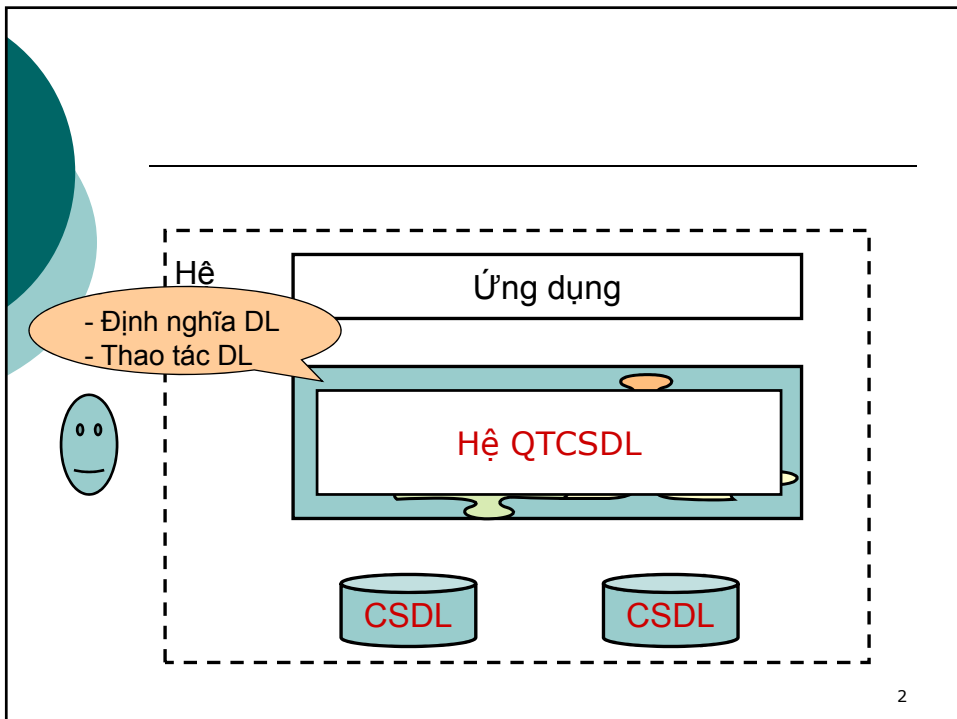


# Các ngôn ngữ dữ liệu đối với mô hình quan hệ

Vũ Tuyết Trinh

[trinhvt@it-hut.edu.vn](mailto:trinhvt@it-hut.edu.vn)

Bộ môn Các hệ thống thông tin, Khoa Công nghệ thông tin  
Đại học Bách Khoa Hà Nội



## Nội dung

---

- Các cách tiếp cận đối với thiết kế ngôn ngữ của CSDL quan hệ
  - Giới thiệu một số ngôn ngữ và phân loại
    - So sánh và đánh giá
- Một số ngôn ngữ dữ liệu mức cao
  - QBE (*Q*uery *B*y *E*xample)
  - SQL (*S*tructured *Q*uery *L*anguage)
- Kết luận

3

## Đặt vấn đề

---

- Mục đích của ngôn ngữ dữ liệu
- Tại sao có nhiều ngôn ngữ dữ liệu?
- Ngôn ngữ cấp thấp vs. Ngôn ngữ cấp cao?

4

## Ví dụ

- Tìm tên của các sinh viên nào sống ở Bundoora

- Tìm các bộ của bảng Student có Suburb = Bundoora
- Đưa ra các giá trị của thuộc tính Name của các bộ này

Student

Id	Name	Suburb
1108	Robert	Kew
3936	Glen	Bundoora
8507	Norman	Bundoora
8452	Mary	Balwyn

5

## Ví dụ (2)

- Tìm các sinh viên đăng ký khoá học có mã số 113

- Tìm các giá trị SID trong bảng Enrol có Course tương ứng là 113
- Đưa các bộ của bảng Student có SID trong các giá trị tìm thấy ở trên

Student

Id	Name	Suburb
1108	Robert	Kew
3936	Glen	Bundoora
8507	Norman	Bundoora
8452	Mary	Balwyn

Enrol

SID	Course
3936	101
1108	113
8507	101

Course

No	Name	Dept
113	BCS	CSCE
101	MCS	CSCE

6



## Phân loại các ngôn ngữ truy vấn

---

- Ngôn ngữ đại số
  - 1 câu hỏi = 1 tập các phép toán trên các quan hệ
  - Được biểu diễn bởi một biểu thức đại số (quan hệ)
- Ngôn ngữ tính toán vị từ
  - 1 câu hỏi = 1 mô tả của các bộ mong muốn
  - Được đặc tả bởi một vị từ mà các bộ phải thoả mãn
  - Phân biệt 2 lớp:
    - ngôn ngữ tính toán vị từ biến bộ
    - ngôn ngữ tính toán vị từ biến miền

7



## Ngôn ngữ đại số quan hệ

---

## Tổng quan

---

- Gồm các phép toán tương ứng với các thao tác trên các quan hệ
- Mỗi phép toán
  - Đầu vào: một hay nhiều quan hệ
  - Đầu ra: một quan hệ
- Biểu thức đại số quan hệ = chuỗi các phép toán
- Kết quả thực hiện một biểu thức đại số là một quan hệ
- Được cài đặt trong phần lớn các hệ CSDL hiện nay

9

## Phân loại các phép toán

---

- Phép toán quan hệ
  - Phép chiếu (*projection*)
  - Phép chọn (*selection*)
  - Phép kết nối (*join*)
  - Phép chia (*division*)
- Phép toán tập hợp
  - Phép hợp (*union*)
  - Phép giao (*intersection*)
  - Phép trừ (*difference*)
  - Phép tích đề-các (*cartesian product*)

10

## Phép chiếu

○ Đ/n: Lựa chọn một số thuộc tính từ một quan hệ.

○ Cú pháp:  $\Pi_{A_1, A_2, \dots}(R)$



❖ Ví dụ: đưa ra danh sách tên của tất cả các sinh viên

$\Pi_{name}(Student)$

Student

Id	Name	Suburb
1108	Robert	Kew
3936	Glen	Bundoora
8507	Norman	Bundoora
8452	Mary	Balwyn

Kết quả

Name
Robert
Glen
Norman
Mary

11

## Phép chọn

○ Đ/n: Lựa chọn các bộ trong một quan hệ thỏa mãn điều kiện cho trước.

