

THIẾT KẾ & LẬP TRÌNH WEBSITE

Chương 3: Hệ Quản Trị CSDL MySql

1. Tổng quan về MySql
2. Tạo CSDL trong MySql
3. Thao tác csdl MySql

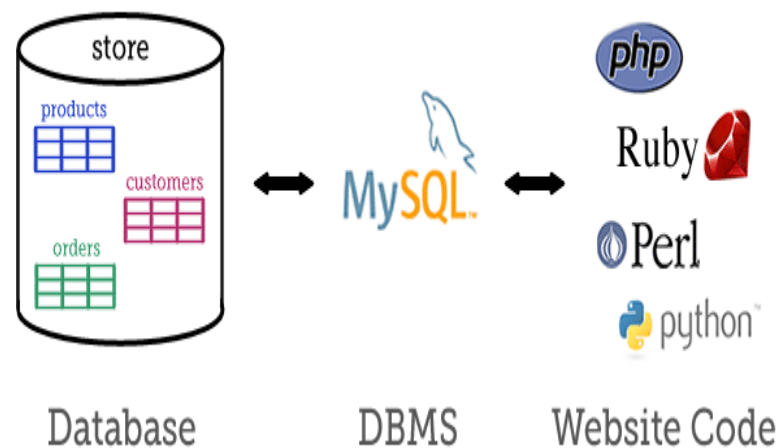
1. TỔNG QUAN VỀ MySQL

1. Giới thiệu
2. Cài đặt Sql

1.1. GIỚI THIỆU

MySQL là gì?

- MySQL là một hệ quản trị cơ sở dữ liệu phổ biến nhất thế giới, được các nhà phát triển ưa chuộng.
- MySQL có tốc độ cao, ổn định, dễ sử dụng, hoạt động trên nhiều hệ điều hành khác nhau.
- MySQL hỗ trợ nhiều ngôn ngữ lập trình như PHP, Perl,.... trong việc phát triển website



1.1. GIỚI THIỆU

Tại sao dùng MySql?

- ✓ Website luôn cần phải làm việc với cơ sở dữ liệu.
- ✓ MySQL là một hệ quản trị cơ sở dữ liệu có tốc độ cao, ổn định, dễ sử dụng.
- ✓ Hỗ trợ tốt các tính năng quản trị.
- ✓ Hỗ trợ thao tác dữ liệu nhanh, đầy đủ.



1.2.Cài đặt

Hướng dẫn cài đặt MySql:

- ✓ Có thể cài MySql bằng nhiều cách: MySQL Installer, Xampp...
- ✓ MySQL Installer cung cấp công cụ quản lý dữ liệu MySQL Workbench
- ✓ Hỗ trợ bộ dữ liệu mẫu “classicmodels”
- ✓ Cài đặt dễ dàng.

1.2.Cài đặt

Hướng dẫn cài đặt MySQL Installer:

- Tải tại đường dẫn : <https://dev.mysql.com/downloads/installer/>
- Có 2 lựa chọn
 - mysql-installer-web-community.exe: cài online từ internet
 - mysql-installer-community.exe : tải về cài offline

MySQL Installer 8.0.12

Select Operating System:

Microsoft Windows ▼

[Looking for previous GA versions?](#)

Windows (x86, 32-bit), MSI Installer

8.0.12

15.9M

[Download](#)

(mysql-installer-web-community-8.0.12.0.msi)

MD5: 387bd57f0fb07e3880d10f0c21b81686 | [Signature](#)

Windows (x86, 32-bit), MSI Installer

8.0.12

273.4M

[Download](#)

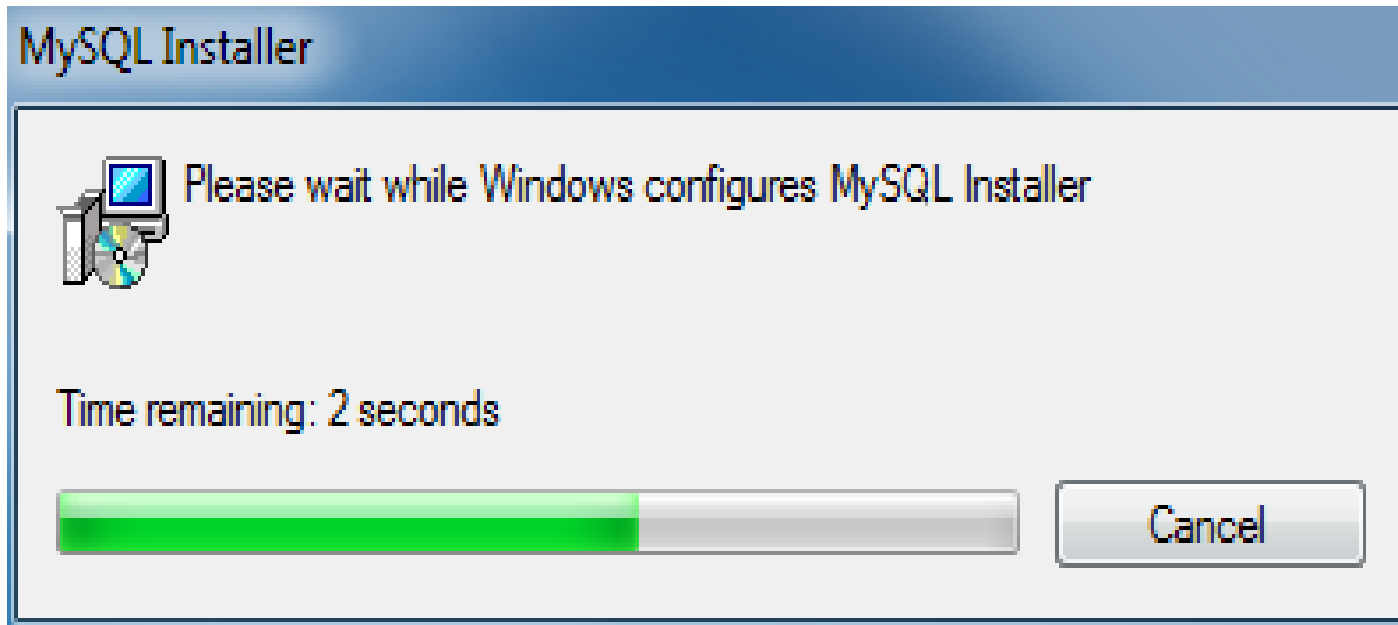
(mysql-installer-community-8.0.12.0.msi)

MD5: 53b3a9bb89db061862969b67c68b6f67 | [Signature](#)

1.2.Cài đặt

Hướng dẫn cài đặt MySQL Installer :

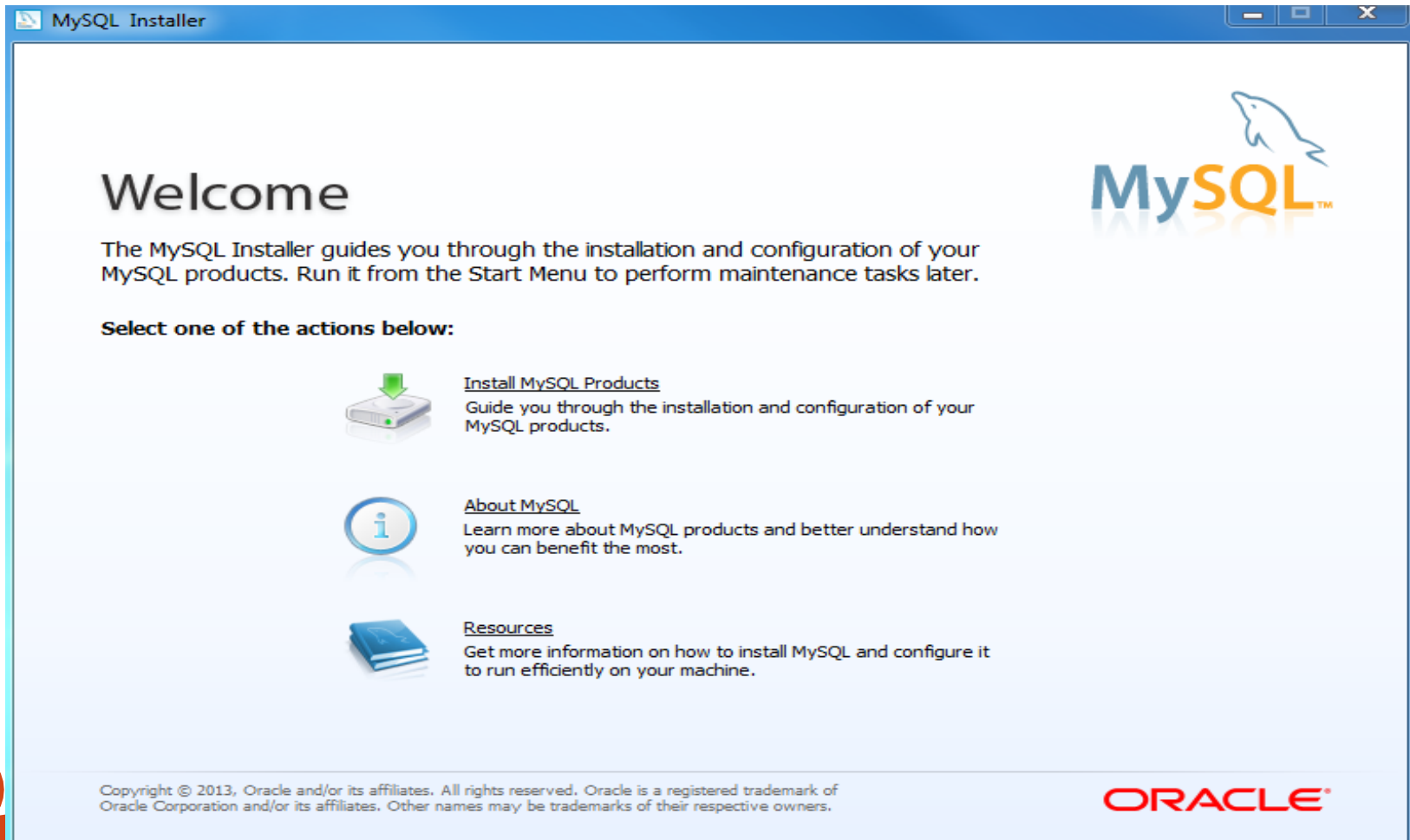
- Thực hiện cài đặt **MySQL Installer**



1.2.Cài đặt

Hướng dẫn cài đặt MySQL Installer :

