

# Makrofunktionen

[e] forms and more

Gültig ab Version 7.3.4

27.02.2017

---

## Inhaltsverzeichnis

### Funktionsbeschreibungen

7

AccObjDeleteRecord .....	7
AccObjGetCurrentRecord .....	7
AccObjGetChangeFlag .....	8
AccObjGetMatchMode .....	8
AccObjGetFieldAsDate .....	9
AccObjGetFieldAsFloat .....	10
AccObjGetFieldAsInt .....	11
AccObjGetFieldAsText .....	12
AccObjGetTotalRecords .....	13
AccObjInsertEmptyRecord .....	14
AccObjInsertRecord .....	15
AccObjMatchModeCancel .....	16
AccObjMatchModeConfirm .....	16
AccObjMatchModeEnter .....	17
AccObjMatchModeToggle .....	18
AccObjNavigateAbsolute .....	19
AccObjNavigateFirst .....	20
AccObjNavigateLast .....	21
AccObjNavigateNext .....	22
AccObjNavigatePrev .....	23
AccObjReloadObjects .....	24
AccObjSelect .....	25
AccObjSelectAll .....	26
AccObjSelectExact .....	27
AccObjSelectLead .....	28
AccObjSelectStdMask .....	29
AccObjSetChangeFlag .....	30
AccObjSetDataSource .....	31
AccObjSetFieldAsDate .....	32
AccObjSetFieldAsFloat .....	33
AccObjSetFieldAsInt .....	34
AccObjSetFieldAsText .....	35
AccObjSetOrderByClause .....	36
AccObjSetWhereClause .....	37
AccObjUnselect .....	38
AccObjUpdateRecord .....	39
AddObjects .....	40
AppendObjectSetExt .....	41
BroadcastAllInputs .....	42
CallFormMacro .....	43

CallObjectMacro .....	44
Close .....	44
CopyText.....	45
CountRuntimeChangedPages .....	45
CreateVisitCardFile.....	46
DeleteObjectsByName.....	47
DeleteObjectsInRect.....	48
DllCall .....	49
DllLoadAllObjekts.....	50
DllLoadObject .....	51
DllSaveAllObjekts .....	52
DllSaveObject .....	53
EditFonts.....	54
FindObject .....	55
FindAccessObject.....	56
GetActiveState.....	57
GetAssociatedComboColor.....	58
GetAssociatedComboValue .....	58
GetCapsState.....	59
GetChangeFlag .....	59
GetCheckbox .....	60
GetColorProperty.....	61
GetComboCount .....	62
GetComboIndex.....	63
GetComboListIndex.....	63
GetComboListItem .....	64
GetCopyName .....	64
GetCurrentObject .....	65
GetCurrentPage .....	65
GetDataField .....	66
GetDisplayState .....	66
GetFloatFormatted.....	67
GetFontHeight.....	68
GetFontName.....	68
GetFontProperty.....	69
GetFormWindow .....	70
GetHoriAlign.....	71
GetLongFormatted.....	72
GetLongProperty.....	73
GetMacroCode .....	74
GetObjectAsFloat.....	75
GetObjectDisplayRect.....	76
GetObjectsEnvelope.....	78
GetObjectPos .....	79
GetObjectSize.....	80

GetObjectType.....	81
GetObjectTypeExt.....	82
GetPage.....	82
GetPageBgrMode.....	83
GetPageBgrColor.....	83
GetPageDisplayRect.....	84
GetPageHeight.....	85
GetPageHeightZoomFactor.....	85
GetPageNrByPageName.....	85
GetPageWidth.....	86
GetPageWidthZoomFactor.....	86
GetPointedObject.....	86
GetPointedObjectHandle.....	87
GetPrintState.....	87
GetReadOnlyState.....	88
GetRuntimeChangeFlag.....	88
GetScrollOffset.....	89
GetStringProperty.....	90
GetText.....	91
GetTextLng.....	92
GetTitleFile.....	92
GetTotalPages.....	93
GetVertAlign.....	93
GetXmlProperty.....	94
GetZoom.....	95
GotoObject.....	95
GotoRelative.....	96
HideMenu.....	96
InsertModule.....	97
LoadBitmap.....	97
LoadFromProfile.....	98
NewForm.....	99
NoCloseOnEscape.....	99
Open.....	100
PostFormMacro.....	101
PostObjectMacro.....	102
Print.....	103
Quit.....	104
RefreshForm.....	104
RefreshObject.....	104
Save.....	105
SaveAs.....	105
SaveToProfile.....	106
SetActiveState.....	106
SetBgrColor.....	107

SetCapsState .....	108
SetCapsStateOnPage .....	109
SetChangeFlag .....	109
SetCheckbox.....	110
SetCheckboxOnPage.....	111
SetColorProperty .....	112
SetComboList .....	113
SetComboListByFile.....	114
SetDefaults.....	114
SetDisplayFrame .....	115
SetDisplayState.....	116
SetFloat .....	117
SetFloatFormatted .....	118
SetFloatOnPage .....	119
SetFontColor .....	120
SetFontHeight.....	121
SetFontName .....	122
SetFontProperty .....	123
SetHoriAlign .....	124
SetLongFormatted .....	125
SetLongProperty .....	126
SetMacroCode .....	127
SetModulePath .....	127
SetObjectPos .....	128
SetPageBgrMode .....	128
SetPageBgrColor .....	129
SetPagekeyMode .....	129
SetPictureFile .....	130
SetPrintState .....	130
SetPrintStateOnPage .....	131
SetReadOnlyState .....	132
SetReadOnlyStateOnPage .....	133
SetRuntimeChangeFlag.....	134
SetScrollOffset.....	135
SetStringProperty.....	136
SetText .....	137
SetTextOnPage .....	138
SetTitleMain .....	139
SetTitleMdi.....	139
SetVertAlign .....	140
SetXmlProperty.....	141
SetZoom .....	142
TableGetColumn .....	142
TableSetColumn.....	143
UpdateWindow .....	143

VerifyInputs.....	144
<b>ANHANG A Schalter für Datenzugriffsfunktionen</b>	<b>145</b>
<b>ANHANG B Wildcards für Seiten- und Objektnamen</b>	<b>148</b>
<b>ANHANG C Formularmakros</b>	<b>149</b>
<b>ANHANG D Objektmakros</b>	<b>150</b>
<b>ANHANG E Funktionsnamen (Deutsch – Englisch)</b>	<b>151</b>
<b>ANHANG F Funktionsnamen (Englisch – Deutsch)</b>	<b>154</b>
<b>ANHANG G Format-Deskriptoren</b>	<b>157</b>
<b>ANHANG H Veraltete Funktionen</b>	<b>164</b>

## Funktionsbeschreibungen

<b>AccObjDeleteRecord</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion löscht den augenblicklich aktiven Datensatz in einer Datenquelle, auf die mittels eines Datenzugriffsobjekts zugegriffen wird. In der Regel ist dies eine ODBC-Datenbank.</p> <p><b>Funktionswert</b>  <b>0</b> Die Funktion wurde fehlerfrei ausgeführt  <b>-1</b> Es trat ein Fehler auf</p> <p><b>Verwandte Funktionen</b>  <i>AccObjInsertRecord</i>  <i>AccObjInsertEmptyRecord</i>  <i>AccObjUpdateRecord</i></p>	<p><b>Handle</b> Ganzzahlige Variable, die den <i>Handle</i> des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der <i>Handle</i> eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>Flags</b> Ganzzahlige Konstante oder Variable. Hier können verschiedene Schalter angegeben werden.</p> <p>Alle möglichen Schalter sind in Anhang A zusammengefasst.</p> <p>Es können mehrere Schalter durch Addition kombiniert werden.</p>
<b>AccObjGetCurrentRecord</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt die Nummer des augenblicklich aktiven Datensatz in einer Datenquelle, auf die mittels eines Datenzugriffsobjekts zugegriffen wird.</p> <p><b>Funktionswert</b>  <b>n</b> Nummer des Satzes (zählweise ab Null)  <b>-1</b> Es trat ein Fehler auf</p> <p><b>Verwandte Funktionen</b>  <i>AccObjGetTotalRecords</i>  <i>FindObject</i></p>	<p><b>Handle</b> Ganzzahlige Variable, die den <i>Handle</i> des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der <i>Handle</i> eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p>

<b>AccObjGetChangeFlag</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt, ob der augenblicklich aktive Datensatz eines Datenzugriffsobjekts verändert wurde oder nicht.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"> <li><b>1</b> Der Datensatz wurde verändert</li> <li><b>0</b> Der Datensatz wurde nicht verändert</li> </ul> <p><b>Bemerkungen</b> Ein Satz wird als <i>verändert</i> gekennzeichnet, wenn irgendein mit dem Datenzugriffsobjekt verbundenes Eingabefeld oder die Zelle einer Tabelle verändert wird.</p> <p><b>Verwandte Funktionen</b> <i>AccObjSetChangeFlag</i></p>	<p><b>Handle</b></p> <p>Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p>

<b>AccObjGetMatchMode</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt, ob sich ein Datenzugriffsobjekt augenblicklich in der Matchmode-Eingabe befindet oder nicht.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"> <li><b>1</b> Matchmode-Eingabe ist aktiv</li> <li><b>0</b> Matchmode-Eingabe ist nicht aktiv</li> </ul> <p><b>Verwandte Funktionen</b> <i>AccObjMatchModeCancel</i> <i>AccObjMatchModeEnter</i></p>	<p><b>Handle</b></p> <p>Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p>



<b>AccObjGetFieldAsDate</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt den Inhalt eines Datenfeldes einer Datenquelle, auf die mittels eines Datenzugriffsobjekts zugegriffen wird als Datum.</p> <p><b>Funktionswert</b>  <b>0</b> Inhalt des Datenfeldes zurückgeliefert  <b>-1</b> Es sind keine Daten verfügbar</p> <p><b>Bemerkungen</b> Die Funktion kann auf den Inhalt eines beliebigen Feldes eines beliebigen Datensatzes der augenblicklich aktiven Selektion zugreifen.</p> <p><b>Verwandte Funktionen</b>  <i>AccObjGetFieldAsText</i>  <i>AccObjGetFieldAsInt</i>  <i>AccObjGetFieldAsFloat</i></p>	<p><b>Handle</b> Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p><b>FieldName</b> Textkonstante oder -variable, die den Namen des Feldes angibt, auf das zugegriffen werden soll.</p> <p><b>Record</b> Ganzzahlige Konstante oder Variable, die die Nummer des Datensatzes (ab Null) innerhalb der augenblicklichen Selektion angibt, auf den zugegriffen werden soll.</p> <p><b>-1</b> greift auf den augenblicklich aktiven Datensatz zu.</p> <p><b>Result</b> Textvariable, keine Konstante, auf der das Ergebnis in der Form <i>TT.MM.JJJJ</i> als Zeichenfolge abgelegt wird.</p>

<b>AccObjGetFieldAsFloat</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt den Inhalt eines Datenfeldes einer Datenquelle, auf die mittels eines Datenzugriffsobjekts zugegriffen wird als <i>Gleitkommazahl</i>.</p> <p><b>Funktionswert</b>  <b>0</b> Inhalt des Datenfeldes zurückgeliefert  <b>-1</b> Es sind keine Daten verfügbar</p> <p><b>Bemerkungen</b> Die Funktion kann auf den Inhalt eines beliebigen Feldes eines beliebigen Datensatzes der augenblicklich aktiven Selektion zugreifen.</p> <p><b>Verwandte Funktionen</b>  <i>AccObjGetFieldAsText</i>  <i>AccObjGetFieldAsInt</i>  <i>AccObjGetFieldAsDate</i>  <i>AccObjSetFieldAsFloat</i></p>	<p><b>Handle</b> Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>FieldName</b> Textkonstante oder -variable, die den Namen des Feldes angibt, auf das zugegriffen werden soll.</p> <p><b>Record</b> Ganzzahlige Konstante oder Variable, die die Nummer des Datensatzes (ab Null) innerhalb der augenblicklichen Selektion angibt, auf den zugegriffen werden soll.</p> <p><b>-1</b> greift auf den augenblicklich aktiven Datensatz zu.</p> <p><b>Result</b> Gleitkommavariablen, keine Konstante, auf der das Ergebnis abgelegt wird.</p>

<b>AccObjGetFieldAsInt</b>	
<p><b>Funktionsbeschreibung</b>                      Diese Funktion ermittelt den Inhalt eines Datenfeldes einer Datenquelle, auf die mittels eines Datenzugriffsobjekts zugegriffen wird als <i>ganze Zahl</i>.</p> <p><b>Funktionswert</b>                      0 Inhalt des Datenfeldes zurückgeliefert                      -1 Es sind keine Daten verfügbar</p> <p><b>Bemerkungen</b>                      Die Funktion kann auf den Inhalt eines beliebigen Feldes eines beliebigen Datensatzes der augenblicklich aktiven Selektion zugreifen.</p> <p><b>Verwandte Funktionen</b>  <i>AccObjGetFieldAsText</i>  <i>AccObjGetFieldAsFloat</i>  <i>AccObjGetFieldAsDate</i>  <i>AccObjSetFieldAsInt</i></p>	<p><b>Handle</b>                      Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>FieldName</b>                      Textkonstante oder -variable, die den Namen des Feldes angibt, auf das zugegriffen werden soll.</p> <p><b>Record</b>                      Ganzzahlige Konstante oder Variable, die die Nummer des Datensatzes (ab Null) innerhalb der augenblicklichen Selektion angibt, auf den zugegriffen werden soll.</p> <p>-1 greift auf den augenblicklich aktiven Datensatz zu.</p> <p><b>Result</b>                      Ganzzahlige Variable, keine Konstante, auf der das Ergebnis abgelegt wird.</p>

<b>AccObjGetFieldAsText</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt den Inhalt eines Datenfeldes einer Datenquelle, auf die mittels eines Datenzugriffsobjekts zugegriffen wird als <i>Text</i>.</p> <p><b>Funktionswert</b> <b>0</b> Inhalt des Datenfeldes zurückgeliefert <b>-1</b> Es sind keine Daten verfügbar</p> <p><b>Bemerkungen</b> Die Funktion kann auf den Inhalt eines beliebigen Feldes eines beliebigen Datensatzes der augenblicklich aktiven Selektion zugreifen.</p> <p><b>Verwandte Funktionen</b> <i>AccObjGetFieldAsFloat</i> <i>AccObjGetFieldAsInt</i> <i>AccObjGetFieldAsDate</i> <i>AccObjSetFieldAsText</i></p>	<p><b>Handle</b> Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>Feldname</b> Textkonstante oder -variable, die den Namen des Feldes angibt, auf das zugegriffen werden soll.</p> <p><b>Record</b> Ganzzahlige Konstante oder Variable, die die Nummer des Datensatzes (ab Null) innerhalb der augenblicklichen Selektion angibt, auf den zugegriffen werden soll.</p> <p><b>-1</b> greift auf den augenblicklich aktiven Datensatz zu.</p> <p><b>Result</b> Textvariable, keine Konstante, auf der das Ergebnis abgelegt wird.</p>

<b>AccObjGetTotalRecords</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt die Anzahl von Datensätzen, die in der augenblicklichen Selektion einer Datenquelle verfügbar sind, auf die mittels eines Datenzugriffsobjekts zugegriffen wird.</p> <p><b>Funktionswert</b> <b>n</b> Anzahl der Sätze (0 = keine Sätze vorhanden)</p> <p><b>-1</b> Es trat ein Fehler auf</p> <p><b>Bemerkungen</b> Wenn alle Sätze der Datenquelle selektiert sind, liefert diese Funktion die Gesamtzahl der Datensätze in der Datenquelle zurück.</p> <p><b>Verwandte Funktionen</b> <i>AccObjGetTotalRecords</i> <i>FindObject</i></p>	<p><b>Handle</b> Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p>

<b>AccObjInsertEmptyRecord</b>	
<p><b>Funktionsbeschreibung</b>                      Diese Funktion fügt einen neuen, leeren Datensatz in eine Datenquelle ein, auf die mittels eines Datenzugriffsobjekts zugegriffen wird. In der Regel ist dies eine ODBC-Datenbank.</p> <p><b>Funktionswert</b>                      0 Funktion wurde fehlerfrei ausgeführt                      -1 Es trat ein Fehler auf</p> <p><b>Bemerkungen</b>                      Wenn bei Aufruf der Funktion keine Sätze der Datenquelle selektiert sind, werden vor Einfügen des Datensatzes automatisch alle Sätze der Datenquelle selektiert.</p> <p>Die Position des neu eingefügten Satzes innerhalb der Datenquelle ist abhängig vom Typ der Datenquelle und vom Treiber der Datenquelle. In der Regel wird der Satz am Ende der augenblicklichen Selektion angefügt.</p> <p>Achtung: Wenn die Datenquelle Felder enthält, die einen Wert enthalten <i>müssen</i>, führt diese Funktion zu einem Fehler.</p> <p><b>Verwandte Funktionen</b>  <i>AccObjInsertRecord</i>  <i>AccObjDeleteRecord</i>  <i>AccObjUpdateRecord</i></p>	<p><b>Handle</b>                      Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>Flags</b>                      Ganzzahlige Konstante oder Variable. Hier können verschiedene Schalter angegeben werden.</p> <p>Alle möglichen Schalter sind in Anhang A zusammengefasst.</p> <p>Es können mehrere Schalter durch Addition kombiniert werden.</p>

<b>AccObjInsertRecord</b>	
<p><b>Funktionsbeschreibung</b>                      Diese Funktion fügt einen neuen Datensatz in eine Datenquelle ein, auf die mittels eines Datenzugriffsobjekts zugegriffen wird. In der Regel ist dies eine ODBC-Datenbank.</p> <p>Der Inhalt der Datenfelder des neuen Datensatzes wird durch den Inhalt der zugehörigen Eingabefelder des Formulars bestimmt.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"> <li>0 Funktion wurde fehlerfrei ausgeführt</li> <li>-1 Es trat ein Fehler auf</li> </ul> <p><b>Bemerkungen</b>                      Achtung: Wenn mehrere Eingabefelder des Formulars an ein und dasselbe Datenfeld gebunden sind, ist nicht eindeutig, welcher der Eingabewerte in die Datenquelle geschrieben wird.</p> <p>Die Position des neu eingefügten Satzes innerhalb der Datenquelle ist abhängig vom Typ der Datenquelle und vom Treiber der Datenquelle. In der Regel wird der Satz am Ende der augenblicklichen Selektion angefügt.</p> <p><b>Verwandte Funktionen</b>  <i>AccObjInsertEmptyRecord</i>  <i>AccObjDeleteRecord</i>  <i>AccObjUpdateRecord</i></p>	<p><b>Handle</b>                      Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>Flags</b>                      Ganzzahlige Konstante oder Variable. Hier können verschiedene Schalter angegeben werden.</p> <p>Alle möglichen Schalter sind in Anhang A zusammengefasst.</p> <p>Es können mehrere Schalter durch Addition kombiniert werden.</p>

<b>AccObjMatchModeCancel</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion bricht die Eingabe von Matchcodes in einem Datenzugriffsobjekt bzw. in den damit verbundenen Eingabefeldern des Formulars ab, ohne einen Matchcode zu setzen und damit eine neue Selektion auszulösen.</p> <p><b>Funktionswert</b> 0 Funktion wurde fehlerfrei ausgeführt -1 Es trat ein Fehler auf</p> <p><b>Verwandte Funktionen</b> <i>AccObjMatchModeEnter</i> <i>AccObjMatchModeConfirm</i> <i>AccObjMatchModeToggle</i></p>	<p><b>Handle</b> Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p><b>Flags</b> Ganzzahlige Konstante oder Variable. Hier können verschiedene Schalter angegeben werden.</p> <p>Alle möglichen Schalter sind in Anhang A zusammengefasst.</p> <p>Es können mehrere Schalter durch Addition kombiniert werden.</p>

<b>AccObjMatchModeConfirm</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion bestätigt die Eingabe von Matchcodes in einem Datenzugriffsobjekt bzw. in den damit verbundenen Eingabefeldern des Formulars. Es wird eine neue Selektionsbedingung aus den Matchcodes konstruiert und eine neue Selektion durchgeführt.</p> <p><b>Funktionswert</b> 0 Funktion wurde fehlerfrei ausgeführt -1 Es trat ein Fehler auf</p> <p><b>Verwandte Funktionen</b> <i>AccObjMatchModeEnter</i> <i>AccObjMatchModeCancel</i> <i>AccObjMatchModeToggle</i></p>	<p><b>Handle</b> Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p><b>Flags</b> Ganzzahlige Konstante oder Variable. Hier können verschiedene Schalter angegeben werden.</p> <p>Alle möglichen Schalter sind in Anhang A zusammengefasst.</p> <p>Es können mehrere Schalter durch Addition kombiniert werden.</p>



<b>AccObjMatchModeEnter</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion schaltet die innerhalb eines Datenzugriffsobjekts bzw. in den damit verbundenen Eingabefeldern des Formulars in die Eingabe von <i>Matchcodes</i> um.</p> <p><b>Funktionswert</b> 0 Funktion wurde fehlerfrei ausgeführt -1 Es trat ein Fehler auf</p> <p><b>Verwandte Funktionen</b> <i>AccObjMatchModeConfirm</i> <i>AccObjMatchModeCancel</i> <i>AccObjMatchModeToggle</i></p>	<p><b>Handle</b> Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>Flags</b> Ganzzahlige Konstante oder Variable. Hier können verschiedene Schalter angegeben werden.</p> <p>Alle möglichen Schalter sind in Anhang A zusammengefasst.</p> <p>Es können mehrere Schalter durch Addition kombiniert werden.</p>

<b>AccObjMatchModeToggle</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion schaltet in einem Datenzugriffsobjekt bzw. in den damit verbundenen Eingabefeldern eines Formulars die Eingabe von <i>Matchcodes</i> ein oder aus.</p> <p>Wenn die Matchcode-Eingabe ausgeschaltet ist, wird sie eingeschaltet - und umgekehrt.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"><li>0 Funktion wurde fehlerfrei ausgeführt</li><li>-1 Es trat ein Fehler auf</li></ul> <p><b>Verwandte Funktionen</b></p> <ul style="list-style-type: none"><li><i>AccObjMatchModeEnter</i></li><li><i>AccObjMatchModeCancel</i></li><li><i>AccObjMatchModeConfirm</i></li></ul>	<p><b>Handle</b></p> <p>Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjektes kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>Flags</b></p> <p>Ganzzahlige Konstante oder Variable. Hier können verschiedene Schalter angegeben werden.</p> <p>Alle möglichen Schalter sind in Anhang A zusammengefasst.</p> <p>Es können mehrere Schalter durch Addition kombiniert werden.</p>

<b>AccObjNavigateAbsolute</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion macht einen beliebigen Datensatz einer Datenquelle, auf die mittels eines Datenzugriffsobjekts zugegriffen wird, zum aktuellen Datensatz.</p> <p>Der Inhalt der Datenfelder des neuen aktuellen Datensatzes wird in die an die Datenquelle gebundenen Eingabefelder des Formulars transportiert.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"><li><b>0</b> Funktion wurde fehlerfrei ausgeführt</li><li><b>-1</b> Es trat ein Fehler auf</li></ul> <p><b>Verwandte Funktionen</b></p> <p><i>AccObjNavigateNext</i> <i>AccObjNavigatePrev</i> <i>FindObject</i></p>	<p><b>Handle</b></p> <p>Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>Record</b></p> <p>Ganzzahlige Konstante oder Variable, die den Datensatz der Selektion bestimmt, auf den positioniert werden soll. Die Zählung beginnt ab 0.</p> <p><b>Flags</b></p> <p>Ganzzahlige Konstante oder Variable. Hier können verschiedene Schalter angegeben werden.</p> <p>Alle möglichen Schalter sind in Anhang A zusammengefasst.</p> <p>Es können mehrere Schalter durch Addition kombiniert werden.</p>

<b>AccObjNavigateFirst</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion macht den ersten Datensatz der Selektion einer Datenquelle, auf die mittels eines Datenzugriffsobjekts zugegriffen wird, zum aktuellen Datensatz.</p> <p>Der Inhalt der Datenfelder des neuen aktuellen Datensatzes wird in die an die Datenquelle gebundenen Eingabefelder des Formulars transportiert.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"><li><b>0</b> Funktion wurde fehlerfrei ausgeführt</li><li><b>-1</b> Es trat ein Fehler auf</li></ul> <p><b>Verwandte Funktionen</b></p> <p><i>AccObjNavigateNext</i> <i>AccObjNavigatePrev</i> <i>FindObject</i></p>	<p><b>Handle</b></p> <p>Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>Flags</b></p> <p>Ganzzahlige Konstante oder Variable. Hier können verschiedene Schalter angegeben werden.</p> <p>Alle möglichen Schalter sind in Anhang A zusammengefasst.</p> <p>Es können mehrere Schalter durch Addition kombiniert werden.</p>

<b>AccObjNavigateLast</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion macht den letzten Datensatz der Selektion einer Datenquelle, auf die mittels eines Datenzugriffsobjekts zugegriffen wird, zum aktuellen Datensatz.</p> <p>Der Inhalt der Datenfelder des neuen aktuellen Datensatzes wird in die an die Datenquelle gebundenen Eingabefelder des Formulars transportiert.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"><li>0 Funktion wurde fehlerfrei ausgeführt</li><li>-1 Es trat ein Fehler auf</li></ul> <p><b>Bemerkungen</b> Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>Verwandte Funktionen</b> <i>AccObjNavigateNext</i> <i>AccObjNavigatePrev</i> <i>FindObject</i></p>	<p><b>Handle</b> Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>Flags</b> Ganzzahlige Konstante oder Variable. Hier können verschiedene Schalter angegeben werden.</p> <p>Alle möglichen Schalter sind in Anhang A zusammengefasst.</p> <p>Es können mehrere Schalter durch Addition kombiniert werden.</p>

<b>AccObjNavigateNext</b>	
<p><b>Funktionsbeschreibung</b>                      Diese Funktion macht den nächsten Datensatz der Selektion einer Datenquelle, auf die mittels eines Datenzugriffsobjekts zugegriffen wird, zum aktuellen Datensatz.</p> <p>Der Inhalt der Datenfelder des neuen aktuellen Datensatzes wird in die an die Datenquelle gebundenen Eingabefelder des Formulars transportiert.</p> <p><b>Funktionswert</b>  <b>0</b> Funktion wurde fehlerfrei ausgeführt  <b>-1</b> Es trat ein Fehler auf</p> <p><b>Verwandte Funktionen</b>  <i>AccObjNavigateAbsolute</i>  <i>AccObjNavigatePrev</i>  <i>FindObject</i></p>	<p><b>Handle</b>                      Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>Flags</b>                      Ganzzahlige Konstante oder Variable. Hier können verschiedene Schalter angegeben werden.</p> <p>Alle möglichen Schalter sind in Anhang A zusammengefasst.</p> <p>Es können mehrere Schalter durch Addition kombiniert werden.</p>

<b>AccObjNavigatePrev</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion macht den vorigen Datensatz der Selektion einer Datenquelle, auf die mittels eines Datenzugriffsobjekts zugegriffen wird, zum aktuellen Datensatz.</p> <p>Der Inhalt der Datenfelder des neuen aktuellen Datensatzes wird in die an die Datenquelle gebundenen Eingabefelder des Formulars transportiert.</p> <p><b>Funktionswert</b> 0 Funktion wurde fehlerfrei ausgeführt -1 Es trat ein Fehler auf</p> <p><b>Verwandte Funktionen</b> <i>AccObjNavigateNext</i> <i>AccObjNavigateAbsolute</i> <i>FindObject</i></p>	<p><b>Handle</b> Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>Flags</b> Ganzzahlige Konstante oder Variable. Hier können verschiedene Schalter angegeben werden.</p> <p>Alle möglichen Schalter sind in Anhang A zusammengefasst.</p> <p>Es können mehrere Schalter durch Addition kombiniert werden.</p>

