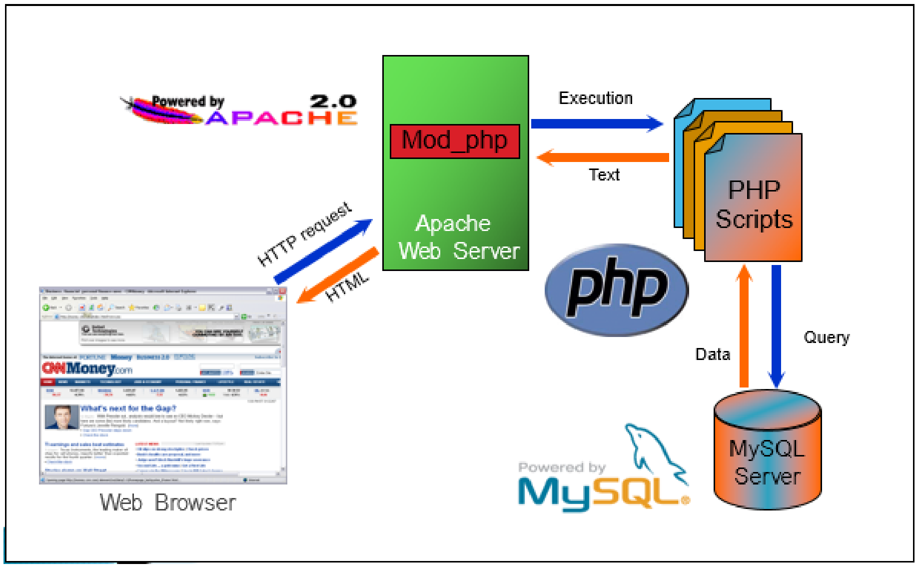
ໂຄງສ້າງການທຳງານຂອງ Web App



1. PHP 5 Syntax

1.1 ພື້ນຖານ PHP Syntax

ເນື່ອງຈາກພາສາ PHP (Hypertext Preprocessor) ສາມາດຂຽນກັບພາສາ HTML ໄດ້ ດັ່ງນັ້ນເມື່ອໃດທີ່ເຮົາຈະຂຽນ PHP ກໍ່ ຈະມີສ່ວນທີ່ຈະມາໃຈ້ແຍກໃຫ້ກັບຕົວແປພາສາ (Interpreter) ຮູ້ໂປຣແກຣມສ່ວນໃດເປັນຂອງ PHP ເຊິ່ງຈະໄດ້ນຳມາທຳການແປໂປຣແກຣມດັ່ງກ່າວ ໂດຍຮູບແບບຂອງ PHP syntax ມີດັ່ງນີ້:

<?php ……Code PHP………?> ເປັນແບບມາດຕະຖານ

<? ………Code PHP……….?> ແບບສັ້ນ

<% ………Code………%> ແບບຂອງ ASP

<SCRIPT language=“php”> ……Code…… </SCRIPT> : ແບບ script

ຮູບແບບແທັກທັງໝົດຂ້າງເທິງ ເຮົາສາມາດໃຊ້ງານໄດ້ກໍ່ຕໍ່ເມື່ອຄ່າຂອງ short_open_tag = on ໃນຟາຍ php.ini ຖືກກຳນົດໄວ້ເປັນ On ເຊິ່ງເປັນຄ່າປົກກະຕິທີ່ເຮົາຕິດຕັ້ງມາແຕ່ເລີ່ມຕົ້ນ ສ່ວນເປັນຮູບແບບແທັກ ASP ເຮົາຕ້ອງທຳການປ່ຽນຄ່າໃນສ່ວນຂອງ asp_tags=On ໃນຟາຍ php.ini ໃຫ້ເປັນ On ເສຍກ່ອນ ເຖິງຈະເພີ່ມຄວາມສາມາດໃນການໃຊ້ຮູບແບບແທັກ ASP ໄດ້

```
<html>
   <head><title>ເລີ່ມຕົ້ນຂຽນ PHP</title>
```

```
    </head>
    <body>
        <h1>My first PHP page</h1>
        <?php
            echo "ສະແດງຜົນດ້ວຍພາສາ PHP";
        ?>
    </body>
</html>
```