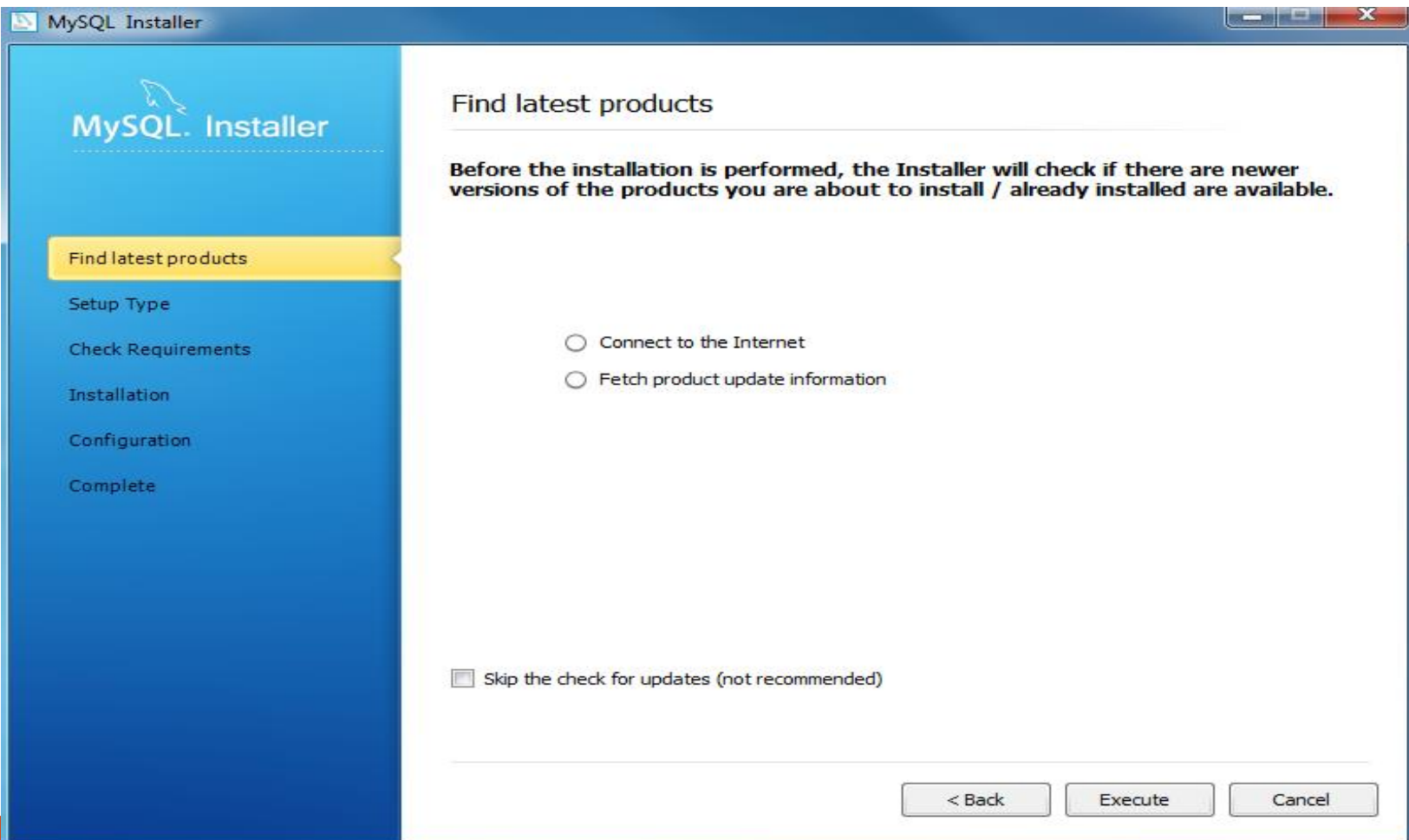
- Chọn Install MySQL Products



1.2.Cài đặt

Hướng dẫn cài đặt MySQL Installer :

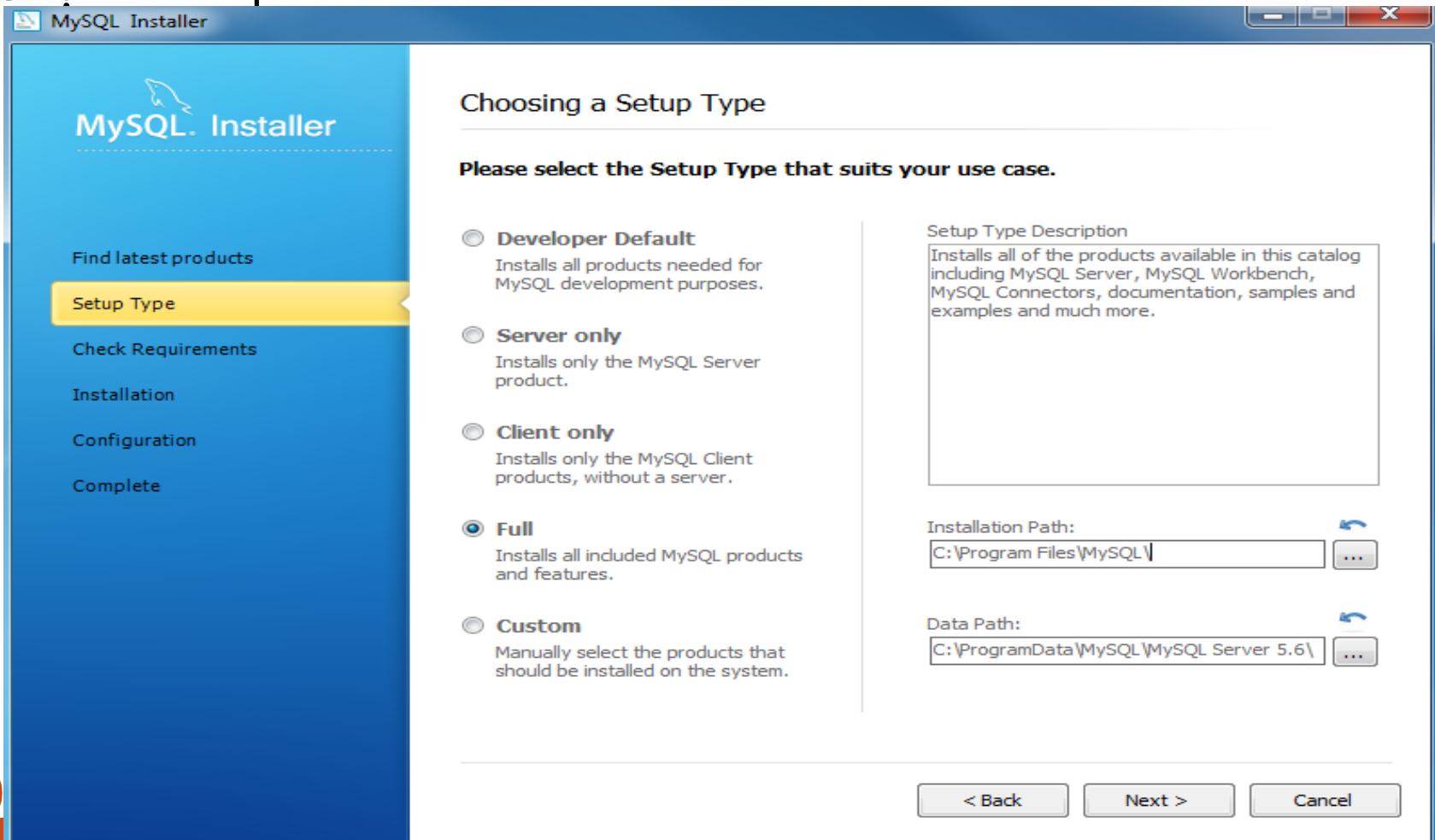
- Cập nhật tính năng mới khi cài đặt



1.2.Cài đặt

Hướng dẫn cài đặt MySQL Installer :

