

GWT - HTMLPANEL WIDGET

http://www.tutorialspoint.com/gwt/gwt_htmlpanel_widget.htm

Copyright © tutorialspoint.com

Introduction

The **HTMLPanel** widget represents a panel that contains HTML, and which can attach child widgets to identified elements within that HTML.

Class declaration

Following is the declaration for **com.google.gwt.user.client.ui.HTMLPanel** class:

```
public class HTMLPanel
    extends ComplexPanel
```

Class constructors

S.N.	Constructor & Description
1	HTMLPanel <i>SafeHtmlsafeHtml</i> Initializes the panel's HTML from a given SafeHtml object.
2	HTMLPanel <i>java. lang. Stringhtml</i> Creates an HTML panel with the specified HTML contents inside a DIV element.
3	HTMLPanel <i>java. lang. Stringtag, java. lang. Stringhtml</i> Creates an HTML panel whose root element has the given tag, and with the specified HTML contents.

Class methods

S.N.	Function name & Description
1	void add <i>Widgetwidget, Elementelem</i> Adds a child widget to the panel, contained within an HTML element.
2	void add <i>Widgetwidget, java. lang. Stringid</i> Adds a child widget to the panel, contained within the HTML element specified by a given id.
3	void addAndReplaceElement <i>Widgetwidget, ElementtoReplace</i> Adds a child widget to the panel, replacing the HTML element.

- 4 **void addAndReplaceElementWidget***widget, java. lang. Stringid*
- Adds a child widget to the panel, replacing the HTML element specified by a given id.
- 5 **static java.lang.String createUniqueld**
- A helper method for creating unique IDs for elements within dynamically- generated HTML.
- 6 **Element getElementById***java. lang. Stringid*
- Finds an element within this panel by its id.

Methods inherited

This class inherits methods from the following classes:

- com.google.gwt.user.client.ui.UIObject
- com.google.gwt.user.client.ui.Widget
- com.google.gwt.user.client.ui.Panel
- com.google.gwt.user.client.ui.ComplexPanel
- java.lang.Object

HTMLPanel Widget Example

This example will take you through simple steps to show usage of a HTMLPanel Widget in GWT. Follow the following steps to update the GWT application we created in *GWT - Create Application* chapter:

Step	Description
1	Create a project with a name <i>HelloWorld</i> under a package <i>com.tutorialspoint</i> as explained in the <i>GWT - Create Application</i> chapter.
2	Modify <i>HelloWorld.gwt.xml</i> , <i>HelloWorld.css</i> , <i>HelloWorld.html</i> and <i>HelloWorld.java</i> as explained below. Keep rest of the files unchanged.
3	Compile and run the application to verify the result of the implemented logic.

Following is the content of the modified module descriptor **src/com.tutorialspoint/HelloWorld.gwt.xml**.

```
<?xml version="1.0" encoding="UTF-8"?>
<module rename-to='helloworld'>
  <!-- Inherit the core Web Toolkit stuff.                        -->
  <inherits name='com.google.gwt.user.User' />

  <!-- Inherit the default GWT style sheet.                      -->
  <inherits name='com.google.gwt.user.theme.clean.Clean' />

  <!-- Specify the app entry point class.                        -->
  <entry-point class='com.tutorialspoint.client.HelloWorld' />

  <!-- Specify the paths for translatable code                  -->
```

```
<source path='client' />
<source path='shared' />

</module>
```

Following is the content of the modified Style Sheet file **war/HelloWorld.css**.

```
body{
    text-align: center;
    font-family: verdana, sans-serif;
}
h1{
    font-size: 2em;
    font-weight: bold;
    color: #777777;
    margin: 40px 0px 70px;
    text-align: center;
}
```

Following is the content of the modified HTML host file **war/HelloWorld.html**.

```
<html>
<head>
<title>Hello World</title>
    <link rel="stylesheet" href="HelloWorld.css"/>
    <script language="javascript" src="helloworld/helloworld.nocache.js">
    </script>
</head>
<body>

<h1>HTMLPanel Widget Demonstration</h1>
<div ></div>

</body>
</html>
```

Let us have following content of Java file **src/com.tutorialspoint/HelloWorld.java** which will demonstrate use of HTMLPanel widget.

```
package com.tutorialspoint.client;

import com.google.gwt.core.client.EntryPoint;
import com.google.gwt.user.client.ui.DecoratorPanel;
import com.google.gwt.user.client.ui.HTMLPanel;
import com.google.gwt.user.client.ui.RootPanel;

public class HelloWorld implements EntryPoint {
    public void onModuleLoad() {
        String htmlString = "This is a <b>HTMLPanel</b> containing"
            +" html contents. "
            +" <i>By putting some fairly large contents in the middle"
            +" and setting its size explicitly, it becomes a scrollable area"
            +" within the page, but without requiring the use of an IFRAME.</i>"
            +" <u>Here's quite a bit more meaningless text that will serve"
            +" to make this thing scroll off the bottom of its visible area."
            +" Otherwise, you might have to make it really, really"
            +" small in order to see the nifty scroll bars!</u>";

        HTMLPanel htmlPanel = new HTMLPanel(htmlString);

        DecoratorPanel panel = new DecoratorPanel();
        panel.add(htmlPanel);

        // Add the widgets to the root panel.
        RootPanel.get().add(panel);
    }
}
```

Once you are ready with all the changes done, let us compile and run the application in development mode as we did in [GWT - Create Application](#) chapter. If everything is fine with your application, this will produce following result:

