
)
} DIMENSION PROPERTIES MEMBER_CAPTION, MEMBER_UNIQUE_NAME ON ROWS FROM ( SELECT ( { [Dim Arrival Time].[Year].&[2016] } ) ON COLUMNS FROM [OLAP_DoAn] ) CELL PROPERTIES VALUE, BACK_COLOR, FORE_COLOR,
FORMATTED_VALUE, FORMAT_STRING, FONT_NAME, FONT_SIZE, FONT_FLAGS

```

Figure 294. Câu truy vấn trong Script Mode sau chỉnh sửa

The screenshot shows the SSAS MDX Editor interface. On the left, there's a navigation pane with 'Metadata' and 'Functions' sections, and a tree view of dimensions like 'Dim Customer'. A 'Calculated Members' section is also present. At the top, there's a toolbar with various icons. In the center, there's a table with columns: Name, Country, Phone-number, Email, Customer Type, Year, and Adr. The data includes rows for Daniel Walter through Daniel Foster.

Name	Country	Phone-number	Email	Customer Type	Year	Adr
Daniel Walter	PRT	865-280-5832	DanielWalter27@comcast.net	Transient	2016	5400
Jessica Morris	PRT	307-008-1571	Jessica.Morris@yahoo.com	Transient-Party	2016	451
Gary Ayers DDS	PRT	452-319-0299	GaryDDS@protonmail.com	Transient	2016	384
Lisa Acevedo	PRT	994-518-7734	Lisa.Acevedo88@mail.com	Transient	2016	382
Duane Lucas	FRA	888-620-7418	Duane_L@xfinity.com	Transient	2016	375
Jacqueline Horne	ESP	199-364-2441	JHorne@att.com	Transient	2016	369
Michael Swanson	PRT	367-031-1515	Swanson.Michael46@att.com	Transient-Party	2016	367
Mark Taylor	FRA	807-454-0583	Mark.Taylor53@verizon.com	Transient-Party	2016	365
Kerry Chen	PRT	186-206-4223	Kerry_C@yahoo.com	Transient	2016	359
Leonard Anthony	ESP	121-809-2001	Anthony.Leonard39@att.com	Transient	2016	359
Michael Kelly	PRT	439-994-7549	Michael_Kelly@comcast.net	Transient	2016	357
Patricia McCormick	ESP	122-099-0077	Mccormick.Patricia@protonm...	Transient	2016	357
Daniel Foster	SVN	573-610-0602	Daniel.F@verizon.com	Transient	2016	353

Figure 295. Kết quả SSAS câu 11

- **Ngôn ngữ MDX trên các khối Cube**

Query:

```

SELECT
    NON EMPTY { [Measures].[Adr] } ON COLUMNS,
    NON EMPTY { TopCount
        ({ {[Dim Customer].[Name].[Name].ALLMEMBERS
            * [Dim Customer].[Country].[Country].ALLMEMBERS
            * [Dim Customer].[Phone-number].[Phone-number].ALLMEMBERS
            * [Dim Customer].[Email].[Email].ALLMEMBERS
            * [Dim Customer].[Customer Type].[Customer Type].ALLMEMBERS
            * [Dim Arrival Time].[Year].[Year].ALLMEMBERS})
        , 50
        , [Measures].[Adr]
    )
} ON ROWS
FROM (SELECT ( { [Dim Arrival Time].[Year].&[2016] } ) ON COLUMNS FROM [OLAP_DoAn])

```

Kết quả:

Name	Country	Address	Type	Year	Adr	
Daniel Walter	PRT	865-280-5832	DanielWalter27@comcast.net	Transient	2016	5400
Jessica Morris	PRT	307-008-1571	Jessica.Morris@yahoo.com	Transient-Party	2016	451
Guillermo DDS	PRT	208-123-4567	Guillermo.DDS@gmail.com	Transient	2016	440
Lisa Acevedo	PRT	994-516-7734	Lisa.Acevedo8@gmail.com	Transient	2016	382
Duane Lucas	FRA	889-420-7418	Duane.L@infinity.com	Transient	2016	375
Jacqueline Horne	ESP	199-364-2441	Jacqueline.Horne@att.com	Transient-Party	2016	369
Michael Swanson	PRT	367-031-1515	Swanson.Michael@att.com	Transient-Party	2016	367
Mark Taylor	FRA	807-454-0583	Mark.Taylor53@verizon.com	Transient-Party	2016	365
Kerry Chen	PRT	186-208-4223	Kerry.C@yahoo.com	Transient	2016	359
Leonard Anthony	ESP	123-456-7890	Anthony.Leonard3@att.com	Transient	2016	359
Michael Kelly	PRT	439-994-7549	Michael.Kelly@comcast.net	Transient	2016	357
Patricia Parikh	ESP	123-099-0977	Patricia.Parikh2000@gmail.com	Transient	2016	357
Daniel Foster	SVN	573-610-0002	Daniel.F@verizon.com	Transient	2016	353
Bruce Abbott	ESP	370-409-0577	Bruce.Abbott@att.com	Transient	2016	352
Jon Smith	BEL	535-238-4748	Smith_Jon@outlook.com	Transient	2016	351
Stephen Villareal	NLD	492-213-9871	Stephen.Villareal@verizon.com	Transient-Party	2016	352
Joshua Shaw	BEL	215-988-0348	Joshua.S@comcast.net	Transient	2016	349
Stacy Oberlin	PRT	435-589-5848	Stacy.Oberlin@comcast.net	Transient	2016	347
Brian Pritchett	ESP	435-589-5849	Brian.Pritchett@q1000.com	Transient	2016	346
Christopher Bright	PRT	720-022-5778	Christopher.Bright@att.com	Transient-Party	2016	340
Debra Monroe	PRT	407-492-7568	Debra.Monroe@yahoo.com	Transient-Party	2016	340

Figure 296. Kết quả MDX câu 11

- **Giải thích câu truy vấn MDX:**

- Hàm TopCount dùng để sắp xếp các dữ liệu của kết quả sau khi tích chéo các thuộc tính bằng hàm CrossJoin theo thứ tự sắp xếp độ đo Adr (doanh thu) giảm dần và trả về dòng đầu tiên có giá trị cao nhất.
- Where dung để lọc ra toàn bộ dữ liệu chính trong năm 2016.

- **Công cụ Power BI**

Kéo thả Year từ bảng Dim_Year vào trường Columns → Name, Country, Phone-number, Email, Customer Type từ Dim Customer và Year từ Dim Arrival Time vào

trường Columns → Kéo thả adr từ bảng Fact vào trường Values → Tại Filters Year, chọn filter chọn giá trị 2016 để truy vấn top 50 năm 2016 → Tại Filters country, chọn filter type Top N, set up số giá trị cần lấy top là 50, kéo thả adr vào ô By value → Sort giá trị tại cột Sum of adr theo thứ tự giảm dần.

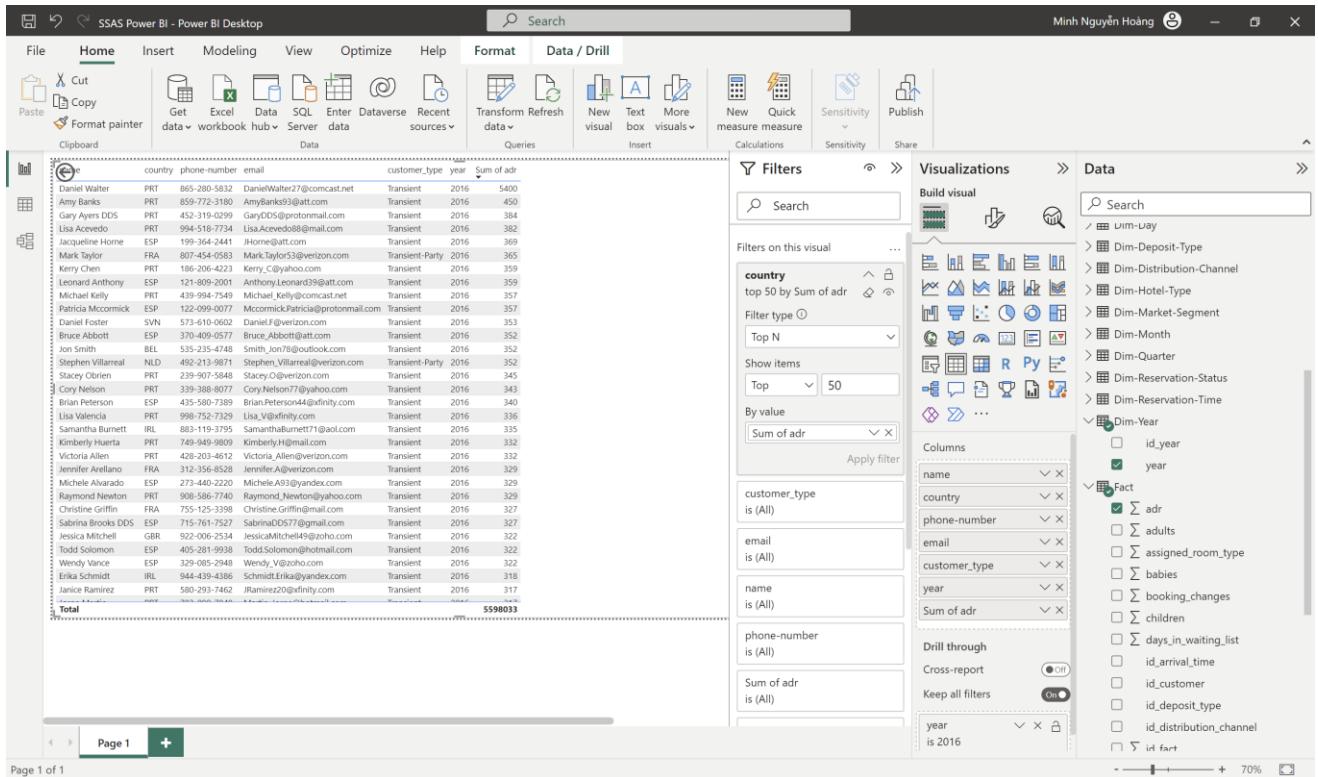


Figure 297. Kết quả Pivot Excel câu 11

- BI Chart

Chọn kiểu Clustered Column Chart. Kéo thả thuộc tính name, email, phone-number từ Dim_Customer, country từ Dim_Country và customer_type từ Dim_Customer_Type vào X-axis → Kéo thả year từ Dim_Year vào Legend → Kéo thả thuộc tính adr từ bảng Fact vào Y-axis. Tại filter year chọn Basic filtering và chọn 2016, tại filter name chọn Top 50 lọc theo Sum of adr.

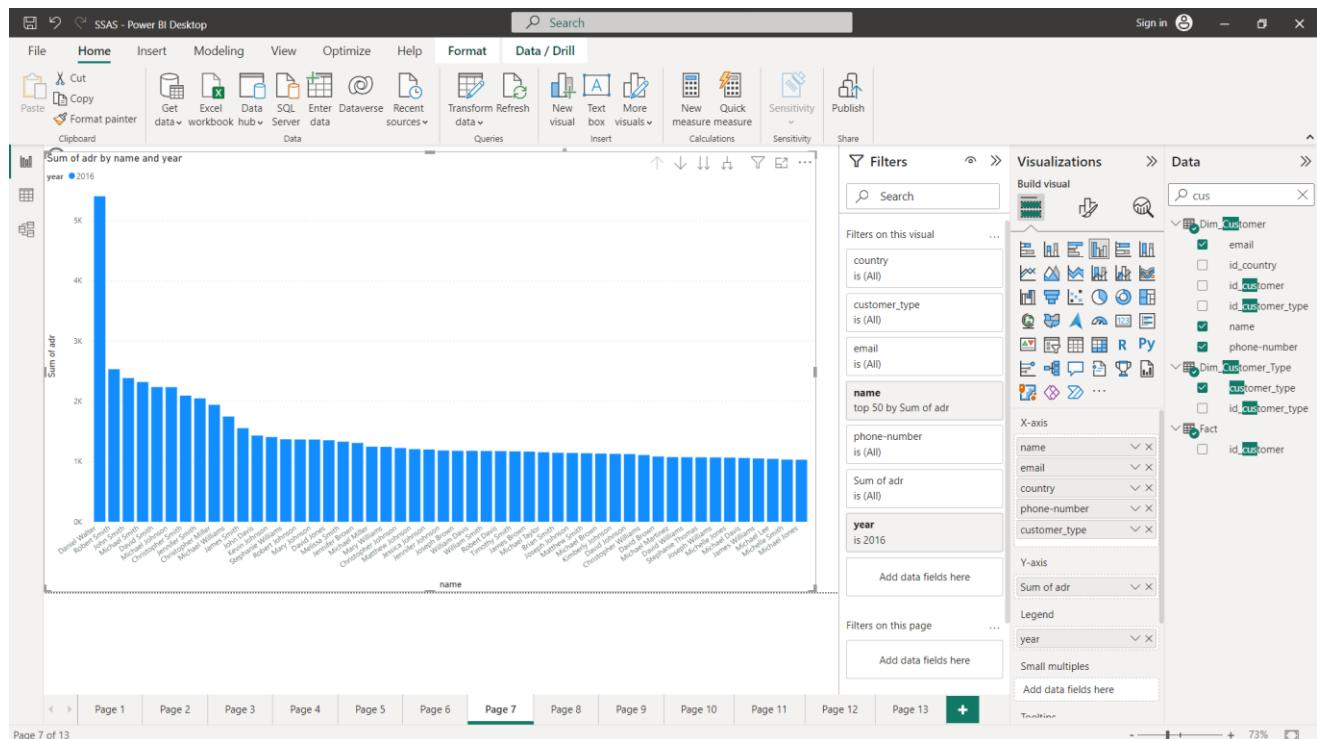


Figure 298. Kết quả trực quan Power BI câu 11

- Pivot Chart Excel

Kéo thả thuộc tính *year* từ *Dim_Year* vào *Filters* → Kéo thả *name*, *email*, *phone-number* từ *Dim_Customer*, *country* từ *Dim_Country* và *customer_type* từ *Dim_Customer_Type* vào *Axis* → Kéo thả thuộc tính *adr* từ bảng Fact vào *Values*. Tại *name* chọn *Value Filters* và chọn *top 50* theo *Sum of adr*.

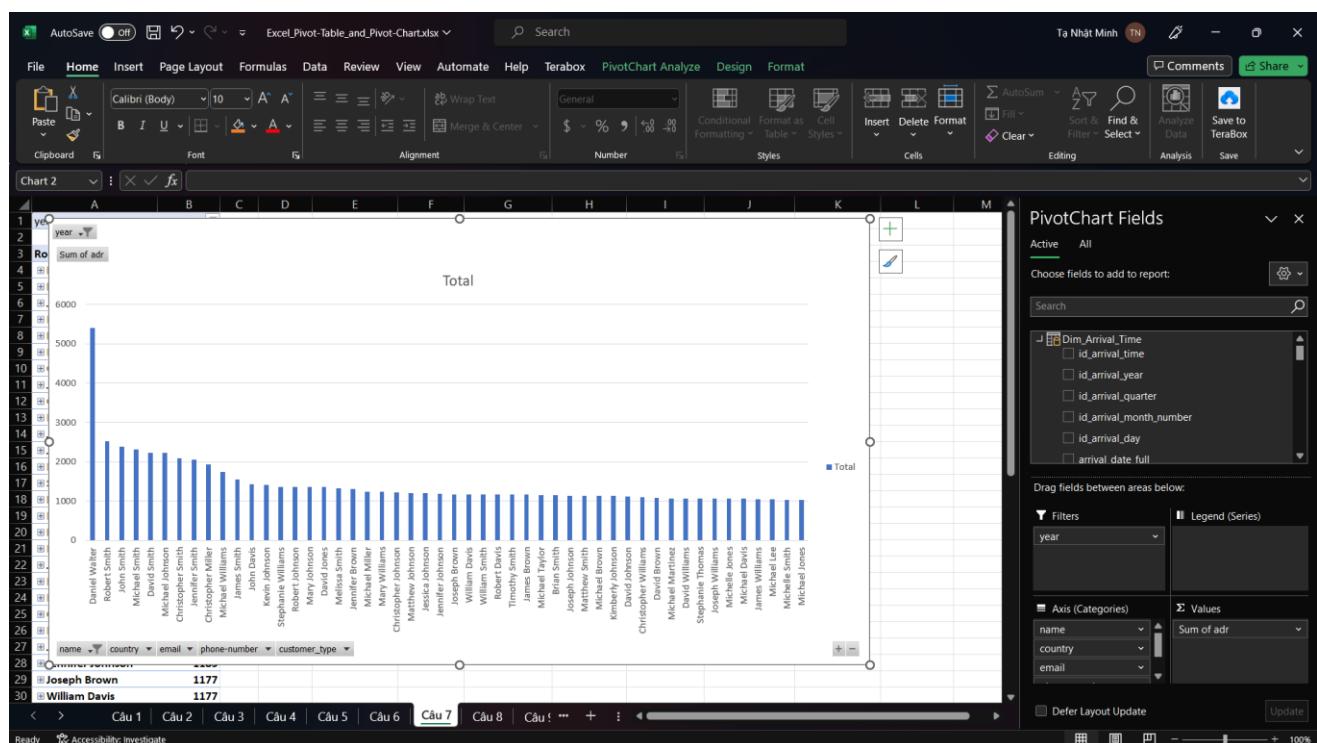


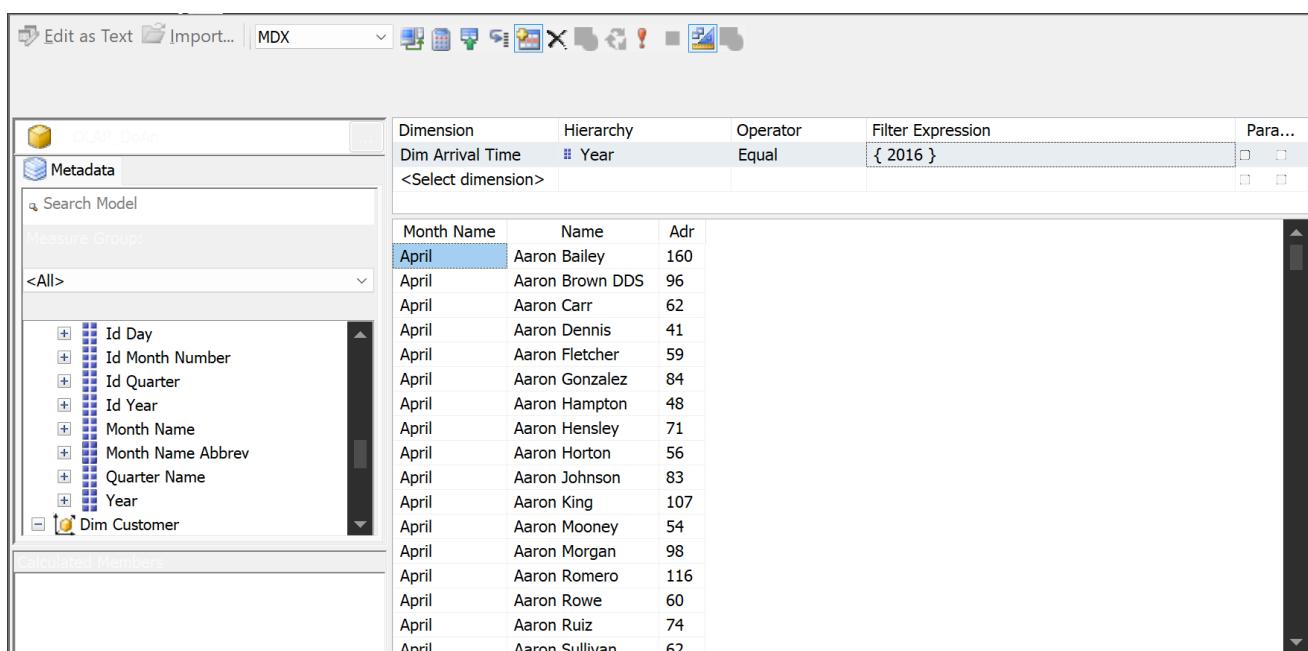
Figure 299. Kết quả Excel Pivot Chart câu 11

12. Thống kê tên top 5 khách hàng có chi tiêu cao nhất theo từng tháng của năm 2016.

Ý nghĩa câu truy vấn: Cho biết các khách hàng tiềm năng của khách sạn từ đó đưa ra các chương trình khuyến mãi đặc biệt nhằm tri ân khách hàng.

- Công cụ SSAS trên các khối Cube

Kéo thả Name từ Dim Customer và Month từ Dim Arrival Time vào vùng truy vấn → Kéo thả Adr từ Measure vào vùng truy vấn → Kéo thả Year từ Dim Arrival Time vào phần Filter, tại Filter Expression chọn 2016 để truy vấn cho năm 2016 → Chọn biểu tượng Design Mode  để chuyển sang script mode. Chỉnh sửa câu truy vấn ban đầu → Chọn click to execute the query.



Month Name	Name	Adr
April	Aaron Bailey	160
April	Aaron Brown DDS	96
April	Aaron Carr	62
April	Aaron Dennis	41
April	Aaron Fletcher	59
April	Aaron Gonzalez	84
April	Aaron Hampton	48
April	Aaron Hensley	71
April	Aaron Horton	56
April	Aaron Johnson	83
April	Aaron King	107
April	Aaron Mooney	54
April	Aaron Morgan	98
April	Aaron Romero	116
April	Aaron Rowe	60
April	Aaron Ruiz	74
April	Aaron Sullivan	62

Figure 300. Kéo thả các trường cần dùng để truy vấn

