Sorting by Column Links

This Solution Recipe shows you how to add sorting capabilities to column links in your database-driven tables. With these steps you'll be able to enable your users to sort the data displayed on results pages however they choose.

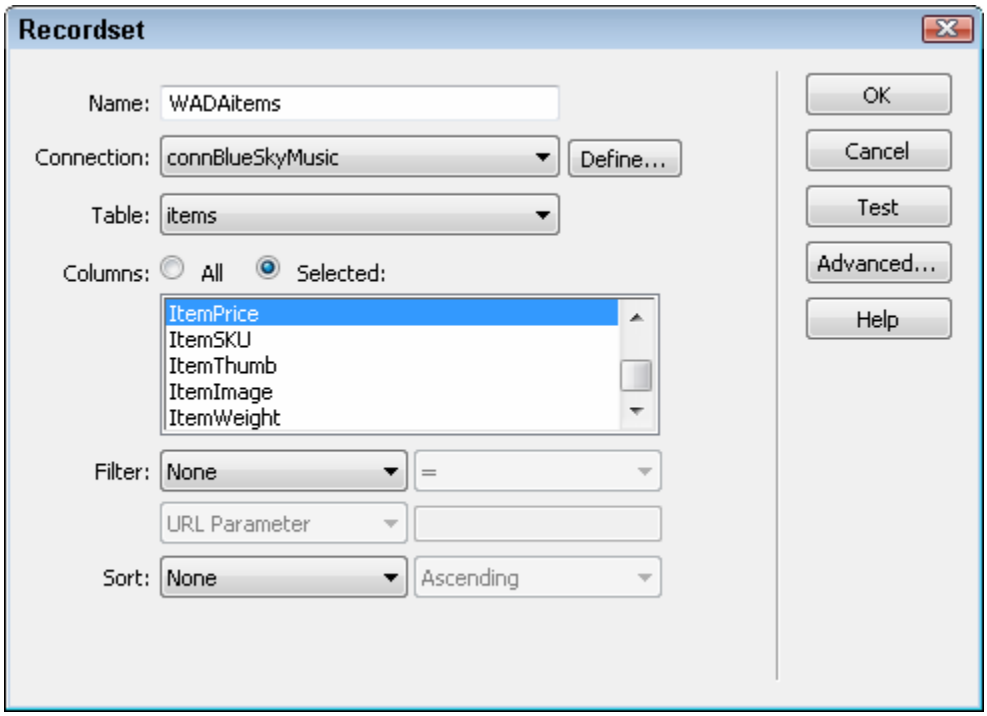
What Do You Need to Start?

1. Dreamweaver CS3 or 8.
2. DataAssist 2.0 or higher.
3. Dreamweaver dynamic site (ASP-JS, ASP-VB, ColdFusion or PHP).
4. Connection to supported database.
5. A recordset for the table to be updated.
6. A table of database-driven content or 2 or more columns of data.

To Set Up Your Page for Sorting

Before you can apply the DataAssist Sort server behavior, you have to prepare your page.

1. Remove any sorting previously applied to the recordset. If you're recordset is viewable in the Simple mode of the Recordset dialog box, choose **None** from the **Sort** list; if your recordset is displayed in the Advanced mode, remove any ORDER BY clause.



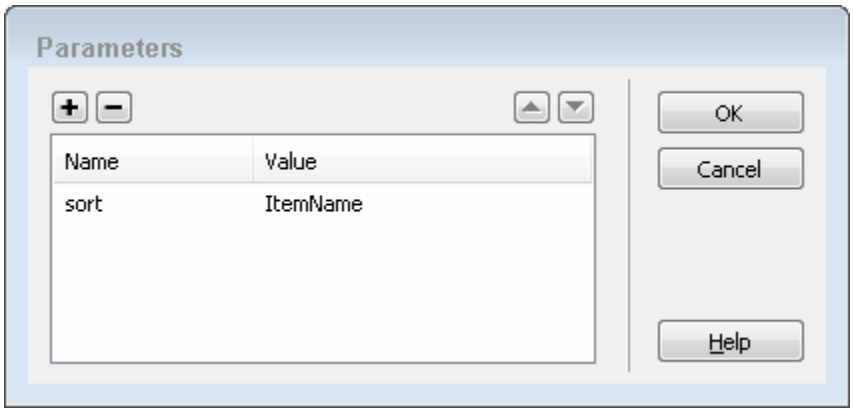
My example uses a simple recordset, so I just need to choose None from the Sort list.

2. Select the text of the first column and, from the Property inspector, click the **Link folder** icon.

   To trigger the sort, you'll need to add a link to the current page for each column label with a parameter variable and value.

3. When the Select File dialog box opens, select the current file; in this example, it's the **item_Results** page.

4. Click **Parameters**. When the Parameters dialog box opens, enter **sort** in the Name column and, in the Value column, the database column name. For the first link in my example, the entered value is **ItemName**. Click OK once to close the Parameters dialog box and again to close the Select File dialog box.



