

Kurzeinführung in o++o

Klaus Benecke

Da die gestrige und heutige Schule nur mit Zahlen rechnet, können die meisten Menschen auch nur mit Zahlen rechnen. Das Morgen verlangt aber auch ein Rechnen mit Worten, Texten, strukturierten Tabellen und und strukturierten Dokumenten. Von diesem Rechnen muss das Programmieren direkt ableitbar sein, da der Computer immer mehr an Bedeutung gewinnen wird.

Dadurch können folgende Verbesserungen erzielt werden:

1. Einfacheres, allgemeineres und standardisiertes Rechnen
2. Verbesserung der Problemlösungsfähigkeiten der Schüler und Mitarbeiter durch mehr Anwendungsorientierung
3. Jeder kann Rechnungen und Anfragen an den Computer verstehen bzw. selbst stellen.

Notwendige neue Stoffeinheiten:

1. Jeder sollte auch selbst selektieren und Daten verbinden können.
2. Einführung weiterer Massendatenoperationen und Verallgemeinerung der bekannten Operationen (+ * sin ...) auf Tabmente (= TABelle + dokuMENT).
3. Visualisierungen von Tabellen mit Konzepten, die in der Schule vermittelt werden.

o++o Beispiele:

1+2*3 (ergibt 9: einfach von links nach rechts rechnen)

1 3 5 2 (Liste von Zahlen, die unmittelbar durch ein Diagramm visualisiert werden kann)

1 3 5 2 ++ (ergibt die Summe 11)

18 *1 37 ++ (methodisch einfachere Multiplikation; ergibt 666)

1 3 5 2 ++1 (ergibt die Anzahl 4)

1 3 5 2 ++: (ergibt den Durchschnitt 2.75)

200 +% 19 (ergibt 238; zu 200 19% hinzufügen)

(“Schultaschenrechner von Texas Instruments rechnet falsch”: “200 + 19 %”)

1 3 7 * 2 (ergibt die Liste 2 6 14)

1 .. 5 (ergibt die Liste 1 2 3 4 5)

1 .. 100 ++ (ergibt 5050; Schulproblem von Gauß)

1 ... 2!0.1 (ergibt Liste 1. 1.1 1.2 1.3 1.4 1.5 1.6 1.7 1.8 1.9 2.0)

1 ... 2!0.000'001 sin max (Näherung für das Maximum 1 der Funktion Sinus)

0 ... 2*pi!0.000'001 sin abs *0.000'001 ++ (ergibt eine gute Näherung für 4; der Fläche unter den Sinusbögen)

Die letzten beiden Programme sind Anwendungen der 11. und 12. Klassen, die dann bereits in den 9. und 10. Klassen vermittelt werden könnten.

365.25 24 60 60 ** '3 (ergibt die Anzahl der Sekunden eines Jahres 31'557'600; durch die Formatierungsoperation '3 werden größere Zahlen besser lesbar; diese Schweizer Idee löst viele Konvertierungsprobleme internationaler Firmen; da der Punkt in einigen Ländern als Dezimalzahltrenner dient und in anderen Ländern das Komma.)

einwohner sei eine kleine Tabelle mit 4 Spalten (NAME, LAND, ORT, GEHALT), die Personen aus Deutschland enthalte.

einwohner

sel ORT=Magdeburg

++:

Berechnet das Durchschnittsgehalt aller Magdeburger einfach von oben nach unten.

einwohner

gib DUR,(LAND,DUR,(ORT,DUR m)m)

DUR:=GEHALT! ++:

berechnet eine strukturierte Tabelle, die die Durchschnittsgehälter für jeden Ort, jedes Bundesland und insgesamt berechnet. Die Daten werden nach Land und innerhalb jedes Landes nach Wohnort sortiert.

meinkonto2022.csv

sel Bernhard

sel BETRAG>0

++

Summiert alle positiven Überweisungen meines Sparkassenkontos von Bernhard des Jahres 2022.

Eine BMI-Berechnung

Sei **bmidaten.tab**: NAME,LÄNGE,(ALTER,GEWICHT m)m eine strukturierte Tabelle, die für bestimmte Personen den Namen, die Länge mit (Gewicht,Alter)-Paaren enthalte:

NAME,	LÄNGE,	(ALTER,	GEWICHT m)m
Klaus	1.68	18	61
		30	65
		60	80
Kathi	1.71	18	55
		40	70
Viktoria	1.61	13	51
Bert	1.72	18	66
		30	70
Peter	1.69	18	70
		60	100

bmidaten.tab

```
sel NAME! ALTER > 20 # selektiert lediglich Personen, für die  
# ein Alterseintrag größer 20 existiert
```

```
gib BMI,(ALTER,BMI,(NAME,BMI m) m)
```

```
BMI:=GEWICHT:LÄNGE:LÄNGE ! ++:
```

```
rnd 1
```

berechnet den Durchschnitts-Body-Mass-Index aller Erwachsenen pro Alter und für jede Person des Alters sowie den Gesamtdurchschnitt der BMIs. Die Daten werden auf eine Stelle nach dem Komma gerundet.

Die strukturierte Ergebnistabelle:

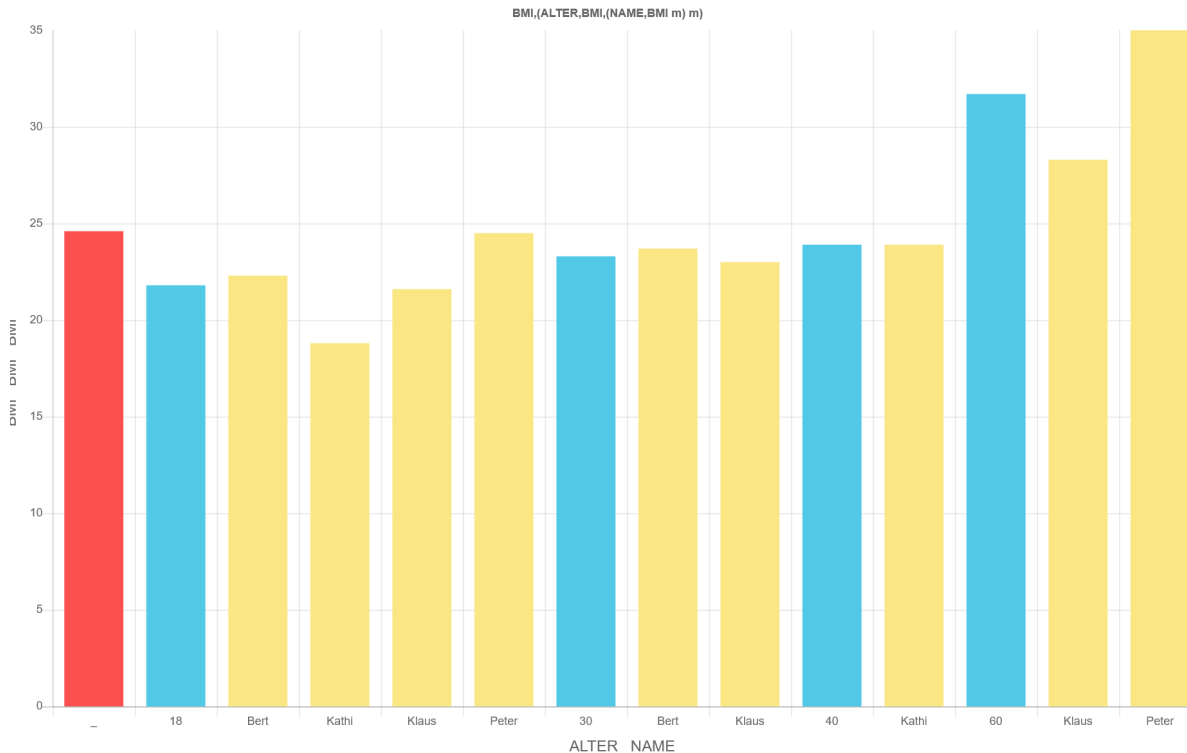
```
BMI, (ALTER,BMI, (NAME, BMI m) m
```

```
24.6 18 21.8 Bert 22.3  
Kathi 18.8  
Klaus 21.6  
Peter 24.5  
30 23.3 Bert 23.7  
Klaus 23.0  
40 23.9 Kathi 23.9  
60 31.7 Klaus 28.3  
Peter 35.0
```

Da man beispielsweise in EXCEL strukturierte Tabellen nicht sortieren kann, erfordert dieses kurze Programm mehr als 6 EXCEL Arbeitsblätter. Mit dem Standard für relationale Datenbanken SQL ist das Problem nicht formulierbar. Anstelle der Ergebnistabelle kann auch ein **strukturiertes Diagramm** einfach durch Klicks ausgegeben werden. Hierfür wurde der Typ des Alters durch

```
ALTER::= ALTER text
```

in Text umgewandelt.



Eine Bilanz

<TAB!

VERMÖGEN,	(VERMÖGENSART,	AKTIVA l) l,	(KAPITAL,	(KAPITALART,	PASSIVA l) l)
I. Anlage	1 Grundst. und Bauten	10092000	I. Eigen	summe	16800000
	2 TA und Maschinen	3243600	II.Fremd	1 Hypothekenschulden	4927200
	3 Fuhrpark	512400		2 Darlehen	1444800
	4 BGA	552000		3 Verbindlichkeiten a.LL	828000
II.Umlauf	1 Rohstoffe	2886000			
	2 Hilfsstoffe	988800			
	3 Betriebsstoffe	184800			
	4 Unfertige Erzeugnisse	753600			
	5 Fertige Erzeugnisse	2462400			
	6 Forderungen a.LL.	1192800			
	7 Bankguthaben	1099200			
	8 Kassenbestand	32400			

!TAB>

total ++

'3

Ergebnis:

VERMÖGEN,	(VERMÖGENSART	, AKTIVA l) l,	(KAPITAL	,(KAPITALART	,PASSIVA l) l)
I. Anlage	1 Grundst. und Bauten	10'092'000	I. Eigen	summe	16'800'000
	2 TA und Maschinen	3'243'600		sum	16'800'000
	3 Fuhrpark	512'400	II.Fremd	1 Hypothekenschulden	4'927'200
	4 BGA	552'000		2 Darlehen	1'444'800
	sum	14'400'000		3 Verbindlichkeiten a.LL	828'000
II.Umlauf	1 Rohstoffe	2'886'000		sum	7'200'000
	2 Hilfsstoffe	988'800	sum	sum	24'000'000
	3 Betriebsstoffe	184'800			
	4 Unfertige Erzeugnisse	753'600			
	5 Fertige Erzeugnisse	2'462'400			
	6 Forderungen a.LL.	1'192'800			
	7 Bankguthaben	1'099'200			
	8 Kassenbestand	32'400			
	sum	9'600'000			
sum	sum	24'000'000			