<b>AccObjReloadObjects</b>	
<p><b>Funktionsbeschreibung</b>                      Diese Funktion füllt alle Eingabeobjekte (Eingabefelder, Auswahlfelder, Tabellen) eines Formulars, die an ein Datenzugriffsobjekt und somit an eine Datenquelle gebunden sind, mit den Daten, die in der augenblicklichen Selektion der Datenquelle bzw. in der Datenquelle selbst zur Verfügung stehen.</p> <p><b>Funktionswert</b>  <b>0</b> Funktion wurde fehlerfrei ausgeführt  <b>-1</b> Es trat ein Fehler auf</p> <p><b>Bemerkungen</b>                      Diese Funktion wird insbesondere dann aufgerufen, wenn die Datenquelle zuvor verändert wurde. Dies kann z.B. mittels der Funktion <i>AccObjSetFieldAsText</i> geschehen.</p> <p><b>Verwandte Funktionen</b>  <i>FindObject</i></p>	<p><b>Handle</b>                      Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjektes kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>Flags</b>                      Ganzzahlige Konstante oder Variable. Hier können verschiedene Schalter angegeben werden.</p> <p>Alle möglichen Schalter sind in Anhang A zusammengefasst.</p> <p>Es können mehrere Schalter durch Addition kombiniert werden.</p>



<b>AccObjSelect</b>	
<p><b>Funktionsbeschreibung</b>                      Diese Funktion führt eine erneute Selektion der Datenquelle durch, auf die mittels eines Datenzugriffsobjekts zugegriffen wird und macht den ersten Satz der Selektion zum aktuellen Satz.</p> <p>Anschließend werden alle Eingabeobjekte (Eingabefelder, Auswahlfelder, Tabellen) des Formulars, die an das Datenzugriffsobjekt gebunden sind, mit den Daten der augenblicklichen Selektion gefüllt.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"> <li><b>0</b> Funktion wurde fehlerfrei ausgeführt</li> <li><b>-1</b> Es trat ein Fehler auf</li> </ul> <p><b>Verwandte Funktionen</b></p> <p><i>FindObject</i>  <i>AccObjSelectAll</i>  <i>AccObjSelectLead</i>  <i>AccObjSelectExact</i>  <i>AccObjSelectStdMask</i></p>	<p><b>Handle</b></p> <p>Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>Flags</b></p> <p>Ganzzahlige Konstante oder Variable. Hier können verschiedene Schalter angegeben werden.</p> <p>Alle möglichen Schalter sind in Anhang A zusammengefasst.</p> <p>Es können mehrere Schalter durch Addition kombiniert werden.</p>

<b>AccObjSelectAll</b>	
<p><b>Funktionsbeschreibung</b>                      Diese Funktion löscht die augenblicklich in einem Datenzugriffsobjekt eingestellte WHERE-Klausel und führt eine Neuselektion aller Datensätze der Datenquelle durch, die in dem Datenzugriffsobjekt eingestellt ist.</p> <p>Der erste Satz der Selektion wird zum aktuellen Satz. Alle Eingabeobjekte (Eingabefelder, Auswahlfelder, Tabellen, Bilder) des Formulars, die an das Datenzugriffsobjekt gebunden sind, werden mit den Daten des ersten Satzes gefüllt.</p> <p><b>Funktionswert</b>  <b>0</b> Funktion wurde fehlerfrei ausgeführt  <b>-1</b> Es trat ein Fehler auf</p> <p><b>Verwandte Funktionen</b>  <i>FindObject</i>  <i>AccObjSelect</i>  <i>AccObjSelectLead</i>  <i>AccObjSelectExact</i>  <i>AccObjSelectStdMask</i></p>	<p><b>Handle</b>                      Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>Flags</b>                      Ganzzahlige Konstante oder Variable. Hier können verschiedene Schalter angegeben werden.</p> <p>Alle möglichen Schalter sind in Anhang A zusammengefasst.</p> <p>Es können mehrere Schalter durch Addition kombiniert werden.</p>

<b>AccObjSelectExact</b>	
<p><b>Funktionsbeschreibung</b>                      Diese Funktion erstellt eine neue WHERE-Klausel für ein Datenzugriffsobjekt, welche alle Sätze selektiert, deren Feldinhalt mit dem im augenblicklich aktiven Eingabefeld befindlichen Text <i>beginnt</i>. Danach wird eine Neuselektion auf Basis dieser WHERE-Klausel durchgeführt.</p> <p>Der erste Satz der Selektion wird zum aktuellen Satz. Alle Eingabeobjekte (Eingabefelder, Auswahlfelder, Tabellen, Bilder) des Formulars, die an das Datenzugriffsobjekt gebunden sind, werden mit den Daten des ersten Satzes gefüllt.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"> <li>0 Funktion wurde fehlerfrei ausgeführt</li> <li>-1 Es trat ein Fehler auf</li> </ul> <p><b>Bemerkungen</b>                      Wenn das augenblicklich aktive Eingabeobjekt nicht zu dem angesprochenen Datenzugriffsobjekt gehört, ist diese Funktion ohne Wirkung und es ertönt ein Piepen.</p> <p><b>Verwandte Funktionen</b>  <i>FindObject</i>  <i>AccObjSelect</i>  <i>AccObjSelectLead</i>  <i>AccObjSelectAll</i>  <i>AccObjSelectStdMask</i></p>	<p><b>Handle</b>                      Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>Flags</b>                      Ganzzahlige Konstante oder Variable. Hier können verschiedene Schalter angegeben werden.</p> <p>Alle möglichen Schalter sind in Anhang A zusammengefasst.</p> <p>Es können mehrere Schalter durch Addition kombiniert werden.</p>

<b>AccObjSelectLead</b>	
<p><b>Funktionsbeschreibung</b>                      Diese Funktion erstellt eine neue WHERE-Klausel für ein Datenzugriffsobjekt, welche alle Sätze selektiert, deren Feldinhalt mit dem im augenblicklich aktiven Eingabefeld befindlichen Text <i>übereinstimmen</i>. Danach wird eine Neuselektion auf Basis dieser WHERE-Klausel durchgeführt.</p> <p>Der erste Satz der Selektion wird zum aktuellen Satz. Alle Eingabeobjekte (Eingabefelder, Auswahlfelder, Tabellen, Bilder) des Formulars, die an das Datenzugriffsobjekt gebunden sind, werden mit den Daten des ersten Satzes gefüllt.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"> <li>0 Funktion wurde fehlerfrei ausgeführt</li> <li>-1 Es trat ein Fehler auf</li> </ul> <p><b>Bemerkungen</b>                      Wenn das augenblicklich aktive Eingabeobjekt nicht zu dem angesprochenen Datenzugriffsobjekt gehört, ist diese Funktion ohne Wirkung und es ertönt ein Piepen.</p> <p><b>Verwandte Funktionen</b>  <i>FindObject</i>  <i>AccObjSelect</i>  <i>AccObjSelectAll</i>  <i>AccObjSelectExact</i>  <i>AccObjSelectStdMask</i></p>	<p><b>Handle</b>                      Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>Flags</b>                      Ganzzahlige Konstante oder Variable. Hier können verschiedene Schalter angegeben werden.</p> <p>Alle möglichen Schalter sind in Anhang A zusammengefasst.</p> <p>Es können mehrere Schalter durch Addition kombiniert werden.</p>

<b>AccObjSelectStdMask</b>	
<p><b>Funktionsbeschreibung</b>                      Diese Funktion öffnet einen Dialog, der die Eingabe von einer oder mehreren Suchbedingungen ermöglicht, die durch <i>und</i> oder <i>oder</i> verknüpft werden können. Aus den eingegebenen Bedingungen wird eine WHERE-Klausel für das angesprochene Datenzugriffsobjekt erstellt. Danach wird eine Neuselektion auf Basis dieser WHERE-Klausel durchgeführt.</p> <p>Der erste Satz der Selektion wird zum aktuellen Satz. Alle Eingabeobjekte (Eingabefelder, Auswahlfelder, Tabellen, Bilder) des Formulars, die an das Datenzugriffsobjekt gebunden sind, werden mit den Daten des ersten Satzes gefüllt.</p> <p><b>Funktionswert</b>                      0 Funktion wurde fehlerfrei ausgeführt                      -1 Es trat ein Fehler auf</p> <p><b>Verwandte Funktionen</b>  <i>FindObject</i>  <i>AccObjSelectAll</i>  <i>AccObjSelectLead</i>  <i>AccObjSelectExact</i>  <i>AccObjSelect</i></p>	<p><b>Handle</b>                      Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>Flags</b>                      Ganzzahlige Konstante oder Variable. Hier können verschiedene Schalter angegeben werden.</p> <p>Alle möglichen Schalter sind in Anhang A zusammengefasst.</p> <p>Es können mehrere Schalter durch Addition kombiniert werden.</p>

<b>AccObjSetChangeFlag</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion manipuliert die Merkvariable eines Datenzugriffsobjekts, welche nachhält, ob der augenblicklich aktive Datensatz verändert wurde oder nicht.</p> <p><b>Funktionswert</b> <b>0</b> Funktion wurde fehlerfrei ausgeführt <b>-1</b> Es trat ein Fehler auf</p> <p><b>Bemerkungen</b> Ein Satz wird als <i>verändert</i> gekennzeichnet, wenn irgendein mit dem Datenzugriffsobjekt verbundenes Eingabefeld oder die Zelle einer Tabelle verändert wird.</p> <p><b>Verwandte Funktionen</b> <i>AccObjGetChangeFlag</i></p>	<p><b>Handle</b> Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>Value</b> Ganzzahlige Konstante oder Variable, die das neue <i>ChangeFlag</i> angibt:</p> <p><b>0</b> Satz nicht verändert <b>1</b> Satz verändert</p>

<b>AccObjSetDataSource</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion stellt für ein Datenzugriffsobjekt, welches auf ODBC-Datenbanken zugreift, eine neue Datenquelle ein und stellt eine Verbindung mit der Datenquelle her.</p> <p><b>Funktionswert</b>  <b>0</b> Funktion wurde fehlerfrei ausgeführt  <b>-1</b> Es trat ein Fehler auf</p> <p><b>Bemerkungen</b> Die neue Datenquelle wird nicht unmittelbar selektiert. Dies muss explizit erfolgen, z.B. durch Aufruf einer der Funktionen <i>AccObjSelect</i> oder <i>AccObjSetWhereClause</i>.</p> <p><b>Verwandte Funktionen</b>  <i>OdbcAccSelec</i>  <i>OdbcAccSetWhereClause</i></p>	<p><b>Handle</b> Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>DataSourceObjectName</b> Textkonstante oder -variable, die den Namen der neu einzustellenden ODBC-Datenquelle angibt.</p> <p><b>Password</b> Textkonstante oder -variable, die ein eventuell zum Öffnen der Datenquelle benötigtes Passwort angibt. Wenn kein Paßwort benötigt wird, kann ein Leerstring "" angegeben werden.</p> <p><b>UserId</b> Textkonstante oder -variable, die ein eventuell zum Öffnen der Datenquelle benötigte Benutzerkennung angibt. Wenn keine Benutzerkennung benötigt wird, kann ein Leerstring "" angegeben werden.</p> <p><b>Flags</b> Ganzzahlige Konstante oder Variable. Ist für spätere Zwecke reserviert und muss 0 sein.</p>

<b>AccObjSetFieldAsDate</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ändert den Inhalt eines Datenfelds einer Datenquelle, auf die mittels eines Datenzugriffsobjekts zugegriffen wird. Die Übergabe des Wertes erfolgt als Datum.</p> <p><b>Funktionswert</b>  <b>0</b> Inhalt des Datenfeldes wurde verändert  <b>-1</b> Es ist ein Fehler aufgetreten</p> <p><b>Bemerkungen</b> Die Funktion kann auf den Inhalt eines beliebigen Feldes eines beliebigen Datensatzes der augenblicklich aktiven Selektion zugreifen.</p> <p><b>Verwandte Funktionen</b>  <i>AccObjSetFieldAsText</i>  <i>AccObjSetFieldAsInt</i>  <i>AccObjSetFieldAsFloat</i>  <i>AccObjGetFieldAsDate</i></p>	<p><b>Handle</b> Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>FieldName</b> Textkonstante oder -variable, die den Namen des Feldes angibt, auf das zugegriffen werden soll.</p> <p><b>Record</b> Ganzzahlige Konstante oder Variable, die die Nummer des Datensatzes (ab Null) innerhalb der augenblicklichen Selektion angibt, auf den zugegriffen werden soll.</p> <p><b>-1</b> greift auf den augenblicklich aktiven Datensatz zu.</p> <p><b>NewValue</b> Textkonstante oder -variable, auf der neue Wert als Datumszeichenfolge in der Form <i>TT.MM.JJJJ</i> übergeben wird.</p>



<b>AccObjSetFieldAsFloat</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ändert den Inhalt eines Datenfelds einer Datenquelle, auf die mittels eines Datenzugriffsobjekts zugegriffen wird. Die Übergabe des Wertes erfolgt als <i>Gleitkommazahl</i>.</p> <p><b>Funktionswert</b>  <b>0</b> Inhalt des Datenfeldes wurde verändert  <b>-1</b> Es ist ein Fehler aufgetreten</p> <p><b>Bemerkungen</b> Die Funktion kann auf den Inhalt eines beliebigen Feldes eines beliebigen Datensatzes der augenblicklich aktiven Selektion zugreifen.</p> <p><b>Verwandte Funktionen</b>  <i>AccObjSetFieldAsText</i>  <i>AccObjSetFieldAsInt</i>  <i>AccObjSetFieldAsDate</i>  <i>AccObjGetFieldAsFloat</i></p>	<p><b>Handle</b> Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>FieldName</b> Textkonstante oder -variable, die den Namen des Feldes angibt, auf das zugegriffen werden soll.</p> <p><b>Record</b> Ganzzahlige Konstante oder Variable, die die Nummer des Datensatzes (ab Null) innerhalb der augenblicklichen Selektion angibt, auf den zugegriffen werden soll.</p> <p><b>-1</b> greift auf den augenblicklich aktiven Datensatz zu.</p> <p><b>NewValue</b> Gleitkommakonstante oder -variable, auf der neue Wert übergeben wird.</p>

<b>AccObjSetFieldAsInt</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ändert den Inhalt eines Datenfelds einer Datenquelle, auf die mittels eines Datenzugriffsobjekts zugegriffen wird. Die Übergabe des Wertes erfolgt als <i>ganze Zahl</i>.</p> <p><b>Funktionswert</b>  <b>0</b> Inhalt des Datenfeldes wurde verändert  <b>-1</b> Es ist ein Fehler aufgetreten</p> <p><b>Bemerkungen</b> Die Funktion kann auf den Inhalt eines beliebigen Feldes eines beliebigen Datensatzes der augenblicklich aktiven Selektion zugreifen.</p> <p><b>Verwandte Funktionen</b>  <i>AccObjSetFieldAsText</i>  <i>AccObjSetFieldAsDate</i>  <i>AccObjSetFieldAsFloat</i>  <i>AccObjGetFieldAsInt</i></p>	<p><b>Handle</b> Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>FieldName</b> Textkonstante oder -variable, die den Namen des Feldes angibt, auf das zugegriffen werden soll.</p> <p><b>Record</b> Ganzzahlige Konstante oder Variable, die die Nummer des Datensatzes (ab Null) innerhalb der augenblicklichen Selektion angibt, auf den zugegriffen werden soll.</p> <p><b>-1</b> greift auf den augenblicklich aktiven Datensatz zu.</p> <p><b>NewValue</b> Ganzzahlige Konstante oder Variable, auf der neue Wert übergeben wird.</p>

<b>AccObjSetFieldAsText</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ändert den Inhalt eines Datenfelds einer Datenquelle, auf die mittels eines Datenzugriffsobjekts zugegriffen wird. Die Übergabe des Wertes erfolgt als Text.</p> <p><b>Funktionswert</b>  <b>0</b> Inhalt des Datenfeldes wurde verändert  <b>-1</b> Es ist ein Fehler aufgetreten</p> <p><b>Bemerkungen</b> Die Funktion kann auf den Inhalt eines beliebigen Feldes eines beliebigen Datensatzes der augenblicklich aktiven Selektion zugreifen.</p> <p><b>Verwandte Funktionen</b>  <i>AccObjSetFieldAsDate</i>  <i>AccObjSetFieldAsInt</i>  <i>AccObjSetFieldAsFloat</i>  <i>AccObjGetFieldAsText</i></p>	<p><b>Handle</b> Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>FieldName</b> Textkonstante oder -variable, die den Namen des Feldes angibt, auf das zugegriffen werden soll.</p> <p><b>Record</b> Ganzzahlige Konstante oder Variable, die die Nummer des Datensatzes (ab Null) innerhalb der augenblicklichen Selektion angibt, auf den zugegriffen werden soll.</p> <p><b>-1</b> greift auf den augenblicklich aktiven Datensatz zu.</p> <p><b>NewValue</b> Textkonstante oder -variable, auf der neue Wert als Text übergeben wird.</p>

<b>AccObjSetOrderByClause</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion verändert die Sortierklausel in der Selektion eines Datenzugriffsobjekts, mit dem auf eine ODBC-Datenbank zugegriffen wird.</p> <p><b>Funktionswert</b> 0 Die Klausel wurde verändert -1 Es ist ein Fehler aufgetreten</p> <p><b>Bemerkungen</b> Nach dem Verändern der Klausel wird unmittelbar eine Neuselektion der Datenquelle mit unveränderten Selektionsbedingungen vorgenommen. Dabei wird der erste Datensatz der neuen Selektion zum aktuellen Datensatz.</p> <p>Die Neuselektion kann durch Angabe des Schalters ACCOBJ_NO_SELECT auch unterbunden werden.</p> <p><b>Verwandte Funktionen</b> <i>AccObjSetWhereClause</i></p>	<p><b>Handle</b> Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>OrderByClause</b> Textkonstante oder -variable, die die neue Sortierklausel angibt. Dies ist einfach der Name des Datenfeldes, wenn nur nach einem Feld aufsteigend sortiert werden soll.</p> <p><b>Flags</b> Ganzzahlige Konstante oder Variable. Hier können verschiedene Schalter angegeben werden.</p> <p>Alle möglichen Schalter sind in Anhang A zusammengefasst.</p> <p>Es können mehrere Schalter durch Addition kombiniert werden.</p>

<b>AccObjSetWhereClause</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion verändert die Selektionsklausel in der Selektion eines Datenzugriffsobjekts, mit dem auf eine ODBC-Datenbank zugegriffen wird.</p> <p><b>Funktionswert</b>  <b>0</b> Die Klausel wurde verändert  <b>-1</b> Es ist ein Fehler aufgetreten</p> <p><b>Bemerkungen</b> Nach dem Verändern der Klausel wird unmittelbar eine Neuselektion der Datenquelle mit unveränderter Sortierung vorgenommen. Dabei wird der erste Datensatz der neuen Selektion zum aktuellen Datensatz.</p> <p>Die Neuselektion kann durch Angabe des Schalters ACCOBJ_NO_SELECT auch unterbunden werden Dies ist insbesondere dann sinnvoll, wenn zusätzlich auch die Sortierklausel verändert werden soll. Es braucht dann nicht mehrfach neu selektiert zu werden.</p> <p><b>Verwandte Funktionen</b> <i>AccObjSetWhereClause</i></p>	<p><b>Handle</b> Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>WhereClause</b> Textkonstante oder -variable, die die neue Selektionsklausel angibt.</p> <p><b>Flags</b> Ganzzahlige Konstante oder Variable. Hier können verschiedene Schalter angegeben werden.</p> <p>Alle möglichen Schalter sind in Anhang A zusammengefasst.</p> <p>Es können mehrere Schalter durch Addition kombiniert werden.</p>

<b>AccObjUnselect</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion beendet die Selektion von Datensätzen in dem angegebenen Datenzugriffsobjekt.</p> <p>Alle Eingabeobjekte (Eingabefelder, Auswahlfelder, Tabellen, Bilder), die mit dem Datenzugriffsobjekt verbunden sind, werden geleert und es kann fortan nicht mehr auf die Datenquelle zugegriffen werden, bis eine neue Selektion durchgeführt worden ist.</p> <p><b>Funktionswert</b> <b>0</b> Die Funktion wurde fehlerfrei ausgeführt <b>-1</b> Es trat ein Fehler auf</p> <p><b>Verwandte Funktionen</b> <i>FindObject</i> <i>AccObjSelectAll</i> <i>AccObjSelectLead</i> <i>AccObjSelectExact</i> <i>AccObjSelect</i></p>	<p><b>Handle</b> Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjektes kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>Flags</b> Ganzzahlige Konstante oder Variable. Hier können verschiedene Schalter angegeben werden.</p> <p>Alle möglichen Schalter sind in Anhang A zusammengefasst.</p> <p>Es können mehrere Schalter durch Addition kombiniert werden.</p>

<b>AccObjUpdateRecord</b>	
<p><b>Funktionsbeschreibung</b>                      Diese Funktion verändert den aktuellen Datensatz, auf den mittels eines Datenzugriffsobjekts zugegriffen wird. In der Regel ist dies eine ODBC-Datenbank.</p> <p>Der Inhalt der Datenfelder des neuen Datensatzes wird durch den Inhalt der zugehörigen Eingabefelder des Formulars bestimmt.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"> <li>0 Funktion wurde fehlerfrei ausgeführt</li> <li>-1 Es trat ein Fehler auf</li> </ul> <p><b>Bemerkungen</b>                      Achtung: Wenn mehrere Eingabefelder des Formulars an ein und dasselbe Datenfeld gebunden sind, ist nicht eindeutig, welcher der Eingabewerte in die Datenquelle geschrieben wird.</p> <p><b>Verwandte Funktionen</b>  <i>AccObjInsertRecord</i>  <i>AccObjDeleteRecord</i></p>	<p><b>Handle</b>                      Ganzzahlige Variable, die den Handle des Zugriffsobjektes angibt, auf das sich der Makrobefehl bezieht.</p> <p>Der Handle eines Datenzugriffsobjekts kann mittels <i>FindObject</i> aus seinem Namen abgeleitet werden.</p> <p><b>Flags</b>                      Ganzzahlige Konstante oder Variable. Hier können verschiedene Schalter angegeben werden.</p> <p>Alle möglichen Schalter sind in Anhang A zusammengefasst.</p> <p>Es können mehrere Schalter durch Addition kombiniert werden.</p>

<b>AddObjects</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion addiert die numerischen Werte einer Gruppe von Objekten und liefert den Saldo als Gleitkommazahl zurück.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkung</b> Es werden alle Objekte addiert, deren Name dem angegebenen Objektnamen entspricht und die auf einer Seite liegen, deren Name dem angegebenen Seitennamen entspricht.  Eingabefelder oder Objekte, die keine numerischen Inhalte haben, werden nicht berücksichtigt.</p>	<p><b>PageName</b> Textkonstante oder -variable, die den Namen der Seiten angibt, deren Objekte addiert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B). Ein leerer Text bezeichnet die aktuelle Seite.</p> <p><b>ObjectName</b> Textkonstante oder -variable, die den Namen der Objekte angibt, die addiert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>Result</b> Gleitkommavariablen, keine Konstante, auf der das Ergebnis als Gleitkommazahl abgespeichert wird.</p>



<b>AppendObjectSetExt</b>	
<p><b>Funktionsbeschreibung</b>                      Diese Funktion fügt während des Ausfüllens eines Formulars die Objekte der ersten Seite eines anderen Formulars hinzu. Die Objekte werden bezüglich der Eingabereihenfolge hinten angehängt.</p> <p><b>Funktionswert</b>                      Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkung</b>                      In Verbindung mit dem Befehl <i>DeleteObjectsInRect</i> mittels dessen Objekte in einem vorgegebenen Bereich eines Formulars auch während des laufenden Betriebs wieder entfernt werden können, kann ein dynamisches Formular angelegt werden, dessen Design und Objektstruktur von bestimmten Situationen abhängig gemacht wird.</p> <p><b>Verwandte Funktionen</b>  <i>DeleteObjectsInRect</i>  <i>DeleteObjectsByName</i></p> <p>Diese Funktion ersetzt die Funktion <i>AppendObjectSet</i></p>	<p><b>FileName</b>                      Textkonstante oder -variable, die den Namen der Formulardatei angibt, aus der die Objekte eingefügt werden sollen.</p> <p>Wenn der Name ohne Pfad angegeben ist, wird die Datei in dem selben Verzeichnis erwartet, in der sich das Formular befindet, in dem das Makro aufgerufen wird.</p> <p><b>PageName</b>                      Textkonstante oder -variable, die den Namen der Seite angibt, auf der die Objekte eingefügt werden sollen.</p> <p><b>x-Offset</b>                      Ganzzahlige Konstante oder Variable. Wenn dieser Wert ungleich Null ist, werden alle eingefügten Objekte in horizontaler Richtung um den angegebenen Wert verschoben. Die Angabe erfolgt in Mikrometern.</p> <p><b>y-Offset</b>                      Ganzzahlige Konstante oder Variable. Wenn dieser Wert ungleich Null ist, werden alle eingefügten Objekte in vertikaler Richtung um den angegebenen Wert verschoben. Die Angabe erfolgt in Mikrometern.</p> <p><b>NamePrefix</b>                      Textkonstante oder -variable, die einen Namensprefix definiert. Dieser wird dem Namen aller eingefügten Objekte vorangestellt, deren Name nicht leer ist.</p> <p><b>NamePostfix</b>                      Textkonstante oder -variable, die einen Namenspostfix definiert. Dieser wird dem Namen aller eingefügten Objekte</p>

	<p>angefügt, deren Name nicht leer ist.</p> <p><b>Flags</b> Muss 0 sein. Ist für spätere Zwecke reserviert.</p>
--	---

<b>BroadcastAllInputs</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion füllt für jedes Objekt des aktuellen Formulars eine Struktur FORM_OBJECT_BROADCAST mit den Daten des Objekts und sendet einen Zeiger auf diese Struktur als Message an ein beliebiges Fenster.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Mittels dieser Funktion kann das Formular anderen Anwendungen Einblick in jegliche Informationen seiner Objekte - insbesondere seiner Eingabeobjekte - gewähren.</p> <p>Die Message, die an das angegebene Fenster gesendet wird, übermittelt den Zeiger auf die Struktur FORM_OBJECT_BROADCAST auf dem Message-Parameter <b>IParam</b>.</p>	<p><b>WindowHandle</b> Ganzzahlige Konstante oder Variable, die den Handle des Fensters angibt, an den die Information gesendet werden soll.</p>

X

<b>CallFormMacro</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ruft eines von mehreren Makros auf, die für das Formular hinterlegt sein können.</p> <p><b>Funktionswert</b> <b>0</b> Funktion wurde fehlerfrei ausgeführt <b>-1</b> Es trat ein Fehler auf</p> <p><b>Bemerkungen</b> Es wird das Makro mit dem angegebenen Makronamen aufgerufen. Ist kein Makro dieses Namens hinterlegt, ist der Aufruf ohne Wirkung.  Der Befehl ruft globale Formularmakros auf. Verwechseln Sie diese nicht mit den Makros, die an einzelne Objekte angeheftet sind.</p> <p><b>Verwandte Funktionen</b> <i>CallObjectMacro</i> <i>PostFormMacro</i> <i>PostObjectMacro</i></p>	<p><b>MacroName</b> Textkonstante oder -variable, die den Namen des Makros angibt, der aufgerufen werden soll.</p> <p>Die Liste aller Makronamen ist im Anhang C (Formularmakros) erläutert.</p>

