

Introduction to Database Part II

รวบรวมโดย

อาจารย์กิตติพิชญ์ คุปตะวานิช

ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยเชียงใหม่

MySQL – String Types

1. The CHAR and VARCHAR Types

- CHAR และ VARCHAR มีลักษณะคล้ายกัน
 - การ declare ระบุความยาวตัวอักษรมากที่สุดที่ต้องการเก็บ
 - VARCHAR จะแยก 1 – 2 byte แรกไว้เก็บความยาวจริง
 - สามารถระบุขนาดได้ถึง 65,535 ตัวอักษร

| Value | CHAR(4) | Storage Required | VARCHAR(4) | Storage Required |
|------------|---------|------------------|------------|------------------|
| " | ' ' | 4 bytes | " | 1 byte |
| 'ab' | 'ab ' | 4 bytes | 'ab' | 3 bytes |
| 'abcd' | 'abcd' | 4 bytes | 'abcd' | 5 bytes |
| 'abcdefgh' | 'abcd' | 4 bytes | 'abcd' | 5 bytes |

MySQL – String Types [2]

2. The BINARY and VARBINARY Types

- ใช้เก็บข้อมูล binary (0|1)
 - ระบุความยาวเป็น byte แทน

3. The BLOB and TEXT Types

- **BLOB** - binary large object
 - TINYBLOB
 - BLOB
 - MEDIUMBLOB
 - LONGBLOB

MySQL – String Types [3]

3. The BLOB and TEXT Types (cont.)

- **TEXT – string ขนาดใหญ่**
 - **TINYTEXT**
 - **TEXT**
 - **MEDIUMTEXT**
 - **LONGTEXT**
 - ดูขนาดจำนวน byte ที่ต้องใช้ของ data type ได้ที่
<http://dev.mysql.com/doc/refman/5.7/en/storage-requirements.html>

MySQL – String Types [4]

4. ENUM

- ใช้เก็บข้อมูล String ที่มีการระบุ list ของทุกค่าที่เป็นไปได้ไว้ก่อน โดยจะต้องเลือกเพียงค่าใดค่าหนึ่งจาก list
 - เช่น `ENUM('x-small', 'small', 'medium', 'large', 'x-large')`

5. SET

- ใช้เก็บข้อมูล String ที่มีการระบุ list ของทุกค่าที่เป็นไปได้ไว้ก่อน โดยเลือกกี่ค่าก็ได้จาก list
 - เช่น `SET('one', 'two')` สามารถมีค่าได้ดังนี้
 - ""
 - 'one'
 - 'two'
 - 'one,two'

3. ชนิดข้อมูล (2)

| <u>ข้อมูลที่เราต้องการ</u> | <u>ชื่อ field</u> | <u>ชนิดข้อมูล</u> |
|----------------------------|-------------------|-------------------|
| รหัสนักศึกษา | student_id | CHAR(9) |
| คำนำหน้านาม | title | VARCHAR(6) |
| ชื่อ | firstName | VARCHAR(20) |
| นามสกุล | lastName | VARCHAR(20) |
| ที่อยู่ | address | VARCHAR(100) |
| เกรดเฉลี่ย | GPA | DECIMAL(3,2) |
| เงินเดือน | allowance | MEDIUMINT |

การตั้งชื่อในฐานข้อมูล

- ใน Database เดียวกัน ห้ามมี Table ชื่อซ้ำกัน
- ใน Table เดียวกัน ห้ามมี ชื่อ field ซ้ำกัน
- และ ใน Table เดียวกัน ห้ามมี record ใด ๆ ที่เหมือนกันทุก field

4. การกำหนด Primary key

- พิจารณา record ข้อมูลในแต่ละ row ID
 - Title ชื่อ นามสกุล ที่อยู่ เบอร์โทรศัพท์บ้าน เบอร์มือถือ
 - อะไรที่ไม่ซ้ำ??
- กำหนด Primary key (คีย์หลัก) ให้กับ
 - field รหัสนักศึกษา
- เพื่อเป็นการกำหนด field ที่ใช้แยกแต่ละ record ออกจากกัน โดยเฉพาะเมื่อมีข้อมูลใน field อื่นซ้ำกัน

