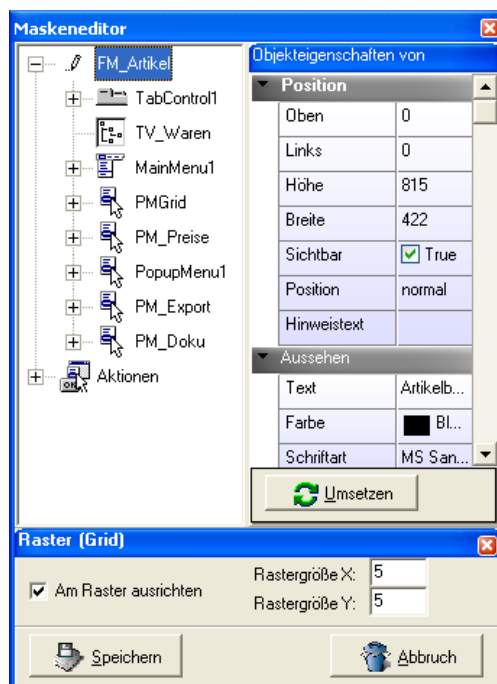


Maskeneditor

für die AFS-Warenwirtschaft



NUTZUNGSLIZENZVERTRAG: (Lizenzbedingungen & Lizenzvertrag)

1. VERTRAGSGEGENSTAND

1.1 Die AFS-Software GmbH & Co.KG räumt dem Käufer ein nicht ausschließliches Recht zur Nutzung des Programms einschließlich der zugehörigen Dokumentation auf einem einzelnen PC bzw. in einem einzelnen Netzwerk ein. Im übrigen verbleiben alle Rechte an dem Programm einschließlich der Dokumentationen bei AFS-Software GmbH & Co.KG und deren Lizenzgebern.

1.2 Der Anwender ist verantwortlich für die Auswahl der Software zur Erreichung der von ihm angestrebten Ergebnisse und für die Installation, die Nutzung und die Ergebnisse, welche er von dieser Software erhält; insbesondere stellen die Leistungsdaten und sonstigen Softwarebeschreibungen keine Zusicherung bestimmter Eigenschaften dar.

1.3 Installation, Schulung im Umgang mit der Software und Beratung sind nicht Gegenstand dieses NUTZUNGSLIZENZVERTRAGES. AFS-Software GmbH & Co.KG ist auf Wunsch des Anwenders bereit, hierüber eine gesonderte Vereinbarung zu treffen.

1.4 Zur Sicherung der vertragsgemäßen Nutzung kann der Kunde Kopien des Programms herstellen, die er ausschließlich zum eigenen Gebrauch verwendet. Bei Weitergabe oder Veröffentlichung der mit dem Programm errechneten Daten muß die AFS-Software GmbH & Co.KG als Quelle angegeben werden.

1.5 Die dem Anwender in diesem Vertrag eingeräumten Nutzungsrechte darf der Anwender erst nach vollständiger Bezahlung der Lizenzgebühr ausüben.

2. GEWÄHRLEISTUNG

2.1 AFS-Software GmbH & Co.KG gewährleistet, daß die Disketten bzw. die CD frei von Material- und Herstellungsfehlern sind und daß das Programm bei Nutzung auf dem angegebenen System im wesentlichen mit den im elektronischen Handbuch oder auf der Verpackung beschriebenen Programmspezifikationen übereinstimmt.

Inhaltsverzeichnis

NUTZUNGSLIZENZVERTRAG: (Lizenzbedingungen & Lizenzvertrag) ..	2
Hinweise und Hotline-Service	10
Vorwort	11
Der Maskeneditor	12
Installation überprüfen	12
Die Fenster des Maskeneditors	13
<i>Die Kontrollleiste</i>	<i>13</i>
<i>Das Fenster „Maskeneditor“</i>	<i>14</i>
Festlegen von Eigenschaften	14
Näheres zu den Objekten	16
<i>Buttons:</i>	<i>17</i>
<i>Panels</i>	<i>17</i>
Die Eigenschaften eines Panel-Objekts:	18
<i>Labelfelder</i>	<i>20</i>
Datenbanktabellen und Navigationsleisten	21
<i>Datenbanktextfelder</i>	<i>24</i>
<i>Datenbanktextfelder</i>	<i>24</i>
Einschub: Original Kaufmann wiederherstellen	25
<i>BLOB-Felder</i>	<i>26</i>
<i>Datenbankfeld mit Auswahlliste (Listbox):</i>	<i>27</i>
<i>Datenbankfeld mit Auswahlklappliste</i>	<i>29</i>
<i>Schalter-Datenbankfeld</i>	<i>29</i>
<i>Datenbankverknüpfung als Klappbox</i>	<i>30</i>
<i>Datenbanklangtext formatiert</i>	<i>31</i>
Benutzerspezifische Masken	31
Aktionen	33
Aktionen für die Artikelverwaltung	33
Aktionen für das Kontaktmanagement	38
Aktionen für die Kundenbearbeitung	43
Aktionen für die Interessentenbearbeitung	46
Aktionen für die Lieferantenbearbeitung	49
Aktionen für das Fenster „Warengruppe“	52
Aktionen für „Artikelliste nach Warengruppen“	53
Aktionen für das Fenster „Lagerbuchung“	55
Aktionen für die Inventur	56
Aktionen für die Preisliste	58

AFS-Maskeneditor	
Aktionen für die Rabattmatrix	59
Aktionen für den Abbuchungsauftrag	63
Aktionen für Angebote	64
Aktionen der Arbeitszeitverwaltung	74
Aktionen der Artikeltarife	74
Aktionen der Auftragsbestätigung	75
Aktionen der Barkasse	80
Aktionen des Bestellwesens	84
Aktionen des DTA-Bankings	85
Aktionen für externe Preislisten	86
Aktionen der Geräteverwaltung	86
Aktionen für Gutschriften	88
Aktionen für Kassenabschlussberichte	94
Aktionen für das Kassenbuch	94
Aktionen der Lagerbuchung	95
Aktionen für Lastschriften	97
Aktionen für Lieferscheine	98
Aktionen für Mitarbeiter	104
Aktionen für MwSt-Einstellungen	104
Aktionen der neuen Aufträge	104
Aktionen für neue Angebote	112
Aktionen für Rechnungen	122
Aktionen des Terminplaners	131
Aktionen des Textwriters	132
Aktionen der Touchkasse	133
Aktionen der Umsatzsteuerauswertung	137
Aktionen für das Verdichten von Vorgängen	137
Aktionen der Vorgangsliste	137
Aktionen der Warengruppen	140
Aktionen für die Warengruppenauswertung	141
Aktionen des Wartungs- und Abowesens	142
aktive Elemente	143
Objekteigenschaften	144
<i>Position</i>	145
<i>Aussehen</i>	145
<i>Datenbanken</i>	145
Datenquelle	145

Datenfeld	145
Eingabe	145
Eingeschaltet	145
Aktionen	145
Anweisungen	145
SQL-Anweisung	145
Basic-Programm	146
Dateiauswahl	146
AFS- Basicscript-Sprache	146
Tabellenobjekte	147
Grundeigenschaft von sichtbaren Objekten	149
AFS-Basic Syntax:	149
<i>Die Script Strukturen</i>	<i>149</i>
<i>Die Bezeichner</i>	<i>150</i>
<i>Zeichen String</i>	<i>151</i>
<i>Kommentare</i>	<i>151</i>
<i>Variablen</i>	<i>151</i>
<i>Index</i>	<i>152</i>
<i>Arrays</i>	<i>152</i>
<i>If - Anweisungen</i>	<i>152</i>
<i>While Anweisung</i>	<i>153</i>
<i>Loop Anweisung</i>	<i>153</i>
<i>For Anweisung</i>	<i>154</i>
<i>Select case Anweisung</i>	<i>154</i>
Funktionen und Sub Deklaration	155
<i>Einbinden von DLL und deren Funktionen:</i>	<i>155</i>
Funktionsübersicht:	156
<i>Abs</i>	<i>156</i>
<i>AnsiCompareStr</i>	<i>156</i>
<i>AnsiCompareText</i>	<i>157</i>
<i>AnsiLowerCase</i>	<i>157</i>
<i>AnsiUpperCase</i>	<i>157</i>
<i>ArcTan</i>	<i>157</i>
<i>Assigned</i>	<i>157</i>
<i>AssignFile</i>	<i>158</i>
<i>Beep</i>	<i>158</i>
<i>Chdir</i>	<i>159</i>
<i>Chr</i>	<i>159</i>
<i>CloseFile</i>	<i>159</i>
<i>CompareStr</i>	<i>159</i>
<i>CompareText</i>	<i>160</i>

AFS-Maskeneditor

Copy	160
CreateOleObject	161
Date	161
DateTimeToStr	161
DateToStr	162
DayOfWeek	162
Dec	162
DecodeDate	162
DecodeTime	163
DeleteFile	163
EncodeDate	163
EncodeTime	164
Exp	165
FileExists	165
FilePos	165
FileSize	166
FileShow	166
FloatToStr	166
Format	167
FormatDateTime	167
FormatFloat	167
Frac	168
GetCurrentDir	168
GetTempDir	168
GetAppDir	168
GetActiveOleObject	168
High	169
Inc	169
IncMonth	169
InputQuery	170
Insert	170
Int	170
IntToHex	171
IntToStr	171
IsLeapYear	171
IsValidIdent	171
Length	172
Ln	172
Low	172
LowerCase	172
MakeDir	172
Now	173

<i>Odd</i>	173
<i>Ord</i>	173
<i>Pos</i>	173
<i>Random</i>	174
<i>ReadLn</i>	174
<i>Reset</i>	174
<i>Rewrite</i>	175
<i>Round</i>	176
<i>RmDir</i>	176
<i>ShowMessage</i>	177
<i>Sin</i>	177
<i>Sqr</i>	177
<i>Sqrt</i>	177
<i>StrToDate</i>	178
<i>StrToDateTime</i>	178
<i>StrToFloat</i>	178
<i>StrToInt</i>	179
<i>StrToIntDef</i>	179
<i>StrToTime</i>	179
<i>Time</i>	180
<i>TimeToStr</i>	180
<i>Trim</i>	180
<i>TrimLeft</i>	181
<i>TrimRight</i>	181
<i>Trunc</i>	181
<i>UpperCase</i>	181
<i>VarArrayCreate</i>	182
<i>VarArrayHighBound</i>	182
<i>VarArrayLowBound</i>	182
<i>VarIsNull</i>	182
<i>VarToStr</i>	183
<i>Write</i>	183
<i>WriteLn</i>	183
Zugriff auf interne Variablen (Formularsprache)	184
GetVariable(string): string	184
GetFormular(string): string	184
Zugriff auf das Dateisystem (eigener Export & Import)	184
.....	184
Dateialoge	186
Arbeiten mit OLE-Objekten	186
Excel	187
.....	187
Outlook	188

AFS-Maskeneditor	
Tobit	189
Word	189
XML-Zugriff	189
.....	189
Externe Datenbanken ADO-Zugriff	190
NEUE AFS-Basic Befehle und Eigenschaften	191
SQL-Anweisungen sind nun via Script übergebbar (Änderbar):	192
E-Mail senden aus dem Programm	192
.....	192
<i>mit entsprechenden Einstellungen:</i>	
.....	192
Drucken aus dem Programm	192
.....	192
<i>mit entsprechenden Einstellungen:</i>	
.....	192
Beliebige SQL-Anweisung ausführen	193
.....	193
<i>mit APPFUNC("DOSQL: HIER DIE SQL ANWEISUNG ") können Sie UPDATE,DELETE, INSERT und StoredProcedures auf dem SQL- Server ausführen lassen.</i>	193
.....	193
<i>Beispiel:</i>	193
.....	193
Beliebige SQL-Abfrage ausführen	193
.....	193
<i>mit APPFUNC("SQL: HIER DIE SQL ANWEISUNG ") können Sie einen SELECT-befehl, eine View oder eine StoredProcedures auf dem SQL-Server ausführen lassen.</i>	193
.....	193
<i>Beispiel:</i>	193
.....	193
Eine Aktion des Hauptfensters ausführen	194
.....	194
<i>mit APPFUNC("ACTION:ACTIONNAME") können Sie eine Action (Funktion) des Hauptfensters ausführen lassen:</i>	194
<i>APPFUNC("ACTION:Ac_Ku_be")</i>	194
Öffnen der Vorgangsliste	194
.....	194
<i>mit APPFUNC("OPEN_VORGANGSLISTE:VORGANGSARTID") können Sie eine die Vorgangsliste öffnen:</i>	194
<i>APPFUNC("OPEN_VORGANGSLISTE:1")</i>	194
Öffnen eines Vorgangs	194
.....	194
<i>mit APPFUNC("OPEN_VORGANG:VORGANGSID") können Sie einen Vorgang öffnen:</i>	194

<i>APPFUNC("OPEN_VORGANG:" + TBL_Auftrag.FIELDBYNAME("AUFTRAG").AsString)</i> 194	
Kopieren eines Vorgangs	194
.....	194
<i>mit</i>	
<i>APPFUNC("VORGANGKOPIE: VORGANGSID~NEUE_VORGANGSARTID")</i> <i>können Sie einen Vorgang kopieren:</i>	194
Forführen eines Vorgangs	195
.....	195
Ausgabe eines Vorgangs (Drucken, Mailen, Archivieren als PDF)	195
Eigene Ereignisse (Eventhandling)	196
.....	196
Dynamische Forms:	197
Timer Objekt (Eventhandling)	197
.....	198
Action Objekt (Funktionszugriff)	198
Index	199

AFS-Maskeneditor

Hinweise und Hotline-Service

Bezugs-bzw. Info-Adresse:

AFS-Software GmbH & Co. KG
Klaustor 3

D- 36251 Bad Hersfeld

E-Mail: post@afs-software.de
Internet: <http://www.afs-software.de>

Keine Zeit, um das Handbuch zu lesen?

Wir verstehen, dass Sie als Kunde aus betriebswirtschaftlichen Gründen sofort loslegen wollen. Sie müssen aber ebenso verstehen, dass wir keine Fragen beantworten möchten, die nachweisbar in unserer Dokumentation oder auf unserem Schulungsvideo erklärt sind. Immer wieder erreichen uns Fragen per Mail oder Fax, die wirklich aufgrund der mitgelieferten Dokumentation erklärt sind. Auch wir bemühen uns, so wirtschaftlich wie möglich zu arbeiten und das mit Softwareprodukten, die preislich wirklich niedrig angesiedelt sind. Für eine gute Zusammenarbeit ist es deshalb wichtig, dass Sie gut vorbereitet Ihre Fragen stellen.

Die Firma AFS bietet auch Schulungen zu Ihren Programmen, wenn Sie eine komplette Einweisung wünschen; sowohl in Schulungscentern wie auch bei Ihnen Vor Ort.

Wichtiger Hinweis:

Egal, für welche Art der Hotline-Unterstützung Sie sich entscheiden, wichtig ist, dass Sie im Vorfeld einige Punkte klären, damit wir Ihnen schnell und ohne Umstände helfen können.

Bitte halten Sie bei jedem Anruf folgende Daten bereit:

- Kunden-Nummer
- Rechnungs-Nummer
- Programmname
- Genaueste Version-Nummer
- Seriennummer

Bei Fragen zu unserer Software AFS-Maskeneditor können Sie sich an unseren kostenpflichtigen Hotline-Service wenden:

Tel. 0900 - 1000 795 (1,70 EUR/min, ggf. abweichende Mobilfunkpreise)

Oder entschliessen Sie sich für einen AFS Servicevertrag mit einer Laufzeit von einem (1) Jahr an. (Nähere Informationen liegen bei der Lieferung oder fordern Sie hierzu Info´s an) .

Mail Service:

Wir bieten auch drei Monate nach Kauf der Software, der aber auf die aktuelle Version beschränkt ist, kostenfrei Hilfe per E-Mail:

Hilfe@afs-software.de

Internetservice:

Häufig gestellte Fragen mit dessen Antworten können Sie immer auch im Internet unter www.afs-software.de/support.htm nachlesen!

Vorwort

Durch die von Ihnen erworbene Software können grundlegende Veränderungen an der Funktionsweise des AFS-Kaufmann vorgenommen werden. Sie sollten daher zumindest mit den Masken und dem Funktionsumfang des Kaufmanns vertraut sein. Desweiteren werden im Nachfolgenden Möglichkeiten aufgezeigt um den Kaufmann wiederherzustellen. Dies kann u.U. nicht immer unproblematisch sein.

Bevor Sie nun beginnen die Masken zu verändern (Maske <=> Fenster), sollten Sie eine komplette Sicherung des Kaufmanns vornehmen. Es wird auch empfohlen für zukünftige Einstellungen vorerst eine Testumgebung einzurichten, in die sie dann die Datenbestände importieren können.

Für einen Verlust der Daten durch den Maskeneditor kann keine Haftung übernommen werden.

Bei Tests des Maskeneditors unter Windows 9x-Systemen traten Speicher-schutzverletzungen auf. Es wird daher empfohlen die Masken auf einem NT-System zu bearbeiten. Beim Kaufmann erscheint im Fehlerfall in der Symbolleiste im Kopf des MDIfähigen-Fensters (Hauptfenster) ein rotes Fragezeichen als Icon. Bei einem Klick auf das Fragezeichen öffnet sich

AFS-Maskeneditor

das Debugger-Fenster. Hier werden alle Arten von Fehlermeldungen ausgegeben.

Dieses Handbuch ist aufgebaut wie ein Tutorial. Wenn Sie das Handbuch sukzessiv (Seite für Seite) durcharbeiten, werden Sie den gewünschten Lerneffekt erzielen.

Der Maskeneditor

Durch die Implementierung des Maskeneditors können nun betriebliche Anwendungsfälle und Geschäftsprozesse auf die individuellen bzw. branchenspezifischen Bedürfnisse der Benutzer bzw. des Betriebes angepasst werden, und das Ganze natürlich mit dem hohen Komfort einer visuellen Hochsprache wie Visual Basic oder Delphi.

Sie benötigen dazu keine Programmierkenntnisse, da sich die Objekte (Buttons, Panels, Datenbankfelder etc...) per „Drag and Drop“ (draufziehen und loslassen) an die gewünschten Positionen beliebig verschieben lassen und anschließend mit Kaufmann-internen vordefinierten, formularspezifischen Funktionen verknüpft werden können.

Falls Sie dennoch Kenntnisse in höheren Programmiersprachen haben, wird Ihnen die Arbeitsweise des Maskeneditors sofort bekannt vorkommen.

In diesem Tutorial möchten wir Ihnen die neuen Möglichkeiten anhand einiger Beispiele näher erläutern.

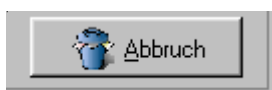
Installation überprüfen

Wenn die Installation des Maskeneditors erfolgreich war, finden Sie im AFS-Kaufmann den folgenden neuen Menüpunkt.



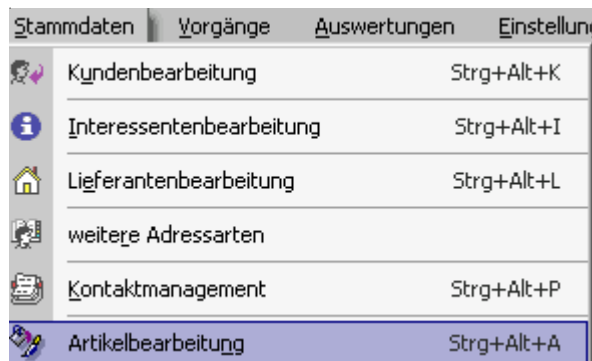
Öffnen Sie den Maskeneditor noch nicht, oder falls Sie es schon getan haben schließen Sie ihn wieder.

Klicken Sie zum Schließen auf den Button „Abbruch“.



Bevor Sie beginnen ein Formular anzupassen, sollten Sie dieses zunächst öffnen. Wir nehmen nun als Beispiel die Artikelbearbeitung.

Klicken Sie auf Stammdaten -> Artikelbearbeitung.



Gehen wir davon aus, uns würde das Feld Zusatznummer stören. Wir brauchen es nicht - der Platz wird verschwendet und außerdem speichern die Mitarbeiter andauernd falsche Sätze darin ab.

Zusatznummer	<input type="text"/>
Warengruppe	test <input type="button" value="..."/>
Umsatzsteuer	16 %

Nun öffnen wir den Maskeneditor.

Die Fenster des Maskeneditors

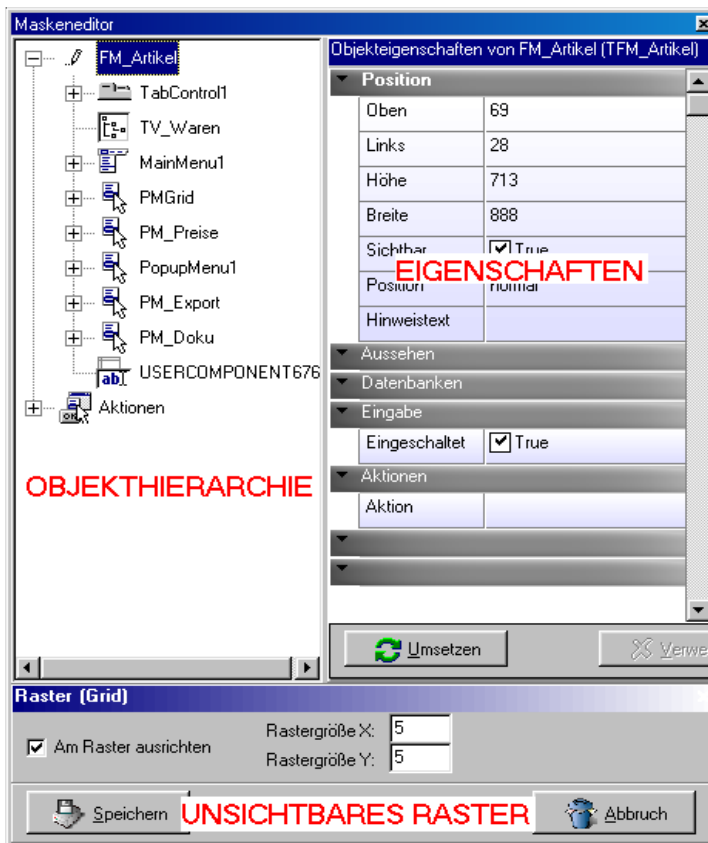
Die Kontrolleiste



In der Kontrolleiste befinden sich unter den Tabs „Standard“ und „Datenbank“ einige bekannte Steuerelemente der Programmiersprache Delphi. Diese **Objekte** können über das „Drag and Drop“-Verfahren auf die bestehenden Formulare gezogen werden und anschließend über das Fenster „Maskeneditor“ diverse Eigenschaften und Aktionen zugewiesen bekommen. Dazu kommen wir später...

AFS-Maskeneditor

Das Fenster „Maskeneditor“



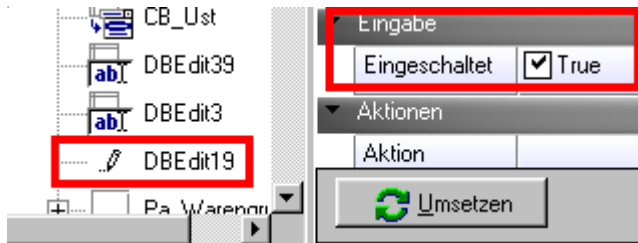
Festlegen von Eigenschaften

Zunächst möchten wir das Feld „Zusatznummer“ deaktivieren. Es sollen keine Nummern mehr reingeschrieben werden. Wir markieren das Feld.



Ein schwarzer Kasten zeigt die Selektion wie im Bild.

Das Objekt des Typs „DBEdit“ und dem Namen „DBEdit19“ wird im Fenster „Maskeneditor“ mit einem Stift versehen. Dies bedeutet, es ist für die Bearbeitung selektiert.



Die Objekteigenschaft „Eingeschaltet“ unter dem Pull-Down-Menü „Eingabe“ hat einen Haken. Wenn wir das Feld deaktivieren möchten, entfernen wir diesen einfach durch einen Klick auf das Kästchen.

Nach einem Klick auf den „Umsetzen“-Button, welcher nach jeder Änderung betätigt werden muss, werden die Eigenschaften auf das Formular und das entsprechende Objekt übertragen.



Unterschiedliche Objekte haben individuelle objektspezifische Objekteigenschaften. Natürlich hat jedes Textfeld die selben Eigenschaften, aber eben nicht unbedingt mit den gleichen gesetzten Parametern.

Nun wählen wir die Schaltfläche „Speichern“. Der Kaufmann hat sich Ihre Änderungen gemerkt. Ab sofort können Sie nichts mehr in das Feld „Zusatznummer“ schreiben, weil Sie nämlich die Eigenschaft „Eingeschaltet“ mit dem Parameter „False“ versehen haben. „False“ bedeutet „falsch“ oder „nein“, „True“ bedeutet „wahr“ oder „ja“.

Leider hat dies zur Folge, dass Sie diese Einstellung nur über die Objekthierarchie rückgängig machen können, da sich das Textfeld nun auch für den Maskeneditor nicht mehr selektieren lässt.

Gehen Sie dazu wieder in den Maskeneditor während sie das geänderte Formular geöffnet haben.

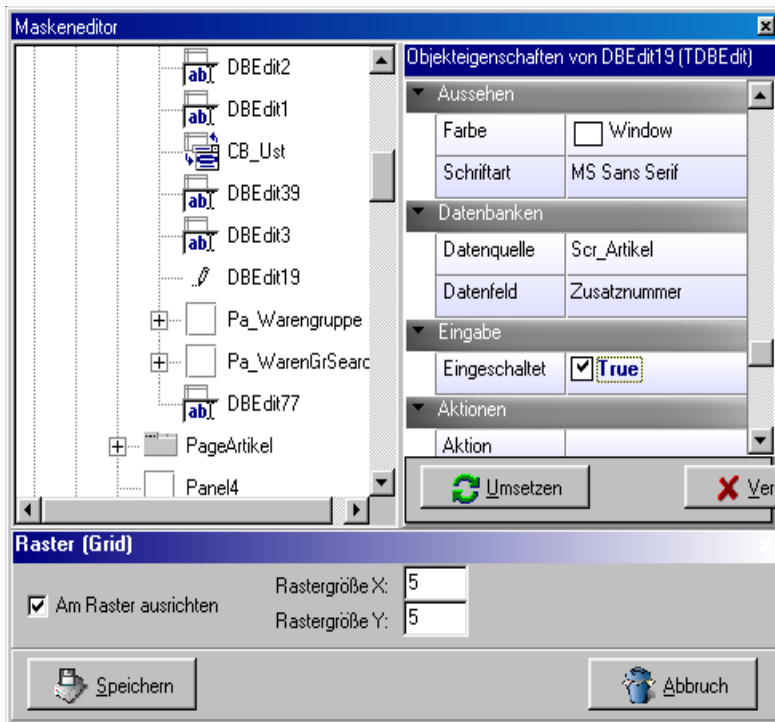
Sie wissen, dass es sich um ein Textfeld mit dem Objektname „DBEdit“ handelte.

Wählen Sie

„FM_Artikel“ (Formular „Artikel“) ==> „Tab Control1“ (Karteikarten oder „Reiter“) ==> „Panel1“ (Elemente-Gruppierung sichtbar als grauer Kasten)

und klicken Sie jedes DBEdit-Objekt mal durch, bis Sie einen schwarzen Rahmen um das gewünschte Feld haben.

Sie werden wieder auf den Namen „DBEdit19“ stoßen.



Aktivieren Sie den Haken „Eingeschaltet“ im Pull-Down-Menü „Eingabe“ wieder, betätigen Sie den Button „Umsetzen“ und anschließend die Schaltfläche „Speichern“. Nun ist es wieder möglich Text in das Textfeld zu schreiben.

Näheres zu den Objekten

Auf den letzten Seiten sprachen wir von Objekten. Das Fenster „Kontrollleiste“ ist eine Toolbox, welche Objekte beinhaltet, die auf ein Formular gezogen werden können.

Es ist nun klar, dass die Textfelder, Buttons, Datenbanktabellen und Labelfelder nichts weiter sind als Objekte.

Ein Objekt hat folgende Merkmale:

- Es besitzt Eigenschaften und Parameter (z.B. Position etc)
- Es hat nur formularspezifische Aktionen/Funktionen
- Es hat einen festen Platz in der Objekthierarchie

Die Objekthierarchie ist die Anordnung und Nacheinanderreihung der verschiedenen Objekte in der jeweiligen topologischen Reihenfolge. Ein Formular „FM_Formularname“ (bsp. FM_Artikel) ist das übergeordnete Objekt. Auf ihr werden alle weiteren Objekte in der jeweiligen Reihenfolge platziert. Einige Objekte (z.B. „Panels“) können eine neue Untergruppierung von Objekten auf einem Formular bilden, und stehen daher topologisch über den enthaltenen Objekten.

An dieser Stelle möchten wir die einzelnen Steuerobjekte genauer erklären

- zunächst der Reiter „Standard“:

Buttons:

Buttons sind Schaltflächen (z.B. „OK“, „Ja“, „Nein“, „Hilfe“, „Abbrechen“). Schaltflächen können Aktionen/Ereignisse zugewiesen werden. Diese Aktionen werden sinnvollerweise von dem aktuellen Formular „FM_Artikel“ zur Verfügung gestellt.

Wir betrachten nun die einzelnen Eigenschaften eines Button-Objekts:

Oben	4
Links	2
Höhe	713
Breite	888
Sichtbar	<input checked="" type="checkbox"/> True
Position	normal
Hinweistext	
Eingeschaltet	<input checked="" type="checkbox"/> True
Aktion	

Die Eigenschaften „Oben“ und „Links“ geben die x- und y-Position des Objekts von der linken oberen Ecke des Buttons an. Die Position lässt sich also erstmal durch das „Drag and Drop“-Verfahren bestimmen, und zweitens durch Eingabe der Koordinaten.

„Höhe“ und „Breite“ legt die Abmessungen des Objekts fest. Auch diese Eigenschaften lassen sich per „Drag and Drop“ lösen.

Die Eigenschaft „Sichtbar“ schaltet Objekte unsichtbar. Dies ist beispielsweise sinnvoll, wenn man etwas kurzzeitig unsichtbar machen will, was man nach einiger Zeit wieder brauchen könnte, und was keinen Platz verschwenden soll.

„Hinweistext“ gibt in einem gelben Kasten und links unten vom Kaufmann-Fenster einen Hinweistext an, wenn man sich mit der Maus einige Sekunden über einer Schaltfläche befindet und diese nicht bewegt.

Die Option „Eingeschaltet“ kann einen Button wie in dem vorigen Beispiel bei der Deaktivierung des Textfeldes ausschalten.

Die Option „Aktion“ weist dem Button die vordefinierten formularspezifischen Funktionen zu.

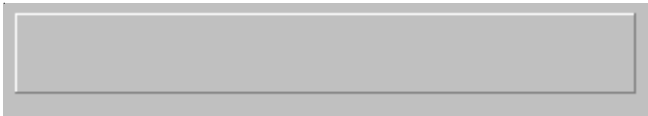
Panels

Panels sollen Objekte gruppieren und als solche kennzeichnen. Es gibt in der Kontrollleiste 2 Arten von Panels.

AFS-Maskeneditor



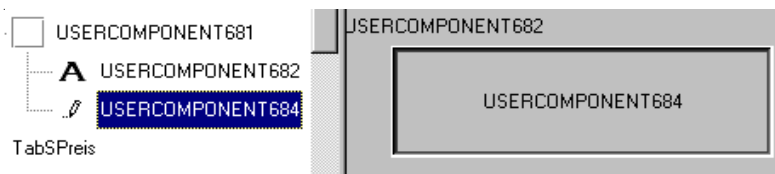
1.) Solche, die einen grauen Kasten auf einem Formular aufziehen und anschließend Gruppierung darauf erlauben



2.) Solche, die einen grauen Kasten auf einem Formular aufziehen, welcher mit einem Text an der linken oberen Ecke zur Kennzeichnung der Gruppe versehen ist und anschließend Gruppierung darauf erlauben



Beachtenswert ist, dass in der Objekthierarchie die Objekte, die auf einem Panel angeordnet werden, dem Panel untergeordnet sind.



Somit hat man eine sichtbare Gruppierung durch den Kasten sowie eine Maskeneditor Interne.

Die Eigenschaften eines Panel-Objekts:

Uden	zbb
Links	9
Höhe	135
Breite	320
Sichtbar	<input checked="" type="checkbox"/> True
autom. Größe	<input type="checkbox"/> False
Position	normal
Hinweistext	
Tabposition	11
Text	USERCOMPONENT681
Textausrichtung	zentriert
Farbe	<input type="checkbox"/> BtnFace
Schriftart	MS Sans Serif
Eingeschaltet	<input checked="" type="checkbox"/> True
Aktion	

Die ersten vier Einstellungen „Oben“, „Links“, „Höhe“ und „Breite“ sind wieder gleich wie bei dem Button-Objekt. Hier können die Einstellungen Pixelgenau vorgenommen werden (als Alternative zu dem „Drag and Drop“-Verfahren).

Die Eigenschaft „sichtbar“ ist ebenfalls äquivalent mit dem Button-Objekt. Die Eigenschaft „Position“ gibt an, wie sich das Panel in dem Fenster ausrichten soll, wenn es größer und kleiner geschoben wird. Dazu gibt es folgende Einstellmöglichkeiten:

- Das Panel lässt sich nicht verändern und bleibt in seiner Ursprungposition.
- Das Panel lässt sich beim „In-die-Breite-ziehen“ des Fensters verändern, bleibt allerdings in seiner Position beim „In-die-Länge-ziehen“.
- Das Panel lässt sich beim „In-die-Länge-ziehen“ des Fensters verändern, bleibt allerdings in seiner Position beim „In-die-Breite-ziehen“
- Das Panel passt sich in der Fensterfläche immer Maximal an.

Die Eigenschaft „Hinweistext“ gibt einen Hinweis am linken unteren Rand des Kaufmanns aus.

Die Eigenschaft „Tabposition“ gibt die Reihenfolge beim „Durch-die-Objekte-springen“ mit der Tabulatortaste an.

Die Eigenschaft „Text“ gibt bei Panels mit Hinweistext den Text an, der an der linken oberen Ecke der Gruppierung stehen soll.

Die Eigenschaft „Farbe“ verändert die Farbe der Fläche des Panels. Sie ist standartmäßig grau.

Die Eigenschaft „Schriftart“ erlaubt das Ersetzen der Standartschriftarten durch individuelle. So ist es z.B. möglich den Kaufmann in der betriebsinternen Firmenschrift erstrahlen zu lassen und somit der Unternehmenskultur anzupassen.

Die Eigenschaften „Eingeschaltet“ und „Aktion“ sind äquivalent mit den Eigenschaften des Button-Objekts.

Labelfelder

Labelfelder sollen Textbereiche definieren. Dabei handelt es sich wirklich nur um eine Textausgabe auf einem Fenster - also nichts außergewöhnliches.