○ Cú pháp:  $\sigma_{\langle condition \rangle}(R)$



❖ Ví dụ: đưa ra danh sách những sinh viên sống ở Bundoora  $\sigma_{suburb="Bundoora"}(Student)$

Student

Id	Name	Suburb
1108	Robert	Kew
3936	Glen	Bundoora
8507	Norman	Bundoora
8452	Mary	Balwyn

Kết quả

Id	Name	Suburb
3936	Glen	Bundoora
8507	Norman	Bundoora

12

## Vi dụ - chọn và chiếu

- đưa ra tên của các sinh viên sống ở Bundoora

$$\Pi_{name} (\sigma_{suburb="Bundoora"} Student)$$

Student

Id	Name	Suburb
1108	Robert	Kew
3936	Glen	Bundoora
8507	Norman	Bundoora
8452	Mary	Balwyn



Kết quả

Name
Glen
Norman

13

## Phép kết nối

- Đ/n: ghép các bộ từ 2 quan hệ thỏa mãn điều kiện kết nối  $R_1 \bowtie_{\langle join\_condition \rangle} R_2$

- Cú pháp:



- ❖ Ví dụ: đưa ra danh sách các sinh viên và khoá học  $Student \bowtie_{Id=SID} Enrol$

Student

Id	Name	Suburb
1108	Robert	Kew
3936	Glen	Bundoora
8507	Norman	Bundoora
8452	Mary	Balwyn

Id=SID

Enrol

SID	Course
3936	101
1108	113
8507	101

Kết quả

SID	Id	Name	Suburb	Course
1108	1108	Robert	Kew	113
3936	3936	Glen	Bundoora	101
8507	8507	Norman	Bundoora	101

## Ví dụ - chọn, chiếu và kết nối

- đưa ra tên của các sinh viên sống ở Bundoora và mã khoá học mà sinh viên đó đăng ký

$$\Pi_{name, Course} (\sigma_{suburb="Bundoora"} (Student \bowtie_{Id=SID} Enrol))$$

Student

Id	Name	Suburb
1108	Robert	Kew
3936	Glen	Bundoora
8507	Norman	Bundoora
8452	Mary	Balwyn

Enrol

SID	Course
3936	101
1108	113
8507	101

Kết quả

Name	Course
Glen	101
Norman	101

15

## Phép kết nối tự nhiên

- Đ/n: là phép kết nối với điều kiện bằng trên các thuộc tính trùng tên

❖ Ví dụ:

Takes

SID	SNO
1108	21
1108	23
8507	23
8507	29

Enrol

SID	Course
3936	101
1108	113
8507	101

\*



SID	SNO	Course
1108	21	113
1108	23	113
8507	23	101
8507	29	101

16



## Phép kết nối ngoài

- Phép kết nối ngoài trái



- Phép kết nối ngoài phải



17

## Ví dụ về phép kết nối ngoài

- Đưa ra danh sách mã số các sinh viên và mã khoá học mà sinh viên đó đăng ký nếu có

Student

ID	Name	Suburb
1108	Robert	Kew
3936	Glen	Bundoora
8507	Norman	Bundoora
8452	Mary	Balwyn

Enrol

SID	Course
3936	101
1108	113
8507	101

Kết quả

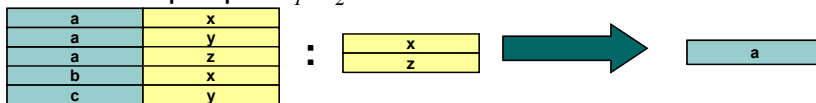
ID	Name	Suburb	Course
1108	Robert	Kew	113
3936	Glen	Bundoora	101
8507	Norman	Bundoora	101
8452	Mary	Balwyn	null

18

## Phép chia

- Đ/n: cho  $R_1$  và  $R_2$  lần lượt là các quan hệ  $n$  và  $m$  ngôi. Kết quả của phép chia  $R_1$  cho  $R_2$  là một quan hệ  $(n-m)$  ngôi

- Cú pháp:  $R_1 : R_2$



- ❖ Ví dụ:

Subject

Name	Course
Systems	BCS
Database	BCS
Database	MCS
Algebra	MCS

Course

Course
BCS
MCS

Kết quả

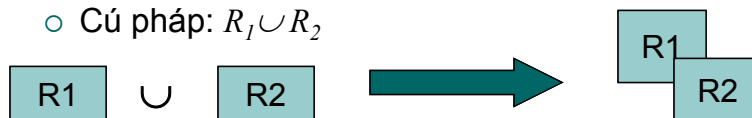
Name
Database

19

## Phép hợp

- Đ/n: gồm các bộ thuộc ít nhất một trong hai quan hệ đầu vào
  - 2 quan hệ khả hợp được xác định trên cùng miền giá trị

- Cú pháp:  $R_1 \cup R_2$



- ❖ Ví dụ:

Subject

Name	Course
Systems	BCS
Database	BCS
Database	MCS
Algebra	MCS

Subject2

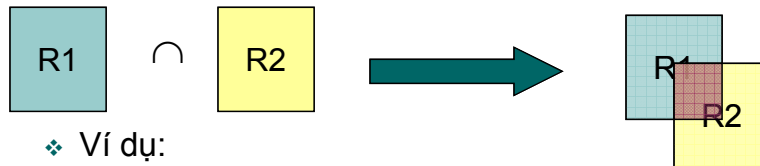
Name	Course
DataMining	MCS
Writing	BCS

Kết quả

Name	Course
Systems	BCS
Database	BCS
Database	MCS
Algebra	MCS
DataMining	MCS
Writing	BCS

## Phép giao

- Đ/n: gồm các bộ thuộc cả hai quan hệ đầu vào
- Cú pháp:  $R_1 \cap R_2$



❖ Ví dụ:

Subject

Name	Course
Systems	BCS
Database	BCS
Database	MCS
Algebra	MCS

Subject2

Name	Course
DataMining	MCS
Database	MCS
Systems	BCS
Writing	BCS

Kết quả

Name	Course
Systems	BCS
Database	MCS

21

## Phép trừ

- Đ/n: gồm các bộ thuộc quan hệ thứ nhất nhưng không thuộc quan hệ thứ hai
  - 2 quan hệ phải là khả hợp
- Cú pháp:  $R_1 \setminus R_2$