Student Table



| <u>STUDENT</u> | |
|-------------------|---------------------|
| student_id | CHAR(9) |
| title | VARCHAR(6) |
| firstName | VARCHAR(20) |
| lastName | VARCHAR(20) |
| address | VARCHAR(100) |
| GPA | DECIMAL(3,2) |
| allowance | MEDIUMINT |

5. Basic SQL Commands

- **SQL** ย่อมาจาก "**Structured Query Language**" เป็นภาษาที่ใช้ในการทำงานร่วมกับ **database**
- **Some of The Most Important SQL Commands**
 1. **SELECT** - extracts data from a database
 2. **UPDATE** - updates data in a database
 3. **DELETE** - deletes data from a database
 4. **INSERT INTO** - inserts new data into a database
 5. **CREATE DATABASE** - creates a new database

5. Basic SQL Commands [2]

7. **ALTER DATABASE** - modifies a database
 8. **CREATE TABLE** - creates a new table
 9. **ALTER TABLE** - modifies a table
 10. **DROP TABLE** - deletes a table
 11. **CREATE INDEX** - creates an index (search key)
 12. **DROP INDEX** - deletes an index
- ในภาษา **SQL** การใช้ตัวพิมพ์เล็กหรือตัวพิมพ์ใหญ่ ไม่มีความแตกต่าง (**case insensitive**)
 - ในกรการพิมพ์คำสั่งในภาษา **SQL** ควรลงท้ายด้วย ; (**semicolon**) ทุกครั้ง

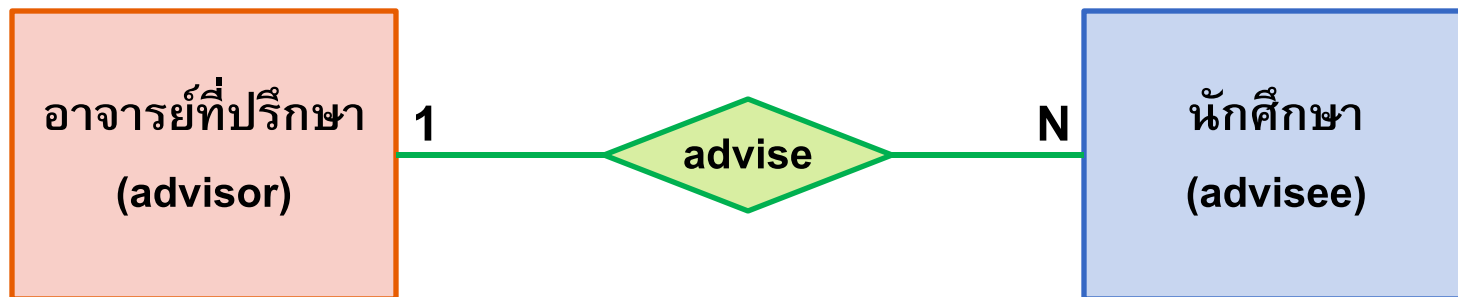
5. Basic SQL Commands [3]

- ตัวอย่าง การสร้างตารางในฐานข้อมูล

```
CREATE TABLE STUDENT
(
student_id char(9),
title varchar(6),
firstName varchar(20),
lastName varchar(20),
address varchar(100),
GPA decimal(3,2),
allowance mediumint,
PRIMART KEY (student_id)
);
```

Student – Advisor (revisited)

- อาจารย์ที่ปรึกษา
 - นักศึกษา 1 คน มี อาจารย์ที่ปรึกษาได้ 1 คน
 - อาจารย์ 1 คน เป็นที่ปรึกษาได้หลายคน

