

# **Lập trình ứng dụng**

## **Ngôn ngữ SQL**

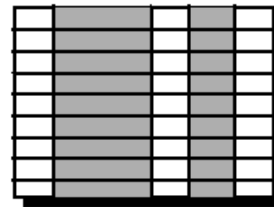
### **Phần 2: Câu truy vấn trên nhiều bảng**

# Lệnh SELECT

Lệnh SELECT là một lệnh đa năng để truy vấn dữ liệu trong CSDL. Nó cho phép thực hiện tất cả các thao tác cơ bản trong đại số quan hệ như:

- Chiếu (Projection)
- Chọn (Selection)
- **Nối (Joining)**
- **Các phép toán tập hợp (Hợp, giao, trừ)**

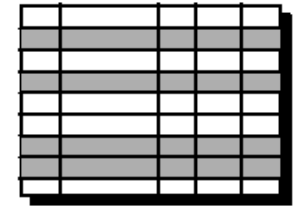
Projection



A 6x6 grid representing a table. The first three columns are shaded gray, indicating they are the result of a projection operation.

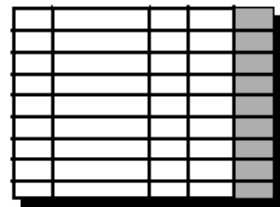
Table 1

Selection



A 6x6 grid representing a table. The first three rows are shaded gray, indicating they are the result of a selection operation.

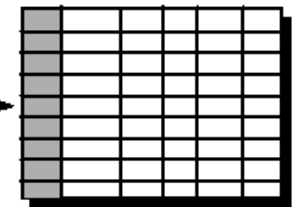
Table 1



A 6x6 grid representing a table. The first three columns and the last three rows are shaded gray, indicating they are the result of a join operation.

Table 1

Join



A 6x6 grid representing a table. The first three columns and the last three rows are shaded gray, indicating they are the result of a join operation.

Table 2

# Phép nối (join)

## Oracle Proprietary Joins (8i and prior):

- Equijoin
- Non-equijoin
- Outer join
- Self join

## SQL: 1999 Compliant Joins:

- Cross joins
- Natural joins
- Using clause
- Full or two sided outer joins
- Arbitrary join conditions for outer joins

# Phép nối (join)

---

- Phân loại:
  - Tích Đề các: cross join
  - Nối bằng: equi-join
    - Nối tự nhiên: natural join
  - Nối không bằng: theta-join
  - Nối trong: inner join
  - Nối ngoài: outer join

# Tích Đề các

Bảng PC

model	speed	hdd	screen	price
1003	2	250	14	1000
1004	2	500	18	900
1005	2.5	500	18	800
1006	2.5	500	18	800

Bảng Product

maker	model	type
A	1002	Laptop
A	1003	PC
B	1004	PC
B	1005	PC
B	1006	PC
B	1007	Laptop

Tính PC x Product

```
SELECT *  
FROM PC, Product
```

# Tích Đề các: PC x Product

model	speed	hdd	screen	price	maker	model	type
1003	2	250	14	1000	A	1002	Laptop
1003	2	250	14	1000	A	1003	PC
1003	2	250	14	1000	B	1004	PC
1003	2	250	14	1000	B	1005	PC
1003	2	250	14	1000	B	1006	PC
1003	2	250	14	1000	B	1007	Laptop
1004	2	500	18	900	A	1002	Laptop
1004	2	500	18	900	A	1003	PC
1004	2	500	18	900	B	1004	PC
1004	2	500	18	900	B	1005	PC
1004	2	500	18	900	B	1006	PC

# Tránh trùng tên bảng và thuộc tính - đặt bí danh

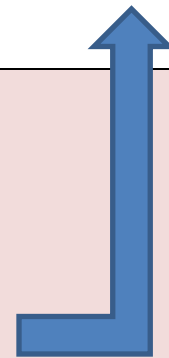
Bảng PC

model	speed	hdd	screen	price
1003	2	250	14	1000
1004	2	500	18	900
1005	2.5	500	18	800
1006	2.5	500	18	800

model 1	model 2
1004	1005
1004	1006
1005	1006

Tìm các cặp PC có ổ cứng bằng nhau:

```
SELECT p1.model [model 1], p2.model [model 2]
FROM PC p1, PC p2
WHERE (p1.model < P2.model ) AND
      (p1.hdd = P2.hdd)
```



# Nối bảng

Bảng PC

model	speed	hdd	screen	price
1003	2	250	14	1000
1004	2	500	18	900
1005	2.5	500	18	800
1006	2.5	500	18	800

Bảng Product

maker	model	type
A	1002	Laptop
A	1003	PC
B	1004	PC
B	1005	PC
B	1006	PC
B	1007	Laptop

