Циклические операторы

Цикл с предусловием for

```
for (expr1; expr2; expr3) {
    statement
}
```

Выражение expr1 выполняется один раз, до начала цикла.

Обычно в этом выражении производится инициализация переменной, которая будет счетчиком цикла.

Перед началом каждой итерации цикла проверяется условие expr2, и если оно истинно (true), то цикл выполняется.

В конце каждой итерации цикла выполняется выражение expr3, в котором чаще всего производится инкремент счетчика цикла.

Каждое из выражений может быть пустым.

Если выражение expr2 будет пустым, цикл станет бесконечным.

Задача №1

Напечатать таблицу соответствия между весом в фунтах и весом в килограммах для значений 1, 2, ..., 10.

P.S> 1 фунт = 453 г.

Цикл с предусловием while

```
while (expr) {
    statement
}
```

Цикл повторяет выполнение команды statement до тех пор, пока expr остается истинным (true).

Если условное выражение не изменяется в команде, то цикл будет выполняться бесконечно.

Если условное выражение изначально ложно (false), то блок команд выполняться не будет.

Задача №2

Спросить количество тарелок и количество моющего средства.

Моющее средство расходуется из расчета 0.5 на одну тарелку.

В цикле выводить сколько моющего средства осталось после мытья каждой тарелки.

В конце вывести сколько тарелок осталось, когда моющее средство закончилось или наоборот.

Цикл с постусловием do..while

```
do {
    statement
} while (expr)
```

Условное выражение expr проверяется на истинность после исполнения блока команд statement.

Цикл будет исполнен как минимум один раз.

Оператор break

Прерывает выполнение циклов for, while, foreach и конструкций switch.

Имеет необязательный аргумент, определяющий сколько вложенных циклов необходимо прервать.

```php
$i = 0;
while (++$i) {
    switch ($i) {
        case 5:
            echo 'At 5';
            break 1; #выйти только из switch
        case 10:
            echo 'At 10';
            break 2; #выйти только из switch и цикла while
        default:
            break;
    }
}
```

Оператор continue

Используется, чтобы прекратить выполнение оставшейся части цикла и перейти к следующей итерации.

Имеет необязательный аргумент, определяющий на сколько вложенных циклов должно распространяться его действие.

```
while ( list($key, $value) = each($arr) ) {
    if ( !($key % 2) ) {
        continue; #пропустить не четный элемент
    }
    #обработать четный элемент
}
```

Массивы

Определение

Массив в PHP – это упорядоченное отображение, которое устанавливает соответствие между *значением* и *ключом (индексом)*.

array array ([mixed …])

Языковая конструкция, которая возвращает созданный массив.

Элементы массива разделяются запятыми.

Допускается наличие запятой после последнего элемента массива.

В качестве индексов используются только целые числа и строки.

Индексы можно указывать с помощью переменной или функции, возвращающей целое или строковое значение.

Индексы и значения элементов разделяются оператором =>.

array[key]

Доступ к элементам массива осуществляется с помощью оператора [].

Нумерованные массивы

Массивы могут быть простыми индексированными списками.

Их называют «векторы» и в них индексом является число.

```
$months = array('Январь', 'Февраль', 'Март');
$months[]  = 'Апрель';  # $months[3] = 'Апрель'
$months[0] = 'Май';     # заменить существующий элемент
```

Нумерация начинается с нуля.

Можно использовать отрицательные индексы.

Если индекс не указывается, то используется числовое значение автоинкремента – максимальный положительный числовой индекс + 1.

```
$months = array(
    -2 => 'Январь',
          'Февраль',  #Какой будет индекс?
     3 => 'Март',
          'Апрель',   #Какой будет индекс?
);
```

Ассоциативные массивы

Массивы могут быть ассоциативными списками.

Их называют «хеш-таблицами» и в них в качестве индекса используется строковое значение.

```
$months = array(
    'january'  => 'Январь',
    'february' => 'Февраль',
    'march'    => 'Март',
);

$months['april'] = 'Апрель';

echo $months['february']; #Выведет "Февраль"
```

Если в качестве индекса указать пустую строку или NULL,

то PHP воспринимает это значение как пустую строку.

Массив может одновременно содержать и числовые и строковые индексы.

Инструкция foreach

```
foreach (array as value) {
    statement
}
```

Перебирает все элементы массива array.

При каждой итерации значение текущего элемента

присваивается переменной value.

```
foreach (array as key => value) {
    statement
}
```

При каждой итерации индекс элемента присваивается переменной key.

foreach оперирует копией массива.

Поэтому изменять элементы массива в конструкции foreach нельзя.

Задача №3

Преподаватель выдает два файла: lipsum.php и blog.php.

Файл lipsum.php содержит исходные данные для работы.

В файле blog.php необходимо реализовать вывод данных на экран в формате новостной ленты: жирный заголовок, дата публикации, краткое описание и ссылка «подробнее»:

- Дата хранится как Unix Timestamp. Необходимо вывести ее в удобочитаемом формате, используя функцию date().
- Заголовок и ссылка «подробнее» ведут на страницу с полным текстом. Формат url – blog.php?id=<уникальный_идентификатор>
- Каждая новость имеет статус публикации. Если новость не опубликована, то ее выводить не нужно.

Многомерные массивы

Каждый элемент массива может иметь значение любого типа.

Поэтому можно создавать вложенные массивы, формируя, таким образом, *многомерные* или даже древовидно-иерархические *массивы*.

```
$a = array(
    'orange' => array(
```

```
$a[][0] = $f;         # двухмерный
$a[1][0] = $f;        # двухмерный

$a['aa'][2] = $f;     # можно комбинировать
$a[3]['bb'] = $f;     # скалярные и ассоциативные индексы
$a[][][] = 'vs_07';   # так тоже можно

$a['aa'][4]['bb'][0] = $f;  # четырехмерный!
```

```
echo $a['orange']['taste']; # выведет "Сладкий"
```

Суперглобальные массивы

$GLOBALS

Содержит ссылки на все глобальные переменные сценария и на другие суперглобальные массивы, а также рекурсивную ссылку на самого себя.

$_SERVER

Ассоциативный массив переменных веб-сервера или среды выполнения текущего скрипта.

$_ENV

Ассоциативный массив переменных среды окружения.

Суперглобальные массивы

$_GET

Ассоциативный массив переменных, посылаемых сценарию методом HTTP GET (строка URL).

$_POST

Ассоциативный массив переменных, посылаемых сценарию методом HTTP POST (HTML формы).

$_COOKIE

Ассоциативный массив переменных, посылаемых сценарию через HTTP cookies.

$_REQUEST

Совокупность переменных из массивов $_GET, $_POST и $_COOKIE.

Суперглобальные массивы

$_FILES

Ассоциативный массив, содержащий информацию о файлах, загруженных на веб-сервер методом HTTP POST.

$_SESSION

Массив, содержащий переменные, в текущей момент зарегистрированные в сессии сценария.

Конец