

## Simple Use Case: YAHOO.widget.TabView

### Markup (optional, using standard module format):

```
<div id="mytabs" class="yui-navset">
  <ul class="yui-nav">
    <li><a href=" ..." >tab one</a></li>
    <li><a href=" ..." >tab two</a></li>
  </ul>
  <div class="yui-content">
    <div><p>Tab one content.</p></div>
    <div><p>Tab two content.</p></div>
  </div>
</div>
```

### Script:

```
var myTabs = new YAHOO.widget.TabView("mytabs");
Creates a TabView instance using existing markup.
```

## Constructor: YAHOO.widget.TabView

```
YAHOO.widget.TabView(str|HTMLElement|obj el[,
  obj config]);
```

### Arguments:

- (1) **el**: HTML ID or HTMLElement of existing markup to use when building tabView. If neither, this is treated as the Configuration object.
- (2) **Configuration Object**: JS object defining configuration properties for the TabView instance. See Configuration section for full list.

## Solutions

### Listen for a TabView Event and make use of the Event's fields.

```
var tabView = new YAHOO.widget.TabView('demo');
var handleActiveTabChange = function(e) {
  alert(e.newValue);
};
tabView.addListener('activeTabChange',
  handleActiveTabChange);
```

### Add a new Tab with with dynamic source to an existing TabView instance:

```
tabView.addTab(new YAHOO.widget.Tab({label: 'My
  Label', dataSrc: 'mySource.html',
  cacheData:true}));
```

### Remove an existing tab from a TabView:

```
tabView.removeTab(tabView.getTab(1));
```

## Key Interesting Moments in TabView

See online docs for a complete list of TabView's Events; see Solutions for how to access Event Fields.

Event:	Event Fields:
available	type (s), target (el)
beforeActiveTabChange	type (s), prevValue (Tab), newValue (Tab)
contentReady	type (s), target (el)
activeTabChange	type (s), prevValue (Tab), newValue (Tab)

All TabView events are Custom Events (see Event Utility docs); subscribe to these events using "addListener": (e.g. `myTabs.addListener('activeTabChange', fn);`).

## Key Interesting Moments in Tab

See online docs for a complete list of Tab's Events; see Solutions for how to access Event Fields.

Event:	Event Fields:
beforeContentChange	type (s), prevValue (s), newValue (s)
beforeActiveChange	type (s), prevValue (Tab), newValue (Tab)
contentChange	type (s), prevValue (s), newValue (s)
activeChange	type (s), prevValue (Tab), newValue (Tab)

All TabView events are Custom Events (see Event Utility docs); subscribe to these events using `addListener` (e.g. `myTabs.addListener('activeChange', fn);`).

## Key TabView Configuration Options

See online docs for complete list of TabView options.

Option (type)	Default	Description
activeTab (Tab)	null	The currently active Tab.
orientation	"top"	The orientation of the Tabs relative to the TabView. ("top", "right", "bottom", "left")
element	null	HTMLElement bound to TabView

TabView options can be set in the constructor's second argument (eg. `{activeTab: tabInstance}`) or at runtime via `set` (eg. `myTabs.set("activeTab", tabInstance);`).

## Key Tab Configuration Options

See online docs for complete list of Tab configuration options.

Option (type)	Default	Description
active (b)	false	Whether or not the Tab is active.
disabled (b)	false	Whether or not the Tab is disabled.
label (s)	null	The text (or innerHTML) to use as the Tab's label.
content (s)	null	The HTML displayed when the Tab is active.
labelEl (el)	null	The HTMLElement containing the <code>label</code> .
contentEl (el)	null	The HTMLElement containing the <code>contentEl</code> .
dataSrc (s)	null	Url to use for retrieving content.
cacheData (b)	false	Whether or not data retrieved from <code>dataSrc</code> should be cached or reloaded each time the Tab is activated.
Element (el)	null	HTMLElement bound to Tab

Tab options can be set in the constructor's second argument (eg. `{disabled: true}`) or at runtime via `set` (eg. `myTab.set("disabled", true);`).

## YAHOO.widget.TabView: Properties

CLASSNAME  
TAB\_PARENT\_CLASSNAME  
CONTENT\_PARENT\_CLASSNAME

## YAHOO.widget.TabView: Methods

### Methods

`addTab(Tab)`  
`removeTab(Tab)`  
`getTab(i)`  
`getTabIndex(Tab)`  
`contentTransition()`  
`set(option, value)`  
`get(option)`

## YAHOO.widget.Tab: Properties

### Tab: Properties

LABEL\_TAGNAME  
ACTIVE\_CLASSNAME  
DISABLED\_CLASSNAME  
LOADING\_CLASSNAME

## YAHOO.widget.Tab: Methods

### Tab: Methods

`set(option, value)`  
`get(option)`

## Dependencies

TabView requires the YAHOO object, Event, Dom, and Element.