

cal_info ถึงข้อมูลระบบปฏิทินที่ระบบใช้อยู่

ตัวอย่าง

```
<?php
$info = cal_info(0);
var_dump($info);
?>
```

ผลลัพธ์



C:\UniServerZ\www\phpdemo\datetime\cal_info.php:3:

array (size=5)

'months' =>

array (size=12)

- 1 => string 'January' (length=7)
- 2 => string 'February' (length=8)
- 3 => string 'March' (length=5)
- 4 => string 'April' (length=5)
- 5 => string 'May' (length=3)
- 6 => string 'June' (length=4)
- 7 => string 'July' (length=4)
- 8 => string 'August' (length=6)
- 9 => string 'September' (length=9)
- 10 => string 'October' (length=7)
- 11 => string 'November' (length=8)
- 12 => string 'December' (length=8)

'abbrevmonths' =>

array (size=12)

- 1 => string 'Jan' (length=3)
- 2 => string 'Feb' (length=3)
- 3 => string 'Mar' (length=3)
- 4 => string 'Apr' (length=3)
- 5 => string 'May' (length=3)
- 6 => string 'Jun' (length=3)
- 7 => string 'Jul' (length=3)
- 8 => string 'Aug' (length=3)
- 9 => string 'Sep' (length=3)
- 10 => string 'Oct' (length=3)
- 11 => string 'Nov' (length=3)
- 12 => string 'Dec' (length=3)

'maxdaysinmonth' => int 31

'calname' => string 'Gregorian' (length=9)

'calsymbol' => string 'CAL_GREGORIAN' (length=13)

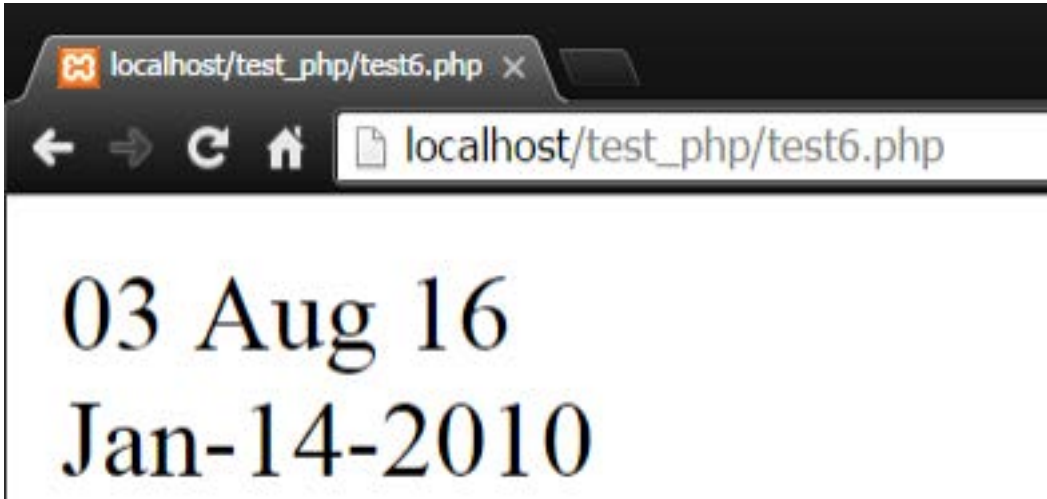
date สร้างข้อมูลวันเวลา

ใช้เพื่อรับค่าวัน เวลา ในรูปแบบ string

ตัวอย่าง

```
<?php
echo date('d M y');
echo "<br>";
echo date("M-d-Y", mktime(0, 0, 0, 1, 14, 2010));
?>
```

