The system provides a set of web parts that allow you to display data from custom tables on the pages of your website.

Custom table web parts	Description
Custom table data source	A separate component that loads data stored in custom tables. To display the data, you need to connect the data source to a **Basic repeater**, **Basic datalist** or **Basic universal viewer** web part.
Custom table datagrid	Loads data from a custom table, and displays selected columns in a table. Contains a built-in custom table data source. You can modify the design of the web part using standard ASP.NET skins.
Custom table repeater Custom table datalist Custom table repeater with effect	Load data from a custom table, and display the records in a format based on transformations. Contain a built-in custom table data source. See Writing transformations for custom tables for more information and examples.
Custom table form	Allows users to manage the data of a custom table on the live site (add new data records or edit existing ones).

Displaying lists of items from custom tables

1. Open the **Pages** application.
2. Select the page where you want to display the data.
3. Switch to the **Design** tab.
4. Add one of the web parts that loads custom table data. The **Web part properties** dialog opens.
5. Choose the **Custom table** that you want to display.
6. Assign a transformation that defines the output format of individual custom table records (skip this step for the **Custom table datagrid** web part).

   > **Note**: If you are using a separate **Custom table data source**, you need to set the transformation for the connected basic listing web part.

7. Click **OK**.

The web part loads the data from the custom table, and displays the information on the page. The appearance of the data depends on the type of the web part and the assigned transformation.

Displaying details for selected custom table items

In addition to creating lists of data, you can use the custom table web parts to display detailed views of individual custom table records. Detail views are useful if the records in your table contain more information than you can display in a list. The web parts identify selected custom table items using a query string URL parameter.

The following steps describe how to extend a list of custom table data to allow users to select and view individual records:

1. Open the **Pages** application.
2. Select the page containing your custom table list.
3. **Configure** (double-click) the web part that loads the custom table data.
4. Enter a name for the selection URL parameter into the **Selected item query string key name** property (for example *item id*).
5. Type the *Field name* of the custom table's identifier field (primary key) into the **Selected item database column name** property.

6. Set the **Selected item validation type** according to the data type of the table's identifier field (*number*, *text*, or *GUID*).
7. Edit the custom table's list transformation and add a detail link into the content. The link's target is the same page, with the addition of the selection query string parameter. Load the parameter's value from the table's identifier field. For example:

```
<a href='?itemid=<%# Eval("TableID") %>'>View details</a>
```

8. Create and assign a **Selected item transformation** that defines the content of the detail view for custom table items.

When a user views the custom table list and clicks the detail link for one of the items:

- The page updates, with the selection parameter added to the URL's query string.
- The web part (data source) reads the parameter's value and loads the custom table record with a matching identifier.
- The web part (listing) displays the content of the *Selected item transformation* for the given item (instead of the list).

Tip

You can use URL wildcards to convert the query string parameter in the selection URL into a more user friendly format. For example:

```
<domain>/CustomTableList.aspx?itemID=10
```

to:

```
<domain>/CustomTableList/10.aspx
```