public abstract class

Summary: Nested Classes | Constants | Fields | Protected Ctors | Methods | Inherited Methods | [Expand All]

Added in API level 1

DateFormat

extends Format

java.lang.Object
↳java.text.Format
↳java.text.DateFormat

▶ Known Direct Subclasses
SimpleDateFormat

Class Overview

Formats or parses dates and times.

This class provides factories for obtaining instances configured for a specific locale. The most common subclass is `SimpleDateFormat` (/reference/java/text/SimpleDateFormat.html).

Sample Code

This code:

```
DateFormat[] formats = new DateFormat[] {
  DateFormat.getDateInstance(),
  DateFormat.getDateTimeInstance(),
  DateFormat.getTimeInstance(),
};
for (DateFormat df : formats) {
  System.out.println(df.format(new Date(0)));
  df.setTimeZone(TimeZone.getTimeZone("UTC"));
  System.out.println(df.format(new Date(0)));
}
```

Produces this output when run on an en_US device in the America/Los_Angeles time zone:

```
Dec 31, 1969
Jan 1, 1970
```

```
Dec 31, 1969 4:00:00 PM
Jan 1, 1970 12:00:00 AM
4:00:00 PM
12:00:00 AM
```

And will produce similarly appropriate localized human-readable output on any user's system. Notice how the same point in time when formatted can appear to be a different time when rendered for a different time zone. This is one reason why formatting should be left until the data will only be presented to a human. Machines should interchange "Unix time" integers.

Summary

Nested Classes	
class DateFormat.Field	The instances of this inner class are used as attribute keys and values in `AttributedCharacterIterator` that the `formatToCharacterIterator(Object)` method returns.

Constants	
int AM_PM_FIELD	FieldPosition selector for 'a' field alignment, corresponds to the AM_PM field.
int DATE_FIELD	The `FieldPosition` selector for 'd' field alignment, corresponds to the DATE field.
int DAY_OF_WEEK_FIELD	FieldPosition selector for 'E' field alignment, corresponds to the DAY_OF_WEEK field.
int DAY_OF_WEEK_IN_MONTH_FIELD	FieldPosition selector for 'F' field alignment, corresponds to the DAY_OF_WEEK_IN_MONTH field.
int DAY_OF_YEAR_FIELD	FieldPosition selector for 'D' field alignment, corresponds to the DAY_OF_YEAR field.
int DEFAULT	The format style constant defining the default format style.
int ERA_FIELD	The `FieldPosition` selector for 'G' field alignment, corresponds to the ERA field.

int FULL	The format style constant defining the full style.
int HOUR0_FIELD	The `FieldPosition` selector for 'K' field alignment, corresponding to the `HOUR` field.
int HOUR1_FIELD	FieldPosition selector for 'h' field alignment, corresponding to the `HOUR` field.
int HOUR_OF_DAY0_FIELD	The `FieldPosition` selector for 'H' field alignment, corresponds to the `HOUR_OF_DAY` field.
int HOUR_OF_DAY1_FIELD	The `FieldPosition` selector for 'k' field alignment, corresponds to the `HOUR_OF_DAY` field.
int LONG	The format style constant defining the long style.
int MEDIUM	The format style constant defining the medium style.
int MILLISECOND_FIELD	FieldPosition selector for 'S' field alignment, corresponds to the `MILLISECOND` field.
int MINUTE_FIELD	FieldPosition selector for 'm' field alignment, corresponds to the `MINUTE` field.
int MONTH_FIELD	The `FieldPosition` selector for 'M' field alignment, corresponds to the `MONTH` field.
int SECOND_FIELD	FieldPosition selector for 's' field alignment, corresponds to the `SECOND` field.
int SHORT	The format style constant defining the short style.
int TIMEZONE_FIELD	The `FieldPosition` selector for 'z' field alignment, corresponds to the `ZONE_OFFSET` and `DST_OFFSET` fields.
int WEEK_OF_MONTH_FIELD	FieldPosition selector for 'W' field alignment, corresponds to the `WEEK_OF_MONTH` field.

int WEEK_OF_YEAR_FIELD	FieldPosition selector for 'w' field alignment, corresponds to the WEEK_OF_YEAR field.
int YEAR_FIELD	The FieldPosition selector for 'y' field alignment, corresponds to the YEAR field.

