

END · SUB ¶

Es wird über den Controller auf die Ansicht des Formulars zugegriffen. Damit wird auf der sichtbaren Oberfläche tatsächlich erscheint. Der ausgewählte Wert ist:

Listenfelder durch Eingabe von Anfangsbuchstaben eins

Manchmal kann es vorkommen, dass der Inhalt für Listenfelder unübersichtlich eine Suche schneller zum Erfolg führt, wäre es sinnvoll, hier den Inhalt des Listgabe eines oder mehrerer Buchstaben einzugrenzen. Das Listenfeld selbst wird einem SQL-Befehl versehen, der nur als Platzhalter dient. Hier könnte z.B. sein:

```
SELECT "Name", "ID" FROM "Tabelle" ORDER BY "Name" LIMIT
```

So wird beim Öffnen des Formulars vermieden, dass Base erst einmal die umf lesen muss. ¶

Das folgende Makro ist dafür an **Eigenschaften: Listenfeld → Ereignisse → Taste lo**

```
GLOBAL stListStart AS STRING ¶
GLOBAL lZeit AS LONG ¶
```

Zuerst werden globale Variablen erstellt. Diese Variablen sind notwendig, damit einem Buchstaben, sondern nach dem Betätigen weiterer Tasten schließlich an Buchstabenkombination gesucht werden kann. ¶

In der globalen Variablen **stListStart** werden die Buchstaben in der eingeg gespeichert. ¶

Formatvorlagen

- LibOListe 1 Start
- LibONum 123 Ende
- LibONum 123 Inhalt
- LibONum 123 Start
- LibOSeitenUmbruch
- LibOTabelleBeschriftung
- LibOTabelleText
- LibOTabelleÜberschrift**
- LibOTextkörper_ListenIntro
- LibOTextkörperEin Erstzeile

Format of a paragraph
 Format isn't shown if the text is defined as "black".
 Format is shown, if the text is defined in any other color.

Listenfelder durch Eingabe von Anfangsbuchstaben eins

Manchmal kann es vorkommen, dass der Inhalt für Listenfelder unübersichtlich eine Suche schneller zum Erfolg führt, wäre es sinnvoll, hier den Inhalt des Listgabe eines oder mehrerer Buchstaben einzugrenzen. Das Listenfeld selbst wird einem SQL-Befehl versehen, der nur als Platzhalter dient. Hier könnte z.B. sein:

```
SELECT "Name", "ID" FROM "Tabelle" ORDER BY "Name" LIMIT
```

So wird beim Öffnen des Formulars vermieden, dass Base erst einmal die umf lesen muss. ¶

Das folgende Makro ist dafür an **Eigenschaften: Listenfeld → Ereignisse → Taste lo**

```
GLOBAL stListStart AS STRING ¶
GLOBAL lZeit AS LONG ¶
```

Zuerst werden globale Variablen erstellt. Diese Variablen sind notwendig, damit einem Buchstaben, sondern nach dem Betätigen weiterer Tasten schließlich an Buchstabenkombination gesucht werden kann. ¶

In der globalen Variablen **stListStart** werden die Buchstaben in der eingeg gespeichert. ¶

Die globale Variable **lZeit** wird mit der aktuellen Zeit in Sekunden versorgt. Eine Pause zwischen den Tastatureingaben soll die Variable **stListStart** wieder den können. Deswegen wird jeweils der Zeitunterschied zur vorhergehenden E

```
SUB ListFilter (oEvent AS OBJECT) ¶
  → oFeld := oEvent.Source.Model ¶
  → IF oEvent.KeyCode < 538 OR oEvent.KeyCode = 1283 OR oEvent.KeyCo
```

Das Makro wird durch einen Tastendruck ausgelöst. Eine Taste hat innerhalb d bestimmten Zahlencode, der unter [com>sun>star>awt>Key](#) nachgeschlagen w derzeichen wie das «ä», «ö» und «ü» haben den **KeyCode** 0, alle anderen ScI Zahlen haben einen **KeyCode** kleiner als 538. Den **KeyCode** 1283 beeat die f

