

## ExpressQuantumGrid – Demos

This document provides an overview of all the demos available for this product.

The list below contains 45 demos. This includes 3 **new** and 7 **updated** demos in version 6.0 and 6.1:

- CardLayoutDemo (new)
- ChartDataDrillingDemo (new)
- FilterDemo (new)
- ViewChartDemo
- MasterDetailMultiDemo
- MasterDetailChartDemo
- SummaryGroupDemo
- SummaryMultiDemo
- ColumnsSimpleDemo
- Office11GroupRowStyleDemo

The table below shows all the demos in the order that we suggest you look at them. They are grouped by category and listed in order starting from simple to complex. The GridDemos project group is provided for your convenience. It contains all the demos in the same order as in the following list.

	Demo	Description
1	ViewTableSimpleDemo	Categories are displayed on the left showing the grid as a single column list. This controls the records displayed on the right in table mode.
2	ViewBandedDemo	Provides options showing all aspects of using bands and column customization at runtime and the simple code required for this.
3	ViewBandedFixedDemo	Displays timesheets grouped by project. User information is in a fixed left band and summary/total details are fixed on the right.
4	ViewNestedBandsDemo	Shows the same bands arranged in regular and nested layouts. Allows columns or bands to be quickly shown/hidden using a built-in feature. Demonstrates cell merging for cells containing the same data.
5	ViewCardSimpleDemo	Departments are displayed on the left showing the grid as a single column list. This controls User records displayed as Cards on the right.
6	ViewCardDemo	Allows cards to be expanded/collapsed and rows with blank values to be hidden. Data can be filtered using the filter box or on a per row basis using each row's filter popup. Rows can be rearranged using drag&drop. The Rows Customization form can be used to show/hide rows.
7	CardLayoutDemo	The same data is shown using different layout features of the CardView including expandable categories, horizontal/vertical row layout inside a card, horizontal/vertical card layout inside a CardView, card auto width.

<b>8</b>	ViewChartDemo	Shows how to use different diagrams in unbound and data-aware chart views. Allows you to customize chart, diagram and series settings. Applies styles with background bitmaps and custom draw to enhance the chart view's appearance. Shows the data used to build charts in the table.
<b>9</b>	ChartDataDrillingDemo	Demonstrates the data drilling capabilities of the ChartView. Allows you to customize a chart's appearance including series and data group visibility. Shows a horizontal scrollbar when not all chart values fit within the view.
<b>10</b>	MasterDetailTableDemo	Displays two tables (as were needed by grid3) and a single menu click converts into a single table displaying master/detail grid views.
<b>11</b>	MasterDetailCardDemo	Starts by displaying a grid with a single master view and separate edit controls for detail records. A single menu click converts the grid into master/detail with the detail displaying in card view.
<b>12</b>	MasterDetailDemo	Displays master/detail levels in grid views.
<b>13</b>	MasterDetailMultiDemo	Similar to the MasterDetailDemo but uses the same view to show different data at different levels. Allows you to show/hide tabs for details with no data and customize their appearance.
<b>14</b>	MasterDetailChartDemo	Displays a chart view as the detail of a master table view. Allows you to scroll data of the first chart to see all its values.
<b>15</b>	UnboundSimpleDemo	Displays information about the solar system. Data is loaded from a simple CSV file.
<b>16</b>	UnboundListDemo	The grid uses Provider Mode to work with the data contained in a plain text file.
<b>17</b>	UnboundExternalDataDemo	Loads an INI file into the grid. Data is displayed in master/detail views by section. Provider Mode is used to handle data.
<b>18</b>	UnboundDesignDefinedDemo	Shows the unbound TableView populated with data at design-time.
<b>19</b>	UnboundModeDemo	MineSweeper! This demo shows the ability of the grid to work with all kinds of unstructured data.
<b>20</b>	UnboundColumnsDemo	Shows how to use unbound columns in a DB TableView.
<b>21</b>	GridModeDemo	Shows how to set up fast mode (GridMode) to work with a large set of data.
<b>22</b>	SummaryFooterDemo	Shows how to provide runtime customization of group summary values. The user can right click on a group footer and change the summary type.
<b>23</b>	SummaryGroupDemo	Displays group summary information within the grouped row and on the footer after its detail records. Provides the ability to change group row summary data layout and show/hide group footers at different levels.
<b>24</b>	SummaryMultiDemo	This demo shows all the main features available in the ExpressQuantumGrid for customizing group and footer summaries including sorting by summary and multiple footer summaries per column.

<b>25</b>	FilterDemo	Demonstrates the column filter popup with the ability to select multiple values and show only filtered items. Filter row allows you to easily enter values for filter conditions. You can also see special filters for the DateTime data.
<b>26</b>	FilterByCodeDemo	Shows how to build complex filters in code.
<b>27</b>	FilterControlDemo	Shows how to set up and use the FilterControl (data & non-data aware) components for filtering data in a dataset or directly in the grid.
<b>28</b>	ColumnsSimpleDemo	Master/Detail using table views and showing various column editors. Custom hints are provided for data cells of one column. You can try the table view's Quick Customization popup to show/hide/move columns.
<b>29</b>	EditorsInPlaceDemo	Experiment with various in-place column editors. They are displayed using three tabbed card views.
<b>30</b>	EditorsLookupDemo	Practice using lookup data-aware editors for reading and writing. Shows the use as stand-alone editors and bound to a grid column.
<b>31</b>	EditorsMaskDemo	Practice using masked data-aware editors for reading and writing. Use the various modes including regular expressions with incremental completion.
<b>32</b>	ColumnsMultiEditorsDemo	Uses provider mode to fill an array and also changes the editor assigned to a column on the fly via an event handler.
<b>33</b>	ColumnsShareDemo	Shows the ability to share an edit repository item between columns of different views.
<b>34</b>	StylesMultiDemo	Practice using stylesheets. Switch between predefined and user defined styles. Accesses the StyleSheet Designer dialog at runtime.
<b>35</b>	StylesCardViewDemo	Displays records displayed in cards within four tabs: DevExpress, Slate, High Contrast and User Defined. Editing facilities are provided for the User Defined style.
<b>36</b>	StylesSimpleDemo	Edit and assign styles to various grid elements.
<b>37</b>	CellLevelMultiSelectDemo	Demonstrates how the cell level multiselect functionality can be used to emulate a spreadsheet's functionality. The indicator's custom draw event is used to draw the row numbers.
<b>38</b>	CustomDrawTableViewDemo	Displays a table view with a custom draw editor enabling custom effects on various grid elements.
<b>39</b>	CustomDrawCardViewDemo	Displays a filtered card view with various background effects via custom draw.
<b>40</b>	Office11GroupRowStyleDemo	Demonstrates functionality similar to that found in MS Outlook 2003: group row style and grouping by sorting. Groups can be kept always expanded using the datacontroller's option. Try different kinds of grouping against DateTime data.
<b>41</b>	DragDropDemo	Demonstrates how to use drag & drop functionality.

<b>42</b>	CustomRowHeightDemo	Shows how to use custom heights for each row to show pictures of different sizes in their cells.
<b>43</b>	GridMenuViewsDemo	Use context sensitive popup menus for headers, detail records, group summaries and footer summaries.
<b>44</b>	IssueList	An example of using several grid controls to manage a popular type of application. A chance to see the grid in action and to apply your own touches to a useful utility application.
<b>45</b>	EditorsStylesDemo	Shows a selection of editors and controls so that you can see how they all work together using a selection of styles, look&feel and hint types.