

342236 WEB APPLICATION PROGRAMMING

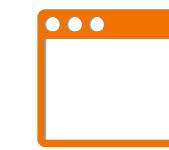
การเขียนโปรแกรมประยุกต์บนเว็บ

04 - PHP and MySQL

อ.ดร.วรวิทยา วรรณศรี

หลักสูตรเทคโนโลยีสารสนเทศ สาขาวิชาการคอมพิวเตอร์ วิทยาลัยการคอมพิวเตอร์ มหาวิทยาลัยขอนแก่น

Outline



ระบบจัดการฐานข้อมูล MySQL



เชื่อมต่อกับ MySQL ด้วยภาษา PHP

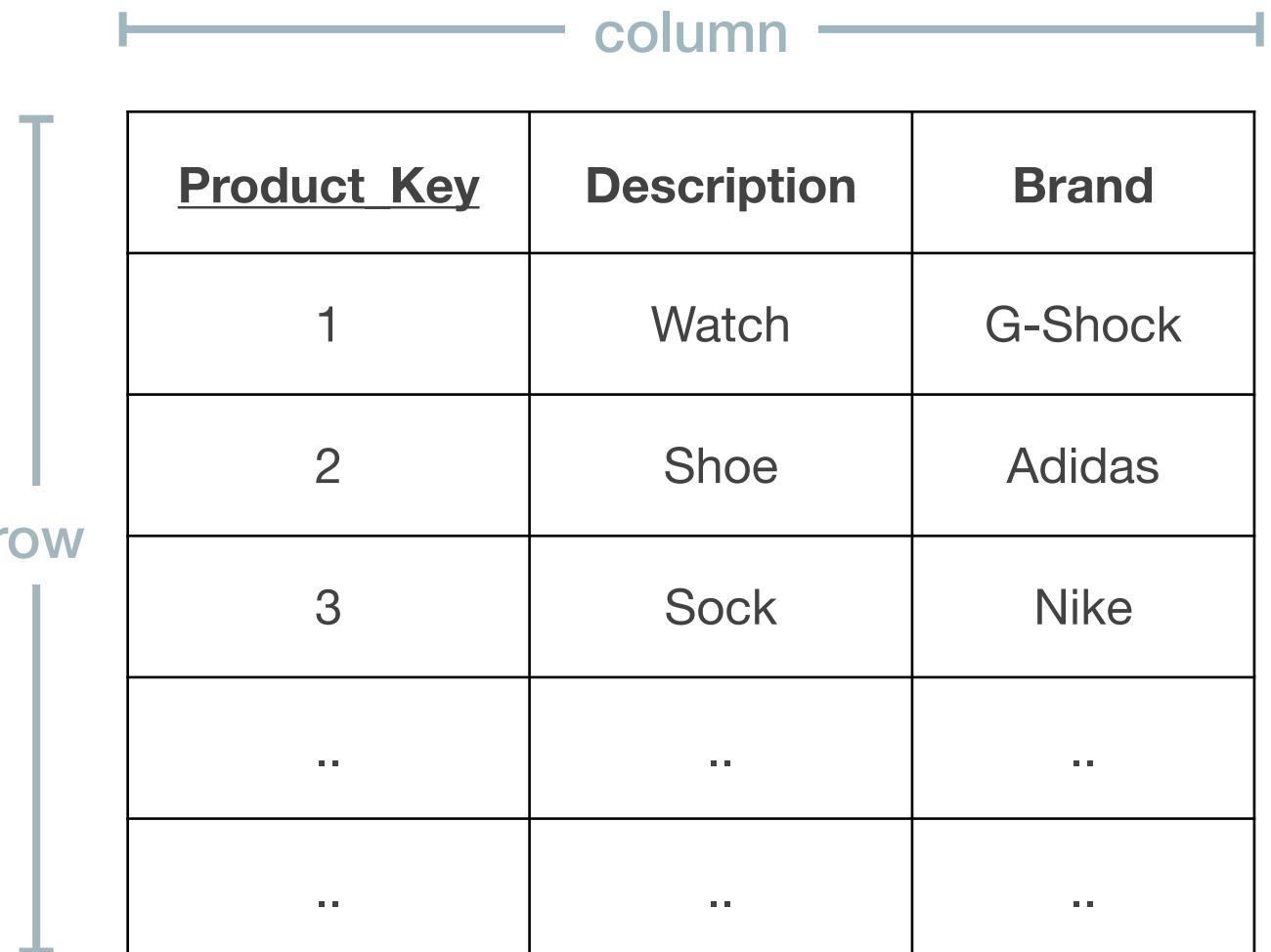
ระบบจัดการฐานข้อมูล MySQL



- ▶ เป็นระบบจัดการฐานข้อมูลเชิงล้มเหลว (Relational Database Management System: RDBMS)
- ▶ เป็น Open Source Software ที่สามารถดาวน์โหลด และนำมาใช้ได้ฟรีตามข้อกำหนดของสิทธิบัตร GPL (GNU General Public License) ปัจจุบันอยู่ภายในการสนับสนุนของบริษัท **ORACLE**®
- ▶ โดยปกติฐานข้อมูลเชิงล้มเหลวมีหน้าที่จัดเก็บข้อมูลที่อยู่ในรูปแบบตารางอย่างเป็นระบบ การเข้าถึงและการจัดการข้อมูลจะทำผ่าน RDBMS
- ▶ สนับสนุนการใช้งานบนหลายระบบปฏิบัติการ ไม่ว่าจะเป็น Unix, MacOS หรือ Microsoft Windows
- ▶ นอกจากนี้ MySQL ยังเป็นที่นิยมในการใช้งานร่วมกับงานทางด้านการพัฒนาเว็บ และการพัฒนาซอฟต์แวร์ต่าง ๆ

RECAP: ฐานข้อมูลเชิงสัมพันธ์

- ▶ ข้อมูลมีรูปแบบอยู่ในลักษณะของโครงสร้างตาราง
- ▶ มีการกำหนดชนิดของข้อมูลในแต่ละ column, attribute มาแล้ว
- ▶ ตารางข้อมูลสามารถเชื่อมโยงกันผ่าน unique keys
- ▶ ใช้ภาษา SQL (Structure Query language) ในการทำงานกับข้อมูล

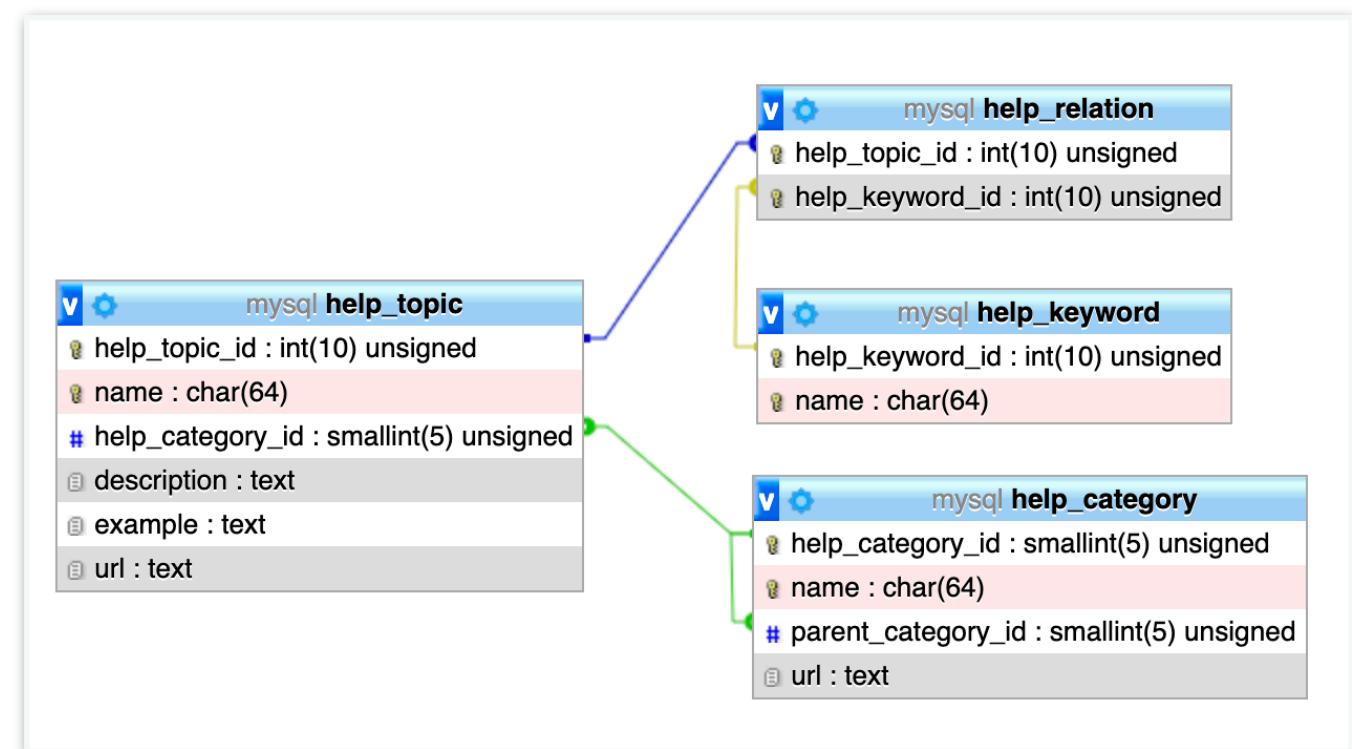


Product Key	Description	Brand
1	Watch	G-Shock
2	Shoe	Adidas
3	Sock	Nike
..
..

Schema

#	Name	Type	Collation	Attributes
1	id	int(5)		UNSIGNED
2	db_name	varchar(64)	utf8_bin	
3	table_name	varchar(64)	utf8_bin	
4	column_name	varchar(64)	utf8_bin	
5	comment	varchar(255)	utf8_general_ci	
6	mimetype	varchar(255)	utf8_general_ci	
7	transformation	varchar(255)	utf8_bin	
8	transformation_options	varchar(255)	utf8_bin	
9	input_transformation	varchar(255)	utf8_bin	
10	input_transformation_options	varchar(255)	utf8_bin	

Relation



MySQL®



PostgreSQL

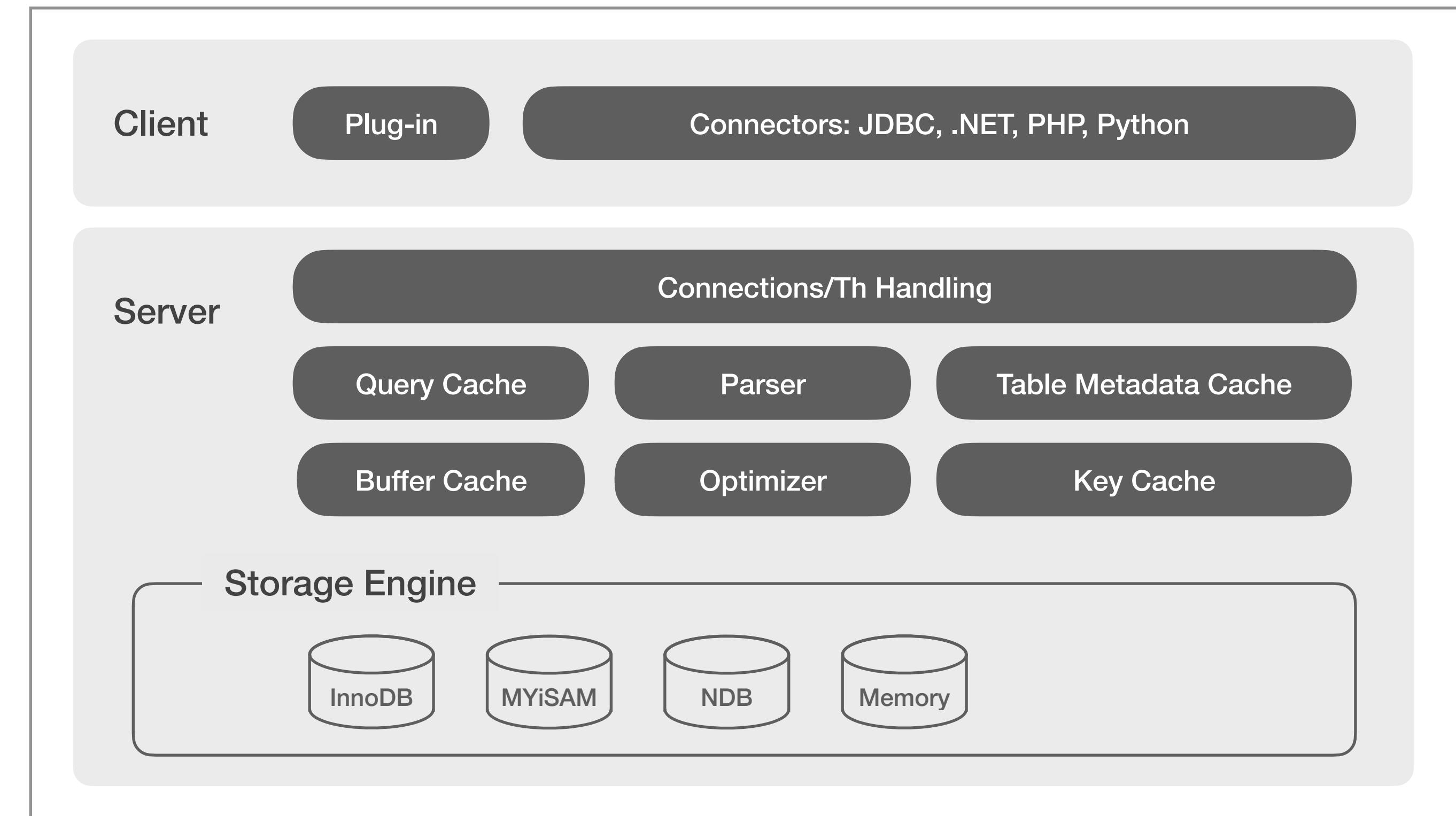


Microsoft
SQL Server®

ORACLE®

ระบบจัดการฐานข้อมูล MySQL

- ▶ สถาปัตยกรรมของ MySQL เป็นรูปแบบ Client-Server Architecture
- ▶ Client สามารถเข้าถึงข้อมูลผ่านเครื่องคอมพิวเตอร์ โดยมี Server ให้บริการผ่านเครือข่าย
- ▶ แบ่งออกเป็น 3 layers หลัก คือ Client Server และ Storage



