ภาษา SQL และ PHP เพื่อการโปรแกรมฐานข้อมูล

1. Structured Query Language (SQL)

1.1 ข้อมูลทั่วไป

SQL ย่อมาจาก Structured Query Language เป็นภาษามาตรฐานสำหรับการจัดเก็บและดึงข้อมูลในฐานข้อมูล ทำให้สามารถเรียกค้น เข้าถึงและจัดการกับฐานข้อมูลได้ และเป็นภาษามาตรฐาน ANSI (American National Standards Institute) แต่อาจมีความแตกต่างกันไปตามรุ่น (Version) ถ้าไม่มี PHP server กับ MySQL สามารถดาวโหลดน์โดยไม่เสียค่าใช้จ่ายได้ที่ http://www.mysql.com

ข้อมูลในฐานข้อมูล MySQL จะถูกเก็บไว้ในตาราง ตารางคือชุดของข้อมูลที่เกี่ยวข้องและประกอบด้วยคอลัมน์และแถว ความสามารถของ MySQL ประกอบด้วย

สามารถสร้างฐานข้อมูลใหม่

สามารถสร้างตารางใหม่ในฐานข้อมูล

สามารถดำเนินการการสอบถามกับฐานข้อมูล ดึงข้อมูลจากฐานข้อมูล เพิ่ม/แทรกระเบียนในฐานข้อมูล ลบระเบียนและปรับปรุงระเบียนในฐานข้อมูล

สามารถสร้างกระบวนงานที่เก็บไว้ในฐานข้อมูลได้

สามารถสร้างมุมมองในฐานข้อมูล

สามารถตั้งค่าสิทธิ์บนตารางวิธีการและมุมมอง

การสร้างเว็บไซต์ที่แสดงข้อมูลจากฐานข้อมูล จำเป็นต้องมีส่วนโปรแกรมการทำงานใน 3 ส่วน หลัก ประกอบด้วย 1) ส่วนโปรแกรมฐานข้อมูลประเภท Relational Database Management System (RDBMS) เช่น MS Access, SQL Server, MySQL เป็นต้น 2) ต้องใช้ภาษาสคริปต์ฝั่งเซิร์ฟเวอร์เช่น PHP หรือ ASP และต้องการใช้ภาษา SQL เพื่อรับข้อมูลที่ต้องการ และ 3) ใช้ภาษา HTML / CSS เพื่อกำหนดลักษณะหน้าเว็บ

การเขียนคำสั่งด้วย PHP 5 ขึ้นไปสามารถทำงานร่วมกับฐานข้อมูล MySQL โดยใช้ MySQLi extension ("i" หมายถึงการปรับปรุง) PHP Data Objects PDO (วัตถุข้อมูล PHP) เวอร์ชันก่อนหน้าของ PHP ใช้ส่วนขยาย MySQL (อย่างไรก็ตามส่วนขยายนี้ถูกเลิกใช้ในปี 2555 (ค.ศ.2012) ทั้งนี้การเลือกใช้ MySQLi หรือ PDO ขึ้นอยู่กับทักษะ ความถนัดหรือความชอบของผู้เขียนคำสั่ง เพราะทั้ง MySQLi และ PDO มีข้อดีข้อเด่นดังข้อเปรียบเทียบดังนี้

PDO	SQL
ทำงานใน 12 ระบบฐานข้อมูลที่แตกต่างกัน	ทำงานกับฐานข้อมูล MySQL เท่านั้น
ถ้าต้องเปลี่ยนความต้องการเพื่อใช้ฐานข้อมูลอื่น PDO ทำให้กระบวนการนี้ง่ายขึ้น เพียงการเปลี่ยนคำสั่งการเชื่อมต่อและแบบสอบถามในชุดคำสั่ง	จะต้องเขียนชุดคำสั่ง (Code) ใหม่ทั้งหมด รวมถึงแบบสอบถาม
เป็นแบบเชิงวัตถุ	เป็นแบบเชิงวัตถุ และ มี API
มีรูปแบบชุดคำสั่ง (Statements) และมีความสำคัญอย่างยิ่งต่อความปลอดภัยของแอปพลิเคชันบนเว็บ	มีรูปแบบชุดคำสั่ง (Statements) และมีความสำคัญอย่างยิ่งต่อความปลอดภัยของแอปพลิเคชันบนเว็บ

