FusionCharts Free for jQuery Documentation

http://www.fusioncharts.com/jquery/

Table of Contents

1. Introduction

What is FusionCharts Free?

FusionCharts Free is an open-source FREE flash charting component that can be used to render data-driven animated charts. Developed in Macromedia Flash MX, FusionCharts can be used with any web scripting language like PHP, ASP, .NET, JSP, ColdFusion, JavaScript, Ruby on Rails etc., to deliver interactive and powerful charts. Using XML as its data interface, FusionCharts makes full use of fluid beauty of Flash to create compact, interactive and visually-arresting charts.

To know more on FusionCharts Free please refer to the FusionCharts Free website http://www.fusioncharts.com/free/

What is jQuery?

jQuery is a powerful and light-weight JavaScript Library that simplifies HTML document traversing, event handling, animating, and Ajax interactions. jQuery is primarily used to rapidly build highly interactive web applications.

To know more about jQuery, please refer to jQuery website http://jquery.com/

What is jQuery Plugin for FusionCharts Free?

FusionCharts Free for jQuery is a jQuery plugin that allows you to integrate FusionCharts Free in your web application using popular jQuery syntax.

You can insert FusionCharts Free anywhere within a web page, manipulate the chart data and chart cosmetics and even provide data to the chart from simple HTML tables. With the jQuery Plugin for FusionCharts Free, your code becomes concise but coherent.

Who Can Use It?

jQuery Plugin for FusionCharts Free is meant for web designers, jQuery enthusiasts, FusionCharts addicts and anybody who is interested in a simplified jQuery enabled charting solution.

2. Installation and Pre-requisites

Installing FusionCharts Free for jQuery merely involves some simple steps.

1. Download the FusionCharts Free for jQuery pack from http://www.fusioncharts.com/jquery/
2. Extract all files from the downloaded package.
3. Upload the extracted "FusionChartsFree" folder to your server root or to the root of your application. Make sure that you have the "Charts", "JS" and other files and folders inside this "FusionChartsFree" folder.
4. You will also require the jQuery core files.The download package already contains the core jQuery file version 1.4.2 in the "JS" folder. However, you can also download the latest version of jQuery from http://docs.jquery.com/Downloading_jQuery and follow the instructions on jQuery website on how to install and use jQuery.

Installation Notes

For uploading files to your web-server or local-server you might need FTP access or some other means to access the files of your server. Contact your hosting provider for more information on this.

For using FusionCharts Free on your local or testing machine you simply need to copy-paste the "FusionChartsFree" folder to the folder from where your project is running.

You can verify whether your installation went right by simply running the samples from the "Examples" folder. For instance, if you have uploaded your files to your server-root, then opening http://www.yourdomain.com/FusionChartsFree/ from your browser, will present you with links that you can use to see the samples.

3. Plugin Usage

Creating Your First Chart

Here, we will guide you through the process of creating your first chart. For a head start, we'll create a simple 2D Column chart to visually depict Annual Sales Summary.

To create a chart you need to create an HTML file. This file will contain the codes to render a chart.

Example Code To Create FusionCharts Using jQuery Plugin

You may copy the code provided below and paste it onto a new HTML file using your favorite text-editor.

```
<html>
<head>
<title>Sample | jQuery Plugin For FusionCharts Free</title>
<script type="text/javascript" src="FusionChartsFree/JS/jquery-
1.4.2.js"></script>
<script type="text/javascript"
src="FusionChartsFree/JS/jquery.fusioncharts.js"></script>
</head>
<body><center>
  <div id="myChartContainer">Loading FusionCharts!</div>
  <script type="text/javascript">
  $('#myChartContainer').insertFusionCharts({
      type: "Column3D", width: "400", height: "300",
      data: "<graph caption='Annual Sales Summary (2004 - 2007)'" +
            "xAxisName='Year' yAxisName='Sales' numberPrefix='$'>" +
            "<set name='2004' value='37800' color='AFD8F8' />" +
            "<set name='2005' value='21900' color='F6BD0F' />" +
            "<set name='2006' value='32900' color='8BBA00' />" +
            "<set name='2007' value='39800' color='FF8E46' /></graph>",
      dataFormat: "XMLData"
  });
  </script>
</center>
</body>
</html>
```

Open this HTML file in your preferred browser. You should now see a chart like this:

How the Above Code Works

Firstly the jQuery core JavaScript file and the FusionCharts Free jQuery plugin i.e. jquery.fusioncharts.js file are included within the `<head>` section.

```
<script type="text/javascript" src="FusionChartsFree/JS/jquery-1.4.2.js"></script>
<script type="text/javascript" src="FusionChartsFree/JS/jquery.fusioncharts.js"></script>
```

A DIV is created in the page body with a unique id, where the chart will be generated.

```
<div id="myChartContainer">Loading FusionCharts!</div>
```

A new FusionCharts object is inserted in the DIV using the `insertFusionCharts` function.

```
$('#myChartContainer').insertFusionCharts({ type: 'Column2D' });
```

The plugin options are passed through the `insertFusionCharts` function in JSON format. These are discussed elaborately in the "Plugin Options" section.

Note: If you do not specify a chart type then, by default, a Column2D chart will be rendered.

Providing the File Paths

While you are building the code, you must remember to provide correct path of the files. Here, in the example, we have provided the paths of the files with respect to the directory structure of the Download Pack when uploaded to server or domain root. You can, of course, make your own directory structure and accordingly provide the paths.

Note: Within the Download Pack, the jQuery Core file (v1.4.2) and the FusionCharts Free jQuery Plugin file are placed inside FusionChartsFree/JS folder and the chart SWFs are placed inside the FusionChartsFree/Charts folder.

Building the Data XML

FusionCharts necessarily needs its data in pre-defined XML format. For detailed description of building the XMLs for FusionCharts Free please refer to FusionCharts Free documentation.

Other formats of data are also supported by this plugin. To know more about them, go to the "Providing Data To Charts" section.

Troubleshooting Your First Chart

After running the HTML file in the browser, if your division displays the text "`Loading FusionCharts!`"

a) Check if the paths for the Core jQuery file and the jquery.fusioncharts.js file have been properly provided.

b) Check if all the plugin options are provided in their correct format.
Specifically check the following:

1. The double quotation marks (") or single quotation marks (') enclose the plugin option values.
2. The plugin option-value pairs are separated by a comma.
3. Each plugin option and its corresponding value is separated by a colon (:).
4. The last element does not have a comma after it.

For more information on troubleshooting, please refer to our "Troubleshooting" section.

Passing Options to Plugin

The plugin options are passed to the chart in JSON format.

```
$('#myChartContainer').insertFusionCharts({
    type: "Column3D", width: "400", height: "300",
    data: "<graph caption='Annual Sales Summary (2004 - 2007)'" +
          "xAxisName='Year' yAxisName='Sales' numberPrefix='$'>" +
          "<set name='2004' value='37800' color='AFD8F8' />" +
          "<set name='2005' value='21900' color='F6BD0F' />" +
          "<set name='2006' value='32900' color='8BBA00' />" +
          "<set name='2007' value='39800' color='FF8E46' /></graph>",
   dataFormat: "XMLData"
});
```

Tips on Passing Options to FusionCharts jQuery Plugin

1. The entire plugin option value pairs section are enclosed within parenthesis.

   ```
   {option1: "value1", option2: "value2", option3: "value3",
   lastOption: "last value"}
   ```

2. Each plugin option and its corresponding value is separated by a colon (:).

   ```
   { option_name: "option_value" }
   ```

3. The plugin option-value pairs are separated by a comma.

   ```
   {option1: "option1 value", option2: "option2 value" }
   ```

4. The quotation marks (") or single quotation marks (') enlcose the plugin option values.

   `"option value"` or `'option value'`

5. The last element does NOT have a comma after it. Else a Runtime Error will be encountered in Internet Explorer.

   ```
   {   type: "Column3D",
       data: "data.xml",
       dataFormat: "URIData" /* No comma after the last option */ }
   ```

Basic Options Reference

The basic options that can be passed to the `insertFusionCharts` function are listed in the table below:

#	Option	Description
1.	id	Here you provide the HTML ID of the chart., which you can later use for direct reference to the chart. In case, you do not provide any id, an ID is automatically generated.
2.	type	Here you provide the type of chart you wish to render. The list of chart aliases is provided in the “Chart Type Aliases" section.
3.	width	Sets the width of the chart in pixels or percent.
4.	height	Sets the height of the chart in pixels or percent.
5.	swfPath	Specifies the path (URL that refers to the directory) where the FusionCharts .swf files exist.
6.	data	Here you provide the data source for the chart. The data source can be string XML data or the URL of the XML document. It can also be a reference to a HTML <table> element containing your data.
7.	dataFormat	Specifies the type of data passed in the 'data' option. For example, if 'data' contains string XML then, dataFormat : "XMLData". In case you are using the URL of an external XML document, then set dataFormat: "URIData"

For detailed explanation of the above options as well as for some advanced options, please refer to the "Plugin Options" section.