1.2 Comments in PHP

Comment ໝາຍເຖິງຄຳອະທິບາຍຂອງໂປຼແກຼມ ຈະບໍ່ມີຜົນເມື່ອໂປຼແກຼມທຳການປະມວນຜົນ, ມີຈຸດປະສົງເພື່ອໃຫ້ຜູ້ທີ່ທຳການອ່ານສະຫັດໂປຣແກຣມໄດ້ສະດວກຂຶ້ນ.

Comment ໃຊ້ເພື່ອ:

- ເພື່ອໃຫ້ຄົນອື່ນເຂົ້າໃຈໃນສິ່ງທີ່ຕົນເອງກະທຳຢູ່
- ເພື່ອເປັນການເຕືອນຕົວເອງໃນສິ່ງທີ່ຕົນເອງກະທຳຢູ່ ວ່າແຕ່ລະສ່ວນຂອງໂປຣແຣມເຮັດໜ້າທີ່ຫຍັງ

ໃນການໃສ່ຄຳອະທິບາຍຂອງໂປຣແກຣມສາມາດໃສ່ໄດ້ຫຼາຍຮູບແບບດັ່ງນີ້:

// ໃຊ້ໄດ້ສະເພາະແຖວດຽວ

ໃຊ້ໄດ້ສະເພາະແຖວດຽວ

/*……………*/ ໃຊ້ກັບຫຼາຍແຖວ

ຕົວຢ່າງ:

```
<!DOCTYPE html>
<html>
<body>
```

```
<?php
// ນີ້ແມ່ນຄຳອະທິບາຍແຖວດຽວ
# ນີ້ກໍ່ແມ່ນຄຳອະທິບາຍແຖວດຽວເຊັ່ນກັນ
/*
ນີ້ແມ່ນຄຳອະທິບາຍທີ່ເປັນບ່ອກ
ຫຼື ຫຼາຍແຖວ
*/

// ເຮົາຍັງສາມາດໃສ່ຄຳອະທິບາຍ ໃນແຕ່ລະແຖວຂອງລະຫັດໂປຣແກຣມອີກດ້ວຍ
$x = 5 /* + 15 */ + 5;
echo $x;
?>

</body>
</html>
```

1.3 PHP Case Sensitivity

ໃນພາສາ PHP ຄີເວີດທັງໝົດ (if, else, while, echo ແລະ ອື່ນໆ), ຄາສ, ຟັງຊັນມາດຕະຖານ, ຟັງຊັນທີ່ຜູ້ສ້າງກຳນົດຂຶ້ນມາ ແມ່ນເປັນແບບ Not case sensitive (ສາມາດຂຽນເປັນຕົວອັກສອນນ້ອຍຫຼືໃຫຍ່ກໍ່ໄດ້).

ຕົວຢ່າງດ້ານລຸ່ມ ຄຳສັ່ງ echo ທັງສາມແມ່ນມີຜົນເທົ່າກັນ

```
<html>
    <head><title>ເລີ່ມຕົ້ນຂຽນ PHP</title>
    </head>
    <body>
        ສະແດງຜົນດ້ວຍພາສາ HTML<br> <!--ສ່ວນສະແດງຜົນດ້ວຍພາສາ HTML-->
        <?php
            echo "ສະແດງຜົນດ້ວຍພາສາ PHP"; //ສະແດງຂໍ້ຄວາມອອກ
            EchO "<br>"; // not case sensitive
            ECHO "ຍິນດີຕ້ອນຮັບສູ່ພາສາ PHP";
        ?>
    </body>
</html>
```

