

Superbase 64 (Teil 5)

Wir stellen Ihnen ein Programm vor, um die Tankbelege eines Autos auszuwerten und gleichzeitig Superbase-Datenbanksprache zu lernen.

Das Hauptprogramm »Verbrauch« (Listing 2) übernimmt folgende Aufgaben:

- Erfragen der Daten
- Kontrolle dieser Daten
- Automatische Bestimmung der Buchungsnummer
- Abspeichern des Datensatzes
- Errechnen der Liter getankten Benzins seit Bestehen der Datei
- Errechnen der dabei aufgetretenen Kosten
- Errechnen der gefahrenen Kilometer seit Bestehen der Datei
- Errechnen der Kosten pro 100 Kilometer seit Bestehen der Datei
- Errechnen der Kosten pro 100 Kilometer über die letzte Tankfüllung
- Darstellung der Kostenentwicklung pro 100 Kilometer über die letzten vier Tankfüllungen
- Errechnen des Benzinverbrauchs pro 100 Kilometer seit Bestehen der Datei
- Errechnen des Benzinverbrauchs pro 100 Kilometer über die letzte Tankfüllung
- Darstellung der Entwicklung des Benzinverbrauchs pro 100 Kilometer über die letzten vier Tankfüllungen

Viel Spaß mit eigenen Anwendungen in Superbase.

Für die beiden hier vorgestellten aber auch für andere Programme in Superbase müssen Sie zunächst das Datenbankprogramm auf die übliche Weise laden. Anschließend legen Sie sich über das Startprogramm eine neue Datenbank mit einer neuen Datei an. Die Namen sind dabei ausnahmsweise ohne Bedeutung, denn im abgedruckten Listing werden die Befehle »DATABASE« und »FILE« nicht verwendet. Sie befinden sich jetzt automatisch im »FORMAT«-Modus. Informationen über das Datensatzlayout entnehmen Sie bitte Bild 1. Die Datensatzstruktur ist in Bild 2 wiedergegeben. Bei einer eventuellen Änderung des Layouts denken Sie bitte daran, daß die Feldnamen — hier nur Zahlen — und Feldlängen nicht verändert werden dürfen, ansonsten ist das Programm dieser Ausgabe nicht lauffähig. Jetzt können Sie, nachdem der »PROG«-Modus über das zweite Menü angewählt wurde, die Listings eingeben. Innerhalb von Superbase müssen Sie leider ohne den bewährten Checksummer arbeiten.

Das Programm »PKW« (Listing 1) erzeugt lediglich eine Variablendatei. Anschließend wird automatisch das Hauptprogramm nachgeladen und gestartet. Speichern Sie beide Programme wie üblich vor dem Starten ab. Die Variablendatei wird durch den Befehl »DUMP«-Name erzeugt. Alle Variablen, mit Ausnahme der Feldvariablen, werden auf die Diskette übertragen. Die mit dem vorliegenden Programm erzeugte Datei mit dem Namen »hpkw« wird für das Hauptprogramm benötigt, welches später wesentlich mehr Variablen abspeichert. Wenden Sie also das Programm »PKW« kein zweites Mal an (zumindest nicht auf der Diskette mit dem Datenbestand von »Verbrauch«). Übrigens wurde dem Dateinamen ein »h« vorangestellt, damit die Datei mit dem »HELP«-Modus gelesen werden kann.

Nun zu einigen Einzelheiten des Programmes »Verbrauch« (Leistungsübersicht siehe Tabelle 1). Das Schlüs-

selfeld enthält die Buchungsnummer als String, wodurch sich Zeile 2040 erklärt. Die alte Buchungsnummer wurde mit anderen Variablen in Zeile 2010 durch den Befehl SET »Name« gelesen. Der Datensatz wird unter Umgehung des ENTER-Modus erstellt. In den Zeilen 4030 bis 4070 findet die Felderbelegung statt. Mit STORE wird der Datensatz gespeichert, mit CLEAR dagegen die Felder im Speicher — nicht auf der Diskette — gelöscht.

