

16.16 TextEditor (gb.form.editor)

Dieses Steuerelement ist ein erweiterter Texteditor mit Unterstützung für Syntax-Hervorhebung und besitzt die folgenden besonderen Eigenschaften:

- Hervorhebung: C, C++, Css, Diff, Gambas, Highlight, Html, Javascript, Sh, Sql und Webpage.
- Vollständig verwaltete Bearbeitung einschließlich Kopieren, Einfügen, Ausschneiden, Rückgängig machen und Wiederherstellen.
- Kontext-Menü, Text-Suche und Auswahl einer Editor-Zeile.
- Auswählbare Anzeigefunktionen wie Zeilennummerierung, Indikatoren für das Falten von Blöcken, Indikatoren für die Übereinstimmung von Klammern, Lesezeichen, Tabulator-Erweiterung, Haltepunkt-Anzeigen u.a. .

So erzeugen Sie ein neues TextEditor-Objekt:

```
Dim hTextEditor As TextEditor
hTextEditor = New TextEditor ( Parent As Container ) [ As "EventName" ]
```

Diese Klasse verhält sich wie ein Nur-Lese-Array.

```
Dim hTextEditor As TextEditor
Dim hTextEditorLine As _TextEditor_Line
hTextEditorLine = hTextEditor [ Line As Integer ]
```

Es wird ein virtuelles Objekt zurück gegeben, das eine bestimmte Zeile repräsentiert. Der optionale Parameter 'Line' ist die Zeilennummer des zurückgegebenen Objekts.

16.16.1 Eigenschaften

Die Klasse TextEditor verfügt über diese Eigenschaften:

Eigenschaft	Datentyp	Beschreibung
Bookmarks	Integer []	Gibt die Liste der Zeilen mit einem Lesezeichen zurück oder definiert diese Liste.
Border	Boolean	Ermittelt oder setzt, ob das Steuerelement einen Rahmen hat.
Breakpoints	Integer []	Gibt die Liste der Zeilen mit einem Haltepunkt zurück oder definiert diese Liste.
CloseBraces	Boolean	Liefert oder setzt, ob Klammern automatisch geschlossen werden. Zu den Klammern gehören Klammern (&), spitze Klammern [&] und geschweifte Klammern { & }. Wenn Sie (drücken, ohne dass Text davor steht, wird die schließende Klammer (&) rechts vom Cursor gesetzt. Wenn Sie (mit Text unmittelbar rechts davon drücken, wird nur das eine ((normalerweise) erzeugt. Markieren Sie eine beliebige Textmenge und drücken Sie (, um automatisch beide geschweiften Klammern an beiden Enden des Textes hinzuzufügen.
CloseStrings	Boolean	Gibt zurück oder legt fest, ob Zeichenketten automatisch geschlossen werden. Wenn Sie " drücken, ohne dass Text davor steht, wird das abschließende " rechts vom Cursor erzeugt. Wenn Sie " drücken, während sich unmittelbar rechts davon Text befindet, wird nur ein " erzeugt (normalerweise). Wählen Sie eine beliebige Textmenge aus und drücken Sie ", um automatisch beide Anführungszeichen an beiden Enden des Textes hinzuzufügen.
Column	Integer	Liefert die Cursorposition in der aktuellen Zeile. Das erste Zeichen der Zeile steht an der Position Null.
Count	Integer	Gibt die Anzahl der Zeilen zurück.
Current	_TextEditor_Line	Gibt die aktuelle Zeile als virtuelles Objekt zurück. Hinweis: Unter https://gambaswiki.org/wiki/comp/gb.form.editor/_text_editor_line wird die virtuelle Klasse ausführlich beschrieben.
DiffMode	Boolean	Liefert oder setzt, ob der "Diff-Modus" aktiviert ist. Wenn der "Diff-Modus" aktiviert ist, wird der Editor schreibgeschützt und zeigt jeden hinzugefügten oder entfernten Text an, der mit der Methode EnterDiffMode definiert wurde.
DisabledBreakpoints	Integer []	Gibt die Liste der Zeilen mit einem deaktivierten Haltepunkt zurück oder setzt sie.

