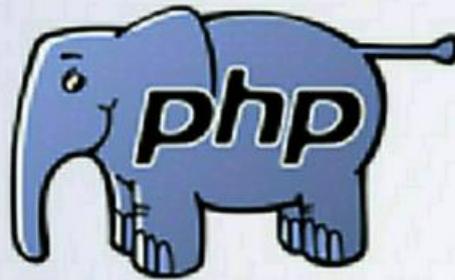


Роман Игоревич Григорьев
Задачник по PHP (с решениями)

Задачник
по



с решениями

12+

Автор: Григорьев Роман

Аннотация

Книга написана для начинающих программистов. Ко всем задачам есть решения. Присутствуют задачи повышенной сложности.

1)

Вычисления и условный оператор в PHP

1.1 Простейшая арифметика

Даны два числа 5 и 7. Найти их сумму и произведение.

1. `<?php`

2. `$a = 5;`

3. `$b = 7;`

4. `$c = $a + $b;`

5. `$d = $a*$b;`

6. `echo $c, $d;`

7. `?>`

Даны два числа 4 и 6. Найдите сумму их квадратов.

1. `<?php`

2. `$a = 4;`

3. `$b = 6;`

4. `$c = $a*$a + $b*$b;`

5. `echo $c;`

6. `?>`

Даны три числа 3, 5, 8. Найдите их среднее арифметическое.

1. `<?php`

2. `$a = 3;`

3. `$b = 5;`

4. `$c = 8;`

5. `$d = ($a + $b + $c)/3`

6. `echo $d;`

7. `?>`

Даны три числа $x = 2, y = 6$ и $z = 9$. Найдите $(x+1)^4 - 2(z - 2x^2 + y^2)$

1. `<?php`

2. `$x = 2;`

3. `$y = 6;`

4. `$z = 9;`

5. `$a = ($x+1)^4 - 2 ($z - 2*$x*$x+$y*$y);`

6. `echo $a`

7. `?>`

Даны три ненулевых числа $a = 4, b = 8, c = 3$. Найдите всевозможные результаты деления суммы двух из них на оставшееся третье число.

1. `<?php`

2. `$a = 4;`

3. `$b = 8;`

4. `$c = 3;`

5. `$d = ($a + $b)/$c;`

6. `$r = ($b+$c)/$a;`

7. `$f = ($a + $c)/$b;`

8. `echo $d,$r,$f;`

9. `?>`

Дано два числа 17 и 54. Найдите сумму 40% от первого числа и 84% от второго числа.

1. `<?php`

2. `$a = 17;`

```
3. $b = 54;
4. $c = 0,4*17 + 0,84*54;
5. echo $c;
```

6. ?>

Дано трехзначное числа. Найдите сумму его цифр.

1. <?php

```
2. $var = 578; // Объявляем число
3. $var .= ' '; // Приводим его к строковому типу
4. echo $a = $var[2] + $var[1] + $var[0]; // Выводим переменную
5. //равную сумме цифр числа
```

6. ?>

1.2. Условный оператор

Дано число 15. Если оно больше 10, то увеличьте его на 100, иначе уменьшите на 30.

1. <?php

```
2. $a = 15;
3. if ($a>10)
4. { $a = $a + 100};
5. else
6. { $a = $a - 30};
7. echo $a
```

8. ?>

Дано натуральное число 8. Если оно четное, то уменьшите его в 2 раза, иначе увеличьте в 3 раза.

1. <?php

```
2. if ($number / 2 == ceil($number / 2))
3. {
4. $number = $number / 2;
5. }
6. else
7. {
8. $number = $number *3;
9. }
10.
echo $number;
```

11.

?>

Дано число. Если оно не меньше 50, то выведите квадрат этого числа, если же это число больше 10 и меньше 30, то выведите ноль, в остальных случаях выведите слово "Ошибка"

1. <?php

```
2. $a = ee; // Задаем число $a, например, 45
3. if ($a > 50) {
4. echo $b=pow($a,2);
5. }
6. elseif ($a > 10 & $a < 30) {
7. echo 0;
8. }
9. else {
10. echo "Ошибка";
11.
}
```

12.

?>

Дано два числа \$a = 15, \$b = 4. Вывести наибольшее из них.

1. <?php

```
2. $a = 15;
3. $b = 4;
4. if ($a > $b)
5. { echo $a};
6. else
7. { echo $b };
```

8. ?>

Дано два числа \$a = 19, \$b = 143. Вывести 'Да', если они отличаются на 100, иначе вывести 'Нет'

1. <?php

```
2. $a = 19;
3. $b = 143;
4. if ($b - $a = 100 )
5. { echo "No" };
6. else
7. { echo "Yes" };
```

8. ?>

Дано два натуральных числа. Вывести 'Да', если они отличаются не более чем на 20, иначе вывести 'Нет'.

1. <?php

```
2. $a = 15;
3. $b = 30;
4. if ($b - $a > 20)
5. { echo "No" };
6. else
7. { echo "Yes" };
```

8. ?>

В данном трехзначном числе переставьте цифры так, чтобы новое число оказалось наибольшим из возможных.

1.3. Работа с формой

Пользователь вводит номер дня недели. Вывести название дня недели.

1. <?php

```
2. if ($a = 1) {echo Monday};
3. if ($a = 2) {echo Tuesday};
4. if ($a = 3) {echo Wednesday};
5. if ($a = 4) {echo Thursday};
6. if ($a = 5) {echo Friday};
7. if ($a = 6) {echo Saturday};
8. if ($a = 7) {echo Sunday};
9. else
10. {echo "неверное число"}
```

11.

?>

Пользователь вводит свой возраст. Если он больше 80 лет, то вывести 'Здравствуй, уважаемый', иначе 'Успехов!'.

1. <?php

```
2. if (0 < $years <= 80) {echo Успехов};
3. else
4. {echo "Здравствуй, Уважаемый"};
```

5. ?>

Пользователь выбирает из выпадающего списка страну (Турция, Египет или Италия), вводит количество дней для отдыха и указывает, есть ли у него скидка (чекбокс). Вывести стоимость отдыха, которая вычисляется как произведение количества дней на 400. Далее это число увеличивается на 10%, если выбран Египет,

и на 12%, если выбрана Италия. И далее это число уменьшается на 5%, если указана скидка.

```
1. <html>
2. <head>
3. <meta charset="utf-8" />
4. <title>Название странички</title>
5. </head>
6. <body>
7.
8.
9. <?php
10.
11. echo '<pre>';
12.
13. print_r($_POST);
14.
15. echo '</pre>';
16.
17. if (isset($_POST['hero']))
18. {
19.
20. if($_POST['hero'] == 1)
21. {
22.
23. if(isset($_POST['b']))
24. {
25.
26. echo " в Италию со скидкой ". ($_POST['a']*400*1.12*0.95);}
27.
28. else{echo " в Италию без скидки ". ($_POST['a']*400*1.12);}
29.
30. }
31.
32. elseif( $_POST['hero'] == 2){
33.
34. if(isset($_POST['b'])){
35.
36. echo " в Грецию со скидкой". ($_POST['a']*400*1.1*0.95);}
37.
38. else{echo"Греция без скидки".($_POST['a']*400*1.1);}
39.
40. }
41.
42. elseif($_POST['hero'] == 3){
43.
44. if(isset($_POST['b'])){
45.
46. echo " в Россию со скидкой". ($_POST['a']*400*0.95);}
47.
48. else{echo"Россия без скидки".($_POST['a']*400);}
49.
50. }
51.
52. }
```

32.
 }
 33.
 ?>
 34.
 <form method="post">
 35.
 36.
 <p><select name="hero"><option>выбирайте страну из списка </option> 37.
 <option value="1">Италия</option>
 38.
 <option value="2">Греция</option>
 39.
 <option value="3">Россия</option>
 40.
 </select></p>
 41.
 <p>сколько дней отдыха?</p>
 42.
 <input type="text" name="a" />
 43.
 <p>есть ли у вас скидка?</p>
 44.
 <input type="checkbox" name="option" value="b">
 45.
 <p><input type="submit" value="Отправить"></p>
 46.
 </form>
 47.
 48.
 49.
 </body>
 50.
 </html>

2) Массивы в PHP

2.1 Массивы

Дан массив с элементами 'Привет, ', 'мир' и '!'. Необходимо вывести на экран фразу 'Привет, мир!'.

1. <?php

2. //Выведем фразу 'Привет, мир!':

3. \$arr = ['Привет, ', 'мир', '!'];

4. echo \$arr[0].\$arr[1].\$arr[2];

5. ?>

Слово 'Привет, ' хранится под номером **0**, это значит, что для доступа к нему мы должны написать **\$arr[0]**.

Для доступа к слову 'мир' мы должны написать **\$arr[1]**, а **\$arr[2]** содержит в себе '!'. Далее с помощью оператора '**точка**' мы сложим три наши строки ('Привет, ', 'мир' и '!') в одну строку таким образом: **\$arr[0].\$arr[1].\$arr[2]**, и выведем на экран с помощью **echo**.

Задача. Решим немного другую задачу: дан массив с элементами 'Привет, ', 'мир' и '!'. Необходимо записать в переменную **\$text** фразу 'Привет, мир!', а затем вывести на экран содержимое этой переменной.

1. <?php

```

2. $arr = ['Привет, ', 'мир, '!'];
3. $text = $arr[0].$arr[1].$arr[2];
4. /*
5.

```

В переменной \$text теперь лежит строка 'Привет, мир!'.

6.

Выведем ее на экран:

```
7. */
```

```
8. echo $text;
```

```
9. ?>
```

Дан массив ['Привет, ', 'мир, '!']. Необходимо записать в первый элемент (то есть элемент с номером ноль) этого массива слово 'Пока, ' (то есть вместо слова 'Привет, ' будет 'Пока, ').

```
1. <?php
```

```
2. $arr = ['Привет, ', 'мир, '!'];
```

```
3. $arr[0] = 'Пока, '; //перезапишем первый элемент массива
```

```
4. var_dump($arr); //посмотрим на массив и убедимся в том, что он изменился
```

```
5. ?>
```

Дан массив с числами. Запишите в новый массив только те числа, в которых есть цифра 5.

```
1. <?php
```

```
2.
```

```
3. function getFives($sArr, String $sInt) {
```

```
4. $result = [];
```

```
5.
```

```
6. foreach ($sArr as $item)
```

```
7. if (strpos(strval($item), $sInt) !== false) $result[] = $item; 8. return $result;
```

```
9. }
```

```
10.
```

```
11. $searchInt = 5;
```

```
12. $searchArr = [2,1,4,3,5,7,6,9,8,11,10,15,29,25,52,13,51];
```

```
13.
```

```
14. echo '<pre>';
```

```
15. var_dump(getFives($searchArr, $searchInt));
```

```
16. echo '</pre>';
```

```
17.
```

```
?>
```

Задача «Странный массив»*:

Создайте массив вида [1, [2], [[3]], [[[4]]]], вместо 4 может быть любое целое значение. К примеру, если 5 – тогда будет массив [1, [2], [[3]], [[[4]]], [[[[5]]]]].

```
1. <?php
```

```
2. function getMultiArray($level) {
```

```
3. $result = $level;
```

```
4. for ($i = 1; $i < $level; $i++) {
```

```
5. $result = [$result];
```

```
6. }
```

```
7. return $result;
```

```
8. }
```

```
9.
```

```
10. function strange($value) {
```

```
11. $result = [];
```

```
12. for ($i = 1; $i <= $value; $i++) {
```

```
13. $result[$i] = getMultiArray($i);
```

```
14. }
```

```
15. return $result;
```

16.

}

17.

18. `print_r(strange(5));`

19.

?>

*Задача Хитрое слияние массивов**

1: Напишите функцию, которая будет сливать два массива таким образом: из, к примеру, [1, 2, 3] и ['a', 'b', 'c'] она сделает [1, 'a', 2, 'b', 3, 'c'].

2: Сделайте аналогичную функцию, которая параметрами будет принимать не два массива, а произвольное количество (пусть функция параметром принимает двухмерный массив, где его подмассивы – это то, что мы будем сливать).

1) <?php

2) **function** array_strange_merge(...\$arrays) {

3) \$maxLength = 0;

4) foreach (\$arrays as \$array) {

5) \$maxLength = max(count(\$array), \$maxLength);

6) }

7) \$result = [];

8) for (\$i = 0; \$i < \$maxLength; \$i++) {

9) foreach (\$arrays as \$array) {

10) if (isset(\$array[\$i])) {

11) \$result[] = \$array[\$i];

12) }

13) };

14) }

15) return \$result;

16)

}

17)

18) `print_r(array_strange_merge([1,2,3], ['a','b','c','d']));`

19)

?>

Дан массив вида [1, ' ', 2, ' ', ' ', 3] – то есть в нем есть пустые строки. Удалите все такие элементы из этого массива.*

1. <?php

2. `$uncleanList = [1, ' ', 2, ' ', ' ', 3];`

3. `$filteredList = array_filter($uncleanList);`

4. `var_dump($filteredList);`

5. `array(3) {`

6. [0]=>

7. int(1)

8. [2]=>

9. int(2)

10.

[5]=>

11.

int(3)

12.

}

13.

?>

Напишите функцию, которая корректно будет складывать часы и минуты.

Примеры: на вход функции подается такое – 1ч20мин+50мин – в результате функция

выведет 2ч10мин.*** (Задача решается разными способами, приведите пример одного из них)

1. <?php

2. function h2s(\$h) {

3. \$h = \$h/3600 % 24;

4. \$h = \$h % 10 ? "0".\$h : \$h;

5. \$h = \$h == 0 ? \$h = "00" : \$h;

6. return \$h;

7. }

8. function m2s(\$m) {

9. \$m = \$m/60 % 60;

10.

\$m = \$m % 10 ? "0".\$m : \$m;

11.

\$m = \$m == 0 ? \$m = "00" : \$m;

12.

return \$m;

13.

}

14.

function s(\$s) {

15.

\$s = \$s % 60;

16.

\$s = \$s % 10 ? "0".\$s : \$s;

17.

\$s = \$s == 0 ? \$s = "00" : \$s;

18.

return \$s;

19.

}

20.

function normal2seconds(\$normal) {

21.

\$hms = explode(":", \$normal);

22.

\$h = \$hms[0]*3600;

23.

\$m = \$hms[1]*60;

24.

\$s = \$hms[2];

25.

return \$h+\$m+\$s;

26.

}

27.

function seconds2normal(\$seconds) {

28.

\$v = max(\$seconds,0);

29.

\$h = h2s(\$v);

30.

\$m = m2s(\$v);

31.

\$s = s(\$s);

```

32.
return " $h:$m:$s";
33.
}
34.
function timePlus() {
35.
$times = func_get_args();
36.
for ($i=0;$i<count($times);$i++) {
37.
$nowtime = explode(":",$times[$i]);
38.
$time += normal2seconds(" $nowtime[0]:$nowtime[1]:$nowtime[2]"); 39.
}
40.
return seconds2normal($time);
41.
}
42.
echo timePlus("01:20:00","00:50:00");
43.
?>

```

2.2 Ассоциативные массивы

Создайте массив заработных плат **\$arr**. Выведите на экран зарплату Коли.

Решение: чтобы вывести зарплату Коли следует вывести значение элемента массива с ключом 'Коля' . Сделаем это:

```

1. <?php
2. $arr = ['Коля'=>'1000$', 'Вася'=>'500$', 'Петя'=>'200$'];
3. echo $arr['Коля']; //выведет 1000$
4. ?>

```

Создайте массив **\$arr** с элементами 1, 2, 3, 4, 5 двумя различными способами. Первый способ создать массив – объявить его через []:

```

1. <?php
2. $arr = [1, 2, 3, 4, 5];
3. ?>

```

Второй способ создания массива – это поступить таким образом:

```

1. <?php
2. $arr[] = 1;
3. $arr[] = 2;
4. $arr[] = 3;
5. $arr[] = 4;
6. $arr[] = 5;
7. ?>

```

Создайте массив **\$arr**. a=>1, b=>2, c=>3. Выведите на экран элемент с ключом 'b' .

```

1. <?php
2. $arr = ['a'=>1, 'b'=>2, 'c'=>3];
3. echo $arr['b'];
4. ?>

```

Создайте массив **\$arr**. Найдите сумму элементов этого массива.

```

$arr = ['a' => 1, 'b' => 2, 'c' => 3]; :
1. <?php
2. $arr = ['a'=>1, 'b'=>2, 'c'=>3];
3. echo $arr['a'] + $arr['b'] + $arr['c'];

```

4. ?>

5.

Создайте ассоциативный массив дней недели. Ключами в нем должны служить номера дней от начала недели (понедельник – должен иметь ключ 1, вторник – 2 и т.д.). Выведите на экран **текущий** день недели. (например сегодня – четверг)

Решение: пусть текущий день – четверг, тогда:

1. <?php

2. \$arr = [1=>'пн', 2=>'вт', 3=>'ср', 4=>'чт', 5=>'пт', 6=>'сб', 7=>'вс']; 3. echo \$arr[4];

4. ?>

Пусть теперь номер дня недели хранится в переменной **\$day**, например там

лежит

число

3.

Выведите

день

недели,

соответствующий

значению

переменной **\$day**.

1. <?php

2. //пусть текущий день – четверг, тогда:

3. \$arr = [1=>'пн', 2=>'вт', 3=>'ср', 4=>'чт', 5=>'пт', 6=>'сб', 7=>'вс']; 4. \$day = 3;

5. echo \$arr[\$day];

6. ?>

2.3 Многомерные массивы

Дан многомерный массив

1. \$arr = [

2. 'sp'=>['azul', 'rojo', 'verde'],

3. 'en'=>['blue', 'red', 'green'],

4.];

Выведите с его помощью слово **'azul'** .

1. <?php

2. \$arr = [

3. 'sp'=>['azul', 'rojo', 'verde'],

4. 'en'=>['blue', 'red', 'green'],

5.];

6. //так как массив многомерный

7. //(в нашем случае двухмерный),

8. //то придется написать несколько

9. //квадратных скобок подряд

10. var_dump(\$arr['sp']);

11. echo \$arr['sp'][0]; //выведет 'azul'

12. echo \$arr['sp'][1]; //выведет 'rojo'

13. echo \$arr['en'][1]; //выведет 'red'

14.

?>

Создайте массив **\$arr=['a', 'b', 'c']**. Выведите значение массива на экран с помощью функции **var_dump()**. С помощью массива **\$arr** из предыдущего номера выведите на экран содержимое первого, второго и третьего элементов.

1. <?php

2. \$arr = ['a', 'b', 'c'];

3. var_dump(\$arr)

4.

5. \$arr = ['a', 'b', 'c'];

6. echo \$arr[0].
';

```
7. echo $arr[1].<br>;
```

```
8. echo $arr[2].<br>;
```

```
9. ?>
```

Создайте массив `$arr=['a', 'b', 'c', 'd']` и с его помощью выведите на экран строку `'a+b, c+d'`.

```
1. <?php
```

```
2. $arr = ['a', 'b', 'c', 'd'];
```

```
3. echo $arr[0].'+' . $arr[1].'.' . $arr[2].'+' . $arr[3];
```

```
4. ?>
```

Создайте массив `$arr` с элементами **2, 8, 5, 3**. Умножьте первый элемент массива на второй, а третий элемент на четвертый. Результаты сложите, присвойте переменной `$a`. Выведите на экран значение этой переменной.

```
1. <?php
```

```
2. $arr = [2, 8, 5, 3];
```

```
3. $a = $arr[0] * $arr[1] + $arr[2] * $arr[3];
```

```
4. echo $a;
```

```
5. ?>
```

Заполните массив `$arr` числами от 1 до 5. Не объявляйте массив, а просто заполните его присваиванием `$arr[] = новое значение`.

```
1. <?php
```

```
2. $arr[] = 1;
```

```
3. $arr[] = 2;
```

```
4. $arr[] = 3;
```

```
5. $arr[] = 4;
```

```
6. $arr[] = 5;
```

```
7. var_dump($arr);
```

```
8. ?>
```

Создайте многомерный массив `$arr`.

```
1. $arr = [
```

```
2. 'cms'=>['joomla', 'wordpress', 'drupal'],
```

```
3. 'colors'=>['blue'=>'голубой', 'red'=>'красный', 'green'=>'зеленый']
```

```
4. ]
```

С его помощью выведите на экран слова `'joomla', 'drupal', 'verde', 'rojo'`.

```
1. <?php
```

```
2. $arr = [
```

```
3. 'cms'=>['joomla', 'wordpress', 'drupal'],
```

```
4. 'colors'=>['blue'=>'azul', 'red'=>'rojo', 'green'=>'verde']
```

```
5. ];
```

```
6.
```

```
7. echo $arr['cms'][0]; //joomla
```

```
8. echo $arr['cms'][2] //drupal;
```

```
9. echo $arr['colors']['green']; //verde
```

```
10. echo $arr['colors']['red']; //rojo
```

```
11.
```

```
?>
```

Создайте двухмерный массив. Первые два ключа – это `'ru'` и `'en'`. Пусть первый ключ содержит элемент, являющийся массивом названий дней недели *по-русски*, а второй – *по-английски*. Выведите с помощью этого массива понедельник по-русски и среду по английски (пусть понедельник – это первый день).

```
1. <?php
```

```
2. $arr = [
```

```
3. 'ru'=>[1=>'пн', 'вт', 'ср', 'чт', 'пт', 'сб', 'вс'],
```

```
4. 'en'=>[1=>'mn', 'ts', 'wd', 'th', 'fr', 'st', 'sn'],
```

```
5. ];
```

```
6. echo $arr['ru'][1]; //пн
```

```
7. echo $arr['en'][2]; //wd
```

8. ?>

... Пусть теперь в переменной **\$lang** хранится язык (она принимает одно из значений или 'ru', или 'en' – либо то, либо то), а в переменной **\$day** – номер дня.

Выведите словом день недели, соответствующий переменным **\$lang** и **\$day**. То есть: если, к примеру, **\$lang = 'ru'** и **\$day = 3** – то выведем 'среда'.

1. <?php

```
2. $arr = [
```

```
3. 'ru'=>['пн', 'вт', 'ср', 'чт', 'пт', 'сб', 'вс'],
```

```
4. 'en'=>['mn', 'ts', 'wd', 'th', 'fr', 'st', 'sn'],
```

```
5. ];
```

```
6. $lang = 'ru';
```

```
7. $day = 3;
```

```
8. echo $arr[$lang][$day]; //cp
```

9. ?>

Дан многомерный массив **\$arr**. Напишите функцию, которая принимает строки вида 'строка1.строка2.строка3' – буквы разделенные точками, а возвращает элемент многомерного массива **\$arr['строка1']['строка2']['строка3']**. Количество точек в строке может быть любым, вложенность массива тоже любая, ключи массива не содержат точек. ***

1. <?php

```
2. function array_get($array, $key) {
```

```
3. $keys = explode('.', $key);
```

```
4. $result = $array;
```

```
5. while ($k = array_shift($keys)) {
```

```
6. if (empty($result[$k])) { throw new Exception('element '.$key.' found'); }
```

```
7. $result = $result[$k];
```

```
8. }
```

```
9. return $result;
```

```
10.
```

```
}
```

```
11. echo array_get([1,2,[1=>2, 2 => [1,5,3]]], '2.2.1'), PHP_EOL;
```

```
12. echo array_get([1,2, 'table' => [1=>2, 'model' => [1, 'key' => 'coool',3]]],  
'table.model.key'), PHP_EOL;
```

13.

?>

Дан многомерный массив. Уровни вложенности его подмассивов могут быть разными, например:

```
$arr = array(
```

```
array(
```

```
1, 2, 3,
```

```
),
```

```
array(
```

```
array(4, 5, array(6, 7, 7)), 9, 10,
```

```
),
```

```
).
```

Выведите на экран столбец всех его элементов.*

1. <?php

```
2. $arr = array(
```

```
3. array(
```

```
4. 1, 2, 3,
```

```
5. ),
```

```
6. array(
```

```
7. array(4, 5, array(6, 7, 8)), 9, 10,
```

```
8. ),
```

```

9. );
10.
11. $new = array();
12. function sort_array($arr){
13. foreach($arr as $value){
14. if(!is_array($value)){
15. echo $value."<br />";
16. } else{
17. sort_array($value);
18.
19. }
20.
21. }
22.
23. sort_array($arr);
24.
?>

```

3) Задачи на if else, switch-case

Переменная **\$num** может принимать одно из значений: 1, 2, 3 или 4. Если она имеет значение '1', то в переменную **\$result** запишем 'зима', если имеет значение '2' – 'лето' и так далее. Решите задачу через **switch-case**.

1. <?php

```

2. $num = 2;
3. switch ($num) {
4. case 1:
5. $result = 'зима';
6. break;
7. case 2:
8. $result = 'весна';
9. break;
10. case 3:
11. $result = 'лето';
12. break;
13. case 4:
14. $result = 'осень';
15. break;
16.
17. }
17. echo $result;
18.
?>

```

В переменной **\$day** лежит какое-то число из интервала от **1** до **31**. Определите в какую декаду месяца попадает это число (в первую, вторую или третью).

1. <?php

```

2. $day = 12;
3. if ($day <= 10) {
4. $decade = 1;
5. }
6. if ($day >= 11 and $day <= 20) {
7. $decade = 2;
8. }
9. if ($day >= 21 and $day <= 31) {

```

```
10. $decade = 3;
11.
}
12. echo $decade;
13.
?>
```

В переменной **\$month** лежит какое-то число из интервала от **1** до **12**.
Определите в какую пору года попадает этот месяц (зима, лето, весна, осень).

```
1. <?php
2. $month = 6;
3. if ($month == 12 or $month == 1 or $month == 2) {
4. $result = 'Зима';
5. }
6. if ($month >= 3 and $month <= 5) {
7. $result = 'Весна';
8. }
9. if ($month >= 6 and $month <= 8) {
10.
$result = 'Лето';
11.
}
12.
if ($month >= 9 and $month <= 11) {
13.
$result = 'Осень';
14.
}
15.
echo $result;
16.
?>
```

В переменной **\$year** хранится год. Определите, является ли он **високосным** (в таком году есть 29 февраля). Год будет високосным в **двух случаях**: либо он делится на **4**, но при этом не делится на **100**, либо делится на **400**. Так, годы 1700, 1800 и 1900 не являются високосными, так как они делятся на **100** и не делятся на **400**. Годы 1600 и 2000 – високосные, так как они делятся на **400**.

```
1. <?php
2. $year = 2000;
3. if ($year % 4 == 0 or ($year % 400 == 0 and $year % 100 != 0)) {
4. echo 'Високосный';
5. } else {
6. echo 'Не високосный';
7. }
8. ?>
```

Дана строка с символами, например, 'abcde'. Проверьте, что первым символом этой строки является буква 'a'. Если это так – выведите 'да', в противном случае выведите 'нет'.

```
1. <?php
2. $str = 'abcde';
3. if ($str[0] == 'a') {
4. echo 'Да';
5. } else {
6. echo 'Нет';
7. }
8. ?>
```

Дана строка с цифрами, например, '12345'. Проверьте, что первым символом этой строки является цифра 1, 2 или 3. Если это так – выведите 'да', в противном случае выведите 'нет'.

1. <?php

```
2. $str = '12345';
3. if ($str[0] == '1' or $str[0] == '2' or $str[0] == '3') {
4. echo 'Да';
5. } else {
6. echo 'Нет';
7. }
```

8. ?>

Дана строка из 3-х цифр. Найдите сумму этих цифр. То есть сложите как числа первый символ строки, второй и третий.

1. <?php

```
2. $str = '123';
3. $result = $str[0] + $str[1] + $str[2];
4. echo $result;
```

5. ?>

Дана строка из 6-ти цифр. Проверьте, что сумма первых трех цифр равняется сумме вторых трех цифр. Если это так – выведите 'да', в противном случае выведите 'нет'.

1. <?php

```
2. $str = '123231';
3. $sum1 = $str[0] + $str[1] + $str[2];
4. $sum2 = $str[3] + $str[4] + $str[5];
5. if ($sum1 == $sum2) {
6. echo 'Да';
7. } else {
8. echo 'Нет';
9. }
```

10.

?>

4) Циклы foreach, while, for

Заполните массив 10-ю иксами с помощью цикла.

1. <?php

```
2. $arr = [];
3. for ($i = 1; $i <= 10; $i++) {
4. $arr[] = 'x';
5. }
6. var_dump($arr);
```

7. ?>

Заполните массив числами от 1 до 10 с помощью цикла.

1. <?php

```
2. $arr = [];
3. for ($i = 1; $i <= 10; $i++) {
4. $arr[] = $i;
5. }
6. var_dump($arr);
```

7. ?>

Заполните массив числами от 10 до 1 с помощью цикла.

1. <?php

```
2. $arr = [];
3. for ($i = 10; $i > 0; $i--) {
4. $arr[] = $i;
5. }
```

6. `var_dump($arr);`

7. `?>`

Выведите столбец чисел от 1 до 100.

Задачу можно решить как циклом `for`, так и циклом `while`. Вначале решим через цикл `while`:

1. `<?php`

2. `$i = 1; //счетчик цикла устанавливаем в 1`

3. `while ($i <= 100) {`

4. `echo $i.
; //выведет 1,2... 100`

5. `/*`

6. *С помощью оператора `++` увеличиваем `$i`*

7. *на единицу при каждом проходе цикла:*

8. `*/`

9. `$i++;`

10.

`}`

11.

`?>`

Решение через цикл `for`

1. `<?php`

2. `/*`

3. *В начале цикла `$i` будет равно 1,*

4. *цикл будет выполняться пока `$i <= 100`,*

5. *после каждого прохода к `$i` прибавляется единица:*

6. `*/`

7. `for ($i = 1; $i <= 100; $i++) {`

8. `echo $i.
; //выведет 1,2... 100`

9. `}`

10.

`?>`

Работа с `foreach`

Дан массив с элементами `'html'`, `'css'`, `'php'`, `'js'`, `'jq'`. С помощью цикла `foreach` выведите эти слова в столбик.

1. `<?php`

2. `$arr = ['html', 'css', 'php', 'js', 'jq'];`

3. `foreach ($arr as $elem) {`

4. `echo $elem.
;`

5. `}`

6. `?>`

Дан массив с элементами 1, 2, 3, 4, 5. С помощью цикла `foreach` найдите сумму элементов этого массива. Запишите ее в переменную `$result`.

1. `<?php`

2. `$arr = [1, 2, 3, 4, 5];`

3. `$result = 0;`

4. `foreach ($arr as $elem) {`

5. `$result += $elem;`

6. `}`

7. `echo $result;`

8. `?>`

Дан массив с элементами 1, 2, 3, 4, 5. С помощью цикла `foreach` найдите сумму квадратов элементов этого массива. Результат запишите переменной `$result`.

1. `<?php`

2. `$arr = [1, 2, 3, 4, 5];`

3. `$result = 0;`

4. `foreach ($arr as $elem) {`

```
5. $result += $elem * $elem;
6. }
7. echo $result;
8. ?>
```

Дан массив \$arr. \$arr = ['green'=>'зеленый', 'red'=>'красный','blue'=>'голубой']; С помощью цикла foreach выведите на экран столбец ключей и элементов в формате 'green – зеленый'.

```
1. <?php
2. $arr = ['green'=>'зеленый', 'red'=>'красный','blue'=>'голубой'];
3. foreach ($arr as $key=>$elem) {
4. echo $key.' – '.$elem.<br>;
5. }
6. ?>
```

Дан массив \$arr с ключами 'Коля', 'Вася', 'Петя' и с элементами '200', '300', '400'. С помощью цикла foreach выведите на экран столбец строк такого формата: 'Коля – зарплата 200 долларов.'

```
1. <?php
2. $arr = ['Коля'=>'200', 'Вася'=>'300','Петя'=>'400'];
3. foreach ($arr as $key=>$elem) {
4. echo $key.' – зарплата '.$elem.' долларов<br>;
5. }
6. ?>
```

5) Строки в PHP

5.1 Работа с регистром символов

Для решения задач данного блока вам понадобятся следующие функции: *strtolower*, *strtoupper*, *ucfirst*, *lcfirst*, *ucwords*.

Дана строка 'php'. Сделайте из нее строку 'PHP'.

```
1. <?php
2. echo strtoupper('php');
3. ?>
```

Дана строка 'PHP'. Сделайте из нее строку 'php'.

```
1. <?php
2. echo strtolower('PHP');
3. ?>
```

Дана строка 'london'. Сделайте из нее строку 'London'.

```
1. <?php
2. echo ucfirst('london');
3. ?
```

Дана строка 'London'. Сделайте из нее строку 'london'.

```
1. <?php
2. echo lcfirst('London');
3. ?>
```

Дана строка 'london is the capital of great britain'. Сделайте из нее строку 'London Is The Capital Of Great Britain'.

```
1. <?php
2. echo ucwords('london is the capital of great britain');
3. ?>
```

Дана строка 'LONDON'. Сделайте из нее строку 'London'.

```
1. <?php
2. echo ucfirst(strtolower('LONDON'));
3. ?>
```

Работа с strlen

Для решения задач данного блока вам понадобятся следующие функции: *strlen*.

Дана строка **'html css php'** . Найдите количество символов в этой строке.

1. <?php

2. echo strlen('html css php');

3. ?>

Дана переменная **\$password**, в которой хранится пароль пользователя. Если количество символов пароля больше 5-ти и меньше 10-ти, то выведите пользователю сообщение о том, что пароль подходит, иначе сообщение о том, что нужно придумать другой пароль.

1. <?php

2. \$password = '123456789';

3. \$len = strlen(\$password);

4. if (\$len > 5 and \$len < 10) {

5. echo 'Пароль подходит';

6. } else {

7. echo 'Пароль не подходит, придумайте другой';

8. }

9. ?>

Работа с substr

Для решения задач данного блока вам понадобятся следующие

функции: substr

Дана строка **'html css php'** . Вырежьте из нее и выведите на экран слово 'html', слово 'css' и слово 'php'.

1. <?php

2. echo substr('html css php', 0, 4); //html

3. echo substr('html css php', 5, 3); //css

4. echo substr('html css php', 9, 3); //php

5. ?>

Дана строка. Вырежьте и выведите на экран последние **3** символа этой строки.

1. <?php

2. echo substr('123456789', -3); //789

3. ?>

Дана строка. Проверьте, что она начинается на **'http://'** . Если это так, выведите 'да', если не так – 'нет'.

1. <?php

2. \$str = 'http://php.net';

3. if (substr(\$str, 0, 7) == 'http://') {

4. echo 'Да';

5. } else {

6. echo 'Нет';

7. }

8. ?>

Дана строка. Проверьте, что она начинается на **'http://'** или на **'https://'** . Если это так, выведите 'да', если не так – 'нет'.

1. <?php

2. \$str = 'http://phphtml.net';

3. \$check1 = substr(\$str, 0, 7);

4. \$check2 = substr(\$str, 0, 8);

5. if (\$check1 == 'http://' or \$check2 == 'https://') {

6. echo 'Да';

7. } else {

8. echo 'Нет';

9. }

10.

?>

Дана строка. Проверьте, что она заканчивается на **'.png'** . Если это так, выведите

'да', если не так – 'нет'.

```
1. <?php
2. $str = '1.png';
3. $check = substr($str, -4);
4. if ($check == '.png') {
5. echo 'Да';
6. } else {
7. echo 'Нет';
8. }
9. ?>
```

Дана строка. Проверьте, что она заканчивается на **'png'** или на **'jpg'** . Если это так, выведите 'да', если не так – 'нет'.

```
1. <?php
2.
3. $str = '1.png';
4.
5. $check = substr($str, -4);
6.
7. if ($check == '.png' or $check == '.jpg') {
8.
9. echo 'Да';
10. } else {
11.
12. echo 'Нет';
13. }
14. ?>
```

Дана строка. Если в этой строке более 5-ти символов – вырежьте из нее первые 5 символов, добавьте троеточие в конец и выведите на экран. Если же в этой строке 5 и менее символов – просто выведите эту строку на экран.

```
1. <?php
2. $str = '123456789';
3. if(strlen($str) > 5) {
4. echo substr($str, 0, 5). '...';
5. } else {
6. echo $str;
7. }
8. ?>
```

Работа с `str_replace`

Для решения задач данного блока вам понадобятся следующие функции: `str_replace`.

Дана строка **'31.12.2013'** . Замените все точки на дефисы.

```
1. <?php
2. echo str_replace('.', '-', '31.12.2013');
3. ?>
```

Дана строка **\$str**. Замените в ней все буквы 'a' на цифру 1, буквы 'b' – на 2, а буквы 'c' – на 3.

```
1. <?php
2. echo str_replace(['a', 'b', 'c'], [1, 2, 3], 'abcabc'); //выведет '123123'
3. ?>
```

Дана строка с буквами и цифрами, например, **'1a2b3c4b5d6e7f8g9h0'**.

Удалите из нее все цифры. То есть в нашем случае должна получиться строка **'abcdbefgh'**.

```
1. <?php
```

```
2. echo str_replace([1, 2, 3, 4, 5, 6, 7, 8, 9, 0], ' ', '1a2b3c4b5d6e7f8g9h0');
```

```
3. ?>
```

Работа с `strtr`

Для решения задач данного блока вам понадобятся следующие функции: `strtr`.

Дана строка `$str`. Замените в ней все буквы 'a' на цифру 1, буквы 'b' – на 2, а буквы 'c' – на 3. Решите задачу двумя способами работы с функцией `strtr` (массив замен и две строки замен).

Первый способ:

```
1. <?php
```

```
2. $str = 'abcabc';
```

```
3. echo strtr($str, ['a'=>'1', 'b'=>'2', 'c'=>'3']); //выведет '123123'
```

```
4. ?>
```

Второй способ:

```
1. <?php
```

```
2. $str = 'abcabc';
```

```
3. echo strtr($str, 'abc', '123'); //выведет '123123'
```

```
4. ?>
```

Работа с `substr_replace`

Для решения задач данного блока вам понадобятся следующие функции: `substr_replace`.

Дана строка `$str`. Вырежьте из нее подстроку с 3-го символа (отсчет с нуля), 5 штук и вместо нее вставьте '!!!'.

```
1. <?php
```

```
2. $str = '1234567890';
```

```
3. echo substr_replace($str, '!!!', 3, 5); //выведет '123!!!90'
```

```
4. ?>
```

Работа с `strpos`, `strrpos`

Для решения задач данного блока вам понадобятся следующие функции: `strpos`, `strrpos`.

Дана строка 'abc abc abc'. Определите позицию первой буквы 'b'.

```
1. <?php
```

```
2. echo strpos('abc abc abc', 'b'); //выведет 1
```

```
3. ?>
```

Дана строка 'abc abc abc'. Определите позицию последней буквы 'b'.

```
1. <?php
```

```
2. echo strrpos('abc abc abc', 'b'); //выведет 9
```

```
3. ?>
```

Дана строка 'abc abc abc'. Определите позицию первой найденной буквы 'b', если начать поиск не с начала строки, а с позиции 3.

```
1. <?php
```

```
2. echo strpos('abc abc abc', 'b', 3); //выведет 5
```

```
3. ?>
```

Дана строка 'aa aa aa aa'. Определите позицию второго пробела.

```
1. <?php
```

```
2. $str = 'aa aa aa aa';
```

```
3. $first = strpos($str, ' '); //позиция первого пробела
```

```
4. echo strpos($str, ' ', $first + 1);
```

```
5. ?>
```

Проверьте, что в строке есть две точки подряд. Если это так – выведите 'есть', если не так – 'нет'.

```
1. <?php
```

```
2. $str = 'aaa..aaaa';
```

```
3. if(strpos($str, '..') !== false) {
```

```
4. echo 'Да';
```

```
5. } else {
```

6. echo 'Нет';

7. }

8. ?>

Проверьте, что строка начинается на **'http://'** . Если это так – выведите 'да', если не так – 'нет'.

1. <?php

2. \$str = 'http://phphtml.net';

3.

4. //Проверяем, что строка начинается с http://:

5. if(strpos(\$str, 'http://') === 0) {

6. echo 'Да';

7. } else {

8. echo 'Нет';

9. }

10.

?>

Работа с explode, implode

Для решения задач данного блока вам понадобятся следующие функции: *explode, implode*.

Дана строка **'html css php'** . С помощью функции *explode* запишите каждое слово этой строки в отдельный элемент массива

1. <?php

2. \$arr = explode(' ', 'html css php');

3. var_dump(\$arr);

4. ?>

Дан массив с элементами **'html', 'css', 'php'** . С помощью функции *implode* создайте строку из этих элементов, разделенных запятыми.

1. <?php

2. \$arr = ['html', 'css', 'php'];

3. echo implode(',', \$arr);

4. ?>

В переменной **\$date** лежит дата в формате **'2016-12-31'** . Преобразуйте эту дату в формат **'31.12.2016'** .

1. <?php

2. \$date = '2016-12-31';

3. \$arr = explode('-', \$date);

4. echo \$arr[2].'.'.\$arr[1].'.'.\$arr[0];

5. ?>

Работа с str_split

Для решения задач данного блока вам понадобятся следующие функции: *str_split*.

Дана строка **'1234567890'** . Разбейте ее на массив с элементами **'12', '34', '56', '78', '90'** .

1. <?php

2. var_dump(str_split('1234567890', 2));

3. ?>

Дана строка **'1234567890'** . Разбейте ее на массив с элементами **'1', '2', '3', '4', '5', '6', '7', '8', '9', '0'** .

1. <?php

2. var_dump(str_split('1234567890', 1));

3. ?>

Дана строка **'1234567890'** . Сделайте из нее строку **'12-34-56-78-90'** не используя цикл.

1. <?php

```
2. echo implode('-', str_split('1234567890', 2));
3. ?>
```

Работа с trim, ltrim, rtrim

Для решения задач данного блока вам понадобятся следующие функции: trim, ltrim, rtrim.

Дана строка. Очистите ее от концевых пробелов.

1. <?php

```
2. $str = ' строка ';
3. echo trim($str);
```

4. ?>

Дана строка '/php/'. Сделайте из нее строку 'php', удалив концевые слешы.

1. <?php

```
2. $str = '/php/';
3. echo trim($str, '/');
```

4. ?>

Дана строка 'слова слова слова.'. В конце этой строки может быть точка, а может и не быть. Сделайте так, чтобы в конце этой строки гарантировано стояла точка. То есть: если этой точки нет – ее надо добавить, а если есть – ничего не делать.

Задачу решите через rtrim без всяких ифов.

1. <?php

```
2. $str = 'слова слова слова.';
3. echo rtrim($str, '.');
```

4. ?>

Работа с strev

Для решения задач данного блока вам понадобятся следующие функции: strev.

Дана строка '12345'. Сделайте из нее строку '54321'.

1. <?php

```
2. $str = '12345';
3. echo strev($str);
```

4. ?>

Проверьте, является ли слово **палиндромом** (одинаково читается во всех направлениях, примеры таких слов: **madam, otto, kayak, nun, level**).

1. <?php

```
2. $str = 'madam';
3. $rev = strev($str);
4. if ($rev == $str) {
5. echo $str. ' – это слово палиндром';
6. }
```

7. ?>

Работа с str_shuffle

Для решения задач данного блока нужны функции: str_shuffle

Дана строка. Перемешайте символы этой строки в случайном порядке.

1. <?php

```
2. $str = 'Hello World!';
3. echo str_shuffle($str);
```

4. ?>

Создайте строку из 6-ти случайных маленьких латинских букв так, чтобы буквы не повторялись. Нужно сделать так, чтобы в нашей строке могла быть любая латинская буква, а не ограниченный набор.

1. <?php

```
2. $str = 'qwertyuiopasdfghjklzxcvbnm'; //все латинские буквы
3. $newStr = str_shuffle($str); //перемешиваем их
4. echo substr($newStr, 0, 6); //берем первые 6 символов
```

5. ?>

Работа с number_format

Для решения задач данного блока вам понадобятся следующие функции: number_format.

Дана строка '12345678'. Сделайте из нее строку '12 345 678'.

1. <?php

2. \$str = '12345678';

3. echo number_format(\$str, 0, ',', '');

4. ?>

Работа с str_repeat

Для решения задач данного блока вам понадобятся следующие функции: str_repeat

Нарисуйте пирамиду, как показано на рисунке, только у вашей пирамиды должно быть 9 рядов, а не 5. Решите задачу с помощью **одного цикла** и функции str_repeat.

x

xx

xxx

xxxx

xxxxx

1. <?php

2. for (\$i = 1; \$i <= 9; \$i++) {

3. echo str_repeat('x' , \$i).
;

4. }

5. ?>

Нарисуйте пирамиду, как показано на рисунке. Решите задачу с помощью **одного цикла** и функции str_repeat.

22

333

4444

55555

666666

7777777

88888888

999999999

1. <?php

2. for (\$i = 1; \$i <= 9; \$i++) {

3. echo str_repeat(\$i , \$i).
;

4. }

5. ?>

Работа с strip_tags и htmlspecialchars

Для решения задач данного блока вам понадобятся следующие функции: htmlspecialchars, strip_tags.

Дана строка 'html, php, js'. Удалите теги из этой строки.

1. <?php

2. echo strip_tags('html, php, js');

3. ?>

Дана строка \$str. Удалите все теги из этой строки, кроме тегов и <i>.

1. <?php

2. echo strip_tags(\$str, '<i>');

3. ?>

Дана строка 'html, php, js'. Выведите ее на экран 'как есть': то есть браузер не должен преобразовать в жирный.

1. <?php

2. echo htmlspecialchars('html, php, js');

3. ?>

Работа с chr и ord

Для решения задач данного блока вам понадобятся следующие функции: *chr*, *ord*.

Узнайте код символов 'a', 'b', 'c', пробела.

1. <?php

2. echo ord('a');

3. echo ord('b');

4. echo ord('c');

5. echo ord(' ');

6. ?>

Изучите таблицу ASCII. Определите границы, в которых располагаются буквы английского алфавита.

Выведите на экран символ с кодом 33.

1. <?php

2. echo chr(33);

3. ?>

Запишите в переменную **\$str** случайный символ – большую букву латинского алфавита. Подсказка: с помощью таблицы ASCII определите какие целые числа соответствуют большим буквам латинского алфавита.

1. <?php

2. \$str = chr(rand(65, 90));

3. echo \$str;

4. ?>

Запишите в переменную **\$str** случайную строку **\$len** длиной, состоящую из маленьких букв латинского алфавита. Подсказка: воспользуйтесь циклом *for* или *while*.

1. <?php

2. \$str = ' ';

3. \$len = 5;

4. for (\$i = 1; \$i <= \$len; \$i++) {

5. \$str .= chr(rand(65, 90));

6. }

7. echo \$str;

8. ?>

Дана буква английского алфавита. Узнайте, она маленькая или большая.

1. <?php

2. \$str = 'A';

3. \$code = ord(\$str);

4. if (\$code >= 65 and \$code <= 90) {

5.

echo 'большая буква';

6. }

7. if (\$code >= 97 and \$code <= 122) {

8.

echo 'маленькая буква';

9. }

10.

?>

Работа с strchr, strrchr

Для решения задач данного блока вам понадобятся следующие функции: *strchr*, *strrchr*.

Дана строка 'ab-cd-ef'. С помощью функции **strchr** выведите на экран строку '-

cd-ef .

1. **<?php**
2. \$str = 'ab-cd-ef';
3. echo strchr(\$str, '-');
4. **?>**

Дана строка **'ab-cd-ef'** . С помощью функции **strchr** выведите на экран строку **'-ef'** .

1. **<?php**
2. \$str = 'ab-cd-ef';
3. echo strrchr(\$str, '-');
4. **?>**

Работа с strstr

Для решения задач данного блока вам понадобятся следующие функции: strstr.

Дана строка **'ab-cd-ef'** . С помощью функции **strstr** выведите на экран строку **'-cd-ef'** .

1. **<?php**
2. \$str = 'ab-cd-ef';
3. echo strstr(\$str, '-');
4. **?>**

5.3 Применение строк в PHP

Пользователь вводит названия городов через пробел. Переставьте названия так, чтобы названия были упорядочены по алфавиту.

1. **<? php**
2. \$text = "london Moskow NewYork";
3. \$cityes = explode(" ", \$text);
4. sort(\$cityes);
5. \$cityes = implode(" ", \$cityes);
6. echo \$cityes;
7. **?>**

Преобразуйте строку **'var_test_text'** в **'varTestText'** . Скрипт, конечно же, должен работать с любыми аналогичными строками.

1. **<?php**
2. \$arr = explode('_', 'var_test_text');
3. \$str = ' ';
4. foreach (\$arr as \$key=>\$elem) {
5. if(\$key != 0) {
6. \$str .= ucfirst(\$elem);
7. } else {
8. \$str .= \$elem;
9. }
10. }
11. echo \$str;
12. **?>**

Дана строка следующего вида: 5 цифр, затем пробел, затем еще 5 цифр.

Например, дана такая строка **'12345 67890'** . Сделайте из нее строку **'54321 09876'** .

1. **<?php**
2. \$str = '12345 67890';
3. \$arr = explode(' ', \$str);
4. \$arr[0] = strrev(\$arr[0]);
5. \$arr[1] = strrev(\$arr[1]);

```
6. echo implode(' ', $arr);
```

```
7. ?>
```

Дан массив с числами. Выведите на экран все числа, в которых есть цифра 3.

```
1. <?php
```

```
2. $arr = [3, 11, 13, 31, 28, 34];
```

```
3. foreach ($arr as $elem) {
```

```
4. if (strpos($elem, '3') !== false) {
```

```
5. echo $elem . '<br>';
```

```
6. }
```

```
7. }
```

```
8. ?>
```

Дан массив с числами. Посчитайте суммарное количество цифр 3 в этих числах.

```
1. <?php
```

```
2. $arr = [3, 11, 13, 31, 28, 34];
```

```
3. $i = 0;
```

```
4. foreach ($arr as $elem) {
```

```
5. if (strpos($elem, '3') !== false) {
```

```
6. $i++;
```

```
7. }
```

```
8. }
```

```
9. echo
```

```
10. ?>
```

Дана строка с цифрами, например: '12345678'. Просуммируйте числа этой строки вот таким образом: 12+34+56+78. Строка, конечно же, может быть произвольной (но только с цифрами внутри)

```
1. <?php
```

```
2. function sum_xx($str)
```

```
3. {
```

```
4. $array=str_split($str, 2);
```

```
5. echo array_sum($array)."<br/>";
```

```
6. }
```

```
7. sum_xx("123456789");
```

```
8. sum_xx("987654321");
```

```
9. ?>
```

Дана строка вида /aaa/222/we33/qqq/. Сделайте функцию, которая заменит энный блок в такой строке на заданный текст. Пример:

func('/aaa/222/we33/qqq', 2, '!') выдаст '/aaa/222!/qqq/ – заменили второй блок (нумерация с нуля).

```
1. <?php
```

```
2. function func($my_string, $block_num, $symbol){
```

```
3. $my_arr = explode("/", $my_string);
```

```
4. $my_arr[$block_num] = $symbol;
```

```
5. $my_string = implode("/", $my_arr);
```

```
6. return $my_string;
```

```
7. }
```

```
8. ?>
```

6) Задачи на анализ кода PHP

Что будет показано в результате выполнения кода?

```
1. <?php
```

```
2. $k=3;
```

```
3. for ($i=0; $i<10; $i++)
```

```
4. {
```

```
5. $k++;
```

```
6. if ($i>5)
```

```
7. {  
8. $i++;  
9. $k++;  
10.  
}  
11.  
}  
12.  
echo $k;  
13.  
?>
```

Ответ: 13

Что будет показано в результате выполнения кода?

```
1. <?php  
2. $k=6;  
3. for ($i=1; $i<10; $i+=2)  
4. {  
5. $k-;  
6. if ($i>6)  
7. {  
8. $i-;  
9. $k++;  
10.  
}  
11.  
}  
12.  
echo $k;  
13.  
?>
```

Ответ: 3

Что будет показано в результате выполнения кода?

```
1. <php  
2. $k=3;  
3. $s=0;  
4. for ($i=10; $i>1; $i-)  
5. {  
6. $k++;  
7. if ($i % 3 == 1)  
8. {  
9. $k-=2;  
10.  
}  
11.  
$s+=$k;  
12.  
}  
13.  
echo $s;  
14.  
?>
```

Ответ: 36

Что будет показано в результате выполнения кода?

```
1. <php  
2. $p=1;
```

3. for (\$i=0; \$i<10; \$i++)

4. {

5. \$p++;

6. if (\$i>5)

7. {

8. \$i-;

9. echo \$p;

10.

}

11.

if (\$p>100)

12.

{

13.

break;

14.

}

15.

}

16.

?>

Ответ: 8910...101

Что будет показано в результате выполнения кода?

1. <php

2. \$a=array();

3. for (\$i=0; \$i<5; \$i++)

4. {

5. \$a[]=\$i+1;

6. }

7. echo array_sum(\$a);

8. ?>

Ответ: 15

Что будет показано в результате выполнения кода?

1. <?php

2. \$i=1;

3. while(\$i<=5)

4. {

5. if(\$i%2==1)

6. {

7. echo \$i.'
';

8. }

9. else

10.

{

11. echo \$i*\$i.'
';

12.

}

13. \$i++;

14.

}

15.

?>

Ответ:

1

4

3

16

5

Что будет показано в результате выполнения кода?

1. `<?php`

2. `$i=1;`

3. `while($i<=4 || $i!=6)`

4. `{`

5. `echo $i.
;`

6. `$i++;`

7. `}`

8. `?>`

Ответ:

1

2

3

4

5

Что будет показано в результате выполнения кода?

1. `<?php`

2. `for($i=0;$i<=5;$i++)`

3. `{`

4. `echo ($i+5).
;`

5. `if($i==0)`

6. `{`

7. `$i++;`

8. `}`

9. `}`

10. `?>`

Ответ:

5

7

8

9

10

Что будет показано в результате выполнения кода?

1. `<?php`

2. `$s=0;`

3. `for($i=0;$i<=5;$i++)`

4. `{`

5. `$s+=$i;`

6. `if($i==3)`

7. `{`

8. `$s+=100;`

9. `}`

10.

`}`

11. `echo $s;`

12.

`?>`

Ответ: 115

7) Файлы в PHP

7.1 Работа с файлами

Создайте файл 'test.txt' и запишите в него фразу 'Привет, мир!'.

1. `<?php`

2. file_put_contents('test.txt', 'Привет, мир!');

3. ?>

Считайте данные из файла 'test.txt' и выведите их на экран.

1. <?php

2. echo file_get_contents('test.txt');

3. ?>

Переименуйте файл 'test.txt' в 'mir.txt'.

1. <?php

2. echo rename('test.txt', 'mir.txt');

3. ?>

Сделайте папку 'folder'. 'test.txt' в 'mir.txt'.

1. <?php

2. echo rename('test.txt', 'mir.txt');

3. ?>

Создайте копию файла 'mir.txt' и назовите ее 'world.txt'.

1. <?php

2. copy(mir.txt, world.txt);

3. ?>

Определите размер файла 'world.txt'. Выведите его на экран. Выведите его в байтах, мегабайтах, гигабайтах.

1. <?php

2. echo filesize('world.txt'); //байт

3. echo filesize('world.txt')/1024/1024; //мегабайт

4. echo filesize('world.txt')/1024/1024/1024; //гигабайт

5. ?>

Удалите файл 'world.txt'.

1. <?php

2. unlink('world.txt');

3. ?>

Проверьте существование файлов 'world.txt' и 'mir.txt'.

1. <?php

2. if (file_exists('world.txt')) {

3. echo "Файл существует";

4. } else {

5. echo "Файл не существует";

6. }

7. ?>

Работа с папками, mkdir, rmdir

Создайте папку 'test'.

1. <?php

2. mkdir('test');

3. ?>

Переименуйте папку 'test' на 'www'.

1. <?php

2. rename('www');

3. ?>

Удалите папку 'www'.

1. <?php

2. rmdir('test','www');

3. ?>

Дан массив со строками. Создайте в папке 'test' папки, названиями которых служат элементы этого массива.

1. <?php

2. mkdir('test');

3. \$arr = ['qqq','eee','ddd'];

```
4. foreach($arr as $elem) {
5. mkdir('test/'.$elem);
6. }
7. ?>
```

На scandir, is_dir, is_file, PHP_EOL

Выведите на экран название всех файлов и подпапок из папки 'test'.

```
1. <?php
2. var_dump(scandir('test'));
3. ?>
```

Выведите на экран название всех файлов, но не подпапок из папки 'test'.

```
1. <?php
2. $arr = [];
3. $dir = scandir('test');
4. foreach($dir as $elem) {
5. if(is_file('test/'.$elem)) $arr []= $elem;
6. }
7. var_dump($arr);
8. ?>
```

В папке 'test' есть файлы и подпапки. Выведите на экран содержимое всех файлов, которые лежат непосредственно в папке 'test'.

```
1. <?php
2. $arr = [];
3. $dir = scandir('test');
4. foreach($dir as $elem) {
5. if(is_file('test/'.$elem)) $arr []= $elem;
6. }
7. var_dump($arr);
8. ?>
```

Выведите на экран название всех файлов с расширением txt из папки 'test'.

```
1. <?php
2. $arr = [];
3. $dir = scandir('test');
4. foreach($dir as $elem) {
5. if(is_file('test/'.$elem) and preg_match('#.txt$#',$elem)) $arr []= $elem;
6. }
7. var_dump($arr);
8. ?>
```

Найдите все файлы из папки 'test' и вставьте в начало каждого файла полный путь к нему (текст файла должен остаться в нем и начинаться с новой строки после пути).

```
1. <?php
2. $arr = [];
3. $dir = scandir('test');
4. foreach($dir as $elem) {
5. if(is_file('test/'.$elem) and preg_match('#.txt$#',$elem)) {
6. $text_file = file_get_contents('test/'.$elem);
7. file_put_contents('test/'.$elem, 'test/'.$elem.PHP_EOL.$text_file);
8. }
9. }
10.
?>
```

Выведите на экран имена всех папок из папки 'test' и их подпапок (может быть любой уровень вложенности).

```
1. <?php
2. $arr = [];
3. function show_dir($dir_name)
```

```

4. {
5. $dir = scandir($dir_name);
6. unset( $dir[0], $dir[1] );
7. foreach($dir as $elem) {
8. $str = $dir_name;
9. if(is_dir($dir_name.'/'.$elem)) {
10. $str .= '.'.$elem;
11. $arr [$elem]= show_dir($str);
12.
13.
14. }
15. }
16. return $arr;
17.
18. }
19. var_dump(show_dir('test'));
20.
21. ?>

```

Выведите на экран содержимое всех файлов из папки 'test' и ее подпапок (может быть любой уровень вложенности).

```

22. <?php
23. $arr = [];
24. function show_dir($dir_name)
25. {
26. $dir = scandir($dir_name);
27. unset($dir[0], $dir[1]);
28. foreach($dir as $elem) {
29. $str = $dir_name;
30. if(is_file($dir_name.'/'.$elem)) {
31. $arr [$elem]= file_get_contents($dir_name.'/'.$elem);
32.
33. }
34. if(is_dir($dir_name.'/'.$elem)) {
35. $str .= '.'.$elem;
36. $arr [$elem]= show_dir($str);
37.
38. }
39. }
40. return $arr;
41.
42. }
43. var_dump(show_dir('test'));
44.
45. ?>

```

Найдите все файлы из папки 'test' и ее подпапок любого уровня вложенности и вставьте в начало каждого файла полный путь к нему (текст файла должен остаться в нем и начинаться с новой строки после пути).

```

46. <?php
47. $arr = [];
48. function show_dir($dir_name)
49. {
50. $dir = scandir($dir_name);
51. unset($dir[0], $dir[1]);
52. foreach($dir as $elem) {

```

```

8. $str = $dir_name;
9. if(is_file($dir_name.'/'.$selem)) {
10. $text_file = file_get_contents($dir_name.'/'.$selem);
11. file_put_contents($dir_name.'/'.$selem, 'tesst/'.$selem.PHP_EOL.$text_file);
12. $arr [$selem]= file_get_contents($dir_name.'/'.$selem);
13.
14. }
14. if(is_dir($dir_name.'/'.$selem)) {
15. $str .=.'/'.$selem;
16. $arr [$selem]= show_dir($str);
17.
18. }
19. return $arr;
20.
21. }
21. var_dump(show_dir('test'));

```

22.

?>

Найдите все файлы из папки 'test', в их содержимом найдите тег <h1>текст</h1>. Переименуйте все файлы на их h1.

1. <?php

```

2. $dir = scandir('test');
3. foreach($dir as $selem) {
4. if(is_file('test/'.$selem)) {
5.     preg_match_all('#<h1>(.*?)</h1>#', file_get_contents('test/'.$selem), $arr);
6.     rename('test/'.$selem, 'test/'.$arr[1][0].'.txt');
7. }
8. }

```

7. }

8. }

9. ?>

Удалите из папки 'test' все файлы размером более 1мб.

1. <?php

```

2. $dir = scandir('test');
3. foreach($dir as $selem) {
4. if(is_file('test/'.$selem) and filesize('test/'.$selem)/1024/1024 > 1) {
5. unlink('test/'.$selem);
6. }
7. }

```

7. }

8. ?>

Имеется папка с файлами, узнайте размер этой папки.

1. <?php

```

2. $dir = scandir('test');
3. $sum = 0;
4. foreach($dir as $selem) {
5. if(is_file('test/'.$selem)) {
6. $sum += round(filesize('test/'.$selem)/1024/1024);
7. }
8. }
9. echo $sum.' мб';

```

10.

?>

Имеется папка с подпапками, узнайте размеры всех подпапок папки и выведите их на экран.

1. <?php

```

2. $sum = 0;
3. $arr = [];
4. function show_dir($dir_name)
5. {
6. $dir = scandir($dir_name);
7. unset($dir[0], $dir[1]);
8. foreach($dir as $elem) {
9. $str = $dir_name;
10. if(is_file($dir_name.'/'.$elem)) {
11. $sum += round(filesize($dir_name.'/'.$elem)/1024/1024);
12.
13. }
14. if(is_dir($dir_name.'/'.$elem)) {
15. $str .=.'/'.$elem;
16. $arr [$elem]= show_dir($str);
17.
18. }
19. }
20. return $sum;
21.
?>

```

8) Задачи на Cookie

Задание 1 : Инициализируйте переменную для подсчета количества посещений. Если соответствующие данные Передавались через cookie сохраняйте их в эту переменную. Нарастите счетчик посещений. Инициализируйте переменную для хранения значения последнего посещения страницы. Если соответствующие данные передавались из cookie , отфильтруйте их и сохраните в эту переменную. Установите соответствующие cookie

Задание 2: Выводите информацию о количестве посещений и дате последнего посещения

1. <?php

```

2. //инициализируем переменную для подрасчета
3. //количества посещений
4. // Если соответствующие данные передавались через cookie
5. //сохраняем их в эту переменную
6. $visit_counter = 0;
7. if (isset($_COOKIE['visitCounter']) &&
8. is_numeric($_COOKIE['visitCounter'])) {
9. $visit_counter = $_COOKIE['visitCounter']*1;
10.
11. }
12. // Приращиваем счетчик посещений
13. $visit_counter++;
14. //Инициализируем переменную для хранения значения последнего посещения
15. // Если соответствующие данные передавались из cookie, сохраняем их в
    эту переменную
16. $last_visit = '';
17. if (isset($_COOKIE['lastVisit'])) {
18. $last_visit=htmlspecialchars($_COOKIE['lastVisit'],
19. ENT_QUOTES);

```

```

20. $last_visit=strripslashes(trim($last_visit));
21.
}
22. //устанавливаем cookie
23. setcookie ('visitCounter', $visit_counter, 0x7FFFFFFF);
24. setcookie ('lastVisit', date ('d/m/Y H:i:s'), 0x7FFFFFFF);
25. //Выводим информацию о количестве посещений и дате последнего
    посещения
26. if($visit_counter == 1) {
27. print '<h2>Добро пожаловать!</h2>';
28. } else {
29. print <<<HTML
30. <h2> Вы здесь уже $visit_counter раз</h2>
31. <p>Последнее сообщение: $last_visit</p>
32. HTML;
33.
}
34.
?>

```

9) Задачи на Session в PHP

Задача. Сделайте

две

страницы: **index.php** и **hello.php**.

При

заходе

на **index.php** спросите с помощью формы имя пользователя, запишите его в **куки**.

При заходе на **hello.php** поприветствуйте пользователя фразой "**Привет, %Имя%!"** .

Решение:

Страница index.php:

```

1. <form action="" method="GET">
2. <input type="text" name="username">
3. <input type="submit">
4. </form>

```

5.

6. <?php

7. //Если форма была отправлена и имя не пустое:

```

8. if (!empty($_REQUEST['username'])) {

```

```

9. //Пишем имя в куки:

```

```

10. setcookie('username', $_REQUEST['username'], time()+3600, '/');

```

```

11.

```

```

}

```

12.

?>

Страница hello.php:

1. <?php

2. //Если есть данные в куки об имени пользователя:

```

3. if (!empty($_COOKIE['username'])) {

```

```

4. echo $_COOKIE['username']; //выведем имя на экран

```

```

5. }

```

6. ?>

Спросите у пользователя телефон с помощью формы. Затем сделайте так, чтобы в **другой форме** (поля: имя, фамилия, телефон) при ее открытии поле телефон было автоматически заполнено.

Спрашиваем телефон:

```

1. <form action="" method="GET">

```

```

2. <input type="text" name="phone">
3. <input type="submit">
4. </form>
5.
6. <?php
7. //Если форма была отправлена и телефон не пустой:
8. if (!empty($_REQUEST['phone'])) {
9. //Пишем телефон в куки:
10. setcookie('phone', $_REQUEST['phone'], time()+3600, '/');
11.
12. }
13. ?>

```

Другая форма:

```

1. <?php
2. //Если телефон есть в куки – запишем его в переменную $phone:
3. if(!empty($_COOKIE['phone']))
4. $phone = $_COOKIE['phone'];
5. else
6. $phone = ' ';
7. ?>
8.
9. <form action="" method="GET">
10.
11. <input type="text" name="name">
12.
13. <input type="text" name="surname">
14.
15. <input type="text" name="phone" value=" <?php echo $phone ?> ">
16.
17. <input type="submit">
18. </form>

```

Задание 1 .

Создайте три страницы с формами, содержащими по одному полю ввода на стр. Обработчик каждой последующей формы будет вести на следующую страницу. На первой странице спросим имя, на второй странице фамилию, на третьей возраст, а на четвертой выведите все данные используя сессию.

Задание 2.

Создайте в сессии массив для хранения всех посещенных страниц и сохраните в качестве его очередного элемента путь к текущей странице.

Выведите в цикле список всех посещенных пользователями страниц

Page 1

```

1. <?php
2. // page 1
3. session_start(); //открываем сессию
4. //создаем в сессии массив для хранения
5. // всех посещенных страниц
6. //и сохраняем в качестве его очередного
7. //элемента текущую страницу

```

8. \$_SESSION['pages'][]=\$_SERVER['PHP_SELF'];

9. ?>

10.

```
<form action="page2.php" method="post">
```

11.

Ваше Имя:

12.

```
<input type="text" name="name"><br><br>
```

13.

```
<input type="submit" value="далее">
```

14.

```
</form>
```

Page 2

1. <?php

2. //страница 2

3.

4. session_start(); //открываем сессию

5. \$_SESSION['pages'][] = \$_SERVER['PHP_SELF'];

6. //если пользователь ввел имя, фильтруем его и заносим в сессию

7. if(isset(\$_POST[name])) {

8. \$_SESSION[name]=strip_tags(trim(\$_POST[name]));

9. }

10.

?>

11.

```
<form action="page3.php" method="post">
```

12.

Фамилия:

13.

```
<input type="text" name="Fname"> <br><br>
```

14.

```
<input type="submit" value="далее">
```

15.

```
</form>
```

Page 3 & 4

1. <?php

2. //page3.php

3.

4. session_start(); //открываем сессию

5. \$_SESSION['pages'][] = \$_SERVER['PHP_SELF'];

6. //если пользователь ввел имя, фильтруем его и заносим в сессию

7. if(isset(\$_POST[Fname])){

8. \$_SESSION[Fname]=strip_tags(trim(\$_POST[Fname]));

9. ?>

10.

11.

```
<form action="page4.php" method="post">
```

12.

Возраст:

13.

```
<input type="text" name="age"> <br><br>
```

14.

```
<input type="submit" value="далее">
```

15.

```

</form>
16.
17.
<?php
18.
// page 4
19.
session_start(); //открываем сессию
20.
$_SESSION ['pages'] [] = $_SERVER['PHP_SELF'];
21.
//если пользователь ввел имя, фильтруем его и заносим в сессию
22.
if(isset($_SESSION[name]) && isset ($_SESSION[Fname])
23.
$$ isset ($_SESSION[age]))
24.
{ echo "Your name $_SESSION[name] $_SESSION[Fname] <br>";
25.
echo "Ваш возраст $_SESSION[age] лет";
26.
}
27.
if(isset($_SESSION['pages']))
28.
{print '<h3> Список посещенных страниц</h3>';
29.
//вывод всех посещенных страниц в цикле
30.
print '<ol>';
31.
foreach ($_SESSION['pages'] as $page)
32.
{
33.
echo '<li>', $page, '</li>';
34.
}
35.
print '</ol>';
36.
}
37.
?>

```

10) MySQL и PHP задачи

Дан массив с таблицами ('user', 'profile'), именами полей (id, name и так далее) и их типами:

1. \$tables = array(
2. 'user'=>['id'=>'int', 'name'=>'varchar 256'],
3. 'profile'=>['id'=>'int', 'value'=>'text']
4.);

Отобразите этот массив в базу данных (создайте в базе данных таблицы с заданными именами и полями).

1. <?php

2. \$tables = array(
3. 'user'=>['id'=>'int', 'name'=>'varchar 256'],

```

4. 'profile'=>['id'=>'int', 'value'=>'text']
5. );
6. function tables ($tables) {
7. $connect = new mysqli('SRV','LOGIN','PSWD','DB');
8. foreach ($tables as $key=>$cols){
9. $connect->query("CREATE TABLE $key(ololo int (10))"); //Создание
10. foreach ($cols as $col=>$value){
11. $value=explode(' ', $value); //поиск размера
12. if(empty($value[1])) $value[1]=50;
13. $connect->query("ALTER TABLE $key ADD $col $value[0]($value[1]) NOT
NULL"); //Добавление столбцов
14.
}
15. $connect->query("ALTER TABLE $key DROP ololo"); //удаление временного
16.
}
17.
}
18. tables($tables);
19.
?>

```

Сформировать базу данных “учетные сведения об учащихся”, включающую след. поля:

Фамилия, Класс, Год поступления, Выполняемое поручение. Ввести в базу данных условную информацию о 5 учениках.

Написать программу просмотра базы данных и добавления в нее новой записи.

Итого мы получим три файла index.php , mysql.php, view.php

Index.php:

1. <?php

```

2. $send=$_POST['send'];
3. $fio=$_POST['fio'];
4. $klass=$_POST['klass'];
5. $year=$_POST['year'];
6. $poruchenie=$_POST['poruchenie'];
7.
8. $Name="Учетные сведения об учащихся";
9. // Коннектимся к базе
10. include ('mysql.php');
11. //Проверка заполнена ли форма удаления
12. if (isset($send))
13.
{
14.
$sql = "INSERT INTO $table_name (id`, `fio`, `klass`, `year`,
`poruchenie`) VALUES (NULL, '$fio', '$klass', '$year', '$poruchenie')"; 15.
$sq = mysql_query($sql,$conn) or die(mysql_error()); // отправляем
16.
}
17. echo "Задача №2. Сформировать базу данных 'Учетные сведения об учащихся',
включающую следующие поля: фамилия, класс, год поступления, выполняемое
поручение. Ввести в базу данных условную информацию о 5 учениках.
Написать программу просмотра базы данных и добавления в нее новой
записи.<br>";
18. include ('view.php');
19. //Форма

```

```

20. echo("<br><div align=left><form action='#' method='POST'><br> 21.
<input type='hidden' name='send' value=' ' size='10' /><br>
22. ФИО: <input type='text' name='fio' value=' ' size='10' /><br>
23. Класс: <input type='text' name='klass' value=' ' size='10' /><br>
24. Год: <input type='text' name='year' value=' ' size='10' /><br>
25. Поручение: <input type='text' name='poruchenie' value=' ' size='10' /><br> 26. <input
type='submit' value='Добавить' />
27.
</div>
28. </form>");
29.
?>
mysql.php
1. <?php
2.
$conn=mysql_connect("localhost", "root", "");
3.
$database = "msn"; //База
4.
$table_name = "zadacha1_itogi";
5.
mysql_select_db($database);
6.
$list_f = mysql_list_fields($database,$table_name);
7.
$n1 = mysql_num_fields($list_f);
8. ?>
view.php
1. <?php
2. // сохраним имена полей в массиве $names
3. for($j=0;$j<$n1; $j++){
4. $names[] = mysql_field_name ($list_f,$j);
5. }
6. $sql = "SELECT * FROM $table_name"; // создаем SQL запрос
7. $q = mysql_query($sql,$conn) or die(); // отправляем
8. // запрос на сервер
9. $n = mysql_num_rows($q); // получаем число строк результата
10.
//рисует HTML-таблицу
11.
12.
echo " <TABLE BORDER=0 CELLSPACING=0 width=90%
13.
align=center><tr><TD BGCOLOR='#003D57' align=center>
14.
<font color='#FFFFFF'><b> $Name</b></font></td>
15.
</tr></TABLE>";
16.
echo "<table cellpadding=1 border=1
17.
width=90% align=center>";
18.
// отображаем названия полей

```

```

19.
echo "<tr>";
20.
foreach ($names as $val){
21.
echo "<th ALIGN=CENTER BGCOLOR='#4D829E'>
22.
<font color='#FFFFFF' size=2> $val</font></th>";
23.
}
24.
// отображаем значения полей
25.
echo "</tr>";
26.
for($i=0;$i<$n; $i++){ // перебираем все строки в
27.
// результате запроса на выборку
28.
echo "<tr>";
29.
foreach ($names as $k => $val) { // перебираем все
30.
// имена полей
31.
$value = mysql_result($q,$i,$val); // получаем
32.
// значение поля
33.
echo "<td><font size=2> $value</font></td>"; 34.
// выводим значение поля
35.
}
36.
echo "</tr>";
37.
}
38.
echo "</table>";
39.
?>

```

11) Занимательные задачи*

Написать функцию, которая преобразует строки вида `var_php_test` в `varPhpTest`.

1. <?php

```

2. function camelCase($ext){
3. $start = microtime(true);
4.
5. $result = str_replace('_', '', $ext);
6. $result = ucwords($result);
7. $result2 = str_replace(' ', ' ', $result);
8. $result2 = lcfirst($result2);
9.
10. $end = microtime(true);
11. $res = $end - $start;

```

12. echo "
Время выполнения php скрипта: ". 1000*\$res . " миллисекунд "; 13.

14. return \$result2;

15.

?>

Числа Фибоначи – это ряд чисел, в котором каждое последующее число равно сумме двух предыдущих чисел.

Выведите на экран первые 100 чисел.

1. <?php

2. \$a = 0;

3. \$b = 1;

4. echo \$a."
".\$b."
";

5. for (\$i=0; \$i<=98; \$i++) {

6. \$c = \$a+\$b;

7. echo \$c."
";

8. \$a = \$b;

9. \$b = \$c;

10.

}

11.

?>

Обрежьте длинную строку так, чтобы ее длина была не более N символов, при этом слова не должны быть разорваны – обрезание всегда проходит по пробелу и новая строка всегда меньше N.

Пример: \$n = 8; \$str = 'abcde abc abc abcde' =>

'abcde
abc abc
abcde';

1. <?php

2. \$var = 'He lo Wor ld';

3. \$strNotGap = substr(\$var, 0, 5);

4. \$strNotGapRev = strrev(\$strNotGap);

5. preg_match('#\S*?\s?(.+)#', \$strNotGapRev, \$res);

6. \$strGapRev = \$res[1];

7. \$result = strrev(\$strGapRev);

8. var_dump(\$result);

9. ?>

Реализуйте треугольник Паскаля произвольного размера (должен нарисоваться в браузере).

1. <?php

2.

3. **function** printArray(\$array, \$tab)

4. {

5. \$result = "";

6. for(\$i=1;\$i<=\$tab;\$i++)

7. \$result.=".";

8.

9. for(\$i=0;\$i<count(\$array);\$i++)

10.

\$result.=\$array[\$i]." ";

11.

return \$result."
";

12.

}

13.

14.

echo "Треугольник Паскаля

";

```
15.
16.
$tab = 15;
17.
$tri = array(1,array(1,1));
18.
19.
echo "....." . $tri[0]. "<br>";
20.
21.
for($i=1;$i<15;$i++)
22.
{
23.
//Взятие последнего элемента
24.
$last = $tri[count($tri)-1];
25.
26.
//Вывод последнего элемента
27.
echo printArray($last,$tab);
28.
29.
//Инициализация пустого массива
30.
$added = array();
31.
32.
/*
33.
Инициализация элементов
34.
*/
35.
array_push($added,1);
36.
for($j=0;$j<count($last)-1;$j++)
37.
array_push($added, $last[$j]+$last[$j+1]);
38.
array_push($added,1);
39.
40.
//Добавление сформированного массива
41.
array_push($tri,$added);
42.
```

43.

//Уменьшение отступа

44.

\$tab-;

```
Треугольник Паскаля

.....1
.....1 1
.....1 2 1
.....1 3 3 1
.....1 4 6 4 1
.....1 5 10 10 5 1
.....1 6 15 20 15 6 1
.....1 7 21 35 35 21 7 1
.....1 8 28 56 70 56 28 8 1
.....1 9 36 84 126 126 84 36 9 1
.....1 10 45 120 210 252 210 120 45 10 1
.....1 11 55 165 330 462 462 330 165 55 11 1
....1 12 66 220 495 792 924 792 495 220 66 12 1
...1 13 78 286 715 1287 1716 1716 1287 715 286 78 13 1
..1 14 91 364 1001 2002 3003 3432 3003 2002 1001 364 91 14 1
```

45.

}

46.

?>

Напишите функцию, которая определяет пол по ФИО.

1. <?php

2. **function** gender(\$fio){if(\$fio){return substr(\$fio, -2)=='ич'?'male':'female';}}

3. ?>

Сделайте функцию, которое будет принимать 2 числа, а возвращать их НОК - наименьшее общее кратное. НОК двух чисел – это самое маленькое число, которое делится и на одно, и на второе число. Пример: числа 12 и 15 имеют НОК 60. Число 60 делится и на 12, и на 15 и это самое минимальное такое число.

1. <?php

2. **function**(x,y) {

3. **var** x = ~~x;

4. **var** y = ~~y;

5. **var** i = 1;

6. **while** ((i % x == 0) && (i % y == 0)) { i++}

7. **return** i;

8. }

9. ?>

Напишите функцию, которая находит разницу между двумя датами в годах, месяцах, днях, минутах, часах, секундах.

Функция принимает две даты в формате 2017-12-31T12:59:59, а возвращает ассоциативный массив с ключами (год, месяц ... секунда).

1. <?php

2. **function** diffdate(\$d1, \$d2){

3. \$dsp = preg_split('#[-T:]#', \$d2);

4. \$diff = '-'.\$dsp[0].'years, -'.\$dsp[1].'months, -'.\$dsp[2].'days, -'.\$dsp[3].'hours, -'.\$dsp[4].'minutes, -'.\$dsp[5].'seconds';

5. \$date = date_create(\$d1);

```

6. date_modify($date, $diff);
7. return date_format($date, 'Y-m-d\TH:i:s');
8. }
9. var_dump(diffdate('2016-12-30T16:12:59', '2015-01-29T15:11:04'));

```

10.

?>

Реализуйте функцию `flatten`, которая в случае, если массив обладает уровнями вложенности, приведёт его к элементарному виду (вложенность может быть любой глубины).

Пример: `flatten([1, [2], [3, [[[4]]]])` вернёт `[1, 2, 3, 4]`.

1. <?php

```

2. function array_flatten($array) {
3.     $result = [];
4.     array_walk_recursive($array, function($element) use (&$result) {
5.         $result[] = $element;
6.     });
7.     return $result;
8. }
9. print_r(array_flatten([1, [2], [3, [[[4]]]]]));

```

10.

?>

Дана папка с файлами и подпапками, в подпапках могут быть свои подпапки и файлы и так далее. Выведите на экран пути ко всем файлам, расположенных в этих папках.

Пример: папка `root`, в ней файл `test.php` и папка `folder`, значит путь к файлу `test` будет такой: `/root/test.php`. Если в `folder` есть файл `subtest.php`, то путь к нему будет `/root/folder/subtest.php`.

Папки могут иметь любой уровень вложенности.

1. <?php

```

2. $path = '.';
3. $files = new RecursiveIteratorIterator(new RecursiveDirectoryIterator($path,
FilesystemIterator::SKIP_DOTS), RecursiveIteratorIterator::SELF_FIRST);
4. foreach ($files as $name => $file){
5.     echo $name, PHP_EOL;
6. }

```

7. ?>

Дана папка с картинками. При обновлении страницы выведите картинку случайным образом. Решите две задачи:

1. Картинки могут повторяться.

2. Показанная картинка не будет повторяться пока не будут показаны все картинки из папки. То есть все картинки из папки нужно показать только 1 раз, но случайным образом. Когда все будут показаны – начинаем заново: показываем по второму кругу, но только 1 раз.

1. <?php

```

2. //Получаем список картинок
3. $dir="img";
4. $images = scandir($dir);
5. //убираем "." и ".."
6. unset ($images[0]);
7. unset ($images[1]);
8. $images = array_values($images);
9. $i=rand(0,(count($images)-1));
10.
11. class ImgDB {
12.     const DB_NAME = "rotator.db";

```

```

13. private $_db;
14. function __construct(array $data){
15. //если база еще не создана – создаем и соединяемся, иначе – соединяемся с
    базой
16. if(is_file(self::DB_NAME)){
17. $this->_db = new SQLite3(self::DB_NAME);
18. }else{
19. $this->_db = new SQLite3(self::DB_NAME);
20. $sql = 'CREATE TABLE imgs(id INTEGER PRIMARY KEY
    AUTOINCREMENT,filename TEXT)';
21. $this->_db->exec($sql) or die($this->_db->lastErrorMsg());
22. foreach ($data as $k=>$v){
23. $sql = "INSERT INTO imgs(filename) VALUES (' $v')";
24. $this->_db->exec($sql) or die($this->_db->lastErrorMsg());
25.
    }
26.
    }
27.
    }
28. public function showImgOnce($data){
29. $sql = 'SELECT * FROM imgs ORDER BY RANDOM() LIMIT 1;';
30. $res=$this->_db->query($sql) or die($this->_db->lastErrorMsg());
31. $res = $this->db2arr($res);
32. $id = $res[0]['id'];
33. $sql = "select count(*) from imgs;";
34. //Если в таблице последняя запись – перезаписываем
35. if (($this->_db->query($sql))==1){
36. foreach ($data as $k=>$v){
37. $sql = "INSERT INTO imgs(filename) VALUES (' $v')";
38. $this->_db->exec($sql) or die($this->_db->lastErrorMsg());
39.
    }
40.
    }
41. //Удаляем запись о показанной картинке
42. $sql = "DELETE FROM imgs WHERE id=$id";
43. $del=$this->_db->exec($sql) or die($this->_db->lastErrorMsg());
44. return $res[0]["filename"];
45.
    }
46. protected function db2arr($res){
47. $arr = array();
48. while($row=$res->fetchArray(SQLITE3_ASSOC))
49. $arr[] = $row;
50. return $arr;
51.
    }
52.
    }
53. $imgdata = new ImgDB($images);
54. echo ' '; 55.
?>

```

Document Outline

- 1) Вычисления и условный оператор в PHP
 - 1.1 Простейшая арифметика
 - 1.2 Условный оператор
 - 1.3 Работа с формой
- 2) Массивы в PHP
 - 2.1 Массивы
 - 2.2 Ассоциативные массивы
 - 2.3 Многомерные массивы
- 3) Задачи на if else, switch-case
- 4) Циклы foreach, while, for
- 5) Строки в PHP
 - 5.1 Работа с регистром символов
Работа с strlen
Работа с substr
Работа с str_replace
Работа с strpos
Работа с substr_replace
Работа с strpos, strrpos
Работа с explode, implode
Работа с str_split
Работа с trim, ltrim, rtrim
Работа с strrev
Работа с str_shuffle
Работа с number_format
Работа с str_repeat
Работа с strip_tags и htmlspecialchars
Работа с chr и ord
Работа с strchr, strrchr
 - 5.3 Применение строк в PHP
- 6) Задачи на анализ кода PHP
- 7) Файлы в PHP
 - 7.1 Работа с файлами
- 8) Задачи на Cookie
- 9) Задачи на Session в PHP
- 10) MySQL и PHP задачи
- 11) Занимательные задачи*