การใช้ PHP เบื้องต้น

Chapter 3

Edit 10-2016

Asst. Prof. Juthawut Chantharamalee

Computer Science @ Suan Dusit University

การเขียนภาษา PHP

สำหรับการเขียนก็จะอาศัยโปรแกรมประเภท text editor ทั่วไป เช่น ใช้โปรแกรม NotePad ในระบบ windows เป็นต้น แต่ที่นี้จะใช้โปรแกรม EditPlus

การเขียนภาษา PHP

โครงสร้างพื้นฐาน

ที่กล่าวไปแล้วว่า PHP สามารถใช้ร่วมกับ HTML ได้ทันทีนั้น เราจะมีสัญลักษณ์พิเศษที่แยก PHP ออกจาก HTML

แบบที่ 1 เปิดด้วยแท็ก <? และ ปิดด้วย ?> ภายใต้แท็ก <?...?> นั้นจะเป็น PHP ทั้งหมด ตัวอย่างเช่น

```
    ----+----1----+----2----+----3----+----4
1   <?
2   echo ("PHP Hello world\n")
3   ?>
4
```

โครงสร้างพื้นฐาน

แบบที่ 2 เปิดด้วยแท็ก <?php และ ปิดด้วย ?> ภายใต้แท็ก <?...?> นั้นจะเป็น PHP ทั้งหมด ตัวอย่างเช่น

```
1  <?php
2  echo ("PHP Hello world\n")
3  ?>
4
```

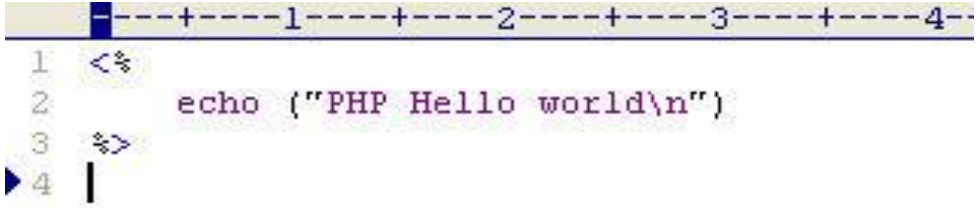
แบบที่ 3 เปิดด้วยแท็ก <script language="php"> และ ปิดด้วย </script> ภายใต้สคริปต์นั้นจะเป็น PHP ทั้งหมด ตัวอย่างเช่น

```
1  <script language="php">
2      echo ("PHP Hello world\n")
3  </script>
4
```

โครงสร้างพื้นฐาน

แบบที่ 4 เปิดด้วยแท็ก <% และ ปิดด้วย %> ภายใต้สคริปต์นั้นจะเป็น PHP ทั้งหมด

ตัวอย่างเช่น



การเขียน Comment

ในการเขียนโปรแกรมใดๆ ก็ตามโดยเฉพาะระบบโปรแกรมใหญ่ๆ ส่วนจะหลงลืม หรือจำไม่ได้ว่า แต่ละส่วนเขียนไปเพื่ออะไร จึงควรใส่หมายเหตุของโปรแกรมลงไปด้วย สำหรับ PHP นั้นใช้สัญลักษณ์ // และ # เพื่อบอกโปรแกรมว่า ไม่ต้องประมวลผล ในส่วนนั้นๆ

ตัวอย่าง

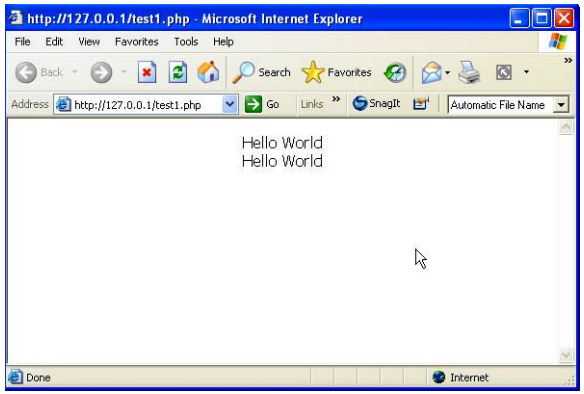
```
    |----+----1----+----2----+----3----+----4-
1   <?
2       echo "test"; /* แสดงข้อความ text */
3       /* comment หลายบรรทัด
4       ก็สามารถทำได้ */
5       echo "$sum"; // The summation of cost
6       echo "$mem_id"; // ID of each member
7       echo "$max_id"; # Maximun of member ID
8   ?>
9
```

การแสดงข้อความออกทาง Browser

ในการแสดงผลได้ 2 คำสั่งคือ echo และ print ซึ่งสามารถใช้แทนกันได้ทันที โดยไม่ต้องเปลี่ยน syntax ใดๆอีก

```
1  <html>
2  <head><title>echo&printf</title></head>
3  <body>
4  <?php
5      echo"<center><font size=3>Hello World<font></center>";
6      echo"<center><font size=3>Hello World<font></center>";
7  ?>
8  </body>
9  </html>
10
```

ผลที่ได้ :