Liệt kê chi tiết thông tin về các sản phẩm của các nhà sản xuất

**Product ⋈ PC**

**Product.model =  
PC.model**

# Nối bằng

Product ⋈ PC  
Product.model =  
PC.model

**SELECT \***  
**FROM** PC, Product  
**WHERE** PC.model =  
Product.model

**Cách 1: dùng logic chọn**

**SELECT \***  
**FROM** PC **JOIN** Product  
**ON** PC.model =  
Product.model

**Cách 2: dùng lệnh JOIN**

# Kết quả


```
SELECT *  
FROM PC, Product  
WHERE PC.model =  
        Product.model
```

```
SELECT *  
FROM PC JOIN Product  
        ON PC.model =  
           Product.model
```

model	speed	hdd	screen	price	maker	model	type
1003	2	250	14	1000	A	1003	PC
1004	2	500	18	900	B	1004	PC
1005	2.5	500	18	800	B	1005	PC
1006	2.5	500	18	800	B	1006	PC

# Sử dụng bí danh trong lệnh JOIN

```
SELECT Pr.maker, PC.model, speed, hdd, screen, price
FROM PC JOIN Product Pr
      ON PC.model = Pr.model
```



maker	model	speed	hdd	screen	price
A	1003	2	250	14	1000
B	1004	2	500	18	900
B	1005	2.5	500	18	800
B	1006	2.5	500	18	800

# Nối không bằng

---

- Trong t/h nối không bằng, thì tương tự như t/h nối bằng, ta cũng có thể sử dụng 1 trong 2 cách:
  - Sử dụng điều kiện nối trong mệnh đề WHERE,
  - Hoặc sử dụng lệnh JOIN với điều kiện nối (ON) không bằng

# Nối trong và nối ngoài

---

- Nối trong (INNER JOIN):
  - kết quả chỉ ghép các bộ khớp nhau (matching tuples) trong 2 bảng thành phần (phép JOIN ở trên mặc định chính là INNER JOIN)
- Nối ngoài (OUTER JOIN): Kết quả chứa 2 thành phần:
  - Thành phần 1 như INNER JOIN
  - Thành phần 2 chứa cả các bộ không khớp nhau trong 2 bảng thành phần

# Nối ngoài

---

```
SELECT *  
FROM A OUTER JOIN B  
ON A.X = B.Y;
```

- Phân loại:
  - **Left Outer Join**: chứa các bộ không khớp của bảng bên trái A
  - **Right Outer Join**: chứa các bộ không khớp của bảng bên phải B
  - **Full Outer Join**: chứa các bộ không khớp của cả hai bảng A và B

# Ví dụ

CLASS	
CLASS_ID	CLASS_NAME
106	Lop 106
107	Lop 107
201	Lop 201
202	Lop 202

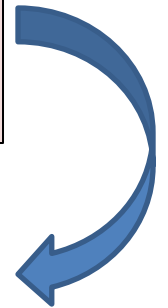
STUDENT		
CLASS_ID	ID	NAME
106	1	A
106	2	B
107	3	C
107	4	D
	5	E
	6	F
	7	G
	8	H

Các bộ  
không khớp

# Left Outer Join

```
SELECT *  
FROM Class LEFT OUTER JOIN Student  
ON Class.Class_ID = Student.Class_ID;
```

CLASS		STUDENT		
CLASS_ID	CLASS_NAME	CLASS_ID	ID	NAME
106	Lop 106	106	1	A
106	Lop 106	106	2	B
107	Lop 107	107	3	C
107	Lop 107	107	4	D
201	Lop 201			
202	Lop 202			



# RIGHT OUTER JOIN

```
SELECT *  
FROM Class RIGHT OUTER JOIN Student  
ON Class.Class_ID = Student.Class_ID;
```

CLASS		STUDENT		
CLASS_ID	CLASS_NAME	CLASS_ID	ID	NAME
106	Lop 106	106	1	A
106	Lop 106	106	2	B
107	Lop 107	107	3	C
107	Lop 107	107	4	D
			5	E
			6	F
			7	G
			8	H

# FULL OUTER JOIN

```
SELECT *  
FROM Class FULL OUTER JOIN Student  
ON Class.Class_ID = Student.Class_ID;
```

CLASS		STUDENT		
CLASS_ID	CLASS_NAME	CLASS_ID	ID	NAME
106	Lop 106	106	1	A
106	Lop 106	106	2	B
107	Lop 107	107	3	C
107	Lop 107	107	4	D
			5	E
			6	F
			7	G
			8	H
201	Lop 201			
202	Lop 202			



# Các phép toán tập hợp

---

- Hợp: UNION và UNION ALL
- Giao: INTERSECT
- Trừ: EXCEPT hoặc MINUS
- ***Lưu ý***: 2 quan hệ cần phải có **tính khả hợp** trước khi thực hiện các phép toán tập hợp.

# Các phép toán tập hợp

---

- Cú pháp:

SELECT  $A_1, A_2, \dots, A_m$

FROM R

**UNION | UNION ALL | EXCEPT | INTERSECT**

SELECT  $B_1, B_2, \dots, B_m$

FROM S

# Tóm tắt

---

- Các phép nối
  - Nối trong
  - Nối ngoài
  - Nối bằng
  - Nối không bằng
  - Tích Đề Các
- Các phép toán tập hợp