ผลลัพธ์

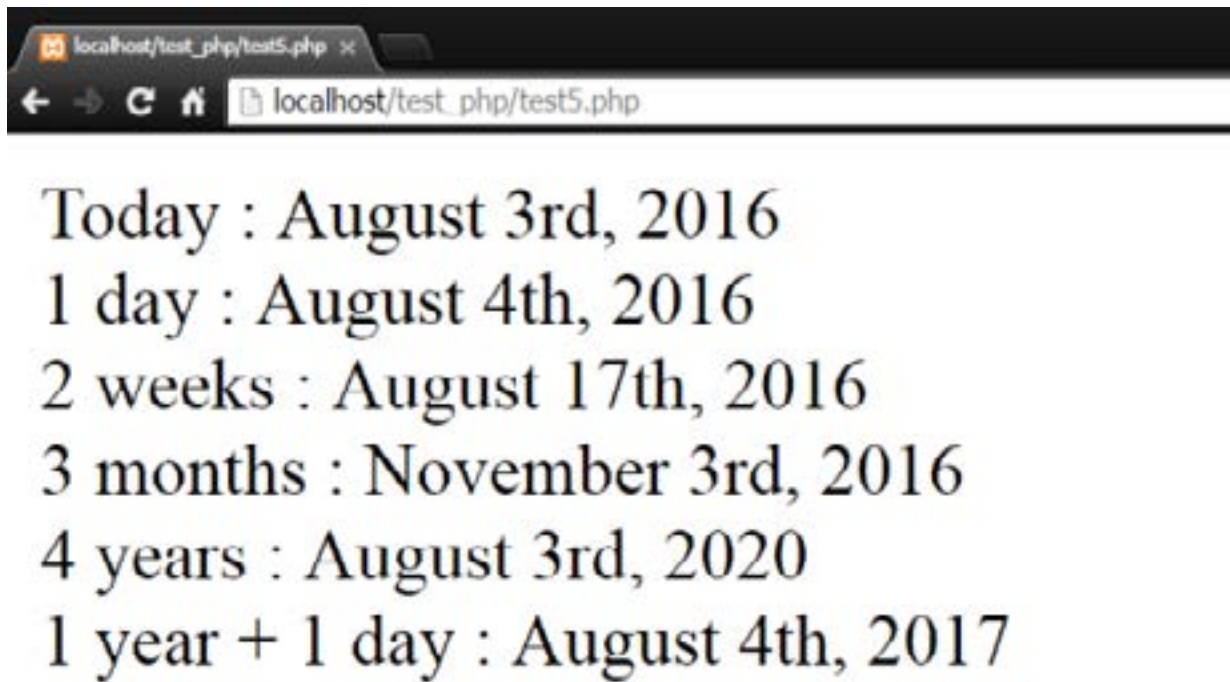


DateInterval::createFromDateString สร้างอวเปกต์ของคลาส โดยใช้คำ
สร้างอวเปกต์ของคลาส DateInterval โดยใช้ชื่อเราระบุไว้

ตัวอย่าง

```
<?php
echo "Today : ".date('F jS, Y', strtotime('today')) . "<br>";
$i = new DateInterval('P1D');
$i = DateInterval::createFromDateString('1 day');
echo "1 day : ".date('F jS, Y', strtotime('1 day')) . "<br>";
$i = new DateInterval('P2W');
$i = DateInterval::createFromDateString('2 weeks');
echo "2 weeks : ".date('F jS, Y', strtotime('2 weeks')) . "<br>";
$i = new DateInterval('P3M');
$i = DateInterval::createFromDateString('3 months');
echo "3 months : ".date('F jS, Y', strtotime('3 months')) . "<br>";
$i = new DateInterval('P4Y');
$i = DateInterval::createFromDateString('4 years');
echo "4 years : ".date('F jS, Y', strtotime('4 years')) . "<br>";
$i = new DateInterval('P1Y1D');
$i = DateInterval::createFromDateString('1 year + 1 day');
echo "1 year + 1 day : ".date('F jS, Y', strtotime('1 year + 1 day'));
?>
```

ผลลัพธ์

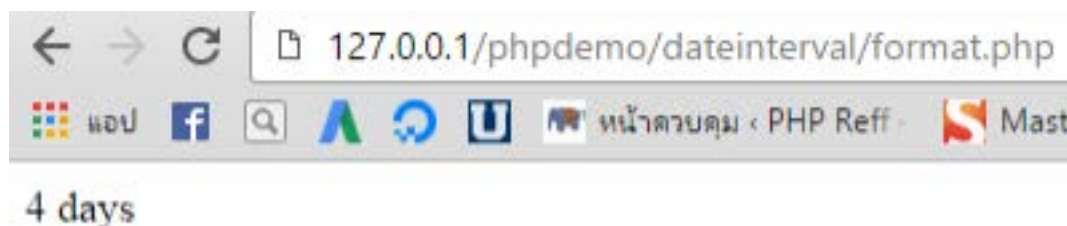


DateInterval::format จัดรูปแบบให้กับค่าวันเวลา

ตัวอย่าง

```
<?php
$interval = new DateInterval('P2Y4DT6H8M');
echo $interval->format('%d days');
?>
```

