

Trees & Tables — widgets for tabular and tree-like data

Tables (/widgets/trees-tables/)

Trees (/widgets/trees-tables/trees)

Tree Tables (/widgets/trees-tables/tree-tables)

MVC Table Views (/widgets/trees-tables/mvc-table-views)

MVC Tree Views (/widgets/trees-tables/mvc-tree-views)

MVC Item models (/widgets/trees-tables/mvc-item-models)

MVC Table Views

A `WTableView` (/www.webtoolkit.eu/wt/doc/reference/html/classWt_1_1WTableView.html) is a widget that renders each cell of a table to an `Item Delegate` (`WItemDelegate` (/www.webtoolkit.eu/wt/doc/reference/html/classWt_1_1WAbstractItemModel.html)). This is a variation of each individual item (i.e. table cell) to an **Item Delegate** (`WItemDelegate`) widget, allowing this aspect to be customized without needing to deal with the rendering of the entire table.

A more direct approach to showing a table of data is to use a `WTable` (/www.webtoolkit.eu/wt/doc/reference/html/classWt_1_1WTable.html) widget, which renders each cell. Compared to this, a `WTableView` supports virtual scrolling (for large tables) and allows you to scroll through many columns of additional data while keeping the first header row visible.

Example

Age	Rural Male
70-74	660
65-69	410
60-64	1269
55-59	181
50-54	117

source

```
#include <Wt/WApplication.h>
#include <Wt/WStandardItemModel.h>
#include <Wt/WTableView.h>
#include "../treeview-dragdrop/CsvUtil.h"

auto tableView = cpp14::make_unique<WTableView>();
tableView->setModel(csvToModel(WApplication::appRoot() + "table.csv"));

tableView->setColumnResizeEnabled(false);
tableView->setColumnAlignment(0, AlignmentFlag::Center);
tableView->setHeaderAlignment(0, AlignmentFlag::Center);
tableView->setAlternatingRowColors(true);
```