

### 3.1.1. PHP

PHP-ийн код нь `<? гэдэг түлхүүр үгээр эхэлж ?>` түлхүүр үгээр дуусна. Жишээ нь:

```
<? | <?php | <script language="php">
?> | ?> | </script>
```

Хамгийн хялбар PHP код бол `print`, `echo` командууд юм. Энэ нь өгөгдсөн текстийг дэлгэцэнд хэвлэж гаргах командууд юм. Жишээ нь:

```
<?
print("Hello world!"); | <?
?> | echo("Hello world!");
?>

<?
print("<font face=\"Arial\" color=\"#FF0000\">Hello world!</font>");
?>
```

PHP-гийн хувьд бүх командыг нэг мөрөнд бичих, команд болгоныг өөр мөрүүдэд бичих нь адилхан. Жишээ нь:

```
<?
echo "This is a test"; | <? echo "This is a test" ?>
?>
```

PHP-гийн талаарх мэдээллийг авах хүсвэл `phpinfo()` түлхүүр үгийг ашиглана. Жишээ нь:

```
<? phpinfo(); ?>

<?
// Show only the general information
phpinfo(INFO_GENERAL);
?>
```

Мөн browser-ийг дахин өөр хаяг уруу ачаалахыг хүсвэл:

```
<?
// Redirect browser
header("Location: http://www.num.edu.mn/");
?>
```

Файлыг нээх

```
<?
$f=fopen("welcome.txt","r");
?>
```

энд:

```
r      read буюу зөвхөн уншина
r+     read/write буюу уншиж бичнэ
w      файлыг дарж бичнэ, хэрэв тийм файл байхгүй бол шинээр үүсгэнэ
w+     файлыг дарж бичнэ, уншина, хэрэв тийм файл байхгүй бол шинээр үүсгэнэ
a      файлын төгсгөлийг заана
x      файлын эхлэлийг заана
```

Өмнөх файлыг хаахын тулд `fclose($f)`; түлхүүр үгийг хэрэглэнэ.

Харин файлаас тодорхой тэмдэгтийг уншихыг хүсвэл:

```
<?
if (!$f=fopen("welcome.txt","r"))
exit("Unable to open file.");
while (!feof($f))
{
$x=fgetc($f);
echo $x;
}
fclose($f);
?>
```

HTML-ийн форм доторх утгыг олж авахын тулд `$_POST` түлхүүр үгийг ашиглана. Жишээ нь:

HTML код:

```
<form action="action.php" method="post">
  Your name: <input type="text" name="name" />
  Your age: <input type="text" name="age" />
  <input type="submit" />
</form>
```

PHP код:

```
Hi <?php echo $_POST['name']; ?>.
You are <?php echo $_POST['age']; ?> years old.
```

### 3.1.2. Тайлбар оруулах

```
<?
// тайлбар
PHP code
?>
```

```
<?
PHP code // тайлбар
?>
```

```
<?
/* ингэж тайлбарыг олон
мөрөнд бичиж болдог,
иймээс үүнийг блок
тайлбар гэж ч нэрлэдэг */
PHP code
?>
```

### 3.1.3. Архметикийн үйлдлүүд

үйлдэл	жишээ	үр дүн	тайлбар	үйлдэл	жишээ	тайлбар
+	\$x=2 \$x+2	4	Нэмэх	=	\$x=\$y	х-д у-гийн утгыг олгох
-	\$x=2 \$x-2	0	Хасах	+=	\$x+=\$y	x=x+y
*	\$x=2 \$x*2	4	Үржүүлэх	--	\$x-=\$y	x=x-y
/	\$x=2 \$x/2	1	Хуваах	*=	\$x*=\$y	x=x*y
%	5%2 10%2	1 0	Үлдэгдэл	/=	\$x/=\$y	x=x/y
++	\$x=2 ++\$x	3	Нэмэгдүүлэх	==	\$x==\$y	х ба у тэнцүү
++	\$x=2 \$x++	3*	Нэмэгдүүлэх	!=	\$x!=\$y	х ба у тэнцүү бус
--	\$x=2 --\$x	1	Хорогдуулах	>	\$x>\$y	х нь у-ээс их
--	\$x=2 \$x--	1*	Хорогдуулах	<	\$x<\$y	х нь у-ээс бага
				=>	\$x>=\$y	х нь у-ээс тэнцүү буюу их
				=<	\$x<=\$y	х нь у-ээс бага буюу тэнцүү
				&&	\$x&&\$y	логик and үйлдэл
					\$x  \$y	логик or үйлдэл
				^	\$x^\$y	логик xor үйлдэл
				!	!\$x	логик not үйлдэл
				>>	\$x>>\$y	right shift үйлдэл
				<<	\$x<<\$y	left shift үйлдэл

Жишээ нь:

```
<?
function double($i)
{
    return $i*2;
}
$b = $a = 5;          /* $a=5 $b=5 */
$c = $a++;           /* $a=6 $c=5 */
$e = $d = ++$b;     /* $b=6 $d=6 $e=6 */
$f = double($d++); /* $d=7 $f=2*6=12 */
$g = double(++$e); /* $e=7 $g=2*7=14 */
$h = $g += 10;     /* $g=24 $h=24 */
$a = ($b = 4) + 5; /* $a=9 $b=4
$a = 3;           // $a=3
$a += 5;         // $a=8
$b = "Hello ";   // $b="Hello"
$b .= "There!";  // $b="Hello There!" энэ нь $b = $b . "There!";-тэй адилхан
echo "hallo" ^ "hello"; // Outputs the ascii values #0 #4 #0 #0 #0
                        // 'a' ^ 'e' = #4
?>
```

Энд ++\$a ба \$a++ нь а хувьсагчийн утгыг 1-ээр нэмэгдүүлнэ. Ялгаа нь \$a++ нь үйлдлийг гүйцэтгэсний дараа а хувьсагч нь 1-ээр нэмэгдэх бол ++\$a нь үйлдлийг гүйцэтгэхээс өмнө а хувьсагчийн утгыг 1-ээр нэмэгдүүлнэ.

Жишээ 1.

```
<?
echo "<h3>Дараа нь нэмэгдүүлнэ</h3>";
$a = 5;
echo "үр дүн нь 5: " . $a++ . "<br />\n";
echo "үр дүн нь 6: " . $a . "<br />\n";

echo "<h3>Эхлээд нэмэгдүүлнэ</h3>";
$a = 5;
echo "үр дүн нь 6: " . ++$a . "<br />\n";
echo "үр дүн нь 6: " . $a . "<br />\n";
?>
```

```

echo "<h3>Дараа нь хорогдуулна</h3>";
$a = 5;
echo "үр дүн нь 5: " . $a-- . "<br />\n";
echo "үр дүн нь 4: " . $a . "<br />\n";

echo "<h3>Эхлээд хорогдуулна</h3>";
$a = 5;
echo "үр дүн нь 4: " . --$a . "<br />\n";
echo "үр дүн нь 4: " . $a . "<br />\n";
?>

```

## Жишээ 2.

```

<?
$i = 'W';
for($n=0; $n<6; $n++)
    echo ++$i . "\n";
?>
үр дүн:
X
Y
Z
AA
AB
AC

```

### 3.2.1. Хувьсагч ба тогтмол

Хувьсагчийг зарлахдаа `$` гэсэн түлхүүр үгийг ашиглаж түүнд анхны утгыг олгоно.

Жишээ нь:

```

$my_year = 6;
$my_name = 'Золоо';

```

Хувьсагч нь `boolean`, `integer`, `float`, `string` гэсэн 4 төрлийнх байж болно.

```

<?
$bool = TRUE; // boolean
$str = "foo"; // string
$int = 12; // integer

echo gettype($bool); // TRUE
echo gettype($str); // foo

// хэрэв &int хувьсагч нь integer бол 4-өөр ихэснэ
if (is_int($int)) {
    $int += 4;
}

// хэрэв $bool хувьсагч string бол түүнийг хэвлэ
if (is_string($bool)) {
    echo "String: $bool";
}
?>

```

Тогтмолыг зарлахдаа `define()` гэсэн түлхүүр үгийг зарлана. Энд нэг анхаарах зүйл бол PHP-д том жижиг үсгүүдийг ялгаатай гэж ойлгоно. Жишээ нь:

```

<?
define("CONSTANT", "Hello world.");
echo CONSTANT; // outputs "Hello world."
echo Constant; // outputs "Constant"
?>
// программын явцад утга нь өөрчлөгдөж байвал хувьсагч, өөрчлөгдөхгүй бол тогтмол

```

### 3.2.2. String хувьсагч

String хувьсагч нь тоо болон тэмдэгтээс тогтох бөгөөд `string`-ийг давхар хашилтанд тэмдэглэнэ.

Жишээ нь:

```

<?
$welcome = "Hello, welcome to my web";
print($welcome);
?>

```

String-ийг дараах байдлаар дан хашилтанд бичиж өгч болно.

```

<?
echo 'энэ бол энгийн string';

echo 'string-ийн утыг
олон мөрөнд хуваан
бичиж болно';
?>

```

Мөн string-үүдийг . тэмдэгийг ашиглан хооронд нэгтгэж болно.

```

<?php
$a = "Hello ";
$b = $a . "World!"; // $b="Hello World!"

$a = "Hello ";
$a .= "World!"; // $a="Hello World!"
?>

```

Энд нэг зүйлийг сануулахад . нь 2 string-ийг шууд нэгтгэдэг тул хоосон зайг тусад нь авдаг.