<b>CallObjectMacro</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ruft eines von mehreren Makros auf, die zu einem Objekt hinterlegt sein können.</p> <p><b>Funktionswert</b>  <b>0</b> Funktion wurde fehlerfrei ausgeführt  <b>-1</b> Es trat ein Fehler auf</p> <p><b>Bemerkungen</b> Es wird das Makro mit dem angegebenen Makronamen aufgerufen. Ist kein Makro dieses Namens hinterlegt, ist der Aufruf ohne Wirkung.  Wenn auf der aktuellen Seite ein solches Objekt nicht gefunden wird, wird die Suche auf das gesamte Formular ausgedehnt.</p> <p><b>Verwandte Funktionen</b>  <i>CallFormMacro</i>  <i>PostFormMacro</i>  <i>PostObjectMacro</i></p>	<p><b>ObjectName</b> Textkonstante oder -variable, die den Namen des Button-Objekts angibt, dessen Makro aufgerufen werden soll.</p> <p><b>MacroName</b> Textkonstante oder -variable, die den Namen des Makros angibt, der aufgerufen werden soll.</p> <p>Die Liste aller Makronamen ist im Anhang D (Objektmakros) erläutert.</p> <p>Es sind nicht alle Makrotypen für alle Objekttypen definiert oder sinnvoll.</p>

<b>Close</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion schließt das augenblicklich aktive Formularfenster. Alle Befehle, die nach diesem Befehl ausgeführt werden, beziehen sich auf das hiernach aktive Fenster.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p>	<p>Die Funktion hat keine Parameter.</p>

<b>CopyText</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion kopiert den Inhalt eines Objekts als Text in ein anderes Objekt.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung</p> <p><b>Bemerkungen</b> Wenn kein Objekt gefunden wird, das den Namen des Quellobjekts hat, bleibt die Funktion ohne Wirkung.</p> <p>Wenn mehr als ein Zielobjekt gefunden wird, welches dem angegebenen Namen oder der angegebenen Wildcard entspricht, werden alle Objekte modifiziert, die mit dem Namen benannt werden.</p>	<p><b>Source</b> Textkonstante oder -variable, die den Namen des Objekts angibt, dessen Text kopiert werden soll.</p> <p><b>Dest</b> Textkonstante oder -variable, die den Namen des Objekts angibt, dessen Text kopiert werden soll. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p>

<b>CountRuntimeChangedPages</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion liefert die Anzahl der Seiten eines Formulars zurück, die zur Laufzeit verändert wurden.</p> <p><b>Funktionswert</b> n Anzahl der veränderten Seiten</p> <p><b>Verwandte Funktionen</b> <i>SetRuntimeChangeFlag</i> <i>GetRuntimeChangeFlag</i></p>	<p>Die Funktion hat keine Parameter</p>

<b>CreateVisitCardFile</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion erzeugt eine Visitenkarte aus dem augenblicklichen Formularinhalt.</p> <p><b>Funktionswert</b>  <b>0</b> Die Funktion wurde fehlerfrei ausgeführt  <b>-1</b> Es trat ein Fehler auf</p> <p><b>Bemerkungen</b> Dieser Makrobefehl bildet die Leistung des Menüeintrags <i>Visitenkarte exportieren</i> ab.</p> <p>Es ist zu beachten, dass allen Feldern, die exportiert werden sollen, die ihnen in der Visitenkarte zugeordnete Bedeutung zugeordnet werden muss.</p>	<p><b>Source</b> Textkonstante oder -variable, die den Namen der Datei angibt, in die exportiert werden soll.</p> <p>Wenn der Dateiname das Synonym %AUTONAME% enthält, wird der Dateiname in der Form:</p> <p style="text-align: center;"><b>Nachname Vorname</b></p> <p>automatisch generiert.</p> <p><b>Dest</b> Reserviert für spätere Zwecke. Hier muss immer eine 0 angegeben werden.</p>
<p><b>Beispiel</b> Die folgende Sequenz erzeugt unter der Voraussetzung, dass das mit der Bedeutung <i>Name</i> gekennzeichnete Eingabefeld den Inhalt <i>Mustermann</i> und das mit der Bedeutung <i>Vorname</i> gekennzeichnete Eingabefeld den Inhalt <i>Karl</i> enthält, die Visitenkartendatei <b>c:\temp\mustermann karl.vcf</b>:</p> <p style="color: green;">CreateVisitCardFile ("c:\\temp\\%AUTONAME%.vcf", 0);</p>	

<b>DeleteObjectsByName</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion löscht alle Objekte auf einer vorgegebenen Seite, die einem Namen oder einem Namensschema entsprechen.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkung</b> In Verbindung mit dem Befehl <i>AppendObjectSetExt</i> mittels dessen Objekte dynamisch in ein Formular eingefügt werden können, kann ein dynamisches Formular angelegt werden, dessen Design und Objektstruktur von bestimmten Situationen abhängig gemacht wird.  <i>Die Funktion kann auch während des Ausfüllens aufgerufen werden!</i></p> <p><b>Verwandte Funktionen</b> <i>DeleteObjectsInRect</i> <i>AppendObjectSetExt</i></p>	<p><b>PageName</b> Textkonstante oder -variable, die den Namen der Seite angibt, auf der die Objekte gelöscht werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>ObjectName</b> Textkonstante oder -variable, die die Namen der Objekte angibt, gelöscht werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>Flags</b> Muss 0 sein. Ist für spätere Zwecke reserviert.</p>

<b>DeleteObjectsInRect</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion löscht alle Objekte, die innerhalb eines angegebenen Rechtecks der aktuellen Seite des Formulars liegen. Die Funktion kann während des Ausfüllens aufgerufen werden.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkung</b> In Verbindung mit dem Befehl <i>AppendObjectSet</i> mittels dessen Objekte dynamisch in ein Formular eingefügt werden können, kann ein dynamisches Formular angelegt werden, dessen Design und Objektstruktur von bestimmten Situationen abhängig gemacht wird.</p> <p><b>Verwandte Funktionen</b> <i>DeleteObjectsByName</i> <i>AppendObjectSetExt</i></p>	<p><b>Left</b> Ganzzahlige Konstante oder Variable, die den linken Rand des Rechtecks in Mikrometern angibt, innerhalb dessen alle Objekte gelöscht werden sollen.</p> <p><b>Top</b> Ganzzahlige Konstante oder Variable, die den oberen Rand des Rechtecks in Mikrometern angibt, innerhalb dessen alle Objekte gelöscht werden sollen.</p> <p><b>Right</b> Ganzzahlige Konstante oder Variable, die den rechten Rand des Rechtecks in Mikrometern angibt, innerhalb dessen alle Objekte gelöscht werden sollen.</p> <p><b>Bottom</b> Ganzzahlige Konstante oder Variable, die den unteren Rand des Rechtecks in Mikrometern angibt, innerhalb dessen alle Objekte gelöscht werden sollen.</p> <p><b>Flags</b> Muss 0 sein. Ist für spätere Zwecke reserviert.</p>



DllCall	
<p><b>Funktionsbeschreibung</b> Diese Funktion ruft ein Unterprogramm in einer beliebigen externen dynamischen Bibliothek (DLL) auf.</p> <p><b>Funktionswert</b> -1 Es trat ein Fehler auf n Wert, den die externe Funktion liefert</p> <p><b>Bemerkung</b> Das Unterprogramm muss in der DLL wie folgt deklariert sein:  long <i>Name</i> (char <i>String</i>*, char *<i>Ergebnis</i>, int <i>Size</i>)  Auf dem Parameter <i>String</i> wird der Funktionsparameter <i>String</i> durchgereicht. Auf dem Parameter <i>Ergebnis</i> kann die Funktion bis zu einer Länge <i>Size</i> ein Ergebnis zurückliefern, welches die Makrofunktion auf dem Parameter <i>Ergebnis</i> zurückliefert.</p> <p><b>Verwandte Funktionen</b> <i>DllLoadAllObjects</i> <i>DllLoadObject</i></p>	<p><b>Dll</b> Textkonstante oder -variable, die den Namen der DLL angibt, in der das Unterprogramm aufgerufen werden soll.</p> <p><b>Function</b> Textkonstante oder -variable, die den Namen des Unterprogramms angibt, das in der DLL aufgerufen werden soll.</p> <p><b>String</b> Textkonstante oder -variable, die den Parameter <i>String</i> angibt, der an das externe Unterprogramm als erster Parameter übergeben wird.</p> <p><b>Result</b> Textvariable, keine Konstante, auf der das Ergebnis, das das externe Unterprogramm auf dem zweiten Parameter übergibt, zurückgeliefert wird.</p>

<b>DllLoadAllObjekts</b>	
<p><b>Funktionsbeschreibung</b>                      Diese Funktion ruft für jedes Eingabeobjekt, welches sich in dem aktuellen Formular befindet, ein Unterprogramm in einer dynamischen Bibliothek (DLL) auf, welches neue Inhalte für die Eingabeobjekte bestimmt.</p> <p><b>Funktionswert</b>                      -1 Es trat ein Fehler auf                      0 Alle Aufrufe wurden durchgeführt</p> <p><b>Bemerkung</b>                      Die Funktion kann dazu verwendet werden, alle Eingabeobjekte eines Formulars durch eine externe Prozedur zu besetzen.                       Das Unterprogramm muss in der DLL wie folgt deklariert sein:                       long <i>Name</i> (char *Name, char *Ergebnis, int Size)                       Auf dem Parameter <i>Name</i> wird der Name des Eingabeobjekts an das Unterprogramm durchgereicht. Auf dem Parameter <i>Ergebnis</i> liefert das Unterprogramm den neuen Inhalt des betreffenden Eingabeobjekts zurück. Es werden nicht mehr als <i>Size</i> Zeichen übertragen.</p> <p><b>Verwandte Funktionen</b>                      DllLoadObject'                      DllSaveAllObjects</p>	<p><b>Dll</b>                      Textkonstante oder -variable, die den Namen der DLL angibt, in der das Unterprogramm aufgerufen werden soll.</p> <p><b>Function</b>                      Textkonstante oder -variable, die den Namen des Unterprogramms angibt, das in der DLL aufgerufen werden soll.</p>

<b>DllLoadObject</b>	
<p><b>Funktionsbeschreibung</b>                      Diese Funktion ruft für ein namentlich angegebenes Eingabeobjekt, das sich in dem aktuellen Formular befindet, ein Unterprogramm in einer dynamischen Bibliothek (DLL) auf, welches den neuen Inhalt für das Eingabeobjekt bestimmt.</p> <p><b>Funktionswert</b>                      -1 Es trat ein Fehler auf                      0 Der Aufruf wurde korrekt durchgeführt</p> <p><b>Bemerkung</b>                      Die Funktion kann dazu verwendet werden, ein bestimmtes Eingabeobjekt eines Formulars durch eine externe Prozedur zu besetzen.                       Das Unterprogramm muss in der DLL wie folgt deklariert sein:                       long <i>Name</i> (char *<i>Name</i>, char *<i>Ergebnis</i>, int <i>Size</i>)                       Auf dem Parameter <i>Name</i> wird der Name des Eingabeobjekts an das Unterprogramm durchgereicht. Auf dem Parameter <i>Ergebnis</i> liefert das Unterprogramm den neuen Inhalt des betreffenden Eingabeobjekts zurück. Es werden nicht mehr als <i>Size</i> Zeichen übertragen.</p> <p><b>Verwandte Funktionen</b>  <i>DllLoadAllObjects</i>  <i>DllSaveObject</i></p>	<p><b>Dll</b>                      Textkonstante oder -variable, die den Namen der DLL angibt, in der das Unterprogramm aufgerufen werden soll.</p> <p><b>Function</b>                      Textkonstante oder -variable, die den Namen des Unterprogramms angibt, das in der DLL aufgerufen werden soll.</p> <p><b>ObjectName</b>                      Textkonstante oder -variable, die den Namen des Eingabeobjekts angibt, für das die externe Routine aufgerufen werden soll.</p>

<b>DllSaveAllObjekts</b>	
<p><b>Funktionsbeschreibung</b>                      Diese Funktion ruft für jedes Eingabeobjekt, welches sich in dem aktuellen Formular befindet, ein Unterprogramm in einer dynamischen Bibliothek (DLL) auf, an welches Name und augenblicklicher Inhalt des Eingabeobjektes übergeben werden.</p> <p><b>Funktionswert</b>                      -1 Es trat ein Fehler auf                      0 Alle Aufrufe wurden durchgeführt</p> <p><b>Bemerkung</b>                      Die Funktion kann dazu verwendet werden, alle Eingabeobjekte eines Formulars durch eine externe Prozedur weiterzuverarbeiten (z.B. diese extern zu speichern).</p> <p>Das Unterprogramm muss in der DLL wie folgt deklariert sein:</p> <p style="padding-left: 20px;">long <i>Name</i> (char *<i>Name</i>, char *<i>Inhalt</i>)</p> <p>Auf dem Parameter <i>Name</i> wird der Name des Eingabeobjekts an das Unterprogramm durchgereicht. Auf dem Parameter <i>Inhalt</i> wird der augenblickliche Inhalt des Feldes <i>Name</i> übergeben.</p> <p><b>Verwandte Funktionen</b>                      DllSaveObject'                      DllLoadAllObjects</p>	<p><b>Dll</b>                      Textkonstante oder -variable, die den Namen der DLL angibt, in der das Unterprogramm aufgerufen werden soll.</p> <p><b>Function</b>                      Textkonstante oder -variable, die den Namen des Unterprogramms angibt, das in der DLL aufgerufen werden soll.</p>

<b>DllSaveObject</b>	
<p><b>Funktionsbeschreibung</b>                      Diese Funktion ruft für ein namentlich angegebenes Eingabeobjekt, das sich in dem aktuellen Formular befindet, ein Unterprogramm in einer dynamischen Bibliothek (DLL) auf und übergibt diesem Unterprogramm seinen aktuellen Inhalt.</p> <p><b>Funktionswert</b>                      -1 Es trat ein Fehler auf                      0 Der Aufruf wurde korrekt durchgeführt</p> <p><b>Bemerkung</b>                      Die Funktion kann dazu verwendet werden, ein bestimmtes Eingabeobjekt eines Formulars durch eine externe Prozedur weiterzuverarbeiten.                       Das Unterprogramm muss in der DLL wie folgt deklariert sein:                       long <i>Name</i> (char *<i>Name</i>, char *<i>Inhalt</i>)                       Auf dem Parameter <i>Name</i> wird der Name des Eingabeobjekts an das Unterprogramm durchgereicht. Auf dem Parameter <i>Inhalt</i> wird der augenblickliche Inhalt übergeben.</p> <p><b>Verwandte Funktionen</b>  <i>DllLoadObject</i>  <i>DllSaveAllObjects</i></p>	<p><b>Dll</b>                      Textkonstante oder -variable, die den Namen der DLL angibt, in der das Unterprogramm aufgerufen werden soll.</p> <p><b>Function</b>                      Textkonstante oder -variable, die den Namen des Unterprogramms angibt, das in der DLL aufgerufen werden soll.</p> <p><b>ObjectName</b>                      Textkonstante oder -variable, die den Namen des Eingabeobjekts angibt, für das die externe Routine aufgerufen werden soll.</p>

<b>EditFonts</b>	
<p><b>Funktionsbeschreibung</b> Dieser Befehl eröffnet einen Dialog, der das Ändern der Schriftart und -größe für Schriften in Objekten ermöglicht.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Es wird die Schrift aller Objekte geändert, die zu dem angegebenen Objektnamen passen und die auf einer Seite mit dem angegebenen Seitennamen liegen.</p> <p><b>Verwandte Funktionen</b> <i>SetFontHeight</i> <i>SetFontName</i></p>	<p><b>PageName</b> Textkonstante oder -variable, die den Namen der Seiten angibt, auf denen Schriften verändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B). Ein leerer Text bezeichnet die aktuelle Seite.</p> <p><b>ObjectName</b> Textkonstante oder -variable, die den Namen der Objekte angibt, deren Schrift verändert werden soll. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>Mode</b> Ganzzahlige Konstante oder Variable die angibt, ob alle Objekte oder nur Eingabeobjekte betroffen sein sollen:</p> <p><b>0</b> nur Eingabeobjekte <b>1</b> alle Objekte</p>

<b>FindObject</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion sucht das erste/nächste Objekt auf einer Seite, in einem Seitenbereich oder im gesamten Formular, das bestimmte Namenskonventionen erfüllt.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"> <li><b>0</b> Kein (weiteres) Objekt gefunden</li> <li><b>n</b> Handle des nächsten gefundenen Objekts</li> </ul> <p><b>Bemerkungen</b> Diese Funktion liefert den Handle eines gefundenen Objekts zurück. Dieser Handle kann in verschiedenen Funktionen verwendet werden, um dieses Objekt zu manipulieren oder um Informationen über das Objekt auszulesen.</p> <p><b>Verwandte Funktionen</b> <i>FindAccessObject</i></p>	<p><b>PageName</b> Textkonstante oder -variable, die den Namen der Seiten angibt, auf denen Objekte gesucht werden sollen.</p> <p>Der Name darf auch eine sogenannte <i>Wildcard</i> (Anhang B) sein.</p> <p>Wenn die Seitennummer bekannt ist, kann auch die Nummer der Seite mit einem vorangestellten "#" (also z.B. "#3" für die dritte Seite) verwendet werden.</p> <p>Ein leerer Text bezeichnet die aktuelle Seite.</p> <p><b>ObjectName</b> Textkonstante oder -variable, die den Namen der Objekte angibt, die gesucht werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>Reference</b> Ganzzahlige Konstante oder Variable, die das Objekt bezeichnet, ab dem gesucht werden soll:</p> <p><b>0:</b> Suche beginnt mit dem ersten Objekt der ersten Seite (Erstaufruf)</p> <p><b>Handle</b> Suche beginnt mit dem Objekt welches in der Reihenfolge der Objekte nach dem Objekt mit dem angegebenen <i>Handle</i> folgt (Folgeaufruf)</p>

<b>FindAccessObject</b>	
<p><b>Funktionsbeschreibung</b>                      Diese Funktion sucht das erste/nächste <i>Datenzugriffsobject</i> auf einer Seite, in einem Seitenbereich oder im gesamten Formular, das bestimmte Namenskonventionen erfüllt.</p> <p><b>Funktionswert</b>                      0 Kein (weiteres) Objekt gefunden                      n Handle des nächsten gefundenen Objekts</p> <p><b>Bemerkungen</b>                      Diese Funktion liefert den Handle des gefundenen <i>Datenzugriffsobjekts</i> zurück. Dieser Handle kann insbesondere verwendet werden, um Datenzugriffsfunktionen aufzurufen.</p> <p><b>Verwandte Funktionen</b>  <i>FindObject</i>  <i>AccObjDeleteRecord</i>  <i>AccObjInsertRecord</i>                      und alle Funktionen, die mit <i>AccObj...</i> beginnen.</p>	<p><b>ObjectName</b>                      Textkonstante oder -variable, die den Namen der Objekte angibt, die gesucht werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p>



<b>GetActiveState</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt, ob ein Objekt den Status <i>aktiv</i> hat oder nicht. Momentan ist dieser Status nur für Objekte vom Typ <i>Button</i> implementiert und repräsentiert den Zustand <i>gedrückt</i>.</p> <p><b>Funktionswert</b> <b>1</b> Objekt ist aktiv <b>0</b> Objekt ist nicht aktiv <b>-1</b> Objekt nicht gefunden</p> <p><b>Bemerkung</b> Es wird der Status des ersten Objekts mit dem angegebenen Namen auf der aktuellen Seite ermittelt.</p> <p>Wenn auf der aktuellen Seite ein solches Objekt nicht gefunden wird, wird die Suche auf das gesamte Formular ausgedehnt.</p> <p><b>Verwandte Funktionen</b> <i>SetActiveState</i></p>	<p><b>ObjectName</b> Textvariable oder -konstante, die den Namen des Objekts angibt, dessen Status ermittelt werden soll.</p>

<b>GetAssociatedComboColor</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt die Farbe, die zu einem Eintrag eines Auswahlfeldes hinterlegt ist.</p> <p><b>Funktionswert</b>  <b>n</b> Nummer des Eintrags in der Box (ab 0)  <b>-1</b> Eintrag in der Box nicht gefunden</p> <p><b>Verwandte Funktionen</b> <i>GetAssociatedComboValue</i></p>	<p><b>Handle</b> Ganzzahlige Konstante oder Variable, die den Handle des Combo-Objekts angibt, das angesprochen werden soll.</p> <p><b>Text</b> Textkonstante oder -variable, die den Anzeigetext angibt, dessen zugehöriger Wert ermittelt werden soll.</p> <p><b>Red</b> Ganzzahlige Variable, keine Konstante, auf der der Wert für den Farbwert <i>rot</i> zurückgeliefert wird.</p> <p><b>Green</b> Ganzzahlige Variable, keine Konstante, auf der der Wert für den Farbwert <i>grün</i> zurückgeliefert wird.</p> <p><b>Blue</b> Ganzzahlige Variable, keine Konstante, auf der der Wert für den Farbwert <i>blau</i> zurückgeliefert wird.</p>

<b>GetAssociatedComboValue</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt den Wert, der zu einem Eintrag eines Auswahlfeldes in der Werteliste hinterlegt ist.</p> <p><b>Funktionswert</b>  <b>n</b> Nummer des Eintrags in der Box (ab 0)  <b>-1</b> Eintrag in der Box nicht gefunden</p> <p><b>Verwandte Funktionen</b> <i>GetAssociatedComboColor</i></p>	<p><b>Handle</b> Ganzzahlige Konstante oder Variable, die den Handle des Combo-Objekts angibt, das angesprochen werden soll.</p> <p><b>Text</b> Textkonstante oder -variable, die den Anzeigetext angibt, dessen zugehöriger Wert ermittelt werden soll.</p> <p><b>Result</b> Textvariable, keine Konstante, auf der der Wert zurückgeliefert wird.</p>

<b>GetCapsState</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt, ob für ein Objekt das Attribut <i>Großschreibung</i> gesetzt ist oder nicht.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"> <li><b>1</b> Großschreibung ist aktiv</li> <li><b>0</b> Großschreibung ist nicht aktiv</li> <li><b>-1</b> Objekt nicht gefunden</li> </ul> <p><b>Bemerkung</b> Es wird der Status des ersten Objekts mit dem angegebenen Namen auf der aktuellen Seite ermittelt.</p> <p>Wenn auf der aktuellen Seite ein solches Objekt nicht gefunden wird, wird die Suche auf das gesamte Formular ausgedehnt.</p> <p><b>Verwandte Funktionen</b> <i>SetCapsState</i></p>	<p><b>ObjectName</b> Textvariable oder -konstante, die den Namen des Objekts angibt, dessen Status ermittelt werden soll.</p> <p>Wenn ein leerer Text angegeben wird, wird der Status des augenblicklich aktiven Objekts ermittelt.</p>

<b>GetChangeFlag</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion liefert die interne Merkvariable, die angibt, ob ein Formular geändert wurde, zurück.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"> <li><b>1</b> Formular wurde geändert</li> <li><b>0</b> Formular wurde nicht geändert</li> </ul> <p><b>Verwandte Funktionen</b> <i>GetChangeFlag</i></p>	<p>Die Funktion hat keine Parameter</p>

<b>GetCheckbox</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt den Zustand eines <i>Ankreuzfeldes</i>.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"><li><b>1</b> Das Ankreuzfeld ist angekreuzt.</li><li><b>0</b> Das Ankreuzfeld ist nicht angekreuzt.</li><li><b>-1</b> Das Objekt ist kein Ankreuzfeld.</li></ul>	<p><b>ObjectName</b> Textvariable oder -konstante, die den Namen des Ankreuzfeldes angibt, dessen Zustand ermittelt werden soll.</p> <p>Wenn hier ein leerer Text angegeben wird, wird das augenblicklich aktive Objekt angesprochen.</p>

<b>GetColorProperty</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion liest eine von verschiedenen möglichen <i>Farb-Eigenschaften</i> eines Formularobjekts aus.</p> <p><b>Funktionswert</b>  <b>0</b> Die Farbe wurde korrekt ermittelt  <b>-1</b> Fehler oder Objekt unbekannt</p> <p><b>Bemerkungen</b> Nicht alle Eigenschaften haben für jeden Objekttyp eine Bedeutung. Manche Eigenschaften haben nur für bestimmte Objekttypen einen Sinn.  Wenn Eigenschaften benannt werden, die es zwar prinzipiell gibt, aber die für den angegebenen Objekttyp keine Bedeutung haben, erfolgt keine Fehlermeldung, aber der Aufruf der Funktion ist ohne Wirkung.  Wenn Eigenschaften benannt werden, die nicht definiert sind, erfolgt eine Fehlermeldung.</p> <p><b>Verwandte Funktionen</b>  <i>SetColorProperty</i>  <i>GetLongProperty</i>  <i>GetStringProperty</i>  <i>GetFontProperty</i></p>	<p><b>Handle</b> Ganzzahlige Konstante oder Variable, die den Handle des Objekts angibt, das angesprochen werden soll.</p> <p><b>Property</b> Textkonstante oder -variable, die den Namen der Eigenschaft angibt, die gelesen werden soll.</p> <p>Folgende Eigenschaften sind definiert:</p> <p><b>ColorBgr</b>  <b>ColorFont</b>  <b>ColorFrame</b>  <b>ColorTempDisp</b></p> <p><b>Red</b> Ganzzahlige Variable, keine Konstante, auf der Rotanteil der Farbe als Wert zwischen 0 und 255 abgeliefert wird.</p> <p><b>Green</b> Ganzzahlige Variable, keine Konstante, auf der Grünanteil der Farbe als Wert zwischen 0 und 255 abgeliefert wird.</p> <p><b>Blue</b> Ganzzahlige Variable, keine Konstante, auf der Blauanteil der Farbe als Wert zwischen 0 und 255 abgeliefert wird.</p>

**Beispiel**

Nach Ausführung der Befehle:

```
int G;  
int G;  
int B;  
int Handle;  
Handle = FindObject ("", "Eingabe", 0);  
GetColorProperty (Handle, "ColorBgr", R,G,B);
```

enthalten die Variablen *R*, *G* und *B* die Hintergrundfarbe des ersten gefundenen Eingabeobjekts mit dem Namen *Eingabe*.

**GetComboCount**

**Funktionsbeschreibung**

Diese Funktion liefert die Anzahl der Listeneinträge eines Auswahlobjekts.

**Funktionswert**

- n** Anzahl der Einträge
- 1** Fehler oder Objekt unbekannt

**Verwandte Funktionen**

- GetComboIndex*
- GetComboListIndex*
- GetComboListItem*

**Handle**

Ganzzahlige Konstante oder Variable, die den Handle des Auswahlobjekts angibt.