ໝາຍເຫດ: ສະເພາະຕົວປ່ຽນເປັນ case sensitive ປະກາດແບບໃດຕ້ອງໃຊ້ແບບນັ້ນ ເຊັ່ນ: ຕົວປ່ຽນ $a ແລະ ຕົວປ່ຽນ $A ມີຄວາມໝາຍຕ່າງກັນ ຈະຖືເປັນສອງຕົວປ່ຽນ

```
<html>
   <head><title>ເລີ່ມຕົ້ນຂຽນ PHP</title>
   </head>
   <body>
      ສະແດງຜົນດ້ວຍພາສາ HTML<br> <!--ສ່ວນສະແດງຜົນດ້ວຍພາສາ HTML-->
      <?php
         $color = "ສີແດງ";
         echo "ເຮືອນຂອງຂ້ອຍສີ: $color <br>";
         Echo "ລົດຂອງຂ້ອຍສີ: $Color"; //ຈະບໍ່ສະແດງຄ່າສີອອກເນື່ອງຈາກຕົວປ່ຽນເປັນ Case sensitive

      ?>
   </body>
</html>
```

2. PHP 5 Variables

2.1 ການປະກາດຕົວປ່ຽນໃນພາສາ PHP

ໃນພາສາ PHP ຕົວປຽນຈະຂຶ້ນຕົ້ນດ້ວຍເຄື່ອງໝາຍ $ ຕົວຢ່າງ: $name, $surname, $age ແລະອື່ນໆ.

```
<!DOCTYPE html>
<html>
<body>

<?php
$txt = "Hello world!";
$x = 5;
$y = 10.5;

echo $txt;
echo "<br>";
echo $x;
echo "<br>";
echo $y;
?>
```

```
</body>
</html>
```

2.2 ຫຼັກການໃນການຕັ້ງຊື່ຂອງຕົວປ່ຽນໃນພາສາ php

- ຕົວປ່ຽນຕ້ອງຂຶ້ນຕົ້ນດ້ວຍເຄື່ອງໝາຍ $ ແລ້ວຕາມດ້ວຍຊື່ຂອງຕົວປ່ຽນ
- ຊື່ຂອງຕົວປ່ຽນຕ້ອງຂຶ້ນຕົ້ນດ້ວຍຕົວອັກສອນ ຫຼື ສັນຍາລັກອັນເດີສກໍ (_)
- ຊື່ຂອງຕົວປ່ຽນບໍ່ສາມາດຂຶ້ນຕົ້ນດ້ວຍຕົວເລກ
- ຕົວປ່ຽນສາມາດໃຊ້ມີໄດ້ຕົວອັກສອນ (A-Z, a-z, 0-9 ແລະ _)
- ຕົວປ່ຽນເປັນແບບ case sensitive ເຊັ່ນ: $a ແລະ $A ຖືເປັນສອງຕົວປ່ຽນ

2.3 ການກຳນົດຄ່າໃຫ້ກັບຕົວປ່ຽນ

ໃນພາສາ php ບໍ່ຈຳເປັນຕ້ອງບອກຊະນິດວ່າຕົວປ່ຽນຮັບຄ່າເປັນຕົວອັກສອນ ຫຼື ເປັນຕົວເລກ ເຊິ່ງຕ່າງຈາກພາສາອື່ນເຊັ່ນ: c, c++, java ແລະ ອື່ນໆ.

```
<?php
   $name="Souksavanh";
   $age=20;
   echo "Your name is $name <br>";
   echo "Your age is $age";
?>
```

2.3 ຂອບເຂດຂອງຕົວປ່ຽນຂອງ php

ເຮົາສາມາດປະກາດຕົວໄວ້ໃນສ່ວນໃດຂອງ script ຫຼື tag ກໍ່ໄດ້, ຂອບເຂດຂອງຕົວປ່ຽນຈະບອກເຖິງທີ່ໃດຂອງ script ສາມາດໃຊ້ໄດ້ ໃນພາສາ php ມີຂອບເຂດຂອງຕົວປ່ຽນ 3 ຊະນິດ ຄື:

- local
- global
- static

❖ ຕົວປ່ຽນແບບ local ປະກາດໄວ້ໃນຂອບເຂດຂອງຟັງຊັນໃດໜຶ່ງ ສາມາດອື່ນໃຊ້ໄດ້ພາຍໃນຂອບເຂດຂອງຟັງຊັນເທົ່ານັ້ນ

❖ ຕົວປ່ຽນແບບ global ປະກາດໄວ້ນອກຂອບເຂດຂອງຟັງຊັນ ແລະ ສາມາດອື່ນໃຊ້ໄດ້ທຸກຟັງຊັນ

```
<?php
$x=5; // ຕົວປ່ຽນແບບ global
//ສ້າງ ຟັງຊັນຊື່ myTest
function myTest() {
  $y=10; //ຕົວປ່ຽນແບບ local ເນື່ອງຈາກປະກາດໄວ້ໃນຂອບເຂດຂອງຟັງຊັນຊື່ myTest
  echo "<p>ທົດສອບຄ່າຂອງຕົວປ່ຽນຢູ່ໃນຟັງຊັນ:</p>";
  echo "ຕົວປ່ຽນ x ແມ່ນ: $x"; //$xບໍ່ສາມາດອື່ນໃຊ້ຕົວປ່ຽນ $x=5 ໄດ້ເນື່ອງຈາກຢູ່ຕ່າງຂອບເຂດກັນແຕ່ຖ້າຕ້ອງການ
  echo "<br>";              //ໃຊ້ $x=5 ຕ້ອງບອກວ່າ $x ເອົາຄ່າມາຈາກພາຍນອກຄືປ່ຽນ $x ເປັນ global $x
  echo "ຕົວປ່ຽນ y ແມ່ນ: $y";
}  //ສຸດຟັງຊັນ myTest

myTest(); //ເອີ້ນໃຊ້ຟັງຊັນ myTest

echo "<p>ທົດສອບຕົວປ່ຽນອອກຈາກຟັງຊັນ:</p>";
echo "ຕົວປ່ຽນ x ແມ່ນ: $x";
echo "<br>";
echo "ຕົວປ່ຽນ y ແມ່ນ: $y"; //$y ບໍ່ສາມາດໃຊ້ໄດ້ເນື່ອງຈາກຢູ່ນອກຂອບເຂດ
?>
```

ໝາຍເຫດ: ຖ້າຕ້ອງການນຳຕົວປ່ຽນແບບ global ມາສະແດງໃນ local ໃຫ້ປະຕິບັດຢ່າງຕົວຢ່າງຂ້າງລຸ່ມ

```
<?php
$x=5; // ຕົວປ່ຽນແບບ global
//ສ້າງ ຟັງຊັນຊື່ myTest
function myTest() {
  global $x; //ເອົາຕົວປ່ຽນພາຍນອກມາໃຊ້ພາຍໃນ
  $y=10; //ຕົວປ່ຽນແບບ local ເນື່ອງຈາກປະກາດໄວ້ໃນຂອບເຂດຂອງຟັງຊັນຊື່ myTest
  echo "<p>ທົດສອບຄ່າຂອງຕົວປ່ຽນຢູ່ໃນຟັງຊັນ:</p>";
  echo "ຕົວປ່ຽນ x ແມ່ນ: $x"; //$xບໍ່ສາມາດອື່ນໃຊ້ຕົວປ່ຽນ $x=5 ໄດ້ເນື່ອງຈາກຢູ່ຕ່າງຂອບເຂດກັນແຕ່ຖ້າຕ້ອງການ
  echo "<br>";              //ໃຊ້ $x=5 ຕ້ອງບອກວ່າ $x ເອົາຄ່າມາຈາກພາຍນອກຄືປ່ຽນ $x ເປັນ global $x
  echo "ຕົວປ່ຽນ y ແມ່ນ: $y";
}  //ສຸດຟັງຊັນ myTest
```

```
myTest(); //ເອີ້ນໃຊ້ຟັງຊັນ myTest

echo "<p>ທົດສອບຕົວປ່ຽນອອກຈາກຟັງຊັນ:</p>";
echo "ຕົວປ່ຽນ x ແມ່ນ: $x";
echo "<br>";
echo "ຕົວປ່ຽນ y ແມ່ນ: $y"; //$y ບໍ່ສາມາດໃຊ້ໄດ້ເນື່ອງຈາກຢູ່ນອກຂອບເຂດ
?>
```

