

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC ĐẠI NAM
KHOA CÔNG NGHỆ THÔNG TIN



**BÀI TẬP LỚN
HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU**

**Đề tài:
THIẾT KẾ VÀ TRIỂN KHAI CƠ SỞ DỮ LIỆU CHO
HỆ THỐNG QUẢN LÝ XE BUÝT LIÊN TỈNH**

Giảng viên hướng dẫn
Nhóm sinh viên thực hiện

: ThS. Nguyễn Ngọc Ân
: Nhóm 5

STT	Họ và tên	Mã sinh viên	Lớp
1	Trần Minh Nghĩa	1771020505	CNTT 17-09
2	Trần Đức Lương	1771020444	CNTT 17-09
3	Đoàn Mai Lan	1771020413	CNTT 17-09

Hà Nội, tháng 03 năm 2025

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC ĐẠI NAM
KHOA CÔNG NGHỆ THÔNG TIN



BÀI TẬP LỚN
HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU

Đề tài:

THIẾT KẾ VÀ TRIỂN KHAI CƠ SỞ DỮ LIỆU CHO
HỆ THỐNG QUẢN LÝ XE BUÝT LIÊN TỈNH

STT	Họ và tên	Mã sinh viên	Ngày sinh	Điểm	
				Băng chữ	Băng số
1	Trần Minh Nghĩa	1771020505	02/08/2002		
2	Trần Đức Lương	1771020444	21/06/2005		
3	Đoàn Mai Lan	1771020413	22/10/2005		

CÁN BỘ CHẤM THI 1

CÁN BỘ CHẤM THI 2

ThS. Nguyễn Ngọc Ân

Hà Nội, tháng 03 năm 2025

MỤC LỤC

MỤC LỤC HÌNH ẢNH	6
MỤC LỤC BẢNG	8
MỤC LỤC KHUNG	9
DANH MỤC TỪ VIẾT TẮT	11
CHƯƠNG 1:	12
GIỚI THIỆU ĐỀ TÀI	12
1.1. Xây dựng bối cảnh giả định.....	12
1.2. Mô tả bài toán	12
1.3. Yêu cầu cần đạt được.....	13
CHƯƠNG 2:	14
PHÂN TÍCH VÀ THIẾT KẾ CƠ SỞ DỮ LIỆU	14
2.1. Xây dựng mô hình thực thể quan hệ (Entity – Relationship Diagram – ERD)	14
2.1.1. Xác định các thực thể và các thuộc tính tương ứng.....	14
2.1.2. Xác định liên kết và kiểu liên kết giữa các thực thể.....	14
2.1.3. Xây dựng mô hình ERD	16
2.2. Xây dựng mô hình quan hệ.....	16
2.2.1. Các quy tắc chuyển hóa từ mô hình ERD sang mô hình quan hệ.....	16
2.2.2. Chuyển hóa từ mô hình ERD sang mô hình quan hệ.....	17
2.3. Xác định các ràng buộc	17
2.4. Xác định các phụ thuộc hàm và chuẩn hóa theo chuẩn 3NF	21
2.4.1. Chuẩn hóa bảng tbl.PHUONGTIEN	21
2.4.2. Chuẩn hóa bảng tbl.TUYENXE	21
2.4.3. Chuẩn hóa bảng tbl.TAIXE	21
2.4.4. Chuẩn hóa bảng tbl.TRAMXE	21
2.4.5. Chuẩn hóa bảng tbl.THANHPHO	22
2.4.6. Chuẩn hóa bảng tbl.CHINHANH	22
CHƯƠNG 3:	23
TRIỂN KHAI CƠ SỞ DỮ LIỆU	23
3.1. Tạo cơ sở dữ liệu	23

3.1.1. Tạo cơ sở dữ liệu.....	23
3.1.2. Tạo các bảng dữ liệu.....	23
3.1.3. Cài đặt PK và FK.....	23
3.2. Nhập dữ liệu cho các bảng.....	24
3.3. Truy vấn cơ bản (SELECT, INSERT, UPDATE, DELETE).....	25
3.3.1. Truy vấn SELECT	25
3.3.2. Truy vấn INSERT	26
3.3.3. Truy vấn UPDATE	27
3.3.4. Câu lệnh DELETE	28
3.4. Truy vấn nâng cao (INNER JOIN, GROUP BY, HAVING, SUBQUERY)	28
3.4.1. Truy vấn sử dụng INNER JOIN	28
3.4.2. Truy vấn sử dụng GROUP BY	30
3.4.3. Truy vấn sử dụng HAVING	31
3.4.4. Truy vấn sử dụng SUBQUERY	33
CHƯƠNG 4:	36
LẬP TRÌNH T-SQL	36
4.1. Tạo view cho CSDL	36
4.2. Tạo INDEX cần thiết cho các bảng	44
4.3. Tạo STORED PROCEDURE.....	46
4.3.1. STORED PROCEDURE không tham số.....	46
4.3.2. STORED PROCEDURE có tham số.....	48
4.3.3. STORED PROCEDURE có đầu ra	51
4.4. Tạo các FUNCTION	53
4.4.1. FUNCTION trả về kiểu vô hướng.....	53
4.4.2. FUNCTION trả về kiểu bảng	55
4.4.3. FUNCTION sử dụng biến bảng	57
4.5. Tạo các TRIGGER để kiểm soát dữ liệu	59
4.5.1. TRIGGER theo dõi cập nhật, thay đổi dữ liệu các bảng	59
4.5.2. Các TRIGGER khác	62
BẢO MẬT VÀ QUẢN TRỊ.....	66
5.1. Tạo người dùng.....	66

5.2. Phân cấp quyền truy cập của người dùng	66
5.3. Quản lý sao lưu và phục hồi dữ liệu	67
KẾT LUẬN	74
PHÂN CÔNG NHIỆM VỤ	76
ĐÁNH GIÁ NỘI BỘ.....	77
TÀI LIỆU THAM KHẢO.....	78

MỤC LỤC HÌNH ẢNH

Ảnh 1. Mô hình ERD.	16
Ảnh 2. Mô hình quan hệ.....	17
Ảnh 3. Kết quả thực hiện truy vấn SELECT.....	26
Ảnh 4. Số lượng bản ghi trong bảng TAIXE trước (25 bản ghi - ảnh trái) và sau (30 bản ghi - ảnh phải) khi thực hiện truy vấn INSERT.	27
Ảnh 5. Dữ liệu của tuyến 01 và tài xế có mã 'TX6' trước (2 ảnh trên) và sau (2 ảnh dưới) khi thực hiện truy vấn UPDATE.	28
Ảnh 6. Kết quả sau khi thực hiện truy xuất thông tin của phương tiện (ảnh trên), trạm dừng (ảnh dưới trái) và tài xế (ảnh dưới phải).	30
Ảnh 7. Kết quả sau khi thực hiện truy vấn sử dụng GROUP BY.	31
Ảnh 8. Kết quả sau khi thực hiện truy vấn sử dụng HAVING.	33
Ảnh 9. Kết quả sau khi thực hiện truy vấn sử dụng SUBQUERY.....	35
Ảnh 10. Kết quả truy vấn VIEW theo dõi chi tiết phương tiện.....	37
Ảnh 11. Kết quả truy vấn VIEW chi tiết tài xế.	38
Ảnh 12. Kết quả truy vấn VIEW chi tiết trạm xe.....	39
Ảnh 13. Kết quả truy vấn VIEW theo dõi tuổi tài xế.....	40
Ảnh 14. Kết quả thực hiện truy vấn VIEW theo dõi số năm còn có thể lao động của tài xế.	41
Ảnh 15. Kết quả thực hiện truy vấn VIEW niên hạn của phương tiện.	42
Ảnh 16. Kết quả sau khi thực hiện truy vấn VIEW theo dõi liên hệ của các chi nhánh.	43
Ảnh 17. Kết quả sau khi thực hiện truy vấn VIEW theo dõi tổng lượng xe mà mỗi chi nhánh quản lý.	44
Ảnh 18. Kết quả sau khi thực hiện truy vấn STORED PROCEDURE.....	48
Ảnh 19. Kết quả sai khi thực hiện STORED PROCEDURE có tham số.	51
Ảnh 20. Kết quả sau khi thực hiện truy vấn STORED PROCEDURE có đầu ra....	53
Ảnh 21. Kết quả sau khi thực hiện truy vấn FUNCTION trả về kiểu vô hướng.	55
Ảnh 22. Kết quả sau khi thực hiện truy vấn FUNCTION trả về kiểu bảng.	57
Ảnh 23. Kết quả sau khi thực hiện truy vấn FUNCTION sử dụng biến bảng.	59

Ảnh 24. Kết quả sau khi cài đặt TRIGGER thành công, toàn bộ lịch sử thay đổi dữ liệu sẽ được lưu vết tại bảng LICHUTHAYDOIDULIEU. 62

MỤC LỤC BẢNG

Bảng 1. Danh sách các ràng buộc.....	17
Bảng 2. Quy trình cài đặt backup tự động cho CSDL.....	67
Bảng 3. Quy trình backup thủ công cho thiết bị chạy hệ điều hành macOS.....	72
Bảng 4. Bảng phân công nhiệm vụ.	76
Bảng 5. Bảng đánh giá thái độ và tinh thần tham gia.....	77
Bảng 6. Bảng đánh giá chất lượng công việc.....	77

MỤC LỤC KHUNG

Khung 1. Truy vấn khởi tạo cơ sở dữ liệu.....	23
Khung 2. Cấu trúc cơ bản của truy vấn khởi tạo bảng trong CSDL.	23
Khung 3. Truy vấn khởi tạo bảng PHUONGTIEN.....	23
Khung 4. Cấu trúc truy vấn thêm PK và FK.	23
Khung 5. Truy vấn thêm PK và FK cho bảng PHUONGTIEN.	24
Khung 6. Cấu trúc cơ bản truy vấn nhập dữ liệu vào bảng.	24
Khung 7. Truy vấn nhập dữ liệu vào bảng PHUONGTIEN.	24
Khung 8. Truy vấn SELECT.....	25
Khung 9. Truy vấn thêm dữ liệu tài xế mới vào bảng TAIXE.....	26
Khung 10. Truy vấn UPDATE.....	27
Khung 11. Truy vấn DELETE.	28
Khung 12. Truy vấn sử dụng INNER JOIN.....	29
Khung 13. Truy vấn sử dụng GROUP BY.....	30
Khung 14. Truy vấn sử dụng HAVING.	32
Khung 15. Truy vấn sử dụng SUBQUERY.	33
Khung 16. Khởi tạo VIEW theo dõi chi tiết phương tiện.	36
Khung 17. Truy vấn VIEW theo dõi chi tiết phương tiện.....	37
Khung 18. Khởi tạo VIEW theo dõi chi tiết tài xế.....	37
Khung 19. Truy vấn VIEW chi tiết tài xế.	38
Khung 20. Khởi tạo VIEW chi tiết trạm xe.	38
Khung 21. Truy vấn VIEW chi tiết trạm xe.....	39
Khung 22. Khởi tạo VIEW theo dõi tuổi tài xế.....	39
Khung 23. Truy vấn VIEW theo dõi tuổi tài xế.	40
Khung 24. Khởi tạo VIEW theo dõi số năm còn có thể lao động của tài xế.....	40
Khung 25. Truy vấn VIEW theo dõi số năm còn có thể lao động của tài xế.....	41
Khung 26. Khởi tạo VIEW theo dõi niên hạn của phương tiện.	42
Khung 27. Truy vấn VIEW theo dõi niên hạn của phương tiện.....	42
Khung 28. Khởi tạo VIEW theo dõi dữ liệu liên hệ của các chi nhánh.	42
Khung 29. Truy vấn VIEW theo dõi dữ liệu liên hệ của các chi nhánh.....	43
Khung 30. Khởi tạo VIEW theo dõi tổng lượng xe mà mỗi chi nhánh quản lý.....	44

Khung 31. Truy vấn VIEW theo dõi tổng lượng xe mà mỗi chi nhánh quản lý.....	44
Khung 32. Khởi tạo chỉ mục (INDEX).....	44
Khung 33. Khởi tạo STORED PROCEDURE không tham số.....	46
Khung 34. Cấu trúc cơ bản truy vấn STORED PROCEDURE.....	47
Khung 35. Khởi tạo truy vấn STORED PROCEDURE có tham số.....	48
Khung 36. Truy vấn STORED PROCEDURE có tham số.....	50
Khung 37. Khởi tạo STORED PROCEDURE có đầu ra.....	51
Khung 38. Truy vấn STORED PROCEDURE có đầu ra.....	53
Khung 39. Khởi tạo FUNCTION trả về kiểu vô hướng.....	54
Khung 40. Truy vấn FUNCTION trả về kiểu vô hướng.....	54
Khung 41. Khởi tạo FUNCTION trả về kiểu bảng.....	55
Khung 42. Truy vấn FUNCTION trả về kiểu bảng.....	57
Khung 43. Khởi tạo FUNCTION sử dụng biến bảng.....	57
Khung 44. Truy vấn FUNCTION sử dụng biến bảng.....	58
Khung 45. Tạo bảng lưu lịch sử thay đổi dữ liệu.....	59
Khung 46. Cài đặt TRIGGER cho các bảng (Ví dụ: Cài đặt cho bảng PHUONGTIEN).....	59
Khung 47. Khởi tạo TRIGGER ngăn xóa tài xế khi còn dữ liệu phương tiện liên quan.....	62
Khung 48. Khởi tạo TRIGGER kiểm tra biến kiểm soát của phương tiện.....	63
Khung 49. Khởi tạo TRIGGER ngăn chặn thay đổi thông tin mã tuyến.....	63
Khung 50. Khởi tạo TRIGGER chặn trùng lặp số điện thoại cho tài xế và chi nhánh.	64
Khung 51. Khởi tạo TRIGGER kiểm tra số chỗ ngồi của phương tiện.....	65
Khung 52. Khởi tạo TRIGGER chặn trùng lặp vị trí của trạm xe.....	65
Khung 53. Khởi tạo người dùng trong CSDL.....	66
Khung 54. Cài đặt quyền cho từng đối tượng người dùng.....	67

DANH MỤC TỪ VIẾT TẮT

STT	Từ viết tắt	Từ đầy đủ
1	CSDL	Cơ sở dữ liệu
2	Transerco	Tổng công ty Vận tải Hà Nội
3	ERD	Entity – Relationship Diagram (Mô hình thực thể - quan hệ)
4	PK	Primary Key (Khóa chính)
5	1 - m	One to Many (Quan hệ một – nhiều)
6	1 - 1	One to One (Quan hệ một – một)
7	FK	Foreign Key (Khóa ngoại)
8	3NF	Third Normal Form
9	1NF	First Normal Form
10	2NF	Second Normal Form
11	SQL MS20	SQL Server Management Studio 20

CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI

1.1. Xây dựng bối cảnh giả định

Đặt tình huống bạn là chuyên viên quản trị cơ sở dữ liệu (CSDL) tại Tổng công ty Vận tải Hà Nội (Transerco). Bạn được giao nhiệm vụ thiết kế và triển khai xây dựng hệ CSDL cho hệ thống quản lý các tuyến xe cho tổng công ty.

1.2. Mô tả bài toán

Tổng công ty Vận tải Hà Nội (Transerco) là Doanh nghiệp Nhà nước trực thuộc Ủy ban nhân dân thành phố Hà Nội, được thành lập từ năm 2004. Hiện nay, Transerco là đơn vị hàng đầu trong lĩnh vực cung cấp dịch vụ vận tải hành khách công cộng bằng xe buýt, quản lý bến bãi và điểm đỗ xe công cộng tại Thủ đô Hà Nội. Với hơn 10.000 lao động đang làm việc tại 12 đơn vị trực thuộc và 5 công ty con, để tối ưu hóa công tác quản lý, Tổng công ty có nhu cầu thiết kế một hệ CSDL đáp ứng các yêu cầu cụ thể sau:

Thứ nhất, hệ thống sẽ quản lý thông tin về các tuyến xe buýt do Transerco vận hành. Mỗi tuyến xe có một mã tuyến và tên tuyến riêng biệt với hành trình bắt đầu từ một trạm khởi hành và kết thúc tại một trạm kết thúc. Trên lộ trình, xe buýt có thể dừng đón – trả khách tại nhiều trạm trung gian khác nhau. Thông tin về các trạm dừng được lưu trữ bao gồm số trạm, tên trạm và vị trí địa lý của trạm đó. Một tuyến xe có thể đi qua nhiều trạm và ngược lại, một trạm có thể phục vụ nhiều tuyến xe khác nhau.

Thứ hai, bên cạnh việc quản lý các tuyến xe buýt và trạm dừng, hệ thống đồng thời đảm nhiệm việc quản lý các thành phố mà các tuyến xe buýt đi qua. Mỗi thành phố có một mã số thành phố và tên thành phố riêng. Một tuyến xe buýt có thể chạy qua nhiều thành phố (xe buýt đường dài) trên lộ trình của mình và một thành phố cũng có thể có nhiều tuyến xe buýt đi qua, tạo thành một mạng lưới vận tải liên kết giữa các khu vực.

Thứ ba, Transerco có nhiều chi nhánh hoạt động tại các địa phương khác nhau, mỗi chi nhánh có mã chi nhánh và tên chi nhánh riêng. Một chi nhánh được đặt tại một vị trí địa lý cụ thể, một thành phố có thể có nhiều chi nhánh cùng hoạt động. Vai trò của chi nhánh là quản lý các tuyến xe buýt trực thuộc, mỗi tuyến xe buýt sẽ thuộc

quyền quản lý của 1 chi nhánh duy nhất. Việc xây dựng hệ thống chi nhánh giúp công ty phân vùng quản lý rõ ràng, từ đó tối ưu hoạt động điều hành và đảm bảo chất lượng dịch vụ.

Thứ tư, hệ thống cần theo dõi thông tin về các xe buýt hoạt động trong mạng lưới vận tải của tổng công ty. Mỗi xe buýt có một mã số xe, biển kiểm soát, số khung và số chỗ ngồi. Mỗi xe buýt được sử dụng cho một tuyến cụ thể, một tuyến có thể có nhiều xe buýt phục vụ.

Cuối cùng, hệ thống cũng cần có chức năng quản lý thông tin của các tài xế. Mỗi tài xế được quản lý dựa trên mã tài xế, họ tên, địa chỉ, số điện thoại và email. Việc phân công tài xế sẽ được thực hiện theo ngày, tức là mỗi xe buýt sẽ được giao cho một tài xế phụ trách trong một khoảng thời gian nhất định.

1.3. Yêu cầu cần đạt được

Sau khi hoàn thành quá trình thiết kế và xây dựng, hệ thống CSDL quản lý xe buýt liên tỉnh của Transerco cần đạt được những yêu cầu sau:

- Đảm bảo lưu trữ đầy đủ, chính xác và đồng nhất tất cả các thông tin liên quan đến tuyến xe, trạm dừng, thành phố, chi nhánh, phương tiện và tài xế của tổng công ty. Dữ liệu phải được tổ chức một cách khoa học, tránh trùng lặp và đảm bảo dễ dàng cập nhật khi có sự thay đổi.

- Hỗ trợ phân công, điều phối tuyến xe một cách hiệu quả. Quản lý các tuyến xe từ điểm xuất phát đến điểm kết thúc, bao gồm danh sách các trạm dừng trên lộ trình. Đồng thời, hệ thống phải cho phép tra cứu thông tin một cách nhanh chóng, hỗ trợ tìm kiếm thông tin theo các tiêu chí như mã tuyến, tên tuyến, mã xe buýt, mã tài xế hoặc chi nhánh quản lý.

- Có cơ chế bảo mật và phân quyền, đảm bảo chỉ những người có thẩm quyền mới có thể truy cập hoặc chỉnh sửa thông tin.

- Đảm bảo hiệu suất hoạt động cao, khả năng mở rộng linh hoạt để có thể thêm tuyến xe mới, trạm dừng mới hoặc chi nhánh mới khi quy mô tổng công ty phát triển.

- Giao diện phải trực quan, dễ sử dụng, giúp nhân viên thao tác nhanh chóng và hạn chế sai sót.

