

เอกสารประกอบการสอน วิชาระบบฐานข้อมูล

โดย อ.วิชาญ หงษ์บิน

Part I

โปรแกรม Mysql

ครอบคลุม Version 3 - 5

บทที่ 1

ขั้นตอนการติดตั้งโปรแกรม

สาระการเรียนรู้

1. ขั้นตอนการติดตั้งโปรแกรม MySQL
2. การเริ่มต้นกับโปรแกรม MySQL
3. คำสั่งพื้นฐานของ SQL

มาตรฐานเชิงสมรรถนะ

1. สามารถติดตั้งโปรแกรม MySQL ได้
2. สามารถเข้าใช้โปรแกรม MySQL ได้อย่างถูกต้อง
3. สามารถใช้คำสั่งพื้นฐานในการแสดงรายการฐานข้อมูล คำสั่งใช้ฐานข้อมูล และออกจากโปรแกรม MySQL ได้

กิจกรรมเสนอแนะ

1. ให้นักศึกษาแบ่งกลุ่มทดสอบการติดตั้งโปรแกรม MySQL
2. ให้นักศึกษาฝึกปฏิบัติการใช้คำสั่งพื้นฐานเรียกดูฐานข้อมูลในระบบจากที่เตรียมไว้ให้

การวัดผล

1. การตอบข้อซักถามในชั้นเรียน
2. การสังเกต
3. ผลการฝึกปฏิบัติ

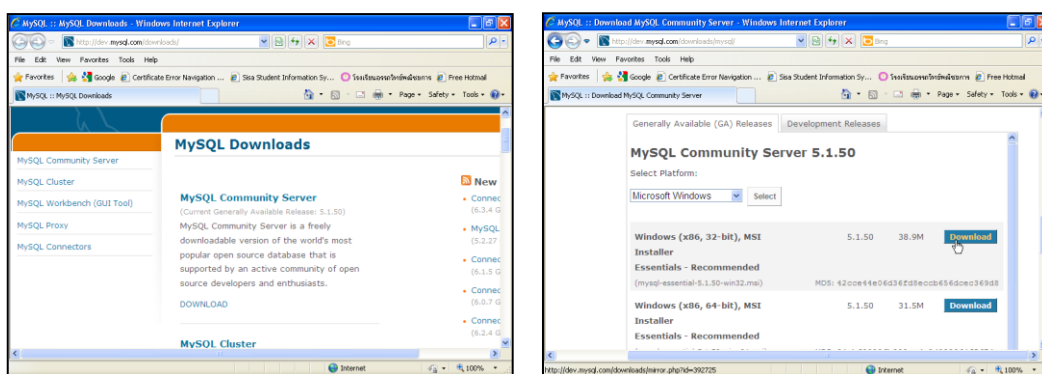
บทที่ 1

ขั้นตอนการติดตั้งโปรแกรม

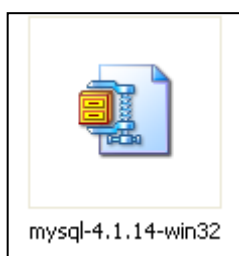
✳ การติดตั้งโปรแกรม MySQL

โปรแกรม MySQL เป็นโปรแกรมที่สามารถดาวน์โหลดมาใช้ฟรีได้จากเว็บไซต์ www.mysql.com/download ได้ โดยเลือกแบบ Free License ซึ่งความสามารถไม่ต่างไปจาก Commercial License มากนัก ตามลำดับขั้นตอนดังนี้

1. เข้าเว็บไซต์ www.mysql.com โดยเลือกรุ่นที่ใช้กับ Windows(x86) ซึ่งเป็นประเภท Zip ไฟล์



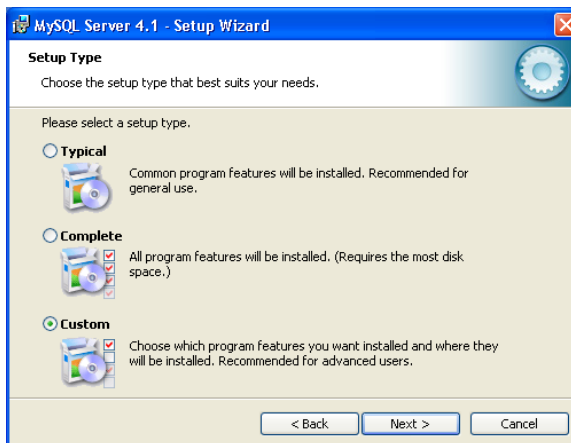
2. หลังจากที่ได้ดาวน์โหลดโปรแกรมมาได้แล้ว ให้ทำการขยายไฟล์ด้วย WinRAR หรือ WinZip หรือโปรแกรมประเภท UnZip อื่น ๆ ซึ่งภายหลังจากขยายไฟล์ได้เรียบร้อยแล้ว จะได้ไฟล์ Setup.exe เพียงไฟล์เดียว



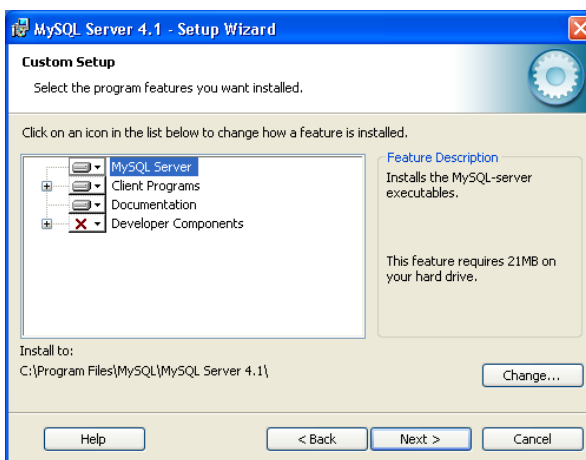
3. ดับเบิลคลิกไฟล์ Setup.exe ก็จะเริ่มเข้าสู่ขั้นตอนการติดตั้ง
4. เมื่อเข้าสู่ขั้นตอนการติดตั้งให้คลิกปุ่ม Next 1 เพื่อเริ่มขั้นตอนต่อไป



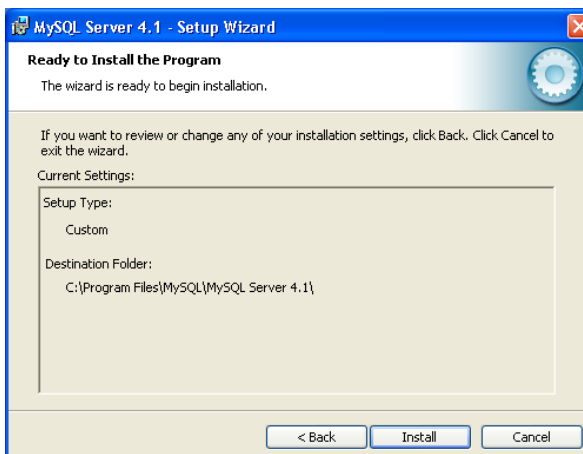
5. ต่อไปเลือกออกแบบการติดตั้งแบบ Custom เพื่อที่จะสามารถกำหนดตำแหน่งในการติดตั้งได้ แล้วคลิกปุ่ม Next



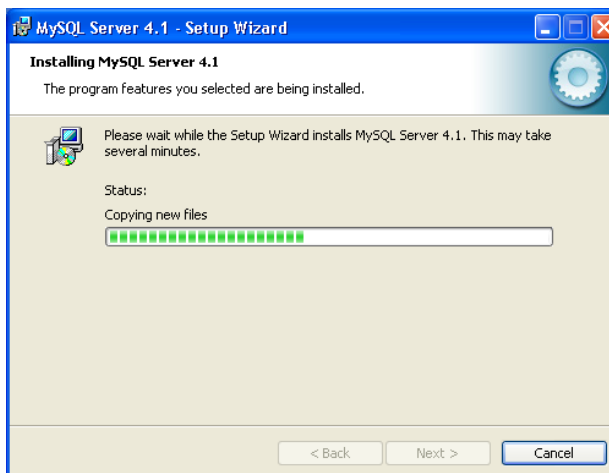
6. ที่ช่อง Install to : เป็นการกำหนดตำแหน่งในการติดตั้ง ซึ่งในกรณีผู้เขียนติดตั้งบน C:\Program Files\MySQL\MySQL Server 4.1 แต่ถ้าต้องการเปลี่ยนแปลงให้คลิกปุ่ม Change... แล้วเลือกแหล่งเก็บข้อมูลที่ต้องการ



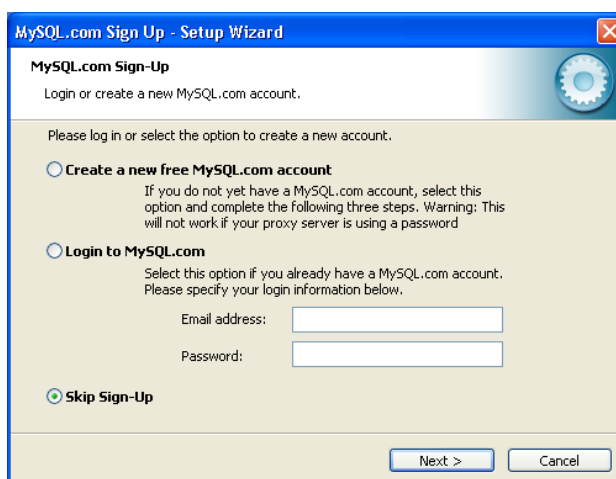
7. คลิกที่ปุ่ม Install เพื่อเริ่มการติดตั้ง



8. รอจนกว่าการคัดลอกไฟล์จะเสร็จสิ้น แล้วคลิกปุ่ม Next



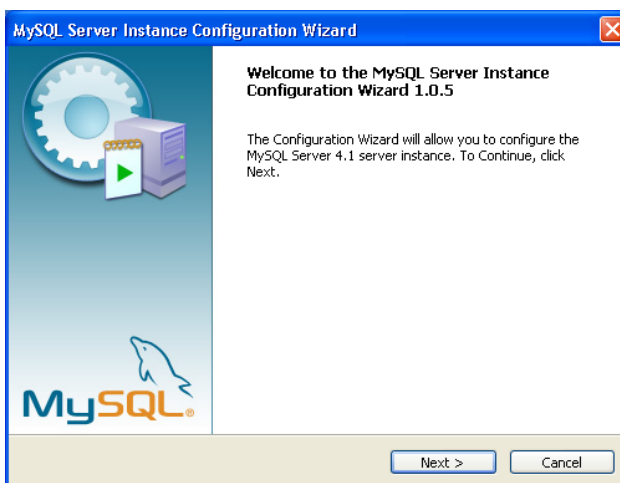
9. ขั้นตอนนี้เป็นการสอบถามการลงทะเบียนกับ MySQL.Com ซึ่งให้คลิก Skip Sign-Up เพื่อข้ามขั้นตอนนี้ไปก่อน แล้วคลิก Next



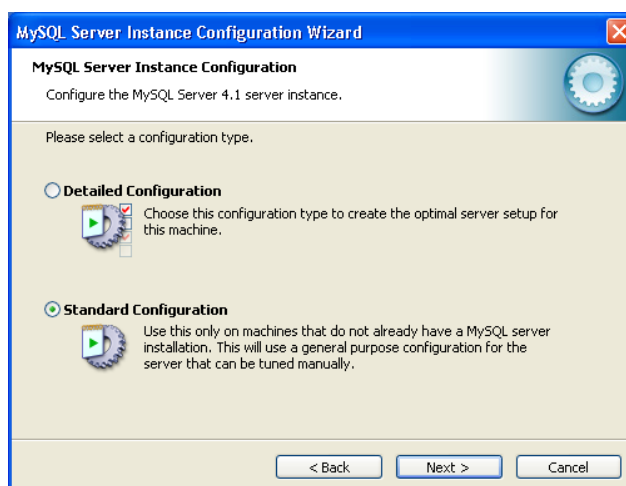
10. คลิกเลือก Configure the MySQL Server now แล้วคลิก Next



11. คลิกปุ่ม Next



12. คลิกเลือก Standard Configuration แล้วคลิกปุ่ม Next



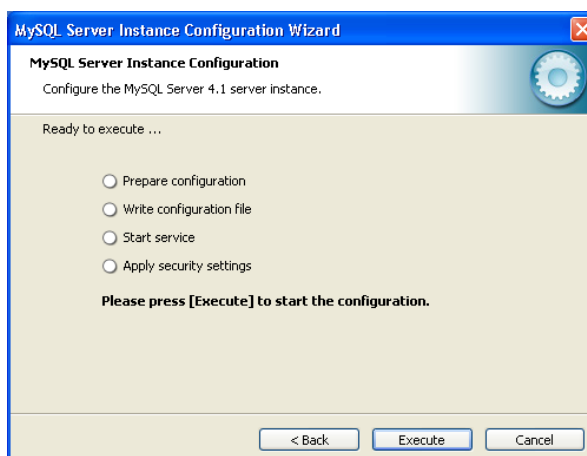
13. คลิกเลือก Install As Windows Service เพื่อให้ MySQL เริ่มทำงานโดยอัตโนมัติเมื่อเปิดโปรแกรมวินโดวส์ และเลือก Include Bin Directory in Windows PATH เพื่อให้สามารถใช้ MySQL จากตำแหน่งใดก็ตามที่เรียกใช้ก็ได้ แล้วคลิกปุ่ม Next



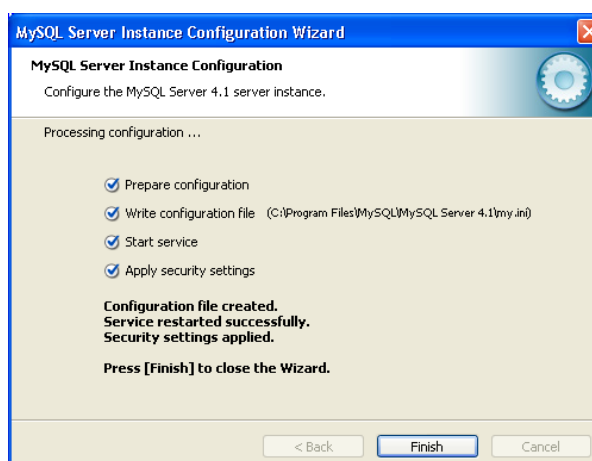
14. กำหนดรหัสผ่านสำหรับผู้ดูแลระบบ ซึ่งรหัสผ่านนี้มีความสำคัญมาก เพราะเวลาเข้าใช้งานโปรแกรม MySQL โปรแกรมจะให้ใส่รหัสผ่านที่เคยกำหนดไว้ในขั้นตอนการติดตั้ง หากลืมจะต้อง UnInstall แล้วทำการติดตั้งใหม่ตั้งแต่ขั้นตอนแรกเพื่อเข้ามากำหนดรหัสผ่าน



15. คลิกปุ่ม Execute เพื่อ Setup ค่าทั้งหมดที่ติดตั้งมาตั้งแต่ขั้นตอนแรก

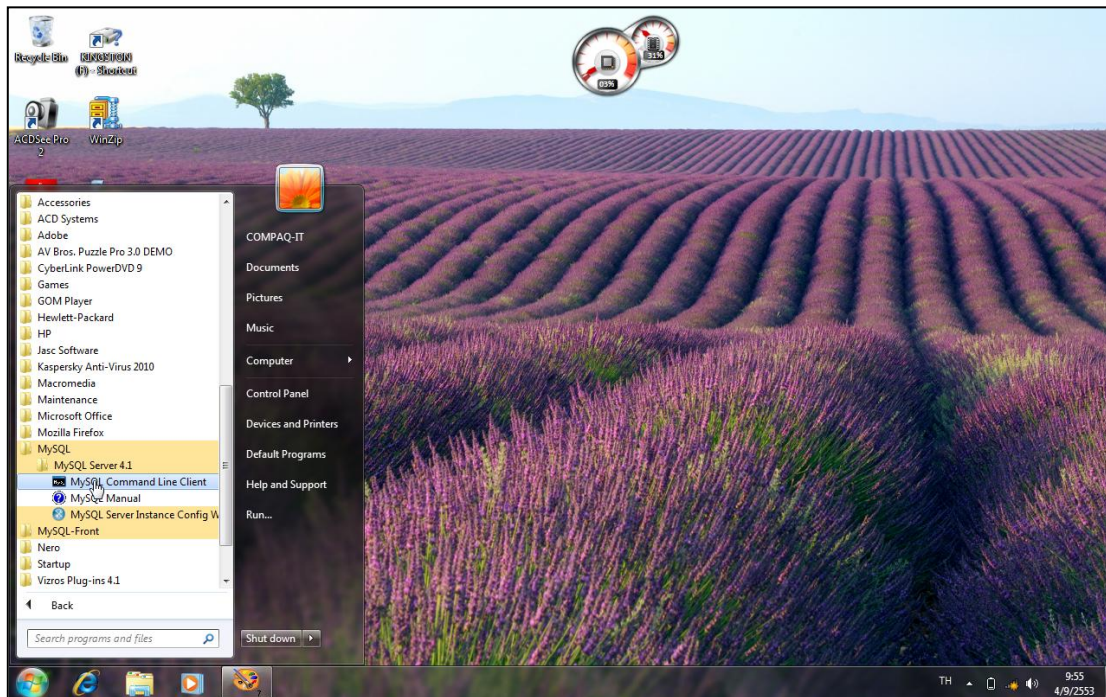


16. คลิกปุ่ม Finish เพื่อสิ้นสุดการติดตั้ง (ถ้ามีปัญหาในการติดตั้ง โปรแกรมจะแจ้งเป็นข้อความ Error แสดงบนจอภาพ ให้แก้ไขไปตามคำแนะนำนั้น)

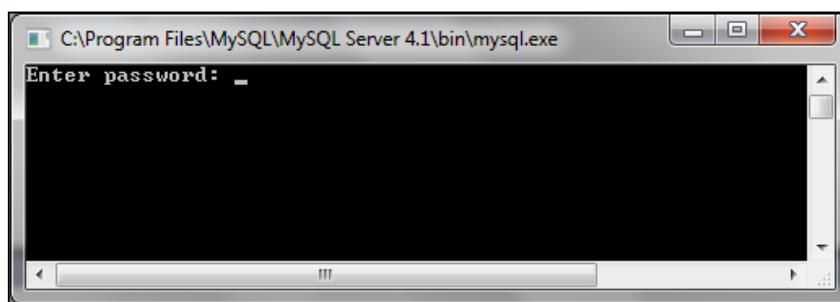


※ การเริ่มต้นใช้งาน

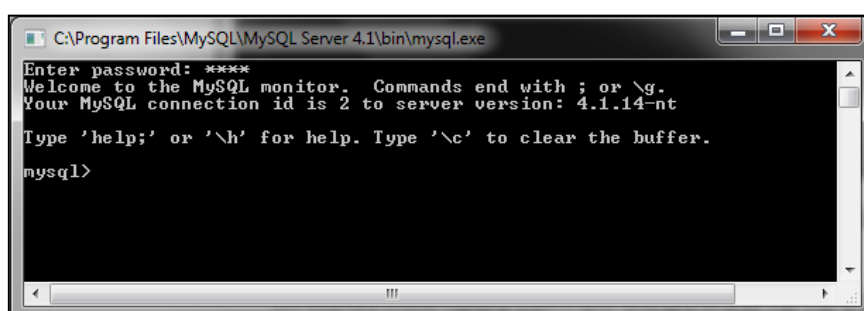
1. คลิกเมนู Start > All Programs > MySQL > MySQLServer 4.1 > MySQL Command Line Client