<b>GetComboIndex</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion liefert den Index des augenblicklich selektierten Eintrags eines Auswahlobjekts.</p> <p><b>Funktionswert</b>  <b>n</b> Index des selektierten Eintrags (ab 0)  <b>-1</b> Fehler oder Objekt unbekannt</p> <p><b>Bemerkungen</b> Die Abfrage des selektierten Eintrags macht nur Sinn, wenn das Auswahlobjekt das zur Zeit aktive Eingabeobjekt ist, da nur dann das entsprechende Windows-Control existiert.</p> <p><b>Verwandte Funktionen</b>  <i>GetComboCount</i>  <i>GetComboListIndex</i>  <i>GetComboListItem</i>  <i>SetComboList</i></p>	<p><b>Handle</b> Ganzzahlige Konstante oder Variable, die den Handle des Auswahlobjekts angibt.</p>

<b>GetComboListIndex</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion liefert die Nummer (den Index) eines Eintrags in der Auswahlliste eines Auswahlobjekts zurück.</p> <p><b>Funktionswert</b>  <b>n</b> Index des Eintrags (ab 0)  <b>-1</b> Fehler oder Objekt unbekannt</p> <p><b>Verwandte Funktionen</b>  <i>GetComboCount</i>  <i>GetComboIndex</i>  <i>GetComboListItem</i>  <i>SetComboList</i></p>	<p><b>Handle</b> Ganzzahlige Konstante oder Variable, die den Handle des Auswahlobjekts angibt.</p> <p><b>Displayname</b> Textkonstante oder -variable, die den Anzeigenamen des Listeneintrags angibt, dessen Index gefunden werden soll.</p>

<b>GetComboListItem</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion liefert den Anzeigenamen eines Eintrags in der Auswahlliste eines Auswahlobjekts auf Basis seiner Position in der Liste zurück.</p> <p><b>Funktionswert</b> n Position des Eintrags (ab 0) -1 Fehler oder Objekt unbekannt</p> <p><b>Verwandte Funktionen</b> <i>GetComboCount</i> <i>GetComboIndex</i> <i>GetComboListIndex</i> <i>SetComboList</i></p>	<p><b>Handle</b> Ganzzahlige Konstante oder Variable, die den Handle des Auswahlobjekts angibt.</p> <p><b>Position</b> Ganzzahlige Konstante oder -variable, die die Position des auszulesenden Anzeigenamens in der Auswahlliste bezeichnet (ab 0).</p>

<b>GetCopyName</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt den Namen der Ausfertigung, die augenblicklich gedruckt wird.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Wenn augenblicklich nicht gedruckt wird, ist das Ergebnis undefiniert.</p>	<p><b>Result</b> Textvariable, keine Konstante, auf der der Name der Ausfertigung zurückgeliefert wird.</p>



<b>GetCurrentObject</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion liefert den Namen des Eingabeobjektes, in dem sich der Cursor momentan befindet.</p> <p><b>Funktionswert</b> 0 Name des Eingabeobjektes wurde ermittelt -1 Der Cursor steht in keinem Eingabeobjekt</p> <p><b>Verwandte Funktionen</b> <i>GetPointedObject</i></p>	<p><b>Result</b> Textvariable, keine Konstante, auf der der Name des Objekts zurückgeliefert wird.</p>

<b>GetCurrentPage</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt den Namen der augenblicklich aktiven Seite eines Formulars. Dies ist die Seite, auf der sich das Eingabeobjekt befindet, in dem sich der Cursor momentan befindet.</p> <p><b>Funktionswert</b> 0 Der Name der Seite wurde ermittelt -1 Es wurde keine aktive Seite gefunden</p>	<p><b>Result</b> Textvariable, keine Konstante, auf der der Name der Seite zurückgeliefert wird.</p>

<b>GetDataField</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion liest den Inhalt eines Datenfeldes der augenblicklich im Zugriff befindlichen Datenbank.</p> <p>Es wird auf das Feld desjenigen Satzes zugegriffen, der sich momentan im Zugriff befindet.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"> <li><b>0</b> Das Datenfeld wurde korrekt gelesen</li> <li><b>-1</b> Feld nicht gefunden oder kein Zugriff offen</li> </ul> <p><b>Bemerkung</b> Der Zugriff auf das Datenfeld ist unabhängig davon möglich, ob das gleichnamige Feld im Formular überhaupt verwendet wird.</p>	<p><b>DataBaseField</b> Textkonstante oder -variable, die den Namen des Datenfeldes angibt.</p> <p><b>Result</b> Textvariable, keine Konstante, auf der der Inhalt des Datenfeldes zurückgeliefert wird.</p>

<b>GetDisplayState</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt, ob ein Objekt augenblicklich sichtbar ist oder nicht.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"> <li><b>1</b> Objekt ist sichtbar</li> <li><b>0</b> Objekt ist unsichtbar</li> <li><b>-1</b> Objekt nicht gefunden</li> </ul> <p><b>Bemerkung</b> Es wird der Status des ersten Objekts mit dem angegebenen Namen auf der aktuellen Seite ermittelt.</p> <p>Wenn auf der aktuellen Seite ein solches Objekt nicht gefunden wird, wird die Suche auf das gesamte Formular ausgedehnt.</p> <p><b>Verwandte Funktionen</b> <i>SetDisplayState</i></p>	<p><b>ObjectName</b> Textvariable oder -konstante, die den Namen des Objekts angibt, dessen Status ermittelt werden soll.</p> <p>Wenn ein leerer Text angegeben wird, wird der Status des augenblicklich aktiven Objekts ermittelt.</p>

<b>GetFloatFormatted</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt den Wert eines beliebigen Objekts als Gleitkommazahl.</p> <p><b>Funktionswert</b>  <b>0</b> Der Wert des Objekts wurde ermittelt  <b>-1</b> Kein Objekt dieses Namens gefunden</p> <p>Wenn mehr als ein Objekt mit dem angegebenen Namen existiert, wird immer der Wert des <i>ersten gefundenen</i> Objekts gelesen. Dies ist ein wesentlicher Unterschied zur Schwesterfunktion <i>SetFloatFormatted</i>, welche immer <i>alle</i> betroffenen Objekte verändert.</p> <p><b>Verwandte Funktionen</b>  <i>SetFloatFormatted</i>  <i>GetText</i>  <i>GetLongFormatted</i></p>	<p><b>ObjectName</b> Textkonstante oder -variable, die den Namen des Objekts angibt, dessen Text ermittelt werden soll.</p> <p><b>Result</b> Gleitkommavariablen, keine Konstante, auf der der ermittelte Wert zurückgeliefert wird.</p> <p><b>Format</b> Textkonstante oder -variable, die den <i>Formatdeskriptor</i> definiert, nach dem die Zahl interpretiert werden soll.</p> <p>Ein leerer Text bedeutet, dass die Zahl im Standardformat formatiert wird (Normalfall).</p> <p>Formatdeskriptoren sind in Anhang G beschrieben.</p>
<p><b>Beispiel</b></p> <pre>float result; GetFloatFormatted ("Preis", result, "");</pre> <p>Nach diesem Befehl befindet sich der Wert des Eingabeobjekts <i>Preis</i> in der Variablen <i>result</i>.</p>	

<b>GetFontHeight</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt die Schrifthöhe eines Objekts.</p> <p><b>Funktionswert</b>  <b>n</b> Schriftart in Zehntelpunkten  <b>-1</b> Objekt nicht gefunden</p> <p><b>Bemerkung</b> Es wird die Schrifthöhe des ersten Objekts mit dem angegebenen Namen auf der aktuellen Seite ermittelt.</p> <p>Wenn auf der aktuellen Seite ein solches Objekt nicht gefunden wird, wird die Suche auf das gesamte Formular ausgedehnt.</p> <p><b>Verwandte Funktionen</b> <i>GetFontName</i></p>	<p><b>ObjectName</b> Textvariable oder -konstante, die den Namen des Objekts angibt, dessen Schrifthöhe ermittelt werden soll.</p> <p>Wenn ein leerer Text angegeben wird, wird die Schrifthöhe des augenblicklich aktiven Objekts ermittelt.</p>

<b>GetFontName</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt die Schriftart eines Objekts.</p> <p><b>Funktionswert</b>  <b>0</b> Die Schriftart des Objekts wurde ermittelt  <b>-1</b> Objekt nicht gefunden</p> <p><b>Bemerkung</b> Es wird die Schriftart des ersten Objekts mit dem angegebenen Namen auf der aktuellen Seite ermittelt.</p> <p>Wenn auf der aktuellen Seite ein solches Objekt nicht gefunden wird, wird die Suche auf das gesamte Formular ausgedehnt.</p> <p><b>Verwandte Funktionen</b> <i>GetFontHeight</i></p>	<p><b>ObjectName</b> Textvariable oder -konstante, die den Namen des Objekts angibt, dessen Schriftart ermittelt werden soll.</p> <p>Wenn ein leerer Text angegeben wird, wird die Schriftart des augenblicklich aktiven Objekts ermittelt.</p>

<b>GetFontProperty</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion liest eine von mehreren Font-Eigenschaften aus und liefert diese als String zurück.</p> <p><b>Funktionswert</b> Kein Funktionswert</p> <p><b>Bemerkungen</b> Wenn versucht, wird, eine nicht definierte Eigenschaft zu lesen, wird ein leerer Text zurückgeliefert.</p> <p><b>Verwandte Funktionen</b> <i>SetFontProperty</i> <i>GetColorProperty</i> <i>GetStringProperty</i> <i>GetLongProperty</i></p>	<p><b>Handle</b> Ganzzahlige Konstante oder Variable, die den Handle des Objekts angibt, das angesprochen werden soll.</p> <p><b>Property</b> Textkonstante oder -variable, die den Namen der Eigenschaft angibt, die gelesen werden soll.</p> <p>Folgende Eigenschaften sind definiert:</p> <ul style="list-style-type: none"> <li><b>Color</b></li> <li><b>Escapement</b></li> <li><b>Face</b></li> <li><b>Height</b></li> <li><b>Italic</b></li> <li><b>Strikeout</b></li> <li><b>Underline</b></li> <li><b>Weight</b></li> </ul> <p><b>Data</b> Textvariable, auf der das Ergebnis zurückgeliefert wird.</p>
<p><b>Beispiel</b> Nach Ausführung der Befehle:</p> <pre style="color: green;">char weight [100]; int Handle; Handle = FindObject ("", "Eingabe", 0); GetFontProperty (Handle, "Weight", weight);</pre> <p>enthält der Puffer <i>weight</i> den Wert "400" wenn das Objekt <i>Eingabe</i> in Normalschrift dargestellt wird, bzw. "700" wenn das Objekt in Fettschrift dargestellt wird.</p>	

<b>GetFormWindow</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt den Handle des augenblicklich aktiven Formulars.</p> <p><b>Funktionswert</b>  <b>0</b> Fehler  <b>Handle</b> Handle des Formularfensters</p> <p><b>Bemerkung</b> Der Handle kann beispielsweise verwendet werden, um Informationen über das Fenster zu erfahren – z.B. dessen Größe.</p>	<p>Die Funktion hat keine Parameter.</p>
<p><b>Beispiel</b> Die folgende Sequenz ermittelt die Größe des Formularfensters:</p> <pre style="color: green;">int Height; int Width; GetWindowClientSize (GetFormWindow (), Height, Width);</pre> <p>Nach dieser Sequenz befindet sich die Größe des Formularfensters auf den Variablen <i>Width</i> und <i>Height</i>.</p>	

<b>GetHoriAlign</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt die horizontale Ausrichtung eines Objekts.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"><li>1 Objekt ist linksbündig</li><li>2 Objekt ist zentriert</li><li>3 Objekt ist rechtsbündig</li><li>-1 Objekt nicht gefunden</li></ul> <p><b>Bemerkung</b> Es wird die Ausrichtung des ersten Objekts mit dem angegebenen Namen auf der aktuellen Seite ermittelt.</p> <p>Wenn auf der aktuellen Seite ein solches Objekt nicht gefunden wird, wird die Suche auf das gesamte Formular ausgedehnt.</p> <p><b>Verwandte Funktionen</b> GetVertAlign SetHoriAlign</p>	<p><b>ObjectName</b> Textvariable oder -konstante, die den Namen des Objekts angibt, dessen Ausrichtung ermittelt werden soll.</p> <p>Wenn ein leerer Text angegeben wird, wird die Ausrichtung des augenblicklich aktiven Objekts ermittelt.</p>

<b>GetLongFormatted</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt den Wert eines beliebigen Objekts als Ganzzahl.</p> <p><b>Funktionswert</b></p> <p><b>x</b> Der Wert des Objekts als ganze Zahl interpretiert</p> <p><b>0</b> Es wurde kein Objekt dieses Namens gefunden oder das Objekt ist leer oder hat den Wert Null.</p> <p>Wenn mehr als ein Objekt mit dem angegebenen Namen existiert, wird immer der Wert des <b>ersten gefundenen</b> Objekts gelesen. Dies ist ein wesentlicher Unterschied zur Schwesterfunktion <i>SetLongFormatted</i>, welche immer <b>alle</b> betroffenen Objekte verändert.</p> <p><b>Verwandte Funktionen</b> <i>SetLongFormatted</i> <i>GetText</i> <i>GetFloatFormatted</i></p>	<p><b>ObjectName</b> Textkonstante oder -variable, die den Namen des Objekts angibt, dessen Text ermittelt werden soll.</p> <p><b>Format</b> Textkonstante oder -variable, die den Formatdeskriptor definiert, nach dem die Zahl interpretiert werden soll.</p> <p>Ein leerer Text bedeutet, dass die Zahl im Standardformat formatiert wird (Normalfall).</p> <p>Formatdeskriptoren sind in Anhang G beschrieben.</p>
<p><b>Beispiel</b></p> <pre>int wert; wert = GetLongFormatted ("Feld", "");</pre> <p>Nach dieser Sequenz befindet sich der Inhalt des Eingabeobjekts <i>Feld</i> auf der Variablen <i>wert</i>.</p>	



<b>GetLongProperty</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion liest eine von verschiedenen möglichen <i>ganzzahligen Eigenschaften</i> eines Formularobjekts aus.</p> <p><b>Funktionswert</b>  <b>n</b> Ergebnis  <b>-1</b> Fehler oder Objekt unbekannt</p> <p><b>Bemerkungen</b> Nicht alle Eigenschaften haben für jeden Objekttyp eine Bedeutung. Manche Eigenschaften haben nur für bestimmte Objekttypen einen Sinn.  Wenn Eigenschaften benannt werden, die es zwar prinzipiell gibt, aber die für den angegebenen Objekttyp keine Bedeutung haben, erfolgt keine Fehlermeldung, aber der Aufruf der Funktion ist ohne Wirkung.  Wenn Eigenschaften benannt werden, die nicht definiert sind, erfolgt eine Fehlermeldung.</p> <p><b>Verwandte Funktionen</b>  <i>SetLongProperty</i>  <i>GetStringProperty</i>  <i>GetColorProperty</i></p>	<p><b>Handle</b> Ganzzahlige Konstante oder Variable, die den Handle des Objekts angibt, das angesprochen werden soll.</p> <p><b>Property</b> Textkonstante oder -variable, die den Namen der Eigenschaft angibt, die gelesen werden soll.</p> <p>Folgende Eigenschaften sind definiert:</p> <p><b>TempDispTime</b>  <b>DataType</b>  <b>SelectLimit</b>  <b>HoriAlign</b>  <b>VertAlign</b></p> <p>Nicht alle Eigenschaften sind für jedes Objekte sinnvoll.</p>
<p><b>Beispiel</b> Nach den Befehlen:</p> <pre style="color: green;">int time; int handle; handle = FindObject ("", "Zugriff", 0); time = GetLongProperty (handle, "TempDispTime");</pre> <p>enthält die Variable <i>Time</i> die Zeit in Zehntelsekunden, in der bei Operationen des Datenzugriffsobjekts <i>Zugriff</i> die Bestätigung eingeblendet wird, dass die Funktion erfolgreich durchgeführt wurde.</p>	

<b>GetMacroCode</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt den Makrocode eines Objekts oder den Makrocodes eines global Formularmakros.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkung</b> Das Auslesen des Makrocodes von Makroereignissen, die nicht definiert sind, liefert immer einen leeren Code zurück.</p> <p>Den Handle eines Objekts erhält man durch verschiedene andere Makrofunktionen, z.B. durch <i>FindObject</i></p> <p><b>Verwandte Funktionen</b> <i>SetMacroCode</i> <i>FindObject</i></p>	<p><b>ObjectName</b> Ganzzahlige Konstante oder Variable, den Handle des Eingabeobjektes angibt, dessen Code ausgelesen werden soll.</p> <p>Wenn der Wert 0 angegeben wird, wird ein globales Formularmakro ausgelesen.</p> <p><b>MacroName</b> Textkonstante oder -variable, die den Namen des Makroereignisses angibt, dessen Code gelesen werden soll.</p> <p>Wenn hier ein leerer Text angegeben wird, liefert die Funktion anstelle des Makrocodes eine durch Tabulatoren getrennte Liste aller Makroereignisse des Objekts bzw. des Formulars.</p> <p><b>Result</b> Textvariable, keine Konstante, auf der der Code zurückgeliefert wird.</p>

<b>GetObjectAsFloat</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt den augenblicklichen Inhalt eines Objekts als Gleitkommazahl.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"><li><b>0</b> Der Inhalt des Objekts wurde als Gleitkommazahl ermittelt.</li><li><b>-1</b> Das Objekt wurde nicht gefunden.</li></ul> <p><b>Bemerkungen</b> Wenn der Text in dem Objekt keine gültige Gleitkommazahl ist, wird der Wert Null zurückgeliefert.</p> <p><b>Verwandte Funktionen</b> <i>GetText</i> <i>SetFloat</i></p>	<p><b>ObjectName</b> Textkonstante oder -variable, die den Namen des Objekts angibt, dessen Inhalt als Gleitkommazahl ermittelt werden soll.</p> <p>Ein leerer Text bedeutet, dass der Inhalt des augenblicklich aktiven Eingabeobjektes ermittelt werden soll.</p> <p><b>Result</b> Gleitkommavariablen, keine Konstante, auf der das Ergebnis zurückgeliefert wird.</p>

<b>GetObjectDisplayRect</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt das umgebende Rechteck eines Objekts in Screen Pixels.</p> <p><b>Funktionswert</b> 0 Koordinaten wurden ermittelt -1 Das Objekt wurde nicht gefunden.</p> <p><b>Bemerkungen</b> Die Koordinaten sind dokumentenrelativ. Das Dokument ist die Gesamtheit aller Seiten einschließlich der Zwischenräume beim Anzeigen.</p> <p><b>Verwandte Funktionen</b> <i>GetObjectPos</i> <i>GetPageDisplayRect</i></p>	<p><b>Handle</b> Ganzzahlige Konstante oder Variable, die den Handle des Objektes angibt.</p> <p>Der Handle eines Objekts kann mittels verschiedener Funktionen ermittelt werden, z.B. durch die Funktion <i>FindObject</i>.</p> <p><b>Left</b> Ganzzahlige Variable, keine Konstante, auf der linke Rand des Objekts zurückgeliefert wird.</p> <p><b>Top</b> Ganzzahlige Variable, keine Konstante, auf der der obere Rand des Objekts zurückgeliefert wird.</p> <p><b>Right</b> Ganzzahlige Variable, keine Konstante, auf der der rechte Rand Objekts zurückgeliefert wird.</p> <p><b>Bottom</b> Ganzzahlige Variable, keine Konstante, auf der untere Rand des Objekts zurückgeliefert wird.</p>

**Beispiel**

Das folgende Beispiel rollt das Formular in vertikaler Richtung so weit, dass der obere Rand des Objekts *MyObject* auf der Seite *MyPage* am oberen Fensterrand zu stehen kommt.

```
// Variablendeklarationen
```

```
int Handle;
```

```
int left;
```

```
int top;
```

```
int right;
```

```
int bottom;
```

```
// Ermittle Position des Objekts
```

```
Handle = FindObject ("MyPage", "MyObject", 0);
```

```
GetObjectDisplayRect (Handle, left, top, right, bottom);
```

```
// Setze Rollposition
```

```
SetScrollOffset (VERT, top, 0);
```

<b>GetObjectsEnvelope</b>	
<p><b>Funktionsbeschreibung</b>                      Diese Funktion ermittelt das kleinste Rechteck, das alle Objekte eines Namens oder eines Namensschemas umschließt. Die ist das sogenannte umhüllende Rechteck.</p> <p>Damit kann man die am weitesten rechts befindliche Rechte Seite aller Objekte, die am weitesten unten befindliche untere Seite aller Objekte ermitteln, etc..</p> <p><b>Funktionswert</b>                      Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkung</b>                      Mit diesem Befehl kann z.B. die Position von Objekten ermittelt werden, die mittels der Funktion <i>AppendObjectSetExt</i> dynamisch zu einem Formular hinzugefügt wird.</p> <p><b>Verwandte Funktionen</b>  <i>DeleteObjectsInRect</i>  <i>AppendObjectSetExt</i></p>	<p><b>PageName</b>                      Textkonstante oder -variable, die den Namen der Seite angibt, auf die Operation stattfinden soll. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B). In diesem Fall werden die Objekte aller betroffenen Seiten einbezogen.</p> <p><b>ObjectName</b>                      Textkonstante oder -variable, die die Namen der Objekte angibt, deren umhüllendes Rechteck ermittelt werden soll. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>Left</b>                      Ganzzahlige Variable – keine Konstante – auf der die linke Seite des Ergebnisrechtecks in Mikrometern zurückgeliefert wird.</p> <p><b>Top</b>                      Ganzzahlige Variable – keine Konstante – auf der die obere Seite des Ergebnisrechtecks in Mikrometern zurückgeliefert wird.</p> <p><b>Right</b>                      Ganzzahlige Variable – keine Konstante – auf der die rechte Seite des Ergebnisrechtecks in Mikrometern zurückgeliefert wird.</p> <p><b>Bottom</b>                      Ganzzahlige Variable – keine Konstante – auf der die untere Seite des Ergebnisrechtecks in Mikrometern zurückgeliefert wird.</p>

<b>GetObjectPos</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt die Position eines Objekts innerhalb der Seite auf der es sich befindet.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Verwandte Funktionen</b> <i>GetObjectSize</i></p>	<p><b>Handle</b> Ganzzahlige Konstante oder Variable, die den Handle des Eingabeobjektes angibt, dessen Position ermittelt werden soll.</p> <p>Der Handle eines Objekts kann mittels verschiedener Funktionen ermittelt werden, z.B. durch die Funktion <i>FindObject</i>.</p> <p><b>Left</b> Ganzzahlige Variable, keine Konstante, auf die linke Position des Objekts bezogen auf den linken Blattrand zurückgeliefert wird.</p> <p><b>Top</b> Ganzzahlige Variable, keine Konstante, auf die obere Position des Objekts in Bezug auf den oberen Blattrand zurückgeliefert wird.</p>

<b>GetObjectSize</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt die Größe eines Objekts.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Verwandte Funktionen</b> <i>GetObjectPos</i></p>	<p><b>Handle</b> Ganzzahlige Konstante oder Variable, die den Handle des Eingabeobjektes angibt, dessen Größe ermittelt werden soll.</p> <p>Der Handle eines Objekts kann mittels verschiedener Funktionen ermittelt werden, z.B. durch die Funktion <i>FindObject</i>.</p> <p><b>Height</b> Ganzzahlige Konstante oder Variable, auf der die Höhe des Objekts in tausendstel Millimetern zurückgeliefert wird.</p> <p><b>Width</b> Ganzzahlige Konstante oder Variable, auf der die Breite des Objekts in tausendstel Millimetern zurückgeliefert wird.</p>



<b>GetObjectType</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt den Typ eines Objekts.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"><li>1 Eingabefeld</li><li>2 Ankreuzfeld</li><li>3 Auswahlfeld</li><li>4 Rechteck</li><li>5 Bitmap</li><li>6 Linie</li><li>7 Kreis oder Ellipse</li><li>8 Text</li><li>9 aktuelles Datum</li><li>10 aktuelle Uhrzeit</li><li>11 Name der Ausfertigung</li></ul> <p>-1 Das Objekt wurde nicht gefunden</p> <p><b>Bemerkung</b> Es wird der Typ des ersten Objekts mit dem angegebenen Namens auf der aktuellen Seite ermittelt.</p> <p>Wenn auf der aktuellen Seite ein solches Objekt nicht gefunden wird, wird die Suche auf das gesamte Formular ausgedehnt.</p>	<p><b>ObjectName</b> Textvariable oder -konstante, die den Namen des Objekts angibt, dessen Typ ermittelt werden soll.</p> <p>Wenn ein leerer Text angegeben wird, wird der Typ des augenblicklich aktiven Objekts ermittelt.</p>

<b>GetObjectTypeExt</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt den Typ eines Objekts.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"> <li>1 Eingabefeld</li> <li>2 Ankreuzfeld</li> <li>3 Auswahlfeld</li> <li>4 Rechteck</li> <li>5 Bitmap</li> <li>6 Linie</li> <li>7 Kreis oder Ellipse</li> <li>8 Text</li> <li>9 aktuelles Datum</li> <li>10 aktuelle Uhrzeit</li> <li>11 Name der Ausfertigung</li> </ul> <p>-1 Das Objekt wurde nicht gefunden</p>	<p><b>Handle</b></p> <p>Ganzzahlige Variable oder Konstante, die den Handle des Objekts angibt, dessen Typ ermittelt werden soll.</p>

<b>GetPage</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt die Nummer der Seite, auf der sich das Objekt mit dem angegebenen Namen befindet. Wenn das Formular mehrere Objekte dieses Namens enthält, wird die Seite ermittelt, auf der es zuerst auftritt.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"> <li>n Nummer der Seite (Numerierung ab 1).</li> <li>-1 Objekt wurde nicht gefunden.</li> </ul>	<p><b>pageName</b></p> <p>Textkonstante oder -variable, die den Namen der Seite angibt, nach dem gesucht werden soll.</p> <p>Wenn ein leerer Text angegeben wird, wird die Seite ermittelt, auf der sich das augenblicklich aktive Eingabeobjekt befindet.</p>

<b>GetPageBgrMode</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt die Art des Seitenhintergrundes einer Seite.</p> <p><b>Funktionswert</b> 1 Keiner 2 Farbe 3 Bitmap -1 Fehler</p> <p><b>Verwandte Funktionen</b> <i>GetPageBgrColor</i> <i>GetPageHeight</i></p>	<p><b>Name</b> Textkonstante oder –Variable, die den Namen der Seite angibt.</p>

<b>GetPageBgrColor</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt die Farbe des Seitenhintergrundes einer Seite.</p> <p><b>Bemerkung</b> Die Farbe ist nur dann wirksam, wenn der Modus für den Seitenhintergrund den Wert 2 hat.</p> <p><b>Funktionswert</b> 1 Keiner 2 Farbe 3 Bitmap -1 Fehler</p> <p><b>Verwandte Funktionen</b> <i>GetPageBgrMode</i> <i>GetPageHeight</i></p>	<p><b>Name</b> Textkonstante oder –Variable, die den Namen der Seite angibt.</p> <p><b>Red</b> Ganzzahlige Variable, keine Konstante, auf der Farbwert für <i>Rot</i> zurückgeliefert wird.</p> <p><b>Green</b> Ganzzahlige Variable, keine Konstante, auf der Farbwert für <i>Grün</i> zurückgeliefert wird.</p> <p><b>Blue</b> Ganzzahlige Variable, keine Konstante, auf der Farbwert für <i>Blau</i> zurückgeliefert wird.</p>

