



User's Guide

Likno Web Tabs Builder

The Complete Tool for Creating Powerful Web Tab Controls

by Likno

Likno Web Tabs Builder is a user-friendly interface (GUI) to the popular jQuery open source library, that allows you to create stylish, feature-rich html tab controls for your web pages, with minimal effort and coding.

All the jQuery tabs it creates are cross-browser and work alike in all browsers.

The present manual will assist you in taking the most out of this software.

The Likno Software team

Likno Web Tabs Builder User's Guide

© 2000-2014 Likno Software, all rights reserved

All rights reserved. No parts of this work may be reproduced in any form or by any means - graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems - without the written permission of the publisher.

Products that are referred to in this document may be either trademarks and/or registered trademarks of the respective owners. The publisher and the author make no claim to these trademarks.

While every precaution has been taken in the preparation of this document, the publisher and the author assume no responsibility for errors or omissions, or for damages resulting from the use of information contained in this document or from the use of programs and source code that may accompany it. In no event shall the publisher and the author be liable for any loss of profit or any other commercial damage caused or alleged to have been caused directly or indirectly by this document.

The javascript code used by this software application is property of Likno Software. You may use this code only as part of this software for creating web projects. Any unauthorized use, transmission, transformation, facsimile, or copying of any means (electronic or not) is strictly prohibited.

Removal of the present copyright notice is prohibited.

Printed: 15-Jan-14

Special thanks to:

We would like to thank all of our users and customers that have contributed their comments and suggestions for improving Likno Web Tabs Builder and the services we offer. It is our aim to listen to all of them carefully in order to bring the best possible result for them. We couldn't have done without you!

Table of Contents

Part I Introduction	2
1 What is Likno Web Tabs Builder	2
2 What's new	3
3 Terminology	3
4 About Likno Software	6
Part II A Tab control in 5 easy steps	8
1 Step 1: Select the content and properties of the tab control	8
2 Step 2: Select the Project Style of your tab control	8
3 Step 3: Compile and Link the Project	8
4 Step 4: Upload the files to your server	8
5 Step 5: Save your work	9
Part III User Interface	11
1 Main window layout	11
2 Keyboard shortcuts	11
Part IV Commands	14
1 Command: Project	14
2 Tools	14
Command: Compile Project	14
Command: Link Compiled Project to Web Pages	15
Command: Project Properties	16
Compile Properties.....	16
Preview Panel Settings.....	18
3 Command: Help	19
Part V Properties	22
1 Properties Overview	22
2 "Project Style" Properties	22
Tabs Show - How	22
Tabs Show - When	22
Tabs Hide - How	23
Tabs Hide - When	23
Initially Selected Tab	23
Play Tabs	26
Container Width/Height	30
Container CSS	30
Headers CSS	31
Body CSS	37

Ignore header-related CSS	37
3 "Project Tabs" Properties	38
Tab Control Populated By	38
Project Type: Headers Within/Outside Container	42
Headers Within Container (Project Type 1 of 2).....	43
Headers Outside Container (Project Type 2 of 2).....	46
Header Source options	49
Is Separator.....	49
Embedded (for header).....	50
Image (for header).....	51
Body Source options	52
Embedded.....	52
Image	52
AJAX	53
External	53
Container Positioning	53
 Part VI How Do I...	 56
1 Find my INSTALLATION ID	56
2 Purchase/Activate	56
3 Add effects when a tab body shows or hides	56
4 Change the position where the tab control is located on page	57
5 Load with AJAX the content of a tab body	57
6 Load an external site as the content of a tab body	57
7 Use multiple Tab projects in the same page	57
8 Properly use paths in the "Website Folder" option	58
9 Display a video/movie in a Tab?	59
10 Use the "background-image" value in CSS properties	60
 Part VII What is...	 64
1 The HTML Linking Code	64
2 The Internet Explorer Information Bar Warning	64
3 The "Colors detected in this CSS" feature	66
4 The "Relative paths detected in this HTML" feature	67
 Part VIII FAQ and Tips	 70
1 Troubleshooting	70
2 "Full Admin rights" and the activation process	70
 Index	 72

Introduction

Part



1 Introduction

1.1 What is Likno Web Tabs Builder

[Create Web Tabs in 5 easy steps!](#)

Likno Web Tabs Builder is an application that serves as an interface to the popular **jQuery** open source library, allowing you to create various "HTML Tabs" for web pages with minimal effort and coding.

It creates **cross-browser** jQuery tabs that work alike in all browsers.

Through our builder you can easily create web tabs with many options, like showing content added easily through the interface (embedded), external or internal (AJAX) pages, and more.

The produced tabs are fully cross-browser, i.e. they work alike in all browsers supporting DHTML (like Firefox, Internet Explorer, Google Chrome, Opera, Mozilla, Konqueror, Safari and more).

Likno Web Tabs Builder will save you a significant amount of time and effort required to fully understand DHTML and Javascript code capable to support all browsers for creating powerful web tabs!