Eigenschaft	Datentyp	Beschreibung
EndOfLine	Integer	Gibt das Trennzeichen für Zeilenumbrüche zurück, das der Editor verwendet oder setzt. Die möglichen Werte sind: gb.Unix für durch Chr\$(10) getrennte Zeilen, gb.Windows für durch Chr\$(13) & Chr\$(10) getrennte Zeilen oder gb.Mac für durch Chr\$(13) getrennte Zeilen.
Font	Font	Gibt die Schriftart zurück oder legt sie fest, die zum Zeichnen von Text im Steuerelement verwendet wird. Um die Schriftart des Steuerelements zur Laufzeit zu ändern gilt: Verwenden Sie die Eigenschaft Font eines anderen Steuerelements. Erstellen Sie ein Font-Objekt aus einer Zeichenkette, die die Schriftart beschreibt.
Highlight	String	Gibt den Hervorhebungsmodus zurück oder setzt ihn. Er kann einen der folgenden Werte annehmen: "c", "c++" oder "cplusplus", "css", "custom", "diff", "gambas", "html", "none", "javascript", "sh", "sql", "webpage". Wenn der Hervorhebungsmodus auf "custom", "none" oder einen Null-String eingestellt ist, wird das Highlight-Ereignis jedes Mal ausgelöst, wenn eine Zeile hervorgehoben werden muss. Wenn Sie den Handler für dieses Ereignis nicht implementieren, wird nichts hervorgehoben. Die Eigenschaft ist standardmäßig NULL. Die erforderlichen Daten sind der Unterklassenname der Hervorhebungssyntax - nicht eine der Konstanten der Klasse Highlight.
Keywords	String []	Gibt die Sprachschlüsselwörter zurück, die mit dem durch die Eigenschaft 'Modus' definierten Editor-Modus verbunden sind.
LastColumn	Integer	Gibt die alte Spaltenposition vor einer Cursorverschiebung zurück. Diese Eigenschaft ist nur während eines Change-Event-Handlers von Bedeutung.
LastLine	Integer	Gibt die alte Zeilenposition vor einer Cursorbewegung zurück. Diese Eigenschaft ist nur während eines Change-Event-Handlers von Bedeutung.
Length	Integer	Berechnet die Länge der Eigenschaft 'Text' als UTF-8-Zeichen und gibt sie zurück.
Line	Integer	Gibt die Position der Cursorzeile zurück. Die erste Zeile befindet sich an der Position Null.
LineHeight	Integer	Gibt die Höhe einer Zeile in Pixeln zurück.
LineNumberOffset	Integer	Gibt den Offset der Zeilennummer zurück oder setzt ihn. Der Wert dieser Eigenschaft wird für die Anzeige zur tatsächlichen Zeilennummer addiert.
Max	Integer	Gibt die Anzahl der Zeilen - minus eins - zurück.
Mode	String	Gibt den Editor-Modus zurück oder legt ihn fest. Der Editor-Modus bestimmt, wie der Text hervorgehoben wird und wie sich der Editor verhält. Die Eigenschaft kann die gleichen Werte annehmen wie die Eigenschaft Highlight. Wenn der Editor-Modus NULL ist, dann gibt es kein benutzerdefiniertes Editor-Verhalten und der Text wird nur hervorgehoben, wenn die Eigenschaft Highlight gesetzt ist.
Overwrite	Boolean	Gibt zurück oder setzt die Eigenschaft, ob sich der Editor im Überschreibmodus befindet.
ReadOnly	Boolean	Gibt den Wert der Eigenschaft zurück, ob der Editor schreibgeschützt ist oder setzt ihn.
ScrollBar	Integer	Gibt zurück oder legt fest, welche Bildlaufleisten angezeigt werden. Eine Liste der Konstanten, die mit dieser Eigenschaft verwendet werden können, finden Sie in der Klasse Scroll.
ScrollH	Integer	Gibt die Höhe des (Text-)Inhalts zurück.
ScrollHeight	Integer	Gibt die Höhe des (Text-)Inhalts zurück.
ScrollW	Integer	Gibt die Weite des (Text-)Inhalts zurück.
ScrollWidth	Integer	Gibt die Weite des (Text-)Inhalts zurück.
ScrollX	Integer	Gibt die horizontale Bildlaufposition zurück oder legt sie fest.
ScrollY	Integer	Gibt die vertikale Bildlaufposition zurück oder legt sie fest.

