

Практическая работа

Задачи на функции работы со строками в PHP

Примеры решения задач

Задача 1

Задача. Дана строка **'minsk'**. Сделайте из нее строку **'MINSK'**.

Решение: воспользуемся функцией `strtoupper` и сразу получим результат:

```
<?php
    echo strtoupper('minsk');
?>
```

Задача 2

Задача. Дана строка **'минск'**. Сделайте из нее строку **'МИНСК'**.

Решение: функцией `strtoupper` мы не можем воспользоваться, так как она некорректно работает с русскими буквами. Воспользуемся функцией `mb_strtoupper` и сразу получим результат:

```
<?php
    echo mb_strtoupper('минск');
```

Задача 3

Задача. Дана строка **'MINSK'**. Сделайте из нее строку **'Minsk'**.

Решение: готовой функции для решения задачи в PHP не существует. Поэтому сначала воспользуемся функцией `strtolower` (в результате получится **'minsk'**), а затем функцией `ucfirst`:

```
<?php
    echo ucfirst(strtolower('minsk'));
```

Задача 4

Задача. В переменной **\$date** лежит дата в формате **'31-12-2030'**. Преобразуйте эту дату в формат **'2030.12.31'**.

Решение: для начала разобьем строку **'31-12-2030'** в массив с помощью функции `explode`:

```
<?php
```

```
//Разбиваем строку в массив по разделителю '-':  
$arr = explode('-', '31-12-2030');  
  
?>
```

В полученном массиве в **\$arr[0]** будет лежать **31** (то есть день), в **\$arr[1]** - месяц, а в **\$arr[2]** - год. Сольем элементы этого массива в новую строку:

```
<?php  
  
//Разбиваем строку в массив по разделителю '-':  
$arr = explode('-', '31-12-2030');  
  
//Получим дату в нужном формате:  
echo $arr[2].'. '.$arr[1].'. '.$arr[0];  
  
?>
```

Задачи для решения

Работа с регистром символов

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

1. Дана строка **'php'**. Сделайте из нее строку **'PHP'**.
2. Дана строка **'PHP'**. Сделайте из нее строку **'php'**.
3. Дана строка **'london'**. Сделайте из нее строку **'London'**.
4. Дана строка **'London'**. Сделайте из нее строку **'london'**.
5. Дана строка **'london is the capital of great britain'**. Сделайте из нее строку **'London Is The Capital Of Great Britain'**.
6. Дана строка **'LONDON'**. Сделайте из нее строку **'London'**.

Работа с strlen

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

7. Дана строка **'html css php'**. Найдите количество символов в этой строке.
8. 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.

Работа с substr

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

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

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

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

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

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

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

15. 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.

Работа с **str_replace**

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

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

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

18. Дана строка с буквами и цифрами, например, **'1a2b3c4b5d6e7f8g9h0'**. Удалите из нее все цифры. То есть в нашем случае должна получиться строка **'abcdbefgh'**.

Работа с **strtr**

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

19. 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).

Работа с **substr_replace**

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

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

Работа с **strpos**, **strrpos**

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

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

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

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

24. Дана строка 'aaa aaa aaa aaa aaa'. Определите позицию второго пробела.

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

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

Работа с explode, implode

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

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

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

29. В переменной \$date лежит дата в формате '2013-12-31'. Преобразуйте эту дату в формат '31.12.2013'.

Работа с str_split

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

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

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

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

Работа с trim, ltrim, rtrim

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

33. Дана строка. Очистите ее от концевых пробелов.

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

35. 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.

Работа с strrev

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

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

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

Работа с `str_shuffle`

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

38. Дана строка. Перемешайте символы этой строки в случайном порядке.

39. 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.

Работа с `number_format`

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

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

Работа с `str_repeat`

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

41. 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
```

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

```
1
22
333
4444
55555
666666
7777777
88888888
999999999
```

Работа с `strip_tags` и `htmlspecialchars`

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

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

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

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

Работа с `chr` и `ord`

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

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

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

48. Выведите на экран символ с кодом **33**.

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

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

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

Работа с `strchr`, `strrchr`

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

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

53. Дана строка `'ab-cd-ef'`. С помощью функции `strrchr` выведите на экран строку `'-ef'`.

Работа с `strstr`

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

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

Задачи

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

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