

# математические функции PHP

## Примеры решения задач

### Задача 1. Округление и ассоциативный массив

**Задача.** Найдите корень из числа **1000**. Округлите его в большую и меньшую стороны. В массив **\$arr** запишите первым элементом корень из числа, вторым элементом - округление в меньшую сторону, третьим элементом - в большую.

**Решение:** корень из числа найдем [функцией sqrt](#). Далее, чтобы округлить число в **меньшую** сторону, воспользуемся [функцией floor](#), а чтобы в большую - [функцией ceil](#):

```
<?php

    $sqrt = sqrt(1000); //найдем корень и запишем его в $sqrt

    echo floor($sqrt); //округлим в меньшую сторону
    echo ceil($sqrt);  //округлим в большую сторону

?>
```

Теперь результаты необходимо записать в массив. Сделать это можно **двумя** способами: объявить через [ ] либо просто воспользоваться присваиванием **\$arr[ ] = 'html'; \$arr[ ] = 'php';** и так далее.

**Первый способ:**

```
<?php

    $sqrt = sqrt(1000)
    $arr = [$sqrt, floor($sqrt), ceil($sqrt));

?>
```

**Второй способ:**

```
<?php

    $sqrt = sqrt(1000)
    $arr[] = $sqrt;
    $arr[] = floor($sqrt);
    $arr[] = ceil($sqrt);

?>
```

### Задача 2. Массив случайных чисел

**Задача.** Заполните массив 30-ю случайными числами от 1 до 10.

**Решение:** для решения воспользуемся циклом **for** - прокрутим его **30** раз, записывая при каждом проходе случайное число в **новый элемент массива**.

Случайные числа будем получать через функцию `mt_rand`.

Чтобы число записалось в новый элемент массива, следует сделать так: `$arr[ ] = 1;` `$arr[ ] = 2;` - первое число запишется в нулевой элемент массива, а второе - в первый (с ключом 1). В случае со случайными числами это будет выглядеть так:

```
<?php

    //Каждое из чисел будет записываться в новый элемент массива:
    $arr[] = mt_rand(1, 10);
    $arr[] = mt_rand(1, 10);
    $arr[] = mt_rand(1, 10);

?>
```

Напоминаю о том, что **ключи можно не оставлять пустыми**, а делать своими:

```
<?php

    //Используем свои ключи, а не автоматические:
    $arr['Первый ключ'] = mt_rand(1, 10);
    $arr['Второй ключ'] = mt_rand(1, 10);
    $arr['Третий ключ'] = mt_rand(1, 10);

?>
```

При большом желании мы можем **вручную** сделать 30 таких строчек - и задача решена. Но пусть **лучше** за нас это сделает цикл:

```
<?php

    //Переменная $i нужна, чтобы цикл сделал 30 итераций (проходов)
    for ($i = 1; $i <= 30; $i++) {
        $arr[] = mt_rand(1, 10);
    }

    var_dump($arr);

    /*
        Пообновляйте страницу -
        вы увидите как меняется массив,
        так как он заполняется случайными числами.
    */

?>
```

# Задачи для решения

## Работа с %

1. Даны переменные **\$a=10** и **\$b=3**. Найдите остаток от деления **\$a** на **\$b**.
2. Даны переменные **\$a** и **\$b**. Проверьте, что **\$a** делится без остатка на **\$b**. Если это так - выведите 'Делится' и результат деления, иначе выведите 'Делится с остатком' и остаток от деления.

## Работа со степенью и корнем

*Для решения задач данного блока вам понадобятся следующие функции: `sqrt`, `pow`.*

3. Возведите **2** в **10** степень. Результат запишите в переменную **\$st**.
4. Найдите квадратный корень из **245**.
5. Дан массив с элементами **4, 2, 5, 19, 13, 0, 10**. Найдите корень из суммы квадратов его элементов. Для решения воспользуйтесь циклом **foreach**.

## Работа с функциями округления

*Для решения задач данного блока вам понадобятся следующие функции: `round`, `ceil`, `floor`.*

6. Найдите квадратный корень из **379**. Результат округлите до целых, до десятых, до сотых.
7. Найдите квадратный корень из **587**. Округлите результат в большую и меньшую сторону, запишите результаты округления в ассоциативный массив с ключами **'floor'** и **'ceil'**.

## Работа с min и max

*Для решения задач данного блока вам понадобятся следующие функции: `min`, `max`.*

8. Даны числа **4, -2, 5, 19, -130, 0, 10**. Найдите минимальное и максимальное число.

## Работа с рандомом

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

9. Выведите на экран случайное число от **1** до **100**.
10. Заполните массив 10-ю случайными числами. Подсказка: нужно воспользоваться циклами **for** или **while**.

## Работа с модулем

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

11. Даны переменные **\$a** и **\$b**. Найдите модуль разности **\$a** и **\$b**. Проверьте работу скрипта самостоятельно для различных **\$a** и **\$b**.

12. Дан массив чисел, к примеру **[1, 2, -1, -2, 3, -3]**. Создайте из него новый массив так, чтобы отрицательные числа стали положительными, то есть у нас должен получиться такой массив: **[1, 2, 1, 2, 3, 3]**.

## Задачи

13. Дано число, например **30**. У этого числа есть делители - числа, на которое оно делится без остатка. Делители числа 30 - это 1, 2, 3, 5, 6, 10, 15, 30. Задача: сделайте массив делителей нашего числа. Число может быть любым, не обязательно, 30.

14. Дан массив **[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]**. Узнайте, сколько первых элементов массива нужно сложить, чтобы сумма получилась больше **10**.