phpMyAdmin

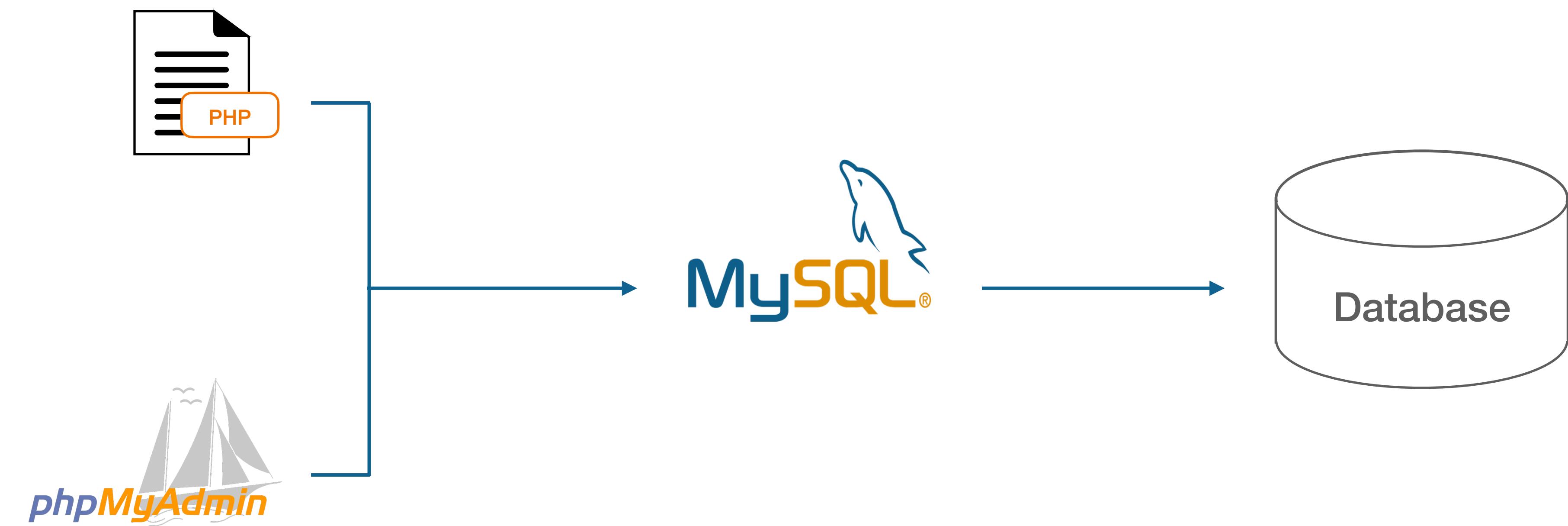


- ▶ เป็นโปรแกรมประเภท MySQL Client ที่ใช้สำหรับใช้งาน MySQL ผ่านเว็บเบราว์เซอร์
- ▶ เป็น PHP Application ที่ทำงานบนเว็บเซิร์ฟเวอร์
- ▶ เป็นโปรแกรม Open Source

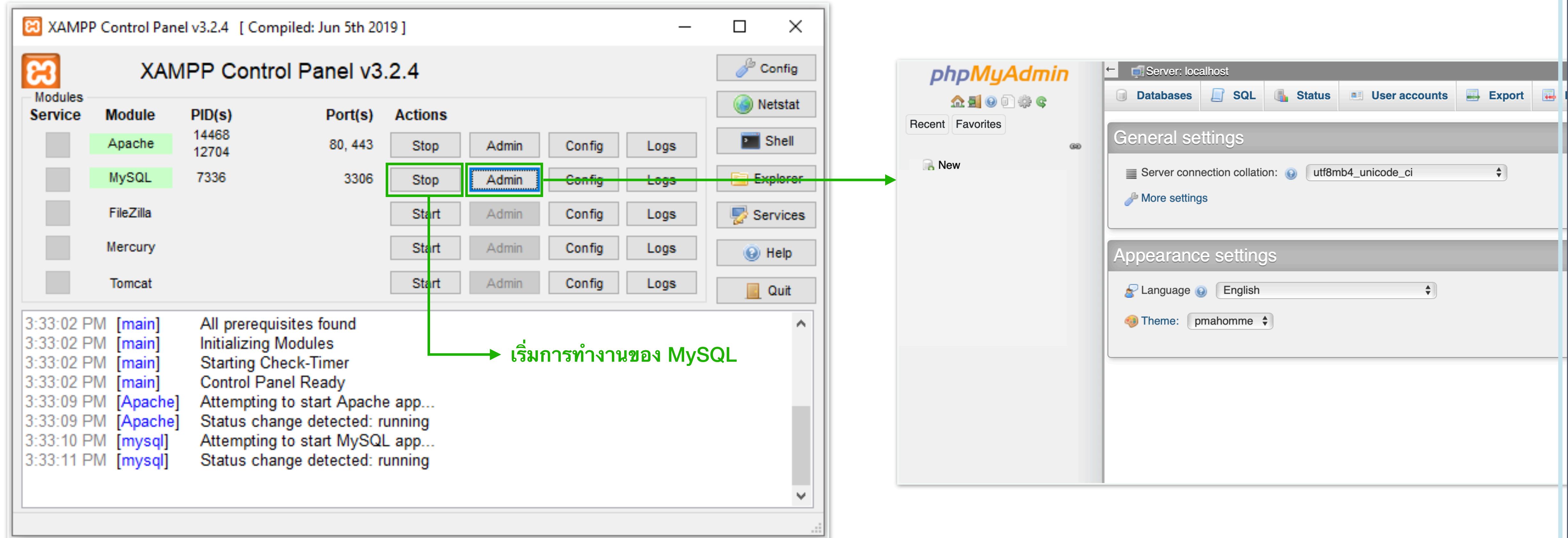
The screenshot shows the phpMyAdmin configuration page for a MySQL server running on localhost. The top navigation bar includes links for Databases, SQL, Status, User accounts, Export, Import, Settings, Replication, Variables, and More. The main content area is divided into several sections:

- General settings:** Shows the server connection collation set to utf8mb4_unicode_ci.
- Appearance settings:** Shows the language set to English and the theme set to pmahomme.
- Database server:** Displays information about the server, including its type (localhost via UNIX socket), version (Protocol version: 10), and character set (UTF-8 Unicode (utf8mb4)).
- Web server:** Displays the Apache version (Apache/2.4.43) and the database client version (libmysql - mysqlnd 7.4.8).

การเชื่อมต่อกับ MySQL



การเชื่อมต่อกับ MySQL



การสร้างฐานข้อมูล



phpMyAdmin

Server: localhost

Databases SQL Status User accounts Export Import Settings

New Create database

Database name: utf8mb4_general_ci

Create

	Database	Collation	Action
		utf8mb4_general_ci	Check privileges
		utf8_general_ci	Check privileges
		utf8_general_ci	Check privileges
		utf8mb4_general_ci	Check privileges

A green arrow points from the 'New' button on the left to the 'Create database' link on the right.

การสร้างฐานข้อมูล

- ▶ สร้างฐานข้อมูลชื่อ webApp

SQL Code

```
CREATE DATABASE webApp  
CHARACTER SET utf8  
COLLATE utf8_general_ci;
```

The screenshot shows the MySQL Workbench interface. In the top navigation bar, the 'SQL' tab is selected. Below it, the main area contains the SQL query:

```
1 CREATE DATABASE webApp  
2 CHARACTER SET utf8  
3 COLLATE utf8_general_ci;
```

Below the query, there are several buttons: 'Clear', 'Format', 'Get auto-saved query', and 'Bind parameters'. There is also a 'Bookmark this SQL query:' field and a 'Delimiter' dropdown set to a semicolon. At the bottom, there are checkboxes for 'Show this query here again', 'Retain query box', 'Rollback when finished', and 'Enable foreign key checks', followed by a 'Go' button.



The screenshot shows the MySQL Workbench interface after the query has been run. The top navigation bar is identical to the previous screenshot. The main area displays the result message:

Show query box

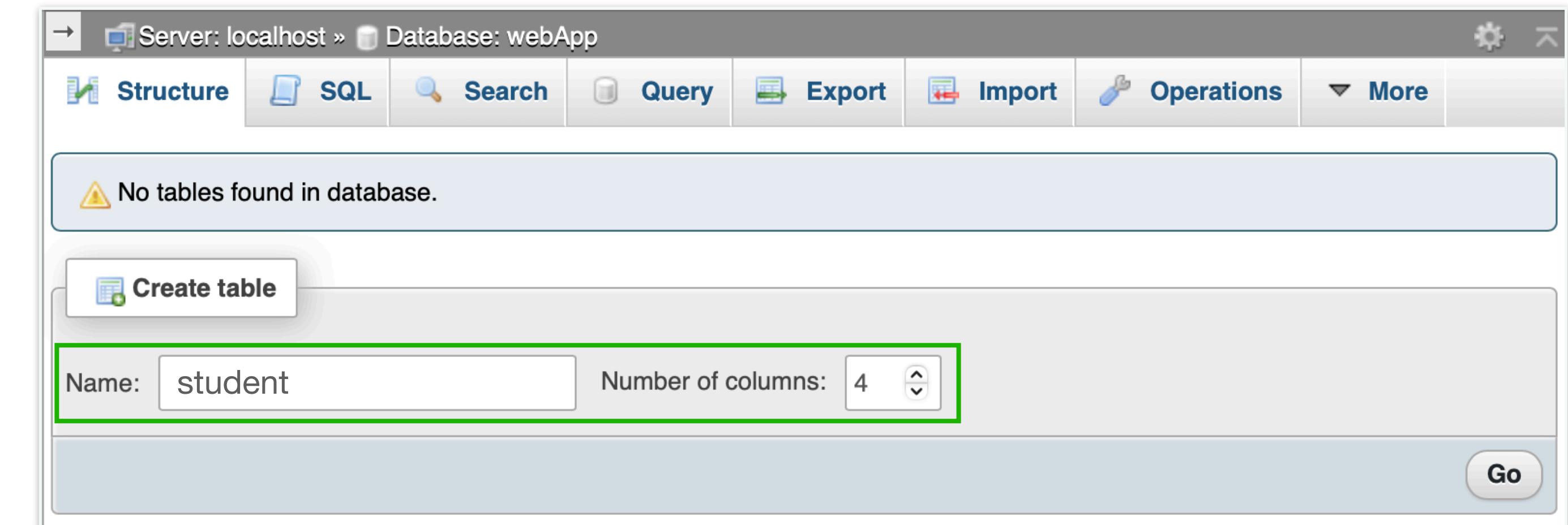
MySQL returned an empty result set (i.e. zero rows). (Query took 0.0027 seconds.)

```
CREATE DATABASE webApp CHARACTER SET utf8 COLLATE utf8_general_ci
```

