2.1.3 Ngôn ngữ dữ liệu quan hệ (Relational Data Languages).

*Thêm 4 phần của sql DDL, DQL, DML, DCL*

Ngôn ngữ thao tác dữ liệu (Data Manipulation Languages) được phát triển cho mô hình quan hệ (thường gọi là ngôn ngữ truy vấn – query languages).

Chia thành 2 nhóm: ngôn ngữ đại số quan hệ (relational algebra languages) và ngôn ngữ phép tính quan hệ (relational calculus languages).

2.1.3.1 Đại số quan hệ (Relational Algebra).

Là ngôn ngữ lý thuyết (hình thức) gồm các phép toán thực hiện trên một hay hai quan hệ nhằm định nghĩa một quan hệ mới mà không làm thay đổi quan hệ ban đầu.

Toán tử: operator.

Toán hạng: operand.

Ví dụ: trong số học, toán tử là: +, -, \*, /. Toán hạng là các biến: a, b,.. .

Biểu thức: (a + b)/(c – d).

*Bổ sung thêm sau*

**Năm phép toán cơ sở trong Đại số quan hệ:**

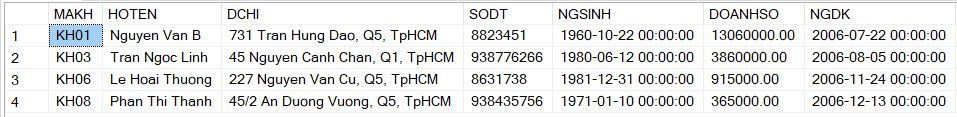
1. Phép chọn (Selection σ): chọn các dòng từ một quan hệ (chiều ngang).

: quan hệ, : điều kiện.



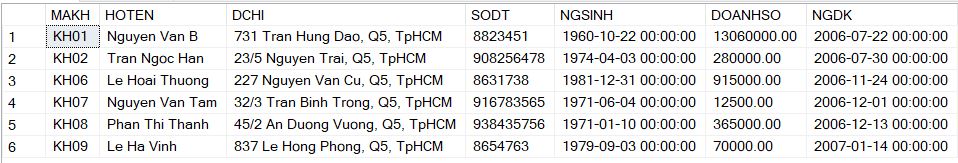
**Error! Use the Home tab to apply 0 to the text that you want to appear here.**‑1Bảng khách hàng.

Ví dụ 1: Chọn khách hang có doanh số lớn hơn 300.000.



**Error! Use the Home tab to apply 0 to the text that you want to appear here.**‑2 Chọn khách hàng có doanh số lớn hơn 300.000.

Ví dụ 2: Chọn khách hàng có địa chỉ Quận 5.



**Error! Use the Home tab to apply 0 to the text that you want to appear here.**‑3 Chọn khách hàng có địa chỉ ở quận 5.

1. Phép chiếu (Projection Π): chọn các cột từ một quan hệ (chiều dọc), nếu các dòng trùng nhau chỉ lấy 1 dòng.

: quan hệ, : các thuộc tính (các cột).



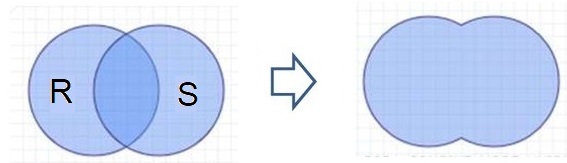
**Error! Use the Home tab to apply 0 to the text that you want to appear here.**‑4 Bảng khách hàng.

Ví dụ 1: Lấy Họ tên và số điện thoại của các khách hàng.



**Error! Use the Home tab to apply 0 to the text that you want to appear here.**‑5 Lấy họ tên và số điện thoại khách hàng.

