	Date Formatting
d	Day of the month, 2 digits with leading zeros *Ex: 01 to 31*
D	A textual representation of a day, three letters *Ex: Mon through Sun*
j	Day of the month without leading zeros *Ex: 1 to 31*
l	A full textual representation of the day of the week *Ex: Sunday through Saturday*
N	ISO-8601 numeric representation of the day of the week *Ex: 1 (for Monday) through 7 (for Sunday)*
S	English ordinal suffix for the day of the month, 2 characters *Ex: st, nd, rd or th. Works well with j*
w	Numeric representation of the day of the week *Ex: 0 (for Sunday) through 6 (for Saturday)*
z	The day of the year (starting from 0) *Ex: 0 through 365*
W	ISO-8601 week number of year, weeks starting on Monday *Ex: 42 (the 42nd week in the year)*
F	A full textual representation of a month, such as January or March *Ex: January through December*
m	Numeric representation of a month, with leading zeros *Ex: 01 through 12*
M	A short textual representation of a month, three letters *Ex: Jan through Dec*
n	Numeric representation of a month, without leading zeros *Ex: 1 through 12*
t	Number of days in the given month *Ex: 28 through 31*
L	Whether it's a leap year *Ex: 1 if it is a leap year, 0 otherwise.*
o	ISO-8601 week-numbering year. This has the same value as Y, except that if the ISO week number (W) belongs to the previous or next year, that year is used instead. *Ex: 1999 or 2003*
Y	A full numeric representation of a year, 4 digits *Ex: 1999 or 2003*
y	A two digit representation of a year *Ex: 99 or 03*
a	Lowercase Ante meridiem and Post meridiem *Ex: am or pm*
A	Uppercase Ante meridiem and Post meridiem *Ex: AM or PM*
B	Swatch Internet time *Ex: 000 through 999*
g	12-hour format of an hour without leading zeros *Ex: 1 through 12*
G	24-hour format of an hour without leading zeros *Ex: 0 through 23*
h	12-hour format of an hour with leading zeros *Ex: 01 through 12*
H	24-hour format of an hour with leading zeros *Ex: 00 through 23*
i	Minutes with leading zeros *Ex: 00 to 59*
s	Seconds, with leading zeros *Ex: 00 through 59*
u	Microseconds. date() will always generate 000000 since it takes an integer parameter. *Ex: 654321*
v	Milliseconds. date() will always generate 000000 since it takes an integer parameter. *Ex: 654*
e	Timezone identifier *Ex: UTC, GMT, Atlantic/Azores*
I	Whether or not the date is in daylight saving time *Ex: 1 if Daylight Saving Time, 0 otherwise.*
O	Difference to Greenwich time (GMT) in hours *Ex: +0200*
P	Difference to Greenwich time (GMT) with colon between hours and minutes *Ex: +02:00*
T	Timezone abbreviation *Ex: EST, MDT ...*
Z	Timezone offset in seconds. The offset for timezones west of UTC is always negative, and for those east of UTC is always positive. *Ex: -43200 through 50400*
c	ISO 8601 date *Ex: 2004-02-12T15:19:21+00:00*
r	RFC 2822 formatted date *Ex: Thu, 21 Dec 2000 16:01:07 +0200*
U	Seconds since the Unix Epoch (January 1 1970 00:00:00 GMT)