❖ Ví dụ:

Subject

Name	Course
Systems	BCS
Database	BCS
Database	MCS
Algebra	MCS

Subject2

Name	Course
DataMining	MCS
Database	MCS
Systems	BCS
Writing	BCS

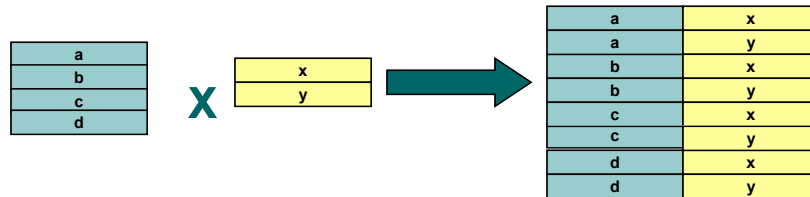
Kết quả

Name	Course
Database	BCS
Algebra	MCS

22

## Phép tích đề-các

- Đ/n: là kết nối giữa từng bộ của quan hệ thứ nhất và mỗi bộ của quan hệ thứ hai
- Cú pháp:  $R_1 \times R_2$



23

## Ví dụ phép tích đề-các

Student

Id	Name	Suburb
1108	Robert	Kew
3936	Glen	Bundoora
8507	Norman	Bundoora
8452	Mary	Balwyn

Sport

SportID	Sport
05	Swimming
09	Dancing

×

Student\_Sport

Id	Name	Suburb	SportID	Sport
1108	Robert	Kew	05	Swimming
3936	Glen	Bundoora	05	Swimming
8507	Norman	Bundoora	05	Swimming
8452	Mary	Balwyn	05	Swimming
1108	Robert	Kew	09	Dancing
3936	Glen	Bundoora	09	Dancing
8507	Norman	Bundoora	09	Dancing
8452	Mary	Balwyn	09	Dancing



## Bài tập

---

- Biểu diễn câu hỏi truy vấn bằng ngôn ngữ đại số quan hệ
- Tính kết quả của biểu thức

25



## Ngôn ngữ tính toán vị từ

---

## Tổng quan

- Ứng dụng logic toán vào CSDL
- Nhắc lại về logic toán
  - **Biểu thức logic**: nhận 1 trong 2 giá trị ĐÚNG hoặc SAI
  - **Biến**: 1 đại lượng biến thiên trong 1 miền giá trị
  - **Hằng**: 1 đại lượng không đổi
  - **Hàm**: 1 ánh xạ từ 1 miền giá trị vào tập hợp gồm 2 giá trị hoặc đúng, hoặc sai
  - **Vị từ**: là 1 biểu thức được xây dựng dựa trên b/t logic
  - **Phép toán logic**: phủ định ( $\neg$ ) kéo theo ( $\Rightarrow$ ), và ( $\wedge$ ) hoặc ( $\vee$ )
  - **Lượng từ**: với mọi ( $\forall$ ), tồn tại ( $\exists$ )
- Phân loại:
  - Phép tính vị từ biến bộ
  - Phép tính vị từ biến miền

27

## Phép tính vị từ biến bộ

- Đ/n: là ngôn ngữ vị từ có biến là các bộ
- Định nghĩa hình thức
$$\{ t \mid P(t) \}$$
  - **t**: tập các bộ kết quả sao cho vị từ **P** là đúng đối với **t**
  - **P**: là một biểu thức có duy nhất 1 biến tự do **t**
- Một số quy ước:
  - **t[A]**: giá trị của bộ **t** tại thuộc tính **A**
  - **t[X]**: giá trị của bộ **t** trên tập các thuộc tính **X**
  - **t ∈ R**: bộ **t** là một bộ trong quan hệ **R**

28

## Biểu thức nguyên tố

- $t \in R$ 
  - $t$  là một biến bộ
  - $R$  là một quan hệ (không sử dụng phép toán  $\notin$ )
- $t[x] \theta u[y]$ 
  - $t$  và  $u$  là các biến bộ
  - $x$  và  $y$  lần lượt là 1 thuộc tính mà trên đó  $t$  và  $u$  được xác định
  - $\theta$  là một phép toán so sánh ( $<$ ,  $=$ ,  $>$ ,  $\leq$ ,  $\neq$ ,  $\geq$ )
- ❖ Ví dụ:
  - $s \in \text{Student}$
  - $e \in \text{Enrol}$
  - $s[\text{Id}] = e[\text{SID}]$

29

## Biểu thức nguyên tố (2)

- $t[x] \theta c$ 
  - $t$  là một biến bộ
  - $x$  là một thuộc tính mà trên đó  $u$  xác định
  - $\theta$  là một phép so sánh
  - $c$  là một hằng trong miền của thuộc tính  $x$ .
- ❖ Ví dụ
  - $s[\text{Suburb}] = \text{''Bundoora''}$

30

## Biểu thức tổng quát

- Một biểu thức nguyên tố là một biểu thức.
- $P1$  là biểu thức  
 $\Rightarrow \neg P1, (P1)$  là các biểu thức
- $P1$  và  $P2$  là biểu thức  
 $\Rightarrow P1 \wedge P2, P1 \vee P2, P1 \Rightarrow P2$  là các biểu thức
- $P1$  là biểu thức chứa 1 biến bộ tự do  $u$ , và  $R$  là 1 quan hệ  
 $\Rightarrow \exists u \in r (P1(u)), \forall u \in r (P1(u))$  cũng là các biểu thức

31

## Các phép biến đổi tương đương

- $P1 \wedge P2 \Leftrightarrow \neg(\neg P1 \vee \neg P2)$
- $t \in r (P1(t)) \Leftrightarrow \neg \exists t \in r (\neg P1(t))$
- $P1 \Rightarrow P2 \Leftrightarrow \neg P1 \vee P2$

32



## Bài tập

- Biểu diễn câu hỏi bằng ngôn ngữ tính toán vị từ biến bộ

33

## Tính an toàn của các biểu thức

- Đặt vấn đề:  
 $\{t \mid \neg(t \in r)\}$
- K/n miền giá trị của biểu thức: **DOM(P)**
  - Các hằng xuất hiện trong **P**
  - Các giá trị của các thuộc tính của các bộ của các quan hệ xuất hiện trong **P**
- ❖ Ví dụ:  **$P(t) = t \in \text{Sport} \wedge t[\text{Sport}] \neq \text{“Football”}$**