CHƯƠNG 2:

PHÂN TÍCH VÀ THIẾT KẾ CƠ SỞ DỮ LIỆU

2.1. Xây dựng mô hình thực thể quan hệ (Entity – Relationship Diagram – ERD)

2.1.1. Xác định các thực thể và các thuộc tính tương ứng

a) Thực thể Tuyến_Xe

- Mô tả: Lưu trữ thông tin về các tuyến xe.
- Thuộc tính: Mã tuyến (Primary Key – PK), tên tuyến, giờ bắt đầu chạy, giờ kết thúc, trạm khởi đầu, trạm trung gian và trạm kết thúc.

b) Thực thể Trạm_Xe

- Mô tả: Lưu trữ thông tin về các trạm dừng xe.
- Thuộc tính: Mã trạm (PK) và vị trí trạm.

c) Thực thể Phương_Tiện

- Mô tả: Lưu trữ thông tin về các phương tiện trong hệ thống.
- Thuộc tính: Mã phương tiện (PK), tên phương tiện, biển kiểm soát, tuyến phục vụ, hãng sản xuất, năm sản xuất, số khung và số chỗ ngồi.

d) Thực thể Tài_Xé

- Mô tả: Lưu trữ thông tin về các tài xế.
- Thuộc tính: Mã tài xế, họ tên, ngày sinh, địa chỉ, phương tiện được giao quản lý, số điện thoại và email.

e) Thực thể Chi_Nhánh

- Mô tả: Lưu trữ thông tin về các chi nhánh.
- Thuộc tính: Mã chi nhánh (PK), tên chi nhánh, địa chỉ, tuyến xe được giao quản lý, số điện thoại và email.

f) Thực thể Thành_Phố

- Mô tả: Lưu trữ thông tin về các thành phố có mạng lưới xe buýt đi qua.
- Thuộc tính: Mã thành phố (PK), tên thành phố và tuyến xe chạy qua địa bàn.,

2.1.2. Xác định liên kết và kiểu liên kết giữa các thực thể

a) Quan hệ giữa Tuyến_Xe và Phương_Tiện

Mỗi tuyến xe sẽ có nhiều phương tiện phục vụ, mỗi quan hệ này là mối quan hệ một – nhiều ($1 - m$). Ngược lại, tại một thời điểm, một phương tiện chỉ có thể phục vụ một tuyến xe. Do đó, mối quan hệ này là mối quan hệ một – một ($1 - 1$).

b) Quan hệ giữa Tuyến_Xe và Chi_Nhánh

Mỗi tuyến xe do một chi nhánh quản lý, mỗi quan hệ này là mối quan hệ $1 - 1$. Ngược lại, một chi nhánh có thể quản lý nhiều tuyến xe. Do đó, mối quan hệ này là mối quan hệ $1 - m$.

c) Quan hệ giữa Tuyến_Xe và Trạm_Xe

Mỗi tuyến xe sẽ đi qua nhiều trạm xe trong lộ trình của mình, mỗi quan hệ này là mối quan hệ $1 - m$. Tương tự, một trạm xe cũng sẽ có nhiều tuyến xe đi qua. Do đó, mối quan hệ này cũng là mối quan hệ $1 - m$.

d) Quan hệ giữa Tuyến_Xe và Thành_Phố

Mỗi tuyến xe có thể đi qua nhiều thành phố khác nhau trong lộ trình của mình (xe buýt đường dài), mỗi quan hệ này là mối quan hệ $1 - m$. Tương tự, một thành phố cũng sẽ có nhiều tuyến xe đi qua. Do đó, mối quan hệ này cũng là mối quan hệ $1 - m$.

e) Quan hệ giữa Phương_Tiện và Tài_Xế

Mỗi phương tiện được giao cho một tài xế quản lý, mỗi quan hệ này là mối quan hệ $1 - 1$. Tương tự, một tài xế cũng sẽ chỉ chịu trách nhiệm quản lý một phương tiện. Do đó, mối quan hệ này cũng là mối quan hệ $1 - 1$.

f) Quan hệ giữa Tài_Xế và Chi_Nhánh

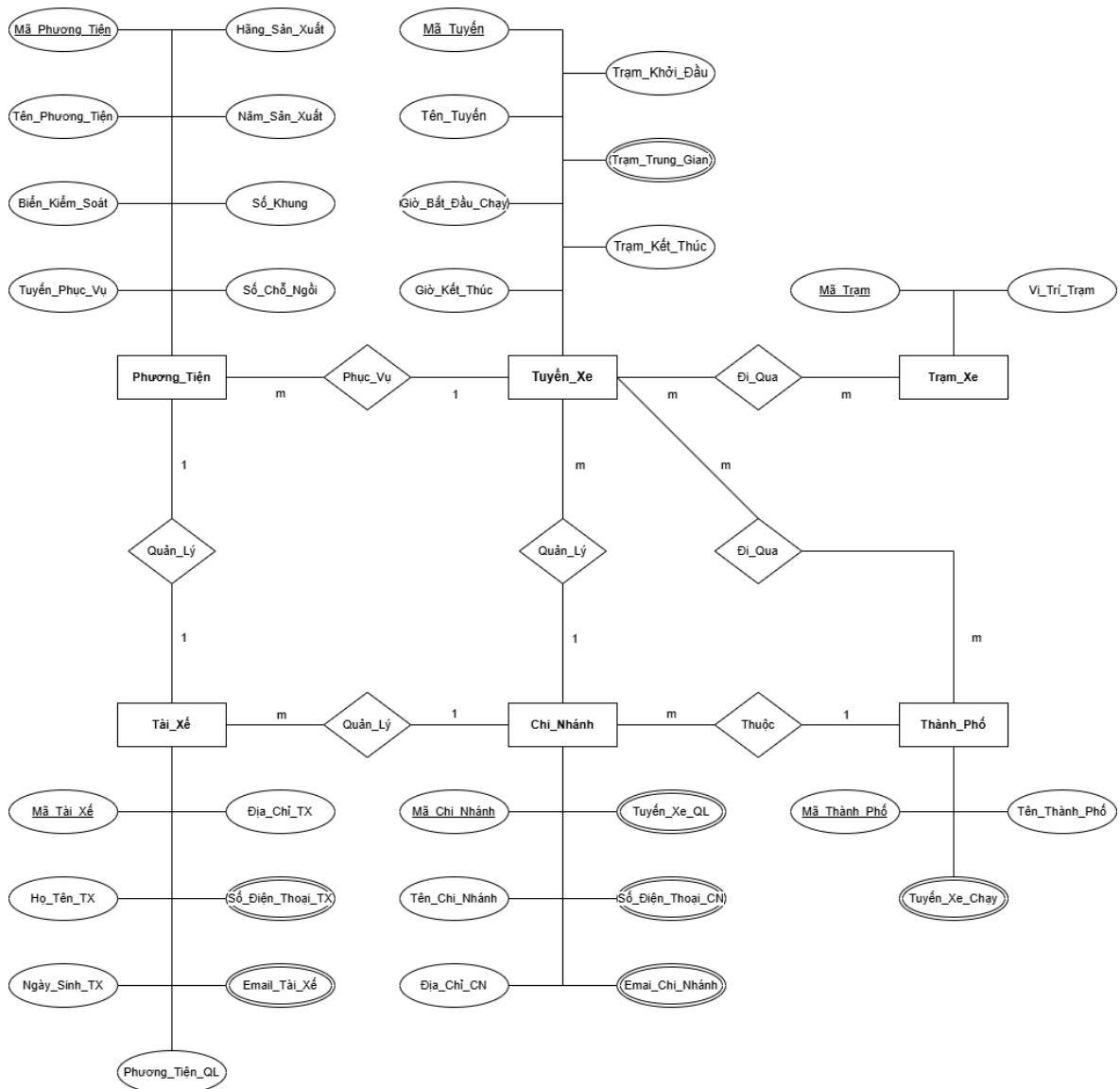
Mỗi tài xế sẽ do một chi nhánh quản lý, mỗi quan hệ này là mối quan hệ $1 - 1$. Ngược lại, một chi nhánh có thể quản lý nhiều tài xế. Do đó, mối quan hệ này là mối quan hệ $1 - m$.

g) Quan hệ giữa Chi_Nhánh và Thành_Phố

Mỗi chi nhánh sẽ nằm tại một thành phố, mỗi quan hệ này là mối quan hệ $1 - 1$. Ngược lại, một thành phố có thể có nhiều chi nhánh. Do đó, mối quan hệ này là mối quan hệ $1 - m$.

2.1.3. Xây dựng mô hình ERD

Từ những thực thể, thuộc tính của từng thực thể, liên kết và kiểu liên kết giữa các thực thể đã xác định tại *Mục 2.1.1.* và *Mục 2.1.2.* xây dựng được mô hình ERD như *Ảnh 1.* dưới:



Ảnh 1. Mô hình ERD.

2.2. Xây dựng mô hình quan hệ

2.2.1. Các quy tắc chuyển hóa từ mô hình ERD sang mô hình quan hệ

- Quy tắc 1: Mỗi thực thể trong mô hình ERD sẽ được chuyển thành một bảng trong mô hình quan hệ. Các thuộc tính của thực thể trở thành các cột của bảng đó. Nếu thực thể có PK, thực thể đó trở thành PK trong bảng tương ứng.

- Quy tắc 2: Đối với thuộc tính đa trị, tách thành một bảng riêng và đặt FK tham chiếu đến bảng đó vào bảng thực thể ban đầu.

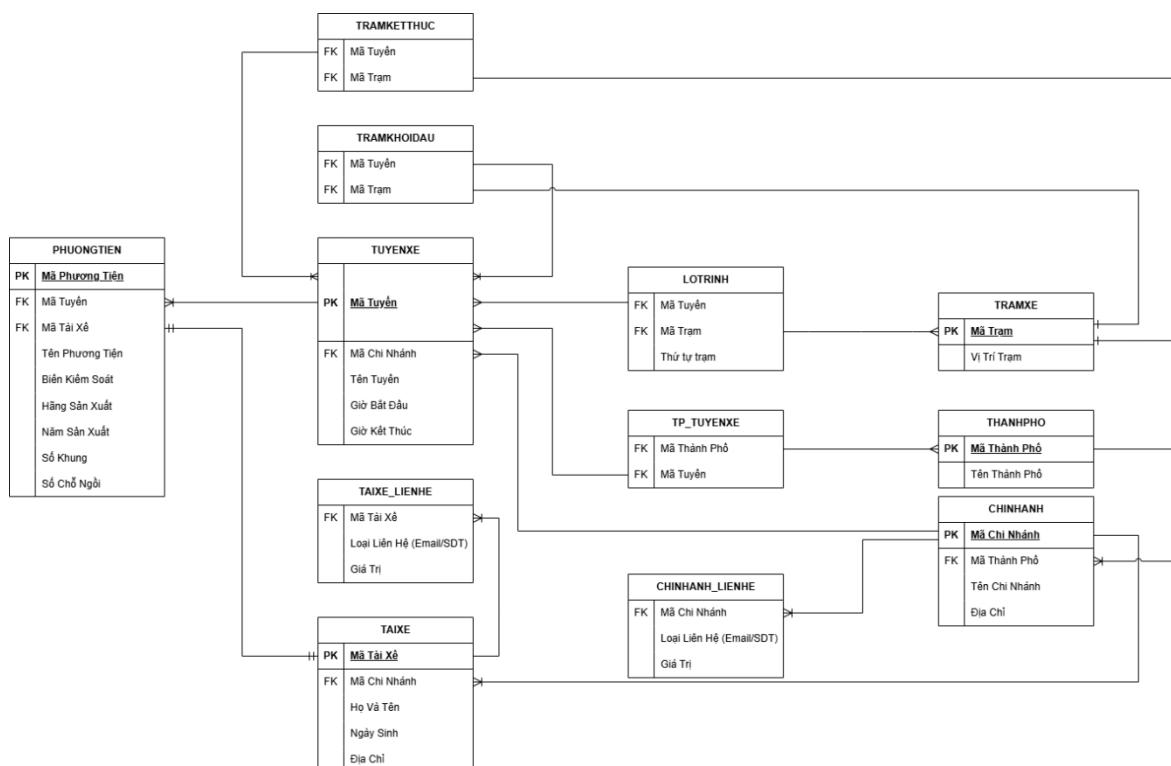
- Quy tắc 3: Đối với các mối quan hệ 1 – 1, đặt PK của phía tuỳ chọn sang phía bắt buộc.

- Quy tắc 4: Đối với các mối quan hệ 1 – m, đặt PK của phía một sang phía nhiều.

- Quy tắc 5: Đối với các mối quan hệ m – m, tạo một bảng trung gian chứa PK của cả hai bảng thực thể ban đầu.

2.2.2. Chuyển hoá từ mô hình ERD sang mô hình quan hệ

Áp dụng các quy tắc chuyển hoá đã nêu ở Mục 2.2.1., ta chuyển hoá mô hình ERD như *Ảnh 1*. thành mô hình quan hệ như *Ảnh 2*. dưới đây:



Ảnh 2. Mô hình quan hệ.

2.3. Xác định các ràng buộc

Bảng 1. Danh sách các ràng buộc.

STT	Tên	Bối cảnh	Nội dung tự nhiên	Nội dung hình thức
1	Ràng buộc 1	tbl.PHUONGTIEN	Mỗi phương tiện chỉ do một và chỉ một tài xế quản lý	$R_1 = \forall pt \in$ tbl.PHUONGTIEN: $pt.mataixe \in$ tbl.TAIXE.mataixe

2	Ràng buộc 2	tbl.PHUONGTIEN	Mỗi phương tiện chỉ hoạt động trong một và chỉ một tuyến xe	$R_2 = \forall pt \in$ tbl.PHUONGTIEN: $pt.matuyen \in$ tbl.TUYENXE.matuyen
3	Ràng buộc 3	tbl.PHUONGTIEN	Mã phương tiện của mỗi phương tiện phải là duy nhất trong hệ thống	$R_3 = \forall pt_1, pt_2 \in$ tbl.PHUONGTIEN: $pt_1 \neq pt_2$ $\rightarrow pt_1.maphuongtien \neq$ $pt_2.maphuongtien$
4	Ràng buộc 4	tbl.PHUONGTIEN	Biển kiểm soát của mỗi phương tiện trên hệ thống không nhận giá trị rỗng	$R_4 = \forall pt \in$ tbl.PHUONGTIEN: $pt.bienkiemsoat \neq NULL$
5	Ràng buộc 5	tbl.PHUONGTIEN	Số khung của mỗi phương tiện trên hệ thống không nhận giá trị rỗng	$R_5 = \forall pt \in$ tbl.PHUONGTIEN: $pt.sokhung \neq NULL$
6	Ràng buộc 6	tbl.PHUONGTIEN	Năm sản xuất của các phương tiện trên hệ thống không nhận giá trị nhỏ hơn 2005	$R_6 = \forall pt \in$ tbl.PHUONGTIEN: $pt.namsanxuat \geq 2005$
7	Ràng buộc 7	tbl.TUYENXE	Mỗi tuyến xe chỉ do một và chỉ một chi nhánh quản lý	$R_7 = \forall tuyenxe \in$ tbl.TUYENXE: $tuyenxe.matuyen \in$ tbl.CN_TUYENXE.matuyen

8	Ràng buộc 8	tbl.TUYENXE	Mã tuyến xe của mỗi tuyến xe phải là duy nhất trong hệ thống	$R_8 = \forall tuyenxe_1, tuyenxe_2 \in tbl.TUYENXE: tuyenxe_1 \neq tuyenxe_2 \rightarrow tuyenxe_1.maphuongtien \neq tuyenxe_2.maphuongtien$
9	Ràng buộc 9	tbl.TUYENXE	Giờ kết thúc của mỗi tuyến xe phải lớn hơn giờ bắt đầu của nó (Giờ được lấy theo định dạng 24 giờ/ngày)	$R_9 = \forall tuyenxe \in tbl.TUYENXE: tuyenxe.giokethuc > tuyenxe.giobatdau$
10	Ràng buộc 10	tbl.TAIXE	Mỗi tài xế chỉ do một và chỉ một chi nhánh quản lý	$R_{10} = \forall tx \in tbl.TAIXE: tx.machinhanh \in tbl.CHINHANH.machinhanh$
11	Ràng buộc 11	tbl.TAIXE	Mã tài xế của mỗi tài xế phải là duy nhất trong hệ thống	$R_{11} = \forall tx_1, tx_2 \in tbl.TAIXE: tx_1 \neq tx_2 \rightarrow tx_1.mataixe \neq tx_2.mataixe$
12	Ràng buộc 12	tbl.TAIXE	Họ và tên của mỗi tài xế trên hệ thống không nhận giá trị rỗng	$R_{12} = \forall tx \in tbl.TAIXE: tx.hovaten \neq NULL$
13	Ràng buộc 13	tbl.THANHPHO	Mã thành phố của mỗi thành phố phải là duy nhất trong hệ thống	$R_{13} = \forall tp_1, tp_2 \in tbl.THANHPHO: tp_1 \neq tp_2 \rightarrow tp_1.mathanhpho \neq tp_2.mathanhpho$
14	Ràng buộc 14	tbl.THANHPHO	Tên thành phố của mỗi thành phố trên hệ	$R_{14} = \forall tp \in tbl.THANHPHO: tp.tenthanhpho \neq NULL$

			thông không nhận giá trị rỗng	
15	Ràng buộc 15	tbl.TRAMXE	Mã trạm xe của mỗi trạm xe phải là duy nhất trong hệ thống	$R_{15} = \forall txe_1, txe_2 \in$ $tbl.TRAMXE: txe_1 \neq txe_2 \rightarrow$ $txe_1.matram \neq txe_2.matram$
16	Ràng buộc 16	tbl.TRAMXE	Vị trí trạm xe của mỗi trạm xe trên hệ thống không nhận giá trị rỗng	$R_{16} = \forall txe \in tbl.TRAMXE:$ $txe.vitritram \neq NULL$
17	Ràng buộc 17	tbl.CHINHANH	Mỗi chi nhánh chỉ thuộc một và chỉ một thành phố	$R_{17} = \forall cn \in$ $tbl.CHINHANH:$ $cn.mathanhpho \in$ $tbl.THANHPHO.mathanhpho$
18	Ràng buộc 18	tbl.CHINHANH	Mã chi nhánh của mỗi chi nhánh phải là duy nhất trong hệ thống	$R_{18} = \forall cn_1, cn_2 \in$ $tbl.CHINHANH: cn_1 \neq cn_2$ $\rightarrow cn_1.machinhanh \neq$ $cn_2.machinhanh$
19	Ràng buộc 19	tbl.CHINHANH	Tên chi nhánh của mỗi chi nhánh trên hệ thống không nhận giá trị rỗng	$R_{19} = \forall cn \in$ $tbl.CHINHANH:$ $cn.tenchinhanh \neq NULL$
29	Ràng buộc 20	tbl.CHINHANH	Địa chỉ chi nhánh của mỗi chi nhánh trên hệ thống không nhận giá trị rỗng	$R_{20} = \forall cn \in$ $tbl.CHINHANH: cn.diachi \neq$ $NULL$

2.4. Xác định các phụ thuộc hàm và chuẩn hóa theo chuẩn 3NF

2.4.1. Chuẩn hóa bảng tbl.PHUONGTIEN

tbl.PHUONGTIEN (Mã phương tiện (PK), Tên phương tiện, Biển kiểm soát, Hãng sản xuất, Năm sản xuất, Số khung, Số chỗ ngồi, Mã tài xế (FK), Mã tuyến (FK)).

- Không có thuộc tính nào trong bảng tbl.PHUONGTIEN là thuộc tính đa trị. Do đó, bảng tbl.PHUONGTIEN đạt chuẩn 1NF.

- Tất cả các thuộc tính không khóa trong bảng tbl.PHUONGTIEN đều phụ thuộc vào PK, không có phụ thuộc hàm một phần và không phụ thuộc vào bất kì thuộc tính không khóa nào khác. Do đó, bảng tbl.PHUONGTIEN đạt chuẩn 2NF và 3NF.

2.4.2. Chuẩn hóa bảng tbl.TUYENXE

tbl.TUYENXE (Mã tuyến (PK), Tên tuyến, Giờ bắt đầu chạy, Giờ kết thúc, Mã chi nhánh).

- Không có thuộc tính nào trong bảng tbl.TUYENXE là thuộc tính đa trị. Do đó, bảng tbl.TUYENXE đạt chuẩn 1NF.

- Tất cả các thuộc tính không khóa trong bảng tbl.TUYENXE đều phụ thuộc vào PK, không có phụ thuộc hàm một phần và không phụ thuộc vào bất kì thuộc tính không khóa nào khác. Do đó, bảng tbl.TUYENXE đạt chuẩn 2NF và 3NF.

2.4.3. Chuẩn hóa bảng tbl.TAIXE

tbl.TAIXE (Mã tài xế (PK), Họ và tên, Ngày sinh, Địa chỉ, Mã chi nhánh (FK)).