```

SELECT NON EMPTY { [Measures].[Adr] } ON COLUMNS, NON EMPTY {
Generate([Dim Arrival Time].[Month Name].children,
Topcount({[Dim Arrival Time].[Month Name].currentmember* [Dim Customer].[Name].children}, 5, [Measures].[Adr])
) } DIMENSION PROPERTIES MEMBER_CAPTION, MEMBER_UNIQUE_NAME ON ROWS FROM ( SELECT ( { [Dim Arrival Time].[Year]&[2016] } ) ) ON COLUMNS FROM [OLAP_DoAn] WHERE ( [Dim Arrival Time].[Year]&[2016] ) CELL
PROPERTIES VALUE, BACK_COLOR, FORE_COLOR, FORMATTED_VALUE, FORMAT_STRING, FONT_NAME, FONT_SIZE,
FONT_FLAGS
    
```

Figure 301. Câu truy vấn trong Script Mode sau chỉnh sửa

The screenshot shows the SSAS Cube Structure interface. On the left, there's a navigation pane with 'Metadata' and 'Functions' tabs, and a search bar. Below it is a tree view of dimensions: 'Dim Day', 'Dim Month Number', 'Dim Quarter', 'Dim Year', 'Month Name', 'Month Name Abbrev', 'Quarter Name', 'Year', 'Dim Customer', 'Country', 'Customer Type', 'Email', 'Id Country', and 'Id Customer'. The main area displays an MDX query and its results. The query is:

```

SELECT NON EMPTY { [Measures].[Adr] } ON COLUMNS, NON EMPTY {
Generate([Dim Arrival Time].[Month Name].children,
Topcount({ [Dim Arrival Time].[Month Name].currentmember*[Dim Customer].[Name].children}, 5, [Measures].[Adr])
) } DIMENSION PROPERTIES MEMBER_CAPTION, MEMBER_UNIQUE_NAME ON ROWS FROM ( SELECT ( { [Dim Arrival Time].[Year].&[2016] } ) ) ON COLUMNS FROM [OLAP_DoAn] WHERE ( [Dim Arrival Time].[Year].&[2016] ) CELL
PROPERTIES VALUE, BACK_COLOR, FORE_COLOR, FORMATTED_VALUE, FORMAT_STRING, FONT_NAME, FONT_SIZE,
FONT_FLAGS

```

The results table has columns 'Month Name', 'Name', and 'Adr'. The data is:

Month Name	Name	Adr
April	Jennifer Smith	635
April	Daniel Brown	456
April	Daniel Moore	416
April	James Smith	397
April	Jeffrey Jones	395
August	Michael Smith	731
August	Kevin Johnson	588
August	Brandon Ma...	569
August	Daniel Foster	550
August	Mary Williams	518
December	Jessica Morris	649
December	Michael Jac...	422
December	Angela Smith	415
December	John Smith	386
December	Sam... August	384

Figure 302. Kết quả SSAS câu 12

- **Ngôn ngữ MDX trên các khối Cube**

Query:

```

SELECT
    NON EMPTY { [Measures].[Adr] } ON COLUMNS,
    NON EMPTY {
        Generate([Dim Arrival Time].[Month Name].children,
        Topcount({ [Dim Arrival Time].[Month Name].currentmember
            * [Dim Customer].[Name].children}
        , 5
        , [Measures].[Adr])
    ) } ON ROWS
FROM ( SELECT ( { [Dim Arrival Time].[Year].&[2016] } ) ) ON COLUMNS FROM [OLAP_DoAn]
WHERE ( [Dim Arrival Time].[Year].&[2016] )

```

Kết quả:

Month Name	Adr
April	Jennifer Smith 635
April	Daniel Brown 456
April	Daniel Moore 416
April	James Smith 397
April	Jeffrey Jones 395
August	Michael Smith 731
August	Kevin Johnson 588
August	Brandon Martinez 569
August	Robert Williams 550
August	Mary Williams 518
December	Jessica Morris 649
December	Michael Jackson 422
December	Angela Smith 415
December	John Smith 386
December	Gary Ayers DDS 384
February	Christopher Sanchez 313
February	Danielle Smith 291
February	Robert Williams 265
February	Christopher Miller 274
February	Jonathan Williams 265
January	Michael Johnson 510

Figure 303. Kết quả MDX câu 12

○ Giải thích câu truy vấn MDX:

- Hàm TopCount dùng để sắp xếp các dữ liệu của kết quả sau khi tích chéo các thuộc tính bằng hàm CrossJoin theo thứ tự sắp xếp độ đo Adr (doanh thu) giảm dần và trả về dòng đầu tiên có giá trị cao nhất.
- Hàm Generate dùng để duyệt từng giá trị trong tháng của Arrival time ở tham số thứ nhất. Tại tham số thứ hai, hàm Generate lấy ra tập dữ liệu đã tạo trong TopCount đã trình bày trước đó.
- Where dung để lọc ra toàn bộ dữ liệu chính trong năm 2016.

- Công cụ Power BI

Kéo thả name từ Dim_Customer vào trường Rows → Kéo thả month_name từ Dim_Month cà year từ Dim_Year vào trường Columns → Kéo thả adr từ Fact vào Values → Tại Filters name, chọn Top N với tham số là 10 By Values Sum of Adr. Tại filters year, chọn Basic Filtering và chọn 2016. Tại filters month_name, chọn Basic Filtering và chọn tháng cần truy vấn → Làm tương tự cho các tháng còn lại.

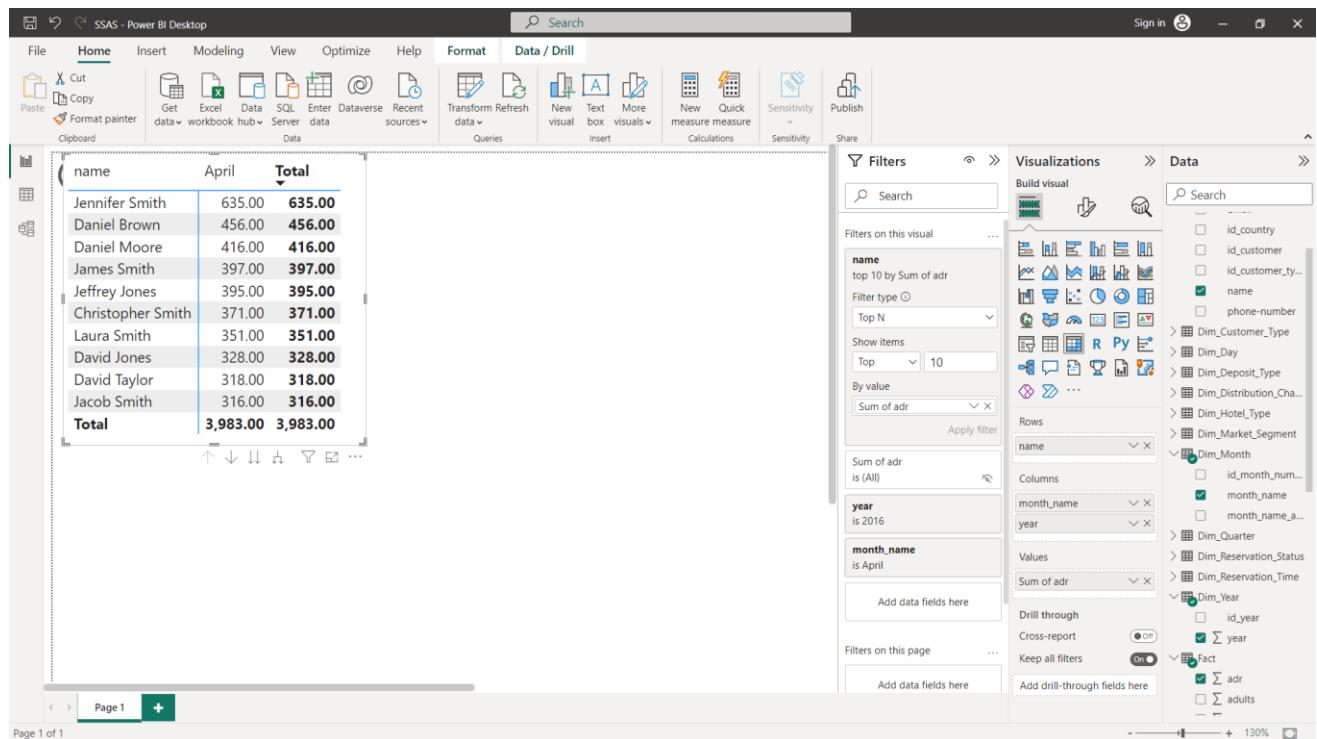


Figure 304. Kết quả Pivot Excel câu 12 (1 tháng)

- BI Chart

Chọn kiểu Clustered Column Chart. Kéo thả thuộc tính year từ Dim_Year và name từ Dim_Customer vào X-axis → Kéo thả month_name từ Dim_Month vào Legend → Kéo thả thuộc tính adr từ bảng Fact vào Y-axis. Tại filter year lọc với giá trị là 2016, tại filter month_name chọn tháng cần lấy top 5. Tại filter name chọn Top 5 lọc theo Sum of adr.

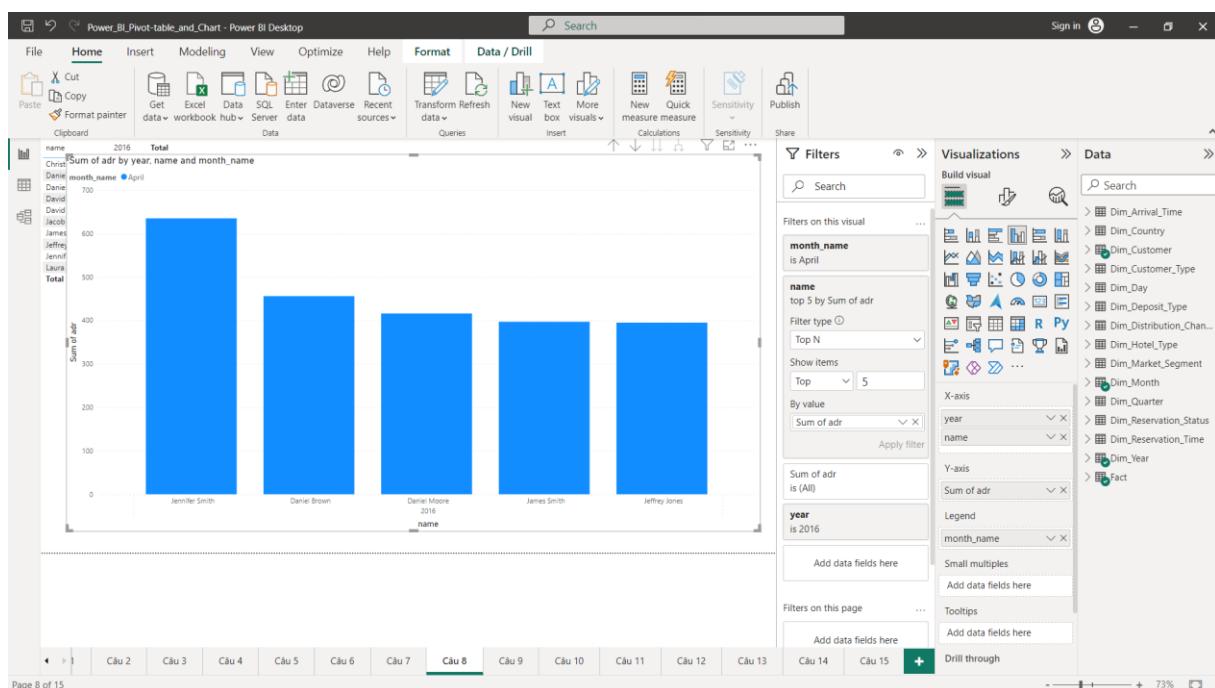


Figure 305. Kết quả trực quan Power BI câu 12

- **Pivot Chart Excel**

Kéo thả thuộc tính year từ Dim_Year và month_name từ Dim_Month vào Filters
 → Kéo thả name từ Dim_Customer vào Legend → Kéo thả thuộc tính adr từ bảng Fact vào Values. Tại name chọn Value Filters chọn top 5 theo Sum of adr.

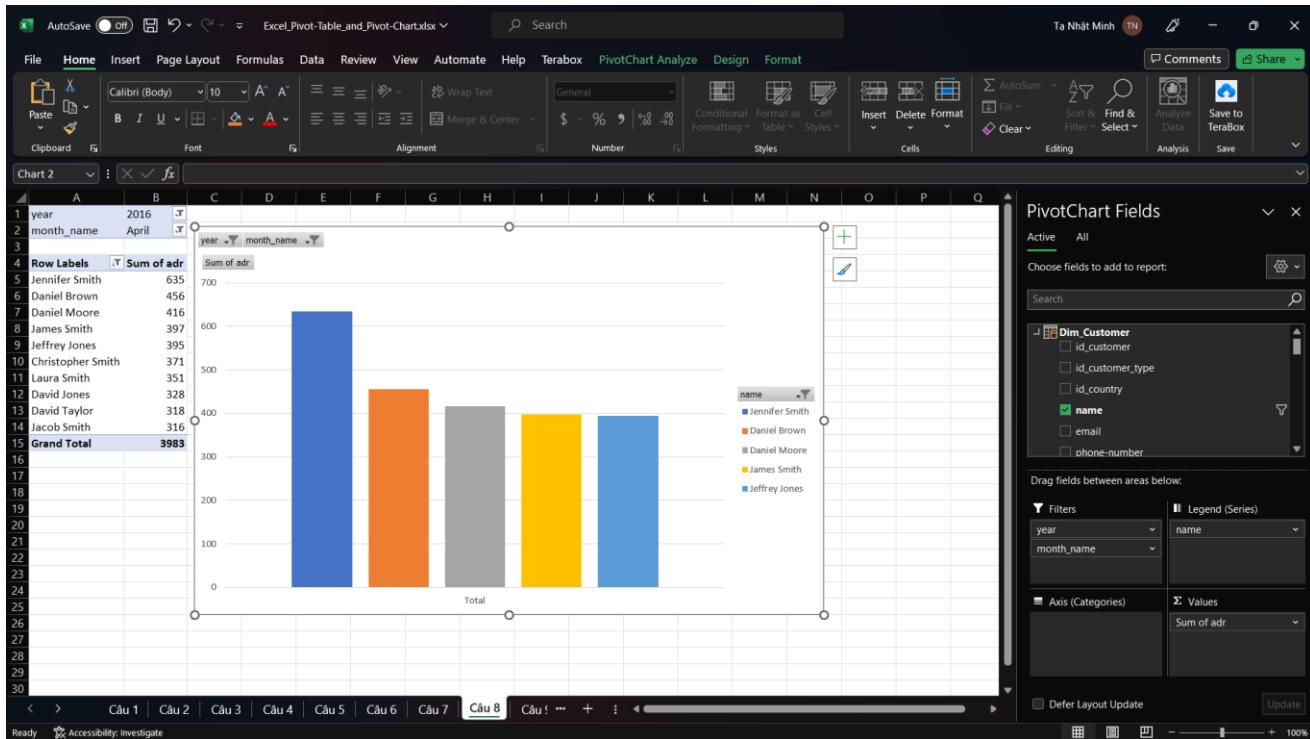


Figure 306. Kết quả Excel Pivot Chart câu 12

13. Thống kê email top 3 khách hàng có doanh số cao nhất theo quốc gia năm 2016.

Ý nghĩa câu truy vấn: Cho biết các khách hàng tiềm năng của khách sạn từ đó đưa ra các chương trình khuyến mãi đặc biệt nhằm tri ân khách hàng.

- **Công cụ SSAS trên các khối Cube**

Tạo Named Set có tên là [Câu 9] dùng để lấy thông tin email 3 khách hàng của mỗi quốc gia có doanh thu cao nhất năm 2016 cho khách sạn



Figure 307. Tạo Named Set cho câu 13

Kéo thả Email và Country từ Dim Customer vào vùng truy vấn → Kéo thả Adr từ Measure vào vùng truy vấn → Kéo thả Year từ Dim Arrival Time vào phần Filter,

tại Filter Expression chọn 2016 để truy vấn cho năm 2016 → Chọn biểu tượng Design Mode để chuyển sang script mode. Chính sửa câu truy vấn ban đầu → Chọn click to execute the query.

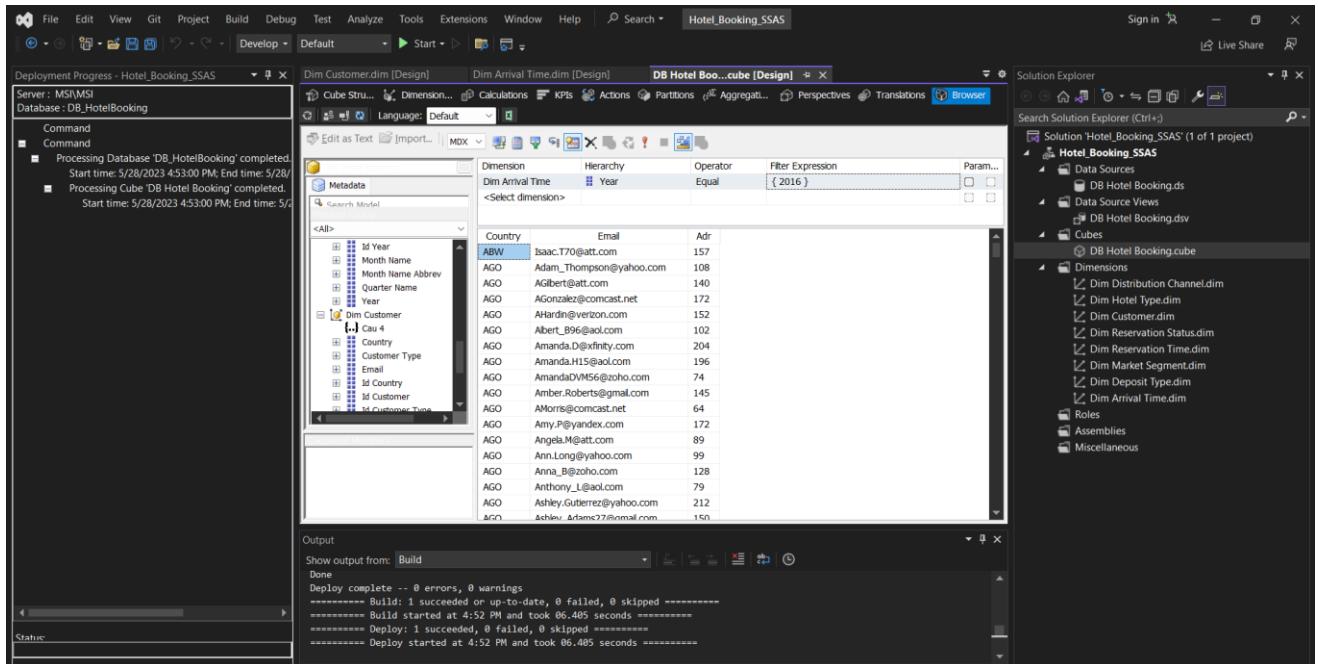


Figure 308. Kéo thả các trường cần dùng để truy vấn

```
SELECT NON EMPTY { [Measures].[Adr] } ON COLUMNS,
NON EMPTY { [Cau 9] } ON ROWS FROM [DB Hotel Booking] WHERE ([Dim Arrival Time].[Year].&[2016]) |
```

Figure 309. Câu truy vấn trong Script Mode sau chỉnh sửa

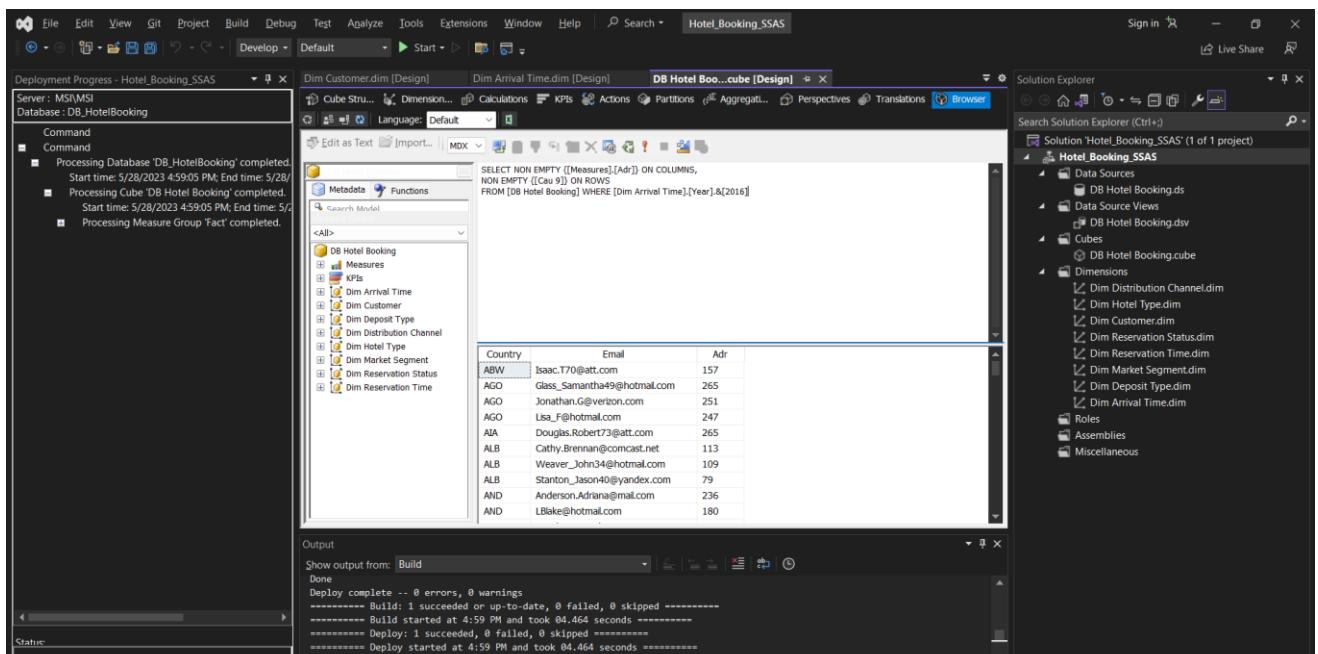


Figure 310. Kết quả SSAS câu 13

- Ngôn ngữ MDX trên các khối Cube

Query:

```

SELECT NON EMPTY {[Measures].[Adr]} ON COLUMNS,
NON EMPTY {GENERATE([Dim Customer].[Country].children,
TOPCOUNT([Dim Customer].[Country].currentmember*[Dim Customer].[Email].children,
3, [Measures].[Adr]))} ON ROWS
FROM [DB Hotel Booking] WHERE [Dim Arrival Time].[Year].&[2016]

```

Kết quả:

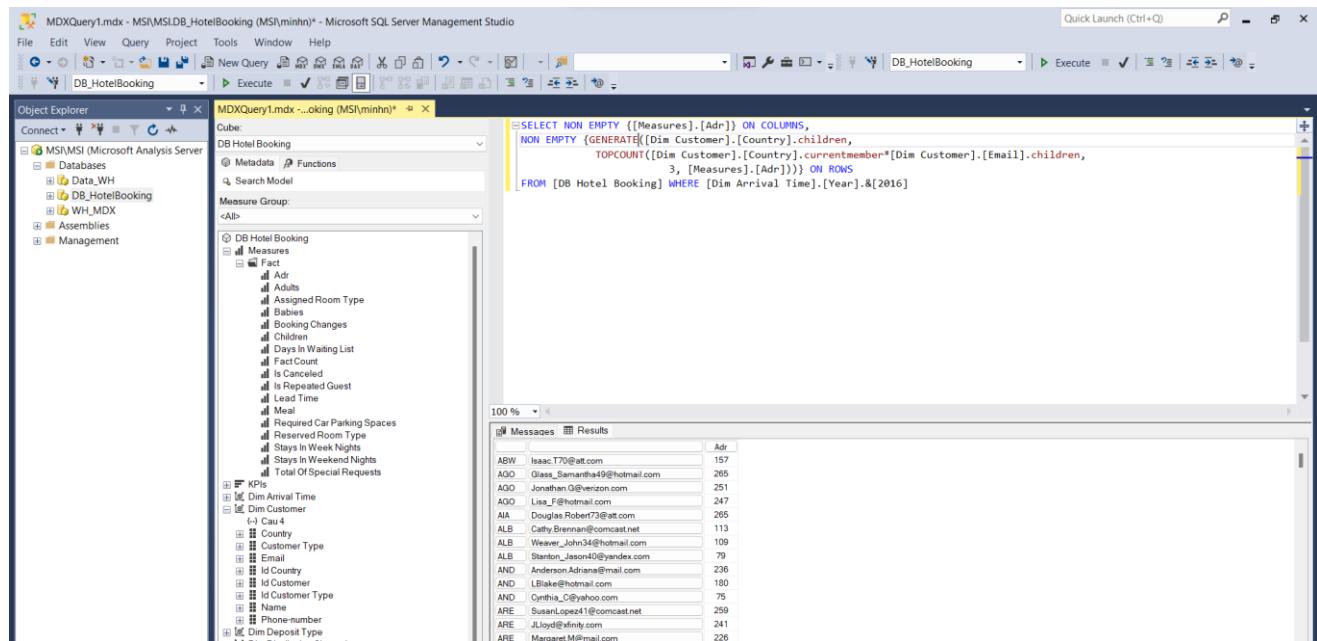


Figure 311. Kết quả MDX câu 13

○ Giải thích câu truy vấn MDX:

- Hàm TopCount dùng để sắp xếp các dữ liệu của kết quả sau khi tích chéo các thuộc tính bằng hàm CrossJoin theo thứ tự sắp xếp độ đo Adr (doanh thu) giảm dần và trả về dòng đầu tiên có giá trị cao nhất.
- Hàm Generate dùng để duyệt từng giá trị trong tháng của Arrival time ở tham số thứ nhất. Tại tham số thứ hai, hàm Generate lấy ra tập dữ liệu đã tạo trong TopCount đã trình bày trước đó.

- Công cụ Power BI

Kéo thả email từ Dim_Customer vào trường Rows → Kéo thả country từ Dim_Country và year từ Dim_Year vào trường Columns → Kéo thả adr từ Fact vào Values → Tại Filters name, chọn Top N với tham số là 3 By Values Sum of Adr. Tại filters year, chọn Basic Filtering và chọn 2016. Tại Filter country, chọn country cần truy vấn → Làm tương tự cho các quốc gia khác.

The screenshot shows the Power BI Desktop interface. On the left, a Pivot Table displays data for the year 2016, categorized by email and total adr. The filters pane on the right shows several filters applied: country (is PRT), email (top 3 by sum of adr), and year (is 2016). The visualizations pane lists various chart types, and the data pane shows a list of dimensions and measures.

email	2016	Total
DanielWalter27@comcast.net	5,400.00	5,400.00
GaryDDS@protonmail.com	384.00	384.00
Jessica.Morris@yahoo.com	451.00	451.00
Total	6,235.00	6,235.00

Figure 312. Kết quả Power BI câu 13 (1 quốc gia)

- BI Chart

Chọn kiểu Clustered Column Chart. Kéo thả thuộc tính year từ Dim_Year vào Legend → Kéo thả country từ Dim_Country và email từ Dim_Customer vào X-axis → Kéo thả thuộc tính adr từ bảng Fact vào Y-axis. Tại filter country chọn Basic filtering với giá trị lọc là quốc gia muốn lấy top 3 email, tại filter email chọn Basic filtering với giá trị 2016. Tại filter email chọn Top 3 và lọc theo Sum of adr.

The screenshot shows the Power BI Desktop interface with a clustered column chart. The chart displays the sum of adr by country (email) and year. The filters pane on the right shows filters for country (is PRT) and email (top 3 by sum of adr). The visualizations pane lists various chart types, and the data pane shows a list of dimensions and measures.

email	2016	Total
DanielWalter27@comcast.net	5,400.00	5,400.00
GaryDDS@protonmail.com	384.00	384.00
Jessica.Morris@yahoo.com	451.00	451.00
Total	6,235.00	6,235.00

Figure 313. Kết quả trực quan Power BI câu 13

- Pivot Chart Excel

Kéo thả thuộc tính year từ Dim_Year vào Filters và email từ Dim_Customer vào Axis → Kéo thả country từ Dim-Country vào Legend → Kéo thả thuộc tính adr từ bảng Fact vào Values. Tại email chọn Value Filters chọn Top 3 theo Sum of Adr.

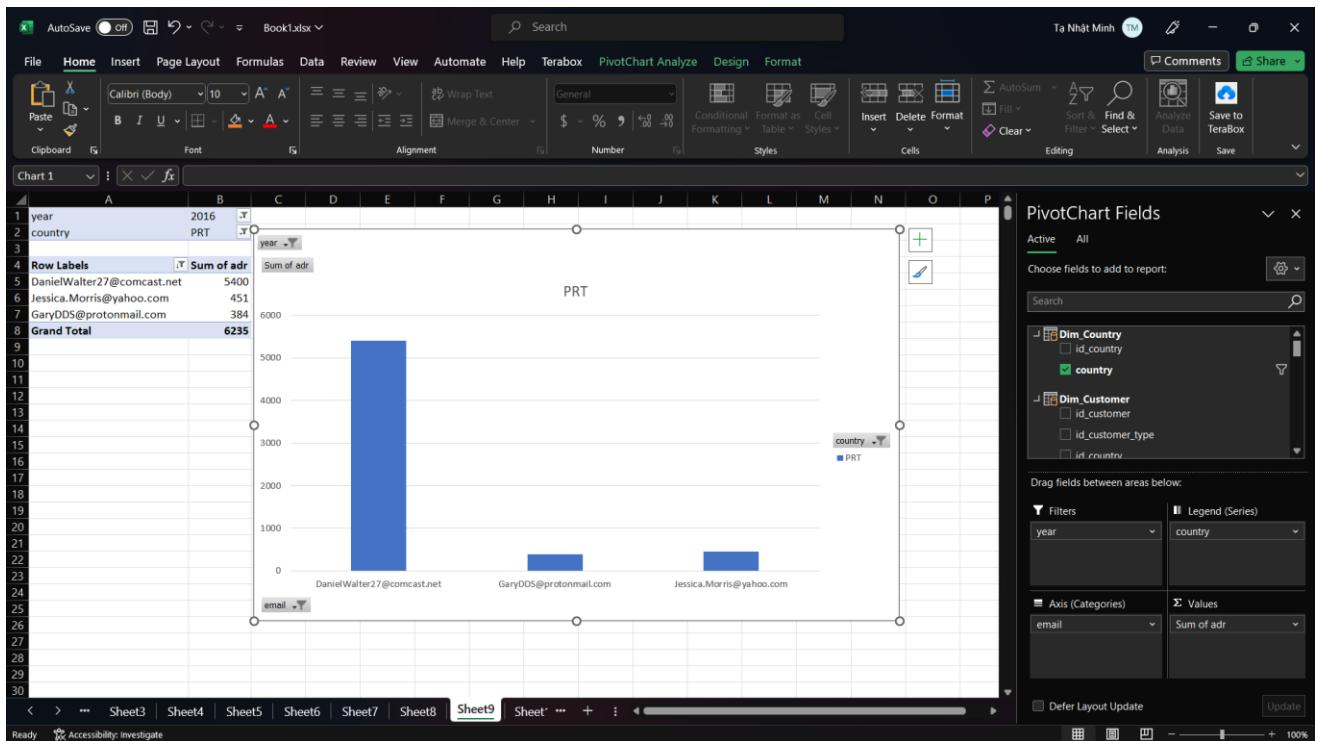


Figure 314. Kết quả Excel Pivot Chart câu 13

14. Thống kê top 10 thông tin khách hàng kèm chi tiêu của họ có thời gian chờ lâu nhất theo từng quốc gia năm 2016.

Ý nghĩa câu truy vấn:

Cho biết các khách hàng có thời gian chờ lâu kèm chi tiêu của họ có thời gian chờ lâu nhất theo từng quốc gia năm 2016.

- Công cụ SSAS trên các khối Cube

Tạo Named Set có tên là [Cau 10] dùng để lấy thông tin khách hàng theo từng quốc gia có thời gian chờ lâu nhất năm 2016 khách sạn.



Figure 315. Tạo Named Set [Cau 10] cho câu 14

Kéo thả Country từ Dim Customer, Name từ Dim Customer và Month từ Dim Arrival Time vào vùng truy vấn → Kéo thả Adr, lead time từ Measure vào vùng truy vấn → Kéo thả Year từ Dim Arrival Time vào phần Filter, tại Filter Expression chọn 2016 để truy vấn cho năm 2016 → Chọn biểu tượng Design Mode để chuyển sang script mode. Chỉnh sửa câu truy vấn ban đầu → Chọn click to execute the query.

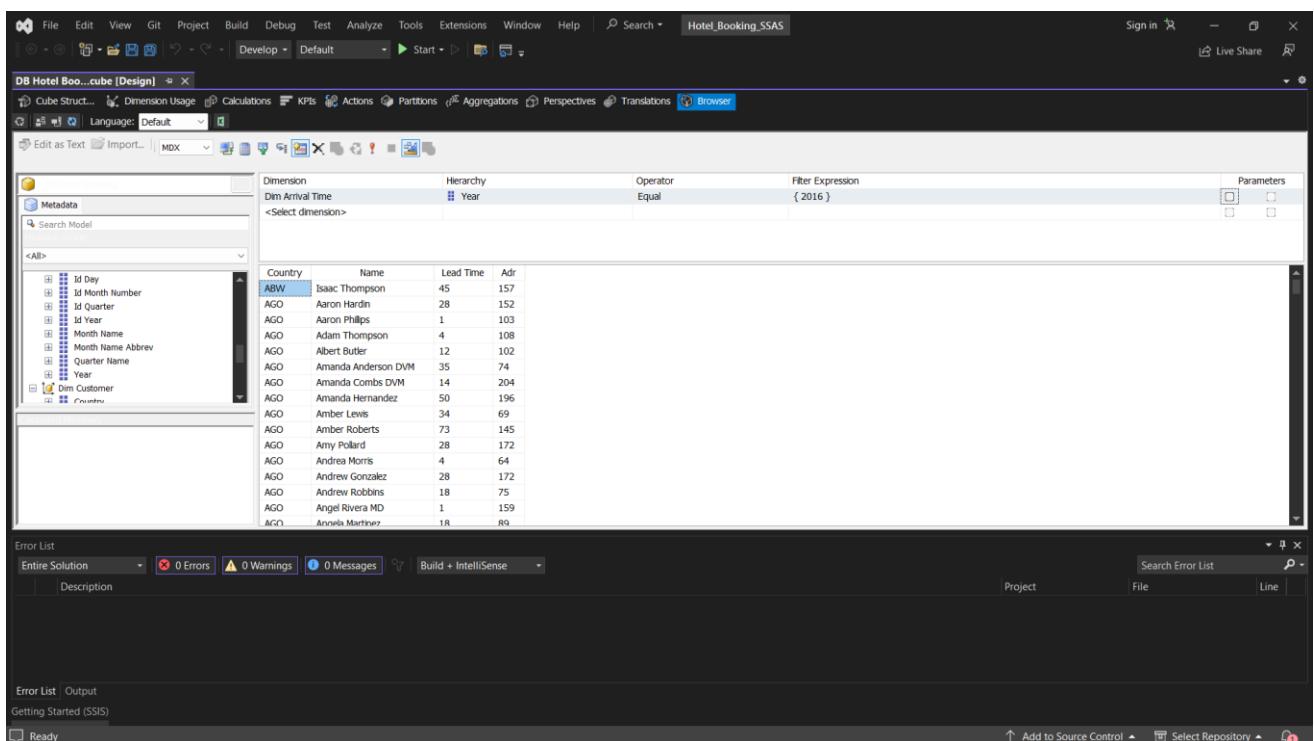


Figure 316. Kéo thả các trường cần dùng để truy vấn

```
SELECT NON EMPTY { [Measures].[Lead Time], [Measures].[Adr] } ON COLUMNS,
NON EMPTY { [Cau 10] } ON ROWS
FROM [DB Hotel Booking] WHERE [Dim Arrival Time].[Year].&[2016]
```

Figure 317. Câu truy vấn trong Script Mode sau chỉnh sửa

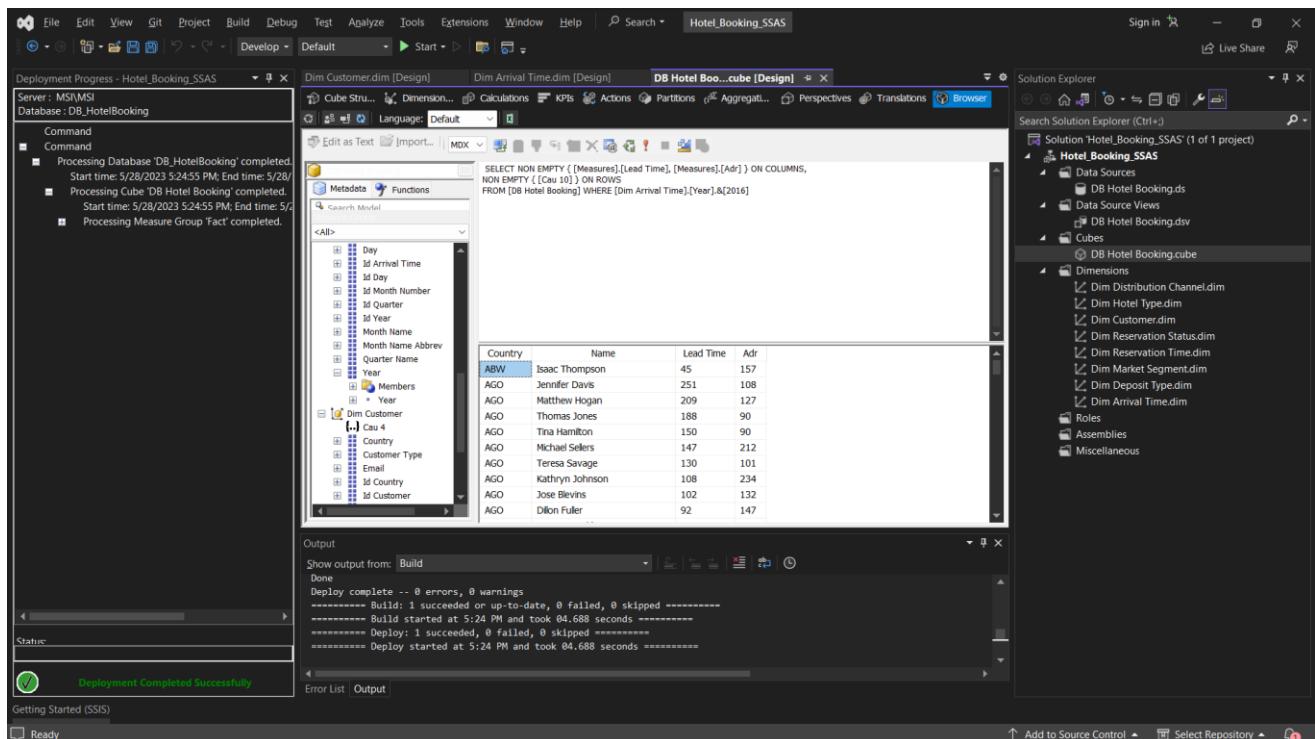


Figure 318. Kết quả SSAS câu 14

- Ngôn ngữ MDX trên các khối Cube

Query:

```
SELECT NON EMPTY {[Measures].[Lead Time], [Measures].[Adr]} ON COLUMNS,
NON EMPTY {GENERATE([Dim Customer].[Country].children,
TOPCOUNT([Dim Customer].[Country].currentmember*[Dim Customer].[Name].children,
10, [Measures].[Lead Time]))} ON ROWS
FROM [DB Hotel Booking] WHERE [Dim Arrival Time].[Year].&[2016]
```

Kết quả:

Country	Name	Lead Time	Adr
ABW	Isaac Thompson	45	157
AGO	Jennifer Davis	251	108
AGO	Matthew Hogan	209	127
AGO	Thomas Jones	188	90
AGO	Tina Hamilton	150	90
AGO	Michael Sellers	147	212
AGO	Teresa Savage	130	101
AGO	Kathryn Johnson	108	234
AGO	Jose Blevins	102	132
AGO	Dillon Fuller	92	147
AGO	Lia Fitzgerald	91	247
AIA	Robert Douglas	0	265
ALB	Michelle Cannon	215	75
ALB	Susan Taylor	215	75
ALB	Jason Stanton	128	79
ALB	Tom Patel	40	35
ALB	Cathy Brennan	32	113
ALB	John Weaver	13	109

Figure 319. Kết quả MDX câu 14

- Công cụ Power BI

Kéo thả email từ Dim_Customer vào trường Rows → Kéo thả month_name từ Dim_Month, country từ Dim_Country và year từ Dim_Year vào trường Columns → Kéo thả adr từ Fact vào Values → Tại Filters name, chọn Top N với tham số là 10 By Values Sum of Adr. Tại filters year, chọn Basic Filtering và chọn 2016. Tại filters month_name, chọn Basic Filtering và chọn tháng cần truy vấn. Tại Filter country, chọn country cần truy vấn → Làm tương tự cho các quốc gia khác.

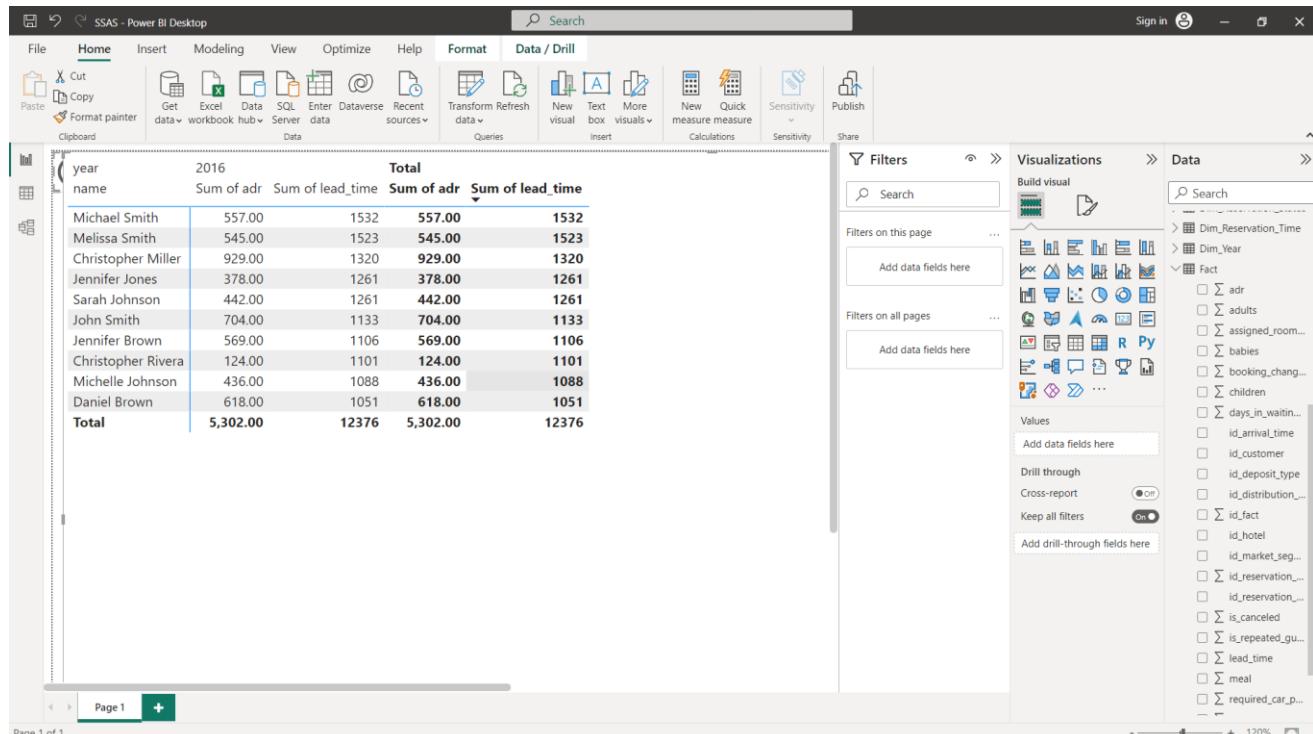


Figure 320. Kết quả Power BI câu 14 (năm 2016)

- BI Chart

Chọn kiểu Line and Stacked Column Chart. Kéo thả thuộc tính name từ Dim_Customer vào X-axis và year từ Dim_Year vào Legend → Kéo thả thuộc tính adr từ bảng Fact vào Line Y-axis và lead-time từ bảng Fact vào Column Y-axis. Tại filter year chọn giá trị lọc là 2016, tại filter name chọn Top 10 lọc theo giá trị Sum of lead_time.

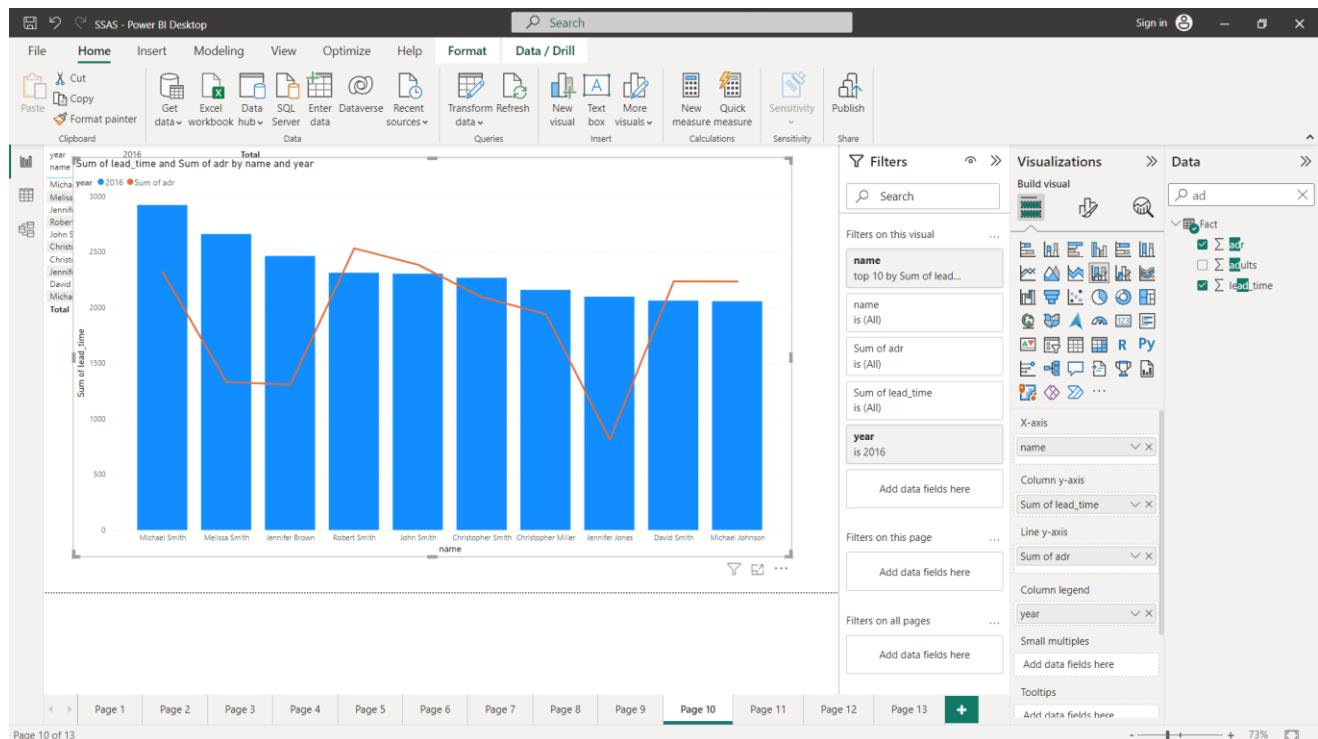


Figure 321. Kết quả trực quan Power BI câu 14

- Pivot Chart Excel

Kéo thả thuộc tính year từ Dim_Year vào Filters và name từ Dim_Customer vào Legend → Kéo thả thuộc tính adr và lead_time từ bảng Fact. Tại year chọn giá trị cần lọc là 2016, tại name chọn Value Filters chọn Top 10 theo Sum of lead_time.

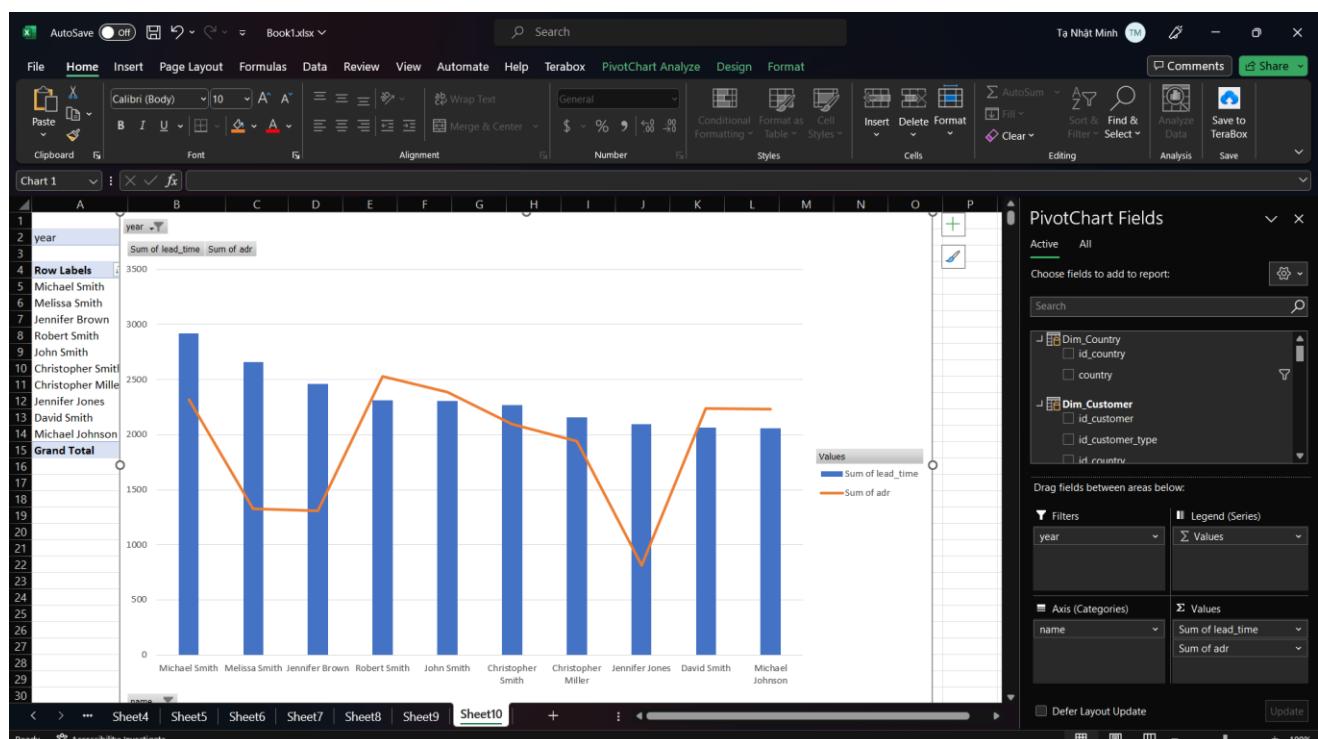


Figure 322. Kết quả Excel Pivot Chart câu 14

15. Thống kê thông tin và doanh thu khách hàng đặt phòng với hình thức không đặt cọc trước và thuộc phân khúc thị trường đặt phòng trực tiếp.

Ý nghĩa câu truy vấn:

Cho biết tiềm năng của nhóm khách thuộc hình thức không đặt cọc trước của phân khúc thị trường đặt phòng trực tiếp

- Công cụ SSAS trên các khối Cube

Kéo thả Name, Phone-number, Email từ Dim Customer vào vùng truy vấn → Kéo thả Market Segment từ Dim Market Segment và Deposit Type từ Dim Deposit Type vào bộ lọc → Kéo thả Adr từ Measure vào vùng truy vấn. Chọn click to execute the query.

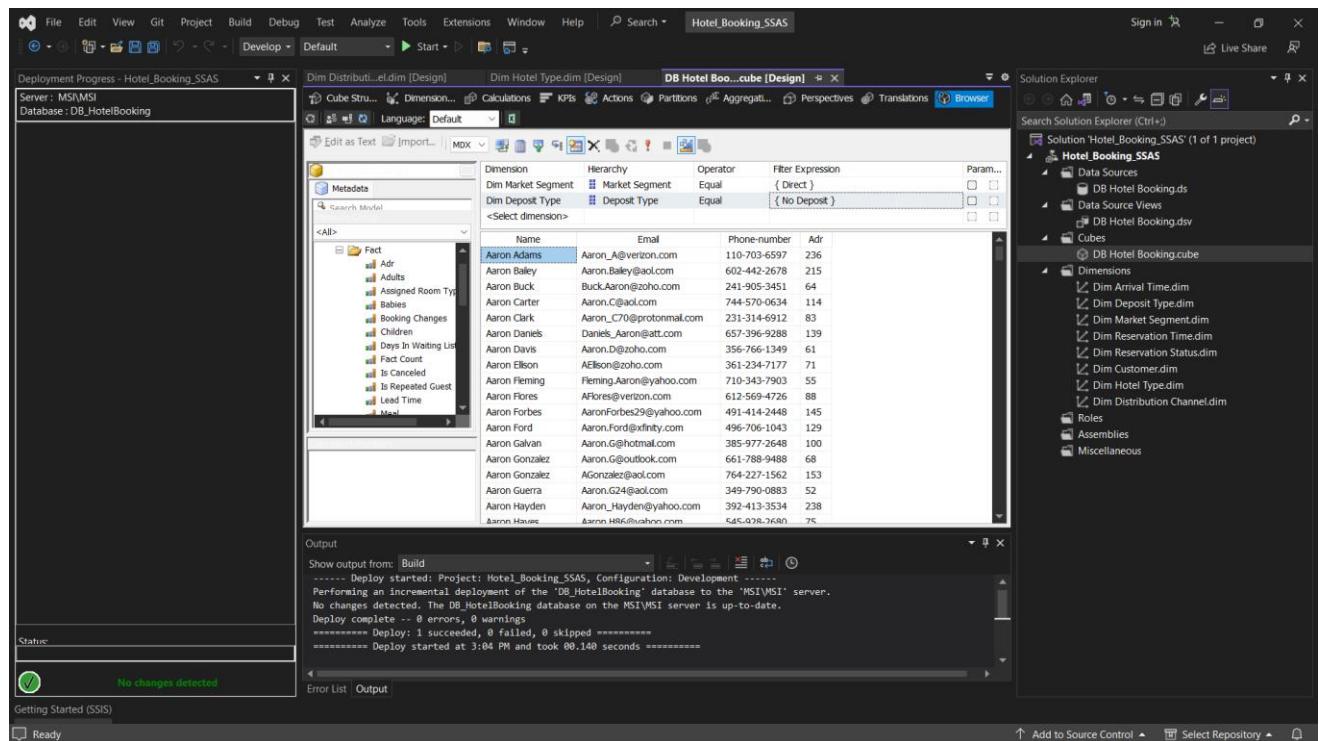


Figure 323. Kéo thả các trường cần dùng để truy vấn

- Ngôn ngữ MDX trên các khối Cube

Query:

```

select non empty {[Measures].[Adr]} on columns,
non empty {[Dim Customer].[Name].children*[Dim Customer].[Email].children*[Dim Customer].[Phone-number].children} on rows
from [DB Hotel Booking]
where ([Dim Deposit Type].[Deposit Type].&[No Deposit], [Dim Market Segment].[Market Segment].&[Direct])

```

Kết quả:

IS217.N21 – Kho dữ liệu và phân tích hoạt động đặt phòng khách sạn

The screenshot shows the SSMS interface with the following details:

- Object Explorer:** Shows the database structure, including the DB_HotelBooking database.
- MDX Query Editor:** Contains the MDX query:


```
select non empty {[Measures].[adr]} on columns,
non empty {[Dim Customer].[Name].children*[Dim Customer].[Email].children*[Dim Customer].[Phone-number].children} on rows
from [DB Hotel Booking]
where {[Dim Deposit Type].[Deposit Type].&[No Deposit], [Dim Market Segment].[Market Segment].&[Direct]}
```
- Results Grid:** Displays the query results in a grid format. The columns are Adr, name, email, phone-number, deposit_type, and market_segment. The rows list various customer entries.
- Status Bar:** Shows "Query executed successfully." and the execution time "00:00:03".

Figure 324. Kết quả MDX câu 15

- Công cụ Power BI

Kéo market_segment từ Dim_Maret_Segment, adr từ Fact, deposit_type từ Dim_Deposit_Type và name, phone-number, email từ Dim_Customer vào trường Columns.

The screenshot shows the Power BI desktop interface with the following details:

- Home Tab:** Selected.
- Data / Drill Tab:** Selected.
- Table Visual:** A table showing the following data:

	name	email	phone-number	deposit_type	market_segment	Sum of adr
Aaron Adams	Aaron.A@verizon.com	110-703-6597	No Deposit	Direct	236.00	
Aaron Bailey	Aaron.Bailey@aol.com	602-442-2678	No Deposit	Direct	215.00	
Aaron Buck	Buck.Aaron@zoho.com	241-905-3451	No Deposit	Direct	64.00	
Aaron Carter	Aaron.C@oak.com	744-570-0634	No Deposit	Direct	114.00	
Aaron Clark	Aaron.C70@protonmail.com	231-314-6912	No Deposit	Direct	83.00	
Aaron Daniels	Daniels_Aaron@att.com	657-396-9288	No Deposit	Direct	139.00	
Aaron Davis	Aaron.D@zoho.com	356-766-1349	No Deposit	Direct	61.00	
Aaron Ellison	AEllison@zoho.com	361-234-7177	No Deposit	Direct	71.00	
Aaron Fleming	Fleming.Aaron@yahoo.com	710-343-1043	No Deposit	Direct	65.00	
Aaron Flores	AFlores@verizon.com	612-569-4726	No Deposit	Direct	88.00	
Aaron Forbes	AaronForbes29@yahoo.com	491-414-2448	No Deposit	Direct	145.00	
Aaron Ford	Aaron.Ford@xfinity.com	496-706-1043	No Deposit	Direct	129.00	
Aaron Galvan	Aaron.G@hotmail.com	385-977-2648	No Deposit	Direct	100.00	
Aaron Gonzalez	Aaron.G@outlook.com	661-788-9488	No Deposit	Direct	68.00	
Aaron Gonzalez	AGonzalez@aol.com	764-227-1562	No Deposit	Direct	153.00	
Aaron Guerra	Aaron.G24@aol.com	349-790-0883	No Deposit	Direct	52.00	
Aaron Hayden	Aaron.Hayden@yahoo.com	392-413-3534	No Deposit	Direct	238.00	
Aaron Hayes	Aaron.H86@yahoo.com	545-928-2680	No Deposit	Direct	75.00	
Aaron Hebert	Aaron.Hebert@outlook.com	139-810-1467	No Deposit	Direct	138.00	
Aaron Hernandez	Aaron.Hernandez@comcast.net	779-437-4240	No Deposit	Direct	131.00	
Aaron Herrera	Aaron.H@gmail.com	709-888-3590	No Deposit	Direct	165.00	
Aaron Hughes	Aaron.H64@hotmail.com	502-845-1288	No Deposit	Direct	95.00	
Aaron Hughes	Hughes_Aaron@zoho.com	557-688-8077	No Deposit	Direct	51.00	
Aaron Jackson	Aaron.Jackson@aol.com	253-742-6126	No Deposit	Direct	39.00	
Aaron Jackson	Aaron.Jackson02@yahoo.com	153-046-0986	No Deposit	Direct	175.00	
Aaron Johnson	Aaron.Johnson48@verizon.com	550-121-9288	No Deposit	Direct	80.00	
Aaron Johnson	A.Johnson@zoho.com	822-850-9150	No Deposit	Direct	83.00	
Total						1,435,183.00
- Fields Pane:** Shows the selected fields: deposit_type, market_segment, name, phone-number, and Sum of adr.

Figure 325. Kết quả Power BI câu 15

CHƯƠNG 4. KHAI PHÁ DỮ LIỆU

4.1. Khởi tạo môi trường khai phá dữ liệu

- **Bước 1:** Vào ứng dụng Visual studio code → Tạo file có đuôi .ipynb.

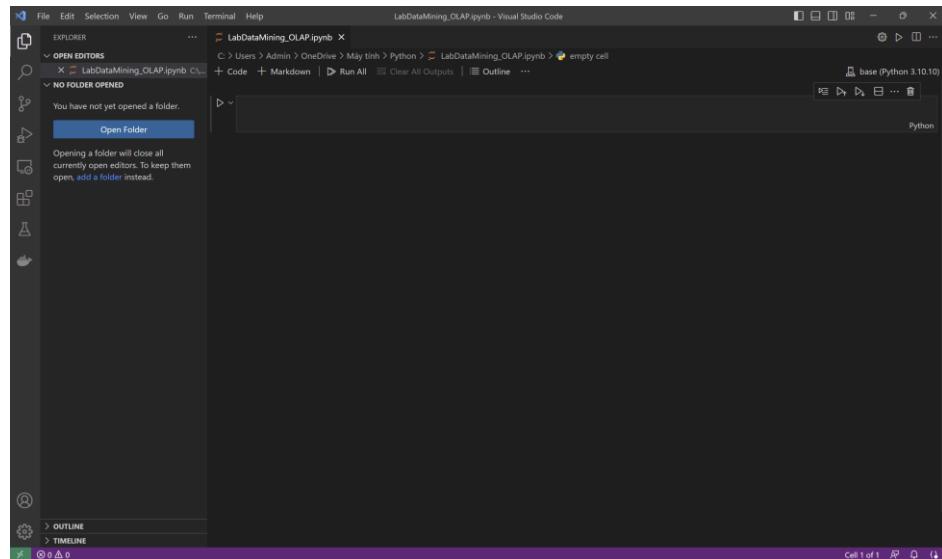


Figure 326. Giao diện môi trường khai phá dữ liệu với Jupiter Notebook

- **Bước 2:** Khai báo các thư viện sử dụng cho quá trình khai phá dữ liệu.

```
# Thư viện xử lý với mảng
import numpy as np
import pandas as pd

# Thư viện vẽ biểu đồ
import matplotlib.pyplot as plt
import seaborn as sns
from mpl_toolkits import mplot3d

# Thư viện tiền xử lý dữ liệu
from sklearn.model_selection import train_test_split
from sklearn.feature_selection import SelectKBest, chi2
from sklearn.preprocessing import StandardScaler

# Thư viện các mô hình Machine Learning
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from sklearn.svm import SVC
from sklearn.ensemble import RandomForestClassifier
from xgboost import XGBClassifier
from sklearn.cluster import KMeans

# Thư viện các mô hình Deep learning
from tensorflow import keras
from keras.models import Sequential
from keras.layers import Dense

# Thư viện đánh giá hiệu quả của các mô hình học máy
from sklearn.metrics import f1_score
from sklearn.metrics import accuracy_score, classification_report, precision_score, recall_score
from sklearn.metrics import confusion_matrix

import warnings
```

Figure 327. Các thư viện sử dụng cho quá trình khai phá

- **Bước 3:** Đọc dữ liệu từ file .csv về hoạt động đặt phòng khách sạn của khách hàng và hiển thị 5 dòng dữ liệu ngẫu nhiên.

```
df = pd.read_csv('hotel_booking_cleaned.csv')
df.sample(5)
```

Python

	hotel	is_canceled	lead_time	arrival_date_year	arrival_date_month_name	arrival_date_week_number	arrival_date_day_of_month	stays_in_weekend_nights	stays_in_week_nights
50836	City Hotel	1	16	2016	May	23		29	
10050	Resort Hotel	1	8	2017	March	9		3	
63223	City Hotel	1	608	2017	March	10		9	
109406	City Hotel	0	14	2017	April	17		25	
71582	City Hotel	1	98	2017	August	31		5	

5 rows × 43 columns

Figure 328. Thông tin 5 dòng dữ liệu bất kỳ trong bộ dữ liệu

4.2. Khảo sát dữ liệu

✚ Xóa các thuộc tính không ảnh hưởng đến việc đặt phòng

Các thuộc tính thời gian và các thông tin của khách hàng không ảnh hưởng đến việc dự đoán khách hàng đó có đặt phòng hay không, chính vì vậy đề tài quyết định không sử dụng các thuộc tính này hỗ trợ cho việc phân tích và xây dựng mô hình dự đoán.

- Thuộc tính về thời gian:
 - o 'id_reservation_status_date'
 - o 'id_arrival_date'
 - o 'arrival_date_year'
 - o 'arrival_date_month_name'
 - o 'arrival_date_week_number'
 - o 'arrival_date_day_of_month'
 - o 'arrival_date_month_number'
 - o 'arrival_date_full'
 - o 'arrival_date_quarter'
 - o 'arrival_date_day_of_week'
 - o 'arrival_date_day_name'
 - o 'arrival_date_day_name_abbrev'
 - o 'arrival_date_month_name_abbrev'
 - o 'arrival_date_weekday_flag'
 - o 'reservation_status_date'
- Thuộc tính về thông tin khách hàng:
 - o 'name'
 - o 'email'
 - o 'phone_number'
 - o 'country'

Mã lệnh xóa các thuộc tính không cần thiết:

```
# khong lay cac cot ngay thang nam vao thuc hien du doan nhan 0, 1  
drop_column = ['is_canceled','arrival_date_year','arrival_date_month_name','arrival_date_week_number','arrival_date_day_of_month','coun  
X = df.drop(drop_column, axis = 1)
```

Python

❖ **Thống kê mô tả dữ liệu (EDA):**

```
X.info()  
  
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 118216 entries, 0 to 118215  
Data columns (total 23 columns):  
 # Column Non-Null Count Dtype  
--- ---  
 0 hotel 118216 non-null object  
 1 lead_time 118216 non-null int64  
 2 stays_in_weekend_nights 118216 non-null int64  
 3 stays_in_week_nights 118216 non-null int64  
 4 adults 118216 non-null int64  
 5 children 118216 non-null int64  
 6 babies 118216 non-null int64  
 7 meal 118216 non-null int64  
 8 market_segment 118216 non-null object  
 9 distribution_channel 118216 non-null object  
 10 is_repeated_guest 118216 non-null int64  
 11 previous_cancellations 118216 non-null int64  
 12 previous_bookings_not_canceled 118216 non-null int64  
 13 reserved_room_type 118216 non-null int64  
 14 assigned_room_type 118216 non-null int64  
 15 booking_changes 118216 non-null int64  
 16 deposit_type 118216 non-null object  
 17 days_in_waiting_list 118216 non-null int64  
 18 customer_type 118216 non-null object  
 19 adr 118216 non-null float64  
 20 required_car_parking_spaces 118216 non-null int64  
 21 total_of_special_requests 118216 non-null int64  
 22 reservation_status 118216 non-null object  
dtypes: float64(1), int64(16), object(6)  
memory usage: 20.7+ MB
```

Figure 329. Thông tin các thuộc tính dùng để khai phá dữ liệu

Mã lệnh trực quan hóa nhãn cần dự đoán:

```
fig,(ax1,ax2) = plt.subplots(figsize=(20,10),ncols=2,nrows=1)  
  
label_IsCanceled = ['Uncanceled','Canceled']  
textprops = {"fontsize":20}  
  
ax1.pie(y_ratio_sales, autopct='%1.2f%%', shadow = True, textprops = textprops)  
ax1.set_title('Tí lệ nhàn hủy phòng và không hủy phòng',fontsize = 20)  
ax1.legend(label_IsCanceled,title = "",loc='upper left',fontsize=18);  
  
ax2.bar(["Uncanceled","Canceled"], y_ratio_sales.values, color='blueviolet');  
ax2.set_title('Số lượng nhàn hủy phòng và không hủy phòng',fontsize = 20)  
plt.yticks(size=18)  
plt.xticks(size=18)
```

[9] Python

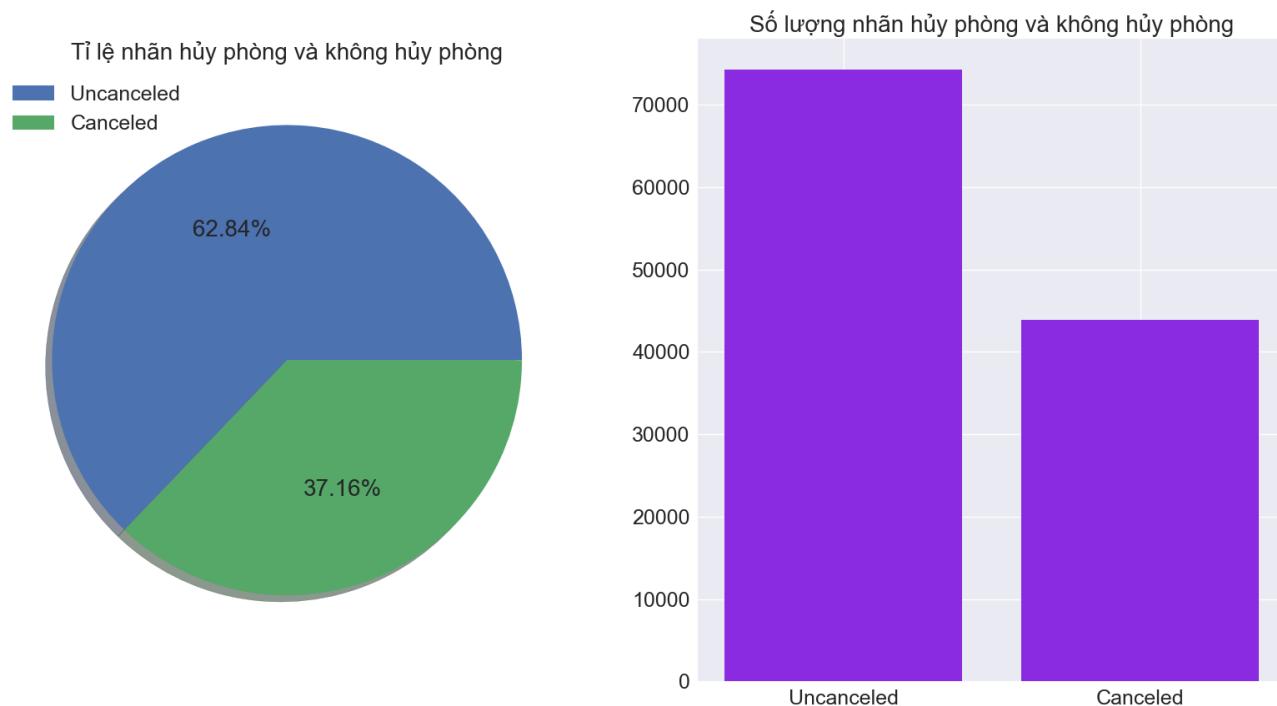


Figure 330. Biểu đồ số lượng đặt phòng và hủy phòng

Nhận xét về tình trạng đặt phòng của bộ dữ liệu:

- Tỉ lệ hủy phòng của khách hàng gần bằng 1/3 tổng bộ dữ liệu (37,16%), **tỉ lệ này khá cao**. Tức là, có 3 giao dịch đặt phòng sẽ có 1 giao dịch đặt phòng bị hủy.

4.3. Tiết xử lý dữ liệu

4.3.1. Xử lý dữ liệu biến rời rạc

Table 10 . Bảng mã hóa các giá trị category thành numeric

Thuộc tính	Giá trị chuyển đổi
hotel_type	<ul style="list-style-type: none"> • 0: 'City Hotel' • 1: 'Resort Hotel'
market_segment	<ul style="list-style-type: none"> • 0: 'Online TA', • 1: 'Offline TA/TO', • 2: 'Groups', • 3: 'Direct', • 4: 'Corporate' • 5: 'Complementary' • 6: 'Aviation'
distribution_channel	<ul style="list-style-type: none"> • 0: 'TA/TO' • 1: 'Direct' • 2: 'Corporate' • 3: 'GDS'

deposit_type	<ul style="list-style-type: none"> • 0: 'No Deposit' • 1: 'Non Refund' • 2: 'Refundable'
customer_type	<ul style="list-style-type: none"> • 0: 'Transient' • 1: 'Transient-Party' • 2: 'Contract' • 3: 'Group'
reservation_status	<ul style="list-style-type: none"> • 0: 'Check-Out' • 1: 'Canceled' • 2: 'No-Show'

Mã lệnh xử lý dữ liệu category:

```
# Xử lý cho biến rời rạc
df['hotel'].value_counts()
hotel_type = {'City Hotel': 0, 'Resort Hotel':1}
for i in [X]:
    i['hotel'] = i['hotel'].map(hotel_type)

df['market_segment'].value_counts()
mar_type = {'Online TA': 0, 'Offline TA/TO':1, 'Groups':2, 'Direct':3, 'Corporate':4, 'Complementary':5, 'Aviation':6}
for i in [X]:
    i['market_segment'] = i['market_segment'].map(mar_type)

df['distribution_channel'].value_counts()
dis_type = {'TA/TO': 0, 'Direct':1, 'Corporate':2, 'GDS':3}
for i in [X]:
    i['distribution_channel'] = i['distribution_channel'].map(dis_type)

df['deposit_type'].value_counts()
des_type = {'No Deposit': 0, 'Non Refund':1, 'Refundable':2}
for i in [X]:
    i['deposit_type'] = i['deposit_type'].map(des_type)

df['customer_type'].value_counts()
cus_type = {'Transient': 0, 'Transient-Party':1, 'Contract':2, 'Group':3}
for i in [X]:
    i['customer_type'] = i['customer_type'].map(cus_type)

df['reservation_status'].value_counts()
res_type = {'Check-Out': 0, 'Canceled':1, 'No-Show':2}
for i in [X]:
    i['reservation_status'] = i['reservation_status'].map(res_type)
```

Python

Mã lệnh trực quan hóa các biến category:

```
fig, ax = plt.subplots(figsize=(20,20), ncols=3, nrows=3)
textprops = {"fontsize":20}

X['hotel'].value_counts().plot(kind='bar',ax=ax[0,0])
X['meal'].value_counts().plot(kind='bar',ax=ax[0,1])
X['market_segment'].value_counts().plot(kind='bar',ax=ax[0,2])

X['distribution_channel'].value_counts().plot(kind='bar',ax=ax[1,0])
X['is_repeated_guest'].value_counts().plot(kind='bar',ax=ax[1,1])
X['customer_type'].value_counts().plot(kind='bar',ax=ax[1,2])

X['total_of_special_requests'].value_counts().plot(kind='bar',ax=ax[2,0])
X['reservation_status'].value_counts().plot(kind='bar',ax=ax[2,1])
X['deposit_type'].value_counts().plot(kind='bar',ax=ax[2,2])
```

Python

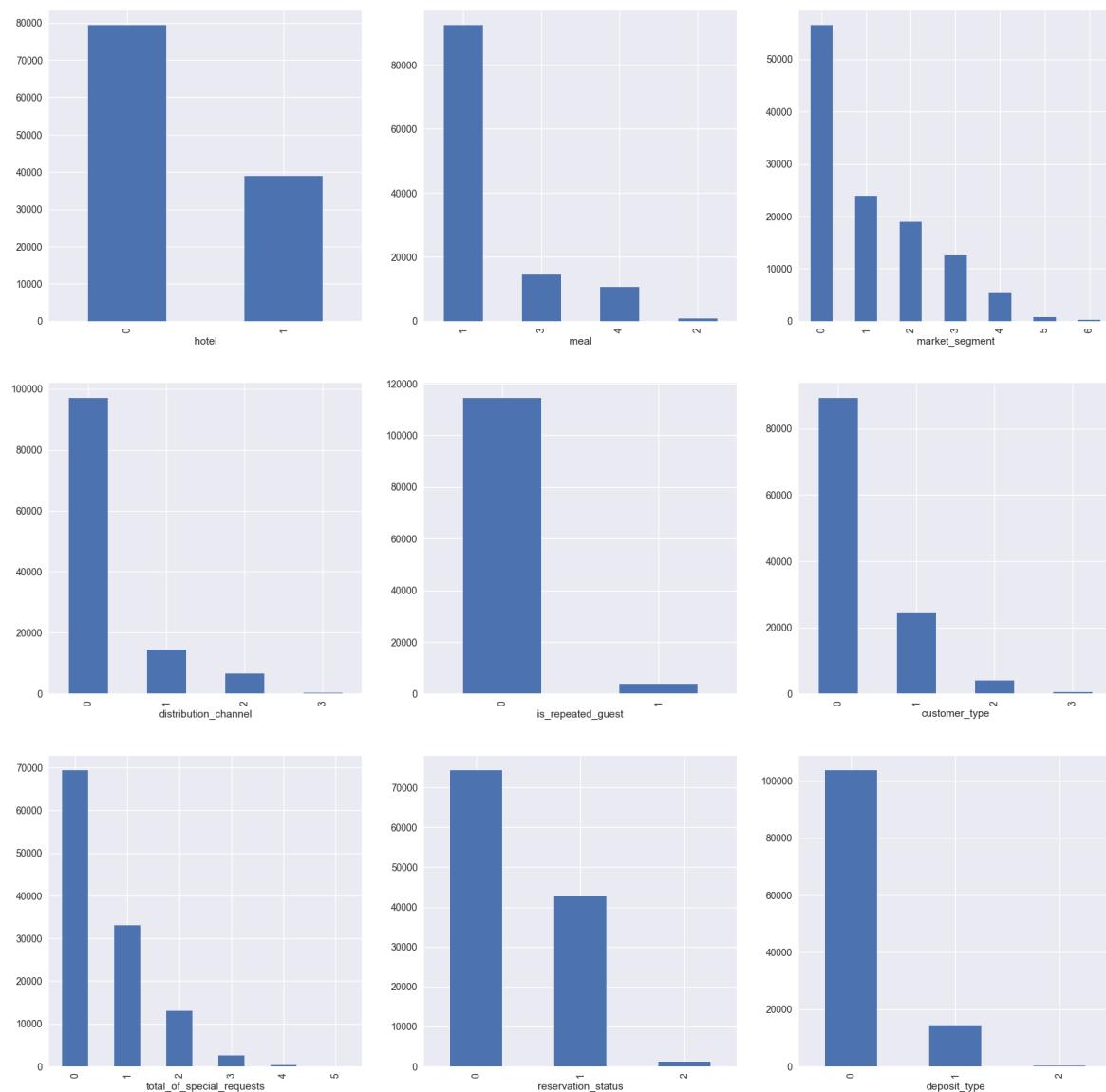


Figure 331. Biểu đồ phân phối của các nhãn category của bộ dữ liệu

4.3.2. Xử lý dữ liệu biến liên tục

Để thực hiện xử lý các giá trị ngoại lai, để tài sử dụng công thức xử lý dữ liệu outlier theo IQR dưới đây, các giá trị nằm ở khoảng thấp hơn Lower Bound và cao hơn Higher Bound sẽ được xóa ra khỏi bộ dữ liệu:

$$IQR = Q_3 - Q_1$$

$$Lower\ Bound = Q_3 - 1,5 * IQR$$

$$Higher\ Bound = Q_3 + 1,5 * IQR$$

Với (Q3) là giá trị tại điểm $\frac{3}{4}$ là (Q1) là giá trị tại điểm $\frac{1}{4}$ của dữ liệu ứng với phân phối của chúng.

Công thức này trong để tài chỉ xử lý với các thuộc tính dạng numeric và có phân phối tương đối chuẩn (lead_time, adr), còn các thuộc tính khác không có dạng

chuẩn, chính vì vậy sẽ không được lựa chọn để xử lý. Bởi vì nếu xử lý các thuộc tính này sẽ rất dễ dẫn đến mất các dòng dữ liệu quan trọng,

Mã lệnh xử lý dữ liệu numeric:

```
# Xử lý cho biến liên tục
percentile25 = X['adr'].quantile(0.25)
percentile75 = X['adr'].quantile(0.75)

iqr = percentile75 - percentile25
Upperlimit = percentile75 + 1.5 * iqr
Lowerlimit = percentile25 - 1.5 * iqr
X = X[(X['adr'] < Upperlimit) & (X['adr'] > Lowerlimit)]

percentile25 = X['lead_time'].quantile(0.25)
percentile75 = X['lead_time'].quantile(0.75)

iqr = percentile75 - percentile25
Upperlimit = percentile75 + 1.5 * iqr
Lowerlimit = percentile25 - 1.5 * iqr

X = X[(X['lead_time'] < Upperlimit) & (X['lead_time'] > Lowerlimit)]
```

Python

Mã lệnh trực quan hóa các thuộc tính numeric:

```
fig, ax = plt.subplots(figsize=(20,20), ncols=4, nrows=4)
textprops = {"fontsize":20}

sns.histplot(x = X['lead_time'],ax=ax[0,0])
sns.histplot(x = X['stays_in_weekend_nights'],ax=ax[0,1])
sns.histplot(x = X['stays_in_week_nights'],ax=ax[0,2])
sns.histplot(x = X['adults'],ax=ax[0,3])

sns.histplot(x = X['children'],ax=ax[1,0])
sns.histplot(x = X['babies'],ax=ax[1,1])
sns.histplot(x = X['previous_cancellations'],ax=ax[1,2])
sns.histplot(x = X['previous_bookings_notCanceled'],ax=ax[1,3])

sns.histplot(x = X['reserved_room_type'],ax=ax[2,0])
sns.histplot(x = X['assigned_room_type'],ax=ax[2,1])
sns.histplot(x = X['booking_changes'],ax=ax[2,2])
sns.histplot(x = X['days_in_waiting_list'],ax=ax[2,3])

sns.histplot(x = X['adr'],ax=ax[3,0])
sns.histplot(x = X['required_car_parking_spaces'],ax=ax[3,1]);
```

Python

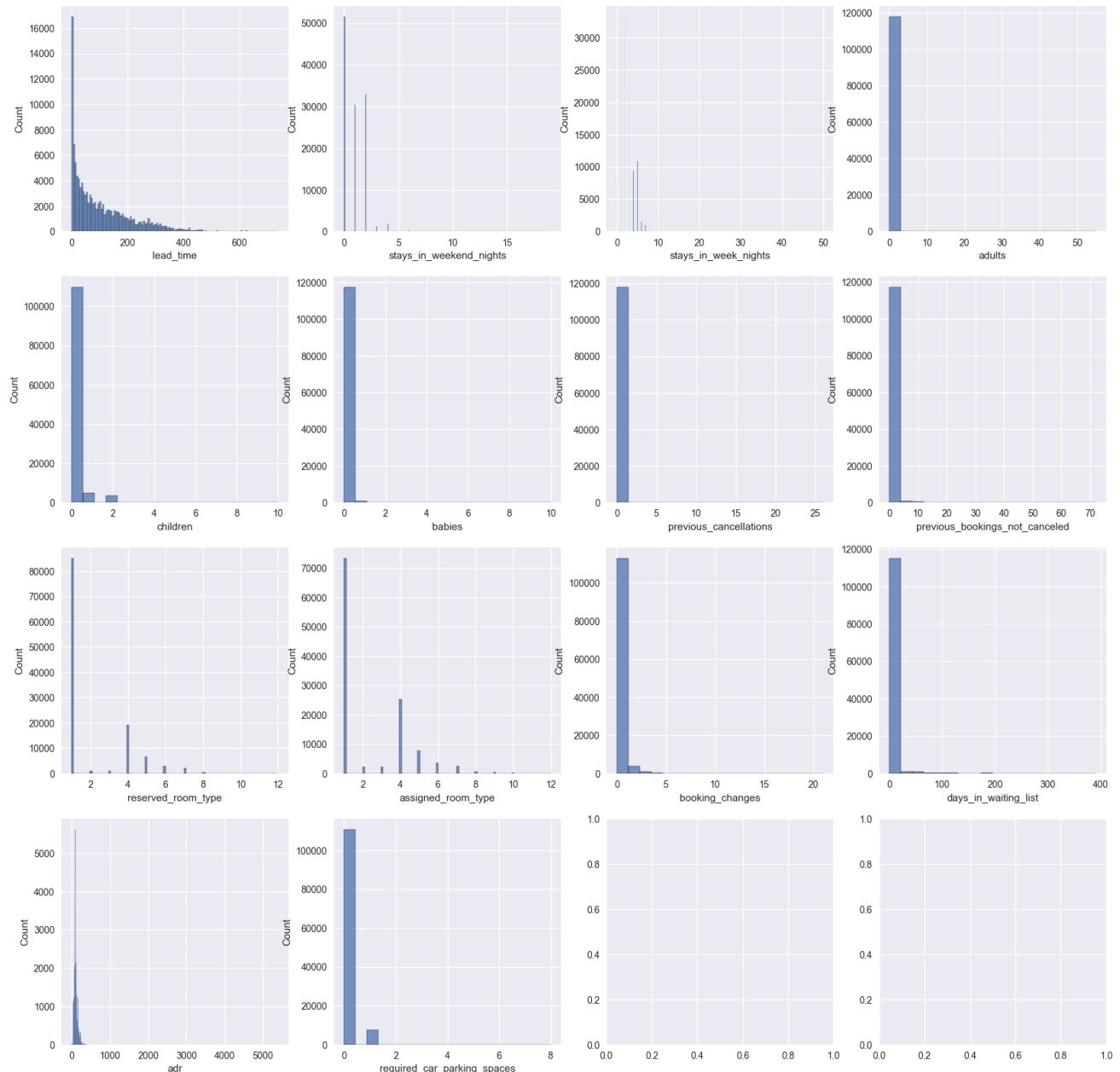


Figure 332. Biểu đồ dữ liệu numeric trước tiền xử lý

IS217.N21 – Kho dữ liệu và phân tích hoạt động đặt phòng khách sạn

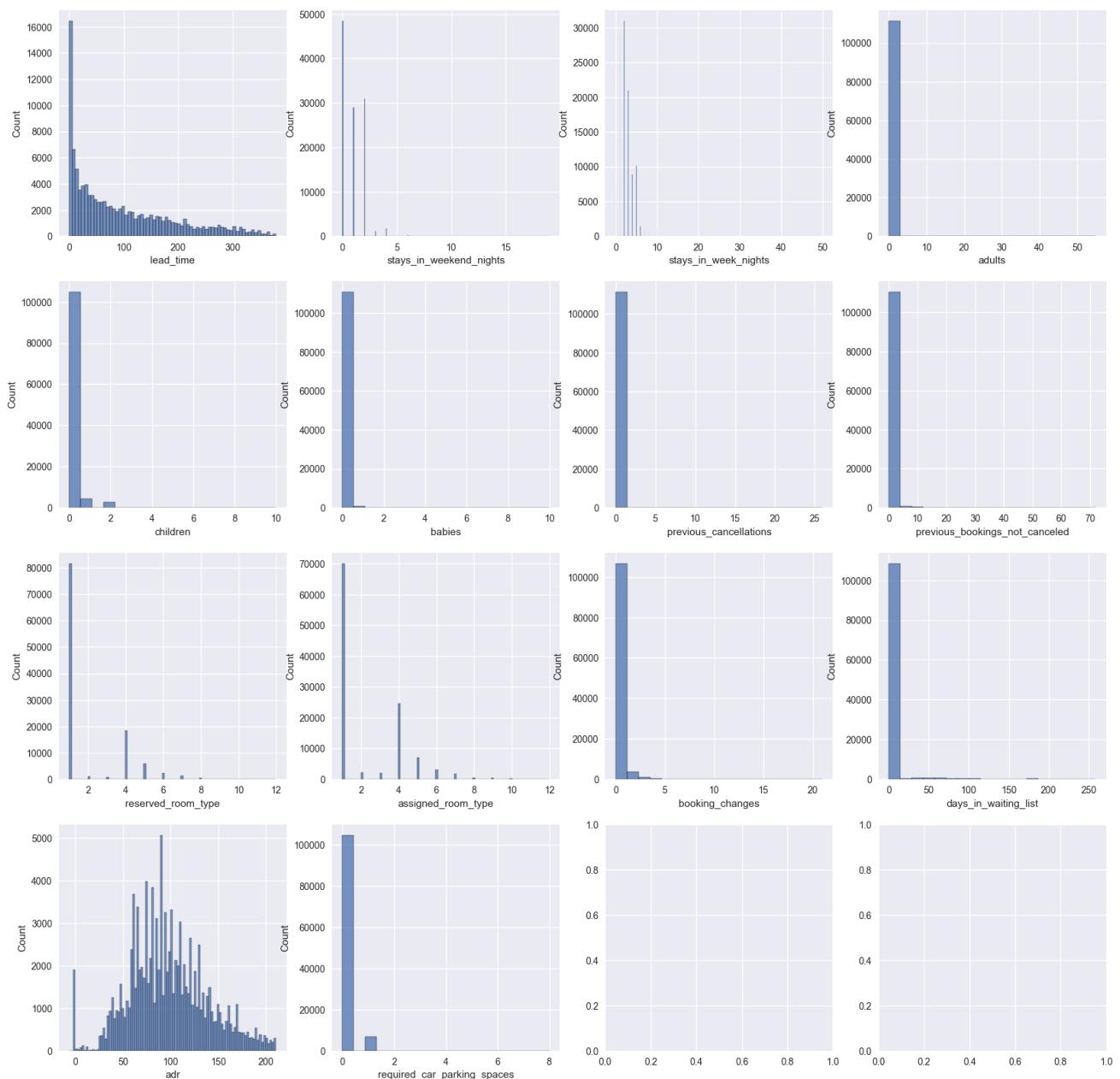


Figure 333. Biểu đồ dữ liệu numeric sau tiền xử lý

X	Python
hotel	
lead_time	
stays_in_weekend_nights	
stays_in_week_nights	
adults	
children	
babies	
meal	
market_segment	
distribution_channel	
adr	
required_car_parking_spaces	
booking_changes	
days_in_waiting_list	
is_canceled	
...	
118210	1
118211	0
118213	0
118214	0
118215	0
111607 rows × 23 columns	

Figure 334. Dữ liệu khi sau khi tiền xử lý (không có thuộc tính *is_canceled*)

4.4. Lựa chọn các thuộc tính ảnh hưởng đến việc hủy phòng của khách hàng

Mã lệnh tìm các biến ảnh hưởng tới biến đặt phòng của khách hàng dựa trên Chi-square:

Table 4.4.1. Bảng các yếu tố ảnh hưởng tới biến hủy phòng

Input		
STT	Thuộc tính	Giải thích thuộc tính
1	hotel	Kiểu khách sạn.
2	lead_time	Số ngày từ ngày khách hàng đặt chỗ đến ngày khách hàng đến nhận phòng.
3	market_segment	Thể hiện các phân khúc khách hàng của khách sạn.
4	distribution_channel	Các loại liên kết đặt phòng khách sạn.
5	is_repeated_guest	Cho biết lượt đặt chỗ có phải từ khách hàng đã lưu trú nhiều lần tại khách sạn hay là lần đầu đến.
6	previous_cancellations	Số lượng lượt đặt phòng đã bị khách hàng hủy trước lượt đặt phòng hiện tại.
7	previous_bookings_not_canceled	Số lượng lượt nhận phòng thành công của khách hàng trước lượt đặt phòng hiện tại.
8	reserved_room_type	Mã loại phòng được đại diện bằng các chữ cái.
9	assigned_room_type	Mã loại phòng được chỉ định cho lượt đặt phòng. Đôi khi loại phòng được chỉ định khác với loại phòng đã đặt vì lý do hoạt động của khách sạn (khách sạn nhận số lượng đặt phòng nhiều hơn số lượng phòng thực tế) hoặc do yêu cầu của khách hàng.
10	booking_changes	Số lượng sửa đổi đối với lượt đặt phòng kể từ thời điểm đặt phòng được nhập trên PMS cho đến thời điểm nhận phòng hoặc lượt đặt phòng bị hủy bỏ.
11	deposit_type	Thể hiện hình thức đặt cọc trước.
12	days_in_waiting_list	Số ngày lượt đặt chỗ ở trong danh sách chờ trước khi nó được xác nhận với khách hàng.
13	customer_type	Thể hiện loại khách hàng.
14	required_car_parking_spaces	Số chỗ đậu xe ô tô theo yêu cầu của khách hàng.

15	total_of_special_requests	Số lượng yêu cầu đặc biệt của khách hàng (ví dụ: có giường đôi hoặc ở tầng cao).
Output		
1	is_canceled	Hủy hoặc không hủy đặt phòng.

4.5. Tiền xử lý dữ liệu huấn luyện mô hình dự đoán hủy đặt phòng

4.5.1. Chuẩn hóa dữ liệu

Để tránh tình trạng những thuộc có giá trị lớn sẽ thiêng vị các thuộc tính có giá trị nhỏ làm ảnh hưởng đến mô hình máy học, nên chúng tôi đã thực hiện phương pháp trích xuất đặc trưng để cải thiện hiệu xuất mô hình máy học. Ở bài báo cáo này, đề tài đã sử dụng phương pháp chuẩn hóa Standardizing scores (Z-score), có công thức như sau:

$$X_{\text{new}} = \frac{X - \mu}{\sigma}$$

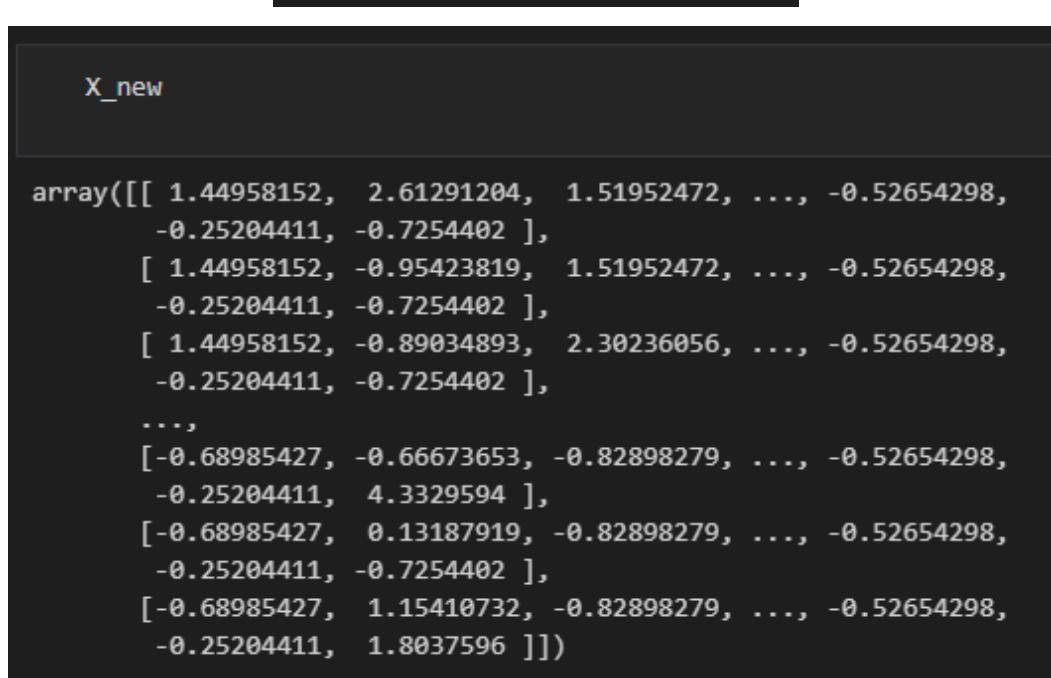
Với X_{new} là giá trị mới, X là giá trị cần chuẩn hóa, μ là trung bình và σ là độ lệch chuẩn của phân phối thuộc tính đó.

Mã lệnh chuẩn hóa dữ liệu bằng python:

```

sc = StandardScaler()
X_new = sc.fit_transform(X_new)

```



```

X_new

array([[ 1.44958152,  2.61291204,  1.51952472, ..., -0.52654298,
       -0.25204411, -0.7254402 ],
       [ 1.44958152, -0.95423819,  1.51952472, ..., -0.52654298,
       -0.25204411, -0.7254402 ],
       [ 1.44958152, -0.89034893,  2.30236056, ..., -0.52654298,
       -0.25204411, -0.7254402 ],
       ...,
       [-0.68985427, -0.66673653, -0.82898279, ..., -0.52654298,
       -0.25204411,  4.3329594 ],
       [-0.68985427,  0.13187919, -0.82898279, ..., -0.52654298,
       -0.25204411, -0.7254402 ],
       [-0.68985427,  1.15410732, -0.82898279, ..., -0.52654298,
       -0.25204411,  1.8037596 ]])

```

Figure 335. Dữ liệu khi sau khi chuẩn hóa

4.5.2. Chia dữ liệu huấn luyện

Dữ liệu được chúng tôi chia thành 2 tập huấn luyện và kiểm thử với tỉ lệ 7:3 (tương ứng với 78124 dòng dữ liệu và 33482 dòng dữ liệu). Mục đích của việc phân tách dữ liệu là để tránh overfitting. Nếu bị overfitting, các thuật toán học máy có thể hoạt động tốt trong các tập dữ liệu train nhưng hoạt động kém trong tập dữ liệu test. Từ đó ta có thể nhận biết và điều chỉnh.

Mã lệnh chia dữ liệu huấn luyện và dữ liệu kiểm thử:

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X_new, y, random_state=0, test_size = 0.3)
print(X_train.shape, y_train.shape)

(78124, 15) (78124, 1)
```

4.6. Huấn luyện mô hình

4.6.1. Mô hình machine learning

1. Logistic regression

```
from sklearn.linear_model import LogisticRegression
model_log = LogisticRegression()
model_log.fit(X_train, y_train)

▼ LogisticRegression
LogisticRegression()
```

Figure 336. Huấn luyện mô hình Logistic regression

2. Decision tree

```
from sklearn.tree import DecisionTreeClassifier
model_dt = DecisionTreeClassifier()
model_dt.fit(X_train, y_train)

▼ DecisionTreeClassifier
DecisionTreeClassifier()
```

Figure 337. Huấn luyện mô hình Decision tree

3. Support vector machine

```
from sklearn.svm import SVC
model_svm = SVC()
model_svm.fit(X_train, y_train)

▼ SVC
SVC()
```

Figure 338. Huấn luyện mô hình Support vector machine

4. Random forest

```
from sklearn.ensemble import RandomForestClassifier  
model_rf = RandomForestClassifier()  
model_rf.fit(X_train, y_train)  
  
* RandomForestClassifier  
RandomForestClassifier()
```

Figure 339. Huấn luyện mô hình Random forest

5. XGBoost

```
from xgboost import XGBClassifier  
model_xgb = XGBClassifier(n_estimators=100)  
model_xgb.fit(X_train, y_train)
```

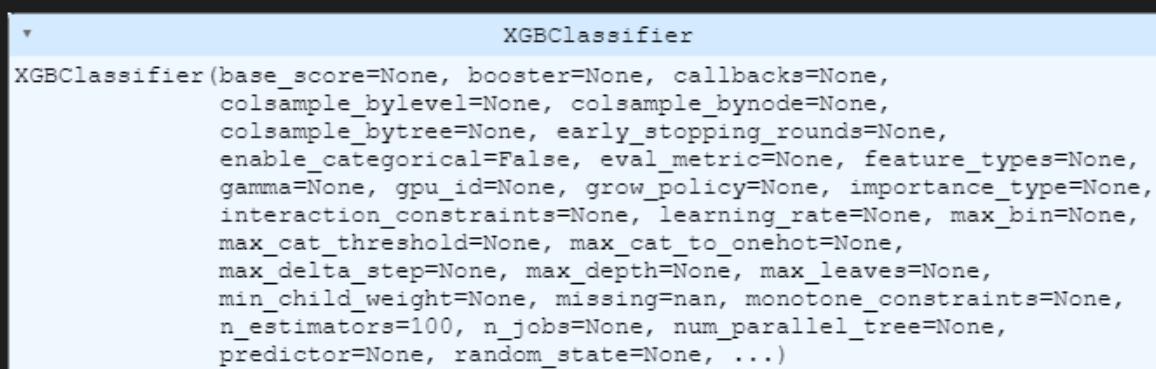



Figure 340. Huấn luyện mô hình XGBoost

4.6.2. Mô hình deep learning

6. Artificial neural network (ANN)

Mã lệnh xây dựng kiến trúc deep learning:

```
from tensorflow import keras  
from sklearn.model_selection import cross_val_score  
from keras.models import Sequential  
from keras.layers import Dense  
  
def build_classifier():  
    classifier = Sequential()  
    classifier.add(Dense(units = 8, kernel_initializer = 'uniform', activation = 'relu', input_dim = X_train.shape[1]))  
    classifier.add(Dense(units = 8, kernel_initializer = 'uniform', activation = 'relu'))  
    classifier.add(Dense(units = 1, kernel_initializer = 'uniform', activation = 'sigmoid'))  
    classifier.compile(optimizer = 'adam', loss = 'binary_crossentropy', metrics = ['accuracy'])  
    return classifier
```

Model: "sequential"		
Layer (type)	Output Shape	Param #
dense (Dense)	(None, 8)	128
dense_1 (Dense)	(None, 8)	72
dense_2 (Dense)	(None, 1)	9
<hr/>		
Total params: 209		
Trainable params: 209		
Non-trainable params: 0		
<hr/>		

Figure 341. Kiến trúc mô hình deep learning

```
history = model_ann.fit(x=X_train, y=y_train, batch_size=10, epochs=10, verbose=1, validation_data=(X_test, y_test))

Epoch 1/10
7813/7813 [=====] - 15s 2ms/step - loss: 0.4343 - accuracy: 0.7970 - val_loss: 0.4134 - val_accuracy: 0.8116
Epoch 2/10
7813/7813 [=====] - 15s 2ms/step - loss: 0.4119 - accuracy: 0.8084 - val_loss: 0.4053 - val_accuracy: 0.8103
Epoch 3/10
7813/7813 [=====] - 16s 2ms/step - loss: 0.4067 - accuracy: 0.8114 - val_loss: 0.4037 - val_accuracy: 0.8136
Epoch 4/10
7813/7813 [=====] - 14s 2ms/step - loss: 0.4021 - accuracy: 0.8124 - val_loss: 0.3969 - val_accuracy: 0.8163
Epoch 5/10
7813/7813 [=====] - 30s 4ms/step - loss: 0.3989 - accuracy: 0.8132 - val_loss: 0.3928 - val_accuracy: 0.8201
Epoch 6/10
7813/7813 [=====] - 26s 3ms/step - loss: 0.3969 - accuracy: 0.8141 - val_loss: 0.3929 - val_accuracy: 0.8183
Epoch 7/10
7813/7813 [=====] - 29s 4ms/step - loss: 0.3959 - accuracy: 0.8145 - val_loss: 0.3900 - val_accuracy: 0.8190
Epoch 8/10
7813/7813 [=====] - 31s 4ms/step - loss: 0.3951 - accuracy: 0.8138 - val_loss: 0.3945 - val_accuracy: 0.8171
Epoch 9/10
7813/7813 [=====] - 28s 4ms/step - loss: 0.3942 - accuracy: 0.8151 - val_loss: 0.3890 - val_accuracy: 0.8206
Epoch 10/10
7813/7813 [=====] - 18s 2ms/step - loss: 0.3936 - accuracy: 0.8152 - val_loss: 0.3891 - val_accuracy: 0.8201
```

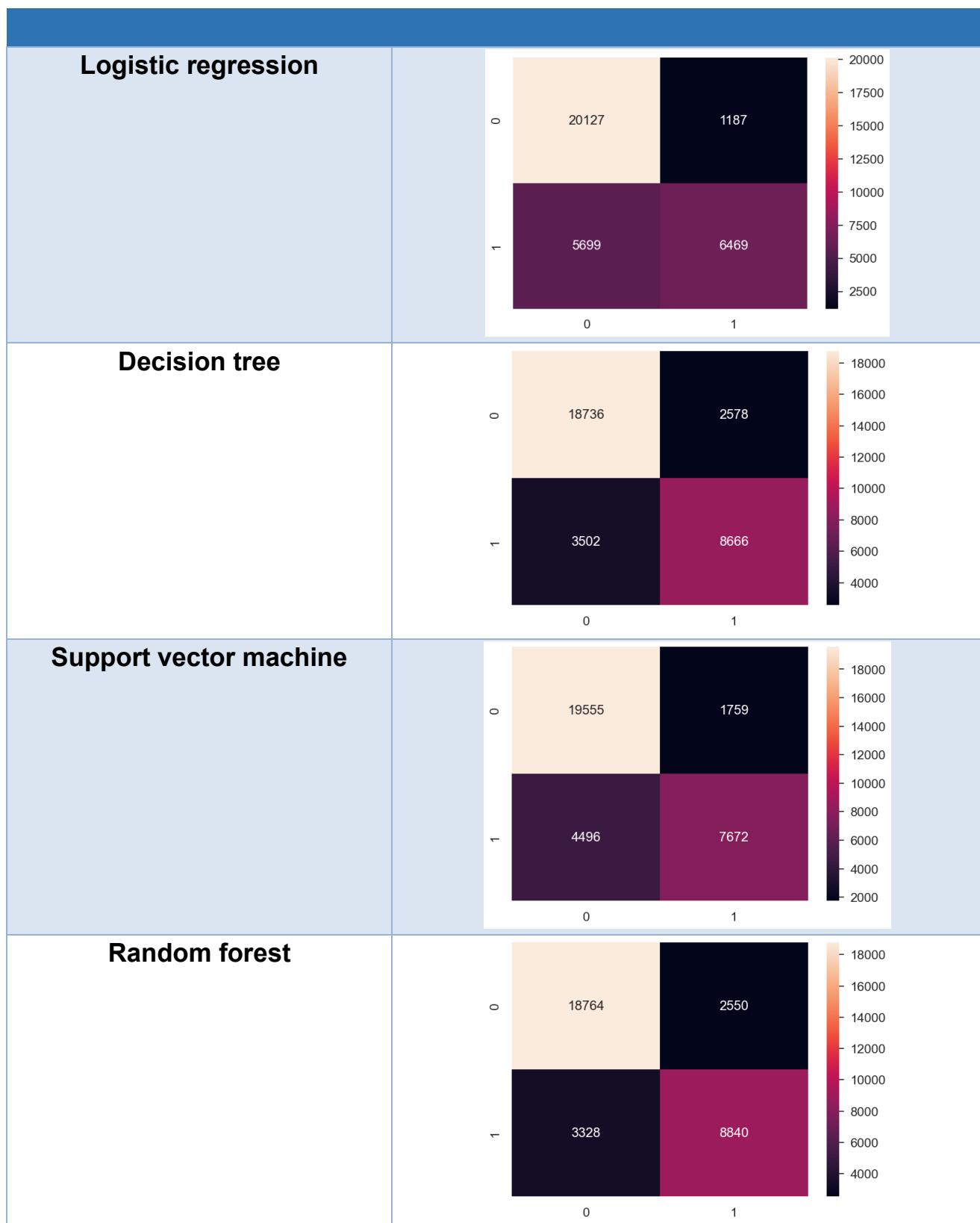
Figure 342. Huấn luyện kiến trúc trên tập dữ liệu huấn luyện

4.7. Đánh giá và so sánh các mô hình

Table 26. Bảng kết quả dự đoán trên tập dữ liệu kiểm thử

	Accuracy	F1-score
Logistic regression	79.4%	75.3%
Decision tree	81.8%	80.0%
Support vector machine	81.3%	78.6%
Random forest	82.4%	80.8%
XGBoost	83.3%	81.0%
Artificial neural network (ANN)	82.0%	79.4%

Table 27. Bảng kết quả Confusion matrix (tạm dịch: ma trận nhầm lẫn) trên dữ liệu kiểm thử