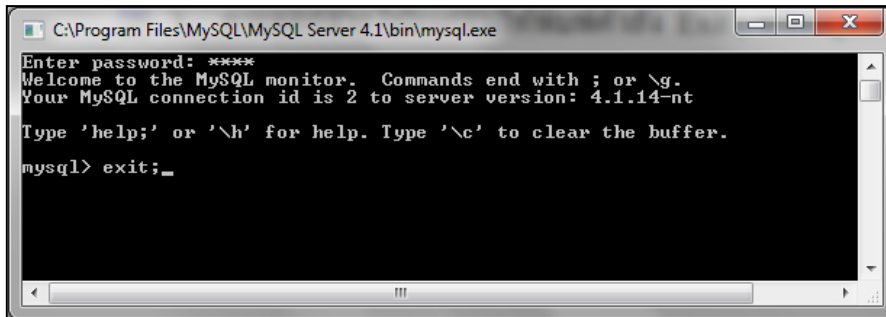
2. จะปรากฏหน้าจอเข้าโปรแกรม ดังรูป



ให้พิมพ์รหัสผ่านลงไปเหมือนกับรหัสผ่านในขั้นตอนการติดตั้งตั้งแต่แรก ถ้าหากใส่รหัสผ่านผิด โปรแกรมจะปิดหน้าต่างการทำงานลงและไม่อนุญาตให้เข้าใช้งาน ดังนั้นหากผู้ใช้ลืมรหัสผ่านจะต้องทำการ Uninstall โปรแกรม MySQL แล้วทำการติดตั้งใหม่ แต่ถ้าใส่รหัสผ่านถูกต้อง โปรแกรมจะปรากฏหน้าจอดังภาพ



3. การออกจากโปรแกรมให้พิมพ์คำสั่ง Exit ; หรือ Quit ; เช่น

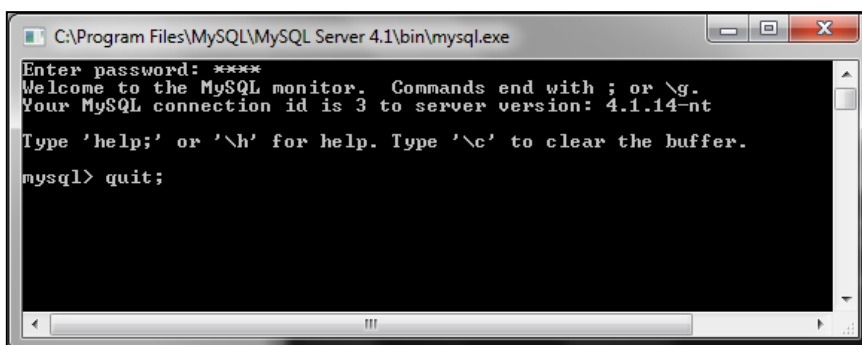


```
C:\Program Files\MySQL\MySQL Server 4.1\bin\mysql.exe
Enter password: ****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 2 to server version: 4.1.14-nt

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> exit;_
```

หรือ



```
C:\Program Files\MySQL\MySQL Server 4.1\bin\mysql.exe
Enter password: ****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 3 to server version: 4.1.14-nt

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> quit;
```

กิจกรรมเสนอแนะเชิงเจตคติ

จงเติมข้อความในช่องว่างให้สมบูรณ์

1. จงบอกขั้นตอนการ Download โปรแกรม MySQL จากเว็บไซต์ที่เกี่ยวข้อง

.....
.....
.....
.....
.....

2. จงบอกขั้นตอนการติดตั้งโปรแกรม MySQL

.....
.....
.....
.....
.....
.....
.....
.....

3. จงบอกขั้นตอนการใช้โปรแกรม MySQL ภายหลังจากติดตั้งเสร็จเรียบร้อยแล้ว

.....
.....

4. จากหน้าจอ ‘ Enter Password : ’ รหัสที่ผู้ใช้จะกีย์เข้าไปจะนำมาจากส่วนไหน

.....
.....

5. ถ้าต้องการออกจากโปรแกรม ควรพิมพ์คำสั่งว่าอะไร

.....
.....

บทที่ 2

คำสั่งพื้นฐานของ MySQL

สาระการเรียนรู้

1. คำสั่งพื้นฐานของ MySQL
2. คำสั่งสร้างฐานข้อมูลด้วย MySQL
3. ชนิดของข้อมูลใน MySQL

มาตรฐานเชิงสมรรถนะ

1. สามารถใช้คำสั่งพื้นฐานในการแสดงฐานข้อมูล และ โครงสร้างของฐานข้อมูลได้
2. สามารถใช้คำสั่งสร้างฐานข้อมูลได้อย่างถูกต้องและเหมาะสม
3. สามารถกำหนดชนิดข้อมูลให้เหมาะสมกับตารางฐานข้อมูลได้
4. สามารถอธิบายแอททริบิวต์ที่เกี่ยวข้องกับฐานข้อมูลในแต่ละคอลัมน์ได้
5. สามารถวิเคราะห์ และเปรียบเทียบวิธีการกำหนดชนิดข้อมูลแบบต่าง ๆ ได้

กิจกรรมเสนอแนะ

1. ให้นักศึกษาฝึกปฏิบัติการใช้คำสั่ง MySQL ในการจัดการฐานข้อมูล
2. จัดหาใบงานพิเศษมอบหมายให้นักศึกษาฝึกปฏิบัติ
3. ให้นักศึกษาค้นคว้ารูปแบบของคำสั่ง MySQL เพิ่มเติมจากแหล่งอื่น ๆ ได้

การวัดผล

1. การตอบข้อซักถามในชั้นเรียน
2. การสังเกต
3. ผลการฝึกปฏิบัติ
4. แบบฝึกหัดท้ายบท / ใบงานพิเศษที่มอบหมาย

บทที่ 2

คำสั่งพื้นฐานของ MySQL

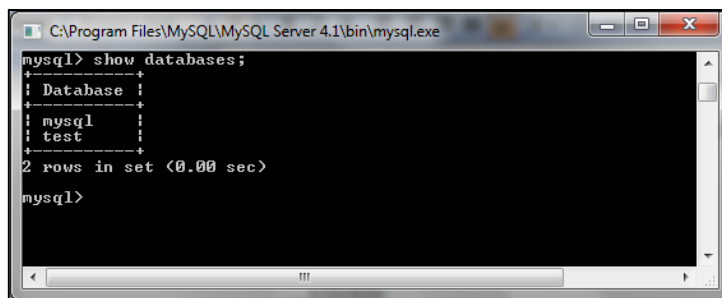
※ คำสั่งพื้นฐานของ MySQL

ในการใช้งานโปรแกรม MySQL จำเป็นต้องพิมพ์คำสั่งในการควบคุมหรือกำหนดขั้นตอนการทำงานทุกขั้นตอน เป็นลักษณะของเท็กซ์โหมด (Text Mode) โดยคำสั่งแต่ละคำสั่งจะมีรูปแบบการทำงานที่แน่นอน สามารถพิมพ์เป็นตัวพิมพ์เล็ก หรือตัวพิมพ์ใหญ่ หรือพิมพ์ผสมทั้งพิมพ์เล็ก และพิมพ์ใหญ่ก็ได้ แต่จะต้องจบประโยคคำสั่งด้วยเครื่องหมาย ; เสมอ

※ คำสั่ง Show Databases

เป็นคำสั่งแสดงรายชื่อฐานข้อมูลทั้งหมดที่มีอยู่
รูปแบบ

```
Show Databases ;
```



```
CA\Program Files\MySQL\MySQL Server 4.1\bin\mysql.exe
mysql> show databases;
+-----+
| Database |
+-----+
| mysql    |
| test     |
+-----+
2 rows in set (0.00 sec)

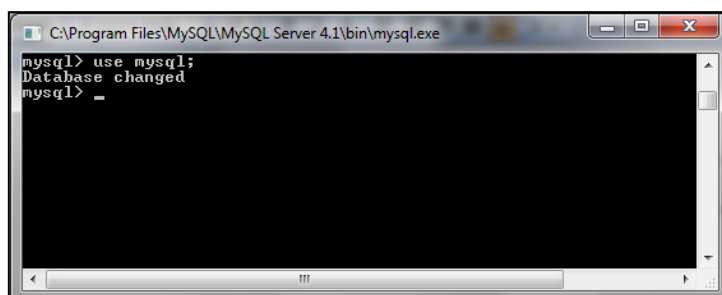
mysql>
```

※ คำสั่ง Use

เป็นคำสั่งเริ่มต้นก่อนใช้งานฐานข้อมูล โดยต้องใช้ทุกครั้งที่เราเริ่มใช้งานฐานข้อมูล
รูปแบบ

```
Use ชื่อฐานข้อมูล ;
```

เช่น ถ้าต้องการใช้ฐานข้อมูลชื่อ mysql ให้พิมพ์คำสั่งดังภาพ

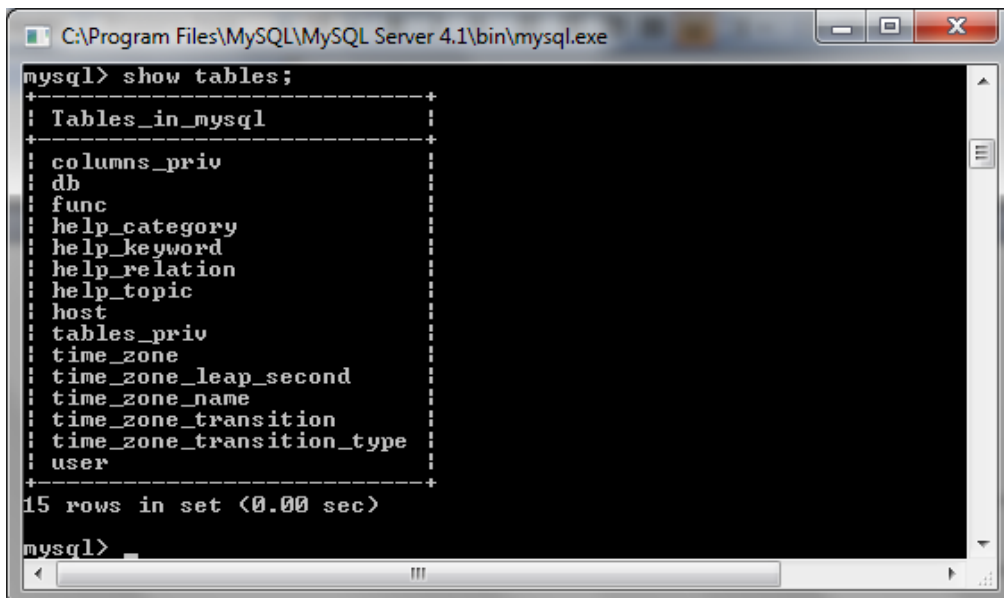
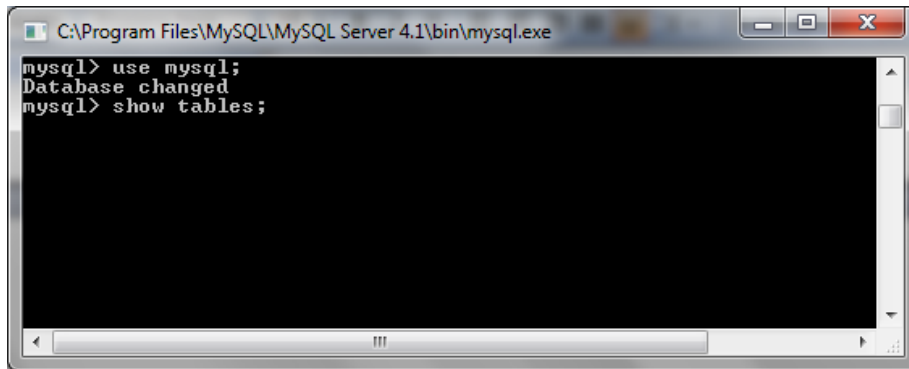


```
CA\Program Files\MySQL\MySQL Server 4.1\bin\mysql.exe
mysql> use mysql;
Database changed
mysql> _
```

※ คำสั่ง Show Tables

เป็นคำสั่งแสดงรายชื่อตาราง (Table) ทั้งหมดที่มีอยู่ในฐานข้อมูล
รูปแบบ

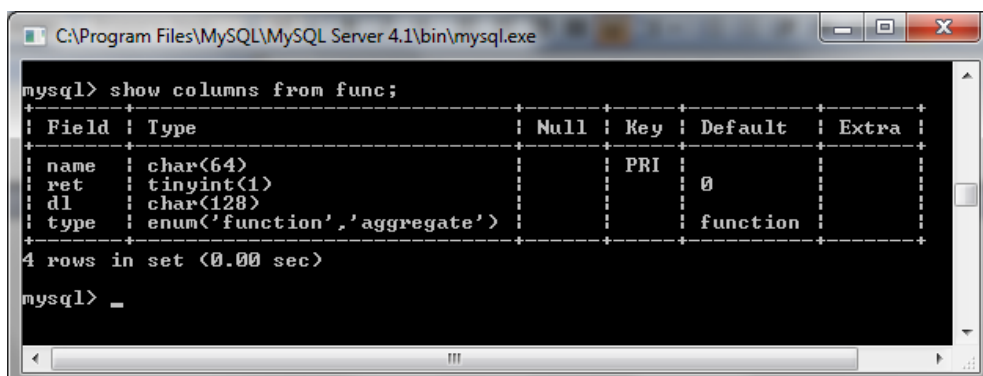
Show Tables ;




※ คำสั่ง Show Columns

เป็นคำสั่งแสดงชื่อคอลัมน์และชนิดข้อมูลของแต่ละตาราง
รูปแบบ

Show Columns From ชื่อตาราง ;



 กิจกรรมทบทวน ให้นักศึกษาทดสอบการใช้คำสั่ง ดังนี้

1. ใช้คำสั่งแสดงรายชื่อฐานข้อมูล
2. ใช้คำสั่งใช้ฐานข้อมูล test
3. ใช้คำสั่งแสดงรายชื่อตารางในฐานข้อมูล
4. ใช้คำสั่งแสดงคอลัมน์ของตาราง อย่างน้อย 2 ตาราง

✳ **การสร้างฐานข้อมูล**

โดยทั่วไปการสร้างฐานข้อมูลจะประกอบไปด้วยโครงสร้างที่มีลำดับขั้นตอน ซึ่งประกอบไปด้วย

1. ฐานข้อมูล (Database)
2. ตารางฐานข้อมูล (Table)
3. คอลัมน์ของตาราง (Column)
4. แถวข้อมูล (Row)

สำหรับการทำงานกับโปรแกรม MySQL ก็เช่นเดียวกัน การดำเนินงานจะต้องเริ่มจากขั้นตอนการสร้างฐานข้อมูลก่อน จากนั้นจึงสร้างตารางฐานข้อมูลที่เกี่ยวข้อง โดยแต่ละคอลัมน์ของตารางจะกำหนดชื่อและชนิดของข้อมูลที่เกี่ยวข้อง หลังจากนั้นจึงเริ่มบันทึกข้อมูลในแต่ละแถวให้กับตารางข้อมูล ลักษณะของตารางก็เหมือนกับการใช้งานโปรแกรมตารางงานหรือโปรแกรม Microsoft Excel โดยทั่วไปที่แยกออกเป็นคอลัมน์และแถว

Database

<p>Table 1</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 33%;">Column1</th> <th style="width: 33%;">Column2</th> <th style="width: 33%;">Column3</th> </tr> </thead> <tbody> <tr> <td> </td> <td> </td> <td> </td> </tr> <tr> <td> </td> <td> </td> <td> </td> </tr> <tr> <td> </td> <td> </td> <td> </td> </tr> </tbody> </table>	Column1	Column2	Column3										<p>Table 2</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 50%;">Column1</th> <th style="width: 50%;">Column2</th> </tr> </thead> <tbody> <tr> <td> </td> <td> </td> </tr> <tr> <td> </td> <td> </td> </tr> <tr> <td> </td> <td> </td> </tr> </tbody> </table>	Column1	Column2						
Column1	Column2	Column3																			
Column1	Column2																				
rows																					

✳ **ชนิดของข้อมูลในคอลัมน์**

เนื่องจากข้อมูลที่จะเก็บในฐานข้อมูลมีความหลากหลายมีทั้งชนิดข้อความ ชนิดตัวเลข ชนิดวันเดือนปี หรือแม้แต่กระทั่งรูปภาพ มัลติมีเดีย ต่าง ๆ มากมาย ดังนั้นการจัดเก็บข้อมูล

เหล่านั้นจำเป็นต้องมีการกำหนดชนิดของข้อมูลให้เหมาะสม สำหรับโปรแกรม MySQL มีการกำหนดประเภทของข้อมูลแต่ละชนิดไว้ ดังนี้

✖ ข้อมูลชนิดข้อความ หรือสตริง

ประเภท	จำนวนอักขรสูงสุด
Char (จำนวนตัวอักษร)	255
Varchar(จำนวนตัวอักษร)	255
TinyText	255
Text	65535
MediumText	16777215
LongText	4294967295

✖ ข้อมูลชนิดตัวเลข (Numeric Type)

ประเภท	Sign	ค่าต่ำสุด	ค่าสูงสุด
Tinyint	Signed	-128	127
	Unsigned	0	255
SmallInt	Signed	-32768	32767
	Unsigned	0	65535
MediumInt	Signed	-8388608	8388607
	Unsigned	0	16777215
Int	Signed	-2147483648	2147483647
	Unsigned	0	4294967295
BigInt	Signed	-9223372036854775808	9223372036854775807
	Unsigned	0	18446744073709551615
Float	Signed	-3.402823466E+38	-1.175494351E-38
	Unsigned	1.175494351E-38	3.402823466E+38
Double	Signed	1.7976931348623157E+308	-2.2250738585072914E-308
	Unsigned	2.2250738585072914E-308	1.7976931348623157E+308

※ ข้อมูลชนิด BLOB

ข้อมูลชนิด BLOB (Binary Large Object) ใช้กับข้อมูลประเภทรูปภาพหรือมัลติมีเดียที่มีรูปแบบข้อมูลเป็นแบบ Binary

ประเภท	จำนวนสูงสุด
Tinyblob	255
Blob	65535
Mediumblob	16777215
Longblob	4294967295

※ ข้อมูลชนิด set และ enum

ข้อมูลประเภท set และ enum คือ กลุ่มของข้อมูลที่มีค่าที่แน่นอน เมื่อกำหนดให้เป็นคุณสมบัติของคอลัมน์ใดแล้ว เวลากรอกข้อมูลให้กับคอลัมน์นั้นจะต้องมีค่าเป็นสมาชิกของ set หรือ enum ที่กำหนดให้เท่านั้น เช่น

ประเภท	รูปแบบ	จำนวนสมาชิกสูงสุด
set	Set("member1", "member2", ..., "member64")	64
enum	Enum("member1", "member2", ..., "member65535")	65535

