GWT - SCROLLPANEL WIDGET

Introduction

The **ScrollPanel** widget represents a simple panel that wraps its contents in a scrollable area.

Class declaration

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

```
public class ScrollPanel
   extends SimplePanel
      implements SourcesScrollEvents, HasScrollHandlers,
         RequiresResize, ProvidesResize
```

Class constructors

S.N.	Constructor & Description
1	**ScrollPanel** Creates an empty scroll panel.
2	**ScrollPanel***Widgetchild* Creates a new scroll panel with the given child widget.

Class methods

S.N.	Function name & Description
1	**HandlerRegistration addScrollHandler***ScrollHandlerhandler* Adds a ScrollEvent handler.
2	**void addScrollListener***ScrollListenerlistener* Deprecated. Use addScrollHandler*com. google. gwt. event. dom. client. ScrollHandler* instead
3	**void ensureVisible***UIObjectitem* Ensures that the specified item is visible, by adjusting the panel's scroll position.
4	**protected Element getContainerElement** Override this method to specify that an element other than the root element be the container for the panel's child widget.

5

int getHorizontalScrollPosition

Gets the horizontal scroll position.

6

int getScrollPosition

Gets the vertical scroll position.

7

void onResize

This method must be called whenever the implementor's size has been modified.

8

void removeScrollListener*ScrollListenerlistener*

Deprecated. Use the HandlerRegistration.removeHandler method on the object returned by addScrollHandler*com. google. gwt. event. dom. client. ScrollHandler* instead

9

void scrollToBottom

Scroll to the bottom of this panel.

10

void scrollToLeft

Scroll to the far left of this panel.

11

void scrollToRight

Scroll to the far right of this panel.

12

void scrollToTop

Scroll to the top of this panel.

13

void setAlwaysShowScrollBars*booleanalwaysShow*

Sets whether this panel always shows its scroll bars, or only when necessary.

14

void setHeight*java. lang. Stringheight*

Sets the object's height.

15

void setHorizontalScrollPosition*intposition*

Sets the horizontal scroll position.

16

void setScrollPosition*intposition*

Sets the vertical scroll position.

17

void setSize*java. lang. Stringwidth, java. lang. Stringheight*

Sets the object's size.

18

void setWidth*java. lang. Stringwidth*

Sets the object's width.

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.SimplePanel
- java.lang.Object

ScrollPanel Widget Example

This example will take you through simple steps to show usage of a ScrollPanel 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 *HelloWorld* under a package *com.tutorialspoint* as explained in the *GWT - Create Application* chapter.
2	Modify *HelloWorld.gwt.xml*, *HelloWorld.css*, *HelloWorld.html* and *HelloWorld.java* 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>ScrollPanel 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 ScrollPanel 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.HTML;
import com.google.gwt.user.client.ui.RootPanel;
import com.google.gwt.user.client.ui.ScrollPanel;

public class HelloWorld implements EntryPoint {

   public void onModuleLoad() {
      // Create scrollable text
      HTML contents = new HTML("This is a ScrollPanel."
      +" 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."
      +" Here's quite a bit more meaningless text that will serve primarily"
      +" 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!");

      //create scrollpanel with content
      ScrollPanel scrollPanel = new ScrollPanel(contents);
      scrollPanel.setSize("400px", "100px");

      DecoratorPanel decoratorPanel = new DecoratorPanel();

      decoratorPanel.add(scrollPanel);

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

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: