**CHƯƠNG 5: CÁC DẠNG CHUẨN & KẾT NỐI BẢNG**

**1. Chuẩn 1NF**

+ Một quan hệ(bảng) sẽ ở dạng chuẩn 1(1NF) nếu nó chứa các giá trị nguyên tử. Giá trị nguyên tử tức là không thể phân chia nhỏ hơn được nữa.

+ Nó phát biểu rằng một thuộc tính của bảng không thể chứa đa giá trị. Mỗi thuộc tính chỉ chứa một giá trị đơn.

+ Chuẩn 1 không cho phép thuộc tính đa trị, thuộc tính tổng hợp và sự kết hợp giữa chúng.

+ Nói cách khác, một bảng sẽ thỏa mãn ở dạng chuẩn 1 nếu:

* Chỉ chứa các thuộc tính đơn giá trị
* Miền thuộc tính không thay đổi
* Mọi thuộc tính/cột trong bảng chỉ có một tên duy nhất
* Thứ tự trong đó dữ liệu được lưu trữ không quan trọng

**Ví dụ minh họa**

+ Ví dụ 1

* Trong ví dụ sau, bảng Employees không thuộc dạng chuẩn 1 vì nó chứa thuộc tính đa trị cột PhoneNumber:

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

Mô tả được tạo tự động

* Để đưa bảng trên về dạng chuẩn 1, ta tách mỗi giá trị thành 1 dòng như sau:

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

Mô tả được tạo tự động

+ Ví dụ 2: bảng sau không thỏa mãn chuẩn 1 và cần chỉnh sửa.

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

Mô tả được tạo tự động

* Sau khi thay đổi theo các tiêu chí, loại bỏ tính đa trị ta có bảng ở chuẩn 1:

Ảnh có chứa văn bản, ảnh chụp màn hình, số, Phông chữ

Mô tả được tạo tự động

**2. Chuẩn 2NF**

+ Trong chuẩn 2NF, quan hệ phải thỏa mãn chuẩn 1NF

+ Trong chuẩn 2, mọi thuộc tính không phải khóa phải phụ thuộc hàm đầy đủ vào thuộc tính khóa. Khóa ở đây là khóa chính.

**Ví dụ minh họa**

* Ví dụ sau ta có bảng Teachers không thuộc chuẩn 2 vì lưu cả thông tin môn học, tên môn học không phụ thuộc vào khóa chính của bảng Teachers (Môn học không liên quan đến thông tin cá nhân của giáo viên)

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

Mô tả được tạo tự động

* Do đó ta tách môn học mà từng giáo viên dạy ra một bảng riêng để đạt chuẩn 2NF. Bảng Teachers:

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

Mô tả được tạo tự động

* Bảng Teaching

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

Mô tả được tạo tự động

**3. Chuẩn 3NF**

+ Một quan hệ sẽ thuộc chuẩn 3NF gọi tắt chuẩn 3 nếu nó thỏa mãn chuẩn 2 và không có bất kì một phụ thuộc một phần bắc cầu nào

+ 3NF là chuẩn giúp giảm trùng lặp dữ liệu và cũng được dùng để đạt được tính toàn vẹn dữ liệu

+ Nếu không có phụ thuộc bắc cầu vào thuộc tính không khóa thì quan hệ đó sẽ thuộc chuẩn 3

+ Một quan hệ ở dạng chuẩn 3 nếu thỏa mãn ít nhất 1 trong 2 điều kiện sau với mọi phụ thuộc hàm X -> Y

* X là khóa chính
* Y là thuộc tính chính, mỗi phần tử Y là một phần của một ứng viên khóa chính nào đó

**Ví dụ minh họa**

* Cho bảng Teachers như sau:

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

Mô tả được tạo tự động

* Xác định khóa chính: {TeacherId}, {TeacherId, Name}, {TeacherId, Name, Age}, … => Khóa chính là TeacherId
* Các thuộc tính không khóa: ở đây SubjectName, SubjectLesson phụ thuộc SubjectId mà SubjectId phụ thuộc TeacherId => do đó hai thuộc tính {SubjectName, SubjectLesson} phụ thuộc bắc cầu qua SubjectId với khóa chính TeacherId. Vi phạm chuẩn 3
* Để đạt được chuẩn 3, ta tách {SubjectId, SubjectName, SubjectLesson} ra một bảng khác và chọn SubjectId làm khóa chính của bảng
* Kết quả là bảng sẽ được thiết kế lại như sau: bảng Teachers

