Time

Inherits From: Object

Declared In: Time.h

Class Description

A Time class stores a number of time related values. Years, months, weeks, days, hours, seconds, and microseconds are stored. The main use of these is to store local times such as the date of creation or the date an event will take place. The base storage of the Time class is the **time_t** variable. The limited range of the time_t data type applies to the Time class (1970 to 2037 ± a day).

The time can be accessed and set as a unit or in pieces through **set...**, **add...**, and **subtract...** methods. The time can be set as a one unit with **setTime_t:** or from another Time object with **setTime:**. Individual parts of the

time are set with the **set...** methods. The names of days and months can be retrieved in both full form and abbreviated form. Abbreviated mode is set through **arbMode:**.

Instance Variables

char* **timeString**;
time_t **myTime**;
struct tm* **timesPtr**;
BOOL **theAbrMode**;
char* **mNames[12]**;
char* **mAbrNames[12]**;
char* **dNames[7]**;
char* **dAbrNames[7]**;

timeString	A string with the current time.
myTime	The current time in seconds since 00:00:00 Jan 1, 1970.
timesPtr	The time structure that stores month, day, minute, etc.

	TimesPtr needs to be updated (calculated) from myTime.
theAbrMode	Determines if names day/month names are 3 letter abbreviations.
mNames	The month names. First letter is capitalized.
mAbrNames	The abbreviated month names. First letter is capitalized.
dNames	The day names. First letter is capitalized.
dAbrNames	The abbreviated day names. First letter is capitalized.

Method Types

Initializing and freeing a Time instance	
	- free
	- init
	- initWithCurrentTime
	- initNames
Textual values	- abrMode
	- abrMode:

- indexOfDayName:
- nameOfDay:
- indexOfMonthName:
- nameOfMonth:
- timeString
- timeZone

Retrieving values

- getTime_t
- leapYear:
- daysInYear:
- daysInMonth:
- dayOfMonth:
- dayOfWeek
- dayOfYear
- weekOfMonth
- year
- month
- week
- day
- hour
- minute
- second
- microSecond

Setting values	- _syncTimesStruct - resetTimeFromTM - setToCurrentTime - setTime: - setTime_t: - setYears: - setMonths: - setWeeks: - setDays: - setHours: - setMinutes: - setSeconds: - setMicroSeconds:
Adding/Subtracting values	- [add][subtract]Time: - [add][subtract]Time_t: - [add][subtract]Years: - [add][subtract]Months: - [add][subtract]Weeks: - [add][subtract]Days: - [add][subtract]Hours: - [add][subtract]Minutes: - [add][subtract]Seconds:

- [add][subtract]MicroSeconds:

Instance Methods

free

- **free**

Frees the memory allocated for the name strings and **timeString**.

init

- (float)**abrMode**

Sets **timeString** to NULL, **myTime** (time_t) to 0, initializes the names through **initNames,** and syncs the tm time structure.

initWithCurrentTime

- **initWithCurrentTime**

Does a normal **init** then calls **setToCurrentTime**.

initNames

- **initNames**

Allocates name space and sets up name strings.

abrMode

- (BOOL)**abrMode**
- (BOOL)**abrMode:**(BOOL)*mode*

Returns or sets the if abbreviated names should be returned for the **nameOf...** methods.

indexOfDayName:

- (long)**indexOfDayName:** (const char *)*dayName*

Returns the index of a day given its name string.

nameOfDay:

- (const char *)**nameOfDay:**(long)*dayIndex*
- (const char *)**nameOfDay**

Returns the name string of the current day or a specified day.

indexOfMonthName:

- (long)**indexOfMonthName:**(const char *)*monthName*

Returns the index of a month given its name string.

nameOfMonth:

- (const char *)**nameOfMonth:**(long)*monthIndex*
- (const char *)**nameOfMonth**

Returns the name string of the current month or a specified month.

timeString

- (const char *)**timeString**

Returns a 26 character ASCII string that has the year, month, day, hour, minute, and second.

timeZone

- (const char *)**timeZone**

Returns a 3 character string representing the current time zone.

getTime_t

- (const time_t *)**getTime_t**

Returns a pointer to **myTime**.

leapYear:

- (BOOL)**leapYear:**(long)*year*
- (BOOL)**leapYear**

Returns YES if the current year or specified year is a leap year.

daysInYear:

- (long)**daysInYear:**(long)*year*
- (long)**daysInYear**

Returns the number of days in the current year or in a specified year.