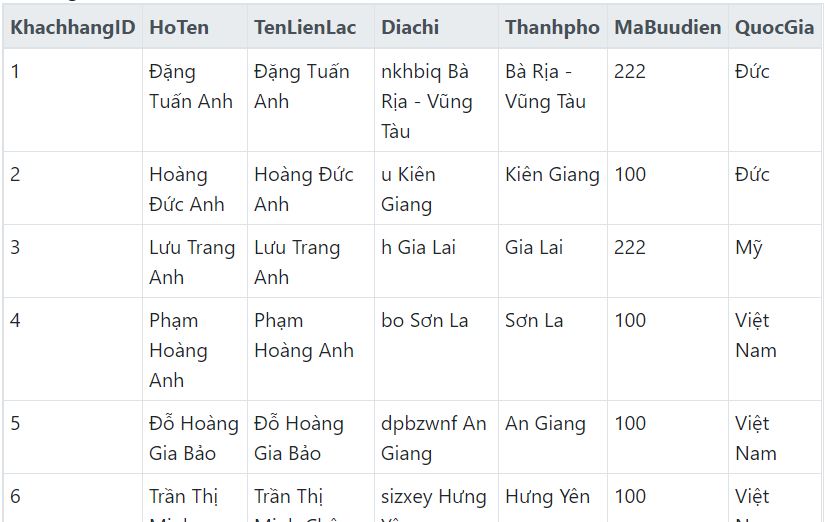
1. Phép hội (Union ):



Hội của hai quan hệ *R* và *S* tạo một quan hệ mới chứa tất cả các bộ của *R*, hoặc của *S* hoặc của cả 2, loại bộ trùng.

Ví dụ: Có 2 bảng: Khách hàng, Cung cấp.

Bảng Khachhang (Khách hàng):



**Error! Use the Home tab to apply 0 to the text that you want to appear here.**‑6 Bảng khách hàng.

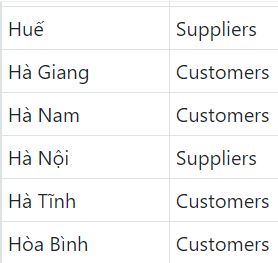
Bảng Cungcap (Cung cấp):

**Error! Use the Home tab to apply 0 to the text that you want to appear here.**‑7 Bảng cung cấp.

Ví dụ: R: Lấy Thanhpho (tên thành phố) từ bảng Khachhang (khách hàng).

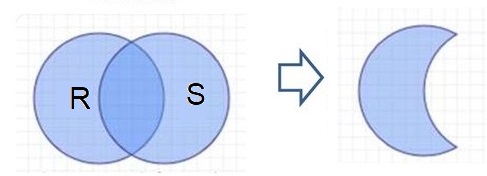
S: Lấy Thanhpho từ bảng Cungcap (cung cấp).

Phép hội của R và S là:



**Error! Use the Home tab to apply 0 to the text that you want to appear here.**‑8 Kết quả phép hội lấy tên Thành phố.

1. Phép hiệu (Set Difference - ):



Hình thành một quan hệ chứa các bộ trong *R* nhưng không thuộc *S*.



**Error! Use the Home tab to apply 0 to the text that you want to appear here.**‑9Bảng SANPHAM

**Error! Use the Home tab to apply 0 to the text that you want to appear here.**‑10 Bảng CTHD

Ví dụ: R: Lấy MASP (mã sản phẩm) từ bảng SANPHAM (sản phẩm).

S: Lấy MASP từ bảng CTHD (chi tiết hóa đơn).

Phép hiệu giữa R và S là: (những sản phẩm chưa bán)



**Error! Use the Home tab to apply 0 to the text that you want to appear here.**‑11 Kết quả phép hiệu.

1. Phép tích Descarte (Cartesian product x).

Hình thành một quan hệ nối các bộ của *R* với mọi bộ của *S.*

Bảng HOADON (hóa đơn):



**Error! Use the Home tab to apply 0 to the text that you want to appear here.**‑12 Bảng HOADON.

Bảng KHACHHANG (khách hàng):

