Formally we introduced DataTable as a readymade solution from jspx to interact with DB and provide a widely common task that is to search DB and apply filters on the search results. Some might say that interacting with DB directly is not preferred way specially from the presentation layer. Other application architecture might impose the use of well defined Service layer as the only way to interact with DB. This Service layer can be a DAO based model, EJB or webservice call. So the return of such interface will be a list of objects of strong type. Presenting such data is not doable using DataTable. So jspx provides a new control ListTable.

1. Overview

Jspx ListTable is very much similar to DataTable but has minor difference. So it is recommended to refer to the DataTable documentation. ListTable is exposing its functionality through set of attributes, that is listed below.

2. Attributes

ListTable is non standard HTML tag, some of its attributes are standard and some are not. The following table shows set of these attributes along with their description.

Name	Type	Description
id	String	The id of the control, if you going to interact with it from java code, you will need to declare a property with the same name.
rendered	Boolean	Whether this control is going to be processed or not.
showNew	Boolean	Indicates whether a link command is added to the footer of the table to allow entering new items or not. The default value is true.
items	String	The name of a property in the page controller, this property should be of type List<?>. This is the list of objects that will be rendered in the ListTable. This property is accessed through the method setList(List<?>) .
saveInSession	Boolean	ListTable can afford the ability to save its items in user session in order to preserve the time required to fetch them. The default value is true.
onInsert	String	The name of the user defined method in the controller that will be invoked when the list Table is inserting a new item. Like DataTable, ListTable provides the ability to insert new item. But unlike DataTable, the new item is added to the List of items in the memory, and not preserved anywhere. This event handler call back feature allows the developer to intercept the event after the new item is created and before saving it. So the developer can beform any kind of action on this item. Like validation for example. The signature of the handler method should be : `public void event(Object insertedObject)`

bindToClass	String	The fully qualified name for class used to convert an item in to fully object oriented instance. This attribute is required in ListTable as it defines the type of the object that will be rendered.
pageSize	int	The number of rows to be displayed per page, the default value is 20
pageindex	int	Property to set and get the index of the currently viewed page.
onEdit	String	The same as the onInsert event name, but it is fired when the ListTable is in edit mode.
onDelete	String	The same as the onInsert but it is fired when an item is being deleted.
showRowIndex	Boolean	It is possible to display a sequence of number to mark the rows displayed. The default value is false.
startRowIndex	int	The row index displayed – based on the previous attribute- is a sequence that starts from the value specified in this attribute. The default value is 0
autobind	Boolean	ListTable can be configured to run automatically or manually. The default value is false.
noresults	String	In case the SQL statement along with applied filters, there might be no returned results. Developer might want to show a message indicating that there are no returned results.
noResultsClass	String	The name of CSS class that is applied on the previous message.
dateformat	String	If there is Data of Type date to be displayed, developer can specify the format of the date through this attribute.
rowStyle/rowClass	String	The style / CSS class applied on the table even rows. This will distinguish even rows from odd.
tdClass	String	The CSS class applied on each TD in the table.
selectRowClass	String	The CSS class applied on the row that is currently under edit
selectedRowStyle	String	The Style applied on the row that is currently under edit.
var	String	Arbitrary name used to declare a variable. This variable represents the currently rendered row. The only use of this variable while rendering item template within a data column.

It is important to note that when developer intercepts the edit, insert or delete. The developer is responsible for completing the operation itself. So if he is intercepting insert, he must add the passed object in the collection. The same while editing and deleting.

1. Child controls

Unlike DataTable, ListTable has less set of child controls. This is due to the special nature of DataTable. ListTable has no direct interaction with DB, so the Child DB controls like DataParam and DataPK do not exist.

1.1.   Data Column

To control the columns to be displayed in the table, jspx provides a control named DataColumn to choose which rows to be rendered.

The order of the DataColumn control within DataTable is the same order from left to right where these columns are displayed.

```
<datacolumn text="User Name" fieldname="USER_NAME" type="string"
sortable="true" />
```
The above example shows a data column used to display the Column USER_NAME from the DB Table USERS.

The following is a list of attributes in this control

Name	Type	Description
text	String	The end user friendly name that will be displayed in the header of the column

fieldname	String	The name of the property in the Object that is being rendered.
		If this ListTable is displaying Users, and one User is like
		public class User
		{
		   int id;
		   public void setId(int id)
		   {
		      this.id=id;
		   }
		   public int getId()
		   {
		      return id;
		   }
		}
		Then the field name should be [id].

type	enum	The type of Data Column which controls the way it is displayed, also the way it is edited. 1. String 2. Number 3. Date 4. Lookup
lookup	String	In case the type of data column is a lookup, this means that whatever data in this column is not the final format that will be displayed to end user. For example, the USERS table has the column user_role which is a number that is foreign key to another Table ROLE. In this case it is not friendly at all to display the user role to the end user as number. Which is meaningless. So, as it is required to display the friendly name of the role obtained from the other table, this attribute points to Lookup control to fetch the desired value.
required	Boolean	While the Data Table in Edit/New mode, this column may be required and cannot accept empty values.