At the bottom, there are links for '[Edit inline]', '[Edit]', and '[Create PHP code]'

การสร้างตารางข้อมูล

- ▶ สร้างตารางข้อมูลชื่อ student



การสร้างตารางข้อมูล

- ▶ สร้าง Field / Attribute ข้อมูลสำหรับตาราง student

The screenshot shows the MySQL Workbench interface for creating a new table named 'student'. The table structure is defined with four columns:

Name	Type	Length/Values	Default	Collation	Attributes	Null	Index	A_I	Comments
std_id	VARCHAR	11	None						
std_name	VARCHAR	100	None						
std_dept	VARCHAR	200	None						
std_gpa	FLOAT		None						

Below the table structure, there are sections for 'Structure' and 'PARTITION definition'. The 'Storage Engine' is set to InnoDB.

การสร้างตารางข้อมูล

▶ Schema ของตาราง student

Server: localhost » Database: webApp » Table: student

Browse Structure SQL Search Insert Export Import Privileges

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	std_id	varchar(11)	utf8_general_ci		No	None			Change Drop More
2	std_name	varchar(100)	utf8_general_ci		No	None			Change Drop More
3	std_dept	varchar(200)	utf8_general_ci		No	None			Change Drop More
4	std_gpa	float			No	None			Change Drop More

Check all With selected: Browse Change Drop Primary Unique Index

Remove from central columns

Print Propose table structure Track table Move columns Normalize

Add 1 column(s) after std_gpa Go

การสร้างตารางข้อมูล

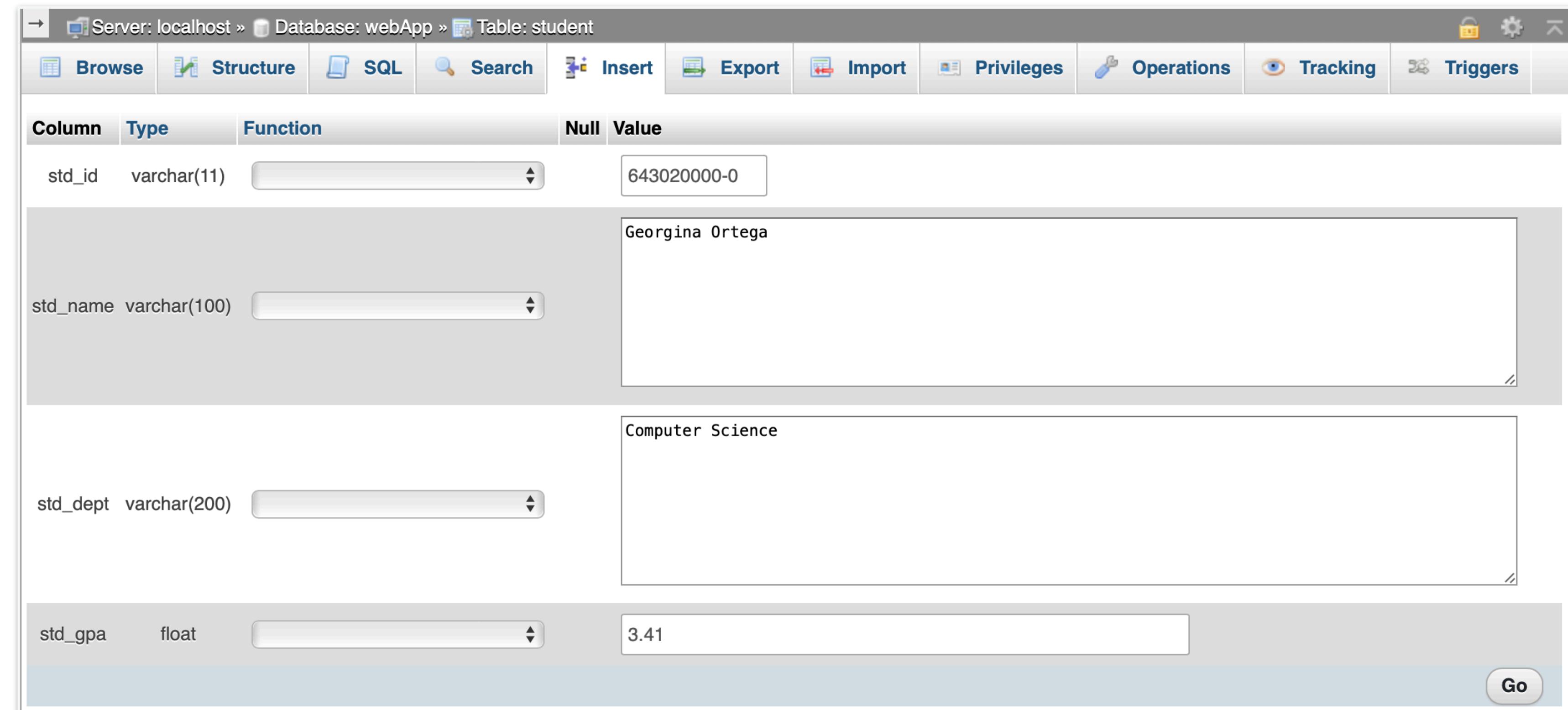
- ▶ เพิ่มข้อมูลในตาราง

Server: localhost » Database: webApp » Table: student

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Column	Type	Function	Null	Value
std_id	varchar(11)			643020000-0
std_name	varchar(100)			Georgina Ortega
std_dept	varchar(200)			Computer Science
std_gpa	float			3.41

Go



การสร้างตารางข้อมูล

▶ เพิ่มข้อมูลในตาราง

The screenshot shows the MySQL Workbench interface for the 'student' table in the 'webApp' database. The 'Insert' tab is selected. A success message at the top states '2 rows inserted.' Below it is the SQL query:

```
INSERT INTO `student` (`std_id`, `std_name`, `std_dept`, `std_gpa`) VALUES ('643020000-0', 'Georgina Ortega', 'Computer Science', '3.41'), ('643020000-1', 'Christina McBride', 'Maths', '3.32');
```

Below the query, there are buttons for [Edit inline], [Edit], and [Create PHP code]. The main area shows the table structure with columns: std_id, std_name, std_dept, and std_gpa. Two rows are displayed:

std_id	std_name	std_dept	std_gpa
643020000-0	Georgina Ortega	Computer Science	3.41
643020000-1	Christina McBride	Maths	3.32

At the bottom, there are buttons for SELECT *, SELECT, INSERT, UPDATE, DELETE, Clear, Format, Get auto-saved query, and Bind parameters. There is also a bookmark field and options for Delimiter, Show this query here again, Retain query box, Rollback when finished, Enable foreign key checks, and Go.



