OpenOffice.org API (calc)

Simplified Reference

Author : **Malcolm Ripley**
Issue : **1**
Date : **11th September 2007**

Introduction

This document is intended to be a simple reference to the OpenOffice API calls that more closely resemble the Microsoft VB Object calls. The use of the UNO dispatcher calls has been avoided. These calls are for use in a spreadsheets although many are applicable to tables and text in write documents.

The document is organised into 4 sections. The first section contains a few notes on how to use the calls effectively that I have picked up from testing and reading! The second section is a list of all object, collections and structures since these are all, essentially, objects. The third section is a list of constants and enumerations. The fourth section is some procedures that I have written that you may find useful, espcially the “Active...” variety. Wherever possible, all properties for an object are listed under that object. However, there are some groups of properties that are quite extensive and apply to multiple objects. I have referred to these as “property groups”.

Most of the examples here have been lifted from various documents, mainly Staroffice. They have been re-written to make things a little clearer especially where the variables are concerned. All variables follow a convention to clearly differentiate a variable from a method, property or object name.

This is currently a work in progress document with sections in blue text being untested and/or unresolved. I am also double checking for consistency in presentation as I have changed my mind a few times.

Notes

These are a few notes based on my discoveries along the way. First object names:

OpenOffice	**Excel**
Component	Document
Data Pilot Table	Pivot Table
Controller	Window

Object Hierarchy

If a level is surrounded in angle brackets this means that this level is optional in order to reach lower levels. For example a sheet object can refer to a range which then refers to a cell. However a sheet can directly refer to a cell.

Star Desktop

- Documents/Components
  - <Document>
    - Sheets - <Sheet>
      - Columns
        - Cells or <Range>
          - <Cell>
      - Rows
        - Cells or <Range>
          - <Cell>
      - Cells or <Range>
        - <Cell>
      - Charts - <Chart>
        - ChartDocument
          - ChartArea
          - ChartTitle
          - Subtitle (see ChartTitle)
          - ChartLegend
          - Diagram
            - ChartLine
          - Data
      - DrawPage/Shapes
        - Shape
          - FormControl
    - NamedRanges
    - DataBaseRanges
    - DrawPages – <DrawPage>
      - <Shapes> - <Shape>

StyleFamilies

CellStyles - <CellStyle>

PageStyles - <PageStyle>

HeaderFooterContent

HeaderFooterText

Parameters

All parameters for methods use a convention to identify the datatype of the parameter:

<l_Aname> Long word

<o_Aname> Object

<s_Aname> String

<w_Aname> Short word

<b_Aname> Boolean

<d_Aname> Date

<v_Aname> Variant

<f_Aname> Float (a double float, I have not yet found a single float being used)

Where “Aname” is a descriptive name to match its use, it can be anything. Some parameters require constants, enumerations and structures:

<const_TheName>

<enum_TheName>

<struct_TheName>

<seq_TheName>

In this case TheName is the name of the constant, enumeration, structure or sequence. The actual values are listed in the relevant section. These should be self explanatory.

As far as a sequence is concerned the name given is that of the property set which is accessed via a com.sun.star.beans.PropertyValue sequence. This fooled me for a while since a sequence is an array so here is an example of the code that reads a property set:

```
Dim oMedia As Object
oMedia = ThisComponent.getArgs
Print oMedia(0).Name
Print oMedia(0).Value
```

In the above example oMedia is a property set object called MediaDescriptor.

Consider this code which writes to a property set.

```
Dim args(2) As New com.sun.star.beans.PropertyValue
args(0).Name = “ReadOnly”
args(0).Value = True
args(1).Name = “AsTemplate”
args(1).value = FALSE
oDoc = StarDesktop.LoadComponentFromUrl(“c:\Temp\test.ods”,”_blank”,2,args())
```

This will load the document test.ods into a new window(frame). In this case we define a specific property set object array called args and set the properties “ReadOnly” and “AsTemplate” for a MediaDescriptor property set.

Issues

Due to the way structures have been implemented the following does not work:

```
oObj.structMember1.element1 = value
```

Instead two steps are required:

```
tStruct.element1 = value
oObj.tSructMember1 = tStruct
```

However, with some structures the element values can be accessed as if they were a property of the object. So, in the above example, if “tStruct” was a “FontDescriptor” structure and “element1” was an element called “Family”. The following example prints the font family name using the structure element and then using the object property.

```
Dim oControl as object
Dim sFontDescr as variant
oControl = ThisComponent.Sheets(0).DrawPage(0).Control
sFontDescr = oControl.FontDescriptor
msgbox sFontDescr.Name
msgbox oControl.FontName
```

Excel “Active” equivelents

Excel refers to a number of “Active” properties which does make life a little simpler. The equivelent OOO coding is shown below:

ActiveWorkbook	ThisComponent
ActiveWindow	ThisComponent.CurrentController
ActiveSheet	ThisComponent.CurrentController.ActiveSheet
ActiveCell	ThisComponent.CurrentSelection (if a cell selected) otherwise the cursor position has to be determined.
ActiveChart	Coding is required to determine this.

There is a section at the end of this document which has some useful functions which perform all of the above.

Case Sensitivity

Method and property names are case insensitive. Structure names, enumeration names, enumerated values and constant names are case sensitive. For example only the last one of the following three is valid.

orient = com.sun.star.table.cellorientation.TOPBOTTOM

orient = com.sun.star.table.CellOrientation.TopBottom

orient = com.sun.star.table.CellOrientation.TOPBOTTOM

Example Code

I have adopted the following convention for variables in any code examples:

lAname	Long word
oAname	Object
sAname	String
wAname	Short word or integer (although long word is strictly an integer!)
bAname	Boolean
dAname	Date
vAname	Variant
fAname	Double Float
tAname	Structure

Sorting

A small warning about sort, filters etc. If you were to perform a sort using the following method you have to ensure that the correct array entry is modified:

```
Sub Main
Dim oRange,oSort as Object
Dim sDescr(0) as new com.sun.star.table.TableSortField
oRange = ActiveSheet.getcellrangebyname("a1:b5")
oSort = oRange.createsortdescriptor
sDescr(0).Field = 1
sDescr(0).IsAscending = false
oSort(3).Value = sDescr
oRange.sort(oSort)
End Sub
```

Note how oSort(3).value is set to the table sort field descriptor. This is because the createsortdescriptor method creates an object with all the properties. It would seem to be either an existing one or a new blank one. It just so happens that the fourth entry is the parameter "SortFields". Cleary this is an awkward implementation since you have to "know" which property applies to a specific array entry. The following procedure creates an "empty" sort descriptor and therefore you can specify any relevant property to any array entry:

```
Sub Main
Dim oRange as Object
dim sDescr(0) as new com.sun.star.table.TableSortField
Dim oSort(0) as new com.sun.star.beans.PropertyValue
oRange = ActiveSheet.getcellrangebyname("a1:b5")
sDescr(0).Field = 1
sDescr(0).IsAscending = true
oSort(0).Name = "SortFields"
oSort(0).Value = sDescr
oRange.sort(oSort)
End Sub
```

Text

I have had difficulty in interpreting how text works! There is the ability to insert and replace parts of a text object. However, the methods and properties are recursive with the descriptions being short and written in very poor english. So for now I have only included the String property which can be used to modify the whole string. For spreadsheets cells and simple text graphics this is not a problem. For paragraphs of text this is far too simple and I will try and resolve the handling of text.

Objects, Collections and Structures

In this list those collections labelled with an asterisk "(*)" have, essentially, a method/property name which matches the collection name. For example:

```
oDoc = ThisComponent
oSheet = oDoc.Sheets.GetByName("Sheet1")
```

Quite often a method like getxxx returns a collection called xxx and xxx can be used directly rather than getxxx. This is also the case for some properties. I cannot confirm if properties have been created for all methods of all objects that are like this. However, wherever this is the case I have only listed the property and not the equivelent method.

For each object and collection I have listed the top of the API reference tree from where I trawled down to the list of methods and properties. I have also included a list of parent objects and collections. I have not included all methods and properties. Some methods, in my opinion, are not relevant to spreadsheet use and most of those excluded are digging deep into UNO internals. I have excluded any properties that are deprecated unless their inclusion is needed for clarity, sometimes the cross references in the API documentation are incomplete. I have also excluded all "listener" related calls (i.e. event handling), for now.

There are some methods and properties that I could not find in the API documentation and so these have no description.

Annotations(*)

Provides methods to access cell annotations via index and to insert and remove annotations.

API Reference, service(s)
com.sun.star.sheet.CellAnnotations

Parent Objects
Sheet

Methods

getByIndex(<l_Index>)
Object – returns the annotation object at the specified zero based index.

hasElements
Boolean – true if the collection has styles.

insertNew(<struct_CellAddress>,<s_AnnotationText>)
Creates a new annotation.

removeByIndex(<l_Index>)
Removes a cell annotation from the collection.

Properties

Count
Long – returns the number of annotations in the collection.

AppletShape

This Shape encapsulates an applet.

An applet is a small Java-based program that is executed and rendered embedded in a document.

API Reference, service(s)
com.sun.star.drawing.AppletShape

Parent Objects
Drawpage, GroupShape, Shapes

Methods

All methods of the object Shape.

Properties

Property Groups: Shape

AppletCode
String - This property specifies one of the following:
The name of the class file that contains the compiled applet subclass.
The path to the class, including the class file itself.

AppletCodeBase
String - This property specifies the base URI for the applet.

AppletCommands
This sequence contains paramaters that are passed to the applet when it is initialized.

AppletIsScript
Boolean readonly - This property specifies whether or not the applet can be scripted.

AppletName
String - this is an optional name for the applet.

Application

This is not an openoffice/staroffice object. However for the sake of clarity it makes sense to have one! This is where the concept of objects and methods breaks down a little, again ;-)

Methods

Thiscomponent
Returns the document where the VB code is currently executing.

AreaLink

Represents a linked cell range. A linked cell range is a range which is linked to an equal-sized range in an external document. The contents of the external range is copied into the range of this document.

API Reference, service(s)
com.sun.star.sheet.CellAreaLink

Parent Objects
CellAreaLinks

Methods

refresh
refreshes the data of the object from the connected data source.

Properties

DestArea
Structure (CellAddress) - the position of the linked range in the destination document.

Filter
String - specifies the name of the filter used to load the source document.

FilterOptions
String - specifies the filter options needed to load the source document.

RefreshPeriod
Long - specifies the time between two refresh actions in seconds.

SourceArea

String - the source of the range within the source document e.g. "Sheet1.A1:C5".

Url
String - specifies the URL of the source document.

AreaLinks(*)

This is a collection of area links.

API Reference, service(s)
com.sun.star.sheet.CellAreaLinks

Parent Objects
Document

Methodstablevalidation

CellAreaLinks
com.sun.star.sheet.CellAreaLinks

createEnumeration
Enumerated list of area links

getByIndex(<l_Index>)
Object (CellAreaLink) - The area link at that index in the collection.

hasElements
Boolean - true if the collection has area links.

insertAtPosition(<struct_CellAddress>, <s_SourceFileName>, <s_SourceArea>, <s_LoadFilter>, <s_FilterOptions>)
Creates an area link and adds it to the collection.

removeByIndex(<l_Index>)
Removes an area link from the collection.

Properties

Count
Long - the number of area links in the collection.

Bitmap

An object that provides a bitmap in the Microsoft DIB format.

API Reference, service(s)
com.sun.star.awt.XBitmap

Properties

Size
Structure (Size) - returns the size of the bitmap in pixels.

DIB
Byte array - returns the device independent bitmap.

MaskDIB
Byte array - returns the transparence mask of the device independent bitmap.

BorderLine

This is a structure that defines borders on pages.

API Reference, service(s)
com.sun.star.table.BorderLine

Elements

Color
Enum(Color) - Contains the color value of the line.

InnerLineWidth
Short - Contains the width of the inner part of a double line (in 1/100 mm).

OuterLineWidth
Short - Contains the width of a single line or the width of outer part of a double line (in 1/100 mm).

LineDistance
Short - Contains the distance between the inner and outer parts of a double line (in 1/100 mm).

CaptionShape

The CaptionShape represents a rectangular drawing shape with an additional set of lines. It can be used as a description for a fixed point inside a drawing.

API Reference, service(s)
com.sun.star.drawing.CaptionShape

Parent Objects
Drawpage, GroupShape, Shapes

Properties

Property Groups: Shape , LineProperties , Fillproperties , Text , ShadowProperties

CaptionAngle
Long - This property specifies the escape angle of the line of a caption. It is only used if CaptionIsFixedAngle is set to true

CaptionEscapeAbsolute
Long - This property specifies the absolut escape distance for the line of a caption.

CaptionEscapeDirection
Long - This property specifies the escape direction for the line of a caption.
CaptionEscapeRelative
Long - This property specifies the relativ escape distance for the line of a caption.

CaptionIsFitLineLength
Boolean - If this property is true , the application determines the best possible length for the caption line.

CaptionIsFixedAngle
Boolean - This property specifies if the escape angle of the line of a caption is fixed or free. If this is set to false , the application can choose the best possible angle. If not, the value in CaptionAngle is used.

CaptionGap
Long - This property specifies the distance between the text area of the caption and the start of the line.

CaptionIsEscapeRelative
Boolean - If this property is true , the property CaptionEscapeRelative is used, else the property CaptionEscapeAbsolute is used.

CaptionLineLength
Long - This property specifies the length of the caption line.

CaptionPoint
Structure (Point) - The caption point property specify the position of the point that is captioned. A set of lines are rendered from the caption area.

CornerRadius
Long - This is the radius of the caption area corners.

CaptionType
Short - This property specifies the geometry of the line of a caption.

Cell

This is an object that represents a single cell in a table or spreadsheet.

API Reference, service(s)
com.sun.star.sheet.SheetCell

Parent object(s)
Sheet, CellRange

Methods

clearContents(<enum_CellFlags>)
Clears the contents of cell as specified by the enumerated flags value.

createReplaceDescriptor
Object (ReplaceDescriptor) – returns a replace descriptor object that specifies a search.

decrementIndent
Shifts the indentation by one default step to the left.

incrementIndent
Shifts the indentation by one default step to the right.

replaceall(<o_ReplaceDescriptor>)
Searches for all described text in the range of cells.

Properties

Property Group(s) : CellProperties, CharacterProperties , CharacterPropertiesAsian , CharacterPropertiesComplex , ParagraphProperties

Annotation
Object (Annotation) - returns the annotation at this anchor.

CellAddress
Structure (CellAddress) - returns the address of the cell.

Columns
Object (Columns) – returns a collection of columns.

ConditionalFormat
Object (ConditionalFormat) - contains the conditional formatting settings for this cell.

ConditionalFormatLocal
Object (ConditionalFormatLocal) -contains the conditional formatting settings for this cell, using localized formulas.

Error
Long – the error number of the formula in the cell

Formula
String(!) - formula value for a cell.

FormulaLocal
String - contains the formula string with localized function names.

FormulaResultType
Long - contains the result type of a formula.

Position
Structure (Point) - contains the position of this cell in the sheet (in 1/100 mm).

Rows
Object (Rows) – returns a collection of rows (i.e 1!)

Size
Structure (Size) - contains the size of this cell (in 1/100 mm).

String
string - string value for a cell .

Type
Enum (CellContentType) – The data content of the cell

Value
Number - numeric value for a cell.

Validation
Object (Validation) - contains the data validation settings for this cell.

ValidationLocal
Object (ValidationLocal) - contains the data validation settings for this cell, using localized formulas.

CellAddress

This is a structure that defines a single cell.

API Reference, structure
com.sun.star.table.CellAddress

Elements

Sheet
Short - number of the sheet (numbering begins with 0).

Column
Long – number of the addressed column (numbering begins with 0).

Row
Long – number of the addressed row (numbering begins with 0).

CellAnnotation

Represents a cell annotation object attached to a spreadsheet cell.

API Reference, service(s)
com.sun.star.sheet.CellAnnotation

Parent object(s)
Annotations

Properties

AnnotationShape
Object (Shape) - returns the shape object for the annotation.

Author
String - returns the name of the user who last changed the annotation.

Date
String - returns a formatted string representing the date when the annotation was last changed.

IsVisible
Boolean - specifies whether the annotation is permanently visible.

Parent
Object (<various>) - specifies the object that the annotation is attached to.

Position
Structure (CellAddress) - returns the position of the cell in the spreadsheet document that contains this annotation.

String
String – the textual content.

CellCursor(*)

An object that represents a cursor in a spreadsheet.

A cursor is a cell range which provides additional methods to move through the table (i.e. to find specific cell ranges).

API Reference, service(s)

com.sun.star.sheet.SheetCellCursor

Parent object(s)
Sheet

Methods

All methods of CellRange plus:

collapseToCurrentArray
Collapses the cursor into the range of the array formula to which it is currently pointing.

collapseToCurrentRegion
Expands the cursor into the region containing the cells to which it currently points.

collapseToMergedArea
Expands the cursor to merged cell ranges.

collapseToSize(<l_Columns>, <l_Rows>)
Changes the size of a cursor range.

expandToEntireColumns
Expands the cursor to include the entire columns of the cells to which it is currently pointing.

expandToEntireRows
Expands the cursor to include the entire rows of the cells to which it is currently pointing.

gotoEnd
Points the cursor to a single cell which is the end of a contiguous series of (filled) cells.

gotoEndOfUsedArea(<b_Expand>)
Points the cursor to the end of the used area and expands the range or collapses to a single cell.

gotoNext
Points the cursor to the next unprotected cell.

gotoOffset(<l_ColumnOffset>,<l_RowOffset>)
Moves the origin of the cursor relative to the current position.

gotoPrevious
Points the cursor to the previous unprotected cell.

gotoStart
Points the cursor to a single cell which is the beginning of a contiguous series of (filled) cells.

gotoStartOfUsedArea(<b_Expand>)
Points the cursor to the start of the used area and expands the range or collapses to a single cell.

Properties

All properties of CellRange.

CellFormatRanges(*)

This represents a collection of equal-formatted cell ranges.

All cells inside a cell range of this collection have the same formatting attributes.

API Reference, service(s)
com.sun.star.sheet.CellFormatRanges

Parent object(s)
CellRange

Methods

createEnumeration()
Returns an enumerated list of all charts in the collection.

getByIndex(<l_index>)

Object (CellRanges) - returns a collection of ranges with the same format.

HasElements
Boolean – true if there are any charts in the collection.

Properties

Count
Integer – the number of sheets in the collection

CellProperties

Contains the properties of a table cell.

Api reference, Service
com.sun.star.table.CellProperties

Properties

AsianVerticalMode
Boolean - selects Asian character orientation in vertical orientation.

BottomBorder
Struct (BorderLine) - contains a description of the bottom border line of each cell.

CellStyle
String - contains the name of the style of the cell.

CellBackColor
Const (Color) – background color of the table cell.

CellProtection
Struct (CellProtection) - contains a description of the cell protection. Cell protection is active only if the sheet is protected.

HoriJustify
Enum (CellHoriJustify) – horizontal justification of the text

IsCellBackgroundTransparent
Boolean – sets the background color to transparent.

IsTextWrapped
Boolean – permits automatic line breaks within the cell

LeftBorder
Struct(BorderLine) - contains a description of the left border line of each cell.

NumberFormat
Long - contains the index of the number format that is used in the cells.

Orientation
Enum (CellOrientation) – orientation of text .

Paraindent
Short - defines the indentation of the cell contents (in 1/100 mm). Does not seem to do anything with a spreadsheet cell.

RightBorder
Struct (BorderLine) - contains a description of the right border line of each cell.

RotateAngle
Long – angle of rotation of text in hundredths of a degree

RotateReference
Enum (CellVertJustify) - defines at which edge rotated cells are aligned.

ShadowFormat
Struct (ShadowFormat) – specifies the shadow for cells.

TableBorder
Struct (TableBorder) – contains a description of the cell or cell range border.

TopBorder
Struct (BorderLine) - contains a description of the top border line of each cell.

UserDefinedAttributes
Object (UserDefinedAttributes) - stores additional attributes.
This property is used i.e. by the XML filters to load and restore unknown attributes.

VertJustify
Enum (CellVertJustify) - vertical justification of the text.

CellProtection

A structure that describes the kind of protection for a protectable cell.

API Reference, service(s)
Enumeration: com.sun.star.util.CellProtection

Elements

IsLocked
Boolean - specifies if the cell is locked from modifications by the user.

IsFormulaHidden
Boolean - specifies if the formula is hidden from the user.

IsHidden
Boolean - specifies if the cell is hidden from the user.

IsPrintHidden
Boolean - specifies if the cell is hidden on printouts.

CellRange

This object is a range of cells (or just one !) and is treated in a similar way as the Cell object with most of the properties being the same. Take care here since you can have ranges of ranges:

oRange = oSheet.getCellRangeByName("C3:E5")

This returns the range C3:E5 on the **sheet.**

oSubRange = oRange.getCellRangeByName("B2")

This will fail since cell B2 is outside the parent range. This is different from Excel where it would return the sheet cell D4. Beware!

oSubRange = oRange.getCellRangeByName("D4")

This returns a range (cell) that is D4 on the **sheet** and within the parent range C3:E5

oSubRange = oRange.getCellByPosition(1,1)

This returns the range (cell) that is D4 on the sheet. This is the same behaviour as Excel.

API Reference, service(s)
com.sun.star.sheet.SheetCellRange

Parent object(s)
Sheet, CellRange

Methods

applySubTotals(<o_SubTotalDescriptor>, <b_Replace>)
Creates subtotals using the settings of the passed descriptor.

autoformat(<s_FormatName>)
Applies an autoformat to the range.

clearContents(<const_CellFlags>)
Clears the cell contents as specified by the flags

computeFunction(<enum_GeneralFunction>)
Performs the specified function on the cells in the range.

createFilterDescriptor

Object (SheetFilterDescriptor) - creates a sheet filter descriptor.

createFilterDescriptorByObject (<o_SheetFilterable>)
Object - creates a *SheetFilterDescriptor* for the specified filterable object from the contents of this object.
*** the parameter sheetfilterable references a sheetfilterdescriptor.....these references go round in circles. More investigation needed.

createImportDescriptor(<b_?>)
Sequence (DatabaseImportDescriptor) – returns descriptor for importing external data. I have no idea what the boolean value does.

createReplaceDescriptor
Object (ReplaceDescriptor) – returns a replace descriptor object that specifies a search.

createSortDescriptor
Sequence (SheetSortDescriptor) – returns a sheet sort descriptor property group. See notes about sorting.

createSubTotalDescriptor(<b_Empty>)
Object (SubTotalDescriptor) - creates a sub total descriptor, if it is not empty then it is filled with the previous settings.

decrementIndent
Shifts the indentation by one default step to the left.

doimport(<seq_DatabaseImportDescriptor>)
Imports data into the range

fillSeries(<enum_FillDirection>, <enum_FillMode>, <enum_FillDateMode, <f_step>, <f_endvalue>)
Fills all cells in the range based on the specified settings.
<f_Step> - contains the value used to increase/decrease the series values.
<f_EndValue> - contains the threshold value on which the calculation of the current series stops.

fillAuto(<enum_FillDirection>, <l_CellsToFill>)
Fills all cells in the range in a way that is specified by the first cell(s) in the range.

filter(<o_SheetFilterDescriptor>)
Performs a filter operation, using the settings of the passed filter descriptor.

getCellByPosition(<l_Column>, <l_Row>)
Returns the cell object at the specified position relative to the parent range.

getCellRangeByName(<s_Address>)
Returns a sub-range of cells within the range.

getCellRangeByPosition(<l_Left>,<l_Top>,<l_Right>,<l_Bottom>
Returns a sub-range of cells within the range relative to the parent range.

incrementIndent
Shifts the indentation by one default step to the right.

isNotANumber(<d_Anumber>)
checks whether the value given is equal to the indicator value for a missing value.

merge
Merges/unmerges the area specified by this object.

queryColumnDifferences
Object (CellRanges) - queries all cells with different values in a specified row.

queryContentCells
Object (CellRanges) - queries all cells with the specified content type(s).

queryDependents
Object (CellRanges) - queries all dependent formula cells.

queryEmptyCells
Object (CellRanges) - queries all empty cells.

queryFormulaCells
Object (CellRanges) - queries all formula cells with the specified result type.

queryIntersection

Object (CellRanges) - intersects the current cell range(s) with the specified cell range.

queryPrecedents
Object (CellRanges) - queries all precedent cells.

queryRowDifferences
Object (CellRanges) - queries all cells with different values in a specified column.

queryVisibleCells
Object (CellRanges) - queries all visible cells.

removeSubTotals
removes the subtotals from the current range.

replaceall(<o_ReplacesDescriptor>)
Searches for all described text in the range of cells.

sort(<seq_SheetSortDescriptor>)
Sorts the range of data.

Properties

Property Group(s) : CellProperties, CharacterProperties, CharacterPropertiesAsian, CharacterPropertiesComplex , ParagraphProperties

CellFormatRanges
Object - returns a collection of equal-formatted cell ranges.

ColumnDescriptions
String array - alist of descriptions for the columns.

Columns
Object – collection of columns

ConditionalFormat
Object (ConditionalFormat) - contains the conditional formatting settings for this range.

ConditionalFormatLocal
Object (ConditionalFormatLocal) -contains the conditional formatting settings for this range, using localized formulas.

DataArray
Array of variant Arrays – the inner array is a list of double and/or string data for a row.

Data
Array of Float Arrays – the inner array is a list of float data for a row.

```
Dim oSheet as Object
Dim vRowArrays(4) as Variant
Dim fData(1) As Double
oSheet = ThisComponent.CurrentController.ActiveSheet
vRowArrays = oSheet.getCellRangeByName("A1:B5").Data
fData = vRowArrays(1)
Print fData(1)
```

NB that the data does not have to be charted for this to work. You can use this method to extract the contents of a range of data.

FormulaArray
Array of string arrays – the inner array (row) is a list of formulas or values in string format if there is no formula.

IsMerged
Boolean - true if the area specified by this object is merged, or false otherwise.

NotANumber
Double - returns the value which is to be used as an indicator for a missing value in the data.

Position
Structure (Point) - contains the top left position of this range in the sheet (in 1/100 mm).

RangeAddress
Structure (CellRangeAddress) - returns the address of the range.

RowDescriptions
String array - alist of descriptions for the rows.

Rows

Object – returns a collection of Rows

Size
Structure (Size) - contains the size of this range (in 1/100 mm).

Spreadsheet
Object (Sheet) – returns the *sheet* that this range is from.

UniqueCellFormatRanges
Object - returns a collection of equal-formatted cell range collections.

Validation
Object (Validation) - contains the data validation settings for this range.

ValidationLocal
Object (ValidationLocal)contains the data validation settings for this range, using localized formulas.

CellRangeAddress

This is a structure that defines a cell range.