<b>GetPageDisplayRect</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt das umgebende Rechteck einer Seite in Screen Pixels.</p> <p><b>Funktionswert</b>  <b>0</b> Die Koordinaten der Seite wurden ermittelt  <b>-1</b> Die Seite wurde nicht gefunden.</p> <p><b>Bemerkungen</b> Die Koordinaten sind dokumentenrelativ. Das Dokument ist die Gesamtheit aller Seiten einschließlich der Zwischenräume beim Anzeigen.</p> <p><b>Verwandte Funktionen</b> <i>GetObjectDisplayRect</i></p>	<p><b>Name</b> Textkonstante oder –Variable, die den Namen der Seite angibt.</p> <p><b>Left</b> Ganzzahlige Variable, keine Konstante, auf der linke Rand der Seite zurückgeliefert wird.</p> <p><b>Top</b> Ganzzahlige Variable, keine Konstante, auf der der obere Rand der Seite zurückgeliefert wird.</p> <p><b>Right</b> Ganzzahlige Variable, keine Konstante, auf der der rechte Rand der Seite zurückgeliefert wird.</p> <p><b>Bottom</b> Ganzzahlige Variable, keine Konstante, auf der untere Rand der Seite zurückgeliefert wird.</p>
<p><b>Beispiel</b></p> <p>Das folgende Beispiel rollt das Formular in vertikaler Richtung so weit, dass der obere Rand der Seite mit dem Namen "MyPage" am oberen Fensterrand zu stehen kommt.</p> <pre style="color: #008000;">// Variablendeklarationen int left; int top; int right; int bottom;  // Ermittle Position der Seite GetPageDisplayRect ("zwei", left, top, right, bottom);  // Setze Rollposition SetScrollOffset (VERT, top, 0);</pre>	

<b>GetPageHeight</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt die Höhe einer Seite in Mikrometern.</p> <p><b>Funktionswert</b>  <b>n</b> Höhe der Seite in Mikrometern  <b>0</b> Die Seite wurde nicht gefunden</p>	<p><b>PageName</b> Textkonstante oder -variable, die den Namen der Seite angibt, deren Höhe ermittelt werden soll.</p>

<b>GetPageHeightZoomFactor</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt den Zoomfaktor in Prozent, der eingestellt werden muss, damit die volle Seitenhöhe des Formulars im aktuellen Dokumentfenster sichtbar ist.</p> <p><b>Funktionswert</b> Zoomfaktor in Prozent.</p> <p><b>Verwandte Funktionen</b> <i>GetPageWidthZoomFactor</i></p>	<p>Die Funktion hat keine Parameter</p>

<b>GetPageNrByPageName</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt die Nummer der ersten Seite des Formulars, deren Name mit dem angegebenen Namen übereinstimmt.</p> <p><b>Funktionswert</b>  <b>n</b> Nummer der Seite (Zählung ab 1)  <b>0</b> Es wurde keine Seite gefunden</p> <p><b>Verwandte Funktionen</b> <i>GetPage</i></p>	<p><b>ObjectName</b> Textkonstante oder -variable, die den Namen der Seite angibt, deren Nummer ermittelt werden soll.</p> <p>Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p>

<b>GetPageWidth</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt die Breite einer Seite in Mikrometern.</p> <p><b>Funktionswert</b>  <b>n</b> Breite der Seite in Mikrometern  <b>0</b> Die Seite wurde nicht gefunden</p>	<p><b>PageName</b> Textkonstante oder -variable, die den Namen der Seite angibt, deren Breite ermittelt werden soll.</p>

<b>GetPageWidthZoomFactor</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt den Zoomfaktor in Prozent, der eingestellt werden muss, damit die volle Seitenbreite des Formulars im aktuellen Dokumentfenster sichtbar ist.</p> <p><b>Funktionswert</b> Zoomfaktor in Prozent.</p> <p><b>Verwandte Funktionen</b> <i>GetPageHeightZoomFactor</i></p>	<p>Die Funktion hat keine Parameter</p>

<b>GetPointedObject</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion liefert den Namen des Objekts, auf das der Mauszeiger augenblicklich zeigt.</p> <p><b>Funktionswert</b>  <b>0</b> Der Name des Objekts wurde ermittelt  <b>-1</b> Der Mauszeiger zeigt auf kein Objekt</p> <p><b>Verwandte Funktionen</b> <i>GetCurrentObject</i> <i>GetPointedObjectHandle</i></p>	<p><b>Result</b> Textvariable, keine Konstante, auf der der Name des Objekts zurückgeliefert wird.</p>

<b>GetPointedObjectHandle</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion liefert den Handle des Objekts, auf das der Mauszeiger augenblicklich zeigt.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"> <li><b>n</b> Der Handle des Objekts</li> <li><b>0</b> Der Mauszeiger zeigt auf kein Objekt</li> </ul> <p><b>Verwandte Funktionen</b></p> <p><i>GetCurrentObject</i> <i>GetPointedObject</i> <i>FindObject</i></p>	<p>Die Funktion hat keine Parameter</p>

<b>GetPrintState</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt, ob für ein Objekt das Attribut <i>Drucken</i> gesetzt ist oder nicht. Objekte, die dieses Attribut nicht gesetzt haben, werden zwar angezeigt, jedoch nicht gedruckt.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"> <li><b>1</b> Objekt wird gedruckt</li> <li><b>0</b> Objekt wird nicht gedruckt</li> <li><b>-1</b> Objekt nicht gefunden</li> </ul> <p><b>Bemerkung</b> Es wird der Status des ersten Objekts mit dem angegebenen Namen auf der aktuellen Seite ermittelt.</p> <p>Wenn auf der aktuellen Seite ein solches Objekt nicht gefunden wird, wird die Suche auf das gesamte Formular ausgedehnt.</p> <p><b>Verwandte Funktionen</b></p> <p><i>SetPrintState</i></p>	<p><b>ObjectName</b> Textvariable oder -konstante, die den Namen des Objekts angibt, dessen Status ermittelt werden soll.</p> <p>Wenn ein leerer Text angegeben wird, wird der Status des augenblicklich aktiven Objekts ermittelt.</p>

<b>GetReadOnlyState</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt, ob für ein Objekt das Attribut <i>Nur Lesen</i> gesetzt ist oder nicht.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"> <li><b>1</b> Objekt kann nur gelesen werden</li> <li><b>0</b> Objekt kann gelesen und geschrieben werden</li> <li><b>-1</b> Objekt nicht gefunden</li> </ul> <p><b>Bemerkung</b> Es wird der Status des ersten Objekts mit dem angegebenen Namen auf der aktuellen Seite ermittelt.</p> <p>Wenn auf der aktuellen Seite ein solches Objekt nicht gefunden wird, wird die Suche auf das gesamte Formular ausgedehnt.</p> <p><b>Verwandte Funktionen</b> <i>SetReadOnlyState</i></p>	<p><b>ObjectName</b> Textvariable oder -konstante, die den Namen des Objekts angibt, dessen Status ermittelt werden soll.</p> <p>Wenn ein leerer Text angegeben wird, wird der Status des augenblicklich aktiven Objekts ermittelt.</p>

<b>GetRuntimeChangeFlag</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion liefert die interne Merkvariable zurück, die anzeigt, ob eine bestimmte Seite zur Laufzeit verändert wurde.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"> <li><b>1</b> Seite wurde geändert</li> <li><b>0</b> Seite wurde nicht geändert</li> </ul> <p><b>Verwandte Funktionen</b> <i>SetRuntimeChangeFlag</i> <i>CountRuntimeChangedPages</i> <i>GetPageNrByPageName</i></p>	<p><b>pageNr</b> Ganzzahlige Konstante oder Variable, die die Seitennummer der Seite bezeichnet, deren Merkvariable gelesen werden soll.</p> <p>Die Seitenzählung beginnt immer bei 1.</p>



<b>GetScrollOffset</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion liefert die augenblickliche Scroll-Position zurück.</p> <p><b>Funktionswert</b> Rollposition in Pixeln.</p> <p><b>Bemerkungen</b> Wenn das Formular ganz nach oben (bzw. links) gerollt ist, ist der Offset 0.</p> <p><b>Verwandte Funktionen</b> <i>SetScrollOffset</i></p>	<p><b>Dimension</b> Ganzzahlige Konstante oder Variable, die angibt, welche der Rollrichtungen zurückgeliefert werden soll:</p> <p>VERT vertikaler Offset HORI horizontaler Offset</p>

<b>GetStringProperty</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion liest eine von verschiedenen möglichen <i>Text-Eigenschaften</i> eines Formularobjekts aus.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Nicht alle Eigenschaften haben für jeden Objekttyp eine Bedeutung. Manche Eigenschaften haben nur für bestimmte Objekttypen einen Sinn.</p> <p>Wenn Eigenschaften benannt werden, die es zwar prinzipiell gibt, aber die für den angegebenen Objekttyp keine Bedeutung haben, erfolgt keine Fehlermeldung, aber der Aufruf der Funktion ist ohne Wirkung.</p> <p>Wenn Eigenschaften benannt werden, die nicht definiert sind, erfolgt eine Fehlermeldung.</p> <p><b>Verwandte Funktionen</b> <i>SetStringProperty</i> <i>GetLongProperty</i> <i>GetColorProperty</i> <i>GetXmlProperty</i></p>	<p><b>Handle</b> Ganzzahlige Konstante oder Variable, die den Handle des Objekts angibt, das angesprochen werden soll.</p> <p><b>Property</b> Textkonstante oder -variable, die den Namen der Eigenschaft angibt, die gelesen werden soll.</p> <p>Folgende Eigenschaften sind definiert:</p> <p><b>Name</b> Name des Objekts</p> <p><b>AccessObjectName</b> Name des assoziierten Zugriffsobjekts</p> <p><b>DataSourceFile</b> Name und Pfad der Datenquelle</p> <p><b>DataFieldName</b> Name des assoziierten Datenfeldes</p> <p><b>TypeName</b> Typ des Objekts als Anzeigename</p> <p><b>NewValue</b> Textvariable, keine Konstante, auf der die ausgelesene Eigenschaft abgelegt wird.</p>
<p><b>Beispiel</b> Nach den Befehlen:</p> <pre style="color: green;">char Text [100]; int Handle;  Handle = FindObject ("", "Eingabe", 0); GetStringProperty (Handle, "DataFieldName", Text);</pre> <p>enthält die Textvariable <i>Text</i> den Namen des Datenfeldes des ersten Objekts des Formulars, das den Namen <i>Eingabe</i> trägt.</p>	

<b>GetText</b>									
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt den Text eines beliebigen Objekts.</p> <p><b>Funktionswert</b>  <b>0</b> Der Text des Objekts wurde ermittelt  <b>-1</b> Kein Objekt dieses Namens gefunden</p> <p><b>Bemerkungen</b> Der Text eines Objekts hängt vom Typ des Objekts ab. Hier einige Beispiele:</p> <table style="margin-left: 20px;"> <tr> <td><i>Eingabefeld</i></td> <td>augenblickliche Eingabe</td> </tr> <tr> <td><i>Text</i></td> <td>Ausgabertext</td> </tr> <tr> <td><i>Datum</i></td> <td>Textdarstellung des Datums</td> </tr> <tr> <td><i>Ankreuzfeld</i></td> <td>Text, der dem Ankreuzzustand zugeordnet wurde</td> </tr> </table> <p>Wenn mehr als ein Objekt mit dem angegebenen Namen existiert, wird immer der Text des <i>ersten gefundenen</i> Objekts gelesen. Dies ist ein wesentlicher Unterschied zur Schwesterfunktion <i>SetText</i>, welche immer <b>alle</b> betroffenen Objekte verändert.</p> <p><b>Verwandte Funktionen</b> <i>SetText</i></p>	<i>Eingabefeld</i>	augenblickliche Eingabe	<i>Text</i>	Ausgabertext	<i>Datum</i>	Textdarstellung des Datums	<i>Ankreuzfeld</i>	Text, der dem Ankreuzzustand zugeordnet wurde	<p><b>ObjectName</b> Textkonstante oder -variable, die den Namen des Objekts angibt, dessen Text ermittelt werden soll.</p> <p><b>Result</b> Textvariable, keine Konstante, auf der der ermittelte Text zurückgeliefert wird.</p>
<i>Eingabefeld</i>	augenblickliche Eingabe								
<i>Text</i>	Ausgabertext								
<i>Datum</i>	Textdarstellung des Datums								
<i>Ankreuzfeld</i>	Text, der dem Ankreuzzustand zugeordnet wurde								

<b>GetTextLng</b>									
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt die Länge des Textes eines beliebigen Objekts.</p> <p><b>Funktionswert</b>  <b>n</b> Der Text ist n Zeichen lang  <b>-1</b> Es wurde kein Objekt dieses Namens gefunden</p> <p><b>Bemerkungen</b> Die Bedeutung des <i>Textes</i> eines Objekts hängt vom Typ des Objekts ab. Hier einige Beispiele:</p> <table style="margin-left: 20px;"> <tr> <td><i>Eingabefeld</i></td> <td>augenblickliche Eingabe</td> </tr> <tr> <td><i>Text</i></td> <td>Ausgabertext</td> </tr> <tr> <td><i>Datum</i></td> <td>Textdarstellung des Datums</td> </tr> <tr> <td><i>Ankreuzfeld</i></td> <td>Text, der dem Ankreuzzustand zugeordnet wurde</td> </tr> </table> <p>Wenn mehr als ein Objekt mit dem angegebenen Namen existiert, wird immer die Länge des Textes des ersten gefundenen Objekts ermittelt.</p> <p><b>Verwandte Funktionen</b> <i>GetText</i></p>	<i>Eingabefeld</i>	augenblickliche Eingabe	<i>Text</i>	Ausgabertext	<i>Datum</i>	Textdarstellung des Datums	<i>Ankreuzfeld</i>	Text, der dem Ankreuzzustand zugeordnet wurde	<p><b>ObjectName</b> Textkonstante oder -variable, die den Namen des Objekts angibt, dessen Textlänge ermittelt werden soll.</p>
<i>Eingabefeld</i>	augenblickliche Eingabe								
<i>Text</i>	Ausgabertext								
<i>Datum</i>	Textdarstellung des Datums								
<i>Ankreuzfeld</i>	Text, der dem Ankreuzzustand zugeordnet wurde								

<b>GetTitleFile</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt den Dateinamen des Formulars, welches gerade in Bearbeitung ist.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Wenn das Formular ein Dokument <i>ohne Namen</i> ist, wird ein leerer Text zurückgeliefert.</p>	<p><b>Result</b> Textvariable, keine Konstante, auf der der Dateiname zurückgeliefert wird.</p>

<b>GetTotalPages</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion liefert die Anzahl der Seiten des Formulars zurück.</p> <p><b>Funktionswert</b> Anzahl der Seiten</p>	<p>Die Funktion hat keine Parameter</p>

<b>GetVertAlign</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt die vertikale Ausrichtung eines Objekts.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"> <li>1 Objekt ist am oberen Rand ausgerichtet</li> <li>2 Objekt ist zentriert</li> <li>3 Objekt ist am unteren Rand ausgerichtet</li> <li>-1 Objekt nicht gefunden</li> </ul> <p><b>Bemerkung</b> Es wird die Ausrichtung des ersten Objekts mit dem angegebenen Namen auf der aktuellen Seite ermittelt.</p> <p>Wenn auf der aktuellen Seite ein solches Objekt nicht gefunden wird, wird die Suche auf das gesamte Formular ausgedehnt.</p> <p><b>Verwandte Funktionen</b> GetVertAlign SetHoriAlign</p>	<p><b>ObjectName</b> Textvariable oder -konstante, die den Namen des Objekts angibt, dessen Ausrichtung ermittelt werden soll.</p> <p>Wenn ein leerer Text angegeben wird, wird die Ausrichtung des augenblicklich aktiven Objekts ermittelt.</p>

<b>GetXmlProperty</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion liest eine beliebige Objekt-Eigenschaft als XML-Wert.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Der Name der Property entspricht dem Namen des <i>XML-Tags</i> bzw. des <i>XML-Attributs</i> unter dem die gewünschte Eigenschaft in XML-Dateien abgespeichert wird.</p> <p>Wenn Eigenschaften benannt werden, die es zwar prinzipiell gibt, aber die für den angegebenen Objekttyp keine Bedeutung haben, erfolgt keine Fehlermeldung, aber der Aufruf der Funktion ist ohne Wirkung.</p> <p><b>Verwandte Funktionen</b> <i>SetXmlProperty</i> <i>GetLongProperty</i> <i>GetColorProperty</i> <i>GetStringProperty</i></p>	<p><b>Handle</b> Ganzzahlige Konstante oder Variable, die den Handle des Objekts angibt, das angesprochen werden soll.</p> <p><b>Property</b> Textkonstante oder -variable, die den Namen der Eigenschaft angibt, die gelesen werden soll.</p> <p><b>NewValue</b> Textvariable, keine Konstante, auf der die ausgelesene Eigenschaft abgelegt wird.</p>
<p><b>Beispiel</b> Nach den Befehlen:</p> <pre style="color: green;">char Text [100]; int Handle;  Handle = FindObject ("", "Eingabe", 0); GetXmlProperty (Handle, "DataField", Text);</pre> <p>enthält die Textvariable <i>Text</i> den Namen des Datenfeldes des ersten Objekts des Formulars, das den Namen <i>Eingabe</i> trägt.</p>	

<b>GetZoom</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion liefert den augenblicklichen Zoomfaktor in Prozent zurück.</p> <p><b>Funktionswert</b> Zoomfaktor in Prozent</p> <p><b>Verwandte Funktionen</b> <i>SetZoom</i></p>	<p>Die Funktion hat keine Parameter</p>

<b>GotoObject</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt den Cursor in ein bestimmtes Objekt des Formulars.</p> <p><b>Diese Funktion darf nur innerhalb der Makros verwendet werden, die für den Vor- und Rückwärtssprung in Eingabeobjekten eingetragen werden können. Bei Verwendung außerhalb dieser Makros kann es zu Fehlersituationen kommen.</b></p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Verwandte Funktionen</b> <i>GotoRelative</i></p>	<p><b>ObjectName</b> Textvariable oder -konstante, die den Namen des Objekts angibt, auf das gesprungen werden soll.</p>

<b>GotoRelative</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt den Cursor in ein Objekt welches sich um eine bestimmte Anzahl von Cursorsprüngen vor oder hinter dem augenblicklichen Objekt befindet.</p> <p><b>Diese Funktion darf nur innerhalb der Makros verwendet werden, die für den Vor- und Rückwärtssprung in Eingabeobjekten eingetragen werden können. Bei Verwendung außerhalb dieser Makros kann es zu Fehlersituationen kommen.</b></p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Verwandte Funktionen</b> <i>GotoObject</i></p>	<p><b>Count</b> Ganzzahlige Konstante oder Variable, die angibt, um wieviele Objekte vorwärts oder rückwärts gesprungen werden soll.</p> <p>Rückwärtssprünge werden durch einen negativen Wert dargestellt.</p>
<b>HideMenu</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion macht das Menu des Ausfüllers sichtbar oder unsichtbar.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Verwandte Funktionen</b> <i>NoCloseOnEscape</i></p>	<p><b>Mode</b> Ganzzahlige Konstante oder Variable, die angibt, ob das Menü sichtbar ist, oder nicht:</p> <p>1 = unsichtbar 0 = sichtbar</p>



<b>InsertModule</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion den Inhalt einer Datei als Textbaustein in ein Eingabefeld ein.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung</p>	<p><b>ObjectName</b> Textkonstante oder -variable, die den Namen des Eingabeobjektes angibt, in das der Baustein eingefügt werden soll. Ein leerer Text bedeutet, dass das augenblicklich aktive Eingabeobjektes gemeint ist.</p> <p><b>FileName</b> Textkonstante oder -variable, die den Namen der Datei angibt, in der sich der Baustein befindet.</p>

<b>LoadBitmap</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion lädt eine Bitmapdatei in ein Bildobjekt.</p> <p><b>Funktionswert</b> 0 Die Bitmap wurde korrekt geladen -1 Es ist ein Fehler aufgetreten</p> <p><b>Bemerkung</b> Wenn das Formular mehrere Bitmapobjekte mit dem angegebenen Namen beinhaltet, wird in alle diese Objekte die neue Bitmap geladen.</p> <p><b>Verwandte Funktionen</b> <i>SetPictureFile</i></p>	<p><b>ObjectName</b> Textkonstante oder -variable, die den Namen des Bitmapobjekts angibt.</p> <p><b>Datei</b> Textkonstante oder -variable, die den Dateinamen angibt, in dem sich die Bitmap befindet.</p>

<b>LoadFromProfile</b>	
<p><b>Funktionsbeschreibung</b>                      Diese Funktion liest den Inhalt von benannten Eingabeobjekten aus einer Profile-Datei. Es werden nur diejenigen Objekte beeinflusst, deren Name ausdrücklich in der Profile-Datei genannt werden. Alle anderen Eingabeobjekte bleiben unverändert.</p> <p><b>Funktionswert</b>                      Die Funktion liefert keinen Funktionswert.</p> <p><b>Bemerkungen</b>                      Die Profile-Datei muss eine Sektion mit dem Namen <i>SECTION_OBJECTS</i> beinhalten. In dieser Sektion können beliebig viele Items stehen. Immer wenn der Name des Items mit dem Namen eines Eingabeobjekts übereinstimmt, wird der Wert des Items in das Eingabeobjekt übertragen.</p> <p><b>Verwandte Funktionen</b>  <i>SaveToProfile</i></p>	<p><b>FileName</b>                      Textkonstante oder -variable, die den Namen der Profile-Datei angibt.</p>

<b>NewForm</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion öffnet ein neues MDI-Fenster und lädt die angegebene Formularvorlage als neues, unbetitelttes Formular. Alle Makrobefehle, die nach diesem Befehl ausgeführt werden, beziehen sich auf das neu geöffnete - also aktive - Formular.</p> <p><b>Bemerkungen</b> Der Makrobefehl kann nur im Ausfüller verwendet werden.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Verwandte Funktionen</b> <i>Open</i> <i>Close</i></p>	<p><b>FileName</b> Textkonstante oder -variable, die den Namen der Formularvorlage enthält, die geladen werden soll. Ein leerer Text bewirkt, dass der Name mittels eines Dialoges angefragt wird.</p>

<b>NoCloseOnEscape</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion schaltet die Eigenschaft des Ausfüllers, beim Drücken der ESC-Taste das aktuelle Fenster zu schließen, aus oder ein.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Verwandte Funktionen</b> <i>HideMenu</i></p>	<p><b>Mode</b> Ganzzahlige Konstante oder Variable, die angibt, ob die ESC-Taste das Fenster schließt, oder nicht:</p> <p>1 = nicht schließen 0 = schließen</p>

<b>Open</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion öffnet ein neues MDI-Fenster und lädt das angegebene, bereits existierende Formular als Titeldatei. Alle Makrobefehle, die nach diesem Befehl ausgeführt werden, beziehen sich auf das neu geöffnete - also aktive - Formular.</p> <p><b>Bemerkungen</b> Der Makrobefehl kann nur im Ausfüller verwendet werden.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Verwandte Funktionen</b> <i>NewForm</i> <i>Close</i></p>	<p><b>FileName</b> Textkonstante oder -variable, die den Namen des Formulars enthält, das geöffnet werden soll. Ein leerer Text bewirkt, dass der Name mittels eines Dialoges angefragt wird.</p>

<b>PostFormMacro</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt eine interne Windows-Message ab, die später – zu einer zeitlich verzögerten Ausführung – ein für das Formular hinterlegtes Makro aufruft.</p> <p><b>Funktionswert</b> Die Funktion liefert immer den Wert 0.</p> <p><b>Bemerkungen</b> Diese Funktion sollte immer dann anstelle der Funktion <i>CallObjectMacro</i> verwendet werden, wenn das Objekt, das die Funktion auslöst (z.B. eine Schaltfläche) durch das auszuführende Makro aus dem Formular entfernt wird. Dadurch wird verhindert, dass das mit dem entfernten Objekt noch weitere Operationen seitens des Formulars durchgeführt werden, wie z.B. die Aktionen, die beim Loslassen der Maustaste ausgeführt werden.</p> <p><b>Verwandte Funktionen</b> <i>PostObjectMacro</i> <i>CallFormMacro</i> <i>CallObjectMacro</i></p>	<p><b>MacroName</b> Textkonstante oder -variable, die den Namen des Makros angibt, das aufgerufen werden soll.</p> <p>Die Liste aller Makros ist im Anhang C (Formularmakros) beschrieben.</p>

<b>PostObjectMacro</b>	
<p><b>Funktionsbeschreibung</b>                      Diese Funktion setzt eine interne Windows-Message ab, die später – zu einer zeitlich verzögerten Ausführung – ein für ein bestimmtes Objekt hinterlegtes Makro aufruft.</p> <p><b>Funktionswert</b>                      Die Funktion liefert immer den Wert 0.</p> <p><b>Bemerkungen</b>                      Diese Funktion sollte immer dann anstelle der Funktion <i>CallObjectMacro</i> verwendet werden, wenn das Objekt, das die Funktion auslöst (z.B. eine Schaltfläche) durch das auszuführende Makro aus dem Formular entfernt wird. Dadurch wird verhindert, dass das mit dem entfernten Objekt noch weitere Operationen seitens des Formulars durchgeführt werden, wie z.B. die Aktionen, die beim Loslassen der Maustaste ausgeführt werden.</p> <p><b>Verwandte Funktionen</b>  <i>PostFormMacro</i>  <i>CallFormMacro</i>  <i>CallObjectMacro</i></p>	<p><b>ObjectName</b>                      Textkonstante oder -variable, die den Namen des Button-Objekts angibt, dessen Makro aufgerufen werden soll.</p> <p><b>MacroName</b>                      Textkonstante oder -variable, die den Namen des Makrotyps angibt, der aufgerufen werden soll.</p> <p>Die Liste aller Makros ist im Anhang D (Objektmakros) beschrieben.</p> <p>Es sind nicht alle Makrotypen für alle Objekttypen definiert oder sinnvoll.</p>

<b>Print</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion druckt das augenblicklich aktive Formular.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Wenn alle Seiten gedruckt werden sollen, muss für den Seitenbereich 0 - 9999 angegeben werden.</p>	<p><b>Dialog</b> Ganzzahlige Konstante oder Variable, die festlegt, ob das Drucken stumm erfolgen soll oder ob zuvor der Drucken-Dialog eröffnet wird:</p> <p>    <b>0</b> keinen Dialog eröffnen     <b>1</b> Dialog eröffnen</p> <p><b>FromPage</b> Ganzzahlige Konstante oder Variable, die die Nummer der ersten zu druckenden Seite (Zählung ab 1) angibt. Mit 0 wird die aktuelle Seite angesprochen.</p> <p><b>ToPage</b> Ganzzahlige Konstante oder Variable, die die Nummer der letzten zu druckenden Seite (Zählung ab 1) angibt. Mit 0 wird die aktuelle Seite angesprochen.</p> <p><b>OnlyContent</b> Ganzzahlige Konstante oder Variable, die bestimmt, ob das gesamte Formular oder nur die Inhalte der Eingabeobjekte gedruckt werden sollen:</p> <p>    <b>0</b> Inhalte und Vordruck     <b>1</b> nur Inhalte drucken</p> <p><b>ListFile</b> Ist für spätere Zwecke reserviert. Es muss immer "" angegeben werden.</p> <p><b>Type</b> Ganzzahlige Konstante oder Variable. Ist für spätere Zwecke reserviert. Muss 0 sein.</p>

Quit	
<b>Funktionsbeschreibung</b> Diese Funktion beendet die Formularanwendung. Das Makro wirkt nur in der Anwendung <i>Ausfüller</i> .	Die Funktion hat keine Parameter.

