

Bài 2

Khởi đầu

Mai Văn Cường
cuongmaivan@gmail.com

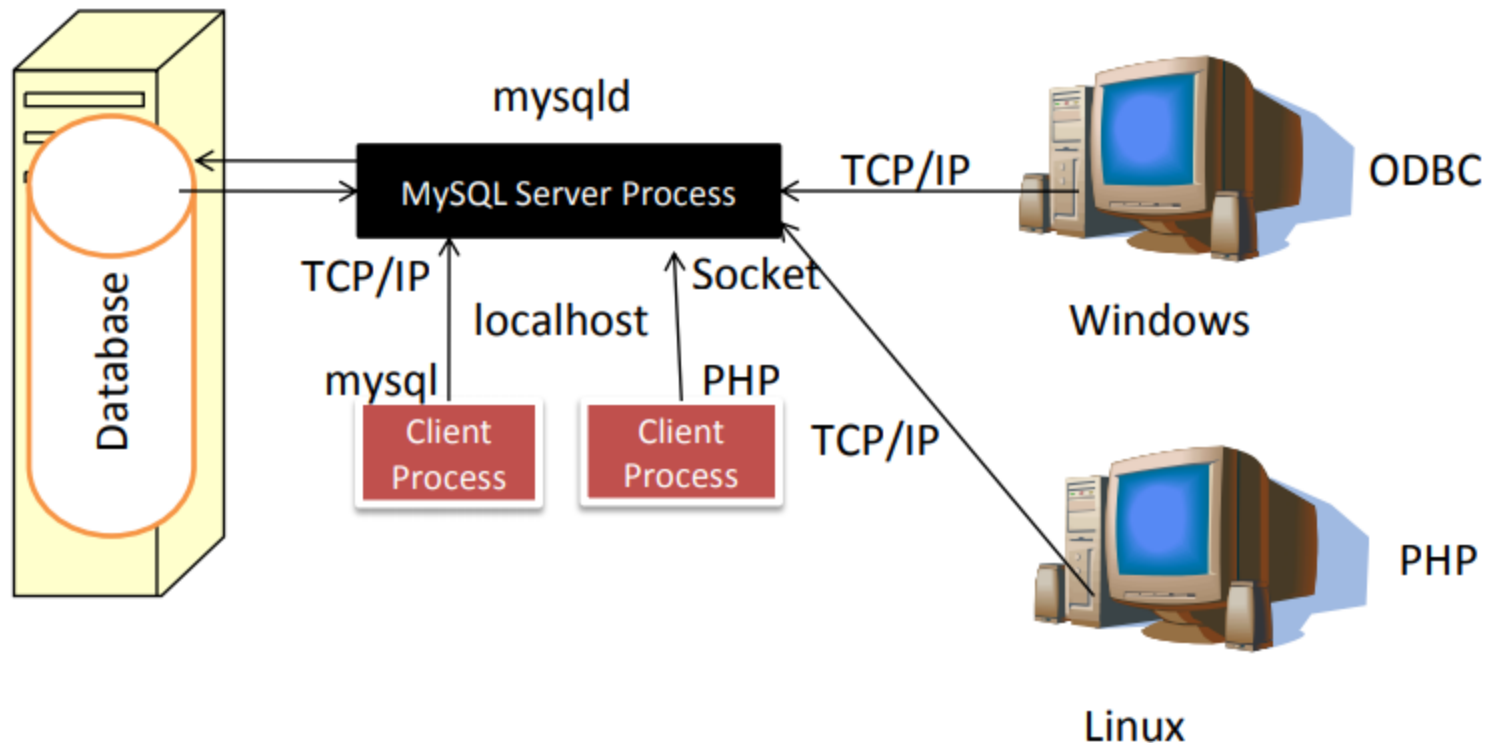
Nội dung

1. Kết nối đến MySQL Server
2. Tương tác với MySQL Server qua lệnh
3. Xem thông tin về Database và Table
4. Làm việc với Databases
5. Làm việc với Tables
6. Xem dữ liệu
7. Cập nhật dữ liệu
8. Primary key, Foreign keys
9. Batch mode

