HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

KHOA CÔNG NGHỆ THÔNG TIN I

BÁO CÁO BÀI TẬP LỚN

CƠ SỞ DỮ LIỆU PHÂN TÁN

ĐỀ TÀI : "HỆ THỐNG ĐĂNG KÍ HỌC PHẦN NHIỀU CƠ SỞ"

Giảng viên hướng dẫn : Kim Ngọc Bách

Nhóm lớp : 10

Nhóm bài tập lớn : 07

Danh sách sinh viên :

Kiều Trường Giang : B23DCCN254

Lê Bùi Quốc Huy : B23DCCN387

Giang Thủy Tiên : B23DCCN814

Hà Nội – 2026

LỜI CẢM ƠN

Em xin gửi lời cảm ơn chân thành đến thầy Kim Ngọc Bách, giảng viên học phần Cơ sở dữ liệu phân tán, đã tận tình giảng dạy và truyền đạt cho chúng em những kiến thức quan trọng trong suốt quá trình học tập.

Thông qua quá trình thực hiện đề tài "Hệ thống đăng ký học phần nhiều cơ sở", em đã có cơ hội củng cố kiến thức về cơ sở dữ liệu phân tán, hiểu rõ hơn về phân mảnh dữ liệu, quản lý dữ liệu tại nhiều cơ sở và đảm bảo tính nhất quán trong hệ thống.

Mặc dù đề tài vẫn còn một số hạn chế do kiến thức và kinh nghiệm thực tế chưa nhiều, nhưng quá trình thực hiện đã giúp em rèn luyện tư duy phân tích, thiết kế hệ thống và nâng cao khả năng vận dụng lý thuyết vào bài toán thực tế.

Em xin chân thành cảm ơn thầy và kính chúc thầy luôn mạnh khỏe, công tác tốt và đạt được nhiều thành công trong sự nghiệp giảng dạy.

MỤC LỤC

LỜI CẢM ƠN **1**
MỤC LỤC **2**
PHẦN 1: ĐẶT VẤN ĐỀ 6
I. Giới thiệu 6
1. Nhu cầu và tầm quan trọng của dự án 6
2. Sơ lược về dự án 7
II. Vị trí và nhiệm vụ, dữ liệu khi triển khai dự án 8
1. Các vị trí 8
2. Các đối tượng dữ liệu triển khai 8
3. Các công việc dữ liệu triển khai 9
III. Các đối tượng tham gia sử dụng dự án 10
1. Sinh viên: 10
2. Giảng viên: 11
3. Nhân viên quản lý (tại các cơ sở chi nhánh): 11
4. Quản trị viên hệ thống (tại máy chủ trung tâm - Hà Đông): 11
PHẦN 2: PHÂN TÍCH 13
I. Các chức năng chính của hệ thống trong dự án 13
II. Phân quyền cho các nhóm đối tượng thực hiện dự án 13
III. Phân tích chức năng của từng vị trí thực hiện dự án 14
IV. Chức năng của máy trạm, máy chủ 14
V. Phân tích cơ sở dữ liệu 15
PHẦN 3: ĐẶT VẤN ĐỀ 21
I. Thiết kế hệ thống mạng tổng quan 21
II. Thiết kế cơ sở dữ liệu của hệ thống 21
1. Cấu trúc các bảng dữ liệu của hệ thống 21
2. Quan hệ giữa các bảng 28
III. Thiết kế cơ sở dữ liệu phân tán 28
1. Thiết kế phân mảnh ngang 28
2. Thiết kế định vị và vẽ sơ đồ định vị 31
2.1. Thiết kế định vị 31
2.2. Sơ đồ định vị 32
3. Lược đồ ánh xạ 33
IV. Thiết kế lược đồ nhân bản, đồng bộ hóa 33
PHẦN 4: CÀI ĐẶT 35
I.Cài đặt SQL Server 2025 35
1.Cài đặt SQL Server 2025 Express 35
2.Cài đặt SQL Server Management Studio Tools (SSMS) 42
II. Cài đặt tường lửa (firewall) để mở cổng cho SQL 43
1. Tắt tường lửa 43

2. Mở port 1433 45
3. Thêm New Rule: csdlpt_nhom_7 49
4. Thêm New Rule: sqlserver 53
5. Thêm New Rule: port udp_1434 57
III. Chuẩn bị thư mục chứa dữ liệu đồng bộ (REPLDATA) 60
IV. Kết nối máy chủ và các chi nhánh bằng phần mềm Radmin VPN 64
1. Tải và cài đặt phần mềm Radmin VPN 64
2. Bật TCP/IP và thêm IP cấu hình cho các máy trạm 67
3. Tạo hoặc cấu hình tài khoản sa (supper admin) 72
V. Tạo Database DkyTinChi 75
1. Bảng headquarter 76
2. Bảng department 77
3. Bảng subject 77
4. Bảng teacher 78
5. Bảng curriculum 78
6. Bảng student 79
7. Bảng term 79
8. Bảng class 80
9. Bảng registration 80
10. Bảng room 81
11. Bảng timeslot 81
12. Bảng session 82
13. Bảng prerequisite 82
14. Bảng curriculum_subject 83
VI. Configure distribution 83
1. Đăng nhập và bật SQL Server Agent 84
2.Confiruge distribution 85
VII. Tạo Publication Database 90
1. Tạo Transactional Replication 90
2.Tạo Merge replication 97
VIII. Tạo Subscriptions 110
1. Thông tin máy trạm 110
2. Tạo Subscriptions 110
IX. Cấu hình đồng bộ dữ liệu giữa máy chủ và máy trạm 124
1. Đồng bộ dữ liệu từ máy chủ xuống máy trạm 124
2. Đồng bộ dữ liệu từ máy trạm lên máy chủ 125
X. Tạo Link Server 127
X. Trigger 132
1. Trigger kiểm tra đăng ký trùng học phần 132
2. Trigger kiểm tra trùng lịch học 134

3. Trigger kiểm tra sĩ số lớp học phần..... 135
4. Trigger kiểm tra sức chứa..... 137
X. Stored Procedures..... 138
1. Procedure kiểm tra điều kiện đăng ký học phần..... 138
2. Procedure đăng ký học phần..... 141
3. Procedure hủy đăng ký học phần..... 143
4. Procedure tra cứu kết quả đăng ký học phần..... 146
5. Procedure xem thời khóa biểu sinh viên..... 147
6. Procedure thống kê tình trạng đăng ký lớp học phần theo học kỳ..... 151
7. Procedure kiểm tra đăng ký chéo cơ sở..... 153
8. Procedure thống kê sinh viên đăng ký theo cơ sở..... 154
9. Procedure lấy danh sách lớp học phần theo cơ sở..... 157
10. Procedure đăng ký chéo cơ sở..... 159
PHẦN 5: TRUY VẤN PHÂN TÁN..... 162
I. Mục tiêu của truy vấn phân tán..... 162
II. Phương pháp thực hiện truy vấn phân tán..... 163
III. Các truy vấn phân tán..... 164
Truy vấn 1: Thống kê số sinh viên đăng ký học phần theo từng cơ sở..... 164
Truy vấn 2: Tìm học phần có nhiều sinh viên đăng ký nhất toàn trường..... 166
Truy vấn 3: Danh sách sinh viên đăng ký chéo cơ sở..... 167
Truy vấn 4: Tỷ lệ lấp đầy của các lớp học phần trên toàn hệ thống..... 168
Truy vấn 5: Thống kê số lớp học phần mở theo khoa của từng cơ sở..... 170
Truy vấn 6: Tra cứu danh sách lớp học phần còn chỗ..... 171
Truy vấn 7: Thống kê khối lượng giảng dạy của giảng viên..... 173
PHẦN 6: XỬ LÝ ĐĂNG KÝ HỌC PHẦN ĐỒNG THỜI..... 175
I. Mục tiêu xử lý đồng thời..... 175
1. Vượt quá sĩ số lớp học phần..... 176
2. Đăng ký trùng lớp học phần..... 176
3. Trùng lịch học..... 176
4. Lost Update..... 176
5. Dữ liệu không nhất quán giữa các site..... 176
IV. Cơ chế kiểm soát đồng thời..... 176
1. Sử dụng Transaction..... 176
2. Sử dụng XACT_ABORT..... 177
3. Sử dụng UPDLOCK..... 177
4. Sử dụng ROWLOCK..... 177
5. Rollback khi giao dịch thất bại..... 177
V. Phân tích Procedure đăng ký học phần..... 177
1. Luồng thực thi kiểm duyệt điều kiện (Validation Pipeline):..... 177
2. Cơ chế kiểm soát đồng thời và cập nhật sĩ số..... 178

3. Ghi nhận và Hoàn tất giao dịch:..... 178
VI. Phân tích Procedure hủy đăng ký học phần..... 179
VII. Demo xử lý đăng ký đồng thời..... 179
1. Môi trường và Mục tiêu kiểm thử.....179
2. Chuẩn bị lớp học phần có số chỗ giới hạn.....180
3. Thực hiện đăng ký từ hai phiên truy vấn..... 181
4. Kiểm tra kết quả sau khi đăng ký tại máy trạm TP. Hồ Chí Minh.....182
6. Kiểm tra dữ liệu đồng bộ về máy chủ trung tâm..... 183
VIII. Đánh giá cơ chế xử lý đồng thời.....184
PHẦN 7: DEMO CHƯƠNG TRÌNH.....185
1. Công nghệ sử dụng..... 185
2. Giao diện.....186
PHẦN 8: ĐÁNH GIÁ, NÂNG CAO VÀ TỔNG KẾT..... 192
I. Đối chiếu mức độ đáp ứng yêu cầu đề tài.....192
II. Đánh giá thiết kế cơ sở dữ liệu..... 193
III. Đánh giá thiết kế phân tán.....193
IV. Đánh giá cơ chế Replication và đồng bộ dữ liệu.....193
V. Đánh giá truy vấn phân tán..... 194
VI. Đánh giá xử lý đăng ký học phần đồng thời..... 194
VII. Đánh giá hiệu năng hệ thống..... 195
VIII. Nội dung nâng cao.....195
1. Đánh giá ảnh hưởng của Replication đối với dữ liệu dùng chung.....195
2. Đề xuất cơ chế xử lý khi một site tạm thời mất kết nối..... 195
3. So sánh phân mảnh theo cơ sở và phân mảnh theo khoa.....196
IX. Hạn chế của hệ thống..... 196
X. Định hướng phát triển.....196
XI. Kết luận..... 197

PHẦN 1: ĐẶT VẤN ĐỀ

I. Giới thiệu

1. Nhu cầu và tầm quan trọng của dự án

Hiện nay, hầu hết các trường đại học đều tổ chức đào tạo theo học chế tín chỉ. Hình thức này giúp sinh viên chủ động hơn trong việc lựa chọn môn học, xây dựng kế hoạch học tập và đăng ký học phần phù hợp. Tuy nhiên, đi kèm với đó là khối lượng dữ liệu rất lớn liên quan đến sinh viên, giảng viên, môn học, lớp học phần, thời khóa biểu, kết quả học tập và thông tin đăng ký tín chỉ.

Đối với các học viện hoặc trường đại học có nhiều cơ sở đào tạo tại các địa điểm khác nhau, việc sử dụng cơ sở dữ liệu tập trung dễ gây quá tải, đặc biệt vào các đợt cao điểm đăng ký học phần. Khi nhiều sinh viên cùng truy cập hệ thống, thời gian xử lý có thể chậm, phát sinh nghẽn mạng, sai lệch dữ liệu hoặc khó đồng bộ thông tin giữa các cơ sở.

Vì vậy, việc xây dựng hệ thống quản lý đăng ký học phần dựa trên cơ sở dữ liệu phân tán là cần thiết. Mô hình này cho phép dữ liệu được lưu trữ và xử lý tại từng cơ sở đào tạo, đồng thời vẫn đảm bảo khả năng đồng bộ, tổng hợp và quản lý dữ liệu toàn hệ thống tại máy chủ trung tâm.

Việc xây dựng một hệ thống cơ sở dữ liệu phân tán để hỗ trợ công tác quản lý sinh viên và đăng ký học phần tại các trường đại học là vấn đề cần thiết trong bối cảnh hiện nay. Đặc biệt đối với những trường có nhiều cơ sở đào tạo, dữ liệu phát sinh tại từng cơ sở ngày càng lớn, nhu cầu truy cập và cập nhật thông tin diễn ra thường xuyên. Nếu chỉ sử dụng mô hình cơ sở dữ liệu tập trung, hệ thống dễ gặp tình trạng quá tải, truy xuất chậm, khó mở rộng và ảnh hưởng trực tiếp đến hoạt động quản lý đào tạo.

Theo đó, hệ thống cơ sở dữ liệu phân tán mang lại nhiều giá trị quan trọng:

- **Giá trị sử dụng đối với sinh viên:** Sinh viên có thể dễ dàng tra cứu thông tin cá nhân, tiến trình học tập, thời khóa biểu, điểm số và kết quả đăng ký học phần một cách nhanh chóng. Trong các đợt cao điểm đăng ký tín chỉ, hệ thống phân tán giúp giảm tình trạng nghẽn truy cập do dữ liệu được xử lý tại các máy chủ theo từng cơ sở đào tạo. Nhờ đó, sinh viên có trải nghiệm sử dụng ổn định hơn, hạn chế tình trạng chậm, lỗi hoặc không thể truy cập vào hệ thống.
- **Giá trị đối với công tác quản lý của nhà trường:** Hệ thống giúp nhà trường quản lý dữ liệu sinh viên, giảng viên, môn học, lớp học phần và kết quả học tập một cách khoa học, chính xác hơn. Dữ liệu tại từng cơ sở được lưu trữ và xử lý cục bộ, đồng thời vẫn có thể đồng bộ về máy chủ trung tâm để phục vụ công tác tổng hợp, thống kê và báo cáo toàn trường. Điều này giúp giảm khối lượng xử

lý cho máy chủ trung tâm, tăng tốc độ truy xuất dữ liệu và nâng cao hiệu quả quản lý đào tạo.

- **Giá trị kinh tế:** Mặc dù chi phí xây dựng ban đầu của hệ thống cơ sở dữ liệu phân tán có thể cao hơn so với hệ thống tập trung, nhưng về lâu dài mô hình này giúp tiết kiệm chi phí vận hành, bảo trì và mở rộng hệ thống. Khi cần bổ sung thêm cơ sở đào tạo mới, nhà trường có thể triển khai thêm máy chủ trạm mà không làm gián đoạn hoạt động của toàn bộ hệ thống. Ngoài ra, việc phân tán dữ liệu cũng giúp giảm rủi ro mất dữ liệu tập trung và hỗ trợ quá trình sao lưu, phục hồi dữ liệu hiệu quả hơn.

- **Giá trị nghiệp vụ:** Hệ thống hỗ trợ xử lý các nghiệp vụ đào tạo một cách tự động, đồng thời và chính xác. Các thao tác như cập nhật hồ sơ sinh viên, mở lớp học phần, đăng ký học phần, nhập điểm, tra cứu thời khóa biểu và thống kê số lượng sinh viên đăng ký có thể được thực hiện nhanh chóng tại từng cơ sở. Nhờ đó, giảm bớt thao tác thủ công, hạn chế sai sót trong quá trình nhập liệu và đảm bảo dữ liệu luôn được cập nhật kịp thời.

- **Giá trị về an toàn và tính nhất quán dữ liệu:** Cơ sở dữ liệu phân tán giúp dữ liệu được quản lý chặt chẽ hơn theo từng khu vực, từng cơ sở đào tạo và từng nhóm người dùng. Mỗi đối tượng sử dụng hệ thống như sinh viên, giảng viên, nhân viên quản lý hay quản trị viên được phân quyền phù hợp với chức năng của mình. Điều này góp phần bảo vệ thông tin cá nhân, hạn chế truy cập trái phép và đảm bảo dữ liệu được khai thác một cách an toàn, chính xác và nhất quán.

2. Sơ lược về dự án

Dự án được triển khai theo mô hình cơ sở dữ liệu phân tán gồm một máy chủ trung tâm đặt tại trụ sở chính và các máy trạm đặt tại các cơ sở đào tạo. Máy chủ trung tâm quản lý dữ liệu toàn hệ thống, các danh mục dùng chung và thực hiện tổng hợp, đồng bộ dữ liệu từ các cơ sở.

Các dữ liệu dùng chung như Chương trình đào tạo, Chi tiết chương trình đào tạo, Học phần, Học kỳ, Học phần tiên quyết, Khung thời gian học được cập nhật tại máy chủ trung tâm, sau đó nhân bản xuống các máy trạm. Các dữ liệu nghiệp vụ như sinh viên, giảng viên, khoa, phòng học, lớp học phần, buổi học và đăng ký học phần được quản lý tại từng cơ sở đào tạo tương ứng.

Dữ liệu phát sinh tại máy trạm sẽ được đồng bộ về máy chủ trung tâm sau một khoảng thời gian nhất định tùy thuộc vào tốc độ mạng. Người dùng hệ thống gồm quản trị viên trung tâm, nhân viên quản lý tại cơ sở, giảng viên và sinh viên. Mỗi nhóm người dùng được phân quyền phù hợp với chức năng của mình nhằm đảm bảo việc quản lý, tra cứu và cập nhật dữ liệu diễn ra chính xác, an toàn và hiệu quả.

II. Vị trí và nhiệm vụ, dữ liệu khi triển khai dự án

1. Các vị trí

Hệ thống được đặt tại 3 vị trí địa lý tương ứng với 3 cơ sở đào tạo của Học viện, tạo thành một mạng lưới phân tán bao gồm 1 Server Trung tâm và 2 Server Chi nhánh:

- Server Trung tâm (Publisher / Máy chủ trung tâm): Đặt tại Trụ sở chính Hà Đông - Hà Nội (HQHĐ). Máy chủ này chứa CSDL toàn cục (Global Database), đóng vai trò xuất bản dữ liệu (Publication), quản lý danh mục chung và thu thập dữ liệu tổng hợp từ các chi nhánh. Đồng thời, máy chủ HQHĐ cũng hoạt động như một cơ sở đào tạo bình thường; các giao dịch phát sinh của cơ sở Hà Đông được xử lý và ghi trực tiếp vào CSDL toàn cục mà không cần thông qua bước đồng bộ hóa.
- Các Server Chi nhánh (Subscribers / Máy trạm): Đặt tại 2 cơ sở còn lại bao gồm: Hòa Lạc - Hà Nội (HQHL) và TP. Hồ Chí Minh (HQHCM). Mỗi server chỉ lưu trữ CSDL cục bộ (Local Database) thuộc quyền quản lý của cơ sở mình

2. Các đối tượng dữ liệu triển khai

a. Nhóm dữ liệu danh mục dùng chung

Nhóm dữ liệu danh mục dùng chung bao gồm các bảng: Curriculum, Curriculum_Subject, Subject, Term, Prerequisite, Timeslot.

Đây là các dữ liệu có tần suất cập nhật không quá cao nhưng được sử dụng thường xuyên tại tất cả các cơ sở đào tạo. Các bảng này được quản lý tập trung tại Server Trung tâm và nhân bản toàn phần xuống các Server Chi nhánh. Tại các chi nhánh, dữ liệu danh mục chủ yếu được sử dụng ở chế độ đọc nhằm phục vụ việc tra cứu môn học, học kỳ, ca học, điều kiện môn tiên quyết và thông tin cơ sở đào tạo.

b. Nhóm dữ liệu nghiệp vụ phân mảnh theo cơ sở

Nhóm dữ liệu nghiệp vụ bao gồm các bảng: department, room, teacher, student, class, session, registration.

Các bảng này phát sinh dữ liệu thường xuyên trong quá trình quản lý đào tạo, mở lớp, xếp lịch và đăng ký học phần. Vì vậy, dữ liệu được phân mảnh theo từng cơ sở đào tạo nhằm giảm tải cho Server Trung tâm, đồng thời tăng tốc độ xử lý tại các Server Chi nhánh.

Trong hệ thống, bảng headquarter được sử dụng làm quan hệ gốc để xác định phạm vi dữ liệu của từng cơ sở. Từ mảnh headquarter, các bảng nghiệp vụ được phân mảnh ngang dẫn xuất thông qua các quan hệ khóa ngoại. Bảng department được phân mảnh dẫn xuất theo headquarter vì mỗi khoa thuộc một cơ sở đào tạo. Bảng room cũng được phân mảnh dẫn xuất theo headquarter vì mỗi phòng học thuộc một cơ sở cụ thể.

Tiếp theo, bảng teacher và bảng student được phân mảnh dẫn xuất thông qua bảng department, do mỗi giảng viên và sinh viên đều thuộc một khoa, mà mỗi khoa lại thuộc một cơ sở đào tạo. Bảng class được phân mảnh dẫn xuất thông qua bảng teacher, vì mỗi lớp học phần do một giảng viên phụ trách và giảng viên thuộc một cơ sở thông qua khoa. Bảng session được phân mảnh dẫn xuất thông qua bảng class và room, nhằm đảm bảo buổi học được xếp đúng theo lớp học phần và phòng học của cơ sở tương ứng. Bảng registration được phân mảnh dẫn xuất thông qua bảng student và class, phản ánh các giao dịch đăng ký học phần phát sinh trong phạm vi quản lý của từng cơ sở.

Đối với các nghiệp vụ đăng ký học phần cục bộ, dữ liệu được xử lý trực tiếp tại Server Chi nhánh nơi sinh viên và lớp học phần thuộc cùng một cơ sở. Đối với trường hợp đăng ký chéo cơ sở, máy trạm gửi yêu cầu lên Server Trung tâm Hà Đông để xử lý. Server Trung tâm đóng vai trò điều phối, kiểm tra thông tin sinh viên, lớp học phần, thời khóa biểu và sĩ số trên CSDL toàn cục, sau đó thực hiện giao dịch đăng ký trong Transaction để đảm bảo tính nhất quán dữ liệu. Cách xử lý này giúp tránh lỗi khóa ngoại khi sinh viên và lớp học phần thuộc hai cơ sở khác nhau.

3. Các công việc dữ liệu triển khai

Để hệ thống phân tán vận hành trơn tru, đảm bảo tính nhất quán dữ liệu và tối ưu hóa hiệu năng giữa 3 cơ sở đào tạo, các công việc tác động lên dữ liệu được phân chia rõ ràng trách nhiệm theo từng vị trí máy chủ như sau:

a. Tại Server Trung tâm (Trụ sở Hà Đông - HQHĐ): Máy chủ trung tâm đảm nhận vai trò quản lý toàn cục, điều phối hệ thống và đồng thời xử lý các nghiệp vụ nội bộ cho cơ sở Hà Đông. Các công việc cụ thể bao gồm:

- Quản trị dữ liệu danh mục dùng chung: Quản trị viên tại trung tâm độc quyền thao tác Thêm/Sửa/Xóa đối với các dữ liệu từ điển như mở môn học mới (subject), mở kỳ học mới (term), thiết lập ca học (timeslot) và quy định môn tiên quyết (prerequisite). Dữ liệu này sau đó được cơ chế Transactional Replication nhân bản một chiều (đẩy xuống) các máy trạm chi nhánh

- Xử lý nghiệp vụ cục bộ của cơ sở Hà Đông: Do đóng vai trò "kép", mọi giao dịch quản lý đào tạo nội bộ (tạo lớp, xếp lịch) và đăng ký tín chỉ của sinh viên cơ sở Hà Đông được xử lý và ghi nhận trực tiếp vào CSDL toàn cục mà không cần thông qua tiến trình đồng bộ ngầm

- Điều phối đăng ký chéo cơ sở: Đối với các trường hợp sinh viên đăng ký lớp học phần thuộc cơ sở khác, Server Trung tâm đóng vai trò điều phối xử lý. Máy trạm chi nhánh gửi yêu cầu đăng ký chéo lên Server Trung tâm. Tại đây, hệ thống kiểm tra thông tin sinh viên, lớp học phần, thời khóa biểu, trạng thái lớp, thời gian đăng ký và sĩ số lớp trên CSDL toàn cục. Nếu hợp lệ, Server Trung tâm thực hiện giao dịch đăng ký trong một Transaction để đảm bảo dữ liệu nhất

quán. Cách xử lý này giúp tránh lỗi khóa ngoại khi sinh viên và lớp học phần thuộc hai cơ sở khác nhau.

- Tổng hợp và Báo cáo cấp trường: Server Trung tâm thực hiện các Stored Procedure và truy vấn phân tán để tổng hợp dữ liệu toàn hệ thống, ví dụ: thống kê số sinh viên đăng ký theo cơ sở, thống kê học phần có nhiều sinh viên đăng ký nhất, tỷ lệ lấp đầy lớp học phần, khối lượng giảng dạy của giảng viên và danh sách sinh viên đăng ký chéo cơ sở.

b. Tại các Server Chi nhánh (Cơ sở Hòa Lạc và TP. HCM): Đóng vai trò là các Subscriber lưu trữ dữ liệu phân mảnh ngang cục bộ, các máy trạm chịu trách nhiệm chính trong việc giảm tải cho máy chủ trung tâm thông qua các công việc sau:

- Xử lý nghiệp vụ đào tạo cục bộ: Tại mỗi chi nhánh, các thao tác đọc, thêm, sửa, xóa đối với dữ liệu nghiệp vụ như student, teacher, room, class, session và registration được thực hiện trực tiếp trên Local Database của cơ sở đó. Ví dụ, máy trạm Hòa Lạc xử lý dữ liệu thuộc cơ sở Hòa Lạc, máy trạm TP. Hồ Chí Minh xử lý dữ liệu thuộc cơ sở TP. Hồ Chí Minh. Cách tổ chức này giúp các nghiệp vụ thường xuyên như mở lớp, xếp lịch và đăng ký học phần cục bộ được xử lý nhanh, không phụ thuộc hoàn toàn vào đường truyền về máy chủ trung tâm.

- Tra cứu dữ liệu dùng chung: Các máy trạm sử dụng dữ liệu danh mục đã được nhân bản từ máy chủ trung tâm như subject, term, timeslot, prerequisite, curriculum và curriculum_subject để phục vụ tra cứu, mở lớp, xếp lịch và kiểm tra điều kiện đăng ký học phần. Các dữ liệu này chủ yếu được đọc tại máy trạm, còn việc cập nhật được quản lý tại Server Trung tâm.

- Xử lý yêu cầu đăng ký chéo cơ sở: Khi sinh viên tại một chi nhánh muốn đăng ký lớp học phần thuộc cơ sở khác, máy trạm không ghi trực tiếp dữ liệu đăng ký vào cơ sở dữ liệu của site khác. Thay vào đó, máy trạm gửi yêu cầu lên Server Trung tâm Hà Đông để xử lý. Server Trung tâm kiểm tra dữ liệu toàn cục và trả về kết quả thành công hoặc thất bại cho máy trạm hiển thị.

- Đồng bộ hóa dữ liệu : Do hệ thống sử dụng Push Subscription, Merge Agent được chạy tại Distributor là máy chủ trung tâm Hà Đông. Agent này chạy theo chế độ Run continuously để đồng bộ dữ liệu hai chiều giữa Publisher và Subscriber.

III. Các đối tượng tham gia sử dụng dự án

Hệ thống phục vụ 4 nhóm đối tượng với phân quyền truy cập khác nhau:

1. Sinh viên:

- Xem và cập nhật thông tin cá nhân: họ tên, ngày sinh, giới tính, địa chỉ, số điện thoại, email.
- Đăng ký học phần và hủy đăng ký trong thời gian cho phép; hệ thống tự động kiểm tra điều kiện còn chỗ trước khi xác nhận.
- Đăng ký chéo cơ sở: được phép đăng ký lớp học phần mở tại cơ sở khác khi lớp tại cơ sở mình không đủ chỗ hoặc không mở; giao dịch được gửi lên Server Trung tâm Hà Đông để kiểm tra và xử lý trên CSDL toàn cục nhằm đảm bảo tính nhất quán dữ liệu.
- Tra cứu kết quả đăng ký: xem danh sách học phần đã đăng ký, trạng thái xác nhận và thời gian đăng ký.
- Xem thời khóa biểu cá nhân: thông tin phòng học và lịch học của các lớp đã đăng ký.

2. Giảng viên:

- Xem và cập nhật thông tin cá nhân: họ tên, học vị, học hàm, địa chỉ, số điện thoại.
- Xem danh sách các lớp học phần được phân công giảng dạy trong kỳ học.
- Xem danh sách sinh viên đã đăng ký trong từng lớp học phần mình phụ trách, bao gồm cả sinh viên đăng ký chéo từ cơ sở khác.
- Xem thông tin phòng học và lịch học của các lớp được phân công.

3. Nhân viên quản lý (tại các cơ sở chi nhánh):

- Quản lý thông tin sinh viên và giảng viên tại cơ sở mình: thêm mới, cập nhật, tra cứu.
- Quản lý lớp học phần: mở lớp, phân công giảng viên, thiết lập sĩ số tối đa và tối thiểu.
- Quản lý lịch học: phân công phòng học và khung giờ cho từng lớp học phần.
- Theo dõi tình trạng đăng ký và xem báo cáo thống kê của cơ sở, bao gồm số lượng sinh viên đăng ký chéo từ cơ sở khác vào lớp tại cơ sở mình và ngược lại.

4. Quản trị viên hệ thống (tại máy chủ trung tâm - Hà Đông):

- Có đầy đủ mọi quyền của các nhóm đối tượng trên đối với toàn bộ dữ liệu hệ thống.
- Quản lý danh mục học phần, khoa và ngành đào tạo dùng chung cho toàn hệ thống.

- Quản lý thông tin các cơ sở đào tạo: tên cơ sở, địa chỉ, hotline, trạng thái hoạt động.
- Giám sát trạng thái kết nối và đồng bộ dữ liệu giữa máy chủ trung tâm và 2 site chi nhánh.
- Xem báo cáo thống kê toàn trường: tổng số sinh viên đăng ký theo cơ sở, học phần được đăng ký nhiều nhất, tỷ lệ lấp đầy lớp, danh sách sinh viên đăng ký chéo cơ sở.

PHẦN 2: PHÂN TÍCH

I. Các chức năng chính của hệ thống trong dự án

Dựa trên lược đồ thực thể của hệ thống, các chức năng nghiệp vụ được chia thành 2 nhóm dữ liệu cốt lõi:

1. Quản lý nhóm dữ liệu Danh mục/Từ điển (Dữ liệu chung):

- Quản lý danh mục đào tạo: Quản lý cấu trúc Chương trình đào tạo (Curriculum), Môn học (Subject), Chi tiết môn học trong chương trình (Curriculum_Subject) và Môn học tiên quyết (Prerequisite)
- Quản lý thời gian: Khởi tạo Học kỳ (Term) và các Khung/Ca học chuẩn (Timeslot) áp dụng cho toàn hệ thống

2. Quản lý nhóm dữ liệu Nghiệp vụ (Dữ liệu phân tán):

- Quản lý tổ chức và cơ sở vật chất: Quản lý các Cơ sở đào tạo (Headquarter), các Khoa (Department) và hệ thống Phòng học (Room)
- Quản lý hồ sơ nhân sự: Quản lý thông tin Giảng viên (Teacher) và Sinh viên (Student)
- Quản lý mở lớp và xếp lịch: Quản lý Lớp học phần (Class), xếp lịch cho từng Buổi học (Session) liên kết với giảng viên, phòng học và ca học cụ thể
- Xử lý giao dịch đăng ký tín chỉ: Hỗ trợ sinh viên thực hiện giao dịch Đăng ký/Hủy đăng ký học phần (Registration), bao gồm cả nghiệp vụ xử lý đăng ký chéo cơ sở
- Báo cáo và thống kê: Thống kê kết quả đăng ký cục bộ tại từng cơ sở và tổng hợp toàn trường

II. Phân quyền cho các nhóm đối tượng thực hiện dự án

Hệ thống phân chia quyền hạn tương ứng với các thực thể giao tiếp

Sinh viên (Tại các cơ sở): Có quyền tra cứu/cập nhật thông tin cá nhân. Được phép xem danh mục Subject, Term, kiểm tra Prerequisite để tiến hành Registration (Đăng ký học phần). Sinh viên có thể xem thời khóa biểu cá nhân thông qua việc kết xuất dữ liệu từ các bảng Class, Session, Room, và Timeslot. Có quyền thực hiện giao dịch đăng ký chéo cơ sở.

Giảng viên (Tại các cơ sở): Xem/cập nhật thông tin cá nhân. Xem danh sách các lớp học phần (Class) và các buổi học (Session) được phân công giảng dạy. Xem danh sách sinh viên (Student) trong lớp, bao gồm cả sinh viên học chéo.

Nhân viên quản lý/Giáo vụ (Tại máy trạm chi nhánh): Có quyền thao tác quản lý dữ liệu nội bộ tại cơ sở như Student, Teacher, Room. Thực hiện mở Class, xếp Session cho lớp. Theo dõi tình trạng Registration tại cơ sở và lập các báo cáo thống kê cục bộ.

Quản trị viên hệ thống (Tại trụ sở chính - Máy chủ): Có toàn quyền truy cập dữ liệu của cả 3 cơ sở. Đây là đối tượng duy nhất có quyền Thêm/Sửa/Xóa các thực thể dùng chung như Subject, Term, Curriculum, Prerequisite, Timeslot. Thực hiện giám sát đồng bộ, xử lý xung đột đăng ký chéo cơ sở và trích xuất báo cáo phân tán toàn trường.

III. Phân tích chức năng của từng vị trí thực hiện dự án

Mô hình triển khai gồm 1 Server Trung tâm và 2 Server Chi nhánh

1. Tại trụ sở chính (Máy chủ trung tâm)

- Lưu trữ CSDL toàn cục (Global Database) và đóng vai trò Publisher xuất bản dữ liệu danh mục.
- Quản trị viên tại đây độc quyền thiết lập dữ liệu khung chương trình đào tạo (Curriculum, Subject, Term, Prerequisite) và nhân bản (Transactional Replication) xuống các trạm.
- Đóng vai trò là trung gian trong việc xử lý đăng ký chéo giữa 2 cơ sở chi nhánh
- Tổng hợp dữ liệu đồng bộ từ các trạm về để chạy báo cáo tổng.

2. Tại các cơ sở Hòa Lạc, TP. HCM (Máy trạm):

- Đóng vai trò là Subscriber, lưu trữ CSDL cục bộ theo phương pháp phân mảnh ngang (dựa trên khóa ngoại *ID_headquarter*)
- Xử lý mọi nghiệp vụ quản lý đào tạo nội bộ (tạo Session, xếp Room) và giao dịch Registration của sinh viên cơ sở đó với tốc độ rất cao nhờ truy xuất dữ liệu Local
- Khi có sinh viên đăng ký chéo, máy trạm gửi truy vấn phân tán lên máy chủ để xác thực. Đồng thời, tiến trình Merge Agent sẽ định kỳ quét các thay đổi tại bảng Student, Class, Registration... để đồng bộ ngược về máy chủ Hà Đông.

IV. Chức năng của máy trạm, máy chủ

1. Chức năng ở máy trạm

Quản lý và vận hành dữ liệu cục bộ (Tần suất H.RWED): Máy trạm trực tiếp xử lý các giao dịch Thêm/Sửa/Xóa/Đọc liên tục đối với các thực thể phát sinh tại cơ sở bao gồm: Student, Room, Class, Session, Registration. Mọi thay đổi sẽ được ghi nhận tại Local DB trước khi đồng bộ.

Tra cứu dữ liệu danh mục (Tần suất H.R): Đối với các thực thể Headquarter, Department, Curriculum, Subject, Term, Timeslot, Prerequisite, Curriculum_Subject,

máy trạm chỉ có quyền Đọc (Read-only) tần suất cao để phục vụ hiển thị khi sinh viên thao tác đăng ký tín chỉ và giáo vụ xếp lịch học.

2. Chức năng ở máy chủ

Bao gồm toàn bộ chức năng của máy trạm đối với dữ liệu của sinh viên theo học trực tiếp tại Hà Đông.

Quản trị và thiết lập danh mục (Tần suất L.WED / H.RWED): Có thẩm quyền thiết lập mới, chỉnh sửa hoặc xóa dữ liệu đối với các bảng Headquarter, Department, Curriculum, Subject, Term, Timeslot, Prerequisite. Dữ liệu này tuy có tần suất thay đổi thấp (L.WED) nhưng mang tính chất sống còn để định tuyến hoạt động toàn hệ thống.

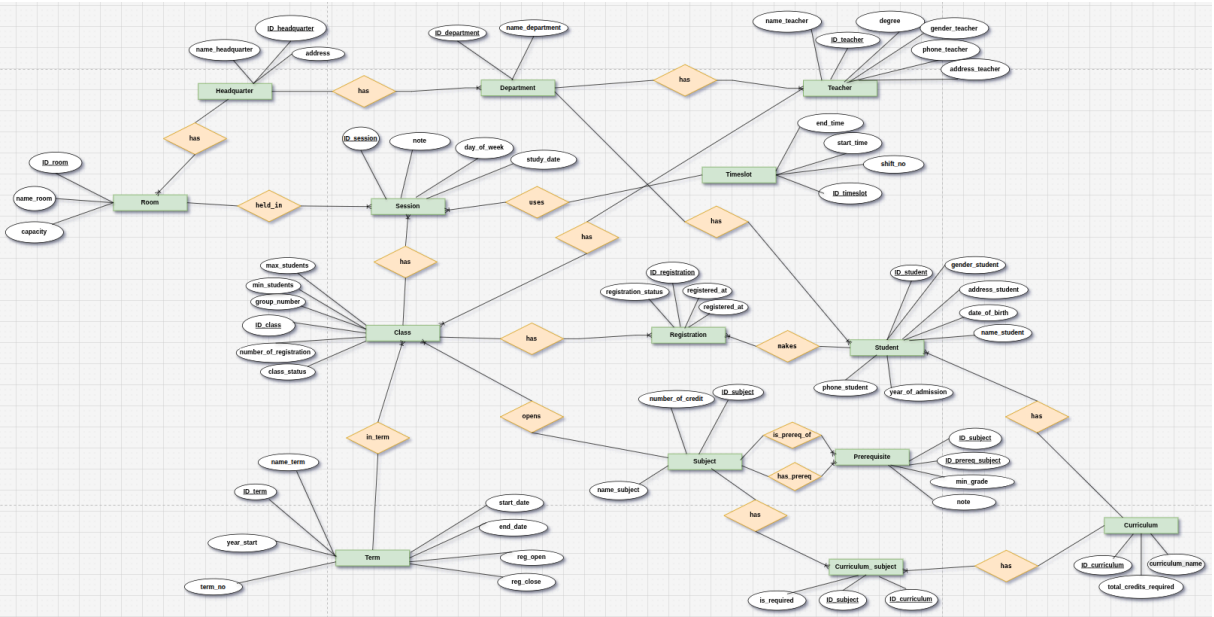
V. Phân tích cơ sở dữ liệu

1. Lược đồ thực thể E-R

Phân tích lược đồ E-R

- Headquarter - Department là mối quan hệ một - nhiều vì một trụ sở có thể có nhiều khoa, nhưng một khoa thuộc một trụ sở.
- Headquarter - Room là mối quan hệ một - nhiều vì một trụ sở có thể có nhiều phòng học, nhưng một phòng học chỉ thuộc một trụ sở.
- Department - Teacher là mối quan hệ một - nhiều vì một khoa có thể có nhiều giảng viên, nhưng một giảng viên thuộc một khoa.
- Department - Student là mối quan hệ một - nhiều vì một khoa có thể có nhiều sinh viên, nhưng một sinh viên thuộc một khoa.
- Curriculum - Student là mối quan hệ một - nhiều vì một chương trình đào tạo có thể áp dụng cho nhiều sinh viên, nhưng một sinh viên theo một chương trình đào tạo.
- Subject - Class là mối quan hệ một - nhiều vì một môn học có thể được mở thành nhiều lớp học phần, nhưng một lớp học phần thuộc một môn học.
- Teacher - Class là mối quan hệ một - nhiều vì một giảng viên có thể dạy nhiều lớp học phần, nhưng một lớp học phần do một giảng viên phụ trách.
- Term - Class là mối quan hệ một - nhiều vì một học kỳ có thể có nhiều lớp học phần được mở, nhưng một lớp học phần thuộc một học kỳ.
- Class - Session là mối quan hệ một - nhiều vì một lớp học phần có thể có nhiều buổi học, nhưng một buổi học thuộc một lớp học phần.

- Room - Session là mối quan hệ một - nhiều vì một phòng học có thể được sử dụng cho nhiều buổi học khác nhau, nhưng một buổi học diễn ra tại một phòng học.
- Timeslot - Session là mối quan hệ một - nhiều vì một ca học có thể được dùng cho nhiều buổi học, nhưng một buổi học chỉ sử dụng một ca học.
- Student - Class là mối quan hệ nhiều - nhiều vì một sinh viên có thể đăng ký nhiều lớp học phần, và một lớp học phần có thể có nhiều sinh viên đăng ký. Mối quan hệ này được tách thành thực thể trung gian Registration.
- Curriculum - Subject là mối quan hệ nhiều - nhiều vì một chương trình đào tạo có thể gồm nhiều môn học, và một môn học có thể thuộc nhiều chương trình đào tạo. Mối quan hệ này được tách thành thực thể trung gian Curriculum_Subject.
- Subject - Prerequisite Subject là mối quan hệ đệ quy nhiều - nhiều vì một môn học có thể có nhiều môn tiên quyết, và một môn học cũng có thể là môn tiên quyết của nhiều môn học khác. Mối quan hệ này được tách thành thực thể Prerequisite.



Xem chi tiết

2. Bảng tần suất truy cập các vị trí

- W: tạo mới và ghi
- E: Sửa
- D: Xóa
- R: Đọc

- H: Tần suất cao
- L: Tần suất thấp

Thực thể	Server Trung tâm	Các máy trạm
Headquarter	H.R, L.W.E.D	H.R
Department	H.R, L.W.E.D	H.R
Room	H.R, H.W.E.D	H.R.W.E.D
Teacher	H.R, L.W.E.D	H.R, L.W.E.D
Curriculum	H.R, L.W.E.D	H.R
Student	H.R, H.W.E.D	H.R.W.E.D
Subject	H.R, L.W.E.D	H.R
Class	H.R, H.W.E.D	H.R.W.E.D
Term	H.R, L.W.E.D	H.R
Session	H.R, H.W.E.D	H.R.W.E.D
Timeslot	H.R, L.W.E.D	H.R
Registration	H.R, H.W.E.D	H.R.W.E.D

Curriculum_Subject	H.R, L.W.E.D	H.R
Prerequisite	H.R, L.W.E.D	H.R

Phân tích:

Headquarter: Các trụ sở

- Ở Server Trung tâm, dữ liệu cơ sở đào tạo được đọc thường xuyên để phục vụ phân quyền, định tuyến dữ liệu, thống kê và quản lý toàn hệ thống. Việc thêm, sửa, xóa cơ sở có tần suất thấp vì số lượng cơ sở đào tạo thường ổn định.
- Ở các máy trạm, dữ liệu cơ sở đào tạo chủ yếu được đọc để xác định cơ sở hiện tại, hiển thị thông tin và phục vụ liên kết với các bảng khác. Máy trạm không trực tiếp cập nhật dữ liệu này để đảm bảo tính nhất quán toàn hệ thống.

Department: Khoa

- Ở Server Trung tâm, dữ liệu khoa được đọc thường xuyên để quản lý sinh viên, giảng viên, phân quyền và thống kê toàn trường. Việc thêm, sửa, xóa khoa có tần suất thấp vì cơ cấu khoa ít thay đổi.
- Ở các máy trạm, dữ liệu khoa chủ yếu được đọc để liên kết với sinh viên và giảng viên thuộc cơ sở đó. Việc cập nhật khoa không thực hiện trực tiếp tại máy trạm mà được quản lý tập trung tại Server Trung tâm.

Room: Phòng học

- Ở Server Trung tâm, dữ liệu phòng học được đọc thường xuyên để tổng hợp tình hình sử dụng phòng tại các cơ sở. Đồng thời, do Hà Đông cũng là một cơ sở đào tạo, Server Trung tâm vẫn có thao tác thêm, sửa, xóa phòng học thuộc cơ sở Hà Đông với tần suất cao.
- Ở các máy trạm, dữ liệu phòng học được đọc, thêm, sửa, xóa thường xuyên vì mỗi cơ sở trực tiếp quản lý phòng học của mình, cập nhật sức chứa, trạng thái và phục vụ xếp lịch học.

Teacher: Giảng viên

- Ở Server Trung tâm, dữ liệu giảng viên được đọc thường xuyên để tổng hợp, thống kê và theo dõi khối lượng giảng dạy toàn hệ thống. Việc thêm, sửa, xóa thông tin giảng viên có tần suất thấp vì thông tin giảng viên không thay đổi liên tục.

- Ở các máy trạm, dữ liệu giảng viên được đọc thường xuyên để mở lớp học phần và xếp lịch giảng dạy. Các thao tác thêm, sửa, xóa có tần suất thấp, chủ yếu khi có thay đổi nhân sự hoặc cập nhật hồ sơ giảng viên tại cơ sở.

Curriculum: Chương trình đào tạo

- Ở Server Trung tâm, chương trình đào tạo được đọc thường xuyên để quản lý sinh viên, môn học và điều kiện đào tạo. Việc thêm, sửa, xóa chương trình đào tạo có tần suất thấp vì chương trình thường ổn định theo khóa học.
- Ở các máy trạm, dữ liệu chương trình đào tạo chủ yếu được đọc để xác định sinh viên học theo chương trình nào và hỗ trợ kiểm tra môn học thuộc chương trình đào tạo.

Student: Sinh viên

- Ở Server Trung tâm, dữ liệu sinh viên được đọc thường xuyên để thống kê, tổng hợp và phục vụ các truy vấn toàn hệ thống. Đồng thời, do Hà Đông là một cơ sở đào tạo, Server Trung tâm cũng xử lý các thao tác thêm, sửa, xóa sinh viên thuộc cơ sở Hà Đông với tần suất cao.
- Ở các máy trạm, dữ liệu sinh viên được đọc, thêm, sửa, xóa với tần suất cao vì mỗi cơ sở trực tiếp quản lý sinh viên thuộc phạm vi của mình.

Subject: Môn học

- Ở Server Trung tâm, dữ liệu môn học được đọc thường xuyên và chỉ thỉnh thoảng được thêm, sửa, xóa khi chương trình đào tạo thay đổi hoặc có môn học mới.
- Ở các máy trạm, dữ liệu môn học chủ yếu được đọc để mở lớp học phần, đăng ký học phần và tra cứu thông tin học phần. Máy trạm không trực tiếp cập nhật dữ liệu môn học để đảm bảo thống nhất danh mục toàn trường.

Class: Lớp học phần

- Ở Server Trung tâm, dữ liệu lớp học phần được đọc thường xuyên để theo dõi số lớp mở, sĩ số và tình trạng đăng ký trên toàn hệ thống. Ngoài ra, các lớp học phần thuộc cơ sở Hà Đông được tạo, sửa và hủy trực tiếp tại Server Trung tâm với tần suất cao trong giai đoạn đầu học kỳ.
- Ở các máy trạm, lớp học phần được đọc, tạo mới, sửa và hủy với tần suất cao, đặc biệt trong giai đoạn lập kế hoạch đào tạo và đăng ký học phần.

Term: Học kỳ

- Ở Server Trung tâm, dữ liệu học kỳ được đọc thường xuyên để quản lý lớp học phần, lịch học và thời gian đăng ký. Việc thêm, sửa, xóa học kỳ có tần suất thấp vì học kỳ được tạo theo kế hoạch đào tạo cố định.

- Ở các máy trạm, dữ liệu học kỳ chủ yếu được đọc để mở lớp, lập lịch học và kiểm tra thời gian đăng ký học phần.

Session: Buổi học

- Ở Server Trung tâm, dữ liệu buổi học được đọc thường xuyên để tổng hợp thời khóa biểu và kiểm tra lịch học toàn hệ thống. Do Hà Đông cũng là một cơ sở đào tạo, dữ liệu buổi học thuộc Hà Đông được thêm, sửa, xóa trực tiếp tại Server Trung tâm với tần suất cao.
- Ở các máy trạm, dữ liệu buổi học được đọc, thêm, sửa, xóa với tần suất cao vì mỗi cơ sở trực tiếp xếp lịch, đổi phòng, đổi ca và cập nhật lịch học.

Timeslot: Ca học

- Ở Server Trung tâm, dữ liệu ca học được đọc thường xuyên và chỉ ít khi thay đổi vì ca học thường được quy định thống nhất cho toàn hệ thống.
- Ở các máy trạm, dữ liệu ca học chủ yếu được đọc khi lập lịch học cho các buổi học và khi sinh viên tra cứu thời khóa biểu. Máy trạm không trực tiếp cập nhật dữ liệu ca học.

Registration: Đăng ký học phần

- Ở Server Trung tâm, dữ liệu đăng ký được đọc thường xuyên để thống kê số lượng sinh viên đăng ký, tỷ lệ lấp đầy lớp và truy vấn toàn trường. Đồng thời, các đăng ký của sinh viên cơ sở Hà Đông được xử lý trực tiếp tại Server Trung tâm với tần suất cao trong thời gian mở đăng ký.
- Ở các máy trạm, dữ liệu đăng ký được đọc, ghi, sửa và xóa với tần suất cao, đặc biệt trong thời gian mở đăng ký học phần. Các đăng ký cục bộ được xử lý tại máy trạm và đồng bộ về Server Trung tâm.

Curriculum_Subject: Môn học trong chương trình đào tạo

- Ở Server Trung tâm, dữ liệu môn học thuộc chương trình đào tạo được đọc thường xuyên và cập nhật với tần suất thấp khi chương trình đào tạo thay đổi.
- Ở các máy trạm, dữ liệu này chủ yếu được đọc để kiểm tra môn học thuộc chương trình nào, môn học bắt buộc hay tự chọn. Máy trạm không trực tiếp cập nhật dữ liệu này.

Prerequisite: Môn học tiên quyết

- Ở Server Trung tâm, dữ liệu môn học tiên quyết được đọc thường xuyên để kiểm tra điều kiện đăng ký học phần. Việc thêm, sửa, xóa có tần suất thấp vì điều kiện tiên quyết ít thay đổi.

- Ở các máy trạm, dữ liệu môn học tiên quyết chủ yếu được đọc khi sinh viên đăng ký lớp học phần để kiểm tra điều kiện học trước. Máy trạm không trực tiếp cập nhật dữ liệu này.

PHẦN 3: ĐẶT VẤN ĐỀ

I. Thiết kế hệ thống mạng tổng quan

- Trụ sở chính Hà Đông – Hà Nội: Đặt máy Server chính tại Hà Đông, Hà Nội, thực hiện các chức năng quản lý trung tâm và lưu trữ CSDL của toàn hệ thống.
- Trụ sở đào tạo Hòa Lạc – Hà Nội: Đặt máy Server trạm 1 và các client tương ứng, thực hiện chức năng và lưu toàn bộ CSDL thuộc cơ sở Hòa Lạc.
- Trụ sở đào tạo TP. Hồ Chí Minh: Đặt máy Server trạm 2 và các client tương ứng, thực hiện chức năng và lưu toàn bộ CSDL thuộc cơ sở TP. Hồ Chí Minh.

II. Thiết kế cơ sở dữ liệu của hệ thống

1. Cấu trúc các bảng dữ liệu của hệ thống

dbo.headquarter				
Tên thuộc tính	**Kiểu**	**Ràng buộc**	**Khóa**	**Mô tả**
ID_headquarter	varchar(20)	not null	PK	ID trụ sở
name_headquarter	nvarchar(100)	not null		Tên trụ sở
address	nvarchar(200)	not null		Địa chỉ trụ sở

dbo.department				
Tên thuộc tính	**Kiểu**	**Ràng buộc**	**Khóa**	**Mô tả**
ID_department	varchar(20)	not null	PK	ID khoa
name_department	nvarchar(100)	not null		Tên khoa
ID_headquarter	varchar(20)	not null	FK	ID trụ sở đào tạo

dbo.subject				
Tên thuộc tính	**Kiểu**	**Ràng buộc**	**Khóa**	**Mô tả**
ID_subject	varchar(20)	not null	PK	ID môn học
name_subject	nvarchar(100)	not null		Tên môn học
number_of_credit	int	not null, check > 0		Số tín chỉ

dbo.teacher				
Tên thuộc tính	**Kiểu**	**Ràng buộc**	**Khóa**	**Mô tả**
ID_teacher	varchar(20)	not null	PK	ID giảng viên
name_teacher	nvarchar(100)	not null		Tên giảng viên
degree	nvarchar(100)			Bằng cấp
address_teacher	nvarchar(200)			Địa chỉ
phone_teacher	varchar(15)			Số điện thoại
gender_teacher	nvarchar(10)			Giới tính giảng viên
ID_department	varchar(20)	not null	FK	ID khoa

dbo.student

Tên thuộc tính	Kiểu	Ràng buộc	Khóa	Mô tả
ID_student	varchar(20)	not null	PK	ID sinh viên
name_student	nvarchar(100)	not null		Tên sinh viên
date_of_birth	date			Ngày sinh
gender_student	nvarchar(10)			Giới tính
address_student	nvarchar(200)			Địa chỉ sinh viên
phone_student	varchar(15)			Số điện thoại sinh viên
year_of_admission	int	not null		Năm nhập học
ID_department	varchar(20)	not null	FK	ID khoa
ID_curriculum	varchar(20)	not null	FK	ID chương trình đào tạo

dbo.room				
Tên thuộc tính	**Kiểu**	**Ràng buộc**	**Khóa**	**Mô tả**
ID_room	varchar(20)	not null	PK	ID phòng học
name_room	nvarchar(50)	not null		Tên phòng
capacity	int	not null		Sức chứa
ID_headquarter	varchar(20)	not null	FK	ID trụ sở

dbo.term				
Tên thuộc tính	**Kiểu**	**Ràng buộc**	**Khóa**	**Mô tả**
ID_term	varchar(20)	not null	PK	ID học kỳ
name_term	nvarchar(100)	not null		Tên học kỳ
year_start	int	not null		Năm bắt đầu
term_no	int	not null		Số kỳ: 1, 2, 3
start_date	date	not null		Ngày bắt đầu kỳ học
end_date	date	not null		Ngày kết thúc kỳ học
reg_open	datetime	not null		Thời gian mở đăng ký
reg_close	datetime	not null		Thời gian đóng đăng ký

dbo.timeslot				
Tên thuộc tính	**Kiểu**	**Ràng buộc**	**Khóa**	**Mô tả**
ID_timeslot	varchar(20)	not null	PK	ID ca học
shift_no	int	not null		Số ca
start_time	time	not null		Giờ bắt đầu
end_time	time	not null		Giờ kết thúc

dbo.class

Tên thuộc tính	Kiểu	Ràng buộc	Khóa	Mô tả
ID_class	varchar(20)	not null	PK	ID lớp học phần
ID_subject	varchar(20)	not null	FK	ID môn học
ID_teacher	varchar(20)	not null	FK	ID giảng viên
ID_term	varchar(20)	not null	FK	ID kỳ học
group_number	int	not null		Nhóm lớp học phần
min_students	int	not null, check >= 0		Số sinh viên tối thiểu
max_students	int	not null, check > 0		Số sinh viên tối đa
number_of_registration	int	not null, default 0		Số lượng sinh viên đã đăng ký
class_status	varchar(20)	not null, default 'OPEN'		Trạng thái lớp học phần

dbo.session				
Tên thuộc tính	**Kiểu**	**Ràng buộc**	**Khóa**	**Mô tả**
ID_session	varchar(20)	not null	PK	ID buổi học
study_date	date	not null		Ngày học
day_of_week	tinyint	not null, check between 2 and 8		Thứ trong tuần

note	nvarchar(200)			Ghi chú
ID_class	varchar(20)	not null	FK	ID lớp học phần
ID_room	varchar(20)	not null	FK	ID phòng học
ID_timeslot	varchar(20)	not null	FK	ID ca học

dbo.registration				
Tên thuộc tính	**Kiểu**	**Ràng buộc**	**Khóa**	**Mô tả**
ID_registration	varchar(20)	not null	PK	ID đăng ký
ID_student	varchar(20)	not null	FK	ID sinh viên
ID_class	varchar(20)	not null	FK	ID lớp học phần
registered_at	datetime	not null, default getdate()		Thời gian đăng ký
cancelled_at	datetime	null		Thời gian hủy đăng ký
registration_status	varchar(20)	not null, default 'REGISTERED'		Trạng thái đăng ký

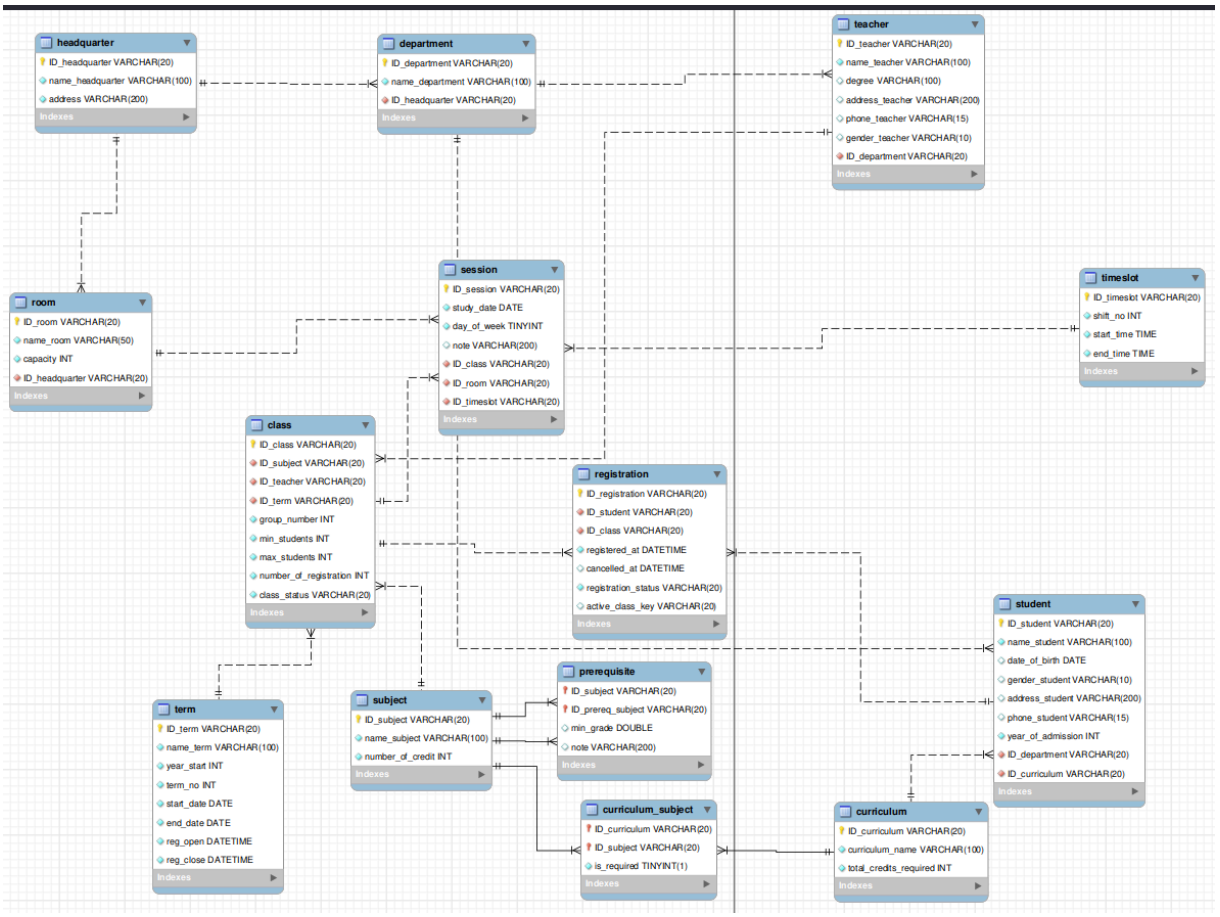
dbo.curriculum				
Tên thuộc tính	**Kiểu**	**Ràng buộc**	**Khóa**	**Mô tả**
ID_curriculum	varchar(20)	not null	PK	ID chương trình đào tạo

curriculum_name	nvarchar(100)	not null		Tên chương trình đào tạo
total_credits_required	int	not null, check > 0		Tổng tín chỉ yêu cầu

dbo.curriculum_subject				
Tên thuộc tính	**Kiểu**	**Ràng buộc**	**Khóa**	**Mô tả**
ID_curriculum	varchar(20)	not null	PK, FK	ID chương trình đào tạo
ID_subject	varchar(20)	not null	PK, FK	ID môn học
is_required	int	not null		1: Bắt buộc, 0: Không bắt buộc

dbo.prerequisite				
Tên thuộc tính	**Kiểu**	**Ràng buộc**	**Khóa**	**Mô tả**
ID_subject	varchar(20)	not null	PK, FK	ID môn học
ID_prereq_subject	varchar(20)	not null	PK, FK	ID môn học tiên quyết
min_grade	float			Điểm tối thiểu
note	nvarchar(200)			Ghi chú

2. Quan hệ giữa các bảng



III. Thiết kế cơ sở dữ liệu phân tán

<u>Lý do lựa chọn phân mảnh ngang theo cơ sở</u>

Hệ thống đăng ký học phần nhiều cơ sở có dữ liệu phát sinh gắn trực tiếp với từng địa điểm đào tạo. Mỗi cơ sở tự quản lý sinh viên, giảng viên, phòng học, lớp học phần, lịch học và đăng ký học phần của mình. Vì vậy, phân mảnh ngang theo cơ sở là phương án phù hợp với mô hình tổ chức của hệ thống.

Với cách phân mảnh này, dữ liệu của từng bảng được chia theo mã cơ sở. Ví dụ, sinh viên Hòa Lạc được lưu tại mảnh Hòa Lạc, lớp học phần do giảng viên TP. Hồ Chí Minh phụ trách được lưu tại mảnh TP. Hồ Chí Minh. Nhờ đó, các thao tác thường xuyên như tra cứu sinh viên, mở lớp, xếp lịch, đăng ký và hủy đăng ký học phần có thể xử lý cục bộ tại máy trạm, giúp giảm tải cho Server Trung tâm và giảm dữ liệu truyền qua mạng.

Các dữ liệu dùng chung như học phần, học kỳ, ca học, chương trình đào tạo và môn tiên quyết không phân mảnh theo cơ sở mà được nhân bản xuống các máy trạm để phục vụ tra cứu. So với phân mảnh theo khoa, phân mảnh theo cơ sở hợp lý hơn vì các nghiệp vụ phòng học, lớp học phần, lịch học và đăng ký phụ thuộc trực tiếp vào địa điểm đào tạo.

1. Thiết kế phân mảnh ngang

- Vị trí 1: Server chi nhánh, nơi đặt máy trạm tại trụ sở đào tạo Hòa Lạc - Hà Nội (HQHL).
- Vị trí 2: Server chi nhánh, nơi đặt máy trạm tại trụ sở đào tạo TP. Hồ Chí Minh (HQHCM).

Mảnh	**Trụ sở**	**Cơ sở dữ liệu**	**Phân mảnh ngang nguyên thủy**	**Phân mảnh ngang dẫn xuất**
1	Hòa Lạc	Quản lý tín chỉ Hòa Lạc	- Bảng phân mảnh: headquarter - Điều kiện phân tán: Headquarter_HL = σ ID_headquarter = 'HQHL' (headquarter)	- Bảng phân mảnh: department - Điều kiện dẫn xuất: Department_HL = department ⋈ Headquarter_HL - Bảng phân mảnh: room - Điều kiện dẫn xuất: Room_HL = room ⋈ Headquarter_HL - Bảng phân mảnh: student - Điều kiện dẫn xuất: Student_HL = student ⋈ Department_HL - Bảng phân mảnh: teacher - Điều kiện dẫn xuất: Teacher_HL = teacher ⋈ Department_HL - Bảng phân mảnh: class

				- Điều kiện dẫn xuất: Class_HL = class ⋈ Teacher_HL - Bảng phân mảnh: session - Điều kiện dẫn xuất: Session_HL = session ⋈ Class_HL - Bảng phân mảnh: registration - Điều kiện dẫn xuất: Registration_HL = registration ⋈ Class_HL
2	TP. Hồ Chí Minh	Quản lý tín chỉ TP. Hồ Chí Minh	- Bảng phân mảnh: headquarter - Điều kiện phân tán: Headquarter_HCM = σ ID_headquarter = 'HQHCM' (headquarter)	- Bảng phân mảnh: department - Điều kiện dẫn xuất: Department_HCM = department ⋈ Headquarter_HCM - Bảng phân mảnh: room - Điều kiện dẫn xuất: Room_HCM = room ⋈ Headquarter_HCM - Bảng phân mảnh: student - Điều kiện dẫn xuất: Student_HCM = student ⋈ Department_HCM

				- Bảng phân mảnh: teacher - Điều kiện dẫn xuất: Teacher_HCM = teacher ⋈ Department_HCM - Bảng phân mảnh: class - Điều kiện dẫn xuất: Class_HCM = class ⋈ Teacher_HCM - Bảng phân mảnh: session - Điều kiện dẫn xuất: Session_HCM = session ⋈ Class_HCM - Bảng phân mảnh: registration - Điều kiện dẫn xuất: Registration_HCM = registration ⋈ Class_HCM

2. Thiết kế định vị và vẽ sơ đồ định vị

2.1. Thiết kế định vị

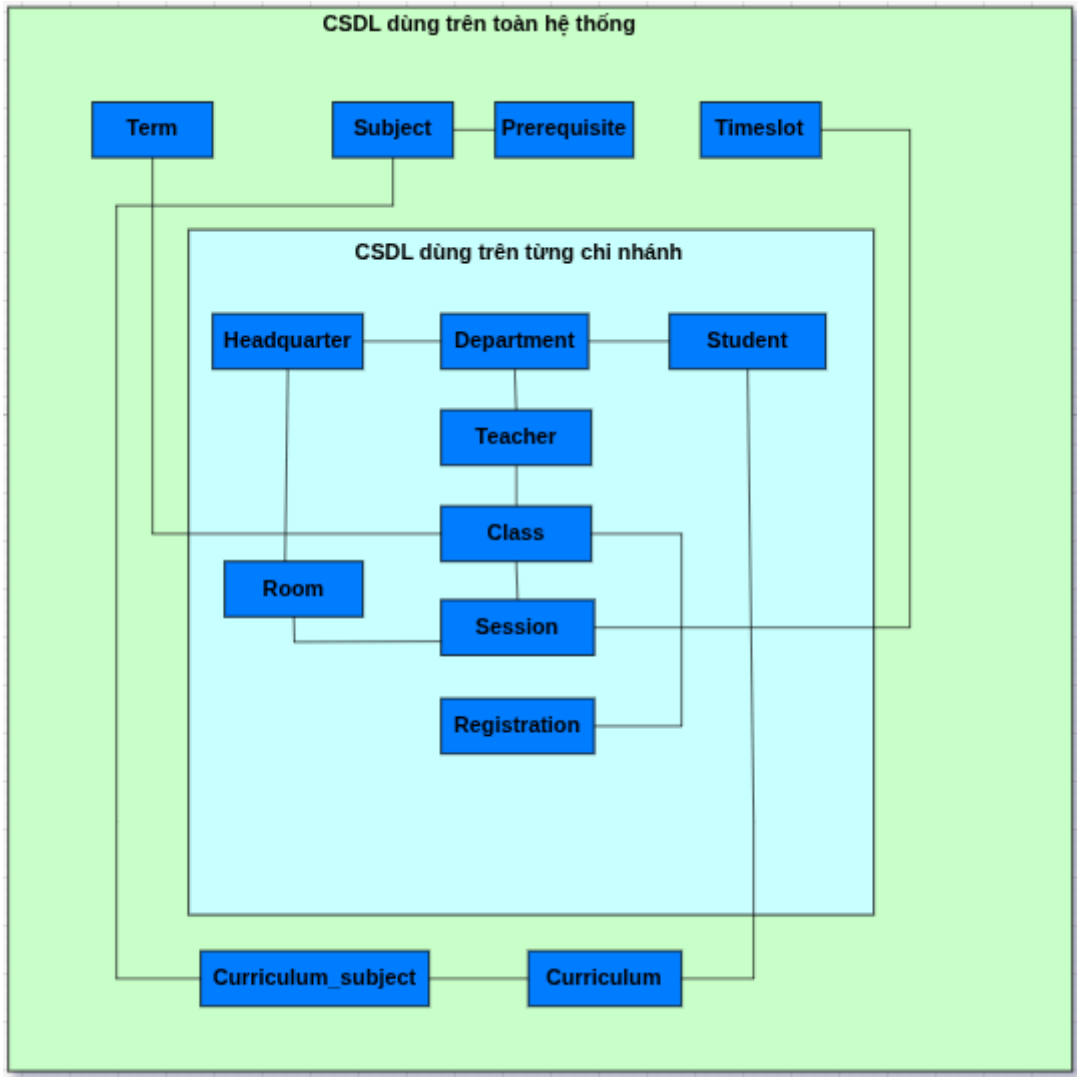
Phân mảnh ngang nguyên thủy và phân mảnh ngang dẫn xuất được sử dụng để chia quan hệ tổng thể thành 2 mảnh đặt tại 2 chi nhánh. Ngoài ra, server tổng đặt tại trụ sở chính Hà Đông - Hà Nội lưu trữ dữ liệu toàn cục của hệ thống.

- Server chính đặt tại Hà Đông - Hà Nội (HQHD) chứa thông tin của toàn bộ hệ thống, bao gồm: các danh mục dùng chung Curriculum (Chương trình đào tạo), Curriculum_Subject (Môn học trong chương trình đào tạo), Subject (Môn học), Term (Học kỳ), Prerequisite (Môn học tiên quyết), Timeslot (Ca học); cùng các dữ liệu nghiệp vụ Headquarter (Trụ sở), Room (Phòng học), Department (Khoa), Teacher (Giảng viên), Student (Sinh viên), Class (Lớp học phần), Session (Buổi học), Registration (Đăng ký học phần) của toàn bộ hệ thống.

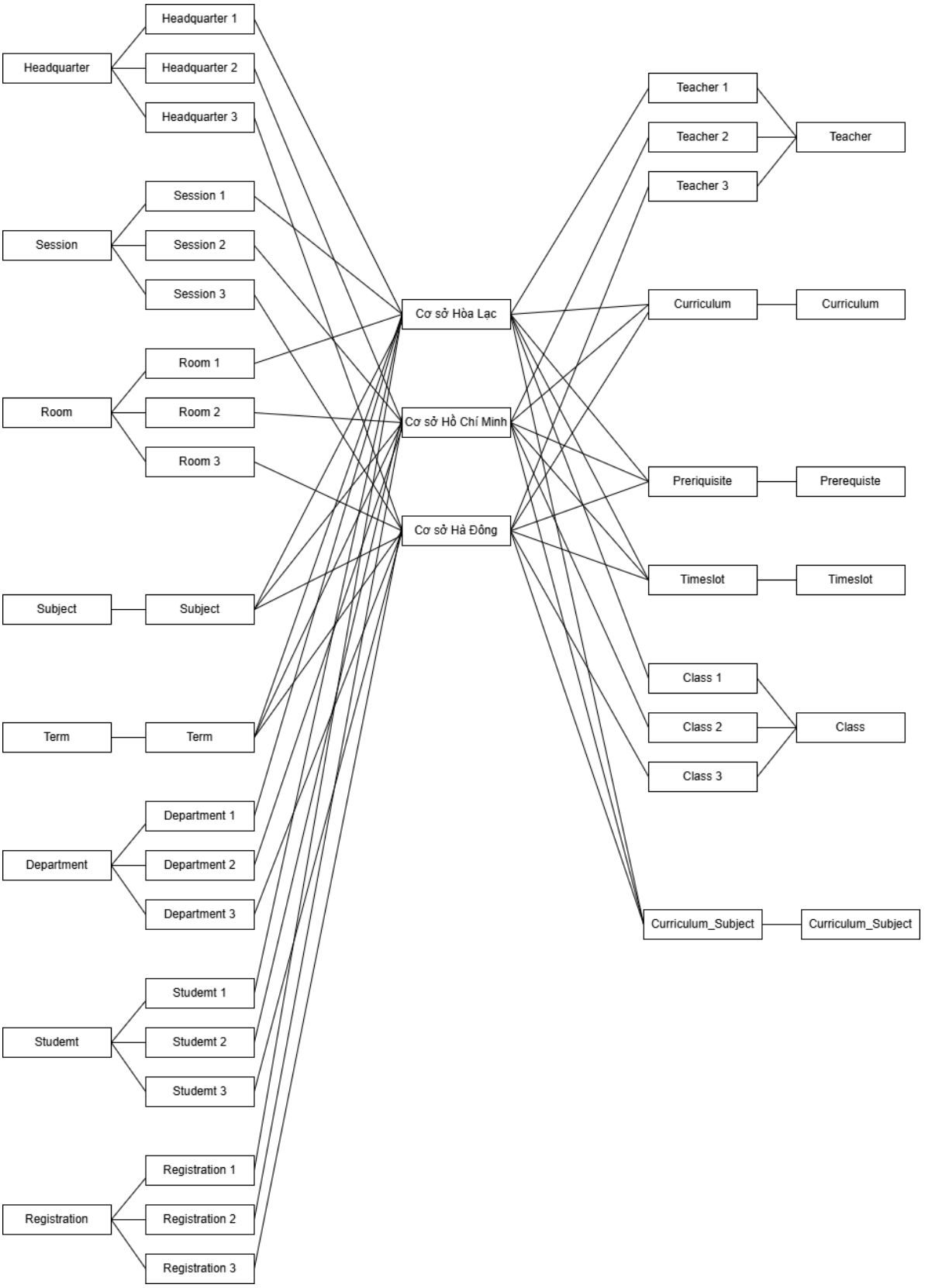
- Trụ sở đào tạo 1: Máy trạm 1 đặt tại Hòa Lạc - Hà Nội (HQHL) chứa thông tin của chi nhánh đó, bao gồm: các danh mục dùng chung được nhân bản từ server chính như Curriculum (Chương trình đào tạo), Curriculum_Subject (Môn học trong chương trình đào tạo), Subject (Môn học), Term (Học kỳ), Prerequisite (Môn học tiên quyết), Timeslot (Ca học); cùng các dữ liệu nghiệp vụ Headquarter (Trụ sở), Room (Phòng học), Department (Khoa), Teacher (Giảng viên), Student (Sinh viên), Class (Lớp học phần), Session (Buổi học), Registration (Đăng ký học phần) của chi nhánh đó.

- Trụ sở đào tạo 2: Máy trạm 2 đặt tại TP. Hồ Chí Minh (HQHCM) chứa thông tin của chi nhánh đó, bao gồm: các danh mục dùng chung được nhân bản từ server chính như Curriculum (Chương trình đào tạo), Curriculum_Subject (Môn học trong chương trình đào tạo), Subject (Môn học), Term (Học kỳ), Prerequisite (Môn học tiên quyết), Timeslot (Ca học); cùng các dữ liệu nghiệp vụ Headquarter (Trụ sở), Room (Phòng học), Department (Khoa), Teacher (Giảng viên), Student (Sinh viên), Class (Lớp học phần), Session (Buổi học), Registration (Đăng ký học phần) của chi nhánh đó.

2.2. Sơ đồ định vị



3. Lược đồ ánh xạ



IV. Thiết kế lược đồ nhân bản, đồng bộ hóa

Thông tin tại máy chủ trung tâm Hà Đông - Hà Nội:

Các bảng danh mục dùng chung bao gồm: Curriculum, Curriculum_Subject, Subject, Term, Prerequisite, Timeslot. Các bảng này được quản lý tập trung tại máy chủ trung tâm và được nhân bản đầy đủ xuống các máy trạm tại Hòa Lạc và TP. Hồ Chí Minh thông qua Transactional Replication.

Khi máy chủ trung tâm có sự thay đổi đối với dữ liệu danh mục dùng chung, ví dụ thêm học phần mới, cập nhật học kỳ, thay đổi ca học hoặc cập nhật điều kiện môn học tiên quyết, dữ liệu sẽ được đồng bộ xuống toàn bộ máy trạm. Tại máy trạm, các bảng danh mục này chủ yếu được sử dụng để đọc, tra cứu và phục vụ quá trình mở lớp, xếp lịch, đăng ký học phần.

Đối với các bảng dữ liệu nghiệp vụ phân mảnh như Headquarter, Department, Room, Teacher, Student, Class, Session, Registration, máy chủ trung tâm lưu trữ dữ liệu toàn cục của toàn hệ thống. Nếu tại máy chủ có sự cập nhật dữ liệu thuộc một cơ sở cụ thể, hệ thống sẽ đồng bộ dữ liệu đó về đúng máy trạm tương ứng dựa trên mã cơ sở.

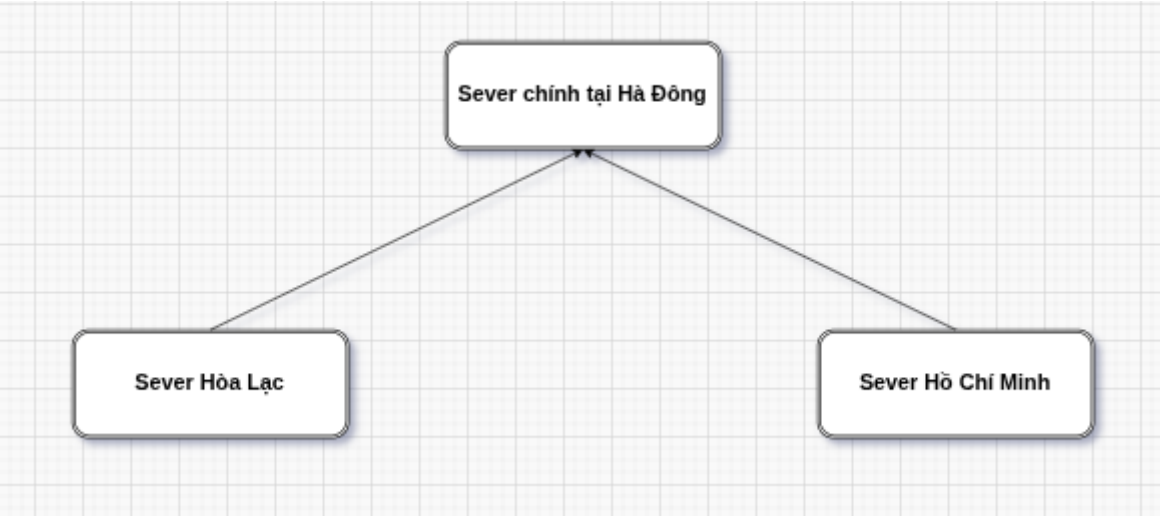
Thông tin tại máy trạm chi nhánh:

Mỗi máy trạm lưu trữ dữ liệu nghiệp vụ phát sinh tại cơ sở mình, bao gồm: sinh viên, giảng viên, phòng học, lớp học phần, buổi học và đăng ký học phần. Khi có thao tác thêm, sửa, xóa dữ liệu nghiệp vụ tại máy trạm, dữ liệu sẽ được đồng bộ ngược về máy chủ trung tâm Hà Đông thông qua Merge Replication.

Đối với các bảng Registration và Class, do liên quan trực tiếp đến nghiệp vụ đăng ký học phần và giới hạn sĩ số lớp, hệ thống cần kiểm soát đồng thời khi nhiều sinh viên cùng đăng ký một lớp học phần. Giao dịch đăng ký phải đảm bảo số lượng sinh viên đăng ký không vượt quá sĩ số tối đa của lớp. Việc kiểm soát này được thực hiện thông qua Stored Procedure kết hợp Transaction và cơ chế khóa dữ liệu.

Đối với đăng ký học phần cục bộ, dữ liệu được xử lý trực tiếp tại máy trạm của cơ sở tương ứng và sau đó đồng bộ về máy chủ trung tâm. Đối với đăng ký chéo cơ sở, máy trạm gửi yêu cầu lên máy chủ trung tâm Hà Đông để xử lý trên CSDL toàn cục, nhằm đảm bảo toàn vẹn dữ liệu giữa sinh viên và lớp học phần thuộc hai cơ sở khác nhau.

Cơ chế đồng bộ hóa giúp máy chủ trung tâm luôn nắm giữ dữ liệu tổng hợp của toàn bộ hệ thống, phục vụ công tác quản lý, thống kê toàn trường và hỗ trợ truy vấn dữ liệu giữa các cơ sở.



PHẦN 4: CÀI ĐẶT

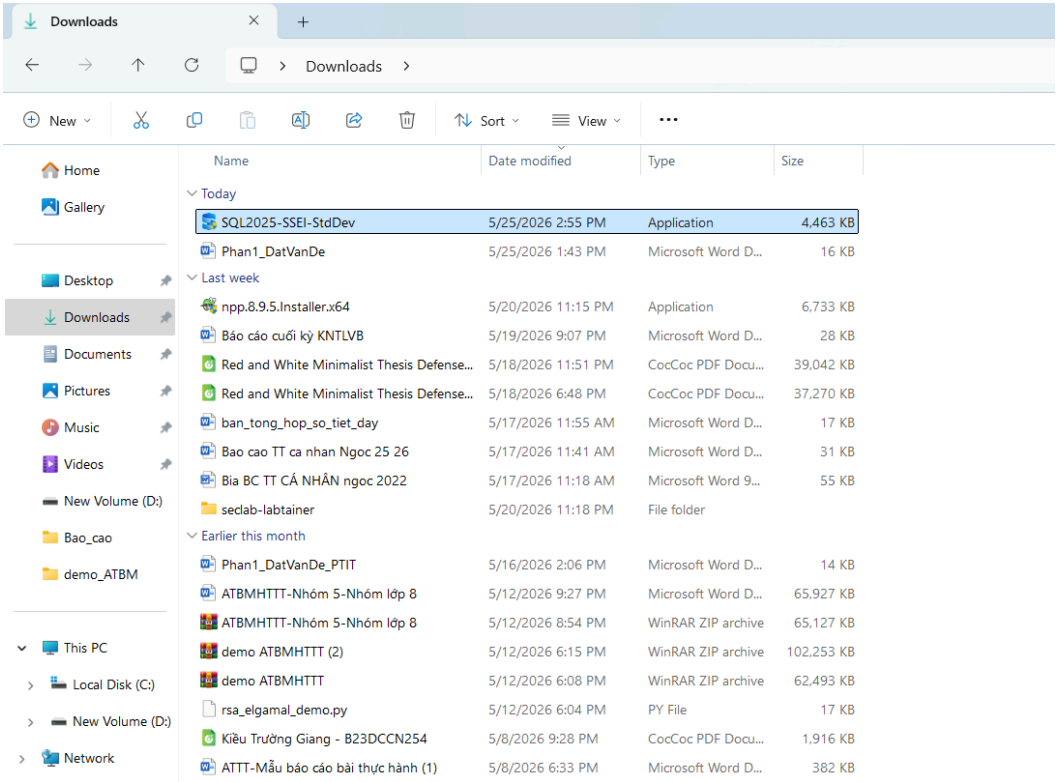
I.Cài đặt SQL Server 2025

1.Cài đặt SQL Server 2025 Express

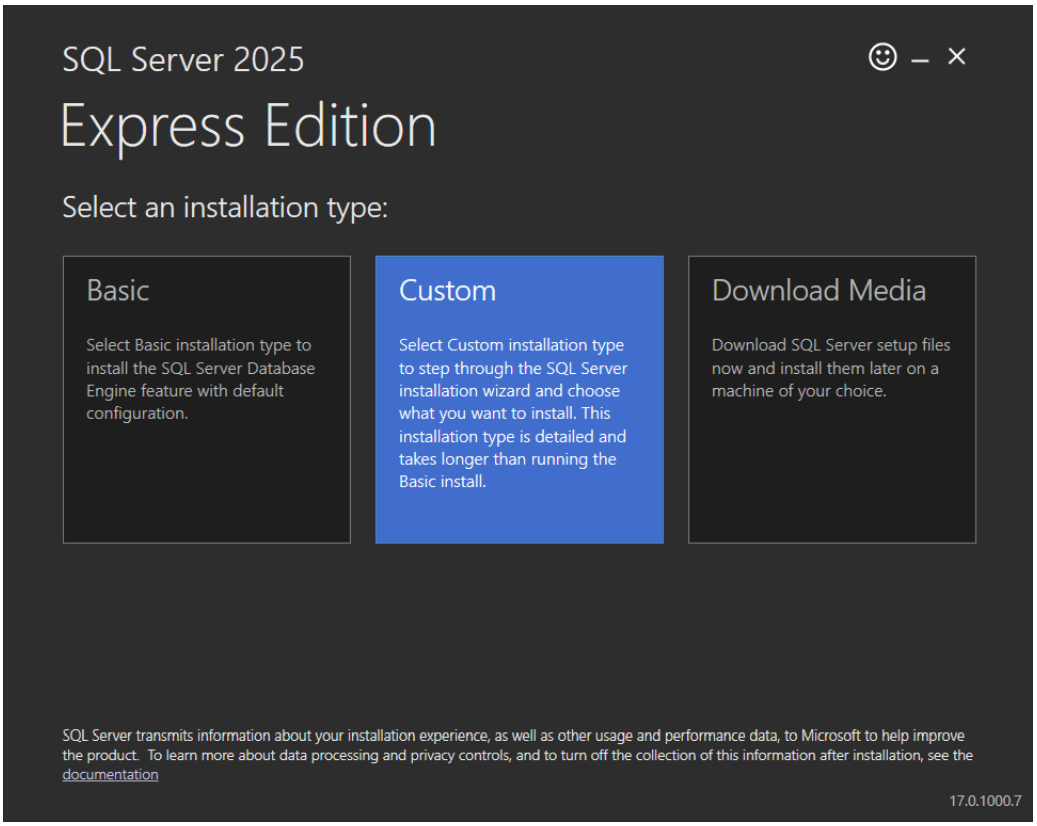
Tải SQL Server trên

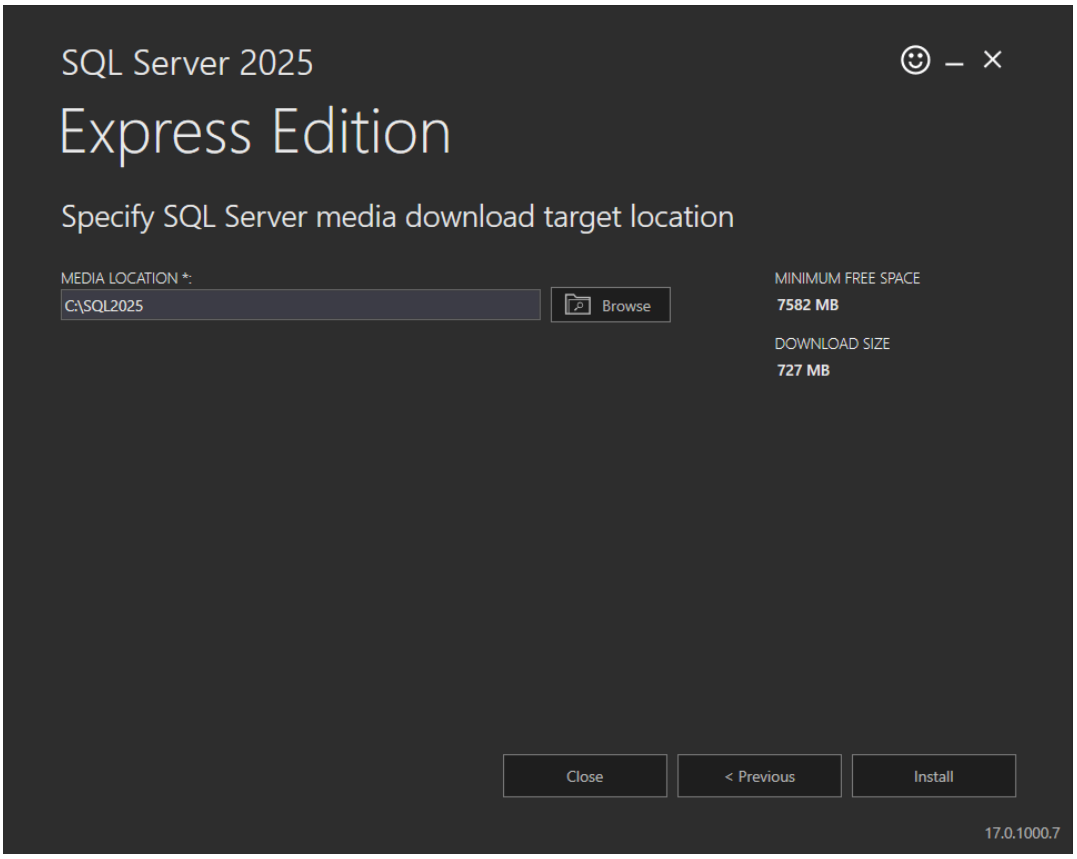
https://www.microsoft.com/en-us/sql-server/sql-server-downloads?rtc=1

Bước 1: Mở tệp vừa tải về

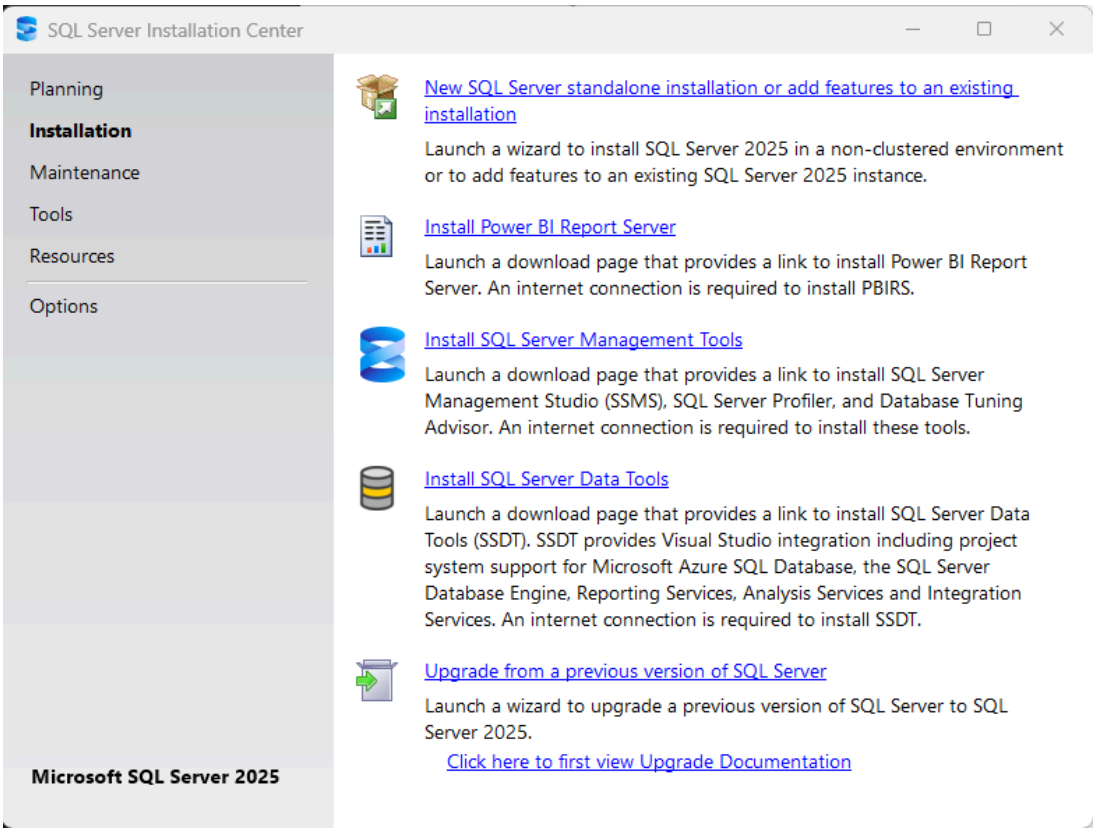


Bước 2: Chọn Custom sau đó chọn đường dẫn rồi ấn Install

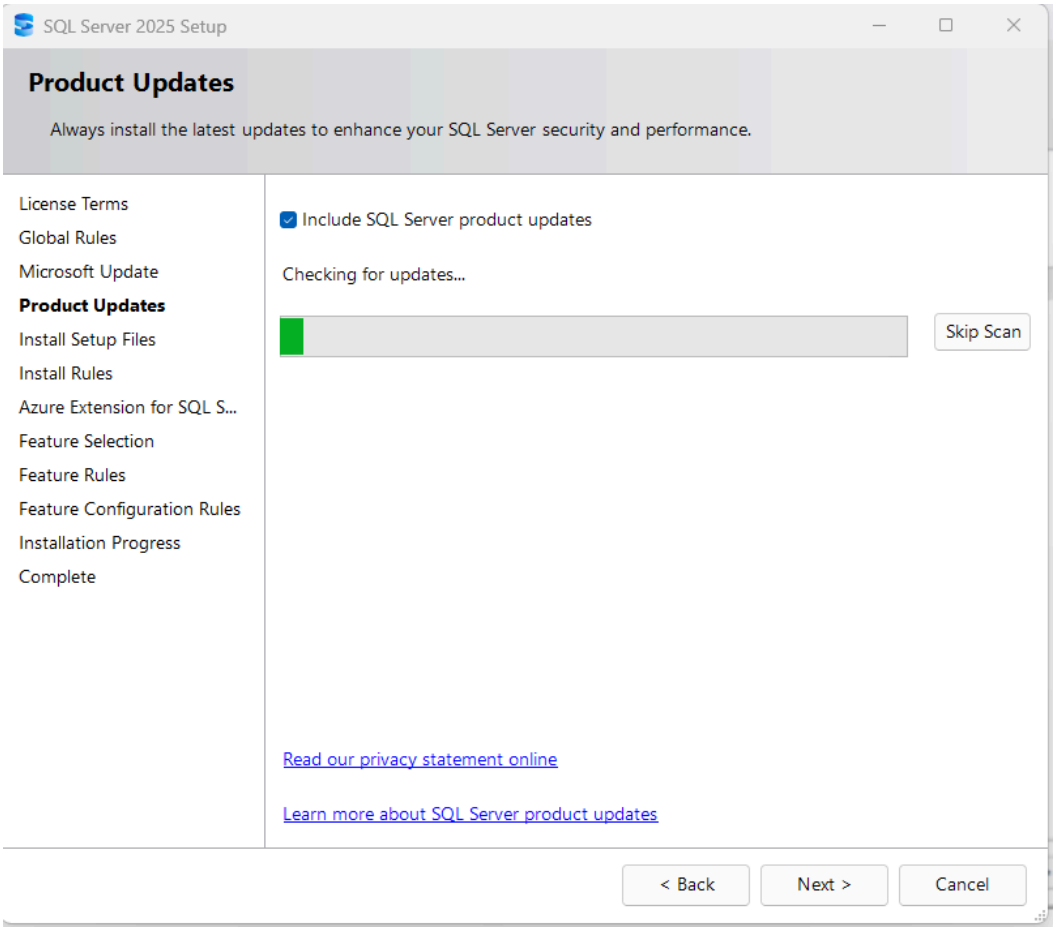




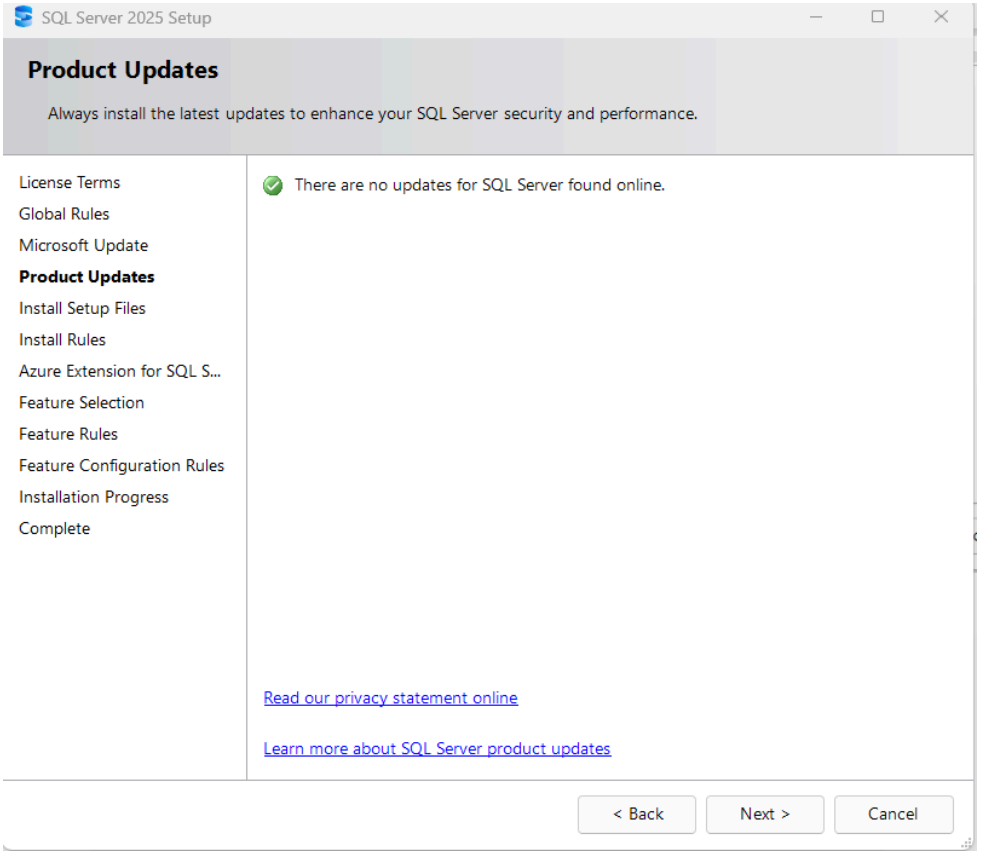
Bước 3: Giao diện cài đặt hiện lên, chọn Installation sau đó chọn tiếp New SQL server.



