

INTRODUCTION TO DATABASE III

Based on materials by Kevin C. Chang

Running Example

2

- เราจะใช้ตัวอย่างเดิมจากครั้งก่อน
 - Students(name, addr, tel)
 - Courses(name, type)
 - Depts(name, addr, tel)
 - Enroll(student, course)
 - Major(student, dept)
 - Offer(dept, course, credit)
- ในวงเล็บคือ **Attribute** ของตารางนั้น ๆ
- สังเกตว่าความสัมพันธ์จะมี **2 Key** เนื่องจากเชื่อม **2** ความสัมพันธ์

Join Review

3

- สมมติว่าในตาราง **Enroll** และ **Major** มีข้อมูลดังนี้

<u>student</u>	<u>course</u>
John Smith	IT2
John Smith	Math SC
Jacky Chan	Math SC

<u>student</u>	<u>dept</u>
John Smith	Maths
Jacky Chan	Stats

Join Review

4

- ❑ `SELECT * FROM Enroll, Offer`
- ❑ ผลลัพธ์ที่ได้จะเป็น **Cross Join**
- ❑ ใน PHP `$row["student"]` จะมีค่าเป็นของคอลัมน์ที่สาม

<u>student</u>	<u>course</u>	<u>student</u>	<u>dept</u>
John Smith	IT2	John Smith	Maths
John Smith	Math SC	John Smith	Maths
Jacky Chan	Math SC	John Smith	Maths
John Smith	IT2	Jacky Chan	Stats
John Smith	Math SC	Jacky Chan	Stats
Jacky Chan	Math SC	Jacky Chan	Stats

<u>student</u>	<u>course</u>
John Smith	IT2
John Smith	Math SC
Jacky Chan	Math SC

<u>student</u>	<u>dept</u>
John Smith	Maths
Jacky Chan	Stats

Join Review

5

- ❑ `SELECT * FROM Enroll, Offer WHERE Enroll.student = Major.student`
- ❑ ผลลัพธ์ที่ได้จะเป็น Inner Join

<u>student</u>	<u>course</u>	<u>student</u>	<u>dept</u>
John Smith	IT2	John Smith	Maths
John Smith	Math SC	John Smith	Maths
Jacky Chan	Math SC	Jacky Chan	Stats

<u>student</u>	<u>course</u>
John Smith	IT2
John Smith	Math SC
Jacky Chan	Math SC

<u>student</u>	<u>dept</u>
John Smith	Maths
Jacky Chan	Stats

Join Review

6

- `SELECT dept FROM Enroll, Offer WHERE Enroll.student = Major.student`
- ผลลัพธ์ที่ได้จะมี **Maths** ซ้ำกันสองครั้ง

<u>dept</u>
Maths
Maths
Stats

<u>student</u>	<u>course</u>
John Smith	IT2
John Smith	Math SC
Jacky Chan	Math SC

<u>student</u>	<u>dept</u>
John Smith	Maths
Jacky Chan	Stats

Join Review

7

- **SELECT DISTINCT** dept FROM Enroll, Offer WHERE Enroll.student = Major.student
- ใช้คำสั่ง **DISTINCT** เพื่อกำกั้ดตัวซ้ำ

dept
Maths
Stats

<u>student</u>	<u>course</u>
John Smith	IT2
John Smith	Math SC
Jacky Chan	Math SC

<u>student</u>	<u>dept</u>
John Smith	Maths
Jacky Chan	Stats

INSERT Statement

8

- **INSERT** เป็นคำสั่งสำหรับการเพิ่มข้อมูลในตาราง โดยมีโครงสร้างดังนี้

INSERT INTO ชื่อตาราง

VALUES ('ค่าที่1', 'ค่าที่2',...)

- ซึ่งแต่ละค่านั้นจะต้องเรียงตาม **Attribute** ในตารางนั้นให้ครบ

INSERT Statement Example

9

- ถ้าต้องการเพิ่มชื่อนักศึกษาคนใหม่ที่ชื่อ Johnny Depp

```
INSERT INTO Students  
VALUES ('Johnny Depp',  
'Kentucky', '502-564-4581')
```

```
Students(name, addr, tel)  
Courses(name, type)  
Depts(name, addr, tel)  
Enroll(student, course)  
Major(student, dept)  
Offer(dept, course, credit)
```