Mô hình MySQL Client/Server

- MySQL hoạt động trong một môi trường mạng, sử dụng kiến trúc Client/Server. Trong đó, một chương trình Server và vài chương trình Client kết nối với Server bằng các cách khác nhau.
- Cài đặt MySQL bao gồm các thành phần sau:
 - MySQL Server
 - Các chương trình MySQL Client
 - Các chương trình tiện ích khác

Mô hình MySQL Client/Server



Kết nối/Ngắt kết nối đến MySQL Server

- Kết nối: Từ dấu nhắc lệnh của shell, gọi chương trình **mysql**

```
1  shell> mysql -h host -u user -p
2  Enter password: *****
3  Welcome to the MySQL monitor.  Commands end with ; or \g.
4  Your MySQL connection id is 25338 to server version: 5.7.21-standard
5
6  Type 'help;' or '\h' for help. Type '\c' to clear the buffer.
7
8  mysql>
```

- Ngắt kết nối:

```
1  mysql> QUIT
2  Bye
```

Tương tác với MySQL Server qua lệnh

- Giả sử đã kết nối đến MySQL Server, có thể tương tác với MySQL Server từ MySQL Client thông qua các câu lệnh.

```
1  mysql> SELECT VERSION(), CURRENT_DATE;
2  +-----+-----+
3  | VERSION() | CURRENT_DATE |
4  +-----+-----+
5  | 5.7.1-m4-log | 2012-12-25 |
6  +-----+-----+
7  1 row in set (0.01 sec)
8  mysql>
```

Tương tác với MySQL Server qua lệnh

- Lưu ý: mysql quy ước mỗi câu lệnh được kết thúc bằng dấu “chấm phẩy” (;). Nếu một câu lệnh gửi thực hiện mà thiếu dấu này=> mysql sẽ chờ bằng dấu nhắc -> ở dòng kế tiếp.
- Nếu muốn hủy ngang: sử dụng \c
- Các dấu nhắc lệnh của mysql và ý nghĩa

Prompt	Meaning
mysql>	Ready for new query
->	Waiting for next line of multiple-line query
'>	Waiting for next line, waiting for completion of a string that began with a single quote (')
">	Waiting for next line, waiting for completion of a string that began with a double quote (")
`>	Waiting for next line, waiting for completion of an identifier that began with a backtick (`)
/*>	Waiting for next line, waiting for completion of a comment that began with /*

Xem thông tin Databases, chọn Database để làm việc

- Giả sử đã kết nối đến MySQL Server, có thể **xem các database** tại MySQL Server từ MySQL Client nhờ các câu lệnh **SHOW DATABASES**

```
1  mysql> SHOW DATABASES;  
2  +-----+  
3  | Database |  
4  +-----+  
5  | mysql    |  
6  | test     |  
7  | tmp      |  
8  +-----+
```

- Muốn **chọn 1 database** để làm việc: sử dụng lệnh **USE**

```
1  mysql> USE test  
2  Database changed
```


Xem các table trong database hiện hành

- Sử dụng lệnh **SHOW TABLES**

```
1  mysql> SHOW TABLES;
2  +-----+
3  | Tables in menagerie |
4  +-----+
5  | pet                |
6  +-----+
```

- Xem cấu trúc của table: Sử dụng lệnh **DESCRIBE**

```
1  mysql> DESCRIBE pet;
2  +-----+-----+-----+-----+-----+-----+
3  | Field  | Type          | Null | Key | Default | Extra |
4  +-----+-----+-----+-----+-----+-----+
5  | name   | varchar(20)   | YES  |     | NULL    |       |
6  | owner  | varchar(20)   | YES  |     | NULL    |       |
7  | species | varchar(20)   | YES  |     | NULL    |       |
8  | sex    | char(1)       | YES  |     | NULL    |       |
9  | birth  | date          | YES  |     | NULL    |       |
10 | death  | date          | YES  |     | NULL    |       |
11 +-----+-----+-----+-----+-----+-----+
```

LAB 2.1

- Nạp 1 database BANHANG

(HD

- Từ shell, chuyển thư mục hiện hành đến nơi chứa file InitializeSampleDB.sql
- Từ shell, chạy lệnh `mysql -u root -p < InitializeSampleDB.sql`

)

- Sử dụng các lệnh đã học, cho biết:
 - Danh sách các database mà bạn cho thể thấy
 - Danh sách các table trong database BANHANG
 - Cho biết các tên các cột và kiểu dữ liệu tương ứng của table HANGHOA

