

## Web Client Utils - Feature #762

### Add to wiki as an example: show a google chart in web client

11/19/2013 11:27 AM - Juan C. Cristobo

<b>Status:</b>	New	<b>Start date:</b>	11/19/2013
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>		<b>% Done:</b>	0%
<b>Category:</b>		<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>			
<b>Browser (if web client):</b>			

#### Description

You can get the original code from Google Documentation: [https://developers.google.com/chart/interactive/docs/quick\\_start](https://developers.google.com/chart/interactive/docs/quick_start)

- First, create a variable form (i.e., 'chart'), use it as datasource of a field of type HTML\_AREA and put it in the form.

- Then, create an onShow method. Add a reference of Google libraries (as you can see in the original code from doc):

```
plugins.WebClientUtils.addJsReference('https://www.google.com/jsapi');
```

- You can assign the remaining code to 'chart' variable, but I prefer to do it this way (this is the complete code for onShow method):

```
function onShow(firstShow, event)
{
  plugins.WebClientUtils.addJsReference('https://www.google.com/jsapi');
  var datasource = [
    ['Topping', 'Slices'],
    ['Mushrooms', 3],
    ['Onions', 1],
    ['Olives', 1],
    ['Zucchini', 1],
    ['Pepperoni', 2]
  ];
  var chartOptions = {
    'title':'How Much Pizza I Ate Last Night',
    'width':400,
    'height':300
  };

  chart = '<script type="text/javascript">\n
  google.load("visualization", "1.0", {"packages":["corechart"]});\n
  google.setOnLoadCallback(drawChart);\n
  function drawChart() {\n
    var data = new google.visualization.arrayToDataTable(' + JSON.stringify(datasource) +
');\n
    var options = ' + JSON.stringify(chartOptions) + ';\n
    var chart = new google.visualization.PieChart(document.getElementById("chart_div"));
\n
    chart.draw(data, options);\n
  }\n
  </script>\n
  <div id="chart_div"></div>';
}
```

As you can see, I define the values and the chart options as variables in Servoy code and then 'inject' them using JSON.stringify, I think that the code is very clear. Of course, you can populate 'datasource' array with data from Servoy.