

Schulungsunterlagen

Formulareditor

© Copyright 2023 by SelectLine Software AG, CH-9016 St. Gallen

Kein Teil dieses Dokumentes darf ohne ausdrückliche Genehmigung in irgendeiner Form ganz oder in Auszügen reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden. Wir behalten uns vor, ohne besondere Ankündigung, Änderungen am Dokument und am Programm vorzunehmen. Die im Dokument verwendeten Soft- und Hardware-Bezeichnungen sind überwiegend eingetragene Warenbezeichnungen und unterliegen als solche den gesetzlichen Bestimmungen des Urheberrechtsschutzes.

03.11.2023/pe/V3.9

Inhalt

1	Vorwort	3
2	Aufbau des Formulareditors	4
2.1	Verwaltung und Struktur	4
2.2	Besonderheit Auftrag: Druckvorlage kopieren von	7
2.3	Parameter	8
2.4	Drucker	9
2.5	Anpassungen von Formularen noch komfortabler (ab 21.3)	10
2.6	Blöcke	11
2.6.1	Frei positionierbarer Block im Kopf	11
2.6.2	Block mit vordefinierter Abfrage	12
2.6.3	Block mit Blockdefinition (Selbstdefinierte Abfrage)	13
2.6.4	Seitenwechsel vermeiden	14
3	Platzhalter	15
3.1	Art und Aufbau	15
3.2	Sonderfunktionen für Textplatzhalter	17
3.3	Verweise	17
3.4	Formularbaustein	18
3.5	Seitenzahl drucken	18
3.6	Sonderplatzhalter	19
4	Formeln in Formelplatzhaltern	21
4.1	Konstanten	21
4.2	Variablen	22
4.3	Funktionen	22
4.3.1	für numerische Operanden	22
4.3.2	Bit-Operationen	23
4.3.3	für Zeichenketten	23
4.3.4	Vergleichsoperatoren	24
4.3.5	für Logikoperatoren	25
4.3.6	Erkennung und Behandlung von Null-Werten	25
4.3.7	Datumsfunktion	25
4.3.8	Umwandlungsfunktion	26
4.3.9	Datenbankfunktionen	27
4.3.10	Funktionen zum Lesen und Schreiben von Einträgen in INI-Dateien	27
4.3.11	Funktionen zum Erstellen der IBAN-Nummer	28
4.4	Spezielle Funktionen für den Auftrag	29
4.5	Drucken von html-formatierten Text	32
5	Anhang	33
5.1	Glossar	33
5.2	Dank	34
5.3	Ihre Notizen und Erkenntnisse	35

1 Vorwort

Vielen Dank für Ihr Interesse an SelectLine und dem Besuch dieses Kurses „Formulareditor“. Wir freuen uns sehr über Ihre Teilnahme an dieser Schulung und sind überzeugt, dass Ihnen diese Software eine grosse Unterstützung in Ihrer täglichen Arbeit sein wird. Die bedienerfreundliche Benutzeroberfläche wird es Ihnen ermöglichen, dass Sie schnell erste Erfolge erzielen können und Ihnen die Arbeit leicht von der Hand gehen wird. Aber lassen Sie sich nicht täuschen. Auch Sie werden stets wieder neue Funktionalitäten und Möglichkeiten entdecken, welche dieses Programm bietet.



Anschliessend an diesen Kurs sind Sie in der Lage bestehende Systemdruckvorlagen zu bearbeiten und neue Formulare zu kreieren. Weiter können Sie Formelplatzhalter einsetzen und mit Unterblöcken arbeiten.

Um Ihnen das Arbeiten mit diesem Lehrmittel so einfach wie möglich zu machen, verwenden wir in diesem Kurs – und später auch in den weiteren Kursen – Symbole, die Ihnen einen raschen Überblick der wichtigsten Punkte geben sollen. Dies, da auch das Programm über Symbole oder sogenannte „Icons“ gesteuert wird. Hier deshalb kurz ein Überblick über die Symbole, die Ihnen in diesen Unterlagen begegnen werden. Das Erste haben Sie bereits kennengelernt.



Lernziele

Neben diesem Symbol sehen Sie, was das Ziel dieser Einheit ist oder welches Wissen Sie neu erwerben.



Hinweise

Hier erfahren Sie wichtige Tipps, Hinweise und Funktionen des Programms oder Einstellungen, welche Sie vornehmen können.



Übungen

Wenn Sie dieses Icon sehen, sind Sie an der Reihe. Hier geht es darum, das erworbene, theoretisches Wissen in die Praxis umzusetzen anhand von Fallbeispielen.



Infos

Diese Möglichkeit steht Ihnen nur in den Versionen Gold oder Platin zur Verfügung.

Nun wünschen wir Ihnen viel Spass und Erfolg in dieser Schulung und anschliessend beim Erkunden der Software und natürlich auch im täglichen Praxiseinsatz.

Wir weisen darauf hin, dass Sie nur dann optimal vom Kurs profitieren können, wenn folgende Voraussetzungen erfüllt sind:

- Sehr gute Kenntnisse von SelectLine
- Kenntnisse der SelectLine Tabellenstruktur

Beachten Sie auch, dass alle Funktionen dieses Programms im Handbuch „SelectLine System Handbuch“ entsprechend ausführlich detailliert geschildert werden. Die Kursunterlagen dienen lediglich als Ergänzung dazu. Für weitere Informationen empfehlen wir Ihnen daher, ebenfalls das Handbuch zu konsultieren. Zudem können Sie an nahezu jeder Stelle des Programms mit der Taste [F1] die Hilfe aufrufen. So werden Ihnen direkt zum aktuellen Programmpunkt weitere Informationen angezeigt. Eine Übersicht des Funktionsumfangs und der Abgrenzung zwischen den Skalierungen Standard, Gold und Platin entnehmen Sie der Leistungsübersicht, welche auf dem USB-Stick oder auf der Homepage zu finden ist. Weiter empfehlen wir Ihnen auch das **Neuerungsdocument** auf dem USB-Stick oder Homepage zu beachten.

2 Aufbau des Formulareditors

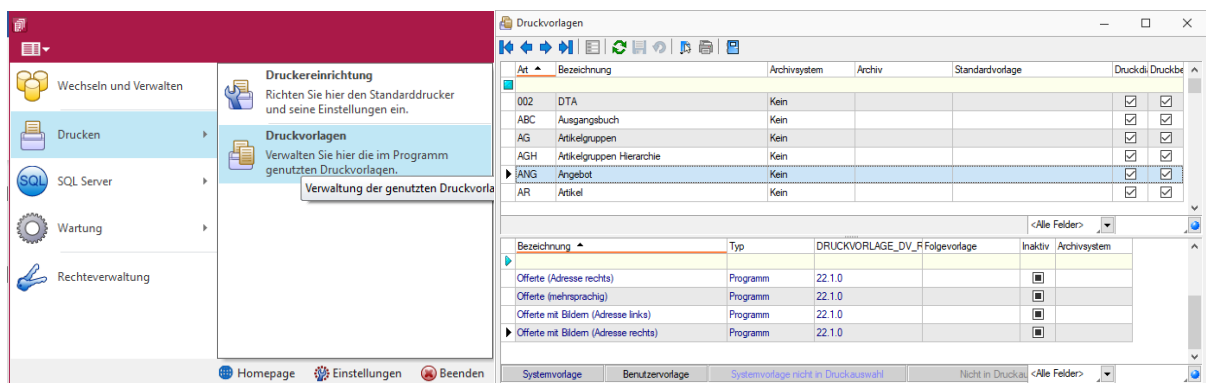
2.1 Verwaltung und Struktur

Mit dem Formulareditor können die mitgelieferten Formulare (Druckvorlagen) individuellen Anforderungen angepasst und neue Formulare entworfen werden.

Den Formulareditor erreicht man ausschliesslich aus der Bildschirmansicht eines jeden Druckvorgangs, auch aus dem Druck einer SQL-Ergebnismenge.

Im Bearbeitungsmodus können immer nur Formulare der gerade zu Grunde liegenden Druckvorlagenart modifiziert werden.

Eine Übersicht zu allen verfügbaren Druckvorlagenarten und den zugehörigen Formularen steht unter „Applikationsmenü/Drucken/Vorlagen“ im Übersichtsmodus zur Verfügung.



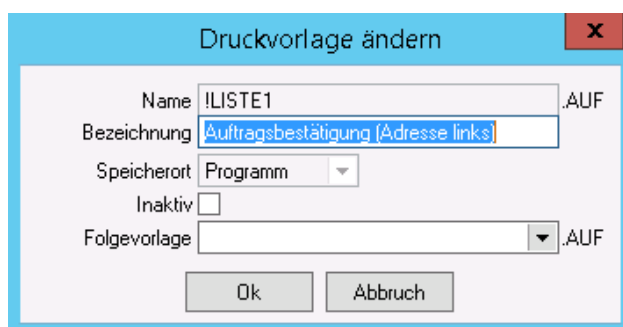
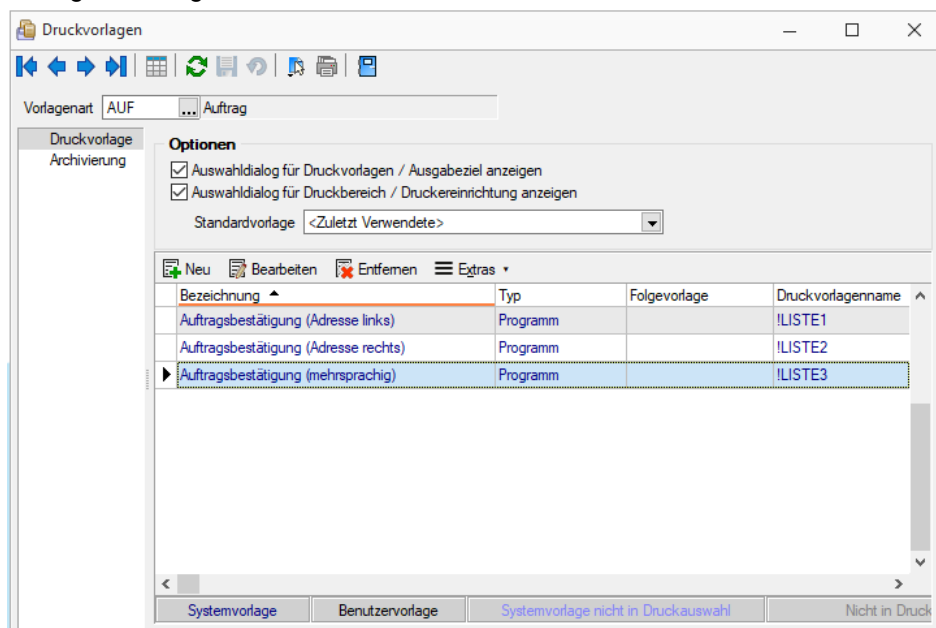
Die Verwaltung der Druckvorlagen ist generell in drei „Typen“ unterteilt.

- mandantenspezifisch
- programmspezifisch
- länderspezifisch

Druckvorlagen vom Typ = Mandant sind nur im aktuellen Mandanten verfügbar. Der Typ = Programm ist mandantenübergreifend. Diese Druckvorlagen können in jedem Mandanten ausgewählt und verwendet werden. Die von SelectLine ausgelieferten Systemdruckvorlagen entsprechen dem Programm- Typ. Des Weiteren gibt es noch länderspezifische Druckvorlagen z.B. für den Fibu-Export die als Typ das Land (Deutschland/Österreich/Schweiz) aufweisen.

Die Entscheidung, zu welchem Typ eine Druckvorlage zugeordnet wird, ist bei der Neuanlage bzw. bei der Integration zu treffen.

Durch das Umschalten in den Bearbeitungsmodus [F11] ist die Verwaltung aller Formulare einer Vorlagenart möglich.



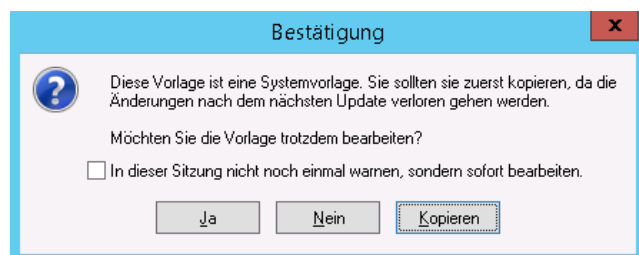
Über **Anlegen** oder **Bearbeiten** ist es möglich neue Druckvorlagen anzulegen bzw. bestehende zu modifizieren. Die einzutragende Bezeichnung erscheint im Druckdialog zur Formulare Auswahl. Sie kann bis zu 80 Zeichen lang sein.

Sollte die Bezeichnung bei zwei oder mehreren Druckvorlagen identisch sein, wird für die Sortierung und Unterscheidung der Dateiname in Klammern mit hinzugefügt.

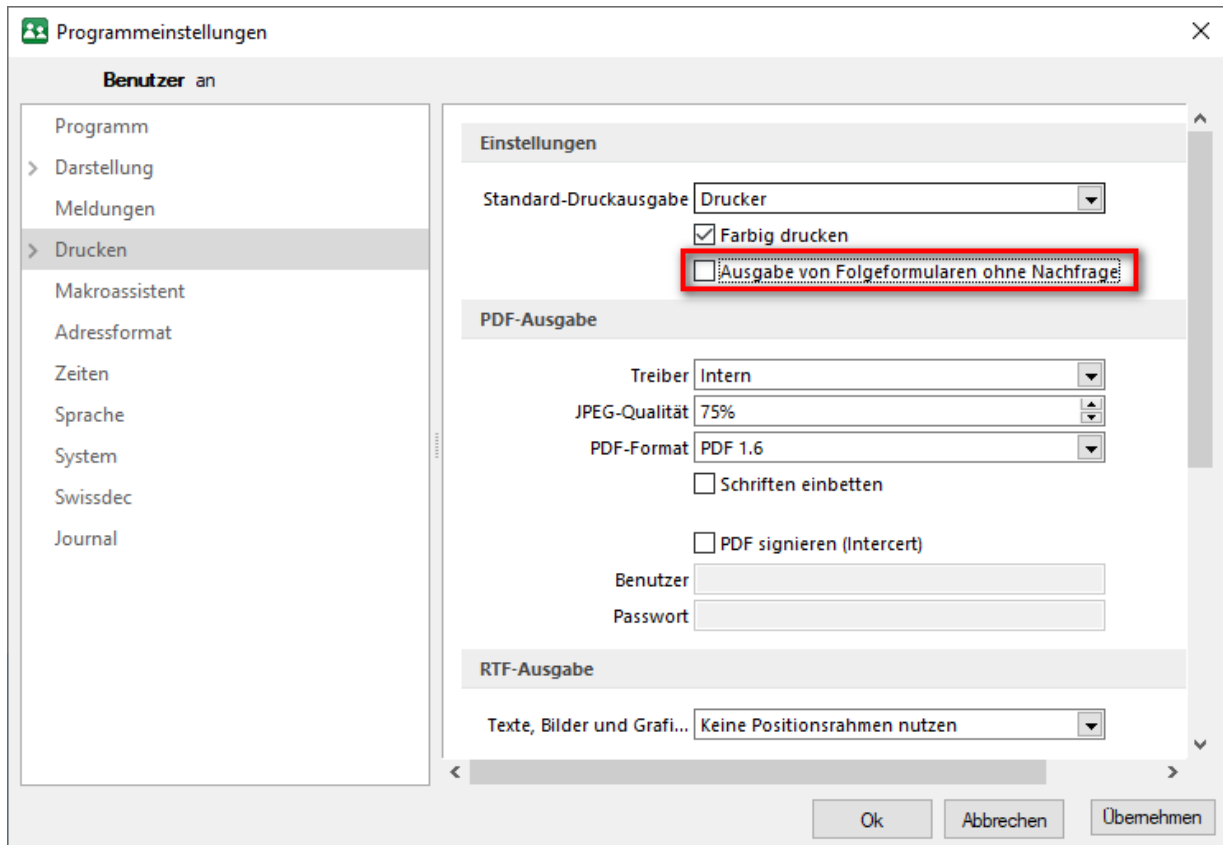
Die Dateinamensvergabe war bis zur Version 12.x noch für die externe ASCII-Datei entsprechend der Konvention „8+3“ notwendig. **Seit der Version 13 werden die Druckvorlagen in der SQL-Datenbank abgespeichert.** Inaktive Druckvorlagen werden nicht im Druckauswahldialog aufgelistet.



Ein vorangestelltes „!“ kennzeichnet **die Systemvorlagen, deren Modifikation nicht empfohlen wird, da sie bei jedem Update aktualisiert** und somit alle Änderungen des Anwenders überschrieben werden. Es wird ein entsprechender Warnhinweis eingeblendet, wenn eine Systemdruckvorlage zum Bearbeiten geöffnet wird. Systemdruckvorlagen werden blau angezeigt.



Mit der Definition „Folgeformular“ besteht die Möglichkeit, im Anschluss an den Ausdruck einer Druckvorlage eine weitere Vorlage zu drucken. Ab der Version 19.3 besteht die Möglichkeit, die Abfrage zur Ausgabe des Folgeformulars unterbinden, dazu ist die Checkbox „Ausgabe von Folgeformularen ohne Nachfrage“ ohne Nachfrage“ unter "Programmeinstellungen / Applikationsmenü / Einstellungen / Drucken"



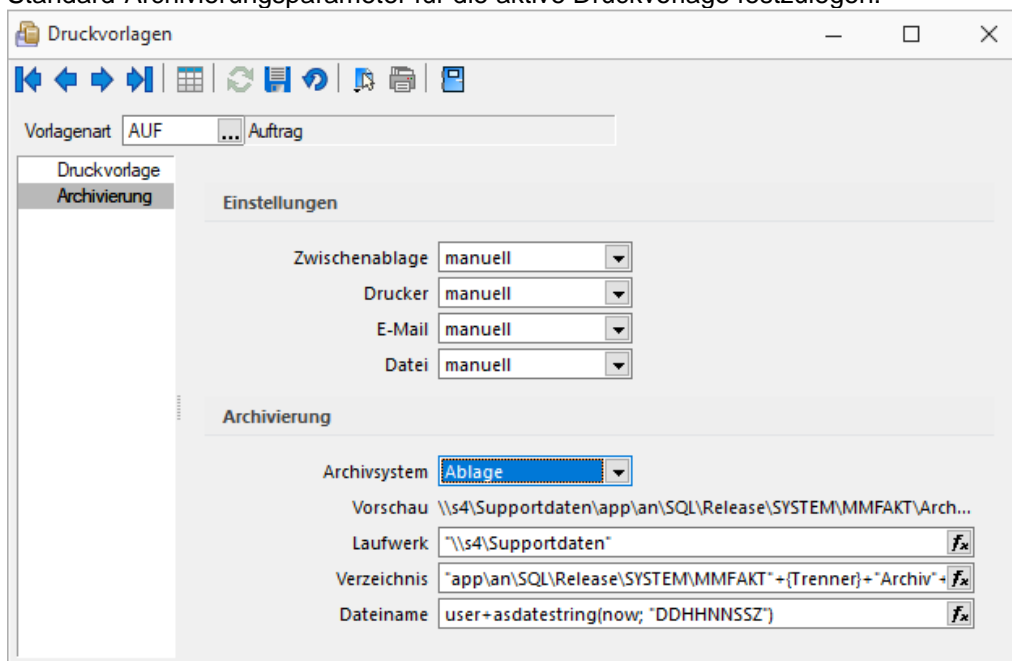
Beim Druck von Folgeformularen aus der Bildschirmvorschau heraus wurde bisher vor der Ausgabe des Folgeformulars eine Sicherheitsabfrage angezeigt, die mit „Ja“ bestätigt werden musste, um fortzufahren. Dies kann vor allem bei der Ausgabe von Belegen über mehrere unterschiedliche Ausgabearten in einer Formulkette als störend empfunden werden.



Um den Versand von Belegen mit Folgeformularen (z. B. Rechnung und Überweisungsträger) zu erleichtern, können seit der Version 21.2.x durch den Platzhalter **„PDF-MitFolgeformularen“** eine Datei erzeugen, die den Beleg und alle Folgeformulare beinhaltet. Diese per PDF-Creator oder Gnostice erzeugte Datei kann dann als Anhang versendet werden.

Unter Extras stehen ausserdem die Funktionen „Kopieren“, „Einschleusen“ und „Ausschleusen“ zur Verfügung. Es wird empfohlen, ein neues Formular für eine Druckvorlagenart durch Kopieren eines vorhandenen zu erstellen und entsprechende Änderungen oder Ergänzungen in der Kopie vorzunehmen, anstatt mit einem neuen, leeren Formular zu beginnen. Das „Kopieren“ ist nicht geeignet, um Druckvorlagen systemübergreifend zu vervielfältigen. Dafür stehen die Funktionen Aus- und Einschleusen zur Verfügung. Um sicherzustellen, dass die im Formular verwendeten Schriften, Formate und Etiketten Bestandteil des Duplikats sind, müssen die Formulare von der „Quelle“ per "Ausschleusen" zum Ziel per „Einschleusen“ übertragen werden. Dabei wird eine spezielle DVL-Datei erzeugt, die das Formular und die entsprechenden Schrift-, Format- und Etiketteninformationen enthält.