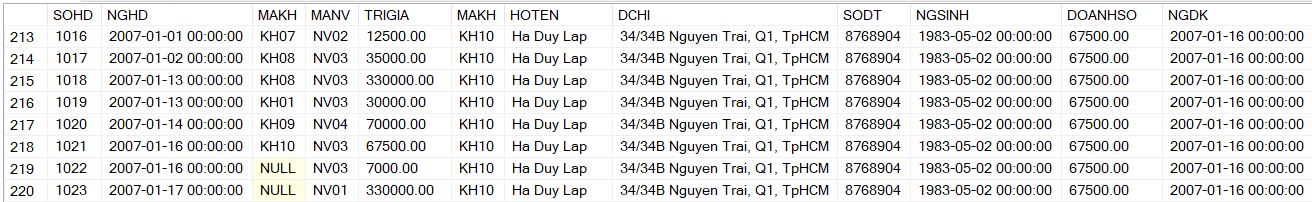


**Error! Use the Home tab to apply 0 to the text that you want to appear here.**‑13 Bảng KHACHHANG.

Ví dụ: R: tất cả thuộc tính của bảng HOADON (hóa đơn).

S: tất cả thuộc tính của bảng KHACHHANG (khách hàng).

Tích của R và S là:



**Error! Use the Home tab to apply 0 to the text that you want to appear here.**‑14 Kết quả phép tích.

Bảng HOADON có 5 cột, 22 dòng.

Bảng KHACHHANG có 7 cột, 10 dòng.

Kết quả phép tích trên có 12 cột, 220 dòng.

*Nhận xét*: khi thực hiện phép tích, cần thêm điều kiện để giảm số dòng không có ý nghĩa. Cụ thể ví dụ: Thêm điều kiện khách hàng có mã khách hàng trùng với mã khách hàng trên hóa đơn.

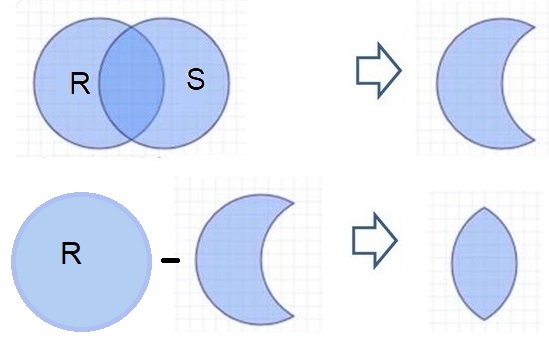


**Error! Use the Home tab to apply 0 to the text that you want to appear here.**‑15 Kết quả phép tích sau khi thêm điều kiện chọn.

Kết quả: mỗi dòng thể hiện thông tin khách hàng cùng với thông tin hóa đơn đúng với khách hàng.

**Các phép toán mở rộng:**

1. Phép giao (Interesection ):

****

Hình thành một quan hệ mới chứa các bộ thuộc cả *R* và *S.*

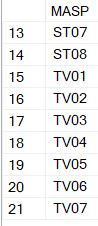
Biểu diễn bằng các toán tử căn bản:

Bảng SANPHAM Bảng CTHD

Ví dụ: R: Lấy MASP (mã sản phẩm) từ bảng SANPHAM (sản phẩm).

S: Lấy MASP từ bảng CTHD (chi tiết hóa đơn).

Phép giao giữa R và S là: (những sản phẩm chưa bán)



**Error! Use the Home tab to apply 0 to the text that you want to appear here.**‑16 Kết quả phép giao

1. Phép Theta kết ():

Hình thành một quan hệ mới chứa các bộ thỏa điều kiện *F* của tích descarte của hai quan hệ *R* và *S.*

*F* có dạng , là một trong các phép so sánh .

Nếu là dấu = thì phép trên được gọi là equi-join.

Một phép equi-join mà bỏ đi các cột trùng nhau thì gọi là natural-join.

Bảng HOADON (hóa đơn):



**Error! Use the Home tab to apply 0 to the text that you want to appear here.**‑17 Bảng HOADON.

Bảng KHACHHANG (khách hàng):



**Error! Use the Home tab to apply 0 to the text that you want to appear here.**‑18 Bảng KHACHHANG.

Ví dụ 1: R: tất cả thuộc tính của bảng HOADON (hóa đơn).

S: tất cả thuộc tính của bảng KHACHHANG (khách hàng).