- ຕົວປ່ຽນແບບ static ເວລາກັບຂຶ້ນໄປເອີ້ນຟັງຊັນອີກຄັ້ງໜຶ່ງ ຈະຂ້າມ ການຕັ້ງຄ່າເບື້ອງຕົ້ນແລະຈະຍັງຮັກສາຄ່າເກົ່າໄວ້

```
<?php

function myTest() {
  static $x=0;
  echo $x;
  $x++;
}

myTest();
echo "<br>";
myTest();
echo "<br>";
myTest();

?>
```

3. PHP 5 echo and print Statements

ໃນພາສາ PHP ມີ 2 ວິທີພື້ນຖານທີ່ໃຊ້ໃນການສະແດງຜົນຂໍ້ມູນຄື: echo ແລະ print. ມີຄວາມແຕກຕ່າງບາງຢ່າງລະຫວ່າງ echo ແລະ print:

- print ສາມາດສະແດງໄດ້ພຽງຂໍ້ຄວາມດຽວ, ແລະ ຄ່າທີ່ສົ່ງຄືນແມ່ນ 1

  ເມື່ອປຽບທຽບລະຫວ່າງ echo ແລະ print ເຫັນໄດ້ວ່າ echo ມີຄວາມໄວກວ່າເລັກນ້ອຍ ແລະ ບໍ່ມີການສົ່ງຄ່າໃດກັບຄືນ.

```
<?php
echo "<h2>PHP is fun!</h2>";
echo "Hello world!<br>";
echo "I'm about to learn PHP!<br>";
echo "This", " string", " was", " made", " with multiple parameters.";
?>
```

```
<?php
print "<h2>PHP is fun!</h2>";
print "Hello world!<br>";
print "I'm about to learn PHP!";
?>
```

ສະແດງໃຫ້ເຫັນລະຫວ່າງ echo ແລະ print

```
<?php
$a=5;
$b=10;
echo "ຜົນໄດ້ຮັບຈາກ a+b = $a+$b = ", ($a+$b);
print "<br>";
print "ຜົນໄດ້ຮັບຈາກ a+b = $a+$b = ". ($a+$b);
?>
```

4. PHP 5 String Functions

string ເປັນລຳດັບຂອງຕົວອັກສອນຕົວຢ່າງເຊັ່ນ: “Hello world!” ໃນຫົວຂໍ້ນີ້ຈະໄດ້ນຳສະເໜີບາງຕົວຢ່າງຂອງຟັງຊັນທີ່ໃຊ້ງານກັບສຕິງເຊັ່ນ: strlen(), strpos() .... ສ່ວນລາຍລະອຽດຂອງຟັງຊັນທັງໝົດເຂົ້າໄປທີ່ http://www.w3schools.com/php/php_ref_string.asp

- strlen() ຟັງຊັນຈະສົ່ງຄ່າຄວາມຍາວຂອງຂໍ້ຄວາມ

```
<?php
echo strlen("Hello world!");
?>
```

- strops() ຟັງຊັນທີ່ໃຊ້ໃນການຄົ້ນຫາຕຳແໜ່ງຂອງຕົວອັກສອນໃນຂໍ້ຄວາມ, ຕຳແໜ່ງຂອງຂໍ້ຄວາມຈະເລີ້ມຈາກຕຳແໜ່ງທີ່ 0

```
<?php
echo strpos("Hello world!","world");
?>
```

