

## *DataTable*

---

Jsp is providing easy to use and readymade solutions for most of the common web tasks developers face daily. Most frequent task is to provide a search interface for the end user to find certain information in data storage like DB. Jsp provides a solution for this task through the jsp control named DataTable.

This control exposes set of powerful tools like searching, filtering, sorting, selecting, custom rendering, edit, delete and exporting the results in MS excel sheet.

### 1. Overview

Jsp is originally meant to connect to Oracle DB and MySQL, however small efforts can help supporting the same functionality on different DB like SQL Server.

Although jsp is using mostly standard HTML tags, to provide extra functionality, it is required to use nonstandard tags. HTML does not provide such a utility to search DB; hence jsp has created new tag DataTable. Jsp is mainly focusing on Declarative code rather than Imperative. For that, all the exposed functionality of the control is controlled from both HTML code and Java code. DataTable attributes are exposing all of characteristic of DataTable.

### 2. Attributes

DataTable 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
<b>id</b>	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.
<b>rendered</b>	Boolean	Whether this control is going to be processed or not.
<b>showNew</b>	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.
<b>sql</b>	String	The SQL statement that is search the DB
<b>dataSource</b>	String	The JNDI name used to fetch the Application Server data source for connection pool
<b>table</b>	String	The name of the DB table. It is required only when the developer chooses to provide the feature of deleting.
<b>bindToClass</b>	String	The fully qualified name for class used to convert an item in to fully object oriented instance.
<b>pageSize</b>	int	The number of rows to be displayed per page, the default value is 20
<b>pageIndex</b>	int	Property to set and get the index of the currently viewed page.
<b>navigationMode</b>	enum	DataTable provides out of the box paging feature. This attribute has one of the following values. 1. Paging 2. Through

		<p>Paging means that DataTable will calculate the total count of results returned by the provided SQL. So the total number of results is displayed along with the number of pages.</p> <p>Through mode is used when the query is heavy, and then calculating the total count of results will double the execution time. Using this mode will provide the paging feature but will not calculate the total count.</p>
<b>showRowIndex</b>	Boolean	It is possible to display a sequence of number to mark the rows displayed. The default value is false.
<b>startRowIndex</b>	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
<b>autobind</b>	Boolean	DataTable can be configured to run automatically or manually. The default value is false.
<b>noresults</b>	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.
<b>noResultsClass</b>	String	The name of CSS class that is applied on the previous message.
<b>dateformat</b>	String	If there is Data of Type date to be displayed, developer can specify the format of the date through this attribute.
<b>rowStyle/rowClass</b>	String	The style / CSS class applied on the table even rows. This will distinguish even rows from odd.
<b>tdClass</b>	String	The CSS class applied on each TD in the table.
<b>selectRowClass</b>	String	The CSS class applied on the row that is currently under edit
<b>selectedRowStyle</b>	String	The Style applied on the row that is currently under edit.
<b>var</b>	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.
<b>caseSensitive</b>	Boolean	<p>Compatibility flag for MySql and Oracle DB.</p> <p>If True→ The Expressions used in the DataParam will not be converted to lower case. This implies that the Expression and Variable name is the same case. For example</p> <pre>&lt;dataparam name="userName" expression ="user name LIKE '%userName%' " control="unTextBox" trim="true" /&gt;</pre> <p>If the attribute <b>name</b> is not the same as in the expression, jsp will not find the variable to replace, and will fail to execute the query. A warning will be thrown in logs.</p> <p>Default value is false</p>

### 3. Child controls

The above attributes are used to generally control the behavior of the DataTable. In order to control the displayed columns and applying filtering, the child controls are used to control the extra features.

#### 3.1. DataParam Controls

Most of the common tasks when providing searching feature are about filtering form. Where end user filters and limits the search criteria to narrow the search results. DataParam controls are used to provide this functionality.

It is imported to initially prepare the SQL statement to be parameterized. So later on in the data binding phase, these parameters are replaced with the collected values out of the filtering form.

Let's consider the following use case.

Your web site provides a page to search over the subscribed users. A simple filtering form is used to filter the search results as the following:

user name	<input type="text"/>
age	<input type="text"/>
salary	<input type="text"/>
	<input type="button" value="search"/>

Assuming there is a DB Table named Users, the SQL statement will be

```
SELECT * FROM users WHERE user_name LIKE '%userName%' AND salary = userSalary  
AND age = userAge
```

Where (userName, userSalary, userAge) are parameters will be replaced with values collected from different resource.