API Reference, structure
com.sun.star.table.CellRangeAddress

Elements

Sheet
Short - number of the sheet (numbering begins with 0).

StartColumn
Long – first column in the cell range (numbering begins with 0).

StartRow
Long – first row in the cell range (numbering begins with 0).

EndColumn
Long – final column in the cell range (numbering begins with 0).

EndRow
Long – final row in the cell range (numbering begins with 0).

CellRanges

Represents a collection of cell ranges in a spreadsheet document.

API Reference, service(s)
com.sun.star.sheet.SheetCellRanges

Parent object(s)
Sheet , UniqueCellFormatRanges

Methods

addRangeAddress(<struct_CellRangeAddress> , <b_MergeRanges>)
adds the given range to the collection of cell ranges. To merge the ranges takes more time, but the memory usage is lower.

addRangeAddresses(<struct_CellRangeAddress array> , <b_MergeRanges>)
adds the given ranges to the collection of cell ranges.

clearContents(<const_CellFlags>)
Clears the cell contents as specified by the flags

computeFunction(<enum_GeneralFunction>)
Performs the specified function on the cells in the range.

createEnumeration
Enumerated list of cell ranges.

createReplaceDescriptor
Object (ReplaceDescriptor) – returns a replace descriptor object that specifies a search.

decrementIndent
Shifts the indentation by one default step to the left.

incrementIndent
Shifts the indentation by one default step to the right.

queryColumnDifferences
Object (CellRanges) - queries all cells with different values in a specified row.

queryContentCells
Object (CellRanges) - queries all cells with the specified content type(s).

queryDependents
Object (CellRanges) - queries all dependent formula cells.

queryEmptyCells
Object (CellRanges) - queries all empty cells.

queryFormulaCells
Object (CellRanges) - queries all formula cells with the specified result type.

queryIntersection
Object (CellRanges) - intersects the current cell range(s) with the specified cell range.

queryPrecedents
Object (CellRanges) - queries all precedent cells.

queryRowDifferences
Object (CellRanges) - queries all cells with different values in a specified column.

queryVisibleCells
Object (CellRanges) - queries all visible cells.

removeByName(<s_CellRangeName>)
Removes the cell range with the specified name.

removeRangeAddress(<struct_CellRangeAddress>)
removes the given range from the collection of cell ranges.

removeRangeAddresses(<struct_CellRangeAddress array>)
removes the given ranges from the collection of cell ranges.

replaceall(<o_ReplacesDescriptor>)
Searches for all described text in the range of cells.

Properties

Property Group(s) : CellProperties, CharacterProperties, CharacterPropertiesAsian, CharacterPropertiesComplex , ParagraphProperties

ColumnDescriptions
String array - alist of descriptions for the columns.

ConditionalFormat
Object (SheetConditionalEntries) - contains the conditional formatting settings for this range.

ConditionalFormatLocal
Object (SheetConditionalEntries) -contains the conditional formatting settings for this range, using localized formulas.

Data
Array of Float Arrays – the inner array is a list of float data for a row.

```
Dim oSheet as Object
Dim vRowArrays(4) as Variant
Dim fData(1) As Double
oSheet = ThisComponent.CurrentController.ActiveSheet
vRowArrays = oSheet.getCellRangeByName("A1:B5").Data
fData = vRowArrays(1)
Print fData(1)
```

NB that the data does not have to be charted for this to work. You can use this method to extract the contents of a range of data.

RowDescriptions
String array - alist of descriptions for the rows.

Validation

Object (Validation) - contains the data validation settings for this range.

ValidationLocal
Object (ValidationLocal)contains the data validation settings for this range, using localized formulas.

CellStyle

An object that defines a spreadsheet cell style.

API Reference, service(s)
com.sun.star.sheet.TableCellStyle

Parent object(s)
StyleFamily or CellStyles

Properties

Property Groups : CellProperties, CharacterProperties, CharacterPropertiesComplex , CharacterPropertiesAsian , ParagraphProperties

CellStyles(*)

A collection of cellstyle objects.

API Reference, service(s)
com.sun.star.style.StyleFamily

Parent object(s)
StyleFamilies

Methods

getByName("<s_stylename>")
Object (CellStyle) - returns the style name object.

getByIndex(<l_Index>)
Object (CellStyle) – returns the style object at the specified zero based index.

hasByName("<s_stylename>")
Boolean – true if the collection has the named style.

hasElements
Boolean – true if the collection has styles.

insertByName(<s_StyleName, <o_Style>)
Inserts the style with the specified name.

replaceByName(<s_styleName>, <o_Style)
Replaces the named style with the new style.

removeByName(<s_styleName>)
Removes the style with the specified name.

Properties

Count
Long – returns the number of styles in the collection.

ElementNames
String array – returns a list of all the style objects.

CharacterProperties

This is a set of properties to describe the style of characters.

Api reference, Service
com.sun.star.style.CharacterProperties

Properties

CharFontName
string - This property specifies the name of the font style.

CharFontStyleName
string - This property contains the name of the font style.

CharFontFamily
Enum (FontFamily) - This property contains font family as specified in .

CharFontCharSet
Enum (Charset) - This property contains the text encoding of the font. DEPRECATED

CharFontPitch
Enum(FontPitch) - This property contains the font pitch.

CharColor
Enum (Color) - This property contains the value of the text color.

CharEscapement
Short - specifies the percentage of raisement/lowerment of superscript/subscript characters.

CharHeight
Float - This value contains the height of the characters in point.

CharUnderline
Const (FontUnderline) - This property contains the value for the character underline.

CharWeight
Const (FontWeight) - This property contains the value of the font weight.

CharPosture
Enum (FontSlant) - This property contains the value of the posture of the document.

CharAutoKerning
Boolean - optional property to determine whether the kerning tables from the current font are used.

CharBackColor
Const (Color) - optional property which contains the text background color.

CharBackTransparent
Boolean - determines if the text background color is set to transparent.

CharCaseMap
Short - optional property which contains the value of the case-mapping of the text for formatting and displaying.

CharCrossedOut
Boolean - This property is true if the character(s) is(are) crossed out.

CharFlash
Boolean - If this optional property is true , then the characters are flashing.

CharStrikeout
? (FontStrikeout) - determines the type of the strike out of the character.

CharWordMode
Boolean - If this property is true , the underline and strike-through properties are not applied to white spaces.

CharKerning
Short - optional property which contains the value of the kerning of the characters.

CharLocale
Struct (Locale) - contains the value of the locale.

CharKeepTogether
Boolean - optional property which marks a range of characters to prevent it from being broken into two lines.

CharNoLineBreak
Boolean - optional property which marks a range of characters to ignore a line break in this area.

CharShadowed

Boolean - specifies if the characters are formatted and displayed with a shadow effect.

CharFontType
Const (FontType) - optional property which specifies the fundamental technology of the font.

CharStyleName
String - specifies the name of the style of the font.

CharContoured
Boolean - specifies if the characters are formatted and displayed with a contour effect.

CharCombineIsOn
Boolean - determins whether text is formatted in two lines.

CharCombinePrefix
String - contains the prefix (usually parenthesis) before text that is formatted in two lines.

CharCombineSuffix
String - contains the suffix (usually parenthesis) after text that is formatted in two lines.

CharEmphasis
Const (FontEmphasis) - contains the font emphasis value.

CharRelief
Const (FontRelief) - contains the relief value.

RubyText
String - contains the text that is set as ruby.

RubyAdjust
Enum (RubyAdjust) - determines the adjustment of the ruby text.

RubyCharStyleName
String - contains the name of the character style that is applied to RubyText.

RubyIsAbove
Boolean - determines whether the ruby text is printed above/left or below/right of the text.

CharRotation
Short - determines the rotation of a character in degree.

CharRotationIsFitToLine
Boolean - determines whether the text formatting tries to fit rotated text into the surrounded line height.

CharScaleWidth
Short - determines the percentage value for scaling the width of characters.

HyperLinkURL
String - contains the URL of a hyperlink (if set).

HyperLinkTarget
String - contains the name of the target for a hyperlink (if set).

HyperLinkName
String - contains the name of the hyperlink (if set).

VisitedCharStyleName
String - contains the character style name for visited hyperlinks.

UnvisitedCharStyleName
String - contains the character style name for unvisited hyperlinks.

CharEscapementHeight
Byte - This is the additional height used for subscript or superscript characters in units of percent. For subscript characters the value is negative and for superscript characters positive.

CharNoHyphenation
Boolean - determines if the word can be hyphenated at the character.

CharUnderlineColor
Const (Color) - gives the color of the underline for that character.

CharUnderlineHasColor
Boolean - specifies if the CharUnderlineColor is used for an underline.

CharStyleNames
Seq (String) - specifies the names of the all styles applied to the font.

CharHidden
Boolean - If this optional property is true , then the characters are invisible.

TextUserDefinedAttributes
? (XNameContainer) - this property stores xml attributes. They will be saved to and restored from automatic styles inside xml files.

CharacterPropertiesAsian

This is a set of properties to describe the style of characters in asian texts.

ApI reference, Service
com.sun.star.style.CharacterPropertiesAsian

Properties

CharFontCharSetAsian
Short - This property contains the text encoding of the font as specified in com.sun.star.awt.CharSet. This is a deprecated set of constants so is the property deprecated as well?

CharFontFamilyAsian
Constant (FontFamily) - This property contains font family.

CharFontNameAsian
String - This property specifies the name of the font style.

CharFontPitchAsian
Constant (FontPitch) - This property contains the font pitch.

CharFontStyleNameAsian
String - This property contains the name of the font style.

CharHeightAsian
Float - This value contains the height of the characters in point.

CharLocaleAsian
Structuer (Locale) - contains the value of the locale.

CharPostureAsian
Constant (FontSlant) - This property contains the value of the posture of the document.

CharWeightAsian
Float - This property contains the value of the font weight.

CharacterPropertiesComplex

ApI reference, Service
com.sun.star.style.CharacterPropertiesComplex

Properties

CharHeightComplex
Float - This value contains the height of the characters in point.

CharWeightComplex
Float - This property contains the value of the font weight.

CharFontNameComplex
String - This property specifies the name of the font style.

CharFontStyleNameComplex
String - This property contains the name of the font style.

CharFontFamilyComplex
Constant (FontFamily) - This property contains font family

CharFontCharSetComplex
Short - This property contains the text encoding of the font as specified in com.sun.star.awt.CharSet. These are deprecated constants so this property may also be

deprecated?

CharFontPitchComplex
Constant (FontPitch) - This property contains the font pitch.

CharPostureComplex
Enumeration (FontSlant) - This property contains the value of the posture of the document.

CharLocaleComplex
Structure (Locale) - contains the value of the locale.

Chart

This object is a chart based on the data in a table or sheet.

API Reference, service(s)
com.sun.star.table.TableChart

Parent object(s)
Charts

Properties

EmbeddedObject
Object (ChartDocument) – returns the chart document for the chart

HasColumnHeaders
Boolean - specifies whether the cells of the topmost row of the source data are interpreted as column headers.

HasRowHeaders
Boolean - specifies whether the cells of the leftmost row of the source data are interpreted as row headers.

Name
String – the internal name of the chart **not** the name you provide via the name dialog.

Ranges
Structure (CellRangeAddress) array – list of all the range data for the chart.

ChartArea

An object that specifies the area elements of a chart, e.g. the background area, the diagram wall, and the diagram floor of three-dimensional charts.

API Reference, service(s)
com.sun.star.chart.ChartArea

Parent object(s)
ChartDocument

Properties

Property Group(s) : FillProperties , LineProperties

Position
Structure (Point) - the position of the area within the chart

Size
Structure (Size) - the size of the area.

ChartAxis

An object that specifies axes in a diagram.

Note: The text properties correlate to all axis description elements, not to just a single text element.

API Reference, service(s)

com.sun.star.chart.

Parent object(s)
Diagram

Properties

Property Groups : CharacterProperties , LineProperties

ChartDataArray

It contains the data values as well as the descriptions for each row and column.

API Reference, service(s)
com.sun.star.chart.ChartDataArray

Parent object(s)
ChartDocument

Methods

isNotANumber(<d_Anumber>)
checks whether the value given is equal to the indicator value for a missing value.

Properties

ColumnDescriptions
String array - alist of descriptions/labels for the columns.

Data
Array of Float Arrays – the inner array is a list of float data for a row.

NotANumber
Double - Returns the value which is to be used as an indicator for a missing value in the data.

RowDescriptions
String array - alist of descriptions/labels for the rows.

ChartDataPointProperties

API Reference, service(s)
com.sun.star.chart.ChartDataPointProperties

Properties

Property Groups : FillProperties , LineProperties , CharacterProperties , Chart3dProperties

DataCaption
Constant (ChartDataCaption) - specifies how the captions of data points are displayed.

SegmentOffset
Long - the offset by which pie segments in a PieDiagram are dragged outside from the center. This value is given in percent of the radius.

SymbolType
Long - specifies the type of symbols if the current chart type supports the usage of symbols.

SymbolBitmapURL
String - In charts that support symbols, you can set this property to any valid URL that points to a graphic file. This graphic is then used as symbol for each data point.

ChartDataRowProperties

API Reference, service(s)
com.sun.star.chart.ChartDataRowProperties

Properties

Property Groups : FillProperties , LineProperties , CharacterProperties , Chart3dProperties

Axis
Long - determines to which axis the data row is assigned.

ConstantErrorHigh
Double - specifies the upper limit of the error range of a data row.

ConstantErrorLow
Double - specifies the lower limit of the error range of a data row.

DataCaption
Constant (ChartDataCaption) - specifies how the captions of data points are displayed.

DataErrorProperties
Object (LineProperties) - holds the properties of the error markers, if those are enabled.

DataMeanValueProperties
Object (LineProperties) - holds the properties of the average line, if such one is enabled.

DataRegressionProperties
Object (LineProperties) - holds the properties of the regression line, if such one is enabled.

ErrorCategory
Enumeration (ChartErrorCategory) - determines the type of error to indicate.

ErrorIndicator
Enumeration (ChartErrorIndicatorType) - determines how the error is indicated.

ErrorMargin
Double - specifies the percentage for the margin of errors.

MeanValue
Boolean - determines if the mean value for a data row is displayed as a line.

PercentageError
Double - specifies the percentage that is used to display error bars.

RegressionCurves
Enumeration (ChartRegressionCurveType) - determines a type of regression for the data

SegmentOffset
Long - the offset by which pie segments in a PieDiagram are dragged outside from the center. This value is given in percent of the radius.

SymbolType
Long - specifies the type of symbols if the current chart type supports the usage of symbols.

SymbolBitmapURL
String - In charts that support symbols, you can set this property to any valid URL that points to a graphic file. This graphic is then used as symbol for each data point.

ChartDocument

A chart document object consists of a reference to the data source, the diagram and some additional elements like a main title, a sub-title or a legend.

This has the same calls as a document for loading and storing.

API Reference, service(s)
com.sun.star.chart.ChartDocument

Parent object(s)
Chart

Properties

AdditionalShapes
Object -

AllowSetModified
Boolean -

Area
Object (ChartArea) – returns a chart area object that describes the properties of the background area of the chart document.

CategoriesRangeAddress
String - contains the address to the cells containing the names of the categories. Note: Each value of a data series belongs exactly to one category.

ChartRangeAddress
String -

Data
Object (ChartDataArray) - the data source of the chart.

DataSourceLabelsInFirstColumn
Boolean - specifies whether the cells of the leftmost row of the source data are interpreted as row headers. (Same result as ChartDocument.HasRowHeaders)

DataSourceLabelsInFirstRow
Boolean - specifies whether the cells of the topmost row of the source data are interpreted as column headers. (Same result as ChartDocument.HasColumnHeaders)

Diagram
Object (Diagram) – the plot area of the chart.

Drawpage
Object (Drawpage) – returns a single drawpage which is a collection of *shapes.*

ExportData
Boolean -

HasLegend
Boolean - determines if the legend is shown or hidden.

HasMainTitle
Boolean - determines if the main title is shown or hidden.

HasSubTitle
Boolean - determines if the subtitle is shown or hidden.

HasTranslatedColumns
Boolean -

HasTranslatedRows
Boolean -

Legend
Object (ChartLegend) – returns a chart legend object to describe the legend

NumberFormats
Object (NumberFormats) - returns the collection of number formats belonging to this object.

NumberFormatSettings
Object (NumberFormatSettings) – return the number format settings of this object.

TableNumberList
String -

Title
Object (ChartTitle) – returns a chart title object to describe the main title

TranslatedColumns
Array -

TranslatedRows
Array -

SeriesAddresses
Structure (ChartSeriesAddress) Array - contains the addresses to the elements of a series. This sequence should contain one element for each series in the chart.

SubTitle
Object (ChartTitle) – returns a chart title object to describe the sub title

ChartLegend

An object that specifies the legend in a chart.

API Reference, service(s)
com.sun.star.chart.ChartLegend

Parent object(s)
ChartDocument

Properties

Property Group(s) : FillProperties , LineProperties , CharacterProperties, CharacterPropertiesAsian

Alignment
Enumeration (ChartLegendPosition) - the position of the legend within the chart

Position
Structure (Point) - the position of the text area within the chart

Size
Structure (Size) - the size of the text area containing the title

Charts(*)

API Reference, service(s)
com.sun.star.table.TableCharts

Parent object(s)
Sheet

Methods

addNewByName(<s_InternalName>, <struct_Rectangle>, <structarray_CellRangeaddress>, <b_ColumnHeaders>, <b_RowHeaders>)
creates a chart and adds it to the collection. The rectangle size is in 1/100th mm.

createEnumeration()
Returns an enumerated list of all charts in the collection.

getByIndex(<l_index>)
Object (Chart) – returns the chart at the specified zero based index.

getByName(<s_InternalName>)
Object (Chart) – returns the chart with the matching internal name.

HasByName(<s_InternalName>)
Boolean – true if the collection has a chart of that internal name.

HasElements
Boolean – true if there are any charts in the collection.

removeByName(<s_InternalName>)
Removes the named chart.

Properties

Count
Long – The number of columns in the collection.

ElementNames
String Array - returns a list of all the internal chart names NB not the name you give it!

ChartSeriesAddress

This structure describes a single data row, specified by its name and a sequence of data points.

The cell addresses are in the format of the application that contains this chart.

API Reference, structure

com.sun.star.chart.ChartSeriesAddress

Elements

DataRangeAddress
String - contains the cell range address of the data for this series.

LabelAddress
String - contains the cell address of label (i.e. name) of this series.

DomainRangeAddresses
String Array - contains cell addresses for each domain of this series.

ChartTitle

An object that specifies titles in a chart.

In a chart there may be the following titles: the main title, the sub title, and axis titles of the x- and y-axis.

API Reference, service(s)
com.sun.star.chart.ChartTitle

Parent object(s)
ChartDocument

Properties

Property Group(s) : FillProperties , LineProperties , CharacterProperties, CharacterPropertiesAsian

Position
Structure (Point) - the position of the text area within the chart

TextRotation
Long - specifies the rotation of the shape in 100th of degrees.

Size
Structure (Size) - the size of the text area containing the title

StackedText
Boolean – determines if the text is stacked

String
String - contains the text of the title.

ClosedBezierShape

This object is for a closed bezier shape.

API Reference, service(s)
com.sun.star.drawing.CloseBezierShape

Parent Objects
Drawpage, GroupShape, Shapes

Properties

Property Groups : Shape , LineProperties , Fillproperties , Text , ShadowProperties , PolypolygonBezierDescriptor

Column

An object that represents a single column.

API Reference, service(s)
com.sun.star.table.TableColumn

Parent object(s)

Rows

Methods

Refer to the methods of a CellRange

Properties

Refer to the properties of a CellRange

IsStartOfNewPage
Boolean - is true , if there is a manual horizontal page break attached to the column.

IsVisible
Boolean - is true , if the column is visible.

Name
String – the column name.

OptimalHeight
Boolean - is true , if the column always keeps its optimal width.

Width
Long - contains the width of the column (in 1/100 mm).

Columns(*)

This is a collection of column objects.

API Reference, service(s)
com.sun.star.table.TableColumns

Parent object(s)
Sheet,Cell,CellRange

Methods

createEnumeration()
Returns an enumerated list of all columns in the collection.

getByIndex(<l_index>)
Object (Column) – returns the column at the specified zero based index.

getByName(<s_ColumnName>)
Object (Column) – returns the column with the matching name.

HasByName(<s_ColumnName>)
Boolean – true if the collection has a column of that name.

insertByIndex(<l_Position>, <l_Quantity>)
Inserts <quantity> columns at the zero based <position>

removeByIndex(<l_Position>, <l_Quantity>)
Removes <quantity> columns starting with the zero based <position>

Properties

Count
Long – The number of columns in the collection.

ElementNames
String Array - returns a list of all the column names.

ColumnLabelRange

An object that represents a cell area that contains labels and values related to the column labels.

API Reference, service(s)
com.sun.star.sheet.LabelRange

Parent object(s)
ColumnLabelRanges

Methods

getLabelArea
Structure (CellRangeAddress) - returns the cell range address that contains the labels.

SetLabelArea(<struct_CellRangeAddress>)
sets the cell range address that contains the labels.

getDataArea
Structure (CellRangeAddress) - returns the cell range address for which the labels are valid.

setDataArea(<struct_CellRangeAddress>)
sets the cell range address for which the labels are valid.

ColumnLabelRanges(*)

A collection of column label ranges in the document.

API Reference, service(s)
com.sun.star.sheet.XLabelRanges

Parent object(s)
Document

Methods

addNew(<struct_CellRangeAddress-titlearea>,<struct_CellRangeAddress-dataarea>
Adds a new label range to the collection

createEnumeration
Returns and enumerated list of column label ranges.

getByIndex(<l_Index>)
Object (ColumnLabelRange) – returns the column label range at the specified zero based index.

removeByIndex(<l_Index>)
Removes a label range from the collection.

Properties

Count
Long – returns the number of column label ranges in the collection.

ConditionalFormat (*)

This is a collection of conditional formats.

API Reference, service(s)
com.sun.star.sheet.SheetConditionalEntries

Parent Objects
Cell , CellRange

Methods

addNew(<struct_ConditionalEntry>)
Adds a conditional entry to the format.

clear
Clears all conditional entries.

createEnumeration
Returns an enumerated list of conditional formats (TableConditionalEntry).

getbyIndex(<l_Index>)

Object (TableConditionalEntry) - returns the conditional entry at the zero based index in the collection.

getbyName(<s_FormatName>)
Object (TableConditionalEntry) - returns the conditional entry with the matching name.

hasByName
Boolean – true if the collection contains the named condition.

hasElements
Boolean – true if there are any elements in the collection.

removeByIndex(<l_Index>)
removes the conditionat the specified index in the collection.

Properties

Count
Integer – the number of conditions in the collection

ElementNames
String Array – list of conditional format names

ConditionalFormatLocal (*)

This is a collection of conditional formats using local formula.

API Reference, service(s)
com.sun.star.sheet.SheetConditionalEntries

Parent Objects
Cell , CellRange

Methods

addNew(<struct_ConditionalEntry>)
Adds a conditional entry to the format.

clear
Clears all conditional entries.

createEnumeration
Returns an enumerated list of conditional formats (TableConditionalEntry).

getbyIndex(<l_Index>)
Object (TableConditionalEntry) - returns the conditional entry at the zero based index in the collection.

getbyName(<s_FormatName>)
Object (TableConditionalEntry) - returns the conditional entry with the matching name.

hasByName
Boolean – true if the collection contains the named condition.

hasElements
Boolean – true if there are any elements in the collection.

removeByIndex(<l_Index>)
removes the conditionat the specified index in the collection.

Properties

Count
Integer – the number of conditions in the collection

ElementNames
String Array – list of conditional format names

ConnectorShape

An object that describes

API Reference, service(s)
com.sun.star.drawing.ConnectorShape

Parent Objects
Drawpage, GroupShape, Shapes

Properties

Property Groups: Shape , LineProperties , Text , ShadowProperties

EndGluePointIndex
Long - this is the position of the connectors end point in 100th/mm. For unconnected end points you can get and set the position. For connected end points you can only get the position.

EdgeKind
Enumeration (ConnectorType) - This property contains the kind of the connector.

EdgeLine1Delta
Long - This property contains the distance of line 1.

EdgeLine2Delta
Long - This property contains the distance of line 2.

EdgeLine3Delta
Long - this property contains the distance of line 3.

EndPosition
Structure (Point) - this property holds the index of the glue point the end point of this connector is glued on.

EdgeNode1HorzDist
Long - This property contains the horizontal distance of node 1.

EdgeNode1VertDist
Long - This property contains the vertical distance of node 1.

EdgeNode2HorzDist
Long - This property contains the horizontal distance of node 2.

EdgeNode2VertDist
Long - This property contains the vertical distance of node 2.

EndShape
Object (Shape) - this property either holds the shape that the end point of this connector is connected to, or is empty when the end point of the connector is not connected to a shape.

StartGluePointIndex
Long - this property holds the index of the glue point the start point of this connector is glued on.

StartPosition
Structure (Point) - this is the position of the connectors start point in 100th/mm. For unconnected start points you can get and set the position. For connected start points you can only get the position.

StartShape
Object (Shape) - this property either holds the shape that the start point of this connector is connected to, or is empty when the start point of the connector is not connected to a shape.

Controller

This is, in essence, the window object that is displaying the selected document. As such all the user viewable methods and properties are available via this object.

API Reference, service(s)
com.sun.star.sheet.SpreadsheetView

Parent object(s)
Document

Methods

createEnumeration
Creates an enumerated list of panes.

freezeAtPosition(<l_columns>,<l_rows>)
Freezes panes with the specified number of columns and rows. To freeze only horizontally, specify rows as 0. To freeze only vertically, specify columns as 0.

getByIndex(<l_Index>)
Object – returns the window pane at the specified index

hasFrozenPanes
Boolean - returns true if the view has frozen panes.

select(<o_various>)
Boolean - Selects the object if possible.

setActiveSheet (<o_Sheet>)
sets the sheet that is shown in the view.

setFirstVisibleColumn(<l_Column>)
sets the first column that is visible in the pane. I can't get this to work for a single pane window.

setFirstVisibleRow(<l_Row>)
sets the first row that is visible in the pane. I can't get this to work for a single pane window.

splitAtPosition(<l_Xpixel>,<l_ypixel>)
Splits the view at the specified position. To split only horizontally, specify nPixelY as 0. To split only vertically, specify nPixelX as 0.

Properties

ActiveSheet
Object (Sheet) - returns the sheet that is shown in the view.

Count
Long – returns the number of panes in the window view

FirstVisibleColumn
Long - returns the first column that is visible in the pane.

FirstVisibleRow
Long - returns the first row that is visible in the pane.

GridColor
Const (Color) - specifies the color in which the cell grid is displayed.

