| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/DateFormat.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/java/text/Collator.html)   [**NEXT CLASS**](http://docs.google.com/java/text/DateFormat.Field.html) | [**FRAMES**](http://docs.google.com/index.html?java/text/DateFormat.html)    [**NO FRAMES**](http://docs.google.com/DateFormat.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#2et92p0) | [FIELD](#tyjcwt) | [CONSTR](#3dy6vkm) | [METHOD](#1t3h5sf) | DETAIL: [FIELD](#17dp8vu) | [CONSTR](#1hmsyys) | [METHOD](#2grqrue) |

## **java.text**

Class DateFormat

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 [java.text.Format](http://docs.google.com/java/text/Format.html)  
 **java.text.DateFormat**

**All Implemented Interfaces:** [Serializable](http://docs.google.com/java/io/Serializable.html), [Cloneable](http://docs.google.com/java/lang/Cloneable.html) **Direct Known Subclasses:** [SimpleDateFormat](http://docs.google.com/java/text/SimpleDateFormat.html)

public abstract class **DateFormat**extends [Format](http://docs.google.com/java/text/Format.html)

DateFormat is an abstract class for date/time formatting subclasses which formats and parses dates or time in a language-independent manner. The date/time formatting subclass, such as SimpleDateFormat, allows for formatting (i.e., date -> text), parsing (text -> date), and normalization. The date is represented as a Date object or as the milliseconds since January 1, 1970, 00:00:00 GMT.

DateFormat provides many class methods for obtaining default date/time formatters based on the default or a given locale and a number of formatting styles. The formatting styles include FULL, LONG, MEDIUM, and SHORT. More detail and examples of using these styles are provided in the method descriptions.

DateFormat helps you to format and parse dates for any locale. Your code can be completely independent of the locale conventions for months, days of the week, or even the calendar format: lunar vs. solar.

To format a date for the current Locale, use one of the static factory methods:

myString = DateFormat.getDateInstance().format(myDate);

If you are formatting multiple dates, it is more efficient to get the format and use it multiple times so that the system doesn't have to fetch the information about the local language and country conventions multiple times.

DateFormat df = DateFormat.getDateInstance();  
 for (int i = 0; i < myDate.length; ++i) {  
 output.println(df.format(myDate[i]) + "; ");  
 }

To format a date for a different Locale, specify it in the call to getDateInstance().

DateFormat df = DateFormat.getDateInstance(DateFormat.LONG, Locale.FRANCE);

You can use a DateFormat to parse also.

myDate = df.parse(myString);

Use getDateInstance to get the normal date format for that country. There are other static factory methods available. Use getTimeInstance to get the time format for that country. Use getDateTimeInstance to get a date and time format. You can pass in different options to these factory methods to control the length of the result; from SHORT to MEDIUM to LONG to FULL. The exact result depends on the locale, but generally:

* SHORT is completely numeric, such as 12.13.52 or 3:30pm
* MEDIUM is longer, such as Jan 12, 1952
* LONG is longer, such as January 12, 1952 or 3:30:32pm
* FULL is pretty completely specified, such as Tuesday, April 12, 1952 AD or 3:30:42pm PST.

You can also set the time zone on the format if you wish. If you want even more control over the format or parsing, (or want to give your users more control), you can try casting the DateFormat you get from the factory methods to a SimpleDateFormat. This will work for the majority of countries; just remember to put it in a try block in case you encounter an unusual one.

You can also use forms of the parse and format methods with ParsePosition and FieldPosition to allow you to

* progressively parse through pieces of a string.
* align any particular field, or find out where it is for selection on the screen.

#### Synchronization

Date formats are not synchronized. It is recommended to create separate format instances for each thread. If multiple threads access a format concurrently, it must be synchronized externally.

**See Also:**[Format](http://docs.google.com/java/text/Format.html), [NumberFormat](http://docs.google.com/java/text/NumberFormat.html), [SimpleDateFormat](http://docs.google.com/java/text/SimpleDateFormat.html), [Calendar](http://docs.google.com/java/util/Calendar.html), [GregorianCalendar](http://docs.google.com/java/util/GregorianCalendar.html), [TimeZone](http://docs.google.com/java/util/TimeZone.html), [Serialized Form](http://docs.google.com/serialized-form.html#java.text.DateFormat)

| **Nested Class Summary** | |
| --- | --- |
| static class | [**DateFormat.Field**](http://docs.google.com/java/text/DateFormat.Field.html)            Defines constants that are used as attribute keys in the AttributedCharacterIterator returned from DateFormat.formatToCharacterIterator and as field identifiers in FieldPosition. |

| **Field Summary** | |
| --- | --- |
| static int | [**AM\_PM\_FIELD**](http://docs.google.com/java/text/DateFormat.html#AM_PM_FIELD)            Useful constant for AM\_PM field alignment. |
| protected  [Calendar](http://docs.google.com/java/util/Calendar.html) | [**calendar**](http://docs.google.com/java/text/DateFormat.html#calendar)            The calendar that DateFormat uses to produce the time field values needed to implement date and time formatting. |
| static int | [**DATE\_FIELD**](http://docs.google.com/java/text/DateFormat.html#DATE_FIELD)            Useful constant for DATE field alignment. |
| static int | [**DAY\_OF\_WEEK\_FIELD**](http://docs.google.com/java/text/DateFormat.html#DAY_OF_WEEK_FIELD)            Useful constant for DAY\_OF\_WEEK field alignment. |
| static int | [**DAY\_OF\_WEEK\_IN\_MONTH\_FIELD**](http://docs.google.com/java/text/DateFormat.html#DAY_OF_WEEK_IN_MONTH_FIELD)            Useful constant for DAY\_OF\_WEEK\_IN\_MONTH field alignment. |
| static int | [**DAY\_OF\_YEAR\_FIELD**](http://docs.google.com/java/text/DateFormat.html#DAY_OF_YEAR_FIELD)            Useful constant for DAY\_OF\_YEAR field alignment. |
| static int | [**DEFAULT**](http://docs.google.com/java/text/DateFormat.html#DEFAULT)            Constant for default style pattern. |
| static int | [**ERA\_FIELD**](http://docs.google.com/java/text/DateFormat.html#ERA_FIELD)            Useful constant for ERA field alignment. |
| static int | [**FULL**](http://docs.google.com/java/text/DateFormat.html#FULL)            Constant for full style pattern. |
| static int | [**HOUR\_OF\_DAY0\_FIELD**](http://docs.google.com/java/text/DateFormat.html#HOUR_OF_DAY0_FIELD)            Useful constant for zero-based HOUR\_OF\_DAY field alignment. |
| static int | [**HOUR\_OF\_DAY1\_FIELD**](http://docs.google.com/java/text/DateFormat.html#HOUR_OF_DAY1_FIELD)            Useful constant for one-based HOUR\_OF\_DAY field alignment. |
| static int | [**HOUR0\_FIELD**](http://docs.google.com/java/text/DateFormat.html#HOUR0_FIELD)            Useful constant for zero-based HOUR field alignment. |
| static int | [**HOUR1\_FIELD**](http://docs.google.com/java/text/DateFormat.html#HOUR1_FIELD)            Useful constant for one-based HOUR field alignment. |
| static int | [**LONG**](http://docs.google.com/java/text/DateFormat.html#LONG)            Constant for long style pattern. |
| static int | [**MEDIUM**](http://docs.google.com/java/text/DateFormat.html#MEDIUM)            Constant for medium style pattern. |
| static int | [**MILLISECOND\_FIELD**](http://docs.google.com/java/text/DateFormat.html#MILLISECOND_FIELD)            Useful constant for MILLISECOND field alignment. |
| static int | [**MINUTE\_FIELD**](http://docs.google.com/java/text/DateFormat.html#MINUTE_FIELD)            Useful constant for MINUTE field alignment. |
| static int | [**MONTH\_FIELD**](http://docs.google.com/java/text/DateFormat.html#MONTH_FIELD)            Useful constant for MONTH field alignment. |
| protected  [NumberFormat](http://docs.google.com/java/text/NumberFormat.html) | [**numberFormat**](http://docs.google.com/java/text/DateFormat.html#numberFormat)            The number formatter that DateFormat uses to format numbers in dates and times. |
| static int | [**SECOND\_FIELD**](http://docs.google.com/java/text/DateFormat.html#SECOND_FIELD)            Useful constant for SECOND field alignment. |
| static int | [**SHORT**](http://docs.google.com/java/text/DateFormat.html#SHORT)            Constant for short style pattern. |
| static int | [**TIMEZONE\_FIELD**](http://docs.google.com/java/text/DateFormat.html#TIMEZONE_FIELD)            Useful constant for TIMEZONE field alignment. |
| static int | [**WEEK\_OF\_MONTH\_FIELD**](http://docs.google.com/java/text/DateFormat.html#WEEK_OF_MONTH_FIELD)            Useful constant for WEEK\_OF\_MONTH field alignment. |
| static int | [**WEEK\_OF\_YEAR\_FIELD**](http://docs.google.com/java/text/DateFormat.html#WEEK_OF_YEAR_FIELD)            Useful constant for WEEK\_OF\_YEAR field alignment. |
| static int | [**YEAR\_FIELD**](http://docs.google.com/java/text/DateFormat.html#YEAR_FIELD)            Useful constant for YEAR field alignment. |

| **Constructor Summary** | |
| --- | --- |
| protected | [**DateFormat**](http://docs.google.com/java/text/DateFormat.html#DateFormat())()            Create a new date format. |

| **Method Summary** | |
| --- | --- |
| [Object](http://docs.google.com/java/lang/Object.html) | [**clone**](http://docs.google.com/java/text/DateFormat.html#clone())()            Overrides Cloneable |
| boolean | [**equals**](http://docs.google.com/java/text/DateFormat.html#equals(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) obj)            Overrides equals |
| [String](http://docs.google.com/java/lang/String.html) | [**format**](http://docs.google.com/java/text/DateFormat.html#format(java.util.Date))([Date](http://docs.google.com/java/util/Date.html) date)            Formats a Date into a date/time string. |
| abstract  [StringBuffer](http://docs.google.com/java/lang/StringBuffer.html) | [**format**](http://docs.google.com/java/text/DateFormat.html#format(java.util.Date,%20java.lang.StringBuffer,%20java.text.FieldPosition))([Date](http://docs.google.com/java/util/Date.html) date, [StringBuffer](http://docs.google.com/java/lang/StringBuffer.html) toAppendTo, [FieldPosition](http://docs.google.com/java/text/FieldPosition.html) fieldPosition)            Formats a Date into a date/time string. |
| [StringBuffer](http://docs.google.com/java/lang/StringBuffer.html) | [**format**](http://docs.google.com/java/text/DateFormat.html#format(java.lang.Object,%20java.lang.StringBuffer,%20java.text.FieldPosition))([Object](http://docs.google.com/java/lang/Object.html) obj, [StringBuffer](http://docs.google.com/java/lang/StringBuffer.html) toAppendTo, [FieldPosition](http://docs.google.com/java/text/FieldPosition.html) fieldPosition)            Overrides Format. |
| static [Locale](http://docs.google.com/java/util/Locale.html)[] | [**getAvailableLocales**](http://docs.google.com/java/text/DateFormat.html#getAvailableLocales())()            Returns an array of all locales for which the get\*Instance methods of this class can return localized instances. |
| [Calendar](http://docs.google.com/java/util/Calendar.html) | [**getCalendar**](http://docs.google.com/java/text/DateFormat.html#getCalendar())()            Gets the calendar associated with this date/time formatter. |
| static [DateFormat](http://docs.google.com/java/text/DateFormat.html) | [**getDateInstance**](http://docs.google.com/java/text/DateFormat.html#getDateInstance())()            Gets the date formatter with the default formatting style for the default locale. |
| static [DateFormat](http://docs.google.com/java/text/DateFormat.html) | [**getDateInstance**](http://docs.google.com/java/text/DateFormat.html#getDateInstance(int))(int style)            Gets the date formatter with the given formatting style for the default locale. |
| static [DateFormat](http://docs.google.com/java/text/DateFormat.html) | [**getDateInstance**](http://docs.google.com/java/text/DateFormat.html#getDateInstance(int,%20java.util.Locale))(int style, [Locale](http://docs.google.com/java/util/Locale.html) aLocale)            Gets the date formatter with the given formatting style for the given locale. |
| static [DateFormat](http://docs.google.com/java/text/DateFormat.html) | [**getDateTimeInstance**](http://docs.google.com/java/text/DateFormat.html#getDateTimeInstance())()            Gets the date/time formatter with the default formatting style for the default locale. |
| static [DateFormat](http://docs.google.com/java/text/DateFormat.html) | [**getDateTimeInstance**](http://docs.google.com/java/text/DateFormat.html#getDateTimeInstance(int,%20int))(int dateStyle, int timeStyle)            Gets the date/time formatter with the given date and time formatting styles for the default locale. |
| static [DateFormat](http://docs.google.com/java/text/DateFormat.html) | [**getDateTimeInstance**](http://docs.google.com/java/text/DateFormat.html#getDateTimeInstance(int,%20int,%20java.util.Locale))(int dateStyle, int timeStyle, [Locale](http://docs.google.com/java/util/Locale.html) aLocale)            Gets the date/time formatter with the given formatting styles for the given locale. |
| static [DateFormat](http://docs.google.com/java/text/DateFormat.html) | [**getInstance**](http://docs.google.com/java/text/DateFormat.html#getInstance())()            Get a default date/time formatter that uses the SHORT style for both the date and the time. |
| [NumberFormat](http://docs.google.com/java/text/NumberFormat.html) | [**getNumberFormat**](http://docs.google.com/java/text/DateFormat.html#getNumberFormat())()            Gets the number formatter which this date/time formatter uses to format and parse a time. |
| static [DateFormat](http://docs.google.com/java/text/DateFormat.html) | [**getTimeInstance**](http://docs.google.com/java/text/DateFormat.html#getTimeInstance())()            Gets the time formatter with the default formatting style for the default locale. |
| static [DateFormat](http://docs.google.com/java/text/DateFormat.html) | [**getTimeInstance**](http://docs.google.com/java/text/DateFormat.html#getTimeInstance(int))(int style)            Gets the time formatter with the given formatting style for the default locale. |
| static [DateFormat](http://docs.google.com/java/text/DateFormat.html) | [**getTimeInstance**](http://docs.google.com/java/text/DateFormat.html#getTimeInstance(int,%20java.util.Locale))(int style, [Locale](http://docs.google.com/java/util/Locale.html) aLocale)            Gets the time formatter with the given formatting style for the given locale. |
| [TimeZone](http://docs.google.com/java/util/TimeZone.html) | [**getTimeZone**](http://docs.google.com/java/text/DateFormat.html#getTimeZone())()            Gets the time zone. |
| int | [**hashCode**](http://docs.google.com/java/text/DateFormat.html#hashCode())()            Overrides hashCode |
| boolean | [**isLenient**](http://docs.google.com/java/text/DateFormat.html#isLenient())()            Tell whether date/time parsing is to be lenient. |
| [Date](http://docs.google.com/java/util/Date.html) | [**parse**](http://docs.google.com/java/text/DateFormat.html#parse(java.lang.String))([String](http://docs.google.com/java/lang/String.html) source)            Parses text from the beginning of the given string to produce a date. |
| abstract  [Date](http://docs.google.com/java/util/Date.html) | [**parse**](http://docs.google.com/java/text/DateFormat.html#parse(java.lang.String,%20java.text.ParsePosition))([String](http://docs.google.com/java/lang/String.html) source, [ParsePosition](http://docs.google.com/java/text/ParsePosition.html) pos)            Parse a date/time string according to the given parse position. |
| [Object](http://docs.google.com/java/lang/Object.html) | [**parseObject**](http://docs.google.com/java/text/DateFormat.html#parseObject(java.lang.String,%20java.text.ParsePosition))([String](http://docs.google.com/java/lang/String.html) source, [ParsePosition](http://docs.google.com/java/text/ParsePosition.html) pos)            Parses text from a string to produce a Date. |
| void | [**setCalendar**](http://docs.google.com/java/text/DateFormat.html#setCalendar(java.util.Calendar))([Calendar](http://docs.google.com/java/util/Calendar.html) newCalendar)            Set the calendar to be used by this date format. |
| void | [**setLenient**](http://docs.google.com/java/text/DateFormat.html#setLenient(boolean))(boolean lenient)            Specify whether or not date/time parsing is to be lenient. |
| void | [**setNumberFormat**](http://docs.google.com/java/text/DateFormat.html#setNumberFormat(java.text.NumberFormat))([NumberFormat](http://docs.google.com/java/text/NumberFormat.html) newNumberFormat)            Allows you to set the number formatter. |
| void | [**setTimeZone**](http://docs.google.com/java/text/DateFormat.html#setTimeZone(java.util.TimeZone))([TimeZone](http://docs.google.com/java/util/TimeZone.html) zone)            Sets the time zone for the calendar of this DateFormat object. |

| **Methods inherited from class java.text.**[**Format**](http://docs.google.com/java/text/Format.html) |
| --- |
| [format](http://docs.google.com/java/text/Format.html#format(java.lang.Object)), [formatToCharacterIterator](http://docs.google.com/java/text/Format.html#formatToCharacterIterator(java.lang.Object)), [parseObject](http://docs.google.com/java/text/Format.html#parseObject(java.lang.String)) |

| **Methods inherited from class java.lang.**[**Object**](http://docs.google.com/java/lang/Object.html) |
| --- |
| [finalize](http://docs.google.com/java/lang/Object.html#finalize()), [getClass](http://docs.google.com/java/lang/Object.html#getClass()), [notify](http://docs.google.com/java/lang/Object.html#notify()), [notifyAll](http://docs.google.com/java/lang/Object.html#notifyAll()), [toString](http://docs.google.com/java/lang/Object.html#toString()), [wait](http://docs.google.com/java/lang/Object.html#wait()), [wait](http://docs.google.com/java/lang/Object.html#wait(long)), [wait](http://docs.google.com/java/lang/Object.html#wait(long,%20int)) |

| **Field Detail** |
| --- |

### calendar

protected [Calendar](http://docs.google.com/java/util/Calendar.html) **calendar**

The calendar that DateFormat uses to produce the time field values needed to implement date and time formatting. Subclasses should initialize this to a calendar appropriate for the locale associated with this DateFormat.

### numberFormat

protected [NumberFormat](http://docs.google.com/java/text/NumberFormat.html) **numberFormat**

The number formatter that DateFormat uses to format numbers in dates and times. Subclasses should initialize this to a number format appropriate for the locale associated with this DateFormat.

### ERA\_FIELD

public static final int **ERA\_FIELD**

Useful constant for ERA field alignment. Used in FieldPosition of date/time formatting.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.text.DateFormat.ERA_FIELD)

### YEAR\_FIELD

public static final int **YEAR\_FIELD**

Useful constant for YEAR field alignment. Used in FieldPosition of date/time formatting.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.text.DateFormat.YEAR_FIELD)

### MONTH\_FIELD

public static final int **MONTH\_FIELD**

Useful constant for MONTH field alignment. Used in FieldPosition of date/time formatting.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.text.DateFormat.MONTH_FIELD)

### DATE\_FIELD

public static final int **DATE\_FIELD**

Useful constant for DATE field alignment. Used in FieldPosition of date/time formatting.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.text.DateFormat.DATE_FIELD)

### HOUR\_OF\_DAY1\_FIELD

public static final int **HOUR\_OF\_DAY1\_FIELD**

Useful constant for one-based HOUR\_OF\_DAY field alignment. Used in FieldPosition of date/time formatting. HOUR\_OF\_DAY1\_FIELD is used for the one-based 24-hour clock. For example, 23:59 + 01:00 results in 24:59.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.text.DateFormat.HOUR_OF_DAY1_FIELD)

### HOUR\_OF\_DAY0\_FIELD

public static final int **HOUR\_OF\_DAY0\_FIELD**

Useful constant for zero-based HOUR\_OF\_DAY field alignment. Used in FieldPosition of date/time formatting. HOUR\_OF\_DAY0\_FIELD is used for the zero-based 24-hour clock. For example, 23:59 + 01:00 results in 00:59.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.text.DateFormat.HOUR_OF_DAY0_FIELD)

### MINUTE\_FIELD

public static final int **MINUTE\_FIELD**

Useful constant for MINUTE field alignment. Used in FieldPosition of date/time formatting.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.text.DateFormat.MINUTE_FIELD)

### SECOND\_FIELD

public static final int **SECOND\_FIELD**

Useful constant for SECOND field alignment. Used in FieldPosition of date/time formatting.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.text.DateFormat.SECOND_FIELD)

### MILLISECOND\_FIELD

public static final int **MILLISECOND\_FIELD**

Useful constant for MILLISECOND field alignment. Used in FieldPosition of date/time formatting.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.text.DateFormat.MILLISECOND_FIELD)

### DAY\_OF\_WEEK\_FIELD

public static final int **DAY\_OF\_WEEK\_FIELD**

Useful constant for DAY\_OF\_WEEK field alignment. Used in FieldPosition of date/time formatting.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.text.DateFormat.DAY_OF_WEEK_FIELD)

### DAY\_OF\_YEAR\_FIELD

public static final int **DAY\_OF\_YEAR\_FIELD**

Useful constant for DAY\_OF\_YEAR field alignment. Used in FieldPosition of date/time formatting.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.text.DateFormat.DAY_OF_YEAR_FIELD)

### DAY\_OF\_WEEK\_IN\_MONTH\_FIELD

public static final int **DAY\_OF\_WEEK\_IN\_MONTH\_FIELD**

Useful constant for DAY\_OF\_WEEK\_IN\_MONTH field alignment. Used in FieldPosition of date/time formatting.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.text.DateFormat.DAY_OF_WEEK_IN_MONTH_FIELD)

### WEEK\_OF\_YEAR\_FIELD

public static final int **WEEK\_OF\_YEAR\_FIELD**

Useful constant for WEEK\_OF\_YEAR field alignment. Used in FieldPosition of date/time formatting.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.text.DateFormat.WEEK_OF_YEAR_FIELD)

### WEEK\_OF\_MONTH\_FIELD

public static final int **WEEK\_OF\_MONTH\_FIELD**

Useful constant for WEEK\_OF\_MONTH field alignment. Used in FieldPosition of date/time formatting.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.text.DateFormat.WEEK_OF_MONTH_FIELD)

### AM\_PM\_FIELD

public static final int **AM\_PM\_FIELD**

Useful constant for AM\_PM field alignment. Used in FieldPosition of date/time formatting.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.text.DateFormat.AM_PM_FIELD)

### HOUR1\_FIELD

public static final int **HOUR1\_FIELD**

Useful constant for one-based HOUR field alignment. Used in FieldPosition of date/time formatting. HOUR1\_FIELD is used for the one-based 12-hour clock. For example, 11:30 PM + 1 hour results in 12:30 AM.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.text.DateFormat.HOUR1_FIELD)

### HOUR0\_FIELD

public static final int **HOUR0\_FIELD**

Useful constant for zero-based HOUR field alignment. Used in FieldPosition of date/time formatting. HOUR0\_FIELD is used for the zero-based 12-hour clock. For example, 11:30 PM + 1 hour results in 00:30 AM.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.text.DateFormat.HOUR0_FIELD)

### TIMEZONE\_FIELD

public static final int **TIMEZONE\_FIELD**

Useful constant for TIMEZONE field alignment. Used in FieldPosition of date/time formatting.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.text.DateFormat.TIMEZONE_FIELD)