Aus diesem Grund erläutern wir jetzt die Eigenschaften:

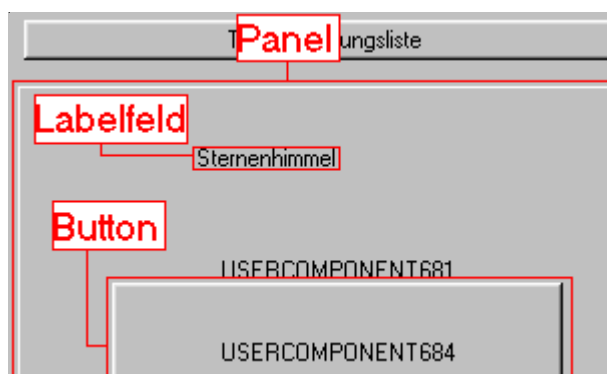
Oben	29
Links	89
Höhe	13
Breite	69
Sichtbar	<input checked="" type="checkbox"/> True
autom. Größe	<input checked="" type="checkbox"/> True
Position	normal
Hinweistext	dies ist ein test...
Text	Sternenhimmel
Textausrichtung	linksbündig
Farbe	<input type="checkbox"/> BtnFace
Schriftart	MS Sans Serif
Transparent	<input type="checkbox"/> False
Eingeschaltet	<input checked="" type="checkbox"/> True

Die Eigenschaften „Oben“, „Links“, „Höhe“, „Breite“, „Sichtbar“, „autom. Größe“ und Position sind wieder äquivalent mit dem Button- und Panel-Objekt.

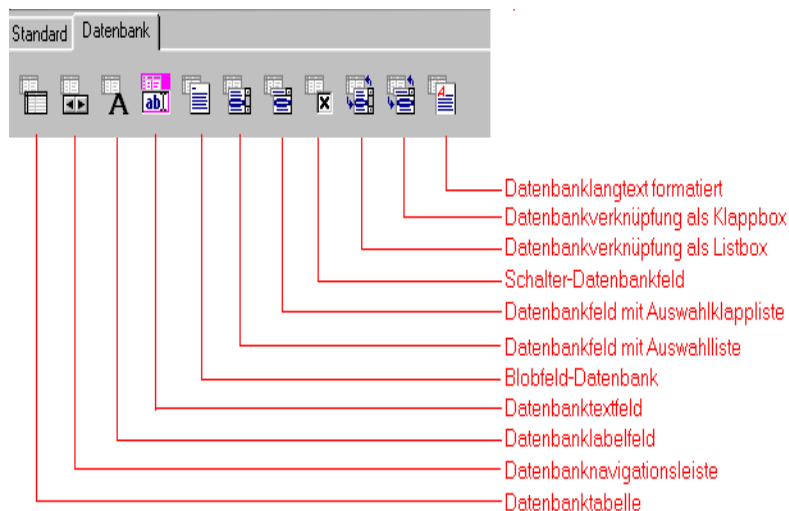
Die Eigenschaft „Hinweistext“ erbt den Hinweistext eines Panels, falls das Objekt auf einem Panel platziert wurde.

Die wichtigste Eigenschaft „Text“ gibt an, was der Textbereich auf dem Bildschirm ausgeben soll. Sie können damit ein Prompt vor Datenfelder und Hinweise schreiben etc...

Die weiteren Eigenschaften sind wieder Äquivalent mit dem Panel- und dem Button-Objekt.



Als nächstes schauen wir uns die Datenbank-Objekte in der Kontrollleiste einmal an:



Datenbanktabellen und Navigationsleisten

Eine Datenbanktabelle ist ein einfaches Anzeige-Objekt zum Darstellen von Datenbanken.

Sehen wir uns nun mal die Eigenschaften einer Datenbanktabelle an:

Oben	270
Links	35
Höhe	170
Breite	275
Sichtbar	<input checked="" type="checkbox"/> True
Position	normal
Hinweistext	
Tabposition	11
Farbe	 Black
Schriftart	MS Sans Serif
Datenquelle	Scr_LagerOrt
Eingeschaltet	<input type="checkbox"/> False
Aktion	

Uns kommen zunächst alle Eigenschaften bekannt vor. Wir wissen nun (nach der sukzessiven Durcharbeitung dieses Handbuchs) was Sie bewirken.

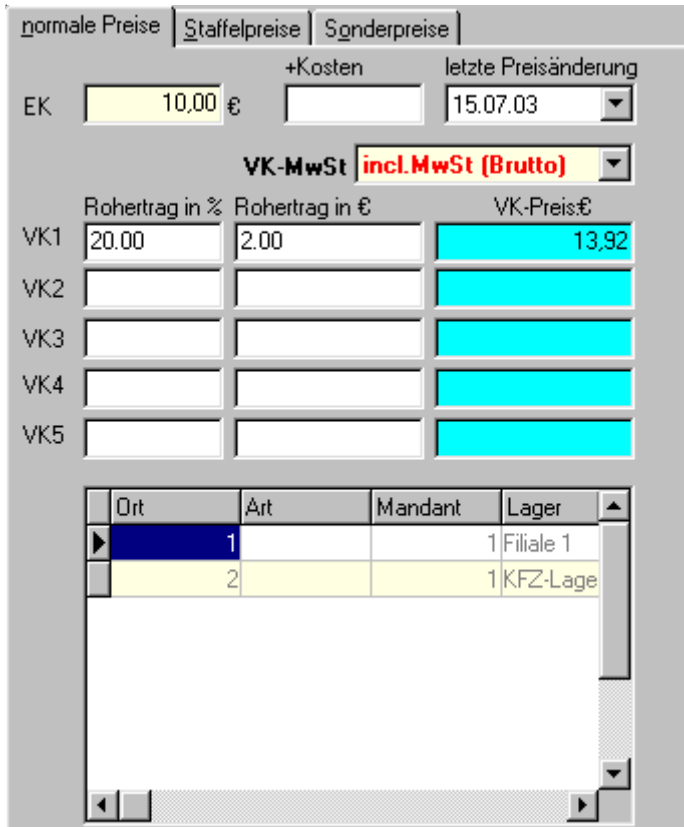
Dennoch gibt es eine neue wichtige Eigenschaft „Datenquelle“.

Hinter dieser Eigenschaft verbirgt sich in einem Pull-Down-Menü die Auswahl für die entsprechenden Schnittstellen zu den einzelnen Tabellen, die Sie gerne Anzeigen lassen wollen. Ich erkläre Ihnen dies nun anhand eines

AFS-Maskeneditor

Beispiels. Öffnen Sie wieder die Artikelbearbeitung (falls Sie diese noch nicht offen haben) und öffnen Sie anschließend den Maskeneditor.

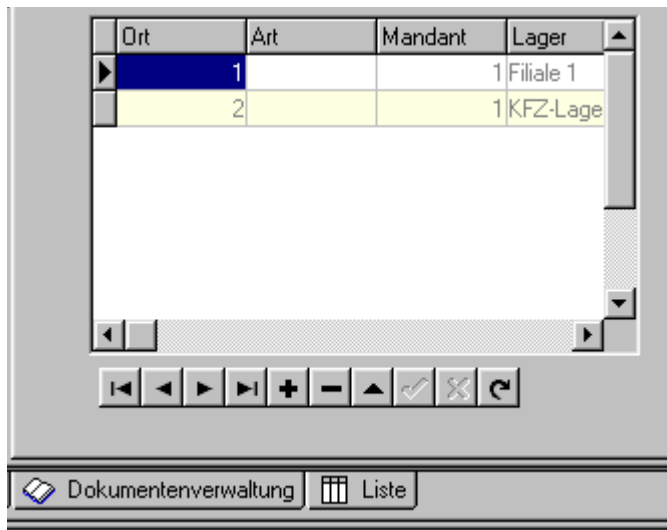
Als Nächstes platzieren wir durch das „Drag and Drop“-Verfahren eine Datenbanktabelle auf der Karteikarte „Preise“ in der Artikelbearbeitung. Wählen Sie am Besten dazu den Standpunkt ganz unten unter den VK-Preisen, wo noch Platz vorhanden ist:



	normale Preise	Staffelpreise	Sonderpreise
EK	10,00 €		+Kosten
			letzte Preisänderung
			15.07.03
			VK-MwSt
			incl. MwSt (Brutto)
	Rohertrag in %	Rohertrag in €	VK-Preis:€
VK1	20.00	2.00	13.92
VK2			
VK3			
VK4			
VK5			

Ort	Art	Mandant	Lager
1			1 Filiale 1
2			1 KFZ-Lage

Gehen wir davon aus, wir möchten von hier aus Lagerorte angezeigt bekommen, anlegen, löschen und bearbeiten. Als nächstes wählen wir bei der Eigenschaft „Datenquelle“ den Parameter „Scr_LagerOrt“. Es erscheinen nun (wie in dem Bild auf der letzten Seite) die Lagerorte in der Datenbanktabelle. Nun wählen wir die Datenbanknavigationsleiste aus (das 2. Objekt von links unter dem Reiter „Datenbank“ der Kontrollleiste) und ziehen eine Navigationsleiste unter dem Datenbankfeld auf. Das sollte dann bei Ihnen folgendermaßen aussehen:



Sollten Sie Probleme bei der Skalierung der Navigationsleiste haben, können Sie die Eigenschaften „Höhe“ und „Breite“ etc... benutzen.

Als nächstes muss die Navigationsleiste ebenfalls der Datenquelle „Scr_LagerOrt“ zugewiesen werden. Wählen Sie dazu die Navigationsleiste aus, so dass die Eigenschaften nun im Fenster „Maskeneditor“ angezeigt werden und wählen Sie die Datenquelle.

Position	
Oben	446
Links	40
Höhe	20
Breite	210
Sichtbar	<input checked="" type="checkbox"/> True
Position	normal
Hinweistext	
Tabposition	12
Datenquelle	Scr_LagerOrt
Eingeschaltet	<input checked="" type="checkbox"/> True
Aktion	

An dieser Stelle zeigt die Datenbanktabelle in Verbindung mit der Datenbanknavigationsleiste die Datensätze an. Klicken Sie auf „Umsetzen“ und auf „Speichern“

Von nun an können Sie von dem Reiter „Preise“ ihre Lagerorte sehen, neue anlegen über die Datenbanknavigationsleiste oder Lagerorte löschen etc... all das, was sie von dem Datenbanknavigator des Kaufmanns gewöhnt sind.

Datenbanktextfelder

Datenbanktextfelder sind im Grunde nichts weiter als Labelfelder, die Daten von Datenbanktabellen anzeigen können. Wir benutzen nun die eben angelegte Datenbanktabelle und die Navigationsleiste in Verbindung mit einem Textfeld. Gehen Sie wieder in den Maskeneditor. Klicken Sie wieder auf den Reiter „Datenbank“ der Kontrollleiste, und wählen sie das 3. Objekt von links aus.

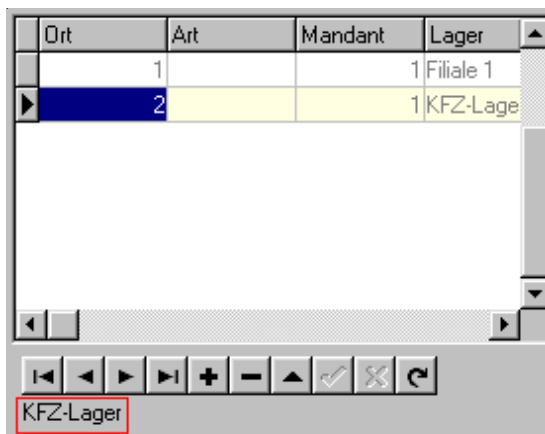
Ziehen Sie ein Datenbanktextfeld unter der Datenbanknavigationsleiste auf und markieren Sie es, so dass in dem Fenster „Maskeneditor“ die Eigenschaften des Objekts aufgelistet werden.

Wählen Sie als Datenquelle wieder „Scr_LagerOrt“ und als Datenfeld „Lager“ aus.

Datenbanken	
Datenquelle	Scr_LagerOrt
Datenfeld	Lager

Klicken Sie auf „Umsetzen“ und „Speichern“.

Navigieren sie nun einmal mit dem Datenbanknavigator, und beobachten Sie dabei das neue Datenbanktextfeld. Ihnen wird immer die ausgewählte Lagerbezeichnung angezeigt.



Die restlichen Eigenschaften des Datenbanktextfeldes sind wie bei dem Labelfeld einzustellen.

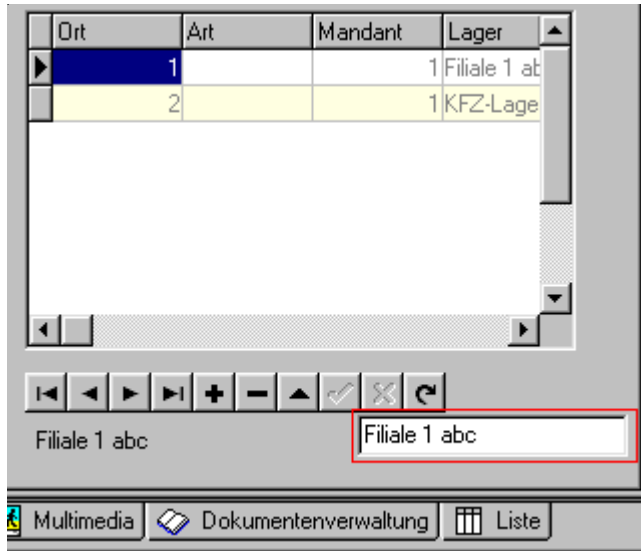
Datenbanktextfelder

Nun öffnen wir wieder den Maskeneditor und fügen das 4. Objekt von links auf dem Reiter „Datenbank“ ein. Hierbei handelt es sich um eine Textbox, die das Editieren und Umändern von einzelnen Datenbankfeldern einer Datenbanktabelle erlaubt.

Platzieren Sie das Feld per „Drag and Drop“ neben dem Labelfeld. und stellen sie die Datenquelle und das Datenfeld wie bei dem Labelfeld ein (siehe Bild auf der linken Seite).

Klicken Sie auf Umsetzen und speichern.

Nun sollte Ihr Formular so aussehen:



Ändern sie mal „Filiale 1“ in „Filiale 1 abc“ und klicken Sie auf den Haken des Navigatorelements um die neue Lagerbezeichnung zu übernehmen. Von nun an können Sie über dieses Feld und dem Datenbanknavigator-element Ihre Lagerbezeichnungen von dem Reiter „Preise“ der Artikelbearbeitung aus ändern.

Einschub: Original Kaufmann wiederherstellen

Wollen Sie die neuen Einstellungen beibehalten? Wir gehen mal davon aus sie brauchen diese nicht. Jetzt haben wir die Möglichkeit einfach alles zu löschen (was wir auch tun), indem wir einen Rechtsklick im Fenster „Maskeneditor“ unter dem Raster-Einstellungen-Bereich machen (z.B. rechts neben speichern). Klicken sie auf „alle Veränderungen löschen“. Schließen Sie das Fenster und den Maskeneditor und klicken sie wieder auf „Artikelbearbeitung“. Nun haben wir wieder den Original-Kaufmann hergestellt. Alternativ können Sie auch die Daten im Verzeichnis %Kaufmann-Pfad%\Einstell löschen. Dies kann bei zerschossenen Masken von Vorteil sein.

AFS-Maskeneditor

EK	10,00 €	+Kosten		letzte Preisänderung	15.07.03
		VK-MwSt	incl.MwSt (Brutto)		
	Rohertrag in %	Rohertrag in €	VK-Preis€		
VK1	20.00	2.00	13,92		
VK2					
VK3					
VK4					
VK5					

BLOB-Felder

Hierbei handelt es sich um ein Objekt, was veränderbare Langtexte anzeigen kann, welche eine größere Länge als 256 Zeichen enthalten können (im Gegensatz zu normalen Datenbanktextfeldern). Der Einsatz von BLOB-Feldern ist datenbankabhängig, da das auswählbare Datenbankfeld dafür vorgesehen sein muss. Z.B. die Artikellangtexte werden in solchen BLOB-Objekten gespeichert.

deutsch	eng	◀	▶

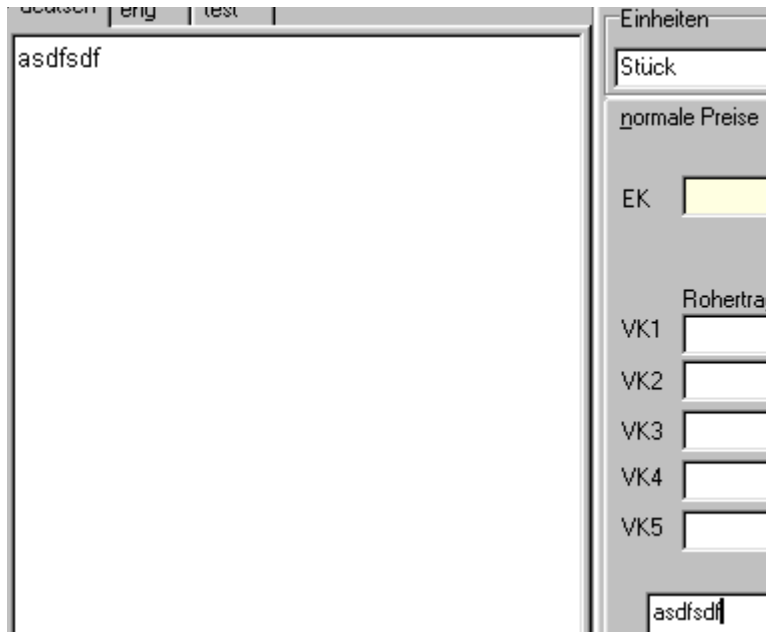
Die Zuweisung einer Datenbankquelle sowie eines bestimmten Feldes erfolgt wie bei den Textfeldern wieder von den Eigenschaften aus. Näheres gibt es nicht zu beschreiben.

Wir öffnen wieder den Maskeneditor und ziehen ein BLOB-Feld an die freie Stelle unter den VK-Preisen. Wir reproduzieren nun das Langtext-Objekt

auf der linken Seite.

Wählen Sie für das BLOB-Objekt die Datenquelle „Scr_Artikel“ und „Langtext“. Wählen Sie „Umsetzen“ und „Speichern“.


Tippen Sie etwas in das neu angelegte Langtextfeld und drücken sie F11 zum Abspeichern. In dem Feld der linken Seite müsste nun das gleiche stehen, wie in dem Feld auf der rechten Seite:



Wir haben nämlich das Langtext-Objekt einfach 2x auf dem Formular, und das andere Langtext-Objekt aktualisiert sofort, nachdem man den Datensatz gespeichert hat.

Entfernen Sie das Objekt wieder aus dem Maskeneditor raus mit einem Rechtsklick auf dem neu erstellten Objekt, und wählen sie „Entferne Objekt“.

Bestätigen Sie mit „Umsetzen“ und „Speichern“.

Alternativ können Sie auch wieder in dem Raster-Bereich des Fensters „Maskeneditor“ mit einem Rechtsklick neben dem Button „Speichern“ das Popup-Feld  lösche alle Veränderungen betätigen.


Datenbankfeld mit Auswahlliste (Listbox):




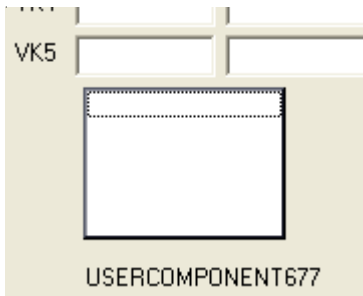
Eine Listbox sieht aus wie ein BLOB-Feld, nur dass die einzelnen Zeilen Listenelemente sind, die man einzeln auswählen kann und welche dann in ein Datenbankfeld geschrieben werden können. Man könnte z.B. Anreden

AFS-Maskeneditor

zur Auswahl stellen (Herr/Frau/Fräulein) oder bestimmte Artikel in den Zusatzfeldern Kennzeichnen.

Öffnen Sie mal die Artikelbearbeitung, den Maskeneditor und platzieren Sie eine solche Listbox  unter den VK-Preisen. Als Nächstes wählen Sie ein

Textfeld  und platzieren dies in der Nähe von der Listbox (ist nicht unbedingt nötig, aber wir lernen dadurch den Maskeneditor besser zu verstehen). Die Maske, die sich nun ergibt, sieht folgendermaßen aus:



Klicken Sie auf die Listbox, und wählen Sie als Datenquelle „Scr_Artikel“, und als Datenfeld „ZusatzFeld01“ aus.


Klicken Sie bei den Paramtern auf die neue Eigenschaft



Nun erscheint ein Button mit 2 Punkten. Wählen Sie diesen aus.

Es öffnet sich ein Fenster „Auswahltext“. In das Fenster können Sie nun in jeder Zeile eine Auswahlmöglichkeit definieren. Gehen wir davon aus, ihre Artikel hätten nur die Eigenschaft, sie seien durch Farben zu unterscheiden. Alternativ - sie verkaufen Drucker und möchten die dazugehörigen Druckerpatronen kennzeichnen. In unserem Beispiel benutzen wir allerdings Farben. Tippen sie folgende Farben in das Fenster ab:



Anschließend schließen Sie das Fenster mit dem  -Icon an der Ecke rechts oben. Wenn es nun Druckerpatronen gewesen wären, könnten wir für Artikel, welche keine Drucker sind, die Auswahlmöglichkeit „kein Drucker“ zusätzlich angeben.

Klicken Sie auf „Umsetzen“ und „Speichern“.

Klicken Sie auf Ihr Datenbanktextfeld (USERCOMPONENT 6xx) und wählen Sie als Datenquelle „Scr_Artikel“ und als Datenbankfeld „ZusatzFeld01“ in den Eigenschaften aus.

Klicken Sie auf „Umsetzen“ und „Speichern“. Beenden sie den Maskeneditor und nochmal die Artikelbearbeitung. Öffnen Sie die Artikelbearbeitung erneut und wählen Sie nun für Ihren Artikel ein Farbe aus. Drücken Sie die <F11>-Taste um den Artikel zu speichern. Von nun an wird für jeden Artikel die Farbe angezeigt.

Datenbankfeld mit Auswahlklappliste

Wird auch „Pull-Down-Menü“ genannt. Eine platzsparende Alternative zur Listbox. Hier können wie bei der Listbox Texteingaben vorgegeben werden, die in dem Feld „Auswahl“ festgelegt wurden. Als nachfolgendes Beispiel fügen wir ein solches Feld wieder in die Artikelbearbeitung ein. Öffnen Sie dazu die Artikelbearbeitung und wählen Sie im Maskeneditor in der Kontrollleiste unter dem Reiter „Datenbank“ das 5. Symbol von rechts.

Wir stellen die Eigenschaften folgendermaßen ein:

Oben	233
Links	36
Höhe	21
Breite	275
Sichtbar	<input checked="" type="checkbox"/> True
Hinweistext	
Tabposition	9
Farbe	<input type="checkbox"/> Window
Schriftart	MS Sans Serif
Auswahl	
Datenquelle	Scr_Artikel
Datenfeld	Bezeichnung
Eingeschaltet	<input checked="" type="checkbox"/> True
Aktion	

Arbeiten Sie dazu das Tutorial für die Listbox durch, und behandeln Sie es genauso wie diese - löschen Sie es wieder, wenn Sie sich im Klaren sind, wie es funktioniert. Damit hat es seine Tätigkeit vorerst erfüllt.

Schalter-Datenbankfeld

Hierbei handelt es sich um ein Schalter-Objekt, welches normalerweise dazu benutzt wird, spezielle Dinge ein- oder auszuschalten (durch einen Haken gekennzeichnet, wie sie sich auch in den Eigenschaften befinden). Damit werden sogenannte BOOL-Werte gespeichert, die entweder wahr oder falsch sein können. Wenn Sie dies ausprobieren möchten, legen Sie ein solches Objekt einfach mal in der Artikelbearbeitung an.

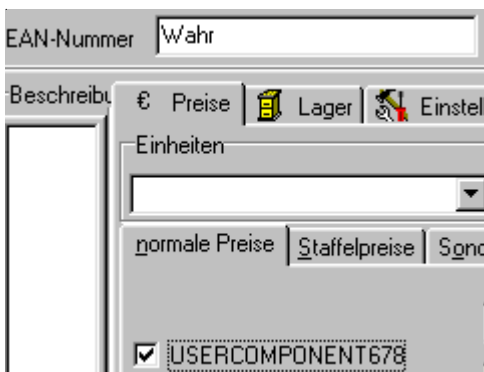
AFS-Maskeneditor



Wählen Sie die Datenbank-Eigenschaften folgendermaßen:

Datenbanken	
Datenquelle	Scr_Artikel
Datenfeld	EANNummer

Aktivieren sie nun das Kästchen und klicken Sie auf F11 um zu speichern.



Im Feld EAN-Nummer erscheint nun „Wahr“. Deaktivieren Sie das Feld und klicken Sie auf F11. Im Feld erscheint nun „Falsch“. „Wahr“ bedeutet eingeschaltet, „falsch“ bedeutet ausgeschaltet. Entfernen Sie die Komponente wieder, da sie hier sinnlos ist. Sie sollte nur zu Demonstrationszwecken und zur Vorführung von BOOL-Werten dienen.

Datenbankverknüpfung als Klappbox

Öffnen Sie nun nochmals die Artikelbearbeitung und starten Sie den Maskeneditor (falls Sie dies noch nicht getan haben). Markieren Sie nun die Klappbox „Umsatzsteuer“.

Im Fenster „Maskeneditor“ bekommen Sie nun die folgenden zusätzlichen Eigenschaften angezeigt:

Datenbanken	
Datenquelle	Scr_Artikel
Datenfeld	Umsatzsteuer
2.Datenquelle	Scr_Ust
Schlüsselfeld	MwSt_Satz
Anzeigefeld	NMwSt

Hierbei wird aus der Primärschlüssel der Umsatzsteuertabelle:

MwSt_Satz	NMwSt	NEL_Kon
1	16	82
2	7	83
3	0	81

In die Artikeltable eingetragene:

Artikel	Umsatzsteuer	Mandant	Art	Artikelnum
1	1	1		01

Die 1. Datenquelle ist somit diejenige, in die der Primärschlüssel eingetragen wird, und die 2. Datenquelle diejenige, in die der Primärschlüssel steht.

Im Schlüsselfeld werden die Primärschlüssel gespeichert, und im Anzeigefeld ist das eingetragen, was in der Klappbox als Auswahl zur Verfügung steht. Somit haben Sie komplette Datenbankrelationen definiert.

In der Kontrollleiste finden Sie ein solches Objekt unter dem Reiter „Datenbank“, zweites Icon auf der rechten Seite.

Datenbanklangtext formatiert

Hierbei handelt es sich um ein BLOB-Feld, welches formatierte Texteingaben speichern kann. Formatiert bedeutet, dass der Text in der Größe änderbar ist, Fett hervorgehoben oder schräg gestellt werden kann. In der Artikelbearbeitung das Feld Artikellangtext ist im Grunde auch kein einfaches BLOB-Feld, sondern ein BLOB-Feld, welches Formatierungen zulässt und gehört somit zum Typ „Datenbanklangtext formatiert“. Wenn Sie ein solches Objekt im Fenster „Artikelbearbeitung“ unter den VK-Preisen platzieren, wird Ihnen auffallen, dass sie nun über die Symbolleisten, welche aus Programmen wie Word bekannt sind,



die Größe des Textes ändern und mit diesen Formatieranweisungen abspeichern können.

Die Eigenschaften sind desweiteren äquivalent mit dem BLOB-Feld.

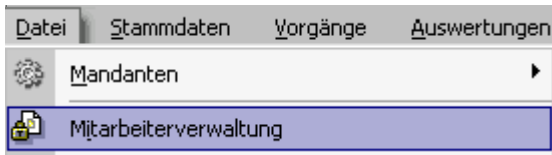
Nun haben wir alle Objekte und deren Anwendungsgebiete näher erläutert.

Benutzerspezifische Masken

Im Kaufmann ist es durch die Mitarbeiterverwaltung möglich, Gruppierungen vorzunehmen und entsprechende Rechte zu vergeben. So sollten z.B. Mitarbeiter, die an einer Bestellannahme oder im Support sitzen, nicht in der Lage sein den Maskeneditor aufzurufen.

Dies erreichen Sie, indem Sie die Mitarbeiterverwaltung öffnen.

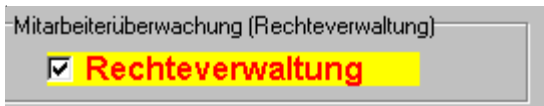
AFS-Maskeneditor



Wechseln Sie auf die 2. Fläche „Rechte Teil 1“:



Hier können Sie für spezielle Benutzergruppen den Maskeneditor und andere Formulare deaktivieren. Diese Einstellungen werden je Gruppe gespeichert (wenn Sie Masken im Maskeneditor entwerfen, und sich mit einem Admin-Account vorher angemeldet haben). Die Anmeldeprozedur erscheint, wenn folgender „BOOL-Wert“ unter Datei -> Einstellungen gesetzt wurde:



Wenn Sie für eine Gruppe Masken entwerfen möchten, aktivieren Sie den Maskeneditor vorerst, bearbeiten Ihre Masken und beenden diesen, gehen in die Mitarbeiterverwaltung und deaktivieren ihn auf dem Tab/Reiter „Rechte“ wieder. Achten Sie dabei darauf, dass Sie der Gruppe Admin nicht den Maskeneditor nehmen.

Beenden Sie den Kaufmann und starten Sie ihn neu. Die neuen Masken sollten nun für die jeweilige Gruppe angelegt sein.

Die Formulare werden im Kaufmann-Verzeichnis unter : „Einstell\“ als TXT-Dateien gespeichert. Der Auslagerungscode wird dabei in Delphi geschrieben. Bitte vermeiden Sie das manuelle Bearbeiten der Dateien. Sollten Fehler bei der Bearbeitung auftreten, oder grobe Veränderungen vorgenommen worden sein, die das normale Weiterarbeiten im Kaufmann nun verhindern, können Sie die TXT-Dateien in diesem Verzeichnis löschen um den Originalkaufmann wieder herzustellen. Alternativ gibt es weiter hin die Option „lösche alle Veränderungen“ im Fenster Maskeneditor mit einem Rechtsklick neben den „Speichern“-Button (wie bereits erwähnt).

Aktionen

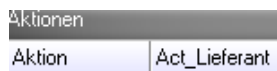
Wie bereits besprochen, können Buttons und anderen Objekten Aktionen zugewiesen werden. Diese Aktionen sind von Formular zu Formular verschieden. Jedes Formular stellt nur diejenigen Aktionen zur Verfügung, die in dem Funktionsumfang des geöffneten Formulars enthalten sind. So lassen sich z.B. durch das Platzieren von Buttons und das Zuweisen von Aktionen oft benutzte Funktionen des Kaufmanns an eine leicht erreichbare Stelle verlagern, und sich somit die Anwendungsfälle erleichtern. Die Aktionen für die Stammdaten werden nun beschrieben:

Aktionen für die Artikelverwaltung

Act_Lieferant

Sie arbeiten in einem Unternehmen, welches Just-In-Time-Lieferungen vornimmt (Lieferung bei Nachfrage)

Wählen Sie einen Button und weisen diesem die Aktion „Act_Lieferant“ zu.



Nun können Sie schnellstmöglich für den aktiven Artikel den/die Lieferanten herausfinden. Dies sieht dann ungefähr so aus:



Das Fenster „Lieferant für den Artikel suchen“ wird sofort eingeblendet, wenn sie auf diesen Button klicken.

Act_Lieferant

Lieferanten suchen

Act_LieferGrid

Spalteneditor

Act_Rabattgruppen

Rabattgruppen und Sonderpreise

Act_GridEdit

Spalteneditor

Act_Lagerbuchung

AFS-Maskeneditor
Lagerbewegung buchen

Act_Artikel_Kunde
welcher Kunde hat den Artikel

Act_Artikel_Lieferant

Act_DruckerAssistent
Druckassistent

Act_Drucken

Act_Artikel_Umsatz

Act_STListeGrid
Spalteneditor

Act_StArtikel
Artikel auswählen

Act_Serienbrief
Textverarbeitung (incl. Serienbrieffunktion für Lieferanten)

Action_InsertLief
Lieferanten einfügen

Act_StArtikel3
Artikel auswählen

Action_InsertStueck
Stücklistenartikel einfügen

Action_SNSuche

Action_Preismodul
Preismodul (Preise neu berechnen)

Action_GesamtInventur
Inventur des gesamten Lagers

Action_Teilinventur
Inventur begrenzt auf ein Lager

Action_Umsatz
Statistik

Action_InsertStueck2
Artikel einfügen

Act_StArtikel2

Artikel auswählen

Action_EinheitDown

Grundpreiseinstellungen zeigen

Action_EinheitUp

Grundpreiseinstellungen ausblenden

Action_PreiseOff

Preise ausschalten

Action_Selection

Selektion dieses Datensatzes

Action_Selection_off

Selektion dieses Datensatzes aufheben

Action_Selection_alloff

Selektion aller Datensätze aufheben

Action_Kopieren

Action_Teileverwendung

Teileverwendungsliste des Artikels(in welchen Stücklisten ist dieser Artikel)

Action_StKalkEK

alle Artikel mit Stücklisten: EK neu berechnen anhand der Stückliste
(der EK wird aus der Summe der Stücklisten EK's gebildet)

Action_StKalkVKS

alle Artikel mit Stücklisten: VK1 neu berechnen anhand der Stückliste
(der VK wird aus der Summe der Stücklisten VK's gebildet)

Action_StKalkVKK

alle Artikel mit Stücklisten: EK neu berechnen anhand der Stückliste
(der EK wird aus der Summe der Stücklisten EK's gebildet)

Action_Pfandverwendung

Teileverkettungsliste des Artikels (mit welchen Artikeln ist dieser Verkettet)

Action_ExcelExport

Export zu Excel

Action_WordExport

Export zu Word

Action1

Export zu Excel

AFS-Maskeneditor
Action Mehrlager View
Mehrlageranzeigen

Action GridDokuEdit
Spalteneditor

Action Shopdaten
Zusatzdaten für den Onlineshop (Diese Optionen werden nur vom
Enterpriseshop ab V2.0 und Standard-Shop ab V3.0 unterstützt)

Action LIEF EAN zu Nummer
wandelt die EANNummer des Lieferanten zur Artikelnummer des Artikels

Action LIEF EAN zu EANNummer
wandelt die EANNummer des Lieferanten zur EANNummer des Artikels

Action LIEF Bestell zu EANNummer
wandelt die Bestellnummer des Lieferanten zur EANNummer des Artikels

Action Undo
Rückgängig

Action Cut
Ausschneiden

Action Copy
Kopieren

Action Past
Einfügen

Action SNGrid
Spalteneditor

Action Eigenschaften
Artikeleigenschaften anzeigen

Action History
Anzeige aller Ein- und Verkäufe mit Preisen

Action Zubehoehr
Zubehörartikel definieren

Action Lieferrueckstand
Lieferrückstandsliste (Ihre Rückstände aus Aufträgen)

Action Bestellrueckstand
Bestellrückstandsliste (Lieferanten-Rückstände aus Bestellungen)

Action Geraete History

Geraete-History

Action_FertigungDruck
Stückliste drucken

Action_Stueckliste_Einbuchen
Fertigung ins Lager buchen (Hauptartikel zubuchen und Stückliste ausbuchen)

Action_StuecklistenDruck
Stückliste drucken

Action_StuecklisteExcel
Export zu Excel

Action_InsertStueck3
Artikel einfügen

Action_DruckerUpdate
Auswahl der Druckänderungen

Action_Sel_Filialpreise
Preise den ausgewählten Filialen zuweisen

Action_All_Filialpreise
Preise allen Filialen zuweisen

Action_Filialauswahl_Close
Filialen ausblenden

Action_FilialpreisSpalten
Spalteneditor

Action_Faktor_Aktuell
Währungsfaktor aktualisieren

Action_EKFremzuLieferEK
EK umrechnen und übergeben

Action_Lieferant_Zeigen
Lieferant bearbeiten

Action_ST_Artikel
Artikel bearbeiten

Action_ST2_Artikel
Artikel bearbeiten

Act_STListeGrid2
Spalteneditor

AFS-Maskeneditor
Action_ST3_Artikel
Artikel bearbeiten

Act_STListeGrid3
Spalteneditor

Action_ProduktionToExec
Export zu Excel

Action_ProduktionDruck
Produktionsstückliste Drucken

Aktionen für das Kontaktmanagement

Action_SpaltenHis
Spaltendesigner

Action_DokumentWaehlen
Dokument bearbeiten

DataSetCancel1
Abbrechen

DataSetDelete1
Löschen

DataSetEdit1
Bearbeiten

DataSetFirst1
Erster

DataSetInsert1
Einfügen

DataSetLast1
Letzter

DataSetNext1
Nächster

DataSetPost1
Übernehmen

DataSetPrior1
Vorheriger

DataSetRefresh1

Aktualisieren

DataSetCancel2

Abbrechen

DataSetDelete2

Löschen

DataSetEdit2

Bearbeiten

DataSetFirst2

Erster

DataSetInsert2

Einfügen

DataSetLast2

Letzter

DataSetNext2

Nächster

DataSetPost2

Übernehmen

DataSetPrior2

Vorheriger

DataSetRefresh2

Aktualisieren

Action_Filter1

Filter setzen (suchen)

Action_SpraltenDoku

Tabellendesigner

Action_Filter2

Filter setzen (suchen)

Action_SpraltenPart

Tabellendesigner

Action_GruppeNeu

Adressgruppen bearbeiten

Action_DruckenAus

Drucken mit Auswahl

AFS-Maskeneditor

Action PrintCFG
Einstellungen

Action Auftrag
Vorgang bearbeiten

Action Artikel
Artikel bearbeiten

Action VorgangNeu
neuen Vorgang anlegen

Action Helfer
der Helfer erklärt das Fenster

Action GruppenSuchen
Adressgruppenrecherche

Action Adressarten
weitere Adressarten

Gruppen Insert
Einfügen

Gruppen Delete
Löschen

Action Selection
Selektion dieses Datensatzes

Action Selection off
Selektion dieses Datensatzes aufheben

Action Selection alloff
Selektion aller Datensätze aufheben

Action Zwischenablage
Adresse in die Zwischenablage kopieren (kann in jedem Programm mit (SHIFT+Einfg) eingefügt werden).