		This attribute tell where this column is required or not. The default value is false.
readonly	Boolean	Tells whether this field is cannot be edited while the DataTable is in Edit mode. The default value is false.
dateFormat	String	In case this control is of type Date, the date formatter can be specified in this attribute

1.1.1. ItemTemplate

Data Columns can also display custom HTML instead of static Text. This feature is supported through the control ItemTemplate. Item Template is a child control for the DataColumn, where custom HTML can be used to display whatever data developer chooses to show.

The following example shows the use of item template to display the user name as a hyperlink to the user profile page instead of just a static text.

```
<datacolumn text="User Name" fieldname="name"  type="string" sortable="true">
      <itemtemplate>
            <a href="/pages/user/userDetails.jspx?id=${user.id}" style="text-decoration: underline;" target="_blank">${user.name}</a>
      </itemtemplate>
</datacolumn>
```

From above, we notice the presence of any HTML/jspx controls within the ItemTemplate tag. The real challenge is how insert data obtained from object instance in this template. In our case we want the user name column in each row to have a link, but this link requires two pieces information
1. User Name
2. User ID

This information is obtained from the currently rendered row. In order to do that, developer must declare a value in the Attribute var in the ListTable (described above), then uses this var in an EL format to obtain the required data.

For example.
```
 ${user.id}    ${user.name}
```
Where user is the name of the var declared in the ListTable Header. While id and name are the name of two properties in the rendered object.

1.2.   Data Column Command

Just like DataTable, ListTable exposes set of operation versus the collection that are rendered. These action may include (Select, Edit, Delete). Jspx ListTable is providing such

functionality through the use of the control DataColumnCommand. The following is an example on the use of this control.

```
<datacolumnCommand text="remove" confirm="are you sure to delete this
item?" type="delete"/>
```

The following attributes are available on this control.

Name	Type	Description
text	String	The end user friendly text that will be displayed as the name of this link command
type	enum	The type of Data Column Command which controls the way it is behaving. | Field name	Description
---	---
Select	Clicking on this command will fire the event specified in the action attribute
Delete	Will remove the current row from the Items of this ListTable
Edit	Will fire the edit mode of the ListTable where the current row will be displayed for editing in the Details panel.
Check	Will display a check box to allow multiple selection on table rows.	
confirm	String	In case the type of data column command is Delete, developer may want to display a confirmation message before executing the action. This attribute specifies the message.
action	String	In case the type of data column command is Select, the developer will need to handle the selection event with an event handler. This attribute tells the name of the event handler method in the page controller. This method should be in the signature. `public void rowSelected(WebControl sender, String args)` Where rowSelected is the value set in the attribute action in the HTML page. It is to be noted that the args parameter will carry the index of the row that is selected.
onClick	String	The name of the client side JavaScript method that should be executed before posting back the form to the server.
postNormal	Boolean	When using ajax, sometimes it is required to override the presence of a jspx ajax panel and force normal posting, this attribute controls this feature. The default value is false.
checkType	enum	The Type of selection applied on this table. When using a data column command of type check, the header field is showing a select all check box. This check box can be selecting all results or only all visible rows. 1. all

		2. visible The method getSelectedRows() from ListTable is using this attribute to return either all rows or only the visible.

1.3. Footer

ListTable as well as DataTable needs a footer layer where the action controls are displayed, including
(Insert new item, navigation, export to excel sheet).

```
<footer class="jspxTableFooter" message="Total search results {0} displaying
page ({1}/{2})" firstcommandtext="first>>"
    lastcommandtext="last <<" prevcommandtext="previous <"
    nextcommandtext="next>" newcommandtext="add new user>" />
```
The following attributes are available on this control.

Name	Type	Description
class	String	The CSS class name applied on the footer.

message	String	The Navigation message that is displayed.
		This message can be parameterized to display 3 items
		    1. Total result count {0}
		    2. The current page index {1}
		    3. The total count of pages. {2}
		The default value is `Total search count={0}, Page({1}/{2})`

firstCommandText	String	The text to be displayed for navigation link to the first page. Default value is <<first
nextCommandText	String	The text to be displayed for navigation link to the next page. Default value is next >
prevCommandText	String	The text to be displayed for navigation link to the previous page. Default value is <prev
lastCommandText	String	The text to be displayed for navigation link to the last page. Default value is last>>
newCommandText	String	The text to be displayed for the link that shows the new item panel The default value is new
exportToExcelText	String	The text to be displayed for the link that is exporting the rows to MS Excel Sheet. The default value is Export to Excel

The above were the essentials features of one of the most important controls in jspx, more features are always added.