เชื่อมต่อกับ MySQL ด้วยภาษา PHP

ขั้นตอนการเชื่อมต่อกับ MySQL

- ▶ ประกอบด้วยการทำงาน 4 ขั้นตอน
 1. ติดต่อ MySQL และเลือกฐานข้อมูลที่ต้องการใช้งาน
 2. กำหนดคำสั่ง SQL และส่ง request ให้ MySQL ไปดำเนินการ
 3. รับผลลัพธ์ที่ได้จาก MySQL และนำไปใช้งาน
 4. ปิดการเชื่อมต่อกับ MySQL

พื้นฐานในการจัดการฐานข้อมูลใน MySQL

- ▶ **mysqli_connect()** : เชื่อมต่อกับฐานข้อมูล
- ▶ ในการติดต่อกับฐานข้อมูลจะต้องเปิดการติดต่อ Database Server ก่อน โดยมีรูปแบบการใช้งาน ดังนี้

```
mysqli_connect(hostname, username, password, database_name);
```

- ▶ **hostname** คือ ชื่อของ Database Server หากต้องการเข้าถึง MySQL ที่ติดตั้งไว้ในเครื่องเดียวกันกับเว็บเซิร์ฟเวอร์ สามารถระบุเป็น localhost ได้
- ▶ **username** คือ ชื่อของผู้ใช้ที่สามารถเข้าถึงฐานข้อมูลได้
- ▶ **password** คือ รหัสผ่านของผู้ใช้
- ▶ **database_name** คือ ชื่อของฐานข้อมูลที่ต้องการเข้าถึง

ฟังก์ชันในการจัดการฐานข้อมูลใน MySQL

- ▶ **mysqli_connect()** : เชื่อมต่อกับฐานข้อมูล
- ▶ หากเชื่อมต่อกับฐานข้อมูลสำเร็จ จะคืนค่าเป็น true หากเชื่อมต่อไม่สำเร็จจะคืนค่าเป็น false

PHP Code

```
$host = "localhost";
$user = "root";
$password = "";
$dbname = "webApp";
$conn = mysqli_connect($host, $user, $password, $dbname);
if($conn){
    echo "Connect to database successfully.";
}
else{
    echo "Error! Cannot connect to database.";
    exit;
}
```



ฟังก์ชันในการจัดการฐานข้อมูลใน MySQL

- ▶ **mysqli_close()** : ยกเลิกการเชื่อมต่อกับฐานข้อมูล
- ▶ หากยกเลิกการเชื่อมต่อกับฐานข้อมูลสำเร็จ จะคืนค่าเป็น true หากไม่สำเร็จจะคืนค่าเป็น false

PHP Code

```
$host = "localhost";
$user = "root";
$password = "";
$dbname = "webApp";
$conn = mysqli_connect($host, $user, $password, $dbname);
if($conn){
    echo "Connect to database successfully.";
    if(mysqli_close($conn)){
        echo "Close connection.";
    }
}
```



ฟังก์ชันในการจัดการฐานข้อมูลใน MySQL

- ▶ **mysqli_query()** : ฟังก์ชันสำหรับสั่งงาน MySQL ด้วยภาษา SQL เพื่อจัดการกับฐานข้อมูล เช่น การค้นหา การเพิ่ม การลบ

```
mysqli_query(db_connection, query_string);
```

- ▶ **db_connection** คือ ตัวแปรที่ใช้เชื่อมต่อกับฐานข้อมูล
- ▶ **query_string** คือ คำสั่งภาษา SQL ที่ใช้จัดการกับฐานข้อมูล

PHP Code

```
$conn = mysqli_connect($host, $user, $password, $dbname);

if($conn){

    $sql = "SELECT * FROM student";

    $result = mysqli_query($conn, $sql);

}
```

ฟังก์ชันในการจัดการฐานข้อมูลใน MySQL

- ▶ **mysqli_free_result()** : ฟังก์ชันสำหรับคืนหน่วยความจำให้กับระบบ เพื่อเป็นการใช้หน่วยความจำให้มีประสิทธิภาพมากที่สุด หากมีการใช้หน่วยความจำเก็บค่าของตัวแปรมาก ๆ อาจจะส่งผลให้หน่วยความจำเต็มและมีผลต่อการทำงานของโปรแกรม

ค่าที่ได้จากการคำสั่ง mysqli_query()

```
mysqli_free_result(result);
```

PHP Code

```
$conn = mysqli_connect($host, $user, $password, $dbname);

if($conn){

    $sql = "SELECT * FROM student";

    $result = mysqli_query($conn, $sql);

    ..

    mysqli_free_result($result);

}
```

ฟังก์ชันในการจัดการฐานข้อมูลใน MySQL

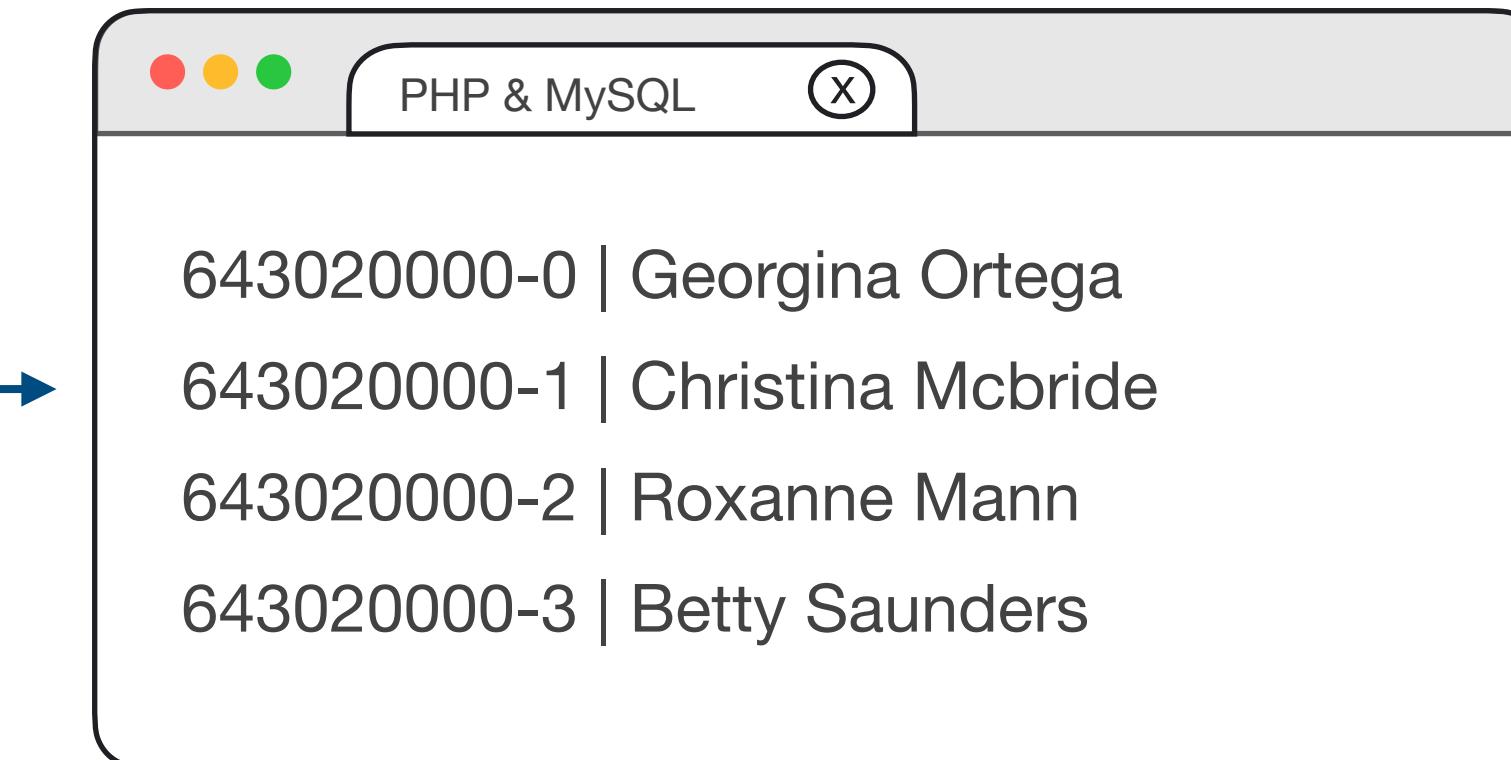
- ▶ **mysqli_fetch_row()** : ฟังก์ชันสำหรับเลื่อนตำแหน่งของตัวชี้ผลลัพธ์ไปยังข้อมูลแถวถัดไป

ค่าที่ได้จากการคำสั่ง `mysqli_query()`

```
mysqli_fetch_row(result);
```

PHP Code

```
$sql = "SELECT * FROM student";  
  
$result = mysqli_query($conn, $sql);  
  
while($rs = mysqli_fetch_row($result)){  
    echo $rs[0]." | ".$rs[1]."  
}  
}
```