<pre> &lt;? &amp;txt1="энэ" &amp;txt2="бол" &amp;txt3=&amp;txt1.&amp;txt2 echo"&amp;txt3"; &lt;/script&gt; </pre>	<pre> &lt;? &amp;txt1="энэ" &amp;txt2="бол" &amp;txt3=&amp;txt1." ".&amp;txt2 echo"&amp;txt3"; &lt;/script&gt; </pre>
Үр дүн: энэбол	Үр дүн: энэ бол

Тусгай тэмдэгтүүдийг string дотор оруулахдаа \ тэмдэгийг ашигладаг. Жишээ нь:

```

// Outputs: Arnold once said: "I'll be back"
echo 'Arnold once said: "I\'ll be back"';

// Outputs: You deleted C:\*.?*
echo 'You deleted C:\\*..*?';

// Outputs: You deleted C:\*.?*
echo 'You deleted C:\*..*?';

// Outputs: This will not expand: \n a newline
echo 'This will not expand: \n a newline';

// Outputs: Variables do not $expand $either
echo 'Variables do not $expand $either';

```

String-ийн дурын тэмдэгтэд хандан, түүнийг өөр тэмдэгтээр сольж болно. Жишээ нь:

```

<?
// Get the first character of a string
$str = 'This is a test.';
$first = $str{0};
// Get the third character of a string
$third = $str{2};
// Get the last character of a string.
$str = 'This is still a test.';
$last = $str{strlen($str)-1};
// Modify the last character of a string
$str = 'Look at the sea';
$str{strlen($str)-1} = 'e';
?>

```

String-ийг дараах байдлаар тоон хувьсагчид шилжүүлж болно. Жишээ нь:

```

<?
$foo = 1 + "10.5"; // $foo is float (11.5)
$foo = 1 + "-1.3e3"; // $foo is float (-1299)
$foo = 1 + "bob-1.3e3"; // $foo is integer (1)
$foo = 1 + "bob3"; // $foo is integer (1)
$foo = 1 + "10 Small Pigs"; // $foo is integer (11)
$foo = 4 + "10.2 Little Piggies"; // $foo is float (14.2)
$foo = "10.0 pigs " + 1; // $foo is float (11)
$foo = "10.0 pigs " + 1.0; // $foo is float (11)
?>

```

### 3.2.3. Логик хувьсагч

Логик хувьсагчийг зарлахдаа new Boolean(утга) гэсэн түлхүүр үгийг ашиглана. Энэ функц тухайн логик утга үнэн бол true, худал бол false гэсэн 2 утга авна.

Энэ функц нь булийн логик FALSE, хоосон "" болон тэг "0" string, null элементээс тогтсон массив, тэг хувьсагчаас тогтсон объект, бүхэл тоо 0, бутархай тоо 0,0 үед FALSE утга, бусад бүх тохиолдолд TRUE утга авна. Жишээ нь:

```
<?php
echo gettype((bool) ""); // (false)
echo gettype((bool) 1); // (true)
echo gettype((bool) -2); // (true)
echo gettype((bool) "foo"); // (true)
echo gettype((bool) 2.3e5); // (true)
echo gettype((bool) array(12)); // (true)
echo gettype((bool) array()); // (false)
?>
```

### 3.2.4. Тоон хувьсагч

Бүхэл тоон хувьсагч нь дараах утгуудыг авна.

```
<?
$a = 1234; # decimal number
$a = -123; # negative number
$a = 0123; # octal number (83 decimal)
$a = 0x1A; # hexadecimal number (26 decimal)
?>
```

Харин бутархайн тоон хувьсагч нь дараах утгуудыг авна.

```
<?
$a = 1.234;
$b = 1.2e3;
$c = 7E-10;
?>
```

### 3.2.5. Массив хувьсагч

Массив хувьсагчийг зарлахдаа аргау түлхүүр үгийг ашиглана. Жишээ нь:

```
<?
$arr = array("foo" => "bar", 12 => true);

echo $arr["foo"]; // bar
echo $arr[12]; // 1
?>
```

Массив хувьсагчийн элементүүдийг зарлах үндсэн 2 арга байдаг.

<pre>\$names[0] = 'John'; \$names[1] = 'Paul'; \$names[2] = 'Steven'; \$names[3] = 'George'; \$names[4] = 'David';</pre>	<pre>\$number = 5; \$x = 0; while (\$x &lt; \$number) {     \$namenumber = \$x + 1;     echo "Name \$namenumber is     \$names[\$x]&lt;br&gt;";     ++\$x }</pre>
--	---

Массивын дурын элементэд дараах байдлаар хандаж болно.

```
echo "The third name is $names[2]";
```

### 3.3.1. Нөхцөл шалгах функц (if, if else)

<pre>if (нөхцөл){ нөхцөл үнэн үед биелэх PHP код }</pre>	<pre>if (&amp;username==&amp;webmaster){ echo"нууц үгээ оруулна уу?" }</pre>
<pre>if (нөхцөл){ нөхцөл үнэн үед биелэх PHP код }else{ нөхцөл худал үед биелэх PHP код }</pre>	<pre>if (&amp;username==&amp;webmaster){ echo"нууц үгээ оруулна уу?" }else{ echo"та бүртгүүлээгүй байна!" }</pre>
<pre>if (нөхцөл){ нөхцөл үнэн үед биелэх PHP код }elseif{ нөхцөл худал үед биелэх PHP код }else{ дээрх 2 нөхцөл худал үед биелэх PHP код }</pre>	

#### Жишээ 1.

```
<?
if ($name == "" || $email == "" || $password == "") {
echo "бүх талбарыг бөглөнө vv?";
}
?>
```

#### Жишээ 2.

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

#### Жишээ 3.

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

#### Жишээ 4.

```
<?
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";
}
?>
```

#### Жишээ 5.

```
<?
if ($a == 5):
    echo "a equals 5";
    echo "...";
elseif ($a == 6):
    echo "a equals 6";
    echo "!!!";
else:
    echo "a is neither 5 nor 6";
endif;
?>
```

### 3.3.2. Нөхцөл шалгах функц (switch)

1-ээс олон нөхцөлийг шалгах тохиолдолд switch функцийг ашиглах нь тохиромжтой. Жишээ нь:

```
<?php
switch (&day)
{
case 6:
    echo("хагас сайн")
    break
case 0:
    echo("бүтэн сайн")
    break
default:
    echo("ажлын өдөр")
}
?>
```

if, else, elseif-ийг switch-тэй харьцуулж үзье.

<pre>&lt;? if (\$i == 0) {     echo "i equals 0"; } elseif (\$i == 1) {     echo "i equals 1"; } elseif (\$i == 2) {     echo "i equals 2"; } ?&gt;</pre>	<pre>&lt;? switch (\$i) { case 0:     echo "i equals 0";     break; case 1:     echo "i equals 1";     break; } ?&gt;</pre>
---	---

```
?>
```

```
case 2:
    echo "i equals 2";
    break;
}
?>
```

#### Жишээ 1.

```
<?
switch ($i) {
case 0:
case 1:
case 2:
    echo "i is less than 3 but not negative";
    break;
case 3:
    echo "i is 3";
}
?>
```

#### Жишээ 2.

```
<?
switch ($i) {
case 0:
    echo "i=0";
    break;
case 1:
    echo "i=1";
    break;
case 2:
    echo "i=2";
    break;
default:
    echo "i нь 0, 1, 2-той тэнцэхгүй";
}
?>
```

### 3.3.3. Давталтын функц (for)

For давталтанд хувьсагчид анхны утга олгох бөгөөд давталт аль хүртэл үргэлжлэхийг тогтоосон нөхцөл, давталт болгоны дараа хувьсагчийн утга хэрхэн өөрчлөгдөхийг тогтоосон нөхцөл (давталт болгоны дараа хувьсагч нэгээр нэмэгдэх, хорогдох г.м) зэргийг тогтоож өгнө.

```
for (эхний утга, нөхцөл1, нөхцөл2){
PHP code
}
for (i=1; i<=6; i++){
echo("<h"+i+">Header"+i)
echo("</h"+i+">")
}
```

#### Жишээ 1.

```
<?
for ($i = 1; $i <= 10; $i++) {
    echo $i;
}
?>
```

#### Жишээ 2.

```
<?
for ($i = 1; ; $i++) {
    if ($i > 10) {
        break;
    }
    echo $i;
}
?>
```

#### Жишээ 3.

```
<?
$i = 1;
for (; ; ) {
    if ($i > 10) {
        break;
    }
    echo $i;
    $i++;
}
```

```
}  
?>
```

#### Жишээ 4.

```
<?  
for ($i = 1; $i <= 10; echo $i, $i++);  
?>
```

### 3.3.4. Давталтын функцууд (while)

While давталт нь for давталтын хялбаршуулсан хувилбар буюу ганцхан давталт аль хүртэл үргэлжлэхийг тогтоосон нөхцөлийг л агуулна.

```
while (нөхцөл){  
PHP code  
}  
$times = 5;  
$x = 0;  
while ($x < $times) {  
echo "Hello World";  
++$x;  
}
```