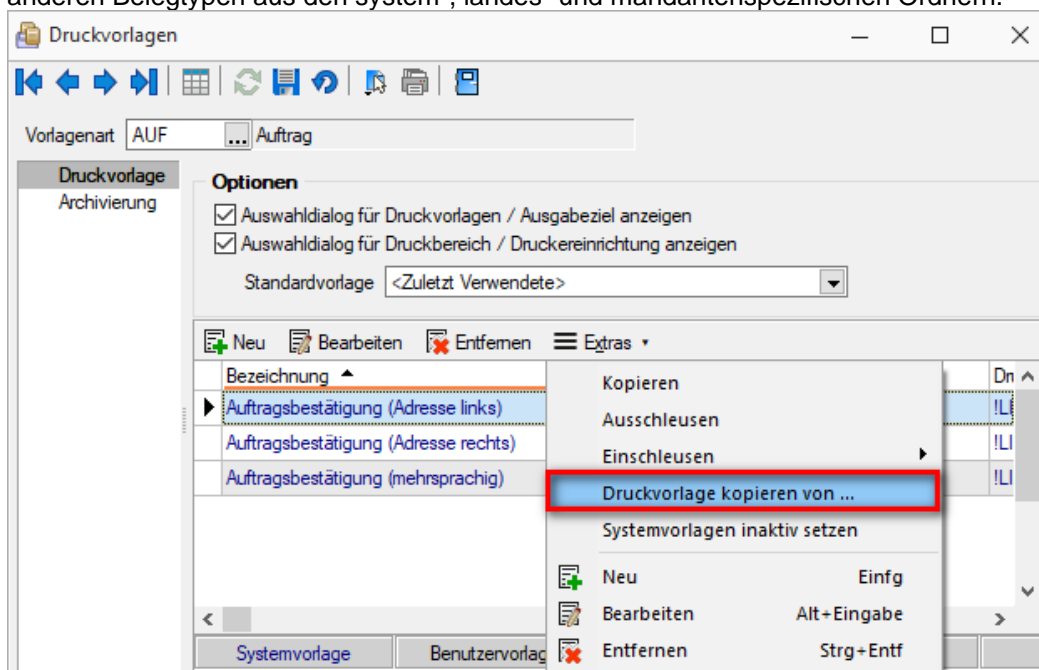
Die gleichen Bearbeitungsmöglichkeiten stehen im Formulareditor unter „Vorlagen verwalten [F11]“ zur Verfügung. Allerdings stehen dort nur noch die für die im Vorfeld gewählte Druckvorlagenart spezifischen Formulare zur Auswahl zur Verfügung. Zusätzlich können hier die Einstellungen der Dialoge „Vorlage/Ausgabeziel“ und „Drucken“ definiert werden. Ist eine dieser Optionen nicht gesetzt, entfällt bei der Ausgabe der entsprechende Dialog. Die Schaltfläche „Archivierung“ bietet eine Möglichkeit, den Standard-Archivierungsparameter für die aktive Druckvorlage festzulegen.

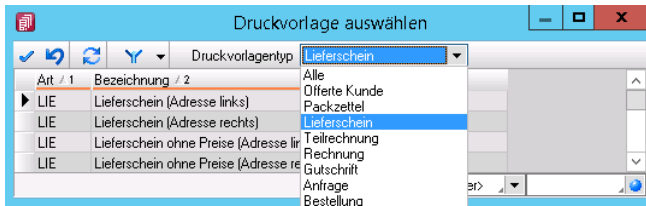


Im Reiter Archivierung der Baumstruktur ist die Liste der bereits archivierten Dokumente jahresweise geordnet und einzusehen.

2.2 Besonderheit Auftrag: Druckvorlage kopieren von...

In der Druckvorlagenverwaltung können Vorlagen von anderen Belegtypen kopiert werden. Der Menüpunkt „Extras/Kopieren von...“ öffnet einen Auswahldialog mit den vorhandenen Druckvorlagen der anderen Belegtypen aus den system-, landes- und mandantenspezifischen Ordnern.





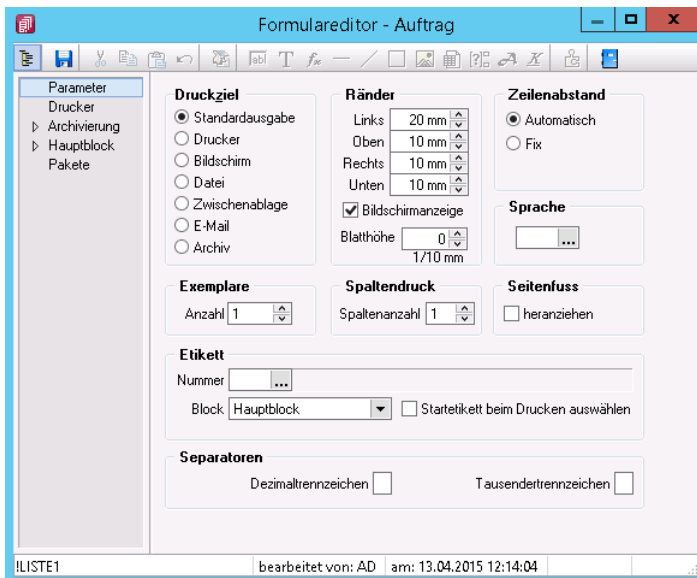
Der Auswahlschalter [F10] oder ein Doppelklick kopiert die markierte Vorlage. Im sich öffnenden Fenster muss die Bezeichnung, der Speicherort (Programm, Land, Mandant) sowie der Dateiname angegeben werden. Optional kann ein Folgedateiname vergeben werden.



In der kopierten Vorlage werden alle Parameter und Platzhalter für den aktuellen Belegtyp angepasst. Textplatzhalter müssen allerdings von Hand geändert werden.

2.3 Parameter

Hier werden unabhängig vom Inhalt Formulareinstellungen bezüglich des Druckziels, der Ränder, des Zeilenabstands, der Blatthöhe, des Spaltendrucks, der Anzahl der Exemplare, der Auswahl der Sprache, des Seitenfusses und der Etikettenart definiert.



"Standardausgabe" als Druckziel bedeutet, dass die Ausgabe immer entsprechend "Standard-Druckausgabe" gemäss „Applikationsmenü/Einstellungen/Drucken“ erfolgt. Soll das Ausgabeziel für diese Vorlage abweichend von anderen definiert werden, muss die Einstellung des gewünschten Druckzieles hier erfolgen. Diese Änderung gilt nicht für andere Vorlagen.

Der Abstand zwischen zwei Zeilen kann mit dem Zeilenabstand (Masseinheit Zehntelmillimeter) festgelegt werden oder er wird in Abhängigkeit von der grössten verwendeten Schrift automatisch berechnet.

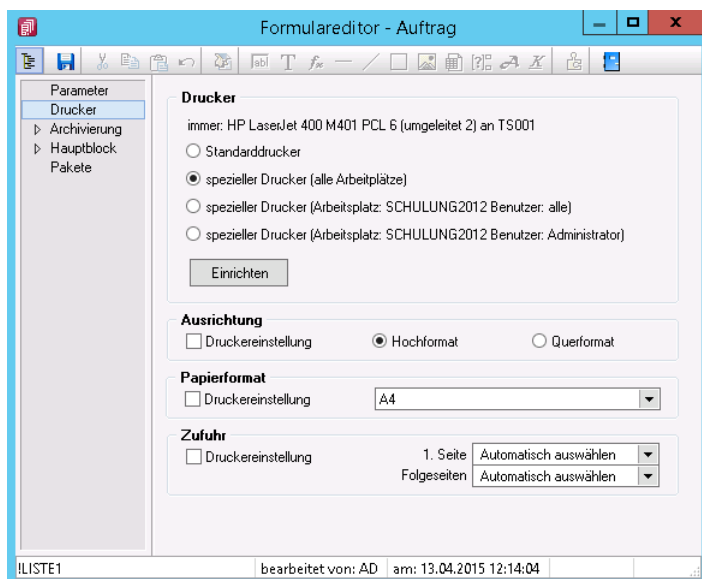
Für das Bedrucken von Etikettenformularen wird das gewünschte Etikettenformat dieser Vorlage zugeordnet. Wenn das Etikettenformular nicht einzeln, zeilenweise und links oben beginnend bedruckt werden soll, muss die Option "Startetikett beim Drucken auswählen" aktiviert werden. Über die Angabe des Formularblocks wird die Information, welcher Block für Etikettenanzahl und Inhalt ausschlaggebend ist, festgelegt, z. B.:

Hauptblock – je Position wird ein Etikett gedruckt (über „Labelcount“ auch entsprechend der Positions menge, aber mit identischem Inhalt).

Block Lagerinformation – je Seriennummer einer Position wird ein spezifisches Etikett gedruckt.

2.4 Drucker

Für die verfügbaren Einstellungen ist zu beachten, dass die Druckertreiber der einzelnen Arbeitsplätze in den häufigsten Fällen eine mögliche Problemursache darstellen. Die Anwendung nutzt dabei ausschliesslich die Funktionalitäten des vorhandenen Druckertreibers bezüglich der verfügbaren Geräteeinstellungen. So können z.B. nur die Schriftarten und auch Papiergrössen genutzt werden, die der jeweilige Treiber anbietet. Dies kann erfahrungsgemäss für dasselbe physikalische Gerät unter verschiedenen Betriebssystemen differieren. Im Bedarfsfall muss ein anderer, funktionellerer Treiber beschafft und benutzt werden.



Standarddrucker

Damit definieren Sie den Drucker des ausführenden (druckenden) Arbeitsplatzes, der zu Beginn der Anwendungssitzung als Standarddrucker des lokalen Betriebssystems definiert war. Diese Einstellung kann unter „Applikationsmenü/Drucken/Druckereinrichtung“ eingesehen werden und ist somit absolut arbeitsplatzabhängig (Ausnahme Terminalserversitzungen).

spezieller Drucker (alle Arbeitsplätze)

Hier wird ein eindeutig über den Namen spezifiziertes Druckausgabegerät benutzt, unabhängig von welchem Arbeitsplatz welche Benutzeranmeldung den Druckauftrag auslöst. Die Treiberinstallation für die Arbeitsplätze und die Berechtigungen der Benutzer für das jeweilige Gerät werden dabei vorausgesetzt. Dieses Gerät muss nicht als Standarddrucker im Betriebssystem definiert sein.

spezieller Drucker (Arbeitsplatz: PC-Name Benutzer: alle)

Es wird damit eingestellt, dass an einem konkreten Arbeitsplatz für jeden beliebigen Benutzer genau auf diesem einen zugewiesenen Drucker ausgegeben wird.

spezieller Drucker (Arbeitsplatz: PC-Name Benutzer: Windows-Benutzeranmeldung)

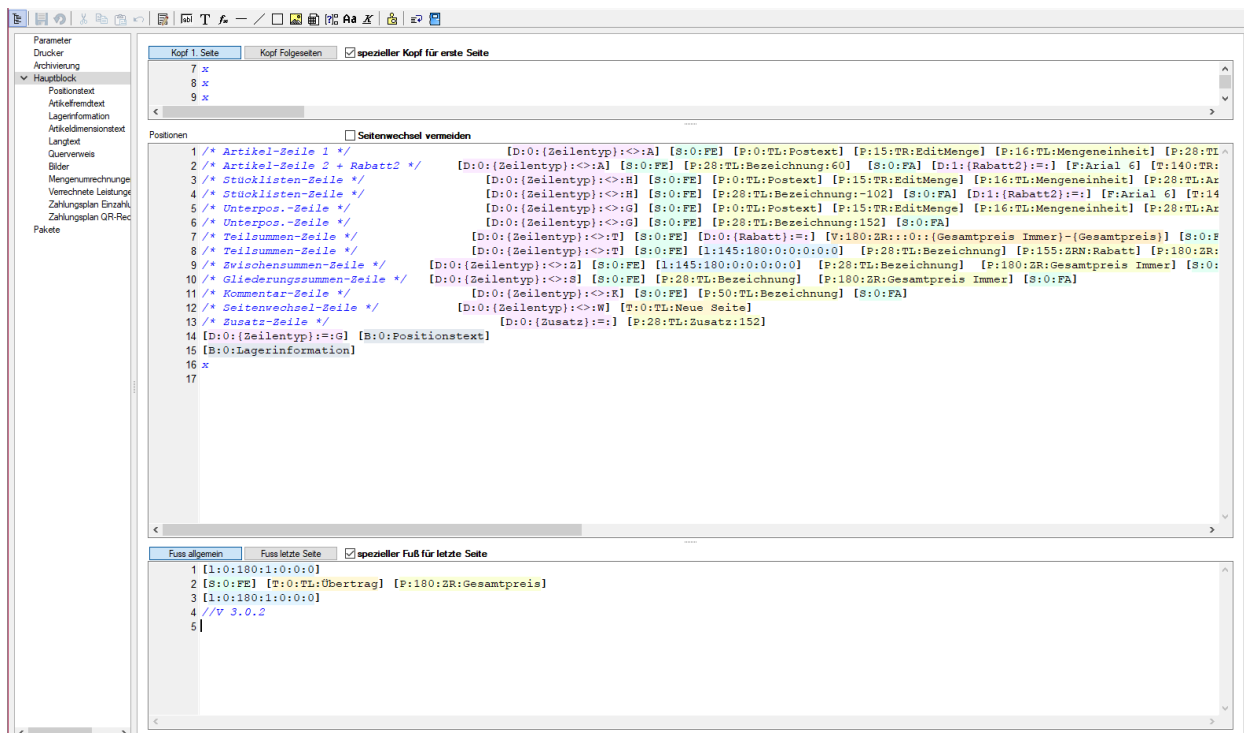
Diese Einstellung gilt für genau eine Maschine (Arbeitsplatz) in unmittelbaren Zusammenhang mit einer eindeutigen Benutzeranmeldung. Für andere Benutzer an diesem oder auch für diesen Benutzer an anderen Arbeitsplätzen gilt diese Einstellung nicht.