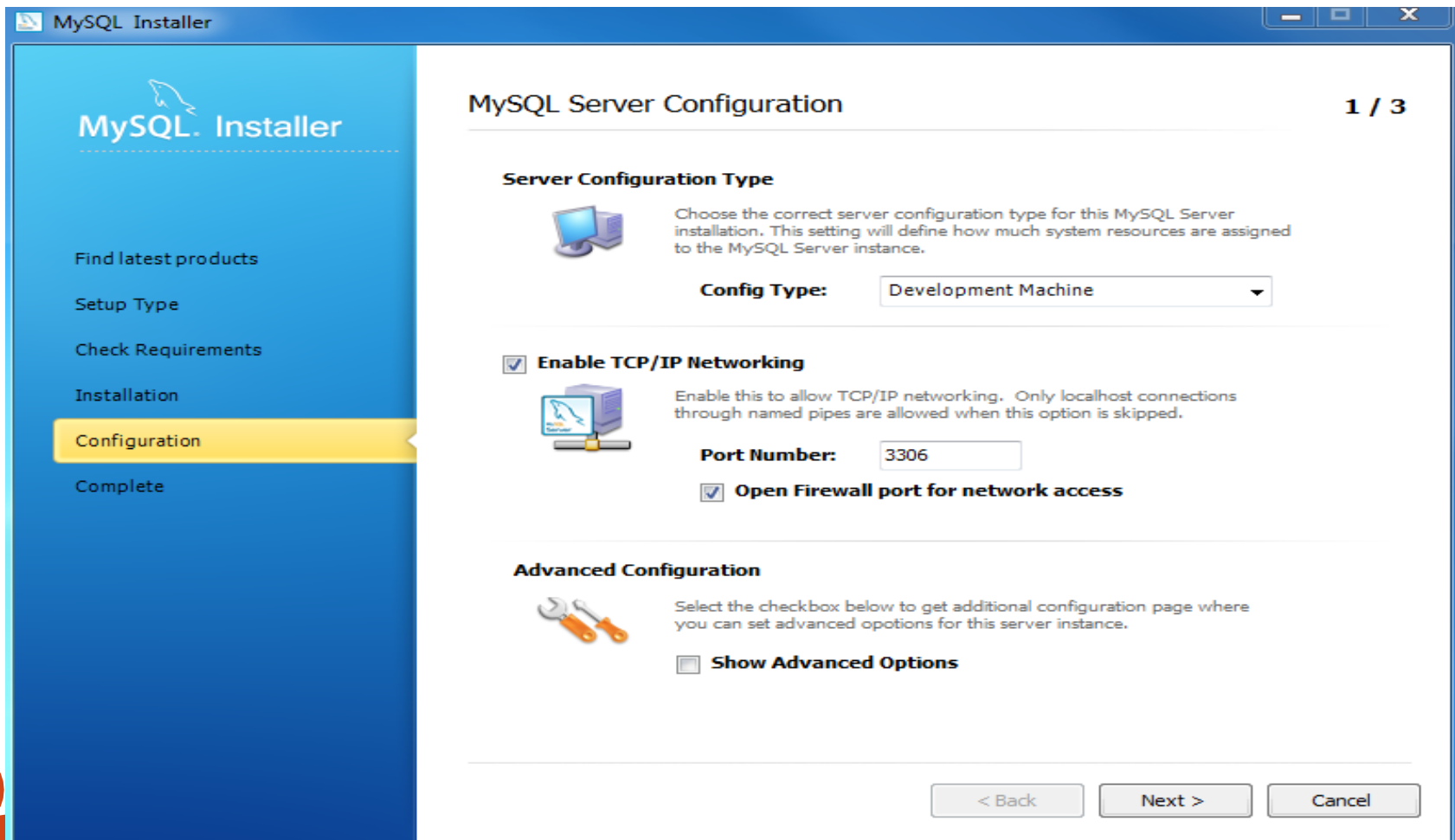
- Chọn Full option



1.2.Cài đặt

Hướng dẫn cài đặt MySQL Installer :

- Cấu hình



1.2.Cài đặt

Hướng dẫn cài đặt MySQL Installer :

- Thiết lập UserName và Password

MySQL Installer

MySQL Server Configuration 2 / 3

Root Account Password

Enter the password for the root account. Please remember to store this password in a secure place.

MySQL Root Password:

Repeat Password:

Password Strength: **Medium**

MySQL User Accounts

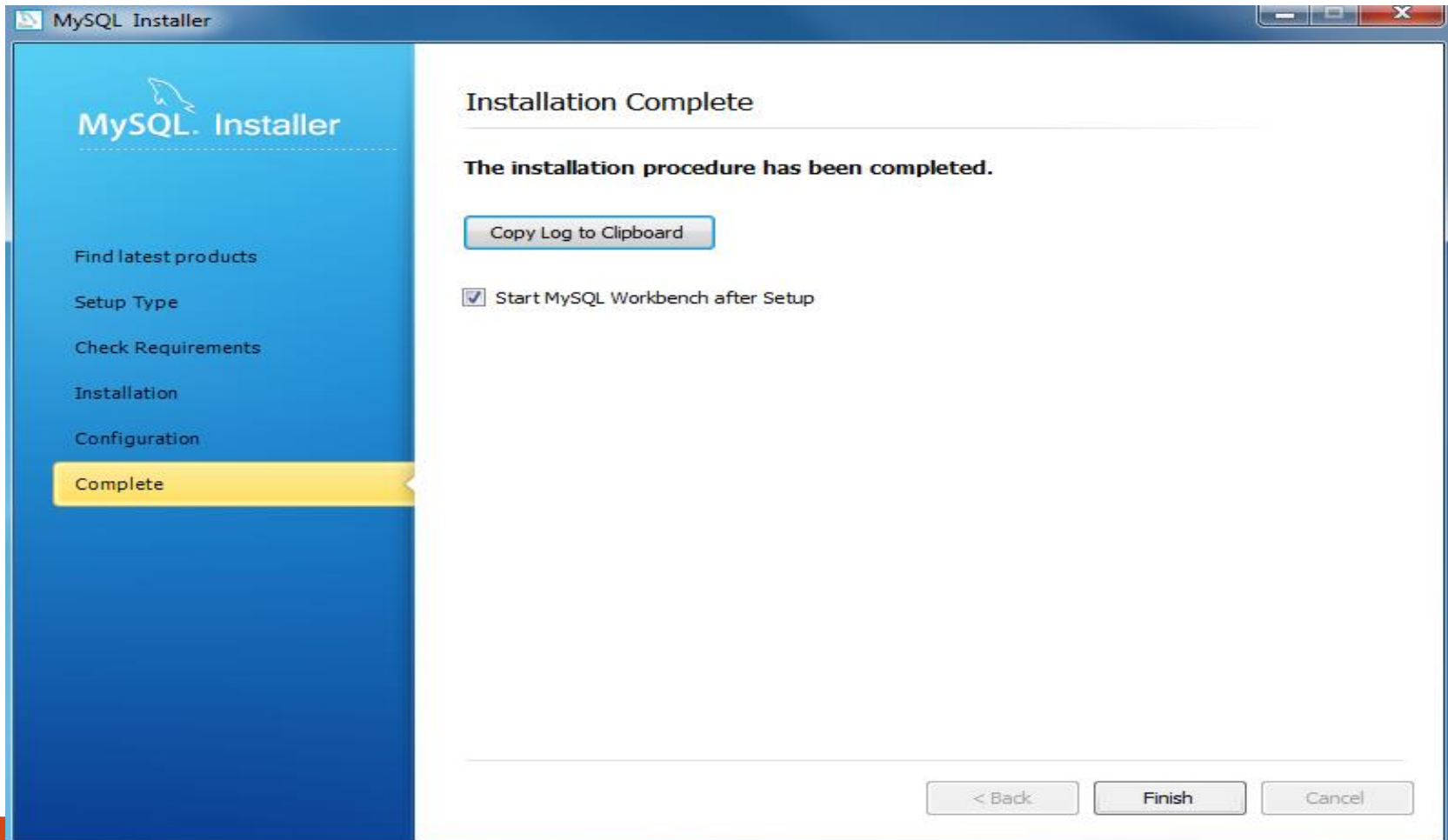
Create MySQL user accounts for your users and applications. Assign a role to the user that consists of a set of privileges.

MySQL Username	Host	User Role
----------------	------	-----------

1.2.Cài đặt

Hướng dẫn cài đặt MySQL Installer :

