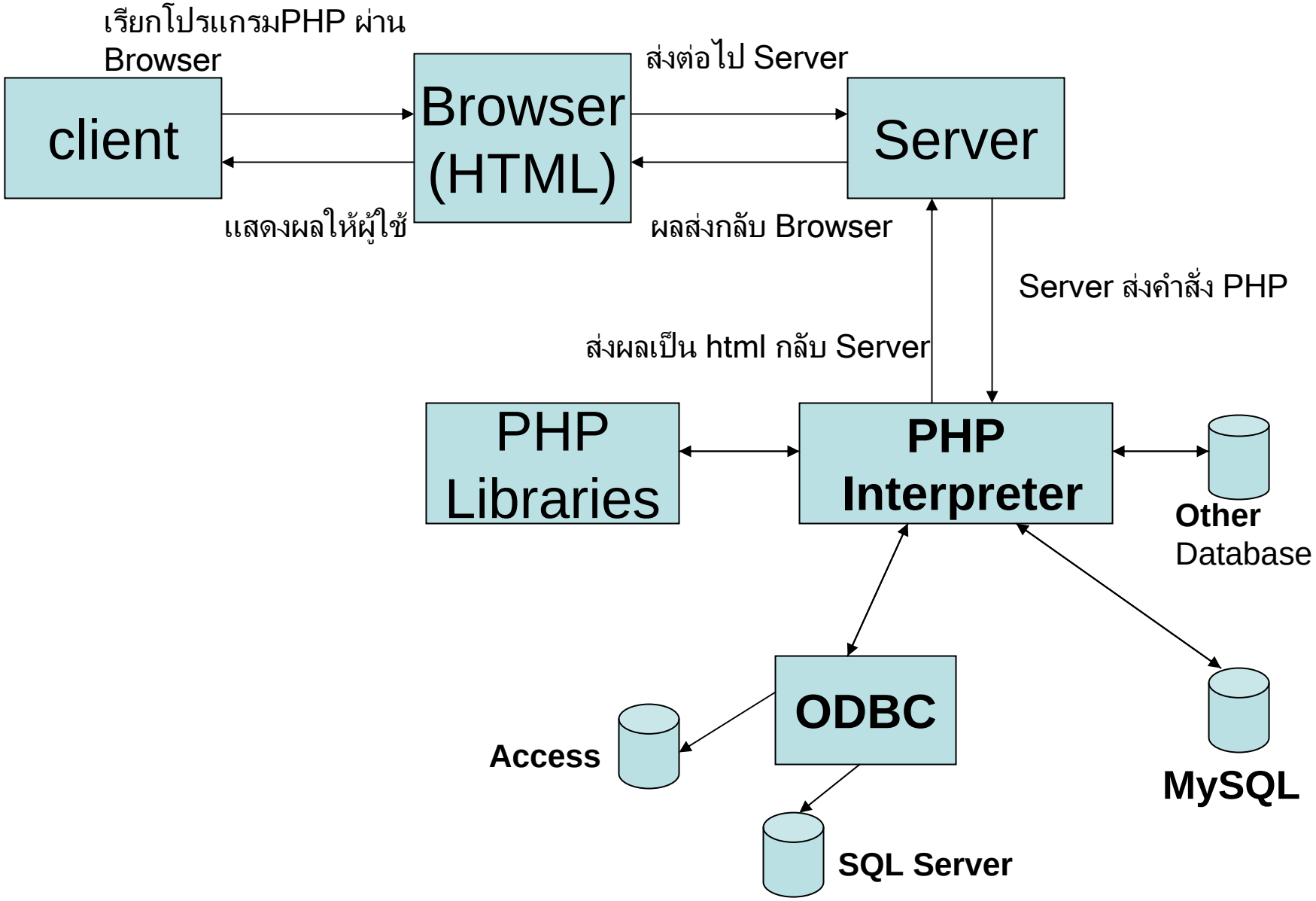
PHP

- Professional Home Page
- PHP Hypertext Processor
- Server Side Script
- www.php.net

จุดเด่น PHP

- Free
- Speed
- Open Source
- Crossable Platform
- Database Access
- Protocol Support
- Library
- Easy

หลักการทำงานของ PHP



รูปแบบการเขียนสคริปต์ PHP

- ทั่วไป (แบบภาษา SGML)

  <?

  คำสั่งในภาษา PHP;

  ?>

- ลักษณะภาษา XML

  <?php

  คำสั่งในภาษา PHP;

  ?>

รูปแบบการเขียนสคริปต์ PHP (ต่อ)

- ลักษณะ Javascript

  <Script Language=“php”>

  คำสั่งในภาษา PHP;

  </script>

- ลักษณะภาษา ASP

  <%

  คำสั่งในภาษา PHP;

  %>

รูปแบบการเขียนสคริปต์ PHP (ต่อ)

- ลักษณะพิเศษ

  <%=$ตัวแปร;

  คำสั่งในภาษา PHP;

  %>

หมายเหตุ 1. ทุกคำสั่งใน PHP ต้องปิดท้ายด้วย ;

2. คำสั่งจะเป็นตัวพิมพ์เล็กหรือใหญ่ก็ได้

3. ทดสอบโปรแกรม phpex_01.php

ชนิดข้อมูล

ชนิด	คำอธิบาย
Integer	เลขจำนวนเต็ม
Floating point Numbers	เลขจำนวนจริง
String	ข้อความ
Array	อาร์เรย์ หรือ แถวลำดับ
Object	ข้อมูลลักษณะออปเจ็กต์
Type juggling	ข้อมูลลักษณะเฉพาะผู้ใช้เติมเข้ามา

ตัวแปร (Variable)

- ชื่อแทนข้อมูล

  $varname=value;

- การตั้งชื่อ ขึ้นต้นด้วย $ และตามด้วยตัวอักษร ยาว 1 ถึง 255 ตัว เช่น

  $Name = “Panutson”;

- นอกจากนี้มีตัวแปรระบบ เช่น

  $SCRIPT_NAME

  เป็นตัวแปรของระบบแสดงชื่อ script PHP ที่กำลังทำงาน

ค่าคงที่ (Constant)

- ค่าคงที่ที่กำหนดโดย PHP (Built-in Constant) เช่น

  TRUE ค่าคงที่เป็นจริง

  PHP_VERSION ค่าคงที่แสดงเวอร์ชันของ PHP

- ค่าคงที่ที่ผู้ใช้กำหนดเอง (User-defined Constant)

  DEFINE(ConstName, Value);

  เช่น

  DEFINE(“NewLine”,”
\n”);

ตัวดำเนินการ (Operator)

- Arithmetic Operators

+	บวก	$c=$a + $b;
-	ลบ	$c=$a - $b;
*	คูณ	$c=$a * $b;
/	หาร	$c=$a / $b;
%	หารเอาเศษ	$c=$a % $b;

ตัวดำเนินการ (Operator) (ต่อ)

- Assignment Operators

=	กำหนดค่า	$c = 1;
+=	เพิ่มค่า	$c += 1;
-=	ลบค่า	$c -= 1;
*=	คูณค่า	$c *= 1;
/=	หารค่า	$c /= 1;

ตัวดำเนินการ (Operator) (ต่อ)

- Bitwise Operators

&	And	$a & $b
|	Or	$a | $b
~	Not	~$a
<<	Shift Left	$a << $b
>>	Shift right	$a >> $b

ตัวดำเนินการ (Operator) (ต่อ)

- Comparison Operators

==	เท่ากับ	$a == $b
!=	ไม่เท่ากับ	$a != $b
<	น้อยกว่า	$a < $b
>	มากกว่า	$a > $b
<=	น้อยกว่าหรือเท่ากับ	$a <= $b
>=	มากกว่าหรือเท่ากับ	$a >= $b

ตัวดำเนินการ (Operator) (ต่อ)

- Incrementing/Decrementing Operators

++$a	เพิ่มค่าก่อน
$a++	เพิ่มค่าทีหลัง
--$a	ลดค่าก่อน
$a--	ลดค่าทีหลัง

- String Operator (.)

  เช่น

```
$a = “Hello...”;
$b = $a.”Students”;
```

ตัวดำเนินการ (Operator) (ต่อ)

- Logical Operators

and	และ	$a and $b
&&	และ	$a && $b
or	หรือ	$a or $b
||	หรือ	$a || $b
xor	exclusive or	$a xor $b
!	ไม่	!$a

การใส่หมายเหตุ (Comment)

- ใช้เครื่องหมาย // หรือ # ตามด้วยข้อความหมายเหตุ

  เช่น

  echo “Hello..
”; //แสดงข้อความ Hello..
- ถ้ามีหมายเหตุหลายบรรทัด ใช้ /* ตามด้วยข้อความ แล้วปิดด้วย */

  เช่น

  /* หมายเหตุบรรทัดที่ 1

  หมายเหตุบรรทัดที่ 2

  */

คำสั่ง echo หรือ print

- คำสั่งแสดงข้อความหรือข้อมูล

  echo “ข้อความ”; หรือ echo ตัวแปร;

  print “ข้อความ”; หรือ print ตัวแปร;

  เช่น

  ```
  $Fname=“Panutson”;
  echo “Hello...”.$Fname;
  print $Fname,” สวัสดี”;
  ```

- ทดสอบโปรแกรม phpex_02.php ถึง phpex_05.php

คำสั่งเกี่ยวกับเงื่อนไข

- If...elseif...else

  รูปแบบคำสั่ง

```
IF (condition1)
    {instructions;}
ELSEIF (condition2)
    {instructions;}
....
ELSE
    {instructions;}
```

คำสั่งเกี่ยวกับเงื่อนไข (ต่อ)

- **Switch...Case**

  <u>รูปแบบคำสั่ง</u>

```
Switch (variable)
        {
                Case expr1;
                        {instructions;
                         break;}
                Case expr2;
                        {instructions;
                         break;}
                ....
                default;
                         {instructions;}
        }
```

คำสั่งการทำงานวนซ้ำ (Loop)

- คำสั่ง For

รูปแบบคำสั่ง

```
For ([start]; [condition]; [step])
    {
        instructions;
    }
```

โดย start คือค่าเริ่มต้น condition คือเงื่อนไข

step คือค่าเพิ่มขึ้นหรือลด ถ้าไม่ระบุถือว่าเพิ่มค่าครั้งละ 1

คำสั่งการทำงานวนซ้ำ (Loop) (ต่อ)

- คำสั่ง While

รูปแบบคำสั่ง

```
While (condition)
    {
        instructions;
        [step];
    }
```

หมายเหตุ step ถ้ามี จะเป็นการกำหนดให้ค่าของ condition เปลี่ยนไป

คำสั่งการทำงานวนซ้ำ (Loop) (ต่อ)

- คำสั่ง Do...While

รูปแบบคำสั่ง

```
Do
    {
        instructions;
        [step];
    }
 While (condition)
```

คำสั่งการทำงานวนซ้ำ (Loop) (ต่อ)

- คำสั่ง Foreach

รูปแบบคำสั่ง

```
Foreach (array_var as value_var)
    {
        instructions;
    }
```

หมายเหตุ 1. เป็นการทำงานวนซ้ำ ตามตัวแปรอาร์เรย์

2. ทดสอบโปรแกรม phpex_12.php ซึ่งมีการใช้คำสั่ง While ร่วมกับ Function list() และ each() แสดงผลเหมือนคำสั่ง Foreach

คำสั่งควบคุมการทำงานอื่นๆ

- คำสั่ง Break คือหยุดการทำงาน ถ้าใช้กับคำสั่งวนซ้ำจะออกจากการวนซ้ำ รูปแบบคำสั่ง

  Break;

- คำสั่ง Continue คือทำงานต่อ
  ถ้าใช้กับคำสั่ง For จะกลับไปเพิ่มค่าให้กับตัวแปรทันที
  ถ้าใช้กับคำสั่ง While จะกลับไปทดสอบตามเงื่อนไขทันที
  รูปแบบคำสั่ง

  Continue;

หมายเหตุ ทดสอบโปรแกรม phpex_13.php และ phpex_14.php

คำสั่งควบคุมการทำงานอื่นๆ (ต่อ)

- คำสั่ง Include คือคำสั่งนำแฟ้มอื่นมาทำงานร่วมกับ script ที่กำลังทำงานอยู่ รูปแบบคำสั่ง

  Include(‘file’);

  file คือแฟ้มที่ต้องการนำมาแทรกใน script ที่ใช้งานอยู่

หมายเหตุ ทดสอบโปรแกรม phpex_15.php

ฟังก์ชันของ PHP

- ฟังก์ที่ผู้ใช้สร้างเอง
- ฟังก์ชันที่มากับ PHP เรียกใช้งานได้ทันที

ฟังก์ชันที่ผู้ใช้สร้างเอง

- ฟังก์ชันที่ไม่มีการส่งค่า

รูปแบบ คือ

```
Function Func_Name()
    {
        Instructions;
    }
```

เรียกใช้งานโดย Func_Name();

หมายเหตุ ทดสอบโปรแกรม phpex_16.php

ฟังก์ชันของ PHP (ต่อ)

ฟังก์ชันที่ผู้ใช้สร้างเอง (ต่อ)