※ ข้อมูลชนิดวันเวลา

ข้อมูลชนิดวันเวลาในโปรแกรม MySQL จะเก็บอยู่ในรูปแบบของ YYYY-MM-DD หรือ ปี ค.ศ.-เดือน-วันที่ เช่น วันที่ 14 พฤษภาคม ค.ศ. 1980 จะต้องเขียนเป็น 1980-05-14 ซึ่งในทางปฏิบัติจะมีฟังก์ชันที่ใช้แปลงค่าของวันเวลาให้อยู่ในรูปแบบที่ต้องการได้

ประเภท	คำอธิบาย	ช่วงข้อมูล
Date	เฉพาะข้อมูลวันเดือนปี	1000-01-01 ถึง 9999-12-31
Time	เฉพาะข้อมูลเวลา	-838:59:59 ถึง 838:59:59
Datetime	ทั้งวันเดือนปีและเวลา	1000-01-01 00:00:00 ถึง 9999-12-31 23:59:59

※ แอทริบิวต์ของคอลัมน์

นอกจากการกำหนดชนิดของข้อมูลให้แต่ละคอลัมน์แล้ว การใช้โปรแกรม MySQL ยังสามารถกำหนดคอปชันเพิ่มเติมให้กับคอลัมน์ได้ ซึ่งแอทริบิวต์เหล่านี้จะช่วยให้งานสะดวกเพิ่มมากขึ้น และสามารถช่วยตรวจสอบความถูกต้อง รวมไปถึงลดข้อผิดพลาดในการบันทึกข้อมูลได้มากขึ้น

แอทริบิวต์	รายละเอียด
Null	ข้อมูลนั้นจะไม่สามารถคีย์ข้อมูลอะไรลงไปได้เลย คือต้องเป็นค่าว่างเท่านั้น ซึ่งไม่นิยมใช้งาน
Not Null	ข้อมูลนั้นจะว่างไม่ได้ ต้องมีการคีย์ข้อมูลลงไปเสมอ โดยเฉพาะข้อมูลที่สำคัญเพื่อควบคุมความถูกต้องของข้อมูล
Index	ใช้เรียงลำดับข้อมูล ปกติตั้งค่าแบบเรียงลำดับจากน้อยไปหามาก Asc ส่วนการเรียงลำดับจากมากไปหาน้อยเรียกว่า Desc การกำหนด Index สามารถทำได้มากกว่า 1 คอลัมน์ โดยไฟล์ Index จะถูกแยกออกไปเป็นไฟล์ต่างหากเนื่องจากเป็นเรื่องที่สำคัญของฐานข้อมูล
Primary Key	เป็นการกำหนดคีย์หลักให้กับตาราง โดยคอลัมน์ที่จะกำหนดให้เป็น Primary Key จะต้องมียุคข้อมูลที่ไม่ซ้ำกัน เช่น รหัส หรือ หมายเลขของข้อมูล ที่มีค่าไม่ซ้ำกันทั้งตาราง การกำหนด Primary Key สามารถทำได้มากกว่า 1 คอลัมน์ แต่ไม่ค่อยได้รับความนิยม อย่างไรก็ตามตารางควรมีอย่างน้อย 1 คอลัมน์ที่กำหนดเป็น Primary Key
Unique	ใช้กำหนดให้ข้อมูลในคอลัมน์นั้น ๆ ห้ามซ้ำกัน หรือมีค่าที่ซ้ำกันไม่ได้ เช่น คอลัมน์ email หากมีการพิมพ์หรือรับข้อมูลซ้ำกับที่เคยมีมาแล้ว โปรแกรมจะไม่ยอมรับ ซึ่งช่วยควบคุมและตรวจสอบฐานข้อมูลให้มีความถูกต้องมากขึ้น
Auto_Increment	หมายถึงโปรแกรม MySQL จะใส่ตัวเลขให้กับคอลัมน์นั้น โดยอัตโนมัติ โดยเพิ่มค่าขึ้นทีละ 1 และผู้ใช้ไม่สามารถเปลี่ยนแปลงค่าได้ ทั้งนี้คอลัมน์ที่จะกำหนดเป็นแบบ Auto_Increment ได้ จะต้องกำหนดชนิดข้อมูลเป็นแบบเลขจำนวนเต็ม เช่น Int และห้ามใช้กับแอทริบิวต์ Null แต่ส่วนมากนิยมใช้คู่กับ Not Null
Binary	ใช้กับข้อมูลชนิด Char หรือ Varchar เพื่อให้ฐานข้อมูลนำความแตกต่างของรูปแบบพิมพ์เล็กพิมพ์ใหญ่มาพิจารณาด้วย
Default	เป็นการกำหนดค่าใดค่าหนึ่งเอาไว้ล่วงหน้าให้กับคอลัมน์นั้น ๆ โดยส่วนใหญ่ค่าที่จะกำหนดเป็น Default ควรเป็นค่าที่ใช้บ่อย ๆ

ตัวอย่างการกำหนดชนิดข้อมูล

※ ตัวอย่างข้อมูล

รหัสวิชา	ชื่อวิชา	หน่วยกิต
3000-1001	ภาษาไทยเพื่ออาชีพ 1	2
3204-1001	การวิเคราะห์และออกแบบระบบ	3
3204-1002	ระบบสารสนเทศเพื่อการจัดการ	3
3000-1423	สถิติ	3
3000-2007	กิจกรรมองค์การวิชาชีพ 7	0

※ โครงสร้างของตาราง

ชื่อคอลัมน์	ชนิดข้อมูล	แอทริบิวต์
subid	Varchar(10)	Not Null
subname	Varchar(30)	Not Null
credit	Int	

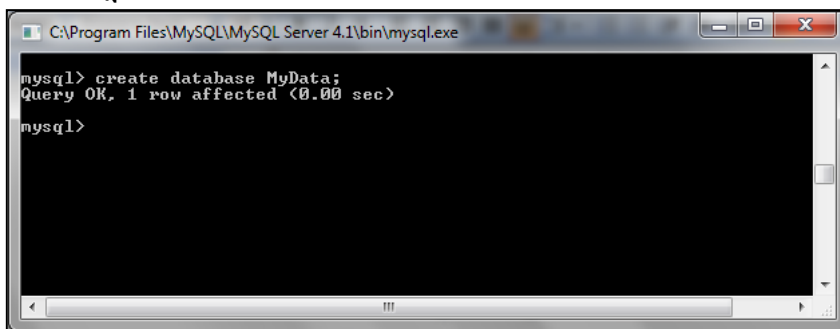
※ เริ่มต้นสร้างฐานข้อมูล

1. สมมติให้สร้างฐานข้อมูลชื่อ MyData

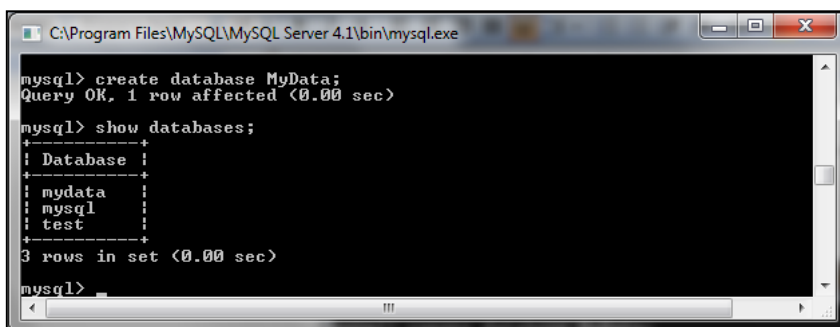
รูปแบบ

Create Database ชื่อฐานข้อมูล ;

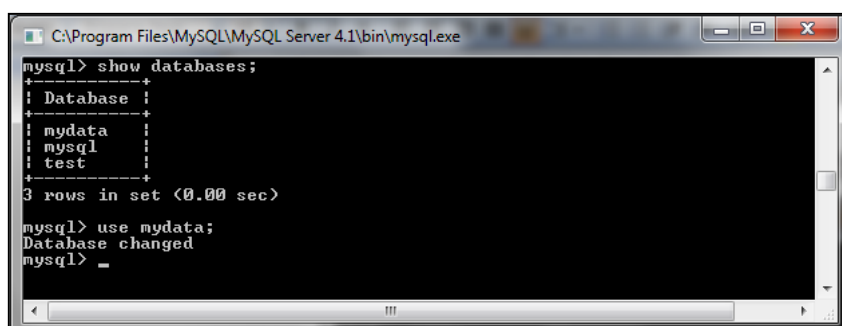
จะปรากฏผลดังภาพ



ทดลองใช้คำสั่ง Show Databases เพื่อดูรายชื่อฐานข้อมูลทั้งหมด ดังรูป



2. ใช้คำสั่ง Use ฐานข้อมูล เพื่อเริ่มสร้างตาราง



3. การสร้างตารางฐานข้อมูล

รูปแบบ

```

Create Table ชื่อตาราง (
ชื่อคอลัมน์_1 ชนิดข้อมูล [attribute(s)] ,
ชื่อคอลัมน์_2 ชนิดข้อมูล [attribute(s)] ,
ชื่อคอลัมน์_3 ชนิดข้อมูล [attribute(s)] ,
.....
ชื่อคอลัมน์_n ชนิดข้อมูล [attribute(s)] ,
);

```

เช่น ให้สร้างตารางข้อมูลชื่อ Subject และกำหนดโครงสร้างข้อมูล ดังนี้

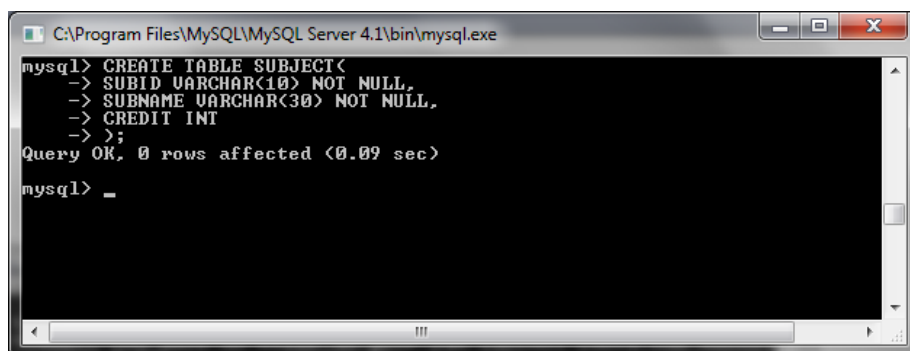
ชื่อคอลัมน์	ชนิดข้อมูล	แอทริบิวต์
Subid	Varchar(10)	Not Null
subname	Varchar(30)	Not Null
Credit	Int	

```

Create Table Subject (
Subid varchar(10) not null ,
Subname varchar(30) not null ,
Credit int
);

```

ดั่งภาพ



กรณีพิมพ์คำสั่งผิด ถ้าตำแหน่งที่ผิดอยู่บรรทัดเดียวกับเคอร์เซอร์ให้ใช้ปุ่มลูกศรเลื่อนเคอร์เซอร์ไปยังตำแหน่งนั้นแล้วแก้ไข แต่ถ้าหากตำแหน่งที่ผิดอยู่คนละบรรทัดกับเคอร์เซอร์เนื่องจากเราไม่สามารถเลื่อนเคอร์เซอร์ข้ามบรรทัดได้ ดังนั้นวิธีการแก้คำสั่งที่ผิดก็ต้องพิมพ์เครื่องหมาย ; แล้วกด <enter> จากนั้นพิมพ์คำสั่งใหม่

หลังจากสร้างตารางแล้ว หากไม่มีข้อผิดพลาดให้พิมพ์คำสั่ง Show tables และ Show Columns เพื่อตรวจสอบผลลัพธ์ที่ได้ว่าถูกต้องตามที่ต้องการหรือไม่

4. ให้นักศึกษาทดลองสร้างตารางสาขาวิชา (major)

ชื่อคอลัมน์	ชนิดข้อมูล	แอทริบิวต์
MCode	Int	Not Null
Mname	Varchar(35)	Not Null
Shotname	Char(10)	

※ การกำหนดแอทริบิวต์ให้กับคอลัมน์

การใช้แอทริบิวต์เพื่อกำหนดคุณลักษณะพิเศษเพิ่มเติมให้กับคอลัมน์ ไม่ว่าจะเป็น Index , Not Null, Unique, Primary Key, Auto_Increment หรือ Default นั้น บางแอทริบิวต์สามารถนำไปวางต่อท้ายคอลัมน์ในขณะที่พิมพ์คำสั่งได้ แต่บางแอทริบิวต์ไม่สามารถทำได้ เช่นคำสั่ง Index เป็นต้น ดังนั้นการสร้างตารางจึงสามารถสร้างได้ในอีกลักษณะหนึ่ง คือ

รูปแบบ

```

Create Table ชื่อตาราง (
ชื่อคอลัมน์_1 ชนิดข้อมูล [attribute(s)] ,
ชื่อคอลัมน์_2 ชนิดข้อมูล [attribute(s)] ,
ชื่อคอลัมน์_3 ชนิดข้อมูล [attribute(s)] ,
.....
ชื่อคอลัมน์_n ชนิดข้อมูล [attribute(s)] ,
attribute_1 (คอลัมน์,คอลัมน์,...),
attribute_2 (คอลัมน์,คอลัมน์,...),
.....
);

```

※ การกำหนด Primary Key ให้กับหลายคอลัมน์

สำหรับการกำหนด Primary Key (PK) ก็ย้หลักนั้น ในตารางแต่ละตารางสามารถกำหนดให้คอลัมน์ใดคอลัมน์หนึ่งเป็น Primary Key เพียงคอลัมน์เดียวก็ได้ หรืออาจจะใช้คอลัมน์มากกว่า 1 คอลัมน์รวมกันเป็น Primary Key ก็ได้ เช่น

```

Create Table Test (
Name varchar(30) ,
Lastname varchar(30) ,

```

```

Address text,
Primary Key (name, lastname)
);

```

✖ **การกำหนด Index**

อินเด็กซ์ (Index) เป็นแอทริบิวต์ที่ใช้สำหรับจัดลำดับของข้อมูล เพื่อให้การค้นหาเป็นไปได้อย่างรวดเร็ว โดยสามารถเรียงลำดับได้ทั้งแบบจากน้อยไปหามาก คือ เรียงจาก A – Z เขียนแทนด้วย ASC และเรียงลำดับจากมากไปหาน้อย คือ เรียงจาก Z – A เขียนแทนด้วย Desc สำหรับค่าปกติกำหนดไว้เป็นแบบ ASC โดยสามารถกำหนด Index ให้กับหลายคอลัมน์ได้ เช่น

```

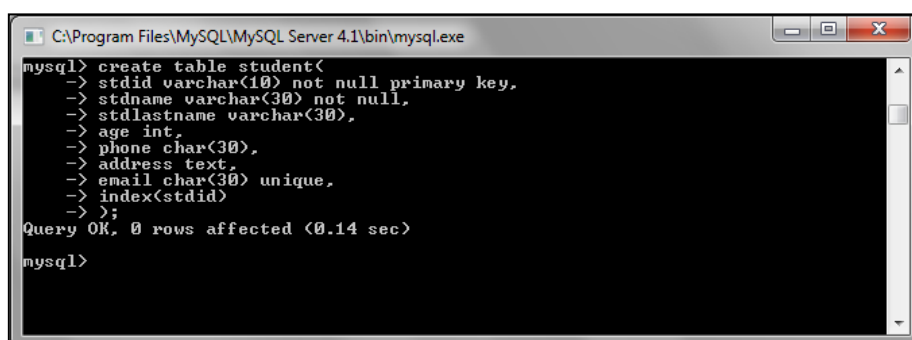
Create Table Test (
Name varchar(30) ,
Lastname varchar(30) ,
Address text,
Index (name, lastname)
);

```

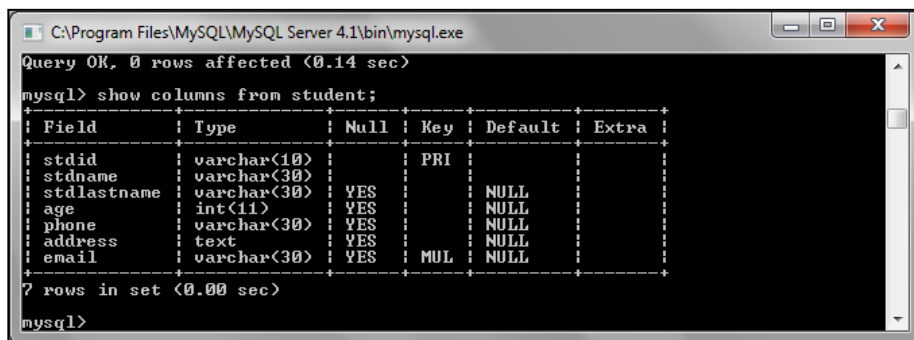
🔗 **กิจกรรมทบทวน** ให้นักศึกษาทดลองสร้างตารางนักศึกษา (Student)

ชื่อคอลัมน์	ชนิดข้อมูล	แอทริบิวต์
Stdid	Varchar(10)	Primary Key, Index
stdname	Varchar(30)	Not Null
stdlastname	Varchar(30)	
Age	Int	
Phone	Char(30)	
Address	Text	
Email	Char(30)	Unique

ถ้าพิมพ์ถูกต้องจะได้ผลลัพธ์ ดังภาพ



จากตารางข้อมูลดังกล่าว เมื่อใช้คำสั่ง Show Columns จะพบโครงสร้างตามที่กำหนด



~~ก~~ **กิจกรรมทบทวน** ให้นักศึกษาใช้คำสั่งสร้างตาราง ดังต่อไปนี้

ตาราง Teacher

ชื่อคอลัมน์	ชนิดข้อมูล	แอทริบิวต์
Tid	Int	Auto_Increment, Primary Key, Index
Tname	Varchar(35)	Not Null
Tsurname	Varchar(35)	Not Null
Age	Int	
Position	Char(30)	
Address	Text	
Email	Char(30)	Unique

ตาราง Register

ชื่อคอลัมน์	ชนิดข้อมูล	แอทริบิวต์
Register	Int	Auto_Increment, Primary Key, Not null, Index
Rdate	date	Not Null
stdid	Varchar(10)	Not Null
semester	Int	
year	Char(5)	

ตาราง Register_detail

ชื่อคอลัมน์	ชนิดข้อมูล	แอทริบิวต์
Register	Int	Not null
Subid	Varchar(10)	Not Null
grade	Varchar(5)	Not Null

กิจกรรมเสนอแนะเชิงเจตคติ

จงเติมข้อความในช่องว่างให้สมบูรณ์

1. จงบอกคำสั่งพื้นฐานของโปรแกรม MySQL

.....
.....
.....

2. จงบอกองค์ประกอบของการสร้างฐานข้อมูล

.....
.....
.....
.....

3. จงบอกลักษณะของการกำหนดประเภทของข้อมูลใน โปรแกรม MySQL

.....
.....
.....
.....

4. จงบอกขั้นตอนการสร้างฐานข้อมูล

.....
.....
.....
.....

5. จงบอกแอทริบิวต์ที่ใช้เพื่อกำหนดคุณลักษณะพิเศษเพิ่มเติมให้กับคอลัมน์

.....
.....
.....
.....