Action AdresseSplaten
Spalteneditor

Gruppen Post
Gruppe speichern

Action KontaktSuchen
Kontaktrecherche und Terminrecherche

Action Selection all
Seite 40

Selektion aller Datensätze

Action_ExcelExport
Export zu Excel

Action_WordExport
Export zu Word

Action_OutlookExport
Outlook Export

Action_Ansprech_Anrufen
Nummer anwählen

Action_Ansprech_Mailen
E-Mail schreiben

Action_Adr_SMS
SMS-Senden

Action_Ansprech_SMS
SMS-Senden

Action_Anruf_Handy
Handy anrufen

Action_Ansprech_Handy
auf Handy anrufen

Action_Outlook_CFG
Outlook Kontaktgruppe wählen

Action_TelCD
Import aus Telefon CD

Action_AdrVollCD
Adresse vervollständigen (aus Telefon CD)

Action_TEL_PlzOrt_Suche
aus Telefon CD: PLZ ORT Suche

Action_Tel_AdressCheck
Adresse Prüfen (aus Telefon CD)

Action_MaskenEd
Maskeneditor

Action_Kon_ArtCfg
Kontaktarten einrichten

Action_KontaktTimerStart

AFS-Maskeneditor
automatische Zeiterfassung starten

Action_KontaktTimerStop
automatische Zeiterfassung stoppen

Action_KontaktListeEdit
Liste direkt editierbar (an/aus)

Action_Kategorien
Kategorien wählen

Action_PartKategorien
Kategorien wählen

Action_AdrKategorien
Kategorien wählen

Action_PartnerListeEdit
Liste direkt editierbar (an/aus)

Action_KontStatusEdit
Kontaktstatus bearbeiten

Action_PartnerRecherche
Ansprechpartner-Kategorierecherche

Action_KatRecherche
Kategorie-Recherche

Action_Partner_Doku
Kontakt dem aktuellen Ansprechpartner zuweisen

Action_KontaktPartnerFilter
Kontakte dieses Ansprechpartners

Action_Ver_Spalten
Spalteneditor

Action_VerbindTxt
Verbindungstexte bearbeiten

Action_Verbind_Tauschen
Tauschen der Beziehung

Action_Verbind_Adrs
Beziehungsadresse suchen

DataSetDelete3
Löschen

DataSetInsert3

Einfügen

DataSetPost3

Übernehmen

Action_Vorgang_Spalten

Spalteneditor

Action_Vorgang_CFG

Öffnet die Registerkarte „Einstellungen“ im Kontaktmanagement.

Aktionen für die Kundenbearbeitung

Act_Rabattgruppen

Rabattgruppen und Sonderpreise

Act_Rabatt

Rabatte bearbeiten

Act_TabDesigner

Spaltendesigner

Act_History

History

Act_Filialen

Filialen anzeigen (untergeordnete Anschriften)

Act_Hauptstelle

Hauptstelle auswählen

Act_Rechnungsjournal

Rechnungsjournal

Act_Lieferschein

Lieferscheinjournal

Act_Angebote

Angebotsjournal

Act_OPs

offene Posten

Act_Abgeschlossen

abgeschlossene Rechnungen

Act_AlleVorgaenge

zeigt alle Vorgänge an

AFS-Maskeneditor

Act DruckerAssistent
Ausdrucksassistent

Act Drucken
Druckt mit den aktuellen Einstellungen.

Act Serienbrief
Textverarbeitung (inkl. Serienbrieffunktion)

Act inKundewandeln
wandelt diese Adresse in einen Kunden um

Act Bestellungen
Bestelljournal

Action Terminliste
Liste der Termine für diesen Kunden

Action Umsatz
Umsatzauswertung

Action VorgangNeu
neuen Vorgang anlegen

Action Gruppen
Adressgruppen

Action Artikel Vorgang
Artikel nach Vorgangsarten gruppiert

Action Selection
Selektion dieses Datensatzes

Action Selection off
Selektion dieses Datensatzes aufheben

Action Zwischenablage
Adresse in die Zwischenablage kopieren (kann in jedem Programm mit (SHIFT+Einfg) eingefügt werden).

Action Selection alloff
Selektion aller Datensätze aufheben

Action ReAnOff
Rechnungsanschrift entfernen

Action Adr SMS
SMS-Senden

Action Anruf Handy

Handy anrufen

Action Ansprech SMS
SMS-Senden

Action Ansprech Handy
auf Handy anrufen

Action ExcelExport
Excelexport

Action WordExport
Wordexport

Action Vertreter Cancel
Vertreter entfernen (leeren)

Action TabDes Doku
Spaltendesigner

Action TelCD
Import aus Telefon CD

Action AdrVollCD
Adresse vervollständigen (aus Telefon CD)

Action TEL PlzOrt Suche
PLZ ORT Suche von Telefon CD

Action Tel AdressCheck
Adresse Prüfen (aus Telefon CD)

Action Lieferrueckstand
Lieferrückstandsliste (Ihre Rückstände aus Aufträgen)

Action Bestellrueckstand
Bestellrückstandsliste(Lieferanten-Rückstände aus Bestellungen)

Action Filiale Leeren
Filialezuweisung entfernen

Action Geraete History
Geraete-History

Action Mitarbeiter Cancel
Betreuer entfernen (leeren)

Action UmsatzCalk
Umsatzanzeige neu berechnen

Action ShopUrl

AFS-Maskeneditor
Shop öffnen

Action_GeraetAnlegen
neues Gerät anlegen

Action_SNManager
Kontaktassistent - Serienbrief, Serienfax, Serienmail, Newsletter, Serien-SMS

Action_Vertreter2_Cancel
Vertreter entfernen (leeren)

Aktionen für die Interessentenbearbeitung

Act_Rabattgruppen
Rabattgruppen und Sonderpreise

Act_Rabatt
Rabattgruppe bearbeiten

Act_TabDesigner
Spaltendesigner

Act_History
Anzeige aller Artikel mit Preisen

Act_Filialen
Filialen anzeigen (untergeordnete Anschriften)

Act_Hauptstelle
Anschrift auswählen

Act_Rechnungsjournal
Rechnungsjournal

Act_Lieferschein
Lieferscheinjournal

Act_Angebote
Angebotsjournal

Act_OPs
Journal der offenen Posten

Act_Abgeschlossen
Öffnet eine Auswertung für abgeschlossene Vorgänge

Act_AlleVorgaenge

zeigt alle Vorgänge an

Act DruckerAssistent
Druckassistent

Act Drucken
Druckt mit den aktuellen Einstellungen

Act Serienbrief
Textverarbeitung (incl. Serienbrieffunktion)

Act inKundewandeln
wandelt diese Adresse in einen Kunden um

Act Bestellungen
Bestelljournal

Action Terminliste
Liste der Termine für diesen Kunden

Action Umsatz
Umsatzauswertung

Action VorgangNeu
neuen Vorgang anlegen

Action Gruppen
Adressgruppen

Action Artikel Vorgang
Artikel nach Vorgangsarten gruppiert

Action Selection
Selektion dieses Datensatzes

Action Selection off
Selektion dieses Datensatzes aufheben

Action Zwischenablage
Adresse in die Zwischenablage kopieren (kann in jedem Programm mit (SHIFT+Einf) eingefügt werden).

Action Selection alloff
Selektion aller Datensätze aufheben

Action ReAnOff
Rechnungsanschrift entfernen

Action Adr SMS
SMS-Senden

AFS-Maskeneditor
Action Anruf Handy
Handy anrufen

Action Ansprech SMS
SMS-Senden

Action Ansprech Handy
auf Handy anrufen

Action ExcelExport
Export zu Excel

Action WordExport
Export zu Word

Action Vertreter Cancel
Vertreter entfernen (leeren)

Action TabDes Doku
Spaltendesigner

Action TelCD
Import aus Telefon CD

Action AdrVollCD
Adresse vervollständigen (aus Telefon CD)

Action TEL PlzOrt Suche
aus Telefon CD: PLZ ORT Suche

Action Tel AdressCheck
Adresse Prüfen (aus Telefon CD)

Action Lieferrueckstand
Lieferrückstandsliste (Ihre Rückstände aus Aufträgen)

Action Bestellrueckstand
Bestellrückstandsliste (Lieferanten-Rückstände aus Bestellungen)

Action Filiale Leeren
Filialezuweisung entfernen

Action Geraete History
Geraete-History

Action Mitarbeiter Cancel
Betreuer entfernen (leeren)

Action UmsatzCalk
Umsatzanzeige neu berechnen

Action ShopUrl
Shop öffnen

Action GeraetAnlegen
neues Gerät anlegen

Action SNManager
Kontaktassistent - Serienbrief, Serienfax, Serienmail, Newsletter, Serien-SMS

Action Vertreter2 Cancel
Vertreter entfernen (leeren)

Aktionen für die Lieferantenbearbeitung

Act Rabattgruppen
Rabattgruppen und Sonderpreise

Act Rabatt
Rabattgruppe bearbeiten

Act TabDesigner
Spaltendesigner

Act History
Anzeige aller Artikel mit Preisen

Act Filialen
Filialen anzeigen (untergeordnete Anschriften)

Act Hauptstelle
Anschrift auswählen

Act Rechnungsjournal
Rechnungsjournal

Act Lieferschein
Lieferscheinjournal

Act Angebote
Angebotsjournal

Act OPs
Journal der offenen Posten

Act Abgeschlossen
Öffnet eine Auswertung über abgeschlossene Vorgänge

AFS-Maskeneditor

Act AlleVorgaenge

zeigt alle Vorgänge an

Act DruckerAssistent

Druckassistent

Act Drucken

Druckt mit den aktuellen Einstellungen

Act Serienbrief

Textverarbeitung (incl. Serienbrieffunktion)

Act inKundewandeln

wandelt diese Adresse in einen Kunden um

Act Bestellungen

Bestelljournal

Action Terminliste

Liste der Termine für diesen Kunden

Action Umsatz

Umsatzauswertung

Action VorgangNeu

neuen Vorgang anlegen

Action Gruppen

Adressgruppen

Action Artikel Vorgang

Artikel nach Vorgangsarten gruppiert

Action Selection

Selektion dieses Datensatzes

Action Selection off

Selektion dieses Datensatzes aufheben

Action Zwischenablage

Adresse in die Zwischenablage kopieren: (kann in jedem Programm mit (SHIFT+Eingf) eingefügt werden

Action Selection alloff

Selektion aller Datensätze aufheben

Action ReAnOff

Rechnungsanschrift entfernen

Action_Adr_SMS

SMS-Senden

Action_Anruf_Handy

Handy anrufen

Action_Ansprech_SMS

SMS-Senden

Action_Ansprech_Handy

auf Handy anrufen

Action_ExcelExport

Export zu Excel

Action_WordExport

Export zu Word

Action_Vertreter_Cancel

Vertreter entfernen (leeren)

Action_TabDes_Doku

Spaltendesigner

Action_TelCD

Import aus Telefon CD

Action_AdrVollCD

Adresse vervollständigen (aus Telefon CD)

Action_TEL_PlzOrt_Suche

aus Telefon CD: PLZ ORT Suche

Action_Tel_AdressCheck

Adresse Prüfen (aus Telefon CD)

Action_Lieferrueckstand

Lieferrückstandsliste (Ihre Rückstände aus Aufträgen)

Action_Bestellrueckstand

Bestellrückstandsliste (Lieferanten-Rückstände aus Bestellungen)

Action_Filiale_Leeren

Filialezuweisung entfernen

Action_Geraete_History

Geraete-History

Action_Mitarbeiter_Cancel

AFS-Maskeneditor
Betreuer entfernen (leeren)

Action UmsatzCalk
Umsatzanzeige neu berechnen

Action ShopUrl
Shop öffnen

Action GeraetAnlegen
neues Gerät anlegen

Action SNManager
Kontaktassistent - Serienbrief, Serienfax, Serienmail, Newsletter, Serien-SMS

Action Vertreter2 Cancel
Vertreter entfernen (leeren)

Aktionen für das Fenster „Warengruppe“

Act Warengruppe up
eine Ebene rauf

Act Warengruppe down
eine Ebene tiefer

Act Warengruppe links
eine Ebene zurück

Act Warengruppe rechts
eine Ebene vor

Act Print
Drucken der Warengruppen

Act PreView
zeigt Ihnen eine Druckvorschau an

Act Formularedit
verändern des Ausdrucks

Act Formularselect
wählen Sie das Formular zum Ausdrucken aus

Act GridEdit
Spalteneditor

Act Close

Warengruppen schließen

Act Rabatt

Rabattgruppen und Sonderpreise

Act WarengruppeNeu

neue Warengruppe

Act WarengruppeBearbeiten

Warengruppe bearbeiten

Act WarengruppeLoeschen

Warengruppe löschen

Act UnterwarengruppeNeu

eine untergeordnete Warengruppe anlegen

Action Umsatz

Statistik

Action Zubehoehr

Zubehörartikel definieren

Action BezeichnungIntern

Action SHOP Anzeigen

Shopwarengruppen anzeigen

Aktionen für „Artikelliste nach Warengruppen“

Action Spalten

Spalteneditor

Action Bearbeiten

Artikel bearbeiten

Action Kalkulation

alle in der Liste angezeigten Artikel neu kalkulieren

Action Refresh

Aktualisieren

Action Print

Drucken mit Auswahl

Action PrintCFG

Einstellungen

Action Selection

AFS-Maskeneditor
Selektion dieses Datensatzes

Action_Selection_off
Selektion dieses Datensatzes aufheben

Action_Helfer
der Helfer erklärt das Fenster

Action_ExcelExport
Export zu Excel

Action_WordExport
Export zu Word

Action_Kunden
Kunden suchen

Action_Lieferant
Lieferant suchen

Action_Interesent
Interesent suchen

Action_MDEExp
Artikelübergabe zum MDE-Gerät

Action_MDEFormat
MDE-Format einstellen

Action_MDEDatei
MDE-Datei auswählen

Action_Eigenschaften
Artikeleigenschaften anzeigen

Action_History
Anzeige aller Ein- und Verkäufe mit Preisen

Action_Zubehoehr
Zubehörartikel definieren

Action_Lieferrueckstand
Lieferrückstandsliste (Ihre Rückstände aus Aufträgen)

Action_Bestellrueckstand
Bestellrückstandsliste (Lieferanten-Rückstände aus Bestellungen)

Action_Auswahl
Auswahl

Action Schliessen
Schliessen

Action WarengruppeFind
Warengruppe suchen

Action WarenDragandDrop
Warengruppen verschieben

Action Kopieren

Action SHOP Anzeigen
Shopwarengruppen anzeigen

Bemerkung: Nicht aufgeführte (auch im Nachfolgenden Teil) Aktionen sind äquivalent mit den vorherigen Fenstern.

Aktionen für das Fenster „Lagerbuchung“

Act LagerNeu
neue Lagerbuchung

Act LagerCancel
Lagerbuchung rückgängig machen

Act LagerSave
Lagerbuchung speichern

Act Print
Aktion Drucken mit den Einstellungen des Drucker-Assistenten

Action Preview
Öffnet die Einstellungen für den Druckerassistenten

Act GridEdit
Tabellendesigner wird gestartet

Act Close
Lagerfenster schließen

Act ArtikelSuchen
Artikel durchsuchen

Act SNGridEdit
Öffnet den Tabellendesigner (Seriennummern)

Act SN Verwaltung
Seriennummernverwaltung

AFS-Maskeneditor

Action Lieferant
Lieferant suchen

Action Artikelzeigen
Artikel anzeigen

Action Lieferanten zeigen
Adresse der Lieferanten anzeigen

Action VorBearbeiten
Vorgang bearbeiten

Action Helfer
Der Helfer erkärt das Fenster
Action Bestandszuweise
Mehrbestand dem Hauptlager zuweisen

Action Bestandentfernen
Mehrbestände entfernen

Aktionen für die Inventur

Action Save
Speichern und nächsten Artikel

Action InventErfass
Spaltendesigner (Inventurerfassung)

Action InventBearb
Spaltendesigner (Inventurbearbeitung)

Action ArtikelListe
Spaltendesigner (Artikelliste)

Action Print
Drucken

Action PrintCFG
Einstellungen der Ausdrücke

Action ExcelExport
Export zu Excel

Action WordExport
Export zu Word

Action OK
Schließe das Fenster

Action LagerArtikel

Spaltendesigner (Artikel eines Lager)

Action Inv Artikelzeigen

Artikel zeigen

Action LA Artikelzeigen

Artikel zeigen

Action Artikelzeigen

Artikel zeigen

Action Invent EA

Inventurübergabe dieses Artikels

Action Inventur

Inventurübergabe aller Artikel

Action Delete

Übergebene Inventurwerte entfernen

Action InventurmarkeWeg

Inventur zurücknehmen

Action alleInventurmarkeWeg

Inventur komplett zurücknehmen

Action fehlendenArtikelNullen

Alle Artikel mit Bestand, die nicht in der Inventurliste sind, mit Nullbestand in die Inventur übernehmen!

Action ArtikelSuchen

Artikel suchen

Action Bestandzuweise

Wenn das Mehrlagermodul nachträglich erworben wurde, den Mehrbestand diesem Lager zuweisen oder Bestände, die importiert wurden, jedoch ohne Zuweisung sind, diesem Lager zuweisen!

Action Bestandentferen

Bestände, die keinem Lager zugewiesen sind entfernen! Artikelbestand = Summe der Lagerbestände

Action InvSuche

Inventursuche

Action ArtikelSichern

Artikelliste für spätere Auswertungen sichern

Action InventurLaden

Inventur von einer anderen Filiale einlesen

AFS-Maskeneditor

Action AlleLagerorte

Inventurliste aller Lagerorte anzeigen

Action ProtokollPrint

Protokolldruck

Action Inventurbewertung

Inventurbewertung aller Artikel, die unten sichtbar sind (Suchbegriff von - bis)

Action InventurGesamtbewertung

Inventurgesamtbewertung

Action Hauptlager Bestand

Hauptlager als Bestand anzeigen

Action Standort

Standorte bearbeiten (in einem Lager bzw. einer Filiale)

Action InvStandSuche

Suche innerhalb des oben gewählten Standorts

Action InvStandAll

Alle Buchungen anzeigen

Action Standorte

Spaltendesigner (Standortinventur)

Action Abwertung

Abwertung aus Inventur übernehmen (zur Datenübergabe an die Zentrale)

Aktionen für die Preisliste

Action SpaltenDesigner

Spaltendesigner

Action Artikelanlegen

der Artikel wird in die Stamdaten übertragen

Action Close

Fenster schließen

Action ZeigeArtikel

Sucht den Artikel im echten Artikelstamm

Action Volltextsuche

Action_Suche

Action_Preisvergleich
Preisvergleich

Aktionen für die Rabattmatrix

Action_Suchen
Warengruppen suchen

Action_Spalteneditor
Spalteneditor

Action_Drucken
Drucken mit Standardeinstellungen

Action_DruckenCFG
Druckerassistent Einstellungen

Action_Helfer
Der Helfer erklärt das Fenster

DataSetDelete1
Position löschen

DataSetEdit1
Position bearbeiten

DataSetFirst1
Auf den ersten Datensatz springen

DataSetLast1
Auf den letzten Datensatz springen

DataSetNext1
Auf den nächsten Datensatz springen

DataSetPost1
Datensatz speichern

DataSetPrior1
Auf vorherigen Datensatz springen

DataSetRefresh
Datensatz aktualisieren

Act_VorgangDruck

AFS-Maskeneditor
Vorgang drucken

Act_VorgangsPreview
Druckvorschau ansehen

Act_VorgangFormular
Druck-Formular auswählen

Act_VorgangEdit
Vorgang bearbeiten (DTP-Formulareditor)

Act_Bezahlung
Bezahlung eingeben

Act_VorgangAbschliessen
Vorgang abschließen

Act_StatusBearbeiten
Fenster Vorgangstatus (Status bearbeiten)

Act_VorgangFortfuehren
Vorgang in einen anderen weiterführen

Act_VorStorno
Vorgang stornieren

Act_Druckerassistent
Ruft den Druckerassistenten auf

Act_Drucken
Drucken mit den Standarteinstellungen des Druckerassistent

Act_Serienbrief
Textverarbeitung

Act_RechLiefAnschrift
Vorgangsanschrift (Lieferanten) wählen

Act_FORVorbereitung
FOR-Vorbereitung

Action_Kopien
Anzahl der Ausdrucke festlegen (Vorgabenwert)

Action_ArtikelGrid
Spalteneditor (Artikel)

Act_PosSchieben
Position verschieben

Action_Up

Position eine Stufe nach oben schieben

Action_Down

Position eine Stufe nach unten schieben

Action_Interessant

Interessentenanschrift auswählen

Action_PosInsert

neue Position anlegen

Action1_PosNum

nummeriert die Position

Action_AllPrint

alle ungedruckten Vorgänge der gleichen Art drucken

Action_History

Öffnet das Fenster „History“

Act_Lagerbuchen

Wareneingang und Warenausgang buchen

Action_Kopieren

Kopiert den gesamten Vorgang

Action_Grunddaten

Grunddaten anzeigen

Action_Posten

Posten anzeigen

Action_Verbuchen

Vorgang verbuchen

Action_Mail

per eMail senden

Action_Fax

per Fax senden

Action_PrintFrage

Drucken mit Auswahl

Action_PrintCFG

Drucker-Assistent

Act_PreiseOFF

Preise ausschalten

AFS-Maskeneditor

Action ZeigeAdresse

Zeigt die Adresse an

Action ZeigeArtikel

Artikel anzeigen

Action ArtikelSQLSuche

Artikel per SQL suchen

Action VorMail Formular

E-Mail Formular auswählen

Action VorFax Formular

Fax-Formular auswählen

Action RepKunden

Reparaturvorgang: Vorgangsanschrift (Kunden) wählen

Action RepPrint

Reparaturvorgang: Drucken- Reparatursonderdrucke

Action RepInter

Vorgangsanschrift (Interessenten) wählen

Action ArtikelWandeln

in Artikel wandeln

Action Handyvertrag

Vertragszuweisung entfernen

Action VertragAnlegen

Neuen Vertrag anlegen

Action VertragZeigen

Verträge anzeigen

Action VertragKundenZeigen

Verträge des Kunden anzeigen

Action HandyPreisberechnen

Preise neu berechnen

Action AbAnschrift

Abweichene Lieferanschrift

Action Dimanzeige

Dimensionsanzeige

Action EKAktual

Sollten sich die EK's seit der Vorgangserstellung verändert haben, werden
Seite 62

diese im Vorgang aktualisiert und der Rohertrag neu ermittelt)

Action_VKAktual

VK's neu aktualisieren. Sollten sich die EK's seit der Vorgangserstellung verändert haben, werden diese im Vorgang aktualisiert

Act_TabDes_Doku

Öffnet den Spaltendesigner

Action_Script_Edit

Druckerscript bearbeiten

Action_Preisliste

Preisliste öffnen

Act_Umsatz

Grafische Umsatzstatistik anzeigen

Alle nachfolgenden Actions werden in alphabetischer Reihenfolge gelistet und beschrieben. Die vorherigen Actions finden mehr Beachtung, da sie häufiger genutzt werden.

Aktionen für den Abbuchungsauftrag

Action_Kunden

Kundenkonto wählen

Action_Lieferant

Lieferantenkonto wählen

Action_Interessenten

Interessentenkonto wählen

Action_Blz1

Bank suchen

Action_OP

Vorgang auswählen (OP)

Action_Mandantenkonto

Mandantenkonto wählen

Action_Drucken

Beleg drucken und verbuchen

Action_DrEinstellung

Einstellungen der Ausdrucke

Action_Helfer

AFS-Maskeneditor
der Helfer erklärt das Fenster

Action_Drucken2
Infodruck (für Listen usw)

Action_DrEinstellung2
Einstellungen der Ausdrucke

Action_DTACreate
DTA-Diskette schreiben

Action_GridEdit
Spalteneditor

Action_Select
Selektion durchführen

Action_ZeigeAdresse
Zeigt die Adresse des Belegs an

Action_ZeigeVor
Zeigt den Vorgang des Belegs an

Action_DTACreate2
DTA-Datei schreiben für Bankingprogramm

Action_ExcelExport
Export zu Excel

Action_WordExport
Export zu Word

Aktionen für Angebote

Act_Artikelsuchen
Artikel suchen

Act_PosSpei_Neu
Position speichern und eine neue Position anlegen

DataSetPost1
Position speichern

Act_PosKopieren
angewählte Position kopieren

Act_Vor_Post
Vorgang speichern

Act_Vor_PostAndNeu

Vorgang speichern und nächsten Vorgang anlegen

Act_PosGridDesign

Postentabelle verändern

Act_DTP_Print

ausdrucken des Vorgangs

Act_ReportEdit

Reportgenerator

Act_ReportPrint

Drucken

Act_ReportPreview

Druckvorschau

Act_ReportVorbereitung

Act_AllgemeinPrintVorbereitung

Act_DTPVorbereitung

Act_DTPEdit

Act_DTP_Preview

Druckvorschau

Act_ArtSN_Select

verfügbare Seriennummern übernehmen

Act_PosSerien

Seriennummernverwaltung

Act_RechAnschrift

Vorgangsanschrift (Kunden) wählen

Act_LiefAnschrift

abweichende Lieferanschrift

Act_VorTermin

Termin für diesen Vorgang anlegen

Act_VorTerminList

Terminliste für diesen Vorgang ansehen

DataSetCancel1

letzte Änderung an der Position rückgängig machen

AFS-Maskeneditor

DataSetDelete1

lösche die angewählte Position aus dem Vorgang

DataSetEdit1

Bearbeiten

DataSetFirst1

Erster

DataSetLast1

Letzter

DataSetNext1

Nächster

DataSetPrior1

Vorheriger

DataSetRefresh1

Aktualisieren

Act_VorgangsDruck

ausdrucken des Vorgangs

Act_VorgangsPreView

Druckvorschau für diesen Vorgang

Act_VorgangFormular

wählen Sie das Formular zum Ausdrucken aus

Act_VorgangEdit

verändern des Ausdrucks

Act_Bezahlung

Zahlung eingeben

Act_VorgangAbschliessen

Vorgang abschließen

Act_StatusBearbeiten

Status bearbeiten

Act_Grafik

grafische Auswertung

Act_VorgangFortfuehren

Vorgangfortführen

Act_VorStorno

Vorgang stornieren

Seite 66

Act DruckerAssistent

Druckassistent

Act Drucken

Druckt mit den aktuellen Einstellungen

Act Serienbrief

Textverarbeitung mit Daten aus dem Vorgang

Act RechLiefAnschrift

Vorgangsanschrift (Lieferanten) wählen

Act FORVorbereitung

Action Kopien

Anzahl der Ausdrücke festlegen

Action ArtikelGrid

Spalteneditor

Act PosSchieben

Position verschieben an eine andere Position

Action Up

Position eine Stufe nach oben schieben

Action Down

Position eine Stufe nach unten schieben

Action Interessent

Interessentenanschrift auswählen: z.B. für Angebot

Action PosInsert

eine neue Position anlegen

Action1 PosNum

nummeriert die Positionen neu

Action AllPrint

alle ungedruckten Vorgänge der gleichen Art drucken

Action History

zeigt die History des Artikel mit dieser Anschrift

Act Lagerbuchen

Wareneingang- und Warenausgang buchen

Action Kopieren

kopiert den gesamten Vorgang

AFS-Maskeneditor
Action Grunddaten

Action Posten

Action Verbuchen
Vorgang verbuchen

Action Mail
per E-Mail senden, statt drucken

Action Fax
per Fax senden

Action PrintFrage
Drucken mit Auswahl

Action PrintCFG
Einstellungen

Action PreiseOff
Preise aus

Action ZeigeAdresse
Zeigt die Adresse des Vorgangs an

Action ZeigeArtikel
Artikel anzeigen

Action ArtikelSQLSuche
Artikelsuche per SQL (Wildcards)

Action VorMail Formular
wählen Sie das Formular zum Mailen aus

Action VorFax Formular
wählen Sie das Formular zum Faxen aus

Action RepKunden
Vorgangsanschrift (Kunden) wählen

Action RepLief
Vorgangsanschrift (Lieferanten) wählen

Action RepPrint
Drucken mit Auswahl

Action RepInter
Vorgangsanschrift (Interessenten) wählen

Action ArtikelWandeln
in Artikel wandeln (anlegen)
Seite 68

Action HandyVertrag

Action VertragAnlegen

Vertrag in der Vertragsverwaltung anlegen

Action VertragZeigen

Verträge, die aus diesem Vorgang angelegt wurden anzeigen

Action VertragKundenZeigen

Verträge, des Kunden anzeigen

Action HandyPreisberechnen

Preis für Gerät + Vertrag neu berechnen

Action AbAnschrift

Abweichende Lieferanschrift

Action DimAnzeige

Dimensionsanzeige

Action EKAktual

EK's neu aktualisieren (sollten sich die EK's seit der Vorgangserstellung verändert haben, werden diese im Vorgang aktualisiert und der Rohertrag neu ermittelt)

Action VKAktual

VK's neu aktualisieren (sollten sich die VK's seit der Vorgangserstellung verändert haben, werden diese im Vorgang aktualisiert und der Rohertrag neu ermittelt)

Action VKGruppe

Wechseln Sie für diesen Vorgang die VK-Gruppe. Bei der automatischen Preisfindung wird eine andere Preisgruppe benutzt. Diese Einstellung gibt nur für diese Eingabe und für den geöffneten Vorgang!

Action TabDes Doku

Spaltendesigner

Action Script Edit

Druckerscript bearbeiten

Action Preisliste

Action Ansprechpartner Speichern

Speichert den Ansprechpartner

Action Adresse Speichern

Speichert die Adresse

AFS-Maskeneditor

Action Artikel Speichern

Speichert den Artikel

Action Warengruppe Speichern

Speichert die Warengruppe

Action Drucke EigenesFormular

Action BuchenOhneDruck

manuelle Lagerbuchung durchführen (ohne Ausdruck)

Action Vertrag Speichern

Speichert den Vertrag

Action KundeBesteller

Besteller (Kunden) wählen

Action Gutschrift Verechnen

Gutschrift verrechnen

Action InteressentBesteller

Besteller (Interessent) wählen

Action PosSort

Sortieren der Positionen

Action Vorgangsnummer

Action PostenInsert

Posten einfügen

Action ArtikelVar1

Spaltendesigner

Action ArtikelVar2

Spaltendesigner

Action Vorverfolgung

Vorgangsverfolgung

Action RueckstandAnz

Rückstand anzeigen - Zeigt die Rückstandsverwaltung mit den Artikeln an, die in diesem Vorgang vorhanden sind!