HasColumnRowHeaders
Boolean - enables the column and row headers of the view.

HasHorizontalScrollBar
boolean - enables the horizontal scroll bar of the view.

HasSheetTabs
boolean - enables the sheet tabs of the view.

HasVerticalScrollBar
Boolean - enables the vertical scroll bar of the view.

HideSpellMarks
Boolean - disables the display of marks from online spelling.

IsOutlineSymbolsSet
Boolean - enables the display of outline symbols.

IsValueHighlightingEnabled
Boolean - controls whether strings, values, and formulas are displayed in different colors.

IsWindowSplit
Boolean - returns true if the view is split into individual panes.

ReferredCells
Object (CellRange) – returns the range represented in the window

Selection
Object (context dependant) – returns the selected object in the window. This can be a shape, cell, range, chart etc.

ShowAnchor
Boolean - enables display of anchor symbols when drawing objects are selected.

ShowCharts
Short - enables the display of charts in the view.

ShowDrawing
Short - enables the display of drawing objects in the view.

ShowFormulas
boolean - controls whether formulas are displayed instead of their results.

ShowGrid
Boolean - enables the display of the cell grid.

ShowHelpLines
Boolean - enables display of help lines when moving drawing objects.

ShowNotes
Boolean - controls whether a marker is shown for notes in cells

ShowObjects
Short - enables display of embedded objects in the view.

ShowPageBreaks
Boolean - enables display of page breaks.

ShowZeroValues
boolean - enables display of zero-values.

SolidHandles
Boolean - enables solid (colored) handles when drawing objects are selected.

SplitColumn
Long - returns the column before which the view is split.

SplitHorizontal
Long - returns the horizontal position in pixels where the view is split.

SplitRow
Long - returns the row before which the view is split.

SplitVertical
Long - returns the vertical position in pixels where the view is split.

VisibleRange
Structure (CellRangeAddress) - returns the address of the cell range that consists of the cells which are visible in the pane.

ZoomType
Short - This property defines the zoom type for the document.

ZoomValue
Short - Defines the zoom value to use. Valid only if the ZoomType is set to BY_VALUE .

ControlShape

An object that describes contains a control.

API Reference, service(s)
com.sun.star.drawing.ControlShape

Parent Objects
Drawpage, GroupShape, Shapes

Properties

Property Groups: Shape

Control
Object (FormControl) - defines the control model of this Shape .

CustomShape

An object that describes a custom shape.

API Reference, service(s)
com.sun.star.drawing.CustomShape

Parent Objects
Drawpage, GroupShape, Shapes

Properties

Property Groups: Shape , LineProperties , Fillproperties , Text , ShadowProperties , RotationDescriptor

CustomShapeData
String - This property can be used to store data that the CustomShapeEngine may use for rendering

CustomShapeEngine
String - This property contains the CustomShapeEngine service name that has to be used for rendering.

CustomShapeGeometry
Sequence - This property describes the geometry of the CustomShape. The CustomShapeEngine that is used should be able to get on with the content of this property. If the CustomShapeEngine property is "com.sun.star.drawing.EnhancedCustomShapeEngine", then this property is containing properties as they are specified in the service com.sun.star.drawing.EnhancedCustomShapeGeometry

CustomShapeReplacementURL
String - This property describes the URL to a replacement graphic that could be displayed if the CustomShape engine is not available.

DatabaseImportDescriptor

Represents a description of how data from an external database is imported.

API Reference, service(s)
com.sun.star.sheet.DataBaseImportDescriptor

Properties

SourceType
DataImportMode - enables importing and specifies from what type of source data is imported. This does niot seem to have an affect, at present. All values of the enumeration are interpreted as SQL i.e. SourceObject has to be a SQL statement.

DatabaseName
String - specifies the name of the database from which data is imported.

SourceObject
String - specifies the table, query, or statement from which data is imported. Currently only a statement seems to apply.

IsNative
Boolean - specifies whether the SQL statement is given directly to the database or is parsed before.

ConnectionResource
String - indicates a connection URL, which locates a database driver.

DatabaseRanges(*)

Represents a collection of database ranges in a spreadsheet document.

API Reference, service(s)

com.sun.star.sheet.DataBaseRanges

Parent object(s)
Document

Methods

addNewByName(<s_NewRangeName>, <s_FormulaExpression>, <struct_CellAddress>, <const_NamedRangeFlag>)
Adds a new named range to the collection.

createEnumeration()
Returns an enumerated list of all columns in the collection.

getByName(<s_Rangename>)
Object (DatabaseRange) - returns the range object with the name <rangename>

getByIndex(l_Index)
Object (DatabaseRange) - returns the database range at the specified zero based index.

hasByName(<sRangename>)
True if the collection has a range of that name.

removeByName(<s_RangeName>)
Removes a named range from the collection.

Properties

Count
Long – returns the number of ranges in the collection.

ElementNames
String Array – returns a list of all the database ranges

DatabaseRange

Represents a database range in a spreadsheet document.

A database range is a name for a cell range that also stores filtering, sorting, subtotal and data import settings and options.

API Reference, service(s)
com.sun.star.sheet.DatabaseRange

Parent object(s)
DatabaseRanges

Methods

refresh
executes the stored import, filter, sorting, and subtotals descriptors again.

Properties

AutoFilter
Boolean - specifies whether the AutoFilter is enabled or not.

DataArea
Structure (CellRangeAddress) - defines the data area of the database range in the spreadsheet document.

FilterCriteriaSource
Structure (CellRangeAddress) - specifies the range where the filter can find the filter criterias.

FilterDescriptor
Object (SheetFilterDescriptor) - returns the filter descriptor stored with the database range.

FromSelection
Boolean - specifies whether the imported data is only a selection of the database.

ImportDescriptor

Sequence (DatabaseImportDescriptor) - returns the database import descriptor stored with this database range.

IsUserDefined
Boolean -

KeepFormats
Boolean - if this property is set, cell formats are extended when the size of the range is changed by an update operation.

LinkDisplayBitmap
? - ?

LinkDisplayName
String readonly - contains a human readable name for this object that could be displayed in a user interface.

MoveCells
Boolean - if this property is set, columns or rows are inserted or deleted when the size of the range is changed by an update operation.

Name
String - the programmatic name of the object.

ReferredCells
Object (CellRange) - returns the cell range object that is represented.

RefreshPeriod
Long - specifies the time between two refresh actions in seconds.

SortDescriptor
Seq (SheetSortDescriptor2) - returns the sort descriptor stored with the database range.

StripData
Boolean - if this property is set, the cell contents within the database range are left out when the document is saved.

SubTotalDescriptor
Object (SubTotalDescriptor) - returns the subtotal descriptor stored with the database range.

UseFilterCriteriaSource
Boolean - specifies whether the filter criteria should be taken from a CellRange.

DataPilotDescriptor

Provides access to the layout settings of a data pilot table.

API Reference, service(s)
com.sun.star.sheet.DataPilotDescriptor

Properties

ColumnFields
Object (DataPilotFields) - returns the collection of the data pilot fields used as column fields.

DataFields
Object (DataPilotFields) - returns the collection of the data pilot fields used as data fields.

DataPilotFields
Object (DataPilotFields) - returns the collection of all the data pilot fields.

FilterDescriptor
Object (SheetFilterDescriptor) - returns the filter descriptor specifying which data from the source cell range are used for the data pilot table.

HiddenFields
Object (DataPilotFields) - returns the collection of the data pilot fields not used as column, row, page, or data fields.

Name
String – The name of the pilot table.

PageFields

Object (DataPilotFields) - returns the collection of the data pilot fields used as page fields.

RowFields
Object (DataPilotFields) - returns the collection of the data pilot fields used as row fields.

SourceRange
Structure (CellRangeAddress) - range containing the data for the data pilot table.

Tag
String - The additional string stored in the data pilot table.

DataPilotField

API Reference, service(s)
com.sun.star.sheet.DataPilotField

Parent object(s)
DataPilotTables

Properties

AutoShowInfo
Structure (DataPilotFieldAutoShowInfo) - enables the automatic inclusion of only a number of items with the highest or lowest result values.

Function
Enumeration (GeneralFunction) - specifies the function used to calculate results for this field.

GroupInfo
Structure (DataPilotFieldGroupInfo) - contains the grouping information of the Field

HasAutoShowInfo
Boolean - specifies whether this field has auto show information.

HasLayoutInfo
Boolean - specifies whether this field has layout information.

HasReference
Boolean - specifies whether this field has a reference.

HasSortInfo
Boolean - specifies whether this field has sorting information.

IsGroupField
Boolean - specifies whether this field is a group field.

LayoutInfo
Structure (DataPilotFieldLayoutInfo) - controls how the field's items are laid out in the result table.

Name
String – the name of the data pilot field

Orientation
Enumeration (DataPilotFieldOrientation) - specifies the orientation of the field.

Reference
Structure (DataPilotFieldReference) - controls how the results are shown in relation to a selected reference result.

SelectedPage
String - specifies the selected page which is used to filter the data pilot.

ShowEmpty
Boolean - specifies whether to show this field also if it is empty or not.

SortInfo
Structure (DataPilotFieldSortInfo) - controls how the field's items are sorted.

UseSelectedPage
Boolean - specifies whether to use the selected page to filter the data pilot or show all.

UsedHierarchy
String - specifies which hierarchy of the dimension is used.

DataPilotFieldAutoShowInfo

Contains the auto show information of a DataPilotField.

API Reference, structure
com.sun.star.sheet.DataPilotFieldAutoShowInfo

Elements

IsEnabled
Boolean - specifies whether the AutoShow feature is enabled or not.

ShowItemsMode
Constant (DataPilotFieldShowItemsMode) - specifies the mode which items have to be shown.

ItemCount
Long - specifies the number of the items to show.

DataField
String - specifies the field where the values to show and select are taken from.

DataPilotFieldGroupInfo

Contains the grouping information of a DataPilotField .

API Reference, structure
com.sun.star.sheet.DataPilotFieldGroupInfo

Elements

HasAutoStart
Boolean - specifies whether the start value for the grouping is taken automatically

HasAutoEnd
Boolean - specifies whether the end value for the grouping is taken automatically

HasDateValues
Boolean - specifies whether date values are grouped

Start
Double - specifies the start value for the grouping if HasAutoStart is false

End
Double - specifies the end value for the grouping if HasAutoEnd is false

Step
Double - specifies the steps between the groups

GroupBy
Constant (DataPilotFieldGroupBy) - specifies the grouping of the date values.

SourceField
Object (DataPilotField) - specifies the source field.

Groups
Object (DataPilotFieldGroups) - specifies the groups if there are some

DataPilotFieldGroups

Represents a collection of groups in a data pilot field.

API Reference, structure
com.sun.star.sheet.DataPilotFieldGroups

Parent object(s)
DataPilotField

Methods

createEnumeration()

Returns an enumerated list of all data pilot field groups in the collection.

getByName("<s_datapilotfieldgroupname>")
Object - returns the data pilot field group object.

hasByName("<s_stylename>")
Boolean – true if the collection has the named data pilot field group.

hasElements
Boolean – true if the collection has data pilot field groups.

Properties

Count
Long – returns the number of datapilotfieldgroups in the collection.

ElementNames
String array – returns a list of all the datapilotfieldgroup name objects.

DataPilotFieldLayoutInfo

Contains the layout information of a DataPilotField .

API Reference, structure
com.sun.star.sheet.DataPilotFieldLayoutInfo

Elements

LayoutMode
Constant (DataFieldLayoutMode) - specifies the layout mode.

AddEmptyLines
Boolean - If true , an empty row is inserted in the DataPilotTable result table after the data (including the subtotals) for each item of the field.

DataPilotFieldReference

Controls how a data pilot field's results are shown in relation to a selected reference result.

API Reference, structure
com.sun.star.sheet.DataPilotFieldReference

Elements

ReferenceType
Constant (DataPilotFieldReferenceType) - contains the type of the reference.

ReferenceField
String - contains the reference field

ReferenceItemType
Constant (DataPilotFieldReferenceItemType)selects between a named reference item and using the previous or next item for each item from the reference field.

ReferenceItemName
String - contains the name of the reference item, when the DataPilotFieldReference::ReferenceItemType is NAMED otherwise is empty

DataPilotFields

API Reference, service(s)
com.sun.star.sheet.DataPilotFields

Parent object(s)
DataPilotTables

Methods

createEnumeration()
Returns an enumerated list of all objects in the collection.

getByIndex(<l_index>)
Object – returns the datafield at the specified zero based index.

getByName(<s_InternalName>)
Object – returns the data field with the matching internal name.

HasByName(<s_InternalName>)
Boolean – true if the collection has a chart of that internal name.

HasElements
Boolean – true if there are any objects in the collection.

Properties

Count
Long – returns the number of objects in the collection.

ElementNames
String Array – List of data field types in the collection

DataPilotFieldSortInfo

Describes how to sort a single DataPilotField

API Reference, structure
com.sun.star.sheet.DataPilotFieldSortInfo

Elements

Field
String - contains the data field to sort by if the Mode is DATA

IsAscending
Boolean - true if data are sorted in ascending order, false if in descending order.

Mode
Constant (DataPilotFieldSortMode) - contains the sort mode

DataPilotTable

A data pilot table object.

API Reference, service
com.sun.star.sheet.DataPilotTable

Parent object(s)
DataPilotTables

Methods

getOutputRange
Structure (CellRangeAddress) - returns the address of the cell range that contains the data pilot table.

refresh
Recreates the data pilot table with current data from the source cell range.

Properties

Property Group : DataPilotDescriptor

DataPilotTables(*)

Represents a collection of data pilot tables.

API Reference, service(s)
com.sun.star.sheet.DataPilotTables

Parent object(s)
Sheet

Methods

createDataPilotDescriptor
Object - creates a DataPilotDescriptor.

createEnumeration()
Returns an enumerated list of all datapilottables in the collection.

getByIndex(<l_index>)
Object – returns the DataPilotTable at the specified zero based index.

getByName(<s_InternalName>)
Object – returns the column with the matching internal name.

HasByName(<s_InternalName>)
Boolean – true if the collection has a chart of that internal name.

HasElements
Boolean – true if there are any DataPilotTables in the collection.

insertNewByName
Creates a new datapilottable and adds it to the collection.

removeByName
Deletes a DataPilotTable from the collection.

Properties

Count
Long – returns the number of DataPilotTables in the collection.

ElementNames
String Array – List of objects in the collection

Date

A structure that defines a date value.

API Reference, service(s)
com.sun.star.util.Date

Elements

Day
Short - contains the day of month (1-31 or 0 for a void date).

Month
Short - contains the month of year (1-12 or 0 for a void date).

Year
Short - contains the year

DateTime

A structure that defines a date+time value.

API Reference, service(s)
com.sun.star.util.DateTime

Elements

HundredthSeconds
Short - contains the hundredth seconds (0-99).

Seconds
Short - contains the seconds (0-59).

Minutes
Short - contains the minutes (0-59).

Hours
Short - contains the hour (0-23).

Day
Short - contains the day of month (1-31 or 0 for a void date).

Month
Short - contains the month of year (1-12 or 0 for a void date).

Year
Short - contains the year

DateTimeRange

A structure that defines a date-time range.

API Reference, service(s)
com.sun.star.util.DateTimeRange

Elements

StartHundredthSeconds
Short - contains the start hundredth seconds (0-99) for the range.

StartSeconds
Short - contains the start seconds (0-59) for the range.

StartMinutes
Short - contains the start minutes (0-59) for the range.

StartHours
Short - contains the start hour (0-23) for the range.

StartDay
Short - contains the start day of month (1-31 or 0 for a void date) for the range.

StartMonth
Short - contains the start month of year (1-12 or 0 for a void date) for the range.

StartYear
Short - contains the start year for the range.

EndHundredthSeconds
Short - contains the end hundredth seconds (0-99) for the range.

EndSeconds
Short - contains the end seconds (0-59) for the range.

EndMinutes
Short - contains the end minutes (0-59) for the range.

EndHours
Short - contains the end hour (0-23) for the range.

EndDay
Short - contains the end day of month (1-31 or 0 for a void date) for the range.

EndMonth
Short - contains the end month of year (1-12 or 0 for a void date) for the range.

EndYear
Short - contains the end year for the range.

DDELink

Represents a DDE link. A DDE link controls the results of a DDE spreadsheet formula.

API Reference, service(s)
com.sun.star.sheet.DDELink

Parent object(s)
DDELinks

Methods

refresh
refreshes the data of the object from the connected data source.

Properties

Application
String - returns the application from which data are requested (the DDE server application).

Item
String - returns the DDE item from which data are requested.

Name
String – The DDE link name

Topic
String – returns the DDE topic from which data are requested.

DDELinks(*)

This is a collection of DDE links.

API Reference, service(s)
com.sun.star.sheet.DDELinks

Parent object(s)
Document

Methods

createEnumeration
An enumerated list of DDE Link objects.

getByIndex(<l_Index>)
Object (DDELink) - returns the DDE link at the index within the collection.

getByName(<s_name>)
Object - Returns the link with the name <s_name>

hasByName(<s_name>)
Boolean – true if the DDE link exists within the collection.

hasElements
Boolean – true if there are DDE Links in the collection

Properties

Count
Integer – the number of DDE Links in the collection

ElementNames
String Array - returns a list of all the DDE link names.

Diagram

The diagram is the object that contains the actual plot of a chart. Not all properties are applicable to all charts.

API Reference, service(s)
com.sun.star.chart.Diagram

Parent object(s)
ChartDocument

Methods

getDataRowProperties(<l_rowseries>)
Object (ChartDataRowProperties) - the properties of the specified data row (series).

getDataPointProperties(<l_Column,<l_row>)
Object (ChartDataPointProperties) - the properties of the specified data point.

Properties

Axis
Long - determines to which axis the data row is assigned.

ConstantErrorHigh
Double - specifies the upper limit of the error range of a data row.

ConstantErrorLow
Double - specifies the lower limit of the error range of a data row.

DataRegressionProperties
Object (LineProperties) - holds the properties of the regression line, if such one is enabled.

DataErrorProperties
Object (LineProperties) - holds the properties of the error markers, if those are enabled.

DataMeanValueProperties
Object (LineProperties) - holds the properties of the average line, if such one is enabled.

DiagramType
String – returns a string representing the diagram type. This string contains the fully qualified name of the corresponding service. The possible diagrams are:
AreaDiagram , BarDiagram, Dim3DDiagram , DonutDiagram , LineDiagram , NetDiagram , PieDiagram , StackableDiagram , StockDiagram , XYDiagram
Some diagram properties are type specific.

Dim3D
Boolean - If set to true , the chart becomes a three-dimensional chart. Otherwise it is two-dimensional.

Deep
Boolean - If true , determines that in a three-dimensional bar chart the bars of each series are arranged behind each other in the z-direction. If false the arrangement of bars is like in two-dimensional bar charts.

DownBar
Object (ChartArea) - the properties of the down bars of a stock chart which has StockDiagram::UpDown set to true .
The DownBar is the box that is drawn between the open and close value of a stock, when the closing value is below the opening value, i.e., the stock price fell.

ErrorCategory
Enumeration (ChartErrorCategory) - determines the type of error to indicate.

ErrorIndicator
Enumeration (ChartErrorIndicatorType) - determines how the error is indicated.

ErrorMargin
Double - specifies the percentage for the margin of errors.

Floor
Object (ChartArea) - the properties of the floor. This is only valid for three-dimensional diagrams.

HasSecondaryYAxis
Boolean - determines if the secondary y-axis is shown or hidden.

HasSecondaryYAxisDescription
Boolean - determines if the description of the secondary y-axis is shown or hidden.

HasXAxis
Boolean - This property determines if the x-axis is shown or hidden.

HasXAxisDescription
Boolean - Determines if the description of the x-axis is shown or hidden.

HasXAxisGrid
Boolean - Determines if the major grid of the x-axis is shown or hidden.

HasXAxisHelpGrid
Boolean - Determines if the minor grid of the x-axis is shown or hidden.

HasXAxisTitle
Boolean - Determines if the title of the x-axis is shown or hidden.

HasYAxis
Boolean - Determines if the y-axis is shown or hidden.

HasYAxisDescription
Boolean - Determines if the description of the y-axis is shown or hidden.

HasYAxisGrid
Boolean - Determines if the major grid of the y-axis is shown or hidden.

HasYAxisHelpGrid
Boolean - Determines if the minor grid of the y-axis is shown or hidden.

HasYAxisTitle
Boolean - Determines if the title of the y-axis

HasZAxis
Boolean - Determines if the z-axis is shown or hidden.

HasZAxisDescription
Boolean - Determines if the description of the z-axis is shown or hidden.

HasZAxisGrid
Boolean - Determines if the major grid of the z-axis is shown or hidden.

HasZAxisHelpGrid
Boolean - Determines if the minor grid of the z-axis is shown or hidden.

HasZAxisTitle
Boolean - Determines if the title of the z-axis

Lines
Boolean - determines if the chart type has lines connecting the data points or contains just symbols.

MeanValue
Boolean - determines if the mean value for a data row is displayed as a line.

MinMaxLine
Object (LineProperties) - the properties of the lines that are drawn between the lowest and highest stock value during a day.

Percent
Boolean - If true , the series of the diagram are stacked and each category sums up to 100%.

Position
Structure (Point) - the current position of this object.

PercentageError
Double - specifies the percentage that is used to display error bars.

RegressionCurves
Enumeration (ChartRegressionCurveType) - determines a type of regression for the data

SecondaryYAxis
Object (ChartAxis) - the properties of the secondary y -axis of the diagram. The returned property set contains scaling properties as well as formatting properties.

Size
Structure (Size) - the size of this object.

SplineOrder
Long - specifies the power of the polynomials used for spline calculation This property is only valid for B-splines

SplineResolution
Long - determines the number of sampling points of a spline

SplineType
Long - determines if the chart is a spline-chart type and specifies the type of splines.

Stacked
Boolean - If true , the series of the diagram are stacked.

StackedBarsConnected
Boolean - determines if, in a stacked chart, there are connecting lines between corresponding bars. Currently, bar charts with horizontal bars do not support this property.

SymbolType
Long - determines which type of symbols are displayed. Only two values are supported NONE (0) and Auto (1)

SymbolSize
Structure (Size) - specifies the size of symbols in 1/100th of a millimeter.

UpBar
Object (ChartArea) - the properties of the up bars of a stock chart which has StockDiagram::UpDown set to true .
The UpBar is the box that is drawn between the open and close value of a stock, when the closing value is above the opening value, i.e., the stock price rose.

UpDown
Boolean - indicates if a stock chart contains data representing the value of stocks on the opening and closing date.

Vertical
Boolean - Determines if the bars of a chart are drawn vertically or horizontally. Default is vertical.

Volume
Boolean - indicates if a stock chart contains data representing the volume of stocks.

Wall
Object (ChartArea) - the properties of the diagram wall(s). This specifies the properties of the two side walls of the chart scene. Note that this property is also valid for two-dimensional diagrams. There the properties returned here affect the background rectangle of the diagram.

XAxis
Object (ChartAxis) - the properties of the x -axis of the diagram.
The returned property set contains scaling properties as well as formatting properties.

XAxisTitle
Object (ChartTitle) - the x -axis title shape.

XHelpGrid
Object (LineProperties) - the properties of the help grid (minor grid) of the x -axis of the diagram.

XMainGrid
Object (LineProperties) - the properties of the main grid (major grid) of the x -axis of the diagram.

YAxis
Object (ChartAxis) - the properties of the Y -axis of the diagram. The returned property set contains scaling properties as well as formatting properties.

YAxisTitle
Object (ChartTitle) - the Y -axis title shape.

YHelpGrid
Object (LineProperties) - the properties of the help grid (minor grid) of the Y -axis of the diagram.

YMainGrid
Object (LineProperties) - the properties of the main grid (major grid) of the Y -axis of the

diagram.

ZAxis
Object (ChartAxis) - the properties of the z -axis of the diagram. The returned property set contains scaling properties as well as formatting properties.

ZAxisTitle
Object (ChartTitle) - the z -axis title shape.

ZHelpGrid
Object (LineProperties) - the properties of the help grid (minor grid) of the z -axis of the diagram.

ZMainGrid
Object (LineProperties) - the properties of the main grid (major grid) of the z -axis of the diagram.

Document

The calc document object. This object is often referred to as a component.

API Reference, service(s)
com.sun.star.sheet.SpreadSheetDocument

Parent object(s)
StarDesktop

Methods

attachResource (s_FileURL>,<seq_MediaDescriptor>)
Boolean - informs a model about its resource description.

calculate
Recalculates all cells with formulas where their precedents have changed.

calculateAll
Recalculates all cells

enableAutomaticCalculation(<b_EnableState>)
enables or disables automatic calculation.

hasLocation
Boolean - The object may know the location because it was loaded from there, or because it is stored there.

isAutomaticCalculationEnabled
Boolean - returns whether automatic calculation is enabled.

isModified
Boolean – returns true if the document is modified.

isProtected
Boolean – returns the state of the protection.

print(<seq_PrintOptions>)
Prints the document.

protect(<s_password>)
Activates the protection.

seekGoal(<strct_CellAddress-Formula>, <struct_CellAddress-Variable>, <s_GoalValue>)
Structure (GoalResult) - returns the result of the goal seek.

SetModified
Sets the status of the modified -flag.

store
Stores the data to the URL from which it was loaded.

storeAsURL(<s_FileURL>,<seq_MediaDescriptor>)
Stores the object's persistent data to a URL and makes this URL the new location of the object.

storeToURL(<s_FileURL>,<seq_MediaDescriptor>)
Stores the object's persistent data to a URL and continues to be a representation of the old

URL.

unprotect(<s_password>)
Unprotects the document

Properties

AreaLinks
Object (AreaLinks) – a collection of area links.

ApplyFormDesignMode
Boolean - controls the initial (on-load) behavior of the form controls in the document

Args
Sequence (MediaDescriptor) - provides read access for the current representation of the MediaDescriptor of this model which describes the model and his state

AutomaticControlFocus
Boolean - controls the focus behaviour of the form controls in the document

CalcAsShown
Boolean - specifies whether calculations are performed with the rounded values displayed in cells (set to true) instead of the internal values (set to false).

CharLocale
Structure (Locale) - contains the standard document language for Western text.

CharLocaleAsian
Structure (Locale) - contains the standard document language for Asian text.

CharLocaleComplex
Structure (Locale) - contains the standard document language for Complex text.

ColumnLabelRanges
Object (ColumnLabelRanges)– returns a collection of column label ranges

CurrentController
Object (Controller) - the current controlling object i.e. window

CurrentSelection
Object - returns the object that is currently selected in the document.

DatabaseRanges
Object (DatabaseRanges) – returns a collection of database ranges.

DDELinks
Object (DDELinks) – a collection of DDE links.