2.5 Anpassungen von Formularen noch komfortabler (ab 21.3)

Um Ihnen die Bearbeitung und Erstellung von Formularen zu vereinfachen, können die verschiedenen Platzhalter nun einfach über ihre Farben unterschieden werden. Die optimierte Darstellung der Begrenzung der Platzhalter ([abc]) macht es Ihnen zudem einfach, den Anfang und das Ende eines Platzhalters zu erkennen.


Die Darstellung der Platzhalter können Sie nach Ihren Vorlieben anpassen. Nutzen Sie hierfür die „Anzeigeeinstellungen“, welche Sie im Kontextmenü der Platzhaltereingabebereiche aufrufen können. Hier können Sie festlegen, welche Schriftart für die Platzhalter verwendet werden soll und in welcher Schriftgröße diese dargestellt werden.

Unter den Platzhaltereinstellungen können Sie für die unterschiedlichen Platzhalterarten individuelle Hintergrund- und Schriftfarben festlegen. Dazu wählen Sie auf der linken Seite eine gewünschte Platzhalterart aus und nehmen auf der rechten Seite die gewünschten Einstellungen vor.



Wenn Sie gerne alles im Blick haben, bietet Ihnen der Schalter „Automatischer Zeilenumbruch“ die Möglichkeit, den Inhalt einer Zeile immer komplett zu sehen, ohne horizontal scrollen zu müssen. Für eine ordentliche Ansicht der verwendeten Platzhalter ist ein einheitlicher Abstand zwischen den Platzhaltern hilfreich. Nutzen Sie hierfür die Funktion „Platzhalterabstände korrigieren“ im Kontextmenü. Anschliessend finden Sie zwischen allen Platzhaltern genau ein Leerzeichen als Trennung.

2.6 Blöcke

Bei der Anlage einer Druckvorlage ist die je Vorlagenart unterschiedliche Blockstruktur zu beachten. Es gibt immer einen Hauptblock. Abhängig von der Art der Auswertung können mehrere hierarchisch angeordnete Unterblöcke existieren, für die jeweils ein Explorereintrag existiert. Zur Bearbeitung eines Blockes wird der entsprechende Explorereintrag bearbeitet. Um den Inhalt des Unterblocks zum Ausdruck zu bringen, ist es zusätzlich notwendig, im Hauptblock einen entsprechenden Blockplatzhalter für den gewünschten Unterblock zu definieren. 

Formulareditor - Kunden

Parameter
Drucker
Archivierung
Hauptblock
 Kommentar
 Lieferadresse
 Liefertext
 Kontakt
 Kontakttext
Memo
Bankverbindungen
Kundenpreise
 Preisstaffel
 Mengenstaffel
Belegabgabe
Journal
Historie
Bilder

☐ spezieller Kopf für erste Seite

[F:Anial 8] [T:0.TL:Mandant:] [P:0.TL:System.Mandant.Mandant:] [T:0.TL:/] [P:0.TL:System.Mandant.Name] [P:115.DTU:System.System.Datum/Uhrzeit] [F:Anial 11] [S:0.FE] [P:90.TZ:Druckvorlagenbezeichnung] [S:0.FA] [F:Anial 8]
[t:0.180.1.0.0.0.0]
[S:0.FE] [T:0.TL:Nummer] [T:20.TL:Name] [T:100.TL:Telefon 1] [T:130.TL:Telefon 3] [T:160.TL:Fibu-Konto]
[T:100.TL:Telefon 2] [T:130.TL:Fax]
[t:0.180.1.0.0.0.0] [S:0.FA]

☒ Seitenwechsel vermeiden

[P:0.TL:Nummer] [S:0.FE] [P:20.TL:Anzeigenname:80] [P:0.Kom:Vorname] [S:0.FA] [D:2.(Telefon1):<.>] [T:100.TL:] [P:100.TL:Telefon:130] [D:2.(Telefon2):<.>] [P:130.TL:Fax:30] [P:20.TL:Anzeigenname:80] [P:0.Kom:Vorname] [S:0.FA] [D:2.(Telefon1):<.>] [T:100.TL:] [P:100.TL:Telefon:230] [D:2.(Fax):<.>] [T:130.TL:] [P:130.TL:Fax:30]
[t:0.180.1.0.0.0.0]
[B:0.Kontakt]

☒ spezieller Fuß für letzte Seite

[t:0.180.2.0.0.0.0]
[F:Anial 6] [V:0.TL::2.0:"Dokument: "+"(Druckvorlagenbezeichnung)+" ("+(Dateiname)+" - 2.3.0)"] [V:180.TR::2.0:"Benutzer: "+"(System.Benutzerkürzel)"]

LISTE1 bearbeitet von: 1 am: 16.09.2015 16:15:42 Dies ist eine Systemvorlage!

Es ist nicht möglich, in einem Formular systemseitig existierende Unterblöcke physisch zu entfernen. Es ist möglich, neue Blöcke individuell zu definieren. Diese können im Bedarfsfall auch wieder entfernt werden.

2.6.1 Frei positionierbarer Block im Kopf

The screenshot shows the 'Formulareditor - Auftrag' window. On the left, the 'Hauptblock' is selected in the tree view. The main area displays the block definition for 'Frei positionierbarer Block'. The 'Block' tab is active, showing the 'Auswahl' (Selection) section with a table of parameters. The 'Parameter' section is also visible, showing the 'X-Position' and 'Y-Position' fields, both set to 120 and 30 respectively. The 'Frei positionierbarer Block' is highlighted in the 'Auswahl' table.

Name	Typ
Pakete	Systemblock
Frei positionierbarer Block	Frei positionierbarer Block

The 'Parameter' section shows the following values:

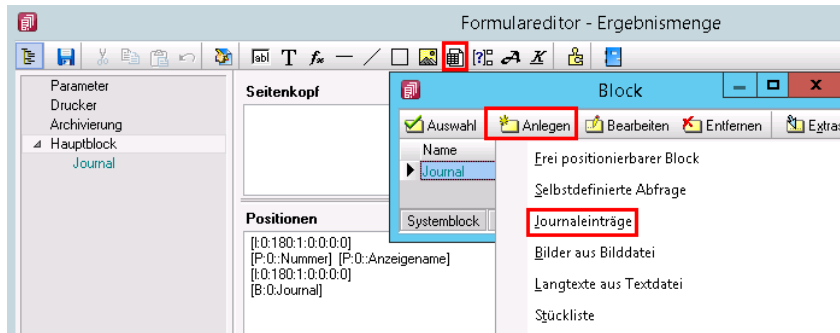
- X-Position in mm [Integer]: 120
- Y-Position in mm [Integer]: 30

The 'Frei positionierbarer Block' is highlighted in the 'Auswahl' table.

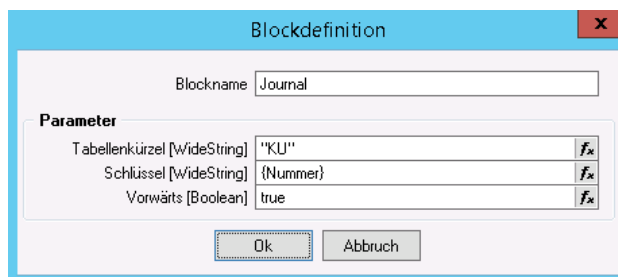
Ab der Version 14.2.0 ist es möglich einen frei positionierbaren Block zu erstellen, der die Funktion des Unterblocks um die Möglichkeit einer absoluten Positionsangabe ergänzt. Über diesen Block ist es möglich, dass nun ohne grossen Aufwand z.B. der Adressteil im Belegkopf zweispaltig aufgebaut werden kann, ohne ungewollte Leerzeilen auszugeben.

Im Gegensatz zu normalen Blockplatzhaltern kann dieser auch im Formularkopf verwendet werden.

2.6.2 Block mit vordefinierter Abfrage



Über den Schalter aus dem Dialog "Einfügen eines Blockplatzhalters" gelangen Sie in die Maske zur Erstellung eines Blocks mit vordefinierter Abfrage.



Vergeben Sie dem neu angelegten Block einen aussagekräftigen Namen, welcher nicht mit einem bereits bestehenden Block übereinstimmt.

Je nach gewähltem Block sind verschiedene Parameter anzugeben. Für die Parameter können auch Variablen in Form von Datenfelder ausgewählt werden. Verwenden Sie dazu .

Das Tabellenkürzel für die vordefinierten Blöcke Bilder aus Bilddatei, Langtexte aus Textdatei und Journaleinträge bezieht sich nicht auf das Tabellenkürzel der abzufragenden Tabelle (BI für Bild, TE für Text oder JN für Journal), sondern auf das Datenfeld Blobkey. Die ersten beiden Zeichen in diesem Feld stehen für das Tabellenkürzel der Haupt-Tabelle, es folgt der Schlüssel, der eventuell in eine Zeichenkette gewandelt werden muss.



Verwenden Sie das Beispiel aus dem Screenshot oben in einer Makro-Abfrage auf die Kundentabelle um alle zugehörigen Journaleinträge auszugeben.

2.6.3 Block mit Blockdefinition (Selbstdefinierte Abfrage)

Über den Schalter aus dem Dialog "Einfügen eines Blockplatzhalters" gelangen Sie in die Maske zur Erstellung eines Blocks mit einer selbstdefinierten Abfrage.

Geben Sie einen beliebigen Blocknamen ein. Hierbei dürfen keine Sonderzeichen verwendet werden. Der Blockname sollte so gewählt werden, dass er nicht mit bereits vorhandenen Blöcken übereinstimmt. Formulieren Sie im Textfeld die Anweisung für die Abfrage der gewünschten Datenmenge. Hierbei ist ein Ausdruck zu definieren, der eine Zeichenkette liefert und eine Select-Anweisung ohne das führende select enthält. Mit dem Schalter "Datenfeld" können Sie Platzhalter der Auswertung auswählen.



Innerhalb einer Makro-Abfrage auf die Kundentabelle sollen je Kunde die erfassten Kontaktadressen ausgedruckt werden.