  

RefreshForm	
<b>Funktionsbeschreibung</b> Diese Funktion bewirkt eine vollständige Neuanzeige des augenblicklichen Formulars.	Die Funktion hat keine Parameter.
<b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.	

  

RefreshObject	
<b>Funktionsbeschreibung</b> Diese Funktion bewirkt eine vollständige Neuanzeige des angegebenen Objekts.	<b>Handle</b> Ganzzahlige Konstante oder Variable, die den Handle des Objekts angibt, das neu angezeigt werden soll.
<b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.	



Save	
<p><b>Funktionsbeschreibung</b>                      Diese Funktion speichert das aktuelle Formular die Titel-Datei. Wenn das Formular ohne Titel ist, schlägt diese Funktion fehl. Es muss dann die Funktion <i>SaveAs</i> verwendet werden.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"> <li><b>0</b> Das Formular wurde gespeichert</li> <li><b>-1</b> Das Formular ist <i>ohne Titel</i> oder ein anderer Fehler ist aufgetreten.</li> </ul> <p><b>Verwandte Funktionen</b>  <i>SaveAs</i></p>	<p>Die Funktion hat keine Parameter</p>

SaveAs	
<p><b>Funktionsbeschreibung</b>                      Diese Funktion speichert ein Formular in eine vorgegebene Datei. Die Datei wird anschließend zur Titeldatei.</p> <p><b>Funktionswert</b></p> <ul style="list-style-type: none"> <li><b>0</b> Das Formular wurde gespeichert.</li> <li><b>-1</b> Es ist ein Fehler aufgetreten.</li> </ul> <p><b>Verwandte Funktionen</b>  <i>Save</i></p>	<p><b>FileName</b>                      Textkonstante oder -variable, die den Namen der Datei angibt, in der das Formular gespeichert werden soll.</p>

<b>SaveToProfile</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion schreibt den Inhalt aller benannten Eingabeobjekte in eine Profile-Datei.</p> <p><b>Funktionswert</b> Die Funktion liefert keinen Funktionswert.</p> <p><b>Bemerkungen</b> In der Profile-Datei wird eine Sektion mit dem Namen <i>SECTION_OBJECTS</i> angelegt. In dieser Sektion werden alle Eingabeobjekte abgelegt, wobei der Name des Objekts als Item-Name und der Inhalt als Item-Wert verwendet werden.</p> <p><b>Verwandte Funktionen</b> <i>LoadFromProfile</i></p>	<p><b>FileName</b> Textkonstante oder -variable, die den Namen der Profile-Datei angibt.</p>

<b>SetActiveState</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt den Status <i>aktiv</i> für ein Objekt oder für eine Gruppe von Objekten.</p> <p>Momentan ist dieser Status nur für Objekte vom Typ <i>Button</i> implementiert und repräsentiert den Zustand <i>gedrückt</i>.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Es wird der Status aller Objekte geändert, die zu dem angegebenen Objektnamen passen und die auf einer Seite mit dem angegebenen Seitennamen liegen.</p> <p><b>Verwandte Funktionen</b> <i>GetActiveState</i></p>	<p><b>PageName</b> Textkonstante oder -variable, die den Namen der Seiten angibt, auf denen Objekte geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B). Ein leerer Text bezeichnet die aktuelle Seite.</p> <p><b>ObjectName</b> Textkonstante oder -variable, die den Namen der Objekte angibt, die geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>State</b> Ganzzahlige Konstante oder Variable, die den neuen Status angibt:</p> <p style="margin-left: 40px;"><b>1</b> <i>aktiv</i> setzen <b>0</b> <i>inaktiv</i> setzen</p>

<b>SetBgrColor</b>	
<p><b>Funktionsbeschreibung</b> Dieser Befehl setzt eine neue Hintergrundfarbe für ein Objekt oder eine Gruppe von Objekten.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Es wird die Hintergrundfarbe aller Objekte geändert, die zu dem angegebenen Objektnamen passen und die auf einer Seite mit dem angegebenen Seitennamen liegen.</p> <p><b>Verwandte Funktionen</b> <i>SetFontColor</i></p>	<p><b>PageName</b> Textkonstante oder -variable, die den Namen der Seiten angibt, auf denen Objekte geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B). Ein leerer Text bezeichnet die aktuelle Seite.</p> <p><b>ObjectName</b> Textkonstante oder -variable, die den Namen der Objekte angibt, die geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>Type</b> Ganzzahlige Konstante oder Variable. Ist für spätere Anwendungen reserviert. Muss 0 sein.</p> <p><b>Red</b> Ganzzahlige Konstante oder Variable, die den RGB-Wert für die Farbe Rot zwischen 0 und 255 angibt.</p> <p><b>Green</b> Ganzzahlige Konstante oder Variable, die den RGB-Wert für die Farbe Grün zwischen 0 und 255 angibt.</p> <p><b>Blue</b> Ganzzahlige Konstante oder Variable, die den RGB-Wert für die Farbe Blau zwischen 0 und 255 angibt.</p>

<b>SetCapsState</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt den Status <i>Großschreibung</i> für ein Objekt oder für eine Gruppe von Objekten.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkung</b> Es wird die <i>Großschreibung</i> aller Objekte des Formulars geändert, die zu dem angegebenen Objektamen passen.</p> <p>Wenn dies nur auf bestimmten Seiten geschehen soll, so kann die Funktion <i>SetCapsStateOnPage</i> verwendet werden.</p> <p><b>Verwandte Funktionen</b> <i>SetCapsStateOnPage</i></p>	<p><b>ObjectName</b> Textkonstante oder -variable, die den Namen der Objekte angibt, die geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>State</b> Ganzzahlige Konstante oder Variable, die den neuen Status angibt:</p> <p><b>1</b> Großschreibung ein <b>0</b> Großschreibung aus</p>

<b>SetCapsStateOnPage</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt den Status <i>Großschreibung</i> für ein Objekt oder eine Gruppe von Objekten.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Es wird der Status aller Objekte geändert, die zu dem angegebenen Objektnamen passen und die auf einer Seite mit dem angegebenen Seitennamen liegen.</p> <p><b>Verwandte Funktionen</b> <i>SetCapsState</i></p>	<p><b>PageName</b> Textkonstante oder -variable, die den Namen der Seiten angibt, auf denen Objekte geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B). Ein leerer Text bezeichnet die aktuelle Seite.</p> <p><b>ObjectName</b> Textkonstante oder -variable, die den Namen der Objekte angibt, die geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>State</b> Ganzzahlige Konstante oder Variable, die den neuen Status angibt:</p> <p style="margin-left: 40px;"><b>1</b> Großschreibung ein <b>0</b> Großschreibung aus</p>

<b>SetChangeFlag</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt die interne Merkvariable, die angibt, ob ein Formular geändert wurde.</p> <p><b>Funktionswert</b> Als Funktionswert wird der augenblickliche Status zurückgemeldet.</p> <p><b>Verwandte Funktionen</b> <i>GetChangeFlag</i></p>	<p><b>State</b> Ganzzahlige Konstante oder Variable, die den neuen Wert der internen Merkvariable angibt:</p> <p style="margin-left: 40px;"><b>0</b> Formular ist unverändert <b>1</b> Formular wurde geändert</p>

<b>SetCheckbox</b>	
<p><b>Funktionsbeschreibung</b> Dieses Makro setzt den Zustand eines Ankreuzfeldes.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Es wird der Zustand aller Ankreuzfelder des Formulars geändert, die zu dem angegebenen Objektnamen passen.</p> <p>Wenn eine Checkbox auf einer bestimmten Seite gesetzt werden soll, kann die Funktion <i>SetCheckboxOnPage</i> verwendet werden.</p> <p><b>Verwandte Funktionen</b> <i>SetCheckboxOnPage</i></p>	<p><b>ObjectName</b> Textkonstante oder -variable, die den Namen der Objekte angibt, die geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>State</b> Ganzzahlige Konstante oder Variable, die den neuen Zustand des Ankreuzfeldes angibt:</p> <p><b>1</b> ankreuzen <b>0</b> nicht ankreuzen</p>

<b>SetCheckboxOnPage</b>	
<p><b>Funktionsbeschreibung</b> Dieses Makro setzt den Zustand eines Ankreuzfeldes.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Es wird der Zustand aller Ankreuzfelder geändert, die zu dem angegebenen Objektnamen passen und die auf einer Seite mit dem angegebenen Seitennamen liegen.</p> <p><b>Verwandte Funktionen</b> <i>SetCheckbox</i></p>	<p><b>PageName</b> Textkonstante oder -variable, die den Namen der Seiten angibt, auf denen Objekte geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B). Ein leerer Text bezeichnet die aktuelle Seite.</p> <p><b>ObjectName</b> Textkonstante oder -variable, die den Namen der Objekte angibt, die geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>State</b> Ganzzahlige Konstante oder Variable, die den neuen Zustand des Ankreuzfeldes angibt:</p> <p style="text-align: right;"><b>1</b> ankreuzen <b>0</b> nicht ankreuzen</p>

<b>SetColorProperty</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt eine von verschiedenen möglichen <i>Farb-Eigenschaften</i> eines Formularobjekts.</p> <p><b>Funktionswert</b>  <b>0</b> Die Farbe wurde korrekt gesetzt  <b>-1</b> Fehler oder Objekt unbekannt</p> <p><b>Bemerkungen</b> Nicht alle Eigenschaften haben für jeden Objekttyp eine Bedeutung. Manche Eigenschaften haben nur für bestimmte Objekttypen einen Sinn.  Wenn Eigenschaften benannt werden, die es zwar prinzipiell gibt, aber die für den angegebenen Objekttyp keine Bedeutung haben, erfolgt keine Fehlermeldung, aber der Aufruf der Funktion ist ohne Wirkung.  Wenn Eigenschaften benannt werden, die nicht definiert sind, erfolgt eine Fehlermeldung.</p> <p><b>Verwandte Funktionen</b>  <i>GetColorProperty</i>  <i>SetLongProperty</i>  <i>SetStringProperty</i></p>	<p><b>Handle</b> Ganzzahlige Konstante oder Variable, die den Handle des Objekts angibt, das angesprochen werden soll.</p> <p><b>Property</b> Textkonstante oder -variable, die den Namen der Eigenschaft angibt, die gesetzt werden soll.  Folgende Eigenschaften sind definiert:   <b>ColorBgr</b>  <b>ColorFont</b>  <b>ColorFrame</b>  <b>ColorTempDisp</b>  <b>ColorCheckCross</b></p> <p><b>Red</b> Ganzzahlige Konstante oder Variable, auf der Rotanteil der Farbe als Wert zwischen 0 und 255 übergeben wird.</p> <p><b>Green</b> Ganzzahlige Konstante oder Variable, auf der Grünanteil der Farbe als Wert zwischen 0 und 255 übergeben wird.</p> <p><b>Blue</b> Ganzzahlige Konstante oder Variable, auf der Blauanteil der Farbe als Wert zwischen 0 und 255 übergeben wird.</p>
<p><b>Beispiel</b> Nach den Befehlen:</p> <pre style="color: green;">int Handle; Handle = FindObject ("", "Eingabe", 0); SetColorProperty (Handle, "ColorBgr", 255,255,0);</pre> <p>wird die Hintergrundfarbe des ersten gefundenen Objekts mit dem Namen <i>Eingabe</i> auf <i>gelb</i> gesetzt.</p>	



<b>SetComboList</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt eine neue Liste für ein Auswahlobjekt.</p> <p><b>Funktionswert</b>  <b>0</b> Liste wurde korrekt gesetzt  <b>-1</b> Fehler oder Objekt unbekannt</p> <p><b>Bemerkungen</b> Die Einträge der Liste sind durch LF (hex 0A) voneinander zu trennen.  Zu jedem Eintrag kann wahlweise nur der Anzeigename oder auch der Wert und die Farbcodierung angegeben werden.  Wenn Wert und Farbcodierung angegeben werden, sind diese jeweils durch einen TAB (hex 09) vom Anzeigenamen zu trennen.</p> <p><b>Verwandte Funktionen</b>  <i>SetComboIndex</i>  <i>GetComboListByFile</i></p>	<p><b>Handle</b> Ganzzahlige Konstante oder Variable, die den Handle des Auswahlobjekts angibt.</p> <p><b>List</b> Textkonstante oder -variable, die die neue Liste repräsentiert.</p>

<p><b>Beispiel</b> Nach Ausführung der Befehle:</p> <pre style="color: green;">int hobj; hobj = FindObject ("", "combo", 0); SetComboList (hobj, "Bochum\tBO\nMünchen\tM");</pre> <p>enthält die Liste des Auswahlobjekts die Anzeigewerte <i>Bochum</i> und <i>München</i> mit Werten, die ihren Autokennzeichen entsprechen, also <i>BO</i> und <i>M</i>.</p>
---

<b>SetComboListByFile</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt eine neue Auswahlliste für ein Auswahlfeld.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Verwandte Funktionen</b> <i>SetComboList</i></p>	<p><b>Handle</b> Ganzzahlige Konstante oder Variable, die den Handle des Auswahlfeldes angibt, dessen Auswahlliste gesetzt werden soll.</p> <p>Der Handle eines Objekts kann mittels verschiedener Funktionen ermittelt werden, z.B. durch die Funktion <i>FindObject</i>.</p> <p><b>List</b> Textkonstante oder -variable, die den Namen einer Datei angibt, in der zeilenweise alle Texte der Auswahlliste abgelegt sind.</p>
<p><b>Beispiel</b> Nach den Befehlen:</p> <pre style="color: green;">int Handle; Handle = FindObject ("", "Auswahl", 0); SetComboListByFile (Handle, "Liste.txt");</pre> <p>ist die Auswahlliste des Auswahlfeldes mit dem Namen <i>Auswahl</i> mit allen Zeilen der Datei <i>Liste.txt</i> gefüllt.</p>	

<b>SetDefaults</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion überträgt die konfigurierte Vorbelegung in ein bestimmtes oder in alle Eingabebjekte des Formulars</p> <p><b>Funktionswert</b>  <b>0</b> Vorbelegung fehlerfrei gesetzt  <b>-1</b> Es ist ein Fehler aufgetreten.</p>	<p><b>ObjectName</b> Textkonstante oder -variable, die den Namen des Objekts angibt, welches vorbelegt werden soll.</p> <p>Wenn ein leerer Text angegeben wird, werden alle Objekte des Formulars vorbelegt.</p>

<b>SetDisplayFrame</b>	
<p><b>Funktionsbeschreibung</b> Mit dieser Funktion können die Blattränder zwischen dem Fenster und dem Blatt eingestellt werden.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Wenn die Blattränder auf Null gesetzt werden, geht der optische Eindruck eines Blattes innerhalb des Dokumentenfensters verloren. Dies kann bei Anwendungen erwünscht sein, die nicht als Formular, sondern als Dateneingabemaske fungieren.</p>	<p><b>Frame</b> Ganzzahlige Konstante oder Variable, die angibt welcher oder welche der vier Ränder verändert werden sollen:</p> <ul style="list-style-type: none"><li><b>1</b> linker Rand</li><li><b>2</b> oberer Rand</li><li><b>4</b> rechter Rand</li><li><b>8</b> unterer Rand</li></ul> <p>Durch Addieren dieser Werte können auch mehrere Ränder gleichzeitig gesetzt werden.</p> <p><b>Size</b> Ganzzahlige Konstante oder Variable, die den neuen Rand in Pixeln angibt.</p>

<b>SetDisplayState</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt den Status <i>sichtbar</i> für ein oder mehrere Objekte.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Es wird der Status aller Objekte geändert, die zu dem angegebenen Objektnamen passen und die auf einer Seite mit dem angegebenen Seitennamen liegen.</p> <p><b>Verwandte Funktionen</b> <i>GetDisplayState</i></p>	<p><b>PageName</b> Textkonstante oder -variable, die den Namen der Seiten angibt, auf denen Objekte geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B). Ein leerer Text bezeichnet die aktuelle Seite.</p> <p><b>ObjectName</b> Textkonstante oder -variable, die den Namen der Objekte angibt, die geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>State</b> Ganzzahlige Konstante oder Variable, die den neuen Status angibt:</p> <p style="text-align: right;"><b>1</b> sichtbar machen <b>0</b> unsichtbar machen</p>

<b>SetFloat</b>	
<p><b>Funktionsbeschreibung</b>                      Diese Funktion wandelt eine Gleitkommazahl in eine formatierte Dezimaldarstellung und schreibt diese in ein Objekt oder eine Gruppe von Objekten.</p> <p><b>Funktionswert</b>                      Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b>                      Es werden alle Objekte des Formulars geändert, die zu dem angegebenen Objektnamen passen.                       Wenn dies nur auf bestimmten Seiten geschehen soll, so kann die Funktion <i>SetFloatOnPage</i> verwendet werden.</p> <p><b>Verwandte Funktionen</b>  <i>SetFloatOnPage</i>  <i>SetText</i></p>	<p><b>ObjectName</b>                      Textkonstante oder -variable, die den Namen der Objekte angibt, die geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>Value</b>                      Gleitkommakonstante oder -variable, die die darzustellende Zahl angibt.</p> <p><b>DecimalDigits</b>                      Ganzzahlige Konstante oder Variable, die die Anzahl der darzustellenden Nachkommastellen festlegt. 0 bedeutet, dass keine Nachkommastellen erzeugt werden. In diesem Fall wird auch kein Dezimalkomma erzeugt.</p> <p>Die Darstellung wird auf die gewünschte Anzahl Stellen gerundet. Als Dezimaltrennzeichen ist das Komma voreingestellt.</p> <p><b>ThousandSep</b>                      Zeichen, das als Tausendertrenner eingeblendet werden soll. Das Zeichen kann entweder als Zeichenkonstante (z.B. '.') oder als Zeichenwert (z.B. 46 anstelle von '.') übergeben werden. Der Wert 0 bedeutet, dass kein Tausendertrenner erzeugt wird.</p>

<b>SetFloatFormatted</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion formatiert eine Gleitkommazahl und setzt diese als Text in ein oder mehrere Objekte.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Es wird der Inhalt aller Objekte geändert, die zu dem angegebenen Objektnamen passen und die auf einer Seite mit dem angegebenen Seitennamen liegen.</p> <p><b>Verwandte Funktionen</b> <i>SetLongFormatted</i> <i>SetText</i> <i>SetTextOnPage</i> <i>GetFloatFormatted</i></p>	<p><b>PageName</b> Textkonstante oder -variable, die den Namen der Seiten angibt, auf denen Objekte geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B). Ein leerer Text bezeichnet die aktuelle Seite.</p> <p><b>ObjectName</b> Textkonstante oder -variable, die den Namen der Objekte angibt, die geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>Value</b> Gleitkommakonstante oder -Variable, die den neuen Inhalt definiert.</p> <p><b>Format</b> Textkonstante oder -variable, die den Formatdeskriptor für das Format der Zahl definiert.</p> <p>Ein leerer Text bedeutet, dass die Zahl im Standardformat formatiert wird.</p> <p>Formatdeskriptoren sind in Anhang G beschrieben.</p>
<p><b>Beispiel</b></p> <p>Der folgende Befehl setzt den Wert 55,00 in alle Felder aller Seiten, die mit „A“ beginnen:</p> <pre style="color: green;">SetFloatFormatted ("*", "A*", 55.0, "");</pre>	

<b>SetFloatOnPage</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion wandelt eine Gleitkommazahl in eine formatierte Dezimaldarstellung und schreibt diese in ein Objekt oder eine Gruppe von Objekten.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Es werden alle Objekte geändert, die zu dem angegebenen Objektnamen passen und die auf einer Seite mit dem angegebenen Seitennamen liegen.</p> <p><b>Verwandte Funktionen</b> <i>SetFloat</i> <i>SetText</i></p>	<p><b>PageName</b> Textkonstante oder -variable, die den Namen der Seiten angibt, auf denen Objekte geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B). Ein leerer Textbezeichnet die aktuelle Seite.</p> <p><b>ObjectName</b> Textkonstante oder -variable, die den Namen der Objekte angibt, die geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>Value</b> Gleitkommakonstante oder -variable, die die darzustellende Zahl angibt.</p> <p><b>DecimalDigits</b> Ganzzahlige Konstante oder Variable, die die Anzahl der darzustellenden Nachkommastellen festlegt. 0 bedeutet, dass keine Nachkommastellen erzeugt werden. In diesem Fall wird auch kein Dezimalkomma erzeugt.</p> <p>Die Darstellung wird auf die gewünschte Anzahl Stellen gerundet. Als Dezimaltrennzeichen ist das Komma voreingestellt.</p> <p><b>ThousandSep</b> Zeichen, das als Tausendertrenner eingebildet werden soll. Das Zeichen kann entweder als Zeichenkonstante (z.B. '.') oder als Zeichenwert (z.B. 46 anstelle von '.') übergeben werden. Der Wert 0 bedeutet, dass kein Tausendertrenner erzeugt wird.</p>

<b>SetFontColor</b>	
<p><b>Funktionsbeschreibung</b> Dieser Befehl setzt eine neue Schriftfarbe für ein Objekt oder eine Gruppe von Objekten.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Es wird die Schriftfarbe aller Objekte geändert, die zu dem angegebenen Objektnamen passen und die auf einer Seite mit dem angegebenen Seitennamen liegen.</p> <p><b>Verwandte Funktionen</b> <i>SetBgrColor</i></p>	<p><b>PageName</b> Textkonstante oder -variable, die den Namen der Seiten angibt, auf denen Objekte geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B). Ein leerer Text bezeichnet die aktuelle Seite.</p> <p><b>ObjectName</b> Textkonstante oder -variable, die den Namen der Objekte angibt, die geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>Type</b> Ganzzahlige Konstante oder Variable. Ist für spätere Anwendungen reserviert. Muss 0 sein.</p> <p><b>Red</b> Ganzzahlige Konstante oder Variable, die den RGB-Wert für die Farbe Rot zwischen 0 und 255 angibt.</p> <p><b>Green</b> Ganzzahlige Konstante oder Variable, die den RGB-Wert für die Farbe Grün zwischen 0 und 255 angibt.</p> <p><b>Blue</b> Ganzzahlige Konstante oder Variable, die den RGB-Wert für die Farbe Blau zwischen 0 und 255 angibt.</p>



<b>SetFontHeight</b>	
<p><b>Funktionsbeschreibung</b> Dieser Befehl setzt eine neue Schrifthöhe für ein Objekt oder eine Gruppe von Objekten.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Es wird die Schrifthöhe aller Objekte geändert, die zu dem angegebenen Objektnamen passen und die auf einer Seite mit dem angegebenen Seitennamen liegen.</p> <p><b>Verwandte Funktionen</b> <i>SetFontName</i> <i>SetFontColor</i></p>	<p><b>PageName</b> Textkonstante oder -variable, die den Namen der Seiten angibt, auf denen Objekte geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B). Ein leerer Text bezeichnet die aktuelle Seite.</p> <p><b>ObjectName</b> Textkonstante oder -variable, die den Namen der Objekte angibt, die geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>Height</b> Ganzzahlige Konstante oder Variable, die die neue Schrifthöhe in Zehntelpunkten angibt.</p>

<b>SetFontName</b>	
<p><b>Funktionsbeschreibung</b> Dieser Befehl setzt eine neue Schriftart für ein Objekt oder eine Gruppe von Objekten.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Es wird die Schriftart aller Objekte geändert, die zu dem angegebenen Objektnamen passen und die auf einer Seite mit dem angegebenen Seitennamen liegen.</p> <p><b>Verwandte Funktionen</b> <i>SetFontHeight</i> <i>SetFontColor</i></p>	<p><b>PageName</b> Textkonstante oder -variable, die den Namen der Seiten angibt, auf denen Objekte geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B). Ein leerer Text bezeichnet die aktuelle Seite.</p> <p><b>ObjectName</b> Textkonstante oder -variable, die den Namen der Objekte angibt, die geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>FontName</b> Textkonstante oder -variable, die den Namen der Schriftart angibt, die die betroffenen Objekte erhalten sollen.</p>

<b>SetFontProperty</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt eine von mehreren Font-Eigenschaften als String.</p> <p><b>Funktionswert</b> Kein Funktionswert</p> <p><b>Bemerkungen</b> Wenn versucht, wird, eine nicht definierte Eigenschaft zu setzen, ist die Funktion ohne Wirkung.</p> <p><b>Verwandte Funktionen</b> <i>GetFontProperty</i> <i>SetColorProperty</i> <i>SetStringProperty</i> <i>SetLongProperty</i></p>	<p><b>Handle</b> Ganzzahlige Konstante oder Variable, die den Handle des Objekts angibt, das angesprochen werden soll.</p> <p><b>Property</b> Textkonstante oder -variable, die den Namen der Eigenschaft angibt, die gesetzt werden soll.</p> <p>Folgende Eigenschaften sind definiert:</p> <ul style="list-style-type: none"> <li><b>Color</b></li> <li><b>Escapement</b></li> <li><b>Face</b></li> <li><b>Height</b></li> <li><b>Italic</b></li> <li><b>Strikeout</b></li> <li><b>Underline</b></li> <li><b>Weight</b></li> </ul> <p><b>Data</b> Textkonstante oder -variable, auf der der zu setzende Wert übergeben wird.</p>
<p><b>Beispiel</b> Nach Ausführung der Befehle:</p> <pre style="color: green;">int Handle; Handle = FindObject ("", "Eingabe", 0); SetFontProperty (Handle, "Weight", "700");</pre> <p>ist der Schrifttyp des Objekts mit dem Namen <i>Eingabe</i> fett und nicht mehr normal.</p>	

<b>SetHoriAlign</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt die horizontale Ausrichtung von Objekten.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Es wird die Ausrichtung aller Objekte geändert, die zu dem angegebenen Objektnamen passen und die auf einer Seite mit dem angegebenen Seitennamen liegen.</p> <p><b>Verwandte Funktionen</b> <i>SetVertAlign</i> <i>GetHoriAlign</i></p>	<p><b>PageName</b> Textkonstante oder -variable, die den Namen der Seiten angibt, auf denen Objekte geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B). Ein leerer Text bezeichnet die aktuelle Seite.</p> <p><b>ObjectName</b> Textkonstante oder -variable, die den Namen der Objekte angibt, die geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>Mode</b> Ganzzahlige Konstante oder Variable, die die Art der gewünschten Ausrichtung angibt:</p> <ul style="list-style-type: none"><li>1 linksbündig</li><li>2 zentriert</li><li>3 rechtsbündig</li></ul>

<b>SetLongFormatted</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion formatiert einen ganzzahligen Wert und setzt ihn als Text für ein oder mehrere Objekte.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Es wird der Inhalt aller Objekte geändert, die zu dem angegebenen Objektnamen passen und die auf einer Seite mit dem angegebenen Seitennamen liegen.</p> <p><b>Verwandte Funktionen</b> <i>SetFloatFormatted</i> <i>SetText</i> <i>SetTextOnPage</i> <i>GetLongFormattedt</i></p>	<p><b>PageName</b> Textkonstante oder -variable, die den Namen der Seiten angibt, auf denen Objekte geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B). Ein leerer Text bezeichnet die aktuelle Seite.</p> <p><b>ObjectName</b> Textkonstante oder -variable, die den Namen der Objekte angibt, die geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>Value</b> Ganzzahlige Konstante oder -variable, die den neuen Inhalt definiert.</p> <p><b>Format</b> Textkonstante oder -variable, die den Formatdeskriptor für das Format der Zahl definiert.</p> <p>Ein leerer Text bedeutet, dass die Zahl im Standardformat formatiert wird.</p> <p>Formatdeskriptoren sind in Anhang G beschrieben.</p>
<p><b>Beispiel</b></p> <p>Der folgende Befehl setzt den Wert 55 in alle Felder aller Seiten, die mit „A“ beginnen:</p> <pre style="color: green;">SetLongFormatted ("*", "A*", 55, "");</pre>	

