

## JClass Table Pro Exposed Properties

Once JClass Table Pro has been installed for use inside Visual Cafe, it can be used like any other component. There are close to 60 table properties exposed to Visual Cafe, not including "inherited" properties. The table below shows all of the JClass Table Pro properties:

<b>Property</b>	<b>Visual Cafe Name</b>	<b>Description</b>
<u>AlignmentSeries</u>	Cell Alignments	Alignment for individual cells.
<u>AllowCellResize</u>	Allow Cell Resize	Determines if the user can resize cells.
<u>BackgroundSeries</u>	Cell Background Colors	Background color for individual cells.
<u>BorderSidesSeries</u>	Cell Border Sides	Visible border sides for individual cells.
<u>BorderTypeSeries</u>	Cell Border Types	Border types for individual cells.
<u>CellValues</u>	Cell Values	Contents for individual cells.
<u>CharHeightSeries</u>	Cell Height in Characters	Height in characters of individual cells.
<u>CharWidthSeries</u>	Cell Width in Characters	Width in characters of individual cells.
<u>ColumnLabelOffset</u>	Column Label Offset	Offset between column labels and table.
<u>ColumnLabelPlacement</u>	Column Label Placement	Location of the column labels.
<u>ColumnLabelSort</u>	Column Label Sort	Determines if the user can sort by column.
<u>ColumnLabels</u>	Column Labels	Column label values.
<u>ComponentSeries</u>	Components in Cells	AWT components in individual cells.
<u>DataTypeSeries</u>	Cell Data Types	Data type for individual cells.
<u>DisplayClipArrows</u>	Display Clip Arrows	Determines whether text clip arrows are shown.
<u>DoubleBuffer</u>	Double Buffer	Determines whether double buffering is used.
<u>EditableSeries</u>	Editable Cells	Editable attribute for individual cells.
<u>FontSeries</u>	Cell Fonts	Display font for individual cells.
<u>ForegroundSeries</u>	Cell Foreground Colors	Foreground color for individual cells.
<u>FrameBorderType</u>	Frame Border Type	Border type for the frame around the table.
<u>FrameShadowThickness</u>	Frame Shadow Thickness	Thickness of the frame around the table.
<u>FrozenColumnPlacement</u>	Frozen Column Placement	Location of frozen columns.
<u>FrozenColumns</u>	Frozen Column	Number of frozen columns.
<u>FrozenRowPlacement</u>	Frozen Row Placement	Location of frozen rows.

<u>FrozenRows</u>	Frozen Rows	Number of frozen rows.
<u>JumpScroll</u>	Jump Scroll	Determines whether jump scrolling is in use.
<u>LeftColumn</u>	Left Column	Left-most column in table.
<u>MarginHeight</u>	Margin Height	Height of the margin around each cell.
<u>MarginWidth</u>	Margin Width	Width of the margin around each cell.
<u>MaxLengthSeries</u>	Cell Max Lengths	Maximum content length for individual cells.
<u>MultiLineSeries</u>	Multiline Cells	Allow multi lines attribute for individual cells.
<u>HorizSBAttachment</u>	Horizontal Scrollbar Attachment	Attach point for horizontal scrollbar.
<u>HorizSBPosition</u>	Horizontal Scrollbar Position	Position of horizontal scrollbar.
<u>HorizSBOffset</u>	Horizontal Scrollbar Offset	Offset between table and horizontal scrollbar.
<u>HorizSBDisplay</u>	Horizontal Scrollbar Display	Determines whether to display horizontal scrollbar.
<u>MinCellVisibility</u>	Minimum Cell Visibility	Minimum visible percentage of a cell.
<u>Mode</u>	Mode	Determines table operation - table or list.
<u>NumColumns</u>	Number of Columns	Number of columns.
<u>NumRows</u>	Number of Rows	Number of rows.
<u>PixelHeightSeries</u>	Cell Height in Pixels	Height in pixels of individual cells.
<u>PixelWidthSeries</u>	Cell Width in Pixels	Width in pixels of individual cells.
<u>RepeatBGColors</u>	Repeat Background Colors	Repeat pattern for background colors.
<u>RepeatFGColors</u>	Repeat Foreground Colors	Repeat pattern for foreground colors.
<u>RowLabelOffset</u>	Row Label Offset	Offset between row labels and table.
<u>RowLabelPlacement</u>	Row Label Placement	Location of the row labels.
<u>RowLabels</u>	Row Labels	Row label values.
<u>SelectedBackground</u>	Selected Background Color	Background color for selected cells.
<u>SelectedForeground</u>	Selected Foreground Color	Foreground color for selected cells.
<u>SelectionPolicy</u>	Selection Policy	Type of allowable selection.
<u>ShadowThickness</u>	Shadow Thickness	Thickness of shadow around each cell.
<u>SpanningCells</u>	Spanning Cells	Span dimensions for individual cells.
<u>StringCaseSeries</u>	Cell String Cases	String case (upper/lower/as-is) for cells.
<u>TopRow</u>	Top Row	Topmost row in the table.

