

# japi

Reference Manual

Merten Joost



# Inhaltsverzeichnis

<b>I</b>	<b>Reference</b>	<b>5</b>
<b>1</b>	<b>Components</b>	<b>7</b>
	Button . . . . .	7
	Borderpanel . . . . .	11
	Canvas . . . . .	16
	Checkbox . . . . .	21
	CheckmenuItem . . . . .	24
	Choice . . . . .	25
	Dialog . . . . .	28
	Focuslistener . . . . .	34
	Frame . . . . .	35
	Helpmenu . . . . .	41
	Hscrollbar . . . . .	43
	Graphicbutton . . . . .	46
	Graphiclabel . . . . .	49
	Image . . . . .	52
	KeyListener . . . . .	55
	Label . . . . .	56
	Led . . . . .	59
	List . . . . .	62
	Menu . . . . .	66
	MenuItem . . . . .	68
	Meter . . . . .	69
	Mouselistener . . . . .	72
	Panel . . . . .	73
	PopupMenu . . . . .	78
	Printer . . . . .	80
	Progressbar . . . . .	83
	Radiobutton . . . . .	86
	Sevensegment . . . . .	89
	ScrollPane . . . . .	92
	Textarea . . . . .	95
	Textfield . . . . .	99
	Vscrollbar . . . . .	103
	Window . . . . .	106

<b>2 Functions</b>	<b>111</b>
additem . . . . .	111
add . . . . .	113
alertbox . . . . .	114
appendtext . . . . .	115
beep . . . . .	116
borderpanel . . . . .	117
button . . . . .	118
canvas . . . . .	119
checkbox . . . . .	120
checkmenuItem . . . . .	121
choicebox2 . . . . .	122
choicebox3 . . . . .	123
choice . . . . .	124
cliprect . . . . .	125
componentlistener . . . . .	126
connect . . . . .	127
delete . . . . .	128
deselect . . . . .	129
dialog . . . . .	130
disable . . . . .	131
dispose . . . . .	132
drawarc . . . . .	133
drawcircle . . . . .	134
drawimagesource . . . . .	135
drawimage . . . . .	136
drawline . . . . .	137
drawoval . . . . .	138
drawpixel . . . . .	139
drawpolygon . . . . .	140
drawpolyline . . . . .	141
drawrect . . . . .	142
drawroundrect . . . . .	143
drawscaledimage . . . . .	144
drawstring . . . . .	145
enable . . . . .	146
filedialog . . . . .	147
fileselect . . . . .	148
fillarc . . . . .	149
fillcircle . . . . .	150
filloval . . . . .	151
fillpolygon . . . . .	152
fillrect . . . . .	153
fillroundrect . . . . .	154
focuslistener . . . . .	155
frame . . . . .	156
getaction . . . . .	157

getcolumns . . . . .	158
getcurpos . . . . .	159
getdanger . . . . .	160
getfontascent . . . . .	161
getfontheight . . . . .	162
getheight . . . . .	163
getimagesource . . . . .	164
getimage . . . . .	165
getinheight . . . . .	166
getinsets . . . . .	167
getinwidth . . . . .	168
getitemcount . . . . .	169
getitem . . . . .	170
getkeychar . . . . .	171
getkeycode . . . . .	172
getlayoutid . . . . .	173
getlength . . . . .	174
getmousebutton . . . . .	175
getmousex . . . . .	176
getmousey . . . . .	177
getparentid . . . . .	178
getparent . . . . .	179
getrows . . . . .	180
getscaledimage . . . . .	181
getscreenheight . . . . .	182
getscreenwidth . . . . .	183
getselect . . . . .	184
getselend . . . . .	185
getselstart . . . . .	186
getseltext . . . . .	187
getstate . . . . .	188
getstringwidth . . . . .	189
gettext . . . . .	190
getvalue . . . . .	191
getviewportheight . . . . .	192
getviewportwidth . . . . .	193
getwidth . . . . .	194
getxpos . . . . .	195
getypos . . . . .	196
graphicbutton . . . . .	197
graphiclabel . . . . .	198
hasfocus . . . . .	199
helpmenu . . . . .	200
hide . . . . .	201
hscrollbar . . . . .	202
image . . . . .	203
insert . . . . .	204

inserttext . . . . .	205
isparent . . . . .	206
isresizable . . . . .	207
isselect . . . . .	208
isvisible . . . . .	209
keylistener . . . . .	210
label . . . . .	211
led . . . . .	212
line . . . . .	213
list . . . . .	214
loadimage . . . . .	215
menubar . . . . .	216
menuItem . . . . .	217
menu . . . . .	218
messagebox . . . . .	219
meter . . . . .	220
mouselistener . . . . .	221
multiplemode . . . . .	222
nextaction . . . . .	223
pack . . . . .	224
panel . . . . .	225
popupmenu . . . . .	226
printer . . . . .	227
print . . . . .	228
progressbar . . . . .	229
quit . . . . .	230
radiobutton . . . . .	231
radiogroup . . . . .	232
random . . . . .	233
releaseall . . . . .	234
release . . . . .	235
removeall . . . . .	236
removeitem . . . . .	237
remove . . . . .	238
replacetext . . . . .	239
saveimage . . . . .	240
scrollpane . . . . .	241
selectall . . . . .	242
select . . . . .	243
selecttext . . . . .	244
separator . . . . .	245
setalign . . . . .	246
setblockinc . . . . .	247
setborderlayout . . . . .	248
setborderpos . . . . .	249
setcolorbg . . . . .	250
setcolor . . . . .	251

setcolumns	252
setcurpos	253
setcursor	254
setdanger	255
setdebug	256
setechochar	257
seteditable	258
setfixlayout	259
setflowfill	260
setflowlayout	261
setfocus	262
setfontname	263
setfontsize	264
setfontstyle	265
setFont	266
setgridlayout	268
sethgap	269
setIcon	270
setImage	271
setinsets	272
setMax	273
setmin	274
setnamedcolorbg	275
setnamedcolor	276
setnolayout	277
start	278
setpos	279
setradiogroup	280
setresizable	281
setrows	282
setshortcut	283
setsize	284
setslidesize	285
setstate	286
setText	287
setunitinc	288
setValue	289
setvgap	290
setxor	291
sevensegment	292
showpopup	293
show	294
sleep	295
start	296
sync	297
textarea	298
textfield	299

translate . . . . .	300
vscrollbar . . . . .	301
windowlistener . . . . .	302
window . . . . .	303



**Teil I**

**Reference**



# Kapitel 1

## Components

Button

- j\_button**      *function j\_button ( obj , label\$ )*  
Erzeugt einen neuen Button mit dem Textinhalt **label**
- j\_add**      *procedure j\_add ( obj , cont )*  
Fügt den Button **obj** in den Container **cont** ein.
- j\_componentlistener** *function j\_componentlistener ( obj , kind )*  
Bindet an den Button **obj** einen neuen Componentlistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.
- j\_disable**      *procedure j\_disable ( obj )*  
disabled den Button **obj**, sodaß diese auf keine Benutzereingaben mehr reagiert.
- j\_dispose**      *procedure j\_dispose ( obj )*  
der Button **obj** wird mit allen Recourcen gelöscht.
- j\_enable**      *procedure j\_enable ( obj )*  
enabled den Button **obj**.
- j\_focuslistener**      *function j\_focuslistener ( obj )*  
Bindet an den Button **obj** einen neuen Focuslistener, und liefert dessen Eventnummer zurück.
- j\_getfontascent**      *function j\_getfontascent ( obj )*  
Liefert die Oberlänge (Ascent) des aktuellen Fonts des Button **obj** in Bildschirmpixeln zurück.
- j\_getfontheight**      *function j\_getfontheight ( obj )*

	Liefert die totale Höhe des aktuellen Fonts des Button <b>obj</b> in Bildschirmpixeln zurück.
<b>j_getheight</b>	<i>function j_getheight ( obj )</i> Liefert die Höhe des Button <b>obj</b> zurück.
<b>j_getlength</b>	<i>function j_getlength ( obj )</i> Liefert die Länge des Textinhaltes den Button zurück.
<b>j_getparentid</b>	<i>function j_getparentid ( obj )</i> Liefert die Eventnummer des umgebenen Containers von <b>obj</b> zurück. Ist <b>obj</b> ein Frame wird $-1$ zurückgegeben.
<b>j_getparent</b>	<i>function j_getparent ( obj )</i> Liefert die Eventnummer des umgebenen Containers von <b>obj</b> zurück. Ist <b>obj</b> ein Frame wird $-1$ zurückgegeben.
<b>j_getstringwidth</b>	<i>function j_getstringwidth ( obj , str\$ )</i> Liefert die Länge des <b>str</b> des aktuellen Fonts des Button <b>obj</b> in Bildschirmpixeln zurück.
<b>j_gettext</b>	<i>function\$ j_gettext ( obj )</i> Liefert den Textinhalt des Button .
<b>j_getwidth</b>	<i>function j_getwidth ( obj )</i> Liefert die Breite des Button <b>obj</b> zurück.
<b>j_getxpos</b>	<i>function j_getxpos ( obj )</i> Liefert die aktuelle horizontale Position des Button <b>obj</b> zurück.
<b>j_getypos</b>	<i>function j_getypos ( obj )</i> Liefert die aktuelle vertikale Position des Button <b>obj</b> zurück.
<b>j_hide</b>	<i>procedure j_hide ( obj )</i> Zeigt den Button <b>obj</b> nicht mehr auf dem Bildschirm an.
<b>j_isparent</b>	<i>function j_isparent ( obj , cont )</i> Liefert J_TRUE zurück, falls das <b>cont</b> die umgebende Komponente von <b>obj</b> ist, sonst J_FALSE
<b>j_isvisible</b>	<i>function j_isvisible ( obj )</i> Liefert J_TRUE zurück, falls <b>obj</b> sichtbar ist, sonst J_FALSE
<b>j_keylistener</b>	<i>function j_keylistener ( obj )</i> Bindet an den Button <b>obj</b> einen neuen Keylistener, und liefert dessen Eventnummer zurück.
<b>j_mouselistener</b>	<i>function j_mouselistener ( obj , kind )</i> Bindet an den Button <b>obj</b> einen neuen Mouselistener, und liefert dessen Eventnummer zurück. Der Parameter <b>kind</b> bestimmt, wann ein Event ausgelöst wird.
<b>j_popupmenu</b>	<i>function j_popupmenu ( obj , label\$ )</i>

	Erzeugt ein neues Popumenu mit dem Inhalt <b>label</b> und liefert dessen Eventnummer zurück.
<b>j_print</b>	<i>procedure j_print ( obj )</i> Druckt den Inhalt des Button .
<b>j_release</b>	<i>procedure j_release ( obj )</i> Gibt den Button <b>obj</b> aus dem umgebenden Container Element frei.
<b>j_setborderpos</b>	<i>procedure j_setborderpos ( obj , pos )</i> Positioniert den Button <b>obj</b> an die angegebene Position <b>pos</b> . Im umgebenden Container muß einen BorderLayout Manager eingestellt sein.
<b>j_setcolorbg</b>	<i>procedure j_setcolorbg ( obj , r , g , b )</i> Setzt die Hintergrundfarbe auf den übergebenen RGB Wert
<b>j_setcolor</b>	<i>procedure j_setcolor ( obj , r , g , b )</i> Setzt die Vordergrundfarbe auf den übergebenen RGB Wert
<b>j_setcursor</b>	<i>function j_setcursor ( obj , cursor )</i> Setzt den Cursor des Button <b>obj</b> auf <b>cursor</b> .
<b>j_setfocus</b>	<i>function j_setfocus ( obj )</i> Weist den Input Focus dem Button <b>obj</b> zu.
<b>j_setfontname</b>	<i>procedure j_setfontname ( obj , name )</i> Setzt den Fonttyp des Button <b>obj</b> auf <b>name</b> .
<b>j_setfont</b>	<i>procedure j_setfont ( obj , name , style , size )</i> Setzt den Font des Button <b>obj</b> entsprechend der Parameterliste.
<b>j_setfontsize</b>	<i>procedure j_setfontsize ( obj , size )</i> Setzt den Fontgröße des Button <b>obj</b> auf <b>size</b> .
<b>j_setfontstyle</b>	<i>procedure j_setfontstyle ( obj , style )</i> Setzt den Fontstyle des Button <b>obj</b> auf <b>style</b> .
<b>j_setnamedcolorbg</b>	<i>procedure j_setnamedcolorbg ( obj , color )</i> Setzt die Hintergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnamedcolor</b>	<i>procedure j_setnamedcolor ( obj , color )</i> Setzt die Vordergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setpos</b>	<i>procedure j_setpos ( obj , xpos , ypos )</i> Verschiebt den Button <b>obj</b> zur angegebenen Koordinate ( <b>xpos,ypos</b> ) an.
<b>j_setsize</b>	<i>procedure j_setsize ( obj , width , height )</i> Setzt Breite und Höhe des Button <b>obj</b> auf <b>width</b> and <b>height</b> .
<b>j_settext</b>	<i>procedure j_settext ( obj , str\$ )</i> Ändert den Textinhalt des Button <b>obj</b> auf <b>str</b>

**j\_show***procedure j\_show ( obj )*Zeigt den Button **obj** auf dem Bildschirm an.

Borderpanel
-------------

- j\_borderpanel**      *function j\_borderpanel ( obj , type )*  
Erzeugt ein neues Borderpanel und liefert dessen Eventnummer zurück.
- j\_add**              *procedure j\_add ( obj , cont )*  
Fügt das Borderpanel **obj** in den Container **cont** ein.
- j\_borderpanel**      *function j\_borderpanel ( obj , type )*  
Erzeugt ein neues Borderpanel und liefert dessen Eventnummer zurück.
- j\_button**            *function j\_button ( obj , label\$ )*  
Erzeugt einen neuen Button mit dem Textinhalt **label**
- j\_canvas**            *function j\_canvas ( obj , width , height )*  
Erzeugt einen neuen Canvas mit der Breite **width** Pixel und einer Höhe von **height** Pixel.
- j\_checkbox**         *function j\_checkbox ( obj , label\$ )*  
Erzeugt eine Checkbox mit dem Textinhalt **label** und liefert dessen Eventnummer zurück.
- j\_choice**            *function j\_choice ( obj )*  
Erzeugt eine neue Choice und gibt deren Eventnummer zurück.
- j\_componentlistener** *function j\_componentlistener ( obj , kind )*  
Bindet an das Borderpanel **obj** einen neuen Componentlistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.
- j\_disable**            *procedure j\_disable ( obj )*  
disabled das Borderpanel **obj**, sodaß diese auf keine Benutzereingaben mehr reagiert.
- j\_dispose**           *procedure j\_dispose ( obj )*  
das Borderpanel **obj** wird mit allen Recourcen gelöscht.
- j\_enable**            *procedure j\_enable ( obj )*  
enabled das Borderpanel **obj**.
- j\_focuslistener**    *function j\_focuslistener ( obj )*  
Bindet an das Borderpanel **obj** einen neuen Focuslistener, und liefert dessen Eventnummer zurück.
- j\_getfontascent**    *function j\_getfontascent ( obj )*  
Liefert die Oberlänge (Ascent) des aktuellen Fonts des Borderpanel **obj** in Bildschirmpixeln zurück.

<b>j_getfontheight</b>	<i>function j_getfontheight ( obj )</i> Liefert die totale Höhe des aktuellen Fonts des Borderpanel <b>obj</b> in Bildschirm-pixeln zurück.
<b>j_getheight</b>	<i>function j_getheight ( obj )</i> Liefert die Höhe des Borderpanel <b>obj</b> zurück.
<b>j_getinheight</b>	<i>function j_getinheight ( cont )</i> Liefert die Höhe des Borderpanel <b>obj</b> des inneren Bereichs zurück.
<b>j_getinsets</b>	<i>function j_getinsets ( obj , side )</i> Liefert die angeforderte Breite des inneren Randes dem Borderpanel zurück.
<b>j_getinwidth</b>	<i>function j_getinwidth ( cont )</i> Liefert die Breite des Borderpanel <b>obj</b> des inneren Bereichs zurück.
<b>j_getlayoutid</b>	<i>function j_getlayoutid ( obj )</i> Liefert die Eventnummer des Layoutmanagers des Containers <b>obj</b> zurück.
<b>j_getparentid</b>	<i>function j_getparentid ( obj )</i> Liefert die Eventnummer des umgebenen Containers von <b>obj</b> zurück. Ist <b>obj</b> ein Frame wird $-1$ zurückgegeben.
<b>j_getparent</b>	<i>function j_getparent ( obj )</i> Liefert die Eventnummer des umgebenen Containers von <b>obj</b> zurück. Ist <b>obj</b> ein Frame wird $-1$ zurückgegeben.
<b>j_getstringwidth</b>	<i>function j_getstringwidth ( obj , str\$ )</i> Liefert die Länge des <b>str</b> des aktuellen Fonts des Borderpanel <b>obj</b> in Bild-schirmpixeln zurück.
<b>j_getwidth</b>	<i>function j_getwidth ( obj )</i> Liefert die Breite des Borderpanel <b>obj</b> zurück.
<b>j_getxpos</b>	<i>function j_getxpos ( obj )</i> Liefert die aktuelle horizontale Position des Borderpanel <b>obj</b> zurück.
<b>j_getypos</b>	<i>function j_getypos ( obj )</i> Liefert die aktuelle vertikale Position des Borderpanel <b>obj</b> zurück.
<b>j_graphicbutton</b>	<i>function j_graphicbutton ( obj , filename\$ )</i> Erzeugt einen neuen Graphicbutton mit dem Bildinhalt des Files, das durch <b>filename</b> spezifiziert wird.
<b>j_graphiclabel</b>	<i>function j_graphiclabel ( obj , str\$ )</i> Erzeugt ein neues Graphiclabel mit dem Bildinhalt des Files, das durch <b>filena-me</b> spezifiziert wird.
<b>j_hide</b>	<i>procedure j_hide ( obj )</i> Zeigt das Borderpanel <b>obj</b> nicht mehr auf dem Bildschirm an.
<b>j_hscrollbar</b>	<i>function j_hscrollbar ( obj )</i>



	Erzeugt eine horizontale Scrollbar und gibt dessen Eventnummer zurück.
<b>j_isparent</b>	<i>function j_isparent ( obj , cont )</i> Liefert J_TRUE zurück, falls das <b>cont</b> die umgebende Komponente von <b>obj</b> ist, sonst J_FALSE
<b>j_isvisible</b>	<i>function j_isvisible ( obj )</i> Liefert J_TRUE zurück, falls <b>obj</b> sichtbar ist, sonst J_FALSE
<b>j_keylistener</b>	<i>function j_keylistener ( obj )</i> Bindet an das Borderpanel <b>obj</b> einen neuen Keylistener, und liefert dessen Eventnummer zurück.
<b>j_label</b>	<i>function j_label ( obj , label\$ )</i> Erzeugt ein neues Label Element mit dem Textinhalt <b>label</b>
<b>j_led</b>	<i>function j_led ( obj , style , color )</i> Erzeugt eine LED der Farbe <b>color</b> und der Form <b>style</b>
<b>j_line</b>	<i>function j_line ( obj , orient , style , length )</i> Erzeugt ein neues Linielement und liefert dessen Eventnummer zurück
<b>j_list</b>	<i>function j_list ( obj , rows )</i> Erzeugt eine neue List und gibt deren Eventnummer zurück.
<b>j_meter</b>	<i>function j_meter ( obj , title\$ )</i> Erzeugt ein analoges Anzeigeelement mit der Beschriftung <b>title</b> .
<b>j_mouselistener</b>	<i>function j_mouselistener ( obj , kind )</i> Bindet an das Borderpanel <b>obj</b> einen neuen Mouselistener, und liefert dessen Eventnummer zurück. Der Parameter <b>kind</b> bestimmt, wann ein Event ausgelöst wird.
<b>j_pack</b>	<i>procedure j_pack ( obj )</i> Veranlaßt den Layoutmanager von <b>obj</b> die minimale Größe zu berechnen.
<b>j_panel</b>	<i>function j_panel ( obj )</i> Erzeugt ein neues panel und liefert dessen Eventnummer zurück.
<b>j_popupmenu</b>	<i>function j_popupmenu ( obj , label\$ )</i> Erzeugt ein neues Popupmenu mit dem Inhalt <b>label</b> und liefert dessen Eventnummer zurück.
<b>j_print</b>	<i>procedure j_print ( obj )</i> Druckt den Inhalt des Borderpanel .
<b>j_progressbar</b>	<i>function j_progressbar ( obj , orient )</i> Erzeugt eine Progress Bar mit der Ausrichtung <b>orient</b> .
<b>j_radiogroup</b>	<i>function j_radiogroup ( obj )</i> Erzeugt eine neue Radiogroup und liefert dessen Eventnummer zurück.
<b>j_releaseall</b>	<i>procedure j_releaseall ( obj )</i>

Gibt alle Elemente aus dem Borderpanel **obj** frei.

<b>j_release</b>	<i>procedure j_release ( obj )</i> Gibt das Borderpanel <b>obj</b> aus dem umgebenden Container Element frei.
<b>j_scrollpane</b>	<i>function j_scrollpane ( obj )</i> Erzeugt eine neue Scrollpane und gibt dessen Eventnummer zurück.
<b>j_setalign</b>	<i>procedure j_setalign ( obj , align )</i> Setzt die Ausrichtung innerhalb des Borderpanel <b>obj</b> auf <b>align</b> . Ein Flowlayout Manager wird vorausgesetzt.
<b>j_setborderlayout</b>	<i>procedure j_setborderlayout ( obj )</i> Integriert einen Borderlayout Manager in das Borderpanel <b>obj</b> .
<b>j_setborderpos</b>	<i>procedure j_setborderpos ( obj , pos )</i> Positioniert das Borderpanel <b>obj</b> an die angegebene Position <b>pos</b> . Im umgebenden Container muß einen Borderlayout Manager eingestellt sein.
<b>j_setcolorbg</b>	<i>procedure j_setcolorbg ( obj , r , g , b )</i> Setzt die Hintergrundfarbe auf den übergebenen RGB Wert
<b>j_setcolor</b>	<i>procedure j_setcolor ( obj , r , g , b )</i> Setzt die Vordergrundfarbe auf den übergebenen RGB Wert
<b>j_setcursor</b>	<i>function j_setcursor ( obj , cursor )</i> Setzt den Cursor des Borderpanel <b>obj</b> auf <b>cursor</b> .
<b>j_setfixlayout</b>	<i>procedure j_setfixlayout ( obj )</i> Integriert einen Fixlayout Manager in das Borderpanel <b>obj</b> (voreingestellte Layoutmanager).
<b>j_setflowfill</b>	<i>procedure j_setflowfill ( obj , bool )</i> Setzt die Höhe (Breite) aller enthaltenen Komponenten auf die Höhe (Breite) von dem Borderpanel <b>obj</b> . Ein Flowlayout Manager wird vorausgesetzt.
<b>j_setflowlayout</b>	<i>procedure j_setflowlayout ( obj , align )</i> Integriert einen Flowlayout Manager in das Borderpanel <b>obj</b> mit der Ausrichtung <b>align</b> .
<b>j_setfocus</b>	<i>function j_setfocus ( obj )</i> Weist den Input Focus dem Borderpanel <b>obj</b> zu.
<b>j_setfontname</b>	<i>procedure j_setfontname ( obj , name )</i> Setzt den Fonttyp des Borderpanel <b>obj</b> auf <b>name</b> .
<b>j_setfont</b>	<i>procedure j_setfont ( obj , name , style , size )</i> Setzt den Font des Borderpanel <b>obj</b> entsprechend der Parameterliste.
<b>j_setfontsize</b>	<i>procedure j_setfontsize ( obj , size )</i> Setzt den Fontgröße des Borderpanel <b>obj</b> auf <b>size</b> .

<b>j_setfontstyle</b>	<i>procedure j_setfontstyle ( obj , style )</i> Setzt den Fontstyle des Borderpanel <b>obj</b> auf <b>style</b> .
<b>j_setgridlayout</b>	<i>procedure j_setgridlayout ( obj , row , col )</i> Integriert einen Gridlayout Manager in das Borderpanel <b>obj</b> mit <b>row</b> Reihen und <b>col</b> Spalten.
<b>j_sethgap</b>	<i>procedure j_sethgap ( obj , hgap )</i> Setzt den horizontalen Abstand innerhalb des Borderpanel <b>obj</b> auf <b>hgap</b> Pixel.
<b>j_setinsets</b>	<i>procedure j_setinsets ( obj , top , bottom , left , right )</i> Setzt den inneren Rahmen auf die angegebenen Größen.
<b>j_setnamedcolorbg</b>	<i>procedure j_setnamedcolorbg ( obj , color )</i> Setzt die Hintergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnamedcolor</b>	<i>procedure j_setnamedcolor ( obj , color )</i> Setzt die Vordergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnolayout</b>	<i>procedure j_setnolayout ( obj )</i> Entfernt den aktuellen Layout Manager aus dem Borderpanel <b>obj</b> .
<b>j_setpos</b>	<i>procedure j_setpos ( obj , xpos , ypos )</i> Verschiebt das Borderpanel <b>obj</b> zur angegebenen Koordinate ( <b>xpos,ypos</b> ) an.
<b>j_setsize</b>	<i>procedure j_setsize ( obj , width , height )</i> Setzt Breite und Höhe des Borderpanel <b>obj</b> auf <b>width</b> and <b>height</b> .
<b>j_setvgap</b>	<i>procedure j_setvgap ( obj , vgap )</i> Setzt den vertikalen Abstand innerhalb des Borderpanel <b>obj</b> auf <b>vgap</b> Pixel.
<b>j_sevensegment</b>	<i>function j_sevensegment ( obj , color )</i> Erzeugt eine Siebensegment Anzeige der Farbe <b>color</b> .
<b>j_show</b>	<i>procedure j_show ( obj )</i> Zeigt das Borderpanel <b>obj</b> auf dem Bildschirm an.
<b>j_textarea</b>	<i>function j_textarea ( obj , rows , columns )</i> Erzeugt eine neue Textarea und gibt dessen Eventnummer zurück.
<b>j_textfield</b>	<i>function j_textfield ( obj , columns )</i> Erzeugt eine neues Textfield und gibt dessen Eventnummer zurück.
<b>j_vscrollbar</b>	<i>function j_vscrollbar ( obj )</i> Erzeugt eine vertikale Scrollbar und gibt dessen Eventnummer zurück.

Canvas
--------

- j\_canvas**      *function j\_canvas ( obj , width , height )*  
Erzeugt einen neuen Canvas mit der Breite **width** Pixel und einer Höhe von **height** Pixel.
- j\_add**      *procedure j\_add ( obj , cont )*  
Fügt den Canvas **obj** in den Container **cont** ein.
- j\_cliprect**      *procedure j\_cliprect ( obj , x , y , width , height )*  
Setzt den Clipping Bereich auf das spezifizierte Rechteck (**x**, **y**, **width**, **height**).
- j\_componentlistener**      *function j\_componentlistener ( obj , kind )*  
Bindet an den Canvas **obj** einen neuen Componentlistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.
- j\_disable**      *procedure j\_disable ( obj )*  
disabled den Canvas **obj**, sodaß diese auf keine Benutzereingaben mehr reagiert.
- j\_dispose**      *procedure j\_dispose ( obj )*  
der Canvas **obj** wird mit allen Recourcen gelöscht.
- j\_drawarc**      *procedure j\_drawarc ( obj , x , y , rx , ry , arc1 , arc2 )*  
Zeichnet einen Kreisbogen mit dem Mittelpunkt (**x**, **y**) und den Hauptradien **rx** und **ry**. Die Parameter **arc1** und **arc2** bestimmen den Start- bzw. den Endwinkel des Kreisbogens.
- j\_drawcircle**      *procedure j\_drawcircle ( obj , x , y , r )*  
Zeichnet einen Kreis mit dem Mittelpunkt (**x,y**) und dem Radius **r**.
- j\_drawimage**      *procedure j\_drawimage ( obj , image , x , y )*  
Kopiert das Image mit der Eventnummer **image** an die Position (**x**, **y**).
- j\_drawimagesource**      *procedure j\_drawimagesource ( obj , x , y , w , h , r , g , b )*  
Die Funktion stellt ein Bild an der Position (**x,y**) mit der Breite **w** und der Höhe **h** dar. Die Bilddaten werden als rot-, grün- und blau- Kanal in den Arrays **r,g,b** übergeben.
- j\_drawline**      *procedure j\_drawline ( obj , x1 , y1 , x2 , y2 )*  
Zeichnet eine Linie in der momentan gesetzten Vordergrundfarbe von (**x1,y1**) nach (**x2,y2**)
- j\_drawoval**      *procedure j\_drawoval ( obj , x , y , rx , ry )*  
Zeichnet einen Oval mit dem Mittelpunkt (**x,y**) und den Hauptradien **rx** und **ry**.

<b>j_drawpixel</b>	<i>procedure j_drawpixel ( obj , x , y )</i> Zeichnet einen Pixel in der momentan gesetzten Vordergrundfarbe an die Position <b>(x,y)</b> .
<b>j_drawpolygon</b>	<i>procedure j_drawpolygon ( obj , len , x , y )</i> Zeichnet ein Polygon durch die Punkte, die durch die Arrays <b>x</b> und <b>y</b> gegeben sind.
<b>j_drawpolyline</b>	<i>procedure j_drawpolyline ( obj , len , x , y )</i> Zeichnet eine Polylinie durch die Punkte, die durch die Arrays <b>x</b> und <b>y</b> gegeben sind.
<b>j_drawrect</b>	<i>procedure j_drawrect ( obj , x , y , width , height )</i> Zeichnet ein Rechteck an die Position <b>(x,y)</b> mit der Breite <b>width</b> und der Höhe <b>height</b> .
<b>j_drawroundrect</b>	<i>procedure j_drawroundrect ( obj , x , y , width , height , arcx , arcy )</i> Zeichnet ein Rechteck mit abgerundeten Ecken an die Position <b>(x,y)</b> mit der Breite <b>width</b> und der Höhe <b>height</b> . Die Radien der Ecken werden durch <b>arcx</b> und <b>arcy</b> bestimmt.
<b>j_drawscaledimage</b>	<i>procedure j_drawscaledimage ( obj , image , sx , sy , sw , sh , tx , ty , tw , th )</i> Kopiert einen Ausschnitt des Bildes <b>image</b> , der durch die Parameter <b>(sx, sy, sw, sh)</b> bestimmt wird an die Position <b>(tx, ty)</b> und scaliert den Ausschnitt auf die Breite <b>tw</b> und die Höhe <b>th</b>
<b>j_drawstring</b>	<i>procedure j_drawstring ( obj , x , y , str\$ )</i> Zeichnet den String <b>str</b> an die durch die Parameter <b>(x,y)</b> festgelegte Koordinate.
<b>j_enable</b>	<i>procedure j_enable ( obj )</i> enabled den Canvas <b>obj</b> .
<b>j_fillarc</b>	<i>procedure j_fillarc ( obj , x , y , rx , ry , arc1 , arc2 )</i> Zeichnet einen gefüllten Kreisbogen mit dem Mittelpunkt <b>(x,y)</b> und den Hauptradien <b>rx</b> und <b>ry</b> . Die Parameter <b>arc1</b> und <b>arc2</b> bestimmen den Start- bzw. den Endwinkel des Kreisbogens.
<b>j_fillcircle</b>	<i>procedure j_fillcircle ( obj , x , y , r )</i> Zeichnet einen gefüllten Kreis mit dem Mittelpunkt <b>(x,y)</b> und dem Radius <b>r</b> .
<b>j_filloval</b>	<i>procedure j_filloval ( obj , x , y , rx , ry )</i> Zeichnet einen gefüllten Oval mit dem Mittelpunkt <b>(x,y)</b> und den Hauptradien <b>rx</b> und <b>ry</b> .
<b>j_fillpolygon</b>	<i>procedure j_fillpolygon ( obj , len , x , y )</i> Zeichnet ein gefülltes Polygon durch die Punkte, die durch die Arrays <b>x</b> und <b>y</b> gegeben sind.
<b>j_fillrect</b>	<i>procedure j_fillrect ( obj , x , y , width , height )</i> Zeichnet ein gefülltes Rechteck an die Position <b>(x,y)</b> mit der Breite <b>width</b> und der Höhe <b>height</b> .

<b>j_fillroundrect</b>	<i>procedure j_fillroundrect ( obj , x , y , width , height , arcx , arcy )</i> Zeichnet ein gefülltes Rechteck mit abgerundeten Ecken an die Position ( <b>x,y</b> ). Das Rechteck hat eine Breite von <b>width</b> und eine Höhe von <b>height</b> Pixeln. Die Form der runden Ecken wird durch die Parameter <b>arcx</b> und <b>arcy</b> bestimmt.
<b>j_focuslistener</b>	<i>function j_focuslistener ( obj )</i> Bindet an den Canvas <b>obj</b> einen neuen Focuslistener, und liefert dessen Eventnummer zurück.
<b>j_getfontascent</b>	<i>function j_getfontascent ( obj )</i> Liefert die Oberlänge (Ascent) des aktuellen Fonts des Canvas <b>obj</b> in Bildschirmpixeln zurück.
<b>j_getfontheight</b>	<i>function j_getfontheight ( obj )</i> Liefert die totale Höhe des aktuellen Fonts des Canvas <b>obj</b> in Bildschirmpixeln zurück.
<b>j_getheight</b>	<i>function j_getheight ( obj )</i> Liefert die Höhe des Canvas <b>obj</b> zurück.
<b>j_getimage</b>	<i>function j_getimage ( obj )</i> Kopiert den Inhalt dem Canvas <b>obj</b> in ein Image und liefert dessen Eventnummer zurück.
<b>j_getimagesource</b>	<i>function j_getimagesource ( obj , x , y , w , h , r , g , b )</i> Die Funktion liefert einen Ausschnitt vom Punkt ( <b>x, y</b> ) der Breite <b>w</b> und der Höhe <b>h</b> zurück. Der Bildinhalt wird als rot-, grün- und blau- Kanal in den Arrays r,g,b abgelegt.
<b>j_getparentid</b>	<i>function j_getparentid ( obj )</i> Liefert die Eventnummer des umgebenen Containers von <b>obj</b> zurück. Ist <b>obj</b> ein Frame wird <b>-1</b> zurückgegeben.
<b>j_getparent</b>	<i>function j_getparent ( obj )</i> Liefert die Eventnummer des umgebenen Containers von <b>obj</b> zurück. Ist <b>obj</b> ein Frame wird <b>-1</b> zurückgegeben.
<b>j_getscaledimage</b>	<i>function j_getscaledimage ( obj , x , y , sw , sh , tw , th )</i> Kopiert den Inhalt des Ausschnitts von Positio ( <b>x, y</b> ) it der Breite <b>sw</b> und der höhe <b>sh</b> in ein Image und liefert dessen Eventnummer zurück. Das Image wird dabei auf eine Breite von <b>tw</b> und einer Höhe von <b>th</b> scaliert.
<b>j_getstringwidth</b>	<i>function j_getstringwidth ( obj , str\$ )</i> Liefert die Länge des <b>str</b> des aktuellen Fonts des Canvas <b>obj</b> in Bildschirmpixeln zurück.
<b>j_getwidth</b>	<i>function j_getwidth ( obj )</i> Liefert die Breite des Canvas <b>obj</b> zurück.
<b>j_getxpos</b>	<i>function j_getxpos ( obj )</i> Liefert die aktuelle horizontale Position des Canvas <b>obj</b> zurück.
<b>j_getypos</b>	<i>function j_getypos ( obj )</i>

Liefert die aktuelle vertikale Position des Canvas **obj** zurück.

<b>j_hide</b>	<i>procedure j_hide ( obj )</i> Zeigt den Canvas <b>obj</b> nicht mehr auf dem Bildschirm an.
<b>j_isparent</b>	<i>function j_isparent ( obj , cont )</i> Liefert J_TRUE zurück, falls das <b>cont</b> die umgebende Komponente von <b>obj</b> ist, sonst J_FALSE
<b>j_isvisible</b>	<i>function j_isvisible ( obj )</i> Liefert J_TRUE zurück, falls <b>obj</b> sichtbar ist, sonst J_FALSE
<b>j_keylistener</b>	<i>function j_keylistener ( obj )</i> Bindet an den Canvas <b>obj</b> einen neuen Keylistener, und liefert dessen Eventnummer zurück.
<b>j_mouselistener</b>	<i>function j_mouselistener ( obj , kind )</i> Bindet an den Canvas <b>obj</b> einen neuen Mouselistener, und liefert dessen Eventnummer zurück. Der Parameter <b>kind</b> bestimmt, wann ein Event ausgelöst wird.
<b>j_popupmenu</b>	<i>function j_popupmenu ( obj , label\$ )</i> Erzeugt ein neues Popupmenu mit dem Inhalt <b>label</b> und liefert dessen Eventnummer zurück.
<b>j_print</b>	<i>procedure j_print ( obj )</i> Druckt den Inhalt des Canvas .
<b>j_release</b>	<i>procedure j_release ( obj )</i> Gibt den Canvas <b>obj</b> aus dem umgebenden Container Element frei.
<b>j_setborderpos</b>	<i>procedure j_setborderpos ( obj , pos )</i> Positioniert den Canvas <b>obj</b> an die angegebene Position <b>pos</b> . Im umgebenden Container muß einen BorderLayout Manager eingestellt sein.
<b>j_setcolorbg</b>	<i>procedure j_setcolorbg ( obj , r , g , b )</i> Setzt die Hintergrundfarbe auf den übergebenen RGB Wert
<b>j_setcolor</b>	<i>procedure j_setcolor ( obj , r , g , b )</i> Setzt die Vordergrundfarbe auf den übergebenen RGB Wert
<b>j_setcursor</b>	<i>function j_setcursor ( obj , cursor )</i> Setzt den Cursor des Canvas <b>obj</b> auf <b>cursor</b> .
<b>j_setfocus</b>	<i>function j_setfocus ( obj )</i> Weist den Input Focus dem Canvas <b>obj</b> zu.
<b>j_setfontname</b>	<i>procedure j_setfontname ( obj , name )</i> Setzt den Fonttyp des Canvas <b>obj</b> auf <b>name</b> .
<b>j_setfont</b>	<i>procedure j_setfont ( obj , name , style , size )</i> Setzt den Font des Canvas <b>obj</b> entsprechend der Parameterliste.
<b>j_setfontsize</b>	<i>procedure j_setfontsize ( obj , size )</i>

	Setzt den Fontgröße des Canvas <b>obj</b> auf <b>size</b> .
<b>j_setfontstyle</b>	<i>procedure j_setfontstyle ( obj , style )</i> Setzt den Fontstyle des Canvas <b>obj</b> auf <b>style</b> .
<b>j_setnamedcolorbg</b>	<i>procedure j_setnamedcolorbg ( obj , color )</i> Setzt die Hintergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnamedcolor</b>	<i>procedure j_setnamedcolor ( obj , color )</i> Setzt die Vordergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setpos</b>	<i>procedure j_setpos ( obj , xpos , ypos )</i> Verschiebt den Canvas <b>obj</b> zur angegebenen Koordinate ( <b>xpos,ypos</b> ) an.
<b>j_setsize</b>	<i>procedure j_setsize ( obj , width , height )</i> Setzt Breite und Höhe des Canvas <b>obj</b> auf <b>width</b> and <b>height</b> .
<b>j_setxor</b>	<i>procedure j_setxor ( obj , bool )</i> Schaltet den Painting Mode auf XOR, falls bool = J_TRUE ist. Im XOR Modus heben sich zwei gleiche Graphikbefehle auf.
<b>j_show</b>	<i>procedure j_show ( obj )</i> Zeigt den Canvas <b>obj</b> auf dem Bildschirm an.
<b>j_translate</b>	<i>procedure j_translate ( obj , x , y )</i> Setzt den Ursprung des Koordinatensystems auf ( <b>x, y</b> ).



Checkbox
----------

- j\_checkbox**      *function j\_checkbox ( obj , label\$ )*  
Erzeugt eine Checkbox mit dem Textinhalt **label** und liefert dessen Eventnummer zurück.
- j\_add**            *procedure j\_add ( obj , cont )*  
Fügt die Checkbox **obj** in den Container **cont** ein.
- j\_componentlistener** *function j\_componentlistener ( obj , kind )*  
Bindet an die Checkbox **obj** einen neuen Componentlistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.
- j\_disable**        *procedure j\_disable ( obj )*  
disabled die Checkbox **obj**, sodaß diese auf keine Benutzereingaben mehr reagiert.
- j\_dispose**        *procedure j\_dispose ( obj )*  
die Checkbox **obj** wird mit allen Recourcen gelöscht.
- j\_enable**         *procedure j\_enable ( obj )*  
enabled die Checkbox **obj**.
- j\_focuslistener**    *function j\_focuslistener ( obj )*  
Bindet an die Checkbox **obj** einen neuen Focuslistener, und liefert dessen Eventnummer zurück.
- j\_getfontascent**    *function j\_getfontascent ( obj )*  
Liefert die Oberlänge (Ascent) des aktuellen Fonts der Checkbox **obj** in Bildschirmpixeln zurück.
- j\_getfontheight**    *function j\_getfontheight ( obj )*  
Liefert die totale Höhe des aktuellen Fonts der Checkbox **obj** in Bildschirmpixeln zurück.
- j\_getheight**        *function j\_getheight ( obj )*  
Liefert die Höhe der Checkbox **obj** zurück.
- j\_getparentid**      *function j\_getparentid ( obj )*  
Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird  $-1$  zurückgegeben.
- j\_getparent**        *function j\_getparent ( obj )*  
Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird  $-1$  zurückgegeben.
- j\_getstate**         *function j\_getstate ( obj )*  
Liefert J\_TRUE , falls die Checkbox selektiert ist, sonst J\_FALSE .

<b>j_getstringwidth</b>	<i>function j_getstringwidth ( obj , str\$ )</i> Liefert die Länge des <b>str</b> des aktuellen Fonts der Checkbox <b>obj</b> in Bildschirm-pixeln zurück.
<b>j_gettext</b>	<i>function\$ j_gettext ( obj )</i> Liefert den Textinhalt der Checkbox .
<b>j_getwidth</b>	<i>function j_getwidth ( obj )</i> Liefert die Breite der Checkbox <b>obj</b> zurück.
<b>j_getxpos</b>	<i>function j_getxpos ( obj )</i> Liefert die aktuelle horizontale Position der Checkbox <b>obj</b> zurück.
<b>j_getypos</b>	<i>function j_getypos ( obj )</i> Liefert die aktuelle vertikale Position der Checkbox <b>obj</b> zurück.
<b>j_hide</b>	<i>procedure j_hide ( obj )</i> Zeigt die Checkbox <b>obj</b> nicht mehr auf dem Bildschirm an.
<b>j_isparent</b>	<i>function j_isparent ( obj , cont )</i> Liefert J_TRUE zurück, falls das <b>cont</b> die umgebende Komponente von <b>obj</b> ist, sonst J_FALSE
<b>j_isvisible</b>	<i>function j_isvisible ( obj )</i> Liefert J_TRUE zurück, falls <b>obj</b> sichtbar ist, sonst J_FALSE
<b>j_keylistener</b>	<i>function j_keylistener ( obj )</i> Bindet an die Checkbox <b>obj</b> einen neuen Keylistener, und liefert dessen Eventnummer zurück.
<b>j_mouselistener</b>	<i>function j_mouselistener ( obj , kind )</i> Bindet an die Checkbox <b>obj</b> einen neuen Mouselistener, und liefert dessen Eventnummer zurück. Der Parameter <b>kind</b> bestimmt, wann ein Event ausgelöst wird.
<b>j_popupmenu</b>	<i>function j_popupmenu ( obj , label\$ )</i> Erzeugt ein neues Popupmenu mit dem Inhalt <b>label</b> und liefert dessen Eventnummer zurück.
<b>j_print</b>	<i>procedure j_print ( obj )</i> Druckt den Inhalt der Checkbox .
<b>j_release</b>	<i>procedure j_release ( obj )</i> Gibt die Checkbox <b>obj</b> aus dem umgebenden Container Element frei.
<b>j_setborderpos</b>	<i>procedure j_setborderpos ( obj , pos )</i> Positioniert die Checkbox <b>obj</b> an die angegebene Position <b>pos</b> . Im umgebenden Container muß einen BorderLayout Manager eingestellt sein.
<b>j_setcolorbg</b>	<i>procedure j_setcolorbg ( obj , r , g , , b )</i> Setzt die Hintergrundfarbe auf den übergebenen RGB Wert
<b>j_setcolor</b>	<i>procedure j_setcolor ( obj , r , g , , b )</i>

	Setzt die Vordergrundfarbe auf den übergebenen RGB Wert
<b>j_setcursor</b>	<i>function j_setcursor ( obj , cursor )</i> Setzt den Cursor der Checkbox <b>obj</b> auf <b>cursor</b> .
<b>j_setfocus</b>	<i>function j_setfocus ( obj )</i> Weist den Input Focus der Checkbox <b>obj</b> zu.
<b>j_setfontname</b>	<i>procedure j_setfontname ( obj , name )</i> Setzt den Fonttyp der Checkbox <b>obj</b> auf <b>name</b> .
<b>j_setfont</b>	<i>procedure j_setfont ( obj , name , style , size )</i> Setzt den Font der Checkbox <b>obj</b> entsprechend der Parameterliste.
<b>j_setfontsize</b>	<i>procedure j_setfontsize ( obj , size )</i> Setzt den Fontgröße der Checkbox <b>obj</b> auf <b>size</b> .
<b>j_setfontstyle</b>	<i>procedure j_setfontstyle ( obj , style )</i> Setzt den Fontstyle der Checkbox <b>obj</b> auf <b>style</b> .
<b>j_setnamedcolorbg</b>	<i>procedure j_setnamedcolorbg ( obj , color )</i> Setzt die Hintergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnamedcolor</b>	<i>procedure j_setnamedcolor ( obj , color )</i> Setzt die Vordergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setpos</b>	<i>procedure j_setpos ( obj , xpos , ypos )</i> Verschiebt die Checkbox <b>obj</b> zur angegebenen Koordinate ( <b>xpos,ypos</b> ) an.
<b>j_setsize</b>	<i>procedure j_setsize ( obj , width , height )</i> Setzt Breite und Höhe der Checkbox <b>obj</b> auf <b>width</b> and <b>height</b> .
<b>j_setstate</b>	<i>procedure j_setstate ( obj , bool )</i> Ist <b>bool</b> J.TRUE so wird die Checkbox selectiert.
<b>j_settext</b>	<i>procedure j_settext ( obj , str\$ )</i> Ändert den Textinhalt der Checkbox <b>obj</b> auf <b>str</b>
<b>j_show</b>	<i>procedure j_show ( obj )</i> Zeigt die Checkbox <b>obj</b> auf dem Bildschirm an.