### FULL

public static final int **FULL**

Constant for full style pattern.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.text.DateFormat.FULL)

### LONG

public static final int **LONG**

Constant for long style pattern.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.text.DateFormat.LONG)

### MEDIUM

public static final int **MEDIUM**

Constant for medium style pattern.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.text.DateFormat.MEDIUM)

### SHORT

public static final int **SHORT**

Constant for short style pattern.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.text.DateFormat.SHORT)

### DEFAULT

public static final int **DEFAULT**

Constant for default style pattern. Its value is MEDIUM.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.text.DateFormat.DEFAULT)

| **Constructor Detail** |
| --- |

### DateFormat

protected **DateFormat**()

Create a new date format.

| **Method Detail** |
| --- |

### format

public final [StringBuffer](http://docs.google.com/java/lang/StringBuffer.html) **format**([Object](http://docs.google.com/java/lang/Object.html) obj,  
 [StringBuffer](http://docs.google.com/java/lang/StringBuffer.html) toAppendTo,  
 [FieldPosition](http://docs.google.com/java/text/FieldPosition.html) fieldPosition)

Overrides Format. Formats a time object into a time string. Examples of time objects are a time value expressed in milliseconds and a Date object.

**Specified by:**[format](http://docs.google.com/java/text/Format.html#format(java.lang.Object,%20java.lang.StringBuffer,%20java.text.FieldPosition)) in class [Format](http://docs.google.com/java/text/Format.html) **Parameters:**obj - must be a Number or a Date.toAppendTo - the string buffer for the returning time string.fieldPosition - keeps track of the position of the field within the returned string. On input: an alignment field, if desired. On output: the offsets of the alignment field. For example, given a time text "1996.07.10 AD at 15:08:56 PDT", if the given fieldPosition is DateFormat.YEAR\_FIELD, the begin index and end index of fieldPosition will be set to 0 and 4, respectively. Notice that if the same time field appears more than once in a pattern, the fieldPosition will be set for the first occurrence of that time field. For instance, formatting a Date to the time string "1 PM PDT (Pacific Daylight Time)" using the pattern "h a z (zzzz)" and the alignment field DateFormat.TIMEZONE\_FIELD, the begin index and end index of fieldPosition will be set to 5 and 8, respectively, for the first occurrence of the timezone pattern character 'z'. **Returns:**the string buffer passed in as toAppendTo, with formatted text appended.**See Also:**[Format](http://docs.google.com/java/text/Format.html)

### format

public abstract [StringBuffer](http://docs.google.com/java/lang/StringBuffer.html) **format**([Date](http://docs.google.com/java/util/Date.html) date,  
 [StringBuffer](http://docs.google.com/java/lang/StringBuffer.html) toAppendTo,  
 [FieldPosition](http://docs.google.com/java/text/FieldPosition.html) fieldPosition)

Formats a Date into a date/time string.

**Parameters:**date - a Date to be formatted into a date/time string.toAppendTo - the string buffer for the returning date/time string.fieldPosition - keeps track of the position of the field within the returned string. On input: an alignment field, if desired. On output: the offsets of the alignment field. For example, given a time text "1996.07.10 AD at 15:08:56 PDT", if the given fieldPosition is DateFormat.YEAR\_FIELD, the begin index and end index of fieldPosition will be set to 0 and 4, respectively. Notice that if the same time field appears more than once in a pattern, the fieldPosition will be set for the first occurrence of that time field. For instance, formatting a Date to the time string "1 PM PDT (Pacific Daylight Time)" using the pattern "h a z (zzzz)" and the alignment field DateFormat.TIMEZONE\_FIELD, the begin index and end index of fieldPosition will be set to 5 and 8, respectively, for the first occurrence of the timezone pattern character 'z'. **Returns:**the string buffer passed in as toAppendTo, with formatted text appended.

### format

public final [String](http://docs.google.com/java/lang/String.html) **format**([Date](http://docs.google.com/java/util/Date.html) date)

Formats a Date into a date/time string.

**Parameters:**date - the time value to be formatted into a time string. **Returns:**the formatted time string.

### parse

public [Date](http://docs.google.com/java/util/Date.html) **parse**([String](http://docs.google.com/java/lang/String.html) source)  
 throws [ParseException](http://docs.google.com/java/text/ParseException.html)

Parses text from the beginning of the given string to produce a date. The method may not use the entire text of the given string.

See the [parse(String, ParsePosition)](http://docs.google.com/java/text/DateFormat.html#parse(java.lang.String,%20java.text.ParsePosition)) method for more information on date parsing.

**Parameters:**source - A String whose beginning should be parsed. **Returns:**A Date parsed from the string. **Throws:** [ParseException](http://docs.google.com/java/text/ParseException.html) - if the beginning of the specified string cannot be parsed.

