

Задачи на даты в PHP

Примеры решения задач

Задача

Задача. Выведите **23 сентября 2031 года, 12:58:59** в формате **timestamp**.

Решение: воспользуемся [функцией mktime](#) (сентябрь - 9-тый месяц):

```
<?php

echo mktime(12, 58, 59, 9, 23, 2031);

/*

    Напоминаю, что месяц и день идут в неправильном порядке:
    поэтому '9, 23,' а не '23, 9'.

*/

?>
```

Можно также воспользоваться [функцией strtotime](#), если представить нужную дату в формате **2031-09-23 12:58:59**:

```
<?php

echo strtotime('2031-09-23 12:58:59');

?>
```

Задача

Задача. Найдите разницу между **1 сентября 2010 года, 7:25:59** и текущим моментом времени в секундах.

Решение: текущий момент времени в формате **timestamp** получим с помощью функции **time**, а **timestamp** для заданной даты - с помощью функции **mktime**. Отнимем одно число от второго и получим искомую разницу:

```
<?php

echo time() - mktime(7, 25, 59, 9, 1, 2010);

?>
```

Задача . Функция date

Задача. Выведите текущую дату-время в формате **'2025.12.31 12:59:59'**.

Решение: воспользуемся функцией **date**, передав ей управляющие команды в таком порядке: год (команда **Y**), потом точку как символ, потом месяц (команда **m**), опять

точку, день (команда **d**), час (команда **H**), двоеточие, минуту (команда **i**), секунду (команда **s**). Получится такая строка: **'Y.m.d H:i:s'**. Подставим ее в функцию **date**:

```
<?php
    echo date('Y.m.d H:i:s');

?>
```

Задача . Функция date

Задача. Выведите 1-го сентября текущего года в формате **'2022.09.01'**.

Решение: для начала с помощью функции **mktime** преобразуем 1-го сентября текущего года в формат **timestamp**. Мы это делаем для того, чтобы подставить найденное число вторым параметром в функцию **date** (а первым параметром для **date** мы укажем формат вывода).

Так как требуется текущий год, то последний параметр (год) для **mktime** мы не указываем - тогда возьмется текущий год:

```
<?php

    //Выведем timestamp 1-го сентября текущего года:
    echo mktime(0, 0, 0, 9, 1);

?>
```

Ну, а теперь подставим найденный **timestamp** в функцию **date**:

```
<?php

    echo date('Y.m.d', mktime(0, 0, 0, 9, 1));

?>
```

Задача . Функция date. Вывод дня недели словом

Задача. Узнайте, какой день недели (словом) был 1 сентября 2010 года.

Решение: решение аналогично предыдущей задаче, только формат вывода для функции **date** мы сделаем в таком виде: **'w'**. В этом случае **date** вернет нам число, соответствующее дню недели за заданную дату (0 - воскресенье, 1 - понедельник и так далее):

```
<?php

    //День недели числом за нужную дату:
    echo date('w', mktime(0, 0, 0, 9, 1, 2010));

?>
```

Кстати, если бы мы хотели узнать, какой день недели сегодня - мы бы просто не передавали второй параметр функции **date** (тогда бы взялся текущий момент времени и, соответственно, вывелась бы **'w'** за текущий день).

Продолжим решать нашу задачу: мы вывели *номер* дня недели, а по задаче его следует вывести *словом*. Для этого составим массив дней недели **\$week** и с его помощью выведем то, что нам нужно. Вот этот массив:

```
<?php

    //Массив дней недели:
```

```
$week = ['вс', 'пн', 'вт', 'ср', 'чт', 'пт', 'сб'];
```

```
//Выведем с его помощью, к примеру, понедельник:
```

```
echo $week[1];
```

```
//А теперь вторник:
```

```
echo $week[2];
```

```
?>
```

Совместим теперь то, что вернет нам **date**, с нашим массивом **\$week**:

```
<?php
```

```
//День недели цифрой за нужную дату:
```

```
$day = date('w', mktime(0, 0, 0, 9, 1, 2010));
```

```
//Массив дней недели:
```

```
$week = ['вс', 'пн', 'вт', 'ср', 'чт', 'пт', 'сб'];
```

```
//День недели словом:
```

```
echo $week[$day];
```

```
?>
```

Задача. Преобразование форматов

Задача. Дана дата в формате '31-12-2025'. С помощью функций **mktime** и **explode** переведите эту дату в формат **timestamp**.

