

---

# phpf1 Documentation

*Release v0.1*

Jan 14, 2021



**CONTENTS**

<b>1</b>	<b>Table of Contents</b>	<b>3</b>
1.1	Dates . . . . .	3



The PHPF1 Utils library provides snippet collection for everyday PHP development.



## TABLE OF CONTENTS

### 1.1 Dates

Date and time related utility functions.

#### 1.1.1 today

```
$today = Dates::today();
```

Returns today's date as a `DateTime` object. [Source](#)

**returns** `DateTime`

#### 1.1.2 todayAsString

Syntax:

```
$today = Dates::todayAsString($pattern);
```

Returns today's date as string. By default it returns the date in ISO format. The output format can be changed by the `$pattern` parameter. [Source](#)

**\$pattern** Date formatting pattern. Optional. The default value is *Y-m-d*

**returns** `string`

#### 1.1.3 yesterday

Syntax:

```
$yesterday = Dates::yesterday();
```

Returns yesterday's date as a `DateTime` object. [Source](#)

**returns** `DateTime`

### 1.1.4 yesterdayAsString

Syntax:

```
$yesterday = Dates::yesterdayAsString($pattern = 'Y-m-d');
```

Returns yesterday's date as string. By default it returns the date in ISO format The output format can be changed by the `$pattern` parameter [Source](#)

**param \$pattern** Date formatting pattern

**returns** string

### 1.1.5 tomorrow

Syntax:

```
$tomorrow = Dates::tomorrow();
```

Returns tomorrow's date as a DateTime object. [Source](#)

**returns** DateTime

### 1.1.6 tomorrowAsString

Syntax:

```
$tomorrow = Dates::tomorrowAsString($pattern = 'Y-m-d')
```

Returns tomorrow's date as string. By default it returns the date in ISO format The output format can be changed by the `$pattern` parameter. [Source](#)

**param \$pattern** Date formatting pattern

**returns** string

### 1.1.7 differenceInDays

Syntax:

```
$differenceInDay = Dates::differenceInDays(DateTime $startDate, DateTime  
↪ $endDate);
```

Returns the difference in days between two dates. [Source](#)

**type \$startDate** DateTime

**param \$startDate**

**type \$endDate** DateTime

**param \$endDate**

**returns** string



### 1.1.8 differenceInDaysFromString

Syntax:

```
$differenceInDays = Dates::differenceInDaysFromString(string $startDate,   
↳string $endDate);
```

Returns the difference in days between two dates that are defined as strings. [Source](#)

**type \$startDate** string

**param \$startDate**

**type \$endDate** string

**param \$endDate**

**returns** string

### 1.1.9 addDaysToDate

Syntax:

```
$newDate = Dates::addDaysToDate(string $originalDate, int $days);
```

Adds days to the original date and returns the new date as a DateTime object. [Source](#)

**type \$originalDate** string

**param \$originalDate**

**type \$days** int

**param \$days**

**returns** DateTime

### 1.1.10 addMinutesToTime

Syntax:

```
$newTime = Dates::addMinutesToTime(string $originalTime, int $minutes);
```

Adds minutes to the original time and returns the new time as a DateTime object. [Source](#)

**type \$originalTime** string

**param \$originalTime**

**type \$minutes** int

**param \$minutes**

**returns** DateTime

### 1.1.11 getLastDayOfMonth

Syntax:

```
$newTime = Dates::getLastDayOfMonth(string $actualDate);
```

Returns the last day of the month based on the specified date. [Source](#)

**type** `$actualDate` string

**param** `$actualDate`

**returns** DateTime

### 1.1.12 timestampToDateTime

Syntax:

```
$date = Dates::timestampToDateTime(int $timestamp);
```

Converts the given Unix timestamp to a DateTime object. [Source](#)

**type** `$timestamp` int

**param** `$timestamp`

**returns** DateTime

### 1.1.13 dateToTimestamp

Syntax:

```
$timestamp = Dates::dateToTimestamp(string $dateStr);
```

Returns the Unix timestamp representation of a given date. [Source](#)

**type** `$dateStr` string

**param** `$dateStr`

**returns** int