643020000-0 | Georgina Ortega
643020000-1 | Christina McBride
643020000-2 | Roxanne Mann
643020000-3 | Betty Saunders

- ▶ ฟังก์ชันจะอ่านค่าและส่งค่าจากมาให้ตัวแปร array ทีละ 1 แถว (row) ถ้าต้องการเข้าถึงข้อมูลครบถ้วน จะต้องให้ทำงานแบบวนซ้ำเพื่อเข้าถึงข้อมูลແவສุดท้าย

ฟังก์ชันในการจัดการฐานข้อมูลใน MySQL

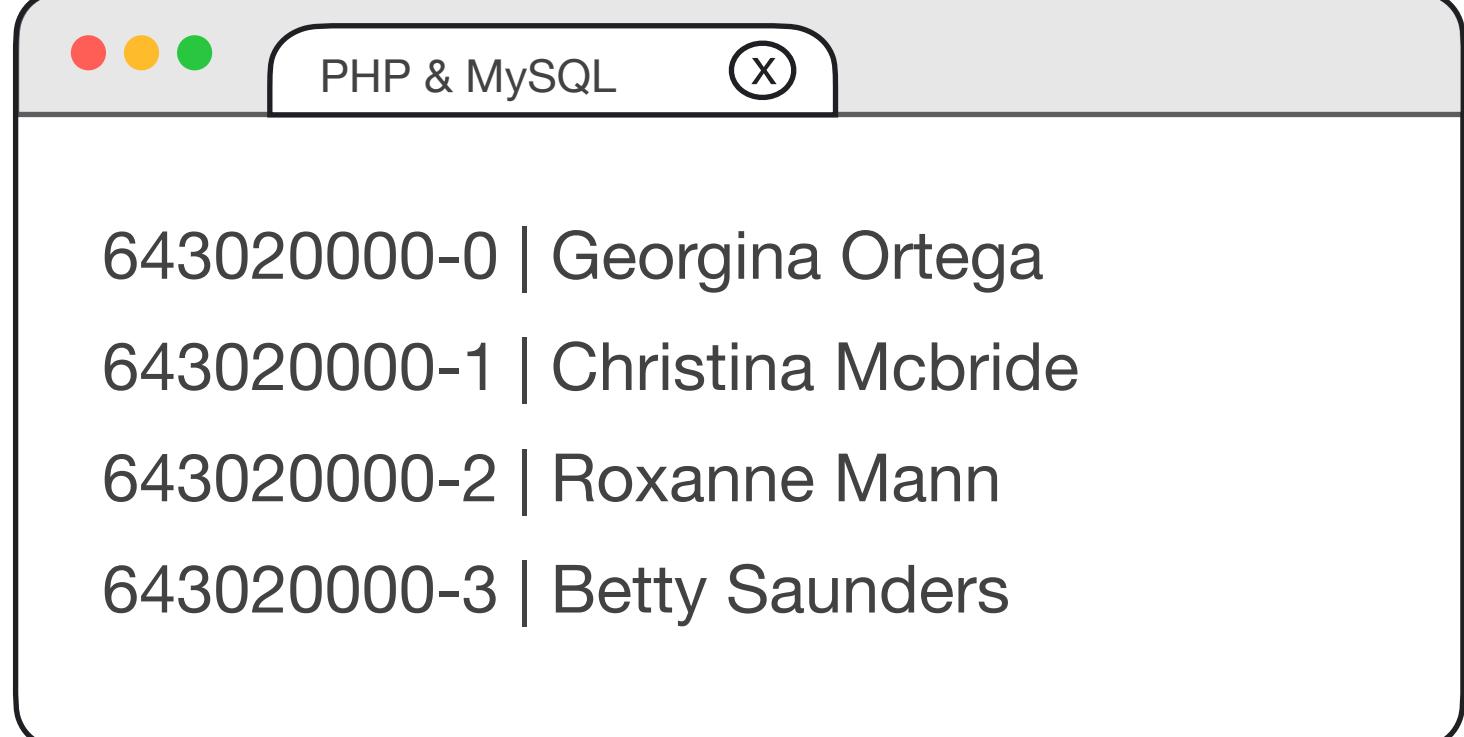
- ▶ **mysqli_fetch_assoc()** : ฟังก์ชันสำหรับดึงค่าผลลัพธ์จากฐานข้อมูลมาเก็บไว้ในตัวแปร associative array โดยสามารถเข้าถึงค่าของแต่ละคอลัมน์ได้โดยการระบุชื่อคอลัมน์

```
mysqli_fetch_assoc(result);
```

ค่าที่ได้จากการคำสั่ง mysqli_query()

PHP Code

```
$sql = "SELECT * FROM student";  
  
$result = mysqli_query($conn, $sql);  
  
while($rs = mysqli_fetch_assoc($result)){  
    echo $rs["std_id"]." | ".$rs["std_name"]."  
}  
"
```



643020000-0 | Georgina Ortega
643020000-1 | Christina McBride
643020000-2 | Roxanne Mann
643020000-3 | Betty Saunders

- ▶ ฟังก์ชันจะอ่านค่าและส่งค่าจากมาให้ตัวแปร associative array ทีละ 1 แถว (row) ถ้าต้องการเข้าถึงข้อมูลครบถ้วน จะต้องให้ทำงานแบบวนซ้ำเพื่อเข้าถึงข้อมูลແລ້ວສຸດທ້າຍ

ฟังก์ชันในการจัดการฐานข้อมูลใน MySQL

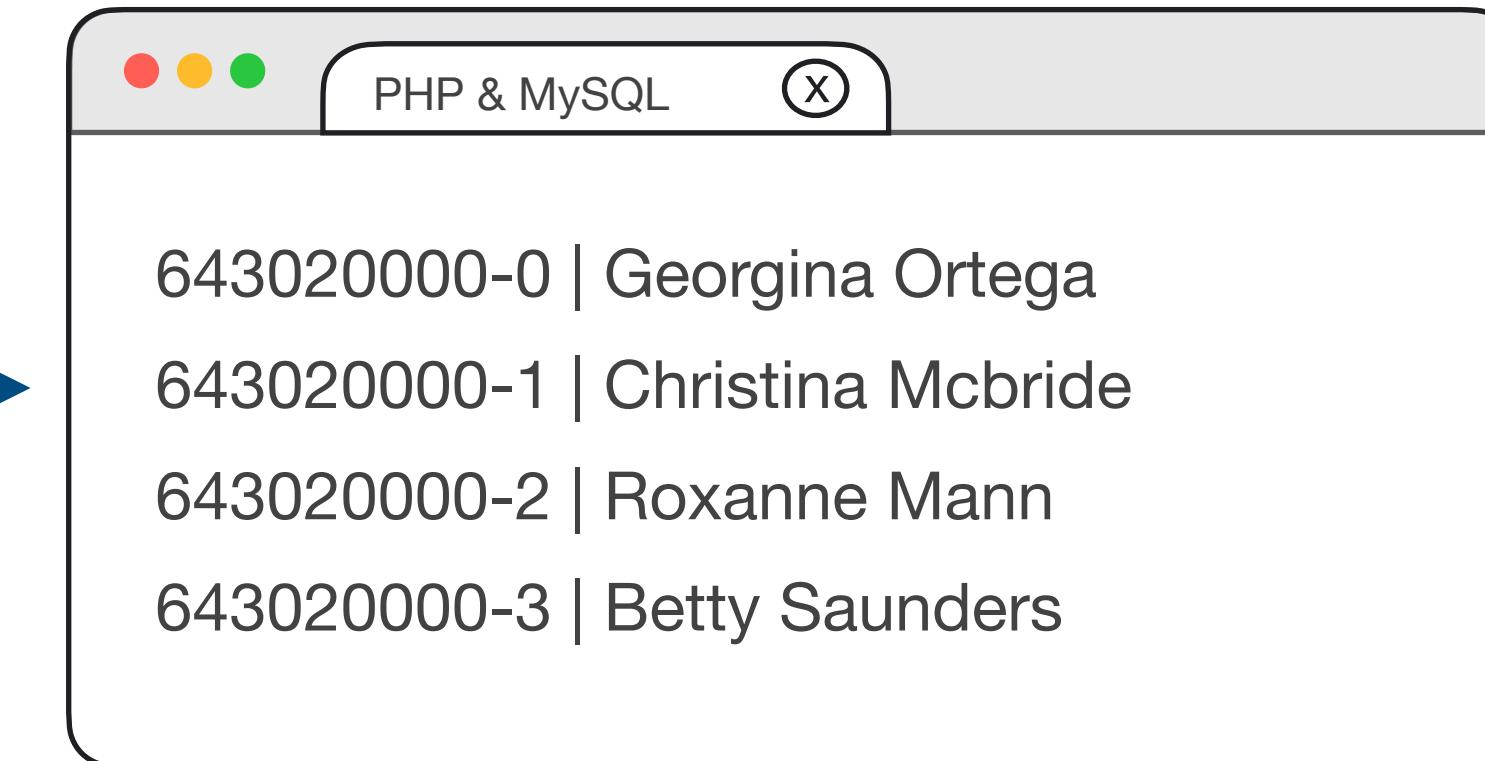
- ▶ **mysqli_fetch_array()** : ฟังก์ชันสำหรับดึงค่าผลลัพธ์จากฐานข้อมูลมาเก็บไว้ในตัวแปร array โดยจำนวนสมาชิกของ array จะมีจำนวนเท่ากับจำนวนคอลัมน์ หรือ attribute ของผลลัพธ์ โดยสามารถเข้าถึงค่าของแต่ละคอลัมน์ได้โดยการระบุชื่อคอลัมน์ หรือหมายเลข index