Action EDIRechnungsliste

EDI-Rechnungsliste

Action Bestellung Erzeugen

Bestellung für diese Position erzeugen

Action Bestellung alleErzeugen

komplette Bestellung für diesen Vorgang erzeugen

Action Geraet Save

diesem Vorgang zuweisen

Action Geraet Del

Zuweisung aufheben

Action Geraet View

zeige Gerät an

Action Geraet Aus

Auswählen

Action Geraet Search

Geräte suche

Action Geraet Adresse

nur Geräte des Kunden anzeigen

Action Geraet Kunde

Gerät dieser Adresse zuweisen

Action Geraet Spalten

Spaltendesign

Action Filiale Leeren

Filialezuweisung entfernen

Action PrintPDF

als PDF ausgeben

Action PrintOn

zeigt die Druckzusatzfunktionen

Action PrintOff

blendet die Druckzusatzfunktionen aus

Action Mahnung Erzeugen

Vorgang mahnen

Action Mahnung Nochmals

letzte Mahnung nochmals ausgeben

Action Vorgangsverfolgung

AFS-Maskeneditor
Sendungsverfolgung öffnen

Action_Pos_Geraet_Del
Gerätezuweisung entfernen

Action_POS_Geraet_Search
Gerät suchen

Action_ArtikelNummernSuche
Artikel suchen nach Nummern

Action_ArtikelBezeichnungSuche
Artikel suchen nach Bezeichnung

Action_GeraetAnlegen
die Position als Gerät in der Geräteverwaltung anlegen

Action_GeraetAnzeigen
zeige Gerät an

Action_GeraetAnlegenVor
neues Gerät in der Geräteverwaltung anlegen

Action_GeraetAnlegenRep
neues Gerät in der Geräteverwaltung anlegen

Action_VE_Set
VE-Übernehmen

Action_VE_Cancel
Abbruch

Action_magiceddy
magic EDI-Ausgabe

Action_Lieferanschrift_Frei
freie Lieferanschrift

Action_Lieferanschrift_DEF
definierte Lieferanschrift

Action_Produktion_IST
IST-Werte Erfassung

Action_Abschlag_Druck
Abschlagsrechnung drucken

Action_Abschlag_Save

Speichern

Action_Abschlag_Eingabe
Abschlag erfassen

Action_Abschlag_Formular
Druckeinstellungen

Action_Abschlag_Del
Abschlag entfernen

Action_Abschlag_Storno
Abschlag stornieren

Action_StartTime
Startzeit = jetzt

Action_EndTime
Endzeit = jetzt

Action_Art_Such_Lief
nur lieferbare Artikel anzeigen

Action_Art_Such_Be
nur Artikel mit Bestand anzeigen

Action_Kalk_ZwischenSumme
Kalkulation der Zwischensumme - Zwischensumme auf die Eingabe umrechnen

Action_Kalk_ZwischenRoh
Kalkulation des Rohertrags der Zwischensumme - Rohertrag der Zwischensumme auf die Eingabe umrechnen

Action_Kalk
Kalkulation des gesamten Vorgangs

Action_Kalk_EndBSumme
Kalkulation der Bruttoendsumme - Endsumme auf die Eingabe umrechnen

Action_Kalk_EndRoh
Kalkulation des Rohertrags - Rohertrag umrechnen

Action_Kalk_EndNSumme
Kalkulation der Nettoendsumme - Endsumme auf die Eingabe umrechnen

Action_Kalk_EndRohPro
Kalkulation des Rohertrags in % - Rohertrag umrechnen

AFS-Maskeneditor

Action Postenverdichten

Postenverdichten - gleiche Posten werden zusammengefasst

Aktionen der Arbeitszeitverwaltung

Action Print

Druck

Action Print CFG

Einstellungen der Ausdrucke

Action Neu

neue manuelle Buchung

Action Bearbeiten

Buchung bearbeiten

Action Delete

Buchung löschen

Action Save

Buchung speichern

Aktionen der Artikeltarife

Action Close

Fenster schliessen

Action SpaltenEdit

Spalteneditor

Action Artikel

Artikel suchen

Action Tarif

Tarif wählen

Action ArtikelZuAus

Artikel suchen

Aktionen der Artikelverkaufsauswertung

Action Close

Abfrage schließen

Action OK

Auswertung starten

Action_Grid
Spalteneditor

Action_BonDruck
Ausgabe auf dem Kasse-Bondrucker

Action_Helfer
der Helfer erklärt das Fenster

Action_DruckenAus
Drucken mit Auswahl

Action_PrintCFG
Einstellungen

Action_BONArWa
Artikel nach Warengruppen (auf den Bondrucker)

Action_ExcelExport
Export zu Excel

Action_WordExport
Export zu Word

Action_Filter

Action_Art_Search

Aktionen der Auftragsbestätigung

Act_Artikelsuchen
Artikel suchen

Act_PosSpei_Neu
Position speichern und eine neue Position anlegen

DataSetPost1
Position speichern

Act_PosKopieren
angewählte Position kopieren

Act_Vor_Post
Vorgang speichern

Act_Vor_PostAndNeu
Vorgang speichern und nächsten Vorgang anlegen

AFS-Maskeneditor

Act_PosGridDesign

Postentabelle verändern

Act_DTP_Print

ausdrucken des Vorgangs

Act_ReportEdit

Reportgenerator

Act_ReportPrint

Drucken

Act_ReportPreview

Druckvorschau

Act_ReportVorbereitung

Act_AllgemeinPrintVorbereitung

Act_DTPVorbereitung

Act_DTPEdit

Act_DTP_Preview

Druckvorschau

Act_ArtSN_Select

verfügbare Seriennummern übernehmen

Act_PosSerien

Seriennummernverwaltung

Act_RechAnschrift

Vorgangsanschrift (Kunden) wählen

Act_LiefAnschrift

abweichende Lieferanschrift

Act_VorTermin

Termin für diesen Vorgang anlegen

Act_VorTerminList

Terminliste für diesen Vorgang ansehen

DataSetCancel1

letzte Änderung an der Position rückgängig machen

DataSetDelete1

lösche die angewählte Position aus dem Vorgang

DataSetEdit1

Bearbeiten

DataSetFirst1

Erster

DataSetLast1

Letzter

DataSetNext1

Nächster

DataSetPrior1

Vorheriger

DataSetRefresh1

Aktualisieren

Act_VorgangsDruck

ausdrucken des Vorgangs

Act_VorgangsPreView

Druckvorschau für diesen Vorgang

Act_VorgangFormular

wählen Sie das Formular zum Ausdrucken aus

Act_VorgangEdit

verändern des Ausdrucks

Act_Bezahlung

Zahlung eingeben

Act_VorgangAbschliessen

Vorgang abschließen

Act_StatusBearbeiten

Status bearbeiten

Act_Grafik

grafische Auswertung

Act_VorgangFortfuehren

Vorgangfortführen

Act_VorStorno

Vorgang stornieren

AFS-Maskeneditor

Act DruckerAssistent
Druckassistent

Act Drucken

Act Serienbrief
Textverarbeitung mit Daten aus dem Vorgang

Act RechLiefAnschrift
Vorgangsanschrift (Lieferanten) wählen

Act FORVorbereitung

Action Kopien
Anzahl der Ausdrucke festlegen

Action ArtikelGrid
Spalteneditor

Act PosSchieben
Position verschieben an eine andere Position

Action Up
Position eine Stufe nach oben schieben

Action Down
Position eine Stufe nach unten schieben

Action PosInsert
eine neue Position anlegen

Action1 PosNum
nummierierte Positionen neu

Action AllPrint
alle ungedruckten Vorgänge der gleichen Art drucken

Action History
zeigt die History des Artikel mit dieser Anschrift

Act Lagerbuchen
Wareneingang- und Warenausgang buchen

Action Kopieren
kopiert den gesamten Vorgang

Action Grunddaten

Action Posten

Action Verbuchen

Vorgang verbuchen

Action Mail

per E-Mail senden, statt drucken

Action Fax

per Fax senden

Action PrintFrage

Drucken mit Auswahl

Action PrintCFG

Einstellungen

Action PreiseOff

Preise aus

Action ZeigeAdresse

Zeigt die Adresse des Vorgangs an

Action ZeigeArtikel

Artikel anzeigen

Action ArtikelSQLSuche

Artikelsuche per SQL (Wildcards)

Action VorMail Formular

wählen Sie das Formular zum Mailen aus

Action VorFax Formular

wählen Sie das Formular zum Faxen aus

Action RepKunden

Vorgangsanschrift (Kunden) wählen

Action RepLief

Vorgangsanschrift (Lieferanten) wählen

Action RepPrint

Drucken mit Auswahl

Action RepInter

Vorgangsanschrift (Interessenten) wählen

Action ArtikelWandeln

in Artikel wandeln (anlegen)

AFS-Maskeneditor

Action_HandyVertrag

Action_VertragAnlegen

Vertrag in der Vertragsverwaltung anlegen

Action_VertragZeigen

Verträge, die aus diesem Vorgang angelegt wurden anzeigen

Action_VertragKundenZeigen

Verträge, des Kunden anzeigen

Action_HandyPreisberechnen

Preis für Gerät + Vertrag neu berechnen

Action_AbAnschrift

Abweichende Lieferanschrift

Action_DimAnzeige

Dimensionsanzeige

Action_EKAktual

EK's neu aktualisieren (sollten sich die EK's seit der Vorgangserstellung verändert haben, werden diese im Vorgang aktualisiert und der Rohertrag neu ermittelt)

Action_VKAktual

VK's neu aktualisieren (sollten sich die VK's seit der Vorgangserstellung verändert haben, werden diese im Vorgang aktualisiert und der Rohertrag neu ermittelt)

Action_VKGruppe

Wechseln Sie für diesen Vorgang die VK-Gruppe. Bei der automatischen Preisfindung wird eine andere Preisgruppe benutzt. Diese Einstellung gibt nur für diese Eingabe und für den geöffneten Vorgang!

Action_TabDes_Doku

Spaltendesigner

Aktionen der Barkasse

Action_Neu

neuer Kassiervorgang

Action_Kunde

einen Kunden auswählen

Action_Zahlen

Kunde will Zahlen

Seite 80

Action Vorgangsuchen

sucht einen alten Kassenvorgang

ActionClose

Kasse beenden

Action Schublade

Schublade öffnen

Action Lieferant

Action Parken

legt diesen Vorgang zur Seite, dieser kann später wieder abgerufen werden

Action Holen

holt einen geparkten Vorgang zurück

Action AltVorgang

alten Kassenvorgang nochmals bearbeiten

Action ecCashStorno

Action Gutschein

Gutscheinverwaltung

Action Gutschein New

Gutschein erstellen und kassieren

Action Zeiterfassung

Arbeitszeiterfassung

Action Gutschein Ein

Gutschein einlösen

Action Kasseneinstellungen

Kasseneinstellungen (Formulare und Zahlungsbedingungen)

Act Artikelsuchen

Artikel suchen

Action KasseneinstellungenTeil2

Kasseneinstellungen (Bedienung)

Act PosKopieren

angewählte Position kopieren

Act AllgemeinPrintVorbereitung

AFS-Maskeneditor

Act_ArtSN_Select

verfügbare Seriennummern übernehmen

Act_Zusatz

Zusatzdaten eingeben (Seriennummern, usw.)

DataSetCancel1

letzte Änderung an der Position rückgängig machen

DataSetDelete1

lösche die angewählte Position aus dem Vorgang

DataSetFirst1

Erster

DataSetInsert1

eine neue Position anlegen

DataSetLast1

Letzter

DataSetNext1

Nächster

DataSetPrior1

Vorheriger

DataSetRefresh1

Aktualisieren

Action_Hardware

Kasse-Hardware einstellen

Act_SNEingeben

Seriennummern/Chargennummern eingeben

Action_Edit

Position bearbeiten

Action_PosDelete

Position löschen

Action_ZBericht

Kassenabschluß (Z-Bericht)

Action_XBericht

Kassenprüfung (X-Bericht)

Action_PostenDes

Seite 82

Postentabelle verändern

Action ArtikelSQLSuche
Artikelsuche per SQL (Wildcards)

Action Artikel
Artikel bearbeiten

Action Kunden
Kunden bearbeiten

Action VertragAnlegen
Vertrag in der Vertragsverwaltung anlegen

Action VertragZeigen
Verträge, die aus diesem Vorgang angelegt wurden anzeigen

Action VertragKundenZeigen
Verträge des Kunden anzeigen

Action HandyPreisberechnen
Preis für Gerät + Vertrag neu berechnen

Action HandyVertrag
Vertragszuweisung entfernen

Action RabattEingeben
Gesamtrabatt eingeben

Action Lieferanschrift
Abweichende Anschrift eingeben (Lieferanschrift)

Action PreisGlaetten
Preis glätten (Festpreis vorgeben)

Action ZahlandererVorgang
andere Vorgang bezahlen

Action RechnungenZahlen
Rechnungen bezahlen

Action KassenvorgangZahlen
alten Kassenvorgang bezahlen

Action RepVorgangZahlen
Reparaturvorgang bezahlen

Action LiefVorgangZahlen
Lieferschein bezahlen

AFS-Maskeneditor

Action Vk aktualisieren

Action Entnahme
Kassen-Entnahme

Action Einlage
Kassen-Einlage

Aktionen des Bestellwesens

Action Bestell
die gewählte Bestellung ausführen

Action BestellKomplett
alle Bestellungen ausführen

Action Tel
Telefonnummer anwählen

Action ArtBestell
aktuellen Artikel in den Warenkorb

Action ArtGesamtBestell
alle Artikel in den Bestellwarenkorb

Action ArtLief Auswahl
Lieferanten auswählen

Action Lieferant
Lieferant auswählen dem ausgewählten Artikel zuweisen

Action Bestellwesen
Bestellwesen voll automatisch ausführen

Action Spalten
Spalteneditor

Action Lieferant Delete
Lieferant löschen

Action Print
Drucken der Liste

Action Einstell
Einstellungen der Ausdrucke

Action Helfer
der Helfer erklärt das Fenster
Seite 84

Action Artikelzeigen
alle Artikel wieder anzeigen

Action LieferantenZeigen
Lieferanten anzeigen

Action SQL_Open
Artikelliste öffnen

Action AlleArtikel
zeigt alle Artikel an

Action ArtikelLieferant
Lieferant auswählen

Aktionen des DTA-Bankings

Action Kunden
Kundenkonto wählen

Action Lieferant
Lieferantenkonto wählen

Action Interessenten
Interessentenkonto wählen

Action Blz1
Bank suchen

Action OP
Vorgang auswählen (OP)

Action Mandantenkonto
Mandantenkonto wählen

Action Drucken
Beleg drucken und verbuchen

Action DrEinstellung
Einstellungen der Ausdrucke

Action Helfer
der Helfer erklärt das Fenster

Action Drucken2
Infodruck (für Listen usw)

Action DrEinstellung2
Einstellungen der Ausdrucke

AFS-Maskeneditor

Action_DTACreate

DTA-Diskette schreiben

Action_GridEdit

Spalteneditor

Action_Select

Selektion durchführen

Action_ZeigeAdresse

Zeigt die Adresse des Belegs an

Action_ZeigeVor

Zeigt den Vorgang des Belegs an

Action_DTACreate2

DTA-Datei schreiben für Bankingprogramm

Action_ExcelExport

Export zu Excel

Action_WordExport

Export zu Word

Aktionen für externe Preislisten

Action_SpaltenDesigner

Spaltendesigner

Action_Artikelanlegen

der Artikel wird in die Stammdaten übertragen

Action_Close

Fenster schließen

Action_ZeigeArtikel

Sucht den Artikel im echten Artikelstamm

Action_Volltextsuche

Action_Suche

Action_Preisvergleich

Preisvergleich

Aktionen der Geräteverwaltung

Action Close

Fertig

Action_SPE_Geraet

Spalteneditor

Action_SPE_History

Spalteneditor

Action_Print

Drucken der Liste

Action_PrintCFG

Einstellungen der Ausdrücke

Action_Kunde_Waehlen

Kunde, der das Gerät besitzt

Action_Artikel_Waehlen

Artikel zuweisen

Action_Lieferant_Waehlen

Lieferant, der das Gerät geliefert hat zuweisen

Action_Kundenauswahl

Geräte eines Kunden anzeigen

Action_Lieferantenauswahl

Geräte eines Lieferanten anzeigen

Action_Gesamtauswahl

Gesamtauswahl anzeigen

Action_Bild1_Laden

Bild laden

Action_Bild2_Laden

Bild laden

Action_Bild3_Laden

Bild laden

Action_Bild4_Laden

Bild laden

Action_Bild5_Laden

Bild laden

Action_Bild1_Anzeigen

Bild anzeigen

AFS-Maskeneditor

Action_Bild2_Anzeigen

Bild anzeigen

Action_Bild3_Anzeigen

Bild anzeigen

Action_Bild4_Anzeigen

Action_Bild5_Anzeigen

Bild anzeigen

Action_Maskeneditor

Maskeneditor

Action_Search

Suchen

Action_ExcelExport

Export zu Excel

Action_WordExport

Export zu Word

Action_SN_Brief

Serienbriefassistent (Kontaktmanager)

Action_SucheAus

Suche beenden: zeigt wieder alle Daten an

Action_Artikel_bearbeiten

Artikel bearbeiten

Action_ZeigeAdresse

Adresse anzeigen

Action_H_Artikel_bearbeiten

Artikel bearbeiten

Action_H_ZeigeAdresse

Adresse anzeigen

Action_H_Vorgangbearbeiten

Action_Termin_New

neuen Termin anlegen

Aktionen für Gutschriften

Seite 88

Act Artikelsuchen

Position speichern und eine neue Position anlegen

Act PosSpei_Neu

Position speichern

DataSetPost1

angewählte Position kopieren

Act PosKopieren

Vorgang speichern

Act Vor Post

Vorgang speichern und nächsten Vorgang anlegen

Act Vor PostAndNeu

Postentabelle verändern

Act PosGridDesign

Postentabelle verändern

Act DTP Print

ausdrucken des Vorgangs

Act ReportEdit

Reportgenerator

Act ReportPrint

Drucken

Act ReportPreview

Druckvorschau

Act ReportVorbereitung

Act AllgemeinPrintVorbereitung

Act DTPVorbereitung

Act DTPEdit

Act DTP Preview

Druckvorschau

Act ArtSN Select

verfügbare Seriennummern übernehmen

AFS-Maskeneditor

Act_PosSerien

Seriennummernverwaltung

Act_RechAnschrift

Vorgangsanschrift (Kunden) wählen

Act_LiefAnschrift

abweichende Lieferanschrift

Act_VorTermin

Termin für diesen Vorgang anlegen

Act_VorTerminList

Terminliste für diesen Vorgang ansehen

DataSetCancel1

letzte Änderung an der Position rückgängig machen

DataSetDelete1

lösche die angewählte Position aus dem Vorgang

DataSetEdit1

Bearbeiten

DataSetFirst1

Erster

DataSetLast1

Letzter

DataSetNext1

Nächster

DataSetPrior1

Vorheriger

DataSetRefresh1

Aktualisieren

Act_VorgangsDruck

ausdrucken des Vorgangs

Act_VorgangsPreView

Druckvorschau für diesen Vorgang

Act_VorgangFormular

wählen Sie das Formular zum Ausdrucken aus

Act_VorgangEdit

verändern des Ausdrucks

Seite 90

Act Bezahlung

Zahlung eingeben

Act VorgangAbschliessen

Vorgang abschließen

Act StatusBearbeiten

Status bearbeiten

Act Grafik

grafische Auswertung

Act VorgangFortfuehren

Vorgangfortführen

Act VorStorno

Vorgang stornieren

Act DruckerAssistent

Druckassistent

Act Drucken

Act Serienbrief

Textverarbeitung mit Daten aus dem Vorgang

Act RechLiefAnschrift

Vorgangsanschrift (Lieferanten) wählen

Act FORVorbereitung

Action Kopien

Anzahl der Ausdrücke festlegen

Action ArtikelGrid

Spalteneditor

Act PosSchieben

Position verschieben an eine andere Position

Action Up

Position eine Stufe nach oben schieben

Action Down

Position eine Stufe nach unten schieben

Action PosInsert

eine neue Position anlegen

AFS-Maskeneditor

Action1_PosNum

nummiiert die Positionen neu

Action_AllPrint

alle ungedruckten Vorgänge der gleichen Art drucken

Action_History

zeigt die History des Artikel mit dieser Anschrift

Act_Lagerbuchen

Wareneingang- und Warenausgang buchen

Action_Kopieren

kopiert den gesamten Vorgang

Action_Grunddaten

Action_Posten

Action_Verbuchen

Vorgang verbuchen

Action_Mail

per E-Mail senden, statt drucken

Action_Fax

per Fax senden

Action_PrintFrage

Drucken mit Auswahl

Action_PrintCFG

Einstellungen

Action_PreiseOff

Preise aus

Action_ZeigeAdresse

Zeigt die Adresse des Vorgangs an

Action_ZeigeArtikel

Artikel anzeigen

Action_ArtikelSQLSuche

Artikelsuche per SQL (Wildcards)

Action_VorMail_Formular

wählen Sie das Formular zum Mailen aus

Action_VorFax_Formular

wählen Sie das Formular zum Faxen aus

Action_RepKunden

Vorgangsanschrift (Kunden) wählen

Action_RepLief

Vorgangsanschrift (Lieferanten) wählen

Action_RepPrint

Drucken mit Auswahl

Action_RepInter

Vorgangsanschrift (Interessenten) wählen

Action_ArtikelWandeln

in Artikel wandeln (anlegen)

Action_HandyVertrag

Action_VertragAnlegen

Vertrag in der Vertragsverwaltung anlegen

Action_VertragZeigen

Verträge, die aus diesem Vorgang angelegt wurden anzeigen

Action_VertragKundenZeigen

Verträge, des Kunden anzeigen

Action_HandyPreisberechnen

Preis für Gerät + Vertrag neu berechnen

Action_AbAnschrift

Abweichende Lieferanschrift

Action_DimAnzeige

Dimensionsanzeige

Action_EKAktual

EK's neu aktualisieren (sollten sich die EK's seit der Vorgangserstellung verändert haben, werden diese im Vorgang aktualisiert und der Rohertrag neu ermittelt)

Action_VKAktual

VK's neu aktualisieren (sollten sich die VK's seit der Vorgangserstellung verändert haben, werden diese im Vorgang aktualisiert und der Rohertrag neu ermittelt)

Action_VKGruppe

AFS-Maskeneditor

Wechseln Sie für diesen Vorgang die VK-Gruppe. Bei der automatischen Preisfindung wird eine andere Preisgruppe benutzt. Diese Einstellung gibt nur für diese Eingabe und für den geöffneten Vorgang!

Action TabDes Doku

Spaltendesigner

Action Script Edit

Druckerscript bearbeiten

Action Preisliste

Action Ansprechpartner Speichern

Speichert den Ansprechpartner

Action Adresse Speichern

Speichert die Adresse

Action Artikel Speichern

Speichert den Artikel

Action Warengruppe Speichern

Speichert die Warengruppe

Action Drucke EigenesFormular

Action BuchenOhneDruck

manuelle Lagerbuchung durchführen (ohne Ausdruck)

Action Vertrag Speichern

Speichert den Vertrag

Aktionen für Kassenabschlussberichte

Action PrintBon

Druck auf Bondrucker (Kassendrucker)

Action Print

Druck auf Standarddrucker (Windowsdrucker)

Aktionen für das Kassenbuch

Action Spalten1

Spalteneditor

Action Spalten2

Spalteneditor

Action K Ein Aus
grafische Auswertung der Einnahmen und Ausgaben

Action K Aus Ko
Ausgaben nach Kontenaufteilung

Action K Ein Ko
Einnahmen nach Kontenaufteilung

Action CashCheck
Kasse zählen

Action CFG OK
Einstellungen übernehmen

Act Print
Drucken der Liste

Act PreView
Einstellungen der Ausdrucke

Act Formularedit
der Helfer erklärt das Fenster

Action ExcelExport
Export zu Excel

Action WordExport
Export zu Word

Action Schublade
Schublade öffnen

Action Kassenhardware

Aktionen der Lagerbuchung

Act LagerNeu
neue lagerbuchung einfügen

Act LagerCancel
Rückgängig

Act LagerSave

AFS-Maskeneditor
Lagereintrag speichern

Act_Print
Drucken der Lagerbewegungen

Act_PreView
Einstellungen der Ausdrücke

Act_GridEdit
Spalteneditor

Act_Close
Lagerbuchung beenden

Act_ArtikelSuchen
Artikel suchen

Act_SNGridEdit
Spalteneditor

Action_SN_Verwaltung
Seriennummernverwaltung

Action_Lieferant
Lieferant suchen

Action_Artikelzeigen
Artikel zeigen

Action_LieferantenZeigen
Adresse zeigen

Action_Vor_Bearbeiten
Vorgang anzeigen und bearbeiten

Action_Helfer
der Helfer erklärt das Fenster

Action_ExcelExport
Export zu Excel

Action_WordExport
Export zu Word

Action_Bestandzuweise
Wenn das Mehrlagermodul nachträglich erworben wurde den Mehrbestand dem Hauptlager zuweisen oder Bestände, die importiert wurden, jedoch ohne Zuweisung sind, dem Hauptlager zuweisen!

Action_Bestandentfernen

Bestände, die keinem Lager zugewiesen sind entfernen! Artikelbestand =
Summe der Lagerbestände

Action ArtikelVarSpalten
Spaltendesigner

Aktionen für Lastschriften

Action Kunden
Kundenkonto wählen

Action Lieferant
Lieferantenkonto wählen

Action Interessenten
Interessentenkonto wählen

Action Blz1
Bank suchen

Action OP
Vorgang auswählen (OP)

Action Mandantenkonto
Vorgang auswählen (OP)

Action Drucken
Beleg drucken und verbuchen

Action DrEinstellung
Einstellungen der Ausdrucke

Action Helfer
der Helfer erklärt das Fenster

Action Drucken2
Infodruck (für Listen usw)

Action DrEinstellung2
Einstellungen der Ausdrucke

Action DTACreate
DTA-Diskette schreiben

Action GridEdit
Spalteneditor

Action Select
Selektion durchführen

AFS-Maskeneditor

Action ZeigeAdresse

Zeigt die Adresse des Belegs an

Action ZeigeVor

Zeigt den Vorgang des Belegs an

Action DTACreate2

DTA-Datei schreiben für Bankingprogramm

Action ExcelExport

Export zu Excel

Action WordExport

Export zu Word

Aktionen für Lieferscheine

Act Artikelsuchen

Artikel suchen

Act PosSpei Neu

Position speichern und eine neue Position anlegen

DataSetPost1

Position speichern

Act PosKopieren

angewählte Position kopieren

Act Vor Post

Vorgang speichern

Act Vor PostAndNeu

Vorgang speichern und nächsten Vorgang anlegen

Act PosGridDesign

Postentabelle verändern

Act DTP Print

ausdrucken des Vorgangs

Act ReportEdit

Reportgenerator

Act ReportPrint

Drucken

Act_ReportPreview
Druckvorschau

Act_ReportVorbereitung

Act_AllgemeinPrintVorbereitung

Act_DTPVorbereitung

Act_DTPEdit

Act_DTP_Preview
Druckvorschau

Act_ArtSN_Select
verfügbare Seriennummern übernehmen

Act_PosSerien
Seriennummernverwaltung

Act_RechAnschrift
Vorgangsanschrift (Kunden) wählen

Act_LiefAnschrift
abweichende Lieferanschrift

Act_VorTermin
Termin für diesen Vorgang anlegen

Act_VorTerminList
Terminliste für diesen Vorgang ansehen

DataSetCancel1
letzte Änderung an der Position rückgängig machen

DataSetDelete1
lösche die angewählte Position aus dem Vorgang

DataSetEdit1
Bearbeiten

DataSetFirst1
Erster

DataSetLast1
Letzter

AFS-Maskeneditor

DataSetNext1

Nächster

DataSetPrior1

Vorheriger

DataSetRefresh1

Aktualisieren

Act_VorgangsDruck

ausdrucken des Vorgangs

Act_VorgangsPreView

Druckvorschau für diesen Vorgang

Act_VorgangFormular

wählen Sie das Formular zum Ausdrucken aus

Act_VorgangEdit

verändern des Ausdrucks

Act_Bezahlung

Zahlung eingeben

Act_VorgangAbschliessen

Vorgang abschließen

Act_StatusBearbeiten

Status bearbeiten

Act_Grafik

grafische Auswertung

Act_VorgangFortfuehren

Vorgangfortführen

Act_VorStorno

Vorgang stornieren

Act_DruckerAssistent

Druckassistent

Act_Drucken

Act_Serienbrief

Textverarbeitung mit Daten aus dem Vorgang

Act_RechLiefAnschrift

Vorgangsanschrift (Lieferanten) wählen

Act FORVorbereitung

Action Kopien

Anzahl der Ausdrucke festlegen

Action ArtikelGrid

Spalteneditor

Act PosSchieben

Position verschieben an eine andere Position

Action Up

Position eine Stufe nach oben schieben

Action Down

Position eine Stufe nach unten schieben

Action PosInsert

nummiiert die Positionen neu

Action1 PosNum

eine neue Position anlegen

Action AllPrint

alle ungedruckten Vorgänge der gleichen Art drucken

Action History

zeigt die History des Artikel mit dieser Anschrift

Act Lagerbuchen

Wareneingang- und Warenausgang buchen

Action Kopieren

kopiert den gesamten Vorgang

Action Grunddaten

Action Posten

Action Verbuchen

Vorgang verbuchen

Action Mail

per E-Mail senden, statt drucken

Action Fax

per Fax senden

AFS-Maskeneditor

Action_PrintFrage

Drucken mit Auswahl

Action_PrintCFG

Einstellungen

Action_PreiseOff

Preise aus

Action_ZeigeAdresse

Zeigt die Adresse des Vorgangs an

Action_ZeigeArtikel

Artikel anzeigen

Action_ArtikelSQLSuche

Artikelsuche per SQL (Wildcards)

Action_VorMail_Formular

wählen Sie das Formular zum Mailen aus

Action_VorFax_Formular

wählen Sie das Formular zum Faxen aus

Action_VorAFS_Formular

wählen Sie das Formular für das AFSBestellsystem aus

Action_AFSBestellsystem

AFS-Bestellsystem - Bestellung mailen zu einem Partner mit AFS-Programm

Action_RepKunden

Vorgangsanschrift (Kunden) wählen

Action_RepLief

Vorgangsanschrift (Lieferanten) wählen

Action_RepPrint

Drucken mit Auswahl

Action_RepInter

Vorgangsanschrift (Interessenten) wählen

Action_ArtikelWandeln

in Artikel wandeln (anlegen)

Action_HandyVertrag

Action_VertragAnlegen

Seite 102

Vertrag in der Vertragsverwaltung anlegen

Action_VertragZeigen

Verträge, die aus diesem Vorgang angelegt wurden anzeigen

Action_VertragKundenZeigen

Verträge, des Kunden anzeigen

Action_HandyPreisberechnen

Preis für Gerät + Vertrag neu berechnen

Action_AbAnschrift

Abweichende Lieferanschrift

Action_DimAnzeige

Dimensionsanzeige

Action_EKAktual

EK's neu aktualisieren (sollten sich die EK's seit der Vorgangserstellung verändert haben, werden diese im Vorgang aktualisiert und der Rohertrag neu ermittelt)

Action_VKAktual

VK's neu aktualisieren (sollten sich die VK's seit der Vorgangserstellung verändert haben, werden diese im Vorgang aktualisiert und der Rohertrag neu ermittelt)

Action_VKGruppe

Wechseln Sie für diesen Vorgang die VK-Gruppe. Bei der automatischen Preisfindung wird eine andere Preisgruppe benutzt. Diese Einstellung gibt nur für diese Eingabe und für den geöffneten Vorgang!

Action_TabDes_Doku

Spaltendesigner

Action_Script_Edit

Druckerscript bearbeiten

Action_Preisliste

Action_Ansprechpartner_Speichern

Speichert den Ansprechpartner

Action_Adresse_Speichern

Speichert die Adresse

Action_Artikel_Speichern

Speichert den Artikel

Action_Warengruppe_Speichern

AFS-Maskeneditor

Aktionen für Mitarbeiter

Act_Designer
Spaltendesigner

Action_Adr_SMS
SMS-Senden

Action_Anruf_Handy
Handy anrufen

Action_Print
Drucken der Liste

Action_PrintCFG
Einstellungen der Ausdrücke

Action_TabDes_Doku
Spaltendesigner

Aktionen für MwSt-Einstellungen

Act_GridEdit

Act_OK

Act_Drucken
ausdrucken der Notiz

Act_Formular
verändern des Ausdrucks

Act_PreView
zeigt Ihnen eine Druckvorschau an

Act_FormWahl
wählen Sie das Formular zum Ausdrucken aus

Act_FibuKonto
Konto aus AFS-Fibu wählen

Aktionen der neuen Aufträge

Act Artikelsuchen

Artikel suchen

Act PosSpei Neu

Position speichern und eine neue Position anlegen

DataSetPost1

Position speichern

Act PosKopieren

angewählte Position kopieren

Act Vor Post

Vorgang speichern

Act Vor PostAndNeu

Vorgang speichern und nächsten Vorgang anlegen

Act PosGridDesign

Postentabelle verändern

Act DTP Print

ausdrucken des Vorgangs

Act ReportEdit

Drucken

Act ReportPrint

Reportgenerator

Act ReportPreview

Druckvorschau

Act ReportVorbereitung

Act AllgemeinPrintVorbereitung

Act DTPVorbereitung

Act DTPEdit

Act DTP Preview

Druckvorschau

Act ArtSN Select

verfügbare Seriennummern übernehmen

AFS-Maskeneditor
Act_PosSerien
Seriennummernverwaltung

Act_RechAnschrift
Vorgangsanschrift (Kunden) wählen

Act_LiefAnschrift
abweichende Lieferanschrift

Act_VorTermin
Termin für diesen Vorgang anlegen

Act_VorTerminList
Terminliste für diesen Vorgang ansehen

DataSetCancel1
letzte Änderung an der Position rückgängig machen

DataSetDelete1
lösche die angewählte Position aus dem Vorgang

DataSetEdit1
Bearbeiten

DataSetFirst1
Erster

DataSetLast1
Letzter

DataSetNext1
Nächster

DataSetPrior1
Vorheriger

DataSetRefresh1
Aktualisieren

Act_VorgangsDruck
ausdrucken des Vorgangs

Act_VorgangsPreView
Druckvorschau für diesen

Act_VorgangFormular
wählen Sie das Formular zum Ausdrucken aus dem Vorgang

Act_VorgangEdit
verändern des Ausdrucks

Act Bezahlung

Zahlung eingeben

Act VorgangAbschliessen

Vorgang abschließen

Act StatusBearbeiten

Status bearbeiten

Act Grafik

grafische Auswertung

Act VorgangFortfuehren

Vorgangfortführen

Act VorStorno

Vorgang stornieren

Act DruckerAssistent

Druckassistent

Act Drucken

Act Serienbrief

Textverarbeitung mit Daten aus dem Vorgang

Act RechLiefAnschrift

Vorgangsanschrift (Lieferanten) wählen

Act FORVorbereitung

Action Kopien

Anzahl der Ausdrücke festlegen

Action ArtikelGrid

Spalteneditor

Act PosSchieben

Position verschieben an eine andere Position

Action Up

Position eine Stufe nach oben schieben

Action Down

Position eine Stufe nach unten schieben

Action PosInsert

eine neue Position anlegen

AFS-Maskeneditor

Action1_PosNum

nummiiert die Positionen neu

Action_AllPrint

alle ungedruckten Vorgänge der gleichen Art drucken

Action_History

zeigt die History des Artikel mit dieser Anschrift

Act_Lagerbuchen

Wareneingang- und Warenausgang buchen

Action_Kopieren

kopiert den gesamten Vorgang

Action_Grunddaten

Action_Posten

Action_Verbuchen

Vorgang verbuchen

Action_Mail

per E-Mail senden, statt drucken

Action_Fax

per Fax senden

Action_PrintFrage

Drucken mit Auswahl

Action_PrintCFG

Einstellungen

Action_PreiseOff

Preise aus

Action_ZeigeAdresse

Zeigt die Adresse des Vorgangs an

Action_ZeigeArtikel

Artikel anzeigen

Action_ArtikelSQLSuche

Artikelsuche per SQL (Wildcards)

Action_VorMail_Formular

wählen Sie das Formular zum Mailen aus

Action_VorFax_Formular

wählen Sie das Formular zum Faxen aus

Action_VorAFS_Formular

wählen Sie das Formular für das AFSBestellsystem aus

Action_AFSBestellsystem

AFS-Bestellsystem - Bestellung mailen zu einem Partner mit AFS-Programm

Action_RepKunden

Vorgangsanschrift (Kunden) wählen

Action_RepLief

Vorgangsanschrift (Lieferanten) wählen

Action_RepPrint

Drucken mit Auswahl

Action_RepInter

Vorgangsanschrift (Interessenten) wählen

Action_ArtikelWandeln

in Artikel wandeln (anlegen)

Action_HandyVertrag

Action_VertragAnlegen

Vertrag in der Vertragsverwaltung anlegen

Action_VertragZeigen

Verträge, die aus diesem Vorgang angelegt wurden anzeigen

Action_VertragKundenZeigen

Verträge, des Kunden anzeigen

Action_HandyPreisberechnen

Preis für Gerät + Vertrag neu berechnen

Action_AbAnschrift

Abweichende Lieferanschrift

Action_DimAnzeige

Dimensionsanzeige

Action_EKAktual

EK's neu aktualisieren (sollten sich die EK's seit der Vorgangserstellung verändert haben, werden diese im Vorgang aktualisiert und der Rohertrag neu ermittelt)

Action_VKAktual

VK's neu aktualisieren (sollten sich die VK's seit der Vorgangserstellung

AFS-Maskeneditor

verändert haben, werden diese im Vorgang aktualisiert und der Rohertrag neu ermittelt)

Action VKGruppe

Wechseln Sie für diesen Vorgang die VK-Gruppe. Bei der automatischen Preisfindung wird eine andere Preisgruppe benutzt. Diese Einstellung gibt nur für diese Eingabe und für den geöffneten Vorgang!

Action TabDes Doku

Spaltendesigner

Action Script Edit

Druckerscript bearbeiten

Action Preisliste

Action Ansprechpartner Speichern

Speichert den Ansprechpartner

Action Adresse Speichern

Speichert die Adresse

Action Artikel Speichern

Speichert den Artikel

Action Warengruppe Speichern

Speichert die Warengruppe

Action Drucke EigenesFormular

Action BuchenOhneDruck

manuelle Lagerbuchung durchführen (ohne Ausdruck)

Action Vertrag Speichern

Speichert den Vertrag

Action KundeBesteller

Besteller (Kunden) wählen

Action Gutschrift Verechnen

Gutschrift verrechnen

Action InteressentBesteller

Besteller (Interessant) wählen

Action PosSort

Sortieren der Positionen

Action Vorgangsnummer

Seite 110

Action PostenInsert
Posten einfügen

Action ArtikelVar1
Spaltendesigner

Action ArtikelVar2
Spaltendesigner

Action Vorverfolgung
Vorgangsverfolgung

Action RueckstandAnz
Rückstand anzeigen - Zeigt die Rückstandsverwaltung mit den Artikeln an,
die in diesem Vorgang vorhanden sind!

Action EDIRechnungsliste
EDI-Rechnungsliste

Action Bestellung Erzeugen
Bestellung für diese Position erzeugen

Action Bestellung alleErzeugen
komplette Bestellung für diesen Vorgang erzeugen

Action Geraet Save
diesem Vorgang zuweisen

Action Geraet Del
Zuweisung aufheben

Action Geraet View
zeige Gerät an

Action Geraet Aus
Auswählen

Action Geraet Search
Geräte sucheN

Action Geraet Adresse
nur Geräte des Kunden anzeigen

Action Geraet Kunde
Gerät dieser Adresse zuweisen

Action Geraet Spalten
Spaltendesign

AFS-Maskeneditor
Action Filiale Leeren
Filialezuweisung entfernen

Action PrintPDF
als PDF ausgeben

Action PrintOn
zeigt die Druckzusatzfunktionen

Action PrintOff
blendet die Druckzusatzfunktionen aus

Action Mahnung Erzeugen
Vorgang mahnen

Action Mahnung Nochmals
letzte Mahnung nochmals ausgeben

Action Vorgangsverfolgung
Sendungsverfolgung öffnen

Action Pos Geraet Del
Gerätezuweisung entfernen

Action POS Geraet Search
Gerät suchen

Action ArtikelNummernSuche

Aktionen für neue Angebote

Act Artikelsuchen
Position speichern und eine neue Position anlegen

Act PosSpei Neu
Position speichern

DataSetPost1
angewählte Position kopieren

Act PosKopieren
Vorgang speichern

Act Vor Post
Vorgang speichern und nächsten Vorgang anlegen

Act Vor PostAndNeu

Postentabelle verändern

Act_PosGridDesign
ausdrucken des Vorgangs

Act_DTP_Print
Reportgenerator

Act_ReportEdit
Drucken

Act_ReportPrint
Druckvorschau

Act_ReportPreview

Act_ReportVorbereitung

Act_AllgemeinPrintVorbereitung

Act_DTPVorbereitung

Act_DTPEdit

Act_DTP_Preview
Druckvorschau

Act_ArtSN_Select
verfügbare Seriennummern übernehmen

Act_PosSerien
Seriennummernverwaltung

Act_RechAnschrift
Vorgangsanschrift (Kunden) wählen

Act_LiefAnschrift
abweichende Lieferanschrift

Act_VorTermin
Termin für diesen Vorgang anlegen

Act_VorTerminList
Terminliste für diesen Vorgang ansehen

AFS-Maskeneditor

DataSetCancel1

letzte Änderung an der Position rückgängig machen

DataSetDelete1

lösche die angewählte Position aus dem Vorgang

DataSetEdit1

Bearbeiten

DataSetFirst1

Erster

DataSetLast1

Letzter

DataSetNext1

Nächster

DataSetPrior1

Vorheriger

DataSetRefresh1

Aktualisieren

Act_VorgangsDruck

ausdrucken des Vorgangs

Act_VorgangsPreView

Druckvorschau für diesen Vorgang

Act_VorgangFormular

wählen Sie das Formular zum Ausdrucken aus

Act_VorgangEdit

verändern des Ausdrucks

Act_Bezahlung

Zahlung eingeben

Act_VorgangAbschliessen

Vorgang abschließen

Act_StatusBearbeiten

Status bearbeiten

Act_Grafik

grafische Auswertung

Act_VorgangFortfuehren

Vorgangfortführen

Seite 114

Act_VorStorno

Vorgang stornieren

Act_DruckerAssistent

Druckassistent

Act_Drucken

Act_Serienbrief

Textverarbeitung mit Daten aus dem Vorgang

Act_RechLiefAnschrift

Vorgangsanschrift (Lieferanten) wählen

Act_FORVorbereitung

Action_Kopien

Anzahl der Ausdrücke festlegen

Action_ArtikelGrid

Spalteneditor

Act_PosSchieben

Position verschieben an eine andere Position

Action_Up

Position eine Stufe nach oben schieben

Action_Down

Position eine Stufe nach unten schieben

Action_Interesent

Interessentenanschrift auswählen: z.B. für Angebot

Action_PosInsert

eine neue Position anlegen

Action1_PosNum

nummiert die Positionen neu

Action_AllPrint

alle ungedruckten Vorgänge der gleichen Art drucken

Action_History

zeigt die History des Artikel mit dieser Anschrift

Act_Lagerbuchen

Wareneingang- und Warenausgang buchen

AFS-Maskeneditor

Action Kopieren

kopiert den gesamten Vorgang

Action Grunddaten

Action Posten

Action Verbuchen

Vorgang verbuchen

Action Mail

per E-Mail senden, statt drucken

Action Fax

per Fax senden

Action PrintFrage

Drucken mit Auswahl

Action PrintCFG

Einstellungen

Action PreiseOff

Preise aus

Action ZeigeAdresse

Zeigt die Adresse des Vorgangs an

Action ZeigeArtikel

Artikel anzeigen

Action ArtikelSQLSuche

Artikelsuche per SQL (Wildcards)

Action VorMail Formular

wählen Sie das Formular zum Mailen aus

Action VorFax Formular

wählen Sie das Formular zum Faxen aus

Action RepKunden

Vorgangsanschrift (Kunden) wählen

Action RepLief

Vorgangsanschrift (Lieferanten) wählen

Action RepPrint

Drucken mit Auswahl

Action_RepInter

Vorgangsanschrift (Interessenten) wählen

Action_ArtikelWandeln

in Artikel wandeln (anlegen)

Action_HandyVertrag

Action_VertragAnlegen

Vertrag in der Vertragsverwaltung anlegen

Action_VertragZeigen

Verträge, die aus diesem Vorgang angelegt wurden anzeigen

Action_VertragKundenZeigen

Verträge, des Kunden anzeigen

Action_HandyPreisberechnen

Preis für Gerät + Vertrag neu berechnen

Action_AbAnschrift

Abweichende Lieferanschrift

Action_DimAnzeige

Dimensionsanzeige

Action_EKAktual

EK's neu aktualisieren (sollten sich die EK's seit der Vorgangserstellung verändert haben, werden diese im Vorgang aktualisiert und der Rohertrag neu ermittelt)

Action_VKAktual

VK's neu aktualisieren (sollten sich die VK's seit der Vorgangserstellung verändert haben, werden diese im Vorgang aktualisiert und der Rohertrag neu ermittelt)

Action_VKGruppe

Wechseln Sie für diesen Vorgang die VK-Gruppe.

Bei der automatischen Preisfindung wird eine andere Preisgruppe benutzt. Diese Einstellung gibt nur für diese Eingabe und für den geöffneten Vorgang!

Action_TabDes_Doku

Spaltendesigner

Action_Script_Edit

Druckerscript bearbeiten

Action_Preisliste

AFS-Maskeneditor

Action Ansprechpartner Speichern

Speichert den Ansprechpartner

Action Adresse Speichern

Speichert die Adresse

Action Artikel Speichern

Speichert den Artikel

Action Warengruppe Speichern

Speichert die Warengruppe

Action Drucke EigenesFormular

Action BuchenOhneDruck

manuelle Lagerbuchung durchführen (ohne Ausdruck)

Action Vertrag Speichern

Speichert den Vertrag

Action KundeBesteller

Besteller (Kunden) wählen

Action Gutschrift Verechnen

Gutschrift verrechnen

Action InteressentBesteller

Besteller (Interessent) wählen

Action PosSort

Sortieren der Positionen

Action Vorgangsnummer

Action PostenInsert

Posten einfügen

Action ArtikelVar1

Spaltendesigner

Action ArtikelVar2

Spaltendesigner

Action Vorverfolgung

Vorgangsverfolgung

Action RueckstandAnz

Rückstand anzeigen - Zeigt die Rückstandsverwaltung mit den Artikeln an, die in diesem Vorgang vorhanden sind!

Action EDIRechnungsliste
EDI-Rechnungsliste

Action Bestellung Erzeugen
Bestellung für diese Position erzeugen

Action Bestellung alleErzeugen
komplette Bestellung für diesen Vorgang erzeugen

Action Geraet Save
diesem Vorgang zuweisen

Action Geraet Del
Zuweisung aufheben

Action Geraet View
zeige Gerät an

Action Geraet Aus
Auswählen

Action Geraet Search
Geräte suche

Action Geraet Adresse
nur Geräte des Kunden anzeigen

Action Geraet Kunde
Gerät dieser Adresse zuweisen

Action Geraet Spalten
Spaltendesign

Action Filiale Leeren
Filialezuweisung entfernen

Action PrintPDF
als PDF ausgeben

Action PrintOn
zeigt die Druckzusatzfunktionen

Action PrintOff
blendet die Druckzusatzfunktionen aus

Action Mahnung Erzeugen

AFS-Maskeneditor
Vorgang mahnen

Action Mahnung Nochmals
letzte Mahnung nochmals ausgeben

Action Vorgangsverfolgung
Sendungsverfolgung öffnen

Action Pos Geraet Del
Gerätezuweisung entfernen

Action POS Geraet Search
Gerät suchen

Action ArtikelNummernSuche
Artikel suchen nach Nummern

Action ArtikelBezeichnungSuche
Artikel suchen nach Bezeichnung

Action GeraetAnlegen
die Position als Gerät in der Geräteverwaltung anlegen

Action GeraetAnzeigen
zeige Gerät an

Action GeraetAnlegenVor
neues Gerät in der Geräteverwaltung anlegen

Action GeraetAnlegenRep
neues Gerät in der Geräteverwaltung anlegen

Action VE Set
VE-Übernehmen

Action VE Cancel
Abbruch

Action magiceddy
magic EDI-Ausgabe

Action Lieferanschrift Frei
freie Lieferanschrift

Action Lieferanschrift DEF
definierte Lieferanschrift

Action Produktion IST
IST-Werte Erfassung

Action Abschlag Druck

Abschlagsrechnung drucken

Action_Abschlag_Save
Speichern

Action_Abschlag_Eingabe
Abschlag erfassen

Action_Abschlag_Formular
Druckeinstellungen

Action_Abschlag_Del
Abschlag entfernen

Action_Abschlag_Storno
Abschlag stornieren

Action_StartTime
Startzeit = jetzt

Action_EndTime
Endzeit = jetzt

Action_Art_Such_Lief
nur lieferbare Artikel anzeigen

Action_Art_Such_Be
nur Artikel mit Bestand anzeigen

Action_Kalk_ZwischenSumme
Kalkulation der Zwischensumme -
Zwischensumme auf die Eingabe umrechnen

Action_Kalk_ZwischenRoh
Kalkulation des Rohertrags der Zwischensumme - Rohertrag der Zwischen-
summe auf die Eingabe umrechnen

Action_Kalk
Kalkulation des gesamten Vorgangs

Action_Kalk_EndBSumme
Kalkulation der Bruttoendsumme - Endsumme auf die Eingabe umrechnen

Action_Kalk_EndRoh
Kalkulation des Rohertrags - Rohertrag umrechnen

Action_Kalk_EndNSumme
Kalkulation der Nettoendsumme - Endsumme auf die Eingabe umrechnen

AFS-Maskeneditor

Action_Kalk_EndRohPro

Kalkulation des Rohertrags in % Rohertrag umrechnen

Action_Postenverdichten

Postenverdichten - gleiche Posten werden zusammengefasst

Aktionen der Notizenverwaltung

Act_Close

Notizfenster schließen

Act_AktNeu

neue Rubrik für Notizen einrichten

Act_AktKill

löscht eine vorhandene Rubrik

Act_AktRename

Rubrik umbenennen

Act_Drucken

ausdrucken der Notiz

Act_Formular

verändern des Ausdrucks

Act_PreView

zeigt Ihnen eine Druckvorschau an

Act_FormWahl

wählen Sie das Formular zum Ausdrucken aus

Aktionen für Rechnungen

Act_Artikelsuchen

Artikel suchen

Act_PosSpei_Neu

Position speichern und eine neue Position anlegen

DataSetPost1

Position speichern

Act_PosKopieren

angewählte Position kopieren

Act_Vor_Post

Vorgang speichern

Act_Vor_PostAndNeu

Vorgang speichern und nächsten Vorgang anlegen

Act_PosGridDesign

Postentabelle verändern

Act_DTP_Print

ausdrucken des Vorgangs

Act_ReportEdit

Reportgenerator

Act_ReportPrint

Drucken

Act_ReportPreview

Druckvorschau

Act_ReportVorbereitung

Act_AllgemeinPrintVorbereitung

Act_DTPVorbereitung

Act_DTPEdit

Act_DTP_Preview

Druckvorschau

Act_ArtSN_Select

verfügbare Seriennummern übernehmen

Act_PosSerien

Seriennummernverwaltung

Act_RechAnschrift

Vorgangsanschrift (Kunden) wählen

Act_LiefAnschrift

abweichende Lieferanschrift

Act_VorTermin

Termin für diesen Vorgang anlegen

Act_VorTerminList

Terminliste für diesen Vorgang ansehen

AFS-Maskeneditor

DataSetCancel1

letzte Änderung an der Position rückgängig machen

DataSetDelete1

lösche die angewählte Position aus dem Vorgang

DataSetEdit1

Bearbeiten

DataSetFirst1

Erster

DataSetLast1

Letzter

DataSetNext1

Nächster

DataSetPrior1

Vorheriger

DataSetRefresh1

Aktualisieren

Act_VorgangsDruck

ausdrucken des Vorgangs

Act_VorgangsPreView

Druckvorschau für diesen Vorgang

Act_VorgangFormular

wählen Sie das Formular zum Ausdrucken aus

Act_VorgangEdit

verändern des Ausdrucks

Act_Bezahlung

Zahlung eingeben

Act_VorgangAbschliessen

Vorgang abschließen

Act_StatusBearbeiten

Status bearbeiten

Act_Grafik

grafische Auswertung

Act_VorgangFortfuehren

Vorgangfortführen

Act_VorStorno

Vorgang stornieren

Act_DruckerAssistent

Druckassistent

Act_Drucken

Act_Serienbrief

Textverarbeitung mit Daten aus dem Vorgang

Act_RechLiefAnschrift

Vorgangsanschrift (Lieferanten) wählen

Act_FORVorbereitung

Action_Kopien

Anzahl der Ausdrücke festlegen

Action_ArtikelGrid

Spalteneditor

Act_PosSchieben

Position verschieben an eine andere Position

Action_Up

Position eine Stufe nach oben schieben

Action_Down

Position eine Stufe nach unten schieben

Action_PosInsert

eine neue Position anlegen

Action1_PosNum

nummierierte Positionen neu

Action_AllPrint

alle ungedruckten Vorgänge der gleichen Art drucken

Action_History

zeigt die History des Artikel mit dieser Anschrift

Act_Lagerbuchen

Wareneingang- und Warenausgang buchen

Action_Kopieren

kopiert den gesamten Vorgang

AFS-Maskeneditor

Action Grunddaten

Action Posten

Action Verbuchen

Vorgang verbuchen

Action Mail

per E-Mail senden, statt drucken

Action Fax

per Fax senden

Action PrintFrage

Drucken mit Auswahl

Action PrintCFG

Einstellungen

Action PreiseOff

Preise aus

Action ZeigeAdresse

Zeigt die Adresse des Vorgangs an

Action ZeigeArtikel

Artikel anzeigen

Action ArtikelSQLSuche

Artikelsuche per SQL (Wildcards)

Action VorMail Formular

wählen Sie das Formular zum Mailen aus

Action VorFax Formular

wählen Sie das Formular zum Faxen aus

Action RepKunden

Vorgangsanschrift (Kunden) wählen

Action RepLief

Vorgangsanschrift (Lieferanten) wählen

Action RepPrint

Drucken mit Auswahl

Action RepInter

Vorgangsanschrift (Interessenten) wählen

Action ArtikelWandeln
in Artikel wandeln (anlegen)

Action HandyVertrag

Action VertragAnlegen
Vertrag in der Vertragsverwaltung anlegen

Action VertragZeigen
Verträge, die aus diesem Vorgang angelegt wurden anzeigen

Action VertragKundenZeigen
Verträge des Kunden anzeigen

Action HandyPreisberechnen
Preis für Gerät + Vertrag neu berechnen

Action AbAnschrift
Abweichende Lieferanschrift

Action DimAnzeige
Dimensionsanzeige

Action EKAktual
EK's neu aktualisieren (sollten sich die EK's seit der Vorgangserstellung verändert haben, werden diese im Vorgang aktualisiert und der Rohertrag neu ermittelt)

Action VKAktual
VK's neu aktualisieren (sollten sich die VK's seit der Vorgangserstellung verändert haben, werden diese im Vorgang aktualisiert und der Rohertrag neu ermittelt)

Action VKGruppe
Wechseln Sie für diesen Vorgang die VK-Gruppe. Bei der automatischen Preisfindung wird eine andere Preisgruppe benutzt. Diese Einstellung gibt nur für diese Eingabe und für den geöffneten Vorgang!

Action TabDes Doku
Spaltendesigner

Action Script Edit
Druckerscript bearbeiten

Action Preisliste

Action Ansprechpartner Speichern
Speichert den Ansprechpartner

AFS-Maskeneditor
Action Adresse Speichern
Speichert die Adresse

Action Artikel Speichern
Speichert den Artikel

Action Warengruppe Speichern
Speichert die Warengruppe

Action Drucke EigenesFormular

Action BuchenOhneDruck
manuelle Lagerbuchung durchführen (ohne Ausdruck)

Action Vertrag Speichern
Speichert den Vertrag

Action KundeBesteller
Besteller (Kunden) wählen

Action Gutschrift Verechnen
Gutschrift verrechnen

Action InteressentBesteller
Besteller (Interessant) wählen

Action PosSort
Sortieren der Positionen

Action Vorgangsnummer

Action PostenInsert
Posten einfügen

Action ArtikelVar1
Spaltendesigner

Action ArtikelVar2
Spaltendesigner

Action Vorverfolgung
Vorgangsverfolgung

Action RueckstandAnz
Rückstand anzeigen - Zeigt die Rückstandsverwaltung mit den Artikeln an,
die in diesem Vorgang vorhanden sind!

Action EDIRechnungsliste

EDI-Rechnungsliste

Action Bestellung Erzeugen

Bestellung für diese Position erzeugen

Action Bestellung alleErzeugen

komplette Bestellung für diesen Vorgang erzeugen

Action Geraet Save

diesem Vorgang zuweisen

Action Geraet Del

Zuweisung aufheben

Action Geraet View

zeige Gerät an

Action Geraet Aus

Auswählen

Action Geraet Search

Geräte suche

Action Geraet Adresse

nur Geräte des Kunden anzeigen

Action Geraet Kunde

Gerät dieser Adresse zuweisen

Action Geraet Spalten

Spaltendesign

Action PrintPDF

als PDF ausgeben

Action PrintOn

zeigt die Druckzusatzfunktionen

Action PrintOff

blendet die Druckzusatzfunktionen aus

Action Mahnung Erzeugen

Vorgang mahnen

Action Mahnung Nochmals

letzte Mahnung nochmals ausgeben

Action Vorgangsverfolgung

Sendungsverfolgung öffnen

Action Pos Geraet Del

AFS-Maskeneditor
Gerätezuweisung entfernen

Action_POS_Geraet_Search
Gerät suchen

Action_ArtikelNummernSuche
Artikel suchen nach Nummern

Action_ArtikelBezeichnungSuche
Artikel suchen nach Bezeichnung

Action_GeraetAnlegen
die Position als Gerät in der Geräteverwaltung anlegen

Action_GeraetAnzeigen
zeige Gerät an

Action_GeraetAnlegenVor
neues Gerät in der Geräteverwaltung anlegen

Action_GeraetAnlegenRep
neues Gerät in der Geräteverwaltung anlegen

Action_VE_Set
VE-Übernehmen

Action_VE_Cancel
Abbruch

Action_magiceddy
magic EDI-Ausgabe

Action_Lieferanschrift_Frei
freie Lieferanschrift

Action_Lieferanschrift_DEF
definierte Lieferanschrift

Action_Produktion_IST
IST-Werte Erfassung

Action_Abschlag_Druck
Abschlagsrechnung drucken

Action_Abschlag_Save
Speichern

Action_Abschlag_Eingabe

Abschlag erfassen

Action_Abschlag_Formular
Druckeinstellungen

Action_Abschlag_Del
Abschlag entfernen

Action_Abschlag_Storno
Abschlag stornieren

Action_StartTime
Startzeit = jetzt

Action_EndTime
Endzeit = jetzt

Action_Art_Such_Lief
nur lieferbare Artikel anzeigen

Action_Art_Such_Be
nur Artikel mit Bestand anzeigen

Action_Kalk_EndBSumme

Aktionen des Terminplaners

Action_TerminBearbeiten
Termin bearbeiten

Action_ZeigeAdresse
Zeigt die Adresse des Termins an

Action_ZeigeArtikel
Artikel anzeigen

Action_ZeigeVorgang
Vorgang anzeigen

Action_InsertTerm
Termin anlegen

Action_Delete
Termin löschen

Action_PrintAus

AFS-Maskeneditor

Action PrintCFG

Aktionen des Textwriters

FileNewCmd

Erzeugt ein neuen Text

FileOpenCmd

öffnet ein Dokument

FileSaveCmd

Speichert den Text

FilePrintCmd

Druckt den Text

FileExitCmd

Beendet die Textverarbeitung

FileSaveAsCmd

Speichert den Text mit einen anderen Namen

Serienbrief

Druckt Serienbrief (alle Datensätze der Hauptdatenbank)

Serienfax

Serienfax senden

Serienmail

Serienmail senden

Act_EMail

Act_EMail Partner

Serienmail senden

EditUndoCmd

Rückgängig

EditCutCmd

Ausschneiden

EditCopyCmd

Kopieren

EditPasteCmd

Einfügen

Seite 132

EditFontCmd
Schriftart auswählen

Aktionen der Touchkasse

Action_Neu
Storno (neuer Kassivorgang)

Action_Kunde
einen Kunden auswählen

Action_Zahlen
Kunde will Zahlen

Action_Vorgangsuchen
sucht einen alten Kassenvorgang

ActionClose
Kasse beenden

Action_Schublade
Schublade öffnen

Action_Lieferant

Action_Parken
legt diesen Vorgang zur Seite, dieser kann später wieder abgerufen werden

Action_Holen
holt einen geparkten Vorgang zurück

Action_AltVorgang
alten Kassenvorgang nochmals bearbeiten

Action_PostenDes
Postentabelle verändern

Action_Artikel
Artikel bearbeiten

Action_Kunden
Kunden bearbeiten

Action_TouchMulti
Anzahl erfassen

Action_Touch_Waren_Left

AFS-Maskeneditor
Action Touch Waren Right

Action TOUCH ENTER
Eingabe abschliessen

Action Touch 1plus
Artikelanzahl um eins erhöhen

Action Touch 1minus
Artikelanzahl um senken erhöhen (min 1)

Action TastaturOnOff

Action weitereFunk
weitere Funktionen

Action CFG
Touchkonfigurator - stellen Sie den Touchbildschirm nach Ihren Wünschen ein

Action Maskeneditor
Maskeneditor

Action Vk aktualisieren

Action TBLPOSTEN Refrsh
Posten aktualisieren

Action TouchPosFill
Touchpanel automatisch aufbauen - Aufgrund der Warengruppen und Artikelvorgaben wird das Touchpanel mit Warengruppen und Artikeln gefüllt (diese müssen die Einstellung TouchPos angehakt haben)

Action Ruecknahme
Stelle den aktuellen Posten (i.d.R. den zu letzt kassierten) auf Rücknahme um

Action Kassen Tastatur

Action POPUP AUSWAHL
Auswahl

Action Gutschein
Gutscheinverwaltung

Action Gutschein New
Gutschein erstellen und kassieren

Action Zeiterfassung
Arbeitszeiterfassung

Action Gutschein Ein
Gutschein einlösen

Action AuswahlfensterCFG
Auswahlfenstereinstellung

Action Auswahl Close
Auswahlfenster schliessen

Act Artikelsuchen
Artikel suchen

Act PosKopieren
angewählte Position kopieren

Act AllgemeinPrintVorbereitung

Act ArtSN Select
verfügbare Seriennummern übernehmen

Act Zusatz
Zusatzdaten eingeben (Seriennummern, usw.)

DataSetCancel1
letzte Änderung an der Position rückgängig machen

Action Storno
lösche die angewählte Position aus dem Vorgang

Action Stornox
lösche die angewählte Position aus dem Vorgang

DataSetFirst1
Erster

DataSetInsert1
eine neue Position anlegen

DataSetLast1
Letzter

DataSetNext1
Nächster

DataSetPrior1
Vorheriger

AFS-Maskeneditor

DataSetRefresh1

Aktualisieren

Action_Kasseneinstellungen

Kasseneinstellungen

Action_Hardware

Kasse-Hardware einstellen

Act_SNEingeben

Seriennummern/Chargennummern eingeben

Action_ZBericht

Kassenabschluß (Z-Bericht)

Action_XBericht

Kassenprüfung (X-Bericht)

Action_RabattEingeben

Gesamtrabatt eingeben

Action_KasseneinstellungenTeil2

Kasseneinstellungen (Bedienung)

Action_PreisGlaetten

Preis glätten (Festpreis vorgeben)

Action_Lieferanschrift

Abweichende Anschrift eingeben (Lieferanschrift)

Action_Pos_Rabatt

Postenrabatt eingeben

Action_Entnahme

Kassen-Entnahme

Action_Einlage

Kassen-Einlage

Action_Preis_Aenderung

Preisänderung

Action_LiefVorgangZahlen

Lieferschein bezahlen

Action_RepVorgangZahlen

Reparaturvorgang bezahlen

Action_Bestellen

als Bestellung speichern

Action_Bestellung_Ed
Bestellung bearbeiten

Action_Bestellung_Lief
Bestellung beim Lieferanten

Action_Auftrag_Ed
Auftragbestätigung bearbeiten

Action_Auftrag
als Auftragsbestätigung speichern

Action_BestellungZahlen
Bestellung bezahlen

Action_AuftragZahlen
Auftragbestätigung bezahlen

Aktionen der Umsatzsteuerauswertung

Action_AusStart
Auswertung starten mit den gewählten Einstellungen

Action_Helfer
der Helfer erklärt das Fenster

Aktionen für das Verdichten von Vorgängen

Action_KundeSuchen
Kunden wählen dessen Vorgänge verdichtet werden sollen

Action_VorAuf
Auftragsbestätigung wählen

Action_VorLie
Lieferschein auswählen

Action_LieferantSuchen
Lieferanten wählen dessen Vorgänge verdichtet werden sollen

Aktionen der Vorgangsliste

Act_VorTerminList
Terminliste für diesen Vorgang ansehen

Act_Bezahlung
Zahlung eingeben

AFS-Maskeneditor

Act_VorgangAbschliessen

Vorgang abschließen

Act_StatusBearbeiten

Status bearbeiten

Act_Grafik

grafische Auswertung

Act_VorgangFortfuehren

Vorgangfortführen

Act_Lagerbuchen

Wareneingang- und Warenausgang buchen

Act_GridEdit

Spalteneditor

Action_Verbuchen

Vorgang verbuchen

Action_alleVerbuchen

Action_KopierenExecute

Action_Selection

Selektion dieses Datensatzes

Action_Selection_off

Selektion dieses Datensatzes aufheben

Action_Selection_alloff

Selektion aller Datensätze aufheben

Action_FibuKonten

Auswertung der Fibukontierung

Action_Versand

Versanddatei für externes Versendprogramm erstellen (mit Dialog)

Action_VersandEinstellung

Versandformulareinstellungen

Action_VersandSave

Versanddatei für externes Versendprogramm erstellen (ohne Dialog und erste Position im Dialog)

Action_Verdichten

Seite 138

Vorgang verdichten

Action AlleAbschliessen
alle Vorgänge abschließen

Action Vorverfolgung
Vorgangsverfolgung

Action RueckstandAnz
Rückstand anzeigen - Zeigt die Rückstandsverwaltung mit den Artikeln an,
die in diesem Vorgang vorhanden sind!

Action PosGrid
Spalteneditor

Act VorStorno
Vorgang stornieren

Action Vorgangsverfolgung
Sendungsverfolgung öffnen

Action letzteTage
Auswahl

Action Bestellung alleErzeugen
komplette Bestellung für diesen Vorgang

Action OK

Action BuchenOhneDruck
manuelle Lagerbuchung durchführen (ohne Ausdruck)

Act Vor Bearbeiten
Vorgang bearbeiten

Act Filter
Filter setzen

Act Print
Drucken der Liste

Act PreView
Einstellungen der Ausdrucke

Act Formularedit
der Helfer erklärt das Fenster

Action Interesent

AFS-Maskeneditor
Interesent suchen

Action_Kunden
Kunden suchen

Action_Lieferant
Lieferant suchen

Action_ZeigeAdresse
Zeigt die Adresse des Vorgangs an

Action_ZeigeArtikel
Artikel anzeigen

Action_ExcelExport
Export zu Excel

Action_WordExport
Export zu Word

Action_Einstellung_Speichern
Einstellung speichern

Aktionen der Warengruppen

Act_Warengruppe_up
eine Ebene rauf

Act_Warengruppe_down
eine Ebene tiefer

Act_Warengruppe_links
eine Ebene zurück

Act_Warengruppe_rechts
eine Ebene tiefer

Act_Print
Drucken der Warengruppen

Act_PreView
zeigt Ihnen eine Druckvorschau an

Act_Formularedit
verändern des Ausdrucks

Act_Formularselect
wählen Sie das Formular zum Ausdrucken aus

Act_GridEdit
Spalteneditor

Act_Close
Warengruppen schließen

Act_Rabatt
Rabattgruppen und Sonderpreise

Act_WarengruppeNeu
neue Warengruppe

Act_WarengruppeBearbeiten
Warengruppe bearbeiten

Act_WarengruppeLoeschen
Warengruppe löschen

Act_UnterwarengruppeNeu
eine untergeordnete Warengruppe anlegen

Action_Umsatz
Statistik

Action_Zubehoehr
Zubehörartikel definieren

Action_BezeichnungIntern

Action_SHOP_Anzeigen
Shopwarengruppen anzeigen

Aktionen für die Warengruppenauswertung

Action_Close
Abfrage schließen

Action_OK
Auswertung starten

Action_Grid
Spalteneditor

Action_BonDruck
Ausgabe auf dem Kasse-Bondrucker

Action_Helfer
der Helfer erklärt das Fenster

AFS-Maskeneditor

Action DruckenAus

Drucken mit Auswahl

Action PrintCFG

Einstellungen

Action BONArWa

Artikel nach Warengruppen (auf den Bondrucker)

Action ExcelExport

Export zu Excel

Action WordExport

Export zu Word

Action Filter

Filter setzen

Action Art Search

Suche starten

Aktionen des Wartungs- und Abwesens

Action Abbruch

Wartungs- und Aboverwaltung beenden

Action Mahnliste

zeigt die Liste der Vorgänge, die fällig sind

Act DruckerAssistent

Druckassistent

Action Text

Textverarbeitung

Act Drucken

Action Wandel

Wartungsvorgang in Vorgang wandeln

Action Wandeln

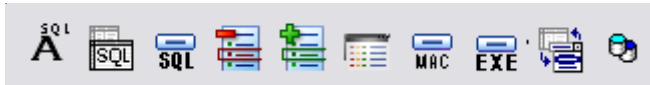
alle Wartungsvorgänge in Vorgänge wandeln

Action Entferne

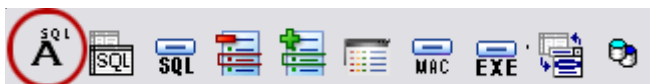
Vorgang aus der Liste entfernen

aktive Elemente

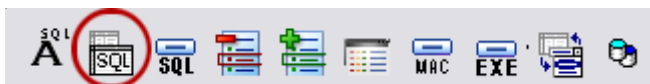
Hier können Sie Objekte erzeugen, mit denen Sie komplexe Ausgaben, Abfragen und Aktionen erstellen können.