### parse

public abstract [Date](http://docs.google.com/java/util/Date.html) **parse**([String](http://docs.google.com/java/lang/String.html) source,  
 [ParsePosition](http://docs.google.com/java/text/ParsePosition.html) pos)

Parse a date/time string according to the given parse position. For example, a time text "07/10/96 4:5 PM, PDT" will be parsed into a Date that is equivalent to Date(837039928046).

By default, parsing is lenient: If the input is not in the form used by this object's format method but can still be parsed as a date, then the parse succeeds. Clients may insist on strict adherence to the format by calling setLenient(false).

**Parameters:**source - The date/time string to be parsedpos - On input, the position at which to start parsing; on output, the position at which parsing terminated, or the start position if the parse failed. **Returns:**A Date, or null if the input could not be parsed**See Also:**[setLenient(boolean)](http://docs.google.com/java/text/DateFormat.html#setLenient(boolean))

### parseObject

public [Object](http://docs.google.com/java/lang/Object.html) **parseObject**([String](http://docs.google.com/java/lang/String.html) source,  
 [ParsePosition](http://docs.google.com/java/text/ParsePosition.html) pos)

Parses text from a string to produce a Date.

The method attempts to parse text starting at the index given by pos. If parsing succeeds, then the index of pos is updated to the index after the last character used (parsing does not necessarily use all characters up to the end of the string), and the parsed date is returned. The updated pos can be used to indicate the starting point for the next call to this method. If an error occurs, then the index of pos is not changed, the error index of pos is set to the index of the character where the error occurred, and null is returned.

See the [parse(String, ParsePosition)](http://docs.google.com/java/text/DateFormat.html#parse(java.lang.String,%20java.text.ParsePosition)) method for more information on date parsing.

**Specified by:**[parseObject](http://docs.google.com/java/text/Format.html#parseObject(java.lang.String,%20java.text.ParsePosition)) in class [Format](http://docs.google.com/java/text/Format.html) **Parameters:**source - A String, part of which should be parsed.pos - A ParsePosition object with index and error index information as described above. **Returns:**A Date parsed from the string. In case of error, returns null. **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if pos is null.

### getTimeInstance

public static final [DateFormat](http://docs.google.com/java/text/DateFormat.html) **getTimeInstance**()

Gets the time formatter with the default formatting style for the default locale.

**Returns:**a time formatter.

### getTimeInstance

public static final [DateFormat](http://docs.google.com/java/text/DateFormat.html) **getTimeInstance**(int style)

Gets the time formatter with the given formatting style for the default locale.

**Parameters:**style - the given formatting style. For example, SHORT for "h:mm a" in the US locale. **Returns:**a time formatter.

### getTimeInstance

public static final [DateFormat](http://docs.google.com/java/text/DateFormat.html) **getTimeInstance**(int style,  
 [Locale](http://docs.google.com/java/util/Locale.html) aLocale)

Gets the time formatter with the given formatting style for the given locale.

**Parameters:**style - the given formatting style. For example, SHORT for "h:mm a" in the US locale.aLocale - the given locale. **Returns:**a time formatter.

### getDateInstance

public static final [DateFormat](http://docs.google.com/java/text/DateFormat.html) **getDateInstance**()

Gets the date formatter with the default formatting style for the default locale.

**Returns:**a date formatter.

### getDateInstance

public static final [DateFormat](http://docs.google.com/java/text/DateFormat.html) **getDateInstance**(int style)

Gets the date formatter with the given formatting style for the default locale.

**Parameters:**style - the given formatting style. For example, SHORT for "M/d/yy" in the US locale. **Returns:**a date formatter.

### getDateInstance

public static final [DateFormat](http://docs.google.com/java/text/DateFormat.html) **getDateInstance**(int style,  
 [Locale](http://docs.google.com/java/util/Locale.html) aLocale)

Gets the date formatter with the given formatting style for the given locale.

**Parameters:**style - the given formatting style. For example, SHORT for "M/d/yy" in the US locale.aLocale - the given locale. **Returns:**a date formatter.

### getDateTimeInstance

public static final [DateFormat](http://docs.google.com/java/text/DateFormat.html) **getDateTimeInstance**()

Gets the date/time formatter with the default formatting style for the default locale.

**Returns:**a date/time formatter.

### getDateTimeInstance

public static final [DateFormat](http://docs.google.com/java/text/DateFormat.html) **getDateTimeInstance**(int dateStyle,  
 int timeStyle)

Gets the date/time formatter with the given date and time formatting styles for the default locale.

**Parameters:**dateStyle - the given date formatting style. For example, SHORT for "M/d/yy" in the US locale.timeStyle - the given time formatting style. For example, SHORT for "h:mm a" in the US locale. **Returns:**a date/time formatter.

