



Dokumentation der SuM-Komponenten

[Komponente](#)

[Textkomponente](#)

[Knopf](#)

[Etikett](#)

[Textfeld](#)

[Kennwortfeld](#)

[Textbereich](#)

[Zeichenbereich](#)

[Zeilenbereich](#)



Dokumentation der Klasse "Komponente"

Klassenname	Komponente
Beschreibung	Die Klasse Komponente ist eine abstrakte Oberklasse für alle Komponenten der SuM-Komponentenbibliothek. Jede einer solchen Komponente hat eine Position, die durch die linke obere Ecke der Komponente angegeben ist. Neben der Größe kann auch die Sichtbarkeit und die Aktivität gesteuert werden. Mit Hilfe der Tabulatortaste und der Maus kann der Fokus auf die jeweilige Komponente gesetzt werden.
Protokoll	<p>Auftrag init(pLinks, pOben, pBreite, pHoehe : Zahl) <i>nachher</i> Die Komponente befindet sich mit der angegebenen Größe und der angegebenen Position. Sie ist sichtbar und aktiv.</p> <p>Auftrag setzePosition(pLinks, pOben : Zahl) <i>nachher</i> Die Komponenten wurde von ihrer aktuellen Position gelöscht und an der angegebenen Position neu dargestellt.</p> <p>Auftrag setzeGroesse(pBreite, pHoehe : Zahl) <i>nachher</i> Die Komponenten wurde in der neuen Größe dargestellt.</p> <p>Anfrage links <i>nachher</i> Diese Anfrage liefert die horizontale Koordinate des linken Randes der Komponente.</p> <p>Anfrage oben <i>nachher</i> Diese Anfrage liefert die vertikale Koordinate des oberen Randes der Komponente.</p> <p>Anfrage breite <i>nachher</i> Diese Anfrage liefert die Breite der Komponente.</p> <p>Anfrage hoehe <i>nachher</i> Diese Anfrage liefert die Höhe der Komponente.</p> <p>Auftrag verstecke <i>vorher</i> Die Komponente ist sichtbar. <i>nachher</i> Die Komponente ist nun unsichtbar auf dem Bildschirm dargestellt.</p> <p>Auftrag zeige <i>vorher</i> Die Komponente ist nicht sichtbar. <i>nachher</i> Die Komponente ist nun sichtbar auf dem Bildschirm dargestellt.</p> <p>Anfrage istSichtbar : Wahrheitswert <i>nachher</i> Diese Anfrage liefert "wahr", genau dann wenn die Komponente sichtbar ist.</p> <p>Auftrag deaktiviere <i>vorher</i> Die Komponente ist aktiviert. <i>nachher</i> Die Komponente ist nun nicht mehr aktiv, reagiert also auf keine Ereignisse mehr. Je nach konkreter Komponente wird sie anders dargestellt.</p> <p>Auftrag aktiviere <i>vorher</i> Die Komponente ist nicht aktiv.</p>



nachher Die Komponente ist nun aktiv, d.h. reagiert auf Ereignisse. Sie wird auch wieder normal dargestellt.

Anfrage istAktiv : Wahrheitswert

nachher Diese Anfrage liefert "wahr", genau dann wenn die Komponente aktiv ist.

Auftrag setzeFokus

nachher Die Komponente besitzt nun den Fokus.

Anfrage besitztFokus : Wahrheitswert

nachher Diese Anfrage liefert "wahr", genau dann wenn die Komponente den Fokus hat.

Auftrag gibFrei

nachher Die Komponente befindet sich nicht mehr auf dem Bildschirm und steht nicht mehr zur Verfügung.

Ereignis fokusErhalten

vorher Die Komponente erhält vom Anwender oder dem Programm den Fokus.

Ereignis fokusVerloren

vorher Eine andere Komponente erhält vom Anwender oder dem Programm den Fokus.



Dokumentation der Klasse "Textkomponente"

Klassenname	Textkomponente
Oberklasse	Komponente
Beschreibung	<p>Die Klasse Textkomponente ist eine abstrakte Oberklasse für alle textorientierten Komponenten der SuM-Komponentenbibliothek. Neben den Diensten und Eigenschaften, die von der Oberklasse geerbt werden, besitzt eine Textkomponente einen Inhalt, der in verschiedenen Datentypen abgefragt oder gesetzt werden kann. Standardmäßig ist der Inhalt leer.</p> <p>Das Aussehen des Inhalts kann in Schriftart, Schriftgröße, Schriftfarbe und Schriftstil verändert werden. Die Voreinstellung hängt hier von der verwendeten Programmierungsumgebung ab.</p>
Protokoll	<p>Auftrag <code>init(pLinks, pOben, pBreite, pHoehe : Zahl; pInhalt: Zeichenkette)</code> <i>nachher</i> Die Komponente befindet sich mit der angegebenen Größe, der Position und dem Inhalt auf dem Bildschirm. Sie ist sichtbar und aktiv.</p> <p>Auftrag <code>setzeInhalt(pInhalt : Zeichenkette)</code> Auftrag <code>setzeInhalt(pInhalt : Zahl)</code> Auftrag <code>setzeInhalt(pInhalt : GanzeZahl)</code> <i>nachher</i> Der Inhalt der Textkomponente wurde entsprechend gesetzt. Der Parameter kann eine Zeichenkette, eine Zahl oder eine GanzeZahl sein.</p> <p>Anfrage <code>inhaltIstText : Wahrheitswert</code> <i>nachher</i> Diese Anfrage ist wahr, wenn der Inhalt als Text zurückgegeben werden kann..</p> <p>Anfrage <code>inhaltIstZahl : Wahrheitswert</code> <i>nachher</i> Diese Anfrage ist wahr, wenn der Inhalt als Zahl zurückgegeben werden kann.</p> <p>Anfrage <code>inhaltIstGanzeZahl : Wahrheitswert</code> <i>nachher</i> Diese Anfrage ist wahr, wenn der Inhalt als GanzeZahl zurückgegeben werden kann.</p> <p>Anfrage <code>inhaltAlsText : Zeichenkette</code> <i>nachher</i> Diese Anfrage liefert den Inhalt der Textkomponente als Zeichenkette.</p> <p>Anfrage <code>inhaltAlsZahl : Zahl</code> <i>vorher</i> Der Inhalt ist eine Zahl. <i>Nachher</i> Diese Anfrage liefert den Inhalt der Textkomponente als Zahl.</p> <p>Anfrage <code>inhaltAlsGanzeZahl : GanzeZahl</code> <i>vorher</i> Der Inhalt ist eine GanzeZahl. <i>Nachher</i> Diese Anfrage liefert den Inhalt der Textkomponente als GanzeZahl.</p> <p>Auftrag <code>setzeSchriftArt (pArt : Zeichenkette)</code> <i>nachher</i> Der Inhalt wurde in der angegebenen Schriftart dargestellt.</p> <p>Auftrag <code>setzeSchriftGroesse (pGroesse : GanzeZahl)</code> <i>nachher</i> Der Inhalt wurde in der angegebenen Schriftgröße dargestellt.</p> <p>Auftrag <code>setzeSchriftFarbe (pFarbe : GanzeZahl)</code> <i>nachher</i> Der Inhalt wurde in der angegebenen Schriftfarbe dargestellt.</p> <p>Auftrag <code>setzeSchriftStil (pStil : GanzeZahl)</code> <i>nachher</i> Der Inhalt wurde mit dem angegebenen Schriftstil dargestellt.</p>



Dokumentation der Klasse "Knopf"

Klassenname	Knopf
Oberklasse	Textkomponente
Beschreibung	Ein Knopf ist eine Textkomponente, die vom Anwender gedrückt werden kann.



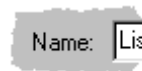
Protokoll	Ereignis geklickt <i>vorher</i> Der Benutzer hat den Knopf entweder mit der Maus auf den Knopf geklickt oder die Eingabetaste (unter erhaltenem Fokus) gedrückt.
-----------	--



Dokumentation der Klasse "Etikett"

Klassenname	Etikett
Oberklasse	Textkomponente
Beschreibung	Ein Etikett ist eine Textkomponente mit der vom Anwender nicht veränderbare Texte auf dem Bildschirm dargestellt werden können. Sie können zur Beschriftung anderer sichtbarer Komponenten eingesetzt werden.