Providing Data to Charts

You can provide data to charts in form of XML. (For detailed description on building the XMLs for FusionCharts Free, refer to http://www.fusioncharts.com/free/docs/.)

There are three data related options that are passed to the `insertFusionCharts` function. They are `data`, `dataFormat` and `dataOptions`.

Option	Description
`data`	In this option you pass the data for the chart. The data provided can be string XML, URL of an XML file or an HTML table.
`dataFormat`	Specifies if the value passed in the 'data' option is a string XML (XMLData), URL of the XML (URIData) or is a reference ID of an HTML <table> element containing the data (HTMLTable).
`dataOptions`	Provide data-format specific options which are applicable when using HTMLData format.The parameters that can be passed in the dataOptions have been provided in the "Plugin Options" section.

Passing XML String As Data

When you are passing data in string XML format you need to provide the XML to the data option and explicitly set `dataFormat` to "XMLData".

An example HTML code:

```
<html>
<head>
<title>XMLData Sample | jQuery Plugin For FusionCharts Free</title>
<script type="text/javascript" src="FusionChartsFree/JS/jquery-
1.4.2.js"></script>
<script type="text/javascript"
src="FusionChartsFree/JS/jquery.fusioncharts.js"></script>
</head>
<body>
  <div id="myChartContainer">Loading FusionCharts!</div>
  <script type="text/javascript">
```

```
  $('#myChartContainer').insertFusionCharts({
      type: "Column3D", width: "400", height: "300",
      data: "<graph caption='Annual Sales Summary (2004 - 2007)' " +
            "xAxisName='Year' yAxisName='Sales' numberPrefix='$'>" +
            "<set name='2004' value='37800' color='AFD8F8' />" +
            "<set name='2005' value='21900' color='F6BD0F' />" +
            "<set name='2006' value='32900' color='8BBA00' />" +
            "<set name='2007' value='39800' color='FF8E46' /></graph>",
      dataFormat: "XMLData"
  });
  </script>
</body>
</html>
```

Using External XML Document as Data Source

When you are passing the URL of an external XML document to the data option you need to set `dataFormat` to "URIData".

An example HTML code:

```
<html>
<head>
<title>URIData Sample | jQuery Plugin for FusionCharts Free</title>
<script type="text/javascript" src="FusionChartsFree/JS/jquery-
1.4.2.js"></script>
<script type="text/javascript"
src="FusionChartsFree/JS/jquery.fusioncharts.js"></script>
</head>
<body>
  <div id="myChartContainer">Loading FusionCharts!</div>
  <script type="text/javascript">
  $('#myChartContainer').insertFusionCharts({
      type: "Column3D", width: "400", height: "300",
      data: "FusionChartsFree/Examples/annual-sales-summary.xml",
      dataFormat: "URIData"
  });
  </script>
</body>
</html>
```

Using Simple HTML <table> As Data Source

FusionChartsFree jQuery Plugin has introduced this new feature in FusionCharts for retrieving data from HTML tables.

In this section we will deal with fetching data from a simple HTML table.

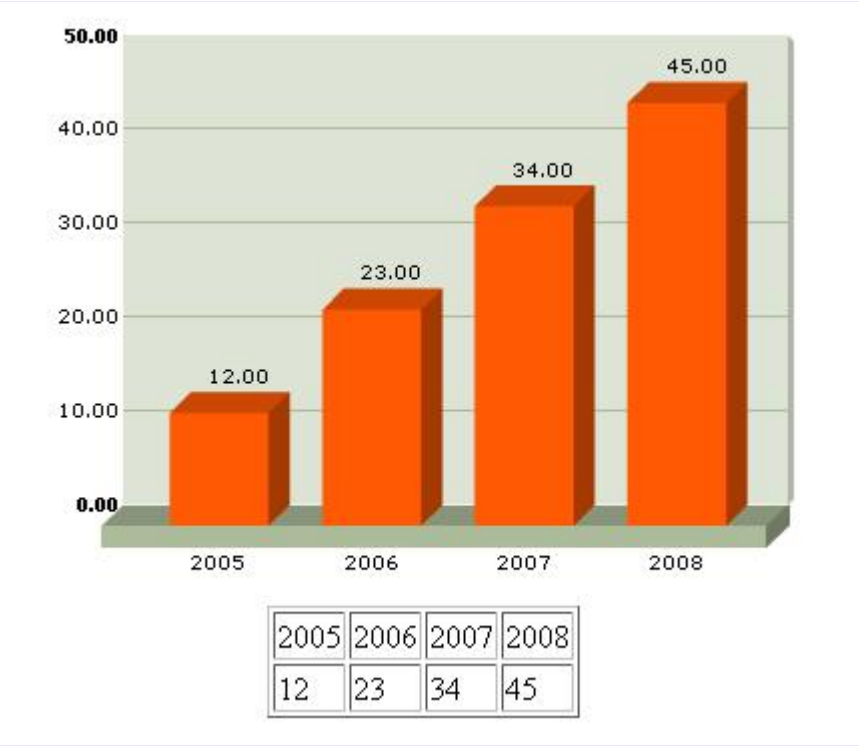
Here is a simple example:

```
<html>
<head>
<title>HTMLTable Sample | jQuery Plugin for FusionCharts Free</title>
<script type="text/javascript" src="FusionChartsFree/JS/jquery-
1.4.2.js"></script>
<script type="text/javascript"
src="FusionChartsFree/JS/jquery.fusioncharts.js"></script>
</head>
<body>
  <div id="myChartContainer" align="center">Loading FusionCharts!</div>
  <table id="myHTMLTable" border="1" align="center">
    <tr> <td>2005</td> <td>2006</td> <td>2007</td> <td>2008</td> </tr>
    <tr> <td>12</td> <td>23</td> <td>34</td> <td>45</td> </tr>
  </table>
  <script type="text/javascript">
  $('#myChartContainer').insertFusionCharts({
      type: "Column3D", width: "400", height: "300",
      data: "#myHTMLTable", dataFormat: "HTMLTable"
  });
  </script>
</body>
</html>
```

In the example above, the Id of HTML Table containing the data is set to “myHTMLTable”.

The table is selected by prefixing a hash (#) to the Id and passed on to the data option. Alternatively, you can also pass a jQuery object that has your table pre-selected. You can also select an HTML table ussing jQuery and directly run convertToFusionCharts function on it. The details of such features are explained in HTMLTable sub-section of Data Sources section.

On running this application in the browser your well get to see the output below:



2005	2006	2007	2008
12	23	34	45

Thus, your data represented by a simple HTML table is transformed into a visual format with our charts simply by selecting the table.

4. Plugin Options

This section primarily explains the major options that can be passed to FusionCharts Free jQuery Plugin.

These options can be passed to four plugin functions: insertFusionCharts(), appendFusionCharts(), prependFusionCharts() and convertToFusionCharts().

Option	Description
id	Sets an HTML Id for the chart.
type	Defines the type of chart to be rendered.
width	Sets the width of the chart.
height	Sets the height of the chart.
swfPath	Defines the URL of the SWF file (excluding the file name.)
data	Here you provide the string XML or URL of an XML document or any of the other data-sources supported by the plugin. **Note:** This parameter is not applicable to convertToFusionCharts.
dataFormat	Defines if the type of data passed in the 'data' option is a string XML (XMLData), URL of the XML (URIData) or is an HTML table (HTMLTable).
dataOptions	dataOptions provide data-format specific options. It is applicable when using HTMLTable format. This has been discussed later in this section.
wMode	Defines the Window Mode for the chart.
className	Sets the CSS "class" of the object/embed tag.
insertMode	Allows you to specify whether to append, prepend FusionCharts to an HTML element or whether to clear contents of the HTML element and then insert FusionCharts.

Note: If you do not provide options which already have a default value, the chart will be rendered with those default values. For example, if you do not specify a chart type in the 'type' option, then, by default, a Column2D chart will be rendered.

id

When you have multiple charts on a page and you need to select a particular chart among them, you require to access the chart through its Id. If you have provided unique Ids for each chart, your task of selecting that particular chart becomes way easier. If you do not specify an Id for a chart, the Id will be Auto-generated by default.

If your jQuery selector returns more than one elements within which FusionCharts are to be inserted, then the id for the chart first in the selection will be same as the id provided. Every subsequent chart in the selection will have the same id but suffixed with numeric 1, 2, 3...