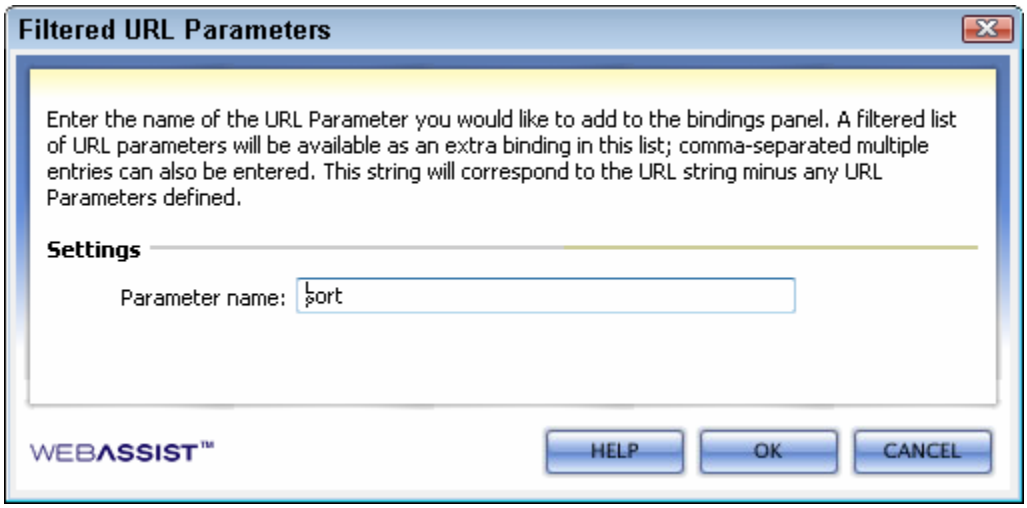
5. Repeat steps 2-4 for each link. Be sure to use the same Name (**sort**) and the proper database column for each link. In my example, I'll set links for the **ItemShortDesc** and **ItemPrice** data columns.

   There's one more prepatory step to make the sort behavior easy to apply: adding a QueryString binding.

6. From the Bindings panel, choose **Add** (+) and select **QueryString Binding**.

   The QueryString Binding option makes the parameter available for triggering the sort and also allows you to filter URL parameters for more complex sorting operations.

7. In the Filtered URL Parameters dialog box opens, enter the parameter value; for our example, that value is **sort**. Click **OK**.