ผลลัพธ์

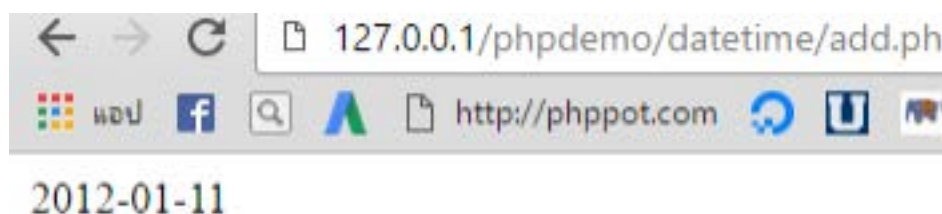


DateTime::add เพิ่ม ค่าวันหรือเวลาตามที่กำหนด

ตัวอย่าง

```
<?php
$date = new DateTime('2012-01-01');
$date->add(new DateInterval('P10D')); // ใช้คลาส DateInterval ซึ่งใช้จัดการช่วงเวลา
echo $date->format('Y-m-d') ;
?>
```

ผลลัพธ์

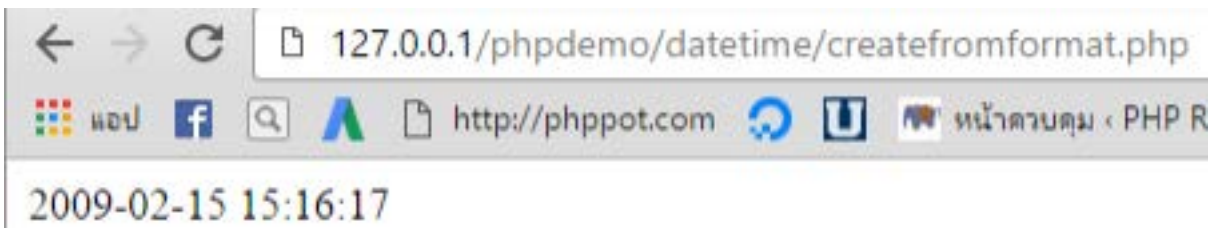


DateTime::createFromFormat สร้างอ็อบเจกต์ของวันเวลา โดยกำหนดรูปแบบให้ด้วย

ตัวอย่าง

```
<?php
$format = 'Y-m-d H s';
$date = DateTime::createFromFormat($format, '2009-02-15 15 17');
echo $date->format('Y-m-d H s') ;
?>
```

ผลลัพธ์

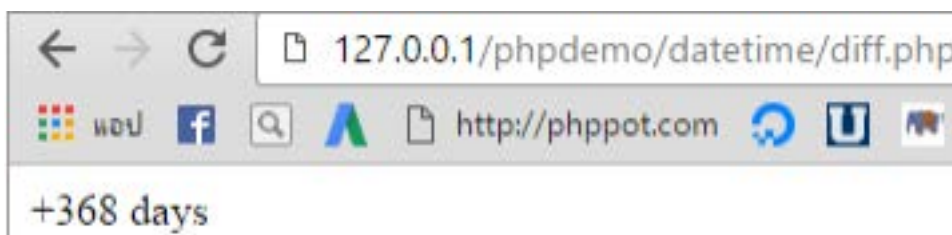


DateTime::diff หาความแตกต่างของอ็อบเจกต์ของคลาส DateTime

ตัวอย่าง

```
<?php
$date1 = new DateTime('2015-10-11');
$date2 = new DateTime('2016-10-13');
$interval = $date1->diff($date2);
echo $interval->format('%R%a days');
?>
```

ผลลัพธ์



DateTime::format เปลี่ยนรูปแบบของวันเวลา

ตัวอย่าง

```
<?php
$date = new DateTime('2000-01-01');
echo $date->format('Y-m-d H s');
?>
```