Theta kết của R và S là:

Hoặc



**Error! Use the Home tab to apply 0 to the text that you want to appear here.**‑19 Kết quả phép tích sau khi thêm điều kiện chọn.

Ví dụ 2: Phép kết equi-join:

Bảng SANPHAM Bảng CTHD

**Error! Use the Home tab to apply 0 to the text that you want to appear here.**‑20 Bảng CTHD

**Error! Use the Home tab to apply 0 to the text that you want to appear here.**‑21 Bảng SANPHAM



**Error! Use the Home tab to apply 0 to the text that you want to appear here.**‑22 Bảng kết quả equijoin.

Ví dụ 3: Phép kết natural-join:



**Error! Use the Home tab to apply 0 to the text that you want to appear here.**‑23 Bảng kết quả natural-join

1. Phép kết ngoài (Outer-join):

* Left outer join:

Là phép kết bảo toàn các bộ của quan hệ bên trái, bộ nào không kết được với quan hệ bên phải thì điền null.

Ví dụ:

Bảng NHANVIEN



Bảng HOADON

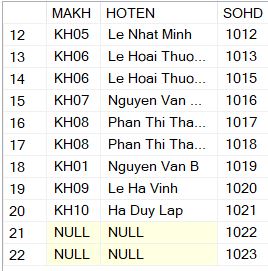




* Right outer join:

Là phép kết bảo toàn các bộ của quan hệ bên phải, bộ nào không kết được với quan hệ bên trái thì điền null.

Ví dụ:

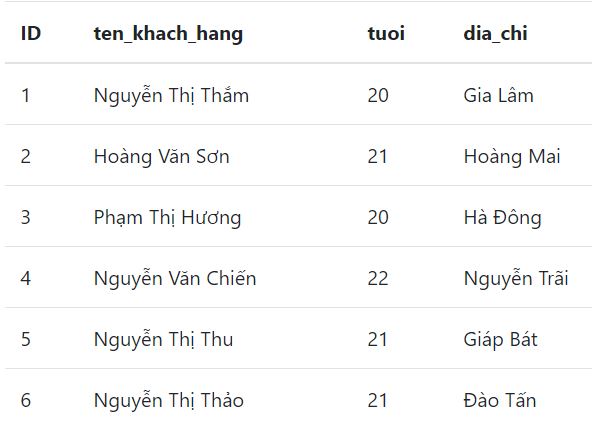


* Full outer join:

Là phép kết bảo toàn các bộ của hai quan hệ, bộ nào không kết được thì điền null.

Ví dụ:

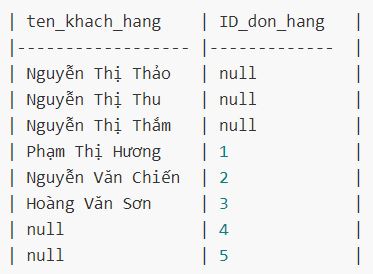
Bảng khách hàng.



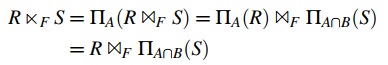
Bảng đơn hàng.



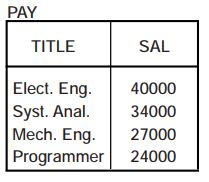
Kết quả full outer join.

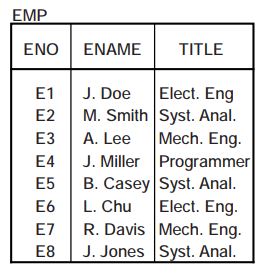


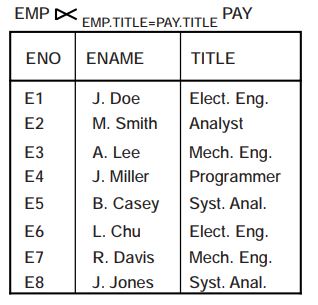
1. Phép kết nửa (semijoin):



Hình thành quan hệ chứa các bộ của *R* có thể tham giá kết với quan hệ *S.*

Ví dụ:





1. Phép chia (Division **÷):**