- Không có thuộc tính nào trong bảng tbl.TAIXE là thuộc tính đa trị. Do đó, bảng tbl.TAIXE đạt chuẩn 1NF.

- Tất cả các thuộc tính không khóa trong bảng tbl.TAIXE đều phụ thuộc vào PK, không có phụ thuộc hàm một phần và không phụ thuộc vào bất kì thuộc tính không khóa nào khác. Do đó, bảng tbl.TAIXE đạt chuẩn 2NF và 3NF.

2.4.4. Chuẩn hóa bảng tbl.TRAMXE

tbl.TRAMXE (Mã trạm (PK), Vị trí trạm).

- Không có thuộc tính nào trong bảng tbl.TRAMXE là thuộc tính đa trị. Do đó, bảng tbl.TRAMXE đạt chuẩn 1NF.

- Tất cả các thuộc tính không khóa trong bảng tbl.TRAMXE đều phụ thuộc vào PK, không có phụ thuộc hàm một phần và không phụ thuộc vào bất kì thuộc tính không khóa nào khác. Do đó, bảng tbl.TRAMXE đạt chuẩn 2NF và 3NF.

2.4.5. Chuẩn hóa bảng tbl.THANHPHO

tbl.THANHPHO (Mã thành phố (PK), Tên thành phố).

- Không có thuộc tính nào trong bảng tbl.THANHPHO là thuộc tính đa trị. Do đó, bảng tbl.THANHPHO đạt chuẩn 1NF.

- Tất cả các thuộc tính không khóa trong bảng tbl.THANHPHO đều phụ thuộc vào PK, không có phụ thuộc hàm một phần và không phụ thuộc vào bất kì thuộc tính không khóa nào khác. Do đó, bảng tbl.THANHPHO đạt chuẩn 2NF và 3NF.

2.4.6. Chuẩn hóa bảng tbl.CHINHANH

tbl.CHINHANH (Mã chi nhánh (PK), Tên chi nhánh, Địa chỉ, Mã thành phố (FK)).

- Không có thuộc tính nào trong bảng tbl.CHINHANH là thuộc tính đa trị. Do đó, bảng tbl.CHINHANH đạt chuẩn 1NF.

- Tất cả các thuộc tính không khóa trong bảng tbl.CHINHANH đều phụ thuộc vào PK, không có phụ thuộc hàm một phần và không phụ thuộc vào bất kì thuộc tính không khóa nào khác. Do đó, bảng tbl.CHINHANH đạt chuẩn 2NF và 3NF.

CHƯƠNG 3: TRIỂN KHAI CƠ SỞ DỮ LIỆU

3.1. Tạo cơ sở dữ liệu

3.1.1. Tạo cơ sở dữ liệu

Khung 1. Truy vấn khởi tạo cơ sở dữ liệu.

```
CREATE DATABASE Transerco_Bus_Management_DB
```

3.1.2. Tạo các bảng dữ liệu

Khung 2. Cấu trúc cơ bản của truy vấn khởi tạo bảng trong CSDL.

```
CREATE TABLE tên_bảng(  
    tên_cột_1 kiểu_dữ_liệu_1 [RÀNG_BUỘC_NẾU_CÓ],  
    tên_cột_2 kiểu_dữ_liệu_2 [RÀNG_BUỘC_NẾU_CÓ],  
    ...  
    tên_cột_n kiểu_dữ_liệu_n [RÀNG_BUỘC_NẾU_CÓ],  
) ;
```

Ví dụ 1: Truy vấn khởi tạo bảng PHUONGTIEN.

Khung 3. Truy vấn khởi tạo bảng PHUONGTIEN.

```
CREATE TABLE PHUONGTIEN(  
    Ma_Phuong_Tien NVARCHAR(10),  
    Ma_Tuyen NVARCHAR(10),  
    Ma_Tai_Xe NVARCHAR(10),  
    Ten_Phuong_Tien NVARCHAR(50),  
    Bien_Kiem_Soat NVARCHAR(12),  
    Hang_San_Xuat NVARCHAR (50),  
    Nam_San_Xuat INT,  
    So_Khung VARCHAR(17),  
    So.Cho.Ngoi INT  
) ;
```

3.1.3. Cài đặt PK và FK

Khung 4. Cấu trúc truy vấn thêm PK và FK.

```
-- Cấu trúc truy vấn thêm khóa chính  
ALTER TABLE tên_bảng ADD CONSTRAINT tên_ràng_buộc PRIMARY  
KEY (tên_cột);
```

```
-- Cấu trúc truy vấn thêm khóa ngoại
ALTER TABLE tên_bảng_con
ADD CONSTRAINT tên_ràng_buộc
FOREIGN KEY (tên_cột_con)
REFERENCES tên_bảng_cha (tên_cột_cha);
```

Ví dụ 2: Truy vấn thêm PK và FK cho bảng PHUONGTIEN.

Khung 5. Truy vấn thêm PK và FK cho bảng PHUONGTIEN.

```
-- Thêm PK
ALTER TABLE PHUONGTIEN ADD CONSTRAINT PK_PHUONGTIEN PRIMARY
KEY (Ma_Phuong_Tien);

-- Thêm FK
ALTER TABLE PHUONGTIEN ADD CONSTRAINT FK_PHUONGTIEN_TUYENXE
FOREIGN KEY (Ma_Tuyen) REFERENCES TUYENXE (Ma_Tuyen);
ALTER TABLE PHUONGTIEN ADD CONSTRAINT FK_PHUONGTIEN_TAIXE
FOREIGN KEY (Ma_Tai_Xe) REFERENCES TAIXE (Ma_Tai_Xe);
```

3.2. Nhập dữ liệu cho các bảng

Khung 6. Cấu trúc cơ bản truy vấn nhập dữ liệu vào bảng.

```
INSERT INTO tên_bảng (cột_1, cột_2, cột_3, ...)
VALUES (giá_trị_1, giá_trị_2, giá_trị_3, ...);
```

Ví dụ 3: Truy vấn nhập dữ liệu vào bảng PHUONGTIEN.

Khung 7. Truy vấn nhập dữ liệu vào bảng PHUONGTIEN.

```
INSERT INTO PHUONGTIEN (Ma_Phuong_Tien, Ma_Tuyen,
Ma_Tai_Xe, Ten_Phuong_Tien, Bien_Kiem_Soat, Hang_San_Xuat,
Nam_San_Xuat, So_Khung, So_Cho_Ngoi)
VALUES
('VH1', 'BUS1', 'TX1', 'Samco City D60', '29B -
184.23', 'Samco', '2020', 'RL4BB42A0R1234567', '60'),
('VH2', 'BUS1', 'TX2', 'Samco City D60', '29B -
937.58', 'Samco', '2020', 'RL4BC32F8R2345678', '60'),
('VH3', 'BUS1', 'TX3', 'Samco City D60', '29B -
462.91', 'Samco', '2020', 'RL4BD21G9R3456789', '60'),
```

```

('VH4', 'BUS1', 'TX4', 'Samco City D60', '29B - 
715.36', 'Samco', '2020', 'RL4BE51H1R4567890', '60'),
('VH5', 'BUS1', 'TX5', 'Samco City D60', '29B - 
528.47', 'Samco', '2020', 'RL4BF62J2R5678901', '60'),
...
('VH25', 'BUS11', 'TX25', 'Thaco Meadow 89CT', '29B - 
967.53', 'Thaco', '2024', 'RL4CB82E4R5678901', '60');

```

3.3. Truy vấn cơ bản (SELECT, INSERT, UPDATE, DELETE)

3.3.1. Truy vấn SELECT

Ví dụ 4: Thực hiện truy vấn theo các yêu cầu sau:

- Lấy toàn bộ thông tin trong bảng PHUONGTIEN.
- Lấy thông tin phương tiện có mã phương tiện ‘VH25’.
- Lấy thông tin toàn bộ phương tiện do hãng Samco sản xuất năm 2023 và có 50 chỗ ngồi.

Khung 8. Truy vấn SELECT.

```

-- a) Lấy toàn bộ dữ liệu trong bảng phương tiện
SELECT * FROM PHUONGTIEN;

-- b) Lấy dữ liệu phương tiện có mã 'VH25'
SELECT * FROM PHUONGTIEN WHERE Ma_Phuong_Tien = 'VH25';

-- c) Lấy toàn bộ dữ liệu các phương tiện do Samco sản xuất
-- năm 2023 có số chỗ ngồi là 50
SELECT * FROM PHUONGTIEN WHERE
Hang_San_Xuat = 'Samco' AND Nam_San_Xuat = 2023 AND
So.Cho.Ngoi = 50;

```

The screenshot displays three SSMS windows. The top-left window shows the raw SQL code for a SELECT query. The other two windows show the execution results, which are identical. Both results grids have the following columns:

Ma_Phong_Tien	Ma_Tren	Ma_Tai_Xe	Ten_Phong_Tien	Bien_Kiem_Soat	Hang_San_Keut	Nam_San_Keut	So_Khung	So_Chi_Nganh
VN1	BUS1	TX01	Saint City 488	298 - 983.32	Saint	2823	RL4827791823454	48
VN2	BUS2	TX02	Thien Garten 797	298 - 894.67	Saint	2823	RL4827791823457	58
VN3	BUS3	TX03	Saint City 1.55	298 - 372.38	Saint	2823	RL4827791823458	58
VN4	BUS4	TX04	Saint City 1.55	298 - 588.92	Saint	2823	RL4827791823459	58
VN5	BUS5	TX05	Saint City 1.55	298 - 372.38	Saint	2823	RL4827791823460	58
VN6	BUS6	TX06	Saint City 1.55	298 - 268.34	Saint	2823	RL4827791823461	58
VN7	BUS7	TX07	Saint City 1.55	298 - 983.21	Saint	2823	RL4827791823462	58
VN8	BUS8	TX08	Saint City 1.55	298 - 983.21	Saint	2823	RL4827791823463	58
VN9	BUS9	TX09	Saint City 1.49	298 - 694.52	Saint	2823	RL4827791823464	48
VN10	BUS10	TX10	Saint City 1.49	298 - 859.13	Saint	2823	RL4827791823465	48
VN11	BUS11	TX11	Saint City 1.49	298 - 372.38	Saint	2823	RL4827791823466	48
VN12	BUS12	TX12	Saint City 1.49	298 - 977.58	Saint	2823	RL4827791823467	68
VN13	BUS13	TX13	Saint City 1.49	298 - 471.69	Saint	2823	RL4827791823468	48
VN14	BUS14	TX14	Saint City 1.49	298 - 983.21	Saint	2823	RL4827791823469	48
VN15	BUS15	TX15	Saint City 1.49	298 - 983.21	Saint	2823	RL4827791823470	48
VN16	BUS16	TX16	Saint City 1.49	298 - 983.21	Saint	2823	RL4827791823471	48
VN17	BUS17	TX17	Saint City 1.49	298 - 694.52	Saint	2823	RL4827791823472	48
VN18	BUS18	TX18	Saint City 1.49	298 - 859.13	Saint	2823	RL4827791823473	48
VN19	BUS19	TX19	Saint City 1.49	298 - 372.38	Saint	2823	RL4827791823474	48
VN20	BUS20	TX20	Saint City 1.49	298 - 977.58	Saint	2823	RL4827791823475	68
VN21	BUS21	TX21	Saint City 1.49	298 - 471.69	Saint	2823	RL4827791823476	48
VN22	BUS22	TX22	Thien Meilleur 997	298 - 987.53	Thien	2824	RL4827791823477	68
VN23	BUS23	TX23	Thien Meilleur 997	298 - 852.31	Thien	2824	RL4827791823478	68
VN24	BUS24	TX24	Thien Meilleur 997	298 - 987.53	Thien	2824	RL4827791823479	68
VN25	BUS25	TX25	Thien Meilleur 997	298 - 987.53	Thien	2824	RL4827791823480	68
VN26	BUS26	TX26	Thien Meilleur 997	298 - 987.53	Thien	2824	RL4827791823481	68

Ảnh 3. Kết quả thực hiện truy vấn SELECT.

3.3.2. Truy vấn INSERT

Ví dụ 5: Thêm dữ liệu của 5 tài xế mới có mã từ ‘TX26’ đến ‘TX30’ vào bảng TAIXE.

Khung 9. Truy vấn thêm dữ liệu tài xế mới vào bảng TAIXE.

```
INSERT INTO TAIXE (Ma_Tai_Xe, Ma_Chi_Nhanh, Ho_Va_Ten,
Ngay_Sinh, Dia_Chi)
VALUES
    ('TX26', 'CN8', N'Ngô Duy Hưng', '1970-03-19', N'Số
112, đường Lê Trọng Tấn, quận Thanh Xuân, Hà Nội'),
    ('TX27', 'CN8', N'Nguyễn Thành Long', '1973-02-01',
N'Số 435, đường Xuân Thủy, quận Cầu Giấy, Hà Nội'),
    ('TX28', 'CN8', N'Phạm Minh Quang', '1979-01-10', N'Số
213, phố Láng Hạ, quận Đống Đa, Hà Nội'),
    ('TX29', 'CN8', N'Đào Mạnh Hùng', '1972-11-18', N'Số
69, đường Đại Cồ Việt, quận Hai Bà Trưng, Hà Nội'),
    ('TX30', 'CN8', N'Lưu Thành Quân', '1972-08-25', N'Số
189, đường Nguyễn Chí Thanh, quận Ba Đình, Hà Nội');
```

Ảnh 4. Số lượng bản ghi trong bảng TAIXE trước (25 bản ghi - ảnh trái) và sau (30 bản ghi - ảnh phải) khi thực hiện truy vấn INSERT.

3.3.3. Truy vấn UPDATE

Ví dụ 6: Sử dụng truy vấn UPDATE để thực hiện các yêu cầu sau:

- Cập nhật thông tin giờ kết thúc cho tuyến xe 01 trong bảng TUYENXE thành 22:00.
- Cập nhật số điện thoại mới cho tài xế có mã tài xế ‘TX6’ trong bảng TAIXE_LIENHE.

Khung 10. Truy vấn UPDATE.

```
-- a) Thay đổi giờ kết thúc tuyến 01
-- Kiểm tra thông tin tuyến 01
SELECT * FROM TUYENXE WHERE Ma_Tuyen = 'BUS1';

-- Cập nhật giờ kết thúc mới
UPDATE TUYENXE SET Gio_Ket_Thuc = '22:00' WHERE Ma_Tuyen = 'BUS1'

-- b) Thay đổi số điện thoại tài xế có mã tài xế 'TX6'
-- Kiểm tra thông tin tài xế có mã tài xế 'TX6'
SELECT * FROM TAIXE_LIENHE
WHERE Ma_Tai_Xe = 'TX6' AND Loai_Lien_He_Email_SDT = N'Số
điện thoại';

-- Cập nhật số điện thoại mới
```

```

UPDATE TAIXE_LIENHE SET Gia_Tri = '0923 738 321'
WHERE Ma_Tai_Xe = 'TX6' AND Loai_Lien_He_Email_SDT = N'Số
điện thoại';

```

The screenshots show the following sequence of operations:

- Initial State:** A screenshot of the SQL Server Management Studio interface showing the query window with the UPDATE statement.
- Before Update:** A screenshot showing the results of the query `SELECT * FROM TAIXE WHERE Ma_Tai_Xe = 'TX6'` which returns one row: Ma_Tai_Xe: TX6, Ma_Xe_Mach: 01, Ten_Xe: 'Xe 1', Gio_Bat_Dau: 05:00:00, Gio_Ket_Thuc: 21:00:00.
- Running Update:** A screenshot of the query window with the UPDATE command executed. The message pane shows the command being run and the affected row count.
- After Update:** A screenshot showing the results of the query `SELECT * FROM TAIXE WHERE Ma_Tai_Xe = 'TX6'` which returns one row: Ma_Tai_Xe: TX6, Ma_Xe_Mach: 01, Ten_Xe: 'Số điện thoại', Gio_Bat_Dau: 05:00:00, Gio_Ket_Thuc: 21:00:00.

Ảnh 5. Dữ liệu của tuyến 01 và tài xế có mã 'TX6' trước (2 ảnh trên) và sau (2 ảnh dưới) khi thực hiện truy vấn UPDATE.

3.3.4. Câu lệnh DELETE

Ví dụ 7: Sử dụng truy vấn DELETE xóa các tài xế có mã từ 'TX26' tới 'TX30' trong bảng TAIXE.

Khung 11. Truy vấn DELETE.

```

DELETE FROM TAIXE WHERE Ma_Tai_XE
IN ( 'TX26', 'TX27', 'TX28', 'TX29', 'TX30' );

```

3.4. Truy vấn nâng cao (INNER JOIN, GROUP BY, HAVING, SUBQUERY)

3.4.1. Truy vấn sử dụng INNER JOIN

Ví dụ 8: Thực hiện các yêu cầu sau bằng cách sử dụng INNER JOIN.

- Lấy dữ liệu phương tiện và tài xế điều khiển phương tiện đó.
- Lấy dữ liệu vị trí các điểm dừng của tuyến 01.
- Lấy tất cả dữ liệu của tài xế có mã 'TX10' trong CSDL.

Khung 12. Truy vấn sử dụng INNER JOIN.

-- a) Lấy dữ liệu phương tiện và tài xế điều khiển chúng

SELECT

```
    pt.Ma_Phuong_Tien, pt.Ten_Phuong_Tien,  
    pt.Bien_Kiem_Soat, pt.Ma_Tai_Xe,  
    tx.Ho_Va_Ten  
FROM PHUONGTIEN pt INNER JOIN TAIXE tx ON pt.Ma_Tai_Xe =  
tx.Ma_Tai_Xe;
```

-- b) Lấy dữ liệu các điểm dừng của tuyến 01

SELECT

```
    lt.Ma_Tuyen, lt.Ma_Tram,  
    trx.Vi_Tri_Tram,  
    lt.Thu_Tu_Tram  
FROM LOTRINH lt INNER JOIN TRAMXE trx  
ON lt.Ma_Tram = trx.Ma_Tram WHERE lt.Ma_Tuyen = 'BUS1';
```

-- c) Lấy tất cả dữ liệu của tài xế có mã tài xế 'TX10'

SELECT

```
    tx.Ma_Tai_Xe, tx.Ho_Va_Ten, tx.Ngay_Sinh, tx.Dia_Chi,  
    MAX(CASE WHEN txlh.Loai_Lien_He_Email_SDT = 'Email'  
    THEN txlh.Gia_Tri END) AS Email,  
    MAX(CASE WHEN txlh.Loai_Lien_He_Email_SDT = N'Số điện  
    thoại' THEN txlh.Gia_Tri END) AS So_Dien_Thoai  
FROM TAIXE tx INNER JOIN TAIXE_LIENHE txlh ON tx.Ma_Tai_Xe  
= txlh.Ma_Tai_Xe  
WHERE tx.Ma_Tai_Xe = 'TX10'  
GROUP BY tx.Ma_Tai_Xe, tx.Ho_Va_Ten, tx.Ngay_Sinh,  
tx.Dia_Chi;
```

Ảnh 6. Kết quả sau khi thực hiện truy xuất thông tin của phương tiện (ảnh trên), trạm dừng (ảnh dưới trái) và tài xế (ảnh dưới phải).

3.4.2. Truy vấn sử dụng GROUP BY

Ví dụ 9: Thực hiện các yêu cầu sau bằng cách sử dụng GROUP BY.

- Đếm số tài xế của mỗi chi nhánh.
- Đếm số trạm dừng của mỗi tuyến.
- Đếm số trạm dừng trên lộ trình của tuyến có mã 'BUS8'.
- Đếm số phương tiện do Thaco sản xuất.

Khung 13. Truy vấn sử dụng GROUP BY.

-- a) Đếm số tài xế của mỗi chi nhánh

```
SELECT Ma_Chi_Nhanh, COUNT(*) AS So_Tai_Xe
FROM TAIXE
GROUP BY Ma_Chi_Nhanh;
```

-- b) Đếm số trạm dừng của mỗi tuyến

```
SELECT Ma_Tuyen, COUNT(*) AS So_Tram_Dung
FROM LOTRINH
GROUP BY Ma_Tuyen;
```

-- c) Đếm số trạm dừng trên lộ trình tuyến buýt có mã 'BUS8'

```
SELECT Ma_Tuyen, COUNT(*) AS So_Tram_Dung
FROM LOTHINH
WHERE Ma_Tuyen = 'BUS8'
GROUP BY Ma_Tuyen;
```

-- d) Đếm số phương tiện do Thaco

```
SELECT Hang_San_Xuat, COUNT(*) AS So_Luong
FROM PHUONGTIEN
WHERE Hang_San_Xuat = 'Thaco'
GROUP BY Hang_San_Xuat;
```

Ma_Tuyen	So_Tram_Dung
BUS1	38
BUS2	5
BUS8	38

Hang_San_Xuat	So_Luong
Thaco	18

Ảnh 7. Kết quả sau khi thực hiện truy vấn sử dụng GROUP BY.