ผลลัพธ์

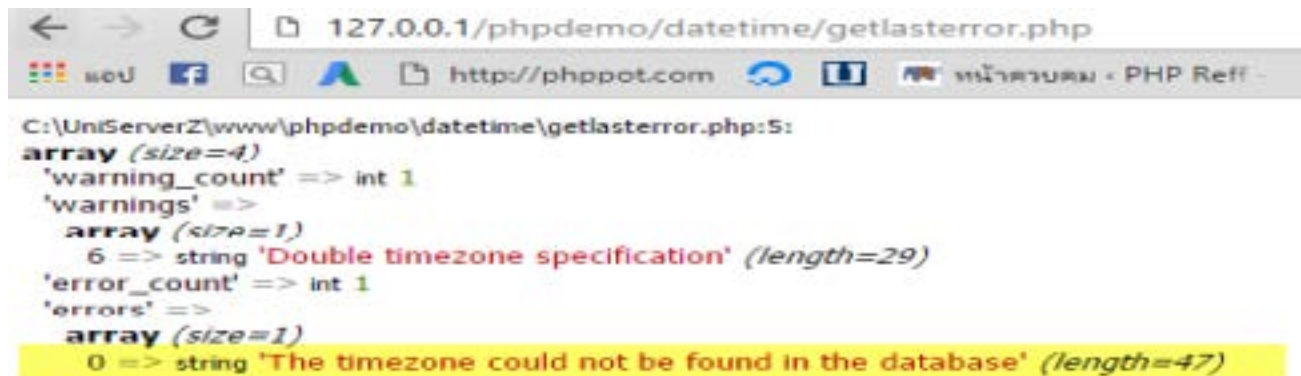


DateTime::getLastErrors ถึงค่าการแจ้งเตือนและข้อผิดพลาด

ตัวอย่าง

```
<?php
try {
    $date = new DateTime('asdfsdf');
} catch (Exception $e) {
    var_dump(DateTime::getLastErrors());
}
?>
```

ผลลัพธ์

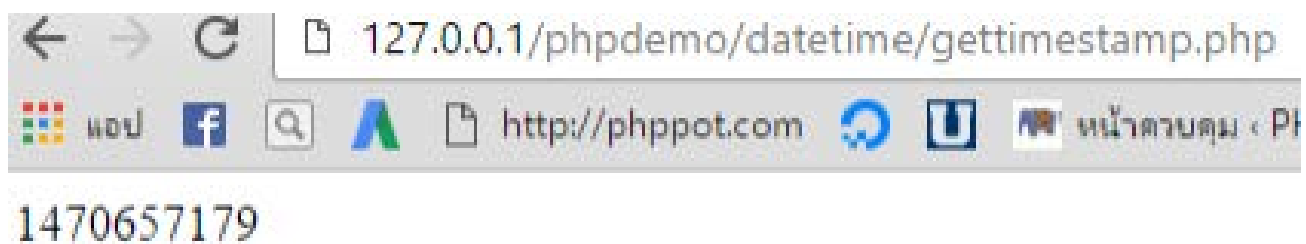


DateTime::getTimestamp แปลงค่าวันเวลาให้อยู่ในแบบ timestamp

ตัวอย่าง

```
<?php
$date = new DateTime();
echo $date->getTimestamp();
?>
```

ผลลัพธ์



DateTime::getTimezone ถึงชื่อโซนเวลาของออบเจกต์DateTime

ฟังก์ชัน `getTimezone()` จะ return ตัวแปรคลาส `DateTimeZone` ออกมาและจะใช้ฟังก์ชัน `getName()` ของคลาส `DateTimeZone` ในการรับค่า `timezone`

ตัวอย่าง

```
<?php
date_default_timezone_set('America/New_York');
$date = new DateTime();
$tz = $date->getTimezone();
echo $tz->getName();
?>
```


บลัฟท์



America/New_York

DateTime::modify แก้ไขค่าในออบเจกต์DateTime

เป็นการใช้ฟังก์ชัน modify ของคลาส DateTime เพื่อปรับวันเวลา
ตัวอย่าง

```
<?php
$date = new DateTime('2009-12-30');
echo $date->format('d-m-Y');
echo "<br>";
$date->modify('+1 day');
echo $date->format('d-m-Y');
echo "<br>";
$date->modify('-1 day');
echo $date->format('d-m-Y');
?>
```

บลัฟท์



30-12-2009

31-12-2009

30-12-2009

DateTime::setDate กำหนดวันที่ให้ออปเจกต์DateTime

เป็นการใช้ฟังก์ชัน setDate ของคลาส DateTime เพื่อกำหนดวันเดือนปี
ตัวอย่าง

```
<?php
$date = new DateTime();
$date->setDate(2001, 2, 28);
echo $date->format('Y-m-d') . "<br>";
$date->setDate(2001, 2, 29);
echo $date->format('Y/m/d') . "<br>";
$date->setDate(2001, 14, 3);
echo $date->format('Y/M/D') ;
?>
```