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

Mô tả được tạo tự động

* Bảng Subjects:

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

Mô tả được tạo tự động

**4. Chuẩn BCNF**

+ BCNF là một chuẩn mở rộng của 3NF. BCNF chặt chẽ hơn 3NF

+ Một bảng sẽ ở chuẩn BCNF nếu mọi phụ thuộc hàm X->Y, X là siêu khóa của bảng

+ Siêu khóa là tập hợp các cột xác định duy nhất bất kì dòng nào trong một bảng

+ Để ở dạng chuẩn BCNF, bảng phải thỏa mãn chuẩn 3NF

**Ví dụ minh họa**

* Xét bảng các giáo viên dạy các môn học như sau:

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

Mô tả được tạo tự động

* Các phục thuộc hàm:
  + TeacherId -> {Name, Age}
  + SubjectId -> {SubjectName, SubjectLesson}
* Các ứng viên khóa chính: {TeacherId, SubjectId}
* Bảng trên không thỏa mãn BCNF vì bản thân TeacherId hoặc SubjectId không phải khóa chính
* Để chuyển đổi bảng sang BCNF ta tách bảng thành 3:
  + Teachers

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

Mô tả được tạo tự động

* + Subjects

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

Mô tả được tạo tự động

* + Teachings

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

Mô tả được tạo tự động

* Ứng viên khóa chính:
  + Bảng Teachers: TeacherId
  + Bảng Subjects: SubjectId
  + Bảng Teaching: {TeacherId, SubjectId}
* Lúc này mọi phụ thuộc hàm X -> Y trong các bảng trên thì X luôn là khóa chính. Do đó các bảng của ta thuộc chuẩn BCNF

**5. Chuẩn 4NF**

+ Một quan hệ sẽ ở dạng chuẩn 4NF nếu nó đạt chuẩn BCNF và không có phụ thuộc đa trị

+ Với một phụ thuộc hàm A -> B, nếu với một giá trị đơn của A, tồn tại nhiều giá trị của B, thì quan hệ sẽ là phụ thuộc đa trị

+ Một phụ thuộc đa trị xảy ra khi hai thuộc tính độc lập nhau nhưng cả hai đều phụ thuộc một thuộc tính thứ ba

+ Ví dụ dưới đây cho thấy phụ thuộc đa trị giữa mã sinh viên với khóa học và sở thích

**Ví dụ minh họa**

* Xét bảng Students như sau:

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

Mô tả được tạo tự động

* Trong bảng trên ta thấy đã đạt chuẩn 3NF nhưng khóa học(Course) và sở thích(Hobby) là hai đối tượng độc lập. Do đó không có mối quan hệ giữa Course và Hobby
* Ta thấy rằng với sinh viên Trần Trung Đức mã sinh viên ST001 có hai khóa học là Java và Toán cao cấp, mặt khác anh này lại có 2 sở thích khác nhau: Khám phá công nghệ và giải toán
* Do đó vi phạm tiêu chuẩn của chuẩn 4NF vì chứa phụ thuộc đa trị dẫn tới dư thừa dữ liệu
* Ta tách bảng ban đầu thành hai bảng để đạt chuẩn 4NF
* Bảng StudentCourse:

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

Mô tả được tạo tự động

* Bảng StudentHobbies:

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

Mô tả được tạo tự động

**6. Mô hình ER**

+ ER là viết tắt của Entity-Relationship model. Đó là mô hình dữ liệu bậc cao sử dụng để  định nghĩa các phần tử dữ liệu và quan hệ giữa các thành phần của hệ thống cụ thể.

+ Mô hình này phát triển một thiết kế mang tính khái niệm cho CSDL. Nó cũng phát triển một chế độ xem dữ liệu đơn giản và dễ thiết kế.

