

C library function - strftime()

Description

The C library function **size_t strftime(char *str, size_t maxsize, const char *format, const struct tm *timeptr)** formats the time represented in the structure **timeptr** according to the formatting rules defined in **format** and stored into **str**.

Declaration

Following is the declaration for strftime() function.

```
size_t strftime(char *str, size_t maxsize, const char *format, const struct tm *timeptr)
```

Parameters

- **str** – This is the pointer to the destination array where the resulting C string is copied.
- **maxsize** – This is the maximum number of characters to be copied to str.
- **format** – This is the C string containing any combination of regular characters and special format specifiers. These format specifiers are replaced by the function to the corresponding values to represent the time specified in tm. The format specifiers are –

Specifier	Replaced By	Example
%a	Abbreviated weekday name	Sun
%A	Full weekday name	Sunday
%b	Abbreviated month name	Mar
%B	Full month name	March
%c	Date and time representation	Sun Aug 19 02:56:02 2012
%d	Day of the month (01-31)	19
%H	Hour in 24h format (00-23)	14
%I	Hour in 12h format (01-12)	05
%j	Day of the year (001-366)	231
%m	Month as a decimal number (01-12)	08
%M	Minute (00-59)	55
%p	AM or PM designation	PM
%S	Second (00-61)	02
%U	Week number with the first Sunday as the first day of week one (00-53)	33
%w	Weekday as a decimal number with Sunday as 0 (0-6)	4
%W	Week number with the first Monday as the first day of week one (00-53)	34
%x	Date representation	08/19/12
%X	Time representation	02:50:06
%y	Year, last two digits (00-99)	01
%Y	Year	2012
%Z	Timezone name or abbreviation	CDT
%%	A % sign	%

- **timeptr** – This is the pointer to a tm structure that contains a calendar time broken down into its components as shown below –

```
struct tm {
    int tm_sec;           /* seconds, range 0 to 59 */
    int tm_min;           /* minutes, range 0 to 59 */
    int tm_hour;          /* hours, range 0 to 23 */
    int tm_mday;          /* day of the month, range 1 to 31 */
    int tm_mon;           /* month, range 0 to 11 */
    int tm_year;          /* The number of years since 1900 */
    int tm_wday;          /* day of the week, range 0 to 6 */
    int tm_yday;          /* day in the year, range 0 to 365 */
    int tm_isdst;         /* daylight saving time */
};
```

Return Value

If the resulting C string fits in less than size characters (which includes the terminating null-character), the total number of characters copied to str (not including the terminating null-character) is returned otherwise, it returns zero.

Example

The following example shows the usage of strftime() function.

[Live Demo](#)

```
#include <stdio.h>
#include <time.h>

int main () {
    time_t rawtime;
    struct tm *info;
    char buffer[80];

    time( &rawtime );

    info = localtime( &rawtime );

    strftime(buffer,80,"%x - %I:%M%p", info);
    printf("Formatted date & time : |%s|\n", buffer );

    return(0);
}
```

Let us compile and run the above program that will produce the following result –

```
Formatted date & time : |08/23/12 - 12:40AM|
```