Eingabe im Feld Anweisung:

" * from [CRM_ADRESSEN] where hauptadresse = ' ' + {CRM_AdressenID} + ' ' "

Beim Beenden/Speichern der neuen Blockdefinition erfolgt im Hintergrund ein Test der formulierten Anweisung. Dazu werden für alle verwendeten Datenfelder (Variablen), im Beispiel für die CRM_AdressenID, Werte abgefragt. Die Eingaben an dieser Stelle müssen nur dem Feldtyp (z.B. String/Float) entsprechen, im Beispiel "1".

Im Anschluss steht der selbstdefinierte Block genauso wie die Blöcke des Programms zur weiteren Bearbeitung zur Verfügung.

2.6.4 Seitenwechsel vermeiden

Im Positionsteil kann für jeden Block die Option "Seitenwechsel vermeiden" gesetzt werden. Dadurch werden alle Zeilen der Position auf die nächste Seite gedruckt, wenn sie nicht gemeinsam auf die aktuelle Seite passen.

The screenshot shows the 'Positionen' dialog box. The 'Seitenwechsel vermeiden' checkbox is checked. The list of positions includes:

- Artikel-Zeile 1 */ [D:0:{Zeilentyp}:<:A] [S:0:FE] [P:0:TL:Postext] [P:15:TR:EditMenge] [P:16:TL:Mengeneinheit] [P:28:TL:Artikelnummer] [P:125:ZR:Einzelpreis] [F:Ar
- Artikel-Zeile 2 + Rabatt2 */ [D:0:{Zeilentyp}:<:A] [S:0:FE] [P:28:TL:Bezeichnung:60] [S:0:FA] [D:1:{Rabatt2}:=] [F:Ar:6] [T:140:TR:+ Rabatt2] [P:155:ZR:Rabatt2] [F:Ar
- Stücklisten-Zeile */ [D:0:{Zeilentyp}:<:H] [S:0:FE] [P:0:TL:Postext] [P:15:TR:EditMenge] [P:16:TL:Mengeneinheit] [P:28:TL:Artikelnummer] [P:125:ZR:Einzelpreis] [P
- Stücklisten-Zeile */ [D:0:{Zeilentyp}:<:H] [S:0:FE] [P:28:TL:Bezeichnung:102] [S:0:FA] [D:1:{Rabatt2}:=] [F:Ar:6] [T:140:TR:+ Rabatt2] [P:155:ZR:Rabatt2] [F:
- Unterpos.-Zeile */ [D:0:{Zeilentyp}:<:G] [S:0:FE] [P:0:TL:Postext] [P:15:TR:EditMenge] [P:16:TL:Mengeneinheit] [P:28:TL:Artikelnummer] [S:0:FA]
- Unterpos.-Zeile */ [D:0:{Zeilentyp}:<:G] [S:0:FE] [P:28:TL:Bezeichnung:152] [S:0:FA]
- Teileinner-Zeile */ [D:0:{Zeilentyp}:<:T] [S:0:FE] [D:0:{Rabatt2}:=] [F:Ar:6] [T:140:TR:+ Rabatt2] [P:155:ZR:Rabatt2] [F:

At the bottom, the 'spezieller Fuss für letzte Seite' checkbox is also checked.



Wenn im Positionsteil Unterblöcke zu drucken sind, werden nur die Positionszeilen vor, zwischen bzw. hinter den Unterblöcken zusammengehalten.

Durch das Zusammenhalten werden Formelplatzhalter unter Umständen mehrfach berechnet bzw. ausgeführt, was insbesondere bei der Verwendung der Funktionen shellexecute, iniget, iniput, get und put zu beachten ist.

3 Platzhalter

3.1 Art und Aufbau

Formulare werden in Zeilenform geschrieben und auch beim Drucken von oben nach unten und von links nach rechts ausgewertet.

Auswerten heisst dabei:

- für jeden Zeilenumbruch wird eine neue Zeile begonnen
- und jeder gefundene Platzhalter wird nach erfolgreicher Interpretation gedruckt.
- Platzhalter sind in eckigen Klammern eingeschlossen. Sie sollten mit mindestens einem Leerzeichen voneinander getrennt sein. Werden innerhalb eines Platzhalters eckige Klammern benötigt, sind diese zu doppeln. Kommentarzeilen beginnen mit //. Sie werden nicht ausgewertet und nicht gedruckt. Eine Zeile kann bis zu 4096 Zeichen enthalten.

Am Ende eines Bereiches werden leere Zeilen automatisch entfernt. Wollen Sie am Bereichende leere Zeilen drucken, müssen Sie in der letzten Zeile einen Kommentar - es reicht ein Leerzeichen - eintragen.



Datenfeld

Ergebnisse des Programms (z.B. Datenbankfelder oder Summen)

Beispiel: [P:0::System.Mandant Anrede]



Text

Text, der auf einen Tabulator zentriert, links- oder rechtsbündig ausgerichtet wird, sowie Sonderplatzhalter.

Beispiel: [T:0::(Bei Zahlung bitte immer angeben!)]



Schrift

Schriftart festlegen, auch Neue definieren.

Beispiel: [F:Arial 10]



Schriftstil

Attribute der Schrift, z.B. Fett ein, Fett aus, etc.

Beispiel: [S:0:FE]



Linie, zeilenbezogen

Horizontale Linie innerhalb der Zeile.

Beispiel: [l:0:100:1:0:0:0:0]



Linie, absolut

Linie in beliebiger Lage, unabhängig von der aktuellen Zeile.

Beispiel: [L:0:10:20:30:0:2:0:0]



Rechteck

Rechteck, mit einer von der aktuellen Zeile unabhängigen Position.

Beispiel: [R:10:10:50:30:2:0:2]



Formel

Lässt die Definition einer Formel zu. Sonderplatzhalter für die Ausgabe in Datei, als Fax oder E-Mail (auch Serienmails und -faxe) werden über diesen Platzhalter definiert.

Beispiel: [V:3000::Netto Rabattfähig:0::{Netto}]



Block

Blöcke bzw. Gruppen einfügen.

Beispiel: [B:0:Positionstext]



Grafik

Fügt Grafiken im *.BMP-, *.JPG- und *.WMF-Format aus dem Systemverzeichnis der SelectLine ein.

Beispiel: [G:98:0:82:30:logo.bmp]



Bedingungen

Fügt Bedingungen ein.

Beispiel: [D:0:{Zeilentyp}<>G]



Formularbaustein

Fügt beliebige Textbausteine ein.

Beispiel: [X:Briefkopf oben]

3.2 Sonderfunktionen für Textplatzhalter

Der Platzhalter Text hat zwei Sonderfunktionen indem ein spezieller Text eingefügt wird:

Seitenwechsel

Mit dem Platzhalter [T:0::Neue Seite] wird ein Seitenwechsel innerhalb eines Positionsblockes erzwungen. Dieser Platzhalter kann natürlich individuell mit Bedingungen kombiniert werden.

Z.B.: [D:0:{Zeilentyp}<>:W] [T:0:TL:Neue Seite] bewirkt, dass eine neue Seite generiert wird, wenn im Beleg ein Seitenwechsel eingefügt wurde.

Absolute und relative Y-Position

Für das Bedrucken von Formularvordrucken ist es wichtig, die genaue Y-Koordinate festlegen zu können. Definieren Sie dazu einen neuen Textplatzhalter und geben Sie als Text YABS: gefolgt von der Y-Position in Zehntelmillimetern an.

Verwenden Sie z.B. den Platzhalter [T:0::YABS:250], wenn Sie die Y-Position auf 25 Millimeter vom Seitenanfang setzen wollen. Die Wirkung dieses Platzhalters gilt nur für die Positionierung der restlichen Platzhalter in derselben Zeile des Formulareditors.

Analog dazu können Sie als Text in einen Textplatzhalter YREL: gefolgt von einem Y-Wert in Zehntelmillimetern eintragen, um die Y-Koordinate relativ zur aktuellen Schreibposition zu setzen.

3.3 Verweise

Verweise können überall dort verwendet werden, wo Datenplatzhalter benutzt werden. Der Verweis ermöglicht es Daten aus einer anderen Tabelle zu lesen, indem mit dem Hauptschlüssel in die entsprechende Tabelle verwiesen wird. Steht Ihnen z.B. in einer bestimmten Druckvorlage nur die Kundennummer zur Verfügung, so können Sie über einen Verweis in die Kundentabelle die restlichen Angaben des Kunden holen. Verweise sind auch über mehrere Tabellen hinweg möglich.

Nachfolgend finden Sie einige Beispiele ausgehend vom Belegkopf eines Beleges:

[P:0::>KU~Email .Adressnummer]

Druckt das Feld Email, des im Beleg ausgewählten Kunden, aus dem Kundenstamm.

[P:0::>OW~Bezeichnung .Standort]

Druckt das Feld Bezeichnung, des im Beleg ausgewählten Standorts, aus dem Standort Stammdaten.

[P:0::>KG~Bezeichnung >KU~Gruppe .Adressnummer]

Druckt das Feld Bezeichnung aus den Kundengruppen. Es wird die Bezeichnung des im Beleg ausgewählten Kunden verwendet.

3.4 Formularbaustein

Von allen Platzhalter-Typen bildet der Formularbaustein eine Besonderheit. Er ist im Prinzip der Platzhalter, um einen „Bereich“ aus einer Druckvorlage auszulagern; ähnlich einem Textbaustein. Immer wiederkehrende „Abschnitte“ können so „ausgelagert“ und bei Bedarf „zentral“ (nur im Formularbaustein“ geändert werden, ohne dass jede betroffene Druckvorlage separat angepasst werden muss. In Belegen bieten sich der Adressbereich im Kopf oder die Bankverbindungen im Fuss für die Auslagerung in Formularbausteine an.

Dieser Platzhalter wird beim Ausdrucken durch die im Formularbaustein eingefügten Platzhalter ersetzt. Formularbausteine werden als separate Dateien mit der Endung .INCLUDE im Systemverzeichnis bzw. in der Datenbank abgelegt. Wird der Formularbaustein in mehreren Druckvorlagen verwendet, muss der Bearbeiter sicherstellen, dass die eingefügten Platzhalter in allen Druckvorlagen zur Verfügung stehen. Beim Aus- und Einschleusen von Druckvorlagen werden die Formularbausteine berücksichtigt und bei Bedarf im Zielsystem neu angelegt.

3.5 Seitenzahl drucken

Mit dem Datenplatzhalter System.Seitenanzahl kann die Gesamtseitenanzahl beim Drucken ausgegeben werden. Beispiel: [P:170:Sei:System.Aktuelle Seite] [T:0:~/] [Y:0:0:System.Seitenanzahl]
Dieser Platzhalter steht nicht in Formeln, Bedingungen oder selbstdefinierten Blöcken zur Verfügung. In den Druckjobs (des Rechnungswesens) liefert dieser Platzhalter nur die Gesamtseitenanzahl zu einer Vorlage und nicht die Gesamtseitenanzahl des gesamten Druckjobs. Um den Platzhalter zu ermitteln, muss intern die gesamte Ausgabe erfolgen, bevor die erste Seite angezeigt bzw. gedruckt werden kann, was insbesondere bei Verwendung in umfangreichen Dokumenten mit HTML-Platzhaltern sehr lange dauern kann.

3.6 Sonderplatzhalter

Um einen Beleg automatisch per Email zu versenden, verwenden Sie im Kopfteil einen Formelplatzhalter. Als Name ist „**Email**“ einzutragen und als Aktion „zuweisen“ einzustellen.
[V:300::Email:2:0::(>KU~Email .Adressnummer)].

