C LIBRARY - <TIME.H>

The **time.h** header defines four variable types, two macro and various functions for manipulating date and time.

Library Variables

Following are the variable types defined in the header time.h:

S.N.	Variable & Description
1	**size_t** This is the unsigned integral type and is the result of the **sizeof** keyword.
2	**clock_t** This is a type suitable for storing the processor time.
3	**time_t is** This is a type suitable for storing the calendar time.
4	**struct tm** This is a structure used to hold the time and date.

The tm structure has the following definition:

```
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             */
};
```

Library Macros

Following are the macros defined in the header time.h:

S.N.	Macro & Description
1	**NULL** This macro is the value of a null pointer constant.

2	**CLOCKS_PER_SEC** This macro represents the number of processor clocks per second.

Library Functions

Following are the functions defined in the header time.h:

S.N.	Function & Description
1	char *asctime$const struct tm * timeptr$ Returns a pointer to a string which represents the day and time of the structure timeptr.
2	clock_t clock$void$ Returns the processor clock time used since the beginning of an implementation defined era $normally the beginning of the program$.
3	char *ctime$const time_t * timer$ Returns a string representing the localtime based on the argument timer.
4	double difftime$time_t time1, time_t time2$ Returns the difference of seconds between time1 and time2 $time1 - time2$.
5	struct tm *gmtime$const time_t * timer$ The value of timer is broken up into the structure tm and expressed in Coordinated Universal Time UTC also known as Greenwich Mean Time GMT.
6	struct tm *localtime$const time_t * timer$ The value of timer is broken up into the structure tm and expressed in the local time zone.
7	time_t mktime$struct tm * timeptr$ Converts the structure pointed to by timeptr into a time_t value according to the local time zone.
8	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.

9	time_t time$time_t * timer$ Calculates the current calender time and encodes it into time_t format.