

1.

Доступ по ссылке

- Вы унаследовали код от другого разработчика. Вы не можете изменить их код, но вам необходимо добавить некоторые дополнительные функции. Вместо использования переменной `$very_bad_unclear_name`, как в вашей части кода. Объявите новую переменную `$order` как ссылку на переменную `$very_bad_unclear_name`.
- С помощью оператора конкатенации добавьте к переменной `$order` любую строку.
- Распечатайте `$very_bad_unclear_name`

```
<?php
/* Imagine a lot of code here */

$very_bad_unclear_name = "15 chicken wings";

// Write your code here:

// Don't change the line below
echo "\nYour order is: $very_bad_unclear_name.";
```

2.

Числа

- Объявите переменную с любым именем и присвойте ей целочисленное значение. Используйте `echo`, чтобы вывести вашу переменную на терминал.
- Мы собираемся создать и напечатать еще одну числовую переменную, но мы не хотим, чтобы она печаталась в той же строке. Используйте `echo`, чтобы напечатать строку `"\n"`.

- Затем объявите переменную с любым именем и присвойте ей значение с плавающей запятой. Используйте `echo`, чтобы вывести эту новую переменную на терминал.
- Используйте `echo`, чтобы напечатать число 12 на терминале. Хитрость: используйте сложение или вычитание, чтобы напечатать оператор, который оценивается как 12, и число 12 не может появиться нигде в вашем коде!
- Насколько больше я потратил в прошлом месяце, чем в этом месяце? В прошлом месяце я потратил 1187,23, а в этом месяце я потратил 1089,98. Создайте переменные `$last_month` и `$this_month` и присвойте им соответствующие числовые значения. Используйте `echo` для вывода разницы между расходами в прошлом месяце и в этом месяце.

11.

Умножение и деление

- Мэг пытается выяснить, сколько времени в среднем ей требуется, чтобы выучить язык программирования. На данный момент она знает Ruby, Python, JavaScript и C++. Создайте переменную `$num_languages` и присвойте ей количество языков программирования, которые она выучила. Создайте вторую переменную, `$months`, и присвойте ей 11 — количество месяцев, которые она потратила на обучение кодированию.
- Мэг понимает, что не каждый день занимается. Создайте переменную `$days` и присвойте ей значение, полученное путем умножения `$months` на количество дней в месяце, которое, по ее мнению, она действительно провела за учебой, то есть 16 дней в месяц.
- Давайте подсчитаем, сколько дней в среднем у нее ушло на изучение каждого языка. Присвойте результат этой операции переменной `$days_per_language`.
- Распечатайте свою переменную `$days_per_language` в терминал.

12.

Степень

Используйте `echo` и оператор `**`, чтобы вывести на терминал значение 8 в квадрате.

13.

Операторы присвоения

- Создайте переменную `$my_num` и назначьте ей любое число.
- Затем создайте вторую переменную, `$answer`, и присвойте ей значение `$my_num`
- Используйте оператор присваивания сложения, чтобы добавить 2 к `$answer`
- Используйте оператор присваивания умножения, чтобы умножить `$answer` на 2.
- Используйте оператор присваивания вычитания, чтобы вычесть 2 из `$answer`.
- Используйте оператор присваивания деления, чтобы разделить `$answer` на 2
- Используйте оператор присваивания вычитания, чтобы вычесть исходное число (`$my_num`) из `$answer`. Наконец, используйте `echo` для вывода `$answer` на терминал.
- Если все прошло как надо, `$answer` должен быть равен 1. Каким бы ни был ваш первоначальный номер

14.

Математические функции

Работа с %

- Даны переменные `$a=10` и `$b=3`. Найдите остаток от деления `$a` на `$b`. Показать решение.
- 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`.

- Возведите 2 в 10 степень. Результат запишите в переменную `$st`.

- Найдите квадратный корень из 245.
- 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`.)

- Найдите квадратный корень из 379. Результат округлите до целых, до десятых, до сотых.
- 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`)

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

Работа с рандомом

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

Работа с модулем

- Даны переменные `$a` и `$b`. Найдите модуль разности `$a` и `$b`. Проверьте работу скрипта для различных `$a` и `$b`.
- Дан массив числами, 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]`.

Общее

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

15.

Функции

- Напишите функцию `printStringReturnNumber()`, которая печатает строку и возвращает числовое значение.
- Запишите возвращаемое значение вашей функции в переменную с именем `$my_num`.
- Распечатайте `$my_num`.

16.