INSERT Statement Example

10

- ถ้า Johnny Depp เลือกวิชาเอกเป็นสถิติ

INSERT INTO Major
VALUES ('Johnny Depp', 'Stats')

```
Students(name, addr, tel)  
Courses(name, type)  
Depts(name, addr, tel)  
Enroll(student, course)  
Major(student, dept)  
Offer(dept, course, credit)
```

INSERT Statement Example

11

- ถ้า Johnny Depp ลงวิชา IT2

```
INSERT INTO Enroll  
VALUES ('Johnny Depp', 'IT2')
```

```
Students(name, addr, tel)  
Courses(name, type)  
Depts(name, addr, tel)  
Enroll(student, course)  
Major(student, dept)  
Offer(dept, course, credit)
```

INSERT Statement Example

12

- Tom Hanks เป็นนักศึกษาใหม่เอกฟิสิกส์และลงเรียนวิชา **Quantum**

INSERT INTO Students

VALUES ('Tom Hanks',
'California', '925-671-CITY');

INSERT INTO Major

VALUES ('Tom Hanks', 'Physics');

INSERT INTO Enroll

VALUES ('Tom Hanks', 'Quantum');

```
Students(name, addr, tel)
Courses(name, type)
Depts(name, addr, tel)
Enroll(student, course)
Major(student, dept)
Offer(dept, course, credit)
```

DELETE Statement

13

- DELETE เป็นคำสั่งสำหรับการลบข้อมูลในตาราง โดยมีโครงสร้างดังนี้

DELETE FROM ชื่อตาราง

WHERE เงื่อนไขของบรรทัดที่ต้องการลบ

- ถ้าไม่ใส่เงื่อนไขที่ต้องการลบ จะถือว่าลบทุกแถวของตาราง

DELETE Statement Example

14

- ถ้า John Smith ดรอปปวิชา IT2

```
DELETE FROM Enroll
WHERE student = 'John Smith'
AND course = 'IT2'
```

```
Students(name, addr, tel)
Courses(name, type)
Depts(name, addr, tel)
Enroll(student, course)
Major(student, dept)
Offer(dept, course, credit)
```

DELETE Statement Example

15

- ถ้า IT2 ปิดครอสเนื่องจากจำนวนนักเรียนไม่พอ

```
DELETE FROM Offer
WHERE course = 'IT2';
DELETE FROM Enroll
WHERE course = 'IT2';
```

```
Students(name, addr, tel)
Courses(name, type)
Depts(name, addr, tel)
Enroll(student, course)
Major(student, dept)
Offer(dept, course, credit)
```

DELETE Statement Example

16

- ถ้า John Smith ลาออกจากการเป็นนักศึกษา

```
DELETE FROM Students
WHERE name = 'John Smith';
DELETE FROM Enroll
WHERE student = 'John Smith';
DELETE FROM Major
WHERE student = 'John Smith';
```

```
Students(name, addr, tel)
Courses(name, type)
Depts(name, addr, tel)
Enroll(student, course)
Major(student, dept)
Offer(dept, course, credit)
```


DELETE Statement Remark

17

- โดยทั่วไปแล้วเราการลบข้อมูลออกจากฐานข้อมูลจะต้องคำนึงถึงความสัมพันธ์ของข้อมูลนั้นด้วย มิเช่นนั้นแล้วอาจจะเกิดปัญหาข้อมูลไม่ถูกต้อง
- หากไม่จำเป็น ไม่ควรลบข้อมูลจากตารางที่เป็น **Entity**
- ข้อมูลที่ลบแล้วไม่สามารถกู้คืนหรือ **Undo** ได้ดังนั้นควรตรวจสอบ **Query** ให้ดีก่อนใช้งาน

UPDATE Statement

18

- UPDATE เป็นคำสั่งสำหรับการแก้ไขข้อมูลในตาราง โดยมีโครงสร้างดังนี้

UPDATE ชื่อตาราง

SET attribute 1='value 1', attribute 2='value 2', ...