Tạo/Hủy 1 Database

- Tạo: sử dụng lệnh **CREATE DATABASE**

```
1 | mysql> CREATE DATABASE menagerie;
```

- Hủy: sử dụng lệnh **DROP DATABASE (Cẩn thận)**

```
1 | DROP {DATABASE | SCHEMA} [IF EXISTS] db_name
```

Tạo/Xóa table trong database hiện hành

- Tạo: Sử dụng lệnh **CREATE TABLE**

```
1  mysql> CREATE TABLE pet (name VARCHAR(20), owner VARCHAR(20),  
2      -> species VARCHAR(20), sex CHAR(1), birth DATE, death DATE);
```

- Xóa: Sử dụng lệnh **DROP TABLE (Cẩn thận)**

```
1  DROP [TEMPORARY] TABLE [IF EXISTS]  
2      tbl_name [, tbl_name] ...  
3      [RESTRICT | CASCADE]
```

LAB 2.2

- Tạo database có tên là EMP
- Trong EMP tạo table tên là EMPLOYEE với cấu trúc như sau
 - ID: char(9),
 - Name: varchar(50);
 - Address: varchar(200);
- Sử dụng lệnh SHOW TABLES, DESCRIBE để kiểm tra lại.
- Hủy bỏ table, database đã tạo

Xem dữ liệu của table

- Sử dụng câu lệnh SELECT

```
mysql> select * from Hanghoa;
```

mahh	tenhh	hangsx	TGBH
HH001	Mouse Quang	Logitech	1 nam
HH002	Removable Disk 256MB	Transcend	1 nam
HH003	HDD 80GB Baracuda	Seagate	2 nam
HH004	Camera SL_36	SlimPre.	3 nam
HH005	Ban phim ViewStar	Logitech	1 nam

```
5 rows in set (0.00 sec)
```

```
mysql>
```

Lab 2.3:

- Thực hiện một số câu lệnh xem dữ liệu đơn giản
 - In ra cột TENKH, DCHI của KHACH
 - In ra cột TENHH, HANGSX của HANGHOA
 - In ra cột SOHD, SL, DG, THANHTIEN của HOADON

Cập nhật dữ liệu

- Cập nhật: Thêm, Xóa, Sửa
- Thêm 1 dòng với câu lệnh INSERT

```
mysql> insert into hanghoa values ('HH006','Ban phim Sony Vaio', 'Sony','2 nam');  
Query OK, 1 row affected (0.04 sec)
```

```
mysql>
```

- Sửa dòng với câu lệnh UPDATE

```
mysql> update hanghoa set TGBH='1 nam' where MAHH='HH006';  
Query OK, 1 row affected (0.10 sec)  
Rows matched: 1  Changed: 1  Warnings: 0  
mysql>
```


Cập nhật dữ liệu

- Xóa dòng với câu lệnh DELETE

```
mysql> delete from hanghoa where mahh='HH006';  
Query OK, 1 row affected (0.09 sec)
```

```
mysql>
```

LAB 2.4

- Thực hiện các câu lệnh sau:
 - Thêm 1 khách <'KH004','An','94 Hoa Binh P5, Q.11'>
 - Thêm 1 dòng cho table HOADON ('HD008' , 'KH002 ' , '2008-05-16', 'HH005', 4,75000, 275000)
 - Cập nhật cột thanhtien cho dòng vừa thêm vào bằng 300000
 - Xóa các dòng đã thêm

Primary key

- Primary key: là 1 hoặc nhiều cột, có (tổ hợp) giá trị duy nhất, khác rỗng; dùng để nhận biết 1 dòng.

```
mysql> describe KHACH; describe HANGHOA; describe BANHANG;
```

Field	Type	Null	Key	Default	Extra
makh	char(8)	NO	PRI	NULL	
tenkh	varchar(30)	NO		NULL	
dchi	varchar(50)	YES		NULL	

3 rows in set (0.00 sec)

Field	Type	Null	Key	Default	Extra
mahh	char(8)	NO	PRI	NULL	
tenhh	varchar(50)	NO		NULL	
hangsx	varchar(10)	YES		NULL	
TGBH	char(10)	YES		NULL	