<u>TrackCursor</u>	Track Cursor	Determines whether the cursor changes.
<u>TraversableSeries</u>	Traversable Cells	Allows traversal of individual cells.
<u>ValidatePolicy</u>	Validation Policy	Validation policy for table.
<u>VertSBAttachment</u>	Vertical Scrollbar Attachment	Attach point for vertical scrollbar.
<u>VertSBPosition</u>	Vertical Scrollbar Position	Position of vertical scrollbar.
<u>VertSBOffset</u>	Vertical Scrollbar Offset	Offset between table and vertical scrollbar.
<u>VertSBDisplay</u>	Vertical Scrollbar Display	Determines whether to display vertical scrollbar.

## ***AlignmentSeries***

Specifies the alignment of elements in cells. The value of this property is a list of cell/value combinations. The following values are defined, each of which represents a possible location of the element in its cell: `topleft`, `left`, `topcenter`, `center`, `topright`, `right`, `middleleft`, `middlecenter`, `middleright`, `bottomleft`, `bottomcenter`, `bottomright`. The default is `topleft`.

### **Example code:**

```
(LABEL ALL center) (ALL LABEL left)
```

## ***AllowCellResize***

Specifies how an end-user is able to interactively resize rows/columns. Valid values are `resize_none`, `resize_column`, `resize_row` and `resize_all`. The default is `resize_all`. Visual Cafe supplies a choice box for editing this property.

## ***BackgroundSeries***

Specifies the background color of cells and labels. The value of this property is a list of cell/value combinations; Each value must be a valid color name or RGB value, or one of REPEAT\_ROW or REPEAT\_COLUMN. REPEAT\_ROW specifies that the cell/label colors are to be taken from the list specified by the RepeatBGColors property, and are to be repeated in row order. REPEAT\_COLUMN specifies that the cell/label colors are to be specified by RepeatBGColors and repeated in column order.

### **Example code:**

(ALL ALL lightGrey) (LABEL ALL white)

## ***BorderSidesSeries***

Specifies the sides of a cell or label that display the border type in their border area. The value of this property is a list of cell/value combinations. Valid values are any or all of `borderside_none`, `borderside_left`, `borderside_top`, `borderside_bottom`, `borderside_right`, `borderside_all`. The default value is `borderside_all`. If more than one value is specified, the values are separated by "+" characters.

### **Example code:**

```
(all all borderside_top+borderside_left)
```

## ***BorderTypeSeries***

Specifies the style used for cell and label borders. The value of this property is a list of cell/value combinations; Valid values are `border_in`, `border_out`, `border_plain`, `border_none`, `border_etched_in`, `border_etched_out`, `border_frame_in`, `border_frame_out`. The default is `border_in`.

### **Example code:**

(ALL ALL `border_etched_in`)

## **CellValues**

Specifies the contents of cells. Round brackets ("()") are used to delimit rows, and individual cell values are separated by pipe characters ("|"). If any of the following characters is to appear, it must be preceded by a backslash character ("\"): | \ [ ] ( ). Newlines are represented by "\n". JCStrings can also be used within column labels.