<b>SetLongProperty</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt eine von verschiedenen möglichen <i>ganzzahligen Eigenschaften</i> eines Formularobjekts.</p> <p><b>Funktionswert</b>  <b>0</b> Die Eigenschaft wurde gesetzt  <b>-1</b> Fehler oder Objekt unbekannt</p> <p><b>Bemerkungen</b> Nicht alle Eigenschaften haben für jeden Objekttyp eine Bedeutung. Manche Eigenschaften haben nur für bestimmte Objekttypen einen Sinn.  Wenn Eigenschaften benannt werden, die es zwar prinzipiell gibt, aber die für den angegebenen Objekttyp keine Bedeutung haben, erfolgt keine Fehlermeldung, aber der Aufruf der Funktion ist ohne Wirkung.  Wenn Eigenschaften benannt werden, die nicht definiert sind, erfolgt eine Fehlermeldung.</p> <p><b>Verwandte Funktionen</b>  <i>GetLongProperty</i>  <i>SetStringProperty</i>  <i>SetColorProperty</i></p>	<p><b>Handle</b> Ganzzahlige Konstante oder Variable, die den Handle des Objekts angibt, das angesprochen werden soll.</p> <p><b>Property</b> Textkonstante oder -variable, die den Namen der Eigenschaft angibt, die gelesen werden soll.</p> <p>Folgende Eigenschaften sind definiert:</p> <p><b>TempDispTime</b>  <b>DataType</b>  <b>SelectLimit</b>  <b>HoriAlign</b>  <b>VertAlign</b></p> <p>Nicht alle Eigenschaften sind für jedes Objekte sinnvoll.</p> <p><b>NewValue</b> Ganzzahlige Konstante oder Variable, die den neuen Wert der Eigenschaft angibt.</p>
<p><b>Beispiel</b> Nach den Befehlen:</p> <pre style="color: green;">int Time; int Hnd; Hnd = FindObject ("", "Zugriff", 0); Time = GetLongProperty (Hnd, "TempDispTime", 10);</pre> <p>wird die Zeit für die Anzeige der Erfolgsmeldung nach einer Operation für das Datenzugriffsobjekt <i>Zugriff</i> auf 10 Zehntelsekunden - also auf 1 Sekunde - gesetzt.</p>	

<b>SetMacroCode</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt den Makrocode eines Objekts oder den Makrocodes eines globalen Formularmakros.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkung</b> Wenn das angegebene Makroereignis nicht existiert, ist diese Funktion ohne Bedeutung. Das Ereignis wird nicht automatisch angelegt.</p> <p>Den Handle eines Objekts erhält man durch verschiedene andere Makrofunktionen, z.B. durch <i>FindObject</i></p> <p><b>Verwandte Funktionen</b> <i>GetMacroCode</i> <i>FindObject</i></p>	<p><b>ObjektName</b> Ganzzahlige Konstante oder Variable, den Handle des Eingabeobjektes angibt, dessen Code gesetzt werden soll.</p> <p>Wenn der Wert 0 angegeben wird, wird ein globales Formularmakro gesetzt.</p> <p><b>MacroName</b> Textkonstante oder -variable, die den Namen des Makroereignisses angibt, dessen Code gesetzt werden soll.</p> <p><b>Code</b> Textkonstante oder -variable, die den neuen Code bezeichnet.</p>

<b>SetModulePath</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt das Verzeichnis, in dem beim Einfügen von Bausteinen die Bausteindateien erwartet werden.</p> <p><b>Funktionswert</b> Der Funktionswert hat keinen Funktionswert.</p>	<p><b>Path</b> Textkonstante oder -variable, die das neue Verzeichnis für die Bausteinverwaltung angibt.</p>

<b>SetObjectPos</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt eine neue Position für ein beliebiges Objekt innerhalb der Seite auf der es sich befindet.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Verwandte Funktionen</b> <i>GetObjectPos</i></p>	<p><b>Handle</b> Ganzzahlige Konstante oder Variable, die den Handle des Eingabeobjektes angibt, dessen Position verändert werden soll.</p> <p>Der Handle eines Objekts kann mittels verschiedener Funktionen ermittelt werden, z.B. durch die Funktion <i>FindObject</i>.</p> <p><b>LeftPos</b> Ganzzahlige Konstante oder Variable, die die neue linke Position des Objekts in tausendstel Millimetern bezogen auf den linken Blattrand angibt.</p> <p><b>TopPos</b> Ganzzahlige Konstante oder Variable, die die neue obere Position des Objekts in tausendstel Millimetern bezogen auf den oberen Blattrand angibt.</p>

<b>SetPageBgrMode</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt den Seitenhintergrundmodus einer Seite.</p> <p><b>Funktionswert</b> 0 OK -1 Fehler</p> <p><b>Verwandte Funktionen</b> <i>GetPageBgrMode</i> <i>SetPageBgrColor</i> <i>SetPageHeight</i></p>	<p><b>Name</b> Textkonstante oder –Variable, die den Namen der Seite angibt.</p> <p><b>Mode</b> Ganzzahlige Konstante oder Variable, die den neuen Modus angibt:</p> <p>1 = Kein Hintergrund 2 = Farbe 3 = Bitmap</p>



<b>SetPageBgrColor</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt die Farbe des Seitenhintergrundes einer Seite.</p> <p><b>Bemerkung</b> Die Farbe ist nur dann wirksam, wenn der Modus für den Seitenhintergrund den Wert 2 hat.</p> <p><b>Funktionswert</b> 0 OK -1 Fehler</p> <p><b>Verwandte Funktionen</b> <i>GetPageBgrColor</i> <i>SetPageBgrMode</i> <i>SetPageHeight</i></p>	<p><b>Name</b> Textkonstante oder –Variable, die den Namen der Seite angibt.</p> <p><b>Red</b> Ganzzahlige Konstante oder Variable, die den Farbwert für <i>Rot</i> definiert</p> <p><b>Green</b> Ganzzahlige Konstante oder Variable, die den Farbwert für <i>Grünrot</i> definiert.</p> <p><b>Blue</b> Ganzzahlige Konstante oder Variable, die den Farbwert für <i>Blau</i> definiert.</p>

<b>SetPagekeyMode</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion steuert die Bedeutung der <i>Blätter-Tasten BildOben</i> und <i>BildUnten</i>. Im Normalfall blättern diese Tasten das Dokument auf oder ab. In Verbindung mit Datenbankanwendungen ist es manchmal sinnvoll, diese Tasten zum Blättern von einem Datensatz zum nächsten zu Verwenden.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p>	<p><b>Mode</b> Ganzzahlige Konstante oder Variable, die angibt, wie die Blätter-Tasten wirken sollen:</p> <p><b>1</b> Blättern im Dokument <b>2</b> Datensatznavigation</p> <p>Der Wert 1 ist voreingestellt.</p>

<b>SetPictureFile</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt den Namen und Pfad der Bilddatei, die mit dem Objekt assoziiert ist.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkung</b> Bei Bildobjekten, deren Quelle eine referenzierte externe Datei ist, wird mit dieser Funktion der Dateiname gesetzt.  Die Funktion sorgt dafür, dass das neue Bild auch angezeigt wird.</p> <p><b>Verwandte Funktionen</b> <i>LoadBitmap</i></p>	<p><b>PageName</b> Textkonstante oder -variable, die den Namen der Seiten angibt, auf denen Objekte geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B). Ein leerer Text bezeichnet die aktuelle Seite.</p> <p><b>ObjectName</b> Textkonstante oder -variable, die den Namen der Objekte angibt, die geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>Dateiname</b> Textkonstante oder -variable, die den Namen und Pfad der Datei bezeichnet.</p>

<b>SetPrintState</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt den Status <i>wird gedruckt</i> für ein oder mehrere Objekte.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkung</b> Es wird das Attribut <i>wird gedruckt</i> aller Objekte des Formulars geändert, die zu dem angegebenen Objektnamen passen.  Wenn dies nur auf bestimmten Seiten geschehen soll, so kann die Funktion <i>SetPrintStateOnPage</i> verwendet werden.</p> <p><b>Verwandte Funktionen</b> <i>SetPrintStateOnPage</i></p>	<p><b>ObjectName</b> Textkonstante oder -variable, die den Namen der Objekte angibt, die geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>State</b> Ganzzahlige Konstante oder Variable, die den neuen Status angibt:</p> <p style="margin-left: 20px;"><b>1</b> Objekte drucken <b>0</b> Objekte nicht drucken</p>

<b>SetPrintStateOnPage</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt den Status <i>wird gedruckt</i> für ein oder mehrere Objekte.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Es wird der Status aller Objekte geändert, die zu dem angegebenen Objektnamen passen und die auf einer Seite mit dem angegebenen Seitennamen liegen.</p> <p><b>Verwandte Funktionen</b> <i>SetPrintState</i></p>	<p><b>PageName</b> Textkonstante oder -variable, die den Namen der Seiten angibt, auf denen Objekte geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B). Ein leerer Text bezeichnet die aktuelle Seite.</p> <p><b>ObjectName</b> Textkonstante oder -variable, die den Namen der Objekte angibt, die geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>State</b> Ganzzahlige Konstante oder Variable, die den neuen Status angibt:</p> <ul style="list-style-type: none"><li><b>1</b> Objekte drucken</li><li><b>0</b> Objekte nicht drucken</li></ul>

<b>SetReadOnlyState</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt den Status <i>nur lesen</i> für ein oder mehrere Objekte.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkung</b> Es wird der Status <i>nur lesen</i> aller Objekte des Formulars geändert, die zu dem angegebenen Objektnamen passen.</p> <p>Wenn dies nur auf bestimmten Seiten geschehen soll, so kann die Funktion <i>SetReadOnlyStateOnPage</i> verwendet werden.</p> <p><b>Verwandte Funktionen</b> <i>SetReadOnlyStateOnPage</i></p>	<p><b>ObjectName</b> Textkonstante oder -variable, die den Namen der Objekte angibt, die geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>State</b> Ganzzahlige Konstante oder Variable, die den neuen Status angibt:</p> <p><b>1</b> nur lesen <b>0</b> lesen und schreiben</p>

<b>SetReadOnlyStateOnPage</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt den Status <i>nur lesen</i> für ein oder mehrere Objekte.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Es wird der Status aller Objekte geändert, die zu dem angegebenen Objektnamen passen und die auf einer Seite mit dem angegebenen Seitennamen liegen.</p> <p><b>Verwandte Funktionen</b> <i>SetReadOnlyState</i></p>	<p><b>PageName</b> Textkonstante oder -variable, die den Namen der Seiten angibt, auf denen Objekte geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B). Ein leerer Text bezeichnet die aktuelle Seite.</p> <p><b>ObjectName</b> Textkonstante oder -variable, die den Namen der Objekte angibt, die geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>State</b> Ganzzahlige Konstante oder Variable, die den neuen Status angibt:</p> <ul style="list-style-type: none"><li><b>1</b> nur lesen</li><li><b>0</b> lesen und schreiben</li></ul>

<b>SetRuntimeChangeFlag</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt die interne Merkvariable einer Seite, die anzeigt, ob eine bestimmte Seite zur Laufzeit verändert wurde.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Wenn als Seitennummer der Wert -1 angegeben wird, wird die Merkvariable aller Seiten des Formulars gesetzt.</p> <p><b>Verwandte Funktionen</b> <i>GetRuntimeChangeFlag</i> <i>CountRuntimeChangedPages</i> <i>GetPageNrByPageName</i></p>	<p><b>pageNr</b> Ganzzahlige Konstante oder Variable, die die Seitennummer der Seite bezeichnet, deren Merkvariable gesetzt werden soll.</p> <p>Die Seitenzählung beginnt bei 1.</p> <p><b>pageNr</b> Ganzzahlige Konstante oder Variable, die den neuen Wert bezeichnet:</p> <p><b>0:</b> Seite wurde nicht verändert <b>1:</b> Seite wurde verändert</p>

<b>SetScrollOffset</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt eine neue Rollposition für das augenblicklich aktive Formularfenster.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Die Werte sind in Pixeln anzugeben.</p> <p><b>Verwandte Funktionen</b> <i>GetScrollOffset</i></p>	<p><b>Dimension</b> Ganzzahlige Konstante oder Variable, die angibt, welche der Rollrichtungen verändert werden soll:</p> <p>VERT vertikaler Offset HORI horizontaler Offset</p> <p>Es können auch beide Werte gleichzeitig gesetzt werden. In diesem Fall muss VERT+HORI angegeben werden.</p> <p><b>Value</b> Ganzzahlige Konstante oder Variable, die den zu setzenden Offset in Pixeln angibt.</p> <p><b>Flags</b> Ganzzahlige Konstante oder Variable. Hier kann ein oder mehrere der folgenden Schalter angegeben werden:</p> <p>FORMFLAG_NO_REFRESH Keinen Refresh nach der Operation durchführen.</p> <p>0 bedeutet, dass keine Schalter gesetzt sind.</p>

<b>SetStringProperty</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt eine von verschiedenen Möglichen <i>Text-Eigenschaften</i> eines Formularobjekts.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Nicht alle Eigenschaften haben für jeden Objekttyp eine Bedeutung. Manche Eigenschaften haben nur für bestimmte Objekttypen einen Sinn.</p> <p>Wenn Eigenschaften benannt werden, die es zwar prinzipiell gibt, aber die für den angegebenen Objekttyp keine Bedeutung haben, erfolgt keine Fehlermeldung, aber der Aufruf der Funktion ist ohne Wirkung.</p> <p>Wenn Eigenschaften benannt werden, die nicht definiert sind, erfolgt eine Fehlermeldung.</p> <p><b>Verwandte Funktionen</b> <i>GetStringProperty</i> <i>SetLongProperty</i> <i>SetColorProperty</i> <i>SetXmlProperty</i></p>	<p><b>Handle</b> Ganzzahlige Konstante oder Variable, die den Handle des Objekts angibt, das angesprochen werden soll.</p> <p><b>Property</b> Textkonstante oder -variable, die den Namen der Eigenschaft angibt, die gesetzt werden soll.</p> <p>Folgende Eigenschaften sind definiert:</p> <p><b>Name</b> <b>AccessObjectName</b> <b>DataFieldName</b> <b>TypeName</b></p> <p><b>Value</b> Textkonstante oder -variable, die die neue Eigenschaft als Text beschreibt.</p>
<p><b>Beispiel</b> Nach den Befehlen:</p> <pre style="color: green;">int Handle; Handle = FindObject ("Seite1", "Eingabe", 0); SetStringProperty (Handle, "Name", "Otto");</pre> <p>heißt das erste Objekt auf Seite mit dem Namen <i>Seite1</i>, das den Namen <i>Eingabe trägt</i>, nicht mehr <i>Eingabe</i> sondern <i>Otto</i>.</p>	



<b>SetText</b>							
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt einen neuen Text für ein oder mehrere Objekte.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkung</b> Es wird der Text aller Objekte des Formulars geändert, die zu dem angegebenen Objektamen passen.</p> <p>Wenn dies nur auf bestimmten Seiten geschehen soll, so kann die Funktion <i>SetTextOnPage</i> verwendet werden.</p> <p>Der Text eines Objekts hängt von seinem Typ ab. Hier einige Beispiele:</p> <table style="margin-left: 20px;"> <tr> <td><i>Eingabefeld</i></td> <td>augenblickliche Eingabe</td> </tr> <tr> <td><i>Text</i></td> <td>Ausgabertext</td> </tr> <tr> <td><i>Ankreuzfeld</i></td> <td>dem Zustand zugeordneter Text</td> </tr> </table> <p>Diese Funktion kann auch verwendet werden, um ein Objekt zu leeren. In diesem Fall wird einfach ein leerer Text gesetzt.</p> <p><b>Verwandte Funktionen</b> <i>SetTextOnPage</i> <i>GetText</i></p>	<i>Eingabefeld</i>	augenblickliche Eingabe	<i>Text</i>	Ausgabertext	<i>Ankreuzfeld</i>	dem Zustand zugeordneter Text	<p><b>ObjectName</b> Textkonstante oder -variable, die den Namen der Objekte angibt, die geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>Text</b> Textkonstante oder -variable, die den neuen Text angibt.</p>
<i>Eingabefeld</i>	augenblickliche Eingabe						
<i>Text</i>	Ausgabertext						
<i>Ankreuzfeld</i>	dem Zustand zugeordneter Text						

<b>SetTextOnPage</b>							
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt einen neuen Text für ein oder mehrere Objekte.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Es wird der Text aller Objekte geändert, die zu dem angegebenen Objektnamen passen und die auf einer Seite mit dem angegebenen Seitennamen liegen.</p> <p>Der Text eines Objekts hängt von seinem Typ ab. Hier einige Beispiele:</p> <table style="margin-left: 20px;"> <tr> <td><i>Eingabefeld</i></td> <td>augenblickliche Eingabe</td> </tr> <tr> <td><i>Text</i></td> <td>Ausgabertext</td> </tr> <tr> <td><i>Ankreuzfeld</i></td> <td>dem Zustand zugeordneter Text</td> </tr> </table> <p>Diese Funktion kann auch verwendet werden, um ein Objekt zu leeren. In diesem Fall wird einfach ein Leertext "" gesetzt.</p> <p><b>Verwandte Funktionen</b> <i>SetText</i> <i>GetText</i></p>	<i>Eingabefeld</i>	augenblickliche Eingabe	<i>Text</i>	Ausgabertext	<i>Ankreuzfeld</i>	dem Zustand zugeordneter Text	<p><b>PageName</b> Textkonstante oder -variable, die den Namen der Seiten angibt, auf denen Objekte geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B). Ein leerer Text bezeichnet die aktuelle Seite.</p> <p><b>ObjectName</b> Textkonstante oder -variable, die den Namen der Objekte angibt, die geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>Text</b> Textkonstante oder -variable, die den neuen Text angibt.</p>
<i>Eingabefeld</i>	augenblickliche Eingabe						
<i>Text</i>	Ausgabertext						
<i>Ankreuzfeld</i>	dem Zustand zugeordneter Text						

<b>SetTitleMain</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt einen Titel für das Hauptfenster.</p> <p><b>Bemerkungen</b> Der Makrobefehl kann nur im Ausfüller verwendet werden.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Verwandte Funktionen</b> <i>SetTitleMdi</i></p>	<p><b>Title</b> Textkonstante oder -variable, die den Titel angibt..</p>

<b>SetTitleMdi</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt einen Titel für das aktuelle MDI-Fenster.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Verwandte Funktionen</b> <i>SetTitleMain</i></p>	<p><b>Title</b> Textkonstante oder -variable, die den Titel angibt..</p>

<b>SetVertAlign</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt die vertikale Ausrichtung von Objekten.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Es wird die Ausrichtung aller Objekte geändert, die zu dem angegebenen Objektnamen passen und die auf einer Seite mit dem angegebenen Seitennamen liegen.</p> <p><b>Verwandte Funktionen</b> <i>SetHoriAlign</i> <i>GetVertAlign</i></p>	<p><b>PageName</b> Textkonstante oder -variable, die den Namen der Seiten angibt, auf denen Objekte geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B). Ein leerer Text bezeichnet die aktuelle Seite.</p> <p><b>ObjectName</b> Textkonstante oder -variable, die den Namen der Objekte angibt, die geändert werden sollen. Der Name darf auch eine sogenannte <i>Wildcard</i> sein (Anhang B).</p> <p><b>Mode</b> Ganzzahlige Konstante oder Variable, die die Art der gewünschten Ausrichtung angibt:</p> <ul style="list-style-type: none"><li><b>1</b> am oberen Rand</li><li><b>2</b> zentriert</li><li><b>3</b> am unteren Rand</li></ul>

<b>SetXmlProperty</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt eine beliebige Eigenschaft eines Formularobjekts al XML-Wert.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkungen</b> Der Name der Property entspricht dem Namen des <i>XML-Tags</i> bzw. des <i>XML-Attributs</i> unter dem die gewünschte Eigenschaft in XML-Dateien abgespeichert wird.</p> <p>Nicht alle Eigenschaften haben für jeden Objekttyp eine Bedeutung. Manche Eigenschaften haben nur für bestimmte Objekttypen einen Sinn.</p> <p>Wenn Eigenschaften benannt werden, die nicht definiert sind, erfolgt eine Fehlermeldung.</p> <p><b>Verwandte Funktionen</b> <i>GetXmlProperty</i> <i>SetLongProperty</i> <i>SetColorProperty</i> <i>SetStringProperty</i></p>	<p><b>Handle</b> Ganzzahlige Konstante oder Variable, die den Handle des Objekts angibt, das angesprochen werden soll.</p> <p><b>Property</b> Textkonstante oder -variable, die den Namen der Eigenschaft angibt, die gesetzt werden soll.</p> <p><b>Value</b> Textkonstante oder -variable, die die neue Eigenschaft als Text beschreibt.</p>
<p><b>Beispiel</b> Nach den Befehlen:</p> <pre style="color: green;">int Handle; Handle = FindObject ("Seite1", "Eingabe", 0); SetXmlProperty (Handle, "Name", "Otto");</pre> <p>heißt das erste Objekt auf Seite mit dem Namen <i>Seite1</i>, das den Namen <i>Eingabe</i> trägt, nicht mehr <i>Eingabe</i> sondern <i>Otto</i>.</p>	

<b>SetZoom</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt einen neuen Zoomfaktor für die Darstellung des Formulars. Die Angabe des Zoomfaktors erfolgt in Prozent.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p> <p><b>Bemerkung</b> Die Funktion <i>SetZoom</i> erbringt die gleiche Leistung, wie die Funktion <i>Zoom</i>. Die Funktion <i>Zoom</i> ist aus Gründen der Kompatibilität ebenfalls verfügbar.</p> <p><b>Verwandte Funktionen</b> <i>GetZoom</i></p>	<p><b>Factor</b> Ganzzahlige Konstante oder Variable, die den neuen Zoomfaktor in Prozent angibt.</p>

<b>TableGetColumn</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion ermittelt die Nummer der Spalte einer Tabelle, die augenblicklich aktiv ist.</p> <p><b>Funktionswert</b>  <ul style="list-style-type: none"> <li><b>n</b> Aktive Spalte, Zählung ab 0.</li> <li><b>-1</b> Es ist ein Fehler aufgetreten.</li> </ul> </p> <p><b>Verwandte Funktionen</b> <i>TableSetColumn</i></p>	<p><b>Handle</b> Ganzzahlige Variable, die den Handle der Tabelle angibt, die angesprochen werden soll.</p>

<b>TableSetColumn</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion setzt die Nummer der Spalte einer Tabelle, die künftig aktiv sein soll.</p> <p><b>Bemerkung</b> Die Zeile der Tabelle bleibt unberührt. Sie wird in der Regel durch das Zugriffsobjekt bestimmt, an das die Tabelle gebunden ist.</p> <p><b>Funktionswert</b>  <ul style="list-style-type: none"> <li><b>0</b> Die Spalte wurde verändert.</li> <li><b>-1</b> Es ist ein Fehler aufgetreten.</li> </ul> </p> <p><b>Verwandte Funktionen</b> <i>TableGetColumn</i></p>	<p><b>Handle</b> Ganzzahlige Variable, die den Handle der Tabelle angibt, die angesprochen werden soll.</p> <p><b>Column</b> Ganzzahlige Variable, die die Nummer Spalte (ab 0) angibt, die aktiviert werden soll.</p>

<b>UpdateWindow</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion bewirkt einen Bildschirm-Refresh des Fensters, in dem sich das Formular befindet, in dem diese Funktion aufgerufen wird.</p> <p><b>Funktionswert</b> Der Funktionswert ist ohne Bedeutung.</p>	<p><b>Flags</b> Ganzzahlige Konstante oder Variable. Ist für spätere Zwecke reserviert. Es muss 0 angegeben werden.</p>

<b>VerifyInputs</b>	
<p><b>Funktionsbeschreibung</b> Diese Funktion überprüft, ob die Eingabe eines oder aller Eingabeobjekte des Formulars gültig sind.</p> <p><b>Funktionswert</b> <b>0</b> Alle Eingaben sind in Ordnung. <b>-1</b> Mindestens eine Eingabe ist nicht in Ordnung.</p> <p><b>Bemerkung</b> Wenn eine Eingabe nicht in Ordnung ist, wird der Cursor in das fehlerhafte Eingabeobjekt gestellt.</p>	<p><b>ObjectName</b> Textkonstante oder -variable, welcher den Namen des Eingabeobjektes angibt, dessen Inhalt überprüft werden soll.</p> <p>Wenn hier eine leerer Text angegeben wird, werden alle Eingabeobjekte des Formulars überprüft.</p>



## ANHANG A

### Schalter für Datenzugriffsfunktionen

Dieser Anhang beschreibt alle Schalter, die zur Steuerung der Datenzugriffsfunktionen angegeben werden können. Dies sind im wesentlichen alle Funktionen, deren Name mit *AccObj...* beginnt.

Grundsätzlich kann jeder der im folgenden beschriebenen Schalter als Parameter bei allen Datenzugriffsfunktionen verwendet werden. Nicht alle Schalter sind jedoch für jede Funktion sinnvoll. Schalter, die keinen Sinn ergeben, werden ignoriert.

Wenn bei einer Funktion mehr als ein Schalter gleichzeitig verwendet werden soll, können mehrere Schalter durch Addition verbunden werden, wie folgendes Beispiel zeigt:

```
int Handle;  
Handle = FindObject ("", "ZUGRIFF", 0);  
AccObjDeleteRecord (Handle, ACCFLAG_NO_BEEP + ACCFLAG_NO_TEMPDISP)
```

Folgende Schalter sind definiert:

```
ACCFLAG_NO_BEEP  
ACCFLAG_NO_CONFIRM  
ACCFLAG_NO_MACROCALL  
ACCFLAG_NO_OBJECTLOAD  
ACCFLAG_NO_SELECT  
ACCFLAG_NO_TEMPDISP
```

Die Bedeutung der Schalter wird im folgenden beschrieben:

*ACCFLAG\_NO\_BEEP*

Wenn die Funktion in einer Situation aufgerufen wird, wo deren Aufruf nicht zulässig ist, oder wenn ein Fehler auftritt, der nicht durch eine gesonderte Fehleranzeige behandelt wird, ertönt ein Piepen, welches den Anwender auf die Fehlersituation hinweisen soll. Dieses Piepen wird durch Angabe dieses Schalters in jedem Fall unterbunden.

### ACCFLAG\_NO\_CONFIRM

In der Konfiguration eines Datenzugriffsobjekts kann festgelegt werden, dass bestimmte Operationen durch einen Bestätigungsdialog quittiert werden müssen. Dieser Bestätigungsdialog erscheint auch dann, wenn die entsprechende Funktion als Makrobefehl ausgeführt wird - z.B. *Datensatz löschen*. Die Angabe dieses Schalters unterdrückt die Bestätigung aller Aktionen, die während der Ausführung des Makrobefehls angefragt würden, wenn der Schalter nicht gesetzt wäre.

### ACCFLAG\_NO\_MACROCALL

In der Konfiguration eines Datenzugriffsobjekts kann festgelegt werden, dass nach bestimmten Operationen automatisch ein eingebettetes Makro ausgeführt werden soll. Wenn dieser Schalter angegeben wird, unterbleibt die Ausführung jeglicher eingebetteter Makros.

