

PHP

1. BASICS

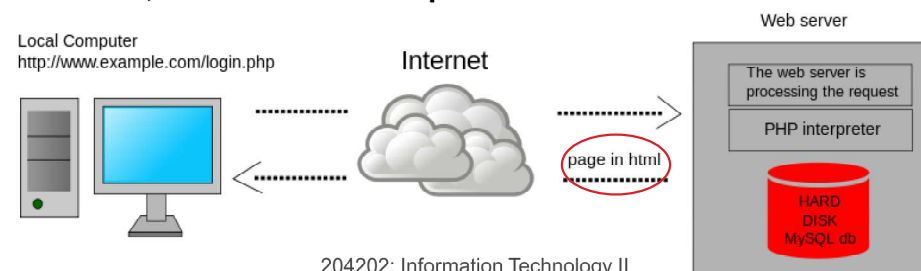
รวบรวมโดย อ. กิตติพิชญ์ คุปตะวาณิช

PHP Basics: Syntax

- PHP standard ที่ใช้ในปัจจุบันเป็น PHP 5.x หรือ PHP 5
- เราสามารถใส่ PHP script ไว้ที่ส่วนไหนของไฟล์ PHP ก็ได้ โดยส่วนที่เป็น PHP script จะมีเครื่องหมาย `<?php ?>` ล้อมรอบโดยการประมวลผล จะเกิดขึ้นกับ code ที่อยู่ระหว่าง Tag เปิดและปิดนี้เท่านั้น
- ในกรณีที่ไฟล์นั้นไม่มี content อื่น ๆ นอกจาก PHP code (pure PHP Code) ไม่ควรต้องใส่ tag ปิด `?>`

PHP Introduction

- PHP เป็น Server-side Scripting Language
- PHP code ในไฟล์นามสกุล .php (PHP file) จะถูกประมวลผลบน server และแสดงผลเป็น HTML ไฟล์
- นอกจาก PHP code แล้ว PHP ไฟล์ สามารถประกอบด้วย HTML, CSS และ JavaScript อีกด้วย



PHP Basics: Syntax [2]

- Content ของไฟล์ที่อยู่นอก Tag เปิดและปิดจะไม่ได้รับการประมวลผล
 - ทำให้เราสามารถแทรก content อื่น ๆ เช่น HTML ลงในไฟล์ PHP ได้
- ```
<p>This is going to be ignored by PHP</p>
<?php echo 'while this is going to be parsed.'; ?>
<p>This will also be ignored by PHP</p>
```
- `echo` เป็นคำสั่ง PHP ที่ใช้แสดงข้อความที่อยู่ในเครื่องหมายคำพูด (e.g. "text" หรือ 'text' ) ออกไปที่ browser

# The echo Command

```
<!DOCTYPE html>
<html>
 <head>
 <meta charset="utf-8" />
 </head>
 <body>
 <h1>My first PHP page</h1>
 <?php
 echo "Hello, my name is John";
 ?>
 </body>
</html>
```

My first PHP page

Hello, my name is John

# PHP Basics: Syntax [3]

- แต่ละบรรทัดของคำสั่งในภาษา PHP ต้องจบด้วยเครื่องหมาย ;
  - <?php echo 'My name is John'; ?>
- ชื่อคำสั่งหรือฟังก์ชัน (function) ในภาษา PHP มีลักษณะ case insensitive คือไม่มีความแตกต่างของตัวพิมพ์เล็กหรือพิมพ์ใหญ่ (echo = EcHo = ECHO...)
- แต่ตัวแปรในภาษา PHP เป็น แบบ case sensitive

□ \$var

□ \$VaR

□ \$vAr

Three **different** variables

# PHP Basics: Variables

- Variable หรือตัวแปรใช้เพื่อเก็บข้อมูล ในโปรแกรม เช่น

```
<?php
 $x = 5;
 $y = 6;
 $z = $x + $y;
 echo $z; // print out z
?>
```

```
x = 5
y = 6
z = x + y
z = ...
```

- ลักษณะการ แทนค่า หรือการคำนวณ เหมือนในวิชาคณิตศาสตร์
- ตัวแปรใน PHP ต้องขึ้นต้นด้วย \$ เสมอ
- ชื่อตัวแปรต้องประกอบด้วยตัวอักษร (aA-zZ) ตัวเลข (0-9) และ underscore (\_)

\*Notice the comments

- ขึ้นต้นด้วยตัวอักษร หรือ underscore เท่านั้น

# PHP Basics: Comments

- Comments หมายถึงส่วนใด ๆ ของ code ที่ไม่ถือเป็นคำสั่งแต่ใส่ไว้เพื่อเป็นการอธิบาย code เพื่อสะดวกในการทำความเข้าใจ Code

- ตัวอย่าง

```
<?php
 echo 'This is a test'; // This is a one line comment
 /* This is a multi line comment
 yet another line of comment */
 echo 'This is yet another test';
 echo 'One Final Test'; # This is a one-line comment
?>
```

## PHP Basics: Variables [2]

### Variable จากแหล่งภายนอก

- เมื่อมีการส่ง Form ด้วยวิธี GET หรือ POST ไปยัง php script เราสามารถเข้าถึงค่าต่าง ๆ หรือ ข้อมูลที่ส่งมาได้โดยการเรียก

variable `$_POST['variable_name']` หรือ `$_GET['variable_name']`

### ตัวอย่าง HTML form

```
<form action="greeting.php" method="post">
 Name: <input type="text" name="username" />

 Email: <input type="text" name="email" />

 <input type="submit" name="submit" value="Submit me!" />
</form>
```