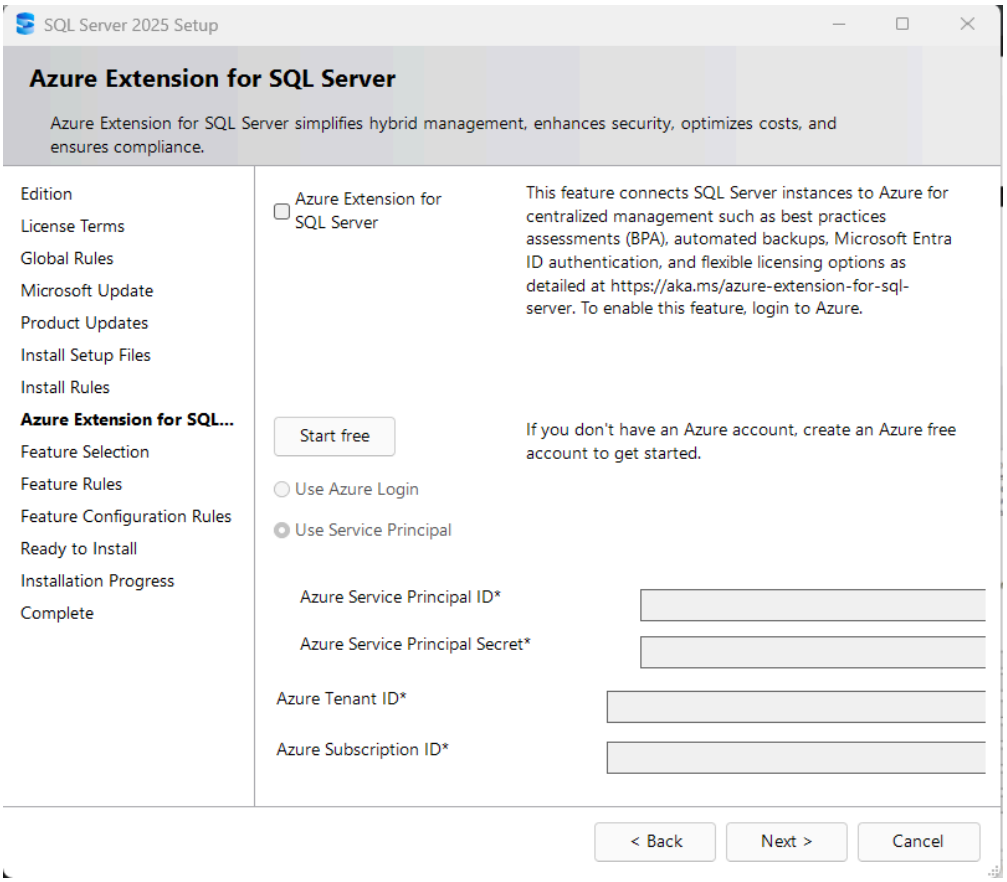
Bước 4: Cài đặt



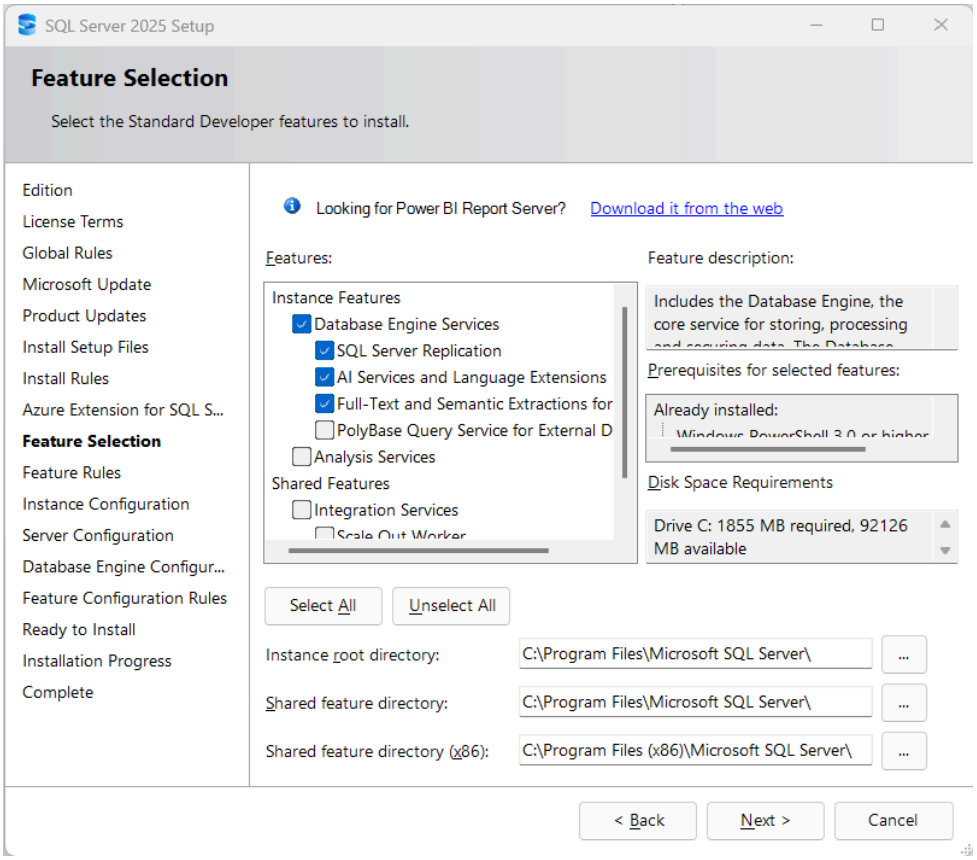
Bước 5: Chọn next



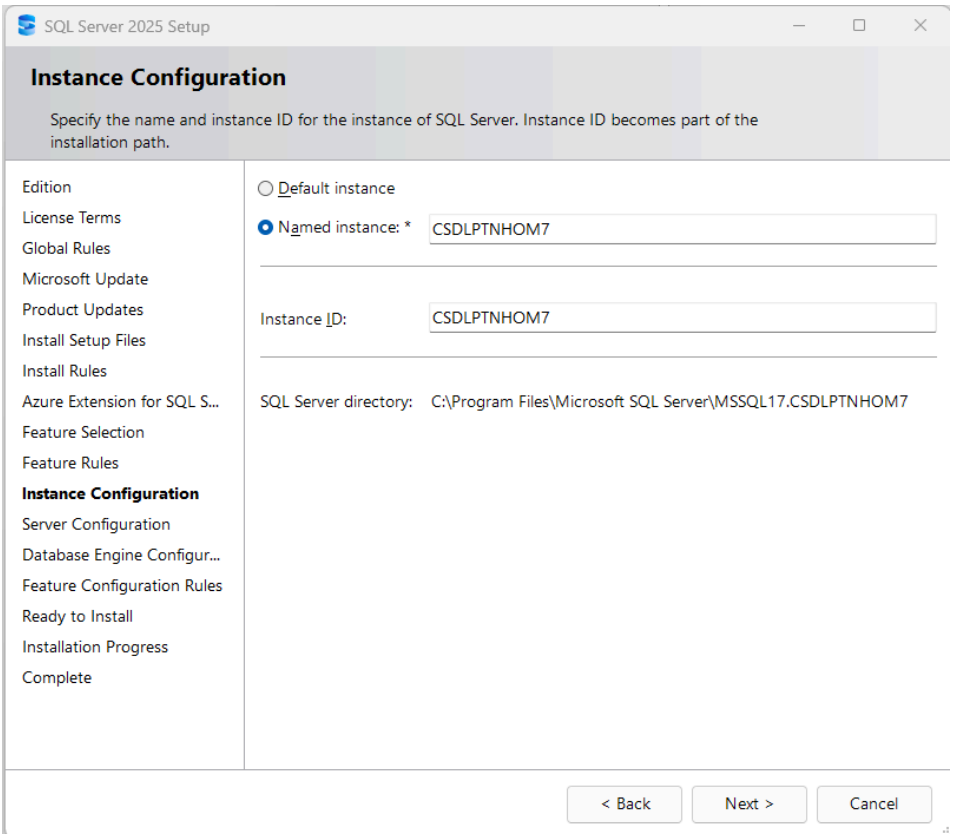
Bước 6: Bỏ chọn Azure Extension và bấm Next



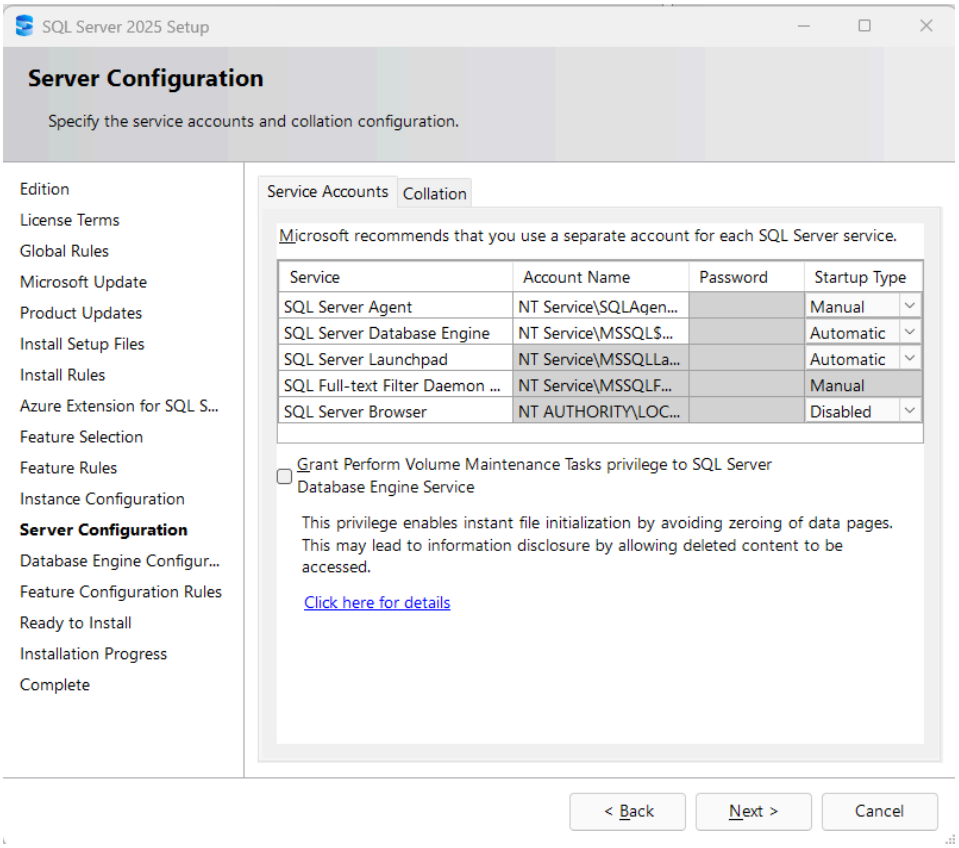
Bước 7: Chọn chức năng phù hợp với môn học ở mục Features sau đó ấn nút Next.



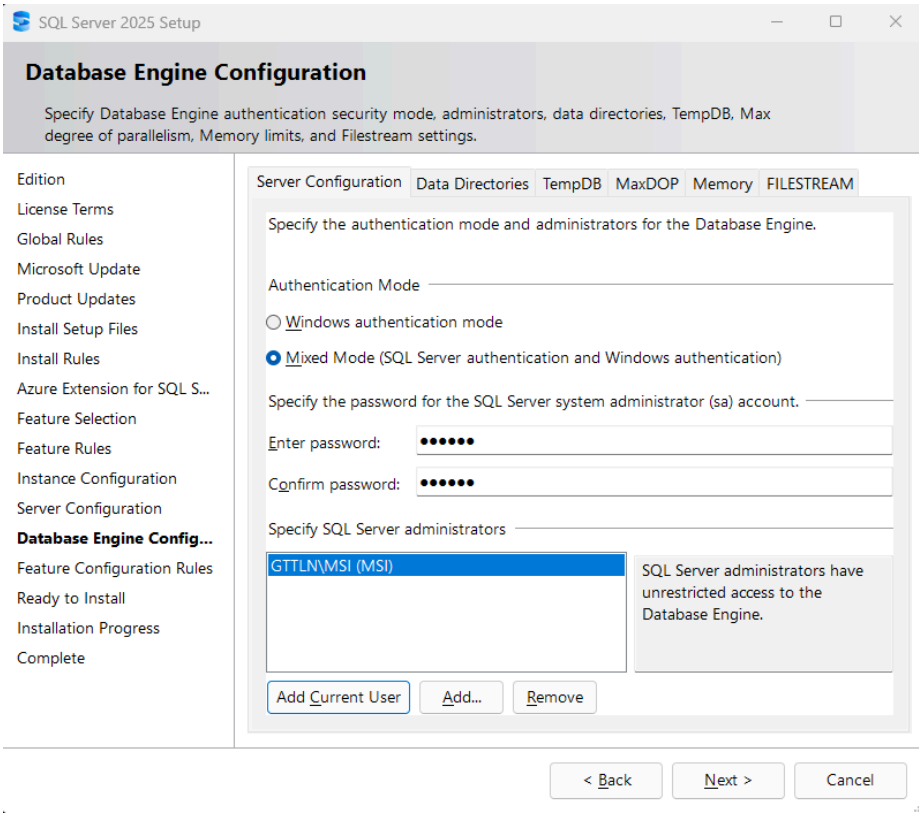
Bước 8: Ta có thể đặt tên cho Instance (tên có thể đặt tùy ý, không dấu, không khoảng trắng) sau đó nhấn Next.



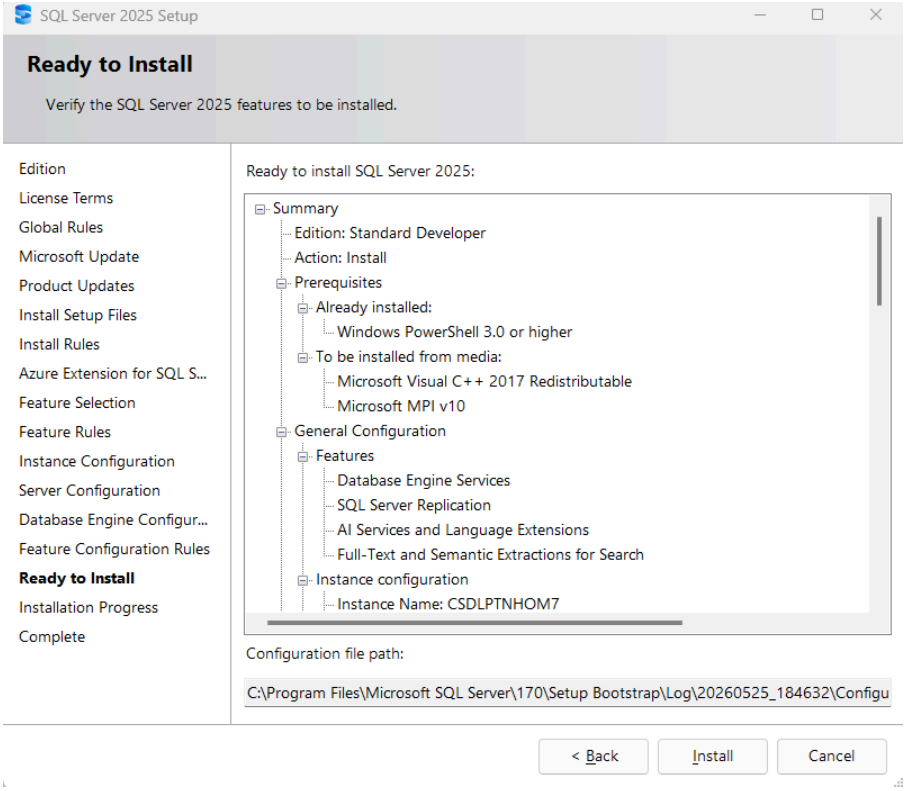
Bước 9: Nhấn Next



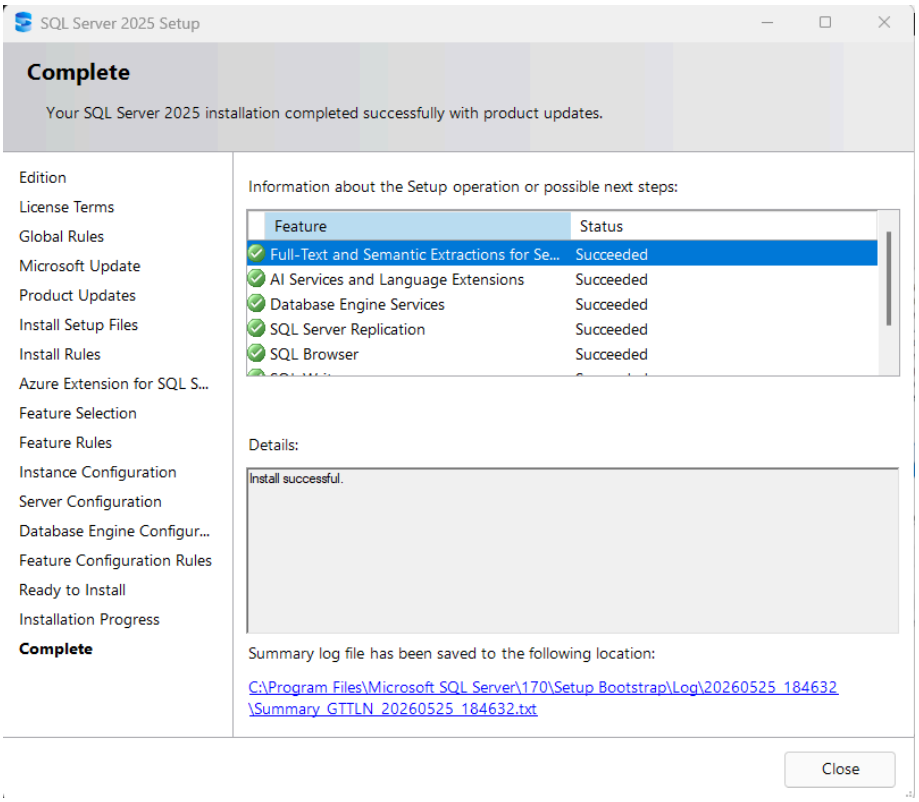
Bước 10: Ở mục này, ta bấm chọn chức năng Mix Mode. Đây là chức năng bảo mật cho cơ sở dữ liệu, nhập mật khẩu cho tài khoản là 123456 và cuối cùng nhấn nút Add current User để thêm tài khoản



Bước 11: Cuối cùng nhấn next cho đến mục Ready to Install và Ấn Install



Bước 12: Cài đặt thành công bấm Close

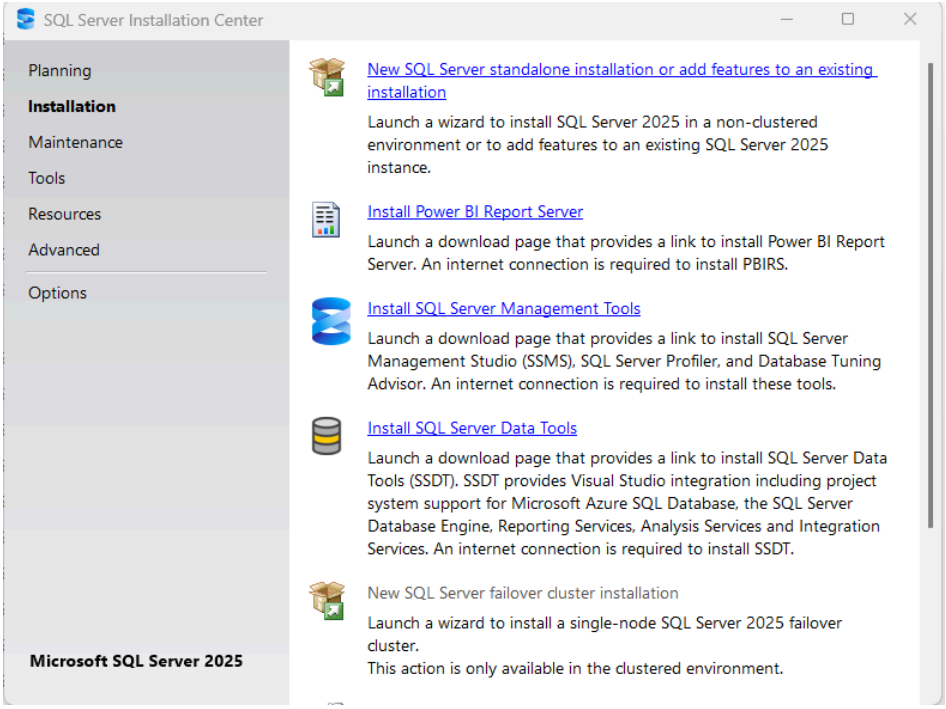


2.Cài đặt SQL Server Management Studio Tools (SSMS).

Giới thiệu : SSMS là một ứng dụng phần mềm thiết kế bởi Microsoft, ra mắt lần đầu năm 2005. Ứng dụng này cho phép lập trình viên cấu hình, quản lý và quản trị bộ máy cơ sở dữ liệu (database engine) SQL Server. SSMS phổ biến và được sử dụng rộng rãi trong cộng đồng lập trình viên và quản trị viên cơ sở dữ liệu bởi những ưu điểm

- Miễn phí
- Trải nghiệm người dùng tốt
- Nhiều lựa chọn add-in
- Dễ cài đặt

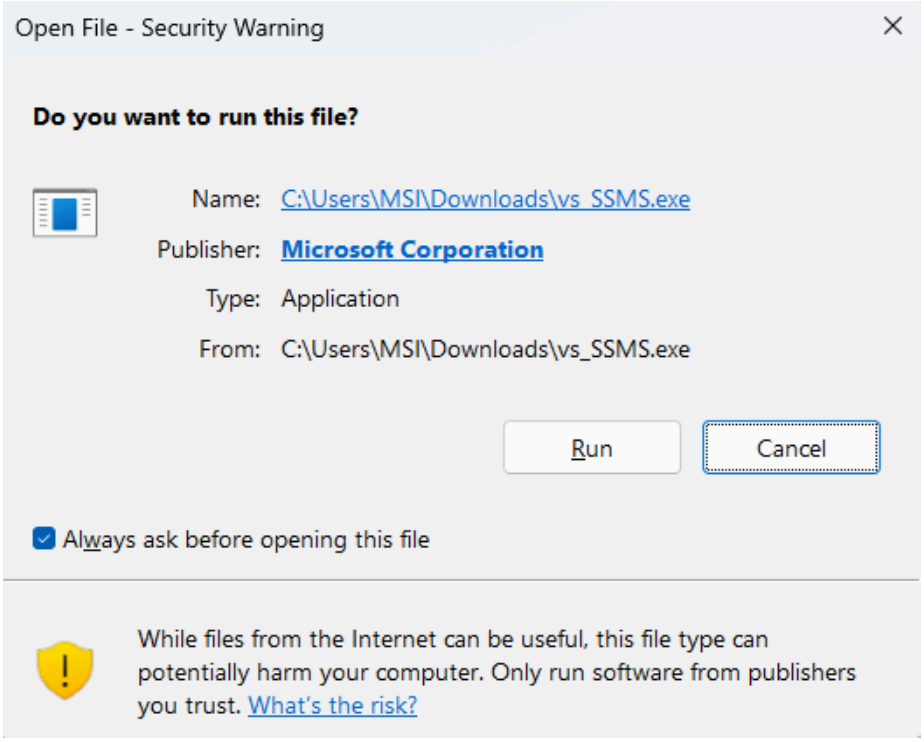
Bước 1: Click vào Install SQL Server Management Tools



Hoặc ấn vào đường link để cài đặt:

https://learn.microsoft.com/en-us/ssms/install/install

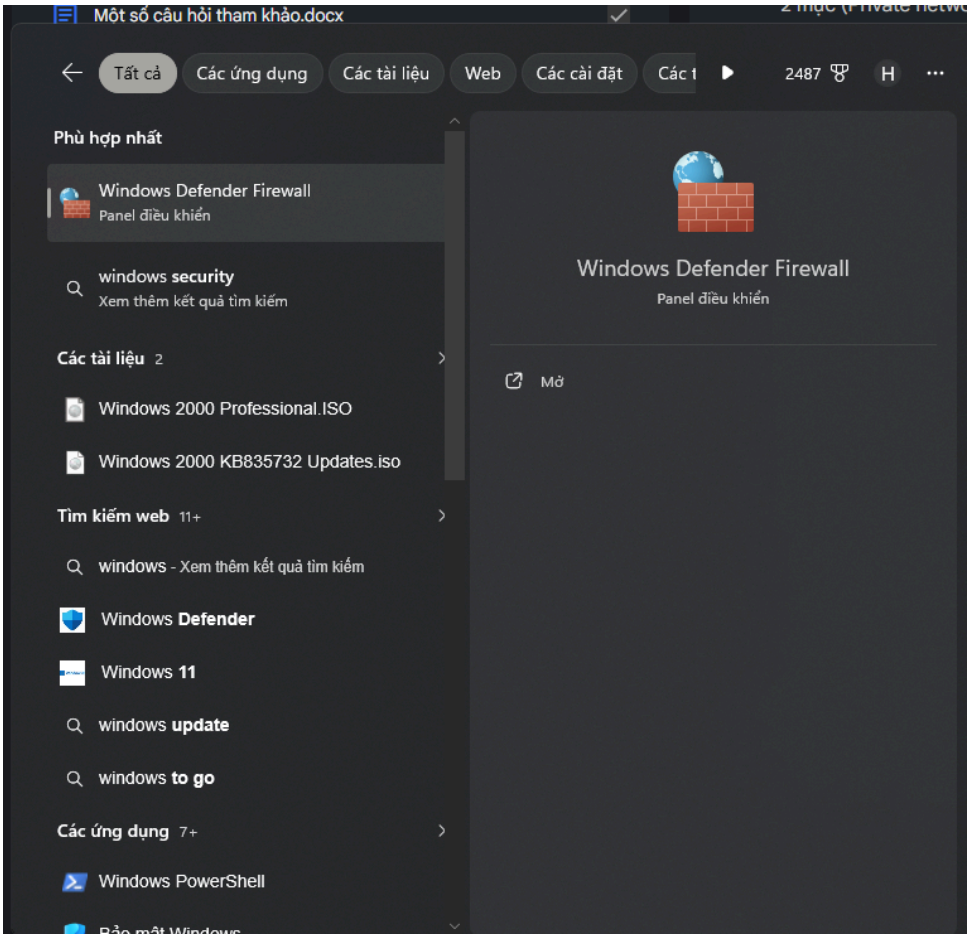
Bước 2: Sau khi download ta chạy file “vs_SSMS.exe” để cài đặt.



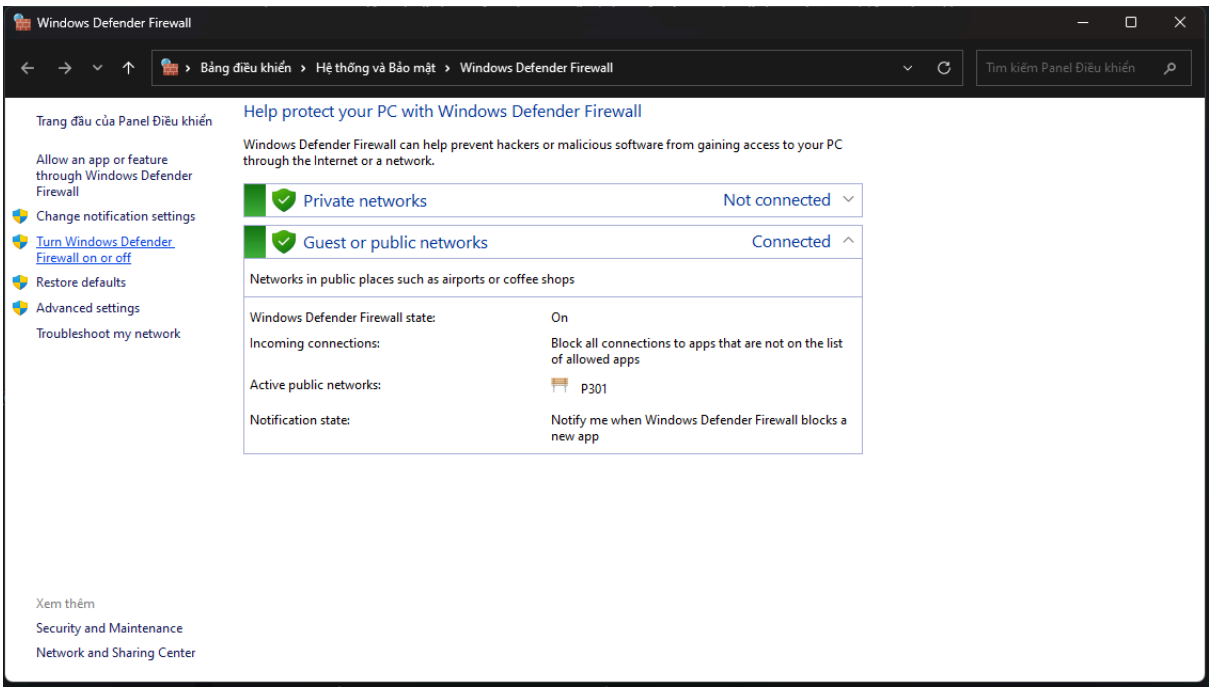
II. Cài đặt tường lửa (firewall) để mở cổng cho SQL

1. Tắt tường lửa

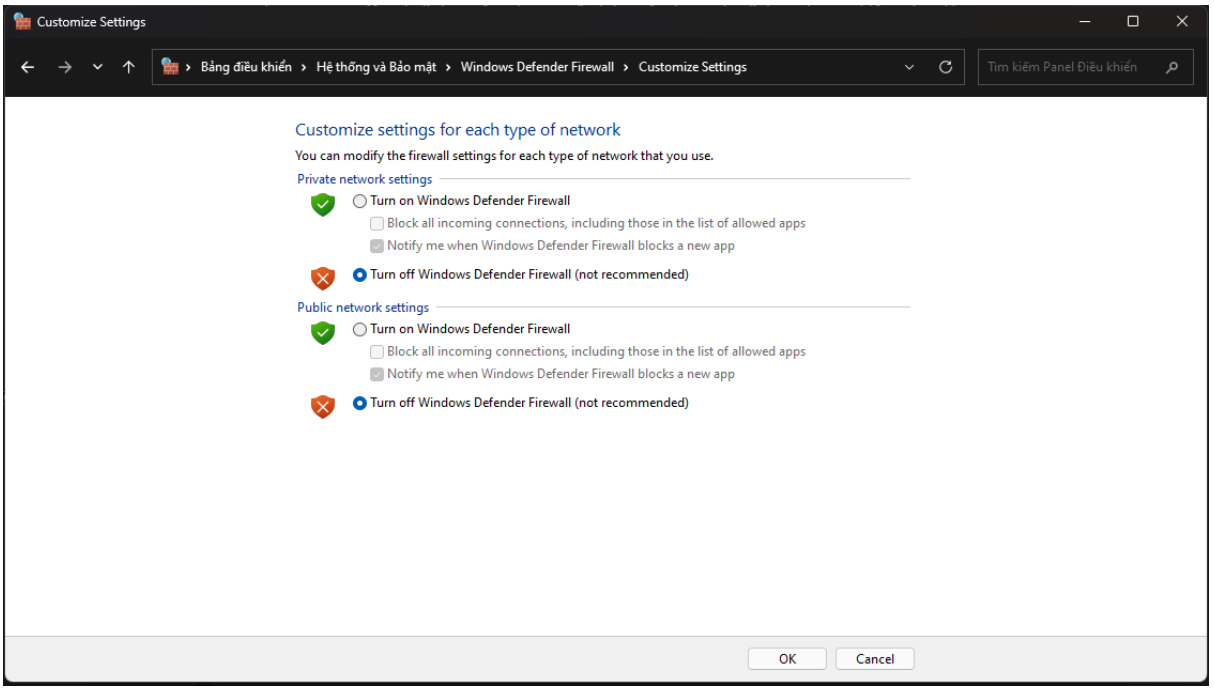
Bước 1: Tìm kiếm Firewall trong thanh công cụ tìm kiếm của Windows.



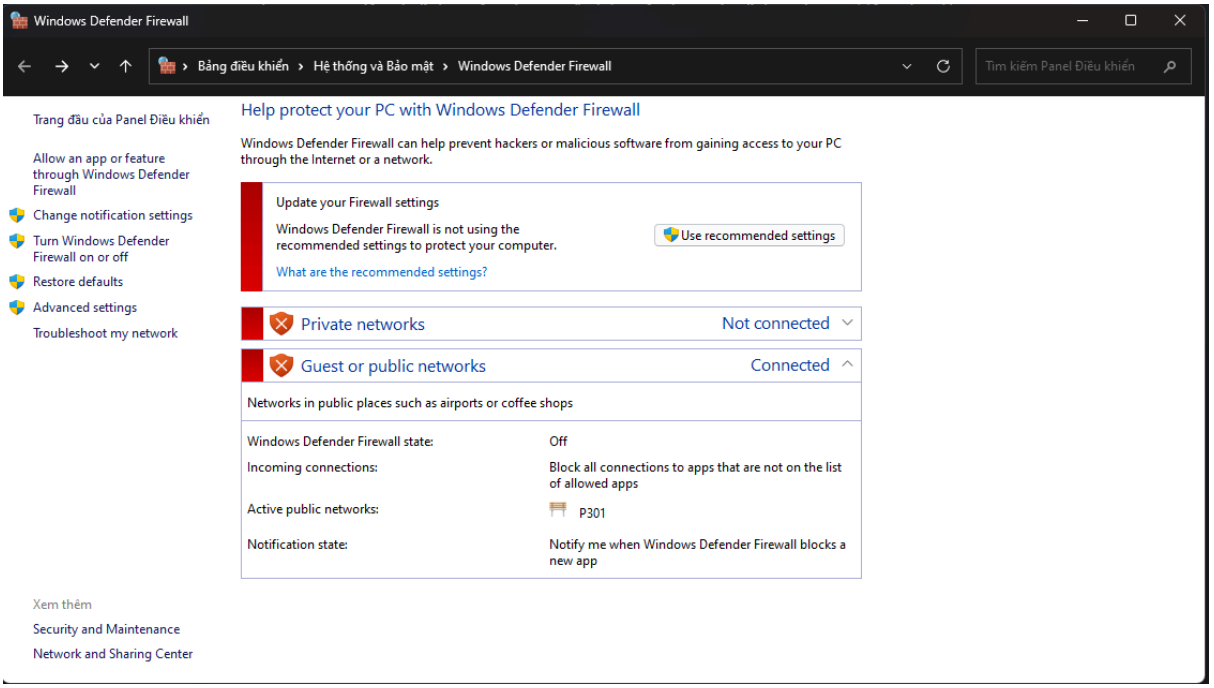
Bước 2: Chọn Turn Windows Defender Firewall on or off.



Bước 3: Thực hiện tắt tường lửa bằng cách chọn Turn off Windows Defender Firewall ở cả 2 mục (Private network settings và Public network settings) -> nhấn OK.



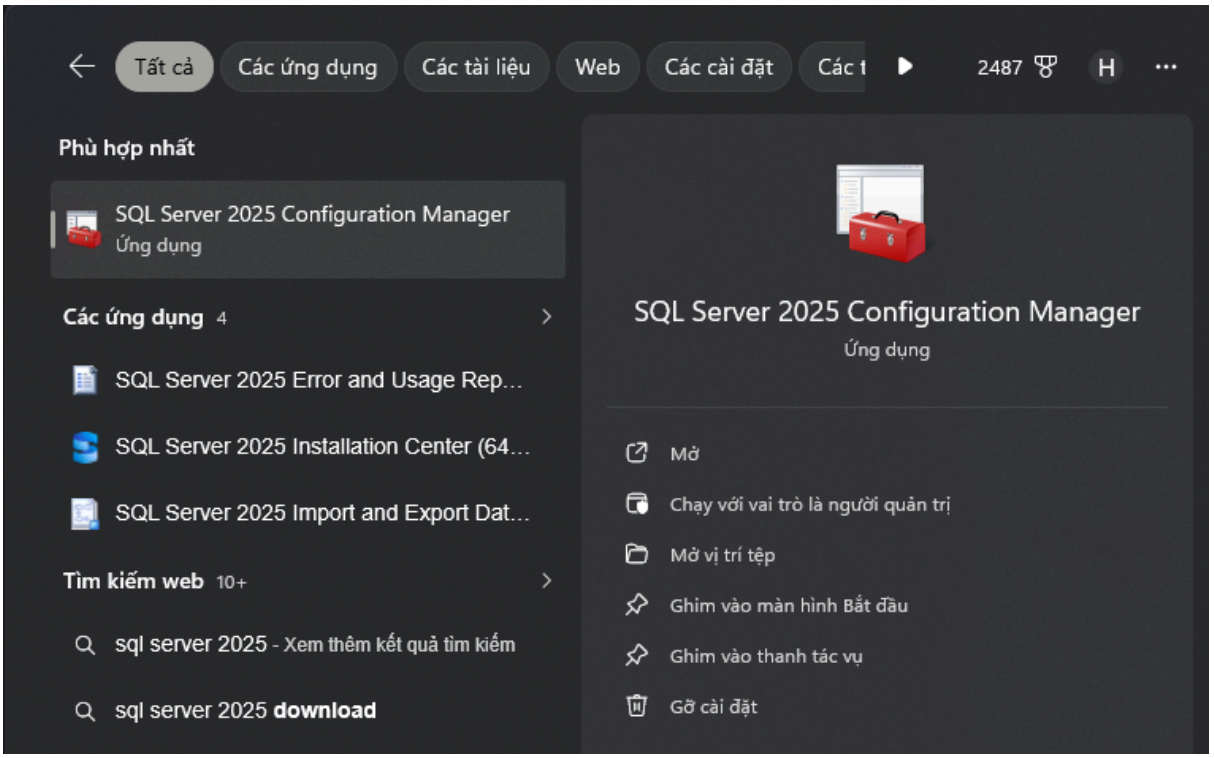
Bước 4: Tường lửa đã được tắt thành công. *(Chú ý: Nếu như không tắt được tường lửa thì có thể là do bạn đang chạy phần mềm diệt virus. Hãy xóa hoặc tắt trình duyệt diệt virus đi thì mới tắt được tường lửa).*



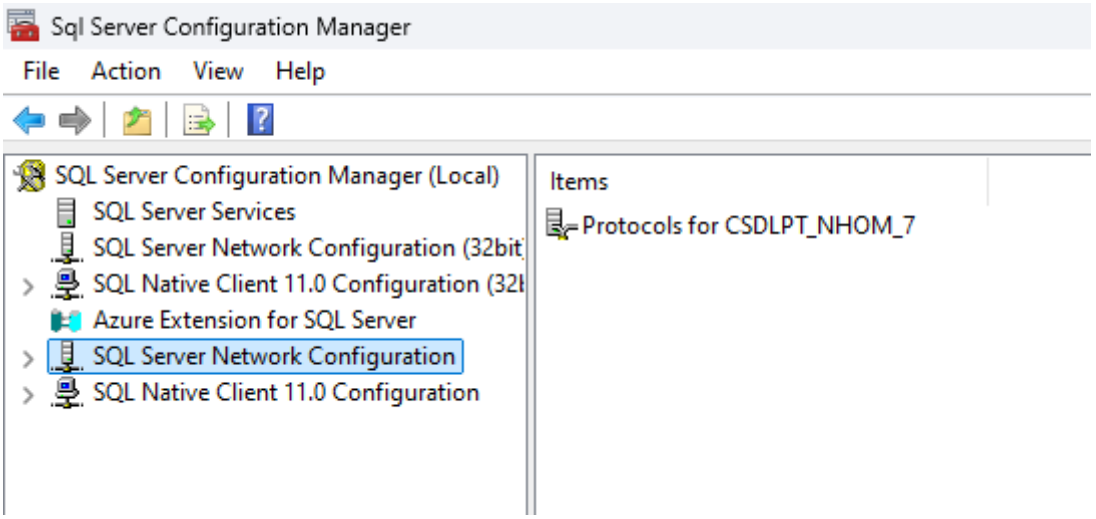
2. Mở port 1433

Mục đích: Theo mặc định, SQL Server sử dụng Cổng TCP 1433 để giao tiếp với các ứng dụng. Nếu ta đang chạy một ứng dụng phân tán yêu cầu kết nối các trạm với SQL Server trung tâm, ta phải bật cổng 1433 cho SQL Server để kết nối.

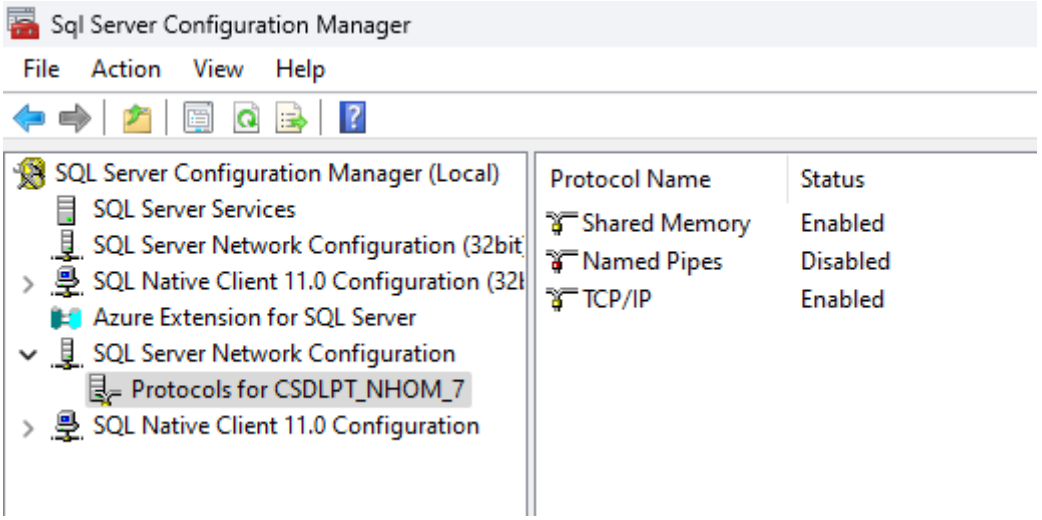
Bước 1: Trong thanh tìm kiếm của Windows, tìm và mở SQL Server 2025 Configuration Manager.



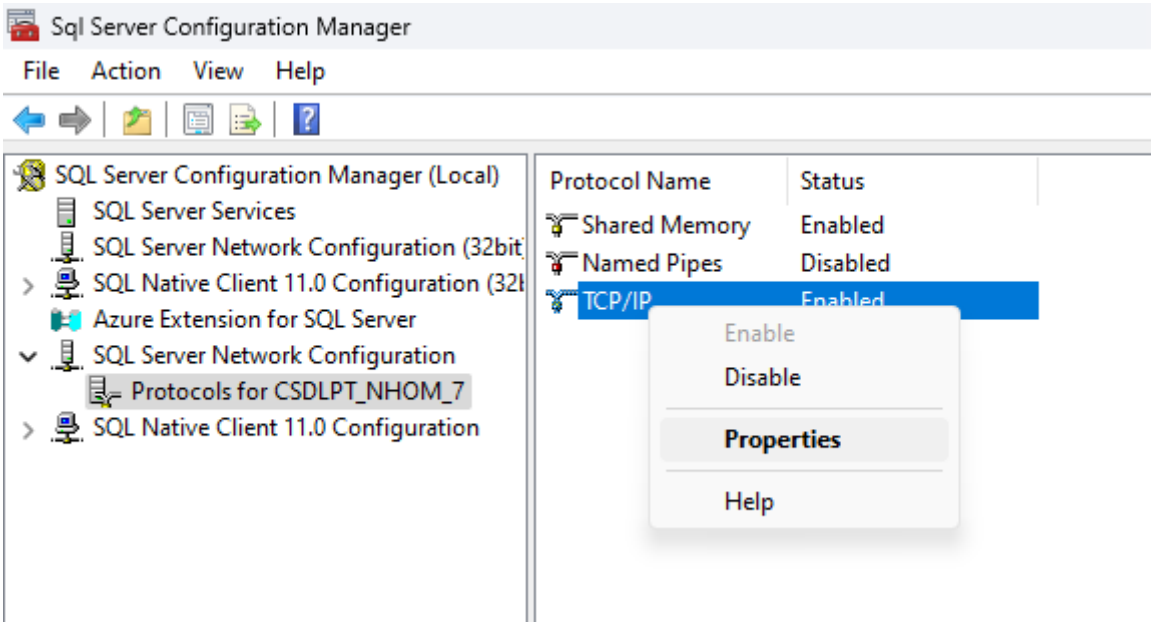
Bước 2: Chọn SQL Server Network Configuration -> Chọn Protocols for [Tên_Instance_Của_Bạn] (Ví dụ: Protocols for MSSQLSERVER).



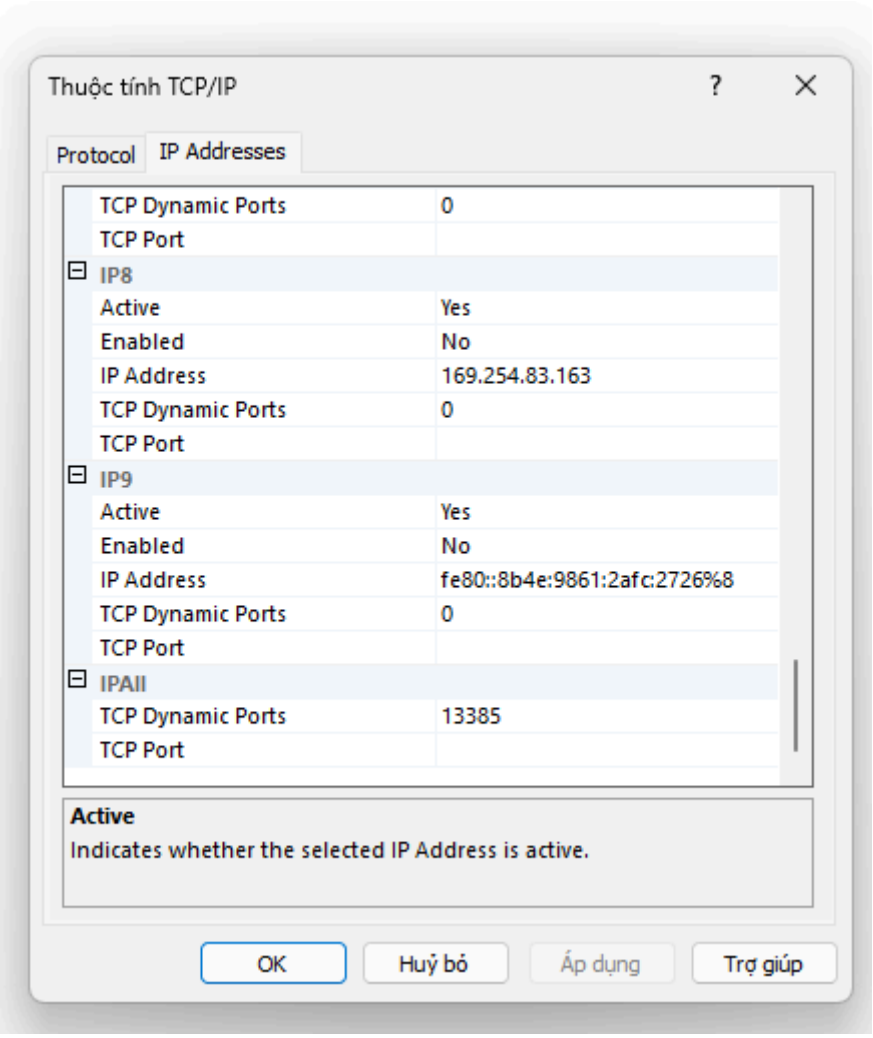
Bước 3: Tại cửa sổ bên phải, nhấn chuột phải vào TCP/IP, chọn Enable.

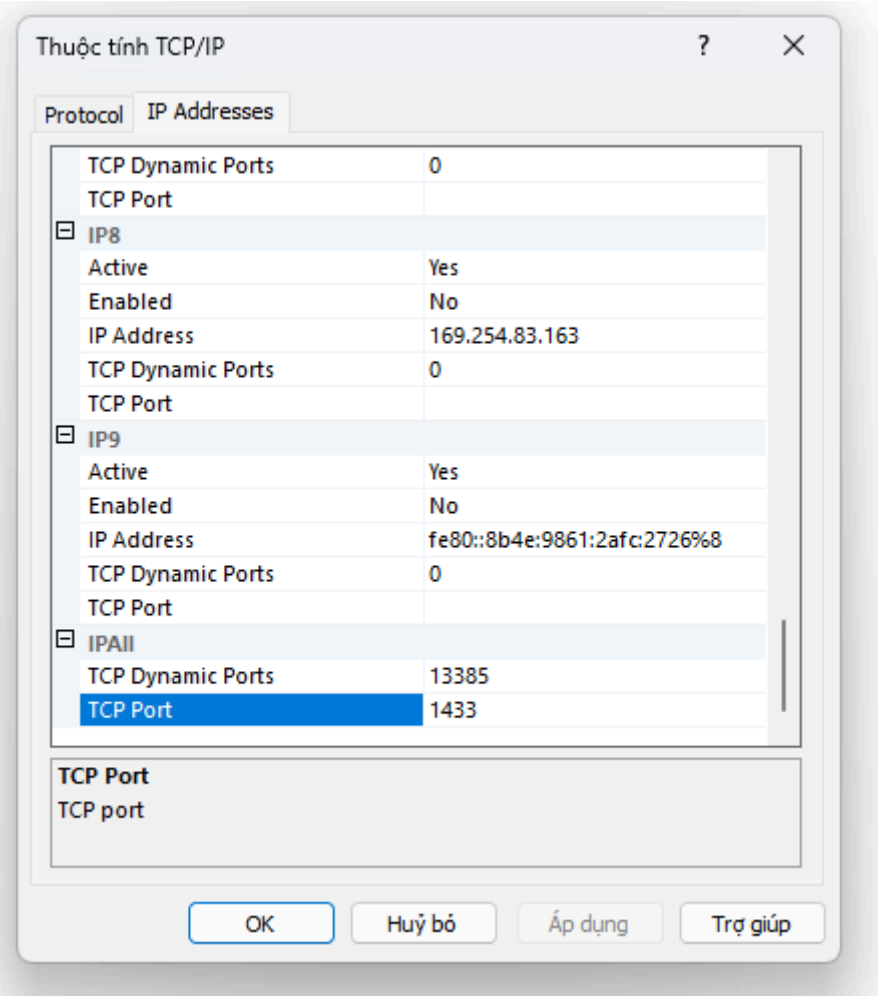


Bước 4: Tiếp tục click chuột phải vào TCP/IP -> Chọn Properties.

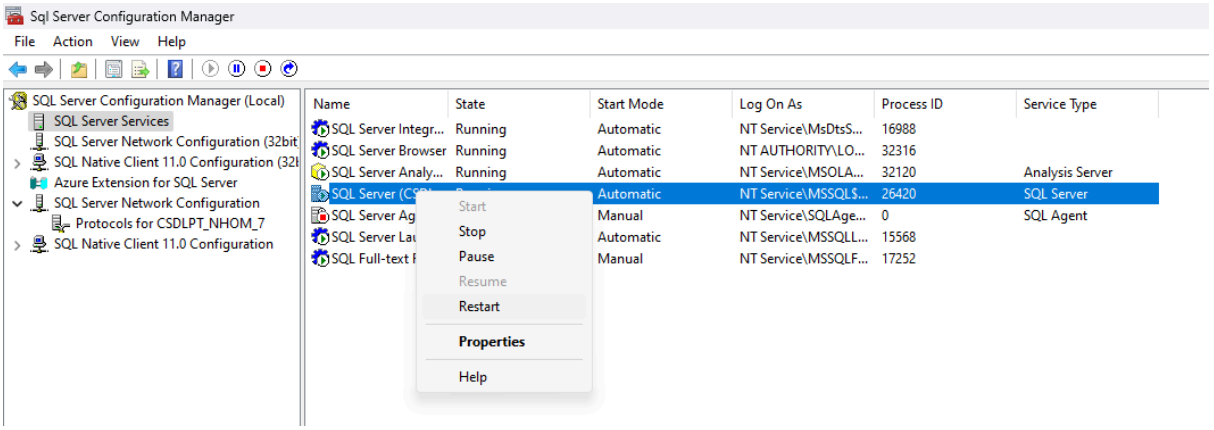


Bước 5: Trong bảng hiện ra, nhấn sang tab IP Addresses, lướt xuống dòng cuối cùng IPAll và thiết lập mục TCP Port là 1433 -> Nhấn Apply -> OK.


Thuộc tính TCP/IP
Protocol
IP Addresses
TCP Dynamic Ports 0
TCP Port
IP8
Active Yes
Enabled No
IP Address 169.254.83.163
TCP Dynamic Ports 0
TCP Port
IP9
Active Yes
Enabled No
IP Address fe80::8b4e:9861:2afc:2726%8
TCP Dynamic Ports 0
TCP Port
IPAll
TCP Dynamic Ports 13385
TCP Port
Active
Indicates whether the selected IP Address is active.
OK
Huỷ bỏ
Áp dụng
Trợ giúp



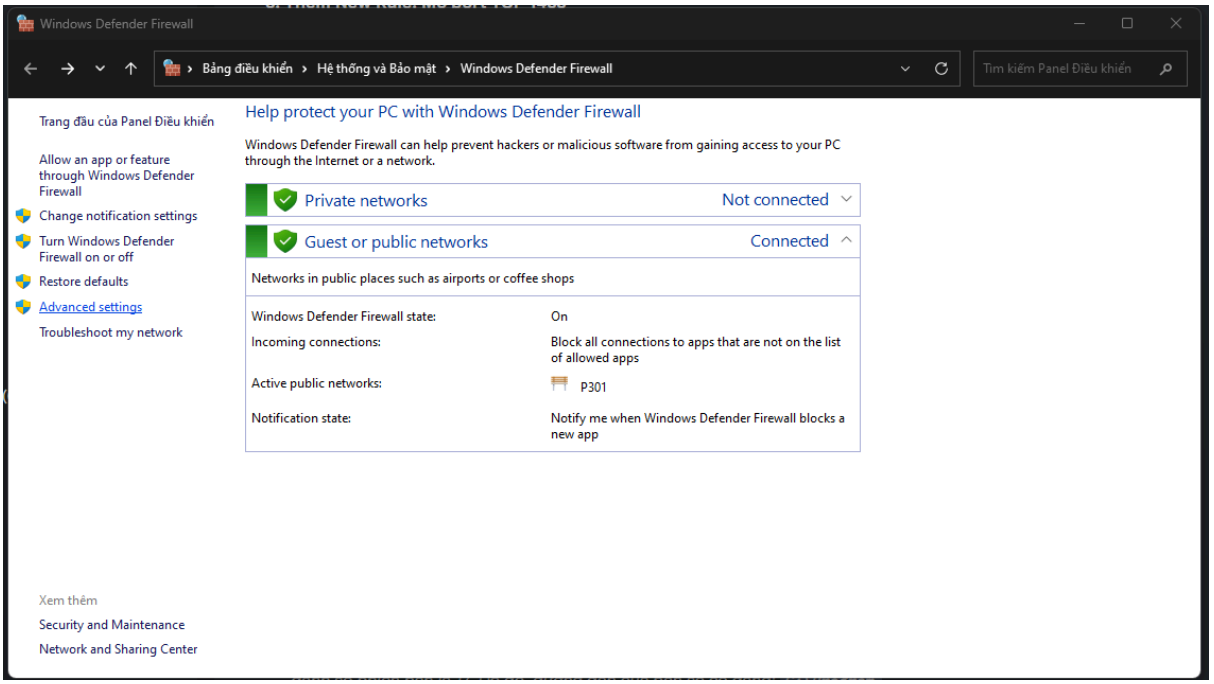
Bước 6: Trở lại mục SQL Server Services ở menu bên trái, click chuột phải vào SQL Server ([Tên_Instance_Của_Bạn]) và chọn Restart lại Server để áp dụng cấu hình.



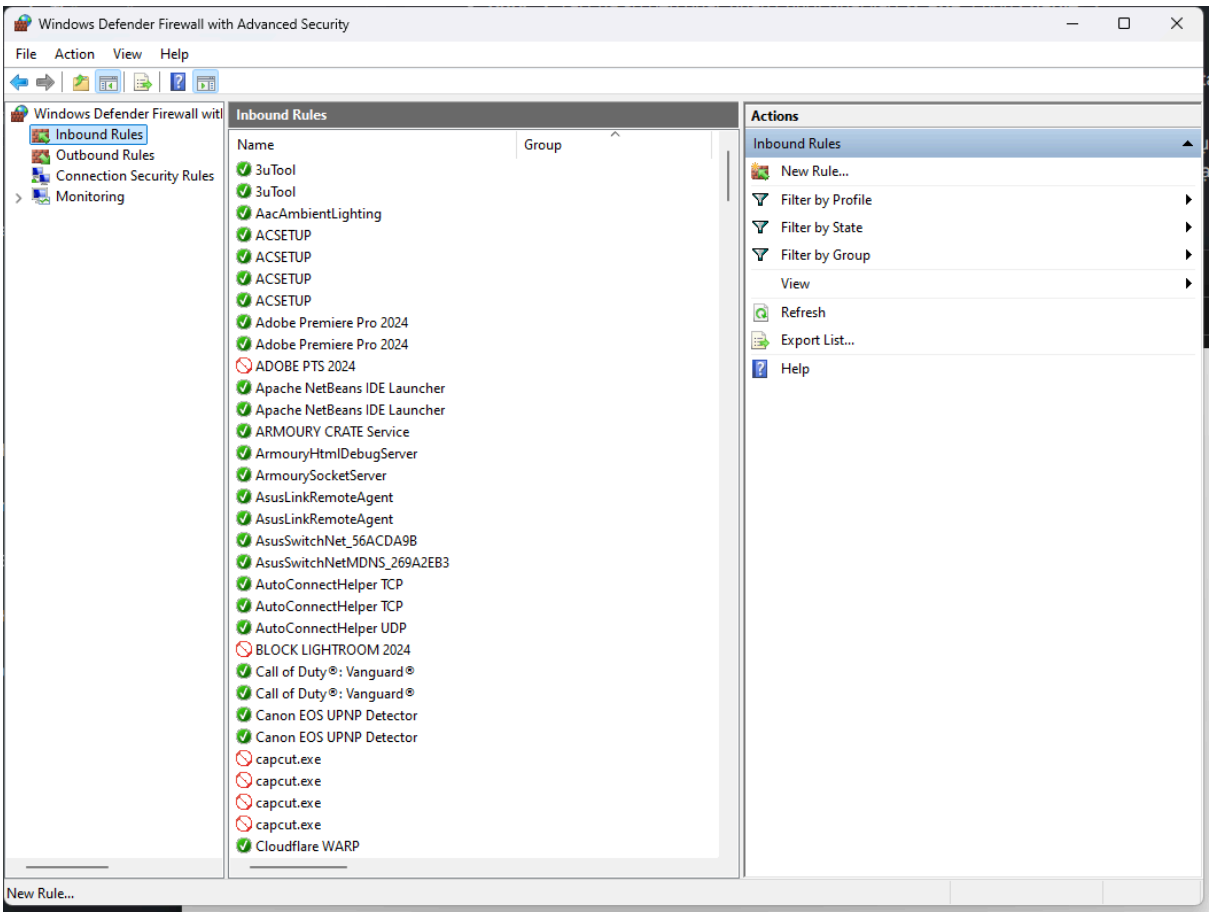
3. Thêm New Rule: csdlpt_nhom_7

Mục đích: Tạo một quy tắc trong tường lửa Windows để cho phép luồng dữ liệu truyền qua cổng TCP 1433.

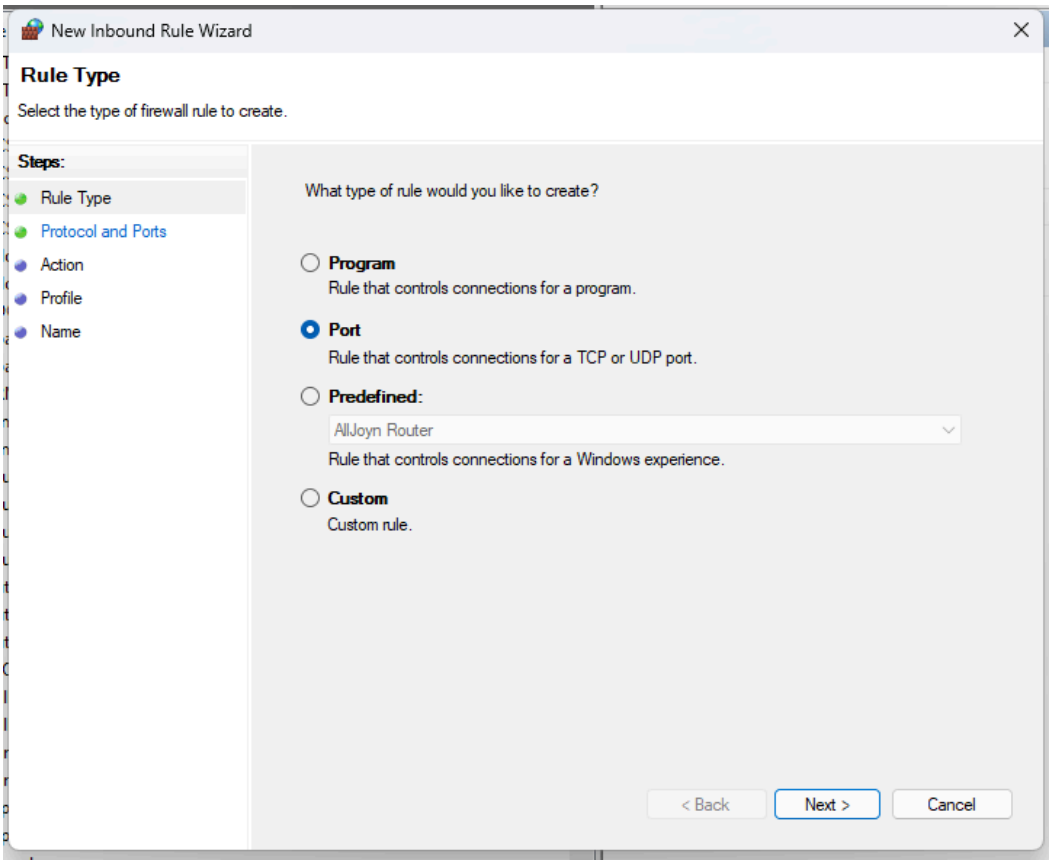
Bước 1: Mở Windows Defender Firewall with Advanced Security, sau đó chọn Advanced Settings.



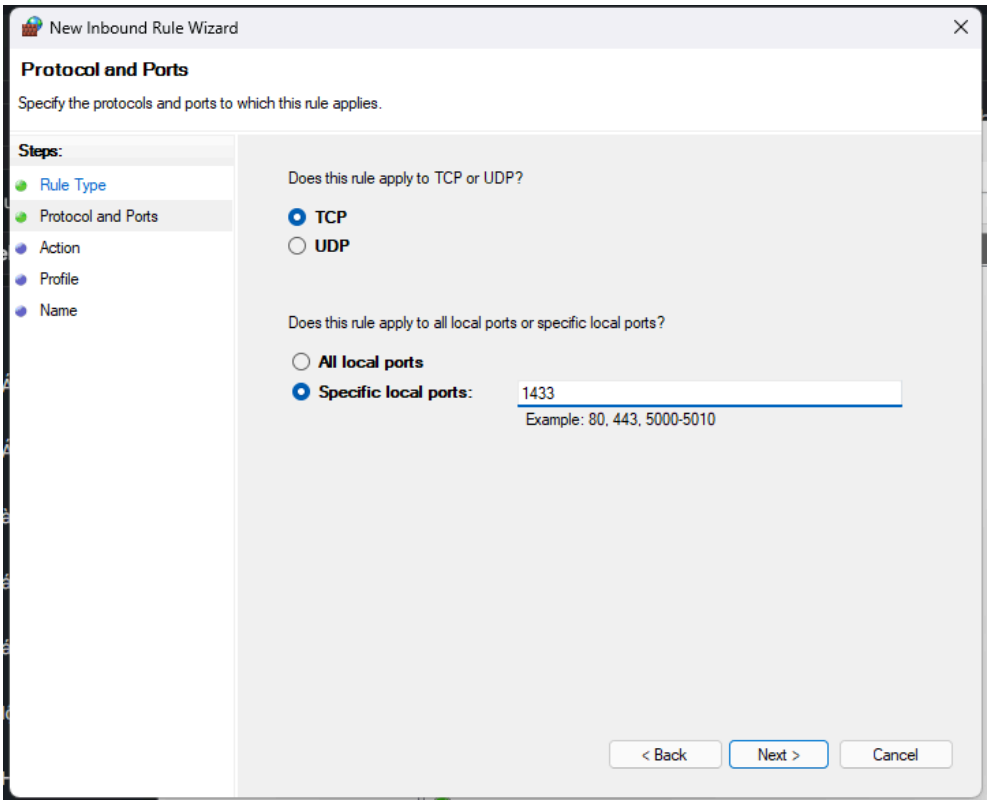
Bước 2: Nhấn chọn Inbound Rules ở cột bên trái -> Chọn New Rule... ở cột Action bên phải,.



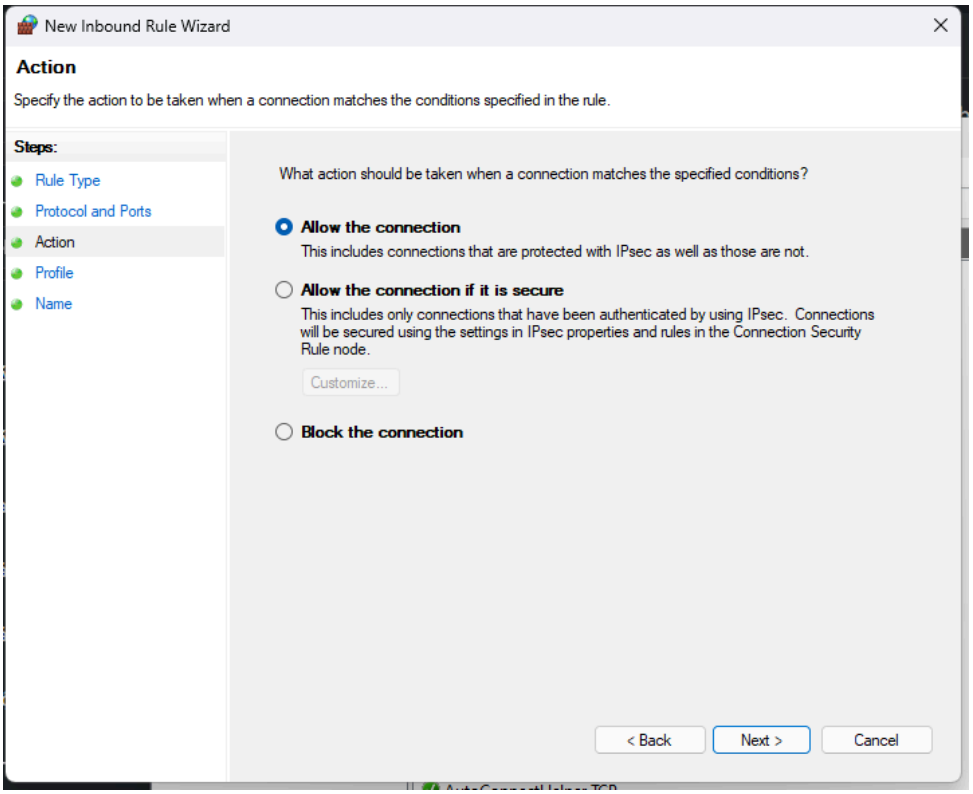
Bước 3: Tại mục Rule Type, tích chọn Port -> Chọn Next,.



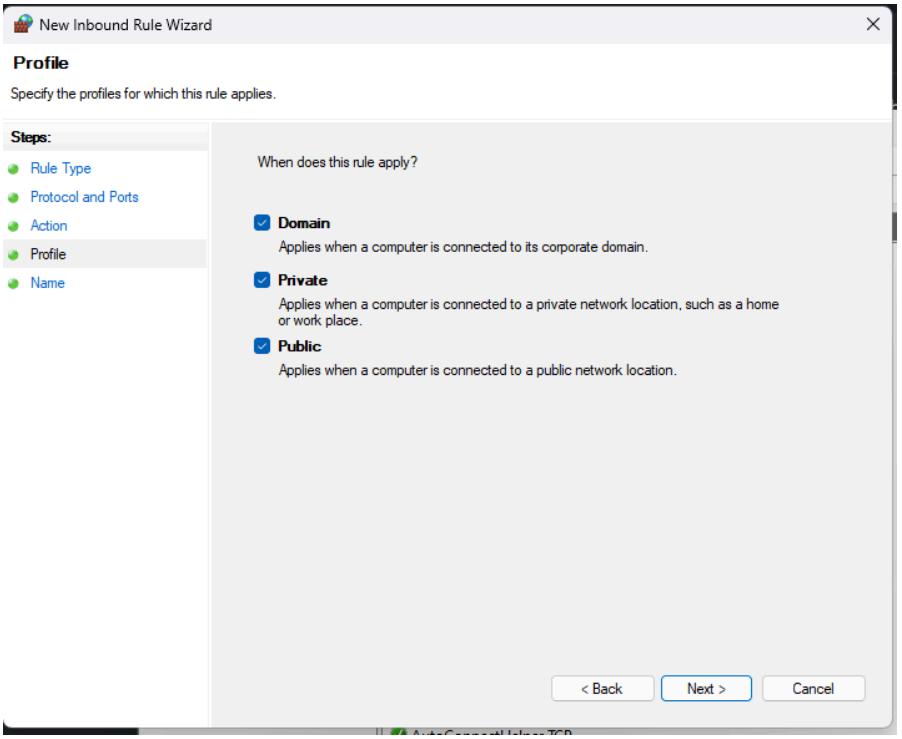
Bước 4: Tại mục Protocol and Ports, chọn giao thức TCP và tích vào ô Specific local ports, nhập vào giá trị 1433 -> Ấn Next,.



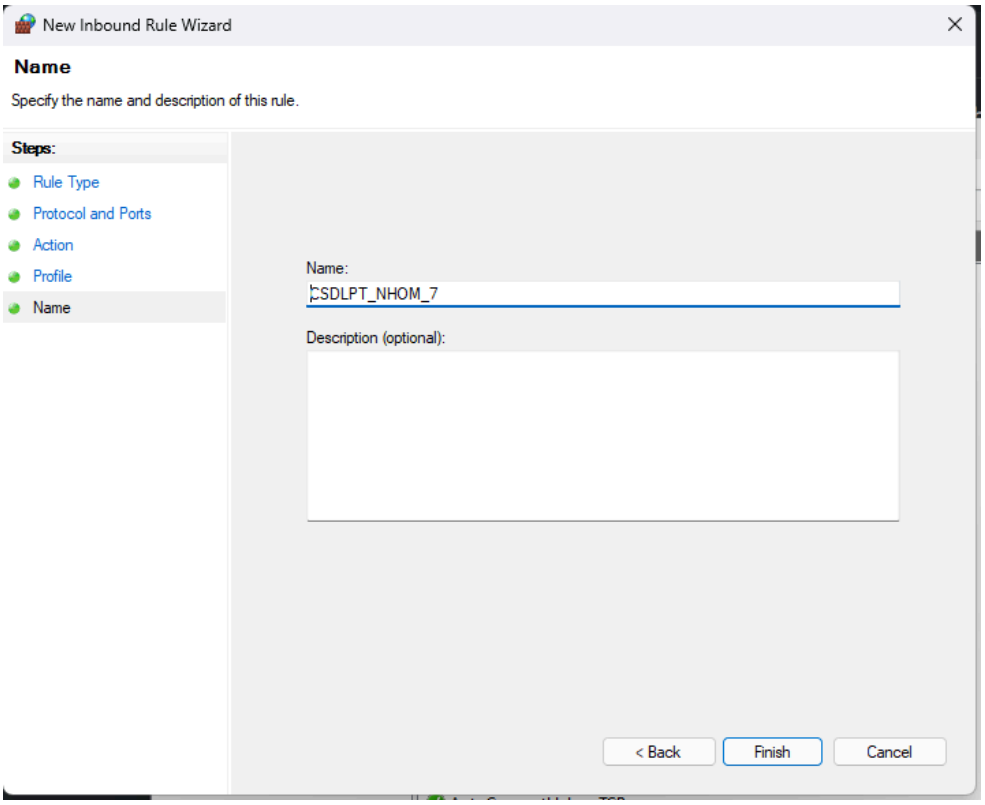
Bước 5: Tại mục Action, chọn Allow the connection -> Ấn Next.

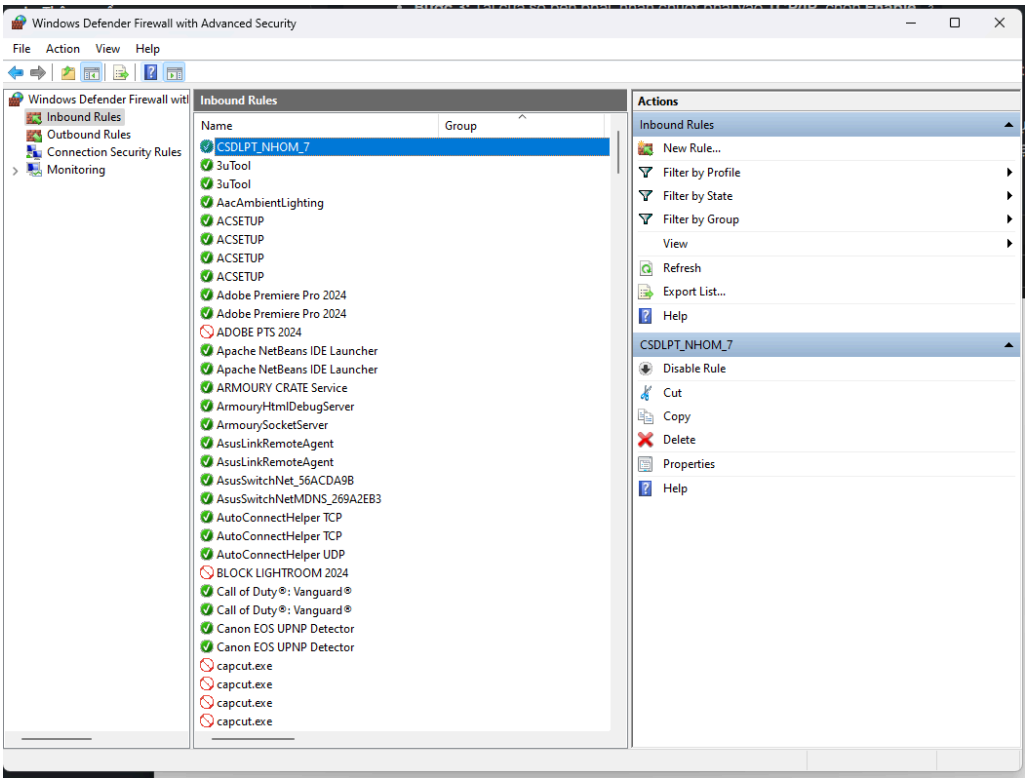


Bước 6: Tại mục Profile, tích chọn tất cả các luồng (Domain, Private, Public) -> Bấm Next.



Bước 7: Tại mục Name, thực hiện đặt tên cho Rule -> Ấn Finish để hoàn tất.

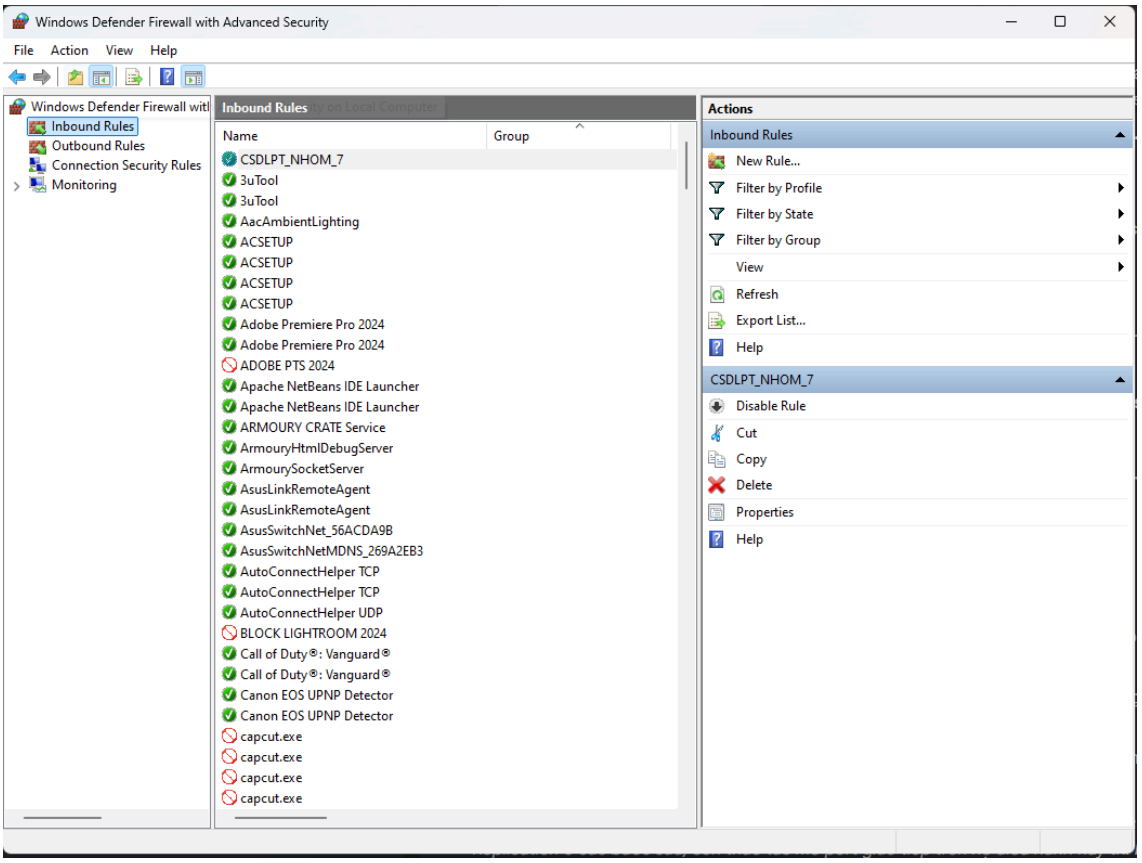




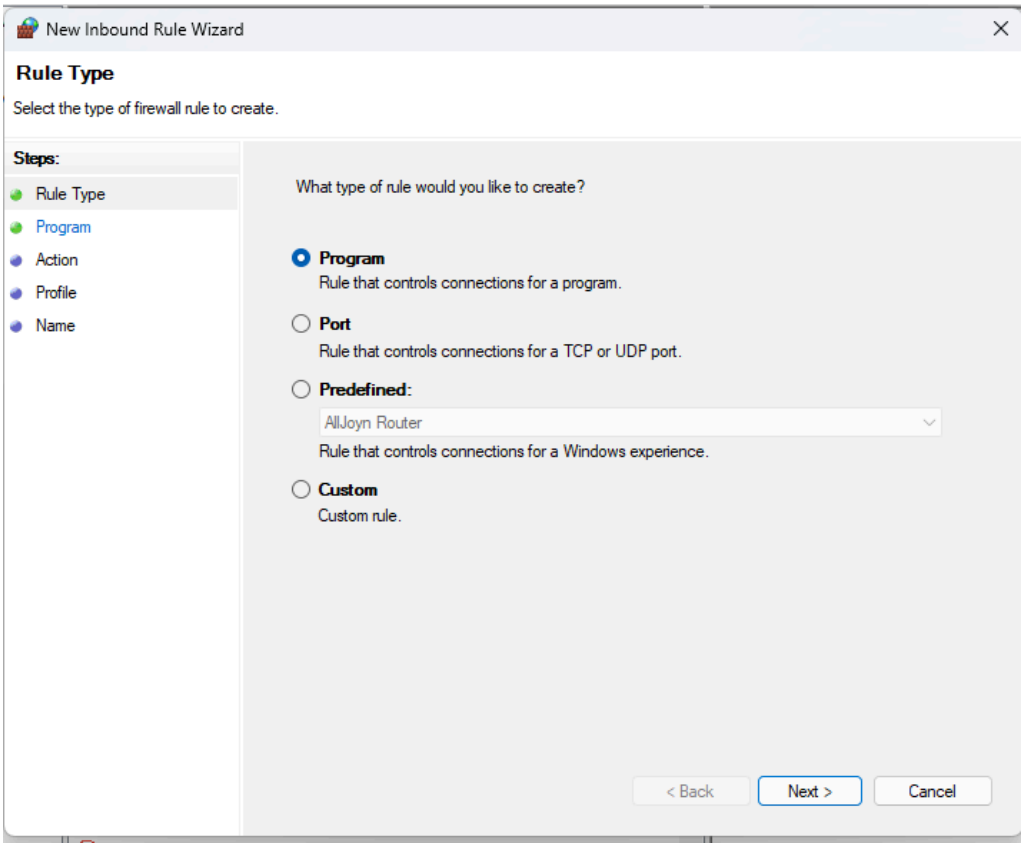
4. Thêm New Rule: sqlserver

Mục đích: Cấp quyền truy cập trực tiếp vào chương trình SQL Server 2025 thông qua file thực thi của hệ thống (sqlservr.exe) để các ứng dụng và SSMS 22 kết nối suôn sẻ,.

Bước 1: Tiếp tục tạo một Rule mới bằng cách chọn Inbound Rules -> New Rule....



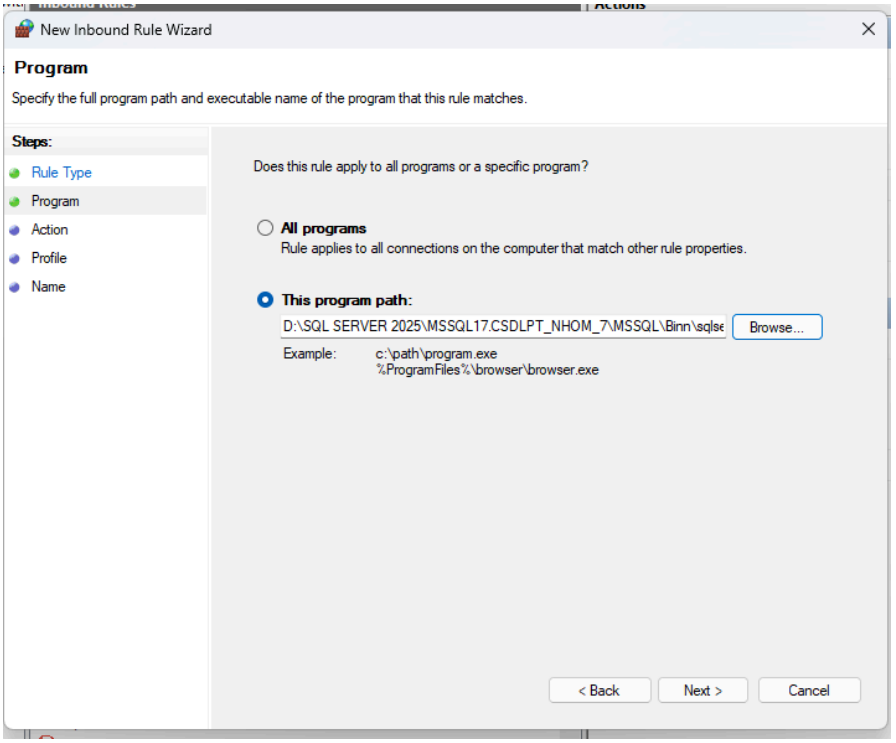
Bước 2: Trong mục Rule Type, chọn Program -> Ấn Next,.



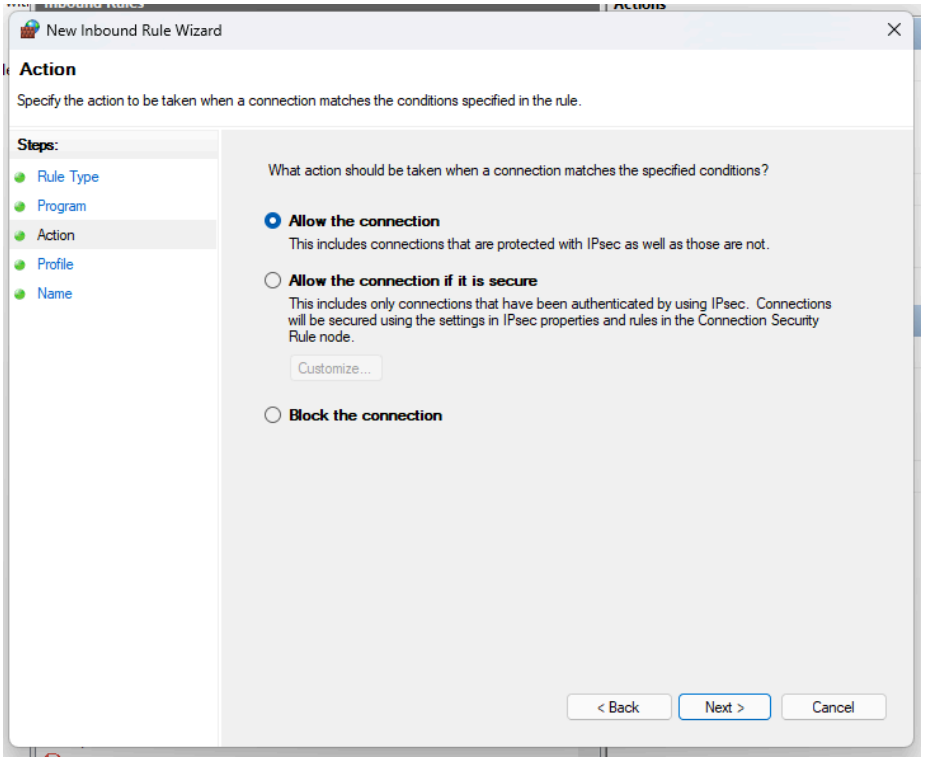
Bước 3: Chọn mục This program path và ấn Browse để trỏ tới file chạy của SQL Server 2025,. *(Lưu ý: Đối với SQL Server 2025, thư mục cài đặt mặc định thường sẽ*

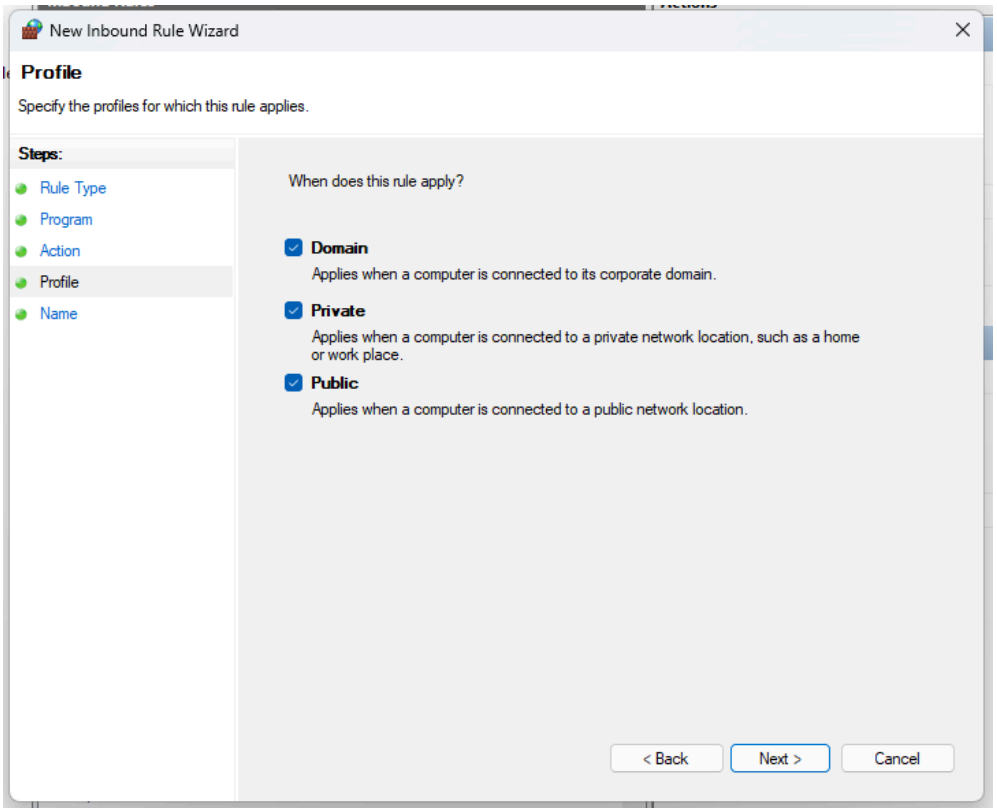
được đánh số phiên bản là 17. Do đó, đường dẫn của bạn sẽ có dạng: C:\Program Files\Microsoft SQL Server\MSSQL17.[Tên_Instance_Của_Bạn]\MSSQL\Binn\sqlservr.exe).

D:\SQL SERVER 2025\MSSQL17.CSDLPT_NHOM_7\MSSQL\Binn\sqlservr.exe

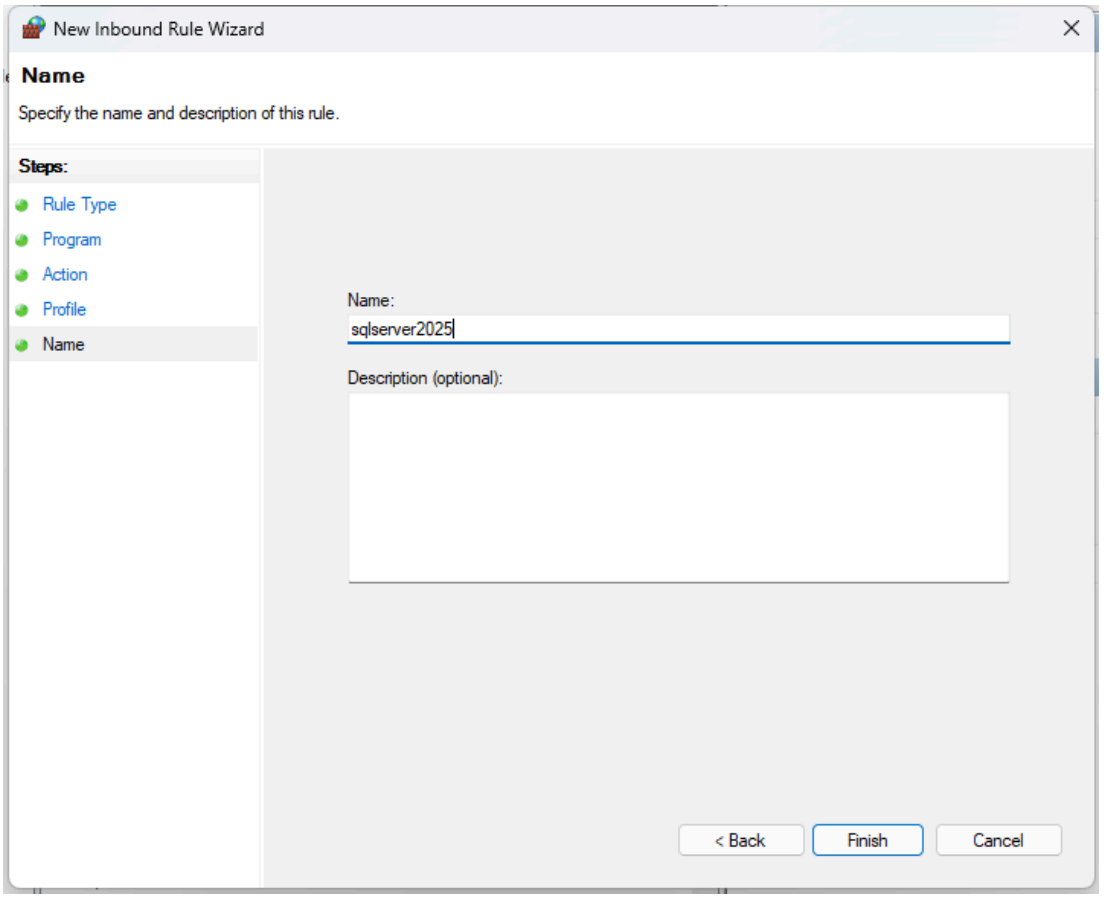


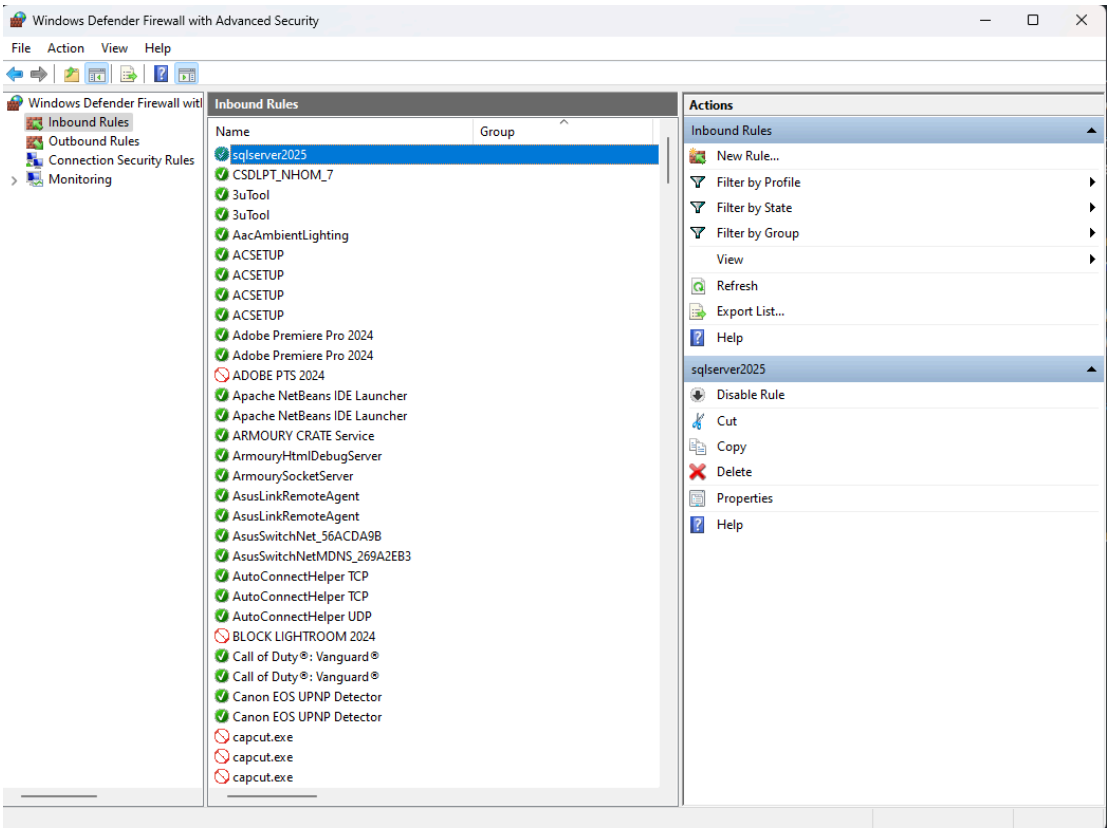
Bước 4: Tiếp tục ấn Next qua các bước tương tự (Chọn Allow the connection -> Tích chọn tất cả các Profile).





Bước 5: Tại bước cuối (Name), đặt tên cho Rule mới là sqlserver2025 -> Ấn Finish.