- Hoàn thành cài đặt



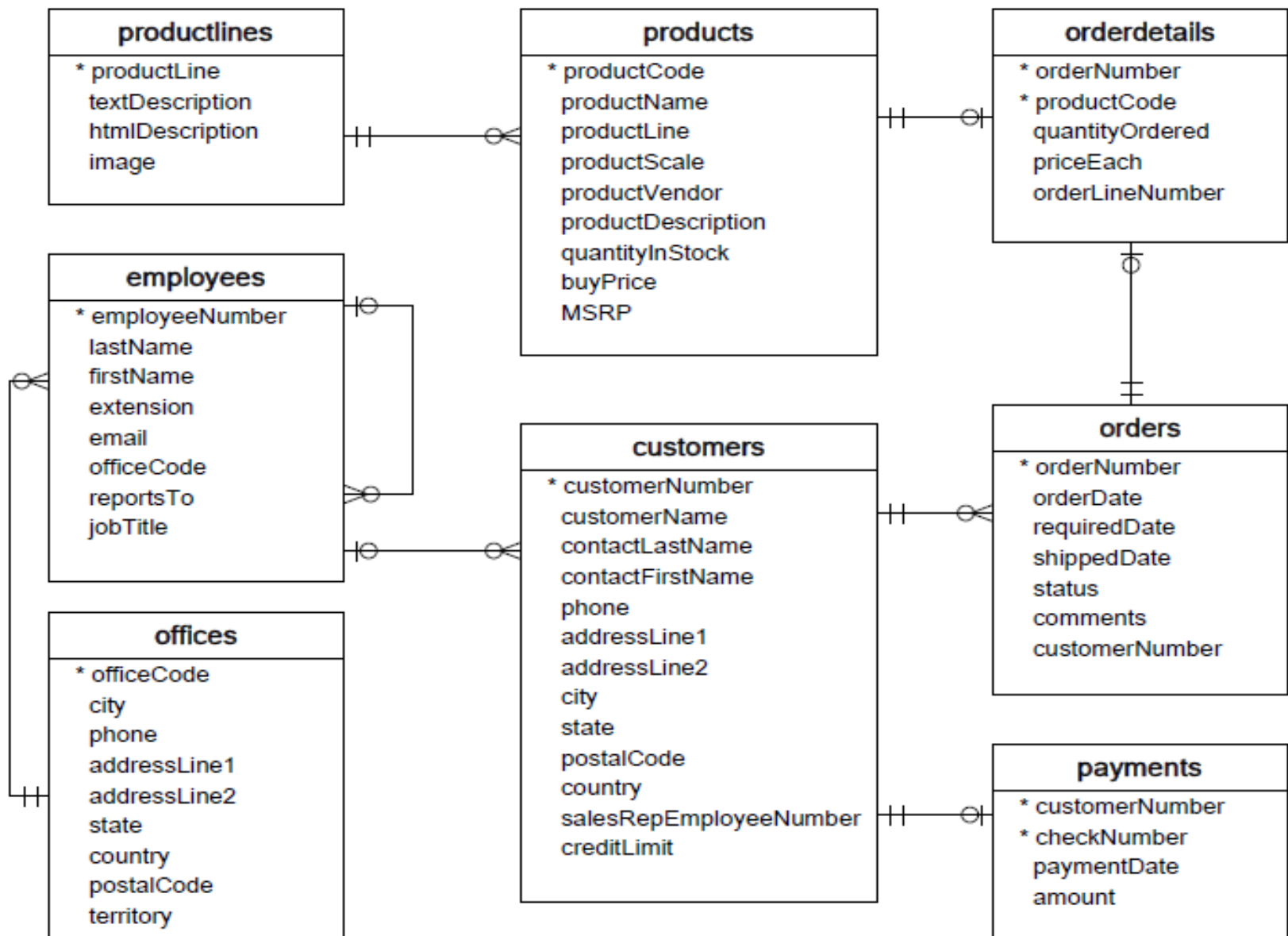
2. Sử dụng CSDL mẫu

1. Tải CSDL mẫu
2. Cài đặt csdl mẫu

2.1 Tải cơ sở dữ liệu mẫu

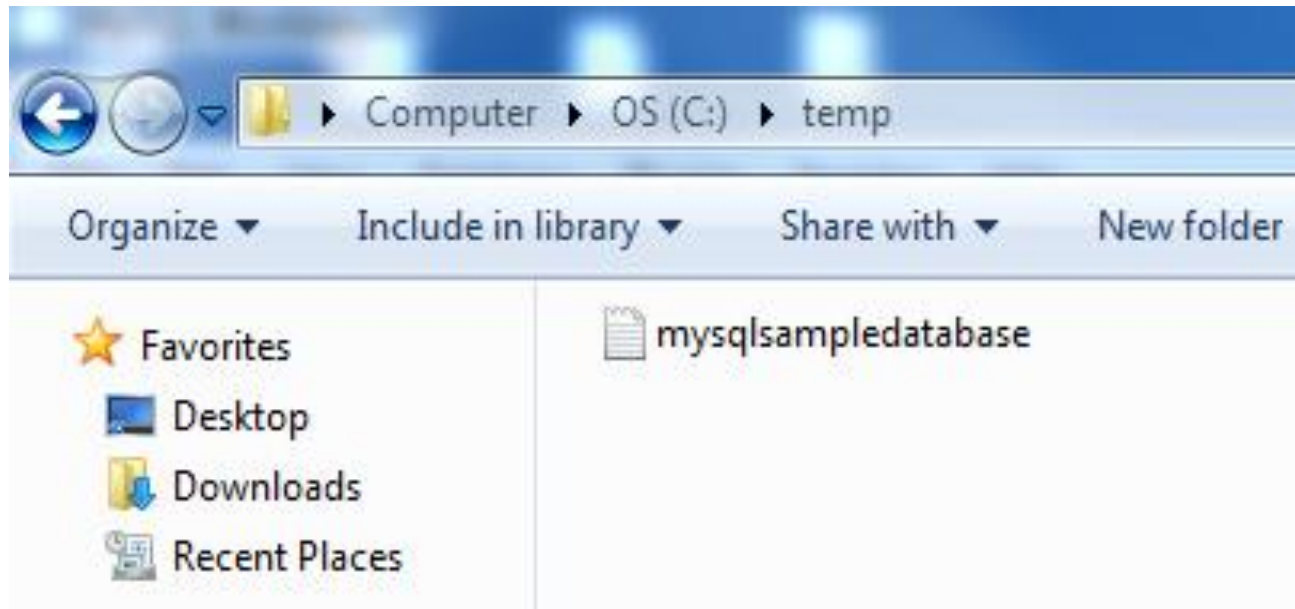
- Mysql cung cấp cơ sở dữ liệu mẫu 'classicmodels'
- Tải csdl mẫu tại: <http://www.mysqltutorial.org/wp-content/uploads/2018/03/mysqlsampledatabase.zip>.
- CSDL mẫu gồm các table: Customers, Products, ProductLines, Orders, OrderDetails, Payments, Employees, Offices.

2.1 Tải cơ sở dữ liệu mẫu



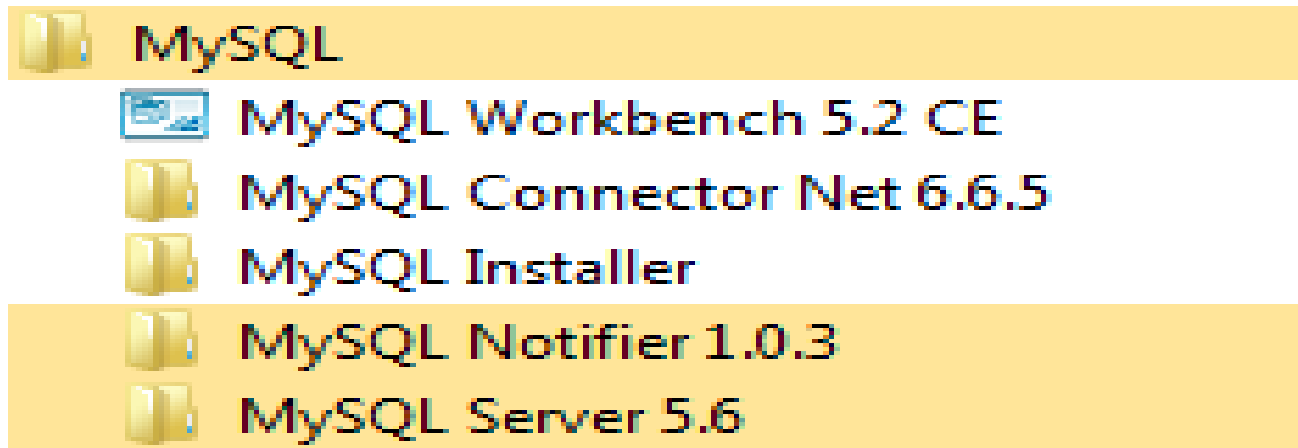
2.2 Cài đặt cơ sở dữ liệu mẫu

- Giải nén csdl mẫu tải về trong bước trước



2.2 Cài đặt cơ sở dữ liệu mẫu

- Chạy ứng dụng MySQL Workbench: Program Files > MySQL > MySQL Workbench



2.2 Cài đặt cơ sở dữ liệu mẫu

- Thêm mới database



2.2 Cài đặt cơ sở dữ liệu mẫu

- Cấu hình kết nối

Setup New Connection

Connection Name: Type a name for the connection

Connection Method: Method to use to connect to the RDBMS

Parameters **Advanced**

Hostname: Port: Name or IP address of the server host. - TCP/IP port.

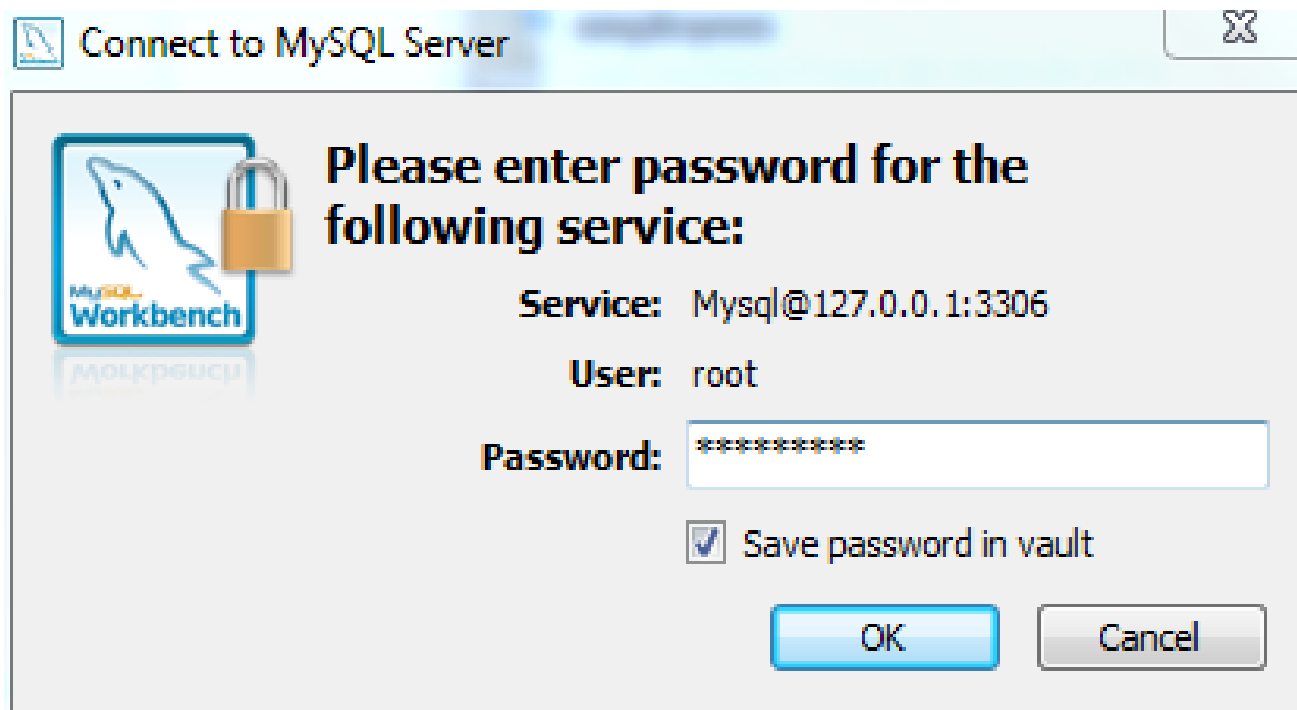
Username: Name of the user to connect with.

Password: The user's password. Will be requested later if it's not set.

Default Schema: The schema to use as default schema. Leave blank to select it later.

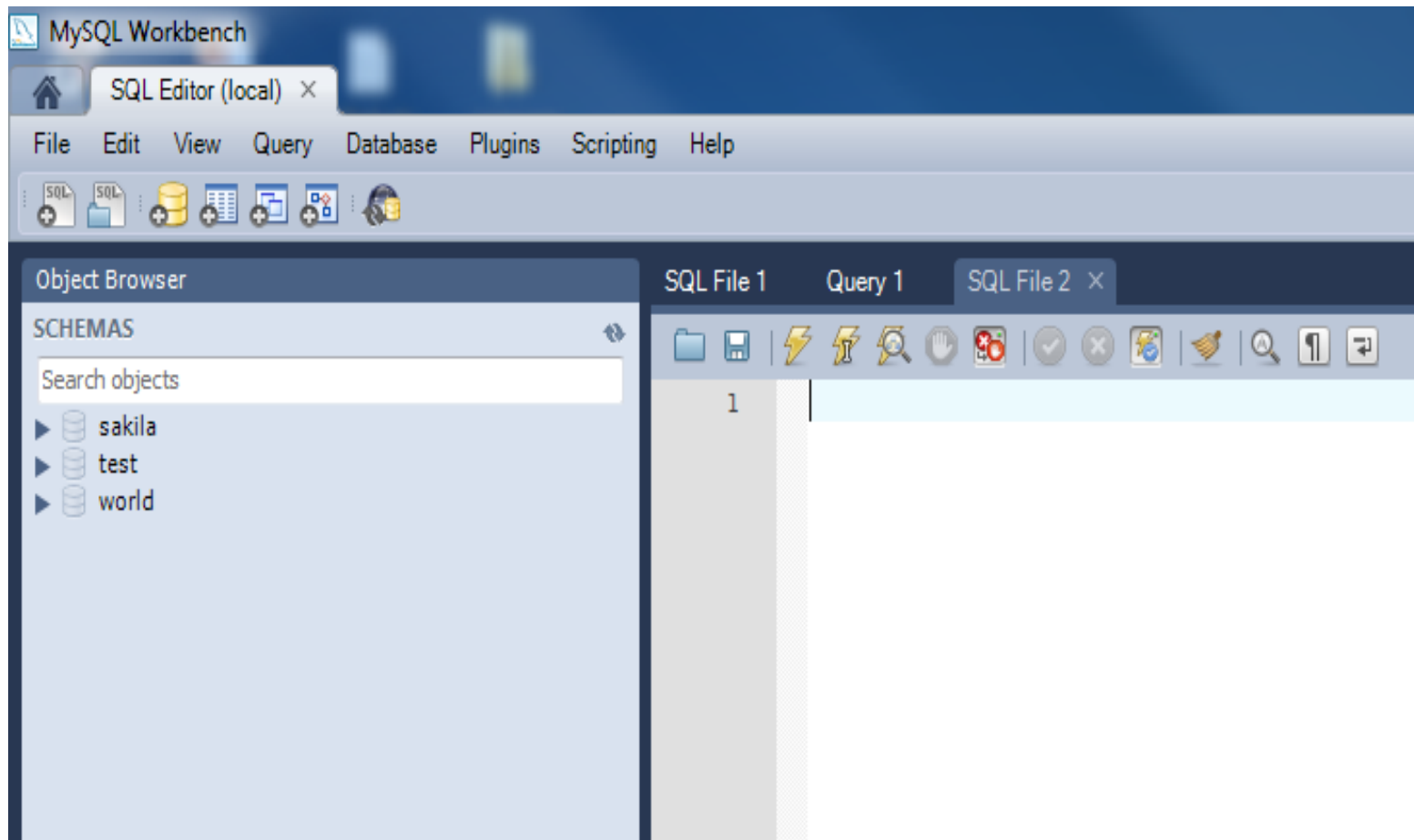
2.2 Cài đặt cơ sở dữ liệu mẫu

- Kết nối tới local database:



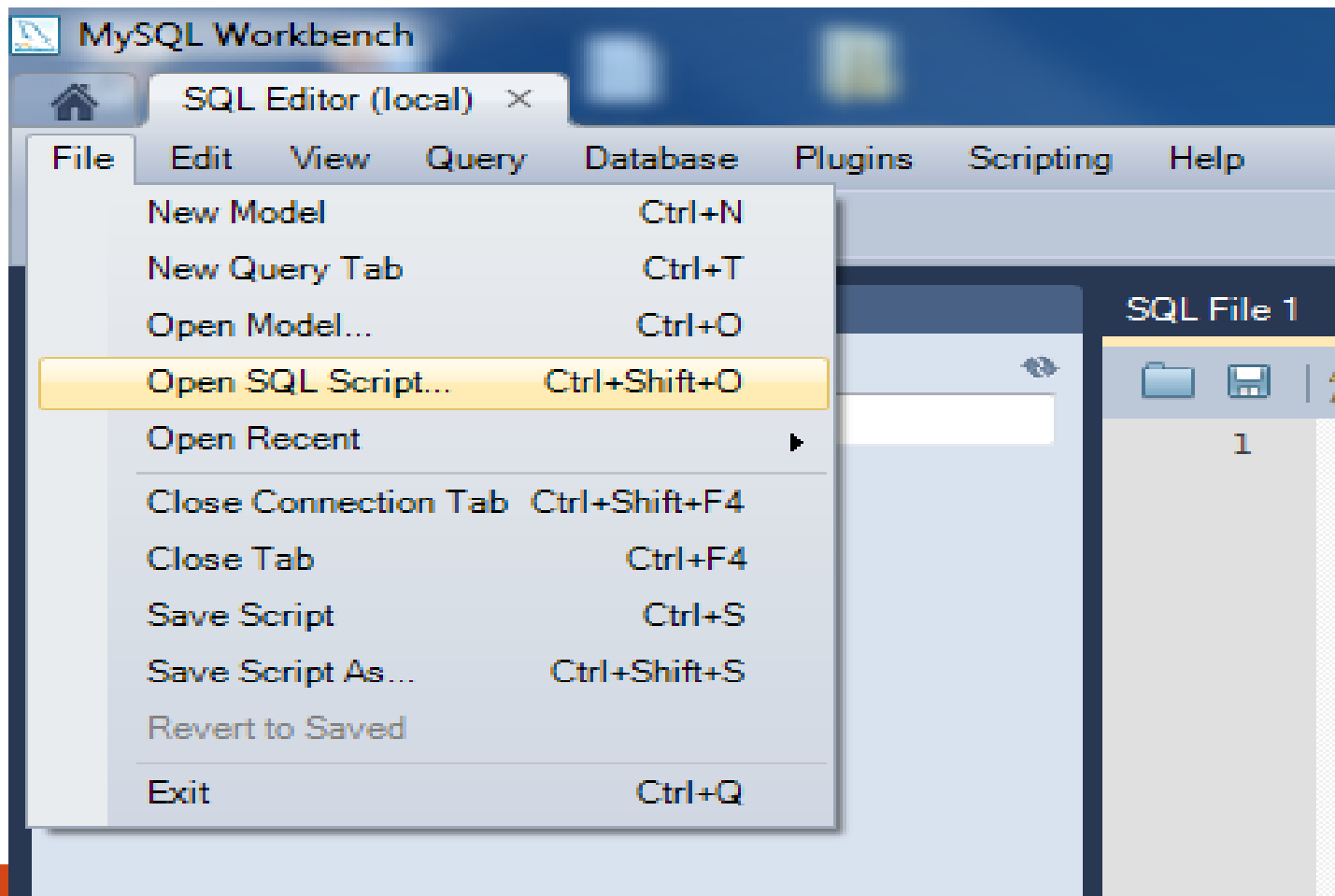
2.2 Cài đặt cơ sở dữ liệu mẫu

- Kết nối thành công:



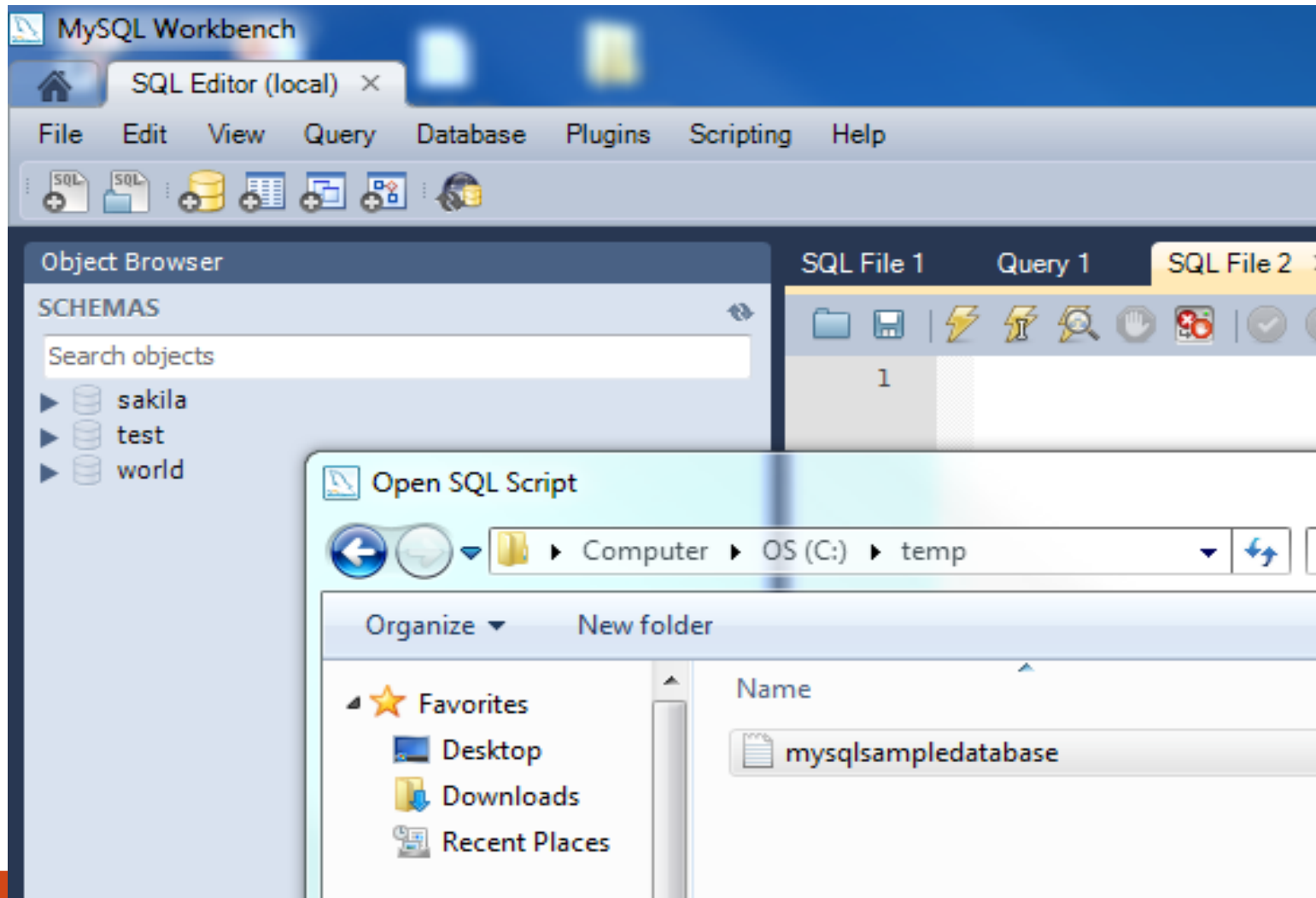
2.2 Cài đặt cơ sở dữ liệu mẫu

- Nhúng cơ sở dữ liệu mẫu vào Workbench: File > Open SQL Script (Ctrl+Shift+O)