Fields

protected Calendar calendar	The calendar that this DateFormat uses to format a number representing a date.
protected NumberFormat numberFormat	The number format used to format a number.

Protected Constructors

DateFormat ()

Constructs a new instance of DateFormat.

Public Methods

clone ()

Object — Returns a new instance of DateFormat with the same properties.

equals (Object object)

boolean — Compares this date format with the specified object and indicates if they are equal.

format (Date date, StringBuffer buffer, FieldPosition field)

abstract StringBuffer — Formats the specified date as a string using the pattern of this date format and appends the string to the specified string buffer.

format (Object object, StringBuffer buffer, FieldPosition field)

final StringBuffer — Formats the specified object as a string using the pattern of this date format and appends the string to the specified string buffer.

format (Date date)

final String — Formats the specified date using the rules of this date format.

getAvailableLocales ()

static Locale[] — Returns an array of locales for which custom DateFormat instances are available.

getCalendar ()

Calendar — Returns the calendar used by this DateFormat.

getDateInstance (int style)

final static DateFormat — Returns a DateFormat instance for formatting and parsing dates in the specified style for the user's default

	locale.
final static DateFormat	getDateInstance (int style, Locale locale) Returns a `DateFormat` instance for formatting and parsing dates in the specified style for the specified locale.
final static DateFormat	getDateInstance () Returns a `DateFormat` instance for formatting and parsing dates in the DEFAULT style for the default locale.
final static DateFormat	getDateTimeInstance (int dateStyle, int timeStyle, Locale locale) Returns a `DateFormat` instance for formatting and parsing dates and time values in the specified styles for the specified locale.
final static DateFormat	getDateTimeInstance (int dateStyle, int timeStyle) Returns a `DateFormat` instance for formatting and parsing of both dates and time values in the manner appropriate for the user's default locale.
final static DateFormat	getDateTimeInstance () Returns a `DateFormat` instance for formatting and parsing dates and time values in the DEFAULT style for the default locale.
final static DateFormat	getInstance () Returns a `DateFormat` instance for formatting and parsing dates and times in the SHORT style for the default locale.
NumberFormat	getNumberFormat () Returns the `NumberFormat` used by this `DateFormat`.
final static DateFormat	getTimeInstance (int style) Returns a `DateFormat` instance for formatting and parsing time values in the specified style for the user's default locale.
final static DateFormat	getTimeInstance () Returns a `DateFormat` instance for formatting and parsing time values in the DEFAULT style for the default locale.
final static DateFormat	getTimeInstance (int style, Locale locale) Returns a `DateFormat` instance for formatting and parsing time values in the specified style for the specified locale.
TimeZone	getTimeZone () Returns the time zone of this date format's calendar.
int	hashCode () Returns an integer hash code for this object.

	isLenient ()
boolean	Indicates whether the calendar used by this date format is lenient.
	parse (String string)
Date	Parses a date from the specified string using the rules of this date format.
	parse (String string, ParsePosition position)
abstract Date	Parses a date from the specified string starting at the index specified by `position`.
	parseObject (String string, ParsePosition position)
Object	Parses a date from the specified string starting at the index specified by `position`.
void	setCalendar (Calendar cal) Sets the calendar used by this date format.
void	setLenient (boolean value) Specifies whether or not date/time parsing shall be lenient.
void	setNumberFormat (NumberFormat format) Sets the `NumberFormat` used by this date format.
	setTimeZone (TimeZone timezone)
void	Sets the time zone of the calendar used by this date format.