+ Trong mô hình ER, cấu trúc CSDL được mô tả dưới dạng một sơ đồ gọi là biểu đồ quan hệ thực thể.

+ Ví dụ ta thiết kế CSDL cho một trường đại học. Trong đó Sinh viên là một thực thể với các thuộc tính như mã sinh viên, email, số điện thoại, họ tên. Trong đó họ tên có thể là một thực thể với các thuộc tính như họ, đệm, tên và sẽ có mối quan hệ giữa chúng.

Ảnh có chứa văn bản, biểu đồ, vòng tròn, ảnh chụp màn hình

Mô tả được tạo tự động

**Các thành phần của biểu đồ ER**

* Các thành phần của biểu đồ ER cho trong hình sau:

Ảnh có chứa văn bản, ảnh chụp màn hình, biểu đồ, Phông chữ

Mô tả được tạo tự động

**Các thực thể**

* Một thực thể có thể là bất kì một đối tượng, lớp, người hay địa chỉ nào. Trong biểu đồ ER, một thực thể được thể hiện bằng hình chữ nhật
* Ví dụ trong một công ty, quản lý, nhân viên, lập trình viên, tester, giám đốc, sản phẩm, dịch vụ,… đều có thể được coi là các thực thể

Ảnh có chứa hàng, biểu đồ, Phông chữ, ảnh chụp màn hình

Mô tả được tạo tự động

* Một thực thể phụ thuộc vào một thực thể khác gọi là thực thể yếu. Thực thể yếu không chứa thuộc tính khóa của riêng nó
* Một thực thể yếu được biểu diễn bằng một hình chữ nhật viền kép

Ảnh có chứa văn bản, Hình chữ nhật, ảnh chụp màn hình, hàng

Mô tả được tạo tự động

**Các thuộc tính**

* Thuộc tính là khái niệm sử dụng để mô tả các đặc trưng, đặc điểm của một thực thể
* Thuộc tính được mô tả bằng hình elipse
* Thuộc tính khóa là thuộc tính chính phân biệt giữa các thực thể với nhau. Đó chính là khóa chính của bảng. Thuộc tính khóa được viết gạch chân
* Thuộc tính tổng hợp là thuộc tính bao gồm nhiều thuộc tính khác cấu thành, ví dụ họ và tên gồm họ, đệm, tên. Các thuộc tính tổng hợp nối với nhau tạo thành cụm
* Thuộc tính đa trị là thuộc tính có nhiều giá trị như số điện thoại, email của một người. Nó được biểu diễn bằng hình elipse có đường viền đôi
* Thuộc tính dẫn xuất là thuộc tính có thể suy ra, tính toán từ các thuộc tính khác. Ví dụ tuổi có thể suy ra từ ngày tháng năm sinh. Biểu diễn thuộc tính dẫn xuất bằng hình elipse viền nét đứt
* Hình sau mô tả các loại thuộc tính kể trên:

Ảnh có chứa văn bản, ảnh chụp màn hình, biểu đồ, vòng tròn

Mô tả được tạo tự động

**Các quan hệ**

* Một quan hệ sử dụng để mô tả quan hệ giữa hai thực thể
* Quan hệ được mô tả bằng hình thoi
* Các dạng quan hệ:
* **Một – Một:** xảy ra khi chỉ một đối tượng cụ thể của thực thể liên kết với quan hệ. Ví dụ trong cuộc hôn nhân, 1 nam kết hôn với chỉ 1 nữ và ngược lại

Ảnh có chứa hàng, biểu đồ, ảnh chụp màn hình, Sơ đồ

Mô tả được tạo tự động

* **Một – Nhiều:** xảy ra khi chỉ một đối tượng của thực thể bên trái liên kết với nhiều đối tượng của thực thể bên phải. Quan hệ này ngược lại với quan hệ nhiều – một. Kí hiệu bên nhiều là M(multiple).

Ảnh có chứa văn bản, ảnh chụp màn hình, biểu đồ, hàng

Mô tả được tạo tự động

* **Nhiều – Nhiều:** xảy ra khi nhiều đối tượng của thực thể bên trái liên kết với nhiều đối tượng của thực thể bên phải. Ví dụ một nhân viên có thể được gán nhiều dự án cùng lúc và một dự án được thực hiện bởi nhiều nhân viên.

