

LAB 1 – Introduction to MySQL

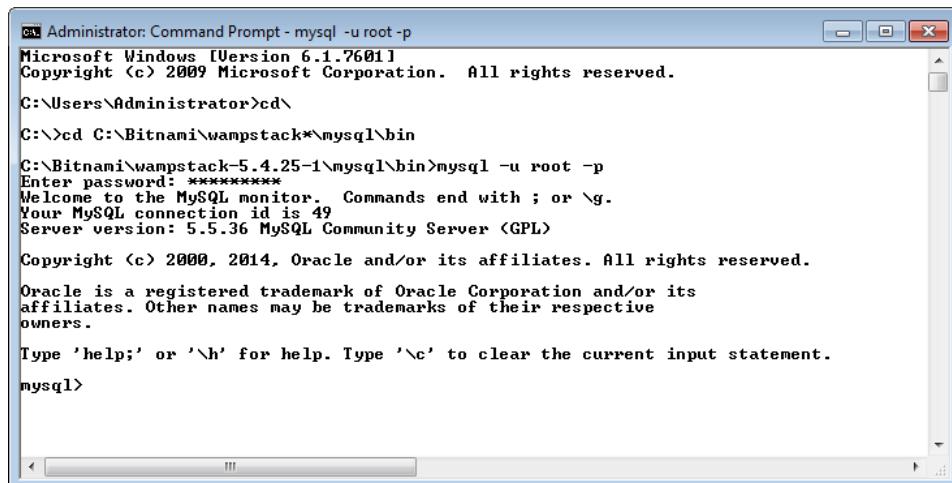
1. คำสั่งพื้นฐานของ MySQL

1.1 คำสั่ง login เข้าสู่ MySQL

1.1.1 พิมพ์ `cd\` เพื่อออก ไปยัง root directory

1.1.2 พิมพ์ `cd C:\Bitnami\wampstack*\mysql\bin`

1.1.3 พิมพ์คำสั่ง `mysql -u root -p` ซึ่งหมายถึงกำหนดชื่อ user เป็น root ส่วน `-p` หมายถึง ให้ทำการป้อนรหัสผ่านสำหรับเข้าไปใช้งาน



```
c:\ Administrator: Command Prompt - mysql -u root -p
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>cd\
C:\>cd C:\Bitnami\wampstack*\mysql\bin
C:\Bitnami\wampstack-5.4.25-1\mysql\bin>mysql -u root -p
Enter password: *****
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 49
Server version: 5.5.36 MySQL Community Server (GPL)

Copyright (c) 2000, 2014, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

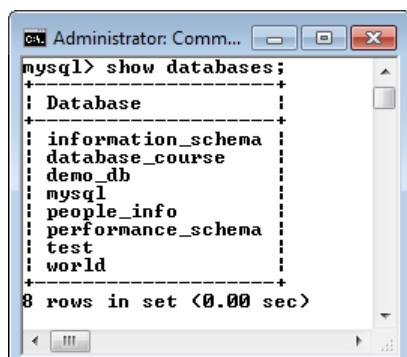
mysql>
```

1.2 คำสั่ง login เข้าสู่ mysql database ที่อยู่บนเครื่อง

`mysql -h hostname -u root -p`

1.3 คำสั่งแสดงรายชื่อฐานข้อมูล

`show databases;`



```
c:\ Administrator: Comm...
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| database_course |
| demo_db |
| mysql |
| people_info |
| performance_schema |
| test |
| world |
+-----+
8 rows in set (0.00 sec)
```

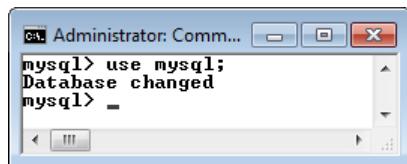
ชื่อ-นามสกุล..... รหัสนิสิต..... เลขที่.....

1.4 คำสั่งเลือกใช้ฐานข้อมูล

```
use database_name;
```

เช่น

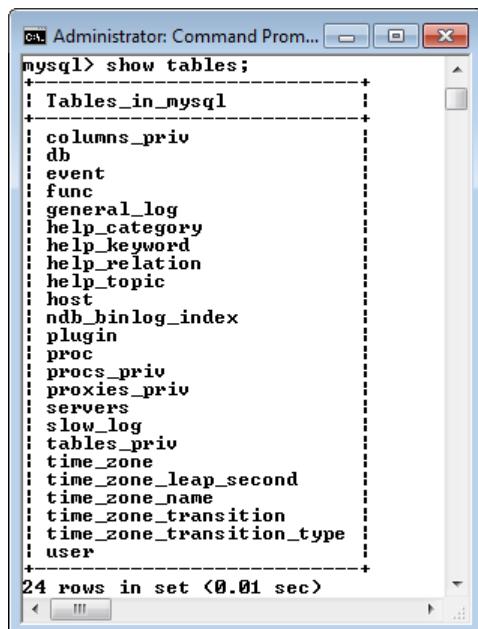
```
use mysql;
```



A screenshot of the MySQL Command Prompt window titled "Administrator: Comm...". The command "use mysql;" is entered, followed by "Database changed" and a prompt symbol.

1.5 คำสั่งแสดงรายชื่อตาราง

```
show tables;
```



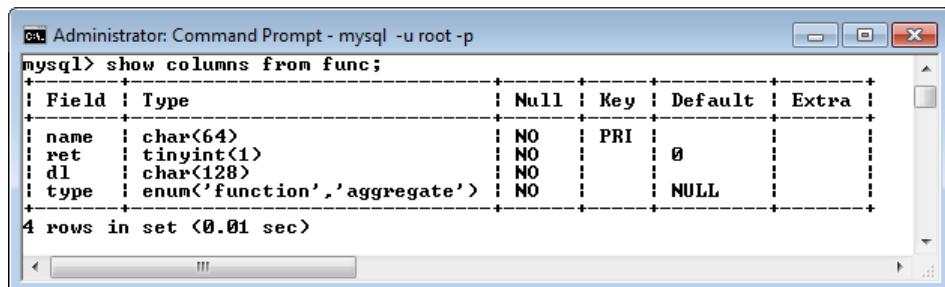
A screenshot of the MySQL Command Prompt window titled "Administrator: Command Prom...". The command "show tables;" is entered, displaying a table titled "Tables_in_mysql" with 24 rows. The rows list various system tables such as columns_priv, db, event, func, general_log, help_category, help_keyword, help_relation, help_topic, host, ndb_binlog_index, plugin, proc, procs_priv, proxies_priv, servers, slow_log, tables_priv, time_zone, time_zone_leap_second, time_zone_name, time_zone_transition, time_zone_transition_type, and user. The output ends with "24 rows in set <0.01 sec>".

1.6. คำสั่งแสดงโครงสร้างของตาราง

```
show columns from table_name
```

เช่น

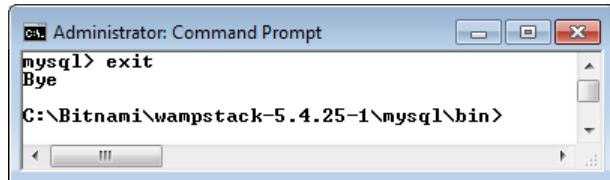
```
show columns from func;
```



A screenshot of the MySQL Command Prompt window titled "Administrator: Command Prompt - mysql -u root -p". The command "show columns from func;" is entered, displaying a table with 4 rows. The columns are Field, Type, Null, Key, Default, and Extra. The data shows four columns: name (char(64), NO, PRI, @), ret (tinyint(1), NO, @), d1 (char(128), NO, NULL), and type (enum('function','aggregate'), NO, NULL). The output ends with "4 rows in set <0.01 sec>".

1.7 คำสั่ง logout ออกจาก MySQL

exit หรือ \q



2. ประเภทของข้อมูลในฐานข้อมูล MySQL

Integer	Float & Double	String & Text	Date & Time
- TINYINT	- FLOAT	- CHAR	- DATE
- SMALLINT	- DOUBLE	- VARCHAR	- TIME
- MEDIUMINT		- TINYTEXT	- DATETIME
- INT		- TEXT	- TIMESTAMP
- BIGINT		- MEDIUMTEXT	- YEAR
		- LONGTEXT	
		- BLOB	
		- TINYBLOB	
		- MEDIUMBLOB	
		- LONGBLOB	
		- ENUM	
		- SET	

3. Storage Engine

Storage Engine คือ ตัวจัดการตาราง (Table Handlers) หรือตัวที่กำหนดลักษณะการจัดเก็บข้อมูล โครงสร้างของตารางภายในฐานข้อมูล โปรแกรมฐานข้อมูล MySQL สนับสนุน Storage Engine หลายประเภท โดยขึ้นอยู่กับ version ของ MySQL ซึ่งสามารถตรวจสอบว่า MySQL ที่ใช้สนับสนุน Storage Engine ไหนบ้าง โดยใช้คำสั่งต่อไปนี้

show engines

ผลลัพธ์จากการใช้คำสั่งแสดง storage engine ที่สนับสนุนแสดงดังรูป

Engine	Support	Comment	Transactions	XA	Savepoints
FEDERATED	NO	Federated MySQL storage engine	NULL	NULL	NULL
MRG_MYISAM	YES	Collection of identical MyISAM tables	NO	NO	NO
MyISAM	YES	MyISAM storage engine	NO	NO	NO
BLACKHOLE	YES	/dev/null storage engine (anything you write to it disappears)	NO	NO	NO
CSV	YES	CSV storage engine	NO	NO	NO
MEMORY	YES	Hash based, stored in memory, useful for temporary tables	NO	NO	NO
ARCHIVE	YES	Archive storage engine	NO	NO	NO
InnoDB	DEFAULT	Supports transactions, row-level locking, and foreign keys	YES	YES	YES
PERFORMANCE_SCHEMA	YES	Performance Schema	NO	NO	NO

4. Data Definition Language

Data Definition Language (DDL) คือกลุ่มภาษา SQL ที่เกี่ยวข้องกับกลุ่มของคำสั่ง 3 กลุ่ม คือ

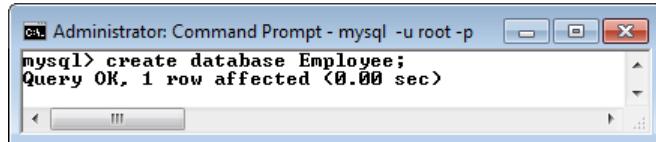
- คำสั่ง create ประกอบไปด้วย การสร้างฐานข้อมูล การสร้างตาราง
- คำสั่ง drop ประกอบไปด้วย การลบฐานข้อมูล การลบตาราง
- คำสั่ง alter ประกอบไปด้วยคำสั่งที่ใช้ในการเปลี่ยนแปลงโครงสร้างตาราง

4.1 คำสั่งสร้างฐานข้อมูล

create database database_name;

เช่น

create database Employee;



4.2 คำสั่งสร้างตาราง

create table table_name (table definition) [engine=storage_engine_type];

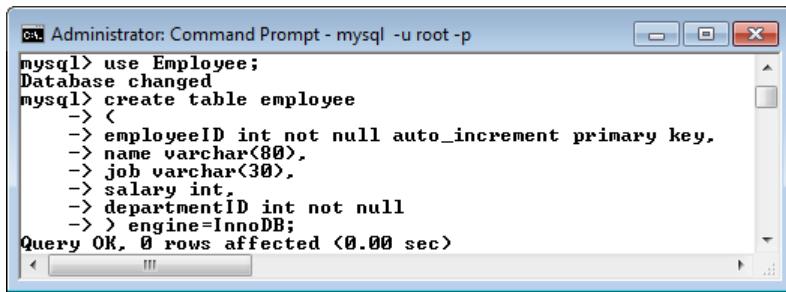
โดยที่

(table definition) คือ การกำหนดชื่อ field และคุณสมบัติต่างๆของ field

[engine=storage_engine_type] คือ การกำหนดประเภทของตาราง (Storage Engine) อาจจะไม่ต้องกำหนดก็ได้

ตัวอย่างการสร้างตารางแสดงดังรูป

ชื่อ-นามสกุล..... รหัสนิสิต..... ลำดับ.....



```
Administrator: Command Prompt - mysql -u root -p
mysql> use Employee;
Database changed
mysql> create table employee
-> (
-> employeeID int not null auto_increment primary key,
-> name varchar(80),
-> job varchar(30),
-> salary int,
-> departmentID int not null
-> ) engine=InnoDB;
Query OK, 0 rows affected (0.00 sec)
```

4.3 คำสั่งลบตาราง

```
drop table table_name;
```

เช่น

```
drop table employee;
```

4.4 คำสั่งลบฐานข้อมูล

```
drop database database_name;
```

เช่น

```
drop database Employee;
```

4.5 คำสั่งเปลี่ยนแปลงชื่อตาราง

```
alter table old_table_name rename new_table_name;
```

โดยที่

old_table_name คือชื่อตารางเดิม

new_table_name คือ ชื่อตารางใหม่

เช่น

```
alter table Employee rename New_Employee;
```

4.6 คำสั่งเปลี่ยนแปลงประเภท field ในตาราง

```
alter table table_name modify column_name new_type;
```

โดยที่

table_name คือ ชื่อตารางที่ต้องการเปลี่ยนแปลง

column_name คือ ชื่อ column หรือ field ที่ต้องการเปลี่ยนแปลงประเภทข้อมูล

new_type คือ ประเภทของข้อมูลที่ต้องการเปลี่ยนแปลงใหม่

เช่น

```
alter table Employee modify name text;
```

ชื่อ-นามสกุล.....รหัสนิสิต.....เลขที่.....

4.7 คำสั่งเปลี่ยนแปลงชื่อ field

alter table table_name change old_field_name new_field_name new_type;

โดยที่

table_name คือ ชื่อตารางที่ต้องการเปลี่ยนแปลง

old_field_name คือ ชื่อ column หรือ field ที่ต้องการเปลี่ยนแปลง

new_field_name คือ ชื่อ column หรือ field ใหม่

new_type คือ ประเภทของข้อมูลที่ต้องการเปลี่ยนแปลงใหม่

เช่น

```
alter table Employee change name emp_name varchar(80);
```

4.8 คำสั่งเพิ่ม field

alter table table_name add column column_name type

โดยที่

table_name คือ ชื่อตารางที่ต้องการเปลี่ยนแปลง

column_name คือ ชื่อ column หรือ field ที่ต้องการเพิ่ม

new_type คือ ประเภทข้อมูลของ field ที่ต้องการเพิ่มใหม่

เช่น

```
alter table Employee add column emp_age int;
```

4.9 คำสั่งลบ field

alter table table_name drop column column_name

โดยที่

table_name คือ ชื่อตารางที่ต้องการเปลี่ยนแปลง

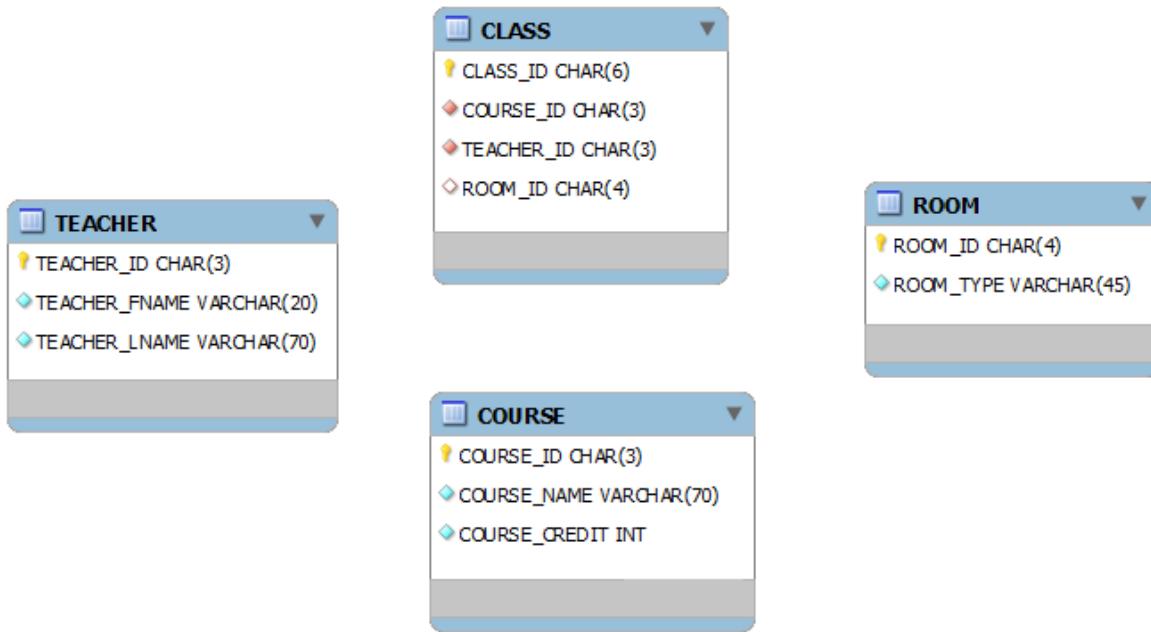
column_name คือ ชื่อ column หรือ field ที่ต้องการเพิ่ม

เช่น

```
alter table Employee drop column emp_age;
```

แบบฝึกหัด

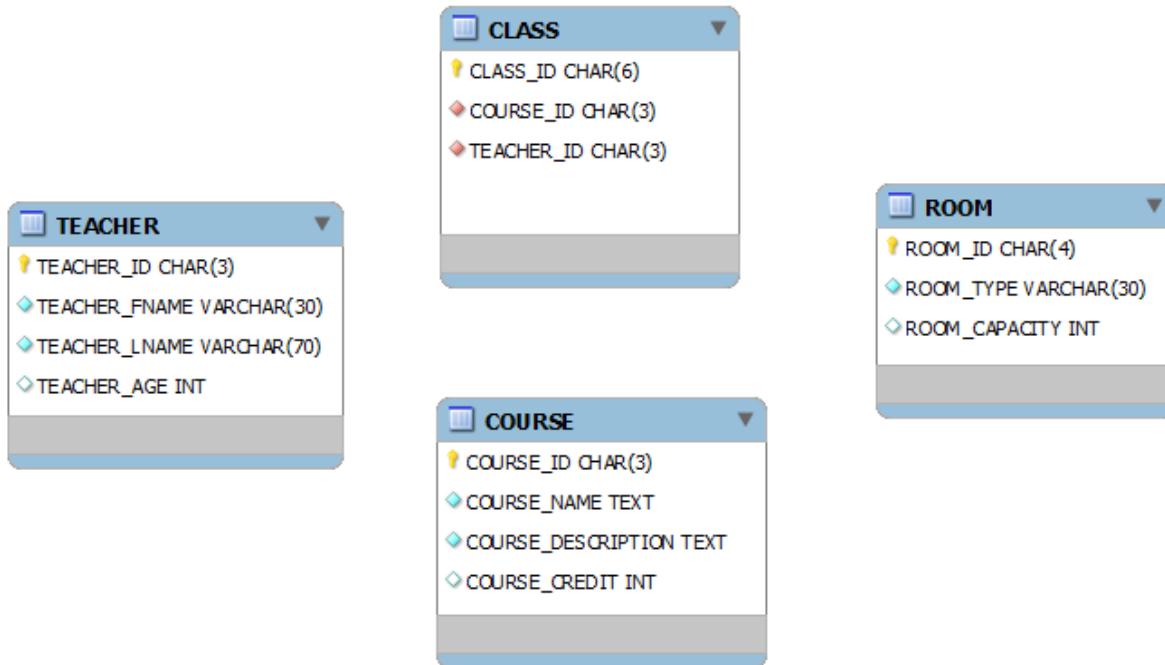
- จงแสดงคำสั่งทั้งหมดเพื่อใช้ในการสร้างฐานข้อมูลดังต่อไปนี้



ชื่อ-นามสกุล..... รหัสนิสิต..... เลขที่.....

ชื่อ-นามสกุล..... รหัสนิสิต..... เลขที่.....

2. จากข้อหนึ่ง หากฐานข้อมูลมีการเปลี่ยนแปลงไปเป็นดังรูปข้างล่างนี้ งดแสดงคำสั่งทั้งหมดเพื่อใช้ในการเปลี่ยนแปลงนี้



ชื่อ-นามสกุล..... รหัสนิสิต..... เลขที่.....

