

# marxer engineering & computing

[www.mec.li](http://www.mec.li)

## *Mathematica* für Software Entwickler

Hier werden einige Eigenschaften von *Mathematica* aufgelistet, die vor allem für Software Entwickler von Interesse sein könnten.

### Funktionale, Prozedurale ... Programmierung

*Mathematica* ist sehr flexibel und bietet umfangreiche Programmier Möglichkeiten. Oft gibt es mehrere Möglichkeiten zur Implementierung eines Algorithmus.

Zum Beispiel werden hier zwölf verschiedene Möglichkeiten, die Fakultät von  $n$  zu berechnen, dargestellt. Man kann eingebaute Funktionen, direkt die Definition der Fakultät, prozedurale Konstrukte mit Do oder For loops sowie Listenfunktionen verwenden.

```
result = {};
Do[ Clear[f];
  Switch[i,
    (* eingebaute Funktionen *)
    1, f := Factorial,
    2, f[n_] := n!,
    3, f[n_] := Gamma[n+1],
    (* Definition der Fakultät, rekursiv *)
    4, f[n_] := n f[n-1]; f[1] = 1,
    5, f[n_] := If[n == 1, 1, n f[n-1]],
    6, f = If[#1 == 1, 1, #1 #0[#1 - 1]] &,
    (* prozedural mit Do oder For *)
    7, f[n_] := Module[{t = 1, i}, Do[t = t*i, {i, n}]; t],
    8, f[n_] := Module[{t = 1, i}, For[i = 1, i <= n, i++, t *= i]; t],
    (* Listenfunktionen *)
    9, f[n_] := Product[i, {i, n}],
    10, f[n_] := Apply[Times, Range[n]],
    11, f[n_] := Fold[Times, 1, Range[n]],
    12, f[n_] := Fold[#2[#1] &, 1, Array[Function[t, #t] &, n]]
  ];
AppendTo[result, f[5]]; (* Teste für alle Definitionen *)
, {i, 1, 12}
];
result

{120, 120, 120, 120, 120, 120, 120, 120, 120, 120, 120, 120}
```

## Aufruf einer Java Funktion

Hier ist ein Beispiel für eine Berechnung, die in Java durchgeführt wird. Java gibt eine ganze Reihe von starken Datums- und Kalender orientierten Klassen. Falls wir einen schön formatierten String mit Zeit und Datum anzeigen wollen, können wir folgendermassen vorgehen.

1. Schritt: wir kreieren ein Java Date Object, das das momentane Datum und die Zeit repräsentiert.

```
Needs["JLink`"]; InstallJava[];
date = JavaNew["java.util.Date"]

«JavaObject[java.util.Date] »
```

2. Schritt: wir laden die DateFormat Klasse und kreieren einen Formatierer.

```
LoadJavaClass["java.text.DateFormat"];
dateFormatter = DateFormat`getInstance[]

«JavaObject[java.text.SimpleDateFormat] »
```

3. Schritt: wir rufen die format() Methode auf und übergeben ihr das Date object.

```
dateFormatter@format[date]

14.02.05 11:38
```

Diese Schritte wurden alle - interaktiv - innerhalb von *Mathematica* aufgerufen.

**Bemerkung:** Ähnlich einfach können eigenentwickelte Java Funktionen (Klassen), DLLs, C++, NET Funktionen aufgerufen werden.