Zusätzlich zu den Eigenschaften der Textkomponente kann die Schrift des Etiketts auch ausgerichtet werden. Dafür stehen die Konstanten RECHTSBUENDIG, LINKSBUENDIG, ZENTRIERT zur Verfügung.



Protokoll	<p>Ein Etikett reagiert nicht auf die Zuweisung des Fokus.</p> <p>Auftrag <code>setzeAusrichtung(pAusrichtung : GanzeZahl)</code></p> <p><i>nachher</i> Das Etikett wird nun mit der angegebenen Ausrichtung dargestellt.</p>
-----------	--



Dokumentation der Klasse "Textfeld"

Klassenname	Textfeld
Oberklasse	Textkomponente
Beschreibung	Ein Textfeld ist eine Textkomponente zur Ein- und Ausgabe einzelner Texte. Der Anwender kann es mit Hilfe der Maus oder der Tabulatortaste fokussieren.



Protokoll	<p>Ereignis inhaltGeaendert <i>vorher</i> Der Inhalt des Textfeldes wurde vom Benutzer verändert.</p> <p>Ereignis eingabeBestaetigt <i>vorher</i> Der Benutzer hat die Eingabetaste gedrückt.</p>
-----------	---



Dokumentation der Klasse "Kennwortfeld"

Klassenname	Kennwortfeld
Oberklasse	Textfeld
Beschreibung	Ein Textfeld ist eine Textkomponente zur Eingabe von Kennwörtern. Die Eingabe wird dabei durch ein einstellbares Kennwortzeichen verdeckt.



Protokoll	<p>Auftrag setzeKennwortzeichen(pKennwortzeichen : Zeichen)</p> <p><i>nachher</i> Die Eingabe wurde mit dem angegebenen Kennwortzeichen verdeckt.</p>
-----------	--



Dokumentation der Klasse "Textbereich"

Klassenname	Textbereich
Oberklasse	Textkomponente
Beschreibung	Ein Textbereich ist eine abstrakte Oberklasse für mehrzeilige Textfelder. Je nach Anwendungszweck kann auf eine zeilen- oder zeichenorientierte Unterklasse zugegriffen werden.

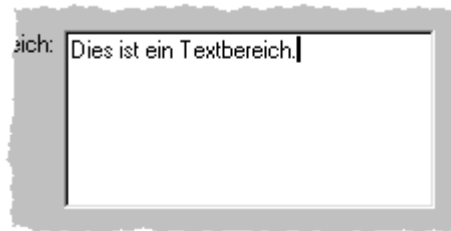


Protokoll	<p>Anfrage istMarkiert : Wahrheitswert <i>nachher</i> Der Inhalt des Bereichs oder ein Teil davon ist markiert.</p> <p>Anfrage markierungsAnfang : GanzeZahl <i>nachher</i> Diese Anfrage liefert den Anfang des markierten Bereichs.</p> <p>Anfrage markierungsEnde : GanzeZahl <i>nachher</i> Diese Anfrage liefert das Ende des markierten Bereichs.</p> <p>Anfrage markierterInhalt : Zeichenkette <i>nachher</i> Diese Anfrage liefert den Inhalt des markierten Bereichs.</p> <p>Auftrag setzeMarkierung(pAnfang, pEnde : GanzeZahl) <i>nachher</i> Die Markierung des Bereichs wurde gemäß der angegebenen Parameter neu gesetzt.</p> <p>Auftrag markiereAlles <i>nachher</i> Der gesamte Inhalt des Bereichs wurde markiert.</p> <p>Auftrag markiereNichts <i>nachher</i> Im Bereich ist nun nichts mehr markiert.</p> <p>Auftrag löscheMarkierung <i>nachher</i> Der markierte Bereich wurde gelöscht.</p> <p>Auftrag loescheAlles <i>nachher</i> Der gesamte Bereich wurde gelöscht.</p> <p>Anfrage teillInhalt(pAnfang, pEnde : GanzeZahl) <i>nachher</i> Diese Anfrage liefert den Inhalt innerhalb der angegebenen Grenzen.</p> <p>Auftrag fuegeEin(pText : Zeichenkette, pStelle : GanzeZahl) <i>nachher</i> Der angegebene Text wurde an die angegebene Stelle eingefügt.</p> <p>Auftrag haengeAn(pText : Zeichenkette) <i>nachher</i> Der angegebene Text wurde an das Ende des Textbereichs angehängt.</p> <p>Ereignis inhaltGeaendert <i>vorher</i> Der Inhalt des Bereichs wurde vom Benutzer geändert.</p> <p>Ereignis markierungGeändert <i>vorher</i> Die Markierung des Bereichs wurde vom Benutzer geändert.</p>
-----------	--



