graphomate

Scripting Documentation of the graphomate tiles 2.2

for SAP Lumira Designer 2.x

Version 2.2 - as of June 2019

https://www.graphomate.com

1 Scripting Documentation

Scripting Documentation of graphomate tiles for SAP Lumira Designer

Rev 2.2 as of June 2019

1.1 Introduction

The graphomate tiles extension supports the scripting language of Designer. Therefore, it is possible to determine properties of the extension at run-time. New data can be set or titles can be adapted. Furthermore, it is possible to use drilldown functionalities of the extension and thus create interactive dashboards.
This document provides a list of available functions, events and examples. We are steadily working on expanding the range of functions. If you miss certain functions, please give us a note. Therefore you can use our online form under http://www.graphomate.com/en/feedback.

1.2 Events

The following events are available. User-made scripts can be deposited on the Property Sheet under the group Events.

Event Name	Description
On element selected	Is triggered when the user clicks on an element of the chart at runtime. In order to allow a drill-down, it is necessary to react on the user input at this point. The return values of the following functions are updated at the moment of the event: getSelectedMember()getSelectedModule()
On Component selected	Is triggered when the user selects the tile component during runtime. Contrary to On element selected, no return value is set.

1.3 Functions

The following functions are available. They can be addressed in the scripting editor of Designer. We differentiate between getter and setter functions as they get information from the extension or set its properties.

Getter Name	Example	Return type	Example Return Value	Description
getBadColor()	GRAPHOMATETILE_1 .getBadColor();	String	#FF0000	Returns the hexadecimal color value of bad deviations.
getBorderColor()	GRAPHOMATETILE_1 .getBorderColor();	String	#666666	Returns the color of the component border and the slot dividers.
getBorderThickness()	GRAPHOMATETILE_1 .getBorderThickness();	int	2	Returns the thickness of the component border in px.
getComparisonGroup()	GRAPHOMATETILE_1 .getComparisonGroup();	String	A	Returns the Comparison Group the tile is registered in. An empty return value ("") indicates that the tile is not part of any Comparison Group.
getDataColor1()	GRAPHOMATETILE_1 .getDataColor1();	String	#00ff00	Returns the color for the first data series.
getDataColor2()	GRAPHOMATETILE_1 .getDataColor2();	String	#008b23	Returns the color of the second data series.
getExtendedNumberFormat()	GRAPHOMATETILE_1 .getExtendedNumberFormat();	String	-||.|,|1|2| tEUR	Returns the format string for the Extended Number Format (see appendix of the manual for the tiles for syntax).
getExtendedNumberFormatPercentage()	GRAPHOMATETILE_1 .getExtendedNumberFormatPercent();	String	-||.|,|1|2|%	Returns the format string for the Extended Number Format for percentages (see appendix of the manual for the tiles for syntax).
getFlipDuration()	GRAPHOMATETILE_1.getFlipDuration()	int	250	Returns the duration of the flip animation.
getFlipTrigger()	GRAPHOMATETILE_1.getFlipTrigger()	String	"Click"	Returns the action that triggers the flip effect. Return value can be either "None", "Click", "Double Click" or "Hover".
getGoodColor()	GRAPHOMATETILE_1 .getGoodColor();	String	#00ff00	Returns the color for positive deviations.
getGridSize()	GRAPHOMATETILE_1 .getGridSize();	int	100	Returns the grid size the tile uses to automatically calculate its size.
getLabelFormatMode()	GRAPHOMATETILE_1 .getLabelFormatMode();	String	basic	Returns whether value labels are formatted with numeral.js (basic) or the Extended Number Format (extended).
getLayout()	GRAPHOMATETILE_1.getLayout();	String	Layout100	Returns the "name" of the currently selected layout.
getLocale()	GRAPHOMATETILE_1.getLocale();	String	de	Returns the locale setting. This governs the display of delimiters and abbreviations for values formatted with numeral.js.
getModuleProperty()	GRAPHOMATETILE_1 .getModuleProperty (1, "title");	String	Total	Returns the current setting of a property for a module on the front side of a tile. The first parameter is the slot number of the module, the second parameter is the name of the property. A list of available properties can be found in the scripting help popup on the Modules tab.