Optional: Yes

Example Implementation:

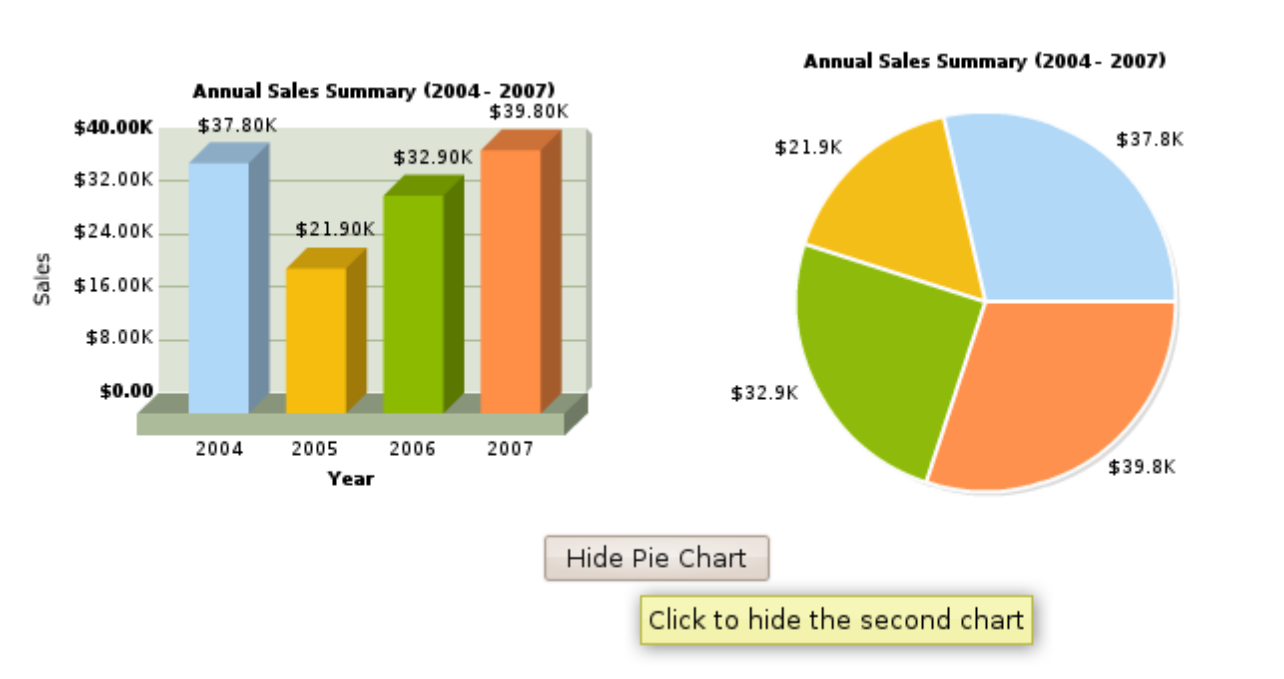
```
<html>
<head>
<title>id Sample | jQuery Plugin for FusionCharts Free</title>
<script type="text/javascript" src="FusionChartsFree/JS/jquery-
1.4.2.js"></script>
<script type="text/javascript"
src="FusionChartsFree/JS/jquery.fusioncharts.js"></script>
<script type="text/javascript" >
var hideChart = function(){
    $('#chart2').hide(); // This will hide the chart with Id 'chart2'
    $('#hideButton').hide();
}
</script>
</head>

<body> <center>
  <span id="myChart1Container">Loading FusionCharts!</span>
  <span id="myChart2Container">Loading FusionCharts!</span>
  <div><input id='hideButton' type='button' value='Hide Pie Chart'
      onClick='hideChart();' title='Click to hide the second chart' /></div>
  <script type="text/javascript">
  $('#myChart1Container').insertFusionCharts({
      id: "chart1",
      type: "Column3D",
      data: "FusionChartsFree/Examples/annual-sales-summary.xml",
      dataFormat: "URIData"
  });
```

```
    $('#myChart2Container').insertFusionCharts({
        id: "chart2",
        type: "Pie2D",
        data: "FusionChartsFree/Examples/annual-sales-summary.xml",
        dataFormat: "URIData"
    });
    </script>
</center></body>
</html>
```

Using this code we have rendered two FusionCharts in a page. Now, if you wish to hide any one of the two charts, you select that chart with its Id and then use the hide() function of jQuery to hide it.

On running this on your browser you will get to see the two charts.



Now, when you click on the 'Hide Pie Chart' button, the Pie Chart will get hidden.

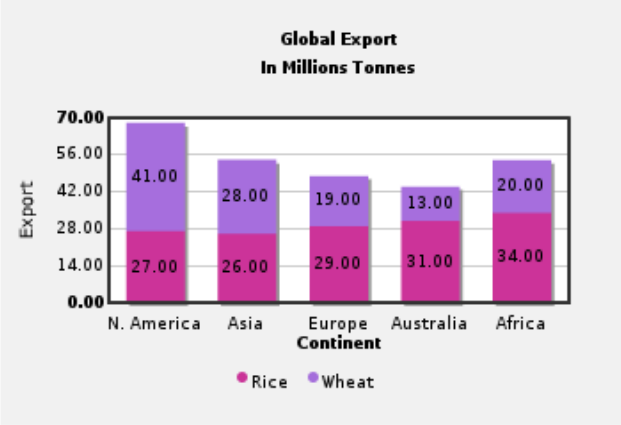
type

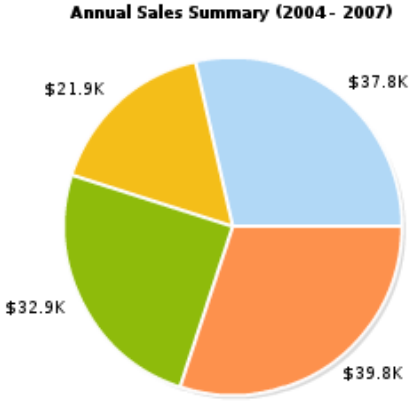
When you are trying to use a specific chart type, you have to use this option to set the type of chart. You can choose the chart type from the list of chart aliases provided in the Chart Aliases section. If you do not specify a chart type, then a Column2D chart will be rendered, by default.

Optional: Yes

Example Implementation:

```
<html>
<head>
<title>type Sample | jQuery Plugin for FusionCharts Free</title>
<script type="text/javascript" src="FusionChartsFree/JS/jquery-
1.4.2.js"></script>
<script type="text/javascript"
src="FusionChartsFree/JS/jquery.fusioncharts.js"></script>
</head>
<body><center>
    <span id="myChart1Container">Loading FusionCharts!</span>
    <span id="myChart2Container">Loading FusionCharts!</span>
    <script type="text/javascript">
    $('#myChart1Container').insertFusionCharts({
        type: "StackedColumn2D",
        data: "FusionChartsFree/Examples/global-export.xml",
        dataFormat: "URIData", width: "450"
    });
    $('#myChart2Container').insertFusionCharts({
        type: "Pie2D",
        data: "FusionChartsFree/Examples/annual-sales-summary.xml",
        dataFormat: "URIData", width: "375"
    });
    </script>
</center></body>
</html>
```





Thus, by specifying the chart name in the type attribute you are able to generate different types of charts.

Note: If you specify a chart type but the value for the swfPath option is set incorrectly, then the chart aliases for the SWF files will not fetch the SWF files.

width

This option is used to set the width of the chart in pixels. If you do not specify the width option, the chart will take the default width of 320 pixels.

Optional: Yes

height

This option is used to set the height of the chart in pixels. If you do not specify the height option, the chart will take the default height of 240 pixels.

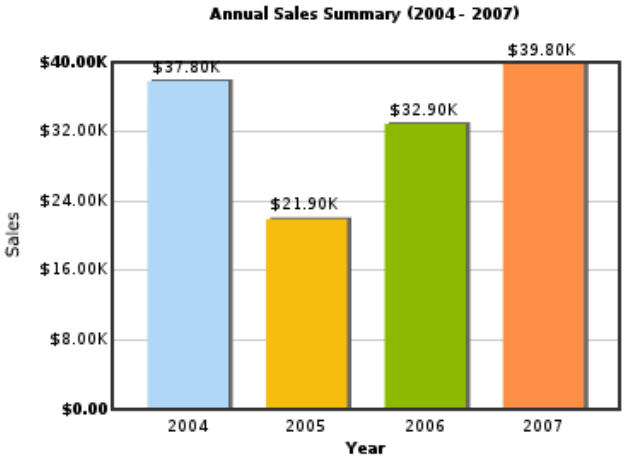
Optional: Yes

Here is a example showing how to use the width and height options:

```
<html>
<head>
<title>Sample | jQuery Plugin for FusionCharts Free</title>
<script type="text/javascript" src="FusionChartsFree/JS/jquery-
1.4.2.js"></script>
<script type="text/javascript"
src="FusionChartsFree/JS/jquery.fusioncharts.js"></script>
</head>

<body><center>
  <span id="myChart1Container">Loading FusionCharts!</span>
  <span id="myChart2Container">Loading FusionCharts!</span>
  <script type="text/javascript">
  $('#myChart1Container').insertFusionCharts({
      type: "Column2D",
      data: 'FusionChartsFree/Examples/annual-sales-summary.xml',
      dataFormat: "URIData",
      width: "300", height: "200"
  });
  $('#myChart2Container').insertFusionCharts({
      type: "Column2D",
      data: 'FusionChartsFree/Examples/annual-sales-summary.xml',
      dataFormat: "URIData",
```

```
        width: "400", height: "300"
    });
    </script>
</center></body>
</html>
```



swfPath

This option is used to define the URL of the SWF file (excluding the file name). If you have followed the installation process described in the "Installation and Pre-requisites" section, you need not provide it. By default the path is set as "FusionChartsFree/Charts". In case you are maintaining your own directory structure or if your test HTMLdocument resides somehwere else than the site root, you will have to provide a correct value of this option.

If your FusionChartsFree package exists in a folder named myFolder in your application folder, you need to set swfPath: "myFolder/FusionChartsFree/Charts".

Optional: Yes

The example code below shows how to set the swfPath in case you are referring to FusionCharts from a folder one-level deep with respect to the server root.

```
<html>
<head>
<title>jQuery Plugin for FusionCharts Free</title>
<script type="text/javascript" src="../FusionChartsFree/JS/jquery-
1.4.2.js"></script>
<script type="text/javascript"
src="../FusionChartsFree/JS/jquery.fusioncharts.js"></script>
</head>
<body>
  <div id="myChartContainer">Loading FusionCharts!</div>
  <script type="text/javascript">
  $('#myChartContainer').insertFusionCharts({
      swfPath: "../FusionChartsFree/Charts",
      type: "Column3D",
      data: "../FusionChartsFree/Examples/annual-sales-summary.xml",
      dataFormat: "URIData", width: "400", height: "300"
  });
  </script>
</body>
</html>
```

data

In this option you provide the string XML/the URL of the XML file/data-source in an HTML table. If you are using the insertFusionCharts() function, this option becomes mandatory. This option is

not required to be passed when using convertToFusionCharts().

Optional: No (Optional for `convertToFusionCharts` function)

The usage of this option has been discussed elaborately in the "Data Sources" section.

dataFormat

The dataFormat option defines the type of data passed in the 'data' option. It specifies whether the data is a string XML, URL of an XML document or an HTML table. In case you are using your own data-handling function, you need to provide its identifier name here.

Optional: No (Optional for `convertToFusionCharts` function)

Type of data	Values for dataFormat
String XML	XMLData
URL of an XML document	URIData
Data from an HTML table	HTMLTable

The usage of this option has been discussed in the "Data Sources" section.

dataOptions

The dataOptions parameter is used to pass data-format specific configurations. The parameters inside the dataOptions are passed in JSON format.

dataOptions are specific to individual data-formats and their name and usage differs between different data-formats.

Details of what configurations you can send to the dataOptions parameter, has been discussed in "Data Sources" section.

wMode

This option sets the Window Mode for the chart. By default a chart is rendered in window mode. Setting the value of `wMode` to 'window' renders the chart on top of all the underlying HTML elements.

You can also set the wMode of the chart to transparent or opaque. When you require to render HTML elements on top of the chart, you need to use these two window modes.

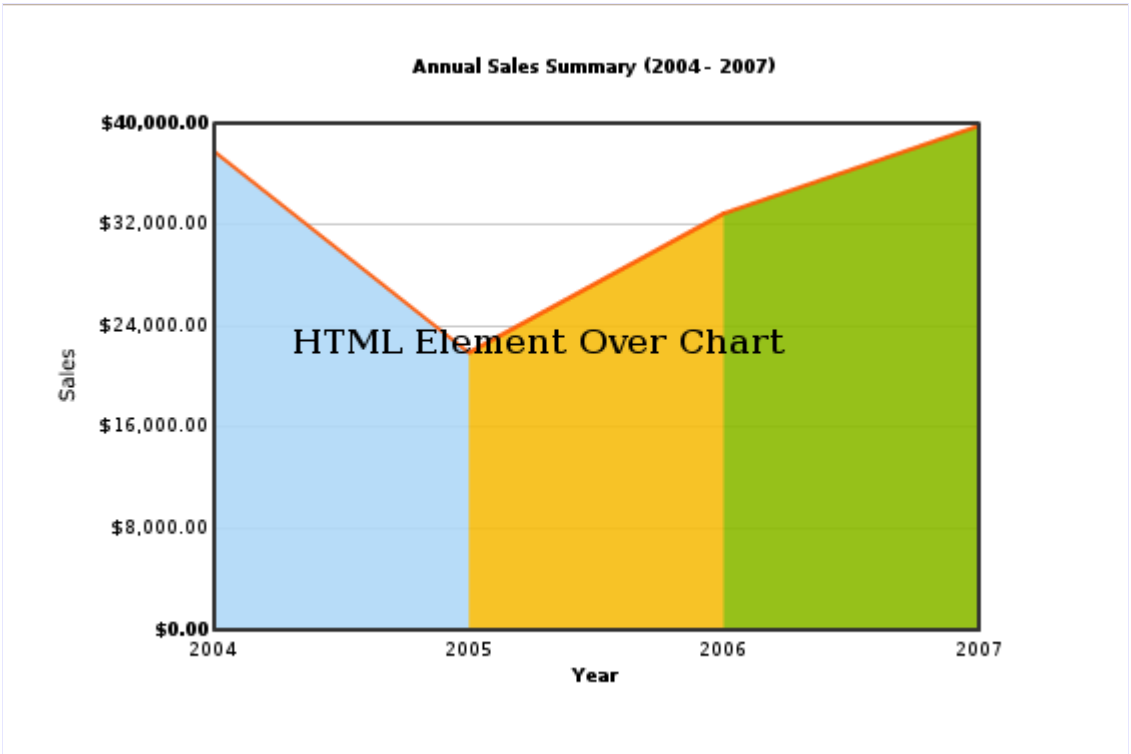
The example below shows how to set the window mode for a chart so that an HTML element can be rendered over it.

```
<html>
<head>
<title>wMode Samples | jQuery Plugin for FusionCharts Free</title>
<script type="text/javascript" src="FusionChartsFree/JS/jquery-
1.4.2.js"></script>
<script type="text/javascript"
src="FusionChartsFree/JS/jquery.fusioncharts.js"></script>
</head>

<body><center>
  <div id="myChartContainer" align="center">Loading FusionCharts!</div>
  <div id="overChart" style="margin-top: -200px;" align="center">
      <strong>HTML Element Over Chart</strong>
  </div>
  <script type="text/javascript">
  $('#myChartContainer').insertFusionCharts({
      type: "Area2D",
      data: 'FusionChartsFree/Examples/annual-sales-summary.xml',
      dataFormat: "URIData",
      width: "500",
      height: "350",
      wMode: "transparent"
  });
  </script>

</center></body>
</html>
```

When you run this on your browser, you should see a chart like this:



className

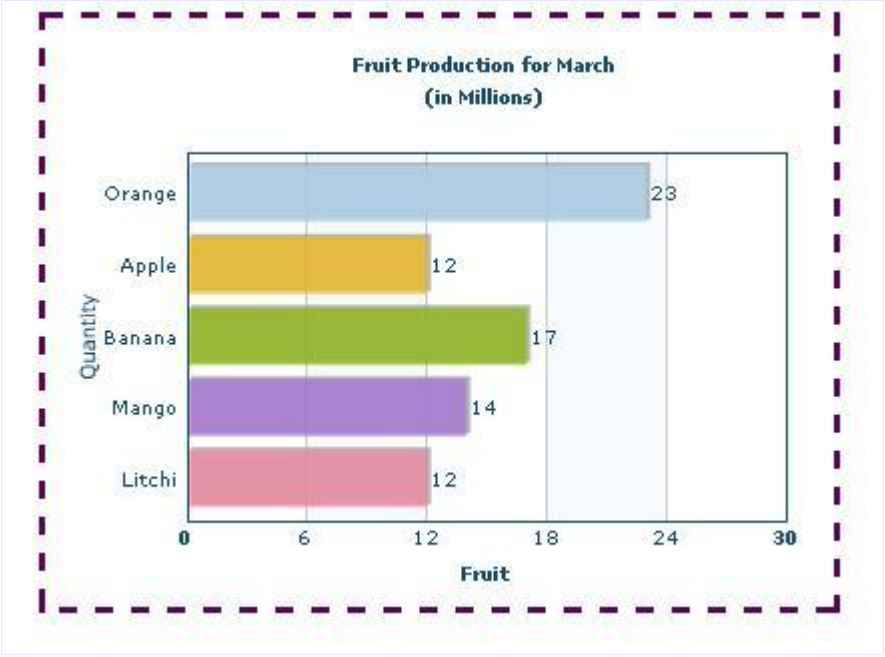
This option sets the CSS "class" of the object/embed tag. For the object it adds class="className"and for its container element (div or span) it adds class="className-container"