Eigenschaft	Datentyp	Beschreibung
Selected	Boolean	Liefert True zurück, wenn ein Text ausgewählt ist.
SelectedText	String	Gibt den im Editor markierten Text zurück.
SelectionColumn	Integer	Gibt die Spaltenposition der Auswahlmarkierung zurück. Ist kein Text ausgewählt, wird der gleiche Wert wie in der Eigenschaft Column zurückgegeben.
SelectionLine	Integer	Gibt die Zeilenposition der Textmarkierung zurück. Wenn kein Text ausgewählt ist, wird derselbe Wert wie in der Eigenschaft Line zurückgegeben.
ShowBraces	Boolean	Gibt zurück, ob übereinstimmende Klammern hervorgehoben sind oder setzt den Wert der Eigenschaft.
ShowCurrent	Boolean	Gibt zurück, ob die aktuelle Zeile hervorgehoben ist oder setzt den Wert der Eigenschaft.
ShowCursor	Boolean	Gibt zurück, ob der Cursor sichtbar ist oder setzt den Wert der Eigenschaft.
ShowExpand	Boolean	Gibt zurück, ob die Funktionen ein- und ausgeklappt werden können oder setzt den Wert der Eigenschaft.
ShowIcon	Boolean	Gibt zurück, ob die Haltepunkt- oder Lesezeichensymbole sichtbar sind oder setzt den Wert der Eigenschaft.
ShowIndent	Boolean	Gibt zurück, ob der Editor Tabulatorzeichen als kleine graue Pfeile anzeigt oder setzt den Wert der Eigenschaft.
ShowLimit	Boolean	Gibt zurück, ob die Funktionsgrenzen sichtbar sind oder setzt den Wert der Eigenschaft.
ShowLineNumber	Boolean	Gibt zurück, ob der Editor Zeilennummern anzeigt oder setzt den Wert der Eigenschaft.
ShowModified	Boolean	Zurückgeben oder festlegen, ob die geänderten Zeilen am Rand farbig markiert/hervorgehoben werden.
ShowPosition	Boolean	Gibt zurück, ob die aktuelle Position des Cursors im Editor angezeigt wird oder setzt den Wert der Eigenschaft.
ShowPreview	Boolean	Gibt zurück, ob der Editor seine Bildlaufleiste durch eine verkleinerte Vorschau ersetzt oder setzt den Wert der Eigenschaft.
ShowSpaces	Boolean	Gibt zurück ob Leerzeichen am Ende einer Zeile als Punkte angezeigt werden oder setzt den Wert der Eigenschaft.
StopLine		Liefert oder setzt die Debugger-Stopzeile.
Theme	TextHighlighterTheme	Gibt das für die Hervorhebung verwendete Thema zurück oder legt es fest. Ersetzt die Eigenschaft Styles seit Version 3.17.
TabIndent	Boolean	Liefert oder setzt, ob der Ident aus Tabulatorzeichen besteht. Wenn diese Eigenschaft TRUE ist, wird für jeden Einzug ein "t"-Zeichen verwendet. Ist diese Eigenschaft FALSE, wird für jeden Einzug die durch die Eigenschaft TabSize festgelegte Anzahl von Leerzeichen verwendet.
TabSize	Integer	iefert oder setzt die Tabulatorgröße. Dies ist die Anzahl der Leerzeichen, die für die Einrückung verwendet werden.
Text	String	Gibt den Textinhalt des Editors zurück oder setzt ihn.
View	TextEditor	Diese Eigenschaft erlaubt es verschiedenen Editoren, denselben Inhalt zu teilen. Wenn diese Eigenschaft NULL ist (Standardeinstellung), dann hat der Editor seinen eigenen Inhalt. Andernfalls verwendet der Editor den Inhalt seiner eigenen View.
Wrap	Boolean	Gibt zurück oder legt fest, ob der Editor die Zeilen umbricht, um sie an die Breite des Steuerelements anzupassen. Wenn diese Eigenschaft gesetzt ist, haben Sie folglich *keine* horizontale Bildlaufleiste.