Dieser Platzhalter „Email“ ist Voraussetzung, dass die folgenden Mail-Platzhalter ausgeführt werden!
Der Ausdruck selbst wird immer in eine Datei erfolgen und an die erzeugte Email als Anhang angebunden.
Der Dateiname des Anhangs entspricht dabei der Bezeichnung der Druckvorlage + Benutzerkürzel +Datum+ Uhrzeit.

Soll der Dateiname „abweichend“ von diesem Schema erfolgen, verwenden Sie zusätzlich einen der „Datei“-Sonderplatzhalter (DateiPDF; DateiTXT...).

Um einen Betreff selbst zu definieren (standardmässig steht dort der Mandantennamen), kann man in einen Formelplatzhalter mit Namen „**Email-Betreff**“ den Betreff festlegen.
[V:300::Email-Betreff:2:0::"Text "+{.Belegnummer}]

Wird dieser Platzhalter nicht belegt, erscheint wie bisher der Mandantennamen. E-Mail-Texte kann man über einen Formelplatzhalter mit Namen „**Email-Text1**“ (bis "Email-Text19" aufklappen) festlegen.
[V:300::Email-Text1:2:0::"Mein Text"]

Weiter kann eine Lesebestätigung (**Email-Lesebestätigung**) eingefordert werden; CC (**Email-CC**) und BCC-Mailadressen (**Email-BCC**) übergeben werden und bis zu 9 zusätzliche Dateien an die zu erzeugende Email angehängt (**Email-Anhang1** bis Email-Anhang9) werden.

Die automatische Versendung von Belegen per Fax erreichen Sie mit gleicher Verfahrensweise über den Formelplatzhalter namens „**FAXNummer**“.
[V:300:: FAXNummer:2:0::(>KU~Fax .Adressnummer)]

Soll die Anzahl der zu druckenden Etiketten aus bestimmten Abhängigkeiten ermittelt werden, verwenden Sie im entsprechenden Blockbereich einen Formelplatzhalter. Als Name ist „**LabelCount**“ einzutragen und als Aktion "zuweisen" einzustellen.
[V:1000::LabelCount:2:0::{Menge}]

Beim Drucken in eine Datei wird der Dateiname nicht abgefragt, wenn ein Formelplatzhalter mit Sonderfunktion „**DateiTXT**“, „**DateiRTF**“ oder „**DateiPDF**“ mit dem entsprechenden Dateinamen versehen wird. Es kann nur der Dateiname oder zusätzlich eine Kombination mit dem Dateipfad erfolgen. Vorhandene Dateien gleichen Namens werden kommentarlos überschrieben.

[V:1000::DateiRTF:2:0::({.Adressnummer}+"_"+{.Belegnummer}+".RTF")]

Mit diesen Platzhaltern wird auch der standardmässige Dateiname beim E-Mail-Versand „überschrieben“.

In Zeiten von Kodierungen kann eine Text-Ausgabe unterschiedlich kodiert werden (ANSI(OEM)= DateiANSI; ASCII(7-bit)= DateiASCII oder UTF-16 = DateiU16 bzw. UTF-8 = DateiTXT).

Soll das Folgeformular in Abhängigkeit bestimmter Bedingungen festgelegt werden, kann dies über die Sonderfunktion „**Folgeformular**“ geregelt werden.

[V:1000::Folgeformular:2:0::BLATT2]

Eine erstellte Datei wird über die Sonderfunktion „**Execute**“ im angegebenen Programm geöffnet.

[V:400::DateiPDF:2:0::"C:\TEST.PDF"] [V:400::Execute:2:0::"C:\TEST.PDF"]

Serienmails und -faxe

Wenn das Programm die Änderung der Adressnummer im Formelplatzhalters "EMail" bzw. "FAXNummer" erkennt, wird die aktuelle Mail / das aktuelle Fax abgeschlossen und die nächste Mail / das nächste Fax für die nächste Adresse erstellt.

Sonderfall Bild

Über einen Formelplatzhalter mit folgendem Aufbau ist es möglich, Artikelbilder in Auswertungen zu integrieren:

- #7 – programminterner Code für das Handling
- + – verknüpft die einzelnen Elemente der String-Konstanten
- „AR“ – ist der Blobkey der Tabelle
- {Artikelnummer} – wird als Datenfeld ausgewählt.

Um ein Bild einer bestimmten Ordnung zu drucken dem Konstrukt noch #7 und die Ordnungsnummer anzuhängen.

Über das Format können Sie die gewünschte Bildgrösse definieren.



Sorgen Sie dafür, dass ein bestimmter Artikel mehr als ein Bild hinterlegt ist und drucken Sie diesen auf den Bildschirm. In der Druckvorlage verwenden Sie nachfolgenden Platzhalter.

[V:140:BldN::0::#7+'AR'+{Artikelnummer}+#7+'0']



Beachten Sie, dass die Ordnungsnummer bei 0 beginnt.

4 Formeln in Formelplatzhaltern

Operanden können Konstanten, Datenfelder oder Formelfelder sein.

- **Numerische Konstanten:**
Konstanten können in dezimaler, binärer, oktaler oder hexadezimaler Schreibweise eingegeben werden.
Bei der binären Schreibweise ist ein 'b' anzuhängen, bei der oktalen ein 'o' und bei der hexadezimalen ein 'h'.
Bei gebrochenen Zahlen ist das in Windows eingestellte Dezimalzeichen zu verwenden.
Ausschliesslich bei der (normalen) dezimalen Schreibweise ist auch eine Darstellung in Exponentialdarstellung möglich. Dabei ist zuerst die Mantisse anzugeben, danach ein 'E' und am Ende der Exponent zur Basis 10.
Eine vordefinierte Konstante ist **pi** mit 3.14159....
- **Logische Konstanten:**
Logikkonstanten sind **true** und **false**;
- **Zeichenkettenkonstanten / -funktionen:**
Zeichenkettenkonstanten sind in Anführungszeichen " einzuschliessen. Innere Anführungszeichen sind zu doppeln.
user (liefert das zweistellige Benutzerkürzel)
- **Datumskonstanten:**
date (liefert das aktuelle Rechnerdatum)
now (liefert das aktuelle Rechnerdatum mit Zeitangabe)
- **Konstante für leere Felder:**
null
Null-Werte sollten in Formularen nicht an Formelplatzhalter zugewiesen werden. Sie können für die Auswertung von select- und dbf-Anweisungen benutzt werden. Der Vergleich mit anderen Typen liefert stets **false**.
- **Variablen** in Formeln (**variable Operanden, Datenfelder**) werden über Auswahlshalter eingefügt und mit geschweiften Klammern gekennzeichnet. Formelausdrücke können Zeichenketten, Zahlen, Wahrheitswerte und eingeschränkt auch null-Werte (unbestimmt vom Wert und Typ) verarbeiten und als Ergebnis liefern.

4.1 Konstanten

	Darstellung	Beispiele
numerisch	pi und Zahlen	1 99,753 pi 1.2E7
logische	true für wahr false für falsch	true false
String	sind in " einzuschliessen	"HP DeskJet 845C " "1" "pi" "true" " " //Zeichenkette ohne Inhalt

4.2 Variablen

	Darstellung	Beispiele
Numerisch	Platzhalter vom Typ Integer, SmallInt, Float, Date, AutoInc	{FreieZahl1} {>LA~Zaehlliste Lager} {System.Stift X}
logische	Platzhalter vom Typ Boolean	{FreiesKennzeichen1} false
String	Platzhalter vom Typ String	{Lager} {System.Mandant Name}

4.3 Funktionen

- Parameter sind im Allgemeinen in runde Klammern zu setzen (Ausnahmen z.B. +, mod, not, ...)
- mehrere Parameter mit Semikolon zu trennen
- Leerzeichen zwischen Parametern können beliebig gesetzt werden
- Funktionsbezeichner enthalten keine Grossbuchstaben

4.3.1 für numerische Operanden

Funktion	Beschreibung	Beispiel
+	Addition	2 + 3 // liefert 5
-	negativer Wert oder Subtraktion	
*	Multiplikation	2 + 3 * 4 // liefert 14 (2 + 3) * 4 // liefert 20
/	Division	
mod	Modulo-Funktion	30 mod 7 // liefert 2
div	ganzzahlige Division	30 div 7 // liefert 4
^	Potenz	2 ^ 4 // liefert 16
!	Fakultät	!5 // liefert 120
ln	natürlicher Logarithmus (Basis e)	ln(2,71828182845905) // liefert 1
ld	Logarithmus zur Basis 2	ld(2) // liefert 1
lg	dekadischer Logarithmus (Basis 10)	lg(10) // liefert 1
abs	Absolutbetrag	abs(-11) // liefert 11
int	ganzzahliger Anteil	int(-1,7) // liefert -1
sign	Signumfunktion (Ergebnis ist : -1 für Werte <0, 1 für Werte >0, 0 bei 0)	sign(11) // liefert 1
frac	gebrochener Anteil (Bsp.: frac -1,7 ergibt - 0,7)	frac(-1,7) // liefert -0,7
rdm	Zufallszahl kleiner als der Operand erzeugen	rdm(4) // liefert 0, 1, 2 oder 3
rnd	Rundung	rnd(1,812) // liefert 2
rn2	Rundung auf 2 Nachkommastellen	rnd(1,812) // liefert 1,81
sqrt	Quadratwurzel	sqrt(9) // liefert 3
sin	Sinusfunktion	
cos	Kosinusfunktion	
tan	Tangensfunktion	
cot	Kotangensfunktion	

4.3.2 Bit-Operationen

Funktion	Beschreibung	Beispiel
not	logische Negation der Binärwerte	not 10 // liefert -11
and	logische Undverknüpfung der Binärwerte	10 and 6 // liefert 2
ior	logische Oderverknüpfung der Binärwerte	10 ior 6 // liefert 14
xor	log. Exklusivoderverknüpfung der Binärwerte	10 xor 6 // liefert 12
shl	bitweises Verschieben nach links	1 shl 3 // liefert 8
shr	bitweises Verschieben nach rechts	11 shr 1 // liefert 5

4.3.3 für Zeichenketten

Funktion	Beschreibung	Beispiel
+	Addition (Aneinanderkettung von Zeichenketten)	„Offene Posten in „ + {Währung}
upper	Grossschreibung	upper({Artikelnummer})
lower	Kleinschreibung	
substring	Teilzeichenkette 1. Parameter: Ausgangszeichenkette 2. Parameter: Startposition (numerisch) 3. Parameter: Länge (numerisch) der 3. Parameter kann weggelassen werden	// 2. bis 5. Zeichen der Artikelnummer substring({Artikelnummer};2;4)
asfilename	Für Dateinamen nicht erlaubte Zeichen der Ausgangszeichenkette werden durch ein Ersatzzeichen ersetzt; 1. Parameter: Ausgangszeichenkette 2. Parameter: Ersatzzeichen	asfilename("A < > ? " : \ / * Z";"x") liefert "A x x x x x x x x Z"
locate	liefert in einem numerischen Ergebnis die erste Position der Suchzeichenkette in einer Ausgangszeichenkette, falls die Suchzeichenkette nicht enthalten ist, ist das Ergebnis -1 1. Parameter: Suchzeichenkette 2. Parameter: Ausgangszeichenkette	// Teil der Zeichenkette bis zum Leerzeichen substring({.Name};1;locate(" ";{.Name}))-1
length	liefert in einem numerischen Ergebnis die Anzahl der Zeichen in der Zeichenkette	length({Konto})
concat	verbindet zwei Zeichenfolgenausdrücke durch einen Verkettungsausdruck; 1. Parameter: eine Zeichenkette 2. Parameter: eine Zeichenkette 3. Parameter: Verkettungszeichenkette Ist eine der Zeichenketten leer, wird die Zeichenkette, die nicht leer ist, ohne Verkettungsausdruck ausgegeben.	concat("Mustermann";"Max";",") liefert Mustermann,Max