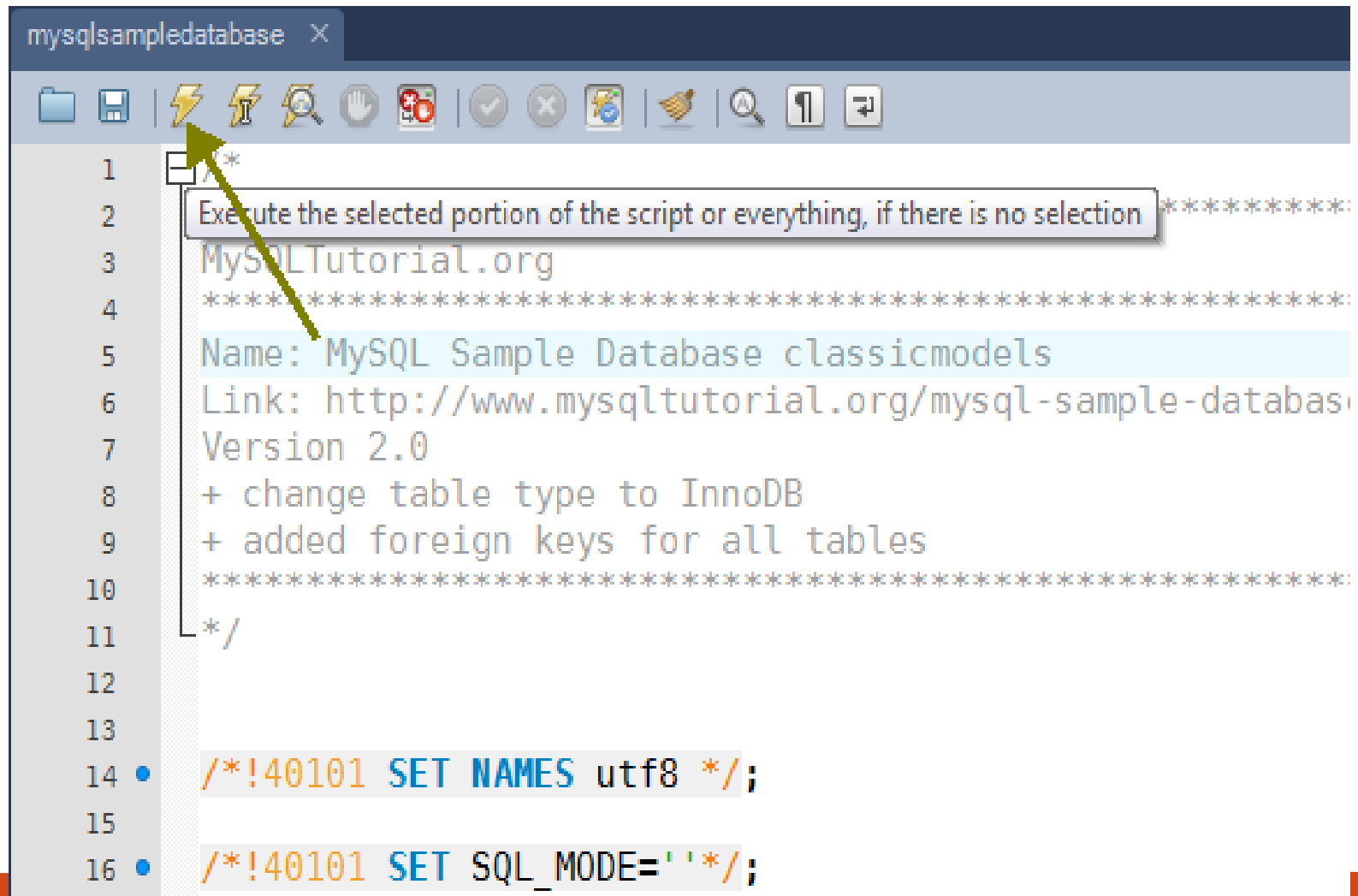
2.2 Cài đặt cơ sở dữ liệu mẫu

- Chọn đường dẫn đến csdl mẫu:



2.2 Cài đặt cơ sở dữ liệu mẫu

- Thực thi code csdl mẫu:

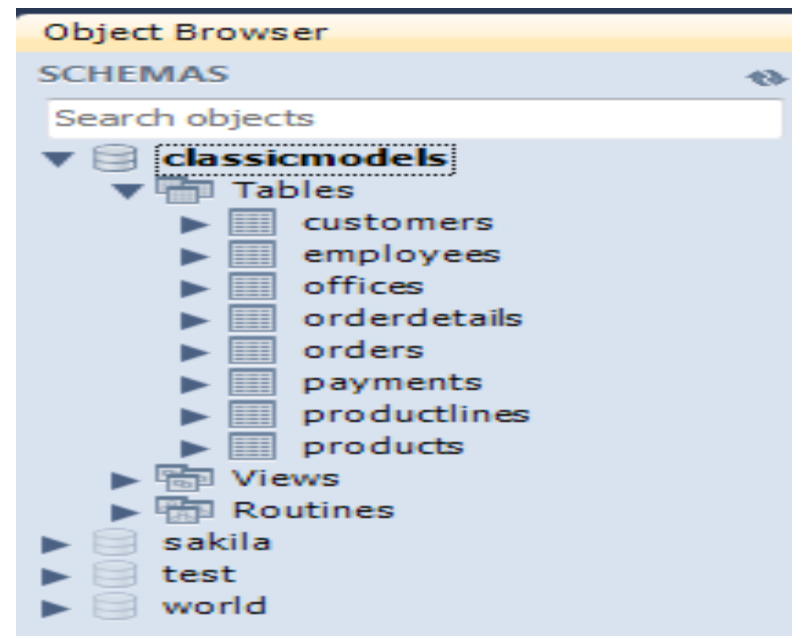
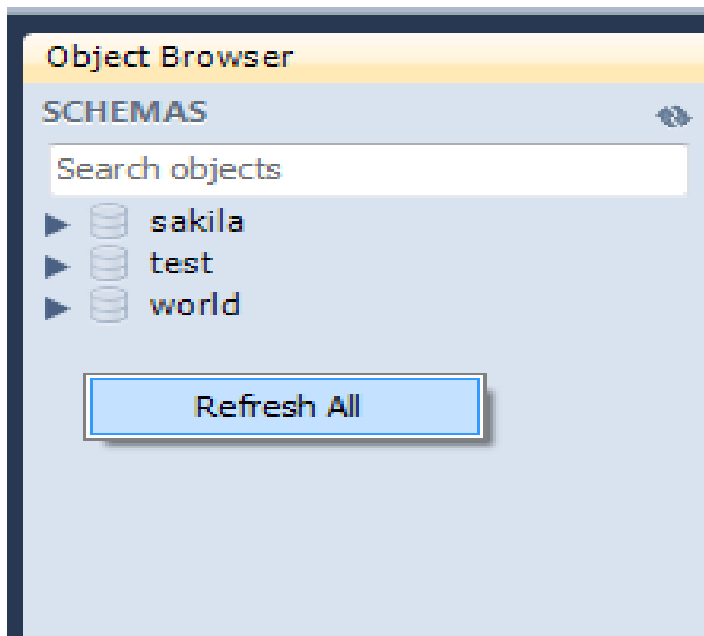


The screenshot shows a MySQL command-line interface window titled 'mysqlsampledatabase'. The window has a toolbar with various icons, including a lightning bolt icon for executing a query. A green arrow points to this icon, and a tooltip box appears over it with the text: 'Execute the selected portion of the script or everything, if there is no selection'. The main area of the window displays a script being executed, with line numbers 1 through 16 on the left. The script content is as follows:

```
1  /*
2  MySQLTutorial.org
3  ****
4  Name: MySQL Sample Database classicmodels
5  Link: http://www.mysqltutorial.org/mysql-sample-database
6  Version 2.0
7  + change table type to InnoDB
8  + added foreign keys for all tables
9  ****
10 */
11
12
13
14 • /*!40101 SET NAMES utf8 */;
15
16 • /*!40101 SET SQL_MODE=' */;
```

2.2 Cài đặt cơ sở dữ liệu mẫu

- Kiểm tra kết quả:



2.2 Demo những csdl mẫu

- Tải csdl mẫu
- Những csdl mẫu



3. Thao tác CSDL

1. Ngôn ngữ định nghĩa dữ liệu (DDL)
2. Ngôn ngữ thao tác dữ liệu (DML)

3.1 Ngôn ngữ định nghĩa dữ liệu (DDL)

- Tạo database

```
1 CREATE DATABASE [IF NOT EXISTS] database_name  
2 [CHARACTER SET charset_name]  
3 [COLLATE collation_name]
```

Ví dụ

Tạo một Database có tên là hocmysql

```
CREATE DATABASE hocmysql;
```

```
1 DROP DATABASE [IF EXISTS] database_name;
```

3.1 Ngôn ngữ định nghĩa dữ liệu (DDL)

- Các kiểu dữ liệu trong MySQL
 - Mỗi trường dữ liệu có kiểu xác định

HỌ TÊN	NGÀY SINH	GIỚI TÍNH	SỐ ĐIỆN THOẠI
Nguyễn Thanh Nhân	11-11-1989	Nam	9999999999