กิจกรรมเสนอแนะเชิงทักษะ

ให้นักศึกษาใช้คำสั่งบนโปรแกรม MySQL สร้างฐานข้อมูลจากตารางข้อมูลที่กำหนดให้ โดยกำหนดประเภทของข้อมูลและแอทริบิวต์ที่เกี่ยวข้องให้เหมาะสมกับลักษณะของข้อมูลที่กำหนดให้

สร้างฐานข้อมูลชื่อ BookStore

※ ตัวอย่างข้อมูล ตารางหนังสือ (Book)

Gcode	name_book	units	price	AID
1511	Word 2007	15	250	A1
1567	Excel 2007	20	300	A1
2511	ฟุตบอล	25	150	A2
2520	วอลเลย์บอล	30	100	A2
3500	อังกฤษธุรกิจ	18	350	A3
3566	PowerPoint 2007	10	260	A1

※ โครงสร้างของตาราง Book ให้นักศึกษาเป็นผู้กำหนด

ชื่อคอลัมน์	ชนิดข้อมูล	แอทริบิวต์

※ ตัวอย่างข้อมูล ตารางใบสั่งซื้อ (Order)

Orderid	Orderdate	CusID	Total
101	2010-01-10	C01	800
102	2010-01-20	C01	1,500
103	2010-01-20	C02	3,500
104	2010-01-21	C03	1,500
105	2010-02-21	C02	1,000
106	2010-02-21	C03	4,000

※ โครงสร้างของตาราง Order ให้นักศึกษาเป็นผู้กำหนด

ชื่อคอลัมน์	ชนิดข้อมูล	แอทริบิวต์

※ ตัวอย่างข้อมูล ตารางรายละเอียดการสั่งซื้อ (Order_Detail)

Orderid	Gcode	amount
101	1511	2
101	1567	1
102	1511	4
102	2520	5
103	3500	10
104	2511	10

※ โครงสร้างของตาราง Order_Detail ให้นักศึกษาเป็นผู้กำหนด

ชื่อคอลัมน์	ชนิดข้อมูล	แอทริบิวต์

※ ตัวอย่างข้อมูล ตารางผู้แต่งหนังสือ (Author)

Aid	name_sur	phone	email
A1	อ.วิษณุ เขียวงาม	0819876933	a@hotmail.com
A2	อ.นภา วีระภาพ	0895435514	b@hotmail.com
A3	อ.นवलเพ็ญ วิสัย	0870001015	c@gmail.com
A4	อ.ฟ้า แก้วงาม	0845555555	d@gmail.com
A5	อ.วิสัย กานดา	0867777777	e@hotmail.com

※ โครงสร้างของตาราง Author ให้นักศึกษาเป็นผู้กำหนด

ชื่อคอลัมน์	ชนิดข้อมูล	แอทริบิวต์

บทที่ 3

การแก้ไขโครงสร้างตาราง

สาระการเรียนรู้

1. หลักการเปลี่ยนแปลงโครงสร้างฐานข้อมูล
2. คำสั่งเปลี่ยนแปลงโครงสร้างตาราง
 - 2.1. คำสั่งลบฐานข้อมูล
 - 2.2. คำสั่งลบตารางฐานข้อมูล
 - 2.3. คำสั่งเปลี่ยนชื่อตาราง
 - 2.4. คำสั่งเพิ่มคอลัมน์ใหม่ให้กับตาราง
 - 2.5. คำสั่งเปลี่ยนชื่อคอลัมน์และชนิดข้อมูลของคอลัมน์
 - 2.6. การสร้างและการลบ Primary Key
 - 2.7. การเพิ่ม Index
 - 2.8. คำสั่งเพิ่ม Unique
 - 2.9. คำสั่งลบคอลัมน์

มาตรฐานเชิงสมรรถนะ

1. สามารถปฏิบัติการใช้คำสั่ง MySQL ในการเปลี่ยนแปลงโครงสร้างฐานข้อมูลได้
2. สามารถวิเคราะห์ และเปรียบเทียบผลดี ผลเสียในการใช้คำสั่งเปลี่ยนแปลงโครงสร้างฐานข้อมูลได้
3. สามารถอธิบายข้อควรระมัดระวังในการใช้คำสั่งแต่ละประเภทได้

กิจกรรมเสนอแนะ

1. ให้นักศึกษาฝึกปฏิบัติการใช้คำสั่งเปลี่ยนแปลงโครงสร้างฐานข้อมูล
2. ให้นักศึกษาทำรายงานพิเศษวิเคราะห์ผลดี ผลเสีย และข้อควรระวังในการใช้คำสั่งแต่ละประเภท

การวัดผล

1. การตอบข้อซักถามในชั้นเรียน
2. การสังเกต
3. ผลการฝึกปฏิบัติ

บทที่ 3 การแก้ไขโครงสร้างตาราง

การปรับปรุงเปลี่ยนแปลงโครงสร้างของตารางในภายหลัง มักจะเกิดอยู่บ่อยครั้ง เนื่องจากระบบมีการเปลี่ยนแปลงให้ตอบสนองต่อความต้องการสารสนเทศ ณ ขณะนั้นให้ทันสมัยหรือสมบูรณ์มากขึ้น ดังนั้นจึงมีการนิยามคำสั่งประเภทแก้ไขโครงสร้างตารางข้อมูลไว้ เพื่อให้เกิดความยืดหยุ่นของฐานข้อมูลทั้งในปัจจุบัน และอนาคต ทั้งนี้แต่ละคำสั่งที่เกี่ยวข้องกับการแก้ไขโครงสร้างตารางจะต้องเรียกใช้งานด้วยความระมัดระวัง และมีการวางแผนงานที่ดี เพราะหากเรียกใช้คำสั่งหรือเปลี่ยนแปลงโครงสร้างข้อมูลผิดพลาด อาจส่งผลเสียต่อข้อมูลที่มีอยู่ในตารางข้อมูลนั้น ๆ ทันที

✖ คำสั่งลบฐานข้อมูล

เป็นคำสั่งลบฐานข้อมูล การใช้คำสั่งดังกล่าวจะส่งผลให้ตารางข้อมูลทั้งหมดที่มีอยู่ในฐานข้อมูลถูกลบไปด้วย

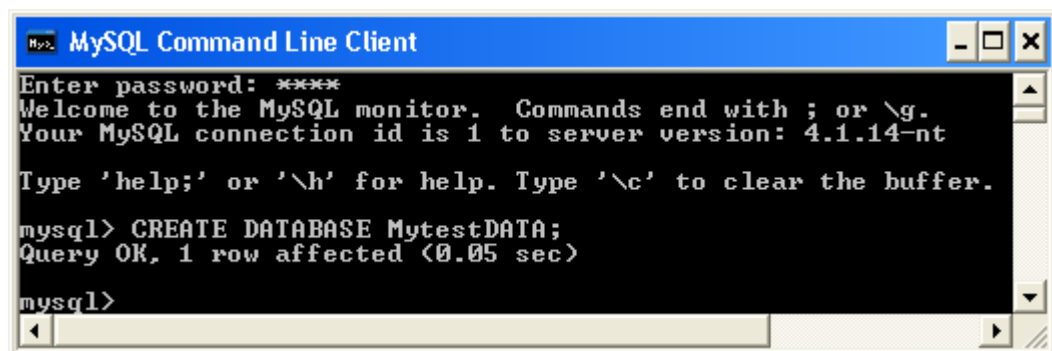
รูปแบบ

```
DROP DATABASE ชื่อฐานข้อมูล ;
```

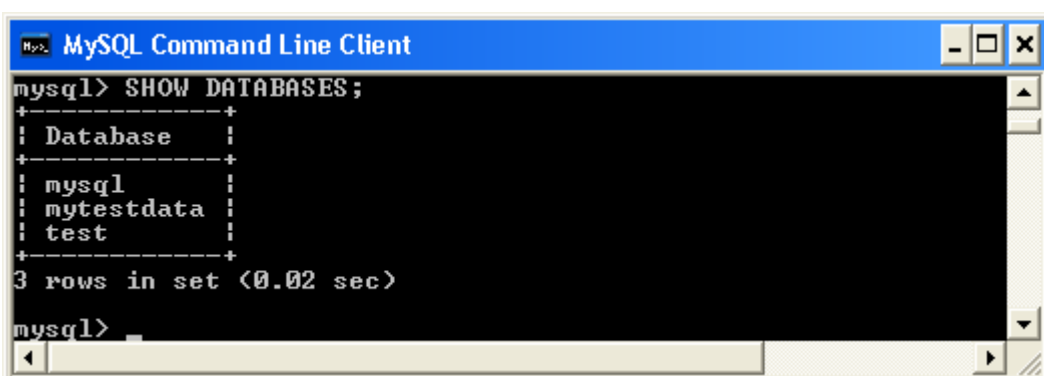
เช่น

1. ใช้คำสั่งสร้างฐานข้อมูล ชื่อ MytestData

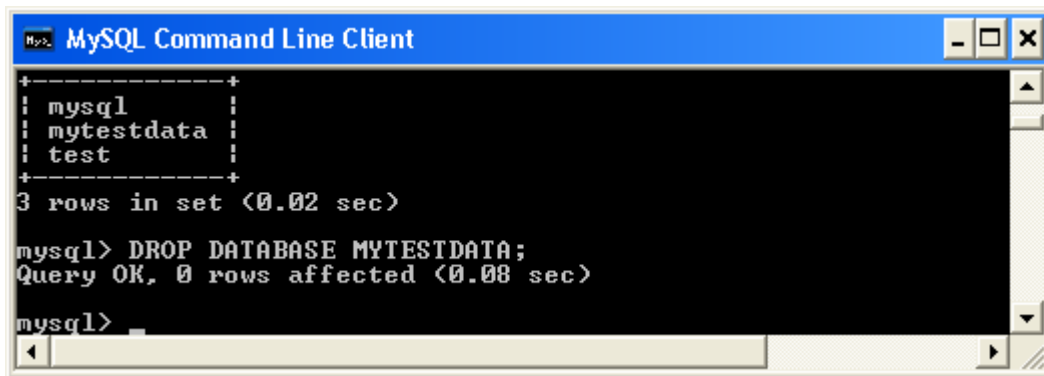
```
Create Database ชื่อฐานข้อมูล ;
```



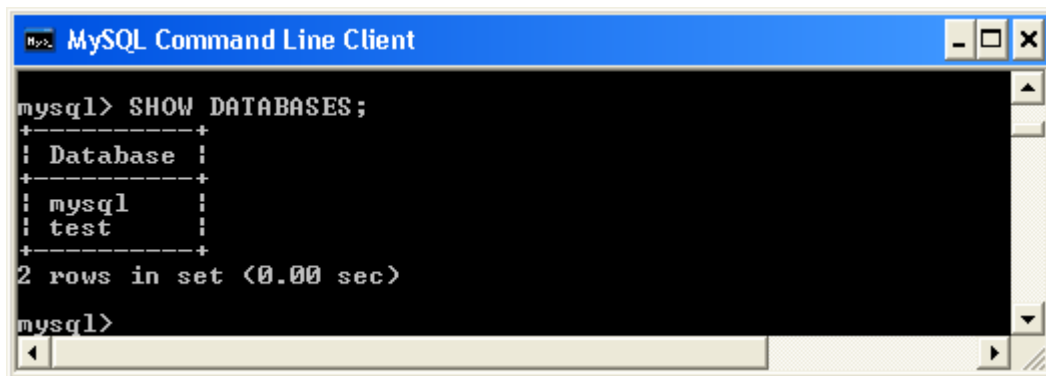
2. ใช้คำสั่งแสดงรายชื่อฐานข้อมูล จะปรากฏรายชื่อ mytestdata ที่เราสร้างขึ้น



3. ใช้คำสั่งลบฐานข้อมูล DROP DATABASE MytestData ;



ใช้คำสั่ง Show Databases อีกครั้งจะพบว่า ฐานข้อมูล MytestData ถูกลบทิ้งออกไปเรียบร้อยแล้ว ดังนั้นการใช้คำสั่งดังกล่าวจึงต้องใช้ความระมัดระวังเป็นพิเศษ



※ คำสั่งลบตารางฐานข้อมูล

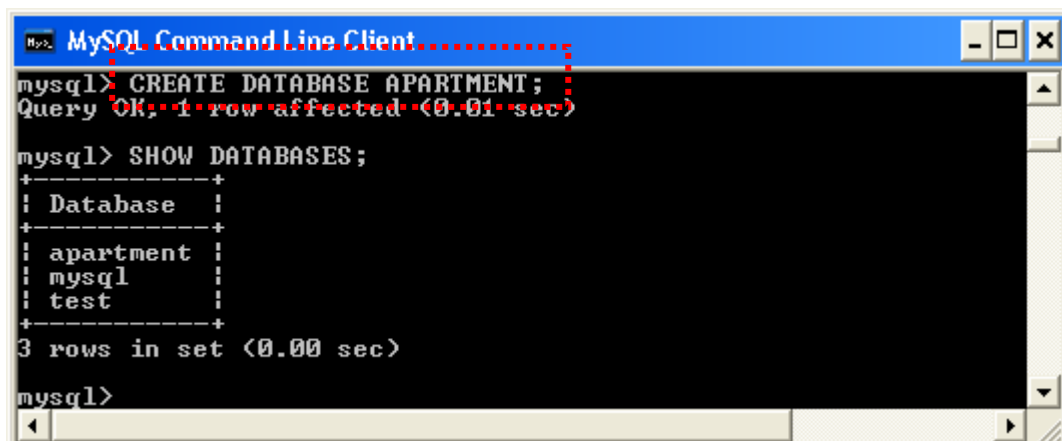
เป็นคำสั่งลบตารางในฐานข้อมูล การใช้คำสั่งดังกล่าวจะทำให้ตารางนั้น ๆ ถูกลบทิ้งออกจากฐานข้อมูล ส่งผลให้ข้อมูลของตารางนั้น ๆ ถูกลบทิ้งไปด้วย มีรูปแบบคำสั่งดังนี้

รูปแบบ

DROP TABLE ชื่อตาราง ;

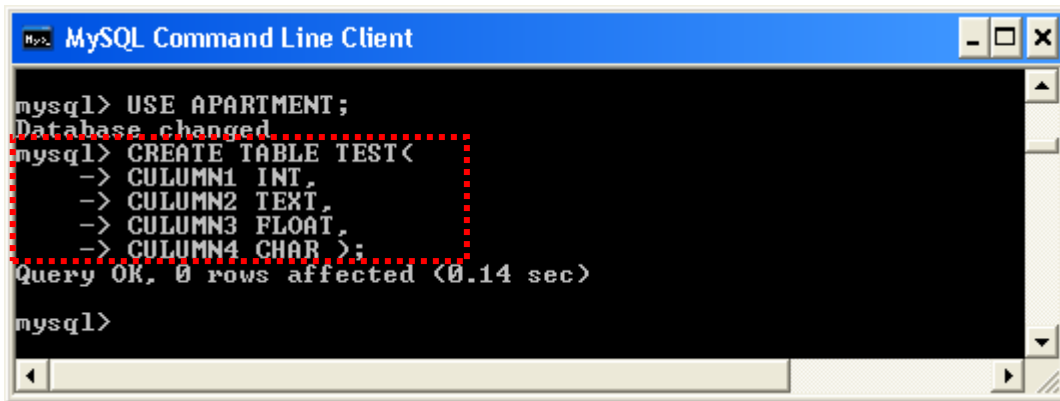
เช่น

1. ใช้คำสั่งสร้างฐานข้อมูลชื่อ Apartment



2. ใช้คำสั่งใช้งานฐานข้อมูล Apartment >> USE APARTMENT ;
3. ใช้คำสั่งสร้างตารางชื่อ Test และตารางชื่อ Room ดังนี้

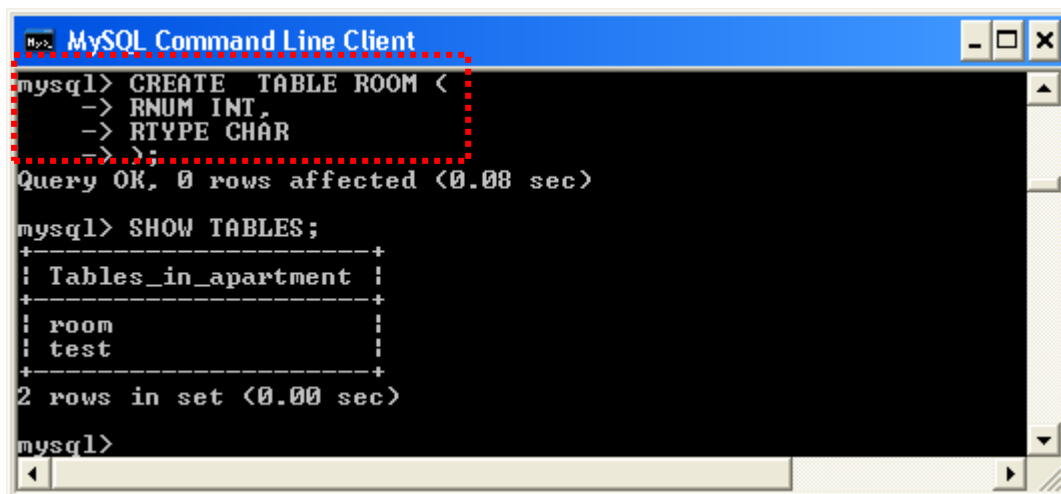
คำสั่งสร้างตาราง Test



```
mysql> USE APARTMENT;
Database changed
mysql> CREATE TABLE TEST (
  -> COLUMN1 INT,
  -> COLUMN2 TEXT,
  -> COLUMN3 FLOAT,
  -> COLUMN4 CHAR );
Query OK, 0 rows affected (0.14 sec)

mysql>
```

คำสั่งสร้างตาราง Room



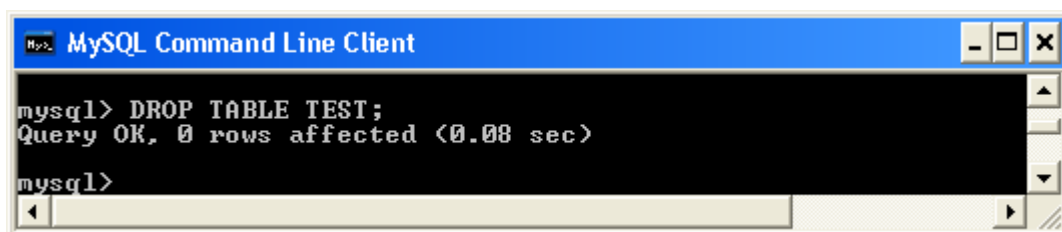
```
mysql> CREATE TABLE ROOM (
  -> RNUM INT,
  -> RTYPE CHAR
  -> );
Query OK, 0 rows affected (0.08 sec)

mysql> SHOW TABLES;
+-----+
| Tables_in_apartment |
+-----+
| room                 |
| test                 |
+-----+
2 rows in set (0.00 sec)

mysql>
```

หลังจากที่สร้างตารางทั้ง 2 แล้ว ทดสอบใช้คำสั่ง Show Table จะปรากฏรายชื่อตารางที่สร้างขึ้นใหม่ ดังภาพข้างต้น

