

EUPHORIA - DATE & TIME

Euphoria has a library routine that returns the date and time to your program.

The date Method:

The date method returns a sequence value composed of eight atom elements. Following example explains it in detail –

```
#!/home/euphoria-4.0b2/bin/eui

integer curr_year, curr_day, curr_day_of_year, curr_hour,
       curr_minute, curr_second
sequence system_date, word_week, word_month, notation,
       curr_day_of_week, curr_month
word_week = {"Sunday",
             "Monday",
             "Tuesday",
             "Wednesday",
             "Thursday",
             "Friday",
             "Saturday"}
word_month = {"January", "February",
              "March", "April", "May",
              "June", "July", "August",
              "September", "October",
              "November", "December"}
-- Get current system date.
system_date = date()

-- Now take individual elements
curr_year = system_date[1] + 1900
curr_month = word_month[system_date[2]]
curr_day = system_date[3]
curr_hour = system_date[4]
curr_minute = system_date[5]
curr_second = system_date[6]
curr_day_of_week = word_week[system_date[7]]
curr_day_of_year = system_date[8]

if curr_hour >= 12 then
    notation = "p.m."
else
    notation = "a.m."
end if

if curr_hour > 12 then
    curr_hour = curr_hour - 12
end if
if curr_hour = 0 then
    curr_hour = 12
end if

puts(1, "\nHello Euphoria!\n\n")
printf(1, "Today is %s, %s %d, %d.\n",
       {curr_day_of_week, curr_month,
        curr_day, curr_year})

printf(1, "The time is %.2d:%.2d:%.2d %s\n",
       {curr_hour, curr_minute,
        curr_second, notation})

printf(1, "It is %3d days into the current year.\n",
       {curr_day_of_year})
```

This would produce following result on your standard screen –

```
Hello Euphoria!  
  
Today is Friday, January 22, 2010.  
The time is 02:54:58 p.m.  
It is 22 days into the current year.
```

The time Method

The time method returns an atom value, representing the number of seconds elapsed since a fixed point in time. Following example explains it in detail –

```
#!/home/euphoria-4.0b2/bin/eui  
  
constant ITERATIONS = 100000000  
integer p  
atom t0, t1, loop_overhead  
  
t0 = time()  
for i = 1 to ITERATIONS do  
    -- time an empty loop  
end for  
  
loop_overhead = time() - t0  
  
printf(1, "Loop overhead:%d\n", loop_overhead)  
  
t0 = time()  
for i = 1 to ITERATIONS do  
    p = power(2, 20)  
end for  
  
t1 = (time() - (t0 + loop_overhead))/ITERATIONS  
  
printf(1, "Time (in seconds) for one call to power:%d\n", t1)
```

This would produce following result –

```
Loop overhead:1  
Time (in seconds) for one call to power:0
```

Date & Time Related Methods

Euphoria provides a list of many methods which helps you in manipulating date and time. These methods are listed in [Euphoria Library Routines](#)

Loading [MathJax]/jax/output/HTML-CSS/fonts/TeX/fontdata.js