

GWT - LISTBOX WIDGET

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

Copyright © tutorialspoint.com

Introduction

The **ListBox** widget represents a list of choices to the user, either as a list box or as a drop-down list.

Class declaration

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

```
public class ListBox
    extends FocusWidget
    implements SourceChangeEvents, HasName
```

CSS style rules

Following default CSS Style rules will be applied to all the ListBox widget. You can override it as per your requirements.

```
.gwt-ListBox {}
```

Class constructors

S.N.	Constructor & Description
1	ListBox Creates an empty list box in single selection mode.
2	ListBoxbooleanisMultipleSelect Creates an empty list box.
3	ListBoxElementelement This constructor may be used by subclasses to explicitly use an existing element.

Class methods

S.N.	Function name & Description
1	void addItem <i>java. lang. Stringitem</i> Adds an item to the list box.
2	void addItem <i>java. lang. Stringitem, java. lang. Stringvalue</i> Adds an item to the list box, specifying an initial value for the item.

3

void clear

Removes all items from the list box.

4

int getItemCount

Gets the number of items present in the list box.

5

java.lang.String getItemText*int index*

Gets the text associated with the item at the specified index.

6

java.lang.String getName

Gets the widget's name.

7

int getSelectedIndex

Gets the currently-selected item.

8

java.lang.String getValue*int index*

Gets the value associated with the item at a given index.

9

int getVisibleItemCount

Gets the number of items that are visible.

10

void insertItem*java. lang. String item, int index*

Inserts an item into the list box.

11

void insertItem*java. lang. String item, java. lang. String value, int index*

Inserts an item into the list box, specifying an initial value for the item.

12

boolean isSelected*int index*

Determines whether an individual list item is selected.

13

boolean isMultipleSelect

Gets whether this list allows multiple selection.

14

void onBrowserEvent*Event event*

Fired whenever a browser event is received.

15

protected void onEnsureDebugId*java. lang. StringbaseID*

Affected Elements: -item# = the option at the specified index.

16

void removeChangeListener*ChangeListenerlistener*

Removes a previously added listener interface.

17

void removeItem*intindex*

Removes the item at the specified index.

18

void setItemSelected*intindex, booleanselected*

Sets whether an individual list item is selected.

19>

void setItemText*intindex, java. lang. Stringtext*

Sets the text at given index.

20

void setMultipleSelect*booleanmultiple*

Sets whether this list allows multiple selections.

21

void setName*java. lang. Stringname*

Sets the widget's name.

22

void setSelectedIndex*intindex*

Sets the currently selected index.

23

void setValue*intindex, java. lang. Stringvalue*

Sets the value associated with the item at a given index.

24

void setVisibleItemCount*intvisibleItems*

Sets the number of items that are visible.

25

static ListBox wrap*Elementelement*

Creates a ListBox widget that wraps an existing <select> element.

26

void addChangeListener*ChangeListenerlistener*

Adds a listener interface to receive change events.

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.FocusWidget
- java.lang.Object

ListBox Widget Example

This example will take you through simple steps to show usage of a ListBox 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;
}
.gwt-ListBox{
  color:green;
```

```
}
```

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>ListBox 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 ListBox widget.

```
package com.tutorialspoint.client;

import com.google.gwt.core.client.EntryPoint;
import com.google.gwt.user.client.ui.ListBox;
import com.google.gwt.user.client.ui.RootPanel;
import com.google.gwt.user.client.ui.VerticalPanel;

public class HelloWorld implements EntryPoint {
    public void onModuleLoad() {

        // Make a new list box, adding a few items to it.
        ListBox listBox1 = new ListBox();
        listBox1.addItem("First");
        listBox1.addItem("Second");
        listBox1.addItem("Third");
        listBox1.addItem("Fourth");
        listBox1.addItem("Fifth");

        // Make a new list box, adding a few items to it.
        ListBox listBox2 = new ListBox();
        listBox2.addItem("First");
        listBox2.addItem("Second");
        listBox2.addItem("Third");
        listBox2.addItem("Fourth");
        listBox2.addItem("Fifth");