- Dữ liệu ở cột HỌ TÊN có dạng chuỗi ký tự, nên gọi cột HỌ TÊN có kiểu dữ liệu là kiểu chuỗi.
- Dữ liệu ở cột NGÀY SINH có dạng ngày tháng, nên cột NGÀY SINH có kiểu dữ liệu là kiểu ngày tháng
- Dữ liệu ở cột SỐ ĐIỆN THOẠI có dạng số nguyên, nên cột SỐ ĐIỆN THOẠI có kiểu dữ liệu là kiểu số nguyên

3.1 Ngôn ngữ định nghĩa dữ liệu (DDL)

- Kiểu số

Kiểu dữ liệu	Mô tả
TINYINT(size)	Lưu trữ một số nguyên có giá trị từ -128 đến -127 hoặc 0 đến 255
SMALLINT(size)	Lưu trữ một số nguyên có giá trị từ -32768 đến 32767 hoặc 0 đến 65535
MEDIUMINT(size)	Lưu trữ một số nguyên có giá trị từ -8388608 đến 8388607 hoặc 0 đến 16777215
INT(size)	Lưu trữ một số nguyên có giá trị từ -2147483648 đến 2147483647 hoặc 0 đến 4294967295
BIGINT(size)	Lưu trữ một số nguyên có giá trị từ -9223372036854775808 đến 9223372036854775807 hoặc 0 đến 18446744073709551615.
FLOAT(size,d)	Lưu trữ một số thập phân loại nhỏ (Ví dụ: 567.25). Tham số “size” dùng để xác định kích thước tối đa của phần nguyên (nằm bên trái dấu chấm). Tham số “d” dùng để xác định kích thước tối đa của phần thập phân (nằm bên phải dấu chấm).
DOUBLE(size,d)	Lưu trữ một số thập phân loại lớn. Tham số “size” dùng để xác định kích thước tối đa của phần nguyên (nằm bên trái dấu chấm). Tham số “d” dùng để xác định kích thước tối đa của phần thập phân (nằm bên phải dấu chấm).
DECIMAL(size,d)	Lưu trữ như một chuỗi, cho phép một dấu thập phân cố định. Tham số “size” dùng để xác định kích thước tối đa của phần nguyên (nằm bên trái dấu chấm). Tham số “d” dùng để xác định kích thước tối đa của phần thập phân (nằm bên phải dấu chấm).

3.1 Ngôn ngữ định nghĩa dữ liệu (DDL)

- Kiểu văn bản

Kiểu dữ liệu	Mô tả
CHAR(size)	Dùng để lưu trữ một chuỗi ký tự có chiều dài tối đa là 255 ký tự
VARCHAR(size)	Dùng để lưu trữ một chuỗi ký tự có chiều dài tối đa là 255 ký tự. Nếu đặt “size” lớn hơn 255 thì nó sẽ chuyển sang kiểu TEXT
TINYTEXT	Dùng để lưu trữ một chuỗi ký tự có chiều dài tối đa là 255 ký tự
TEXT	Dùng để lưu trữ một chuỗi ký tự có chiều dài tối đa là 65,535 ký tự
BLOB	Dùng để lưu trữ dữ liệu nhị phân tối đa là 65,535 byte
MEDIUMTEXT	Dùng để lưu trữ một chuỗi ký tự có chiều dài tối đa là 16,777,215 ký tự
MEDIUMBLOB	Dùng để lưu trữ dữ liệu nhị phân tối đa là 16,777,215 byte
LONGTEXT	Dùng để lưu trữ một chuỗi ký tự có chiều dài tối đa là 4,294,967,295 ký tự
LOBLOB	Dùng để lưu trữ dữ liệu nhị phân tối đa là 4,294,967,295 byte

3.1 Ngôn ngữ định nghĩa dữ liệu (DDL)

- Kiểu ngày tháng

Kiểu dữ liệu	Mô tả
DATE()	Lưu trữ một ngày theo định dạng YYYY-MM-DD (Ví dụ: 2016-09-12 tức là lưu ngày 12 tháng 9 năm 2016)
TIME()	Lưu trữ thời gian theo định dạng HH:MI:SS (Ví dụ 17:25:36 tức là lưu 17 giờ 25 phút 36 giây)
YEAR()	Lưu trữ một năm theo định dạng hai số hoặc bốn số
DATETIME()	Lưu trữ một ngày cùng với thời gian theo định dạng YYYY-MM-DD HH:MI:SS (Ví dụ: 2016-09-12 17:25:36 tức là lưu ngày 12 tháng 9 năm 2016 lúc 17 giờ 25 phút 36 giây)

3.1 Ngôn ngữ định nghĩa dữ liệu (DDL)

- Tạo bảng trong MySql
 - TABLE gồm nhiều cột, và nhiều hàng. Trong đó:
 - Mỗi hàng lưu trữ thông tin của một đối tượng
 - Mỗi cột là một thuộc tính của loại đối tượng đó

```
CREATE TABLE table_name
(
    column_name1 data_type(size),
    column_name2 data_type(size),
    column_name3 data_type(size),
    ....
);
```

```
CREATE TABLE sinh_vien
(
    Full_name VARCHAR(150),
    Gender VARCHAR(3),
    Age INT,
    City VARCHAR(50)
);
```

3.1 Ngôn ngữ định nghĩa dữ liệu (DDL)

- PRIMARY KEY (Khóa chính)
 - PRIMARY KEY được dùng để xác định mỗi mẫu tin (record) trong bảng là duy nhất. Từ đó nó giúp chúng ta phân biệt được các mẫu tin
 - Mỗi bảng chỉ có thể tồn tại một PRIMARY KEY
 - Giá trị PRIMARY KEY của từng mẫu tin phải là duy nhất (không được trùng nhau)

Full_name
Nguyễn Thành Nhân
Tân Thúc Bảo
Trình Giào Kim
Nguyễn Thành Nhân
La Thành
Nguyễn Thành Nhân

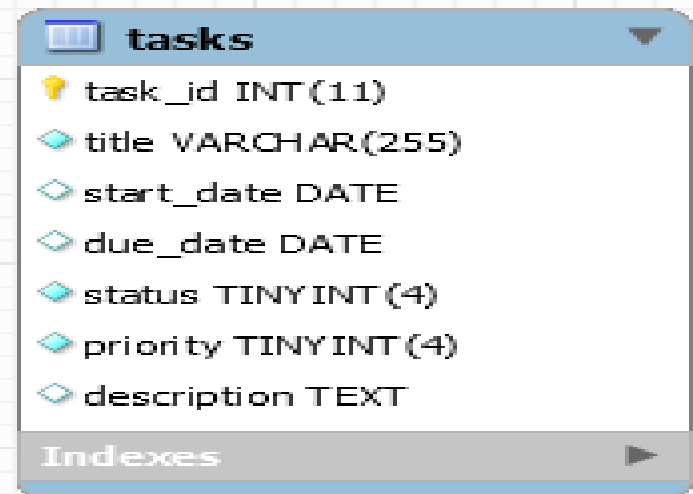


Mssv	Full_name
101	Nguyễn Thành Nhân
102	Tân Thúc Bảo
103	Trình Giào Kim
104	Nguyễn Thành Nhân
105	La Thành
106	Nguyễn Thành Nhân

3.1 Ngôn ngữ định nghĩa dữ liệu (DDL)

- Tạo bảng trong MySql với khóa chính

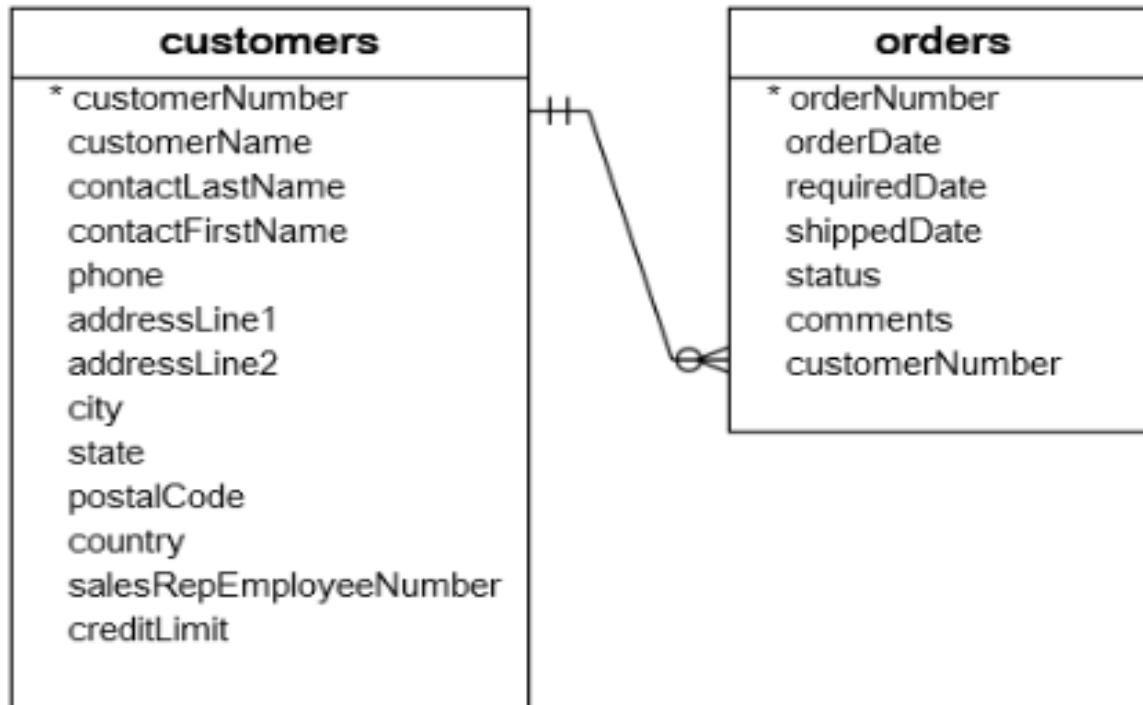
```
1 CREATE TABLE IF NOT EXISTS tasks (  
2     task_id INT AUTO_INCREMENT,  
3     title VARCHAR(255) NOT NULL,  
4     start_date DATE,  
5     due_date DATE,  
6     status TINYINT NOT NULL,  
7     priority TINYINT NOT NULL,  
8     description TEXT,  
9     PRIMARY KEY (task_id)  
10 ) ENGINE=INNODB;
```