getModulePropertyBackside()	GRAPHOMATETILE_1 .getModulePropertyBackside (1, "title");	String	Net	Returns the current setting of a property for a module on the back side of a tile. The first parameter is the slot number of the module, the second parameter is the name of the property. A list of available properties can be found in the scripting help popup on the Modules tab.
getNumberFormat()	GRAPHOMATETILE_1 .getNumberFormat();	String	0.0a	Returns the format string for numeral.js formatting (see appendix of the manual for the tiles for syntax).
getNumberFormatPercentage()	GRAPHOMATETILE_1 .getNumberFormatPercent();	String	0.0%	Returns the format string for numeral.js formatting for percentages (see appendix of the manual for the tiles for syntax).
getSelectedMember(Dimension dimensionKey)	GRAPHOMATETILE_1 .getSelectedMember("0D_NWI_RCOD").text;	Member		After a user has clicked an element in a given tile, this function returns the member of the specified dimension dimensionKey. The member provides access to several properties, such as text, internalKey and externalKey.
getSelectedModule()	GRAPHOMATETILE_1 .getSelectedModule();	int	2	Returns the number of the layout slot the clicked module is located in. Counting starts with one, proceeding from left to right and top to bottom.
getSelectionsForLayoutSlot(int indexOfLayoutSlots)	GRAPHOMATETILE_1.getSelectionsForLayoutSlot(1)	String[]	[{"(MEASURES_DIMENSION)":"DS:2,MEAS:id_55"}]	Returns the selection for the module that is located in the layout slot on the front side of the tile. Counting starts with one, proceeding from left to right and top to bottom.
getSelectionsForLayoutSlotBack(int indexOfLayout Slots)	GRAPHOMATETILE_1.getSelectionsForLayoutSlotBack(1)	String[]	[{"(MEASURES_DIMENSION)":"DS:2,MEAS:id_55"}]	Returns the selection for the module that is located in the layout slot on the back side of the tile. Counting starts with one, proceeding from left to right and top to bottom.
getSnapToGrid ()	GRAPHOMATETILE_1 .getSnapToGrid ();	boolean	false	Returns true if the tile determines its own height/width according to the specified grid size.
getTextColor1()	GRAPHOMATETILE_1.getTextColor1();	String	#333333	Returns the headline color.
getTextColor2()	GRAPHOMATETILE_1.getTextColor2();	String	#666666	Returns the color of the Category Labels.
getVisibleSide()	GRAPHOMATETILE_1.getVisibleSide();	String	"Front"	Returns the currently visible side. This function only makes sense if the visible side is set with setVisibleSide().