ชื่อ-นามสกุล.....รหัสนิสิต.....เลขที่.....

3. จงสร้างฐานข้อมูลชื่อ Employee สำหรับเก็บข้อมูลพนักงาน โดยฐานข้อมูลนี้ประกอบไปด้วยตารางทั้งหมด 5 ตารางดังนี้ (ฐานข้อมูลที่สร้างในข้อนี้จะนำไปใช้ในแบบฝึกหัดครั้งหน้า)

Table: Employee เก็บข้อมูลรายละเอียดของพนักงานแต่ละคน

ชื่อ Filed	ประเภท	ความหมาย
Employee_ID	INT	รหัสพนักงาน โดยกำหนดให้เป็น Primary Key, มีลักษณะเป็น Auto Increment (เพิ่มค่าโดยอัตโนมัติ) และกำหนดไม่ให้มีค่าว่าง (Not Null)
Name	VARCHAR(80)	ชื่อพนักงาน
Job	VARCHAR(30)	ตำแหน่งงานของพนักงาน
Salary	INT	เงินเดือนของพนักงาน
Department_ID	INT	รหัสแผนกที่พนักงานสังกัดอยู่ โดยกำหนดไม่ให้มีค่าว่าง

Table: Department เก็บข้อมูลรายละเอียดของแผนกงานในบริษัท

ชื่อ Filed	ประเภท	ความหมาย
Department_ID	INT	รหัสแผนก โดยกำหนดให้เป็น Primary Key, มีลักษณะเป็น Auto Increment (เพิ่มค่าโดยอัตโนมัติ) และกำหนดไม่ให้มีค่าว่าง (Not Null)
Name	VARCHAR(30)	ชื่อแผนก

Table: Employee_Skill เก็บข้อมูลความชำนาญของพนักงาน

ชื่อ Filed	ประเภท	ความหมาย
Employee_ID	INT	รหัสแผนก โดยกำหนดให้เป็น Primary Key, มีลักษณะเป็น Auto Increment (เพิ่มค่าโดยอัตโนมัติ) และกำหนดไม่ให้มีค่าว่าง (Not Null)
Skill	VARCHAR(15)	ความชำนาญของพนักงาน โดยกำหนดให้เป็น Primary Key ร่วมกับ filed ที่ชื่อ Employee_ID

Table: Client

เก็บข้อมูลรายละเอียดของบริษัทของลูกค้า

ชื่อ Filed	ประเภท	ความหมาย
Client_ID	INT	รหัสลูกค้า โดยกำหนดให้เป็น Primary Key, มีลักษณะเป็น Auto Increment (เพิ่มค่าโดยอัตโนมัติ) และ กำหนดไม่ให้มีค่าว่าง (Not Null)
Name	VARCHAR(40)	ชื่อบริษัทของลูกค้า
Address	VARCHAR(100)	ที่อยู่บริษัทของลูกค้า
Contact_Person	VARCHAR(80)	ชื่อของลูกค้าที่เป็นตัวแทนของบริษัท
Contact_Number	CHAR(12)	หมายเลขโทรศัพท์ของลูกค้าตัวแทน

Table: Assignment

เก็บข้อมูลรายละเอียดการติดต่อ

ชื่อ Filed	ประเภท	ความหมาย
Client_ID	INT	รหัสลูกค้า โดยกำหนดให้เป็น Primary Key และ กำหนดไม่ให้มีค่าว่าง (Not Null)
Employee_ID	INT	รหัสพนักงาน โดยกำหนดให้เป็น Primary Key ร่วม และ กำหนดไม่ให้มีค่าว่าง (Not Null)
Work_Date	DATE	วัน เดือน ปี ที่ติดต่อ โดยกำหนดให้เป็น Primary Key ร่วม และ กำหนดไม่ให้มีค่าว่าง (Not Null)
Hours	FLOAT	จำนวนชั่วโมงในการติดต่อ

ชื่อ-นามสกุล..... รหัสนิสิต..... เลขที่.....

ชื่อ-นามสกุล..... รหัสนิสิต..... เลขที่.....