The code below adds a border to the chart by applying a CSS Class to the chart division.

```
<html>
<head>
<title>className Sample | jQuery Plugin For FusionCharts Free</title>
<script type="text/javascript" src="FusionChartsFree/JS/jquery-
1.4.2.js"></script>
<script type="text/javascript"
src="FusionChartsFree/JS/jquery.fusioncharts.js"></script>
<style type="text/css">
.borderedChart {
    border-color: #5C005C;
    border-width: 3px;
    border-style: dashed;
}
</style>
</head>

<body><center>
  <div id="myChartContainer" align="center">Loading FusionCharts!</div>
  <script type="text/javascript">
  $('#myChartContainer').insertFusionCharts({
      type: "Bar2D",
      data: 'FusionChartsFree/Examples/fruit-production.xml',
      dataFormat: "URIData",
      width: "400", height: "300",
      className: "borderedChart"
   });
   </script>
</center></body>
</html>
```

When you run the code on your browser, you will get a chart as shown below:



5. Plugin Functions

FusionCharts Free jQuery plugin adds new functions to jQuery. These functions allow easy insertion and manipulation of FusionCharts anywhere inside your web-page using simple jQuery syntax.

It also extends the jQuery selectors for easy access to FusionCharts elements.

insertFusionCharts()

Inserts a new FusionCharts object inside selected HTML elements.

.insertFusionCharts(options)

.appendFusionCharts(options)

.prependFusionCharts(options)

convertToFusionCharts()

Converts a selected HTML table containing data to a new FusionCharts object.

.convertToFusionCharts(options)

:FusionCharts Selector

Easily access FusionCharts objects using jQuery selectors.

.insertFusionCharts()

Inserts a new FusionCharts object within selected HTML elements.

.insertFusionCharts(options)

Description:

Inserts a new FusionCharts within every item of selection.

Basic Syntax:

```
$('container-selector').insertFusionCharts(options);
```

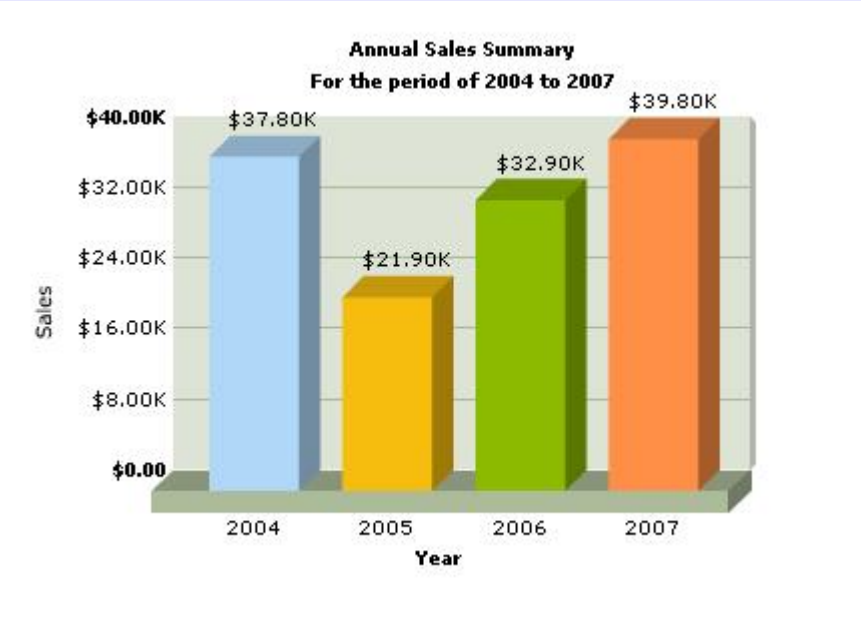
- 'container-selector' is the jQuery selector that selects one or more html elements (like or) where FusionCharts is to be inserted.
- 'options' is the JavaScript object that passes various plugin specific configrations.

An example HTML code:

```
<html>
<head>
<title>jQuery Plugin for FusionCharts Free</title>
<script type="text/javascript" src="FusionChartsFree/JS/jquery-
1.4.2.js"></script>
<script type="text/javascript"
src="FusionChartsFree/JS/jquery.fusioncharts.js"></script>
</head>
<body>
  <div id="myChartContainer">Loading FusionCharts!</div>
  <script type="text/javascript">
      $('#myChartContainer').insertFusionCharts({
          type: "Column3D",
          data: "FusionChartsFree/Examples/annual-sales-summary.xml",
          dataFormat: "URIData",
          width: "400", height: "300"
      });
  </script>
</body> </html>
```

This code will insert a Column3D chart in the DIV with ID 'myChartContainer'.

If you run it in your preferred browser you should see a chart like this:



Parameter	Description	Data-type	Possible Values	Optionality
`options`	Configuration options to generate FusionCharts. All options are passed in key-value pairs (JSON format).	JSON object	The major Plugin Options that can be passed have been discussed in the Plugin Options section.	*required*

.appendFusionCharts(options)

Description:

Appends a new FusionCharts to every item of selection.

Basic Syntax:

```
$('container-selector').appendFusionCharts(options);
```

- 'container-selector' is the jQuery selector that selects one or more html elements (like or) to which FusionCharts is to be appended.
- 'options' is the JavaScript object that passes various plugin specific configrations.

Parameter	Description	Data-type	Possible Values	Optionality
`options`	Configuration options to generate FusionCharts. All options are passed in key-value pairs (JSON format).	JSON object	The major Plugin Options that can be passed have been discussed in the Plugin Options section.	*required*

.prependFusionCharts(options)

Description:

Prepends a new FusionCharts to every item of selection.

Basic Syntax:

```
$('container-selector').prependFusionCharts(options);
```

- 'container-selector' is the jQuery selector that selects one or more html elements (like or) to which FusionCharts is to be prepended.
- 'options' is the JavaScript object that passes various plugin specific configrations.

Parameter	Description	Data-type	Possible Values	Optionality
`options`	Configuration options to generate FusionCharts. All options are passed in key-value pairs (JSON format).	JSON object	The major Plugin Options that can be passed have been discussed in the Plugin Options section.	*required*

.convertToFusionCharts()

Converts a selected HTML table containing data to a new FusionCharts object.

.convertToFusionCharts(options)

Description:

Converts a selected HTML table containing data into the corresponding XML and inserts a new FusionCharts object in a division just above the table.

Basic Syntax:

```
$('datasource-selector').convertToFusionCharts(options);
```

- 'datasource-selector' is the jQuery selection that returns one <table> element. In case more than one elements are returned, it works using the first table element.
- 'options' is the JavaScript object that passes various plugin specific configrations.

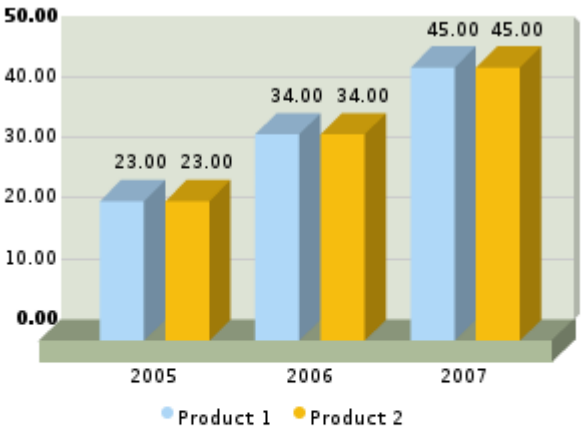
An example HTML code:

```
<html>
<head>
<title>jQuery Plugin for FusionCharts Free</title>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<script type="text/javascript" src="FusionChartsFree/JS/jquery-
1.4.2.js"></script>
<script type="text/javascript"
src="FusionChartsFree/JS/jquery.fusioncharts.js"></script>
</head>
<body>
  <div id="myChartContainer" align="center">Loading FusionCharts!</div>
  <table id="myHTMLTable" border="1" align="center">
    <tr><th> </th> <th>2005</th> <th>2006</th> <th>2007</th> </tr>
    <tr> <td>Product 1</td> <td>23</td> <td>34</td> <td>45</td> </tr>
    <tr> <td>Product 2</td> <td>23</td> <td>34</td> <td>45</td> </tr>
  </table>
  <script type="text/javascript">
  $('#myHTMLTable').convertToFusionCharts({
```

```
        type: "MSColumn3D",
        data: "#myHTMLTable",
        dataFormat:"HTMLTable"
    });
    </script>
</body>
</html>
```

This code will render a Column3D chart in a DIV just above the HTML table.