การใช้ตัวแปรในภาษา PHP

สำหรับการเขียนโปรแกรมสำหรับภาษาคอมพิวเตอร์ระดับสูง สิ่งที่จะขาดเสียมิได้คือ การกำหนดและใช้ตัวแปร (variable) ตัวแปรในภาษา PHP จะเหมือนกับในภาษา Perl คือเริ่มต้นด้วยเครื่องหมาย dollar ($) โดยเราไม่จำเป็นต้องกำหนดแบบของข้อมูล (data type) อย่างเจาะจงเหมือนในภาษาซี เพราะว่า ตัวแปลภาษาจะจำแนกเองโดยอัตโนมัติว่า ตัวแปรดังกล่าว ใช้ข้อมูลแบบใด ในช่วงเวลานั้นๆ เช่น ข้อความ จำนวนเต็ม จำนวนที่มีเลขจุดทศนิยมตรรก เป็นต้น

ตัวอย่างการใช้งาน เช่น

```
----+----1----+----2----+----3----+
1  <?
2     $mystring = "Hello World!";
3     $myinteger = 1031;
4     $myfloat = 3.14;
5  ?>
6
```

การใช้ตัวแปรในภาษา PHP

ถ้าเราต้องการจะแสดงค่าของตัวแปร ก็อาจจะใช้คำสั่ง echo ได้

ตัวอย่าง เช่น

```
<?
   echo "$mystring\n";
   echo "$myinteger\n";
   echo "$myfloat\n";
?>
```

คำสั่งใส่รูปภาพลงเว็บเพจ

เราสามารถใช้คำสั่งแสดงรูปภาพที่เราต้องให้ปรากฏบนเว็บเพจเราได้ด้วยการใช้คำสั่ง <IMG SRC=\"ชื่อไฟล์.gif หรือ.jpg\"> โดยจะต้องมีการใช้ \ ด้วย เช่น

```
----+----1----+----2----+----3----+----4----+----5----+----
<HEAD>
<TITLE>ตุ๊กตา</TITLE></HEAD>
<BODY>
<?php

        echo"<center><img src=\"bg/b.gif\" height=150
        width=150></center>";
?>
</BODY>
</HTML>
```

คำสั่งใส่รูปภาพลงเว็บเพจ

การกำหนดขนาดรูปภาพ ให้ตรงกับความต้องการ WIDTH หมายถึง ความกว้างของรูปภาพ และHEIGHT หมายถึง ความสูงของรูปภาพ

<IMG SRC= \“picture.gif\” WIDTH=number% | HEIGHT=number%>

การกำหนดกรอบให้กับรูปภาพ <BORDER=n>

การวางตำแหน่งรูปภาพ

แบบแนวนอน ประกอบด้วย LEFT | RIGHT

แบบแนวตั้ง ประกอบด้วย เสมอบน มี 2 คำสั่ง คือ TOP | TEXTTOP

กึ่งกลาง มี 2 คำสั่ง คือ MIDDLE | ABSMIDDLE เสมอล่าง มี 3 คำสั่ง คือ

BASELINE | BOTTOM | ABSBOTTOM

ชนิดของข้อมูล

ชนิดของตัวแปร	ข้อมูลที่เก็บ
Integers	ตัวเลขจำนวนเต็ม เช่น 123,-123
Floating point numbers	เก็บข้อมูลที่มีทศนิยมเช่น 125.50
Strings	ข้อความเช่น “Hello”, “123”
Arrays	เก็บข้อมูลเป็นชุด
Object	เก็บข้อมูลในของ Class Object หรือ Method

Integers *ใช้สำหรับเก็บข้อมูลจำนวนเต็มทั้งจำนวนเต็มบวกและจำนวนเต็มลบ รวมทั้งแสดงค่าเป็น เลขฐานสิบ (0-9) ฐานแปด (0-7) และเลขฐานสิบหก (0-9, A-F หรือ a-f) โดยที่เลขฐานแปดจะขึ้นต้นด้วย 0 และเลขฐานสิบหกจะขึ้นต้นด้วย 0x หรือ 0X มีค่าได้ทั้งบวกและลบ*

ชนิดของข้อมูล

Floating point *ใช้สำหรับเก็บข้อมูลจำนวนจริงบวกและลบ จะมีทศนิยมหรือไม่มีก็ได้และรูปแบบยกกำลัง*

String *ใช้เก็บข้อมูลที่เป็นข้อความ รวมทั้งตัวเลข (ไม่สามารถนำไปคำนวณได้)*

รหัสควบคุมพิเศษต่างๆ

\n	ใช้ขึ้นบรรทัดใหม่
\f	เลื่อนเคอร์เซอร์ไปต้นบรรทัด
\t	ใช้เลื่อน Tab
\\	ใช้พิมพ์เครื่องหมาย \ (Backslash)
\$	ใช้พิมพ์เครื่องหมาย $ (Dollar)
\"	ใช้พิมพ์เครื่องหมาย " (Double Quote)
\[0-7]{1,3}	ใช้กำหนดอักขระเป็นรหัสฐาน 8
\x[0-9A-Fa-f]{1,2}	ใช้กำหนดอักขระเป็นรหัสฐาน 16