- **ฟังก์ชันที่มีการส่งค่าระหว่างกัน**

รูปแบบ คือ

```
Function Func_Name(parameter)
    {
        return Instructions;
    }
```

เรียกใช้งานโดย Func_Name(parameter_value);

หมายเหตุ ทดสอบโปรแกรม phpex_17.php

ฟังก์ชันของ PHP (ต่อ)

ฟังก์ชันที่มากับ PHP เรียกใช้งานได้ทันที เช่น

ฟังก์ชัน	ความหมาย	หมายเหตุ
date()	วันเวลาปัจจุบัน	อาจกำหนดรูปแบบได้
time()	เวลาปัจจุบัน	
Abs(x)	ค่าสัมบูรณ์	
Cos(radain)	ค่า cosine	มุม radian
max(x, y,..)	ค่าสูงสุด	
min(x, y,..)	ค่าต่ำสุด	
log(x)	logarithm ฐานธรรมชาติ (e)	
rand()	ค่าสุ่ม (random no.)	

Format date

- a am/pm
- A AM/PM
- d 01-31
- D Mon-Sun
- F January-December
- g 1-12
- G 0-24
- h 01-12

- H 00-23
- i 00-59 (minute)
- J 1-31
- l Sunday-Satueday
- L Leap year 1 or 0
- m 01-12
- t จำนวนวันในเดือนนั้น
- Y 2005
- y 05

ฟังก์ชันของ PHP (ต่อ)

ฟังก์ชันที่มากับ PHP เรียกใช้งานได้ทันที (ต่อ) เช่น

ฟังก์ชัน	ความหมาย	หมายเหตุ
chdir(directory)	เปลี่ยนไปยังสาระบบที่ต้องการ	
mkdir(directory)	สร้างสาระบบ	
copy(sour,dest)	สำเนาแฟ้มข้อมูล	
rename(old,new)	เปลี่ยนชื่อแฟ้มข้อมูล	
mail(to,sub,mes,[h])	ส่ง mail	
chr(x)	แปลงรหัส ascii เป็นอักขระ	
substr(str,start,[len])	ตัดอักขระที่ต้องการออกมา	
Etc.		

ฟังก์ชันของ PHP (ต่อ)

ฟังก์ชันที่มากับ PHP เรียกใช้งานได้ทันที ด้านแฟ้มข้อมูล เช่น

ฟังก์ชัน	ความหมาย
Fopen(filename,mode)	เปิดแฟ้มข้อมูล
Fgetc(file_handle)	อ่านข้อมูลทีละอักขระ
Feof(file_handle)	ตรวจสอบ end-of-file
Fgets(file_handle,len)	อ่านข้อมูลทีละบรรทัดหรือ ตาม len
Rewind(file_handle)	เลื่อนพอยเตอร์ไปต้นแฟ้มข้อมูล
Fgetss(file_handle,len)	อ่านข้อมูลทีละบรรทัดหรือ ตาม len โดยตัดแท็ก html
Fclose(file_handle)	ปิดแฟ้มข้อมูล

ฟังก์ชันของ PHP (ต่อ)

ฟังก์ชันที่มากับ PHP เรียกใช้งานได้ทันที (ต่อ) ด้านแฟ้มข้อมูล เช่น

ฟังก์ชัน	ความหมาย
File(filename)	อ่านแฟ้มข้อมูลทีละบรรทัดบันทึกลง array
Fputs(filename,text)	เขียนข้อความลงแฟ้ม ถ้าเดิมมีข้อมูลอยู่จะลบข้อมูลเดิม
Fread(file_handle,len)	อ่านข้อมูลจากแฟ้ม ระบุความยาว len ได้
Fseek(file_handle,offset)	เลื่อนพอยเตอร์ไปข้างหน้าตามต้องการ
Readfile(filename)	อ่านข้อมูลจากแฟ้มข้อมูลโดยไม่ต้องเปิดแฟ้ม
Unlink(filename)	ลบแฟ้มข้อมูล
etc.	