#### Жишээ 1.

```
<?  
$i = 1;  
while ($i <= 10) {  
echo $i++;  
}  
?>
```

#### Жишээ 2.

```
<?  
$i = 1;  
while ($i <= 10):  
echo $i;  
$i++;  
endwhile;  
?>
```

### 3.3.5. Давталтын функцууд (do, while)

While давталт нь нөхцөлийг давталт эхлэхээс өмнө шалгадаг бол do давталт нь нөхцөлийг давталтыг дууссаны дараа шалгана.

```
do{  
PHP code  
}  
while(нөхцөл)  
i=0  
do{  
echo("The number is"+i)  
echo("<br>")  
i++  
} while(i<5)
```

#### Жишээ 1.

```
<?  
do {  
if ($i < 5) {  
echo "i is not big enough";  
break;  
}  
$i *= $factor;  
if ($i < $minimum_limit) {  
break;  
}  
echo "i is ok";  
/* process i */  
} while (0);  
?>
```

### 3.3.6. Break, continue

Давталтын үйлдлийг зогсооход break-ийг ашиглана. Жишээ нь:

#### Жишээ 1.

```
var i = 0  
while (i < 6)
```



```

{
if (i == 3)
break
i++
}

```

## Жишээ 2.

```

<?
$arr = array('one', 'two', 'three', 'four', 'stop', 'five');
while (list ($val) = each ($arr)) {
    if ($val == 'stop') {
        break; /* You could also write 'break 1;' here. */
    }
    echo "$val<br />\n";
}
$i = 0;
while (++$i) {
    switch ($i) {
        case 5:
            echo "At 5<br />\n";
            break 1; /* Exit only the switch. */
        case 10:
            echo "At 10; quitting<br />\n";
            break 2; /* Exit the switch and the while. */
        default:
            break;
    }
}
?>

```

Харин давталтыг дахин эхлүүлэхэд continue-г ашиглана.

## Жишээ 1.

```

i = 0
n = 0
while (i < 5) {
i++
if (i == 3)
continue
n += i
}

```

i нь 3-тай тэнцэх үед давталтыг дахин эхлүүлэх учраас n нь 3, 7, 12 г.м утга авна.

## Жишээ 2.

```

<?
while (list ($key, $value) = each ($arr)) {
    if (!$key % 2) { // skip odd members
        continue;
    }
    do_something_odd ($value);
}

$i = 0;
while ($i++ < 5) {
    echo "Outer<br />\n";
    while (1) {
        echo "&nbsp;&nbsp;&nbsp;Middle<br />\n";
        while (1) {
            echo "&nbsp;&nbsp;&nbsp;Inner<br />\n";
            continue 3;
        }
        echo "This never gets output.<br />\n";
    }
    echo "Neither does this.<br />\n";
}
?>

```

## Жишээ 3.

```

for ($i = 0; $i < 5; ++$i) {
    if ($i == 2)
        continue
    print "$i\n";
}
ҮР дҮН:
0

```

```
1
3
4
```

### 3.3.7. Include, require

```
include.php
<?php echo "Hi!"; ?>

require.php
<?php echo "Bye!"; ?>

main.php
<?php
include("include.php");
echo "<br>How are you?<br>";
require("require.php");
?>

үр дүн:
Hi!
How are you?
Bye!
```

### 3.4.1. Функци

Хэд хэдэн үйлдлүүдийг нэгтгэн нэр өгөх замаар функцийг үүсгэнэ. Функцийг программын дурын хэсгээс дуудан ажиллуулж болно. Функцийг программд олон дахин ашиглагдах үйлдлүүдийг функц болгон ашигласнаар программын хэмжээг багасгах болон программын бүтцийг цэгцтэй болгох үүднээс ашигладаг. Функцийг программ зохиогч өөрөө үүсгэж болохоос гадна бэлэн функцуудыг шууд авч ашиглаж болно.

Функцийг шинээр үүсгэхдээ `function` нэр (хувьсагч) гэсэн түлхүүр үгээр функцийн нэр, ашиглах хувьсагчуудыг зарлах ба `{ }` хаалтанд энэ функцийн гүйцэтгэх үйлдлийг тогтоосон кодыг бичиж өгнө. Жишээ нь:

```
<?
function foo ($arg_1, $arg_2, /* ..., */ $arg_n)
{
    echo "Example function.\n";
    return $retval;
}
?>
```

Функцийг дуудан ажиллуулахдаа шууд функцийн нэрийг бичиж өгөх ба функц биелэгдсэний дараа удирдлага функцийг дуудсан газарт буцаж ирнэ.

Жишээ 1.

```
<?
function add_some_extra(&$string)
{
    $string .= 'and something extra.';
}
$str = 'This is a string, ';
add_some_extra($str);
echo $str;    // outputs 'This is a string, and something extra.'
?>
```

Жишээ 2.

```
<?
function makecoffee ($type = "cappuccino")
{
    return "Making a cup of $type.\n";
}
echo makecoffee ();
echo makecoffee ("espresso");
?>
```

```
үр дүн:
Making a cup of cappuccino.
Making a cup of espresso.
```

Жишээ 3.

```
<?
function makeyogurt ($flavour, $type = "acidophilus")
```

```

    {
        return "Making a bowl of $type $flavour.\n";
    }
echo makeyogurt ("raspberry"); // works as expected
?>
ҮР ДҮН:
Making a bowl of acidophilus raspberry.

```

#### Жишээ 4.

```

<?
function square ($num)
{
    return $num * $num;
}
echo square (4); // outputs '16'.
?>

```

#### Жишээ 5.

```

<?
function foo (&$var)
{
    $var++;
}
$a=5;
foo ($a); // $a is 6 here
?>

```

#### Жишээ 6.

```

<?
function foo() {
    echo "In foo()<br />\n";
}
function bar($arg = '')
{
    echo "In bar(); argument was '$arg'.<br />\n";
}
function echoit($string)
{
    echo $string;
}
$func = 'foo';
$func(); // This calls foo()
$func = 'bar';
$func('test'); // This calls bar()
$func = 'echoit';
$func('test'); // This calls echoit()
?>

```

#### Жишээ 7.

```

<?
class Foo
{
    function Variable()
    {
        $name = 'Bar';
        $this->$name(); // This calls the Bar() method
    }
    function Bar()
    {
        echo "This is Bar";
    }
}
$foo = new Foo();
$funcname = "Variable";
$foo->$funcname(); // This calls $foo->Variable()
?>

```

### 3.4.2. Class

Class нь тухайн хувьсагчтай ажиллах хувьсагч, функцуудын цуглуулга. Жишээ нь:

```

<?
class Cart {
    var $items; // $items-ийг дэлгүүрээс юм худалдан авдаг сагс гэж үзье.
}

```

```

// $artnr төрлийн $num гэсэн барааг сонгон сагсанд хийе.
function add_item($artnr, $num) {
    $this->items[$artnr] += $num;
}
// $artnr төрлийн $num барааг сагснаас буцааж гаргая.
function remove_item($artnr, $num) {
    if ($this->items[$artnr] > $num) {
        $this->items[$artnr] -= $num;
        return true;
    } else {
        return false;
    }
}
}
?>

```

### 3.4.3. Mail функц

#### Жишээ 1.

```

<?
mail("someone@num.edu.mn", "Захианы нэр", "Энд захиагаа бичиж өгнө");
?>

```

#### Жишээ 2.

```

<?
mail("someone@num.edu.mn", "the subject", $message,
    "From: webmaster@{_SERVER['SERVER_NAME']}\r\n" .
    "Reply-To: webmaster@{_SERVER['SERVER_NAME']}\r\n" .
    "X-Mailer: PHP/" . phpversion());
?>

```

#### Жишээ 3.

```

<?
mail("someone@num.edu.mn", "the subject", $message,
    "From: webmaster@{_SERVER['SERVER_NAME']}",
    "-fwebmaster@{_SERVER['SERVER_NAME']}");
?>

```

#### Жишээ 4.

```

<?
/* хүлээн авагч */
$to = "someone@num.edu.mn" . ", " ;
$to .= "someone1@num.edu.mn";

/* захианы нэр */
$subject = "8 сард төрөгсөдийн мэдээлэл";

/* захиа */
$message = '
<html>
<head>
<title>Mail</title>
</head>
<body>
<p>8 сард төрөгсөдийн мэдээлэл!</p>
<table>
<tr>
    <th>Person</th><th>Day</th><th>Month</th><th>Year</th>
</tr>
<tr>
    <td>Joe</td><td>3rd</td><td>August</td><td>1970</td>
</tr>
<tr>
    <td>Sally</td><td>17th</td><td>August</td><td>1973</td>
</tr>
</table>
</body>
</html>
';

/* Content-type header */
$headers = "MIME-Version: 1.0\r\n";
$headers .= "Content-type: text/html; charset=iso-8859-1\r\n";

```

```

/* майлийг явуулах */
mail($to, $subject, $message, $headers);
?>

```

#### Жишээ 5.

```

$to = "php@gowansnet.com";
$subject = "PHP Is Great";
$body = "PHP is one of the best scripting languages around";
$headers = "From: webmaster@gowansnet.com\n";
mail($to,$subject,$body,$headers);
echo "Mail sent to $to";

```

#### Жишээ 6.

```

mail("php@gowansnet.com","PHP Is Great","PHP is one of the best scripting
languages around","From: webmaster@gowansnet.com\n");

if(mail($to,$subject,$body,$headers)) {
echo "An e-mail was sent to $to with the subject: $subject";
} else {
echo "There was a problem sending the mail. Check your code and make sure that the
e-mail address $to is valid";
}

```

#### Жишээ 7.

```

<form action="mail.php" method="post">
Your Name: <input type="text" name="name"><br>
E-mail: <input type="text" name = "email"><br><br>
Comments<br>
<textarea name="comments"></textarea><br><br>
<input type="submit" value="Submit">
</form>
<?
$name=$_POST['name'];
$email=$_POST['email'];
$comments=$_POST['comments'];
$to="php@gowansnet.com";
$message="$name just filled in your comments form. They said:\n$comments\n\nTheir
e-mail address was: $email";
if(mail($to,"Comments From Your Site",$message,"From: $email\n")) {
echo "Thanks for your comments.";
} else {
echo "There was a problem sending the mail. Please check that you filled in the
form correctly.";
}
?>

```

### 3.4.4. Date функц

a	"am" эсвэл "pm"
A	"AM" эсвэл "PM"
B	Swatch Internet time (000-999)
d	өдөр (01-31)
D	гаригийн эхний гурван тэмдэгт (Mon-Sun)
F	сарын бүтэн нэр (January-December)
g	цаг (1-12)
G	цаг (0-23)
h	цаг (01-12)
H	цаг (00-23)
i	минут (00-59)
j	өдөр (1-31)
l	гариг (Monday-Sunday)
L	хэрэв 2-р сар нь 29 хоногтой жил бол "1", бусад тохиолдолд"0"
m	сар (01-12)
M	сарын нэрний эхний гурван тэмдэгт (Jan-Dec)
n	сар (1-12)
O	Greenwich time (GMT)-ээс цагийн зөрүү
r	огноо (e.g. "Tue, 10 Apr 2005 18:34:07 +0300")
s	секунд (00-59)
t	сард оногдох хоногийн тоо (28-31)
T	цагийн бүс (e.g. "GMT")
U	(January 1 1970 00:00:00 GMT)-ээс хойш өнгөрсөн хугцааг секундээр тооцвол
w	гаригийг тоогоор илэрхийлэх (0-6, 0=Sunday)
W	гаригийг Даваа гаригаас эхлэн тоолох

```
Y      жил (e.g. 2003)
y      жил (e.g. 03)
z      тухайн жилийн хэд дэх өдөр (0-366)
```

### Жишээ 1.

```
<?
//гариг: Monday
echo date("l");

//огноо: Monday 15th of January 2003 05:51:38 AM
echo date("l dS of F Y h:i:s A");

//өдөр: Monday the 15th
echo date("l \\t\\h\\e jS");
?>
```

### Жишээ 2.

```
<?
var_dump(checkdate(12, 31, 2000));
var_dump(checkdate(2, 29, 2001));
?>
үр дүн:
bool(true)
bool(false)
```

### Жишээ 3.

```
<?
// prints: July 1, 2000 is on a Saturday
echo "July 1, 2000 is on a " . date("l", mktime(0, 0, 0, 7, 1, 2000));
?>
```

### Жишээ 4.

```
<?
// prints: Wednesday the 15th
echo date("l \\t\\h\\e jS");
?>
```

### Жишээ 5.

```
<?
// Assuming today is: March 10th, 2001, 5:16:18 pm
$today = date("F j, Y, g:i a");           // March 10, 2001, 5:16 pm
$today = date("m.d.y");                   // 03.10.01
$today = date("j, n, Y");                 // 10, 3, 2001
$today = date("Ymd");                     // 20010310
$today = date('h-i-s, j-m-y, it is w Day z '); // 05-16-17, 10-03-01, 1631 1618 6
Fripm01
$today = date('\\i\\t \\i\\s \\t\\h\\e jS \\d\\a\\y. '); // It is the 10th day.
$today = date("D M j G:i:s T Y");         // Sat Mar 10 15:16:08 MST 2001
$today = date('H:m:s \\m \\i\\s\\ \\m\\o\\n\\t\\h'); // 17:03:17 m is month
$today = date("H:i:s");                   // 17:16:17
?>
```

### Жишээ 6.

```
<?
$today = getdate();
print_r($today);
?>
үр дүн:
Array
(
    [seconds] => 40           // секунд
    [minutes] => 58          // минут
    [hours]   => 21          // цаг
    [mday]    => 17          // өдөр
    [wday]    => 2           // гариг тоогоор
    [mon]     => 6           // сар тоогоор
    [year]    => 2003        // жил
    [yday]    => 167         // жилийн хэд дэх өдөр
    [weekday] => Tuesday    // гариг үсгээр
    [month]   => June        // сар үсгээр
    [0]       => 1055901520 // гринвичийн цагаас хойш өнгөрсөн хугацаа секунд
)
```

### 3.4.5. Calendar функц

#### Жишээ 1.

```
<?
$num = cal_days_in_month(CAL_GREGORIAN, 8, 2003); // 31
echo "There was $num days in August 2003";
?>
```

#### Жишээ 2.

```
<?
$today = unixtojd(mktime(0, 0, 0, 8, 16, 2003));
print_r(cal_from_jd($today, CAL_GREGORIAN));
?>
ҮР дүн:
Array
(
    [date] => 8/16/2003
    [month] => 8
    [day] => 16
    [year] => 2003
    [dow] => 6
    [abbrevdayname] => Sat
    [dayname] => Saturday
    [abbrevmonth] => Aug
    [monthname] => August
)
```

#### Жишээ 3.

```
<?
echo date("M-d-Y", easter_date(1999)); // Apr-04-1999
echo date("M-d-Y", easter_date(2000)); // Apr-23-2000
echo date("M-d-Y", easter_date(2001)); // Apr-15-2001
?>
```

#### Жишээ 4.

```
<?
echo easter_days(1999); // 14, i.e. April 4
echo easter_days(1492); // 32, i.e. April 22
echo easter_days(1913); // 2, i.e. March 23
?>
```

### 3.4.6. Variable функц

#### хувьсагчийг хоосон эсэхийг шалгах

```
<?
$var = 0;
// Evaluates to true because $var is empty
if (empty($var)) {
    echo '$var is either 0, empty, or not set at all';
}
// Evaluates as true because $var is set
if (isset($var)) {
    echo '$var is set even though it is empty';
}
?>
```

#### хувьсагчийн float утгыг авах

```
<?
$var = '122.34343The';
$float_value_of_var = floatval($var);
echo $float_value_of_var; // prints 122.34343
?>
```

#### хувьсагчийг логик хувьсагч эсэхийг шалгах

```
<?
$a = false;
$b = 0;
// Since $a is a boolean, this is true
if (is_bool($a)) {
    echo "Yes, this is a boolean";
}
// Since $b is not a boolean, this is not true
if (is_bool($b)) {
    echo "Yes, this is a boolean";
}
```

```
}  
?>
```

#### хувьсагчийг массив хувьсагч эсэхийг шалгах

```
<?  
$yes = array('this', 'is', 'an array');  
echo is_array($yes) ? 'Array' : 'not an Array';  
echo "\n";  
$no = 'this is a string';  
echo is_array($no) ? 'Array' : 'not an Array';  
?>
```

#### хувьсагчийн утгыг ойлгомжтой байдлаар хэвлэх

```
<pre>  
<?  
    $a = array ('a' => 'apple', 'b' => 'banana', 'c' => array ('x', 'y', 'z'));  
    print_r ($a);  
?>  
</pre>  
үр дүн:  
<pre>  
Array  
(  
    [a] => apple  
    [b] => banana  
    [c] => Array  
        (  
            [0] => x  
            [1] => y  
            [2] => z  
        )  
    )  
</pre>
```

#### хувьсагчийн төрлийг тогтоож өгөх

```
<?  
$foo = "5bar"; // string  
$bar = true; // boolean  
settype($foo, "integer"); // $foo is now 5 (integer)  
settype($bar, "string"); // $bar is now "1" (string)  
?>
```

### 3.4.7. Image функц

#### зургийн төрлийг тогтоох

```
<?  
if (exif_imagetype("image.gif") != IMAGETYPE_GIF) {  
    echo "The picture is not a gif";  
}  
?>
```

1	IMAGETYPE_GIF
2	IMAGETYPE_JPEG
3	IMAGETYPE_PNG
4	IMAGETYPE_SWF
5	IMAGETYPE_PSD
6	IMAGETYPE_BMP
7	IMAGETYPE_TIFF_II
8	IMAGETYPE_TIFF_MM
9	IMAGETYPE_JPC
10	IMAGETYPE_JP2
11	IMAGETYPE_JPX
12	IMAGETYPE_JB2
13	IMAGETYPE_SWC
14	IMAGETYPE_IFF
15	IMAGETYPE_WBMP



**зургийн хэмжээг тогтоох**

```
<?
list($width, $height, $type, $attr) = getimagesize("img/flag.jpg");
echo "<img src=\"img/flag.jpg\" $attr alt=\"getimagesize() example\" />";
?>
```

**цагираг зурах**

```
<?
// create a 200*200 image
$img = imagecreate(200, 200);
// allocate some colors
$white = imagecolorallocate($img, 255, 255, 255);
// draw a white circle
imagearc($img, 100, 100, 150, 150, 0, 360, $white);
// output image in the browser
header("Content-type: image/png");
imagepng($img);
// free memory
imagedestroy($img);
?>
```