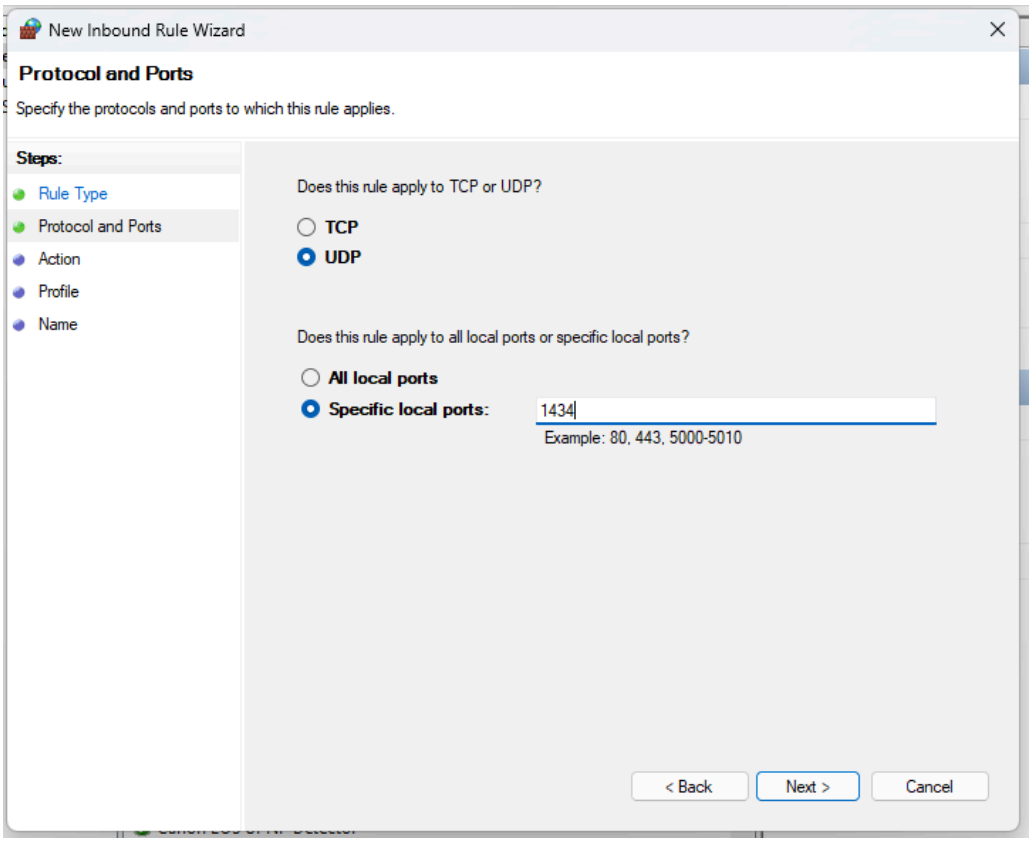
5. Thêm New Rule: port udp_1434

Mục đích: Mở cổng 1434 trong tường lửa nhằm cho phép trình duyệt dò tìm SQL Server Browser hoạt động qua giao thức UDP, điều này rất quan trọng để các máy trạm chi nhánh có thể "nhìn thấy" máy chủ khi phân tán dữ liệu.

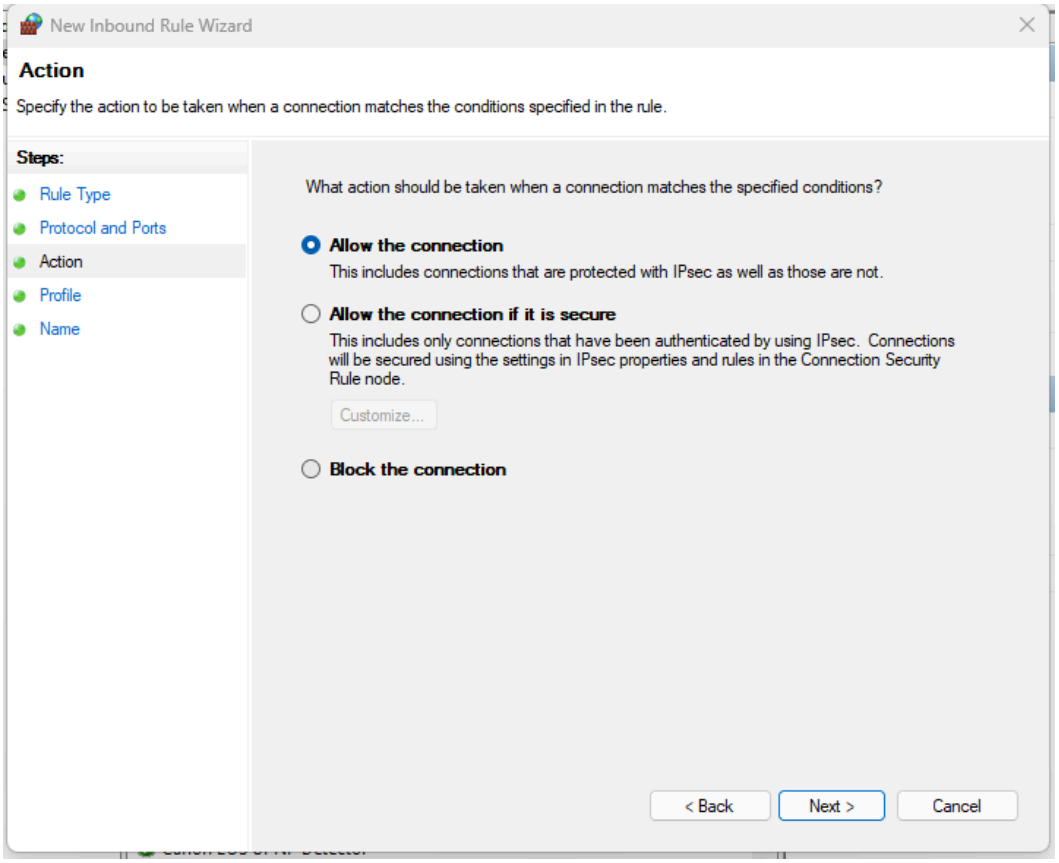
Bước 1: Tiếp tục chọn Inbound Rules -> New Rule....

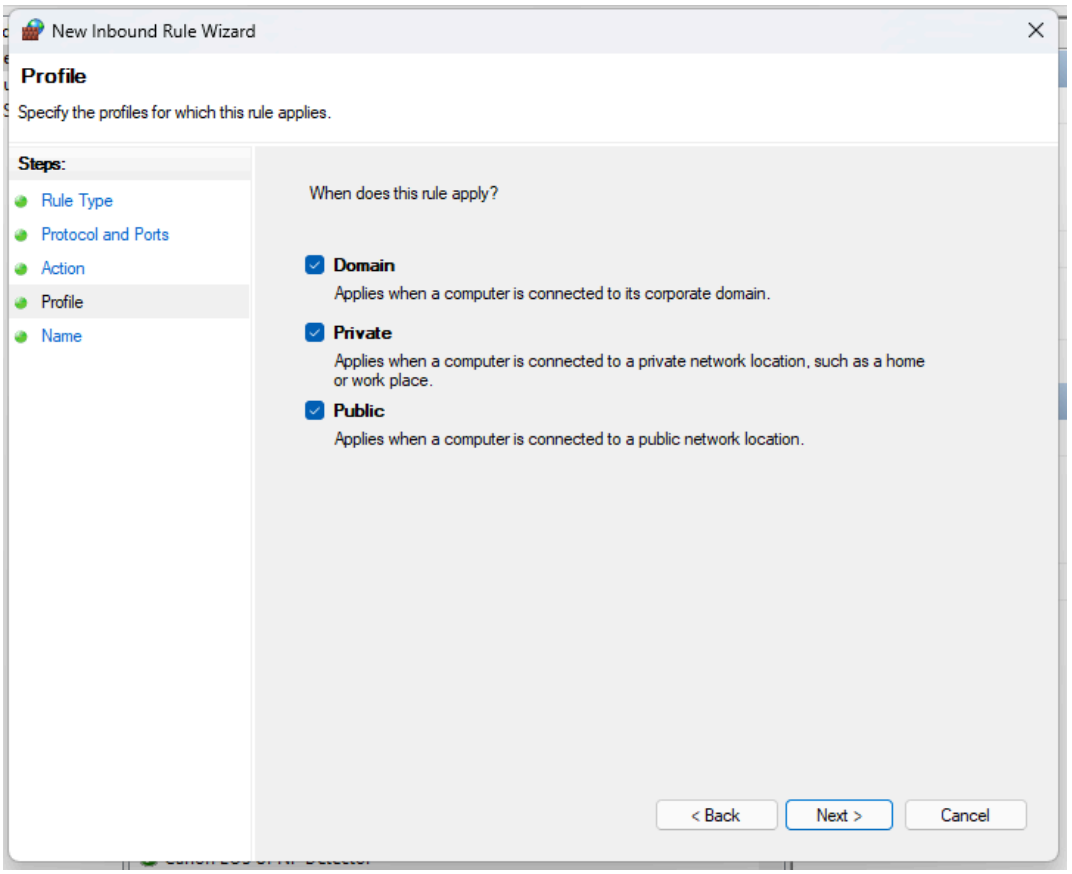
Bước 2: Tại mục Rule Type, chọn Port -> Ấn Next.

Bước 3: Tại mục Protocol and Ports, chọn giao thức UDP và mục Specific local ports nhập vào giá trị: 1434 -> Ấn Next.

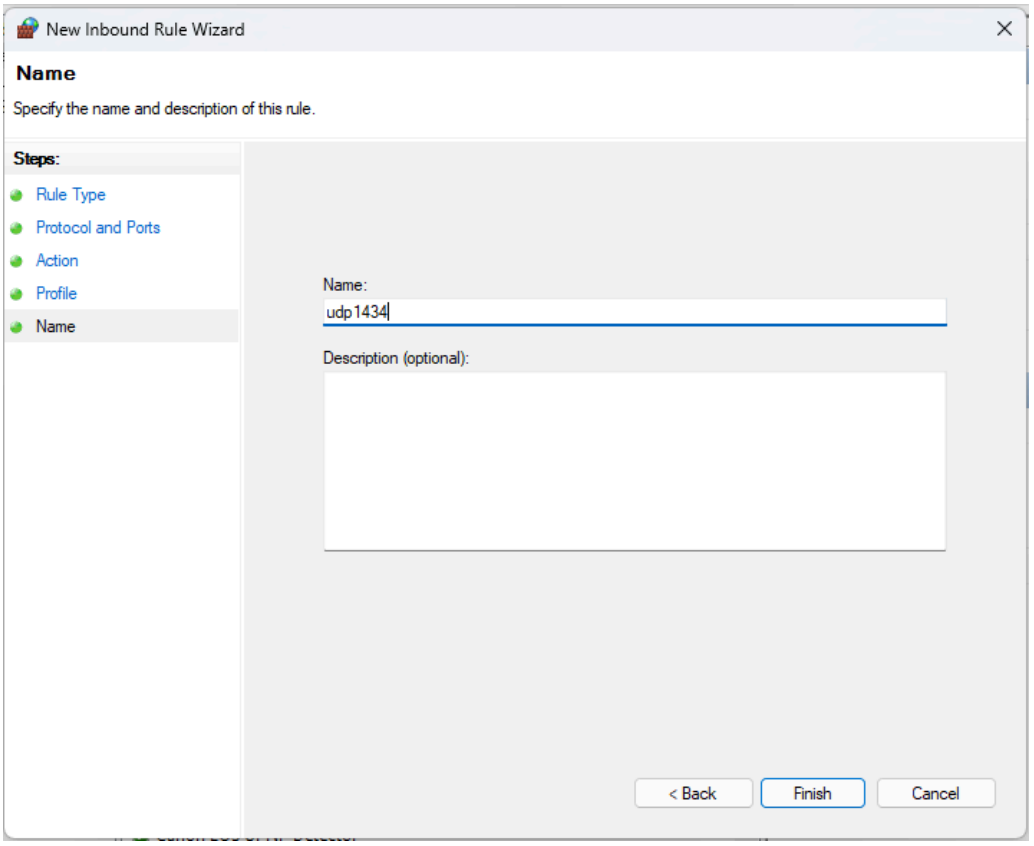


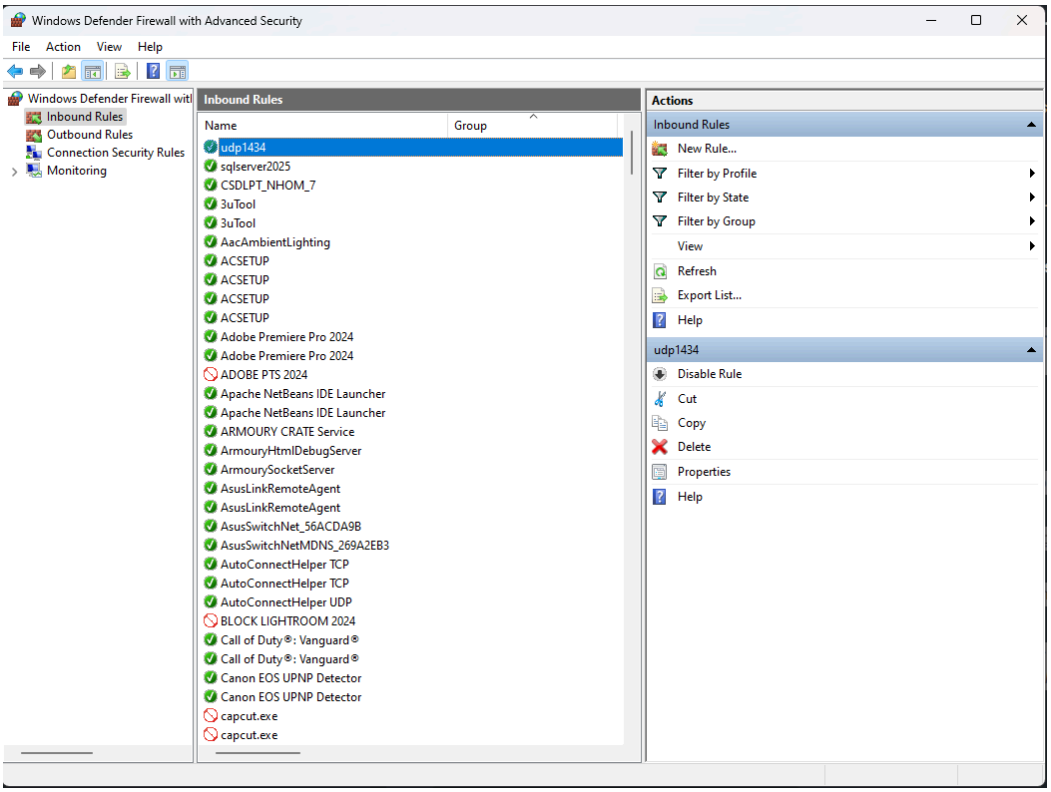
Bước 4: Tại mục Action, chọn Allow the connection -> Next. Tại mục Profile, chọn tất cả các mạng (Domain, Private, Public) -> Next.





Bước 5: Đặt tên cho Rule là udp1434 -> Ấn Finish để hoàn tất toàn bộ quá trình mở cổng trên tường lửa.



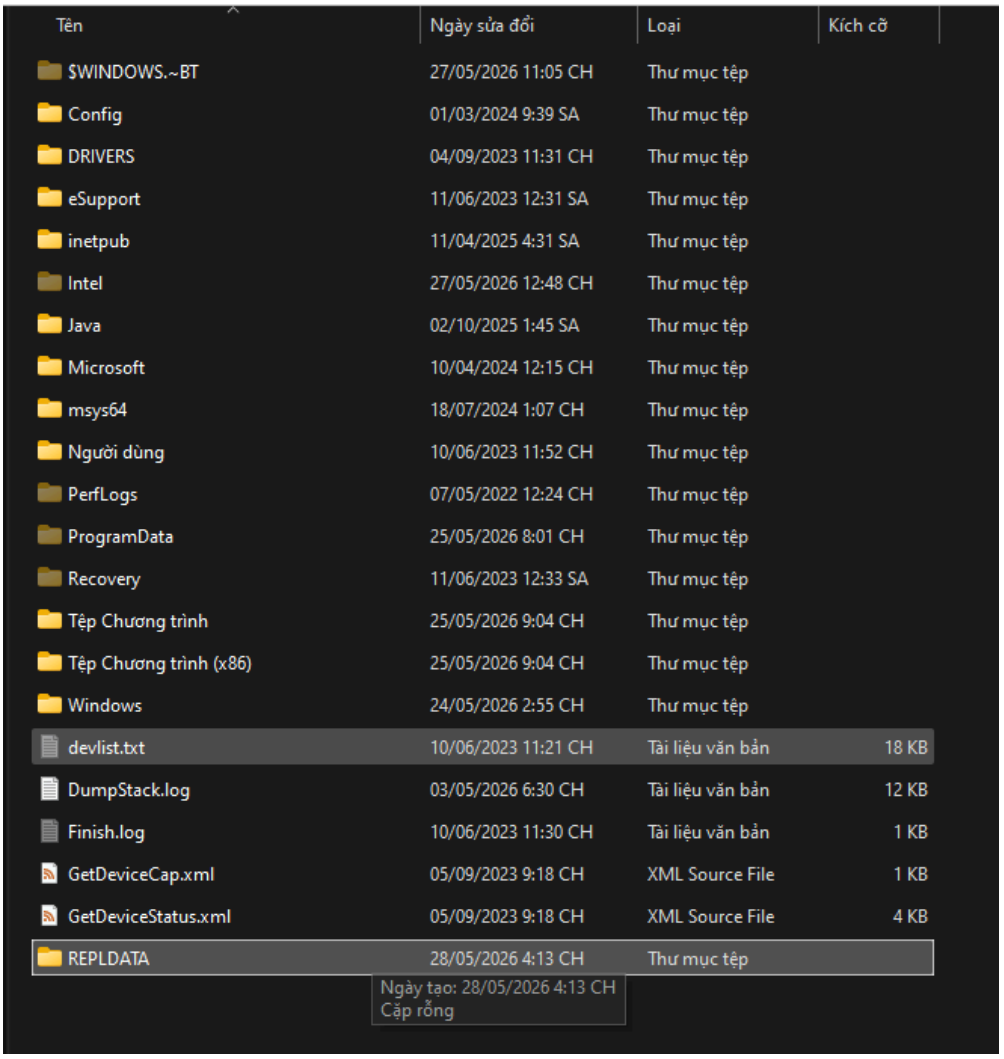


(Việc bạn dùng SSMS 22 sẽ chủ yếu thay đổi giao diện lúc viết câu lệnh (Query) hoặc cấu hình Replication ở các bước sau, còn thao tác mở port giao tiếp trên hệ điều hành này thì không bị ảnh hưởng).

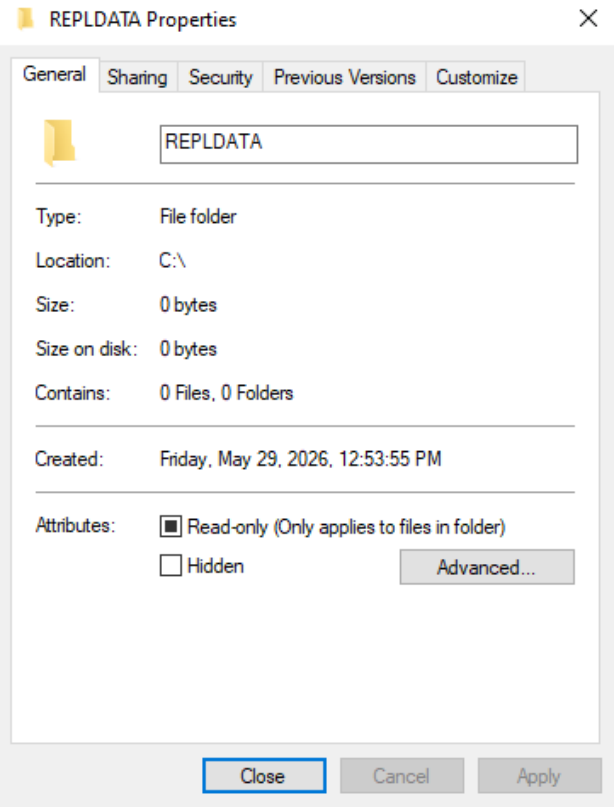
III. Chuẩn bị thư mục chứa dữ liệu đồng bộ (REPLDATA)

Mục đích: Khởi tạo thư mục REPLDATA dùng để chứa các dữ liệu trao đổi trong quá trình đồng bộ hóa (update) dữ liệu từ các máy trạm (phân mảnh) về cơ sở dữ liệu gốc (máy chủ), và ngược lại từ máy chủ đến các máy trạm.

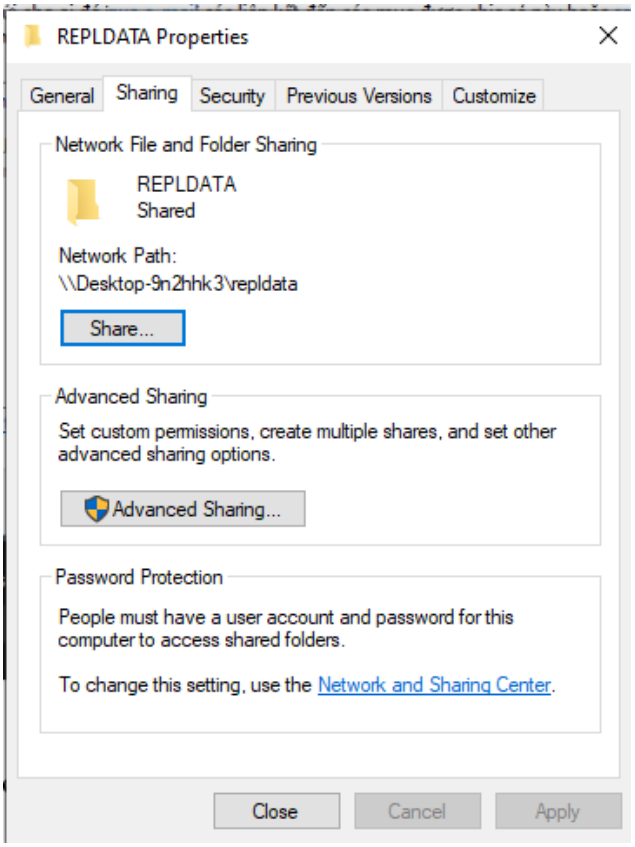
Bước 1: Tại ổ đĩa C của máy chủ, tạo một thư mục mới và đặt tên là REPLDATA (Đường dẫn: C:\REPLDATA).



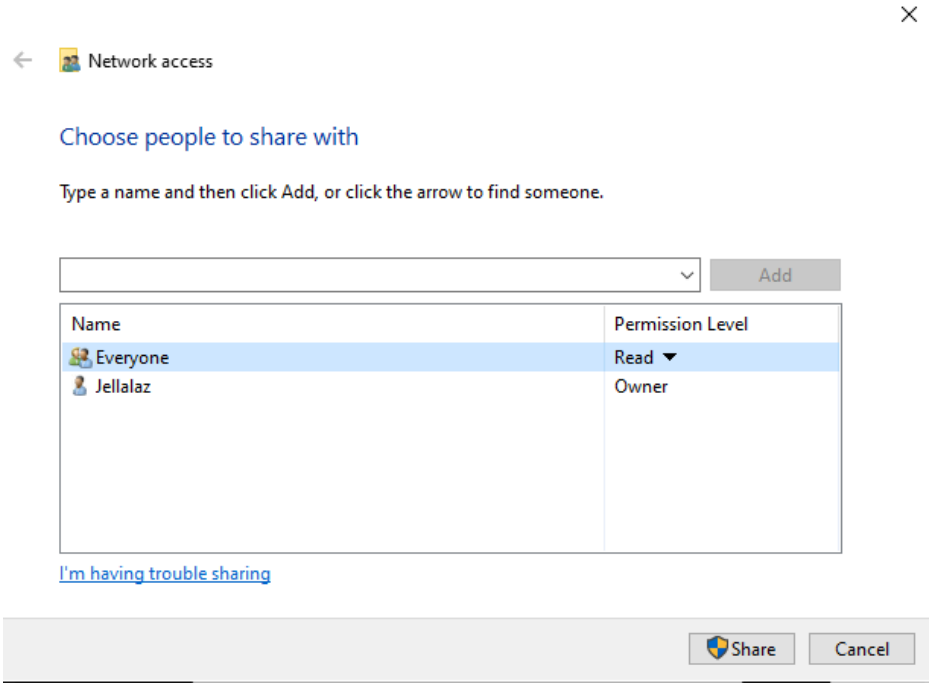
Bước 2: Click chuột phải vào thư mục REPLDATA vừa tạo, chọn Properties.



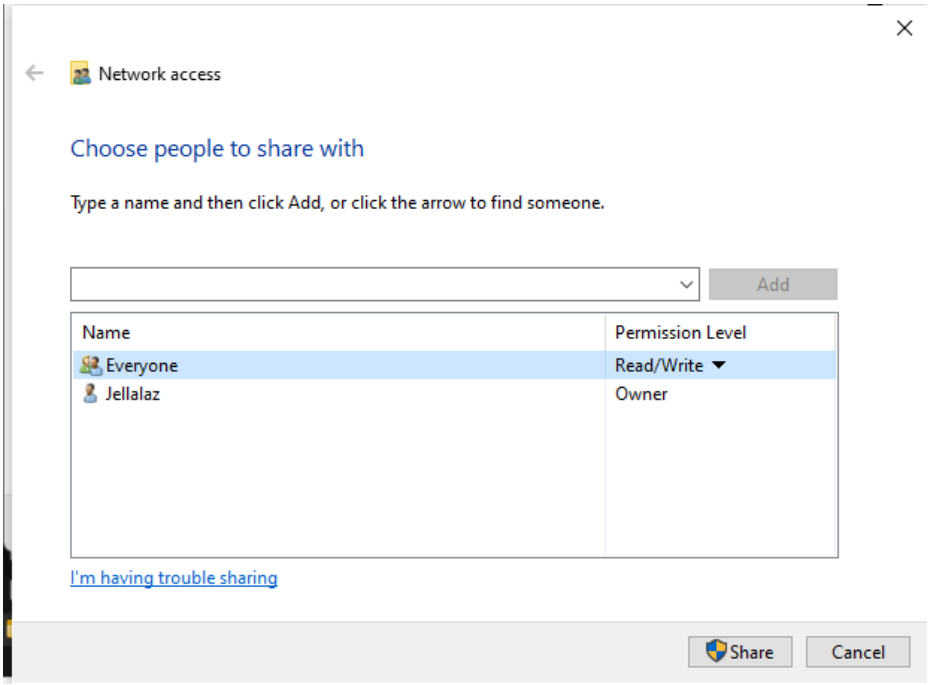
Bước 3: Chuyển sang tab Sharing, sau đó click vào nút Share....



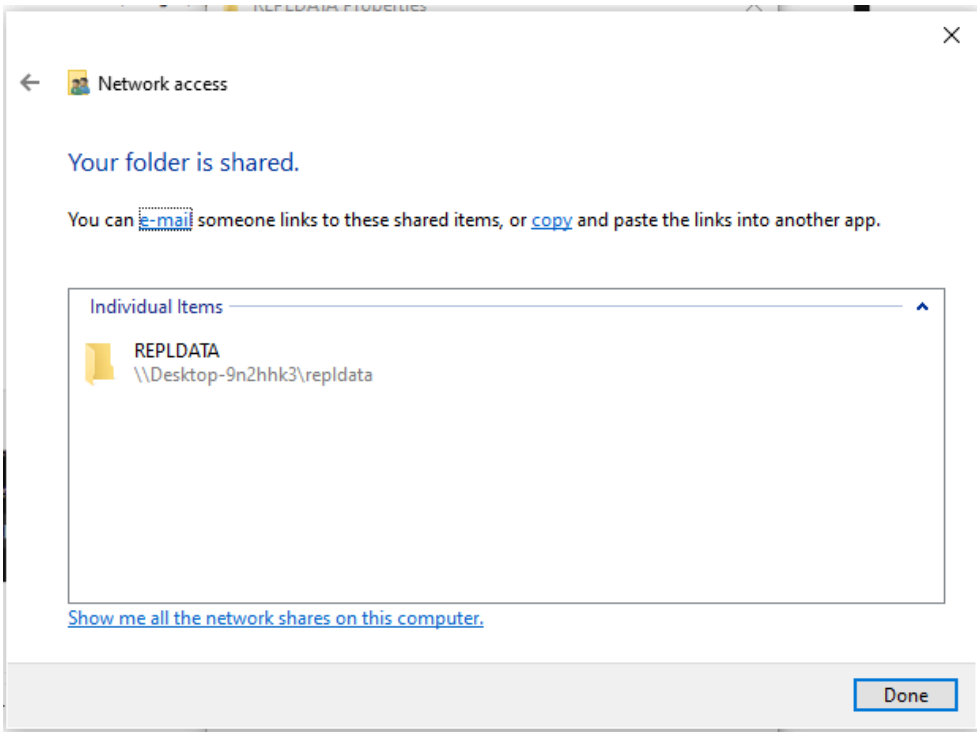
Bước 4: Tại thanh menu xổ xuống, tìm hoặc gõ chữ Everyone rồi click Add.



Bước 5: Ở cột *Permission Level* tương ứng với dòng Everyone, chuyển quyền truy cập thành Read/Write.



Bước 6: Click nút Share và sau đó nhấn Done để hoàn tất quá trình chia sẻ mạng cho thư mục.



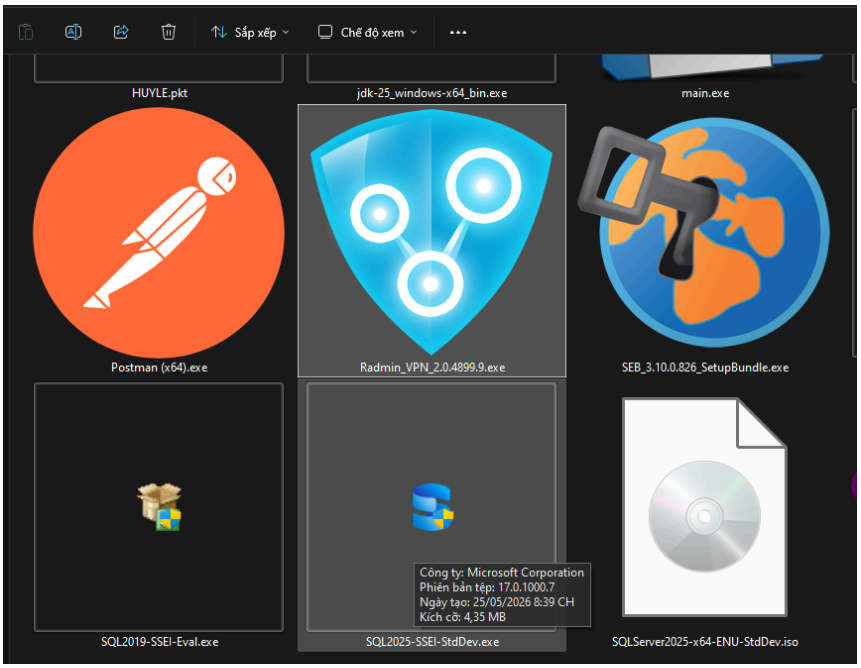
IV. Kết nối máy chủ và các chi nhánh bằng phần mềm Radmin VPN

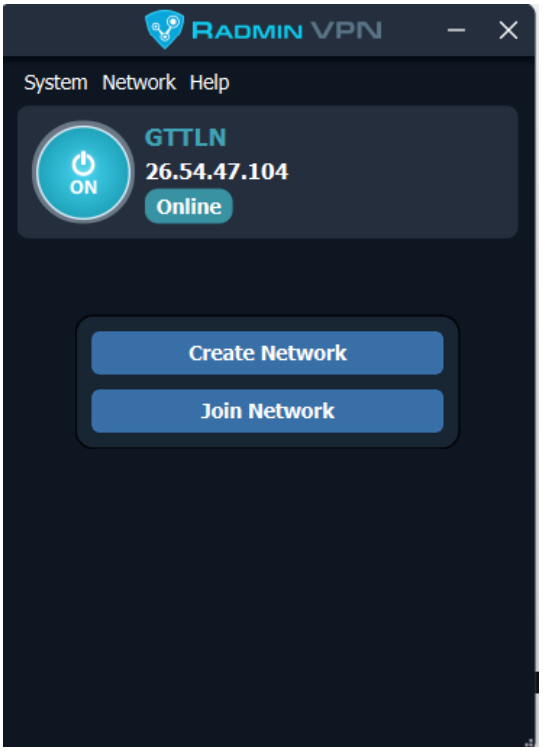
1. Tải và cài đặt phần mềm Radmin VPN

Giới thiệu: Phần mềm Radmin VPN được sử dụng để tạo mạng riêng ảo (VPN). Công cụ này cho phép thiết lập kết nối bảo mật giữa các máy tính (các máy trạm cơ sở đào tạo) thông qua Internet giống như thể chúng đang kết nối với nhau trên cùng một mạng LAN cục bộ.

Bước 1: Truy cập vào trang chủ để tải phần mềm Radmin VPN theo đường link: https://www.radmin-vpn.com/.

Bước 2: Khởi chạy file .exe vừa tải về máy, chọn Install và đợi quá trình cài đặt hoàn tất.





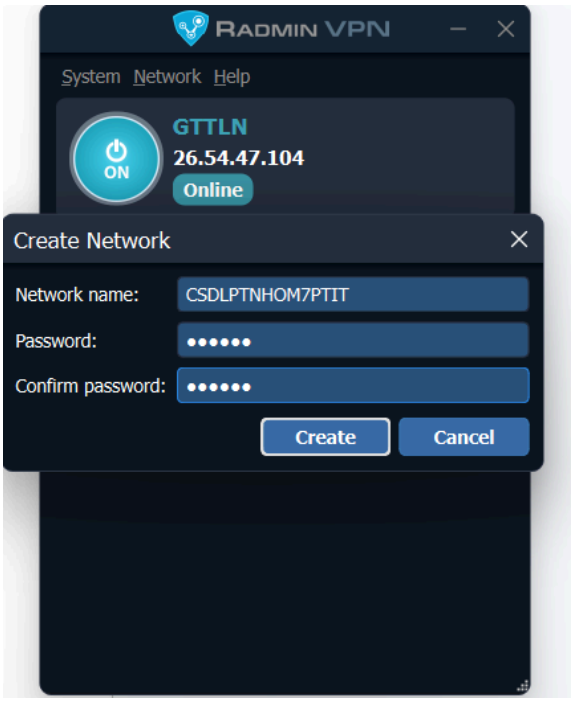
Cài đặt thành công

Bước 3: Tạo mạng dùng chung (Thực hiện trên máy chủ):

Mở Radmin VPN lên, nhấn vào *Create Network*.

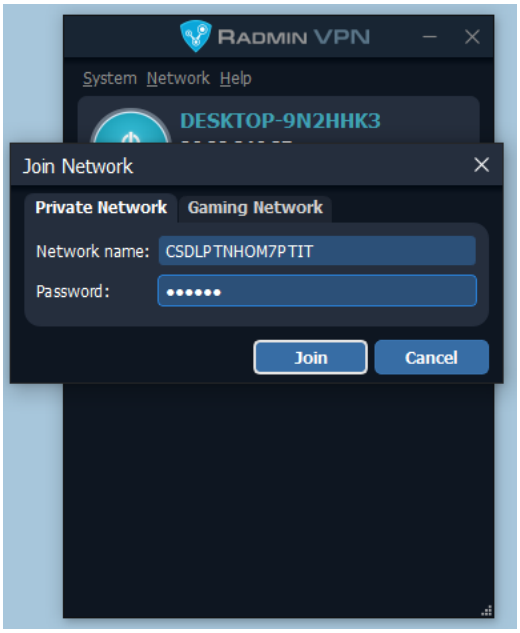
Tiến hành nhập Network name (CSDLPTNHOM7PTIT) và Password (123456).

Nhấn Create để khởi tạo mạng ảo.

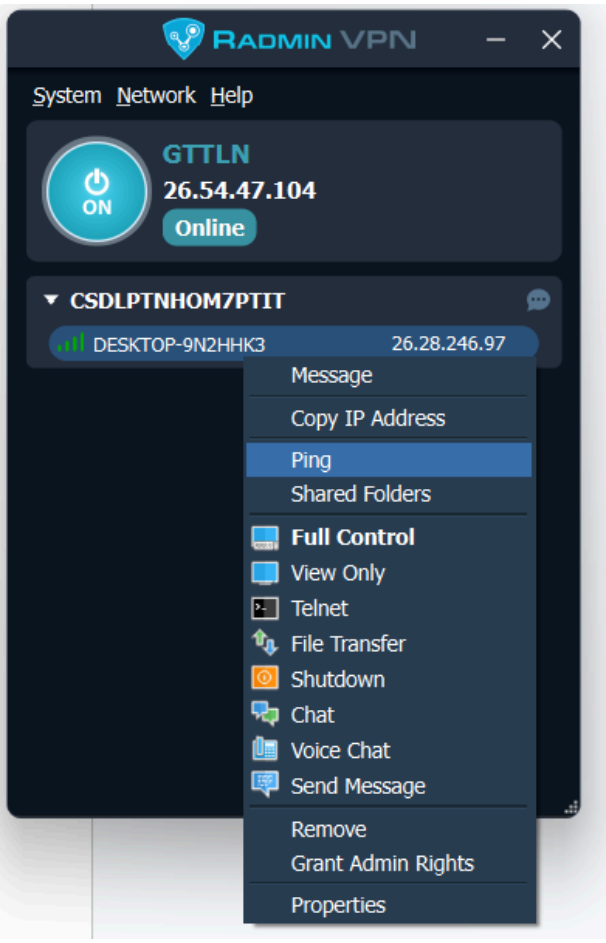


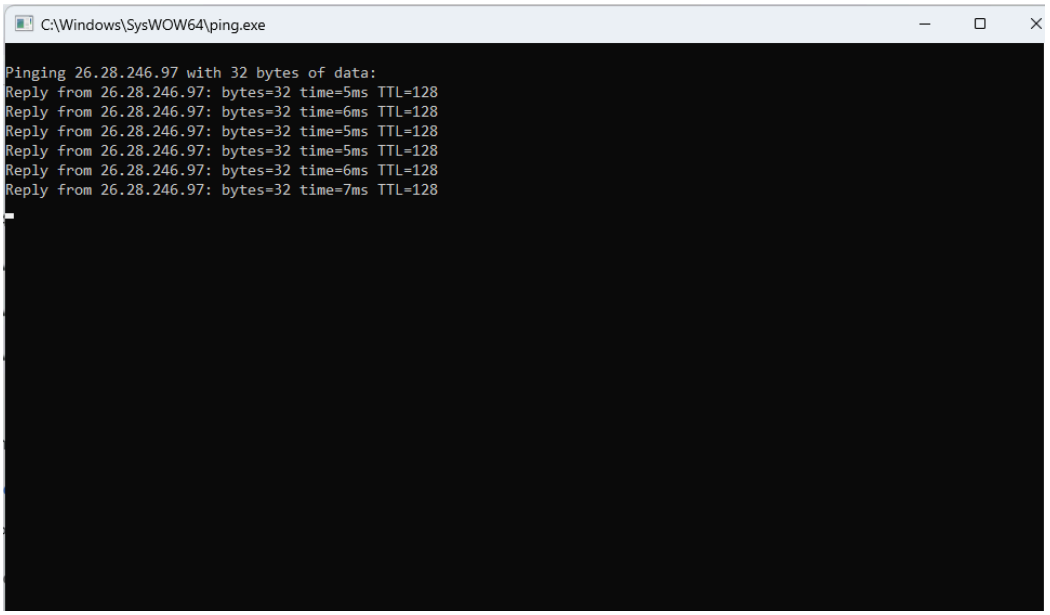
Bước 4: Tham gia mạng (Thực hiện trên các máy trạm):

Tại các máy trạm chi nhánh, mở Radmin VPN, chọn Join Network và nhập chính xác tên vả mật khẩu mạng vừa tạo ở trên để tham gia vào hệ thống.



Bước 5: Kiểm tra kết nối: Để đảm bảo đường truyền ổn định, nhấp chuột phải vào tên một máy thành viên trong danh sách của Radmin VPN và chọn Ping. Nếu cửa sổ cmd hiện ra các dòng chữ Reply from... liên tục nghĩa là các máy đã nhìn thấy nhau và kết nối thành công.

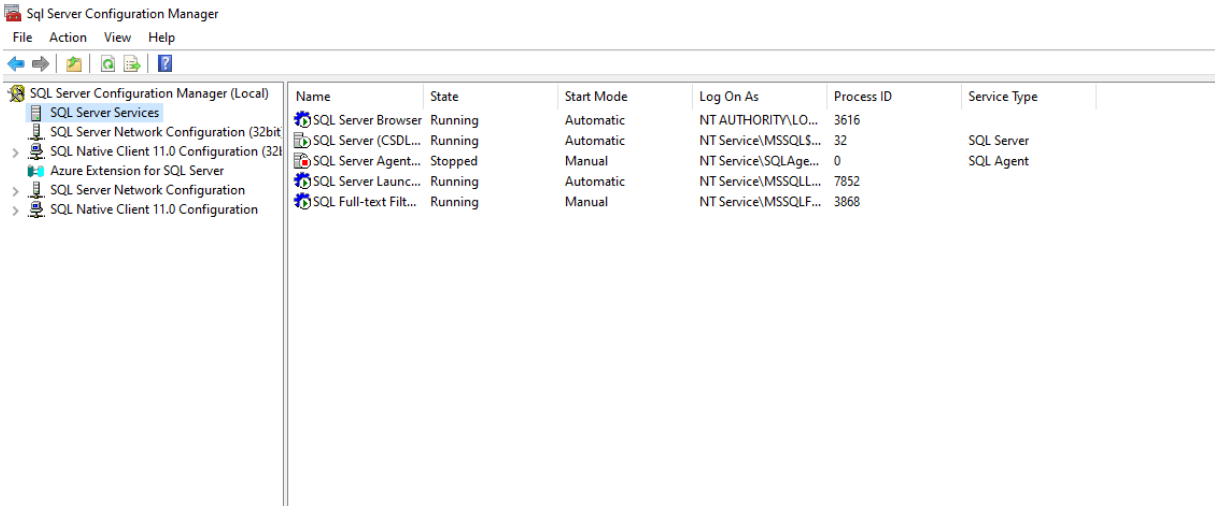




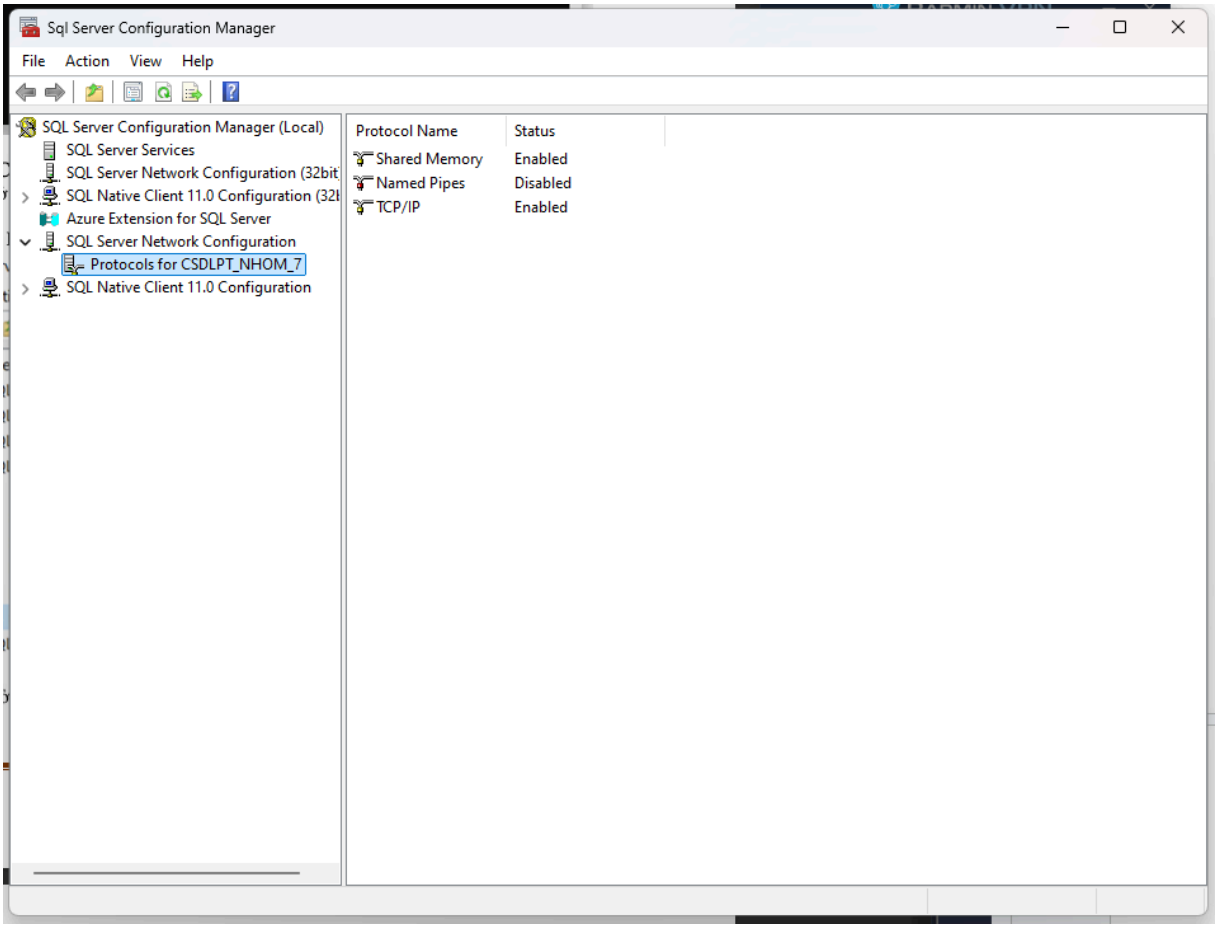
2. Bật TCP/IP và thêm IP cấu hình cho các máy trạm

Mục đích: Cấu hình để SQL Server cho phép kết nối từ xa thông qua địa chỉ IP ảo mà Radmin VPN vừa cấp phát.

Bước 1: Mở SQL Server Configuration Manager trên Windows.



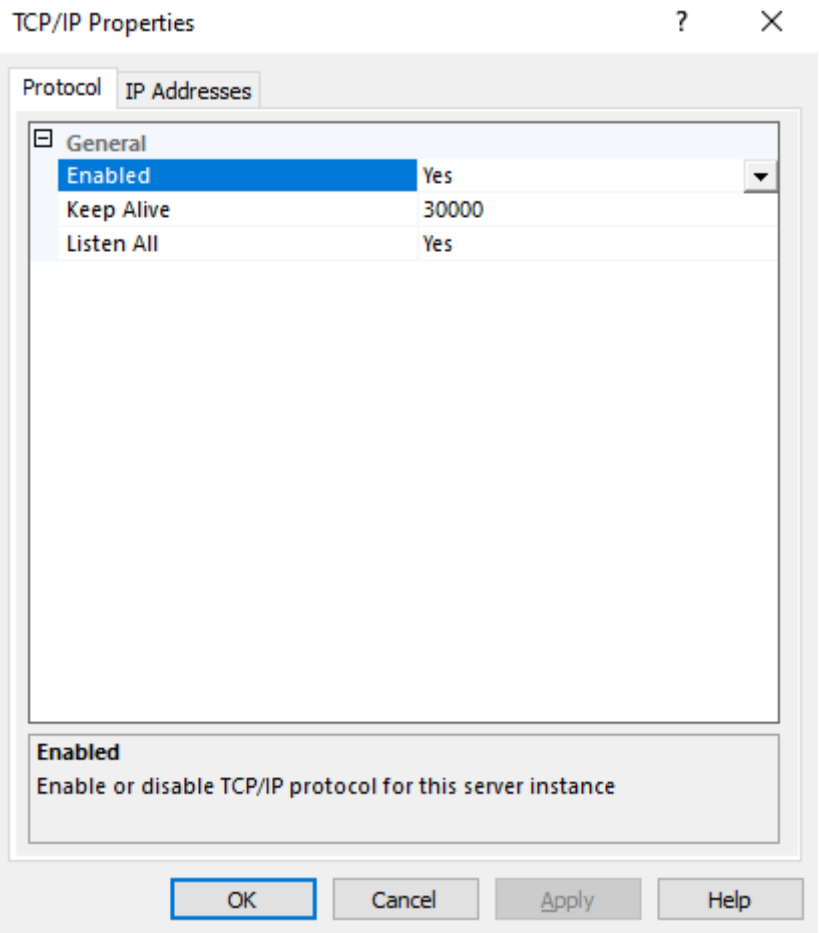
Bước 2: Ở menu bên trái, chọn SQL Server Network Configuration -> Protocols for CSDLPT_NHOM_7.

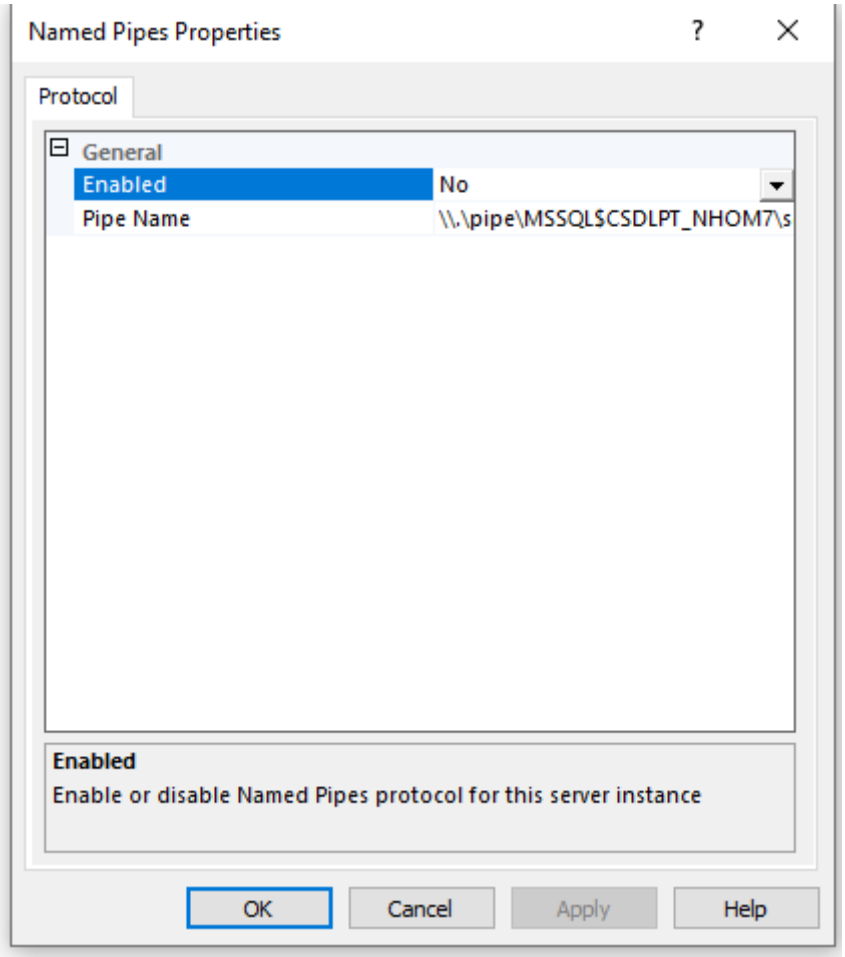


Bước 3: Đảm bảo cấu hình các giao thức như sau:

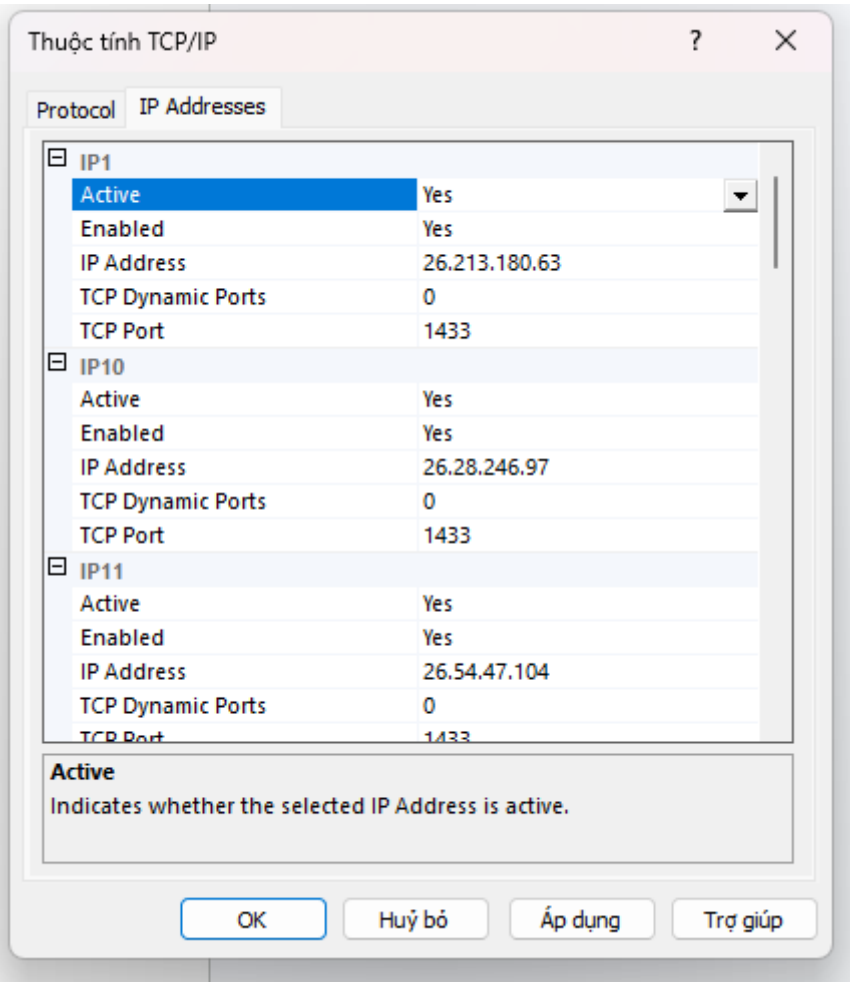
TCP/IP: Chọn Enable.

Named Pipes: Chọn Disabled.


TCP/IP Properties
Protocol
IP Addresses
General
Enabled
Yes
Keep Alive
30000
Listen All
Yes
Enabled
Enable or disable TCP/IP protocol for this server instance
OK
Cancel
Apply
Help


Named Pipes Properties
Protocol
General
Enabled
No
Pipe Name
\\.\pipe\MSSQL$CSDLPT_NHOM7\s
Enabled
Enable or disable Named Pipes protocol for this server instance
OK
Cancel
Apply
Help

Bước 4: Click chuột phải vào TCP/IP chọn Properties. Chuyển sang tab IP Addresses.

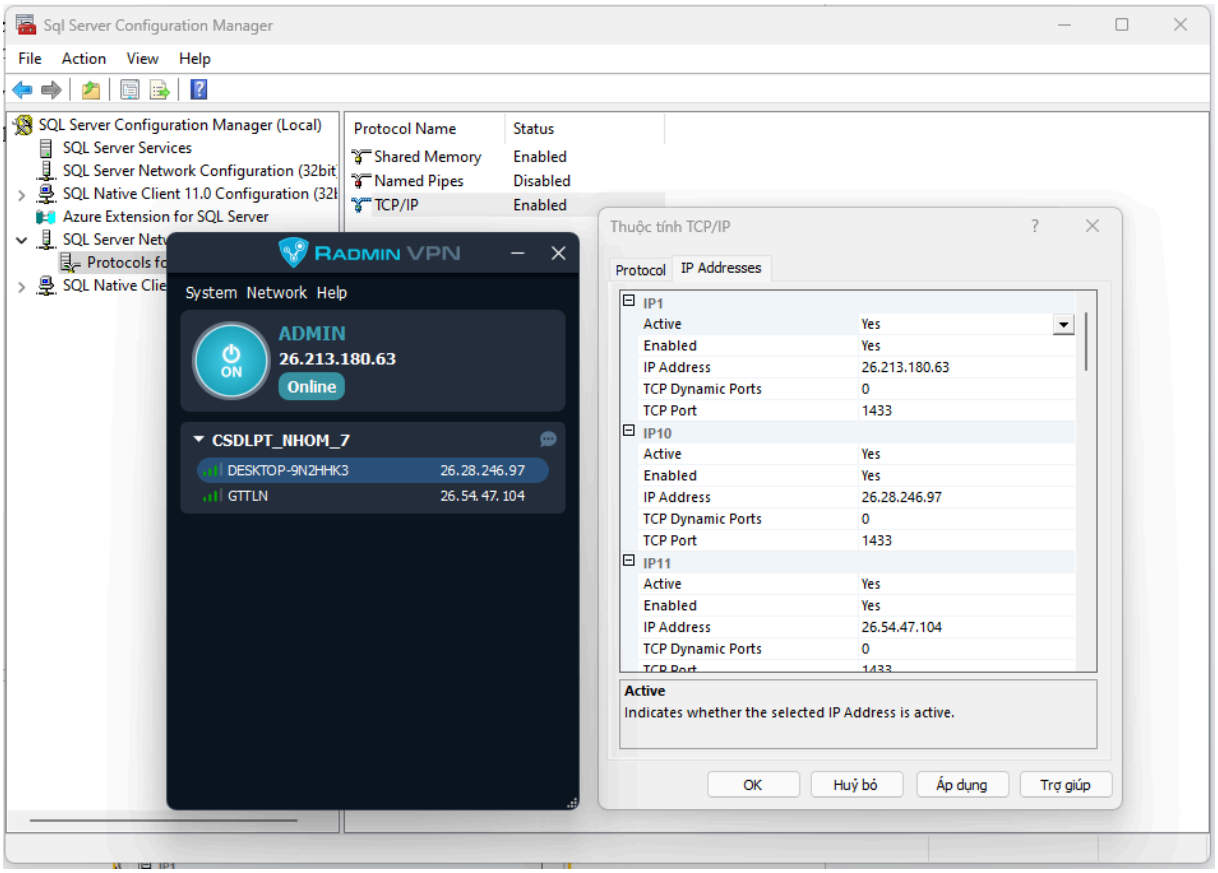


Bước 5: Tìm đến mục chứa IP được Radmin VPN cấp phát (Ví dụ mục IP1 đang có địa chỉ dạng 26.x.x.x), thực hiện cài đặt như sau:

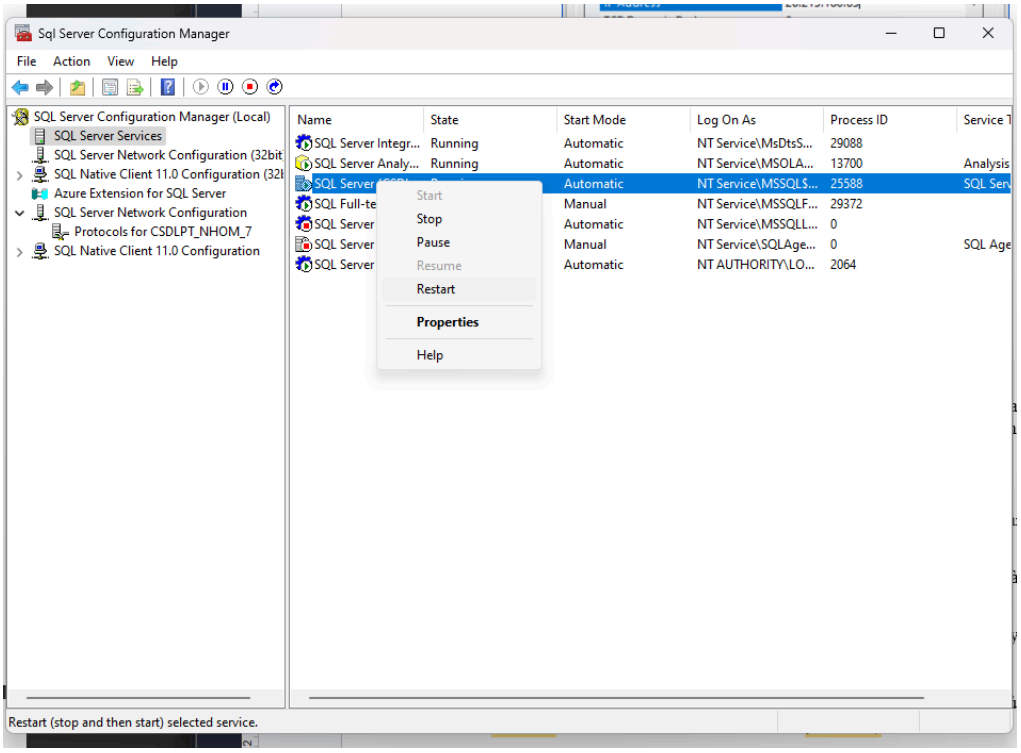
Active: Chọn Yes.

Enabled: Chọn Yes.

Tại ô TCP Port: Nhập 1433 (thực hiện tương tự cho tất cả các cổng TCP Port đang có).



Bước 6: Click Apply -> OK. Sau đó lùi lại mục SQL Server Services ở thanh bên trái, chuột phải vào SQL Server (CSDLPT_NHOM_7) và chọn Restart để hệ thống nhận cấu hình IP mới.

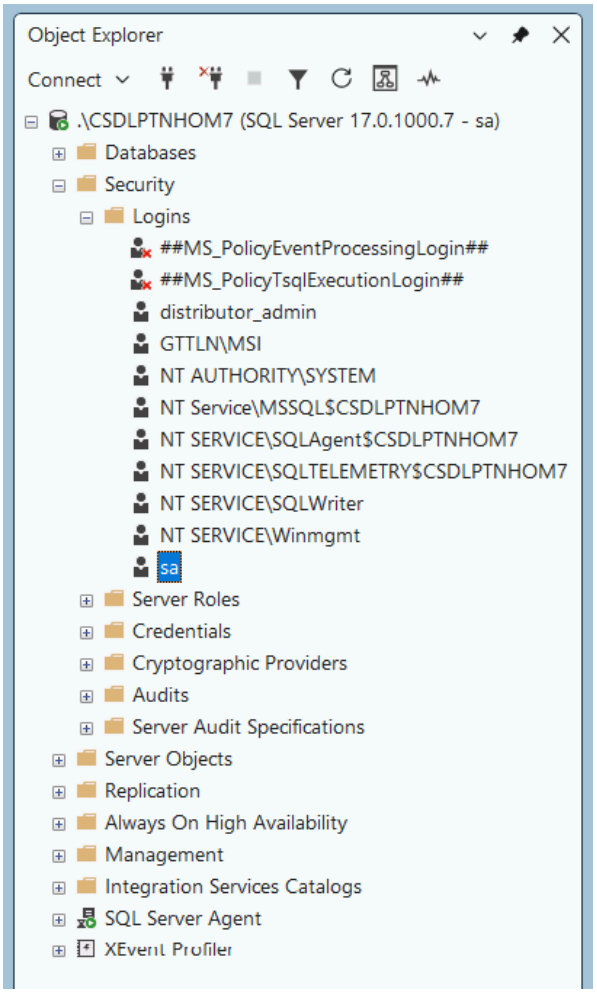


3. Tạo hoặc cấu hình tài khoản sa (supper admin)

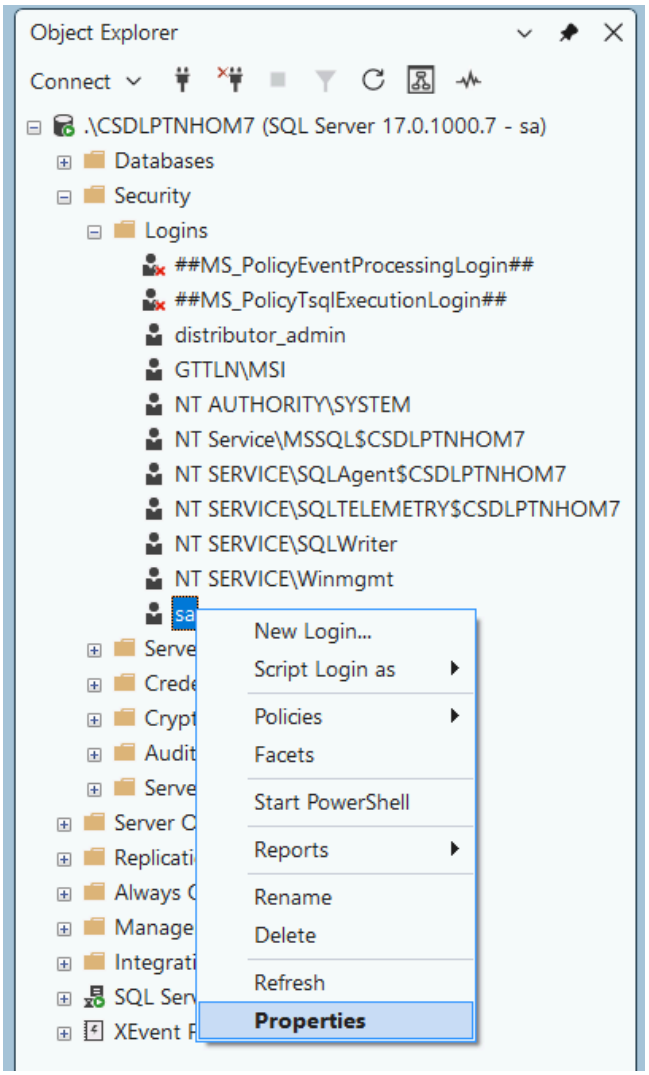
Mục đích: Kích hoạt tài khoản quản trị viên cao nhất của SQL Server để làm tài khoản xác thực khi phân tán dữ liệu giữa máy chủ và máy trạm.

Bước 1: Mở SQL Server Management Studio (SSMS), chọn Server type là Database Engine và đăng nhập bằng chế độ Windows Authentication mặc định.

Bước 2: Tại cửa sổ Object Explorer, tìm theo đường dẫn: thư mục Security -> Logins.

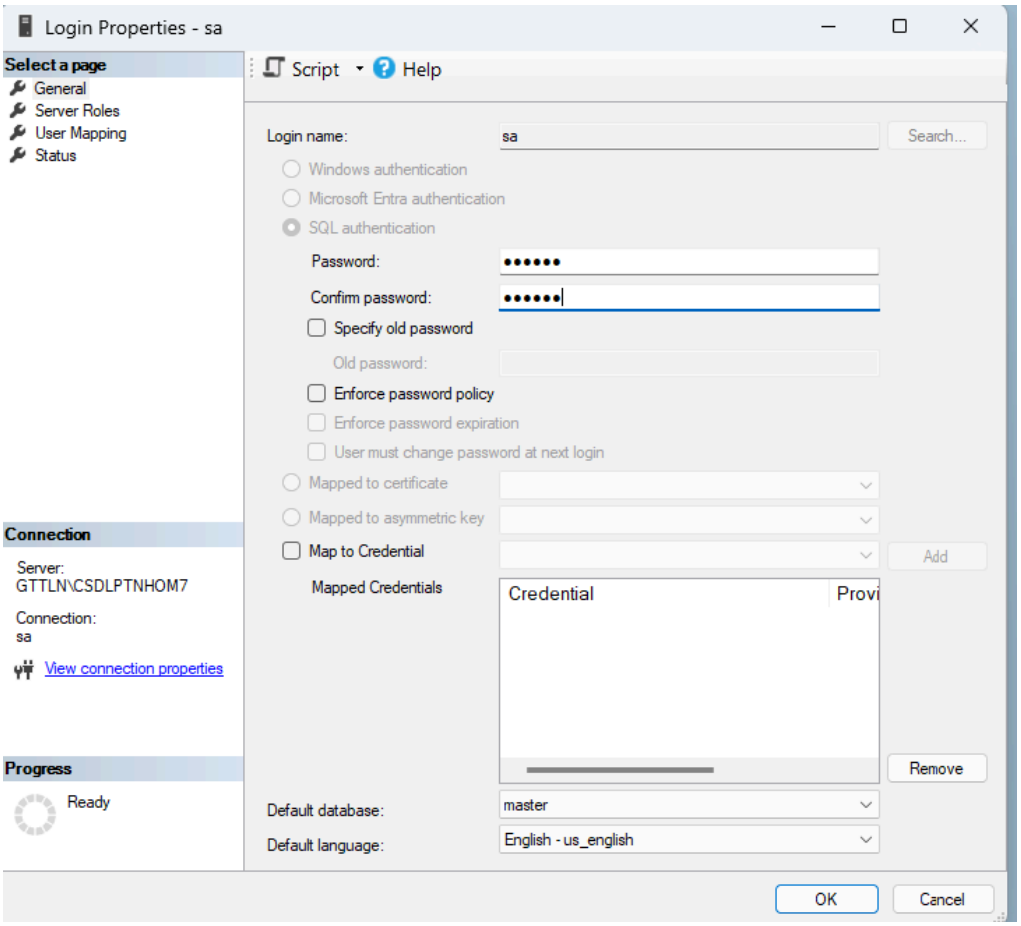


Bước 3: Kích chuột phải vào tài khoản có tên là sa, chọn Properties.



Bước 4: Tại tab General, nhập mật khẩu thiết lập cho hệ thống phân tán của nhóm (Ví dụ: 123456 vào cả 2 ô Password và Confirm password).

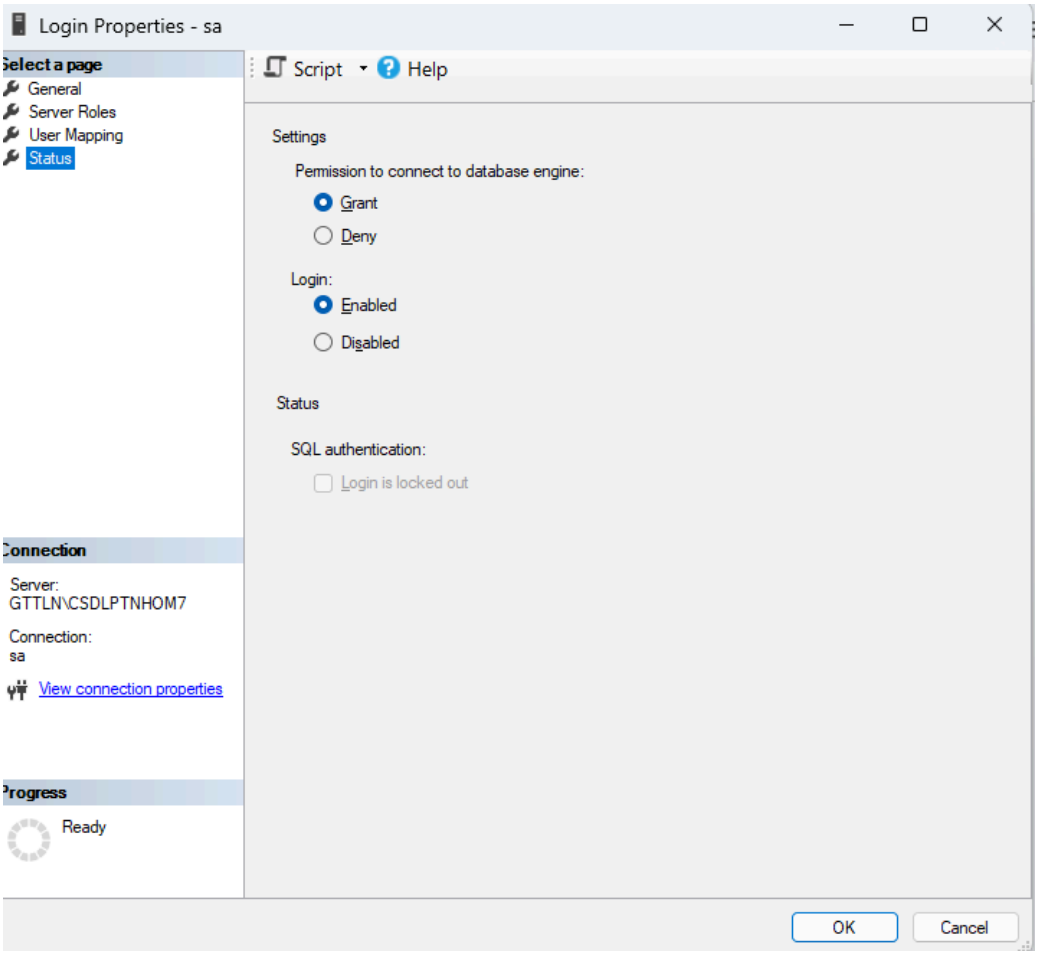
Bước 5**:** Bỏ dấu tích ở ô Enforce password policy để tránh bị lỗi báo mật khẩu yếu hoặc báo hạn sử dụng mật khẩu.



Bước 6: Chuyển sang tab Status ở cột bên trái. Đảm bảo cấu hình thiết lập như sau:

Permission to connect to database engine: Chọn Grant.

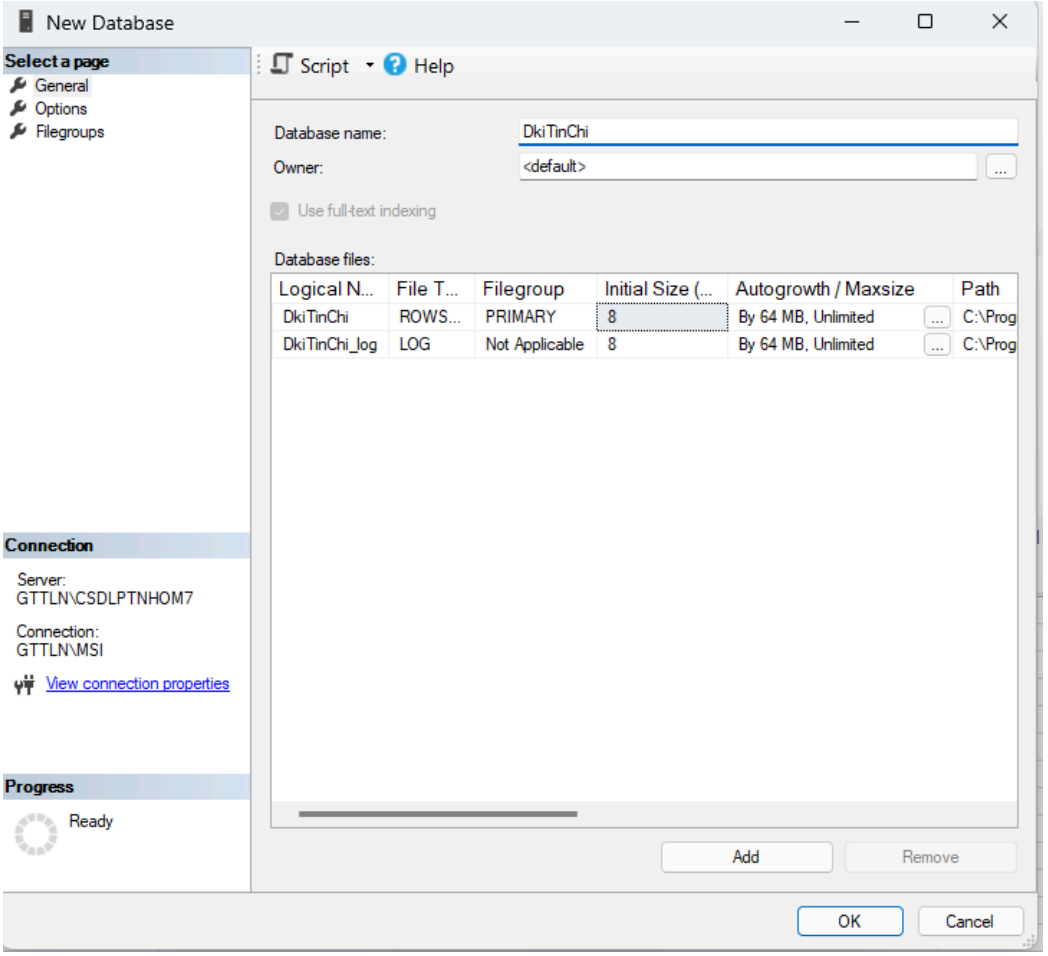
Login: Chọn Enabled.



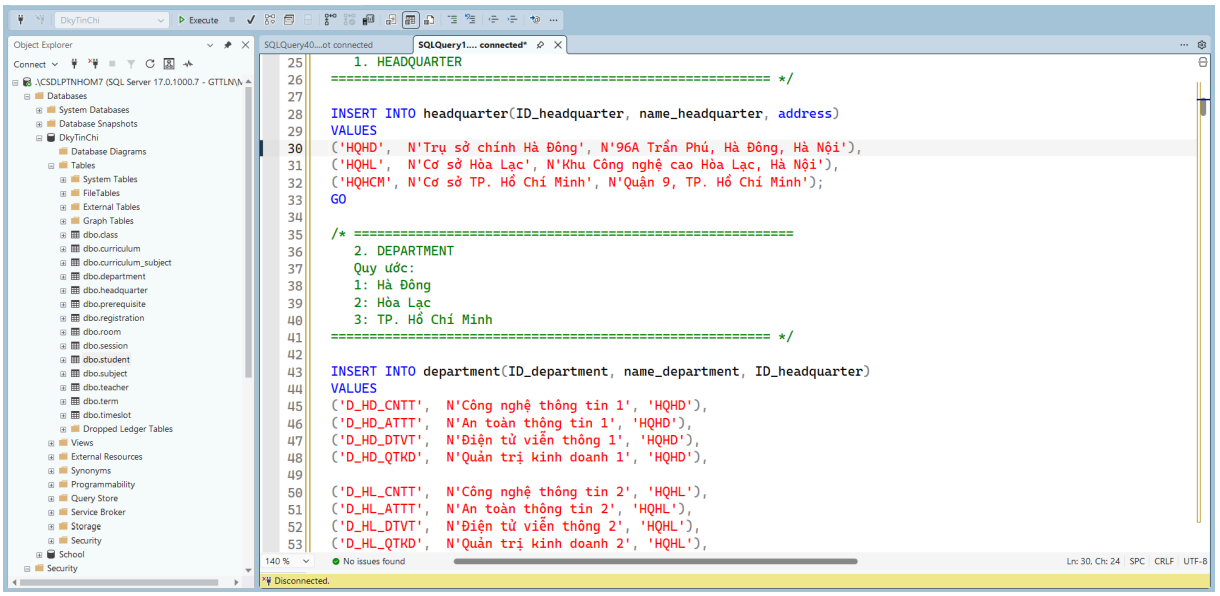
Bước 7: Nhấn OK để hoàn tất quá trình kích hoạt tài khoản hệ thống.

V. Tạo Database DkyTinChi

Bước 1: Tạo Database

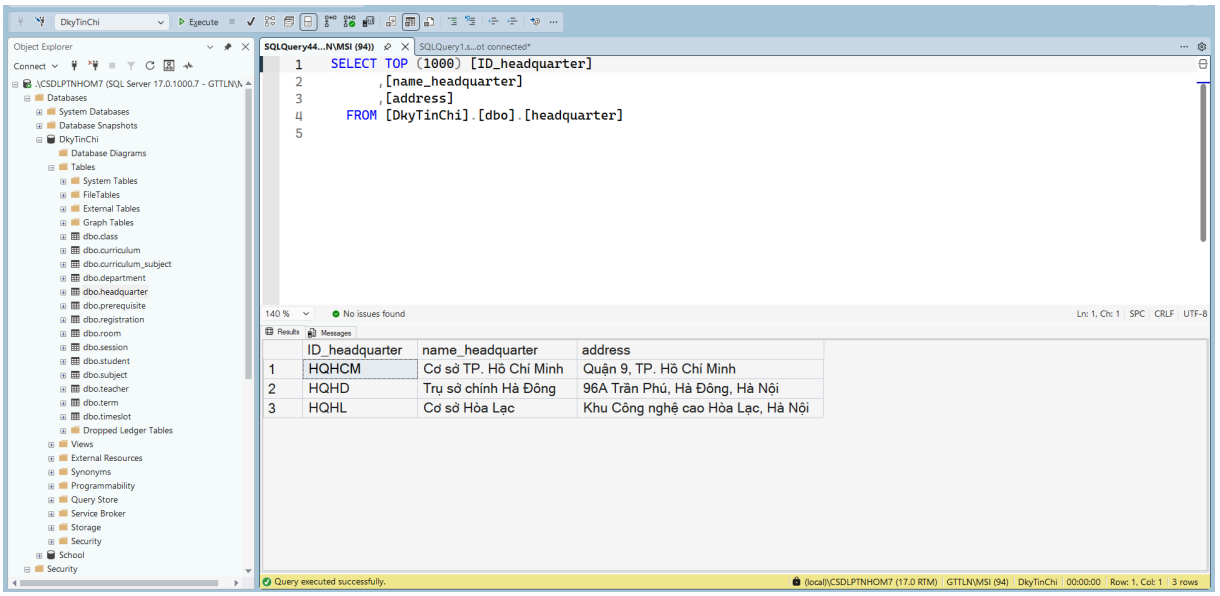


Bước 2: Thêm dữ liệu



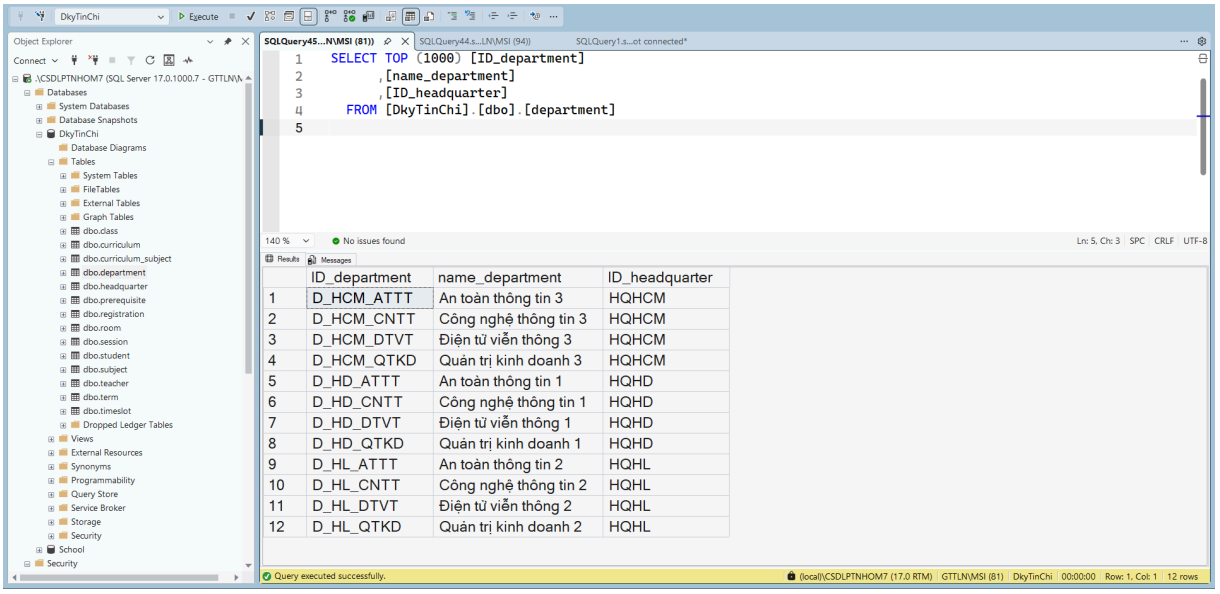
1. Bảng headquarter

Bảng headquarter dùng để lưu thông tin về các trụ sở đào tạo của học viện. Mỗi trụ sở được định danh bằng mã trụ sở ID_headquarter. Ngoài ra, bảng còn lưu tên trụ sở và địa chỉ trụ sở. Đây là bảng dữ liệu gốc để liên kết với các bảng khác như department và room



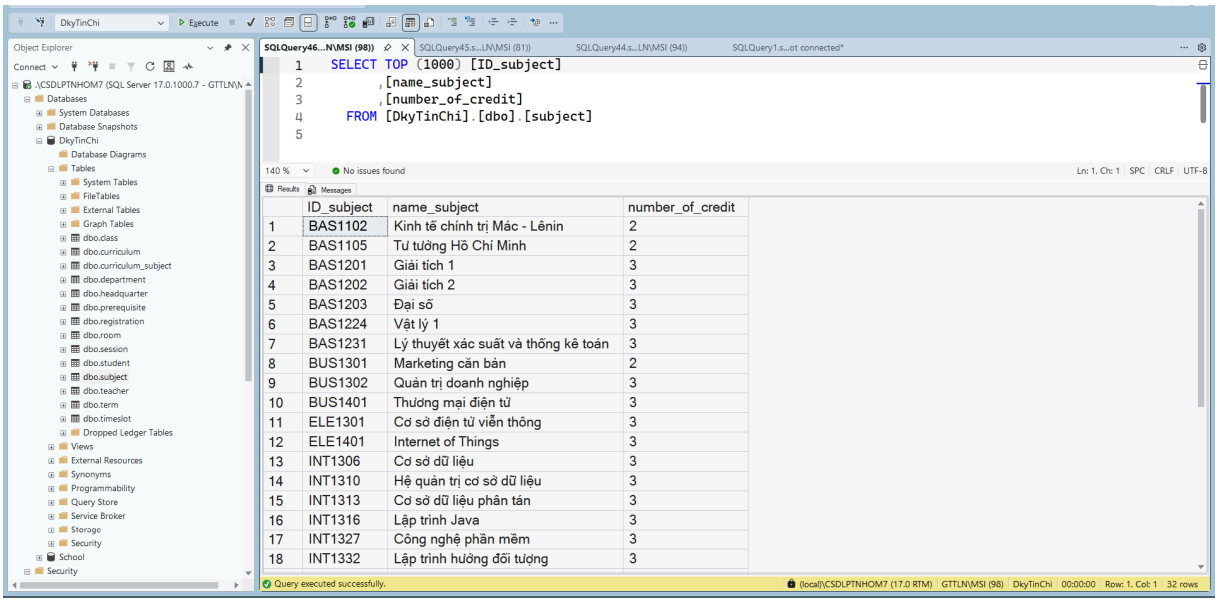
2. Bảng department

Bảng department dùng để quản lý thông tin các khoa thuộc từng trụ sở đào tạo. Mỗi khoa có mã khoa riêng, tên khoa và mã trụ sở tương ứng. Trường ID_headquarter là khóa ngoại tham chiếu đến bảng headquarter, giúp xác định khoa đó thuộc trụ sở nào.



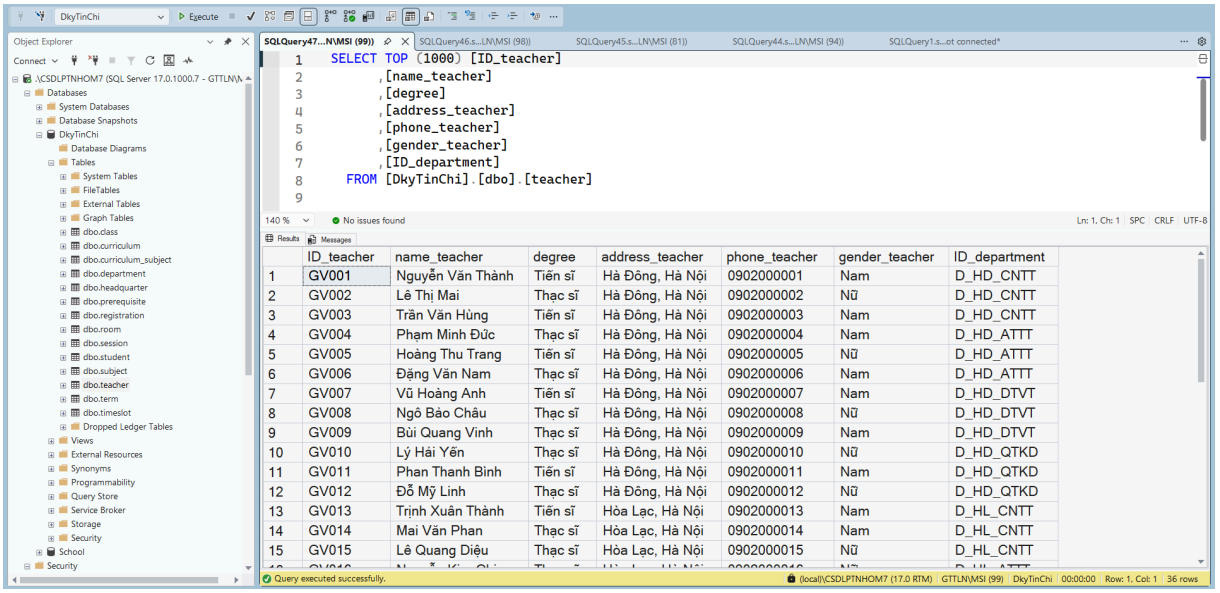
3. Bảng subject

Bảng subject lưu thông tin các môn học trong hệ thống. Mỗi môn học được quản lý bằng mã môn học, tên môn học và số tín chỉ. Bảng này là cơ sở để tạo các lớp học phần trong bảng class.



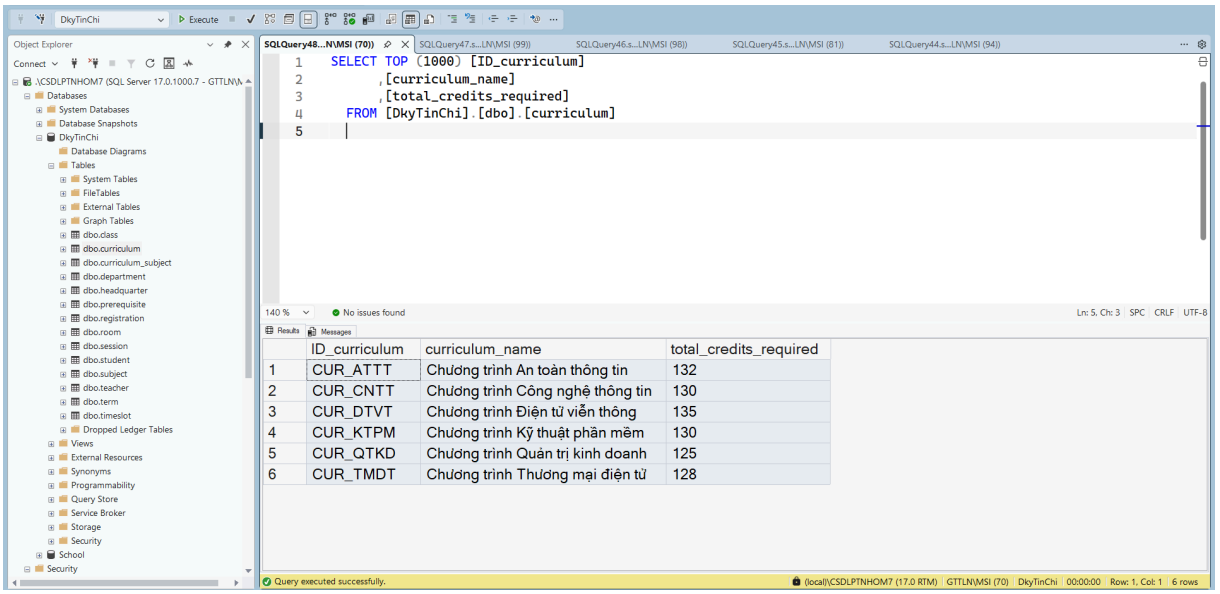
4. Bảng teacher

Bảng teacher dùng để lưu thông tin giảng viên. Dữ liệu trong bảng bao gồm mã giảng viên, họ tên, học vị, địa chỉ, số điện thoại, giới tính và mã khoa mà giảng viên trực thuộc. Trường ID_department giúp liên kết giảng viên với khoa tương ứng.



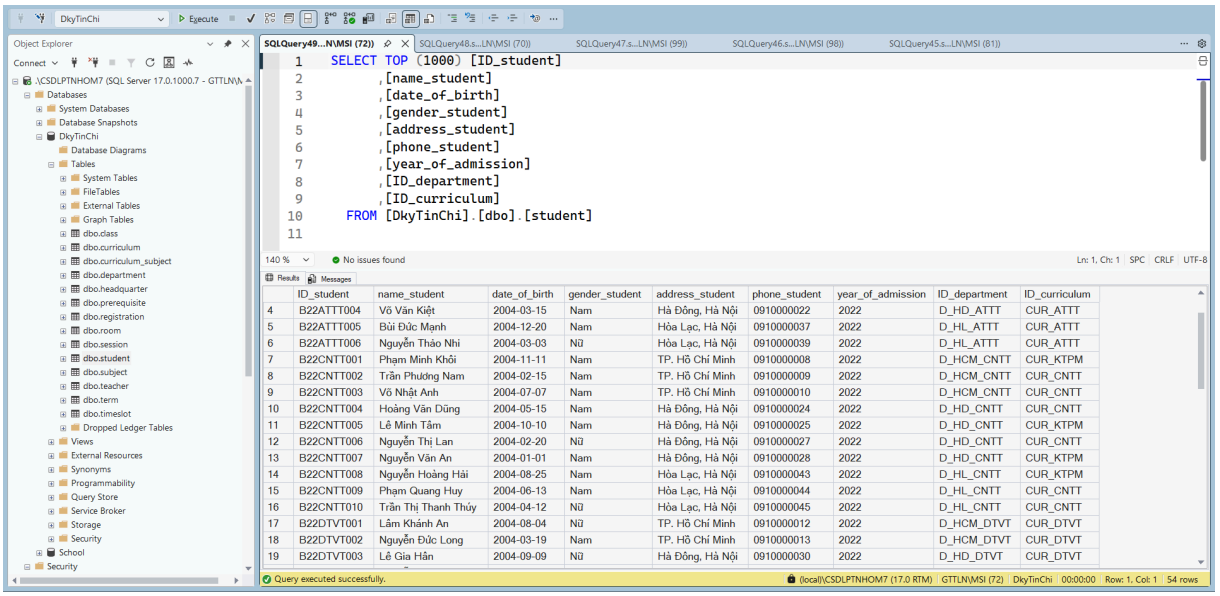
5. Bảng curriculum

Bảng curriculum dùng để lưu thông tin về chương trình đào tạo của từng khoa. Mỗi chương trình đào tạo có mã riêng, tên và tổng số tín chỉ yêu cầu của chương trình đó.



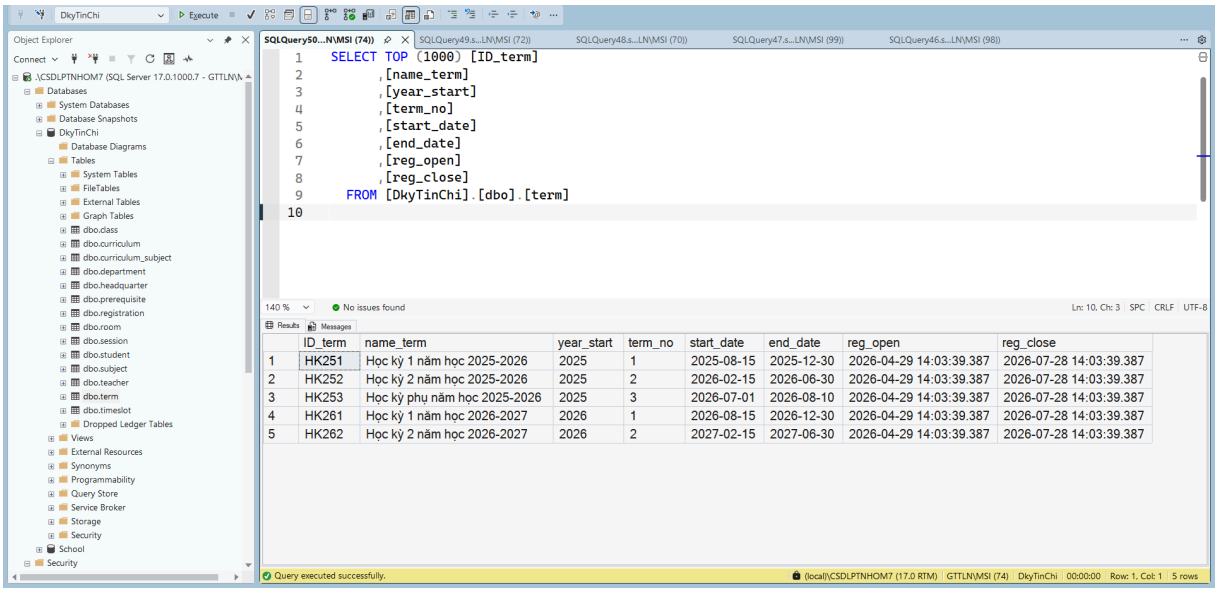
6. Bảng student

Bảng student dùng để quản lý thông tin sinh viên. Mỗi sinh viên có mã sinh viên duy nhất, họ tên, ngày sinh, giới tính, địa chỉ, số điện thoại, năm nhập học, mã khoa và mã chương trình đào tạo. Bảng này liên kết với bảng department để xác định sinh viên thuộc khoa nào và liên kết với bảng curriculum để xác định sinh viên theo học chương trình đào tạo nào.



7. Bảng term

Bảng term dùng để quản lý thông tin học kỳ. Mỗi học kỳ có mã học kỳ, tên học kỳ, năm bắt đầu, số thứ tự học kỳ, ngày bắt đầu, ngày kết thúc, thời gian mở đăng ký và thời gian đóng đăng ký. Bảng này giúp kiểm soát thời gian tổ chức học phần và thời gian sinh viên được phép đăng ký tín chỉ.