Ảnh có chứa văn bản, hàng, ảnh chụp màn hình, Phông chữ

Mô tả được tạo tự động

**7. Tìm hiểu tổng quan về các kiểu nối bảng**

+ Trong CSDL quan hệ, dữ liệu được phân tán trên các bảng khác nhau

+ Để lấy được tập dữ liệu có đầy đủ ý nghĩa sử dụng cho một mục đích nào đó, ta cần truy vấn các bảng thành phần sử dụng các kết nối bảng

+ SQL Server hỗ trợ nhiều loại nối bảng:

* + INNER JOIN
  + LEFT JOIN
  + RIGHT JOIN
  + FULL OUTER JOIN
  + CROSS JOIN

+ Mỗi loại kết nối bảng lại có cách sử dụng dữ liệu khác nhau từ một bảng để chọn các dòng dữ liệu của bảng khác.

+ Giả sử ta có bảng nhân viên và ứng viên. Xét mối liên hệ dữ liệu giữa hai bảng này:

* + INNER JOIN: là kiểu nối bảng cho ra kết quả là phần dữ liệu giao nhau của hai bảng. Sau đây là biểu đồ mô tả phần dữ liệu khớp nhau ở hai bảng:

Ảnh có chứa vòng tròn, Phông chữ, biểu tượng, Đồ họa

Mô tả được tạo tự động

* + LEFT JOIN: còn gọi là LEFT OUTER JOIN là kiểu nối bảng cho ra kết quả là phần dữ liệu trong bảng bên trái và phần dữ liệu khớp nhau giữa hai bảng. Những cột của dòng dữ liệu không khớp ở phía bảng bên phải sẽ nhận giá trị NULL. Sau đây là biểu đồ mô tả phần dữ liệu của phép nối bảng và phần dữ liệu chỉ tồn tại bên bảng trái:

Ảnh có chứa vòng tròn, màu vàng, Phông chữ, thiết kế

Mô tả được tạo tự độngẢnh có chứa vòng tròn, Phông chữ, biểu tượng, Đồ họa

Mô tả được tạo tự động

* + RIGHT JOIN còn gọi là RIGHT OUTER JOIN là kiểu nối bảng cho ra kết quả là phần dữ liệu trong bảng bên phải và phần dữ liệu khớp nhau giữa hai bảng. Những cột của các dòng không khớp trong bảng bên trái sẽ nhận giá trị NULL. Sau đây là biểu đồ thể hiện kết quả nối bảng và dữ liệu chỉ thuộc bảng bên phải phép nối:

Ảnh có chứa vòng tròn, Phông chữ, màu vàng, văn bản

Mô tả được tạo tự độngẢnh có chứa vòng tròn, Phông chữ, văn bản, biểu tượng

Mô tả được tạo tự động

* + FULL JOIN hay FULL OUTER JOIN trả về tập kết quả chứa tất cả các dòng trong cả hai bảng. Dữ liệu trả về gồm các bản ghi khớp nhau hoặc các bản ghi chỉ khớp bảng bên trái, hoặc chỉ khớp với bảng bên phải. Những phần không khớp sẽ có giá trị NULL tương ứng. Sau đây là biểu đồ của kết quả nối bảng FULL JOIN và kết quả phần không giao nhau của tập dữ liệu trong hai bảng:

Ảnh có chứa biểu tượng, vòng tròn, Phông chữ

Mô tả được tạo tự động

Ảnh có chứa vòng tròn, Phông chữ, biểu tượng, văn bản

Mô tả được tạo tự động

**8. INNER JOIN**

+ INNER JOIN cho phép ta truy vấn dữ liệu từ nhiều bảng có liên quan đến nhau trong CSDL.

+ Xét quan hệ các bảng như sau:

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

Mô tả được tạo tự động

+ Bây giờ nếu ta muốn biết sinh viên nào đã đăng ký môn học, sinh viên nào chưa đăng ký, sinh viên A đăng ký những môn học nào, không đăng ký những môn học nào,… thì không thể chỉ thực hiện truy vấn trong bảng Student.

