

# GJax gjs:treeColumnHeading

## From Intranet

gJax Developers Guide

### Table of contents

- 1 gjs:treeColumnHeading
  - 1.1 Synopsis
    - 1.1.1 XSL Template
    - 1.1.2 XML Document
  - 1.2 Example
    - 1.2.1 Server-side
    - 1.2.2 XSL Template
  - 1.3 Persisting column width changes
  - 1.4 Notes
  - 1.5 TODO

## gjs:treeColumnHeading

The xsl template *gjs:treeColumnHeading* renders column headings and provides functionality for sizing and sorting the data columns of a corresponding table. Full support is provided for nested tables with independently sizable and sortable columns in each nested table.

## Synopsis

### XSL Template

```
<xsl:import href='gjsTree.xsl' />
<xsl:include href='ServerLink.xsl' />

...

<xsl:template name='divMainHead'>
  <table class='layoutTable' width='1000px' gjsTreeId=' TreeId '>
    <tr>
      <xsl:call-template name='gjs:treeMoresMainHeading' />

      <xsl:call-template name='gjs:treeColumnHeading'>
        <xsl:with-param name='width'> 999 </xsl:with-param>
        <xsl:with-param name='caption'> Caption </xsl:with-param>
        [ <xsl:with-param name='columnId'> ColumnId </xsl:with-param> ]
        [ <xsl:with-param name='selected'> ascending|descending </xsl:with-param> ]
        [ <xsl:with-param name='divMainHeadId'> divMainHeadId </xsl:with-param> ]
        [ <xsl:with-param name='divReferenceHeadingId'> divReferenceHeadingId </xsl:with-param> ]
        [ <xsl:with-param name='previousColumnResizable'> 0|1 </xsl:with-param> ]
      </xsl:call-template>

      ...

      <td width='*'></td>
    </tr>
  </table>
</xsl:template>
```

The data cells in the leaf content template should have a left padding of 5px to align the content under the column heading caption rather than under the column's resize handle. The *treeCell* class should be used to achieve this:

```
<xsl:template name='leafContent'>

  <td class='treeCell'>
    <xsl:value-of select='@column1data' />
  </td>

  <td class='treeCell'>
    <xsl:value-of select='@column2data' />
  </td>

  ...

</xsl:template>
```

## XML Document

Use the following server-side call to generate the xml dataset containing persisted column widths for the current logged in user:

```
d xml^%gjsTree(indent,%usr,"tableId","tree|columnWidths|@*")
```

This will insert the following xml into your document:

```
<tree>
  <columnWidths
    columnId1='999'
    columnId2='999'
    ...
  />
</tree>
```

## Example

### Server-side

```
d xml^%gjsTree(indent,%usr,"Category_List","tree|columnWidths|@*")
```

### XSL Template

```

<xsl:import href='gjsTree.xml' />
<xsl:include href='ServerLink.xml' />

<xsl:template name='divMainHead'>
  <table class='layoutTable' width='1000px' gjsTreeId='Category_List'>
    <tr>
      <xsl:call-template name='gjs:treeMoresMainHeading' />

      <td width='25px'><td>

        <xsl:call-template name='gjs:treeColumnHeading'>
          <xsl:with-param name='width' select='"200"' />
          <xsl:with-param name='columnId'>Category</xsl:with-param>
          <xsl:with-param name='caption'>Category</xsl:with-param>
          <xsl:with-param name='selected'>ascending</xsl:with-param>
          <xsl:with-param name='previousColumnResizable'>0</xsl:with-param>
        </xsl:call-template>

        <xsl:call-template name='gjs:treeColumnHeading'>
          <xsl:with-param name='width' select='"100"' />
          <xsl:with-param name='columnId'>Icon</xsl:with-param>
          <xsl:with-param name='caption'>Icon</xsl:with-param>
        </xsl:call-template>

      <td width='*'></td>
    </tr>
  </table>
</xsl:template>

```

## Persisting column width changes

All columns are automatically resizable, but if you want to make the width changes persist from one user session to another there are three things that you must do:

1. Give the table a unique **treeId**. It is recommended that this is the name of the xml document without the extension (eg Customer\_List).
2. Give each column a unique **columnId** (within treeId).
3. On the server-side call **xml^%vcTree** to insert the persisted column width settings into the xml document.

If you do not give a column a *columnId* then changes to that column's width will not be persisted.

## Notes

- gjsTree.xml contains the templates that implement column heading functionality. You must *import* this file into your xml file in order to be able to use it.
- You will also need to *include* ServerLink.xml if you want column width changes to be persisted.
- The *columnId* should be unique within a table. It **must** be a valid XML attribute name. Two columns with the same Id will interfere with each other's column widths.
- The width does not need to be specified on the main data table's columns, they will be set automatically when tree is loaded.
- The selected parameter indicates which (if any) column is pre-sorted and in which direction.
- The parameter *divMainHead* does not normally need to be supplied. It is used, when the tree control is not being using in the main panel, to specify the id of the div that contains the column heading.
- The parameter *divReferenceHeadingId* does not normally need to be supplied. It is used when the reference copy of the column heading is not created as part of the current document (where it defaults to runId\_mainHead). It would be appropriate to use it when appending to an existing table created in a previous document.
- The parameter *previousColumnResizable* controls the behaviour of the resize handle immediately to the left of this column heading. By default this handle allows the previous column to be resized. It should be set to 0 for the first column to prevent the user from changing the width of the column containing the row's icon.

## TODO

- Maybe the control should be wrapped inside a <gjs:treeColumnHeading> control?
- The use of the columnId as an attribute name in the <columnWidths> element is naff.

Retrieved from "[http://10.101.68.18/intranet/index.php/GJax\\_gjs:treeColumnHeading](http://10.101.68.18/intranet/index.php/GJax_gjs:treeColumnHeading)"

---

- This page was last modified 12:27, 24 Sep 2005.