### **Example code:**

```
(James Q. Doohan|1701 Planetia Blvd.\nAnytown, U.S.A.|\n|President|$245,000)\n(John F. Kricfalusi|983 Nickelodeon Street\nAnytown,\nU.S.A.|\(999\) 555-9876|Animator|$1,000)
```

## ***CharHeightSeries***

Specifies the number of lines of text a row can display. The value of this property is a list of row/value combinations; each combination is enclosed in parentheses, and consists of a row and its value. The value can be any number of lines. This property is ignored if `PixelHeightSeries` is set for this row to anything other than `NOVALUE`.

### **Example code:**

(3 5)

## ***CharWidthSeries***

Specifies the number of characters that a column can display. The value of this property is a list of column/value combinations; each combination is enclosed in parentheses, and consists of a column and its value. The value can be any number of characters. This property is ignored if `PixelWidthSeries` is set for this column to anything other than `NOVALUE`.

### **Example code:**

```
(2 15)
```

***ColumnLabelOffset***

Specifies the spacing between the column labels and the cell area in pixels. If a negative value is specified, the labels overlap the cell area. Visual Cafe supplies an integer editor for entering the column label offset.

### ***ColumnLabelPlacement***

Specifies the position of the column labels relative to the cells. Valid values are `place_top` and `place_bottom`. Visual Cafe supplies a choice box for editing this property.

### ***ColumnLabelSort***

Specifies whether column sorting is to be performed whenever the user clicks on the column label. Valid values are `true` (enable sorting) and `false` (column sorting not enabled); the default is `false`. Visual Cafe provides a choice box for choosing between `true` and `false`.

## ***ColumnLabels***

Specifies the column labels. In VisualCafe, column labels are entered using the standard `String[]` editor. Labels must be entered in order. A blank label (" ") can be used to "skip" a label.

## ***ComponentSeries***

Specifies a component to be used in a cell or series of cells. The value is a list of cell/component combinations. Any fully-named component should work.

### **Example code:**

(1 2 Button)

## ***DataTypeSeries***

Specifies a list of data types. The value of this property is a list of cell/value combinations; Valid values are `type_double`, `type_float`, `type_integer` and `type_string` (default). In the JClass LiveTable Applet, this property is primarily used by `columnLabelSort` and `rowLabelSort` to determine the column or row sorting order. Values of `type_string` are sorted in alphabetical order; others are sorted in numerical order.

### **Example code:**

```
(ALL 1 type_integer)
```

## ***DisplayClipArrows***

Specifies whether to display clipping arrows when a cell/label value cannot be fully displayed in its cell/label. Clipping arrows appear at the bottom-right corner of a cell/label. Valid values are `true` (display clip arrows) and `false` (do not display); the default value is `true`. Visual Cafe supplies a choice box to choose between `true` and `false`.

## ***DoubleBuffer***

Specifies whether double buffering is enabled for the table. Double buffering will make table scrolling smoother but slightly slower. Valid values are `true` and `false`. Visual Cafe provides a choice box for choosing between `true` and `false`.

## ***EditableSeries***

Specifies a list of editable attributes. The value of this property is a list of cell/boolean combinations. This property is used to determine if a cell or series of cells can be edited.

### **Example code:**

```
(1 1 false)(1 2 true)
```

## ***FontSeries***

Specifies the font list for cells and/or labels. The value of this property is a list of cell/value combinations.

### **Example code:**

```
(ALL ALL courier-plain-12) (LABEL ALL helvetica-italic-12)
```

## ***ForegroundSeries***

Specifies the foreground color of cells and labels. The value of this property is a list of cell/value combinations. Each value must be a valid color name or RGB value, or one of `REPEAT_ROW` or `REPEAT_COLUMN`. `REPEAT_ROW` specifies that the cell/label colors are to be taken from the list specified by the `RepeatFGColors` property, and are to be repeated in row order. `REPEAT_COLUMN` specifies that the cell/label colors are to be specified by `RepeatFGColors` and repeated in column order.