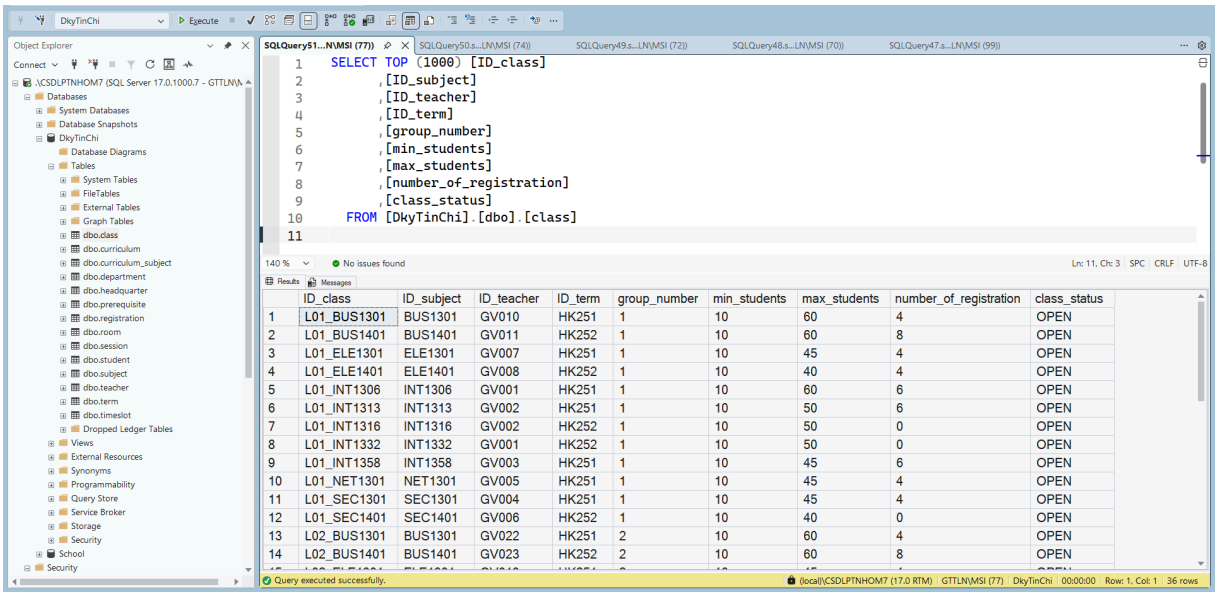


```
SELECT TOP (1000) [ID_term]
        ,[name_term]
        ,[year_start]
        ,[term_no]
        ,[start_date]
        ,[end_date]
        ,[reg_open]
        ,[reg_close]
    FROM [DkyTinChi].[dbo].[term]
```

	ID_term	name_term	year_start	term_no	start_date	end_date	reg_open	reg_close
1	HK251	Học kỳ 1 năm học 2025-2026	2025	1	2025-08-15	2025-12-30	2026-04-29 14:03:39.387	2026-07-28 14:03:39.387
2	HK252	Học kỳ 2 năm học 2025-2026	2025	2	2026-02-15	2026-06-30	2026-04-29 14:03:39.387	2026-07-28 14:03:39.387
3	HK253	Học kỳ phụ năm học 2025-2026	2025	3	2026-07-01	2026-08-10	2026-04-29 14:03:39.387	2026-07-28 14:03:39.387
4	HK261	Học kỳ 1 năm học 2026-2027	2026	1	2026-08-15	2026-12-30	2026-04-29 14:03:39.387	2026-07-28 14:03:39.387
5	HK262	Học kỳ 2 năm học 2026-2027	2026	2	2027-02-15	2027-06-30	2026-04-29 14:03:39.387	2026-07-28 14:03:39.387

8. Bảng class

Bảng class dùng để quản lý các lớp học phần được mở trong từng học kỳ. Mỗi lớp học phần có mã lớp, số lượng sinh viên đăng ký, mã môn học, mã giảng viên, mã học kỳ, mã số nhóm, số lượng sinh viên tối thiểu, số lượng sinh viên tối đa, trạng thái lớp học. Bảng này liên kết với subject, teacher và term để xác định lớp học phần thuộc môn nào, do giảng viên nào phụ trách và được mở trong học kỳ nào.



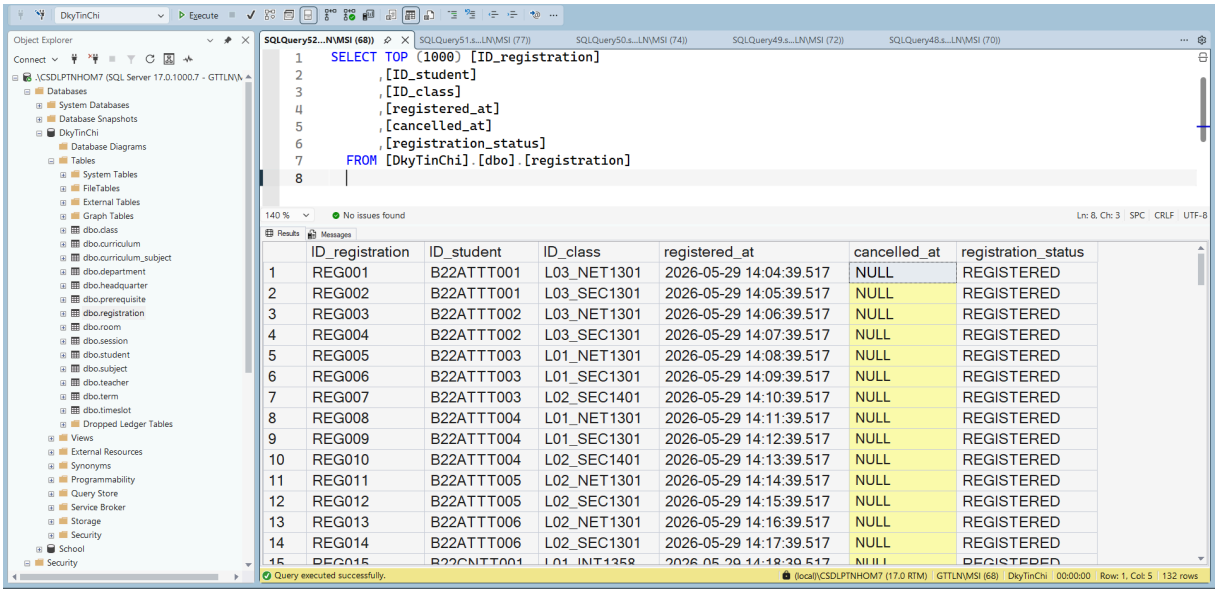
```
SELECT TOP (1000) [ID_class]
        ,[ID_subject]
        ,[ID_teacher]
        ,[ID_term]
        ,[group_number]
        ,[min_students]
        ,[max_students]
        ,[number_of_registration]
        ,[class_status]
    FROM [DkyTinChi].[dbo].[class]
```

	ID_class	ID_subject	ID_teacher	ID_term	group_number	min_students	max_students	number_of_registration	class_status
1	L01_BUS1301	BUS1301	GV010	HK251	1	10	60	4	OPEN
2	L01_BUS1401	BUS1401	GV011	HK252	1	10	60	8	OPEN
3	L01_ELE1301	ELE1301	GV007	HK251	1	10	45	4	OPEN
4	L01_ELE1401	ELE1401	GV008	HK252	1	10	40	4	OPEN
5	L01_INT1306	INT1306	GV001	HK251	1	10	60	6	OPEN
6	L01_INT1313	INT1313	GV002	HK251	1	10	50	6	OPEN
7	L01_INT1316	INT1316	GV002	HK252	1	10	50	0	OPEN
8	L01_INT1332	INT1332	GV001	HK252	1	10	50	0	OPEN
9	L01_INT1358	INT1358	GV003	HK251	1	10	45	6	OPEN
10	L01_NET1301	NET1301	GV005	HK251	1	10	45	4	OPEN
11	L01_SEC1301	SEC1301	GV004	HK251	1	10	45	4	OPEN
12	L01_SEC1401	SEC1401	GV006	HK252	1	10	40	0	OPEN
13	L02_BUS1301	BUS1301	GV022	HK251	2	10	60	4	OPEN
14	L02_BUS1401	BUS1401	GV023	HK252	2	10	60	8	OPEN

9. Bảng registration

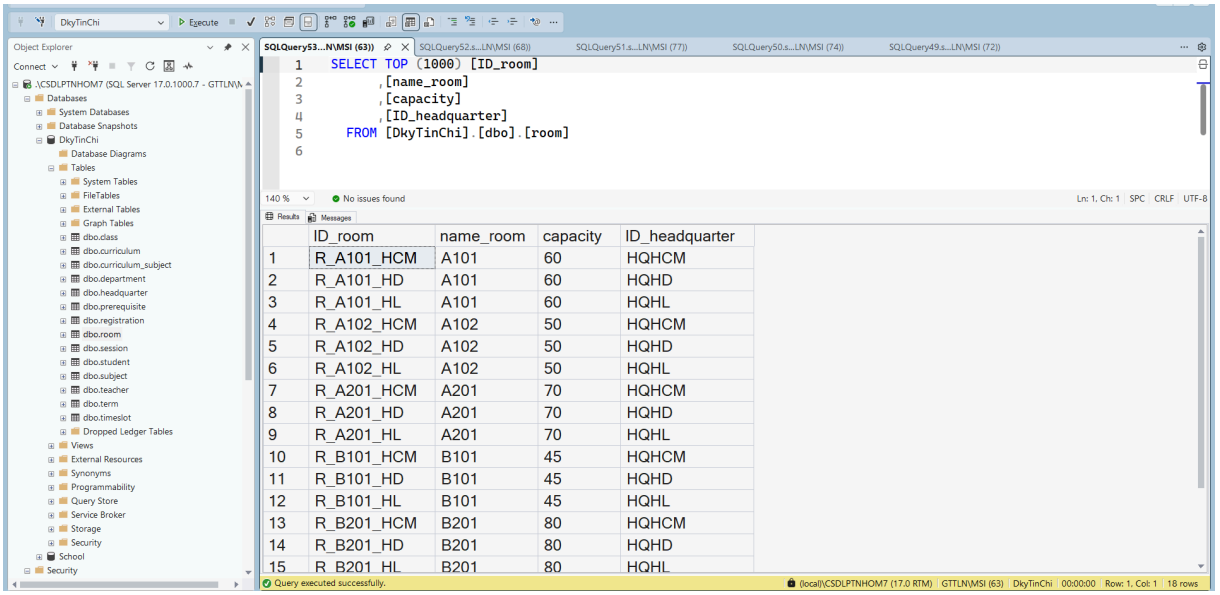
Bảng registration dùng để lưu thông tin đăng ký học phần của sinh viên. Đây là bảng liên kết giữa sinh viên và lớp học phần. Mỗi bản ghi thể hiện một lần sinh viên đăng ký vào một lớp học phần cụ thể, bao gồm mã đăng ký, thời gian đăng ký, mã sinh viên, mã lớp học phần, thời gian hủy đăng kí môn học(nếu có) và trạng thái đăng kí. Bảng này cho phép hệ thống quản lý quan hệ nhiều - nhiều giữa sinh viên và lớp học

phần, vì một sinh viên có thể đăng ký nhiều lớp, và một lớp học phần có thể có nhiều sinh viên đăng ký.



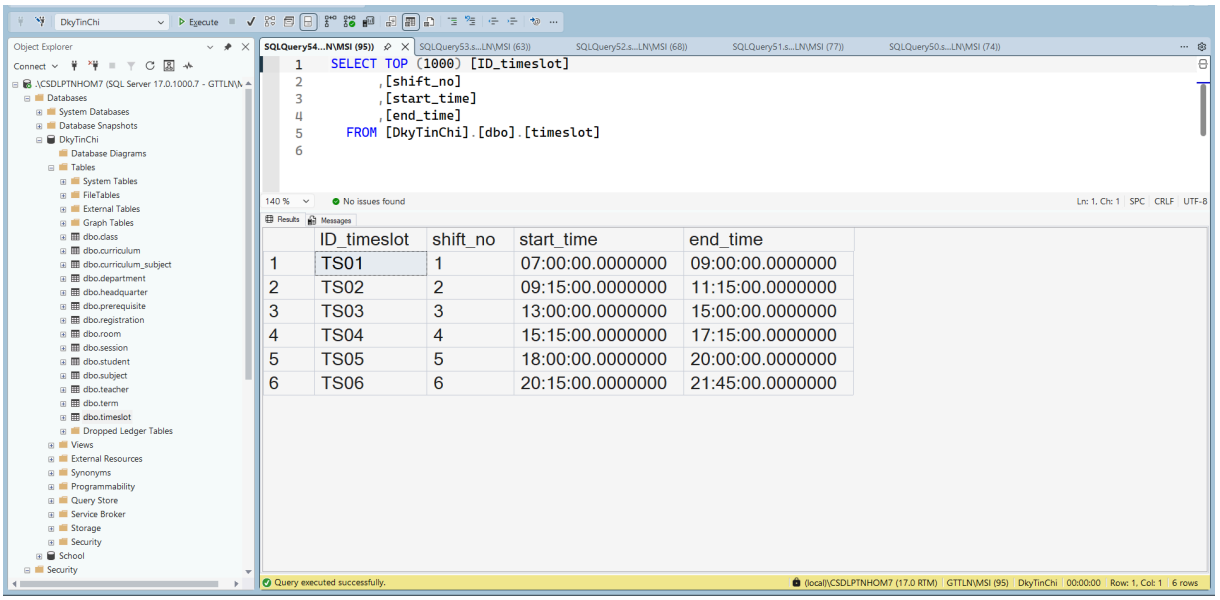
10. Bảng room

Bảng room dùng để lưu thông tin phòng học tại các trụ sở đào tạo. Mỗi phòng học có mã phòng, tên phòng, sức chứa và mã trụ sở. Bảng này giúp hệ thống quản lý tài nguyên phòng học và phục vụ việc lập lịch học.



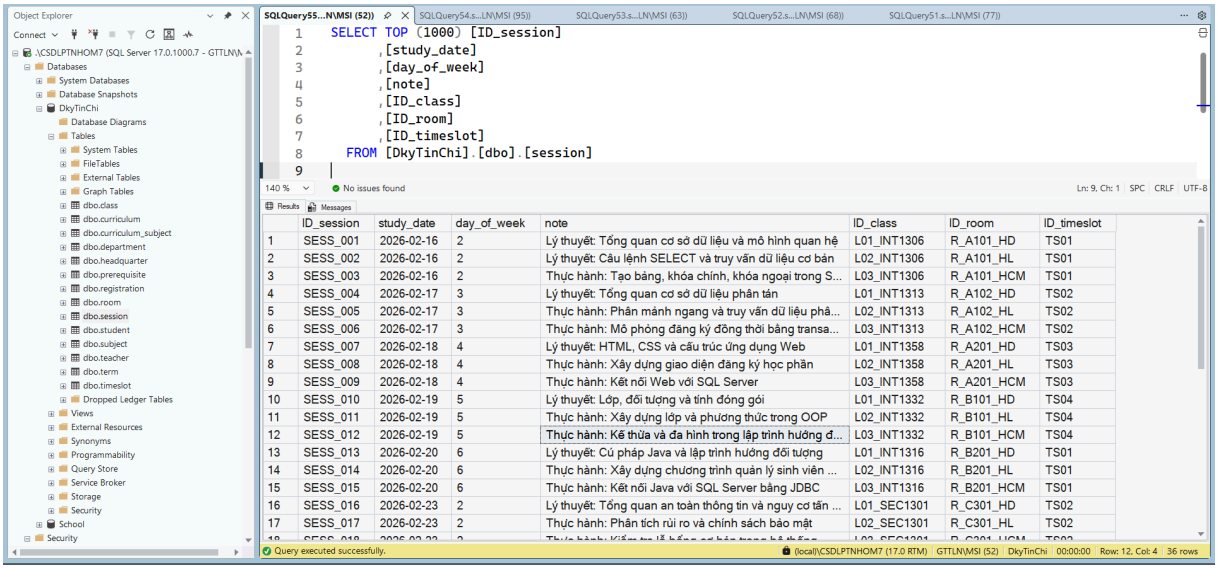
11. Bảng timeslot

Bảng timeslot dùng để quản lý các ca học trong ngày. Mỗi ca học có mã ca, số ca, giờ bắt đầu và giờ kết thúc. Bảng này được sử dụng khi lập lịch cho các buổi học cụ thể trong bảng session.



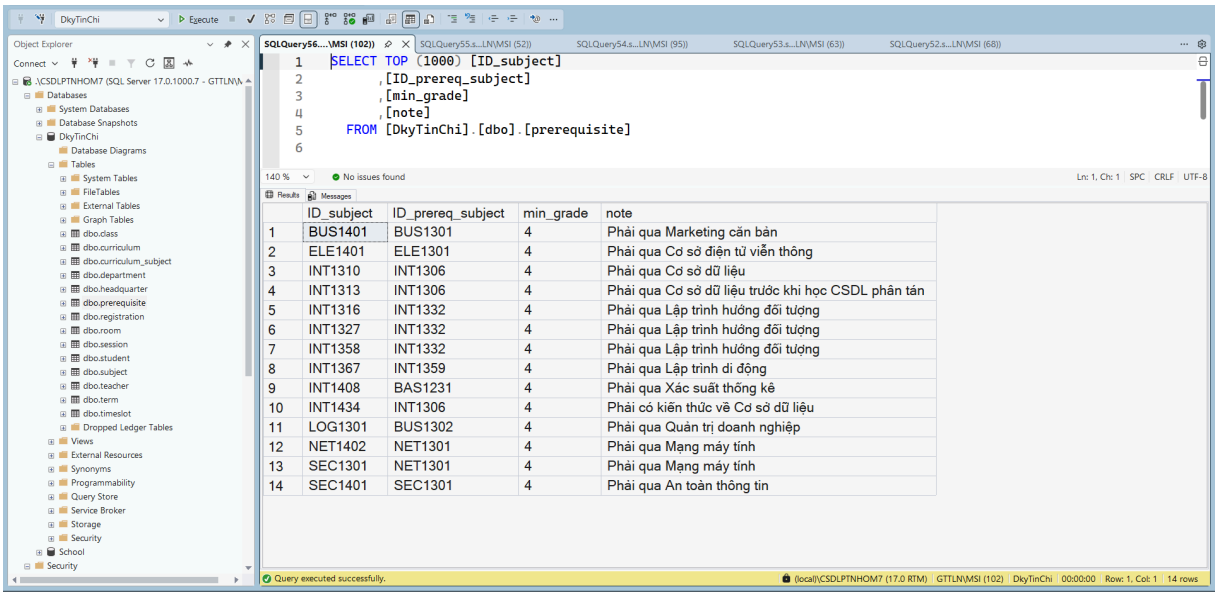
12. Bảng session

Bảng session dùng để lưu thông tin chi tiết về từng buổi học của lớp học phần. Mỗi buổi học có mã buổi học, ngày học, thứ trong tuần, ghi chú, mã lớp học phần, mã phòng học và mã ca học. Bảng này liên kết với class, room và timeslot để xác định buổi học thuộc lớp nào, học ở phòng nào và vào ca nào.



13. Bảng prerequisite

Bảng prerequisite dùng để quản lý điều kiện tiên quyết giữa các môn học. Bảng này cho biết một môn học cần học trước môn nào, điểm tối thiểu cần đạt là bao nhiêu và các ghi chú liên quan. Đây là bảng quan trọng để hỗ trợ kiểm tra điều kiện đăng ký học phần của sinh viên.

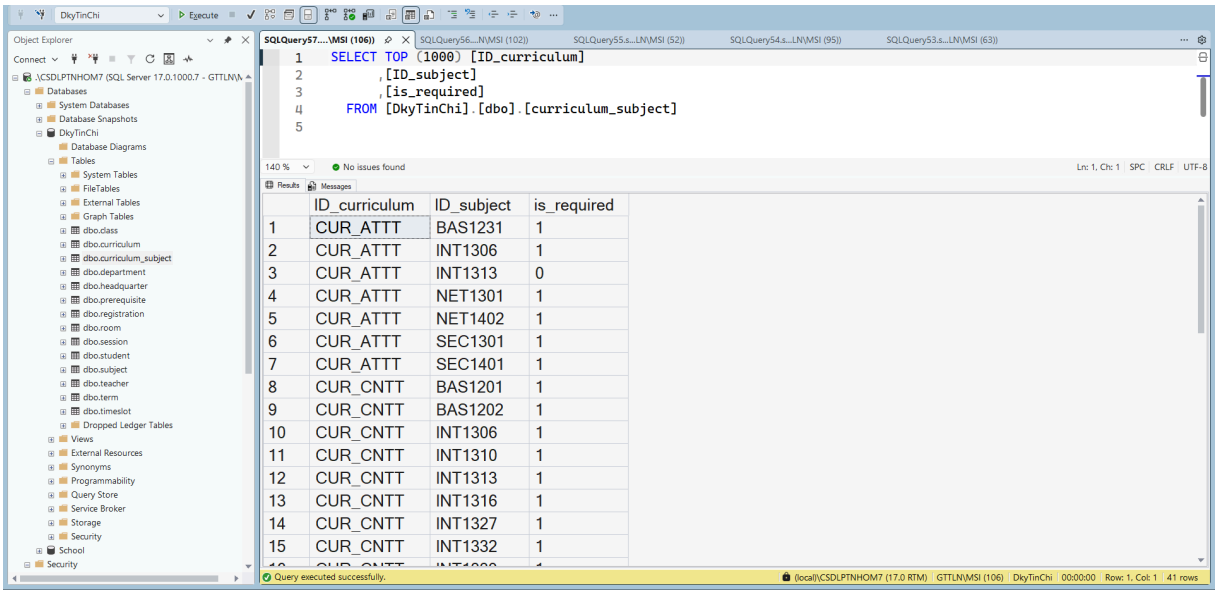


```
SELECT TOP (1000) [ID_subject]
        ,[ID_prereq_subject]
        ,[min_grade]
        ,[note]
    FROM [DkyTinChi].[dbo].[prerequisite]
```

	ID_subject	ID_prereq_subject	min_grade	note
1	BUS1401	BUS1301	4	Phải qua Marketing căn bản
2	ELE1401	ELE1301	4	Phải qua Cơ sở điện tử viễn thông
3	INT1310	INT1306	4	Phải qua Cơ sở dữ liệu
4	INT1313	INT1306	4	Phải qua Cơ sở dữ liệu trước khi học CSDL phân tán
5	INT1316	INT1332	4	Phải qua Lập trình hướng đối tượng
6	INT1327	INT1332	4	Phải qua Lập trình hướng đối tượng
7	INT1358	INT1332	4	Phải qua Lập trình hướng đối tượng
8	INT1367	INT1359	4	Phải qua Lập trình di động
9	INT1408	BAS1231	4	Phải qua Xác suất thống kê
10	INT1434	INT1306	4	Phải có kiến thức về Cơ sở dữ liệu
11	LOG1301	BUS1302	4	Phải qua Quản trị doanh nghiệp
12	NET1402	NET1301	4	Phải qua Mạng máy tính
13	SEC1301	NET1301	4	Phải qua Mạng máy tính
14	SEC1401	SEC1301	4	Phải qua An toàn thông tin

14. Bảng curriculum_subject

Bảng curriculum_subject là bảng trung gian dùng để thể hiện mối quan hệ giữa chương trình đào tạo và môn học. Một chương trình đào tạo có thể gồm nhiều môn học, đồng thời một môn học cũng có thể thuộc nhiều chương trình đào tạo khác nhau. Trường is_required cho biết môn học đó là bắt buộc hay không bắt buộc trong chương trình đào tạo.



```
SELECT TOP (1000) [ID_curriculum]
        ,[ID_subject]
        ,[is_required]
    FROM [DkyTinChi].[dbo].[curriculum_subject]
```

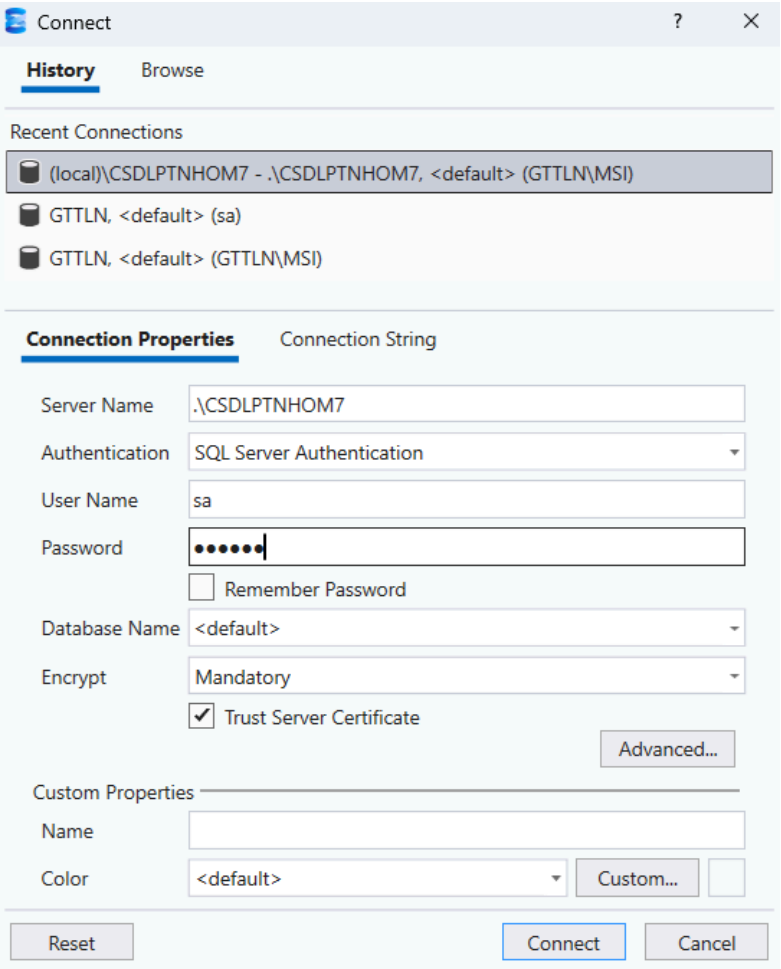
	ID_curriculum	ID_subject	is_required
1	CUR_ATTT	BAS1231	1
2	CUR_ATTT	INT1306	1
3	CUR_ATTT	INT1313	0
4	CUR_ATTT	NET1301	1
5	CUR_ATTT	NET1402	1
6	CUR_ATTT	SEC1301	1
7	CUR_ATTT	SEC1401	1
8	CUR_CNTT	BAS1201	1
9	CUR_CNTT	BAS1202	1
10	CUR_CNTT	INT1306	1
11	CUR_CNTT	INT1310	1
12	CUR_CNTT	INT1313	1
13	CUR_CNTT	INT1316	1
14	CUR_CNTT	INT1327	1
15	CUR_CNTT	INT1332	1

VI. Configure distribution

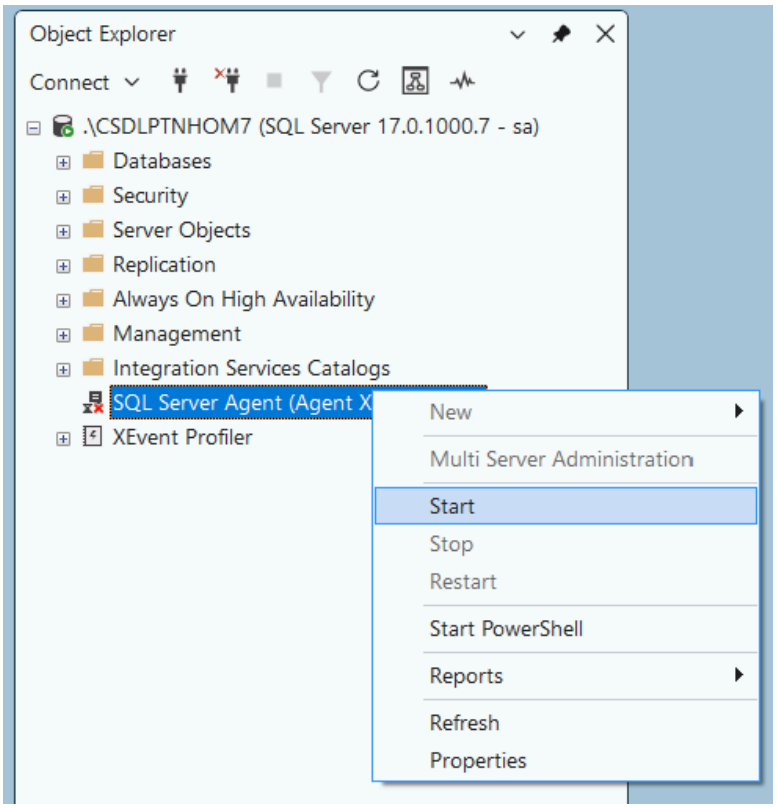
Distributor: Là một server mà chứa CSDL phân tán (distribution database) và lưu trữ metadata,history data và transaction. SQL Server sử dụng CSDL phân tán để lưu và chuyển(store_and_forward) dữ liệu nhân bản từ Publisher đến các Subscriber. Có 2 loại Distributor : Local Distributor và remote Distributor.

1. Đăng nhập và bật SQL Server Agent

Bước 1: Mở SQL Server và thực hiện việc đăng nhập

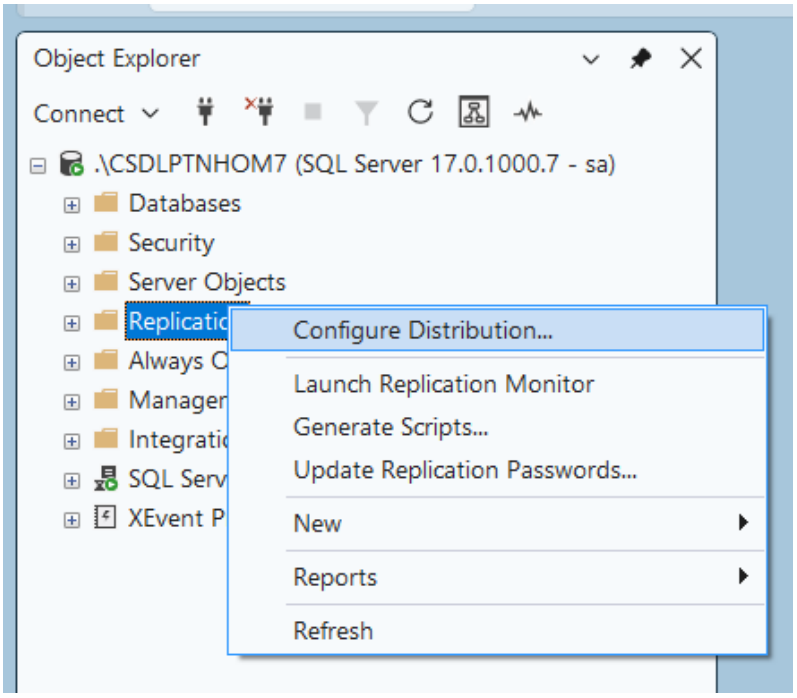


Bước 2: Sau khi đăng nhập, chọn mục SQL Server Agent => Click chuột phải =>Chọn start

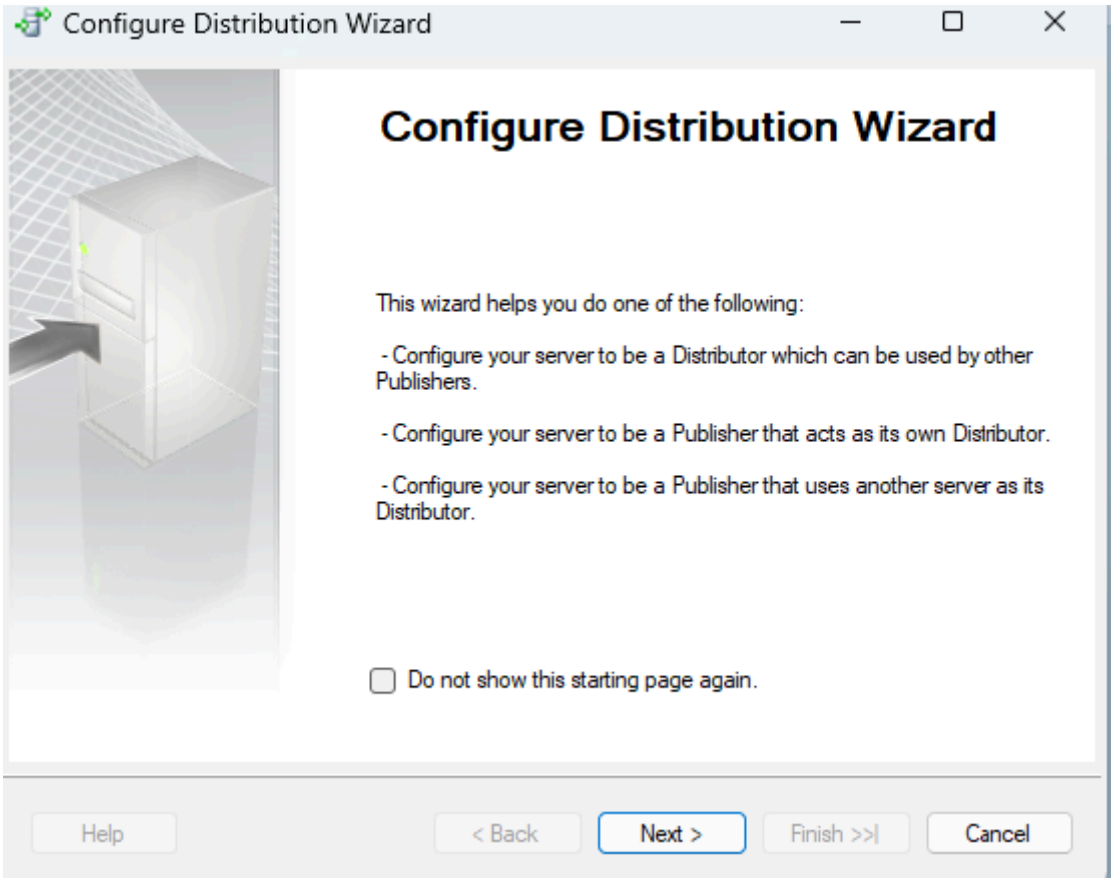


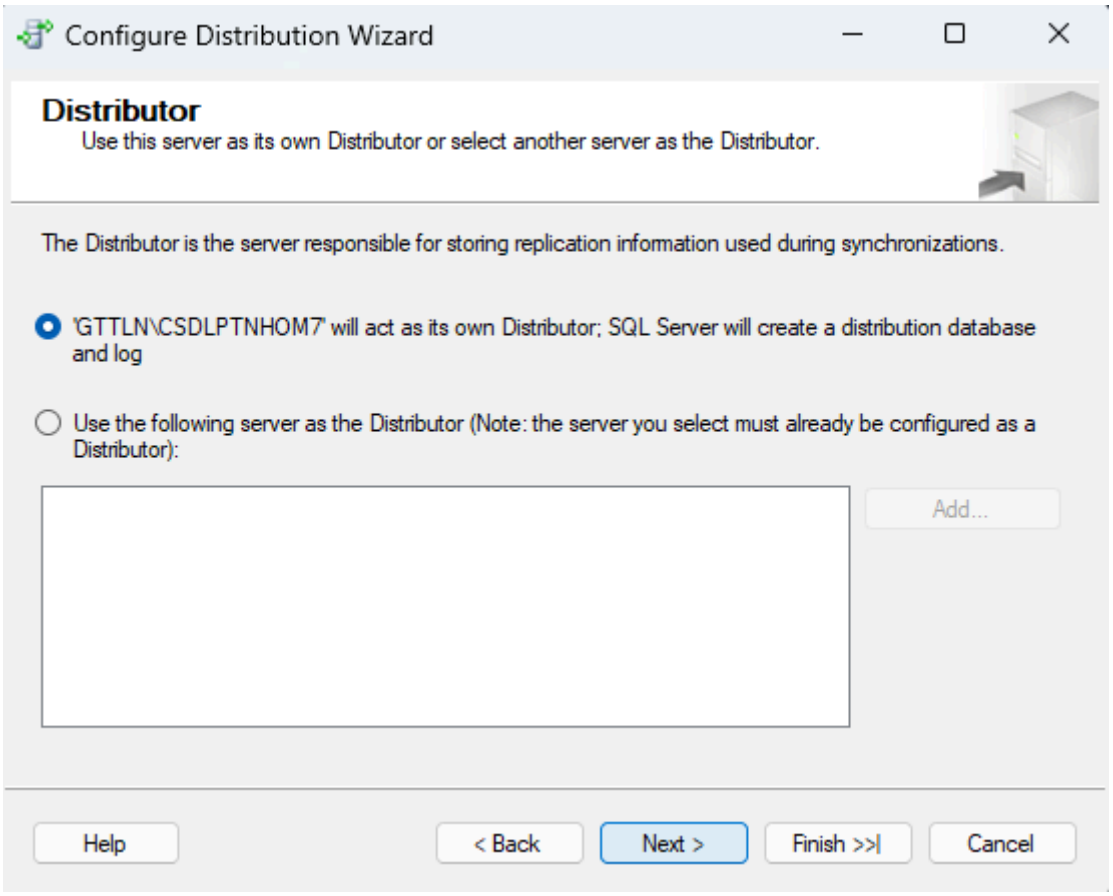
2.Confiruge distribution

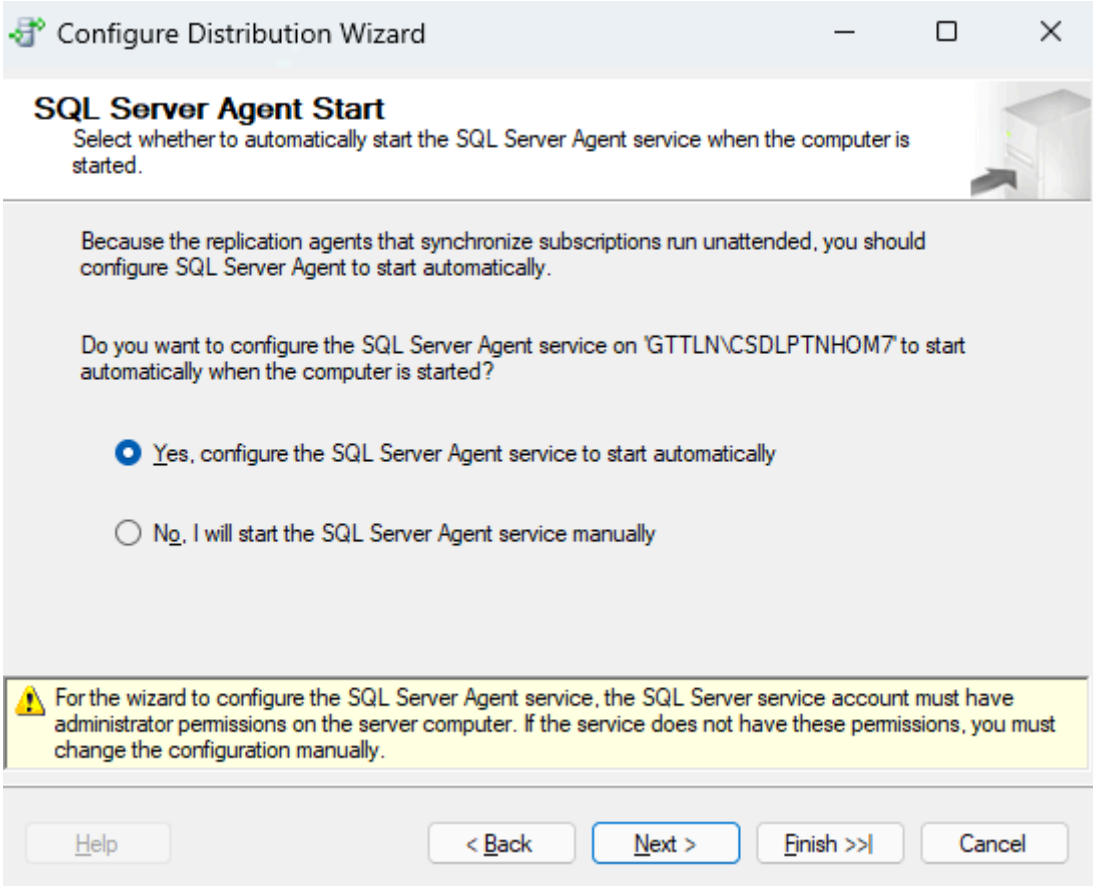
Bước 1: Sau khi chạy Agent,tiếp theo chọn mục Replication => Click chuột phải chọn Configure Distribution



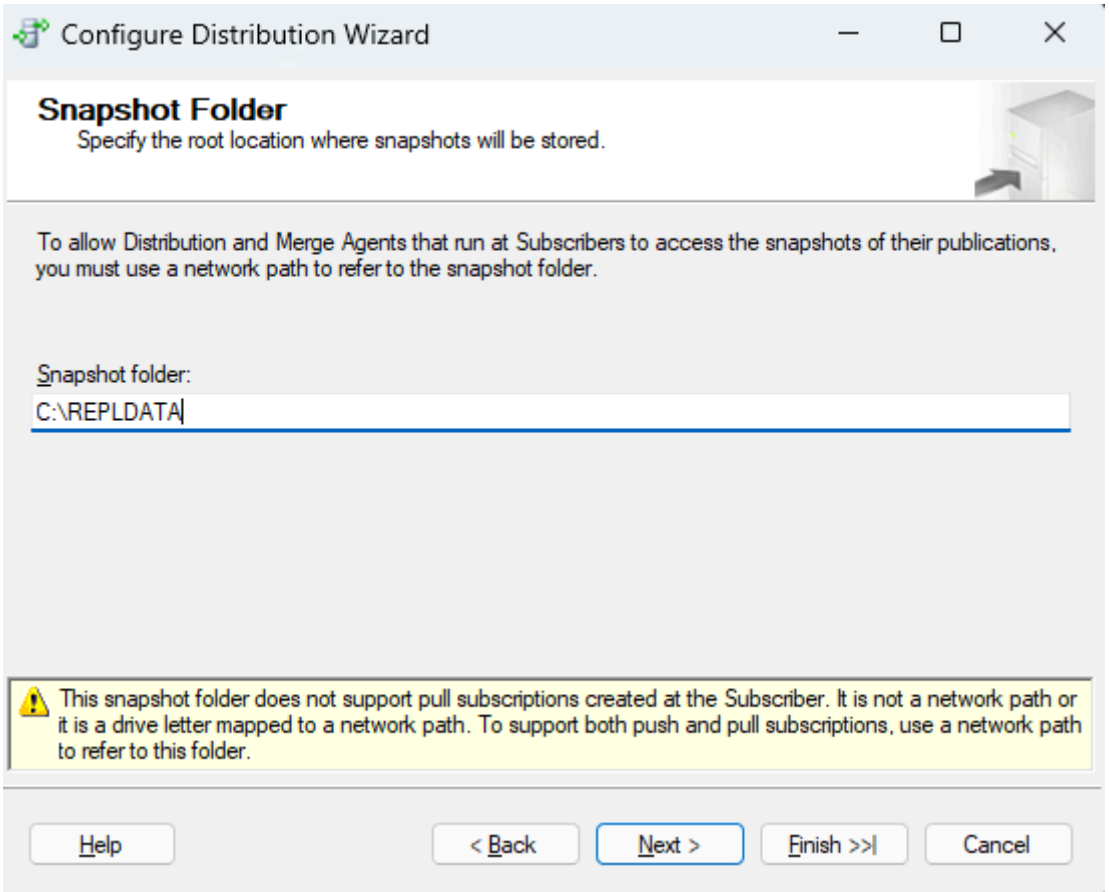
Bước 2: Tiếp tục thực hiện các thao tác Next đến Snapshot Folder


Configure Distribution Wizard
Configure Distribution Wizard
This wizard helps you do one of the following:
- Configure your server to be a Distributor which can be used by other Publishers.
- Configure your server to be a Publisher that acts as its own Distributor.
- Configure your server to be a Publisher that uses another server as its Distributor.
Do not show this starting page again.
Help
< Back
Next >
Finish >>|
Cancel

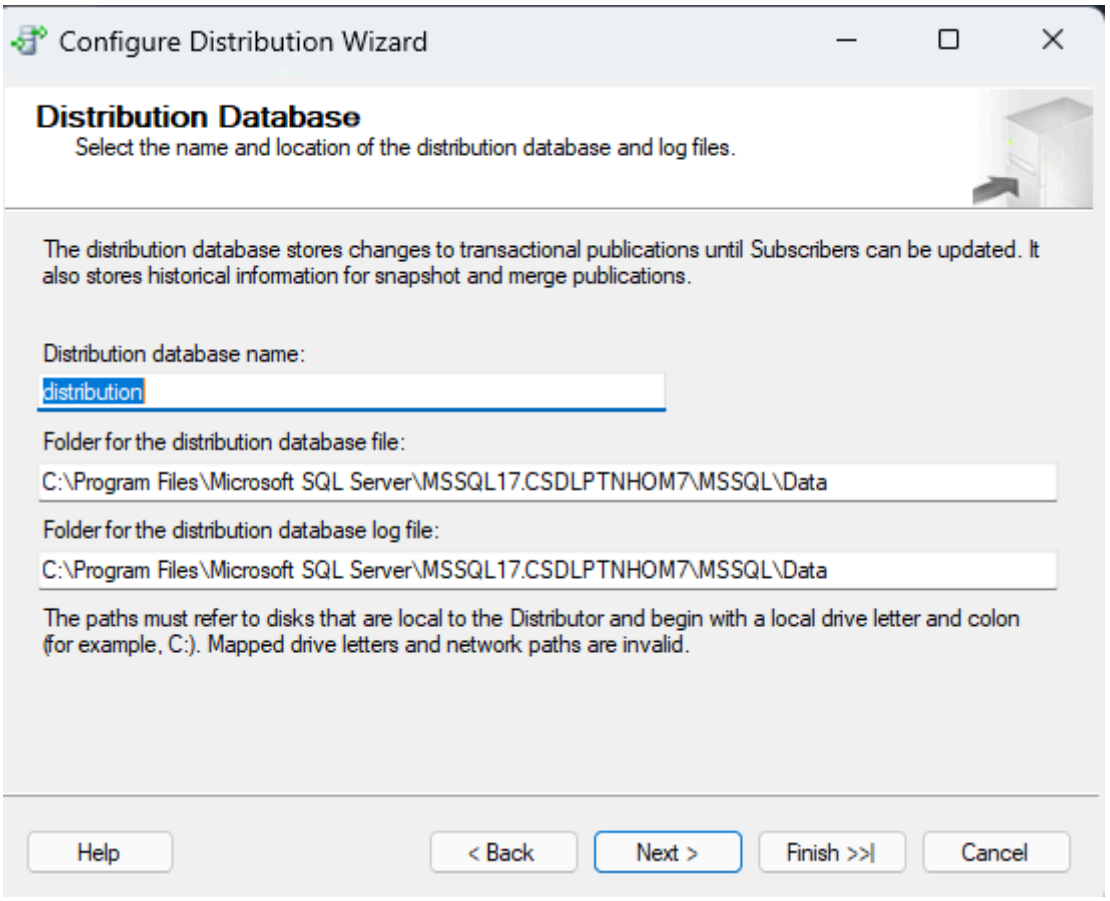

Configure Distribution Wizard
Distributor
Use this server as its own Distributor or select another server as the Distributor.
The Distributor is the server responsible for storing replication information used during synchronizations.
'GTTLN\CSDLPTNHOM7' will act as its own Distributor; SQL Server will create a distribution database and log
Use the following server as the Distributor (Note: the server you select must already be configured as a Distributor):
Add...
Help
< Back
Next >
Finish >>|
Cancel

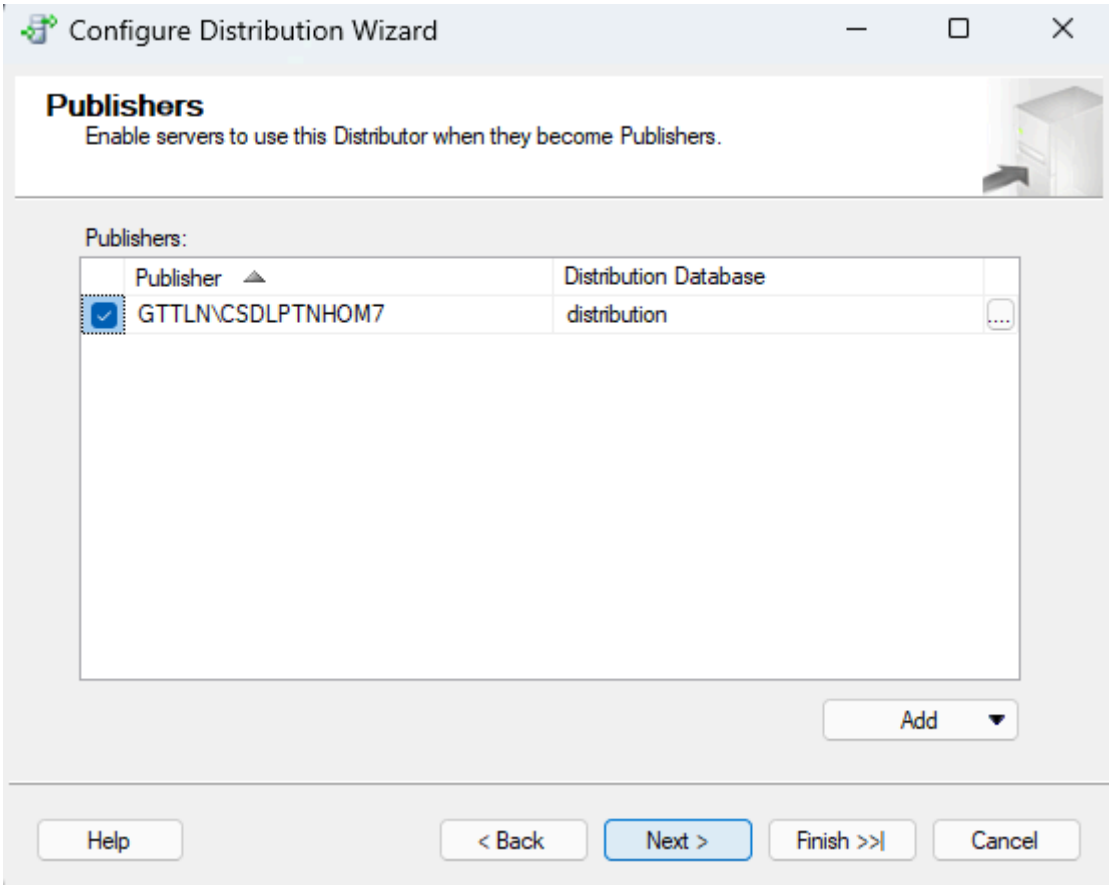


Bước 3: Điền đường dẫn của folder REPTDATA tạo trước đó -> ấn Next

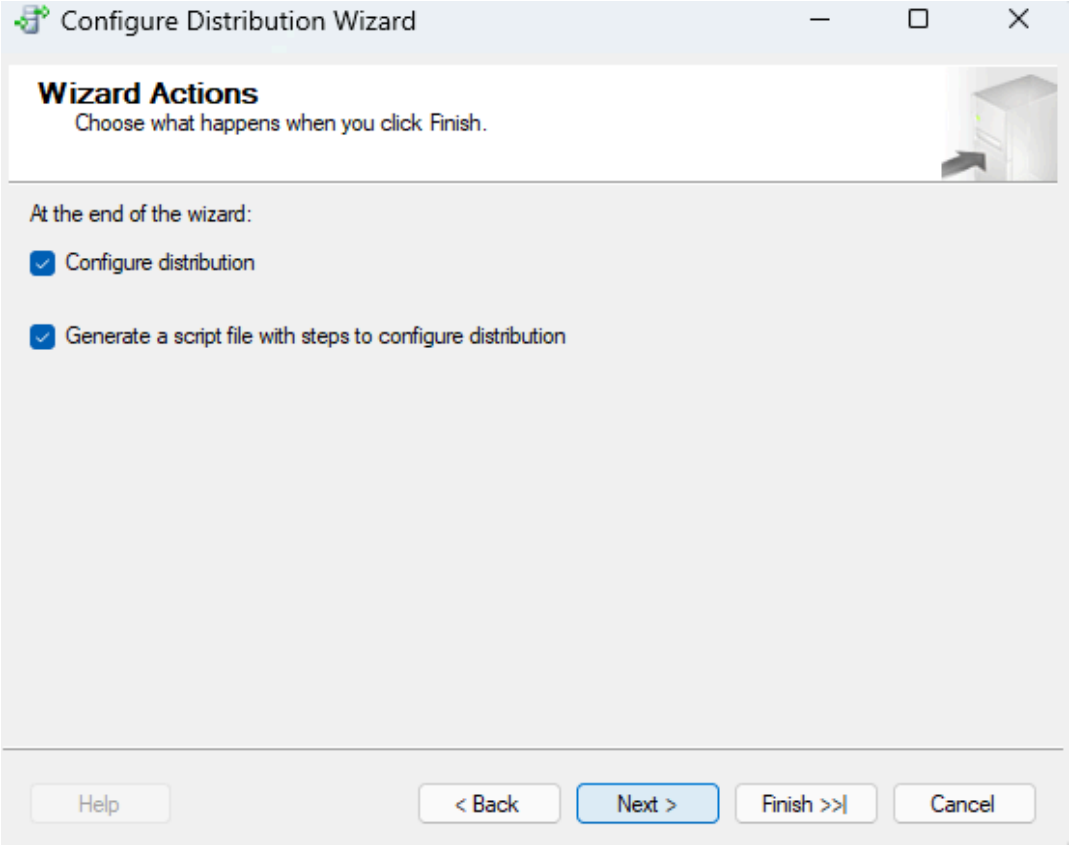


Bước 4: Tiếp tục ấn Next

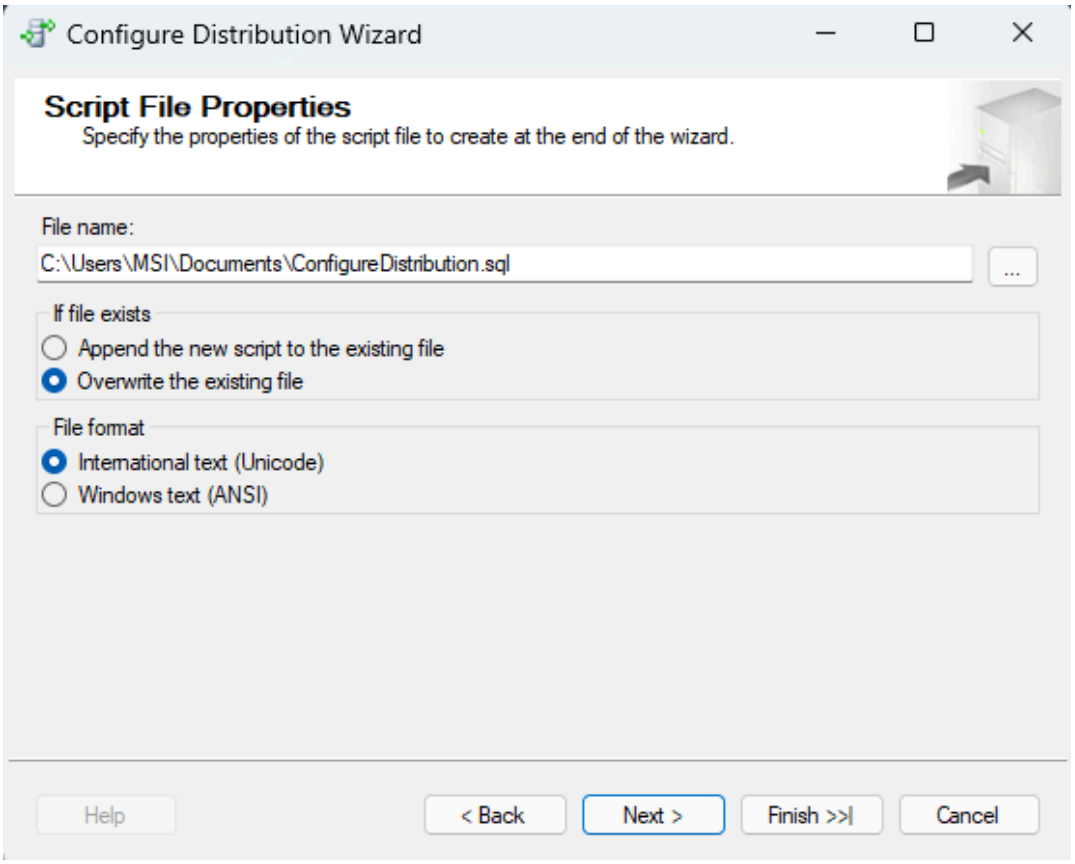




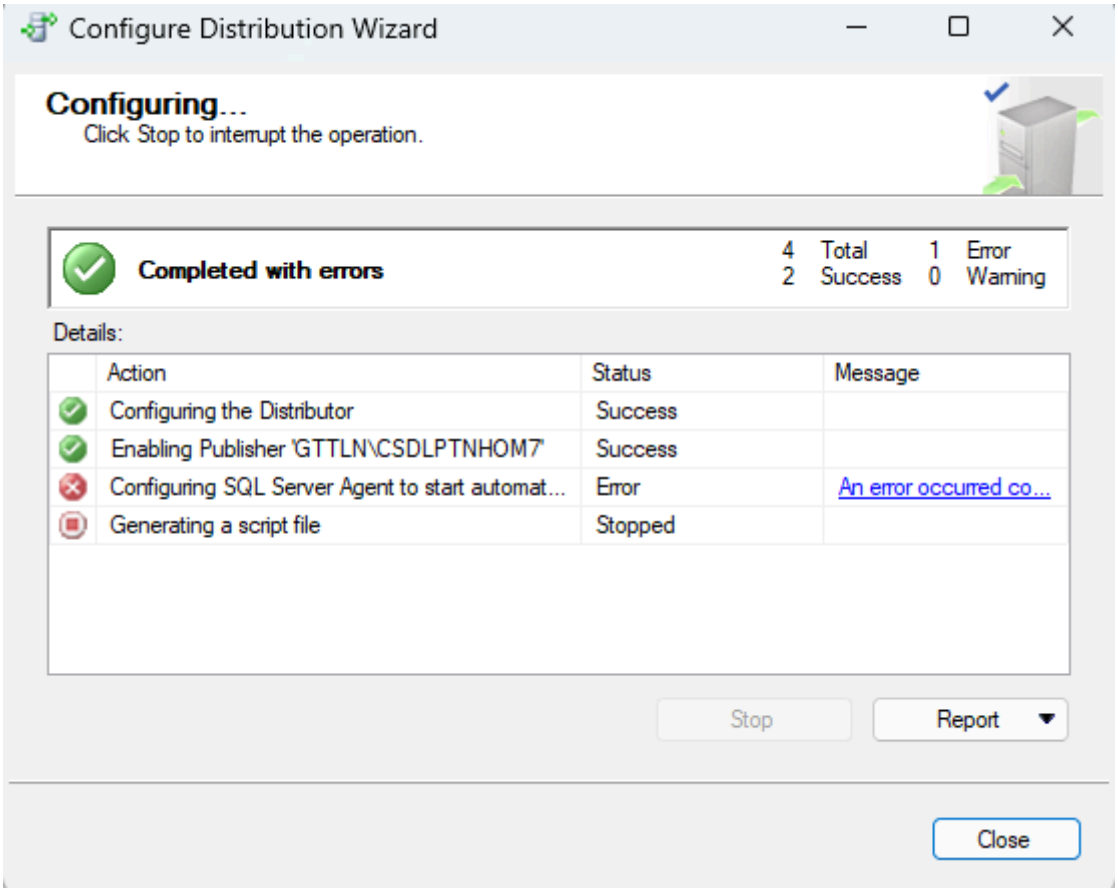
Bước 5: Chọn 2 mục bên dưới => bấm Next



Bước 6: Chọn các mục tương ứng sau đây (không bắt buộc y hệt) => bấm Next



Bước 7: Cuối cùng ấn Finish để hoàn thành Configure



Đỏ 2 chỗ trên là không sao , do không cho SQL Server Agent tự động chạy mỗi khi mở SQL Server nên mới báo lỗi,tuy nhiên đến bước này là config thành công rồi.

VII. Tạo Publication Database

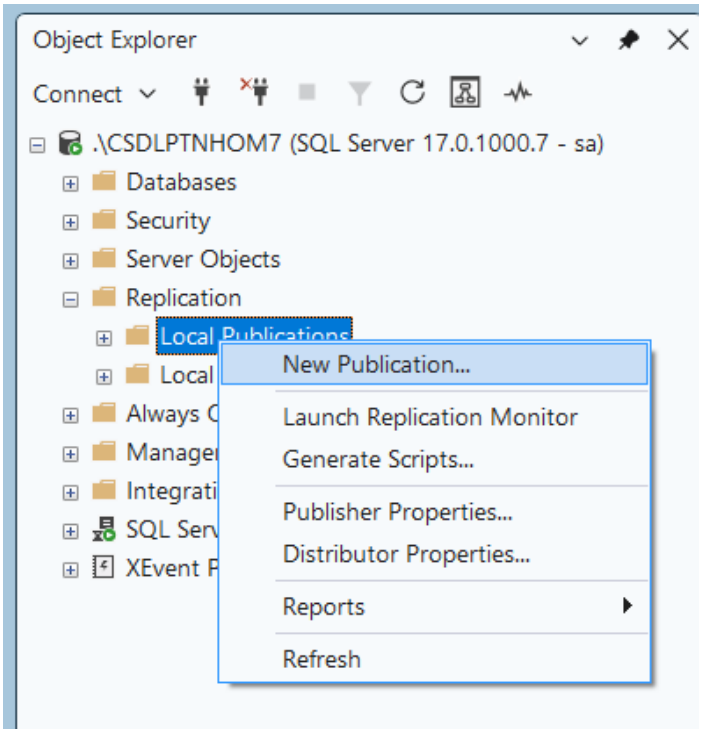
Replication – Nhân bản dữ liệu: Replication là công nghệ sao chép, phân phối dữ liệu và là một trong những giải pháp 'khôi phục dữ liệu sau thảm họa' có sẵn trong SSMS rất hữu ích để duy trì bản sao thứ hai hoặc bản sao dự phòng của các đối tượng (Table, View, Stored Procedure) và CSDL. Replication trong Microsoft SQL Server có 4 loại sau:

+ Snapshot Replication
+ Transactional Replication
+ Peer-to-Peer Replication
+ Merge Replication

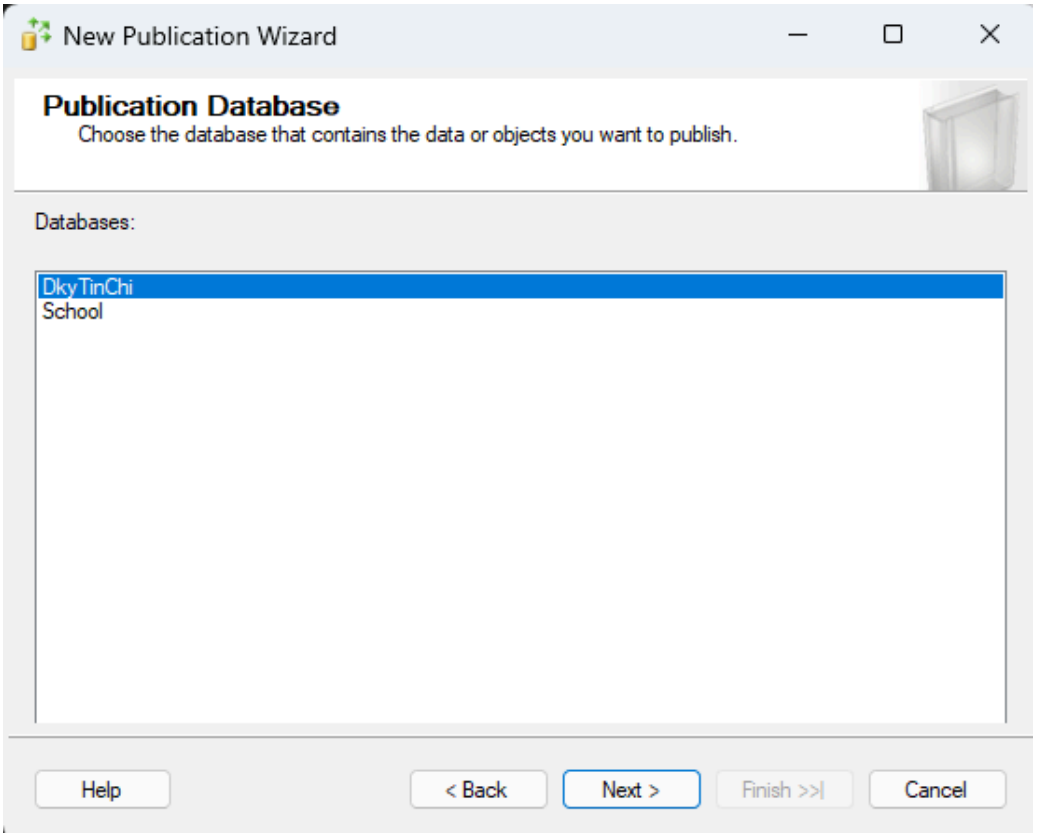
1. Tạo Transactional Replication

Giới thiệu: Nhân bản giao dịch một chiều là một kỹ thuật được sử dụng để sao chép tự động liên tục hoặc định kỳ các dữ liệu giao dịch (Transaction Data) một chiều từ Publisher đến Subscriber.

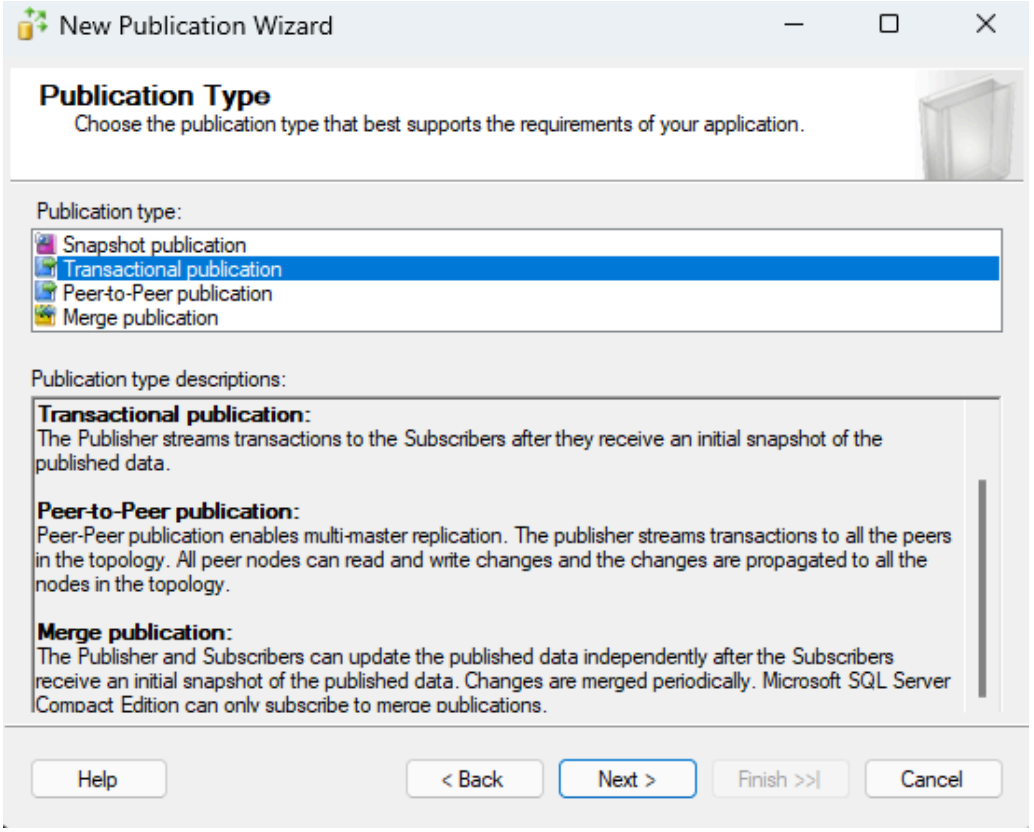
Bước 1: Trong mục Replication tìm tới Local Publications . Chuột phải vào Local Publication -> chọn New Publication…



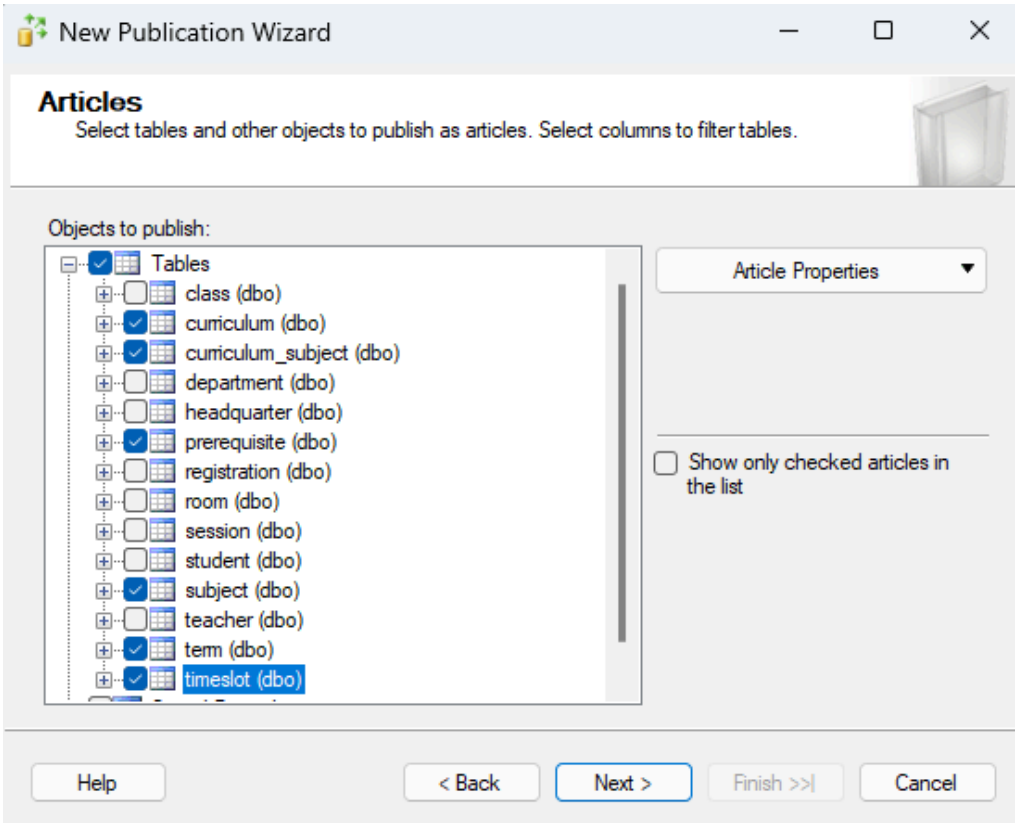
Bước 2: Cửa sổ New Publication Wizard hiện lên. Chọn Database chứa các bảng muốn nhân bản. Click next



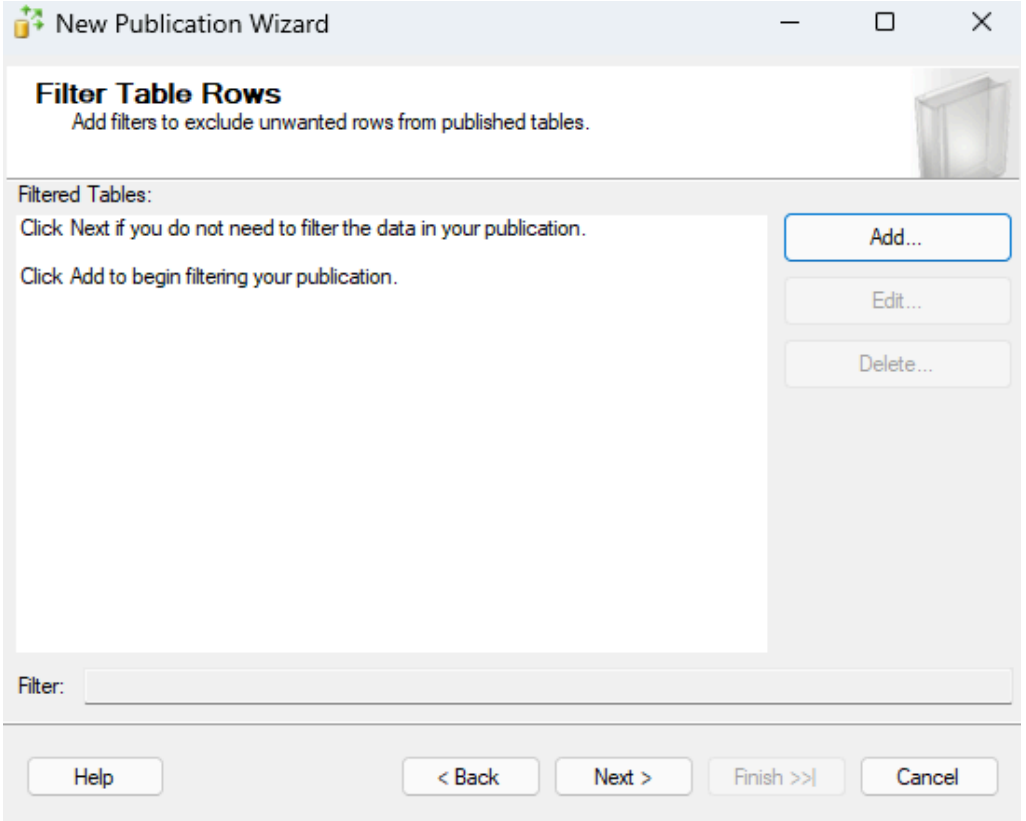
Bước 3: Chọn Transactional publication để cho phép nhân bản -> Chọn Next



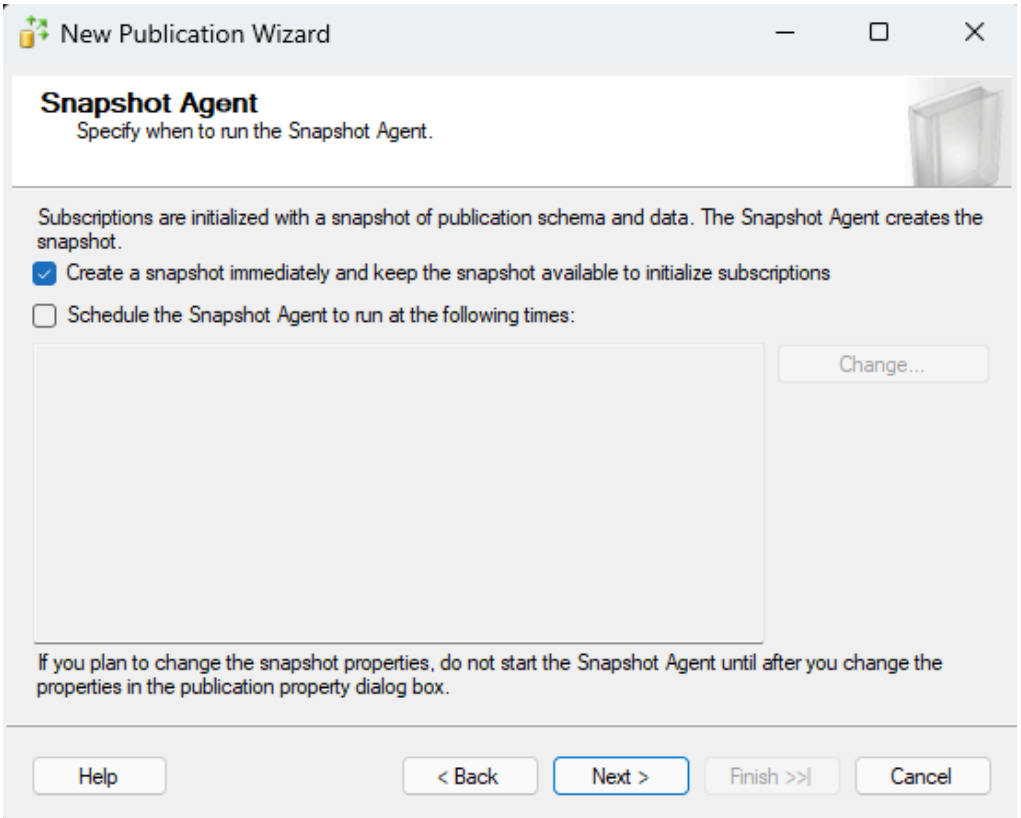
Bước 4: Mở rộng mục table để xem các bảng có trong database. Tích vào bảng muốn nhân bản, ở đây là các bảng curriculum, curriculum_subject, subject, term, prerequisite, timeslot -> click next



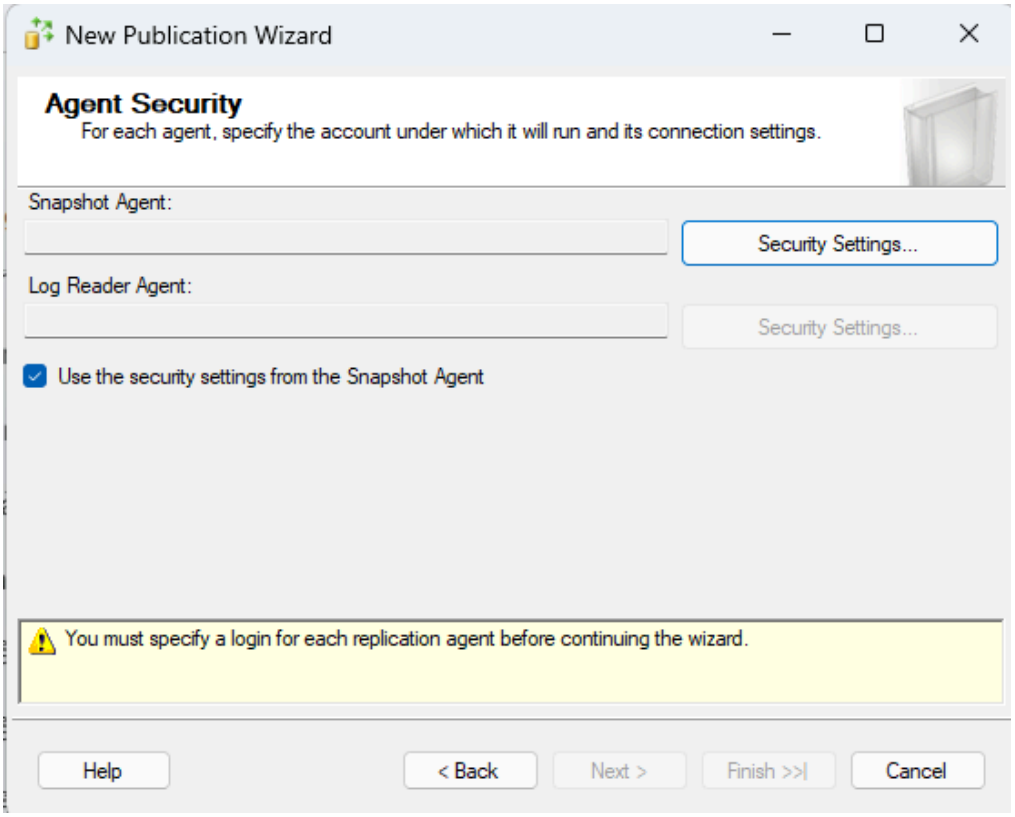
Bước 5: Click Next



Bước 6: Tích chọn “Create a snapshot immediately and keep the snapshot available to initialize subscriptions”. Click Next



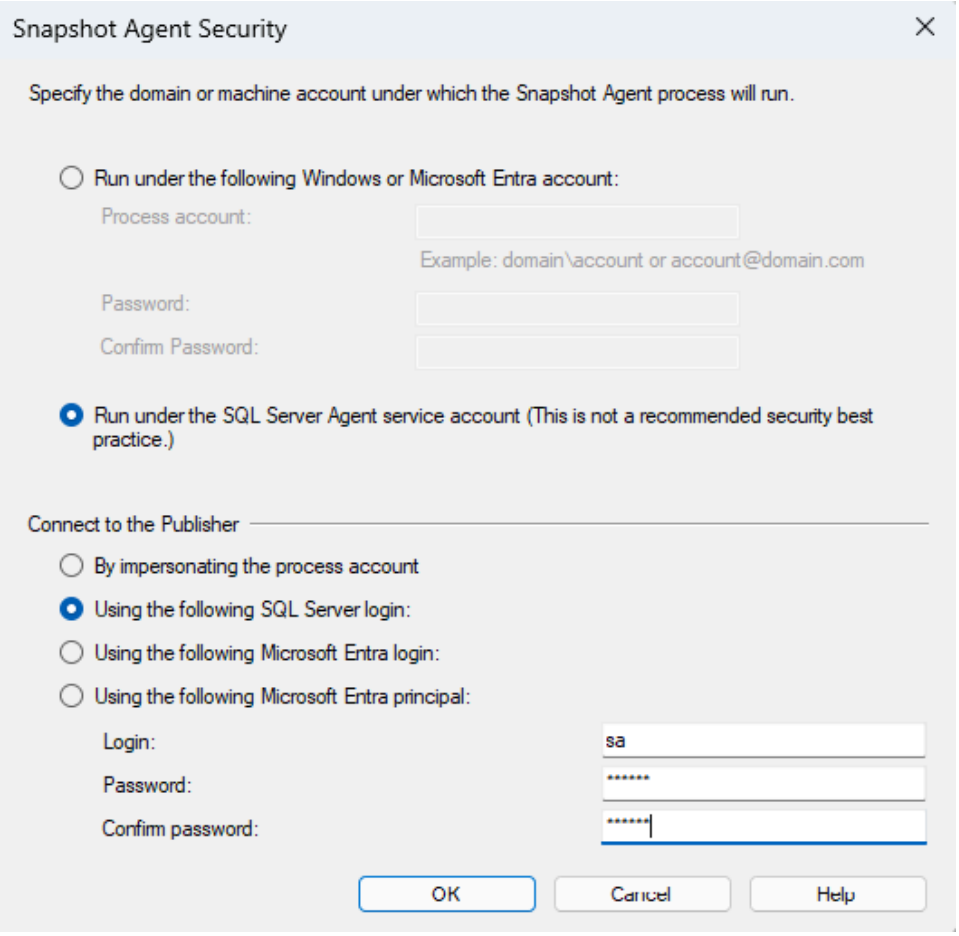
Bước 7: Chọn Security Settings..



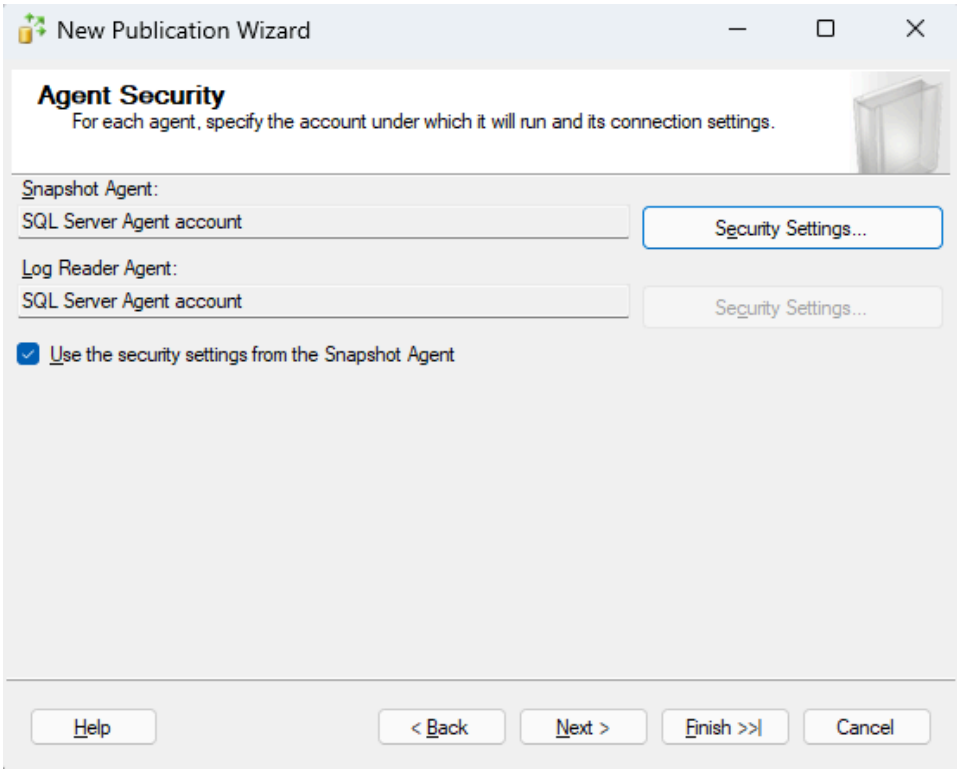
Bước 8: Tích chọn "Run under the SQL Server Agent service account" và "Using the following SQL Server login:". Sau đó nhập tài khoản mật khẩu sa (mà nhóm đã thống nhất từ trước):

Login: sa

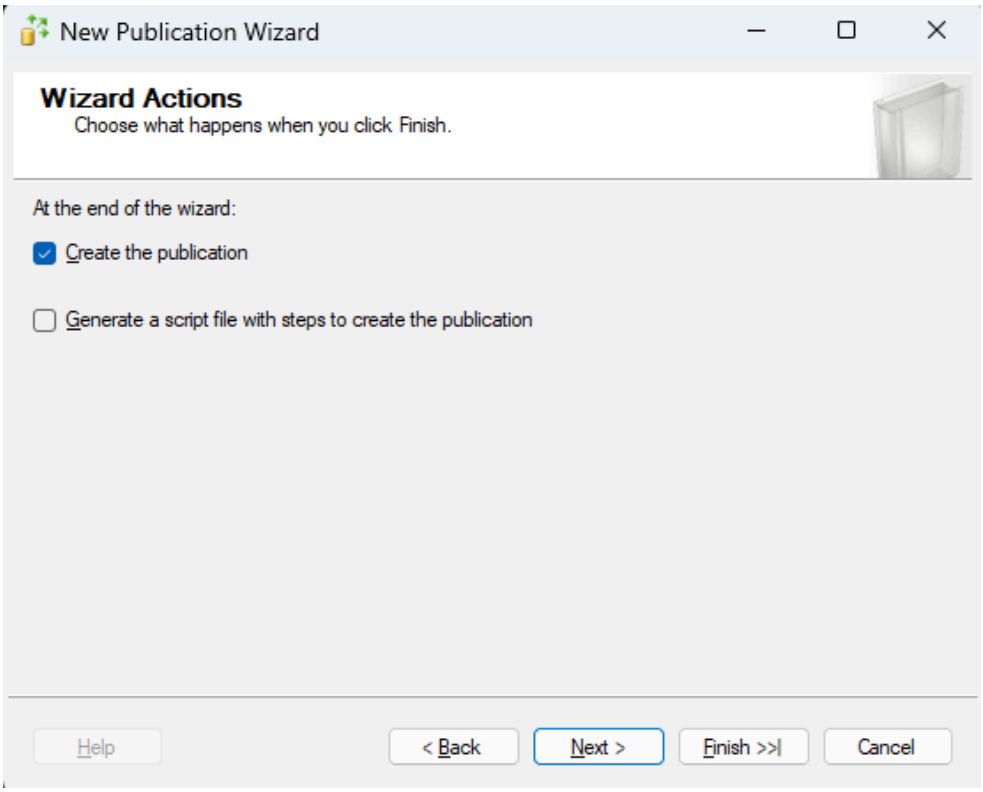
Password: 123456



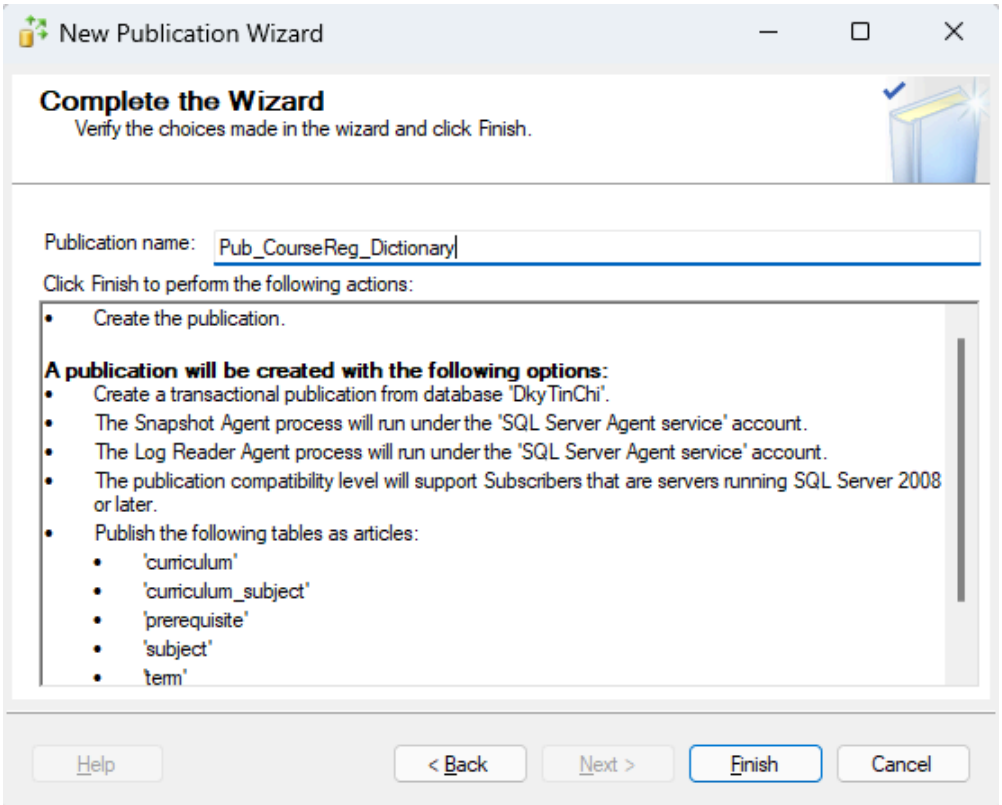
Bước 9: Chọn Next



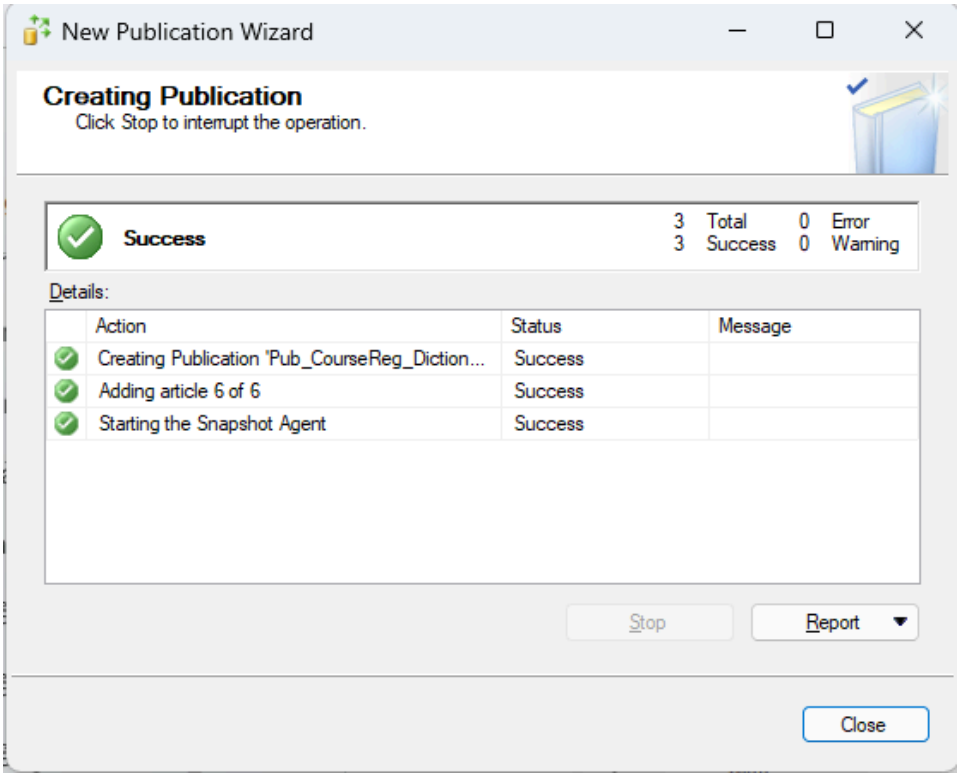
Bước 10: Tích chọn “Create the publication”. Click Next



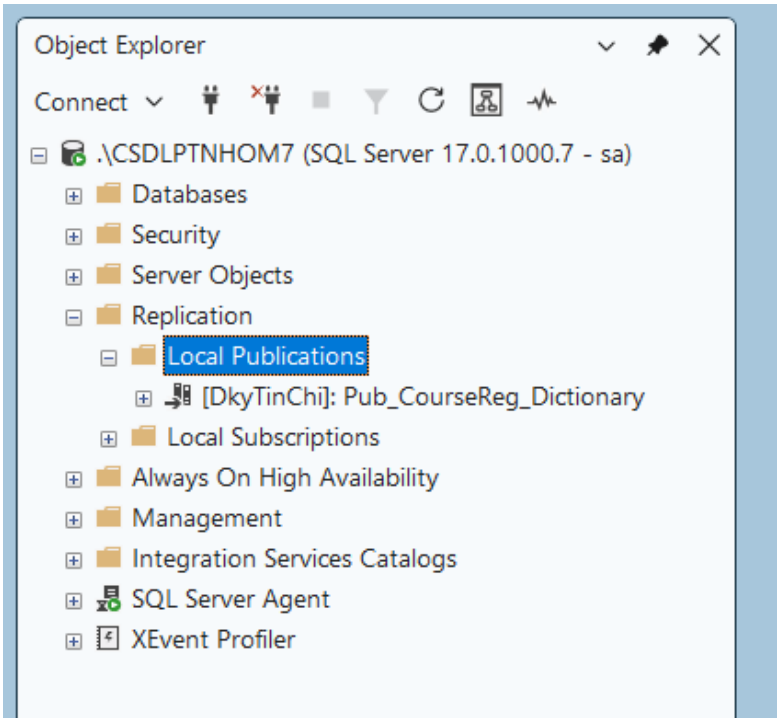
Bước 11: Đặt tên cho nhân bản muốn tạo ra trong ô Publication Name, ở đây là "Pub_CourseReg_Dictionary". Click "Finish"



Bước 12: Thông báo thành công -> ấn close



Bước 13: Refresh lại Local Publication để check lại xem đã có Pub_CourseReg_Dictionary chưa. Ở đây chúng ta đã tạo nhân bản Pub_CourseReg_Dictionary thành công



2.Tạo Merge replication

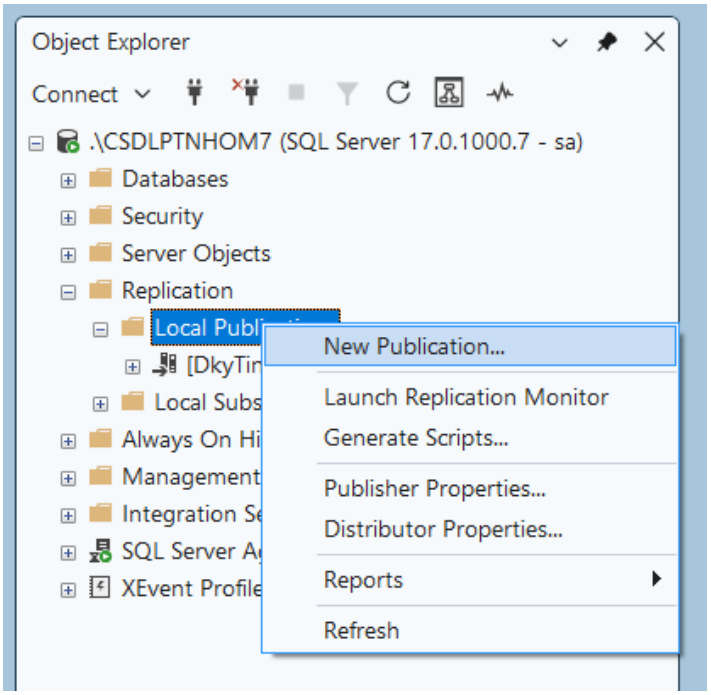
Merge replication cho phép nhiều server làm việc độc lập (online hay ofline) sau đó hợp nhất dữ liệu đã thay đổi lại dựa vào độ ưu tiên, thời điểm chỉnh sửa hoặc do người dùng tự quy định. Subscriber sẽ đồng bộ với publisher khi được kết nối vào

mạng và sẽ chuyển giao tất cả các row đã thực hiện thay đổi giữa publisher và subscriber kể từ lần đồng bộ cuối cùng.

