



Time Class

Contains methods for the Time primitive data type.

Namespace

[System](#)

Usage

For more information on time, see [Time Data Type](#).

Time Methods

The following are methods for `Time`.

- [addHours\(additionalHours\)](#)
Adds the specified number of hours to a Time.
- [addMilliseconds\(additionalMilliseconds\)](#)
Adds the specified number of milliseconds to a Time.
- [addMinutes\(additionalMinutes\)](#)
Adds the specified number of minutes to a Time.
- [addSeconds\(additionalSeconds\)](#)
Adds the specified number of seconds to a Time.
- [hour\(\)](#)
Returns the hour component of a Time.
- [millisecond\(\)](#)
Returns the millisecond component of a Time.
- [minute\(\)](#)
Returns the minute component of a Time.
- [newInstance\(hour, minutes, seconds, milliseconds\)](#)
Constructs a Time from Integer representations of the specified hour, minutes, seconds, and milliseconds. (UTC is assumed.)
- [second\(\)](#)
Returns the second component of a Time.

addHours(additionalHours)

Adds the specified number of hours to a Time.

Signature

```
public Time addHours(Integer additionalHours)
```

Parameters

additionalHours

Type: [Integer](#)

Return Value



```
Time myTime = Time.newInstance(1, 2, 3, 4);
Time expected = Time.newInstance(4, 2, 3, 4);
System.assertEquals(expected, myTime.addHours(3));
```

addMilliseconds(additionalMilliseconds)

Adds the specified number of milliseconds to a Time.

Signature

```
public Time addMilliseconds(Integer additionalMilliseconds)
```

Parameters

additionalMilliseconds

Type: [Integer](#)

Return Value

Type: [Time](#)

Example

```
Time myTime = Time.newInstance(1, 2, 3, 0);
Time expected = Time.newInstance(1, 2, 4, 400);
System.assertEquals(expected, myTime.addMilliseconds(1400));
```

addMinutes(additionalMinutes)

Adds the specified number of minutes to a Time.

Signature

```
public Time addMinutes(Integer additionalMinutes)
```

Parameters

additionalMinutes

Type: [Integer](#)

Return Value

Type: [Time](#)

Example

```
Time myTime = Time.newInstance(18, 30, 2, 20);
Integer myMinutes = myTime.minute();
myMinutes = myMinutes + 5;
System.assertEquals(myMinutes, 35);
```

addSeconds(additionalSeconds)

Adds the specified number of seconds to a Time.

Signature



Type: [Integer](#)

Return Value

Type: [Time](#)

Example

```
Time myTime = Time.newInstance(1, 2, 55, 0);
Time expected = Time.newInstance(1, 3, 5, 0);
System.assertEquals(expected, myTime.addSeconds(10));
```

hour()

Returns the hour component of a Time.

Signature

```
public Integer hour()
```

Return Value

Type: [Integer](#)

Example

```
Time myTime = Time.newInstance(18, 30, 2, 20);
myTime = myTime.addHours(2);
Integer myHour = myTime.hour();
System.assertEquals(myHour, 20);
```

millisecond()

Returns the millisecond component of a Time.

Signature

```
public Integer millisecond()
```

Return Value

Type: [Integer](#)

Example

```
Time myTime = Time.newInstance(3, 14, 15, 926);
System.assertEquals(926, myTime.millisecond());
```

minute()

Returns the minute component of a Time.

Signature

```
public Integer minute()
```



```
Time myTime = Time.newInstance(3, 14, 15, 926);
System.assertEquals(14, myTime.minute());
```

newInstance(hour, minutes, seconds, milliseconds)

Constructs a Time from Integer representations of the specified hour, minutes, seconds, and milliseconds. (UTC is assumed.)

Signature

```
public static Time newInstance(Integer hour, Integer minutes, Integer seconds, Integer milliseconds)
```

Parameters

hour

Type: [Integer](#)

minutes

Type: [Integer](#)

seconds

Type: [Integer](#)

milliseconds

Type: [Integer](#)

Return Value

Type: [Time](#)

Example

The following example creates a time of 18:30:2:20 (UTC).

```
Time myTime =
Time.newInstance(18, 30, 2, 20);
```

second()

Returns the second component of a Time.

Signature

```
public Integer second()
```

Return Value

Type: [Integer](#)

Example

```
Time myTime = Time.newInstance(3, 14, 15, 926);
System.assertEquals(15, myTime.second());
```



DEVELOPER CENTERS

- Heroku
- MuleSoft
- Tableau
- Commerce Cloud
- Lightning Design System
- Einstein
- Quip

POPULAR RESOURCES

- Documentation
- Component Library
- APIs
- Trailhead
- Sample Apps
- Podcasts
- AppExchange

COMMUNITY

- Trailblazer Community
- Events and Calendar
- Partner Community
- Blog
- Salesforce Admins
- Salesforce Architects

© Copyright 2025 Salesforce, Inc. [All rights reserved.](#) Various trademarks held by their respective owners. Salesforce, Inc. Salesforce Tower, 415 Mission Street, 3rd Floor, San Francisco, CA 94105, United States

- [Privacy Information](#)
- [Terms of Service](#)
- [Legal](#)
- [Use of Cookies](#)
- [Trust](#)
- [Cookie Preferences](#)
- [Your Privacy Choices](#)
- [Responsible Disclosure](#)
- [Contact](#)