Sport

SportID	Sport
05	Swimming
09	Dancing

$\text{DOM}(P) = \{\text{“05”}, \text{“09”}, \text{“Swimming”}, \text{“Dancing”}, \text{“Football”}\}$

34

## Biểu thức an toàn

- Đ/n :  $\{t \mid P(t)\}$  là an toàn nếu tất cả các giá trị xuất hiện trong kết quả là các giá trị từ **DOM(P)**
- ❖ Ví dụ
  - ✓  $P(t) = t \in \text{Sport} \wedge t[\text{Sport}] \neq \text{“Football”}$  (an toàn)
  - ×  $P(t) = \neg(t \in \text{Sport})$  (không an toàn)

35

## Phép tính vị từ biến miền

- Đ/n: là ngôn ngữ vị từ có biến là các miền giá trị
- Định nghĩa hình thức

$$\{ \langle x_1, \dots, x_n \rangle \mid P(x_1, \dots, x_n) \}$$

- $x_1, \dots, x_n$  là các biến miền hay các hằng miền
- $P$ : là một biểu thức chỉ có các biến tự do  $x_i$

36

## Các biểu thức nguyên tố

- $\langle x_1, \dots, x_n \rangle \in r$ 
  - $r$  là 1 quan hệ trên  $n$  thuộc tính
  - $x_1, \dots, x_n$  là các biến miền hay các hằng miền.
- $x \theta y$ 
  - $x$  và  $y$  là các biến miền
  - $\theta$  là một phép so sánh đơn giản ( $<, =, >, \leq, \neq, \geq$ ).
- ❖ Ví dụ

$\langle x, y, z \rangle \in \text{Student}$

$\langle u, v \rangle \in \text{Enrol}$

$x = u$

37

## Các biểu thức nguyên tố (2)

- $x \theta c$ 
  - $x$  là một biến miền
  - $\theta$  là một phép so sánh
  - $c$  là một hằng trong miền của thuộc tính của  $x$
- ❖ Ví dụ

$Z = \text{'Bundoora'}$

38

## Biểu thức tổng quát

---

- Một biểu thức nguyên tố là một biểu thức.
- $P1$  là 1  
 $\Rightarrow \neg P1, (P1)$  là biểu thức
- $P1$  và  $P2$  là biểu thức  
 $\Rightarrow P1 \wedge P2, P1 \vee P2, P1 \Rightarrow P2$  là biểu thức
- $P1(x)$  là một biểu thức  
 $\Rightarrow \exists x (P1(x)), \forall x (P1(x))$  là biểu thức

39

## Tính an toàn của các biểu thức

---

- Đ/n: một biểu thức  $\{ \langle x_1, \dots, x_n \rangle \mid P(x_1, \dots, x_n) \}$  là an toàn nếu tất cả các giá trị xuất hiện trong kết quả là các giá trị từ  $\text{DOM}(P)$

40



## Bài tập

---

- Biểu diễn câu hỏi bằng ngôn ngữ tính toán vị từ biến miền

41



## Nhận xét

---

- Sự tương đương của 3 ngôn ngữ
  - Đại số quan hệ
  - Phép tính vị từ biến bộ hạn chế với các biểu thức an toàn
  - Phép tính vị từ biến miền hạn chế với các biểu thức an toàn
- So sánh đặc điểm của 3 ngôn ngữ

42



## Khả năng bổ sung của các ngôn ngữ

---

- Tính toán số học: các phép toán số học  $+$ ,  $-$ ,  $*$ ,  $/$
- Lệnh gán và hiển thị: hiển thị quan hệ kết quả hay gán một quan hệ đã được tính toán đến một tên quan hệ khác.
- Hàm tập hợp: tính giá trị trung bình, tính tổng, chọn giá trị nhỏ nhất hay lớn nhất

43

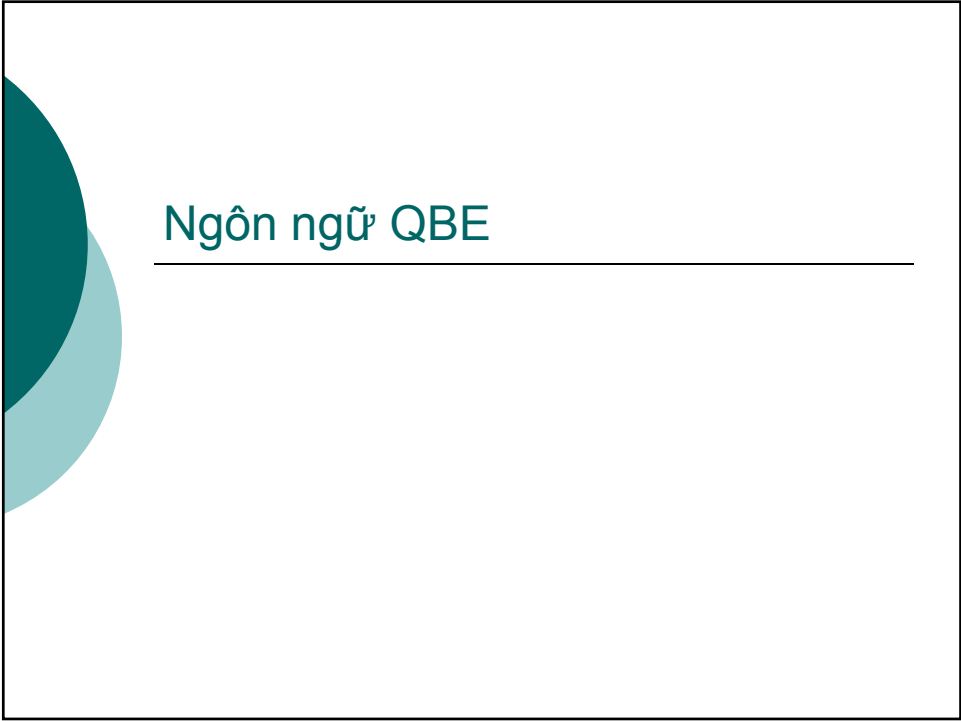
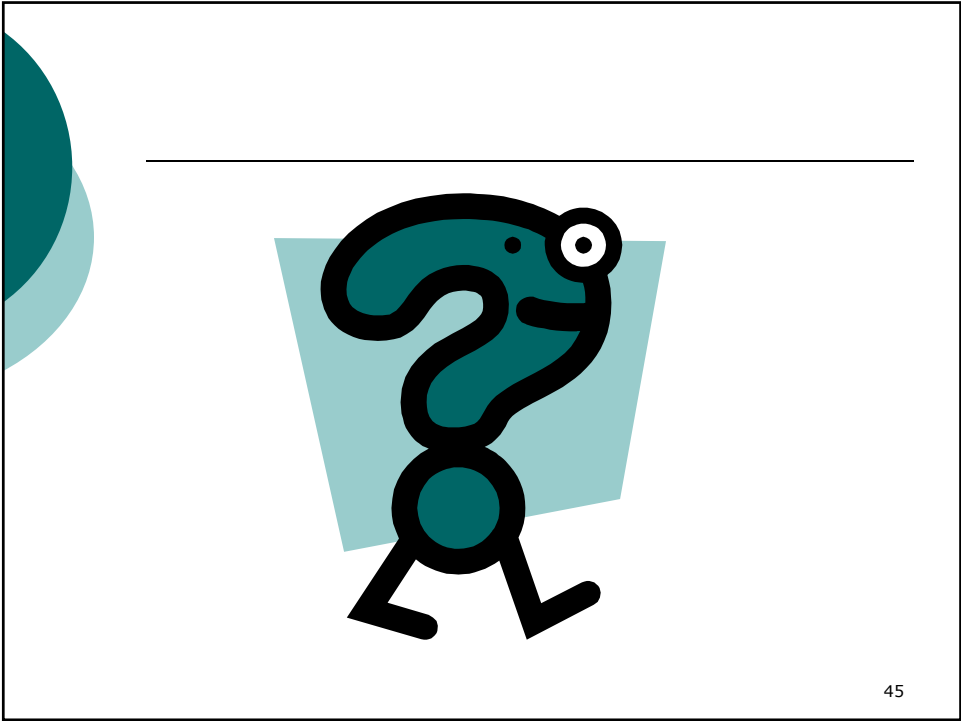