Wenn Sie übrigens bei der Gestaltung einer Bildschirmseite in einem eigenen Superbase-Programm die Meldung »drücken Sie RETURN um fortzufahren Seitenende« erhalten, kann es daran liegen, daß Ihre DISPLAY-Programmzeile ohne »;« abschließt und Sie sie mit einer zweiten DISPLAY-Befehlszeile anspringen wollen. Anders verhält es sich dagegen mit Anspringen der Bildschirmzeile durch den ASK-Befehl.

In Zeile 8150 wird das Programm beendet. Wenn QUIT anstatt MENU in dieser Zeile steht, wird Superbase ordnungsgemäß abgeschlossen und der C 64 meldet sich in seinem Einschaltzustand.

(Gerd Wiechering/bj)

```

999 rem *****
1000 rem Titeldbild
1001 rem
*****
1010 display @14,2 "Programm zur"
1020 display @5,4 "Erzeugung einer Variablendatei"
1030 display @16,6 "fuer die"
1040 display @10,10 "Verbrauchsberechnung"
1050 display @15,11 "fuer einen"
1060 display @18,12 "PKW"
1070 display @12,16 "ein Programm von"
1080 display @14,18 "G.Wiechering"
1090 display @23,21 chr$(18) "Taste druecken"
1100 wait
1999 rem *****
2000 rem Programmbeschreibung
2001 rem *****
2010 display chr$(147)
2020 display "Das Programm 'Verbrauchsberechnung';":display @1
2030 display "welches automatisch nachgeladen und ge-":display @1
2040 display "startet wird, benoetigt den Kilometer-":display @1
2050 display "stand, bei dem die Berechnungen begin-":display @1
2060 display "nen sollen. Dies muss ein Zeitpunkt":display @1
2070 display "sein, bei dem der Wagen vollgetankt ist."
2080 display "Erst ab der folgenden Tankfuellung koen-":display @1
2090 display "nen die Berechnungen beginnen.":display @1
2100 display @23,21 chr$(18) "Taste druecken"
2110 wait
2120 display chr$(147)

2130 display "Das Programm 'pkw' darf nur ein einziges"
2140 display "Mal angewendet werden. Ich empfehle Ih-"
2150 display "nen, das Programm so schnell wie moeg-":
display @1
2160 display "lich zu loeschen."
2170 display @23,21 chr$(18) "Taste druecken"
2180 wait
2999 rem *****
3000 rem Kilometerstanderfassung
3001 rem *****
3010 display chr$(147)
3020 display @1,3 "Bei welchem Kilometerstand haben Sie"
3030 display @1,5 "den Wagen vollgetankt?"
3040 ask @1,7ka
3050 display @1,10 "(Bitte lassen Sie die Liter-Zahl und"
3060 display @1,12 "die Kosten fuer die folgenden Berech-"
3070 display @1,14 "nungen unberuecksichtigt!)"
3080 display @1,16&6,0ka
3090 display @1,18 "Ist dies der richtige Ausgangskilometer-"
3100 display "stand (j/n)?";
3110 display @14,20 "j"
3120 ask @14,20a$
3130 if a$ = "n" then 3010
3140 if a$ < > "j" then 3120
3150 a$ = "wg"
3160 rem Speichern der Variablen
3170 :dump "hpkw"
3180 rem Programmteil nachladen
3190 :load "verbrauch"

```

Listing 1. »PKW« erzeugt die Variablendatei für »Verbrauch«. Beide Programme müssen im Superbase-PROG-Modus eingegeben werden und sind auch nur dort lauffähig.

