

# FreeBasic: Datum und Uhrzeit

---

## Time

---

Die einfachste Möglichkeit die Zeit auszugeben ist `Time`.

```
print time
sleep
```

Ok, die Zeit wird ausgegeben, aber wie speichert man die Uhrzeit? Das geht so:

```
dim Zeit as string
Zeit = time
print Zeit
sleep
```

Die Uhrzeit wird bei `Time` im Format "hh:mm:ss" wiedergegeben. Wenn du aufgepasst hast siehst du, dass die Variable `Zeit` im Beispiel ein String ist. Wo liegt das Problem könntest du dich Fragen. Das Problem ist das man die Uhrzeit im allgemeinen als Zahlen haben möchte. Im Thema String Manipulation hat du die Funktion `Mid()` kennen gelernt. Mit dieser Funktion werden wir uns jetzt drei Hilfsfunktionen schreiben:

```
declare function GetHour() as integer
declare function GetMinute() as integer
declare function GetSecond() as integer

function GetHour() as integer
    dim Zeit as string
    Zeit = mid(time,1,2)
    return val(Zeit)
end function

function GetMinute() as integer
    dim Zeit as string
    Zeit = mid(time,4,5)
    return val(Zeit)
end function

function GetSecond() as integer
    dim Zeit as string
    Zeit = mid(time,7,8)
    return val(Zeit)
end function
```

So jetzt kannst du die einzelnen Bestandteile der Uhrzeit als Integer benutzen. Aber eins haben wir noch vergessen, die Sekundenbruchteile. Man braucht oft genug zum Vergleich wie lang bestimmte Vorgänge dauern Sekundenbruchteile. Hier noch eine Hilfsfunktion um Sekundenbruchteile zu benutzen. Die Erklärung reicht ins nächste Thema herein.

```
declare function GetSplitSecond() as double

function GetSplitSecond() as double
    return timer-int(timer)
end function
```

## Zeitmessung

---

Im letzten Unterkapitel hast du gesehen, dass man auch Sekundenbruchteile messen kann. Wenn du ins letzte Beispiel schaust, wirst du sehen, dass in der Funktion GetSplitSecond() nicht Time sondern Timer benutzt wird. Der Buchstabe macht viel aus. Time gibt die aktuelle Uhrzeit zurück, während Timer die Zeit die der Computer an ist auf 16 Nachkommastellen genau angibt. Mit dieser unscheinbaren Funktion lässt sich einiges anstellen. Im Kapitel Zufall hast du von ihr gebrauch gemacht, um die Zufallsreihen zu initialisieren. Hier wirst du sie benutzen um die Zeit zu messen. Ein kleines Beispiel:

```
dim Zeit as double
dim i as integer
Zeit = timer
for i = 0 to 50000
    print "Ich verschwende Zeit!"
next
Zeit = timer - Zeit
print Zeit
sleep
```

Der Text "Ich verschwende Zeit!" wird 50000 mal ausgegeben. Die Differenz zwischen Endzeit und Startzeit ist die dauer der For-Schleife. Bei mir waren es 4.56 Sekunden. Ihr werdet sicher ein anderes Ergebnis bekommen. Versucht einfach mal.

## Datum

---

Das Datum ermittelt man mit Date. Das Format ist "mm-dd-yyyy". Ansonsten ist alles gleich wie bei Time. Für das Datum brauchen wir auch Hilfsfunktionen, die so aussehen:

```
declare function GetMonth() as integer
declare function GetDay() as integer
declare function GetYear() as integer

function GetMonth() as integer
    dim Zeit as string
    Zeit = mid(date,1,2)
    return val(Zeit)
end function

function GetDay() as integer
    dim Zeit as string
    Zeit = mid(date,4,5)
    return val(Zeit)
end function

function GetYear() as integer
    dim Zeit as string
    Zeit = mid(date,7,10)
    return val(Zeit)
end function
```

Mehr gibt es nicht wirklich zu sagen ausser, dass man mit Setdate das Datum setzen kann. Die Funktion erwartet einen String wie ihn Date liefern würde, allerdings solltest du von dieser Funktion nur gebrauch machen wenn es wirklich sein muss.

---

Abgerufen von [https://de.wikibooks.org/w/index.php?title=FreeBasic:\\_Datum\\_und\\_Uhrzeit&oldid=442976](https://de.wikibooks.org/w/index.php?title=FreeBasic:_Datum_und_Uhrzeit&oldid=442976)

---

Diese Seite wurde zuletzt am 23. März 2009 um 13:37 Uhr bearbeitet.

Der Text ist unter der Lizenz "Creative Commons" „Namensnennung – Weitergabe unter gleichen Bedingungen“ verfügbar. Zusätzliche Bedingungen könnengelten. Einzelheiten sind in den [Nutzungsbedingungen](#) beschrieben.