

The QueryDataGrid control loads any type of data from the Kentico database and displays it in a customizable table.

- Retrieves data using a pre-defined database query assigned through the **QueryName** property.
- Automatically binds the data. You do not need to write any code.
- You can use the Visual Studio DataGrid designer to set up the appearance and behavior of the grid.

Queries allow you to display the following types of data:

- **Pages** (Page types -> Edit page type -> Queries)
- **Custom table records** (Custom tables -> Edit table -> Queries)
- **Other objects** (Modules -> Edit module -> Classes -> Edit class -> Queries)



**Inherits from:** [BasicDataGrid](#)

**Web part equivalent (portal engine):** Grid with custom query



**Tip:** If you only need to display page data from a website's content tree, consider using the [CMSDataGrid](#) control instead.

## Getting started

The following tutorial shows how to use the QueryDataGrid control to display a table containing news items (CMS.News pages) from the sample Corporate Site:

1. Create a new **Web form** in your web project.
2. Switch to the **Design** tab and drag a **QueryDataGrid** control from the toolbox onto the form.
3. Set the control's **QueryName** property to: *cms.news.selectall*
  - This assigns the query that the control uses to load news pages.
4. Right-click the CMSDataGrid on the form, select **AutoFormat...** and choose a scheme.
5. Right-click the CMSDataGrid again, select **Show Smart Tag** and then **Property Builder**.
  - The **QueryDataGrid1 Properties** dialog opens.
6. On the **General** tab, check **Allow sorting**.
7. Switch to the **Columns** tab and uncheck **Create columns automatically at run time**.
8. Add a new **Bound Column** from the **Available columns** list to the **Selected columns** list. Enter the following values into the appropriate fields:
  - **Header text:** News Title
  - **Data Field:** NewsTitle
  - **Sort expression:** NewsTitle
9. Add another **Bound column**. Enter the following values in the appropriate fields:
  - **Header text:** Release Date
  - **Data Field:** NewsReleaseDate
  - **Sort expression:** NewsReleaseDate
10. Click **OK**.
11. Save the web form.
12. Right-click the web form in the Solution explorer and select **View in Browser**.

The resulting page displays a table similar to the following (depending on the chosen scheme):

| News Title ▲                        | Release Date          |
|-------------------------------------|-----------------------|
| Apple iPad 2 In Stock               | 2/6/2014 12:00:00 AM  |
| Community Website Section           | 3/28/2014 12:00:00 AM |
| Company Growth Exceeds Expectations | 6/17/2014 12:00:00 AM |
| New Consulting Services             | 6/5/2014 12:00:00 AM  |

## Configuration

You can set the following properties for the QueryDataGrid control:

| QueryDataGrid properties    | Description  | Sample value                   |
|-----------------------------|--|--------------------------------|
| DataBindByDefault           | Indicates whether the control automatically performs data binding during the <b>Init</b> event. (Inherited from <a href="#">BasicDataGrid</a> )  |                                |
| HideControlForZeroRows      | Indicates whether the control should be hidden when no data is loaded. The default value is False. (Inherited from <a href="#">BasicDataGrid</a> )                                     |                                |
| PageSize                    | Sets the number of rows displayed per page.  |                                |
| ProcessSorting              | Indicates whether the control processes the grid sorting in the DataView instead of on the SQL level. (Inherited from <a href="#">BasicDataGrid</a> )                                  |                                |
| QueryName                   | Name of the query that the control uses to load data.<br><br>Enter the full query name in format: <class code name>.<query name>   | "CMS.MenuItem.selectdocuments" |
| QueryParameters             | Gets or sets an array containing parameters for the used query.  |                                |
| RelatedData                 | Custom data connected to the object. (Inherited from <a href="#">BasicDataGrid</a> )   |                                |
| SetFirstPageAfterSortChange | Indicates if the page index resets to the first page when sorting is changed. (Inherited from <a href="#">BasicDataGrid</a> )  |                                |
| SortAscending               | Indicates whether the grid sorting is in ascending order. The default value is True. (Inherited from <a href="#">BasicDataGrid</a> )   |                                |
| SortField                   | Gets or sets the sort field. Allows you to set the default sort field. (Inherited from <a href="#">BasicDataGrid</a> )   | "NewsReleaseDate"              |
| ZeroRowsText                | Text shown if no records are found. This text is not visible when the control is hidden by the <b>HideControlForZeroRows</b> property. (Inherited from <a href="#">BasicDataGrid</a> ) | "No records found"             |