Tabelle 16.16.1.1 : Eigenschaften der Klasse TextEditor

16.16.2 Methoden

Die Klasse TextEditor verfügt über diese Methoden:

Methoden	Rückgabebetyp	Beschreibung
Append (Text As String)	-	Fügt einen Text an das Text-Ende des Editors an, ohne den Cursor zu bewegen. Text ist der Text, der am Ende des Editorinhalts eingefügt wird. Der Text kann mehrzeilig sein. Der Zeilentrenner ist in diesem Fall immer "\n".
Begin ()	-	Startet eine Reihe von Änderungen, die auf einen Schlag rückgängig gemacht werden können.
End ()	-	Beendet eine Reihe von Änderungen.
Clear ()	-	Löscht den Text-Inhalt des Editors.
CollapseAll ()	-	Bewirkt das Ausklappen aller Funktionen.
ExpandAll ()	-	Bewirkt das Expandieren aller Funktionen.
Copy ()	-	Kopiert den aktuell ausgewählten Text in die Zwischenablage.
Paste ()	-	Fügt den Inhalt der Zwischenablage in den Editor ein.
CursorAt ([Line As Integer, Column As Integer])	Point	Konvertiert eine Cursorposition in eine Pixelposition. Die beiden optionalen Parameter Line und Column definieren die Cursorposition. Wenn sie weggelassen werden, werden sie durch die (aktuelle) Cursorposition ersetzt. Diese Methode gibt die Position in Pixeln als Point-Objekt zurück.
Cut ()	-	Kopiert den aktuell ausgewählten Text in die Zwischenablage und löscht den ausgewählten Text.
EnterDiffMode (Added As Integer[], Removed As Collection)	-	Aufruf des "Diff-Modus". Added ist die Liste der Zeilennummern, die hinzugefügt werden. Diese Zeilen werden mit der Farbe Highlight.Added hervorgehoben. Removed ist eine Sammlung von String-Arrays, die durch die Zeilennummer indiziert sind, wobei das String-Array die Textzeilen enthält, die bei dieser Zeilennummer entfernt wurden. Die entfernten Zeilen werden mit der Farbe Highlight.Removed hervorgehoben. Im "Diff-Modus" wird der Editor schreibgeschützt.
LeaveDiffMode ()	-	Verlässt den Diff-Modus. Der Editor kehrt in seinen normalen Zustand zurück.
FindNextDiff (Y As Integer)	Integer	Sucht die Zeilenposition der nächsten Differenz. Y ist die Zeilenposition, an der die Suche beginnt. Wenn kein Unterschied gefunden wird, wird -1 zurückgegeben.
FindPreviousDiff (Y As Integer)	Integer	Sucht die Zeilenposition der vorherigen Differenz. Y ist die Zeilenposition, an der die Suche beginnt. Wenn kein Unterschied gefunden wird, wird -1 zurückgegeben.
FindNextLimit (Y As Integer)	Integer	Sucht die Zeilenposition der nächsten Funktionsgrenze. Y ist die Zeilenposition, an der die Suche beginnt. Wenn keine Grenze gefunden wird, wird -1 zurückgegeben.
FindNextString (Search As String, ByRef Start As Integer [, ByRef Column As Integer])	String	Suche nach der nächsten Zeile, die eine bestimmte Zeichenfolge enthält oder nach dem nächsten Vorkommen einer Zeichenfolge. String ist die zu suchende Zeichenfolge. Bei der Suche wird die Groß- und Kleinschreibung nicht berücksichtigt. Start ist die Zeilenposition, an der die Suche beginnt. Column ist die optionale Zeilenposition, ab der gesucht werden soll. Wenn Column angegeben wird, wird die Zeichenkette ab dieser Spalte gesucht. Das Ergebnis wird in den Argumenten Start und Column zurückgegeben. Wenn keine Zeichenkette gefunden wird, wird -1 zurückgegeben. Gibt die Zeilennummer der nächsten gefundenen Suchzeichenfolge ab der Position Start zurück. Gibt -1 zurück, wenn keine Zeichenkette gefunden wurde. Sucht nach einer gegebenen Suchzeichenkette ab der gegebenen Start-Zeilennummer und gibt die Zeilennummer der nächsten erfolgreichen Suche zurück, wobei die erste Zeile 0 ist. Gibt -1 zurück, wenn die Suchzeichenkette nicht gefunden wird. Bei der Suche wird die Groß-/Kleinschreibung nicht berücksichtigt.

