$(document).ready(function() {

$('a.navigator-link').on("click", function() {

// Extract the panel for this link

var panel = getPanelName($(this));

// Mark this link as currently selected

$('.navigator-link').parent().removeClass('navigator-selected');

$(this).parent().addClass('navigator-selected');

showPanel(panel);

});

installMethodHandlers('failed');

installMethodHandlers('skipped');

installMethodHandlers('passed', true); // hide passed methods by default

$('a.method').on("click", function() {

showMethod($(this));

return false;

});

// Hide all the panels and display the first one (do this last

// to make sure the click() will invoke the listeners)

$('.panel').hide();

$('.navigator-link').first().trigger("click");

// Collapse/expand the suites

$('a.collapse-all-link').on("click", function() {

var contents = $('.navigator-suite-content');

if (contents.css('display') == 'none') {

contents.show();

} else {

contents.hide();

}

});

});

// The handlers that take care of showing/hiding the methods

function installMethodHandlers(name, hide) {

function getContent(t) {

return $('.method-list-content.' + name + "." + t.attr('panel-name'));

}

function getHideLink(t, name) {

var s = 'a.hide-methods.' + name + "." + t.attr('panel-name');

return $(s);

}

function getShowLink(t, name) {

return $('a.show-methods.' + name + "." + t.attr('panel-name'));

}

function getMethodPanelClassSel(element, name) {

var panelName = getPanelName(element);

var sel = '.' + panelName + "-class-" + name;

return $(sel);

}

$('a.hide-methods.' + name).on("click", function() {

var w = getContent($(this));

w.hide();

getHideLink($(this), name).hide();

getShowLink($(this), name).show();

getMethodPanelClassSel($(this), name).hide();

});

$('a.show-methods.' + name).on("click", function() {

var w = getContent($(this));

w.show();

getHideLink($(this), name).show();

getShowLink($(this), name).hide();

showPanel(getPanelName($(this)));

getMethodPanelClassSel($(this), name).show();

});

if (hide) {

$('a.hide-methods.' + name).trigger("click");

} else {

$('a.show-methods.' + name).trigger("click");

}

}

function getHashForMethod(element) {

return element.attr('hash-for-method');

}

function getPanelName(element) {

return element.attr('panel-name');

}

function showPanel(panelName) {

$('.panel').hide();

var panel = $('.panel[panel-name="' + panelName + '"]');

panel.show();

}

function showMethod(element) {

var hashTag = getHashForMethod(element);

var panelName = getPanelName(element);

showPanel(panelName);

var current = document.location.href;

var base = current.substring(0, current.indexOf('#'))

document.location.href = base + '#' + hashTag;

var newPosition = $(document).scrollTop() - 65;

$(document).scrollTop(newPosition);

}

function drawTable() {

for (var i = 0; i < suiteTableInitFunctions.length; i++) {

window[suiteTableInitFunctions[i]]();

}

for (var k in window.suiteTableData) {

var v = window.suiteTableData[k];

var div = v.tableDiv;

var data = v.tableData

var table = new google.visualization.Table(document.getElementById(div));

table.draw(data, {

showRowNumber : false

});

}

}