## Bài tập biến đổi tương đương

---

- Viết định nghĩa các phép toán ĐSQH với các biểu thức tính toán vị từ
- Tìm biểu thức tương đương trong ngôn ngữ ĐSQH và ngôn ngữ tính toán vị từ

44



Ngôn ngữ QBE

## QBE (Query-By-Example)

- Là một ngôn ngữ truy vấn dữ liệu
- Các câu truy vấn được thiết lập bởi một giao diện đồ họa
- Phù hợp với các câu truy vấn đơn giản, tham chiếu đến ít bảng
- Một số sản phẩm: IBM™ (IBM Query Management Facility), Paradox, MS. Access, ...

47

## Truy vấn trên một quan hệ

Student	ID	Name	Suburb
		P._x	Bundoora

- P.~ Print
- phép tính vị từ biến miền tương đương:  
 $\{ \langle x \rangle \mid \exists i, x, s (i, x, s) \in \text{Student} \wedge s = \text{"Bundoora"} \}$
- Biểu thức đại số quan hệ tương đương  
 $\sigma_{\text{suburb}=\text{"Bundoora"}}(\text{Student})$

48



## Truy vấn trên một quan hệ (tiếp)

- Lựa chọn tất cả các cột

Student	ID	Name	Suburb
P.			Bundoora

- Sắp xếp

Student	ID	Name	Suburb
		P.AO(1)	P.AO(2)

- AO: sắp xếp tăng dần
- DO: sắp xếp giảm dần

49

## Các truy vấn trên nhiều quan hệ

- Đưa ra tên của các sinh viên có đăng ký ít nhất một khoá học

Student	ID	Name	Suburb
	_id	P._name	

Enrol	SID	Course
	_id	

- Đưa ra tên các sinh viên không đăng ký một khoá học nào

Student	ID	Name	Suburb
	_id	P._name	

Enrol	SID	Course
¬	_id	

50

## Các tính toán tập hợp

- Các phép toán: AVG, COUNT, MAX, MIN, SUM
- Ví dụ: đưa ra tên các thành phố và số lượng sinh viên đến từ thành phố đó

Student	ID	Name	Suburb	
	_id		G.P.	P.COUNT._id

- G. ~ Grouping

51

## Hộp điều kiện

- Được sử dụng để biểu diễn
  - Điều kiện trên nhiều hơn 1 thuộc tính
  - Điều kiện trên các trường tính toán tập hợp
- Ví dụ: đưa ra danh sách các thành phố có nhiều hơn 5 sinh viên

Student	ID	Name	Suburb	Condition
	_id		G.P.	COUNT._id > 5

52

## Các thao tác thay đổi dữ liệu

- Xoá

Student	ID	Name	Suburb
D.	1108		

- Thêm

Student	ID	Name	Suburb
I.	1179	David	Evry

- Sửa đổi

Student	ID	Name	Suburb
	1179		U.Paris

53

## Tính đầy đủ của QBE

- Có thể biểu diễn cả 5 phép toán đại số cơ sở ( $\sigma, \Pi, \cup, \setminus, \times$ )
- Bài tập: chứng minh tính đầy đủ của QBE

54

## Định nghĩa dữ liệu trong QBE

- sử dụng cùng qui cách và giao diện đồ họa như đối với truy vấn.