DefaultTabStop
Short - specifies the width of default tabulators.

DocumentInfo
Object (DocumentInfo) - provides the document information object

Drawpages
Object (DrawPages) – returns a collection of draw pages which for a spreadsheet document is a list of sheets, however, they are not sheet objects the hierarchy is through the draw services.

ForbiddenCharacters
Readonly Object (ForbiddenCharacters) - contains the interface to the forbidden character settings

HasDrawPages
Readonly boolean - If this property is set the document has DrawPages. Use this property to find out, whether the document has DrawPages or not, because the getDrawPage method on the XDrawPageSupplier and the getDrawPages method on the XDrawPagesSupplier always creates the DrawPages if there are none; and this is very slow and needs more memory.

IgnoreCase
Boolean - specifies whether upper and lower cases are treated as equal when comparing cells.

IsIterationEnabled
Boolean - enables iterated calculation of circular references.

IterationCount

Long - specifies how many iterations are carried out.

IterationEpsilon
Double - specifies the point at which a change in results will stop the iteration.

Location
String - After using storeAsURL() it returns the URL the object was stored to.

LookUpLabels
Boolean - specifies whether column or row labels are looked up from anywhere on the sheet.

MatchWholeCell
Boolean - specifies whether filter criteria must match entire cell contents.

NamedRanges
Object (NamedRanges) – returns a collection of named ranges.

NullDate
Structure (Date) - specifies the date that is represented by the value zero.

NumberFormats
Object (NumberFormats) - returns the collection of number formats belonging to this document.

NumberFormatSettings
Object (Number Format Settings) – return the Number Format Settings of this object.

Printer
Sequence (PrinterDescriptor) - a descriptor of the current printer. The attributes of the current printer are used for formatting.

RegularExpressions
Boolean - specifies whether regular expressions in formulas are enabled, e.g., for functions which look up spreadsheet contents.

RowLabelRanges
Object – a collection of row label ranges.

RuntimeUID
(readonly) string - contains a unique id for the document

SheetLinks
Object (SheetLinks) – a collection of sheet links.

Sheets
Object (Sheets) – returns a collection of sheets.

SpellOnline
Boolean - enables online spell checking.

StandardDecimals
Short - specifies the number of decimals in the default number format.

StyleFamilies
Object (StyleFamilies) – returns a collection of style families

URL
String - provides information about the location of this model

DocumentInfo

An object that provides access to the user fields for the information regarding the document.

API Reference, service(s)
com.sun.star.document.DocumentInfo

Parent object(s)
Document

Methods

getUserFieldName(<w_Index>)
String - returns the name of an user field

getUserFieldValue(<w_Index)

String - returns the value of an user field

setUserFieldName(<w_Index>,<s_FieldName>)
changes the name of one of the user fields

setUserFieldValue(<w_Index>,<s_FieldValue>)
changes the value of one of the user fields

Properties

Author
String - contains the intial author of the document

AutoloadSecs
Long - contains the number of seconds after which a specified URL is to be loaded after the document is loaded into a desktop frame

AutoloadURL
String - contains the URL to load automatically after a specified time after the document is loaded into a desktop frame

CreationDate
Structure (DateTime) - contains the date and time of the first time the document was stored

DefaultTarget
String - contains the name of the default frame into which links should be loaded if no target is specified

Description
String - contains a multi-line comment of the document

DocumentStatistic
Structure (NamedValue) array - contains some statistics about the document

EditingCycles
Short - describes how often the document was edited and saved

EditingDuration
Long - contains the net time of editing the document (in seconds)

Generator
String - identifies application was used to create or last modify the document

Keywords
String - contains a comma separated list of keywords for the document

MIMEType
String readonly - contains the MIME-type of the document's resource

Language
Structure (Locale) - default language of the document

ModifiedBy
String - contains the name of the editor who was the last person to store this document

ModifyDate
Structure (DateTime) - contains the date and time of the last time the document was stored

PrintDate
Structure (DateTime) - contains the date and time of when the document was last printed

PrintedBy
String - contains the name of the editor who was the last person to print the document

Subject
String - subject of document

Template
String - contains logical name of the template from which the document was created

TemplateFileName
String - contains the file path name of the template from which the document was created

TemplateDate
Structure (DateTime) - contains the date and time of when the document was created or updated from the template

Title
String - contains the title of the document

UserFieldCount
Short - provides information about count of available fields

Documents

A collection of documents (i.e. components).

API Reference, service(s)
com.sun.star.frame.Components

Parent object(s)
StarDesktop

Methods

CreateEnumeration
An enumerated list of documents.

DrawPage(*)

This is a collection of all shapes and groupshapes on a page.

API Reference, service(s)
com.sun.star.sheet.SpreadsheetDrawPage

Parent object(s)
Sheet, ChartDocument

Methods

add(<o_Shape>)
Inserts a Shape into this collection.

getByIndex(<l_Index>)
Object – returns the style object at the specified zero based index.

group(<o_Shapes>)
Object (ShapeGroup) - groups the Shape s inside a collection.

hasElements
Boolean – true if the collection has styles.

remove(<o_Shape>)
Removes a Shape from this collection.

ungroup(<o_ShapeGroup>)
ungroups a given GroupShape .

Properties

Count
Long – returns the number of styles in the collection.

DrawPages(*)

A collection of draw pages. There, so far seems to be, a one to one relation between the drawpages and sheets. For example drawpage(1) has the same number of shapes as sheet(1).

The list of methods is not the same as indicated by the service hierarchy. Some don't work.

API Reference, service(s)
com.sun.star.drawing.DrawPages

Parent object(s)
Document

Methods

getByIndex(<l_Index>)
Object – returns a collection of draw objects on the page(sheet) <l_index>.

insertNewByIndex(<l_Index>)
Object - creates and inserts a new DrawPage(sheet) at the specified zero based index.

removeByIndex(<l_Index>)
Removes the drawpage(sheet) from the specified zero based index

Properties

Count
Long – returns the number of pages in the collection.

EllipseShape

An object that describes

API Reference, service(s)
com.sun.star.drawing.EllllipseShape

Parent Objects
Drawpage, GroupShape, Shapes

Properties

Property Groups: Shape , LineProperties , Fillproperties , Text , ShadowProperties

CircleEndAngle
Long - If the kind specifies an open circle, this is the end angle

CircleKind
Enumeration (CircleKind) - This is the kind of circle.

CircleStartAngle
Long - If the kind specifies an open circle, this is the start angle.

Enumeration

This is also very very unclear (in an object-method-property scenario). However it is best to think of this as a searchable list of all the objects in a parent collection. The list is created from some but not all collections as follows:

```
oEnumeratedObject = <a collection object>.createEnumeration
```
When created the list is “pointing” to the first element".

Allowed collections
Documents(components)

Cells

Columns

DataPilotField

DataPilotTables

TableCharts

Sheets

Methods

HasMoreElements
Boolean - returns true if the enumeration object has more elements.

NextElement

Object – returns the current object and advances the internal pointer to the next element in the enumerated collection.

FillProperties

This is a set of properties to describe the style for rendering an area.

ApI reference, Service
com.sun.star.drawing.FillProperties

Properties

FillBackground
Boolean - if this is true , the transparent background of a hatch filled area is drawn in the current background color

FillBitmap
Object (BitMap) - If the FillStyle is set to FillStyle::BITMAP , this is the bitmap used.

FillBitmapLogicalSize
Boolean - specifies if the size is given in percentage or as an absolute value.

FillBitmapMode
Enumeration (BitmapMode) - this enum selects how a area is filled with a single bitmap.

FillBitmapName
String - If the FillStyle is set to FillStyle::BITMAP , this is the name of the used fill bitmap style.

FillBitmapOffsetX
Short - This is the horizontal offset where the tile starts.

FillBitmapOffsetY
Short - This is the vertical offset where the tile starts. It is given in percent in relation to the width of the bitmap.

FillBitmapPositionOffsetX
Short - Every second line of tiles is moved the given percent of the witdh of the bitmap.

FillBitmapPositionOffsetY
Short - Every second row of tiles is moved the given percent of the width of the bitmap.

FillBitmapRectanglePoint
Enumeration (RectanglePoint) - The RectanglePoint specifies the position inside of the bitmap to use as the top left position for rendering.

FillBitmapSizeX
Long - This is the width of the tile for filling.

FillBitmapSizeY
Long - This is the height of the tile for filling.

FillBitmapURL
String - If the FillStyle is set to FillStyle::BITMAP , this is a URL to the bitmap used.

FillColor
Typedef (Color) - If the FillStyle is set to FillStyle::SOLID , this is the color used.

FillGradientName
String - If the FillStyle is set to FillStyle::GRADIENT , this is the name of the used fill gradient style.

FillGradient
Structure (Gradient) - If the FillStyle is set to FillStyle::GRADIENT , this describes the gradient used.

FillHatchName
String - If the FillStyle is set to FillStyle::GRADIENT , this is the name of the used fill hatch style.

FillHatch
Structure (Hatch) - If the FillStyle is set to FillStyle::HATCH , this describes the hatch used.
FillStyle
Enumeration (FillStyle) - This enumeration selects the style with which the area will be filled.

FillTransparence
Short - This is the transparence of the filled area.

FillTransparenceGradient
Structure (Gradient) - This describes the transparency of the fill area as a gradient.

FillTransparenceGradientName
String - If a gradient is used for transparency, this is the name of the used transparence gradient style or it is empty. If you set the name of a transparence gradient style contained in the document, this style used.

FontDescriptor

A structure that specifies the model of a check box control. The model supports the properties required for HTML, thus you can build up HTMLForm s with it.

The elements of this structure can be referenced directly as properties of the calling object by preceding the element name with “Font”, for example:

```
Print oControl.FontFamily
```

is the same as

```
sDescriptor = oControl.FontDescriptor
Print sDescriptor.Family
```

API Reference, structure
com.sun.star.awt.FontDescriptor

Elements

CharacterWidth
Float - specifies the character width.

CharSet
Short - specifies the character set which is supported by the font.

Family
Constant (FontFamily) - specifies the general style of the font.

Height
Short - specifies the height of the font in the measure of the destination.

Kerning
Boolean - For requesting, it specifies if there is a kerning table available. For selecting, it specifies if the kerning table is to be used.

Name
String - specifies the exact name of the font.

Orientation
Float - specifies the rotation of the font. The unit of measure is degrees; 0 is the baseline.

Pitch
Short - specifies the pitch of the font.

Slant
Enumeration (FontSlant) - specifies the slant of the font.

Strikeout
Constant (FontStrikeOut) - specifies the kind of strikeout.

StyleName
String - specifies the style name of the font.

Type
Constant (FontType) - specifies the technology of the font representation.

Underline
Constant (FontUnderline) - specifies the kind of underlining.

Weight
Float - specifies the thickness of the line.

Width

Short - specifies the width of the font in the measure of the destination.

WordLineMode
Boolean - specifies if only words get underlined.

ForbiddenCharacters

API Reference, structure

com.sun.star.i18n.ForbiddenCharacters

Elements

beginLine
String - Characters forbidden at the start of a line

endLine
String - Characters forbidden at the end of a line.

FormControl

An object that specifies a control. Listed below is all methods and properties for all spreadsheet controls. Not all properties and methods apply to all controls and there follows after the properties a grid showing what are applicable.

There are a number of controls that can have a database link. These additional details follow the grid.

API Reference, service(s)
com.sun.star.form.FormControlModel

Parent Objects
Shape (ControlShape)

Methods

reset
Resets a component to some default value.

Properties

Property Groups: FormControl (This is the generic control)

Align
Short - specifies the horizontal alignment of the text in the control.
specifies the horizontal alignment of the text in the control.
0: left
1: center
2: right

Autocomplete
Boolean - specifies whether automatic completion of text is enabled.

AutoHScroll
Boolean - If set to true an horizontal scrollbar will be added automaticly when needed.

AutoVScroll
Boolean - If set to true an vertical scrollbar will be added automaticly when

BackgroundColor
Long - specifies the background color (RGB) of the control.

BlockIncrement
Long - specifies the increment for a block move.

Border
Short - specifies the border style of the control.
specifies the border style of the control.
0: No border

1: 3D border
2: simple border

BorderColor
Long - specifies the color of the border, if present

ButtonType
Enumeration : (FormButtonType) - describes the action to be executed by the button when pressed.

ClassId
Constant (FormComponentType) - specifies the ID for classification of the component.

Count
Long – the number of columns in the grid control.

CurrencySymbol
String - specifies the currency symbol.

Date
Long - specifies the date displayed in the control. The date must be specified in the format YYYYMMDD.

DateFormat
Short - specifies the format of the displayed date.
specifies the format of the displayed date.
0: system short
1: system short YY
2: system short YYYY
3: system long
4: short DDMMYY
5: short MMDDYY
6: short YYMMDD
7: short DDMMYYYY
8: short MMDDYYYY
9: short YYYYMMDD
10: short YYMMDD DIN5008
11: short YYYYMMDD DIN5008

DateMax
Long - specifies the maximum date that can be entered.

DateMin
Long - specifies the minimum date that can be entered.

DateShowCentury
Boolean - specifies, if the date century is displayed.

DecimalAccuracy
Short - specifies the decimal accuracy.

DefaultButton
Boolean - specifies that the button is the default button on the document.

DefaultDate
Long - contains a default value for the control, date format is YYYYMMDD.

DefaultScrollValue
Long - contains a default value for the control.

DefaultSelection
Short Array - contains the indexes of entries of the listbox, which should selected by default.

DefaultSpinValue
Long - contains a default value for the control.

DefaultState
Short - contains a default value for the control.

DefaultText
String - contains a default value for the control.

DefaultTime
Long- contains a default value for the control.

DefaultValue
Double - contains a default value for the control.

Dropdown
Boolean - specifies if the control has a drop down button.

EchoChar
Short - specifies the echo character for a password edit field.

EditMask
String - specifies the edit mask.

ElementNames
String Array – List of internal column names for a grid control.

Enabled
Boolean - determines whether the control is enabled or disabled.

FocusOnClick
Boolean - specifies whether the button control should grab the focus when clicked.

FontDescriptor
Structure (FontDescriptor) - specifies the font attributes of the text in the control. The elements can be used directly as a property of this object by preceding the element name with "Font". For example the font family is o_control.FontFamily

FontEmphasisMark
Constant (FontEmphasis) - specifies the FontEmphasis value of the text in the control.

FontRelief
Constant (FontRelief) - specifies the FontRelief value of the text in the control.

Graphic
Object (Graphic) - specifies a graphic to be displayed besides the label of the control

HardLineBreaks
Boolean - specifies if hard line breaks will be returned in the XTextComponent::getText method.

HelpText
String - specifies the help text of the control.

HelpURL
String - specifies the help URL of the control.

HiddenValue
String - specifies the value of the component.

HideInactiveSelection
Boolean - specifies whether the selection in the control should be hidden when the control is not active (focused).

IconSize
Short - specifies the size of the icons in the control

ImageAlign
Constant (ImageAlign) - specifies the alignment of the image inside the button as ImageAlign value.

ImagePosition
Short - specifies the position of the image, if any, relative to the text, if any

ImageURL
String - specifies an URL to an image to display besides the label of the control

Label
String - specifies the label of the control.

LineCount
Short - specifies the maximum line count displayed in the drop down box.

LineEndFormat
Constant (LineEndFormat) - specifies which line end type should be used for multi line text

LineIncrement
Long - specifies the increment for a single line move.

ListSource
String Array - contains the listbox entries.

LiteralMask
String - specifies the literal mask.

LiveScroll
Boolean - specifies the scrolling behaviour of the control.

MaxTextLen
Short - specifies the maximum character count.

MultiLine
Boolean - specifies that the text may be displayed on more than one line.

MultiSelection
Boolean - specifies if more than one entry can be selected.

Name
String – the name of the control set by the user.

Orientation
Long - specifies the ScrollBarOrientation of the control.
0 - horizontal scroll bar.
1 - vertical scroll bar.

PaintTransparent
Boolean - specifies whether the control paints it background or not.

PrependCurrencySymbol
Boolean - specifies whether the currency symbol is to be prepended.

Printable
Boolean - specifies that the control will be printed with the document.

PushButtonType
Short Enumeration (PushButtonType) - specifies the default action of the button as PushButtonType value.

ReadOnly
Boolean - specifies that the content of the control cannot be modified by the user.

RefValue
String - contains a reference value which is used for submission in a HTML form

Repeat
Boolean - specifies whether the control should show repeating behaviour.

RepeatDelay
Long - specifies the mouse repeat delay, in milliseconds.

RichText
Boolean - specifies whether the control should display the text including all its formatting.

RowHeight
Long - specifies the height of a row of the grid.

ScaleImage
Boolean - specifies if the image is automatically scaled to the size of the control.

ScrollValue
Long - specifies the scroll value of the control.

ScrollValueMin
Long - specifies the minimum scroll value of the control.

ScrollValueMax
Long - specifies the maximum scroll value of the control.

SelectedItems
Short Array - specifies the sequence of selected items, identfied by the position.

ShowFilterSort
Boolean - determines whether the control should provide functionality for filtering and sorting the parent form

ShowNavigation

Boolean - determines whether the control should provide functionality for navigating the parent form

ShowPosition
Boolean - determines whether the control should provide functionality for positioning the parent form

ShowRecordActions
Boolean - determines whether the control should provide functionality for acting on the current record of the parent form

ShowThousandsSeparator
Boolean - specifies whether the thousands separator is to be displayed.

Spin
Boolean - specifies that the control has a spin button.

SpinIncrement
Long - specifies the increment by which the value is changed when using operating the spin button.

SpinValue
Long - specifies the current value of the control.

SpinValueMin
Long - specifies the minimum value of the control.

SpinValueMax
Long - specifies the maximum value of the control.

State
Short - specifies the state of the control.

StrictFormat
Boolean - specifies that the value is checked during the user input.

StringItemList
String Array - specifies the list of items.

SymbolColor
Typedef (Color) - specifies the RGB color to be used when painting symbols which are part of the control's appearance, such as the arrow buttons.

TabIndex
Short - determines the relative taborder of the control associated with the model.

Tabstop
Boolean - specifies that the control can be reached with the TAB key.

Tag
String - used for additional information.
No semantics is given for this property, it will usually be used by the creator of a document containing form controls.

TargetFrame
String - describes the frame, where to open the document specified by the TargetURL.

TargetURL
String - specifies the URL, which should be opened if the button was clicked.

Text
String - specifies the text displayed in the control.

TextColor
Typedef (Color) - specifies the text color (RGB) of the control.

TextLineColor
Typedef (Color) - specifies the text line color (RGB) of the control.

Time
Long - specifies the time displayed in the control.

TimeFormat
Short - specifies the format of the displayed time.
0: 24h short
1: 24h long

2: 12h short
3: 12h long
4: Duration short
5: Duration long

TimeMax
Long - specifies the maximum time that can be entered.

TimeMin
Long - specifies the minimum time that can be entered.

Toggle
Boolean - specifies whether the button should toggle on a single operation.

TriState
Boolean - specifies that the control may have the state "don't know".

UncheckedRefValue
String - specifies a value which is to be associated with the control when it's not selected.

Value
Double - specifies the value displayed in the control.

ValueMax
Double - specifies the maximum value that can be entered.

ValueMin
Double - specifies the minimum value that can be entered.

ValueStep
Double - specifies the value step when using the spin button.

VerticalAlign
Enumeration (VerticalAlignment) - specifies the vertical alignment of the text in the control.

VisibleSize
Long - specifies the visible size of the scroll bar.

VisualEffect
Short - specifies a visual effect to apply to the check box control

VScroll
Boolean - specifies if the content of the control can be scrolled in the vertical direction.

	CheckBox	ComboBox	CommandButton	CurrencyField	DateField	FileControl	FixedText	GridControl	GroupBox	HiddenControl	ImageButton	ListBox	NavigationToolBar	NumericField	PatterField	RadioButton	Scrollbar	SpinButton	TextField	TimeField
Methods																				
reset	Y	Y		Y	Y	Y		Y				Y		Y	Y	Y	Y	Y	Y	Y
Properties																				
Align	Y	Y					Y					Y				Y			Y	
Autocomplete		Y																		
AutoHScroll																			Y	
AutoVScroll																			Y	
BackgroundColor	Y	Y	Y	Y	Y	Y	Y				Y	Y		Y	Y	Y	Y	Y	Y	Y
BlockIncrement																	Y			
Border		Y		Y	Y	Y	Y	Y			Y	Y	Y	Y	Y		Y	Y	Y	Y
BorderColor		Y		Y	Y	Y	Y	Y			Y	Y		Y	Y		Y	Y	Y	Y
ButtonType			Y								Y									
ClassId	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Count								Y												
CurrencySymbol				Y																
Date					Y															
DateFormat					Y															
DateMax					Y															
DateMin					Y															
DateShowCentury					Y															
DecimalAccuracy				Y										Y						
DefaultButton			Y																	
DefaultDate					Y															
DefaultScrollValue																	Y			
DefaultSelection												Y								
DefaultSpinValue																		Y		
DefaultState	Y															Y				
DefaultText		Y				Y									Y				Y	
DefaultTime																				Y
DefaultValue				Y										Y						
DropDown		Y			Y							Y								
EditMask															Y					
EchoChar																			Y	
Elementnames								Y												
Enabled	Y	Y	Y	Y	Y			Y	Y		Y	Y	Y			Y	Y	Y	Y	
FocusOnClick			Y																	
FontDescriptor	Y	Y	Y	Y	Y	Y	Y	Y	Y			Y	Y	Y	Y	Y			Y	Y
FontEmphasisMark	Y	Y	Y	Y	Y	Y	Y		Y			Y	Y	Y	Y	Y			Y	Y
FontRelief	Y	Y	Y	Y	Y	Y	Y		Y			Y	Y	Y	Y	Y			Y	Y
Graphic	Y		Y								Y					Y				

	CheckBox	ComboBox	CommandBu	CurrencyFiel	DateField	FileControl	FixedText	GridControl	GroupBox	HiddenContr	ImageButton	ListBox	NavigationTc	NumericFielc	PatterField	RadioButton	Scrollbar	SpinButton	TextField	TimeField
HardLineBreaks																			Y	
HelpText	Y	Y	Y	Y	Y	Y	Y		Y		Y	Y		Y	Y	Y	Y	Y	Y	Y
HelpURL	Y	Y	Y	Y	Y	Y	Y		Y		Y	Y		Y	Y	Y	Y	Y	Y	Y
HiddenValue										Y										
HideInactiveSection		Y		Y	Y	Y								Y	Y				Y	Y
HScroll																			Y	
IconSize													Y							
ImageAlign			Y																	
ImagePosition	Y		Y													Y				
ImageURL	Y		Y								Y					Y				
Label	Y		Y				Y		Y											
LineCount		Y										Y								
LineEndFormat																			Y	
LineIncrement																				
ListSource												Y								
LiteralMask															Y					
LiveScroll																				
MaxTextLen		Y													Y				Y	
MultiLine	Y		Y				Y									Y			Y	
MultiSelection												Y								
Name	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Orientation																		Y		
PaintTransparent																			Y	
PrependCurrencySymbol				Y																
Printable	Y	Y	Y	Y	Y	Y	Y		Y		Y	Y		Y	Y	Y	Y	Y	Y	Y
PushButtonType			Y																	
ReadOnly		Y		Y	Y	Y						Y		Y	Y				Y	Y
RefValue	Y															Y				
Repeat			Y	Y	Y									Y				Y		Y
RepeatDelay			Y	Y	Y								Y	Y			Y	Y		Y
RichText																			Y	
RowHeight								Y												
ScaleImage											Y									
ScrollValue																	Y			
ScrollValueMin																	Y			
ScrollValueMax																	Y			
SelectedItems												Y								
ShowFilterSort													Y							
ShowNavigation													Y							
ShowPosition													Y							
ShowRecordActions													Y							
ShowThousandsSeparator				Y										Y						

Spin				Y	Y									Y						Y
SpinIncrement																		Y		
SpinValue																		Y		
SpinValueMin																		Y		
SpinValueMax																		Y		
State	Y		Y													Y				
StrictFormat				Y	Y									Y	Y					Y
StringItemList		Y										Y								
SymbolColor																	Y	Y		
TabIndex	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Tabstop	Y	Y	Y	Y	Y	Y		Y			Y	Y		Y	Y	Y	Y		Y	Y
Tag	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
TargetFrame			Y								Y									
TargetURL			Y								Y									
Text		Y			Y	Y									Y				Y	Y
TextColor	Y	Y	Y	Y	Y	Y	Y	Y	Y			Y	Y	Y	Y	Y			Y	Y
TextLineColor	Y	Y	Y	Y	Y	Y	Y		Y			Y	Y	Y	Y	Y			Y	Y
Time																				Y
TimeFormat																				Y
TimeMax																				Y
TimeMin																				Y
Toggle			Y																	
TriState	Y																			
UncheckedRefValue																Y				
Value				Y										Y						
ValueMax				Y										Y						
ValueMin				Y										Y						
ValueStep				Y										Y						
VerticalAlign	Y		Y				Y									Y				
VisibleSize																	Y			
VisualEffect	Y															Y				
Vscroll																			Y	
Property Groups:																				
CharacterProperties																			Y	
CharacterPropertiesAsian																			Y	
CharacterPropertiesComplex																			Y	
ParagraphProperties																			Y	
ParagraphPropertiesAsian																			Y	
ParagraphPropertiesComplex																			Y	

The following controls can have database links attatched to them

CheckBox
ComboBox

CurrencyField
DateField
Formattedfield
ImageControl
ListBox
NumericField
PatternField
RadioButton
TextField
TimeField

The additional methods and properties for database controls are:

Methods

commit
Boolean - commits the content of the component into the data source it is bound to. Return value is the success state.

Properties

BoundField
Object (?) - references to the cursor field to which the control is bound. Only if defined.

DataField
String - specifies the name of the bound database field. Only if defined.

LabelControl
Object (?) - references to a control model within the same document which should be used as a label. Only if defined.

GoalResult

A structure that is the result of a goal seek operation.

API Reference, service(s)
com.sun.star.sheet.GoalResult

Elements

Divergence
Double - the amount by which the result changed in the last iteration.

Result
Double - the resulting value.

Gradient

A structure that describes a color dispersion within an area.

API Reference, structure
com.sun.star.awt.Gradient

Elements

Angle
Short - angle of the gradient in 1/10 degree.

Border
Short - per cent of the total width where just the start color is used.

EndColor
Typedef (Color) - specifies the color at the end point of the gradient.

EndIntensity

Short - specifies the intensity at the end point of the gradient.

StartColor
Typedef (Color) - specifies the color at the start point of the gradient.

StartIntensity
Short - specifies the intensity at the start point of the gradient.

StepCount
Short - specifies the number of steps of change color.

Style
Enumeration (GradientStyle) - specifies the style of the gradient.