**хэвтээ шулуун зурах**

```
<?
$img = imagecreate(100, 100);
$string = 'PHP';
$bg = imagecolorallocate($img, 255, 255, 255);
$black = imagecolorallocate($img, 0, 0, 0);
// prints a black "P" in the top left corner
imagechar($img, 1, 0, 0, $string, $black);
header('Content-type: image/png');
imagepng($img);
?>
```

**босоо шулуун зурах**

```
<?
$img = imagecreate(100, 100);
$string = 'Note that the first letter is a N';
$bg = imagecolorallocate($img, 255, 255, 255);
$black = imagecolorallocate($img, 0, 0, 0);
// prints a black "Z" on a white background
imagecharup($img, 3, 10, 10, $string, $black);
header('Content-type: image/png');
imagepng($img);
?>
```

**зургийн өнгийг тогтоох**

```
<?
// sets background to red
$background = imagecolorallocate($img, 255, 0, 0);
// sets some colors
$white = imagecolorallocate($img, 255, 255, 255);
$black = imagecolorallocate($img, 0, 0, 0);
// hexadecimal way
$white = imagecolorallocate($img, 0xFF, 0xFF, 0xFF);
$black = imagecolorallocate($img, 0x00, 0x00, 0x00);
?>
```

**зургийн өнгөний кодыг олж авах**

```
<?
// open an image
$img = imagecreatefrompng('nexen.png');
// get a color
$start_x = 40;
$start_y = 50;
$color_index = imagecolorat($img, $start_x, $start_y);
// make it human readable
$color_tran = imagecolorsforindex($img, $color_index);
// what is it ?
echo '<pre>';
print_r($color_tran);
echo '</pre>';
?>
Үр дүн:
Array
```

```
(
    [red] => 226
    [green] => 222
    [blue] => 252
    [alpha] => 0
)
```

### зураг үүсгэх

```
<?
header("Content-type: image/png");
$im = @imagecreate(50, 100)
    or die("Cannot Initialize new GD image stream");
$background_color = imagecolorallocate($im, 255, 255, 255);
$text_color = imagecolorallocate($im, 233, 14, 91);
imagestring($im, 1, 5, 5, "A Simple Text String", $text_color);
imagepng($im);
imagedestroy($im);
?>
```

### элпис зурах

```
<?
// create a blank image
$image = imagecreate(400, 300);
// fill the background color
$bg = imagecolorallocate($image, 0, 0, 0);
// choose a color for the ellipse
$col_ellipse = imagecolorallocate($image, 255, 255, 255);
// draw the ellipse
imageellipse($image, 200, 150, 300, 200, $col_ellipse);
// output the picture
header("Content-type: image/png");
imagepng($image);
?>
```

### 3 хэмжээст зураг зурах

```
<?
// this example is provided by poxy at klam dot is
// create image
$image = imagecreate(100, 100);
// allocate some colors
$white = imagecolorallocate($image, 0xFF, 0xFF, 0xFF);
$gray = imagecolorallocate($image, 0xC0, 0xC0, 0xC0);
$darkgray = imagecolorallocate($image, 0x90, 0x90, 0x90);
$navy = imagecolorallocate($image, 0x00, 0x00, 0x80);
$darknavy = imagecolorallocate($image, 0x00, 0x00, 0x50);
$red = imagecolorallocate($image, 0xFF, 0x00, 0x00);
$darkred = imagecolorallocate($image, 0x90, 0x00, 0x00);
// make the 3D effect
for ($i = 60; $i > 50; $i--) {
    imagefilledarc($image, 50, $i, 100, 50, 0, 45, $darknavy, IMG_ARC_PIE);
    imagefilledarc($image, 50, $i, 100, 50, 45, 75, $darkgray, IMG_ARC_PIE);
    imagefilledarc($image, 50, $i, 100, 50, 75, 360, $darkred, IMG_ARC_PIE);
}
imagefilledarc($image, 50, 50, 100, 50, 0, 45, $navy, IMG_ARC_PIE);
imagefilledarc($image, 50, 50, 100, 50, 45, 75, $gray, IMG_ARC_PIE);
imagefilledarc($image, 50, 50, 100, 50, 75, 360, $red, IMG_ARC_PIE);
// flush image
header('Content-type: image/png');
imagepng($image);
imagedestroy($image);
?>
```

### зургийг будах

```
<?
// create a blank image
$image = imagecreate(400, 300);
// fill the background color
$bg = imagecolorallocate($image, 0, 0, 0);
// choose a color for the ellipse
$col_ellipse = imagecolorallocate($image, 255, 255, 255);
// draw the white ellipse
imagefilledellipse($image, 200, 150, 300, 200, $col_ellipse);
// output the picture
header("Content-type: image/png");
```

```
imagepng($image);
?>
```

### зургийг browser руу гаргах

```
<?
if (function_exists("imagegif")) {
    header("Content-type: image/gif");
    imagegif($im);
} elseif (function_exists("imagejpeg")) {
    header("Content-type: image/jpeg");
    imagejpeg($im, "", 0.5);
} elseif (function_exists("imagepng")) {
    header("Content-type: image/png");
    imagepng($im);
} elseif (function_exists("imagewbmp")) {
    header("Content-type: image/vnd.wap.wbmp");
    imagewbmp($im);
} else {
    die("No image support in this PHP server");
}
?>
<?
if (imagetypes() & IMG_GIF) {
    header ("Content-type: image/gif");
    imagegif ($im);
} elseif (imagetypes() & IMG_JPG) {
    /* ... etc. */
}
?>
```

### шулуун зурах

```
<?
function imagelinethick($image, $x1, $y1, $x2, $y2, $color, $thick = 1)
{
    /* this way it works well only for orthogonal lines
    imagesetthickness($image, $thick);
    return imageline($image, $x1, $y1, $x2, $y2, $color);
    */
    if ($thick == 1) {
        return imageline($image, $x1, $y1, $x2, $y2, $color);
    }
    $t = $thick / 2 - 0.5;
    if ($x1 == $x2 || $y1 == $y2) {
        return imagefilledrectangle($image, round(min($x1, $x2) - $t),
round(min($y1, $y2) - $t), round(max($x1, $x2) + $t), round(max($y1, $y2) + $t),
$color);
    }
    $k = ($y2 - $y1) / ($x2 - $x1); //y = kx + q
    $a = $t / sqrt(1 + pow($k, 2));
    $points = array(
        round($x1 - (1+$k)*$a), round($y1 + (1-$k)*$a),
        round($x1 - (1-$k)*$a), round($y1 - (1+$k)*$a),
        round($x2 + (1+$k)*$a), round($y2 - (1-$k)*$a),
        round($x2 + (1-$k)*$a), round($y2 + (1+$k)*$a),
    );
    imagefilledpolygon($image, $points, 4, $color);
    return imagepolygon($image, $points, 4, $color);
}
?>
```

### png зургийг browser уруу гаргах

```
<?
$im = imagecreatefrompng("test.png");
imagepng($im);
?>
```

### полигон зурах

```
<?
// create a blank image
$image = imagecreate(400, 300);
// fill the background color
$bg = imagecolorallocate($image, 0, 0, 0);
// choose a color for the polygon
$col_poly = imagecolorallocate($image, 255, 255, 255);
```

```
// draw the polygon
imagepolygon($image,
    array (
        0, 0,
        100, 200,
        300, 200
    ),
    3,
    $col_poly);
// output the picture
header("Content-type: image/png");
imagepng($image);
?>
```

#### үсгэн шугам зурах

```
<?
// create a 100*30 image
$im = imagecreate(100, 30);
// white background and blue text
$bg = imagecolorallocate($im, 255, 255, 255);
$textcolor = imagecolorallocate($im, 0, 0, 255);
// write the string at the top left
imagestring($im, 5, 0, 0, "Hello world!", $textcolor);
// output the image
header("Content-type: image/jpg");
imagejpeg($im);
?>
```

#### зургийн өргөний хэмжээг олж авах

```
<?
// create a 300*200 image
$img = imagecreate(300, 200);
echo imagesx($img); // 300
?>
```

#### зургийн уртын хэмжээг олж авах

```
<?
// create a 300*200 image
$img = imagecreate(300, 200);
echo imagesy($img); // 200
?>
```

#### зурган дээр текст бичих

```
<?
header("Content-type: image/jpeg");
$im = imagecreate(400, 30);
$white = imagecolorallocate($im, 255, 255, 255);
$black = imagecolorallocate($im, 0, 0, 0);
// Replace path by your own font path
imaggottext($im, 20, 0, 10, 20, $black, "/path/arial.ttf",
    "Testing... Omega: &#937;");
imagejpeg($im);
imagedestroy($im);
?>
```

### 3.4.8. Mathematical функц

#### абсолют утгыг олох

```
<?
$abs = abs(-4.2); // $abs = 4.2; (double/float)
$abs2 = abs(5); // $abs2 = 5; (integer)
$abs3 = abs(-5); // $abs3 = 5; (integer)
?>
```

#### тоог хөврүүлэх

```
<?
$hexadecimal = 'A37334';
echo base_convert($hexadecimal, 16, 2);
?>
үр дүн:
101000110111001100110100
```