| CMS Base control properties | Description | Sample value |
|-----------------------------|-------------|--------------|
|-----------------------------|-------------|--------------|

|                   |   |   |
|-------------------|---|---|
| CacheDependencies | <p>List of the cache keys on which the control's cached data depends. When the specified cache items change, the control clears its cache.</p> <p>Each item (dependency) must be on one line.</p> <p>If you leave this property empty, the control uses default dependencies.</p> <p><b>See also:</b> <a href="#">Setting cache dependencies</a>, <a href="#">Configuring caching</a></p>   | cms.<br>user all  |
| CacheItemName     | <p>Sets the name of the cache key used to store the control's content. If you leave the value empty, the system generates a default name containing variables, such as the control ID, the selected culture and the name of the user who loaded the page.</p> <p>The system cache is shared by all pages in your application, so cache item names representing different data must be unique globally. If you have multiple controls that load the same data, you can share the cache keys between the controls (optimizes loading of content and avoids redundant data in the cache).</p> <p>If the content displayed by the control depends on variables, such as URL parameters, you can set a custom name dynamically in the page's code behind.</p> <p><b>See also:</b> <a href="#">Caching the data of page components</a>, <a href="#">Configuring caching</a></p> | "CMSRep<br>eaterNew<br>s" +<br>Request.<br>QueryStri<br>ng["id"].<br>ToString() |
| CacheMinutes      | <p>Sets the number of minutes for which the control caches content retrieved from the database.</p> <ul style="list-style-type: none"> <li>0 indicates that control does not cache content</li> <li>-1 indicates that the control uses the site-level content caching settings</li> </ul> <p>Allows you to set up caching of content so that the control doesn't have to retrieve content from the database on each request.</p> <p>The caching mechanism uses absolute expiration time. This means that cache items expire after a specified time period even if the page containing the control wasn't requested.</p> <p><b>See also:</b> <a href="#">Caching the data of page components</a>, <a href="#">Configuring caching</a></p>  |   |
| FilterControl     | Gets or sets the filter control used to limit the data read by the control.   |   |
| FilterName        | Gets or sets the code name of the filter control used to limit the data read by this control.   |   |
| OrderBy           | Gets or sets the ORDER BY clause of the SQL query that the control uses to load data.   | "NewsRel<br>easeDate<br>DESC"   |
| SelectedColumns   | Database table columns that the control loads for pages, separated by commas ( , ). If null or empty, the control loads all available columns.  |   |
| SiteName          | Specifies the code name of the Kentico website for which the control loads data.  |   |
| StopProcessing    | If true, the control stops all processing — does not load or display any data or other HTML output.   |   |
| TopN              | Specifies the maximum number of database records that the control loads.  |   |
| WhereCondition    | Gets or sets the WHERE clause of the SQL query that the control uses to loads data.   | "Product<br>Price ><br>100"   |



**Note:** The QueryDataGrid is derived from the ASP.NET DataGrid control, so you can also set any of the [base properties](#).

## Appearance and styling

You can modify the appearance of the QueryDataGrid control by setting the standard properties available for the [ASP.NET DataGrid](#) control (inherited through the BasicDataGrid).

A common way to set the appearance of DataGrid controls is to assign a skin through the **SkinID** property. You can define skins in *.skin* files under individual themes in the **App\_Themes** folder. See the [.NET Skins and Themes](#) documentation for more information.