I.Student	I.	ID	Name	Suburb
KEY	I.	Y	N	N
TYPE	I.	CHAR(5)	CHAR(30)	CHAR(30)
DOMAIN	I.	Sid	SName	Surb
INVERSION	I.	Y	N	N

55

## Định nghĩa dữ liệu trong QBE (2)

- Các khung nhìn

I.View V	I.	ID	Name	Course
	I.	_id	_name	_course

Student	ID	Name	Suburb	Enrol	SID	Course
	_id	_name			_id	_course

56



## Ngôn ngữ SQL

---



## SQL (*Structured Query Language*)

---

- 1975: SEQUEL
  - System-R
- 1976: SEQUEL2
- 1978/79: SQL
  - System-R
- 1986: chuẩn SQL-86
- 1989: chuẩn SQL-89
- 1992: chuẩn SQL-92
- 1996: chuẩn SQL-96

## Các thành phần của SQL

- Ngôn ngữ mô tả dữ liệu (Data Definition Language)
  - Cấu trúc các bảng CSDL
  - Các mối liên hệ của dữ liệu
  - Quy tắc, ràng buộc áp đặt lên dữ liệu
- Ngôn ngữ thao tác dữ liệu (Data Manipulation Language)
  - Thêm, xoá, sửa dữ liệu trong CSDL
- Ngôn ngữ quản lý dữ liệu (Data Control Language)
  - Thay đổi cấu trúc của các bảng dữ liệu
  - Khai báo bảo mật thông tin
  - Quyền hạn của người dùng trong khai thác CSDL

59

## Cú pháp câu lệnh truy vấn SQL

```
SELECT [DISTINCT] <bt1>, <bt2>, ...  
FROM      <bang1>,<bang2>, ...  
[WHERE   <dieu kien chon>]  
[GROUP BY <tt1>, <tt2>, ...]  
[ORDER BY <tt1> | <bieu thuc so 1> [ASC | DESC]]  
[HAVING <dieu kien in ket qua>]
```

60

## Truy vấn đơn giản trên 1 bảng

- Tìm thông tin từ các cột của bảng

> **SELECT** ColumnName, ColumnName, ...  
**FROM** TableName;  
> **SELECT** \*  
**FROM** TableName;

- Ví dụ

**SELECT** Name  
**FROM** Student;

Student

Id	Name	Suburb
1108	Robert	Kew
3936	Glen	Bundoora
8507	Norman	Bundoora
8452	Mary	Balwyn

$\Pi_{name}(Student)$

Name
Robert
Glen
Norman
Mary

61

## Truy vấn với điều kiện lựa chọn

- Chọn các bản ghi (dòng)

**SELECT** ColumnName, ColumnName, ...  
**FROM** TableName  
**WHERE** condition\_expression;

- Ví dụ

**SELECT** \*  
**FROM** Student  
**WHERE** suburb='Bundoora';

Student

Id	Name	Suburb
1108	Robert	Kew
3936	Glen	Bundoora
8507	Norman	Bundoora
8452	Mary	Balwyn

$\sigma_{suburb='Bundoora'}(Student)$

Id	Name	Suburb
3936	Glen	Bundoora
8507	Norman	Bundoora

62

## Biểu diễn điều kiện lựa chọn

---

- Các phép toán quan hệ: =, !=, <, >, <=, >=
- Các phép toán logic: NOT, AND, OR
- Phép toán phạm vi: BETWEEN, IN, LIKE
  - Kiểu dữ liệu số
    - attr **BETWEEN** val1 **AND** val2 ( $\Leftrightarrow$  (attr>=val1) and (attr<=val2) )
    - attr **IN** (val1, val2, ...) ( $\Leftrightarrow$  (attr=val1) or (attr=val2) or ... )
  - Kiểu dữ liệu chuỗi
    - **LIKE**: sử dụng đối sánh mẫu chuỗi với các ký tự % (thay thế cho 1 ký tự bất kỳ), \* (thay thế cho 1 chuỗi ký tự bất kỳ)

63

## Bài tập

---

- Viết câu lệnh SQL đưa ra danh sách tên sinh viên học môn "Database" hoặc môn "VB"
- Viết câu lệnh SQL đưa ra danh sách các sinh viên đăng ký các khoá học có mã 113 hoặc 101
- Đưa ra danh sách các khoá học (Course) mà tên của khoá học chứa cụm "CS"

64



## Ví dụ

Student			Takes		Enrol	
Id	Name	Suburb	SID	SNO	SID	Course
1108	Robert	Kew	1108	21	3936	101
3936	Glen	Bundoora	1108	23	1108	113
8507	Norman	Bundoora	1108	29	8507	101
8452	Mary	Balwyn	8507	23		
			8507	29		

Course			Subject		
No	Name	Dept	No	Name	Dept
113	BCS	CSCE	21	Systems	CSCE
101	MCS	CSCE	23	Database	CSCE
			29	VB	CSCE
			18	Algebra	Maths

65

## Truy vấn phức tạp trên nhiều bảng

- Điều kiện kết nối

```
SELECT  T1.C1,T1.C2,T2.C1,T2.C4, ...  
FROM    T1, T2  
WHERE   condition_expression
```

- Ví dụ: đưa ra danh sách mã sinh viên (Id), tên sinh viên (Name), thành phố (Suburb), mã khoá học (Course) mà các sinh viên đã đăng ký

```
SELECT  Id, Name, Suburb, Course  
FROM    Student, Enrol  
WHERE   Id = SID
```

66

## Phép toán đổi tên

---

- Từ khoá **AS**

```
SELECT    c1 as name1, c2 as name2
FROM      TableName
```

- Ví dụ: đưa ra danh sách mã sinh viên, tên sinh viên và tên môn học mà sinh viên đó tham gia

```
SELECT    SID, Student.Name as SName,
            Subject.Name as Subject
FROM      Student, Takes, Subject
WHERE     (Id=SID) and (SNO = No)
```

67

## Sử dụng biến bộ trong SQL

---

- Sử dụng từ khoá **AS** trong mệnh đề FROM

- Ví dụ

```
SELECT    SID, Stud.Name as SName,
            Sub.Name as Subject
FROM      Student as Stud, Takes,
            Subject as Sub
WHERE     (Id=SID) and (SNO = No)
```

68

## Loại trừ các bản ghi trùng nhau

- Từ khoá **DISTINCT**

```
SELECT DISTINCT <bt1>, <bt2>, ...  
FROM <bang1>, <bang2>, ...
```

- Ví dụ: đưa ra danh sách tên các khoa (dept) tương ứng với các khoá học (Course). Mỗi giá trị chỉ hiện thị một lần

```
SELECT DISTINCT Dept  
FROM Course
```

69

## Tìm kiếm có sắp xếp

- Sắp xếp các bản ghi kết quả theo một thứ tự cho trước

```
SELECT <bt1>, <bt2>, ...  
FROM <bang1>, <bang2>, ...  
[WHERE <dieu kien chon>  
ORDER BY <tt1> | <bieu thuc so 1> [ASC | DESC]
```

