Основы PHP

Инструкция switch

Позволяет сравнивать выражение
с несколькими значениями и
отлично подходит для реализации выбора с большим количеством вариантов.

Синтаксис конструкции **switch-case**

```
switch(выражение) {
    case значение1:   команды1;   break;
    case значение2:   команды2;   break;
    . . .
    case значениеN:   командыN;   break;
}
```

```
$x=1;

if ($x == 0)
  { echo "x=0";  }
elseif ($x == 1)
  { echo "x=1";  }
 elseif ($x == 2)
  { echo "x=2";  }
```

```
switch ($x) {
    case 0:
      echo "x=0";
      break;
  case 1:
      echo "x=1";
      break;
  case 2:
      echo "x=2";
      break;
  }
```

```
$x="Яблоко";
switch ($x) {
case "Яблоко":
    echo "Это Яблоко";
    break;
case "Груша":
    echo "Это Груша";
    break;
case "Арбуз":
    echo "Это Арбуз";
    break;
```

Блок **default** – если ни одно значение из набора не совпало со значением выражения.

```
$x=3;
switch ($x) {
  case 0:
    echo "x=0";
    break;
 case 1:
   echo "x=1";
   break;
  default:
    echo "x не равен 0, 1";
 }
```

Циклы в PHP

Цикл с предусловием (while);

1. Вычисляется значение логического выражения.
2. Если значение истинно, выполняется тело цикла, иначе - переходим на следующий оператор.

Цикл с постусловием (do-while);

1. Проверяет значение выражения не до, а после каждого прохода (итерации)

Цикл с предусловием while

Кострукция:

while (логическое выражение)
инструкция;

Пример:

```
$x=0;
while ($x<10)
{
    $x++;   echo $x;
}
// Выводит 12345678910
```

Цикл с постусловием do while

проверяет значение выражения не до, а после каждого прохода (итерации)

Конструкция:

```
do
{
    тело цикла;
}
while (логическое выражение);
```

Пример

```
$x = 1;
do {
    echo $x;
}
while ($x++<10);
```

Цикл со счетчиком for

Цикл со счетчиком используется для выполнения тела цикла определенное число раз.

Синтаксис цикла:

```
for (инициализирующие команды; условие цикла; команды после итерации)
 {
    тело цикла;
 }
```

Цикл **for** начинает свою работу с выполнения инициализирующих команд. Данные команды выполняются только один раз. После этого проверяется условие цикла, если оно истинно (**true**), то выполняется тело цикла. После того, как будет выполнен последний оператор тела, выполняются команды после итерации. Затем снова проверяется условие цикла. Если оно истинно (**true**), выполняется тело цикла и команды после итерации, и т. д.

Пример

```
for ($x=0; $x<10; $x++) echo $x;
```

```
for ($x=0; $x<10; $x++)
{ echo $x; echo ’123’; }
```

```php
$rows = 10; // количество строк, tr
$cols = 10; // количество столбцов, td

echo '<table border="1">';

echo "<br>";

for ($tr=1; $tr<=$rows; $tr++){

    echo '<tr>';

    for ($td=1; $td<=$cols; $td++){
        echo '<td>'. 1 .'</td>';
    }
    echo '</tr>';
}

echo '</table>';
```