daysInMonth:

- (long)**daysInMonth:**(long)*monthIndex* forYear:(long)*year*
- (long)**daysInMonth:**(long)*monthIndex*
- (long)**daysInMonth**

Returns the number of days in the current month, number of days in a specified month in the current year, or in

a specified month and year.

dayOfMonth
- (long)**dayOfMonth**

Returns day of the month (1-31).

dayOfWeek
- (long)**dayOfWeek**

Returns days since Sunday (1-7).

dayOfYear
- (long)**dayOfYear**

Returns days of year (1-366). Identical to **year**.

weekOfMonth
- (long)**weekOfMonth**

Returns the week of the current month (1-4).

year
- (long)**year**

Returns the current year (1970-2037).

month
- (long)**month**

Returns months since January (1-12).

week
- (long)**week**

Returns week of current year (1-52).

day
- (long)**day**

Returns day of year (1-366).

hour
- (long)**hour**

Returns hours since midnight (0-23).

minute
- (long)**minute**

Returns minutes after the hour (0-59).

second
- (long)**second**

Returns seconds after the minute (0-59).

microSecond
- (long)**microSecond**

Returns microseconds after the second (0-99).

_syncTimeStruct
- **_syncTimeStruct**

Updates the tm struct with the time_t in **myTime.** Returns **self**.

resetTimeFromTM
- **resetTimeFromTM**

Updates **myTime** with the tm struct **timesPtr**. Returns **self**.

setToCurrentTime
- **setToCurrentTime**

Updates the time to the current local time. Returns **self**.

[set][add][subtract]Time:
- **[set][add][subtract]Time:**(Time *)*aTimeObj*

set Sets the current time to *aTimeObj*.
add Adds the time in *aTimeObj* to the current time.
subtract Subtracts the time in *aTimeObj* from the current time.

[set][add][subtract]Time_t:
- **[set][add][subtract]Time_t:**(time_t)*num*

set Sets the current time to *num*.
add Adds the time in *num* to the current time.

subtract Subtracts the time in *num* from the current time.

[set][add][subtract]Years:

- **[set][add][subtract]Years:**(long)*num*

set Sets the current years to *num*.
add Adds the years in *num* to the current years.
subtract Subtracts the years in *num* from the current years.

[set][add][subtract]Months:

- **[set][add][subtract]Months:**(long)*num*

set Sets the current months to *num*.
add Adds the months in *num* to the current months.
subtract Subtracts the months in *num* from the current months.

A bug of the month addition and subtraction is that if you attempt to add/subtract months when the numerical day is not in one of the in between months, it will wrap the days. For instance, if one month is added to January 31, 1970, the new date will be March 3, 1970 This problem also occurs when the day is not going to wrap. This happens occasionally such as the fourth day of the month going between October and November. This is because of some weird problem with the **mktime()** when it converts the tm structure to a time_t.

[set][add][subtract]Weeks:

- **[set][add][subtract]Weeks:**(long)*num*

set Sets the current weeks to *num*.
add Adds the weeks in *num* to the current weeks.
subtract Subtracts the weeks in *num* from the current weeks.

[set][add][subtract]Days:

- **[set][add][subtract]Days:**(long)*num*

set Sets the current days to *num*.
add Adds the days in *num* to the current days.
subtract Subtracts the days in *num* from the current days.

[set][add][subtract]Hours:

- **[set][add][subtract]Hours:**(long)*num*

set Sets the current hours to *num*.
add Adds the hours in *num* to the current hours.
subtract Subtracts the hours in *num* from the current hours.

[set][add][subtract]Minutes:

- **[set][add][subtract]Minutes:**(long)*num*

set Sets the current minutes to *num*.
add Adds the minutes in *num* to the current minutes.
subtract Subtracts the minutes in *num* from the current minutes.

[set][add][subtract]Seconds:
- **[set][add][subtract]Seconds:**(long)*num* :(long)*microSeconds*
- **[set][add][subtract]Seconds:**(long)*num*

set Sets the current seconds to *num*.
add Adds the seconds in *num* to the current seconds.
subtract Subtracts the seconds in *num* from the current seconds.

[set][add][subtract]MicroSeconds:
- **[set][add][subtract]MicroSeconds:**(long)*num*

set Sets the current microseconds to *num*.
add Adds the microseconds in *num* to the current microseconds.
subtract Subtracts the microseconds in *num* from the current microseconds.