| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/FormView.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/javax/swing/text/html/FormSubmitEvent.MethodType.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/text/html/FormView.MouseEventListener.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/text/html/FormView.html)    [**NO FRAMES**](http://docs.google.com/FormView.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | [FIELD](#2et92p0) | [CONSTR](#1t3h5sf) | [METHOD](#4d34og8) | DETAIL: [FIELD](#26in1rg) | [CONSTR](#1ksv4uv) | [METHOD](#2jxsxqh) |

## **javax.swing.text.html**

Class FormView

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 [javax.swing.text.View](http://docs.google.com/javax/swing/text/View.html)  
 [javax.swing.text.ComponentView](http://docs.google.com/javax/swing/text/ComponentView.html)  
 **javax.swing.text.html.FormView**

**All Implemented Interfaces:** [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html), [EventListener](http://docs.google.com/java/util/EventListener.html), [SwingConstants](http://docs.google.com/javax/swing/SwingConstants.html)

public class **FormView**extends [ComponentView](http://docs.google.com/javax/swing/text/ComponentView.html)implements [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html)

Component decorator that implements the view interface for form elements, <input>, <textarea>, and <select>. The model for the component is stored as an attribute of the the element (using StyleConstants.ModelAttribute), and is used to build the component of the view. The type of the model is assumed to of the type that would be set by HTMLDocument.HTMLReader.FormAction. If there are multiple views mapped over the document, they will share the embedded component models.

The following table shows what components get built by this view.

| Element Type | Component built |
| --- | --- |
| input, type button | JButton |
| input, type checkbox | JCheckBox |
| input, type image | JButton |
| input, type password | JPasswordField |
| input, type radio | JRadioButton |
| input, type reset | JButton |
| input, type submit | JButton |
| input, type text | JTextField |
| select, size > 1 or multiple attribute defined | JList in a JScrollPane |
| select, size unspecified or 1 | JComboBox |
| textarea | JTextArea in a JScrollPane |
| input, type file | JTextField |

| **Nested Class Summary** | |
| --- | --- |
| protected  class | [**FormView.MouseEventListener**](http://docs.google.com/javax/swing/text/html/FormView.MouseEventListener.html)            MouseEventListener class to handle form submissions when an input with type equal to image is clicked on. |

| **Field Summary** | |
| --- | --- |
| static [String](http://docs.google.com/java/lang/String.html) | [**RESET**](http://docs.google.com/javax/swing/text/html/FormView.html#RESET)  **Deprecated.** *As of 1.3, value comes from UIManager UIManager property FormView.resetButtonText* |
| static [String](http://docs.google.com/java/lang/String.html) | [**SUBMIT**](http://docs.google.com/javax/swing/text/html/FormView.html#SUBMIT)  **Deprecated.** *As of 1.3, value now comes from UIManager property FormView.submitButtonText* |

| **Fields inherited from class javax.swing.text.**[**View**](http://docs.google.com/javax/swing/text/View.html) |
| --- |
| [BadBreakWeight](http://docs.google.com/javax/swing/text/View.html#BadBreakWeight), [ExcellentBreakWeight](http://docs.google.com/javax/swing/text/View.html#ExcellentBreakWeight), [ForcedBreakWeight](http://docs.google.com/javax/swing/text/View.html#ForcedBreakWeight), [GoodBreakWeight](http://docs.google.com/javax/swing/text/View.html#GoodBreakWeight), [X\_AXIS](http://docs.google.com/javax/swing/text/View.html#X_AXIS), [Y\_AXIS](http://docs.google.com/javax/swing/text/View.html#Y_AXIS) |