5.PHP 5 Constants

ຄ່າຄົງຄ່າ (constants): ຄືກັນກັບຕົວປ່ຽນ ແຕ່ມີຄວາມແຕກຕ່າງຢູ່ວ່າເມື່ອກຳນົດຂຶ້ນແລ້ວບໍ່ສາມາດປ່ຽນແປງ ຫຼື ກຳນົດຄ່າໃໝ່ໄດ້. ການກຳນົດຄ່າຄົງຄ່າໂດຍຕ້ອງໃຊ້ define() ຟັງຊັນ ເຊິ່ງປະກອບດ້ວຍຄ່າພາລາມິດເຕີ 3 ຄ່າຄື: 1 ຊື່ຂອງຄ່າຄົງຄ່າ, 2 ຄ່າຂອງຄ່າຄົງຄ່າ ແລະ ສຸດທ້າຍແມ່ນຊື່ຄ່າຄົງຄ່າຄວນຈະເປັນ case-incentive. ຄ່າອັດຕະໂນມັດ (Default) ແມ່ນ false.

```
<?php
define("GREETING", "Welcome to W3Schools.com!");
echo GREETING;
?>
```

```
<?php
define("GREETING", "Welcome to W3Schools.com!", true);
echo greeting;
?>
```

6. PHP 5 Operators

6.1 ຕົວດຳເນີນການທາງຄະນິດສາດ (PHP Arithmetic Operators)

ຕົວດຳເນີນການ	ຊື່	ຕົວຢ່າງ	ຜົນໄດ້ຮັບ
+	ບວກ	$x + $y	ຜົນບວກຂອງ $x ແລະ $y
-	ລົບ	$x - $y	ຄ່າແຕກຕ່າງຂອງ $x ແລະ $y
*	ຄູນ	$x * $y	ຜົນຄູນຂອງ $x ແລະ $y
/	ຫານ	$x / $y	ຜົນຫານຂອງ $x ແລະ $y
%	ຫານເອົາຄ່າເສດ	$x % $y	ຄ່າເສດຂອງ $x ຫານໃຫ້ $y
**	ຂຶ້ນກຳລັງ	$x ** $y	ຜົນໄດ້ຮັບຂອງ $x ຂຶ້ນກຳລັງ $y (ແນະນຳໃນ PHP 5.6)

6.2 ຕົວດຳເນີການພິເສດ (PHP Assignment Operators)

ຕົວດຳເນີນການ	ຕົວຢ່າງ	ຄືກັນກັບ	ຄວາມໝາຍ
=	$x = $y	$x = $y	ນຳຄ່າຕົວປ່ຽນ $y ມາເກັບໄວ້ໃນຕົວປ່ຽນ $x
+=	$x += $y	$x = $x + $y	ບວກສະສົມ
-=	$x -= $y	$x = $x - $y	ລົບສະສົມ
*=	$x *= $y	$x = $x * $y	ຄູນສະສົມ

/=	$x /= $y	$x = $x / $y	ຫານສະສົມ
%=	$x %= $y	$x = $x % $y	ຫານເອົາຄ່າເສດສະສົມ

6.4 ຕົວດຳເນີການຂອງຂໍ້ຄວາມ (PHP String Operators)

ຕົວດຳເນີນການ	ຊື່	ຕົວຢ່າງ	ຜົນໄດ້ຮັບ
.	ຕໍ່ຂໍ້ຄວາມ	$txt1 = "Hello" $txt2 = $txt1 . " world!"	ປະຈຸບັນ $txt2 ເກັບຄ່າ "Hello world!"
.=	ຕໍ່ຂໍ້ຄວາມສະສົມ	$txt1 = "Hello" $txt1 .= " world!"	ປະຈຸບັນ $txt1 ເກັບຄ່າ "Hello world!"