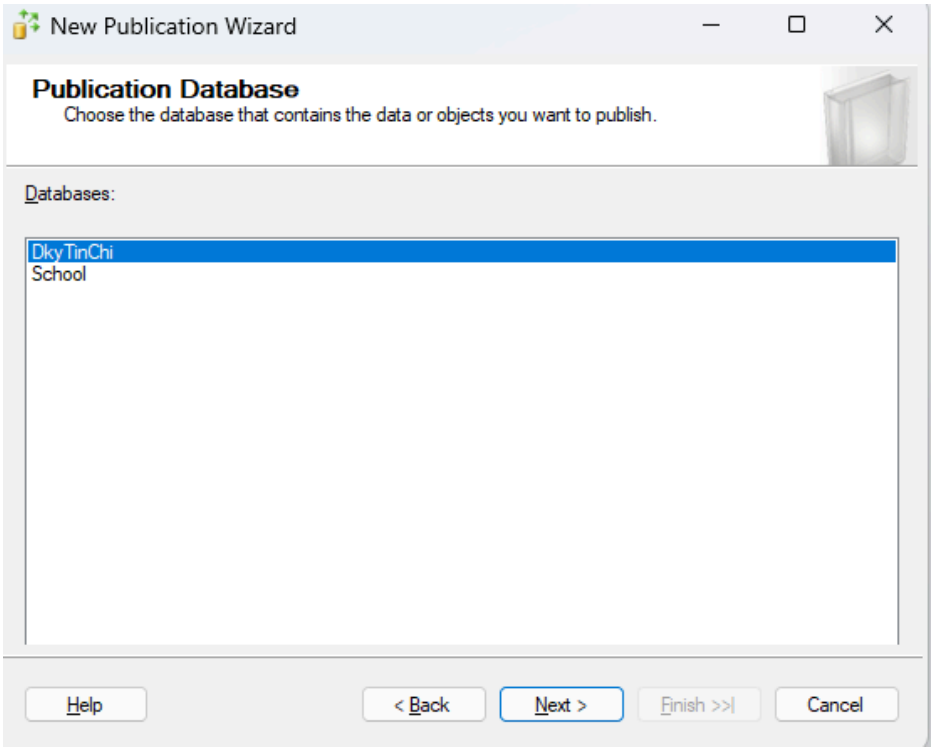
Merge replication thường được dùng trong các trường hợp sau:

- Nhiều subscriber cập nhật cùng một dữ liệu nhiều lần và muốn phân phối những thay đổi đó tới publisher và các subscriber khác
- Subscriber có nhu cầu nhận dữ liệu, thay đổi dữ liệu offline, sau đó đồng bộ hóa những thay đổi tới publisher và các subscriber khác
- Mỗi subscriber yêu cầu một phân vùng dữ liệu khác nhau

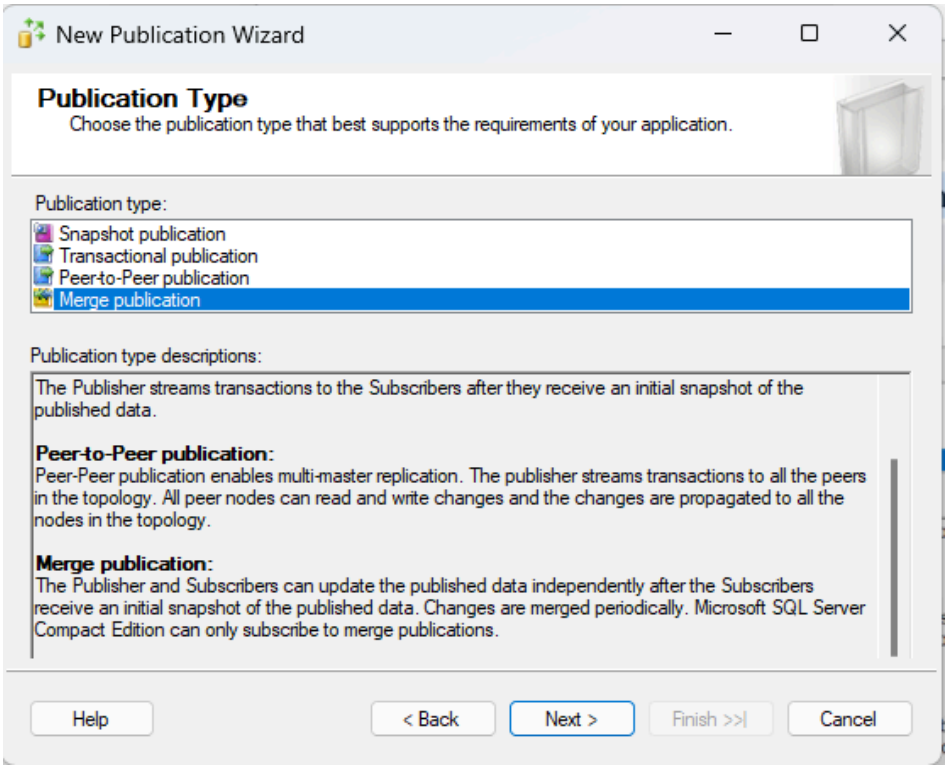
Bước 1: Click chuột phải vào Local Publication. Chọn New Publication



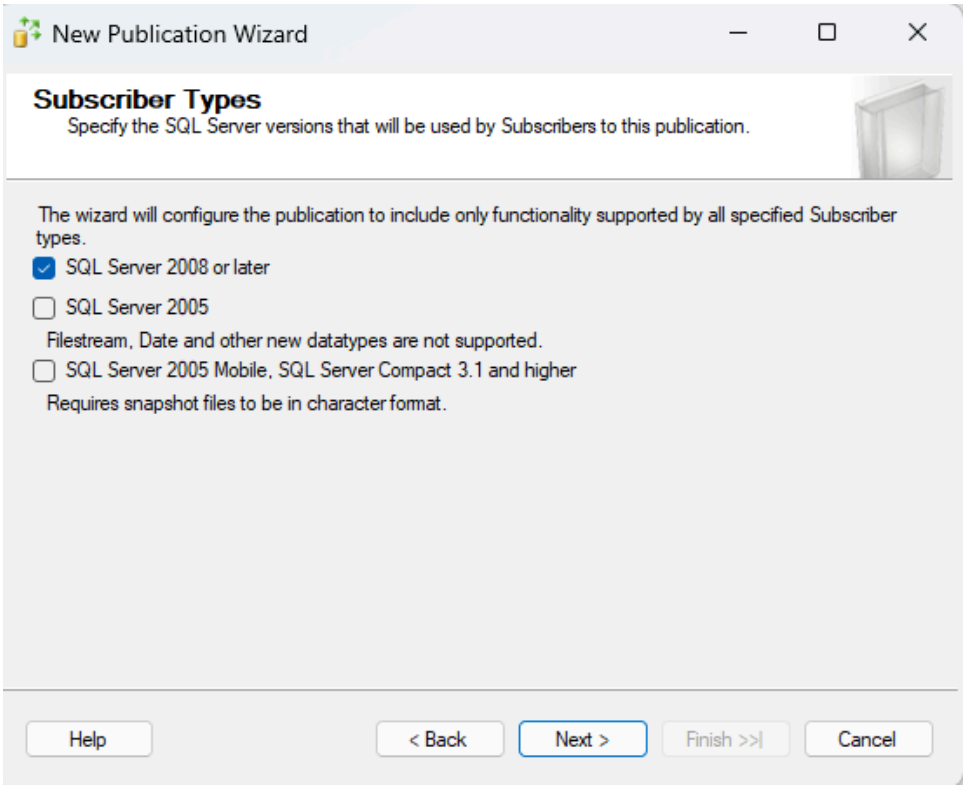
Bước 2: Chọn Database muốn phân mảnh => Chọn Next



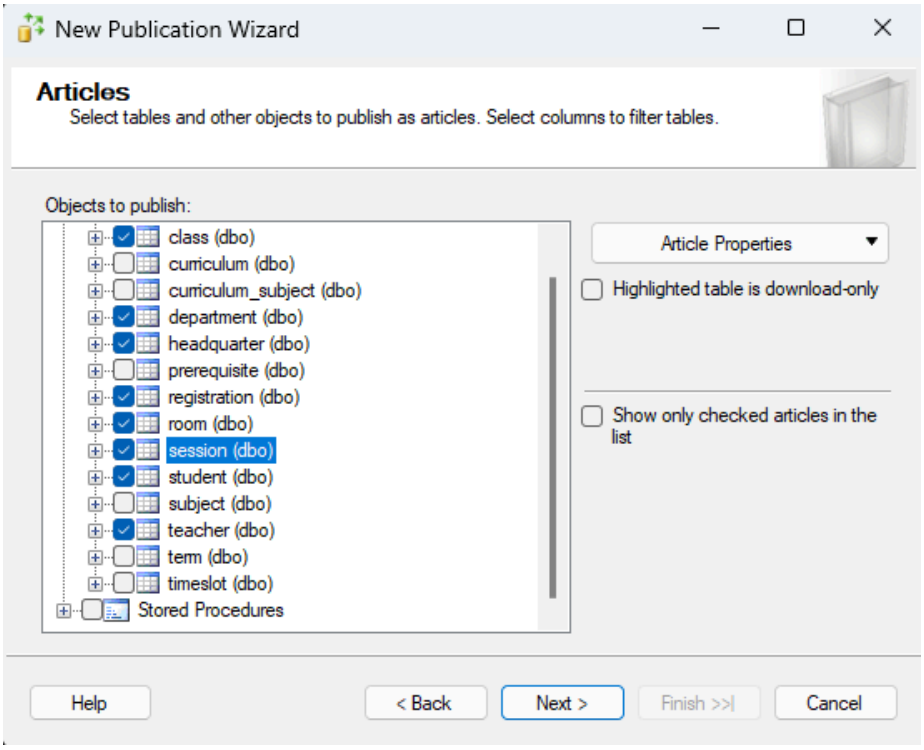
Bước 3: Chọn Merge publication để thực hiện phân mảnh => Click Next



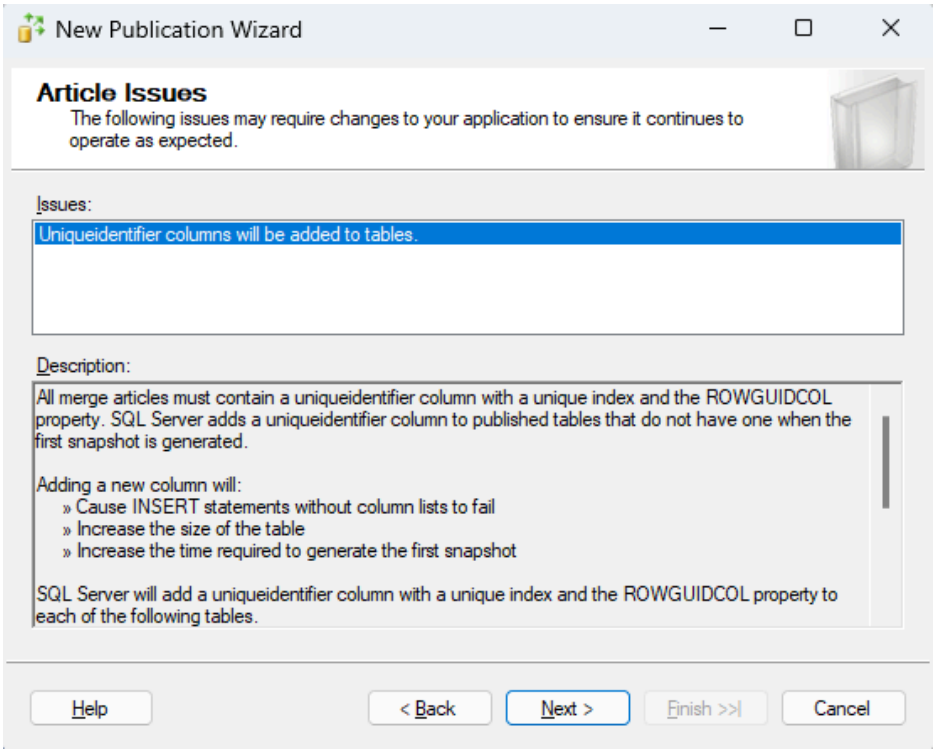
Bước 4: Tích “SQL Server 2008 or later”. Click Next



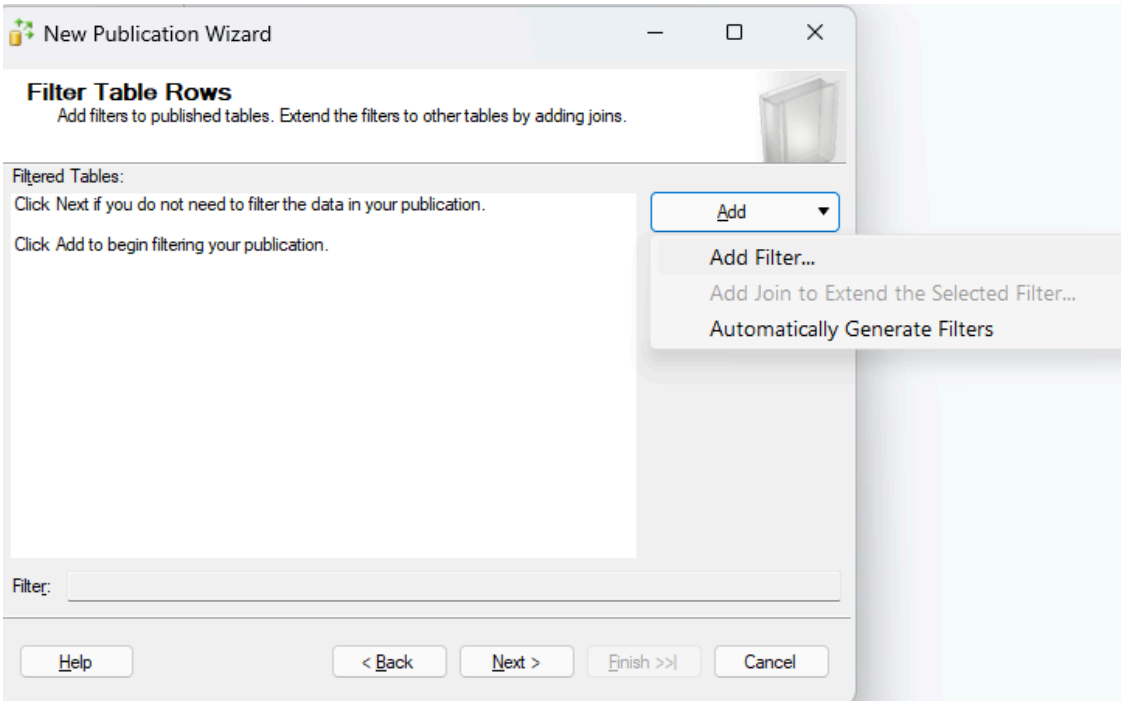
Bước 5: Dựa theo cấu trúc quan hệ, ta lựa chọn các bảng cần phân mảnh ngang bao gồm: headquarter, department, room, student, teacher, class, registration và session. Riêng bảng subject cùng với term, prerequisite, timeslot là các bảng danh mục dùng chung cho tất cả các cơ sở và đã được thiết lập nhân bản toàn phần (Transactional Replication), nên ta sẽ không chọn để phân mảnh



Bước 6: Chọn Next

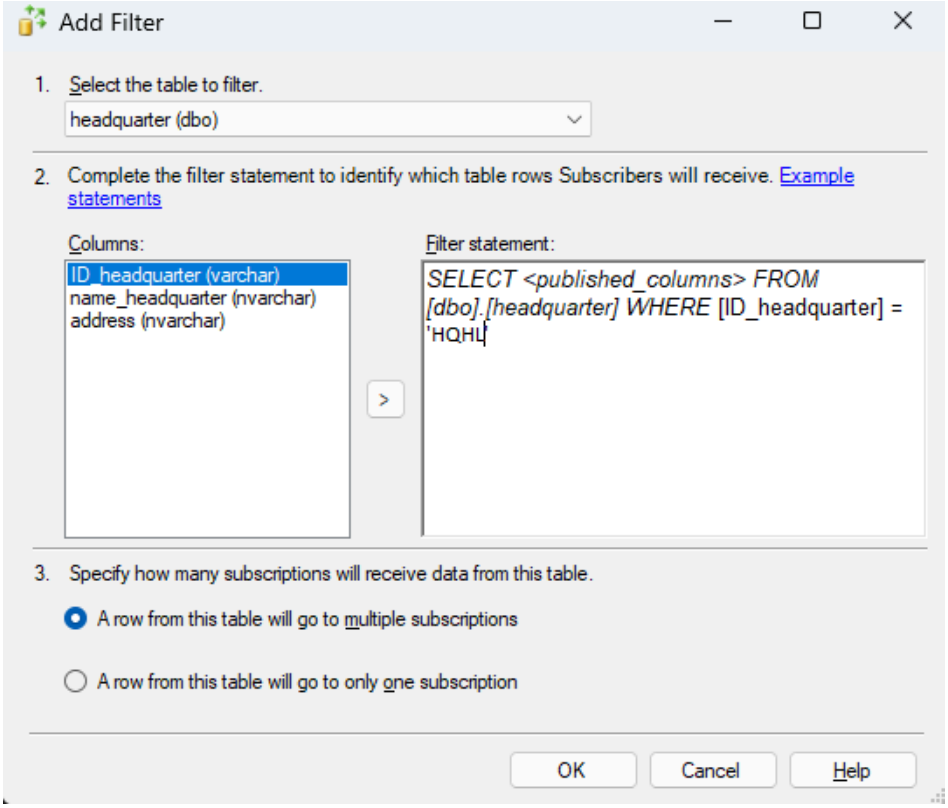


Bước 7: Click Add. Chọn Add Filter

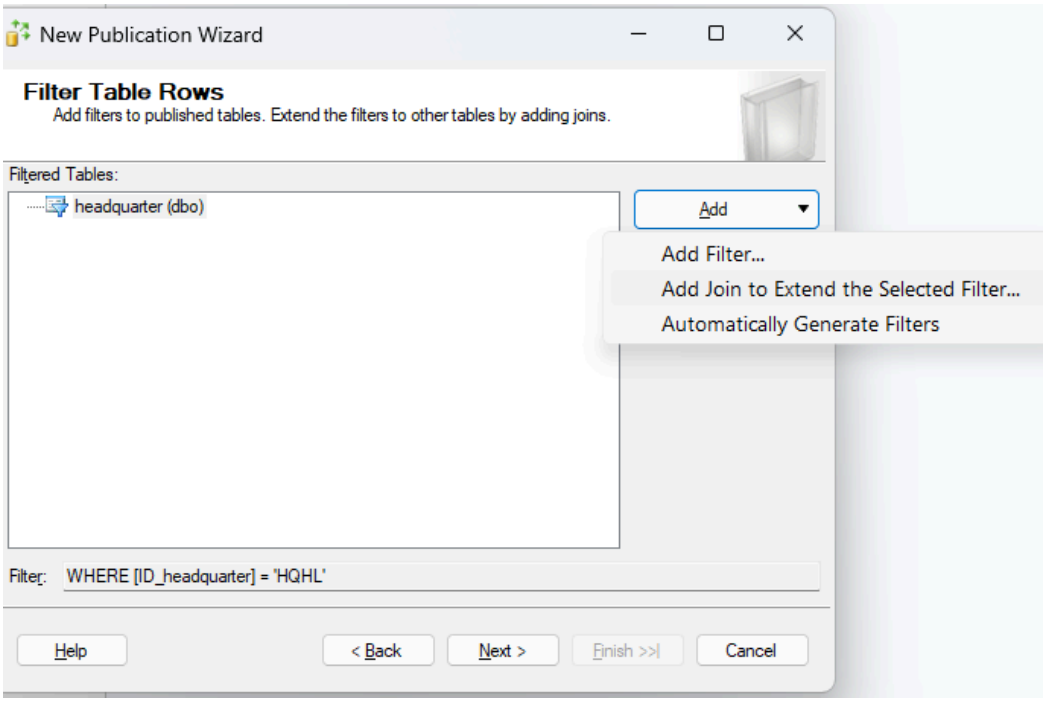


Bước 8: Tạo phân mảnh ngang nguyên thủy

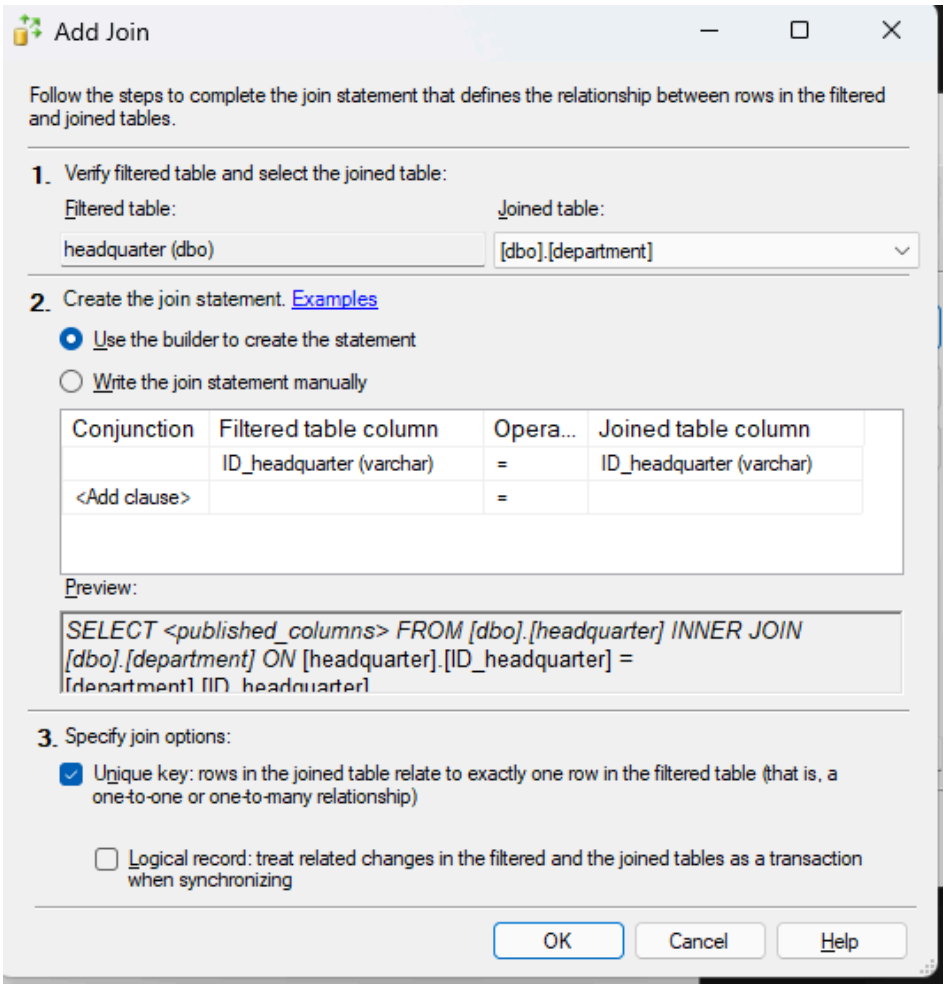
Ta sẽ phân mảnh từ bảng headquarter đổ xuống. Ở Select the table to filter lựa chọn headquarter => Nháy đúp vào ID_headquarter => từ “WHERE [ID_headquarter] “ ta nhập thêm “=’HQHL’” => Click OK



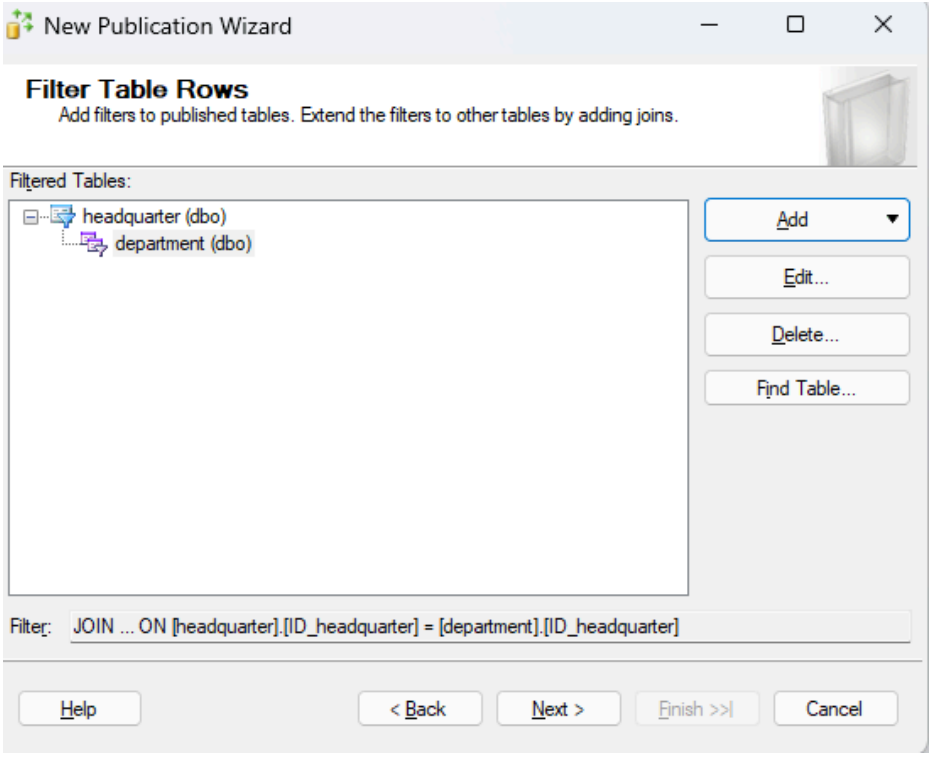
Bước 9: Sau headquarter ta sẽ phân mảnh các bảng liên quan. Chọn "Add join to Extend..."



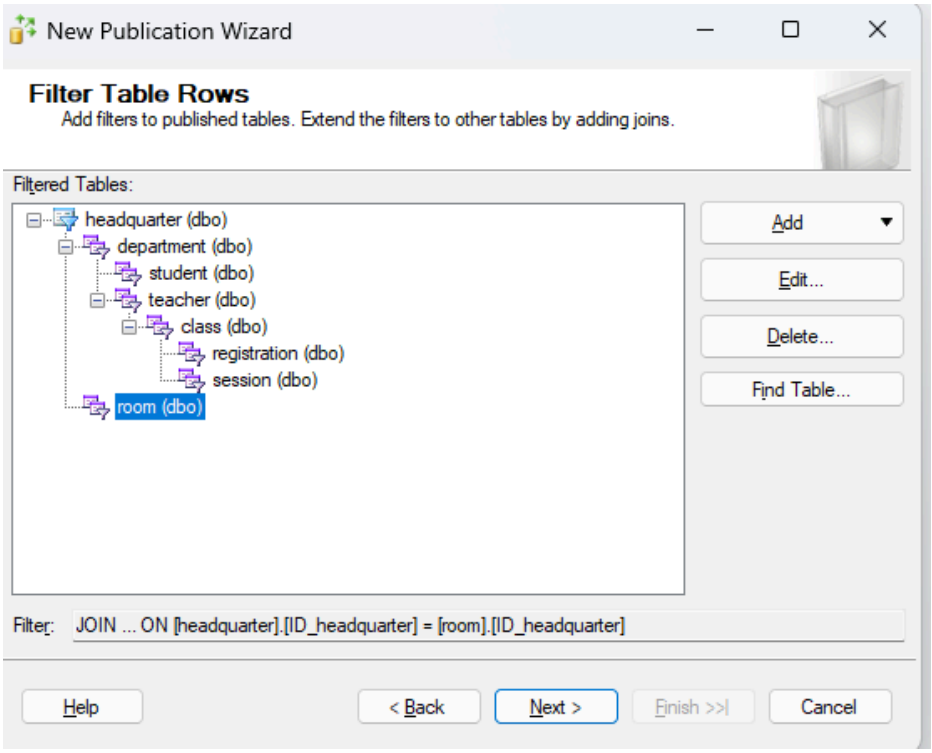
Bước 10: Ở Joined table ta sẽ chọn department. Click OK



Bước 11: department đã được thêm vào

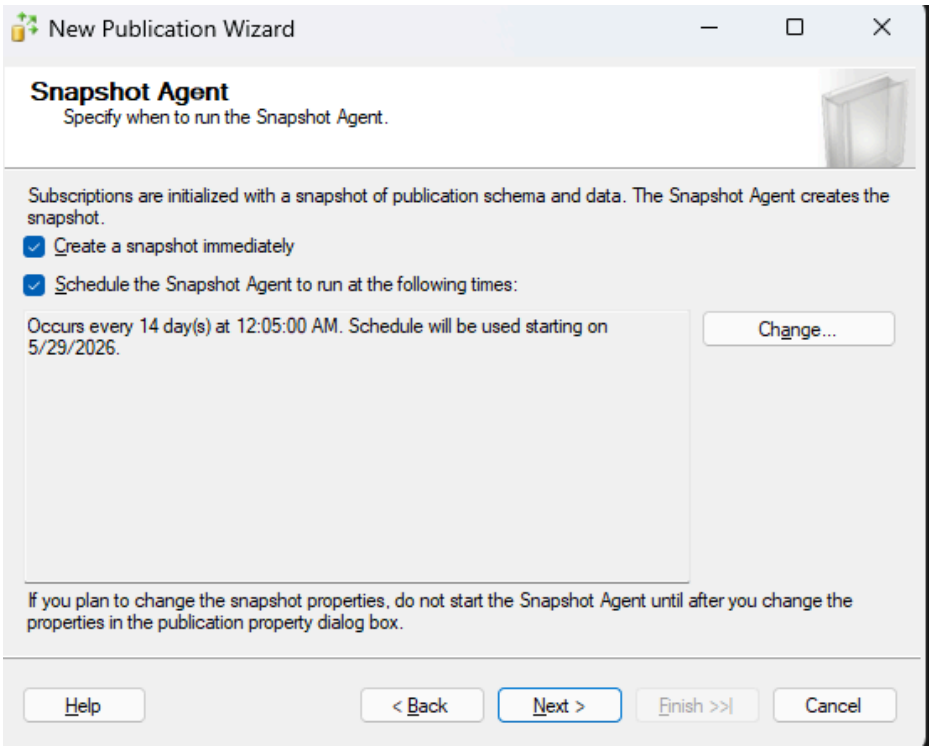


Bước 12: Làm tương tự với các bảng khác ta được kết quả dưới đây

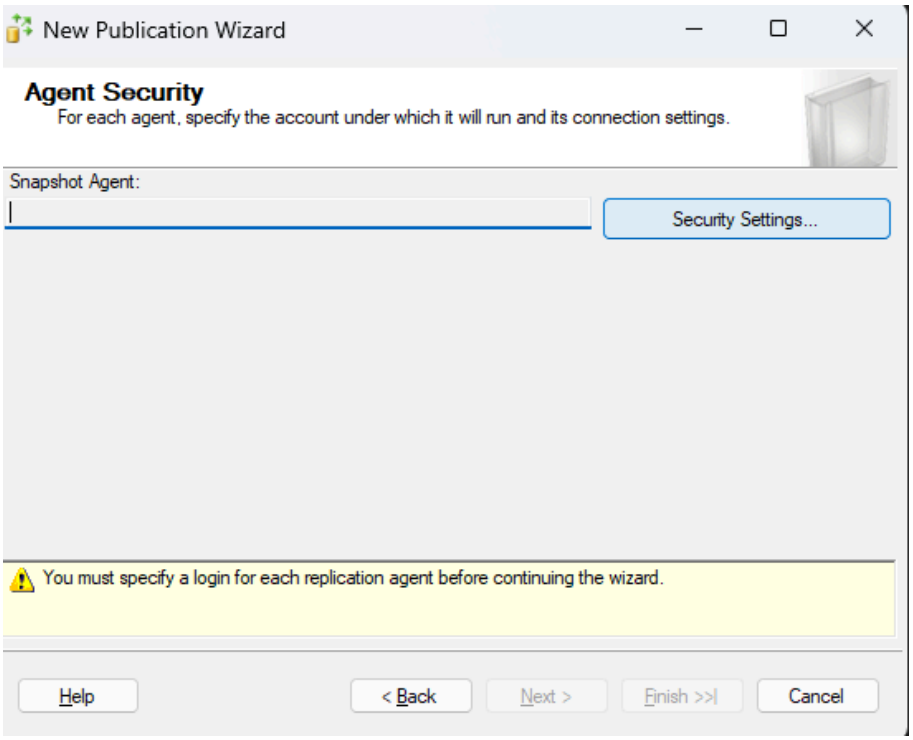


Bước 13: Chọn Next

Bước 14: Tích “Create a snapshot immediately” và “Schedule the Snapshot...”(không bắt buộc) => Click Next

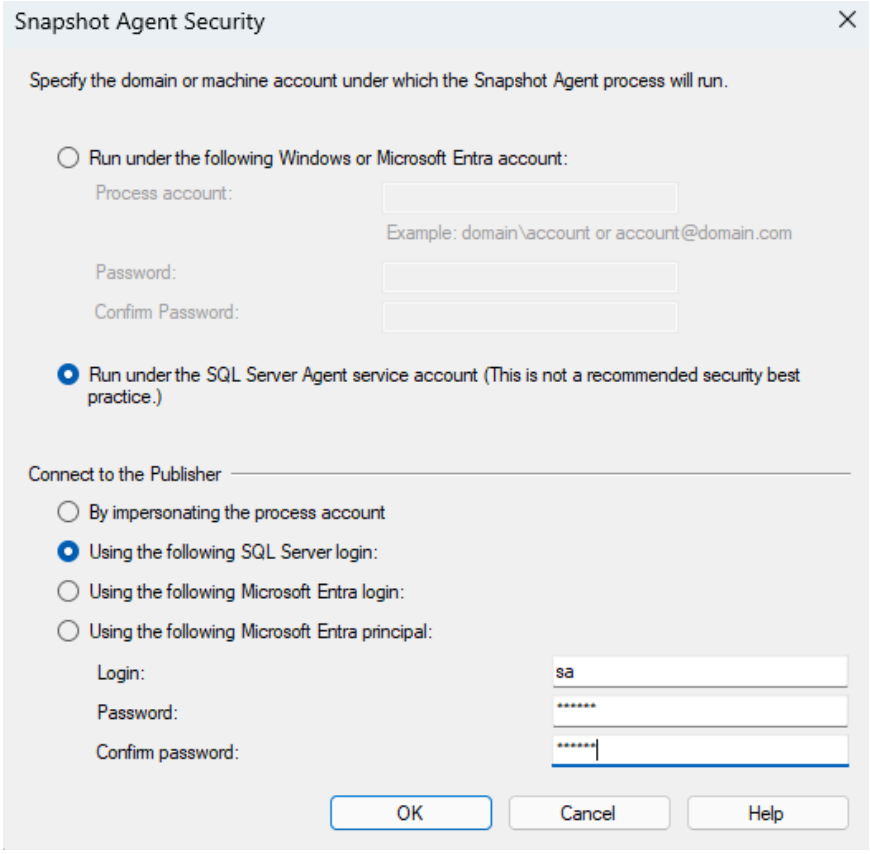


Bước 15: Chọn Security Settings....

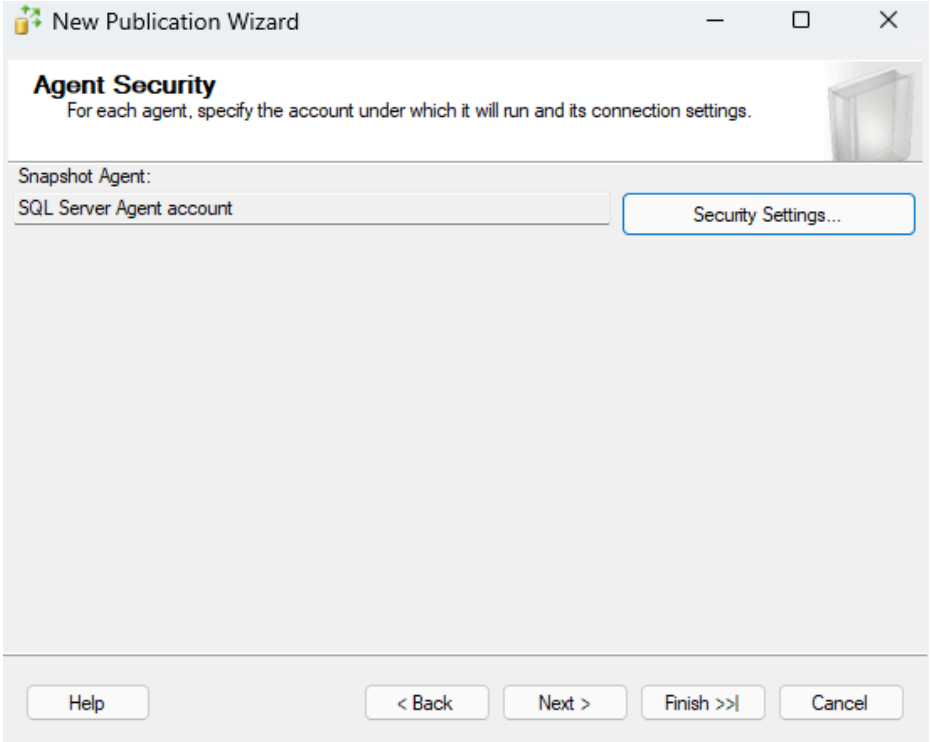


Bước 16: Tích chọn "Run under the SQL Server Agent service account" và "Using the following SQL Server login:". Sau đó nhập tài khoản mật khẩu sa (mà nhóm đã thống nhất từ trước):

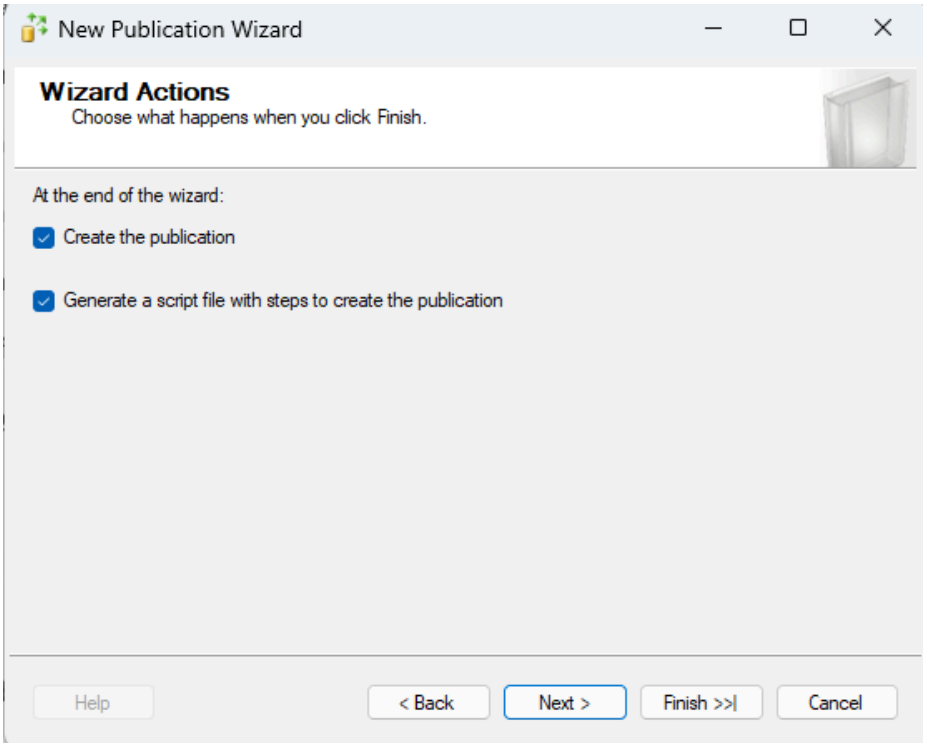
Login: sa - Password: 123456



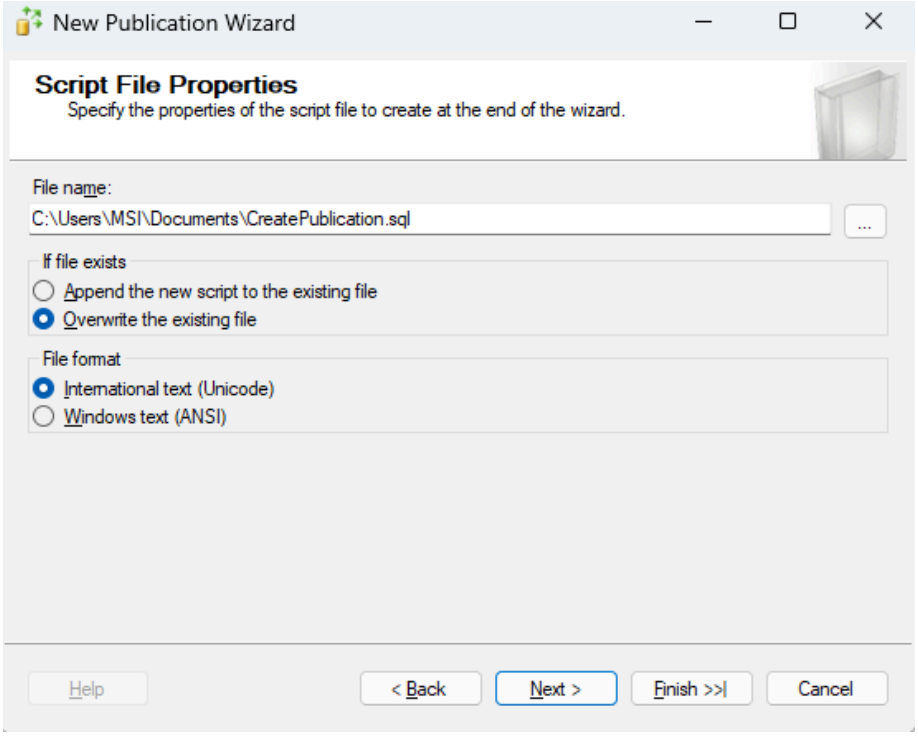
Bước 17: Chọn Next



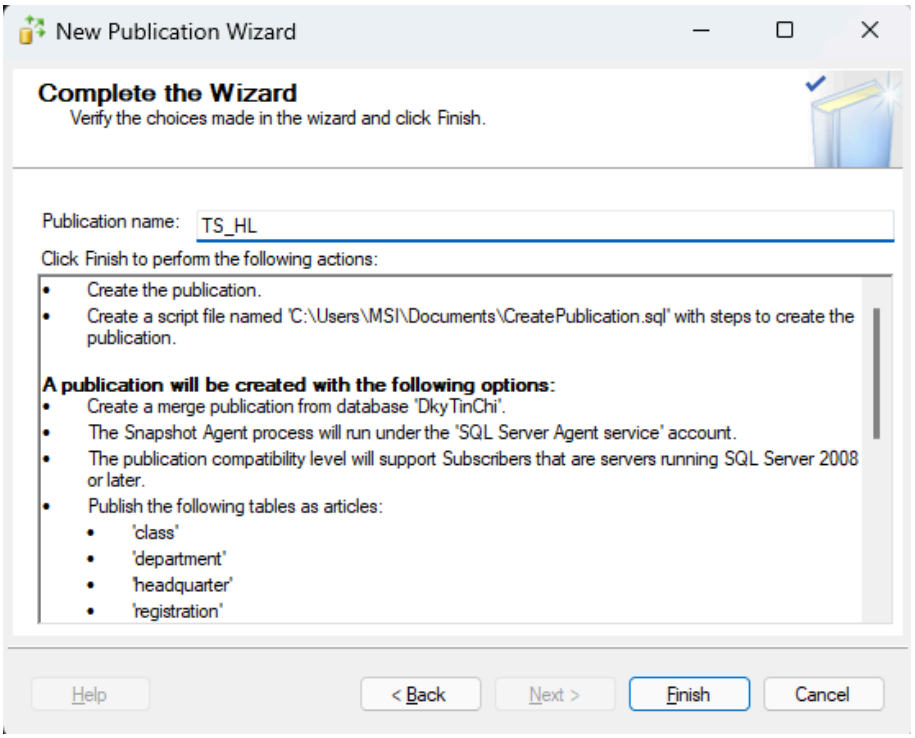
Bước 18: Tích chọn “Create the publication” và “Generate a script ...” => Click Next



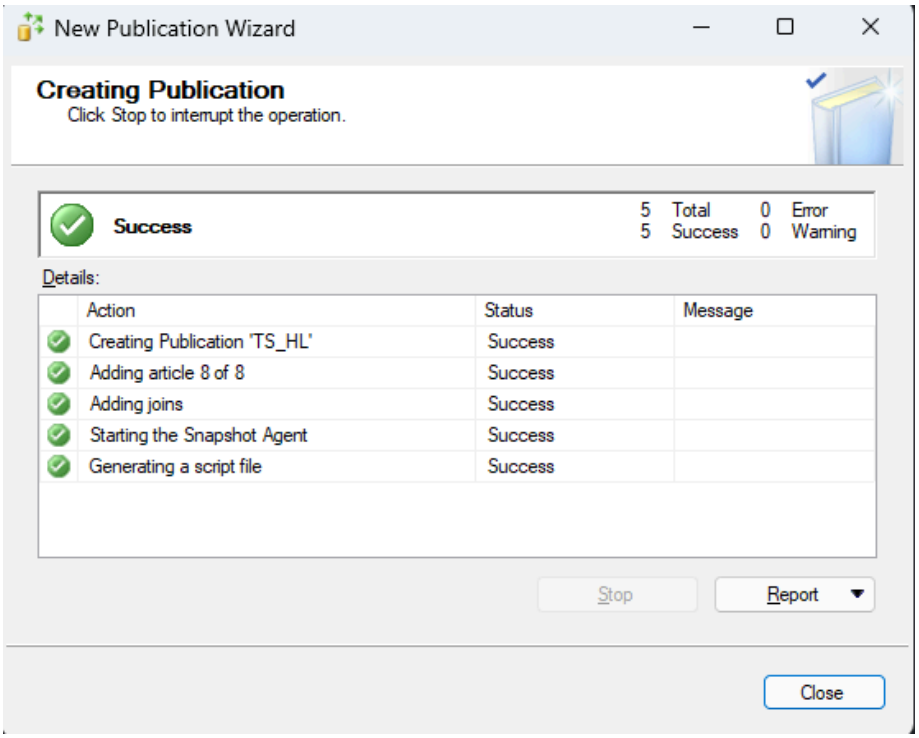
Bước 19: Tích chọn “Overwrite the existing file” và “International text” => Click Next



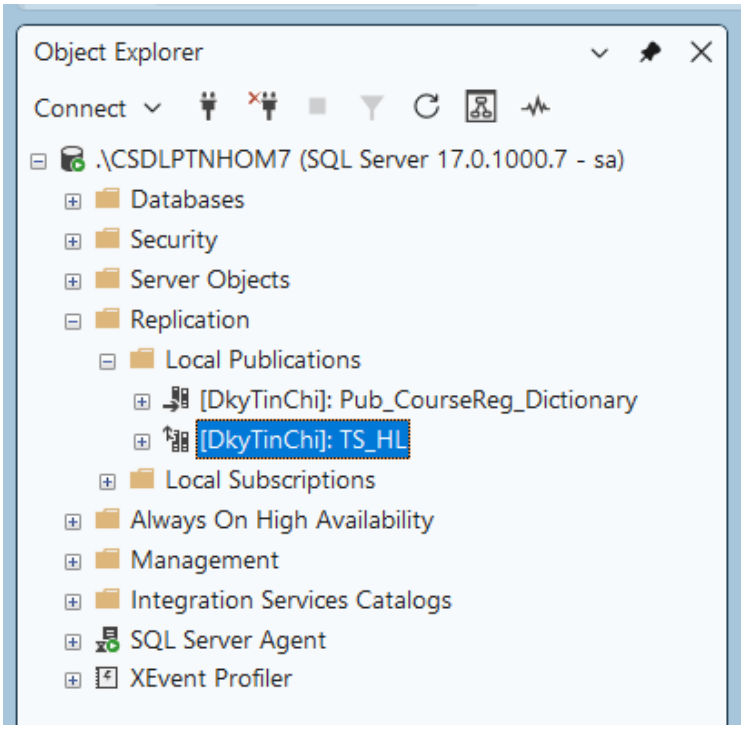
Bước 20: Đặt tên phân mảnh tại ô Publication name -> Click Finish



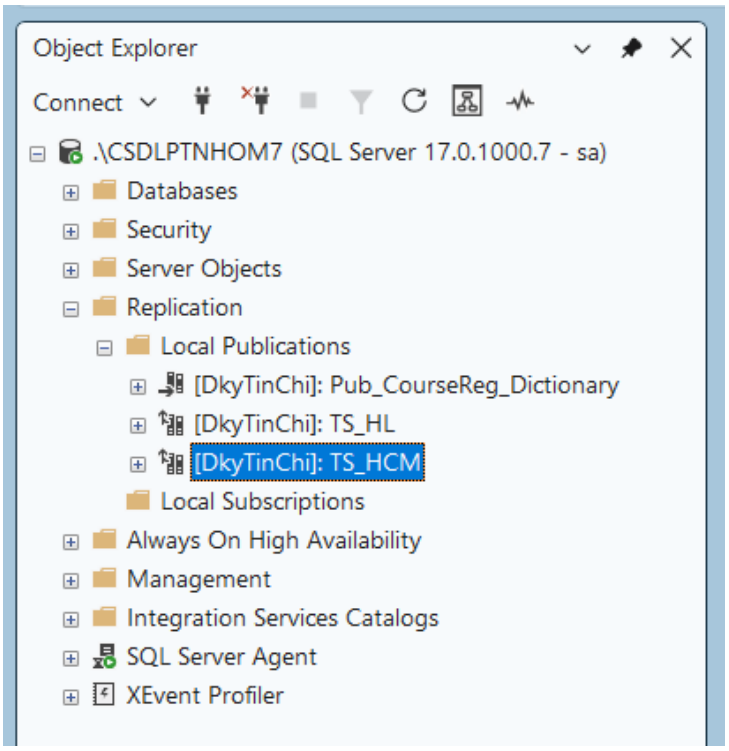
Bước 21: Thông báo thành công



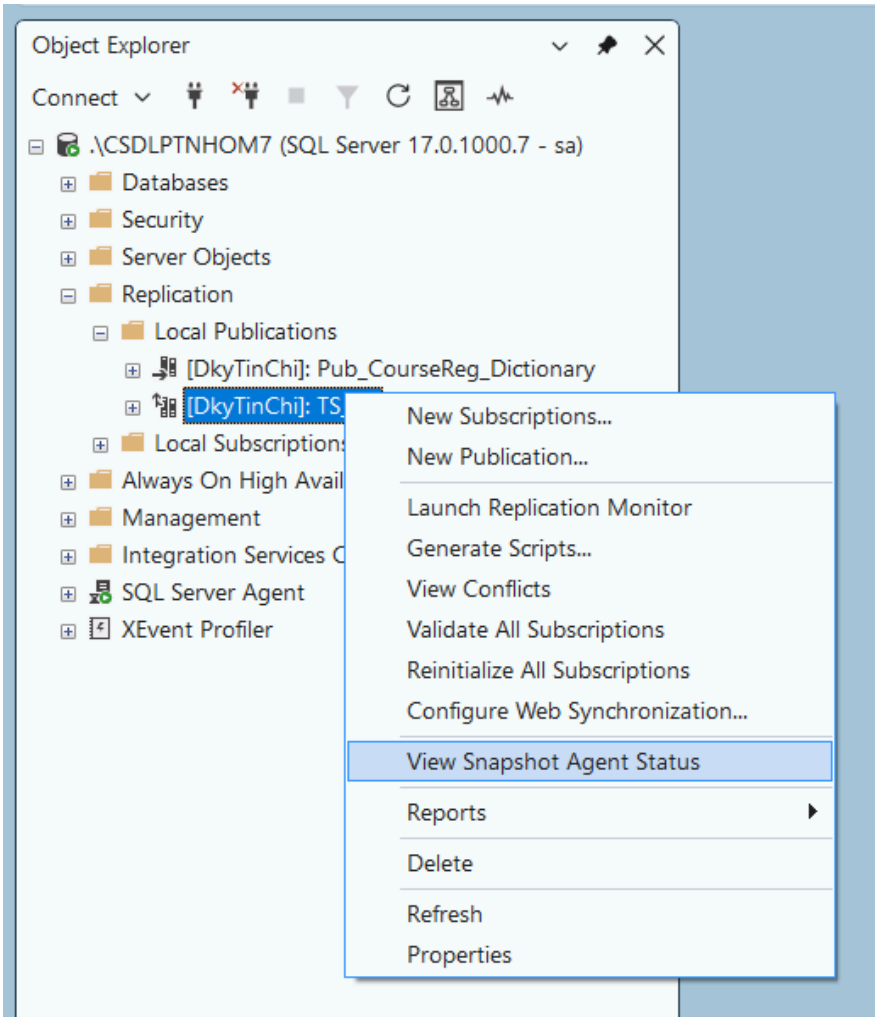
Bước 22: Refresh lại Local Publications để check xem đã có phân mảnh vừa tạo chưa.



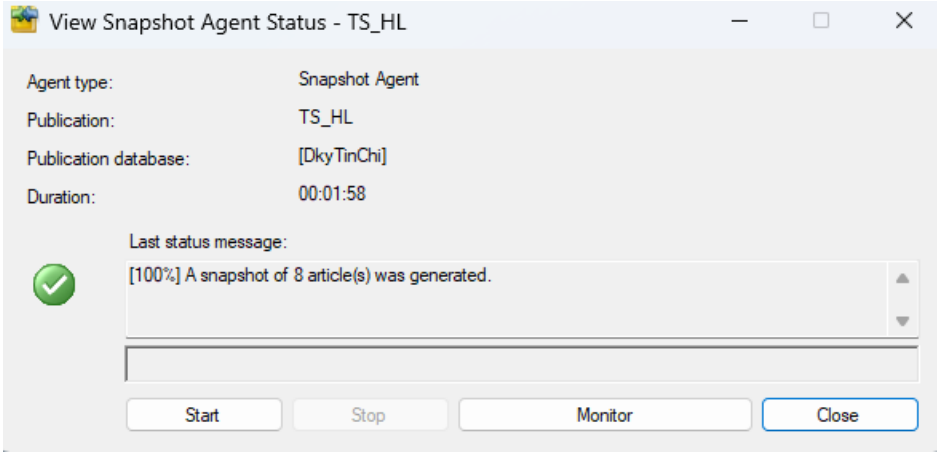
Làm tương tự với các trụ sở khác



Sau khi được thì chú ý phải view snapshot cho publication đó



Nếu như này thì sẽ thành công



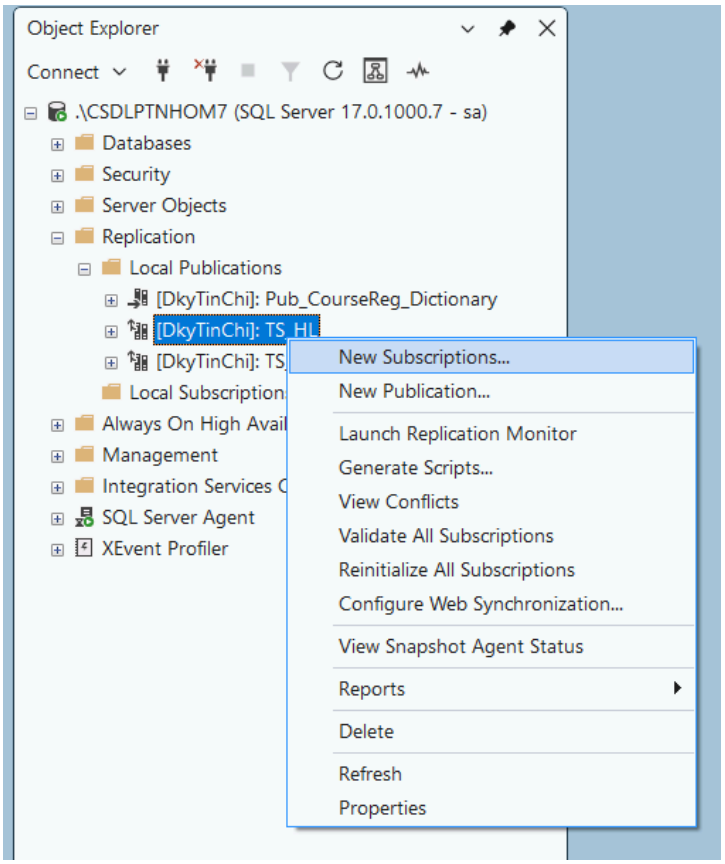
VIII. Tạo Subscriptions

1. Thông tin máy trạm

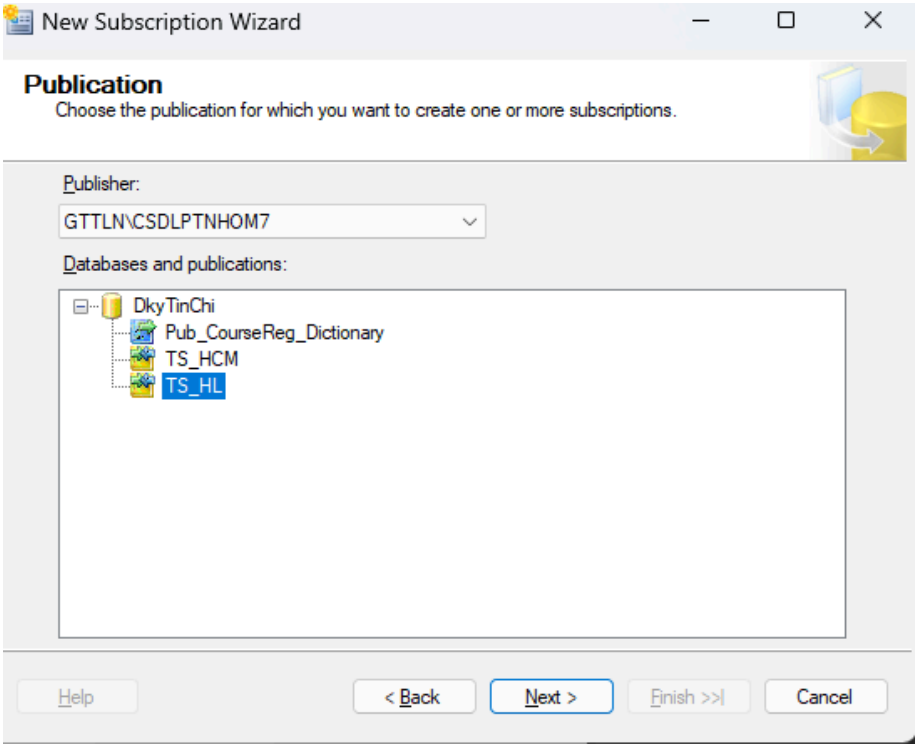
Tên	Tên Sever	Địa điểm
Tiên	GTTLN\CSDLPTNHOM7	TS_HL
Huy	ADMIN\CSDLPT_NHOM_7	TS_HCM

2. Tạo Subscriptions

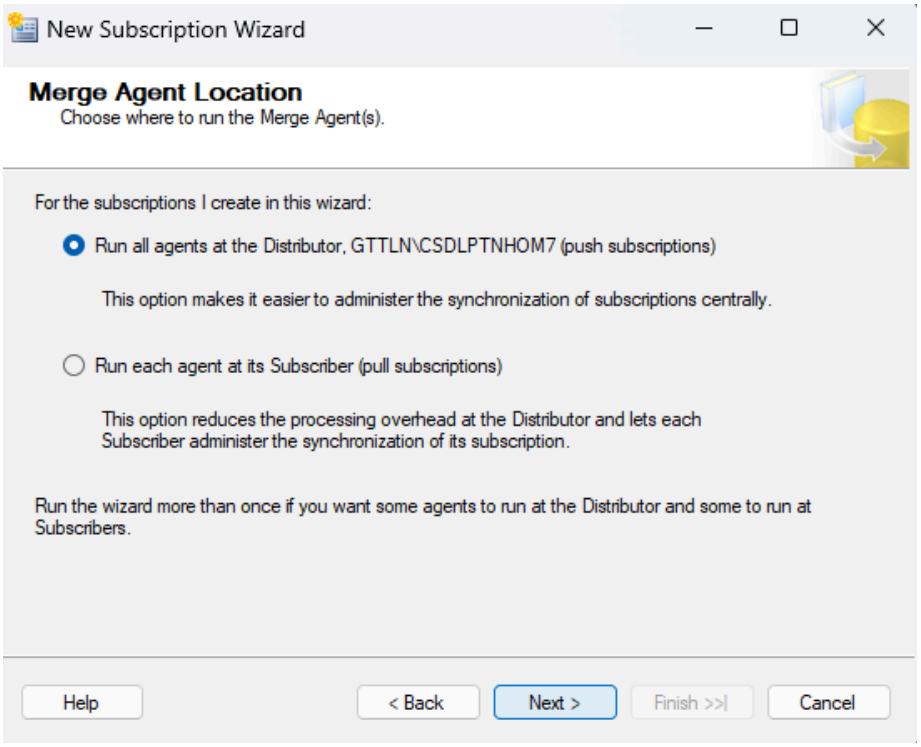
Bước 1: Chuột phải vào mảnh và chọn New Subscriptions -> bảng New Subscription Wizard hiện ra chọn next.



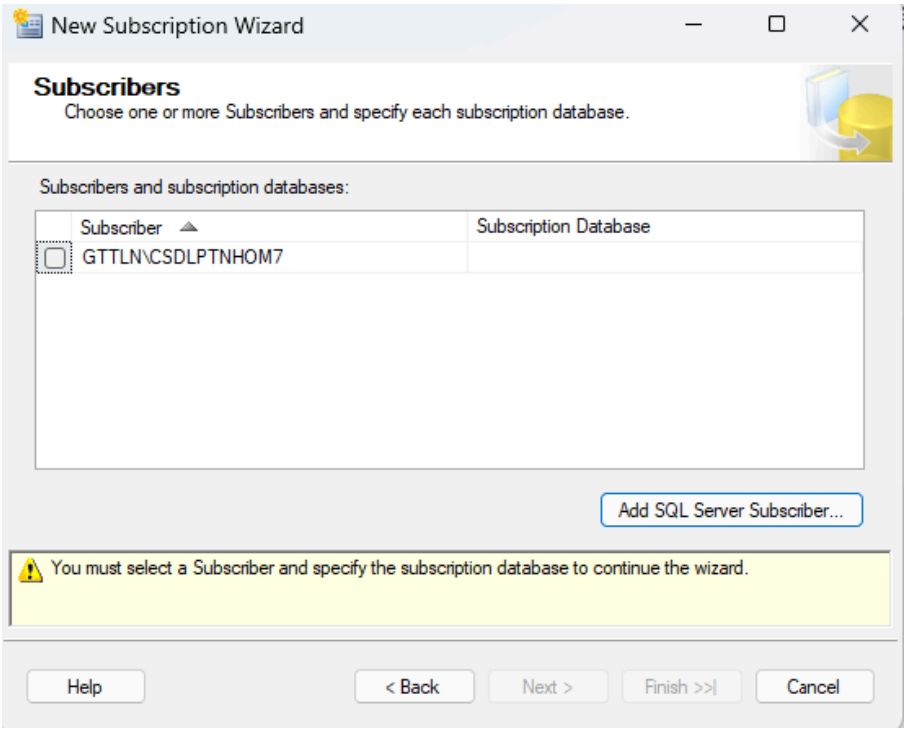
Bước 2: Chọn database và publication

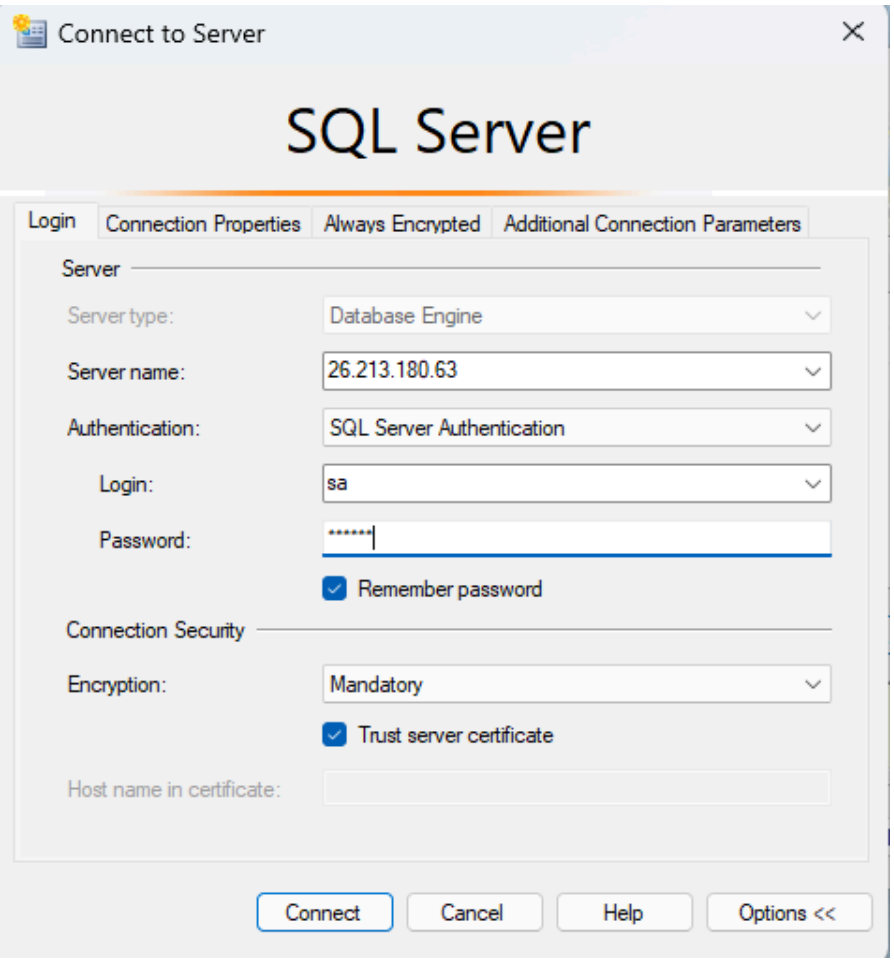


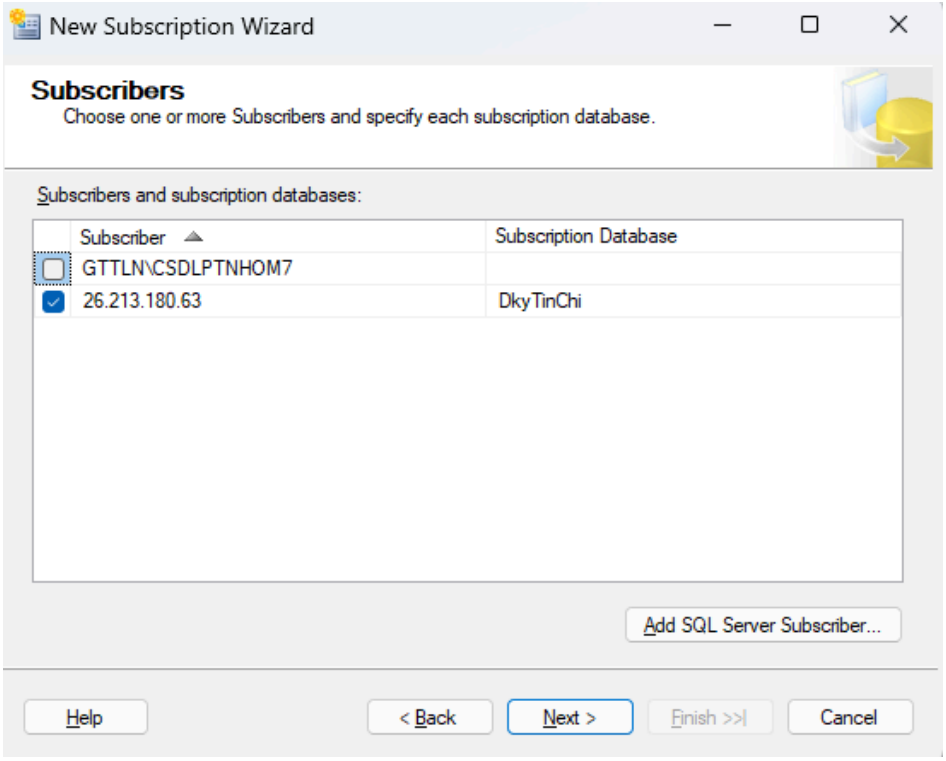
Bước 3: Tại bảng Merge Agent Location chọn “Run all agents the Distributor,...” và chọn Next



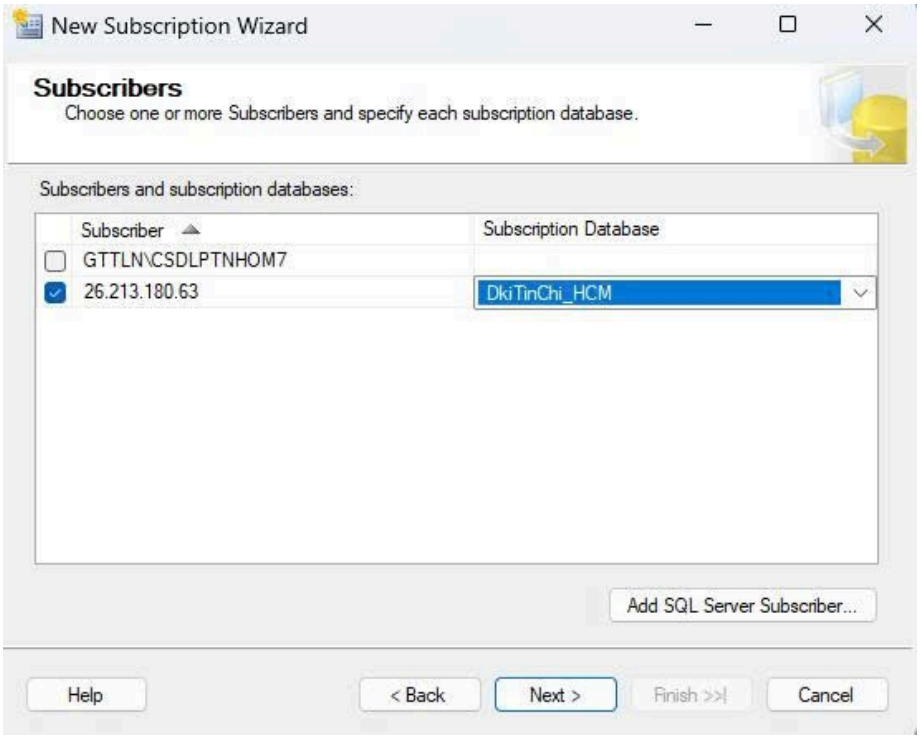
Bước 4: Bấm "Add SQL Server Subscriber...", và đăng nhập tài khoản để kết nối Server



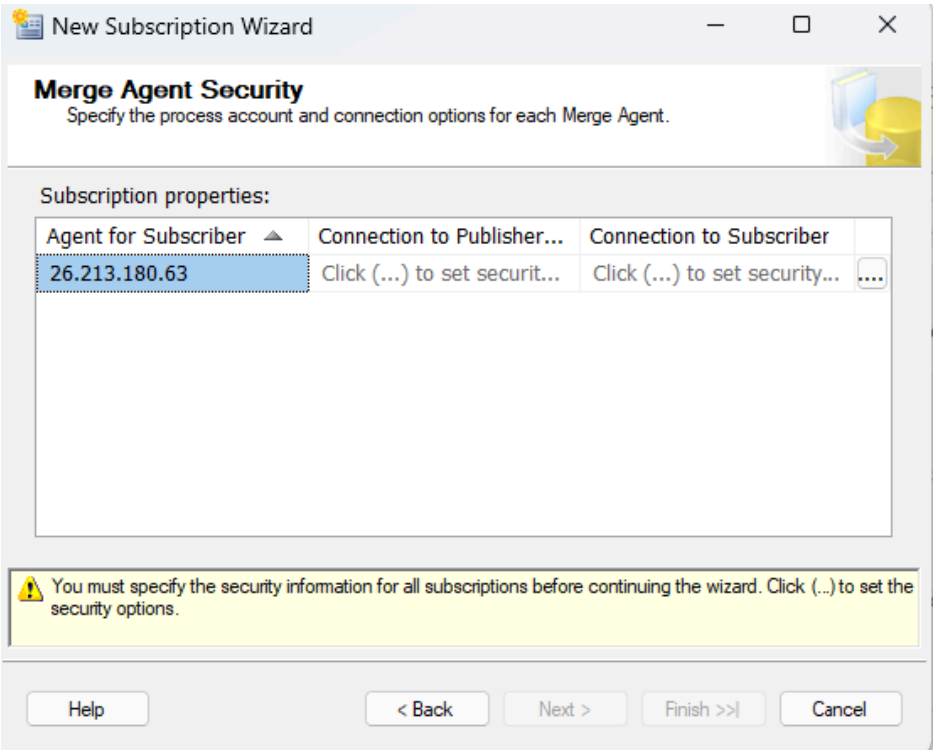


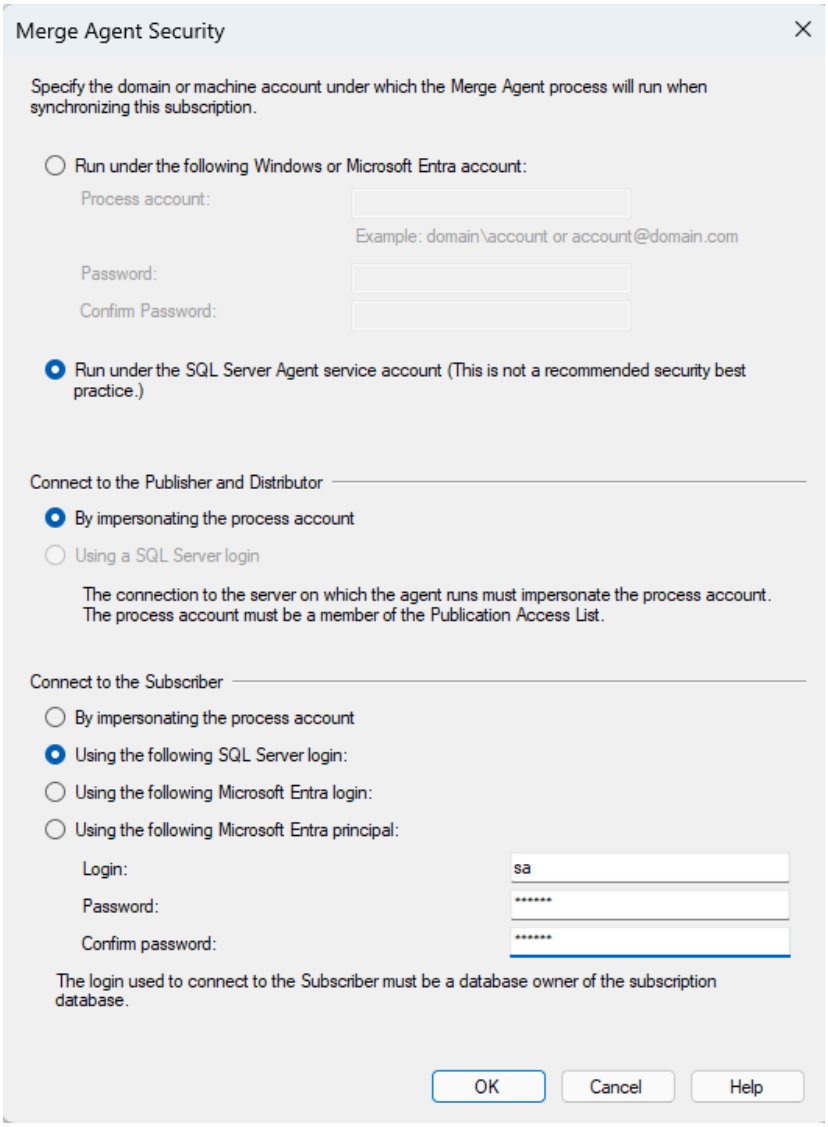


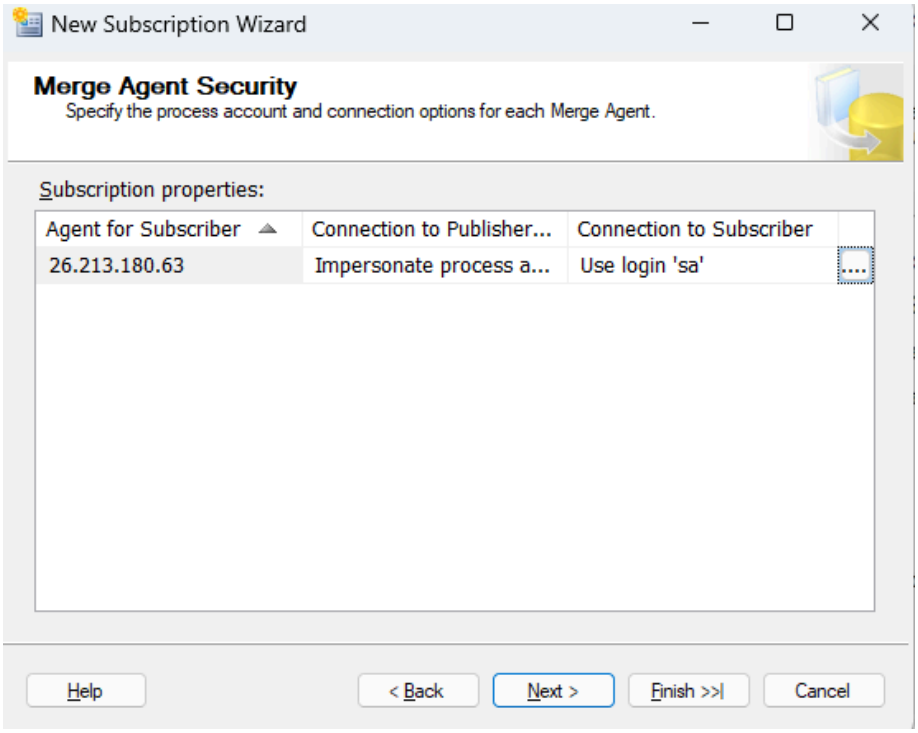
Bước 5: Tại subscribers cài đặt Subscription Database chọn “New database” tương ứng



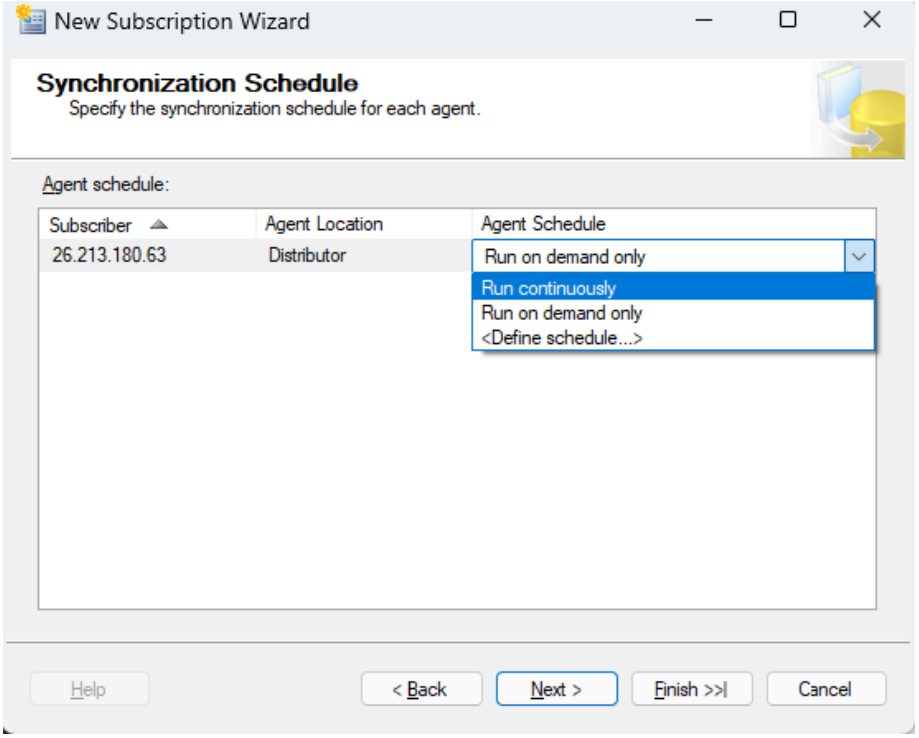
Bước 6: Trong Merge Agent Security tích chọn các nút như hình và đăng nhập tài khoản vào connect to the subscriber -> click "Ok"



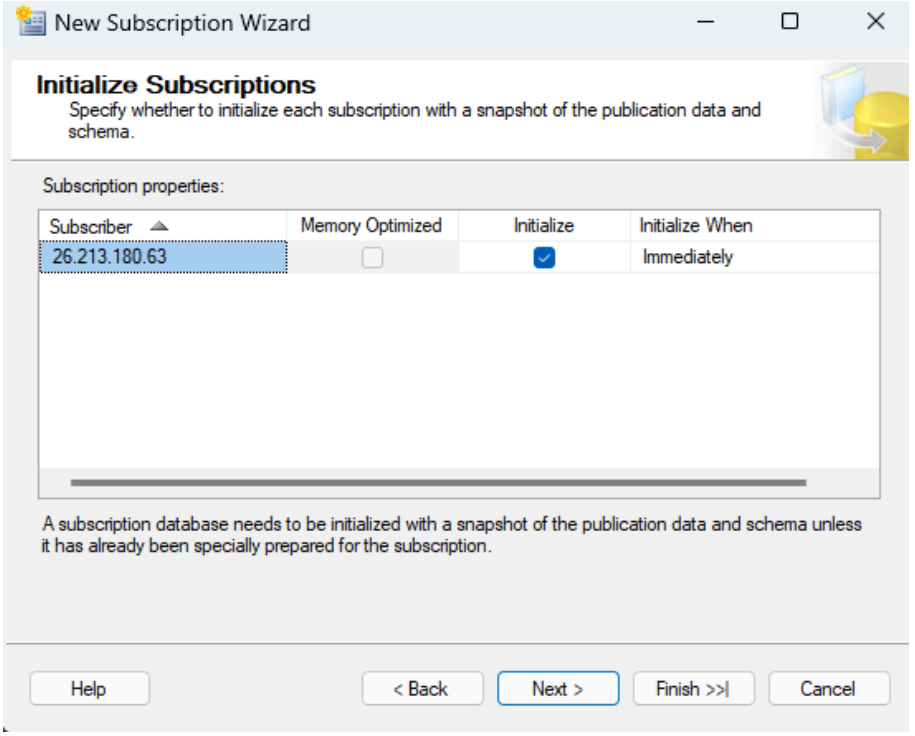

Merge Agent Security
Specify the domain or machine account under which the Merge Agent process will run when synchronizing this subscription.
Run under the following Windows or Microsoft Entra account:
Process account:
Example: domain\account or account@domain.com
Password:
Confirm Password:
Run under the SQL Server Agent service account (This is not a recommended security best practice.)
Connect to the Publisher and Distributor
By impersonating the process account
Using a SQL Server login
The connection to the server on which the agent runs must impersonate the process account. The process account must be a member of the Publication Access List.
Connect to the Subscriber
By impersonating the process account
Using the following SQL Server login:
Using the following Microsoft Entra login:
Using the following Microsoft Entra principal:
Login:
sa
Password:
Confirm password:
The login used to connect to the Subscriber must be a database owner of the subscription database.
OK
Cancel
Help


New Subscription Wizard
Merge Agent Security
Specify the process account and connection options for each Merge Agent.
Subscription properties:
Agent for Subscriber
Connection to Publisher...
Connection to Subscriber
26.213.180.63
Impersonate process a...
Use login 'sa'
Help
< Back
Next >
Finish >>|
Cancel

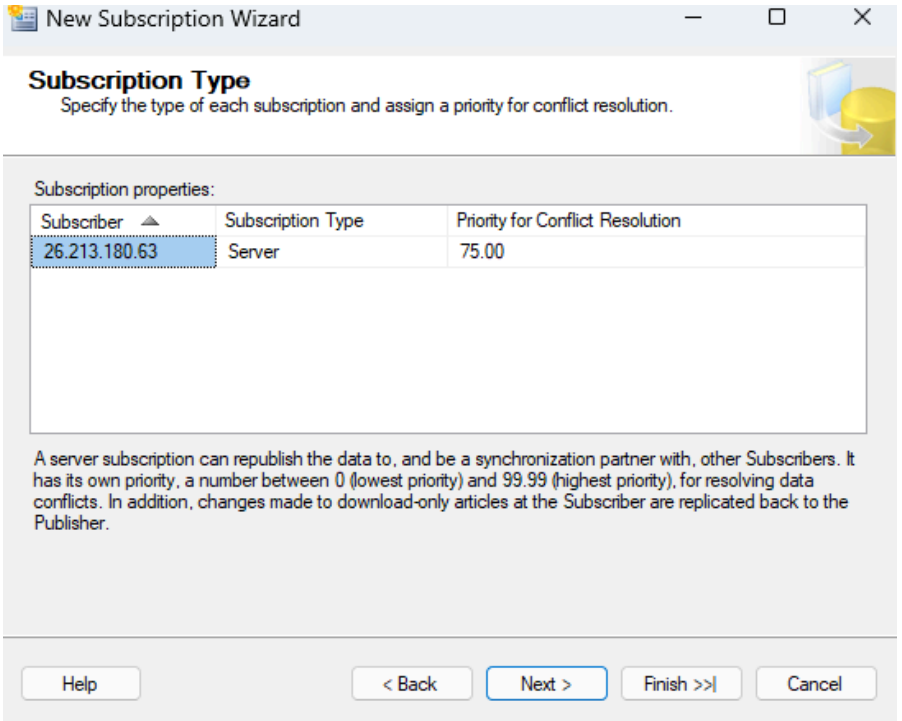
Bước 7: “Next” -> set “Agent Schedule” là Run continuously



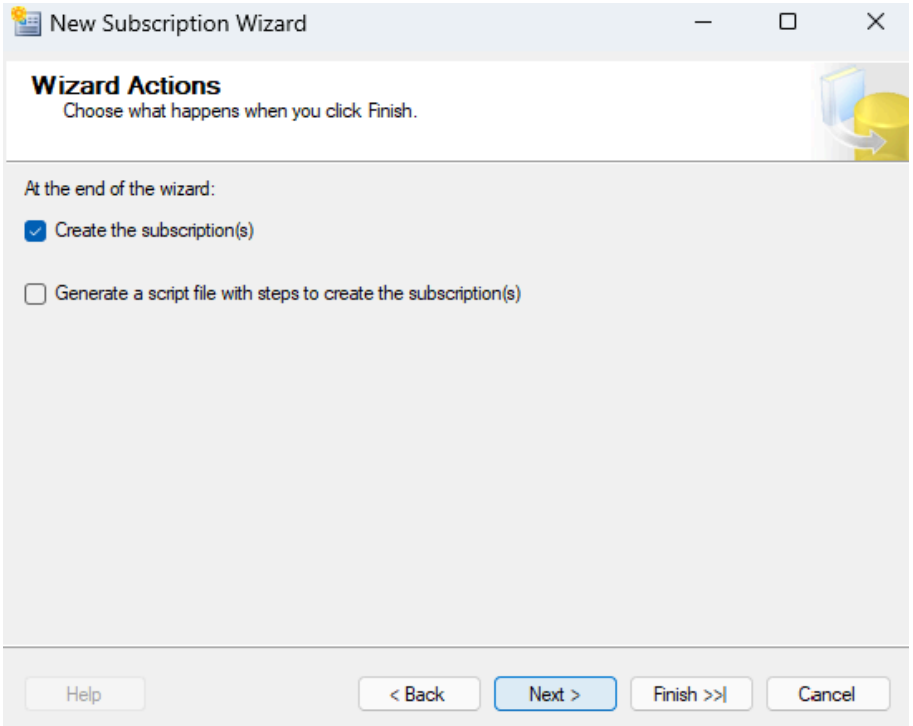
Bước 8: Chọn “Next” và set up tiếp như ảnh bên dưới



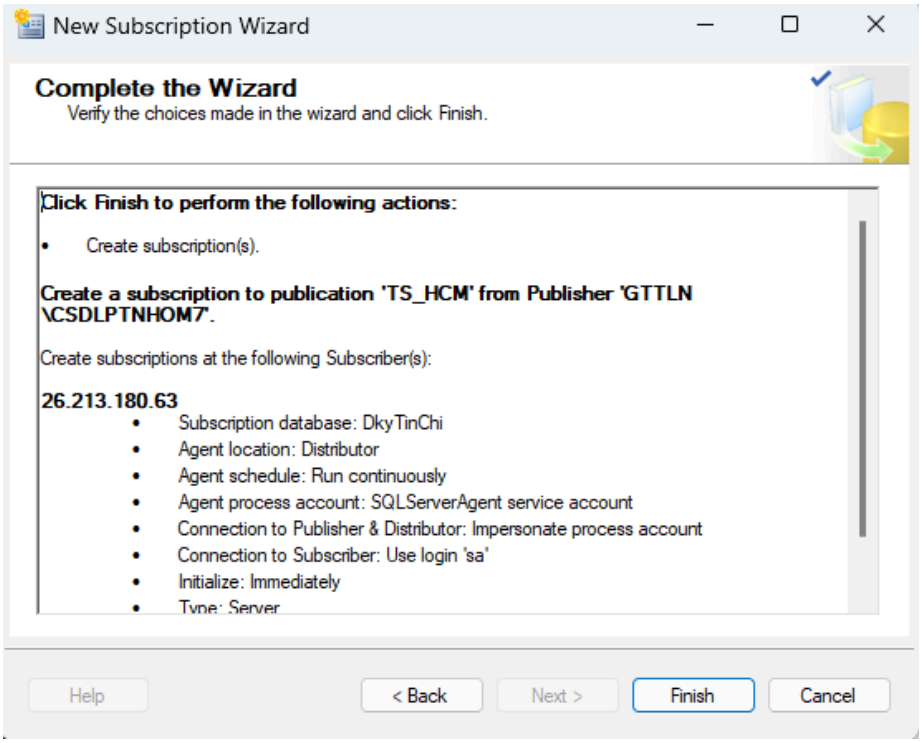
Bước 9: Tiếp tục chọn “Next”



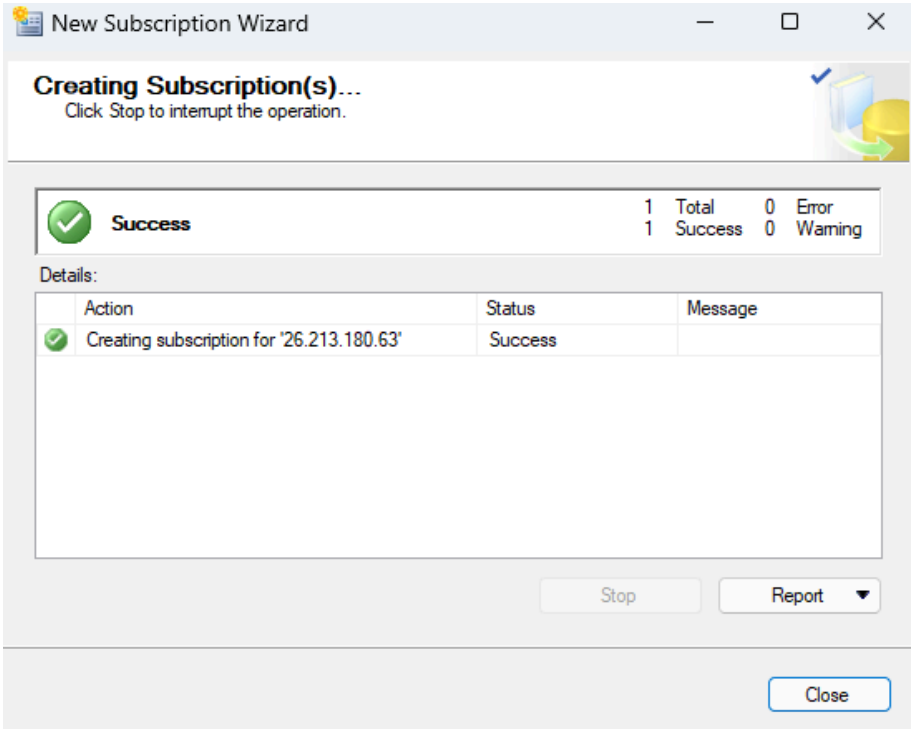
Bước 10: Tiếp tục chọn "Next" và tích chọn Create the subsripstion



Bước 11: Chọn Next và Finish

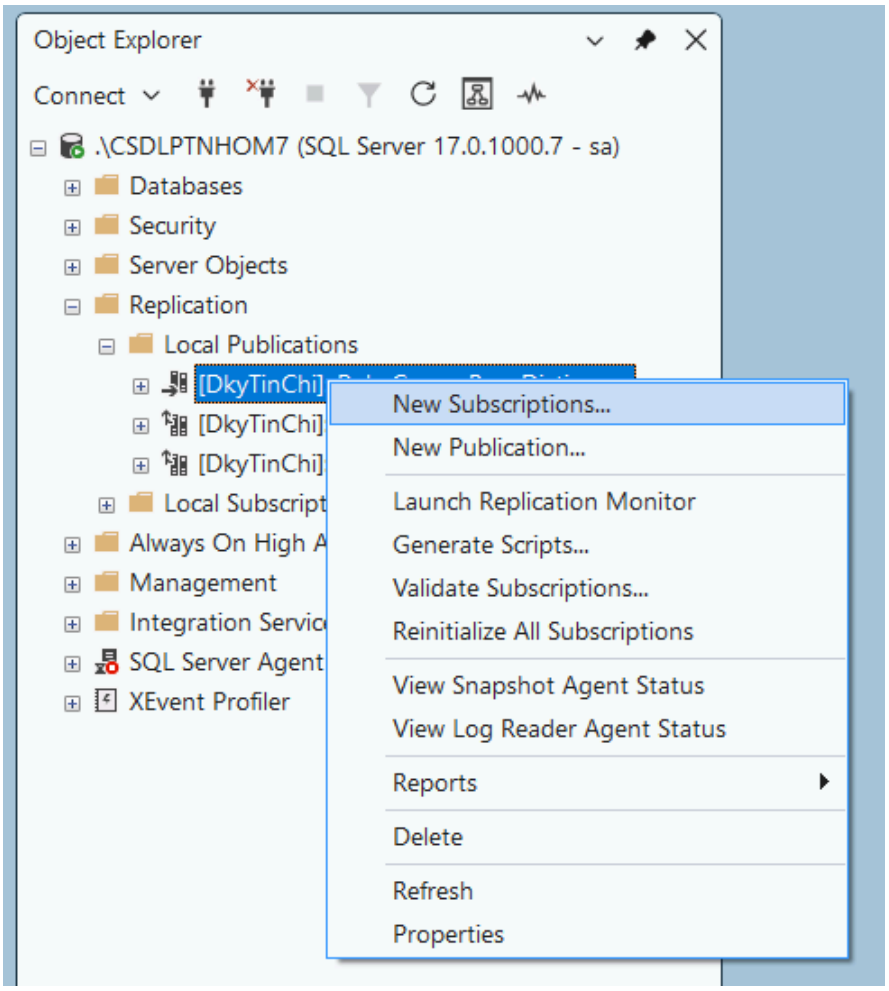


Bước 12: Thành công

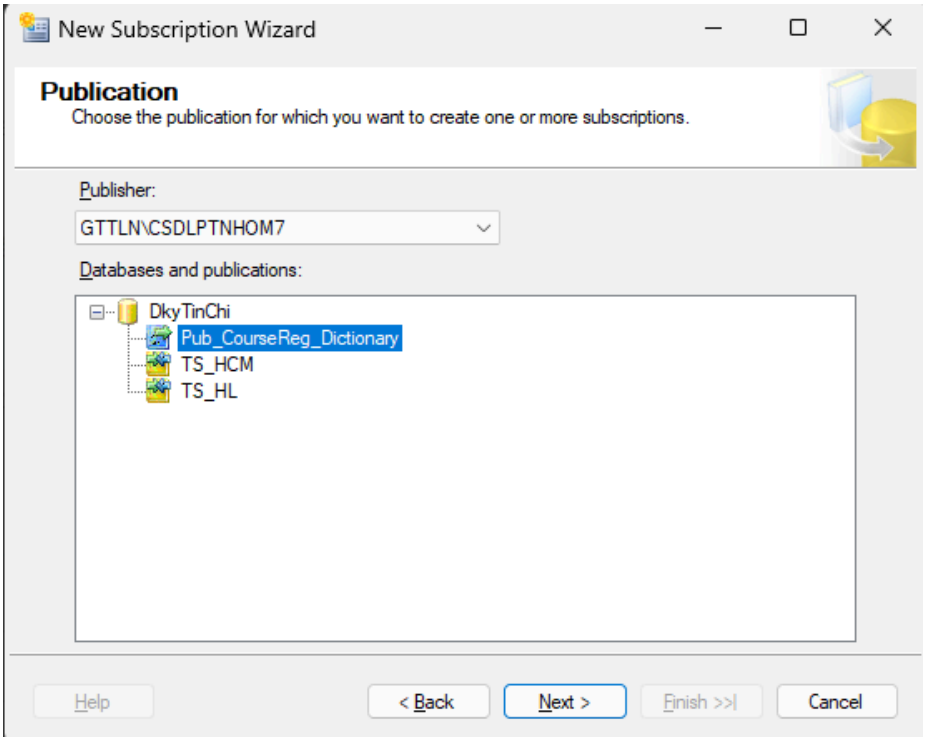


Lúc này database bên máy trạm đã nhận được thông tin bảng và dữ liệu bảng. Nhưng ở đây bị thiếu bảng subject, term, prerequisite, timeslot nên ta thêm bảng nhân bản vào nữa