3.4.3. Truy vấn sử dụng HAVING

Ví dụ 10: Thực hiện các yêu cầu sau bằng cách sử dụng HAVING.

- Lọc ra các chi nhánh có nhiều hơn 6 tài xế.
- Lọc các tuyến xe có nhiều hơn 30 trạm dừng.

c) Lọc các chi nhánh quản lý nhiều hơn một tuyến xe.

Khung 14. Truy vấn sử dụng HAVING.

-- a) Lọc chi nhánh có nhiều hơn 6 tài xế

SELECT

```
tx.Ma_Chi_Nhanh,  
cn.Ten_Chi_Nhanh, cn.Dia_Chi,  
COUNT(*) AS So_Luong  
FROM TAIXE tx INNER JOIN CHINHANH cn ON tx.Ma_Chi_Nhanh =  
cn.Ma_Chi_Nhanh  
GROUP BY tx.Ma_Chi_Nhanh, cn.Ten_Chi_Nhanh, cn.Dia_Chi  
HAVING COUNT(*) > 6;
```

-- b) Lọc các tuyến xe có nhiều hơn 30 trạm dừng

SELECT

```
lt.Ma_Tuyen,  
tx.Ten_Tuyen,  
COUNT(*) AS So_Tram  
FROM LOTRINH lt INNER JOIN TUYENXE tx ON lt.Ma_Tuyen =  
tx.Ma_Tuyen  
GROUP BY lt.Ma_Tuyen, tx.Ten_Tuyen  
HAVING COUNT(*) > 30;
```

-- c) Lọc các chi nhánh quản lý nhiều hơn 1 tuyến xe

SELECT

```
cn.Ma_Chi_Nhanh, cn.Ten_Chi_Nhanh,  
COUNT(*) AS So_Tuyen,  
cn.Dia_Chi  
FROM CHINHANH cn INNER JOIN TUYENXE tx ON cn.Ma_Chi_Nhanh =  
tx.Ma_Chi_Nhanh  
GROUP BY cn.Ma_Chi_Nhanh, cn.Ten_Chi_Nhanh,  
tx.Ma_Chi_Nhanh, cn.Dia_Chi  
HAVING COUNT(*) > 1;
```

```

-- Lấy dữ liệu các tuyến xe có số điểm dừng lớn hơn 20
SELECT Ma_Tuyen, Ten_Tuyen, Chi_Nhanh, So_Diem_Dung
FROM TUYENXE
HAVING So_Diem_Dung > 20
-- Lấy dữ liệu các trạm xe có tên xe có số điểm dừng lớn hơn 20
SELECT So_Tram, Ten_Tram, Chi_Nhanh
FROM TRAMXE
HAVING So_Diem_Dung > 20
-- Lấy dữ liệu tài xế và các phương tiện có số điểm dừng lớn hơn 40 chỗ ngồi thuộc chi nhánh 6
SELECT Ma_Tai_Xe, Ten_Xe, Chi_Nhanh, So_Ghe_Ngoi
FROM TAIXE
INNER JOIN CHINHANH ON TAIXE.Ma_Chi_Nhanh = CHINHANH.Ma_Chi_Nhanh
HAVING So_Ghe_Ngoi > 40

```

Ảnh 8. Kết quả sau khi thực hiện truy vấn sử dụng HAVING.

3.4.4. Truy vấn sử dụng SUBQUERY

Ví dụ 11: Thực hiện các yêu cầu sau bằng cách sử dụng SUBQUERY.

a) Lấy dữ liệu các tuyến xe thuộc chi nhánh 1 mà có số điểm dừng lớn hơn

20.

b) Lấy dữ liệu các trạm xe có nhiều hơn 1 tuyến xe đi qua.

c) Lấy dữ liệu tài xế và các phương tiện có nhiều hơn 40 chỗ ngồi thuộc quản lý của chi nhánh 6.

d) Lấy dữ liệu các phương tiện có giờ kết thúc là 21:00.

Khung 15. Truy vấn sử dụng SUBQUERY.

```
-- a) Lấy dữ liệu các tuyến xe thuộc chi nhánh 1 mà có số  
-- điểm dừng lớn hơn 20
```

SELECT

```
tx.Ma_Tuyen, tx.Ten_Tuyen,  
cn.Ten_Chi_Nhanh,  
lt.So_Tram
```

FROM TUYENXE tx

```
INNER JOIN CHINHANH cn ON tx.Ma_Chi_Nhanh = cn.Ma_Chi_Nhanh
```

```

INNER JOIN
    (SELECT Ma_Tuyen, COUNT(*) AS So_Tram FROM LOTRINH
     GROUP BY Ma_Tuyen HAVING COUNT(*) > 20) lt ON
tx.Ma_Tuyen = lt.Ma_Tuyen
WHERE tx.Ma_Chi_Nhanh = 'CN1';

-- b) Lấy dữ liệu các trạm có nhiều hơn 1 tuyến xe đi qua
SELECT
    tx.Ma_Tram, tx.Vi_Tri_Tram,
    lt.So_Tuyen
FROM TRAMXE tx
INNER JOIN
    (SELECT Ma_Tram, COUNT(*) AS So_Tuyen FROM LOTRINH
     GROUP BY Ma_Tram HAVING COUNT(*) > 1) lt ON tx.Ma_Tram
= lt.Ma_Tram;

-- c) Lấy dữ liệu tài xế và các phương tiện có số chỗ ngồi
-- lớn hơn 40 thuộc quản lý của chi nhánh 6
SELECT
    tx.Ma_Tai_Xe, tx.Ho_Va_Ten, tx.Ma_Chi_Nhanh,
    tx.Ngay_Sinh, tx.Dia_Chi,
    pt.Ma_Phuong_Tien, pt.Ten_Phuong_Tien,
    pt.Bien_Kiem_Soat, pt.So.Cho_Ngoi
FROM TAIXE tx
INNER JOIN
    (SELECT Ma_Tai_Xe, Ma_Phuong_Tien,
     Ten_Phuong_Tien, Bien_Kiem_Soat, So.Cho_Ngoi
     FROM PHUONGTIEN WHERE So.Cho_Ngoi > 40) pt ON
tx.Ma_Tai_Xe = pt.Ma_Tai_Xe
WHERE tx.Ma_Chi_Nhanh = 'CN6';

-- d) Lấy dữ liệu các phương tiện có giờ kết thúc là 21:00
SELECT

```

```

pt.Ma_Phuong_Tien, pt.Ten_Phuong_Tien,
pt.Bien_Kiem_Soat,
tx.Gio_Ket_Thuc
FROM PHUONGTIEN pt
INNER JOIN
    (SELECT Ma_Tuyen, Gio_Ket_Thuc
     FROM TUYENXE WHERE Gio_Ket_Thuc = '21:00')
tx ON pt.Ma_Tuyen = tx.Ma_Tuyen;

```

The screenshots illustrate the execution of the provided T-SQL query. The first window shows the overall query structure. The subsequent three windows show the results of the subquery execution, which lists various districts and their names.

Top Left Window (Initial Query):

```

pt.Ma_Phuong_Tien, pt.Ten_Phuong_Tien,
pt.Bien_Kiem_Soat,
tx.Gio_Ket_Thuc
FROM PHUONGTIEN pt
INNER JOIN
    (SELECT Ma_Tuyen, Gio_Ket_Thuc
     FROM TUYENXE WHERE Gio_Ket_Thuc = '21:00')
tx ON pt.Ma_Tuyen = tx.Ma_Tuyen;

```

Top Right Window (Subquery Results):

Ma_Tuyen	Ten_Tuyen	Ten_Chi_Manh	Su_Tran
0051	01	XI nguyễn huy Ngu	33

Bottom Left Window (Subquery Results):

Ma_Tuyen	Ten_Tuyen	Vl_Vt_Tri_Thanh	Su_Tran
0051	01	XI nguyễn huy Ngu	33

Bottom Right Window (Subquery Results):

Ma_Tuyen	Vl_Vt_Tri_Thanh	Su_Tran	
0051	01	XI nguyễn huy Ngu	33

Ảnh 9. Kết quả sau khi thực hiện truy vấn sử dụng SUBQUERY.

CHƯƠNG 4: LẬP TRÌNH T-SQL

4.1. Tạo view cho CSDL

Ví dụ 12: Tạo VIEW để theo dõi tài xế, tuyển và chi nhánh quản lý của 1 phương tiện.

Khung 16. Khởi tạo VIEW theo dõi chi tiết phương tiện.

```
GO
CREATE VIEW view_CHITIET_PHUONGTIEN AS
SELECT
    pt.Ma_Phuong_Tien,
    pt.Ma_Tuyen, tx.Ten_Tuyen,
    tx.Ma_Chi_Nhanh, cn.Ten_Chi_Nhanh,
    pt.Ma_Tai_Xe, txe.Ho_Va_Ten,
    STRING_AGG(CASE WHEN txlh.Loai_Lien_He_Email_SDT =
'Email' THEN txlh.Gia_Tri END, ', ') AS Email,
    STRING_AGG(CASE WHEN txlh.Loai_Lien_He_Email_SDT = N'Số
điện thoại' THEN txlh.Gia_Tri END, ', ') AS So_Dien_Thoai,
    pt.Ten_Phuong_Tien, pt.Bien_Kiem_Soat,
    pt.Hang_San_Xuat, pt.Nam_San_Xuat,
    pt.So_Khung, pt.So_Cho_Ngoi
FROM PHUONGTIEN pt
INNER JOIN TUYENXE tx ON pt.Ma_Tuyen = tx.Ma_Tuyen
INNER JOIN CHINHANH cn ON tx.Ma_Chi_Nhanh = cn.Ma_Chi_Nhanh
INNER JOIN TAIXE txe ON pt.Ma_Tai_Xe = txe.Ma_Tai_Xe
INNER JOIN TAIXE_LIENHE txlh ON txe.Ma_Tai_Xe =
txlh.Ma_Tai_Xe
GROUP BY
    pt.Ma_Phuong_Tien, pt.Ma_Tuyen, tx.Ten_Tuyen,
    tx.Ma_Chi_Nhanh, cn.Ten_Chi_Nhanh,
    pt.Ma_Tai_Xe, txe.Ho_Va_Ten,
    pt.Ten_Phuong_Tien, pt.Bien_Kiem_Soat,
    pt.Hang_San_Xuat, pt.Nam_San_Xuat,
```

pt.So_Khung, pt.So_Cho_Ngoi;

Khung 17. Truy vấn VIEW theo dõi chi tiết phương tiện.

GO

SELECT * FROM view_CHITIET_PHUONGTIEN;

	Ma_Phuong_Tien	Ma_Tuyen	Ten_Tuyen	Ma_Chi_Nhanh	Ten_Chi_Nhanh	Ma_Tai_Xe	Ho_Va_Ten	Email	So_Dien_Thoai
1	VH1	BUS1	01	CN1	Xí nghiệp Bus Hà Nội	TX1	Nguyễn Văn Hùng	hungnv@transerco.com.vn	0983 811 231
2	VH10	BUS2	03A	CN1	Xí nghiệp Bus Hà Nội	TX10	Trịnh Văn Dũng	dungtv@transerco.com.vn	0389 654 321
3	VH11	BUS8	42	CN6	Xí nghiệp Xe khách Nam Hà Nội	TX11	Vũ Quang Đạt	datvq@transerco.com.vn	0901 567 890
4	VH12	BUS8	42	CN6	Xí nghiệp Xe khách Nam Hà Nội	TX12	Lý Xuân Trường	truonglx@transerco.com.vn	0936 789 012
5	VH13	BUS8	42	CN6	Xí nghiệp Xe khách Nam Hà Nội	TX13	Tạ Đình Thắng	thangtd@transerco.com.vn	0333 222 111
6	VH14	BUS8	42	CN6	Xí nghiệp Xe khách Nam Hà Nội	TX14	Lương Văn Hải	hailv@transerco.com.vn	0353 888 777
7	VH15	BUS8	42	CN6	Xí nghiệp Xe khách Nam Hà Nội	TX15	Nguyễn Hữu Phúc	phucnhn@transerco.com.vn	0974 123 456
8	VH16	BUS8	42	CN6	Xí nghiệp Xe khách Nam Hà Nội	TX16	Mai Văn Bảo	baomv@transerco.com.vn	0902 345 678
9	VH17	BUS8	42	CN6	Xí nghiệp Xe khách Nam Hà Nội	TX17	Tô Hồng Phúc	phucth@transerco.com.vn	0918 987 654
10	VH18	BUS8	42	CN6	Xí nghiệp Xe khách Nam Hà Nội	TX18	Cao Ngọc Sơn	soncn@transerco.com.vn	0383 765 432
11	VH19	BUS8	42	CN6	Xí nghiệp Xe khách Nam Hà Nội	TX19	Lâm Nhật Minh	minhn@transerco.com.vn	0396 543 210
12	VH2	BUS1	01	CN1	Xí nghiệp Bus Hà Nội	TX2	Trần Dinh Hòa	hoatd@transerco.com.vn	0938 765 432
13	VH20	BUS8	42	CN6	Xí nghiệp Xe khách Nam Hà Nội	TX20	Châu Bình Long	longcd@transerco.com.vn	0989 876 543
14	VH21	BUS11	54	CN4	Xí nghiệp xe buýt Yên Viên	TX21	Hà Mạnh Đức	ducmh@transerco.com.vn	0935 678 901
15	VH22	BUS11	54	CN4	Xí nghiệp xe buýt Yên Viên	TX22	Triệu Quang Tùng	tungt@transerco.com.vn	0321 456 789
16	VH23	BUS11	54	CN4	Xí nghiệp xe buýt Yên Viên	TX23	Kiều Hải Nam	namkh@transerco.com.vn	0377 654 321
17	VH24	BUS11	54	CN4	Xí nghiệp xe buýt Yên Viên	TX24	Nguyễn Hữu Toàn	toannh@transerco.com.vn	0972 789 654
18	VH25	BUS11	54	CN4	Xí nghiệp xe buýt Yên Viên	TX25	Hoàng Bình Lâm	lamhd@transerco.com.vn	0905 432 109

Ảnh 10. Kết quả truy vấn VIEW theo dõi chi tiết phương tiện.

Ví dụ 13: Tạo VIEW để theo dõi dữ liệu tài xế và chi nhánh quản lý.

Khung 18. Khởi tạo VIEW theo dõi chi tiết tài xế.

GO

CREATE VIEW view_CHITIET_TAIXE AS

SELECT

```

tx.Ma_Tai_Xe, tx.Ho_Va_Ten,
tx.Ma_Chi_Nhanh, cn.Ten_Chi_Nhanh,
tx.Ngay_Sinh, tx.Dia_Chi,
STRING_AGG(CASE WHEN txlh.Loai_Lien_He_Email_SDT =
'Email' THEN txlh.Gia_Tri END, ', ') AS Email,
STRING_AGG(CASE WHEN txlh.Loai_Lien_He_Email_SDT = N'Số
điện thoại' THEN txlh.Gia_Tri END, ', ') AS So_Dien_Thoai

```

```

FROM TAIXE tx
INNER JOIN CHINHANH cn ON tx.Ma_Chi_Nhanh = tx.Ma_Chi_Nhanh
INNER JOIN TAIXE_LIENHE txlh ON tx.Ma_Tai_Xe =
txlh.Ma_Tai_Xe
GROUP BY
    tx.Ma_Tai_Xe, tx.Ho_Va_Ten,
    tx.Ma_Chi_Nhanh, cn.Ten_Chi_Nhanh,
    tx.Ngay_Sinh, tx.Dia_Chi;

```

Khung 19. Truy vấn VIEW chi tiết tài xe.

```

GO
SELECT * FROM view_CHITIET_TAIXE;

```

	Ma_Tai_Xe	Ho_Va_Ten	Ma_Chi_Nhanh	Ten_Chi_Nhanh	Ngay_Sinh	Dia_Chi	Email
1	TX1	Nguyễn Văn Hùng	CN1	Công ty Cổ phần Vận tải dịch vụ Liên Ninh	1978-05-12	Số 12, đường Nguyễn Trãi, quận Thanh Xuân, Hà Nội	hungnv@transer
2	TX10	Trịnh Văn Dũng	CN1	Công ty Cổ phần Vận tải dịch vụ Liên Ninh	1972-09-05	Số 90, đường Trường Chinh, quận Đống Đa, Hà Nội	dungtv@transer
3	TX11	Vũ Quang Đạt	CN4	Công ty Cổ phần Vận tải dịch vụ Liên Ninh	1979-11-23	Số 78, phố Tây Sơn, quận Đống Đa, Hà Nội	datvq@transer
4	TX12	Lý Xuân Trường	CN4	Công ty Cổ phần Vận tải dịch vụ Liên Ninh	1988-02-17	Số 45, đường Giải Phóng, quận Hoàng Mai, Hà Nội	truonglx@tran
5	TX13	Tạ Bình Thành	CN4	Công ty Cổ phần Vận tải dịch vụ Liên Ninh	1971-07-20	Số 13, phố Thái Hà, quận Đống Đa, Hà Nội	thangtd@trans
6	TX14	Lương Văn Hải	CN4	Công ty Cổ phần Vận tải dịch vụ Liên Ninh	1983-05-03	Số 29, phố Đội Cấn, quận Ba Đình, Hà Nội	haihv@transer
7	TX15	Nguyễn Hữu Phước	CN4	Công ty Cổ phần Vận tải dịch vụ Liên Ninh	1970-10-10	Số 60, đường Nguyễn Du, quận Hai Bà Trưng, Hà Nội	phuocngh@trans
8	TX16	Mai Văn Bảo	CN6	Công ty Cổ phần Vận tải dịch vụ Liên Ninh	1975-01-11	Số 99, phố Tràng Thi, quận Hoàn Kiếm, Hà Nội	baomv@transer
9	TX17	Tô Hồng Phúc	CN6	Công ty Cổ phần Vận tải dịch vụ Liên Ninh	1978-09-14	Số 12, đường Quang Trung, quận Hà Đông, Hà Nội	phucth@transer
10	TX18	Cao Ngọc Sơn	CN6	Công ty Cổ phần Vận tải dịch vụ Liên Ninh	1974-06-06	Số 47, phố Ngọc Khánh, quận Ba Đình, Hà Nội	socng@transer
11	TX19	Lâm Nhật Minh	CN6	Công ty Cổ phần Vận tải dịch vụ Liên Ninh	1982-08-22	Số 88, đường Lê Văn Lương, quận Thanh Xuân, Hà Nội	minhln@trans
12	TX2	Trần Dinh Hòa	CN1	Công ty Cổ phần Vận tải dịch vụ Liên Ninh	1975-07-23	Số 45, đường Hồ Tùng Mậu, quận Cầu Giấy, Hà Nội	hoatd@transer
13	TX20	Châu Bình Long	CN6	Công ty Cổ phần Vận tải dịch vụ Liên Ninh	1973-03-19	Số 22, phố Hàng Bông, quận Hoàn Kiếm, Hà Nội	longcd@transer
14	TX21	Hà Mạnh Đức	CN6	Công ty Cổ phần Vận tải dịch vụ Liên Ninh	1980-04-15	Số 37, đường Xuân Thủy, quận Cầu Giấy, Hà Nội	ducmh@transer
15	TX22	Triệu Quang Tùng	CN6	Công ty Cổ phần Vận tải dịch vụ Liên Ninh	1972-12-02	Số 19, đường Nguyễn Phong Sắc, quận Cầu Giấy, Hà Nội	tungtq@transer
16	TX23	Kiều Hải Nam	CN6	Công ty Cổ phần Vận tải dịch vụ Liên Ninh	1976-10-28	Số 55, đường Lê Duẩn, quận Hoàn Kiếm, Hà Nội	namkh@transer
17	TX24	Nguyễn Hữu Toản	CN6	Công ty Cổ phần Vận tải dịch vụ Liên Ninh	1981-07-09	Số 102, đường Phạm Hùng, quận Nam Từ Liêm, Hà Nội	toannh@transer
18	TX25	Hoàng Đình Lâm	CN6	Công ty Cổ phần Vận tải dịch vụ Liên Ninh	1979-02-25	Số 88, đường Nguyễn Khuyển, quận Đống Đa, Hà Nội	lamhd@transer

Ảnh 11. Kết quả truy vấn VIEW chi tiết tài xe.