Formatvorlagen

- LibOListe 1 Start
- LibONum 123 Ende
- LibONum 123 Inhalt
- LibONum 123 Start
- LibOSeitenUmbruch
- LibOTabelleBeschriftung
- LibOTabelleText
- LibOTabelleÜberschrift**
- LibOTextkörper
- LibOTextkörper_ListenIntro
- LibOTextkörperEin Erstzeile
- LibOTipp/Hinweis/Warnun
- LibOÜberschrift**
- LibOÜberschl**
- LibOÜberschrift 1**
- LibOÜberschrift 2**
- LibOÜberschrift 3**

The screenshot shows a code editor on the left and a 'Formatvorlagen' (Format Templates) panel on the right. The code editor contains a macro for filtering list items based on their first character. The 'Formatvorlagen' panel lists various templates, with 'MakroReservierterAusdruck' highlighted in blue.

```

→ oForm := oDoc.Drawpage.Forms(0)
→ oListBox := oForm.getByname("Listenfeld")
→ oController := oDoc.GetCurrentController()
→ oView := oController.getControl(oListBox)
→ print."Angezeigter Inhalt: " & oView.SelectedItem
END-SUB

```

Es wird über den Controller auf die Ansicht des Formulars zugegriffen. Damit wird auf der sichtbaren Oberfläche tatsächlich erscheinend. Der ausgewählte Wert ist:

Listenfelder durch Eingabe von Anfangsbuchstaben einschränken

Manchmal kann es vorkommen, dass der Inhalt für Listenfelder unübersichtlich ist. Eine Suche schneller zum Erfolg führt, wäre es sinnvoll, hier den Inhalt des Listenfeldes selbst mit einem SQL-Befehl versehen, der nur als Platzhalter dient. Hier könnte z.B. stehen:

```
SELECT "Name", "ID" FROM "Tabelle" ORDER BY "Name" LIMIT 10
```

So wird beim Öffnen des Formulars vermieden, dass Base erst einmal die umfassen muss.

Das folgende Makro ist dafür an **Eigenschaften: Listenfeld → Ereignisse → Taste** zugeordnet:

```

GLOBAL stListStart AS STRING
GLOBAL lZeit AS LONG

```

Zuerst werden globale Variablen erstellt. Diese Variablen sind notwendig, damit ein Buchstabe, sondern nach dem Betätigen weiterer Tasten schließlich an Buchstabenkombination gesucht werden kann.

In der globalen Variablen **stListStart** werden die Buchstaben in der eingetragten gespeichert.

Die globale Variable **lZeit** wird mit der aktuellen Zeit in Sekunden versorgt. Eine Pause zwischen den Tastatureingaben soll die Variable **stListStart** wieder neu setzen können. Deswegen wird jeweils der Zeitunterschied zur vorhergehenden Zeit gespeichert.

```

SUB ListFilter(oEvent AS OBJECT)
→ oFeld := oEvent.Source.Model
→ IF oEvent.KeyCode < 538 OR oEvent.KeyCode = 1283 OR oEvent.KeyCode

```

The 'Formatvorlagen' panel on the right contains the following items:

- LibOButton
- LibOComputerCode
- LibOLinkEinzeldokument
- LibOLinkGlobaldokument
- LibOMenüPfad
- LibOStandard
- LibOStarkeBetonung
- LibOTaste
- MakroBezeichner
- MakroKommentar
- MakroOperator
- MakroReservierterAusdruck** (highlighted in blue)
- MakroZahl
- MakroZeichenkette
- Quelltext
- Standard

Format for Chars:
 Format is shown, if the marked char is defined in a special color like blue here.

This screenshot shows a similar setup to the first one, but with a different format template selected. The 'MakroZeichenkette' template is highlighted in black, indicating that no special color formatting is applied to the characters.

```

SUB ListFilter(oEvent AS OBJECT)
→ oFeld := oEvent.Source.Model
→ IF oEvent.KeyCode < 538 OR oEvent.KeyCode = 1283 OR oEvent.KeyCode

```

The 'Formatvorlagen' panel on the right shows:

- MakroZeichenkette (highlighted in black)
- Quelltext

Format for Chars:
 Format is not shown, if the marked char is defined as "black" (Standard)