Method	Rückgabotyp	Beschreibung
FindPreviousLimit (Y As Integer)	Integer	Sucht die Zeilenposition der vorherigen Funktionsgrenze. Y ist die Zeilenposition, an der die Suche beginnt. Wenn keine Grenze gefunden wird, wird -1 zurückgegeben.
FindPreviousString (Search As String, ByRef Start As Integer [, ByRef Column As Integer])	Integer	Sucht die vorherige Zeile, die eine bestimmte Zeichenfolge enthält oder sucht das vorherige Vorkommen einer Zeichenfolge. Zeichenfolge ist die zu suchende Zeichenfolge. Bei der Suche wird die Groß- und Kleinschreibung nicht berücksichtigt. Start ist die Zeilenposition, an der die Suche beginnt. Column ist die optionale Zeilenposition, ab der gesucht werden soll. Wenn Column angegeben wird, wird die Zeichenkette ab dieser Spalte gesucht, und das Ergebnis wird in den Argumenten Start und Column zurückgegeben. Wenn keine Zeichenkette gefunden wird, wird -1 zurückgegeben. Gibt die Zeilennummer der nächsten gefundenen "umgekehrten" Suchzeichenfolge ab der Position Start zurück. Gibt -1 zurück, wenn nicht gefunden. Sucht rückwärts nach einer gegebenen Suchzeichenkette ab der gegebenen Start-Zeilenummer und gibt die Zeilennummer der erfolgreichen Suche zurück, wobei die erste Zeile 0 ist. Gibt -1 zurück, wenn die Suchzeichenkette nicht gefunden wird. Bei der Suche wird die Groß-/Kleinschreibung nicht berücksichtigt.
GetHighlight (Y As Integer, ByRef Symbols As String[], ByRef Types As Integer[])	-	Gibt die Hervorhebung einer bestimmten Zeile zurück. Y ist die Zeilennummer. Die Liste der Token wird in Symbols zurückgegeben. Die Hervorhebungsfarben für jedes Token werden in Types zurückgegeben. Wenn der angegebenen Zeile keine Hervorhebung zugeordnet ist, werden Symbols und Types auf NULL gesetzt.
GetIndent (StartLine As Integer [, EndLine As Integer])	String	Gibt den Einzug einer Zeile oder eines Zeilenbereichs zurück. StartLine ist die erste Zeile des Bereichs. EndLine ist die letzte Zeile des Bereichs (nicht eingeschlossen). Wenn nicht angegeben, wird der Einzug der einzigen Startzeile zurückgegeben. Der Einzug ist der kleinere Teil der anfänglichen Leerzeichen einer jeden Zeile.
GetSelectedLines	String []	Seit Version 3.16: Gibt die markierten Zeilen zurück, falls vorhanden. Die Auswahl wird zunächst erweitert, so dass teilweise ausgewählte Zeilen in die Auswahl aufgenommen werden. Wenn nichts ausgewählt ist, wird NULL zurückgegeben.
Goto (NX As Integer, NY As Integer [, Mark As Boolean])	-	Bewegt den Cursor an die angegebene Position. NX und NY sind jeweils die Spalte bzw. die Zeile der neuen Position. Wenn Mark TRUE ist, dann wird der Text zwischen der alten Position und der neuen Position ausgewählt.
GotoCenter (NX As Integer, NY As Integer)	-	Verschiebt den Cursor an die angegebene Position und zentriert die Ansicht um diese Position. NX und NY sind jeweils die Spalte und die Zeile der neuen Position. Bewegt den Cursor mittig an die angegebene Position NX (Spaltennummer) und NY (Zeilennummer).
HideSelection ()	-	Blendet die aktuelle Auswahl aus, falls vorhanden.
HighlightString (Text As String [, IgnoreCase As Boolean, WordOnly As Boolean])	-	Hervorhebung einer bestimmten Zeichenfolge überall im Editor. Text ist der Text, der hervorgehoben werden soll. Wenn IgnoreCase TRUE ist, wird die Groß- und Kleinschreibung des Textes ignoriert. Wenn WordOnly TRUE ist, wird der Text nur hervorgehoben, wenn er ein ganzes Wort ist. Der Text wird mit der Farbe Highlight.Highlight hervorgehoben.
Indent ()	-	Rückt den markierten Text ein oder die aktuelle Zeile, wenn kein markierter Text vorhanden ist.
Insert (Text As String)	-	Fügt einen Text an der aktuellen Cursorposition ein. Text ist der einzufügende Text. Der Text kann mehrzeilig sein. Der Zeilentrenner ist in diesem Fall immer "\n".
IsEmpty ()	Boolean	Gibt True zurück, wenn der Inhalt des Editors leer ist.
Load (Path As String [, AddNewLine As Boolean])	-	Lädt eine Datei in den Editor. Path ist der Pfad der zu ladenden Datei. Wenn AddNewLine TRUE ist, wird ein Zeilenumbruch am Ende der Datei eingefügt, falls kein Zeilen-