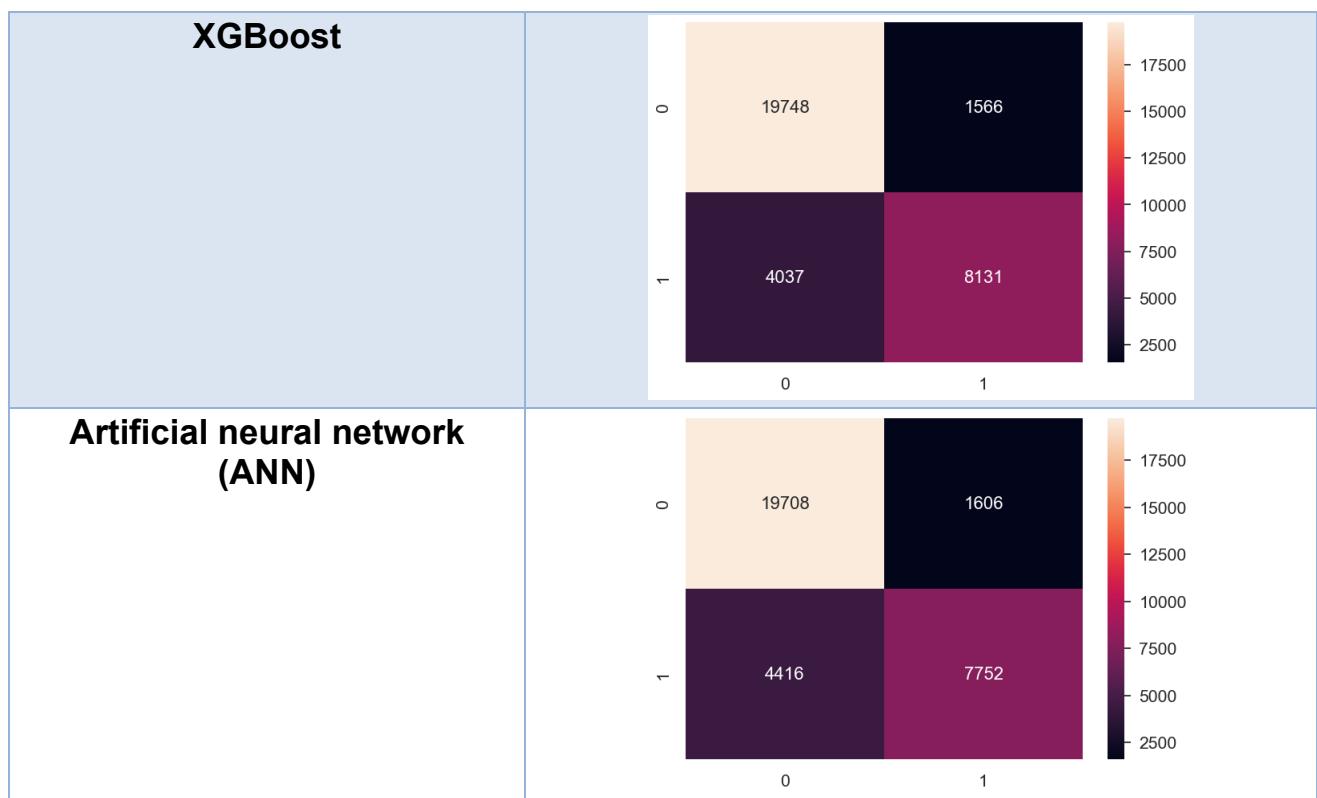


Table 28. Bảng kết quả Classification Report trên tập dữ liệu kiểm thử

Logistic regression	Classification report				
	Precision, recall, f1-score, support				
	0	0.78	0.94	0.85	21314
	1	0.84	0.53	0.65	12168
	accuracy			0.79	33482
	macro avg	0.81	0.74	0.75	33482
	weighted avg	0.80	0.79	0.78	33482
	Confusion matrix				
	0	1	0	1	
	[0.78, 0.53]	[0.84, 0.65]	[0.94, 0.53]	[0.85, 0.65]	[21314, 12168]
Decision tree	Classification report				
Decision tree	Precision, recall, f1-score, support				
Decision tree	0	0.84	0.88	0.86	21314
Decision tree	1	0.77	0.71	0.74	12168
Decision tree	accuracy			0.82	33482
Decision tree	macro avg	0.81	0.80	0.80	33482
Decision tree	weighted avg	0.82	0.82	0.82	33482

Support vector machine					
			precision	recall	f1-score
	0	0.81	0.92	0.86	21314
	1	0.81	0.63	0.71	12168
			accuracy		0.81
	macro avg		0.81	0.77	0.79
	weighted avg		0.81	0.81	0.81
Random forest					
			precision	recall	f1-score
	0	0.85	0.88	0.86	21314
	1	0.78	0.73	0.75	12168
			accuracy		0.82
	macro avg		0.81	0.80	0.81
	weighted avg		0.82	0.82	0.82
XGBoost					
			precision	recall	f1-score
	0	0.83	0.93	0.88	21314
	1	0.84	0.67	0.74	12168
			accuracy		0.83
	macro avg		0.83	0.80	0.81
	weighted avg		0.83	0.83	0.83
Artificial neural network (ANN)					
			precision	recall	f1-score
	0	0.82	0.92	0.87	21314
	1	0.83	0.64	0.72	12168
			accuracy		0.82
	macro avg		0.82	0.78	0.79
	weighted avg		0.82	0.82	0.81

Phân tích kết quả huấn luyện mô hình:

- Kết quả từ bảng 4.7.1 cho ta thấy tỷ lệ chính xác của các thuật toán khi mining trường hợp dự đoán khả năng hủy đặt phòng của khách hàng trên bộ dữ liệu kiểm thử:
 - Mô hình XGBoost đạt kết quả dự đoán tốt nhất trên bộ dữ liệu với độ đo Accuracy là 83,3% và F1-score là 81%. Các mô hình còn lại cũng cho kết quả khá tốt với độ chính xác trên 79%.
 - Các mô hình học máy dạng học kết hợp (ensemble learning) (*Random forest, XGBoost*) cho kết quả tốt hơn so với các mô hình đơn giản (logistic,

decision tree, SVM). Đối với kiến trúc ANN, kết quả không quá khác biệt với mô hình các mô hình khác, hiệu quả hơn mô hình cơ bản và kém hơn mô hình học kết hợp.

- Dựa vào kết quả F1 score, 6 mô hình đều cho thấy được việc huấn luyện trên bộ dữ liệu này cho dự đoán khá ổn định (không có dấu hiệu bất thường).
- Kết quả từ bảng bảng 4.7.2 cho ta thấy mức độ nhầm lẫn của các thuật toán khi dự đoán khả năng hủy đặt phòng của khách hàng trên bộ dữ liệu kiểm thử:
 - Với sự mất cân bằng của dữ liệu thì mô hình logistic regression bị ảnh hưởng nhiều nhất khi dự đoán nhầm cao nhất trong 6 mô hình (nhầm lẫn 5966 dự đoán).
 - Cả 6 mô hình đều có thiên hướng thiên vị nhãn không hủy phòng bởi vì sự mất cân bằng dữ liệu trong bộ dữ liệu huấn luyện mô hình.
 - Logistic regression cho dự đoán tốt nhất nhãn không hủy phòng với hơn 20000 dự đoán đúng, Random forest cho kết quả dự đoán tốt nhất với nhãn hủy phòng với 8840 dự đoán đúng.
- Kết quả từ bảng bảng 4.7.3 cho ta thấy độ đo precision và recall của từng nhãn ứng với từng mô hình của với các thuật toán khi dự đoán khả năng hủy đặt phòng của khách hàng trên bộ dữ liệu kiểm thử:
 - Logistic regression cho tỉ lệ nhận nhầm cao nhất ở nhãn khách hàng hủy phòng (với recall 0.53).
 - Tỉ lệ recall ở nhãn dự đoán hủy phòng (nhãn 1) của 6 mô hình thấp hơn khá nhiều so với các chỉ số khác. Điều này có thể giải thích là do sự mất cân bằng giữa nhãn hủy phòng và không hủy phòng của bộ dữ liệu

Kết luận:

- Mô hình Random forest là mô hình mạng lại hiệu quả ổn định nhất mặc dù các chỉ số chung (F1 score và Accuracy) không cao bằng các mô hình khác.

Ứng dụng của mô hình dự đoán đặt phòng:

- Với mô hình có hiệu suất tốt nhất trong thực nghiệm này là Random forest với độ chính xác 82,4%. Tức là mô hình có thể dự đoán chính xác khoảng 82% khách hàng có thể sẽ hủy phòng hay không. Từ đó ta có thể sử dụng

để dự đoán các khách hàng có nguy cơ hủy phòng, Để từ đó, khách sạn có thể đưa ra các quyết định hoặc phương pháp để có thể đạt được lợi nhuận cao nhất.

4.8. Phân tích các yếu tố ảnh hưởng đến việc hủy đặt phòng và rút ra luật

Đề tài sử dụng MDI của mô hình Random forest vừa được huấn luyện trên bộ dữ liệu để xem xét mức độ ảnh hưởng của các yếu tố đến việc hủy đặt phòng của khách hàng.

Mã lệnh trực quan hóa mức độ ảnh hưởng của các thuộc tính:

```
importances = model_rf.feature_importances_
std = np.std([tree.feature_importances_ for tree in model_rf.estimators_], axis=0)

feature_names = ['hotel', 'lead_time', 'market_segment', 'distribution_channel',
                 'is_repeated_guest', 'previous_cancellations',
                 'previous_bookings_not_canceled', 'reserved_room_type',
                 'assigned_room_type', 'booking_changes', 'deposit_type',
                 'days_in_waiting_list', 'customer_type', 'required_car_parking_spaces',
                 'total_of_special_requests']

forest_importances = pd.Series(importances, index=feature_names)

fig, ax = plt.subplots(figsize=(7,7))
forest_importances.plot.bar(yerr=std, ax=ax)
ax.set_title("Feature importances using MDI")
ax.set_ylabel("Mean decrease in impurity")
fig.tight_layout()
```

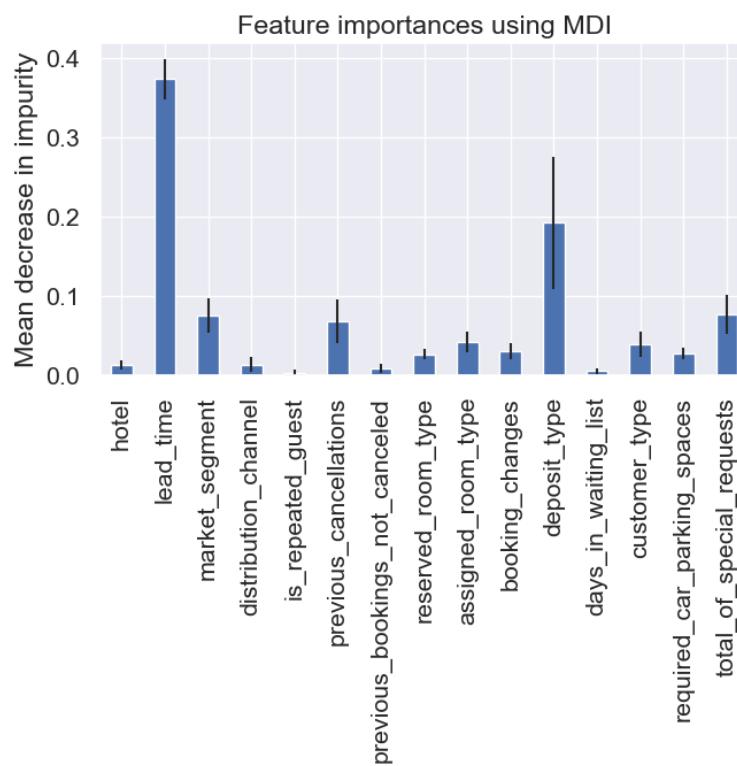


Figure 343. Biểu đồ mức độ các yếu tố ảnh hưởng đến việc hủy phòng của khách hàng

Rút ra luật:

- Thời gian chờ và loại thanh toán ảnh hưởng cao đến việc hủy đặt phòng.

Vẽ biểu đồ thể hiện sự ảnh hưởng của 2 yếu tố đến việc đặt phòng (0: Không hủy phòng, 1: Hủy phòng)

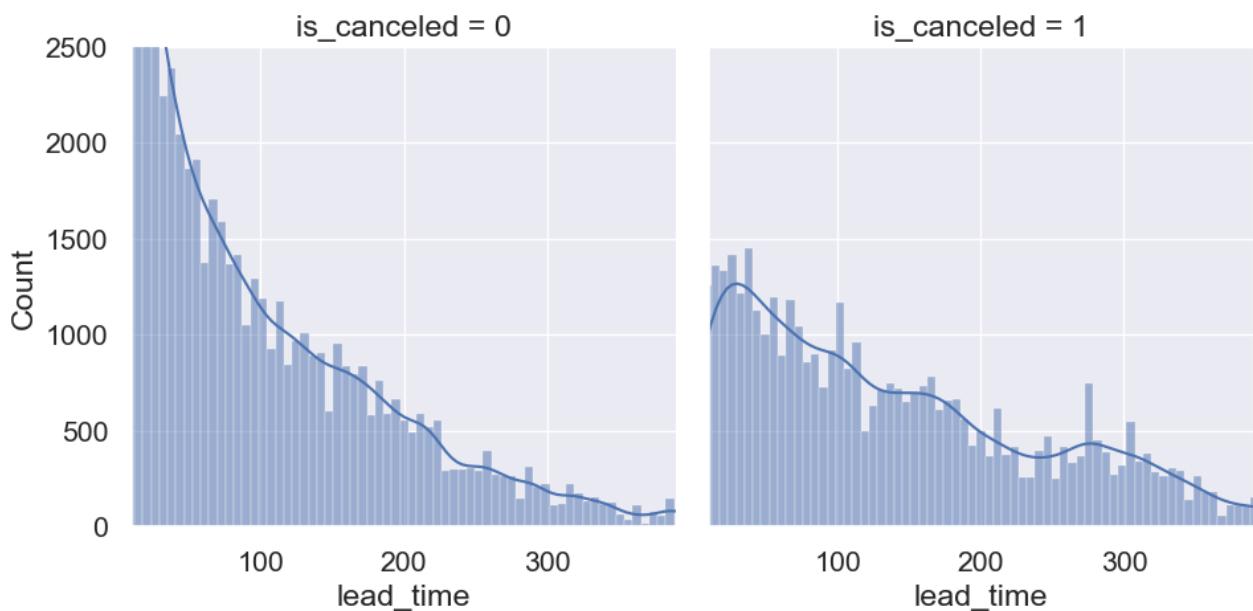


Figure 344. Biểu đồ phân phối giữa thời gian chờ và sự hủy phòng

Nhận xét:

- Từ 100 - 200: thời gian chờ càng ít thì ít hủy phòng
- Từ 250 - 350: thời gian chờ lâu hơn thì dễ hủy phòng

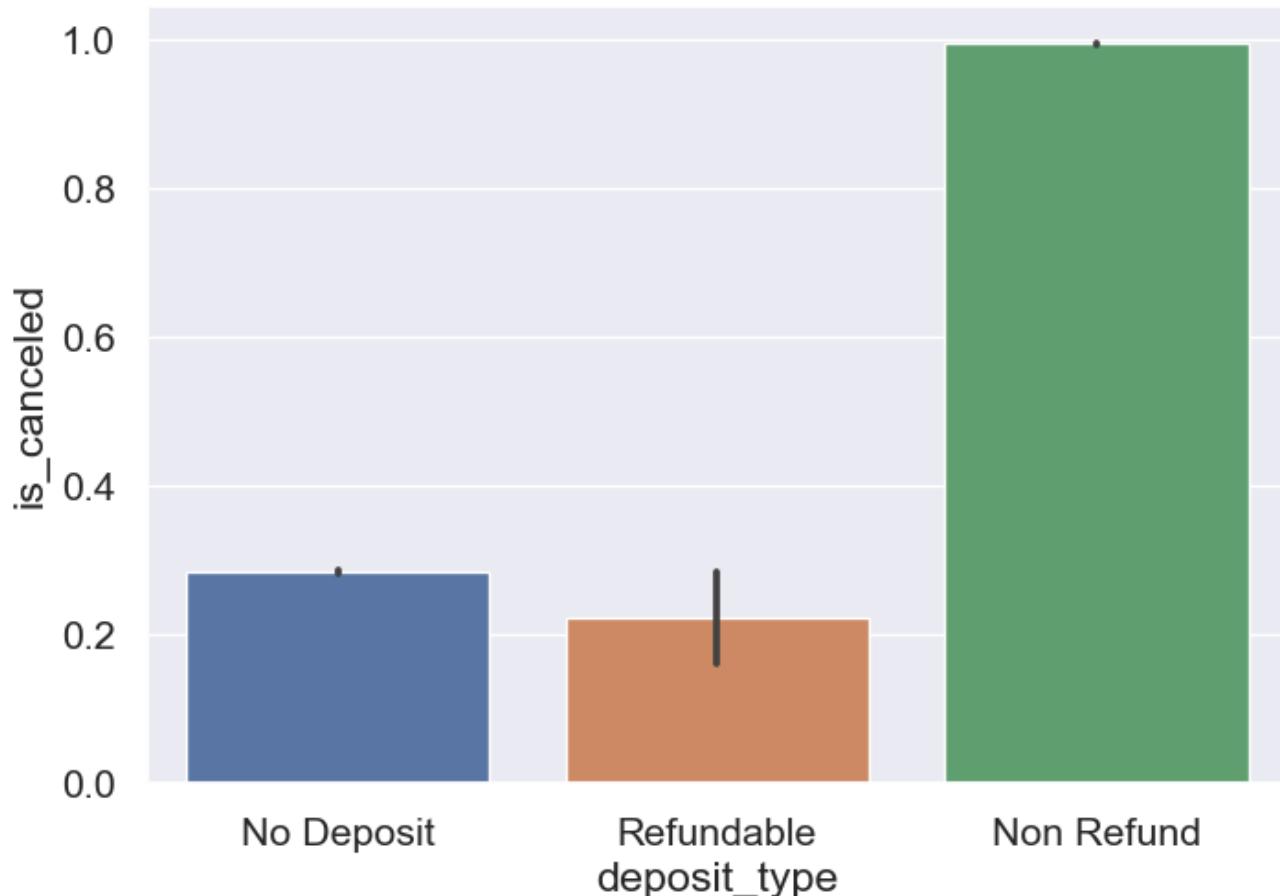


Figure 345. Biểu đồ giữa loại thanh toán và sự hủy phòng

Nhận xét:

- Không có tiền cọc trước, khách hàng dễ hủy phòng.

4.9. Gom nhóm khách hàng và đặc trưng của từng nhóm

4.9.1. Lựa chọn số nhóm khách hàng cần gom nhóm

Để lựa chọn số cụm phù hợp cho nhóm khách hàng, đề tài sử dụng Inertia (trục x là số nhóm, trục y là lỗi). Số cụm mà từ đó về sau lỗi có xu hướng giảm chậm thì sẽ là tốt nhất cho việc phân chia.

Mã lệnh trực quan hóa độ lỗi theo nhóm được chia:

```
from sklearn.cluster import KMeans
distance = []
K = range (1,15)
for k in K :
    k_mean = KMeans(n_clusters=k)
    k_mean.fit(df_cluster)
    distance.append(k_mean.inertia_)
plt.plot(K,distance,marker='o');
```

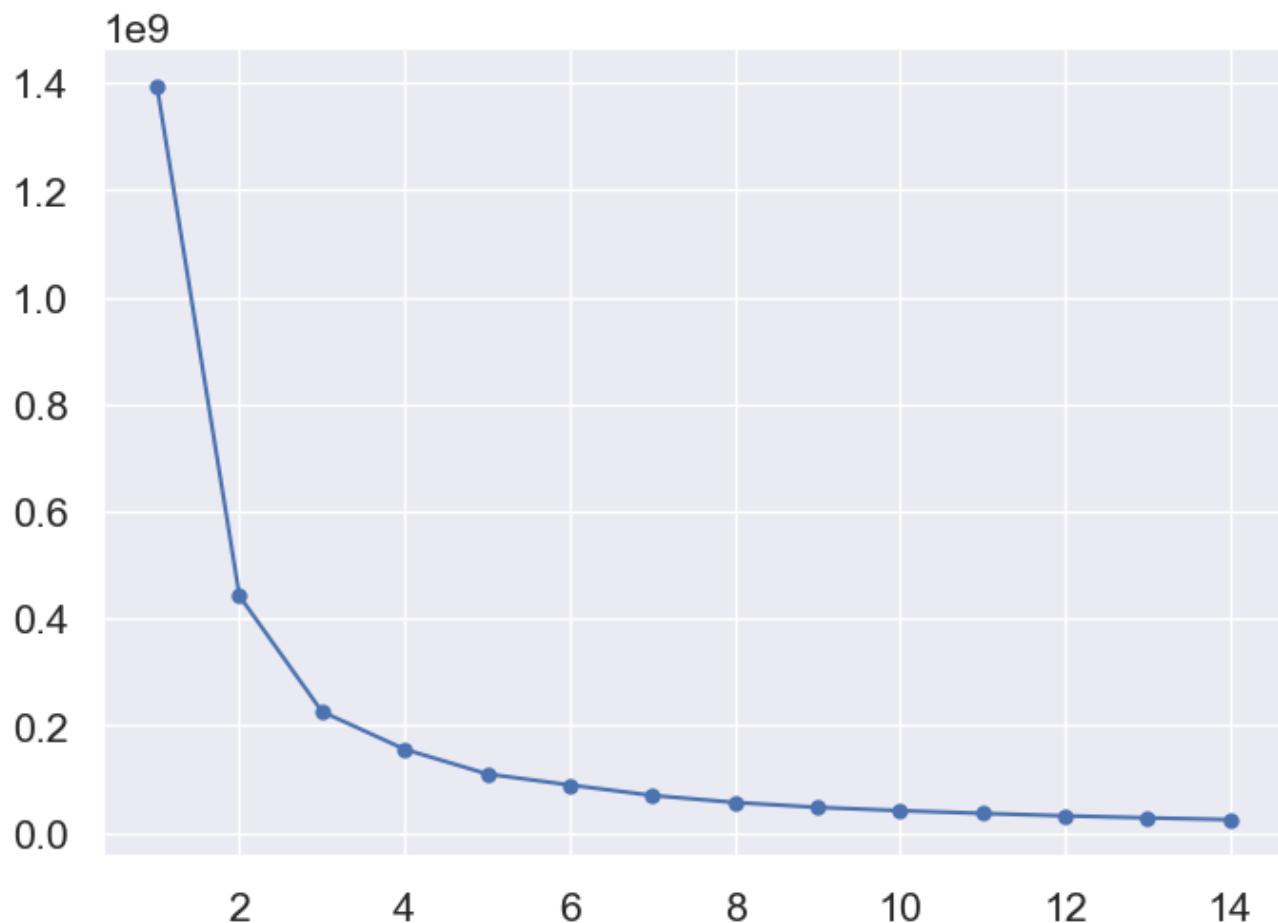


Figure 346. Biểu đồ lối ứng với các nhóm khách hàng được phân chia

Nhận xét:

- Từ 3 nhóm trở đi, lối ứng có xu hướng giảm chậm nên đề tài chọn chia làm 3 nhóm khách hàng là phù hợp.

