

NJX:FIELDLIST

The NJX:FIELDLIST control is used to arrange fields or groups of fields in a horizontal line. The difference of using the NJX:FIELDLIST control instead of individual fields is that the NJX:FIELDLIST control binds the contained fields to an array or array group in the application, while individual fields are bound to individual variables.

The following topics are covered below:

- Example
- Adapter Interface
- Built-in Events
- Properties

Example

The screenshot displays two examples of the NJX:FIELDLIST control. The first, 'Complex Field List', is a grid of 5 columns and 3 rows of input fields. The second, 'Simple Field List', is a single row of 10 input fields.

Complex Field List				
11100102	11100105	11100106	11100107	11100108
Schindler	Schirm	Schmitt	Schmidt	Schneider
Edgar	Christian	Reiner	Helga	Wolfgang

Simple Field List									
Schindl	Schirm	Schmitt	Schmidt	Schneid	Schneid	Bunger	Thiele	Thoma	Treiber

The XML code for the example looks as follows:

```
<rowarea name="Complex Field List">
  <itr>
    <njx:fieldlist fieldlistprop="columns" fieldcount="5"
      hdist="60">
      <njx:fielditem valueprop="id" width="80"
        invisiblemode="cleared">
      </njx:fielditem>
    </njx:fieldlist>
  </itr>
  <itr>
    <njx:fieldlist fieldlistprop="columns" fieldcount="5"
      hdist="10">
      <njx:fielditem valueprop="last" width="130"
        invisiblemode="invisible">
      </njx:fielditem>
    </njx:fieldlist>
  </itr>
  <itr>
    <njx:fieldlist fieldlistprop="columns" fieldcount="5"
      hdist="40">
      <njx:fielditem valueprop="first" width="100"
        invisiblemode="invisible">
      </njx:fielditem>
    </njx:fieldlist>
  </itr>
</rowarea>
```

```

</rowarea>
<rowarea name="Simple Field List">
  <itr>
    <njx:fieldlist fieldlistprop="simple" fieldcount="10">
      <njx:fieldvalue width="50">
        </njx:fieldvalue>
      </njx:fieldlist>
    </itr>
  </rowarea>

```

Adapter Interface

```

DEFINE DATA PARAMETER
1 COLUMNS (1:*)
2 FIRST (A) DYNAMIC
2 ID (A) DYNAMIC
2 LAST (A) DYNAMIC
2 STATUS (A) DYNAMIC
1 SIMPLE (A/1:*) DYNAMIC
END-DEFINE

```

For all NJX:FIELDLIST controls that are bound to the same value in `fieldlistprop` (here: `columns`), one common structure array is generated (here: `COLUMNS`).

For each NJX:FIELDITEM control, an element in the structure is generated according to the value bound in `valueprop` (here: `FIRST`, `ID` and `LAST`).

For each occurrence of the structure array, a parameter with the fixed name `STATUS` is generated. This parameter can be used to control the status of the elements in a similar way as it is done with the `statusprop` of the `FIELD` control.

For a simple field list (one that contains an NJX:FIELDVALUE control), a simple array is generated according to the value bound in `valueprop` (here: `SIMPLE`).

Built-in Events

The fields in the NJX:FIELDLIST control (NJX:FIELDITEM controls or NJX:FIELDVALUE controls) behave like `FIELD` controls.

Properties

Basic		
fieldlistprop	Name of the adapter parameter that represents the control in the adapter.	Obligatory
fieldcount	Maximum count of fields in the fieldlist. If no fieldcount is defined then a default of 10 is assigned.	Optional
hdist	Horizontal distance between the fields Can be specified either in pixels or as percentage value. If no width is defined then a default width of 2 pixels is assigned.	Optional
njx:natname	If a Natural variable with a name not valid for Application Designer (for instance #FIELD1) shall be bound to the control, a different name (for instance HFIELD1) can be bound instead. If the original name (in this case #FIELD1) is then specified in this attribute, the original name is generated into the parameter data area of the Natural adapter and a mapping between the two names is generated into the PROCESS PAGE statement of the Natural adapter.	Optional
njx:natcomment	The value of this attribute is generated as comment line into the parameter data area of the Natural adapter, before the field name. The Map Converter, for instance, uses this attributes to indicate for a generated statusprop variable to which field the statusprop belongs.	Optional