Funktion	Beschreibung	Beispiel
replace	ersetzt (nicht rekursiv) alle Vorkommen des zweiten Zeichenfolgenausdrucks im ersten Zeichenfolgenausdruck durch einen dritten Ausdruck; 1. Parameter: Ausgangszeichenkette 2. Parameter: Suchzeichenkette 3. Parameter: Ersatzzeichenkette Der 3. Parameter kann weggelassen werden zum Löschen aller Vorkommen der Suchzeichenkette aus der Ausgangszeichenkette.	<code>replace("Staumauer";"u";"hl")</code> liefert <code>Stahlmahler</code>
newline	mit dieser Konstante kann ein Zeilenumbruch innerhalb des aktuellen Platzhalters erzwungen werden	<code>concat({Anrede}; concat({Name}; {Vorname}; " "); newline)</code> liefert <code>Herr Mustermann,Max</code>
trim	entfernt am Anfang und Ende einer Zeichenkette alle Leer- und Steuerzeichen	
ltrim	entfernt am Anfang einer Zeichenkette alle Leer- und Steuerzeichen	
rtrim	entfernt am Ende einer Zeichenkette alle Leer- und Steuerzeichen	
lines	liefert die Anzahl der Zeilen eines Langtextes; Parameter: eine Zeichenkette	<code>//Anzahl der Zeilen von {.Kopftext}</code> <code>lines({.Kopftext})</code>
xmltag	wandelt eine Zeichenkette in das Format UTF8 um und schliesst das Ergebnis in < > ein Parameter: eine Zeichenkette	<code>xmltag("Text")</code> liefert <code><Text></code>
xmlstring	wandelt eine Zeichenkette in UTF8 um, nachdem zuvor <, > und & ersetzt wurden Parameter: eine Zeichenkette	
fill	füllt eine Zeichenkette bis zur definierten Länge auf 1. Parameter: Ausgangszeichenkette 2. Parameter: Gesamtlänge (numerisch) 3. Parameter: Füllzeichen(-kette) 4. Parameter: links auffüllen (true); rechts auffüllen (false) der 4. Parameter kann weggelassen werden (Standard ist true)	<code>fill(asstring({FreieZahl1});5;"0")</code> <code>fill({Bezeichnung};80;" ";false)</code>

4.3.4 Vergleichsoperatoren

(die Operanden müssen vom gleichen Typ sein und liefern ein logisches Ergebnis)

Operator	Bedeutung
<code>=</code>	Gleichheit
<code><=</code>	kleiner gleich
<code>>=</code>	größer gleich
<code><></code>	Ungleichheit
<code><</code>	kleiner
<code>></code>	größer

4.3.5 für Logikoperatoren

Funktion	Beschreibung	Beispiel
not	logische Negation	not ((3 > 2) and (2 > 1)) //liefert false
and	logische Undverknüpfung true, wenn beide Operanden true sind	(3 > 2) and (2 > 1) //liefert true
ior	logische Oderverknüpfung true, wenn mindestens ein Operand true ist	(3 > 2) ior (2 > 1) //liefert true
xor	logische Exklusivoderverknüpfung true, wenn genau ein Operand true ist	(3 > 2) xor (2 > 1) //liefert false
if	Fallunterscheidung 1. Parameter: logischer Ausdruck 2. Parameter: Ausdruck beliebigen Formats, welcher das Ergebnis liefert, wenn der 1. Parameter den Wert true besitzt 3. Parameter: Ausdruck beliebigen Formats, welcher das Ergebnis liefert, wenn der 1. Parameter den Wert false besitzt	//verschachteltes if if({Belegtyp}="R";1;if({Belegtyp}="G",-1;0)) //Test auf bestimmte Belegtypen if(locate({Belegtyp},"FLR")>0;"Belegtyp F, L oder R";"sonstiger Belegtyp")

4.3.6 Erkennung und Behandlung von Null-Werten

Funktion	Parameter	Beispiel
isnull	1. Parameter: ein Ausdruck von beliebigem Format(Zeichenkette, Zahl, Datum, logisch, Null-Wert) 2. Parameter: Ergebnis, wenn der erste Parameter null ist	// Ersetzung eventueller Null-Werte durch einen Leerstring isnull(select("FreierText1 from kunden where nummer = '10000'");"") // Ersetzung eventueller Null-Wert durch die Zahl 0 isnull(select("FreieZahl1 from kunden where nummer = '10000'");0)

4.3.7 Datumsfunktion

Funktion	Beschreibung/Parameter
Zinstage	ermittelt die Zinstage entsprechend der 30tägigen Zinsperiode der Bank; 1. Parameter: Beginndatum 2. Parameter: Enddatum
Zinsmonate	ermittelt die Zinsmonate; 1. Parameter: Beginndatum 2. Parameter: Enddatum
kweek	ermittelt die KW; als Parameter ist ein Datum als Variable z.B. kweek({.Datum}) oder internes Datumsformat anzugeben z.B. kweek (asdatetime("30.12.2012"))
kyear	ermittelt das Kalenderjahr zu dem die Kalenderwoche gehört; als Parameter ist ein Datum als Variable z.B. kyear({.Datum}) oder internes Datumsformat anzugeben z.B. kyear(asdatetime("30.12.2012"))

4.3.8 Umwandlungsfunktion

Funktion	Beschreibung	Beispiel
asfloat	liefert den numerischen Wert einer Zeichenkette 1. Parameter: Ausgangszeichenkette 2. Parameter: numerischer Wert, der das Ergebnis liefert, wenn die Ausgangszeichenkette nicht in eine Zahl umgewandelt werden kann. Der 2.Parameter kann weggelassen werden.	asfloat({FreierText1}) asfloat({Konto}; 0) asfloat("UNSINN"; 999) // liefert immer 999
asstring	wandelt einen numerischen Wert in eine Zeichenkette um 1. Parameter: Ausgangszahl 2. Parameter: ein Format für die Darstellung wie z.B. „#,##0.00“ der 2. Parameter kann weggelassen werden	asstring({FreieZahl1}) asstring(1E3) // liefert 1000 asstring(1E3; „#,##0.00“) //liefert 1.000,00
asdatestring	wandelt ein Datum in eine Zeichenkette um 1. Parameter: Ausgangsdatum 2. Parameter: ein Format für die Darstellung wie z.B. „MMM.YYY“ der 2. Parameter kann weggelassen werden	asdatestring({FreiesDatum1}) asdatestring(3800+ 100) //liefert 23.04.2004-06-18 asdatestring(38080; „D MMM YY“) //liefert 3 Apr 04
asdatetime	wandelt ein Datum in einen numerischen Wert (internes Datumsformat, mit dem gerechnet werden kann) um; ein Parameter vom Typ Zeichenkette: Ausgangsdatum im Windowsformat	
monat	wandelt einen numerischen Wert zwischen 1 und 12 in den Monatsnamen um, Werte > 12 bzw. < 1 geben einen Leerstring aus; z.B. monat(6) liefert "Juni"	
guid36	liefert als Ergebnis eine Zeichenkette (36 Zeichen) ohne die führende und schliessende Klammer	guid36({Kennung})

4.3.9 Datenbankfunktionen

Funktion	Beschreibung	Beispiel
select	liefert den Wert eines Tabellenfeldes als Parameter ist eine SQL-Selectanweisung anzugeben, wobei das führende select weggelassen wird	// Verweis {>LA~Bezeichnung Lager} mit select select("Bezeichnung from lager where lager = ' " + {Lager} + "'") //mit Integer-Parameter select("menge from belegp where kennung = " + asstring({Kennung}))

4.3.10 Funktionen zum Lesen und Schreiben von Einträgen in INI-Dateien

Funktion	Beschreibung	Beispiel
iniget	liest eine Zeichenkette aus einer Ini-Datei: 1. Parameter: Dateiname, Datenamen ohne Laufwerk beziehen sich auf das Mandantenverzeichnis 2. Parameter: Sektionsbezeichnung; 3. Parameter: Bezeichnung des Eintrags; 4. Parameter: Ergebnis, wenn die Datei nicht existiert oder kein Eintrag in der Ini-Datei gefunden wird. Bei Weglassen des 4. Parameters ist die leere Zeichenkette der Standardwert.	iniget(„...\\DATEN.INI“,“SYSTEM“,“OLELOGIN“); iniget(„C:\\TEST.INI“,“Sektion“,“Wert“,“xxx“)
iniput	schreibt einen Eintrag in eine Ini-Datei oder löscht einen Eintrag, liefert als Ergebnis einen logischen Wert, ob der Befehl erfolgreich war: 1. Parameter: Dateiname, Datenamen ohne Laufwerk beziehen sich auf das Mandantenverzeichnis; 2. Parameter: Sektionsbezeichnung; 3. Parameter: Bezeichnung des Eintrags; 4. Parameter: Zeichenkette, die eingetragen werden soll. Bei Weglassen des 4. Parameters wird ein eventuell vorhandener Eintrag gelöscht.	//Eintrag erzeugen iniput(„C:\\TEST.INI“,“Sektion“,“Wert“,“100“) //Eintrag löschen iniput(„C:\\TEST.INI“,“Sektion“,“Wert“)

Funktion	Beschreibung	Beispiel
put	speichert programmintern einen Wert, solange das Programm läuft: 1. Parameter: eine Zeichenkette für die Identifikation des Wertes (keine Unterscheidung von Groß- und Kleinschreibung) 2. Parameter: ein Ausdruck von beliebigem Format (Zeichenkette, Zahl, Datum, logisch, Null-Wert)	// Abspeichern des aktuellen Zeitpunktes put("zeit"; now) // Anlegen oder Ändern der programminternen Variable "variable1" mit dem logischen Wert WAHR put("variable1"; true)
get	liest programminterne Werte: 1. Param.: eine Zeichenkette für die Identifikation des Wertes (keine Unterscheidung von Gross/Kleinschreibung) 2. Parameter: Ergebnis, wenn programminterner Wert nicht vorhanden ist	// Lesen der programminternen Variable "variable1" get("variable1"; false) //ein Durchlaufzähler mit Namen "a" put("a"; 1 + get("a"; 0))

4.3.11 Funktionen zum Erstellen der IBAN-Nummer

Funktion	Beschreibung	Beispiel
ibane	bildet aus Länderkennung, Banknummer, Kontonummer die IBAN-Nummer in elektronisch lesbarer Form (ohne Leerzeichen): 1. Parameter: Länderkennung; 2. Parameter: Banknummer; 3. Parameter: Kontonummer.	ibane(„DE“; {>KB~Banknummer .KasseBank}; {>KB~Kontonummer .KasseBank})
ibanp	bildet aus Länderkennung, Banknummer, Kontonummer die IBAN-Nummer in visuell lesbarer Form (mit Leerzeichen) - im s.g. Papierformat	