Inherited Methods [Expand]

- From class java.text.Format
- From class java.lang.Object

Constants

public static final int **AM_PM_FIELD** Added in API level 1

FieldPosition selector for 'a' field alignment, corresponds to the `AM_PM (/reference/java/util/Calendar.html#AM_PM)` field.

Constant Value: 14 (0x0000000e)

public static final int **DATE_FIELD** Added in API level 1

The `FieldPosition` selector for 'd' field alignment, corresponds to the `DATE (/reference/java/util/Calendar.html#DATE)` field.

Constant Value: 3 (0x00000003)

public static final int **DAY_OF_WEEK_FIELD** Added in API level 1

FieldPosition selector for 'E' field alignment, corresponds to the DAY_OF_WEEK (/reference/java/util/Calendar.html#DAY_OF_WEEK) field.

Constant Value: 9 (0x00000009)

public static final int **DAY_OF_WEEK_IN_MONTH_FIELD** Added in API level 1

FieldPosition selector for 'F' field alignment, corresponds to the DAY_OF_WEEK_IN_MONTH (/reference/java/util/Calendar.html#DAY_OF_WEEK_IN_MONTH) field.

Constant Value: 11 (0x0000000b)

public static final int **DAY_OF_YEAR_FIELD** Added in API level 1

FieldPosition selector for 'D' field alignment, corresponds to the DAY_OF_YEAR (/reference/java/util/Calendar.html#DAY_OF_YEAR) field.

Constant Value: 10 (0x0000000a)

public static final int **DEFAULT** Added in API level 1

The format style constant defining the default format style. The default is MEDIUM.

Constant Value: 2 (0x00000002)

public static final int **ERA_FIELD** Added in API level 1

The `FieldPosition` selector for 'G' field alignment, corresponds to the ERA (/reference/java/util/Calendar.html#ERA) field.

Constant Value: 0 (0x00000000)

public static final int **FULL** Added in API level 1

The format style constant defining the full style.

Constant Value: 0 (0x00000000)

public static final int **HOUR0_FIELD** Added in API level 1

The `FieldPosition` selector for 'K' field alignment, corresponding to the HOUR (/reference/java/util/Calendar.html#HOUR) field.

Constant Value: 16 (0x00000010)

public static final int **HOUR1_FIELD**

Added in API level 1

FieldPosition selector for 'h' field alignment, corresponding to the HOUR (/reference/java/util/Calendar.html#HOUR) field.

Constant Value: 15 (0x0000000f)

public static final int **HOUR_OF_DAY0_FIELD**

Added in API level 1

The `FieldPosition` selector for 'H' field alignment, corresponds to the HOUR_OF_DAY (/reference/java/util/Calendar.html#HOUR_OF_DAY) field. `HOUR_OF_DAY0_FIELD` is used for the zero-based 24-hour clock. For example, 23:59 + 01:00 results in 00:59.

Constant Value: 5 (0x00000005)

public static final int **HOUR_OF_DAY1_FIELD**

Added in API level 1

The `FieldPosition` selector for 'k' field alignment, corresponds to the HOUR_OF_DAY (/reference/java/util/Calendar.html#HOUR_OF_DAY) field. `HOUR_OF_DAY1_FIELD` is used for the one-based 24-hour clock. For example, 23:59 + 01:00 results in 24:59.

Constant Value: 4 (0x00000004)

public static final int **LONG**

Added in API level 1

The format style constant defining the long style.

Constant Value: 1 (0x00000001)

public static final int **MEDIUM**

Added in API level 1

The format style constant defining the medium style.

Constant Value: 2 (0x00000002)

public static final int **MILLISECOND_FIELD**

Added in API level 1

FieldPosition selector for 'S' field alignment, corresponds to the MILLISECOND (/reference/java/util/Calendar.html#MILLISECOND) field.

Constant Value: 8 (0x00000008)

public static final int **MINUTE_FIELD**

Added in API level 1

FieldPosition selector for 'm' field alignment, corresponds to the MINUTE (/reference/java/util/Calendar.html#MINUTE) field.