Checkmenüitem
---------------

- j\_checkmenüitem**    *function j\_checkmenüitem ( obj , label\$ )*  
 Erzeugt ein Checkmenüitem mit der Beschriftung **label** und liefert dessen Eventnummer zurück.
- j\_disable**            *procedure j\_disable ( obj )*  
 disabled das Checkmenüitem **obj**, sodaß diese auf keine Benutzereingaben mehr reagiert.
- j\_dispose**            *procedure j\_dispose ( obj )*  
 das Checkmenüitem **obj** wird mit allen Recourcen gelöscht.
- j\_enable**             *procedure j\_enable ( obj )*  
 enabled das Checkmenüitem **obj**.
- j\_getlength**         *function j\_getlength ( obj )*  
 Liefert die Länge des Textinhaltes des Checkmenüitem zurück.
- j\_getstate**           *function j\_getstate ( obj )*  
 Liefert J\_TRUE , falls das Checkmenüitem selektiert ist, sonst J\_FALSE .
- j\_gettext**            *function\$ j\_gettext ( obj )*  
 Liefert den Textinhalt des Checkmenüitem .
- j\_setfontname**        *procedure j\_setfontname ( obj , name )*  
 Setzt den Fonttyp des Checkmenüitem **obj** auf **name**.
- j\_setfont**             *procedure j\_setfont ( obj , name , style , size )*  
 Setzt den Font des Checkmenüitem **obj** entsprechend der Parameterliste.
- j\_setfontsize**        *procedure j\_setfontsize ( obj , size )*  
 Setzt den Fontgröße des Checkmenüitem **obj** auf **size**.
- j\_setfontstyle**       *procedure j\_setfontstyle ( obj , style )*  
 Setzt den Fontstyle des Checkmenüitem **obj** auf **style**.
- j\_setshortcut**        *procedure j\_setshortcut ( obj , chr )*  
 Bestimmt das Zeichen **chr** als Shortcut für das Checkmenüitem .
- j\_setstate**            *procedure j\_setstate ( obj , bool )*  
 Ist **bool** J\_TRUE so wird das Checkmenüitem selectiert.
- j\_settext**             *procedure j\_settext ( obj , str\$ )*  
 Ändert den Textinhalt des Checkmenüitem **obj** auf **str**

Choice
--------

- j\_choice**                    *function j\_choice ( obj )*  
Erzeugt eine neue Choice und gibt deren Eventnummer zurück.
- j\_additem**                    *procedure j\_additem ( obj , str\$ )*  
Hängt ein neues Item mit dem Inhalt **str** an die Auswahlliste an.
- j\_add**                            *procedure j\_add ( obj , cont )*  
Fügt die Choice **obj** in den Container **cont** ein.
- j\_componentlistener** *function j\_componentlistener ( obj , kind )*  
Bindet an die Choice **obj** einen neuen Componentlistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.
- j\_disable**                    *procedure j\_disable ( obj )*  
disabled die Choice **obj**, sodaß diese auf keine Benutzereingaben mehr reagiert.
- j\_dispose**                    *procedure j\_dispose ( obj )*  
die Choice **obj** wird mit allen Recourcen gelöscht.
- j\_enable**                      *procedure j\_enable ( obj )*  
enabled die Choice **obj**.
- j\_focuslistener**            *function j\_focuslistener ( obj )*  
Bindet an die Choice **obj** einen neuen Focuslistener, und liefert dessen Eventnummer zurück.
- j\_getfontascent**            *function j\_getfontascent ( obj )*  
Liefert die Oberlänge (Ascent) des aktuellen Fonts der Choice **obj** in Bildschirmpixeln zurück.
- j\_getfontheight**            *function j\_getfontheight ( obj )*  
Liefert die totale Höhe des aktuellen Fonts der Choice **obj** in Bildschirmpixeln zurück.
- j\_getheight**                *function j\_getheight ( obj )*  
Liefert die Höhe der Choice **obj** zurück.
- j\_getitemcount**            *function j\_getitemcount ( obj )*  
Liefert die Anzahl der Items der Choice **obj** zurück.
- j\_getitem**                    *function\$ j\_getitem ( obj , item )*  
liefert den Inhalt des Items mit dem Index **item** zurück.
- j\_getparentid**              *function j\_getparentid ( obj )*  
Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird **-1** zurückgegeben.

<b>j_getparent</b>	<i>function j_getparent ( obj )</i> Liefert die Eventnummer des umgebenen Containers von <b>obj</b> zurück. Ist <b>obj</b> ein Frame wird $-1$ zurückgegeben.
<b>j_getselect</b>	<i>function j_getselect ( obj )</i> Liefert den Index des selectierten Items der Choice .
<b>j_getstringwidth</b>	<i>function j_getstringwidth ( obj , str\$ )</i> Liefert die Länge des <b>str</b> des aktuellen Fonts der Choice <b>obj</b> in Bildschirmpixeln zurück.
<b>j_getwidth</b>	<i>function j_getwidth ( obj )</i> Liefert die Breite der Choice <b>obj</b> zurück.
<b>j_getxpos</b>	<i>function j_getxpos ( obj )</i> Liefert die aktuelle horizontale Position der Choice <b>obj</b> zurück.
<b>j_getypos</b>	<i>function j_getypos ( obj )</i> Liefert die aktuelle vertikale Position der Choice <b>obj</b> zurück.
<b>j_hide</b>	<i>procedure j_hide ( obj )</i> Zeigt die Choice <b>obj</b> nicht mehr auf dem Bildschirm an.
<b>j_insert</b>	<i>function j_insert ( obj , pos , label\$ )</i> fügt ein neues Item in an der Position <b>pos</b> ein. Das Item trägt den Inhalt <b>label</b> .
<b>j_isparent</b>	<i>function j_isparent ( obj , cont )</i> Liefert J_TRUE zurück, falls das <b>cont</b> die umgebende Komponente von <b>obj</b> ist, sonst J_FALSE
<b>j_isvisible</b>	<i>function j_isvisible ( obj )</i> Liefert J_TRUE zurück, falls <b>obj</b> sichtbar ist, sonst J_FALSE
<b>j_keylistener</b>	<i>function j_keylistener ( obj )</i> Bindet an die Choice <b>obj</b> einen neuen Keylistener, und liefert dessen Eventnummer zurück.
<b>j_mouselistener</b>	<i>function j_mouselistener ( obj , kind )</i> Bindet an die Choice <b>obj</b> einen neuen Mouselistener, und liefert dessen Eventnummer zurück. Der Parameter <b>kind</b> bestimmt, wann ein Event ausgelöst wird.
<b>j_popupmenu</b>	<i>function j_popupmenu ( obj , label\$ )</i> Erzeugt ein neues Popupmenu mit dem Inhalt <b>label</b> und liefert dessen Eventnummer zurück.
<b>j_print</b>	<i>procedure j_print ( obj )</i> Druckt den Inhalt der Choice .
<b>j_release</b>	<i>procedure j_release ( obj )</i> Gibt die Choice <b>obj</b> aus dem umgebenden Container Element frei.
<b>j_removeall</b>	<i>function j_removeall ( obj )</i> entfernt alle Items aus der Choice .

<b>j_removeitem</b>	<i>function j_removeitem ( obj , item\$ )</i> entfernt den ersten Eintrag <b>item</b> aus der Choice .
<b>j_remove</b>	<i>function j_remove ( obj , item )</i> entfernt das Item mit dem Index <b>item</b> aus der Choice .
<b>j_select</b>	<i>function j_select ( obj , item )</i> selectiert das Item mit dem Index <b>item</b> .
<b>j_setborderpos</b>	<i>procedure j_setborderpos ( obj , pos )</i> Positioniert die Choice <b>obj</b> an die angegebene Position <b>pos</b> . Im umgebenden Container muß einen BorderLayout Manager eingestellt sein.
<b>j_setcolorbg</b>	<i>procedure j_setcolorbg ( obj , r , g , b )</i> Setzt die Hintergrundfarbe auf den übergebenen RGB Wert
<b>j_setcolor</b>	<i>procedure j_setcolor ( obj , r , g , b )</i> Setzt die Vordergrundfarbe auf den übergebenen RGB Wert
<b>j_setcursor</b>	<i>function j_setcursor ( obj , cursor )</i> Setzt den Cursor der Choice <b>obj</b> auf <b>cursor</b> .
<b>j_setfocus</b>	<i>function j_setfocus ( obj )</i> Weist den Input Focus der Choice <b>obj</b> zu.
<b>j_setfontname</b>	<i>procedure j_setfontname ( obj , name )</i> Setzt den Fonttyp der Choice <b>obj</b> auf <b>name</b> .
<b>j_setfont</b>	<i>procedure j_setfont ( obj , name , style , size )</i> Setzt den Font der Choice <b>obj</b> entsprechend der Parameterliste.
<b>j_setfontsize</b>	<i>procedure j_setfontsize ( obj , size )</i> Setzt den Fontgröße der Choice <b>obj</b> auf <b>size</b> .
<b>j_setfontstyle</b>	<i>procedure j_setfontstyle ( obj , style )</i> Setzt den Fontstyle der Choice <b>obj</b> auf <b>style</b> .
<b>j_setnamedcolorbg</b>	<i>procedure j_setnamedcolorbg ( obj , color )</i> Setzt die Hintergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnamedcolor</b>	<i>procedure j_setnamedcolor ( obj , color )</i> Setzt die Vordergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setpos</b>	<i>procedure j_setpos ( obj , xpos , ypos )</i> Verschiebt die Choice <b>obj</b> zur angegebenen Koordinate ( <b>xpos,ypos</b> ) an.
<b>j_setsize</b>	<i>procedure j_setsize ( obj , width , height )</i> Setzt Breite und Höhe der Choice <b>obj</b> auf <b>width</b> and <b>height</b> .
<b>j_show</b>	<i>procedure j_show ( obj )</i> Zeigt die Choice <b>obj</b> auf dem Bildschirm an.

Dialog
--------

- j\_dialog**            *function j\_dialog ( obj , label\$ )*  
Erzeugt ein neues Dialogwindow mit dem Titel **label** und liefert dessen Eventnummer zurück.
- j\_add**                *procedure j\_add ( obj , cont )*  
Fügt den Dialog **obj** in den Container **cont** ein.
- j\_borderpanel**      *function j\_borderpanel ( obj , type )*  
Erzeugt ein neues Borderpanel und liefert dessen Eventnummer zurück.
- j\_button**            *function j\_button ( obj , label\$ )*  
Erzeugt einen neuen Button mit dem Textinhalt **label**
- j\_canvas**            *function j\_canvas ( obj , width , height )*  
Erzeugt einen neuen Canvas mit der Breite **width** Pixel und einer Höhe von **height** Pixel.
- j\_checkbox**         *function j\_checkbox ( obj , label\$ )*  
Erzeugt eine Checkbox mit dem Textinhalt **label** und liefert dessen Eventnummer zurück.
- j\_choice**            *function j\_choice ( obj )*  
Erzeugt eine neue Choice und gibt deren Eventnummer zurück.
- j\_componentlistener** *function j\_componentlistener ( obj , kind )*  
Bindet an den Dialog **obj** einen neuen Componentlistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.
- j\_disable**           *procedure j\_disable ( obj )*  
disabled den Dialog **obj**, sodaß diese auf keine Benutzereingaben mehr reagiert.
- j\_dispose**          *procedure j\_dispose ( obj )*  
der Dialog **obj** wird mit allen Recourcen gelöscht.
- j\_enable**            *procedure j\_enable ( obj )*  
enabled den Dialog **obj**.
- j\_focuslistener**    *function j\_focuslistener ( obj )*  
Bindet an den Dialog **obj** einen neuen Focuslistener, und liefert dessen Eventnummer zurück.
- j\_getfontascent**    *function j\_getfontascent ( obj )*  
Liefert die Oberlänge (Ascent) des aktuellen Fonts des Dialog **obj** in Bildschirmpixeln zurück.
- j\_getfontheight**    *function j\_getfontheight ( obj )*



	Liefert die totale Höhe des aktuellen Fonts des Dialog <b>obj</b> in Bildschirmpixeln zurück.
<b>j_getheight</b>	<i>function j_getheight ( obj )</i> Liefert die Höhe des Dialog <b>obj</b> zurück.
<b>j_getinheight</b>	<i>function j_getinheight ( cont )</i> Liefert die Höhe des Dialog <b>obj</b> des inneren Bereichs zurück.
<b>j_getinsets</b>	<i>function j_getinsets ( obj , side )</i> Liefert die angeforderte Breite des inneren Randes dem Dialog zurück.
<b>j_getinwidth</b>	<i>function j_getinwidth ( cont )</i> Liefert die Breite des Dialog <b>obj</b> des inneren Bereichs zurück.
<b>j_getlayoutid</b>	<i>function j_getlayoutid ( obj )</i> Liefert die Eventnummer des Layoutmanagers des Containers <b>obj</b> zurück.
<b>j_getlength</b>	<i>function j_getlength ( obj )</i> Liefert die Länge des Textinhaltes den Dialog zurück.
<b>j_getparentid</b>	<i>function j_getparentid ( obj )</i> Liefert die Eventnummer des umgebenen Containers von <b>obj</b> zurück. Ist <b>obj</b> ein Frame wird $-1$ zurückgegeben.
<b>j_getparent</b>	<i>function j_getparent ( obj )</i> Liefert die Eventnummer des umgebenen Containers von <b>obj</b> zurück. Ist <b>obj</b> ein Frame wird $-1$ zurückgegeben.
<b>j_getstringwidth</b>	<i>function j_getstringwidth ( obj , str\$ )</i> Liefert die Länge des <b>str</b> des aktuellen Fonts des Dialog <b>obj</b> in Bildschirmpixeln zurück.
<b>j_gettext</b>	<i>function\$ j_gettext ( obj )</i> Liefert den Textinhalt des Dialog .
<b>j_getwidth</b>	<i>function j_getwidth ( obj )</i> Liefert die Breite des Dialog <b>obj</b> zurück.
<b>j_getxpos</b>	<i>function j_getxpos ( obj )</i> Liefert die aktuelle horizontale Position des Dialog <b>obj</b> zurück.
<b>j_getypos</b>	<i>function j_getypos ( obj )</i> Liefert die aktuelle vertikale Position des Dialog <b>obj</b> zurück.
<b>j_graphicbutton</b>	<i>function j_graphicbutton ( obj , filename\$ )</i> Erzeugt einen neuen Graphicbutton mit dem Bildinhalt des Files, das durch <b>filename</b> spezifiziert wird.
<b>j_graphiclabel</b>	<i>function j_graphiclabel ( obj , str\$ )</i> Erzeugt ein neues Graphiclabel mit dem Bildinhalt des Files, das durch <b>filename</b> spezifiziert wird.

<b>j_hide</b>	<i>procedure j_hide ( obj )</i> Zeigt den Dialog <b>obj</b> nicht mehr auf dem Bildschirm an.
<b>j_hscrollbar</b>	<i>function j_hscrollbar ( obj )</i> Erzeugt eine horizontale Scrollbar und gibt dessen Eventnummer zurück.
<b>j_isparent</b>	<i>function j_isparent ( obj , cont )</i> Liefert J_TRUE zurück, falls das <b>cont</b> die umgebende Komponente von <b>obj</b> ist, sonst J_FALSE
<b>j_isresizable</b>	<i>function j_isresizable ( obj )</i> Liefert J_FALSE zurück, falls der Dialog eine fixe Größe besitzt, J_TRUE sonst.
<b>j_isvisible</b>	<i>function j_isvisible ( obj )</i> Liefert J_TRUE zurück, falls <b>obj</b> sichtbar ist, sonst J_FALSE
<b>j_keylistener</b>	<i>function j_keylistener ( obj )</i> Bindet an den Dialog <b>obj</b> einen neuen Keylistener, und liefert dessen Eventnummer zurück.
<b>j_label</b>	<i>function j_label ( obj , label\$ )</i> Erzeugt ein neues Label Element mit dem Textinhalt <b>label</b>
<b>j_led</b>	<i>function j_led ( obj , style , color )</i> Erzeugt eine LED der Farbe <b>color</b> und der Form <b>style</b>
<b>j_line</b>	<i>function j_line ( obj , orient , style , length )</i> Erzeugt ein neues Linienelement und liefert dessen Eventnummer zurück
<b>j_list</b>	<i>function j_list ( obj , rows )</i> Erzeugt eine neue List und gibt deren Eventnummer zurück.
<b>j_meter</b>	<i>function j_meter ( obj , title\$ )</i> Erzeugt ein analoges Anzeigeelement mit der Beschriftung <b>title</b> .
<b>j_mouselistener</b>	<i>function j_mouselistener ( obj , kind )</i> Bindet an den Dialog <b>obj</b> einen neuen Mouselistener, und liefert dessen Eventnummer zurück. Der Parameter <b>kind</b> bestimmt, wann ein Event ausgelöst wird.
<b>j_pack</b>	<i>procedure j_pack ( obj )</i> Veranlaßt den Layoutmanager von <b>obj</b> die minimale Größe zu berechnen.
<b>j_panel</b>	<i>function j_panel ( obj )</i> Erzeugt ein neues panel und liefert dessen Eventnummer zurück.
<b>j_popupmenu</b>	<i>function j_popupmenu ( obj , label\$ )</i> Erzeugt ein neues Popuptastelement mit dem Inhalt <b>label</b> und liefert dessen Eventnummer zurück.
<b>j_print</b>	<i>procedure j_print ( obj )</i> Druckt den Inhalt des Dialog .

<b>j_progressbar</b>	<i>function j_progressbar ( obj , orient )</i> Erzeugt eine Progress Bar mit der Ausrichtung <b>orient</b> .
<b>j_radiogroup</b>	<i>function j_radiogroup ( obj )</i> Erzeugt eine neue Radiogroup und liefert dessen Eventnummer zurück.
<b>j_releaseall</b>	<i>procedure j_releaseall ( obj )</i> Gibt alle Elemente aus dem Dialog <b>obj</b> frei.
<b>j_release</b>	<i>procedure j_release ( obj )</i> Gibt den Dialog <b>obj</b> aus dem umgebenden Container Element frei.
<b>j_scrollpane</b>	<i>function j_scrollpane ( obj )</i> Erzeugt eine neue Scrollpane und gibt dessen Eventnummer zurück.
<b>j_setalign</b>	<i>procedure j_setalign ( obj , align )</i> Setzt die Ausrichtung innerhalb des Dialog <b>obj</b> auf <b>align</b> . Ein Flowlayout Manager wird vorausgesetzt.
<b>j_setborderlayout</b>	<i>procedure j_setborderlayout ( obj )</i> Integriert einen BorderLayout Manager in den Dialog <b>obj</b> .
<b>j_setborderpos</b>	<i>procedure j_setborderpos ( obj , pos )</i> Positioniert den Dialog <b>obj</b> an die angegebene Position <b>pos</b> . Im umgebenden Container muß einen BorderLayout Manager eingestellt sein.
<b>j_setcolorbg</b>	<i>procedure j_setcolorbg ( obj , r , g , b )</i> Setzt die Hintergrundfarbe auf den übergebenen RGB Wert
<b>j_setcolor</b>	<i>procedure j_setcolor ( obj , r , g , b )</i> Setzt die Vordergrundfarbe auf den übergebenen RGB Wert
<b>j_setcursor</b>	<i>function j_setcursor ( obj , cursor )</i> Setzt den Cursor des Dialog <b>obj</b> auf <b>cursor</b> .
<b>j_setfixlayout</b>	<i>procedure j_setfixlayout ( obj )</i> Integriert einen Fixlayout Manager in den Dialog <b>obj</b> (voreingestellte Layoutmanager).
<b>j_setflowfill</b>	<i>procedure j_setflowfill ( obj , bool )</i> Setzt die Höhe (Breite) aller enthaltenen Komponenten auf die Höhe (Breite) von dem Dialog <b>obj</b> . Ein Flowlayout Manager wird vorausgesetzt.
<b>j_setflowlayout</b>	<i>procedure j_setflowlayout ( obj , align )</i> Integriert einen Flowlayout Manager in den Dialog <b>obj</b> mit der Ausrichtung <b>align</b> .
<b>j_setfocus</b>	<i>function j_setfocus ( obj )</i> Weist den Input Focus dem Dialog <b>obj</b> zu.
<b>j_setfontname</b>	<i>procedure j_setfontname ( obj , name )</i> Setzt den Fonttyp des Dialog <b>obj</b> auf <b>name</b> .

<b>j_setfont</b>	<i>procedure j_setfont ( obj , name , style , size )</i> Setzt den Font des Dialog <b>obj</b> entsprechend der Parameterliste.
<b>j_setfontsize</b>	<i>procedure j_setfontsize ( obj , size )</i> Setzt den Fontgröße des Dialog <b>obj</b> auf <b>size</b> .
<b>j_setfontstyle</b>	<i>procedure j_setfontstyle ( obj , style )</i> Setzt den Fontstyle des Dialog <b>obj</b> auf <b>style</b> .
<b>j_setgridlayout</b>	<i>procedure j_setgridlayout ( obj , row , col )</i> Integriert einen Gridlayout Manager in den Dialog <b>obj</b> mit <b>row</b> Reihen und <b>col</b> Spalten.
<b>j_sethgap</b>	<i>procedure j_sethgap ( obj , hgap )</i> Setzt den horizontalen Abstand innerhalb des Dialog <b>obj</b> auf <b>hgap</b> Pixel.
<b>j_setinsets</b>	<i>procedure j_setinsets ( obj , top , bottom , left , right )</i> Setzt den inneren Rahmen auf die angegebenen Größen.
<b>j_setnamedcolorbg</b>	<i>procedure j_setnamedcolorbg ( obj , color )</i> Setzt die Hintergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnamedcolor</b>	<i>procedure j_setnamedcolor ( obj , color )</i> Setzt die Vordergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnolayout</b>	<i>procedure j_setnolayout ( obj )</i> Entfernt den aktuellen Layout Manager aus dem Dialog <b>obj</b> .
<b>j_setpos</b>	<i>procedure j_setpos ( obj , xpos , ypos )</i> Verschiebt den Dialog <b>obj</b> zur angegebenen Koordinate ( <b>xpos,ypos</b> ) an.
<b>j_setresizable</b>	<i>procedure j_setresizable ( obj , resizable )</i> Ist <b>resizable</b> <b>JFALSE</b> , kann die Größe des Dialog nicht vom Benutzer verändert werden.
<b>j_setsize</b>	<i>procedure j_setsize ( obj , width , height )</i> Setzt Breite und Höhe des Dialog <b>obj</b> auf <b>width</b> and <b>height</b> .
<b>j_settext</b>	<i>procedure j_settext ( obj , str\$ )</i> Ändert den Textinhalt des Dialog <b>obj</b> auf <b>str</b>
<b>j_setvgap</b>	<i>procedure j_setvgap ( obj , vgap )</i> Setzt den vertikalen Abstand innerhalb des Dialog <b>obj</b> auf <b>vgap</b> Pixel.
<b>j_sevensegment</b>	<i>function j_sevensegment ( obj , color )</i> Erzeugt eine Siebensegment Anzeige der Farbe <b>color</b> .
<b>j_show</b>	<i>procedure j_show ( obj )</i> Zeigt den Dialog <b>obj</b> auf dem Bildschirm an.
<b>j_textarea</b>	<i>function j_textarea ( obj , rows , columns )</i> Erzeugt eine neue Textarea und gibt dessen Eventnummer zurück.

- j\_textfield**      *function j\_textfield ( obj , columns )*  
Erzeugt eine neues Textfield und gibt dessen Eventnummer zurück.
- j\_vscrollbar**      *function j\_vscrollbar ( obj )*  
Erzeugt eine vertikale Scrollbar und gibt dessen Eventnummer zurück.
- j\_windowlistener**      *function j\_windowlistener ( window , kind )*  
Bindet an den Dialog **window** einen neuen Windowlistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.

Focuslistener
---------------

- j\_focuslistener**      *function j\_focuslistener ( obj )*  
Bindet an den Focuslistener **obj** einen neuen Focuslistener, und liefert dessen Eventnummer zurück.
- j\_dispose**            *procedure j\_dispose ( obj )*  
der Focuslistener **obj** wird mit allen Recourcen gelöscht.
- j\_hasfocus**         *function j\_hasfocus ( obj )*  
Liefert J\_TRUE zurück, falls der Focuslistener den Focus besitzt, sonst J\_FALSE .

Frame
-------

- j\_frame**            *function j\_frame ( label\$ )*  
Erzeugt einen Frame mit der Titelzeile **label**.
- j\_add**                *procedure j\_add ( obj , cont )*  
Fügt den Frame **obj** in den Container **cont** ein.
- j\_alertbox**        *procedure j\_alertbox ( obj , title\$ , text\$ , button\$ )*  
Öffnet eine Alertbox mit der Überschrift **title**, Textinhalt **text** und einem Button.
- j\_borderpanel**    *function j\_borderpanel ( obj , type )*  
Erzeugt ein neues Borderpanel und liefert dessen Eventnummer zurück.
- j\_button**            *function j\_button ( obj , label\$ )*  
Erzeugt einen neuen Button mit dem Textinhalt **label**
- j\_canvas**            *function j\_canvas ( obj , width , height )*  
Erzeugt einen neuen Canvas mit der Breite **width** Pixel und einer Höhe von **height** Pixel.
- j\_checkbox**        *function j\_checkbox ( obj , label\$ )*  
Erzeugt eine Checkbox mit dem Textinhalt **label** und liefert dessen Eventnummer zurück.
- j\_choicebox2**      *procedure j\_choicebox2 ( obj , title\$ , text\$ , button1\$ , button2\$ )*  
Öffnet eine Choicebox mit der Überschrift **title**, Textinhalt **text** und zwei Buttons.
- j\_choicebox3**      *procedure j\_choicebox3 ( obj , title\$ , text\$ , button1\$ , button2\$ , button3\$ )*  
Öffnet eine Choicebox mit der Überschrift **title**, Textinhalt **text** und drei Buttons.
- j\_choice**            *function j\_choice ( obj )*  
Erzeugt eine neue Choice und gibt deren Eventnummer zurück.
- j\_componentlistener** *function j\_componentlistener ( obj , kind )*  
Bindet an den Frame **obj** einen neuen Componentlistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.
- j\_dialog**            *function j\_dialog ( obj , label\$ )*  
Erzeugt ein neues Dialogwindow mit dem Titel **label** und liefert dessen Eventnummer zurück.
- j\_disable**            *procedure j\_disable ( obj )*  
disabled den Frame **obj**, sodaß diese auf keine Benutzereingaben mehr reagiert.

<b>j_dispose</b>	<i>procedure j_dispose ( obj )</i> der Frame <b>obj</b> wird mit allen Recourcen gelöscht.
<b>j_enable</b>	<i>procedure j_enable ( obj )</i> enabled den Frame <b>obj</b> .
<b>j_filedialog</b>	<i>function\$ j_filedialog ( frame , title\$ , directory\$ )</i> Öffnet eine Filedialogbox in dem angegebenen <b>directory</b> mit dem Title <b>title</b> und liefert den ausgewählten Dateiname in <b>filename</b> zurück.
<b>j_fileselect</b>	<i>function\$ j_fileselect ( frame , title\$ , filter\$ )</i> Öffnet eine Fileslectorbox mit dem voreingestellten <b>filename</b> und dem Title <b>title</b> und liefert den ausgewählten Dateiname in <b>filename</b> zurück.
<b>j_focuslistener</b>	<i>function j_focuslistener ( obj )</i> Bindet an den Frame <b>obj</b> einen neuen Focuslistener, und liefert dessen Eventnummer zurück.
<b>j_getfontascent</b>	<i>function j_getfontascent ( obj )</i> Liefert die Oberlänge (Ascent) des aktuellen Fonts des Frame <b>obj</b> in Bildschirmpixeln zurück.
<b>j_getfontheight</b>	<i>function j_getfontheight ( obj )</i> Liefert die totale Höhe des aktuellen Fonts des Frame <b>obj</b> in Bildschirmpixeln zurück.
<b>j_getheight</b>	<i>function j_getheight ( obj )</i> Liefert die Höhe des Frame <b>obj</b> zurück.
<b>j_getinheight</b>	<i>function j_getinheight ( cont )</i> Liefert die Höhe des Frame <b>obj</b> des inneren Bereichs zurück.
<b>j_getinsets</b>	<i>function j_getinsets ( obj , side )</i> Liefert die angeforderte Breite des inneren Randes dem Frame zurück.
<b>j_getinwidth</b>	<i>function j_getinwidth ( cont )</i> Liefert die Breite des Frame <b>obj</b> des inneren Bereichs zurück.
<b>j_getlayoutid</b>	<i>function j_getlayoutid ( obj )</i> Liefert die Eventnummer des Layoutmanagers des Containers <b>obj</b> zurück.
<b>j_getlength</b>	<i>function j_getlength ( obj )</i> Liefert die Länge des Textinhaltes den Frame zurück.
<b>j_getparentid</b>	<i>function j_getparentid ( obj )</i> Liefert die Eventnummer des umgebenen Containers von <b>obj</b> zurück. Ist <b>obj</b> ein Frame wird $-1$ zurückgegeben.
<b>j_getparent</b>	<i>function j_getparent ( obj )</i> Liefert die Eventnummer des umgebenen Containers von <b>obj</b> zurück. Ist <b>obj</b> ein Frame wird $-1$ zurückgegeben.
<b>j_getstringwidth</b>	<i>function j_getstringwidth ( obj , str\$ )</i>



	Liefert die Länge des <b>str</b> des aktuellen Fonts des Frame <b>obj</b> in Bildschirmpixeln zurück.
<b>j_gettext</b>	<i>function j_gettext ( obj )</i> Liefert den Textinhalt des Frame .
<b>j_getwidth</b>	<i>function j_getwidth ( obj )</i> Liefert die Breite des Frame <b>obj</b> zurück.
<b>j_getxpos</b>	<i>function j_getxpos ( obj )</i> Liefert die aktuelle horizontale Position des Frame <b>obj</b> zurück.
<b>j_getypos</b>	<i>function j_getypos ( obj )</i> Liefert die aktuelle vertikale Position des Frame <b>obj</b> zurück.
<b>j_graphicbutton</b>	<i>function j_graphicbutton ( obj , filename\$ )</i> Erzeugt einen neuen Graphicbutton mit dem Bildinhalt des Files, das durch <b>filename</b> spezifiziert wird.
<b>j_graphiclabel</b>	<i>function j_graphiclabel ( obj , str\$ )</i> Erzeugt ein neues Graphiclabel mit dem Bildinhalt des Files, das durch <b>filename</b> spezifiziert wird.
<b>j_hide</b>	<i>procedure j_hide ( obj )</i> Zeigt den Frame <b>obj</b> nicht mehr auf dem Bildschirm an.
<b>j_hscrollbar</b>	<i>function j_hscrollbar ( obj )</i> Erzeugt eine horizontale Scrollbar und gibt dessen Eventnummer zurück.
<b>j_isparent</b>	<i>function j_isparent ( obj , cont )</i> Liefert J_TRUE zurück, falls das <b>cont</b> die umgebende Komponente von <b>obj</b> ist, sonst J_FALSE
<b>j_isresizable</b>	<i>function j_isresizable ( obj )</i> Liefert J_FALSE zurueck, falls der Frame eine fixe Größe besitzt, J_TRUE sonst.
<b>j_isvisible</b>	<i>function j_isvisible ( obj )</i> Liefert J_TRUE zurück, falls <b>obj</b> sichtbar ist, sonst J_FALSE
<b>j_keylistener</b>	<i>function j_keylistener ( obj )</i> Bindet an den Frame <b>obj</b> einen neuen Keylistener, und liefert dessen Eventnummer zurück.
<b>j_label</b>	<i>function j_label ( obj , label\$ )</i> Erzeugt ein neues Label Element mit dem Textinhalt <b>label</b>
<b>j_led</b>	<i>function j_led ( obj , style , color )</i> Erzeugt eine LED der Farbe <b>color</b> und der Form <b>style</b>
<b>j_line</b>	<i>function j_line ( obj , orient , style , length )</i> Erzeugt ein neues Linienelement und liefert dessen Eventnummer zurück

<b>j_list</b>	<i>function j_list ( obj , rows )</i> Erzeugt eine neue List und gibt deren Eventnummer zurück.
<b>j_menubar</b>	<i>function j_menubar ( obj )</i> Erzeugt eine neue Menubar in <b>obj</b> und liefert dessen Eventnummer zurück.
<b>j_messagebox</b>	<i>procedure j_messagebox ( obj , title\$ , text\$ )</i> Öffnet eine Messagebox mit der Überschrift <b>title</b> und dem Textinhalt <b>text</b>
<b>j_meter</b>	<i>function j_meter ( obj , title\$ )</i> Erzeugt ein analoges Anzeigeelement mit der Beschriftung <b>title</b> .
<b>j_mouselistener</b>	<i>function j_mouselistener ( obj , kind )</i> Bindet an den Frame <b>obj</b> einen neuen Mouselistener, und liefert dessen Eventnummer zurück. Der Parameter <b>kind</b> bestimmt, wann ein Event ausgelöst wird.
<b>j_pack</b>	<i>procedure j_pack ( obj )</i> Veranlaßt den Layoutmanager von <b>obj</b> die minimale Größe zu berechnen.
<b>j_panel</b>	<i>function j_panel ( obj )</i> Erzeugt ein neues panel und liefert dessen Eventnummer zurück.
<b>j_popupmenu</b>	<i>function j_popupmenu ( obj , label\$ )</i> Erzeugt ein neues Popupmenu mit dem Inhalt <b>label</b> und liefert dessen Eventnummer zurück.
<b>j_printer</b>	<i>function j_printer ( frame )</i> Erzeugt ein neues Objekt, das eine Papierseite des Druckes repräsentiert.
<b>j_print</b>	<i>procedure j_print ( obj )</i> Druckt den Inhalt des Frame .
<b>j_progressbar</b>	<i>function j_progressbar ( obj , orient )</i> Erzeugt eine Progress Bar mit der Ausrichtung <b>orient</b> .
<b>j_radiogroup</b>	<i>function j_radiogroup ( obj )</i> Erzeugt eine neue Radiogroup und liefert dessen Eventnummer zurück.
<b>j_releaseall</b>	<i>procedure j_releaseall ( obj )</i> Gibt alle Elemente aus dem Frame <b>obj</b> frei.
<b>j_release</b>	<i>procedure j_release ( obj )</i> Gibt den Frame <b>obj</b> aus dem umgebenden Container Element frei.
<b>j_scrollpane</b>	<i>function j_scrollpane ( obj )</i> Erzeugt eine neue Scrollpane und gibt dessen Eventnummer zurück.
<b>j_setalign</b>	<i>procedure j_setalign ( obj , align )</i> Setzt die Ausrichtung innerhalb des Frame <b>obj</b> auf <b>align</b> . Ein Flowlayout Manager wird vorausgesetzt.
<b>j_setborderlayout</b>	<i>procedure j_setborderlayout ( obj )</i>

Integriert einen BorderLayout Manager in den Frame **obj**.

- j\_setborderpos**      *procedure j\_setborderpos ( obj , pos )*  
Positioniert den Frame **obj** an die angegebene Position **pos**. Im umgebenden Container muß einen BorderLayout Manager eingestellt sein.
- j\_setcolorbg**      *procedure j\_setcolorbg ( obj , r , g , b )*  
Setzt die Hintergrundfarbe auf den übergebenen RGB Wert
- j\_setcolor**      *procedure j\_setcolor ( obj , r , g , b )*  
Setzt die Vordergrundfarbe auf den übergebenen RGB Wert
- j\_setcursor**      *function j\_setcursor ( obj , cursor )*  
Setzt den Cursor des Frame **obj** auf **cursor**.
- j\_setfixlayout**      *procedure j\_setfixlayout ( obj )*  
Integriert einen Fixlayout Manager in den Frame **obj** (voreingestellte Layoutmanager).
- j\_setflowfill**      *procedure j\_setflowfill ( obj , bool )*  
Setzt die Höhe (Breite) aller enthaltenen Komponenten auf die Höhe (Breite) von dem Frame **obj**. Ein Flowlayout Manager wird vorausgesetzt.
- j\_setflowlayout**      *procedure j\_setflowlayout ( obj , align )*  
Integriert einen Flowlayout Manager in den Frame **obj** mit der Ausrichtung **align**.
- j\_setfocus**      *function j\_setfocus ( obj )*  
Weist den Input Focus dem Frame **obj** zu.
- j\_setfontname**      *procedure j\_setfontname ( obj , name )*  
Setzt den Fonttyp des Frame **obj** auf **name**.
- j\_setfont**      *procedure j\_setfont ( obj , name , style , size )*  
Setzt den Font des Frame **obj** entsprechend der Parameterliste.
- j\_setfontsize**      *procedure j\_setfontsize ( obj , size )*  
Setzt den Fontgröße des Frame **obj** auf **size**.
- j\_setfontstyle**      *procedure j\_setfontstyle ( obj , style )*  
Setzt den Fontstyle des Frame **obj** auf **style**.
- j\_setgridlayout**      *procedure j\_setgridlayout ( obj , row , col )*  
Integriert einen Gridlayout Manager in den Frame **obj** mit **row** Reihen und **col** Spalten.
- j\_sethgap**      *procedure j\_sethgap ( obj , hgap )*  
Setzt den horizontalen Abstand innerhalb des Frame **obj** auf **hgap** Pixel.
- j\_seticon**      *procedure j\_seticon ( frame , icon )*  
Ordnet dem **frame** das Icon **icon** zu. Eine iconisierte Darstellung wird nicht von allen Plattformen unterstützt.

<b>j_setinsets</b>	<i>procedure j_setinsets ( obj , top , bottom , left , right )</i> Setzt den inneren Rahmen auf die angegebenen Größen.
<b>j_setnamedcolorbg</b>	<i>procedure j_setnamedcolorbg ( obj , color )</i> Setzt die Hintergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnamedcolor</b>	<i>procedure j_setnamedcolor ( obj , color )</i> Setzt die Vordergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnolayout</b>	<i>procedure j_setnolayout ( obj )</i> Entfernt den aktuellen Layout Manager aus dem Frame <b>obj</b> .
<b>j_setpos</b>	<i>procedure j_setpos ( obj , xpos , ypos )</i> Verschiebt den Frame <b>obj</b> zur angegebenen Koordinate ( <b>xpos,ypos</b> ) an.
<b>j_setresizable</b>	<i>procedure j_setresizable ( obj , resizable )</i> Ist <b>resizable</b> <b>J.FALSE</b> , kann die Größe des Frame nicht vom Benutzer verändert werden.
<b>j_setsize</b>	<i>procedure j_setsize ( obj , width , height )</i> Setzt Breite und Höhe des Frame <b>obj</b> auf <b>width</b> and <b>height</b> .
<b>j_settext</b>	<i>procedure j_settext ( obj , str\$ )</i> Ändert den Textinhalt des Frame <b>obj</b> auf <b>str</b>
<b>j_setvgap</b>	<i>procedure j_setvgap ( obj , vgap )</i> Setzt den vertikalen Abstand innerhalb des Frame <b>obj</b> auf <b>vgap</b> Pixel.
<b>j_sevensegment</b>	<i>function j_sevensegment ( obj , color )</i> Erzeugt eine Siebensegment Anzeige der Farbe <b>color</b> .
<b>j_show</b>	<i>procedure j_show ( obj )</i> Zeigt den Frame <b>obj</b> auf dem Bildschirm an.
<b>j_textarea</b>	<i>function j_textarea ( obj , rows , columns )</i> Erzeugt eine neue Textarea und gibt dessen Eventnummer zurück.
<b>j_textfield</b>	<i>function j_textfield ( obj , columns )</i> Erzeugt eine neues Textfield und gibt dessen Eventnummer zurück.
<b>j_vscrollbar</b>	<i>function j_vscrollbar ( obj )</i> Erzeugt eine vertikale Scrollbar und gibt dessen Eventnummer zurück.
<b>j_windowlistener</b>	<i>function j_windowlistener ( window , kind )</i> Bindet an den Frame <b>window</b> einen neuen Windowlistener, und liefert dessen Eventnummer zurück. Der Parameter <b>kind</b> bestimmt, wann ein Event ausgelöst wird.
<b>j_window</b>	<i>function j_window ( obj )</i> Erzeugt ein neues Window und liefert dessen Eventnummer zurück.

Helpmenu
----------

<b>j_helpmenu</b>	<p><i>function j_helpmenu ( obj , label\$ )</i>          Erzeugt ein Hilfe-Menueintrag mit der Beschriftung <b>label</b> und liefert dessen Eventnummer zurück.</p>
<b>j_checkmenuitem</b>	<p><i>function j_checkmenuitem ( obj , label\$ )</i>          Erzeugt ein Checkmenuitem mit der Beschriftung <b>label</b> und liefert dessen Eventnummer zurück.</p>
<b>j_disable</b>	<p><i>procedure j_disable ( obj )</i>          disabled das Helpmenu <b>obj</b>, sodaß diese auf keine Benutzereingaben mehr reagiert.</p>
<b>j_dispose</b>	<p><i>procedure j_dispose ( obj )</i>          das Helpmenu <b>obj</b> wird mit allen Recourcen gelöscht.</p>
<b>j_enable</b>	<p><i>procedure j_enable ( obj )</i>          enabled das Helpmenu <b>obj</b>.</p>
<b>j_getlength</b>	<p><i>function j_getlength ( obj )</i>          Liefert die Länge des Textinhaltes das Helpmenu zurück.</p>
<b>j_gettext</b>	<p><i>function\$ j_gettext ( obj )</i>          Liefert den Textinhalt des Helpmenu .</p>
<b>j_menuitem</b>	<p><i>function j_menuitem ( obj , label\$ )</i>          Erzeugt ein Menuitem mit der Beschriftung <b>label</b> und liefert dessen Eventnummer zurück.</p>
<b>j_seperator</b>	<p><i>procedure j_seperator ( obj )</i>          Erzeugt eine Trennlinie in das Helpmenu .</p>
<b>j_setfontname</b>	<p><i>procedure j_setfontname ( obj , name )</i>          Setzt den Fonttyp des Helpmenu <b>obj</b> auf <b>name</b>.</p>
<b>j_setfont</b>	<p><i>procedure j_setfont ( obj , name , style , size )</i>          Setzt den Font des Helpmenu <b>obj</b> entsprechend der Parameterliste.</p>
<b>j_setfontsize</b>	<p><i>procedure j_setfontsize ( obj , size )</i>          Setzt den Fontgröße des Helpmenu <b>obj</b> auf <b>size</b>.</p>
<b>j_setfontstyle</b>	<p><i>procedure j_setfontstyle ( obj , style )</i>          Setzt den Fontstyle des Helpmenu <b>obj</b> auf <b>style</b>.</p>
<b>j_setshortcut</b>	<p><i>procedure j_setshortcut ( obj , chr )</i>          Bestimmt das Zeichen <b>chr</b> als Shortcut für das Helpmenu .</p>
<b>j_settext</b>	<p><i>procedure j_settext ( obj , str\$ )</i></p>

Ändert den Textinhalt des Helpmenu **obj** auf **str**

Hscrollbar
------------

- j\_hscrollbar**      *function j\_hscrollbar ( obj )*  
Erzeugt eine horizontale Scrollbar und gibt dessen Eventnummer zurück.
- j\_add**              *procedure j\_add ( obj , cont )*  
Fügt die Hscrollbar **obj** in den Container **cont** ein.
- j\_componentlistener** *function j\_componentlistener ( obj , kind )*  
Bindet an die Hscrollbar **obj** einen neuen Componentlistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.
- j\_disable**          *procedure j\_disable ( obj )*  
disabled die Hscrollbar **obj**, sodaß diese auf keine Benutzereingaben mehr reagiert.
- j\_dispose**          *procedure j\_dispose ( obj )*  
die Hscrollbar **obj** wird mit allen Recourcen gelöscht.
- j\_enable**            *procedure j\_enable ( obj )*  
enabled die Hscrollbar **obj**.
- j\_focuslistener**    *function j\_focuslistener ( obj )*  
Bindet an die Hscrollbar **obj** einen neuen Focuslistener, und liefert dessen Eventnummer zurück.
- j\_getfontascent**    *function j\_getfontascent ( obj )*  
Liefert die Oberlänge (Ascent) des aktuellen Fonts der Hscrollbar **obj** in Bildschirmpixeln zurück.
- j\_getfontheight**    *function j\_getfontheight ( obj )*  
Liefert die totale Höhe des aktuellen Fonts der Hscrollbar **obj** in Bildschirmpixeln zurück.
- j\_getheight**        *function j\_getheight ( obj )*  
Liefert die Höhe der Hscrollbar **obj** zurück.
- j\_getparentid**      *function j\_getparentid ( obj )*  
Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird  $-1$  zurückgegeben.
- j\_getparent**        *function j\_getparent ( obj )*  
Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird  $-1$  zurückgegeben.
- j\_getstringwidth**   *function j\_getstringwidth ( obj , str\$ )*  
Liefert die Länge des **str** des aktuellen Fonts der Hscrollbar **obj** in Bildschirmpixeln zurück.