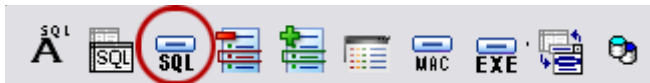
Die Menüleiste ist simpel aufgebaut und leicht verständlich. Zum auswählen der einzelnen Komponenten klicken Sie einfach auf den entsprechenden Button.



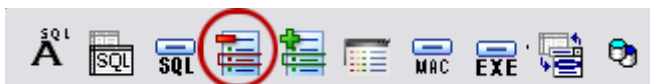
Wählen Sie diesen Button, um ein Label zu erstellen, bei welchem Sie SQL-Abfragen hinterlegen können. Sie können damit einzelne Ausgaben realisieren.



Über diesen Button erstellen Sie ein SQL-Grid, mit dem Sie mehrere Ausgaben für einen Datensatz oder unabhängig eines Datensatzes in einer Tabelle dargestellt bekommen.

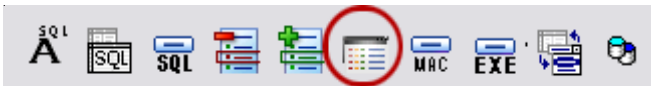


Wollen Sie Änderungen an einem Datensatz oder der kompletten Datenbank vornehmen (UPDATE-, SET-Befehle) wählen Sie diesen Button aus und geben Sie Ihren gewünschten Befehl ein.



Mit diesen Buttons können Sie eigene BasicScripts ausführen. Unterscheidungen finden Sie nur bei der Umsetzung und dem Aussehen der Buttons. Das Testen bei den aktiven Buttons ist um einiges leichter, weil Sie diese mit einem Klick auf „Umsetzen“ testen können. Bei den normalen Schaltern müssen Sie umsetzen, speichern und können erst dann die Funktionalität testen.

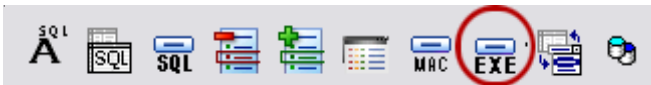
AFS-Maskeneditor



Benötigen Sie ein Script, das sofort beim Laden der Maske aktiv ist? Benutzen Sie dazu einfach den Button und klicken Sie auf eine Stelle im Fenster. Achten Sie darauf, dass das Laden der Maske bei großen Scripten beeinflusst wird.



Macros abzuspielen ist über diesen Button sehr einfach möglich. Wählen Sie das gewünschte Macro aus und spielen Sie es ab, sobald Sie auf den Schalter klicken.



Sie können auch Programme direkt aus dem Kaufmann/Manager ausführen - dieser Schalter ermöglicht Ihnen dieses. Stellen Sie ein, welches Programm Sie aufrufen wollen und speichern Sie.



Über diesen Button können Sie eine Dropdownliste (Combobox) einrichten, in der Sie SQL-Statements auswählen können. Die Dropdownliste wird im Maskeneditor zuerst mit einer Datenquelle, Datenfeld und SQL-Abfrage gefüllt.

Beim Laden der Fenster können Sie über einen Klick auf eine Selection der Combobox den Inhalt hinzufügen und speichern diesen dann wie gewohnt in einer gewünschten Maske mit F11.



Ermöglicht die Einbindung einer nicht sichtbaren Komponente zum Verknüpfen von Datenbanktabellen der Warenwirtschaft. Diese können dann auch in Formularen eingebettet werden oder für weitere Maskeneditorelemente verwendet werden.

Objekteigenschaften

Über die Objekteigenschaften können Sie Einstellungen an den einzelnen Komponenten vornehmen.

Hier erklären wir Ihnen die einzelnen Optionsmöglichkeiten.

Position

Legen Sie hier die Position, Größe und Sichtbarkeit des Objektes fest.

Aussehen

Das Aussehen der einzelnen Komponenten können Sie hier sehr einfach gestalten.

Datenbanken***Datenquelle***

Hier wählen Sie die Datenquelle. Aus der dort hinterlegten Tabelle können Felder in den SQL-Befehl integriert werden, sodass beim Scrollen in dieser Tabelle der aktive Datensatz aktualisiert und durch eine SQL-Anweisung aufgerufen wird.

Datenfeld

Das Feld der Tabelle wird hier ausgewählt.

Eingabe***Eingeschaltet***

Diese Funktion ermöglicht es Ihnen das gewählte Objekt zu aktivieren oder zu deaktivieren ohne es zu löschen.

Aktionen

Die vorher beschriebenen Aktionen können mit Hilfe dieser Funktion aufgerufen werden. Näheres finden Sie bei den AKTIONEN.

Anweisungen

Das Anweisungsfenster muss immer aufgeklappt sein.
Wenn nur etwas in der Zeile steht wird es nicht ausgeführt.

SQL-Anweisung

Hinter diesem Punkt befindet sich das SQL-Anweisungsfenster. Jegliche SQL_anweisungen werden hier eingegeben.

Bsp.:

Als Datenquelle wählen wir scr_Kunden. Die Anweisung lautet:

```
SELECT * FROM auftrag WHERE adresse1=:adresse
```

Mit dieser Anweisung werden alle Aufträge gezeigt, die dem derzeit aktiven Kunden zugeordnet sind.

AFS-Maskeneditor

Die Anweisung ist wie folgt aufgebaut:

SELECT * FROM = Zeige alles aus

auftrag = Name der Tabelle

WHERE = in der

adresse1= = Inhalt des Feldes ist gleich

: = ersetzt den Namen, der ausgewählten Tabelle
(in diesem Fall „scr_kunden.“)

adresse = diesem Inhalt des Feldes

Die Auswahl der Tabellen ist von Maske zu Maske unterschiedlich. Tabellen, die in der Kundenbearbeitung zu finden sind, sind möglicherweise nicht in der Artikelbearbeitung geöffnet.

Basic-Programm

Den Quelltext geben Sie bei diesem Punkt an.

Mit dieser Funktion ist es Ihnen beispielsweise möglich, vorhandene Komponenten (z.B. SQL-Grids) zu manipulieren, ohne den Maskeneditor aufzurufen. Das ist nur eine der vielen Optionen, die Ihnen zur Verfügung steht.

Dateiauswahl

Auch Verknüpfungen zu externen Programmen können hinzugefügt werden. Hierzu klicken Sie einfach auf die Schaltfläche und wählen die gewünschte Verknüpfung aus.

AFS- Basicscript-Sprache

Die Basicscriptsprache richtet sich nach der Basicsyntax gepaart mit den Objekten, welche Delphi zur Verfügung stellt.

Die Basicscriptsprache basiert auf der Basicsyntax im Zusammenspiel mit den Objekten der Delphi-Entwicklungsumgebung.

Die Möglichkeit ein Pascalscript anzubieten bot sich ebenfalls, doch auf Grund der bekannteren Syntax entschieden wir uns für Basic.

1. Variablen müssen für den normalen Gebrauch NICHT deklariert werden. Dieses gilt nicht, wenn eine so genannte VAR Variable benötigt wird. In diesem Fall muss diese bereits einen Wert bekommen haben oder mit DIM deklariert werden.

2. Sie haben die Möglichkeit auf alle Objekte zuzugreifen, die sich in der Form befinden. Die Objekte werden mit Namen angesprochen, welchen Sie aus dem Objektbaum des Maskeneditors entnehmen können.

3. Zugriff auf die nicht sichtbaren Tabellenobjekte können Sie über deren Namen entnehmen:

Namen sind folgendermaßen zu ermitteln:

3.1 Formulareditor

hier sehen Sie alle möglichen Tabellenobjekte mit TBL_xxx bzw QU_xxx bzw im Maskeneditor bei Datenobjekten im Punkt Datenquelle statt SCR_xxx TBL_xxx

3.2 Die Feldnamen können Sie ebenfalls aus dem Formulareditor bzw. Maskeneditor entnehmen

Beispiel (aus der Adressmaske):

```
s=TBL_Kunden.FieldByName(„Suchbegriff“).AsString
TabGrunddaten.Caption=s
TBL_Kunden.LoadAll
Tabsheet3.Caption=TBL_Kunden.RecordCount
TBL_Kunden.Append
TBL_Kunden.FieldByName(„Suchbegriff“).AsString=„Test“
TBL_Kunden.FieldByName(„Ort“).AsString=„Aber Hallo“
TBL_Kunden.Post
```

Erklärung :

s ist hier eine String-Variable

TBL_Kunden ist das Adresstabellenobjekt

Tabellenobjekte

Auf die Feldinhalte können Sie via FIELDBYNAME(„FELDNAME) zugreifen. Rückgabewert ist dann ein FELD-Objekt, auf dessen Inhalt via .AsString, .AsInteger, .AsFloat oder .Value Zugeriffen werden kann.

```
s=TBL_Kunden.FieldByName(„Suchbegriff“).AsString
```

Schreibt in die Variable „s“ den Inhalt des Feldes Suchbegriff vom aktuellen Adressdatensatz.

```
TBL_Kunden.FieldByName(„Suchbegriff“).AsString=„huhu“
```

Schreibt in das Adressfeld Suchbegriff den Text huhu

```
TBL_Kunden.Append
```

AFS-Maskeneditor

fügt einen leeren Datensatz in die Tabelle ein

TBL_Kunden.Post

speichert den Datensatz ab

TBL_Kunden.LoadAll

diese Option lädt alle Datensätze vom SQL-Server nach

TBL_Kunden.RecordCount

gibt die Anzahl der Datensätze der Tabelle zurück

TBL_Kunden.First

Springt zum ersten Datensatz

TBL_Kunden.Last

Springt zum letzten Datensatz

TBL_Kunden.Next

Springt zum nächsten Datensatz

TBL_Kunden.Prior

Springt zum vorigen Datensatz

TBL_Kunden.Locate(„FELDE“,„SUCHTEXT“) Rückgabewert 1 = gefunden

0=nicht gefunden

Sucht einen Datensatz aus der Tabelle heraus

Beispiel:

```
If TBL_Kunden.Locate(„Firma“,„Schneider und Sohn“)=1 then
```

```
  ShowMessage(„ gefunden“)
```

```
Else
```

```
  ShowMessage(„ nicht gefunden“)
```

```
END IF
```

TBL_Kunden.EOF

Gibt an, ob es der letzte Datensatz ist 1=letzter 0=es kommen noch Datensätze

Grundeigenschaft von sichtbaren Objekten

OBJEKTNAME.Caption
Beschriftung (String)

OBJEKTNAME.Hint
Tooltip Beschriftung (String)

OBJEKTNAME.TOP
Position des Objekts von oben in Pixel (Integer)

OBJEKTNAME.LEFT
Position des Objekts von links in Pixel (Integer)

OBJEKTNAME.WIDTH
Breite des Objekts in Pixel (Integer)

OBJEKTNAME.HEIGHT
Höhe des Objekts in Pixel (Integer)

OBJEKTNAME.VISIBLE
Sichtbar oder Nicht (Boolean True/False)

AFS-Basic Syntax:

Syntax-Übersicht

Die derzeitige Syntax unterstützt:

- sub .. end und function .. end Deklarationen
- byref und dim Richtlinien
- if .. then .. else .. end Kontrollstrukturen
- for .. to .. step .. next Kontrollstrukturen
- do .. while .. loop und do .. loop .. while Kontrollstrukturen
- do .. until .. loop und do .. loop .. until Kontrollstrukturen
- ^ , * , / , and , + , - , or , <> , >= , <= , = , > , < , div , mod , xor , shl , shr Operatoren
- try .. except und try .. finally Blöcke
- select case .. end select Kontrollstrukturen
- array Kontrollstrukturen (x:=[1, 2, 3];)
- exit Befehl
- Zugriffe auf Objekteigenschaften und –methoden (ObjectName.SubObject.Property)

Die Script Strukturen

Die Script Strukturen bestehen aus zwei wichtigen Blöcken:

- a) Funktion und Unterdeklarierungen und

AFS-Maskeneditor

b) Hauptblock.

Beide sind optional, aber es sollte zumindest einer vorhanden sein.
Hier einige Beispiele:

```
SCRIPT 1:  
SUB DoSomething  
    CallSomething  
END SUB
```

CallSomethingElse

```
SCRIPT 2:  
CallSomethingElse
```

```
SCRIPT 3:  
  
FUNCTION MyFunction  
    MyFunction = „Ok!“  
END FUNCTION
```

Wie im normalen Basic können Anweisungen in einer einzigen Zeile durch „:“ getrennt werden.

Die Bezeichner

Die Bezeichnungsnamen im Script (Variablen, Namen, Funktionen und Prozedurnamen, usw.) folgen den meisten herkömmlichen Regeln von Basic: sie sollten mit einem Zeichen (a...z oder A...Z) oder „_“ beginnen und können von alphanumerischen Zeichen oder „_“ gefolgt werden. Sie können keine anderen Zeichen oder Leerzeichen beinhalten.

Gültige Bezeichner:

```
VarName  
_Some  
V1A2
```

Ungültige Bezeichner:

```
2Var  
My Name  
Some-more  
This,is,not,valid
```

Zugewiesene Anweisungen

Zugewiesene Anweisungen (weisen Sie einen Ausdruck, Ergebnis in eine Variable oder Objekteigenschaft ein) werden mit dem Benutzen von „=“ aufgebaut.

```
MyVar = 2
Button.Caption = „This „ + „is ok.“
```

Zeichen String

Strings (Zeichenkette) werden in Basic durch das doppelte Hochkomma („) deklariert.

Einige Beispiele

```
A = „This is a text“
Str = „Text „ + „concat“
```

Kommentare

Kommentare können in das Skript eingefügt werden. Man kann ´ (Hochkommas) oder REM nutzen. Kommentare werden am Ende einer Zeile eingefügt.

Hier einige Beispiele:

```
, Dies ist ein Kommentar vor ShowMessage
ShowMessage(„Ok“)
REM Dies ist noch ein Kommentar
ShowMessage(„More ok!“)
```

```
' Und dies ist ein Kommentar
, mit 2 Zeilen
ShowMessage(„End of okays“)
```

Variablen

Es nicht nötig, die Variablentypen im Skript zu deklarieren, also kann man die Variable deklarieren, indem man DIM und den Name eingibt. Es ist nicht nötig die Variablen zu deklarieren, wenn die Eigenschaft OptionExplicit auf false gesetzt ist. In diesem Fall sind die Variablen bereits deklariert. Wenn man mehr Kontrolle über sein Skript haben will, muss man OptionExplicit auf True stellen. Es wird ein Fehler auftreten, wenn die Variable benutzt wird, aber vorher nicht deklariert wurde.

Beispiele:

```
SCRIPT 1:
SUB Msg
  DIM S
  S = „Hello world!“
  ShowMessage(S)
END SUB
```

```
SCRIPT 2:
DIM A
A = 0
A = A+1
ShowMessage(A)
```

In diesem Skript wurde OptionExplicit auf false gesetzt, dann ist die Variablendeklaration in anderen Teilen des Skripts nicht erforderlich.

Index

Strings und Arrays können mit Hilfe von Eckigen-Klammern zugegriffen werden. Zum Beispiel, wenn „Str“ eine String-Variable ist, der Ausdruck Str[3] liefert das Dritte Zeichen. Auch Str[k + 1] liefert das k + 1 Zeichen, das heißt, wenn k = 3 ist, würde da 3 + 1 = 4, also das 4te Zeichen zurückgegeben.

```
MyChar = MyStr[2]
MyStr[1] = „A“
MyArray[1,2] = 1530
Lines.Strings[2] = „Some text“
```

Arrays

Das Script unterstützt Array-Konstrukte und variable Arrays. Ein Array verwendet „[“ und „]“ Zeichen. Sie können Multi-Index-Arrays erstellen, welche Array Konstrukte beinhalten. Sie können dann auf Arrays zugreifen, wenn Sie das Multi-Index Array trennen möchten, verwenden Sie „.“ als Trennzeichen. Wenn die Variable ein variables Array ist, so werden die Daten automatisch Indexe in diese Variable eingefügt. Eine Variable ist ein variables Array, wenn es mit einem Array-Konstrukteur erstellt wurde; ist es eine direkte Verweisung auf eine Delphi Variable, die ein variables Arrays ist (siehe Delphi Integration später), oder wenn es durch VarArrayCreate erstellt wurde. Arrays in den Skripten sind 0-basierte Indexe.

Hier ein Beispiel:

```
NewArray = [ 2,4,6,8 ]
Num = NewArray[1] //Num erhält „4“
MultiArray = [ [„green“,„red“,„blue“] , [„apple“,„orange“,„lemon“] ]
Str = MultiArray[0,2] //Str erhält ‚blue‘
MultiArray[1,1] = „new orange“
```

If - Anweisungen

Es gib zwei Formen der If – Anweisung: If...then...end und die If...then...else...end if. Wenn der If Ausdruck wahr ist, so wird die Anweisung ausgeführt. Wenn else in der IF – Anweisung vorhanden ist und der Ausdruck auf false steht wird direkt die Anweisung die nach dem else steht ausgeführt.

Beispiel:

```
IF J <> 0 THEN Result = I/J END IF
IF J = 0 THEN Exit ELSE Result := I/J END IF
IF J <> 0 THEN
  Result = I/J
```



```

    Count = Count + 1
ELSE
    Done = True
END IF

```

While Anweisung

Eine while – Anweisung ist dafür da, eine Funktion mehrmals zu wiederholen, wenn die Kontrollbedingung auf true umgestellt wurde. Die Kontrollbedingung wird vor der Anweisung durchgeführt, deshalb wird der Anweisungsablauf nicht durchgeführt, wenn die Kontrollbedingung auf false gestellt ist. Die while – Anweisung führt wiederholt mehrmals die Komponente der Anweisung aus. Solange der Ausdruck True zurück gibt, wiederholt sich der Ablauf:

Beispiel:

```

WHILE (Data[I] <> X) I = I + 1 END WHILE
WHILE (I > 0)
    IF Odd(I) THEN Z = Z * X END IF
    X = Sqr(X)
END WHILE

```

```

WHILE (not Eof(InputFile))
    Readln(InputFile, Line)
    Process(Line)
END WHILE

```

Loop Anweisung

Der Scripter unterstützt loop Anweisungen. Mögliche Schreibweisen sind:

```

DO WHILE expr statements LOOP
DO UNTIL expr statements LOOP
DO statements LOOP WHILE expr
DO statement LOOP UNTIL expr

```

Anweisungen führen WHILE expr als true aus, oder UNTIL extr ist true. Wenn expr vor der Anweisung steht, dann werden die Kontrollkonditionen getestet, bevor Sie wiederholt werden, ansonsten werden die Kontrollkonditionen nach der Wiederholung getestet.

Beispiel:

```

DO
    K = I mod J
    I = J
    J = K
LOOP UNTIL J = 0

```

```

DO UNTIL I >= 0
    Write(„Enter a value (0..9): „)
    Readln(I)
LOOP

```

```
AFS-Maskeneditor
DO
  K = I mod J
  I = J
  J = K
LOOP WHILE J <> 0
```

```
DO WHILE I < 0
  Write(„Enter a value (0..9): „)
  Readln(I)
LOOP
```

For Anweisung

Skripter unterstützt for Anweisungen mit folgender Syntax:

FOR counter = initialValue TO finalValue STEP stepValue statements NEXT
 For statement setzt den Zähler auf initialValue, wiederholt die Durchführung von von statement until „next“ und erhöht den Zähler mit stepValue, bis der Zähler finalValue erreicht hat. Dieser Schritt ist optional, wenn dieser ausgelassen wird, erhält der counter den Wert.

Beispiel:

```
SCRIPT 1:
FOR c = 1 TO 10 STEP 2
  a = a + c
NEXT
```

```
SCRIPT 2:
FOR I = a TO b
  j = i ^ 2
  sum = sum + j
NEXT
```

Select case Anweisung

Skripter unterstützt select- case Anweisungen mit der folgenden Syntax

```
:
SELECT CASE selectorExpression
CASE caseexpr1
statement1
...
CASE caseexprn
statementn
CASE ELSE
elsestatement
END SELECT
```

Wenn selectorExpression auf das Ergebnis des caseexprn Ausdrucks zu- trifft, wird die betreffende Anweisung abgearbeitet. Ansonsten wird die else-Anweisung ausgeführt. Else ist ein freigestellter Teil der case Anwei- sung.

Beispiel:

```
SELECT CASE uppercase(Fruit)
  CASE „lime“ ShowMessage(„green“)
  CASE „orange“
    ShowMessage(„orange“)
  CASE „apple“ ShowMessage(„red“)
CASE ELSE
  ShowMessage(„black“)
END SELECT
```

Funktionen und Sub Deklaration

Die Deklaration der Funktionen und der Prozeduren(Subs) sind dieselben wie in Basic. Um aus Funktionen die Funktionswerte zu erhalten, verwenden Sie implizit deklarierte Variablen, die denselben Namen wie die Funktionen haben. Parameter, welche eine Referenz darstellen, können ebenfalls verwendet werden, die durch BYREF verwenden werden. Einige Beispiele:

```
SUB HelloWorld
  ShowMessage(„Hello world!“)
END SUB
```

```
SUB UpcaseMessage(Msg)
  ShowMessage(Uppercase(Msg))
END SUB
```

```
FUNCTION TodayAsString
  TodayAsString = DateToStr(Date)
END FUNCTION
```

```
FUNCTION Max(A,B)
  IF A>B THEN
    MAX = A
  ELSE
    MAX = B
  END IF
END FUNCTION
```

```
SUB SwapValues(BYREF A, B)
  DIM TEMP
  TEMP = A
  A = B
  B = TEMP
END SUB
```

Einbinden von DLL und deren Funktionen:

```
function lib ‚libName.dll‘ [alias ExternalFunctionName] [callingConvention]
functionName(arguments) as resultType;
```

Zum Beispiel die folgende Deklaration:

AFS-Maskeneditor

```
function lib „CustomLib.dll“ MyFunction(arg as integer) as integer
```

importiert eine Funktion mit dem Namen MyFunction von CustomLib.dll.Default

Der Skriptler erlaubt auch verschiedenartige Aufrufe zu deklarieren (stdcall, register, pascal, cdecl or safecall) und verschiedene Namen für DDL Funktionen wie die folgende Deklaration:

```
function MessageBox lib „User32.dll“ alias „MessageBoxA“ stdcall (hwnd as pointer, text as string, caption as string, msgtype as integer) as integer
```

das Importiert 'MessageBoxA' function from User32.dll (Windows API library), namen 'MessageBox' um es im Skript zu nutzen.

Obenstehende Deklaration kann für Funktionen und Unterprogramme(Subs) benutzt werden (Routinen ohne Ergebniswerte)

Funktionsübersicht:

Abs

```
funktion Abs(X:double):double
```

Beschreibung

Abs gibt den absoluten Wert des Arguments X zurück.
X ist eine Zahl.

Beispiel:

```
ShowMessage(Abs(-2.3)) , gibt 2.2 aus
```

AnsiCompareStr

```
function AnsiCompareStr( S1, S2: string): Integer
```

Beschreibung

AnsiCompareStr vergleicht die beiden Strings S1 und S2 unter Beachtung der Groß-/Kleinschreibung. Die Vergleichsoperation wird vom aktuellen Windows-Gebietsschema gesteuert. Folgende Rückgabewerte sind möglich:

Bedingung	Rückgabewert
$S1 > S2$	> 0
$S1 < S2$	< 0
$S1 = S2$	$= 0$

Hinweis

Für die meisten Windows-Gebietsschemas sind Kleinbuchstaben kleiner als die entsprechenden Großbuchstaben. Dies widerspricht der ASCII-Sortierfolge, in der Kleinbuchstaben einen größeren Wert haben. Somit gibt der Vergleich

```
AnsiCompareStr(„a“,„A“)
```

einen Wert kleiner Null zurück, während `CompareStr(„a“,„A“)` einen Wert größer Null liefert.

AnsiCompareText

function `AnsiCompareText(S1, S2: string): Integer`

Beschreibung

`AnsiCompareStr` vergleicht die beiden Strings `S1` und `S2`, ohne dabei zwischen Groß- und Kleinschreibung zu unterscheiden. Die Vergleichsoperation wird vom aktuellen Gebietsschema gesteuert. Der Rückgabewert ist mit dem von `CompareStr` identisch.

AnsiLowerCase

function `AnsiLowerCase(S: string): string`

Beschreibung

`AnsiLowerCase` gibt einen String zurück, der eine in Kleinbuchstaben konvertierte Kopie des angegebenen String-Wertes ist. Bei der Konvertierung wird das aktuelle Gebietsschema berücksichtigt. Diese Funktion unterstützt Multibyte-Zeichensätze (MBCS).

AnsiUpperCase

function `AnsiUpperCase(S: string): string`

Beschreibung

`AnsiUpperCase` gibt eine Kopie des in `S` angegebenen Strings in einem groß geschriebenen String zurück. Bei der Konvertierung wird das aktuelle Windows-Gebietsschema berücksichtigt.

Hinweis

Diese Funktion unterstützt Multibyte-Zeichensätze (MBC)

ArcTan

function `ArcTan(X: double): double`

Beschreibung

Die anderen trigonometrischen Operationen werden mit Hilfe von `Sin`, `Cos` und `ArcTan` in den folgenden Ausdrücken berechnet:

$$\text{Tan}(x) = \text{Sin}(x) / \text{Cos}(x)$$

$$\text{ArcSin}(x) = \text{ArcTan}(x/\sqrt{1-\text{sqr}(x)})$$

$$\text{ArcCos}(x) = \text{ArcTan}(\sqrt{1-\text{sqr}(x)}) /$$

$$R = \text{ArcTan}(3.14)$$

Assigned

function `Assigned(var P): Boolean`

AFS-Maskeneditor

Beschreibung

Mit Assigned können Sie prüfen, ob der in P übergebene Zeiger bzw. Prozedur nil ist. P muß eine Variablenreferenz eines Zeigers oder prozeduralen Typs sein. Der Aufruf Assigned(P) entspricht bei einer Zeigervariablen dem Test P <> nil und bei einer Prozedurvariablen dem Test @P <> nil.

Assigned gibt False zurück, wenn P den Wert nil hat, andernfalls True.

Hinweis

Assigned kann keine Zeiger erkennen, die zwar nicht nil sind, aber trotzdem nicht mehr auf gültige Daten verweisen.

AssignFile

procedure AssignFile(var F FileName: string)

Beschreibung

Mit AssignFile können Sie eine Dateivariablen initialisieren. F ist eine Dateivariablen eines beliebigen Dateityps. FileName enthält einen Ausdruck des Typs String bzw. Pchar (wenn die erweiterte Syntax aktiviert ist).

Nach dem Aufruf von AssignFile ist F der externen Datei zugeordnet, bis F wieder geschlossen wird.

Enthält der Parameter FileName keinen Wert, wird F die Standardeingabe oder Standardausgabe zugewiesen. Wenn Sie F ein leeren String zuweisen, verweist F nach dem Aufruf Reset (F) auf die Standardeingabe und nach dem Aufruf Rewrite (F) auf die Standardausgabe.

Verwenden Sie AssignFile auf keinen Fall für eine bereits geöffnete Dateivariablen.

Beispiel: öffnet die Datei C:\Text.txt und zeigt die erste Zeile an

```
DIM F
DIM inhalt
s=„C:\test\test.txt“
if InputQuery(„Dateinamen“,„Dateinamen eingeben“, s) then
    AssignFile(F, s)
    Reset(F)
    inhalt=Readln(F)
    ShowMessage(inhalt)
    CloseFile(F)
end if
```

Beep

procedure Beep

Beschreibung

Beep ruft die Windows-API-Funktion MessageBeep auf.

Chdir

procedure ChDir(S: string)

Beschreibung

Mit ChDir kann das aktuelle Verzeichnis in den mit S angegebenen Pfad geändert werden. Gegebenenfalls wird dabei auch das aktuelle Laufwerk gewechselt.

Beispiel:

```
ChDir(„C:\test“)
```

Chr

function Chr(X: Byte): Char

Beschreibung

Chr gibt das Zeichen zurück, das dem im Byte-Wert X angegebenen Ordinalwert (ASCII-Wert) entspricht.

ShowMessage(Chr(„A“)) ,gibt 65 aus

CloseFile

procedure CloseFile(var F)

Beschreibung

CloseFile die Prozedur CloseFile, entfernt die Zuordnung zwischen einer Dateivariablen und einer externen Datei.

F ist eine Dateivariablen eines beliebigen Typs, die zuvor mit Reset, Rewrite oder Append geöffnet wurde. Die F zugeordnete externe Datei wird vollständig aktualisiert und anschließend geschlossen, wodurch das Dateihandle für die erneute Verwendung freigegeben wird.

CompareStr

function CompareStr(S1, S2: string): Integer

Beschreibung

CompareStr vergleicht S1 mit S2 unter Berücksichtigung der Groß-/Kleinschreibung. Der Rückgabewert ist kleiner als 0, wenn S1 kleiner als S2 ist. Der Rückgabewert 0 bedeutet, daß S1 gleich S2 ist, und bei einem Rückgabewert größer als 0 ist S1 größer als S2. Der Vergleich basiert auf dem 8-Bit-Ordinalwert der einzelnen Zeichen und wird vom aktuellen Windows-Gebietsschema nicht beeinflusst.

Beispiel:

```
String1 = „STEVE“  
String2 = „Steve“
```

AFS-Maskeneditor

```
I = CompareStr(String1, String2) „ Der Wert von I ist < 0  
if I <> 0 then ShowMessage(„Die Strings sind nicht identisch“)
```

CompareText

function CompareText(S1, S2: string): Integer

Beschreibung

CompareText vergleicht S1 mit S2 und gibt 0 zurück, wenn beide identisch sind. Ist S1 größer als S2, wird ein Integer-Wert größer als 0 zurückgegeben. Ist S1 kleiner als S2, ist der Rückgabewert kleiner als 0. Beim Vergleich wird nicht zwischen Groß- und Kleinschreibung unterschieden und die aktuelle Windows-Ländereinstellung nicht berücksichtigt.

Zum Beispiel gibt CompareText an, daß der String

ABC

kleiner als

aaa

ist, weil A in der ASCII-Reihenfolge kleiner als a ist. Dies steht im Gegensatz zu einem Vergleich ohne Berücksichtigung der Groß-/Kleinschreibung. In diesem Fall würde durch den Buchstaben B oder durch einen Vergleich, der auf der Ländereinstellung (in die meisten Sprachtreibern haben Großbuchstaben höhere Werte als Kleinbuchstaben) beruht, der erste String als der größer ausgewiesen werden.

Copy

function Copy(S Index, Count: Integer): string

Beschreibung

S ist ein Ausdruck des Typs String. Index und Count sind Integer-Ausdrücke. Copy gibt einen Substring oder ein Array zurück, das Count Zeichen oder Elemente ab S[Index] enthält.

Ist Index größer als die Länge von S, gibt Copy einen leeren String oder ein leeres Array zurück.

Gibt Count mehr Zeichen oder Array-Elemente an, als verfügbar sind, werden nur die Zeichen oder Elemente von S[Index] bis zum Ende von S zurückgegeben.

ShowMessage(Copy(„Aber Hallo“,1,4)) , gibt Aber aus

Cos

function Cos(X: double): double

Beschreibung

Cos gibt den Cosinus des Winkels X im Bogenmaß zurück.

Beispiel

R = Cos(3.14)

CreateOleObject

function CreateOleObject(ClassName: string): OleObjekt

Beschreibung

CreateOleObject erzeugt ein einzelnes, nicht initialisiertes Objekt der mit dem Parameter ClassName angegebenen Klasse. ClassName enthält den der Klassen-ID (CLSID) entsprechenden String. Mit CreateOleObject läßt sich ein Objekt des angegebenen Typs erstellen, wenn die CLSID bekannt ist und sich das Objekt auf einem lokalen oder In-Process-Server befindet. Mit CreateOleObject werden nur Objekte erzeugt, die nicht Bestandteil eines Aggregats sind.

CreateOleObject wird einmal für jede neue Instanz einer Klasse aufgerufen. Um mehrere Instanzen einer Klasse zu erzeugen, verwenden Sie einen Klassengenerator.

CreateOleObject gibt eine Referenz auf die Schnittstelle zurück, die für die Kommunikation mit dem Objekt verwendet wird. Bei CreateOleObject ist diese Schnittstelle vom Typ IDispatch. Um ein COM-Objekt zu erstellen, das keine IDispatch-Schnittstelle unterstützt, verwenden Sie CreateComObject.

Siehe Anleitung zu OLE-Objekten

Date

function Date: TDateTime

Beschreibung

Mit Date können Sie das aktuelle Datum als TDateTime-Wert abrufen.
Beispiel:

```
Label1.Caption = „Heute ist der „ + DateToStr(Date)
```

DateTimeToStr

function DateTimeToStr(DateTime: TDateTime): string

Beschreibung

DateTimeToString konvertiert den TDateTime-Wert im Parameter DateTime im durch die globalen Variablen ShortDateFormat und LongTimeFormat festgelegten Format. Hat der Nachkommaanteil des Parameters DateTime den Wert Null, wird die Zeit nicht angezeigt.

Um die String-Formatierung zu modifizieren, ändern Sie die globalen Datums- und Uhrzeit-Variablen ShortDateFormat und LongTimeFormat.

```
Label1.Caption = DateTimeToStr(Now)
```

DateToStr

function DateTimeToStr(DateTime: TDateTimeTDateTime): string

Beschreibung

Mit DateToStr erhalten Sie die String-Entsprechung eines Datumswertes für Anzeigewecke. Der Konvertierung liegt das Format zugrunde, das die globale Variable ShortDateFormat bezeichnet.

Beispiel:

```
Label1.Caption = „Heute ist der „ + DateToStr(Date)
```

DayOfWeek

function DayOfWeek(Date: TDateTime): Integer

Beschreibung

DayOfWeek gibt den Wochentag des angegebenen Datums als Integer zwischen 1 und 7 zurück. Hierbei gilt der Sonntag als erster Tag der Woche und der Samstag als der siebte Tag.

Beispiel:

```
Days=[„“,„Sonntag“,„Montag“,„Dienstag“,„Mittwoch“,„Donnerstag“,„Freitag“,„Samstag“]  
ADate = StrToDate(„15.01.2050“)  
ShowMessage(„15.01.2050 ist ein „ + days[DayOfWeek(ADate)])
```

Dec

procedure Dec(var X[N: integer])

Beschreibung

Dec subtrahiert von der angegebenen Variablen den Wert Eins oder N.

X ist eine Variable vom Typ Ordinal oder ein Zeigertyp, falls die erweiterte Syntax aktiviert ist.

N ist ein Integer-Ausdruck.

X dekrementiert um 1, oder um N, wenn N angegeben wurde d.h., daß Dec(X) der Anweisung $X = X - 1$ und Dec(X, N) der Anweisung $X = X - N$ entspricht. Dec erzeugt jedoch hochoptimierten Maschinencode und ist besonders für enge Schleifen geeignet.

```
P=10
```

```
Dec(P)
```

```
ShowMessage(p) , gibt 9 aus
```

DecodeDate

procedure DecodeDate(Date: TDateTime var Year, Month, Day: integer)

Beschreibung

DecodeDate zerlegt den im Parameter Date übergebenen Wert in die Werte Year, Month und Day. Ist TDateTime kleiner oder gleich null, werden die var-Parameter auf null gesetzt.