XOffset
Short - specifies the X-coordinate, where the gradient begins.

YOffset
Short - specifies the Y-coordinate, where the gradient begins.

Graphic

This object acts as a container for graphics.

API Reference, service(s)
com.sun.star.graphic.Graphic

Parent Objects
Shape (GraphicObjectShape)

Properties

Alpha
Boolean - Indicates that it is a pixel graphic with an alpha channel

Animated
Boolean - Indicates that it is a graphic that consists of several frames that can be played as an animation

BitsPerPixel
Byte - The number of bits per pixel used for the pixel graphic

GraphicType
Constant (GraphicType) - The type of the graphic

MimeType
String - The MimeType of the loaded graphic. The following types are supported:
image/bmp
image/gif
image/jpeg
image/x-photo-cd
image/x-pcx
image/png
image/tiff
image/x-xbitmap
image/x-xpixmap
image/x-portable-bitmap
image/x-portable-graymap
image/x-portable-pixmap
image/x-cmu-raster
image/x-targa
image/x-photoshop
image/x-eps
image/x-dxf
image/x-met
image/x-pict
image/x-sgf
image/x-svm

image/x-wmf
image/x-sgv
image/x-emf
image/x-vclgraphic

SizePixel
Structure (Size) - The Size of the graphic in pixel.

Size100thMM
Structure (Size) The Size of the graphic in 100th mm.

Transparent
Boolean - Indicates that it is a transparent graphic

GraphicObjectShape

An object that describes

API Reference, service(s)
com.sun.star.drawing.GraphicObjectShape

Parent Objects
Drawpage, GroupShape, Shapes

Properties

Property Groups: Shape , Text , ShadowProperties

AdjustBlue
Short - If this property is set, the blue channel of this graphic shape is adjusted by the given signed percent value.

AdjustContrast
Short - If this property is set, the contrast of this graphic shape is adjusted by the given signed percent value.

AdjustGreen
Short - If this property is set, the green channel of this graphic shape is adjusted by the given signed percent value.

AdjustLuminance
Short - If this property is set, the luminance of this graphic shape is adjusted by the given signed percent value.

AdjustRed
Short - If this property is set, the red channel of this graphic shape is adjusted by the given signed percent value.

Gamma
Double - If this property is set, the gama value of this graphic shape is adjusted by the given value.

Graphic
Object (Graphic) - This is the graphic that represents this graphic shape

GraphicColorMode
Enumeration (ColorMode) - This property selects the color mode that is used for rendering.

GraphicStreamURL
String - This is an url to the stream ('in document' or linked graphic) for this graphic shape.

GraphicURL
String - This is an url to the source bitmap for this graphic shape.

ImageMap
Object (ImageMap) - This property contains an HTML image map for this graphic. This object is not described in this document.

Transparency
Short - If this property is set, the transparency value of this graphic shape is adjusted by the given unsigned percent value. 100% is fully transparent, 0 % is fully opaque.

GroupShape

An object the describes a group of shapes.

API Reference, service(s)
com.sun.star.drawing.GroupShape

Parent Objects
Shape, GroupShape , DrawPage, Shapes

Methods

enterGroup
Enters the group which enables the editing function for the parts of a grouped Shape . Then the parts can be edited instead of the group as a whole.

leaveGroup
Leaves the group, which disables the editing function for the parts of a grouped Shape . Then only the group as a whole can be edited.

Properties

Property Groups: Shape

Hatch

This structure defines the appearence of a hatch.

A hatch is a texture made of straight lines.

API Reference, structure
com.sun.star.drawing.Hatch

Elements

Angle
Long - You can rotate the lines of the hatch with this angle.

Color
Typedef (Color) - This is the color of the hatch lines.

Distance
Long - This is the distance between the lines in the hatch.

Style
Enumeration (HatchStyle) - The HatchStyle defines the kind of lines used to draw this hatch.

HeaderFooterContent

This is an object that is the textual content of the headers and footers.

API Reference, service(s)
com.sun.star.sheet.HeaderFooterContent

Parent object(s)
PageStyle

Properties

LeftText
Object (HeaderFooterText) - returns the text which is printed in the left part of the header or footer.

CenterText
Object (HeaderFooterText) - returns the text which is printed in the center part of the header or footer.

RightText
Object (HeaderFooterText) - returns the text which is printed in the right part of the header or footer.

HeaderFooterText

This is an object that is the text.

API Reference, service(s)
com.sun.star.sheet.Text

Parent object(s)
HeaderFooterContent

Properties

String
The textual content

HomogenMatrix3

Specifies a homogenous matrix by three homogenous lines

ApI reference, Structure
com.sun.star.drawing.HomogenMatrix3

Elements

Line1
Structure (HomogenMatrixLine3) - Line 1

Line2
Structure (HomogenMatrixLine3) - Line 2

Line3
Structure (HomogenMatrixLine3) - Line 3

HomogenMatrixLine3

Specifies a single line for a HomogenMatrix3 .

ApI reference, Structure
com.sun.star.drawing.HomogenMatrixLine3

Elements

Column1
Double

Column2
Double

Column3
Double

LineDash

A LineDash defines a non-continuous line.

ApI reference, structure
com.sun.star.drawing.LineDash

Elements

Style

Enumeration (Dashstyle) - This sets the style of this LineDash .

Dots
Short - This is the number of dots in this LineDash .

DotLen
Long - This is the length of a dot.

Dashes
Short - This is the number of dashes.

DashLen
Long - This is the length of a single dash.

Distance
Long - This is the distance between the dots.

LineProperties

This is a set of properties to describe the style for rendering a Line. The properties for line ends and line starts are only supported by shapes with open line ends.

ApI reference, Service
com.sun.star.drawing.LineProperties

Properties

LineColor
Typedef (Color) - This property contains the line color.

LineDash
Structure (LineDash) - This property contains the dash of the line.

LineDashName
String - This property contains the name of the dash of the line.

LineEnd
Structure (PolyPolygonBezierCoords) - This property contains the line end in the form of a poly polygon bezier.

LineEndCenter
Boolean - If this property is true , the line will end in the center of the polygon.

LineEndName
String - This property contains the name of the line end poly polygon bezier.

LineEndWidth
Long - This property contains the width of the line end polygon.

LineJoint
Enumeration (LineJoint) - This property defines the rendering of joints between thick lines

LineStart
Structure (PolyPolygonBezierCoords) - This property contains the line start in the form of a poly polygon bezier.

LineStartCenter
Boolean - If this property is true , the line will start from the center of the polygon.

LineStartName
String - This property contains the name of the line start poly polygon bezier.

LineStyle
Enumeration (LineStyle) - This property contains the type of the line.

LineTransparence
Short - This property contains the extent of transparency.

LineWidth
Long - This property contains the width of the line in 1/100th mm.

LineStartWidth
Long - This property contains the width of the line start polygon.

LineShape

An object that describes a line shape.

API Reference, service(s)
com.sun.star.drawing.LineShape

Parent Objects
Drawpage, GroupShape, Shapes

Properties

Property Groups: Shape , LineProperties , Fillproperties , Text , ShadowProperties

Locale

This is a structure that represents a specific geographical, political or cultural region.

API Reference, service(s)
com.sun.star.lang.Locale

Elements

Language
String – The abbreviated ISO language code.

Country
String – The abbreviated ISO country code

Variant
contains a variant of the locale; codes are vendor and browser-specific.

Example

```
Dim LocalSettings As New com.sun.star.lang.Locale

LocalSettings.Language = "en"
LocalSettings.Country = "us"
```

MeasureShape

An object that describes

API Reference, service(s)
com.sun.star.drawing.MeasureShape

Parent Objects
Drawpage, GroupShape, Shapes

Properties

Property Groups: Shape , LineProperties , Fillproperties , Text , ShadowProperties , RotationDescriptor

EndPosition
Structure (Point) - this point is the end of the measured distance

MeasureBelowReferenceEdge
Boolean - If this property is true , the measure is drawn below the reference edge instead of above it.

MeasureDecimalPlaces
Short - This value is the number of decimal places that is used to format the measure value.

MeasureHelpLine1Length
Long - This is the length of the first help line.

MeasureHelpLine2Length

Long - This is the length of the second help line.

MeasureHelpLineDistance
Long - This is the distance from the measure line to the start of the help lines.

MeasureHelpLineOverhang
Long - This is the overhang of the two help lines.

MeasureKind
Enumeration (MeasureKind) - determines whether a measured shape is a standard measure or a radius measure.

MeasureLineDistance
Long - This is the distance from the reference edge to the measure line.

MeasureOverhang
Long - This is the overhang of the reference line over the help lines.

MeasureShowUnit
Boolean - If this is true , the unit of measure is shown in the measure text.

MeasureTextAutoAngle
Boolean - If this is true , the angle of the measure is set automatically.

MeasureTextAutoAngleView
Long - This is the automatic angle.

MeasureTextFixedAngle
Long - This is the fixed angle.

MeasureTextHorizontalPosition
Enumeration (MeasureTextHorzPos) - This is the horizontal position of the measure text.

MeasureTextVerticalPosition
Enumeration (MeasureTextVertPos) - This is the vertical position of the text.

MeasureTextIsFixedAngle
Boolean - If this value is true , the measure has a fixed angle.

MeasureTextRotate90
Boolean - If this value is true , the text is rotated 90 degrees.

MeasureTextUpsideDown
Boolean - If this value is true , the text is printed upside down.

StartPosition
Structure (Point) - this point is the start of the measured distance

MediaDescriptor

This is a property set.

API Reference, service
com.sun.star.document.MediaDescriptor

Properties

AsTemplate
Boolean - document is a template

Author
String - the author of the document

CharacterSet
String - identifier of used character set

Comment
String - description of document

ComponentData
Any - pack specific properties of caller

DocumentBaseURL
String - The base URL of the document to be used to resolve relative links.

DocumentTitle

String - document title

FilterName
String - internal filter name

FilterOptions
String - additional properties for filter

FilterData
Any - additional properties for filter

Hidden
Boolean - load document invisible

HierarchicalDocumentName
String - The hierarchical path to the embedded document from topmost container.

OutputStream
Object (XoutputStream) - a stream to receive the document data.

InputStream
Object (XinputStream) - content of document

InteractionHandler
Object (XinteractionHandler) - handle exceptional situations

JumpMark
String - jump to a marked position after loading

MediaType
String - specify mime type of content

OpenNewView
Boolean - opens a new view for an already loaded document

Overwrite
Boolean - overwrite any existing file

Password
String - password for loading storing documents

PostData
Sequence< byte > - contains the data for HTTP post method as a sequence of bytes.

Preview
Boolean - show preview

ReadOnly
Boolean - open document readonly

StartPresentation
Boolean - start presentation from a document

Referer
String - name of document referrer

RepairPackage
Boolean - let the document be opened in repair mode

StatusIndicator
Object (XstatusIndicator) - can be used for status informations

TemplateName
String - name of the template instead of the URL

TemplateRegionName
String - name of the template instead of the URL

Unpacked
Boolean - regulate using of compressing

URL
String - URL of the document

Version
Short - storage version

ViewData
Any - set special view state

ViewId
Short - id of the initial view

MacroExecutionMode
Short - should the macro be executed. the value should be one from MacroExecMode constant list.

UpdateDocMode
Short - can the document be updated depending from links. the value should be one from UpdateDocMode constant list.

NamedRange

A named range object.

I have not yet been able to get a relative formula e.g. =B3+1 to work. The absolute version =b3+1 works. The reference position methods seem to work except that the relative position does not affect a relative formula. So although I have verified that the method syntax is correct I have not proved the functionality.

API Reference, service(s)
com.sun.star.sheet.NamedRange

Parent object(s)
NamedRanges

Properties

Content
String - the content of the named range.

Name
String – the name of the range.

ReferencePosition
Struct (CellAddress) - the position in the document which is used as a base for relative references in the content.

Type
Long - the type of the named range.

NamedRanges(*)

A collection of named ranges.

API Reference, service(s)
com.sun.star.sheet.NamedRanges

Parent object(s)
Document

Methods

addNewByName(<sNewRangeName>,<sFormulaExpression>,<tCellAddress>,<const NamedRangeFlag>)
Adds a new named range to the collection. The celladdress specifies the base reference for relative addresses

addNewFromTitles(<tCellRangeAddress>,<enum Border>)
Creates named cell ranges from titles in a cell range.

CreateEnumeration
Object - returns an enumerated list of ranges.

getByIndex(<l_Index>)
Object – returns the named range at the zero based index in the collection.

getByName(<s_Rangename>)
Object - Returns the range object with the name <rangename>

hasByName(<sRangename>)
True if the collection has a range of that name.

removeByName(<sRangeName>)
Removes a named range from the collection.

outputList(<tCellAddress>)
Writes a list of all named ranges into the document.

Properties

Count
Long – returns the number of ranges in the collection.

ElementNames
String Array – returns a list of all the named ranges

NamedValue

Specifies a pair assembled from a name and a value.

API Reference, structure
com.sun.star.beans.NamedValue

Elements

Name
String - the name part.

Value
Any - the value part

NumberFormatProperties

Contains properties specifying the behavior of a NumberFormatter .

API Reference, service(s)
com.sun.star.util.NumberFormatproperties
There must be additional services not listed in the API documentation

Parent object(s)
Document

Properties

Comment
String - contains a comment

CurrencyAbbreviation
String -

CurrencyExtension
String -

CurrencySymbol
String -

Decimals
Integer -

FormatString
String - contains the format string of the number format.

LeadingZeros
Integer -

Locale
Structure (Locale) - contains the locale of the number format.

NegativeRed

Boolean -

StandardFormat
Boolean -

ThousandsSeperator
Boolean -

Type
Constant (NumberFormat) - contains the type of the number format.

UserDefined
Boolean -

NumberFormats (*)

These are pre-defined number format objects. The format is dependant on country specific abbreviations. This is not a collection.

API Reference, service(s)
com.sun.star.util.NumberFormats

Parent object(s)
Document.

Methods

addNew(<s_Format>,<struct_Locale>)
Integer – the new format id for the <format string>.

addNewConverted(<s_Format>, <struct_Locale-source>, <struct_Locale-destination>)
Long - adds a new number format to the list, using a format string in a different locale than the desired locale of the resulting number format.

generateFormat(<l_BaseKey>, <struct_Locale>, <b_Thousands>, <b_Red>, <w_Decimals>, <w_Leading>)
String - generates a format string from several parameters without creating an actual number format.

getByKey(<l_Index>)
Object – returns a NumberFormatProperties object

getFormatForLocale(<l_Key>, <struct_Locale>)
Long - returns the index of the converted number format if successful, the old index if the number format could not be converted.

getFormatIndex(<const_NumberFormatIndex>, <struct_Locale>)
Long - returns the key of a built-in format for a locale.

getStandardFormat(<const NumberFormat>, <struct Locale>)
Long - returns the key of the standard format within a specified type for a given locale.

getStandardIndex(<struct Locale>);
Long - returns the key of the standard format for a locale.

isTypeCompatible(<wOldType>, <wNewType>)
boolean verifies if one type of number format is compatible with another type.

queryKey(<sFormat>,<struct Locale>,<?bScan?>)
Integer – the internal format id for the <format string> or -1 if not found.

queryKeys(<const NumberFormat>,<struct Locale>,<bCreateNew>)
Long Seq – returns a list of all the format index keys.

removeByKey(<lKey>)
Removes a number format from the list.

Properties

Example

```
Dim oDoc As Object
```

```
Dim oSheet As Object
Dim oCell As Object
Dim oFormats As Object
Dim sFormatString As String
Dim lFormatId As Long
Dim tLocalSettings As New com.sun.star.lang.Locale

oDoc = StarDesktop.CurrentComponent
oSheet = oDoc.Sheets(0)
oCell = oSheet.getCellByPosition(1,1)

oCell.Value = 23400.3523565

tLocalSettings.Language = "en"
tLocalSettings.Country = "us"

oFormats = Doc.NumberFormats
sFormatString = "#,##0.000"

lFormatId = oFormats.queryKey(sFormatString, tLocalSettings, True)
If lFormatId = -1 Then
   lFormatId = Formats.addNew(sFormatString, tLocalSettings)
End If

MsgBox lFormatId
Cell.NumberFormat = NumberFormatId
```

NumberFormatSettings (*)

An object that specifies the settings for number formatting.

API Reference, service(s)
com.sun.star.util.NumberFormatSettings

Parent object(s)
Document

Properties

NullDate
Date - specifies the date which is represented by the value 0.

StandardDecimals
Short - specifies the maximum number of decimals used for the standard number format ("General").

NoZero
Boolean - is set to indicate that a zero value should be formatted as an empty string.

TwoDigitDateStart
Short - specifies the first year to be generated from a two-digit year input.

OLE2Shape

This object is an ole shape, for a spreadsheet this is a chart.

API Reference, service(s)
com.sun.star.drawing.OLE2Shape

Parent Objects
Drawpage, GroupShape, Shapes

Properties

Property Groups: Shape

CLSID
String - If you get this property you get the CLSID of the OLE2 document stream contained in this OLE2 shape. If you set it, an empty OLE2 document stream with this CLSID is created within this OLE2 shape.

PersistName

String - this is the internal storage name. It matches the Name property for a chart.

IsInternal
Boolean - This property returns true for all OLE2 that are internal Office components.

OpenBezierShape

An object that describes

API Reference, service(s)
com.sun.star.drawing.OpenBezierShape

Parent Objects
Drawpage, GroupShape, Shapes

Properties

Property Groups: Shape , LineProperties , Fillproperties , Text , ShadowProperties , PolyPolygonBezierDescriptor

PageShape

An object that describes

API Reference, service(s)
com.sun.star.drawing.PageShape

Parent Objects
Drawpage, GroupShape, Shapes

Properties

Property Groups: Shape

PageNumber
Long - This is the page number that is used for the preview. For page shapes on notes pages, this can't be changed. For page shapes on handout pages, this value only describes the relative order of the different page shapes on the page.

PageStyle

This is an object that defines a single style for a spreadsheet page or pages.

API Reference, service(s)
com.sun.star.sheet.TablePageStyle

Parent object(s)
PageStyles

Properties

BackColor
Long – color of background

BackGraphicURL
String – URL of the background graphics that you want to use

BackGraphicFilter
String – name of the filter for interpreting the background graphics

BackGraphicLocation
Enum – position of the background graphics (value according to enumeration)

BackTransparent
Boolean - makes the background transparent

BottomBorder
Struct (BorderLine) – specifications for bottom line of page border.

BottomBorderDistance
Long – distance between bottom page border and page content in hundredths of a millimeter

BottomMargin
Long – width of the bottom page margin in hundredths of a millimeter

CenterHorizontally
Boolean - table content is centered horizontally. Applies only to spreadsheets.

CenterVertically
Boolean - table content is centered vertically. Applies only to spreadsheets.

FirstPageNumber
Short – the page number for the first page of this sheet

FooterBackColor
Long – background color of footer

FooterBackGraphicFilter
String – name of the filter for interpreting the background graphics for the footer

FooterBackGraphicLocation
Enum – position of background graphics for the footer (value according to com.sun.star.style.GraphicLocation enumeration)

FooterBackGraphicURL
String – URL of the background graphics that you want to use

FooterBackTransparent
Boolean – shows the background of the footer as transparent

FooterBodyDistance
Long – distance between footer and main body of document in hundredths of a millimeter

FooterBottomBorder
Struct (BorderLine) – details of bottom line of border around footer.

FooterBottomBorderDistance
Long – distance between bottom border and content of footer in hundredths of a millimeter

FooterHeight
Long – height of footer in hundredths of a millimeter

FooterIsDynamicHeight
Boolean – height of footer is adapted automatically to the content

FooterIsOn
Boolean – footer is activated

FooterIsShared
Boolean – the footers on the even and odd pages have the same content (refer to FooterText, FooterTextLeft, and FooterTextRight).

FooterLeftBorder
Struct (BorderLine) – details of left-hand line of border around footer.

FooterLeftBorderDistance
Long – distance between left-hand border and content of footer in hundredths of a millimeter

FooterLeftMargin
Long – distance between footer and left-hand page margin in hundredths of a millimeter

FooterRightBorder
Struct (BorderLine) – details of right-hand line of border around footer.

FooterRightMargin
Long – distance between footer and right-hand page margin in hundredths of a millimeter

FooterRightBorderDistance
Long – distance between right-hand border and content of footer in hundredths of a millimeter

FooterShadowFormat
Struct (ShadowFormat) – details of shadow of footer.

FooterTopBorder
Struct (BorderLine) – details of top line of border around footer.

FooterTopBorderDistance

Long – distance between top border and content of footer in hundredths of a millimeter

HeaderBackColor
Long – background color of header

HeaderBackGraphicFilter
String – name of the filter for interpreting the background graphics for the header

HeaderBackGraphicLocation
Enum – position of the background graphics for the header (value according to com.sun.star.style.GraphicLocation enumeration)

HeaderBackGraphicURL
String – URL of the background graphics that you want to use

HeaderBodyDistance
Long – distance between header and main body of document in hundredths of a millimeter

HeaderBackTransparent
Boolean – shows the background of the header as transparent

HeaderBottomBorder
Struct (BorderLine) – details of the bottom line of the border around header.

HeaderBottomBorderDistance
Long – distance between bottom border and content of header in hundredths of a millimeter

HeaderIsDynamicHeight
Boolean – height of header is automatically adapted to content

HeaderHeight
Long – height of header in hundredths of a millimeter

HeaderIsOn

HeaderOn
Boolean – header is activated

HeaderIsShared

HeaderShared
Boolean – headers on even and odd pages have the same content. In this case the right content defines both.

HeaderLeftBorder
Struct (BorderLine) – details of the left-hand border of frame around header.

HeaderLeftBorderDistance
Long – distance between left-hand border and content of header in hundredths of a millimeter

HeaderLeftMargin
Long – distance between header and left-hand page margin in hundredths of a millimeter.

HeaderRightBorder
Struct (BorderLine) – details of the right-hand border of frame around header.

HeaderRightBorderDistance
Long – distance between right-hand border and content of header in hundredths of a millimeter

HeaderRightMargin
Long – distance between header and right-hand page margin in hundredths of a millimeter

HeaderShadowFormat
Struct (ShadowFormat) – details of shadow of header.

HeaderTopBorder
Struct (BorderLine) – details of the top line of the border around header.

HeaderTopBorderDistance
Long – distance between top border and content of header in hundredths of a millimeter

Height
Long – height of page in hundredths of a millimeter

IsLandscape
Boolean – landscape format

LeftBorder
Struct (BorderLine) – specifications for left-hand line of page border.

LeftBorderDistance
Long – distance between left-hand page border and page content in hundredths of a millimeter

LeftMargin
Long – width of the left hand page margin in hundredths of a millimeter

LeftPageFooterContent
Object (HeaderFooterContent) - contains the content of the footer for left pages. When both left and right have the same content then the right content defines both.

LeftPageHeaderContent
Object (HeaderFooterContent) - contains the content of the header for left pages. When both left and right have the same content then the right content defines both.

PageScale
Short – the scaling factor in percent for printing the sheet.

PrintAnnotations
Boolean - prints cell comments. Applies only to spreadsheets.

PrintCharts
Boolean - prints charts contained in a sheet. Applies only to spreadsheets.

PrintDownFirst
Boolean - if the contents of a sheet extend across several pages, they are first printed in vertically descending order, and then down the right-hand side. Applies only to spreadsheets.

PrintDrawing
Boolean - prints draw objects. Applies only to spreadsheets.

PrintFormulas
Boolean - prints the formulas instead of the calculated values. Applies only to spreadsheets.

PrintGrid
Boolean - prints the cell gridlines. Applies only to spreadsheets.

PrintHeaders
Boolean - prints the row and column headings. Applies only to spreadsheets.

PrintObjects
Boolean - prints embedded objects. Applies only to spreadsheets.

PrintZeroValues
Boolean - prints the zero values. Applies only to spreadsheets.

PrinterPaperTray
String – name of the printer paper tray that you want to use

RightBorder
Structure (Borderline) – specifications for right-hand line of page border.

RightBorderDistance
Long – distance between right-hand page border and page content in hundredths of a millimeter

RightMargin
Long – width of the right hand page margin in hundredths of a millimeter.

RightPageFooterContent
Object (HeaderFooterContent) - contains the content of the footer for right pages or both left and right if shared.

RightPageHeaderContent
Object (HeaderFooterContent) - contains the content of the header for right pages or both left and right if shared.

ScaleToPages
Short – the number of pages the sheet will be printed on.

ScaleToPagesX
Short – the number of horizontal pages the sheet will be printed on.

ScaleToPagesY
Short – the number of vertical pages the sheet will be printed on.

ShadowFormat
Structure (ShadowFormat) – specifications for shadow of content area of page.

TopBorder
Structure (BorderLine) – specifications for top line of page border.

TopBorderDistance
Long – distance between top page border and page content in hundredths of a millimeter

TopMargin
Long – width of the top page margin in hundredths of a millimeter

Width
Long – width of page in hundredths of a millimeter

PageStyles(*)

A collection of pagestyle objects. Please note that this collection name is case sensitive!:

o_pagestylecollection = Thiscomponent.stylefamilies.PageStyles (valid)

o_pagestylecollection = ThisComponent.StyleFamilies.Pagestyles (invalid)

API Reference, service(s)
com.sun.star.style.StyleFamily

Parent object(s)
StyleFamilies

Methods

getByName(<s_stylename>)
Object - returns the style name object.

getByIndex(<l_Index>)
Object – returns the style object at the specified zero based index.

hasByName(<s_stylename>)
Boolean – true if the collection has the named style.

hasElements
Boolean – true if the collection has styles.

insertByName(<s_StyleName, <o_Style>)
Inserts the style with the specified name.

replaceByName(<s_styleName>, <o_Style)
Replaces the named style with the new style.

removeByName(<s_styleName>)
Removes the style with the specified name.

Properties

Count
Long – returns the number of styles in the collection.

ElementNames
String array – returns a list of all the style objects.

Pane

A window pane object.

API Reference, service(s)
com.sun.star.sheet.SpreadsheetPane

Parent object(s)
Controller

Properties

FirstVisibleColumn
Long - returns the first column that is visible in the pane.

FirstVisibleRow
Long - returns the first row that is visible in the pane.

ReferredCells
Object (CellRange) – returns the range represented in the pane.

VisibleRange
Structure (CellRangeAddress) - returns the address of the cell range that consists of the cells which are visible in the pane.

ParagraphProperties

Describes the style of paragraphs.

API Reference, service(s)
com.sun.star.style.ParagraphProperties

Properties

ParaAdjust
Enumeration (ParagraphAdjust) - determines the adjustment of a paragraph.

ParaBottomMargin
Long - determines the bottom margin of the paragraph in 100th mm.