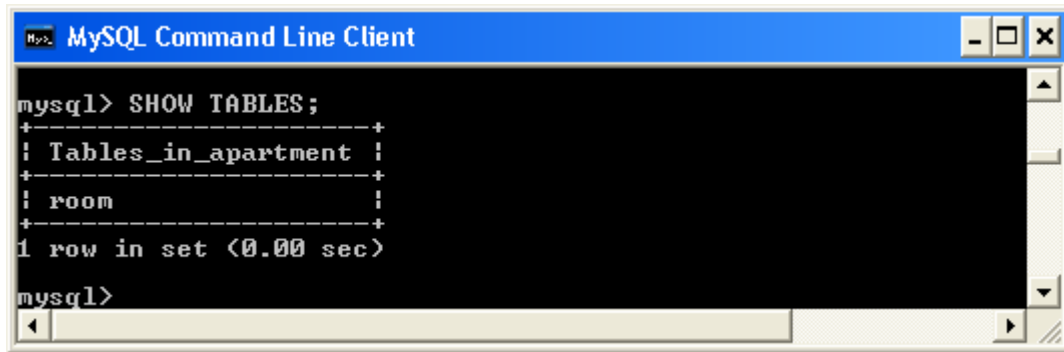
4. ใช้คำสั่งลบตารางข้อมูล Test >> Drop Table Test ;



```
mysql> DROP TABLE TEST;
Query OK, 0 rows affected (0.08 sec)

mysql>
```

ทดลองใช้คำสั่งแสดงตารางอีกครั้ง จะพบว่าตารางชื่อ Test ถูกลบออกจากรายชื่อข้อมูลไปเรียบร้อยแล้ว



※ คำสั่งเปลี่ยนชื่อตาราง

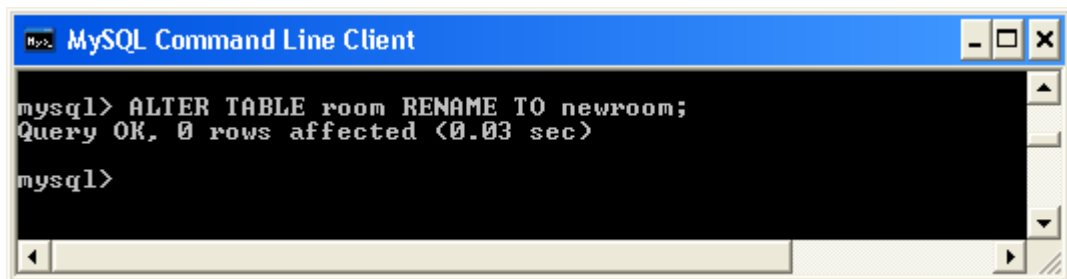
เป็นคำสั่งเปลี่ยนชื่อของตารางจากชื่อหนึ่ง เป็นอีกชื่อหนึ่ง มีรูปแบบคำสั่งดังนี้

รูปแบบ

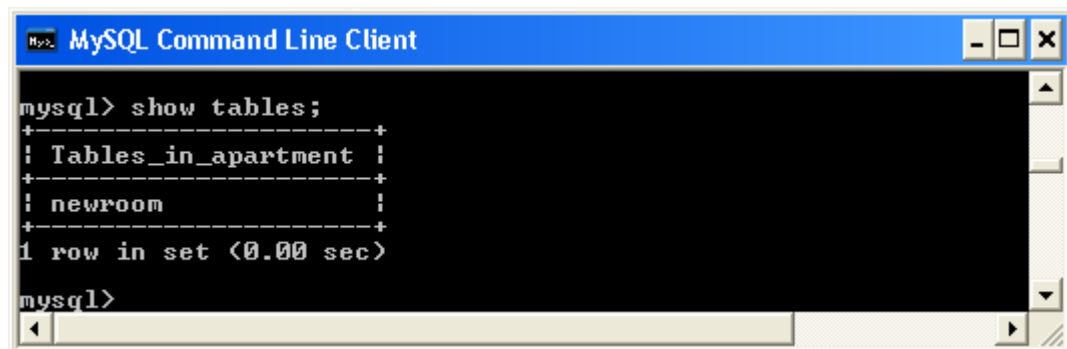
```
ALTER TABLE ชื่อตารางเดิม RENAME TO ชื่อตารางใหม่ ;
```

เช่น

1. ใช้คำสั่งเปลี่ยนชื่อตาราง Room เป็น NewRoom



ใช้คำสั่งแสดงรายชื่อตารางจะพบว่า ชื่อตารางได้เปลี่ยนไปเป็นชื่อใหม่เรียบร้อยแล้ว



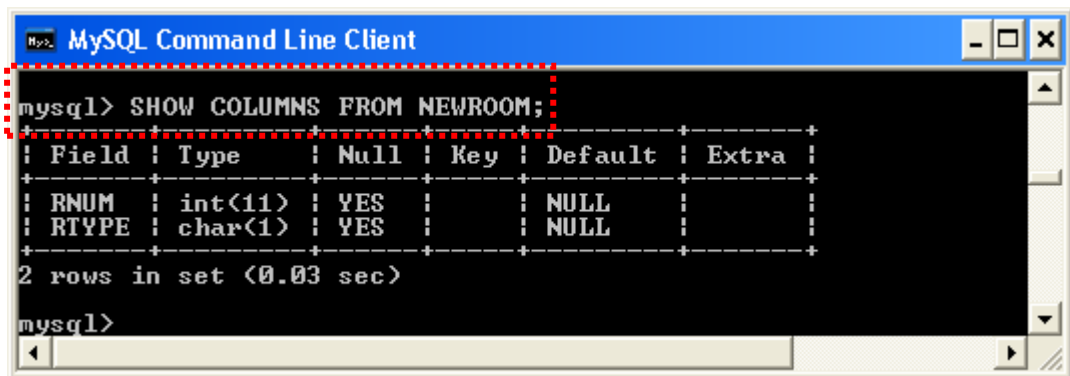
✳ คำสั่งเพิ่มคอลัมน์ใหม่ให้กับตาราง

รูปแบบ

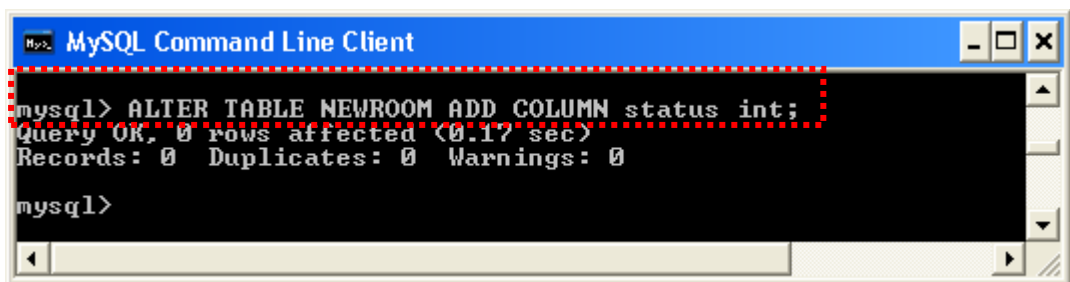
```
ALTER TABLE ชื่อตาราง ADD COLUMN ชื่อคอลัมน์ ชนิดข้อมูล [แตริวิวด์]
[FIRST | AFTER] คอลัมน์อ้างอิง ;
```

เช่น

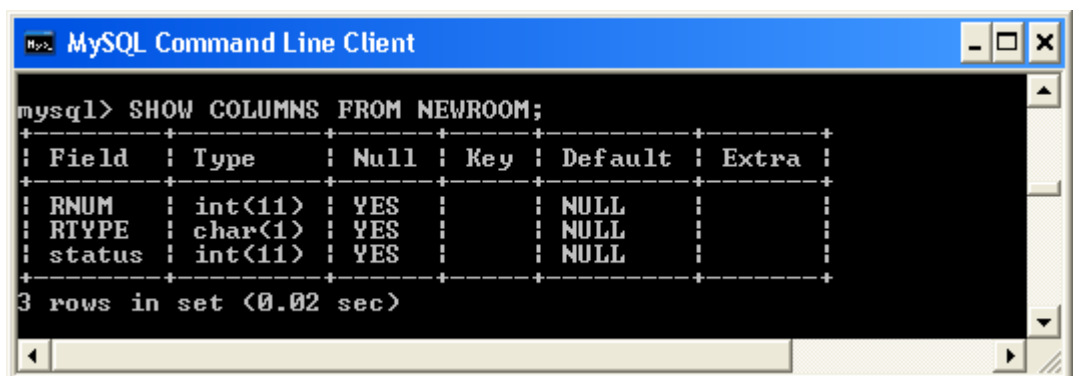
1. ใช้คำสั่งแสดงคอลัมน์ของตาราง NewRoom >> SHOW COLUMNS FROM NEWROOM;



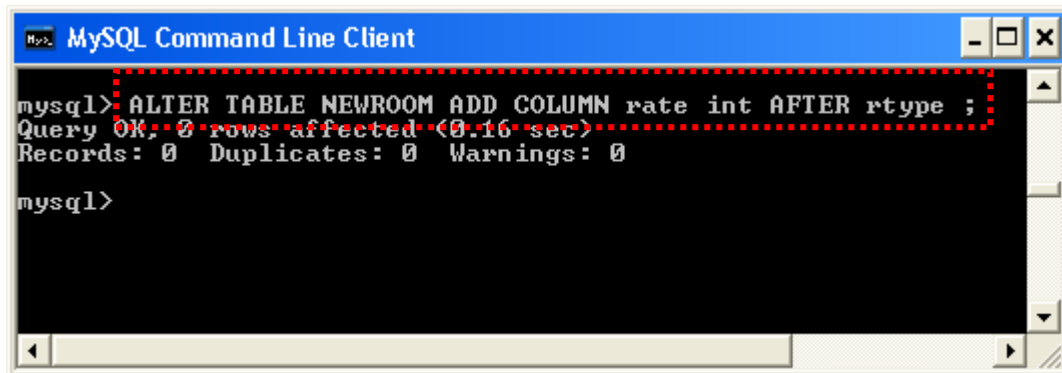
2. ใช้คำสั่งสร้างคอลัมน์ Status กำหนดชนิดข้อมูลเป็นแบบ Int



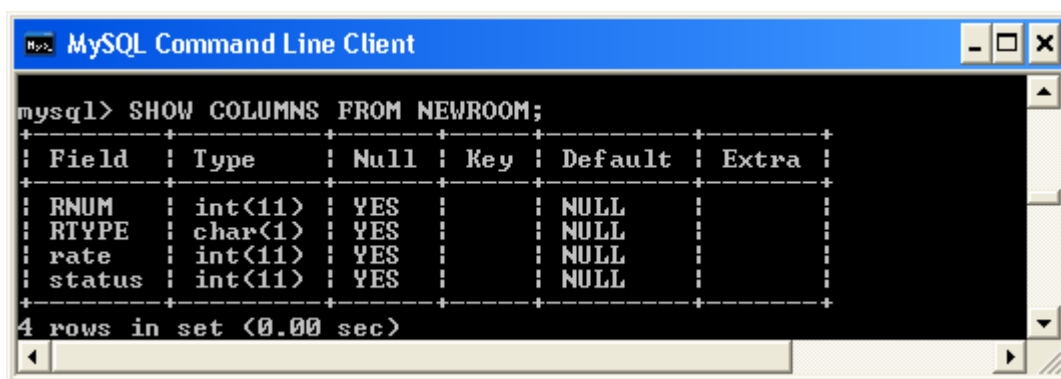
ใช้คำสั่งแสดงรายชื่อคอลัมน์อีกครั้ง จะพบว่ามียคอลัมน์ Status เพิ่มขึ้นมามาก 1 คอลัมน์



- 3. ใช้คำสั่งสร้างคอลัมน์ Rate กำหนดชนิดข้อมูลเป็นแบบ Int และวางต่อท้ายคอลัมน์ Rtype



ใช้คำสั่งแสดงรายชื่อคอลัมน์อีกครั้ง จะพบว่าคอลัมน์ Rate ได้ถูกสร้างขึ้นและจัดวางลำดับตามที่โจทย์กำหนดเรียบร้อยแล้ว



※ คำสั่งเปลี่ยนชื่อคอลัมน์และชนิดข้อมูลของคอลัมน์

การใช้คำสั่งเปลี่ยนชนิดของข้อมูล โดยเฉพาะการเปลี่ยนประเภทของข้อมูล เช่นจากเดิมเป็นตัวเลข ต่อมาเปลี่ยนเป็นข้อความ หรือการเปลี่ยนแปลงขนาดของข้อมูล เช่นจากเดิมกำหนดเป็นตัวอักษรขนาด 50 ต่อมาลดลงเหลือ 20 อาจส่งผลเสียหายกับข้อมูลที่มีอยู่ในตารางได้ ดังนั้นการเปลี่ยนแปลงชนิดและขนาดของข้อมูลจึงเหมาะกับการดำเนินการตั้งแต่แรก ก่อนที่จะมีการรับข้อมูลเข้าไปเก็บไว้ในตารางนั้น ๆ

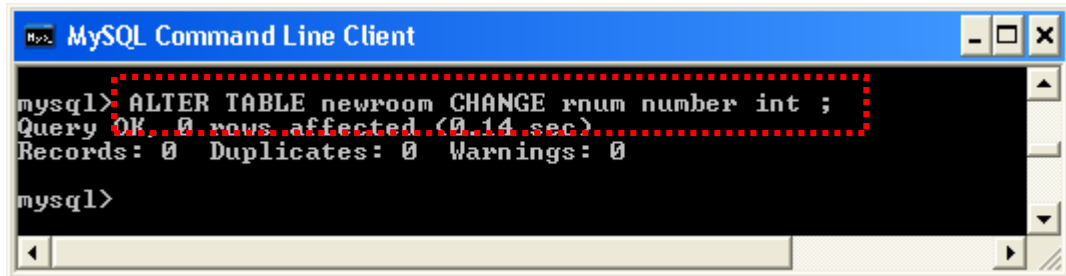
รูปแบบ

```
ALTER TABLE ชื่อตาราง CHANGE ชื่อคอลัมน์เดิม ชื่อคอลัมน์ใหม่ ชนิดข้อมูล [เอทริบิวต์];
```

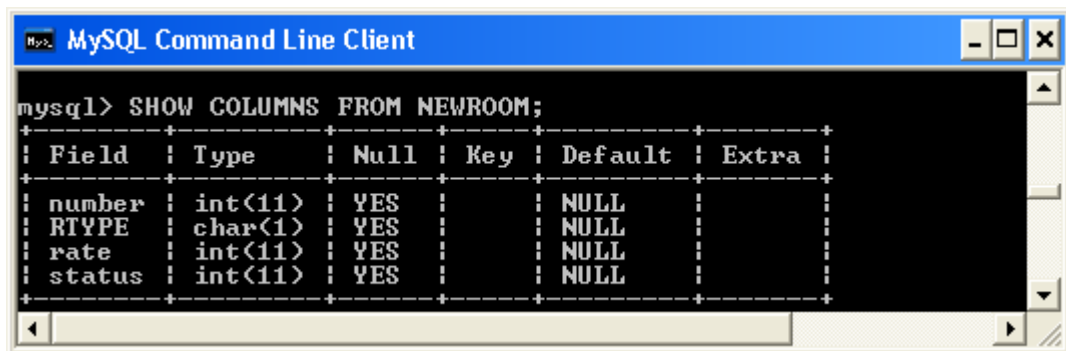
เช่น

1. สมมติต้องการเปลี่ยนชื่อคอลัมน์ Rnum เป็น Number และชนิดข้อมูลยังเหมือนเดิม

>> ALTER TABLE newroom CHANGE rnum number int;

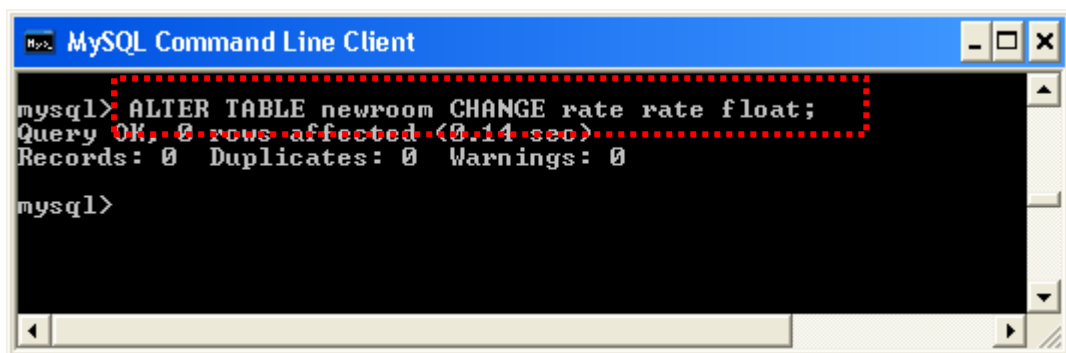


```
mysql> ALTER TABLE newroom CHANGE rnum number int ;
Query OK, 0 rows affected (0.14 sec)
Records: 0 Duplicates: 0 Warnings: 0
mysql>
```

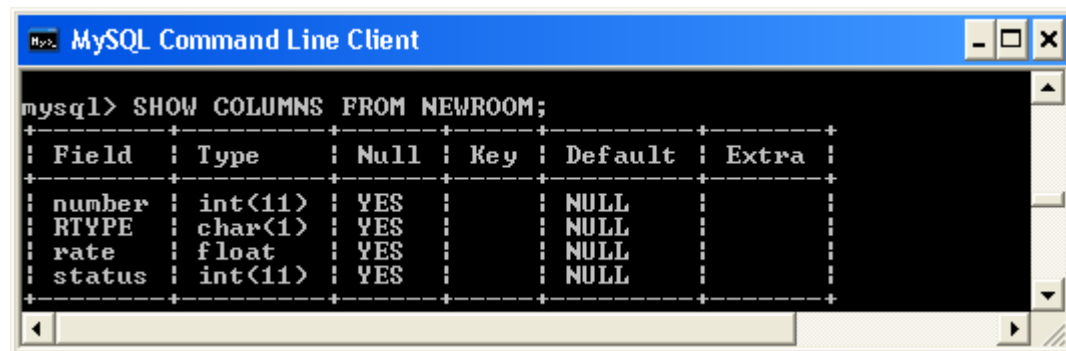


```
mysql> SHOW COLUMNS FROM NEWROOM;
+-----+-----+-----+-----+-----+-----+
| Field | Type  | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| number | int(11) | YES  |     | NULL    |       |
| RTYPE  | char(1) | YES  |     | NULL    |       |
| rate   | int(11) | YES  |     | NULL    |       |
| status | int(11) | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
```

2. ใช้คำสั่งเปลี่ยนชนิดข้อมูลโดยใช้ชื่อคอลัมน์เดิม



```
mysql> ALTER TABLE newroom CHANGE rate rate float;
Query OK, 0 rows affected (0.14 sec)
Records: 0 Duplicates: 0 Warnings: 0
mysql>
```



```
mysql> SHOW COLUMNS FROM NEWROOM;
+-----+-----+-----+-----+-----+-----+
| Field | Type  | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| number | int(11) | YES  |     | NULL    |       |
| RTYPE  | char(1) | YES  |     | NULL    |       |
| rate   | float  | YES  |     | NULL    |       |
| status | int(11) | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
```

