

Introduction

The **Hidden** widget represents a hidden field in an HTML form.

Class declaration

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

```
public class Hidden
    extends Widget
    implements HasName
```

Class constructors

S.N.	Constructor & Description
1	Hidden Constructor for Hidden.
2	HiddenElementelement This constructor may be used by subclasses to explicitly use an existing element.
3	Hiddenjava. lang. Stringname Constructor for Hidden.
4	Hiddenjava. lang. Stringname, java. lang. Stringvalue Constructor for Hidden.

Class methods

S.N.	Function name & Description
1	java.lang.String getDefaultValue Gets the default value of the hidden field.
2	java.lang.String getID Gets the id of the hidden field.
3	java.lang.String getName

Gets the name of the hidden field.

4

java.lang.String getValue

Gets the value of the hidden field.

5

void setDefaultValue*java. lang. StringdefaultValue*

Sets the default value of the hidden field.

6

void setID*java. lang. Stringid*

Sets the id of the hidden field.

7

void setName*java. lang. Stringname*

Sets the name of the hidden field.

8

void setValue*java. lang. Stringvalue*

Sets the value of the hidden field.

9

static Hidden wrap*Elementelement*

Creates a Hidden widget that wraps an existing <input type='hidden'> element.

Methods inherited

This class inherits methods from the following classes:

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

Hidden Widget Example

This example will take you through simple steps to show usage of a Hidden 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>Hidden 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 Hidden widget.

```
package com.tutorialspoint.client;

import com.google.gwt.core.client.EntryPoint;
import com.google.gwt.event.dom.client.ClickEvent;
import com.google.gwt.event.dom.client.ClickHandler;
import com.google.gwt.user.client.Window;
import com.google.gwt.user.client.ui.Button;
import com.google.gwt.user.client.ui.Hidden;
import com.google.gwt.user.client.ui.RootPanel;
import com.google.gwt.user.client.ui.TextBox;
import com.google.gwt.user.client.ui.VerticalPanel;

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

```

//create textboxes
final TextBox textBox = new TextBox();
textBox.setWidth("275");
Button button1 = new Button("Set Value of Hidden Input");
Button button2 = new Button("Get Value of Hidden Input");
final Hidden hidden = new Hidden();

button1.addClickHandler(new ClickHandler() {
    @Override
    public void onClick(ClickEvent event) {
        hidden.setValue(textBox.getValue());
        Window.alert("Value of Hidden Widget Updated!");
    }
});

button2.addClickHandler(new ClickHandler() {
    @Override
    public void onClick(ClickEvent event) {
        Window.alert("Value of Hidden Widget: " + hidden.getValue());
    }
});

// Add widgets to the root panel.
VerticalPanel panel = new VerticalPanel();
panel.setSpacing(10);
panel.add(textBox);
panel.add(button1);
panel.add(hidden);
panel.add(button2);

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:





Value of Hidden Widget: This is value to be stored in Hidden widget

OK

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