<b>j_getvalue</b>	<i>function j_getvalue ( obj )</i> Liefert den aktuellen Wert der Hscrollbar .
<b>j_getwidth</b>	<i>function j_getwidth ( obj )</i> Liefert die Breite der Hscrollbar <b>obj</b> zurück.
<b>j_getxpos</b>	<i>function j_getxpos ( obj )</i> Liefert die aktuelle horizontale Position der Hscrollbar <b>obj</b> zurück.
<b>j_getypos</b>	<i>function j_getypos ( obj )</i> Liefert die aktuelle vertikale Position der Hscrollbar <b>obj</b> zurück.
<b>j_hide</b>	<i>procedure j_hide ( obj )</i> Zeigt die Hscrollbar <b>obj</b> nicht mehr auf dem Bildschirm an.
<b>j_isparent</b>	<i>function j_isparent ( obj , cont )</i> Liefert J_TRUE zurück, falls das <b>cont</b> die umgebende Komponente von <b>obj</b> ist, sonst J_FALSE
<b>j_isvisible</b>	<i>function j_isvisible ( obj )</i> Liefert J_TRUE zurück, falls <b>obj</b> sichtbar ist, sonst J_FALSE
<b>j_keylistener</b>	<i>function j_keylistener ( obj )</i> Bindet an die Hscrollbar <b>obj</b> einen neuen Keylistener, und liefert dessen Eventnummer zurück.
<b>j_mouselistener</b>	<i>function j_mouselistener ( obj , kind )</i> Bindet an die Hscrollbar <b>obj</b> einen neuen Mouselistener, und liefert dessen Eventnummer zurück. Der Parameter <b>kind</b> bestimmt, wann ein Event ausgelöst wird.
<b>j_popupmenu</b>	<i>function j_popupmenu ( obj , label\$ )</i> Erzeugt ein neues Popupmenu mit dem Inhalt <b>label</b> und liefert dessen Eventnummer zurück.
<b>j_print</b>	<i>procedure j_print ( obj )</i> Druckt den Inhalt der Hscrollbar .
<b>j_release</b>	<i>procedure j_release ( obj )</i> Gibt die Hscrollbar <b>obj</b> aus dem umgebenden Container Element frei.
<b>j_setblockinc</b>	<i>function j_setblockinc ( obj , val )</i> Stellt die Blockänderung von der Hscrollbar auf den Wert von <b>val</b> ein.
<b>j_setborderpos</b>	<i>procedure j_setborderpos ( obj , pos )</i> Positioniert die Hscrollbar <b>obj</b> an die angegebene Position <b>pos</b> . Im umgebenden Container muß ein BorderLayout Manager eingestellt sein.
<b>j_setcolorbg</b>	<i>procedure j_setcolorbg ( obj , r , g , b )</i> Setzt die Hintergrundfarbe auf den übergebenen RGB Wert
<b>j_setcolor</b>	<i>procedure j_setcolor ( obj , r , g , b )</i>



	Setzt die Vordergrundfarbe auf den übergebenen RGB Wert
<b>j_setcursor</b>	<i>function j_setcursor ( obj , cursor )</i> Setzt den Cursor der Hscrollbar <b>obj</b> auf <b>cursor</b> .
<b>j_setfocus</b>	<i>function j_setfocus ( obj )</i> Weist den Input Focus der Hscrollbar <b>obj</b> zu.
<b>j_setfontname</b>	<i>procedure j_setfontname ( obj , name )</i> Setzt den Fonttyp der Hscrollbar <b>obj</b> auf <b>name</b> .
<b>j_setfont</b>	<i>procedure j_setfont ( obj , name , style , size )</i> Setzt den Font der Hscrollbar <b>obj</b> entsprechend der Parameterliste.
<b>j_setfontsize</b>	<i>procedure j_setfontsize ( obj , size )</i> Setzt den Fontgröße der Hscrollbar <b>obj</b> auf <b>size</b> .
<b>j_setfontstyle</b>	<i>procedure j_setfontstyle ( obj , style )</i> Setzt den Fontstyle der Hscrollbar <b>obj</b> auf <b>style</b> .
<b>j_setmax</b>	<i>function j_setmax ( obj , val )</i> Stellt die obere Grenze der Hscrollbar auf den Wert von <b>val</b> ein.
<b>j_setmin</b>	<i>function j_setmin ( obj , val )</i> Stellt die untere Grenze der Hscrollbar auf den Wert von <b>val</b> ein.
<b>j_setnamedcolorbg</b>	<i>procedure j_setnamedcolorbg ( obj , color )</i> Setzt die Hintergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnamedcolor</b>	<i>procedure j_setnamedcolor ( obj , color )</i> Setzt die Vordergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setpos</b>	<i>procedure j_setpos ( obj , xpos , ypos )</i> Verschiebt die Hscrollbar <b>obj</b> zur angegebenen Koordinate ( <b>xpos,ypos</b> ) an.
<b>j_setsize</b>	<i>procedure j_setsize ( obj , width , height )</i> Setzt Breite und Höhe der Hscrollbar <b>obj</b> auf <b>width</b> and <b>height</b> .
<b>j_setslidesize</b>	<i>function j_setslidesize ( obj , val )</i> Setzt die Größe des Schiebers auf den Wert von <b>val</b> .
<b>j_setunitinc</b>	<i>function j_setunitinc ( obj , val )</i> Stellt die minimale Wertänderung von die Hscrollbar auf den Wert von <b>val</b> ein.
<b>j_setvalue</b>	<i>procedure j_setvalue ( obj , val )</i> Stellt den aktuellen Wert der Hscrollbar <b>obj</b> auf <b>val</b> .
<b>j_show</b>	<i>procedure j_show ( obj )</i> Zeigt die Hscrollbar <b>obj</b> auf dem Bildschirm an.

Graphicbutton
---------------

- j\_graphicbutton**     *function j\_graphicbutton ( obj , filename\$ )*  
 Erzeugt einen neuen Graphicbutton mit dem Bildinhalt des Files, das durch **filename** spezifiziert wird.
- j\_add**                 *procedure j\_add ( obj , cont )*  
 Fügt den Graphicbutton **obj** in den Container **cont** ein.
- j\_componentlistener** *function j\_componentlistener ( obj , kind )*  
 Bindet an den Graphicbutton **obj** einen neuen Componentlistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.
- j\_disable**             *procedure j\_disable ( obj )*  
 disabled den Graphicbutton **obj**, sodaß diese auf keine Benutzereingaben mehr reagiert.
- j\_dispose**            *procedure j\_dispose ( obj )*  
 der Graphicbutton **obj** wird mit allen Recourcen gelöscht.
- j\_enable**              *procedure j\_enable ( obj )*  
 enabled den Graphicbutton **obj**.
- j\_focuslistener**     *function j\_focuslistener ( obj )*  
 Bindet an den Graphicbutton **obj** einen neuen Focuslistener, und liefert dessen Eventnummer zurück.
- j\_getfontascent**     *function j\_getfontascent ( obj )*  
 Liefert die Oberlänge (Ascent) des aktuellen Fonts des Graphicbutton **obj** in Bildschirmpixeln zurück.
- j\_getfontheight**     *function j\_getfontheight ( obj )*  
 Liefert die totale Höhe des aktuellen Fonts des Graphicbutton **obj** in Bildschirmpixeln zurück.
- j\_getheight**         *function j\_getheight ( obj )*  
 Liefert die Höhe des Graphicbutton **obj** zurück.
- j\_getparentid**       *function j\_getparentid ( obj )*  
 Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird  $-1$  zurückgegeben.
- j\_getparent**         *function j\_getparent ( obj )*  
 Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird  $-1$  zurückgegeben.
- j\_getstringwidth**   *function j\_getstringwidth ( obj , str\$ )*

Liefert die Länge des **str** des aktuellen Fonts des Graphicbutton **obj** in Bildschirmpixeln zurück.

- j\_getwidth**      *function j\_getwidth ( obj )*  
Liefert die Breite des Graphicbutton **obj** zurück.
- j\_getxpos**      *function j\_getxpos ( obj )*  
Liefert die aktuelle horizontale Position des Graphicbutton **obj** zurück.
- j\_getypos**      *function j\_getypos ( obj )*  
Liefert die aktuelle vertikale Position des Graphicbutton **obj** zurück.
- j\_hide**          *procedure j\_hide ( obj )*  
Zeigt den Graphicbutton **obj** nicht mehr auf dem Bildschirm an.
- j\_isparent**      *function j\_isparent ( obj , cont )*  
Liefert J\_TRUE zurück, falls das **cont** die umgebende Komponente von **obj** ist, sonst J\_FALSE
- j\_isvisible**      *function j\_isvisible ( obj )*  
Liefert J\_TRUE zurück, falls **obj** sichtbar ist, sonst J\_FALSE
- j\_keylistener**    *function j\_keylistener ( obj )*  
Bindet an den Graphicbutton **obj** einen neuen Keylistener, und liefert dessen Eventnummer zurück.
- j\_mouselistener** *function j\_mouselistener ( obj , kind )*  
Bindet an den Graphicbutton **obj** einen neuen Mouselistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.
- j\_popupmenu**    *function j\_popupmenu ( obj , label\$ )*  
Erzeugt ein neues Popupmenu mit dem Inhalt **label** und liefert dessen Eventnummer zurück.
- j\_print**          *procedure j\_print ( obj )*  
Druckt den Inhalt des Graphicbutton .
- j\_release**        *procedure j\_release ( obj )*  
Gibt den Graphicbutton **obj** aus dem umgebenden Container Element frei.
- j\_setborderpos**   *procedure j\_setborderpos ( obj , pos )*  
Positioniert den Graphicbutton **obj** an die angegebene Position **pos**. Im umgebenden Container muß einen Borderlayout Manager eingestellt sein.
- j\_setcolorbg**     *procedure j\_setcolorbg ( obj , r , g , b )*  
Setzt die Hintergrundfarbe auf den übergebenen RGB Wert
- j\_setcolor**       *procedure j\_setcolor ( obj , r , g , b )*  
Setzt die Vordergrundfarbe auf den übergebenen RGB Wert
- j\_setcursor**      *function j\_setcursor ( obj , cursor )*  
Setzt den Cursor des Graphicbutton **obj** auf **cursor**.

<b>j_setfocus</b>	<i>function j_setfocus ( obj )</i> Weist den Input Focus dem Graphicbutton <b>obj</b> zu.
<b>j_setfontname</b>	<i>procedure j_setfontname ( obj , name )</i> Setzt den Fonttyp des Graphicbutton <b>obj</b> auf <b>name</b> .
<b>j_setfont</b>	<i>procedure j_setfont ( obj , name , style , size )</i> Setzt den Font des Graphicbutton <b>obj</b> entsprechend der Parameterliste.
<b>j_setfontsize</b>	<i>procedure j_setfontsize ( obj , size )</i> Setzt den Fontgröße des Graphicbutton <b>obj</b> auf <b>size</b> .
<b>j_setfontstyle</b>	<i>procedure j_setfontstyle ( obj , style )</i> Setzt den Fontstyle des Graphicbutton <b>obj</b> auf <b>style</b> .
<b>j_setimage</b>	<i>procedure j_setimage ( obj , image )</i> Ordnet dem <b>obj</b> das Bild <b>image</b> zu.
<b>j_setnamedcolorbg</b>	<i>procedure j_setnamedcolorbg ( obj , color )</i> Setzt die Hintergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnamedcolor</b>	<i>procedure j_setnamedcolor ( obj , color )</i> Setzt die Vordergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setpos</b>	<i>procedure j_setpos ( obj , xpos , ypos )</i> Verschiebt den Graphicbutton <b>obj</b> zur angegebenen Koordinate ( <b>xpos,ypos</b> ) an.
<b>j_setsize</b>	<i>procedure j_setsize ( obj , width , height )</i> Setzt Breite und Höhe des Graphicbutton <b>obj</b> auf <b>width</b> and <b>height</b> .
<b>j_show</b>	<i>procedure j_show ( obj )</i> Zeigt den Graphicbutton <b>obj</b> auf dem Bildschirm an.

## Graphiclabel

- j\_graphiclabel**      *function j\_graphiclabel ( obj , str\$ )*  
Erzeugt ein neues Graphiclabel mit dem Bildinhalt des Files, das durch **filena-**  
**me** spezifiziert wird.
- j\_add**                      *procedure j\_add ( obj , cont )*  
Fügt das Graphiclabel **obj** in den Container **cont** ein.
- j\_componentlistener** *function j\_componentlistener ( obj , kind )*  
Bindet an das Graphiclabel **obj** einen neuen Componentlistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.
- j\_disable**                      *procedure j\_disable ( obj )*  
disabled das Graphiclabel **obj**, sodaß diese auf keine Benutzereingaben mehr reagiert.
- j\_dispose**                      *procedure j\_dispose ( obj )*  
das Graphiclabel **obj** wird mit allen Recourcen gelöscht.
- j\_enable**                      *procedure j\_enable ( obj )*  
enabled das Graphiclabel **obj**.
- j\_focuslistener**              *function j\_focuslistener ( obj )*  
Bindet an das Graphiclabel **obj** einen neuen Focuslistener, und liefert dessen Eventnummer zurück.
- j\_getfontascent**              *function j\_getfontascent ( obj )*  
Liefert die Oberlänge (Ascent) des aktuellen Fonts des Graphiclabel **obj** in Bildschirmpixeln zurück.
- j\_getfontheight**              *function j\_getfontheight ( obj )*  
Liefert die totale Höhe des aktuellen Fonts des Graphiclabel **obj** in Bildschirmpixeln zurück.
- j\_getheight**                      *function j\_getheight ( obj )*  
Liefert die Höhe des Graphiclabel **obj** zurück.
- j\_getparentid**                      *function j\_getparentid ( obj )*  
Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird  $-1$  zurückgegeben.
- j\_getparent**                      *function j\_getparent ( obj )*  
Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird  $-1$  zurückgegeben.
- j\_getstringwidth**              *function j\_getstringwidth ( obj , str\$ )*

Liefert die Länge des **str** des aktuellen Fonts des Graphiclabel **obj** in Bildschirmpixeln zurück.

- j\_getwidth** *function j\_getwidth ( obj )*  
Liefert die Breite des Graphiclabel **obj** zurück.
- j\_getxpos** *function j\_getxpos ( obj )*  
Liefert die aktuelle horizontale Position des Graphiclabel **obj** zurück.
- j\_getypos** *function j\_getypos ( obj )*  
Liefert die aktuelle vertikale Position des Graphiclabel **obj** zurück.
- j\_hide** *procedure j\_hide ( obj )*  
Zeigt das Graphiclabel **obj** nicht mehr auf dem Bildschirm an.
- j\_isparent** *function j\_isparent ( obj , cont )*  
Liefert J\_TRUE zurück, falls das **cont** die umgebende Komponente von **obj** ist, sonst J\_FALSE
- j\_isvisible** *function j\_isvisible ( obj )*  
Liefert J\_TRUE zurück, falls **obj** sichtbar ist, sonst J\_FALSE
- j\_keylistener** *function j\_keylistener ( obj )*  
Bindet an das Graphiclabel **obj** einen neuen Keylistener, und liefert dessen Eventnummer zurück.
- j\_mouselistener** *function j\_mouselistener ( obj , kind )*  
Bindet an das Graphiclabel **obj** einen neuen Mouselistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.
- j\_popupmenu** *function j\_popupmenu ( obj , label\$ )*  
Erzeugt ein neues Popupmenu mit dem Inhalt **label** und liefert dessen Eventnummer zurück.
- j\_print** *procedure j\_print ( obj )*  
Druckt den Inhalt des Graphiclabel .
- j\_release** *procedure j\_release ( obj )*  
Gibt das Graphiclabel **obj** aus dem umgebenden Container Element frei.
- j\_setborderpos** *procedure j\_setborderpos ( obj , pos )*  
Positioniert das Graphiclabel **obj** an die angegebene Position **pos**. Im umgebenden Container muß einen BorderLayout Manager eingestellt sein.
- j\_setcolorbg** *procedure j\_setcolorbg ( obj , r , g , , b )*  
Setzt die Hintergrundfarbe auf den übergebenen RGB Wert
- j\_setcolor** *procedure j\_setcolor ( obj , r , g , , b )*  
Setzt die Vordergrundfarbe auf den übergebenen RGB Wert
- j\_setcursor** *function j\_setcursor ( obj , cursor )*  
Setzt den Cursor des Graphiclabel **obj** auf **cursor**.

<b>j_setfocus</b>	<i>function j_setfocus ( obj )</i> Weist den Input Focus dem Graphiclabel <b>obj</b> zu.
<b>j_setfontname</b>	<i>procedure j_setfontname ( obj , name )</i> Setzt den Fonttyp des Graphiclabel <b>obj</b> auf <b>name</b> .
<b>j_setfont</b>	<i>procedure j_setfont ( obj , name , style , size )</i> Setzt den Font des Graphiclabel <b>obj</b> entsprechend der Parameterliste.
<b>j_setfontsize</b>	<i>procedure j_setfontsize ( obj , size )</i> Setzt den Fontgröße des Graphiclabel <b>obj</b> auf <b>size</b> .
<b>j_setfontstyle</b>	<i>procedure j_setfontstyle ( obj , style )</i> Setzt den Fontstyle des Graphiclabel <b>obj</b> auf <b>style</b> .
<b>j_setimage</b>	<i>procedure j_setimage ( obj , image )</i> Ordnet dem <b>obj</b> das Bild <b>image</b> zu.
<b>j_setnamedcolorbg</b>	<i>procedure j_setnamedcolorbg ( obj , color )</i> Setzt die Hintergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnamedcolor</b>	<i>procedure j_setnamedcolor ( obj , color )</i> Setzt die Vordergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setpos</b>	<i>procedure j_setpos ( obj , xpos , ypos )</i> Verschiebt das Graphiclabel <b>obj</b> zur angegebenen Koordinate ( <b>xpos,ypos</b> ) an.
<b>j_setsize</b>	<i>procedure j_setsize ( obj , width , height )</i> Setzt Breite und Höhe des Graphiclabel <b>obj</b> auf <b>width</b> and <b>height</b> .
<b>j_show</b>	<i>procedure j_show ( obj )</i> Zeigt das Graphiclabel <b>obj</b> auf dem Bildschirm an.

Image
-------

- j\_image**      *function j\_image ( width , height )*  
Erzeugt einen neues (Memory) Image mit der Breite **width** Pixel und einer Höhe von **height** Pixel.
- j\_cliprect**      *procedure j\_cliprect ( obj , x , y , width , height )*  
Setzt den Clipping Bereich auf das spezifizierte Rechteck (**x**, **y**, **width**, **height**).
- j\_dispose**      *procedure j\_dispose ( obj )*  
das Image **obj** wird mit allen Recourcen gelöscht.
- j\_drawarc**      *procedure j\_drawarc ( obj , x , y , rx , ry , arc1 , arc2 )*  
Zeichnet einen Kreisbogen mit dem Mittelpunkt (**x**, **y**) und den Hauptradien **rx** und **ry**. Die Parameter **arc1** und **arc2** bestimmen den Start- bzw. den Endwinkel des Kreisbogens.
- j\_drawcircle**      *procedure j\_drawcircle ( obj , x , y , r )*  
Zeichnet einen Kreis mit dem Mittelpunkt (**x,y**) und dem Radius **r**.
- j\_drawimage**      *procedure j\_drawimage ( obj , image , x , y )*  
Kopiert das Image mit der Eventnummer **image** an die Position (**x**, **y**).
- j\_drawimagesource**      *procedure j\_drawimagesource ( obj , x , y , w , h , r , g , b )*  
Die Funktion stellt ein Bild an der Position (**x,y**) mit der Breite **w** und der Höhe **h** dar. Die Bilddaten werden als rot-, grün- und blau- Kanal in den Arrays **r,g,b** übergeben.
- j\_drawline**      *procedure j\_drawline ( obj , x1 , y1 , x2 , y2 )*  
Zeichnet eine Linie in der momentan gesetzten Vordergrundfarbe von (**x1,y1**) nach (**x2,y2**)
- j\_drawoval**      *procedure j\_drawoval ( obj , x , y , rx , ry )*  
Zeichnet einen Oval mit dem Mittelpunkt (**x,y**) und den Hauptradien **rx** und **ry**.
- j\_drawpixel**      *procedure j\_drawpixel ( obj , x , y )*  
Zeichnet einen Pixel in der momentan gesetzten Vordergrundfarbe an die Position (**x,y**).
- j\_drawpolygon**      *procedure j\_drawpolygon ( obj , len , x , y )*  
Zeichnet ein Polygon durch die Punkte, die durch die Arrays **x** und **y** gegeben sind.
- j\_drawpolyline**      *procedure j\_drawpolyline ( obj , len , x , y )*  
Zeichnet eine Polylinie durch die Punkte, die durch die Arrays **x** und **y** gegeben sind.



- j\_drawrect**      *procedure j\_drawrect ( obj , x , y , width , height )*  
Zeichnet ein Rechteck an die Position **(x,y)** mit der Breite **width** und der Höhe **height**.
- j\_drawroundrect**      *procedure j\_drawroundrect ( obj , x , y , width , height , arcx , arcy )*  
Zeichnet ein Rechteck mit abgerundeten Ecken an die Position **(x,y)** mit der Breite **width** und der Höhe **height**. Die Radien der Ecken werden durch **arcx** und **arcy** bestimmt.
- j\_drawscaledimage**      *procedure j\_drawscaledimage ( obj , image , sx , sy , sw , sh , tx , ty , tw , th )*  
Kopiert einen Ausschnitt des Bildes **image**, der durch die Parameter **(sx, sy, sw, sh)** bestimmt wird an die Position **(tx, ty)** und scaliert den Ausschnitt auf die Breite **tw** und die Höhe **th**
- j\_drawstring**      *procedure j\_drawstring ( obj , x , y , str\$ )*  
Zeichnet den String **str** an die durch die Parameter **(x,y)** festgelegte Koordinate.
- j\_fillarc**      *procedure j\_fillarc ( obj , x , y , rx , ry , arc1 , arc2 )*  
Zeichnet einen gefüllten Kreisbogen mit dem Mittelpunkt **(x,y)** und den Hauptradien **rx** und **ry**. Die Parameter **arc1** und **arc2** bestimmen den Start- bzw. den Endwinkel des Kreisbogens.
- j\_fillcircle**      *procedure j\_fillcircle ( obj , x , y , r )*  
Zeichnet einen gefüllten Kreis mit dem Mittelpunkt **(x,y)** und dem Radius **r**.
- j\_filloval**      *procedure j\_filloval ( obj , x , y , rx , ry )*  
Zeichnet einen gefüllten Oval mit dem Mittelpunkt **(x,y)** und den Hauptradien **rx** und **ry**.
- j\_fillpolygon**      *procedure j\_fillpolygon ( obj , len , x , y )*  
Zeichnet ein gefülltes Polygon durch die Punkte, die durch die Arrays **x** und **y** gegeben sind.
- j\_fillrect**      *procedure j\_fillrect ( obj , x , y , width , height )*  
Zeichnet ein gefülltes Rechteck an die Position **(x,y)** mit der Breite **width** und der Höhe **height**.
- j\_fillroundrect**      *procedure j\_fillroundrect ( obj , x , y , width , height , arcx , arcy )*  
Zeichnet ein gefülltes Rechteck mit abgerundeten Ecken an die Position **(x,y)**. Das Rechteck hat eine Breite von **width** und eine Höhe von **height** Pixeln. Die Form der runden Ecken wird durch die Parameter **arcx** und **arcy** bestimmt.
- j\_getheight**      *function j\_getheight ( obj )*  
Liefert die Höhe des Image **obj** zurück.
- j\_getimage**      *function j\_getimage ( obj )*  
Kopiert den Inhalt dem Image **obj** in ein Image und liefert dessen Eventnummer zurück.
- j\_getimagesource**      *function j\_getimagesource ( obj , x , y , w , h , r , g , b )*

Die Funktion liefert einen Ausschnitt vom Punkt  $(\mathbf{x}, \mathbf{y})$  der Breite  $\mathbf{w}$  und der Höhe  $\mathbf{h}$  zurück. Der Bildinhalt wird als rot-, grün- und blau- Kanal in den Arrays r,g,b abgelegt.

- j\_getscaledimage**     *function j\_getscaledimage ( obj , x , y , sw , sh , tw , th )*  
Kopiert den Inhalt des Ausschnitts von Positio  $(\mathbf{x}, \mathbf{y})$  it der Breite  $\mathbf{sw}$  und der höhe  $\mathbf{sh}$  in ein Image und liefert dessen Eventnummer zurück. Das Image wird dabei auf eine Breite von  $\mathbf{tw}$  und einer Höhe von  $\mathbf{th}$  scaliert.
- j\_getwidth**     *function j\_getwidth ( obj )*  
Liefert die Breite des Image  $\mathbf{obj}$  zurück.
- j\_print**     *procedure j\_print ( obj )*  
Druckt den Inhalt des Image .
- j\_setxor**     *procedure j\_setxor ( obj , bool )*  
Schaltet den Painting Mode auf XOR, falls bool = J\_TRUE ist. Im XOR Modus heben sich zwei gleiche Graphikbefehle auf.
- j\_translate**     *procedure j\_translate ( obj , x , y )*  
Setzt den Ursprung des Koordinatensystems auf  $(\mathbf{x}, \mathbf{y})$ .

## KeyListener

- j\_keylistener**      *function j\_keylistener ( obj )*  
Bindet an den Keylistener **obj** einen neuen Keylistener, und liefert dessen Eventnummer zurück.
- j\_dispose**          *procedure j\_dispose ( obj )*  
der Keylistener **obj** wird mit allen Recourcen gelöscht.
- j\_getkeychar**      *function j\_getkeychar ( obj )*  
Liefert den ASCII Code, der zuletzt gedrückten Taste zurück.
- j\_getkeycode**      *function j\_getkeycode ( obj )*  
Liefert den Keycode, der zuletzt gedrückten Taste zurück.

Label
-------

- j\_label**                    *function j\_label ( obj , label\$ )*  
Erzeugt ein neues Label Element mit dem Textinhalt **label**
- j\_add**                      *procedure j\_add ( obj , cont )*  
Fügt das Label **obj** in den Container **cont** ein.
- j\_componentlistener** *function j\_componentlistener ( obj , kind )*  
Bindet an das Label **obj** einen neuen Componentlistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.
- j\_disable**                 *procedure j\_disable ( obj )*  
disabled das Label **obj**, sodaß diese auf keine Benutzereingaben mehr reagiert.
- j\_dispose**                *procedure j\_dispose ( obj )*  
das Label **obj** wird mit allen Recourcen gelöscht.
- j\_enable**                 *procedure j\_enable ( obj )*  
enabled das Label **obj**.
- j\_focuslistener**         *function j\_focuslistener ( obj )*  
Bindet an das Label **obj** einen neuen Focuslistener, und liefert dessen Eventnummer zurück.
- j\_getfontascent**         *function j\_getfontascent ( obj )*  
Liefert die Oberlänge (Ascent) des aktuellen Fonts des Label **obj** in Bildschirm-pixeln zurück.
- j\_getfontheight**         *function j\_getfontheight ( obj )*  
Liefert die totale Höhe des aktuellen Fonts des Label **obj** in Bildschirm-pixeln zurück.
- j\_getheight**             *function j\_getheight ( obj )*  
Liefert die Höhe des Label **obj** zurück.
- j\_getparentid**           *function j\_getparentid ( obj )*  
Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird  $-1$  zurückgegeben.
- j\_getparent**             *function j\_getparent ( obj )*  
Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird  $-1$  zurückgegeben.
- j\_getstringwidth**       *function j\_getstringwidth ( obj , str\$ )*  
Liefert die Länge des **str** des aktuellen Fonts des Label **obj** in Bildschirm-pixeln zurück.

<b>j_gettext</b>	<i>function</i> \$ <i>j_gettext</i> ( <i>obj</i> ) Liefert den Textinhalt des Label .
<b>j_getwidth</b>	<i>function</i> <i>j_getwidth</i> ( <i>obj</i> ) Liefert die Breite des Label <b>obj</b> zurück.
<b>j_getxpos</b>	<i>function</i> <i>j_getxpos</i> ( <i>obj</i> ) Liefert die aktuelle horizontale Position des Label <b>obj</b> zurück.
<b>j_getypos</b>	<i>function</i> <i>j_getypos</i> ( <i>obj</i> ) Liefert die aktuelle vertikale Position des Label <b>obj</b> zurück.
<b>j_hide</b>	<i>procedure</i> <i>j_hide</i> ( <i>obj</i> ) Zeigt das Label <b>obj</b> nicht mehr auf dem Bildschirm an.
<b>j_isparent</b>	<i>function</i> <i>j_isparent</i> ( <i>obj</i> , <i>cont</i> ) Liefert J_TRUE zurück, falls das <b>cont</b> die umgebende Komponente von <b>obj</b> ist, sonst J_FALSE
<b>j_isvisible</b>	<i>function</i> <i>j_isvisible</i> ( <i>obj</i> ) Liefert J_TRUE zurück, falls <b>obj</b> sichtbar ist, sonst J_FALSE
<b>j_keylistener</b>	<i>function</i> <i>j_keylistener</i> ( <i>obj</i> ) Bindet an das Label <b>obj</b> einen neuen Keylistener, und liefert dessen Eventnummer zurück.
<b>j_mouselistener</b>	<i>function</i> <i>j_mouselistener</i> ( <i>obj</i> , <i>kind</i> ) Bindet an das Label <b>obj</b> einen neuen Mouselistener, und liefert dessen Eventnummer zurück. Der Parameter <b>kind</b> bestimmt, wann ein Event ausgelöst wird.
<b>j_popupmenu</b>	<i>function</i> <i>j_popupmenu</i> ( <i>obj</i> , <i>label</i> \$ ) Erzeugt ein neues Popupmenu mit dem Inhalt <b>label</b> und liefert dessen Eventnummer zurück.
<b>j_print</b>	<i>procedure</i> <i>j_print</i> ( <i>obj</i> ) Druckt den Inhalt des Label .
<b>j_release</b>	<i>procedure</i> <i>j_release</i> ( <i>obj</i> ) Gibt das Label <b>obj</b> aus dem umgebenden Container Element frei.
<b>j_setborderpos</b>	<i>procedure</i> <i>j_setborderpos</i> ( <i>obj</i> , <i>pos</i> ) Positioniert das Label <b>obj</b> an die angegebene Position <b>pos</b> . Im umgebenden Container muß einen BorderLayout Manager eingestellt sein.
<b>j_setcolorbg</b>	<i>procedure</i> <i>j_setcolorbg</i> ( <i>obj</i> , <i>r</i> , <i>g</i> , <i>b</i> ) Setzt die Hintergrundfarbe auf den übergebenen RGB Wert
<b>j_setcolor</b>	<i>procedure</i> <i>j_setcolor</i> ( <i>obj</i> , <i>r</i> , <i>g</i> , <i>b</i> ) Setzt die Vordergrundfarbe auf den übergebenen RGB Wert
<b>j_setcursor</b>	<i>function</i> <i>j_setcursor</i> ( <i>obj</i> , <i>cursor</i> ) Setzt den Cursor des Label <b>obj</b> auf <b>cursor</b> .

<b>j_setfocus</b>	<i>function j_setfocus ( obj )</i> Weist den Input Focus dem Label <b>obj</b> zu.
<b>j_setfontname</b>	<i>procedure j_setfontname ( obj , name )</i> Setzt den Fonttyp des Label <b>obj</b> auf <b>name</b> .
<b>j_setfont</b>	<i>procedure j_setfont ( obj , name , style , size )</i> Setzt den Font des Label <b>obj</b> entsprechend der Parameterliste.
<b>j_setfontsize</b>	<i>procedure j_setfontsize ( obj , size )</i> Setzt den Fontgröße des Label <b>obj</b> auf <b>size</b> .
<b>j_setfontstyle</b>	<i>procedure j_setfontstyle ( obj , style )</i> Setzt den Fontstyle des Label <b>obj</b> auf <b>style</b> .
<b>j_setnamedcolorbg</b>	<i>procedure j_setnamedcolorbg ( obj , color )</i> Setzt die Hintergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnamedcolor</b>	<i>procedure j_setnamedcolor ( obj , color )</i> Setzt die Vordergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setpos</b>	<i>procedure j_setpos ( obj , xpos , ypos )</i> Verschiebt das Label <b>obj</b> zur angegebenen Koordinate ( <b>xpos,ypos</b> ) an.
<b>j_setsize</b>	<i>procedure j_setsize ( obj , width , height )</i> Setzt Breite und Höhe des Label <b>obj</b> auf <b>width</b> and <b>height</b> .
<b>j_settext</b>	<i>procedure j_settext ( obj , str\$ )</i> Ändert den Textinhalt des Label <b>obj</b> auf <b>str</b>
<b>j_show</b>	<i>procedure j_show ( obj )</i> Zeigt das Label <b>obj</b> auf dem Bildschirm an.

Led
-----

- j\_led**                    *function j\_led ( obj , style , color )*  
Erzeugt eine LED der Farbe **color** und der Form **style**
- j\_add**                    *procedure j\_add ( obj , cont )*  
Fügt die LED **obj** in den Container **cont** ein.
- j\_componentlistener** *function j\_componentlistener ( obj , kind )*  
Bindet an die LED **obj** einen neuen Componentlistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.
- j\_disable**                *procedure j\_disable ( obj )*  
disabled die LED **obj**, sodaß diese auf keine Benutzereingaben mehr reagiert.
- j\_dispose**                *procedure j\_dispose ( obj )*  
die LED **obj** wird mit allen Recourcen gelöscht.
- j\_enable**                 *procedure j\_enable ( obj )*  
enabled die LED **obj**.
- j\_focuslistener**        *function j\_focuslistener ( obj )*  
Bindet an die LED **obj** einen neuen Focuslistener, und liefert dessen Eventnummer zurück.
- j\_getfontascent**        *function j\_getfontascent ( obj )*  
Liefert die Oberlänge (Ascent) des aktuellen Fonts der LED **obj** in Bildschirm-pixeln zurück.
- j\_getfontheight**        *function j\_getfontheight ( obj )*  
Liefert die totale Höhe des aktuellen Fonts der LED **obj** in Bildschirmpixeln zurück.
- j\_getheight**            *function j\_getheight ( obj )*  
Liefert die Höhe der LED **obj** zurück.
- j\_getparentid**         *function j\_getparentid ( obj )*  
Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird **-1** zurückgegeben.
- j\_getparent**            *function j\_getparent ( obj )*  
Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird **-1** zurückgegeben.
- j\_getstate**             *function j\_getstate ( obj )*  
Liefert **J\_TRUE** , falls die LED selektiert ist, sonst **J\_FALSE** .
- j\_getstringwidth**      *function j\_getstringwidth ( obj , str\$ )*

	Liefert die Länge des <b>str</b> des aktuellen Fonts der LED <b>obj</b> in Bildschirmpixeln zurück.
<b>j_getwidth</b>	<i>function j_getwidth ( obj )</i> Liefert die Breite der LED <b>obj</b> zurück.
<b>j_getxpos</b>	<i>function j_getxpos ( obj )</i> Liefert die aktuelle horizontale Position der LED <b>obj</b> zurück.
<b>j_getypos</b>	<i>function j_getypos ( obj )</i> Liefert die aktuelle vertikale Position der LED <b>obj</b> zurück.
<b>j_hide</b>	<i>procedure j_hide ( obj )</i> Zeigt die LED <b>obj</b> nicht mehr auf dem Bildschirm an.
<b>j_isparent</b>	<i>function j_isparent ( obj , cont )</i> Liefert J_TRUE zurück, falls das <b>cont</b> die umgebende Komponente von <b>obj</b> ist, sonst J_FALSE
<b>j_isvisible</b>	<i>function j_isvisible ( obj )</i> Liefert J_TRUE zurück, falls <b>obj</b> sichtbar ist, sonst J_FALSE
<b>j_keylistener</b>	<i>function j_keylistener ( obj )</i> Bindet an die LED <b>obj</b> einen neuen Keylistener, und liefert dessen Eventnummer zurück.
<b>j_mouselistener</b>	<i>function j_mouselistener ( obj , kind )</i> Bindet an die LED <b>obj</b> einen neuen Mouselistener, und liefert dessen Eventnummer zurück. Der Parameter <b>kind</b> bestimmt, wann ein Event ausgelöst wird.
<b>j_popupmenu</b>	<i>function j_popupmenu ( obj , label\$ )</i> Erzeugt ein neues Popupmenu mit dem Inhalt <b>label</b> und liefert dessen Eventnummer zurück.
<b>j_print</b>	<i>procedure j_print ( obj )</i> Druckt den Inhalt der LED .
<b>j_release</b>	<i>procedure j_release ( obj )</i> Gibt die LED <b>obj</b> aus dem umgebenden Container Element frei.
<b>j_setborderpos</b>	<i>procedure j_setborderpos ( obj , pos )</i> Positioniert die LED <b>obj</b> an die angegebene Position <b>pos</b> . Im umgebenden Container muß einen BorderLayout Manager eingestellt sein.
<b>j_setcolorbg</b>	<i>procedure j_setcolorbg ( obj , r , g , b )</i> Setzt die Hintergrundfarbe auf den übergebenen RGB Wert
<b>j_setcolor</b>	<i>procedure j_setcolor ( obj , r , g , b )</i> Setzt die Vordergrundfarbe auf den übergebenen RGB Wert
<b>j_setcursor</b>	<i>function j_setcursor ( obj , cursor )</i> Setzt den Cursor der LED <b>obj</b> auf <b>cursor</b> .



<b>j_setfocus</b>	<i>function j_setfocus ( obj )</i> Weist den Input Focus der LED <b>obj</b> zu.
<b>j_setfontname</b>	<i>procedure j_setfontname ( obj , name )</i> Setzt den Fonttyp der LED <b>obj</b> auf <b>name</b> .
<b>j_setfont</b>	<i>procedure j_setfont ( obj , name , style , size )</i> Setzt den Font der LED <b>obj</b> entsprechend der Parameterliste.
<b>j_setfontsize</b>	<i>procedure j_setfontsize ( obj , size )</i> Setzt den Fontgröße der LED <b>obj</b> auf <b>size</b> .
<b>j_setfontstyle</b>	<i>procedure j_setfontstyle ( obj , style )</i> Setzt den Fontstyle der LED <b>obj</b> auf <b>style</b> .
<b>j_setnamedcolorbg</b>	<i>procedure j_setnamedcolorbg ( obj , color )</i> Setzt die Hintergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnamedcolor</b>	<i>procedure j_setnamedcolor ( obj , color )</i> Setzt die Vordergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setpos</b>	<i>procedure j_setpos ( obj , xpos , ypos )</i> Verschiebt die LED <b>obj</b> zur angegebenen Koordinate ( <b>xpos,ypos</b> ) an.
<b>j_setsize</b>	<i>procedure j_setsize ( obj , width , height )</i> Setzt Breite und Höhe der LED <b>obj</b> auf <b>width</b> and <b>height</b> .
<b>j_setstate</b>	<i>procedure j_setstate ( obj , bool )</i> Ist <b>bool</b> J_TRUE so wird die LED selectiert.
<b>j_show</b>	<i>procedure j_show ( obj )</i> Zeigt die LED <b>obj</b> auf dem Bildschirm an.

List
------

- j\_list**                    *function j\_list ( obj , rows )*  
Erzeugt eine neue List und gibt deren Eventnummer zurück.
- j\_additem**                *procedure j\_additem ( obj , str\$ )*  
Hängt ein neues Item mit dem Inhalt **str** an die Auswahlliste an.
- j\_add**                     *procedure j\_add ( obj , cont )*  
Fügt die List **obj** in den Container **cont** ein.
- j\_componentlistener** *function j\_componentlistener ( obj , kind )*  
Bindet an die List **obj** einen neuen Componentlistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.
- j\_deselect**               *function j\_deselect ( obj , item )*  
Deselectiert das Item mit dem Index **item**.
- j\_disable**                *procedure j\_disable ( obj )*  
disabled die List **obj**, sodaß diese auf keine Benutzereingaben mehr reagiert.
- j\_dispose**                *procedure j\_dispose ( obj )*  
die List **obj** wird mit allen Recourcen gelöscht.
- j\_enable**                 *procedure j\_enable ( obj )*  
enabled die List **obj**.
- j\_focuslistener**        *function j\_focuslistener ( obj )*  
Bindet an die List **obj** einen neuen Focuslistener, und liefert dessen Eventnummer zurück.
- j\_getfontascent**        *function j\_getfontascent ( obj )*  
Liefert die Oberlänge (Ascent) des aktuellen Fonts der List **obj** in Bildschirm-pixeln zurück.
- j\_getfontheight**        *function j\_getfontheight ( obj )*  
Liefert die totale Höhe des aktuellen Fonts der List **obj** in Bildschirmpixeln zurück.
- j\_getheight**             *function j\_getheight ( obj )*  
Liefert die Höhe der List **obj** zurück.
- j\_getitemcount**        *function j\_getitemcount ( obj )*  
Liefert die Anzahl der Items der List **obj** zurück.
- j\_getitem**                *function\$ j\_getitem ( obj , item )*  
liefert den Inhalt des Items mit dem Index **item** zurück.

<b>j_getparentid</b>	<i>function j_getparentid ( obj )</i> Liefert die Eventnummer des umgebenen Containers von <b>obj</b> zurück. Ist <b>obj</b> ein Frame wird <b>-1</b> zurückgegeben.
<b>j_getparent</b>	<i>function j_getparent ( obj )</i> Liefert die Eventnummer des umgebenen Containers von <b>obj</b> zurück. Ist <b>obj</b> ein Frame wird <b>-1</b> zurückgegeben.
<b>j_getselect</b>	<i>function j_getselect ( obj )</i> Liefert den Index des selectierten Items der List .
<b>j_getstringwidth</b>	<i>function j_getstringwidth ( obj , str\$ )</i> Liefert die Länge des <b>str</b> des aktuellen Fonts der List <b>obj</b> in Bildschirmpixeln zurück.
<b>j_getwidth</b>	<i>function j_getwidth ( obj )</i> Liefert die Breite der List <b>obj</b> zurück.
<b>j_getxpos</b>	<i>function j_getxpos ( obj )</i> Liefert die aktuelle horizontale Position der List <b>obj</b> zurück.
<b>j_getypos</b>	<i>function j_getypos ( obj )</i> Liefert die aktuelle vertikale Position der List <b>obj</b> zurück.
<b>j_hide</b>	<i>procedure j_hide ( obj )</i> Zeigt die List <b>obj</b> nicht mehr auf dem Bildschirm an.
<b>j_insert</b>	<i>function j_insert ( obj , pos , label\$ )</i> fügt ein neues Item in an der Position <b>pos</b> ein. Das Item trägt den Inhalt <b>label</b> .
<b>j_isparent</b>	<i>function j_isparent ( obj , cont )</i> Liefert <b>J_TRUE</b> zurück, falls das <b>cont</b> die umgebende Komponente von <b>obj</b> ist, sonst <b>J_FALSE</b>
<b>j_iselect</b>	<i>function j_iselect ( obj , item )</i> Liefert <b>J_TRUE</b> zurück, falls das Item mit dem Index <b>item</b> selectiert ist, sonst <b>J_FALSE</b>
<b>j_isvisible</b>	<i>function j_isvisible ( obj )</i> Liefert <b>J_TRUE</b> zurück, falls <b>obj</b> sichtbar ist, sonst <b>J_FALSE</b>
<b>j_keylistener</b>	<i>function j_keylistener ( obj )</i> Bindet an die List <b>obj</b> einen neuen Keylistener, und liefert dessen Eventnummer zurück.
<b>j_mouselistener</b>	<i>function j_mouselistener ( obj , kind )</i> Bindet an die List <b>obj</b> einen neuen Mouselistener, und liefert dessen Eventnummer zurück. Der Parameter <b>kind</b> bestimmt, wann ein Event ausgelöst wird.
<b>j_multiplemode</b>	<i>function j_multiplemode ( obj , bool )</i> die List erlaubt mehrfache Selectierung, falls <b>bool</b> <b>J_TRUE</b> ist.
<b>j_popupmenu</b>	<i>function j_popupmenu ( obj , label\$ )</i>

	Erzeugt ein neues Popumenu mit dem Inhalt <b>label</b> und liefert dessen Eventnummer zurück.
<b>j_print</b>	<i>procedure j_print ( obj )</i> Druckt den Inhalt der List .
<b>j_release</b>	<i>procedure j_release ( obj )</i> Gibt die List <b>obj</b> aus dem umgebenden Container Element frei.
<b>j_removeall</b>	<i>function j_removeall ( obj )</i> entfernt alle Items aus der List .
<b>j_removeitem</b>	<i>function j_removeitem ( obj , item\$ )</i> entfernt den ersten Eintrag <b>item</b> aus der List .
<b>j_remove</b>	<i>function j_remove ( obj , item )</i> entfernt das Item mit dem Index <b>item</b> aus der List .
<b>j_select</b>	<i>function j_select ( obj , item )</i> selectiert das Item mit dem Index <b>item</b> .
<b>j_setborderpos</b>	<i>procedure j_setborderpos ( obj , pos )</i> Positioniert die List <b>obj</b> an die angegebene Position <b>pos</b> . Im umgebenden Container muß einen BorderLayout Manager eingestellt sein.
<b>j_setcolorbg</b>	<i>procedure j_setcolorbg ( obj , r , g , b )</i> Setzt die Hintergrundfarbe auf den übergebenen RGB Wert
<b>j_setcolor</b>	<i>procedure j_setcolor ( obj , r , g , b )</i> Setzt die Vordergrundfarbe auf den übergebenen RGB Wert
<b>j_setcursor</b>	<i>function j_setcursor ( obj , cursor )</i> Setzt den Cursor der List <b>obj</b> auf <b>cursor</b> .
<b>j_setfocus</b>	<i>function j_setfocus ( obj )</i> Weist den Input Focus der List <b>obj</b> zu.
<b>j_setfontname</b>	<i>procedure j_setfontname ( obj , name )</i> Setzt den Fonttyp der List <b>obj</b> auf <b>name</b> .
<b>j_setfont</b>	<i>procedure j_setfont ( obj , name , style , size )</i> Setzt den Font der List <b>obj</b> entsprechend der Parameterliste.
<b>j_setfontsize</b>	<i>procedure j_setfontsize ( obj , size )</i> Setzt den Fontgröße der List <b>obj</b> auf <b>size</b> .
<b>j_setfontstyle</b>	<i>procedure j_setfontstyle ( obj , style )</i> Setzt den Fontstyle der List <b>obj</b> auf <b>style</b> .
<b>j_setnamedcolorbg</b>	<i>procedure j_setnamedcolorbg ( obj , color )</i> Setzt die Hintergrundfarbe auf eine vordefinierte Farbe <b>color</b> .

- j\_setnamedcolor**     *procedure j\_setnamedcolor ( obj , color )*  
Setzt die Vordergrundfarbe auf eine vordefinierte Farbe **color**.
- j\_setpos**             *procedure j\_setpos ( obj , xpos , ypos )*  
Verschiebt die List **obj** zur angegebenen Koordinate (**xpos,ypos**) an.
- j\_setsize**            *procedure j\_setsize ( obj , width , height )*  
Setzt Breite und Höhe der List **obj** auf **width** and **height**.
- j\_show**                *procedure j\_show ( obj )*  
Zeigt die List **obj** auf dem Bildschirm an.

Menu
------

<b>j_menu</b>	<p><i>function j_menu ( obj , str\$ )</i>          Erzeugt ein Menueintrag mit der Beschriftung <b>label</b> und liefert dessen Eventnummer zurück.</p>
<b>j_checkmenuitem</b>	<p><i>function j_checkmenuitem ( obj , label\$ )</i>          Erzeugt ein Checkmenuitem mit der Beschriftung <b>label</b> und liefert dessen Eventnummer zurück.</p>
<b>j_disable</b>	<p><i>procedure j_disable ( obj )</i>          disabled das Menu <b>obj</b>, sodaß diese auf keine Benutzereingaben mehr reagiert.</p>
<b>j_dispose</b>	<p><i>procedure j_dispose ( obj )</i>          das Menu <b>obj</b> wird mit allen Recourcen gelöscht.</p>
<b>j_enable</b>	<p><i>procedure j_enable ( obj )</i>          enabled das Menu <b>obj</b>.</p>
<b>j_getlength</b>	<p><i>function j_getlength ( obj )</i>          Liefert die Länge des Textinhaltes das Menu zurück.</p>
<b>j_gettext</b>	<p><i>function\$ j_gettext ( obj )</i>          Liefert den Textinhalt des Menu .</p>
<b>j_helpmenu</b>	<p><i>function j_helpmenu ( obj , label\$ )</i>          Erzeugt ein Hilfe-Menueintrag mit der Beschriftung <b>label</b> und liefert dessen Eventnummer zurück.</p>
<b>j_menuitem</b>	<p><i>function j_menuitem ( obj , label\$ )</i>          Erzeugt ein MenuItem mit der Beschriftung <b>label</b> und liefert dessen Eventnummer zurück.</p>
<b>j_menu</b>	<p><i>function j_menu ( obj , str\$ )</i>          Erzeugt ein Menueintrag mit der Beschriftung <b>label</b> und liefert dessen Eventnummer zurück.</p>
<b>j_seperator</b>	<p><i>procedure j_seperator ( obj )</i>          Erzeugt eine Trennlinie in das Menu .</p>
<b>j_setfontname</b>	<p><i>procedure j_setfontname ( obj , name )</i>          Setzt den Fonttyp des Menu <b>obj</b> auf <b>name</b>.</p>
<b>j_setfont</b>	<p><i>procedure j_setfont ( obj , name , style , size )</i>          Setzt den Font des Menu <b>obj</b> entsprechend der Parameterliste.</p>
<b>j_setfontsize</b>	<p><i>procedure j_setfontsize ( obj , size )</i>          Setzt den Fontgröße des Menu <b>obj</b> auf <b>size</b>.</p>

**j\_setfontstyle**      *procedure j\_setfontstyle ( obj , style )*  
Setzt den Fontstyle des Menu **obj** auf **style**.

**j\_setshortcut**      *procedure j\_setshortcut ( obj , chr )*  
Bestimmt das Zeichen **chr** als Shortcut für das Menu .

**j\_settext**            *procedure j\_settext ( obj , str\$ )*  
Ändert den Textinhalt des Menu **obj** auf **str**

Menuitem
----------

<b>j_menuitem</b>	<p><i>function j_menuitem ( obj , label\$ )</i>          Erzeugt ein Menuitem mit der Beschriftung <b>label</b> und liefert dessen Eventnummer zurück.</p>
<b>j_disable</b>	<p><i>procedure j_disable ( obj )</i>          disabled das Menuitem <b>obj</b>, sodaß diese auf keine Benutzereingaben mehr reagiert.</p>
<b>j_dispose</b>	<p><i>procedure j_dispose ( obj )</i>          das Menuitem <b>obj</b> wird mit allen Recourcen gelöscht.</p>
<b>j_enable</b>	<p><i>procedure j_enable ( obj )</i>          enabled das Menuitem <b>obj</b>.</p>
<b>j_getlength</b>	<p><i>function j_getlength ( obj )</i>          Liefert die Länge des Textinhaltes das Menuitem zurück.</p>
<b>j_gettext</b>	<p><i>function\$ j_gettext ( obj )</i>          Liefert den Textinhalt des Menuitem .</p>
<b>j_setfontname</b>	<p><i>procedure j_setfontname ( obj , name )</i>          Setzt den Fonttyp des Menuitem <b>obj</b> auf <b>name</b>.</p>
<b>j_setfont</b>	<p><i>procedure j_setfont ( obj , name , style , size )</i>          Setzt den Font des Menuitem <b>obj</b> entsprechend der Parameterliste.</p>
<b>j_setfontsize</b>	<p><i>procedure j_setfontsize ( obj , size )</i>          Setzt den Fontgröße des Menuitem <b>obj</b> auf <b>size</b>.</p>
<b>j_setfontstyle</b>	<p><i>procedure j_setfontstyle ( obj , style )</i>          Setzt den Fontstyle des Menuitem <b>obj</b> auf <b>style</b>.</p>
<b>j_setshortcut</b>	<p><i>procedure j_setshortcut ( obj , chr )</i>          Bestimmt das Zeichen <b>chr</b> als Shortcut für das Menuitem .</p>
<b>j_settext</b>	<p><i>procedure j_settext ( obj , str\$ )</i>          Ändert den Textinhalt des Menuitem <b>obj</b> auf <b>str</b></p>



Meter
-------

- j\_meter**            *function j\_meter ( obj , title\$ )*  
Erzeugt ein analoges Anzeigeinstrument mit der Beschriftung **title**.
- j\_add**                *procedure j\_add ( obj , cont )*  
Fügt das Anzeigeinstrument **obj** in den Container **cont** ein.
- j\_componentlistener** *function j\_componentlistener ( obj , kind )*  
Bindet an das Anzeigeinstrument **obj** einen neuen Componentlistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.
- j\_disable**            *procedure j\_disable ( obj )*  
disabled das Anzeigeinstrument **obj**, sodaß diese auf keine Benutzereingaben mehr reagiert.
- j\_dispose**            *procedure j\_dispose ( obj )*  
das Anzeigeinstrument **obj** wird mit allen Recourcen gelöscht.
- j\_enable**             *procedure j\_enable ( obj )*  
enabled das Anzeigeinstrument **obj**.
- j\_focuslistener**    *function j\_focuslistener ( obj )*  
Bindet an das Anzeigeinstrument **obj** einen neuen Focuslistener, und liefert dessen Eventnummer zurück.
- j\_getdanger**         *procedure j\_getdanger ( obj )*  
Liefert den Warnlevel des Anzeigeinstrument **obj** zurück.
- j\_getfontascent**    *function j\_getfontascent ( obj )*  
Liefert die Oberlänge (Ascent) des aktuellen Fonts des Anzeigeinstrument **obj** in Bildschirmpixeln zurück.
- j\_getfontheight**    *function j\_getfontheight ( obj )*  
Liefert die totale Höhe des aktuellen Fonts des Anzeigeinstrument **obj** in Bildschirmpixeln zurück.
- j\_getheight**         *function j\_getheight ( obj )*  
Liefert die Höhe des Anzeigeinstrument **obj** zurück.
- j\_getparentid**       *function j\_getparentid ( obj )*  
Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird  $-1$  zurückgegeben.
- j\_getparent**         *function j\_getparent ( obj )*  
Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird  $-1$  zurückgegeben.

<b>j_getstringwidth</b>	<i>function j_getstringwidth ( obj , str\$ )</i> Liefert die Länge des <b>str</b> des aktuellen Fonts des Anzeigeelement <b>obj</b> in Bildschirmpixeln zurück.
<b>j_getwidth</b>	<i>function j_getwidth ( obj )</i> Liefert die Breite des Anzeigeelement <b>obj</b> zurück.
<b>j_getxpos</b>	<i>function j_getxpos ( obj )</i> Liefert die aktuelle horizontale Position des Anzeigeelement <b>obj</b> zurück.
<b>j_getypos</b>	<i>function j_getypos ( obj )</i> Liefert die aktuelle vertikale Position des Anzeigeelement <b>obj</b> zurück.
<b>j_hide</b>	<i>procedure j_hide ( obj )</i> Zeigt das Anzeigeelement <b>obj</b> nicht mehr auf dem Bildschirm an.
<b>j_isparent</b>	<i>function j_isparent ( obj , cont )</i> Liefert J_TRUE zurück, falls das <b>cont</b> die umgebende Komponente von <b>obj</b> ist, sonst J_FALSE
<b>j_isvisible</b>	<i>function j_isvisible ( obj )</i> Liefert J_TRUE zurück, falls <b>obj</b> sichtbar ist, sonst J_FALSE
<b>j_keylistener</b>	<i>function j_keylistener ( obj )</i> Bindet an das Anzeigeelement <b>obj</b> einen neuen Keylistener, und liefert dessen Eventnummer zurück.
<b>j_mouselistener</b>	<i>function j_mouselistener ( obj , kind )</i> Bindet an das Anzeigeelement <b>obj</b> einen neuen Mouselistener, und liefert dessen Eventnummer zurück. Der Parameter <b>kind</b> bestimmt, wann ein Event ausgelöst wird.
<b>j_popupmenu</b>	<i>function j_popupmenu ( obj , label\$ )</i> Erzeugt ein neues Popupmenu mit dem Inhalt <b>label</b> und liefert dessen Eventnummer zurück.
<b>j_print</b>	<i>procedure j_print ( obj )</i> Druckt den Inhalt des Anzeigeelement .
<b>j_release</b>	<i>procedure j_release ( obj )</i> Gibt das Anzeigeelement <b>obj</b> aus dem umgebenden Container Element frei.
<b>j_setborderpos</b>	<i>procedure j_setborderpos ( obj , pos )</i> Positioniert das Anzeigeelement <b>obj</b> an die angegebene Position <b>pos</b> . Im umgebenden Container muß ein BorderLayout Manager eingestellt sein.
<b>j_setcolorbg</b>	<i>procedure j_setcolorbg ( obj , r , g , b )</i> Setzt die Hintergrundfarbe auf den übergebenen RGB Wert
<b>j_setcolor</b>	<i>procedure j_setcolor ( obj , r , g , b )</i> Setzt die Vordergrundfarbe auf den übergebenen RGB Wert

<b>j_setcursor</b>	<i>function j_setcursor ( obj , cursor )</i> Setzt den Cursor des Anzeigeinstrument <b>obj</b> auf <b>cursor</b> .
<b>j_setdanger</b>	<i>procedure j_setdanger ( obj , val )</i> Stellt den Warnlevel dem Anzeigeinstrument <b>obj</b> auf <b>val</b> .
<b>j_setfocus</b>	<i>function j_setfocus ( obj )</i> Weist den Input Focus dem Anzeigeinstrument <b>obj</b> zu.
<b>j_setfontname</b>	<i>procedure j_setfontname ( obj , name )</i> Setzt den Fonttyp des Anzeigeinstrument <b>obj</b> auf <b>name</b> .
<b>j_setfont</b>	<i>procedure j_setfont ( obj , name , style , size )</i> Setzt den Font des Anzeigeinstrument <b>obj</b> entsprechend der Parameterliste.
<b>j_setfontsize</b>	<i>procedure j_setfontsize ( obj , size )</i> Setzt den Fontgröße des Anzeigeinstrument <b>obj</b> auf <b>size</b> .
<b>j_setfontstyle</b>	<i>procedure j_setfontstyle ( obj , style )</i> Setzt den Fontstyle des Anzeigeinstrument <b>obj</b> auf <b>style</b> .
<b>j_setmax</b>	<i>function j_setmax ( obj , val )</i> Stellt die obere Grenze dem Anzeigeinstrument auf den Wert von <b>val</b> ein.
<b>j_setmin</b>	<i>function j_setmin ( obj , val )</i> Stellt die untere Grenze dem Anzeigeinstrument auf den Wert von <b>val</b> ein.
<b>j_setnamedcolorbg</b>	<i>procedure j_setnamedcolorbg ( obj , color )</i> Setzt die Hintergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnamedcolor</b>	<i>procedure j_setnamedcolor ( obj , color )</i> Setzt die Vordergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setpos</b>	<i>procedure j_setpos ( obj , xpos , ypos )</i> Verschiebt das Anzeigeinstrument <b>obj</b> zur angegebenen Koordinate ( <b>xpos,ypos</b> ) an.
<b>j_setsize</b>	<i>procedure j_setsize ( obj , width , height )</i> Setzt Breite und Höhe des Anzeigeinstrument <b>obj</b> auf <b>width</b> and <b>height</b> .
<b>j_setvalue</b>	<i>procedure j_setvalue ( obj , val )</i> Stellt den aktuellen Wert dem Anzeigeinstrument <b>obj</b> auf <b>val</b> .
<b>j_show</b>	<i>procedure j_show ( obj )</i> Zeigt das Anzeigeinstrument <b>obj</b> auf dem Bildschirm an.

Mouselistener
---------------

- j\_mouselistener**     *function j\_mouselistener ( obj , kind )*  
Bindet an den Mouselistener **obj** einen neuen Mouselistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.
- j\_dispose**             *procedure j\_dispose ( obj )*  
der Mouselistener **obj** wird mit allen Recourcen gelöscht.
- j\_getmousebutton**     *function j\_getmousebutton ( mouselistener )*  
Liefert die letzte benutzte Mousetaste zurück.
- j\_getmousex**           *function j\_getmousex ( mouselistener )*  
Liefert die aktuelle horizontale Position der Mouse zurück.
- j\_getmousey**           *function j\_getmousey ( mouselistener )*  
Liefert die aktuelle vertikale Position der Mouse zurück.

Panel
-------

- j\_panel**                    *function j\_panel ( obj )*  
Erzeugt ein neues panel und liefert dessen Eventnummer zurück.
- j\_add**                        *procedure j\_add ( obj , cont )*  
Fügt das Panel **obj** in den Container **cont** ein.
- j\_borderpanel**            *function j\_borderpanel ( obj , type )*  
Erzeugt ein neues Borderpanel und liefert dessen Eventnummer zurück.
- j\_button**                    *function j\_button ( obj , label\$ )*  
Erzeugt einen neuen Button mit dem Textinhalt **label**
- j\_canvas**                    *function j\_canvas ( obj , width , height )*  
Erzeugt einen neuen Canvas mit der Breite **width** Pixel und einer Höhe von **height** Pixel.
- j\_checkbox**                *function j\_checkbox ( obj , label\$ )*  
Erzeugt eine Checkbox mit dem Textinhalt **label** und liefert dessen Eventnummer zurück.
- j\_choice**                    *function j\_choice ( obj )*  
Erzeugt eine neue Choice und gibt deren Eventnummer zurück.
- j\_componentlistener** *function j\_componentlistener ( obj , kind )*  
Bindet an das Panel **obj** einen neuen Componentlistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.
- j\_disable**                    *procedure j\_disable ( obj )*  
disabled das Panel **obj**, sodaß diese auf keine Benutzereingaben mehr reagiert.
- j\_dispose**                 *procedure j\_dispose ( obj )*  
das Panel **obj** wird mit allen Recourcen gelöscht.
- j\_enable**                    *procedure j\_enable ( obj )*  
enabled das Panel **obj**.
- j\_focuslistener**           *function j\_focuslistener ( obj )*  
Bindet an das Panel **obj** einen neuen Focuslistener, und liefert dessen Eventnummer zurück.
- j\_getfontascent**           *function j\_getfontascent ( obj )*  
Liefert die Oberlänge (Ascent) des aktuellen Fonts des Panel **obj** in Bildschirm-pixeln zurück.
- j\_getfontheight**           *function j\_getfontheight ( obj )*

	Liefert die totale Höhe des aktuellen Fonts des Panel <b>obj</b> in Bildschirmpixeln zurück.
<b>j_getheight</b>	<i>function j_getheight ( obj )</i> Liefert die Höhe des Panel <b>obj</b> zurück.
<b>j_getinheight</b>	<i>function j_getinheight ( cont )</i> Liefert die Höhe des Panel <b>obj</b> des inneren Bereichs zurück.
<b>j_getinsets</b>	<i>function j_getinsets ( obj , side )</i> Liefert die angeforderte Breite des inneren Randes dem Panel zurück.
<b>j_getinwidth</b>	<i>function j_getinwidth ( cont )</i> Liefert die Breite des Panel <b>obj</b> des inneren Bereichs zurück.
<b>j_getlayoutid</b>	<i>function j_getlayoutid ( obj )</i> Liefert die Eventnummer des Layoutmanagers des Containers <b>obj</b> zurück.
<b>j_getparentid</b>	<i>function j_getparentid ( obj )</i> Liefert die Eventnummer des umgebenen Containers von <b>obj</b> zurück. Ist <b>obj</b> ein Frame wird $-1$ zurückgegeben.
<b>j_getparent</b>	<i>function j_getparent ( obj )</i> Liefert die Eventnummer des umgebenen Containers von <b>obj</b> zurück. Ist <b>obj</b> ein Frame wird $-1$ zurückgegeben.
<b>j_getstringwidth</b>	<i>function j_getstringwidth ( obj , str\$ )</i> Liefert die Länge des <b>str</b> des aktuellen Fonts des Panel <b>obj</b> in Bildschirmpixeln zurück.
<b>j_getwidth</b>	<i>function j_getwidth ( obj )</i> Liefert die Breite des Panel <b>obj</b> zurück.
<b>j_getxpos</b>	<i>function j_getxpos ( obj )</i> Liefert die aktuelle horizontale Position des Panel <b>obj</b> zurück.
<b>j_getypos</b>	<i>function j_getypos ( obj )</i> Liefert die aktuelle vertikale Position des Panel <b>obj</b> zurück.
<b>j_graphicbutton</b>	<i>function j_graphicbutton ( obj , filename\$ )</i> Erzeugt einen neuen Graphicbutton mit dem Bildinhalt des Files, das durch <b>filename</b> spezifiziert wird.
<b>j_graphiclabel</b>	<i>function j_graphiclabel ( obj , str\$ )</i> Erzeugt ein neues Graphiclabel mit dem Bildinhalt des Files, das durch <b>filename</b> spezifiziert wird.
<b>j_hide</b>	<i>procedure j_hide ( obj )</i> Zeigt das Panel <b>obj</b> nicht mehr auf dem Bildschirm an.
<b>j_hscrollbar</b>	<i>function j_hscrollbar ( obj )</i> Erzeugt eine horizontale Scrollbar und gibt dessen Eventnummer zurück.

<b>j_isparent</b>	<i>function j_isparent ( obj , cont )</i> Liefert J_TRUE zurück, falls das <b>cont</b> die umgebende Komponente von <b>obj</b> ist, sonst J_FALSE
<b>j_isvisible</b>	<i>function j_isvisible ( obj )</i> Liefert J_TRUE zurück, falls <b>obj</b> sichtbar ist, sonst J_FALSE
<b>j_keylistener</b>	<i>function j_keylistener ( obj )</i> Bindet an das Panel <b>obj</b> einen neuen Keylistener, und liefert dessen Eventnummer zurück.
<b>j_label</b>	<i>function j_label ( obj , label\$ )</i> Erzeugt ein neues Label Element mit dem Textinhalt <b>label</b>
<b>j_led</b>	<i>function j_led ( obj , style , color )</i> Erzeugt eine LED der Farbe <b>color</b> und der Form <b>style</b>
<b>j_line</b>	<i>function j_line ( obj , orient , style , length )</i> Erzeugt ein neues Linienelement und liefert dessen Eventnummer zurück
<b>j_list</b>	<i>function j_list ( obj , rows )</i> Erzeugt eine neue List und gibt deren Eventnummer zurück.
<b>j_meter</b>	<i>function j_meter ( obj , title\$ )</i> Erzeugt ein analoges Anzeigeelement mit der Beschriftung <b>title</b> .
<b>j_mouselistener</b>	<i>function j_mouselistener ( obj , kind )</i> Bindet an das Panel <b>obj</b> einen neuen Mouselistener, und liefert dessen Eventnummer zurück. Der Parameter <b>kind</b> bestimmt, wann ein Event ausgelöst wird.
<b>j_pack</b>	<i>procedure j_pack ( obj )</i> Veranlaßt den Layoutmanager von <b>obj</b> die minimale Größe zu berechnen.
<b>j_panel</b>	<i>function j_panel ( obj )</i> Erzeugt ein neues panel und liefert dessen Eventnummer zurück.
<b>j_popupmenu</b>	<i>function j_popupmenu ( obj , label\$ )</i> Erzeugt ein neues Popupmenu mit dem Inhalt <b>label</b> und liefert dessen Eventnummer zurück.
<b>j_print</b>	<i>procedure j_print ( obj )</i> Druckt den Inhalt des Panel .
<b>j_progressbar</b>	<i>function j_progressbar ( obj , orient )</i> Erzeugt eine Progress Bar mit der Ausrichtung <b>orient</b> .
<b>j_radiogroup</b>	<i>function j_radiogroup ( obj )</i> Erzeugt eine neue Radiogroup und liefert dessen Eventnummer zurück.
<b>j_releaseall</b>	<i>procedure j_releaseall ( obj )</i> Gibt alle Elemente aus dem Panel <b>obj</b> frei.

<b>j_release</b>	<i>procedure j_release ( obj )</i> Gibt das Panel <b>obj</b> aus dem umgebenden Container Element frei.
<b>j_scrollpane</b>	<i>function j_scrollpane ( obj )</i> Erzeugt eine neue Scrollpane und gibt dessen Eventnummer zurück.
<b>j_setalign</b>	<i>procedure j_setalign ( obj , align )</i> Setzt die Ausrichtung innerhalb des Panel <b>obj</b> auf <b>align</b> . Ein Flowlayout Manager wird vorausgesetzt.
<b>j_setborderlayout</b>	<i>procedure j_setborderlayout ( obj )</i> Integriert einen BorderLayout Manager in das Panel <b>obj</b> .
<b>j_setborderpos</b>	<i>procedure j_setborderpos ( obj , pos )</i> Positioniert das Panel <b>obj</b> an die angegebene Position <b>pos</b> . Im umgebenden Container muß einen BorderLayout Manager eingestellt sein.
<b>j_setcolorbg</b>	<i>procedure j_setcolorbg ( obj , r , g , b )</i> Setzt die Hintergrundfarbe auf den übergebenen RGB Wert
<b>j_setcolor</b>	<i>procedure j_setcolor ( obj , r , g , b )</i> Setzt die Vordergrundfarbe auf den übergebenen RGB Wert
<b>j_setcursor</b>	<i>function j_setcursor ( obj , cursor )</i> Setzt den Cursor des Panel <b>obj</b> auf <b>cursor</b> .
<b>j_setfixlayout</b>	<i>procedure j_setfixlayout ( obj )</i> Integriert einen Fixlayout Manager in das Panel <b>obj</b> (voreingestellte Layoutmanager).
<b>j_setflowfill</b>	<i>procedure j_setflowfill ( obj , bool )</i> Setzt die Höhe (Breite) aller enthaltenen Komponenten auf die Höhe (Breite) von dem Panel <b>obj</b> . Ein Flowlayout Manager wird vorausgesetzt.
<b>j_setflowlayout</b>	<i>procedure j_setflowlayout ( obj , align )</i> Integriert einen Flowlayout Manager in das Panel <b>obj</b> mit der Ausrichtung <b>align</b> .
<b>j_setfocus</b>	<i>function j_setfocus ( obj )</i> Weist den Input Focus dem Panel <b>obj</b> zu.
<b>j_setfontname</b>	<i>procedure j_setfontname ( obj , name )</i> Setzt den Fonttyp des Panel <b>obj</b> auf <b>name</b> .
<b>j_setfont</b>	<i>procedure j_setfont ( obj , name , style , size )</i> Setzt den Font des Panel <b>obj</b> entsprechend der Parameterliste.
<b>j_setfontsize</b>	<i>procedure j_setfontsize ( obj , size )</i> Setzt den Fontgröße des Panel <b>obj</b> auf <b>size</b> .
<b>j_setfontstyle</b>	<i>procedure j_setfontstyle ( obj , style )</i> Setzt den Fontstyle des Panel <b>obj</b> auf <b>style</b> .



<b>j_setgridlayout</b>	<i>procedure j_setgridlayout ( obj , row , col )</i> Integriert einen Gridlayout Manager in das Panel <b>obj</b> mit <b>row</b> Reihen und <b>col</b> Spalten.
<b>j_sethgap</b>	<i>procedure j_sethgap ( obj , hgap )</i> Setzt den horizontalen Abstand innerhalb des Panel <b>obj</b> auf <b>hgap</b> Pixel.
<b>j_setinsets</b>	<i>procedure j_setinsets ( obj , top , bottom , left , right )</i> Setzt den inneren Rahmen auf die angegebenen Größen.
<b>j_setnamedcolorbg</b>	<i>procedure j_setnamedcolorbg ( obj , color )</i> Setzt die Hintergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnamedcolor</b>	<i>procedure j_setnamedcolor ( obj , color )</i> Setzt die Vordergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnolayout</b>	<i>procedure j_setnolayout ( obj )</i> Entfernt den aktuellen Layout Manager aus dem Panel <b>obj</b> .
<b>j_setpos</b>	<i>procedure j_setpos ( obj , xpos , ypos )</i> Verschiebt das Panel <b>obj</b> zur angegebenen Koordinate ( <b>xpos,ypos</b> ) an.
<b>j_setsize</b>	<i>procedure j_setsize ( obj , width , height )</i> Setzt Breite und Höhe des Panel <b>obj</b> auf <b>width</b> and <b>height</b> .
<b>j_setvgap</b>	<i>procedure j_setvgap ( obj , vgap )</i> Setzt den vertikalen Abstand innerhalb des Panel <b>obj</b> auf <b>vgap</b> Pixel.
<b>j_sevensegment</b>	<i>function j_sevensegment ( obj , color )</i> Erzeugt eine Siebensegment Anzeige der Farbe <b>color</b> .
<b>j_show</b>	<i>procedure j_show ( obj )</i> Zeigt das Panel <b>obj</b> auf dem Bildschirm an.
<b>j_textarea</b>	<i>function j_textarea ( obj , rows , columns )</i> Erzeugt eine neue Textarea und gibt dessen Eventnummer zurück.
<b>j_textfield</b>	<i>function j_textfield ( obj , columns )</i> Erzeugt eine neues Textfield und gibt dessen Eventnummer zurück.
<b>j_vscrollbar</b>	<i>function j_vscrollbar ( obj )</i> Erzeugt eine vertikale Scrollbar und gibt dessen Eventnummer zurück.

<h2 style="margin: 0;">Popupmenu</h2>
---------------------------------------

<b>j_popupmenu</b>	<i>function j_popupmenu ( obj , label\$ )</i> Erzeugt ein neues Popupmenu mit dem Inhalt <b>label</b> und liefert dessen Eventnummer zurück.
<b>j_checkmenuitem</b>	<i>function j_checkmenuitem ( obj , label\$ )</i> Erzeugt ein Checkmenuitem mit der Beschriftung <b>label</b> und liefert dessen Eventnummer zurück.
<b>j_disable</b>	<i>procedure j_disable ( obj )</i> disabled das Popupmenu <b>obj</b> , sodaß diese auf keine Benutzereingaben mehr reagiert.
<b>j_dispose</b>	<i>procedure j_dispose ( obj )</i> das Popupmenu <b>obj</b> wird mit allen Recourcen gelöscht.
<b>j_enable</b>	<i>procedure j_enable ( obj )</i> enabled das Popupmenu <b>obj</b> .
<b>j_getlength</b>	<i>function j_getlength ( obj )</i> Liefert die Länge des Textinhaltes das Popupmenu zurück.
<b>j_gettext</b>	<i>function\$ j_gettext ( obj )</i> Liefert den Textinhalt des Popupmenu .
<b>j_menuitem</b>	<i>function j_menuitem ( obj , label\$ )</i> Erzeugt ein Menuitem mit der Beschriftung <b>label</b> und liefert dessen Eventnummer zurück.
<b>j_seperator</b>	<i>procedure j_seperator ( obj )</i> Erzeugt eine Trennlinie in das Popupmenu .
<b>j_setfontname</b>	<i>procedure j_setfontname ( obj , name )</i> Setzt den Fonttyp des Popupmenu <b>obj</b> auf <b>name</b> .
<b>j_setfont</b>	<i>procedure j_setfont ( obj , name , style , size )</i> Setzt den Font des Popupmenu <b>obj</b> entsprechend der Parameterliste.
<b>j_setfontsize</b>	<i>procedure j_setfontsize ( obj , size )</i> Setzt den Fontgröße des Popupmenu <b>obj</b> auf <b>size</b> .
<b>j_setfontstyle</b>	<i>procedure j_setfontstyle ( obj , style )</i> Setzt den Fontstyle des Popupmenu <b>obj</b> auf <b>style</b> .
<b>j_setshortcut</b>	<i>procedure j_setshortcut ( obj , chr )</i> Bestimmt das Zeichen <b>chr</b> als Shortcut für das Popupmenu .
<b>j_settext</b>	<i>procedure j_settext ( obj , str\$ )</i>

**j\_showpopup**      Ändert den Textinhalt des Popupmenu **obj** auf **str**  
*procedure j\_showpopup ( obj , xpos , ypos )*  
Zeigt das Popupmenu an der Koordinate (**xpos,ypos**) an.

Printer
---------

- j\_printer**      *function j\_printer ( frame )*  
Erzeugt ein neues Objekt, das eine Papierseite des Druckes repräsentiert.
- j\_cliprect**      *procedure j\_cliprect ( obj , x , y , width , height )*  
Setzt den Clipping Bereich auf das spezifizierte Rechteck (**x**, **y**, **width**, **height**).
- j\_dispose**      *procedure j\_dispose ( obj )*  
der Printer **obj** wird mit allen Recourcen gelöscht.
- j\_drawarc**      *procedure j\_drawarc ( obj , x , y , rx , ry , arc1 , arc2 )*  
Zeichnet einen Kreisbogen mit dem Mittelpunkt (**x**, **y**) und den Hauptradien **rx** und **ry**. Die Parameter **arc1** und **arc2** bestimmen den Start- bzw. den Endwinkel des Kreisbogens.
- j\_drawcircle**      *procedure j\_drawcircle ( obj , x , y , r )*  
Zeichnet einen Kreis mit dem Mittelpunkt (**x,y**) und dem Radius **r**.
- j\_drawimage**      *procedure j\_drawimage ( obj , image , x , y )*  
Kopiert das Image mit der Eventnummer **image** an die Position (**x**, **y**).
- j\_drawimagesource**      *procedure j\_drawimagesource ( obj , x , y , w , h , r , g , b )*  
Die Funktion stellt ein Bild an der Position (**x,y**) mit der Breite **w** und der Höhe **h** dar. Die Bilddaten werden als rot-, grün- und blau- Kanal in den Arrays **r,g,b** übergeben.
- j\_drawline**      *procedure j\_drawline ( obj , x1 , y1 , x2 , y2 )*  
Zeichnet eine Linie in der momentan gesetzten Vordergrundfarbe von (**x1,y1**) nach (**x2,y2**).
- j\_drawoval**      *procedure j\_drawoval ( obj , x , y , rx , ry )*  
Zeichnet einen Oval mit dem Mittelpunkt (**x,y**) und den Hauptradien **rx** und **ry**.
- j\_drawpixel**      *procedure j\_drawpixel ( obj , x , y )*  
Zeichnet einen Pixel in der momentan gesetzten Vordergrundfarbe an die Position (**x,y**).
- j\_drawpolygon**      *procedure j\_drawpolygon ( obj , len , x , y )*  
Zeichnet ein Polygon durch die Punkte, die durch die Arrays **x** und **y** gegeben sind.
- j\_drawpolyline**      *procedure j\_drawpolyline ( obj , len , x , y )*  
Zeichnet eine Polylinie durch die Punkte, die durch die Arrays **x** und **y** gegeben sind.

- j\_drawrect**      *procedure j\_drawrect ( obj , x , y , width , height )*  
Zeichnet ein Rechteck an die Position **(x,y)** mit der Breite **width** und der Höhe **height**.
- j\_drawroundrect**      *procedure j\_drawroundrect ( obj , x , y , width , height , arcx , arcy )*  
Zeichnet ein Rechteck mit abgerundeten Ecken an die Position **(x,y)** mit der Breite **width** und der Höhe **height**. Die Radien der Ecken werden durch **arcx** und **arcy** bestimmt.
- j\_drawscaledimage**      *procedure j\_drawscaledimage ( obj , image , sx , sy , sw , sh , tx , ty , tw , th )*  
Kopiert einen Ausschnitt des Bildes **image**, der durch die Parameter **(sx, sy, sw, sh)** bestimmt wird an die Position **(tx, ty)** und scaliert den Ausschnitt auf die Breite **tw** und die Höhe **th**
- j\_drawstring**      *procedure j\_drawstring ( obj , x , y , str\$ )*  
Zeichnet den String **str** an die durch die Parameter **(x,y)** festgelegte Koordinate.
- j\_fillarc**      *procedure j\_fillarc ( obj , x , y , rx , ry , arc1 , arc2 )*  
Zeichnet einen gefüllten Kreisbogen mit dem Mittelpunkt **(x,y)** und den Hauptradien **rx** und **ry**. Die Parameter **arc1** und **arc2** bestimmen den Start- bzw. den Endwinkel des Kreisbogens.
- j\_fillcircle**      *procedure j\_fillcircle ( obj , x , y , r )*  
Zeichnet einen gefüllten Kreis mit dem Mittelpunkt **(x,y)** und dem Radius **r**.
- j\_filloval**      *procedure j\_filloval ( obj , x , y , rx , ry )*  
Zeichnet einen gefüllten Oval mit dem Mittelpunkt **(x,y)** und den Hauptradien **rx** und **ry**.
- j\_fillpolygon**      *procedure j\_fillpolygon ( obj , len , x , y )*  
Zeichnet ein gefülltes Polygon durch die Punkte, die durch die Arrays **x** und **y** gegeben sind.
- j\_fillrect**      *procedure j\_fillrect ( obj , x , y , width , height )*  
Zeichnet ein gefülltes Rechteck an die Position **(x,y)** mit der Breite **width** und der Höhe **height**.
- j\_fillroundrect**      *procedure j\_fillroundrect ( obj , x , y , width , height , arcx , arcy )*  
Zeichnet ein gefülltes Rechteck mit abgerundeten Ecken an die Position **(x,y)**. Das Rechteck hat eine Breite von **width** und eine Höhe von **height** Pixeln. Die Form der runden Ecken wird durch die Parameter **arcx** und **arcy** bestimmt.
- j\_print**      *procedure j\_print ( obj )*  
Druckt den Inhalt des Printer .
- j\_setxor**      *procedure j\_setxor ( obj , bool )*  
Schaltet den Painting Mode auf XOR, falls **bool = J\_TRUE** ist. Im XOR Modus heben sich zwei gleiche Graphikbefehle auf.
- j\_translate**      *procedure j\_translate ( obj , x , y )*

Setzt den Ursprung des Koordinatensystems auf  $(\mathbf{x}, \mathbf{y})$ .

Progressbar
-------------

- j\_progressbar**      *function j\_progressbar ( obj , orient )*  
Erzeugt eine Progress Bar mit der Ausrichtung **orient**.
- j\_add**              *procedure j\_add ( obj , cont )*  
Fügt die Progressbar **obj** in den Container **cont** ein.
- j\_componentlistener** *function j\_componentlistener ( obj , kind )*  
Bindet an die Progressbar **obj** einen neuen Componentlistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.
- j\_disable**          *procedure j\_disable ( obj )*  
disabled die Progressbar **obj**, sodaß diese auf keine Benutzereingaben mehr reagiert.
- j\_dispose**         *procedure j\_dispose ( obj )*  
die Progressbar **obj** wird mit allen Recourcen gelöscht.
- j\_enable**           *procedure j\_enable ( obj )*  
enabled die Progressbar **obj**.
- j\_focuslistener**   *function j\_focuslistener ( obj )*  
Bindet an die Progressbar **obj** einen neuen Focuslistener, und liefert dessen Eventnummer zurück.
- j\_getfontascent**   *function j\_getfontascent ( obj )*  
Liefert die Oberlänge (Ascent) des aktuellen Fonts der Progressbar **obj** in Bildschirmpixeln zurück.
- j\_getfontheight**   *function j\_getfontheight ( obj )*  
Liefert die totale Höhe des aktuellen Fonts der Progressbar **obj** in Bildschirmpixeln zurück.
- j\_getheight**        *function j\_getheight ( obj )*  
Liefert die Höhe der Progressbar **obj** zurück.
- j\_getparentid**     *function j\_getparentid ( obj )*  
Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird  $-1$  zurückgegeben.
- j\_getparent**        *function j\_getparent ( obj )*  
Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird  $-1$  zurückgegeben.
- j\_getstringwidth**   *function j\_getstringwidth ( obj , str\$ )*

Liefert die Länge des **str** des aktuellen Fonts der Progressbar **obj** in Bildschirmpixeln zurück.

- j\_getwidth** *function j\_getwidth ( obj )*  
Liefert die Breite der Progressbar **obj** zurück.
- j\_getxpos** *function j\_getxpos ( obj )*  
Liefert die aktuelle horizontale Position der Progressbar **obj** zurück.
- j\_getypos** *function j\_getypos ( obj )*  
Liefert die aktuelle vertikale Position der Progressbar **obj** zurück.
- j\_hide** *procedure j\_hide ( obj )*  
Zeigt die Progressbar **obj** nicht mehr auf dem Bildschirm an.
- j\_isparent** *function j\_isparent ( obj , cont )*  
Liefert J\_TRUE zurück, falls das **cont** die umgebende Komponente von **obj** ist, sonst J\_FALSE
- j\_isvisible** *function j\_isvisible ( obj )*  
Liefert J\_TRUE zurück, falls **obj** sichtbar ist, sonst J\_FALSE
- j\_keylistener** *function j\_keylistener ( obj )*  
Bindet an die Progressbar **obj** einen neuen Keylistener, und liefert dessen Eventnummer zurück.
- j\_mouselistener** *function j\_mouselistener ( obj , kind )*  
Bindet an die Progressbar **obj** einen neuen Mouselistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.
- j\_popupmenu** *function j\_popupmenu ( obj , label\$ )*  
Erzeugt ein neues Popupmenu mit dem Inhalt **label** und liefert dessen Eventnummer zurück.
- j\_print** *procedure j\_print ( obj )*  
Druckt den Inhalt der Progressbar .
- j\_release** *procedure j\_release ( obj )*  
Gibt die Progressbar **obj** aus dem umgebenden Container Element frei.
- j\_setborderpos** *procedure j\_setborderpos ( obj , pos )*  
Positioniert die Progressbar **obj** an die angegebene Position **pos**. Im umgebenden Container muß einen BorderLayout Manager eingestellt sein.
- j\_setcolorbg** *procedure j\_setcolorbg ( obj , r , g , b )*  
Setzt die Hintergrundfarbe auf den übergebenen RGB Wert
- j\_setcolor** *procedure j\_setcolor ( obj , r , g , b )*  
Setzt die Vordergrundfarbe auf den übergebenen RGB Wert
- j\_setcursor** *function j\_setcursor ( obj , cursor )*  
Setzt den Cursor der Progressbar **obj** auf **cursor**.



<b>j_setfocus</b>	<i>function j_setfocus ( obj )</i> Weist den Input Focus der Progressbar <b>obj</b> zu.
<b>j_setfontname</b>	<i>procedure j_setfontname ( obj , name )</i> Setzt den Fonttyp der Progressbar <b>obj</b> auf <b>name</b> .
<b>j_setfont</b>	<i>procedure j_setfont ( obj , name , style , size )</i> Setzt den Font der Progressbar <b>obj</b> entsprechend der Parameterliste.
<b>j_setfontsize</b>	<i>procedure j_setfontsize ( obj , size )</i> Setzt den Fontgröße der Progressbar <b>obj</b> auf <b>size</b> .
<b>j_setfontstyle</b>	<i>procedure j_setfontstyle ( obj , style )</i> Setzt den Fontstyle der Progressbar <b>obj</b> auf <b>style</b> .
<b>j_setnamedcolorbg</b>	<i>procedure j_setnamedcolorbg ( obj , color )</i> Setzt die Hintergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnamedcolor</b>	<i>procedure j_setnamedcolor ( obj , color )</i> Setzt die Vordergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setpos</b>	<i>procedure j_setpos ( obj , xpos , ypos )</i> Verschiebt die Progressbar <b>obj</b> zur angegebenen Koordinate ( <b>xpos,ypos</b> ) an.
<b>j_setsize</b>	<i>procedure j_setsize ( obj , width , height )</i> Setzt Breite und Höhe der Progressbar <b>obj</b> auf <b>width</b> and <b>height</b> .
<b>j_show</b>	<i>procedure j_show ( obj )</i> Zeigt die Progressbar <b>obj</b> auf dem Bildschirm an.