3. ใช้คำสั่งเปลี่ยนทั้งชื่อคอลัมน์ และชนิดข้อมูล

```
mysql> ALTER TABLE newroom CHANGE rate price double not null;
Query OK, 0 rows affected (0.14 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql>
```

```
mysql> SHOW COLUMNS FROM NEWROOM;
+----+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra |
+----+-----+-----+-----+-----+-----+
| number | int(11)   | YES  |     | NULL    |       |
| RTYPE  | char(1)   | YES  |     | NULL    |       |
| price  | double    |      |     | 0       |       |
| status | int(11)   | YES  |     | NULL    |       |
+----+-----+-----+-----+-----+-----+
```

✳ การสร้างและการลบ Primary Key

รูปแบบ คำสั่งการสร้าง Primary Key

```
ALTER TABLE ชื่อตาราง ADD PRIMARY KEY (ชื่อคอลัมน์1, ชื่อคอลัมน์2,...) ;
```

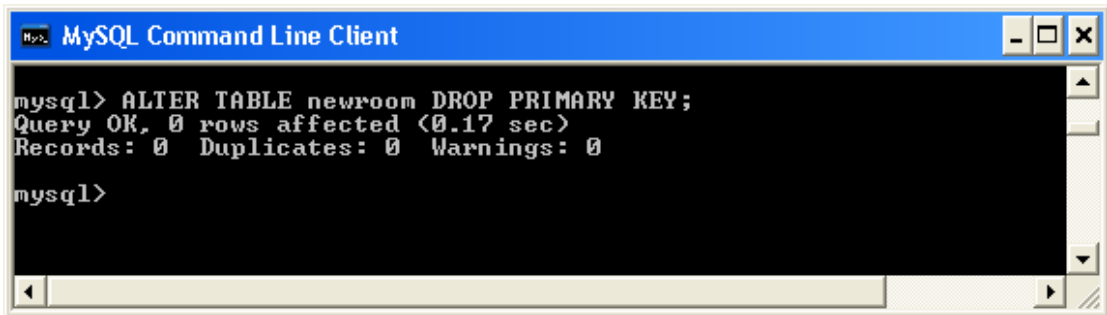
```
mysql> ALTER TABLE newroom ADD PRIMARY KEY <NUMBER>;
Query OK, 0 rows affected (0.14 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql>
```

```
mysql> show columns from newroom;
+----+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra |
+----+-----+-----+-----+-----+-----+
| number | int(11)   | YES  | PRI | 0       |       |
| RTYPE  | char(1)   | YES  |     | NULL    |       |
| price  | double    |      |     | 0       |       |
| status | int(11)   | YES  |     | NULL    |       |
+----+-----+-----+-----+-----+-----+
```

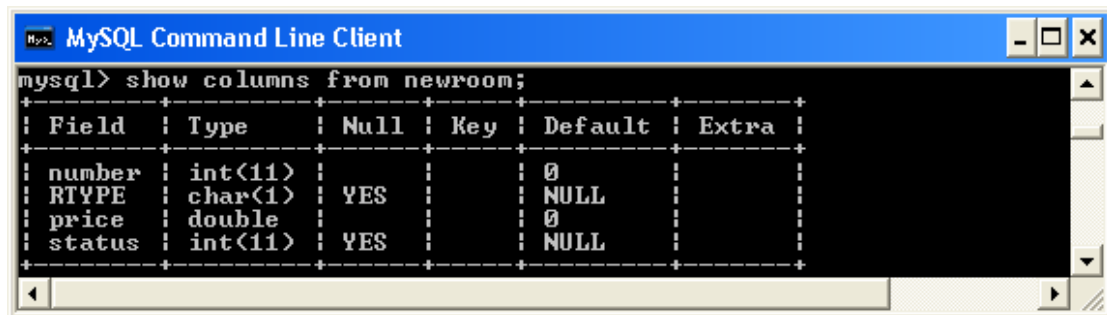
รูปแบบ คำสั่งยกเลิกการสร้าง Primary Key

ALTER TABLE ชื่อตาราง DROP PRIMARY KEY ;



```
mysql> ALTER TABLE newroom DROP PRIMARY KEY;
Query OK, 0 rows affected (0.17 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql>
```

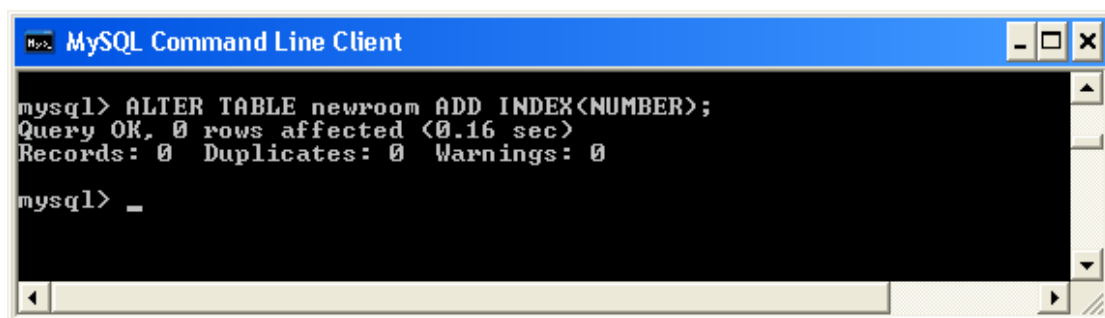


```
mysql> show columns from newroom;
+----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+----+-----+-----+-----+-----+-----+
| number | int(11) | YES |  | 0 |  |
| RTYPE | char(1) | YES |  | NULL |  |
| price | double |  |  | 0 |  |
| status | int(11) | YES |  | NULL |  |
+----+-----+-----+-----+-----+-----+
```

※ การเพิ่ม INDEX

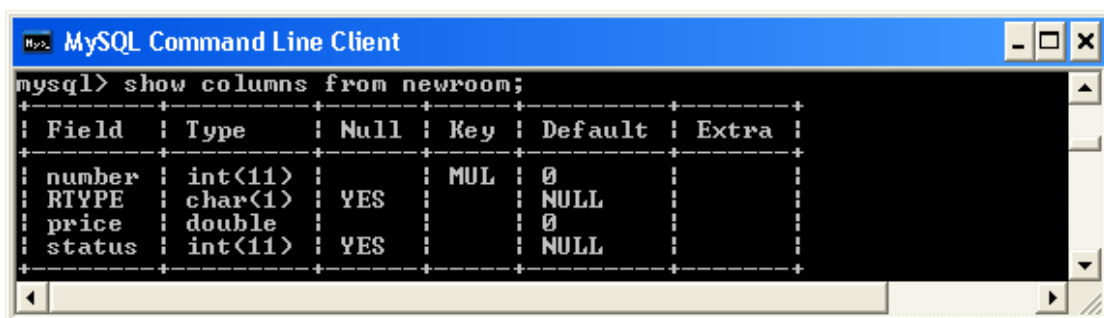
รูปแบบ

ALTER TABLE ชื่อตาราง ADD INDEX (ชื่อคอลัมน์1, ชื่อคอลัมน์2,...) ;



```
mysql> ALTER TABLE newroom ADD INDEX(NUMBER);
Query OK, 0 rows affected (0.16 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> _
```

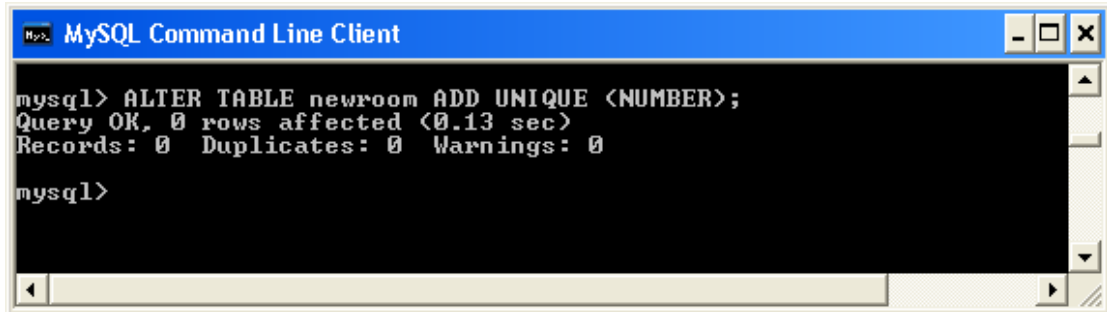


```
mysql> show columns from newroom;
+----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+----+-----+-----+-----+-----+-----+
| number | int(11) | YES | MUL | 0 |  |
| RTYPE | char(1) | YES |  | NULL |  |
| price | double |  |  | 0 |  |
| status | int(11) | YES |  | NULL |  |
+----+-----+-----+-----+-----+-----+
```

※ การเพิ่ม UNIQUE

รูปแบบ

```
ALTER TABLE ชื่อตาราง ADD UNIQUE (ชื่อคอลัมน์1,ชื่อคอลัมน์2,...) ;
```

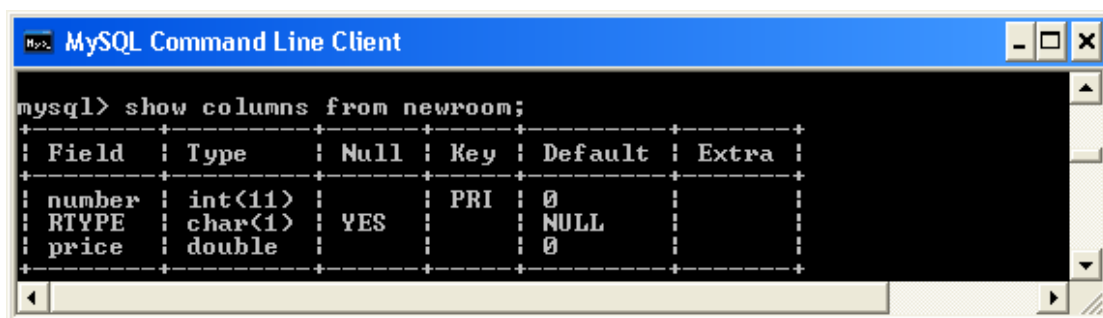
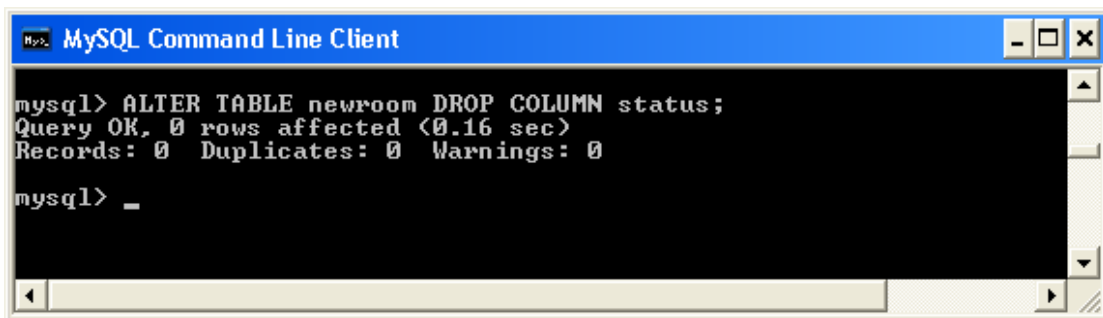


※ คำสั่งการลบคอลัมน์

รูปแบบ

```
ALTER TABLE ชื่อตาราง DROP COLUMN ชื่อคอลัมน์ ;
```

เช่น ใช้คำสั่งลบคอลัมน์ Status ออกจากตาราง Newroom >> ALTER TABLE newroom DROP COLUMN status ;



กิจกรรมแนะเชิงเจตคติ

จงเติมข้อความในช่องว่างให้สมบูรณ์

1. จงบอกขั้นตอนการใช้คำสั่งลบฐานข้อมูล

.....
.....
.....

2. จงบอกขั้นตอนการใช้คำสั่งลบตารางฐานข้อมูล

.....
.....
.....
.....

3. จงบอกขั้นตอนการใช้คำสั่งลบตารางฐานข้อมูล

.....
.....
.....
.....

4. จงบอกขั้นตอนการใช้คำสั่งเพิ่มคอลัมน์ใหม่ให้กับตาราง

.....
.....
.....
.....

5. จงบอกขั้นตอนการใช้คำสั่งเปลี่ยนชื่อคอลัมน์และชนิดข้อมูลของคอลัมน์

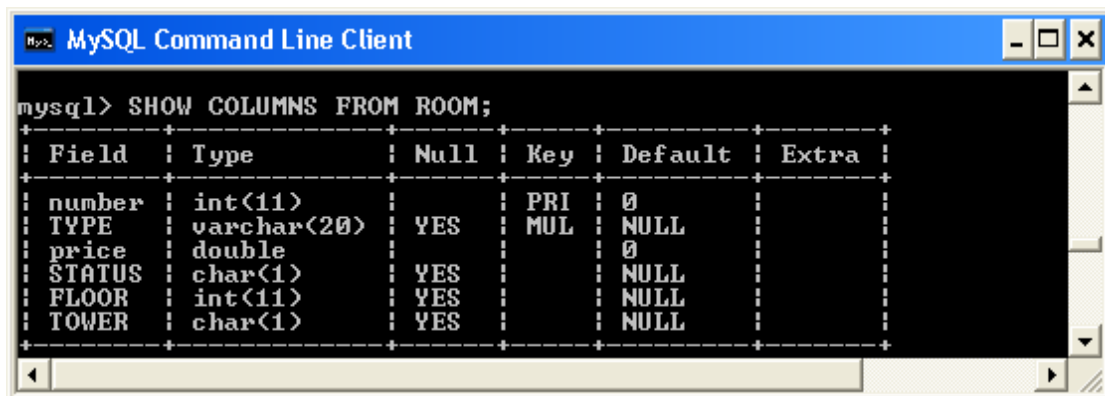
.....
.....
.....
.....

กิจกรรมเสนอแนะเชิงทักษะ

จากตารางข้อมูล Newroom ให้นักศึกษาเปลี่ยนแปลงและแก้ไขโครงสร้างของตาราง ตามคำสั่งดังต่อไปนี้

1. เปลี่ยนชื่อตาราง “Newroom” เป็น “Room”
2. เพิ่มคอลัมน์ “Status” ในตาราง Room กำหนดชนิดข้อมูลเป็นแบบ VARCHAR (1)
3. เพิ่มคอลัมน์ “Floor” ในตาราง Room กำหนดชนิดข้อมูลเป็นแบบ INT
4. เพิ่มคอลัมน์ “Tower” ในตาราง Room กำหนดชนิดข้อมูลเป็นแบบ VARCHAR (1)
5. เปลี่ยนชื่อคอลัมน์ “RTYPE” เป็น “TYPE” กำหนดชนิดข้อมูลเป็น VARCHAR(20)
6. กำหนด INDEX ให้กับคอลัมน์ TYPE
7. กำหนด PRIMARY KEY ให้กับคอลัมน์ “ROOM”

ผลลัพธ์ที่ได้ ภายหลังใช้คำสั่งแสดงชื่อคอลัมน์จากตาราง ROOM



```
mysql> SHOW COLUMNS FROM ROOM;
```

Field	Type	Null	Key	Default	Extra
number	int(11)		PRI	0	
TYPE	varchar(20)	YES	MUL	NULL	
price	double			0	
STATUS	char(1)	YES		NULL	
FLOOR	int(11)	YES		NULL	
TOWER	char(1)	YES		NULL	

บทที่ 4

การนำข้อมูลใส่ในตาราง

สาระการเรียนรู้

1. ข้อควรระมัดระวังในการนำข้อมูลเข้าไปเก็บในตารางฐานข้อมูล
2. คำสั่งการนำข้อมูลใส่ในตาราง
 - 2.1. คำสั่งการเพิ่มข้อมูลใหม่
 - 2.2. คำสั่งเพิ่มข้อมูลทุกคอลัมน์
 - 2.3. การปรับปรุงข้อมูลในตาราง
 - 2.4. การลบข้อมูลในตาราง

มาตรฐานเชิงสมรรถนะ

1. สามารถอธิบายข้อควรระมัดระวังในการนำข้อมูลเข้าตารางฐานข้อมูลได้
2. สามารถปฏิบัติการใช้คำสั่ง MySQL ในการนำข้อมูลเข้าตารางได้
3. สามารถวิเคราะห์ เปรียบเทียบ และเลือกใช้คำสั่งที่เหมาะสมได้
4. สามารถแก้ไขปัญหาข้อผิดพลาดที่เกิดขึ้นจากการใช้คำสั่ง MySQL ได้

กิจกรรมเสนอแนะ

1. ให้นักศึกษาฝึกปฏิบัติการนำข้อมูลเข้าสู่ตาราง การแก้ไข ปรับปรุงตาราง
2. ให้นักศึกษาฝึกปฏิบัติทบทวนคำสั่งเพิ่มเติมจากใบงานที่กำหนดให้

การวัดผล

1. การตอบข้อซักถามในชั้นเรียน
2. การสังเกต
3. ผลการฝึกปฏิบัติ
4. แบบฝึกหัดท้ายบท

บทที่ 4

การนำข้อมูลใส่ในตาราง

หลังจากที่มีการกำหนดโครงสร้างของตารางเป็นที่เรียบร้อยแล้ว และตรวจสอบแล้วว่าถูกต้อง สมบูรณ์ สามารถรองรับข้อมูลที่จะนำเข้ามาเก็บไว้ได้อย่างถูกต้องเหมาะสมแล้วนั้น ขั้นตอนต่อไป ก็จะเป็นการใช้คำสั่งเพื่อนำข้อมูลเข้าไปเก็บไว้ในตาราง โดยจะต้องคำนึงถึงสิ่งที่สำคัญ ดังนี้

1. ต้องใส่ข้อมูลให้ตรงกับชนิดข้อมูลของแต่ละคอลัมน์ที่กำหนดไว้
2. ต้องวางลำดับของข้อมูลให้ตรงกับลำดับของคอลัมน์ในตาราง
3. ต้องใส่ข้อมูลให้ครบทุกคอลัมน์ที่กำหนด
4. ต้องพิมพ์ข้อมูลไว้ในเครื่องหมาย ‘ ’ หรือ ‘”’ เสมอ ยกเว้นข้อมูลชนิดตัวเลข
5. ขนาดของข้อมูลต้องไม่เกินขนาดที่กำหนดไว้ในโครงสร้างตารางตั้งแต่ตอนแรก ทั้งนี้ หากพิมพ์เกินจำนวนที่กำหนดไว้ ข้อมูลบางส่วนอาจสูญหาย โดย MySQL จัดตัดข้อมูล ส่วนที่เกินจากที่กำหนดทิ้งไป และไม่แสดงข้อผิดพลาดให้ทราบ)
6. คอลัมน์ที่กำหนดแอทริบิวต์เป็นแบบ AUTO_INCREMENT ให้ใส่เป็นสตริงว่าง หรือไม่ ก็เลข 0 แทน เช่น สมมติ คอลัมน์แรก เป็น AUTO_INCREMENT ให้พิมพ์ดังนี้

```
VALUE (‘’, ‘AIR’,1000) ; หรือ VALUE (0, ‘AIR’,1000) ;
```