#### 2-тын тооллоос 10-тын тоолол уруу шилжүүлэх

```
<?
echo bindec('110011') . "\n";
```

```

echo bindec('000110011') . "\n";

echo bindec('111');
?>
бутархай тооноос их бүхэл хэсгийг авах
<?
echo ceil(4.3); // 5
echo ceil(9.999); // 10
?>
cos-ийг олох
<?
echo cos(M_PI); // -1
?>
10-тын тооллоос 2-тын тооллын систем рүү хөврүүлэх
<?
echo decbin(12) . "\n";
echo decbin(26);
?>
10-тын тооллын системээс 16-тын тооллын систем рүү хөврүүлэх
<?
echo dechex(10) . "\n";
echo dechex(47);
?>
10-тын тооллын системээс 8-тын тооллын систем рүү хөврүүлэх
<?
echo decoct(15) . "\n";
echo decoct(264);
?>
градусыг радиан уруу хөврүүлэх
<?
echo deg2rad(45); // 0.785398163397
var_dump(deg2rad(45) === M_PI_4); // bool(true)
?>
e-гийн зэргийн олох
<?
echo exp(12) . "\n";
echo exp(5.7);
?>
ҮР дүн:
1.6275E+005
298.87
бутархай тооноос бага бүхэл хэсгийг нь сонгох
<?
echo floor(4.3); // 4
echo floor(9.999); // 9
?>
үлдэгдлийг олох
<?
$x = 5.7;
$y = 1.3;
$r = fmod($x, $y);
// $r equals 0.5, because 4 * 1.3 + 0.5 = 5.7
?>
16-тын тооллын системээс 10-тын тооллын систем рүү хөврүүлэх
<?
var_dump(hexdec("See"));
var_dump(hexdec("ee"));
// both print "int(238)"
var_dump(hexdec("that")); // print "int(10)"
var_dump(hexdec("a0")); // print "int(160)"
?>
тоонуудаас ихийг нь олох
<?
echo max(1, 3, 5, 6, 7); // 7
echo max(array(2, 4, 5)); // 5
echo max(0, 'hello'); // 0
echo max('hello', 0); // hello
echo max(-1, 'hello'); // hello
// With multiple arrays, max compares from left to right

```

```
// so in our example: 2 == 2, but 4 < 5
$val = max(array(2, 4, 8), array(2, 5, 7)); // array(2, 5, 7)
// If both an array and non-array are given, the array
// is always returned as it's seen as the largest
$val = max('string', array(2, 5, 7), 42); // array(2, 5, 7)
?>
```

#### тоонуудаас багыг нь олох

```
<?
echo min(2, 3, 1, 6, 7); // 1
echo min(array(2, 4, 5)); // 2
echo min(0, 'hello'); // 0
echo min('hello', 0); // hello
echo min('hello', -1); // -1
// With multiple arrays, min compares from left to right
// so in our example: 2 == 2, but 4 < 5
$val = min(array(2, 4, 8), array(2, 5, 1)); // array(2, 4, 8)
// If both an array and non-array are given, the array
// is never returned as it's considered the largest
$val = min('string', array(2, 5, 7), 42); // string
?>
```

#### санамсаргүй тоог гаргах

```
<?
echo mt_rand() . "\n";
echo mt_rand() . "\n";
echo mt_rand(5, 15);
?>
үр дүн:
1604716014
1478613278
6
```

#### 8-тын тооллын системээс 10-тын тооллын систем рүү хөврүүлэх

```
<?
echo octdec('77') . "\n";
echo octdec(decoct(45));
?>
```

#### pi тоог олох

```
<?
echo pi(); // 3.1415926535898
echo M_PI; // 3.1415926535898
?>
```

#### радианыг градус уруу хөврүүлэх

```
<?
echo rad2deg(M_PI_4); // 45
?>
```

#### бүхэл санамсаргүй тоог олох

```
<?
echo rand() . "\n";
echo rand() . "\n";
echo rand(5, 15);
?>
үр дүн:
7771
22264
11
```

#### бутархай тооноос ойролцоо бүхэл тоог олох

```
<?
echo round(3.4); // 3
echo round(3.5); // 4
echo round(3.6); // 4
echo round(3.6, 0); // 4
echo round(1.95583, 2); // 1.96
echo round(1241757, -3); // 1242000
echo round(5.045, 2); // 5.05
echo round(5.055, 2); // 5.06
?>
```

#### синусыг олох

```
<?
// Precision depends on your precision directive
echo sin(deg2rad(60)); // 0.866025403 ...
```

```

echo sin(60); // -0.304810621 ...
?>
квадрат язгуур гаргах
<?
// Precision depends on your precision directive
echo sqrt(9); // 3
echo sqrt(10); // 3.16227766 ...
?>
тангенсийг олох
<?
echo tan(M_PI_2); // 1
?>

```

### 3.4.9. String функц

**string-д орсон тэмдэгтүүдийг тоолох**

```

<?
$data = "Two Ts and one F.";
$result = count_chars($data, 0);
for ($i=0; $i < count($result); $i++) {
    if ($result[$i] != 0)
        echo "There were $result[$i] instance(s) of \"", chr($i), "\" in the
string.\n";
}
?>

```

ҮР ДҮН:

```

There were 4 instance(s) of " " in the string.
There were 1 instance(s) of "." in the string.
There were 1 instance(s) of "F" in the string.
There were 2 instance(s) of "T" in the string.
There were 1 instance(s) of "a" in the string.
There were 1 instance(s) of "d" in the string.
There were 1 instance(s) of "e" in the string.
There were 2 instance(s) of "n" in the string.
There were 2 instance(s) of "o" in the string.
There were 1 instance(s) of "s" in the string.
There were 1 instance(s) of "w" in the string.

```

**string-ийг хэвлэх**

```

print
<?
print("Hello World");
print "print() also works without parentheses.";
print "This spans
multiple lines. The newlines will be
output as well";
print "This spans\nmultiple lines. The newlines will be\noutput as well.";
print "escaping characters is done \"Like this\".";
// You can use variables inside of a print statement
$foo = "foobar";
$bar = "barbaz";
print "foo is $foo"; // foo is foobar
// You can also use arrays
$bar = array("value" => "foo");
print "this is {$bar['value']} !"; // this is foo !
// Using single quotes will print the variable name, not the value
print 'foo is $foo'; // foo is $foo
// If you are not using any other characters, you can just print variables
print $foo; // foobar
print <<<END
This uses the "here document" syntax to output
multiple lines with $variable interpolation. Note
that the here document terminator must appear on a
line with just a semicolon no extra whitespace!
END;
?>

```

**echo**

```

<?
echo "Hello World";
echo "This spans
multiple lines. The newlines will be

```

```

output as well";
echo "This spans\nmultiple lines. The newlines will be\noutput as well.";
echo "Escaping characters is done \"Like this\".";
// You can use variables inside of an echo statement
$foo = "foobar";
$bar = "barbaz";
echo "foo is $foo"; // foo is foobar
// You can also use arrays
$bar = array("value" => "foo");
echo "this is {$bar['value']} !"; // this is foo !
// Using single quotes will print the variable name, not the value
echo 'foo is $foo'; // foo is $foo
// If you are not using any other characters, you can just echo variables
echo $foo; // foobar
echo $foo,$bar; // foobarbarbaz
// Some people prefer passing multiple parameters to echo over concatenation.
echo 'This ', 'string ', 'was ', 'made ', 'with multiple parameters.', chr(10);
echo 'This ' . 'string ' . 'was ' . 'made ' . 'with concatenation.' . "\n";
echo <<<END
This uses the "here document" syntax to output
multiple lines with $variable interpolation. Note
that the here document terminator must appear on a
line with just a semicolon. no extra whitespace!
END;
// Because echo is not a function, following code is invalid.
($some_var) ? echo 'true' : echo 'false';
// However, the following examples will work:
($some_var) ? print('true'): print('false'); // print is a function
echo $some_var ? 'true': 'false'; // changing the statement around
?>

```

#### string-ийг хуваах

```

<?
$pizza = "piece1 piece2 piece3 piece4 piece5 piece6";
$pieces = explode(" ", $pizza);
echo $pieces[0]; // piece1
echo $pieces[1]; // piece2
?>

<?
$data = "foo*:~:1023:1000::~/home/foo:/bin/sh";
list($user, $pass, $uid, $gid, $gecos, $home, $shell) = explode(":", $data);
echo $user; // foo
echo $pass; // *
?>

```

#### массивын элементүүдийг нэгтгэн string болгох

```

<?
$array = array('lastname', 'email', 'phone');
$comma_separated = implode(",", $array);
echo $comma_separated; // lastname,email,phone
?>

```

#### string дотор өөр тэмдэгт нэмэх

```

<?
$input = "Alien";
echo str_pad($input, 10); // produces "Alien "
echo str_pad($input, 10, "--", STR_PAD_LEFT); // produces "--Alien"
echo str_pad($input, 10, "_", STR_PAD_BOTH); // produces "_Alien_"
echo str_pad($input, 6, "___"); // produces "Alien_"
?>

```

#### string-ийг давтах

```

<?
echo str_repeat("-", 10);
?>
үр дүн:
-----

```

#### string доторх тэмдэгтийг өөр тэмдэгтээр солих

```

<?
// Provides: <body text='black'>
$bodytag = str_replace("%body%", "black", "<body text='%body%'>");
// Provides: Hll Wrld f PHP
$vowels = array("a", "e", "i", "o", "u", "A", "E", "I", "O", "U");