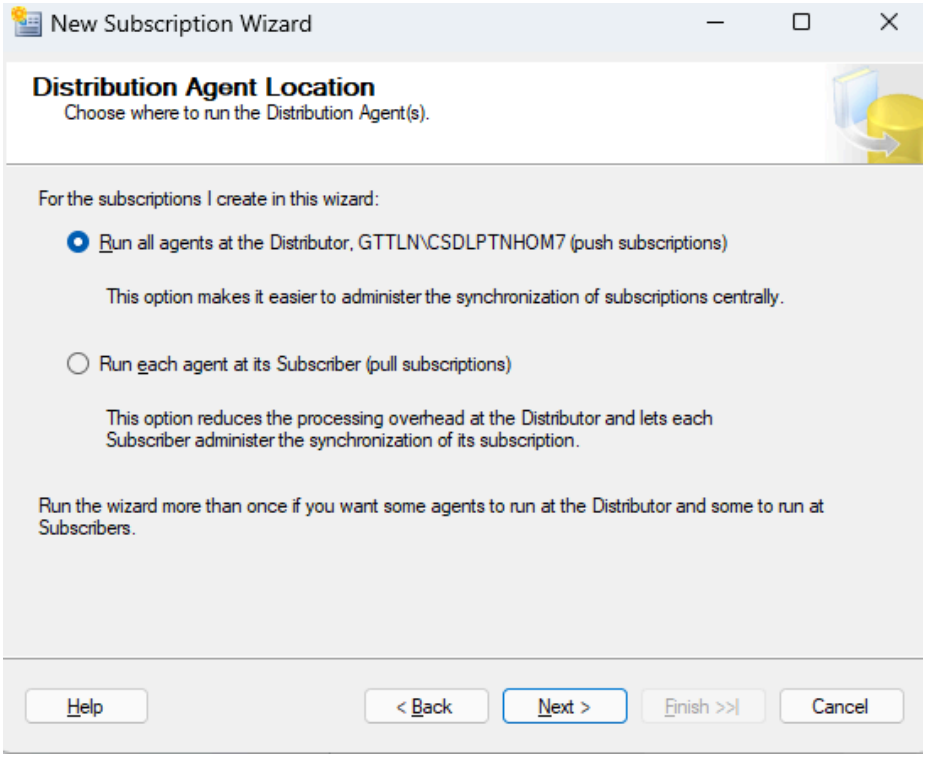
Bước 13: Chọn “New Subscriptions”



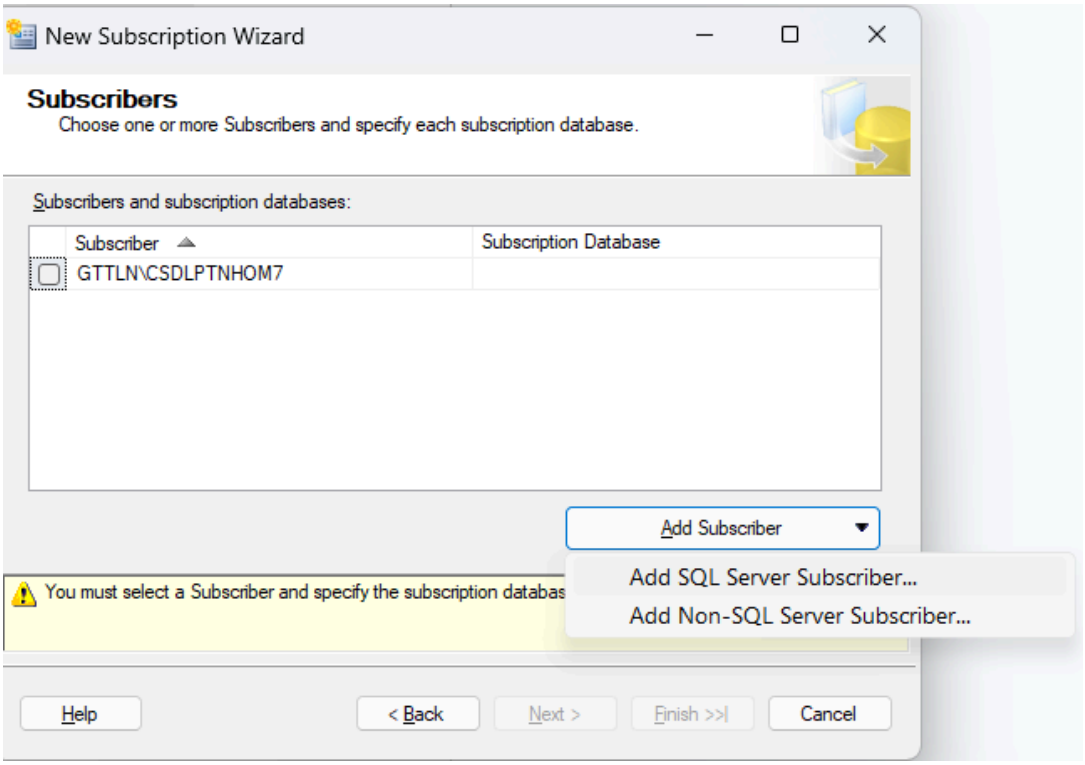
Bước 14: Chọn như ảnh dưới và Next

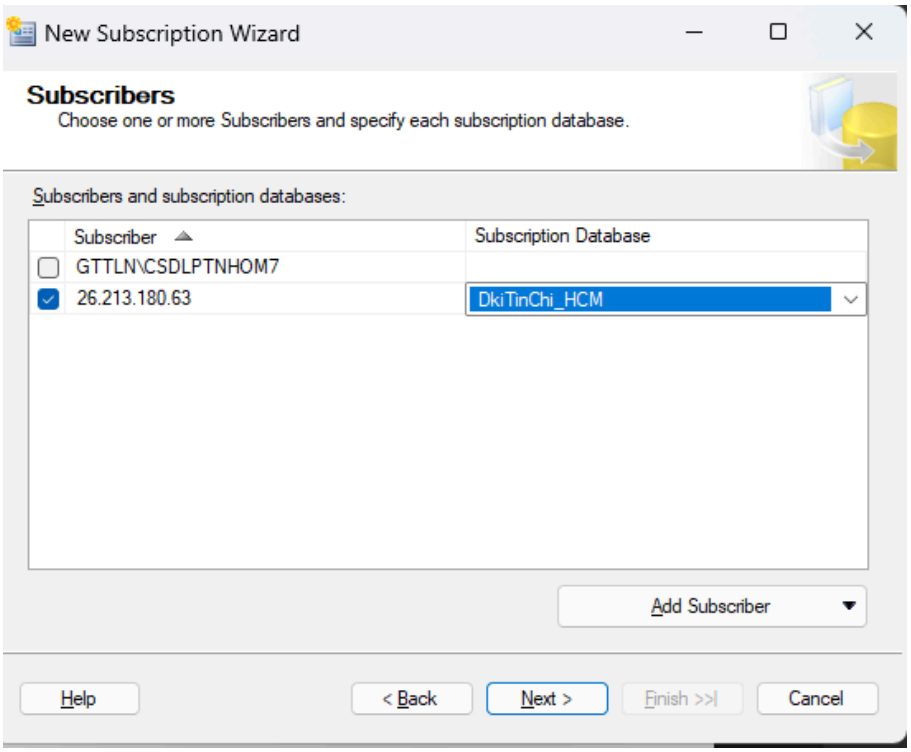


Bước 15: Chọn “Run all ...” như hình bên dưới

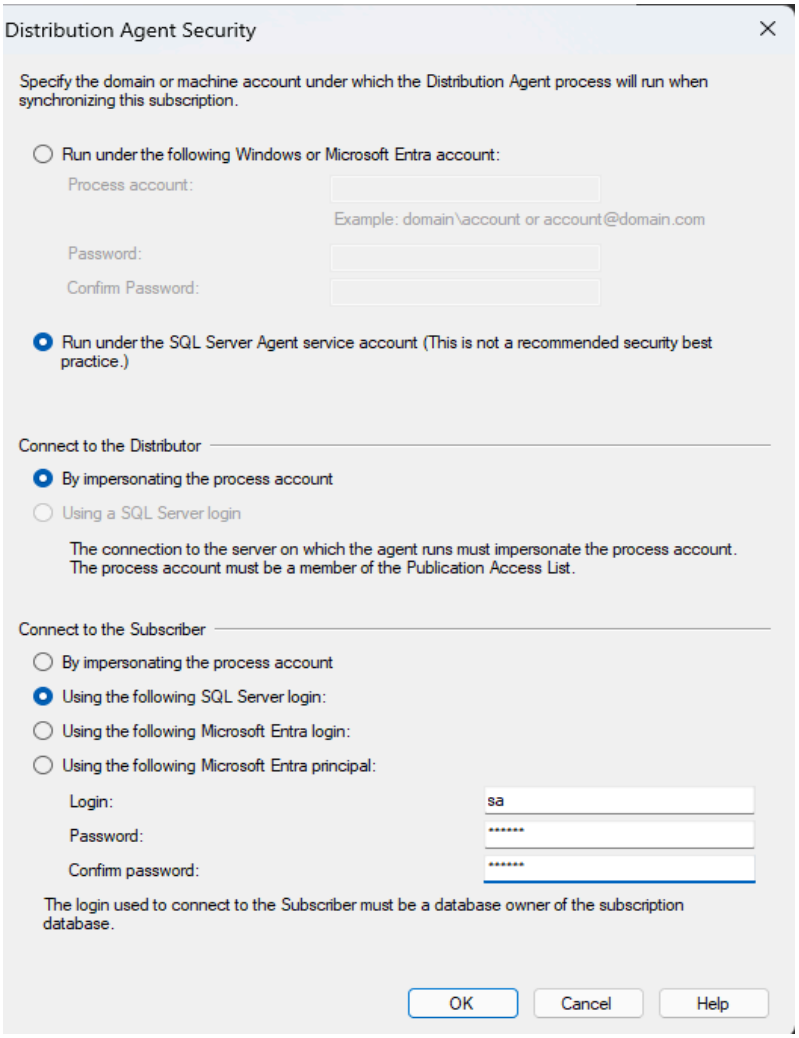


Bước 16: Chọn "Next" click vào "Add Subcriber" và chọn "Add SQL Server Subscriber"

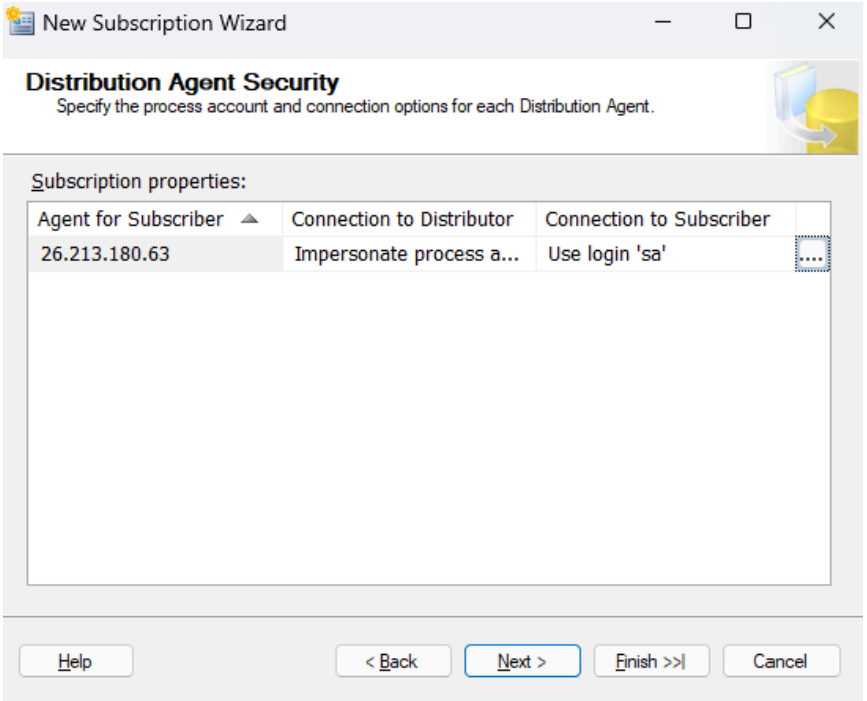




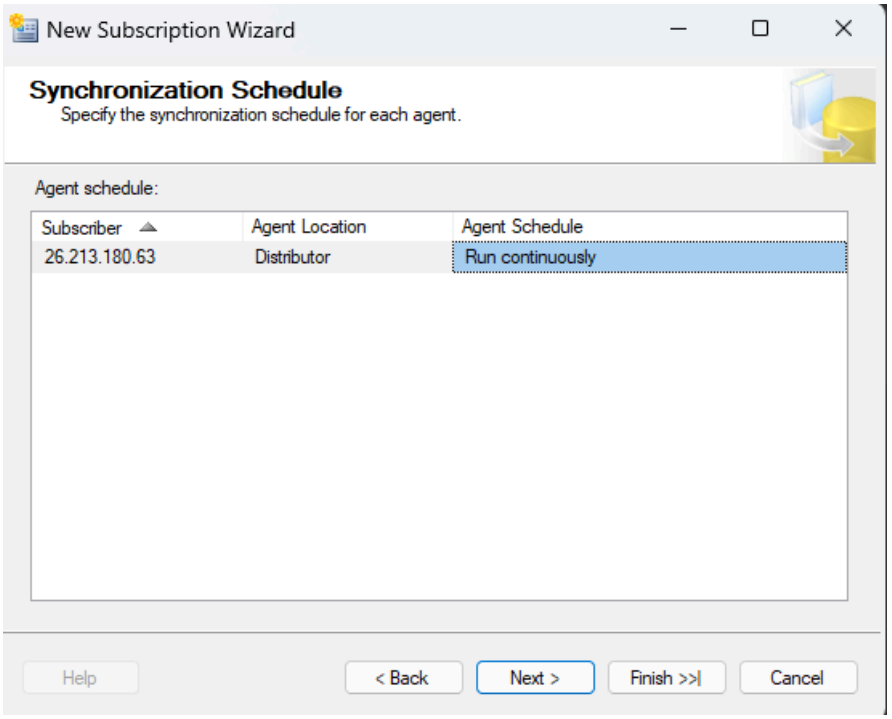
Bước 17: Click “Next” và click set up như hình bên dưới

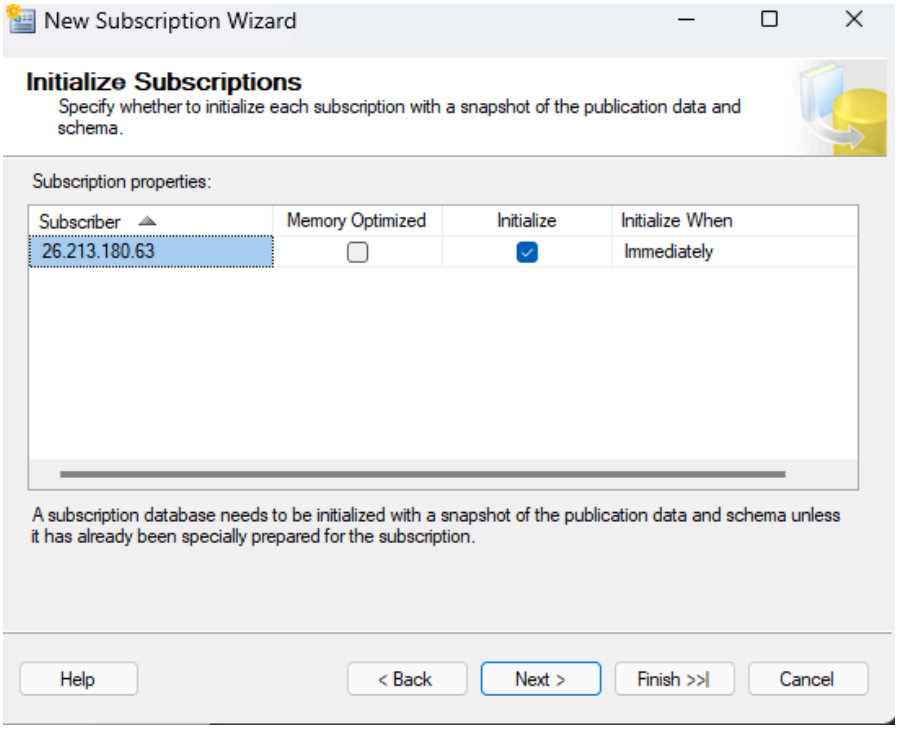


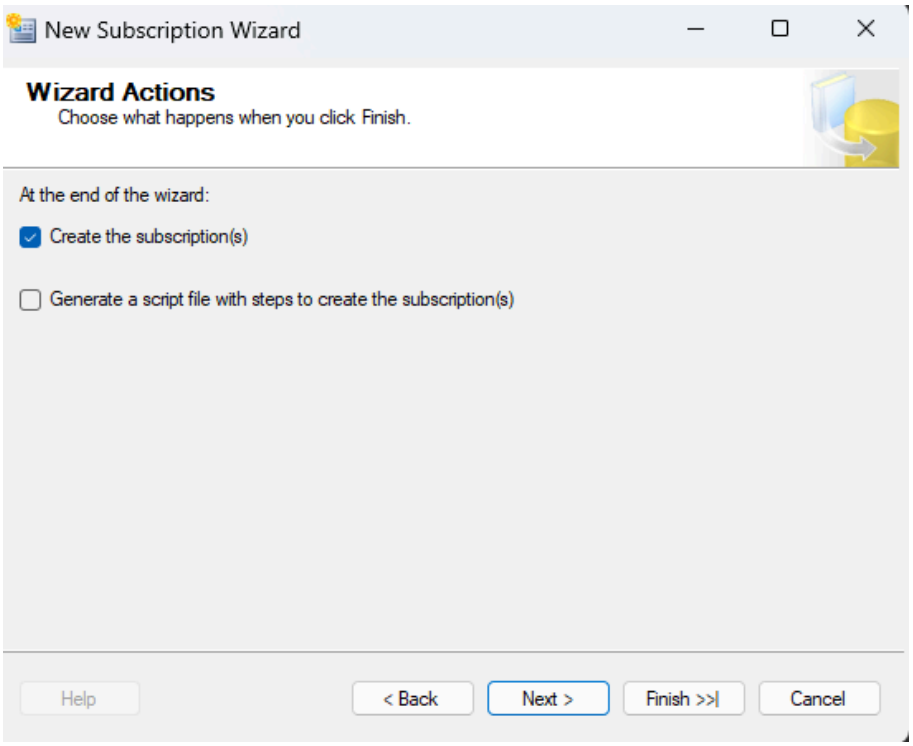
Bước 18: Đăng nhập tài khoản và OK



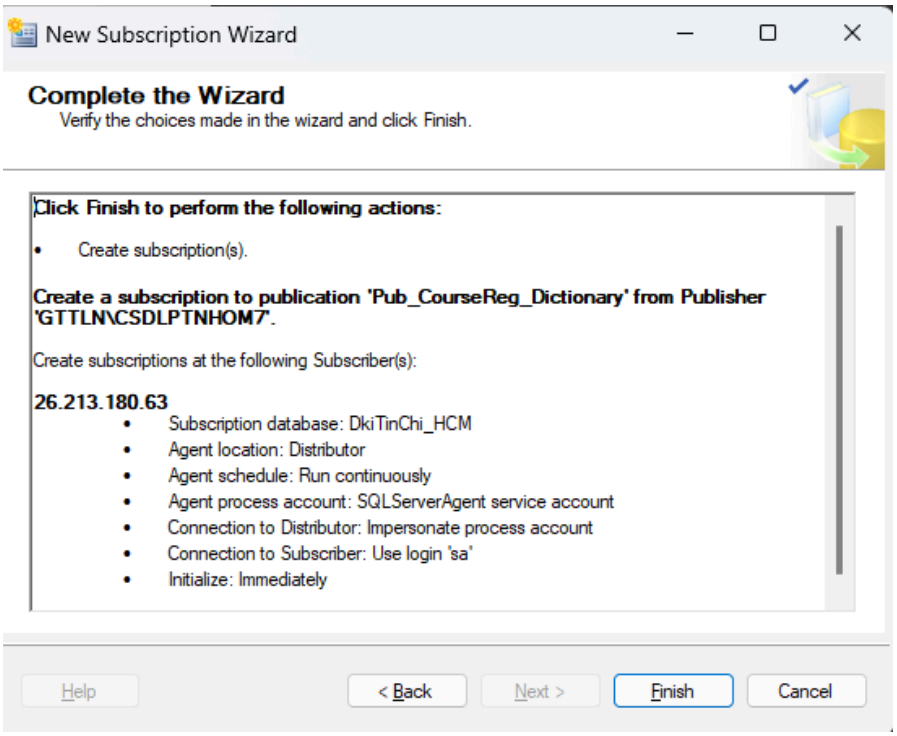
Bước 19: Note phải có continuous ĐỂ ĐỒNG BỘ TỨC THỜI







Bước 20: Bấm Finish và thành công



Làm tương tự các trụ sở khác Với cách làm tương tự như vậy ta đã hoàn toàn phân tán được dữ liệu đến các server, mỗi server đều đã nhận được database tương ứng

IX. Cấu hình đồng bộ dữ liệu giữa máy chủ và máy trạm

1. Đồng bộ dữ liệu từ máy chủ xuống máy trạm

Bên máy chủ chèn 1 dữ liệu thuộc về bảng dùng chung

```sql
USE DkyTinChi;
GO

INSERT INTO subject(ID_subject, name_subject, number_of_credit)
VALUES ('SUB_AUTO_DOWN', N'Môn tự xuống trạm', 3);
```

```
(1 row affected)

Completion time: 2026-05-30T17:55:19.5966806+07:00
```

Kiểm tra bên máy trạm :

```sql
USE CSDL_HL;
GO

SELECT *
FROM subject
WHERE ID_subject = 'SUB_AUTO_DOWN';
```

ID_subject	name_subject	number_of_credit
SUB_AUTO_DOWN	Môn tự xuống trạm	3

Kết quả : Bên máy trạm đã nhận được dữ liệu mà máy chủ chèn vào

2. Đồng bộ dữ liệu từ máy trạm lên máy chủ

Phía máy trạm chèn dữ liệu của bảng room

```sql
USE CSDL_HL;
GO

INSERT INTO room(ID_room, name_room, capacity, ID_headquarter)
VALUES ('R_AUTO_UP_HL', N'Phòng tự lên chủ', 60, 'HQHL');
```

```
(1 row affected)

Completion time: 2026-05-30T18:01:42.6832043+07:00
```

Kiểm tra phía máy chủ :

```sql
USE DkyTinChi;
GO

SELECT *
FROM room
WHERE ID_room = 'R_AUTO_UP_HL';
```

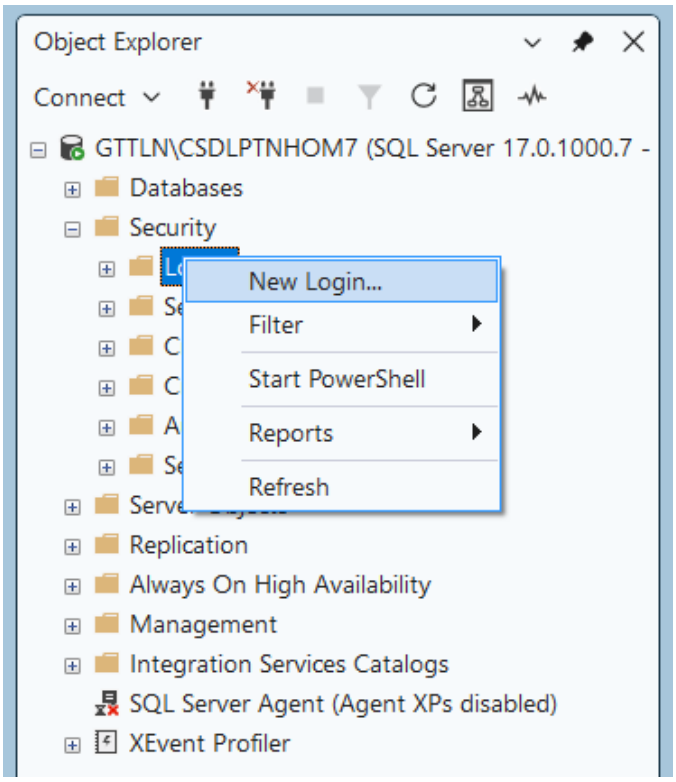
	ID_room	name_room	capacity	ID_headquarter	rowguid
1	R_AUTO_UP_HL	Phòng tự lên chủ	60	HQHL	230159F0-165C-F111-8568-D4D853CF58EA

Kết quả : Máy chủ đã nhận được dữ liệu mà máy trạm chèn vào

X. Tạo Link Server

Mục tiêu : Linked Server được tạo nhằm thiết lập kết nối truy vấn giữa server trung tâm và các server máy trạm trong hệ thống cơ sở dữ liệu phân tán. Thông qua Linked Server, server trung tâm có thể truy cập trực tiếp dữ liệu tại các site Hòa Lạc và TP.HCM để thực hiện các truy vấn phân tán .

Bước 1: Tạo new Login trong phần kết nối của cơ sở dữ liệu của dung để tạo một kết nối riêng trên máy chủ A.

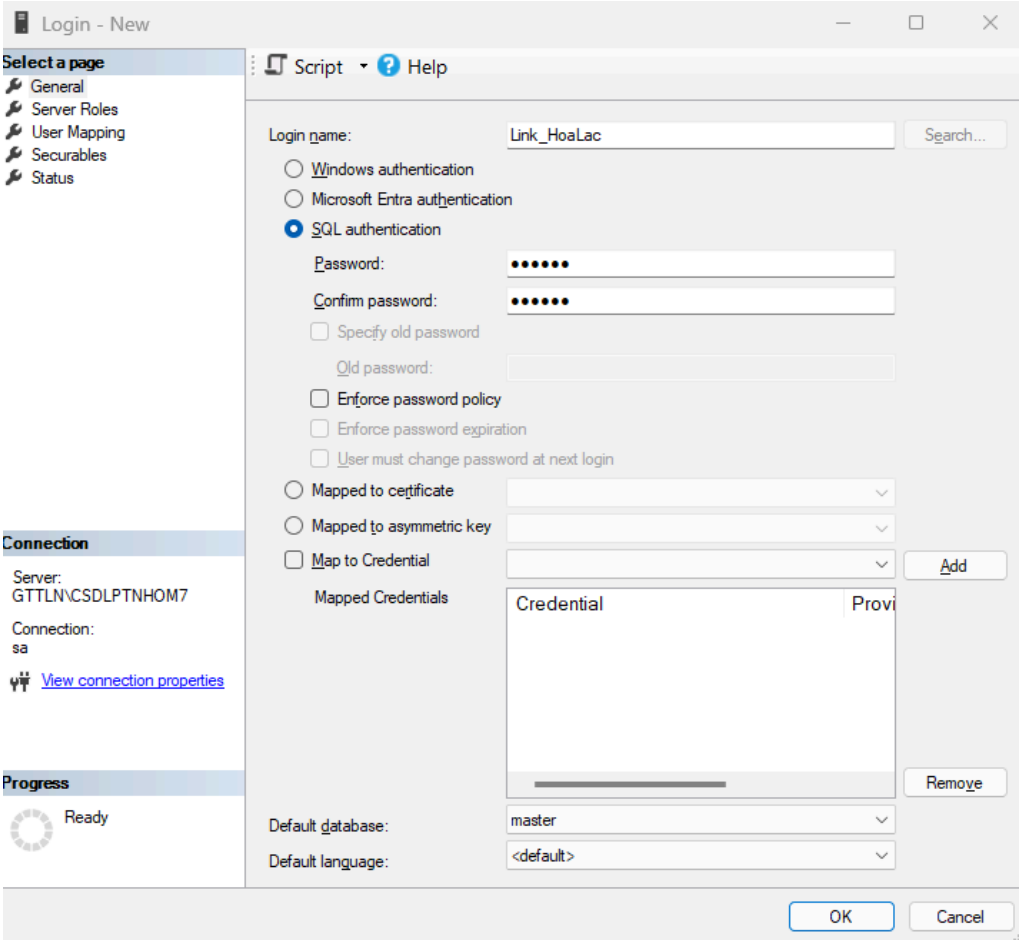


Bước 2:

- Đặt tên kết nối ở đây ta đặt là Link_HoaLac

- Chọn SQL Server đặt password và confirm password ở đây chọn là 123456

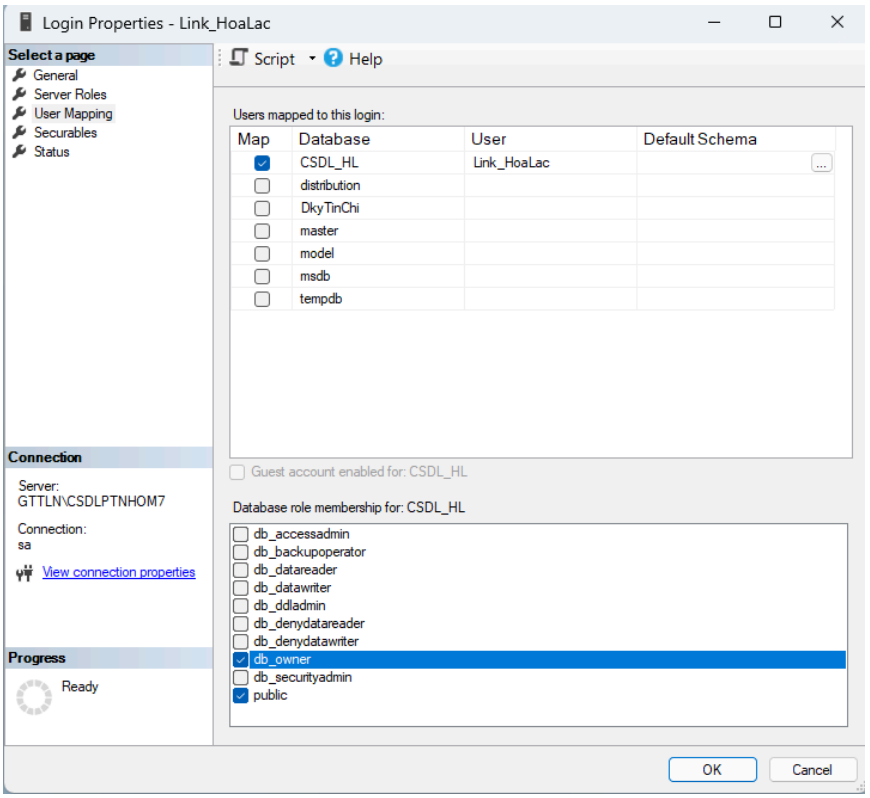
- Bỏ chọn Enforce password policy



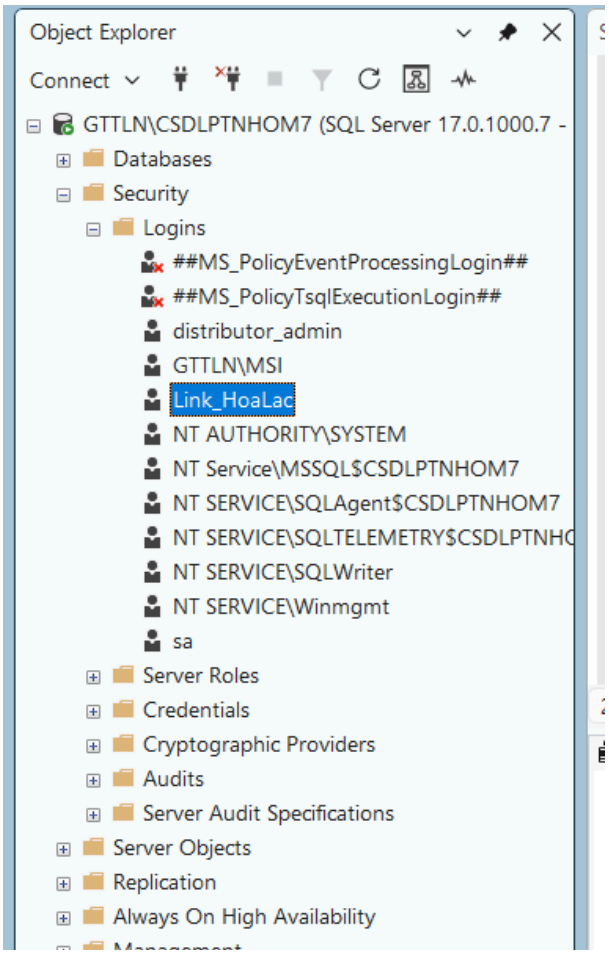
Bước 3:

- Chọn User Mapping Chọn cơ sở dữ liệu muốn chia sẻ (trong trường hợp này chọn CSDL_HL).

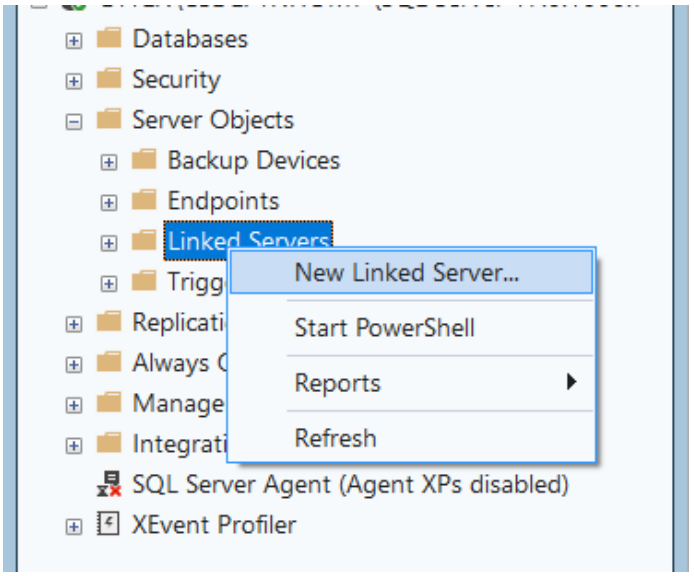
- Tiếp theo chọn public và db_ower để cập nhật dữ liệu sang máy B với tất cả dữ liệu có khả năng thao tác thêm, bớt, sửa, xóa, tạo mới.



Bước 4: Bấm OK để hoàn tất việc tạo trên máy A.

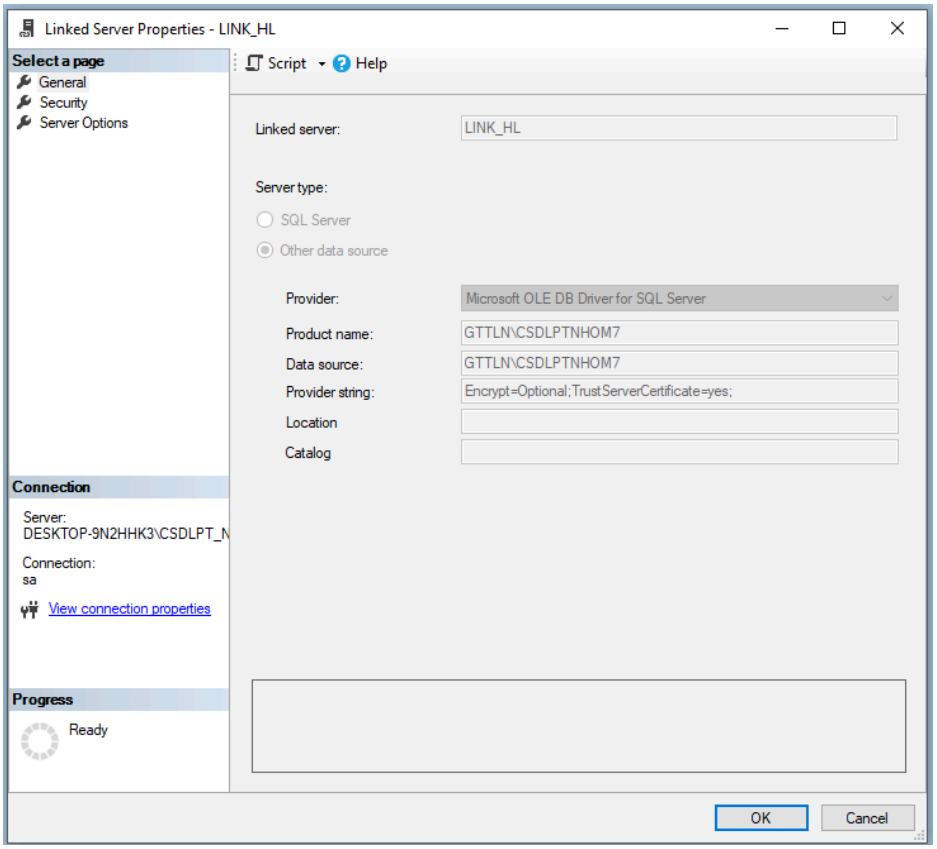


Bước 5: Trên máy B chọn new Linked Server ở Linked Servers



Bước 6: Thực hiện cài đặt Linked Server đó. Nhập Product name và Provider string là nơi mà máy B muốn chọc dữ liệu đến (máy A)

Encrypt=Optional;TrustServerCertificate=yes;



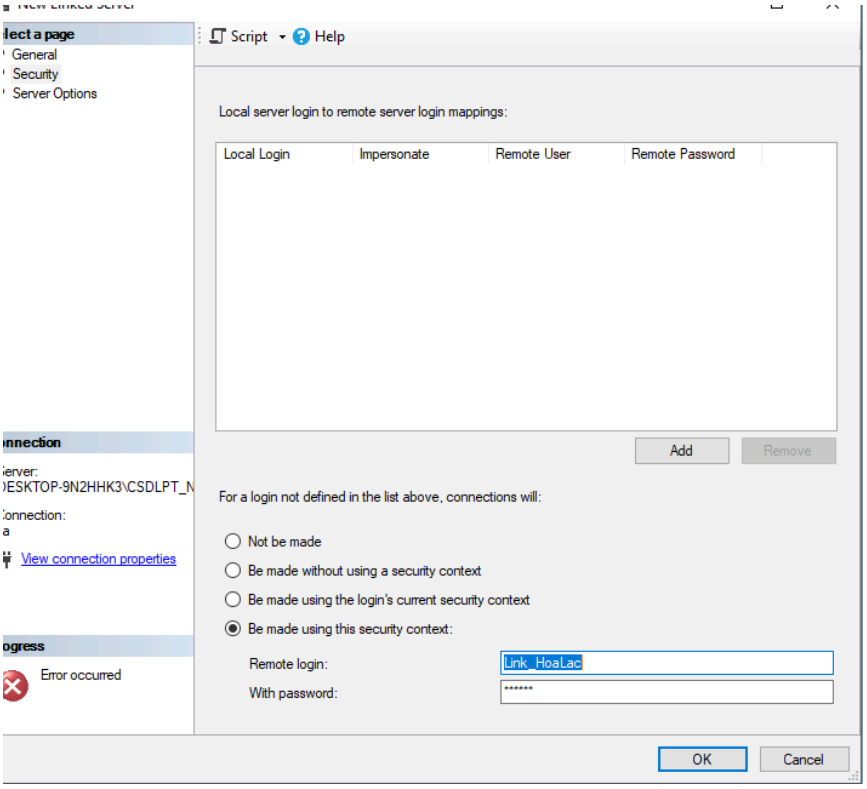
Bước 7:

- Chọn vào phần Security

- Chọn Be made using this security context

- Chọn Remote login: Link_HoaLac

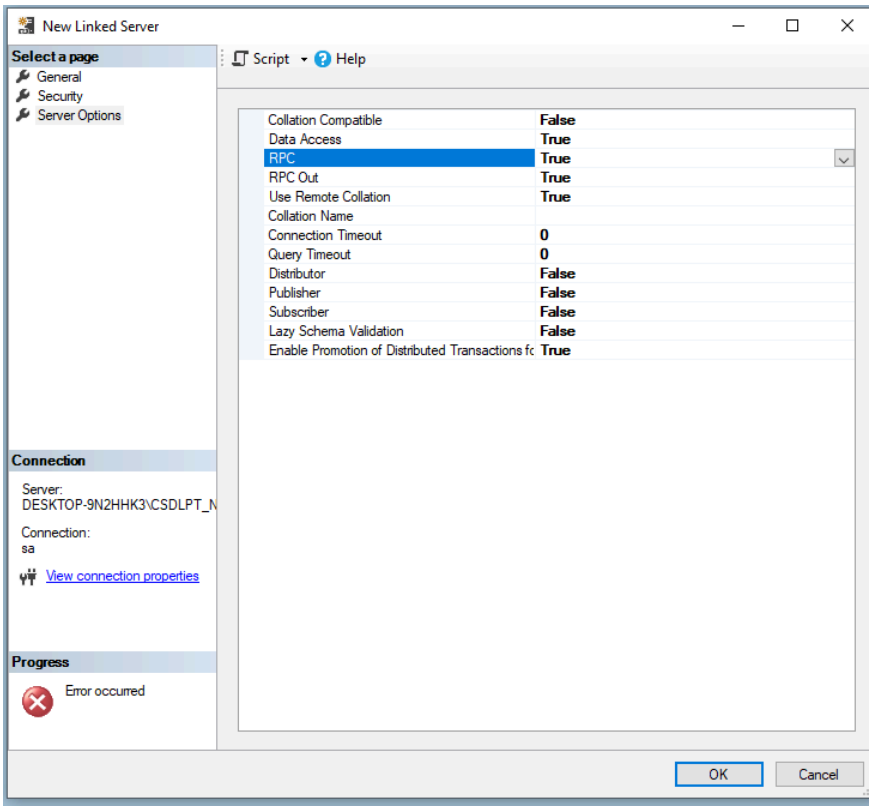
- Chọn password: 123456 (phần mật khẩu đã tạo từ trước).



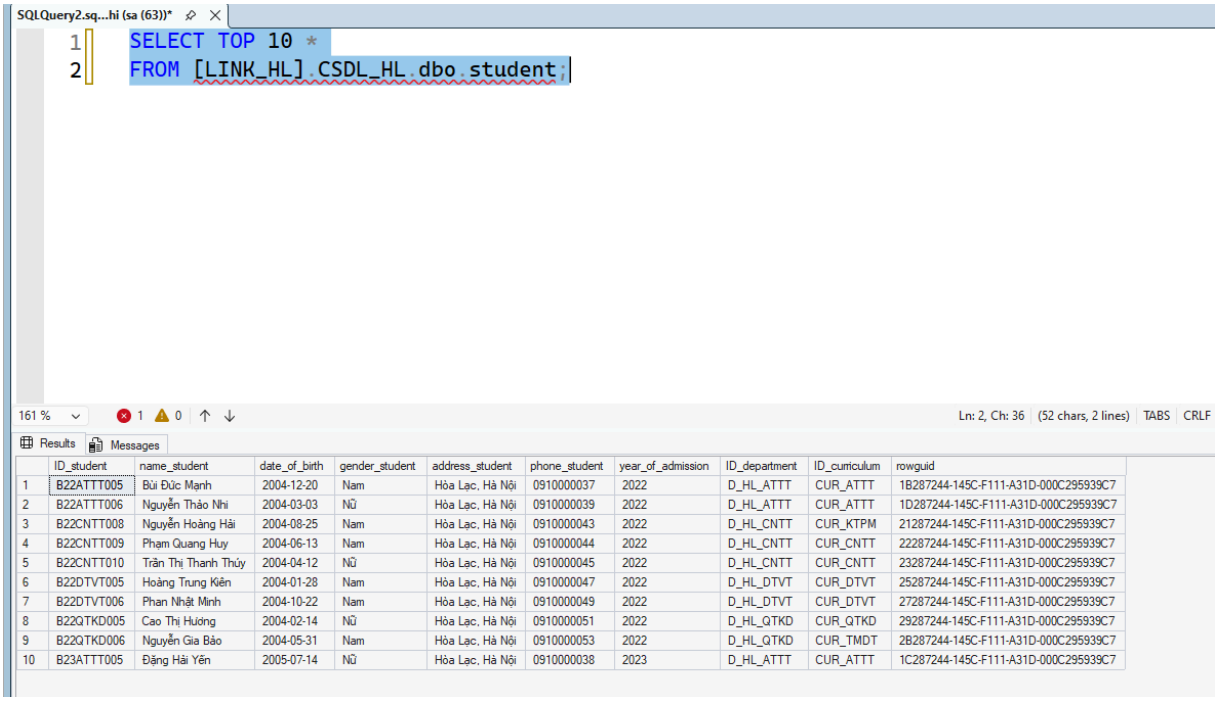
Bước 8:

- Phần Server Options

- Chọn tất cả các phần ở trên là True và nhấn OK



Bước 9: Tiến hành thử truy vấn dữ liệu trên máy A từ máy B



	ID_student	name_student	date_of_birth	gender_student	address_student	phone_student	year_of_admission	ID_department	ID_curriculum	rowguid
1	B22ATTT005	Bùi Đức Mạnh	2004-12-20	Nam	Hòa Lạc, Hà Nội	0910000037	2022	D_HL_ATTT	CUR_ATTT	1B287244-145C-F111-A31D-000C295939C7
2	B22ATTT006	Nguyễn Thảo Nhi	2004-03-03	Nữ	Hòa Lạc, Hà Nội	0910000039	2022	D_HL_ATTT	CUR_ATTT	1D287244-145C-F111-A31D-000C295939C7
3	B22CNTT008	Nguyễn Hoàng Hải	2004-08-25	Nam	Hòa Lạc, Hà Nội	0910000043	2022	D_HL_CNTT	CUR_KTPM	21287244-145C-F111-A31D-000C295939C7
4	B22CNTT009	Phạm Quang Huy	2004-06-13	Nam	Hòa Lạc, Hà Nội	0910000044	2022	D_HL_CNTT	CUR_CNTT	22287244-145C-F111-A31D-000C295939C7
5	B22CNTT010	Trần Thị Thanh Thúy	2004-04-12	Nữ	Hòa Lạc, Hà Nội	0910000045	2022	D_HL_CNTT	CUR_CNTT	23287244-145C-F111-A31D-000C295939C7
6	B22DTVT005	Hoàng Trung Kiên	2004-01-28	Nam	Hòa Lạc, Hà Nội	0910000047	2022	D_HL_DTVT	CUR_DTVT	25287244-145C-F111-A31D-000C295939C7
7	B22DTVT006	Phan Nhật Minh	2004-10-22	Nam	Hòa Lạc, Hà Nội	0910000049	2022	D_HL_DTVT	CUR_DTVT	27287244-145C-F111-A31D-000C295939C7
8	B22QTKD005	Cao Thị Hương	2004-02-14	Nữ	Hòa Lạc, Hà Nội	0910000051	2022	D_HL_QTKD	CUR_QTKD	29287244-145C-F111-A31D-000C295939C7
9	B22QTKD006	Nguyễn Gia Bảo	2004-05-31	Nam	Hòa Lạc, Hà Nội	0910000053	2022	D_HL_QTKD	CUR_TMDT	2B287244-145C-F111-A31D-000C295939C7
10	B23ATTT005	Đặng Hải Yến	2005-07-14	Nữ	Hòa Lạc, Hà Nội	0910000038	2023	D_HL_ATTT	CUR_ATTT	1C287244-145C-F111-A31D-000C295939C7

X. Trigger

Trong hệ thống cơ sở dữ liệu phân tán quản lý đăng ký học phần nhiều cơ sở, Trigger được sử dụng như một lớp kiểm soát bổ sung ở tầng cơ sở dữ liệu. Các Trigger giúp phát hiện các thao tác không hợp lệ khi thêm hoặc cập nhật dữ liệu đăng ký học phần, lịch học và phòng học.

Trigger không thay thế Stored Procedure xử lý đăng ký học phần, mà đóng vai trò hỗ trợ kiểm soát toàn vẹn dữ liệu và ghi nhận nhật ký thao tác. Trong hệ thống, các nghiệp vụ quan trọng như đăng ký học phần, hủy đăng ký học phần và xử lý đồng thời vẫn được thực hiện chính thông qua Stored Procedure kết hợp Transaction và cơ chế khóa dữ liệu.

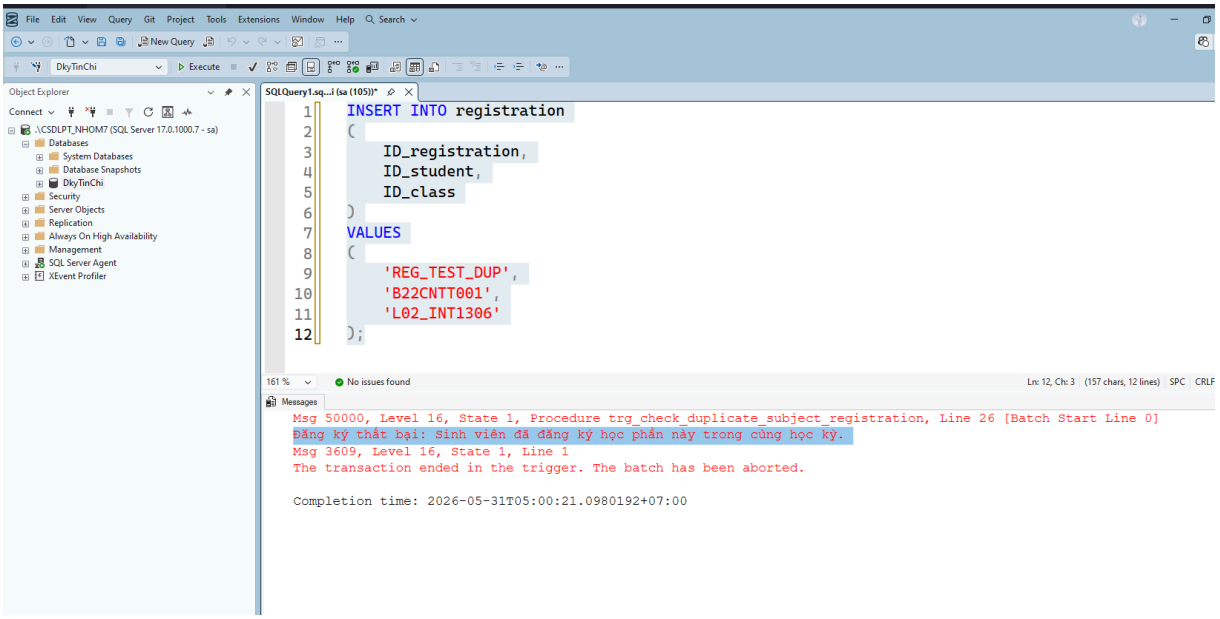
1. Trigger kiểm tra đăng ký trùng học phần

- Tên Trigger: trg_check_duplicate_subject_registration
- Bảng thao tác (Target Table): registration
- Sự kiện kích hoạt (Event): AFTER INSERT, UPDATE
- Mục đích: Giám sát và ngăn chặn tình trạng một sinh viên đăng ký vào nhiều lớp học phần khác nhau nhưng lại thuộc cùng một môn học (Subject) trong cùng một học kỳ (Term). Ràng buộc này giúp đảm bảo sinh viên không học trùng một học phần nhiều lần trong cùng một kỳ học.

Mô tả thuật toán và logic xử lý: Quá trình kiểm soát của Trigger được thực hiện tuần tự qua các bước sau:

- Bước 1 (Ghi nhận dữ liệu đầu vào): Khi có một giao dịch thêm mới (INSERT) hoặc cập nhật (UPDATE) dữ liệu đăng ký học phần, hệ thống sẽ tự động bắt lấy bản ghi vừa thao tác thông qua bảng ảo inserted.
- Bước 2 (Xác định thông tin học phần đang đăng ký): Hệ thống thực hiện liên kết (Join) bản ghi mới trong inserted với bảng class để trích xuất ra Mã môn học (ID_subject) và Mã học kỳ (ID_term) tương ứng với lớp học phần mà sinh viên đang yêu cầu đăng ký.
- Bước 3 (Kiểm tra lịch sử đăng ký): Hệ thống tiếp tục truy vấn ngược lại bảng registration kết hợp với bảng class để quét toàn bộ các lớp học phần mà sinh viên này đã đăng ký thành công trước đó (trạng thái REGISTERED), loại trừ bản ghi đang xét hiện tại.
- Bước 4 (Đánh giá điều kiện và ra quyết định):
    - Trường hợp vi phạm: Nếu phát hiện Mã môn học và Mã học kỳ của lớp đang xin đăng ký *hoàn toàn trùng khớp* với một lớp học phần đã đăng ký thành công trong lịch sử quét ở Bước 3, hệ thống xác định giao dịch không hợp lệ.
    - Xử lý vi phạm: Trigger lập tức ngắt tiến trình, trả về thông báo lỗi: *"Đăng ký thất bại: Sinh viên đã đăng ký học phần này trong cùng học kỳ."* Đồng thời, thực thi lệnh ROLLBACK TRANSACTION để hủy bỏ toàn bộ thao tác, đảm bảo không có dữ liệu sai lệch nào được ghi vào cơ sở dữ liệu.

Kiểm thử Trigger



Nhận xét: Trigger đã phát hiện trường hợp sinh viên đăng ký trùng học phần trong cùng học kỳ và tự động hủy thao tác ghi dữ liệu không hợp lệ.

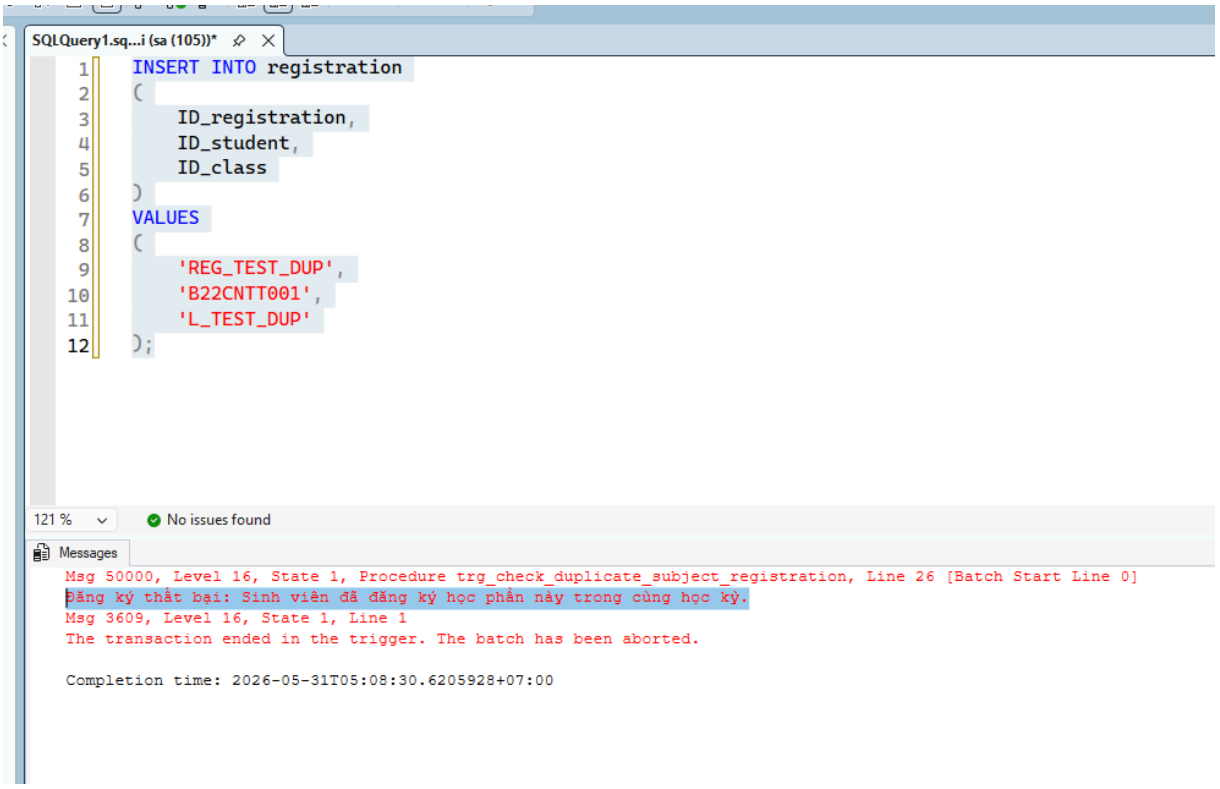
2. Trigger kiểm tra trùng lịch học

- Tên Trigger: trg_check_schedule_conflict
- Bảng thao tác (Target Table): registration
- Sự kiện kích hoạt (Event): AFTER INSERT, UPDATE
- Mục đích: Giám sát và ngăn chặn tình trạng sinh viên đăng ký vào một lớp học phần có lịch học bị xung đột (trùng lặp thời gian) với các lớp học phần khác đã được đăng ký thành công trước đó. Ràng buộc này dựa trên việc đối chiếu ngày học và ca học, giúp đảm bảo tính khả thi trong thời khóa biểu cá nhân của sinh viên.

Mô tả thuật toán và logic xử lý: Quá trình kiểm soát của Trigger được thực thi tuần tự thông qua các bước như sau:

- Bước 1 (Ghi nhận yêu cầu đăng ký): Bắt lấy bản ghi chứa thông tin đăng ký học phần vừa được thêm mới hoặc cập nhật thông qua bảng tạm inserted.
- Bước 2 (Trích xuất lịch học của lớp đang yêu cầu): Hệ thống thực hiện phép nối (Join) bản ghi từ inserted với bảng session (Buổi học) thông qua khóa ngoại Mã lớp học phần (ID_class). Mục đích là để lấy ra chính xác Ngày học (study_date) và Ca học (ID_timeslot) của lớp mà sinh viên đang muốn đăng ký.
- Bước 3 (Truy xuất thời khóa biểu hiện tại của sinh viên): Tiếp tục truy vấn bảng registration để quét toàn bộ các lớp học phần sinh viên này đã đăng ký thành công (trạng thái REGISTERED, ngoại trừ giao dịch đang xét). Sau đó, nối tiếp với bảng session để lấy ra chi tiết lịch học (Ngày học và Ca học) của tất cả các lớp cũ này.
- Bước 4 (Đánh giá xung đột và ra quyết định):
    - Trường hợp vi phạm: Thuật toán tiến hành so khớp lịch học lấy được ở Bước 2 và Bước 3. Nếu phát hiện tồn tại bất kỳ một buổi học nào có Ngày học và Ca học trùng khớp hoàn toàn với thời khóa biểu hiện có của sinh viên, hệ thống xác định đây là một giao dịch gây xung đột lịch.
    - Xử lý vi phạm: Trigger ngay lập tức ngắt tiến trình, trả về thông báo lỗi: *"Đăng ký thất bại: Lịch học bị trùng với lớp học phần đã đăng ký."* Sau đó, lệnh ROLLBACK TRANSACTION được thực thi để hủy toàn bộ quá trình, đưa cơ sở dữ liệu về trạng thái an toàn ban đầu.

Kiểm thử Trigger



Nhận xét: Trigger giúp đảm bảo mỗi sinh viên không thể đăng ký hai lớp học phần có lịch học trùng nhau.

3. Trigger kiểm tra sĩ số lớp học phần

- Tên Trigger: trg_check_class_capacity
- Bảng thao tác (Target Table): registration
- Sự kiện kích hoạt (Event): AFTER INSERT, UPDATE
- Mục đích: Giám sát, kiểm soát và đảm bảo số lượng sinh viên đăng ký thực tế vào một lớp học phần không bao giờ được phép vượt quá giới hạn sĩ số tối đa quy định của lớp học đó.

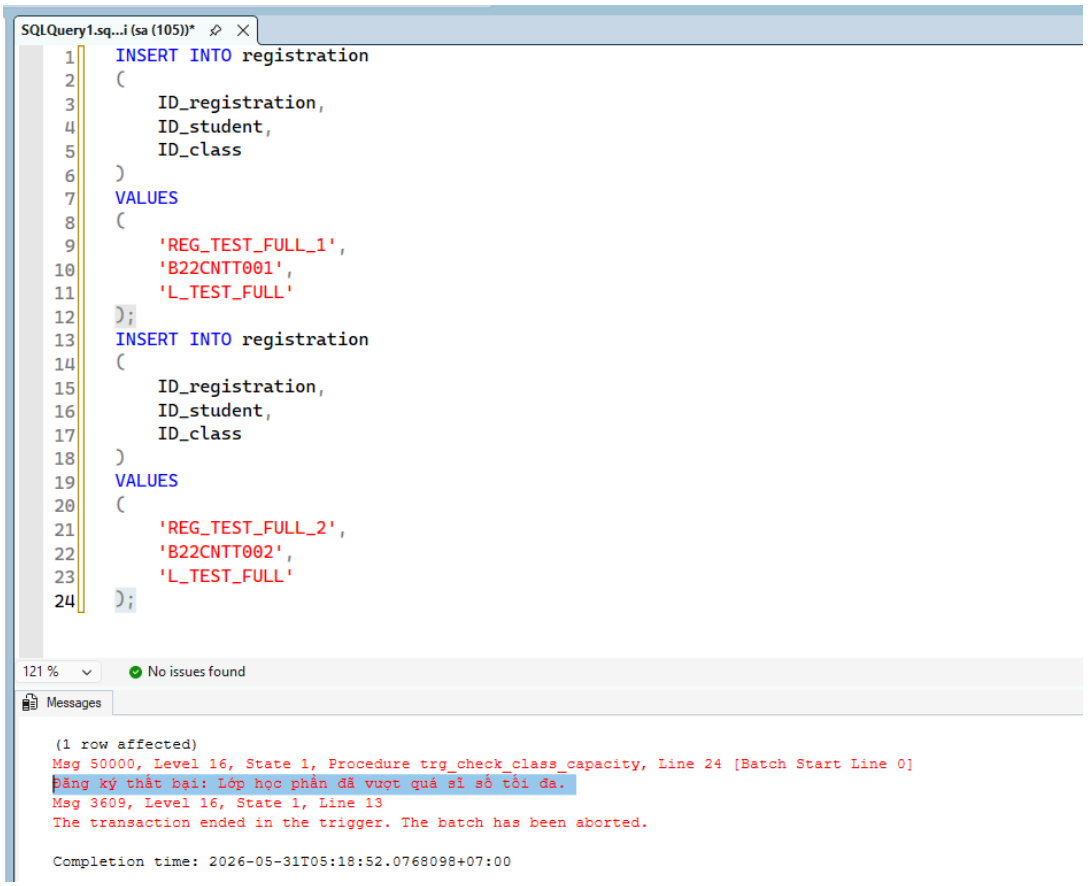
Mô tả thuật toán và logic xử lý: Quá trình kiểm soát sức chứa của lớp học phần được Trigger thực thi tuần tự qua các bước sau:

- Bước 1 (Lắng nghe sự kiện): Hệ thống được kích hoạt ngay khi có một giao dịch thêm mới (INSERT) hoặc cập nhật (UPDATE) dữ liệu vào bảng đăng ký học phần (registration).
- Bước 2 (Thống kê số lượng thực tế): Hệ thống thực hiện đếm tổng số lượng bản ghi (tương đương số lượng sinh viên) đang có trạng thái là 'REGISTERED' (đã đăng ký thành công) trong bảng registration. Dữ liệu này được gom nhóm

(GROUP BY) theo từng Mã lớp học phần (ID_class) để tính ra chính xác số lượng sinh viên hiện tại của mỗi lớp.

- Bước 3 (Đối chiếu dữ liệu): Thuật toán tiến hành phép nối (Join) kết quả thống kê vừa thu được ở Bước 2 với bảng class (lớp học phần). Mục đích là để lấy thông số Sĩ số tối đa (max_students) đã được cấu hình từ trước cho lớp học đó.
- Bước 4 (Đánh giá điều kiện và ra quyết định):
    - Trường hợp vi phạm: Hệ thống so sánh trực tiếp "Tổng số sinh viên đã đăng ký" với "Sĩ số tối đa". Nếu phát hiện số lượng đăng ký thực tế lớn hơn mức trần cho phép (total_registered > max_students), hệ thống xác định giao dịch gây ra tình trạng quá tải (over-allocation).
    - Xử lý vi phạm: Trigger lập tức ngắt tiến trình thao tác, trả về thông báo lỗi: *"Đăng ký thất bại: Lớp học phần đã vượt quá sĩ số tối đa."* Ngay sau đó, lệnh ROLLBACK TRANSACTION được thực thi để hủy bỏ hoàn toàn phiên giao dịch, đưa dữ liệu an toàn trở về trạng thái trước khi có lệnh đăng ký.

Kiểm thử Trigger



Nhận xét: Trigger hỗ trợ đảm bảo số lượng sinh viên đăng ký không vượt quá sĩ số tối đa của lớp học phần.

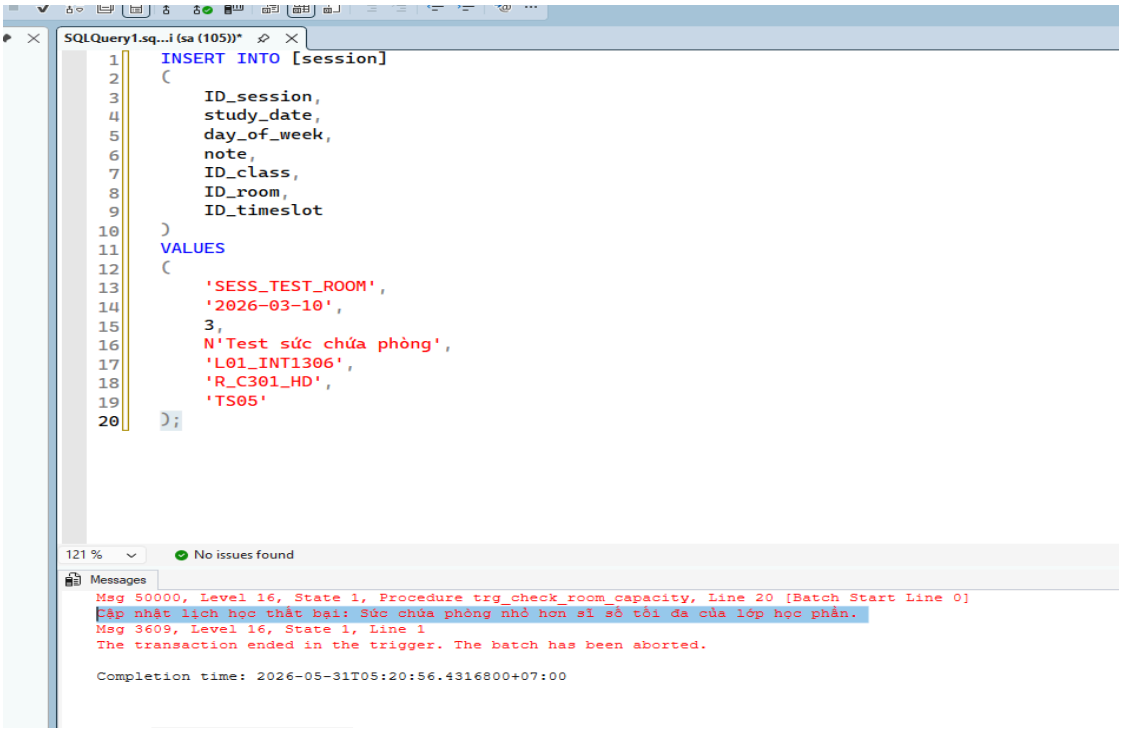
4. Trigger kiểm tra sức chứa

- Tên Trigger: trg_check_room_capacity
- Bảng thao tác (Target Table): session (Buổi học / Lịch học)
- Sự kiện kích hoạt (Event): AFTER INSERT, UPDATE
- Mục đích: Giám sát quá trình phân bổ tài nguyên vật lý, đảm bảo sức chứa của phòng học được gán luôn đáp ứng đủ hoặc vượt mức quy mô tối đa (sĩ số tối đa) của lớp học phần. Ràng buộc này giúp ngăn chặn tình trạng xếp lớp vào phòng quá nhỏ, không đủ không gian học tập cho sinh viên.

Mô tả thuật toán và logic xử lý**:** Quá trình kiểm soát và đối chiếu năng lực phòng học được hệ thống thực thi theo các bước như sau:

- Bước 1 (Ghi nhận yêu cầu xếp lịch): Hệ thống tự động bắt lấy thông tin chi tiết của buổi học (session) vừa được yêu cầu thêm mới hoặc cập nhật thông qua bảng tạm inserted.
- Bước 2 (Trích xuất thông số quy mô lớp học): Tiến hành phép nối (Join) bản ghi từ bảng inserted với bảng class thông qua khóa ngoại Mã lớp học phần (ID_class). Mục đích là để trích xuất chỉ số Sĩ số tối đa (max_students) đã cấu hình cho lớp học đó.
- Bước 3 (Xác định năng lực phòng học): Tiếp tục nối bản ghi với bảng room thông qua khóa ngoại Mã phòng học (ID_room) để lấy ra chỉ số Sức chứa thực tế (capacity) của căn phòng đang được dự kiến sử dụng.
- Bước 4 (Đối chiếu và ra quyết định):
    - Trường hợp vi phạm: Thuật toán tiến hành so sánh trực tiếp hai thông số thu thập được ở Bước 2 và Bước 3. Nếu phát hiện Sĩ số tối đa của lớp học phần lớn hơn mức Sức chứa của phòng học (max_students > capacity), hệ thống nhận định đây là một phân bổ phi logic về mặt vật lý.
    - Xử lý vi phạm: Trigger ngay lập tức ngắt tiến trình cập nhật, trả về thông báo lỗi: *"Cập nhật lịch học thất bại: Sức chứa phòng nhỏ hơn sĩ số tối đa của lớp học phần."*. Ngay sau đó, câu lệnh ROLLBACK TRANSACTION được gọi để hủy bỏ toàn bộ phiên giao dịch, từ chối việc ghi nhận lịch học không hợp lệ vào cơ sở dữ liệu.

Kiểm thử Trigger



Nhận xét: Trigger giúp đảm bảo việc xếp lịch học phù hợp với sức chứa phòng học tại từng cơ sở.

X. Stored Procedures

1. Procedure kiểm tra điều kiện đăng ký học phần

- Tên Thủ tục (Procedure Name): usp_CheckRegisterCondition
- Tham số đầu vào (Input Parameters):
  - @ID_student (Kiểu chuỗi): Mã định danh duy nhất của sinh viên yêu cầu đăng ký.
  - @ID_class (Kiểu chuỗi): Mã định danh duy nhất của lớp học phần được chọn để đăng ký.
  - @ID_headquarter (Kiểu chuỗi, mặc định NULL): Mã cơ sở/trụ sở đào tạo. Tham số này dùng để kiểm tra tính hợp lệ về mặt vị trí phân mảnh của dữ liệu.
- Mục đích: Đóng vai trò là một "bộ lọc điều kiện" tiên quyết trước khi thực hiện ghi nhận một giao dịch đăng ký vào cơ sở dữ liệu. Thủ tục này tích hợp chuỗi kiểm tra đa lớp (về mặt nhân thân, thời gian, không gian, tính hợp lệ nghiệp vụ và ràng buộc hệ thống) để đưa ra quyết định sinh viên đó có đủ tư cách pháp lý và kỹ thuật để tham gia lớp học phần hay không.

Mô tả giải thuật và quy trình logic: Thủ tục vận hành theo cơ chế kiểm tra tuần tự (Sequential Validation Pipeline). Tại mỗi chặng, nếu dữ liệu không thỏa mãn, thủ tục

sẽ lập tức ngắt tiến trình và trả về trạng thái lỗi (is_valid = 0) cùng thông báo cụ thể. Quy trình chi tiết gồm các bước sau:

- Bước 1 (Xác thực thông tin và vị trí của Sinh viên):
    - *Logic xử lý:* Hệ thống thực hiện phép nối (Join) bảng dữ liệu sinh viên (student) với bảng khoa/viện (department) dựa trên mã khoa. Thuật toán đối chiếu xem mã sinh viên đầu vào có tồn tại hay không. Nếu @ID_headquarter được chỉ định, hệ thống sẽ kiểm tra thêm xem khoa của sinh viên này có thuộc đúng cơ sở đào tạo đó hay không.
    - *Xử lý lỗi:* Nếu sinh viên không tồn tại hoặc không nằm trong cơ sở được chọn, thủ tục trả về kết quả is_valid = 0 kèm thông báo: *"Sinh viên không tồn tại hoặc không thuộc cơ sở được chọn."* và kết thúc thủ tục.
- Bước 2 (Kiểm tra trạng thái lớp và thời gian mở cổng đăng ký):
    - *Logic xử lý:* Thực hiện nối chuỗi giữa các bảng lớp học phần (class), học kỳ (term), giảng viên (teacher) và khoa (department). Thuật toán rà soát xem lớp học phần có thỏa mãn đồng thời các điều kiện: Trạng thái đang mở (class_status = 'OPEN'), thời gian hiện tại (GETDATE()) nằm trong khoảng thời gian cho phép đăng ký tín chỉ của học kỳ (reg_open đến reg_close), và giảng viên phụ trách lớp phải thuộc đúng cơ sở dữ liệu đang quản lý (nếu có lọc theo @ID_headquarter).
    - *Xử lý lỗi:* Nếu lớp đã đóng, quá hạn hoặc sai cơ sở, thủ tục trả về kết quả is_valid = 0 kèm thông báo: *"Lớp không mở đăng ký, ngoài thời gian đăng ký hoặc không thuộc cơ sở được chọn."* và kết thúc thủ tục.
- Bước 3 (Kiểm tra đăng ký trùng lặp lớp học phần):
    - *Logic xử lý:* Truy vấn trực tiếp vào bảng lịch sử đăng ký (registration) để quét xem sinh viên (@ID_student) đã tồn tại bản ghi đăng ký thành công (trạng thái REGISTERED) đối với chính mã lớp học phần (@ID_class) này chưa.
    - *Xử lý lỗi:* Nếu đã tồn tại, thủ tục trả về kết quả is_valid = 0 kèm thông báo: *"Sinh viên đã đăng ký lớp học phần này."* và kết thúc thủ tục.
- Bước 4 (Kiểm tra xung đột thời khóa biểu - Trùng lịch học):
    - *Logic xử lý:* Kết nối bảng registration với bảng lịch học (session) của các lớp cũ mà sinh viên đã đăng ký thành công, sau đó đối chiếu chéo với lịch học (session) của lớp học phần mới đang xin đăng ký. Thuật toán tìm kiếm xem có bất kỳ buổi học nào trùng nhau về cả Ngày học (study_date) lẫn Ca học (ID_timeslot) hay không.

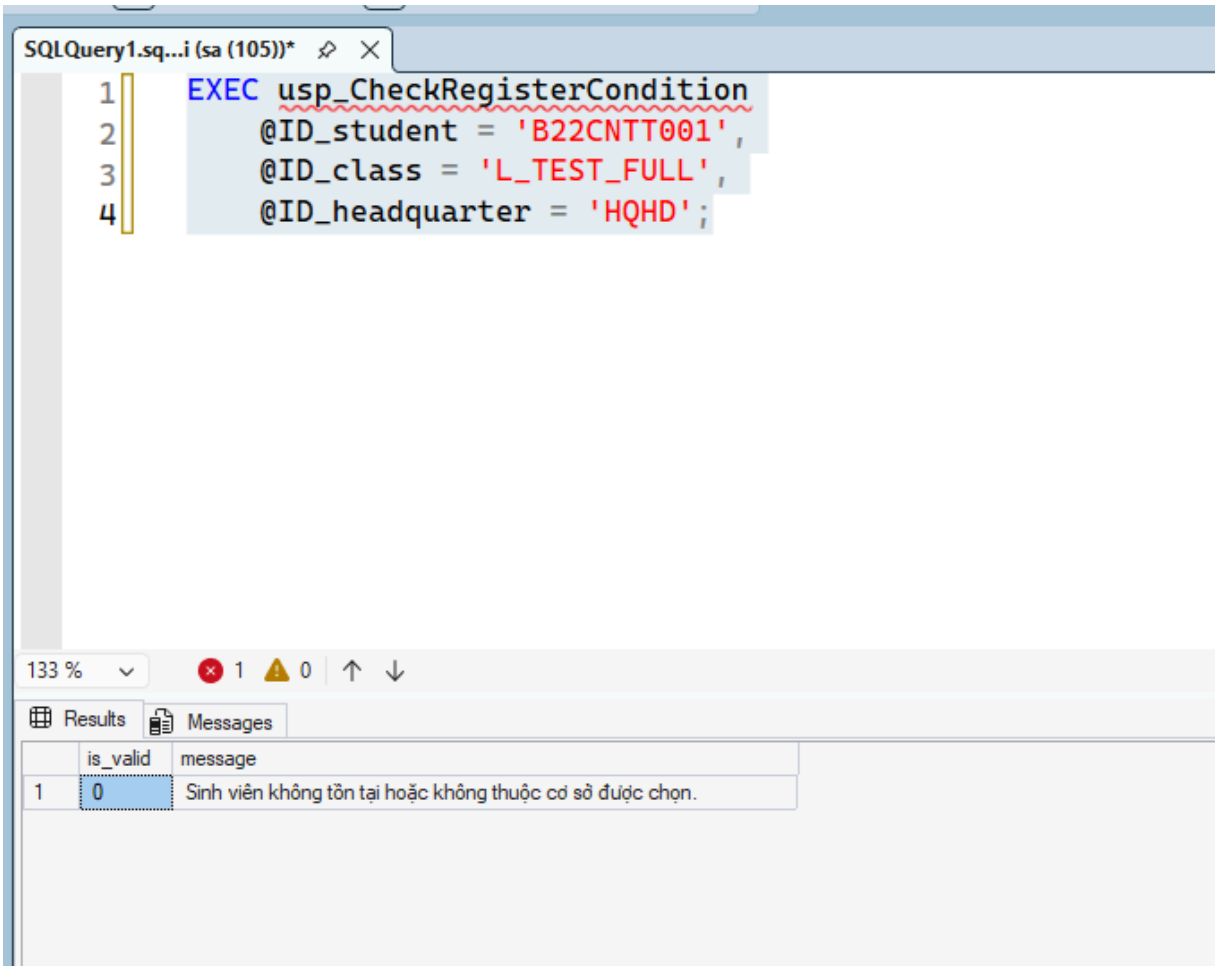
    - *Xử lý lỗi:* Nếu phát hiện có sự giao thoa/trùng lịch, thủ tục trả về kết quả is_valid = 0 kèm thông báo: *"Lịch học bị trùng với lớp học phần đã đăng ký."* và kết thúc thủ tục.

- Bước 5 (Kiểm tra giới hạn quy mô sĩ số phòng/lớp):
    - *Logic xử lý:* Truy vấn vào bảng dữ liệu lớp học phần (class) để so sánh thuộc tính "Số lượng đã đăng ký hiện tại" (number_of_registration) với mức trần "Sĩ số tối đa" (max_students).
    - *Xử lý lỗi:* Nếu số lượng hiện tại đã chạm hoặc vượt quá mức trần, thủ tục trả về kết quả is_valid = 0 kèm thông báo: *"Lớp học phần đã đủ sĩ số."* và kết thúc thủ tục.
- Bước 6 (Thông qua điều kiện - Thành công):
    - *Logic xử lý:* Khi luồng dữ liệu vượt qua tất cả 5 chốt kiểm tra nghiêm ngặt phía trên mà không bị ngắt giữa chừng, thủ tục đưa ra phán quyết cuối cùng, trả về kết quả thành công is_valid = 1 kèm thông báo: *"Đủ điều kiện đăng ký học phần."*

Kiểm thử Procedure và Nhận xét:

- Kịch bản kiểm thử: Thực hiện gọi thủ tục bằng cách truyền vào các bộ tham số khác nhau: Một bộ tham số hoàn toàn hợp lệ, một bộ tham số có mã sinh viên sai, một bộ có lịch học của lớp mới trùng với lớp sinh viên đang học, và một bộ ứng với lớp đã khóa sĩ số.
- Nhận xét: Thủ tục hoạt động như một tầng kiểm soát logic (Business Logic Layer) vô cùng chặt chẽ tại tầng cơ sở dữ liệu. Việc tích hợp nhiều bước kiểm tra vào một Procedure duy nhất giúp tối ưu hóa hiệu năng hệ thống mạng phân tán, giảm thiểu số lần gọi truy vấn từ ứng dụng máy trạm lên máy chủ, đồng thời ngăn chặn tuyệt đối các dữ liệu rác hoặc sai quy chế đào tạo tín chỉ lọt vào hệ thống.

Kiểm thử Procedure



2. Procedure đăng ký học phần

- Tên Thủ tục (Procedure Name): usp_RegisterClass
- Mục đích: Đây là chức năng trung tâm của hệ thống, đảm nhiệm toàn bộ vòng đời của một yêu cầu đăng ký tín chỉ từ sinh viên. Nó không chỉ kiểm tra các điều kiện nghiệp vụ mà còn áp dụng cơ chế khóa dữ liệu (Locking) kết hợp với Transaction để kiểm soát chặt chẽ vấn đề tranh chấp dữ liệu (xử lý đồng thời) trên môi trường phân tán.
- Tham số đầu vào (Input):
    - @ID_registration: Mã giao dịch đăng ký (ID tự sinh).
    - @ID_student: Mã sinh viên thực hiện giao dịch.
    - @ID_class: Mã lớp học phần mong muốn đăng ký.
    - @ID_headquarter: (Tùy chọn) Mã cơ sở đào tạo, dùng để đảm bảo tính phân mảnh cục bộ.

Mô tả giải thuật và logic xử lý (Bảo vệ bằng Transaction): Để đảm bảo tính nhất quán (ACID), toàn bộ luồng thực thi được đặt trong một khối BEGIN TRY ... BEGIN

TRAN. Nếu bất kỳ bước nào thất bại, hệ thống sẽ tự động bắt lỗi tại CATCH và thực thi ROLLBACK để bảo vệ dữ liệu.

- Bước 1 (Xác thực Sinh viên): Hệ thống kiểm tra sự tồn tại và tính hợp lệ của mã sinh viên @ID_student tại cơ sở @ID_headquarter. Nếu không hợp lệ, ném lỗi 50001 và ngắt giao dịch.
- Bước 2 (Xác thực Lớp học phần & Thời gian): Hệ thống rà soát trạng thái lớp học phần (OPEN) và đối chiếu thời gian hệ thống hiện tại (GETDATE()) có nằm trong khung thời gian đăng ký hợp lệ của học kỳ hay không. Nếu vi phạm, ném lỗi 50002.
- Bước 3 (Kiểm tra Lịch sử đăng ký trùng lặp): Quét bảng registration để đảm bảo sinh viên chưa từng đăng ký thành công lớp học phần này trước đó. Nếu phát hiện trùng, ném lỗi 50003.
- Bước 4 (Kiểm tra Xung đột thời khóa biểu): So khớp lịch học (Ngày học và Ca học) của lớp mới xin đăng ký với toàn bộ các lớp mà sinh viên đã đăng ký trước đó. Nếu xảy ra giao thoa, ném lỗi 50004.
- Bước 5 (Xử lý đồng thời & Cập nhật Sĩ số - Cực kỳ quan trọng):
    - *Logic khóa (Locking):* Đây là điểm mấu chốt ngăn chặn lỗi Over-allocation (vượt quá sĩ số). Hệ thống thực hiện lệnh UPDATE tăng trường number_of_registration lên 1 đơn vị. Tuy nhiên, lệnh này được đính kèm từ khóa chỉ thị WITH (UPDLOCK, ROWLOCK).
    - *Cơ chế hoạt động:* ROWLOCK sẽ khóa chính xác dòng dữ liệu của lớp học phần đó, UPDLOCK sẽ giữ khóa cập nhật cho đến khi Transaction kết thúc. Nếu có hàng trăm sinh viên cùng đăng ký 1 lớp tại 1 thời điểm, các giao dịch sẽ bị ép phải "xếp hàng" chờ đọc và cập nhật sĩ số một cách tuần tự, triệt tiêu hoàn toàn lỗi Lost Update.
    - *Xử lý lỗi:* Lệnh UPDATE này có kèm điều kiện number_of_registration < max_students. Do đó, nếu sĩ số đã đầy, lệnh UPDATE sẽ không tác động đến bất kỳ dòng nào (@@ROWCOUNT = 0). Lúc này, hệ thống sẽ ném lỗi 50005: *"Lớp học phần đã đủ sĩ số"* và Rollback.
- Bước 6 (Ghi nhận giao dịch): Nếu vượt qua Bước 5 (tức là lớp còn chỗ và đã được giữ chỗ thành công), hệ thống tiến hành INSERT bản ghi chính thức vào bảng registration với trạng thái 'REGISTERED'.
- Bước 7 (Hoàn tất): Gọi lệnh COMMIT để lưu vĩnh viễn các thay đổi vào cơ sở dữ liệu và giải phóng các khóa (UPDLOCK, ROWLOCK).

Kiểm thử Procedure

Chạy

EXEC usp_RegisterClass

@ID_registration = 'REG_TEST_HD_001',

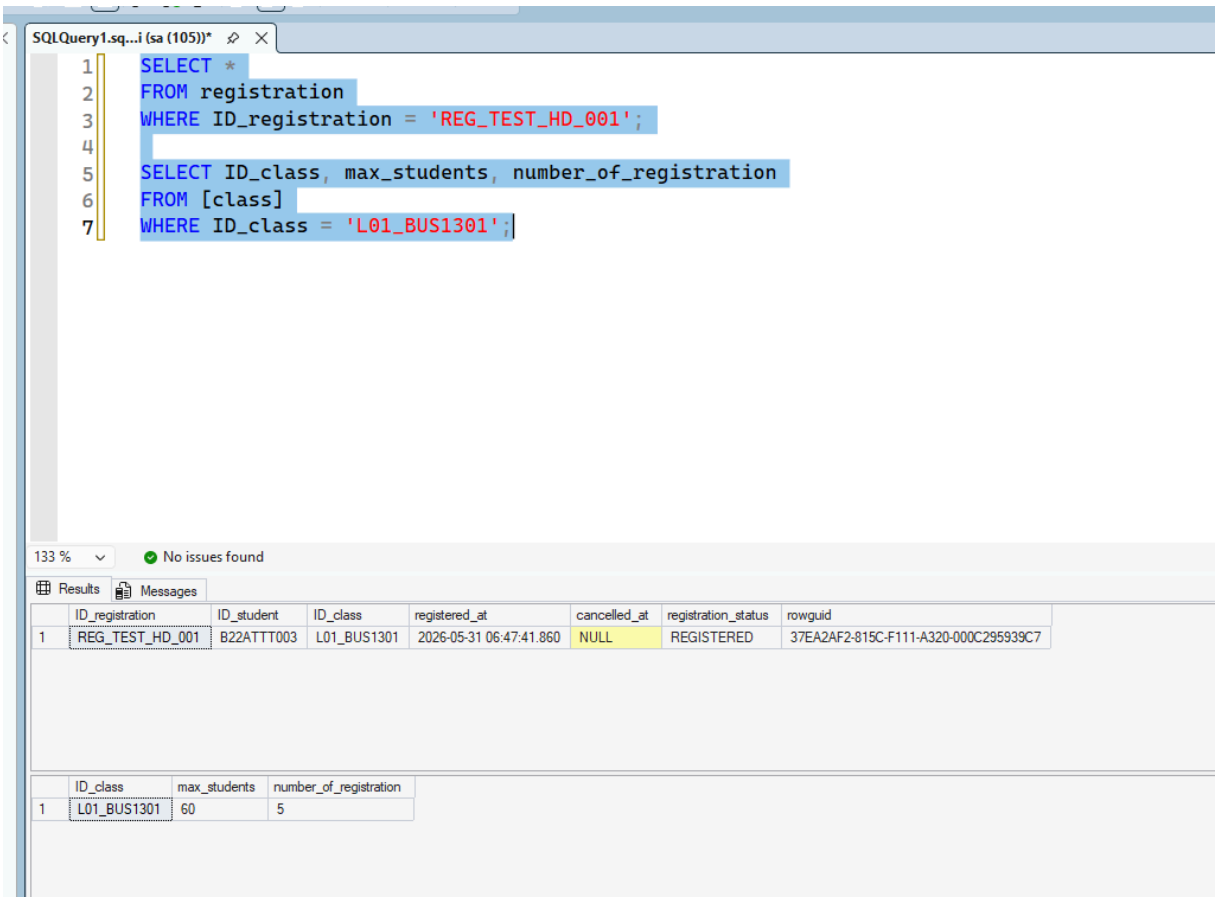
@ID_student = 'B22ATTT003',

@ID_class = 'L01_BUS1301',

@ID_headquarter = 'HQHD';

=> Đăng ký học phần thành công

Sau đó kiểm tra



=> Kiểm thử đăng ký thành công -> sĩ số đã đầy, sẽ không đăng ký được nữa

3. Procedure hủy đăng ký học phần

- Tên Thủ tục (Procedure Name): usp_CancelRegistration
- Tham số đầu vào (Input Parameters):
    - @ID_registration (Kiểu chuỗi): Mã giao dịch đăng ký cần hủy.

    - @ID_headquarter (Kiểu chuỗi, tùy chọn): Mã cơ sở đào tạo, dùng để đảm bảo tính phân mảnh và quyền thao tác trên dữ liệu cục bộ.

- Mục đích: Xử lý nghiệp vụ rút/hủy học phần của sinh viên. Thủ tục không xóa cứng (hard delete) dữ liệu mà thực hiện cập nhật trạng thái lịch sử, đồng thời hoàn trả lại an toàn số lượng chỗ trống (slot) cho lớp học phần. Toàn bộ quy trình được kiểm soát chặt chẽ bằng Transaction và cơ chế Locking nhằm tránh sai lệch dữ liệu trong môi trường phân tán đa luồng.

Mô tả giải thuật và logic xử lý (Bảo vệ bằng Transaction): Để tuân thủ nguyên tắc ACID, luồng thực thi được đặt trong khối quản lý lỗi TRY...CATCH và bảo vệ bởi một Transaction thống nhất.

- Bước 1 (Xác thực giao dịch và Trích xuất thông tin):
    - *Logic xử lý:* Hệ thống tìm kiếm trong bảng registration dựa trên Mã giao dịch đầu vào, kết hợp (Join) với các bảng class, teacher, department để đối chiếu xem giao dịch này có đang ở trạng thái hợp lệ (REGISTERED) và thuộc đúng cơ sở đào tạo được cấp quyền hay không (@ID_headquarter). Trong quá trình đọc, hệ thống áp dụng ngay khóa UPDLOCK và ROWLOCK để giữ chỗ dữ liệu.
    - *Xử lý lỗi:* Nếu không tìm thấy bản ghi thỏa mãn, hệ thống ném ra lỗi 50101: *"Không tìm thấy đăng ký hợp lệ để hủy"* và lập tức ngắt tiến trình. Nếu hợp lệ, hệ thống trích xuất và lưu giữ lại Mã lớp học phần (@ID_class) để phục vụ cho các bước sau.
- Bước 2 (Ghi nhận lịch sử hủy đăng ký):
    - Hệ thống tiến hành cập nhật trực tiếp trên bảng registration tại dòng dữ liệu của giao dịch vừa tìm thấy.
    - Trạng thái đăng ký được chuyển từ 'REGISTERED' (Đã đăng ký) sang 'CANCELLED' (Đã hủy), đồng thời hệ thống tự động ghi nhận thời gian thực thi hủy vào trường cancelled_at để phục vụ công tác truy vết (audit) sau này.
- Bước 3 (Hoàn trả sĩ số & Kiểm soát đồng thời - Bước mấu chốt):
    - *Logic xử lý:* Trái ngược với nghiệp vụ đăng ký, hệ thống thực hiện lệnh cập nhật giảm (UPDATE ... SET number_of_registration = number_of_registration - 1) trên bảng class đối với lớp học phần @ID_class.
    - *Cơ chế khóa (Locking):* Lệnh cập nhật này tiếp tục được đính kèm từ khóa WITH (UPDLOCK, ROWLOCK). Cơ chế này đặc biệt quan trọng để ngăn chặn lỗi Lost Update khi có sự kiện hủy đăng ký và sự kiện đăng

ký mới diễn ra tại cùng một thời điểm (miligiây). Hệ thống sẽ ép các giao dịch phải xếp hàng xử lý tuần tự trên dòng dữ liệu của lớp đó.

  - Thuật toán cũng đi kèm điều kiện an toàn: sĩ số hiện tại phải lớn hơn 0 (number_of_registration > 0) để tránh trường hợp sĩ số bị âm do lỗi logic.

- Bước 4 (Hoàn tất giao dịch):
  - Nếu toàn bộ 3 bước trên thực thi trơn tru, hệ thống gọi lệnh COMMIT để lưu lại thay đổi vĩnh viễn và giải phóng mọi khóa (Lock) trên hệ thống.
  - Trường hợp có bất kỳ lỗi nào xảy ra trong chuỗi tiến trình, khối CATCH sẽ bắt lỗi và gọi ROLLBACK để đưa cơ sở dữ liệu về lại trạng thái ban đầu, không có bản ghi nào bị cập nhật sai lệch.

Kiểm thử Procedure

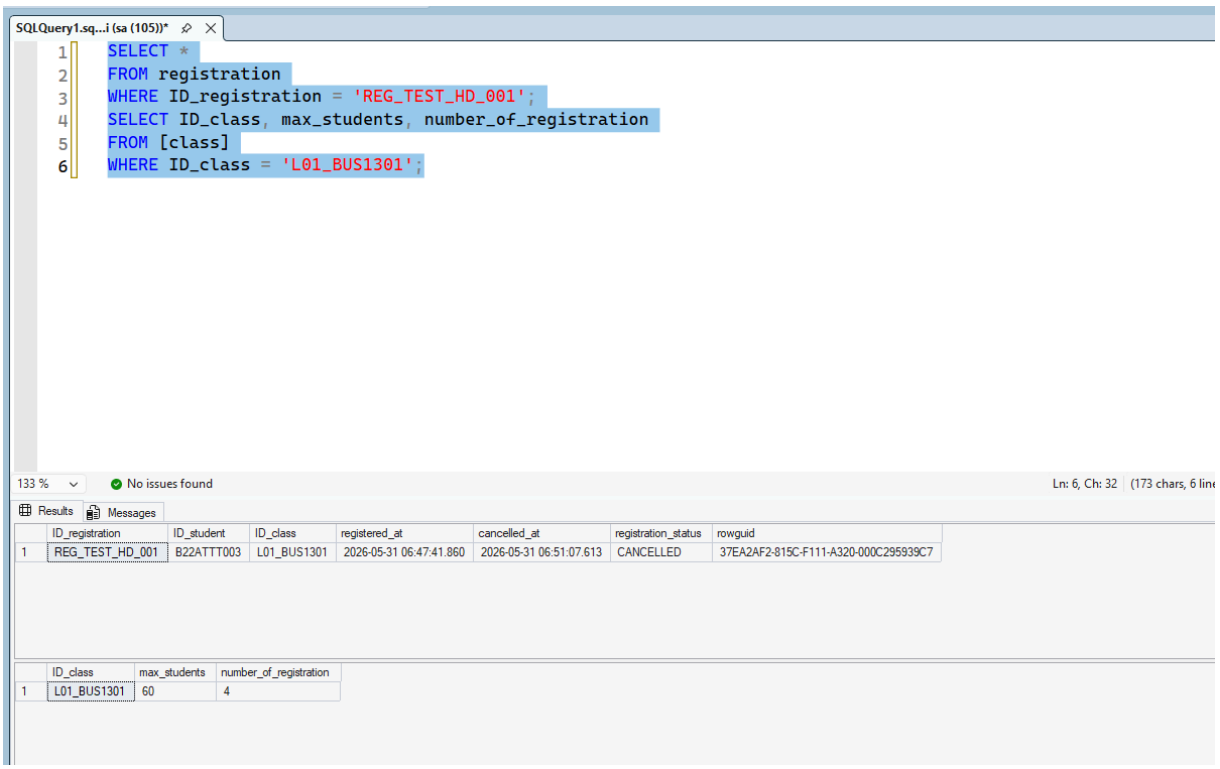
Chạy trước lệnh
EXEC usp_CancelRegistration

@ID_registration = 'REG_TEST_HD_001',

@ID_headquarter = 'HQHD';

để hủy đăng ký của bản ghi vừa thành công vừa trên luôn

Kiểm tra lại



Nhận xét: Thủ tục hoạt động chính xác. Trạng thái giao dịch chuyển sang 'CANCELLED', thời gian hủy được ghi nhận và sĩ số của lớp học phần giảm xuống đúng 1 đơn vị.