Constant Value: 6 (0x00000006)

public static final int **MONTH_FIELD**

Added in API level 1

The FieldPosition selector for 'M' field alignment, corresponds to the MONTH (/reference/java/util/Calendar.html#MONTH) field.

Constant Value: 2 (0x00000002)

public static final int **SECOND_FIELD**

Added in API level 1

FieldPosition selector for 's' field alignment, corresponds to the SECOND (/reference/java/util/Calendar.html#SECOND) field.

Constant Value: 7 (0x00000007)

public static final int **SHORT**

Added in API level 1

The format style constant defining the short style.

Constant Value: 3 (0x00000003)

public static final int **TIMEZONE_FIELD**

Added in API level 1

The FieldPosition selector for 'z' field alignment, corresponds to the ZONE_OFFSET (/reference/java/util/Calendar.html#ZONE_OFFSET) and DST_OFFSET (/reference/java/util/Calendar.html#DST_OFFSET) fields.

Constant Value: 17 (0x00000011)

public static final int **WEEK_OF_MONTH_FIELD**

Added in API level 1

FieldPosition selector for 'W' field alignment, corresponds to the WEEK_OF_MONTH (/reference/java/util/Calendar.html#WEEK_OF_MONTH) field.

Constant Value: 13 (0x0000000d)

public static final int **WEEK_OF_YEAR_FIELD**

Added in API level 1

FieldPosition selector for 'w' field alignment, corresponds to the WEEK_OF_YEAR (/reference/java/util/Calendar.html#WEEK_OF_YEAR) field.

Constant Value: 12 (0x0000000c)

public static final int **YEAR_FIELD** Added in API level 1

The `FieldPosition` selector for 'y' field alignment, corresponds to the YEAR (/reference/java/util/Calendar.html#YEAR) field.

Constant Value: 1 (0x00000001)

Fields

protected Calendar **calendar** Added in API level 1

The calendar that this `DateFormat` uses to format a number representing a date.

protected NumberFormat **numberFormat** Added in API level 1

The number format used to format a number.

Protected Constructors

protected **DateFormat** () Added in API level 1

Constructs a new instance of `DateFormat`.

Public Methods

public Object **clone** () Added in API level 1

Returns a new instance of `DateFormat` with the same properties.

Returns
a shallow copy of this format.

public boolean **equals** (Object object) Added in API level 1

Compares this date format with the specified object and indicates if they are equal.

Parameters

object	the object to compare with this date format.

Returns

`true` if `object` is a `DateFormat` object and it has the same properties as this date format; `false` otherwise.

See Also

`hashCode()`

public abstract StringBuffer **format** (Date date, StringBuffer buffer, FieldPosition field) Added in API level 1

Formats the specified date as a string using the pattern of this date format and appends the string to the specified string buffer.

If the `field` member of `field` contains a value specifying a format field, then its `beginIndex` and `endIndex` members will be updated with the position of the first occurrence of this field in the formatted text.

Parameters

date	the date to format.
buffer	the target string buffer to append the formatted date/time to.
field	on input: an optional alignment field; on output: the offsets of the alignment field in the formatted text.

Returns

the string buffer.

public final StringBuffer **format** (Object object, StringBuffer buffer, FieldPosition field) Added in API level 1

Formats the specified object as a string using the pattern of this date format and appends the string to the specified string buffer.

If the `field` member of `field` contains a value specifying a format field, then its `beginIndex` and `endIndex` members will be updated with the position of the first occurrence of this field in the formatted text.

Parameters

object	the source object to format, must be a `Date` or a `Number`. If `object` is a number then a date is constructed using the `longValue()` of the number.
buffer	the target string buffer to append the formatted date/time to.

field on input: an optional alignment field; on output: the offsets of the alignment field in the formatted text.

Returns

the string buffer.

Throws

IllegalArgumentException if `object` is neither a `Date` nor a `Number` instance.

public final String **format** (Date date)

Added in API level 1

Formats the specified date using the rules of this date format.