Ví dụ 14: Tạo VIEW để theo dõi danh sách trạm xe và số tuyến đi qua.

Khung 20. Khởi tạo VIEW chi tiết trạm xe.

```

GO
CREATE VIEW view_CHITIET_TRAMXE AS
SELECT

```

```

tx.Ma_Tram, tx.Vi_Tri_Tram,
STRING_AGG(lt.Ma_Tuyen, ', ') WITHIN GROUP (ORDER BY
Ma_Tuyen) AS Tuyen_Di_Qua
FROM TRAMXE tx
INNER JOIN LOTRINH lt ON lt.Ma_Tram = tx.Ma_Tram
GROUP BY tx.Ma_Tram, tx.Vi_Tri_Tram;

```

Khung 21. Truy vấn VIEW chi tiết trạm xe.

```
SELECT * FROM view_CHITIET_TRAMXE
```

	Ma_Tram	Vi_Tri_Tram	Tuyen_Di_Qua
1	MT1	Bến xe Gia Lâm, số 9, đường Ngõ Gia Khảm, quận Long Biên, Hà Nội	BUS1, BUS2
2	MT10	Ga Hà Nội, số 120, đường Lê Duẩn, quận Hoàn Kiếm, Hà Nội	BUS1, BUS9
3	MT100	Điểm dừng xe buýt, ngõ 1072, đường Đề La Thành, quận Cầu Giấy, Hà Nội	BUS9
4	MT101	Điểm trung chuyển xe buýt Cầu Giấy - Thủ Lệ 01	BUS9
5	MT102	Số 148-150, đường Cầu Giấy, quận Cầu Giấy, Hà Nội	BUS9
6	MT103	Ga tàu điện Chùa Hè, quận Cầu Giấy, Hà Nội	BUS9
7	MT104	Số 378, đường Cầu Giấy, quận Cầu Giấy, Hà Nội	BUS9
8	MT105	Học viện Báo chí và Tuyên truyền, số 36, đường Xuân Thủy, quận Cầu Giấy, Hà Nội	BUS9
9	MT106	Chợ Xanh, quận Cầu Giấy, Hà Nội	BUS9
10	MT107	Đại học Sư phạm Hà Nội, số 136, đường Xuân Thủy, quận Cầu Giấy, Hà Nội	BUS9
11	MT108	Điểm dừng xe buýt, đối diện Đại học Thương mại, số 7...	BUS9
12	MT109	Điểm dừng xe buýt, chung cư A1 - ngân hàng Vietcombank, số 78-80A, phố Khâm Thiên, quận Đống Đa, Hà Nội	BUS9
13	MT11	Số 78-80A, phố Khâm Thiên, quận Đống Đa, Hà Nội	BUS1, BUS9
14	MT110	Khu đô thị Mỹ Đình II, quận Nam Từ Liêm, Hà Nội	BUS9
15	MT111	Khu đô thị Mỹ Đình I, quận Nam Từ Liêm, Hà Nội	BUS9
16	MT112	Điểm dừng xe buýt, tòa nhà CT2A, khu đô thị Xuân Phúc, xã Xuân Phúc, huyện Gia Lâm, Hà Nội	BUS9
17	MT113	Điểm dừng xe buýt, cầu vượt Xuân Phượng, quận Nam Từ Liêm, Hà Nội	BUS9
18	MT114	Nhà máy Bia Sài Gòn, quận Nam Từ Liêm, Hà Nội	BUS9
19	MT115	Chùa Xuân Phương, xã Xuân Phương, huyện Nam Từ Liêm, Hà Nội	BUS9

Ảnh 12. Kết quả truy vấn VIEW chi tiết trạm xe.

Ví dụ 15: Tạo VIEW để theo dõi số tuổi của tài xế.

Khung 22. Khởi tạo VIEW theo dõi tuổi tài xế.

```

GO
CREATE VIEW view_TUOI_TAIXE AS
SELECT
    Ma_Tai_Xe, Ho_Va_Ten, Ngay_Sinh,
    YEAR(GETDATE()) - YEAR(Ngay_Sinh) AS Tuoi_Tai_Xe
FROM TAIXE

```

Khung 23. Truy vấn VIEW theo dõi tuổi tài xế.

GO

SELECT * FROM view_TUOI_TAIXE

	Ma_Tai_Xe	Ho_Va_Ten	Ngay_Sinh	Tuoi_Tai_Xe
1	TX1	Nguyễn Văn Hùng	1978-05-12	47
2	TX10	Trịnh Văn Dũng	1972-09-05	53
3	TX11	Vũ Quang Đạt	1979-11-23	46
4	TX12	Lý Xuân Trường	1988-02-17	45
5	TX13	Tạ Đình Thủ	1971-07-20	54
6	TX14	Lương Văn Hải	1983-05-03	42
7	TX15	Nguyễn Hữu Phước	1970-10-10	55
8	TX16	Mai Văn Bảo	1975-01-11	50
9	TX17	Tô Hồng Phúc	1978-09-14	47
10	TX18	Cao Ngọc Sơn	1974-06-06	51
11	TX19	Lâm Nhật Minh	1982-08-22	43
12	TX2	Trần Bình Hòa	1975-07-23	50
13	TX20	Châu Bình Long	1973-03-19	52
14	TX21	Hà Mạnh Đức	1980-04-15	45
15	TX22	Triệu Quang Tùng	1972-12-02	53
16	TX23	Kiều Hải Nam	1976-10-28	49
17	TX24	Nguyễn Hữu Toàn	1981-07-09	44
18	TX25	Hoàng Bình Lâm	1979-02-25	46
19	TX3	Phạm Văn Cường	1975-04-30	50

Ảnh 13. Kết quả truy vấn VIEW theo dõi tuổi tài xế.

Ví dụ 16: Tạo VIEW để theo dõi số năm còn có thể lao động của tài xế. Biết rằng công ty chỉ cho phép tài xế lao động đến hết năm 55 tuổi.

Khung 24. Khởi tạo VIEW theo dõi số năm còn có thể lao động của tài xế.

GO

CREATE VIEW view_TUOI_NGHIHUU_TAIXE AS

SELECT

```

    Ma_Tai_Xe, Ho_Va_Ten, Ngay_Sinh,
    YEAR(GETDATE()) - YEAR(Ngay_Sinh) AS Tuoi_Tai_Xe,
    STRING_AGG(
        CAST(DATEDIFF(MONTH, GETDATE(), DATEADD(YEAR, 55,
        Ngay_Sinh)) / 12 AS NVARCHAR)
        + N' năm ' +
    )

```

```

        CAST((DATEDIFF(MONTH, GETDATE(), DATEADD(YEAR, 55,
Ngay_Sinh)) % 12) AS NVARCHAR)
        + N' tháng ' +
        CAST(DATEDIFF(DAY, GETDATE(), DATEADD(YEAR, 55,
Ngay_Sinh)) % 30 AS NVARCHAR)
        + N' ngày', ''
    ) AS Thoi_Gian_Lam_Viec_Con_Lai
FROM TAIXE
GROUP BY Ma_Tai_Xe, Ho_Va_Ten, Ngay_Sinh;

```

Khung 25. Truy vấn VIEW theo dõi số năm còn có thể lao động của tài xế.

GO

SELECT * FROM view_TUOI_NGHIHUU_TAIXE;

	Ma_Tai_Xe	Ho_Va_Ten	Ngay_Sinh	Tuoi_Tai_Xe	Thoi_Gian_Lam_Viec_Con_Lai
1	TX1	Nguyễn Văn Hùng	1978-05-12	47	8 năm 2 tháng 6 ngày
2	TX10	Trịnh Văn Dũng	1972-09-05	53	2 năm 6 tháng 0 ngày
3	TX11	Vũ Quang Đạt	1979-11-23	46	9 năm 8 tháng 26 ngày
4	TX12	Lý Xuân Trường	1988-02-17	45	9 năm 11 tháng 22 ngày
5	TX13	Tạ Bình Thành	1971-07-20	54	1 năm 4 tháng 8 ngày
6	TX14	Lương Văn Hải	1983-05-03	42	13 năm 2 tháng 23 ngày
7	TX15	Nguyễn Hữu Phước	1970-10-10	55	0 năm 7 tháng 25 ngày
8	TX16	Mai Văn Bảo	1975-01-11	50	4 năm 10 tháng 19 ngày
9	TX17	Tô Hồng Phúc	1978-09-14	47	8 năm 6 tháng 11 ngày
10	TX18	Cao Ngọc Sơn	1974-06-06	51	4 năm 3 tháng 10 ngày
11	TX19	Lâm Nhật Minh	1982-08-22	43	12 năm 5 tháng 9 ngày
12	TX2	Trần Dinh Hòa	1975-07-23	50	5 năm 4 tháng 2 ngày
13	TX20	Châu Bình Long	1973-03-19	52	3 năm 0 tháng 16 ngày
14	TX21	Hà Mạnh Đức	1988-04-15	45	10 năm 1 tháng 19 ngày
15	TX22	Triệu Quang Tùng	1972-12-02	53	2 năm 9 tháng 28 ngày
16	TX23	Kiều Hải Nam	1976-10-28	49	6 năm 7 tháng 14 ngày
17	TX24	Nguyễn Hữu Toản	1981-07-09	44	11 năm 4 tháng 20 ngày
18	TX25	Hoàng Đình Lâm	1979-02-25	46	8 năm 11 tháng 25 ngày
19	TX3	Phạm Văn Cường	1975-04-30	50	5 năm 1 tháng 8 ngày

Ảnh 14. Kết quả thực hiện truy vấn VIEW theo dõi số năm còn có thể lao động của tài xế.

Ví dụ 17: Tạo VIEW để theo dõi số năm còn có thể sử dụng của phương tiện.

Biết rằng luật pháp Việt Nam cho phép xe chở khách có niên hạn sử dụng tối đa là 20 năm.

Khung 26. Khởi tạo VIEW theo dõi niên hạn của phuong tiện.

```

GO
CREATE VIEW view_NIENHAN_PHUONGTIEN AS
SELECT
    Ma_Phuong_Tien, Ten_Phuong_Tien, Bien_Kiem_Soat,
    Hang_San_Xuat,
    Nam_San_Xuat,
    YEAR(GETDATE()) - Nam_San_Xuat AS
    So_Nam_Da_Su_Dung,
    20 - (YEAR(GETDATE()) - Nam_San_Xuat) AS
    So_Nam_Con_Co_The_Su_Dung
FROM PHUONGTIEN;

```

Khung 27. Truy vấn VIEW theo dõi niên hạn của phuong tiện.

```

GO
SELECT * FROM view_NIENHAN_PHUONGTIEN

```

	Ma_Phuong_Tien	Ten_Phuong_Tien	Bien_Kiem_Soat	Hang_San_Xuat	Nam_San_Xuat	So_Nam_Da_Su_Dung	So_Nam_Con_Co_The_Su_Dung
1	VH1	Sanco City D60	29C - 891.12	Sanco	2020	5	15
2	VH10	Thaco Garden 79CT	29B - 894.67	Thaco	2021	4	16
3	VH11	Sanco City 1.55	29B - 372.18	Sanco	2023	2	18
4	VH12	Sanco City 1.55	29B - 588.92	Sanco	2023	2	18
5	VH13	Sanco City 1.55	29B - 431.76	Sanco	2023	2	18
6	VH14	Sanco City 1.55	29B - 268.34	Sanco	2023	2	18
7	VH15	Sanco City 1.55	29B - 983.21	Sanco	2023	2	18
8	VH16	Sanco City 1.40	29B - 147.85	Sanco	2021	4	16
9	VH17	Sanco City 1.40	29B - 694.52	Sanco	2021	4	16
10	VH18	Sanco City 1.40	29B - 859.13	Sanco	2021	4	16
11	VH19	Sanco City 1.40	29B - 236.78	Sanco	2021	4	16
12	VH2	Sanco City D60	29B - 937.58	Sanco	2020	5	15
13	VH20	Sanco City 1.40	29B - 471.89	Sanco	2021	4	16
14	VH21	Thaco Meadow 89CT	29B - 325.64	Thaco	2024	1	19
15	VH22	Thaco Meadow 89CT	29B - 789.48	Thaco	2024	1	19
16	VH23	Thaco Meadow 89CT	29B - 852.31	Thaco	2024	1	19
17	VH24	Thaco Meadow 89CT	29B - 418.27	Thaco	2024	1	19
18	VH25	Thaco Meadow 89CT	29B - 967.53	Thaco	2024	1	19
19	VH4	Sanco City D60	29B - 715.36	Sanco	2020	5	15

Ảnh 15. Kết quả thực hiện truy vấn VIEW niên hạn của phuong tiện.

Ví dụ 18: Tạo VIEW để theo dõi dữ liệu liên hệ của các chi nhánh.

Khung 28. Khởi tạo VIEW theo dõi dữ liệu liên hệ của các chi nhánh.

```

GO

```

```

CREATE VIEW view_LIENHE_CHINHANH AS
SELECT
    cn.Ma_Chi_Nhanh, cn.Ten_Chi_Nhanh, cn.Dia_Chi,
    STRING_AGG(CASE WHEN cnlh.Loai_Lien_He_Email_SDT =
'Email' THEN cnlh.Gia_Tri END, ', ') AS Email,
    STRING_AGG(CASE WHEN cnlh.Loai_Lien_He_Email_SDT = N'Số
điện thoại' THEN cnlh.Gia_Tri END, ', ') AS So_Dien_Thoai
FROM CHINHANH cn
INNER JOIN CHINHANH_LIENHE cnlh ON cn.Ma_Chi_Nhanh =
cnlh.Ma_Chi_Nhanh
GROUP BY cn.Ma_Chi_Nhanh, cn.Ten_Chi_Nhanh, cn.Dia_Chi;

```

Khung 29. Truy vấn VIEW theo dõi dữ liệu liên hệ của các chi nhánh.

GO

SELECT * FROM view_LIENHE_CHINHANH;

	Ma_Chi_Nhanh	Ten_Chi_Nhanh	Dia_Chi	Email	So_Dien_Thoai
1	CN1	Xí nghiệp Bus Hà Nội	Số 29, phố Lạc Trung, quận Hai Bà Trưng, Hà Nội	xnbuythanoi@gmail.com	0243 821 2816
2	CN10	Công ty Cổ phần Vận tải Newway	Khu Kim Ngưu II, phường Hoàng Mai, quận Hoàng Mai, H...	cty_newway@transerco.com.vn	0243 565 4898
3	CN2	Xí nghiệp xe buýt 10-10 Hà Nội	Đường Trần Vỹ, Cầu Giấy, Hà Nội	xn_bus10@transerco.com.vn	0243 558 4673
4	CN3	Xí nghiệp xe buýt Cầu Bươu	Thôn Cầu Bươu, huyện Thanh Trì, Hà Nội	xn_buscaubuoou@transerco.com.vn	0243 268 6696
5	CN4	Xí nghiệp xe buýt Yên Viên	Thôn Lã Cối, thị trấn Yên Viên, huyện Gia Lâm, Hà Nội	xn_busyv@transerco.com.vn	0243 521 1999
6	CN5	Xí nghiệp xe buýt nhanh BRT Hà Nội	Tầng 2, số 1, đường Kim Mã, quận Ba Đình, Hà Nội	xn_brt@transerco.com.vn	0243 733 6262
7	CN6	Xí nghiệp Xe khách Nam Hà Nội	Khu Đền Lừ 1, quận Hoàng Mai, Hà Nội	xn_xkn@transerco.com.vn	0243 858 4362
8	CN7	Trung tâm Tân Đạt	Đường Phan Trọng Tuệ, huyện Thanh Trì, Hà Nội	tandat@transerco.com.vn	0243 793 8696
9	CN8	Công ty Cổ phần Vận tải dịch vụ Liên Ninh	Km 15+200, quốc lộ 1A, thôn Yên Phú, huyện Thanh Trì...	lienninh@transerco.com.vn	0243 686 6506
10	CN9	Công ty Cổ phần Xe điện Hà Nội	Số 454, đường Phạm Văn Đồng, quận Bắc Từ Liêm, Hà Nội	vanthuxdhn@gmail.com	0243 752 2222

Ảnh 16. Kết quả sau khi thực hiện truy vấn VIEW theo dõi liên hệ của các chi nhánh.

Ví dụ 19: Tạo VIEW để theo dõi tổng lượng xe mà mỗi chi nhánh quản lý.

Khung 30. Khởi tạo VIEW theo dõi tổng lượng xe mà mỗi chi nhánh quản lý

```
GO  
CREATE VIEW view_PHUONGTIEN_CHINHANH AS  
SELECT  
    cn.Ma_Chi_Nhanh, cn.Ten_Chi_Nhanh,  
    COUNT(DISTINCT pt.Ma_Phuong_Tien) AS So_Phuong_Tien  
FROM PHUONGTIEN pt  
INNER JOIN TAIXE tx ON tx.Ma_Tai_Xe = pt.Ma_Tai_Xe  
INNER JOIN CHINHANH cn ON tx.Ma_Chi_Nhanh = cn.Ma_Chi_Nhanh  
GROUP BY cn.Ma_Chi_Nhanh, cn.Ten_Chi_Nhanh;
```

Khung 31. Truy vấn VIEW theo dõi tổng lượng xe mà mỗi chi nhánh quản lý.

```
GO  
SELECT * FROM view_PHUONGTIEN_CHINHANH;
```

The screenshot shows the SQL Server Management Studio interface. In the top pane, there are two tabs: 'SQLQuery_1 - (63)...DB (sa)' and 'SQLQuery_1.sql - (51)...DB (sa)'. The bottom pane shows the results of the query:

	Ma_Chi_Nhanh	Ten_Chi_Nhanh	So_Phuong_Tien
1	CN1	Xí nghiệp Bus Hà Nội	9
2	CN4	Xí nghiệp xe buýt Yên Viên	5
3	CN6	Xí nghiệp Xe khách Nam Hà Nội	10

Ảnh 17. Kết quả sau khi thực hiện truy vấn VIEW theo dõi tổng lượng xe mà mỗi chi nhánh quản lý.

4.2. Tạo INDEX cần thiết cho các bảng

Khung 32. Khởi tạo chỉ mục (INDEX).

```
-- Tạo chỉ mục thông tin của tuyến xe  
GO
```

```
CREATE INDEX idx_LOTRINH_TUYEN
ON L0TRINH(Ma_Tuyen);

-- Tạo chỉ mục đảm bảo giá trị của biến kiểm soát là duy
-- nhất
GO
CREATE UNIQUE INDEX idx_BIENKIEMSOAT_PHUONGTIEN
ON PHUONGTIEN(Bien_Kiem_Soat);

-- Tạo chỉ mục đảm bảo 1 phương tiện chỉ được giao cho 1
-- tài xế
GO
CREATE UNIQUE INDEX idx_TAIXE_PHUONGTIEN
ON PHUONGTIEN(Ma_Tai_Xe);

-- Tạo chỉ mục tìm phương tiện theo tuyến xe
GO
CREATE INDEX idx_PHUONGTIEN_TUYEN
ON PHUONGTIEN(Ma_Tuyen);

-- Tạo chỉ mục tìm tuyến xe theo chi nhánh
GO
CREATE INDEX idx_TUYENXE_CHINHANH
ON TUYENXE(Ma_Chi_Nhanh);

-- Tạo chỉ mục tìm tuyến xe theo trạm khởi đầu
GO
CREATE INDEX idx_TRAMKHOIDAU_TUYENXE
ON TRAMKHOIDAU(Ma_Tuyen);

-- Tạo chỉ mục tìm tuyến xe theo trạm kết thúc
GO
```

```

CREATE INDEX idx_TRAMKETTHUC_TUYENXE
ON TRAMKETTHUC(Ma_Tuyen);

-- Tạo chỉ mục tìm thông tin liên hệ của tài xế
GO
CREATE INDEX idx_LIENHE_TAIXE
ON TAIXE_LIENHE(Ma_Tai_Xe);

-- Tạo chỉ mục tìm thông tin của tài xế
GO
CREATE INDEX idx_THONGTIN_TAIXE
ON TAIXE(Ma_Tai_Xe);

-- Tạo chỉ mục tìm thông tin liên hệ của chi nhánh
GO
CREATE INDEX idx_LIENHE_CHINHANH
ON CHINHANH_LIENHE(Ma_Chi_Nhanh);

-- Tạo chỉ mục tìm thông tin của chi nhánh
GO
CREATE INDEX idx_THONGTIN_CHINHANH
ON CHINHANH(Ma_Chi_Nhanh);

```

4.3. Tạo STORED PROCEDURE

4.3.1. STORED PROCEDURE không tham số

Ví dụ 20: Tạo các STORED PROCEDURE không tham số thực hiện các yêu cầu sau:

- a) Lấy danh sách tất cả các tài xế.
- b) Lấy danh sách tất cả các phương tiện.
- c) Lấy danh sách tất cả các chi nhánh.
- d) Lấy danh sách theo dõi tuổi nghỉ hưu của tài xế.