4. Procedure tra cứu kết quả đăng ký học phần

- Tên Thủ tục (Procedure Name): usp_GetRegistrationResult
- Tham số đầu vào (Input Parameters):
    - @ID_student (Kiểu chuỗi, mặc định NULL): Mã định danh của sinh viên cần tra cứu.
    - @ID_headquarter (Kiểu chuỗi, mặc định NULL): Mã cơ sở/trụ sở đào tạo dùng để giới hạn phạm vi tra cứu.
- Mục đích: Cung cấp một công cụ truy vấn động (Dynamic Query) để trích xuất toàn cảnh lịch sử và kết quả đăng ký tín chỉ. Thủ tục này được thiết kế linh hoạt để phục vụ đa đối tượng: từ việc một cá nhân sinh viên xem thời khóa biểu của riêng mình, cho đến việc ban quản lý đào tạo xuất báo cáo tổng thể toàn trường hoặc theo từng cơ sở phân mảnh.

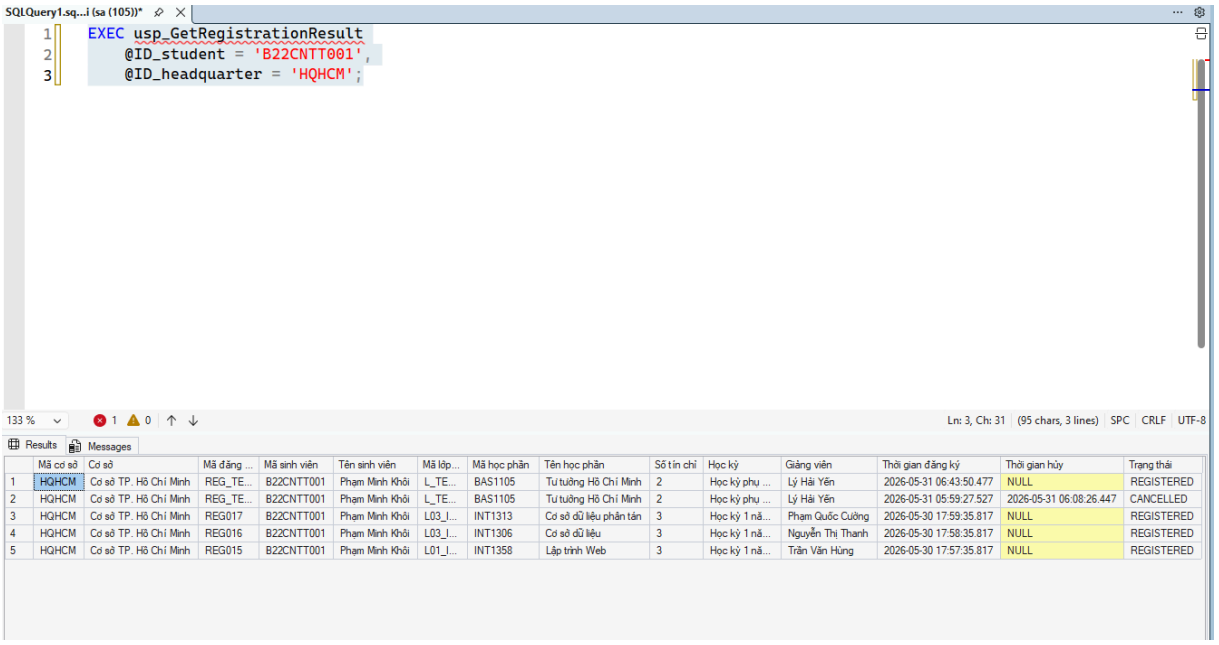
Mô tả giải thuật và logic xử lý: Vì đây là một thủ tục dạng Đọc (Read-only), thuật toán tập trung vào kỹ thuật gom nhóm dữ liệu và xử lý bộ lọc động trong môi trường phân tán:

- Bước 1 (Gom cụm và liên kết dữ liệu - Data Aggregation): * Bắt đầu từ bảng trung tâm là lịch sử giao dịch (registration), hệ thống thực hiện một chuỗi các phép nối (JOIN) diện rộng tới các bảng danh mục liên quan.
    - Cụ thể: Nối với student, department và headquarter để lấy thông tin nhân thân và vị trí cơ sở của sinh viên; Nối với class, subject, term và teacher để giải mã các thông số chi tiết của lớp học phần (Tên môn học, số tín chỉ, học kỳ, và giảng viên phụ trách).
- Bước 2 (Đồng bộ bảng mã ký tự phân tán): * Một đặc thù rất quan trọng của thuật toán này là việc áp dụng chỉ thị đồng bộ tập ký tự (COLLATE DATABASE_DEFAULT) tại tất cả các điều kiện nối bảng. Kỹ thuật này giúp triệt tiêu hoàn toàn lỗi xung đột Collation khi thực thi thủ tục truy vấn chéo giữa các máy chủ (Server) có cấu hình ngôn ngữ khác nhau trong hệ thống phân tán.
- Bước 3 (Xử lý bộ lọc động - Dynamic Filtering):

- Thuật toán kiểm tra các tham số đầu vào để đưa ra ngữ cảnh truy vấn phù hợp thông qua mệnh đề logic OR:
    - *Bộ lọc Sinh viên:* Nếu @ID_student được truyền vào, hệ thống chỉ xuất kết quả của riêng sinh viên đó. Nếu bỏ trống (NULL), hệ thống hiểu là yêu cầu lấy toàn bộ sinh viên.
    - *Bộ lọc Cơ sở:* Nếu @ID_headquarter được truyền vào, hệ thống sẽ giới hạn kết quả chỉ thuộc cơ sở đó (Ví dụ: Chỉ xem dữ liệu tại cơ sở TP.HCM). Nếu bỏ trống (NULL), hệ thống quét dữ liệu trên toàn cục.

- Bước 4 (Sắp xếp và Kết xuất):
    - Dữ liệu cuối cùng được trình bày dưới dạng một bảng thống kê trực quan.
    - Các bản ghi được hệ thống sắp xếp có chủ đích (ORDER BY) theo thứ tự ưu tiên: Phân cụm theo Cơ sở -> Phân nhóm theo Sinh viên -> Sắp xếp theo Thời gian đăng ký mới nhất giảm dần.

Kiểm thử Procedure



	Mã cơ sở	Cơ sở	Mã đăng ...	Mã sinh viên	Tên sinh viên	Mã lớp...	Mã học phần	Tên học phần	Số tín chỉ	Học kỳ	Giảng viên	Thời gian đăng ký	Thời gian hủy	Trạng thái
1	HQHCM	Cơ sở TP. Hồ Chí Minh	REG_TE...	B22CNTT001	Phạm Minh Khôi	L_TE...	BAS1105	Tư tưởng Hồ Chí Minh	2	Học kỳ phụ ...	Lý Hải Yến	2026-05-31 06:43:50.477	NULL	REGISTERED
2	HQHCM	Cơ sở TP. Hồ Chí Minh	REG_TE...	B22CNTT001	Phạm Minh Khôi	L_TE...	BAS1105	Tư tưởng Hồ Chí Minh	2	Học kỳ phụ ...	Lý Hải Yến	2026-05-31 05:59:27.527	2026-05-31 06:08:26.447	CANCELLED
3	HQHCM	Cơ sở TP. Hồ Chí Minh	REG017	B22CNTT001	Phạm Minh Khôi	L03_I...	INT1313	Cơ sở dữ liệu phân tán	3	Học kỳ 1 nă...	Phạm Quốc Cường	2026-05-30 17:59:35.817	NULL	REGISTERED
4	HQHCM	Cơ sở TP. Hồ Chí Minh	REG016	B22CNTT001	Phạm Minh Khôi	L03_I...	INT1306	Cơ sở dữ liệu	3	Học kỳ 1 nă...	Nguyễn Thị Thanh	2026-05-30 17:58:35.817	NULL	REGISTERED
5	HQHCM	Cơ sở TP. Hồ Chí Minh	REG015	B22CNTT001	Phạm Minh Khôi	L01_I...	INT1358	Lập trình Web	3	Học kỳ 1 nă...	Trần Văn Hùng	2026-05-30 17:57:35.817	NULL	REGISTERED

Nhận xét: Procedure hoạt động trơn tru và linh hoạt. Việc thiết kế tham số mặc định (NULL) giúp tối ưu hóa mã nguồn, chỉ cần viết một Procedure duy nhất nhưng đáp ứng được nhiều kịch bản sử dụng (Use-case) khác nhau của hệ thống.

5. Procedure xem thời khóa biểu sinh viên

- Tên Thủ tục (Procedure Name): usp_GetStudentTimetable

- Tham số đầu vào (Input Parameters):
    - @ID_student (Kiểu chuỗi, mặc định NULL): Mã định danh của sinh viên cần xem thời khóa biểu.
    - @ID_headquarter (Kiểu chuỗi, mặc định NULL): Mã cơ sở/trụ sở đào tạo dùng để giới hạn phạm vi truy xuất dữ liệu.
- Mục đích: Chức năng này đóng vai trò trích xuất và xây dựng một lịch trình học tập (thời khóa biểu) chi tiết và trực quan cho sinh viên. Dữ liệu đầu ra được tổng hợp dựa trên những lớp học phần mà sinh viên đã giữ chỗ và đăng ký thành công. Thủ tục có tính linh hoạt cao, hỗ trợ cả việc tra cứu lịch cá nhân lẫn việc xuất báo cáo thời khóa biểu tổng thể của toàn bộ sinh viên tại một cơ sở hoặc toàn trường (nếu chạy tại Server trung tâm).

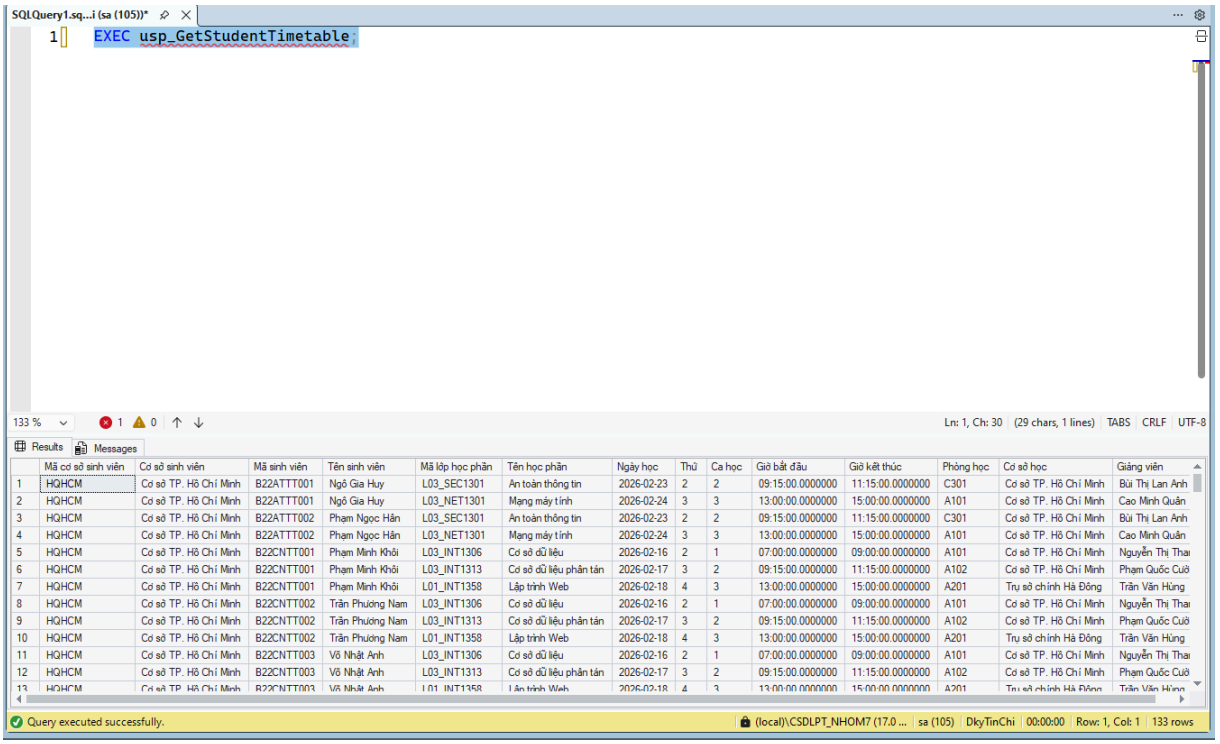
Mô tả giải thuật và logic xử lý: Quá trình tổng hợp thời khóa biểu yêu cầu liên kết một mạng lưới dữ liệu khá phức tạp, được thuật toán xử lý qua các bước sau:

- Bước 1 (Xác định mảng dữ liệu hợp lệ): * Thuật toán lấy bảng lịch sử giao dịch (registration) làm gốc. Bộ lọc đầu tiên và bắt buộc phải đi qua là điều kiện trạng thái đăng ký: chỉ lấy những bản ghi có registration_status = 'REGISTERED' (đã đăng ký thành công), tự động bỏ qua các môn đã hủy.
- Bước 2 (Gom cụm và liên kết dữ liệu đa chiều - Complex JOINs): * Từ mảng dữ liệu hợp lệ, hệ thống thực hiện các phép nối (JOIN) mở rộng ra 3 nhóm thông tin chính: * *Nhóm thông tin nhân sự:* Nối với bảng student, department, teacher và headquarter (của sinh viên) để định danh người học và người dạy. * *Nhóm thông tin môn học:* Nối với bảng class và subject để lấy mã lớp và tên môn học cụ thể. * *Nhóm không gian - thời gian (Cốt lõi của thời khóa biểu):* Nối với bảng lịch học (session), bảng khung giờ/ca học (timeslot), và bảng phòng học (room) kèm theo cơ sở của phòng học (headquarter). Phép nối này giúp giải mã chính xác Ngày học, Thứ trong tuần, Ca học, Giờ bắt đầu, Giờ kết thúc và Vị trí phòng học.
- Bước 3 (Đồng bộ hóa bảng mã phân tán): * Toàn bộ các phép nối (JOIN) và phép so sánh (WHERE) trong thủ tục đều được ép kiểu với chỉ thị COLLATE DATABASE_DEFAULT. Kỹ thuật này là bắt buộc trong môi trường cơ sở dữ liệu phân tán để đảm bảo không xảy ra hiện tượng xung đột đối chiếu ký tự (Collation conflict) khi máy chủ và các máy trạm có thiết lập ngôn ngữ hoặc bảng mã khác nhau.
- Bước 4 (Xử lý bộ lọc động):
    - Tương tự như thủ tục tra cứu kết quả, thuật toán sử dụng cấu trúc logic OR để xử lý tham số NULL:

- Nếu truyền @ID_student, hệ thống chỉ lọc ra thời khóa biểu của sinh viên đó.
- Nếu truyền @ID_headquarter, hệ thống chỉ lấy thời khóa biểu của sinh viên trực thuộc cơ sở này.
- Nếu cả 2 tham số đều NULL (ví dụ: khi admin chạy lệnh trên máy chủ Hà Đông), thủ tục sẽ quét và trả về toàn bộ dữ liệu thời khóa biểu hiện có trên hệ thống.

- Bước 5 (Sắp xếp thời khóa biểu):
  - Dữ liệu cuối cùng được trình bày dạng bảng và sắp xếp (ORDER BY) theo tính logic của không gian và thời gian thực tế: Cụm theo Cơ sở -> Nhóm theo Sinh viên -> Trình tự Ngày học tăng dần -> Trình tự Giờ bắt đầu tăng dần.

Kiểm thử Procedure



	Mã cơ sở sinh viên	Cơ sở sinh viên	Mã sinh viên	Tên sinh viên	Mã lớp học phần	Tên học phần	Ngày học	Thứ	Ca học	Giờ bắt đầu	Giờ kết thúc	Phòng học	Cơ sở học	Giảng viên
1	HQHCM	Cơ sở TP. Hồ Chí Minh	B22ATTT001	Ngô Gia Huy	L03_SEC1301	An toàn thông tin	2026-02-23	2	2	09:15:00.0000000	11:15:00.0000000	C301	Cơ sở TP. Hồ Chí Minh	Bùi Thị Lan Anh
2	HQHCM	Cơ sở TP. Hồ Chí Minh	B22ATTT001	Ngô Gia Huy	L03_NET1301	Mạng máy tính	2026-02-24	3	3	13:00:00.0000000	15:00:00.0000000	A101	Cơ sở TP. Hồ Chí Minh	Cao Minh Quân
3	HQHCM	Cơ sở TP. Hồ Chí Minh	B22ATTT002	Phạm Ngọc Hân	L03_SEC1301	An toàn thông tin	2026-02-23	2	2	09:15:00.0000000	11:15:00.0000000	C301	Cơ sở TP. Hồ Chí Minh	Bùi Thị Lan Anh
4	HQHCM	Cơ sở TP. Hồ Chí Minh	B22ATTT002	Phạm Ngọc Hân	L03_NET1301	Mạng máy tính	2026-02-24	3	3	13:00:00.0000000	15:00:00.0000000	A101	Cơ sở TP. Hồ Chí Minh	Cao Minh Quân
5	HQHCM	Cơ sở TP. Hồ Chí Minh	B22CNTT001	Phạm Minh Khôi	L03_INT1306	Cơ sở dữ liệu	2026-02-16	2	1	07:00:00.0000000	09:00:00.0000000	A101	Cơ sở TP. Hồ Chí Minh	Nguyễn Thị Thai
6	HQHCM	Cơ sở TP. Hồ Chí Minh	B22CNTT001	Phạm Minh Khôi	L03_INT1313	Cơ sở dữ liệu phân tán	2026-02-17	3	2	09:15:00.0000000	11:15:00.0000000	A102	Cơ sở TP. Hồ Chí Minh	Phạm Quốc Cuở
7	HQHCM	Cơ sở TP. Hồ Chí Minh	B22CNTT001	Phạm Minh Khôi	L01_INT1358	Lập trình Web	2026-02-18	4	3	13:00:00.0000000	15:00:00.0000000	A201	Trụ sở chính Hà Đông	Trần Văn Hùng
8	HQHCM	Cơ sở TP. Hồ Chí Minh	B22CNTT002	Trần Phương Nam	L03_INT1306	Cơ sở dữ liệu	2026-02-16	2	1	07:00:00.0000000	09:00:00.0000000	A101	Cơ sở TP. Hồ Chí Minh	Nguyễn Thị Thai
9	HQHCM	Cơ sở TP. Hồ Chí Minh	B22CNTT002	Trần Phương Nam	L03_INT1313	Cơ sở dữ liệu phân tán	2026-02-17	3	2	09:15:00.0000000	11:15:00.0000000	A102	Cơ sở TP. Hồ Chí Minh	Phạm Quốc Cuở
10	HQHCM	Cơ sở TP. Hồ Chí Minh	B22CNTT002	Trần Phương Nam	L01_INT1358	Lập trình Web	2026-02-18	4	3	13:00:00.0000000	15:00:00.0000000	A201	Trụ sở chính Hà Đông	Trần Văn Hùng
11	HQHCM	Cơ sở TP. Hồ Chí Minh	B22CNTT003	Võ Nhật Anh	L03_INT1306	Cơ sở dữ liệu	2026-02-16	2	1	07:00:00.0000000	09:00:00.0000000	A101	Cơ sở TP. Hồ Chí Minh	Nguyễn Thị Thai
12	HQHCM	Cơ sở TP. Hồ Chí Minh	B22CNTT003	Võ Nhật Anh	L03_INT1313	Cơ sở dữ liệu phân tán	2026-02-17	3	2	09:15:00.0000000	11:15:00.0000000	A102	Cơ sở TP. Hồ Chí Minh	Phạm Quốc Cuở
13	HQHCM	Cơ sở TP. Hồ Chí Minh	B22CNTT003	Võ Nhật Anh	L01_INT1358	Lập trình Web	2026-02-18	4	3	13:00:00.0000000	15:00:00.0000000	A201	Trụ sở chính Hà Đông	Trần Văn Hùng

SQLQuery1.sq...i (sa (105))*

```sql
EXEC usp_GetStudentTimetable
    @ID_headquarter = 'HQHD';
```

133 % | 1 | 0 | Ln: 2, Ch: 30 | (59 chars, 2 lines) | SPC | CRLF | UTF-8

Results | Messages

	Mã cơ sở sinh viên	Cơ sở sinh viên	Mã sinh viên	Tên sinh viên	Mã lớp học phần	Tên học phần	Ngày học	Thứ	Ca học	Giờ bắt đầu	Giờ kết thúc	Phòng học	Cơ sở học	Giảng viên
1	HQHD	Trụ sở chính Hà Đông	B22ATTT003	Phạm Thu Trang	L01_SEC1301	An toàn thông tin	2026-02-23	2	2	09:15:00.0000000	11:15:00.0000000	C301	Trụ sở chính Hà Đông	Phạm Minh Đức
2	HQHD	Trụ sở chính Hà Đông	B22ATTT003	Phạm Thu Trang	L01_NET1301	Mạng máy tính	2026-02-24	3	3	13:00:00.0000000	15:00:00.0000000	A101	Trụ sở chính Hà Đông	Hoàng Thu Trang
3	HQHD	Trụ sở chính Hà Đông	B22ATTT003	Phạm Thu Trang	L02_SEC1401	Mật mã học	2026-02-25	4	4	15:15:00.0000000	17:15:00.0000000	A102	Cơ sở Hòa Lạc	Trần Cẩm Tú
4	HQHD	Trụ sở chính Hà Đông	B22ATTT004	Võ Văn Kiệt	L01_SEC1301	An toàn thông tin	2026-02-23	2	2	09:15:00.0000000	11:15:00.0000000	C301	Trụ sở chính Hà Đông	Phạm Minh Đức
5	HQHD	Trụ sở chính Hà Đông	B22ATTT004	Võ Văn Kiệt	L01_NET1301	Mạng máy tính	2026-02-24	3	3	13:00:00.0000000	15:00:00.0000000	A101	Trụ sở chính Hà Đông	Hoàng Thu Trang
6	HQHD	Trụ sở chính Hà Đông	B22ATTT004	Võ Văn Kiệt	L02_SEC1401	Mật mã học	2026-02-25	4	4	15:15:00.0000000	17:15:00.0000000	A102	Cơ sở Hòa Lạc	Trần Cẩm Tú
7	HQHD	Trụ sở chính Hà Đông	B22CNTT004	Hoàng Văn Dũng	L01_INT1306	Cơ sở dữ liệu	2026-02-16	2	1	07:00:00.0000000	09:00:00.0000000	A101	Trụ sở chính Hà Đông	Nguyễn Văn Thái
8	HQHD	Trụ sở chính Hà Đông	B22CNTT004	Hoàng Văn Dũng	L01_INT1313	Cơ sở dữ liệu phân tán	2026-02-17	3	2	09:15:00.0000000	11:15:00.0000000	A102	Trụ sở chính Hà Đông	Lê Thị Mai
9	HQHD	Trụ sở chính Hà Đông	B22CNTT004	Hoàng Văn Dũng	L03_INT1358	Lập trình Web	2026-02-18	4	3	13:00:00.0000000	15:00:00.0000000	A201	Cơ sở TP. Hồ Chí Minh	Đinh Tuấn Kiệt
10	HQHD	Trụ sở chính Hà Đông	B22CNTT005	Lê Minh Tâm	L01_INT1306	Cơ sở dữ liệu	2026-02-16	2	1	07:00:00.0000000	09:00:00.0000000	A101	Trụ sở chính Hà Đông	Nguyễn Văn Thái
11	HQHD	Trụ sở chính Hà Đông	B22CNTT005	Lê Minh Tâm	L01_INT1313	Cơ sở dữ liệu phân tán	2026-02-17	3	2	09:15:00.0000000	11:15:00.0000000	A102	Trụ sở chính Hà Đông	Lê Thị Mai
12	HQHD	Trụ sở chính Hà Đông	B22CNTT005	Lê Minh Tâm	L03_INT1358	Lập trình Web	2026-02-18	4	3	13:00:00.0000000	15:00:00.0000000	A201	Cơ sở TP. Hồ Chí Minh	Đinh Tuấn Kiệt
13	HQHD	Trụ sở chính Hà Đông	B22CNTT006	Nguyễn Thị Lan	L01_INT1306	Cơ sở dữ liệu	2026-02-16	2	1	07:00:00.0000000	09:00:00.0000000	A101	Trụ sở chính Hà Đông	Nguyễn Văn Thái

SQLQuery1.sq...i (sa (105))*

```sql
EXEC usp_GetStudentTimetable
    @ID_student = 'B22CNTT001';
```

133 % | 1 | 0 | Ln: 2, Ch: 32 | (61 chars, 2 lines) | SPC | CRLF | UTF-8

Results | Messages

	Mã cơ sở sinh viên	Cơ sở sinh viên	Mã sinh viên	Tên sinh viên	Mã lớp học phần	Tên học phần	Ngày học	Thứ	Ca học	Giờ bắt đầu	Giờ kết thúc	Phòng học	Cơ sở học	Giảng viên
1	HQHCM	Cơ sở TP. Hồ Chí Minh	B22CNTT001	Phạm Minh Khôi	L03_INT1306	Cơ sở dữ liệu	2026-02-16	2	1	07:00:00.0000000	09:00:00.0000000	A101	Cơ sở TP. Hồ Chí Minh	Nguyễn Thị Thanh
2	HQHCM	Cơ sở TP. Hồ Chí Minh	B22CNTT001	Phạm Minh Khôi	L03_INT1313	Cơ sở dữ liệu phân tán	2026-02-17	3	2	09:15:00.0000000	11:15:00.0000000	A102	Cơ sở TP. Hồ Chí Minh	Phạm Quốc Cường
3	HQHCM	Cơ sở TP. Hồ Chí Minh	B22CNTT001	Phạm Minh Khôi	L01_INT1358	Lập trình Web	2026-02-18	4	3	13:00:00.0000000	15:00:00.0000000	A201	Trụ sở chính Hà Đông	Trần Văn Hùng

SQLQuery1.sq...i (sa (105))*

```sql
EXEC usp_GetStudentTimetable
    @ID_student = 'B22ATTT003',
    @ID_headquarter = 'HQHD';
```

133 % | 1 | 0 | Ln: 3, Ch: 30 | (92 chars, 3 lines) | SPC | CRLF | UTF-8

Results | Messages

	Mã cơ sở sinh viên	Cơ sở sinh viên	Mã sinh viên	Tên sinh viên	Mã lớp học phần	Tên học phần	Ngày học	Thứ	Ca học	Giờ bắt đầu	Giờ kết thúc	Phòng học	Cơ sở học	Giảng viên
1	HQHD	Trụ sở chính Hà Đông	B22ATTT003	Phạm Thu Trang	L01_SEC1301	An toàn thông tin	2026-02-23	2	2	09:15:00.0000000	11:15:00.0000000	C301	Trụ sở chính Hà Đông	Phạm Minh Đức
2	HQHD	Trụ sở chính Hà Đông	B22ATTT003	Phạm Thu Trang	L01_NET1301	Mạng máy tính	2026-02-24	3	3	13:00:00.0000000	15:00:00.0000000	A101	Trụ sở chính Hà Đông	Hoàng Thu Trang
3	HQHD	Trụ sở chính Hà Đông	B22ATTT003	Phạm Thu Trang	L02_SEC1401	Mật mã học	2026-02-25	4	4	15:15:00.0000000	17:15:00.0000000	A102	Cơ sở Hòa Lạc	Trần Cẩm Tú

6. Procedure thống kê tình trạng đăng ký lớp học phần theo học kỳ

- Tên Thủ tục (Procedure Name): usp_StatsClassRegistrationByTerm
- Tham số đầu vào (Input Parameters):
    - @ID_term (Kiểu chuỗi, bắt buộc): Mã học kỳ cần truy xuất dữ liệu thống kê.
    - @ID_headquarter (Kiểu chuỗi, tùy chọn, mặc định NULL): Mã cơ sở/trụ sở đào tạo dùng để giới hạn phạm vi thống kê theo vùng phân mảnh.
- Mục đích: Cung cấp báo cáo phân tích tổng quan cho Ban quản lý đào tạo về tình hình ghi danh tại các lớp học phần trong một học kỳ cụ thể. Báo cáo không chỉ liệt kê số liệu thô mà còn tính toán tỷ lệ lấp đầy (Fill Rate) của từng lớp, từ đó giúp nhà trường đánh giá độ "hot" của môn học, tối ưu hóa việc phân bổ giảng viên, hoặc ra quyết định đóng/mở thêm các lớp học phần mới.

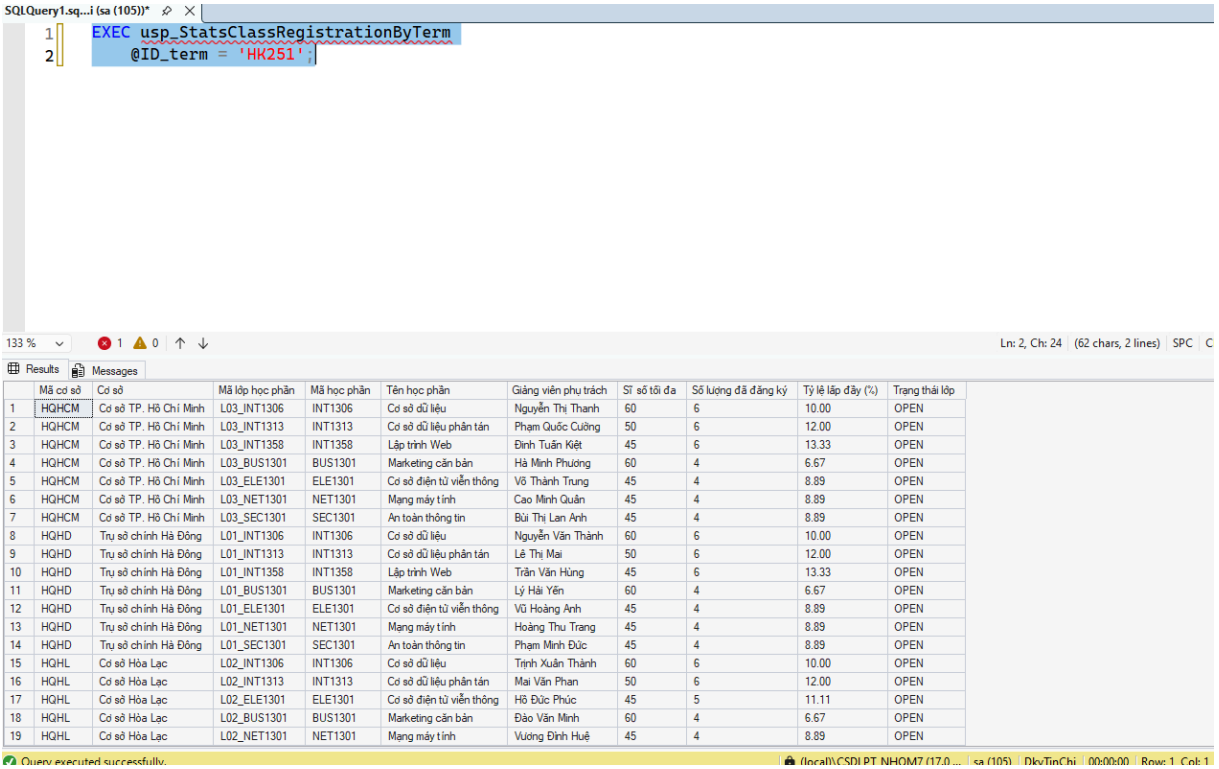
Mô tả giải thuật và logic xử lý: Là một thủ tục dạng báo cáo thống kê (Reporting), thuật toán tập trung vào việc gom nhóm dữ liệu, tính toán các chỉ số phái sinh và xử lý đồng bộ chuỗi trong môi trường CSDL phân tán.

- Bước 1 (Gom cụm và giải mã danh mục): * Xuất phát từ bảng class (lớp học phần) - nơi chứa các chỉ số gốc về sĩ số đăng ký và sĩ số tối đa. Thuật toán tiến hành các phép nối (JOIN) lần lượt tới các bảng danh mục: * Nối với bảng subject để giải mã Tên môn học. * Nối với bảng teacher, department, và headquarter để định danh Giảng viên phụ trách và Cơ sở phân bổ lớp học đó.
- Bước 2 (Kiểm soát đồng bộ bảng mã - Collation): * Giống như các truy vấn phân tán trước đó, toàn bộ các phép so sánh khóa chính/khóa ngoại trong quá trình JOIN đều được gắn chỉ thị COLLATE DATABASE_DEFAULT. Việc này đảm bảo báo cáo thống kê có thể chạy mượt mà trên Server trung tâm (nơi tổng hợp dữ liệu từ nhiều node) mà không gặp lỗi xung đột đối chiếu ký tự.
- Bước 3 (Xử lý bộ lọc tham số):
    - *Lọc theo học kỳ:* Hệ thống bắt buộc lọc các lớp học phần thuộc đúng học kỳ @ID_term được truyền vào.
    - *Lọc theo cơ sở:* Sử dụng logic OR để kiểm tra: Nếu @ID_headquarter được chỉ định, báo cáo chỉ thống kê các lớp thuộc cơ sở đó. Nếu không chỉ định (NULL), hệ thống tổng hợp báo cáo trên toàn mạng lưới các cơ sở.
- Bước 4 (Tính toán chỉ số phái sinh - Derived Metrics):

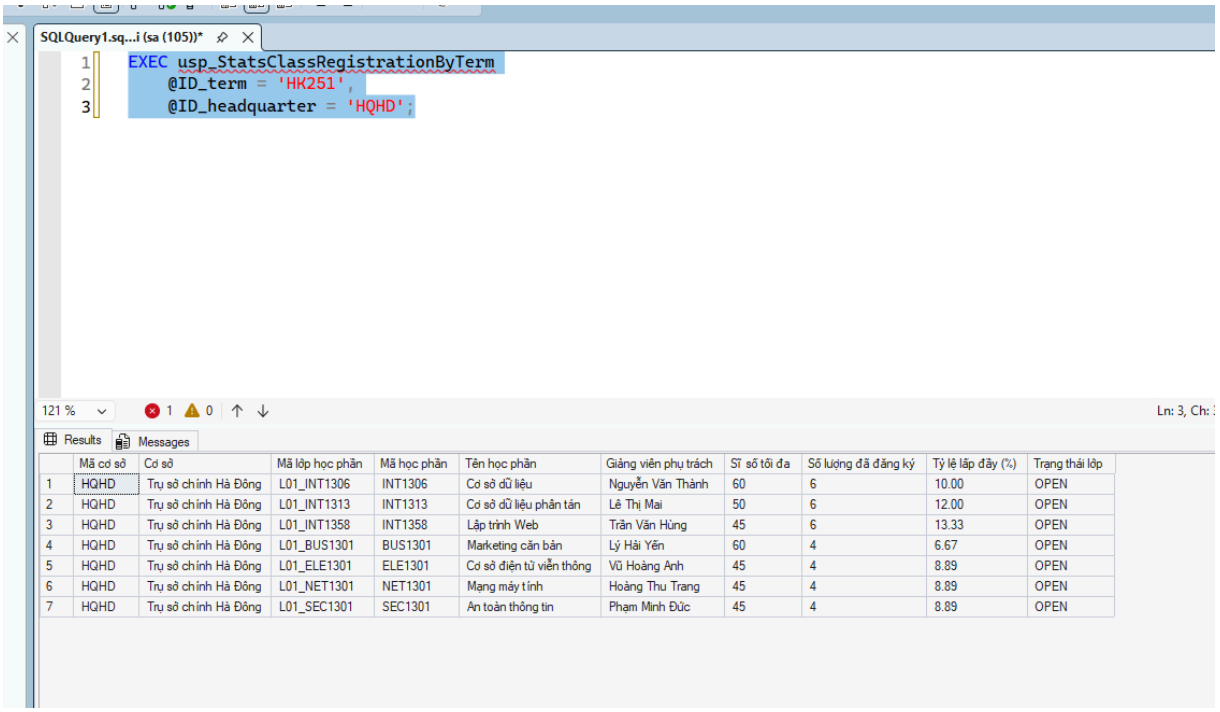
- *Logic tính Tỷ lệ lấp đầy:* Thuật toán sử dụng công thức (Số lượng đã đăng ký * 100.0) / Sĩ số tối đa.
- *Xử lý lỗi chia cho 0:* Để bảo vệ hệ thống khỏi lỗi toán học (Divide by zero) trong trường hợp có lớp học phần khai báo sĩ số tối đa bằng 0, thuật toán sử dụng hàm NULLIF(max_students, 0). Nếu mẫu số là 0, kết quả phép chia sẽ an toàn trả về NULL thay vì làm sập Procedure. Kết quả tính toán sau đó được ép kiểu (CAST) về dạng thập phân với 2 chữ số sau dấu phẩy để báo cáo gọn gàng hơn.

- Bước 5 (Sắp xếp và Kết xuất):
  - Dữ liệu được trả về dưới dạng bảng thống kê rõ ràng.
  - Các lớp học phần được sắp xếp (ORDER BY) theo thứ tự ưu tiên: Gom theo Cơ sở -> Ưu tiên các lớp có Số lượng đăng ký cao nhất xếp lên đầu (giảm dần) -> Nếu trùng số lượng thì sắp xếp theo Mã lớp.

Kiểm thử Procedure



	Mã cơ sở	Cơ sở	Mã lớp học phần	Mã học phần	Tên học phần	Giảng viên phụ trách	Sĩ số tối đa	Số lượng đã đăng ký	Tỷ lệ lấp đầy (%)	Trạng thái lớp
1	HQHCM	Cơ sở TP. Hồ Chí Minh	L03_INT1306	INT1306	Cơ sở dữ liệu	Nguyễn Thị Thanh	60	6	10.00	OPEN
2	HQHCM	Cơ sở TP. Hồ Chí Minh	L03_INT1313	INT1313	Cơ sở dữ liệu phân tán	Phạm Quốc Cường	50	6	12.00	OPEN
3	HQHCM	Cơ sở TP. Hồ Chí Minh	L03_INT1358	INT1358	Lập trình Web	Đinh Tuấn Kiệt	45	6	13.33	OPEN
4	HQHCM	Cơ sở TP. Hồ Chí Minh	L03_BUS1301	BUS1301	Marketing căn bản	Hà Minh Phương	60	4	6.67	OPEN
5	HQHCM	Cơ sở TP. Hồ Chí Minh	L03_ELE1301	ELE1301	Cơ sở điện tử viễn thông	Võ Thành Trung	45	4	8.89	OPEN
6	HQHCM	Cơ sở TP. Hồ Chí Minh	L03_NET1301	NET1301	Mạng máy tính	Cao Minh Quân	45	4	8.89	OPEN
7	HQHCM	Cơ sở TP. Hồ Chí Minh	L03_SEC1301	SEC1301	An toàn thông tin	Bùi Thị Lan Anh	45	4	8.89	OPEN
8	HQHD	Trụ sở chính Hà Đông	L01_INT1306	INT1306	Cơ sở dữ liệu	Nguyễn Văn Thành	60	6	10.00	OPEN
9	HQHD	Trụ sở chính Hà Đông	L01_INT1313	INT1313	Cơ sở dữ liệu phân tán	Lê Thị Mai	50	6	12.00	OPEN
10	HQHD	Trụ sở chính Hà Đông	L01_INT1358	INT1358	Lập trình Web	Trần Văn Hùng	45	6	13.33	OPEN
11	HQHD	Trụ sở chính Hà Đông	L01_BUS1301	BUS1301	Marketing căn bản	Lý Hải Yến	60	4	6.67	OPEN
12	HQHD	Trụ sở chính Hà Đông	L01_ELE1301	ELE1301	Cơ sở điện tử viễn thông	Vũ Hoàng Anh	45	4	8.89	OPEN
13	HQHD	Trụ sở chính Hà Đông	L01_NET1301	NET1301	Mạng máy tính	Hoàng Thu Trang	45	4	8.89	OPEN
14	HQHD	Trụ sở chính Hà Đông	L01_SEC1301	SEC1301	An toàn thông tin	Phạm Minh Đức	45	4	8.89	OPEN
15	HQHL	Cơ sở Hòa Lạc	L02_INT1306	INT1306	Cơ sở dữ liệu	Trịnh Xuân Thành	60	6	10.00	OPEN
16	HQHL	Cơ sở Hòa Lạc	L02_INT1313	INT1313	Cơ sở dữ liệu phân tán	Mai Văn Phan	50	6	12.00	OPEN
17	HQHL	Cơ sở Hòa Lạc	L02_ELE1301	ELE1301	Cơ sở điện tử viễn thông	Hồ Đức Phúc	45	5	11.11	OPEN
18	HQHL	Cơ sở Hòa Lạc	L02_BUS1301	BUS1301	Marketing căn bản	Đào Văn Minh	60	4	6.67	OPEN
19	HQHL	Cơ sở Hòa Lạc	L02_NET1301	NET1301	Mạng máy tính	Vương Đình Huệ	45	4	8.89	OPEN



```sql
EXEC usp_StatsClassRegistrationByTerm
    @ID_term = 'HK251',
    @ID_headquarter = 'HQHD';
```

Results | Messages

	Mã cơ sở	Cơ sở	Mã lớp học phần	Mã học phần	Tên học phần	Giảng viên phụ trách	Sĩ số tối đa	Số lượng đã đăng ký	Tỷ lệ lấp đầy (%)	Trạng thái lớp
1	HQHD	Trụ sở chính Hà Đông	L01_INT1306	INT1306	Cơ sở dữ liệu	Nguyễn Văn Thành	60	6	10.00	OPEN
2	HQHD	Trụ sở chính Hà Đông	L01_INT1313	INT1313	Cơ sở dữ liệu phân tán	Lê Thị Mai	50	6	12.00	OPEN
3	HQHD	Trụ sở chính Hà Đông	L01_INT1358	INT1358	Lập trình Web	Trần Văn Hùng	45	6	13.33	OPEN
4	HQHD	Trụ sở chính Hà Đông	L01_BUS1301	BUS1301	Marketing căn bản	Lý Hải Yến	60	4	6.67	OPEN
5	HQHD	Trụ sở chính Hà Đông	L01_ELE1301	ELE1301	Cơ sở điện tử viễn thông	Vũ Hoàng Anh	45	4	8.89	OPEN
6	HQHD	Trụ sở chính Hà Đông	L01_NET1301	NET1301	Mạng máy tính	Hoàng Thu Trang	45	4	8.89	OPEN
7	HQHD	Trụ sở chính Hà Đông	L01_SEC1301	SEC1301	An toàn thông tin	Phạm Minh Đức	45	4	8.89	OPEN

7. Procedure kiểm tra đăng ký chéo cơ sở

- Tên Thủ tục (Procedure Name): usp_CheckCrossCampusRegistration
- Tham số đầu vào (Input Parameters): Không có. (Thủ tục này được thiết kế để quét và báo cáo trên toàn bộ dữ liệu hiện có của hệ thống).
- Mục đích: Đóng vai trò là một công cụ kiểm toán (Audit Tool) để phát hiện và thống kê các trường hợp "đăng ký chéo" – tức là sinh viên thuộc sự quản lý của cơ sở này nhưng lại đăng ký vào lớp học phần do giảng viên của cơ sở khác phụ trách. Việc này rất quan trọng trong môi trường phân tán để theo dõi sự luân chuyển người học, tính toán chi phí đào tạo liên cơ sở, hoặc đơn giản là rà soát các bất thường trong dữ liệu.

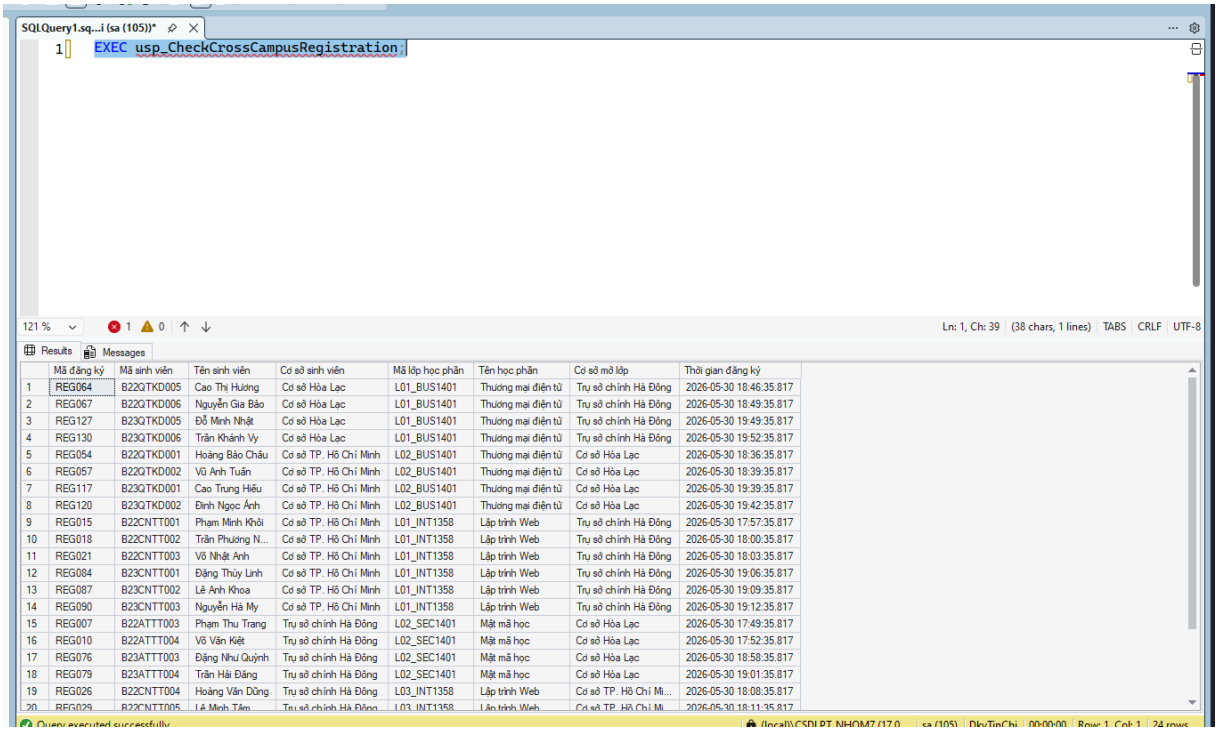
Mô tả giải thuật và logic xử lý: Bài toán tìm kiếm dữ liệu chéo cơ sở thực chất là việc so sánh hai nhánh dữ liệu độc lập xuất phát từ cùng một điểm giao dịch (registration). Thuật toán được xử lý qua các bước sau:

- Bước 1 (Xác định tập dữ liệu quét): Khởi tạo luồng truy vấn từ bảng lịch sử đăng ký (registration), và chỉ giữ lại những giao dịch đang có trạng thái hợp lệ (registration_status = 'REGISTERED').
- Bước 2 (Truy vết nguồn gốc Sinh viên): Hệ thống thực hiện một chuỗi các phép nối (JOIN) ngược từ bản ghi đăng ký về bảng student, sau đó qua department (Khoa) và cuối cùng đến headquarter (Cơ sở). Bước này nhằm xác định chính xác Cơ sở gốc mà sinh viên đó đang trực thuộc (gọi là nhánh hq_st).
- Bước 3 (Truy vết nguồn gốc Lớp học phần): Song song với Bước 2, hệ thống thực hiện một chuỗi phép nối khác từ bản ghi đăng ký sang bảng class (Lớp học

phần), rồi đến teacher (Giảng viên phụ trách), qua department của giảng viên và đến headquarter. Bước này nhằm xác định Cơ sở tổ chức lớp học (gọi là nhánh hq_cl).

- Bước 4 (Đồng bộ bảng mã phân tán): Toàn bộ các phép nối dữ liệu giữa các nhánh đều được áp dụng nghiêm ngặt chỉ thị COLLATE DATABASE_DEFAULT. Do tính chất truy vấn chéo giữa các phân mảnh (các chi nhánh khác nhau), kỹ thuật này đảm bảo không xảy ra lỗi xung đột bộ mã ký tự (Collation Conflict).
- Bước 5 (Sàng lọc logic chéo cơ sở - Cốt lõi thuật toán): Thuật toán tiến hành đối chiếu kết quả của Bước 2 và Bước 3 thông qua mệnh đề WHERE. Chỉ những bản ghi nào có Cơ sở gốc của sinh viên khác với Cơ sở tổ chức lớp học (hq_st <> hq_cl) mới được hệ thống giữ lại.
- Bước 6 (Sắp xếp và Kết xuất): Tập kết quả cuối cùng được xuất ra dưới dạng báo cáo danh sách. Dữ liệu được sắp xếp (ORDER BY) một cách có hệ thống: Phân cụm theo Cơ sở của sinh viên -> Phân nhóm tiếp theo Cơ sở mở lớp -> Trình tự Mã sinh viên.

Kiểm thử Procedure



	Mã đăng ký	Mã sinh viên	Tên sinh viên	Cơ sở sinh viên	Mã lớp học phần	Tên học phần	Cơ sở mở lớp	Thời gian đăng ký
1	REG064	B22QTKD005	Cao Thị Hương	Cơ sở Hòa Lạc	L01_BUS1401	Thương mại điện tử	Trụ sở chính Hà Đông	2026-05-30 18:46:35.817
2	REG067	B22QTKD006	Nguyễn Gia Bảo	Cơ sở Hòa Lạc	L01_BUS1401	Thương mại điện tử	Trụ sở chính Hà Đông	2026-05-30 18:49:35.817
3	REG127	B23QTKD005	Đỗ Minh Nhật	Cơ sở Hòa Lạc	L01_BUS1401	Thương mại điện tử	Trụ sở chính Hà Đông	2026-05-30 19:49:35.817
4	REG130	B23QTKD006	Trần Khánh Vy	Cơ sở Hòa Lạc	L01_BUS1401	Thương mại điện tử	Trụ sở chính Hà Đông	2026-05-30 19:52:35.817
5	REG054	B22QTKD001	Hoàng Bảo Châu	Cơ sở TP. Hồ Chí Minh	L02_BUS1401	Thương mại điện tử	Cơ sở Hòa Lạc	2026-05-30 18:36:35.817
6	REG057	B22QTKD002	Vũ Anh Tuấn	Cơ sở TP. Hồ Chí Minh	L02_BUS1401	Thương mại điện tử	Cơ sở Hòa Lạc	2026-05-30 18:39:35.817
7	REG117	B23QTKD001	Cao Trung Hiếu	Cơ sở TP. Hồ Chí Minh	L02_BUS1401	Thương mại điện tử	Cơ sở Hòa Lạc	2026-05-30 19:39:35.817
8	REG120	B23QTKD002	Đinh Ngọc Ánh	Cơ sở TP. Hồ Chí Minh	L02_BUS1401	Thương mại điện tử	Cơ sở Hòa Lạc	2026-05-30 19:42:35.817
9	REG015	B22CNTT001	Phạm Minh Khôi	Cơ sở TP. Hồ Chí Minh	L01_INT1358	Lập trình Web	Trụ sở chính Hà Đông	2026-05-30 17:57:35.817
10	REG018	B22CNTT002	Trần Phương N...	Cơ sở TP. Hồ Chí Minh	L01_INT1358	Lập trình Web	Trụ sở chính Hà Đông	2026-05-30 18:00:35.817
11	REG021	B22CNTT003	Võ Nhật Anh	Cơ sở TP. Hồ Chí Minh	L01_INT1358	Lập trình Web	Trụ sở chính Hà Đông	2026-05-30 18:03:35.817
12	REG084	B23CNTT001	Đặng Thùy Linh	Cơ sở TP. Hồ Chí Minh	L01_INT1358	Lập trình Web	Trụ sở chính Hà Đông	2026-05-30 19:06:35.817
13	REG087	B23CNTT002	Lê Anh Khoa	Cơ sở TP. Hồ Chí Minh	L01_INT1358	Lập trình Web	Trụ sở chính Hà Đông	2026-05-30 19:09:35.817
14	REG090	B23CNTT003	Nguyễn Hà My	Cơ sở TP. Hồ Chí Minh	L01_INT1358	Lập trình Web	Trụ sở chính Hà Đông	2026-05-30 19:12:35.817
15	REG007	B22ATTT003	Phạm Thu Trang	Trụ sở chính Hà Đông	L02_SEC1401	Mật mã học	Cơ sở Hòa Lạc	2026-05-30 17:49:35.817
16	REG010	B22ATTT004	Võ Văn Kiệt	Trụ sở chính Hà Đông	L02_SEC1401	Mật mã học	Cơ sở Hòa Lạc	2026-05-30 17:52:35.817
17	REG076	B23ATTT003	Đặng Như Quỳnh	Trụ sở chính Hà Đông	L02_SEC1401	Mật mã học	Cơ sở Hòa Lạc	2026-05-30 18:58:35.817
18	REG079	B23ATTT004	Trần Hải Đăng	Trụ sở chính Hà Đông	L02_SEC1401	Mật mã học	Cơ sở Hòa Lạc	2026-05-30 19:01:35.817
19	REG026	B22CNTT004	Hoàng Văn Dũng	Trụ sở chính Hà Đông	L03_INT1358	Lập trình Web	Cơ sở TP. Hồ Chí Mi...	2026-05-30 18:08:35.817
20	REG029	B22CNTT005	Lê Minh Tâm	Trụ sở chính Hà Đông	L03_INT1358	Lập trình Web	Cơ sở TP. Hồ Chí Mi...	2026-05-30 18:11:35.817

Nhận xét: Procedure hỗ trợ phát hiện và thống kê các trường hợp đăng ký chéo cơ sở.

8. Procedure thống kê sinh viên đăng ký theo cơ sở

- Tên Thủ tục (Procedure Name): usp_StatsRegisteredStudentsByCampus
- Tham số đầu vào (Input Parameters):

    - @ID_headquarter (Kiểu chuỗi, tùy chọn, mặc định NULL): Mã cơ sở/trụ sở đào tạo dùng để giới hạn phạm vi thống kê.

- Mục đích: Chức năng này cung cấp báo cáo tổng hợp về quy mô tham gia đăng ký học phần của người học trên toàn hệ thống. Báo cáo giúp Ban quản lý đào tạo nắm bắt được chính xác "số lượng đầu người" (số sinh viên thực tế) đã thực hiện giao dịch đăng ký thành công tại từng cơ sở, từ đó đánh giá được tiến độ và tỷ lệ tham gia trong đợt đăng ký tín chỉ.

Mô tả giải thuật và logic xử lý: Là một thủ tục mang tính chất báo cáo tổng hợp (Aggregation Report), thuật toán đặc biệt chú trọng vào kỹ thuật đếm chính xác và bảo toàn cấu trúc danh mục trong môi trường phân tán:

- Bước 1 (Xác định khung danh mục gốc): Khác với các truy vấn thông thường xuất phát từ bảng giao dịch, thuật toán này lấy bảng headquarter (Cơ sở) làm gốc. Tiếp đó, hệ thống thực hiện nối (JOIN) tuần tự qua bảng department (Khoa) và student (Sinh viên) để khoanh vùng chính xác tập hợp tất cả sinh viên thuộc từng cơ sở.

- Bước 2 (Liên kết dữ liệu giao dịch - Kỹ thuật LEFT JOIN): * Thuật toán thực hiện phép nối trái (LEFT JOIN) từ tập dữ liệu sinh viên ở Bước 1 sang bảng lịch sử giao dịch registration.

    - Kèm theo đó là điều kiện lọc khắt khe: chỉ lấy những giao dịch đã thành công (registration_status = 'REGISTERED').

    - *Ý nghĩa logic:* Việc sử dụng LEFT JOIN là một thiết kế rất tinh tế. Kỹ thuật này đảm bảo rằng ngay cả khi một cơ sở chưa có bất kỳ sinh viên nào đăng ký học phần, tên cơ sở đó vẫn được xuất hiện trên báo cáo tổng với số lượng là 0, thay vì bị ẩn đi hoặc biến mất hoàn toàn do thiếu dữ liệu.

- Bước 3 (Đồng bộ bảng mã phân tán): Để bảo vệ an toàn cho các truy vấn chéo site, tất cả các điều kiện nối bảng (ON) và điều kiện lọc (WHERE) đều được ép kiểu với chỉ thị COLLATE DATABASE_DEFAULT. Kỹ thuật này triệt tiêu hoàn toàn rủi ro xung đột đối chiếu ký tự (Collation conflict) giữa các phân mảnh.

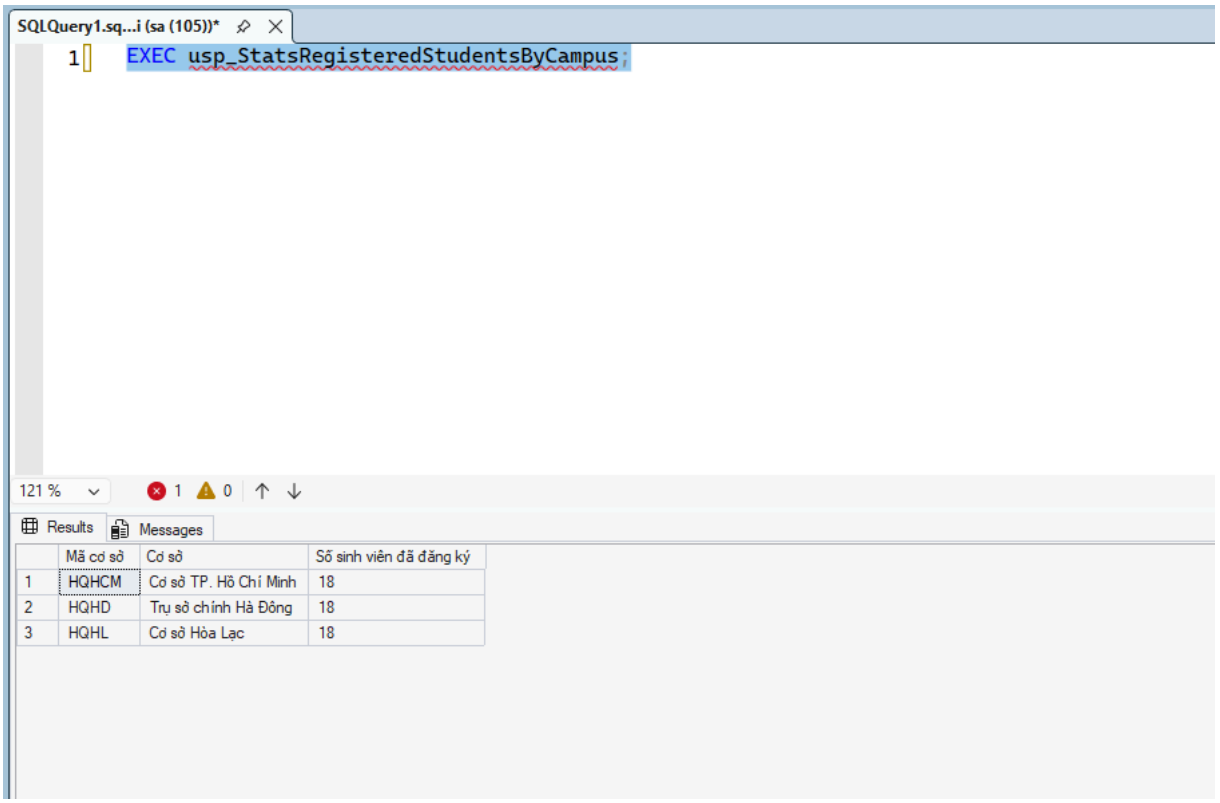
- Bước 4 (Xử lý bộ lọc tham số động): Thuật toán sử dụng cấu trúc logic OR để kiểm tra: Nếu tham số @ID_headquarter mang giá trị NULL, hệ thống sẽ quét và thống kê toàn bộ các cơ sở hiện có. Nếu truyền vào một mã cụ thể (ví dụ: 'HCM'), hệ thống chỉ tập trung đếm dữ liệu của riêng cơ sở đó.

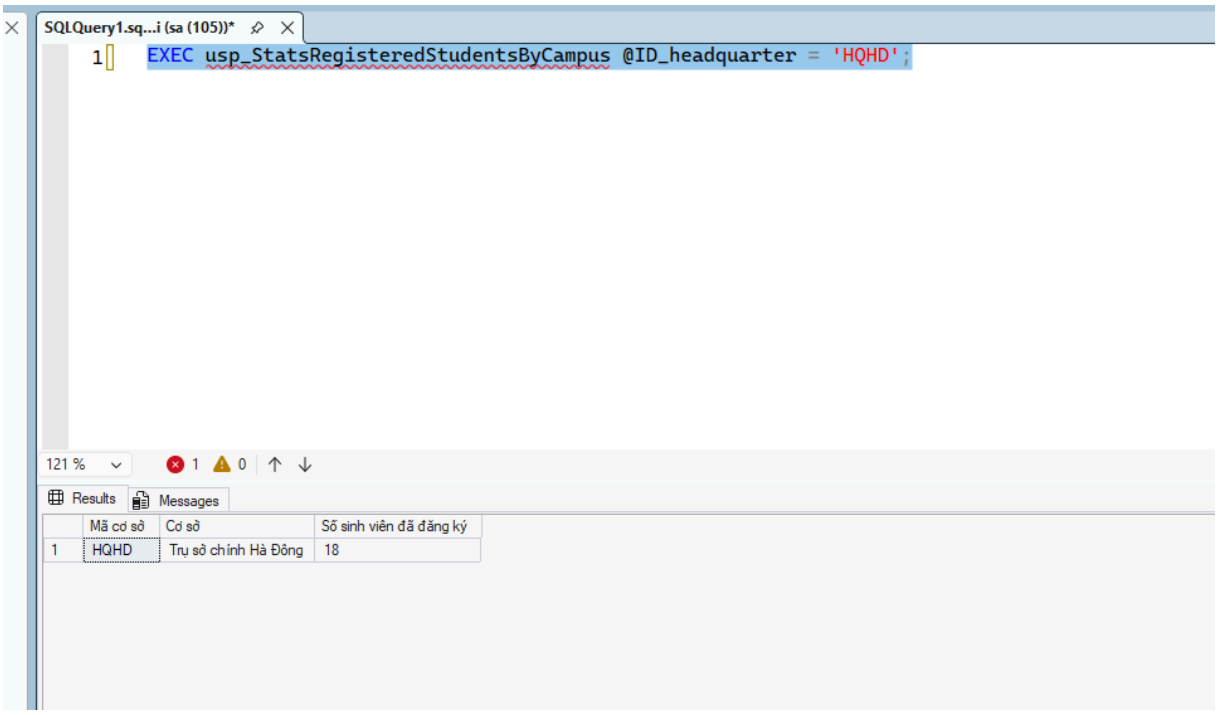
- Bước 5 (Gom nhóm và Tính toán chỉ số):

- *Gom nhóm (GROUP BY):* Dữ liệu được phân cụm theo Mã cơ sở và Tên cơ sở.
- *Logic đếm (COUNT DISTINCT):* Hệ thống sử dụng hàm COUNT(DISTINCT r.ID_student). Vì một sinh viên có thể đăng ký rất nhiều lớp học phần khác nhau (tạo ra nhiều bản ghi trong bảng đăng ký), từ khóa DISTINCT giúp thuật toán loại bỏ các kết quả trùng lặp, đảm bảo mỗi sinh viên chỉ được đếm duy nhất 1 lần (đếm số lượng "đầu người" thay vì đếm số lượt đăng ký).

- Bước 6 (Kết xuất): Dữ liệu được sắp xếp (ORDER BY) theo thứ tự Mã cơ sở và trả về dưới dạng bảng thống kê trực quan.

Kiểm thử Procedure





9. Procedure lấy danh sách lớp học phần theo cơ sở

- Tên Thủ tục (Procedure Name): usp_GetClassesByCampus
- Tham số đầu vào (Input Parameters):
  - @ID_headquarter (Kiểu chuỗi, bắt buộc): Mã cơ sở/trụ sở đào tạo cần tra cứu dữ liệu.
  - @ID_term (Kiểu chuỗi, tùy chọn, mặc định NULL): Mã học kỳ dùng để thu hẹp phạm vi tìm kiếm.
  - @ID_subject (Kiểu chuỗi, tùy chọn, mặc định NULL): Mã môn học cụ thể muốn tra cứu.
- Mục đích: Thủ tục này đóng vai trò là một bộ lọc tìm kiếm động (Dynamic Search Engine) nhằm cung cấp danh mục các lớp học phần khả dụng tại một cơ sở nhất định. Chức năng này đặc biệt quan trọng để phục vụ nghiệp vụ đăng ký học phần chéo cơ sở (ví dụ: Sinh viên cơ sở Hà Nội muốn xem và đăng ký một lớp đang được mở tại cơ sở TP.HCM). Kết quả trả về cung cấp cái nhìn toàn cảnh từ thông tin tổng quát của lớp đến chi tiết thời khóa biểu.

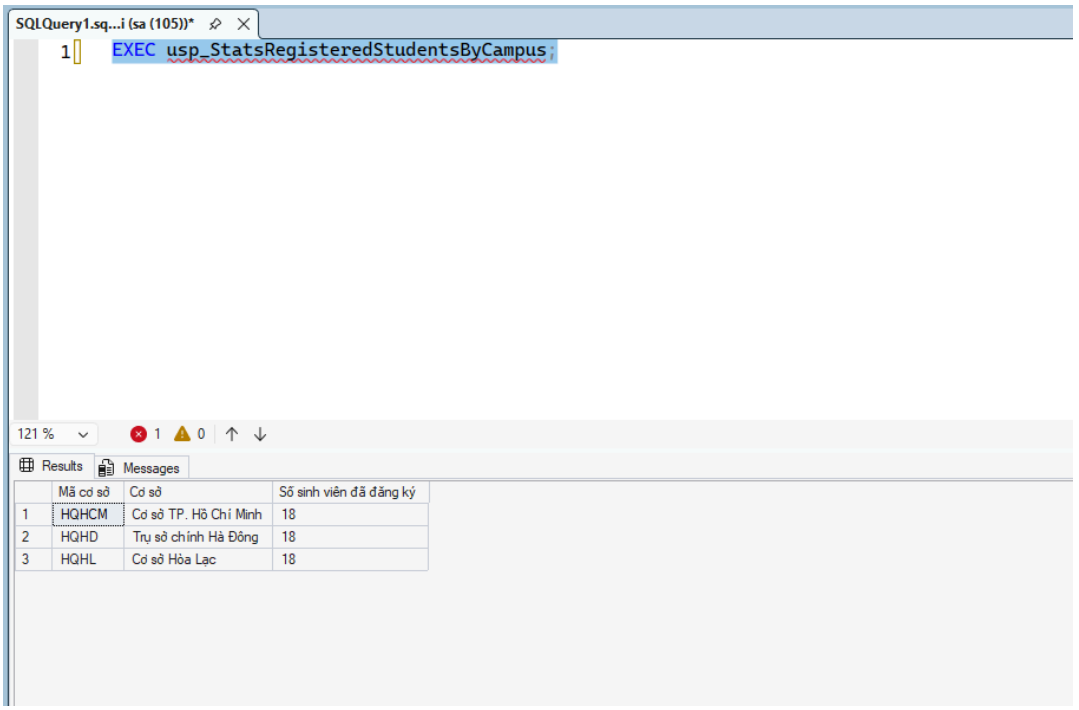
Mô tả giải thuật và logic xử lý: Là một thủ tục truy vấn phục vụ giao diện người dùng (UI), thuật toán tập trung vào kỹ thuật liên kết dữ liệu diện rộng và lọc trạng thái thời gian thực:

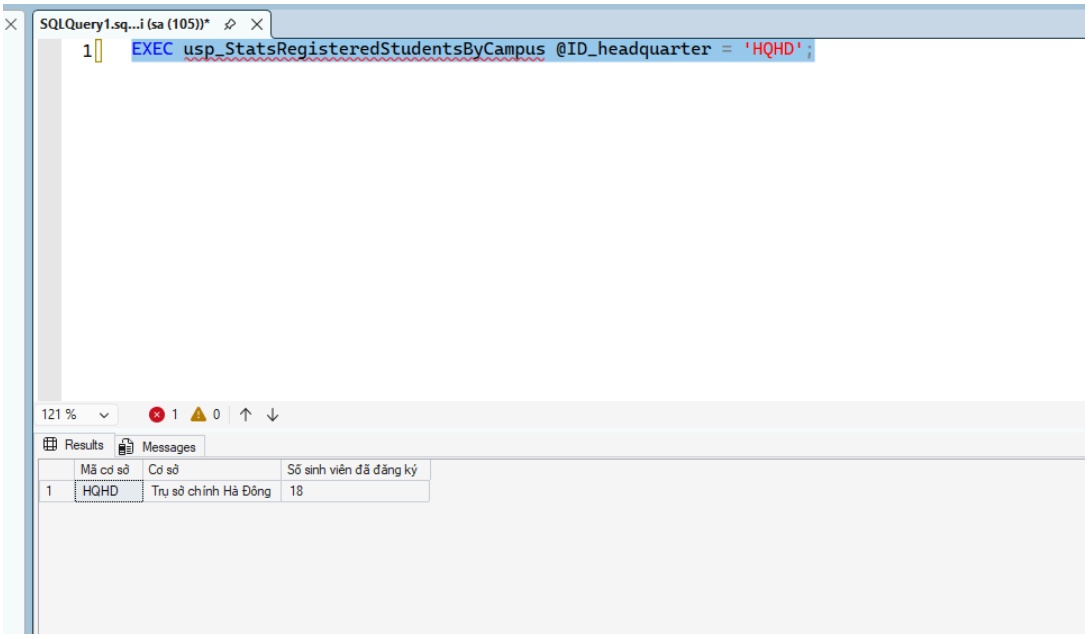
- Bước 1 (Gom cụm và giải mã dữ liệu đa chiều): Thuật toán lấy bảng Lớp học phần (class) làm trung tâm, sau đó tỏa ra thực hiện hàng loạt các phép nối (JOIN) để thu thập đủ 3 nhóm thông tin:

- *Nhóm thông tin học thuật:* Nối với bảng subject để giải mã Tên môn và Số tín chỉ.
- *Nhóm thông tin nhân sự & định vị:* Nối với bảng teacher, department và headquarter để xác định chính xác giảng viên đứng lớp và cơ sở đào tạo quản lý lớp học này.
- *Nhóm thông tin không gian - thời gian:* Nối với các bảng session (buổi học), timeslot (ca học), và room (phòng học) để thiết lập một mảng thời khóa biểu chi tiết (Ngày học, Thứ, Giờ bắt đầu/kết thúc, Phòng học).

- Bước 2 (Kiểm soát đồng bộ bảng mã - Đặc thù CSDL Phân tán): Để đáp ứng kiến trúc phân tán, nơi truy vấn này có thể được gọi từ một trạm này để trích xuất dữ liệu từ một trạm khác (thông qua Linked Server), tất cả các phép nối (ON) và điều kiện (WHERE) đều được hệ thống gắn chỉ thị đồng bộ tập ký tự COLLATE DATABASE_DEFAULT. Điều này giúp triệt tiêu nguy cơ sập luồng truy vấn do xung đột Collation giữa các Server.
- Bước 3 (Sàng lọc nghiệp vụ cốt lõi): Hệ thống áp dụng 3 màng lọc bắt buộc để đảm bảo tính hợp lệ của lớp học được hiển thị:
  - Lớp phải thuộc đúng cơ sở được yêu cầu (@ID_headquarter).
  - Trạng thái lớp phải đang mở (class_status = 'OPEN').
  - Lớp phải còn khả năng tiếp nhận sinh viên. Thuật toán kiểm tra điều kiện: Số lượng đã đăng ký phải nhỏ hơn Sĩ số tối đa (number_of_registration < max_students). Các lớp đã đầy sẽ tự động bị loại khỏi danh sách hiển thị để tối ưu hóa trải nghiệm người dùng.
- Bước 4 (Xử lý truy vấn động - Dynamic Filtering): Thuật toán sử dụng cấu trúc OR ... IS NULL để xử lý các tham số tùy chọn. Nếu người dùng (hoặc ứng dụng) truyền vào mã học kỳ (@ID_term) hoặc mã môn học (@ID_subject), hệ thống sẽ lọc chính xác theo các tiêu chí đó. Nếu để trống, hệ thống sẽ bỏ qua bộ lọc này.
- Bước 5 (Tính toán phái sinh và Kết xuất): * Hệ thống tự động tính toán ra một trường dữ liệu ảo là "Số chỗ còn lại" dựa trên công thức lấy sĩ số tối đa trừ đi số lượng đã đăng ký.
  - Cuối cùng, dữ liệu được trình bày dưới dạng bảng và sắp xếp (ORDER BY) theo tính logic: Nhóm theo Môn học -> Nhóm theo Mã Lớp -> Trình tự Ngày học và Giờ bắt đầu.

Kiểm thử Procedure và Nhận xét:





Nhận xét: Procedure hoạt động trơn tru, trả về đầy đủ các thông tin cần thiết nhất để sinh viên có thể ra quyết định đăng ký. Việc kết hợp kiểm tra trạng thái 'OPEN' và 'còn chỗ trống' ngay từ khâu truy xuất danh sách giúp giảm tải đáng kể các giao dịch đăng ký rác (đăng ký vào lớp đã đầy) đẩy lên Server, góp phần tối ưu hóa hiệu năng hệ thống mạng phân tán.

10. Procedure đăng ký chéo cơ sở

- Tên Thủ tục (Procedure Name): usp_RegisterCrossCampusClass
- Tham số đầu vào (Input Parameters):
  - @ID_registration (Kiểu chuỗi): Mã định danh duy nhất của giao dịch đăng ký (tự sinh).

    - @ID_student (Kiểu chuỗi): Mã sinh viên thực hiện giao dịch.
    - @ID_class (Kiểu chuỗi): Mã lớp học phần mong muốn đăng ký.
- Mục đích: Thủ tục này xử lý một nghiệp vụ đặc thù và phức tạp trong mô hình phân tán: Đăng ký học vượt tuyến/vùng. Nó cho phép sinh viên thuộc sự quản lý của cơ sở đào tạo này (ví dụ: Hà Nội) có thể ghi danh vào một lớp học phần do cơ sở đào tạo khác (ví dụ: TP.HCM) tổ chức. Thủ tục kết hợp một chuỗi các màng lọc kiểm duyệt học vụ khắt khe và cơ chế khóa dữ liệu cấp cao để bảo vệ tính toàn vẹn của hệ thống khi có giao dịch chéo.

Mô tả giải thuật và logic xử lý (Bảo vệ bằng Transaction): Toàn bộ luồng nghiệp vụ được đóng gói trong một khối giao dịch (TRY...CATCH / BEGIN TRAN) để đảm bảo tính nguyên tử (Atomicity). Thuật toán vận hành qua các bước sau:

- Bước 1 (Truy vết và xác thực cơ sở gốc của Sinh viên): * Hệ thống kết nối dữ liệu từ bảng student sang bảng department để trích xuất ra mã cơ sở gốc (@student_site) nơi sinh viên đang trực thuộc.
    - Nếu không tìm thấy sinh viên, giao dịch lập tức bị ngắt với mã lỗi 51001.
- Bước 2 (Truy vết và xác thực cơ sở mở Lớp học phần):
    - Tương tự, hệ thống dò tìm từ bảng class qua teacher và department để định vị chính xác cơ sở nào đang đứng ra tổ chức lớp học phần này (@class_site).
    - Nếu lớp không tồn tại, giao dịch bị ngắt với mã lỗi 51002.
- Bước 3 (Kiểm duyệt logic chéo cơ sở):
    - Thuật toán đối chiếu trực tiếp cơ sở của sinh viên (@student_site) và cơ sở của lớp học (@class_site).
    - Nếu hai giá trị này trùng khớp, hệ thống nhận định đây là một giao dịch đăng ký nội bộ thông thường chứ không phải "đăng ký chéo", qua đó từ chối thực thi với mã lỗi 51003 để điều hướng người dùng về đúng Procedure nghiệp vụ chuẩn.
- Bước 4 (Chuỗi kiểm duyệt quy chế đào tạo): Thuật toán thực hiện hàng loạt các rà soát lịch sử và thời gian thực, bao gồm:
    - Chống trùng lặp mã giao dịch.
    - Đảm bảo lớp đang ở trạng thái 'OPEN' và thời gian hiện tại nằm trong hạn mức cho phép của học kỳ.
    - Đảm bảo sinh viên chưa từng đăng ký lớp này.
    - Đảm bảo sinh viên không đăng ký hai lớp khác nhau nhưng thuộc cùng một môn học trong cùng một kỳ.
    - Đảm bảo lịch học (ngày học, ca học) của lớp mới không bị giao thoa với bất kỳ lớp nào mà sinh viên đã đăng ký trước đó.
    - *(Nếu bất kỳ điều kiện nào vi phạm, tiến trình sẽ bị ngắt và trả về các mã lỗi tương ứng từ 51004 đến 51008).*
- Bước 5 (Kiểm soát đồng thời và Cập nhật sĩ số - Lõi thuật toán):

- Sau khi qua vòng kiểm duyệt cá nhân, thuật toán thực hiện lệnh tăng sĩ số lớp học (number_of_registration) lên 1 đơn vị với điều kiện sức chứa vẫn chưa đầy (< max_students).
- *Cơ chế Locking cấp cao:* Lệnh cập nhật này được ép từ khóa WITH (UPDLOCK, HOLDLOCK, ROWLOCK). Sự kết hợp này mang lại mức độ bảo vệ tối đa: ROWLOCK chỉ khóa dòng dữ liệu của đúng lớp đó; UPDLOCK ngăn chặn các giao dịch khác tranh cướp quyền đọc-ghi; và đặc biệt HOLDLOCK giữ chặt khóa này cho đến khi toàn bộ Transaction kết thúc (tương đương mức cô lập Serializable).
- Nếu lớp đã hết chỗ, lệnh cập nhật sẽ không tác động đến dòng nào (bắt sự kiện thông qua @@ROWCOUNT = 0), hệ thống lập tức ném lỗi 51009: *"Lớp học phần đã đủ sĩ số"* và từ chối ghi nhận.

- Bước 6 (Ghi nhận giao dịch và Hoàn tất):
  - Nếu vượt qua Bước 5 thành công (nghĩa là đã giữ được chỗ), hệ thống chính thức thêm mới dữ liệu vào bảng registration và gọi lệnh COMMIT để lưu vĩnh viễn dữ liệu, kết thúc giao dịch xuyên cơ sở an toàn. Ngược lại, nếu có lỗi, ROLLBACK sẽ được kích hoạt để khôi phục nguyên trạng.

Kiểm thử Procedure và Nhận xét:

```sql
USE DkyTinChi;
GO

EXEC usp_RegisterCrossCampusClass
    @ID_registration = 'REG_CROSS_HL_HCM_001',
    @ID_student = 'B22ATTT006',
    @ID_class = 'L03_BUS1301';
GO

USE DkyTinChi;
GO

EXEC usp_CheckCrossCampusRegistration;
GO
```

133 % 1 0 Ln: 14, Ch:

Results | Messages

	Mã đăng ký	Mã sinh viên	Tên sinh viên	Cơ sở sinh viên	Mã lớp học phần	Tên học phần	Cơ sở mở lớp	Thời gian đăng ký
1	REG_CROSS_HL_HCM_001	B22ATTT006	Nguyễn Thảo Nhi	Cơ sở Hòa Lạc	L03_BUS1301	Marketing căn bản	Cơ sở TP. Hồ Chí Minh	2026-06-02 02:33:47.667
2	REG064	B22QTKD005	Cao Thị Hương	Cơ sở Hòa Lạc	L01_BUS1401	Thương mại điện tử	Trụ sở chính Hà Đông	2026-05-30 18:46:35.817
3	REG067	B22QTKD006	Nguyễn Gia Bảo	Cơ sở Hòa Lạc	L01_BUS1401	Thương mại điện tử	Trụ sở chính Hà Đông	2026-05-30 18:49:35.817
4	REG127	B23QTKD005	Đỗ Minh Nhật	Cơ sở Hòa Lạc	L01_BUS1401	Thương mại điện tử	Trụ sở chính Hà Đông	2026-05-30 19:49:35.817
5	REG130	B23QTKD006	Trần Khánh Vy	Cơ sở Hòa Lạc	L01_BUS1401	Thương mại điện tử	Trụ sở chính Hà Đông	2026-05-30 19:52:35.817
6	REG054	B22QTKD001	Hoàng Bảo Châu	Cơ sở TP. Hồ Chí Minh	L02_BUS1401	Thương mại điện tử	Cơ sở Hòa Lạc	2026-05-30 18:36:35.817
7	REG057	B22QTKD002	Vũ Anh Tuấn	Cơ sở TP. Hồ Chí Minh	L02_BUS1401	Thương mại điện tử	Cơ sở Hòa Lạc	2026-05-30 18:39:35.817
8	REG117	B23QTKD001	Cao Trung Hiếu	Cơ sở TP. Hồ Chí Minh	L02_BUS1401	Thương mại điện tử	Cơ sở Hòa Lạc	2026-05-30 19:39:35.817
9	REG120	B23QTKD002	Đinh Ngọc Ánh	Cơ sở TP. Hồ Chí Minh	L02_BUS1401	Thương mại điện tử	Cơ sở Hòa Lạc	2026-05-30 19:42:35.817
10	REG_TEST_OK_004	B22CNTT001	Phạm Minh Khôi	Cơ sở TP. Hồ Chí Minh	L_TEST_FULL	Tư tưởng Hồ Chí Minh	Trụ sở chính Hà Đông	2026-05-31 06:43:50.477
11	REG015	B22CNTT001	Phạm Minh Khôi	Cơ sở TP. Hồ Chí Minh	L01_INT1358	Lập trình Web	Trụ sở chính Hà Đông	2026-05-30 17:57:35.817
12	REG018	B22CNTT002	Trần Phương Nam	Cơ sở TP. Hồ Chí Minh	L01_INT1358	Lập trình Web	Trụ sở chính Hà Đông	2026-05-30 18:00:35.817
13	REG021	B22CNTT003	Võ Nhật Anh	Cơ sở TP. Hồ Chí Minh	L01_INT1358	Lập trình Web	Trụ sở chính Hà Đông	2026-05-30 18:03:35.817
14	REG084	B23CNTT001	Đặng Thùy Linh	Cơ sở TP. Hồ Chí Minh	L01_INT1358	Lập trình Web	Trụ sở chính Hà Đông	2026-05-30 19:06:35.817
15	REG087	B23CNTT002	Lê Anh Khoa	Cơ sở TP. Hồ Chí Minh	L01_INT1358	Lập trình Web	Trụ sở chính Hà Đông	2026-05-30 19:09:35.817
16	REG090	B23CNTT003	Nguyễn Hà My	Cơ sở TP. Hồ Chí Minh	L01_INT1358	Lập trình Web	Trụ sở chính Hà Đông	2026-05-30 19:12:35.817

Nhận xét: Procedure hoạt động vô cùng chặt chẽ. Nó không chỉ giải quyết bài toán định tuyến dữ liệu chéo giữa các phân mảnh mà còn áp dụng thành công các kỹ thuật kiểm soát tranh chấp khắt khe nhất (UPDLOCK, HOLDLOCK), đảm bảo an

toàn tuyệt đối cho tài nguyên lớp học khi có hiện tượng cạnh tranh đăng ký liên cơ sở trong hệ thống.

PHẦN 5: TRUY VẤN PHÂN TÁN

I. Mục tiêu của truy vấn phân tán

Truy vấn phân tán được sử dụng để lấy và tổng hợp dữ liệu từ nhiều site trong hệ thống cơ sở dữ liệu phân tán. Trong hệ thống đăng ký học phần nhiều cơ sở, dữ liệu sinh viên, lớp học phần, lịch học và đăng ký học phần được phân mảnh theo từng cơ sở. Vì vậy, để thống kê toàn trường, máy chủ trung tâm cần truy vấn dữ liệu từ các site cơ sở.

Trong hệ thống này, các truy vấn phân tán được thực hiện tại máy chủ trung tâm Hà Đông. Dữ liệu cục bộ Hà Đông được lấy trực tiếp từ cơ sở dữ liệu DkyTinChi. Hệ thống có sử dụng Replication để đồng bộ dữ liệu giữa máy chủ và máy trạm; tuy nhiên, trong phần truy vấn phân tán, dữ liệu Hòa Lạc và TP. Hồ Chí Minh được truy vấn thông qua Linked Server nhằm thể hiện rõ quá trình lấy dữ liệu trực tiếp từ nhiều site và tổng hợp kết quả tại máy chủ trung tâm.

Các site sử dụng trong truy vấn phân tán:

Site Hà Đông:

- Cơ sở dữ liệu: DkyTinChi
- Vai trò: máy chủ trung tâm kiêm cơ sở chính

Site Hòa Lạc:

- Cơ sở dữ liệu: CSDL_HL
- Linked Server: LINK_HL

Site TP. Hồ Chí Minh:

- Cơ sở dữ liệu: DkiTinChi_HCM
- Linked Server: Link_HoChiMinh

Mục tiêu của phần này là xây dựng các truy vấn có khả năng tổng hợp dữ liệu toàn trường, bao gồm: thống kê số sinh viên đăng ký theo cơ sở, tìm học phần có nhiều sinh viên đăng ký nhất, liệt kê sinh viên đăng ký chéo cơ sở, tính tỷ lệ lấp đầy lớp học phần và thống kê số lớp mở theo khoa hoặc cơ sở.

II. Phương pháp thực hiện truy vấn phân tán

Các truy vấn phân tán được chạy tại máy chủ trung tâm Hà Đông. Máy chủ trung tâm truy vấn dữ liệu Hà Đông trực tiếp từ cơ sở dữ liệu DkyTinChi, đồng thời truy vấn dữ liệu từ các site Hòa Lạc và TP. Hồ Chí Minh thông qua Linked Server.

Để tối ưu hiệu năng, các truy vấn được thiết kế theo hướng đẩy điều kiện lọc và phép tổng hợp xuống từng site. Thay vì lấy toàn bộ dữ liệu từ máy trạm về máy chủ rồi mới xử lý, mỗi site tự thực hiện các điều kiện WHERE, GROUP BY, COUNT trước. Máy chủ trung tâm chỉ nhận kết quả đã được rút gọn và tổng hợp lại bằng UNION ALL.

Đối với dữ liệu từ xa, hệ thống sử dụng OPENQUERY để câu lệnh SQL được thực thi trực tiếp tại máy trạm. Cách này giúp giảm lượng dữ liệu truyền qua mạng, giảm chi phí xử lý tại máy chủ trung tâm và phù hợp với mô hình cơ sở dữ liệu phân tán.

Do dữ liệu giữa các site có thể khác Collation, các truy vấn sử dụng COLLATE DATABASE_DEFAULT trong các phép so sánh chuỗi và khi gom dữ liệu từ nhiều site. Cách này giúp hạn chế lỗi xung đột Collation trong quá trình JOIN, WHERE và UNION ALL.

Các kỹ thuật sử dụng trong truy vấn:

- Linked Server để kết nối đến các site từ xa.
- OPENQUERY để đẩy truy vấn xuống máy trạm.
- UNION ALL để gom dữ liệu từ nhiều site.
- GROUP BY, COUNT, SUM để tổng hợp dữ liệu.
- COLLATE DATABASE_DEFAULT để tránh lỗi khác Collation giữa các cơ sở dữ liệu.
- Điều kiện lọc theo cơ sở như HQHD, HQHL, HQHCM để hạn chế dữ liệu truyền qua mạng.

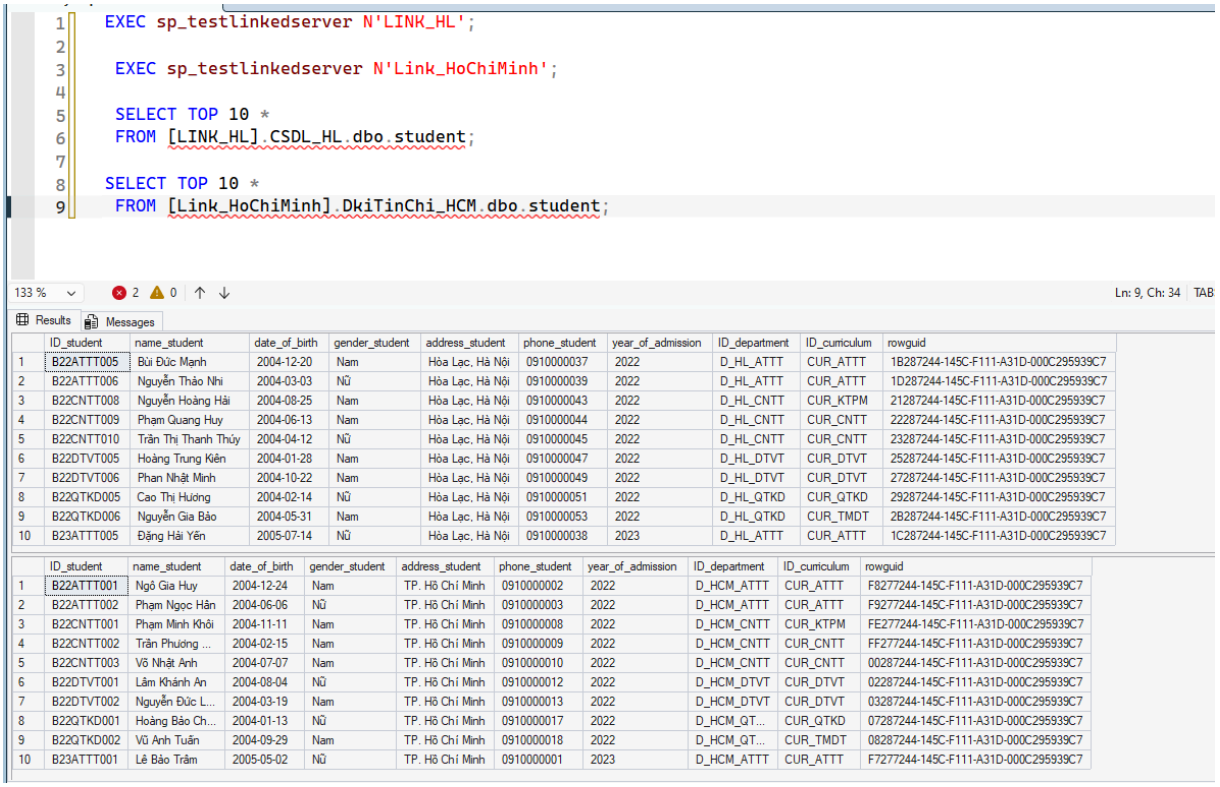
Kiểm tra kết nối Linked Server:

```
EXEC sp_testlinkedserver N'LINK_HL';
EXEC sp_testlinkedserver N'Link_HoChiMinh';
```

Kiểm tra truy vấn dữ liệu từ xa:

```
SELECT TOP 10 *
FROM [LINK_HL].CSDL_HL.dbo.student;
```

SELECT TOP 10 *
FROM [Link_HoChiMinh].DkiTinChi_HCM.dbo.student;



```
EXEC sp_testlinkedserver N'LINK_HL';

 EXEC sp_testlinkedserver N'Link_HoChiMinh';

 SELECT TOP 10 *
 FROM [LINK_HL].CSDL_HL.dbo.student;

SELECT TOP 10 *
 FROM [Link_HoChiMinh].DkiTinChi_HCM.dbo.student;
```

133 % 2 0 Ln: 9, Ch: 34 TAB

Results Messages

	ID_student	name_student	date_of_birth	gender_student	address_student	phone_student	year_of_admission	ID_department	ID_curriculum	rowguid
1	B22ATTT005	Bùi Đức Mạnh	2004-12-20	Nam	Hòa Lạc, Hà Nội	0910000037	2022	D_HL_ATTT	CUR_ATTT	1B287244-145C-F111-A31D-000C295939C7
2	B22ATTT006	Nguyễn Thảo Nhi	2004-03-03	Nữ	Hòa Lạc, Hà Nội	0910000039	2022	D_HL_ATTT	CUR_ATTT	1D287244-145C-F111-A31D-000C295939C7
3	B22CNTT008	Nguyễn Hoàng Hải	2004-08-25	Nam	Hòa Lạc, Hà Nội	0910000043	2022	D_HL_CNTT	CUR_KTPM	21287244-145C-F111-A31D-000C295939C7
4	B22CNTT009	Phạm Quang Huy	2004-06-13	Nam	Hòa Lạc, Hà Nội	0910000044	2022	D_HL_CNTT	CUR_CNTT	22287244-145C-F111-A31D-000C295939C7
5	B22CNTT010	Trần Thị Thanh Thúy	2004-04-12	Nữ	Hòa Lạc, Hà Nội	0910000045	2022	D_HL_CNTT	CUR_CNTT	23287244-145C-F111-A31D-000C295939C7
6	B22DTVT005	Hoàng Trung Kiên	2004-01-28	Nam	Hòa Lạc, Hà Nội	0910000047	2022	D_HL_DTVT	CUR_DTVT	25287244-145C-F111-A31D-000C295939C7
7	B22DTVT006	Phan Nhật Minh	2004-10-22	Nam	Hòa Lạc, Hà Nội	0910000049	2022	D_HL_DTVT	CUR_DTVT	27287244-145C-F111-A31D-000C295939C7
8	B22QTKD005	Cao Thị Hương	2004-02-14	Nữ	Hòa Lạc, Hà Nội	0910000051	2022	D_HL_QTKD	CUR_QTKD	29287244-145C-F111-A31D-000C295939C7
9	B22QTKD006	Nguyễn Gia Bảo	2004-05-31	Nam	Hòa Lạc, Hà Nội	0910000053	2022	D_HL_QTKD	CUR_TMDT	2B287244-145C-F111-A31D-000C295939C7
10	B23ATTT005	Đặng Hải Yến	2005-07-14	Nữ	Hòa Lạc, Hà Nội	0910000038	2023	D_HL_ATTT	CUR_ATTT	1C287244-145C-F111-A31D-000C295939C7

	ID_student	name_student	date_of_birth	gender_student	address_student	phone_student	year_of_admission	ID_department	ID_curriculum	rowguid
1	B22ATTT001	Ngô Gia Huy	2004-12-24	Nam	TP. Hồ Chí Minh	0910000002	2022	D_HCM_ATTT	CUR_ATTT	F8277244-145C-F111-A31D-000C295939C7
2	B22ATTT002	Phạm Ngọc Hân	2004-06-06	Nữ	TP. Hồ Chí Minh	0910000003	2022	D_HCM_ATTT	CUR_ATTT	F9277244-145C-F111-A31D-000C295939C7
3	B22CNTT001	Phạm Minh Khôi	2004-11-11	Nam	TP. Hồ Chí Minh	0910000008	2022	D_HCM_CNTT	CUR_KTPM	FE277244-145C-F111-A31D-000C295939C7
4	B22CNTT002	Trần Phương ...	2004-02-15	Nam	TP. Hồ Chí Minh	0910000009	2022	D_HCM_CNTT	CUR_CNTT	FF277244-145C-F111-A31D-000C295939C7
5	B22CNTT003	Võ Nhật Anh	2004-07-07	Nam	TP. Hồ Chí Minh	0910000010	2022	D_HCM_CNTT	CUR_CNTT	00287244-145C-F111-A31D-000C295939C7
6	B22DTVT001	Lâm Khánh An	2004-08-04	Nữ	TP. Hồ Chí Minh	0910000012	2022	D_HCM_DTVT	CUR_DTVT	02287244-145C-F111-A31D-000C295939C7
7	B22DTVT002	Nguyễn Đức L...	2004-03-19	Nam	TP. Hồ Chí Minh	0910000013	2022	D_HCM_DTVT	CUR_DTVT	03287244-145C-F111-A31D-000C295939C7
8	B22QTKD001	Hoàng Bảo Ch...	2004-01-13	Nữ	TP. Hồ Chí Minh	0910000017	2022	D_HCM_QT...	CUR_QTKD	07287244-145C-F111-A31D-000C295939C7
9	B22QTKD002	Vũ Anh Tuấn	2004-09-29	Nam	TP. Hồ Chí Minh	0910000018	2022	D_HCM_QT...	CUR_TMDT	08287244-145C-F111-A31D-000C295939C7
10	B23ATTT001	Lê Bảo Trâm	2005-05-02	Nữ	TP. Hồ Chí Minh	0910000001	2023	D_HCM_ATTT	CUR_ATTT	F7277244-145C-F111-A31D-000C295939C7

Nhận xét: Việc truy vấn thành công dữ liệu từ các site cho thấy Linked Server đã được cấu hình đúng và máy chủ trung tâm có thể thực hiện truy vấn phân tán.

III. Các truy vấn phân tán

<u>Truy vấn 1: Thống kê số sinh viên đăng ký học phần theo từng cơ sở</u>

1. Mục tiêu truy vấn

Truy vấn này dùng để thống kê số lượng sinh viên đã đăng ký học phần tại từng cơ sở đào tạo. Kết quả giúp nhà trường theo dõi quy mô đăng ký học phần tại Hà Đông, Hòa Lạc và TP. Hồ Chí Minh.