- Ví dụ: đưa ra danh sách tên các sinh viên theo thứ tự tăng dần

```
SELECT Name  
FROM Student  
ORDER BY Name ASC
```

70

## Phân nhóm các bản ghi kết quả

- Phân nhóm các bản ghi kết quả theo giá trị của 1 hoặc nhiều thuộc tính

```
SELECT      <bt1>, <bt2>, ...
FROM        <bang1>,<bang2>, ...
[WHERE      <dieu kien chon>]
[GROUP BY  <tt1>, <tt2>, ...]
```

- Ví dụ đưa ra tên các sinh viên nhóm theo thành phố của sinh viên đó

```
SELECT      Suburb, Name
FROM        Student
GROUP BY    Suburb
```

```
SELECT      Suburb, Count(Id)
FROM        Student
GROUP BY    Suburb
```

71

## Điều kiện hiển thị các bản ghi kết quả

- Lựa chọn các bản ghi kết quả để hiển thị

```
SELECT      <bt1>, <bt2>, ...
FROM        <bang1>,<bang2>, ...
[WHERE      <dieu kien chon>]
HAVING      <dieu kien in ket qua>
```

- Ví dụ: đưa ra tên các thành phố có nhiều hơn 3 sinh viên

```
SELECT      Suburb, COUNT(ID)
FROM        Student
GROUP BY    Suburb
HAVING      COUNT(ID) > 3
```

72

## Các phép toán tập hợp

- **UNION, MINUS, INTERSECT**
- Ví dụ: đưa ra danh sách tên các môn học không có sinh viên nào tham dự

```
SELECT DISTINCT Subject.Name
FROM Subject
MINUS
SELECT DISTINCT Subject.Name
FROM Student, Takes, Subject
WHERE Student.Id = Takes.SID and
Takes.SNO = Subject.No
```

73

## Các câu truy vấn lồng nhau

- Là trường hợp các câu truy vấn (con) được viết lồng nhau
- Thường được sử dụng với để
  - Kiểm tra thành viên tập hợp (**IN**)
  - So sánh tập hợp (**>ALL, >=ALL, <ALL, <=ALL, =ALL, NOT IN, SOME, )**
  - Kiểm tra các bảng rỗng (**EXISTS** hoặc **NOT EXISTS**)
- Các truy vấn con lồng nhau thông qua mệnh đề **WHERE**

74



## Các hàm thư viện

---

- Hàm tính toán trên nhóm các bản ghi
  - MAX/MIN
  - SUM
  - AVR
  - COUNT

75



## Các hàm thư viện (2)

---

- Hàm tính toán trên bản ghi
  - Hàm toán học: ABS, SQRT, LOG, EXP, SIGN, ROUND
  - Hàm xử lý chuỗi ký tự: LEN, LEFT, RIGHT, MID
  - Hàm xử lý thời gian: DATE, DAY, MONTH, YEAR, HOUR, MINUTE, SECOND
  - Hàm chuyển đổi kiểu giá trị: FORMAT

76

## Bài tập

---

- Viết các câu lệnh SQL biểu diễn các câu hỏi truy vấn
- Viết biểu thức đại số quan hệ tương đương với các câu lệnh SQL

77

## Các câu lệnh cập nhật dữ liệu

---

- Thêm
  - > **INSERT INTO** table[(col1,col2,...)]  
**VALUES** (exp1,exp2,...)
  - > **INSERT INTO** table[(col1,col2,...)]  
**SELECT** col1,col2, ...  
**FROM** tab1, tab2, ...  
**WHERE** <dieu\_kien>
- Ví dụ
  - > **INSERT INTO** Student[Id, Name, Suburb]  
**VALUES** ('1179','David','Evr')

78

## Các câu lệnh cập nhật dữ liệu (2)

- Xoá

```
DELETE FROM    table  
WHERE          cond_exp;
```

- Sửa

```
UPDATE    table  
SET      col1 = exp1,  
          col2=exp2,  
          col2=exp2,  
WHERE    cond_exp;
```

- Ví dụ

- **DELETE FROM** Student  
**WHERE** Suburb = "Bundoora";
- **UPDATE** Student  
**SET** Suburb = "Evry"  
**WHERE** Suburb = "Evr";

79

## Định nghĩa dữ liệu với SQL

- Các thông tin được định nghĩa bao gồm

- Sơ đồ quan hệ
- Kiểu dữ liệu hay miền giá trị của mỗi thuộc tính
- Các ràng buộc toàn vẹn
- Các chỉ số đối với mỗi bảng
- Thông tin an toàn và uỷ quyền đối với mỗi bảng
- Cấu trúc lưu trữ vật lý của mỗi bảng trên đĩa

- Được biểu diễn bởi các lệnh định nghĩa dữ liệu

80



## Cú pháp

---

- Tạo bảng

```
CREATE TABLE tab(  
  col1 type1(size1) [NOT NULL], ...,  
  col2 type2(size2) [NOT NULL], ...,  
  .....  
  [CONSTRAINT <constraint name> <constraint type> clause]  
  ...  
)
```

- Xoá bảng

```
DROP TABLE tab
```

81

## Quy ước đặt tên và kiểu dữ liệu

---

- Quy ước đặt tên
  - 32 ký tự: chữ cái, số, dấu \_
- Kiểu dữ liệu (SQL-92)
  - CHAR(n)
  - VARCHAR(n)
  - Int
  - Smallint
  - Numeric(p,d)
  - Real, double
  - float(n)
  - Date
  - time

82

## Kiểu ràng buộc

---

- RBTV về giá trị miền  
**CONSTRAINT** <name>  
**CHECK** <condition>
- RBTV về khoá chính  
**CONSTRAINT** <name> **PRIMARY KEY** (fk1,fk2,...)
- RBTV về khoá ngoại hay phụ thuộc tồn tại  
**CONSTRAINT** <name> **FOREIGN KEY** (fk1,fk2,...)  
**REFERENCES** tab(k1,k2)

83

## Thêm/xoá/sửa cột của các bảng

---

- Thêm  
**ALTER TABLE** <tên bảng>  
**ADD COLUMN** <tên cột> <kiểu dữ liệu> [NOT NULL]
- Xoá  
**ALTER TABLE** <tên bảng>  
**DROP COLUMN** <tên cột>
- Sửa  
**ALTER TABLE** <tên bảng>  
**CHANGE COLUMN** <tên cột> <kiểu dữ liệu mới>

