

# Chapter 10

## PHP Database Management System (phpMyAdmin)

954242 Web Programming  
Sec.002 Semester 02/2555

Sakol Liamprawat  
sakolcnx@gmail.com

# วัตถุประสงค์

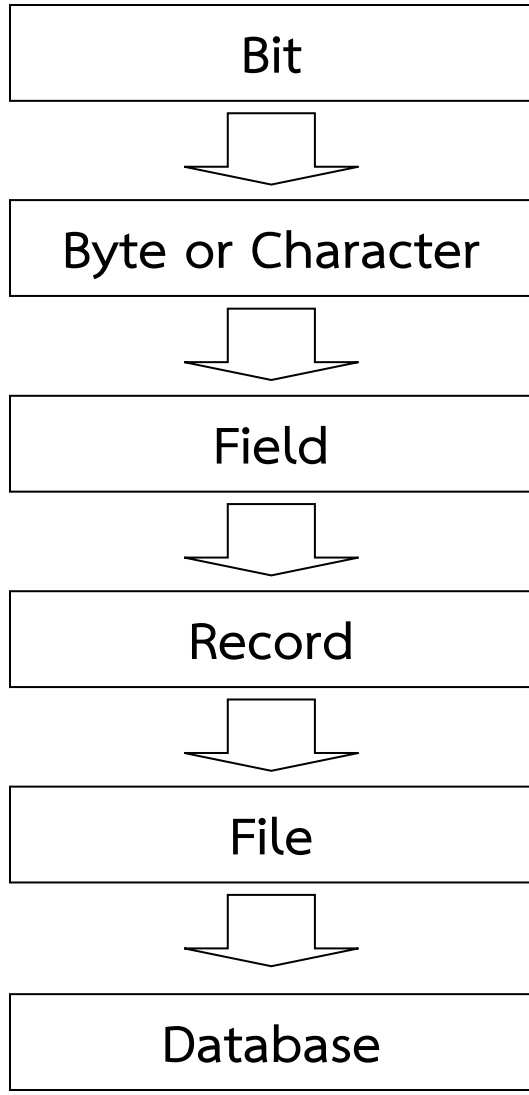
- สามารถอธิบายวิธีการจัดการฐานข้อมูลโดยใช้ phpMyAdmin ได้
- สามารถจัดการฐานข้อมูลโดยใช้ phpMyAdmin ได้

# Introduction to Database

- โครงสร้างของข้อมูล

- **บิต (Bit)** คือ หน่วยที่เล็กที่สุดของข้อมูลที่คอมพิวเตอร์สามารถจัดเก็บข้อมูลในหน่วยความจำ ประกอบด้วยเลขฐาน สอง ก็คือ 0 และ 1
- **ไบต์ (Byte) หรือ อักขระ (Characters)** คือ การนำกลุ่มของบิต 7 Bits หรือ 8 Bits มารวมกันเป็นตัวอักษร ตัวเลขหรือสัญลักษณ์พิเศษ หนึ่งตัวอักษร เช่น รหัส ASCII 1 ไบต์ ซึ่งจะเก็บข้อมูล 01000001 แทนตัวอักษร A
- **ฟิลด์ (Field)** คือ ข้อมูลที่ประกอบด้วยอักขระ เพื่อให้สามารถสื่อความหมายตามที่ใช้ต้องการ เช่น ชื่อนักศึกษา นามสกุลนักศึกษา เป็นต้น
- **ระเบียน (Record)** คือ การรวมกลุ่มของฟิลด์หลายๆ ฟิลด์ที่มีความหมายเกี่ยวพันกันมารวมกัน อย่างมีความหมาย
- **ไฟล์หรือแฟ้มข้อมูล (File)** คือ เป็นการรวมกันของระเบียนหลายๆ ระเบียนที่เกี่ยวข้องกันมารวมกัน เช่น แฟ้มข้อมูลนักศึกษา แฟ้มข้อมูลการขายสินค้า
- **ฐานข้อมูล (Database)** คือ กลุ่มของข้อมูลที่เป็นข้อเท็จจริง (Real Fact) ที่ถูกนำมารวมไว้ในที่เดียวกันอย่างเป็นระบบ เพื่อนำไปใช้ในวัตถุประสงค์อย่างใดอย่างหนึ่ง ข้อมูลนั้นอาจเป็นข้อความ รูปภาพ ซึ่งแต่ละไฟล์ที่นำมารวมกันจะมีความซ้ำซ้อนของข้อมูลน้อยที่สุด

# Introduction to Database



0 กับ 1

น  
นายเว็บ โปรแกรมมิ่ง

542110999	นายเว็บ โปรแกรมมิ่ง	954242	A
-----------	---------------------	--------	---

รหัสนักศึกษา	ชื่อ-สกุล	วิชา	เกรด
542110999	นายเว็บ โปรแกรมมิ่ง	954242	A
542111000	นายคอม โปรแกรมมิ่ง	954241	B+

ไฟล์ข้อมูลนักศึกษา



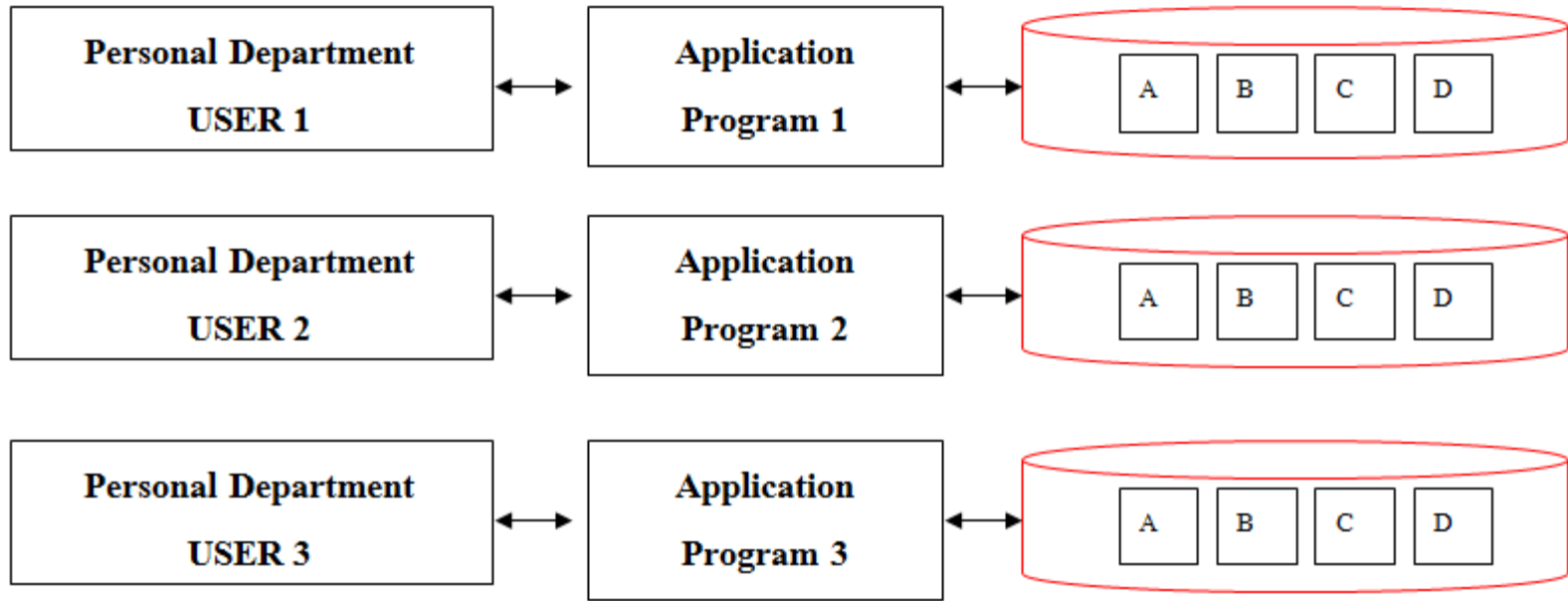
ไฟล์ข้อมูลนักศึกษา

ไฟล์ข้อมูลวิชา

ไฟล์ข้อมูลการลงทะเบียน

# Introduction to Database

- การจัดเก็บเพิ่มข้อมูลด้วยคอมพิวเตอร์ในยุคแรกๆ มีรูปแบบจัดเก็บคล้ายการจัดเก็บแฟ้มเอกสารด้วยมือ แต่เป็นการเก็บไว้ในคอมพิวเตอร์



การประมวลผลด้วยระบบเพิ่มข้อมูล

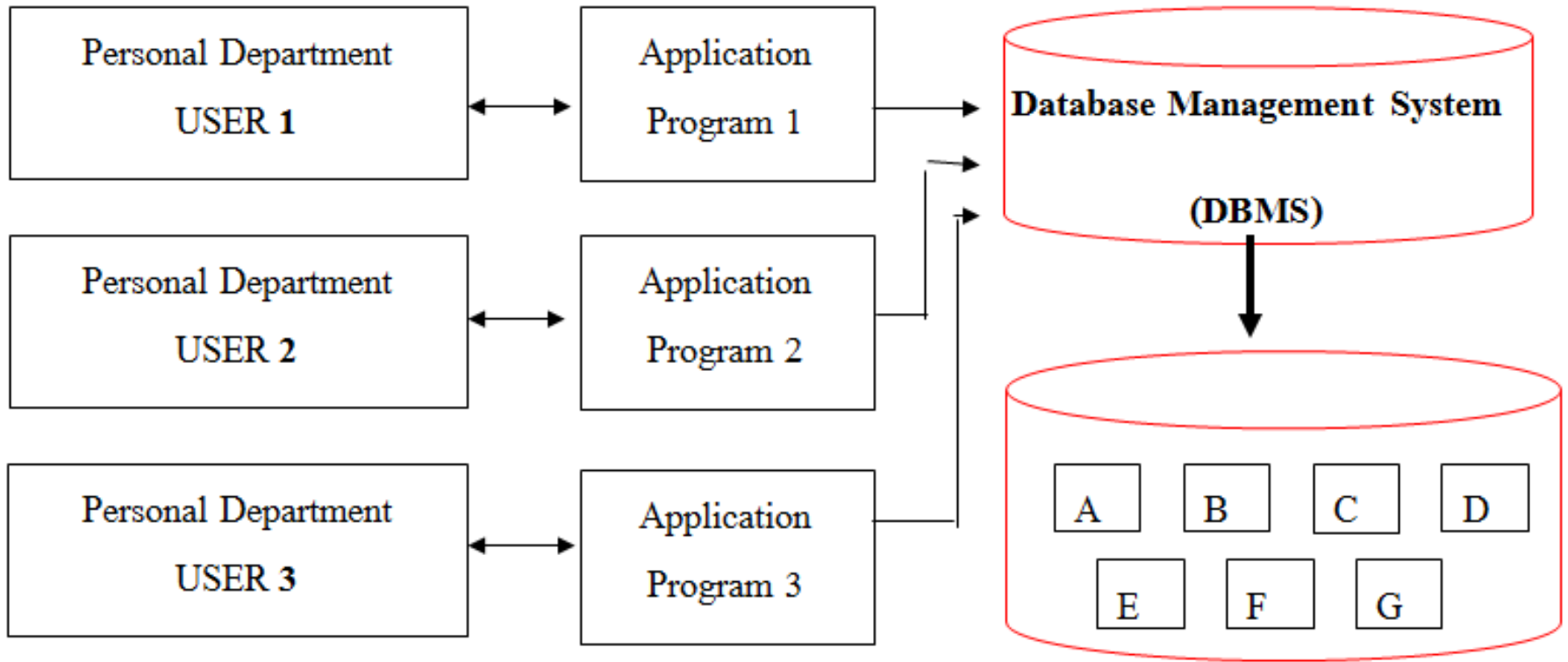
- แต่ละหน่วยงานมีการจัดเก็บข้อมูลเหมือนกันไว้ที่ต่างกัน
- เช่น A คือเพิ่มข้อมูลพนักงาน เวลาจะเรียกใช้งานเรียกใช้งานผ่านโปรแกรมประยุกต์ของตนเองที่สร้างขึ้นมา ทำให้ข้อมูลผ่านโปรแกรมตัวเอง
- ข้อมูลในองค์กรมีความซ้ำซ้อนกัน กระจัดกระจาย ยากต่อการแก้ไข บำรุงรักษา สับสนขัดแย้งกัน

ข้อดีระบบแฟ้มข้อมูล	ข้อจำกัดระบบแฟ้มข้อมูล
<ul style="list-style-type: none"><li>■ ง่ายต่อการออกแบบและพัฒนา ความสลับซับซ้อนในขั้นตอนการออกแบบและการพัฒนามีไม่มาก เพราะพัฒนาระบบเพียงหนึ่งแอปพลิเคชัน (Single Application)</li><li>■ การประมวลแบบแฟ้มข้อมูลเป็นวิธีดั้งเดิม มีความรวดเร็ว เพราะการประมวลผลข้อมูลสามารถกำหนดแฟ้มที่เกี่ยวข้องจากโปรแกรมประยุกต์ที่เขียนไว้โดยตรง ทำให้ประมวลผลข้อมูลได้อย่างรวดเร็ว</li></ul>	<ul style="list-style-type: none"><li>• ข้อมูลซ้ำซ้อน เช่น แฟ้มข้อมูล A B D E ถูกเก็บโดยแต่ละ User</li><li>• ยากต่อการปรับปรุงแก้ไข ถ้าต้องแก้ไขข้อมูลในแฟ้มข้อมูล A ของ User 1 ก็ต้องไปแก้ไขข้อมูลของ User 2 และ User 3 ด้วย</li><li>• การผูกติดกับข้อมูล การเขียนโปรแกรมในยุคเก่าต้องกำหนดโครงสร้างของแฟ้มข้อมูลในไว้โปรแกรม ยากต่อการปรับปรุงโครงสร้างของแฟ้มข้อมูล</li><li>• ข้อมูลกระจัดกระจาย เช่นแฟ้มข้อมูล A มีอยู่ในพื้นที่เก็บข้อมูลของผู้ใช้ทั้งสามคน</li></ul>

# Introduction to Database

- ประเภทของแฟ้มข้อมูล
  - แฟ้มหลัก (Master File) เป็นไฟล์ที่จัดเก็บข้อมูลที่มีรายการเปลี่ยนแปลง หรือมีสภาพค่อนข้างคงที่
  - แฟ้มรายการธุรกรรม (Transaction File) เป็นไฟล์ที่จัดเก็บข้อมูลการดำเนินธุรกรรมประจำวันที่มีแนวโน้มความเคลื่อนไหวอยู่เสมอ
  - แฟ้มอ้างอิง (Reference File) เป็นไฟล์หรือตารางที่ใช้สำหรับการอ้างอิง เพื่อใช้งานร่วมกันโดยข้อมูลต่าง ๆ ที่จัดเก็บลงในไฟล์นี้ค่อนข้างคงที่หรือมีค่าเปลี่ยนแปลงใด ๆ
- ฐานข้อมูล (Database) คือ กลุ่มของแฟ้มข้อมูลที่มีความสัมพันธ์กันและถูกนำมารวมกัน เช่น ฐานข้อมูลในมหาวิทยาลัยแห่งหนึ่ง ประกอบด้วยแฟ้มข้อมูลต่างๆ ซึ่งแต่ละแฟ้มต่างมีความสัมพันธ์กัน เช่น แฟ้มข้อมูลนักศึกษา แฟ้มข้อมูลกระบวนวิชา แฟ้มข้อมูลอาจารย์ แฟ้มข้อมูลการลงทะเบียน แฟ้มข้อมูลผลการเรียน เป็นต้น

# Introduction to Database



การประมวลผลด้วยระบบฐานข้อมูล

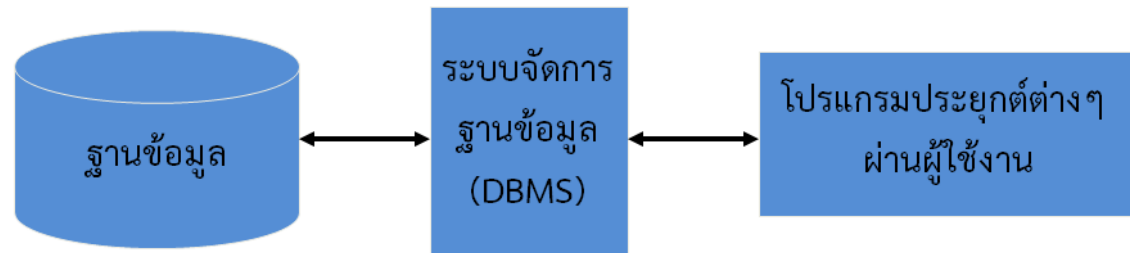
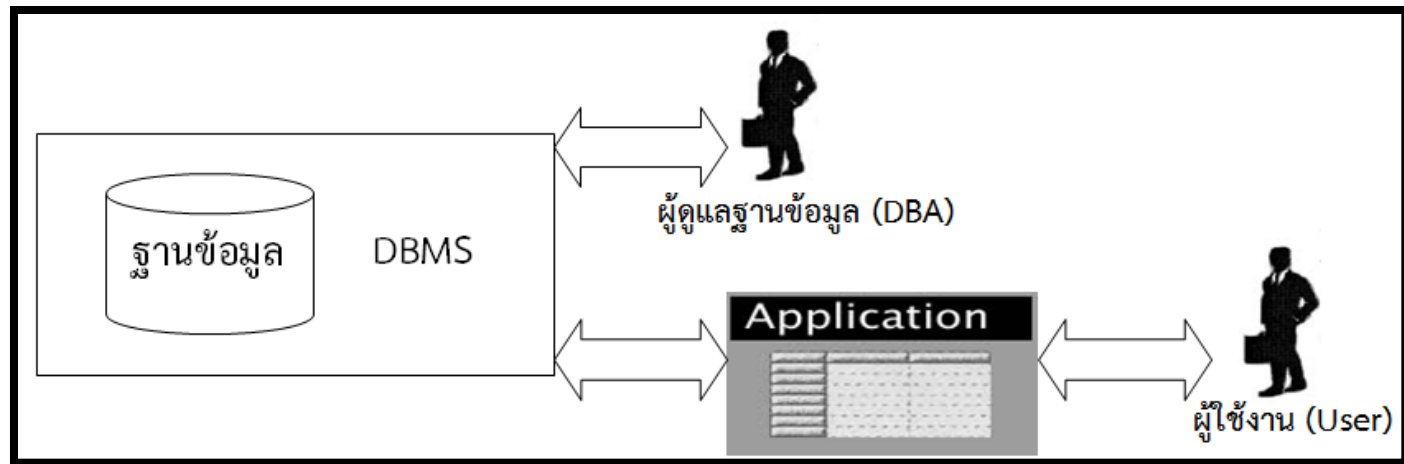


# Introduction to Database

- **ข้อดีของระบบฐานข้อมูลเปรียบเทียบกับระบบแฟ้มข้อมูล**
  - **ลดการเก็บข้อมูลซ้ำซ้อน** ข้อมูลในรูปแฟ้มข้อมูลที่มีอยู่หลายๆ แห่ง เพราะมีผู้ใช้ข้อมูลหลายคน เมื่อใช้ระบบฐานข้อมูลแล้วจะช่วยให้ความซ้ำซ้อนของข้อมูลลดลง
  - **รักษาความถูกต้องของข้อมูล** เพราะมีเพียงฐานข้อมูลเดียว เมื่อมีการแก้ไขข้อมูล ก็จะแก้ไขที่เดียว
  - **สามารถใช้ข้อมูลร่วมกันได้** ระบบฐานข้อมูลเป็นที่เก็บรวบรวมข้อมูลทุกอย่าง ผู้ใช้งานสามารถที่จะใช้ข้อมูลที่อยู่ในระบบได้
  - **ข้อมูลมีอิสระ** เมื่อผู้ใช้งานต้องการเปลี่ยนแปลงข้อมูลหรือนำข้อมูลมาประยุกต์ใช้ให้เหมาะสมกับโปรแกรมที่พัฒนาขึ้น สามารถสร้างข้อมูลนั้นขึ้นมาใช้ใหม่ได้ โดยไม่มีผลกระทบต่อระบบฐานข้อมูล
  - **สามารถขยายงานได้ง่าย** หากต้องการเพิ่มเติมข้อมูลที่เกี่ยวข้อง สามารถเพิ่มได้ง่าย เพราะข้อมูลมีอิสระ ไม่กระทบต่อข้อมูลเดิมที่มีอยู่

# Introduction to Database

- ระบบจัดการฐานข้อมูล (Database Management System : DBMS) คือ ซอฟต์แวร์ที่ใช้ในการจัดการฐานข้อมูล เปรียบเสมือนตัวกลางระหว่างผู้ใช้งานกับฐานข้อมูล ทำหน้าที่สร้าง เรียกใช้ข้อมูล หรือปรับปรุงฐานข้อมูล โดยการทำงานกับฐานข้อมูล จะต้องทำงานอ่านผ่าน DBMS ซึ่งผู้ใช้งานทำหน้าที่ออกคำสั่งให้กับ DBMS แล้ว DBMS จะทำหน้าที่ในการจัดการฐานข้อมูลตามคำสั่งนั้น

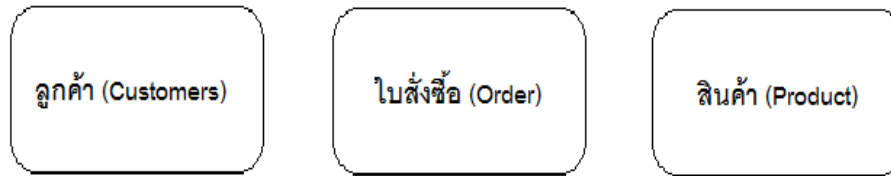


# Introduction to Database

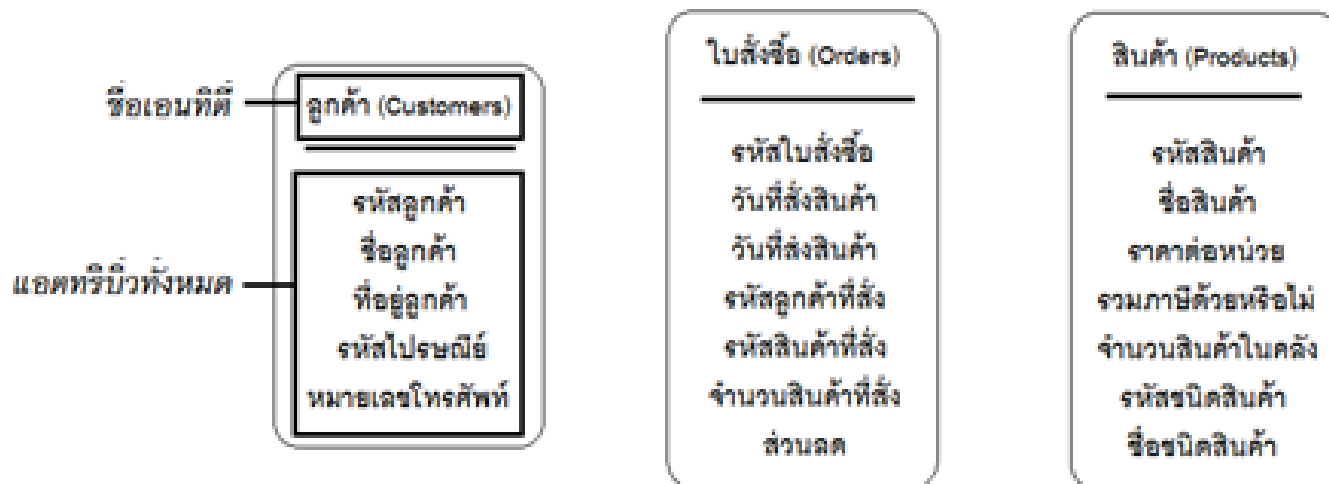
- ฐานข้อมูลเชิงสัมพันธ์ (Relational Database) คิดค้นพัฒนาโดย อี เอฟ คอดด์ (E.F.Codd) เป็นรูปแบบของฐานข้อมูลที่นิยมใช้กันมากในปัจจุบัน เนื่องจากการใช้งานง่าย ไม่ยุ่งยาก มีเครื่องมือช่วยในการจัดการข้อมูลอย่างเช่นคำสั่ง SQL ตัวอย่างของระบบเช่น DB2, Oracle, Ingres, dBase, Foxpro และ Access เป็นต้น
- ฐานข้อมูลเชิงสัมพันธ์ คือ กลุ่มของข้อมูลที่มีการจัดโครงสร้างในตารางที่สัมพันธ์กัน แต่ละตารางเป็นกลุ่มของแบบแผนของสิ่งที่สนใจ ได้แก่ กลุ่มลูกค้า ผลิตภัณฑ์ แต่ละตารางจะประกอบด้วยชุดของแถว (Row หรือ Record) และคอลัมน์ (Attribute หรือ Field) ที่แสดงถึงข้อมูลในตาราง แต่ละคอลัมน์แสดงถึงข้อมูลชนิดเดียวกัน เช่น ชื่อ ตำแหน่ง ที่อยู่ ส่วนแต่ละแถวจะประกอบด้วยชุดของข้อมูลที่สมบูรณ์หนึ่งรายการ ซึ่งก็คือ การรวบรวม Entity อยู่ระบบที่มีความสัมพันธ์ระหว่างกันเข้าไว้ด้วยกัน

# Introduction to Database

- Entity เป็นคำที่อ้างถึงบุคคล สถานที่ และสิ่งของต่างๆ เช่น สินค้า ใบสั่งซื้อ และลูกค้า เป็นต้น ถ้าเราสนใจในการสร้างระบบฐานข้อมูลการสั่งซื้อสินค้า Entity ของระบบนี้จะประกอบด้วย Entity ลูกค้า ใบสั่งซื้อสินค้า และสินค้า



- Attribute เป็นข้อมูลที่แสดงลักษณะของ Entity เช่น Attribute ของ Entityลูกค้า จะมีชื่อ ที่อยู่ และรหัสไปรษณีย์ ส่วน Attribute ของ Entity ใบสั่งซื้อสินค้า มีรหัสใบสั่งซื้อ วันที่สั่งซื้อ ชื่อสินค้า จำนวนสินค้าที่สั่ง และราคาสินค้า



# Introduction to Database

- ความสัมพันธ์ (Relationships) หมายถึง ความสัมพันธ์ระหว่าง Entity ต่างๆในระบบ เช่น ในระบบการสั่งซื้อสินค้า จะประกอบด้วย Entity ใบสั่งซื้อสินค้า และ Entity ลูกค้า ซึ่งมีความสัมพันธ์จากลูกค้าไปยังใบสั่งซื้อสินค้าเป็นแบบหนึ่งต่อกลุ่ม (One-to-Many) หมายความว่า ลูกค้าสามารถสั่งซื้อสินค้าได้หลายๆครั้ง คือ มีใบสั่งซื้อสินค้าหลายใบนั่นเอง แต่ใบสั่งซื้อแต่ละใบจะมาจากลูกค้าเพียงรายเดียวเท่านั้น เป็นต้น



# Introduction to Database

EMPLOYEE (พนักงาน)

EMPNUM	EMPNAME	HIREDATE	SALARY	POSITION	DEPNO	MGRNO
1001	SIRIWAN	06/13/93	9000	CLERK	10	1002
3001	ARLEE	08/15/93	17000	SALESMAN	30	3004
4001	WICHAI	12/26/93	33000	MANAGER	40	2002
1002	JINTANA	10/31/93	30000	CONTROLLER	10	1003
3002	MITREE	12/05/93	13000	SALESMAN	30	3004
3003	BENJAWAN	06/11/94	29000	MANAGER	30	2002
2001	CHAI	05/14/93	14000	CLERK	20	2003
1003	SURASIT	03/15/94	30000	MANAGER	10	2002
2002	KANJANA	07/10/94	50000	DIRECTOR	20	
3004	TANACHOTE	06/14/94	25000	SUPERVISOR	30	3003
1004	AMPORN	06/04/94	12000	CLERK	10	1002
3005	TAWATCHAI	07/03/94	10000	SALESMAN	30	3004
4002	THIDARAT	12/01/94	9000	CLERK	40	4001
2003	TERNJAI	11/01/94	24000	MANAGER	20	2002

EMPLOYEE Entity

มี Attribute "DEPNO"  
เป็นตัวเชื่อมความสัมพันธ์  
ระหว่างสอง Entity

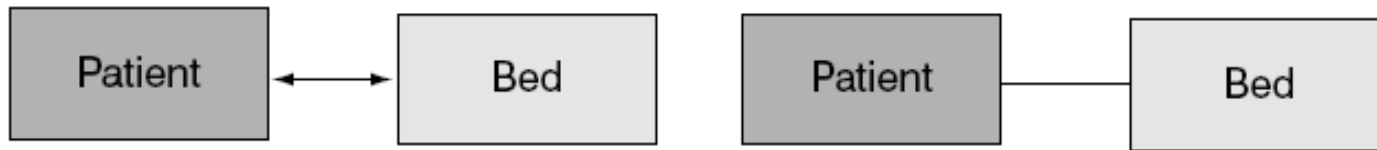
DEP (แผนก)

DEPNO	DEPNAME	LOCATION
10	ACCOUNTING	SILOM
20	ADMINISTRATION	SUKUMVIT
30	MARKETING	RATCHADA
40	FINANCE	SILOM
50	RESEARCH	SUKUMVIT

DEP Entity

# Introduction to Database

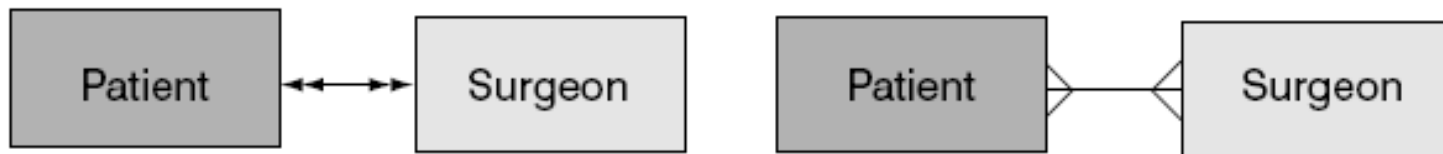
- **ความสัมพันธ์หนึ่งต่อหนึ่ง (One-to-One Relationships)** Record หนึ่งของตารางหนึ่ง จะมีความสัมพันธ์กับอีก Record ในอีกตารางหนึ่งเท่านั้น ไม่สามารถมีเกิน 1 ได้



- **ความสัมพันธ์แบบหนึ่งต่อกลุ่ม (One-to-Many Relationships)** เป็นความสัมพันธ์ที่ Record หนึ่งของตารางหนึ่ง จะสัมพันธ์กับสอง Record หรือมากกว่าในอีกตารางหนึ่ง



- **ความสัมพันธ์แบบกลุ่มต่อกลุ่ม (Many-to-Many Relationships)** สามารถสร้างความสัมพันธ์แบบนี้ได้ โดยสร้างตารางใหม่ที่มีความสัมพันธ์แบบ Many-to-One กับตาราง 2 ตารางที่มีความสัมพันธ์แบบ Many-to-Many relationships





# Introduction to Database

- **ชนิดของคีย์ในฐานะข้อมูลเชิงสัมพันธ์** ในระบบฐานข้อมูลเชิงสัมพันธ์ ต้องกำหนดชนิดของคีย์ เพื่อเป็น Attribute พิเศษที่ทำหน้าที่บางอย่าง เช่น เป็นตัวแทนของตาราง ซึ่งถูกกำหนดขึ้นมาเพื่อความสะดวกในการอ้างอิงถึงทฤษฎีเกี่ยวกับ Normalization
  - **Primary Key (คีย์หลัก)** จะเป็นฟิลด์ที่มีค่าไม่ซ้ำกันเลยในแต่ละ Record ในตารางนั้น สามารถใช้ฟิลด์ที่เป็น Primary Key นี้ เป็นตัวแทนของตารางนั้นได้ทันที
  - **Candidate Key (คีย์คู่แข่ง)** เป็นฟิลด์หนึ่ง หรือหลายฟิลด์ที่มีคุณสมบัติที่เป็น Primary Key ได้ แต่ไม่ได้เป็นคีย์หลัก เช่น ชื่อ และนามสกุล สามารถรวมกันเป็นคีย์คู่แข่งได้ เป็นต้น
  - **Composite Key (คีย์ผสม)** เป็นฟิลด์ที่ใช้รวมกับฟิลด์อื่นๆ ที่เป็น Composite Key เหมือนกัน มาใช้เป็น Primary Key ของตาราง
  - **คีย์นอก (Foreign Key)** เป็นฟิลด์ในตารางหนึ่ง (ฝั่ง Many) ที่มีความสัมพันธ์กับฟิลด์ที่เป็น Primary Key ในอีกตารางหนึ่ง (ฝั่ง One) โดยที่ตารางทั้งสองมีความสัมพันธ์แบบ One-to-Many ต่อกัน



# MySQL

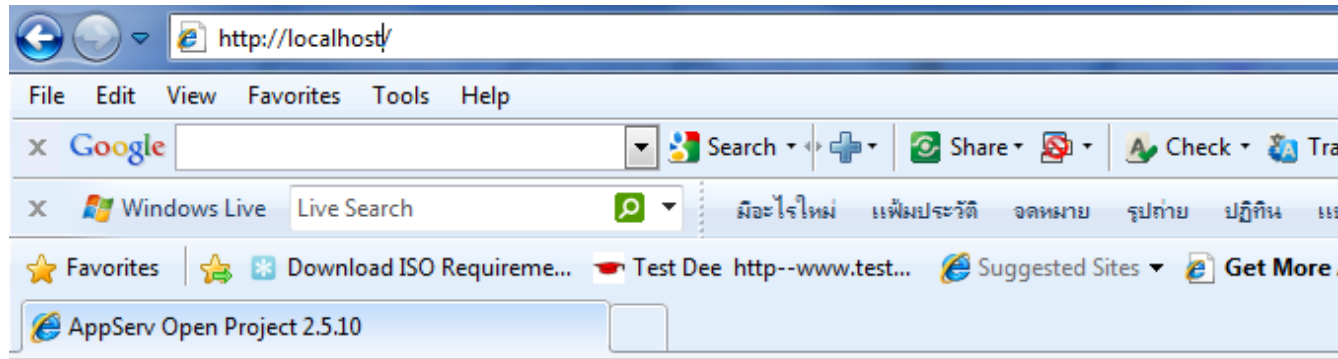
- MySQL เป็น Database Server ที่เหมาะกับองค์กรขนาดเล็กที่มีข้อมูลไม่มากนัก
- เป็นระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (Relational Database Management System; RDBMS)
- เป็น Freeware ด้านฐานข้อมูลที่ได้รับความนิยมอย่างมากในปัจจุบัน
- มีระบบการจัดการเข้าถึงข้อมูลจากผู้ใช้งานแต่ละคนได้การใช้ภาษาสอบถามเชิงโครงสร้างข้อมูล ที่เรียกว่าเอสคิวแอล (SQL: Structured Query Language)
- สนับสนุนการใช้งานระบบปฏิบัติการ ตัวอย่างเช่น Unix, Mac และ Windows
- ทำงานร่วมกับ Java, C, C++, PHP, ASP หรือ Perl

# phpMyAdmin

- เป็นสคริปต์ติดต่อฐานข้อมูลที่สร้างโดยภาษา php ซึ่งใช้จัดการฐานข้อมูล MySQL ผ่าน Web Browser เขียนขึ้นโดยนาย Tobias Retschiller และเป็นโปรแกรมแบบ Open Source ด้วย เพราะว่าเขียนด้วย PHP ทั้งหมด มีความสามารถดังนี้
  - สร้างและลบ Database
  - สร้างและจัดการ Table เช่น แทรก record, ลบ record, แก้ไข record หรือลบ Table, เพิ่มหรือแก้ไข field ในตาราง
  - โหลด Text file เข้าไปเก็บเป็นข้อมูลในตารางได้
  - หาผลสรุป (Query) ด้วยคำสั่ง SQL

# phpMyAdmin

- การเรียกใช้งาน ให้เรียกผ่าน Web Browser โดยพิมพ์ URL ไปยัง Directory ของ phpMyAdmin <http://localhost/phpMyAdmin>



## The AppServ Open Project - 2.5.10 สำหรับ วินโดวส์

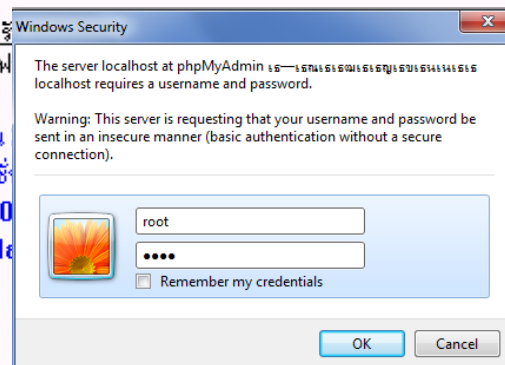
phpMyAdmin Database Manager เวอร์ชัน 2.10.3  
 PHP Information เวอร์ชัน 5.2.6

← หรือเรียก localhost แล้ว Click ที่นี่

เกี่ยวกับโปรแกรม AppServ เวอร์ชัน 2.5.10 สำหรับ  
 AppServ คือ โปรแกรมที่รวบรวมโอเพ่นซอร์สซอฟต์แวร์


- Apache Web Server เวอร์ชัน
- PHP Script Language เวอร์ชัน
- MySQL Database เวอร์ชัน 5.0
- phpMyAdmin Database M

- มีอะไรใหม่
- ไปรตอ่านคำแนะนำวิธีใช้งาน



ระบบจะเรียกหา username  
 กับ password ให้ใส่  
 เหมือนกับตอนติดตั้ง โดยปกติ  
 จะเป็น root , 1234

# phpMyAdmin



phpMyAdmin














[Home](#)
[Exit](#)
[SQL](#)
[Help](#)
[Feedback](#)

ฐานข้อมูล






(ฐานข้อมูล) ...

โปรดเลือกฐานข้อมูล

## localhost

- 
 รุ่นของเซิร์ฟเวอร์: 5.0.51b-community-nt-log
  - ▶ Protocol version: 10
- 
 เซิร์ฟเวอร์: localhost via TCP/IP
  - ▶ ผู้ใช้: root@localhost
- 
 ชุดตัวอักษร (charset) ของ MySQL: **UTF-8 Unicode (utf8)**
- 
 MySQL connection collation: utf8\_unicode\_ci
- 
 สร้างฐานข้อมูลใหม่
 
 การเรียงลำดับ
- 
 แสดงสถานะของ MySQL
- 
 แสดงตัวแปรระบบของ MySQL
- 
 โพรเซส
- 
 ชุดตัวอักษร และการเรียงลำดับ
- 
 Storage Engines
- 
 Reload privileges
- 
 สิทธิ์
- 
 ฐานข้อมูล
- 
 สงออก

## phpMyAdmin - 2.10.3

- ▶ MySQL client version: 5.0.51a
- ▶ Used PHP extensions: mysql
- 
 Language: ภาษาไทย - Thai
- 
 หน้าตา: Original
  - ▶ Font size: 100%
- 
[เอกสารการใช้ phpMyAdmin](#)
- 
[phpMyAdmin wiki](#)
- 
[โฮมเพจอย่างเป็นทางการของ phpMyAdmin](#)
  - ▶ [\[ChangeLog\]](#) [\[Subversion\]](#) [\[Lists\]](#)

# phpMyAdmin – Create database

The screenshot shows the phpMyAdmin interface for a localhost connection. The main content area is titled 'localhost' and displays server information such as 'รุ่นของเซิร์ฟเวอร์: 5.0.51b-community-nt-log' and 'ผู้ใช้: root@localhost'. Under the 'สร้างฐานข้อมูลใหม่' (Create new database) section, the database name 'testweb' is entered in the input field. A red arrow points to this field with the Thai text 'พิมพ์ชื่อฐานข้อมูล'. To the right of the input field is a dropdown menu set to 'การเรียงลำดับ' (Collation) and a blue 'สร้าง' (Create) button, which is also pointed to by a red arrow with the Thai text 'Click สร้าง'. The left sidebar shows the 'ฐานข้อมูล' (Databases) section with 'testoption (3)' selected and a list of databases: 'allcountry', 'allcustomer', and 'allproduct'.

# phpMyAdmin – Create database

เซิร์ฟเวอร์: localhost ▶ ฐานข้อมูล: testweb

โครงสร้าง SQL ค้นหา คำค้นจากตัวอย่าง ส่งออก Import กระบวนการ สิทธิ โยนทิ้ง

ฐานข้อมูล testweb สร้างเสร็จแล้ว

คำสั่ง SQL:  
`CREATE DATABASE `testweb` ;`

คำสั่ง SQL สร้างฐานข้อมูล

สร้างตารางในฐานข้อมูลนี้ testweb

ชื่อ:  Number of fields:

พิมพ์ชื่อตาราง และจำนวนคอลัมน์

รูปแบบคำสั่งภาษา SQL ในการสร้างฐานข้อมูล

`CREATE DATABASE 'ชื่อฐานข้อมูล' ;`

ชื่อ: table1 Number of fields: 3

ลงมือ

# phpMyAdmin – Create TABLE

กำหนดชนิดข้อมูล  
และความยาวให้

เลือกลักษณะ  
เฉพาะของข้อมูล

null=ไม่มีก็ได้  
not null - ต้องมี

Index

key

Unique

Primary

key

key

ชื่อฟิลด์

เหมาะสม

เซิร์ฟเวอร์: localhost ▶ ฐานข้อมูล: testweb ▶ ตาราง: table1

ฟิลด์	ชนิด	ความยาว/ เขต*	การเรียงลำดับ	แอตทริบิวต์	ว่างเปล่า (null)	ค่าปริยาย <sup>2</sup>	เพิ่มเติม				...	
id	TINYINT				not null		auto_increment	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>
fname	VARCHAR	20			not null			<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="checkbox"/>
lname	VARCHAR	30			not null			<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="checkbox"/>

หมายเหตุของตาราง:

Storage Engine:

InnoDB

การเรียงลำดับ:

ค่าเริ่มต้น  
อัตโนมัติ

Autonumber  
หรือไม่กำหนด

บันทึก หรือ Add 1 field(s) ลงมือ

# phpMyAdmin – Create TABLE










เซิร์ฟเวอร์: localhost ▶ ฐานข้อมูล: testweb ▶ ตาราง : table1







เปิดดู โครงสร้าง SQL ค้นหา แทรก ส่งออก Import กระบวนการ ลบข้อมูล โยนทิ้ง

**i** ตาราง `testweb`.`table1` สร้างเสร็จแล้ว

คำสั่ง SQL:

```
CREATE TABLE `table1` (
  `id` TINYINT NOT NULL AUTO_INCREMENT PRIMARY KEY,
  `fname` VARCHAR( 20 ) NOT NULL,
  `lname` VARCHAR( 30 ) NOT NULL
) ENGINE = INNODB;
```

	ฟิลด์	ชนิด	การเรียงลำดับ	แอตทริบิวต์	ว่างเปล่า (null)	ค่าปริยาย	เพิ่มเติม	กระทำการ
<input type="checkbox"/>	id	tinyint(4)			ไม่		auto_increment	      
<input type="checkbox"/>	fname	varchar(20)	utf8_general_ci		ไม่			      
<input type="checkbox"/>	lname	varchar(30)	utf8_general_ci		ไม่			      

เลือกทั้งหมด / ไม่เลือกเลย ทำกับที่เลือก:      

แสดง เสนอโครงสร้างตาราง ?

Add 1 field(s) ที่จุดสุดท้ายของตาราง  ที่จุดเริ่มต้นของตาราง  หลัง id



# phpMyAdmin – Create TABLE

## รูปแบบคำสั่งภาษา SQL ในการสร้างตาราง

**CREATE TABLE** ชื่อตาราง

(

ชื่อฟิลด์ที่ 1      ชนิดของฟิลด์ (ขนาดของฟิลด์),

ชื่อฟิลด์ที่ 2      ชนิดของฟิลด์ (ขนาดของฟิลด์),

... ,

ชื่อฟิลด์ที่ n      ชนิดของฟิลด์ (ขนาดของฟิลด์),

**PRIMARY KEY** ('ชื่อฟิลด์')

);

# phpMyAdmin – INSERT field into TABLE

แสดง เสนอโครงสร้างตาราง

Add 1 field(s)  ที่จุดสุดท้ายของตาราง  ที่จุดเริ่มต้นของตาราง  หลัง id

ดัชนี: ?					เนื้อที่ที่ใช้			สถิติของแถว	
ชื่อคีย์	ชนิด	Cardinality	กระทำการ	ฟิลด์	ชนิด	ใช้งาน		คำสั่ง	ค่า
PRIMARY	PRIMARY	0		id	ข้อมูล	0 ไบต์			
สร้างดัชนีโดยคอลัมน์ 1 <input type="button" value="ลงมือ"/>					ดัชนี	0 ไบต์			
					รวม	0 ไบต์			

เซิร์ฟเวอร์: localhost ▶ ฐานข้อมูล: testweb ▶ ตาราง : table1 "InnoDB free: 3072 kB"

ฟิลด์	ชนิด	ความยาว/เซด*	การเรียงลำดับ	แอตทริบิวต์	ว่างเปล่า (null)	ค่าปริยาย²	เพิ่มเติม					
nickname	VARCHAR	10			not null							

หรือ Add 1 field(s)

# phpMyAdmin – INSERT field into TABLE

เซิร์ฟเวอร์: localhost ▶ ฐานข้อมูล: testweb ▶ ตาราง : table1 "InnoDB free: 3072 kB"

เปิดดู โครงสร้าง SQL ค้นหา แทรก ส่งออก Import กระบวนการ ลบข้อมูล โยนทิ้ง

**i** ตาราง table1 เปลี่ยนเสร็จแล้ว

คำสั่ง SQL:

```
ALTER TABLE `table1` ADD `nickname` VARCHAR( 10 ) NOT NULL ;
```

	ฟิลด์	ชนิด	การเรียงลำดับ	แอตทริบิวต์	ว่างเปล่า (null)	ค่าปริยาย	เพิ่มเติม	กระทำการ						
<input type="checkbox"/>	id	tinyint(4)			ไม่		auto_increment							
<input type="checkbox"/>	fname	varchar(20)	utf8_general_ci		ไม่									
<input checked="" type="checkbox"/>	<del>lname</del>	<del>varchar(30)</del>	<del>utf8_general_ci</del>	<del></del>	<del>ไม่</del>	<del></del>	<del></del>							
<input type="checkbox"/>	nickname	varchar(10)	utf8_general_ci		ไม่									

เลือกทั้งหมด / ไม่เลือกเลย ทำกับที่เลือก:

# phpMyAdmin – INSERT field into TABLE

## รูปแบบคำสั่งภาษา SQL ในการแทรกฟิลด์ใหม่ในตาราง

### 1. การแทรกฟิลด์ท้ายของตาราง

**ALTER TABLE** 'ชื่อตาราง' **ADD** 'ชื่อฟิลด์ที่ต้องการแทรกเพิ่มเติม' ชนิดของฟิลด์(ขนาดของฟิลด์),  
การกำหนดคุณลักษณะของฟิลด์ ;

### 2. การแทรกฟิลด์ที่จุดเริ่มต้นของตาราง

**ALTER TABLE** 'ชื่อตาราง' **ADD** 'ชื่อฟิลด์ที่ต้องการแทรกเพิ่มเติม' ชนิดของฟิลด์(ขนาดของฟิลด์),  
การกำหนดคุณลักษณะของฟิลด์ **FIRST** ;

### 3. การแทรกฟิลด์หลังฟิลด์ใดฟิลด์หนึ่ง

**ALTER TABLE** 'ชื่อตาราง' **ADD** 'ชื่อฟิลด์ที่ต้องการแทรกเพิ่มเติม' ชนิดของฟิลด์(ขนาดของฟิลด์),  
การกำหนดคุณลักษณะของฟิลด์ **AFTER** 'ชื่อฟิลด์ที่อยู่ข้างหน้า' ;

# phpMyAdmin – Insert data into TABLE

เซิร์ฟเวอร์: localhost ▶ ฐานข้อมูล: testweb ▶ ตาราง : table1 "InnoDB free: 3072 kB"

เปิด โครงสร้าง SQL ค้นหา **แทรก** ส่งออก Import กระบวนการ ลบข้อมูล โยนทิ้ง



**i** ตาราง table1 เปลี่ยนเสร็จแล้ว




คำสั่ง SQL:










```
ALTER TABLE `table1` ADD `nickname` VARCHAR( 10 ) NOT NULL ;
```

	ฟิลด์	ชนิด	การเรียงลำดับ	แอตทริบิวต์	ว่างเปล่า (null)	ค่าปริยาย	เพิ่มเติม	กระทำการ
<input type="checkbox"/>	id	tinyint(4)			ไม่		auto_increment	
<input type="checkbox"/>	fname	varchar(20)	utf8_general_ci		ไม่			
<input type="checkbox"/>	lname	varchar(30)	utf8_general_ci		ไม่			
<input type="checkbox"/>	nickname	varchar(10)	utf8_general_ci		ไม่			

เลือกทั้งหมด / ไม่เลือกเลย ทำกับที่เลือก:

# phpMyAdmin – Insert data into TABLE

 เซิร์ฟเวอร์: localhost ▶ 
  ฐานข้อมูล: testweb ▶ 
  ตาราง : table1 "InnoDB free: 3072 kB



 เปิดดู 
  โครงสร้าง 
  SQL 
  ค้นหา 
  แทรก 
  ส่งออก 
  Import 
  กระบวนการ 
  ล้าง

ฟิลด์	ชนิด	ฟังก์ชัน	ว่างเปล่า (null)	ค่า
id	tinyint(4)	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
fname	varchar(20)	<input type="text"/>	<input type="checkbox"/>	wayne
lname	varchar(30)	<input type="text"/>	<input type="checkbox"/>	rooney
nickname	varchar(10)	<input type="text"/>	<input type="checkbox"/>	roon

ไม่สนใจ

ฟิลด์	ชนิด	ฟังก์ชัน	ว่างเปล่า (null)	ค่า
id	tinyint(4)	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
fname	varchar(20)	<input type="text"/>	<input type="checkbox"/>	michael
lname	varchar(30)	<input type="text"/>	<input type="checkbox"/>	owen
nickname	varchar(10)	<input type="text"/>	<input type="checkbox"/>	babygoal

แทรกเป็นแถวใหม่ ▼ and then  ▼

# phpMyAdmin – Insert data into TABLE

เซิร์ฟเวอร์: localhost ▶ ฐานข้อมูล: testweb ▶ ตาราง: table1 "InnoDB free: 3072 kB"

เปิด โครงสร้าง SQL ค้นหา แทรก ส่งออก Import กระบวนการ ลบข้อมูล โยนทิ้ง

แถวที่ถูกแทรก: 2  
แทรกเลขประจำแถว: 2

คำสั่ง SQL:

```
INSERT INTO `testweb`.`table1` (
  `id`,
  `fname`,
  `lname`,
  `nickname`
)
VALUES (
  NULL, 'wayne', 'rooney', 'roon'
), (
  NULL, 'michael', 'owen', 'babygoal'
);
```

คำสั่ง SQL:

```
SELECT *
FROM `table1`
LIMIT 0, 30
```

Query results operations

แสดง Print view (with full texts) ส่งออก

แสดง: 30 แถว เริ่มจากแถวที่ 0

อยู่ใน  และข้ามหัวแถวทุกๆ  เซลล์

เรียงโดยคีย์: ไม่มี

		id	fname	lname	nickname
<input type="checkbox"/>		1	wayne	rooney	roon
<input type="checkbox"/>		2	michael	owen	babygoal

เลือกทั้งหมด ไม่เลือกเลข ลำดับที่เลือก

แสดง: 30 แถว เริ่มจากแถวที่ 0

# phpMyAdmin – Insert data into TABLE

รูปแบบคำสั่งภาษา SQL ในการเพิ่มข้อมูลในตาราง

```
INSERT INTO ชื่อตาราง ('ชื่อฟิลด์ที่ 1', 'ชื่อฟิลด์ที่ 2', ..., 'ชื่อฟิลด์ที่ n',)  
VALUES ('ค่าที่ 1', 'ค่าที่ 2', ..., 'ค่าที่ n',) ;
```





# phpMyAdmin – Edit data in TABLE

แสดง: 30 แถว เริ่มจากแถวที่ 0

อยู่ใน  และข้ามหัวแถวทุกๆ 100 เซลล์

เรียงโดยคีย์:  ลงมือ

	id	fname	lname	nickname
<input type="checkbox"/>	1	wayne	rooney	roon
<input type="checkbox"/>	2	michael	owen	babygoal

เลือกทั้งหมด / ไม่เลือกเลย ทำกับที่เลือก:   

แสดง: 30 แถว เริ่มจากแถวที่ 0


Click



เพื่อแก้ไข

ฟิลด์	ชนิด	ฟังก์ชัน	ว่างเปล่า (null)	ค่า
id	tinyint(4)	<input type="text"/>	<input type="checkbox"/>	1
fname	varchar(20)	<input type="text"/>	<input type="checkbox"/>	Darren
lname	varchar(30)	<input type="text"/>	<input type="checkbox"/>	Fletcher
nickname	varchar(10)	<input type="text"/>	<input type="checkbox"/>	Fletch

บันทึก  and then



# phpMyAdmin – Edit data in TABLE

คำสั่ง SQL:

```
UPDATE `testweb`.`table1` SET `fname` = 'Darren',
`lname` = 'Fletcher',
`nickname` = 'Fletch' WHERE `table1`.`id` = 1 LIMIT 1 ;
```

แสดง: 30 แถว เริ่มจากแถวที่ 0  
 อยู่ใน  และข้ามหัวแถวทุกๆ 100 เซลล์  
 เรียงโดยคีย์:

	←T→	id	fname	lname	nickname	
<input type="checkbox"/>			1	Darren	Fletcher	Fletch
<input type="checkbox"/>			2	michael	owen	babygoal

เลือกทั้งหมด / ไม่เลือกเลย ทำกับที่เลือก:

แสดง: 30 แถว เริ่มจากแถวที่ 0  
 อยู่ใน  และข้ามหัวแถวทุกๆ 100 เซลล์

# phpMyAdmin – Edit data in TABLE

รูปแบบคำสั่งภาษา SQL ในการแก้ไขข้อมูลในตาราง

**UPDATE** 'ชื่อตาราง'

**SET** 'ชื่อฟิลด์ที่ 1' = 'ค่าใหม่ของฟิลด์ที่ 1' ,

'ชื่อฟิลด์ที่ 2' = 'ค่าใหม่ของฟิลด์ที่ 2' ,

... ,

'ชื่อฟิลด์ที่ n' = 'ค่าใหม่ของฟิลด์ที่ n'

**WHERE** 'ชื่อฟิลด์ที่เป็นคีย์หลัก' = 'ค่าของคีย์หลักของ Record ที่ต้องการแก้ไข' ;

# phpMyAdmin – Delete record from TABLE

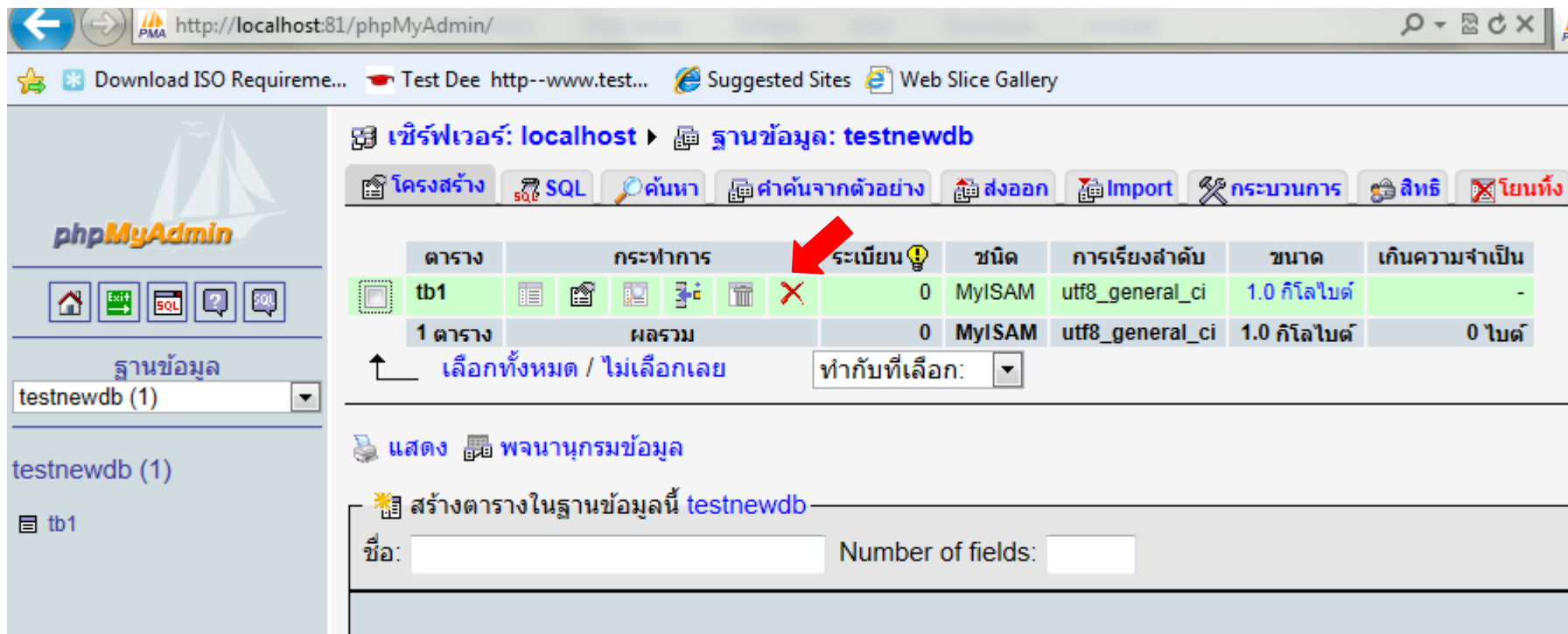


รูปแบบคำสั่งภาษา SQL ในการลบรายการข้อมูลในตาราง

**DELETE FROM** 'ชื่อตาราง'

**WHERE** 'ชื่อฟิลด์ที่เป็นคีย์หลัก' = 'ค่าของคีย์หลักของ Record ที่ต้องการลบ' ;

# phpMyAdmin – Delete TABLE



The screenshot shows the phpMyAdmin interface for a database named 'testnewdb'. The table 'tb1' is selected. The interface includes a navigation menu on the left, a top toolbar with various actions, and a main table view. A red arrow points to the delete icon (X) in the 'การกระทำ' (Actions) column for the table 'tb1'.

ตาราง	การกระทำ	ระเบียน	ชนิด	การเรียงลำดับ	ขนาด	เกินความจำเป็น
tb1		0	MyISAM	utf8_general_ci	1.0 กิโลไบต์	-
1 ตาราง	ผลรวม	0	MyISAM	utf8_general_ci	1.0 กิโลไบต์	0 ไบต์

เลือกทั้งหมด / ไม่เลือกเลย      ทำกับที่เลือก:

สร้างตารางในฐานข้อมูลนี้ testnewdb

ชื่อ:       Number of fields:

รูปแบบคำสั่งภาษา SQL ในการลบตาราง

**DROP TABLE** 'ชื่อตาราง' ;

# phpMyAdmin – Delete DATABASE

The screenshot shows the phpMyAdmin interface in a browser window. The address bar shows `http://localhost:81/phpMyAdmin/`. The main content area displays the database 'testnewdb' with a table 'tb1'. A red arrow points to the 'ลบทิ้ง' (Delete) button in the top toolbar. A confirmation dialog box titled 'Message from webpage' is overlaid on the screen, containing the following text:

You are about to DESTROY a complete database!  
 ต้องการจะ :  
 DROP DATABASE `testnewdb`

The dialog has 'OK' and 'Cancel' buttons. A red arrow points to the 'OK' button.

รูปแบบคำสั่งภาษา SQL ในการลบฐานข้อมูล

**DROP DATABASE** 'ชื่อฐานข้อมูล'

# phpMyAdmin – Data Type

ชนิดของข้อมูลในคอลัมน์มี 3 ประเภท โดยมีขอบเขตการเก็บข้อมูล และขนาดในการใช้พื้นที่ในหน่วยความจำแตกต่างกัน

- **ชนิดตัวเลข (Numeric Type)**

- ประกอบด้วยเลขจำนวนเต็มและเลขจำนวนจริง

- **ชนิดวันและเวลา (Date and Time Type)**

- กำหนดเป็นวัน หรือ/เวลาได้

- **ชนิดข้อความ (String Type)**

- ข้อความขนาดสั้น ได้แก่ CHAR และ VARCHAR
- ข้อความขนาดยาว ได้แก่ TEXT และ BLOB (Binary Large Objects)
- ข้อความขนาดพิเศษ ได้แก่ SET และ ENUM

# phpMyAdmin – Numeric Type

## เลขจำนวนเต็ม

Type	ช่วงข้อมูล	เนื้อที่เก็บ (Byte)	ความหมาย
TINYINT	0...255 หรือ -128...127	1	เก็บ interger ขนาดเล็กมาก
SMALLINT	0...65535 หรือ -32768...32767	2	เก็บ interger ขนาดเล็ก
MEDIUMINT	0...16777215 หรือ -8388608...8388607	3	เก็บ interger ขนาดกลาง
INT	0..4294967295 หรือ -2147483648...2147483647	4	เก็บ interger ขนาดปกติ
BIGINT	0...18446744073709551615 หรือ -9223372036854775808...-922337203685477587	8	เก็บ interger ขนาดใหญ่



# phpMyAdmin – Numeric Type

- ข้อมูลประเภทตัวเลขสามารถกำหนดส่วนของ “แอดทริบิวต์” เพิ่มเป็น UNSIGNED หรือ UNSIGNED ZEROFILL ได้
  - UNSIGNED : จะหมายถึงเก็บค่าตัวเลขแบบไม่มีเครื่องหมาย แบบนี้จะทำให้สามารถเก็บค่าได้ตั้งแต่ 0 - 255
  - UNSIGNED ZEROFILL : เหมือน UNSIGNED แต่ถ้าข้อมูลที่กรอกเข้ามาไม่ครบตามจำนวนหลักที่กำหนด MySQL จะเติม 0 ให้ครบหลักเอง เช่น กำหนดให้ใส่ 3 หลัก แล้วเก็บข้อมูล 12 ลงไป เวลาที่สืบค้นดู จะได้ค่าออกมาเป็น 012
- ถ้าไม่กำหนด "แอดทริบิวต์" จะได้ SIGNED คือต้องเสีย 1 บิตเพื่อเก็บเครื่องหมาย บวก/ลบ ทำให้สามารถเก็บข้อมูลได้อยู่ในช่วง -128 ถึง 127 เท่านั้น

# phpMyAdmin – Numeric Type

## เลขจำนวนจริง

Type	ช่วงข้อมูล	เนื้อที่เก็บ (Byte)	ความหมาย
FLOAT	$\pm 1.175494351\text{E-}38 \dots \pm 3.402823466\text{E+}38$	4	เก็บเลขจำนวนจริงขนาดปกติ
DOUBLE	$\pm 2.2250738585072014\text{E-}308 \dots$ $\pm 1.7976931348623157\text{E+}308$	8	เก็บเลขจำนวนจริงขนาดใหญ่