### ACCFLAG\_NO\_OBJECTLOAD

Nach jeder Funktion, welche den aktuellen Datensatz eines Datenzugriffsobjekts verändert oder die bewirkt, dass ein anderer als der augenblickliche Datensatz zum aktuellen Datensatz wird, sorgt die Formularsoftware dafür, dass alle Eingabefelder, Auswahlfelder und Tabellen, die mit dem Datenzugriffsobjekt verbunden sind, automatisch mit den Daten den neuen aktuellen Datensatzes befüllt werden. Das automatische Übertragen der Feldinhalte kann unterdrückt werden, in dem dieser Schalter angegeben wird.

Eine mögliche Anwendung für diesen Schalter ist Stapellauf, der über alle Sätze einer Datenquelle läuft und gewisse Veränderungen vornimmt. In diesem Fall ist die Anzeige der Daten unnötig, da sie bei jedem Datensatz nur für kurze Zeit sichtbar ist und daher vom Anwender sowieso nicht gelesen werden kann. Durch Angabe des Schalters wird die Anzeige unterdrückt und die Performance der Operation erhöht.

### ACCFLAG\_NO\_SELECT

Bestimme Makrofunktionen ziehen eine automatische Neuselektion der Datenquelle nach sich, falls diese eine ODBC-Datenquelle ist. Insbesondere die Funktionen *AccObjSetWhereClause* und *AccObjSetOrderByClause*, die ja direkt die Selektion an sich verändern, führen nach ihrer Ausführung automatisch eine Neuselektion durch. Diese Neuselektion wird durch Angabe dieses Schalters unterdrückt.

Die Anwendung des Schalters ist z.B. dann sinnvoll, wenn Nacheinander sowohl die Sortierklausel, als auch die Selektionskriterien verändert werden sollen. In diesem Fall würden beide obengenannten Funktionen in Folge aufgerufen. Nach jedem Aufruf würde eine Selektion durchgeführt. Das ist unnötig. Daher kann - zumindest bei dem ersten der beiden Aufrufe - die Selektion unterdrückt werden.

#### ACCFLAG\_NO\_TEMPDISP

In der Konfiguration eines Datenzugriffsobjekts kann festgelegt werden, dass nach bestimmten Operationen eine Quittung auf dem Schirm eingeblendet wird, die nach kurzer Zeit von selbst wieder erlischt. Durch Angabe dieses Schalters kann diese Funktion unterdrückt werden.

## ANHANG B

### Wildcards für Seiten- und Objektnamen

In verschiedenen Funktionen können anstelle von Namen für Seiten oder Objekte auch sogenannte *Wildcards* angegeben werden. *Wildcards* enthalten im Gegensatz zu herkömmlichen Namen *Jokerzeichen*. Im wesentlichen können zwei Jokerzeichen verwendet werden:

- ? Dieses Zeichen ersetzt ein beliebiges einzelnes Zeichen
- \* Dieses Zeichen ersetzt alle Zeichen bis zum Ende des Namens

Folgende Beispiele verdeutlichen die Anwendung:

<b>Wildcard</b>	<b>Namen die beim Vergleichen erkannt werden</b>
AA??BB	AAAABB, AA12BB AAXXBB, etc.
???	alle Namen, die genau drei Zeichen lang sind
AAA*	alle Namen, die mit AAA beginnen, egal, was danach kommt
A?A*	alle Namen, die an erster und dritter Stelle ein 'A' haben
???*	alle Namen, die mindestens drei Zeichen lang sind.

Es ist zu beachten, dass der \* nur am Ende des Namens stehen darf.

## ANHANG C

### Formularmakros

In verschiedenen Funktionen können Makros aufgerufen werden, die für ein Formular definiert sind. Diese werden namentlich angesprochen. Die folgende Liste nennt alle verfügbaren Makronamen:

- OnSessionOpen
- OnSessionClose
- OnSessionSave

- OnAnyObjectGetFocus
- OnAnyObjectKillFocus
- OnAnyObjectChange
- OnAnyButtonClick
- OnAnyObjectClick
- OnAnyObjectDbClick
- OnAnyObjectRightClick

- OnRecordAccess
- OnRecordSelection
- OnRecordUnselect

- OnPrintStart
- OnPrintEnd
- OnPrintCopyStart

## ANHANG D

### Objektmakros

In verschiedenen Funktionen können Makros aufgerufen werden, die für ein bestimmtes Objekt definiert sind. Diese werden namentlich angesprochen. Nicht alle Makros können auf jedes Objekt angewendet werden. Nur sinnvolle Zuordnungen sind möglich.

Die folgende Liste nennt alle Makronamen, die zur Zeit definiert sind:

- OnObjectGetFocus
- OnObjectKillFocus
- OnObjectChange
- OnObjectForward
- OnObjectBackward
- OnObjectClick

Für Datenzugriffsobjekte gibt es darüber hinaus folgende Ereignisse:

- OnRecordAccess
- OnRecordUpdate
- OnRecordDelete
- OnRecordInsert
- OnAccessStatusChange
- OnRecordSelection

## ANHANG E

### Funktionsnamen (Deutsch – Englisch)

In früheren Versionen der Makrosprache wurde eine Mischung aus deutschen und englischen Funktionsnamen verwendet. Aus Gründen der Internationalisierung sind heute alle Funktionsnamen englisch. Selbstverständlich werden alle deutschen Namen aus Gründen der Aufwärtskompatibilität weiter unterstützt.

Da in dieser Dokumentation alle Funktionsnamen in der englischen Form beschrieben sind, können mit Hilfe der folgenden Zuordnungstabelle aus den ehemals deutschen Funktionsnamen die aktuellen englischen Funktionsnamen abgeleitet werden.

Die deutschen Makronamen werden weder von der Online-Hilfe genannt, noch in dieser Dokumentation beschrieben.

Addiere .....	AddObjects
BausteinEinfügen .....	InsertModule
DIIAufruf.....	DIICall
DIIladeAlleObj .....	DIIloadAllObjects
DIIladeObjekt .....	DIIloadObject
DIIspeichereAlleObj .....	DIIsaveAllObjects
DIIspeichereObjekt .....	DIIsaveObject
Drucken.....	Print
EingabenPrüfen .....	VerifyInputs
GibAktivStatus .....	GetActiveState
GibAktiveSeite .....	GetCurrentPage
GibAktivesObjekt .....	GetCurrentObject
GibAnkreuzfeld .....	GetCheckbox
GibAnzeigeStatus.....	GetDisplayState
GibAusfertigungsName.....	GetCopyName
GibCapsStatus.....	GetCapsState
GibDatenFeld .....	GetDataField
GibDruckStatus .....	GetPrintState
GibEingabefeld.....	GetText
GibGleitkommazahl .....	GetObjectAsFloat
GibHoriAusrichtung .....	GetHoriAlign
GibNurLesenStatus .....	GetReadOnlyState
GibObjektTyp .....	GetObjectType

GibSchriftHöhe.....	GetFontHeight
GibSchriftName.....	GetFontName
GibSeite.....	GetPage
GibText.....	GetText
GibTextLänge .....	GetTextLng
GibTitelDatei .....	GetTitleFile
GibVertAusrichtung .....	GetVertAlign
LadeBitmap .....	LoadBitmap
LadenAusProfile .....	LoadFromProfile
NeueVorlage .....	NewForm
RufeBetretenMakro.....	CallEnterMacro
RufeDrückenMakro.....	CallPrintMacro
RufeLoslassenMakro.....	CallReleaseMacro
RufeRückwärtsMakro .....	CallBackwardMacro
RufeVerlassenMakro .....	CallExitMacro
RufeVorwärtsMakro .....	CallForwardMacro
RufeÄnderungsMakro.....	CallChangeMacro
Schließen.....	Close
SchriftenEinstellen .....	EditFonts
SendeAlleFelder .....	BroadcastAllInputs
SetzeAktivStatus .....	SetActiveState
SetzeAnkreuzfeld .....	SetCheckbox
SetzeAnkreuzfeldAufSeite .....	SetCheckboxOnPage
SetzeAnzeigeStatus.....	SetDisplayState
SetzeBausteinPfad .....	SetModulePath
SetzeCapsStatus.....	SetCapsState
SetzeCapsStatusAufSeite .....	SetCapsStateOnPage
SetzeDruckStatus .....	SetPrintState
SetzeDruckStatusAufSeite .....	SetPrintStateOnPage
SetzeEingabefeld .....	SetText
SetzeGeändertStatus .....	SetChangeFlag
SetzeGleitAufSeite .....	SetFloatOnPage
SetzeGleitkommazahl .....	SetFloat
SetzeHoriAusrichtung .....	SetHoriAlign
SetzeNurLesenStatus .....	SetReadOnlyState
SetzeNurLesenStatusAufSeite .....	SetReadOnlyStateOnPage
SetzeSchriftHöhe .....	SetFontHeight
SetzeSchriftName .....	SetFontName
SetzeText.....	SetText
SetzeTextAufSeite.....	SetTextOnPage
SetzeVertAusrichtung .....	SetVertAlign



Speichern .....	Save
SpeichernAls .....	SaveAs
SpeichernAlsErweitert .....	SaveAsExtended
SpeichernInProfile .....	SaveToProfile
SpeichernInTextDatei .....	SaveToTextfile
SprungAufObjekt .....	GotoObjekt
SprungRelativ.....	GotoRelative
Vorbesetzen.....	SetDefaults
Öffnen .....	Open

## ANHANG F

### Funktionsnamen (Englisch – Deutsch)

In früheren Versionen der Makrosprache wurde eine Mischung aus deutschen und englischen Funktionsnamen verwendet. Aus Gründen der Internationalisierung sind heute alle Funktionsnamen englisch. Selbstverständlich werden alle deutschen Namen aus Gründen der Aufwärtskompatibilität weiter unterstützt.

Mit Hilfe der folgenden Zuordnungstabelle können denjenigen englischen Funktionsnamen, die früher eine deutsche Entsprechung hatten, den jeweiligen ehemaligen deutschen Namen zugeordnet werden.

Die deutschen Makronamen werden weder von der Online-Hilfe genannt, noch in dieser Dokumentation beschrieben.

BroadcastAllInputs .....	SendeAlleFelder
CallBackwardMacro .....	RufeRückwärtsMakro
CallChangeMakro .....	RufeÄnderungsMakro
CallEnterMacro .....	RufeBetretenMakro
CallExitMacro .....	RufeVerlassenMakro
CallForwardMakro .....	RufeVorwärtsMakro
AddObjects .....	Addiere
CallPrintMacro .....	RufeDrückenMakro
CallReleaseMacro .....	RufeLoslassenMakro
Close .....	Schließen
DllCall .....	DllAufruf
DllLoadAllObjects .....	DllLadeAlleObj
DllLoadObject .....	DllLadeObjekt
DllSaveAllObjects .....	DllSpeichereAlleObj
DllSaveObject .....	DllSpeichereObjekt
EditFonts .....	SchriftenEinstellen
GetActiveState .....	GibAktivStatus
GetCapsState .....	GibCapsStatus
GetCheckbox .....	GibAnkreuzfeld
GetCopyName .....	GibAusfertigungsName
GetCurrentObject .....	GibAktivesObjekt
GetCurrentPage .....	GibAktiveSeite
GetDataField .....	GibDatenFeld

GetDisplayState .....	GibAnzeigeStatus
GetFontHeight .....	GibSchriftHöhe
GetFontName .....	GibSchriftName
GetHoriAlign .....	GibHoriAusrichtung
GetObjectAsFloat .....	GibGleitkommazahl
GetType .....	GibObjektTyp
GetPage .....	GibSeite
GetPrintState .....	GibDruckStatus
GetReadOnlyState .....	GibNurLesenStatus
GetText .....	GibEingabefeld
GetText .....	GibText
GetTextLng .....	GibTextLänge
GetTitleFile .....	GibTitelDatei
GetVertAlign .....	GibVertAusrichtung
GotoObjekt .....	SprungAufObjekt
GotoRelative .....	SprungRelativ
InsertModule .....	BausteinEinfügen
LoadBitmap .....	LadeBitmap
LoadFromProfile .....	LadenAusProfile
NewForm .....	NeueVorlage
Open .....	Öffnen
Print .....	Drucken
Save .....	Speichern
SaveAs .....	SpeichernAls
SaveAsExtended .....	SpeichernAlsErweitert
SaveToProfile .....	SpeichernInProfile
SaveToTextfile .....	SpeichernInTextDatei
SendAllFields .....	SendeAlleFelder
SetActiveState .....	SetzeAktivStatus
SetCapsState .....	SetzeCapsStatus
SetCapsStateOnPage .....	SetzeCapsStatusAufSeite
SetChangeFlag .....	SetzeGeändertStatus
SetCheckboxOnPage .....	SetzeAnkreuzfeldAufSeite
SetCheckbox .....	SetzeAnkreuzfeld
SetDefaults .....	Vorbesetzen
SetDisplayState .....	SetzeAnzeigeStatus
SetFloat .....	SetzeGleitkommazahl
SetFloatOnPage .....	SetzeGleitAufSeite
SetFontHeight .....	SetzeSchriftHöhe
SetFontName .....	SetzeSchriftName
SetHoriAlign .....	SetzeHoriAusrichtung

SetModulePath.....	SetzeBausteinPfad
SetPrintState.....	SetzeDruckStatus
SetPrintStateOnPage .....	SetzeDruckStatusAufSeite
SetReadOnlyState.....	SetzeNurLesenStatus
SetReadOnlyStateOnPage .....	SetzeNurLesenStatusAufSeite
SetText.....	SetzeEingabefeld
SetText.....	SetzeText
SetTextOnPage .....	SetzeTextAufSeite
SetVertAlign.....	SetzeVertAusrichtung
VerifyInputs .....	EingabenPrüfen

## ANHANG G

### Format-Deskriptoren

Für die automatische Formatierung von Zahlen, Datums- und Uhrzeitangaben muss eine detaillierte Beschreibung des gewünschten Formates definiert werden. Um eine möglichst system- und maschinenunabhängige Darstellung zu gewährleisten, wurde eine Formatbeschreibung in Stringform gewählt.

Insgesamt werden vier verschiedene Formate definiert:

- Format für ganze Zahlen**
- Format für Gleitkommazahlen**
- Format für Datumsangaben**
- Format für Uhrzeiten**

Diese vier Formate werden in den folgenden Kapiteln erläutert. Sie sind inhaltlich verwandt, unterscheiden sich jedoch wesentlich voneinander.

#### Datumsformat

Ein Datumsformat ist ein String, der aus einer Aneinanderreihung von Platzhaltern für die Komponenten *Wochentag*, *Tag*, *Monat* und *Jahr* besteht. Vor, zwischen und hinter den Platzhaltern kann ein beliebiger Text stehen, der Bestandteil des fertig formatierten Datums wird.

Jeder Platzhalter beginnt mit einem Prozentzeichen. Dann folgt ein Buchstabe, der den Platzhaltertyp bestimmt. Es sind folgende Platzhalter definiert, die ein Datumsformat enthalten kann:

- %Wx** Wochentagsname
- %Dx** Tag
- %Mx** Monat
- %Yx** Jahresangabe

Dahinter folgt eine Variantenbezeichnung x. Die Variantenbezeichnung ist eine einstellige Ziffer und bedeutet je nach Platzhalter:

- %W0** Keine Wochentageeinblendung
- %W1** Wochentag als zweibuchstabiges Kürzel einblenden
- %W2** Wochentag als vollen Namen einblenden
  
- %D0** Keine Einblendung des Tages
- %D1** Tag als möglichst kurze Zahl ohne führende Null einblenden
- %D2** Tag als zweistellige Zahl, ggf. mit einer führenden Null einblenden
- %D3** Tag als Zahl mit englischem Ordinalindex (1st, 2nd, 3rd, 4th, etc.)
  
- %M0** Keine Einblendung des Monats
- %M1** Monat als möglichst kurze Zahl ohne führende Null einblenden
- %M2** Monat als zweistellige Zahl, ggf. mit einer führenden Null einblenden
- %M3** Monatsname als dreibuchstabile Abkürzung einblenden
- %M4** Monatsname in voller Länge einblenden
  
- %Y0** Keine Einblendung des Jahres
- %Y1** Jahr als zweistellige Zahl ohne Jahrhundert einblenden
- %Y2** Jahr als vierstellige Zahl mit Jahrhundert einblenden

Durch die Reihenfolge der Platzhalter können auch die englischsprachige Anordnung (Monat, Tag, Jahr) und die sortierfreudige Anordnung (Jahr, Monat, Tag) bestimmt werden. insgesamt werden folgende Anordnungen der Platzhalter unterstützt:

**Tag, Monat, Jahr**  
**Monat, Tag, Jahr**  
**Jahr, Monat, Tag**

Zwischen den Platzhaltern dürfen beliebige Zeichen oder Textfolgen stehen. Die Anordnung der Wochentagsbezeichnung (%W) ist beliebig.

Folgende Beispiele zeigen die Darstellung des Datums 3. Januar 1955 mit verschiedenen Datumsformaten:

<b>Format</b>	<b>Darstellung</b>
%W0%D1.%M1.%Y2	3.1.1955
%W0%D2/%M2/%Y1	03/01/55
%W0%D1-%M3-%Y2	3-Jan-1955
%W0%M3, %D1, %Y2	Jan, 3, 1955
%W2, den %D1. %M4 %Y2	Montag, den 3. Januar 1955
%M4, %D3, %Y2	Januar, 3rd, 1955

### Verweis auf das System-Datumsformat

Anstelle eines expliziten Formats kann der Formatstring auch einen Verweis auf das im System länderspezifisch vorgegebene System-Datumsformat enthalten, welches in den Ländereinstellungen definiert werden kann. Unter Windows steht ein kurzes und ein langes System-Datumsformat zur Verfügung.

In diesem Fall besteht der gesamte Formatstring nur aus einer der beiden Zeichenfolgen:

<b>%SS</b>	Kurzes System-Datumsformat verwenden
<b>%SL</b>	Langes System-Datumsformat verwendet

Sollte der Formatstring weitere Zeichen beinhalten, werden diese ignoriert.

### Uhrzeitformat

Ein Uhrzeitformat ist ein String, der aus einer Aneinanderreihung von Platzhaltern für die Komponenten *Stunden*, *Minuten* und *Sekunden* besteht. Vor, zwischen und hinter den Platzhaltern kann ein beliebiger Text stehen, der Bestandteil der fertig formatierten Uhrzeit wird.

Jeder Platzhalter beginnt mit einem Prozentzeichen. Dann folgt ein Buchstabe, der den Platzhalterttyp bestimmt. Es sind folgende Platzhalter definiert, die ein Uhrzeitformat enthalten kann:

**%Hx**    Stunden (24-Stunden-Anzeige)  
**%hx**    Stunden (12-Stunden-Anzeige)  
**%Ux**    Minuten  
**%Sx**    Sekunden

Dahinter folgt eine Variantenbezeichnung x. Die Variantenbezeichnung ist eine einstellige Ziffer und bedeutet je nach Platzhalter:

**%H0**    Keine Stundenanzeige  
**%H1**    Stunden als möglichst kurze Zahl ohne führende Null einblenden  
**%H2**    Stunden zweistellig, ggf. mit einer führenden Null einblenden

Entsprechendes gilt für die 12-Stunden-Anzeige (**%hx**).

**%U0**    Keine Einblendung der Minuten  
**%U1**    Minuten als möglichst kurze Zahl ohne führende Null einblenden  
**%U2**    Minuten zweistellig, ggf. mit einer führenden Null einblenden

**%S0**    Keine Einblendung der Sekunden  
**%S1**    Sekunden als möglichst kurze Zahl ohne führende Null einblenden  
**%S2**    Sekunden zweistellig, ggf. mit einer führenden Null einblenden

Die Reihenfolge der Platzhalter ist zwar im Prinzip beliebig, jedoch wird in der Praxis nur die gewohnte Reihenfolge (Stunden, Minuten, Sekunden) auftreten.

Zwischen den Platzhaltern dürfen beliebige Zeichen oder Textfolgen stehen.

Folgende Beispiele zeigen die Darstellung der Uhrzeit 13:03:22 mit verschiedenen Zeitformaten:

<b>Format</b>	<b>Darstellung</b>
<code>%H1:%U2%S0</code>	13:03
<code>%H1:%U2:%S2</code>	13:03:22
<code>%H1 Uhr %U2%S0</code>	13 Uhr 03



## Verweis auf das System-Uhrzeitformat

Anstelle eines expliziten Formats kann der Formatstring auch einen Verweis auf das im System länderspezifisch vorgegebene System-Uhrzeitformat enthalten, welches in den Ländereinstellungen definiert werden kann. In diesem Fall besteht der gesamte Formatstring nur aus der Zeichenfolge:

**%SS** System-Uhrzeitformat

Sollte der Formatstring weitere Zeichen beinhalten, werden diese ignoriert.

## Ganzzahlformat

Anders als die Datums- und Zeitformate - aber ähnlich dem Gleitkommaformat - bestehen Ganzzahlformate lediglich aus einer Reihe aneinandergereihter Steueranweisungen. Vor, zwischen und hinter den Steueranweisungen darf kein anderer Text stehen.

Alle Steueranweisungen beginnen mit einem Prozentzeichen. Dahinter folgt ein Buchstabe, der den Typ der Anweisung bestimmt. Folgende Steueranweisungen sind möglich:

**%Wx** Definition der Stellenzahl  
**%Tx** Definition des Tausendertrenners  
**%Bx** Definition eines Prefix-Strings, der vor der Zahl eingeblendet wird  
**%Ex** Definition eines Postfix-Strings, der hinter der Zahl eingeblendet wird  
**%+** Anweisung, dass auch bei positiven Zahlen ein Vorzeichen erscheint  
**%-** Anweisung, dass bei negativen Zahlen das Vorzeichen unterdrückt wird

Je nach Typ der Anweisung folgt hinter dem Kennbuchstaben noch weitere Information.

Die Anweisung **%Wx** bestimmt die Anzahl Stellen, die die ganze Zahl haben soll. Die Stellenzahl folgt dem %W als einstellige Ziffer. Wenn die zu formatierende Zahl weniger Stellen hat, als gewünscht, werden führende Nullen erzeugt.

Die Anweisung **%Tx** bestimmt den Tausendertrenner, mit dem Tausendergruppen voneinander getrennt werden sollen. Die Angabe x kann aus einem oder mehreren Zeichen bestehen. Im Normalfall wird in Deutschland nur ein Zeichen verwendet, nämlich der Punkt. Wenn kein Tausendertrenner erwünscht ist, so wird für x einfach nichts angegeben, sondern direkt mit der nächsten Steueranweisung fortgefahren.

Die Anweisung **%BX** (begin) und **%Ex** (end) definieren Zeichenfolgen x, die bis zum nächsten Prozentzeichen reichen. Diese Zeichenfolgen werden als Prefix bzw. Postfix vor bzw. hinter die formatierte Zahl gekettet. Wenn kein Prefix oder Postfix erwünscht ist, wird für x einfach nichts angegeben und direkt mit der nächsten Steueranweisung fortgefahren.

Die Anweisung **%+** bewirkt, dass positive Zahlen mit einem Plus-Zeichen dargestellt werden. Negative Zahlen werden ohnehin mit einem Minus-Zeichen versehen.

Folgende Beispiele zeigen die Darstellung der Zahl 1234 mit verschiedenen Ganzzahlformaten:

Format	Darstellung
<b>%T.</b>	1.234
<b>%T</b>	1234
<b>%BDM %T. %EZaster</b>	DM 1.234 Zaster
<b>%+%W9%T,%EMio DM</b>	+000,001,234 Mio DM

### Gleitkommazahlformat

Anders als die Datums- und Zeitformate - aber ähnlich dem Ganzzahlformat - bestehen Gleitkommazahlformate lediglich aus einer Reihe aneinandergereihter Steueranweisungen. Vor, zwischen und hinter den Steueranweisungen darf kein anderer Text stehen.

Alle Steueranweisungen beginnen mit einem Prozentzeichen. Dahinter folgt ein Buchstabe, der den Typ der Anweisung bestimmt. Folgende Steueranweisungen sind möglich:

<b>%Fx</b>	Definition der Nachkommastellenzahl
<b>%Tx</b>	Definition des Tausendertrenners
<b>%Dx</b>	Definition des Dezimalkommas
<b>%Bx</b>	Definition eines Prefix-Strings, der vor der Zahl eingeblendet wird
<b>%Ex</b>	Definition eines Postfix-Strings, der hinter der Zahl eingeblendet wird
<b>%+</b>	Anweisung, dass bei positiven Zahlen ein Vorzeichen erscheint
<b>%-</b>	Anweisung, dass bei negativen Zahlen das Vorzeichen unterdrückt wird

Die Anweisung **%Fx** bestimmt die Anzahl der Nachkommastellen, mit der die Zahl ausgegeben werden soll. Die Anzahl Nachkommastellen x wird als ein- oder zweistellige Zahl angegeben.

Die Anweisung **%Tx** bestimmt den Tausendertrenner, mit dem Tausendergruppen voneinander getrennt werden sollen. Die Angabe x kann aus einem oder mehreren Zeichen bestehen. Im Normalfall wird in Deutschland nur ein Zeichen verwendet, nämlich der Punkt. Wenn kein Tausendertrenner erwünscht ist, so wird für x einfach nichts angegeben, sondern direkt mit der nächsten Steueranweisung fortgefahren.

Die Anweisung **%Dx** bestimmt das Dezimalkomma. x besteht genau aus einem Zeichen, welches das Dezimalkomma repräsentiert. Im Normalfall wird in Deutschland das Komma verwendet. Wenn diese Angabe fehlt, wird ein Komma als Voreinstellung benutzt.

Die Anweisung **%Bx** (begin) und **%Ex** (end) definieren Zeichenfolgen x, die bis zum nächsten Prozentzeichen reichen. Diese Zeichenfolgen werden als Prefix bzw. Postfix vor bzw. hinter die formatierte Zahl gekettet. Wenn kein Prefix oder Postfix erwünscht ist, wird für x einfach nichts angegeben und direkt mit der nächsten Steueranweisung fortgefahren.

Die Anweisung **%+** bewirkt, dass positive Zahlen mit einem Plus-Zeichen dargestellt werden. Negative Zahlen werden ohnehin mit einem Minus-Zeichen versehen.

Folgende Beispiele zeigen die Darstellung der Zahl 12345,67 mit verschiedenen Gleitkommazahlformaten:

<b>Format</b>	<b>Darstellung</b>
<code>%T.</code>	12.345,67
<code>%T.%F5</code>	12.345,67000
<code>%BDM %T%FO</code>	DM 12346

## ANHANG H

### Veraltete Funktionen

Im Rahmen der Weiterentwicklung werden manche Funktionen durch bessere ersetzt. Dabei ändern sich gelegentlich auch die Parameter der Funktionen. Aus Kompatibilitätsgründen werden die alten Funktionen auch weiterhin unterstützt. Jedoch werden diese aus der Dokumentation entfernt und durch die Beschreibung der neueren Funktion ersetzt.

Der folgende Anhang zeigt die Namen dieser Funktionen und ihre Entsprechungen:

<b>Alte Funktion.....</b>	<b>Neue Funktion</b>
AppObjectSet .....	AppObjectSetExt
CallBackwardMacro .....	CallObjectMacro
CallChangeMacro .....	CallObjectMacro
CallEnterMacro .....	CallObjectMacro
CallExitMacro .....	CallObjectMacro
CallForwardMacro.....	CallObjectMacro
CallPressMacro.....	CallObjectMacro
CallReleaseMacro.....	CallObjectMacro