4.9.2. Gom nhóm khách hàng bằng thuật toán K-means và trích xuất đặc trưng của từng nhóm khách hàng

Mã lệnh gom nhóm khách hàng:

```
k_mean_cus = KMeans(n_clusters=3)
k_mean_CustomerClus = k_mean_cus.fit(df_cluster)
k_mean_CustomerClus.labels_
```

Mã lệnh trích xuất đặc trưng của từng nhóm khách hàng:

```

center = [k_mean_CustomerClus.cluster_centers_[:, 0].round(0), k_mean_CustomerClus.cluster_centers_[:, 1].round(0),
          k_mean_CustomerClus.cluster_centers_[:, 2].round(0), k_mean_CustomerClus.cluster_centers_[:, 3].round(0),
          k_mean_CustomerClus.cluster_centers_[:, 4].round(0), k_mean_CustomerClus.cluster_centers_[:, 5].round(0),
          k_mean_CustomerClus.cluster_centers_[:, 6].round(0), k_mean_CustomerClus.cluster_centers_[:, 7].round(0),
          k_mean_CustomerClus.cluster_centers_[:, 8].round(0), k_mean_CustomerClus.cluster_centers_[:, 9].round(0),
          k_mean_CustomerClus.cluster_centers_[:, 10].round(0), k_mean_CustomerClus.cluster_centers_[:, 11].round(0),
          k_mean_CustomerClus.cluster_centers_[:, 12].round(0), k_mean_CustomerClus.cluster_centers_[:, 13].round(0),
          k_mean_CustomerClus.cluster_centers_[:, 14].round(0), k_mean_CustomerClus.cluster_centers_[:, 15].round(0)]
center = np.array(center)
center = center.astype(int)

hotel_type_idx2value = {0:'City Hotel', 1:'Resort Hotel'}
mar_type_idx2value = {0: 'Online TA', 1: 'Offline TA/TO', 2: 'Groups', 3:'Direct', 4:'Corporate', 5:'Complementary', 6:'Aviation'}
dis_type_idx2value = {0: 'TA/TO', 1: 'Direct', 2: 'Corporate', 3: 'GDS'}
des_type_idx2value = {0: 'No Deposit', 1: 'Non Refund', 2: 'Refundable'}
cus_type_idx2value = {0: 'Transient', 1: 'Transient-Party', 2: 'Contract', 3: 'Group'}
res_type_idx2value = {0: 'Check-Out', 1: 'Canceled', 2: 'No-Show'}

columns = df_cluster.columns
index = ['Cluster 1','Cluster 2','Cluster 3']

custom_features = pd.DataFrame(center.T,columns=columns,index=index)

custom_features['hotel'] = custom_features['hotel'].map(hotel_type_idx2value)
custom_features['market_segment'] = custom_features['market_segment'].map(mar_type_idx2value)
custom_features['distribution_channel'] = custom_features['distribution_channel'].map(dis_type_idx2value)
custom_features['deposit_type'] = custom_features['deposit_type'].map(des_type_idx2value)
custom_features['customer_type'] = custom_features['customer_type'].map(cus_type_idx2value)

```

Table 29. Bảng đặc trưng của từng nhóm khách hàng

Thuộc tính	Nhóm 1	Nhóm 2	Nhóm 3
is_canceled	Không hủy phòng	Không hủy phòng	Hủy phòng
hotel	City Hotel	City Hotel	City Hotel
lead_time	152	31	323
market_segment	Offline TA/TO	Offline TA/TO	Offline TA/TO
distribution_channel	TA/TO	TA/TO	TA/TO
is_repeated_guest	0	0	0
previous_cancellations	0	0	0
previous_bookings_not_canceled	0	0	0
reserved_room_type	2	2	1
assigned_room_type	2	3	2
booking_changes	0	0	0
deposit_type	No Deposit	No Deposit	No Deposit
days_in_waiting_list	3	0	9
customer_type	Transient	Transient	Transient-Party
required_car_parking_spaces	0	0	0
total_of_special_requests	1	1	0

Nhận xét về nhóm có xu hướng dễ hủy phòng:

- Nhóm 3 là nhóm có xu hướng dễ hủy phòng
- Đặc trưng của nhóm 3:
 - o Thời gian chờ từ ngày khách hàng đặt chỗ đến ngày khách hàng đến nhận phòng cao hơn rất nhiều 2 nhóm còn lại (323 so với 152 và 31)
 - o Thời gian chờ để được khách sạn xác nhận việc đặt phòng rất lâu (9 so với 3 và 0)
 - o Là loại khách hàng tạm trú theo nhóm và không có ý định ở lâu dài (Transient-Party).

Mã lệnh trực quan hóa 3 nhóm khách hàng theo 3 yếu tố phân biệt:

```
result = k_mean_CustomerClus.labels_
df_cluster_np = np.array(df_cluster)
fig = plt.figure(figsize=(7,7))
ax = mplot3d.Axes3D(fig, auto_add_to_figure=False)
fig.add_axes(ax)

ax.scatter3D(
    df_cluster_np[result == 0,12], df_cluster_np[result == 0,2], df_cluster_np[result == 0,13],
    c='orangered', s=25,
    edgecolor='black',
    label='Cluster 1')
ax.scatter3D(
    df_cluster_np[result == 1,12], df_cluster_np[result == 1,2], df_cluster_np[result == 1,13],
    c='royalblue', s=25,
    edgecolor='black',
    label='Cluster 2')
ax.scatter3D(
    df_cluster_np[result == 2,12], df_cluster_np[result == 2,2], df_cluster_np[result == 2,13],
    c='lightgreen', s=25,
    edgecolor='black',
    label='Cluster 3')

ax.scatter3D(
    k_mean_CustomerClus.cluster_centers_[:, 12], k_mean_CustomerClus.cluster_centers_[:, 2],
    k_mean_CustomerClus.cluster_centers_[:, 13],
    s=300, marker='*',
    c='red', edgecolor='black',
    label='Centroids')

ax.set_xlabel('Days_in_waiting_list', fontsize = 10)
ax.set_ylabel('Lead_time', fontsize = 10)
ax.set_zlabel('Customer_type', fontsize = 10)
ax.set_xlim(0, 250)
ax.set_ylim(0, 600)

plt.legend(scatterpoints=1, loc = 'upper left', title='Cluster Groups');
```

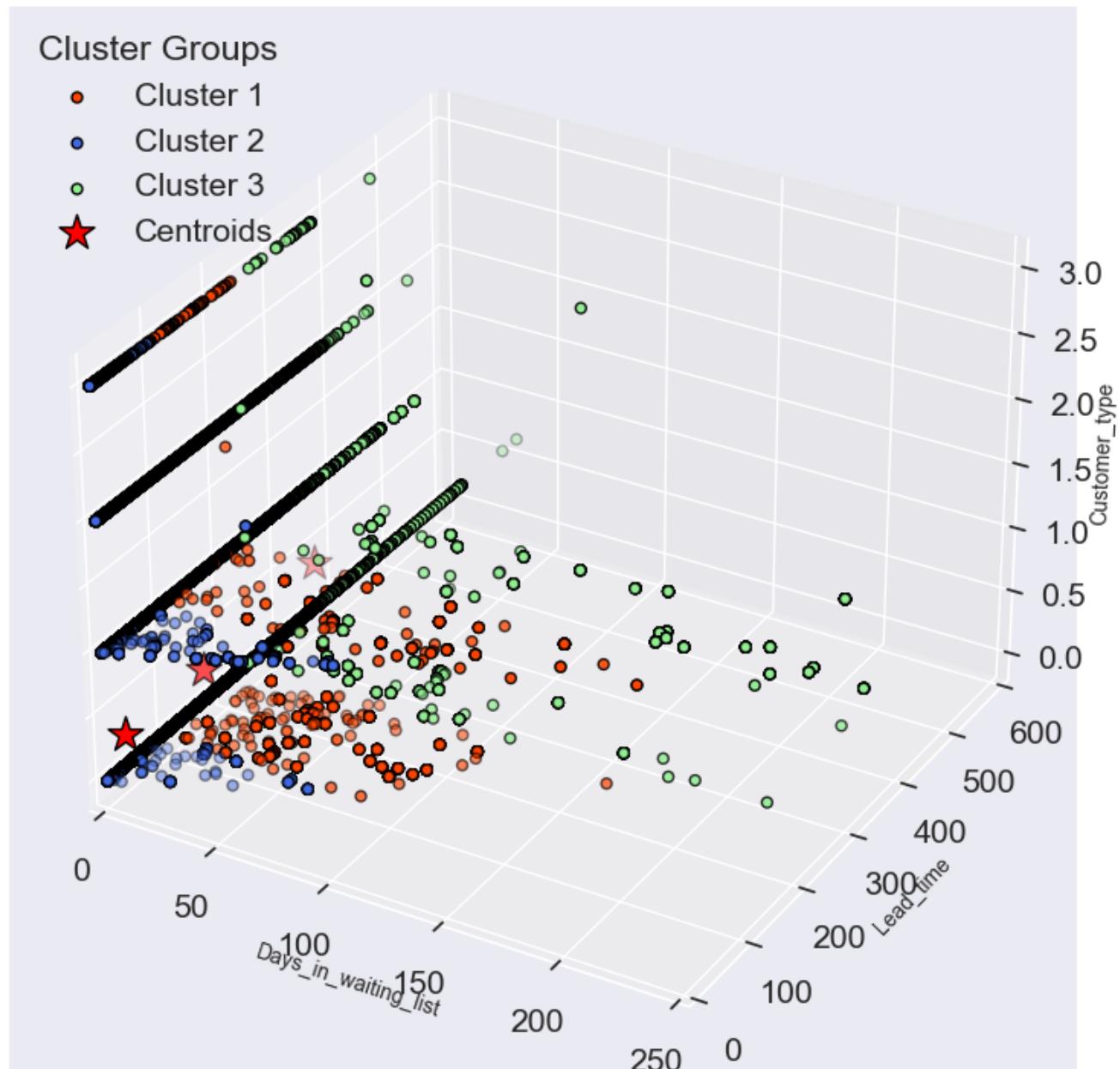


Figure 347. Biểu đồ trực quan 3 nhóm khách hàng theo 3 yếu tố phân biệt

DANH MỤC TÀI LIỆU THAM KHẢO

- [1] N. Antonio, A. Almeida and L. Nunes, “Kaggle,” [Online]. Available: <https://www.kaggle.com/datasets/mojtaba142/hotel-booking>. [Accessed 1 April 2023].
- [2] N. Antonio, A. d. Almeida and L. Nunes, “Hotel booking demand datasets,” *Data in Brief*, vol. 22, pp. 41-49, 2019.
- [3] Admin, “ralphkimball,” 14 August 2020. [Online]. Available: <https://ralphkimball.com/dimensional-modeling/>. [Accessed 1 April 2023].
- [4] N.T.K.Phụng and H.T.Thành, Slide Bài 4 - Cơ sở dữ liệu đa chiều và các mô hình biểu diễn.

PHỤ LỤC

Phụ lục 1: Bộ mã lệnh thống kê dữ liệu

```
#Import thư viện
import pandas as pd
import seaborn as sns
import numpy as np
from matplotlib import pyplot as plt
from matplotlib import style
from datetime import datetime

#Đọc dữ liệu
df = pd.read_csv('hotel_booking.csv')
#Kiểm tra 5 dòng dữ liệu đầu tiên
df.head(5)

#Thống kê số lượng nhận phòng và hủy đặt phòng
fig, ax = plt.subplots(figsize=(8, 6))
visual_is_canceled = df['is_canceled'].value_counts()
ax.bar(visual_is_canceled.index, visual_is_canceled)
ax.set_xticks([0,1])
ax.set_xticklabels(['not canceled', 'canceled'])
ax.set_title('Biểu đồ thể hiện số lượng phòng đặt và hủy phòng', fontsize=20)

#Xem xét mối quan hệ giữa dữ liệu bị hủy đặt phòng tại hai loại hình khách sạn trong khoảng thời gian từ
khi đặt phòng cho tới khi nhận được phòng
canceled = 'canceled'
not_canceled ='not canceled'
fig, axes = plt.subplots(nrows = 1,ncols =2,figsize = (10,4))
city = df[df['hotel']=='City Hotel']
resort = df[df['hotel']=='Resort Hotel']
ax = sns.distplot(city[city['is_canceled']==1].lead_time.dropna(), bins = 18,label=canceled, ax=axes[0],kde=False)
ax = sns.distplot(city[city['is_canceled']==0].lead_time.dropna(), bins = 40,label=not_canceled, ax=axes[0],kde=False)
ax.legend()
ax.set_title('City Hotel')
ax = sns.distplot(resort[resort['is_canceled']==1].lead_time.dropna(), bins=18,
label=canceled, ax=axes[1],kde=False)
ax = sns.distplot(resort[resort['is_canceled']==0].lead_time.dropna(), bins=40,
label=not_canceled, ax=axes[1],kde=False)
ax.legend()
ax.set_title('Resort Hotel')

#Xem xét sự tương quan giữa việc thuê phòng vào cuối tuần với hủy đặt phòng tại khách sạn
total = df.groupby('stays_in_weekend_nights')['is_canceled'].count()[:5]
cancels = df.groupby('stays_in_weekend_nights')['is_canceled'].sum()[:5]
cutoff = 3
```

```
x = list(total.index[:cutoff])
x.append(cutoff)
y_total = list(total.values[:cutoff])
y_total.append(np.sum(total.values[cutoff:]))
y_cancel = list(cancels.values[:cutoff])
y_cancel.append(np.sum(cancels.values[cutoff:]))
fig, ax = plt.subplots(figsize=(8, 6))
ax.bar(x, y_total, label='total reservations')
ax.bar(x, y_cancel, label='canceled reservations')
ax.legend()
ax.set_xticks([0, 1, 2, 3])
ax.set_xticklabels(['0', '1', '2', '3+'])
ax.set_xlabel('# of weekend nights')

#Thống kê số giá trị bị khuyết ở từng thuộc tính
missing_data = df.isnull().sum()
print(missing_data)
```

Phụ lục 2: Bộ mã lệnh tiền xử lý dữ liệu

```
#Xóa thuộc tính agent và comany khỏi dataframe
# Xóa cột agent
df = df.drop(["agent"], axis = 1)
# Xóa cột company
df = df.drop(["company"], axis = 1)

#Xóa thuộc tính credit_card
df = df.drop(["credit_card"], axis = 1)

#Xóa dòng dữ liệu mà thuộc tính children có giá trị null
df.dropna(axis=0,inplace=True,subset=['children'])

# Chọn giá trị "PRT" để điền vào các vị trí null
common_value='PRT'
for dataset in [df]:
    dataset['country'] = dataset['country'].fillna(common_value)

# Xóa các dòng dữ liệu có giá trị Undefined df.replace("Undefined", np.nan, inplace=True)

#Chuyển đổi thuộc tính "meal" từ chữ sang số
meals = {"BB":1 , "FB":2, "HB":3, "SC":4}
for dataset in [df]:
    dataset['meal'] = dataset['meal'].map(meals)

#Tokenize thuộc tính "reserved_room_type" và "assigned_room_type" từ chữ sang số

rooms = {"A":1, "B":2, "C":3, "D":4, "E":5, "F":6, "G":7, "H":8, "I":9, "K":10, "L":11, "P":12}
for dataset in [df]:
    dataset['reserved_room_type'] = dataset['reserved_room_type'].map(rooms)
```

```
dataset['assigned_room_type'] = dataset['assigned_room_type'].map(rooms)

#Chuyển đổi thuộc tính tháng (arrival_date_month) từ chữ sang số
for dataset in [df]:
    dataset.loc[dataset['arrival_date_month_number']=="January",'arrival_date_month_number']=1
    dataset.loc[dataset['arrival_date_month_number']=="February",'arrival_date_month_number']=2
    dataset.loc[dataset['arrival_date_month_number']=="March",'arrival_date_month_number']=3
    dataset.loc[dataset['arrival_date_month_number']=="April",'arrival_date_month_number']=4
    dataset.loc[dataset['arrival_date_month_number']=="May",'arrival_date_month_number']=5
    dataset.loc[dataset['arrival_date_month_number']=="June",'arrival_date_month_number']=6
    dataset.loc[dataset['arrival_date_month_number']=="July",'arrival_date_month_number']=7
    dataset.loc[dataset['arrival_date_month_number']=="August",'arrival_date_month_number']=8
    dataset.loc[dataset['arrival_date_month_number']=="September",'arrival_date_month_number']=9
    dataset.loc[dataset['arrival_date_month_number']=="October",'arrival_date_month_number']=10
    dataset.loc[dataset['arrival_date_month_number']=="November",'arrival_date_month_number']=11
    dataset.loc[dataset['arrival_date_month_number']=="December",'arrival_date_month_number']=12
    dataset['arrival_date_month_number']=dataset['arrival_date_month_number'].astype(int)

#Sửa tên thuộc tính arrival_date_month thành arrival_date_month_name
df.rename(index=str, columns={"arrival_date_month": "arrival_date_month_name"}, inplace=True)

#Thêm thuộc tính id_reservation_status_date theo định dạng YYYYMMDD của reservation_status_date
df['reservation_status_date'] = pd.to_datetime(df['reservation_status_date'])
df["id_reservation_status_date"] = df["reservation_status_date"].dt.strftime("%Y%m%d")

# Tạo thuộc tính arrival_date_full dựa vào các thuộc tính khác như arrival_date_year (YYYY),
# arrival_date_month_number (MM) và arrival_date_day_of_month (DD)
cols=["arrival_date_year","arrival_date_month_number","arrival_date_day_of_month"]
df['arrival_date_full']=df[cols].apply(lambda x: '-'.join(x.values.astype(str)), axis="columns")
df['arrival_date_full']=pd.to_datetime(df['arrival_date_full'])

# Tạo thuộc tính chi tiết cho arrival_date_full
df["id_arrival_date"] = df["arrival_date_full"].dt.strftime("%Y%m%d")
df["arrival_date_quarter"] = df["arrival_date_full"].dt.quarter
df["arrival_date_day_of_week"] = df["arrival_date_full"].dt.dayofweek
df["arrival_date_day_name"] = df["arrival_date_full"].dt.day_name()
df['arrival_date_day_name_abbrev'] = df['arrival_date_full'].dt.day_name().str[:3]
df['arrival_date_month_name_abbrev'] = df['arrival_date_full'].dt.month_name().str[:3]
df["arrival_date_weekday_flag"] = df["arrival_date_day_of_week"]


```

Phụ lục 3: Bộ mã lệnh SQL tạo khóa cho bảng

```
alter table Fact
add constraint fk_F_HT
foreign key (id_hotel) references [dbo].[Dim-Hotel-Type] (id_hotel_type);
alter table Fact
add constraint fk_F_RS
foreign key (id_reservation_status) references [dbo] [Dim-Reservation-Status] (id_reservation_status);
alter table Fact
add constraint fk_F_MK
foreign key (id_market_segment) references [dbo].[Dim-Market-Segment] (id_market_segment);
alter table Fact
add constraint fk_F_DC
```

IS217.N21 – Kho dữ liệu và phân tích hoạt động đặt phòng khách sạn

```
foreign key (id_distribution_channel) references [dbo].[Dim-Distribution-Channel] (id_distribution_channel);
alter table Fact
add constraint fk_F_DT
foreign key (id_deposit_type) references [dbo].[Dim-Deposit-Type] (id_deposit_type);
alter table Fact
add constraint fk_F_C
foreign key (id_customer) references [dbo].[Dim-Customer] (id_customer);
alter table [dbo].[Dim-Customer]
add constraint fk_C_CT
foreign key (id_customer_type) references [dbo].[Dim-Customer-Type] (id_customer_type);
alter table [dbo].[Dim-Customer]
add constraint fk_C_C
foreign key (id_country) references [dbo].[Dim-Country] (id_country);
alter table Fact
add constraint fk_F_AT
foreign key (id_arrival_time) references [dbo].[Dim-Arrival-Time] (id_arrival_time);
alter table Fact
add constraint fk_F_RT
foreign key (id_reservation_date) references [dbo].[Dim-Reservation-Time] (id_reservation_time);
alter table [dbo].[Dim-Arrival-Time]
add constraint fk_AT_Y
foreign key (id_arrival_year) references [dbo].[Dim-Year] (id_year);
alter table [dbo].[Dim-Reservation-Time]
add constraint fk_RT_Y
foreign key (id_reservation_year) references [dbo].[Dim-Year] (id_year);
alter table [dbo].[Dim-Arrival-Time]
add constraint fk_AT_M
foreign key (id_arrival_month_number) references [dbo].[Dim-Month] (id_month_number);
alter table [dbo].[Dim-Reservation-Time]
add constraint fk_RT_M
foreign key (id_reservation_month_number) references [dbo].[Dim-Month] (id_month_number);
alter table [dbo].[Dim-Arrival-Time]
add constraint fk_AT_Q
foreign key (id_arrival_quarter) references [dbo].[Dim-Quarter] (id_quarter);
alter table [dbo].[Dim-Reservation-Time]
add constraint fk_RT_Q
foreign key (id_reservation_quarter) references [dbo].[Dim-Quarter] (id_quarter);
alter table [dbo].[Dim-Arrival-Time]
add constraint fk_AT_D
foreign key (id_arrival_day) references [dbo].[Dim-Day] (id_day);
alter table [dbo].[Dim-Reservation-Time]
add constraint fk_RT_D
foreign key (id_reservation_day) references [dbo].[Dim-Day] (id_day);
```