| **Fields inherited from interface javax.swing.**[**SwingConstants**](http://docs.google.com/javax/swing/SwingConstants.html) |
| --- |
| [BOTTOM](http://docs.google.com/javax/swing/SwingConstants.html#BOTTOM), [CENTER](http://docs.google.com/javax/swing/SwingConstants.html#CENTER), [EAST](http://docs.google.com/javax/swing/SwingConstants.html#EAST), [HORIZONTAL](http://docs.google.com/javax/swing/SwingConstants.html#HORIZONTAL), [LEADING](http://docs.google.com/javax/swing/SwingConstants.html#LEADING), [LEFT](http://docs.google.com/javax/swing/SwingConstants.html#LEFT), [NEXT](http://docs.google.com/javax/swing/SwingConstants.html#NEXT), [NORTH](http://docs.google.com/javax/swing/SwingConstants.html#NORTH), [NORTH\_EAST](http://docs.google.com/javax/swing/SwingConstants.html#NORTH_EAST), [NORTH\_WEST](http://docs.google.com/javax/swing/SwingConstants.html#NORTH_WEST), [PREVIOUS](http://docs.google.com/javax/swing/SwingConstants.html#PREVIOUS), [RIGHT](http://docs.google.com/javax/swing/SwingConstants.html#RIGHT), [SOUTH](http://docs.google.com/javax/swing/SwingConstants.html#SOUTH), [SOUTH\_EAST](http://docs.google.com/javax/swing/SwingConstants.html#SOUTH_EAST), [SOUTH\_WEST](http://docs.google.com/javax/swing/SwingConstants.html#SOUTH_WEST), [TOP](http://docs.google.com/javax/swing/SwingConstants.html#TOP), [TRAILING](http://docs.google.com/javax/swing/SwingConstants.html#TRAILING), [VERTICAL](http://docs.google.com/javax/swing/SwingConstants.html#VERTICAL), [WEST](http://docs.google.com/javax/swing/SwingConstants.html#WEST) |

| **Constructor Summary** | |
| --- | --- |
| [**FormView**](http://docs.google.com/javax/swing/text/html/FormView.html#FormView(javax.swing.text.Element))([Element](http://docs.google.com/javax/swing/text/Element.html) elem)            Creates a new FormView object. |

| **Method Summary** | |
| --- | --- |
| void | [**actionPerformed**](http://docs.google.com/javax/swing/text/html/FormView.html#actionPerformed(java.awt.event.ActionEvent))([ActionEvent](http://docs.google.com/java/awt/event/ActionEvent.html) evt)            Responsible for processeing the ActionEvent. |
| protected  [Component](http://docs.google.com/java/awt/Component.html) | [**createComponent**](http://docs.google.com/javax/swing/text/html/FormView.html#createComponent())()            Create the component. |
| float | [**getMaximumSpan**](http://docs.google.com/javax/swing/text/html/FormView.html#getMaximumSpan(int))(int axis)            Determines the maximum span for this view along an axis. |
| protected  void | [**imageSubmit**](http://docs.google.com/javax/swing/text/html/FormView.html#imageSubmit(java.lang.String))([String](http://docs.google.com/java/lang/String.html) imageData)            This method is called to submit a form in response to a click on an image -- an <INPUT> form element of type "image". |
| protected  void | [**submitData**](http://docs.google.com/javax/swing/text/html/FormView.html#submitData(java.lang.String))([String](http://docs.google.com/java/lang/String.html) data)            This method is responsible for submitting the form data. |

| **Methods inherited from class javax.swing.text.**[**ComponentView**](http://docs.google.com/javax/swing/text/ComponentView.html) |
| --- |
| [getAlignment](http://docs.google.com/javax/swing/text/ComponentView.html#getAlignment(int)), [getComponent](http://docs.google.com/javax/swing/text/ComponentView.html#getComponent()), [getMinimumSpan](http://docs.google.com/javax/swing/text/ComponentView.html#getMinimumSpan(int)), [getPreferredSpan](http://docs.google.com/javax/swing/text/ComponentView.html#getPreferredSpan(int)), [modelToView](http://docs.google.com/javax/swing/text/ComponentView.html#modelToView(int,%20java.awt.Shape,%20javax.swing.text.Position.Bias)), [paint](http://docs.google.com/javax/swing/text/ComponentView.html#paint(java.awt.Graphics,%20java.awt.Shape)), [setParent](http://docs.google.com/javax/swing/text/ComponentView.html#setParent(javax.swing.text.View)), [viewToModel](http://docs.google.com/javax/swing/text/ComponentView.html#viewToModel(float,%20float,%20java.awt.Shape,%20javax.swing.text.Position.Bias%5B%5D)) |

| **Methods inherited from class javax.swing.text.**[**View**](http://docs.google.com/javax/swing/text/View.html) |
| --- |
| [append](http://docs.google.com/javax/swing/text/View.html#append(javax.swing.text.View)), [breakView](http://docs.google.com/javax/swing/text/View.html#breakView(int,%20int,%20float,%20float)), [changedUpdate](http://docs.google.com/javax/swing/text/View.html#changedUpdate(javax.swing.event.DocumentEvent,%20java.awt.Shape,%20javax.swing.text.ViewFactory)), [createFragment](http://docs.google.com/javax/swing/text/View.html#createFragment(int,%20int)), [forwardUpdate](http://docs.google.com/javax/swing/text/View.html#forwardUpdate(javax.swing.event.DocumentEvent.ElementChange,%20javax.swing.event.DocumentEvent,%20java.awt.Shape,%20javax.swing.text.ViewFactory)), [forwardUpdateToView](http://docs.google.com/javax/swing/text/View.html#forwardUpdateToView(javax.swing.text.View,%20javax.swing.event.DocumentEvent,%20java.awt.Shape,%20javax.swing.text.ViewFactory)), [getAttributes](http://docs.google.com/javax/swing/text/View.html#getAttributes()), [getBreakWeight](http://docs.google.com/javax/swing/text/View.html#getBreakWeight(int,%20float,%20float)), [getChildAllocation](http://docs.google.com/javax/swing/text/View.html#getChildAllocation(int,%20java.awt.Shape)), [getContainer](http://docs.google.com/javax/swing/text/View.html#getContainer()), [getDocument](http://docs.google.com/javax/swing/text/View.html#getDocument()), [getElement](http://docs.google.com/javax/swing/text/View.html#getElement()), [getEndOffset](http://docs.google.com/javax/swing/text/View.html#getEndOffset()), [getGraphics](http://docs.google.com/javax/swing/text/View.html#getGraphics()), [getNextVisualPositionFrom](http://docs.google.com/javax/swing/text/View.html#getNextVisualPositionFrom(int,%20javax.swing.text.Position.Bias,%20java.awt.Shape,%20int,%20javax.swing.text.Position.Bias%5B%5D)), [getParent](http://docs.google.com/javax/swing/text/View.html#getParent()), [getResizeWeight](http://docs.google.com/javax/swing/text/View.html#getResizeWeight(int)), [getStartOffset](http://docs.google.com/javax/swing/text/View.html#getStartOffset()), [getToolTipText](http://docs.google.com/javax/swing/text/View.html#getToolTipText(float,%20float,%20java.awt.Shape)), [getView](http://docs.google.com/javax/swing/text/View.html#getView(int)), [getViewCount](http://docs.google.com/javax/swing/text/View.html#getViewCount()), [getViewFactory](http://docs.google.com/javax/swing/text/View.html#getViewFactory()), [getViewIndex](http://docs.google.com/javax/swing/text/View.html#getViewIndex(float,%20float,%20java.awt.Shape)), [getViewIndex](http://docs.google.com/javax/swing/text/View.html#getViewIndex(int,%20javax.swing.text.Position.Bias)), [insert](http://docs.google.com/javax/swing/text/View.html#insert(int,%20javax.swing.text.View)), [insertUpdate](http://docs.google.com/javax/swing/text/View.html#insertUpdate(javax.swing.event.DocumentEvent,%20java.awt.Shape,%20javax.swing.text.ViewFactory)), [isVisible](http://docs.google.com/javax/swing/text/View.html#isVisible()), [modelToView](http://docs.google.com/javax/swing/text/View.html#modelToView(int,%20javax.swing.text.Position.Bias,%20int,%20javax.swing.text.Position.Bias,%20java.awt.Shape)), [modelToView](http://docs.google.com/javax/swing/text/View.html#modelToView(int,%20java.awt.Shape)), [preferenceChanged](http://docs.google.com/javax/swing/text/View.html#preferenceChanged(javax.swing.text.View,%20boolean,%20boolean)), [remove](http://docs.google.com/javax/swing/text/View.html#remove(int)), [removeAll](http://docs.google.com/javax/swing/text/View.html#removeAll()), [removeUpdate](http://docs.google.com/javax/swing/text/View.html#removeUpdate(javax.swing.event.DocumentEvent,%20java.awt.Shape,%20javax.swing.text.ViewFactory)), [replace](http://docs.google.com/javax/swing/text/View.html#replace(int,%20int,%20javax.swing.text.View%5B%5D)), [setSize](http://docs.google.com/javax/swing/text/View.html#setSize(float,%20float)), [updateChildren](http://docs.google.com/javax/swing/text/View.html#updateChildren(javax.swing.event.DocumentEvent.ElementChange,%20javax.swing.event.DocumentEvent,%20javax.swing.text.ViewFactory)), [updateLayout](http://docs.google.com/javax/swing/text/View.html#updateLayout(javax.swing.event.DocumentEvent.ElementChange,%20javax.swing.event.DocumentEvent,%20java.awt.Shape)), [viewToModel](http://docs.google.com/javax/swing/text/View.html#viewToModel(float,%20float,%20java.awt.Shape)) |