Beispiel:

```
DIM Year,Month,Day
Present= Now
DecodeDate(Present, Year, Month, Day)
Label1.Caption = „Heute ist Tag „ + Day + „ des Monats „ + Month + „
im Jahr „ + Year
```

DecodeTime

```
procedure DecodeTime(Time: TDateTime var Hour, Min, Sec, MSec: integer)
```

Beschreibung

DecodeTime zerlegt das im Parameter Time übergebene Objekt in Stunden, Minuten, Sekunden und Millisekunden.

DeleteFile

```
procedure DeleteFile(F: string)
```

Beschreibung

Löscht die angegebene Datei

Beispiel:

```
DeleteFile(„C:\TEST\TEST.TXT“)
```

DIM

DIM definiert Variable vor dem Gebrauch im Speicher

z.B.

```
DIM A,B,C
```

```
DIM Z
```

Intern werden alle Variablen als VARIANT behandelt und haben dadurch keinen Datentype im eigentlichen Sinne! DIM muss auch nur angewendet werden, wenn mit der Variablen eine sogenannte VAR Funktion aufgerufen wird, die eine veränderbare Variable benötigt (z.B. EncodeDate)

EncodeDate

```
function EncodeDate(Year, Month, Day: Word): TDateTime
```

AFS-Maskeneditor

Beschreibung

EncodeDate gibt einen Wert des Typs TDateTime für die als Parameter angegebenen Werte Year, Month und Day zurück.

Das Jahr muß zwischen 1 und 9999 liegen.

Die gültigen Monatswerte liegen zwischen 1 und 12.

Als Tageswert können die Werte 1 bis 28, 29, 30 oder 31 (je nach Monatswert) angegeben werden. So liegen beispielsweise die möglichen Tageswerte für den Monat 2 (Februar) zwischen 1 und 28 bzw. zwischen 1 und 29, abhängig davon, ob als Parameter Year ein Schaltjahr übergeben wird.

EncodeTime

function EncodeTime(Hour, Min, Sec, MSec: Word): TDateTime

Beschreibung

EncodeTime codiert die angegebenen Werte für Stunde, Minute, Sekunde und Millisekunde in einen Wert des Typs TDateTime. Für die Stunde (Hour) muß ein Wert zwischen 0 und 23, für die Minute (Min) ein Wert zwischen 0 und 59, für die Sekunde (Sec) ebenfalls ein Wert zwischen 0 und 59 und für die Millisekunde (MSec) ein Wert zwischen 0 und 999 angegeben werden. Befindet sich einer der angegebenen Werte nicht innerhalb des genannten Bereichs, wird eine EConvertError-Exception ausgelöst.

Der Rückgabewert ist eine Zahl zwischen 0 (inklusive) und 1 (exklusiv), die die angegebene Zeit in Nachkommastellen ausdrückt. Der Wert 0 entspricht Mitternacht, 0.5 ist Mittag, 0.75 ist 18:00 abends usw.

EOF

function Eof(var F): Boolean

Für Textdateien:

Beschreibung

Eof prüft, ob die aktuelle Dateiposition mit dem Dateiende identisch ist. F ist eine Dateivariablen. Fehlt dieser Parameter, wird die Standardvariable Input verwendet.

Eof(F) gibt True zurück, wenn sich die aktuelle Dateiposition hinter dem letzten Zeichen der Datei befindet oder wenn die Datei leer ist.

Andernfalls wird False zurückgegeben.

Beispiel:

```
DIM F1,F2
DIM inhalt
s=„C:\test\test.txt“
s2=„C:\test\copy.txt“
if InputQuery(„Dateinamen“,„Dateinamen eingaben“, s) then
    AssignFile(F1, s)
    AssignFile(F2, s2)
```

```

Reset(F1)
Rewrite(F2)
Do while EOF(F1)=False
  inhalt=Readln(F1)
  WriteLN(F2,inhalt)
LOOP
CloseFile(F1)
CloseFile(F2)
ShowMessage(„Datei kopiert“)
end if

```

Exp

```
function Exp(X: double): double
```

Beschreibung

Exp gibt e hoch X zurück, wobei e die Basis des natürlichen Logarithmus ist.

FileExists

```
function FileExists(F:string): Integer
```

Beschreibung

Mit FileExists können Sie abfragen, ob eine Datei existiert (1) oder nicht (0).

Beispiel:

```

If FileExists(„C:\Export\test.txt“)=1 then
  ShowMessage(„Datei ist vorhanden“)
ELSE
  ShowMessage(„Datei nicht ist vorhanden“)
END IF

```

FilePos

```
function FilePos(var F): Integer
```

Beschreibung

Mit FilePos können Sie die aktuelle Position in der angegebenen geöffneten Datei ermitteln. Befindet sich der Dateizeiger am Anfang der Datei, wird 0 zurückgegeben. Ansonsten gibt FilePos den Byte-Offset der aktuellen Position vom Dateianfang zurück.

Diese Funktion kann nicht für geschlossene oder Textdateien verwendet werden.

Beispiel:

```

DIM F1,F2
DIM inhalt
s=„C:\test\test.txt“

```

```

AFS-Maskeneditor
s2=„C:\test\copy.txt“
if InputQuery(„Dateinamen“,„Dateinamen eingaben“, s) then
  AssignFile(F1, s)
  AssignFile(F2, s2)
  Reset(F1)
  Rewrite(F2)
  Do while EOF(F1)=False
    inhalt=Readln(F1)
    Label1.Caption=100*FilePos(F1)/FileSize(F1)+“% kopiert“
    WriteLN(F2,inhalt)
  LOOP
  CloseFile(F1)
  CloseFile(F2)
  ShowMessage(„Datei kopiert“)
end if

```

FileSize

function FileSize(var F): Integer

Beschreibung

Mit FileSize können Sie die Größe der angegebenen Datei (F) ermitteln. Die Funktion kann nur für geöffnete Dateien verwendet werden. Wenn die Datei leer ist, gibt FileSize(F) 0 zurück.

Hinweis

FileSize kann nicht für Textdateien verwendet werden.

Beispiel:

Siehe FilePos

FileShow

Procedure FileShow(File,Param:string)

Beschreibung

Führt eine Datei aus und startet diese. Mit File wird der Dateiname angegeben und Param ist der Ausrufparameter, falls benötigt!

Beispiel:

```
FileShow(„C:\test.pdf“, „“)
```

Zeigt die pdf Datei an

FloatToStr

function FloatToStr(Value: double): string

Beschreibung

FloatToStr konvertiert die mit Value angegebene Gleitkommazahl in die entsprechende String-Darstellung. Bei der Operation wird das allgemeine Zahlenformat mit 15 signifikanten Stellen verwendet.

Wenn Sie die Formatierung des Strings genauer steuern wollen, verwenden Sie die Funktion `FloatToStrF`.

Hinweis:

Die Basicscriptsprache macht solche Typenumwandlungen selbstständig. Diese Funktion ist aber sinnvoll, wenn explizit Daten eines bestimmten Typs übergeben werden sollen (z.B. an Excel, da sonst Excel selbst versucht zu erkennen, um welchen Datentype es sich handelt, und dies oftmals nicht gelingt!)

Format

function `Format`(Format: string Args: array of const): string

Beschreibung

`Format` formatiert die Argumente in dem mit `Args` angegebenen offenen Array. Die Formatierung wird durch den Format-String `Format` gesteuert. Der Rückgabewert der Funktion ist ein String..

Informationen über Format-Strings finden Sie unter `Format-Strings`.

Beispiel:

```
StatusBar1.SimpleText = Format(„In der Tabelle befinden sich nun %d
Datensätze.“,
[TBL_Kunden.RecordCount])
```

FormatDateTime

function `FormatDateTime`(Format: string DateTime: TDateTime): string

Beschreibung

`FormatDateTime` formatiert einen Wert des Typs `TDateTime` (`DateTime`). Dabei wird das mit `Format` angegebene Format verwendet. Die folgende Tabelle enthält die unterstützten Formatbezeichner

Beispiel:

```
S = FormatDateTime(„Die Besprechung ist am „ dddd, mmmm d, yyyy, „ um
„ hh:mm AM/PM“,
StrToDateTime(TBL_Kunden.FieldByName(„Wiedervorlage“).AsString))
```

FormatFloat

function `FormatFloat`(Format: string Value: double): string

Beschreibung

`FormatFloat` formatiert den mit `Value` angegebenen Gleitkommawert. Dazu wird der im Parameter `Format` übergebene Format-String verwendet. Die folgende Tabelle enthält die unterstützten Formatbezeichner

Beispiel:

```
Format-String 1234    -1234    0,5 0
```

AFS-Maskeneditor

```
—          1234    -1234    0,5  0
0          1234    -1234    1    0
0.00       1234.00 -1234.00 0,500,00
#.#.#      1234    -1234    ,5
#,###0.00  1,234.00 -1.234,00  0,500,00
#,###0.00(#,###0.00) 1,234.00 (1.234,00)  0,500,00
#,###0.00Zero 1,234.00 -1.234,00  0,50Zero
0.000E+00   1.234E+03  -1,234E+03  5,000E-01  0,000E+00
#.###E-0    1.234E3  -1,234E3 5E-10E0
```

Frac

function Frac(X: double): double

Beschreibung

Frac gibt den Nachkommaanteil der im Parameter X übergebenen reellen Zahl zurück.

X ist ein Ausdruck eines Real-Typs. Der Rückgabewert ist der Nachkommaanteil von X, d.h. $\text{Frac}(X) = X - \text{Int}(X)$.

Beispiel:

```
R = Frac(123.456) „ 0.456
R = Frac(-123.456) „ -0.456
```

GetCurrentDir

function GetCurrentDir: string

Gibt das aktuelle Verzeichnis zurück

GetTempDir

function GetTempDir: string

Gibt das Tempverzeichnis des Rechners zurück

GetAppDir

function GetAppDir: string

Gibt das Verzeichnis des Programms zurück

GetActiveOleObject

function GetActiveOleObject(ClassName: string): IDispatch

Beschreibung

GetActiveOleObject gibt eine Referenz auf die IDispatch-Schnittstelle des aktiven Objekts zurück. Die Klasse des Objekts wird im Parameter ClassName angegeben. Es handelt sich dabei um die Programm-ID der Klasse des Objekts, das in der Tabelle der ausgeführten OLE-Objekte aktiv ist. Mit GetActiveOleObject kann auf diese Tabelle zugegriffen werden.

Verwenden Sie die API-Funktion RegisterActiveObject, um ein Objekt in diese Tabelle einzufügen.

Wenn das Klassenobjekt oder die IDispatch-Schnittstelle nicht ermittelt werden konnten, löst GetActiveOleObject eine EOleSysError-Exception aus. Siehe Arbeiten mit OLE-Objekten

High

function High(X)

Beschreibung

Mit High können Sie die Obergrenze eines Ordinal-, Array- oder String-Wertes abrufen. Der Typ des Rückgabewertes ist X oder der Indextyp von X.

X ist entweder ein Typbezeichner oder eine Variablenreferenz und muß einer der folgenden Typen sein

Inc

procedure Inc(var X [N: Integer])

Beschreibung

Inc addiert 1 oder N zu der mit X angegebenen Variable.

X ist eine Variable mit einem Typ Ordinal (einschließlich Integer) ein Zeigertyp, falls die erweiterte Syntax aktiviert ist.

N ist ein Integer-Ausdruck.

X wird um 1 inkrementiert oder um N, wenn N angegeben ist. Das bedeutet, Inc(X) entspricht der Anweisung $X = X + 1$, und Inc(X, N) ist mit der Anweisung $X = X + N$ identisch. Inc generiert jedoch hochoptimierten Maschinencode und ist besonders für Schleifen geeignet.

Inc(IntVar) „ IntVar = IntVar + 1

Inc(IntegerVar, 5) „ IntegerVar = IntegerVar + 5

IncMonth

function IncMonth(Date: TDateTime NumberOfMonths: Integer): TDateTime

Beschreibung

IncMonth liefert den Wert des Parameters Date zurück, der um die Anzahl der in NumberOfMonths angegebenen Monate erhöht wurde. Wenn NumberOfMonths einen negativen Wert enthält, wird ein Datum zurückgeliefert, das um die angegebene Anzahl von Monaten zurückliegt.

Wenn der angegebene Monatstag größer ist als der letzte Tag des resultierenden Monats, wird der Tag auf den letzten Tag des Monats gesetzt. Die Uhrzeit des im Parameter Date angegebenen Tages wird in das Ergebnis kopiert.

InputQuery

function InputQuery(ACaption, APrompt: string var Value: string): Boolean

Beschreibung

Mit InputQuery können Sie ein Dialogfeld anzeigen, in dessen Eingabefeld der Benutzer einen String eingeben kann. Der Parameter ACaption enthält den Titel des Dialogfeldes. Der Parameter APrompt ist der Text, der den Benutzer zur Eingabe auffordert. Der Parameter Value gibt den String an, der beim erstmaligen Öffnen des Dialogfeldes angezeigt wird.

Gibt der Benutzer einen String in das Eingabefeld ein und klickt auf OK, wird dem Parameter Value der neue Wert zugewiesen.

InputQuery gibt True zurück, wenn der Benutzer auf OK klickt, und False, wenn der Benutzer auf Abbrechen klickt oder die ESC-Taste drückt.

Wenn beim Verlassen des Dialogfeldes ein Standardwert verwendet werden soll, verwenden Sie InputBox.

Beispiel:

```
if InputQuery(„Dateinamen“,„Dateinamen eingaben“, s) then
  ShowMessage(„es wurde „+s+“ eingegeben“)
ELSE
  ShowMessage(„es wurde abbruch gedrückt!“)
END IF
```

Insert

procedure Insert(Source: string var S: string Index: Integer)

Beschreibung

Insert fügt Source in S an der Position S[Index] ein.

Source ist ein String-Ausdruck und S eine String-Variable beliebiger Länge. Der Integer-Ausdruck Index ist ein Zeichen-Index und kein Byte-Index.

Beispiel

```
S = „Honest Lincoln“
Insert(„Abe „, S, 8) 'Honest Abe Lincoln
```

Int

function Int(X: Double): Double

Beschreibung

Int gibt den ganzzahligen Anteil von X zurück, also X gegen 0 gerundet. X ist ein Ausdruck des Typs Double.

Beispiel:

```
begin
  R = Int(123.456) ' 123.0
  R = Int(-123.456) ' -123.0
end
```

IntToHex

function IntToHex(Value: Integer Digits: Integer): string

Beschreibung

IntToHex konvertiert den mit Value angegebenen Integer-Wert in einen String mit der hexadezimalen (Basis 16) Darstellung der Zahl. Mit dem Parameter Digits kann festgelegt werden, wieviele Hexadezimalstellen mindestens zurückgegeben werden.

Beispiel:

```
Label1.Caption = ""
for i = 1 to Length(Edit1.Text)
  Label1.Caption = Label1.Caption + IntToHex(Edit1.Text[i],4) + " "
end
```

IntToStr

function IntToStr(Value: Integer): string

Beschreibung

IntToStr konvertiert den mit Value angegebenen Integer-Wert in einen String mit der dezimalen Darstellung der Zahl.

Beispiel:

```
Label1.Caption = IntToStr(StrToInt(Edit1.Text) * StrToInt(Edit2.Text))
```

IsLeapYear

function IsLeapYear(Year: Word): Boolean

Beschreibung

Mit IsLeapYear können Sie feststellen, ob das im Parameter Year angegebene Jahr ein Schaltjahr ist. Year gibt das Kalenderjahr an.

Mit Hilfe von DecodeDate können Sie den Wert für den Parameter Year aus einem TDateTime-Wert ableiten.

IsValidIdent

function IsValidIdent(Ident: string): Boolean

Beschreibung

IsValidIdent gibt True zurück, wenn der mit Ident angegebene String ein gültiger Bezeichner ist. Ein Bezeichner muß mit einem Zeichen aus der Menge [„A“..„Z“, „a“..„z“, „_“] beginnen. Danach können null oder mehr

AFS-Maskeneditor

Zeichen aus der Menge [„A“..„Z“, „a“..„z“, „0“..„9“, „_“] folgen.

Length

function Length(S): Integer

Beschreibung

Length gibt die Anzahl der im angegebenen String vorhandenen Zeichen oder die Anzahl der Elemente im Array zurück.

S ist ein String oder ein Array.

Beispiel:

```
Label1.Caption = ""
for i = 1 to Length(Edit1.Text)
    Label1.Caption = Label1.Caption + IntToHex(Edit1.Text[i],4) + " "
end
```

Ln

function Ln(X: Double): Double

Beschreibung

Ln gibt den natürlichen Logarithmus ($\ln(e) = 1$) des mit X angegebenen Double-Ausdrucks zurück.

Low

function Low(X)

Mit Low können Sie den kleinsten Wert bzw. das erste Element eines Ordinal-, Array- oder String-Wertes abrufen. Der Typ des Rückgabewertes ist X oder der Indextyp von X. X ist entweder ein Typbezeichner oder eine Variablenreferenz.

LowerCase

function LowerCase(S: string): string

Beschreibung

LowerCase wandelt die Zeichen des mit S angegebenen String-Wertes in Kleinbuchstaben um. Die Konvertierung betrifft jedoch nur die mit 7 Bit darstellbaren ASCII-Zeichen zwischen „A“ und „Z“ (keine Umlaute). Sollen die landesspezifischen Zeichen des 8-Bit-Zeichensatzes konvertiert werden, verwenden Sie statt dessen die Funktion AnsiLowerCase.

Beispiel:

```
ShowMessage(LowerCase(„Aber Hallo“))
```

MakeDir

procedure MakeDir(F: string)

Beschreibung

Seite 172

Erstelle das angegebene Verzeichnis

Beispiel:

```
MakeDir(„C:\TEST“)
```

Now

```
function Now: TDateTime
```

Beschreibung

Now gibt das aktuelle Datum und die aktuelle Uhrzeit zurück. Dies entspricht Date + Time.

Beispiel:

```
Label1.Caption = DateTimeToStr(Now)
```

Odd

```
function Odd(X: Integer): Boolean
```

Beschreibung

Odd prüft, ob der mit X angegebene Wert eine ungerade Zahl ist. Bei einer ungeraden Zahl wird True zurückgegeben, bei einer geraden False.

Beispiel:

```
if Odd(5) then
  Canvas.TextOut(10, 10, „5 ist ungerade.“)
else
  Canvas.TextOut(10, 10, „Da stimmt was nicht!“)
end
```

Ord

```
function Ord(X): Integer
```

Beschreibung

X ist ein Ausdruck des Typs Ordinal oder Char. Der Rückgabewert hat den Datentyp Integer und enthält die ordinale Position von X.

Ord arbeitet nicht mit Integer-Werten.

Pos

```
function Pos(Substr: string S: string): Integer
```

Beschreibung

Pos sucht in dem String S nach dem Teil-String Substr. Die Parameter Substr und S sind String-Ausdrücke.

Wird der Teil-String gefunden, gibt Pos den Integer-Index des ersten Zeichens von Substr in S zurück. Die Groß/Kleinschreibung wird von Pos nicht

AFS-Maskeneditor

berücksichtigt. Ist Substr nicht vorhanden, wird der Wert Null zurückgegeben.

Beispiel:

```
S=„Hallo Herr Doktor“
```

```
ShowMessahe(POS(„Herr“,s))
```

Random

```
function Random [ ( Range: Integer ) ]
```

Beschreibung

Random gibt eine Zufallszahl im Bereich $0 \leq X < \text{Range}$ zurück.

Wird der Parameter Range nicht angegeben, liefert Random einen Double-Wert im Bereich

$0 \leq X < 1$.

Um den Zufallszahlengenerator zu initialisieren, rufen Sie Randomize einmal auf oder weisen der Variable RandSeed vor dem Aufruf von Random einen Wert zu.

ReadLn

```
function ReadLn(var F: Text): string
```

Beschreibung

ReadLn liest eine Textzeile aus der angegebenen Datei und setzt den Dateizeiger in die nächste Zeile.

```
DIM F1,F2
```

```
DIM inhalt
```

```
s=„C:\test\test.txt“
```

```
s2=„C:\test\copy.txt“
```

```
if InputQuery(„Dateinamen“,„Dateinamen eingaben“, s) then
```

```
    AssignFile(F1, s)
```

```
    AssignFile(F2, s2)
```

```
    Reset(F1)
```

```
    Rewrite(F2)
```

```
    Do while EOF(F1)=False
```

```
        inhalt=ReadLn(F1)
```

```
        Label1.Caption=100*FilePos(F1)/FileSize(F1)+„% kopiert“
```

```
        WriteLN(F2,inhalt)
```

```
    LOOP
```

```
    CloseFile(F1)
```

```
    CloseFile(F2)
```

```
    ShowMessage(„Datei kopiert“)
```

```
end if
```

Reset

```
procedure Reset(var F [: File; RecSize: Word ] );
```

Beschreibung

Reset öffnet die mit F angegebene externe Datei. Ist keine Datei mit diesem Namen vorhanden, tritt ein Fehler auf. Ist F bereits offen, wird sie zuerst geschlossen und dann erneut geöffnet. Nach dem Öffnen wird der Dateizeiger an den Anfang der Datei gesetzt.

Nach dem Aufruf von Reset hat Eof(F) den Wert True, wenn die Datei leer ist. Andernfalls ist Eof(F) False.

Beispiel

Importiert E-Mailadressen und einen Namen aus einer Textdatei (hier Stelle 1 -20 Name und 21 bis 70)

```
DIM F1,F2
s=„C:\test\test.txt“
AssignFile(F1, s)
Reset(F1)
Do while EOF(F1)=False
    inhalt=Readln(F1)
    mail=Trim(Copy(inhalt,21,49))
    if mail<>“” then
        TBL_Kunden.Append
        TBL_Kunden.FieldName(„Name“).AsString=Copy(s,1,20)
        TBL_Kunden.FieldName(„E-Mail“).AsString=mail
        TBL_Kunden.Post
    End if
LOOP
CloseFile(F1)
```

Rewrite

```
procedure Rewrite(var F: File [ Recsize: Word ] )
```

Beschreibung

Rewrite erstellt eine neue externe Datei unter dem F zugewiesenen Namen.

F ist eine Variable eines beliebigen Dateityps, die zuvor mit AssignFile einer externen Datei zugewiesen wurde. Der optionale Ausdruck RecSize kann nur bei nicht typisierten Dateien verwendet werden und gibt die Datenblockgröße für das Lesen und Schreiben der Datei an. Wird RecSize nicht angegeben, wird der Standardwert 128 verwendet.

Ist schon eine gleichnamige externe Datei vorhanden, wird sie gelöscht und an ihrer Stelle die neue Datei angelegt.

Ist F bereits offen, wird sie zuerst geschlossen und dann erneut erstellt. Nach dem Erstellen wird der Dateizeiger an den Anfang der leeren Datei gesetzt.

AFS-Maskeneditor

Bei einer Textdatei ist F nach dem Öffnen schreibgeschützt.

Nach dem Aufruf von Rewrite hat Eof(F) immer den Wert True.

Beispiel:

```
DIM F1
s=„C:\test\test.txt“
AssignFile(F1, s)
ReWrite(F1)
WriteLN(F1,TBL_Auftrag.FieldByName(„Sendungsnummer“).AsString)
CloseFile(F1)
```

Round

function Round(X: Double): Integer

Beschreibung

Round rundet einen Wert des Typ Double auf einen Integer-Wert.

X ist ein Ausdruck des Typs Double. Zurückgegeben wird ein Integer-Wert mit dem auf die nächste ganze Zahl gerundeten Wert von X. Liegt X genau in der Mitte zwischen zwei ganzen Zahlen, wird immer die gerade Zahl zurückgeliefert.

Liegt der gerundete Wert von X außerhalb des Integer-Wertebereichs, wird ein Laufzeitfehler ausgelöst, der mit einer EInvalidOp-Exception behandelt werden kann.

Beispiel:

begin

```
Str(1.4:2:1, T)
```

```
S = T + „ wird gerundet zu „ + IntToStr(Round(1.4))
```

```
Str(1.5:2:1, T)
```

```
S = S + T + „ wird gerundet zu „ + IntToStr(Round(1.5))
```

```
Str(-1.4:2:1, T)
```

```
S = S + T + „ wird gerundet zu „ + IntToStr(Round(-1.4))
```

```
Str(-1.5:2:1, T)
```

```
S = S + T + „ wird gerundet zu „ + IntToStr(Round(-1.5))
```

```
MessageDlg(S, mtInformation, [mbOk], 0)
```

end

Rmdir

procedure Rmdir(dir: string)

Beschreibung

Löscht das angegebene Verzeichnis (nur wenn keine Dateien mehr enthalten sind).

ShowMessage

procedure ShowMessage(Msg: string)

Beschreibung

ShowMessage zeigt ein einfaches Meldungsfenster mit der Schaltfläche OK an. Der Parameter Msg gibt die anzuzeigende Meldung an. Der Name der ausführbaren Datei wird als Titel des Fensters verwendet.

Soll in einem Dialogfeld eine Meldung zusammen mit anderen Schaltflächen oder mit einem Symbol angezeigt werden, verwenden Sie statt dessen MessageDlg.

Beispiel:

```
if InputQuery(„Dateinamen“,„Dateinamen eingaben“, s) then
  ShowMessage(„es wurde „+s+“ eingegeben“)
ELSE
  ShowMessage(„es wurde abbruch gedrückt!“)
END IF
```

Sin

function Sin(X: Double): Double

Beschreibung

Sin gibt den Sinus des mit X angegebenen Winkels im Bogenmaß (Rad) zurück.

X ist ein Ausdruck des Typs Double.

Beispiel:

```
R = Sin(Pi)
ShowMessage(Str(R:5:3, S))
```

Sqr

function Sqr(X: Double): Double

Beschreibung

Sqr berechnet das Quadrat des mit X angegebenen Wertes.

X ist ein Gleitkommaausdruck. Der Rückgabewert (das Quadrat) hat den selben Typ wie X und ist mit der Anweisung X*X identisch.

Sqrt

function Sqrt(X: Double): Double

Beschreibung

Sqr berechnet die Quadratwurzel des mit X angegebenen Gleitkommaausdrucks.

StrToDate

function StrToDate(S: string): TDateTime

Beschreibung

Mit StrToDate kann ein String mit einer Datumsangabe geparkt werden. Enthält S kein gültiges Datum, wird eine EConvertError-Exception ausgelöst.

Der String muß aus zwei oder drei Zahlen bestehen, die durch das in der globalen Variable DateSeparator definierte Zeichen voneinander getrennt sind. Die Reihenfolge von Tag, Monat und Jahr wird von der globalen Variable ShortDateFormat bestimmt (mögliche Kombinationen sind M/T/J, T/M/J und J/M/T).

Beispiel:

```
Days=[„“,„Sonntag“,„Montag“,„Dienstag“,„Mittwoch“,„Donnerstag“,„Freitag“,„Samstag“]
ADate = StrToDate(„15.01.2050“)
ShowMessage(„15.01.2050 ist ein „ + days[DayOfWeek(ADate)]
```

StrToDateTime

function StrToDateTime(S: string): TDateTime

Beschreibung

Mit StrToDateTime kann ein String mit einer Datums-/Zeitangabe geparkt werden. Enthält S kein gültiges Datum, wird eine EConvertError-Exception ausgelöst.

Der in S angegebene String muß das Format MM/TT/JJ HH:MM:SS haben. Die Angaben AM und PM sowie die Sekunden sind optional. Verwenden Sie das 24-Stunden-Format (7:45 PM entspricht dann 19:45), wenn AM oder PM nicht angegeben wird.

Hinweis:

Die Basicscriptsprache macht solche Typenumwandlungen selbstständig. Diese Funktion ist aber sinnvoll, wenn explizit Daten eines Bestimmten Typs übergeben werden sollen (z.B. an Excel, da sonst Excel selbst versucht zu erkennen, um welchen Datentype es sich handelt, und dies oftmals nicht gelingt!)

StrToFloat

function StrToFloat(S: string): Double

Beschreibung

Mit StrToFloat kann der String S in einen Gleitkommawert konvertiert werden. S muß ein optionales Vorzeichen (+ oder -), einen String mit Ziffern und einem optionalen Dezimaltrennzeichen und einer optionalen Mantisse. Die Mantisse setzt sich aus „E“ oder „e“, einem optionalen Vorzeichen und einer ganzen Zahl zusammen. Leerzeichen am Anfang und Ende werden

ignoriert.

Die globale Variable `DecimalSeparator` definiert das Zeichen, das als Dezimaltrennzeichen verwendet werden muß. Tausendertrennzeichen und Währungssymbole sind nicht zulässig. Enthält `S` keinen gültigen Wert, wird eine `EConvertError-Exception` ausgelöst.

Hinweis:

Die Basicscriptsprache macht solche Typenumwandlungen selbstständig. Diese Funktion ist aber sinnvoll, wenn explizit Daten eines bestimmten Typs übergeben werden sollen (z.B. an Excel, da sonst Excel selbst versucht zu erkennen, um welchen Datentype es sich handelt, und dies oftmals nicht gelingt!)

StrToInt

function `StrToInt(S: string): Integer`

Beschreibung

`StrToInt` konvertiert den in `S` angegebenen String in einen Integer-Wert. `S` muß die Entsprechung eines ganzzahligen Wertes (dezimal oder hexadezimal) sein. Enthält er keinen gültigen Wert, wird eine `EConvertError-Exception` ausgelöst.

Hinweis:

Die Basicscriptsprache macht solche Typenumwandlungen selbstständig. Diese Funktion ist aber sinnvoll, wenn explizit Daten eines bestimmten Typs übergeben werden sollen (z.B. an Excel, da sonst Excel selbst versucht zu erkennen, um welchen Datentype es sich handelt, und dies oftmals nicht gelingt!)

StrToIntDef

function `StrToIntDef(S: string Default: Integer): Integer`

Beschreibung

`StrToIntDef` konvertiert den String `S`, der eine Integer repräsentiert (in dezimaler oder hexadezimaler Form), in eine Zahl. Wenn `S` keinen gültigen Wert enthält, wird der mit `Default` angegebene Standardwert zurückgegeben.

Hinweis:

Die Basicscriptsprache macht solche Typenumwandlungen selbstständig. Diese Funktion ist aber sinnvoll, wenn explizit Daten eines bestimmten Typs übergeben werden sollen (z.B. an Excel, da sonst Excel selbst versucht zu erkennen, um welchen Datentype es sich handelt, und dies oftmals nicht gelingt!)

StrToTime

function `StrToTime(S: string): TDateTime`

AFS-Maskeneditor

Beschreibung

Mit StrToTime kann ein String mit einer Zeitangabe geparkt werden. Enthält S keine gültige Uhrzeit, wird eine EConvertError-Exception ausgelöst.

Der Parameter S muß aus zwei oder drei Zahlen bestehen, die durch das in der globalen Variable DateSeparator gespeicherte Zeichen voneinander getrennt sind. Optional kann danach noch die Angabe AM oder PM folgen. Die Zahlen entsprechen den Stunden, Minuten und (optional) Sekunden. Für das in Europa gebräuchliche 24-Stunden-Format braucht AM oder PM nicht angegeben zu werden.

Hinweis

Das Format des Datumsstrings ändert sich, wenn der Wert einer Formatvariablen für Währungs- und Datums-/Zeitangaben geändert wird.

Hinweis:

Die Basicscriptsprache macht solche Typenumwandlungen selbstständig. Diese Funktion ist aber sinnvoll, wenn explizit Daten eines bestimmten Typs übergeben werden sollen (z.B. an Excel, da sonst Excel selbst versucht zu erkennen, um welchen Datentype es sich handelt, und dies oftmals nicht gelingt!)

Time

function Time: TDateTime

Beschreibung

Time gibt die aktuelle Uhrzeit als Wert des Typs TDateTime zurück.

Beispiel:

```
DateTime = Time ' aktuelle Uhrzeit und Datum speichern
str = TimeToStr(DateTime) ' Uhrzeit in einen String konvertieren
Caption = str ' Uhrzeit in der Titelleiste des Formulars anzeigen
„ Hinweis: Dies hätte auch mit der folgenden Programmzeile ausgeführt
werden können:
Label1.Caption = TimeToStr(Time)
```

TimeToStr

function TimeToStr(Time: TDateTime): string

Beschreibung

TimeToStr konvertiert einen TDateTime-Wert (Time) in einen String. Dabei wird das mit der globalen Variablen LongTimeFormat festgelegte Format verwendet. Das Format des zurückgegebenen Strings kann mit den entsprechenden Datums- und Zeitvariablen geändert werden.

Trim

function Trim(S: string): string

Beschreibung

Trim entfernt entfernt alle am Anfang und Ende des mit S angegebenen Strings vorhandenen Leer- und Steuerzeichen.

```
S=Trim(TBL_Kunden.FieldName(„Land“).AsString+
„TBL_Kunden.FieldName(„Plz“).AsString)+
„+TBL_Kunden.FieldName(„Ort“).AsString
```

Wenn kein Land eingegeben wurde so wird der Platzhalterfreizeichen zwischen Land und Plz entfernt!

TrimLeft

```
function TrimLeft(S: string): string
```

Beschreibung

TrimLeft entfernt alle am Anfang des mit S angegebenen Strings vorhandenen Leer- und Steuerzeichen.

TrimRight

```
function TrimRight(S: string): string
```

Beschreibung

TrimRight entfernt alle am Ende des mit S angegebenen Strings vorhandenen Leer- und Steuerzeichen.

Trunc

```
function Trunc(X: Double): Integer
```

Beschreibung

Trunc konvertiert eine Gleitkommazahl in einen Integer-Wert.

X ist ein Gleitkommaausdruck. Die Funktion gibt einen Integer-Wert mit dem gegen 0 gerundeten Wert von X zurück.

Liegt der Integer-Wert von X außerhalb des Wertebereichs einer Integer-Zahl, tritt ein Fehler auf, der mit einer EInvalidOp-Exception behandelt werden kann. Wird keine Fehlerbehandlung durchgeführt, tritt ein Laufzeitfehler auf.

```
ShowMessage(Trunc(3.98)) ' gibt 3 aus
```

UpperCase

```
function UpperCase(S: string): string
```

Beschreibung

UpperCase konvertiert den Inhalt von S in Großbuchstaben und gibt einen Zeiger auf den String zurück. Bei der Operation werden nur 7-Bit-ASCII-Zeichen zwischen „a“ und „z“ berücksichtigt. Sollen internationale 8-Bit-Zeichen konvertiert werden, verwenden Sie statt dessen die Funktion AnsiUpperCase.

AFS-Maskeneditor

Beispiel:

ShowMessage(„Aber Hallo“) ' gibt ABER HALLO aus

VarArrayCreate

function VarArrayCreate(Bounds: array of Integer VarType: Integer): Variant

Beschreibung

VarArrayCreate erstellt ein variantes Array. Die Ober- und Untergrenzen werden mit Bounds und der Datentyp der Elemente mit VarType festgelegt. Für jede Array-Dimension muß im Parameter Bounds ein Wertepaar für die Ober- und Untergrenze angegeben werden.