ผลลัพธ์



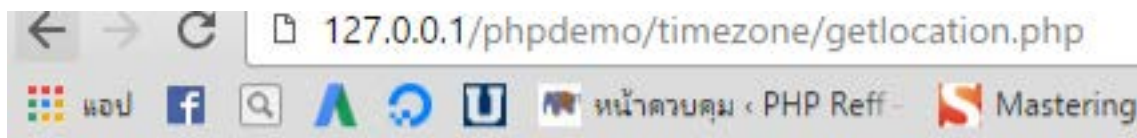
2001-02-28
2001/03/01
2002/Feb/Sun

DateTimeZone::getLocation ดึงค่าที่อยู่ของผู้ใช้งาน เช่น ละติจูด,ลองติจูด

ตัวอย่าง

```
<?php
$tz = new DateTimeZone("Asia/Bangkok");
var_dump($tz->getLocation());
?>
```

ผลลัพธ์



```
C:\UniServerZ\www\phpdemo\timezone\getlocation.php:3:
array (size=4)
  'country_code' => string 'TH' (length=2)
  'latitude' => float 13.75
  'longitude' => float 100.51666
  'comments' => string '' (length=0)
```

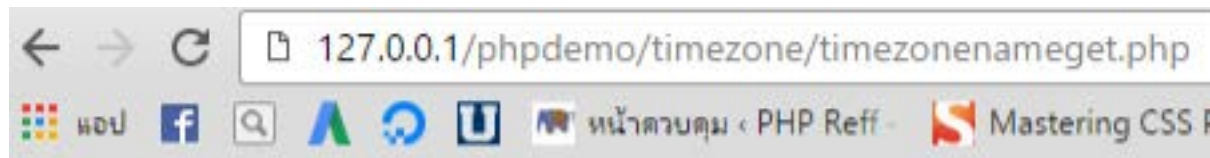
DateTimeZone::getName ถึงค่าชื่อของเขตเวลานั้น

getName คือ ฟังก์ชันที่ใช้ดึงชื่อของเขตเวลาจากออบเจกต์ของคลาส DateTimeZone

ตัวอย่าง

```
<?php
$DateTimeZone = timezone_open ( 'America/Chicago' );
echo 'ชื่อของเขตเวลาคือ '. timezone_name_get($DateTimeZone);
?>
```

ผลลัพธ์



ชื่อของเขตเวลาคือ America/Chicago

DateTimeZone::listIdentifiers ถึงรายชื่อเขตเวลาทั้งหมด

ตัวอย่าง

```
<?php
$timezone_identifier = DateTimeZone::listIdentifiers();
foreach ($timezone_identifier as $timezone_identifiers) {
    echo $timezone_identifiers.'<br>';
}
?>
```

ผลลัพธ์

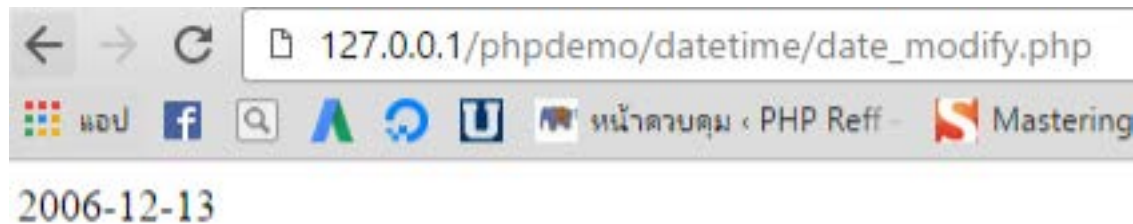


date_modify ใช้แก้ไขค่าในออบเจกต์DateTime

ตัวอย่าง

```
<?php
$date = date_create('2006-12-12');
date_modify($date, '+1 day');
echo date_format($date, 'Y-m-d');
?>
```