6.5 ຕົວດຳເນີການເພີ່ມຫຼືຫຼຸດຄ່າ (PHP Increment/Decrement Operators)

ຕົວດຳເນີນການ	ຊື່	ຄວາມໝາຍ
++$x	ເພີ່ມຄ່າກ່ອນ	ເພີ່ມຄ່າ $x ຂຶ້ນ 1 ຄ່າ, ຫຼັງຈາກນັ້ນ ສົ່ງຄ່າ $x ກັບຄືນ
$x++	ເພີ່ມຄ່າຕາມຫຼັງ	ສົ່ງຄ່າ $x, ຫຼັງຈາກນັ້ນ ເພີ່ມຄ່າ $x ຂຶ້ນ 1 ຄ່າ
--$x	ຫຼຸດຄ່າກ່ອນ	ຫຼຸດຄ່າ $x ລົງ 1 ຄ່າ, ຫຼັງຈາກນັ້ນ ສົ່ງຄ່າ $x ກັບຄືນ
$x--	ຫຼຸດຄ່າຕາມຫຼັງ	ສົ່ງຄ່າ $x, ຫຼັງຈາກນັ້ນ ຫຼຸດຄ່າ $x ລົງ 1 ຄ່າ

6.5 ຕົວດຳເນີການປຽບທຽບ (PHP Comparison Operators)

ຕົວດຳເນີນການ	ຊື່	ຕົວຢ່າງ	ຜົນໄດ້ຮັບ
==	ເທົ່າກັນ	$x == $y	ເປັນຈິງ ເມື່ອ $x ເທົ່າກັບ $y
===	ຄືກັນ	$x === $y	ເປັນຈິງ ເມື່ອ $x ເທົ່າກັບ $y, ແລະ ຕ້ອງແມ່ນຊະນິດຂໍ້ມູນດຽວກັນ
!=	ບໍ່ເທົ່າກັນ	$x != $y	ເປັນຈິງ ເມື່ອ $x ບໍ່ເທົ່າກັບ $y
<>	ບໍ່ເທົ່າກັນ	$x <> $y	ເປັນຈິງ ເມື່ອ $x ບໍ່ເທົ່າກັບ $y

!==	ບໍ່ຄືກັນ	$x !== $y	ເປັນຈິງ ເມື່ອ $x ບໍ່ເທົ່າກັບ $y, ຊະນິດຂໍ້ມູນບໍ່ຄືກັນ
>	ໃຫຍ່ກວ່າ	$x > $y	ເປັນຈິງເມື່ອ $x ມີຄ່າໃຫຍ່ກວ່າ $y
<	ນ້ອຍກວ່າ	$x < $y	ເປັນຈິງເມື່ອ $x ມີຄ່ານ້ອຍກວ່າ $y
>=	ໃຫຍ່ກວ່າເທົ່າກັບ	$x >= $y	ເປັນຈິງເມື່ອ $x ມີຄ່າໃຫຍ່ກວ່າຫຼືເທົ່າກັບ $y
<=	ນ້ອຍກວ່າເທົ່າກັບ	$x <= $y	ເປັນຈິງເມື່ອ $x ມີຄ່ານ້ອຍກວ່າຫຼືເທົ່າກັບ $y

6.6 ຕົວດຳເນີການຕັກກະສາດ (PHP Logical Operators)

ຕົວດຳເນີນການ	ຊື່	ຕົວຢ່າງ	ຜົນໄດ້ຮັບ
and	ແລະ	$x and $y	ເປັນຈິງເມື່ອ $x ແລະ $y ເປັນຈິງ
or	ຫຼື	$x or $y	ເປັນຈິງເມື່ອ $x ຫຼື $y ເປັນຈິງ
xor	Xor	$x xor $y	ເປັນຈິງເມື່ອ $x ຫຼື $y ເປັນຈິງ ແລະ ບໍ່ເທົ່າກັນ
&&	ແລະ	$x && $y	ເປັນຈິງເມື່ອ $x ແລະ $y ເປັນຈິງ
||	ຫຼື	$x || $y	ເປັນຈິງເມື່ອ $x ຫຼື $y ເປັນຈິງ
!	ປະຕິເສດ	!$x	ເປັນຈິງເມື່ອ $x ບໍ່ເປັນຈິງ