Radiobutton
-------------

- j\_radiobutton**      *function j\_radiobutton ( obj , label\$ )*  
Erzeugt einen Radiobutton mit dem Textinhalt **label** und liefert dessen Eventnummer zurück.
- j\_add**                *procedure j\_add ( obj , cont )*  
Fügt den Radiobutton **obj** in den Container **cont** ein.
- j\_componentlistener** *function j\_componentlistener ( obj , kind )*  
Bindet an den Radiobutton **obj** einen neuen Componentlistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.
- j\_disable**            *procedure j\_disable ( obj )*  
disabled den Radiobutton **obj**, sodaß diese auf keine Benutzereingaben mehr reagiert.
- j\_dispose**            *procedure j\_dispose ( obj )*  
der Radiobutton **obj** wird mit allen Recourcen gelöscht.
- j\_enable**             *procedure j\_enable ( obj )*  
enabled den Radiobutton **obj**.
- j\_focuslistener**     *function j\_focuslistener ( obj )*  
Bindet an den Radiobutton **obj** einen neuen Focuslistener, und liefert dessen Eventnummer zurück.
- j\_getfontascent**     *function j\_getfontascent ( obj )*  
Liefert die Oberlänge (Ascent) des aktuellen Fonts des Radiobutton **obj** in Bildschirmpixeln zurück.
- j\_getfontheight**     *function j\_getfontheight ( obj )*  
Liefert die totale Höhe des aktuellen Fonts des Radiobutton **obj** in Bildschirmpixeln zurück.
- j\_getheight**         *function j\_getheight ( obj )*  
Liefert die Höhe des Radiobutton **obj** zurück.
- j\_getparentid**       *function j\_getparentid ( obj )*  
Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird  $-1$  zurückgegeben.
- j\_getparent**         *function j\_getparent ( obj )*  
Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird  $-1$  zurückgegeben.
- j\_getstate**          *function j\_getstate ( obj )*

Liefert J\_TRUE , falls der Radiobutton selektiert ist, sonst J\_FALSE .

<b>j_getstringwidth</b>	<i>function j_getstringwidth ( obj , str\$ )</i> Liefert die Länge des <b>str</b> des aktuellen Fonts des Radiobutton <b>obj</b> in Bildschirmpixeln zurück.
<b>j_gettext</b>	<i>function\$ j_gettext ( obj )</i> Liefert den Textinhalt des Radiobutton .
<b>j_getwidth</b>	<i>function j_getwidth ( obj )</i> Liefert die Breite des Radiobutton <b>obj</b> zurück.
<b>j_getxpos</b>	<i>function j_getxpos ( obj )</i> Liefert die aktuelle horizontale Position des Radiobutton <b>obj</b> zurück.
<b>j_getypos</b>	<i>function j_getypos ( obj )</i> Liefert die aktuelle vertikale Position des Radiobutton <b>obj</b> zurück.
<b>j_hide</b>	<i>procedure j_hide ( obj )</i> Zeigt den Radiobutton <b>obj</b> nicht mehr auf dem Bildschirm an.
<b>j_isparent</b>	<i>function j_isparent ( obj , cont )</i> Liefert J_TRUE zurück, falls das <b>cont</b> die umgebende Komponente von <b>obj</b> ist, sonst J_FALSE
<b>j_isvisible</b>	<i>function j_isvisible ( obj )</i> Liefert J_TRUE zurück, falls <b>obj</b> sichtbar ist, sonst J_FALSE
<b>j_keylistener</b>	<i>function j_keylistener ( obj )</i> Bindet an den Radiobutton <b>obj</b> einen neuen Keylistener, und liefert dessen Eventnummer zurück.
<b>j_mouselistener</b>	<i>function j_mouselistener ( obj , kind )</i> Bindet an den Radiobutton <b>obj</b> einen neuen Mouselistener, und liefert dessen Eventnummer zurück. Der Parameter <b>kind</b> bestimmt, wann ein Event ausgelöst wird.
<b>j_popupmenu</b>	<i>function j_popupmenu ( obj , label\$ )</i> Erzeugt ein neues Popumenu mit dem Inhalt <b>label</b> und liefert dessen Eventnummer zurück.
<b>j_print</b>	<i>procedure j_print ( obj )</i> Druckt den Inhalt des Radiobutton .
<b>j_release</b>	<i>procedure j_release ( obj )</i> Gibt den Radiobutton <b>obj</b> aus dem umgebenden Container Element frei.
<b>j_setborderpos</b>	<i>procedure j_setborderpos ( obj , pos )</i> Positioniert den Radiobutton <b>obj</b> an die angegebene Position <b>pos</b> . Im umgebenden Container muß einen Borderlayout Manager eingestellt sein.
<b>j_setcolorbg</b>	<i>procedure j_setcolorbg ( obj , r , g , , b )</i> Setzt die Hintergrundfarbe auf den übergebenen RGB Wert

<b>j_setcolor</b>	<i>procedure j_setcolor ( obj , r , g , , b )</i> Setzt die Vordergrundfarbe auf den übergebenen RGB Wert
<b>j_setcursor</b>	<i>function j_setcursor ( obj , cursor )</i> Setzt den Cursor des Radiobutton <b>obj</b> auf <b>cursor</b> .
<b>j_setfocus</b>	<i>function j_setfocus ( obj )</i> Weist den Input Focus dem Radiobutton <b>obj</b> zu.
<b>j_setfontname</b>	<i>procedure j_setfontname ( obj , name )</i> Setzt den Fonttyp des Radiobutton <b>obj</b> auf <b>name</b> .
<b>j_setfont</b>	<i>procedure j_setfont ( obj , name , style , size )</i> Setzt den Font des Radiobutton <b>obj</b> entsprechend der Parameterliste.
<b>j_setfontsize</b>	<i>procedure j_setfontsize ( obj , size )</i> Setzt den Fontgröße des Radiobutton <b>obj</b> auf <b>size</b> .
<b>j_setfontstyle</b>	<i>procedure j_setfontstyle ( obj , style )</i> Setzt den Fontstyle des Radiobutton <b>obj</b> auf <b>style</b> .
<b>j_setnamedcolorbg</b>	<i>procedure j_setnamedcolorbg ( obj , color )</i> Setzt die Hintergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnamedcolor</b>	<i>procedure j_setnamedcolor ( obj , color )</i> Setzt die Vordergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setpos</b>	<i>procedure j_setpos ( obj , xpos , ypos )</i> Verschiebt den Radiobutton <b>obj</b> zur angegebenen Koordinate ( <b>xpos.ypos</b> ) an.
<b>j_setradiogroup</b>	<i>function j_setradiogroup ( rbutton, , rgroup )</i> Weist dem Radiobutton <b>rbutton</b> die Radiogroup <b>rgroup</b> zu. Der Radiobutton wird aus der alten Radiogroup entfernt.
<b>j_setsize</b>	<i>procedure j_setsize ( obj , width , height )</i> Setzt Breite und Höhe des Radiobutton <b>obj</b> auf <b>width</b> and <b>height</b> .
<b>j_setstate</b>	<i>procedure j_setstate ( obj , bool )</i> Ist <b>bool</b> J.TRUE so wird den Radiobutton selectiert.
<b>j_settext</b>	<i>procedure j_settext ( obj , str\$ )</i> Ändert den Textinhalt des Radiobutton <b>obj</b> auf <b>str</b>
<b>j_show</b>	<i>procedure j_show ( obj )</i> Zeigt den Radiobutton <b>obj</b> auf dem Bildschirm an.

Sevensegment
--------------

- j\_sevensegment**     *function j\_sevensegment ( obj , color )*  
Erzeugt eine Siebensegment Anzeige der Farbe **color**.
- j\_add**     *procedure j\_add ( obj , cont )*  
Fügt die Siebensegment-Anzeige **obj** in den Container **cont** ein.
- j\_componentlistener** *function j\_componentlistener ( obj , kind )*  
Bindet an die Siebensegment-Anzeige **obj** einen neuen Componentlistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.
- j\_disable**     *procedure j\_disable ( obj )*  
disabled die Siebensegment-Anzeige **obj**, sodaß diese auf keine Benutzereingaben mehr reagiert.
- j\_dispose**     *procedure j\_dispose ( obj )*  
die Siebensegment-Anzeige **obj** wird mit allen Recourcen gelöscht.
- j\_enable**     *procedure j\_enable ( obj )*  
enabled die Siebensegment-Anzeige **obj**.
- j\_focuslistener**     *function j\_focuslistener ( obj )*  
Bindet an die Siebensegment-Anzeige **obj** einen neuen Focuslistener, und liefert dessen Eventnummer zurück.
- j\_getfontascent**     *function j\_getfontascent ( obj )*  
Liefert die Oberlänge (Ascent) des aktuellen Fonts der Siebensegment-Anzeige **obj** in Bildschirmpixeln zurück.
- j\_getfontheight**     *function j\_getfontheight ( obj )*  
Liefert die totale Höhe des aktuellen Fonts der Siebensegment-Anzeige **obj** in Bildschirmpixeln zurück.
- j\_getheight**     *function j\_getheight ( obj )*  
Liefert die Höhe der Siebensegment-Anzeige **obj** zurück.
- j\_getparentid**     *function j\_getparentid ( obj )*  
Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird **-1** zurückgegeben.
- j\_getparent**     *function j\_getparent ( obj )*  
Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird **-1** zurückgegeben.
- j\_getstringwidth**     *function j\_getstringwidth ( obj , str\$ )*

Liefert die Länge des **str** des aktuellen Fonts der Siebensegment-Anzeige **obj** in Bildschirmpixeln zurück.

- j\_getwidth** *function j\_getwidth ( obj )*  
Liefert die Breite der Siebensegment-Anzeige **obj** zurück.
- j\_getxpos** *function j\_getxpos ( obj )*  
Liefert die aktuelle horizontale Position der Siebensegment-Anzeige **obj** zurück.
- j\_getypos** *function j\_getypos ( obj )*  
Liefert die aktuelle vertikale Position der Siebensegment-Anzeige **obj** zurück.
- j\_hide** *procedure j\_hide ( obj )*  
Zeigt die Siebensegment-Anzeige **obj** nicht mehr auf dem Bildschirm an.
- j\_isparent** *function j\_isparent ( obj , cont )*  
Liefert J\_TRUE zurück, falls das **cont** die umgebende Komponente von **obj** ist, sonst J\_FALSE
- j\_isvisible** *function j\_isvisible ( obj )*  
Liefert J\_TRUE zurück, falls **obj** sichtbar ist, sonst J\_FALSE
- j\_keylistener** *function j\_keylistener ( obj )*  
Bindet an die Siebensegment-Anzeige **obj** einen neuen Keylistener, und liefert dessen Eventnummer zurück.
- j\_mouselistener** *function j\_mouselistener ( obj , kind )*  
Bindet an die Siebensegment-Anzeige **obj** einen neuen Mouselistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.
- j\_popupmenu** *function j\_popupmenu ( obj , label\$ )*  
Erzeugt ein neues Popupmenu mit dem Inhalt **label** und liefert dessen Eventnummer zurück.
- j\_print** *procedure j\_print ( obj )*  
Druckt den Inhalt der Siebensegment-Anzeige .
- j\_release** *procedure j\_release ( obj )*  
Gibt die Siebensegment-Anzeige **obj** aus dem umgebenden Container Element frei.
- j\_setborderpos** *procedure j\_setborderpos ( obj , pos )*  
Positioniert die Siebensegment-Anzeige **obj** an die angegebene Position **pos**. Im umgebenden Container muß einen BorderLayout Manager eingestellt sein.
- j\_setcolorbg** *procedure j\_setcolorbg ( obj , r , g , b )*  
Setzt die Hintergrundfarbe auf den übergebenen RGB Wert
- j\_setcolor** *procedure j\_setcolor ( obj , r , g , b )*  
Setzt die Vordergrundfarbe auf den übergebenen RGB Wert

<b>j_setcursor</b>	<i>function j_setcursor ( obj , cursor )</i> Setzt den Cursor der Siebensegment-Anzeige <b>obj</b> auf <b>cursor</b> .
<b>j_setfocus</b>	<i>function j_setfocus ( obj )</i> Weist den Input Focus der Siebensegment-Anzeige <b>obj</b> zu.
<b>j_setfontname</b>	<i>procedure j_setfontname ( obj , name )</i> Setzt den Fonttyp der Siebensegment-Anzeige <b>obj</b> auf <b>name</b> .
<b>j_setfont</b>	<i>procedure j_setfont ( obj , name , style , size )</i> Setzt den Font der Siebensegment-Anzeige <b>obj</b> entsprechend der Parameterliste.
<b>j_setfontsize</b>	<i>procedure j_setfontsize ( obj , size )</i> Setzt den Fontgröße der Siebensegment-Anzeige <b>obj</b> auf <b>size</b> .
<b>j_setfontstyle</b>	<i>procedure j_setfontstyle ( obj , style )</i> Setzt den Fontstyle der Siebensegment-Anzeige <b>obj</b> auf <b>style</b> .
<b>j_setnamedcolorbg</b>	<i>procedure j_setnamedcolorbg ( obj , color )</i> Setzt die Hintergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnamedcolor</b>	<i>procedure j_setnamedcolor ( obj , color )</i> Setzt die Vordergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setpos</b>	<i>procedure j_setpos ( obj , xpos , ypos )</i> Verschiebt die Siebensegment-Anzeige <b>obj</b> zur angegebenen Koordinate ( <b>xpos,ypos</b> ) an.
<b>j_setsize</b>	<i>procedure j_setsize ( obj , width , height )</i> Setzt Breite und Höhe der Siebensegment-Anzeige <b>obj</b> auf <b>width</b> and <b>height</b> .
<b>j_setvalue</b>	<i>procedure j_setvalue ( obj , val )</i> Stellt den aktuellen Wert der Siebensegment-Anzeige <b>obj</b> auf <b>val</b> .
<b>j_show</b>	<i>procedure j_show ( obj )</i> Zeigt die Siebensegment-Anzeige <b>obj</b> auf dem Bildschirm an.

Scrollpane
------------

- j\_scrollpane**      *function j\_scrollpane ( obj )*  
Erzeugt eine neue Scrollpane und gibt dessen Eventnummer zurück.
- j\_add**              *procedure j\_add ( obj , cont )*  
Fügt die Scrollpane **obj** in den Container **cont** ein.
- j\_componentlistener** *function j\_componentlistener ( obj , kind )*  
Bindet an die Scrollpane **obj** einen neuen Componentlistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.
- j\_disable**          *procedure j\_disable ( obj )*  
disabled die Scrollpane **obj**, sodaß diese auf keine Benutzereingaben mehr reagiert.
- j\_dispose**          *procedure j\_dispose ( obj )*  
die Scrollpane **obj** wird mit allen Recourcen gelöscht.
- j\_enable**            *procedure j\_enable ( obj )*  
enabled die Scrollpane **obj**.
- j\_focuslistener**    *function j\_focuslistener ( obj )*  
Bindet an die Scrollpane **obj** einen neuen Focuslistener, und liefert dessen Eventnummer zurück.
- j\_getfontascent**    *function j\_getfontascent ( obj )*  
Liefert die Oberlänge (Ascent) des aktuellen Fonts der Scrollpane **obj** in Bildschirmpixeln zurück.
- j\_getfontheight**    *function j\_getfontheight ( obj )*  
Liefert die totale Höhe des aktuellen Fonts der Scrollpane **obj** in Bildschirmpixeln zurück.
- j\_getheight**        *function j\_getheight ( obj )*  
Liefert die Höhe der Scrollpane **obj** zurück.
- j\_getparentid**      *function j\_getparentid ( obj )*  
Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird  $-1$  zurückgegeben.
- j\_getparent**        *function j\_getparent ( obj )*  
Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird  $-1$  zurückgegeben.
- j\_getstringwidth**   *function j\_getstringwidth ( obj , str\$ )*  
Liefert die Länge des **str** des aktuellen Fonts der Scrollpane **obj** in Bildschirmpixeln zurück.



<b>j_getviewportheight</b>	<i>function j_getviewportheight ( obj )</i> Gibt die Höhe des sichtbaren Teils der Scrollpane <b>obj</b> zurück.
<b>j_getviewportwidth</b>	<i>function j_getviewportwidth ( obj )</i> Gibt die Breite des sichtbaren Teils der Scrollpane <b>obj</b> zurück.
<b>j_getwidth</b>	<i>function j_getwidth ( obj )</i> Liefert die Breite der Scrollpane <b>obj</b> zurück.
<b>j_getxpos</b>	<i>function j_getxpos ( obj )</i> Liefert die aktuelle horizontale Position der Scrollpane <b>obj</b> zurück.
<b>j_getypos</b>	<i>function j_getypos ( obj )</i> Liefert die aktuelle vertikale Position der Scrollpane <b>obj</b> zurück.
<b>j_hide</b>	<i>procedure j_hide ( obj )</i> Zeigt die Scrollpane <b>obj</b> nicht mehr auf dem Bildschirm an.
<b>j_hscrollbar</b>	<i>function j_hscrollbar ( obj )</i> Erzeugt eine horizontale Scrollbar und gibt dessen Eventnummer zurück.
<b>j_isparent</b>	<i>function j_isparent ( obj , cont )</i> Liefert J_TRUE zurück, falls das <b>cont</b> die umgebende Komponente von <b>obj</b> ist, sonst J_FALSE
<b>j_isvisible</b>	<i>function j_isvisible ( obj )</i> Liefert J_TRUE zurück, falls <b>obj</b> sichtbar ist, sonst J_FALSE
<b>j_keylistener</b>	<i>function j_keylistener ( obj )</i> Bindet an die Scrollpane <b>obj</b> einen neuen Keylistener, und liefert dessen Eventnummer zurück.
<b>j_mouselistener</b>	<i>function j_mouselistener ( obj , kind )</i> Bindet an die Scrollpane <b>obj</b> einen neuen Mouselistener, und liefert dessen Eventnummer zurück. Der Parameter <b>kind</b> bestimmt, wann ein Event ausgelöst wird.
<b>j_popupmenu</b>	<i>function j_popupmenu ( obj , label\$ )</i> Erzeugt ein neues Popupmenu mit dem Inhalt <b>label</b> und liefert dessen Eventnummer zurück.
<b>j_print</b>	<i>procedure j_print ( obj )</i> Druckt den Inhalt der Scrollpane .
<b>j_release</b>	<i>procedure j_release ( obj )</i> Gibt die Scrollpane <b>obj</b> aus dem umgebenden Container Element frei.
<b>j_setborderpos</b>	<i>procedure j_setborderpos ( obj , pos )</i> Positioniert die Scrollpane <b>obj</b> an die angegebene Position <b>pos</b> . Im umgebenden Container muß einen BorderLayout Manager eingestellt sein.
<b>j_setcolorbg</b>	<i>procedure j_setcolorbg ( obj , r , g , b )</i>

	Setzt die Hintergrundfarbe auf den übergebenen RGB Wert
<b>j_setcolor</b>	<i>procedure j_setcolor ( obj , r , g , b )</i> Setzt die Vordergrundfarbe auf den übergebenen RGB Wert
<b>j_setcursor</b>	<i>function j_setcursor ( obj , cursor )</i> Setzt den Cursor der Scrollpane <b>obj</b> auf <b>cursor</b> .
<b>j_setfocus</b>	<i>function j_setfocus ( obj )</i> Weist den Input Focus der Scrollpane <b>obj</b> zu.
<b>j_setfontname</b>	<i>procedure j_setfontname ( obj , name )</i> Setzt den Fonttyp der Scrollpane <b>obj</b> auf <b>name</b> .
<b>j_setfont</b>	<i>procedure j_setfont ( obj , name , style , size )</i> Setzt den Font der Scrollpane <b>obj</b> entsprechend der Parameterliste.
<b>j_setfontsize</b>	<i>procedure j_setfontsize ( obj , size )</i> Setzt den Fontgröße der Scrollpane <b>obj</b> auf <b>size</b> .
<b>j_setfontstyle</b>	<i>procedure j_setfontstyle ( obj , style )</i> Setzt den Fontstyle der Scrollpane <b>obj</b> auf <b>style</b> .
<b>j_setnamedcolorbg</b>	<i>procedure j_setnamedcolorbg ( obj , color )</i> Setzt die Hintergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnamedcolor</b>	<i>procedure j_setnamedcolor ( obj , color )</i> Setzt die Vordergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setpos</b>	<i>procedure j_setpos ( obj , xpos , ypos )</i> Verschiebt die Scrollpane <b>obj</b> zur angegebenen Koordinate ( <b>xpos,ypos</b> ) an.
<b>j_setsize</b>	<i>procedure j_setsize ( obj , width , height )</i> Setzt Breite und Höhe der Scrollpane <b>obj</b> auf <b>width</b> and <b>height</b> .
<b>j_show</b>	<i>procedure j_show ( obj )</i> Zeigt die Scrollpane <b>obj</b> auf dem Bildschirm an.
<b>j_vscrollbar</b>	<i>function j_vscrollbar ( obj )</i> Erzeugt eine vertikale Scrollbar und gibt dessen Eventnummer zurück.

Textarea
----------

<b>j_textarea</b>	<i>function j_textarea ( obj , rows , columns )</i> Erzeugt eine neue Textarea und gibt dessen Eventnummer zurück.
<b>j_add</b>	<i>procedure j_add ( obj , cont )</i> Fügt die Textarea <b>obj</b> in den Container <b>cont</b> ein.
<b>j_appendtext</b>	<i>procedure j_appendtext ( obj , text\$ )</i> Hängt den Text <b>text</b> am Ende von <b>obj</b> an.
<b>j_componentlistener</b>	<i>function j_componentlistener ( obj , kind )</i> Bindet an die Textarea <b>obj</b> einen neuen Componentlistener, und liefert dessen Eventnummer zurück. Der Parameter <b>kind</b> bestimmt, wann ein Event ausgelöst wird.
<b>j_delete</b>	<i>procedure j_delete ( obj , start , end )</i> löscht den Text von Position <b>start</b> bis Position <b>end</b> .
<b>j_disable</b>	<i>procedure j_disable ( obj )</i> disabled die Textarea <b>obj</b> , sodaß diese auf keine Benutzereingaben mehr reagiert.
<b>j_dispose</b>	<i>procedure j_dispose ( obj )</i> die Textarea <b>obj</b> wird mit allen Recourcen gelöscht.
<b>j_enable</b>	<i>procedure j_enable ( obj )</i> enabled die Textarea <b>obj</b> .
<b>j_focuslistener</b>	<i>function j_focuslistener ( obj )</i> Bindet an die Textarea <b>obj</b> einen neuen Focuslistener, und liefert dessen Eventnummer zurück.
<b>j_getcolumns</b>	<i>procedure j_getcolumns ( obj )</i> Gibt die Anzahl der Spalten von <b>obj</b> zurück.
<b>j_getcurpos</b>	<i>function j_getcurpos ( obj )</i> Liefert die Position des Text Cursors zurück.
<b>j_getfontascent</b>	<i>function j_getfontascent ( obj )</i> Liefert die Oberlänge (Ascent) des aktuellen Fonts der Textarea <b>obj</b> in Bildschirmpixeln zurück.
<b>j_getfontheight</b>	<i>function j_getfontheight ( obj )</i> Liefert die totale Höhe des aktuellen Fonts der Textarea <b>obj</b> in Bildschirmpixeln zurück.
<b>j_getheight</b>	<i>function j_getheight ( obj )</i>

	Liefert die Höhe der Textarea <b>obj</b> zurück.
<b>j_getlength</b>	<i>function j_getlength ( obj )</i> Liefert die Länge des Textinhaltes die Textarea zurück.
<b>j_getparentid</b>	<i>function j_getparentid ( obj )</i> Liefert die Eventnummer des umgebenen Containers von <b>obj</b> zurück. Ist <b>obj</b> ein Frame wird $-1$ zurückgegeben.
<b>j_getparent</b>	<i>function j_getparent ( obj )</i> Liefert die Eventnummer des umgebenen Containers von <b>obj</b> zurück. Ist <b>obj</b> ein Frame wird $-1$ zurückgegeben.
<b>j_getrows</b>	<i>procedure j_getrows ( obj )</i> Gibt die Anzahl der Reihen von <b>obj</b> zurück.
<b>j_getselend</b>	<i>function j_getselend ( obj )</i> Liefert die Endposition des selectierten Textes.
<b>j_getselstart</b>	<i>function j_getselstart ( obj )</i> Liefert die Anfangsposition des selectierten Textes.
<b>j_getseltext</b>	<i>function\$ j_getseltext ( obj )</i> Liefert den selectierten Text die Textarea <b>obj</b> zurück.
<b>j_getstringwidth</b>	<i>function j_getstringwidth ( obj , str\$ )</i> Liefert die Länge des <b>str</b> des aktuellen Fonts der Textarea <b>obj</b> in Bildschirm-pixeln zurück.
<b>j_gettext</b>	<i>function\$ j_gettext ( obj )</i> Liefert den Textinhalt der Textarea .
<b>j_getwidth</b>	<i>function j_getwidth ( obj )</i> Liefert die Breite der Textarea <b>obj</b> zurück.
<b>j_getxpos</b>	<i>function j_getxpos ( obj )</i> Liefert die aktuelle horizontale Position der Textarea <b>obj</b> zurück.
<b>j_getypos</b>	<i>function j_getypos ( obj )</i> Liefert die aktuelle vertikale Position der Textarea <b>obj</b> zurück.
<b>j_hide</b>	<i>procedure j_hide ( obj )</i> Zeigt die Textarea <b>obj</b> nicht mehr auf dem Bildschirm an.
<b>j_inserttext</b>	<i>procedure j_inserttext ( obj , text\$ , pos )</i> Fügt zusätzlichen Text <b>text</b> an der Position <b>pos</b> ein.
<b>j_isparent</b>	<i>function j_isparent ( obj , cont )</i> Liefert J_TRUE zurück, falls das <b>cont</b> die umgebende Komponente von <b>obj</b> ist, sonst J_FALSE
<b>j_isvisible</b>	<i>function j_isvisible ( obj )</i>

Liefert J\_TRUE zurück, falls **obj** sichtbar ist, sonst J\_FALSE

<b>j_keylistener</b>	<i>function j_keylistener ( obj )</i> Bindet an die Textarea <b>obj</b> einen neuen Keylistener, und liefert dessen Eventnummer zurück.
<b>j_mouselistener</b>	<i>function j_mouselistener ( obj , kind )</i> Bindet an die Textarea <b>obj</b> einen neuen Mouselistener, und liefert dessen Eventnummer zurück. Der Parameter <b>kind</b> bestimmt, wann ein Event ausgelöst wird.
<b>j_popupmenu</b>	<i>function j_popupmenu ( obj , label\$ )</i> Erzeugt ein neues Popupmenu mit dem Inhalt <b>label</b> und liefert dessen Eventnummer zurück.
<b>j_print</b>	<i>procedure j_print ( obj )</i> Druckt den Inhalt der Textarea .
<b>j_release</b>	<i>procedure j_release ( obj )</i> Gibt die Textarea <b>obj</b> aus dem umgebenden Container Element frei.
<b>j_replacetext</b>	<i>procedure j_replacetext ( obj , text\$ , start , end )</i> Ersetzt den Text zwischen Position <b>start</b> bis <b>end</b> durch <b>text</b> .
<b>j_selectall</b>	<i>procedure j_selectall ( obj )</i> Selectiert den gesamten Textinhalt der Textarea .
<b>j_selecttext</b>	<i>procedure j_selecttext ( obj , start , end )</i> Selectiert den Text von Position <b>start</b> bis <b>end</b> .
<b>j_setborderpos</b>	<i>procedure j_setborderpos ( obj , pos )</i> Positioniert die Textarea <b>obj</b> an die angegebene Position <b>pos</b> . Im umgebenden Container muß einen Borderlayout Manager eingestellt sein.
<b>j_setcolorbg</b>	<i>procedure j_setcolorbg ( obj , r , g , b )</i> Setzt die Hintergrundfarbe auf den übergebenen RGB Wert
<b>j_setcolor</b>	<i>procedure j_setcolor ( obj , r , g , b )</i> Setzt die Vordergrundfarbe auf den übergebenen RGB Wert
<b>j_setcolumns</b>	<i>procedure j_setcolumns ( obj , columns )</i> Setzt die Anzahl Spalten von <b>obj</b> auf <b>columns</b> .
<b>j_setcurpos</b>	<i>procedure j_setcurpos ( obj , pos )</i> Setzt den Text Cursor auf die Position <b>pos</b> .
<b>j_setcursor</b>	<i>function j_setcursor ( obj , cursor )</i> Setzt den Cursor der Textarea <b>obj</b> auf <b>cursor</b> .
<b>j_seteditable</b>	<i>procedure j_seteditable ( obj , bool )</i> Ist <b>bool</b> J_TRUE , so ist die Textarea editierbar, ansonsten ist die Textarea nur lesbar für den Benutzer.
<b>j_setfocus</b>	<i>function j_setfocus ( obj )</i>

	Weist den Input Focus der Textarea <b>obj</b> zu.
<b>j_setfontname</b>	<i>procedure j_setfontname ( obj , name )</i> Setzt den Fonttyp der Textarea <b>obj</b> auf <b>name</b> .
<b>j_setfont</b>	<i>procedure j_setfont ( obj , name , style , size )</i> Setzt den Font der Textarea <b>obj</b> entsprechend der Parameterliste.
<b>j_setfontsize</b>	<i>procedure j_setfontsize ( obj , size )</i> Setzt den Fontgröße der Textarea <b>obj</b> auf <b>size</b> .
<b>j_setfontstyle</b>	<i>procedure j_setfontstyle ( obj , style )</i> Setzt den Fontstyle der Textarea <b>obj</b> auf <b>style</b> .
<b>j_setnamedcolorbg</b>	<i>procedure j_setnamedcolorbg ( obj , color )</i> Setzt die Hintergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnamedcolor</b>	<i>procedure j_setnamedcolor ( obj , color )</i> Setzt die Vordergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setpos</b>	<i>procedure j_setpos ( obj , xpos , ypos )</i> Verschiebt die Textarea <b>obj</b> zur angegebenen Koordinate ( <b>xpos,ypos</b> ) an.
<b>j_setrows</b>	<i>procedure j_setrows ( obj , rows )</i> Setzt die Anzahl Reihen von <b>obj</b> auf <b>rows</b> .
<b>j_setsize</b>	<i>procedure j_setsize ( obj , width , height )</i> Setzt Breite und Höhe der Textarea <b>obj</b> auf <b>width</b> and <b>height</b> .
<b>j_settext</b>	<i>procedure j_settext ( obj , str\$ )</i> Ändert den Textinhalt der Textarea <b>obj</b> auf <b>str</b>
<b>j_show</b>	<i>procedure j_show ( obj )</i> Zeigt die Textarea <b>obj</b> auf dem Bildschirm an.

Textfield
-----------

- j\_textfield**      *function j\_textfield ( obj , columns )*  
Erzeugt eine neues Textfield und gibt dessen Eventnummer zurück.
- j\_add**              *procedure j\_add ( obj , cont )*  
Fügt das Textfield **obj** in den Container **cont** ein.
- j\_componentlistener** *function j\_componentlistener ( obj , kind )*  
Bindet an das Textfield **obj** einen neuen Componentlistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.
- j\_disable**          *procedure j\_disable ( obj )*  
disabled das Textfield **obj**, sodaß diese auf keine Benutzereingaben mehr reagiert.
- j\_dispose**          *procedure j\_dispose ( obj )*  
das Textfield **obj** wird mit allen Recourcen gelöscht.
- j\_enable**            *procedure j\_enable ( obj )*  
enabled das Textfield **obj**.
- j\_focuslistener**    *function j\_focuslistener ( obj )*  
Bindet an das Textfield **obj** einen neuen Focuslistener, und liefert dessen Eventnummer zurück.
- j\_getcolumns**        *procedure j\_getcolumns ( obj )*  
Gibt die Anzahl der Spalten von **obj** zurück.
- j\_getcurpos**         *function j\_getcurpos ( obj )*  
Liefert die Position des Text Cursors zurück.
- j\_getfontascent**    *function j\_getfontascent ( obj )*  
Liefert die Oberlänge (Ascent) des aktuellen Fonts des Textfield **obj** in Bildschirmpixeln zurück.
- j\_getfontheight**    *function j\_getfontheight ( obj )*  
Liefert die totale Höhe des aktuellen Fonts des Textfield **obj** in Bildschirmpixeln zurück.
- j\_getheight**         *function j\_getheight ( obj )*  
Liefert die Höhe des Textfield **obj** zurück.
- j\_getlength**         *function j\_getlength ( obj )*  
Liefert die Länge des Textinhaltes das Textfield zurück.
- j\_getparentid**       *function j\_getparentid ( obj )*

	Liefert die Eventnummer des umgebenen Containers von <b>obj</b> zurück. Ist <b>obj</b> ein Frame wird $-1$ zurückgegeben.
<b>j_getparent</b>	<i>function j_getparent ( obj )</i> Liefert die Eventnummer des umgebenen Containers von <b>obj</b> zurück. Ist <b>obj</b> ein Frame wird $-1$ zurückgegeben.
<b>j_getselend</b>	<i>function j_getselend ( obj )</i> Liefert die Endposition des selectierten Textes.
<b>j_getselstart</b>	<i>function j_getselstart ( obj )</i> Liefert die Anfangsposition des selectierten Textes.
<b>j_getseltext</b>	<i>function\$ j_getseltext ( obj )</i> Liefert den selectierten Text das Textfield <b>obj</b> zurück.
<b>j_getstringwidth</b>	<i>function j_getstringwidth ( obj , str\$ )</i> Liefert die Länge des <b>str</b> des aktuellen Fonts des Textfield <b>obj</b> in Bildschirm-pixeln zurück.
<b>j_gettext</b>	<i>function\$ j_gettext ( obj )</i> Liefert den Textinhalt des Textfield .
<b>j_getwidth</b>	<i>function j_getwidth ( obj )</i> Liefert die Breite des Textfield <b>obj</b> zurück.
<b>j_getxpos</b>	<i>function j_getxpos ( obj )</i> Liefert die aktuelle horizontale Position des Textfield <b>obj</b> zurück.
<b>j_getypos</b>	<i>function j_getypos ( obj )</i> Liefert die aktuelle vertikale Position des Textfield <b>obj</b> zurück.
<b>j_hide</b>	<i>procedure j_hide ( obj )</i> Zeigt das Textfield <b>obj</b> nicht mehr auf dem Bildschirm an.
<b>j_isparent</b>	<i>function j_isparent ( obj , cont )</i> Liefert J_TRUE zurück, falls das <b>cont</b> die umgebende Komponente von <b>obj</b> ist, sonst J_FALSE
<b>j_isvisible</b>	<i>function j_isvisible ( obj )</i> Liefert J_TRUE zurück, falls <b>obj</b> sichtbar ist, sonst J_FALSE
<b>j_keylistener</b>	<i>function j_keylistener ( obj )</i> Bindet an das Textfield <b>obj</b> einen neuen Keylistener, und liefert dessen Eventnummer zurück.
<b>j_mouselistener</b>	<i>function j_mouselistener ( obj , kind )</i> Bindet an das Textfield <b>obj</b> einen neuen Mouselistener, und liefert dessen Eventnummer zurück. Der Parameter <b>kind</b> bestimmt, wann ein Event ausgelöst wird.
<b>j_popupmenu</b>	<i>function j_popupmenu ( obj , label\$ )</i>



	Erzeugt ein neues Popumenu mit dem Inhalt <b>label</b> und liefert dessen Eventnummer zurück.
<b>j_print</b>	<i>procedure j_print ( obj )</i> Druckt den Inhalt des Textfield .
<b>j_release</b>	<i>procedure j_release ( obj )</i> Gibt das Textfield <b>obj</b> aus dem umgebenden Container Element frei.
<b>j_selectall</b>	<i>procedure j_selectall ( obj )</i> Selectiert den gesamten Textinhalt des Textfield .
<b>j_selecttext</b>	<i>procedure j_selecttext ( obj , start , end )</i> Selectiert den Text von Position <b>start</b> bis <b>end</b> .
<b>j_setborderpos</b>	<i>procedure j_setborderpos ( obj , pos )</i> Positioniert das Textfield <b>obj</b> an die angegebene Position <b>pos</b> . Im umgebenden Container muß einen BorderLayout Manager eingestellt sein.
<b>j_setcolorbg</b>	<i>procedure j_setcolorbg ( obj , r , g , b )</i> Setzt die Hintergrundfarbe auf den übergebenen RGB Wert
<b>j_setcolor</b>	<i>procedure j_setcolor ( obj , r , g , b )</i> Setzt die Vordergrundfarbe auf den übergebenen RGB Wert
<b>j_setcolumns</b>	<i>procedure j_setcolumns ( obj , columns )</i> Setzt die Anzahl Spalten von <b>obj</b> auf <b>columns</b> .
<b>j_setcurpos</b>	<i>procedure j_setcurpos ( obj , pos )</i> Setzt den Text Cursor auf die Position <b>pos</b> .
<b>j_setcursor</b>	<i>function j_setcursor ( obj , cursor )</i> Setzt den Cursor des Textfield <b>obj</b> auf <b>cursor</b> .
<b>j_setechochar</b>	<i>procedure j_setechochar ( obj , chr )</i> Bestimmt das Zeichen <b>chr</b> als Echo Zeichen für alle Benutzereingaben in das Textfield .
<b>j_seteditable</b>	<i>procedure j_seteditable ( obj , bool )</i> Ist <b>bool</b> <b>J_TRUE</b> , so ist das Textfield editierbar, ansonsten ist das Textfield nur lesbar für den Benutzer.
<b>j_setfocus</b>	<i>function j_setfocus ( obj )</i> Weist den Input Focus dem Textfield <b>obj</b> zu.
<b>j_setfontname</b>	<i>procedure j_setfontname ( obj , name )</i> Setzt den Fonttyp des Textfield <b>obj</b> auf <b>name</b> .
<b>j_setfont</b>	<i>procedure j_setfont ( obj , name , style , size )</i> Setzt den Font des Textfield <b>obj</b> entsprechend der Parameterliste.
<b>j_setfontsize</b>	<i>procedure j_setfontsize ( obj , size )</i> Setzt den Fontgröße des Textfield <b>obj</b> auf <b>size</b> .

<b>j_setfontstyle</b>	<i>procedure j_setfontstyle ( obj , style )</i> Setzt den Fontstyle des Textfield <b>obj</b> auf <b>style</b> .
<b>j_setnamedcolorbg</b>	<i>procedure j_setnamedcolorbg ( obj , color )</i> Setzt die Hintergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnamedcolor</b>	<i>procedure j_setnamedcolor ( obj , color )</i> Setzt die Vordergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setpos</b>	<i>procedure j_setpos ( obj , xpos , ypos )</i> Verschiebt das Textfield <b>obj</b> zur angegebenen Koordinate ( <b>xpos,ypos</b> ) an.
<b>j_setsize</b>	<i>procedure j_setsize ( obj , width , height )</i> Setzt Breite und Höhe des Textfield <b>obj</b> auf <b>width</b> and <b>height</b> .
<b>j_settext</b>	<i>procedure j_settext ( obj , str\$ )</i> Ändert den Textinhalt des Textfield <b>obj</b> auf <b>str</b>
<b>j_show</b>	<i>procedure j_show ( obj )</i> Zeigt das Textfield <b>obj</b> auf dem Bildschirm an.

Vscrollbar
------------

- j\_vscrollbar**      *function j\_vscrollbar ( obj )*  
Erzeugt eine vertikale Scrollbar und gibt dessen Eventnummer zurück.
- j\_add**              *procedure j\_add ( obj , cont )*  
Fügt die Vscrollbar **obj** in den Container **cont** ein.
- j\_componentlistener** *function j\_componentlistener ( obj , kind )*  
Bindet an die Vscrollbar **obj** einen neuen Componentlistener, und liefert dessen Eventnummer zurück. Der Parameter **kind** bestimmt, wann ein Event ausgelöst wird.
- j\_disable**          *procedure j\_disable ( obj )*  
disabled die Vscrollbar **obj**, sodaß diese auf keine Benutzereingaben mehr reagiert.
- j\_dispose**          *procedure j\_dispose ( obj )*  
die Vscrollbar **obj** wird mit allen Recourcen gelöscht.
- j\_enable**            *procedure j\_enable ( obj )*  
enabled die Vscrollbar **obj**.
- j\_focuslistener**    *function j\_focuslistener ( obj )*  
Bindet an die Vscrollbar **obj** einen neuen Focuslistener, und liefert dessen Eventnummer zurück.
- j\_getfontascent**    *function j\_getfontascent ( obj )*  
Liefert die Oberlänge (Ascent) des aktuellen Fonts der Vscrollbar **obj** in Bildschirmpixeln zurück.
- j\_getfontheight**    *function j\_getfontheight ( obj )*  
Liefert die totale Höhe des aktuellen Fonts der Vscrollbar **obj** in Bildschirmpixeln zurück.
- j\_getheight**        *function j\_getheight ( obj )*  
Liefert die Höhe der Vscrollbar **obj** zurück.
- j\_getparentid**      *function j\_getparentid ( obj )*  
Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird  $-1$  zurückgegeben.
- j\_getparent**        *function j\_getparent ( obj )*  
Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird  $-1$  zurückgegeben.
- j\_getstringwidth**   *function j\_getstringwidth ( obj , str\$ )*  
Liefert die Länge des **str** des aktuellen Fonts der Vscrollbar **obj** in Bildschirmpixeln zurück.

<b>j_getvalue</b>	<i>function j_getvalue ( obj )</i> Liefert den aktuellen Wert der Vscrollbar .
<b>j_getwidth</b>	<i>function j_getwidth ( obj )</i> Liefert die Breite der Vscrollbar <b>obj</b> zurück.
<b>j_getxpos</b>	<i>function j_getxpos ( obj )</i> Liefert die aktuelle horizontale Position der Vscrollbar <b>obj</b> zurück.
<b>j_getypos</b>	<i>function j_getypos ( obj )</i> Liefert die aktuelle vertikale Position der Vscrollbar <b>obj</b> zurück.
<b>j_hide</b>	<i>procedure j_hide ( obj )</i> Zeigt die Vscrollbar <b>obj</b> nicht mehr auf dem Bildschirm an.
<b>j_isparent</b>	<i>function j_isparent ( obj , cont )</i> Liefert J_TRUE zurück, falls das <b>cont</b> die umgebende Komponente von <b>obj</b> ist, sonst J_FALSE
<b>j_isvisible</b>	<i>function j_isvisible ( obj )</i> Liefert J_TRUE zurück, falls <b>obj</b> sichtbar ist, sonst J_FALSE
<b>j_keylistener</b>	<i>function j_keylistener ( obj )</i> Bindet an die Vscrollbar <b>obj</b> einen neuen Keylistener, und liefert dessen Eventnummer zurück.
<b>j_mouselistener</b>	<i>function j_mouselistener ( obj , kind )</i> Bindet an die Vscrollbar <b>obj</b> einen neuen Mouselistener, und liefert dessen Eventnummer zurück. Der Parameter <b>kind</b> bestimmt, wann ein Event ausgelöst wird.
<b>j_popupmenu</b>	<i>function j_popupmenu ( obj , label\$ )</i> Erzeugt ein neues Popupmenu mit dem Inhalt <b>label</b> und liefert dessen Eventnummer zurück.
<b>j_print</b>	<i>procedure j_print ( obj )</i> Druckt den Inhalt der Vscrollbar .
<b>j_release</b>	<i>procedure j_release ( obj )</i> Gibt die Vscrollbar <b>obj</b> aus dem umgebenden Container Element frei.
<b>j_setblockinc</b>	<i>function j_setblockinc ( obj , val )</i> Stellt die Blockänderung von der Vscrollbar auf den Wert von <b>val</b> ein.
<b>j_setborderpos</b>	<i>procedure j_setborderpos ( obj , pos )</i> Positioniert die Vscrollbar <b>obj</b> an die angegebene Position <b>pos</b> . Im umgebenden Container muß einen Borderlayout Manager eingestellt sein.
<b>j_setcolorbg</b>	<i>procedure j_setcolorbg ( obj , r , g , b )</i> Setzt die Hintergrundfarbe auf den übergebenen RGB Wert
<b>j_setcolor</b>	<i>procedure j_setcolor ( obj , r , g , b )</i>

	Setzt die Vordergrundfarbe auf den übergebenen RGB Wert
<b>j_setcursor</b>	<i>function j_setcursor ( obj , cursor )</i> Setzt den Cursor der Vscrollbar <b>obj</b> auf <b>cursor</b> .
<b>j_setfocus</b>	<i>function j_setfocus ( obj )</i> Weist den Input Focus der Vscrollbar <b>obj</b> zu.
<b>j_setfontname</b>	<i>procedure j_setfontname ( obj , name )</i> Setzt den Fonttyp der Vscrollbar <b>obj</b> auf <b>name</b> .
<b>j_setfont</b>	<i>procedure j_setfont ( obj , name , style , size )</i> Setzt den Font der Vscrollbar <b>obj</b> entsprechend der Parameterliste.
<b>j_setfontsize</b>	<i>procedure j_setfontsize ( obj , size )</i> Setzt den Fontgröße der Vscrollbar <b>obj</b> auf <b>size</b> .
<b>j_setfontstyle</b>	<i>procedure j_setfontstyle ( obj , style )</i> Setzt den Fontstyle der Vscrollbar <b>obj</b> auf <b>style</b> .
<b>j_setmax</b>	<i>function j_setmax ( obj , val )</i> Stellt die obere Grenze der Vscrollbar auf den Wert von <b>val</b> ein.
<b>j_setmin</b>	<i>function j_setmin ( obj , val )</i> Stellt die untere Grenze der Vscrollbar auf den Wert von <b>val</b> ein.
<b>j_setnamedcolorbg</b>	<i>procedure j_setnamedcolorbg ( obj , color )</i> Setzt die Hintergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnamedcolor</b>	<i>procedure j_setnamedcolor ( obj , color )</i> Setzt die Vordergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setpos</b>	<i>procedure j_setpos ( obj , xpos , ypos )</i> Verschiebt die Vscrollbar <b>obj</b> zur angegebenen Koordinate ( <b>xpos,ypos</b> ) an.
<b>j_setsize</b>	<i>procedure j_setsize ( obj , width , height )</i> Setzt Breite und Höhe der Vscrollbar <b>obj</b> auf <b>width</b> and <b>height</b> .
<b>j_setslidesize</b>	<i>function j_setslidesize ( obj , val )</i> Setzt die Größe des Schiebers auf den Wert von <b>val</b> .
<b>j_setunitinc</b>	<i>function j_setunitinc ( obj , val )</i> Stellt die minimale Wertänderung von die Vscrollbar auf den Wert von <b>val</b> ein.
<b>j_setvalue</b>	<i>procedure j_setvalue ( obj , val )</i> Stellt den aktuellen Wert der Vscrollbar <b>obj</b> auf <b>val</b> .
<b>j_show</b>	<i>procedure j_show ( obj )</i> Zeigt die Vscrollbar <b>obj</b> auf dem Bildschirm an.