Int fopen(string filename, string mode)

- filename ชื่อไฟล์ที่ต้องการเปิดหรือสร้างขึ้นใหม่
- Mode วิธีการเปิดไฟล์ได้แก่
- ‘r’ อ่านได้อย่างเดียวตัวชี้จะเริ่มตำแหน่งเริ่มต้นไฟล์
- ‘r+’ อ่านและเขียนตัวชี้จะเริ่มตำแหน่งเริ่มต้นไฟล์
- ‘w’ เขียนได้อย่างเดียวตัวชี้จะเริ่มตำแหน่งเริ่มต้นไฟล์ถ้ามีไฟล์อยู่แล้วจะลบข้อมูลเดิมทั้งหมด ถ้าไม่มีจะสร้างขึ้นใหม่
- ‘w+’ อ่านและเขียนได้อย่างเดียวตัวชี้จะเริ่มตำแหน่งเริ่มต้นไฟล์ถ้ามีไฟล์อยู่แล้วจะลบข้อมูลเดิมทั้งหมด ถ้าไม่มีจะสร้างขึ้นใหม่

- ‘a’ เขียนได้อย่างเดียวตัวชี้จะเริ่มตำแหน่งสิ้นสุดไฟล์ ถ้าไม่มีจะสร้างขึ้นใหม่
- ‘a+’ อ่านและเขียน ตัวชี้จะเริ่มตำแหน่งสิ้นสุดไฟล์ ถ้าไม่มีจะสร้างขึ้นใหม่
- ตัวอย่าง

```
$fxx = @fopen(“c:/php_nuj/test.php”,”r”);
```

การอ่านไฟล์

- String fgets (int fp [, int length])

  fp ไฟล์พอยเตอร์

  length ความยาวของข้อความโดยเริ่มจากตำแหน่งตัวชี้เท่ากับ length - 1 หากม่ระบุ 1024

- String fgetc(int fp)

  fp ไฟล์พอยเตอร์

- หากพบ EOFจะคืนค่า False

- String fgetss(int fp, int length [, string allowable_tags])

  fp ไฟล์พอยเตอร์

  length ความยาวของข้อความโดยเริ่มจากตำแหน่งตัวชี้เท่ากับ length - 1 หากม่ระบุ 1024

  allowable_tags ระบุแท็ก HTML ที่ต้องการแสดง

Int fputs(int fp, string str [,int length])

fp	ไฟล์พอยเตอร์
str	ข้อความที่ต้องการเขียนลงไฟล์
length	ความยาวของข้อความที่เขียน

- String fread(int fp, int length)

fp	ไฟล์พอยเตอร์
length	ความยาวของข้อความที่อ่าน

PHP กับฐานข้อมูล

- dBase
- Informix
- Microsoft SQL Server
- ODBC โดยใช้ฐานข้อมูล Microsoft Access
- Oracle
- MySQL
- etc.

PHP กับฐานข้อมูล MySQL

ฟังก์ชันสำหรับการติดต่อฐานข้อมูล MySQL

ฟังก์ชัน	ความหมาย
mysql_affected_rows	จำนวนระเบียนในการใช้ ก่อนหน้า
mysql_change_user	เปลี่ยนชื่อผู้ใช้
mysql_close	ปิดการติดต่อ MySQL
mysql_connect	ติดต่อกับ MySQL Server
mysql_create_db	สร้างฐานข้อมูล MySQL
mysql_data_seek	เลื่อนพอยเตอร์ภายใน
mysql_db_query	ส่งคิวรีไปยัง MySQL

PHP กับฐานข้อมูล MySQL (ต่อ)

ฟังก์ชันสำหรับการติดต่อฐานข้อมูล MySQL

ฟังก์ชัน	ความหมาย
mysql_drop_db	ลบฐานข้อมูล MySQL
mysql_errno	ค่าบอกข้อผิดพลาดจาก MySQL
mysql_error	ข้อความบอกข้อผิดพลาดจาก MySQL
mysql_fetch_array	ค่าระเบียนเก็บใน array
mysql_fetch_field	ค่าข้อมูล field เก็บเป็น object
mysql_fetch_lengths	ค่าขอบเขตแต่ละผลลัพธ์
mysql_fetch_object	ค่าผลลัพธ์เป็น object

PHP กับฐานข้อมูล MySQL (ต่อ)

ฟังก์ชันสำหรับการติดต่อฐานข้อมูล MySQL

ฟังก์ชัน	ความหมาย
mysql_fetch_row	ค่าผลลัพธ์ระบุลง array
mysql_field_name	ค่าผลลัพธ์เป็นชื่อ field
mysql_field_seek	กำหนดพอยเตอร์ชี้ไปยัง field
mysql_field_table	ค่าผลลัพธ์เป็นชื่อตารางจากชื่อ field
mysql_field_type	ผลลัพธ์เป็นประเภทของ field
mysql_field_len	ขนาดของ field ที่กำหนด
mysql_free_result	ทำให้หน่วยความจำว่างมากขึ้น