| **Methods inherited from class java.lang.**[**Object**](http://docs.google.com/java/lang/Object.html) |
| --- |
| [clone](http://docs.google.com/java/lang/Object.html#clone()), [equals](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)), [finalize](http://docs.google.com/java/lang/Object.html#finalize()), [getClass](http://docs.google.com/java/lang/Object.html#getClass()), [hashCode](http://docs.google.com/java/lang/Object.html#hashCode()), [notify](http://docs.google.com/java/lang/Object.html#notify()), [notifyAll](http://docs.google.com/java/lang/Object.html#notifyAll()), [toString](http://docs.google.com/java/lang/Object.html#toString()), [wait](http://docs.google.com/java/lang/Object.html#wait()), [wait](http://docs.google.com/java/lang/Object.html#wait(long)), [wait](http://docs.google.com/java/lang/Object.html#wait(long,%20int)) |

| **Field Detail** |
| --- |

### SUBMIT

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
public static final [String](http://docs.google.com/java/lang/String.html) **SUBMIT**

**Deprecated.** *As of 1.3, value now comes from UIManager property FormView.submitButtonText*If a value attribute is not specified for a FORM input element of type "submit", then this default string is used.

### RESET

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
public static final [String](http://docs.google.com/java/lang/String.html) **RESET**

**Deprecated.** *As of 1.3, value comes from UIManager UIManager property FormView.resetButtonText*If a value attribute is not specified for a FORM input element of type "reset", then this default string is used.

| **Constructor Detail** |
| --- |

### FormView

public **FormView**([Element](http://docs.google.com/javax/swing/text/Element.html) elem)

Creates a new FormView object.

**Parameters:**elem - the element to decorate

| **Method Detail** |
| --- |

### createComponent

protected [Component](http://docs.google.com/java/awt/Component.html) **createComponent**()

Create the component. This is basically a big switch statement based upon the tag type and html attributes of the associated element.

**Overrides:**[createComponent](http://docs.google.com/javax/swing/text/ComponentView.html#createComponent()) in class [ComponentView](http://docs.google.com/javax/swing/text/ComponentView.html)

### getMaximumSpan

public float **getMaximumSpan**(int axis)

Determines the maximum span for this view along an axis. For certain components, the maximum and preferred span are the same. For others this will return the value returned by Component.getMaximumSize along the axis of interest.

**Overrides:**[getMaximumSpan](http://docs.google.com/javax/swing/text/ComponentView.html#getMaximumSpan(int)) in class [ComponentView](http://docs.google.com/javax/swing/text/ComponentView.html) **Parameters:**axis - may be either View.X\_AXIS or View.Y\_AXIS **Returns:**the span the view would like to be rendered into >= 0. Typically the view is told to render into the span that is returned, although there is no guarantee. The parent may choose to resize or break the view. **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - for an invalid axis**See Also:**[View.getPreferredSpan(int)](http://docs.google.com/javax/swing/text/View.html#getPreferredSpan(int))

### actionPerformed

public void **actionPerformed**([ActionEvent](http://docs.google.com/java/awt/event/ActionEvent.html) evt)

Responsible for processeing the ActionEvent. If the element associated with the FormView, has a type of "submit", "reset", "text" or "password" then the action is processed. In the case of a "submit" the form is submitted. In the case of a "reset" the form is reset to its original state. In the case of "text" or "password", if the element is the last one of type "text" or "password", the form is submitted. Otherwise, focus is transferred to the next component in the form.

**Specified by:**[actionPerformed](http://docs.google.com/java/awt/event/ActionListener.html#actionPerformed(java.awt.event.ActionEvent)) in interface [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html) **Parameters:**evt - the ActionEvent.

### submitData

protected void **submitData**([String](http://docs.google.com/java/lang/String.html) data)

This method is responsible for submitting the form data. A thread is forked to undertake the submission.

### imageSubmit

protected void **imageSubmit**([String](http://docs.google.com/java/lang/String.html) imageData)

This method is called to submit a form in response to a click on an image -- an <INPUT> form element of type "image".

**Parameters:**imageData - the mouse click coordinates.

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/FormView.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/javax/swing/text/html/FormSubmitEvent.MethodType.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/text/html/FormView.MouseEventListener.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/text/html/FormView.html)    [**NO FRAMES**](http://docs.google.com/FormView.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | [FIELD](#2et92p0) | [CONSTR](#1t3h5sf) | [METHOD](#4d34og8) | DETAIL: [FIELD](#26in1rg) | [CONSTR](#1ksv4uv) | [METHOD](#2jxsxqh) |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).