ชนิดของข้อมูล

Array ข้อมูลแบบนี้เป็นการเก็บข้อมูลเป็นชุดๆ แต่ละชุดมีสมาชิกเป็นของตัวเองจะมีมากน้อยแค่ไหนก็ได้ ทำให้มีความคล่องตัวในการใช้งานมากขึ้น การสร้างตัวแปรอาเรย์จะใช้ฟังก์ชัน array()

Object เป็นการเขียนชุดคำสั่งเพื่อเก็บข้อมูลในลักษณะออปเจกต์ เพื่อการเรียกใช้ Class Object หรือ Function

```
1  <?
2     class say
3     {
4
5        function say_hello(){
6        echo "Hell World";
7
8     }
9  }
10 $bar = new say;
11 $bar->say_hello;
12 ?>
13
```

ผลลัพธ์ที่ได้คือข้อความว่า Hello World

ในการเรียกใช้ฟังก์ชันที่อยู่ภายใน Class จะใช้เครื่องหมาย -> เป็นการอ้างอิง

ตัวดำเนินการ หรือ Operator

ในภาษา PHP มี Operator ต่างๆ ให้ใช้ ไม่ว่าจะเป็นโอเปอเรเตอร์ทางคณิตศาสตร์ โอเปอเรเตอร์เชิงตรรกะ เช่นเดียวดับภาษาอื่นดังนี้

การใช้งาน	ชื่อตัวดำเนินการ	ความหมาย
$a + $b	บวก	หาผลรวมระหว่าง $a กับ $b
$a – $b	ลบ	หารผลต่างระหว่าง $a กับ $b
$a * $b	คูณ	หาผลคูณระหว่าง $a กับ $b
$a / $b	หาร	หาผลหารระหว่าง $a กับ $b
$a % $b	หารเอาเศษ	หาเศษของการหารระหว่าง $a กับ $b

ตัวดำเนินการ หรือ Operator

การใช้งาน	ชื่อตัวดำเนินการ	ความหมาย
++$a	Pre-increment	เพิ่มค่า 1 ก่อนแล้วค่อยให้ค่ากับตัวแปร
$a++	Post-increment	ให้ค่ากับตัวแปรก่อน แล้วค่อยเพิ่มค่า 1
--$a	Pre-decrement	ลดค่า 1 ก่อนแล้วค่อยให้ค่ากับตัวแปร
$a--	Post-decrement	ให้ค่ากับตัวแปรก่อน แล้วค่อยลดค่า 1
$a and $b	and	เป็นจริงเมื่อ a และ b เป็นจริง
$a or $b	or	เป็นจริงเมื่อ a หรือ b เป็นจริง
$a xor $b	Exclusive or	เป็นจริงเมื่อ a และ b ตัวใดตัวหนึ่งเป็นจริง
! $a	not	เป็นจริงเมื่อ a เป็นเท็จ
$a && $b	and	เป็นจริงเมื่อ a และ b เป็นจริง
$a || $b	or	เป็นจริงเมื่อ a หรือ b เป็นจริง
$a == $b	เท่ากับ	เป็นจริงเมื่อ a เท่ากับ b
$a != $b	ไม่เท่ากับ	เป็นจริงเมื่อ a ไม่เท่ากับ b
$a < $b	น้อยกว่า	เป็นจริงเมื่อ a น้อยกว่า b
$a > $b	มากกว่า	เป็นจริงเมื่อ a มากกว่า b
$a <= $b	น้อยกว่าหรือเท่ากับ	เป็นจริงเมื่อ a น้อยกว่าหรือเท่ากับ b
$a >= $b	มากกว่าหรือเท่ากับ	เป็นจริงเมื่อ a มากกว่าหรือเท่ากับ b

การใช้เงื่อนไข (condition) เพื่อการตัดสินใจ

การใช้ IF...ELSE Condition เป็นการกำหนดเงื่อนไขที่ธรรมดาที่สุด คือกำหนดเงื่อนไข แล้วโปรแกรมตรวจสอบเงื่อนไขนั้น ถ้าเงื่อนไขนั้นเป็นจริงก็จะทำตามคำสั่งที่กำหนด ถ้าเป็นเท็จก็จะไม่ทำ

```
<?
  $sum=10;
  if ($sum==0) {
     echo "Summation is 0";
  }
  else {
     echo "Summation = ". $sum;
  }
?>
```

ผลที่ได้ : Summation = 10

การใช้เงื่อนไข (condition) เพื่อการตัดสินใจ

การใช้ Switch...Case ในบางครั้งในการกำหนดทางเลือกของโปรแกรมโดยการใช้ If...Else อาจจะทำให้เขียนโปรแกรมยาวและทำความเข้าใจยาก ดังนั้นเราอาจใช้ Switch แทนซึ่งเขียนโปรแกรมง่ายกว่าและมีความกระชับมากกว่า

```
<?
   $i=2;
   switch ($i) {
      case 0: print "i equals 0"; break;
      case 1: print "i equals 1"; break;
      case 2: print "i equals 2"; break;
   }
?>
```