แต่ถ้าต้องการให้เริ่มนับจากตัวเลขที่ต้องการ เช่น เริ่มจาก 100 เป็นต้นไป ในการ INSERT แถวแรกให้พิมพ์ว่า VALUE (100, ‘AIR’,1000) ; แล้วแถวต่อไปให้พิมพ์ค่า 0 ตามปกติ VALUE (0, ‘AIR’,2000) ; ซึ่งค่าที่จะได้ก็คือ 100, 101, 102, ตามลำดับ

7. คอลัมน์ที่มีค่าดีฟอลต์ ถ้าไม่ต้องการใช้ค่าดีฟอลต์นั้น ก็ให้ระบุค่าที่ต้องการเหมือนข้อมูลทั่วไป แต่ถ้าต้องการใช้ค่าของดีฟอลต์ที่ตั้งไว้ ให้พิมพ์ดังนี้

```
DEFAULT(ชื่อคอลัมน์)
```

เช่น สมมติต้องการกำหนดค่าดีฟอลต์ให้กับคอลัมน์ STATUS ให้พิมพ์คำสั่ง ดังนี้

```
VALUE (0, ‘AIR’,1000, DEFAULT(status) ) ;
```

8. ข้อมูลประเภทวันเวลา จะต้องระบุเป็นสตริงในรูปแบบ ‘Year-Month-Date’ โดย Year ควรระบุเป็นปี ค.ศ. แบบเลข 4 หลัก เช่น 2010, 2012

Month ให้ระบุเป็นตัวเลขลำดับของเดือน เช่น 01, 02, 03, ..., 12

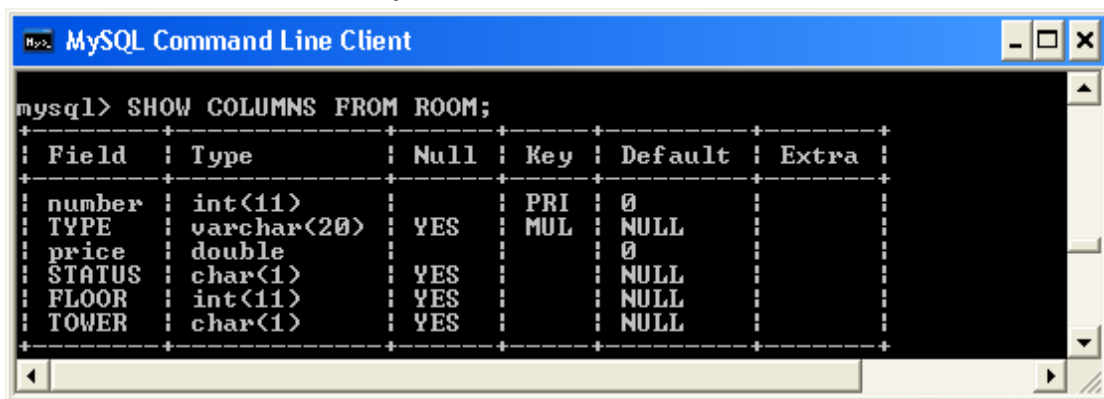
Date ให้ระบุเป็นวันที่ เช่น 01,02,03,...,31

✳ คำสั่งการเพิ่มข้อมูลใหม่

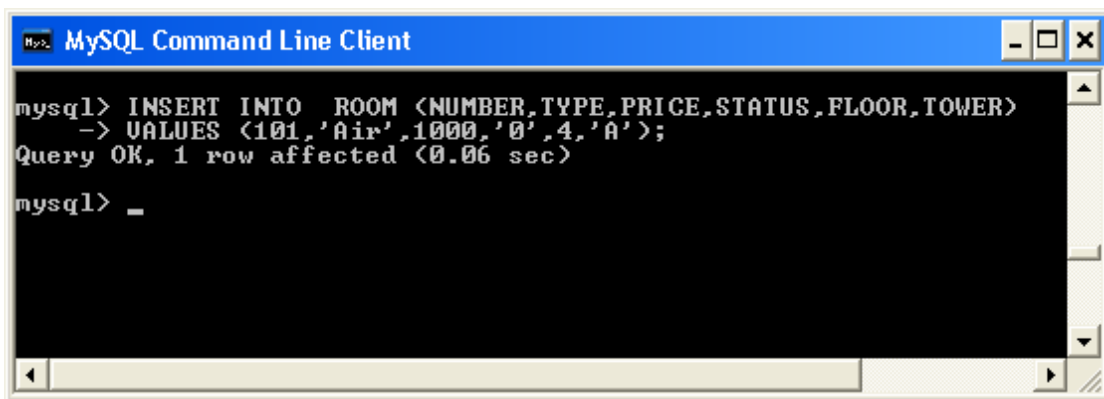
รูปแบบ

```
INSERT INTO ชื่อตาราง (ชื่อคอลัมน์1, ชื่อคอลัมน์2, ..., ชื่อคอลัมน์N)
VALUE (ค่าข้อมูล 1, ค่าข้อมูล 2, ..., ค่าข้อมูล N);
```

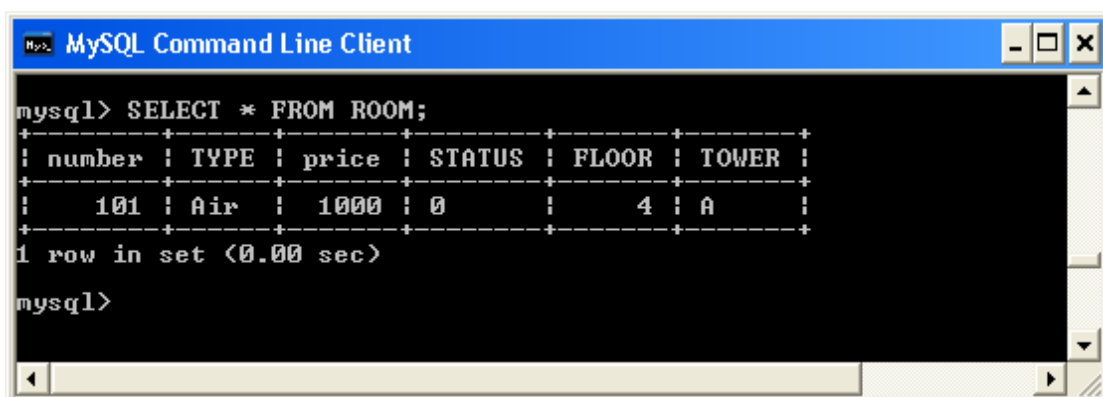
เช่น จากตารางโครงสร้างข้อมูล



```
INSERT INTO Room (number, type, price, status, floor, tower)
VALUE (101, 'Air', 1000, '0', 4, 'A');
```

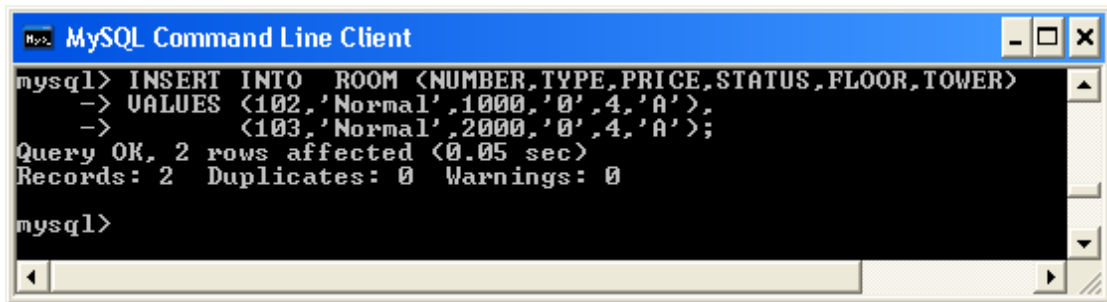


ใช้คำสั่งเรียกดูข้อมูลอย่างง่าย > `Select * From Room ;` ; จะปรากฏรายการข้อมูลที่คีย์เข้าไป

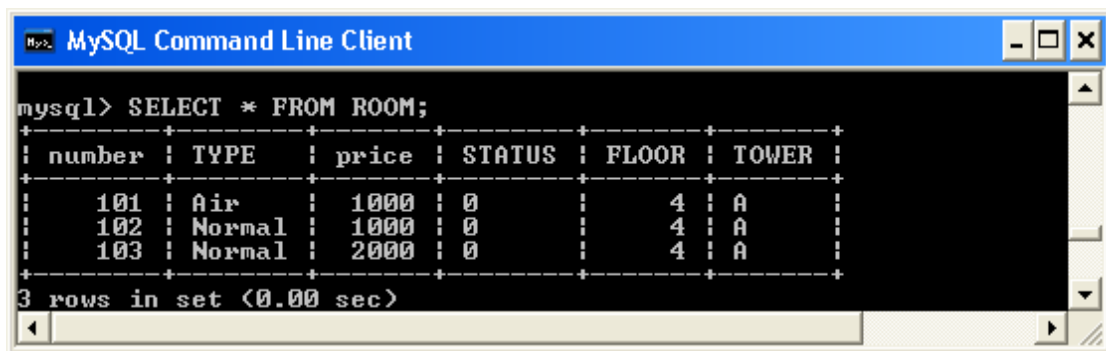


การเพิ่มข้อมูลครั้งละหลายแถวพร้อม ๆ กัน

```
INSERT INTO Room (number, type, price, status, floor,tower)
VALUE          (102, 'Normal', 1000, '0', 4, 'A' ),
              (103, 'Normal', 2000, '0', 4, 'A');
```



ใช้คำสั่งเรียกดูข้อมูลอย่างง่าย > Select * From Room ; จะปรากฏรายการข้อมูลที่ก็ย้เข้าไป



✍ ให้นักศึกษาเพิ่มเติมข้อมูลจากตารางที่กำหนดให้ และใช้คำสั่งแสดงผลข้อมูลเพื่อตรวจสอบผลลัพธ์ที่ได้

Number	Type	Price	Status	Floor	Tower
101	Air	1,000	0	4	A
102	Normal	1,000	0	4	A
103	Normal	2,000	0	4	A
104	Air	3,000	0	4	B
105	Air	2,000	1	5	B

✍ กิจกรรมทบทวน ให้นักศึกษาสร้างตารางผู้เช่า (Member) และจัดเก็บข้อมูลจากตัวอย่างที่กำหนดให้

✳ ตัวอย่างโครงสร้างตาราง Member

ชื่อคอลัมน์	ชนิดข้อมูล	แอทริบิวต์
Mid	Char (4)	Index, Not Null, Primary key
Mname	Char (20)	Not Null
Msurname	Char (20)	Not Null
IDCard	Char (13)	Not Null
Sex	Char (1)	
Career	Char (10)	
Status	INT	

ผลลัพธ์ที่ได้จากการสร้างตาราง Member

```
mysql> CREATE TABLE MEMBER(  
-> Mid CHAR(4) NOT NULL PRIMARY KEY,  
-> Mname CHAR(20) NOT NULL,  
-> Msurname CHAR(20) NOT NULL,  
-> IDCard CHAR(13) NOT NULL,  
-> Sex CHAR(1),  
-> Career CHAR(10),  
-> Status INT,  
-> INDEX(Mid)  
-> );  
Query OK, 0 rows affected (0.08 sec)
```

```
mysql> SHOW COLUMNS FROM MEMBER;  
+-----+-----+-----+-----+-----+-----+  
| Field | Type | Null | Key | Default | Extra |  
+-----+-----+-----+-----+-----+-----+  
| Mid   | char(4) |      | PRI |          |       |  
| Mname | char(20) |      |     |          |       |  
| Msurname | char(20) |      |     |          |       |  
| IDCard | char(13) |      |     |          |       |  
| Sex   | char(1) | YES  |     | NULL    |       |  
| Career | char(10) | YES  |     | NULL    |       |  
| Status | int(11) | YES  |     | NULL    |       |  
+-----+-----+-----+-----+-----+-----+  
7 rows in set (0.00 sec)
```

※ ตัวอย่างข้อมูลตาราง Member

Mid	Mname	Msurname	IDCard	Sex	Career	Status
0001	Mr.A	Vat	111111111111	M	Teacher	1
0002	Mr.B	Yonok	222222222222	M	Teacher	1
0003	Ms.C	Onair	333333333333	F	DBA	1
0004	Mrs.D	Tai	444444444444	F	SA	0
0005	Mr.M	Vat	555555555555	M	Police	0

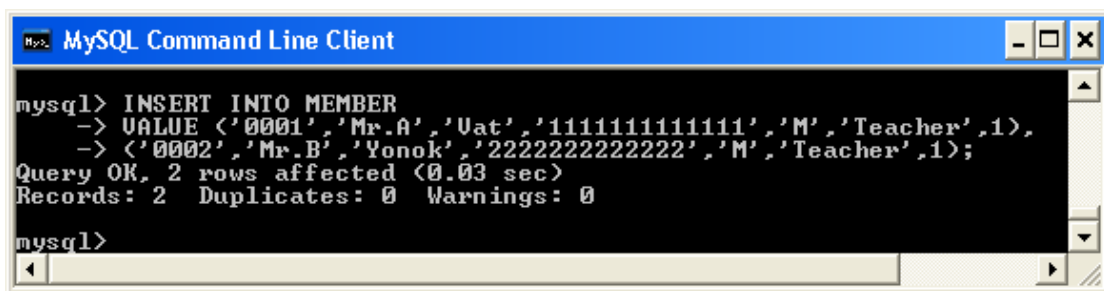
※ การเพิ่มข้อมูลทุกคอลัมน์

เป็นรูปแบบการนำข้อมูลเข้าสู่ตารางฐานข้อมูลโดยไม่ต้องระบุชื่อของคอลัมน์รูปแบบ

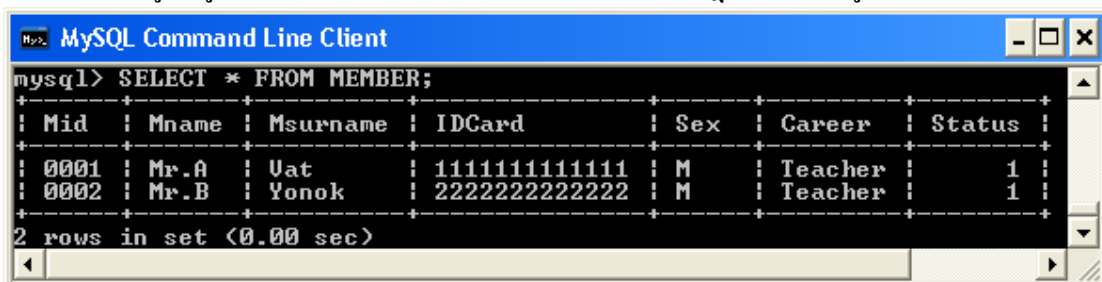
```
INSERT INTO ชื่อตาราง
VALUE (ค่าของคอลัมน์1, ค่าของคอลัมน์2, ...,ค่าของคอลัมน์N),
(ค่าของคอลัมน์1, ค่าของคอลัมน์2, ...,ค่าของคอลัมน์N),
.....
(ค่าของคอลัมน์1, ค่าของคอลัมน์2, ...,ค่าของคอลัมน์N);
```

เช่น

```
INSERT INTO Member
VALUE ('0001', 'Mr.A', 'Vat', '111111111111', 'M', 'Teacher', 1),
('0002', 'Mr.B', 'Yonok', '222222222222', 'M', 'Teacher', 1);
```



ใช้คำสั่งเรียกดูข้อมูลอย่างง่าย > Select * From Room ; จะปรากฏรายการข้อมูลที่คีย์เข้าไป



✍ ให้นักศึกษาใช้คำสั่ง SQL เพิ่มข้อมูลลงในตารางจนครบตามตัวอย่างที่กำหนดให้

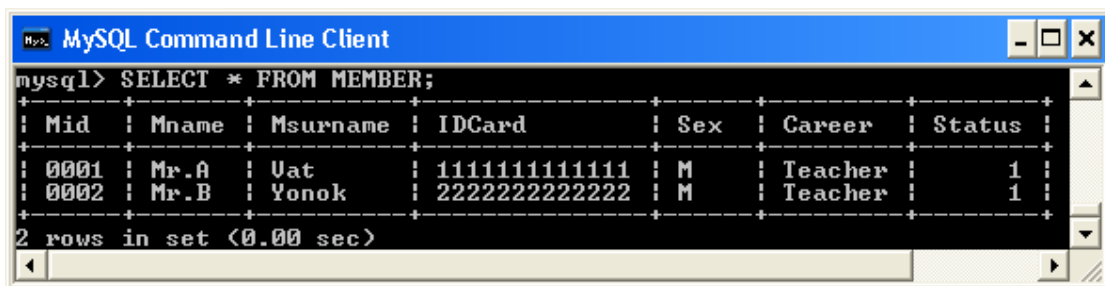
✚ การปรับปรุงข้อมูลในตาราง

ในกรณีที่ต้องการแก้ไขข้อมูลบางส่วนในแถวใช้คำสั่ง UPDATE โดยระบุว่าต้องการปรับปรุงข้อมูลในแถวใด ซึ่งจะใช้ร่วมกับคีย์เวิร์ด WHERE ที่กำหนดเงื่อนไขทดสอบ และจะปรับปรุงหรือแก้ไขข้อมูลนั้นก็ต่อเมื่อผลการทดสอบเงื่อนไขนั้นเป็นจริงเท่านั้น

รูปแบบ

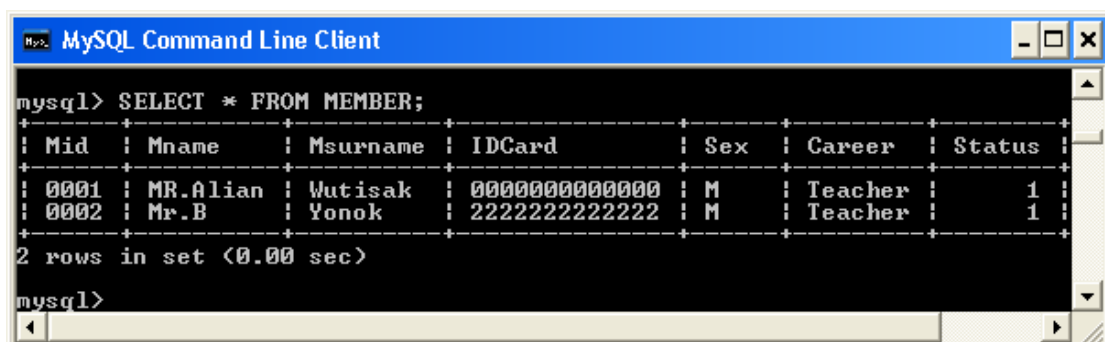
```
UPDATE ชื่อตาราง
SET ชื่อคอลัมน์ที่1 = ค่าข้อมูล , ชื่อคอลัมน์ที่2 = ค่าข้อมูล...,ชื่อคอลัมน์ที่N = ค่าข้อมูล
WHERE เงื่อนไข ;
```

