SLIDER

The SLIDER control represents a slider. The main use of the slider is to limit the user input to specific values. It uses a number representation for its values, but the numbers can also be used to express string values.

The following topics are covered below:

- Example

- Properties

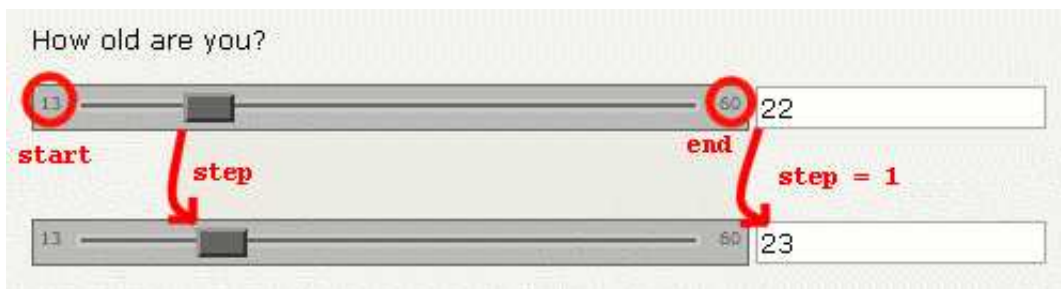## Example

The XML layout definition is:

```
<rowarea name="Number Output">
    <itr>
        <slider valueprop="slider1" from="13" to="60" showrange="true"
                                        showcurrentvalue="false">
        </slider>
    </itr>
</rowarea>
```

The control can be customized by setting its start value, end value and a step. The start and end values form a closed interval. The step defines the distance between two valid values represented by the slider in this interval.

In the above example, the value for the step is the default value "1". The possible values represented by the slider are the integers from "13" to "60". It is possible to specify a floating-point number as a step, for example "0,25". The slider can be further customized by setting the properties showrange and showcurrentvalue which show the range (start and end value) and the current value of the slider while the user is moving it. The width and height of the slider point is adjustable. The slider point is the element which the user drags and drops. The colors, the borders of the slider, the point, the line, the range and the current value can also be customized.

The Java code of the adapter is:

```
package com.softwareag.cis.test22;
// This class is a generated one.

import com.softwareag.cis.server.*;
import com.softwareag.cis.server.util.*;

public class SliderAdapter
    extends Adapter
{

    public class NumberSLIDERInfo extends SLIDERInfo
    {
        public void trigger()
        {
            m_fieldNumber = this.giveFormattedSliderValueAsInt();
        }
    }

    public class StringSLIDERInfo extends SLIDERInfo
    {
        public void trigger()
        {
                int inpValue = this.giveFormattedSliderValueAsInt();
                if(inpValue == 1) m_fieldString = "very bad";
                else if(inpValue == 2) m_fieldString = "bad";
                else if(inpValue == 3) m_fieldString = "ok";
                else if(inpValue == 4) m_fieldString = "good";
                else if(inpValue == 5) m_fieldString = "very good";
        }
    }

    // property >fieldNumber<
    int m_fieldNumber;
    public int getFieldNumber() { return m_fieldNumber; }
    public void setFieldNumber(int value)
    {
        m_fieldNumber = value;
    }

    // property >slider1<
    NumberSLIDERInfo m_slider1=new NumberSLIDERInfo();
    public NumberSLIDERInfo getSlider1() { return m_slider1; }
    public void setSlider1(NumberSLIDERInfo value) { m_slider1 = value; }

    // property >fieldString<
    String m_fieldString;
    public String getFieldString() { return m_fieldString; }
    public void setFieldString(String value) { m_fieldString = value; }
```

```
    // property >slider2<
    StringSLIDERInfo m_slider2=new StringSLIDERInfo();
    public StringSLIDERInfo getSlider2() { return m_slider2; }
    public void setSlider2(StringSLIDERInfo value) { m_slider2 = value; }

    // property >slider3<
    SLIDERInfo m_slider3=new SLIDERInfo();
    public SLIDERInfo getSlider3() { return m_slider3; }
    public void setSlider3(SLIDERInfo value) { m_slider3 = value; }

    /** initialisation - called when creating this instance*/
    public void init()
    {
        m_slider1.setSliderValue(18);
        m_slider2.setSliderValue(3);
    }

    /** */
    public void onEnterNumber()
    {
        m_slider1.setSliderValue(m_fieldNumber);
        m_fieldNumber = m_slider1.giveFormattedSliderValueAsInt();
    }

}
```

Every slider is bound to a `SLIDERInfo` object. Calling the `setSliderValue()` method sets the value of the slider which automatically moves it. However, the slider is not set to every given value, but to one valid value which fits in the given interval. The valid value to which the slider is currently set can be obtained by calling the `giveFormattedSliderValueAsInt()` or `giveFormattedSliderValueAsFloat()` method. When the user drops the slider, the method `trigger()` is called. This can be used to define specific behavior when the value in the user interface is changed. As in the example above, the `SLIDERInfo` can be extended, and the `trigger()` method can be overwritten to define specific behavior. You can show the new value in another control or use it for other purposes.