If you run it in your preferred browser you should see a chart like this:



	2005	2006	2007
Product 1	23	34	45
Product 2	23	34	45

Parameter	Description	Data-type	Possible Values	Optionality
`options`	Configuration options to generate FusionCharts. All options are passed in key-value pairs (JSON format).	JSON object	The major Plugin Options that can be passed have been discussed in the Plugin Options section.	*required*

:FusionCharts Selector

jQuery selector to easily access FusionCharts objects using jQuery selectors.

Basic syntax:

```
$('selector:FusionCharts');
```

An example JavaScript code:

```
<script type="text/javascript"
function hideAllCharts() {
    return $('object:FusionCharts, embed:FusionCharts').hide();
};
</script>
```

The function hideAllCharts in above code will hide all charts on a page.

6. Data Sources

FusionCharts Free can natively accept XML as the only data-source. This XML can either be directly passed to the chart or loaded from an external XML document. For detailed description on building the XMLs for FusionCharts Free, refer to FusionCharts Free documentation.

Added to the above, jQuery plugin for FusionCharts Free has certain functions (that we call data-handlers) that adds the functionality of handling some more data-formats.

- XMLData
- URIData
- HTMLTable

There are 3 data related options that are passed to the `insertFusionCharts` function.

Namely: data, dataFormat and dataOptions

Options	Description
`data`	In this option you pass the data for the chart. The data provided can be string XML, URL of an XML document or it can also be a reference to a HTML <table> element containing your data.
`dataFormat`	Specifies if the type of data passed in the 'data' option is a string XML (XMLData), URL of an XML document(URIData) or is an HTML table(HTMLTable).
`dataOptions`	dataOptions provide data-format specific options which are applicable when using HTMLData format.

XMLData

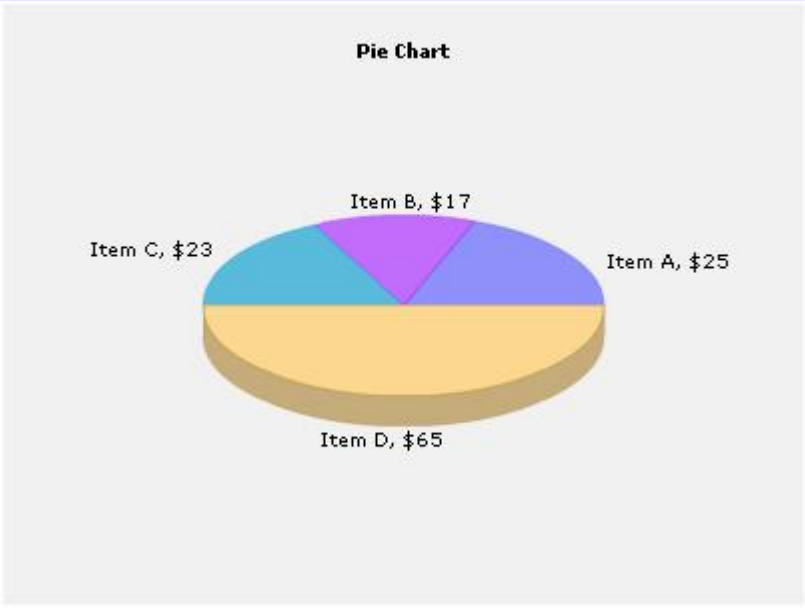
In this method, the string XML data exists in the same web-page in which FusionCharts is embedded. When working with this method, you do not need to create an external XML document. You can simply provide it in the web-page itself.

The example HTML code:

```
<html>
<head>
<title>XMLData Sample | jQuery Plugin for FusionCharts Free</title>
<script type="text/javascript" src="FusionChartsFree/JS/jquery-
1.4.2.js"></script>
<script type="text/javascript"
src="FusionChartsFree/JS/jquery.fusioncharts.js"></script>
</head>

<body><center>
  <div id="myChartContainer">Loading FusionCharts!</div>
  <script type="text/javascript">
  $('#myChartContainer').insertFusionCharts({
      type: "Pie3D",
      data: "<graph caption='Pie Chart' bgColor='F1f1f1' " +
            "decimalPrecision='0' showPercentageValues='0' showNames='1' " +
            "numberPrefix='$' showValues='1' showPercentageInLabel='0' " +
            "pieYScale='45' pieBorderAlpha='40' pieFillAlpha='70'" +
            "pieSliceDepth='15' pieRadius='100'>" +
            "<set value='25' name='Item A' color='6666FF' />" +
            "<set value='17' name='Item B' color='AD33FF' />" +
            "<set value='23' name='Item C' color='19A3D1' />" +
            "<set value='65' name='Item D' color='FFCC66' /></graph>",
      dataFormat: "XMLData",
      width: "400", height: "300"
  });
  </script>
</center></body>
</html>
```

If you run it in your preferred browser you should see a chart like this:



Note: When your XML size is considerably small, it is advisable to use the XMLData method.

Option	Description
`data`	This option contains the string XML data. e.g. data: "<graph><set value='5' /><set value='10' /></graph>"
`dataFormat`	When your **data** contains a string XML, the **dataFormat** option should be specified as **XMLData**. e.g. dataFormat: "XMLData"

URIData

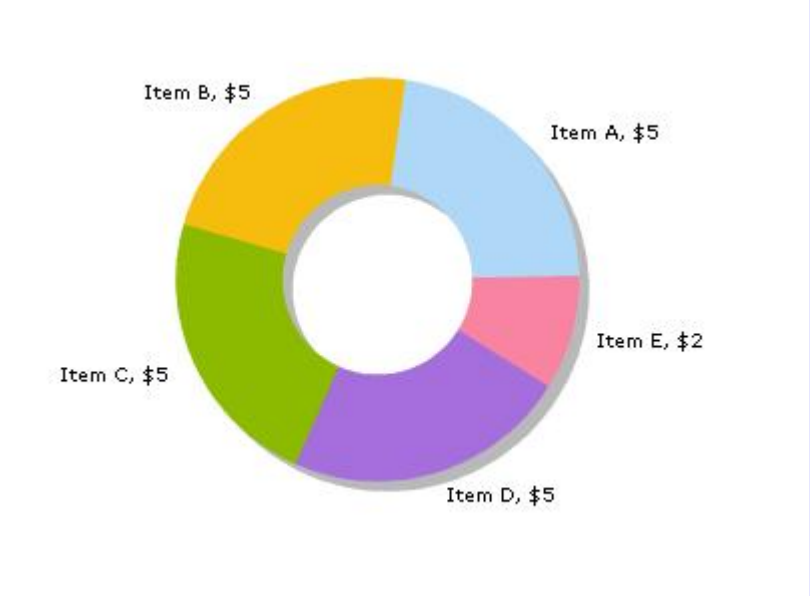
In this method, the data is contained in an external XML document (like data.xml). You can also have a server-side script generate this XML document's data (like ReturnXMLData.asp).

A sample content of an external XML document (say, data.xml):

```
<graph decimalPrecision='0' showPercentageValues='0' showNames='1'
       numberPrefix='$' showValues='1' showPercentageInLabel='0'
       pieRadius='100'>
      <set value='5' name='Item A' color='AFD8F8' />
      <set value='5' name='Item B' color='F6BD0F' />
      <set value='5' name='Item C' color='8BBA00' />
      <set value='5' name='Item D' color='A66EDD' />
      <set value='2' name='Item E' color='F984A1' />
</graph>
```

The example HTML code will be as below:

```
<html>
<head>
<title>Sample | jQuery Plugin for FusionCharts Free</title>
<script type="text/javascript" src="FusionChartsFree/JS/jquery-
1.4.2.js"></script>
<script type="text/javascript"
src="FusionChartsFree/JS/jquery.fusioncharts.js"></script>
</head>
<body><center>
  <div id="myChartContainer">Loading FusionCharts!</div>
  <script type="text/javascript">
      $('#myChartContainer').insertFusionCharts({
          type: "Doughnut2D",
          data: "data.xml",
          dataFormat: "URIData",
          width: "400",
          height: "300"
      });
  </script>
</center></body>
</html>
```



Note: If you are dealing with large amount of data, it is advisable to employ this method.

Option	Description
`data`	This option contains the URL of the external XML document. e.g. data: "data.xml"
`dataFormat`	When your **data** contains the URL of an external XML document, the **dataFormat** option should be specified as **URIData**. e.g. dataFormat: "URIData"

HTMLTable

FusionChartsFree jQuery Plugin has introduced this new feature in FusionCharts for retrieving data from HTML tables.