# phpMyAdmin – Date and Time Type

## วันและเวลา

Type	ช่วงข้อมูล	เนื้อที่เก็บ (Byte)	ความหมาย
DATE	1000-01-10 หรือ 9999-12-31	3	เก็บข้อมูลเป็น คศ.-เดือน-วัน (YYYY-MM-DD)
TIME	-838:56:59...838:56:59	5	เก็บข้อมูลเป็น ชั่วโมง:นาที:วินาที (HH-MM-SS)
DATETIME	1000-01-10 00:00:00 หรือ 9999-12-31 23:59:59	8	เก็บข้อมูลเป็น คศ.-เดือน-วัน ชั่วโมง:นาที:วินาที (YYYY-MM-DD HH-MM-SS)

# phpMyAdmin – Date and Time Type

## วันและเวลา

Type	ช่วงข้อมูล	เนื้อที่เก็บ (Byte)	ความหมาย
TIMESTAMP[M]	19700101000000... 20371231235959	4	เก็บข้อมูลเป็น คศ.-เดือน-วัน ชั่วโมง: นาที:วินาที รูปแบบขึ้นกับค่า M
YEAR[(2 4)]	70-69 หรือ 1970-2069	1	เก็บข้อมูลปี (ระบุเป็น 2 จะได้ 70-69 ระบุ 4 จะได้ 1970-2069)

# phpMyAdmin – Date and Time Type

## การแสดงผลของ TIMESTAMP

ชนิดที่กำหนด	การแสดงผล
TIMESTAMP	YYYYMMDDHHMMSS
TIMESTAMP(14)	YYYYMMDDHHMMSS
TIMESTAMP(12)	YYMMDDHHMMSS
TIMESTAMP(10)	YYMMDDHHMM
TIMESTAMP(8)	YYYYMMDD
TIMESTAMP(6)	YYMMDD
TIMESTAMP(4)	YYMM
TIMESTAMP(2)	YY

# phpMyAdmin – String Type

## ● ข้อความขนาดสั้น

- CHAR ระบุจำนวนตัวอักษรที่สามารถเก็บได้ (Fixed length character) ถ้ามีข้อมูลไม่ถึงจำนวนระบุ จะใส่ช่องว่างในส่วนที่เหลือ การประมวลผลจะเร็วกว่า
- VARCHAR ไม่ระบุจำนวนตัวอักษรที่สามารถเก็บได้ (Variable length character) ใช้เนื้อที่เท่ากับจำนวนตัวอักษรที่เก็บ การประมวลผลจะช้ากว่า

Type	ความยาวสูงสุด	ความหมาย
CHAR	1	สามารถเก็บจำนวนตัวอักษรได้เท่ากับ 1 ตัวอักษร
CHAR(n)	255	สามารถเก็บจำนวนตัวอักษรได้เท่ากับ n ตัวอักษร โดยมีความยาวคงที่ (n)
VARCHAR	65535	สามารถเก็บจำนวนตัวอักษร โดยมีความยาวไม่คงที่

# phpMyAdmin – String Type

- ข้อความขนาดสั้น
  - CHAR และ VARCHAR โดยปกติแล้วการจัดเรียงข้อมูลเวลาสืบค้น (query) เป็นแบบ case sensitive
  - แต่สามารถเลือก "แอดทริบิวต์" เป็น BINARY ได้ซึ่งเวลาสืบค้น (query) จะไม่สนใจว่าตัวอักษรเป็นตัวใหญ่หรือตัวเล็ก
  - TINYTEXT ต่างจาก CHAR กับ VARCHAR ตรงที่สามารถทำ Full Text Search ได้

# phpMyAdmin – String Type

## ● ข้อความขนาดยาว

- TEXT เก็บข้อความยาวๆ คำนี้ถึงเรื่อง **Case Sensitive**
- BLOB (Binary Large Object) ทำงานเหมือนกับ TEXT เก็บข้อมูลประเภท Binary เช่น ไฟล์ข้อมูลต่างๆ ไฟล์รูปภาพ ไฟล์มัลติมีเดีย

## ● ข้อความแบบพิเศษ

- SET เป็นการกำหนดให้ค่าในคอลัมน์นั้น ต้องมาจากกลุ่มของค่าที่ระบุเท่านั้น ค่าที่อยู่ในคอลัมน์จะ**มีมากกว่า 1 ค่าได้** แต่กลุ่มของค่าที่จะระบุสามารถมีได้ถึง 64 ค่า
- ENUM คล้ายกับ SET แต่ค่าที่อยู่ในคอลัมน์**ต้องมีค่าเดียว** กลุ่มของค่าที่ระบุมีได้ถึง 65,535 ค่า



# phpMyAdmin – String Type

## ชนิดข้อความแบบ TEXT และ BLOB

ชนิด	ความยาวสูงสุด	ความหมาย
TINYBLOB	255	แบบ BLOB ขนาดเล็ก
TINYTEXT	255	แบบ TEXT ขนาดเล็ก
BLOB	65,535	แบบ BLOB ขนาดปกติ
TEXT	65,535	แบบ TEXT ขนาดปกติ
MEDIUMBLOB	16,777,215	แบบ BLOB ขนาดกลาง
MEDIUMTEXT	16,777,215	แบบ TEXT ขนาดกลาง
LONGBLOB	4,294,967,295	แบบ BLOB ขนาดยาว
LONGTEXT	4,294,967,295	แบบ TEXT ขนาดยาว

# phpMyAdmin – String Type

## ชนิดข้อความพิเศษ SET และ ENUM

ชนิด	ความยาว สูงสุด	ความหมาย
SET('value1',value2,...)	64	เก็บค่าได้หลายค่าจากกลุ่มข้อความที่ระบุ
ENUM('value1',value2,...)	65,535	เก็บค่าได้ค่าเดียวได้จากกลุ่มข้อความที่ระบุ

# phpMyAdmin – Export file

The screenshot shows the phpMyAdmin 2.10.3 interface. The browser address bar indicates the URL is localhost:81 / localhost | phpMyAdmin 2.10.3. The interface is divided into several sections:

- Localhost Information:**
  - รุ่นของเซิร์ฟเวอร์: 5.0.51b-community-nt-log
    - Protocol version: 10
  - เซิร์ฟเวอร์: localhost via TCP/IP
    - ผู้ใช้: root@localhost
  - ชุดตัวอักษร (charset) ของ MySQL: UTF-8 Unicode (utf8)
  - MySQL connection collation: utf8\_unicode\_ci
  - สร้างฐานข้อมูลใหม่ (New Database): [input field] การเรียงลำดับ [dropdown] สร้าง
  - แสดงสถานะของ MySQL
  - แสดงตัวแปรระบบของ MySQL
  - โพรเซส
  - ชุดตัวอักษร และการเรียงลำดับ
  - Storage Engines
  - Reload privileges
  - สิทธิ์
  - ฐานข้อมูล
  - ส่งออก (Export) ← Click ที่ Export หรือส่งออก
  - Import
  - ออกจากระบบ
- phpMyAdmin - 2.10.3 Settings:**
  - MySQL client version: 5.0.51a
  - Used PHP extensions: mysql
  - Language: ภาษาไทย - Thai
  - หน้าตา: Original
  - Font size: 100%
  - เอกสารการใช้ phpMyAdmin
  - phpMyAdmin wiki
  - โฮมเพจอย่างเป็นทางการของ phpMyAdmin
    - [ChangeLog] [Subversion] [Lists]
- Database List:**
  - ฐานข้อมูล: 50xx (4)
  - orderheader
  - orderitem
  - product
  - user

# phpMyAdmin – Export file

เซิร์ฟเวอร์: localhost

ฐานข้อมูล SQL สถานะ ตัวแปร ชุดตัวอักษร Engines สิทธิ์ โพรเซส ส่งออก Import

View dump (schema) of databases

ส่งออก

เลือกฐานข้อมูลที่ต้องการ

phpmyadmin  
project  
question  
test  
**testoption**  
ttttttt

เลือกทั้งหมด / ไม่เลือกเลย

ข้อมูล CSV (ต้นด้วยเครื่องหมายลูกน้ำ ",")  
 ข้อมูล CSV สำหรับไมโครซอฟต์เอ็กเซล  
 Microsoft Excel 2000  
 Microsoft Word 2000  
 LaTeX  
 Open Document Spreadsheet

ตัวเลือก SQL

Add custom comment into header (In splits lines)

Enclose export in a transaction  
 Disable foreign key checks  
 SQL compatibility mode NONE

Database export options

Add DROP DATABASE

โครงสร้าง

Add DROP TABLE / DROP VIEW  
 Add IF NOT EXISTS  
 เพิ่มค่า AUTO\_INCREMENT  
 ใส่ 'backquote' ให้กับชื่อตารางและฟิลด์  
 Add CREATE PROCEDURE / FUNCTION

Add into comments

Creation/Update/Check dates

# phpMyAdmin – Export file

The screenshot shows the phpMyAdmin export interface. On the left, the 'ฐานข้อมูล' (Database) dropdown is set to 'testoption (3)'. A red arrow points from the text 'เลือกฐานข้อมูลที่ต้องการ' (Select the database you need) to this dropdown. The central panel shows a list of databases under 'ส่งเลือก' (Select): phpmyadmin, project, question, test, testoption (highlighted), and ttttttt. A red dashed box surrounds this list. Below it, the 'เลือกทั้งหมด / ไม่เลือกเลย' (Select all / Deselect all) link is visible. The 'ข้อมูล CSV' (CSV) and 'SQL' options are selected. On the right, the 'ตัวเลือก SQL' (SQL options) section is visible, with a red dashed box around the 'ส่งมาเป็นไฟล์' (Save as file) checkbox and the 'รูปแบบของชื่อไฟล์' (File name format) dropdown, which is set to 'SERVER'. The text 'Click เลือก checkbox' (Click select checkbox) is written in red at the bottom left. The 'Maximal length of created query' is set to 50000, and the 'Export type' is set to 'INSERT'.