SUPERBASE-Steckbrief: PROG-Modus (Fortsetzung)

WAIT	Warten auf Tastendruck; I;	DATABASE "Name",x,y	Datenbankauswahl; I; x = Ge- rätenummer, y = Laufwerk- nummer, Defaultwerte: 8,0
SET "Name"	auch »WAITa« oder »WAITa\$« Abspeichern einer Variablen- datei; I	DATABASE "\$"	Directory-Anzeige (wenn man sich nicht an den Datenbank- namen erinnert); I; Daten- banken erscheinen in Groß- buchstaben
DUMP "Name"	Laden einer Variablendatei; I	FILE "Name"	Dateiauswahl; I
CLEAR	Löschen der Felder im Spei- cher; I	CONVERTa,a\$	Umwandlung einer Zahlenva- riable in eine Stringvariable; I
STORE	Abspeichern des Datensatzes; I	CONVERT[Datum- feld],d\$	Umwandlung eines Datums in eine String- oder auch eine Zahlenvariable; I
ASKa\$	Abfrage einer Stringvariablen bis maximal 40 Zeichen; I; auch »ASKa«, »ASK[Feldna- me]«, »ASKText«;a\$; auch als Positionierungs- und Verkür- zungsbefehl anwendbar	Anmerkung:	Ausstieg aus Superbase; I
DATED\$d,d	Zuordnung der Monatszahl der Variablen »d«; I; bei kor- rekter Datumseingabe ist »d« ungleich Null	Das »I« im Steckbrief ist stellvertretend für »Primärbe- fehl« verwendet worden. Siehe hierzu auch die Steckbriefe aus Ausgabe 7/86.	

Bild 1. Datensatzlayout

```

Mode: Format
      Tankbelege
Buchungsnummer 01
Tankdatum      02
Kilometerstand 03 + # # # # #
Benzin         04 + # # . # #
Rechnung       05 + # # # . # #

```

Bild 2. Datensatzstruktur

```

Dateidefinition: auto/verbrauch
#  Name      Typ      Format/Berechnung
1  01        Schlüsselfeld  Länge      4
2  02        Datumsfeld   Länge      7
3  03        Zahlenfeld   + # # # # #
4  04        Zahlenfeld   + # # . # #
5  05        Zahlenfeld   + # # # . # #

```

```

999 rem *****
1000 rem Titelbild
1001 rem *****
1010 display chr$(147)
1020 display @10,4"Verbrauchsberechnung"
1030 display @15,5"fuer einen"
1040 display @18,7"PKW"
1050 display @14,10"ein Programm"
1060 display @18,11"von"
1070 display @14,14"GWiechering"
1080 display @23,21 chr$(18) "Taste druecken"
1090 wait
1999 rem *****
2000 rem Aufrufen der Variablendatei
2001 rem *****
2010 set "hpkw"
2020 kl=k0
2030 rem Erhoehung der Buchungsnummer
2040 :bu$=str$(val(bu$)+1)
2999 rem *****
3000 rem Erfragen des neuen Datensatzes
3001 rem *****
3010 display chr$(147)
3020 display "Buchungsnummer"bu$
3030 display @1,4"Wann wurde der Wagen betankt?"
3040 display @1,7"(ttmmjj)"
3050 ask @2,6d$
3060 rem ueberpruefen des Datums
3070 :dated$d
3080 :if d=0 then 3050
3090 display @1,9"Wie lautet der Kilometerstand?"
3100 ask @1,11k0
3110 display @1,13"Wieviel Liter wurden getankt?"
3120 ask @1,15li
3130 display @1,17"Wie hoch war die Rechnung?"
3140 ask @1,19dm
3999 rem *****
4000 rem Felder mit Variablen belegen
4001 rem *****
4010 rem bestehende Feldinhalte loeschen
4020 :clear
4030 [01]=bu$
4040 [02]=d$
4050 [03]=k0
4060 [04]=li
4070 [05]=dm
4999 rem *****
5000 rem Ueberpruefen des Datensatzes

5001 rem *****
5010 display chr$(147)
5020 display @4,7"Buchungsnummer";
5030 display @28[01]
5040 display @4,9"Tankdatum";
5050 display @23[02]
5060 display @4,11"Kilometerstand";
5070 display @23&6,[03]
5080 display @4,13"Benzin";
5090 display @17[04];
5100 display @31"Liter"
5110 display @4,15"Rechnung";
5120 display @17[05];
5130 display @31"DM"
5140 display @4,18"Sind diese Angaben richtig? (j/n)";
5150 display @38"j"
5160 ask @38,18a$
5170 ifa$="n" then 2010
5180 if a$<>"j" then 5160
5190 rem Abspeichern des Datensatzes
5200 :store
5999 rem *****
6000 rem Berechnungen
6001 rem *****
6010 rem Gesamtliterzahl
6020 :lg=lg+li
6030 rem Gesamtkosten
6040 :kg=kg+dm
6050 rem gefahrene Kilometer insgesamt
6060 :f0=k0-ka
6070 rem Durchschnittsverbrauch
6080 :d0=lg*100/f0
6090 rem Durchschnittskosten
6100 :d5=kg*100/f0
6110 rem zuletzt gefahrene Kilometer
6120 :f1=k0-kl
6130 rem letzter durchschnittl. V.
6140 :d1=d2,d2=d3,d3=d4
6150 :d4=li*100/f1
6160 rem letzte durchschnittl. K.
6170 :d6=d7,d7=d8,d8=d9
6180 :d9=dm*100/f1
6999 rem *****
7000 rem Verbrauchsanzeige
7001 rem *****
7010 display chr$(147)
7020 display @1,2"Verbrauch, gesamt:"lg
7030 display @4,3"Kosten, gesamt:"kg

7040 display @1,5"Durchschnittsverbrauch, gesamt:"
7050 display @1,6&2,2d0
7060 display @1,7"Entwicklung des durchschnittlichen"
7070 display @1,8"Verbrauchs ueber die letzten vier"
7080 display @1,9"Tankfuellungen:"
7090 display @2,2d1&2,2d2
7100 display @2,2d3&2,2d4
7110 display @1,13"Durchschnittskosten, gesamt:"
7120 display @1,14&2,2d5
7130 display @1,15"Entwicklung der durchschnittlichen"
7140 display @1,16"Kosten ueber die letzten vier"
7150 display @1,17"Tankfuellungen:"
7160 display @2,2d6&2,2d7
7170 display @2,2d8&2,2d9
7180 display @23,21 chr$(18) "Taste druecken"
7190 wait:a$="wg"
7999 rem *****
8000 rem Speichern der Variablen
8001 rem *****
8010 dump "hpkw"
8099 rem *****
8100 rem Programmabschluss
8101 rem *****
8110 display chr$(147)
8120 display @1,5"Wuenschen Sie eine weitere Eingabe?";
8130 display @37,5"n"
8140 ask @37,5"a$"
8150 if a$="n" then menu
8160 if a$<>"j" then 8140
8170 goto 2010
8999 rem *****
9000 rem Variablenliste
9001 rem *****
9010 rem bu$ = Buchungsnummer
9020 rem d$ = Datum
9030 rem d = Monatszahl
9040 rem ka = alter Kilometerstand
9050 rem k0 = neuer Kilometerstand
9060 rem kl = letzter Kilometerstand
9070 rem li = Liter
9080 rem lg = Liter, gesamt
9090 rem dm = Rechnungsbetrag
9100 rem kg = Kosten gesamt
9110 rem d0 = durchschn Verbrauch gesa
9120 rem d1-d4 = durchschn Verbrauch
9130 rem d5 = durchschn Kosten gesamt
9140 rem d6-d9 = durchschn Kosten
9150 rem a$ = Befehlsstring**

```

Listing 2. »Verbrauch« erfaßt und liefert Ihnen alle wichtigen Daten über den Verbrauch, die Kosten und Kilometerleistung Ihres Autos.