### **Example code:**

```
(0 4 red)
```

## ***FrameBorderType***

Specifies the type of border for the frame surrounding the cell and label areas. Valid values are `border_none`, `border_etched_in`, `border_etched_out`, `border_in`, `border_out` and `border_plain`. The default value is `border_none`. The frame border only appears if the `FrameShadowThickness` property is set. Visual Cafe provides a text field for entering the integer value.

### ***FrameShadowThickness***

Specifies the thickness of the frame shadows surrounding the cell and label areas. The thickness is measured in pixels. The default value is 0 (no frame shadows). Visual Cafe provides a text field for entering the integer value.

### ***FrozenColumnPlacement***

Specifies the location of the frozen columns relative to the scrolling cells. Changing the position of the frozen columns does not change the location of the columns in the table structure. Valid values are `place_left` (default) and `place_right`. Visual Cafe provides a choice box for choosing between the valid values.

## ***FrozenColumns***

Specifies the number of columns from the start of the table that are not horizontally scrollable. The default is 0 (no frozen columns). The position of the frozen columns on the screen is specified by `FrozenColumnPlacement`. Visual Cafe provides a text field for entering the integer value.

## ***FrozenRowPlacement***

Specifies the location of the frozen rows relative to the scrolling cells. Changing the position of the frozen rows does not change the location of the rows in the table structure. Valid values are `place_top` (default) and `place_bottom`. Visual Cafe provides a choice box for choosing between the valid values.

## ***FrozenRows***

Specifies the number of rows from the start of the table that are not vertically scrollable. The default is 0 (no frozen rows). The position of the frozen rows on the screen is specified by `frozenRowPlacement`. Visual Cafe provides a text field for entering the integer value.

## ***JumpScroll***

Specifies the scrolling behavior of each scrollbar. Scrollbars can either scroll smoothly or "jump" scroll in whole row/column increments. Valid values are `jump_none` (default), `jump_horizontal`, `jump_vertical` and `jump_all`. Visual Cafe provides a choice box for choosing between the valid values.

***LeftColumn***

Specifies the non-frozen column initially visible at the left side of the table display. Visual Cafe provides a text field for entering the integer value.

***MarginHeight***

Specifies the margin (in pixels) between the inside edge of the cell border and the top and bottom edge of the cell's contents. This property affects the entire table. Visual Cafe provides a text field for entering the integer value.

***MarginWidth***

Specifies the margin (in pixels) between the inside edge of the cell border and the left and right edge of the cell's contents. This property affects the entire table. Visual Cafe provides a text field for entering the integer value.

## ***MaxLengthSeries***

Specifies the maximum length of cell contents as entered by the user. The value of this property is a list of cell/integer combinations.

### **Example code:**

```
(0 all 10)(4 4 33)
```

## ***MultiLineSeries***

Specifies the multi-line property for each cell. The value of this property is a list of cell/boolean combinations.

### **Example code:**

```
(0 0 true)(1 all false)
```

### ***HorizSBAttachment***

Specifies how the end of the horizontal scrollbar is attached to the table. Valid values are `attach_cells` (scrollbar ends at the edge of the visible cells) and `attach_side` (scrollbar ends at the edge of the table). Visual Cafe provides a choice box for choosing between the valid values.

### ***HorizSBPosition***

Specifies how the horizontal scrollbar is attached to the table. Valid values are `sposition_cells` (scrollbar is attached to the cell/label viewport) and `sposition_side` (scrollbar is attached to the side of the table). Visual Cafe provides a choice box for choosing between the valid values.

***HorizSBOffset***

Specifies the distance between the horizontal scrollbar and the table. The default value is 6. Visual Cafe provides a text field for entering the integer value.

## ***HorizSBDisplay***

Specifies when to display the horizontal scrollbar. Valid values are `sbdisplay_always` (always displayed), `sbdisplay_never` (never displayed) and `sbdisplay_as_needed` (displayed when the rows or columns are not visible). Visual Cafe provides a choice box for choosing between the valid values.