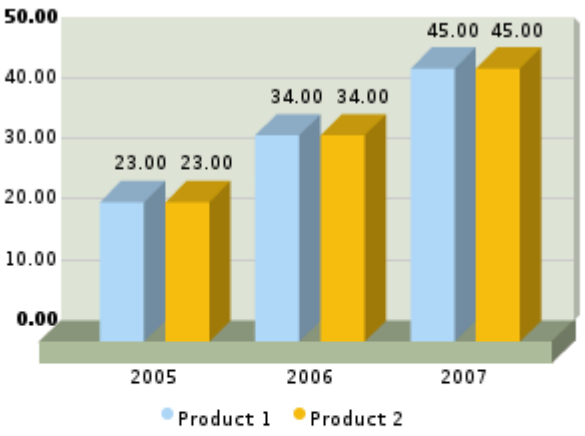
Here is a simple example:

```
<html>
<head>
<title>HTMLTable Sample | jQuery Plugin for FusionCharts Free</title>
<script type="text/javascript" src="FusionChartsFree/JS/jquery-
1.4.2.js"></script>
<script type="text/javascript"
src="FusionChartsFree/JS/jquery.fusioncharts.js"></script>
</head>
<body>
  <div id="myChartContainer" align="center">Loading FusionCharts!</div>
  <table id="myHTMLTable" border="1" align="center">
    <tr> <th> </th> <th>2005</th> <th>2006</th> <th>2007</th> </tr>
    <tr> <td>Product 1</td> <td>23</td> <td>34</td> <td>45</td> </tr>
    <tr> <td>Product 2</td> <td>23</td> <td>34</td> <td>45</td> </tr>
  </table>
  <script type="text/javascript">
  $('#myHTMLTable').convertToFusionCharts({
      type: "MSColumn3D",
      data: "#myHTMLTable",
      dataFormat: "HTMLTable"
  });
  </script>
</body>
</html>
```

This code will render a Multi-Series Column3D chart in a DIV just above the HTML table.

In the example above, the ID of HTML Table containing the data is myHTMLTable. The table is selected by prefixing a hash (#) to the Id and passed on to the data option. Thus, your data represented by a simple HTML table is transformed into a visual format with our charts simply by selecting the table.

If you run this code in your preferred browser you should see a chart like this:



	2005	2006	2007
Product 1	23	34	45
Product 2	23	34	45

Options	Description	Type/Values
`data`	This option contains the selected HTML Table. e.g. data: "#myHTMLTable"	data from HTML table
`dataFormat`	When your **data** contains the data from an HTML Table, the **dataFormat** option should be specified as **HTMLData**. e.g. dataFormat: "HTMLData"	HTMLData
`dataOptions`	dataOptions provide data-format specific options which are applicable when using HTMLData format.	All the parameters provided in the dataOptions section passed in JSON format.

DataOptions Parameters for HTMLTable Data Handler

Parameter	Description	Values	Default
`major`	Allows to group and read source data either by rows or by columns	row, col	row

Parameter	Description	Values	Default
`chartAttributes`	The valid chart attributes are passed with this parameter in JSON format. e.g. chartAttributes: {caption: "Annual Sales Summary", bgColor: "b22400"}	All the valid chart attributes	*empty*
`useLabels`	Specifies if the labels in the chart are to be fetched from a row or a column.	true, false	true
`labelSourceIndex`	Specifies the row or column index from which labels in the chart are to be fetched.	integer	1
`useLegend`	Specifies if the seriesNames for a Multi-series chart are to be fetched from a row or a column.	true, false	true
`legendSourceIndex`	Specifies the row or column index from which seriesNames for the chart are to be fetched.	integer	1
`ignoreCols`	Specifies the index of columns in the HTML Table which are to be ignored.	Array of integers	*Empty Array*
`ignoreRows`	Specifies the index of rows in the HTML Table which are to be ignored. You can specify the index of more than one row by separating them with comma.	Array of integers	*Empty Array*

Limitations of HTMLTable format

This data-format currently supports only the following chart types (aliases):

Funnel, CandleStick, Column3D, Column2D, Pie2D, Pie3D, Doughnut2D, Line2D, Bar2D, Area2D, MSColumn3D, MSColumn2D, StackedBar2D, StackedColumn3D, StackedColumn2D, Stackedarea2D, MSLine2D, MSBar2D, MSArea2D, MSCombiDY2D, MSColumn3DLineDY.

7. Additional Features

The jQuery Plugin for FusionCharts Free allows users to tweak and extend the plugin itself to achieve extreme levels of customization.

Customizing Default Plugin Options

In case, you have plugin options that you need to repeatedly use in your codes, you may extend the default plugin configuration.

The following snippet of script changes the default swfPath and className.

```
<script type="text/javascript">
$.FusionCharts.config.extend({
    swfPath: 'myPath/FusionChartsFree/Charts',
    className: 'myCharts'
});
</script>
```

Whenever the above code is executed, all subsequent charts rendered with default values will have swfPath and className as provided above.

8. Troubleshooting

When trying to make a chart using FusionCharts, if you get any errors (or if the chart doesn't render), there could a lot of reasons for it. Here, we'll try and cover them up.

While creating your chart, if for some reasons you do not see your chart like it should, check for the following actions:

A blank division is displayed where the chart is supposed to render

When viewing your page containing the chart, if a blank division is displayed, check the following:

1. The paths for the Core jQuery file and the jquery.fusioncharts.js file have been properly provided.
2. All the plugin options are provided in their correct format.

Specifically check the following:

1. The quotation marks (") enclose the plugin option values.
2. The plugin option-value pairs are separated by comma.
3. Each plugin option and its corresponding value is separated by a colon (:)
4. The last element does not have a comma after it.

SWF movie not loading or no chart shown

When viewing your page containing the chart, if you see an endless loading progress bar in your browser, or if the right click menu (right click at the place where the chart is supposed to be) shows "Movie not loaded", check the following:

1. The SWF path is properly provided in your HTML page. Also, check if the SWF file actually exists there.
2. If you're working on a case-sensitive file system Operating System, check for the case of path and SWF file.
3. If you've Adobe Flash Player 6 (or above) installed on your machine.
4. Whether you've enabled your browser to show ActiveX controls.

"Error in Loading Data" message

If you get a "Error in Loading Data" message in your chart, it means that FusionCharts could not find XML data(URIData) at the specified URL. In that case, check the following:

1. If you've actually provided an URL for the external XML file. If you do not provide an URL, FusionCharts looks for a default data.xml file in the same path. However, if that is also not found, it shows the "Error in Loading Data" error.
2. If you're using URIData method, paste this URL in your browser to check if it's returning a valid XML. Make sure there are no scripting or time-out errors and a valid XML is being returned. Also make sure that the XML isn't intermingled with HTML content. The data provider page should return clean XML only - not even HTML <head> or <body> tags.
3. If you've to pass parameters to your URIData provider page from FusionCharts, make sure they're URLEncoded in the while specifying the URIData, when providing to FusionCharts. e.g., if your URIData needs to be Data.asp?id=43&subId=454, you'll need to URL Encode it so that it becomes Data%2Easp%3Fid%3D43%26subId%3D454. Only then FusionCharts will invoke the URL with proper parameters appended to it.
4. When using URIData, make sure that the SWF File and data provider page are on the same sub-domain. Due to Flash's sandbox security model, it cannot access data from external domains, unless otherwise configured.

"Invalid XML Data" message

If you get an "Invalid XML Data" message, it means that the XML data provided is malformed. Check it again for common errors like:

1. Difference in case of tags. <graph> should end with </graph> and not </Graph> or </GRAPH>
2. Missing opening/closing quotation marks for any attributes. e.g., <graph caption=Monthly Sales' should be <graph caption='Monthly Sales'
3. Missing closing tag for any element.
4. If you're using any special characters in your XML, make sure they're properly encoded. When using XMLData, % needs to be encoded as %25, & as %26 and so on. While using URIData, you can provide most of the characters directly, without the need to encode.
5. If you are using XMLData, check for conflict of ' (XML Attribute Character) and " (HTML

Parameter Character).

6. If you've quotes as part of your data, XML Encode them to ' Example: <set name='John's House' />

"No data to display" message

If your chart shows a "No data to display" message, it could be the following scenarios:

1. Your XML data doesn't contain any data that could be plotted by FusionCharts. In this case, your XML just contains the <graph> and </graph> tags without any data between them.
2. You might be using a single-series chart SWF and providing data in multi-series format or vice-versa. In this case too, you'll get "No data to display" message.
3. In Dual Y Combination charts, you need to provide atleast one data-set for both the axis. Otherwise, you'll get a "No data to display" message.