Решение: разобьем строку '31-12-2025' функцией **explode** в массив **\$arr**:

```
<?php
```

```
$arr = explode('31-12-2025'); //получим ['31', '12', '2025']
```

```
?>
```

В элементе **\$arr[0]** будет лежать день, в элементе **\$arr[1]** - месяц, в элементе **\$arr[2]** - год. Подставим эти данные в функцию **mktime** (напоминаю, что она принимает данные в формате '...месяц-день-год', не '...день-месяц-год')

```
<?php
```

```
$arr = explode('31-12-2025');
```

```
echo mktime(0, 0, 0, $arr[1], $arr[0], $arr[2]);
```

```
?>
```

Задачи для решения

Timestamp: time и mktime

Для решения задач данного блока вам понадобятся следующие функции: time, mktime.

1. Выведите текущее время в формате **timestamp**.
2. Выведите 1 марта 2025 года в формате **timestamp**.
3. Выведите 31 декабря текущего года в формате **timestamp**. Скрипт должен работать независимо от года, в котором он запущен.
4. Найдите количество секунд, прошедших с **13:12:59 15-го марта 2000 года** до настоящего момента времени.
5. Найдите количество **целых часов**, прошедших с **7:23:48 текущего дня** до настоящего момента времени.

Функция date

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

6. Выведите на экран текущий год, месяц, день, час, минуту, секунду.
7. Выведите текущую дату-время в форматах **'2025-12-31'**, **'31.12.2025'**, **'31.12.13'**, **'12:59:59'**.
8. С помощью функций mktime и date выведите 12 февраля 2025 года в формате **'12.02.2025'**.
9. Создайте массив дней недели **\$week**. Выведите на экран название текущего дня недели с помощью массива **\$week** и функции date. Узнайте какой день недели был **06.06.2006**, в ваш день рождения.
10. Создайте массив месяцев **\$month**. Выведите на экран название текущего месяца с помощью массива **\$month** и функции date.
11. Найдите количество дней в текущем месяце. Скрипт должен работать независимо от месяца, в котором он запущен.
12. Сделайте поле ввода, в которое пользователь вводит год (4 цифры), а скрипт определяет **високосный** ли год.
13. Сделайте форму, которая спрашивает дату в формате **'31.12.2025'**. С помощью функций mktime и explode переведите эту дату в формат **timestamp**. Узнайте день недели (словом) за введенную дату.
14. Сделайте форму, которая спрашивает дату в формате **'2025-12-31'**. С помощью функций mktime и explode переведите эту дату в формат **timestamp**. Узнайте месяц (словом) за введенную дату.

Сравнение дат

15. Сделайте форму, которая спрашивает две даты в формате **'2025-12-31'**. Первую дату запишите в переменную **\$date1**, а вторую в **\$date2**. Сравните, какая из введенных дат больше. Выведите ее на экран.

На strtotime

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

16. Дана дата в формате **'2025-12-31'**. С помощью функции `strtotime` и функции `date` преобразуйте ее в формат **'31-12-2025'**.

17. Сделайте форму, которая спрашивает дату-время в формате **'2025-12-31T12:13:59'**. С помощью функции `strtotime` и функции `date` преобразуйте ее в формат **'12:13:59 31.12.2025'**.

Прибавление и отнимание дат

Для решения задач данного блока вам понадобятся следующие функции: `date_create`, `date_modify`, `date_format`.

18. В переменной **\$date** лежит дата в формате **'2025-12-31'**. Прибавьте к этой дате 2 дня, 1 месяц и 3 дня, 1 год. Отнимите от этой даты 3 дня.

Задачи

19. Узнайте сколько дней осталось до Нового Года. Скрипт должен работать в любом году.

20. Сделайте форму с одним полем ввода, в которое пользователь вводит год. Найдите все пятницы 13-е в этом году. Результат выведите в виде массива дат.

21. Узнайте какой день недели был 100 дней назад.