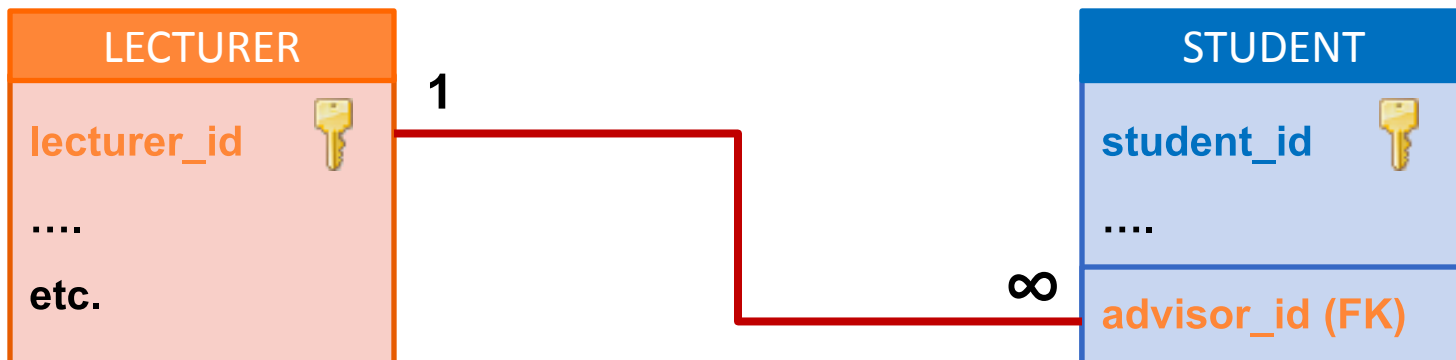


- 1:N or N:1?

6. การสร้างความสัมพันธ์ ด้วย Foreign Key

- ความสัมพันธ์เป็นแบบ 1:N
 - เพื่อสร้างความสัมพันธ์เราต้องเพิ่ม field รหัสอาจารย์ (คีย์นอก: FK) ที่ ตาราง Student เพื่อบอกว่านักศึกษาคนนี้มีอาจารย์ท่านนี้เป็นที่ปรึกษา

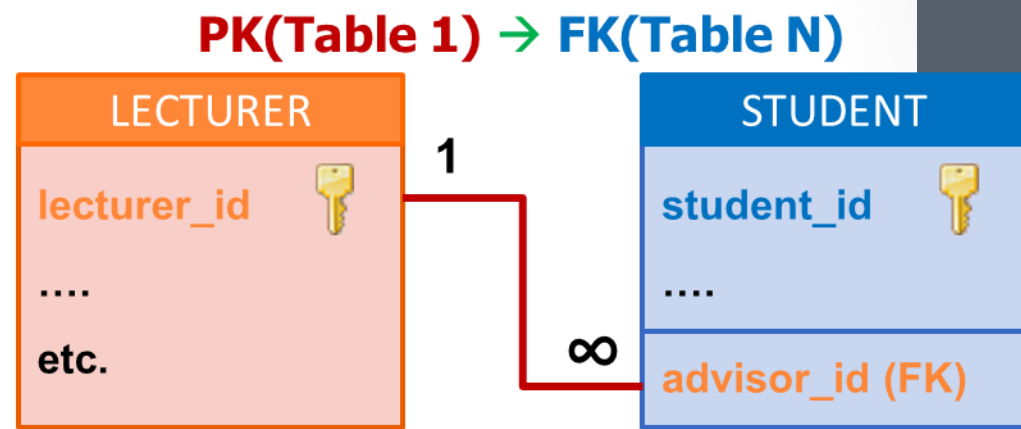
PK(Table 1) → FK(Table N)



6. การสร้างความสัมพันธ์ ด้วย Foreign Key [2]

- ตัวอย่าง

```
CREATE TABLE STUDENT  
(  
  student_id char(9),  
  title varchar(6),  
  firstName varchar(20),  
  lastName varchar(20),  
  address varchar(100),  
  GPA decimal(3,2),  
  allowance mediumint,  
  PRIMART KEY (student_id),  
  FOREIGN KEY (advisor_id) REFERENCES  
  LECTURER(lecturer_id)  
);
```



Summary

- ข้อมูลสารสนเทศ (information) คือข้อมูลที่ผ่านการประมวลผลแล้ว สามารถนำไปใช้ประโยชน์ได้
- การจัดการข้อมูลจำนวนมากอย่างมีประสิทธิภาพ โดยทั่วไป จำเป็นต้องใช้ฐานข้อมูล (Database)
- ในแต่ละฐานข้อมูลอาจประกอบด้วยตาราง (Table) หลายตารางที่มีความสัมพันธ์กัน
- การสร้างความสัมพันธ์ระหว่างตารางสามารถทำได้โดยการกำหนดคีย์นอก (Foreign Key)