4.4 Spezielle Funktionen für den Auftrag

Funktion	Beschreibung	Beispiel
bestand	liefert den aktuellen Bestand von Artikeln; 1. Parameter: die Artikelnummer 2. Parameter: der Standort 3. Parameter: ein Lager oder Lagerplatz als Ausdruck oder Datenbankfeld, der 2. und 3. Parameter können weggelassen werden	bestand({Artikelnummer};{Standort};{Lager})
bestellt	liefert die aktuell bestellte Menge von Artikeln; 1. Parameter: die Artikelnummer 2. Parameter: der Standort als Ausdruck oder Datenbankfeld, der 2. Parameter kann weggelassen werden	bestellt({Artikelnummer};{Standort})
gepackt	liefert die aktuell gepackte Menge von Artikeln; 1. Parameter: die Artikelnummer 2. Parameter: der Standort 3. Parameter: ein Lager oder Lagerplatz als Ausdruck oder Datenbankfeld, der 2. und 3. Parameter können weggelassen werden	gepackt({Artikelnummer};{Standort};{Lager})
inproduktion	liefert die Menge von Artikeln, die produziert wird; 1. Parameter: die Artikelnummer 2. Parameter: der Standort als Ausdruck oder Datenbankfeld, der 2. Parameter kann weggelassen werden	inproduktion({Artikelnummer};{Standort})
reserviert	liefert die aktuell reservierte Menge von Artikeln; 1. Parameter: die Artikelnummer 2. Parameter: der Standort als Ausdruck oder Datenbankfeld, der 2. Parameter kann weggelassen werden	reserviert({Artikelnummer};{Standort})
kalkulationspreis	liefert den aktuellen Kalkulationspreis von Artikeln; als Parameter ist die Artikelnummer als Ausdruck oder Datenbankfeld anzugeben.	kalkulationspreis({Artikelnummer})

Funktion	Beschreibung	Beispiel
listenpreis	liefert den aktuellen Listenpreis von Artikeln; als Parameter ist eine Artikelnummer anzugeben	listenpreis({Artikelnummer})
belegtyp	liefert die Beschreibung des Belegtyps; als Parameter ist der Belegtyp anzugeben.	belegtyp({.Belegtyp}) // liefert z.B. Rechnung
belegadresskuerzel	liefert das Adresskürzel des Belegtyps (bspw. "KU" bei Kundenbelegen); als Parameter ist der Belegtyp anzugeben.	belegadresskuerzel({.Belegtyp}) // liefert z.B. KU
belegeafaktor	liefert die Information zur Interpretation von RNetto (bspw. bei Rechnungen +1, bei Gutschriften -1); als Parameter ist ein Belegtyp anzugeben	belegeafaktor({.Belegtyp})
mwst	liefert die Steuerprocente eines Steuerschlüssels zum aktuellen Datum; als Parameter ist ein Steuerschlüssel anzugeben	mwst(„3“) mwst({SteuerCode}) // liefern 19
checksumpost	Berechnet eine spezielle Prüfziffer; der Parameter muss eine Zeichenkette mit 11 Ziffern liefern	

Funktion	Beschreibung	Beispiel
shellexecute	zum Absetzen von Kommandos für das Betriebssystem; 1. Parameter (String): Name des Programms, das geöffnet werden soll 2. Parameter (String): er enthält die Kommandozeilenparameter für das Programm. 3. Parameter (Typ Zahl zwischen 0 und 10): Er ist dafür verantwortlich, in welchem Zustand das Fenster des gerufenen Programms geöffnet werden soll. Als Ergebnis liefert die Funktion eine Zahl, die bei Zahlen kleiner 32 eine Fehlernummer darstellt, der 2. und 3. Parameter können weggelassen werden.	<code>shellexecute("notepad";"c:\test.txt";3)</code> //öffnet die Datei c:\test.txt im maximierten NotePad. <code>[D:0:{System.Druckziel}:<>:0]</code> <code>[V:0:::0:0::shellexecute("AcroRd32.exe";" /n /t C:\Tmp\test.pdf")]</code> Sorgt dafür, dass eine externe PDF gedruckt wird, sobald ein Beleg gedruckt wird. Die Bedingungen sorgt einfach dafür, dass es beim Bildschirmdruck nicht passiert.
projektsumme	liefert in Abhängigkeit vom Preistyp des Projektes die Netto- oder Bruttoprojektsumme; als Parameter ist die Nummer des Projektes anzugeben.	<code>projektsumme({Projekt})</code> // liefert Netto- oder Bruttoprojektsumme
datenpfad	liefert den Datenpfad mit abschliessenden Backslash	
mandantenpfad	liefert den Mandantenpfad mit abschliessende Backslash	

4.5 Drucken von html-formatierten Text

Die HTML-Formatierungsmöglichkeiten sind bereits seit der Version 11.0.0 nutzbar. Mit der SelectLine Version 11.0.2 Hotfix 4 wurde erstmals die Möglichkeit bereitgestellt, bestimmte Langtexte in SelectLine Auftrag über den Formulareditor zu drucken.

Dazu werden spezielle Platzhalter verwendet, welche neu in die entsprechenden Druckvorlagen eingefügt werden müssen. Formatierter Text wird nicht automatisch formatiert ausgedruckt.

Welche Platzhalter stehen neu **im Artikelstamm** zur Verfügung?

- HTMLLangtext
- HTMLDimensionstext
- HTMLBestelltext
- HTMLInfotext

In der Druckvorlage werden diese Platzhalter über „Platzhalter Datenfeld“ auf der Seite Artikel angeboten. Verwendet werden diese Platzhalter dann direkt im Hauptblock oder in Unterblöcken. Jeder Platzhalter druckt den kompletten mehrzeiligen Langtext aus. Es ist deshalb nicht notwendig, dafür einen eigenen Unterblock zu erzeugen.



Bitte beachten Sie:

Nur wenn diese neuen Platzhalter verwendet werden, wird auch formatierter Text gedruckt!

Die neuen Druckplatzhalter haben eingeschränkte Parameter.
[H:0:0:HTMLLangtext]

- der Feldbezeichner steht mit Auswahl des Feldes fest, kein Wechsel vorgesehen
- es kann eine horizontale Positionierung vorgenommen werden (Abstand vom linken Rand)
- die Breite kann eingeschränkt werden (Breite)
- es gibt keine Formatauswahl, da alles bereits im HTML-Text formatiert ist
- Der HTML-formatierte Lang- oder Bestelltext wird aus den Artikelstammdaten mit der Artikelauswahl in die Belegposition übernommen oder kann dort neu eingegeben werden. Letzteres trifft z.B. bei Kommentaren zu.
- Welche Platzhalter stehen neu im Beleg zur Verfügung?
- HTMLLangtext der HTML-Positionslangtext
- ArtikelHTMLLangtext der HTML-Langtext aus dem Artikelstamm
- ArtikelHTMLDimensionstext der HTML-Dimensionstext aus dem Artikelstamm
- ArtikelHTMLBestelltext der HTML-Bestelltext aus dem Artikelstamm
- ArtikelHTMLInfotext der HTML-Infotext aus dem Artikelstamm

Die neuen Platzhalter stehen nur in den Druckvorlagen des Artikelstamms oder der Belege zur Verfügung. In Druckvorlagen eines Makros oder anderen Stammdaten kann man mittels **Verweisen** an die HTML-Texte gelangen.

5 Anhang

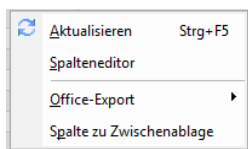
5.1 Glossar

Alphanumerik:

Es ist wichtig, dass Sie sich bei der Erfassung von Stammdaten oder Belegen mit der Alphanumerik auseinandersetzen. Dies kann sich auf die Sortierung, Darstellung und Auswertung der Daten weiterführend auswirken.

Machen Sie sich Gedanken über die ungefähre Anzahl an Stammdaten und Belegen, die als Anzahl Stellen (inkl. führenden Nullen) definiert werden. Dies bedeutet konkret, wenn Sie etwa 1'000 Kunden haben, beginnen Sie mit der Kundennummer 1001 oder 0001. Die Daten werden ansonsten immer nach der vordersten Zahl gegliedert, wie z. B. folgendermassen:
1,10,11,...,100,101,...,2,20,21,...,etc.

Spalteneditor:



In allen Tabellenansichten haben Sie die Möglichkeit, diese auf Ihre individuellen Bedürfnisse anzupassen. Dies ist auf verschiedene Arten möglich: einerseits kommen Sie in den „Spalteneditor“ indem Sie in der Tabellenansicht in der Tabelle über das Kontextmenü der rechten Maustaste klicken und anschliessend die Spaltenüberschriften mit der linken Maustaste verschieben.

Andererseits können Sie auch in der Tabelle selber die Spalten mit der rechten Maustaste, in der Kopfzeile, an die gewünschte Position verschieben.

Quickfilter:



Den Quickfilter finden Sie in den meisten Fenstern des Programmes. Durch diesen ist es möglich, im geöffneten Fenster nach einem gewünschten Datensatz zu suchen. Es kann in allen Feldern oder nur in einer gewünschten Spalte gesucht werden. Der Kreis ganz rechts ändert die Farbe von **blau** zu **rot** wenn er aktiviert ist. Sie sehen dann nur die Auswahl gemäss Ihren Suchkriterien.

Icons:

	Wechseln zum ersten Eintrag der Datenbank		Löschen eines Datensatzes
	Einen Datensatz Rückwärts blättern		Wechseln in die Tabellenansicht
	Wechseln zum letzten Eintrag der Datenbank		Wechseln in den Bearbeitungsmodus
	Einen Datensatz Vorwärts blättern		Marke setzen, ▽ = zum markierten Artikel
	Historie-Button (die letzten 10 Datensätze)		Filter setzen, ▽ = neuen Filter anlegen
	Neuen Datensatz erstellen		Einstellungen und Zusatzfunktionen
	Kopie eines bestehenden Datensatzes anlegen		Einzeldruck, Listendruck
	Speichern		Eingabemaske schliessen und speichern
	Verwerfen einer Eingabe		

5.2 Dank

Vielen Dank für Ihre Aufmerksamkeit und herzliche Gratulation zur erfolgreichen Teilnahme an diesem Kurs. Wir wünschen Ihnen viel Spass und Erfolg beim Umsetzen in Ihrem Geschäftsalltag. Wenn nur einige Punkte dabei waren, die Sie für sich mitnehmen und anwenden können und sich damit Ihr Alltag etwas vereinfacht, ist dies schon einiges an Profit, den Sie gewonnen haben. Denn Zeit ist und bleibt eine der knappsten Ressourcen, die wir haben und diese gilt es, möglichst effizient einzusetzen.

Um diese erworbenen Kompetenzen erweitern und ausbauen zu können empfehlen wir Ihnen, die Erkenntnisse in Ihrem täglichen Arbeiten mit SelectLine Produkten einzusetzen und Ihre Fähigkeiten zu erweitern und aufzufrischen. Deshalb freuen wir uns schon jetzt, Sie bei einem weiteren Kurs wieder bei uns zu begrüßen. Als Beilage finden Sie ein Formular mit den nächsten Kursen und Terminen, womit Sie sich anmelden können. Die Anmeldung ist aber auch weiterhin über unsere Website www.selectline.ch unter „Unterstützung/Schulungen“ möglich.

Freundliche Grüsse

SelectLine Software AG

5.3 Ihre Notizen und Erkenntnisse