ค่าที่ได้จากการคำสั่ง mysqli_query()

```
mysqli_fetch_array(result);
```

PHP Code

```
$sql = "SELECT * FROM student";  
  
$result = mysqli_query($conn, $sql);  
  
while($rs = mysqli_fetch_array($result)){  
    echo $rs[0] . " | " . $rs["std_name"] . "<br>";  
}
```



643020000-0 | Georgina Ortega
643020000-1 | Christina McBride
643020000-2 | Roxanne Mann
643020000-3 | Betty Saunders

- ▶ ฟังก์ชันจะอ่านค่าและส่งค่าจากมาให้ตัวแปร array ทีละ 1 แถว (row) ถ้าต้องการเข้าถึงข้อมูลครบทุกแถว จะต้องให้ทำงานแบบวนซ้ำเพื่อเข้าถึงข้อมูลແຕวสุดท้าย

ฟังก์ชันในการจัดการฐานข้อมูลใน MySQL

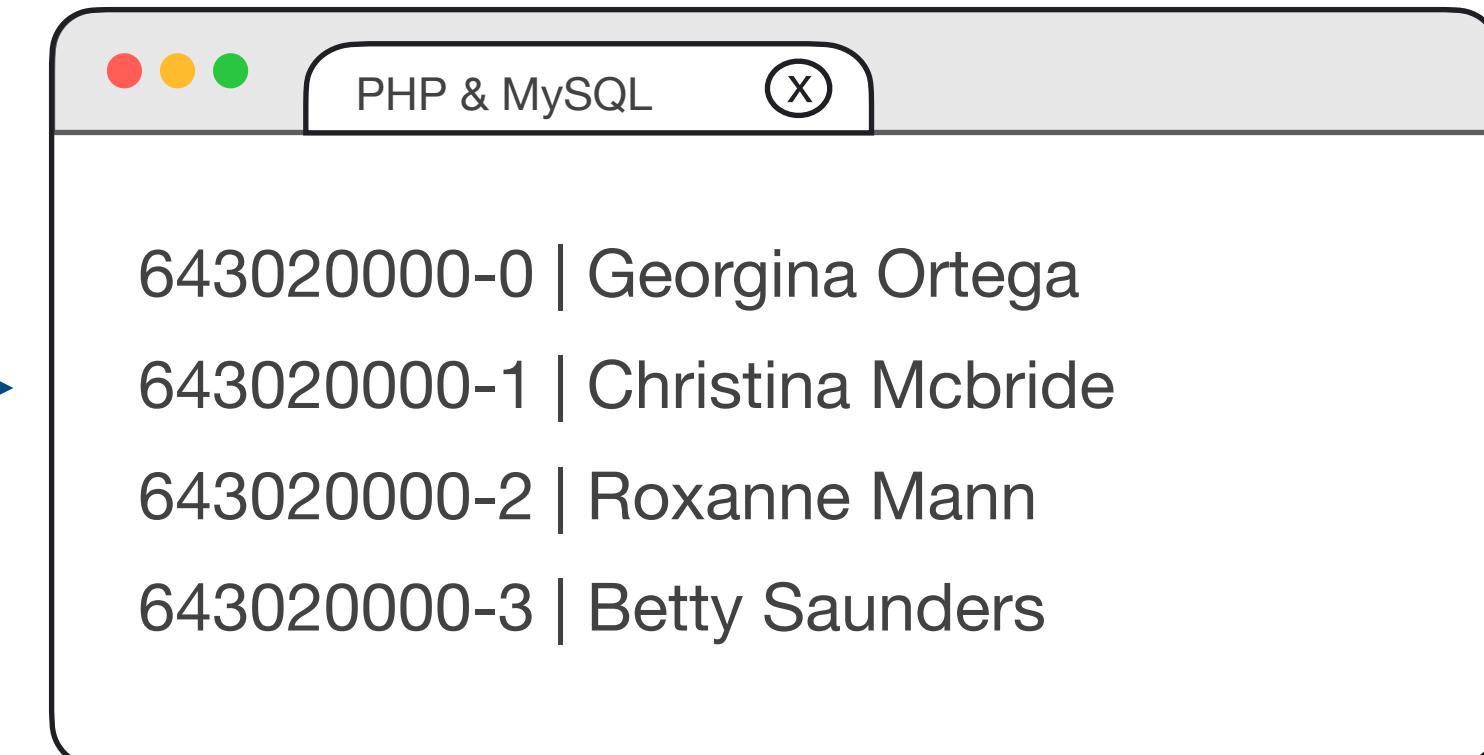
- ▶ **mysqli_fetch_object()** : ฟังก์ชันสำหรับดึงค่าผลลัพธ์จากฐานข้อมูลมาเก็บไว้ในตัวแปร object โดยสามารถเข้าถึงค่าของแต่ละคอลัมน์ได้โดยการระบุ attribute ของ object เป็นชื่อคอลัมน์

ค่าที่ได้จากการคำสั่ง mysqli_query()

```
mysqli_fetch_object(result);
```

PHP Code

```
$sql = "SELECT * FROM student";  
  
$result = mysqli_query($conn, $sql);  
  
while($rs = mysqli_fetch_object($result)){  
    echo $rs->std_id." | ".$rs->std_name."  
}  
}
```



- ▶ ฟังก์ชันจะอ่านค่าและส่งค่าจากมาให้ตัวแปร object ทีละ 1 แถว (row) ถ้าต้องการเข้าถึงข้อมูลครบถ้วน จะต้องให้ทำงานแบบวนซ้ำเพื่อเข้าถึงข้อมูลແຕวสุดท้าย

ฟังก์ชันในการจัดการฐานข้อมูลใน MySQL

- ▶ **mysqli_data_seek()** : ฟังก์ชันสำหรับเลื่อนตัวชี้ผลลัพธ์ไปยังลำดับที่ระบุ

```
mysqli_data_seek(result, index);
```