## PHP Basics: Variables [3]

### Variable จากแหล่งภายนอก [2]

```
<head>
 <meta charset="utf-8" />
 <title>Greetings!</title>
</head>
<body>
 Hello <?php echo $_POST["username"]; ?>

 Your email address is: <?php echo $_POST["email"]; ?>
</body>
```

- เราสามารถใส่ " " หรือ ' ' ก็ได้ในการดึงค่าที่ถูกส่งมากับฟอร์มผ่าน

variable `$_POST`  
(หรือ `$_GET`)

```
Hello, John
Your email address is: john@email.com
```

## PHP Basics: Operators

### Arithmetic

Example	Name	Result
<code>-\$a</code>	Negation (นิเสธ)	Opposite of <code>\$a</code> .
<code>\$a + \$b</code>	Addition (การบวก)	Sum of <code>\$a</code> and <code>\$b</code> .
<code>\$a - \$b</code>	Subtraction (การลบ)	Difference of <code>\$a</code> and <code>\$b</code> .
<code>\$a * \$b</code>	Multiplication (การคูณ)	Product of <code>\$a</code> and <code>\$b</code> .
<code>\$a / \$b</code>	Division (การหาร)	Quotient of <code>\$a</code> and <code>\$b</code> .
<code>\$a % \$b</code>	Modulus (การหารเก็บเศษ)	Remainder of <code>\$a</code> divided by <code>\$b</code> .

## PHP Basics: Operators [2]

### Comparison

Example	Name	Result
<code>\$a == \$b</code>	Equal	<b>TRUE</b> if <code>\$a</code> is equal to <code>\$b</code> after type juggling.
<code>\$a != \$b</code> <code>\$a &lt;&gt; \$b</code>	Not equal	<b>TRUE</b> if <code>\$a</code> is not equal to <code>\$b</code> after type juggling.
<code>\$a &lt; \$b</code>	Less than	<b>TRUE</b> if <code>\$a</code> is strictly less than <code>\$b</code> .
<code>\$a &gt; \$b</code>	Greater than	<b>TRUE</b> if <code>\$a</code> is strictly greater than <code>\$b</code> .
<code>\$a &lt;= \$b</code>	Less than or equal to	<b>TRUE</b> if <code>\$a</code> is less than or equal to <code>\$b</code> .
<code>\$a &gt;= \$b</code>	Greater than or equal to	<b>TRUE</b> if <code>\$a</code> is greater than or equal to <code>\$b</code> .

## PHP Basics: Operators [3]

□ **Assignment:** การกำหนดค่าให้ variable ต่าง ๆ ทำได้โดยการ  
ใช้เครื่องหมายเท่ากับ =

□ การกำหนดค่า จะอยู่ในรูปแบบของการนำค่าที่อยู่ด้านขวา ของ  
operator ไปใส่ใน variable ที่อยู่ด้านซ้าย เช่น

```
$a = 3;
$b = "Hello ";
```

Assignment	Same as
<code>\$a += \$b</code>	<code>\$a = \$a + \$b</code>
<code>\$a -= \$b</code>	<code>\$a = \$a - \$b</code>
<code>\$a *= \$b</code>	<code>\$a = \$a * \$b</code>
<code>\$a /= \$b</code>	<code>\$a = \$a / \$b</code>
<code>\$a %= \$b</code>	<code>\$a = \$a % \$b</code>
...	...

`$a` ← 3

13

Ref: <http://us3.php.net/manual/en/language.operators.php>

## PHP Basics: Control Structure – if

□ **if** ใช้เพื่อกำหนดเงื่อนไขในรูปแบบ

```
if (expr)
statement
```

□ โดย คำสั่งในบรรทัด **statement** จะถูกทำเมื่อค่าที่อยู่ในวงเล็บ (**expr**) เป็น  
จริง

```
<?php
if ($a > $b)
 echo "a is bigger than b";
?>
```

14

Ref: <http://www.php.net/manual/en/language.control-structures.php>

## PHP Basics: Control Structure – else

□ **else** ใช้เพื่อกำหนดเงื่อนไขในรูปแบบ

```
if (expr)
statement_A
else
statement_B
```

```
<?php
if ($a > $b) {
 echo "a is greater than b";
} else {
 echo "a is NOT greater than b";
}
?>
```

\*Notice the curly brackets

15

Ref: <http://www.php.net/manual/en/language.control-structures.php>

## PHP Basics: Control Structure – elseif

□ **else** ใช้เพื่อกำหนด เงื่อนไขในรูปแบบ

```
if (expr)
statement_A
elseif (expr2)
statement_B
else
statement_C
```

\*Notice how **elseif** is written as one word

```
<?php
if ($a > $b) {
 echo "a is bigger than b";
} elseif ($a == $b) {
 echo "a is equal to b";
} else {
 echo "a is smaller than b";
}
?>
```

16

Ref: <http://www.php.net/manual/en/language.control-structures.php>

# PHP Basics: Control Structure Example

## □ Example: Letter grade assigning

```
<?php
 if ($score >= 80) {
 $grade = "A";
 } elseif ($score >= 70) {
 $grade = "B";
 } elseif ($score >= 60) {
 $grade = "C";
 } elseif ($score >= 50) {
 $grade = "D";
 } else {
 $grade = "F";
 }
?>
```

17

Ref: <http://www.php.net/manual/en/language.control-structures.php>

# References

## □ PHP Basic Syntax Reference

<http://us3.php.net/manual/en/language.basic-syntax.php>

## □ PHP Variable Reference

<http://us3.php.net/manual/en/language.variables.php>

## □ PHP Operators

<http://us3.php.net/manual/en/language.operators.php>

## □ PHP Control Structures

<http://www.php.net/manual/en/language.control-structures.php>

18