



pinTab Manual

Version 2.0

www.pintexx.com

Table of contents

1	Introduction	3
1.1	Fundamentals	3
2	Integration/Deployment	4
3	Tab modes	5
3.1	Standard mode.....	5
3.2	Multiple rows	5
3.3	Scrolling mode	5
4	Using pinTab	6
4.1	Integration on a HTML page.....	6
4.2	Creating tab items	6
5	Caching pages	8
6	pinTab properties	9
6.1.1	Properties of tab object	9
6.1.2	Properties of tab item	10
6.1.3	Properties of separator item	10
7	pinTab Events	11
7.1	TabClick	11
7.2	BeforeTabClick.....	11
7.3	Loaded	11
7.4	PageLoaded.....	11

1 Introduction

pinTab gives you the possibility to create web based applications with the same look and feel as desktop applications.

The navigation area allows selecting between different tabs that contains any web page.

The split area can be loaded from the navigation area or accessed to modify content.

1.1 Fundamentals

pinTab consist of pure DHTML/Javascript and doesn't need ActiveX/Applets or Plug-Ins.

pinTab allows easy tab creation with high quality design especially in IMAGE mode.

To make the design easy the tab area is not based on HTML content but on URL.

That makes it easy to use any web page like HTML, ASP, JSP or others.

2 Integration/Deployment

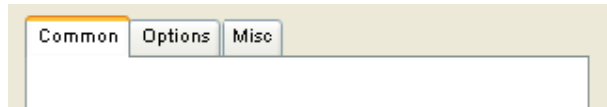
For integration in a web application the following steps has to be done.

- Copy the “tab” directory in your web applications directory
or
in a folder of the same domain

3 Tab modes

3.1 Standard mode

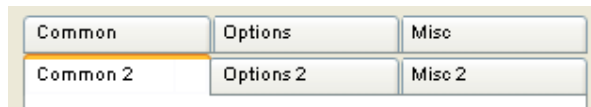
In standard mode there is one row of tab items without scrolling.



3.2 Multiple rows

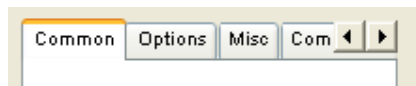
In this mode the tab control can have multiple tab rows.

Please specify the maximum number of tab items per row and the control automatically determines the number of tab rows.



3.3 Scrolling mode

In the scrolling mode the width of tab items can exceed the tab control width. The scroll buttons allow scrolling in both sides.



4 Using pinTab

pinTab consist of pure DHTML/Javascript and doesn't need ActiveX/Applets or Plug-Ins.

4.1 Integration on a HTML page

To use pinTab on a HTML page simple create a iframe tag and load the pinTab.html.

```
<iframe id="tab" src="../../tab/pinTab.html" style="Z-INDEX: 101; LEFT: 23px;
POSITION: absolute; TOP: 57px;Width:232px;Height:367px" frameborder=0></iframe>
```

The src tag points to the pinTab.html in "tab" directory.
Please specify top, left, width and height within the style tag.

Additionally create a JavaScript function called "tabCreate" with 2 parameters in the script section:

```
<head>
<script>
// this function is called if navigator is ready
function tabCreate(objTab,id)
{
    // put your code here to create the tab
}
</script>
</head>
```

This function is called from pinTab during startup.

The parameter "objTab" is the empty tab object to which items can be added and properties set.
The id parameter (optional) is the id of the current iframe container, so that you can decide between several pinTab.

4.2 Creating tab items

Use the "objTab" object to create items.

```
function tabCreate(objTab)
{
    // set the design to Windows XP like
    objTab.design      = "1";
    // orientation is top
    objTab.orientation = "0";
    // we want a tab area
    objTab.tabarea     = true;
    // we want images
    objTab.designmode  = "IMAGE";

    // create a new tab item object
    var item = objTab.createItem();
}
```

```
// set the tab text
item.text = "Common";
// set the title
item.title = "Common";
// set it to active
item.active = true;
// set the URL
item.url = "../evaluation/p1.html";
// add the item
objTab.add(item);

// add more tabs
// ...

// create a separator object
var item = objTab.createItem("TabSeparator");
// specify the width in pexel
item.width = "200";
objTab.add(item);

// create the tab
objTab.create();
}
```

5 Caching pages

A lot of tab controls are based on defining HTML content for the tab areas. pinTab allows defining any URL for each tab area.

To avoid reposting and to allow fast tab switching the tab pages are cached internally (by default). This caching is independent of browser settings.

The advantage is, that

- repeatedly repostings are avoided (saves server performance)
- the current status of a page is retained
 - o Scroll positions are not lost during tab switch
 - o Changes within pages like open tree nodes or entered values are not lost

The caching can turned off by using the “cached” property of the “Tab” object.

6 pinTab properties

The following properties are available for tab object, tab item and tab separator.

6.1.1 Properties of tab object

Method	Description
design	Specifies the current design (a number from 1-n) IMAGE mode: "1" Windows XP like "2">Office 2000 "3">Office 2003 like "4">3D "5">Mac like COLOR mode: "1" Inset - Gray "2">Inset - Blue "3">Inset - Red "4">Inset - Green "5">Inset - Pink "6">Outset - Gray "7">Outset - Blue "8">Outset - Red "9">Outset - Green "10">Outset - Pink
orientation	"0" top "1" bottom
tabarea	True/false Specifies if tab area
designmode	"IMAGE": use images "COLOR": use only colors
standardstyle	Specifies the standard style in COLOR mode
hooverstyle	Specifies the hoover style in COLOR mode
activestyle	Specifies the active style in COLOR mode
separatorstyle	Specifies the separator style in COLOR mode
backgroundstyle	Specifies the background style in COLOR mode
bodystyle	Specifies the body style in COLOR mode
textstyle	Specifies the text style in COLOR mode
activetextstyle	Specifies the active text style in COLOR mode
New in 2.0:	
loadOnStartup	If true, all pages are loaded during startup. If false, only the active page is loaded
refreshAfterLoad	If true then the page is refreshed after page
tabMode	0 = Standard 1 = Multiple Rows 2 = Scrolling mode
maxTabItemsPerRow	Needed for tabMode=1. Specifies the maximum number of tab items per row
textAlignment	0 = left

	1 = center 2 = right
imageAlignment	0 = left 2 = right
getItemByIndex (value)	Returns the tab or separator item with the given index
getTabItems ()	Returns only tab items (no separators)
getTabItemByIndex (value)	Returns tab item with given index

6.1.2 Properties of tab item

Property/Method	Description
text	Tab item text.
image	URL of tab item image (relative to tab directory or absolute)
active	True/false Specifies if the tab item is active or not. Please set only one item to active or use the <code>setActiveTab</code> method.
url	URL of the tab page (relative to tab directory or absolute)
title	Tooltip
cached	True/false Specifies if the tab page is cached or not
New in 2.0	
tab	Reference to parent object (tab object)
tag	For any purpose
index	Index starting from 0
visible	(Read only). Determines if tab item is visible
enabled	(Read only). Determines if tab item is enabled
setVisible (value)	Sets a tab item visible/unvisible
setEnabled (value)	Sets a tab item enabled/disabled
refresh ()	Refreshes the tab item page
getDocument ()	Gets the document object of the tab item page
getWindow ()	Gets the window of tab item page
call ("function")	Calls a Javascript function on the tab item page

6.1.3 Properties of separator item

Method	Description
Text	Separator text (left aligned)
Width	Width of separator in pixel (default value is 100%)

7 pinTab Events

7.1 TabClick

To get an event if a tab item is clicked please add the following function to your page:

```
// this function is called when an item is clicked
function tabEventTabClick(objItem)
{
    alert("Item clicked: " + objItem.text);
}
```

objItem is the current tab item

7.2 BeforeTabClick

If you specify this function on the page it will be called from pinTab after the user pressed a tab item and before a TabClick event is initiated.

If you return true the TabClick event is not executed.

```
// add this function and it will be called after pressing a tab header
// item is the clicked tab item
function tabEventBeforeTabClick(lastTabItem)
{
    alert("Last item: " + lastTabItem.text)
    // do something

    // to abort return true, then tab is now switched
    // return true
}
```

7.3 Loaded

If all pages have been loaded this event is raised.

id is the id of the tab control.

```
// event is rasied when all pages are loaded
function tabEventLoaded(id)
{
    // add your code here ...
    alert("All pages loaded !")
}
```

7.4 PageLoaded

If a page is loaded then this event is raised.

id is the id of the tab control.

```
function tabEventPageLoaded(item, id)
{
    alert("Page loaded: " + item.text);
}
```