+ Lúc này cần truy vấn cả bảng Register và Subject để lấy các thông tin liên quan.

+ Ở đây giữa bảng Student và Subject có quan hệ nhiều nhiều nên phải tách bảng Register thể hiện quan hệ nhiều – một giữa việc đăng ký với từng sinh viên và từng môn học.

+ Việc sử dụng kết nối bảng INNER JOIN cho phép ta tìm ra những bản ghi xuất hiện trong tất cả các bảng liên quan:

* Nếu muốn biết sinh viên nào đăng ký môn học, ta chỉ cần kiểm tra xem mã sinh viên đang xét có tồn tại trong bảng đăng ký không
* Nếu muốn biết môn học nào được sinh viên X đăng ký thì ta chỉ cần tìm xem mã sinh viên X có xuất hiện trong bảng đăng ký không và ứng với dòng có mã sinh viên X đó ta tìm được mã môn học sinh viên X đăng ký. Tiếp theo tìm trong bảng môn học xem môn nào có mã trùng với mã của môn học vừa tìm được là ra thông tin môn học cần tìm
* Tương tự ta sẽ tìm được những sinh viên không đăng ký môn học X, những sinh viên không đăng ký bất kỳ môn học nào,…

**9. LEFT JOIN**

+ Mệnh đề LEFT JOIN thường được sử dụng đi kèm trong câu lệnh SELECT.

+ Mệnh đề LEFT JOIN cho phép ta truy vấn dữ liệu từ nhiều bảng có liên quan đến nhau.

+ Mệnh đề LEFT JOIN trả về tất cả các dòng trong bảng bên tay trái và các hàng trùng khớp ở bảng bên phải. Nếu không có dòng trùng khớp với dòng dữ liệu ở bảng bên trái thì giá trị NULL sẽ được sử dụng để tạo tập kết quả.

+ Cú pháp của LEFT JOIN:

SELECT

columns

FROM

table1

LEFT JOIN table2

ON join\_predicate;

+ Trong đó:

* table1, table2 là hai bảng trái và phải của mệnh đề
* Với mỗi dòng trong bảng table1, câu truy vấn sẽ so sánh nó với tất cả các dòng trong bảng table2. Nếu có cặp dòng thỏa mãn điều kiện join\_predicate thì các cột của cặp dòng thỏa mãn đó sẽ được sử dụng để kết hợp lại tạo thành 1 dòng mới trong bảng kết quả
* Nếu một dòng nào đó trong bảng table1 mà không có dòng trùng khớp trong bảng bên phải thì câu truy vấn sẽ kết hợp giá trị các cột trên dòng đó và thay giá trị các cột bảng bên phải bởi các giá trị NULL để làm kết quả truy vấn

+ Sau đây là ví dụ minh họa hai bảng table1 và table2 với các dòng tương ứng table1(1, 2, 3), table2(A, B, C) được kết nối bảng LEFT JOIN và kết quả của nó:

Ảnh có chứa văn bản, Hình chữ nhật, biểu đồ, đen và trắng

Mô tả được tạo tự động

+ Dễ thấy rằng hàng 1 của bảng table1 không có hàng trùng khớp trong bảng table2 nên trong kết quả ta nhận được giá trị cột tương ứng phần của table2 là NULL.

**10. RIGHT JOIN**

+ Mệnh đề RIGHT JOIN là mệnh đề của câu lệnh truy vấn SELECT. Mệnh đề này sử dụng để kết hợp dữ liệu từ 2 hoặc nhiều bảng.

+ Mệnh đề này hoạt động bằng cách lấy từng dòng trong bảng bên phải và so sánh chúng với các hàng của bảng bên trái.

+ Mệnh đề RIGHT JOIN trả về một tập kết quả bao gồm tất cả các dòng của bảng bên phải cho dù chúng có khớp với điều kiện đưa ra hay không.

+ Nếu một dòng trong bảng bên phải không khớp bất kì dòng nào trong bảng bên trái, các cột thể hiện cho phần dữ liệu của bảng bên trái sẽ được thay bằng các giá trị NULL.