### ***MinCellVisibility***

Specifies the minimum amount of a cell that is made visible when edited. When the table scrolls to edit a non-visible cell, this property determines the percentage of the cell scrolled into view. When set to 100, the cell is made completely visible. When set to 10, only 10% of the cell is made visible. If set to 0, the cell will not scroll to be made visible. Visual Cafe provides a text field for entering the integer value.

## **Mode**

Specifies the mode of operation of the table. When this property is set to `MODE_LIST`, selecting a cell causes all cells in the row to be selected. In addition, the following properties are set:

- `HorizSBAttachment` and `VertSBAttachment` are set to `ATTACH_SIDE`
- `HorizSBPosition` and `VertSBPosition` are set to `SBPOSITION_SIDE`
- `SelectionPolicy` is set to `SELECT_SINGLE`
- `ShadowThickness` is set to 0
- Cell traversal is disallowed.

(The `SelectionPolicy` value controls whether users are allowed to select single rows, a range of rows, or multiple ranges of rows).

Visual Cafe provides a choice box for choosing between the valid values.

## ***NumColumns***

Specifies the current number of columns in the table (default: 5). This value does not affect the table's internal `CellValues` vector. `JCTable.addColumn()` and `JCTable.deleteColumns()` update this property by the number of columns added/deleted. If set to zero, no cells are displayed. Visual Cafe provides a text field for entering the integer value.

## ***NumRows***

Sets the current number of rows in the table (default: 10). This value does not affect the table's internal `CellValues` vector. `JCTable.addRow()` and `JCTable.deleteRow()` update this property by the number of rows added/deleted. If set to zero, no cells are displayed. Visual Cafe provides a text field for entering the integer value.

## ***PixelHeightSeries***

Specifies the row height in pixels. The value of this property is a list of row/value combinations; each combination is enclosed in parentheses, and consists of a row and its value. The value can be any number of pixels, or can be `VARIABLE`, which sets a row to the necessary height to display all of the text (or other elements) for that row. This property overrides the value of `CharHeightSeries`.

### **Example code:**

(all variable)

## ***PixelWidthSeries***

Specifies the column width in pixels. The value of this property is a list of column/value combinations; each combination is enclosed in parentheses, and consists of a column and its value. The value can be any number of pixels, or can be `VARIABLE`, which sets a column to the necessary width to display all of the text (or other elements) for that column. This property overrides the value of `CharWidthSeries`.

### **Example code:**

(all 50)

## ***RepeatBGColors***

Specifies the list of colors to be used when a `BackgroundSeries` value is set to `REPEAT_ROW` or `REPEAT_COLUMN`. The value of this property is a list of colors separated by commas. Each color in the list must be a valid color name or RGB value.

### **Example code:**

```
yellow, blue
```

## ***RepeatFGColors***

Specifies the list of colors to be used when a `ForegroundSeries` value is set to `REPEAT_ROW` or `REPEAT_COLUMN`. The value of this property is a list of colors separated by commas. Each color in the list must be a valid color name or RGB value.

### **Example code:**

```
blue, yellow
```

***RowLabelOffset***

Specifies the spacing between the row labels and the cell area in pixels. If a negative value is specified, the labels overlap the cell area. Visual Cafe provides a text field for entering the integer value.

### ***RowLabelPlacement***

Specifies the position of the row labels relative to the cells. Valid values are `place_left` (default) and `place_right`. Visual Cafe provides a choice box for choosing between the valid values.

## ***RowLabels***

Specifies the row labels. In Visual Cafe, row labels are entered using the standard `String[]` editor. Labels must be entered in order. A blank label ("" ) can be used to "skip" a label.

### ***SelectedBackground***

Specifies the background color of selected cells (default: cell's foreground color). When set to null, selected cells look identical to unselected cells - the background and foreground colors are the same as the colors defined for the cells (`Foreground` and `Background`). Visual Cafe provides a color chooser for selecting the desired color.

## ***SelectedForeground***

Specifies the foreground color of selected cells (default: cell's background color). If set to null when `SelectedBackground` is not null, the foreground of selected cells is the same as the foreground colors defined for the cells (`Foreground`). This value is ignored when `SelectedBackground` is set to null. Visual Cafe provides a color chooser for selecting the desired color.

## ***SelectionPolicy***

Sets the selection behavior allowed on the table, both by end-users, and by the application. Valid values are:

<code>SELECT_NONE</code>	No selection is allowed
<code>SELECT_SINGLE</code>	Selection of only one cell allowed
<code>SELECT_RANGE</code>	Selection of only one range of cells allowed
<code>SELECT_MULTIRANGE</code>	Selection of multiple ranges of cells allowed

When `SelectionPolicy` is set to `SELECT_NONE`, `SELECT` callbacks are not called as a user edits or attempts to select cells. Setting this property does not change the selected cell list.

Visual Cafe provides a choice box for choosing between the valid values.

***ShadowThickness***

Specifies the width (in pixels) of the borders surrounding each cell and label. The default value is 1 pixel. Visual Cafe provides a text field for entering the integer value.

## ***SpanningCells***

Specifies a list of spanned ranges of cells or labels. A spanned range is a range of cells or labels that appear joined and can be treated as one cell/label. The value of this property is a list of ranges; each range is enclosed in parentheses. Each range is specified using one of the following formats:

r1-r2 c1-c2    The row and column boundaries of the range (for example, 1-3 2-4 specifies a range consisting of the square bounded by rows 2 and 4, and by columns 3 and 5)

r1,c1 r2,c2    The top-left and bottom-right corners of the range (for example, 1,2 3,4 specifies a range whose top-left corner is the cell (1,2) and whose bottom-right corner is the cell (3,4))

### **Example code:**

```
(0,0 0,1)(2-3 5-7)
```

## ***StringCaseSeries***

Specifies the string case for individual cells. The value of this resource is a list of cell/value combinations. Valid values include `CASE_AS_IS`, `CASE_UPPER` and `CASE_LOWER`.

### **Example code:**

```
(0 1 case_as_is)(0 2 case_upper)
```

## ***TopRow***

Specifies the non-frozen row initially visible at the top of the table display. Visual Cafe provides a text field for entering the integer value.

## ***TrackCursor***

Specifies whether the mouse pointer should be tracked as a user moves the mouse across the widget. When `trackCursor` is `true`, it means that the mouse pointer is tracked, and changes its shape to reflect the interactions available to the user at that location. By default, `trackCursor` is `false`, and the mouse pointer is not tracked. Visual Cafe provides a choice box for choosing between `true` and `false`.

## ***TraversableSeries***

Specifies the traversable property for individual cells. The value of this property is a list of cell/boolean combinations that determine whether the specified cells are traversable.

### **Example code:**

```
(0 0 true) (1 all false)
```

## ***ValidatePolicy***

Specifies when to validate changes to cell or label values against its data type (specified by `DataType`). When set to `VALIDATE_USER_EDIT`, only changes made by a user are validated. When set to `VALIDATE_SET`, only changes made by an application are validated. When set to `VALIDATE_ALWAYS`, changes made by both a user and an application are validated. When set to `VALIDATE_NEVER`, changes are never validated. Visual Cafe provides a choice box for choosing between the valid values.

### ***VertSBAttachment***

Specifies how the end of the vertical scrollbar is attached to the table. Valid values are `attach_cells` (scrollbar ends at the edge of the visible cells) and `attach_side` (scrollbar ends at the edge of the table). Visual Cafe provides a choice box for choosing between the valid values.

### ***VertSBPosition***

Specifies how the vertical scrollbar is attached to the table. Valid values are `sbposition_cells` (scrollbar is attached to the cell/label viewport) and `sbposition_side` (scrollbar is attached to the side of the table). Visual Cafe provides a choice box for choosing between the valid values.

***VertSBOffset***

Specifies the distance between the vertical scrollbar and the table. The default value is 6. Visual Cafe provides a text field for entering the integer value.

## ***VertSBDisplay***

Specifies when to display the vertical scrollbar. Valid values are `sdisplay_always` (always displayed), `sdisplay_never` (never displayed) and `sdisplay_as_needed` (displayed when the rows or columns are not visible). Visual Cafe provides a choice box for choosing between the valid values.