ParaIsCharacterDistance
Boolean - determines if a distance between asian text, western text or complex text is set.

ParaIsForbiddenRules
Boolean - determines if the rules for forbidden characters at the start or end of text lines are considered.

ParaIsHangingPunctuation
Boolean - determines if hanging punctuation is allowed.

ParaIsHyphenation
Boolean - specifies if automatic hyphenation is applied.

ParaLastLineAdjust
Short - determines the adjustment of the last line.

ParaLeftMargin
Long - determines the left margin of the paragraph in 100th mm.

ParaRightMargin
Long - determines the right margin of the paragraph in 100th mm.

ParaTopMargin
Long - determines the top margin of the paragraph in 100th mm.

ParagraphPropertiesAsian

Contains settings for the style of paragraphs with complex text layout.

API Reference, service(s)
com.sun.star.style.ParagraphPropertiesAsian

Properties

ParaIsCharacterDistance
Boolean - determines if a distance between asian text, western text or complex text is set.

ParaIsForbiddenRules
Boolean - determines if the rules for forbidden characters at the start or end of text lines are considered.

ParaIsHangingPunctuation
Boolean - determines if hanging punctuation is allowed.

ParagraphPropertiesComplex

Contains settings for the style of paragraphs with complex text layout.

API Reference, service(s)
com.sun.star.style.ParagraphPropertiesComplex

Properties

WritingMode
Constant (WritingMode2) - contains the writing direction.

PluginShape

This object encapsulates a plugin. A plugin is a binary object that is plugged into a document to represent a media-type that is not handled natively by the application.

API Reference, service(s)
com.sun.star.drawing.PluginShape
Parent Objects
Drawpage, GroupShape, Shapes

Properties

Property Groups: Shape

PluginMimeType
String - This property specifies the media-type to which this plugin should be registered.
PluginURL
String - This property specifies the url to the binary object.
PluginCommands
Sequence - This sequence contains paramaters that are passed to the application that renders the plugin when it is initialized.

Point

A structure that defines a single cartesian point.

API Reference, structure
com.sun.star.awt.Point

Elements

X
Specifies the X coordinate.
Y
Specifies the Y coordinate.

PolyLineShape

This object is for a polyline shape. A polyline has one or more straight lines.

API Reference, service(s)
com.sun.star.drawing.PolyLineShape
Parent Objects
Drawpage, GroupShape, Shapes

Properties

Property Groups: Shape , LineProperties , Fillproperties , Text , ShadowProperties , PolyPolygonDescriptor

PolyPolygonBezierCoords

Specifies the coordinates for a poly polygon bezier.

API Reference, structure
com.sun.star.drawing.PolyPolygonBezierCoords

Elements

Coordinates
Array of Arrays of Arrays of Structure (Point) - this is quite a deep nesting of points:

```
outerArray = oLineShape.LineEnd.Coordinates   ' list of polygons e.g triangles
innerArray = outerArray(n)   ' list of polygon parts e.g. Lines of a triangle
pointArray = innerArray(n)   ' list of points  e.g. vertices of a triangle
msgbox pointArray(0).X       ' a point  e.g. The x coordinate of one vertice
```

Flags
Array of Arrays of Enumeration (PolygonFlags)

PolypolygonBezierDescriptor

This object describes a polypolygonbezier. A polypolygonbezier consists of multiple bezier polygons combined in one.

API Reference, service
com.sun.star.drawing.PolyPolygonBezierDescriptor

Properties

PolygonKind
Enumeration (PolygonKind) readonly - This is the type of this polygon.

PolyPolygonBezier
Structure (PolyPolygonBezierCoords) - These are the bezier points of this polygon.

Geometry
Structure (PolyPolygonBezierCoords) - These are the untransformed bezier coordinates of this polygon.

PolyPolygonBezierShape

An object that describes a polypolygon bezier shape.

API Reference, service(s)
com.sun.star.drawing.PolyPolygonBezierShape

Parent Objects
Drawpage, GroupShape, Shapes

Properties

Property Groups: Shape , LineProperties , Fillproperties , Text , ShadowProperties , PolyPolygonDescriptor

PolypolygonDescriptor

This service describes a polypolygon. A polypolygon consists of multiple polygons combined in one.

API Reference, service
com.sun.star.drawing.PolyPolygonDescriptor

Properties

PolygonKind
Enumeration (PolygonKind) readonly - This is the type of polygon.

PolyPolygon
Array of Arrays of Arrays of Structure (Point) - These are the reference points for this polygon. This is quite a deep nesting of points:

```
    oPolygon = oLineShape.PolyPolygonDescriptor
    outerArray = oLineShape.PolyPolygon   ' list of polygons e.g triangles
    innerArray = outerArray(n)   ' list of polygon parts e.g. Lines of a triangle
    pointArray = innerArray(n)   ' list of points  e.g. vertices of a triangle
msgbox pointArray(0).X     ' a point  e.g. The x coordinate of one vertice
```

Geometry
Array of Arrays of Structure (Point) - These are the untransformed points of this polygon.

PolyPolygonShape

This object is a polygon shape. A polypolygon has 2 or more straight lines, with the first and last point connected by a straight line.

API Reference, service(s)
com.sun.star.drawing.PolyPolygonShape

Parent Objects
Drawpage, GroupShape, Shapes

Properties

Property Groups: Shape , LineProperties , Fillproperties , Text , ShadowProperties , RotationDescriptor , PolyPolygonDescriptor

PrinterDescriptor

This is a property set that describes a printer by specifying the queue name and some settings.

API Reference, service(s)
com.sun.star.view.PrinterDescriptor

Properties

Name
String - specifies the name of the printer queue to be used. Which printer queues are available, can be figured out with the system library of the used programming language/environment.

PaperOrientation
Enum (PaperOrientation) - specifies the orientation of the paper.

PaperFormat
Enum (PaperFormat) - specifies a predefined paper size or if the paper size is a user-defined size. Setting this property may change the value of PrinterDescriptor and PaperSize

PaperSize
Enum (PaperSize) - specifies the size of the paper in 100th mm. Setting this property may change the value of PrinterDescriptor PaperFormat

IsBusy
Boolean - indicates, whether the printer is busy or not.

CanSetPaperOrientation
Boolean - indicates, whether the printer allows changes to PaperOrientation .

CanSetPaperFormat
Boolean - indicates, whether the printer allows changes to PaperFormat .

CanSetPaperSize
Boolean - indicates if the printer allows changes to PaperSize .

PrintOptions

This is a property set that describes the options for print jobs. These options are only valid for a single print job. They do not change layout or formatting of the document.

API Reference, service
com.sun.star.view.PrintOptions

Properties

Collate
Boolean - advises the printer to collate the pages of the copies.

CopyCount
short - specifies the number of copies to print.

FileName
String - if set, specifies the name of a file to print to.

Pages
String - specifies which pages to print. This range is given as at the user interface. For example: "1-4;10" to print the pages 1 to 4 and 10.

Wait
Boolean - if set to TRUE, the corresponding XPrintable .print() request will be executed synchronous.

PropertyValue

This is an array structure for accessing a property value of a property set.

API Reference, service(s)
com.sun.star.beans.PropertyValue

Elements

Name
String - specifies the name of the property.

Handle
Long - contains an implementation-specific handle for the property.

Value
Any - contains the value of the property or void , if no value is available.

State
Enum (PropertyState) - determines if the value comes from the object itself or from a default and if the value cannot be determined exactly.

Rectangle

This structure specifies a rectangular area by position and size.

API Reference, service(s)
com.sun.star.awt.Rectangle

Elements

X
Long - specifies the x-coordinate.

Y
Long - specifies the y-coordinate.

Width
Long - specifies the width.

Height

Long - specifies the height.

RectangleShape

This object is for a rectangle Shape .

API Reference, service(s)
com.sun.star.drawing.RectangleShape

Parent Objects
Drawpage, GroupShape, Shapes

Properties

Property Groups: Shape , LineProperties , Fillproperties , Text , ShadowProperties

CornerRadius
Long - For Shape s with rounded corners, this is the radius of the corners.

ReplaceDescriptor

An object that describes what and how to replace strings.

API Reference, service(s)
com.sun.star.util.ReplaceDescriptor

Parent object(s)
CellRange , Cell

Properties

SearchString
String – the string to be replaced.

SearchBackwards
Boolean - If true , the search is done backwards in the document.

SearchCaseSensitive
Boolean - If true , the case of the letters is important for the match.

SearchWords
Boolean - If true , only complete words will be found.

SearchRegularExpression
Boolean - If true , the search string is evaluated as a regular expression.

SearchStyles
Boolean - If true , it is searched for positions where the paragraph style with the name of the search pattern is applied.

SearchSimilarity
Boolean - If true , a "similarity search" is performed.

SearchSimilarityRelax
Boolean - If true , all similarity rules are applied together.

SearchSimilarityRemove
Short - This property specifies the number of characters that may be ignored to match the search pattern.

SearchSimilarityAdd
Short - specifies the number of characters that must be added to match the search pattern.

SearchSimilarityExchange
Short - This property specifies the number of characters that must be replaced to match the search pattern.

ReplaceString
String – the string to replace with.

Row

An object that represents a single row.

API Reference, service(s)
com.sun.star.table.TableRow

Parent object(s)
Rows

Methods

Refer to the methods of a CellRange

Properties

Refer to the properties of a CellRange

Height
Long - contains the height of the row (in 1/100 mm).

IsStartOfNewPage
Boolean - is true , if there is a manual vertical page break attached to the row.

IsVisible
Boolean - is true , if the row is visible.

OptimalHeight
Boolean - is true , if the row always keeps its optimal height.

RowLabelRanges(*)

A collection of row label ranges in the document.

API Reference, service(s)
com.sun.star.sheet.XLabelRanges

Parent object(s)
Document

Methods

addNew(<struct_CellRangeAddress-titlearea>,<struct_CellRangeAddress-dataarea>
adds a new label range to the collection

createEnumeration
Returns and enumerated list of row label ranges.

getByIndex(<l_Index>)
Object – returns the row label range at the specified zero based index.

removeByIndex(<l_Index>)
removes a label range from the collection.

Properties

Count
Long – returns the number of row label ranges in the collection.

RowLabelRange

An object that represents a cell area that contains labels and values related to the row labels.

API Reference, service(s)
com.sun.star.sheet.LabelRange

Parent object(s)
RowLabelRanges

Properties

LabelArea
Structure (CellRangeAddress) - specifies the cell range address that contains the labels.
DataArea
Structure (CellRangeAddress) - specifies the cell range address for which the labels are valid.

Rows(*)

This is a collection of row objects.

API Reference, service(s)
com.sun.star.table.TableRows
Parent object(s)
Sheet,Cell,CellRange

Methods

createEnumeration()
Returns an enumerated list of all rows in the collection.
getByIndex(<l_index>)
Object – returns the row at the specified zero based index.
getByName(<s_ColumnName>)
Object – returns the row with the matching name.
HasByName(<s_ColumnName>)
Boolean – true if the collection has a row of that name.
insertByIndex(<l_Position>, <l_Quantity>)
inserts <quantity> rows at the zero based <position>
removeByIndex(<l_Position>, <l_Quantity>)
removes <quantity> rows starting with the zero based <position>

Properties

Count
Long – The number of rows in the collection.
ElementNames
String Array - returns a list of all the row names.

Scenario

API Reference, service(s)
com.sun.star.sheet.Scenario
Parent object(s)
Scenarios

Methods

All methods of a sheet

Properties

All properties of a Sheet

Scenarios(*)

API Reference, service(s)

com.sun.star.sheet.Scenarios

Parent object(s)
Sheet

Methods

addNewbyName(<s_ScenarioName>, <struct_CellRangeAddress array> , <s_comment>)
Adds a new scenario to the collection.

createEnumeration
Returns an enumerated list of scenario objects.

getByName(<s_scenarioname>)
Object - returns the named scenario object.

getByIndex(<l_Index>)
Object – returns the scenario object at the specified zero based index.

hasByName(<s_scenarioname>)
Boolean – true if the collection has the named scenario.

hasElements
Boolean – true if the collection has scenarios.

removeByName(<s_scenarioName>)
Removes the scenario with the specified name.

Properties

Count
Long – returns the number of scenarios in the collection.

ElementNames
String array – returns a list of all the scenario objects.

ShadowFormat

This is a structure that defines a shadow area on a page.

API Reference, service(s)
com.sun.star.table.ShadowFormat

Elements

Location
Enum (ShadowLocation) - contains the location of the shadow.

ShadowWidth
Short - contains the size of the shadow.

IsTransparent
Boolean - is true , if shadow is transparent.

Color
Enum (Color) - contains the color value of the shadow.

ShadowProperties

This is a set of properties to describe the style for rendering a shadow.

ApI reference, Service
com.sun.star.drawing.ShadowProperties

Properties

Shadow
Boolean - enables/disables the shadow of a Shape .

ShadowColor

Typedef (Color) - This is the color of the shadow of this Shape .

ShadowTransparence
Short - This defines the degree of transparence of the shadow in percent.

ShadowXDistance
Long - This is the horizontal distance of the left edge of the Shape to the shadow.

ShadowYDistance
Long - This is the vertical distance of the top edge of the Shape to the shadow.

Shape

An object that describes a single chart, tool object or graphic object on a sheet. The methods and properties described here are applicable to all shape types. Please refer to the shape description for specific methods and properties. All possible shapes are listed under the ShapeType property.

API Reference, service(s)
com.sun.star.drawing.Shape

Parent Objects
Shapes , GroupShape , DrawPage

Properties

LayerID
short - This is the ID of the Layer to which this Shape is attached.

LayerName
string - This is the name of the Layer to which this Shape is attached.

MoveProtect
boolean - With this set to true , this Shape cannot be moved interactively in the user interface.

Name
string - This is the name of this Shape .

Position
Structure (Point) - the position of the top left edge in 100/th mm

Printable
boolean - If this is false , the Shape is not visible on printer outputs.

Shapetype
String – returns the type of the shape. This determines the methods and properties in addition to the generic ones described here. Possible shape types are:
AppletShape
CaptionShape
ClosedBezierShape
ConnectorShape
ControlShape
CustomShape
EllipseShape
GraphicObjectShape
GroupShape
LineShape
MeasureShape
OLE2Shape
OpenBezierShape
PageShape
PluginShape
PolyLineShape
PolyPolygonBezierShape
PolPolygonShape
RectangleShape
TextShape

ShapeUserDefinedAttributes

Object (AttributeContainer) - this property stores a collection of xml attributes. They will be saved to and restored from automatic styles inside xml files.

Size
Structure (Size) - the size of this object in 100/th mm

SizeProtect
boolean - With this set to true , this Shape may not be sized interactively in the user interface.

Style
Object (Style) - this property lets you get and set a style for this shape.

Transformation
Structure (HomogenMatrix3) - this property lets you get and set the transformation matrix for this shape. The transformation is a 3x3 homogen matrix and can contain translation, rotation, shearing and scaling.

ZOrder
long - is used to query or change the ZOrder of this Shape .
Example
sTheTypeShape = ThisComponent.Sheets(0).Drawpage(0).ShapeType
oControl = ThisComponent.Sheets(0).Drawpage(0).Control
Example assumes the first shape on the first sheet is a command button.

Shapes

A collection of shapes on a page/sheet. The actual API object of a spreadsheet shapes collection is a drawpage. A drawpage allows shapes to be grouped and ungrouped.

API Reference, service(s)
com.sun.star.drawing.Shapes

Parent Objects
(Drawpage), GroupShape

Methods

add(<o_Shape>)
Adds a shape to the collection.

getByIndex(<l_Index>)
Object – returns a shape object.

hasElements
Boolean – true if the collection has styles.

remove(<o_Shape>)
Removes the shape object from the collection.

Properties

Count
Long – returns the number of shapes in the collection.

Sheet

A single spreadsheet sheet object. This is also the range of all cells on the sheet.

API Reference, service(s)
com.sun.star.sheet.SpreadSheet

Parent object(s)
Sheets

Methods

Includes all methods of a CellRange.

autoOutline(<struct_CellRangeAddress>)

Creates outline groups from formula references in a range.

clearOutline
Removes all outline groups from the sheet.

copyRange(<struct_CellAddress>, <struct_CellRangeAddress>)
This copies the range of cells defined by the structure <CellRangeAddress> to the origin defined by the structure <CellAddress>.
NB this works irrespective of which sheet object the method is applied to !

createCursor
Object - creates a *CellCursor* including the whole spreadsheet.

createCursorByRange(<o_CellRange>)
Object - creates a *CellCursor* to travel in the given range context.

getCellByPosition(<l_Column>, <l_Row>)
Object - Returns the *cell* object on the sheet at the specified position.

getCellRangeByPosition(<l_Left>, <l_Top>, <l_Right>, <l_Bottom>)
Object – returns a *CellRange* object as for the specified position.

getCellRangeByName(<s_Range>)
Object - Returns the range object for the specified address or range name.

group(<struct_CellRangeAddress>, <enum_Orientation>)
Creates an outline group of rows or columns.

hideDetail(<struct_CellRangeAddress>)
Collapses an outline group.

insertCells(<struct_CellRangeAddress>, <Enum_CellInsertMode>)
This inserts cells at the location and with the size defined by the structure <cellrangeaddress>.
The <CellInsertMode> values define how the existing cells are moved.
NB this works irrespective of which sheet object the method is applied to !

Isprotected
Boolean – true if the sheet is protected

moveRange(<struct_CellAddress>,< struct_CellRangeAddress>)
This moves the range of cells defined by the structure <CellRangeAddress> to the origin defined by the structure <CellAddress>.
NB this works irrespective of which sheet object the method is applied to !

protect(<password string>)
Protects the sheet with given password.

removeAllManualPageBreaks
removes all manual page breaks on the sheet.

removeRange(<struct_CellRangeAddress>, <enum_CellDeleteMode>)
This deletes cells at the location and with the size defined by the structure <cellrangeaddress>.
The <CellDeleteMode> values determines how the deleted area is closed.
NB this works irrespective of which sheet object the method is applied to !

showDetail(<struct_CellRangeAddress>)
Reopens an outline group.

showLevel(<l_Level>, <enum_Orientation>)
Shows all outlined groups below a specific level.

ungroup(<struct_CellRangeAddress>, <enum_Orientation>)
Removes outline groups of rows or columns. In the specified range, all outline groups on the innermost level are removed.

unProtect(<password string>)
Unprotects the sheet that has <password string> as the password.

Properties

Includes all properties of a CellRange.

AutomaticPrintArea
Boolean – specifies if the sheet has an automatic print area.

Annotations
Object (Annotations) - returns the collection of cell annotations.

Charts
Object (Charts) – returns a collection of Charts.

ColumnPageBreaks
Structure (TableBreakPageData) Array - returns a sequence of descriptions of all horizontal page breaks on the sheet.

Columns
Object (Columns) – returns a collection of all the columns in the sheet

DataPilotTables
Object (DataPilotTables) - Returns the collection of data pilot (pivot) tables.

DrawPage
Object (DrawPage) – returns a single draw page (Shapes collection ?) object

IsVisible
Boolean - Specifies if the sheet is visible

Name
String - the name of the sheet.

PageStyle
String - Specifies the pagestyle of the sheet.

PrintAreas
Structure (CellRangeAddress) - defines the print areas of the sheet.

PrintTitleColumns
Boolean - specifies whether the title columns are repeated on all subsequent print pages to the right.

PrintTitleRows
Boolean - specifies whether the title rows are repeated on all subsequent print pages to the bottom.

RowPageBreaks
Structure (TableBreakPageData) array - returns a sequence of descriptions of all vertical page breaks on the sheet.

Scenarios
Object (Scenarios) - returns the collection of scenarios.

TitleColumns
Structure (CellRangeAddress) - specifies a range of columns as title columns range.

TitleRows
Structure (CellRangeAddress) - specifies a range of rows as title rows range.

Sheets (*)

This is a collection of sheet objects.

API Reference, service(s)
com.sun.star.

Parent object(s)
Document

Methods

copyByName(<s_Sourcesheet>,<s_Destinationsheet>,<l_Index>)
Copies a sheet within the collection and inserts it at the zero based index.

CreateEnumeration
Returns an enumerated list of all sheets.

getByIndex(<l_Index>)
Object - Returns the sheet object at the specified zero based index.

getByName(<s_Sheetname>)

Object - Returns the sheet object with the name <sheetname>

hasByName(<s_Sheetname>)
True if the collection has a sheet of that case insensitive name.

insertNewByName(<s_Sheetname>,<l_Index>)
Inserts a new sheet into the collection at the specified zero based index.

moveByName(<s_Sheetname>,<l_Index>)
Moves a sheet within the collection to the zero based index.

removeByName(<s_Sheetname>)
removes the sheet with the specified name.

Properties

Count
Integer – the number of sheets in the collection

ElementNames
String Array - returns a list of all the sheet names.

SheetFilterDescriptor

A collection that represents a description of how a cell range is to be filtered.

The descriptor contains properties and a collection of filter conditions (filter fields) which control the behaviour of a filter operation.

API Reference, service(s)
com.sun.star.table.SheetFilterDescriptor

Parent object(s)
DatabaseRange, Sheet , CellRange

Properties

CopyOutputData
Boolean - specifies if the filtered data should be copied to another position in the document. If not then the rows that don't match the filter are hidden.

ContainsHeader
Boolean - specifies if the first row (or column) contains headers which should not be filtered.

FilterFields
Structure (TableFilterField) array - returns the collection of filter fields.

IsCaseSensitive
Boolean - specifies if the case of letters is important when comparing entries.

MaxFieldCount
Long - readonly,returns the maximum number of filter fields in the descriptor.

Orientation
Enumeration (TableOrientation) - specifies if columns or rows are filtered.

OutputPosition
Structure (CellAddress) - specifies the position where filtered data are to be copied.

SaveOutputPosition
Boolean - specifies if the SheetFilterDescriptor::OutputPosition position is saved for future calls.

SkipDuplicates
Boolean - specifies if duplicate entries are left out of the result.

UseRegularExpressions
Boolean - specifies if the TableFilterField::StringValue strings are interpreted as regular expressions.

SheetLink

Represents a sheet link. A sheet link contains the source data of linked sheets, i.e. the URL and

sheet name of the external document.

API Reference, service(s)
com.sun.star.sheet.SheetLink

Parent object(s)
SheetLinks

Methods

refresh
Refreshes the link with data from the source.

Properties

Filter
String - specifies the name of the filter needed to load the source document.

FilterOptions
String - specifies the filter options needed to load the source document.

Name
String - the name of the sheet link.

Url
String - specifies the URL of the source document.

SheetLinks(*)

This is a collection of sheet links.

API Reference, service(s)
com.sun.star.sheet.SheetLinks

Parent object(s)
Document

Methods

createEnumeration
An enumerated list of DDE Link objects.

getByIndex(<l_Index>)
Object (SheetLink) - returns the sheet link object at the index within the collection.

getByName(<s_name>)
Object - Returns the link with the name <s_name>

hasByName(<s_name>)
Boolean – true if the sheet link exists within the collection.

hasElements
Boolean – true if there are sheet Links in the collection

Properties

Count
Integer – the number of DDE Links in the collection

ElementNames
String Array - returns a list of all the sheet link names.

SheetSortDescriptor

A description of how a cell range is to be sorted.

API Reference, service(s)
com.sun.star.table.SheetSortDescriptor2

Properties

BindFormatsToContent
Boolean specifies if cell formats are moved with the contents they belong to.

ContainsHeader
Boolean - specifies whether the first row or column (depending on *TableSortDescriptor* Orientation) is a header which should not be sorted.

CopyOutputData
Boolean specifies if the sorted data should be copied to another position in the document.

IsSortColumns
Boolean - specifies if the columns or rows are to be sorted.

IsUserListEnabled
Boolean specifies if a user defined sorting list is used.

MaxSortFieldsCount
Long - Readonly, contains the maximum number of sort fields the descriptor can hold.

OutputPosition
Structure (CellAddress) - specifies the position where sorted data are to be copied.

SortFields
Structure(TableSortField) array - specifies a list of individual sort fields.

UserListIndex
Long specifies which user defined sorting list is used.

Size

This is a structure that specifies the 2-dimensional size of an area using width and height.

API Reference, service(s)
com.sun.star.awt.Size

Elements

Width
Long - specifies the width.

Height
Long - specifies the height.

StarDesktop(*)

The open office application. For all intents and purposes this is the top object although it is actually within a frame object (from what I can make out!).

API Reference, service(s)
com.sun.star.frame.Desktop

Parent object(s)
Frame (not documented here)

Methods

loadComponentFromURL(<s_Filename>, <s_TargetFrameName>, <const_FrameSearchFlag>, <seq_MediaDescriptor>)
Loads a component/document specified by an URL into the specified new or existing frame and returns a document object. The parameter <sTargetFrameName> specifies the name of the frame to view the document in. If a frame with the specified name already exists, it is used, otherwise it is created. There exist some special targets which never can be used as real frame names: "_blank" always creates a new frame
"_default" special UI functionality
(e.g. detecting of already loaded documents, using of empty frames of creating of new top frames as fallback)

"_self", means frame himself
"" (!)
"_parent" address direct parent of frame
"_top" indicates top frame of current path in tree
"_beamer" means special sub frame

If a document is “loaded” with MediaDescriptor property “AsTemplate” equal to true then a new document is created.

terminate()

Tries to terminate the desktop

Properties

Components

Object (Documents) - Returns a collection of all the components/documents for the application.

CurrentComponent

Object (Document) - Returns the top most document in the document stack. Please note that this is not necessarily the one where the VB code is executing.

CurrentFrame

Object (Frame) - provides read access to the frame which contains the current component. Please note frame objects are not documented here.

StyleFamilies (*)

A collection of style collections(!).

StyleFamiles or getStyleFamiles are methods of document.

API Reference, service(s)

com.sun.star.style.StyleFamilies

Parent object(s)

Document

Methods

getByName(“<s_stylefamilyname>”)

Object - returns the style collection with the corresponding name. These are:
CellStyles
CharacterStyles
FrameStyles
NumberingStyles
PageStyles
ParagraphStyles
ShapeStyles
NB not all styles exist within all collections. It looks like only Cellstyles and PageStyles exist for a spreadsheet document.

getByIndex(<l_Index>)

Object – returns the style family at the specified zero based index.

hasByName(“<s_stylefamilyname>”)

Boolean – true if the collection has the syle family.

hasElements

Boolean – true if the collection has styles

Properties

Count

Long – returns the number of style families in the collection.

ElementNames

String array – returns a list of all the style families.

SubTotalColumn

A structure that describes how a single data column is treated when creating subtotals.

API Reference, structure
com.sun.star.sheet.SubTotalColumn

Elements

Column
Long - the index of the column inside the source data area.

Function
Enumeration (GeneralFunction) - specifies what kind of subtotals are calculated.