2. ตัวอย่างชุดคำสั่งภาษา SQL และ PHP

ในบทเรียนนี้เป็นการใช้โปรแกรม phpMyAdmin และจำลองเครื่องคอมพิวเตอร์ที่ใช้งานเป็น server กำหนดให้ มีurl ของ server = “localhost” ชื่อ server = “root” รหัสผ่าน pass = “ ” ดังนี้

2.1 ชุดคำสั่งการสร้างฐานข้อมูลและตาราง ประกอบด้วย

1) ชุดคำสั่งการสร้างฐานข้อมูล

คำสั่งสร้างฐานข้อมูล `CREATE DATABASE databasename;`

คำสั่งลบฐานข้อมูล `DROP DATABASE databasename;`

ตัวอย่างการเขียนชุดคำสั่ง

```
<?php
$servername = "localhost";
$username = "root";
$password = " ";
// Create connection
     $conn = new mysqli($servername, $username, $password);
// Check connection
if ($conn->connect_error) {
    die ("Connection failed: " . $conn->connect_error);
}
// Create database
$sql = "CREATE DATABASE GeogDB";
if ($conn->query($sql) === TRUE) {
    echo "Database created successfully";
} else {
    echo "Error creating database: " . $conn->error;
}
$conn->close();
?>
```

2) ชุดคำสั่งการสร้างตาราง

ต้องการสร้างตาราง ชื่อ MyStudent ประกอบด้วยฟิลด์ รหัส (ลำดับอัตโนมัติ) ชื่อ-นามสกุล เบอร์โทรศัพท์ สาขาวิชาที่เรียน เขียนคำสั่งดังนี้

```
CREATE TABLE MyStudent (
id INT(6) AUTO_INCREMENT,
studentname VARCHAR(50) NOT NULL,
Phone VARCHAR(30),
Major VARCHAR(50),
PRIMARY KEY (id)
);
```

ตัวอย่างการเขียนชุดคำสั่ง

```php
<?php
$servername = "localhost";
$username = "root";
$password = " ";
$dbname = "GeogDB";

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

// sql to create table
$sql = "CREATE TABLE MyStudent (
   id INT(6) AUTO_INCREMENT,
   studentname VARCHAR(50) NOT NULL,
   Phone VARCHAR(30),
   Major VARCHAR(50),
   PRIMARY KEY (id)
   )";

if ($conn->query($sql) === TRUE) {
    echo "Table MyStudent created successfully";
} else {
    echo "Error creating table: " . $conn->error;
}

$conn->close();
?>
```

แบบฝึกหัด

1. สร้างตาราง

```
CREATE TABLE `location` (
  `LOC_ID` int(10) NOT NULL auto_increment,
  `LOC_NAME` varchar(250) NOT NULL,
  `LAT` varchar(50) NOT NULL,
  `LNG` varchar(50) NOT NULL,
  PRIMARY KEY  (`LOC_ID`)
) ENGINE=MyISAM  DEFAULT CHARSET=utf8 AUTO_INCREMENT=5 ;
```

2. แทรกข้อมูลในตาราง

INSERT INTO `location` VALUES (1, 'สถานีอนามัย', '6.846876', '101.234481');

INSERT INTO `location` VALUES (2, 'หมู่บ้านทดสอบ', '6.247766', '101.255768');

INSERT INTO `location` VALUES (3, 'อ่าวปัตตานี', '6.985235', '101.282711');

INSERT INTO `location` VALUES (4, 'สเต็ก ลุงหนวด', '6.692970', '101.113834');