Of course when end user enters a data in one the fields, the SQL should be updated. But if the end user did not enter anything the whole results should be displayed.

In order to do this requirement jsp provides a control named DataParam.

Data Param is used to collect information from different resources to replace a parameter in the SQL.

For example, DataParam can be

```
<dataparam name="userName" expression ="user_name LIKE '%userName%' "  
control="unTextBox" trim="true" />
```

The following attributes are used to control the filtering operation.

Name	Type	Description
<b>Id/name</b>	String	The id of the control, it is the same name of the parameter specified in the SQL. The SQL containing this name will be replaced with the value.
<b>expression</b>	String	Filtering expression like user_name LIKE '%userName%' is specified in order to replace it 1=1 when end user doesn't enter data.
<b>control</b>	String	The id of the form control that will be used to extract the data from. So whatever entered in this control will be replacing the parameter in the SQL statement.
<b>request</b>	String	The value of the parameter can be extracted from Http Servlet Request. This attribute specifies the name of the request attribute that is containing the required value.
<b>session</b>	String	The value of the parameter can be extracted from Http Servlet Session. This attribute specifies the name of the session attribute that is containing the required value.
<b>trim</b>	Boolean	Sometimes user enters data with leading and trailing white spaces; this attribute is used to trim the value. The default value for attribute is false.

The final SQL statement that is used to search DB considering the end user choose to search users with name like 'amr' and age equals 30 without a specific salary .

```
SELECT * FROM users WHERE user_name LIKE '%amr%' AND 1 = 1 AND age = 30
```

It has to be noted that the replacing of the place holders inside an Expression with the final values is subjected to the attribute **casesensitive** in the DataTable.

### 3.2. Data PK

Almost all DB Tables have one or more Primary Key. These special fields are used to uniquely identify a row. Jsp provides a way to declare primary keys and use them while inserting, editing and deleting operation.

PK are either manually inserted or automatically generated. Automatic generation varies based on the type of DB. In Oracle, they can be generated by Trigger or Sequence. On the other hand they can be identity like in MS Access and MS SQL Server. So no value is specified while inserting new records. Also they can be generated as a result of SQL statement. In some cases, PK can be a combination of set of fields. So the one of these PK can be always a fixed value, while the rest of the PKs are dynamically generated.

PKs are important in case of Deleting or editing rows, because the update/delete statement is executed on the field using PK. Otherwise the action will be applied on all rows.

Example on DataPK that can be used on our case

```
<datapk name="id" />
```

Where id is the PK column name in the DB Table users.

The following table describes the different features supported by DataPK.

Name	Type	Description
<b>Id/name</b>	String	The id of the control, it is the same name of the DB Table PK filed
<b>rendered</b>	Boolean	Controls whether this PK will be rendered while insert/edit mode or not.
<b>type</b>	enum	The type of PK which controls the way it is generated and used. 1. Sequence 2. Trigger 3. SQL 4. Identity.  1. The Value is obtained from SeqName.nextVal expression. 2. The Value is inserted dummy and the trigger in DB will alter it right before entering. 3. The Value is obtained from the return of an SQL statement. 4. No value is entered, MS DB will automatically insert it.
<b>sql</b>	String	If the PK Type attribute is set to SQL, then the Statement should be specified in this attribute, in order to be used while inserting.
<b>sequence</b>	String	If the PK Type attribute is set to sequence, then the sequence name should be set in this attribute, so it is used as SquenceName.nextVal .
<b>defaultValue</b>	String	Sometimes Developer want to set a fixed default value to this PK, as the table has PK as a combination of set of fields. The value specified here will be used.
<b>dataType</b>	enum	The type of Data of this PK. 1. String 2. Number 3. Date

### 3.3. Data Column

To control the columns to be displayed in the table, jsp 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
<b>text</b>	String	The end user friendly name that will be displayed in the header of the column
<b>fieldname</b>	String	The name of the DB Table Column name, this name is used to fetch the returned data from resultset for the desired column.

<b>type</b>	enum	The type of Data Column which controls the way it is displayed, also the way it is edited. <ol style="list-style-type: none"> <li>String</li> <li>Number</li> <li>Date</li> <li>Lookup</li> </ol>
<b>lookup</b>	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.
<b>requires</b>	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.
<b>readonly</b>	Boolean	Tells whether this field is cannot be edited while the DataTable is in Edit mode. The default value is false.
<b>sortable</b>	Boolean	Tells whether this filed is providing a sorting on the table or not. if this value is set to true, the header of the table will be rendered as link rather than a text. Clicking this link will bi-directionally sort the table by the selected column. The default value is false.
<b>sortDir</b>	enum	Sorting on a column can be initially set to a certain direction. <ol style="list-style-type: none"> <li>asc</li> <li>desc</li> </ol> The default value is desc.
<b>dateFormat</b>	String	In case this control is of type Date, the date formatter can be specified in this attribute

### 3.3.1.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="USER_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/jsp controls within the ItemTemplate tag. The real challenge is how insert data obtained from DB 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 DataTable (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 DataTable Header. And ID, NAME are the name of the columns returned from the SQL statement.