3.1 Ngôn ngữ định nghĩa dữ liệu (DDL)

- Foreign key (khóa ngoại) : ràng buộc tính nhất quán dữ liệu giữa 2 table

```
CONSTRAINT constraint_name  
FOREIGN KEY foreign_key_name (columns)  
REFERENCES parent_table(columns)  
ON DELETE action  
ON UPDATE action
```



3.1 Ngôn ngữ định nghĩa dữ liệu (DDL)

- **Chỉnh sửa bảng trong MySql:** có thể thêm, xóa cột, thay đổi kiểu dữ liệu, thêm khóa chính, đổi tên bảng

```
1 ALTER TABLE table_name action1[,action2,...]
```

```
1 ALTER TABLE tasks  
2 RENAME TO work_items;
```

```
1 ALTER TABLE tasks  
2 CHANGE COLUMN task_id task_id INT(11) NOT NULL AUTO_INCREMENT;
```

```
1 ALTER TABLE tasks  
2 ADD COLUMN complete DECIMAL(2,1) NULL  
3 AFTER description;
```

```
1 ALTER TABLE tasks  
2 DROP COLUMN description;
```

3.1 Ngôn ngữ định nghĩa dữ liệu (DDL)

- Thêm dữ liệu vào bảng

Cú pháp 1

```
INSERT INTO table_name VALUES (value1,value2,value3);
```

Trong đó:

- table_name là tên của bảng mà bạn muốn chèn dữ liệu vào.
- value1, value2, value3 lần lượt là giá trị của cột 1, cột 2, cột 3 của bảng.

Lưu ý: Đối với cú pháp này, bạn phải đảm bảo điền đủ số value (Ví dụ: bảng có 6 cột, thì phải điền đủ 6 value). Nếu không, khi chạy lệnh sẽ bị lỗi.

```
INSERT INTO sinh_vien VALUES ("Trình Giao Kim","Nam","44","Bac Lieu");
```

3.1 Ngôn ngữ định nghĩa dữ liệu (DDL)

- Thêm dữ liệu vào bảng

Cú pháp 2

```
INSERT INTO table_name (column1,column2,column3)
VALUES (value1,value2,value3);
```

Trong đó:

- column1, column2, column3 lần lượt là tên của cột 1, cột 2, cột 3.
- value1, value2, value3 lần lượt là giá trị của cột 1, cột 2, cột 3.

Lưu ý: Đối với cú pháp này, bạn phải đảm bảo số lượng column phải bằng với số lượng value. Nếu không, khi chạy lệnh sẽ bị lỗi.

```
INSERT INTO sinh_vien (Full_name, Gender, Age, City)
VALUES ("Trinh Giao Kim", "Nam", "44", "Bac Lieu");
```

Lưu ý: Các tên cột như: Full_name, Gender, Age, City **KHÔNG** đặt trong cặp dấu móc đơn " hoặc móc kép ""

3.1. Ngôn ngữ định nghĩa dữ liệu (DDL)

- Tạo database
- Tạo table (khóa chính, khóa ngoại)
- Thêm dữ liệu vào table



3.2. Ngôn ngữ thao tác dữ liệu (DML)

- Truy xuất dữ liệu

```
SELECT name_column1, name_column2 FROM name_table;
```

```
1 SELECT
2     lastname, firstname, jobtitle
3 FROM
4     employees;
```

	employeeNum	lastName	firstName	extension	email	officeCode	reportsTo	jobTitle
►	1002	Murphy	Diane	x5800	dmurphy@classicmodelcars.com	1	NULL	President
	1056	Patterson	Mary	x4611	mpatterso@classicmodelcars.com	1	1002	VP Sales
	1076	Firelli	Jeff	x9273	jfirelli@classicmodelcars.com	1	1002	VP Marketing
	1088	Patterson	William	x4871	wpatterson@classicmodelcars.com	6	1056	Sales Manager (APAC)
	1102	Bondur	Gerard	x5408	gbondur@classicmodelcars.com	4	1056	Sale Manager (EMEA)
	1143	Bow	Anthony	x5428	abow@classicmodelcars.com	1	1056	Sales Manager (NA)
	1165	Jennings	Leslie	x3291	ljennings@classicmodelcars.com	1	1143	Sales Rep
	1166	Thompson	Leslie	x4065	lthompson@classicmodelcars.com	1	1143	Sales Rep
	1188	Firelli	Julie	x2173	jfirelli@classicmodelcars.com	2	1143	Sales Rep
	1216	Patterson	Steve	x4334	spatterson@classicmodelcars.com	2	1143	Sales Rep
	1286	Tsang	Foon Yue	x2248	ftsang@classicmodelcars.com	3	1143	Sales Rep

3.2. Ngôn ngữ thao tác dữ liệu (DML)

- Truy xuất dữ liệu có điều kiện

Cú pháp

```
SELECT *  
FROM name_table  
WHERE name_column operator value_column;
```

Giải thích: Cú pháp trên truy xuất tất cả dữ liệu của các cột trong bảng có tên là name_table với điều kiện là giá trị của cột name_column phải đúng với value_column dựa theo điều kiện so sánh operator.

3.2. Ngôn ngữ thao tác dữ liệu (DML)

- Truy xuất dữ liệu có điều kiện

=	So sánh hai giá trị bằng nhau
<>	So sánh hai giá trị khác nhau
!=	So sánh hai giá trị khác nhau
>	Giá trị bên trái lớn hơn giá trị bên phải
<	Giá trị bên trái nhỏ hơn giá trị bên phải
>=	Giá trị bên trái lớn hơn hoặc bằng giá trị bên phải
<=	Giá trị bên trái nhỏ hơn hoặc bằng giá trị bên phải
BETWEEN	Giá trị nằm trong một khoảng nào đó (Sẽ nói rõ trong những bài hướng dẫn tiếp theo)
LIKE	Dùng trong tìm kiếm chuỗi ký tự (Sẽ nói rõ trong những bài hướng dẫn tiếp theo)
IN	Giá trị là một trong số các giá trị được nêu (Sẽ nói rõ trong những bài hướng dẫn tiếp theo)

3.2. Ngôn ngữ thao tác dữ liệu (DML)

- Truy xuất dữ liệu có điều kiện

```
SELECT
    lastname,
    firstname,
    jobtitle
FROM
    employees
WHERE
    jobtitle = 'Sales Rep';
```

employeeNumber	lastName	firstName	extension	email	officeCode	reportsTo	job Title
1002	Murphy	Diane	x5800	dmurphy@classicmodelcars.com	1	NULL	President
1056	Patterson	Mary	x4611	mpatterso@classicmodelcars.com	1	1002	VP Sales
1076	Firelli	Jeff	x9273	jfirelli@classicmodelcars.com	1	1002	VP Marketing
1088	Patterson	William	x4871	wpatterson@classicmodelcars.com	6	1056	Sales Manager (APAC)
1102	Bondur	Gerard	x5408	gbondur@classicmodelcars.com	4	1056	Sale Manager (EMEA)
1143	Bow	Anthony	x5428	abow@classicmodelcars.com	1	1056	Sales Manager (NA)
1165	Jennings	Leslie	x3291	ljennings@classicmodelcars.com	1	1143	Sales Rep
1166	Thompson	Leslie	x4065	lthompson@classicmodelcars.com	1	1143	Sales Rep
1188	Firelli	Julie	x2173	jfirelli@classicmodelcars.com	2	1143	Sales Rep
1216	Patterson	Steve	x4334	spatterson@classicmodelcars.com	2	1143	Sales Rep

3.2. Ngôn ngữ thao tác dữ liệu (DML)

- Truy xuất dữ liệu có điều kiện

```
SELECT *  
FROM name_table  
WHERE condition1 AND condition2 AND condition3;
```

Trong đó: condition là một biểu thức điều kiện(Ví dụ: Age > 22)

```
SELECT *  
FROM name_table  
WHERE condition1 OR condition2 OR condition3;
```

Trong đó: condition là một biểu thức điều kiện(Ví dụ: Age > 22)

3.2. Ngôn ngữ thao tác dữ liệu (DML)

- Truy xuất dữ liệu có điều kiện

```
SELECT Full_name, Gender
FROM sinh_vien
WHERE Gender="Nam" AND City="Can Tho";
```

```
SELECT *
FROM sinh_vien
WHERE City="Can Tho" OR City="Vinh Long" OR City="Soc Trang";
```

```
SELECT Full_name
FROM sinh_vien
WHERE (Gender="Nam" AND City="Can Tho") OR (Gender="Nu" AND City="Soc Trang");
```

3.2. Ngôn ngữ thao tác dữ liệu (DML)

- Truy xuất dữ liệu có gom theo nhóm

```
SELECT
    c1, c2, ..., cn, aggregate_function(ci)
FROM
    table
WHERE
    where_conditions
GROUP BY c1 , c2, ...,cn;
```

```
SELECT
    status
FROM
    orders
GROUP BY status;
```

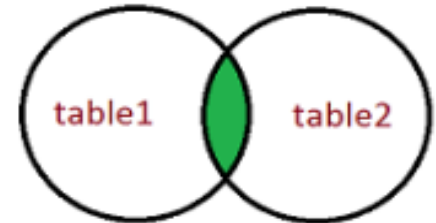
```
SELECT
    status, COUNT(*)
FROM
    orders
GROUP BY status;
```

3.2. Ngôn ngữ thao tác dữ liệu (DML)

- Kết nối các bảng

```
SELECT *  
FROM table1  
INNER JOIN table2  
ON table1.column_name=table2.column_name;
```

INNER JOIN



```
SELECT hoadon.ID_hoadon, khachhang.NAME_khachhang  
FROM hoadon  
INNER JOIN khachhang  
ON hoadon.ID_khachhang=khachhang.ID_khachhang;
```

3.2. Ngôn ngữ thao tác dữ liệu (DML)

- Truy vấn đơn giản
- Truy vấn có điều kiện
- Truy vấn từ nhiều table
- Truy vấn gom nhóm

