

Тестовое задание по PHP

1. Создайте массив \$arr, в котором ключи будут буквы, а значения - числа. Затем напишите функцию sum_string(array). В ней нужно вернуть строку из ключей, а так же сумму всех значений, умноженных на 13.

```
<?
$arr = ["a" => 12, "b" => 2, "c" => 4];
echo sum_string($arr);
// "abc 234"
```

2. Создайте массив \$days, где внутри ключей ru и en будут сокращенные названия дней (пн, вт, ... | mn, ts, ...). Создайте переменную \$lang и присвойте ей id языка (ru или en). Создайте переменную \$day, присвойте ей номер дня. А теперь выведите названия дня в заданном языке.

3. Создать функцию checkFunc(), в которую передавать один параметр. Если параметр не число - отдавать false, если число - проверить, больше ли оно 170, если больше - отдавать "Big", иначе - "Small".

4. Создать функцию HowManyBetween(), которая будет принимать два значения и будет отдавать разницу между большим и меньшим. Учесть, что могут быть не числа, отдавать ошибку с соответствующим текстом. Функция должна сама понимать какое из чисел меньшее.

5. Создать функцию с одним параметром - вес в тоннах. Параметр всегда число. Внутри функции определять сколько элементов можно создать из этого числа, если на один элемент необходимо 7 тонн. Результат вывести в виде "Создано x штук", если есть остаток - добавить к выводу "остаток - у грамм".

6. Создать форму, назвать "Поиск". Завести `input[type="text"]`. При отправке формы с непустым значением - выводить "Вы искали: text", где text - значение отправленного input.

7. Создайте форму "Авторизация". Сделайте `input[type="text"]`, `input[type="password"]`. Форма при отправке не должна добавлять данные в адресную строку. В скрипте завести переменные `$login`, `$password`. Если форма отправлена - проверять, прошел ли человек авторизацию по заданным в `$login` и `$password` значениям. Если вошел - задавать в сессию ключ "LOGIN" и значение "YES". Если этот ключ задан - вместо формы выводить "Привет, LOGIN!".

8. Создайте функцию `checkValue()`, которая принимает два значения - ассоциативный массив и строку. В ней проверять есть ли значение в массиве, если есть - прервать `foreach` и вернуть ключ, в котором записано это значение. Не использовать `in_array()`, только `foreach()`, только `hardcode`.

9. Создайте функцию `Pyramid()`, которая будет принимать одно значение - число строк. Она должна формировать пирамиду, как на рисунке.

```
0
11
222
3333
44444
```

10. Создать переменную `$num`. Создать функцию `minusNumber`, которая будет принимать эту переменную и менять её внутри функции. В себе она вычитает из него 1, если число больше 3 - вызывает себя. Использовать рекурсию. Вызов функции должен быть `minusNumber($num); echo $num;`, а не `$num = minusNumber($num);`

11. Создать форму с `input[type="file"]`. В неё загружать файл. Сохранять его рядом в папку `image` если он расширения `jpg` или `png`, иначе отдавать ошибку. Проверку сделать в виде функции `validateFile($name)`;

12. Создать класс `Product`. В нем объявить свойство `name` и `price`, а так же создать метод `createProduct()`. В нем принимаются два значения - название и цена. Так же создать метод `showPrice()`, который отдаст "Цена товара: `price`".

13. Создайте массив продуктов с параметрами `id`, `price`, `name`. Выведите каждый товар в виде "`id. name - price`". После каждого добавьте форму с кнопкой купить. Сделать обработку этой кнопки с добавлением товара в корзину. Сделать через сессию. При повторном нажатие, если товар уже в корзине, то его количество должно увеличиваться на 1. Снизу сделать вывод, что уже есть в корзине.