PHP กับฐานข้อมูล MySQL (ต่อ)

ฟังก์ชันสำหรับการติดต่อฐานข้อมูล MySQL

ฟังก์ชัน	ความหมาย
mysql_insert_id	ได้ค่า id ที่สร้างจากคำสั่ง insert
mysql_list_fields	แสดงชื่อ fields
mysql_list_dbs	แสดงชื่อฐานข้อมูลที่มีใน MySQL Server
mysql_list_tables	แสดงชื่อตารางในฐานข้อมูล MySQL
mysql_num_fields	ค่าผลลัพธ์เป็นจำนวน field
mysql_num_rows	ค่าผลลัพธ์เป็นจำนวนระเบียน
mysql_pconnect	เปิดการติดต่อกับ MySQL Server

PHP กับฐานข้อมูล MySQL (ต่อ)

ฟังก์ชันสำหรับการติดต่อฐานข้อมูล MySQL

ฟังก์ชัน	ความหมาย
mysql_query	ส่ง query ไป MySQL
mysql_result	ผลลัพธ์เป็นข้อมูล
mysql_select_db	เลือกฐานข้อมูล MySQL
mysql_tablename	ให้ชื่อตาราง

การติดต่อฐานข้อมูลและแสดงผล

- เริ่มติดต่อฐานข้อมูล
- เลือกฐานข้อมูลที่ต้องการ
- กำหนดคำสั่ง **SQL** และสั่งให้ทำงาน
- เก็บข้อมูลลงตัวแปร **array**
- นับจำนวนระเบียน
- แสดงผลทาง **browser**
- ปิดการติดต่อกับฐานข้อมูล **MySQL**

เริ่มติดต่อฐานข้อมูล

- ใช้ฟังก์ชัน ดังนี้

```
mysql_connect($hostname, $user,
        $password) or die(“message”);
```

เช่น

```
$hostname = “localhost”;
$user = “teach”;
$password = “”;
mysql_connect($hostname, $user,
        $password) or die(“ติดต่อฐานข้อมูลไม่ได้”);
```

เลือกฐานข้อมูลที่ต้องการ

- ใช้ฟังก์ชัน ดังนี้

mysql_select_db($dbname) or die(“message”);

เช่น

```
$dbname = “bookshop”;
mysql_select_db($dbname) or die(“เลือกฐานข้อมูลไม่ได้”);
```

กำหนดคำสั่ง SQL และสั่งให้ทำงาน

- ใช้ฟังก์ชัน ดังนี้

```
mysql_query($sql)
```

เช่น

```
$tblname = “book”;
$sql = “select * from $tblname”;
mysql_query($sql);
```

เก็บข้อมูลลงตัวแปร **array**

- ใช้ฟังก์ชัน ดังนี้

  mysql_fetch_array($dbname)

- โดยมีรูปแบบคำสั่ง เช่น

$dbquery = mysql_db_query($dbname, “select * from $tblname”);

$result = mysql_fetch_array($dbname);

หมายเหตุ $result เป็น array ดังนั้น

$result[0] คือสมาชิกตัวแรกของ array

หรือ $result[“barcode”] คือ ค่าของ field ชื่อ barcode

นับจำนวนระเบียน

- ใช้ฟังก์ชัน ดังนี้

  mysql_num_rows($dbquery)
- จากการเก็บผลลัพธ์ใน array ตามจำนวนระเบียนที่มีอยู่ ดังนั้นก่อนแสดงผลอาจจำเป็นต้องทราบจำนวนระเบียน
- รูปแบบการใช้งาน เช่น

```
$dbquery = mysql_db_query($dbname, “select * from $tblname”);
$num_rows = mysql_num_rows($dbquery);
```

แสดงผลทาง browser

- รูปแบบคำสั่ง ดังนี้เช่น

```
$i=0;
while ($i < $num_rows)
{
$result = mysql_fetch_array($dbquery);
echo “<tr> <td>$result[“barcode”]</td>
<td>$result[“title”]</td><td>$result[“author”]</td>
<td>$result[“price”]</td> <td>$result[“type”]</td>
 <td>$result[“year”]</td></tr>;
$i++;
}
```

ปิดการติดต่อกับฐานข้อมูล MySQL

- ใช้ฟังก์ชัน ดังนี้

  mysql_close()

หมายเหตุ ทดสอบตัวอย่างโปรแกรม phpex_24.php