

GWT - CHECKBOX WIDGET

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

Copyright © tutorialspoint.com

Introduction

The **Checkbox** widget represents a standard checkbox.

Class declaration

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

```
public class CheckBox
    extends ButtonBase
    implements HasName
```

CSS style rules

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

```
.gwt-CheckBox {}
.gwt-CheckBox-disabled {}
```

Class constructors

S.N.	Constructor & Description
1	CheckBox Constructor for CheckkBox.
2	CheckBoxElementelement This constructor may be used by subclasses to explicitly use an existing element.
3	CheckBoxjava.lang.Stringlabel Creates a check box with the specified text label.
4	CheckBoxjava.lang.Stringlabel, booleanasHTML Creates a check box with the specified text label and set its contents as HTML.

Class methods

S.N.	Function name & Description
1	java.lang.String getName Gets the widget's name.

2	int getTabIndex	Gets the widget's position in the tab index.
3	java.lang.String getText	Gets this object's text.
4	boolean isChecked	Determines whether this check box is currently checked.
5	boolean isEnabled	Gets whether this widget is enabled.
6	protected void onEnsureDebugId <i>java.lang.StringbaseID</i>	Affected Elements: -label = label next to checkbox.
7	protected void onLoad	This method is called when a widget is attached to the browser's document.
8	protected void onUnload	This method is called when a widget is detached from the browser's document.
9	protected void replaceInputElement <i>Elementelem</i>	Replace the current input element with a new one.
10	void setAccessKey <i>charkey</i>	Sets the widget's 'access key'.
11	void setChecked <i>booleanchecked</i>	Checks or unchecks this check box.
12	void setEnabled <i>booleanenabled</i>	Sets whether this widget is enabled.
13	void setFocus <i>booleanfocused</i>	Explicitly focus/unfocus this widget.

- 14 **void setHTML***java. lang. Stringhtml*
Sets this object's contents via HTML.
- 15 **void setName***java. lang. Stringname*
Sets the widget's name.
- 16 **void setTabIndex***intindex*
Sets the widget's position in the tab index.
- 17 **void setText***java. lang. Stringtext*
Sets this object's text.
- 18 **void sinkEvents***sinteventBitsToAdd*
Adds a set of events to be sunk by this object.
- 19 **java.lang.String getHTML**
Gets this object's contents as HTML.

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

CheckBox Widget Example

This example will take you through simple steps to show usage of a CheckBox 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-CheckBox{
    color:green;
}
.gwt-CheckBox-disabled {
    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>CheckBox 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 CheckBox widget.

```
package com.tutorialspoint.client;

import com.google.gwt.core.client.EntryPoint;
import com.google.gwt.user.client.ui.CheckBox;
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 check box, and select it by default.
        CheckBox checkBox1 = new CheckBox("Check Me!");
        CheckBox checkBox2 = new CheckBox("Check Me!");

        // set check box as selected
        checkBox1.setValue(true);

        //disable a checkbox
        checkBox2.setEnabled(false);

        checkBox1.addClickHandler(new ClickHandler() {

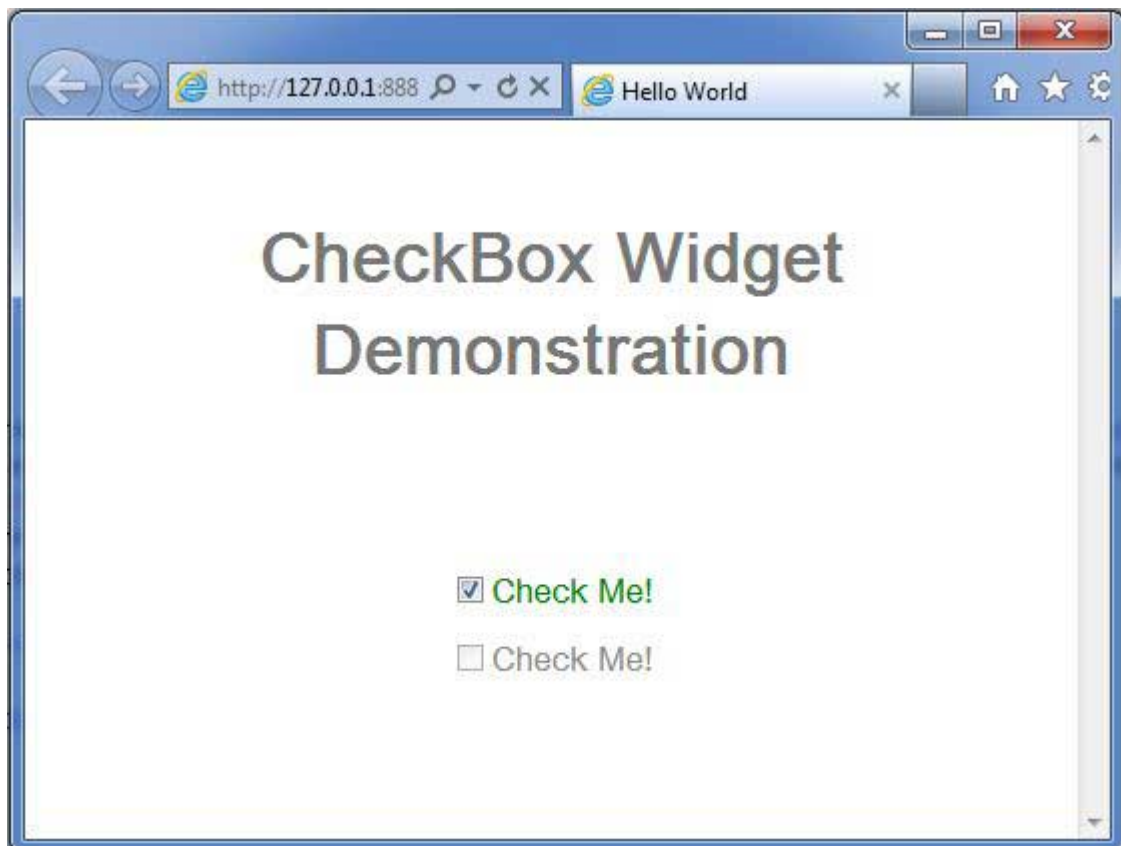
            @Override
            public void onClick(ClickEvent event) {
                CheckBox checkBox = (CheckBox)event.getSource();
                Window.alert("CheckBox is " +
                    (checkBox.getValue() ? "" : "not") + " checked");
            }
        });

        // Add checkboxes to the root panel.
        VerticalPanel panel = new VerticalPanel();
        panel.setSpacing(10);
        panel.add(checkBox1);
        panel.add(checkBox2);

        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:



When you click **Click Me** CheckBox, it will show an alert message saying **CheckBox** is checked or

not checked

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