WHERE เงื่อนไขของบรรทัดที่ต้องการแก้ไข

- ถ้าไม่ใส่เงื่อนไขที่ต้องการลบ จะถือว่าลบทุกแถวของตาราง

UPDATE Statement Example

19

- ถ้า John Smith เปลี่ยนเบอร์โทรเป็น 123-456-7890

UPDATE Students

SET tel = '123-456-7890'

WHERE name = 'John Smith'

Students(name, addr, tel)

Courses(name, type)

Depts(name, addr, tel)

Enroll(student, course)

Major(student, dept)

Offer(dept, course, credit)

UPDATE Statement Example

20

- ถ้า John Smith เปลี่ยนวิชาเอกเป็นคณิตศาสตร์

```
UPDATE Major  
SET dept = 'Maths'  
WHERE student = 'John Smith'
```

```
Students(name, addr, tel)  
Courses(name, type)  
Depts(name, addr, tel)  
Enroll(student, course)  
Major(student, dept)  
Offer(dept, course, credit)
```

UPDATE Statement Example

21

- ถ้า John Smith เปลี่ยนชื่อเป็น Jonathan Smith

UPDATE Students

SET name = 'Jonathan Smith'

WHERE name = 'John Smith';

UPDATE Enroll

SET student = 'Jonathan Smith'

WHERE student = 'John Smith';

UPDATE Major SET student = 'Jonathan Smith' WHERE
student = 'John Smith'

```
Students(name, addr, tel)
Courses(name, type)
Depts(name, addr, tel)
Enroll(student, course)
Major(student, dept)
Offer(dept, course, credit)
```

Key Consensus

22

- จะสังเกตว่าการเปลี่ยนค่า **Key** ของหนึ่งตารางมีผลต่อ **Key** ที่เชื่อมโยงกันในตารางอื่น ๆ ดังนั้นเราจึงนิยมใช้ **ID** ตัวเลขเป็น **Key** เพื่อให้การแก้ไขเป็นไปอย่างรวดเร็ว
- ข้อเสียคือคำสั่งอื่นบนตารางความสัมพันธ์จำเป็นต้องดึงข้อมูลจากตาราง **Entity** เสมอ

UPDATE Students

SET name = 'Jonathan Smith'

WHERE name = 'John Smith'

Students(sid, name, addr, tel)

Courses(cid, name, type)

Depts(did, name, addr, tel)

Enroll(sid, cid)

Major(sid, did)

Offer(did, cid, credit)

Key Consensus

23

- สมมติว่าใช้ ID เป็น **Key** แล้วเราต้องการทราบว่านักศึกษาที่ชื่อ **John Smith** เรียนวิชาใดบ้าง

```
SELECT Courses.name
FROM Students, Enroll, Courses
WHERE Students.sid = Enroll.sid
AND Courses.cid = Enroll.cid
AND Students.name = 'John Smith'
```

```
Students(sid, name, addr, tel)
Courses(cid, name, type)
Depts(did, name, addr, tel)
Enroll(sid, cid)
Major(sid, did)
Offer(did, cid, credit)
```

Key Consensus

24

- สำหรับคำสั่ง **INSERT** และ **DELETE** เราจำเป็นต้องรู้ค่า **ID** ของสิ่งที่ต้องการเพิ่มหรือลบข้อมูลในตารางความสัมพันธ์
- เพราะฉะนั้นเวลาเขียนโปรแกรมในภาษา **PHP** เราควรจะเก็บค่า **ID** ของผู้ใช้ปัจจุบันไว้เสมอ (เราจะได้เรียนการจดจำค่าในคาบต่อไป)

Students(sid, name, addr, tel)

Courses(cid, name, type)

Depts(did, name, addr, tel)

Enroll(sid, cid)

Major(sid, did)

Offer(did, cid, credit)

More Examples

25

- ถ้าอยากรอบว่าแต่ละวิชามีนักเรียนลงทะเบียนกี่คน

```
SELECT course, COUNT(student)
FROM Enroll
GROUP BY course
```

```
Students(name, addr, tel)
Courses(name, type)
Depts(name, addr, tel)
Enroll(student, course)
Major(student, dept)
Offer(dept, course, credit)
```

More Examples

26

- ถ้าอยากรทราบว่าแต่ละภาควิชามีนักเรียนลงทะเบียนเรียนวิชา IT2 กี่คน

```
SELECT Major.dept, COUNT(student)
```

```
FROM Enroll, Major
```

```
WHERE Enroll.course = 'IT2'
```

```
Enroll.student = Major.student
```

```
GROUP BY dept
```

```
Students(name, addr, tel)
```

```
Courses(name, type)
```

```
Depts(name, addr, tel)
```

```
Enroll(student, course)
```

```
Major(student, dept)
```

```
Offer(dept, course, credit)
```

References

27

- <http://www.w3schools.com/sql>