Parameters

date the date to format.

Returns

the formatted string.

public static Locale[] **getAvailableLocales** ()

Added in API level 1

Returns an array of locales for which custom `DateFormat` instances are available.

Note that Android does not support user-supplied locale service providers.

public Calendar **getCalendar** ()

Added in API level 1

Returns the calendar used by this `DateFormat`.

Returns

the calendar used by this date format.

public static final DateFormat **getDateInstance** (int style)

Added in API level 1

Returns a `DateFormat` instance for formatting and parsing dates in the specified style for the user's default locale. See "Be wary of the default locale (../util/Locale.html#default_locale)".

Parameters

style one of SHORT, MEDIUM, LONG, FULL, or DEFAULT.

Returns

the `DateFormat` instance for `style` and the default locale.

Throws

IllegalArgumentException	if `style` is not one of SHORT, MEDIUM, LONG, FULL, or DEFAULT.

public static final DateFormat **getDateInstance** (int style, Locale locale) Added in API level 1

Returns a `DateFormat` instance for formatting and parsing dates in the specified style for the specified locale.

Parameters

style	one of SHORT, MEDIUM, LONG, FULL, or DEFAULT.
locale	the locale.

Returns

the `DateFormat` instance for `style` and `locale`.

Throws

IllegalArgumentException	if `style` is not one of SHORT, MEDIUM, LONG, FULL, or DEFAULT.

public static final DateFormat **getDateInstance** () Added in API level 1

Returns a `DateFormat` instance for formatting and parsing dates in the DEFAULT style for the default locale.

Returns

the `DateFormat` instance for the default style and locale.

public static final DateFormat **getDateTimeInstance** (int dateStyle, int timeStyle, Locale locale) Added in API level 1

Returns a `DateFormat` instance for formatting and parsing dates and time values in the specified styles for the specified locale.

Parameters

dateStyle	one of SHORT, MEDIUM, LONG, FULL, or DEFAULT.
timeStyle	one of SHORT, MEDIUM, LONG, FULL, or DEFAULT.
locale	the locale.

Returns

the `DateFormat` instance for `dateStyle`, `timeStyle` and

`locale`.

Throws

IllegalArgumentException	if `dateStyle` or `timeStyle` is not one of SHORT, MEDIUM, LONG, FULL, or DEFAULT.

public static final DateFormat
getDateTimeInstance (int dateStyle, int timeStyle) Added in API level 1

Returns a `DateFormat` instance for formatting and parsing of both dates and time values in the manner appropriate for the user's default locale. See "Be wary of the default locale (../util/Locale.html#default_locale)".

Parameters

dateStyle	one of SHORT, MEDIUM, LONG, FULL, or DEFAULT.
timeStyle	one of SHORT, MEDIUM, LONG, FULL, or DEFAULT.

Returns

the `DateFormat` instance for `dateStyle`, `timeStyle` and the default locale.

Throws

IllegalArgumentException	if `dateStyle` or `timeStyle` is not one of SHORT, MEDIUM, LONG, FULL, or DEFAULT.

public static final DateFormat
getDateTimeInstance () Added in API level 1

Returns a `DateFormat` instance for formatting and parsing dates and time values in the DEFAULT style for the default locale.

Returns

the `DateFormat` instance for the default style and locale.

public static final DateFormat **getInstance** () Added in API level 1

Returns a `DateFormat` instance for formatting and parsing dates and times in the SHORT style for the default locale.

Returns

the `DateFormat` instance for the SHORT style and default locale.

public NumberFormat **getNumberFormat** ()

Returns the `NumberFormat` used by this `DateFormat`.

Returns

the `NumberFormat` used by this date format.

public static final DateFormat **getTimeInstance** (int style)

Added in API level 1

Returns a `DateFormat` instance for formatting and parsing time values in the specified style for the user's default locale. See "Be wary of the default locale (../util/Locale.html#default_locale)".