SubTotalDescriptor

Represents a collection of how subtotals are created.

The descriptor contains properties and a collection of subtotal fields which control the behaviour of operation.

API Reference, service(s)
com.sun.star.sheet.SubTotalDescriptor

Parent object(s)
DatabaseRange, CellRange

Methods

addNew(<struct_SubTotalColumn>, <l_GroupColumn>)
adds a subtotal field definition to the descriptor.

clear
removes all subtotal field definitions from the descriptor.

CreateEnumeration
Creates an enumerated list of sub total descriptors.

haselements
Boolean – true if there are any subtotals in the collection.

Properties

BindFormatsToContent
Boolean - specifies if cell formats are moved with the contents they belong to.

EnableSort
Boolean – specifies if the contents of the fields will be sorted to groups while performing a subtotal operation.

EnableUserSortList
Boolean - specifies if a user defined sorting list is used.

InsertPageBreaks
Boolean - specifies if page breaks are inserted after each group change.

IsCaseSensitive
Boolean - specifies if the case of letters is important when comparing entries.

MaxFieldCount
Long - Readonly, returns the maximum number of subtotal fields the descriptor can hold.

SortAscending
Boolean - specifies the sorting order if SubTotalDescriptor::EnableSort is set to true .

UserSortListIndex
Long - specifies which user defined sorting list is used.

TableBorder

This is a structure that defines the border for a range of cells.

API reference, structure: com.sun.star.table.TableBorder

API Reference, service
com.sun.star.sheet.TableBorder

Parent object(s)
Sheet , Row , Column , Cell , CellRange

Elements

TopLine
Struct (Borderline) - determines the line style at the top edge.

IsTopLineValid
Boolean - specifies whether the value of TableBorder::TopLine is used.

BottomLine
Struct (Borderline) - determines the line style at the bottom edge.

IsBottomLineValid
Boolean - specifies whether the value of TableBorder::BottomLine is used.

LeftLine
Struct (BorderLine) - determines the line style at the left edge.

IsLeftLineValid
Boolean - specifies whether the value of TableBorder::LeftLine is used.

RightLine
Struct (BorderLine) - determines the line style at the right edge.

IsRightLineValid
Boolean - specifies whether the value of TableBorder::RightLine is used.

HorizontalLine
Struct (BorderLine) - determines the line style of horizontal lines for the inner part of a cell range.

IsHorizontalLineValid
Boolean - specifies whether the value of TableBorder::HorizontalLine is used.

VerticalLine
Struct (BorderLine) - determines the line style of vertical lines for the inner part of a cell range.

IsVerticalLineValid
Boolean - specifies whether the value of TableBorder::VerticalLine is used.

Distance
Short - contains the distance between the lines and other contents.

IsDistanceValid
Boolean - specifies whether the value of TableBorder::Distance is used.

TableConditionalEntry

This object represents a single condition in a conditional format.

API Reference, service(s)
com.sun.star.sheet.TableConditionalEntry

Parent Objects
ConditionalFormat , ConditionalFormalLocal

Properties

Operator
Enumeration (ConditionOperator) - contains the operation to perform for this condition.

Formula1

String - contains the value or formula for the operation.

Formula2
String - contains the second value or formula for the operation (used with ConditionOperator::BETWEEN or ConditionOperator::NOT_BETWEEN operations).

SourcePosition
Object (CellAddress) - contains the base address for relative cell references in formulas.

StyleName
String - contains the name of the cell style used by this condition.

TableFilterField

A structure that describes a single condition in a filter descriptor.

API Reference, structure
com.sun.star.sheet.TableFilterField

Elements

Connection
Enumeration (FilterConnection) - specifies how the condition is connected to the previous condition.

Field
Long - specifies which field (column) is used for the condition.

Operator
Enumeration (FilterOperator) - specifies the type of the condition.

IsNumeric
Boolean - selects whether the TableFilterField::NumericValue or the TableFilterField::StringValue is used.

NumericValue
Double - specifies a numeric value for the condition.

StringValue
String - specifies a string value for the condition.

TablePageBreakData

Describes a page break in a spreadsheet.

API Reference, structure
com.sun.star.sheet.TablePageBreakData

Elements

Position
Long - the position (column or row index) of the page break.

ManualBreak
Boolean - is true for a manual page break, false for an automatic one.

TableSortField

Structure that describes how to sort a single field (row/column) in a tables sort descriptor.

API Reference, structure
com.sun.star.table.TableSortField

Elements

Field
Long - index of the row or column in the table to be sorted; 0-based.

IsAscending
Boolean - true if data are sorted in ascending order, false if in descending order.

IsCaseSensitive
Boolean - specifies if the case of letters is important when comparing entries.

FieldType
Enumeration (TableSortFieldType) - type of contents in the field.

CollatorLocale
Structure (Locale) - the locale used by the collator when comparing/sorting text.

CollatorAlgorithm
String - the algorithm used by the collator when comparing/sorting text.

TextShape

This object is a text shape.

API Reference, service(s)
com.sun.star.drawing.TextShape

Parent Objects
Drawpage, GroupShape, Shapes

Properties

Property Groups: Shape , LineProperties , Fillproperties , ShadowProperties , RotationDescriptor , CharacterProperties, CharacterPropertiesAsian , CharacterPropertiesComplex , ParagraphProperties , ParagraphPropertiesAsian , ParagraphPropertiesComplex

CornerRadius
Long - This is the radius of the corners.

String
String – the textual content.

UserDefinedAttributes

This is a list of properties defined by the user.

API Reference, service(s)
com.sun.star.sheet.UnoAttributeContainer

Parent object(s)
Cell

Methods

getByName(<s_name>)
Object - Returns the element with the name <name>

hasByName(<s_name>)
True if the collection has an element of that case insensitive name.

hasElements
Boolean – true if there are any elements in the collection.

insertByName(<s_Name, <o_Object>)
Inserts the element with the specified name.

replaceByName(<s_Name>, <o_object)
Replaces the named element.

removeByName(<s_name>)
removes the element with the specified name.

Properties

Count

Integer – the number of elements in the collection

ElementNames
String Array - returns a list of all the element names.

UniqueCellFormatRanges (*)

Represents a collection of equal-formatted cell range collections. All cells inside a cell range collection (a member of this collection) have the same formatting attributes.

API Reference, service(s)
com.sun.star.sheet.UniqueCellFormatRanges

Parent object(s)
CellRange

Methods

getByIndex(<l_index>)
Object (CellRanges) – a collection of cell ranges with the same format.

CreateEnumeration
An enumerated list of cell ranges

Properties

Count
Integer – the number of format ranges in the collection

Validation (*)

This object represents the data validation settings for a cell or cell range.

API Reference, service(s)
com.sun.star.sheet.TableValidation

Parent Objects
Cell , CellRange

Properties

Type
Enumeration (ValidationType) - specifies the type of validation.

ShowInputMessage
Boolean - specifies if an input message is shown when the cursor is in a cell with these validation settings.

InputTitle
String - specifies the title of the window showing the input message.

InputMessage
String - specifies the text of the input message.

ShowErrorMessage
Boolean - specifies if an error message is displayed when invalid data is entered.

ErrorTitle
String - specifies the title of the window showing the error message.

ErrorMessage
String - specifies the text of the error message.

IgnoreBlankCells
Boolean - specifies if blank cells should be allowed.

ErrorAlertStyle
Enumeration (ValidationAlertStyle) - specifies the style of the error message.

ShowList

Constant (TableValidationVisibility) - specifies if the list of possible values should be shown on the cell and how. See also TableValidationVisibility

ValidationLocal (*)

This object represents the data validation settings for a cell or cell range.

API Reference, service(s)
com.sun.star.sheet.TableValidation

Parent Objects
Cell , CellRange

Properties

All the properties of Validation.

Window

See Controller.

XForbiddenCharacters

API Reference, service(s)
com.sun.star.i18n.XForbiddenCharacters

Parent object(s)
Document

Methods

getForbiddenCharacters(<struct_Locale>)
Structure (ForbiddenCharacters) - returns the forbidden characters for a given locale.

hasForbiddenCharacters
Boolean - determines if forbidden characters are set for a given locale.

setForbiddenCharacters(<struct_Locale> , <struct_ForbiddenCharacters>)
Sets the forbidden characters for a given Locale.

removeForbiddenCharacters(<struct_Locale>)
Removes the setting of forbidden characters for a given locale.

Enumerations & Constants

These are used as follows, constant:

```
iFlags = com.sun.star.sheet.CellFlags.STRING +_
      com.sun.star.sheet.CellFlags.HARDATTR
```

The above example returns the value for the two flags which when used to clear the contents of a range will remove the string values and cell formatting.

Enumeration:

```
oSheet.removeRange(oCellRangeAddress, com.sun.star.sheet.CellDeleteMode.UP)
```

This removes a range of cells and the ones below the removed range are moved up. Note how only a single value is used.

I find this is the easiest way to consider these values. Constants are like unique flags/bits which can be added together. Whereas an enumeration is a list of values where only one can be used.

Please note that these names are case sensitive.

BitmapMode

The BitmapMode selects an algorithm for filling an area with a bitmap.

Enumeration: com.sun.star.drawing.BitmapMode

REPEAT	the bitmap is repeated over the fill area.
STRETCH	the bitmap is stretched to fill the area.
NO_REPEAT	the bitmap is painted in its original or selected size.

Border

Determines which border of a cell or range is selected

Enumeration: com.sun.star.sheet.Border

TOP	selects the top border.
BOTTOM	selects the bottom border.
RIGHT	selects the right border.
LEFT	selects the left border.

CellContentType

Used to determine the type of contents in a cell.

Enumeration: com.sun.star.table.CellContentType

EMPTY	no value
VALUE	number
TEXT	strings
FORMULA	formula

Cellflags

These constants select different types of cell contents. The values can be combined. They are used to insert, copy, or delete contents.

Constant list : com.sun.star.sheet.CellFlags

VALUE	1 - numerical values that are not formatted as date or time
DATETIME	2 - numerical values that are formatted as date or time
STRING	4 - strings

ANNOTATION	8 - comments that are linked to cells
FORMULA	16 - formulas
HARDATTR	32 - direct formatting of cells
STYLES	64 - indirect formatting
OBJECTS	128 - drawing objects that are connected to cells
EDITATTR	256 - character formatting that only applies to parts of the cells

CellDelete Mode

Used to specify how remaining cells are moved when cells are deleted.

Enumeration: com.sun.star.sheet.CellDeleteMode

NONE	the current values remain in their current position.
UP	the cells at and below the insert position are moved upwards.
LEFT	the cells at and to the right of the insert position are moved to the left.
ROWS	the rows after the insert position are moved upwards.
COLUMNS	the columns after the insert position are moved to the left.

CellHorijustify

Specifies how cell contents are aligned horizontally.

Enumeration: com.sun.star.table.CellHoriJustify

STANDARD	default alignment is used (left for numbers, right for text).
LEFT	contents are aligned to the left edge of the cell.
CENTER	contents are horizontally centered.
RIGHT	contents are aligned to the right edge of the cell.
BLOCK	contents are justified to the cell width.
REPEAT	contents are repeated to fill the cell.

CellInsertMode

Used to specify how cells are moved when new cells are inserted.

Enumeration: com.sun.star.sheet.CellInsertMode

NONE	the current values remain in their present position.
DOWN	the cells at and under the insert position are moved downwards.
RIGHT	the cells at and to the right of the insert position are moved to the right.
ROWS	the rows after the insert position are moved downwards.
COLUMNS	the columns after the insert position are moved to the right.

CellOrientation

Specifies the orientation of a cell.

Enumeration: com.sun.star.table.CellOrientation

STANDARD	contents are printed from left to right.
TOPBOTTOM	contents are printed from top to bottom.
BOTTOMTOP	contents are printed from bottom to top.
STACKED	contents are printed from top to bottom with individual characters in normal (horizontal) orientation.

CellVertJustify

Specifies how cell contents are aligned vertically.

Enumeration: com.sun.star.table.CellVertJustify

STANDARD	default alignment is used.

TOP	contents are aligned with the upper edge of the cell.
CENTER	contents are aligned to the vertical middle of the cell.
BOTTOM	contents are aligned to the lower edge of the cell.

ChartDataCaption

These values specify how the captions of data points are displayed.

Constant (Long) : com.sun.star.chart.ChartDataCaption

NONE	0 -No captions are displayed.
VALUE	1 - The caption contains the value of the data point in the number format of the axis that is attached to the respective data series.
PERCENT	2 - The caption contains the value of the data point in percent of all data points of one category.
TEXT	4 - The caption contains the category name of the category to which a data point belongs.
SYMBOL	16 - The symbol of data column/row is additionally displayed in the caption.

ChartErrorCategory

Specifies the category of error indicators.

Enumeration : com.sun.star.chart.ChartErrorCategory

NONE	error indicators are not displayed.
VARIANCE	displays error indicators for the variance of the data row.
STANDARD_DEVIATION	displays error indicators for the standard deviation (square root of variance) of the data row.
PERCENT	The length of the error indicators is calculated for each data point by taking the percentage given as ChartStatistics::PercentageError of its value.
ERROR_MARGIN	The length of the error indicators for all data points is calculated by taking the percentage given as ChartStatistics::ErrorMargin of the largest data point value.
CONSTANT_VALUE	displays the same lower and upper error indicators for all data points.

ChartErrorIndicatorType

Specifies how the error is indicated.

Enumeration : com.sun.star.chart.ChartErrorIndicatorType

NONE	displays no error indicators.
TOP_AND_BOTTOM	displays both the upper and lower values.
UPPER	displays only the upper value.
LOWER	displays only the lower value.

ChartLegendPosition

Specifies one of the default positions of the legend in relation to the diagram.

Enumeration: com.sun.star.chart.ChartLegendPosition

NONE	no chart legend is displayed.
LEFT	displays the chart legend on the left side of the diagram.
TOP	displays the chart legend above the diagram.
RIGHT	displays the chart legend on the right side of the diagram.
BOTTOM	displays the chart legend beneath the diagram.

ChartRegressionCurveType

specifiest the type of the regression curve to be displayed.

Enumeration : com.sun.star.chart.ChartRegressionCurveType

NONE	displays no regression curve.
LINEAR	displays a linear regression curve.
LOGARITHM	displays a linear logarithmic regression curve.
EXPONENTIAL	displays an exponential regression curve.
POWER	displays a regression curve using a power function.

ChartSymbolType

These values specify the type of the symbol used for data points. This only applies to diagrams that use symbols like line diagrams. The default symbols are currently:

Symbol 0 a square
Symbol 1 a diamond
Symbol 2 a triangle pointing down
Symbol 3 a triangle pointing up
Symbol 4 a triangle pointing right
Symbol 5 a triangle pointing left
Symbol 6 a bow tie
Symbol 7 a rotated bow tie

Constant (Long) : com.sun.star.chart.ChartSymbolType

NONE	-3 - No symbol is used.
AUTO	-2 - The symbol is selected automatically. The size of symbol will be dynamic and the type depends on the data row number.
BITMAPURL	-1 - Take a Bitmap from a URL and use this as symbol.
SYMBOL0	0 - The default symbol for row 0 is used.
SYMBOL1	1 - The default symbol for row 1 is used.
SYMBOL2	2 - The default symbol for row 2 is used.
SYMBOL3	3 - The default symbol for row 3 is used.
SYMBOL4	4 - The default symbol for row 4 is used.
SYMBOL5	5 - The default symbol for row 5 is used.
SYMBOL6	6 - The default symbol for row 6 is used.
SYMBOL7	7 - The default symbol for row 7 is used.

CharSet

Deprecated but properties that reference it seem to have no alternate.

These values are used to specify the characters which are available in a font and their codes.

Constant (Short) : com.sun.star.CharSet

DONTKNOW	0 - specifies an unknown character set.
ANSI	1 - specifies the ANSI character set.
MAC	2 - specifies the Apple Macintosh character set.
IBMPC_437	3 - specifies the IBM PC character set number 437.
IBMPC_850	4 - specifies the IBM PC character set number 850.
IBMPC_860	5 - specifies the IBM PC character set number 860.
IBMPC_861	6 - specifies the IBM PC character set number 861.
IBMPC_863	7 - specifies the IBM PC character set number 863.
IBMPC_865	8 - specifies the IBM PC character set number 865.
SYSTEM	9 - specifies the system character set.
SYMBOL	10 - specifies a set of symbols.

CircleKind

This enumeration defines a circle.

Enumeration: com.sun.star.drawing.CircleKind

FULL	a full circle
SECTION	a circle with a cut connected by a line
CUT	a circle with a cut connected by two lines
ARC	a circle with an open cut

Color

Describes an RGB color value with an optional alpha channel.

Typedef (Long) : com.sun.star.util.Color

This is a 4 byte value, in order from low byte to high byte:

byte0 - blue

byte1 - green

byte2 - red

byte3 – alpha channel

ColorMode

The ColorMode defines the output style of colors for a graphic.

Enumeration: com.sun.star.drawing.ColorMode

STANDARD	the graphic is rendered in the default color style of the output device,
GREYS	the graphic is rendered in greyscales on the output device,
MONO	the graphic is rendered in black and white only,
WATERMARK	the graphic is rendered in a watermark like style,

ConditionOperator

Is used to specify the type of XSheetCondition .

Enumeration : com.sun.star.sheet.ConditionOperator

NONE	no condition is specified.
EQUAL	value has to be equal to the specified value.
NOT_EQUAL	the value must not be equal to the specified value.
GREATER	the value has to be greater than the specified value.
GREATER_EQUAL	the value has to be greater than or equal to the specified value.
LESS	the value has to be less than the specified value.
LESS_EQUAL	the value has to be less than or equal to the specified value.
BETWEEN	the value has to be between the two specified values.
NOT_BETWEEN	the value has to be outside of the two specified values.
FORMULA	the specified formula has to give a non-zero result.

ConnectorType

The ConnectorType specifies the appearance of a connector.

Enumeration : com.sun.star.drawing.ConnectorType

STANDARD	the connector is drawn with three lines, with the middle line perpendicular to the other two
CURVE	the ConnectorShape is drawn as a curve
LINE	the ConnectorShape is drawn as a straight line
LINES	the connector is drawn with three lines

Dashstyle

This enumeration defines the style of a dash on a line.

Enumeration: com.sun.star.drawing.Dashstyle

RECT	the dash is a rectangle
ROUND	the dash is a point
RECTRELATIVE	the dash is a rectangle, with the size of the dash given in relation to the length of the line
ROUNDRELATIVE	the dash is a point, with the size of the dash given in relation to the length of the line

DataFieldLayoutMode

Describes the layout mode of the data field

Constant (Long) : com.sun.star.sheet.DataFieldLayoutMode

TABULAR_LAYOUT	0 - Tabular layout mode is the layout, where each item's name is on the same row as the first item from the following field. Subtotals are always shown below an item's data in this mode.
OUTLINE_SUBTOTALS_TOP	1 - In outline layout mode, the items from the following field start in the row below an item's name, like in traditional database reports.
OUTLINE_SUBTOTALS_BOTTOM	2 - In outline layout mode, the items from the following field start in the row below an item's name, like in traditional database reports.

DataImportMode

Used to specify which database contents are imported.

Enumeration : com.sun.star.sheet.DataImportMode

NONE	nothing is imported.
SQL	a SQL query string is supplied.
TABLE	the name of a database table is supplied.
QUERY	the name of a database query is supplied.

DataPilotFieldGroupBy

These constants select different types of DataPilotFieldGroups.

Constant (Long) : com.sun.star.sheet.DataPilotFieldGroupBy

SECONDS	1
MINUTES	2
HOURS	4
DAYS	8
MONTHS	16
QUARTERS	32
YEARS	64

DataPilotFieldOrientation

Used to specify where a field in a data pilot table is laid out.

Enumeration : com.sun.star.sheet.DataPilotFieldOrientation

HIDDEN	the field is not used in the table.
COLUMN	the field is used as a column field.
ROW	the field is used as a row field.

PAGE	the field is used as a page field.
DATA	the field is used as a data field.

DataPilotFieldReferenceItemType

Used to select the reference item

Constant (Long) : com.sun.star.sheet.DataPilotFieldReferenceItemType

NAMED	0 - the reference item is given by a name.
PREVIOUS	1 - the reference item is the previous one.
NEXT	2 - the reference item is the next one.

DataPilotFieldReferenceType

These constants select different types of References to calculate the data fields.

Constant (Long) : com.sun.star.sheet.DataPilotFieldReferenceType

NONE	0 - This type means, that the results in the data fields are displayed like they are.
ITEM_DIFFERENCE	1 - From each result, its reference value (see below) is subtracted, and the difference is shown. Totals outside of the base field are shown as empty results.
ITEM_PERCENTAGE	2 - Each result is divided by its reference value. The reference value is determined in the same way as for DataPilotFieldReferenceType::ITEM_DIFFERENCE . Totals outside of the base field are shown as empty results.
ITEM_PERCENTAGE_DIFFERENCE	3 - From each result, its reference value is subtracted, and the difference divided by the reference value. The reference value is determined in the same way as for DataPilotFieldReferenceType::ITEM_DIFFERENCE . Totals outside of the base field are shown as empty results.
RUNNING_TOTAL	4 - Each result is added to the sum of the results for preceding items in the base field, in the base field's sort order, and the total sum is shown.
ROW_PERCENTAGE	5 - Each result is divided by the total result for its row in the DataPilot table.
COLUMN_PERCENTAGE	6 - Same as DataPilotFieldReferenceType::ROW_PERCENTAGE , but the total for the result's column is used.
TOTAL_PERCENTAGE	7 - Same as DataPilotFieldReferenceType::ROW_PERCENTAGE , but the grand total for the result's data field is used.
INDEX	8 - The row and column totals and the grand total, following the same rules as above, are used to calculate the following expression.

DataPilotFieldShowItemsMode

These constants select different types of showing a selection of items.

Constant (Long) : com.sun.star.sheet.DataPilotFieldShowItemsMode

FROM_TOP	0 - The first items are shown.
FROM_BOTTOM	1 - The last items are shown.

DataPilotFieldSortMode

Describes the sort mode of the data field

Constant (Long) : com.sun.star.sheet.DataPilotFieldSortMode

NONE	0 - the data are taken as they come from the DataPilotSource.
MANUAL	1 - the user can sort the fields

NAME	2 - the field is sorted by its names
DATA	3 - the field ist sorted by the data in the given field

FillDateMode

Used to specify how an arithmetic date series is calculated.

Enumeration:com.sun.star.sheet.FillDateMode

FILL_DATE_DAY	for every new value a single day is added.
FILL_DATE_WEEKDAY	for evey new value a single day is added, but Saturdays and Sundays are skipped.
FILL_DATE_MONTH	for every new value one month is added (day keeps unchanged).
FILL_DATE_YEAR	for every new value one year is added (day and month keep unchanged).

FillDirection

Used to specify the direction of filling cells, for example, with a series.

Enumeration:com.sun.star.sheet.FillDirection

TO_BOTTOM	specifies that rows are filled from top to bottom.
TO_RIGHT	specifies that columns are filled from left to right.
TO_TOP	specifies that rows are filled from bottom to top.
TO_LEFT	specifies that columns are filled from right to left.

FillMode

Used to specify the series type used to fill cells.

Enumeration:com.sun.star.sheet.FillMode

SIMPLE	specifies a constant series.
LINEAR	specifies an arithmetic series.
GROWTH	specifies a geometric series.
DATE	specifies an arithmetic series for date values.
AUTO	specifies the use of a user-defined list.

FillStyle

Specifies how an area will be filled.

Enumeration: com.sun.star.drawing.FillStyle

NONE	the area is not filled.
SOLID	use a solid color to fill the area.
GRADIENT	use a gradient color to fill the area.
HATCH	use a hatch to fill the area.
BITMAP	use a bitmap to fill the area.

FilterConnection

Used to specify how two conditions in a filter descriptor are connected.

Enumeration : com.sun.star.sheet.FilterConnection

AND	both conditions have to be fulfilled.
OR	at least one of the conditions has to be fulfilled.

FilterOperator

Specifies the type of a single condition in a filter descriptor.

Enumeration:com.sun.star.sheet.FilterOperator

EMPTY	selects empty entries.
NOT_EMPTY	selects non-empty entries.
EQUAL	value has to be equal to the specified value.
NOT_EQUAL	value must not be equal to the specified value.
GREATER	value has to be greater than the specified value.
GREATER_EQUAL	value has to be greater than or equal to the specified value.
LESS	value has to be less than the specified value.
LESS_EQUAL	value has to be less than or equal to the specified value.
TOP_VALUES	selects a specified number of entries with the greatest values.
TOP_PERCENT	selects a specified percentage of entries with the greatest values.
BOTTOM_VALUES	selects a specified number of entries with the lowest values.
BOTTOM_PERCENT	selects a specified percentage of entries with the lowest values.

FontEmphasis

Determines the type and position of an emphasis mark in asian texts.

Constant (Short) : com.sun.star.text.FontEmphasis

NONE	0 - no emphasis mark is used.
DOT_ABOVE	1 - a dot is set above (or right from vertical text) the text.
CIRCLE_ABOVE	2 - a circle is set above (or right from vertical text) the text.
DISK_ABOVE	3 - a disc is set above (or right from vertical text) the text.
ACCENT_ABOVE	4 - an accent is set above (or right from vertical text) the text.
DOT_BELOW	5 - a dot is set below (or left from vertical text) the text.
CIRCLE_BELOW	6 - a circle is set below (or left from vertical text) the text.
DISK_BELOW	7 - a disk is set below (or left from vertical text) the text.
ACCENT_BELOW	8 - an accent is set below (or left from vertical text) the text.

FontFamily

These values are used to specify the general kind of font. They may be expanded in future versions.

Constant (Short):com.sun.star.awt.FontFamily

DONTKNOW	0 - specifies an unknown font family.
DECORATIVE	1 - specifies the family of decorative fonts.
MODERN	2 - specifies the family of modern fonts.
ROMAN	3 - specifies the family roman fonts (with serifes).
SCRIPT	4 - specifies the family of script fonts.
SWISS	5 - specifies the family roman fonts (without serifes).
SYSTEM	6 - specifies the family system fonts.

FontPitch

These values are used to specify whether the width of a character is fixed or variable. They may be expanded in future versions.

Constant (Short) : com.sun.star.awt.FontPitch.

DONTKNOW	0 - specifies that the pitch for this font is unknown.
FIXED	1 - specifies a font with a fixed character width.
VARIABLE	2 - specifies a font with a variable character width.

FontRelief

Constants that determine the relief type of a font.

Constant (Short) : com.sun.star.text.FontRelief

NONE	0 - no relief is used.
EMBOSSED	1 - the font relief is embossed.

ENGRAVED	2 - the font relief is engraved.

FontSlant

Used to specify the slant of a font.