เช่น สมมติต้องการเปลี่ยน ข้อมูลของรหัส 0001 ใหม่ทั้งหมด ให้ใช้คำสั่ง ดังนี้



```
UPDATE MEMBER
SET Mname = 'MR.Alian', Msurname = 'Wutisak', IDCard = '000000000000'
WHERE Mid = '0001' ;
```

ใช้คำสั่งเรียกดูข้อมูลอย่างง่าย > Select * From MEMBER ; จะปรากฏรายการข้อมูลที่เปลี่ยนแปลง



ตารางแสดงตัวอย่างการกำหนดเงื่อนไขของ WHERE

ตัวดำเนินการเปรียบเทียบ	ชื่อ	ความหมาย	ตัวอย่าง
=	เท่ากับ	เป็นจริงเมื่อนิพจน์แรกเท่ากับ นิพจน์ที่สอง	WHERE STATUS = 1
<>	ไม่เท่ากับ	เป็นจริงเมื่อนิพจน์แรกไม่เท่ากับ นิพจน์ที่สอง	WHERE STATUS <> 1
>	มากกว่า	เป็นจริงเมื่อนิพจน์แรกมากกว่า นิพจน์ที่สอง	WHERE AGE > 60
<	น้อยกว่า	เป็นจริงเมื่อนิพจน์แรกน้อยกว่า นิพจน์ที่สอง	WHERE AGE < 60
>=	มากกว่า หรือเท่ากับ	เป็นจริงเมื่อนิพจน์แรกมากกว่า หรือเท่ากับนิพจน์ที่สอง	WHERE PRICE >= 200
<=	น้อยกว่า หรือเท่ากับ	เป็นจริงเมื่อนิพจน์แรกน้อยกว่า หรือเท่ากับนิพจน์ที่สอง	WHERE PRICE <= 200



การลบข้อมูลในตาราง

การลบข้อมูลในตารางสามารถทำได้โดยการกำหนดเงื่อนไขผ่านคีย์เวิร์ด WHERE ซึ่ง
เหมือนกับการปรับปรุงข้อมูล หรือจะเลือกลบข้อมูลทุกแถวก็ได้

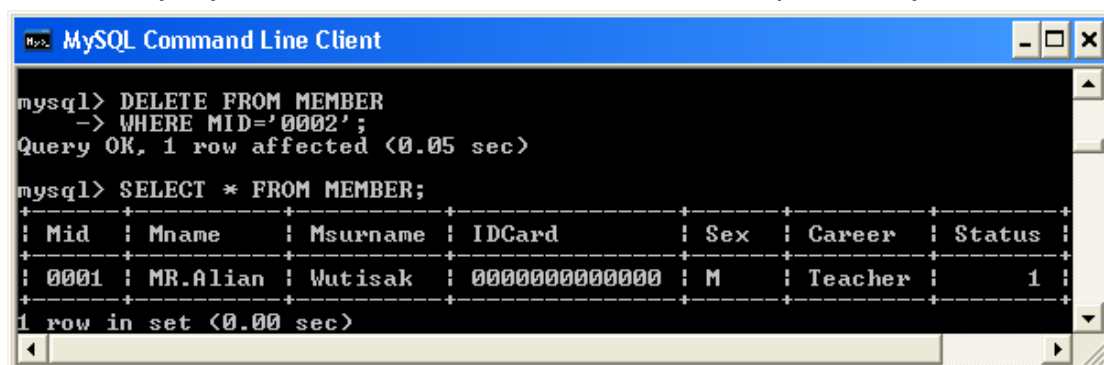
รูปแบบ การลบข้อมูลบางแถวตามเงื่อนไขที่ต้องการ

```
DELETE FROM ชื่อตาราง
WHERE เงื่อนไข ;
```

เช่น ต้องการลบข้อมูลของรหัส 002

```
DELETE FROM MEMBER
WHERE Mid = '0002' ;
```

ใช้คำสั่งเรียกดูข้อมูลอย่างง่าย > Select * From MEMBER ; จะปรากฏรายการข้อมูลที่เปลี่ยนแปลง



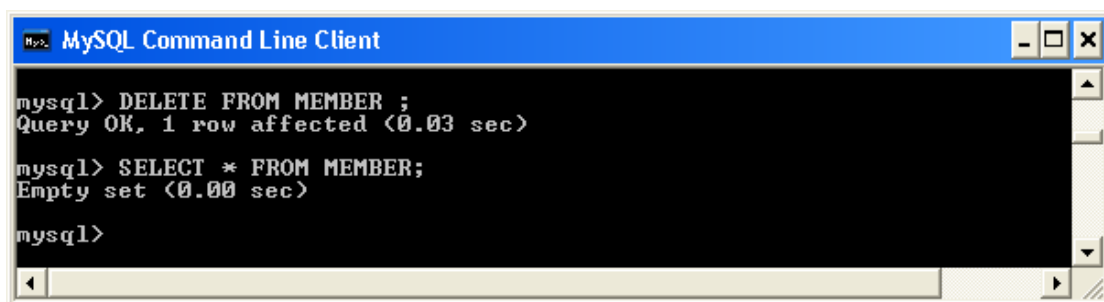
รูปแบบ การลบข้อมูลทุกแถวของตาราง

```
DELETE FROM ชื่อตาราง ;
```

เช่น ต้องการลบข้อมูลทุกแถวของตาราง MEMBER

```
DELETE FROM MEMBER ;
```

ใช้คำสั่งเรียกดูข้อมูลอย่างง่าย > Select * From MEMBER ; จะปรากฏรายการข้อมูลที่เปลี่ยนแปลง



```
MySQL Command Line Client
mysql> DELETE FROM MEMBER ;
Query OK, 1 row affected (0.03 sec)

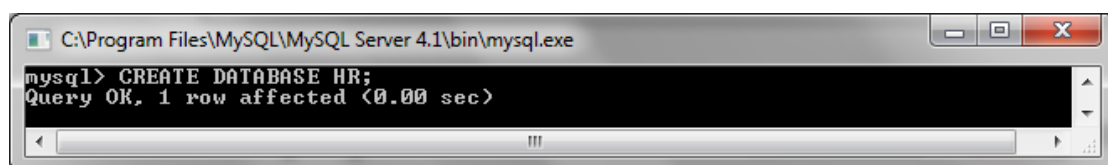
mysql> SELECT * FROM MEMBER;
Empty set (0.00 sec)

mysql>
```

จากภาพจะเห็นได้ว่า โปรแกรมจะแสดงข้อความ 'Empty set' แสดงว่าไม่พบข้อมูลที่ค้นหา หรือมีค่าว่างเปล่า ซึ่งเกิดขึ้นจากการใช้คำสั่งลบข้อมูลทิ้งแถวออกไปนั่นเอง

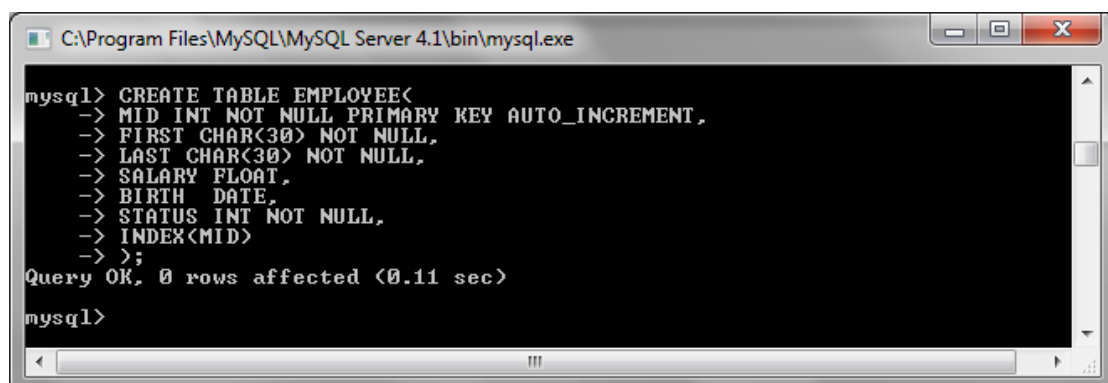
✍ กิจกรรมทบทวนบทเรียน

1. สร้างฐานข้อมูลชื่อ HR



```
C:\Program Files\MySQL\MySQL Server 4.1\bin\mysql.exe
mysql> CREATE DATABASE HR;
Query OK, 1 row affected (0.00 sec)
```

2. สร้างตารางข้อมูล ชื่อ EMPLOYEE



```
C:\Program Files\MySQL\MySQL Server 4.1\bin\mysql.exe
mysql> CREATE TABLE EMPLOYEE<
-> MID INT NOT NULL PRIMARY KEY AUTO_INCREMENT,
-> FIRST CHAR(30) NOT NULL,
-> LAST CHAR(30) NOT NULL,
-> SALARY FLOAT,
-> BIRTH DATE,
-> STATUS INT NOT NULL,
-> INDEX(MID)
-> >;
Query OK, 0 rows affected (0.11 sec)

mysql>
```


※ ตัวอย่างข้อมูลตาราง Employee

Mid	First	Last	Salary	Birth	Status
200	John	P.	10,000.00	1976-01-17	1
201	Jenny	J.	10,000.00	1977-01-17	1
202	Anthony	K.	15,000.00	1976-02-09	1
203	Moon	L.	30,000.00	1980-10-11	0
204	Polo	T.	20,000.00	1979-11-20	1

```

C:\Program Files\MySQL\MySQL Server 4.1\bin\mysql.exe
mysql> SELECT * FROM EMPLOYEE;
+----+-----+-----+-----+-----+-----+
| MID | FIRST | LAST | SALARY | BIRTH | STATUS |
+----+-----+-----+-----+-----+-----+
| 200 | JOHN  | P.   | 10000  | 1976-01-17 | 1      |
| 201 | JENNY | J.   | 10000  | 1977-01-17 | 1      |
| 202 | ANTHONY | K.   | 15000  | 1976-02-09 | 1      |
| 203 | MOON  | L.   | 30000  | 1980-10-11 | 0      |
| 204 | POLO  | T.   | 20000  | 1979-11-20 | 1      |
+----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
    
```

3. สร้างตารางข้อมูล ชื่อ PAYSLIP

```

C:\Program Files\MySQL\MySQL Server 4.1\bin\mysql.exe
mysql> CREATE TABLE PAYSLIP(
-> SLIPID INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
-> MID INT NOT NULL,
-> OT FLOAT,
-> EXTRA FLOAT,
-> TOTAL FLOAT,
-> INDEX(SLIPID)
-> );
Query OK, 0 rows affected (0.09 sec)
    
```

※ ตัวอย่างข้อมูลตาราง PAYSLIP

SLIPID	Mid	OT	Extra	Total
1	200	3,000	1,000	14,000
2	201	5,000	2,000	17,000
3	202	2,500	3,000	20,500
4	200	4,000	500	14,500
5	201	1,000	2,500	13,500

```

C:\Program Files\MySQL\MySQL Server 4.1\bin\mysql.exe
mysql> SELECT * FROM PAYSLIP;
+----+-----+-----+-----+-----+
| SLIPID | MID | OT | EXTRA | TOTAL |
+----+-----+-----+-----+-----+
| 1      | 200 | 3000 | 1000  | 14000 |
| 2      | 201 | 5000 | 2000  | 17000 |
| 3      | 202 | 2500 | 3000  | 20500 |
| 4      | 200 | 4000 | 500   | 14500 |
| 5      | 201 | 1000 | 2500  | 13500 |
+----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
    
```

กิจกรรมแนะเชิงเจตคติ

จงเติมข้อความในช่องว่างให้สมบูรณ์

1. จงบอกข้อคำนึงที่สำคัญของการนำข้อมูลใส่ในตาราง

.....

.....

.....

.....

.....

.....

.....

2. จงบอกขั้นตอนของคำสั่งการเพิ่มข้อมูลใหม่

.....

.....

.....

3. จงบอกขั้นตอนการเพิ่มข้อมูลทุกคอลัมน์

.....

.....

.....

4. จงบอกขั้นตอนการปรับปรุงข้อมูลในตาราง

.....

.....

.....

5. จงบอกขั้นตอนการลบข้อมูลในตาราง

.....

.....

.....

กิจกรรมเสนอแนะเชิงทักษะ

จากตารางข้อมูลที่กำหนดให้ ให้นักศึกษาสร้างตารางข้อมูลเพิ่มเติมลงในฐานข้อมูล Apartment พร้อมทั้งบันทึกข้อมูลลงในตารางให้ครบตามที่กำหนดให้

1. สร้างตารางการเช่า **Rent**

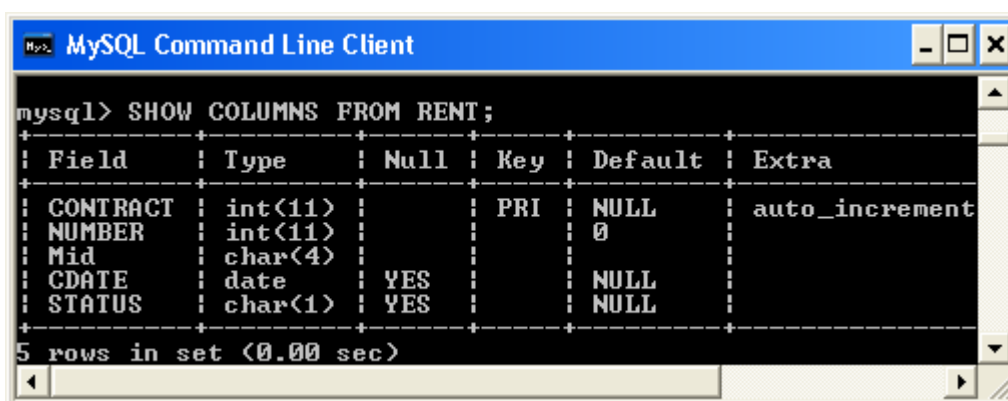
※ ตัวอย่างโครงสร้างตาราง Rent

ชื่อคอลัมน์	ชนิดข้อมูล	แอทริบิวต์
Contract	INT	Index, Not Null, Primary key, Auto_Increment
Number	INT	Not Null
Mid	Char (4)	Not Null
CDate	Date	
Status	Char (1)	Not Null

※ ตัวอย่างข้อมูลตาราง Rent

Contract	Number	Mid	Cdate	Status
1001	101	0005	2010-09-01	T
1002	102	0004	2010-09-09	T
1003	103	0003	2010-10-10	F
1004	104	0002	2010-11-30	T
1005	105	0001	2012-12-31	F

* การใช้ข้อมูลประเภทวันที่ ต้องอยู่ภายใต้เครื่องหมาย ‘‘



2. สร้างตารางใบเสร็จ Bill

✘ ตัวอย่างโครงสร้างตาราง Bill

ชื่อคอลัมน์	ชนิดข้อมูล	แอทริบิวต์
BillCode	INT	Index, Not Null, Primary key, Auto_Increment
Number	INT	Not Null
BillDate	DATE	Not Null
Total	FLOAT	Not Null
Status	Char (1)	Not Null

✘ ตัวอย่างข้อมูลตาราง Bill

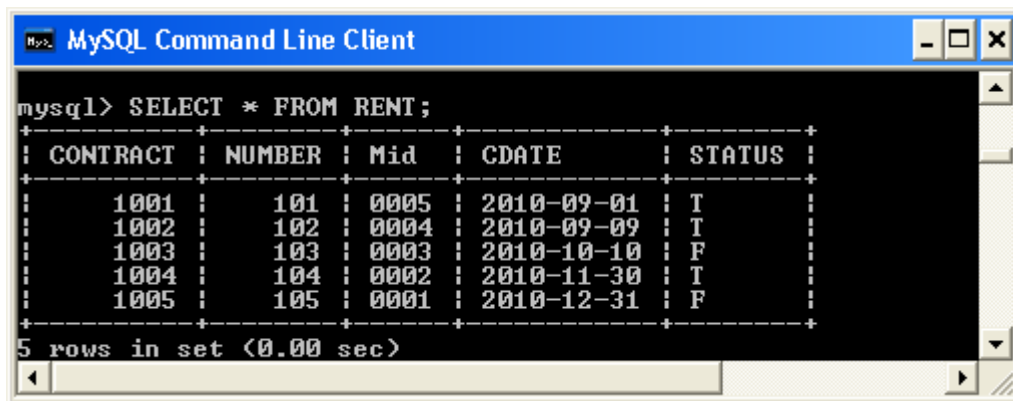
BillCode	Number	BillDate	Total	Status
1	101	2010-10-30	2,000	Y
2	102	2010-10-30	2,500	Y
3	103	2010-11-30	3,500	N
4	101	2010-11-30	4,000	N
5	102	2010-11-30	4,000	Y

* การใส่ข้อมูลประเภทวันที่ ต้องอยู่ภายใต้เครื่องหมาย ‘ ‘

```

mysql> SHOW COLUMNS FROM BILL;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| BILLCODE | int(11) | | PRI | NULL | auto_increment |
| NUMBER | int(11) | | | 0 | |
| BILLDATE | date | YES | | NULL | |
| TOTAL | float | | | 0 | |
| STATUS | char(1) | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
    
```

ผลลัพธ์ที่ได้จากการใส่ข้อมูลลงในตาราง Rent

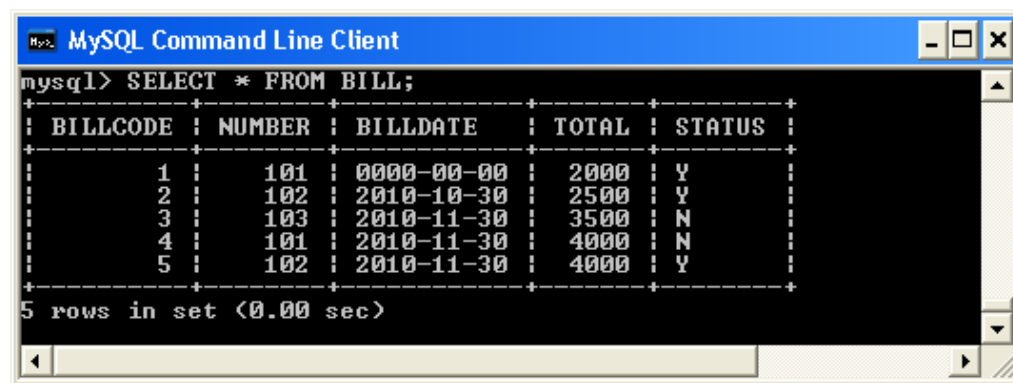


```
mysql> SELECT * FROM RENT;
```

CONTRACT	NUMBER	Mid	CDATE	STATUS
1001	101	0005	2010-09-01	T
1002	102	0004	2010-09-09	T
1003	103	0003	2010-10-10	F
1004	104	0002	2010-11-30	T
1005	105	0001	2010-12-31	F

5 rows in set (0.00 sec)

ผลลัพธ์ที่ได้จากการใส่ข้อมูลลงในตาราง Bill



```
mysql> SELECT * FROM BILL;
```

BILLCODE	NUMBER	BILLDATE	TOTAL	STATUS
1	101	0000-00-00	2000	Y
2	102	2010-10-30	2500	Y
3	103	2010-11-30	3500	N
4	101	2010-11-30	4000	N
5	102	2010-11-30	4000	Y

5 rows in set (0.00 sec)