2. Dữ liệu sử dụng

Site Hà Đông:

- DkyTinChi.dbo.registration
- DkyTinChi.dbo.student
- DkyTinChi.dbo.department
- DkyTinChi.dbo.headquarter

Site Hòa Lạc:

- CSDL_HL.dbo.registration

- CSDL_HL.dbo.student
- CSDL_HL.dbo.department
- CSDL_HL.dbo.headquarter
- Truy vấn qua Linked Server LINK_HL

Site TP. Hồ Chí Minh:

- DkiTinChi_HCM.dbo.registration
- DkiTinChi_HCM.dbo.student
- DkiTinChi_HCM.dbo.department
- DkiTinChi_HCM.dbo.headquarter
- Truy vấn qua Linked Server Link_HoChiMinh

3. Cách tổng hợp

Mỗi site tự lọc dữ liệu theo cơ sở của mình và đếm số sinh viên đã đăng ký học phần. Cụ thể, Hà Đông lọc HQHD, Hòa Lạc lọc HQHL, TP. Hồ Chí Minh lọc HQHCM. Kết quả từ các site sau đó được gom lại bằng UNION ALL. Vì mỗi site đã tự tổng hợp trước nên lượng dữ liệu truyền về máy chủ trung tâm nhỏ hơn so với việc lấy toàn bộ bảng đăng ký.

```
129                 AND hq.ID_headquarter = ''HQHCM''
130
131             GROUP BY hq.ID_headquarter, hq.name_headquarter
132
133             '
134
135         )
136
137     )
138
139     SELECT
140
141         ID_headquarter AS [Mã cơ sở],
142
143         name_headquarter AS [Cơ sở],
144
145         student_count AS [Số sinh viên đã đăng ký]
146
147     FROM q
148
149     ORDER BY ID_headquarter;
```

133 % No issues found

Results | Messages

	Mã cơ sở	Cơ sở	Số sinh viên đã đăng ký
1	HQHCM	Cơ sở TP. Hồ Chí Minh	18
2	HQHD	Trụ sở chính Hà Đông	18
3	HQHL	Cơ sở Hòa Lạc	19

Nhận xét: Truy vấn đã lấy dữ liệu đăng ký từ ba site và tổng hợp theo từng cơ sở. Việc sử dụng OPENQUERY giúp mỗi máy trạm tự đếm dữ liệu tại chỗ, máy chủ trung tâm chỉ nhận kết quả tổng hợp. Điều này giúp giảm dữ liệu truyền qua mạng và phù hợp với yêu cầu truy vấn phân tán.

Truy vấn 2: Tìm học phần có nhiều sinh viên đăng ký nhất toàn trường

1. Mục tiêu truy vấn

Truy vấn này dùng để tìm học phần có số lượng sinh viên đăng ký nhiều nhất trên toàn hệ thống. Kết quả giúp nhà trường xác định các học phần có nhu cầu học cao.

2. Dữ liệu sử dụng

Site Hà Đông:

- DkyTinChi.dbo.registration
- DkyTinChi.dbo.class
- DkyTinChi.dbo.subject
- DkyTinChi.dbo.teacher
- DkyTinChi.dbo.department

Site Hòa Lạc:

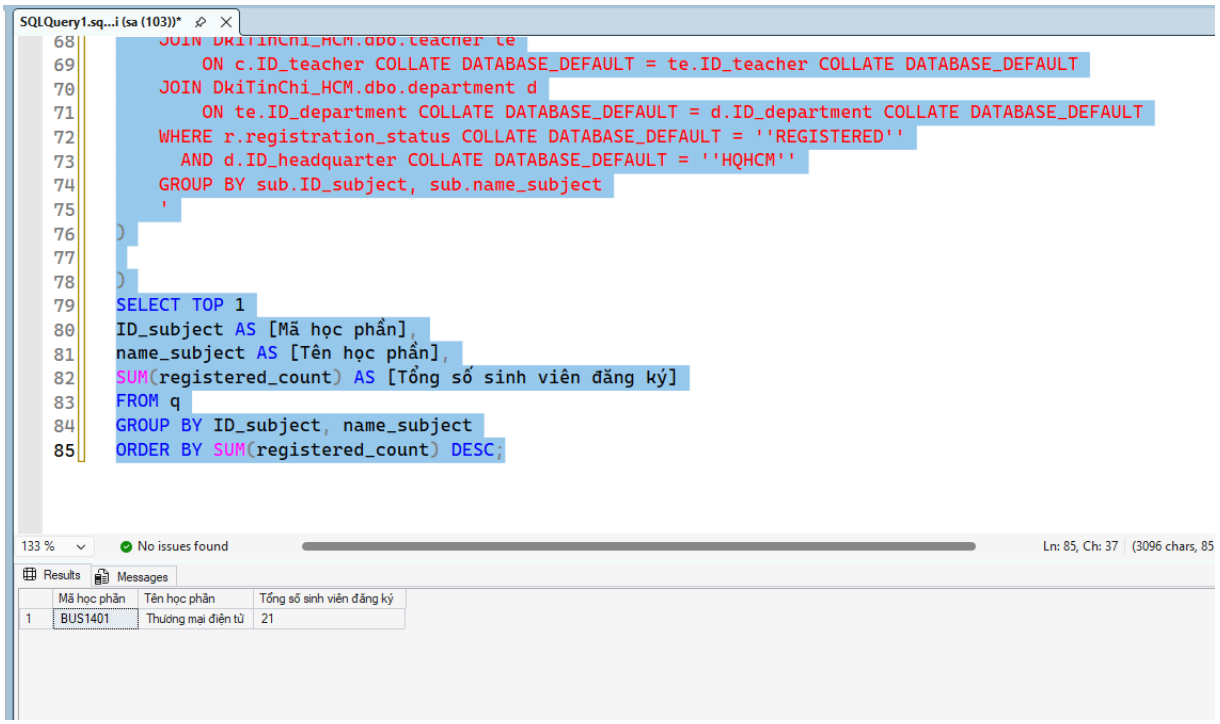
- CSDL_HL.dbo.registration
- CSDL_HL.dbo.class
- CSDL_HL.dbo.subject
- CSDL_HL.dbo.teacher
- CSDL_HL.dbo.department
- Truy vấn qua LINK_HL

Site TP. Hồ Chí Minh:

- DkiTinChi_HCM.dbo.registration
- DkiTinChi_HCM.dbo.class
- DkiTinChi_HCM.dbo.subject
- DkiTinChi_HCM.dbo.teacher
- DkiTinChi_HCM.dbo.department
- Truy vấn qua Link_HoChiMinh

3. Cách tổng hợp

Mỗi site tự thống kê số lượt đăng ký theo từng học phần. Sau đó máy chủ trung tâm gom kết quả bằng UNION ALL, nhóm lại theo mã học phần và cộng tổng số đăng ký bằng SUM. Học phần có số đăng ký cao nhất được lấy bằng TOP 1 kết hợp ORDER BY giảm dần.



Nhận xét: Truy vấn đã tổng hợp số lượng đăng ký học phần từ ba site. Do mỗi site tự thực hiện GROUP BY trước, máy chủ trung tâm không cần lấy toàn bộ bảng registration từ các máy trạm. Kết quả cho biết học phần có tổng số lượt đăng ký cao nhất toàn trường.

Truy vấn 3: Danh sách sinh viên đăng ký chéo cơ sở

1. Mục tiêu truy vấn

Truy vấn này dùng để liệt kê các trường hợp sinh viên đăng ký lớp học phần thuộc cơ sở khác với cơ sở quản lý sinh viên. Đây là một yêu cầu quan trọng trong hệ thống đăng ký học phần nhiều cơ sở.

2. Dữ liệu sử dụng

Để xác định cơ sở sinh viên:

- student
- department
- headquarter

Để xác định cơ sở mở lớp:

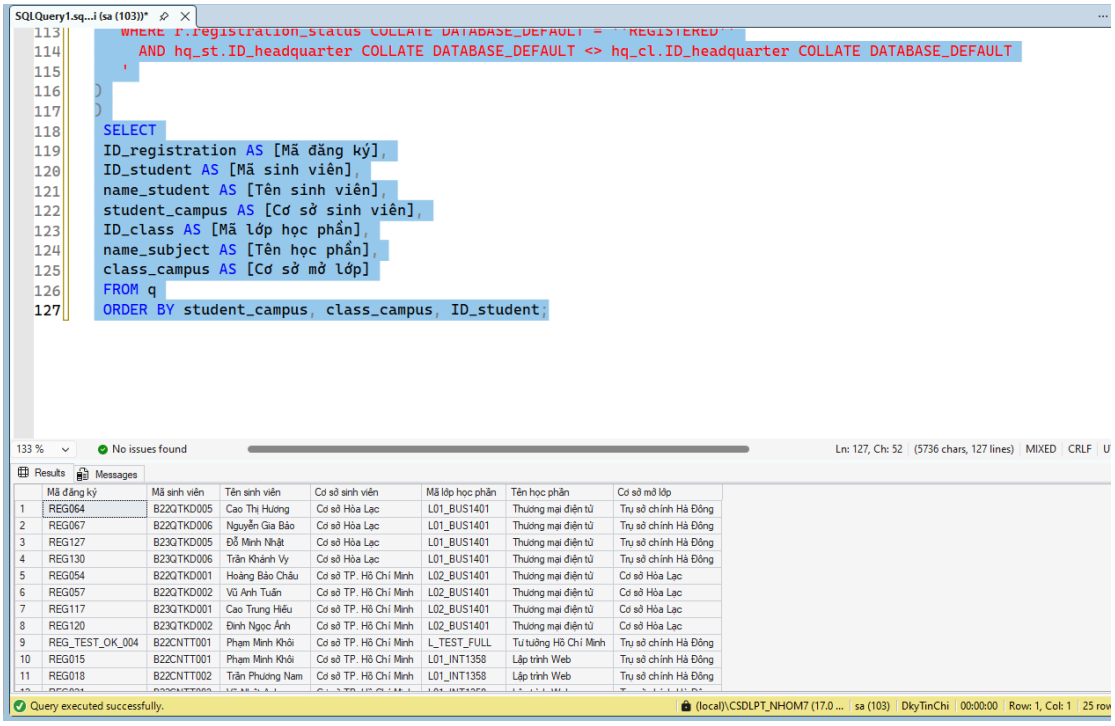
- registration
- class
- subject
- teacher
- department
- headquarter

Dữ liệu được lấy từ ba site:

- Hà Đông: DkyTinChi
- Hòa Lạc: CSDL_HL qua LINK_HL
- TP. Hồ Chí Minh: DkiTinChi_HCM qua Link_HoChiMinh

3. Cách tổng hợp

Tại mỗi site, truy vấn so sánh cơ sở quản lý sinh viên với cơ sở mở lớp học phần. Nếu ID_headquarter của sinh viên khác ID_headquarter của lớp học phần thì bản ghi được xem là đăng ký chéo cơ sở. Mỗi site tự lọc các bản ghi đăng ký chéo, sau đó máy chủ trung tâm gom kết quả bằng UNION ALL.



Nhận xét: Truy vấn đã phát hiện các trường hợp đăng ký chéo cơ sở bằng cách so sánh cơ sở của sinh viên và cơ sở mở lớp học phần. Việc lọc đăng ký chéo được thực hiện trực tiếp tại từng site, sau đó máy chủ trung tâm chỉ gom các bản ghi phù hợp. Truy vấn này đáp ứng yêu cầu của đề tài về danh sách sinh viên đăng ký chéo cơ sở.

<u>Truy vấn 4: Tỷ lệ lấp đầy của các lớp học phần trên toàn hệ thống</u>

1. Mục tiêu truy vấn

Truy vấn này dùng để tính tỷ lệ lấp đầy của các lớp học phần trên toàn hệ thống. Kết quả cho biết lớp học phần nào đã gần đủ sĩ số, lớp nào còn nhiều chỗ trống.

2. Dữ liệu sử dụng

Site Hà Đông:

- DkyTinChi.dbo.class
- DkyTinChi.dbo.subject
- DkyTinChi.dbo.teacher
- DkyTinChi.dbo.department

Site Hòa Lạc:

- CSDL_HL.dbo.class
- CSDL_HL.dbo.subject
- CSDL_HL.dbo.teacher
- CSDL_HL.dbo.department
- Truy vấn qua LINK_HL

Site TP. Hồ Chí Minh:

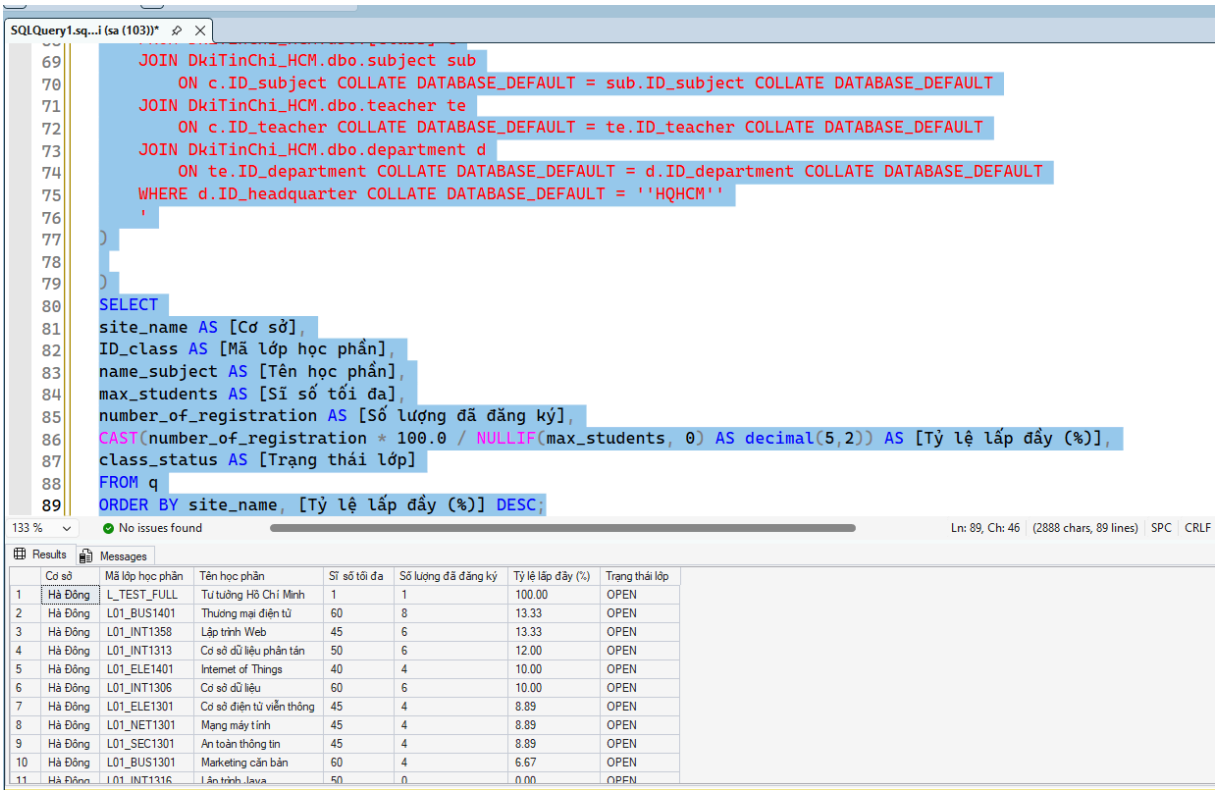
- DkiTinChi_HCM.dbo.class
- DkiTinChi_HCM.dbo.subject
- DkiTinChi_HCM.dbo.teacher
- DkiTinChi_HCM.dbo.department
- Truy vấn qua Link_HoChiMinh

3. Cách tổng hợp

Mỗi site trả về danh sách lớp học phần thuộc cơ sở của mình. Máy chủ trung tâm gom dữ liệu bằng UNION ALL và tính tỷ lệ lấp đầy theo công thức:

Tỷ lệ lấp đầy = number_of_registration * 100 / max_students

Sử dụng NULLIF(max_students, 0) để tránh lỗi chia cho 0.



```
SQLQuery1.sq...i (sa (103))*
69      JOIN DkiTinChi_HCM.dbo.subject sub
70          ON c.ID_subject COLLATE DATABASE_DEFAULT = sub.ID_subject COLLATE DATABASE_DEFAULT
71      JOIN DkiTinChi_HCM.dbo.teacher te
72          ON c.ID_teacher COLLATE DATABASE_DEFAULT = te.ID_teacher COLLATE DATABASE_DEFAULT
73      JOIN DkiTinChi_HCM.dbo.department d
74          ON te.ID_department COLLATE DATABASE_DEFAULT = d.ID_department COLLATE DATABASE_DEFAULT
75      WHERE d.ID_headquarter COLLATE DATABASE_DEFAULT = ''HQHCM''
76      '
77  )
78
79  )
80  SELECT
81  site_name AS [Cơ sở],
82  ID_class AS [Mã lớp học phần],
83  name_subject AS [Tên học phần],
84  max_students AS [Sĩ số tối đa],
85  number_of_registration AS [Số lượng đã đăng ký],
86  CAST(number_of_registration * 100.0 / NULLIF(max_students, 0) AS decimal(5,2)) AS [Tỷ lệ lấp đầy (%)],
87  class_status AS [Trạng thái lớp]
88  FROM q
89  ORDER BY site_name, [Tỷ lệ lấp đầy (%)] DESC;
133 %   No issues found   Ln: 89, Ch: 46   (2888 chars, 89 lines)   SPC   CRLF
Results   Messages
```

	Cơ sở	Mã lớp học phần	Tên học phần	Sĩ số tối đa	Số lượng đã đăng ký	Tỷ lệ lấp đầy (%)	Trạng thái lớp
1	Hà Đông	L_TEST_FULL	Tư tưởng Hồ Chí Minh	1	1	100.00	OPEN
2	Hà Đông	L01_BUS1401	Thương mại điện tử	60	8	13.33	OPEN
3	Hà Đông	L01_INT1358	Lập trình Web	45	6	13.33	OPEN
4	Hà Đông	L01_INT1313	Cơ sở dữ liệu phân tán	50	6	12.00	OPEN
5	Hà Đông	L01_ELE1401	Internet of Things	40	4	10.00	OPEN
6	Hà Đông	L01_INT1306	Cơ sở dữ liệu	60	6	10.00	OPEN
7	Hà Đông	L01_ELE1301	Cơ sở điện tử viễn thông	45	4	8.89	OPEN
8	Hà Đông	L01_NET1301	Mạng máy tính	45	4	8.89	OPEN
9	Hà Đông	L01_SEC1301	An toàn thông tin	45	4	8.89	OPEN
10	Hà Đông	L01_BUS1301	Marketing căn bản	60	4	6.67	OPEN
11	Hà Đông	L01_INT1316	Lập trình Java	50	0	0.00	OPEN

Nhận xét: Truy vấn đã tổng hợp danh sách lớp học phần từ các site và tính tỷ lệ lấp đầy của từng lớp. Kết quả hỗ trợ quản lý đào tạo trong việc theo dõi tình trạng đăng ký và điều chỉnh kế hoạch mở lớp.

Truy vấn 5: Thống kê số lớp học phần mở theo khoa của từng cơ sở

1. Mục tiêu truy vấn

Truy vấn này dùng để thống kê số lượng lớp học phần đang mở theo từng khoa của từng cơ sở. Kết quả phục vụ công tác quản lý kế hoạch mở lớp học phần.

2. Dữ liệu sử dụng

Site Hà Đông:

- DkyTinChi.dbo.class
- DkyTinChi.dbo.teacher
- DkyTinChi.dbo.department
- DkyTinChi.dbo.headquarter

Site Hòa Lạc:

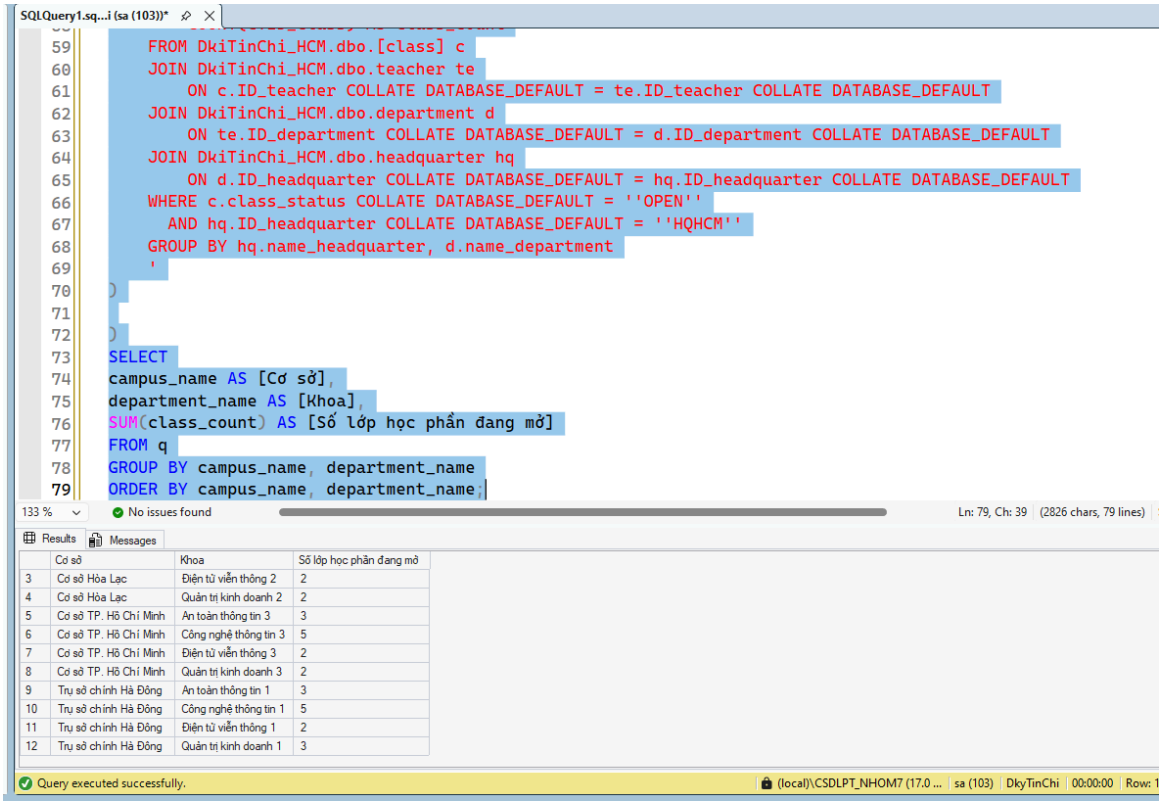
- CSDL_HL.dbo.class
- CSDL_HL.dbo.teacher
- CSDL_HL.dbo.department
- CSDL_HL.dbo.headquarter
- Truy vấn qua LINK_HL

Site TP. Hồ Chí Minh:

- DkiTinChi_HCM.dbo.class
- DkiTinChi_HCM.dbo.teacher
- DkiTinChi_HCM.dbo.department
- DkiTinChi_HCM.dbo.headquarter
- Truy vấn qua Link_HoChiMinh

3. Cách tổng hợp

Mỗi site tự thống kê số lớp học phần đang mở theo khoa. Chỉ các lớp có class_status = 'OPEN' được đưa vào thống kê. Máy chủ trung tâm gom kết quả từ các site bằng UNION ALL, sau đó nhóm lại theo cơ sở và khoa để hiển thị kết quả cuối cùng.



Nhận xét: Truy vấn đã thống kê số lớp học phần đang mở theo cơ sở và khoa. Việc để mỗi site tự GROUP BY trước giúp giảm dữ liệu truyền về máy chủ trung tâm, đồng thời vẫn đảm bảo kết quả tổng hợp toàn trường.

<u>Truy vấn 6: Tra cứu danh sách lớp học phần còn chỗ</u>

1. Mục tiêu truy vấn

Truy vấn này dùng để liệt kê các lớp học phần đang mở và còn chỗ trống trên toàn hệ thống. Kết quả giúp sinh viên biết những lớp học phần nào vẫn còn khả năng đăng ký, đồng thời giúp quản trị viên theo dõi tình trạng đăng ký của các lớp tại từng cơ sở.

2. Dữ liệu sử dụng

Site Hà Đông:

- DkyTinChi.dbo.class
- DkyTinChi.dbo.subject
- DkyTinChi.dbo.teacher
- DkyTinChi.dbo.department

Site Hòa Lạc:

- CSDL_HL.dbo.class
- CSDL_HL.dbo.subject
- CSDL_HL.dbo.teacher
- CSDL_HL.dbo.department
- Truy vấn thông qua Linked Server LINK_HL

Site TP. Hồ Chí Minh:

- DkiTinChi_HCM.dbo.class
- DkiTinChi_HCM.dbo.subject
- DkiTinChi_HCM.dbo.teacher
- DkiTinChi_HCM.dbo.department
- Truy vấn thông qua Linked Server Link_HoChiMinh

3. Cách tổng hợp

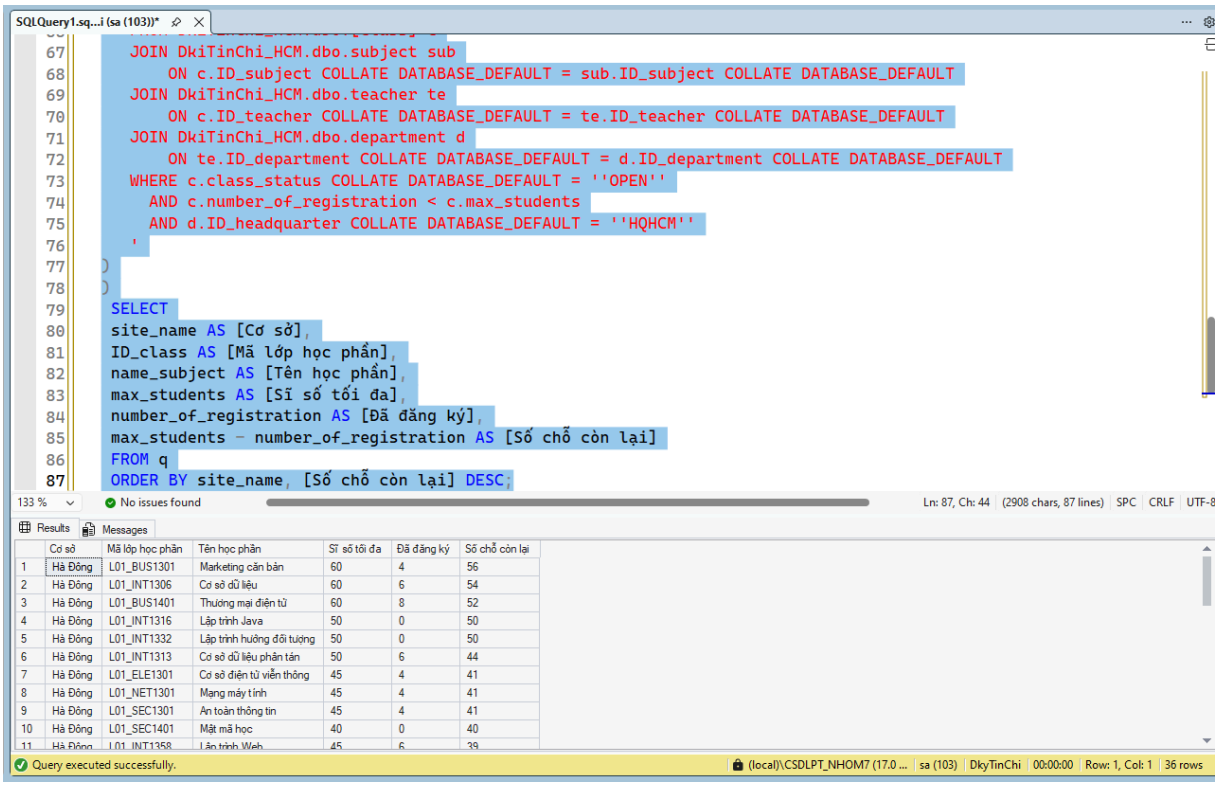
Mỗi site tự lọc các lớp học phần thuộc cơ sở của mình, có trạng thái OPEN và còn chỗ đăng ký. Điều kiện còn chỗ được xác định bằng:

number_of_registration < max_students

Sau đó máy chủ trung tâm Hà Đông gom kết quả từ các site bằng UNION ALL. Số chỗ còn lại được tính bằng:

max_students - number_of_registration

Cách làm này giúp giảm dữ liệu truyền qua mạng vì các máy trạm chỉ trả về các lớp còn chỗ, không trả về toàn bộ danh sách lớp học phần.



```
SQLQuery1.sq...i (sa (103))*
67      JOIN DkiTinChi_HCM.dbo.subject sub
68          ON c.ID_subject COLLATE DATABASE_DEFAULT = sub.ID_subject COLLATE DATABASE_DEFAULT
69      JOIN DkiTinChi_HCM.dbo.teacher te
70          ON c.ID_teacher COLLATE DATABASE_DEFAULT = te.ID_teacher COLLATE DATABASE_DEFAULT
71      JOIN DkiTinChi_HCM.dbo.department d
72          ON te.ID_department COLLATE DATABASE_DEFAULT = d.ID_department COLLATE DATABASE_DEFAULT
73      WHERE c.class_status COLLATE DATABASE_DEFAULT = ''OPEN''
74        AND c.number_of_registration < c.max_students
75        AND d.ID_headquarter COLLATE DATABASE_DEFAULT = ''HQHCM''
76      '
77  )
78  )
79  SELECT
80  site_name AS [Cơ sở],
81  ID_class AS [Mã lớp học phần],
82  name_subject AS [Tên học phần],
83  max_students AS [Sĩ số tối đa],
84  number_of_registration AS [Đã đăng ký],
85  max_students - number_of_registration AS [Số chỗ còn lại]
86  FROM q
87  ORDER BY site_name, [Số chỗ còn lại] DESC;
```

133 % | No issues found | Ln: 87, Ch: 44 | (2908 chars, 87 lines) | SPC | CRLF | UTF-8

Results | Messages

	Cơ sở	Mã lớp học phần	Tên học phần	Sĩ số tối đa	Đã đăng ký	Số chỗ còn lại
1	Hà Đông	L01_BUS1301	Marketing căn bản	60	4	56
2	Hà Đông	L01_INT1306	Cơ sở dữ liệu	60	6	54
3	Hà Đông	L01_BUS1401	Thương mại điện tử	60	8	52
4	Hà Đông	L01_INT1316	Lập trình Java	50	0	50
5	Hà Đông	L01_INT1332	Lập trình hướng đối tượng	50	0	50
6	Hà Đông	L01_INT1313	Cơ sở dữ liệu phân tán	50	6	44
7	Hà Đông	L01_ELE1301	Cơ sở điện tử viễn thông	45	4	41
8	Hà Đông	L01_NET1301	Mạng máy tính	45	4	41
9	Hà Đông	L01_SEC1301	An toàn thông tin	45	4	41
10	Hà Đông	L01_SEC1401	Mật mã học	40	0	40
11	Hà Đông	L01_INT1358	Lập trình Web	45	6	39

Query executed successfully. | (local)\CSDLPT_NHOM7 (17.0 ... | sa (103) | DkyTinChi | 00:00:00 | Row: 1, Col: 1 | 36 rows

Nhận xét: Truy vấn đã lấy danh sách các lớp học phần còn chỗ từ ba site Hà Đông, Hòa Lạc và TP. Hồ Chí Minh. Mỗi site tự lọc dữ liệu theo cơ sở, trạng thái lớp và điều kiện còn chỗ trước khi trả kết quả về máy chủ trung tâm. Kết quả truy vấn hỗ trợ nghiệp vụ đăng ký học phần vì sinh viên có thể biết lớp nào còn khả năng đăng ký.

Truy vấn 7: Thống kê khối lượng giảng dạy của giảng viên

1. Mục tiêu truy vấn

Truy vấn này dùng để thống kê số lớp học phần mà mỗi giảng viên phụ trách tại từng cơ sở. Kết quả giúp nhà trường đánh giá khối lượng giảng dạy của giảng viên và hỗ trợ phân bổ lớp học phần giữa các khoa, các cơ sở.

2. Dữ liệu sử dụng

Site Hà Đông:

- DkyTinChi.dbo.class
- DkyTinChi.dbo.teacher
- DkyTinChi.dbo.department
- DkyTinChi.dbo.headquarter

Site Hòa Lạc:

- CSDL_HL.dbo.class
- CSDL_HL.dbo.teacher
- CSDL_HL.dbo.department

- CSDL_HL.dbo.headquarter
- Truy vấn thông qua Linked Server LINK_HL

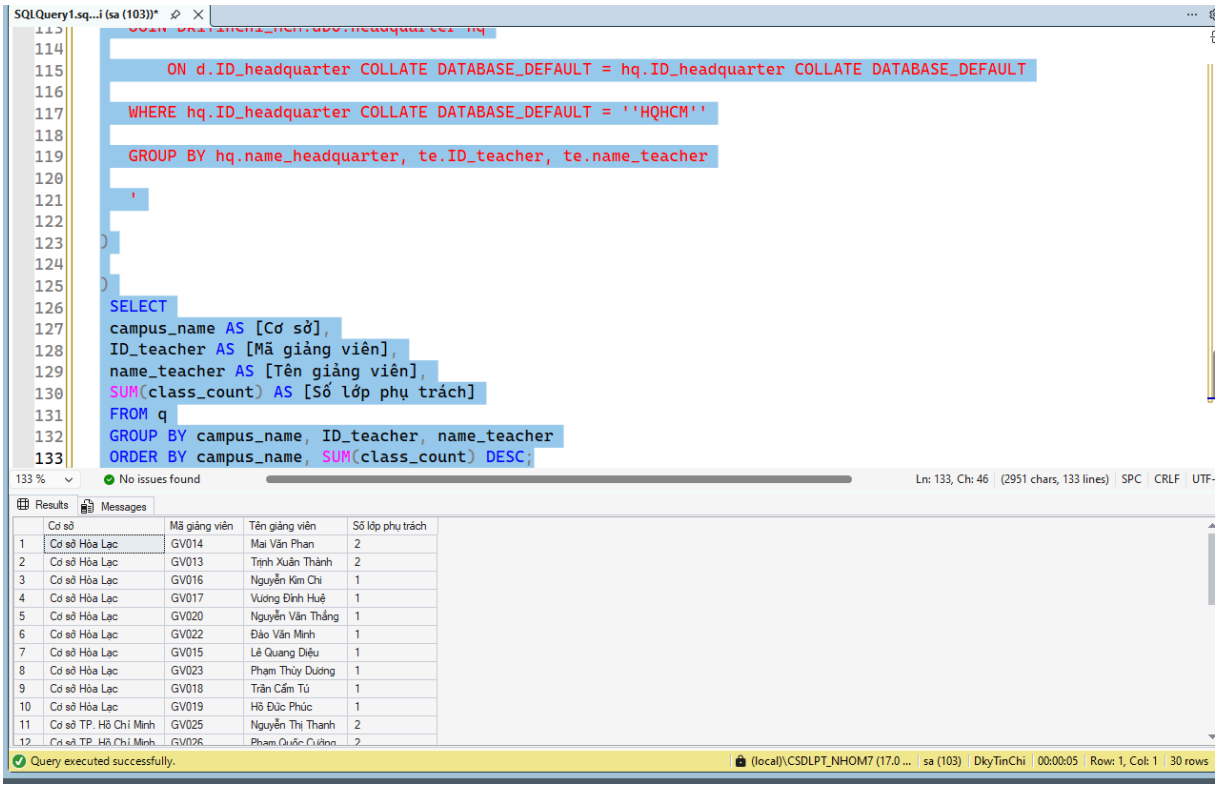
Site TP. Hồ Chí Minh:

- DkiTinChi_HCM.dbo.class
- DkiTinChi_HCM.dbo.teacher
- DkiTinChi_HCM.dbo.department
- DkiTinChi_HCM.dbo.headquarter
- Truy vấn thông qua Linked Server Link_HoChiMinh

3. Cách tổng hợp

Mỗi site tự thống kê số lớp học phần mà từng giảng viên phụ trách tại cơ sở của mình. Dữ liệu được nhóm theo cơ sở, mã giảng viên và tên giảng viên. Sau đó máy chủ trung tâm Hà Đông gom kết quả từ các site bằng UNION ALL.

Do mỗi site đã thực hiện GROUP BY và COUNT tại chỗ, máy chủ trung tâm chỉ nhận dữ liệu tổng hợp thay vì toàn bộ bảng lớp học phần. Điều này giúp giảm dữ liệu truyền qua mạng và tối ưu hiệu năng truy vấn phân tán.



Nhận xét**:** Truy vấn đã thống kê số lớp học phần mà mỗi giảng viên phụ trách tại từng cơ sở. Việc tổng hợp được thực hiện theo hướng tối ưu: mỗi site tự đếm số lớp của giảng viên, sau đó máy chủ trung tâm gom kết quả. Truy vấn này hỗ trợ nhà trường đánh giá khối lượng giảng dạy và phân bổ lớp học phần hợp lý hơn giữa các giảng viên, khoa và cơ sở.

PHẦN 6: XỬ LÝ ĐĂNG KÝ HỌC PHẦN ĐỒNG THỜI

I. Mục tiêu xử lý đồng thời

Trong hệ thống đăng ký học phần, nhiều sinh viên có thể đăng ký cùng một lớp học phần tại cùng thời điểm. Nếu lớp học phần có số chỗ giới hạn, các giao dịch đăng ký đồng thời có thể cùng tác động đến sĩ số của cùng một lớp học phần. Nếu không có cơ chế kiểm soát đồng thời, hệ thống có thể phát sinh lỗi vượt quá sĩ số tối đa hoặc cập nhật sai số lượng đăng ký.

Mục tiêu của phần này là mô phỏng tình huống nhiều sinh viên cùng đăng ký một lớp học phần còn ít chỗ tại cơ sở TP. Hồ Chí Minh, từ đó kiểm chứng cơ chế kiểm soát đồng thời của hệ thống.

Hệ thống cần đảm bảo:

- Mỗi giao dịch đăng ký được xử lý nhất quán.
- Số lượng đăng ký không vượt quá sĩ số tối đa của lớp học phần.
- Nếu đăng ký thất bại, toàn bộ thao tác phải được rollback.
- Không xảy ra lỗi Lost Update khi nhiều giao dịch cùng cập nhật sĩ số lớp học phần.

Trong hệ thống, nghiệp vụ đăng ký học phần được xử lý chính bằng Stored Procedure usp_RegisterClass. Procedure này sử dụng Transaction kết hợp với UPDLOCK và ROWLOCK khi cập nhật sĩ số lớp học phần. Nhờ đó, khi nhiều sinh viên cùng đăng ký một lớp có số chỗ giới hạn, hệ thống chỉ cho phép giao dịch hợp lệ được ghi nhận. Các giao dịch còn lại sẽ bị từ chối nếu lớp đã đủ sĩ số.

Demo xử lý đồng thời được thực hiện tại máy trạm TP. Hồ Chí Minh vì lớp học phần được chọn thuộc cơ sở TP. Hồ Chí Minh. Sau khi xử lý tại máy trạm, dữ liệu có thể được đồng bộ về máy chủ trung tâm thông qua Replication.

II. Tình huống tranh chấp dữ liệu

Tình huống được mô phỏng là nhiều sinh viên cùng đăng ký vào một lớp học phần có số chỗ giới hạn. Ví dụ, lớp học phần chỉ còn 1 chỗ trống nhưng có 2 sinh viên cùng đăng ký gần như cùng lúc.

Nếu không có cơ chế kiểm soát đồng thời, hai giao dịch có thể cùng đọc được trạng thái lớp còn chỗ và cùng thực hiện đăng ký. Khi đó, số lượng đăng ký thực tế có thể vượt quá max_students.

Tình huống demo:

- Site thực hiện: TP. Hồ Chí Minh.
- Cơ sở dữ liệu: DkiTinChi_HCM.

- Lớp học phần được chọn: L03_NET1301.
- Cơ sở lớp học phần: HQHCM.
- Sĩ số tối đa được đặt bằng số lượng đăng ký hiện tại cộng thêm 1.
- Hai sinh viên test cùng đăng ký lớp này.
- Kết quả mong đợi: chỉ một sinh viên đăng ký thành công, sinh viên còn lại bị từ chối vì lớp đã đủ sĩ số.

Nếu lớp L03_NET1301 không có trong dữ liệu hiện tại, nhóm chọn một lớp khác thuộc TP. Hồ Chí Minh và thay mã lớp trong các câu lệnh SQL.

III. Các lỗi dữ liệu có thể xảy ra

1. Vượt quá sĩ số lớp học phần

Nếu nhiều giao dịch cùng đăng ký thành công trong khi lớp chỉ còn ít chỗ, số lượng đăng ký có thể lớn hơn max_students.

2. Đăng ký trùng lớp học phần

Một sinh viên có thể cố gắng đăng ký cùng một lớp học phần nhiều lần. Procedure cần kiểm tra để không cho sinh viên có nhiều bản ghi REGISTERED cho cùng một lớp học phần.

3. Trùng lịch học

Sinh viên có thể đăng ký hai lớp học phần có cùng ngày học và ca học. Hệ thống cần kiểm tra bảng session để tránh sinh viên bị trùng lịch học.

4. Lost Update

Hai giao dịch cùng cập nhật number_of_registration của một lớp học phần. Nếu không khóa dữ liệu đúng cách, một cập nhật có thể ghi đè hoặc làm sai kết quả cập nhật của giao dịch khác.

5. Dữ liệu không nhất quán giữa các site

Trong môi trường phân tán, dữ liệu phát sinh tại máy trạm TP. Hồ Chí Minh cần được đồng bộ về máy chủ trung tâm. Nếu giao dịch lỗi hoặc đồng bộ không đúng, dữ liệu giữa máy trạm và máy chủ có thể không nhất quán.

IV. Cơ chế kiểm soát đồng thời

1. Sử dụng Transaction

Toàn bộ quá trình đăng ký học phần được đặt trong Transaction. Nếu các điều kiện đăng ký hợp lệ, hệ thống cập nhật sĩ số lớp học phần và thêm bản ghi vào bảng registration. Nếu có lỗi, Transaction được rollback.

2. Sử dụng XACT_ABORT

Procedure sử dụng SET XACT_ABORT ON để đảm bảo khi phát sinh lỗi trong quá trình thực hiện giao dịch, SQL Server tự động đưa giao dịch vào trạng thái cần rollback. Điều này giúp tránh trường hợp giao dịch bị lỗi nhưng vẫn ghi dở dữ liệu.

3. Sử dụng UPDLOCK

Khi cập nhật sĩ số lớp học phần, hệ thống sử dụng UPDLOCK để khóa dòng dữ liệu lớp học phần đang được đăng ký. Giao dịch khác muốn cập nhật cùng dòng dữ liệu phải chờ giao dịch hiện tại hoàn tất.

4. Sử dụng ROWLOCK

ROWLOCK giúp định hướng SQL Server khóa ở mức dòng dữ liệu, hạn chế khóa phạm vi lớn hơn khi không cần thiết.

5. Rollback khi giao dịch thất bại

Nếu sinh viên không tồn tại, lớp học phần không hợp lệ, lớp đã đủ sĩ số, sinh viên đã đăng ký lớp này hoặc bị trùng lịch học, hệ thống rollback giao dịch để không ghi dữ liệu sai vào cơ sở dữ liệu.

V. Phân tích Procedure đăng ký học phần

Procedure usp_RegisterClass đóng vai trò là lõi nghiệp vụ trung tâm của toàn bộ hệ thống quản lý tín chỉ. Chức năng này tiếp nhận yêu cầu từ người dùng với các tham số cơ bản gồm: Mã đăng ký (tự sinh), Mã sinh viên, Mã lớp học phần, và Mã cơ sở (tùy chọn để phục vụ truy vấn phân tán).

Để đảm bảo nguyên tắc ACID (Tính nguyên tử, Nhất quán, Cô lập và Bền vững) trong cơ sở dữ liệu, toàn bộ luồng thực thi của thủ tục được đặt trong một khối giao dịch thống nhất (Transaction).

1. Luồng thực thi kiểm duyệt điều kiện (Validation Pipeline):

Trước khi tiến hành ghi nhận dữ liệu, hệ thống thực hiện kiểm tra tuần tự qua 4 chốt chặn logic:

- Bước 1 (Xác thực Sinh viên): Kiểm tra đối chiếu mã sinh viên có tồn tại trong hệ thống và trực thuộc đúng cơ sở đào tạo đang xét hay không.

- Bước 2 (Xác thực Lớp học phần): Rà soát cấu hình của lớp học phần, đảm bảo lớp đang ở trạng thái mở đăng ký (OPEN), thời điểm hiện tại nằm trong khung thời gian cho phép của học kỳ, và lớp được mở tại đúng cơ sở yêu cầu.
- Bước 3 (Kiểm tra trùng lặp giao dịch): Quét lịch sử hệ thống để đảm bảo sinh viên chưa từng đăng ký thành công chính lớp học phần này trước đó.
- Bước 4 (Kiểm tra xung đột thời khóa biểu): Kiểm tra lịch học có bị trùng với các lớp học phần đã đăng ký hay không

2. Cơ chế kiểm soát đồng thời và cập nhật sĩ số

Sau khi vượt qua các bước kiểm duyệt cá nhân, thuật toán bước vào giai đoạn quan trọng nhất: Cập nhật sĩ số lớp. Đây là nơi xảy ra nguy cơ tranh chấp dữ liệu cao nhất khi hàng trăm sinh viên cùng đăng ký một lớp.

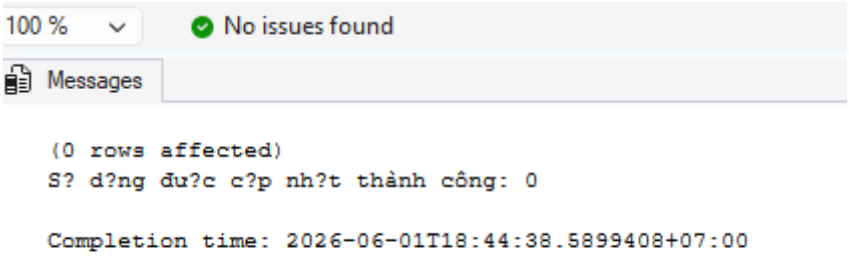
- Kỹ thuật Locking (Khóa dữ liệu): Thuật toán thực hiện yêu cầu tăng sĩ số lớp lên 1 đơn vị, đồng thời áp dụng cơ chế khóa cập nhật cấp dòng (UPDLOCK, ROWLOCK). Kỹ thuật này ép buộc các giao dịch đến cùng lúc phải "xếp hàng" xử lý tuần tự trên đúng dòng dữ liệu của lớp học đó, triệt tiêu hoàn toàn lỗi Lost Update (Cập nhật đè).
- Ràng buộc sức chứa : Lệnh tăng sĩ số đi kèm với một màng lọc điều kiện nghiêm ngặt: Sĩ số hiện tại của lớp bắt buộc phải nhỏ hơn Sĩ số tối đa (number_of_registration < max_students).
- Xử lý ngoại lệ "Hết chỗ": Nếu lớp đã đạt mức tối đa, màng lọc điều kiện trên sẽ làm cho lệnh cập nhật bị vô hiệu hóa (không có dòng dữ liệu nào được cập nhật - tương đương @@ROWCOUNT = 0). Thuật toán ngay lập tức nhận diện trạng thái này, phát ra thông báo lỗi: *"Lớp học phần đã đủ sĩ số"* và kích hoạt lệnh từ chối giao dịch. Nhờ đó, dù có nhiều sinh viên nhấn đăng ký cùng một mili-giây, hệ thống vẫn đảm bảo tuyệt đối không xảy ra tình trạng Over-allocation (Vượt quá sĩ số).

3. Ghi nhận và Hoàn tất giao dịch:

- Bước 5: Nếu giao dịch giữ chỗ ở Bước 4 thành công, hệ thống chính thức khởi tạo một bản ghi mới trong bảng lịch sử đăng ký (registration) với trạng thái là đã đăng ký (REGISTERED).
- Bước 6: Thuật toán gọi lệnh COMMIT để lưu vĩnh viễn các thay đổi và giải phóng khóa. Trường hợp có bất kỳ vi phạm nào từ Bước 1 đến Bước 5, khối CATCH sẽ bắt lỗi và gọi ROLLBACK để hủy bỏ toàn bộ phiên làm việc, đưa dữ liệu về trạng thái an toàn.

Lưu ý: Procedure này kiểm tra sinh viên đã đăng ký cùng một lớp hay chưa và kiểm tra trùng lịch học. Việc kiểm tra sinh viên đăng ký nhiều lớp khác nhau nhưng

cùng một học phần trong cùng học kỳ được kiểm soát bổ sung bằng Trigger kiểm tra đăng ký trùng học phần.



VI. Phân tích Procedure hủy đăng ký học phần

Procedure usp_CancelRegistration xử lý nghiệp vụ hủy đăng ký học phần. Khi hủy đăng ký, hệ thống cập nhật trạng thái đăng ký từ REGISTERED sang CANCELLED, ghi thời gian hủy và giảm sĩ số lớp học phần.

Procedure thực hiện các bước sau:

- Tìm bản ghi đăng ký đang ở trạng thái REGISTERED.
- Khóa bản ghi đăng ký bằng UPDLOCK và ROWLOCK.
- Cập nhật trạng thái đăng ký thành CANCELLED.
- Ghi thời gian hủy vào cancelled_at.
- Giảm number_of_registration của lớp học phần đi 1.
- Commit nếu giao dịch thành công.
- Rollback nếu xảy ra lỗi.

Procedure này đảm bảo việc hủy đăng ký và cập nhật sĩ số được thực hiện trong cùng một Transaction. Nhờ đó, hệ thống tránh trường hợp bản ghi đăng ký đã hủy nhưng sĩ số lớp chưa giảm, hoặc sĩ số giảm nhưng trạng thái đăng ký chưa được cập nhật.

VII. Demo xử lý đăng ký đồng thời

1. Môi trường và Mục tiêu kiểm thử

- Môi trường thực thi: Kịch bản được tiến hành hoàn toàn trên phân mảnh cục bộ tại máy trạm TP. Hồ Chí Minh (Cơ sở dữ liệu DkiTinChi_HCM).
- Mục tiêu cốt lõi: Kịch bản được thiết kế để chứng minh tính đúng đắn của cơ chế kiểm soát đồng thời. Cụ thể, bài kiểm tra sẽ đánh giá khả năng của hệ thống trong việc ngăn chặn hiện tượng "Cập nhật đè" (Lost Update) và đảm bảo số

lượng đăng ký thực tế tuyệt đối không vượt quá giới hạn phân bổ (Over-allocation) khi có nhiều người dùng tranh chấp một tài nguyên.

Cách thức thực hiện:

- Kiểm tra danh sách lớp học phần thuộc cơ sở TP. Hồ Chí Minh.
- Chọn một lớp để demo xử lý đồng thời.
- Trong bản mẫu này, lớp được chọn là L03_NET1301.

100 % No issues found

Results Messages

	ID_class	name_subject	ID_headquarter	max_students	number_of_registration	class_status
1	L03_BUS1301	Marketing căn bản	HQHCM	60	4	OPEN
2	L03_BUS1401	Thương mại điện tử	HQHCM	60	4	OPEN
3	L03_ELE1301	Cơ sở điện tử viễn thông	HQHCM	45	4	OPEN
4	L03_ELE1401	Internet of Things	HQHCM	40	4	OPEN
5	L03_INT1306	Cơ sở dữ liệu	HQHCM	60	6	OPEN
6	L03_INT1313	Cơ sở dữ liệu phân tán	HQHCM	50	6	OPEN
7	L03_INT1316	Lập trình Java	HQHCM	50	0	OPEN
8	L03_INT1332	Lập trình hướng đối tượng	HQHCM	50	0	OPEN
9	L03_INT1358	Lập trình Web	HQHCM	45	6	OPEN
10	L03_NET1301	Mạng máy tính	HQHCM	45	4	OPEN
11	L03_SEC1301	An toàn thông tin	HQHCM	45	4	OPEN
12	L03_SEC1401	Mật mã học	HQHCM	40	0	OPEN

2. Chuẩn bị lớp học phần có số chỗ giới hạn

Ý nghĩa chuẩn bị dữ liệu:

- Tạo 2 sinh viên test thuộc cơ sở TP. Hồ Chí Minh.
- Đảm bảo học kỳ của lớp đang trong thời gian đăng ký.
- Đặt lớp L03_NET1301 ở trạng thái OPEN.
- Đặt max_students = số lượng đăng ký hiện tại + 1 để lớp chỉ còn đúng 1 chỗ trống.

Ảnh tạo 2 sinh viên test.

Results Messages

	ID_student	name_student	ID_department	ID_curriculum
1	SV_CONC_HCM_001	Sinh viên đồng thời HCM 1	D_HCM_ATTT	CUR_ATTT
2	SV_CONC_HCM_002	Sinh viên đồng thời HCM 2	D_HCM_ATTT	CUR_ATTT

Ảnh lớp L03_NET1301 có trạng thái OPEN.

100 % No issues found

Results Messages

	ID_class	max_students	number_of_registration	so_cho_con_lai	class_status
1	L03_NET1301	45	4	41	OPEN

Chỉnh sửa number_of_registration lên 44 để chỉ còn 1 slot cho lớp

Ảnh lớp L03_NET1301 chỉ còn 1 chỗ trống.

Results Messages

	ID_class	max_students	number_of_registration	so_cho_con_lai	class_status
1	L03_NET1301	45	44	1	OPEN

3. Thực hiện đăng ký từ hai phiên truy vấn

- Khởi tạo 2 phiên làm việc hoàn toàn độc lập trên phần mềm quản trị CSDL, đại diện cho 2 sinh viên vừa tạo. Cả 2 phiên này được thao tác để gọi thủ tục usp_RegisterClass nhắm vào cùng một lớp L03_NET1301 tại cùng một thời điểm .
- Kết quả nghiệm thu dựa trên logic hệ thống:
    - Nhờ cơ chế khóa dữ liệu cấp dòng (UPDLOCK, ROWLOCK) đã cài đặt trong Procedure, giao dịch nào đến Server trước (dù chỉ một phần ngàn giây) sẽ chiếm được khóa, ghi nhận đăng ký thành công, đồng thời lấp đầy vị trí cuối cùng của lớp học.
    - Giao dịch đến sau bị ép vào trạng thái chờ. Ngay khi giao dịch 1 thành công thì sẽ nhả khóa, giao dịch 2 được hệ thống giải phóng để đọc lại dữ liệu. Lúc này, thuật toán kiểm tra nếu phát hiện điều kiện "sĩ số hiện tại < sĩ số tối đa" không còn thỏa mãn thì hệ thống tự động từ chối giao dịch, Rollback dữ liệu và ném ra thông báo ngoại lệ: *"Đăng ký thất bại: Lớp học phần đã đủ sĩ số"*.

Kết quả mong đợi:

- Một sinh viên đăng ký thành công.

```sql
USE DkiTinChi_HCM;
GO

EXEC usp_RegisterClass
   @ID_registration = 'REG_CONC_HCM_001',
   @ID_student = 'SV_CONC_HCM_001',
   @ID_class = 'L03_NET1301',
   @ID_headquarter = 'HQHCM';
GO
```

```
Commands completed successfully.

Completion time: 2026-06-01T19:10:38.4706601+07:00
```

- Một sinh viên đăng ký thất bại với thông báo: Lớp học phần đã đủ sĩ số.

```sql
USE DkiTinChi_HCM;
GO

EXEC usp_RegisterClass
   @ID_registration = 'REG_CONC_HCM_002',
   @ID_student = 'SV_CONC_HCM_002',
   @ID_class = 'L03_NET1301',
   @ID_headquarter = 'HQHCM';
GO
```

```
Msg 50005, Level 16, State 1, Procedure usp_RegisterClass, Line 100 [Batch Start Line 2]
Lớp học phần đã đủ sĩ số.

Completion time: 2026-06-01T19:11:54.4786324+07:00
```

4. Kiểm tra kết quả sau khi đăng ký tại máy trạm TP. Hồ Chí Minh

- Mục đích: Đánh giá trực tiếp trạng thái lưu trữ của cơ sở dữ liệu tại phân mảnh cục bộ ngay sau khi xảy ra hiện tượng tranh chấp giao dịch.
- Logic kiểm chứng:
  - *Kiểm tra lịch sử giao dịch:* Truy xuất bảng registration để quét và đối chiếu trạng thái cuối cùng của hai mã giao dịch REG_CONC_HCM_001 và REG_CONC_HCM_002 vừa được đẩy vào hệ thống.
  - *Kiểm tra giới hạn tài nguyên:* Tiếp tục truy xuất bảng class đối với lớp học phần mục tiêu (L03_NET1301). Hệ thống tự động tính toán đối

chiếu giữa thuộc tính Sĩ số hiện tại (number_of_registration) và Sĩ số tối đa (max_students) để xác định chính xác số chỗ còn lại sau đợt đăng ký.

- Kết quả mong đợi: Dữ liệu xuất ra chứng minh chỉ có một giao dịch được ghi nhận trạng thái 'REGISTERED'. Đồng thời, tham số sĩ số của lớp học phần đạt đúng ngưỡng trần tối đa, tuyệt đối không bị vượt mức (số chỗ còn lại = 0). Điều này khẳng định cơ chế Locking đã bảo vệ thành công sự nguyên vẹn của dữ liệu tại máy trạm.

Ảnh bảng registration sau demo.

Results | Messages

	ID_registration	ID_student	ID_class	registered_at	registration_status
1	REG_CONC_HCM_001	SV_CONC_HCM_001	L03_NET1301	2026-06-01 19:10:38.420	REGISTERED

Ảnh bảng [class] cho thấy number_of_registration không vượt max_students.

	ID_class	max_students	number_of_registration	so_cho_con_lai
1	L03_NET1301	45	45	0

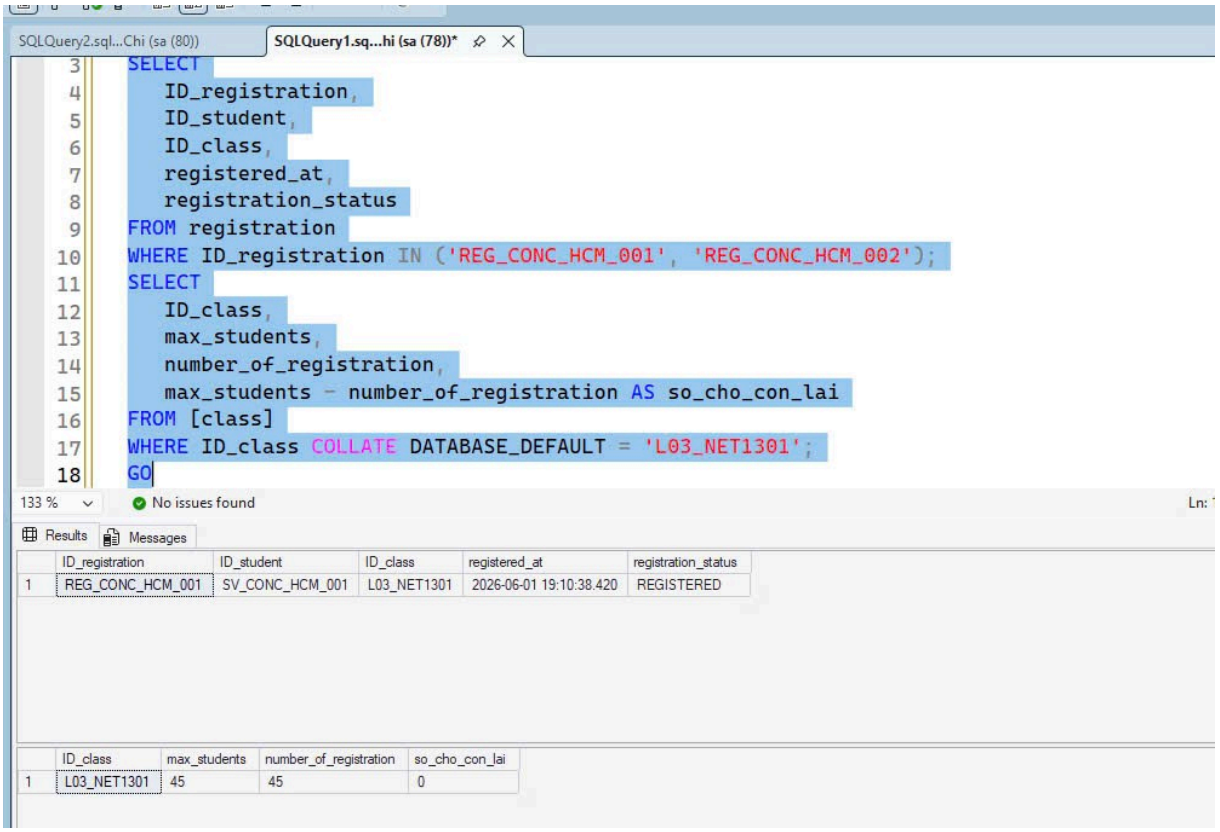
6. Kiểm tra dữ liệu đồng bộ về máy chủ trung tâm

- Sau một khoảng thời gian trễ nhất định để tiến trình nhân bản dữ liệu (Replication) hoàn tất chu kỳ đồng bộ, quản trị viên chuyển hướng kiểm tra sang cơ sở dữ liệu tổng (DkyTinChi) đặt tại máy chủ Hà Đông.
- Thuật toán lặp lại chính xác các phép truy xuất lịch sử giao dịch và rà soát trạng thái sĩ số của lớp học phần mục tiêu L03_NET1301 giống như đã thực hiện ở trên .

Ý nghĩa:

- Kiểm tra dữ liệu đăng ký phát sinh tại máy trạm TP. Hồ Chí Minh đã được đồng bộ về máy chủ trung tâm.
- Chứng minh hệ thống vừa xử lý đúng giao dịch tại site cục bộ, vừa có khả năng đồng bộ dữ liệu phục vụ quản lý toàn trường.

Ảnh kiểm tra bản ghi đăng ký và sĩ số lớp tại máy chủ Hà Đông sau khi đồng bộ.



Đánh giá và kết luận: Kết quả trả về tại máy chủ trung tâm khớp hoàn toàn với dữ liệu đã được ghi nhận tại máy trạm TP. Hồ Chí Minh. Điều này chứng minh được sức mạnh toàn diện của hệ thống: Nó không chỉ xử lý chính xác các tranh chấp giao dịch đa luồng tại site cục bộ , mà còn đáp ứng xuất sắc khả năng đồng bộ hóa dữ liệu, phục vụ cho công tác quản lý, thống kê và báo cáo toàn trường.

VIII. Đánh giá cơ chế xử lý đồng thời

Qua thực nghiệm mô phỏng, cơ chế kiểm soát đồng thời của hệ thống đã chứng minh được tính hiệu quả và đáp ứng tốt các yêu cầu nghiệp vụ:

- Đảm bảo toàn vẹn dữ liệu (Transaction): Các thao tác đăng ký/hủy đăng ký được đặt trong một khối giao dịch thống nhất. Nếu có lỗi, hệ thống lập tức Rollback, loại trừ hoàn toàn tình trạng rác dữ liệu hay cập nhật dở dang.
- Kiểm soát tranh chấp (UPDLOCK & ROWLOCK): Khóa cập nhật cấp dòng ép các giao dịch đồng thời phải "xếp hàng" chờ đọc dữ liệu một cách tuần tự. Chỉ giao dịch nào cập nhật sĩ số thành công khi lớp thực sự còn chỗ mới được đi tiếp.
- Triệt tiêu lỗi Lost Update và Over-allocation: Kết quả Demo cho thấy hệ thống đã chặn đứng hoàn toàn hiện tượng cập nhật đè (Lost Update) và đảm bảo sĩ số đăng ký tuyệt đối không bao giờ vượt quá ngưỡng max_students.

- Tương thích với mô hình phân tán (Replication): Các giao dịch được xử lý độc lập, nhanh chóng tại site cục bộ (máy trạm), sau đó dữ liệu mới được đồng bộ an toàn về máy chủ trung tâm, đảm bảo tính nhất quán cho toàn bộ hệ thống.

Kết luận: Phần xử lý đăng ký đồng thời đã giải quyết trọn vẹn yêu cầu của đề tài. Hệ thống áp dụng thành công các kỹ thuật nâng cao của SQL Server (Transaction, Locking) để xử lý triệt để tình huống tranh chấp dữ liệu trong môi trường phân tán.

PHẦN 7: DEMO CHƯƠNG TRÌNH

1. Công nghệ sử dụng

Project sử dụng React + Vite để xây dựng giao diện người dùng. Frontend có nhiệm vụ hiển thị các màn hình đăng nhập, đăng ký học phần, danh sách học phần đã đăng ký, thời khóa biểu và giao diện quản trị.

Backend được xây dựng bằng Node.js + Express.js, dùng để tiếp nhận request từ frontend, xác thực người dùng, phân quyền và gọi dữ liệu từ SQL Server. Backend cũng hỗ trợ điều hướng request giữa các node khi cần demo đăng ký học phần khác cơ sở.

Cơ sở dữ liệu sử dụng Microsoft SQL Server. Project được triển khai theo mô hình nhiều node gồm HQHD, HQHL và HQHCM để mô phỏng hệ thống cơ sở dữ liệu phân tán. Trong đó, HQHD đóng vai trò trung tâm/gateway, còn HQHL và HQHCM là các cơ sở chi nhánh.

Project này không phải hệ thống quản lý đào tạo hoàn chỉnh, mà chỉ xây dựng một phần chức năng để demo cho môn Cơ sở dữ liệu phân tán, tập trung vào luồng đăng ký học phần, đăng ký khác cơ sở và hiển thị dữ liệu theo từng node.

2. Giao diện

Danh sách đăng ký | Thời khóa biểu

Quản lý đăng ký học phần

Chọn cơ sở: Hòa Lạc

Danh sách môn học mở cho đăng ký (Hòa Lạc)

Mã MH	Tên môn học	Nhóm	Giảng viên	Số TC	Số lượng	Còn lại	Thời khóa biểu	Đăng ký
L02_BUS1301	Marketing căn bản	2	Đào Văn Minh	2	4	56	Thứ 2, 2/3/2026, 13:00 - 15:00, phòng B201	Đăng ký
TEST_DK_CHEO_01	Marketing căn bản	1	Nguyễn Kim Chi	2	1	0	-	Hết
L02_BUS1401	Thương mại điện tử	2	Phạm Thùy Dương	3	8	52	Thứ 3, 3/3/2026, 15:15 - 17:15, phòng C301	Đăng ký
L02_ELE1301	Cơ sở điện tử viễn thông	2	Hồ Đức Phúc	3	6	39	Thứ 5, 26/2/2026, 07:00 - 09:00, phòng A201	Đăng ký
L02_ELE1401	Internet of Things	2	Nguyễn Văn Thắng	3	4	36	Thứ 6, 27/2/2026, 09:15 - 11:15, phòng B101	Đăng ký
L02_INT1306	Cơ sở dữ liệu	2	Trịnh Xuân Thành	3	6	54	Thứ 2, 16/2/2026, 07:00 - 09:00, phòng A101	Đăng ký
L02_INT1313	Cơ sở dữ liệu phân tán	2	Mai Văn Phan	3	7	43	Thứ 3, 17/2/2026, 09:15 - 11:15, phòng A102	Đăng ký
L02_INT1316	Lập trình Java	2	Mai Văn Phan	3	0	50	Thứ 6, 20/2/2026, 07:00 - 09:00, phòng B201	Đăng ký

Danh sách môn học đã đăng ký

Xóa	Mã lớp	Tên môn học	Giảng viên	Số TC	Ngày đăng ký	Trạng thái
×	TEST_DK_CHEO_01	Marketing căn bản	Nguyễn Kim Chi	2	3/6/2026	Đã đăng ký
	TEST_DK_CHEO_01	Marketing căn bản	Nguyễn Kim Chi	2	3/6/2026	Đã hủy
×	L02_SEC1301	An toàn thông tin	Nguyễn Kim Chi	3	31/5/2026	Đã đăng ký
×	L02_NET1301	Mạng máy tính	Vương Đình Huệ	3	31/5/2026	Đã đăng ký
×	L03_BUS1301	Marketing căn bản	Hà Minh Phương	2	2/6/2026	Đã đăng ký

Danh sách đăng ký | **Thời khóa biểu**

Thời khóa biểu môn học

Tổng số tiết học: **6**

Thứ	Tiết	Môn học	Giảng viên	Phòng	Ngày học	Ghi chú
Thứ 2	09:15 - 11:15	**An toàn thông tin** ca 2 (Mã lớp: L01_SEC1301)	Phạm Minh Đức	**C301**	23/2/2026	*Lý thuyết: Tổng quan toàn thông tin và ngu cơ tấn công*
Thứ 3	09:15 - 11:15	**Cơ sở dữ liệu phân tán** ca 2 (Mã lớp: L01_INT1313)	Lê Thị Mai	**A102**	17/2/2026	*Lý thuyết: Tổng quan sở dữ liệu phân tán*
Thứ 4	13:00 - 15:00	**Lập trình Web** ca 3 (Mã lớp: L02_INT1358)	Lê Quang Diệu	**A201**	18/2/2026	-
	15:15 - 17:15	**Mật mã học** ca 4 (Mã lớp: L02_SEC1401)	Trần Cẩm Tú	**A102**	25/2/2026	-
Thứ 5	15:15 - 17:15	**Lập trình hướng đối tượng** ca 4 (Mã lớp: L02_INT1332)	Trịnh Xuân Thành	**B101**	19/2/2026	-
	07:00 - 09:00	**Cơ sở điện tử viễn thông** ca 1 (Mã lớp: L02_ELE1301)	Hồ Đức Phúc	**A201**	26/2/2026	-

Tổng quan giảng viên

Tải lại

Lớp phụ trách, lịch dạy và sĩ số hiện tại tại cơ sở HQHD

Nguyễn Văn Thành
Mã GV: GV001
Học vị: Tiến sĩ

Công nghệ thông tin 1
Mã khoa: D_HD_CNTT
SĐT: 0902000001

Thống kê nhanh

2
Lớp phụ trách
2 lớp đang mở

2
Buổi dạy
Theo lịch học đã xếp

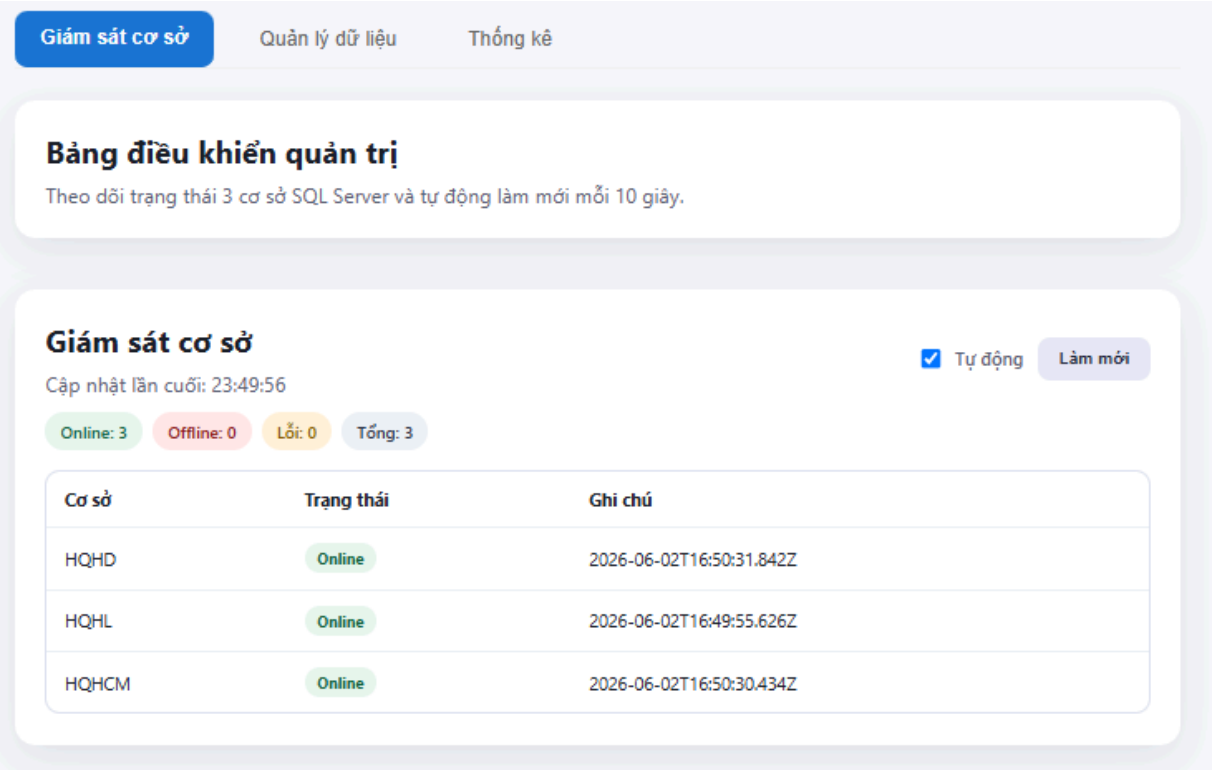
6
Sinh viên đăng ký
Tối đa 110

Lớp học phần phụ trách

Mã lớp	Học phần	Học kỳ	Nhóm	Sĩ số	Buổi	Trạng thái
L01_INT1332	Lập trình hướng đối tượng INT1332 - 3 tín chỉ	Học kỳ 2 năm học 2025-2026	1	0 / 50	1	Đang mở
L01_INT1306	Cơ sở dữ liệu INT1306 - 3 tín chỉ	Học kỳ 1 năm học 2025-2026	1	6 / 60	1	Đang mở

Lịch dạy

Ngày	Ca	Lớp	Học phần	Phòng	Ghi chú
16/2/2026 2	07:00 - 09:00 Ca 1	L01_INT1306	Cơ sở dữ liệu	A101 Trụ sở chính Hà Đông	Lý thuyết: Tổng quan cơ sở dữ liệu và mô hình quan hệ
19/2/2026 5	15:15 - 17:15 Ca 4	L01_INT1332	Lập trình hướng đối tượng	B101 Trụ sở chính Hà Đông	Lý thuyết: Lớp, đối tượng và tính đóng gói


Giám sát cơ sở
Quản lý dữ liệu
Thống kê
Bảng điều khiển quản trị
Theo dõi trạng thái 3 cơ sở SQL Server và tự động làm mới mỗi 10 giây.
Giám sát cơ sở
Cập nhật lần cuối: 23:49:56
Tự động
Làm mới
Online: 3
Offline: 0
Lỗi: 0
Tổng: 3
Cơ sở
Trạng thái
Ghi chú
HQHD
Online
2026-06-02T16:50:31.842Z
HQHL
Online
2026-06-02T16:49:55.626Z
HQHCM
Online
2026-06-02T16:50:30.434Z

Giám sát cơ sở | **Quản lý dữ liệu** | Thống kê

Quản lý dữ liệu

Quản lý thông tin theo bảng dữ liệu và cơ sở.

Bảng dữ liệu: Sinh viên

Cơ sở: Hà Đông

- Cơ sở đào tạo
- Phòng ban
- Chương trình đào tạo
- Sinh viên
- Giảng viên
- Học phần
- Lớp học phần
- Phòng học
- Lịch học
- Học kỳ
- Khung giờ

Tìm nhanh trong bảng...

Tổng: 57

		ày sinh	Giới tính	Địa chỉ SV	Số điện thoại	Năm nhập học	Mã khoa	Mã chương trình	Thao tác
		12/2004	Nam	TP. Hồ Chí Minh	0910000002	2022	D_HCM_ATTT	CUR_ATTT	Sửa Xóa
	Hân	/2004	Nữ	TP. Hồ Chí Minh	0910000003	2022	D_HCM_ATTT	CUR_ATTT	Sửa Xóa
B22ATTT003	Phạm Thu Trang	15/11/2004	Nữ	Hà Đông, Hà Nội	0910000020	2022	D_HD_ATTT	CUR_ATTT	Sửa Xóa
B22ATTT004	Võ Văn Kiệt	15/3/2004	Nam	Hà Đông, Hà Nội	0910000022	2022	D_HD_ATTT	CUR_ATTT	Sửa Xóa
B22ATTT005	Bùi Đức Mạnh	20/12/2004	Nam	Hòa Lạc, Hà Nội	0910000037	2022	D_HL_ATTT	CUR_ATTT	Sửa Xóa
B22ATTT006	Nguyễn Thảo	3/3/2004	Nữ	Hòa Lạc, Hà	0910000039	2022	D_HL_ATTT	CUR_ATTT	Sửa

Thêm mới dữ liệu

Mã sinh viên

Tên sinh viên

Ngày sinh

mm/dd/yyyy

Giới tính

Địa chỉ SV

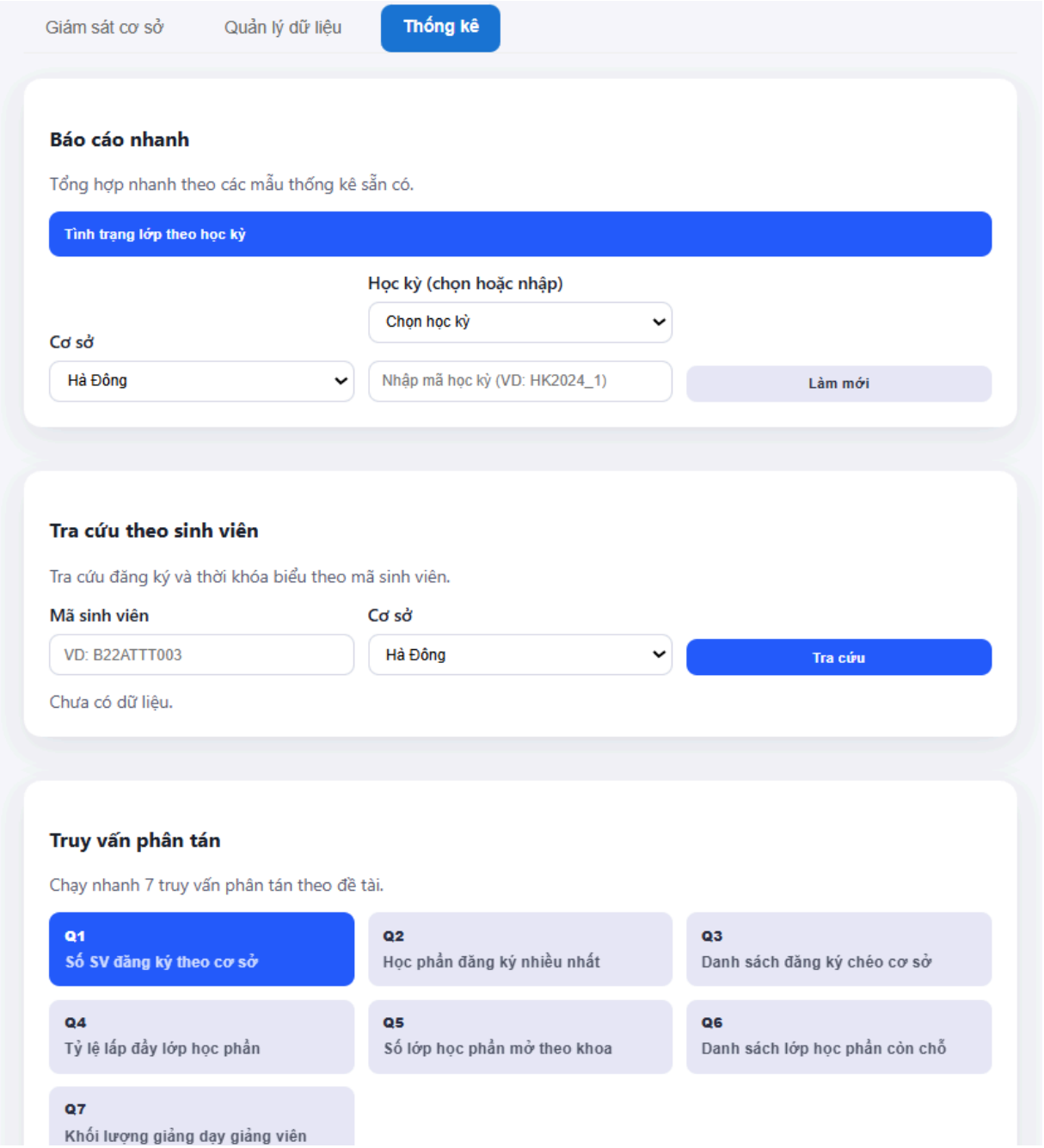
Số điện thoại

Năm nhập học

Mã khoa

Mã chương trình

Thêm mới



PHẦN 8: ĐÁNH GIÁ, NÂNG CAO VÀ TỔNG KẾT

I. Đối chiếu mức độ đáp ứng yêu cầu đề tài

Thông qua quá trình nghiên cứu và triển khai thực tế đề tài "Hệ thống đăng ký học phần nhiều cơ sở", hệ thống đã hoàn thành xuất sắc và đáp ứng trọn vẹn các mục tiêu, yêu cầu kỹ thuật cũng như nghiệp vụ được đặt ra ban đầu:

- Hệ thống đã xây dựng thành công mô hình cơ sở dữ liệu phân tán hoàn chỉnh gồm một máy chủ trung tâm đặt tại trụ sở chính Hà Đông và hai máy trạm chi nhánh đặt tại các cơ sở đào tạo Hòa Lạc và TP. Hồ Chí Minh.

- Giải quyết triệt để bài toán phân tải dữ liệu nghiệp vụ nhờ cơ chế xử lý cục bộ tại từng Server Chi nhánh, giúp giảm áp lực truy cập và loại bỏ hiện tượng nghẽn cổ chai tại máy chủ trung tâm trong các kỳ đăng ký tín chỉ cao điểm.

II. Đánh giá thiết kế cơ sở dữ liệu

- Cơ sở dữ liệu được thiết kế theo các nhóm dữ liệu chính gồm dữ liệu tổ chức, dữ liệu đào tạo, dữ liệu lớp học phần, dữ liệu thời khóa biểu và dữ liệu đăng ký học phần. Các bảng có khóa chính, khóa ngoại rõ ràng, phản ánh đúng quan hệ giữa sinh viên, giảng viên, học phần, lớp học phần, buổi học và đăng ký học phần.

- Thiết kế này giúp đảm bảo tính toàn vẹn dữ liệu. Ví dụ, bảng registration liên kết giữa student và class, bảng class liên kết với subject, teacher và term, bảng session liên kết với class, room và timeslot. Nhờ đó, hệ thống có thể kiểm tra được trùng lịch học, trùng đăng ký, sĩ số lớp học phần và sức chứa phòng học.

- Tuy nhiên, do phạm vi bài tập lớn có giới hạn, hệ thống chưa triển khai đầy đủ một số nghiệp vụ thực tế như quản lý điểm, học phí, lịch sử học lại và xét điều kiện tiên quyết dựa trên kết quả học tập.

III. Đánh giá thiết kế phân tán

- Hệ thống sử dụng mô hình 1 Server Trung tâm và 2 Server Chi nhánh. Server Trung tâm đặt tại Hà Đông lưu trữ CSDL toàn cục và quản lý dữ liệu dùng chung. Hai Server Chi nhánh tại Hòa Lạc và TP. Hồ Chí Minh lưu trữ dữ liệu cục bộ theo phạm vi cơ sở.

- Dữ liệu nghiệp vụ như student, teacher, room, class, session và registration được phân mảnh theo cơ sở đào tạo. Cách thiết kế này giúp mỗi cơ sở xử lý dữ liệu cục bộ nhanh hơn, giảm tải cho Server Trung tâm và phù hợp với thực tế quản lý đào tạo nhiều cơ sở.

- Các dữ liệu dùng chung như subject, term, timeslot, curriculum, curriculum_subject và prerequisite được nhân bản xuống các máy trạm để phục vụ tra cứu và xử lý nghiệp vụ. Nhờ vậy, máy trạm không cần truy vấn liên tục về Server Trung tâm khi cần đọc dữ liệu danh mục.

IV. Đánh giá cơ chế Replication và đồng bộ dữ liệu

- Hệ thống sử dụng Transactional Replication cho các bảng dữ liệu dùng chung. Các bảng này được quản lý tập trung tại Server Trung tâm và nhân bản một chiều xuống các máy trạm. Cách này phù hợp vì dữ liệu danh mục có tần suất đọc cao nhưng tần suất cập nhật thấp.

- Đối với dữ liệu nghiệp vụ phân mảnh, hệ thống sử dụng Merge Replication để đồng bộ hai chiều giữa Server Trung tâm và các máy trạm. Khi dữ liệu phát sinh tại máy trạm, thay đổi sẽ được đồng bộ về Server Trung tâm. Khi Server Trung tâm cập nhật dữ liệu thuộc một cơ sở cụ thể, dữ liệu đó cũng có thể được đồng bộ xuống máy trạm tương ứng.

- Qua kiểm thử, hệ thống đã thực hiện được đồng bộ từ máy chủ xuống máy trạm và từ máy trạm lên máy chủ. Điều này giúp Server Trung tâm luôn nắm được dữ liệu tổng hợp toàn hệ thống, phục vụ báo cáo và thống kê toàn trường.

V. Đánh giá truy vấn phân tán

- Hệ thống đã xây dựng các truy vấn phân tán chạy tại Server Trung tâm Hà Đông. Dữ liệu cục bộ Hà Đông được truy vấn trực tiếp trên database DkyTinChi, còn dữ liệu tại Hòa Lạc và TP. Hồ Chí Minh được truy vấn thông qua Linked Server.

- Các truy vấn sử dụng OPENQUERY để đẩy điều kiện lọc và phép tổng hợp xuống từng site. Sau đó, Server Trung tâm gom kết quả bằng UNION ALL. Cách làm này giúp giảm lượng dữ liệu truyền qua mạng và phù hợp với mô hình cơ sở dữ liệu phân tán.

- Các truy vấn đã đáp ứng các yêu cầu như thống kê sinh viên đăng ký theo cơ sở, tìm học phần có nhiều lượt đăng ký nhất, liệt kê đăng ký chéo cơ sở, tính tỷ lệ lấp đầy lớp học phần, thống kê số lớp mở theo khoa và thống kê khối lượng giảng dạy của giảng viên.

VI. Đánh giá xử lý đăng ký học phần đồng thời

Nghiệp vụ đăng ký học phần đồng thời được xử lý bằng Stored Procedure kết hợp Transaction và cơ chế khóa dữ liệu. Khi nhiều sinh viên cùng đăng ký một lớp học phần còn ít chỗ, hệ thống khóa dòng dữ liệu của lớp học phần bằng UPDLOCK và ROWLOCK trong quá trình cập nhật sĩ số.

Điều kiện number_of_registration < max_students được đặt trực tiếp trong câu UPDATE. Vì vậy, chỉ giao dịch nào cập nhật sĩ số thành công mới được ghi nhận vào bảng registration. Nếu lớp đã đủ sĩ số, giao dịch đến sau sẽ bị từ chối và rollback.

Kết quả demo cho thấy khi hai sinh viên cùng đăng ký một lớp chỉ còn một chỗ, hệ thống chỉ cho phép một giao dịch thành công. Giao dịch còn lại bị từ chối do lớp đã đủ sĩ số. Điều này chứng minh hệ thống tránh được lỗi Lost Update và không làm vượt sĩ số lớp học phần.

VII. Đánh giá hiệu năng hệ thống

- Truy vấn cục bộ có tốc độ nhanh hơn vì dữ liệu nằm trực tiếp trên máy đang xử lý. Các thao tác như mở lớp, xếp lịch và đăng ký học phần cục bộ tại máy trạm có thể thực hiện nhanh, không phụ thuộc hoàn toàn vào Server Trung tâm.
- Truy vấn phân tán mất thêm thời gian do phải kết nối đến các site khác và truyền dữ liệu qua mạng. Tuy nhiên, việc sử dụng OPENQUERY giúp mỗi site tự lọc và tổng hợp dữ liệu trước khi trả kết quả về Server Trung tâm. Điều này giúp giảm lượng dữ liệu truyền qua mạng.
- Thời gian đồng bộ dữ liệu phụ thuộc vào tốc độ mạng, trạng thái SQL Server Agent và cấu hình Replication. Trong môi trường demo qua Radmin VPN, thời gian đồng bộ có thể dao động do chất lượng kết nối giữa các máy.
- Khi số lượng site tăng, hệ thống có khả năng mở rộng tốt hơn vì dữ liệu được xử lý cục bộ tại từng cơ sở. Tuy nhiên, việc quản trị cũng phức tạp hơn do phải cấu hình thêm Replication, Linked Server, subscription và cơ chế kiểm tra đồng bộ cho từng site.

VIII. Nội dung nâng cao

1. Đánh giá ảnh hưởng của Replication đối với dữ liệu dùng chung

- Replication giúp các bảng dữ liệu dùng chung như subject, term, timeslot, curriculum và prerequisite được nhân bản xuống các máy trạm. Nhờ đó, máy trạm có thể tra cứu dữ liệu nhanh mà không cần truy vấn liên tục về Server Trung tâm.
- Tuy nhiên, dữ liệu dùng chung cần được quản lý tập trung tại Server Trung tâm để tránh xung đột. Các máy trạm chủ yếu sử dụng dữ liệu này ở chế độ đọc, không trực tiếp cập nhật.

2. Đề xuất cơ chế xử lý khi một site tạm thời mất kết nối

- Trong hệ thống cơ sở dữ liệu phân tán, việc một site tạm thời mất kết nối có thể xảy ra do lỗi mạng, máy chủ chi nhánh bị tắt, SQL Server Agent ngừng hoạt động hoặc kết nối VPN không ổn định. Để đảm bảo hệ thống vẫn vận hành an toàn, cần phân biệt rõ giữa nghiệp vụ cục bộ và nghiệp vụ cần dữ liệu toàn cục.

- Đối với các nghiệp vụ cục bộ tại máy trạm, hệ thống vẫn cho phép tiếp tục xử lý trên Local Database nếu dữ liệu cần thao tác thuộc phạm vi cơ sở đó. Ví dụ, cơ sở Hòa Lạc vẫn có thể quản lý sinh viên, giảng viên, phòng học, lớp học phần, buổi học và đăng ký học phần cục bộ của Hòa Lạc. Các thay đổi phát sinh trong thời gian mất kết nối được lưu tại cơ sở dữ liệu cục bộ. Khi kết nối được khôi phục, cơ chế Merge Replication sẽ đồng bộ các thay đổi này về Server Trung tâm Hà Đông.

- Đối với dữ liệu danh mục dùng chung như học phần, học kỳ, ca học, chương trình đào tạo và môn học tiên quyết, máy trạm có thể tiếp tục đọc dữ liệu đã

được nhân bản trước đó. Tuy nhiên, máy trạm không được tự ý cập nhật các dữ liệu dùng chung này để tránh xung đột và sai lệch dữ liệu giữa các site. Việc thêm, sửa, xóa dữ liệu danh mục vẫn được quản lý tập trung tại Server Trung tâm.

- Đối với các nghiệp vụ cần dữ liệu toàn cục, đặc biệt là đăng ký chéo cơ sở, hệ thống nên tạm thời từ chối xử lý hoặc chuyển yêu cầu sang trạng thái chờ. Lý do là đăng ký chéo cần kiểm tra dữ liệu của cả sinh viên, lớp học phần, thời khóa biểu và sĩ số lớp tại nhiều cơ sở. Nếu một site đang mất kết nối, việc xử lý ngay có thể dẫn đến kết quả không chính xác hoặc làm dữ liệu không nhất quán. Vì vậy, yêu cầu đăng ký chéo chỉ nên được thực hiện khi Server Trung tâm và các site liên quan kết nối ổn định.

- Sau khi site kết nối trở lại, quản trị viên cần kiểm tra trạng thái Replication, chạy lại tiến trình đồng bộ nếu cần và đối chiếu dữ liệu giữa máy trạm với Server Trung tâm. Các dữ liệu phát sinh trong thời gian mất kết nối sẽ được đồng bộ về máy chủ trung tâm. Nếu xảy ra xung đột dữ liệu, hệ thống ưu tiên dữ liệu hợp lệ theo thời điểm cập nhật, phạm vi sở hữu dữ liệu của từng cơ sở và quy tắc nghiệp vụ đã thiết lập.

- Cơ chế này giúp hệ thống vừa đảm bảo tính tự chủ cục bộ của từng cơ sở, vừa giữ được tính nhất quán toàn cục. Các nghiệp vụ nội bộ vẫn có thể tiếp tục hoạt động khi mất kết nối tạm thời, trong khi các nghiệp vụ nhạy cảm như đăng ký chéo được kiểm soát chặt chẽ để tránh sai lệch dữ liệu.

3. So sánh phân mảnh theo cơ sở và phân mảnh theo khoa

- Phân mảnh theo cơ sở phù hợp với bài toán đăng ký học phần nhiều cơ sở vì dữ liệu sinh viên, giảng viên, phòng học, lớp học phần và đăng ký học phần thường phát sinh theo địa điểm đào tạo. Cách này giúp mỗi cơ sở tự xử lý dữ liệu của mình nhanh hơn.

- Phân mảnh theo khoa phù hợp khi mỗi khoa quản lý dữ liệu độc lập. Tuy nhiên, trong hệ thống nhiều cơ sở, cùng một nhóm ngành hoặc khoa có thể xuất hiện ở nhiều địa điểm đào tạo khác nhau. Vì vậy, phân mảnh theo cơ sở là lựa chọn hợp lý hơn cho hệ thống này.

IX. Hạn chế của hệ thống

Mặc dù đạt được những kết quả rất khả quan, hệ thống vẫn không tránh khỏi một số hạn chế nhất định do giới hạn về mặt kiến thức thực tế và kinh nghiệm triển khai hạ tầng mạng diện rộng:

- Hệ thống mới được triển khai ở quy mô demo với 3 site gồm Hà Đông, Hòa Lạc và TP. Hồ Chí Minh. Một số nghiệp vụ thực tế như quản lý điểm, học phí,

lịch sử học lại, xét điều kiện tiên quyết theo điểm và phân quyền chi tiết theo vai trò chưa được triển khai đầy đủ.

- Việc đồng bộ dữ liệu còn phụ thuộc vào trạng thái kết nối mạng, SQL Server Agent và cấu hình Replication. Khi mạng chậm hoặc một site mất kết nối, dữ liệu có thể chưa được cập nhật ngay lập tức giữa các site.

- Đăng ký chéo cơ sở được xử lý theo hướng tập trung tại Server Trung tâm để đảm bảo toàn vẹn dữ liệu. Cách này phù hợp với phạm vi bài tập lớn, nhưng chưa triển khai đầy đủ mô hình giao dịch phân tán phức tạp giữa nhiều site độc lập.

X. Định hướng phát triển

- Trong tương lai, hệ thống có thể mở rộng thêm nhiều cơ sở đào tạo, bổ sung các nghiệp vụ như quản lý điểm, học phí, lịch thi, xét điều kiện tiên quyết theo kết quả học tập và quản lý học lại.

- Hệ thống cũng có thể phát triển thêm giao diện giám sát trạng thái Replication, cảnh báo khi một site mất kết nối, ghi nhận lịch sử đồng bộ và hỗ trợ xử lý xung đột dữ liệu tự động.

- Về hiệu năng, hệ thống có thể bổ sung cơ chế cache cho dữ liệu dùng chung, tối ưu chỉ mục cho các bảng registration, class, student và triển khai phân quyền chi tiết hơn cho sinh viên, giảng viên, giáo vụ và quản trị viên.

XI. Kết luận

Đề tài "Hệ thống đăng ký học phần nhiều cơ sở" đã hoàn thành toàn diện các mục tiêu nghiên cứu và ứng dụng thực tiễn. Quá trình thực hiện dự án này là một cơ hội quý báu giúp chúng em củng cố sâu sắc các kiến thức lý thuyết nền tảng về cơ sở dữ liệu phân tán; từ kỹ thuật phân mảnh ngang dữ liệu, cấu hình cơ chế nhân bản đồng bộ, tối ưu hóa câu lệnh truy vấn từ xa cho đến các phương pháp nâng cao để kiểm soát giao dịch và giải quyết triệt để tranh chấp dữ liệu đồng thời trong môi trường phân tán.

Mặc dù sản phẩm vẫn còn những điểm cần tiếp tục cải tiến do giới hạn về mặt kinh nghiệm thực tế, nhưng những tư duy phân tích hệ thống, kỹ năng thiết kế cơ sở dữ liệu và năng lực vận dụng lý thuyết vào giải quyết các bài toán phức tạp thu được từ dự án này sẽ là hành trang vô cùng quan trọng cho công việc của chúng em sau này. Chúng em xin chân thành cảm ơn Thầy hướng dẫn đã luôn theo sát, chỉ bảo và định hướng tận tình để nhóm có thể hoàn thành tốt báo cáo bài tập lớn này!