Online Support

In case you require assistance with this product, feel free to send an email to support@fusioncharts.com, mentioning your product version and the nature of the issue.

You can also get interactive help in our online support forum at http://www.fusioncharts.com/forum/

9. API Reference

Complete tabular reference of jQuery Plugin for FusionCharts Free.

Plugin Functions

Following are the functions that are added to jQuery by this plugin. These can be called immidiately after the plugin script inclusion.

Function	Parameters	Return
`insertFusionCharts` Inserts a new FusionCharts object within selected HTML elements.	*object* options Configuration options to generate FusionCharts.	jQuery Object
`appendFusionCharts` Appends a new FusionCharts within every item of selection.	*object* options Configuration options to generate FusionCharts.	jQuery Object
`prependFusionCharts` Appends a new FusionCharts within every item of selection.	*object* options Configuration options to generate FusionCharts.	jQuery Object
`convertToFusionCharts` Converts a selected HTML table containing data to a new FusionCharts object.	*object* options Configuration options to generate FusionCharts.	jQuery Object

Plugin Options

The list of parameters that are configurable for this plugin.

Option	Descripton	Ranges/Values	Default
`id`	Defines an unique HTML Id for a chart.	Unique, alpha-numeric	*Autogenerated*
`type`	Defines the type of chart to be rendered.	Any of the chart aliases provided in the Chart Type	Column2D

Option	Descripton	Ranges/Values	Default
		Alias list.	
`width`	Sets the chart width.	numeric (pixels or percent)	320
`height`	Sets the chart height.	numeric (pixels or percent)	240
`swfPath`	Defines the document path URL where the FusionCharts SWF files reside. (excluding the file name).	URL of the SWF file (excluding the file name)	
`swfURL`	Directly defines the URL of the SWF file.	URL	undefined
`data`	Contains the string XML data or URL of an external XML document or referencedata from an HTML table.	string XML, URL of an external XML file, data from HTML table	undefined
`dataFormat`	Defines the type of data passed in the '***data***' option.	XMLData, URIData, HTMLTable	undefined
`dataOptions`	dataFormat specific options.	All the parameters provided in the dataOptions section passed in JSON format.	empty

DataOptions

The dataOptions option is used to pass data-format specific options. The parameters inside the dataOptions are passed in JSON format.

dataFormat	Parameter	Description	Values	Default
`HTMLTable`	`major`	Allows to group and read source data either by rows or by columns	row, col	row
	`chartAttributes`	The valid chart attributes are passed with this parameter in JSON format.	All the valid chart attributes	*empty*
	`useLabels`	Specifies if the labels in the chart are to be fetched from a row or a column.	true, false	true

dataFormat	Parameter	Description	Values	Default
	`labelSourceIndex`	Specifies the row or column index from which labels in the chart are to be fetched.	integer	1
	`useLegend`	Specifies if the seriesNames for a Multi-series chart are to be fetched from a row or a column.	true, false	true
	`legendSourceIndex`	Specifies the row or column index from which seriesNames for the chart are to be fetched.	integer	1
	`ignoreCols`	Specifies the index of columns in the HTML Table which are to be ignored.	Array of integers	*Empty Array*
	`ignoreRows`	Specifies the index of rows in the HTML Table which are to be ignored. You can specify the index of more than one row by separating them with comma.	Array of integers	*Empty Array*

Chart Type Aliases

The following alias names are used while specifying chart type.

Alias	vFree SWF FileName	v3 SWF FileName
Pie2D	FCF_Pie2D.swf	Pie2D.swf
Pie3D	FCF_Pie3D.swf	Pie3D.swf
Doughnut2D	FCF_Doughnut2D.swf	Doughnut2D.swf
Donut2D	FCF_Doughnut2D.swf	Doughnut2D.swf
Line2D	FCF_Line.swf	Line.swf
Column3D	FCF_Column3D.swf	Column3D.swf
Column2D	FCF_Column2D.swf	Column2D.swf
Bar2D	FCF_Bar2D.swf	Bar2D.swf

Alias	vFree SWF FileName	v3 SWF FileName
Area2D	FCF_Area2D.swf	Area2D.swf
MSColumn3D	FCF_MSColumn3D.swf	MSColumn3D.swf
MSColumn2D	FCF_MSColumn2D.swf	MSColumn2D.swf
MSBar2D	FCF_MSBar2D.swf	MSBar2D.swf
MSLine2D	FCF_MSLine.swf	MSLine.swf
MSArea2D	FCF_MSArea2D.swf	MSArea.swf
MSCombiDY2D	FCF_MSColumn2DLineDY.swf	MSCombiDY2D.swf
MSColumn3DLineDY	FCF_MSColumn3DLineDY.swf	MSColumn3DLineDY.swf
StackedBar2D	FCF_StackedBar2D.swf	StackedBar2D.swf
StackedColumn3D	FCF_StackedColumn3D.swf	StackedColumn3D.swf
StackedColumn2D	FCF_StackedColumn2D.swf	StackedColumn2D.swf
StackedArea2D	FCF_StackedArea2D.swf	StackedArea2D.swf
Funnel	FCF_Funnel.swf	Funnel.swf
Candlestick	FCF_Candlestick.swf	Candlestick.swf
Gantt	FCF_Gantt.swf	Gantt.swf
Doughnut3D	Not Available	Doughnut3D.swf
Donut3D	Not Available	Doughnut3D.swf
DragColumn2D	Not Available	DragColumn2D.swf
DragLine	Not Available	DragLine.swf
DragArea	Not Available	DragArea.swf
ErrorBar2D	Not Available	ErrorBar2D.swf
SelectScatter	Not Available	SelectScatter.swf
DragNode	Not Available	DragNode.swf
Kagi	Not Available	Kagi.swf
LogColumn2D	Not Available	LogMSColumn2D.swf
LogLine2D	Not Available	LogMSLine.swf

Alias	vFree SWF FileName	v3 SWF FileName
MultiLevelPie	Not Available	MultiLevelPie.swf
MultiAxisLine	Not Available	MultiAxisLine.swf
Radar	Not Available	Radar.swf
Spline2D	Not Available	Spline.swf
MSSpline2D	Not Available	MSSpline.swf
SplineArea2D	Not Available	SplineArea.swf
MSSplineArea2D	Not Available	MSSplineArea.swf
InverseArea2D	Not Available	InverseMSArea.swf
InverseColumn2D	Not Available	InverseMSColumn2D.swf
InverseLine2D	Not Available	InverseMSLine.swf
Waterfall	Not Available	Waterfall2D.swf
Scatter	Not Available	Scatter.swf
Bubble	Not Available	Bubble.swf
StackedBar3D	Not Available	StackedBar3D.swf
StackedColumn3DLineDY	Not Available	StackedColumn3DLineDY.swf
MSStackedColumn2D	Not Available	MSStackedColumn2D.swf
MSStackedColumn2DLineDY	Not Available	MSStackedColumn2DLineDY.swf
MSCombi2D	Not Available	MSCombi2D.swf
MSCombi3D	Not Available	MSCombi3D.swf
MSColumn3DLine	Not Available	MSColumnLine3D.swf
ScrollArea2D	Not Available	ScrollArea2D.swf
Scrollcolumn2D	Not Available	ScrollColumn2D.swf
ScrollLine2D	Not Available	ScrollLine2D.swf
ScrollCombi2D	Not Available	ScrollCombi2D.swf
ScrollCombiDY2D	Not Available	ScrollCombiDY2D.swf
ScrollStackedColumn2D	Not Available	ScrollStackedColumn2D.swf

Alias	vFree SWF FileName	v3 SWF FileName
RealtimeArea	Not Available	RealTimeArea.swf
RealtimeColumn	Not Available	RealTimeColumn.swf
RealtimeLine	Not Available	RealTimeLine.swf
RealtimeStackedArea	Not Available	RealTimeStackedArea.swf
RealtimeStackedColumn	Not Available	RealTimeStackedColumn.swf
RealtimeAngular	Not Available	AngularGauge.swf
RealtimeBulb	Not Available	Bulb.swf
RealtimeCylinder	Not Available	Cylinder.swf
RealtimeHorizontalLED	Not Available	HLED.swf
RealtimeHorizontalLinear	Not Available	HLinearGauge.swf
RealtimeThermometer	Not Available	Thermometer.swf
RealtimeVerticalLED	Not Available	VLED.swf
SparkLine	Not Available	SparkLine.swf
SparkColumn	Not Available	SparkColumn.swf
SparkWinLoss	Not Available	SparkWinLoss.swf
HorizontalBullet	Not Available	HBullet.swf
VerticalBullet	Not Available	VBullet.swf
Pyramid	Not Available	Pyramid.swf
DrawingPad	Not Available	DrawingPad.swf
ExportComponent	Not Available	FCExporter.swf