Slick-grid API

Tuesday, February 02, 2016   9:28 PM

	Port. Weight	Bench. Weight	Difference
Total	**100.00**	**100.00**	--
⊞ Commercial Services	0.98	1.54	-0.56
⊞ Communications	3.21	3.09	0.12
⊞ Consumer Durables	1.72	2.03	-0.32
⊞ Consumer Non-Durables	10.70	9.85	0.85
⊞ Consumer Services	6.84	7.35	-0.51
⊞ Distribution Services	1.08	1.21	-0.13
⊞ Electronic Technology	11.59	11.31	0.27
⊞ Energy Minerals	6.03	5.78	0.26
⊞ Health Services	2.65	2.71	-0.06
⊞ Health Technology	14.32	13.76	0.56
⊞ Industrial Services	1.95	2.05	-0.10
⊞ Non-Energy Minerals	0.39	0.51	-0.12
⊞ Process Industries	2.80	3.23	-0.43
⊞ Producer Manufacturing	8.09	8.21	-0.12
⊞ Retail Trade	8.33	8.12	0.21
⊞ Technology Services	12.89	12.69	0.20
⊞ Transportation	2.36	2.40	-0.04
⊞ Utilities	4.06	4.15	-0.09

Whole documentation is available at https://github.com/mleibman/SlickGrid/wiki/API-Reference

(index) -returns complete row object
    Ex: getDataItem(8).name ==> 'Energy Minerals'

getDataLength() - returns number of rows in the slick-grid
    Ex: getDataLength() ==> 19

getSelectionModel().getSelectedRows() - Returns array of currently selected rows in the order of selection.
    Ex: getSelectionModel().getSelectedRows() ==> [2]

getSelectionModel().setSelectedRows(array of row indices) - Selects the specified rows
    Ex: getSelectionModel().setSelectedRows([1, 3]) ==> selects row 1 and 3
    NOTE: it is throwing error in browser console, need to find why.

autosizeColumns() - Adjusts the columns into display size(removes the horizontal scrollbar)

getColumns() - Returns array of column objects
    Ex: getColumns()[0].id ==> returns the id of first column i.e 'name'

getColumnIndex(*id*) - return the index of column
    Ex: getColumnIndex('name') ==> 0

editActiveCell(Editor) - turns Active cell to edit mode(double clicks), returns false if the cell is not editable
    Ex: editActiveCell(Slick.Editors.Number) ==> turns active cell to number edit mode

getActiveCell() - returns object contains row index and column index of active cell

getActiveCellNode() - returns DOM element of active cell
    Ex: getActiveCellNode().getText() ==> 3.21   //getText() is protractor function

getCellNode(row, column) - returns the DOM element of particular Cell

Ex: getCellNode(3, 3).getText() ==> -0.32

gotoCell(row,column) - scrolls to the given cell and activates it.
Ex: gotoCell(3,3) ==> activates the cell 3,3
NOTE: It is able scroll to given cell in PA3 but not in BC2 as the slick-grid rendering is different in both the apps.

navigateDown() - navigates the active cell one position down, returns false if no current active cell

navigateUp() - navigates the active cell one position up, returns false if no current active cell

navigateLeft() - navigates the active cell one position left, returns false if no current active cell

navigateRight() - navigates the active cell one position right, returns false if no current active cell
//this function can be used to scroll horizontally one cell after other

setActiveCell(*row, cell*) - activates the given cell
//unlike gotoCell() it cannot scroll if the element is not visible

scrollRowIntoView(row, padding) - scrolls to the given row
Padding - If `false`, the grid will scroll so the indicated row is at the top of the view. If `true`, the grid will scroll so the indicated row is at the bottom of the view. Defaults to `false`.

scrollCellIntoView(*row, column*) - scrolls to given cell
Note that this does nothing unless the indicated **column** is already not in view. For example, if the grid is scrolled to the far left and you were looking at row 0, calling `scrollCellIntoView(100,0)` would not simply scroll you to row 100. But if column 8 were out of view and you called `scrollCellIntoView(100,8)`, then it would scroll down and to the right.

```
Sample usage of slick-grid API via protractor

browser.driver.executeScript(function() {
var nam = [];
var slickObject = $('.tf-slick-grid').data('$tfSlickGridController');
nam = slickObject.grid.getColumns();
return nam;
}).then( function( value ){
value.forEach( function( col ) {
console.log( col.name );
expect( col. ).toBe("Consumer Services");
});
});
```