ผลที่ได้ : i equals 2

การวนลูป

การใช้ While Loop คำสั่ง while จะทำงานโดยการตรวจสอบเงื่อนไขก่อน ถ้าเงื่อนไขเป็นจริงก็จะทำตามคำสั่ง

```
<?
   $i=1 // ให้ค่าเริ่มต้น
   while ($i<=5) {
      print $i;
      $i++;
   }
?>
```

ผลที่ได้ : 12345

การวนลูป

Do while เป็นคำสั่งที่คล้ายกับ While Loop แต่ต่างกันที่ Do while นั้นจะทำงานโดยการตรวจสอบเงื่อนไขภายหลังจากการทำงานไปแล้วแต่ While นั้นจะตรวจสอบเงื่อนไขก่อนการทำงาน

```
    +----+----1----+----2----+----3
1   <?
2     $i=5 // ให้ค่าเริ่มต้น
3       do {
4          print $i;
5          $i++;
6       } while ($i<=5);
7   ?>
8
```

ผลที่ได้ : 5

กรณีที่ใช้ While...Loop จะทำการตรวจสอบเงื่อนไขก่อน แล้วจึงค่อยทำในลูป

กรณีที่ใช้ Do...Loop จะทำคำสั่งในลูปก่อน แล้วจึงค่อยตรวจสอบเงื่อนไข

การวนลูป

For Loop คำสั่งนี้จะทำหน้าที่สั่งให้โปรแกรมทำงานวนรอบตามต้องการ ซึ่งกำหนดเป็นเงื่อนไข โดยจะทำเมื่อเงื่อนไขเป็นจริง และจะมีลักษณะการวนรอบที่รู้จำนวนรอบที่แน่นอน

```
----+----1----+----2----+----3-
1  <?
2     for ($i=1; $i <=5; $i++) {
3        print $i;
4     }
5  ?>
6
```

ผลที่ได้ : 12345

การวนลูป

Foreach เป็นการทำงานในลักษณะวนรอบที่ทำงานกับตัวแปรอาร์เรย์ โดยสามารถเข้าถึงข้อมูลทั่วไป โดย $Value เป็นตัวกำหนดค่าให้กับ array expression โดยพอยน์เตอร์จะเลื่อนไปตามสมาชิดถัดไปของอาร์เรย์ตามการเปลี่ยนแปลงรอบที่เปลี่ยนไป

```
    -----+----1----+----2----+----3----+----4----+----5----+
1   ...
2   <select name="month">
3   <?
4     $month=array("มกราคม","กุมภาพันธ์","มีนาคม","เมษายน",
      "พฤษภาคม",    "มิถุนายน","กรกฎาคม","สิงหาคม","กันยายน",
      "ตุลาคม","พฤศจิกายน","ธันวาคม");
5     foreach ($momth as $value)
6     {
7       echo "<option>".$value
8     }
9   ?>
10  </select>
11  ...
12
```

การวนลูป

การใช้ break และ continue ภายในลูป

คำสั่ง break เป็นคำสั่งจะใช้เพื่อให้หยุดการทำงาน จากการใช้คำสั่งเพื่อวนรอบที่ผ่านมาจะเห็นว่าจะออกจากการวนรอบเมื่อสิ้นสุดการทำงานแล้วเท่านั้น แต่ถ้าต้องการให้หยุดทำงานกะทันหัน สามารถใช้คำสั่ง break ก็ได้

คำสั่ง continue เป็นคำสั่งที่ทำงานตรงข้ามกับคำสั่ง break คือ จะสั่งให้โปรแกรมทำงานต่อไป ถ้าใช้คำสั่ง Continue กับ For เมื่อพบคำสั่งนี้จะเป็นการสั่งให้กลับไปเพิ่มค่าให้กับตัวแปรทันที หรือถ้าใช้กับคำสั่ง While เมื่อพบคำสั่งนี้จะเป็นการสั่งให้กลับไปทดสอบเงื่อนไขใหม่ทันที

การวนลูป

```
<?
  unset($a);
  $a[]=1;
  $a[]=2;
  $a[]=3;
  $a[]="red";
  $a[]="green";
  $a[]="blue";
  $a[]="none";

  $i=0;
  $found="not found";
  for ($i=0; $i < count($a); $i++) {
      if ( is_long($a[$i]) ) { // skip all integer elements
      continue;
    }
  if ($a[$i] == "blue") {
      $found=$a[$i];
      break;
    }
  }
echo $found,"<BR>\n";
?>
```

ผลที่ได้ :Blue

คำสั่ง continue บังคับให้ไปเริ่มต้นทำขั้นตอนในการวนลูปครั้งต่อไป ส่วน break นั้นส่งผลให้หยุดการทำงานของลูป