Khung 33. Khởi tạo STORED PROCEDURE không tham số.

```
-- a) Lấy danh sách tất cả các tài xế
```

```

GO
CREATE PROCEDURE sp_DANHSACH_TAIXE
AS
BEGIN
    SELECT * FROM TAIXE;
END;

-- b) Lấy danh sách tất cả các phương tiện
GO
CREATE PROCEDURE sp_DANHSACH_PHUONGTIEN
AS
BEGIN
    SELECT * FROM PHUONGTIEN;
END;

-- c) Lấy danh sách tất cả các chi nhánh
GO
CREATE PROCEDURE sp_DANHSACH_CHINHANH
AS
BEGIN
    SELECT * FROM CHINHANH;
END;

-- d) Lấy danh sách theo dõi tuổi nghỉ hưu của các tài xế
GO
CREATE PROCEDURE sp_DANHSACH_TUOI_NGHIHUU
AS
BEGIN
    SELECT * FROM view_TUOI_NGHIHUU_TAIXE;
END;

```

Khung 34. Cấu trúc cơ bản truy vấn STORED PROCEDURE.

<code>EXEC tên_Stored_Procedure;</code>

The screenshot displays three separate SSMS windows, each showing the results of a different stored procedure:

- Window 1 (Left):** Results of `sp_DANHSACH_TAIXE_CHINHANH`. The table has columns: Ma_Tai_Xe, Ma_Chi_Nhanh, Ma_Tuyen, Ngay_Xuat_Kinh, and Tinh_Quoc.
- Window 2 (Middle):** Results of `sp_DANHSACH_PHUONGTIEN`. The table has columns: Ma_Phuong_Tien, Ma_Tuyen, Ma_Tai_Xe, Ten_Phuong_Tien, Bien_Kiem_Soat, Hang_San_Xuat, Nam_San_Xuat, So_Khung, and So_Chi_Nguoi.
- Window 3 (Right):** Results of `sp_DANHSACH_TUYEN`. The table has columns: Ma_Phuong_Tien, Ma_Tuyen, Ma_Tai_Xe, Ten_Phuong_Tien, Bien_Kiem_Soat, Hang_San_Xuat, Nam_San_Xuat, So_Khung, and So_Chi_Nguoi.

Ảnh 18. Kết quả sau khi thực hiện truy vấn STORED PROCEDURE.

4.3.2. STORED PROCEDURE có tham số

Ví dụ 21: Tạo các STORED PROCEDURE có tham số thực hiện các yêu cầu sau:

- Lấy danh sách tất cả các tài xế theo chi nhánh.
- Lấy danh sách tất cả các phương tiện theo hãng sản xuất.
- Lấy danh sách tất cả phương tiện theo tuyến.
- Lấy danh sách tất cả tài xế theo tuyến.
- Lấy danh sách tất cả trạm dừng theo tuyến.

Khung 35. Khởi tạo truy vấn STORED PROCEDURE có tham số.

-- a) Lấy danh sách tài xế theo chi nhánh

GO

CREATE PROCEDURE sp_DANHSACH_TAIXE_CHINHANH

@Ma_Chi_Nhanh NVARCHAR(10)

AS

BEGIN

SELECT * FROM TAIXE WHERE Ma_Chi_Nhanh = @Ma_Chi_Nhanh;

END;

```

-- b) Lấy danh sách phương tiện theo hãng sản xuất
GO
CREATE PROCEDURE sp_DANHSACH_PHUONGTIEN_HANGSX
    @Hang_San_Xuat NVARCHAR(50)
AS
BEGIN
    SELECT
        Ma_Phuong_Tien, Ten_Phuong_Tien,
        Bien_Kiem_Soat, Hang_San_Xuat,
        Nam_San_Xuat, So_Khung, So_Cho_Ngoi
    FROM PHUONGTIEN WHERE Hang_San_Xuat = @Hang_San_Xuat;
END;

-- c) Lấy danh sách phương tiện theo tuyến
GO
CREATE PROCEDURE sp_DANHSACH_PHUONGTIEN_TUYENXE_MATUYEN
    @Ma_Tuyen NVARCHAR(10)
AS
BEGIN
    SELECT
        Ma_Phuong_Tien, Ten_Phuong_Tien,
        Bien_Kiem_Soat, Hang_San_Xuat,
        Nam_San_Xuat, So_Khung, So_Cho_Ngoi
    FROM PHUONGTIEN WHERE Ma_Tuyen = @Ma_Tuyen;
END;

-- d) Lấy danh sách tài xế theo tuyến
GO
CREATE PROCEDURE sp_DANHSACH_TAIXE_TUYEN_MATUYEN
    @Ma_Tuyen NVARCHAR(10)
AS
BEGIN

```

```

SELECT
    tx.Ma_Tai_Xe, pt.Ma_Tuyen,
    tx.Ho_Va_Ten, tx.Dia_Chi,
    STRING_AGG(CASE WHEN txlh.Loai_Lien_He_Email_SDT =
'Email' THEN txlh.Gia_Tri END, ', ') AS Email,
    STRING_AGG(CASE WHEN txlh.Loai_Lien_He_Email_SDT =
N'Số điện thoại' THEN txlh.Gia_Tri END, ', ') AS
So_Dien_Thoai
    FROM TAIXE tx
    INNER JOIN TAIXE LIENHE txlh ON tx.Ma_Tai_Xe =
txlh.Ma_Tai_Xe
    INNER JOIN PHUONGTIEN pt ON tx.Ma_Tai_Xe =pt.Ma_Tai_Xe
    WHERE pt.Ma_Tuyen = @Ma_Tuyen
    GROUP BY tx.Ma_Tai_Xe, pt.Ma_Tuyen, tx.Ho_Va_Ten,
tx.Dia_Chi;
END;

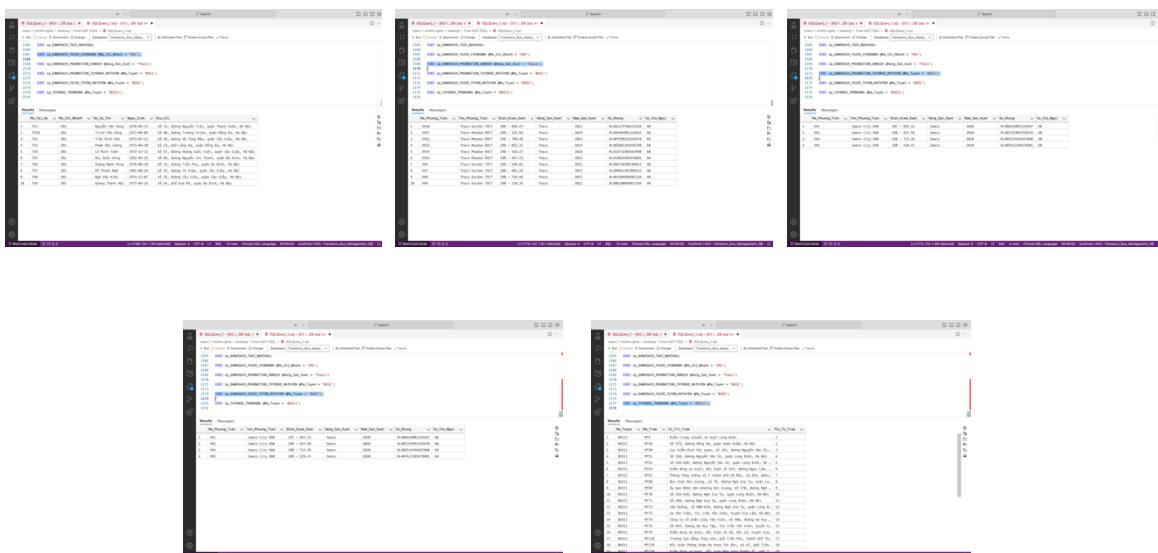
-- e) Lấy danh sách trạm dừng theo tuyến
GO
CREATE PROCEDURE sp_TUYENXE_TRAMDUNG
    @Ma_Tuyen NVARCHAR(10)
AS
BEGIN
    SELECT
        lt.Ma_Tuyen, lt.Ma_Tram,
        tx.Vi_Tri_Tram, lt.Thu_Tu_Tram
    FROM LOTRINH lt
    INNER JOIN TRAMXE tx ON lt.Ma_Tram = tx.Ma_Tram
    WHERE Ma_Tuyen = @Ma_Tuyen;
END;

```

Khung 36. Truy vấn STORED PROCEDURE có tham số.

<code>EXEC sp_DANHSACH_TAIXE_CHINHANH @Ma_Chi_Nhanh = 'CN1';</code>

```
EXEC sp_DANHSACH_PHUONGTIEN_HANGSX @Hang_San_Xuat =  
'Thaco';  
  
EXEC sp_DANHSACH_PHUONGTIEN_TUYENXE_MATUYEN @Ma_Tuyen =  
'BUS1';  
  
EXEC sp_DANHSACH_TAIXE_TUYEN_MATUYEN @Ma_Tuyen = 'BUS2';  
  
EXEC sp_TUYENXE_TRAMDUNG @Ma_Tuyen = 'BUS11';
```



Ảnh 19. Kết quả sai khi thực hiện STORED PROCEDURE có tham số.

4.3.3. STORED PROCEDURE có đầu ra

Ví dụ 22: Tạo các STORED PROCEDURE có đầu ra thực hiện các yêu cầu sau:

- a) Lấy số lượng trạm dừng của một tuyến xe.
 - b) Lấy thông tin tài xế.

Khung 37. Khởi tạo STORED PROCEDURE có đầu ra

-- a) Lấy số lượng trạm dừng của một tuyến xe

GO

```
CREATE PROCEDURE sp_TONG_TBAMDUONG
```

@Ma Tuyen NVABCHAB (10) -

```

@So_Luong_Tram_Dung INT OUTPUT
AS
BEGIN
    SELECT @So_Luong_Tram_Dung = COUNT(*)
    FROM L0TRINH WHERE Ma_Tuyen = @Ma_Tuyen;
END;

-- b) Lấy thông tin tài xế
GO
CREATE PROCEDURE sp LIENHE_TAIXE
    @Ma_Tai_Xe NVARCHAR(10),
    @Ho_Ten NVARCHAR(100) OUTPUT,
    @So_Dien_Thoai NVARCHAR(15) OUTPUT,
    @Email NVARCHAR(100) OUTPUT
AS
BEGIN
    SELECT
        @Ma_Tai_Xe = tx.Ma_Tai_Xe,
        @Ho_Ten = tx.Ho_Va_Ten,
        @So_Dien_Thoai = STRING_AGG(CASE WHEN
txlh.Loai_Lien_He_Email_SDT = N'Số điện thoại' THEN
txlh.Gia_Tri END, ', '),
        @Email = STRING_AGG(CASE WHEN
txlh.Loai_Lien_He_Email_SDT = 'Email' THEN txlh.Gia_Tri
END, ', ')
    FROM TAIXE tx
    INNER JOIN TAIXE_LIENHE txlh ON tx.Ma_Tai_Xe =
txlh.Ma_Tai_Xe
    WHERE @Ma_Tai_Xe = tx.Ma_Tai_Xe
    GROUP BY tx.Ma_Tai_Xe, tx.Ho_Va_Ten;
END;

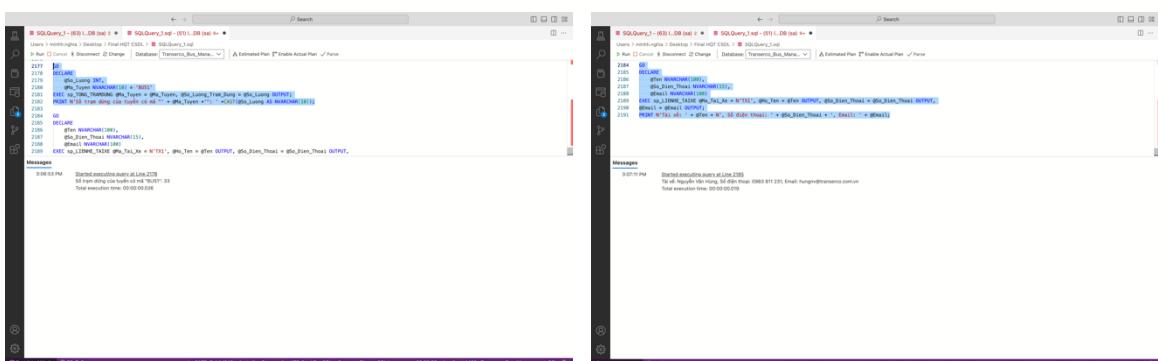
```

Khung 38. Truy vấn STORED PROCEDURE có đầu ra.

```

GO
DECLARE
    @So_Luong INT,
    @Ma_Tuyen NVARCHAR(10) = 'BUS1'
EXEC sp_TONG_TRAMDUNG @Ma_Tuyen = @Ma_Tuyen,
@So_Luong_Tram_Dung = @So_Luong OUTPUT;
PRINT N'Số trạm dừng của tuyến có mã "' + @Ma_Tuyen + '"';
+CAST(@So_Luong AS NVARCHAR(10));
GO
DECLARE
    @Ten NVARCHAR(100),
    @So_Dien_Thoai NVARCHAR(15),
    @Email NVARCHAR(100)
EXEC sp_LIENHE_TAIXE @Ma_Tai_Xe = N'TX1', @Ho_Ten = @Ten
OUTPUT, @So_Dien_Thoai = @So_Dien_Thoai OUTPUT,
@email = @Email OUTPUT;
PRINT N'Tài xế: ' + @Ten + N', Số điện thoại: ' +
@So_Dien_Thoai + ', Email: ' + @Email;

```



Ảnh 20. Kết quả sau khi thực hiện truy vấn STORED PROCEDURE có đầu ra.

4.4. Tạo các FUNCTION

4.4.1. FUNCTION trả về kiểu vô hướng

Ví dụ 23: Tạo các FUNCTION trả về kiểu vô hướng thực hiện các yêu cầu sau:

- a) Tính tổng số trạm dừng của 1 tuyến.
- b) Tính tổng số tài xế đang làm việc trong hệ thống.

Khung 39. Khởi tạo FUNCTION trả về kiểu vô hướng

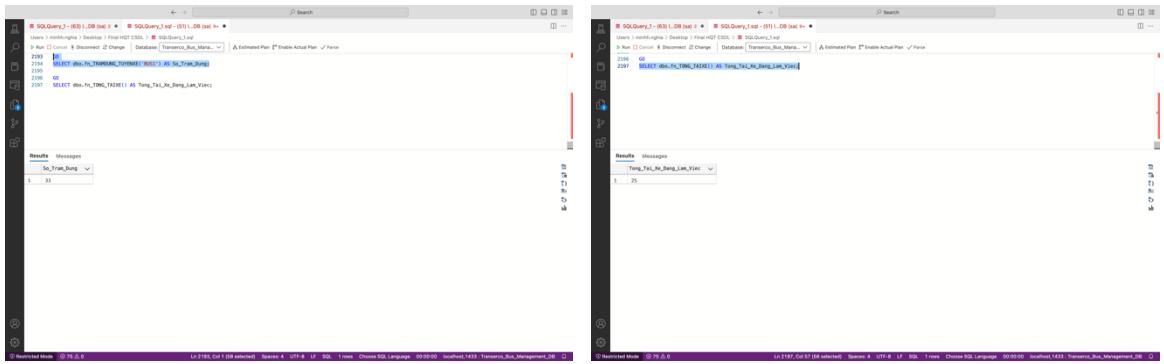
```
-- a) Tính tổng số trạm dừng của 1 tuyến
GO
CREATE FUNCTION fn_TRAMDUNG_TUYENXE (@Ma_Tuyen
NVARCHAR(10))
RETURNS INT
AS
BEGIN
    DECLARE @So_Tram INT;
    SELECT @So_Tram = COUNT(*) FROM LOTRINH WHERE Ma_Tuyen
= @Ma_Tuyen;
    RETURN @So_Tram;
END;

-- b) Tính tổng số tài xế đang làm việc trong hệ thống
GO
CREATE FUNCTION fn_TONG_TAIXE ()
RETURNS INT
AS
BEGIN
    DECLARE @Tong_Tai_Xe INT;
    SELECT @Tong_Tai_Xe = COUNT(*) FROM TAIXE;
    RETURN @Tong_Tai_Xe;
END;
```

Khung 40. Truy vấn FUNCTION trả về kiểu vô hướng.

```
GO
SELECT dbo.fn_TRAMDUNG_TUYENXE('BUS1') AS So_Tram_Dung;

GO
SELECT dbo.fn_TONG_TAIXE() AS Tong_Tai_Xe_Dang_Lam_Viec;
```



Ảnh 21. Kết quả sau khi thực hiện truy vấn FUNCTION trả về kiểu vô hướng.

4.4.2. FUNCTION trả về kiểu bảng

Ví dụ 24: Tạo các FUNCTION trả về kiểu bảng thực hiện các yêu cầu sau:

- Lấy dữ liệu số điện thoại của tài xế trong hệ thống.
- Lấy dữ liệu email của tài xế trong hệ thống.
- Lấy dữ liệu các tuyến xe.

