Interacting With the Chart

What we've covered so far is sufficient for many web pages: you've drawn your chart on the page. However, if you want to catch user clicks, or need to manipulate properties or data in a chart that you've already drawn, you need to listen for events thrown by the chart.

All charts throw some kinds of events. Here are the most common:

- *ready* - Thrown when the chart is drawn on the page and ready to respond to methods. Listen for this event if you need to request information from the chart.
- *select* - Thrown when the user selects something on the chart: typically by clicking on a bar or pie slice.
- *error* - Thrown when the chart can't render the data passed in, typically because the `DataTable` format is wrong.
- *onmouseover* and *onmouseout* - Thrown when the user mouses over or off of a specific chart element, respectively.

Listening for events is simple; simply call `google.visualization.events.addListener()` (https://developers.google.com/chart/interactive/docs/reference#addlistener) passing in a handle to the chart, the name of the event to catch, and the name of a handler to call when the event is thrown. You can call this method with any chart handle, even if you haven't called `draw()` yet. Note that you can call `google.visualization.events.addOneTimeListener()` (https://developers.google.com/chart/interactive/docs/reference#addonetimelistener) if you want the listener to be called exactly once before removing itself.

Here's a partial code snippet showing how to register to catch a chart's *select* event:

```
load libraries...

function drawChart() {

  prepare data...

  var chart = new google.visualization.PieChart(document.getElementById('char

  // The select handler. Call the chart's getSelection() method
  function selectHandler() {
    var selectedItem = chart.getSelection()[0];
    if (selectedItem) {
      var value = data.getValue(selectedItem.row, selectedItem.column);
      alert('The user selected ' + value);
    }
```

```
  }

  // Listen for the 'select' event, and call my function selectHandler() when
  // the user selects something on the chart.
  google.visualization.events.addListener(chart, 'select', selectHandler);

  draw the chart...

}
```

The following shows the Hello Charts code example with a new select event listener. Try it out yourself.

```
<html>
  <head>
    <!--Load the AJAX API-->
    <script type="text/javascript" src="https://www.gstatic.com/charts/loader
    <script type="text/javascript">

      // Load the Visualization API and the piechart package.
      google.charts.load('current', {'packages':['corechart']});

      // Set a callback to run when the Google Visualization API is loaded.
      google.charts.setOnLoadCallback(drawChart);

      // Callback that creates and populates a data table,
      // instantiates the pie chart, passes in the data and
      // draws it.
      function drawChart() {

        // Create the data table.
        var data = new google.visualization.DataTable();
        data.addColumn('string', 'Topping');
        data.addColumn('number', 'Slices');
        data.addRows([
          ['Mushrooms', 3],
          ['Onions', 1],
          ['Olives', 1],
          ['Zucchini', 1],
          ['Pepperoni', 2]
        ]);

        // Set chart options
        var options = {'title':'How Much Pizza I Ate Last Night',
                       'width':400,
                       'height':300};
```

```
        // Instantiate and draw our chart, passing in some options.
        var chart = new google.visualization.PieChart(document.getElementById

        function selectHandler() {
          var selectedItem = chart.getSelection()[0];
          if (selectedItem) {
            var topping = data.getValue(selectedItem.row, 0);
            alert('The user selected ' + topping);
          }
        }

        google.visualization.events.addListener(chart, 'select', selectHandle
        chart.draw(data, options);
      }

    </script>
  </head>
  <body>
    <!--Div that will hold the pie chart-->
    <div id="chart_div" style="width:400; height:300"></div>
  </body>
</html>
```

Next: *Security and Privacy*

(https://developers.google.com/chart/interactive/docs/security_privacy)

More Information

- Handling Events (https://developers.google.com/chart/interactive/docs/events)
- Controls and Dashboards (https://developers.google.com/chart/interactive/docs/gallery/controls)
- getSelection() (https://developers.google.com/chart/interactive/docs/reference#visgetselection)
- `google.visualization.events.addListener()` (https://developers.google.com/chart/interactive/docs/reference)

(https://developers.google.com/terms/site-policies). *Java is a registered trademark of Oracle and/or its affiliates.*

上次更新日期：一月 28, 2016