

The BasicRepeater control displays items from a data source in a list based on specified templates.

You can use the BasicRepeater with any bindable data source – not only Kentico data and objects.

i **Derived from:** [ASP.NET Repeater](#)
Web part equivalent (portal engine): Basic repeater (see [Loading and displaying data on websites](#) for an example of use)

✓ **Tip:** If you want to display pages from Kentico, you can use the [CMSRepeater](#) control, which has built-in support for loading Kentico pages.

Getting started

The following is a step-by-step tutorial that shows how to use the BasicRepeater to display a list of laptops (*CMS.Laptop* pages) from the sample Corporate Site:

1. Create a new **Web form** somewhere in your web project.
2. Drag the **BasicRepeater** control from the toolbox onto the form.
3. Add the following code marked by the **BasicRepeater templates** comments between the `<cms:BasicRepeater>` tags. The overall code of the BasicRepeater control should look like this:

```
<cms:BasicRepeater ID="BasicRepeater1" runat="server">

    <!-- BasicRepeater template
    ----- --%>

    <itemtemplate>
        <h3>
            <%# Eval("DocumentSKUName") %>
        </h3>
        " alt="<%# Eval("DocumentSKUName") %>" />
    </itemtemplate>

    <!-- BasicRepeater template
    ----- --%>

</cms:BasicRepeater>
```

i This defines the template used by the BasicRepeater to display items. The control uses inline code to insert the values of items into the template. This process is repeated for all records in the data source.

4. Switch to the web form's code behind and add the following references:

```
using System.Data;

using CMS.DocumentEngine;
```

5. Add the following code to the **Page_Load** method:

```
// Retrieves all CMS.Laptop pages from the Kentico database as a DataSet
DataSet ds = DocumentHelper.GetDocuments("CMS.Laptop").Path("/", PathTypeEnum.
Children).OrderBy("DocumentSKUName");

// Assigns the DataSet as the data source of the BasicRepeater control
BasicRepeater1.DataSource = ds;
BasicRepeater1.DataBind();
```

6. Save the changes to the web form and its code behind file.
7. Right-click the web form in the Solution explorer and select **View in Browser**.

The resulting page displays a list similar to the following:



Configuration

You can set the following properties for the BasicRepeater control:

Property name	Description	Sample value
DataBindByDefault	Indicates whether the control automatically performs data binding during the Init event.	
HideControlForZeroRows	Indicates whether the control should be hidden when no data is loaded. The default value is False.	
PagerDataItem	Gets or sets the pager data item object.	
PagerForceNumberOfResults	If set, the pager does not modify the DataSet containing paged items, but the pager itself behaves as if the amount of paged items were identical to this value. The value must be set to -1 for the property to be disabled.	
RelatedData	Custom data connected to the object.	
ZeroRowsText	Text shown if no records are found. This text is not visible when the control is hidden by the HideControlForZeroRows property.	"No records found."



Note: The BasicRepeater inherits from the ASP.NET Repeater control, so you can also set any of the [base properties](#).

Appearance and styling

You can modify the appearance of the BasicRepeater control by setting the standard properties inherited from the ASP.NET Repeater control.

The design of the listed items is determined by the code of the templates defined within the tags of the BasicRepeater control. The following templates are available:

Template name	Description
AlternatingItemTemplate	Template applied to alternating items.
FooterTemplate	Template used for the footer of the list.
HeaderTemplate	Template used for the header of the list.
ItemTemplate	Template applied to standard items.
SeparatorTemplate	Template used for separating the listed items.