Window
--------

<b>j_window</b>	<p><i>function j_window ( obj )</i> Erzeugt ein neues Window und liefert dessen Eventnummer zurück.</p>
<b>j_add</b>	<p><i>procedure j_add ( obj , cont )</i> Fügt das Window <b>obj</b> in den Container <b>cont</b> ein.</p>
<b>j_borderpanel</b>	<p><i>function j_borderpanel ( obj , type )</i> Erzeugt ein neues Borderpanel und liefert dessen Eventnummer zurück.</p>
<b>j_button</b>	<p><i>function j_button ( obj , label\$ )</i> Erzeugt einen neuen Button mit dem Textinhalt <b>label</b></p>
<b>j_canvas</b>	<p><i>function j_canvas ( obj , width , height )</i> Erzeugt einen neuen Canvas mit der Breite <b>width</b> Pixel und einer Höhe von <b>height</b> Pixel.</p>
<b>j_checkbox</b>	<p><i>function j_checkbox ( obj , label\$ )</i> Erzeugt eine Checkbox mit dem Textinhalt <b>label</b> und liefert dessen Eventnummer zurück.</p>
<b>j_choice</b>	<p><i>function j_choice ( obj )</i> Erzeugt eine neue Choice und gibt deren Eventnummer zurück.</p>
<b>j_componentlistener</b>	<p><i>function j_componentlistener ( obj , kind )</i> Bindet an das Window <b>obj</b> einen neuen Componentlistener, und liefert dessen Eventnummer zurück. Der Parameter <b>kind</b> bestimmt, wann ein Event ausgelöst wird.</p>
<b>j_disable</b>	<p><i>procedure j_disable ( obj )</i> disabled das Window <b>obj</b>, sodaß diese auf keine Benutzereingaben mehr reagiert.</p>
<b>j_dispose</b>	<p><i>procedure j_dispose ( obj )</i> das Window <b>obj</b> wird mit allen Recourcen gelöscht.</p>
<b>j_enable</b>	<p><i>procedure j_enable ( obj )</i> enabled das Window <b>obj</b>.</p>
<b>j_focuslistener</b>	<p><i>function j_focuslistener ( obj )</i> Bindet an das Window <b>obj</b> einen neuen Focuslistener, und liefert dessen Eventnummer zurück.</p>
<b>j_getfontascent</b>	<p><i>function j_getfontascent ( obj )</i> Liefert die Oberlänge (Ascent) des aktuellen Fonts des Window <b>obj</b> in Bildschirmpixeln zurück.</p>
<b>j_getfontheight</b>	<p><i>function j_getfontheight ( obj )</i></p>

	Liefert die totale Höhe des aktuellen Fonts des Window <b>obj</b> in Bildschirmpixeln zurück.
<b>j_getheight</b>	<i>function j_getheight ( obj )</i> Liefert die Höhe des Window <b>obj</b> zurück.
<b>j_getinheight</b>	<i>function j_getinheight ( cont )</i> Liefert die Höhe des Window <b>obj</b> des inneren Bereichs zurück.
<b>j_getinsets</b>	<i>function j_getinsets ( obj , side )</i> Liefert die angeforderte Breite des inneren Randes dem Window zurück.
<b>j_getinwidth</b>	<i>function j_getinwidth ( cont )</i> Liefert die Breite des Window <b>obj</b> des inneren Bereichs zurück.
<b>j_getlayoutid</b>	<i>function j_getlayoutid ( obj )</i> Liefert die Eventnummer des Layoutmanagers des Containers <b>obj</b> zurück.
<b>j_getparentid</b>	<i>function j_getparentid ( obj )</i> Liefert die Eventnummer des umgebenen Containers von <b>obj</b> zurück. Ist <b>obj</b> ein Frame wird $-1$ zurückgegeben.
<b>j_getparent</b>	<i>function j_getparent ( obj )</i> Liefert die Eventnummer des umgebenen Containers von <b>obj</b> zurück. Ist <b>obj</b> ein Frame wird $-1$ zurückgegeben.
<b>j_getstringwidth</b>	<i>function j_getstringwidth ( obj , str\$ )</i> Liefert die Länge des <b>str</b> des aktuellen Fonts des Window <b>obj</b> in Bildschirmpixeln zurück.
<b>j_getwidth</b>	<i>function j_getwidth ( obj )</i> Liefert die Breite des Window <b>obj</b> zurück.
<b>j_getxpos</b>	<i>function j_getxpos ( obj )</i> Liefert die aktuelle horizontale Position des Window <b>obj</b> zurück.
<b>j_getypos</b>	<i>function j_getypos ( obj )</i> Liefert die aktuelle vertikale Position des Window <b>obj</b> zurück.
<b>j_graphicbutton</b>	<i>function j_graphicbutton ( obj , filename\$ )</i> Erzeugt einen neuen Graphicbutton mit dem Bildinhalt des Files, das durch <b>filename</b> spezifiziert wird.
<b>j_graphiclabel</b>	<i>function j_graphiclabel ( obj , str\$ )</i> Erzeugt ein neues Graphiclabel mit dem Bildinhalt des Files, das durch <b>filename</b> spezifiziert wird.
<b>j_hide</b>	<i>procedure j_hide ( obj )</i> Zeigt das Window <b>obj</b> nicht mehr auf dem Bildschirm an.
<b>j_hscrollbar</b>	<i>function j_hscrollbar ( obj )</i> Erzeugt eine horizontale Scrollbar und gibt dessen Eventnummer zurück.

<b>j_isparent</b>	<i>function j_isparent ( obj , cont )</i> Liefert J_TRUE zurück, falls das <b>cont</b> die umgebende Komponente von <b>obj</b> ist, sonst J_FALSE
<b>j_isvisible</b>	<i>function j_isvisible ( obj )</i> Liefert J_TRUE zurück, falls <b>obj</b> sichtbar ist, sonst J_FALSE
<b>j_keylistener</b>	<i>function j_keylistener ( obj )</i> Bindet an das Window <b>obj</b> einen neuen Keylistener, und liefert dessen Eventnummer zurück.
<b>j_label</b>	<i>function j_label ( obj , label\$ )</i> Erzeugt ein neues Label Element mit dem Textinhalt <b>label</b>
<b>j_led</b>	<i>function j_led ( obj , style , color )</i> Erzeugt eine LED der Farbe <b>color</b> und der Form <b>style</b>
<b>j_line</b>	<i>function j_line ( obj , orient , style , length )</i> Erzeugt ein neues Linienelement und liefert dessen Eventnummer zurück
<b>j_list</b>	<i>function j_list ( obj , rows )</i> Erzeugt eine neue List und gibt deren Eventnummer zurück.
<b>j_meter</b>	<i>function j_meter ( obj , title\$ )</i> Erzeugt ein analoges Anzeigeelement mit der Beschriftung <b>title</b> .
<b>j_mouselistener</b>	<i>function j_mouselistener ( obj , kind )</i> Bindet an das Window <b>obj</b> einen neuen Mouselistener, und liefert dessen Eventnummer zurück. Der Parameter <b>kind</b> bestimmt, wann ein Event ausgelöst wird.
<b>j_pack</b>	<i>procedure j_pack ( obj )</i> Veranlaßt den Layoutmanager von <b>obj</b> die minimale Größe zu berechnen.
<b>j_panel</b>	<i>function j_panel ( obj )</i> Erzeugt ein neues panel und liefert dessen Eventnummer zurück.
<b>j_popupmenu</b>	<i>function j_popupmenu ( obj , label\$ )</i> Erzeugt ein neues Popuptaste mit dem Inhalt <b>label</b> und liefert dessen Eventnummer zurück.
<b>j_print</b>	<i>procedure j_print ( obj )</i> Druckt den Inhalt des Window .
<b>j_progressbar</b>	<i>function j_progressbar ( obj , orient )</i> Erzeugt eine Progress Bar mit der Ausrichtung <b>orient</b> .
<b>j_radiogroup</b>	<i>function j_radiogroup ( obj )</i> Erzeugt eine neue Radiogruppe und liefert dessen Eventnummer zurück.
<b>j_releaseall</b>	<i>procedure j_releaseall ( obj )</i> Gibt alle Elemente aus dem Window <b>obj</b> frei.



<b>j_release</b>	<i>procedure j_release ( obj )</i> Gibt das Window <b>obj</b> aus dem umgebenden Container Element frei.
<b>j_scrollpane</b>	<i>function j_scrollpane ( obj )</i> Erzeugt eine neue Scrollpane und gibt dessen Eventnummer zurück.
<b>j_setalign</b>	<i>procedure j_setalign ( obj , align )</i> Setzt die Ausrichtung innerhalb des Window <b>obj</b> auf <b>align</b> . Ein Flowlayout Manager wird vorausgesetzt.
<b>j_setborderlayout</b>	<i>procedure j_setborderlayout ( obj )</i> Integriert einen BorderLayout Manager in das Window <b>obj</b> .
<b>j_setborderpos</b>	<i>procedure j_setborderpos ( obj , pos )</i> Positioniert das Window <b>obj</b> an die angegebene Position <b>pos</b> . Im umgebenden Container muß einen BorderLayout Manager eingestellt sein.
<b>j_setcolorbg</b>	<i>procedure j_setcolorbg ( obj , r , g , b )</i> Setzt die Hintergrundfarbe auf den übergebenen RGB Wert
<b>j_setcolor</b>	<i>procedure j_setcolor ( obj , r , g , b )</i> Setzt die Vordergrundfarbe auf den übergebenen RGB Wert
<b>j_setcursor</b>	<i>function j_setcursor ( obj , cursor )</i> Setzt den Cursor des Window <b>obj</b> auf <b>cursor</b> .
<b>j_setfixlayout</b>	<i>procedure j_setfixlayout ( obj )</i> Integriert einen Fixlayout Manager in das Window <b>obj</b> (voreingestellte Layoutmanager).
<b>j_setflowfill</b>	<i>procedure j_setflowfill ( obj , bool )</i> Setzt die Höhe (Breite) aller enthaltenen Komponenten auf die Höhe (Breite) von dem Window <b>obj</b> . Ein Flowlayout Manager wird vorausgesetzt.
<b>j_setflowlayout</b>	<i>procedure j_setflowlayout ( obj , align )</i> Integriert einen Flowlayout Manager in das Window <b>obj</b> mit der Ausrichtung <b>align</b> .
<b>j_setfocus</b>	<i>function j_setfocus ( obj )</i> Weist den Input Focus dem Window <b>obj</b> zu.
<b>j_setfontname</b>	<i>procedure j_setfontname ( obj , name )</i> Setzt den Fonttyp des Window <b>obj</b> auf <b>name</b> .
<b>j_setfont</b>	<i>procedure j_setfont ( obj , name , style , size )</i> Setzt den Font des Window <b>obj</b> entsprechend der Parameterliste.
<b>j_setfontsize</b>	<i>procedure j_setfontsize ( obj , size )</i> Setzt den Fontgröße des Window <b>obj</b> auf <b>size</b> .
<b>j_setfontstyle</b>	<i>procedure j_setfontstyle ( obj , style )</i> Setzt den Fontstyle des Window <b>obj</b> auf <b>style</b> .

<b>j_setgridlayout</b>	<i>procedure j_setgridlayout ( obj , row , col )</i> Integriert einen Gridlayout Manager in das Window <b>obj</b> mit <b>row</b> Reihen und <b>col</b> Spalten.
<b>j_sethgap</b>	<i>procedure j_sethgap ( obj , hgap )</i> Setzt den horizontalen Abstand innerhalb des Window <b>obj</b> auf <b>hgap</b> Pixel.
<b>j_setinsets</b>	<i>procedure j_setinsets ( obj , top , bottom , left , right )</i> Setzt den inneren Rahmen auf die angegebenen Größen.
<b>j_setnamedcolorbg</b>	<i>procedure j_setnamedcolorbg ( obj , color )</i> Setzt die Hintergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnamedcolor</b>	<i>procedure j_setnamedcolor ( obj , color )</i> Setzt die Vordergrundfarbe auf eine vordefinierte Farbe <b>color</b> .
<b>j_setnolayout</b>	<i>procedure j_setnolayout ( obj )</i> Entfernt den aktuellen Layout Manager aus dem Window <b>obj</b> .
<b>j_setpos</b>	<i>procedure j_setpos ( obj , xpos , ypos )</i> Verschiebt das Window <b>obj</b> zur angegebenen Koordinate ( <b>xpos,ypos</b> ) an.
<b>j_setsize</b>	<i>procedure j_setsize ( obj , width , height )</i> Setzt Breite und Höhe des Window <b>obj</b> auf <b>width</b> and <b>height</b> .
<b>j_setvgap</b>	<i>procedure j_setvgap ( obj , vgap )</i> Setzt den vertikalen Abstand innerhalb des Window <b>obj</b> auf <b>vgap</b> Pixel.
<b>j_sevensegment</b>	<i>function j_sevensegment ( obj , color )</i> Erzeugt eine Siebensegment Anzeige der Farbe <b>color</b> .
<b>j_show</b>	<i>procedure j_show ( obj )</i> Zeigt das Window <b>obj</b> auf dem Bildschirm an.
<b>j_textarea</b>	<i>function j_textarea ( obj , rows , columns )</i> Erzeugt eine neue Textarea und gibt dessen Eventnummer zurück.
<b>j_textfield</b>	<i>function j_textfield ( obj , columns )</i> Erzeugt eine neues Textfield und gibt dessen Eventnummer zurück.
<b>j_vscrollbar</b>	<i>function j_vscrollbar ( obj )</i> Erzeugt eine vertikale Scrollbar und gibt dessen Eventnummer zurück.
<b>j_windowlistener</b>	<i>function j_windowlistener ( window , kind )</i> Bindet an das Window <b>window</b> einen neuen Windowlistener, und liefert dessen Eventnummer zurück. Der Parameter <b>kind</b> bestimmt, wann ein Event ausgelöst wird.

## Kapitel 2

# Functions

additem

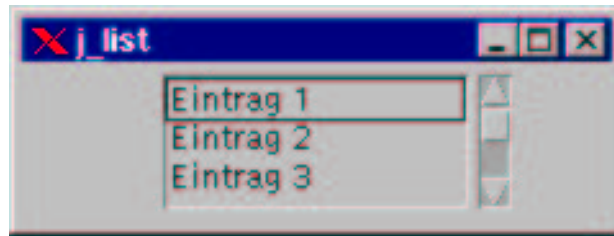
Synopsis      subroutine **j\_additem** ( obj , str\$ )

Beschreibung      Diese Prozedur hängt ein neues Item mit dem Inhalt **str** an die Auswahlliste an. Es wird keine Eventnummer vergeben, statt dessen werden alle Item intern durchnummeriert, beginnend mit 0. Die Zielkomponente erzeugt bei einer Auswahl einen Event. Welches Item ausgewählt wurde, kann mit der Funktion ***j\_getselect()*** ermittelt werden.

Zielkomponenten      List, Choice

Beispiel

```
:  
list = j_list(frame,3)  
j_additem(list,"Eintrag 1")  
j_additem(list,"Eintrag 2")  
:
```



add
-----

Synopsis	subroutine <b>j_add</b> ( obj , cont )
Beschreibung	Fügt die Komponente <b>obj</b> in den Container <b>cont</b> ein.
Zielkomponenten	Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphicalabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevensegment

## alertbox

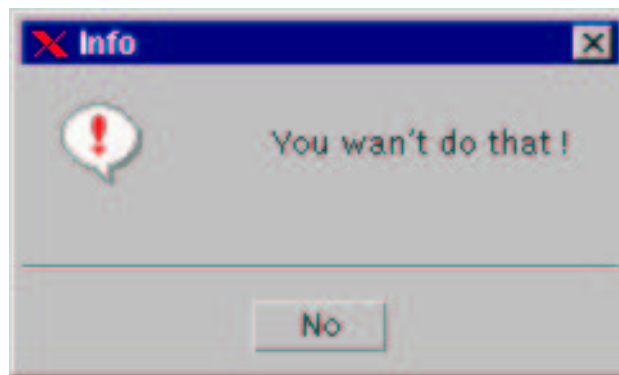
Synopsis            subroutine **j\_alertbox** ( obj , title\$ , text\$ , button\$ )

Beschreibung     Öffnet eine Alertbox mit der Überschrift **title**, Textinhalt **text** und einem Button. Eine Alertbox ist modal, dh. die Applikation wird solange blockiert, bis der Button oder das Closeicon angeklickt wird. Der Rückgabewert der Funktion beträgt 0 falls das Closeicon geklickt wird, wird der Button geklickt, so wird 1 zurückgegeben.

Zielkomponenten   Frame

Beispiel

```
:  
retval = j_alertbox(frame,"Info","You wan't do that !", " No  ")  
:
```



appendtext
------------

Synopsis            subroutine **j\_appendtext** ( obj , text\$ )

Beschreibung      Hängt den Text **text** am Ende von **obj** an.

Zielkomponenten   Textarea

beep
------

Synopsis      subroutine **j\_beep** ( )

Beschreibung      Es ertönt ein Systembeep.



## borderpanel

Synopsis            `function j_borderpanel ( obj , type )`

Beschreibung      Diese Funktion erzeugt ein neues Borderpanel und liefert dessen Eventnummer zurück. Im Fehlerfall wird `-1` zurückgegeben. Das Panel wird der Komponente zugeordnet, die durch **obj** referenziert wird. Das Borderpanel ist funktionell identisch mit einem normalen Panel, es enthält lediglich einen sichtbaren Rahmen, der die enthaltenen Komponenten auch optisch gruppiert. Der Stil des Rahmens kann aus vier Stiltypen gewählt werden:

- `J_LINEDOWN` : Der Rahmen besteht aus einer Linie, die optisch abgesenkt erscheint.
- `J_LINEUP` : Der Rahmen besteht aus einer Linie, die optisch hervorzutreten scheint.
- `J_AREADOWN` : Das Panel scheint optisch abgesenkt.
- `J_AREAUP` : Das Panel scheint optisch hervorzutreten.

Ein Borderpanel liefert bei einer Größenänderung einen Event.

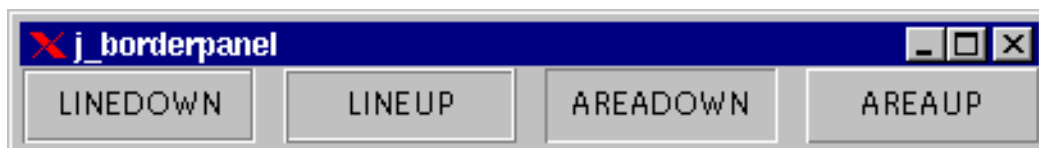
Zielkomponenten   Panel, Borderpanel, Window, Dialog, Frame

Beispiel

```

:
j_setgridlayout(frame,1,4)
p1 = j_borderpanel(frame,J_LINEDOWN)
p2 = j_borderpanel(frame,J_LINEUP)
p3 = j_borderpanel(frame,J_AREADOWN)
p4 = j_borderpanel(frame,J_AREAUP)
:

```



## button

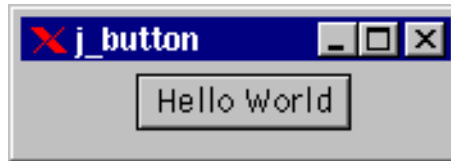
Synopsis      `function j_button ( obj , label$ )`

Beschreibung      Die Funktion erzeugt einen neuen Button mit dem Textinhalt **label**. Die Funktion liefert die Eventnummer des Buttons zurück, oder im Fehlerfall `-1`. Ein Event wird erzeugt, wenn der Benutzer mit der Maus den Button betätigt.

Zielkomponenten      Panel, Borderpanel, Window, Dialog, Frame

Beispiel

```
:  
frame = j_frame("j_button")  
button = j_button(frame,"Hello World")  
:
```



## canvas

Synopsis      function **j\_canvas** ( obj , width , height )

Beschreibung      Diese Funktion erzeugt einen neuen Canvas mit der Breite **width** Pixel und einer Höhe von **height** Pixel. Ein Canvas (Leinwand) stellt eine universelle Zeichenfläche zur Verfügung. Die Funktion liefert die Eventnummer des Canvas zurück, oder im Fehlerfall  $-1$ . Ein Event wird erzeugt, wenn sich die Größe des Canvas verändert.

Zielkomponenten      Panel, Borderpanel, Window, Dialog, Frame

Beispiel

```
:  
canvas = j_canvas(frame,200,50)  
j_setnamedcolorbg(canvas,J_RED)  
:
```



## checkbox

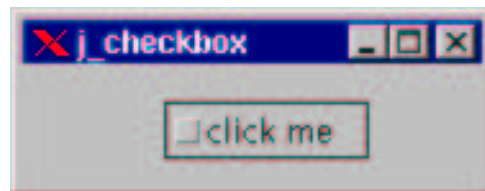
Synopsis      `function j_checkbox ( obj , label$ )`

Beschreibung      Diese Funktion erzeugt eine Checkbox mit dem Textinhalt **label** und liefert dessen Eventnummer zurück. Im Fehlerfall wird `-1` zurückgegeben. Eine Checkbox ist ein graphisches Element, das zwei Zustände annehmen kann. Sie ist entweder gesetzt oder nicht gesetzt. Checkboxes bestehen intern aus zwei Elementen, einem Label, und einem kleinen graphischen Element, das den Zustand der Checkbox anzeigt. Eine Checkbox liefert immer dann einen Event, wenn sich ihr Zustand ändert.

Zielkomponenten      Panel, Borderpanel, Window, Dialog, Frame

Beispiel

```
:  
frame = j_frame("j_checkbox")  
checkbox = j_checkbox(frame,"click me")  
:
```



## checkmenuItemem

Synopsis            `function j_checkmenuItemem ( obj , label$ )`

Beschreibung     Die Funktion erzeugt ein CheckmenuItemem mit der Beschriftung **label** in der Komponente, die durch **obj** referenziert wird, und liefert dessen Eventnummer zurück. Eine CheckmenuItemem erzeugt einen Event, wenn es durch einen Mausklick angewählt wurde. Ein CheckmenuItemem kann zwei Zustände annehmen (selected / not selected), die durch ein kleine Icons dargestellt werden. Der Zustand eines Checkmenüitems kann mit der Funktion **j\_getstate()** abgefragt werden. Als Zielkomponenten sind nur Menu, Popumenu und Helpmenu zulässig.

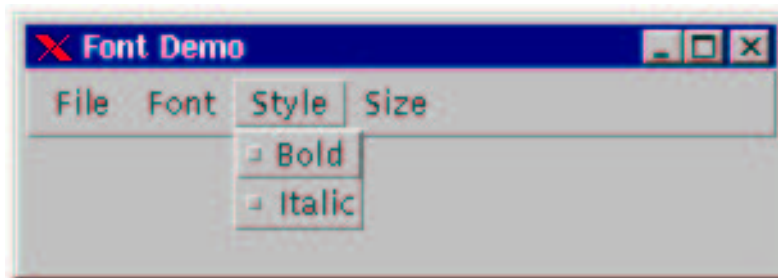
Zielkomponenten    Menu, Popumenu, Helpmenu

Beispiel

```

:
menubar = j_menubar(frame)
:
style = j_menu(menubar,"Style")
bold = j_checkmenuItemem(style,"Bold")
italic= j_checkmenuItemem(style,"Italic")
:

```



choicebox2

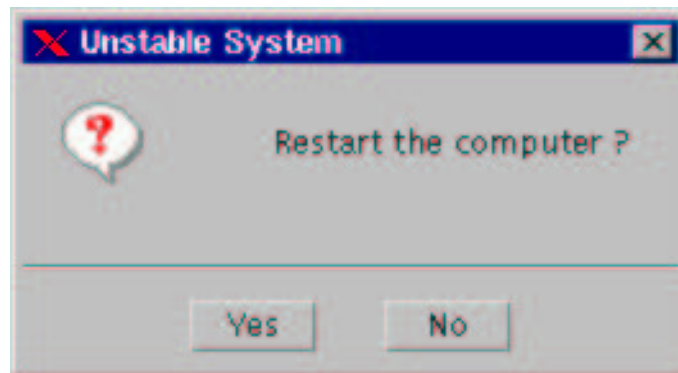
Synopsis            subroutine **j\_choicebox2** ( obj , title\$ , text\$ , button1\$ ,  
                          button2\$ )

Beschreibung      Öffnet eine Choicebox mit der Überschrift **title**, Textinhalt **text**  
                          und zwei Buttons. Eine Choicebox ist modal, dh. die Applikati-  
                          on wird solange blockiert, bis ein Button oder das Closeicon an-  
                          geklickt wird. Der Focus ist auf den ersten Button gesetzt. Der  
                          Rückgabewert der Funktion beträgt 0 falls das Closeicon geklickt  
                          wird, 1 wird Button1 geklickt und 2 bei Button2.

Zielkomponenten    Frame

Beispiel

```
:  
retval = j_choicebox2(frame,"Unstable System","Restart the computer ?",  
                          " Yes ", "No")  
:
```



## choicebox3

Synopsis            subroutine **j\_choicebox3** ( obj , title\$ , text\$ , button1\$ ,  
                         button2\$ , button3\$ )

Beschreibung      Öffnet eine Choicebox mit der Überschrift **title**, Textinhalt **text**  
                         und drei Buttons. Eine Choicebox ist modal, dh. die Applikati-  
                         on wird solange blockiert, bis ein Button oder das Closeicon an-  
                         geklickt wird. Der Focus ist auf den ersten Button gesetzt. Der  
                         Rückgabewert der Funktion beträgt 0 falls das Closeicon geklickt  
                         wird, 1 bei Button1, 2 bei Button2 und 3 bei Button3.

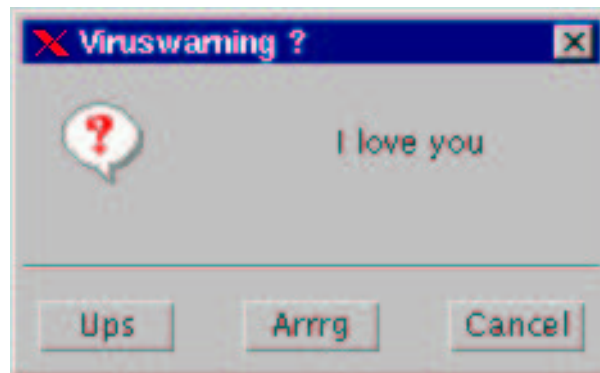
Zielkomponenten   Frame

Beispiel

```

:
retval = j_choicebox2(frame,"Viruswarning ?","I love you",
                    "Ups","Arrrg","Cancel")
:

```



choice

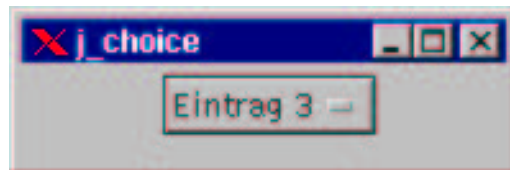
Synopsis           function **j\_choice** ( obj )

Beschreibung     Diese Funktion erzeugt eine neue Choice und gibt deren Eventnummer zurück. Im Fehlerfall wird  $-1$  zurückgegeben. Eine Choice bietet dem Benutzer die Auswahl zwischen mehreren vordefinierten Einträgen. Dabei befindet sich auf der Oberfläche zunächst ein Element das einem Button ähnlich sieht. Wird es angeklickt, so klappt ein Menu heraus, aus dem man ein Element auswählen kann. Wird ein Element ausgewählt, so erzeugt die Choice einen Event.

Zielkomponenten   Panel, Borderpanel, Window, Dialog, Frame

Beispiel

```
:  
choice = j_choice(frame)  
call j_additem(choice,"Eintrag 1")  
call j_additem(choice,"Eintrag 2")  
:
```





cliprect
----------

Synopsis	subroutine <b>j_cliprect</b> ( obj , x , y , width , height )
Beschreibung	Setzt den Clipping Bereich auf das spezifizierte Rechteck ( <b>x</b> , <b>y</b> , <b>width</b> , <b>height</b> ).
Zielkomponenten	Canvas, Image, Printer

componentlistener

Synopsis	function <b>j_componentlistener</b> ( obj , kind )
Beschreibung	<p>Bindet an die Komponente <b>obj</b> einen neuen Componentlistener, und liefert dessen Eventnummer zurück. Der Parameter <b>kind</b> bestimmt, wann ein Event ausgelöst wird. Zulässige Werte für <b>kind</b> sind:</p> <ul style="list-style-type: none"><li>• J.RESIZED : Der Componentlistener liefert einen Event, wenn die Komponente <b>obj</b> die Größe ändert.</li><li>• J.HIDDEN : Der Componentlistener liefert einen Event, wenn die Komponente <b>obj</b> verdeckt wird.</li><li>• J.SHOWN : Der Componentlistener liefert einen Event, wenn die Komponente <b>obj</b> sichtbar wird.</li></ul>
Zielkomponenten	Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphiclabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevenssegment

connect
---------

Synopsis            `function j_connect ( hostname$ )`

Beschreibung     Diese Funktion stellt eine Verbindung zu einem laufenden JAPI Kernel her. Der Kernel wird auf dem Rechner **hostname** gesucht. **hostname** kann einen Namen oder eine IP Adresse beinhalten. Auf diese Weise kann eine Ausgabeumlenkung erfolgen, dh. Applikation und Oberfläche laufen auf verschiedenen Hosts. Schlägt die Verbindung fehl, so wird als Fehlercode J\_FALSE zurückgegeben

Beispiel

```
:  
if( j_connect("atan.japi.de") = J_FALSE ) then  
  
    or  
  
if( j_connect("127.0.0.1") = J_FALSE ) then  
:
```

delete

Synopsis            subroutine **j\_delete** ( obj , start , end )

Beschreibung      löscht den Text von Position **start** bis Position **end**.

Zielkomponenten   Textarea

deselect

Synopsis        function **j\_deselect** ( obj , item )

Beschreibung        Deselectiert das Item mit dem Index **item**.

Zielkomponenten    List

dialog

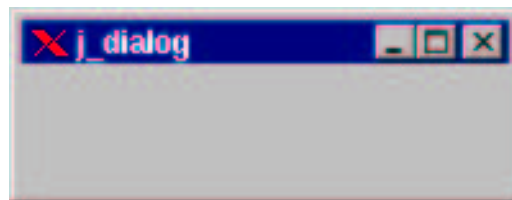
Synopsis            function **j\_dialog** ( obj , label\$ )

Beschreibung       Dies Funktion erzeugt ein neues Dialogwindow mit dem Titel **label** und liefert dessen Eventnummer zurück. Im Fehlerfall wird  $-1$  zurückgeliefert. Ein Dialogfenster ist ein eigenständiges Fenster, ebenso wie ein Frame. Allerdings kann ein Dialogfenster keine Menüleiste besitzen. Als Zielobjekt dieser Funktion ist nur ein Frame zulässig. Ein Dialog wird erst angezeigt, wenn ein entsprechendes *j\_show()* Kommando ausgeführt wurde. Ein Dialog erzeugt einen Event, wenn das Close Icon im Fensterrahmen angeklickt wird.

Zielkomponenten    Frame

Beispiel

```
:  
dialog = j_dialog(frame,"j_dialog")  
j_setsize(dialog,200,80)  
j_show(dialog)  
:
```



disable
---------

Synopsis	subroutine <b>j_disable</b> ( obj )
Beschreibung	disabled die Komponente <b>obj</b> , sodaß diese auf keine Benutzereingaben mehr reagiert.
Zielkomponenten	Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphiclabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevenssegment, MenuItem, CheckBoxMenuItem, Menu, HelpMenu, Popumenu

dispose
---------

Synopsis	subroutine <b>j_dispose</b> ( obj )
Beschreibung	die Komponente <b>obj</b> wird mit allen Recourcen gelöscht.
Zielkomponenten	Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphiclabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevensegment, Canvas, Image, Printer, Keylistener, Focuslistener, Mouselistener

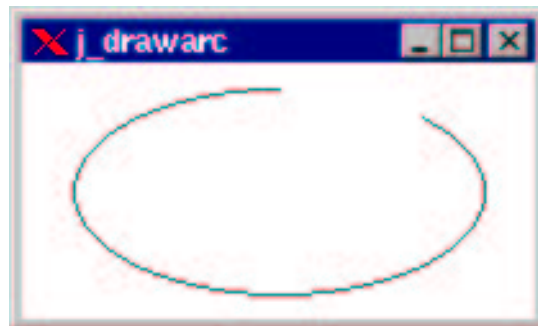


## drawarc

- Synopsis      subroutine **j\_drawarc** ( obj , x , y , rx , ry , arc1 , arc2 )
- Beschreibung      Zeichnet einen Kreisbogen mit dem Mittelpunkt **(x,y)** und den Hauptradien **rx** und **ry**. Die Parameter **arc1** und **arc2** bestimmen den Start- bzw. den Endwinkel des Kreisbogens. Bezugswert ist die Horizontale, von der die Winkel entgegen des Uhrzeigersinns abgetragen werden. Die Winkel sind im Bogenmaß anzugeben. Nullpunkt des Koordinatensystems ist die linke obere Ecke der Komponenten, die durch **obj** referenziert wird.
- Zielkomponenten      Canvas, Image, Printer

### Beispiel

```
:  
canvas = j_canvas(frame,200,100)  
j_drawarc(canvas,100,50,80,40,45,-270)  
:
```



## drawcircle

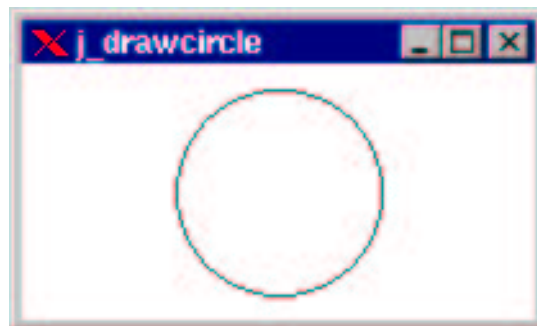
Synopsis            subroutine **j\_drawcircle** ( obj , x , y , r )

Beschreibung     Zeichnet einen Kreis mit dem Mittelpunkt (**x,y**) und dem Radius **r**. Nullpunkt des Koordinatensystems ist die linke obere Ecke der Komponenten, die durch **obj** referenziert wird.

Zielkomponenten   Canvas, Image, Printer

Beispiel

```
:  
canvas = j_canvas(frame,200,100)  
j_drawcircle(canvas,100,50,40)  
:
```



drawimagesource

Synopsis	subroutine <b>j_drawimagesource</b> ( obj , x , y , w , h , r , g , b )
Beschreibung	Die Funktion stellt ein Bild an der Position ( <b>x,y</b> ) mit der Breite <b>w</b> und der Höhe <b>h</b> dar. Die Bilddaten werden als rot-, grün- und blau- Kanal in den Arrays r,g,b übergeben.
Zielkomponenten	Canvas, Image, Printer

drawimage

Synopsis	subroutine <b>j_drawimage</b> ( obj , image , x , y )
Beschreibung	Kopiert das Image mit der Eventnummer <b>image</b> an die Position ( <b>x</b> , <b>y</b> ).
Zielkomponenten	Canvas, Image, Printer

## drawline

- Synopsis            subroutine **j\_drawline**    ( obj , x1 , y1 , x2 , y2 )
- Beschreibung        Zeichnet eine Linie in der momentan gesetzten Vordergrundfarbe von Position (**x1,y1**) nach Position (**x2,y2**). Nullpunkt des Koordinatensystems ist die linke obere Ecke der Komponenten, die durch **obj** referenziert wird.
- Zielkomponenten    Canvas, Image, Printer
- Beispiel

```
:  
canvas = j_canvas(frame,256,50)  
j_drawline(canvas,0,0,256,50)  
:
```



## drawoval

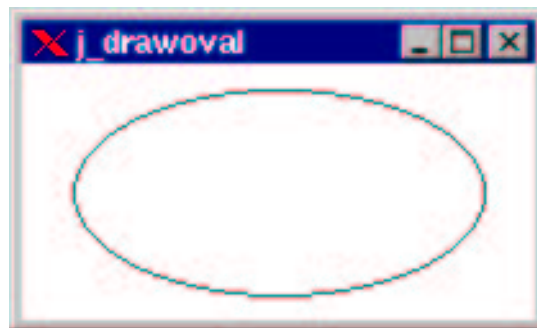
Synopsis            subroutine **j\_drawoval** ( obj , x , y , rx , ry )

Beschreibung     Zeichnet ein Oval mit dem Mittelpunkt **(x,y)** und den Hauptachsen **rx** und **ry** in der momentan gesetzten Vordergrundfarbe. Nullpunkt des Koordinatensystems ist die linke obere Ecke der Komponenten, die durch **obj** referenziert wird.

Zielkomponenten   Canvas, Image, Printer

Beispiel

```
:  
canvas = j_canvas(frame,200,100)  
j_drawoval(canvas,100,50,80,40)  
:
```



## drawpixel

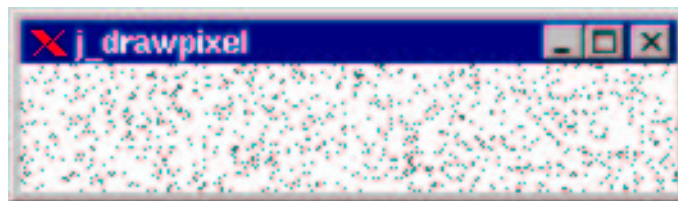
Synopsis            subroutine **j\_drawpixel** ( obj , x , y )

Beschreibung        Zeichnet einen Pixel in der momentan gesetzten Vordergrundfarbe an die Position (**x,y**). Bezugspunkt ist die linke obere Ecke der Komponenten, die durch **obj** referenziert wird.

Zielkomponenten    Canvas, Image, Printer

Beispiel

```
:  
canvas = j_canvas(frame,256,50)  
for i=0 to 1000  
    j_drawpixel(canvas, mod(j_random(),256), mod(j_random(),50))  
next i  
:
```



## drawpolygon

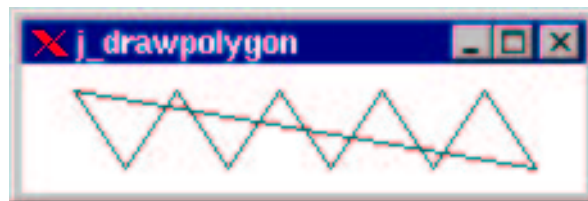
Synopsis            subroutine **j\_drawpolygon** ( obj , len , x , y )

Beschreibung        Zeichnet eine Polylinie in der momentan gesetzten Vordergrundfarbe durch die Punkte, die durch die Arrays **x** und **y** gegeben sind. Beide Arrays müssen die mindestens **len** Elemente besitzen. Das Polygon wird automatisch geschlossen. Daher müssen die erste und letzte Koordinate nicht übereinstimmen. Nullpunkt des Koordinatensystems ist die linke obere Ecke der Komponenten, die durch **obj** referenziert wird.

Zielkomponenten    Canvas, Image, Printer

Beispiel

```
:  
rem data x = 20,40,60,80,100,120,140,160,180,200  
rem data y = 10,40,10,40,10,40,10,40,10,40  
canvas = j_canvas(frame,256,50)  
j_drawpolygon(canvas,10,x,y)  
:
```





## drawpolyline

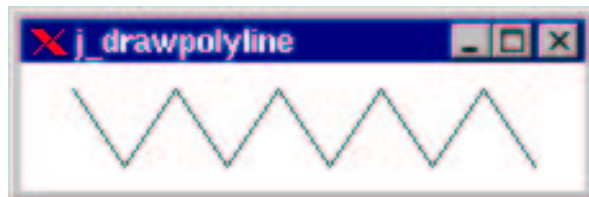
Synopsis            subroutine **j\_drawpolyline** ( obj , len , x , y )

Beschreibung        Zeichnet eine Polylinie in der momentan gesetzten Vordergrundfarbe durch die Punkte, die durch die Arrays **x** und **y** gegeben sind. Beide Arrays müssen die mindestens **len** Elemente besitzen. Nullpunkt des Koordinatensystems ist die linke obere Ecke der Komponenten, die durch **obj** referenziert wird.

Zielkomponenten    Canvas, Image, Printer

Beispiel

```
:  
rem data x = 20,40,60,80,100,120,140,160,180,200  
rem data y = 10,40,10,40,10,40,10,40,10,40  
canvas = j_canvas(frame,256,50)  
j_drawpolyline(canvas,10,x,y)  
:
```



## drawrect

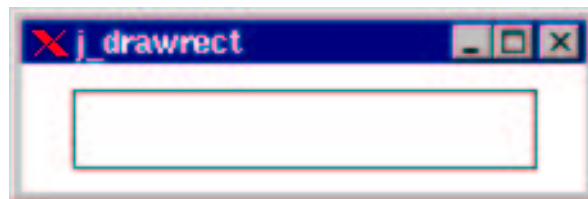
Synopsis            subroutine **j\_drawrect**    ( obj , x , y , width , height )

Beschreibung      Zeichnet eine Rechteck in der momentan gesetzten Vordergrundfarbe an die Position (**x,y**) mit der Breite **width** und der Höhe **height**. Bezugspunkt des Rechtecks ist die linke obere Ecke. Nullpunkt des Koordinatensystems ist die linke obere Ecke der Komponenten, die durch **obj** referenziert wird.

Zielkomponenten   Canvas, Image, Printer

Beispiel

```
:  
canvas = j_canvas(frame,220,50)  
j_drawrect(canvas,20,10,180,30)  
:
```



## drawroundrect

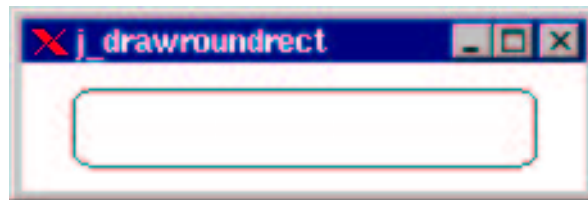
**Synopsis**            subroutine **j\_drawroundrect** ( obj , x , y , width , height ,  
                              arcx , arcy )

**Beschreibung**        Zeichnet ein Rechteck mit abgerundeten Ecken an die Position (x,y). Das Rechteck hat eine Breite von **width** und eine Höhe von **height** Pixeln. Die Form der runden Ecken wird durch die Parameter **arcx** und **arcy** bestimmt. Diese Parameter bezeichnen die Radien der Rundungen in horizontaler bzw. vertikaler Richtung.  
Bezugspunkt des Rechtecks ist die linke obere Ecke. Nullpunkt des Koordinatensystems ist die linke obere Ecke der Komponenten, die durch **obj** referenziert wird.