การวนลูป

การใช้คำสั่ง include และ require

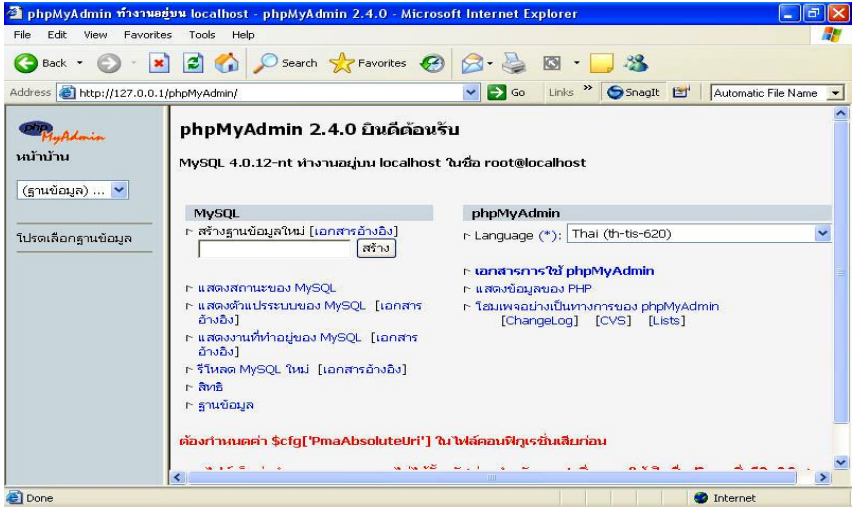
คำสั่งทั้งสองเอาไว้แทรกเนื้อหาจากไฟล์อื่นที่ต้องการ ข้อแตกต่างระหว่าง include และ require อยู่ตรงที่ว่า ในกรณีของการแทรกไฟล์ใช้ชื่อต่างๆ กันมากกว่าหนึ่งครั้งโดยใช้ลูป คำสั่ง require จะอ่านเพียงแค่ครั้งเดียว คือไฟล์แรก และจะแทรกไฟล์นี้เท่านั้นไปตามจำนวนครั้งที่วนลูป ในขณะที่ include สามารถอ่านได้ไฟล์ต่างๆ กันตามจำนวนครั้งที่ต้องการ

```
<?
include ("script.inc");
?>
```

การใช้งาน MySQL

การสร้างฐานข้อมูล

ในการสร้างฐานข้อมูลของ MySQL สามารถสร้างผ่าน phpMyAdmin ได้เลย โดยการเลือก Internet Explorer ขึ้นมาพิมพ์ 127.0.0.1 ที่ address bar จะได้หน้าต่างดังนี้



ชนิดของข้อมูลใน MySQL

ชนิดของข้อมูลพื้นฐาน มี 3 ชนิด คือ ตัวเลข, วันที่เวลา และตัวอักษร แต่ละชนิดจะมีขนาดไม่เท่ากัน ดังนั้น เมื่อกำหนดคอลัมน์หรือฟิลด์ข้อมูลในตารางบนฐานข้อมูล จะต้องคำนึงถึงชนิดของข้อมูลด้วย เพื่อความเหมาะสมของข้อมูล โดยข้อมูลแต่ละชนิดมีรายละเอียดดังนี้

ชนิดตัวเลข แบ่งได้เป็น เลขจำนวนเต็มและเลขจำนวนจริง

ตารางแสดงชนิดของตัวเลขจำนวนเต็ม

ชนิด	ความจุ (Bytes)	คำอธิบาย
TINYINT	1	เป็นตัวเลขที่มีช่วงความยาวตั้งแต่ -127 ถึง 128 หรือ 0 ถึง255
SMALLINT	2	เป็นตัวเลขที่มีช่วงความยาวตั้งแต่ -32468 ถึง 32768 หรือ 0 ถึง 65535
MEDIUMINT	3	เป็นตัวเลขที่มีช่วงความยาวตั้งแต่ -8388608 ถึง 8388607 หรือ 0 ถึง 16777215
INT , INTEGER	4	เป็นตัวเลขที่มีช่วงความยาวตั้งแต่-2^{31} ถึง 2^{31}-1 หรือ 0 ถึง 2^{32}-1
BIGINT	8	เป็นตัวเลขที่มีช่วงความยาวตั้งแต่-2^{63} ถึง 2^{63}-1 หรือ 0 ถึง 2^{64}-1

ชนิดของข้อมูลใน MySQL

ตารางแสดงชนิดของเลขจำนวนจริง

ชนิด	ความจุ (Bytes)	คำอธิบาย
FLOAT	4	เป็น precision แบบเดี่ยว ซึ่งจะแสดงตัวเลขเป็นทศนิยม
DOUBLE	8	เป็น precision แบบคู่ ซึ่งจะแสดงตัวเลขเป็นทศนิยม

ชนิดวันที่และวันเวลา

ตารางแสดงชนิดวันที่และเวลา

ชนิด	ช่วงของข้อมูล	คำอธิบาย
DATE	1000-01-01 ถึง 9999-12-31	แสดงวันที่ซึ่งมีรูปแบบคือ YYYY-MM-DD
TIME	-838:59:59 ถึง 838:59:59	แสดงวันที่ซึ่งมีรูปแบบคือ HH:MM:SS
DATETIME	1000-01-01 00:00:00 ถึง 9999-12-31 23:59:59	แสดงวันที่พร้อมเวลา ซึ่งมีรูปแบบคือ YYYY-MM-DD HH:MM:SS
TIMESTAME	1970-01-01 00:00:00	แสดงวันที่และเวลาซึ่งมีรูปแบบตามที่กำหนด
YEAR	0-69 (1970-2069) 1910-2155	แสดงปีซึ่งสามารถกำหนดให้แสดงได้ 2 รูปแบบ คือให้แสดงทั้งหมด 2 และ 4 หลัก

