

# Самостоятельная работа 3-4

## Примеры решения задач

### Задача 1

**Задача.** Создайте массив, заполненный числами от 1 до 100. Найдите **сумму** элементов данного массива.

**Решение:** для начала создадим массив с числами от 1 до 100. Вручную это сделать будет очень сложно, поэтому воспользуемся функцией range:

```
<?php
    $arr = range(1, 100);

?>
```

**Сумму элементов** полученного массива найдем с помощью функции array\_sum (можно использовать цикл **foreach**, как мы это делали [раньше](#), но **array\_sum** в данном случае гораздо удобнее и проще):

```
<?php
    //Найдем сумму элементов нашего массива:
    echo array_sum(range(1, 100));

?>
```

### Задача 2. Функция array\_map

**Задача.** Дан массив с элементами 'a', 'b', 'c', 'd', 'e'. С помощью функции **array\_sum** сделайте из него массив 'A', 'B', 'C', 'D', 'E'.

**Решение:** с помощью функции **array\_map** задача решается в одну строчку - первым параметром передадим ей функцию **strtoupper**, а вторым - массив, к каждому элементу которого мы хотим применить **strtoupper**:

```
<?php
    $arr = ['a', 'b', 'c', 'd', 'e'];
    $arr = array_map('strtoupper', $arr);
    var_dump($arr);

?>
```

## Задачи для решения

### Работа с count

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

1. Дан массив **\$arr**. Подсчитайте количество элементов этого массива.
2. Дан массив **\$arr**. С помощью функции **count** выведите последний элемент данного массива.

## Работа с **in\_array**

Для решения задач данного блока вам понадобятся следующие функции: **in\_array**.

3. Дан массив с числами. Проверьте, что в нем есть элемент со значением **3**.

## Работа с **array\_sum** и **array\_product**

Для решения задач данного блока вам понадобятся следующие функции: **array\_sum**, **array\_product**.

4. Дан массив **[1, 2, 3, 4, 5]**. Найдите сумму элементов данного массива.
5. Дан массив **[1, 2, 3, 4, 5]**. Найдите произведение (умножение) элементов данного массива.
6. Дан массив **\$arr**. С помощью функций **array\_sum** и **count** найдите среднее арифметическое элементов (сумма элементов делить на их количество) данного массива.

## Работа с **range**

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

7. Создайте массив, заполненный числами от **1** до **100**.
8. Создайте массив, заполненный буквами от **'a'** до **'z'**.
9. Создайте строку **'1-2-3-4-5-6-7-8-9'** не используя цикл.
10. Найдите сумму чисел от **1** до **100** не используя цикл.
11. Найдите произведение чисел от **1** до **10** не используя цикл.

## Работа с **array\_merge**

Для решения задач данного блока вам понадобятся следующие функции: **array\_merge**.

12. Даны два массива: первый с элементами **1, 2, 3**, второй с элементами **'a', 'b', 'c'**. Сделайте из них массив с элементами **1, 2, 3, 'a', 'b', 'c'**.

## Работа с **array\_slice**

Для решения задач данного блока вам понадобятся следующие функции: **array\_slice**.

13. Дан массив с элементами **1, 2, 3, 4, 5**. С помощью функции **array\_slice** создайте из него массив **\$result** с элементами **2, 3, 4**.

## Работа с **array\_splice**

Для решения задач данного блока вам понадобятся следующие функции: **array\_splice**.

14. Дан массив [1, 2, 3, 4, 5]. С помощью функции **array\_splice** преобразуйте массив в [1, 4, 5].

15. Дан массив [1, 2, 3, 4, 5]. С помощью функции **array\_splice** запишите в новый массив элементы [2, 3, 4].

16. Дан массив [1, 2, 3, 4, 5]. С помощью функции **array\_splice** сделайте из него массив [1, 2, 3, 'a', 'b', 'c', 4, 5].

17. Дан массив [1, 2, 3, 4, 5]. С помощью функции **array\_splice** сделайте из него массив [1, 'a', 'b', 2, 3, 4, 'c', 5, 'e'].

## Работа с array\_keys, array\_values, array\_combine

*Для решения задач данного блока вам понадобятся следующие функции: array\_keys, array\_values, array\_combine.*

18. Дан массив 'a'=>1, 'b'=>2, 'c'=>3. Запишите в массив **\$keys** ключи из этого массива, а в **\$values** — значения.

19. Даны два массива: ['a', 'b', 'c'] и [1, 2, 3]. Создайте с их помощью массив 'a'=>1, 'b'=>2, 'c'=>3'.

## Работа с array\_flip, array\_reverse

*Для решения задач данного блока вам понадобятся следующие функции: array\_flip, array\_reverse.*

20. Дан массив 'a'=>1, 'b'=>2, 'c'=>3. Поменяйте в нем местами ключи и значения.

21. Дан массив с элементами 1, 2, 3, 4, 5. Сделайте из него массив с элементами 5, 4, 3, 2, 1.