Dokumentation der Klasse "Zeichenbereich"

Klassenname	Zeichenbereich
Oberklasse	Textbereich
Beschreibung	Ein Zeichenbereich ist ein mehrzeiliges Textfeld, auf dessen Zeichen zugegriffen werden kann. Die Dienste aus der abstrakten Oberklasse Textbereich beziehen sich auf die Zeichen des Bereichs.



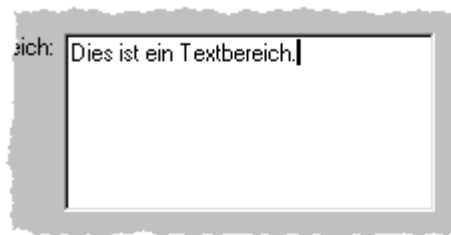
Es kann die Konstante "NEUERABSATZ" benutzt werden, um neue Absätze in dem Zeichenbereich zu erzeugen.

Protokoll	<p>Anfrage markierungsAnfang : GanzeZahl <i>nachher</i> Diese Anfrage liefert die Nummer vom ersten Zeichen des markierten Bereichs.</p> <p>Anfrage markierungsEnde : GanzeZahl <i>nachher</i> Diese Anfrage liefert die Nummer vom letzten Zeichen des markierten Bereichs.</p> <p>Auftrag setzeMarkierung(pAnfang, pEnde : GanzeZahl) <i>nachher</i> Beginnend mit dem angegebenen ersten Zeichen bis zum letzten Zeichen wurde der Inhalt des Bereichs markiert.</p> <p>Anfrage teillnhalt(pAnfang, pEnde : GanzeZahl) <i>nachher</i> Diese Anfrage liefert einen Teil des Bereichs beginnend mit dem angegebenen ersten Zeichen bis zum letzten Zeichen.</p> <p>Auftrag fuegeEin(pText : Zeichenkette, pStelle : GanzeZahl) <i>nachher</i> Der angegebene Text wurde an die angegebene Zeichennummer eingefügt.</p>
-----------	--



Dokumentation der Klasse "Zeilenbereich"

Klassenname	Zeilenbereich
Oberklasse	Textbereich
Beschreibung	Ein Zeilenbereich ist ein mehrzeiliges Textfeld, auf dessen Zeilen zugegriffen werden kann. Die Dienste aus der abstrakten Oberklasse Textbereich beziehen sich auf die Zeilen des Bereichs.



Protokoll	<p>Anfrage markierungsAnfang : GanzeZahl <i>nachher</i> Diese Anfrage liefert die Zeilennummer vom Anfang des markierten Bereichs.</p> <p>Anfrage markierungsEnde : GanzeZahl <i>nachher</i> Diese Anfrage liefert die Zeilennummer vom Ende des markierten Bereichs.</p> <p>Auftrag setzeMarkierung(pAnfang, pEnde : GanzeZahl) <i>nachher</i> Die Markierung des Bereichs erstreckt sich von der angegebenen Anfangszeile bis zur Endzeile des Bereichs.</p> <p>Anfrage teilInhalt(pAnfang, pEnde : GanzeZahl) <i>nachher</i> Diese Anfrage liefert den Inhalt von der angegebenen Anfangszeile bis zur Endzeile des Bereichs.</p> <p>Auftrag fuegeEin(pText : Zeichenkette, pStelle : GanzeZahl) <i>nachher</i> Der angegebene Text wurde an die angegebene Zeile eingefügt.</p>
-----------	--