Methode	Rückgabety	Beschreibung
		umbruch vorhanden ist. Lädt eine Datei aus " Path " in den TextEditor und stellt optional sicher, dass das letzte Zeichen ein Zeilenumbruch ist. Beispiel: TextEditor1.Load("/Filepath/Filename.txt", True). Damit wird die Datei "/Filepath/Filename.txt" in den Editor geladen und sichergestellt, dass die letzte Textzeile einen Zeilenumbruch enthält (fügt einen Zeilenumbruch hinzu, wenn die letzte Zeile einen Text enthält).
Save (Path As String)	-	Speichert den Inhalt des Editors in eine Datei. Path ist der Pfad der Datei.
LowerCase ()	-	Wandelt den markierten Text in Kleinbuchstaben um. Wenn kein Text ausgewählt ist, bewirkt diese Methode nichts!
UpperCase ()	-	Wandelt den markierten Text in Großbuchstaben um. Wenn kein Text ausgewählt ist, bewirkt diese Methode nichts!
Redo ()	-	Die letzte rückgängig gemachte Textänderung im Editor wird wiederhergestellt.
Undo ()	-	Die letzte Textänderung des Editors rückgängig machen.
Refresh ()	-	Aktualisiert den Text im Editor.
Remove (X1 As Integer, Y1 As Integer, X2 As Integer, Y2 As Integer [, bRight As Boolean])	-	Entfernt den Text, der sich zwischen zwei Cursorpositionen befindet. X1 und Y1 sind die Spalte und Zeile der ersten Position. X2 und Y2 sind die Spalte und die Zeile der anderen Position. Setzen Sie bRight auf TRUE, wenn der Cursor am Anfang der Auswahl stand.
Reset ()	-	Setzt alle Änderungs- und Speicherkennzeichen jeder Zeile zurück.
RestoreCursor ()	-	Stellt die Cursorposition und den Auswahlbereich wieder her, die mit der Methode SaveCursor() gespeichert wurden. Die Aufrufe von SaveCursor() und dieser Methode können ineinander verschachtelt werden.
Select (Column As Integer, Line As Integer, Column2 As Integer, Line2 As Integer)	-	Wählt einen bestimmten Textbereich aus. Column und Line definieren die Anfangsspalte und -zeile des Textbereichs. Column2 und Line2 definieren die Endspalte und -zeile des Textbereichs.
SelectAll ()	-	Markiert den kompletten Text des Editors.
ShowFindPanel ()	-	Zeigt das Feld für die Textsuche unterhalb des Textfeldes an.
ShowGotoPanel ()	-	Zeigt das Feld für die Sprungszeile an.
ToColumn (X As Integer, Y As Integer)	Integer	Gibt die Spalte zurück, die sich unter der angegebenen Mauszeigerposition befindet. X und Y sind die Position des Mauszeigers relativ zum TextEditor-Steuerelement.
ToLine (Y As Integer)	Integer	Gibt die Zeile zurück, die sich unter der angegebenen Mauszeigerposition befindet. Y ist die vertikale Position des Mauszeigers relativ zum TextEditor-Steuerelement.
ToPos ([Line As Integer, Column As Integer])	Point	Gibt die relative Position einer bestimmten Cursorposition in Pixeln zurück. Column und Line sind die Spalten- und Zeilenposition. Wird sie weggelassen, wird die aktuelle Cursorposition verwendet.
ToPosX ([Column As Integer, Line As Integer])	Integer	Gibt die relative horizontale Position einer bestimmten Cursorposition in Pixeln zurück. Column und Line sind die Spalten- und Zeilenposition. Wird nichts angegeben, wird die aktuelle Cursorposition verwendet.
ToPosY ([Line As Integer])	Integer	Gibt die relative vertikale Position einer bestimmten Cursorposition in Pixeln zurück. Line ist die Zeilenposition. Wird sie weggelassen, wird die aktuelle Cursorposition verwendet.
Unindent ()	-	Rückt den markierten Text ein oder die aktuelle Zeile, wenn kein Text markiert ist.