ชนิดของข้อมูลใน MySQL

ชนิดตัวอักษร

ตารางแสดงชนิดของสตริง

ชนิด	ความยาว	คำอธิบาย
CHAR	1 ถึง 255 ตัวอักษร	มีค่าระหว่าง 1 ถึง 255 ตัวอักษร จะเก็บข้อมูลทั้งหมดรวมทั้งค่าว่างด้วย(เก็บข้อมูลทั้งหมดเท่าที่กำหนด)
VARCHAR	1 ถึง 255 ตัวอักษร	เหมือนกับ CHAR แต่รองรับค่าที่มีความยาวมากกว่า จะเก็บขนาดเท่ากับขนาดของข้อมูล

ฟังก์ชันในการจัดการฐานข้อมูลใน MySQL

การเชื่อมต่อกับฐานข้อมูล

ในการติดต่อกับฐานข้อมูลจะต้องทำหารเปิดการติดต่อดาต้าเบสเซิร์ฟเวอร์ก่อน โดยมีรูปแบบการใช้งานดังนี้

mysql_connect(hostname, username, password);

hostname คือ ชื่อของดาต้าเบสเซิร์ฟเวอร์ ในการที่ติดตั้ง MySQL ไว้ในเครื่องเดียวกับเว็บเซิร์เวอร์ ก็สามารถระบุเป็น localhost แทนชื่อจริงได้เลย

username คือ ชื่อผู้ใช้ที่ถูกกำหนดให้สามารถทำงานกับ MySQL ได้

password คือ รหัสผ่านของผู้ใช้ หรือจะระบุหรือไม่ก็ได้

ฟังก์ชันในการจัดการฐานข้อมูลใน MySQL

ค่าที่คืนออกมาจากการเรียกใช้ฟังก์ชันนี้จะได้ค่าเป็นจริงหากสามารถติดต่อกับ MySQL ได้สำเร็จแต่ถ้าไม่สามารถติดต่อได้หรือติดต่อไม่สำเร็จจะมีค่าเป็นเท็จ เช่น

```
$host = "localhost";
$user = "user";
$password = "";
mysql_connect($host,$user,$passwordname) or die("ไม่สามารถติดต่อฐานข้อมูลได้");
```

ฟังก์ชันในการจัดการฐานข้อมูลใน MySQL

การยกเลิกการเชื่อมต่อ

ฟังก์ชันที่ใช้ในการยกเลิกหรือปิดการติดต่อดาต้าเบสเซิร์ฟเวอร์

mysql_close(database_connect);

โดยผลลัพธ์ที่คืนออกมาจากฟังก์ชันนี้ ถ้าปิดการติดต่อกับ MySQL ได้สำเร็จก็จะมีค่าเป็นจริง ถ้าไม่สำเร็จจะมีค่าเป็นเท็จ เช่น

```
mysql_close($conn);
```

ฟังก์ชันในการจัดการฐานข้อมูลใน MySQL

การเรียกใช้ฐานข้อมูลผ่านเว็บ

ก่อนการเรียกใช้ฟังก์ชันนี้ จะต้องมีการเรียกใช้ฟังก์ชัน mysql_connect เพื่อกำหนดฐานข้อมูลที่จะเชื่อมต่อเสียก่อน

mysql_select_db(string databasename);

Databasename คือ ชื่อของฐานข้อมูล เช่น

```
mysql_select_db("my data");
```

การนำภาษา SQL มาใช้ในฐานข้อมูล MySQL

ฟังก์ชัน mysql_query()

เป็นฟังก์ชันสำหรับสั่งงาน MySQL ด้วยภาษา SQL เพื่อจัดการกับข้อมูลในฐานข้อมูล เช่น การเพิ่ม การลบ เป็นต้น ต้องใช้กับฟังก์ชัน mysql_select_db()

mysql_query(string query, [database_connect]);

query หมายถึง คิวรีที่เรียกใช้ฐานข้อมูล

database_connect หมายถึง ตัวแปรที่ใช้เชื่อมต่อกับฐานข้อมูล จะกำหนดหรือไม่ก็ได้ เช่น

```
$sql = "select * from Shop";
$result = mysql_query($sql);
```

การนำภาษา SQL มาใช้ในฐานข้อมูล MySQL

ฟังก์ชัน mysql_db_query()

เป็นฟังก์ชันสำหรับสั่งงาน MySQL ด้วยภาษา SQL เพื่อจัดการกับข้อมูลในฐานข้อมูลเหมือนกับฟังก์ชัน mysql_query แต่ไม่ต้องใช้ร่วมกับฟังก์ชัน mysql_select_db()เพราะสามารถกำหนดชื่อฐานข้อมูลไว้ในฟังก์ชันได้เลย

mysql_db_query(string databasename, string query);

เช่น

```
$dbname = "Gift";

$sql = "select * from Shop";

$result = mysql_db_query($dbname,$sql);
```

การนำภาษา SQL มาใช้ในฐานข้อมูล MySQL

ฟังก์ชัน mysql_free_result()