Der Datentyp der Elemente wird mit dem Parameter VarType festgelegt. Es muß einer der TVarData-Typcodes verwendet werden, jedoch sind die Bits varArray und varByRef nicht zulässig. Der Elementtyp varString darf ebenfalls nicht verwendet werden. Um ein variantes String-Array zu erstellen, verwenden Sie statt dessen den Code varOleStr. Ist der Elementtyp varVariant, sind die Array-Elemente selbst Varianten und können ihrerseits variante Arrays enthalten.

Hinweis

Variante Arrays mit dem Elementtyp varByte sind besonders für das Übergeben binärer Daten zwischen OLE-Automatisierungs-Controllern und OLE-Automatisierungs-Servern geeignet. Bei diesen Arrays findet keine Datenkonvertierung statt, und mit den Routinen VarArrayLock und VarArrayUnlock kann effizient auf sie zugegriffen werden.

VarArrayHighBound

function VarArrayHighBound(A: Variant Dim: Integer): Integer

Beschreibung

VarArrayHighBound gibt die Obergrenze der Dimension Dim des in A angegebenen varianten Arrays zurück. Für die erste Dimension sollte im Parameter Dim der Wert 1, für die zweite Dimension der Wert 2 usw. übergeben werden. Wird in A kein variantes Array angegeben, oder ist die in Dim angegebene Dimension nicht vorhanden, wird eine EVariantError-Exception ausgelöst.

VarArrayLowBound

function VarArrayLowBound(A: Variant Dim: Integer): Integer

Beschreibung

VarArrayLowBound gibt die Untergrenze der Dimension Dim des angegebenen varianten Arrays zurück. Für die erste Dimension sollte im Parameter Dim der Wert 1, für die zweite Dimension der Wert 2 usw. übergeben werden. Wird in A kein variantes Array angegeben, oder ist die in Dim angegebene Dimension nicht vorhanden, wird eine EVariantError-Exception ausgelöst

```
function VarIsNull(V: Variant): Boolean
```

Beschreibung

VarIsNull gibt True zurück, wenn die angegebene Variante den Wert Null hat. Enthält sie einen anderen Wert, wird False zurückgeliefert.

Hinweis

Verwechseln Sie nicht das Löschen einer Variante mit dem Zuweisen eines Nullwertes an eine Variante. Letztere ist immer noch zugewiesen, hat aber den Wert Null. Sie kann in Ausdrücken verwendet und in andere Typen von Varianten konvertiert werden.

VarToStr

```
function VarToStr(V: Variant): string
```

Beschreibung

VarToStr konvertiert den Inhalt der Variante V in einen String.

Write

```
procedure Write(F, V1,...,Vn)
```

Beschreibung

Write schreibt eine Datei in eine Dateikomponente. F ist eine Dateivariable und jedes V eine Variable mit dem Typ der in F angegebenen Komponente. Nach jedem Schreiben einer Variablen wird der Dateizeiger auf die nächste Komponente gesetzt. Befindet sich der Dateizeiger bereits am Dateiende (Eof(F) liefert True), wird die Datei vergrößert.

WriteLn

```
procedure WriteLn(var F: Text , s : string )
```

Beschreibung

WriteLn ist eine Erweiterung der Prozedur Write, die für Textdateien definiert ist.

Nach dem Ausführen von Write schreibt WriteLn ein Zeilenendezeichen (Wagenrücklauf/Zeilenvorschub) in die Datei. WriteLn(F) ohne Parameter schreibt nur ein Zeilenendezeichen.

Die angegebene Datei muß für die Ausgabe geöffnet sein.

Beispiel:

```
DIM F1
s=„C:\test\test.txt“
AssignFile(F1, s)
ReWrite(F1)
WriteLN(F1,TBL_Auftrag.FieldName(„Sendungsnummer“).AsString)
CloseFile(F1)
```

Zugriff auf interne Variablen (Formularsprache)

Die Software besitzt seit von Grundauf eine leistungsstarke Formularsprache, die für alle Ausgaben Anwendung findet. Diese wird im Adminhandbuch erklärt. Damit Sie auch via Basicscript auf die Funktionen Zugriff haben wurden stehen folgende Funktionen zur Verfügung:

GetVariable(string):string

GetVariable gibt den Inhalt einer Variable zurück

Beispiel:

```
ShowMessage(GetVariable(„TBL_Kunden.Bemerkung.RTF“))
```

Gibt den Inhalt der Bemerkung aus und wandelt den als RTF-Text (formatierter Text) in ASC-Text (unformatiert). Diese Umwandlung das der Formularbefehl .RTF

Beispiel 2:

```
ShowMessage(GetVariable(„TBL_Kunden.Nummer.EAN“))
```

Gibt die Kundennummer als EANCode 13 zurück mit aufgefüllten 0 und Checksumme an der 13 Stelle

GetFormular(string):string

Diese Funktion verhält sich wie GetVariable, nur mit dem Unterschied, das nicht nur eine Variable übergeben werden kann, sondern mehrere. Diese werden innerhalb des Übergabestrings mit geschweiften Klammer {VARIABLE} getrennt.

Beispiel:

```
ShowMessage(GETFormular(„Ihre Kundenr ist {TBL_Kunden.Nummer}“))
```

Beispiel 2:

```
ShowMessage(GETFormular(„Ihre Kundenr ist {TBL_Kunden.Nummer} und Ihre Auftragsnr {TBL_Auftrag.Vorgangsnummer}“))
```

Zugriff auf das Dateisystem (eigener Export & Import)

Mit der Basic-Scriptsprache steht Ihnen auch der volle Zugriff auf das Dateisystem Ihres Netzwerks zur Verfügung. Diese werden wir hier einmal an drei Beispielen erläutern.

Grundfunktionen für den Dateizugriff

GetCurrentDir Gibt das aktuelle Verzeichnis zurück
GetTempDir Gibt das Tempverzeichnis des Rechners zurück
GetAppDir Gibt das Verzeichnis des Programms zurück
FileExists Gibt an ob eine Datei existiert (Rückgabewert 1)
AssignFile Weist eine Variablen eine Datei zu
ReWrite Öffnet die Datei zum Schreiben
Rest Öffnet die Datei zum Lesen
Write Schreibt in die Datei
WriteLn Schreibt in die Datei incl. ENTER
ReadLn Liest eine Zeile aus einer Datei
CloseFile Schließt die Datei
MakeDir Erstellt ein Verzeichnis
DeleteFile Löscht die angegebene Datei
FileShow Führt ein Programm aus

Beispiel 1:

Schreibzugriff in eine Datei (hier die Sendungsnummer ausgeben)

```

DIM F1
s="C:\test\test.txt"
AssignFile(F1, s)
ReWrite(F1)
WriteLN(F1,TBL_Auftrag.FieldByName(„Sendungsnummer“).AsString)
CloseFile(F1)
  
```

Beispiel 2:

Lesezugriff auf eine Datei (hier die Sendungsnummer einlesen)

```

DIM F1
s="C:\test\test.txt"
AssignFile(F1, s)
Rest(F1)
TBL_Auftrag.FieldByName(„Sendungsnummer“).AsString =ReadLn(F1)
CloseFile(F1)
  
```

Beispiel 3:

Lesezugriff und Schreibzugriff auf zwei Dateien (hier eine Datei kopieren)

```

DIM F1,F2
DIM inhalt
s="C:\test\test.txt"
s2="C:\test\copy.txt"
  
```

```

if InputQuery(„Dateinamen“,„Dateinamen eingaben“, s) then
  AssignFile(F1, s)
  AssignFile(F2, s2)
  Reset(F1)
  Rewrite(F2)
  Do while EOF(F1)=False
  
```

```

AFS-Maskeneditor
  inhalt=Readln(F1)
  Label1.Caption=100*FilePos(F1)/FileSize(F1)+ "% kopiert"
  WriteLN(F2,inhalt)
LOOP
CloseFile(F1)
CloseFile(F2)
ShowMessage(„Datei kopiert“)
end if

```

Dateidialoge

TOpenDialog Dieser kapselt eine Datei öffnen Dialog
 TSaveDialog Dieser kapselt eine Datei speichern Dialog

Beide Dialogtypen haben die gleichen Eigenschaften:

.EXECUTE Zeigt den Dialog und hat den Rückgabewert TRUE,
 wenn der Benutzer auf OK Klickt

.FILENAME Dateiname der ausgewählt wurde

Beispiel:

```

dia= TOpenDialog.Create( Self )
if dia.Execute then
  ShowMessage(„ausgewählt wurde „+dia.FileName)
end if

```

Arbeiten mit OLE-Objekten

OLE-Objekte sind Programme, die einen Programmierschnittstelle bieten und Funktionen, Eigenschaften und Unterobjekt zur Verfügung stellen. Dies unterstützen z.B. alle Officeprogramme wie Word, Excel, Outlook usw. Dieses mächtigste Werkzeug ermöglicht es Ihnen diese Programm komplett Fernzubedienen (wie von Geisterhand).

Ein OLE-Objekt (die Verbindung zum Fremden Programm) erzeugt man mit der Funktion

PROGRAMMOBJEKT = **createoleobject**(„Programmklassenname“)

Programmklassenname: dieser Name ist von dem Anzubindenden Programm festvorgegeben und diesen finden Sie in der Anleitung des anderen Programms (Ansonsten mal Google fragen OLE und Programmnamen)

Programmobjekt : dieses beinhaltet das ganze Programm und ermöglicht den Zugriff. Welche Funktionen und Unterobjekt hier zur Verfügung stehen finden Sie in der Anleitung des anderen Programms (Ansonsten mal Google fragen OLE und Programmnamen)

Tip: Bei Officeprogrammen öffnen Sie den VBA Funktion, dann finden Sie

hier die Onlinehilfe mit allen Funktionen.

Hinweis: damit OLE benutzt werden kann muss das entsprechende Programm auf dem Rechner installiert sein!

Damit der Speicherplatz endgültig wieder freigegeben wird, sollten Sie bei allen Objekten, die Sie nicht sichtbar machen, dieses Programmtechnisch beenden und das PROGRAMMOBJEKT wieder freigeben, indem man diesem null zuweist:

Hier folgen nun einige Beispiele:

Excel

Export des Feld E-Mail der Adressenauswahl zu Excel:

```
excel = createoleobject(„excel.application“)
excel.visible = True
workbook = excel.workbooks.add
worksheet = workbook.ActiveSheet
TBL_Kunden.First ' auf ersten Datensatzstellen
i=1
do while (TBL_Kunden.EOF=0) ' bis zum letzten Datensatz
  s=TBL_Kunden.FieldName(„E-Mail“).Value
  if Trim(s)<>"" then
    worksheet.Cells(i,1).Value=TBL_Kunden.FieldName(„E-Mail“).Value
    i = i + 1 , Zähler erhöhen
  end if
  TBL_Kunden.Next ' zum nächsten Datensatz blättern
loop ' Schleifen
```

Erklärung:

```
excel = createoleobject(„excel.application“)
```

Dies erzeugt das Excelobjekt dieses steht nun in der Variable excel zur Verfügung.

```
excel.visible = True
```

Das Excelobjekt stellt viele Funktionen zur Verfügung welche Sie in der Excel Dokumentation (VBA) finden.

Visible steht für sichtbar und mit Visible=True wird Excel nun angezeigt

```
workbook = excel.workbooks.add
```

Excel besteht aus verschiedenen Tabellen und einem zentralen Objekt, welches diese verwaltet (workbook). Mit der Methode Add wird eine Sammelliste angelegt.

```
worksheet = workbook.ActiveSheet
```

Das Workbook Objekt enthält nun die Tabellen und mit ActiveSheet wird die aktuelle ermittelt

AFS-Maskeneditor

```
worksheet.Cells(i,1).Value=TBL_Kunden.FieldByName(„E-Mail“).Value
```

Eine Tabelle besteht nun aus Zellen, die mit CELLS(x,y) abgefragt werden kann.

Mit Value wird dieser Zelle ein Wert übergeben

Wenn Sie nun Excel wieder beenden möchten, so gibt es hier die Funktion .QUIT

Excel.Quit

Damit der Speicherplatz endgültig wieder freigegeben wird, sollten Sie bei allen Objekten, die Sie nicht sichtbar machen, dieses Programmtechnisch beenden und das PROGRAMMOBJEKT wieder freigeben, indem man diesem null zuweist:

Excel = null

Outlook

Hier ein Beispiel welches eine E-Mail sofort sendet:

```
oMailItem = 0
```

```
Set oOApp = CreateOleObject(„Outlook.Application“)
```

```
Set oOMail = oOApp.CreateItem(oMailItem)
```

```
oOMail.To = TBL_Kunden.FieldbyName(„E-Mail“).AsString
```

```
oOMail.Subject = „Test-Betreff“
```

```
oOMail.Body = „Sehr geehrter Herr/Frau“ +
```

```
TBL_Kunden.FieldbyName(„Name“).AsString + „ test“
```

```
oOMail.Send
```

Wenn Sie die letzte Zeile ersetzen mit oOMail.Display statt oOMail.Send, dann wird die Mail nicht sofort gesendet, sondern Ihnen zum bearbeiten nochmals angezeigt!

Beispiel:

Einen Newsletter versenden:

```
oMailItem = 0
```

```
Set oOApp = CreateOleObject(„Outlook.Application“)
```

```
Set oOMail = oOApp.CreateItem(oMailItem)
```

```
oOMail.Subject = „Test-Betreff“
```

```
oOMail.Body = „Sehr geehrter Herr/Frau“ +
```

```
TBL_Kunden.FieldbyName(„Name“).AsString + „, test“
```

```
TBL_Kunden.First ' auf ersten Datensatzstellen
```

```
i=1
```

```
do while (TBL_Kunden.EOF=0) ' bis zum letzten Datensatz
```

```
  s=TBL_Kunden.FieldByName(„E-Mail“).Value
```

```
  if Trim(s)<>"" then
```

```
    oOMail.To = TBL_Kunden.FieldbyName(„E-Mail“).AsString
```

```
    oOMail.Send
```

```
  end if
```

```
  TBL_Kunden.Next ' zum nächsten Datensatz blättern
```

Seite 188

loop ' Schleifen

Tobit

Hier ein Beispiel welches eine E-Mail sofort sendet:

```
DvEMailItem = 2 ' // MailItem Objekt
Set oApp = CreateOleObject("DVOBJAPILib.DvISEAPI")
Set oAccount = oApp.Logon("", "", "", "", "", "AUTH")
Set oArchive = oAccount.GetSpecialArchive(102)
Set oMailItem = oArchive.NewItem(DvEMailItem)
oMailItem.Recipients.Add(TBL_Kunden.FieldName("E-Mail").AsString,
„MAIL“, "")
oMailItem.Subject = "Test-Betreff"
oMailItem.BodyText.HTMLText = "<html><body>Sehr geehrte/r Herr/Frau
"+TBL_Kunden.FieldByName(„Name“).AsString + ", test... <br><body></
html>"
oMailItem.Options.UserHold = True
oMailItem.Send
```

Word

Hier ein Beispiel, wie man Daten an Word übergibt

```
Set Word = CreateOLEObject("Word.Application")
Word.Visible = True
Word.Documents.Add
S = Word.Selection
S.TypeText("Hallo Welt")
S.TypeParagraph
S.TypeParagraph
S.TypeText("Und das war")
```

XML-Zugriff

Hier ein Beispiel, wie man Daten aus einer XML-Datei auslesen kann

, XMLDOM demo

```
xmld = CreateOleObject("microsoft.xmlDOM")
if xmld.Load("orders.xml") then
nodel = xmld.DocumentElement.ChildNodes
for i = 1 to nodel.Length
node = nodel.Item(i-1)
nodec = node.ChildNodes
TBL_Kunden.Append
TBL_Kunden.FieldName("Firma").AsString=nodec.Item(0).Text
TBL_Kunden.FieldName("Vorname").AsString=nodec.Item(1).Text
TBL_Kunden.FieldName("Name").AsString=nodec.Item(2).Text
```

AFS-Maskeneditor

```
TBL_Kunden.FieldName("Strasse").AsString=nodec.Item(3).Text
TBL_Kunden.FieldName("Land").AsString=nodec.Item(4).Text
TBL_Kunden.FieldName("Ort").AsString=nodec.Item(5).Text
TBL_Kunden.FieldName("E-Mail").AsString=nodec.Item(6).Text
TBL_Kunden.FieldName("Telefon").AsString=nodec.Item(7).Text
TBL_Kunden.Post
next
end if
xmld = null
```

In diesem Beispiel wird eine Datei ORDER.XML geöffnet, die Adressen enthält.

Externe Datenbanken ADO-Zugriff

Hier ein Beispiel, wie man Daten aus einer Access Datei auslesen kann. Über den Connectionstring (hier myDSN) kann auf beliebige Datenbanken und SQL-Server zugegriffen werden.

```
ObjConnC = CreateOleObject("ADODB.Connection")
log.Items.Add("ADO connection created")
myDSN = "Driver={Microsoft Access Driver (*.mdb)};DBQ=countries.mdb"
ObjConnC.Open(myDSN)
log.Items.Add("ADO connection opened")
ObjRSC = CreateOleObject("ADODB.RecordSet")
mySQL = "SELECT * FROM COUNTRIES"
ObjRSC.ActiveConnection = ObjConnC
ObjRSC.Open(mySQL)
i = 0
countrycombo.Items.Clear()

do while (ObjRSC.EOF = False)
' get the field
fld = ObjRSC.Fields("COUNTRY")
' add field value to combobox
countrycombo.Items.Add( fld.value)
' count nr. of items added
i = i + 1
ObjRSC.MoveNext
loop
log.items.Add("Added " +inttostr(i)+" records")
ObjRSC.Close
ObjConnC.Close
log.items.Add("ADO connection closed")
countrycombo.ItemIndex = 12
ShowMessage("Script executed")
```

NEUE AFS-Basic Befehle und Eigenschaften für noch tiefere Systemeingriffe:

DataSet Events sind nun verfügbar und können mit eigenem Code beeinflusst werden!

Beispielprogramm:

```
Sub myAfterInsert(Dataset)
  ShowMessage(„nach dem Einfügen“)
end Sub
```

```
Sub myAfterScroll(Dataset)
  ShowMessage(„nach dem Bewegen“)
end Sub
```

```
Sub myBeforePost(Dataset)
  ShowMessage(„vor dem Speichern“)
end Sub
```

```
Sub myAfterPost(Dataset)
  ShowMessage(„nach dem Speichern“)
end Sub
```

```
Sub myBeforeDelete(Dataset)
  ShowMessage(„vor dem Löschen“)
end Sub
```

```
Sub myBeforeCancel(Dataset)
  ShowMessage(„vor dem Rückgängig“)
end Sub
```

```
Sub myAfterEdit(Dataset)
  ShowMessage(„jetzt im Einfügemodus“)
end Sub
```

```
Sub myBeforeEdit(Dataset)
  ShowMessage(„vor dem Einfügen“)
end Sub
```

```
TBL_Kunden.BeforePost="myBeforePost"
TBL_Kunden.AfterScroll="myAfterScroll"
TBL_Kunden.AfterInsert="myAfterInsert"
TBL_Kunden.BeforeDelete="myBeforeDelete"
TBL_Kunden.BeforeCancel="myBeforeCancel"
TBL_Kunden.BeforeEdit="myBeforeEdit"
TBL_Kunden.AfterEdit="myAfterEdit"
```

AFS-Maskeneditor

SQL-Anweisungen sind nun via Script übergebbar (Änderbar):

TDBGRIDALL.SQL

TQueryData.SQL

TSQLLabel.SQL

Beispiel:

USERCOMPONENT664.SQL="SELECT COUNT(*) FROM Positionen WHERE Adresse=:Adresse"

E-Mail senden aus dem Programm

mit entsprechenden Einstellungen:

AppFunc("MAIL:...")

Ruft die Mailfunktion auf.

Die Parameter werden mit dem ~ Zeichen getrennt, dies darf nicht im Subjekt oder Body der Mail vorkommen.

Parameter- Reihenfolge:

- Empfänger (an)
- Cc
- Betreff
- Body (Text)
- Anhang (Dateinamen), bei mehreren mit Zeichen ; getrennt
- Sofort senden (JA; TRUE) bei senden ohne weiteren Dialog

Beispiel:

AppFunc("MAIL:post@afs-software.de~post@afs-software.com~TESTMAIL aus dem Script~BODY~C:\Arno\Temp\0boddddn.txt")

Drucken aus dem Programm

mit entsprechenden Einstellungen:

AppFunc("PRINT:...")

Ruft die Druckfunktion auf.

Die Parameter werden mit dem ~ Zeichen getrennt, dies darf nicht in ei-

nem Parameterwert vorkommen.

Parameter- Reihenfolge:

- Formular
- Kopien
- Titel
- Hauptdataset, die gedruckt werden soll
- Gruppendataset

Beispiel:

Inerhalb des Artikelfensters Etiketten der gesamten Artikelliste drucken

```
AppFunc("PRINT:DRUCKEN\ETIEKTT.LBL~1~Etiketten~TBL_ARTIKEL")
```

Inerhalb des Artikelfensters Etiketten des angezeigten Artikels drucken
(Hauptdataset weglassen, dann wird nicht das gesamte Dataset durchgescrollt zum Drucken)

```
AppFunc("PRINT:DRUCKEN\ETIEKTT.LBL~1~Etiketten")
```

Beliebige SQL-Anweisung ausführen

mit APPFUNC("DOSQL: HIER DIE SQL ANWEISUNG ") können Sie UPDATE, DELETE, INSERT und StoredProcedures auf dem SQL-Server ausführen lassen.

Beispiel:

hier in der Warengruppenbearbeitung alle Artikel ausblenden, die zu der aktuellen angeklickten Warengruppe gehören:

```
SHOWMESSAGE(APPFUNC("DOSQL:UPDATE ARTIKEL SET ART=255 WHERE WARENGRUPPE="+TBL_WARENGR.FIELDBYNAME("WARENGRUPPE").AsString))
```

Diese Funktion gibt als String entweder OK zurück oder im Fehlerfall SQL-Error: mit der Rückgabe des SQL-Servers

Beliebige SQL-Abfrage ausführen

mit APPFUNC("SQL: HIER DIE SQL ANWEISUNG ") können Sie einen SELECT-befehl, eine View oder eine StoredProcedures auf dem SQL-Server ausführen lassen.

Beispiel:

hier in der Warengruppenbearbeitung wieviele alle Artikel hat die Waren-

AFS-Maskeneditor
gruppe, die aktuelle angeklickt ist:

```
SHOWMESSAGE(APPFUNC("SQL: SELECT COUNT(*) FROM ARTIKEL WHERE  
ART=0 AND WARENGRUPPE =" + TBL_WARENGR.FIELDBYNAME  
("WARENGRUPPE").AsString))
```

Diese Funktion gibt als String den Wert des ersten Feldes zurück.

Eine Aktion des Hauptfensters ausführen

mit APPFUNC("ACTION:ACTIONNAME") können Sie eine Action (Funktion) des Hauptfensters ausführen lassen:

Beispiel:

Öffnet die Kundenbearbeitung:

```
APPFUNC("ACTION:Ac_Ku_be")
```

Öffnen der Vorgangsliste

mit APPFUNC("OPEN_VORGANGSLISTE:VORGANGSARTID") können Sie eine die Vorgangsliste öffnen:

Beispiel:

Öffnet die Angebotsliste:

```
APPFUNC("OPEN_VORGANGSLISTE:1")
```

Öffnen eines Vorgangs

mit APPFUNC("OPEN_VORGANG:VORGANGSID") können Sie einen Vorgang öffnen:

Beispiel:

Öffnet ein Vorgang aus der Vorgangsliste:

```
APPFUNC("OPEN_VORGANG:" + TBL_Auftrag.FIELDBYNAME("AUFTRAG").AsString)
```

Kopieren eines Vorgangs

mit APPFUNC("VORGANGKOPIE:VORGANGSID~NEUE_VORGANGSARTID")

können Sie einen Vorgang kopieren:

Beispiel:

Aus der Vorgangsliste den aktuellen Vorgang in ein Angebot kopieren

```
APPFUNC("VORGANGKOPIE: "+TBL_Auftrag.FIELDBYNAME("AUFTRAG").AsString~2)
```

Forführen eines Vorgangs

mit

```
APPFUNC("VORGANGFORT: VORGANGSID~NEUE_VORGANGSARTID~RÜCKSTAND  
_beachten~REST_EINER~TEILLIEFERUNG~alten_Vorgangs_abschließen")
```

können Sie einen Vorgang fortführen:

Parameter:

- **VORGANGSID**
- **NEUE_VORGANGSARTID**
- Rückstand beachten (True/False)
- nur Restbestand Auftrag behalten (Rest einer Teillieferung) (True/False)
- alten Vorgang abschließen (True/False)

Beispiel:

Aus der Vorgangsliste den aktuellen Vorgang in eine Auftragsbestätigung fortführen

```
APPFUNC("VORGANGKOPIE: "+TBL_Auftrag.FIELDBYNAME("AUFTRAG").AsString~3)
```

Ausgabe eines Vorgangs (Drucken, Mailen, Archivieren als PDF)

mit

```
APPFUNC("VORGANGAUSGABE: VORGANGSID~AUSGABEART~ZUSATZINFOS")
```

können Sie einen Vorgang ausgeben

AFS-Maskeneditor

Parameter:

- VORGANGSID
- Ausgabeart (0=Druck; 1=Mailen)
- Zusatzinfos:
 - DIREKT = Drucken ohne Druckerdialog; bei Mailen ohne weitere Nachfrage

Beispiel:

Aus der Vorgangsliste den aktuellen Vorgang direkt Mailen

```
APPFUNC("VORGANGAUSGABE:"+TBL_Auftrag.FIELDBYNAME("AUFTRAG").AsString~1~DIREKT)
```

Eigene Ereignisse (Eventhandling)

Mit der Basic-Scriptsprache stellt Ihnen auch den Zugriff auf gewisse Ereignisse (Events) zu. Hierzu können Sie eigene Procedures Anlagen, die die Ereignisbehandlung enthalten können. Diese werden wir an einem Beispiel erläutern.

```
Sub miOption1Click(Sender)  
  ShowMessage(„Heute ist „+DateToStr(Date))  
end Sub
```

```
Sub SetEvents  
  Button1.OnClick = „miOption1Click“  
  ShowMessage(„Ab jetzt wird der Event abgearbeitet“)  
end Sub
```

SetEvents

Dynamische Forms:

Um neue Objekte zu erzeugen, benötigen die einen Besitzer und Elternobjekt (Parent) auf dem diese Angezeigt werden.

Den Besitzer übergibt man beim Create Befehl, der das Objekt erzeugt.

SELF ist das Fenster, was gerade offen ist!

In diesem Beispiel wird ein Fenster erzeugt (fm), danach werden weitere Unterelemente erzeugt die diesem Fenster gehören (.Create(fm)) und diesem schließlich zugewiesen werden (.Parent=fm).

Dieses Team ist nur für erfahrene Programmierer geeignet und Google hält zu den Objekttypen zahlreiche Beschreibungen bereit, welches den Umfang dieser Einführungsanleitung deutlich übersteigen würde!

Beispiel:

```
fm = TForm.CreateNew( Self )
fm.Caption = "eine neue form!"
fm.Left = 200
fm.Top = 120
fm.Width = 150
fm.Height = 130
fm.BorderStyle = bsToolWindow
lb = TLabel.Create( fm )
lb.Parent = fm
lb.Left = 10
lb.Top = 10
lb.Caption = "Ihr Name: "
ed = TEdit.Create( fm )
ed.Parent = fm
ed.Left = lb.Left
ed.Top = lb.Top + lb.Height + 1
bt = TBitBtn.Create( fm )
bt.Parent = fm
bt.Left = ed.Left
bt.Top = ed.Top + ed.Height + 20
bt.Kind = bkOK
if fm.ShowModal = mrOk then
  ShowMessage("Hallo "+ed.Text+" !!!")
end if
```

Timer Objekt (Eventhandling)

Mit der Basic-Scriptsprache stellt Ihnen auch den Zugriff eigene Timer zur Verfügung. Timer führen in gewissen Abständen ein Ereignis onTimer aus, wenn

dieser eingeschaltet wird.

Der Timer hat folgende Eigenschaften:

.Enabled Ein- (True) oder Ausgeschaltet (False)

.Interval Zeit in milli sec, in dem das Ereignis onTimer aufgerufen wird Standardwert ist 1000 mili sec = 1 Sekunde

AFS-Maskeneditor

Beispiel:

Dieses Beispiel erzeugt einen Timer, der bei einem eigenen Schalter die aktuelle Uhrzeit anzeigt:

```
Sub myonTimer(Sender)
  USERCOMPONENT667.Caption=TimeToStr(now)
end Sub
ti= TTimer.Create( Self )
ti.OnTimer="myonTimer"
ti.Enabled=True
```

Action Objekt (Funktionszugriff)

Mit der Basic-Scriptsprache stellt Ihnen den Zugriff auf programmeigene Funktionen zur Verfügung.

Actions kapseln Programmfunktionen incl. Beschriftungen, die Menüs und Schaltern wieder zugewiesen werden können:

```
xxx.Action=ActionName
```

Aufrufen können Sie diese Actionions mittels Execute:

```
xxx.Execute
```

Beispiel

```
Sub myonclick(Sender)
  USERCOMPONENT648.Caption="Hallo Welt"
end Sub
```

```
act=TAction.Create(self)
act.Caption="Hallo"
act.OnExecute="myonclick"
USERCOMPONENT649.Action=act
```

Beispiel aus einen Script in der Adressmaske die History aufrufen:

```
Act_History.Execute
```

Index

A

Abs 156
AFS- Basicscript-Sprache 146
AFS-Basic Syntax: 149
Aktionen 33
Aktionen der Arbeitszeitverwaltung 74
Aktionen der Artikeltarife 74
Aktionen der Auftragsbestätigung 75
Aktionen der Barkasse 80
Aktionen der Geräteverwaltung 86
Aktionen der Lagerbuchung 95
Aktionen der neuen Aufträge 104
Aktionen der Notizenverwaltung 122
Aktionen der Touchkasse 133
Aktionen der Umsatzsteuerauswertung 137
Aktionen der Vorgangsliste 137
Aktionen der Warengruppen 140
Aktionen des Bestellwesens 84
Aktionen des DTA-Bankings 85
Aktionen des Terminplaners 131
Aktionen des Textwriters 132
Aktionen des Wartungs- und Abowesen 142
Aktionen für „Artikelliste nach Warengruppen“ 53
Aktionen für das Fenster „Lagerbuchung“ 55
Aktionen für die Inventur 56
Aktionen für die Preisliste 58
Aktionen für Angebote 64
Aktionen für das Fenster „Warengruppe“ 52
Aktionen für das Kassenbuch 94
Aktionen für das Kontaktmanagement 38
Aktionen für das Verdichten von Vorgängen 137
Aktionen für den Abbuchungsauftrag 63
Aktionen für die Artikelverwaltung 33
Aktionen für die Interessentenbearbeitung 46
Aktionen für die Kundenbearbeitung 43
Aktionen für die Lieferantenbearbeitung 49
Aktionen für die Rabattmatrix 59
Aktionen für die Warengruppenauswertung 141
Aktionen für externe Preislisten 86
Aktionen für Gutschriften 88
Aktionen für Kassenabschlussberichte 94
Aktionen für Lastschriften 97
Aktionen für Lieferscheine 98
Aktionen für Mitarbeiter 104
Aktionen für MwSt-Einstellungen 104
Aktionen für neue Angebote 112

AFS-Maskeneditor
Aktionen für Rechnungen 122
aktive Elemente 143
AnsiCompareStr 156
AnsiCompareText 157
AnsiLowerCase 157
AnsiUpperCase 157
ArcTan 157
Arrays 152
Assigned 157, 158
AssignFile 158

B

Beep 158, 159
Benutzerspezifische Masken 31
BLOB-Felder 26
Buttons 17

C

Chdir 159
Chr 159
CloseFile 159
CompareStr 159
CompareText 160
Copy 160
CreateOleObject 161

D

Date 161
Datenbankfeld mit Auswahlklappliste 29
Datenbankfeld mit Auswahlliste (Listbox): 27
Datenbanklangtext formatiert 31
Datenbanktabellen und Navigationsleisten 21
Datenbanktextfelder 24
Datenbankverknüpfung als Klappbox 30
DateTimeToStr 161
DateToStr 162
DayOfWeek 162
Dec 162
DecodeDate 162, 163
DecodeTime 163
DeleteFile 163
Die Bezeichner 150
Die Fenster des Maskeneditors 13
Die Kontrollleiste 13
Die Script Strukturen 149
DIM 163

E

E-Mail senden aus dem Programm 192
Einbinden von DLL und deren Funktionen: 155
Einschub: Original Kaufmann wiederherstellen 25
EncodeDate 163, 164
EOF 164
Exp 165

F

Festlegen von Eigenschaften 14
FileExists 165
FilePos 165
FileShow 166
FileSize 166
FloatToStr 166, 167
For Anweisung 154
Format 167
FormatDateTime 167
FormatFloat 167
Frac 168
Funktionen und Sub Deklaration 155
Funktionsübersicht: 156

G

GetActiveOleObject 168
GetAppDir 168
GetCurrentDir 168
GetTempDir 168
GetVariable(string):string 184
Grundeigenschaft von sichtbaren Objekten 149

H

High 169
Hinweise und Hotline-Service 10

I

If - Anweisungen 152
Inc 169
IncMonth 169
Index 152
InputQuery 170
Insert 170
Installation überprüfen 12
Int 170
Internetservice 11
IntToHex 171
IntToStr 171

AFS-Maskeneditor
IsLeapYear 171
IsValidIdent 171

K

Keine Zeit, um das Handbuch zu lesen ??? 10
Kommentare 151

L

Labelfelder 20
Length 172
Ln 172
Loop Anweisung 153
Low 172
LowerCase 172

M

MakeDir 172, 173

N

Näheres zu den Objekten 16
NEUE AFS-Basic Befehle 191
Now 173
Nutzungslizenzvertrag 2

O

Odd 173
Ord 173

P

Panels 17
Pos 173, 174
procedure Reset(var F [: File RecSize: Word]) 175

R

Random 174
ReadLn 174
Reset 174
Rewrite 175
Rmdir 176, 177
Round 176

S

Schalter-Datenbankfeld 29
Select case Anweisung 154
ShowMessage 177
Sin 177
Seite 202

Sqr 177
Sqrt 177, 178
StrToDate 178
StrToDateTime 178
StrToFloat 178, 179
StrToInt 179
StrToIntDef 179
StrToTime 179, 180

T

Tabellenobjekte 147
Time 180
TimeToStr 180
Trim 180
TrimLeft 181
TrimRight 181
Trunc 181

U

UpperCase 181

V

VarArrayCreate 182
VarArrayHighBound 182
VarArrayLowBound 183
Variablen 151
VarIsNull 183
VarToStr 183

W

While Anweisung 153
Write 183
WriteLn 183

Z

Zeichen String 151
Zugewiesene Anweisungen 150
Zugriff auf interne Variablen (Formularsprache) 184