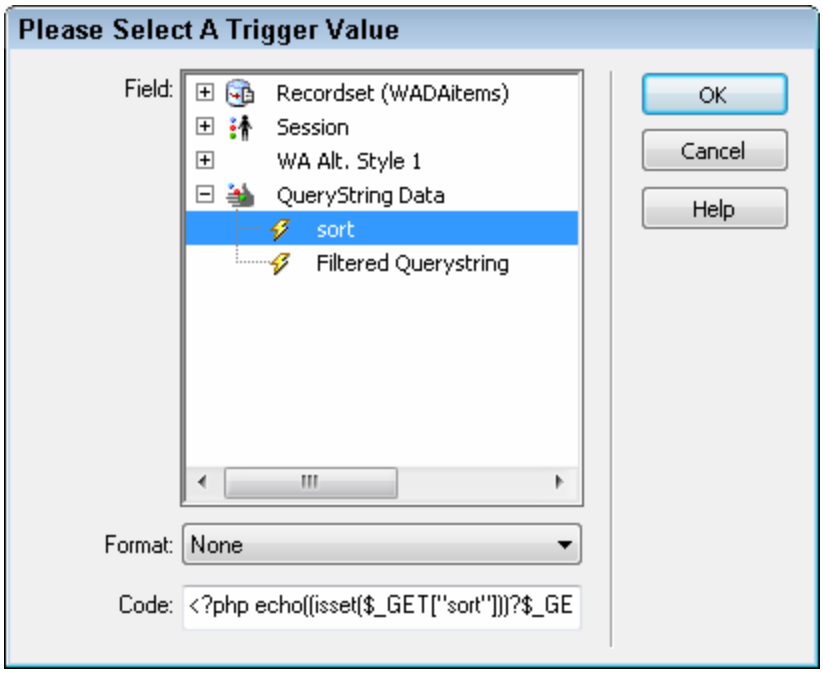


8. Save your page.

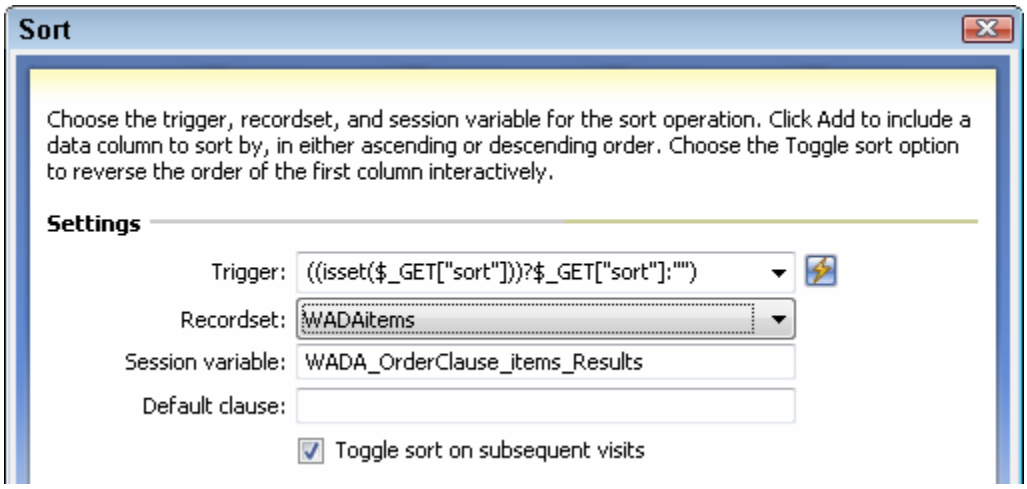
To Add the Sort Server Behavior

Now that the page is properly set up, let's apply the Sort server behavior.

1. From the Server Behaviors panel, click **Add** (+) and choose **DataAssist > Sort**.
2. When the Sort dialog box opens, you'll first select the event to execute the server behavior from the **Trigger** list. To assign the link parameter as the trigger, click the Trigger lightning bolt icon. When the dialog box opens, expand the QueryString Data entry and choose your parameter value; here I'll choose **sort**. Click **OK**.



3. From the **Recordset** list, choose the desired recordset. Here, the choice is **WADAitems**.
4. Set the Session variable to store the sort criteria; DataAssist creates the proper session variable for you, so you can leave the default entry.
5. If you have a default sort criteria, enter it in the Default clause field. For this example, let's leave it blank.
6. If you want to toggle the sort between ascending and descending, click the **Toggle sort on subsequent visits** option.



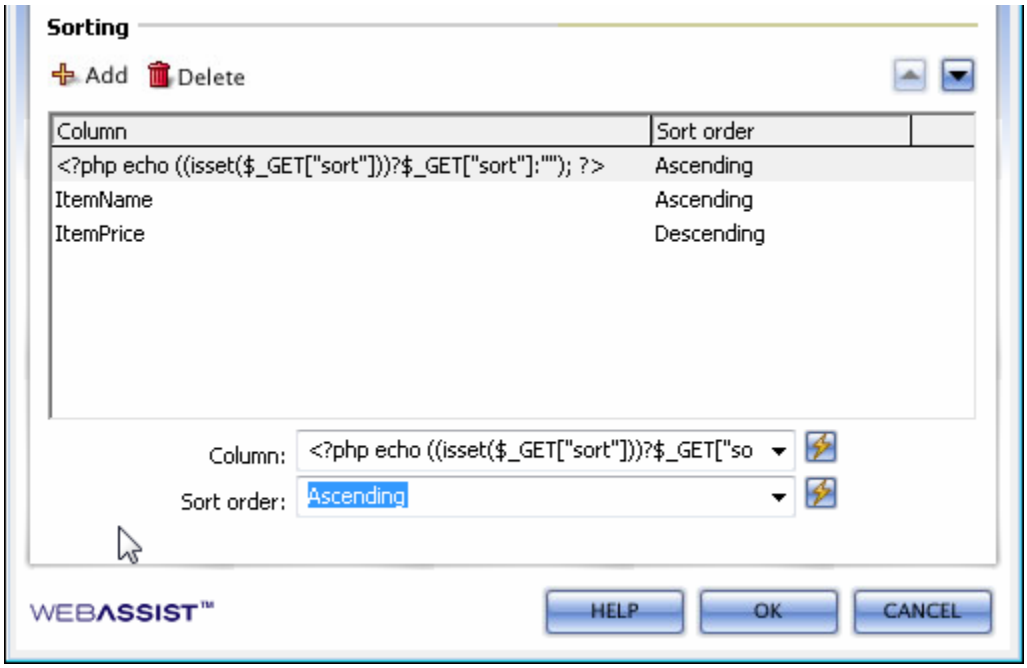
Next, let's add the sort order. The primary sorting again uses the sort variable stored in the QueryString Binding.

7. Click **Add** (+) and click the **Column lightning bolt** icon. When the dialog box opens, select the variable under the QueryString Binding entry; in this example, the variable is **sort**. Click OK.
8. Choose the Sort order from the available options; here, I'll choose **Ascending**.

   Optionally, you can now add additional sort criteria. Let's add two to see how it works.

9. Choose Add (+) and from the Column list, choose the next column to sort by; here, let's choose ItemName and set the Sort order to Ascending.
10. Now, let's add our final criteria by choosing Add and selecting ItemPrice from the Column list and set the sort order to Descending, so the most expensive product will be displayed first.

    These additional sort criteria take effect whenever the sorting is triggered. For example, if a user clicks the Name sort column and two or more items have the same name, they will be listed by the highest price first; if the list is sorted by Price, items with the same price will be listed alphabetically.



11. When you're done, click OK.

12. Save your page and press F12 to try out the page on your Testing server.

Now you're application is ready to be sorted by clicking the column header. DataAssist also gives you the option to sort your data by selecting an entry from a drop-down list.