Функции

- Напишите функцию `increaseEnthusiasm()`, которая принимает строковый параметр и возвращает эту строку, дополненную восклицательным знаком.
- Используйте `echo`, чтобы напечатать результат вызова вашей функции `increaseEnthusiasm()` со строкой по вашему выбору.
- Напишите функцию `repeatThreeTimes()`, которая принимает строковый параметр и возвращает эту строку, повторенную три раза (без добавления символов, которых не было в исходной строке)
- Используйте `echo`, чтобы напечатать результат вызова вашей функции `repeatThreeTimes()` со строкой по вашему выбору.
- Используйте `echo`, чтобы напечатать результат вызова функции `increaseEnthusiasm()` с результатом вызова `repeatThreeTimes()` в качестве аргумента, переданного в функцию `increaseEnthusiasm()`. Вы можете выбрать любую строку в качестве аргумента для `repeatThreeTimes()`
- Сделайте функцию `cut`, которая первым параметром будет принимать строку, а вторым параметром - сколько первых символов оставить в

этой строке. Второй параметр должен быть необязательным и по умолчанию принимать значение 10.

- Дан массив с числами. Выведите последовательно его элементы используя рекурсию и не используя цикл.
- Дано число. Сложите его цифры. Если сумма получилась более 9-ти, опять сложите его цифры. И так, пока сумма не станет однозначным числом (9 и менее)

17.

Массивы

- Заполните массив следующим образом: в первый элемент запишите 'x', во второй 'xx', в третий 'xxx' и так далее
- Сделайте функцию `arrayFill`, которая будет заполнять массив заданными значениями. Первым параметром функция принимает значение, которым заполнять массив, а вторым - сколько элементов должно быть в массиве. Пример: `arrayFill('x', 5)` сделает массив `['x', 'x', 'x', 'x', 'x']`.
- Дан двумерный массив с числами, например `[[1, 2, 3], [4, 5], [6]]`. Найдите сумму элементов этого массива. Массив, конечно же, может быть произвольным
- С помощью двух циклов создайте массив `[[1, 2, 3], [4, 5, 6], [7, 8, 9]]`
- Создайте массив с элементами 2, 5, 3, 9. Умножьте первый элемент массива на второй, а третий элемент на четвертый. Результаты сложите, присвойте переменной `$result`. Выведите на экран значение этой переменной.
- Создайте массив `$user` с ключами 'name', 'surname', 'patronymic' и какими-то произвольными значениями. Выведите на экран фамилию, имя и отчество через пробел.
- Создайте массив `$date` с ключами 'year', 'month' и 'day' и значениями, соответствующими текущему дню. Выведите созданную дату на экран в формате год-месяц-день.
- Пусть дан такой массив:

```
<?php
    $arr = ['a', 'b', 'c', 'd', 'e'];
?>
```

Выведите на экран количество элементов в этом массиве.

- Пусть дан такой массив:

```
<?php
    ○ $arr = ['a', 'b', 'c', 'd', 'e'];
?>
```

Выведите на экран последний элемент этого массива.

Выведите на экран предпоследний элемент этого массива.

18.

Конструкция if else

- Сделайте функцию, которая параметрами принимает 2 числа. Если их сумма больше 10 - пусть функция вернет true, а если нет - false.
- Сделайте функцию, которая параметрами принимает 2 числа. Если эти числа равны - пусть функция вернет true, а если не равны - false..
- Перепишите следующий код в сокращенной форме:

```
<?php
    if ($test == 0) {
        echo 'верно';
    }
?>
```

- Пусть в переменной \$age хранится число. Если это число меньше 10 или больше 99, то выведите на экран сообщение об этом. Если же число попадает в указанный диапазон, то найдите сумму цифр этого числа. Если полученная сумма меньше или равна 9, то выведите на экран сообщение о том, что сумма цифр однозначна, в противном случае выведите сообщение о том, что сумма цифр двузначна.

- В переменной \$arr содержится некоторый массив с числами. Напишите условие, которое проверит, что в массиве 3 элемента. Если это так, выведите на экран сумму элементов массива.

19.

Циклы

- Нарисуйте пирамиду, как показано на рисунке, только у вашей пирамиды должно быть 20 рядов, а не 5:

```
x
xx
xxx
xxxx
xxxxx
```

20.

Комбинация функций

- Дан массив с числами. Найдите среднее арифметическое его элементов (сумма элементов делить на количество) не используя цикл.
- Найдите сумму чисел от 1 до 100 не используя цикл
- Дан массив с числами. Получите из него массив с квадратными корнями этих чисел не используя цикл.
- Заполните массив числами от 1 до 26 так, чтобы ключами этих чисел были буквы английского алфавита: ['a'=>1, 'b'=>2...]. Сделайте это не используя цикл.
- Дана строка с числами '1234567890'. Найдите сумму пар чисел: 12+34+56+78+90. Решите задачу, не используя цикл