Tabelle 16.16.2.1 : Methoden der Klasse TextEditor

16.16.3 Ereignisse

Die Klasse TextEditor verfügt über die folgenden relevanten Ereignisse:

Ereignis	Beschreibung
Change ()	Dieses Ereignis wird ausgelöst, wenn sich der Inhalt des Editors geändert hat.
Cursor ()	Dieses Ereignis wird ausgelöst, wenn der Cursor bewegt wurde.
Highlight (Text As String)	Dieses Ereignis wird ausgelöst, wenn eine Textzeile hervorgehoben werden muss. Text ist der Text, der Text hervorgehoben werden soll. Dieses Ereignis wird nur ausgelöst, wenn die Eigenschaft Highlight auf "custom", "none" oder einen Null-String gesetzt ist.
Margin ()	Dieses Ereignis wird ausgelöst, wenn der Benutzer zweimal innerhalb des Randes geklickt hat. Der Cursor wurde vor dem Auslösen des Ereignisses auf die angeklickte Zeile bewegt.
Scroll ()	Dieses Ereignis wird ausgelöst, wenn im Editor gescrollt wurde.

Tabelle 16.16.3.1 : Ereignisse der Klasse TextEditor

16.16.3.1 Beispiel-Projekt

Das vorgestellte Projekt erlaubt es Ihnen,

- ein neues HTML-Dokument anzulegen,
- eine HTML-Datei zu öffnen und deren Inhalt in den Editor einzufügen,
- den HTML-Text im Editor zu bearbeiten und in der (existierenden) Datei zu speichern,
- den HTML-Text in einer Datei unter einem frei gewählten Namen in einer Datei zu speichern,
- das Text-Highlighting für die Auszeichnungssprache HTML ein- und auszuschalten,
- nach Text im Editor zu suchen,
- den HTML-Text zu rendern und in einem internen Webbrowser anzuzeigen und
- den Inhalt des HTML-Dokumentes auszudrucken sowie
- mit der Taste F1 eine umfangreichen Dokumentation (PDF-Dokument) aufzurufen.

Das Projekt wird Ihnen in einem Projekt-Archiv im Download-Bereich zur Verfügung gestellt. Im nächsten Abschnitt wird Ihnen die grafische Benutzer-Oberfläche (GUI) vorgestellt.

16.16.3.1.1 Benutzer-Oberfläche (GUI)

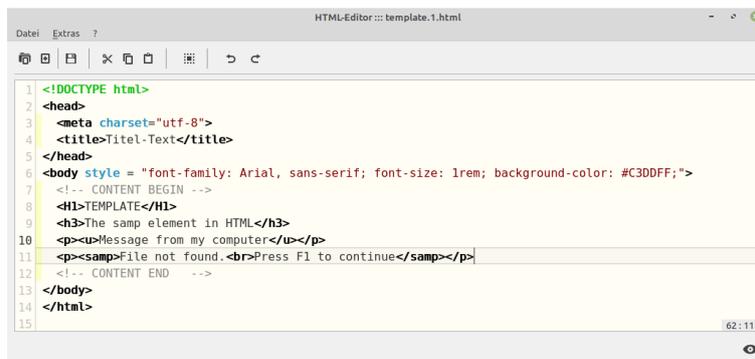


Abbildung 16.16.3.1.1: Benutzer-Oberfläche : Editor



Abbildung 16.16.3.1.2: Benutzer-Oberfläche: Anzeige der Webseite im Webbrowser