4 rows in set (0.00 sec)

Field	Type	Null	Key	Default	Extra
sohd	char(8)	NO	PRI	NULL	
makh	char(8)	NO	MUL	NULL	
ngayhd	datetime	YES		NULL	
mahh	char(8)	NO	PRI	NULL	
sl	int(11)	NO		NULL	
dg	decimal(10,2)	NO		NULL	
thanhtien	decimal(10,2)	YES		NULL	

7 rows in set (0.00 sec)

Primary key

Primary key

- Primary key thường được khai báo lúc tạo table.
- VD:

```
create table hanghoa(mahh char(8) not null,  
tenhh varchar(50) not null,  
hangsx char(10),  
TGBH char(10),  
primary key (mahh));
```

Foreign key

- Foreign key là tổ hợp gồm 1 hoặc nhiều cột, với ràng buộc là giá trị của tổ hợp cột đó (nếu khác null thì) phải được lấy từ tập các giá trị của một khóa chính của 1 table

```
# sohd, makh, ngayhd, mahh, sl, dg, thanh tien
HD005, KH003, 2008-05-14 00:00:00, HH001, 5, 50000.00, 100000.00
HD006, KH001, 2008-05-15 00:00:00, HH001, 1, 55000.00, 55000.00
HD006, KH001, 2008-05-15 00:00:00, HH003, 2, 900000.00, 180000.00
HD007, KH003, 2008-05-16 00:00:00, HH005, 3, 75000.00, 225000.00
```

```
# makh, tenkh, dchi
KH001, Nam, 123 Nguyen Dinh Chieu, Q.3, TpHCM
KH002, Nga,
KH003, Duy, 50 Nguyen Tri Phuong Q.10 TpHCM
```

Foreign key

- Foreign key thường được khai báo lúc tạo table
- VD:

```
create table BANHANG(sohd char(8) not null ,  
makh char(8) not null,  
ngayhd datetime,  
mahh char(8) not null ,  
sl int not null,  
dg decimal(10,2) not null,  
thanhtien decimal(10,2),  
primary key (sohd,mahh),  
CONSTRAINT foreign key (makh) references KHACH(makh),  
CONSTRAINT foreign key (mahh) references  
HANGHOA(mahh));
```

LAB 2.5

- Tạo database có tên là EMP
- Trong EMP tạo table tên là DEPARTMENT với cấu trúc như sau, trong đó DeptNo là primary key
 - DeptNo: char(2),
 - DeptName: varchar(50);
- Trong EMP tạo table tên là EMPLOYEE với cấu trúc như sau, trong đó SSN là primary key, DeptNo là foreign key tham chiếu đến DEPARTMENT(DeptNo)
 - SSN: char(9),
 - Name: varchar(50);
 - DeptNo: char(2);
- Thêm 1 dòng vào DEPARTMENT: ('PN','Personel');
- Thêm 1 dòng vào EMPLOYEE: ('123123123','John','PN')
- Thêm 1 dòng vào EMPLOYEE: ('111222333','Bill','TN'). Giải thích tại sao không thành công? Sửa lại câu lệnh như thế nào để có thể thực hiện được.
- Xóa các table đã tạo

Batch mode

- Thay vì phải chạy từng câu lệnh, ta có thể chạy liên tiếp nhiều câu lệnh liên tiếp được đặt trong một batch-file (hoặc script file).
- Lợi ích:
 - Đỡ phải lặp lại việc thực hiện các câu lệnh thường chạy hàng ngày
 - Có thể chia sẻ lại script file

Batch mode

- Cách gọi thực hiện batch-file

```
1 shell> mysql -h host -u user -p < batch-file
2 Enter password: *****
```

- Nếu muốn xuất kết quả thực hiện ra file:

```
1 shell> mysql < batch-file > mysql.out
```

- Trong mysql, có thể gọi thực hiện batch-file bằng **source** hoặc **\.**

```
1 mysql> source filename;
2 mysql> \. filename
```

LAB 2.6

- Soạn các batch file tên lab-2.4.sql thực hiện các nội dung ở LAB 2.4 Cho chạy lab-2.4.sql dưới Batch mode theo các cách khác nhau.
- Làm tương tự như trên cho LAB 2.5