84

## Thêm/sửa các ràng buộc

- Thêm

**ALTER TABLE** <tên bảng>

**ADD CONSTRAINT** <tên ràng buộc> <kiểu ràng buộc>

- Sửa

**ALTER TABLE** <tên bảng>

**DROP CONSTRAINT** <tên ràng buộc>

85

## Ví dụ

```
CREATE TABLE Student(  
  Id char(10) [NOT NULL],  
  Name char(30) [NOT NULL],  
  Suburb char(30),  
  CONSTRAINT key_Stud  
    PRIMARY KEY Id  
)
```

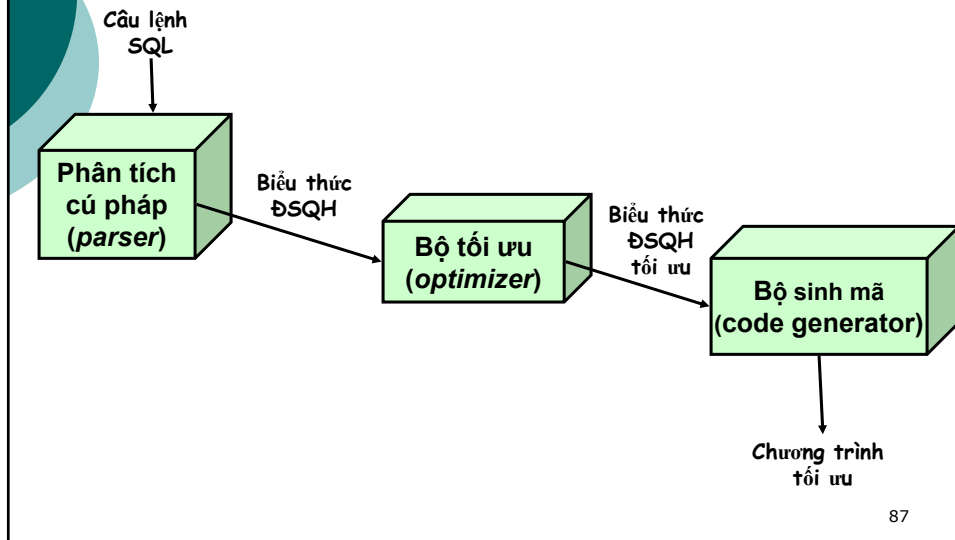
Id	Name	Suburb
1108	Robert	Kew
3936	Glen	Bundoora
8507	Norman	Bundoora
8452	Mary	Balwyn

```
CREATE TABLE Takes(  
  SID char(10) [NOT NULL],  
  SNO char(5) [NOT NULL],  
  CONSTRAINT key_takes  
    PRIMARY KEY (SID,SNO),  
  CONSTRAINT key_2Stud  
    FOREIGN KEY (SID)  
    REFERENCES Student(Id)  
)
```

SID	SNO
1108	21
1108	23
8507	23
8507	29

86

## Xử lý câu hỏi truy vấn

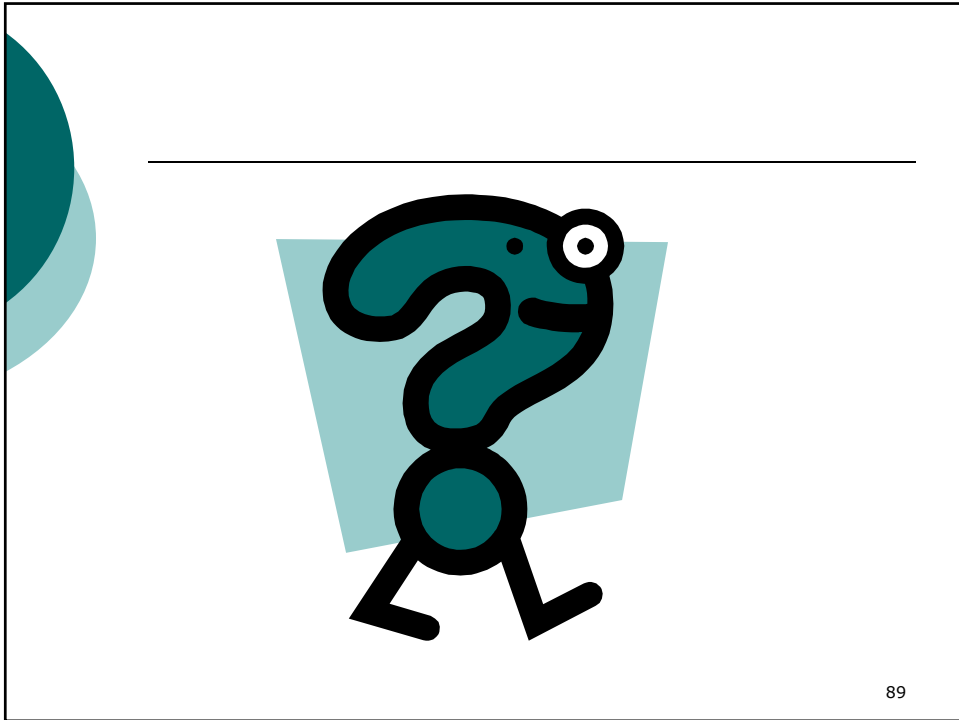


87

## Các điểm cần lưu ý

- Các ngôn ngữ dữ liệu
  - ĐSQH vs. vị từ
  - QBE vs. SQL
- Sự tương đương của các ngôn ngữ
  - Ngôn ngữ ĐSQH và ngôn ngữ vị từ
  - Biến đổi giữa câu truy vấn SQL và biểu thức đại số quan hệ

88



## Ví dụ

Student

Id	Name	Suburb
1108	Robert	Kew
3936	Glen	Bundoora
8507	Norman	Bundoora
8452	Mary	Balwyn

Takes

SID	SNO
1108	21
1108	23
8507	23
8507	29

Enrol

SID	Course
3936	101
1108	113
8507	101

Course

No	Name	Dept
113	BCS	CSCE
101	MCS	CSCE

Subject

No	Name	Dept
21	Systems	CSCE
23	Database	CSCE
29	VB	CSCE
18	Algebra	Maths