เลือกฐานข้อมูลที่ต้องการ

Click เลือก checkbox

ส่งเลือก

phpmyadmin  
project  
question  
test  
testoption  
ttttttt

เลือกทั้งหมด / ไม่เลือกเลย

ข้อมูล CSV (เส้นด้วยเครื่องหมายลูกน้ำ ",")  
ข้อมูล CSV สำหรับไมโครซอฟต์เอ็กเซล  
Microsoft Excel 2000  
Microsoft Word 2000  
LaTeX  
Open Document Spreadsheet  
Open Document Text  
PDF  
SQL

ตัวเลือก SQL

Add custom comment into header (in splits lines)

Enclose export in a transaction  
 Disable foreign key checks

SQL compatibility mode: NONE

Database export options

Add DROP DATABASE

โครงสร้าง

โครงสร้าง  
 Add DROP TABLE / DROP VIEW  
 Add IF NOT EXISTS  
 เพิ่มค่า AUTO\_INCREMENT  
 ใส่ 'backquote' ให้กับชื่อตารางและฟิลด์  
 Add CREATE PROCEDURE / FUNCTION

Add into comments

Creation/Update/Check dates

ข้อมูล

ข้อมูล  
 คำสั่ง INSERT เสริมรูปแบบ  
 แทรกหลายระเบียนในคราวเดียว

Maximal length of created query: 50000

ใช้ delayed inserts  
 Use ignore inserts  
 ใช้เลขฐานสิบหกสำหรับช่องข้อมูลไบนารี

Export type: INSERT

ส่งมาเป็นไฟล์

รูปแบบของชื่อไฟล์<sup>(\*)</sup>: SERVER (  จำรูปแบบ )

# phpMyAdmin – Export file

Click เลือก checkbox

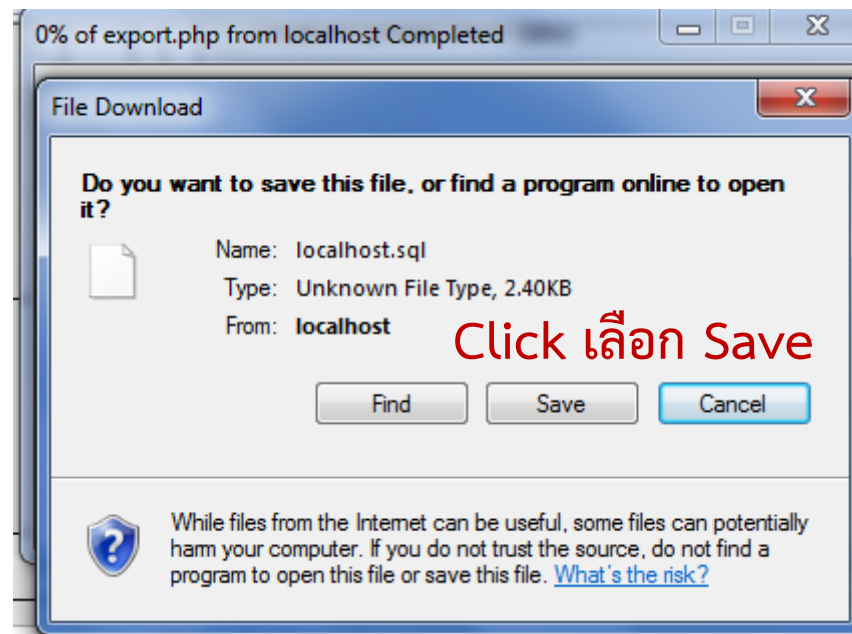
ส่งมาเป็นไฟล์

รูปแบบของชื่อไฟล์:  (  จำรูปแบบ )

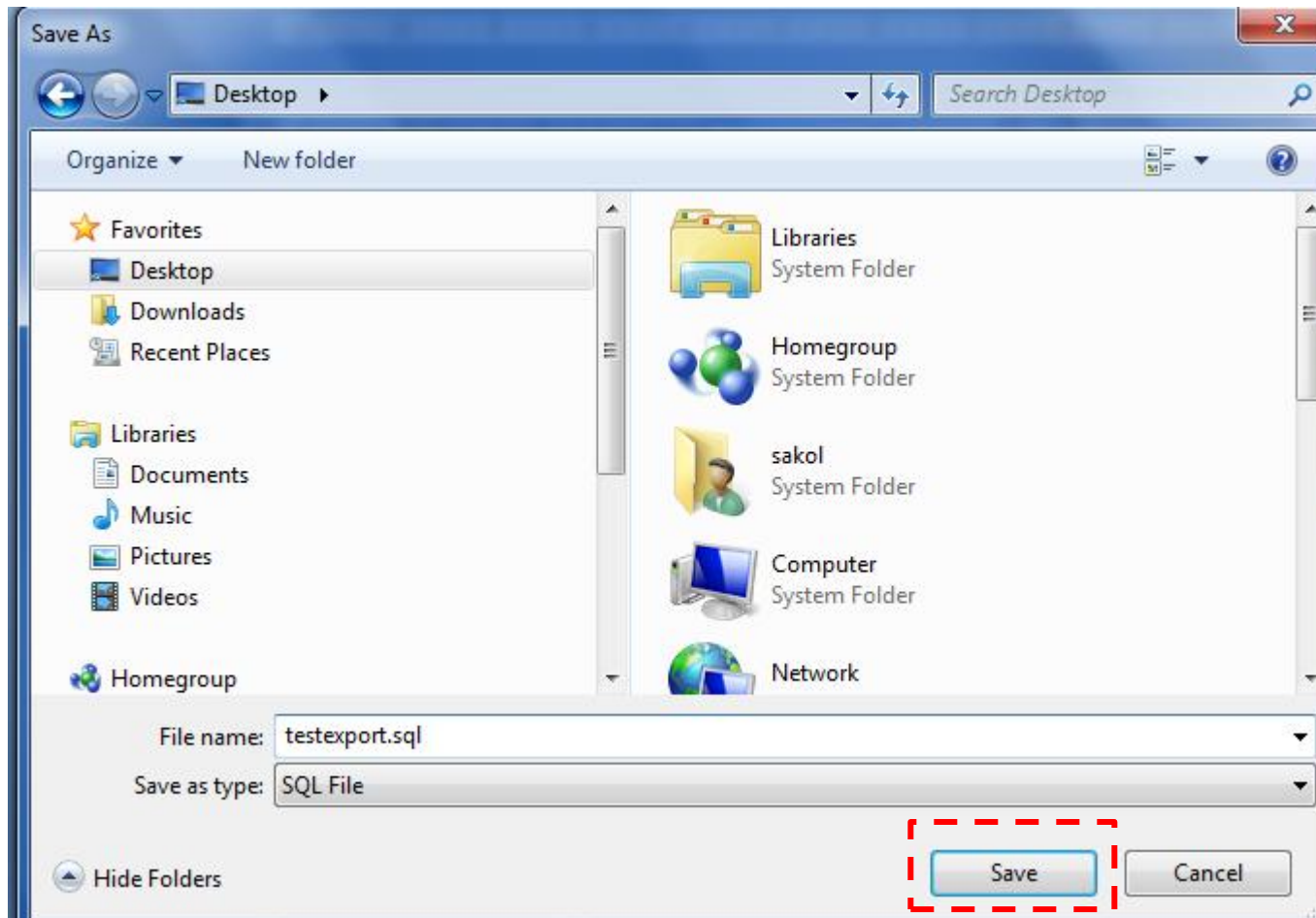
บีบอัดข้อมูล:  ไม่มี  "ถูกบีบอัดอยู่ (zip)"  "gzipped"

ลงมือ

Click เลือก

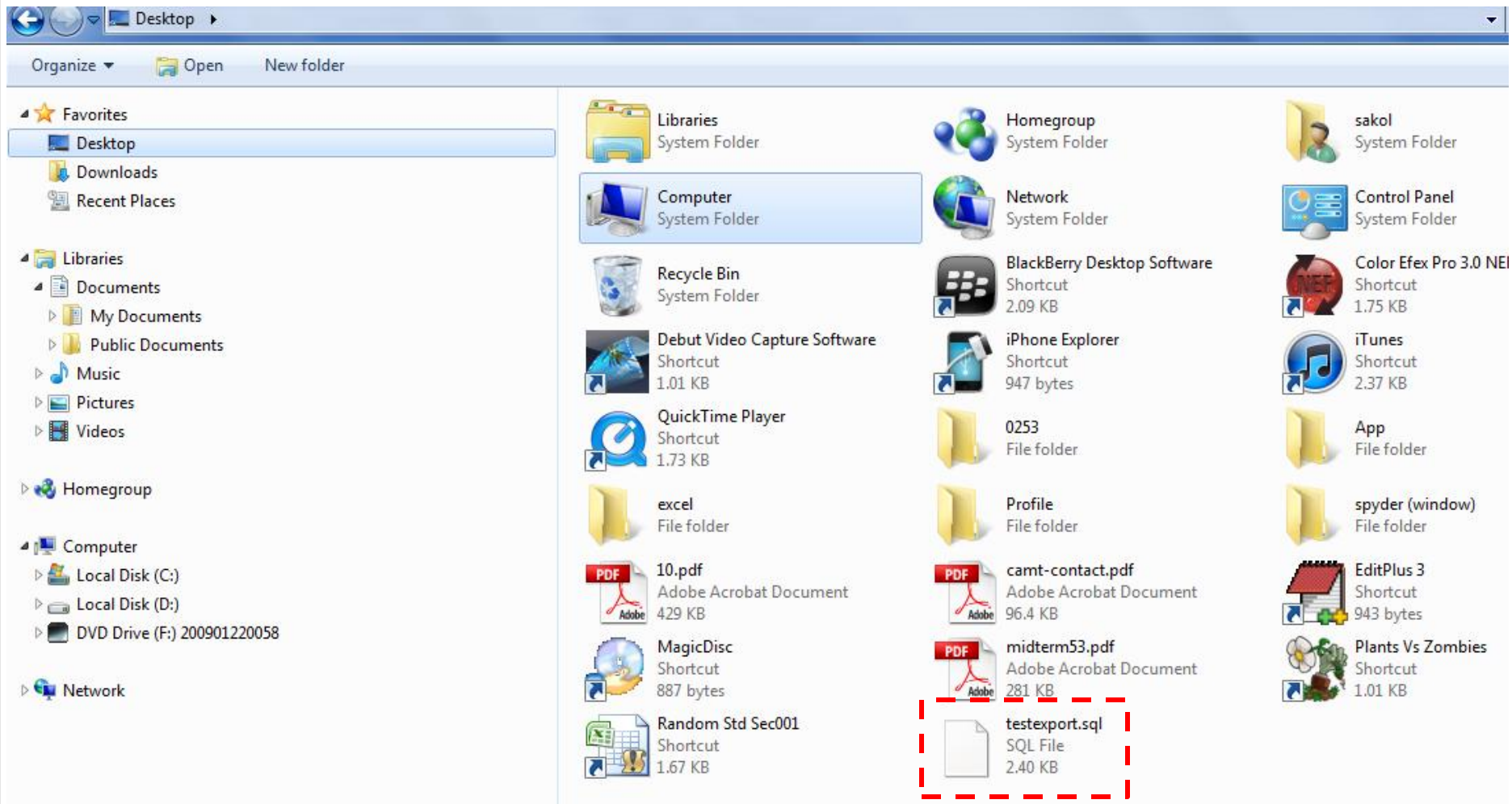


# phpMyAdmin – Export file



ทำเลือกที่เก็บไฟล์ฐานข้อมูลการบันทึก โดย  
สามารถตั้งชื่อไฟล์ฐานข้อมูล ต้องมีนามสกุล .sql

# phpMyAdmin – Export file



จะเห็นว่า มีไฟล์ฐานข้อมูลนามสกุล .sql  
ที่จะสามารถนำไปใช้ต่อไป