### getDateTimeInstance

public static final [DateFormat](http://docs.google.com/java/text/DateFormat.html) **getDateTimeInstance**(int dateStyle,  
 int timeStyle,  
 [Locale](http://docs.google.com/java/util/Locale.html) aLocale)

Gets the date/time formatter with the given formatting styles for the given locale.

**Parameters:**dateStyle - the given date formatting style.timeStyle - the given time formatting style.aLocale - the given locale. **Returns:**a date/time formatter.

### getInstance

public static final [DateFormat](http://docs.google.com/java/text/DateFormat.html) **getInstance**()

Get a default date/time formatter that uses the SHORT style for both the date and the time.

### getAvailableLocales

public static [Locale](http://docs.google.com/java/util/Locale.html)[] **getAvailableLocales**()

Returns an array of all locales for which the get\*Instance methods of this class can return localized instances. The returned array represents the union of locales supported by the Java runtime and by installed [DateFormatProvider](http://docs.google.com/java/text/spi/DateFormatProvider.html) implementations. It must contain at least a Locale instance equal to [Locale.US](http://docs.google.com/java/util/Locale.html#US).

**Returns:**An array of locales for which localized DateFormat instances are available.

### setCalendar

public void **setCalendar**([Calendar](http://docs.google.com/java/util/Calendar.html) newCalendar)

Set the calendar to be used by this date format. Initially, the default calendar for the specified or default locale is used.

**Parameters:**newCalendar - the new Calendar to be used by the date format

### getCalendar

public [Calendar](http://docs.google.com/java/util/Calendar.html) **getCalendar**()

Gets the calendar associated with this date/time formatter.

**Returns:**the calendar associated with this date/time formatter.

### setNumberFormat

public void **setNumberFormat**([NumberFormat](http://docs.google.com/java/text/NumberFormat.html) newNumberFormat)

Allows you to set the number formatter.

**Parameters:**newNumberFormat - the given new NumberFormat.

### getNumberFormat

public [NumberFormat](http://docs.google.com/java/text/NumberFormat.html) **getNumberFormat**()

Gets the number formatter which this date/time formatter uses to format and parse a time.

**Returns:**the number formatter which this date/time formatter uses.

### setTimeZone

public void **setTimeZone**([TimeZone](http://docs.google.com/java/util/TimeZone.html) zone)

Sets the time zone for the calendar of this DateFormat object.

**Parameters:**zone - the given new time zone.

### getTimeZone

public [TimeZone](http://docs.google.com/java/util/TimeZone.html) **getTimeZone**()

Gets the time zone.

**Returns:**the time zone associated with the calendar of DateFormat.

### setLenient

public void **setLenient**(boolean lenient)

Specify whether or not date/time parsing is to be lenient. With lenient parsing, the parser may use heuristics to interpret inputs that do not precisely match this object's format. With strict parsing, inputs must match this object's format.

**Parameters:**lenient - when true, parsing is lenient**See Also:**[Calendar.setLenient(boolean)](http://docs.google.com/java/util/Calendar.html#setLenient(boolean))

### isLenient

public boolean **isLenient**()

Tell whether date/time parsing is to be lenient.

### hashCode

public int **hashCode**()

Overrides hashCode

**Overrides:**[hashCode](http://docs.google.com/java/lang/Object.html#hashCode()) in class [Object](http://docs.google.com/java/lang/Object.html) **Returns:**a hash code value for this object.**See Also:**[Object.equals(java.lang.Object)](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)), [Hashtable](http://docs.google.com/java/util/Hashtable.html)

### equals

public boolean **equals**([Object](http://docs.google.com/java/lang/Object.html) obj)

Overrides equals

**Overrides:**[equals](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)) in class [Object](http://docs.google.com/java/lang/Object.html) **Parameters:**obj - the reference object with which to compare. **Returns:**true if this object is the same as the obj argument; false otherwise.**See Also:**[Object.hashCode()](http://docs.google.com/java/lang/Object.html#hashCode()), [Hashtable](http://docs.google.com/java/util/Hashtable.html)

### clone

public [Object](http://docs.google.com/java/lang/Object.html) **clone**()

Overrides Cloneable

**Overrides:**[clone](http://docs.google.com/java/text/Format.html#clone()) in class [Format](http://docs.google.com/java/text/Format.html) **Returns:**a clone of this instance.**See Also:**[Cloneable](http://docs.google.com/java/lang/Cloneable.html)

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/DateFormat.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/java/text/Collator.html)   [**NEXT CLASS**](http://docs.google.com/java/text/DateFormat.Field.html) | [**FRAMES**](http://docs.google.com/index.html?java/text/DateFormat.html)    [**NO FRAMES**](http://docs.google.com/DateFormat.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#2et92p0) | [FIELD](#tyjcwt) | [CONSTR](#3dy6vkm) | [METHOD](#1t3h5sf) | DETAIL: [FIELD](#17dp8vu) | [CONSTR](#1hmsyys) | [METHOD](#2grqrue) |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).