

The h:panel tag renders an HTML "table" element.

JSF Tag

```
<h:panelGrid
  cellpadding="10" cellspacing="1">
  <f:facet name="header">
    <h:outputText value="Login"/>
  </f:facet>
  <h:outputLabel value="Username" />
  <h:inputText />
  <h:outputLabel value="Password" />
  <h:inputSecret />
  <f:facet name="footer">
    <h:panelGroup style="display:block; text-align:center">
      <h:commandButton />
    </h:panelGroup>
  </f:facet>
</h:panelGrid>
```

Rendered Output

```
<table >
<thead>
  <tr><th colspan="2" scope="colgroup">Login</th></tr>
</thead>
<tfoot>
  <tr>
    <td colspan="2">
      <span style="display:block; text-align:center">
        <input
          name="j_idt10:submit" value="Submit" />
        </span></td></tr>
</tfoot>
<tbody>
  <tr>
    <td><label>Username</label></td>
    <td><input type="text" name="j_idt10:j_idt17" /></td>
  </tr>
  <tr>
    <td><label>Password</label></td>
    <td><input type="password" name="j_idt10:j_idt21" value="" /></td>
  </tr>
</tbody>
</table>
```

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	bgcolor	Background color for the table
7	border	Width of the table's border
8	cellpadding	Padding around table cells
9	cellspacing	Spacing between table cells
10	columnClasses	Comma-separated list of CSS classes for columns
11	columns	Number of columns in the table
12	footerClass	CSS class for the table footer
13	frame	frame Specification for sides of the frame surrounding the table that are to be drawn; valid values: none, above, below, hside, vside, lhs, rhs, box, border
14	headerClass	CSS class for the table header
15	rowClasses	Comma-separated list of CSS classes for columns
16	rules	Specification for lines drawn between cells; valid values: groups, rows, columns, all

17	summary	Summary of the table's purpose and structure used for non-visual feedback such as speech
18	dir	Direction for text. Valid values are ltr <i>lefttoright</i> and rtl <i>righttoleft</i> .
19	lang	Base language of an element's attributes and text
20	border	Pixel value for an element's border width
21	title	A title, used for accessibility, that describes an element. Visual browsers typically create tooltips for the title's value
22	width	Width of an element
23	onblur	Element loses focus
24	onchange	Element's value changes
25	onclick	Mouse button is clicked over the element
26	ondblclick	Mouse button is double-clicked over the element
27	onfocus	Element receives focus
28	onkeydown	Key is pressed
29	onkeypress	Key is pressed and subsequently released
30	onkeyup	Key is released

- 31 **onmousedown**
Mouse button is pressed over the element
- 32 **onmousemove**
Mouse moves over the element
- 33 **onmouseout**
Mouse leaves the element's area
- 34 **onmouseover**
Mouse moves onto an element
- 35 **onmouseup**
Mouse button is released

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:panelGrid example</h2>
    <hr />
    <h:form>
        <h:panelGrid
            cellpadding="10" cellspacing="1">
            <f:facet name="header">
                <h:outputText value="Login"/>
            </f:facet>
            <h:outputLabel value="Username" />
            <h:inputText />
        </h:panelGrid>
    </h:form>
</body>
</html>
```

```

        <h:outputLabel value="Password" />
        <h:inputSecret />
        <f:facet name="footer">
            <h:panelGroup style="display:block; text-align:center">
                <h:commandButton />
            </h:panelGroup>
        </f:facet>
    </h:panelGrid>
</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:



The screenshot shows a web browser window with the address bar displaying `http://localhost:8080/helloworld/home.jsf`. The browser has a single tab titled "JSF Tutorial!". The page content is titled "h:panelGrid example". Below the title is a login form. The form has a header section labeled "Login". Below this header are two input fields: "Username" and "Password". Below these fields is a "Submit" button. The browser's status bar at the bottom shows the message "Loading [MathJax]/jax/output/HTML-CSS/jax.js".