Khung 41. Khởi tạo FUNCTION trả về kiểu bảng.

```
-- a) Lấy dữ liệu số điện thoại của tài xế trong hệ thống
```

```
GO
```

```
CREATE FUNCTION fn_SDT_TAIXE ()
RETURNS TABLE
AS
RETURN
(SELECT
    tx.Ma_Tai_Xe, tx.Ho_Va_Ten,
    STRING_AGG(txlh.Gia_Tri, ', ') AS So_Dien_Thoai
    FROM TAIXE tx
    INNER JOIN TAIXE_LIENHE txlh ON tx.Ma_Tai_Xe =
    txlh.Ma_Tai_Xe
    WHERE txlh.Loai_Lien_He_Email_SDT = N'Số điện thoại'
    GROUP BY tx.Ma_Tai_Xe, tx.Ho_Va_Ten
);
```

```
-- b) Lấy dữ liệu email của tài xế trong hệ thống
```

```
GO
```

```

CREATE FUNCTION fn_EMAIL_TAIXE ()
RETURNS TABLE
AS
RETURN
(SELECT
    tx.Ma_Tai_Xe, tx.Ho_Va_Ten,
    STRING_AGG(txlh.Gia_Tri, ', ') AS Email
    FROM TAIXE tx
    INNER JOIN TAIXE_LIENHE txlh ON tx.Ma_Tai_Xe =
txlh.Ma_Tai_Xe
    WHERE txlh.Loai_Lien_He_Email_SDT = 'Email'
    GROUP BY tx.Ma_Tai_Xe, tx.Ho_Va_Ten
);

-- c) Lấy dữ liệu các tuyến xe
GO
CREATE FUNCTION fn_THONGTIN_TUYENXE ()
RETURNS TABLE
AS
RETURN
(SELECT
    tx.Ma_Tuyen, tx.Ten_Tuyen,
    tx.Ma_Chi_Nhanh, cn.Ten_Chi_Nhanh,
    tx.Gio_Bat_Dau, tramkd.Vi_Tri_Tram AS Tram_Khoi_Dau,
    tx.Gio_Ket_Thuc, tramkt.Vi_Tri_Tram AS Tram_Ket_Thuc
    FROM TUYENXE tx
    INNER JOIN CHINHANH cn ON tx.Ma_Chi_Nhanh =
cn.Ma_Chi_Nhanh
    INNER JOIN TRAMKHOIDAU kd ON tx.Ma_Tuyen = kd.Ma_Tuyen
    INNER JOIN TRAMKETTHUC kt ON tx.Ma_Tuyen = kt.Ma_Tuyen
    INNER JOIN TRAMXE tramkd ON kd.Ma_Tram = tramkd.Ma_Tram
    INNER JOIN TRAMXE tramkt ON kt.Ma_Tram = tramkt.Ma_Tram
);

```

Khung 42. Truy vấn FUNCTION trả về kiểu bảng.

```

GO
SELECT * FROM dbo.fn_SDT_TAIXE();

GO
SELECT * FROM dbo.fn_EMAIL_TAIXE();

GO
SELECT * FROM dbo.fn_THONGTIN_TUYENXE();

```

Ma_Tai_Xe	Ho_Vi_Ten	So_Dien_Theo
TX1	Nguyễn Văn Hùng	0903 831 231
TX08	Trần Văn Đông	0903 831 231
TX11	Võ Quang Sát	0903 567 098
TX12	Lê Hữu Truyền	0906 789 812
TX13	Tạ Khanh Thảo	0903 888 111
TX14	Đỗ Văn Anh	0903 888 777
TX15	Nguyễn Thị Phượng	0974 323 456
TX16	Mai Văn Bé	0903 888 111
TX17	Trần Văn Hùng	0903 887 454
TX18	Cao Ngọc Sơn	0903 765 432
TX19	Lâm Nhật Minh	0906 541 148
TX20	Trần Văn Hùng	0906 541 432
TX21	Chử Bình Long	0906 878 543
TX22	Hà Mạnh Hồi	0905 678 981
TX23	Trần Văn Hùng	0903 888 111
TX24	Kiều Mai Nhung	0977 654 321
TX25	Nguyễn Hồi Tám	0907 789 654
TX26	Nguyễn Văn Linh	0906 431 389

Ma_Tai_Xe	Ho_Vi_Ten	Email
TX08	Trần Văn Đông	donghangtranserco.com.vn
TX11	Võ Quang Sát	datviettranserco.com.vn
TX12	Lê Hữu Truyền	luuthientranhco.com.vn
TX13	Tạ Khanh Thảo	thang08transerco.com.vn
TX14	Mai Văn Bé	haibinhtranhco.com.vn
TX15	Nguyễn Thị Phượng	thanhbinhtranhco.com.vn
TX16	Mai Văn Bé	haibinhtranhco.com.vn
TX17	Trần Văn Hùng	donghangtranserco.com.vn
TX18	Cao Ngọc Sơn	datviettranserco.com.vn
TX19	Lâm Nhật Minh	luuthientranhco.com.vn
TX20	Chử Bình Long	thang08transerco.com.vn
TX21	Hà Mạnh Hồi	haibinhtranhco.com.vn
TX22	Trần Văn Hùng	luuthientranhco.com.vn
TX23	Kiều Mai Nhung	thang08transerco.com.vn
TX24	Nguyễn Hồi Tám	datviettranserco.com.vn
TX25	Nguyễn Văn Linh	luuthientranhco.com.vn
TX26	Nguyễn Bình Lập	thang08transerco.com.vn

Ma_Tuyen	Ten_Tuyen	Ma_Chi_Nhanh	Ten_Chi_Nhanh	Loai_Tuyen	Trao_Ket_Bo	Trao_Ket_Ba
B01	Đèn Rồng	HN	Trung tâm kinh doanh xe KIA HN	Đèn Rồng	0903 888 111	bán xe
B02	03A	CN1	Xe ô tô KIA K5	Đèn Rồng	0903 888 111	bán xe
B03	42	CN2	Xe ô tô KIA K5	Đèn Rồng	0903 888 111	bán xe
B04	49	CN3	Xe ô tô KIA K5	Đèn Rồng	0903 888 111	bán xe
B05	54	CN4	Xe ô tô KIA K5	Đèn Rồng	0903 888 111	bán xe

Ảnh 22. Kết quả sau khi thực hiện truy vấn FUNCTION trả về kiểu bảng.

4.4.3. FUNCTION sử dụng biến bảng

Ví dụ 25: Tạo các FUNCTION sử dụng biến bảng thực hiện các yêu cầu sau:

a) Sắp xếp số trạm dừng của từng tuyến xe.

b) Sắp xếp số lượng tuyến xe đi qua từng trạm dừng.

Khung 43. Khởi tạo FUNCTION sử dụng biến bảng.

```

-- a) Sắp xếp số trạm dừng của từng tuyến xe
GO

```

```

CREATE FUNCTION fn_SOTRAM_TUYENXE()
RETURNS @Danh_Sach TABLE (Ma_Tuyen NVARCHAR(10), Ten_Tuyen
NVARCHAR(100), So_Tram INT)
AS
BEGIN
    INSERT INTO @Danh_Sach
    SELECT tx.Ma_Tuyen, tx.Ten_Tuyen, COUNT(DISTINCT
lt.Ma_Tram) AS So_Tram
    FROM TUYENXE tx
    INNER JOIN LOTRINH lt ON tx.Ma_Tuyen = lt.Ma_Tuyen
    GROUP BY tx.Ma_Tuyen, tx.Ten_Tuyen;
    RETURN;
END;

-- b) Sắp xếp số lượng tuyến xe đi qua của từng trạm
GO
CREATE FUNCTION fn_TUYENXE_TRAMDUNG_SAPXEP()
RETURNS @Danh_Sach TABLE (Ma_Tram NVARCHAR(10), So_Tuyen
INT)
AS
BEGIN
    INERT INTO @Danh_Sach
    SELECT lt.Ma_Tram, COUNT(DISTINCT lt.Ma_Tuyen) AS
So_Tuyen
    FROM LOTRINH lt
    GROUP BY lt.Ma_Tram;
    RETURN;
END;

```

Khung 44. Truy vấn FUNCTION sử dụng biến bảng.

GO

```

SELECT * FROM dbo.fn_SOTRAM_TUYENXE() ORDER BY So_Tram
DESC;

GO

SELECT * FROM dbo.fn_TUYENXE_TRAMDUNG_SAPXEP() ORDER
BY So_Tuyen DESC;

```

	Ma_Tram	Ten_Tuyen	So_Tram
1	M018	A2	44
2	M019	A4	35
3	M020	A9	33
4	M021	B1	32
5	M022	B3A	28

	Ma_Tram	So_Tuyen
1	M154	3
2	M155	3
3	M152	3
4	M153	3
5	M154	3
6	M155	3
7	M154	2
8	M155	2
9	M153	2
10	M154	2
11	M155	2
12	M153	2
13	M154	2
14	M155	2
15	M153	2
16	M154	2
17	M155	2
18	M153	2
19	M154	2
20	M155	2
21	M153	2
22	M154	2

Ảnh 23. Kết quả sau khi thực hiện truy vấn FUNCTION sử dụng biến bảng.

4.5. Tạo các TRIGGER để kiểm soát dữ liệu

4.5.1. TRIGGER theo dõi cập nhật, thay đổi dữ liệu các bảng

Khung 45. Tạo bảng lưu lịch sử thay đổi dữ liệu.

```

GO

CREATE TABLE LICHUTHAYDOIDULIEU(
    ID INT IDENTITY (1, 1) PRIMARY KEY,
    Bang_Tac_Dong NVARCHAR(50),
    Loai_Tac_dong NVARCHAR(50),
    Ma_Doi_Tuong_Bi_Tac_Dong NVARCHAR(10),
    Thong_Tin_Cu NVARCHAR(MAX),
    Thong_Tin_Moi NVARCHAR(MAX),
    Ngay_Thay_Doi DATETIME DEFAULT GETDATE()
);

```

Khung 46. Cài đặt TRIGGER cho các bảng (Ví dụ: Cài đặt cho bảng PHUONGTIEN).

```

GO

CREATE TRIGGER trg_LICHSU_PHUONGTIEN

```

```

ON PHUONGTIEN
AFTER INSERT, UPDATE, DELETE
AS
BEGIN
    SET NOCOUNT ON;
    -- INSERT
    INSERT INTO LICHUTHAYDOIDULIEU (Bang_Tac_Dong,
Loai_Tac_Dong, Ma_Doi_Tuong_Bi_Tac_Dong, Thong_Tin_Cu,
Thong_Tin_Moi, Ngay_Thay_Doi)
    SELECT
        'PHUONGTIEN',
        'INSERT',
        i.Ma_Phuong_Tien,
        NULL,
        (SELECT Ma_Phuong_Tien, Ma_Tuyen, Ma_Tai_Xe,
Ten_Phuong_Tien, Bien_Kiem_Soat, Hang_San_Xuat,
Nam_San_Xuat, So_Khung, So.Cho_Ngoi
        FROM inserted i FOR JSON AUTO),
        GETDATE()
    FROM inserted i;
    -- UPDATE
    INSERT INTO LICHUTHAYDOIDULIEU (Bang_Tac_Dong,
Loai_Tac_Dong, Ma_Doi_Tuong_Bi_Tac_Dong, Thong_Tin_Cu,
Thong_Tin_Moi, Ngay_Thay_Doi)
    SELECT
        'PHUONGTIEN',
        'UPDATE',
        i.Ma_Phuong_Tien,
        (SELECT Ma_Phuong_Tien, Ma_Tuyen, Ma_Tai_Xe,
Ten_Phuong_Tien, Bien_Kiem_Soat, Hang_San_Xuat,
Nam_San_Xuat, So_Khung, So.Cho_Ngoi
        FROM deleted d FOR JSON AUTO),

```

```

        (SELECT Ma_Phuong_Tien, Ma_Tuyen, Ma_Tai_Xe,
Ten_Phuong_Tien, Bien_Kiem_Soat, Hang_San_Xuat,
Nam_San_Xuat, So_Khung, So_Cho_Ngoi
        FROM inserted i FOR JSON AUTO),
        GETDATE()
    FROM deleted d
    JOIN inserted i ON d.Ma_Phuong_Tien = i.Ma_Phuong_Tien;
-- DELETE
    INSERT INTO LICHUTHAYDOIDULIEU (Bang_Tac_Dong,
Loai_Tac_Dong, Ma_Doi_Tuong_Bi_Tac_Dong, Thong_Tin_Cu,
Thong_Tin_Moi, Ngay_Thay_Doi)
    SELECT
        'PHUONGTIEN',
        'DELETE',
        d.Ma_Phuong_Tien,
        (SELECT Ma_Phuong_Tien, Ma_Tuyen, Ma_Tai_Xe,
Ten_Phuong_Tien, Bien_Kiem_Soat, Hang_San_Xuat,
Nam_San_Xuat, So_Khung, So_Cho_Ngoi
        FROM deleted d FOR JSON AUTO),
        NULL,
        GETDATE()
    FROM deleted d;
END;

```

Lặp lại các bước cài đặt TRIGGER cho toàn bộ các bảng có trong CSDL. Khi thực hiện cài đặt hoàn tất, toàn bộ lịch sử thay đổi dữ liệu sẽ được lưu vết lại tại bảng LICHUTHAYDOIDULIEU như ảnh dưới đây.

The screenshot shows the SQL Server Management Studio interface with two tabs open: 'SQLQuery_1 - (63) ...DB (sa) 2' and 'SQLQuery_1.sql - (51) ...DB (sa) 9+'. The database is set to 'Transerco_Bus_Management_DB'. The results pane displays a table with 22 rows, each representing a record from the 'LICHUTHAYDOIDULIEU' table. The columns include ID, Bang_Tac_Dong, Loai_Tac_dong, Ma_Doi_Tuong_Bi_Tac_Dong, Thong_Tin_Cu, Thong_Tin_Moi, and Ngay_Thay_Doi. The data shows various operations like INSERT, UPDATE, and DELETE on different columns, with timestamps ranging from 2025-03-14 to 2025-03-19.

Ảnh 24. Kết quả sau khi cài đặt TRIGGER thành công, toàn bộ lịch sử thay đổi dữ liệu sẽ được lưu vết tại bảng LICHUTHAYDOIDULIEU.

4.5.2. Các TRIGGER khác

Khung 47. Khởi tạo TRIGGER ngăn xóa tài xế khi còn dữ liệu phương tiện liên quan.

```

GO
CREATE TRIGGER trg_CHECK_PHUONGTIEN_TAIXE
ON TAIXE
INSTEAD OF DELETE
AS
BEGIN
    IF EXISTS (SELECT 1 FROM deleted d INNER JOIN
    PHUONGTIEN pt ON d.Ma_Tai_Xe = pt.Ma_Tai_Xe)
        BEGIN
            RAISERROR (N'Tài xế đang có phương tiện, không thể
xóa', 16, 1);
            RETURN;
        END
    DELETE FROM TAIXE WHERE Ma_Tai_Xe IN (SELECT Ma_Tai_Xe
from deleted);

```

```
END;
```

Khung 48. Khởi tạo TRIGGER kiểm tra biển kiểm soát của phương tiện.

```
GO
CREATE TRIGGER trg_CHECK_BIENKIEMSOAT
ON PHUONGTIEN
AFTER INSERT
AS
BEGIN
    IF EXISTS (SELECT 1 FROM inserted WHERE Bien_Kiem_Soat
NOT LIKE '[0-9][0-9][A-Z]-[0-9][0-9][0-9].[0-9][0-9]')
        BEGIN
            RAISERROR (N'Biển số xe không hợp lệ!', 16, 1);
            ROLLBACK TRANSACTION;
        END
    END
END;
```

Khung 49. Khởi tạo TRIGGER ngăn chặn thay đổi thông tin mã tuyến.

```
GO
CREATE TRIGGER trg_CHECK_MATUYEN
ON TUYENXE
INSTEAD OF UPDATE
AS
BEGIN
    IF UPDATE(Ma_Tuyen)
        BEGIN
            RAISERROR (N'Không được phép cập nhật mã tuyến!', 16, 1);
            RETURN;
        END
    UPDATE TUYENXE
        SET Ten_Tuyen = i.Ten_Tuyen, Gio_Bat_Dau =
i.Gio_Bat_Dau, Gio_Ket_Thuc = i.Gio_Ket_Thuc
```

```

        FROM TUYENXE t INNER JOIN inserted i ON t.Ma_Tuyen =
i.Ma_Tuyen;
END;

```

Khung 50. Khởi tạo TRIGGER chặn trùng lặp số điện thoại cho tài xế và chi nhánh.

```

GO
CREATE TRIGGER trg_CHECK_SDT_TAIXE
ON TAIXE_LIENHE
AFTER INSERT
AS
BEGIN
    IF EXISTS (SELECT Gia_Tri FROM inserted i WHERE
Loai_Lien_He_Email_SDT = N'Số điện thoại'
                GROUP BY Gia_Tri HAVING COUNT(*) > 1)
        BEGIN
            RAISERROR (N'Số điện thoại đã tồn tại!', 16, 1);
            ROLLBACK TRANSACTION;
        END
    END;

```



```

GO
CREATE TRIGGER trg_CHECK_SDT_CHINHANH
ON CHINHANH_LIENHE
AFTER INSERT
AS
BEGIN
    IF EXISTS (SELECT Gia_Tri FROM inserted i WHERE
Loai_Lien_He_Email_SDT = N'Số điện thoại'
                GROUP BY Gia_Tri HAVING COUNT(*) > 1)
        BEGIN
            RAISERROR (N'Số điện thoại đã tồn tại!', 16, 1);
            ROLLBACK TRANSACTION;
        END
    END

```

```
END;
```

Khung 51. Khởi tạo TRIGGER kiểm tra số chỗ ngồi của phương tiện.

```
GO
CREATE TRIGGER trg_CHECK_SOCHONGOI
ON PHUONGTIEN
AFTER INSERT, UPDATE
AS
BEGIN
    IF EXISTS (SELECT 1 FROM inserted WHERE So.Cho_Ngoi <
16)
        BEGIN
            RAISERROR (N'Số chỗ ngồi không hợp lệ, phải lớn hơn
16!', 16, 1);
            ROLLBACK TRANSACTION;
        END
    END;
```

Khung 52. Khởi tạo TRIGGER chặn trùng lắp vị trí của trạm xe.

```
GO
CREATE TRIGGER trg_CHECK_VITRI_TRAMDUNG
ON TRAMXE
AFTER INSERT
AS
BEGIN
    IF EXISTS (SELECT 1 FROM inserted i INNER JOIN TRAMXE t
ON i.Vi_Tri_Tram = t.Vi_Tri_Tram)
        BEGIN
            RAISERROR (N'Vị trí trạm đã tồn tại!', 16, 1);
            ROLLBACK TRANSACTION;
        END
    END;
```

CHƯƠNG 5: BẢO MẬT VÀ QUẢN TRỊ

5.1. Tạo người dùng

Khung 53. Khởi tạo người dùng trong CSDL.

```
GO  
  
USE master;  
  
-- Tạo tài khoản cho Admin  
CREATE LOGIN USER_ADMIN WITH PASSWORD = 'Admin123@';  
  
-- Tạo tài khoản cho nhân viên level 1  
CREATE LOGIN USER_EMP_LV1 WITH PASSWORD = 'EMP123@1';  
  
-- Tạo tài khoản cho nhân viên level 2  
CREATE LOGIN USER_EMP_LV2 WITH PASSWORD = 'EMP123@2';  
  
-- Tạo tài khoản cho nhân viên level 3  
CREATE LOGIN USER_EMP_LV3 WITH PASSWORD = 'EMP123@3';  
  
-- Tạo user trong DB  
USE Transerco_Bus_Management_DB;  
  
CREATE USER USER_ADMIN FOR LOGIN USER_ADMIN;  
CREATE USER USER_EMP_LV1 FOR LOGIN USER_EMP_LV1;  
CREATE USER USER_EMP_LV2 FOR LOGIN USER_EMP_LV2;  
CREATE USER USER_EMP_LV3 FOR LOGIN USER_EMP_LV3;
```

5.2. Phân cấp quyền truy cập của người dùng

Quyền truy cập của từng đối tượng người dùng được phân cấp như sau:

- Đối với tài khoản Admin: Có toàn quyền đọc, thay đổi dữ liệu hoặc cấu trúc bảng trên CSDL. Đồng thời có quyền phân quyền cho các đối tượng người dùng khác.

- Đối với tài khoản nhân viên mức 1: Có thể đọc dữ liệu, thay đổi dữ liệu hoặc cấu trúc bảng trên CSDL. Không có quyền phân quyền cho các đối tượng người dùng khác.

- Đối với tài khoản nhân viên mức 2: Có thể đọc và thay đổi dữ liệu trên CSDL, nhưng không thể thay đổi cấu trúc bảng.

- Đối với tài khoản nhân viên mức 3: Chỉ có thể đọc dữ liệu trên CSDL mà không thể thay đổi chúng.

Để thực hiện cài đặt phân quyền như trên, ta dùng các câu lệnh sau:

Khung 54. Cài đặt quyền cho từng đối tượng người dùng.

```
-- Tài khoản Admin - Full Access - db_owner  
ALTER ROLE db_owner ADD MEMBER USER_ADMIN;  
  
-- Tài khoản level 1 - db_ddladmin, db_datawriter,  
-- db_datareader  
ALTER ROLE db_ddladmin ADD MEMBER USER_EMP_LV1;  
ALTER ROLE db_datawriter ADD MEMBER USER_EMP_LV1;  
ALTER ROLE db_datareader ADD MEMBER USER_EMP_LV1;  
  
-- Tài khoản level 2 - db_datawriter, db_datareader  
ALTER ROLE db_datawriter ADD MEMBER USER_EMP_LV2;  
ALTER ROLE db_datareader ADD MEMBER USER_EMP_LV2;  
  
-- Tài khoản level 3 - db_datareader  
ALTER ROLE db_datareader ADD MEMBER USER_EMP_LV3;
```

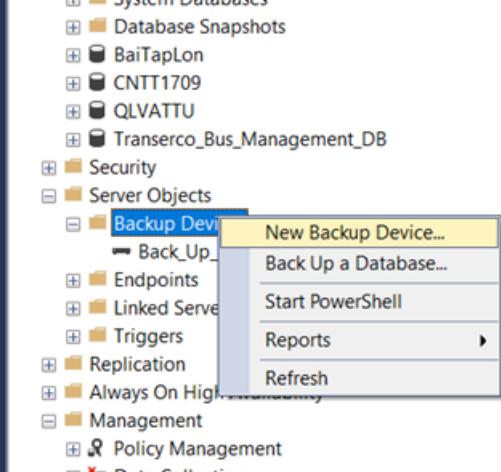
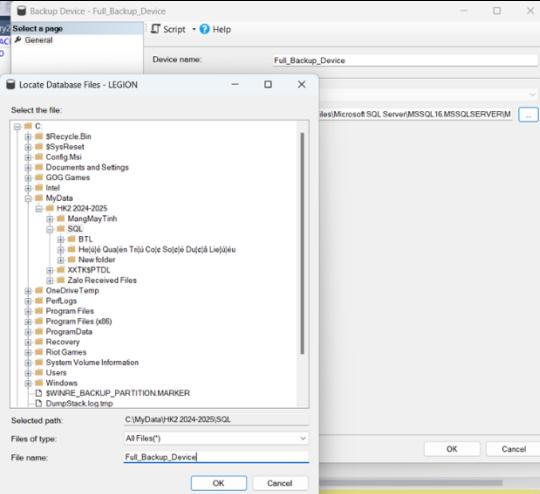
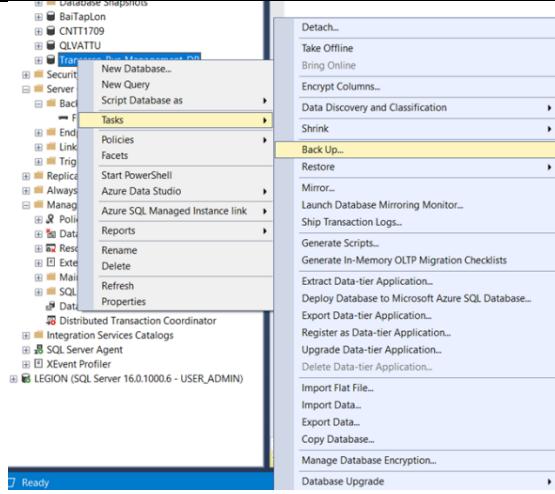
5.3. Quản lý sao lưu và phục hồi dữ liệu

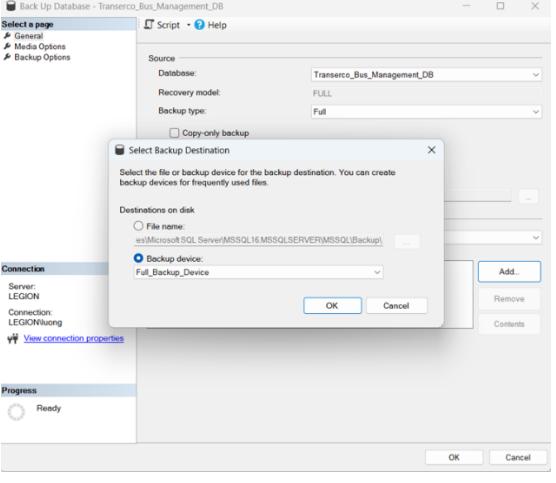
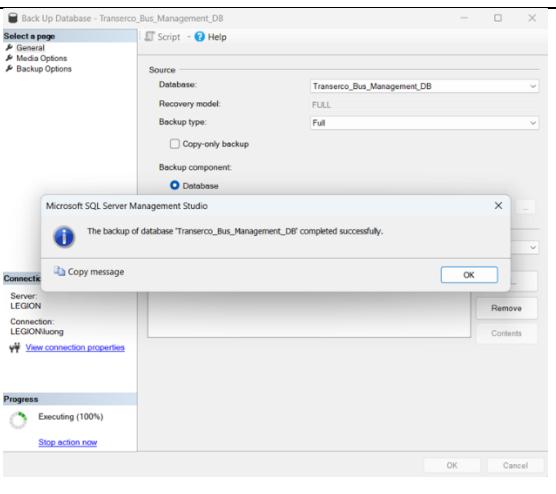
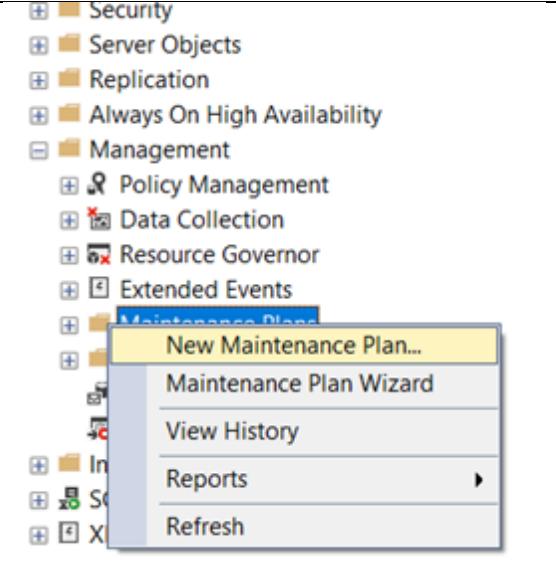
5.3.1. Sao lưu CSDL trên thiết bị chạy hệ điều hành Windows

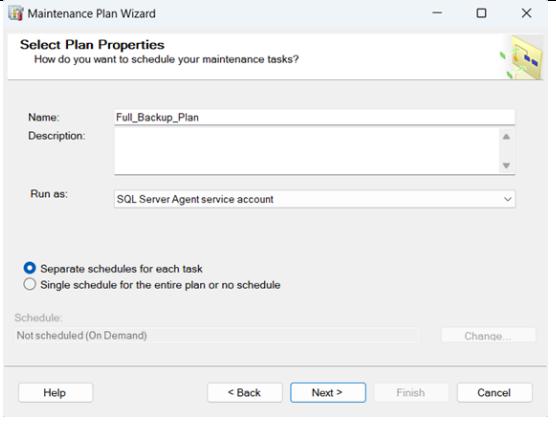
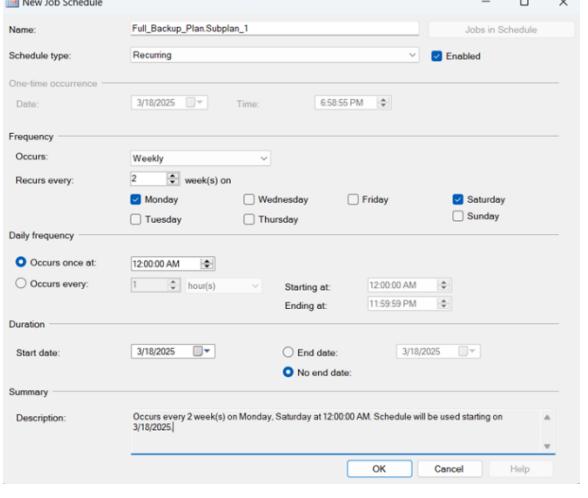
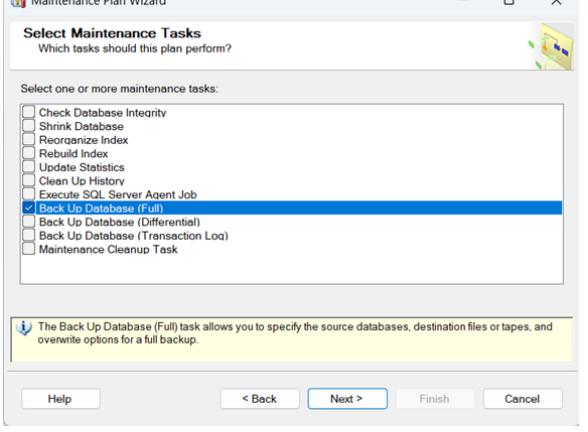
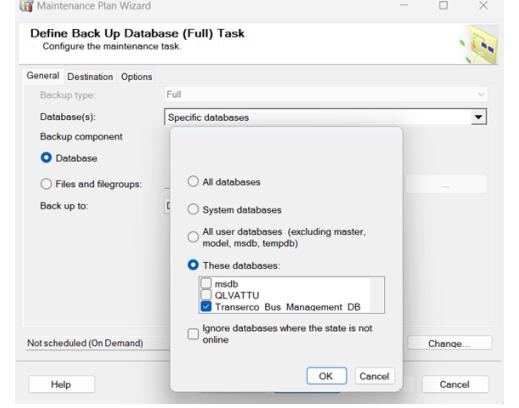
Trên các thiết bị chạy hệ điều hành Windows, để tự động sao lưu dữ liệu, có thể sử dụng SQL Server Agent (yêu cầu SQL Server Enterprise Edition). Quá trình thực hiện bao gồm các bước sau:

Bảng 2. Quy trình cài đặt backup tự động cho CSDL.

Bước	Hình ảnh minh họa	Nội dung
------	-------------------	----------

1		Tại thẻ Object Explorer => Chọn Server Objects => Chọn Backup Device => Chọn New Backup Device
2		Tại cửa sổ Backup Device => Cài đặt tên tại Device name => Lựa chọn vị trí lưu bản backup => Tại cửa sổ Locate Database Files => Đặt tên cho file tại File name
3		Sau khi tạo Backup Device => Tại CSDL cần backup, nhấn chuột phải => Chọn Tasks => Chọn Backup

4		<p>Tại cửa sổ Back Up Database => Xoá đường dẫn mặc định => Chọn Add => Tại cửa sổ Select Backup Destination => Chọn Backup device => Chọn OK => Chọn OK</p>
5		<p>Sau khi chọn OK => Nếu backup thành công sẽ hiện thông báo như ảnh minh họa => Chọn OK để kết thúc.</p>
6		<p>Tại thẻ Object Explorer => Chọn Management => Chọn Maintenance Plans => Chọn Maintenance Plan Wizard</p>

7		<p>Tại cửa sổ Maintenance Plan Wizard => Đặt tên cho plan tại Name => Chọn Separate schedules for each task => Tại mục Schedule => Chọn Change để cài đặt lịch tự động backup</p>
8		<p>Tại cửa sổ New Job Schedule => Tại mục Frequency, có thể cài đặt tần suất backup dữ liệu => Tại mục Daily frequency, có thể cài đặt giờ thực hiện backup trong ngày => Tại mục Duration, có thể cài đặt ngày ngừng backup => Lịch backup tự động sau khi cài đặt được mô tả tại mục Summary</p>
9		<p>Sau khi cài đặt lịch backup tự động thành công => Tại cửa sổ Maintenance Plan Wizard => Chọn Next => Khi được hỏi Select Maintenance Tasks => Chọn Back Up Database (Full) => Chọn Next</p>
10		<p>Khi xuất hiện Define Back Up Database (Full) Task => Tại thẻ General, mục Database(s) => Chọn đúng database cần backup tại mục These databases => Chọn OK</p>

11		<p>Tại thẻ Destination => Chọn Back up databases across one or more files => Khi hiện cửa sổ Select Backup Destination => Chọn Backup device</p>
12		<p>Tại thẻ Options => Chọn Verify backup integrity => Chọn Next</p>
13		<p>Sau khi chạy thành công => Hiện thông báo như ảnh minh họa => Chọn Close</p>
14		<p>Để cài đặt tính năng gửi thông báo về email, tại SQL Server Agent => Chọn Properties => Chọn Alert System => Chọn Enable mail profile => Chọn profile tại Mail profile => Chọn OK</p>

15		Tại Object Explorer => Chọn SQL Server Agent => Chọn Operators => Chọn New Operator => Nhập tên tại Name => Chọn Enabled => Nhập địa chỉ email nhận thông báo tại E-mail name => Chọn OK
16		Tại Object Explorer => Chọn SQL Server Agent => Chọn Jobs => Chọn Subplan vừa tạo => Chọn Properties => Chọn Email => Cài đặt khi nào thì nhận email thông báo => Chọn Write to the Windows Application event log => Chọn OK
17		Tại Object Explorer => Chọn SQL Server Agent => Chọn Jobs => Chọn Subplan vừa tạo => Chọn Start Job at Step... => Nếu job khởi động thành công sẽ hiện thông báo như ảnh

5.3.2. Sao lưu CSDL trên thiết bị chạy hệ điều hành macOS

Do SQL Server Agent hiện chỉ hỗ trợ hệ điều hành Windows mà không hỗ trợ trên macOS hoặc Linux. Vì vậy, trên các thiết bị chạy hệ điều hành macOS, không thể sử dụng SQL Server Agent để lên lịch backup tự động như trên Windows. Thay vào đó, người dùng phải thực hiện backup thủ công theo các bước sau:

Bảng 3. Quy trình backup thủ công cho thiết bị chạy hệ điều hành macOS.

Bước	Cách thực hiện
------	----------------

1	<p>Chạy câu lệnh dưới đây trong Azure Data Studio:</p> <pre>BACKUP DATABASE Transerco_Bus_Management_DB TO DISK = '/var/opt/mssql/backup/Transerco_Backup.bak' WITH FORMAT, COMPRESSION, INIT;</pre>
2	<p>Sau khi chạy câu lệnh trên, mở Terminal và chạy lệnh sau để liệt kê các file backup trong container:</p> <pre>docker exec -it sql2022 ls -lh /var/opt/mssql/backup</pre> <p>Nếu thấy file backup vừa tạo ở bước 1 nghĩa là CSDL đã được backup.</p>
3	<p>Sao chép file backup từ Docker ra máy macOS bằng cách gõ lệnh sau trong Terminal:</p> <pre>docker cp sql2022:/var/opt/mssql/backup/Transerco_Backup.bak ~/Documents/Transerco_Backup.bak</pre>

KẾT LUẬN

Quá trình thực hiện:

Quá trình thực hiện bài tập lớn đề tài “Thiết kế và triển khai cơ sở dữ liệu cho hệ thống quản lý xe buýt liên tỉnh” với sự hướng dẫn của Thạc sĩ Nguyễn Ngọc Ân, đã giúp nhóm chúng tôi có cơ hội tiếp cận một cách thực tế và khoa học với các vấn đề trong quản lý dữ liệu. Để hoàn thành đề tài, nhóm chúng tôi đã trải qua các giai đoạn chính sau:

- Tìm hiểu và phân tích bài toán: Nhóm đã nghiên cứu kĩ càng về mô hình tổ chức của một đơn vị cung cấp dịch vụ vận tải công cộng từ thực tế (Transerco), qua đó xác định các thực thể quan trọng trong bài toán quản lý như tuyến xe, trạm dừng, phương tiện, tài xế, chi nhánh và thành phố. Từ đó xây dựng một hệ thống có thể đáp ứng tối đa các yêu cầu khi đưa vào vận hành thực tế.

- Thiết kế các mô hình ERD và mô hình quan hệ: Từ những dữ liệu thu thập được qua quá trình tìm hiểu thực tế, nhóm đã xây dựng mô hình ERD cho bài toán, nhằm trực quan hóa cách mà các thực thể liên kết với nhau. Việc xác định các mối quan hệ giữa các thực thể là bước quan trọng để đảm bảo tính logic và hiệu quả của hệ thống. Từ mô hình ERD, nhóm đã áp dụng các quy tắc chuyển đổi để đảm bảo các bảng trong CSDL được tổ chức một cách khoa học, không có dữ liệu dư thừa, đồng thời duy trì tính toàn vẹn của dữ liệu.

- Chuẩn hóa và cài đặt CSDL bằng SQL: Nhóm đã thực hiện việc chuẩn hóa các bảng dữ liệu đến chuẩn 3NF để loại bỏ dư thừa và tránh xung đột dữ liệu. Sau đó, nhóm tiếp tục sử dụng các câu lệnh SQL để tạo CSDL.

- Tìm hiểu và triển khai các câu lệnh nâng cao: Sau khi xây dựng CSDL bằng SQL, nhóm đã thực hiện việc triển khai các câu lệnh nâng cao như JOIN, GROUP BY, HAVING, SUBQUERY, tạo View, tạo Stored Procedure, tạo Trigger,... Quá trình này giúp nhóm hiểu sâu hơn về cách tổ chức và quản lý dữ liệu.

Kết quả đạt được:

Sản phẩm thu được sau quá trình thực hiện đề tài có thể truy cập thông qua hai đường dẫn sau:

- Github: <https://github.com/MinhNghia2802/BTL-HQT-CSDL>

- Onedrive: https://dainamedu-my.sharepoint.com/:f/g/personal/1771020505_dnu_edu_vn/EtHRBVsKb99DhSUVhd_J4XUBPyYZDsGlFXRGbdE5QzWDdw?e=Fa7JM1

Về ưu điểm của hệ thống đã được triển khai:

- Hệ thống được thiết kế chặt chẽ theo mô hình chuẩn hoá 3NF, đảm bảo tính toàn vẹn dữ liệu và tránh trùng lặp.
- Các ràng buộc khoá chính và khoá ngoại được thiết lập rõ ràng, giúp duy trì tính nhất quán của dữ liệu.
- Cung cấp nhiều truy vấn giúp tối ưu hoá việc tìm kiếm và phân tích dữ liệu.
- Sử dụng các kỹ thuật nâng cao như Stored Procedure và Trigger giúp kiểm soát dữ liệu một cách hiệu quả và tăng tính tự động hoá.
- Hệ thống được thiết lập tự động backup giúp ngăn ngừa rủi ro mất mát dữ liệu.

Về nhược điểm của hệ thống đã được triển khai:

- Do quy mô của một bài tập lớn và số lượng nhân sự của nhóm có hạn, nên hệ thống chưa được kiểm thử trên dữ liệu thực tế với quy mô lớn.
- Một số truy vấn có thể cần tối ưu hoá thêm để đảm bảo tốc độ phản hồi tốt nhất.
- Hệ thống chưa được thiết kế giao diện UI/UX, hiện tại việc tương tác với hệ thống chủ yếu dựa vào truy vấn SQL.

Công việc trong tương lai:

Trong tương lai, nếu có đủ thời gian và các nguồn lực cần thiết, nhóm dự định:

- Thu thập thêm dữ liệu từ thực tế, xây dựng bộ dữ liệu mẫu quy mô lớn hơn để kiểm thử hiệu suất của hệ thống.
- Thiết kế giao diện người dùng bằng các HTML, CSS, JavaScript hoặc sử dụng các nền tảng như Streamlit để giúp người dùng dễ dàng thao tác với hệ thống.

Tổng kết lại, đợt tài đã giúp nhóm có cơ hội áp dụng những kiến thức đã học trong học phần Hệ quản trị cơ sở dữ liệu vào thực tế. Dù còn một số hạn chế, nhưng sản phẩm cuối cùng đã đạt được những yêu cầu đề ra và có tiềm năng phát triển hơn nữa trong tương lai.

PHÂN CÔNG NHIỆM VỤ

Bảng 4. Bảng phân công nhiệm vụ.

Tham gia: X Không tham gia: O

STT	Đầu việc	Thành viên		
		Nghĩa	Lương	Lan
1	Xây dựng bối cảnh giả định, mô tả bài toán và yêu cầu cần đạt được	o	o	x
2	Xác định thực thể và các thuộc tính tương ứng	o	x	o
3	Xác định liên kết và kiểu liên kết giữa các thực thể	o	x	x
4	Xây dựng mô hình ERD	x	o	o
5	Chuyển hoá mô hình ERD sang mô hình quan hệ	x	o	o
6	Xác định các ràng buộc	x	o	o
7	Xác định các phụ thuộc hàm	x	o	o
8	Chuẩn hoá dữ liệu theo chuẩn 3NF	x	o	o
9	Tạo cơ sở dữ liệu và các bảng	o	x	x
10	Cài đặt các ràng buộc	o	x	x
11	Cài đặt PK và FK	o	x	x
12	Nhập dữ liệu cho các bảng	x	x	x
13	Truy vấn cơ bản (SELECT, INSERT, UPDATE, DELETE)	o	x	x
14	Truy vấn nâng cao (INNER JOIN, GROUP BY, HAVING, SUBQUERY)	x	o	o
15	Tạo view cho CSDL	x	o	o
16	Tạo index cần thiết cho các bảng	x	o	o
17	Tạo Stored Procedure	x	o	o
18	Tạo các Function	x	o	o
19	Tạo các Trigger	x	o	o
20	Tạo người dùng và cấp quyền truy cập của người dùng	o	x	x
21	Cài đặt sao lưu tự động cho CSDL	x	x	o
22	Thiết kế slide thuyết trình báo cáo	o	x	x
TỶ LỆ THAM GIA VÀO ĐỀ TÀI		59,09%	45,45%	45,45%

ĐÁNH GIÁ NỘI BỘ

Bảng 5. Bảng đánh giá thái độ và tinh thần tham gia.

STT	Thành viên	MSV	Đánh giá thái độ và tinh thần tham gia (.../10)			
			Nghĩa	Lương	Lan	TB
1	Nghĩa	1771020505	x	10	9	9.5
2	Lương	1771020444	10	x	8	9
3	Lan	1771020413	9	9	x	9

Bảng 6. Bảng đánh giá chất lượng công việc.

STT	Thành viên	MSV	Đánh giá chất lượng công việc (.../10)			
			Nghĩa	Lương	Lan	TB
1	Nghĩa	1771020505	x	10	10	10
2	Lương	1771020444	9	x	9	9
3	Lan	1771020413	10	9	x	9.5

TÀI LIỆU THAM KHẢO

- [1]. ThS. Nguyễn Ngọc Ân (2025), *Tài liệu học tập học phần Hệ quản trị cơ sở dữ liệu*, Khoa Công nghệ - Thông tin, Trường Đại học Đại Nam.
- [2]. ThS. Nguyễn Ngọc Ân (2025), *Bài giảng học phần Hệ quản trị cơ sở dữ liệu*, Khoa Công nghệ - Thông tin, Trường Đại học Đại Nam.