**Zielkomponenten**    Canvas, Image, Printer

**Beispiel**

```
:
canvas = j_canvas(frame,220,50)
j_drawroundrect(canvas,20,10,180,30,10,5)
:
```



drawscaledimage

Synopsis	subroutine <b>j_drawscaledimage</b> ( obj , image , sx , sy , sw , sh , tx , ty , tw , th )
Beschreibung	Kopiert einen Ausschnitt des Bildes <b>image</b> , der durch die Parameter ( <b>sx</b> , <b>sy</b> , <b>sw</b> , <b>sh</b> ) bestimmt wird an die Position ( <b>tx</b> , <b>ty</b> ) und scaliert den Ausschnitt auf die Breite <b>tw</b> und die Höhe <b>th</b>
Zielkomponenten	Canvas, Image, Printer

## drawstring

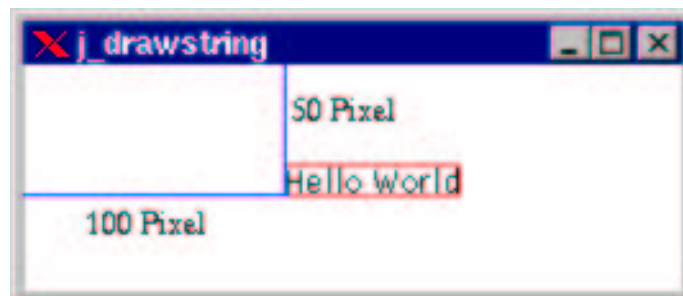
Synopsis            subroutine **j\_drawstring** ( obj , x , y , str\$ )

Beschreibung     Die Prozedur zeichnet einen String an die Position (**x,y**). Die durch **obj** referenzierte Komponente muss vom Typ Drawable sein (siehe Zielkomponenten). Die übergebenen Koordinaten beziehen sich auf die linke untere Ecke des Rechtecks, welches sich gedanklich um den gezeichneten String legen läßt.

Zielkomponenten   Canvas, Image, Printer

Beispiel

```
:  
j_drawstring(canvas,100,50,"Hello World")  
:
```



enable
--------

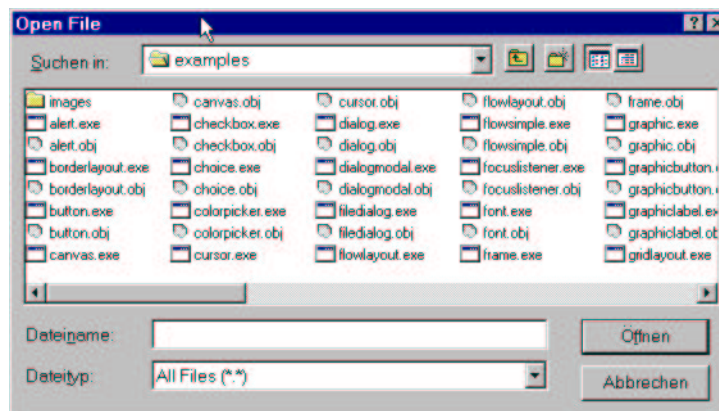
Synopsis            subroutine **j\_enable** ( obj )

Beschreibung      enabled die Komponente **obj**.

Zielkomponenten   Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphiclabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevensegment, MenuItem, CheckBoxMenuItem, Menu, HelpMenu, Popupmenu

## filedialog

Synopsis	<code>function\$ <b>j_filedialog</b> ( frame , title\$ , directory\$ )</code>
Beschreibung	Öffnet eine Filedialogbox in dem angegebenen <b>directory</b> mit dem Title <b>title</b> und liefert den ausgewählten Dateiname in <b>filename</b> zurück. Ist <code>"/S"</code> im Titel enthalten wird die SAVE-Filedialogbox aufgerufen. Der Teilstring <code>"/S"</code> wird aus dem Titelstring entfernt.
Zielkomponenten	Frame
Beispiel	<pre> :  filename = j_filedialog(frame,"Save/S File","..  : </pre>



## fileselect

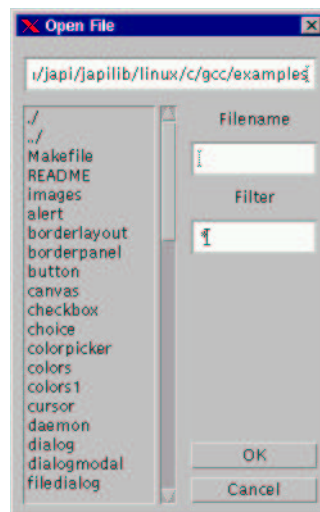
Synopsis            `function$ j_fileselect ( frame , title$ , filter$ )`

Beschreibung      Öffnet eine Fileselectorbox mit dem voreingestellten **filename** und dem Title **title** und liefert den ausgewählten Dateiname in **filename** zurück. Im Gegensatz zum Filedialog kann ein **filter** zur Dateiauswahl angegeben werden. Da der Fileselector auf Applikationsebene realisiert ist, kann er auch bei einer Ausgabeumlenkung verwendet werden.

Zielkomponenten   Frame

Beispiel

```
:  
filename = j_fileselect(frame,"Open File","*")  
:
```





## fillarc

- Synopsis            subroutine **j\_fillarc** ( obj , x , y , rx , ry , arc1 , arc2 )
- Beschreibung        Zeichnet einen gefüllten Kreisbogen mit dem Mittelpunkt **(x,y)** und den Hauptradien **rx** und **ry**. Die Parameter **arc1** und **arc2** bestimmen den Start- bzw. den Endwinkel des Kreisbogens. Bezugswert ist die Horizontale von der die Winkel entgegen des Uhrzeigersinns abgetragen werden. Die Winkel sind im Bogenmaß anzugeben.  
Nullpunkt des Koordinatensystems ist die linke obere Ecke der Komponente die durch **obj** referenziert wird.
- Zielkomponenten    Canvas, Image, Printer

### Beispiel

```

:
canvas = j_canvas(frame,200,100)
j_fillarc(canvas,100,50,80,40,45,-270)
:

```



## fillcircle

Synopsis            subroutine **j\_fillcircle** ( obj , x , y , r )

Beschreibung     Zeichnet einen gefülltes Kreis mit dem Mittelpunkt **(x,y)** und dem Radius **r**. Nullpunkt des Koordinatensystems ist die linke obere Ecke der Komponenten, die durch **obj** referenziert wird.

Zielkomponenten   Canvas, Image, Printer

Beispiel

```
:  
canvas = j_canvas(frame,200,100)  
j_fillcircle(canvas,100,50,40)  
:
```



## filloval

- Synopsis            subroutine **j\_filloval** ( obj , x , y , rx , ry )
- Beschreibung        Zeichnet einen gefüllten Oval mit dem Mittelpunkt (**x,y**) und den Hauptradien **rx** und **ry** in der momentan gesetzten Vordergrundfarbe. Nullpunkt des Koordinatensystems ist die linke obere Ecke der Komponenten, die durch **obj** referenziert wird.
- Zielkomponenten    Canvas, Image, Printer
- Beispiel

```
:  
canvas = j_canvas(frame,200,100)  
j_filloval(canvas,100,50,80,40)  
:
```



## fillpolygon

Synopsis            subroutine **j\_fillpolygon** ( obj , len , x , y )

Beschreibung     Zeichnet eine gefülltes Polygon in der momentan gesetzten Vordergrundfarbe durch die Punkte, die durch die Arrays **x** und **y** gegeben sind. Beide Arrays müssen die mindestens **len** Elemente besitzen. Das Polygon wird automatisch geschlossen. Daher müssen die erste und letzte Koordinate nicht übereinstimmen. Eingeschlossene Bereiche werden gefüllt dargestellt. Nullpunkt des Koordinatensystems ist die linke obere Ecke der Komponente die durch **obj** referenziert wird.

Zielkomponenten   Canvas, Image, Printer

Beispiel

```
:  
rem data x = 20,40,60,80,100,120,140,160,180,200  
rem data y = 10,40,10,40,10,40,10,40,10,40  
canvas = j_canvas(frame,256,50)  
j_fillpolygon(canvas,10,x,y)  
:
```



## fillrect

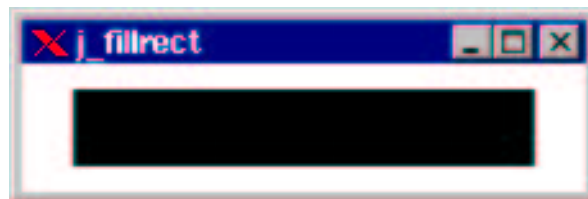
Synopsis            subroutine **j\_fillrect** ( obj , x , y , width , height )

Beschreibung        Zeichnet eine gefülltes Rechteck in der momentan gesetzten Vordergrundfarbe an die Position (**x,y**) mit der Breite **width** und der Höhe **height**. Bezugspunkt des Rechtecks ist die linke obere Ecke. Nullpunkt des Koordinatensystems ist die linke obere Ecke der Komponenten, die durch **obj** referenziert wird.

Zielkomponenten    Canvas, Image, Printer

Beispiel

```
:  
canvas = j_canvas(frame,220,50)  
j_fillrect(canvas,20,10,180,30)  
:
```





focuslistener
---------------

Synopsis	function <b>j_focuslistener</b> ( obj )
Beschreibung	Bindet an die Komponente <b>obj</b> einen neuen Focuslistener, und liefert dessen Eventnummer zurück.
Zielkomponenten	Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphiclabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevenssegment

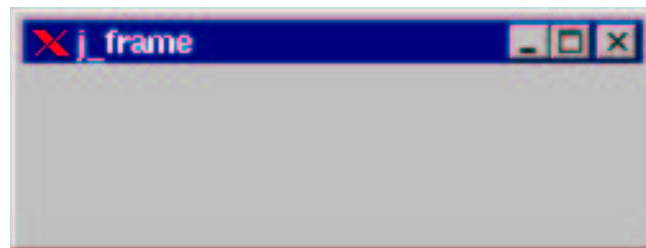
## frame

Synopsis      `function j_frame ( label$ )`

Beschreibung      Diese Funktion erzeugt einen Frame mit der Titelzeile **label**. Ein Frame ist *das* elementare Fenster einer Applikation, an die die anderen graphischen Elemente gebunden werden. Die Funktion liefert die Eventnummer des Frames zurück. Ein Event wird erzeugt, indem der Benutzer das Close Icon im Fensterrahmen anklickt. Ein Frame wird zunächst nur erzeugt, aber nicht angezeigt. Erst nach dem Aufruf der Prozedur *j\_show(frame)* wird der Frame sichtbar.

Beispiel

```
:  
frame = j_frame("j_frame")  
j_show(frame)  
:
```





getaction
-----------

Synopsis      function **j\_getaction** ( )

Beschreibung      Liefert den nächsten Benutzerevent, oder 0 falls kein weiterer Event vorliegt.

## getcolumns

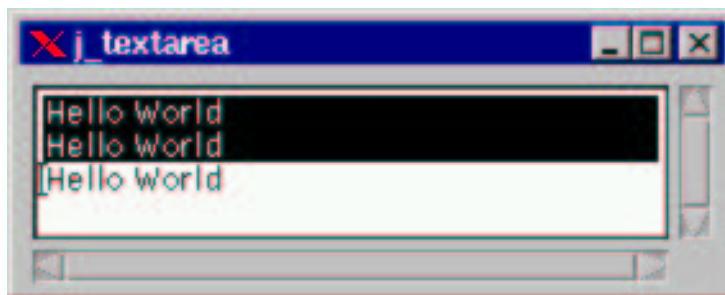
Synopsis            subroutine **j\_getcolumns** ( obj )

Beschreibung      Gibt die Anzahl der Spalten von **obj** zurück.

Zielkomponenten   Textarea, Textfield, Gridlayout

Beispiel

```
:  
text = j_text(frame,30,4)  
j_getcolumns(text)  
:  
> 30
```



getcurpos
-----------

Synopsis      function **j\_getcurpos** ( obj )

Beschreibung      Liefert die Position des Text Cursors zurück.

Zielkomponenten      Textarea, Textfield

getdanger
-----------

Synopsis            subroutine **j\_getdanger** ( obj )

Beschreibung      Liefert den Warnlevel der Komponenten **obj** zurück.

Zielkomponenten   Meter

getfontascent
---------------

Synopsis	function <b>j_getfontascent</b> ( obj )
Beschreibung	Liefert die Oberlänge (Ascent) des aktuellen Fonts der Komponenten <b>obj</b> in Bildschirmpixeln zurück.
Zielkomponenten	Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphiclabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevenssegment

getfontheight
---------------

Synopsis	function <b>j_getfontheight</b> ( obj )
Beschreibung	Liefert die totale Höhe des aktuellen Fonts der Komponenten <b>obj</b> in Bildschirmpixeln zurück.
Zielkomponenten	Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphicalabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevensegment

## getheight

Synopsis	function <b>j_getheight</b> ( obj )
Beschreibung	Die Funktion liefert die Höhe der Komponenten <b>obj</b> zurück. Neben den graphischen Elementen kann auch die Höhe eines Image ermittelt werden. Die Maßeinheit sind Bildschirmpixel.
Zielkomponenten	Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphiclabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevenssegment, Image
Beispiel	<pre>: label = j_getlabel(frame,"Hello World") print j_getheight(label) : &gt; 22</pre>

getimagesource

Synopsis	function <b>j_getimagesource</b> ( obj , x , y , w , h , r , g , b )
Beschreibung	Die Funktion liefert einen Ausschnitt vom Punkt ( <b>x</b> , <b>y</b> ) der Breite <b>w</b> und der Höhe <b>h</b> zurück. Der Bildinhalt wird als rot-, grün- und blau- Kanal in den Arrays r,g,b abgelegt.
Zielkomponenten	Canvas, Image



getimage

Synopsis	function <b>j_getimage</b> ( obj )
Beschreibung	Kopiert den Inhalt der Komponenten <b>obj</b> in ein Image und liefert dessen Eventnummer zurück.
Zielkomponenten	Canvas, Image

getinheight
-------------

Synopsis           function **j\_getinheight** ( cont )

Beschreibung       Die Funktion liefert die Höhe des inneren Bereich der Komponenten **obj** zurück. Die Maßeinheit sind Bildschirmpixel.

Zielkomponenten   Panel, Borderpanel, Window, Dialog, Frame

Beispiel

```
:  
frame = j_frame("Hello World")  
j_setsize(frame,300,400)  
print j_getinheight(frame))  
:  
> 370
```

## getinsets

Synopsis            function **j\_getinsets** ( obj , side )

Beschreibung      Liefert die angeforderte Breite des inneren Randes der Komponenten zurück. Der Parameter **side** kann folgende Werte annehmen:

- J.TOP: die Höhe des oberen Rands wird zurueckgeliefert.
- J.BOTTOM: die Höhe des unteren Rands wird zurueckgeliefert.
- J.LEFT: die Breite des linken Rands wird zurueckgeliefert.
- J.RIGHT: die Breite des rechten Rands wird zurueckgeliefert.

Zielkomponenten   Panel, Borderpanel, Window, Dialog, Frame

Beispiel

```

:
frame = j_frame("j_getinsets")
print j_getinsets(frame,J_TOP) ,j_getinsets(frame,J_BOTTOM) ,
      j_getinsets(frame,J_LEFT) ,j_getinsets(frame,J_RIGHT)
:
> 25 5 5 6

```



getinwidth
------------

Synopsis           function **j\_getinwidth** ( cont )

Beschreibung       Die Funktion liefert die Breite des inneren Bereich der Komponenten **obj** zurück. Die Maßeinheit sind Bildschirmpixel.

Zielkomponenten   Panel, Borderpanel, Window, Dialog, Frame

Beispiel

```
:  
frame = j_frame("Hello World")  
j_setsize(frame,300,400)  
print j_getinwidth(frame)  
:  
> 289
```

getitemcount
--------------

- Synopsis      function **j\_getitemcount** ( obj )
- Beschreibung      Liefert die Anzahl der Items der Komponenten **obj** zurück.
- Zielkomponenten      List, Choice

getitem
---------

Synopsis      `function$ j_getitem ( obj , item )`

Beschreibung      liefert den Inhalt des Items mit dem Index **item** zurück.

Zielkomponenten      List, Choice

getkeychar
------------

Synopsis	function <b>j_getkeychar</b> ( obj )
Beschreibung	Liefert den ASCII Code, der zuletzt gedrückten Taste zurück.
Zielkomponenten	KeyListener

getkeycode
------------

Synopsis      function **j\_getkeycode** ( obj )

Beschreibung      Liefert den Keycode, der zuletzt gedrückten Taste zurück.

Zielkomponenten      Keylistener



getlayoutid
-------------

Synopsis	function <b>j_getlayoutid</b> ( obj )
Beschreibung	Liefert die Eventnummer des Layoutmanagers des Containers <b>obj</b> zurück.
Zielkomponenten	Panel, Borderpanel, Window, Dialog, Frame
Beispiel	<pre>: j_setgridlayout(frame,2,2) grid = j_getlayoutid(frame) :</pre>

getlength
-----------

Synopsis	function <b>j_getlength</b> ( obj )
Beschreibung	Liefert die Länge des Textinhaltes die Komponente zurück.
Zielkomponenten	Textarea, Textfield, Dialog, Frame, Button, MenuItem, CheckBox-MenuItem, Menu, HelpMenu, Popupmenu

getmousebutton
----------------

Synopsis	function <b>j_getmousebutton</b> ( mouset listener )
Beschreibung	Liefert die letzte gedrückte Mousetaste zurück. Der Rückgabewert ist <ul style="list-style-type: none"><li>• J_LEFT linke Mousetaste</li><li>• J_CENTER mittlere Mousetaste</li><li>• J_RIGHT rechte Mousetaste</li></ul>
Zielkomponenten	Mouset listener

getmousex

Synopsis      function **j\_getmousex** ( mouset listener )

Beschreibung      Liefert die aktuelle horizontale Position der Mouse zurück.

Zielkomponenten      Mouset listener

getmousey
-----------

Synopsis      function **j\_getmousey** ( mouset listener )

Beschreibung      Liefert die aktuelle vertikale Position der Mouse zurück.

Zielkomponenten      Mouset listener

## getparentid

Synopsis           function **j\_getparentid** ( obj )

Beschreibung     Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird **-1** zurückgegeben.

Zielkomponenten  Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphiclabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevensegment, Menubar, MenuItem, CheckBoxMenuItem, Menu, HelpMenu, Popupmenu, Radiogroup

Beispiel

```
:  
radio1       = j_radiobutton(j_radiogroup(frame), "Radiobutton 1")  
radio2       = j_radiobutton(j_getparentid(radio1), "Radiobutton 2")  
:
```



## getparent

Synopsis            `function j_getparent ( obj )`

Beschreibung      Liefert die Eventnummer des umgebenen Containers von **obj** zurück. Ist **obj** ein Frame wird `-1` zurückgegeben.

Zielkomponenten   Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphiclabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevenssegment, Menubar, MenuItem, CheckBoxMenuItem, Menu, HelpMenu, Popupmenu, Radiogroup

Beispiel

```

:
radio1        = j_radiobutton(j_radiogroup(frame),"Radiobutton 1")
radio2        = j_radiobutton(j_getparent(radio1),"Radiobutton 2")
:

```



getrows

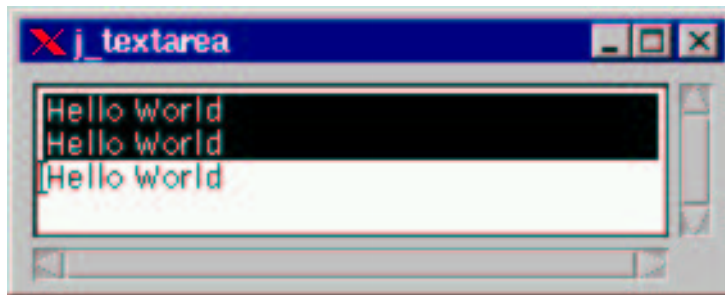
Synopsis            subroutine **j\_getrows** ( obj )

Beschreibung      Gibt die Anzahl der Reihen von **obj** zurck.

Zielkomponenten   Textarea, Gridlayout

Beispiel

```
:  
text = j_text(frame,30,4)  
j_getrows(text)  
:  
> 4
```





getscaledimage

Synopsis	function <b>j_getscaledimage</b> ( obj , x , y , sw , sh , tw , th )
Beschreibung	Kopiert den Inhalt des Ausschnitts von Positio ( <b>x</b> , <b>y</b> ) it der Breite <b>sw</b> und der höhe <b>sh</b> in ein Image und liefert dessen Eventnummer zurück. Das Image wird dabei auf eine Breite von <b>tw</b> und einer Höhe von <b>th</b> scaliert.
Zielkomponenten	Canvas, Image

getscreenheight

Synopsis           function **j\_getscreenheight** ()

Beschreibung     Liefert die Höhe des Bildschirms in Pixeln zurück. Ist ein virtueller Bildschirm installiert, so wird dessen Höhe zurückgeliefert.

Beispiel

```
:  
print j_getscreenwidth(), j_getscreenheight()  
:  
  
> 1280 1024
```

getscreenwidth
----------------

Synopsis           function **j\_getscreenwidth** ()

Beschreibung      Liefert die Breite des Bildschirms in Pixeln zurück. Ist ein virtueller Bildschirm installiert, so wird dessen Breite zurückgeliefert.

Beispiel

```
:  
print j_getscreenwidth(), j_getscreenheight()  
:  
  
> 1280 1024
```

getselect
-----------

Synopsis        `function j_getselect ( obj )`

Beschreibung    Liefert den Index des selectierten Items der Komponenten .

Zielkomponenten   List, Choice

getselend
-----------

Synopsis	function <b>j_getselend</b> ( obj )
Beschreibung	Liefert die Endposition des selectierten Textes.
Zielkomponenten	Textarea, Textfield

getselstart

Synopsis      function **j\_getselstart** ( obj )

Beschreibung      Liefert die Anfangsposition des selectierten Textes.

Zielkomponenten      Textarea, Textfield

getseltext

Synopsis	function\$ <b>j_getseltext</b> ( obj )
Beschreibung	Liefert den selectierten Text die Komponente <b>obj</b> zurück.
Zielkomponenten	Textarea, Textfield

getstate
----------

Synopsis      function **j\_getstate** ( obj )

Beschreibung      Liefert J\_TRUE , falls die Komponente selektiert ist, sonst J\_FALSE .

Zielkomponenten      Checkbox, Radiobutton, CheckmenuItem, Led



getstringwidth
----------------

Synopsis	function <b>j_getstringwidth</b> ( obj , str\$ )
Beschreibung	Liefert die Länge des <b>str</b> des aktuellen Fonts der Komponenten <b>obj</b> in Bildschirmpixeln zurück.
Zielkomponenten	Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphiclabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevenssegment

gettext
---------

Synopsis	function\$ <b>j_gettext</b> ( obj )
Beschreibung	Liefert den Textinhalt der, durch <b>obj</b> referenzierten, Komponenten im Parameter <b>str</b> zurück. In C und in Pascal wird ein Pointer auf <b>str</b> zurückgeliefert. Der übergebene String muß groß genug sein, um den gesamten Inhalt des Textes aufzunehmen. Ist die Komponente vom Typ Frame oder Dialog, so wird der Titelstring des Fensters zurückgegeben. Ansonsten wird der Textinhalt der Komponenten zurückgeliefert.
Zielkomponenten	Button, Label, Checkbox, Radiobutton, Dialog, Frame, MenuItem, CheckBoxMenuItem, Menu, HelpMenu, Popupmenu, Textarea, Textfield
Beispiel	<pre>: label = j_label(frame,"Hello World") print j_gettext(label) :  &gt; Hello World</pre>

getvalue
----------

Synopsis      function **j\_getvalue** ( obj )

Beschreibung      Liefert den aktuellen Wert der Komponenten .

Zielkomponenten      Scrollbar

getviewportheight

Synopsis      function **j\_getviewportheight** ( obj )

Beschreibung      Gibt die Höhe des sichtbaren Teils der Komponenten **obj** zurück.

Zielkomponenten      Scrollpane

getviewportwidth
------------------

Synopsis      function **j\_getviewportwidth** ( obj )

Beschreibung      Gibt die Breite des sichtbaren Teils der Komponenten **obj** zurück.

Zielkomponenten      Scrollpane

getwidth

Synopsis	function <b>j_getwidth</b> ( obj )
Beschreibung	Die Funktion liefert die Breite der Komponenten <b>obj</b> zurück. Neben den graphischen Elementen kann auch die Breite eines Image ermittelt werden. Die Maßeinheit sind Bildschirmpixel.
Zielkomponenten	Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphiclabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevenssegment, Image
Beispiel	<pre>: label = j_getlabel(frame,"Hello World") print j_getwidth(label) : &gt; 84</pre>

getxpos
---------

Synopsis	function <b>j_getxpos</b> ( obj )
Beschreibung	Liefert die aktuelle horizontale Position der Komponenten <b>obj</b> zurück.
Zielkomponenten	Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphiclabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevenssegment

getypos

Synopsis	function <b>j_getypos</b> ( obj )
Beschreibung	Liefert die aktuelle vertikale Position der Komponenten <b>obj</b> zurück.
Zielkomponenten	Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphiclabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevenssegment



## graphicbutton

Synopsis            `function j_graphicbutton ( obj , filename$ )`

Beschreibung      Die Funktion erzeugt einen neuen Button mit einem Icon als Inhalt. Die Datei, die das Icon enthält, ist im Parameter **filename** anzugeben. Das Icon muß im GIF oder JPEG-Format vorliegen. Die Funktion liefert die Eventnummer des Buttons zurück, oder im Fehlerfall `-1`. Ein Event wird erzeugt, wenn der Benutzer mit der Maus den Button betätigt.

Zielkomponenten   Panel, Borderpanel, Window, Dialog, Frame

Beispiel

```
:  
frame = j_frame("j_graphicbutton")  
button = j_graphicbutton(frame,"save.gif")  
:
```



## graphiclabel

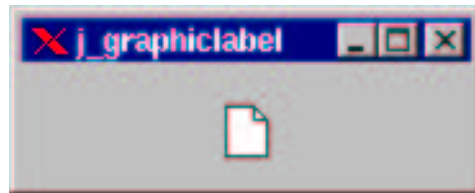
Synopsis           function **j\_graphiclabel** ( obj , str\$ )

Beschreibung       Die Funktion erzeugt ein neues Label mit einem Icon als Inhalt. Die Datei, die das Icon enthält, ist im Parameter **filename** anzugeben. Das Icon muß im GIF oder JPEG-Format vorliegen. Die Funktion liefert die Eventnummer des Labels zurück, oder im Fehlerfall -1.

Zielkomponenten   Panel, Borderpanel, Window, Dialog, Frame

Beispiel

```
:  
frame = j_frame("j_graphiclabel")  
label = j_graphiclabel(frame,"new.gif")  
:
```



hasfocus
----------

Synopsis	function <b>j_hasfocus</b> ( obj )
Beschreibung	Liefert J_TRUE zurück, falls die Komponente den Focus besitzt, sonst J_FALSE .
Zielkomponenten	Focuslistener

## helpmenu

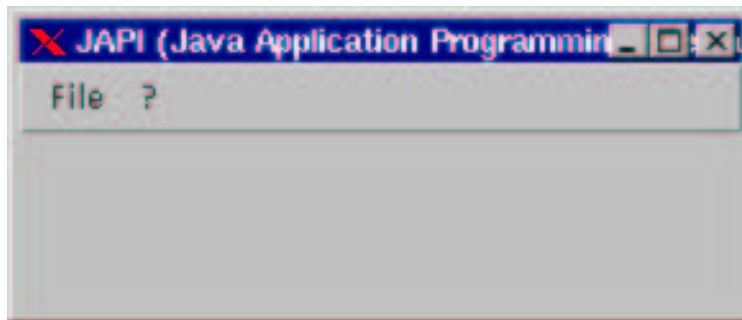
Synopsis      `function j_helpmenu ( obj , label$ )`

Beschreibung      Die Funktion erzeugt einen Hilfe-Menueintrag mit der Beschriftung **label** in der Komponenten **obj**, und liefert dessen Eventnummer zurück. An diese Menukomponente können in weiteren Schritten MenuItem gebunden werden. Eine Menukomponente selbst erzeugt keinen Event. Hilfe Menü unterscheiden sich von normalen Menüs dadurch, daß sie je nach Oberfläche auch rechteckig erscheinen können. Zudem sind als Zielkomponenten nur Menubars zulässig.

Zielkomponenten      Menubar

Beispiel

```
:  
frame = j_frame("Menu Komponenten")  
menubar = j_menubar(frame)  
file= j_menu(menubar,"File")  
help= j_helpmenu(menubar,"?")  
:
```



hide
------

Synopsis	subroutine <b>j_hide</b> ( obj )
Beschreibung	Zeigt die Komponente <b>obj</b> nicht mehr auf dem Bildschirm an.
Zielkomponenten	Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphiclable, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevensegment

## hscrollbar

Synopsis      `function j_hscrollbar ( obj )`

Beschreibung      Diese Funktion erzeugt eine horizontale Scrollbar und gibt dessen Eventnummer zurück. Im Fehlerfall wird `-1` zurückgegeben. Ist die Zielkomponente **obj** vom Typ Scrollpane, so wird die Eventnummer der vorhandenen horizontalen Scrollbar zurückgegeben. Eine Scrollbar erzeugt einen Event, sobald der Schieber bewegt wird.

Zielkomponenten      Panel, Borderpanel, Window, Dialog, Frame, Scrollpane

Beispiel

```
:  
scroll=j_hscrollbar(frame)  
j_setpos(scroll,20,40)  
j_setsize(scroll,150,20)  
:
```

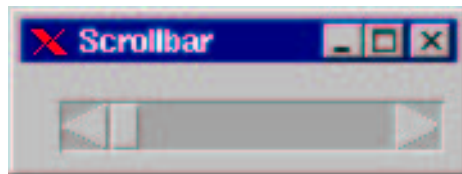


image
-------

Synopsis           function **j\_image** ( width , height )

Beschreibung     Diese Funktion erzeugt einen neues (Memory) Image mit der Breite **width** Pixel und einer Höhe von **height** Pixel. Die Funktion liefert die Eventnummer des Image zurück, oder im Fehlerfall  $-1$ . Ein Image erzeugt keinen Event

Beispiel

```
:  
image = j_image(200,200)  
:
```

insert
--------

Synopsis      function **j\_insert** ( obj , pos , label\$ )

Beschreibung      fügt ein neues Item in an der Position **pos** ein. Das Item trägt den Inhalt **label**.

Zielkomponenten      List, Choice



inserttext
------------

- Synopsis            subroutine **j\_inserttext** ( obj , text\$ , pos )
- Beschreibung       Fügt zusätzlichen Text **text** an der Position **pos** ein.
- Zielkomponenten   Textarea

isparent
----------

Synopsis	function <b>j_isparent</b> ( obj , cont )
Beschreibung	Liefert J_TRUE zurück, falls das <b>cont</b> die umgebende Komponente von <b>obj</b> ist, sonst J_FALSE
Zielkomponenten	Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphiclabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevenssegment, Menubar, MenuItem, CheckBoxMenuItem, Menu, HelpMenu, Popupmenu, Radiogroup

isresizable
-------------

Synopsis	function <b>j_isresizable</b> ( obj )
Beschreibung	Liefert J_FALSE zurueck, falls die Komponente eine fixe Größe besitzt, J_TRUE sonst.
Zielkomponenten	Dialog, Frame

isselect
----------

Synopsis           function **j\_isselect** ( obj , item )

Beschreibung     Liefert J\_TRUE zurück, falls das Item mit dem Index **item** selectiert ist, sonst J\_FALSE

Zielkomponenten   List

isvisible

Synopsis	function <b>j_isvisible</b> ( obj )
Beschreibung	Liefert J_TRUE zurück, falls <b>obj</b> sichtbar ist, sonst J_FALSE
Zielkomponenten	Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphiclabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevensegment

keylistener
-------------

Synopsis	function <b>j_keylistener</b> ( obj )
Beschreibung	Bindet an die Komponente <b>obj</b> einen neuen Keylistener, und liefert dessen Eventnummer zurück.
Zielkomponenten	Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphiclabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevenssegment

## label

Synopsis      `function j_label ( obj , label$ )`

Beschreibung      Die Funktion erzeugt ein neues Label Element mit dem Textinhalt **label**. Ein Label stellt einen einzeiligen Text beliebiger Länge dar. Die Funktion liefert die Eventnummer des Labels zurück, oder im Fehlerfall `-1`.

Zielkomponenten      Panel, Borderpanel, Window, Dialog, Frame

Beispiel

```
:  
frame = j_frame("j_label")  
label = j_label(frame,"Hello World")  
:
```



## led

Synopsis      function **j\_led** ( obj , style , color )

Beschreibung      Die Funktion erzeugt ein neues LED-Element und liefert die Eventnummer der Komponente zurück. Die LED kann eine runde Form besitzen **style = J\_ROUND** oder eine eckige Form besitzen **style=J\_RECT**. Die Farbe der LED kann eine der vordefinierten Farben sein (zB. J\_RED, J\_GREEN).

Zielkomponenten      Panel, Borderpanel, Window, Dialog, Frame

Beispiel

```
:  
led1 = j_led(frame,J_ROUND,J_RED)  
led2 = j_led(frame,J_RECT,J_BLUE)  
:
```





## line

Synopsis            `function j_line ( obj , orient , style , length )`

Beschreibung     Die Funktion erzeugt ein Linienelement und liefert dessen Eventnummer zurück. Im Fehlerfall wird `-1` zurückgegeben. Das Element wird der Komponente zugeordnet, die durch **obj** referenziert wird. Die initiale Laenge der Linie wird durch den Parameter **length** bestimmt. Der Ruler dient der optischen Trennung anderer Komponenten.  
Der Parameter **orient** gibt an, ob eine senkrechte oder waagerechte Linie gezogen werden soll:

- `J.HORIZONTAL` : waagerechte Linie
- `J.VERTICAL` : senkrechte Linie

Der Parameter **style** gibt den Linientyp vor:

- `J.LINEDOWN` : Der Linie scheint optisch abgesenkt.
- `J.LINEUP` : Der Linie scheint optisch hervorzutreten.

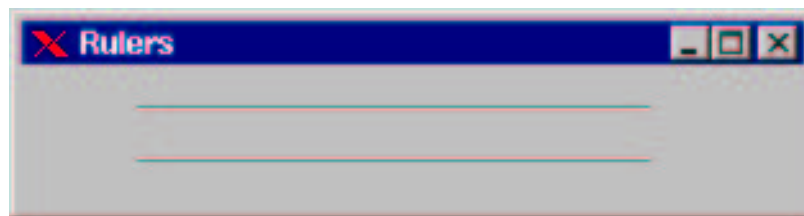
Zielkomponenten   Panel, Borderpanel, Window, Dialog, Frame

Beispiel

```

:
line1 = j_line(frame,J_HORIZONTAL,J_LINEDOWN,200)
line2 = j_line(frame,J_HORIZONTAL,J_LINEUP,200)
:

```



## list

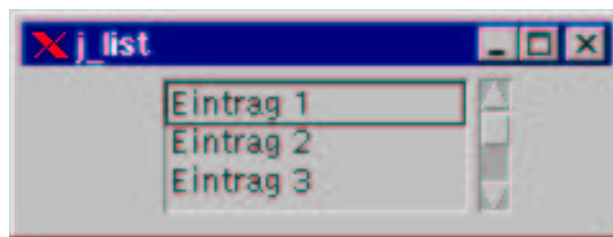
Synopsis      function **j\_list** ( obj , rows )

Beschreibung      Diese Funktion erzeugt eine neue List und gibt deren Eventnummer zurück.Im Fehlerfall wird  $-1$  zurückgegeben. Eine List bietet dem Benutzer die Auswahl zwischen mehreren vordefinierten Einträgen. Es sind immer **rows** Einträge sichtbar, die anderen Einträge sind über Rollbalken zu erreichen. Eine List bietet die Möglichkeit, sowohl Einfach- als auch Mehrfachselektionen zuzulassen (siehe *j\_multiplemode()*). Bei Einfachselektion liefert die List bei einem Doppelklick auf einen Eintrag einen Event. Ist eine Mehrfachselektion eingestellt, löst jeder einfache Klick auf einen Eintrag einen Event aus.

Zielkomponenten      Panel, Borderpanel, Window, Dialog, Frame

Beispiel

```
:  
list = j_list(frame,3)  
j_additem(list,"Eintrag 1")  
j_additem(list,"Eintrag 2")  
:
```



loadimage
-----------

Synopsis            `function j_loadimage ( filename$ )`

Beschreibung      Liest das Image aus dem File **filename** ein und liefert dessen Eventnummer zurück. Es werden folgende Fileformate akzeptiert:

- GIF
- JPEG
- BMP
- PPM

Beispiel

```
:  
image = j_loadimage("mandel.jpg")  
:
```

## menubar

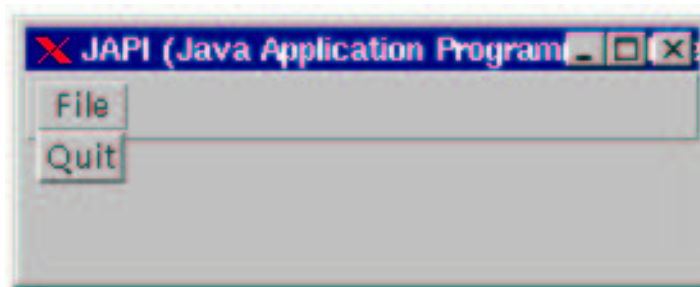
Synopsis      `function j_menubar ( obj )`

Beschreibung      Die Funktion erzeugt eine neue Menubar in der Komponente, die durch **obj** referenziert wird, und liefert dessen Eventnummer zurück. An die Menubar können in weiteren Schritten Menu Komponenten gebunden werden. Eine Menubar selbst erzeugt keinen Event. Als Zielkomponenten sind nur Frames zulässig.

Zielkomponenten      Frame

Beispiel

```
:
frame   = j_frame("Menu Komponenten")
menubar = j_menubar(frame)
file    = j_menu(menubar,"File")
quit    = j_menuitem(file,"Quit")
:
```



## menuItem

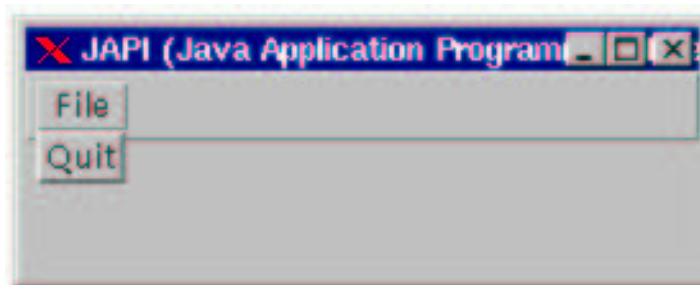
Synopsis            `function j_menuitem ( obj , label$ )`

Beschreibung      Die Funktion erzeugt ein MenuItem mit der Beschriftung **label** in der Komponente **obj**, und liefert dessen Eventnummer zurück. Eine MenuItem erzeugt einen Event, wenn es durch einen Mausklick angewählt wurde. Als Zielkomponenten sind nur Menu, Popupmenu und Helpmenu zulässig.

Zielkomponenten   Menu, Popupmenu, Helpmenu

Beispiel

```
:  
frame   = j_frame("Menu Komponenten")  
menubar = j_menubar(frame)  
file    = j_menu(menubar,"File")  
quit    = j_menuitem(file,"Quit")  
:
```



menu
------

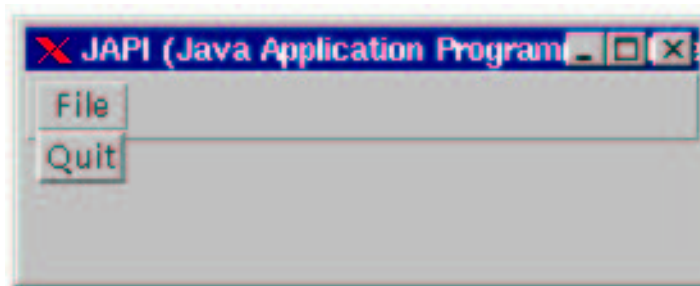
Synopsis      `function j_menu ( obj , str$ )`

Beschreibung      Die Funktion erzeugt einen Menueintrag mit der Beschriftung **label** in der Komponente, die durch **obj** referenziert wird, und liefert dessen Eventnummer zurück. An diese Menukomponente können in weiteren Schritten MenuItem gebunden werden. Eine Menukomponente selbst erzeugt keinen Event. Als Zielkomponenten sind nur Menubars und andere Menukomponenten zulässig.

Zielkomponenten      Menubar, Menu

Beispiel

```
:  
frame = j_frame("Menu Komponenten")  
menubar = j_menubar(frame)  
file = j_menu(menubar,"File")  
quit = j_menuitem(file,"Quit")  
:
```



## messagebox

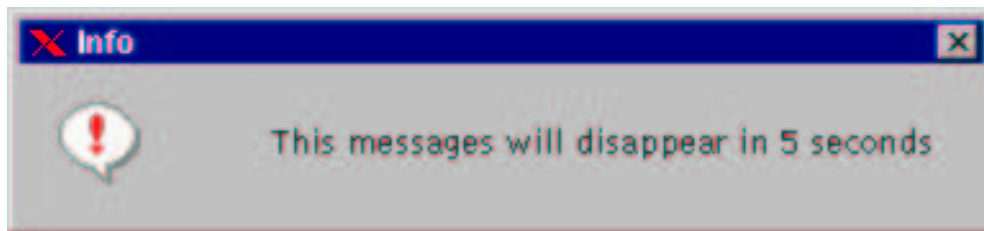
Synopsis            subroutine **j\_messagebox** ( obj , title\$ , text\$ )

Beschreibung      Öffnet eine Messagebox mit der Überschrift **title** und dem Textinhalt **text** und liefert dessen Eventnummer zurück. Im Fehlerfall wird `-1` zurückgeliefert. Eine Messagebox erzeugt einen Event, wenn das Close Icon im Fensterrahmen angeklickt wird.

Zielkomponenten   Frame

Beispiel

```
:  
mbox = j_messagebox(frame,"Info","This messages will disappear in 5 seconds")  
j_sleep(5000)  
j_dispose(mbox)  
:
```



meter

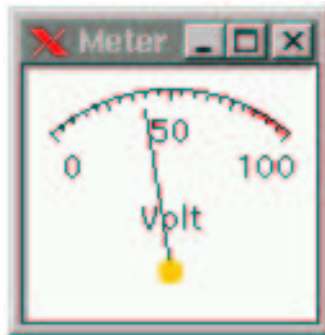
Synopsis            `function j_meter ( obj , title$ )`

Beschreibung        Die Funktion erzeugt ein analoges Anzeigedisplay und liefert die Eventnummer der Komponente zurück. Das Instrument trägt die Beschriftung **title**. Das Instrument besitzt einen vordefinierten Bereich von 0 bis 100. Die Werte können mit `j_setmin()` und `j_setmax()` verändert werden. Ein Gefahr-Bereich ist ab dem Wert 80 voreingestellt. Dies kann mit `j_setdanger()` verändert werden.

Zielkomponenten    Panel, Borderpanel, Window, Dialog, Frame

Beispiel

```
:  
meter = j_meter(frame,"Volt")  
j_setvalue(meter,40)  
:
```





mouselistener
---------------

Synopsis	function <b>j_mouselistener</b> ( obj , kind )
Beschreibung	<p>Bindet an die Komponente <b>obj</b> einen neuen Mouselistener, und liefert dessen Eventnummer zurück. Der Parameter <b>kind</b> bestimmt, wann ein Event ausgelöst wird. Zulässige Werte für <b>kind</b> sind:</p> <ul style="list-style-type: none"> <li>• J_ENTERED : Der Mouselistener liefert einen Event, wenn der Mauszeiger die Komponente <b>obj</b> betritt.</li> <li>• J_MOVED : Der Mouselistener liefert einen Event, wenn der Mauszeiger sich innerhalb der Komponenten <b>obj</b> bewegt.</li> <li>• J_EXITED : Der Mouselistener liefert einen Event, wenn der Mauszeiger die Komponente <b>obj</b> verläßt.</li> <li>• J_PRESSED : Der Mouselistener liefert beim Drücken einer Mousetaste einen Event.</li> <li>• J_DRAGGED : Der Mouselistener liefert einen Event, wenn der Mauszeiger sich innerhalb der Komponenten <b>obj</b> mit gedrückter Taste bewegt.</li> <li>• J_RELEASED : Der Mouselistener liefert beim Loslassen einer Mousetaste einen Event.</li> <li>• J_DOUBLECLICK : Der Mouselistener liefert bei einem Doppelclick einen Event.</li> </ul>
Zielkomponenten	Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphiclabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevensegment

multiplemode
--------------

Synopsis	function <b>j_multiplemode</b> ( obj , bool )
Beschreibung	die Komponente erlaubt mehrfache Selectierung, falls <b>bool</b> J_TRUE ist.
Zielkomponenten	List

nextaction
------------

Synopsis      function **j\_nextaction** ( )

Beschreibung      Wartet auf den nächsten Benutzerevent und liefert diesen zurück.