### 3.4. Data Column Command

Commonly a table displaying a result coming from DB may provide a set of defined actions on the rendered rows. These action may include (Select, Edit, Delete). Jsp DataTable 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										
<b>text</b>	String	The end user friendly text that will be displayed as the name of this link command										
<b>type</b>	enum	The type of Data Column Command which controls the way it is behaving. <table border="1" data-bbox="528 1507 1385 1809"> <thead> <tr> <th>Field name</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td><b>Select</b></td> <td>Clicking on this command will fire the event specified in the action attribute</td> </tr> <tr> <td><b>Delete</b></td> <td>Will remove the current row from DB</td> </tr> <tr> <td><b>Edit</b></td> <td>Will fire the edit mode of the DataTable where the current row will be displayed for editing in the Details panel.</td> </tr> <tr> <td><b>Check</b></td> <td>Will display a check box to allow multiple selection on table rows.</td> </tr> </tbody> </table>	Field name	Description	<b>Select</b>	Clicking on this command will fire the event specified in the action attribute	<b>Delete</b>	Will remove the current row from DB	<b>Edit</b>	Will fire the edit mode of the DataTable where the current row will be displayed for editing in the Details panel.	<b>Check</b>	Will display a check box to allow multiple selection on table rows.
Field name	Description											
<b>Select</b>	Clicking on this command will fire the event specified in the action attribute											
<b>Delete</b>	Will remove the current row from DB											
<b>Edit</b>	Will fire the edit mode of the DataTable where the current row will be displayed for editing in the Details panel.											
<b>Check</b>	Will display a check box to allow multiple selection on table rows.											
<b>confirm</b>	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.										
<b>action</b>	String	In case the type of data column command is Select, the developer will need										

		<p>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.  <b>public void</b> rowSelected(<b>WebControl</b> sender, <b>String</b> args)</p> <p>Where rowSelected is the value set in the attribute action in the HTML page.</p> <p>It is to be noted that the args parameter will carry the index of the row that is selected.</p>
<b>onClick</b>	String	The name of the client side JavaScript method that should be executed before posting back the form to the server.
<b>postNormal</b>	Boolean	When using ajax, sometimes it is required to override the presence of a jsp ajax panel and force normal posting, this attribute controls this feature. The default value is false.
<b>checkType</b>	enum	<p>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.</p> <ol style="list-style-type: none"> <li>1. all</li> <li>2. visible</li> </ol> <p>The method getSelectedRows() from DataTable is using this attribute to return either all rows or only the visible.</p>

### 3.5. Footer

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

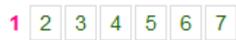
```
<footer class="jspxTableFooter" message="Total search results {0} displaying
page ({1}/{2})" firstcommandtext="first">>
  lastcommandtext="last <<" prevcommandtext="previous <"
  nextcommandtext="next">" newcommandtext="add new user">" />
```

The following attributes are available on this control.

Name	Type	Description
<b>class</b>	String	The CSS class name applied on the footer.
<b>message</b>	String	<p>The Navigation message that is displayed.  This message can be parameterized to display 3 items</p> <ol style="list-style-type: none"> <li>1. Total result count {0}</li> <li>2. The current page index {1}</li> <li>3. The total count of pages. {2}</li> </ol> <p>The default value is Total search count={0},</p>

		Page ({1} / {2})
<b>firstCommandText</b>	String	The text to be displayed for navigation link to the first page. Default value is <<first
<b>nextCommandText</b>	String	The text to be displayed for navigation link to the next page. Default value is next >
<b>prevCommandText</b>	String	The text to be displayed for navigation link to the previous page. Default value is <prev
<b>lastCommandText</b>	String	The text to be displayed for navigation link to the last page. Default value is last>>
<b>newCommandText</b>	String	The text to be displayed for the link that shows the new item panel The default value is new
<b>exportToExcelText</b>	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"
<b>hidePaging</b>	Boolean	True → Hides the Flickr like pagination in the footer Default value is False

Here is the Flickr like pagination



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