Enumeration:com.sun.star.awt.FontSlant

NONE	specifies a font without slant.
OBLIQUE	specifies an oblique font (slant not designed into the font).
ITALIC	specifies an italic font (slant designed into the font).
DONTKNOW	specifies a font with an unknown slant.
REVERSE_OBLIQUE	specifies a reverse oblique font (slant not designed into the font).
REVERSE_ITALIC	specifies a reverse italic font (slant designed into the font).

FontStrikeOut

These values are used to specify the kind of strikeout. They may be expanded in future versions.

Constant (Short):com.sun.star.awt.FontStrikeout

NONE	0 - specifies not to strike out the characters.
SINGLE	1 - specifies to strike out the characters with a single line.
DOUBLE	2 - specifies to strike out the characters with a double line.
DONTKNOW	3 - The strikeout mode is not specified.
BOLD	4 - specifies to strike out the characters with a bold line.
SLASH	5 - specifies to strike out the characters with slashes.
X	6 - specifies to strike out the characters with X's.

FontType

These values are used to specify the technology of the font representation. They may be expanded in future versions.

Constant (Short) : com.sun.star.awt.FontType

DONTKNOW	0 - The type of the font is not known.
RASTER	1 - specifies a raster font.
DEVICE	2 - specifies a device font.
SCALABLE	4 - specifies a scalable font.

FontUnderline

These values are used to specify the kind of underlining. They may be expanded in future versions.

Constant (Short):com.sun.star.awt.FontUnderline

NONE	0 - specifies no underlining.
SINGLE	1 - specifies underlining with a single line.
DOUBLE	2 - specifies underlining with a double line.
DOTTED	3 - specifies underlining with a dotted line.
DONTKNOW	4 - The kind of underlining is not known.
DASH	5 - specifies underlining with a dashed line.
LONGDASH	6 - specifies underlining with long dashes.
DASHDOT	7 - specifies underlining with a dash and dot sequence.
DASHDOTDOT	8 - specifies underlining with a dash, dot, dot sequence.
SMALLWAVE	9 - specifies underlining with a small wave. ~
WAVE	10 - specifies underlining with a wave.
DOUBLEWAVE	11 - specifies underlining with a double wave.
BOLD	12 - specifies underlining with a bold line.
BOLDDOTTED	13 - specifies underlining with bold dots.
BOLDDASH	14 - specifies underlining with bold dashes.

BOLDLONGDASH	15 - specifies underlining with long bold dashes.
BOLDDASHDOT	16 - specifies underlining with a dash and dot sequence in bold.
BOLDDASHDOTDOT	17 - specifies underlining with a dash, dot, dot sequence in bold.
BOLDWAVE	18 - specifies underlining with a bold wave.

FontWeight

These values are used to specify whether a font is thin or bold. They may be expanded in future versions.

Constant (Float) : com.sun.star.awt.FontWeight

DONTKNOW	0 - The font weight is not specified/known.
THIN	50.0 - specifies a 50% font weight.
ULTRALIGHT	60.0 - specifies a 60% font weight.
LIGHT	75.0 - specifies a 75% font weight.
SEMILIGHT	90.0 - specifies a 90% font weight.
NORMAL	100.0 - specifies a normal font weight.
SEMIBOLD	110.0 - specifies a 110% font weight.
BOLD	150.0 - specifies a 150% font weight.
ULTRABOLD	175.0 - specifies a 175% font weight.
BLACK	200.0 - specifies a 200% font weight.

FormButtonType

Specifies the action to execute when a button is pressed.

Enumeration: com.sun.star.form.FormButtonType

PUSH	requires the button to act like a common push button, means no special action is triggered.
SUBMIT	When the button is clicked, it performs a submit on its containing form.
RESET	When the button is clicked, it performs a reset on its containing form.
URL	When the button is clicked, an URL set for the button is opened.

FormComponentType

These constants specify the class types used to identify a component.

Constant (Short): com.sun.star.form.FormComponentType

CONTROL	1 - This generic identifier is for controls which cannot be identified by an other specific identifier.
COMMANDBUTTON	2 - specifies a control that is used to begin, interrupt, or end a process.
RADIOBUTTON	3 - specifies a control that acts like a radio button. Grouped together, such radio buttons present a set of two or more mutually exclusive choices to the user.
IMAGEBUTTON	4 - specifies a control that displays an image that responds to mouse clicks.
CHECKBOX	5 - specifies a control that is used to check or uncheck to turn an option on or off.
LISTBOX	6 - specifies a control that displays a list from which the user can select one or more items.
COMBOBOX	7 - specifies a control that is used when a list box combined with a static text control or an edit control is needed.
GROUPBOX	8 - specifies a control that displays a frame around a group of controls with or without a caption.
TEXTFIELD	9 - specifies a control that is a text component that allows for the editing of a single line of text.
FIXEDTEXT	10 - specifies a control to display a fixed text, usually used to label other controls.
GRIDCONTROL	11 - is a table like control to display database data. Dialogs refer to this as a TableControl.

FILECONTROL	12 - specifies an control which can be used to enter text, extended by an (user-startable) file dialog to browse for files.
HIDDENCONTROL	13 - specifies a control that should not be visible.
IMAGECONTROL	14 - specifies a control to display an image.
DATEFIELD	15 - specifies a control to display and edit a date value.
TIMEFIELD	16 - specifies a control to display and edit a time value.
NUMERICFIELD	17 - specifies a field to display and edit a numeric value.
CURRENCYFIELD	18 - specifies a field to display and edit a currency value.
PATTERNFIELD	19 - specifies a control to display and edit a string according to a pattern.
SCROLLBAR	20 - specifies a control to display and edit, in the form of a scrollbar, a value from a continuous value range
SPINBUTTON	21 - specifies a control to edit, in the form of a spin field, a value from a continuous range of values
NAVIGATIONBAR	22 - specifies a control which provides controller functionality for the ::com::sun::star::form::component::DataForm it belongs to, such as functionality to navigate or filter this form.

FrameSearchFlag

These types describe the algorithm to be used to search a frame

Constant (Long) : com.sun.star.frame.FrameSearchFlag

PARENT	1 - allows search on the parent frames
SELF	2 - includes the start frame himself
CHILDREN	4 - includes all child frames of the start frame
CREATE	8 - frame will be created if not found
SIBLINGS	16 - includes the direct siblings of the start frame
TASKS	32 - allow the search outside the current sub task tree of the whole possible frame tree
ALL	64 - includes all frames except frames in other tasks sub trees but doesn't create any new frame
GLOBAL	128 - searches in the whole hierarchy of frames but doesn't create any new frame

GeneralFunction

Used to specify a function to be calculated from values.

Enumeration: com.sun.star.sheet.GeneralFunction

SUM	sum of all numerical values
COUNT	total number of all values (including non-numerical values)
COUNTNUMS	total number of all numerical values
AVERAGE	average of all numerical values
MAX	largest numerical value
MIN	smallest numerical value
PRODUCT	product of all numerical values
STDEV	standard deviation
VAR	variance
STDEVP	standard deviation based on the total population
VARP	variance based on the total population

GradientStyle

Specify the style of color dispersion.

Enumeration: com.sun.star.awt.GradientStyle

LINEAR	specifies a linear gradient.
AXIAL	specifies an axial gradient.
RADIAL	specifies a radial gradient.

ELLIPTICAL	specifies an elliptical gradient.
SQUARE	specifies a gradient in the shape of a square.
RECT	specifies a gradient in the shape of a rectangle.

GraphicType

Constants that describe the type of graphic

Constant (byte) : com.sun.star.graphic.GraphicType

EMPTY	0 - Graphic is empty
PIXEL	1 - Graphic is represented through single pixels
VECTOR	2 - Graphic is represented through vectors

HatchStyle

The HatchStyle defines the style of the lines in a hatch.

Enumeration: com.sun.star.drawing.HatchStyle

SINGLE	the hatch consists of a single horizontal line
DOUBLE	the hatch has a horizontal and a vertical line
TRIPLE	the hatch has a horizontal, a vertical and a diagonal line

ImageAlign

Specifies the alignment of an image.

Constant (Short): com.sun.star.awt.ImageAlign

LEFT	0 - specifies to align left.
TOP	1 - specifies to align top.
RIGHT	2 - specifies to align right.
BOTTOM	3 - specifies to align bottom.

ImagePosition

This is an unpublished constant and so it may not work....yet

Specifies the position of a image, relative to another object

Constant (Short) - com.sun.star.awt.ImagePosition

LeftTop	0 - to the left of, and top-aligned to, the other object
LeftCenter	1 - to the left of, and vertically centered to, the other object
LeftBottom	2 - to the left of, and bottom-aligned to, the other object
RightTop	3 - to the right of, and top-aligned to, the other object
RightCenter	4 - to the right of, and vertically centered to, the other object
RightBottom	5 - to the right of, and bottom-aligned to, the other object
AboveLeft	6 - above and left-aligned to the other object
AboveCenter	7 - above and horizontally centered to the other object
AboveRight	8 - above and right-aligned to the other object
BelowLeft	9 - below and left-aligned to the other object
BelowCenter	10 - below and horizontally centered to the other object
BelowRight	11 - below and right-aligned centered to the other object
Centered	12 - horizontally and vertically centered to the other object.

LineEndFormat

These values are used to specify which line end format should be used in strings

Constant (Short) : com.sun.star.awt.LineEndFormat

CARRIAGE_RETURN	0 - carriage return character (\r)

LINE_FEED	1 - line feed character (\n)
CARRIAGE_RETURN_LINE_FEED	2 - line feed then carriage return character (\n\r)

LineJoint

The LineJoint defines rendering of joints between thick lines

Enumeration: com.sun.star.drawing.LineJoint

NONE	the joint between lines will not be connected
MIDDLE	the middle value between the joints is used
BEVEL	the edges of the thick lines will be joined by lines
MITER	the lines join at intersections
ROUND	the lines join with an arc

LineStyle

Specifies the appearance of the lines of a shape.

Enumeration: com.sun.star.drawing.LineStyle

NONE	the line is hidden.
SOLID	the line is solid.
DASH	the line use dashes.

MeasureKind

Determines whether a measured shape is a standard measure or a radius measure.

Enumeration: com.sun.star.drawing.MeasureKind

STANDARD	use the length measurement.
RADIUS	use the radius measurement. This option cannot be used from the GUI Interface.

MeasureTextHorzPos

This enumeration defines the relative horizontal placement of the text inside a measure shape.

Enumeration: com.sun.star.drawing.MeasureTextHorzPos

AUTO
LEFTOUTSIDE
INSIDE
RIGHTOUTSIDE

MeasureTextVertPos

This enumeration defines the relative vertical placement of the text inside a measure shape.

Enumeration: com.sun.star.drawing.MeasureTextVertPos

AUTO
EAST
BREAKEDLINE
WEST
CENTERED

NamedRangeFlag

Used to specify the purpose of a named range.

Constant (Long) : com.sun.star.sheet.NamedRangeFlag

FILTER_CRITERIA	1 - The range contains filter criteria.
PRINT_AREA	2 - The range can be used as a print range.
COLUMN_HEADER	4 - The range can be used as column headers for printing.
ROW_HEADER	8 - The range can be used as row headers for printing.

NumberFormat

Contains constants that are used to specify the type of a number format.

Constant (Short) : com.sun.star.util.NumberFormat

ALL	0 - selects all number formats.
DEFINED	1 - selects only user-defined number formats.
DATE	2 - selects date formats.
TIME	4 - selects time formats.
CURRENCY	8 - selects currency formats.
NUMBER	16 - selects decimal number formats.
SCIENTIFIC	32 - selects scientific number formats.
FRACTION	64 - selects number formats for fractions.
PERCENT	128 - selects percentage number formats.
TEXT	256 - selects text number formats.
DATETIME	6 - selects number formats which contain date and time.
LOGICAL	1024 - selects boolean number formats.
UNDEFINED	2048 - is used as a return value if no format exists.

NumberFormatIndex

Pre defined number formats. I have purposely not included the constant values, if you read the API documentation you will understand why! Please note that these constant are for a deprecated draft module (i18n).

Constant (Short) : com.sun.star.i18n.NumberFormatIndex

NUMBER_START	Start of simple numerical formats (first format)
NUMBER_STANDARD	The "General" standard format
NUMBER_INT	0 Integer number
NUMBER_DEC2	0.00 Decimal number with 2 decimals
NUMBER_1000INT	#,##0 Integer number with group separator
NUMBER_1000DEC2	#,##0.00 Decimal number with group separator
NUMBER_SYSTEM	#,##0.00 In SO5/Win this format was retrieved from the Regional Settings
NUMBER_END	End of simple numerical formats (last format)
SCIENTIFIC_START	Start of Scientific formats (first format)
SCIENTIFIC_000E000	0.00E+000 Number in scientific notation with exponent in 3 digit placeholders
SCIENTIFIC_000E00	0.00E+00 Number in scientific notation with exponent in 2 digit placeholders
SCIENTIFIC_END	End of Scientific formats (last format)
PERCENT_START	Start of Percent formats (first format)
PERCENT_INT	0% Percentage format, rounded to integer
PERCENT_DEC2	0.00% Percentage format, rounded to 2 decimals
PERCENT_END	End of Percent formats (last format)
FRACTION_START	Start of Fraction formats (first format)
FRACTION_1	# ?/? Number with decimal in fraction in 1 digit placeholder
FRACTION_2	# ??/?? Number with decimal in fraction in 2 digit placeholders
FRACTION_END	End of Fraction formats (last format)
CURRENCY_START	Start of Currency formats (first format)
CURRENCY_1000INT	#,##0 DM Integer currency format with group separator
CURRENCY_1000DEC2	#,##0.00 DM Decimal currency format with group separator
CURRENCY_1000INT_RED	#,##0 DM Integer currency format with negative in red
CURRENCY_1000DEC2_RED	#,##0.00 DM Decimal currency format with negative in red
CURRENCY_1000DEC2_CCC	#,##0.00 DEM Currency in ISO-4217 abbreviation format

CURRENCY_1000DEC2_DASHED #,##0.-- DM Currency format with dash representing 0 in decimals
CURRENCY_END End of Currency formats (last format)
DATE_START Start of Date formats (first format)

DATE_SYSTEM_SHORT 08.10.97
DATE_SYSTEM_LONG Wednesday, 8. October 1997
DATE_SYS_DDMMYY 08.10.97
DATE_SYS_DDMMYYYY 08.10.1997

Note: When editing already existing date data this format is forced in order to always edit the full century.

DATE_SYS_DMMMYY 8. Oct 97
DATE_SYS_DMMMYYYY 8. Oct 1997
DATE_DIN_DMMMYYYY 8. Oct. 1997 DIN/EN
DATE_SYS_DMMMMYYYY 8. October 1997
DATE_DIN_DMMMMYYYY 8. October 1997 DIN/EN
DATE_SYS_NNDMMMYY Wed, 8. Oct 97
DATE_DEF_NNDDMMMYY Wed 08.Oct 97
DATE_SYS_NNDMMMMYYYY Wed, 8. October 1997
DATE_SYS_NNNNDMMMMYYYY Wednesday, 8. October 1997
DATE_DIN_MMDD 10-08 DIN/EN
DATE_DIN_YYMMDD 97-10-08 DIN/EN/ISO
DATE_DIN_YYYYMMDD 1997-10-08 DIN/EN/ISO
DATE_SYS_MMYY 10.97
DATE_SYS_DDMMM 08.Oct
DATE_MMMM October
DATE_QQJJ 4th quarter 97
DATE_WW week of year
DATE_END End of Date formats (last format)
TIME_START Start of Time formats (first format)
TIME_HHMM HH:MM Time format with hour and minute
TIME_HHMMSS HH:MM:SS Time format with hour, minute and second
TIME_HHMMAMPM HH:MM AM/PM Time format with hour, minute and morning/afternoon notation
TIME_HHMMSSAMPM HH:MM:SS AM/PM time format with hour, minute, second and morning/afternoon notation
TIME_HH_MMSS [HH]:MM:SS Time format with amount of hours
TIME_MMSS00 MM:SS,00 Time format with second in fraction
TIME_HH_MMSS00 [HH]:MM:SS,00 Time format with amount of hours and seconds with fraction
TIME_END End of Time formats (last format)
DATETIME_START Start of DateTime formats (first format)
DATETIME_SYSTEM_SHORT_HHMM 08.10.97 01:23 Date/time format
DATETIME_SYS_DDMMYYYY_HHMMSS 08.10.1997 01:23:45 Date/time format with second

Note: When editing already existing date/time data this format is forced in order to always edit the full century.

DATETIME_END End of DateTime formats (last format)
BOOLEAN BOOLEAN format @ATTENTION Not defined in locale data, but generated by the number formatter. If you want to access this format you MUST do it via ::com::sun::star::util::XNumberFormatTypes::getFormatIndex() instead of XNumberFormatCode::getFormatCode() .
TEXT Text format @ATTENTION Not defined in locale data, but generated by the number formatter. If you want to access this format you MUST do it via ::com::sun::star::util::XNumberFormatTypes::getFormatIndex() instead of XNumberFormatCode::getFormatCode()
INDEX_TABLE_ENTRIES count of built-in format codes.

PaperFormat

Specifies the format (size) of the paper on a text document.

Enumeration : com.sun.star.view.PaperFormat

A3	Specifies the paper format as A3.
A4	Specifies the paper format as A4.
A5	Specifies the paper format as A5.
B4	Specifies the paper format as B4.
B5	Specifies the paper format as B5.
LETTER	Specifies the paper format as Letter.
LEGAL	Specifies the paper format as Legal.
TABLOID	Specifies the paper format as Tabloid.
USER	The real paper size is user defined in 100th mm.

ParagraphAdjust

These enumeration values describe the formatting of a text paragraph.

Enumeration : com.sun.star.style.ParagraphAdjust

LEFT	adjusted to the left border
RIGHT	adjusted to the right border
BLOCK	adjusted to both borders / stretched, except for last line
CENTER	adjusted to the center
STRETCH	adjusted to both borders / stretched, including last line

PolygonFlags

Defines how a bezier curve goes through a point.

Enumeration: com.sun.star.drawing.PolygonFlags

NORMAL	the point is normal, from the curve discussion view.
SMOOTH	the point is smooth, the first derivation from the curve discussion view.
CONTROL	the point is a control point, to control the curve from the user interface.
SYMMETRIC	the point is symmetric, the second derivation from the curve discussion view.

PolygonKind

This enumeration defines the type of polygon.

Enumeration: com.sun.star.drawing.PolygonKind

LINE	This is the PolygonKind for a LineShape.
POLY	This is the PolygonKind for a PolyPolygonShape.
PLIN	This is the PolygonKind for a PolyLineShape.
PATHLINE	This is the PolygonKind for an OpenBezierShape.
PATHFILL	This is the PolygonKind for a ClosedBezierShape.
FREELINE	This is the PolygonKind for an OpenFreeHandShape.
FREEFILL	This is the PolygonKind for a ClosedFreeHandShape.
PATHPOLY	This is the PolygonKind for a PolyPolygonPathShape.
PATHPLIN	This is the PolygonKind for a PolyLinePathShape.

PropertyState

This enumeration lists the states that a property value can have.

The state consists of two aspects:

whether a value is available or void,
whether the value is stored in the property set itself or is a default, or ambiguous.

Enumeration : com.sun.star.beans.PropertyState

DIRECT_VALUE	The value of the property is stored in the PropertySet itself.
DEFAULT_VALUE	The value of the property is available from a master (e.g., template).
AMBIGUOUS_VALUE	The value of the property is only a recommendation because there are multiple values for this property (e.g., from a multi selection).

RectanglePoint

Specifies one of nine points in a rectangle.

Enumeration : com.sun.star.drawing.RectanglePoint

LEFT_TOP	specify to the point on the left side from the top of the rectangle.
MIDDLE_TOP	specify to the point on the middle of the top from the rectangle.
RIGHT_TOP	specify to the point on the right side from the top of the rectangle.
LEFT_MIDDLE	specify to the point on the middle of the left side from the rectangle.
MIDDLE_MIDDLE	specify to the point on the center from the rectangle.
RIGHT_MIDDLE	specify to the point on the middle of the right side from the rectangle.
LEFT_BOTTOM	specify to the point on the bottom of the left side from the rectangle.
MIDDLE_BOTTOM	specify to the point on the middle of the bottom from the rectangle.
RIGHT_BOTTOM	specify to the point on the bottom of the right side from the rectangle.

RubyAdjust

These enumeration values describe the adjustment of ruby text.

Enumeration: com.sun.star.text.RubyAdjust

LEFT	adjusted to the left.
CENTER	centric adjusted.
RIGHT	adjusted to the right.
BLOCK	adjusted to both borders / stretched
INDENT_BLOCK	adjusted to both borders except for a small indent on both sides

ShadowLocation

Specifies the location of the shadow in a ShadowFormat.

Enumeration: com.sun.star.sheet.ShadowLocation

NONE	no shadow.
TOP_LEFT	shadow is located along the upper and left sides.
TOP_RIGHT	shadow is located along the upper and right sides.
BOTTOM_LEFT	shadow is located along the lower and left sides.
BOTTOM_RIGHT	shadow is located along the lower and right sides.

TableOrientation

Used to select whether operations are carried out on columns or rows.

Enumeration: com.sun.star.table.TableOrientation

COLUMNS	operations are carried out on columns.
ROWS	operations are carried out on rows.

TableSortFieldType

Used to specify the type of contents in a sort field (row/column) of a table.

Enumeration: com.sun.star.table.TableSortFieldType

AUTOMATIC	type is determined automatically.
NUMERIC	sort field contains numerical data.
ALPHANUMERIC	sort field contains text data.

TableValidationVisibility

These constants specify whether and how a list of possible values of a cell should be shown.

Constant (Short): com.sun.star.sheet.TableValidationVisibility

INVISIBLE	0 - The List is not shown.
UNSORTED	1 - The List is shown unsorted.
SORTEDASCENDING	2 - The List is shown sorted ascending.

ValidationAlertStyle

used to specify how invalid cell contents are treated.

Enumeration: com.sun.star.sheet.ValidationAlertStyle

STOP	error message is shown and the change is rejected.
WARNING	warning message is shown and the user is asked whether the change will be accepted (defaulted to "No").
INFO	information message is shown and the user is asked whether the change will be accepted (defaulted to "Yes").
MACRO	macro is executed.

ValidationType

Used to specify which cell contents are treated as valid.

Enumeration : com.sun.star.sheet.ValidationType

ANY	any cell content is valid; no conditions are used.
WHOLE	any whole number matching the specified condition is valid.
DECIMAL	any number matching the specified condition is valid.
DATE	any date value matching the specified condition is valid.
TIME	any time value matching the specified condition is valid.
TEXT_LEN	string is valid if its length matches the specified condition.
LIST	Only strings from a specified list are valid.
CUSTOM	The specified formula detemines which contents are valid.

VerticalAlignment

Specifies the horizontal alignment of an object within a container object.

Enumeration: Com.sun.star.style VerticalAlignment

TOP	set the vertical alignment to the center between the top and bottom margins from the container object.
MIDDLE	set the vertical alignment to the top margin from the container object.
BOTTOM	set the vertical alignment to the bottom margin from the container object.

WritingMode2

This set of constants describes different writing directions, and * also allows for a value 'PAGE' which causes the writing mode to be * obtained from the current page.

Constant (Short) - com.sun.star.text.WritingMode2

LR_TB	0 - characters are written left-to-right. Lines and blocks, top-to-bottom.
RL_TB	1 - characters are written right-to-left. Lines and blocks, top-to-bottom.
TB_RL	2 - characters are written top-to-bottom. Lines and blocks, right-to-left.

TB_LR	3 - characters are written top-to-bottom. Lines and blocks , left-ro-right.
PAGE	4 - obtain writing mode from the current page.

Useful Procedures

```
'=============================================================

Sub ObjectHelp(oHelp)

' Displays internal information about an object

Msgbox "Object type " & oHelp.getImplementationName()
Msgbox oHelp.dbg_methods
Msgbox oHelp.dbg_properties
Msgbox oHelp.dbg_supportedinterfaces
End Sub

'=============================================================

Function ActiveSheet

' mimics the Excel object of the same name
' NB if you include this function you will not be able to ue the
' method of the same name although functionally equivelent

ActiveSheet = ThisComponent.CurrentController.ActiveSheet
End Function

'=============================================================

Function ActiveWorkbook

' mimics the Excel object of the same name

ActiveWorkbook = ThisComponent
End Function

'=============================================================

Function ActiveWindow

' mimics the Excel object of the same name

ActiveWindow = ThisComponent.CurrentController
End Function

'=============================================================

Function ActiveCell

' mimics the Excel object of the same name

' This was lifted from an example. I have no idea why it works
' I did try and work it out but gave up... As far as I could see
' this should return the whole sheet as a selected range

Dim oRanges as Object

oRanges = ThisComponent.createInstance("com.sun.star.sheet.SheetCellRanges")
ThisComponent.CurrentController.Select(oRanges)
ActiveCell = ThisComponent.CurrentSelection
End Function

'=============================================================

Function ActiveChart

' mimics the Excel object of the same name

' If a chart is not selected then a null is returned
' a Null is easier to handle than a Nothing !

Dim oSelected as Object

oSelected = ThisComponent.CurrentSelection
If NOT oSelected.SupportsService("com.sun.star.drawing.ShapeCollection") Then
  ActiveChart = Null
```

```
Else
  ActiveChart = GetChart(oSelected)
End If
End Function
'=============================================================

Function GetChart(oShapeCollection)
'
' Returns the chart object for the corresponding shape collection
' The shape collection must be a shape type of OLE2Shape
'
' requires the function ActiveSheet
'
GetChart = ActiveSheet.Charts.getbyname(oShapeCollection.getbyindex(0).Persistname)
End Function
```

construction of a chart cross reference

Charts

AreaDiagram

ChartStatistics , ChartAxisXSupplier , ChartTwoAxisYSupplier , ChartAxisZSupplier , Dim3dDiagram , StackableDiagram

BarDiagram

ChartStatistics , ChartAxisXSupplier , ChartTwoAxisYSupplier , ChartAxisZSupplier , Dim3dDiagram , StackableDiagram

--Dim3DDiagram

DonutDiagram

LineDiagram

ChartStatistics , ChartAxisXSupplier , ChartTwoAxisYSupplier , ChartAxisZSupplier , Dim3dDiagram , StackableDiagram

NetDiagram

ChartAxisYSupplier , StackableDiagram

PieDiagram

Dim3dDiagram

--StackableDiagram

StockDiagram

ChartStatistics , ChartAxisXSupplier , ChartTwoAxisYSupplier

XYDiagram

ChartStatistics , ChartAxisXSupplier , ChartTwoAxisYSupplier , ChartAxisZSupplier , Dim3dDiagram , StackableDiagram