Properties

Basic			
valueprop	Server side property representation of the control.	Obligatory	
Appearance			

width	Width of the slider. Can be given in pixels or percentage.	Optional	100
			120
			140
			160
			180
			200
			50%
			100%
displayonly	If set to true, the SLIDER will not be accessible for input. It is just used as an output.	Optional	true
			false
showrange	Boolean value. Whether to show the range of the slider. The range is the "from" and "to" values.	Optional	true
			false
showcurrentvalue	Boolean value. Whether to show the current value of the slider while it is moving.	Optional	true
			false
mainbgcolor	Background color of the slider container. This should be a valid CSS color value. For example a name(blue, red), a hexadecimal value(#99CCFF) or others.	Optional	#FF0000
			#00FF00
			#0000FF
			#FFFFFF
			#808080
			#000000
mainbordercolor	Border color of the slider container. This should be a valid CSS border-color value. You can specify a different color for the top, right, bottom and left border in this sequence. For example: #BBBBBB #666666 #666666 #BBBBBB	Optional	#bbb #666 #666 #bbb #BFCFFF #00248F #00248F #BFCFFF

mainborderwidth	Border width of the slider container.	Optional	thin
			medium
			thick
			1px
			2px
			5px
			10px
pointbgcolor	Background color of the slider point. This should be a valid CSS color value. For example a name(blue, red), a hexadecimal value(#99CCFF) or others.	Optional	#FF0000
			#00FF00
			#0000FF
			#FFFFFF
			#808080
			#000000
pointbordercolor	Border color of the slider point. This should be a valid CSS border-color value. You can specify a different color for the top, right, bottom and left border in this sequence. For example: #BBBBBB #666666 #666666 #BBBBBB	Optional	#bbb #666 #666 #bbb #BFCFFF #00248F #00248F #BFCFFF
pointborderwidth	Border width of the slider point.	Optional	thin
			medium
			thick
			1px
			2px
			5px
			10px

pointwidth	Width of the slider point in pixels. The value must be an integer value.	Optional	10 20 40 100 300
pointheight	Height of the slider point in pixels. The value must be an integer value.	Optional	10 20 40 100 300
linebgcolor	Background color of the slider line. This should be a valid CSS color value. For example a name(blue, red), a hexadecimal value(#99CCFF) or others.	Optional	#FF0000 #00FF00 #0000FF #FFFFFF #808080 #000000
linebordercolor	Border color of the slider line. This should be a valid CSS border-color value. You can specify a different color for the top, right, bottom and left border in this sequence. For example: #BBBBBB #666666 #666666 #BBBBBB	Optional	#bbb #666 #666 #bbb #BFCFFF #00248F #00248F #BFCFFF
lineborderwidth	Border width of the slider line.	Optional	thin medium thick 1px 2px 5px 10px

rangefontsize	Font size of the slider range.	Optional	xx-small
			x-small
			small
			medium
			large
			x-large
			xx-large
			smaller
			larger
			150%
valuebgcolor	Background color of the slider current value which is shown if the "showcurrentvalue" property is set to true. This should be a valid CSS color value. For example a name(blue, red), a hexadecimal value(#99CCFF) or others.	Optional	#FF0000 #00FF00 #0000FF #FFFFFF #808080 #000000
valuebordercolor	Background color of the slider current value which is shown if the "showcurrentvalue" property is set to true. This should be a valid CSS border-color value. You can specify a different color for the top, right, bottom and left border in this sequence. For example: #bbb #666 #666 #bbb	Optional	#bbb #666 #666 #bbb #BFCFFF #00248F #00248F #BFCFFF
valueborderwidth	Border width of the slider current value which is shown if the "showcurrentvalue" property is set to true.	Optional	thin medium thick 1px 2px 5px 10px

valuefontsize	Font size of the slider current value which is shown if the "showcurrentvalue" property is set to true.	Optional	xx-small
			x-small
			small
			medium
			large
			x-large
			xx-large
			smaller
			larger
			150%