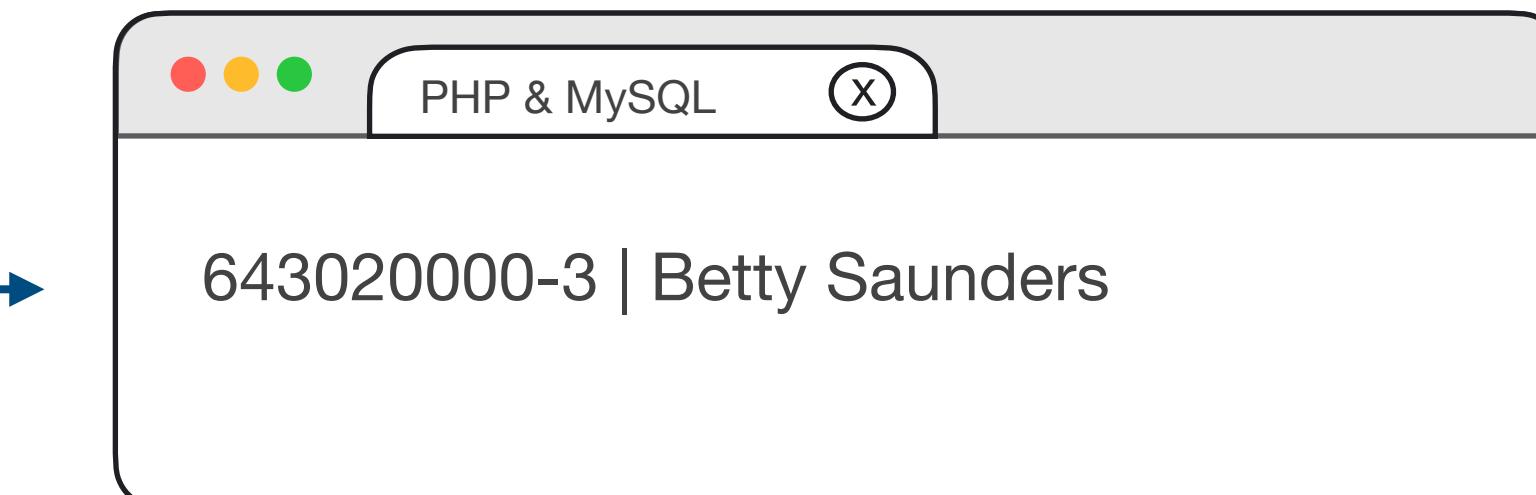
- ▶ **result** คือ ค่าที่ได้จากการคำสั่ง **mysqli_query()**
- ▶ **index** คือ ลำดับของแถว (row) ของผลลัพธ์

PHP Code

```
$sql = "SELECT * FROM student";
$result = mysqli_query($conn, $sql);
mysqli_data_seek($result, 3);

while($rs = mysqli_fetch_row($result)){
    echo $rs[0]." | ".$rs[1]."<br>";
}
```

	std_id	std_name	std_dept	std_gpa
0	643020000-0	Georgina Ortega	Computer Science	3.41
1	643020000-1	Christina McBride	Maths	3.32
2	643020000-2	Roxanne Mann	Physics	3.11
3	643020000-3	Betty Saunders	Computer Science	3.52



ตัวอย่างการเขียนคำสั่งเพื่อเชื่อมต่อฐานข้อมูล

ตารางข้อมูล student

std_id	std_name	std_dept	std_gpa
643020000-0	Georgina Ortega	Computer Science	3.41
643020000-1	Christina McBride	Maths	3.32
643020000-2	Roxanne Mann	Physics	3.11
643020000-3	Betty Saunders	Computer Science	3.52

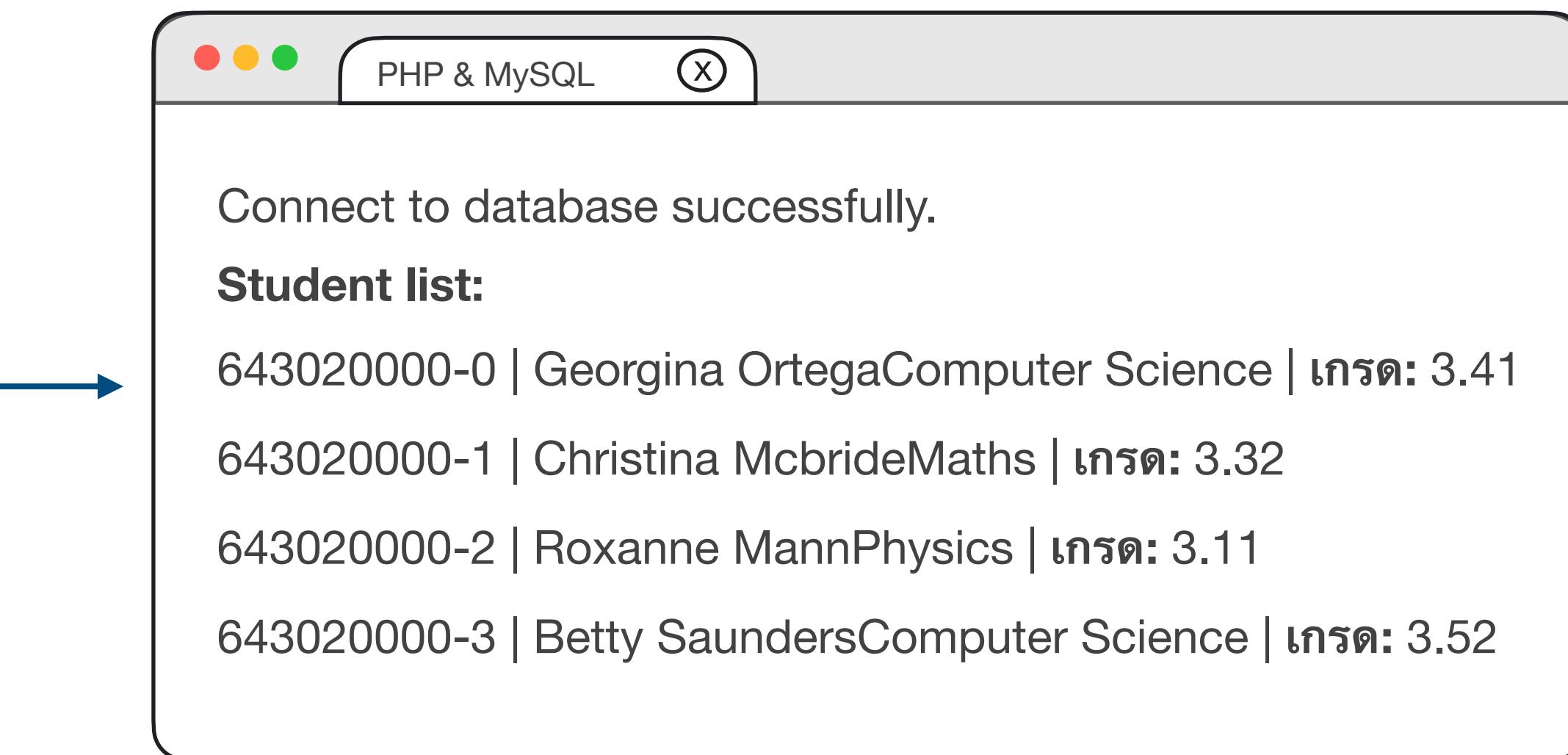
studentList.php

```
$host = "localhost";
$user = "root";
$password = "";
$dbname = "webApp";
$conn = mysqli_connect($host, $user, $password, $dbname);
```

ตัวอย่างการเขียนคำสั่งเพื่อเชื่อมต่อฐานข้อมูล

studentList.php

```
if($conn){  
    echo "Connect to database successfully.<br>";  
    $sql = "SELECT * FROM student";  
    $result = mysqli_query($conn, $sql);  
    echo "<b>Student list:</b> <br>";  
    while($rs = mysqli_fetch_assoc($result)){  
        echo $rs["std_id"]." | ".$rs["std_name"].  
$rs["std_dept"]." | <b>เกรด:</b> ".$rs["std_gpa"]."<br>";  
    }  
    mysqli_close($conn);  
}  
else{  
    echo "Error! Cannot connect to database.";  
    exit;  
}
```



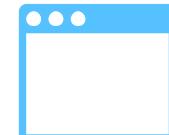
PHP & MySQL

Connect to database successfully.

Student list:

ID	Name	Department	Grade
643020000-0	Georgina Ortega	Computer Science	เกรด: 3.41
643020000-1	Christina McBride	Maths	เกรด: 3.32
643020000-2	Roxanne Mann	Physics	เกรด: 3.11
643020000-3	Betty Saunders	Computer Science	เกรด: 3.52

References

-  <https://www.php.net/manual/en>
-  <https://www.w3schools.com/php>