PHP: Функции.

Функции, определяемые пользователем

PHP – язык процедурного программирования. Работа с ним предполагает знание средств языка для работы с функциями и умение применять их на практике

Определение функций пользователем

- Пользователь может объявить необходимые ему функции
- Общий синтаксис определения функций:
  function имя_функции ($par1, $par2, … , $parN)
  { инструкции блока действий;
  return; }
- Синтаксис вызова функций:
  имя_функции (var1, val2, … , expN);
  $var = имя_функции (var1, val2, … , expN);
- Функции могут быть объявлены в любом месте кода

Пример определения функции пользователем

- Объявляем функцию:

```
function MyEcho ($var)
{ echo '*** ', $var, ' ***'; }
```

- Вызываем функцию:

```
...
$a=2;
MyEcho ($a);
```

- В результате работы функции в поток выводится:

```
*** 2 ***
```

Возврат значений

- Функция может возвращать значение, но не более одного
- Возвращаемое значение может быть любого типа
- Значения возвращаются с помощью оператора возврата return
- Оператор return приводит к завершению выполнения функции и возврату к той строке кода, из которой функция была вызвана
- Возвращаемым функцией значением будет значение выражения в операторе return
- Оператор return в теле функции может отсутствовать. В этом случае возвращение из функции происходит после выполнения последней инструкции тела функции, а никакое значение не возвращается

Аргументы функций

- Три способа передачи данных в функцию через аргументы:
- Передача аргументов по значению:
  function_name ($a, $b, $c)
- передача аргументов по ссылке
  function_name ($a, &$b, $c)
- задание значения аргументов по умолчанию
  function_name ($a, $b, $c=1)

Передача аргументов по значению

- Задание передачи аргументов по значению в объявлении функции:
  function имя_функции ($par1, $par2)
  { *** }

- Синтаксис вызова функций:
  имя_функции (exp1, exp2)
  имя_функции (exp1, exp2, exp3, ***, expN);

- Для необъявленных аргументов действует передача аргументов по значению

Пример передачи аргументов функций по значению

```
<?php
function Sum ($a, $b) {
    $a=$a+$b; $c=$a;
    return $c};

$k=3; $l=4; $m=5;
$n=Sum ($k, $l);
$o=Sum ($m+2, 1);
echo $n, '<BR>', $o, '<BR>', $a;
?>
```

```
7
8
```

Передача аргументов по ссылке

- Задание передачи аргументов по ссылке в объявлении функции:
  function имя_функции ($par1, &$par2)
  { *** }

- Синтаксис вызова функций:
  имя_функции (exp1, $var)

- Фактическим аргументом, формального аргумента, указывающего на передачу по ссылке может выступать только переменная!
- Если формальный аргумент передает значения по ссылке, то фактический параметр не может быть опущен!

Пример передачи аргументов функций по ссылке

```
<?php
function Sum (&$a, $b) {
    $a=$a+$b ;
    $b=$a;
    return $a; };

$k=3; $l=4;
$n=Sum ($k, $l);
echo $k, '<BR>', $l, '<BR>', $n;
?>
```

```
7
4
7
```

Значение аргументов по умолчанию

- Задание значений аргументов по умолчанию в объявлении функции:
  function имя_функции ($par1=val1, $par2=val2)
  { *** }

- Синтаксис вызова функций с аргументами для которых задано значение по-умолчанию:
  имя_функции (exp1)

- Значения по умолчанию могут быть заданы только для аргументов, передаваемых по значению

Пример значений аргументов функций по умолчанию

```
<?php
function Sum ($a, $b=1) {
    return $a+$b};

$k=3; $l=4;
$n=Sum ($k, $l);
$o=Sum ($k);
echo $n, '<BR>', o;
?>
```

```
7
4
```

Пройденный материал:
Функции. Функции, определяемые пользователем

- Синтаксис объявления функций пользователем
- Возврат значений функции и оператор return
- Способы задания аргументов функций:
  - Передача аргументов по значению
  - Передача аргументов по ссылке
  - Задание значений аргументов по умолчанию
- Примеры различных способов задания аргументов