## pack

Synopsis            subroutine **j\_pack** ( obj )

Beschreibung     Diese Prozedur veranlaßt den Layoutmanager von **obj** die minimale Größe zu berechnen. Ist die Komponente sichtbar, so wird das neue Layout direkt angezeigt. Der Layoutmanager erfragt von allen enthaltenen Komponenten deren initiale bzw. explizit zugewiesene Größe, und berechnet daraus ein neues Layout.

Zielkomponenten   Panel, Borderpanel, Window, Dialog, Frame

Beispiel

```
:  
j_setflowlayout(jframe,J_HORIZONTAL)  
canvas = j_canvas(frame,200,50)  
j_setnamedcolorbg(canvas,J_RED)  
j_pack(frame)  
:
```



## panel

Synopsis            `function j_panel ( obj )`

Beschreibung        Diese Funktion erzeugt ein neues panel und liefert dessen Eventnummer zurück. Im Fehlerfall wird `-1` zurückgegeben. Das Panel wird der Komponente zugeordnet, die durch `obj` referenziert wird. Ein Panel ist ein (unsichtbares) Behälterelement, in dem andere Elemente angeordnet werden können. Da einem Panel auch ein Layoutmanager zugewiesen werden kann, können mit Hilfe der Panels komplexe hierarchisch aufgebaute Benutzerschnittstellen erstellt werden. Ein Panel liefert bei einer Größenänderung einen Event.

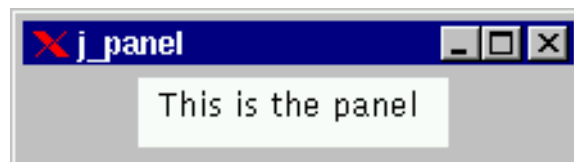
Zielkomponenten    Panel, Borderpanel, Window, Dialog, Frame

Beispiel

```

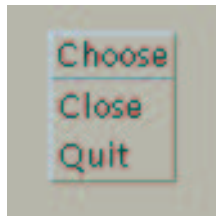
:
panel = j_panel(frame)
j_setnamedcolorbg(panel,J_WHITE)
j_setpos(panel,50,30)
label = j_label(panel,"This is the panel")
j_setpos(label,0,0)
:

```



## popupmenu

Synopsis	function <b>j_popupmenu</b> ( obj , label\$ )
Beschreibung	Diese Funktion erzeugt ein neues Popupmenu und liefert dessen Eventnummer zurück. Im Fehlerfall wird $-1$ zurückgegeben. Ein Popupmenu ist funktional identisch mit einem Menu, es ist allerdings an keine Menuleiste gebunden. Daher kann ein Popupmenu prinzipiell an jeder Position innerhalb der Oberfläche platziert werden. Wie an Menus kann auch an ein Popupmenu MenuItem's gebunden werden, die bei Anklicken des Popupmenus sichtbar werden.
Zielkomponenten	Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphicalabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevenssegment
Beispiel	<pre>: choose = j_popupmenu(frame,"Choose") close  = j_menuitem(choose,"Close") quit   = j_menuitem(choose,"Quit") j_showpopup(choose,100,100) :</pre>



printer
---------

Synopsis            `function j_printer ( frame )`

Beschreibung        Diese Funktion erzeugt ein neues Objekt, das eine Papierseite eines Druckes repräsentiert. Bis auf wenige Ausnahmen ist ein Printer Objekt funktional einem Canvas gleichgestellt. Die Funktion liefert die Eventnummer des Printers zurück, oder im Fehlerfall -1. Ein Printer erzeugt keinen Event.

Zielkomponenten    `Frame`

Beispiel

```
:  
printer = j_printer(frame)  
j_drawimage(printer,image,100,100)  
:
```

## print

Synopsis            subroutine **j\_print** ( obj )

Beschreibung       Die Prozedur druckt den Inhalt der Komponente **obj**. Ist die Komponente von Typ Container, so wird der gesamte Inhalt des Containers gedruckt. Nach dem Aufruf dieser Funktion erscheint ein Plattformabhängiger Print Dialog, in dem weitere Optionen (beispielsweise das Speichern in eine Datei) bereitgestellt werden. Unter X-Windows besitzt der Ausdruck immer Motif Look, unabhängig von der Bildschirmdarstellung.  
Ist die Komponente von Type j\_printer, dann wird die aktuelle Seite geschlossen, und eine neue Seite begonnen. Die Seiten werden noch nicht auf dem Drucker ausgegeben. Um den Druck zu starten muss j\_dispose(printer) aufgerufen werden.

Zielkomponenten   Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphiclabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevensegment, Canvas, Image, Printer

Beispiel

```
:  
frame = j_frame("j_textfield")  
text = j_textfield(frame,30)  
:  
j_print(frame);  
:
```





## progressbar

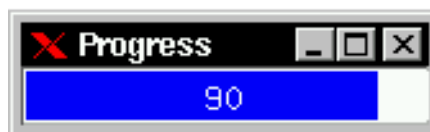
Synopsis            `function j_progressbar ( obj , orient )`

Beschreibung     Die Funktion erzeugt eine Progressbar und liefert die Eventnummer der Komponente zurück. Die Ausrichtung kann `J_HORIZONTAL` oder `J_VERTICAL` sein. Das Instrument besitzt einen vordefinierten Bereich von 0 bis 100. Die Werte können mit `j_setmin()` und `j_setmax()` verändert werden.

Zielkomponenten   Panel, Borderpanel, Window, Dialog, Frame

Beispiel

```
:  
progress = j_progressbar(frame,J_HORIZONTAL)  
j_setvalue(progress,90)  
:
```



quit
------

Synopsis        subroutine **j\_quit** ( )

Beschreibung    Beendet die Verbindung zum JAPI Kernel.

## radiobutton

Synopsis            `function j_radiobutton ( obj , label$ )`

Beschreibung      Diese Funktion erzeugt einen Radiobutton mit dem Textinhalt `label` und liefert dessen Eventnummer zurück. Im Fehlerfall wird `-1` zurückgegeben. Ein Radiobutton muss immer in einer Radiogroup eingebettet sein. Innerhalb dieser Radiogroup kann von allen enthaltenen Radiobuttons immer nur maximal ein Button aktiviert sein. Ein Radiobutton liefert immer dann einen Event, wenn sich sein Zustand ändert.

Zielkomponenten   Radiogroup

Beispiel

```
:  
radiogroup = j_radiogroup(frame)  
radio1     = j_radiobutton(radiogroup,"Radiobutton 1")  
radio2     = j_radiobutton(radiogroup,"Radiobutton 2")  
:
```



## radiogroup

Synopsis            `function j_radiogroup ( obj )`

Beschreibung     Diese Funktion erzeugt eine neue Radiogroup und liefert dessen Eventnummer zurück. Eine Radiogroup ist eine virtuelle Komponente, die dazu dient, Radiobutton funktionell zusammenzuschalten. Von allen Radiobutton, die in einer Radiogroup zusammengefasst sind, kann immer nur maximal ein Button aktiviert sein.

Zielkomponenten   Panel, Borderpanel, Window, Dialog, Frame

Beispiel

```
:  
radiogroup = j_radiogroup(frame)  
radio1     = j_radiobutton(radiogroup,"Radiobutton 1")  
radio2     = j_radiobutton(radiogroup,"Radiobutton 2")  
:
```



random
--------

Synopsis      function **j\_random** ( )

Beschreibung      Liefert eine Pseudo Zufallszahl zwischen 0 und 2147483647 ( $2^{31} - 1$ ) zurück .

releaseall
------------

Synopsis            subroutine **j\_releaseall** ( obj )

Beschreibung      Gibt alle Elemente aus der Komponenten **obj** frei.

Zielkomponenten   Panel, Borderpanel, Window, Dialog, Frame

release
---------

Synopsis	subroutine <b>j_release</b> ( obj )
Beschreibung	Gibt die Komponente <b>obj</b> aus dem umgebenden Container Element frei.
Zielkomponenten	Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphicalabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevensegment

removeall
-----------

Synopsis      function **j\_removeall** ( obj )

Beschreibung      entfernt alle Items aus der Komponenten .

Zielkomponenten      List, Choice



removeitem
------------

Synopsis      function **j\_removeitem** ( obj , item\$ )

Beschreibung      entfernt den ersten Eintrag **item** aus der Komponenten .

Zielkomponenten      List, Choice

remove
--------

Synopsis      function **j\_remove** ( obj , item )

Beschreibung      entfernt das Item mit dem Index **item** aus der Komponenten .

Zielkomponenten      List, Choice

replacetext
-------------

- Synopsis            subroutine **j\_replacetext** ( obj , text\$ , start , end )
- Beschreibung       Ersetzt den Text zwischen Position **start** bis **end** durch **text**.
- Zielkomponenten   Textarea

saveimage

Synopsis            `function j_saveimage ( obj , filename$ , filetype )`

Beschreibung     Speichert den Inhalt der Komponenten in einem File mit dem Namen **filename**. Es werden zwei Fileformate unterstützt:

- BMP
- PPM

Beispiel

```
:  
if(j_saveimage(canvas,"mandel.bmp",J_BMP) = J_FALSE)  
    print "Error saving Bitmap file"  
:
```

## scrollpane

Synopsis            `function j_scrollpane ( obj )`

Beschreibung        Diese Funktion erzeugt eine neue Scrollpane und gibt dessen Eventnummer zurück. Im Fehlerfall wird `-1` zurückgegeben. Eine Scrollpane ist eine Behälterkomponente, die ein weiteres Objekt aufnehmen kann. Dabei kann das aufgenommene Objekt größere Ausmaße haben als die Scrollpane. Über zwei automatisch angepaßte Scrollbars, kann das enthaltene Object gescrollt werden. Ein Layoutmanager kann nicht in eine Scrollpane integriert werden. Soll mehr als eine Komponente aufgenommen werden, so ist in die Scrollpane zunächst ein Panel zu integrieren. Eine Scrollpane liefert bei einer Größenänderung einen Event.

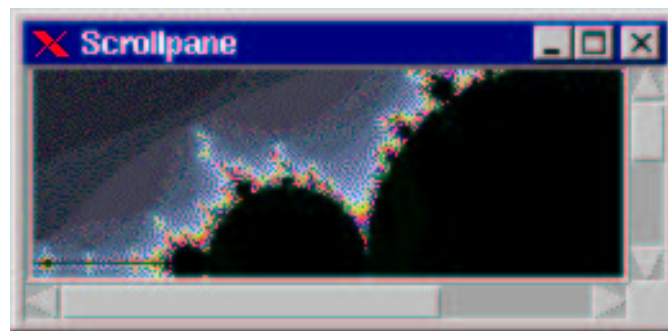
Zielkomponenten    Panel, Borderpanel, Window, Dialog, Frame

Beispiel

```

:
scrollpane = j_scrollpane(frame)
image = j_graphiclabel(scrollpane,"mandel.gif")
j_setsize(scrollpane,240,100)
:

```



selectall
-----------

Synopsis            subroutine **j\_selectall** ( obj )

Beschreibung      Selectiert den gesamten Textinhalt der Komponenten .

Zielkomponenten   Textarea, Textfield

select
--------

Synopsis      function **j\_select** ( obj , item )

Beschreibung      selectiert das Item mit dem Index **item**.

Zielkomponenten      List, Choice

selecttext
------------

Synopsis            subroutine **j\_selecttext** ( obj , start , end )

Beschreibung      Selectiert den Text von Position **start** bis **end**.

Zielkomponenten   Textarea, Textfield



## seperator

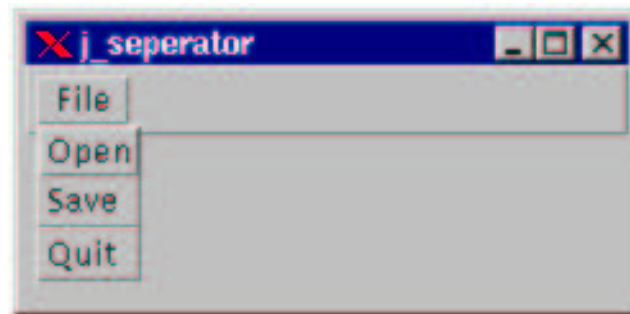
Synopsis            subroutine **j\_seperator**    ( obj )

Beschreibung       Diese Prozedur erzeugt eine Trennlinie in einem Menu, das durch **obj** referenziert wird. Trennlinien dienen ausschließlich der optischen Trennung von Menuitem, und besitzen keine weitere Funktionalität.

Zielkomponenten    Menu, HelpMenu, Popupmenu

Beispiel

```
:  
file = j_menu(menubar,"File")  
open = j_menuitem(file,"Open")  
save = j_menuitem(file,"Save")  
j_seperator(file)  
quit = j_menuitem(file,"Quit")  
:
```



setalign
----------

Synopsis            subroutine **j\_setalign** ( obj , align )

Beschreibung     Setzt die Ausrichtung innerhalb der Komponenten **obj** auf **align**.  
Ein Flowlayout Manager wird vorausgesetzt.

Zielkomponenten   Panel, Borderpanel, Window, Dialog, Frame

setblockinc
-------------

Synopsis	function <b>j_setblockinc</b> ( obj , val )
Beschreibung	Stellt die Blockänderung von der Komponenten auf den Wert von <b>val</b> ein.
Zielkomponenten	Scrollbar

setborderlayout
-----------------

Synopsis            subroutine **j\_setborderlayout** ( obj )

Beschreibung     Integriert einen BorderLayout Manager in die Komponente **obj**.

Zielkomponenten Panel, Borderpanel, Window, Dialog, Frame

setborderpos
--------------

Synopsis	subroutine <b>j_setborderpos</b> ( obj , pos )
Beschreibung	Positioniert die Komponente <b>obj</b> an die angegebene Position <b>pos</b> . Im umgebenden Container muß einen BorderLayout Manager eingestellt sein.
Zielkomponenten	Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphicalabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevensegment

setcolorbg

Synopsis	subroutine <b>j_setcolorbg</b> ( obj , r , g , b )
Beschreibung	Diese Prozedur setzt in dem Komponenten <b>obj</b> die Hintergrundfarbe. Als Parameter sind der <b>rot</b> , <b>grün</b> und <b>blau</b> -Anteil der Farbe zu übergeben. Unabhängig von der tatsächlichen vorhandenen Farbtiefe des Systems wird immer ein 24 Bit Farbwert angenommen. Dieser teilt sich in je 8 Bit für den Rot, den Grün und den Blau-Kanal auf. Somit können diese Parameter Werte zwischen 0 (dunkel) und 255 (hell) annehmen. Das Setzen einer neuen Hintergrundfarbe in einem Canvas bewirkt, das der Inhalt des Cavas überzeichnet wird.
Zielkomponenten	Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphicalabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevenssegment
Beispiel	<pre>: button = j_button(frame,"Hello World") j_setcolorbg(button,150,0,0) j_settext(button,"Hello World") :</pre>



## setcolor

Synopsis	subroutine <b>j_setcolor</b> ( obj , r , g , , b )
Beschreibung	Diese Prozedur setzt in der, durch <b>obj</b> referenzierten, Komponente die Vordergrundfarbe. Als Parameter sind der <b>rot</b> , <b>grün</b> und <b>blau</b> -Anteil der Farbe zu übergeben. Unabhängig von der tatsächlichen vorhandenen Farbtiefe des Systems wird immer ein 24 Bit Farbwert angenommen. Dieser teilt sich in je 8 Bit für den Rot, den Grün und den Blau-Kanal auf. Somit können diese Parameter Werte zwischen 0 (dunkel) und 255 (hell) annehmen. Die Vordergrundfarbe der Komponenten bleibt solange aktiv, bis diese durch einen neuen Aufruf einer Farbfunktion überschrieben wird.
Zielkomponenten	Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphicalabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevenssegment
Beispiel	<pre> : button = j_button(frame,"Hello World") j_setcolor(button,150,0,0) j_settext(button,"Hello World") : </pre>



## setcolumns

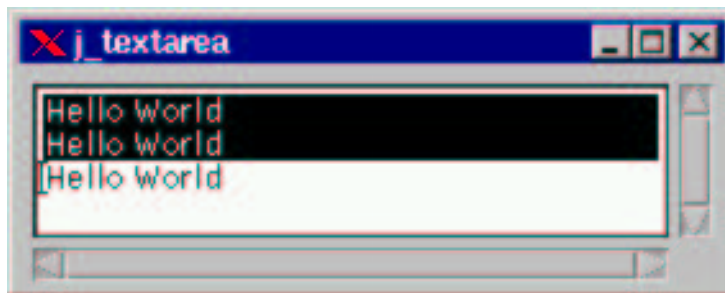
Synopsis            subroutine **j\_setcolumns** ( obj , columns )

Beschreibung      Setzt die Anzahl Spalten von **obj** auf **columns**.

Zielkomponenten   Textarea, Textfield, Gridlayout

Beispiel

```
:  
text = j_text(frame,10,4)  
j_setcolumns(text,30)  
:
```





setcurpos
-----------

- Synopsis      subroutine **j\_setcurpos** ( obj , pos )
- Beschreibung      Setzt den Text Cursor auf die Position **pos**.
- Zielkomponenten      Textarea, Textfield

setcursor
-----------

Synopsis           function **j\_setcursor** ( obj , cursor )

Beschreibung     Setzt den Cursor der Komponenten **obj** auf **cursor**.

Zielkomponenten Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphicalabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevensegment

setdanger
-----------

- Synopsis            subroutine **j\_setdanger** ( obj , val )
- Beschreibung        Stellt den Warnlevel der Komponenten **obj** auf **val**.
- Zielkomponenten    Meter

setdebug

Synopsis            subroutine **j\_setdebug** ( level )

Beschreibung     Diese Prozedur setzt den Debuglevel auf **level**. Ab einem Level größer 0 erscheint ein Debugwindow, in dem die JAPI Calls protokolliert werden. Über den Parameter **level** kann die Protokolltiefe eingestellt werden. **j\_setdebug** ist die einzige Funktion, die aufgerufen werden kann, bevor **j\_start()** aufgerufen wurde. Die Levelbenen im einzelnen:

- 0:** keine Ausgabe (default Wert)
- 1:** Rückmeldung der konstruktiven Funktionen. Nur das Erzeugen der graphischen Objekte wird protokolliert.
- 2:** Wie 1, zusätzliche Ausgabe aller Aktionen, die vom Benutzer ausgeführt werden.
- 3:** Wie 2, zusätzlich werden alle weiteren Funktionen (außer den graphischen Befehlen) protokolliert.
- 4:** Wie 3, zusätzlich mit allen graphischen Befehlen.

setechochar
-------------

Synopsis	subroutine <b>j_setechochar</b> ( obj , chr )
Beschreibung	Bestimmt das Zeichen <b>chr</b> als Echo Zeichen für alle Benutzereingaben in die Komponente .
Zielkomponenten	Textfield

seteditable
-------------

Synopsis            subroutine **j\_seteditable** ( obj , bool )

Beschreibung     Ist **bool** J\_TRUE , so ist die Komponente editierbar, ansonsten  
ist die Komponente nur lesbar für den Benutzer.

Zielkomponenten   Textarea, Textfield

setfixlayout
--------------

Synopsis	subroutine <b>j_setfixlayout</b> ( obj )
Beschreibung	Integriert einen Fixlayout Manager in die Komponente <b>obj</b> (voreingestellte Layoutmanager).
Zielkomponenten	Panel, Borderpanel, Window, Dialog, Frame

setflowfill
-------------

Synopsis            subroutine **j\_setflowfill** ( obj , bool )

Beschreibung     Setzt die Höhe (Breite) aller enthaltenen Komponenten auf die Höhe (Breite) von der Komponenten **obj**. Ein Flowlayout Manager wird vorausgesetzt.

Zielkomponenten   Panel, Borderpanel, Window, Dialog, Frame



setflowlayout
---------------

Synopsis	subroutine <b>j_setflowlayout</b> ( obj , align )
Beschreibung	Integriert einen Flowlayout Manager in die Komponente <b>obj</b> mit der Ausrichtung <b>align</b> .
Zielkomponenten	Panel, Borderpanel, Window, Dialog, Frame

setfocus
----------

Synopsis           function **j\_setfocus** ( obj )

Beschreibung     Weist den Input Focus der Komponenten **obj** zu.

Zielkomponenten Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphicalabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevensegment

## setfontname

- Synopsis**            subroutine **j\_setfontname**    ( obj , name )
- Beschreibung**        Diese Prozedur setzt den Fonttyp in der Komponenten **obj**, auf **name**.  
Der Parameter **name** kann folgende Werte annehmen:
- J\_COURIER : Courier (Monospaced)
  - J\_HELVETIA : Helvetia (Sans Serif)
  - J\_TIMES : Times (Serif)
  - J\_DIALOGIN : Dialog In
  - J\_DIALOGOUT : Dialog Out
- Zielkomponenten**    Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphiclabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevenssegment, MenuItem, CheckBoxMenuItem, Menu, HelpMenu, Popupmenu
- Beispiel**
- ```

:
label = j_label(jframe,"Hello World")
j_setfontname(label,J_HELVETIA)
:

```



setfontsize

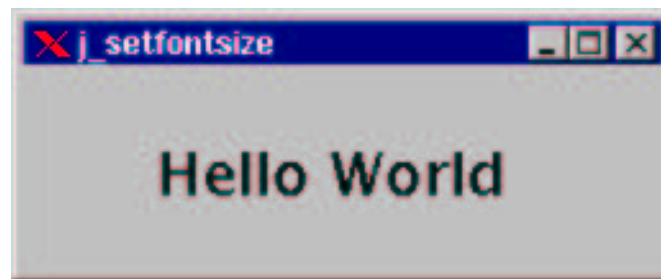
Synopsis            subroutine **j\_setfontsize** ( obj , size )

Beschreibung     Diese Prozedur setzt die Fontgröße in der Komponenten, **obj**, auf **height**. Die Größe **size**, kann prinzipiell beliebige positive Zahlen annehmen. Die Größe des Fonts hat die Einheit Pixel.

Zielkomponenten Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphiclabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevenssegment, MenuItem, CheckBoxMenuItem, Menu, HelpMenu, Popupmenu

Beispiel

```
:  
label = j_label(jframe,"Hello World")  
j_setfontsize(label,24)  
:
```



## setfontstyle

|                 |                                                                                                                                                                                                                                                                                                                                                                  |
|-----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Synopsis        | subroutine <b>j_setfontstyle</b> ( obj , style )                                                                                                                                                                                                                                                                                                                 |
| Beschreibung    | <p>Diese Prozedur setzt den Fontstyle in der Komponenten, die durch <b>obj</b> referenziert wird, auf <b>style</b>.</p> <p>Für den Parameter <b>style</b> sind folgende Werte möglich:</p> <ul style="list-style-type: none"> <li>• J_PLAIN : normal</li> <li>• J_BOLD : fett</li> <li>• J_ITALIC : italic</li> <li>• J_BOLD + J_ITALIC : italic fett</li> </ul> |
| Zielkomponenten | <p>Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphiclabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevensegment, MenuItem, CheckBoxMenuItem, Menu, HelpMenu, Popumenu</p>                                                                               |
| Beispiel        | <pre> : label = j_label(jframe,"Hello World") j_setfontstyle(label,J_BOLD+J_ITALIC) : </pre>                                                                                                                                                                                                                                                                     |



|         |
|---------|
| setfont |
|---------|

|                 |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|-----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Synopsis        | subroutine <b>j_setfont</b> ( obj , name , style , size )                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| Beschreibung    | <p>Diese Prozedur setzt erzeugt einen neuen Font in der Komponente, die durch <b>obj</b> referenziert wird. Der Font wird durch die folgenden drei Parameter spezifiziert:</p> <p>Der Parameter <b>name</b> kann folgende Werte annehmen:</p> <ul style="list-style-type: none"> <li>• J_COURIER : Courier (Monospaced)</li> <li>• J_HELVETIA : Helvetia (Sans Serif)</li> <li>• J_TIMES : Times (Serif)</li> <li>• J_DIALOGIN : Dialog In</li> <li>• J_DIALOGOUT : Dialog Out</li> </ul> <p>Für den Parameter <b>style</b> sind folgende Werte möglich:</p> <ul style="list-style-type: none"> <li>• J_PLAIN : normal</li> <li>• J_BOLD : fett</li> <li>• J_ITALIC : italic</li> <li>• J_BOLD + J_ITALIC : italic fett</li> </ul> <p>Die Größe <b>size</b> wird als Integerzahl übergeben. Sie kann prinzipiell beliebige positive Zahlen annehmen.</p> |
| Zielkomponenten | Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphiclabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevenssegment, MenuItem, CheckBoxMenuItem, Menu, HelpMenu, Popumenu                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| Beispiel        | <pre> : label = j_label(jframe,"Hello World") j_setfont(label,J_TIMES,J_PLAIN,18) : </pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |



|               |
|---------------|
| setgridlayout |
|---------------|

|                 |                                                                                                                |
|-----------------|----------------------------------------------------------------------------------------------------------------|
| Synopsis        | subroutine <b>j_setgridlayout</b> ( obj , row , col )                                                          |
| Beschreibung    | Integriert einen Gridlayout Manager in die Komponente <b>obj</b> mit <b>row</b> Reihen und <b>col</b> Spalten. |
| Zielkomponenten | Panel, Borderpanel, Window, Dialog, Frame                                                                      |



## sethgap

Synopsis            subroutine **j\_sethgap**    ( obj , hgap )

Beschreibung        Setzt den horizontalen Abstand innerhalb der, durch **obj** referenzierten, Komponenten auf **hgap** Pixel. Die Komponente muss vom Typ Container sein (siehe Zielkomponenten). Weiterhin sollte in der Komponente ein Layoutmanager definiert sein, da diese Funktion nur in Verbindung mit einem Layoutmanager Sinn macht. Der Layoutmanager ordnet nun alle Objekte so an, daß deren horizontaler Abstand zueinander genau **hgap** Pixel beträgt.

Zielkomponenten    Panel, Borderpanel, Window, Dialog, Frame

Beispiel

```
:  
j_flowlayout(frame,J_HORIZONTAL)  
button1 = j_button(frame,"Button1")  
button2 = j_button(frame,"Button2")  
j_sethgap(frame,30)  
:
```



|         |
|---------|
| seticon |
|---------|

Synopsis            subroutine **j\_seticon** ( frame , icon )

Beschreibung      Ordnet dem **frame** das Icon **icon** zu. Eine iconisierte Darstellung wird nicht von allen Plattformen unterstützt.

Zielkomponenten   Frame

Beispiel

```
:  
frame = j_frame("Hello World")  
j_seticon(frame,j_loadimage("icon.gif"))  
:
```

## setimage

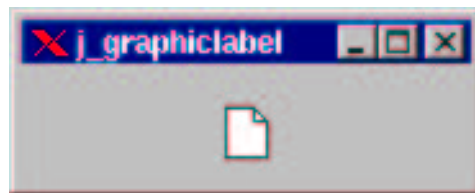
Synopsis            subroutine **j\_setimage** ( obj , image )

Beschreibung      Ordnet dem **obj** das Bild **image** zu.

Zielkomponenten   Graphicbutton, Graphiclabel

Beispiel

```
:  
label = j_graphiclabel(frame,"mandel.gif")  
image = j_image("new.gif")  
j_setimage(label,image)  
:
```



## setinsets

Synopsis            subroutine **j\_setinsets** ( obj , top , bottom , left , right )

Beschreibung      Setzt den inneren Rahmen auf die angegebenen Größen.

Zielkomponenten   Panel, Borderpanel, Window, Dialog, Frame

Beispiel

```
:  
frame = j_frame("j_setinsets")  
j_setinsets(frame,30,10,10,10)  
:  
  
> 25 5 5 6
```



|        |
|--------|
| setmax |
|--------|

|                 |                                                                          |
|-----------------|--------------------------------------------------------------------------|
| Synopsis        | function <b>j_setmax</b> ( obj , val )                                   |
| Beschreibung    | Stellt die obere Grenze der Komponenten auf den Wert von <b>val</b> ein. |
| Zielkomponenten | Scrollbar, Meter, Progress                                               |

|        |
|--------|
| setmin |
|--------|

Synopsis      function **j\_setmin** ( obj , val )

Beschreibung      Stellt die untere Grenze der Komponenten auf den Wert von **val** ein.

Zielkomponenten      Scrollbar, Meter, Progress

|                 |
|-----------------|
| setnamedcolorbg |
|-----------------|

|                 |                                                                                                                                                                                                                         |
|-----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Synopsis        | subroutine <b>j_setnamedcolorbg</b> ( obj , color )                                                                                                                                                                     |
| Beschreibung    | Setzt die Hintergrundfarbe auf eine vordefinierte Farbe <b>color</b> .                                                                                                                                                  |
| Zielkomponenten | Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphiclabeled, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevensegment |

|               |
|---------------|
| setnamedcolor |
|---------------|

|                 |                                                                                                                                                                                                                       |
|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Synopsis        | subroutine <b>j_setnamedcolor</b> ( obj , color )                                                                                                                                                                     |
| Beschreibung    | Setzt die Vordergrundfarbe auf eine vordefinierte Farbe <b>color</b> .                                                                                                                                                |
| Zielkomponenten | Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphiclabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevensegment |



|             |
|-------------|
| setnolayout |
|-------------|

|                 |                                                                        |
|-----------------|------------------------------------------------------------------------|
| Synopsis        | subroutine <b>j_setnolayout</b> ( obj )                                |
| Beschreibung    | Entfernt den aktuellen Layout Manager aus der Komponenten <b>obj</b> . |
| Zielkomponenten | Panel, Borderpanel, Window, Dialog, Frame                              |

|       |
|-------|
| start |
|-------|

Synopsis            subroutine **j\_start** ( port )

Beschreibung     Diese Funktions ersetzt den default Port durch den übergebenen Parameter. Damit kann mehr als ein Kernel auf einem Rechner gestartet werden. Diese Funktion muss vor j\_start() aufgerufen werden.

Beispiel

```
:  
j_setport(12345)  
if(j_start() = J_FALSE) then  
:
```

|        |
|--------|
| setpos |
|--------|

|                 |                                                                                                                                                                                                                        |
|-----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Synopsis        | subroutine <b>j_setpos</b> ( obj , xpos , ypos )                                                                                                                                                                       |
| Beschreibung    | Verschiebt die Komponente <b>obj</b> zur angegebenen Koordinate ( <b>xpos,ypos</b> ) an.                                                                                                                               |
| Zielkomponenten | Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphicalabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevensegment |

|               |
|---------------|
| setradiogroup |
|---------------|

Synopsis      function **j\_setradiogroup** ( rbutton, , rgroup )

Beschreibung      Weist dem Radiobutton **rbutton** die Radiogroup **rgroup** zu. Der Radiobutton wird aus der alten Radiogroup entfernt.

Zielkomponenten      Radiobutton

|              |
|--------------|
| setresizable |
|--------------|

- Synopsis            subroutine **j\_setresizable** ( obj , resizable )
- Beschreibung        Ist **resizable** J\_FALSE , kann die Größe der Komponenten nicht vom Benutzer verändert werden.
- Zielkomponenten    Dialog, Frame
- Beispiel
- ```
          :  
          frame = j_frame("fixsized Frame")  
          j_setrezisable(frame,J_FALSE)  
          :
```

setrows

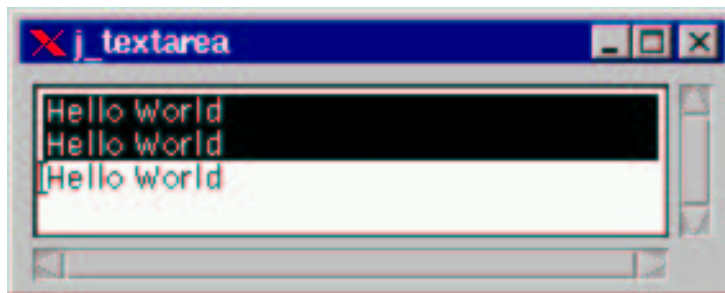
Synopsis            subroutine **j\_setrows**    ( obj , rows )

Beschreibung     Setzt die Anzahl Reihen von **obj** auf **rows**.

Zielkomponenten   Textarea, Gridlayout

Beispiel

```
:  
text = j_text(frame,30,10)  
j_setcolumns(text,4)  
:
```



setshortcut
-------------

Synopsis            subroutine **j\_setshortcut** ( obj , chr )

Beschreibung      Bestimmt das Zeichen **chr** als Shortcut für die Komponente .

Zielkomponenten   Menuitem, CheckBoxMenuItem, Menu, HelpMenu, Popupmenu

setsize

Synopsis	subroutine <b>j_setsize</b> ( obj , width , height )
Beschreibung	Mit dieser Prozedur wird einer Komponente, die durch <b>obj</b> referenziert wird, eine neue Größe zugewiesen. Die Komponente erhält eine Breite von <b>width</b> Pixeln und eine Höhe von <b>height</b> Pixeln zugewiesen. Obwohl diese Prozedur prinzipiell auf alle Komponenten anwendbar ist, hat sie bei einigen Komponenten keine Auswirkung. So hat eine Choice beispielweise eine unveränderliche Höhe. Diese Höhe wird durch die Peer Komponente der jeweiligen Oberfläche vorgegeben. Eine weitere Ausnahme stellen die beiden Text Komponenten dar. Bei diesen Komponenten erfolgt die Größenangabe nicht in Bildschirmpunkten, sondern in der Anzahl der sichtbaren Textzeichen.
Zielkomponenten	Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphiclabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevensegment
Beispiel	<pre>: button = j_button(frame,"Button") j_setsize(button,100,100) :</pre>





setslidesize
--------------

Synopsis            `function j_setslidesize ( obj , val )`

Beschreibung      Setzt die Größe des Schiebers auf den Wert von **val**.

Zielkomponenten   Scrollbar

setstate
----------

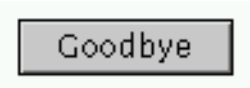
Synopsis            subroutine **j\_setstate** ( obj , bool )

Beschreibung     Ist **bool J\_TRUE** so wird die Komponente selectiert.

Zielkomponenten    Checkbox, Radiobutton, CheckmenuItem, Led

settext
---------

Synopsis	subroutine <b>j_settext</b> ( obj , str\$ )
Beschreibung	Setzt den Textinhalt in der, durch <b>obj</b> referenzierten, Komponenten auf den Inhalt des Parameter <b>str</b> . Ist die Komponente vom Typ Frame oder Dialog, so wird der übergebene Text im Rahmen des Fensters eingetragen. Ansonsten wird der Textinhalt der Komponenten gesetzt. Der übergebene Text sollte keine Zeilenumbrüche enthalten, da fast alle Komponenten nur einzeilige Texte darstellen können. Einzige Ausnahme ist die Textarea Komponente, die mehrere Zeilen darstellen kann. Wird an diese Komponente ein Text übergeben, so darf dieser natürlich auch Zeilenumbrüche enthalten.
Zielkomponenten	Button, Label, Checkbox, Radiobutton, Dialog, Frame, MenuItem, CheckBoxMenuItem, Menu, HelpMenu, Popupmenu, Textarea, Textfield
Beispiel	<pre> : button = j_button(frame,"Hello World") j_settext(button,"Goodbye") : </pre>



setunitinc
------------

Synopsis      `function j_setunitinc ( obj , val )`

Beschreibung      Stellt die minimale Wertänderung von die Komponente auf den Wert von **val** ein.

Zielkomponenten      Scrollbar

setvalue
----------

Synopsis            subroutine **j\_setvalue** ( obj , val )

Beschreibung        Stellt den aktuellen Wert der Komponenten **obj** auf **val**.

Zielkomponenten    Scrollbar, Progress, Meter, Sevensegment

setvgap

Synopsis            subroutine **j\_setvgap**    ( obj , vgap )

Beschreibung        Setzt den vertikalen Abstand innerhalb der, durch **obj** referenzierten, Komponenten auf **vgap** Pixel. Die Komponente muss vom Typ Container sein (siehe Zielkomponenten). Weiterhin sollte in der Komponente ein Layoutmanager definiert sein, da diese Funktion nur in Verbindung mit einem Layoutmanager Sinn macht. Der Layoutmanager ordnet nun alle Objekte so an, daß deren vertikaler Abstand zueinander genau **vgap** Pixel beträgt.

Zielkomponenten    Panel, Borderpanel, Window, Dialog, Frame

Beispiel

```
:  
j_setflowlayout(frame,J_VERTICAL)  
button1 = j_button(frame,"Button1")  
button2 = j_button(frame,"Button2")  
j_setvgap(frame,30)  
:
```



setxor
--------

Synopsis	subroutine <b>j_setxor</b> ( obj , bool )
Beschreibung	Schaltet den Painting Mode auf XOR, falls bool = J_TRUE ist. Im XOR Modus heben sich zwei gleiche Graphikbefehle auf.
Zielkomponenten	Canvas, Image, Printer

## sevensegment

Synopsis      function **j\_sevensegment** ( obj , color )

Beschreibung      Die Funktion erzeugt eine Siebensegment-Anzeige und liefert die Eventnummer der Komponente zurück. Die Farbe der Siebensegment-Anzeige kann eine der vordefinierten Farben sein (zB. J\_RED, J\_GREEN).

Zielkomponenten      Panel, Borderpanel, Window, Dialog, Frame

Beispiel

```
:  
seven = j_sevensegment(frame,J_GREEN)  
j_setvalue(seven,5)  
:
```





showpopup
-----------

- Synopsis            subroutine **j\_showpopup** ( obj , xpos , ypos )
- Beschreibung        Zeigt die Komponente an der Koordinate (**xpos,ypos**) an.
- Zielkomponenten    Popupmenu

show
------

Synopsis            subroutine **j\_show** ( obj )

Beschreibung     Zeigt die Komponente **obj** auf dem Bildschirm an.

Zielkomponenten Button, Graphicbutton, Canvas, Checkbox, Radiobutton, Choice, Label, Graphicalabel, List, Scrollbar, Panel, Borderpanel, Window, Dialog, Frame, Scrollpane, Textarea, Textfield, Led, Progressbar, Meter, Sevensegment

sleep
-------

Synopsis      function **j\_sleep** ( msec )

Beschreibung      Unterbricht die Ausführung der Applikation für **msec** Millisekunden.

start
-------

Synopsis            `function j_start ( )`

Beschreibung        Diese Funktion stellt eine Verbindung zu einem JAPI Kernel her, sofern auf der lokalen Maschine bereits ein Kernel läuft. Kann kein Kernel gefunden werden, so wird ein neuer Kernel gestartet und eine Verbindung aufgebaut. Schlägt dies auch fehl, so wird als Fehlercode `J_FALSE` zurückgegeben

Beispiel

```
:  
if(j_start() = J_FALSE) then  
  print "can't connect to JAPI Kernel"  
  exit  
end if  
:
```

sync
------

Synopsis        subroutine **j\_sync** ( )

Beschreibung    Synchronisiert die Applikation mit dem JAPI Kernel.

textarea

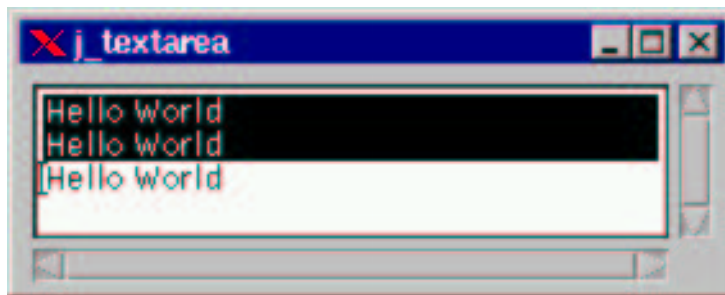
Synopsis            function **j\_textarea** ( obj , rows , columns )

Beschreibung     Diese Funktion erzeugt eine neue Textarea und gibt dessen Eventnummer zurück. Im Fehlerfall wird `-1` zurückgegeben. Die Textarea hat eine sichtbare Größe von **rows** Reihen und **columns** Spalten, und ist somit abhängig vom voreingestellten Font. Soll eine Größe in Pixeleinheiten eingestellt werden, so kann nachträglich die Funktion *j\_setsize()* verwendet werden. Die Textarea besitzt voreingestellt zwei Scrollbars, die sich automatisch anpassen. Eine Textarea liefert einen Event, sobald sich der Textinhalt ändert.

Zielkomponenten   Panel, Borderpanel, Window, Dialog, Frame

Beispiel

```
:  
frame = j_frame("j_textarea")  
text  = j_textarea(frame,30,4)  
:
```



## textfield

Synopsis            `function j_textfield ( obj , columns )`

Beschreibung     Diese Funktion erzeugt ein neues Textfield und gibt dessen Eventnummer zurück. Im Fehlerfall wird `-1` zurückgegeben. Ein Textfield dient für alle einzeiligen Eingaben. Das Textfield hat eine sichtbare Größe von genau einer Reihe und **columns** Spalten, und ist somit abhängig vom voreingestellten Font. Soll eine Größe in Pixeleinheiten eingestellt werden, so kann nachträglich die Funktion `j_setsize()` verwendet werden.  
Ein Textfield liefert einen Event, sobald die Eingabe mit **Return** abgeschlossen wird.

Zielkomponenten   Panel, Borderpanel, Window, Dialog, Frame

Beispiel

```

:
frame = j_frame("j_textfield")
text  = j_textfield(frame,30)
:

```



translate
-----------

Synopsis      subroutine **j\_translate** ( obj , x , y )

Beschreibung      Setzt den Ursprung des Koordinatensystems auf (**x**, **y**).

Zielkomponenten      Canvas, Image, Printer



## vscrollbar

Synopsis            `function j_vscrollbar ( obj )`

Beschreibung      Diese Funktion erzeugt eine vertikale Scrollbar und gibt dessen Eventnummer zurück. Im Fehlerfall wird `-1` zurückgegeben. Ist die Zielkomponente **obj** vom Typ `Scrollpane`, so wird die Eventnummer der vorhandenen vertikalen Scrollbar zurückgegeben. Eine Scrollbar erzeugt einen Event, sobald der Schieber bewegt wird.

Zielkomponenten   `Panel, Borderpanel, Window, Dialog, Frame, Scrollpane`

Beispiel

```

:
scroll=j_vscrollbar(frame)
j_setpos(scroll,120,40)
j_setsize(scroll,20,100)
:

```



windowlistener

Synopsis	function <b>j_windowlistener</b> ( window , kind )
Beschreibung	<p>Bindet an die Komponente <b>window</b> einen neuen Windowlistener, und liefert dessen Eventnummer zurück. Der Parameter <b>kind</b> bestimmt, wann ein Event ausgelöst wird. Zulässige Werte für <b>kind</b> sind:</p> <ul style="list-style-type: none"><li>• <b>J_ACTIVATED</b> : Der Componentlistener liefert einen Event, wenn die Komponente <b>obj</b> activiert wird.</li><li>• <b>J_DEACTIVATED</b> : Der Componentlistener liefert einen Event, wenn die Komponente <b>obj</b> deaktiviert wird.</li><li>• <b>J_OPENED</b> : Der Componentlistener liefert einen Event, wenn die Komponente <b>obj</b> geöffnet ist.</li><li>• <b>J_CLOSED</b> : Der Componentlistener liefert einen Event, wenn die Komponente <b>obj</b> geschlossen ist.</li><li>• <b>J_ICONFIED</b> : Der Componentlistener liefert einen Event, wenn die Komponente <b>obj</b> iconifiziert wird.</li><li>• <b>J_DEICONFIED</b> : Der Componentlistener liefert einen Event, wenn die Komponente <b>obj</b> deiconifiziert wird.</li><li>• <b>J_CLOSING</b> : Der Componentlistener liefert einen Event, wenn das close icon der Komponenten <b>obj</b> angeklickt wird.</li></ul>
Zielkomponenten	Window, Dialog, Frame

## window

Synopsis            `function j_window ( obj )`

Beschreibung       Dies Funktion erzeugt ein neues Window und liefert dessen Eventnummer zurück. Im Fehlerfall wird `-1` zurückgeliefert. Ein Window ist ein einfaches eigenständiges Fenster, das keinen Fensterahmen besitzt. Wie ein Dialog, kann ein Window keine Menüleiste besitzen. Als Zielobjekt dieser Funktion ist nur ein Frame zulässig. Ein Window wird erst angezeigt, wenn ein entsprechendes `j_show()` Kommando ausgeführt wurde. Ein Window liefert keinen Event.

Zielkomponenten    Frame

Beispiel

```
:  
window = j_window(frame)  
label = j_label(window,"Mouse pressed at ... ")  
j_setnamedcolorbg(label,J_YELLOW)  
:
```

```
Mouse pressed at 108:179
```