## Работа с array\_search

*Для решения задач данного блока вам понадобятся следующие функции: array\_search*

22. Дан массив ['a', '-', 'b', '-', 'c', '-', 'd']. Найдите позицию первого элемента '-'.

23. Дан массив ['a', '-', 'b', '-', 'c', '-', 'd']. Найдите позицию первого элемента '-' и удалите его с помощью функции **array\_splice**.

## Работа с array\_replace

*Для решения задач данного блока вам понадобятся следующие функции: array\_replace.*

24. Дан массив ['a', 'b', 'c', 'd', 'e']. Поменяйте элемент с ключом 0 на '!', а элемент с ключом 3 - на '!!'.

## Работа с сортировку

*Для решения задач данного блока вам понадобятся следующие функции: [функции для сортировки](#),*

25. Дан массив '3'=>'a', '1'=>'c', '2'=>'e', '4'=>'b'. Попробуйте на нем различные типы сортировок.

## Работа с array\_rand

Для решения задач данного блока вам понадобятся следующие функции: **array\_rand**.

26. Дан массив с элементами 'a'=>1, 'b'=>2, 'c'=>3. Выведите на экран случайный **ключ** из данного массива.

27. Дан массив с элементами 'a'=>1, 'b'=>2, 'c'=>3. Выведите на экран случайный **элемент** данного массива.

## Работа с shuffle

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

28. Дан массив \$arr. Перемешайте его элементы в случайном порядке.

29. Заполните массив числами от 1 до 25 с помощью range, а затем перемешайте его элементы в случайном порядке.

30. Создайте массив, заполненный буквами от 'a' до 'z' так, чтобы буквы шли в случайном порядке и не повторялись.

31. Сделайте строку длиной 6 символов, состоящую из маленьких английских букв, расположенных в случайном порядке. Буквы не должны повторяться.

## Работа с array\_unique

Для решения задач данного блока вам понадобятся следующие функции: **array\_unique**.

32. Дан массив с элементами 'a', 'b', 'c', 'b', 'a'. Удалите из него повторяющиеся элементы.

## Работа с array\_shift, array\_pop, array\_unshift, array\_push

Для решения задач данного блока вам понадобятся следующие функции: **array\_shift**, **array\_pop**, **array\_unshift**, **array\_push**.

33. Дан массив с элементами 1, 2, 3, 4, 5. Выведите на экран его первый и последний элемент, причем так, чтобы в исходном массиве они исчезли.

34. Дан массив с элементами 1, 2, 3, 4, 5. Добавьте ему в начало элемент 0, а в конец - элемент 6.

35. Дан массив с элементами 1, 2, 3, 4, 5, 6, 7, 8. С помощью цикла и функций **array\_shift** и **array\_pop** выведите на экран его элементы в следующем порядке: 18273645.

## Работа с array\_pad, array\_fill, array\_fill\_keys, array\_chunk

Для решения задач данного блока вам понадобятся следующие функции: **array\_pad**, **array\_fill**, **array\_fill\_keys**, **array\_chunk**.

36. Дан массив с элементами 'a', 'b', 'c'. Сделайте из него массив с элементами 'a', 'b', 'c', '-', '-', '-'.

37. Заполните массив 10-ю буквами 'x'.

38. Создайте массив, заполненный целыми числами от 1 до 20. С помощью функции `array_chunk` разбейте этот массив на 5 подмассивов ([1, 2, 3, 4]; [5, 6, 7, 8] и т.д.).

## Работа с `array_count_values`

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

39. Дан массив с элементами 'a', 'b', 'c', 'b', 'a'. Подсчитайте сколько раз встречается каждая из букв.

## Работа с `array_map`

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

40. Дан массив с элементами 1, 2, 3, 4, 5. Создайте новый массив, в котором будут лежать квадратные корни данных элементов.

41. Дан массив с элементами '<b>php</b>', '<i>html</i>'. Создайте новый массив, в котором из элементов будут удалены теги.

42. Дан массив с элементами ' a ', ' b ', ' c '. Создайте новый массив, в котором будут данные элементы без концевых пробелов.

## Работа с `array_intersect`, `array_diff`

Для решения задач данного блока вам понадобятся следующие функции: `array_intersect`, `array_diff`.

43. Дан массив с элементами 1, 2, 3, 4, 5 и массив с элементами 3, 4, 5, 6, 7. Запишите в новый массив элементы, которые есть и в том, и в другом массиве.

44. Дан массив с элементами 1, 2, 3, 4, 5 и массив с элементами 3, 4, 5, 6, 7. Запишите в новый массив элементы, которые не присутствуют в обоих массивах одновременно.

## Задачи

45. Дана строка '1234567890'. Найдите сумму цифр из этой строки не используя цикл.

46. Создайте массив ['a'=>1, 'b'=2... 'z'=>26] не используя цикл.

47. Создайте массив вида [[1, 2, 3], [4, 5, 6], [7, 8, 9]] не используя цикл.

48. Дан массив с элементами 1, 2, 4, 5, 5. Найдите второй по величине элемент. В нашем случае это будет 4.