Date ← Date and Time → Time Zone and Offset

Date and Time

The LocalTime Class

The `LocalTime` class is similar to the other classes whose names are prefixed with `Local`, but deals in time only. This class is useful for representing human-based time of day, such as movie times, or the opening and closing times of the local library. It could also be used to create a digital clock, as shown in the following example:

```
LocalTime thisSec;

for (;;) {
    thisSec = LocalTime.now();

    // implementation of display code is left to the reader
    display(thisSec.getHour(), thisSec.getMinute(), thisSec.getSecond());
}
```

The `LocalTime` class does not store time zone or daylight saving time information.

The LocalDateTime Class

The class that handles both date and time, without a time zone, is `LocalDateTime`, one of the core classes of the Date-Time API. This class is used to represent date (month-day-year) together with time (hour-minute-second-nanosecond) and is, in effect, a combination of `LocalDate` with `LocalTime`. This class can be used to represent a specific event, such as the first race for the Louis Vuitton Cup Finals in the America's Cup Challenger Series, which began at 1:10 p.m. on August 17, 2013. Note that this means 1:10 p.m. in local time. To include a time zone, you must use a `ZonedDateTime` or an `OffsetDateTime`, as discussed in Time Zone and Offset Classes.

In addition to the now() method that every temporal-based class provides, the LocalDateTime class has various of() methods (or methods prefixed with of) that create an instance of LocalDateTime. There is a from() method that converts an instance from another temporal format to a LocalDateTime instance. There are also methods for adding or subtracting hours, minutes, days, weeks, and months. The following example shows a few of these methods:

```
System.out.printf("now: %s%n", LocalDateTime.now());

System.out.printf("Apr 15, 1994 @ 11:30am: %s%n",
                  LocalDateTime.of(1994, Month.APRIL, 15, 11, 30));

System.out.printf("now (from Instant): %s%n",
                  LocalDateTime.ofInstant(Instant.now(), ZoneId.systemDefault()));

System.out.printf("6 months from now: %s%n",
                  LocalDateTime.now().plusMonths(6));

System.out.printf("6 months ago: %s%n",
                  LocalDateTime.now().minusMonths(6));
```

This code produces output that will look similar to the following:

```
now: 2013-07-24T17:13:59.985
Apr 15, 1994 @ 11:30am: 1994-04-15T11:30
now (from Instant): 2013-07-24T17:14:00.479
6 months from now: 2014-01-24T17:14:00.480
6 months ago: 2013-01-24T17:14:00.481
```

Last update: *January 27, 2022*

Date ← Date and Time → Time Zone and Offset

Home > Tutorials > The Date Time API > Date and Time