Parameters

style	one of SHORT, MEDIUM, LONG, FULL, or DEFAULT.

Returns

the `DateFormat` instance for `style` and the default locale.

Throws

IllegalArgumentException	if `style` is not one of SHORT, MEDIUM, LONG, FULL, or DEFAULT.

public static final DateFormat **getTimeInstance** ()

Added in API level 1

Returns a `DateFormat` instance for formatting and parsing time values in the DEFAULT style for the default locale.

Returns

the `DateFormat` instance for the default style and locale.

public static final DateFormat **getTimeInstance** (int style, Locale locale)

Added in API level 1

Returns a `DateFormat` instance for formatting and parsing time values in the specified style for the specified locale.

Parameters

style	one of SHORT, MEDIUM, LONG, FULL, or DEFAULT.
locale	the locale.

Returns

the `DateFormat` instance for `style` and `locale`.

Throws

IllegalArgumentException	if `style` is not one of SHORT, MEDIUM, LONG, FULL, or

DEFAULT.

public TimeZone **getTimeZone** ()

Added in API level 1

Returns the time zone of this date format's calendar.

Returns

the time zone of the calendar used by this date format.

public int **hashCode** ()

Added in API level 1

Returns an integer hash code for this object. By contract, any two objects for which `equals(Object)` (/reference/java/lang/Object.html#equals(java.lang.Object)) returns `true` must return the same hash code value. This means that subclasses of `Object` usually override both methods or neither method.

Note that hash values must not change over time unless information used in equals comparisons also changes.

See Writing a correct `hashCode` method (/reference/java/lang/Object.html#writing_hashCode) if you intend implementing your own `hashCode` method.

Returns

this object's hash code.

public boolean **isLenient** ()

Added in API level 1

Indicates whether the calendar used by this date format is lenient.

Returns

`true` if the calendar is lenient; `false` otherwise.

public Date **parse** (String string)

Added in API level 1

Parses a date from the specified string using the rules of this date format.

Parameters

string the string to parse.

Returns

the `Date` resulting from the parsing.

Throws

ParseException if an error occurs during parsing.

public abstract Date **parse** (String string, ParsePosition position) Added in API level 1

Parses a date from the specified string starting at the index specified by `position`. If the string is successfully parsed then the index of the `ParsePosition` is updated to the index following the parsed text. On error, the index is unchanged and the error index of `ParsePosition` is set to the index where the error occurred.

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

string	the string to parse.
position	input/output parameter, specifies the start index in `string` from where to start parsing. If parsing is successful, it is updated with the index following the parsed text; on error, the index is unchanged and the error index is set to the index where the error occurred.

Returns

the date resulting from the parse, or `null` if there is an error.

public Object **parseObject** (String string, ParsePosition position) Added in API level 1

Parses a date from the specified string starting at the index specified by `position`. If the string is successfully parsed then the index of the `ParsePosition` is updated to the index following the parsed text. On error, the index is unchanged and the error index of `ParsePosition` is set to the index where the error occurred.

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

string	the string to parse.
position	input/output parameter, specifies the start index in `string` from where to start parsing. If parsing is successful, it is updated with the index following the parsed text; on error, the index is unchanged

and the error index is set to the index where the error occurred.

Returns

the date resulting from the parsing, or `null` if there is an error.

public void **setCalendar** (Calendar cal)

Added in API level 1

Sets the calendar used by this date format.

Parameters

cal the new calendar.

public void **setLenient** (boolean value)

Added in API level 1

Specifies whether or not date/time parsing shall 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

value `true` to set the calendar to be lenient, `false` otherwise.

public void **setNumberFormat** (NumberFormat format)

Added in API level 1

Sets the `NumberFormat` used by this date format.

Parameters

format the new number format.

public void **setTimeZone** (TimeZone timezone)

Added in API level 1

Sets the time zone of the calendar used by this date format.

Parameters

timezone the new time zone.