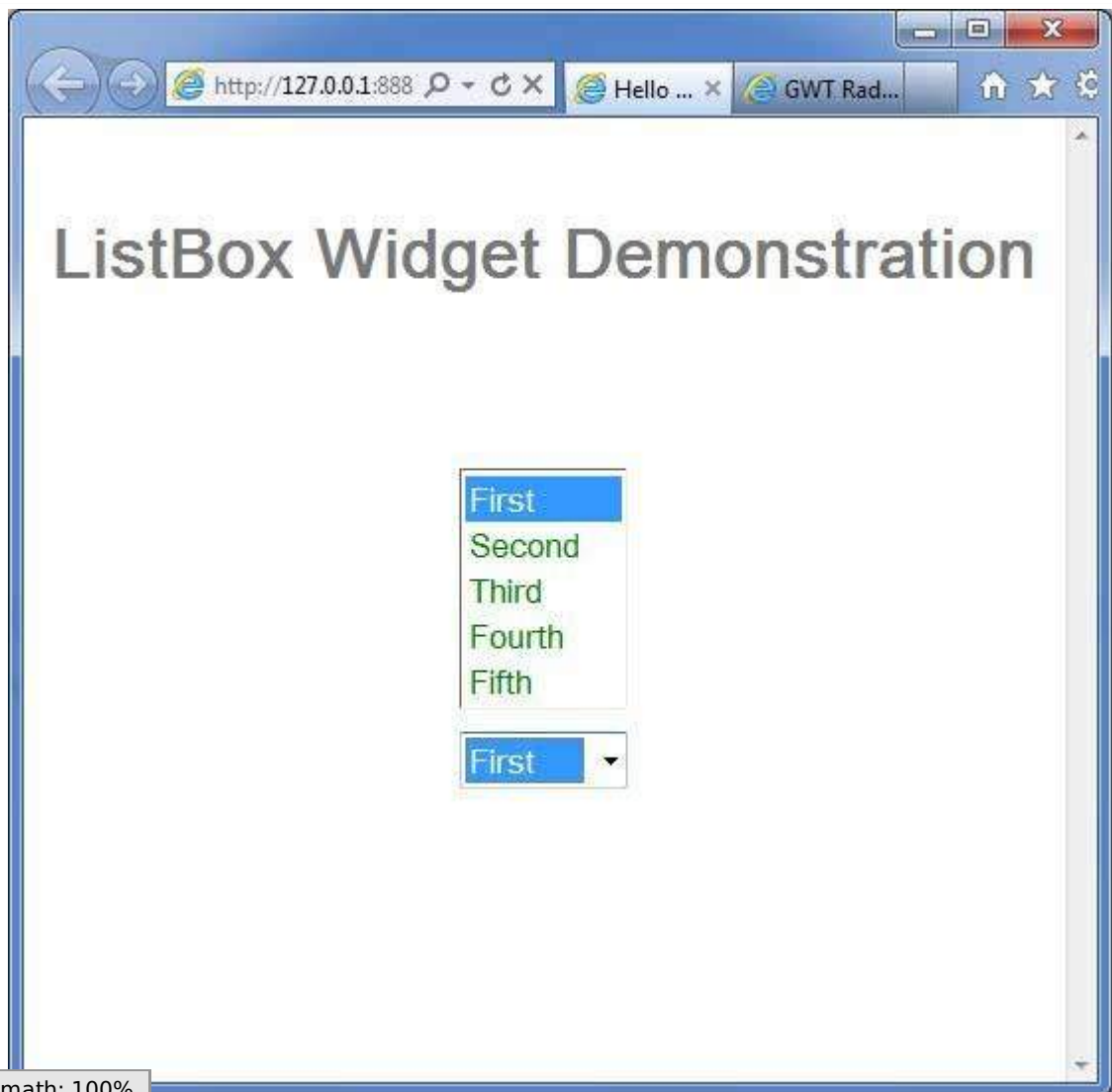
        // Make enough room for all five items
        listBox1.setVisibleItemCount(5);

        //setting itemcount value to 1 turns listbox into a drop-down list.
        listBox2.setVisibleItemCount(1);

        // Add listboxes to the root panel.
        VerticalPanel panel = new VerticalPanel();
        panel.setSpacing(10);
        panel.add(listBox1);
        panel.add(listBox2);

        RootPanel.get("gwtContainer").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:



Processing math: 100%