เป็นฟังก์ชันสำหรับคืนหน่วยความจำให้กับระบบ เพื่อใช้หน่วยความจำให้มีประสิทธิภาพมากที่สุด ถ้ามีการใช้ตัวแปรมากๆ แล้วไม่มีการคืนหน่วยความจำจะส่งผลให้หน่วยความจำเต็มและมีผลต่อการทำงานของระบบได้

mysql_free_result(int result);

result หมายถึง ค่าที่ได้จากการใช้คำสั่งคิวรี เช่น

```
mysql_free_result($result);
```

การนำภาษา SQL มาใช้ในฐานข้อมูล MySQL

ฟังก์ชัน mysql_create_db()

เป็นฟังก์ชันสำหรับสร้างฐานข้อมูลใหม่

mysql_create_db(string databasename, [int database_connect]);

databasename คือ ชื่อฐานข้อมูลที่ต้องการสร้างใหม่

database_connect คือ ตัวแปรที่ใช้เชื่อมต่อกับฐานข้อมูล จะกำหนดหรือไม่ก็ได้

```
mysql_create_db("BookDB", $conn);
```

การนำภาษา SQL มาใช้ในฐานข้อมูล MySQL

ฟังก์ชัน mysql_fetch_array()

เป็นฟังก์ชันที่ใช้สำหรับดึงค่าผลลัพธ์ของฐานข้อมูลเก็บไว้ในอาร์เรย์ ผลลัพธ์ที่คืนออกมาจากฟังก์ชันนี้ จะเป็นข้อมูลอาร์เรย์ที่มีสมาชิกเท่ากับจำนวนคอลัมน์ของตาราง

mysql_fetch_array(int result);

จากการใช้ฟังก์ชันนี้ จะเป็นการอ่านค่าและถ่ายค่าลงตัวแปรอาร์เรย์ทีละ 1 รายการ หากเราต้องการแสดงค่าของข้อมูลไปเรื่อยๆ จนกว่าจะครบทุกรายการที่มีในตารางผลลัพธ์ ก็จะต้องกำหนดคำสั่งให้วนรอบการทำงานของฟังก์ชัน เช่น

```
...
while($row = mysql_fetch_array($result))
{
        echo $row[0], $row[2],.....,$row[n-1];
}
...
```

การนำภาษา SQL มาใช้ในฐานข้อมูล MySQL

ฟังก์ชัน mysql_fetch_row()

เป็นฟังก์ชันที่ใช้สำหรับเลื่อนตำแหน่งของตัวชี้ข้อมูลไปยังเรคอร์ดถัดไป

```
mysql_fetch_row(int result);
```

การนำภาษา SQL มาใช้ในฐานข้อมูล MySQL

ฟังก์ชัน mysql_num_fields()

เป็นฟังก์ชันที่ใช้ในการหาจำนวนคอลัมน์ที่มีทั้งหมด

mysql_num_fields(int result);

ผลลัพธ์ที่คืนออกมาจากฟังก์ชันนี้ เป็นชนิดตัวเลข ได้แก่ จำนวนคอลัมน์ทั้งหมดของตาราง เช่น

```
$numfield = mysql_num_fields$ result);
while($row = mysql_fetch_array($result))
{
        for($i=0; $i<$numfield; $i++)
            echo $row[$i];
}
```

การนำภาษา SQL มาใช้ในฐานข้อมูล MySQL

ฟังก์ชัน mysql_num_rows()

เป็นฟังก์ชันที่ใช้สำหรับคำนวณหาจำนวนแถวหรือจำนวนรายการทั้งหมด

```
mysql_num_rows(int result);
```

ผลลัพธ์ที่คืนออกมาจากฟังก์ชันนี้ เป็นข้อมูลชนิดตัวเลข ได้แก่ จำนวนรายการทั้งหมดของตารางผลลัพธ์

การอัปโหลดเว็บเพจเข้าสู่ระบบอินเตอร์เน็ต

วิธีการคือ เมื่อสร้างเว็บเพจสำเร็จแล้ว ก็ถึงขั้นตอนของการนำเว็บเพจไปฝังหรือฝากไว้ที่คอมพิวเตอร์แม่ข่าย หรือเว็บเซิร์ฟเวอร์ของ ISP ที่เราเป็นสมาชิกอยู่ หรืออาจจะมี Server เป็นของตัวเองเพื่อให้ทุกคนที่เป็นสมาชิกอินเตอร์เน็ตมองเห็นเว็บเพจของเรา ด้วยวิธีการ Upload หรือทำการ Transfer File ซึ่งการอัปโหลด (Upload) คือการก๊อปปี้ไฟล์จากเครื่องพีซีของเราไปไว้ที่เครื่อง Host โดยใช้ FTP (File Transfer Protocal) เป็นโปรโตคอลที่ใช้ในการถ่ายโอนข้อมูลระหว่างเครื่องพีซีและเครื่องคอมพิวเตอร์ที่เป็น Host สำหรับเครื่องพีซีจะต้องติดตั้งซอฟแวร์ในการอัปโหลดไฟล์ จากนั้นก็ทำการอัปโหลดไฟล์ไปไว้ในไดเร็กทอรีของตัวเอง

การอัปโหลดเว็บเพจเข้าสู่ระบบอินเตอร์เน็ต

ที่หน้าจอด้านขวาจะเป็นส่วนของเซิร์ฟเวอร์ และทางซ้ายคือฝั่งพีซี การอัปโหลดไฟล์ทำได้โดยการเลือกไฟล์ที่ต้องการอัปโหลด แล้วคลิกที่รูปลูกศรชี้ขึ้นที่อยู่บนแถบเมนูบาร์หรือดับเบิ้ลคลิกไฟล์ที่ฝั่งพีซีหรือคลิกที่ไฟล์ แล้วลากเมาส์ไปยังด้านเซิร์ฟเวอร์ โปรแกรมจะรายงานผลการอัปโหลดในทุกระยะ จนกระทั่งการอัปโหลดเสร็จสมบูรณ์ และหากเราต้องการสร้างไดเร็กทอรี ก็สามารถทำได้โดยคลิกเมาส์ขวาที่ฝั่งเซิร์ฟเวอร์ แล้วเลื่อนเมาส์ไปที่ Make new directory จะปรากฏหน้าจอ Create new dir ให้ใส่ชื่อไดเร็กทอรีใหม่ แล้วคลิก OK หากต้องการอัปโหลดไฟล์ไปไว้ในไดเร็กทอรีใหม่ ก็ดับเบิ้ลคลิกที่ชื่อไดเร็กทอรีที่สร้างไว้ แล้วอัปโหลดไฟล์ด้วยวิธีเดิม

การจัดสร้างไดเร็กทอรีเป็นเว็บเพจย่อย

จากหลักการข้างต้นนี้ เราสามารถจัดสร้างไดเร็กทอรีย่อย เพื่อจัดสร้างเป็น URL ย่อยสำหรับการเรียกเข้าถึงโดยตรง เพื่อใช้ประโยชน์ในการจัดสร้างร้านค้าย่อยหรือสร้างเว็บเพจย่อย โดยไม่ต้องคีย์ชื่อไฟล์ ก็สามารถทำได้โดยกำหนดชื่อไฟล์ ไฟล์แรก ชื่อ index.html

การตั้งชื่อเรียกอยู่ภายใต้ไดเร็กทอรี มีข้อดีในการนำมาใช้เรียกชื่อร้านค้าย่อยที่ร่วมอยู่ในห้างออนไลน์เดียวกัน ทำให้เกิดภาพลักษณ์ที่ดีเพราะชื่อที่เรียกไม่ยาวจนเกินไป และเป็นการใช้ชื่อร่วมกันอันทำให้เกิดความมั่นใจต่อผู้ซื้อ อย่างไรก็ตาม หากจะให้มีชื่อเรียกเป็นของตนเอง โดยส่วนใหญ่ก็มักจะไปจดชื่อโดเมนเป็นของตนเอง ซึ่งชื่อเหล่านี้ถือเป็นตรายี่ห้อสินค้าอย่างหนึ่ง ทำให้กลุ่มเป้าหมายจดจำได้ง่าย และเมื่อมีชื่อเสียงก็สามารถกลายเป็นทรัพย์สินทางปัญญาอย่างหนึ่งด้วย

วันนี้ ให้ นศ. 5/2 ออกแบบหน้า Home Page (5 คะแนน) ส่งเวลา 10.30 น.

1. ร้านบ้านขนมไทย
2. ร้านโลมาสปา
3. ร้านดอกไม้
4. ร้านหมวกปักชื่อ
5. ร้านรองเท้ากีฬามือสอง
6. B&N IT
7. ร้านแหวน
8. ร้านขายอุปกรณ์เสริมโทรศัพท์
9. ร้านขายเครื่องครัว
10. ร้ายขายสัตว์เลี้ยงและอุปกรณ์
11. ร้ายขายวิกผมออนไลน์
12. ร้ายขายจักรยาน
13. ร้ายขายเสื้อคู่
14. ร้านขายอุปกรณ์กีฬา
15. ร้านเต็นท์รถมือสอง
16. ร้ายขายเครื่องดนตรี
17. ร้านหนังสื่อการ์ตูน
18. ร้านขายนาฬิกา
19. ร้านขายร้องเท้า Converse

ให้ นศ. ใช้ WordPress ในการพัฒนาเท่านั้น (ลองศึกษาด้วยตัวเองดู ให้เวลาศึกษา 2 ชั่วโมง)

วันนี้ ให้ นศ. 5/3 ออกแบบหน้า Home Page (5 คะแนน) ส่งเวลา 14.00 น.

1. ร้านขายเครื่องดื่มน้ำสมุนไพรออนไลน์
2. ร้านขายจักรยานยนต์ บุญมียานยนต์ จำกัด
3. ร้านขายขนมไทยโบราณ
4. ธุรกิจท้องเที่ยวของบริษัท R.A
5. ร้านขายหนังดีวีดี
6. ร้านขายกาแฟสด
7. ร้านเค้ก & เบอรี่

ให้ นศ. ใช้ WordPress ในการพัฒนาเท่านั้น (ลองศึกษาด้วยตัวเองดู ให้เวลาศึกษา 2 ชั่วโมง)

The End

Chapter 3