ผลลัพธ์

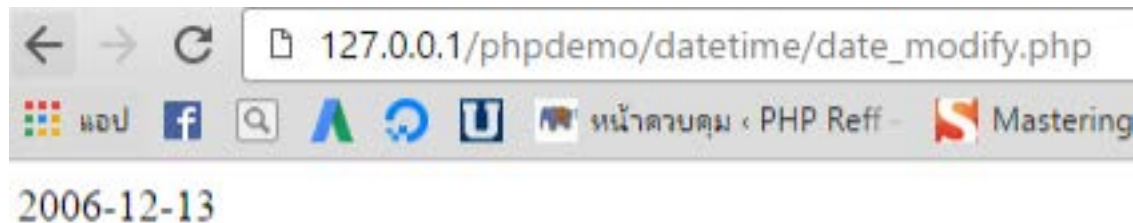


php localtime รับข้อมูลวันและเวลาในรูปแบบ array

ตัวอย่าง

```
<?php
$date = date_create('2006-12-12');
date_modify($date, '+1 day');
echo date_format($date, 'Y-m-d');
?>
```

ผลลัพธ์



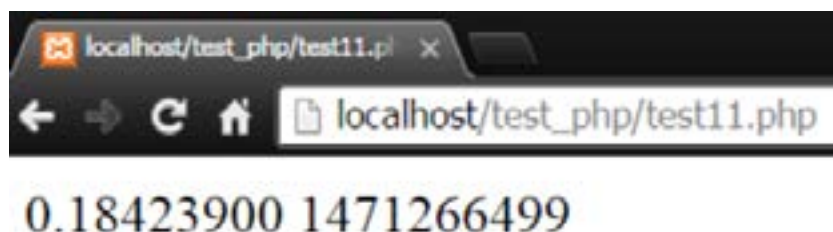
microtime แสดงวันเวลาปัจจุบันแบบ ไมโครวินาที

แสดงวันเวลาปัจจุบันแบบ ไมโครวินาที

ตัวอย่าง

```
<?php
echo microtime() ;
?>
```

ผลลัพธ์



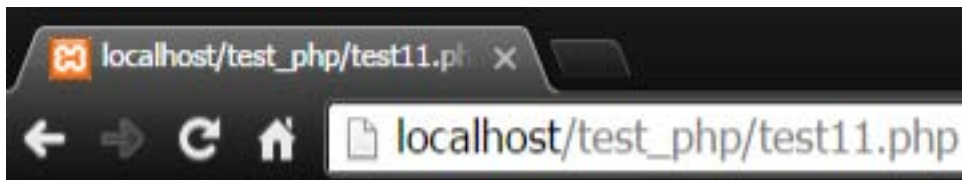
mktime สร้างข้อมูลของวันเวลาตามที่ระบุไว้

สร้างข้อมูลของวันเวลาตามที่ระบุไว้

ตัวอย่าง

```
<?php
echo date("M-d-Y", mktime(0, 0, 0, 1, 1, 1997)); // mktime(hour,minute
,second,month,day,year)
?>
```

ผลลัพธ์



Jan-01-1997

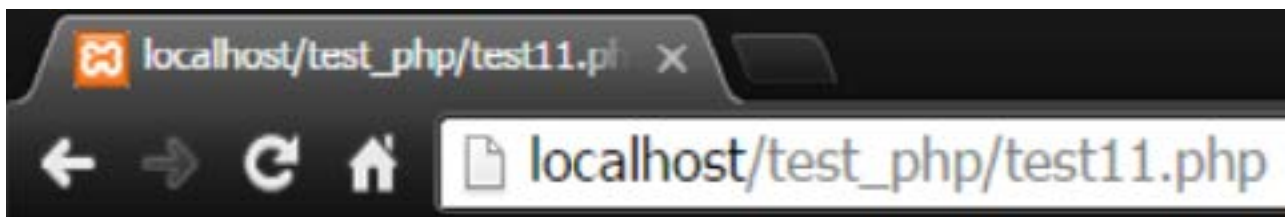
time แสดงข้อมูลของวันเวลาปัจจุบัน

แสดงข้อมูลของวันเวลาปัจจุบัน

ตัวอย่าง

```
<?php
echo time();
echo "<br>" ;
echo date('Y-m-d',time());
?>
```

ผลลัพธ์



1471264542

2016-08-15

strtotime รับค่าเวลาของวันเวลาตาม string ที่ใส่มาเป็นพารามิเตอร์

รับค่าเวลาของวันเวลาตาม string ที่ใส่เป็นพารามิเตอร์

ตัวอย่าง

```
<?php
echo strtotime("now"), "\n <br>";
echo strtotime("10 September 2000"), "\n <br>";
echo strtotime("+1 day"), "\n <br>";
?>
```

ผลลัพธ์

