

15.1.2 Methode SendKeys – Klasse Desktop (gb.desktop)

Um einen einzelnen Tastendruck oder eine Kombination von Tasten zu generieren und an das Fenster mit dem Fokus zu senden, können Sie die Methode `SendKeys(..)` einsetzen.

Syntax:

```
Static Sub SendKeys ( Keys As String )
```

Keys ist dabei eine Liste von Tasten die nacheinander gesendet werden. Die (formalen) Namen der Tasten sind in der X11-Header-Datei `/usr/include/X11/keysymdef.h` aufgelistet. Ausgewählte Tastennamen auf dem Ziffernblock (Keypad) ermitteln Sie zum Beispiel mit diesem Befehl in einer Konsole:

```
hans@linux:~$ cat /usr/include/X11/keysymdef.h | grep 'KP_E'
#define XK_KP_Enter          0xff8d /* Enter */
#define XK_KP_End           0xff9c
#define XK_KP_Equal         0xffbd /* Equals */
hans@linux:~$
```

Um einen Tastennamen zu extrahieren, müssen Sie den `XK_`-Präfix aus der Konstanten in der Datei `/usr/include/X11/keysymdef.h` entfernen. Die Taste mit der Bezeichnung `[1/End]` im Ziffernblock hat den Tastennamen `KP_End` – aber nur dann, wenn der Ziffernblock mit der Taste `NumLock` de-aktiviert wurde – sonst wird die Ziffer 1 wirksam.

Leider erkennen Sie nicht, dass bei den Tastennamen zwischen Groß- und Kleinschreibung unterschieden wird und das nicht immer konsistent ist. Zum Beispiel beginnt 'Return' mit einem großen R, aber die Leertaste mit dem Tastennamen 'space' mit kleinem s. In der Datei `/usr/include/X11/keysymdef.h` finden Sie den Namen `XK_KP_Space`, während das Tastenanalyse-Programm `xev` die gedrückte Leertaste als 'space' beschreibt:

```
KeyRelease event, serial 33, synthetic NO, window 0x3400001,
  root 0xbc, subw 0x3400002, time 15995041, (33,37), root:(149,566),
  state 0x0, keycode 65 (keysym 0x20, space), same_screen YES,
  XLookupString gives 1 bytes: (20) " "
  XFilterEvent returns: False
```

Das i-Tüpfelchen setzt eine Bemerkung von B. Minisini zu den Änderungen bei den Tastenkombinationen: "X11 fand es toll, vor kurzem die Bedeutung einzelner Tasten zu ändern. Bisher wurden Großbuchstaben benutzt, jetzt benutzen sie eine Mischung aus Groß- und Kleinbuchstaben. Sehr lustig."

Aus diesem Grunde funktioniert zur Zeit nur `Desktop.SendKeys("{[Control_L]c}")` mit kleinem c korrekt.

Bedenken Sie die Unterschiede der Bezeichnungen für die einzelnen Tasten, wie sie

- in der X11-Header-Datei `/usr/include/X11/keysymdef.h` stehen,
- vom Tastenanalyse-Programm `xev` angezeigt werden,
- auf den realen Tasten auf der Tastatur stehen und
- über Tastatur-Konstanten in Gambas gekennzeichnet werden.

15.1.2.1 Modifizierer

Um zum Beispiel auf einer deutschen Tastatur im alpha-numerischen Block das *Fragezeichen* oder den *Backslash* zu erreichen, müssen Sie in die 2. oder 3. Belegung der *ß-Taste* umschalten. Der Umschalter oder *Modifizierer* für die 2. Ebene ist die Shift-Taste (`→ ?`) und mit der Taste `AltGr` (alternativer Grafik-Zeichensatz; Tastenname: `ISO_Level3_Shift`) kommen Sie in die 3. Ebene und erreichen den *Backslash* (`→ \`). Die zweite Belegung der Druck-Taste erreichen Sie zum Beispiel im Normalfall mit dem Modifizierer `Alt`-Taste.

15.1.2.2 Tastendruck generieren

Die Syntax ist recht einfach, doch es sind noch einige Bemerkungen notwendig:

- Wenn mehrere Tasten gleichzeitig gedrückt werden müssen, müssen sie zwischen '{' und '}' ein-

geschlossen werden. Um das ASCII-Zeichen '{' zu senden, müssen Sie "{}" benutzen.

- Was gesendet wird ist keine Zeichenkette (String), sondern die internen Tasten-Codes. Die erforderlichen Modifikatoren werden automatisch vor dem Senden eines (ASCII-)Zeichens gesendet – wie zum Beispiel bei Desktop.SendKeys("M").
- Jeder Tastenname muss zwischen '[' und ']' eingeschlossen werden. Aber ein beliebiges ASCII-Zeichen können Sie so senden, wie es ist. Sie können etwa das '\n'-Zeichen für die Return-Taste oder das '\t'-Zeichen für die Tab-Taste senden.

Beispiele:

```
Desktop.SendKeys("A") ↔ Desktop.SendKeys("[Shift_L]a") ↔ Desktop.SendKeys("[Shift_R]a")
Desktop.SendKeys("\n") ↔ Desktop.SendKeys("[Return]")
Desktop.SendKeys("[Control_L]c")
Desktop.SendKeys("Gambas Almost Means BASic")
Desktop.SendKeys(TextArea.Text)
Desktop.SendKeys("[Control_L][Alt_L][Print]") ' Bildschirmkopie
Desktop.SendKeys("[Alt_L][Print]") ' Bildschirmkopie des aktiven Fensters
Desktop.SendKeys("[NumLock]")
Desktop.SendKeys("[F1]")
```