+ Cú pháp của RIGHT JOIN:

SELECT

columns

FROM

table1

RIGHT JOIN table2

ON join\_predicate;

* Trong cú pháp trên, table1, table2 là hai bảng trái, phải tham gia vào mệnh đề RIGHT JOIN.
* Lưu ý rằng RIGHT JOIN và RIGHT OUTER JOIN là như nhau. Keyword OUTER là tùy chọn.
* Tương tự LEFT JOIN và LEFT OUTER JOIN là như nhau.
* Biểu đồ ven mô tả RIGHT JOIN: T1 thể hiện bảng trái, T2 thể hiện bảng bên phải

Ảnh có chứa vòng tròn, màu vàng, thiết kế

Mô tả được tạo tự động

**11. FULL OUTER JOIN**

+ Mệnh đề FULL OUTER JOIN là mệnh đề của câu lệnh SELECT trả về tập kết quả bao gồm các hàng từ hai bảng trái, phải.

+ Khi không có hàng trùng khớp với hàng đang xét ở bảng bên trái, các cột ở bảng bên phải sẽ chứa các giá trị NULL.

+ Tương tự nếu đang xét hàng trong bảng bên phải mà không có bất kì hàng nào trùng khớp trong bảng bên trái thì các cột trong phần bảng bên trái của dòng kết quả sẽ nhận giá trị NULL.

+ Cú pháp của FULL OUTER JOIN:

SELECT

columns

FROM

table1

FULL OUTER JOIN table2

ON join\_predicate;

* Keyword OUTER là tùy chọn do đó ta có thể bỏ qua hoặc cho vào lệnh truy vấn đều cho cùng kết quả.
* Trong cú pháp trên, ta cần xác định các cột dữ liệu cần lấy trong phần columns.
* Tiếp theo chỉ định bảng thành phần trái phải trong table1, table2 và điều kiện kết nối hai bảng.
* Biểu đồ minh họa cho FULL OUTER JOIN:

Ảnh có chứa màu vàng, vòng tròn, sáng tạo

Mô tả được tạo tự động

**12. CROSS JOIN**

+ CROSS JOIN nối mỗi hàng trong bảng table1 với mọi hàng trong bảng table2. Nói cách khác, CROSS JOIN trả về tích các hàng hiện có trong hai bảng thành phần.

+ Không giống LEFT JOIN hay INNER JOIN, CROSS JOIN không thiết lập một quan hệ giữa các bảng được nối.

+ Giả sử bảng table1 có 3 hàng 1, 2, 3. bảng table2 có 3 hàng A, B, C.

+ CROSS JOIN trả về kết quả là 9 hàng được minh họa như sau:

Ảnh có chứa mẫu, Hình chữ nhật, văn bản

Mô tả được tạo tự động

+ Như vậy nếu bảng tham gia có m hàng và n hàng tương ứng thì kết quả ta được một bảng mxn hàng.

+ Cú pháp của CROSS JOIN:

SELECT

columns

FROM

table1

CROSS JOIN table2

**13. SELF JOIN**

+ Self join cho phép kết nối một bảng với chính bản thân nó để truy vấn các dữ liệu phân cấp hoặc so sánh các hàng trong cùng một bảng

+ Self join sử dụng mệnh đề INNER JOIN hoặc LEFT JOIN

+ Câu truy vấn sử dụng Self join tham chiếu tới chính nó nên người ta thường đặt tên bí danh khác nhau cho cùng bảng nhằm dễ phân biệt trong câu truy vấn

+ Lưu ý rằng nếu ta tham chiếu cùng một bảng nhiều hơn 1 lần trong 1 câu truy vấn nhưng không sử dụng tên bí danh cho bảng sẽ gây ra lỗi

+ Cú pháp của Self join:

SELECT

select\_list

FROM

T t1

[INNER|LEFT] JOIN T t2

ON join\_predicate;

* Trong cú pháp trên ta đặt bí danh t1, t2 cho cùng bảng T để dễ phân biệt

Trên cùng của Biểu mẫu

Dưới cùng của Biểu mẫu