Setter Name	Example	Description
setBadColor(String)	GRAPHOMATETILE_1 .setBadColor("#FF0000");	Sets the hexadecimal color value for negative deviations.
setBorderColor(String)	GRAPHOMATETILE_1.setBorderColor("#666666");	Sets the color of the component border and the slot dividers to the passed value.
setBorderThickness(integer)	GRAPHOMATETILE_1.setBorderThickness(2);	Sets the thickness of the component border to the passed value.
setComparisonGroup(String)	GRAPHOMATETILE_1.setComparisonGroup ("A");	Registers the tile in the Comparison Group with the passed id. Components within the same Comparison Group are scaled identically.
setDataColor1(string value)	GRAPHOMATETILE_1.setDataColor1 ("#ff0000");	Sets the color of the first data series to the passed value.
setDataColor2(string value)	GRAPHOMATETILE_1.setDataColor2("#008b23");	Sets the color of the second data series to the passed value.
setExtendedNumberFormat (String)	GRAPHOMATETILE_1.setExtendedNumberFormat("-||.|,|1|2| tEUR");	Sets the format string for the Extended Number Format to the passed value (see appendix of the manual for the tiles for the correct syntax).
setExtendedNumberFormatPercentage (String)	GRAPHOMATETILE_1.setExtendedNumberFormatPercent("-||.|,|1|2|%");	Sets the format string for the Extended Number Format for percentages (see appendix of the manual for the tiles for the correct syntax).
setFlipDuration(integer)	GRAPHOMATETILE_1.setFlipDuration(250);	Sets the duration of the flip animation.
setFlipTrigger(String)	GRAPHOMATETILE_1.setFlipTrigger("Click");	Sets the trigger for the flip effect. The parameter can be either "None", "Click", "Double Click" or "Hover".
setGoodColor(String)	GRAPHOMATETILE_1.setGoodColor ("#00ff00");	Sets the color for positive deviations to the passed value.
setGridSize(integer)	GRAPHOMATETILE_1.setGridSize(100);	Sets the grid size to the passed value.
setLabelFormatMode (String)	GRAPHOMATETILE_1.setLabelFormatMode("basic");	Governs whether value labels are formatted using numeral.js or the Extended Number Format. Valid values are basic for numeral.js or extended for the Extended Number Format.
setLayout (String)	GRAPHOMATETILE_1.setLayout("Layout122");	Sets the layout of a given tile to the passed format. The following values are valid: Layout100, Layout110, Layout111, Layout112, Layout120, Layout121, Layout122,Layout200, Layout211, Layout212, Layout220, Layout221, Layout222, Layout133, Layout123, Layout333 The individual numbers represent the numbers of cells per row. Layout111 represents a three-row layout with a single cell each. Layout122 would be a layout with a single headline cell and two cells in rows 2 and 3.
setLocale(string value)	GRAPHOMATETILE_1.setLocale("de");	Sets the locale setting to the passed value. Valid values are de (German), fr (French), en (English).
setModuleProperty (integer targetModule, string valu e targetProperty, string newValue)	GRAPHOMATETILE_1.setModuleProperty(1, "title", "Title set by script");	This script command can be used to change the properties of a given viewmodule. Depending on the viewmodule, the available properties can vary. As described above, the available commands can be read (and copied/pasted) from the scripting help popup. The parameter targetModule describes the layout slot the module is placed in (in a three-row layout, this would be 1, 2, and 3). targetProperty is the name of the property to be set. The third parameter should contain the new value of the property.
setNumberFormat (string value)	GRAPHOMATETILE_1.setExtendedNumberFormat("0,0.00a");	Sets the format string for numeral.js formatting to the passed value (see appendix of the manual of the tiles for syntax)
setNumberFormatPercentage (string value)	Sets the format string for numeral.js formatting for percentages to the passed value.	GRAPHOMATETILE_1.setNumberFormatPercent("0,0.00%");
setSelection(string value selectionName, string val ue selectionString)	GRAPHOMATETILE_1.setSelection("dataSingleCell4", ’{"Keyfigures": "0D_NWI_IQTY", "0D_NWI_CHAN": "4"}’);	Sets the passed data property to the passed selection. SelectionString needs to be passed as valid JSON string. This string describes either a single row, column, or single cell in the initial view. The strings to be used for selectionName deviate slightly from those visible in the Standard Property Sheet: for a single ResultCell, this is dataSingleCell, rows or columns need to be addressed as dataCellList, each followed by the index number. The scripting help popup (see Modules tab in the manual) already contains the valid parameters for a given module. The order of the setSelection commands in the popup corresponds to the order in the Additional Property Sheet.

setSnapToGrid (boolean value)	GRAPHOMATETILE_1.setSnapToGrid (true);	If true is passed, the tile determines its own width/height according to the specified grid size.
setTextColor1(string value)	GRAPHOMATETILE_1.setTextColor1 ("#ff0000");	Sets the headline color to the passed value.
setTextColor2(string color)	GRAPHOMATETILE_1.setTextColor2 ("#666666");	Sets the Category Label color to the passed value.
setVisibleSide(string)	GRAPHOMATETILE_1.setVisibleSide("Back");	Sets the visible side to the passed value. Valid values are "Front" and "Back".
toggleFlip()	GRAPHOMATETILE_1.toggleFlip();	Triggers the flip effect once.