

The h:inputSecret tag renders an HTML input element of the type "password".

JSF Tag

```
<h:inputSecret value="password" />
```

Rendered Output

```
<input type="password" name="j_idt12:j_idt16" value="password" />
```

Tag Attributes

S.N.	Attribute & Description
1	id Identifier for a component
2	binding Reference to the component that can be used in a backing bean
3	rendered A boolean; false suppresses rendering
4	styleClass Cascading stylesheet CSS class name
5	value A component's value, typically a value binding
6	valueChangeListener A method binding to a method that responds to value changes
7	converter Converter class name
8	validator Class name of a validator that's created and attached to a component
9	required A boolean; if true, requires a value to be entered in the associated field
10	accesskey

A key, typically combined with a system-defined metakey, that gives focus to an element

11 **accept**

Comma-separated list of content types for a form

12 **accept-charset**

Comma- or space-separated list of character encodings for a form. The **accept-charset** attribute is specified with the JSF HTML attribute named **acceptcharset**.

13 **alt**

Alternative text for nontextual elements such as images or applets

14 **border**

Pixel value for an element's border width

15 **charset**

Character encoding for a linked resource

16 **coords**

Coordinates for an element whose shape is a rectangle, circle, or polygon

17 **dir**

Direction for text. Valid values are **ltr** *lefttoright* and **rtl** *righttoleft*.

18 **disabled**

Disabled state of an input element or button

19 **hreflang**

Base language of a resource specified with the **href** attribute; **hreflang** may only be used with **href**.

20 **lang**

Base language of an element's attributes and text

21 **maxlength**

Maximum number of characters for text fields

22 **readonly**

Read-only state of an input field; text can be selected in a readonly field but not edited

23 **style**

Inline style information

24	tabindex	Numerical value specifying a tab index
25	target	The name of a frame in which a document is opened
26	title	A title, used for accessibility, that describes an element. Visual browsers typically create tooltips for the title's value
27	type	Type of a link; for example, stylesheet
28	width	Width of an element
29	onblur	Element loses focus
30	onchange	Element's value changes
31	onclick	Mouse button is clicked over the element
32	ondblclick	Mouse button is double-clicked over the element
33	onfocus	Element receives focus
34	onkeydown	Key is pressed
35	onkeypress	Key is pressed and subsequently released
36	onkeyup	Key is released
37	onmousedown	Mouse button is pressed over the element

38	onmousemove	Mouse moves over the element
39	onmouseout	Mouse leaves the element's area
40	onmouseover	Mouse moves onto an element
41	onmouseup	Mouse button is released
42	onreset	Form is reset
43	onselect	Text is selected in an input field
44	immediate	Process validation early in the life cycle
45	redisplay	when true, the input field's value is redisplayed when the web page is reloaded

Example Application

Let us create a test JSF application to test the above tag.

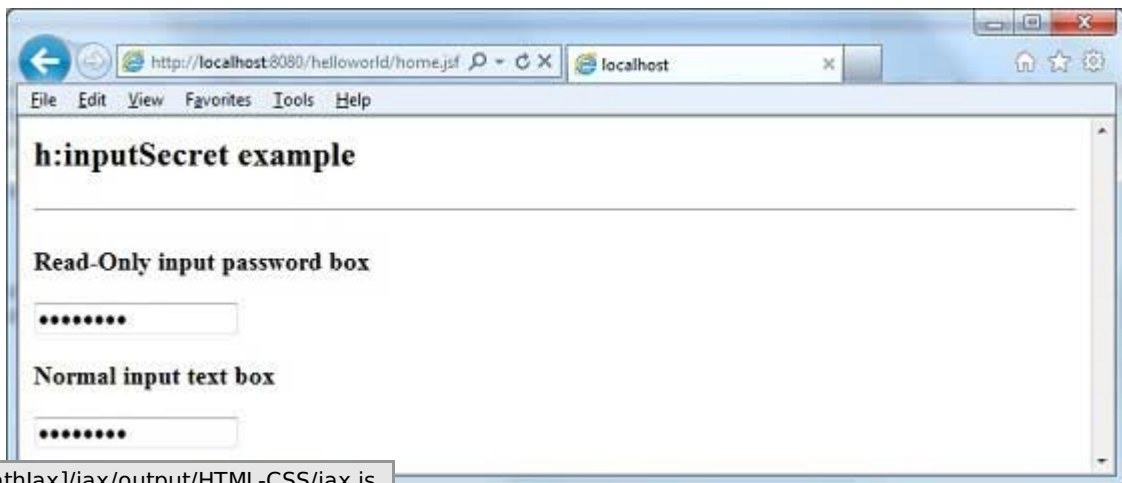
Step	Description
1	Create a project with a name <i>helloworld</i> under a package <i>com.tutorialspoint.test</i> as explained in the <i>JSF - First Application</i> chapter.
2	Modify <i>home.xhtml</i> as explained below. Keep rest of the files unchanged.
3	Compile and run the application to make sure business logic is working as per the requirements.
4	Finally, build the application in the form of war file and deploy it in Apache Tomcat Webserver.
5	Launch your web application using appropriate URL as explained below in the last step.

home.xhtml

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
<head>
  <title>JSF Tutorial!</title>
</head>
<body>
  <h2>h:inputSecret example</h2>
  <hr />
  <h:form>
    <h3>Read-Only input password box</h3>
    <h:inputSecret value="password" readonly="true"/>
    <h3>Read-Only input text box</h3>
    <h:inputText value="password"/>
  </h:form>
</body>
</html>
```

Once you are ready with all the changes done, let us compile and run the application as we did in JSF - First Application chapter. If everything is fine with your application, this will produce following result:



Loading [MathJax]/jax/output/HTML-CSS/jax.js