```



```

$onlyconsonants = str_replace($vowels, "", "Hello World of PHP");
// Provides: You should eat pizza, beer, and ice cream every day
$phrase = "You should eat fruits, vegetables, and fiber every day.";
$healthy = array("fruits", "vegetables", "fiber");
$yummy = array("pizza", "beer", "ice cream");
$newphrase = str_replace($healthy, $yummy, $phrase);
// Use of the count parameter is available as of PHP 5.0.0
$str = str_replace("ll", "", "good golly miss molly!", $count);
echo $count; // 2
?>

```

**string-ийг түүний тэмдэгтүүдийг дурын байрлалд байруулсан string-ээр солих**

```

<?
$str = 'abcdef';
$shuffled = str_shuffle($str);
// This will echo something like: bfdaec
echo $shuffled;
?>

```

**string-ийг түүнд орсон тэмдэгтүүдээр нь хуваах**

```

<?
$str = "Hello Friend";
$arr1 = str_split($str);
$arr2 = str_split($str, 3);
print_r($arr1);
print_r($arr2);
?>

```

үр дүн:

```

Array
(
    [0] => H
    [1] => e
    [2] => l
    [3] => l
    [4] => o
    [5] =>
    [6] => F
    [7] => r
    [8] => i
    [9] => e
    [10] => n
    [11] => d
)
Array
(
    [0] => Hel
    [1] => lo
    [2] => Fri
    [3] => end
)

```

**string дэх үгийг тоолох**

```

<?
$str = "Hello friend, you're
        looking good today!";
$a = str_word_count($str, 1);
$b = str_word_count($str, 2);
$c = str_word_count($str);
print_r($a);
print_r($b);
echo $c;
?>

```

үр дүн:

```

Array
(
    [0] => Hello
    [1] => friend
    [2] => you're
    [3] => looking
    [4] => good
    [5] => today
)

```

```

Array
(
    [0] => Hello
    [6] => friend
    [14] => you're
    [29] => looking
    [46] => good
    [51] => today
)

```

6

#### string-ийн тэмдэгтүүдийг харьцуулах

```

<?
$var1 = "Hello";
$var2 = "hello";
if (strcasecmp($var1, $var2) == 0) {
    echo '$var1 is equal to $var2 in a case-insensitive string comparison';
}
?>

```

#### string-ийн уртын хэмжээг тогтоох

```

<?
$str = 'abcdef';
echo strlen($str); // 6
$str = ' ab cd ';
echo strlen($str); // 7
?>

```

#### string-ээс тэмдэгт хайх

```

<?
$haystack = 'ababcd';
$needle = 'aB';
$pos = stripos($haystack, $needle);
if ($pos === false) {
    echo "Sorry, we did not find ($needle) in ($haystack)";
} else {
    echo "Congratulations!\n";
    echo "We found the last ($needle) in ($haystack) at position ($pos)";
}
?>

```

ҮР ДҮН:

Congratulations!

We found the last (aB) in (ababcd) at position (2)

#### тодорхой тэмдэгээс өмнөхийг тэмдэгтүүдийг хасах

```

<?
$email = 'user@example.com';
$domain = strstr($email, '@');
echo $domain; // prints @example.com
?>

```

#### үгнүүдийн байрыг солих

```

<?
$trans = array("hello" => "hi", "hi" => "hello");
echo strstr("hi all, I said hello", $trans);
?>

```

ҮР ДҮН:

hello all, I said hi

#### тухайн тэмдэгт хэдэн удаа орсныг тоолох

```

<?
echo substr_count("This is a test", "is"); // prints out 2
?>

```

#### тэмдэгт хасах

```

<?
$rest = substr("abcdef", 1); // returns "bcdef"
$rest = substr("abcdef", 1, 3); // returns "bcd"
$rest = substr("abcdef", 0, 4); // returns "abcd"
$rest = substr("abcdef", 0, 8); // returns "abcdef"
// Accessing via curly braces is another option
$string = 'abcdef';
echo $string{0}; // returns a
echo $string{3}; // returns d
?>

```

### ЭХНИЙ ТЭМДЭГТИЙГ ТОМРУУЛАХ

```
<?
$foo = 'hello world!';
$foo = ucfirst($foo);           // Hello world!
$bar = 'HELLO WORLD!';
$bar = ucfirst($bar);         // HELLO WORLD!
$bar = ucfirst(strtolower($bar)); // Hello world!
?>
```

### ҮГ БОЛГОНЫ ЭХНИЙ ҮСГИЙГ ТОМРУУЛАХ

```
<?
$foo = 'hello world!';
$foo = ucwords($foo);         // Hello World!
$bar = 'HELLO WORLD!';
$bar = ucwords($bar);        // HELLO WORLD!
$bar = ucwords(strtolower($bar)); // Hello World!
?>
```

### ТОДОРХОЙ УРТАЙГААР ТАСЛАХ

```
<?
$text = "The quick brown fox jumped over the lazy dog.";
$newtext = wordwrap($text, 20, "<br />");
echo "$newtext\n";
?>
Үр дүн:
The quick brown fox
jumped over the
lazy dog.

<?
$text = "A very long woooooooooooooord.";
$newtext = wordwrap($text, 8, "\n", 1);
echo "$newtext\n";
?>
Үр дүн:
A very
long
wooooooo
ooooord.
```

## 3.4.10. Regular Expression функц

	Pattern	Жишээ:
	/pattern/	</\w+>/
	[pattern]	(\d{3})-\d+ Sm
	{pattern}	/(?i)php[34]/
	<pattern>	{^\s+(\s+)?\$}
^	эхлэл	^n n-ээр эхэлсэн
\$	төгсгөл	\$n n-ээр төгссөн
.	бүх тэмдэгт	
[	class-ийн эхлэл	
]	class-ийн төгсгөл	
	нөхцөл шалгах	(n m) n ба m-ийн аль нэг нь орсон
(	subpattern-ийн эхлэл	
)	subpattern-ийн төгсгөл	(nm) n ба m 2-улаа орсон
?	0 эсвэл 1 агуулсан	n? 0 эсвэл 1 n орсон
*	0 ба түүнээс их	n* 0 ба түүнээс их n орсон
+	1 ба түүнээс их	n+ 1 ба түүнээс их n орсон
{	min/max тооны эхлэл	n{2} дараалан 2 n орсон
}	min/max тооны төгсгөл	n{1,} 1-ээс олон дараалсан n орсон
\a	alarm (hex 07)	
\cx	"control-x", x нь дурын тэмдэгт	
\e	escape (hex 1B)	
\f	formfeed (hex 0C)	
\n	шинэ мөр (hex 0A)	
\r	return (hex 0D)	
\t	tab (hex 09)	
\xhh	hex code нь hh байх тэмдэгт	
\ddd	octal code ddd	
\d	decimal digit	
\D	character (not a decimal digit)	

string-ээс тодорхой тэмдэгтүүдийг ялган авахдаа `preg_grep()` түлхүүр үгийг ашиглана.

### Жишээ 1.

```
<?
// return all array elements
// containing floating point numbers
$fl_array = preg_grep("/^(\d+)?\.\d+$/", $array);
?>
```

### Жишээ 2.

```
<?
preg_match_all("|<[^>]+>(.*?)</[^>]+>|U",
    "<b>example: </b><div align=left>this is a test</div>",
    $out, PREG_PATTERN_ORDER);
echo $out[0][0] . " , " . $out[0][1] . "\n";
echo $out[1][0] . " , " . $out[1][1] . "\n";
?>
Үр дүн:
<b>example: </b>, <div align=left>this is a test</div>
example: , this is a test
```

### Жишээ 3.

```
<?
preg_match_all("|<[^>]+>(.*?)</[^>]+>|U",
    "<b>example: </b><div align=\"left\">this is a test</div>",
    $out, PREG_SET_ORDER);
echo $out[0][0] . " , " . $out[0][1] . "\n";
echo $out[1][0] . " , " . $out[1][1] . "\n";
?>
Үр дүн:
<b>example: </b>, example:
<div align="left">this is a test</div>, this is a test
```

### Жишээ 4.

```
<?
preg_match_all("/\((? (\d{3})? \))? (? (1) [\-\s] ) \d{3}-\d{4}/x",
    "Call 555-1212 or 1-800-555-1212", $phones);
?>
```

### Жишээ 5.

```
<?
// The \2 is an example of backreferencing. This tells pcre that
// it must match the second set of parentheses in the regular expression
// itself, which would be the ([w]+) in this case. The extra backslash is
// required because the string is in double quotes.
$html = "<b>bold text</b><a href=howdy.html>click me</a>";
preg_match_all("/(<([\w]+)[^>]*>)(.*)(<\/\2>)/", $html, $matches);
for ($i=0; $i< count($matches[0]); $i++) {
    echo "matched: " . $matches[0][$i] . "\n";
    echo "part 1: " . $matches[1][$i] . "\n";
    echo "part 2: " . $matches[3][$i] . "\n";
    echo "part 3: " . $matches[4][$i] . "\n\n";
}
?>
Үр дүн:
matched: <b>bold text</b>
part 1: <b>
part 2: bold text
part 3: </b>
matched: <a href=howdy.html>click me</a>
part 1: <a href=howdy.html>
part 2: click me
part 3: </a>
```

### Жишээ 6.

```
<?
// The "i" after the pattern delimiter indicates a case-insensitive search
if (preg_match("/php/i", "PHP is the web scripting language of choice.")) {
    echo "A match was found.";
} else {
    echo "A match was not found.";
}
?>
```

### Жишээ 7.

```
<?
/* The \b in the pattern indicates a word boundary, so only the distinct
* word "web" is matched, and not a word partial like "webbing" or "cobweb" */
```

```

if (preg_match("/\bweb\b/i", "PHP is the web scripting language of choice.)) {
    echo "A match was found.";
} else {
    echo "A match was not found.";
}
if (preg_match("/\bweb\b/i", "PHP is the website scripting language of choice.))
{
    echo "A match was found.";
} else {
    echo "A match was not found.";
}
?>

```

#### Жишээ 8.

```

<?
// get host name from URL
preg_match("/^(http:\\/\\/)?([^\./]+)/i",
    "http://www.php.net/index.html", $matches);
$host = $matches[2];
// get last two segments of host name
preg_match("/[^\./]+\.[^\./]+$/", $host, $matches);
echo "domain name is: {$matches[0]}\n";
?>
ҮР дүн:
domain name is: php.net

```

#### Жишээ 9.

```

<?
$keywords = "$40 for a g3/400";
$keywords = preg_quote($keywords, "/");
echo $keywords; // returns \$40 for a g3\/400
?>

```

#### Жишээ 10.

```

<?
// In this example, preg_quote($word) is used to keep the
// asterisks from having special meaning to the regular
// expression.
$textbody = "This book is *very* difficult to find.";
$word = "*very*";
$textbody = preg_replace ("/" . preg_quote($word) . "/",
    "<i>" . $word . "</i>",
    $textbody);
?>

```

#### Жишээ 11.

```

<?
// this text was used in 2002
// we want to get this up to date for 2003
$text = "April fools day is 04/01/2002\n";
$text.= "Last christmas was 12/24/2001\n";
// the callback function
function next_year($matches)
{
    // as usual: $matches[0] is the complete match
    // $matches[1] the match for the first subpattern
    // enclosed in '(...)' and so on
    return $matches[1].($matches[2]+1);
}
echo preg_replace_callback(
    "|(\d{2}/\d{2}) (\d{4})|",
    "next_year",
    $text);
// result is:
// April fools day is 04/01/2003
// Last christmas was 12/24/2002
?>

```

#### Жишээ 12.

```

<?
/* a unix-style command line filter to convert uppercase
 * letters at the beginning of paragraphs to lowercase */
$fp = fopen("php://stdin", "r") or die("can't read stdin");
while (!feof($fp)) {
    $line = fgets($fp);
}

```

```

    $line = preg_replace_callback(
        '|<p>\s*\w|',
        create_function(
            // single quotes are essential here,
            // or alternative escape all $ as \$
            '$matches',
            'return strtolower($matches[0]);'
        ),
        $line
    );
    echo $line;
}
fclose($fp);
?>

```

string-ийн доторх үгийг өөр үгээр солихдоо `preg_replace()` түлхүүр үгийг ашиглана.

#### Жишээ 1.

```

<?
$string = "April 15, 2003";
$pattern = "/(\w+) (\d+), (\d+)/i";
$replacement = "\${1}1,\${3}";
echo preg_replace($pattern, $replacement, $string);
?>

```

үр дүн:  
Aprill,2003

#### Жишээ 2.

```

<?
$string = "The quick brown fox jumped over the lazy dog.";
$patterns[0] = "/quick/";
$patterns[1] = "/brown/";
$patterns[2] = "/fox/";
$replacements[2] = "bear";
$replacements[1] = "black";
$replacements[0] = "slow";
echo preg_replace($patterns, $replacements, $string);
?>

```

үр дүн:  
The bear black slow jumped over the lazy dog.

#### Жишээ 3.

```

<?
ksort($patterns);
ksort($replacements);
echo preg_replace($patterns, $replacements, $string);
?>

```

үр дүн:  
The slow black bear jumped over the lazy dog.

#### Жишээ 4.

```

<?
$patterns = array ("/(19|20) (\d{2})-(\d{1,2})-(\d{1,2})/",
    "^\s*(\w+)\s*="/);
$replace = array ("\3/\4/\1\2", "$\1 =");
echo preg_replace($patterns, $replace, "{startDate} = 1999-5-27");
?>

```

үр дүн:  
\$startDate = 5/27/1999

#### Жишээ 5.

```

<?
preg_replace("/(<\/?) (\w+) ([^>]*)/e",
    "'\1'.strtoupper('\2').'\3'",
    $html_body);
?>

```

#### Жишээ 6.

```

<?
// $document should contain an HTML document.
// This will remove HTML tags, javascript sections
// and white space. It will also convert some
// common HTML entities to their text equivalent.
$search = array ("<script[^>]*?>.*/script>'si", // Strip out javascript
    "<[\/\!]?*[^<>]*?>'si", // Strip out HTML tags
    "'([\r\n])[\s]+'", // Strip out white space
    "'&(quot|#34);'i", // Replace HTML entities

```

```

        "'&(amp|#38);'i",
        "'&(lt|#60);'i",
        "'&(gt|#62);'i",
        "'&(nbsp|#160);'i",
        "'&(iexcl|#161);'i",
        "'&(cent|#162);'i",
        "'&(pound|#163);'i",
        "'&(copy|#169);'i",
        "'&#(\d+);'e");
    // evaluate as php
$replace = array (
    "",
    "",
    "\\1",
    "\"",
    "&",
    "<",
    ">",
    " ",
    chr(161),
    chr(162),
    chr(163),
    chr(169),
    "chr(\\1)");
$text = preg_replace($search, $replace, $document);
?>

```

string-ийг үг үгээр нь салгахдаа `preg_split()` түлхүүр үгийг ашиглана.

Жишээ 1.

```

<?
// split the phrase by any number of commas or space characters,
// which include " ", \r, \t, \n and \f
$keywords = preg_split("/[\\s,]+/", "hypertext language, programming");
?>

```

Жишээ 2.

```

<?
$str = 'string';
$chars = preg_split('//', $str, -1, PREG_SPLIT_NO_EMPTY);
print_r($chars);
?>

```

Жишээ 3.

```

<?
$str = 'hypertext language programming';
$chars = preg_split('/ /', $str, -1, PREG_SPLIT_OFFSET_CAPTURE);
print_r($chars);
?>
үр дүн:
Array
(
    [0] => Array
        (
            [0] => hypertext
            [1] => 0
        )
    [1] => Array
        (
            [0] => language
            [1] => 10
        )
    [2] => Array
        (
            [0] => programming
            [1] => 19
        )
)

```

string дотроос үг болон тэмдэгтийг хайхад `ereg()` ба `eregi()`-ийг ашиглана. `eregi()` нь string дотроос тэмдэгтийг хайхдаа том жижиг үсгийг ялгаатай гэж үздэг бол `ereg()` нь string дотроос тэмдэгтийг хайхдаа том жижиг үсгийг ялгаагүй гэж үздэг.

Жишээ 1.

```

<?
// Returns true if "abc" is found anywhere in $string.
ereg("abc", $string);

```

```

// Returns true if "abc" is found at the beginning of $string.
ereg("^abc", $string);
// Returns true if "abc" is found at the end of $string.
ereg("abc$", $string);
// Returns true if client browser is Netscape 2, 3 or MSIE 3.
eregi("ozilla.[23]|MSIE.3", $HTTP_USER_AGENT);
// Places three space separated words into $regs[1], $regs[2] and $regs[3].
ereg("([[:alnum:]]+) ([[:alnum:]]+) ([[:alnum:]]+)", $string, $regs);
// Put a <br /> tag at the beginning of $string.
$string = ereg_replace("^", "<br />", $string);
// Put a <br /> tag at the end of $string.
$string = ereg_replace("$", "<br />", $string);
// Get rid of any newline characters in $string.
$string = ereg_replace("\n", "", $string);
?>

```

### Жишээ 2.

```

<?
if (ereg ("([0-9]{4})-([0-9]{1,2})-([0-9]{1,2})", $date, $regs)) {
    echo "$regs[3].$regs[2].$regs[1]";
} else {
    echo "Invalid date format: $date";
}
?>

```

### Жишээ 3.

```

<?
if (eregi("z", $string)) {
    echo "'$string' contains a 'z' or 'Z!";
}
?>

```

string доторх тэмдэгтийг өөр тэмдэгтээр солихдоо `ereg_replace()` түлхүүр үгийг ашиглана.

### Жишээ 1.

```

<?
$string = "This is a test";
echo str_replace(" is", " was", $string);
echo ereg_replace("( )is", "\\1was", $string);
echo ereg_replace("(( )is)", "\\2was", $string);
?>

```

### Жишээ 2.

```

<?
/* This will not work as expected. */
$num = 4;
$string = "This string has four words.";
$string = ereg_replace('four', $num, $string);
echo $string; /* Output: 'This string has words.' */
/* This will work. */
$num = '4';
$string = "This string has four words.";
$string = ereg_replace('four', $num, $string);
echo $string; /* Output: 'This string has 4 words.' */
?>

```

### Жишээ 3.

```

<?
$text = ereg_replace("([[:alpha:]]+//[^<>[:space:]]+[[:alnum:]]/)",
    "<a href=\"\\0\">\\0</a>", $text);
?>

```

string-ийг хуваахдаа `split()` гэсэн түлхүүр үгийг ашиглана.

### Жишээ 1.

```

<?
// Delimiters may be slash, dot, or hyphen
$date = "04/30/1973";
list($month, $day, $year) = split('[/.-]', $date);
echo "Month: $month; Day: $day; Year: $year<br />\n";
?>

```