Major Features Include:

- * Rich and powerful interface!
- * Create cross-browser jQuery tabs to display content when visitors click on (or even mouse over!) page elements (headers) of your choice.
 - * Use simple text, HTML code, images, videos (YouTube, etc.), animations, Flash, or whole pages (AJAX, URL) as the content of your tabs.
 - * Create projects fast by using ready-made examples already installed in the application.
 - * Generic: Use in all authoring tools (Dreamweaver, Expression Web, etc.) and CMS platforms (Joomla, Drupal, Wordpress, etc.).
 - * Create any type of jQuery tab control, such as: html tab, javascript tab, css tab, ajax tab, web tab, vertical tab, horizontal tab, dhtml tab, jquery ajax tab, etc.
 - * Use any of 3 methods to populate your tabs with content:
 1. Tab Control is populated by the project properties (within the application)
OR
 2. Tab Control is populated by a UL/LI structure, located inside your page.
OR
 3. Tab Control is populated by DIV elements, located inside your page.
 - * Position your tab controls on your pages easily and anywhere you like:
 1. Insert into a page element of your choice (div, span, td, etc.).
OR
 2. Replace a page element (instead of "inserting into" the element).
OR
 3. Replace a page image of your choice (just use a "placeholder" image).
 - * Use any of 2 methods to place your tab headers relative to the tab body:
 1. Within Container (standard, all headers are "glued" to the tab body)
OR
 2. Outside Container (*advanced*, headers are "not glued" but are free DIV elements *anywhere* on page).
 - * Customize the style of your tabs through CSS ("css tabs").
 - * Apply images creatively to create interesting tabs (images on tab headers, etc.).

- * Use multiple tab controls on the same page (with the same or different styling).
- * Add dynamic content (database-driven or programmable) to your tabs through AJAX techniques ("ajax tabs").
- * Show and hide your tabs using a variety of show/hide effects.
- * Set a specific tab to show on page load.
- * Use complete help for all properties and procedures. Press F1 while you are on a property to immediately receive Help for that property.
- * Access additional designs (examples) of interesting tab controls, besides the designs coming with the application (note: applies to "Unlimited Domains" users).
- * Powered by the popular jQuery open source library.

...and many more (view the [complete properties listing](#) for more info).

1.2 What's new

Please check the new features, enhancements and fixes of the application by visiting the following page online:

<http://www.likno.com/jquery-tabs/whatsnew.php>

1.3 Terminology

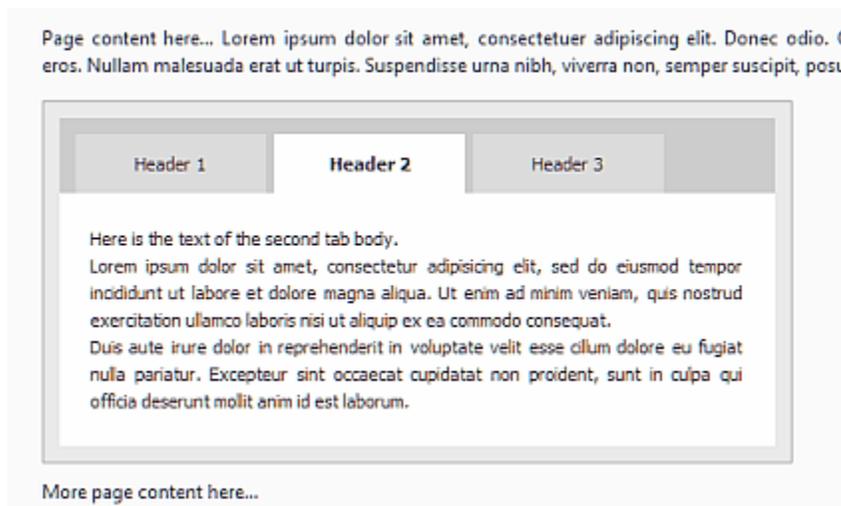
Tab Project

All tabs contained within a Project share the same "Project Style" properties (located in the "Style Editor" tab).

For separate tab projects (tab controls) within the same page, read more information [here](#).

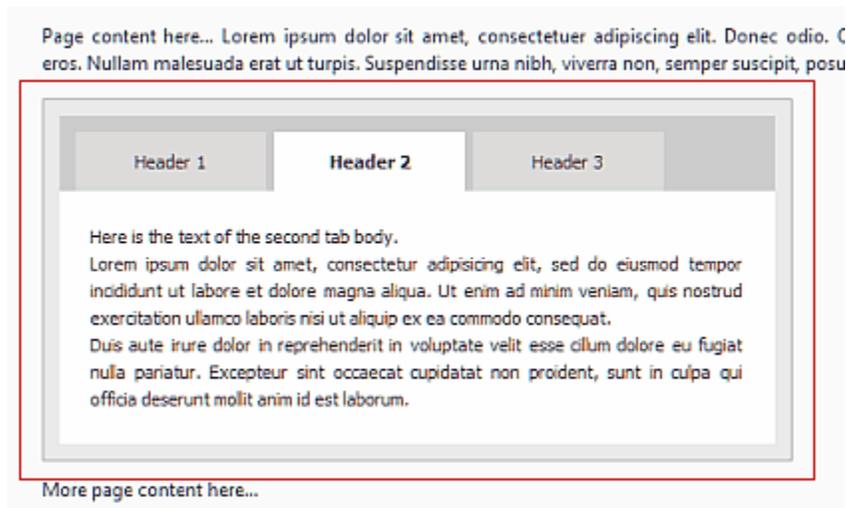
Tab Control

Your entire project is a tab control (all tabs within a "container"):



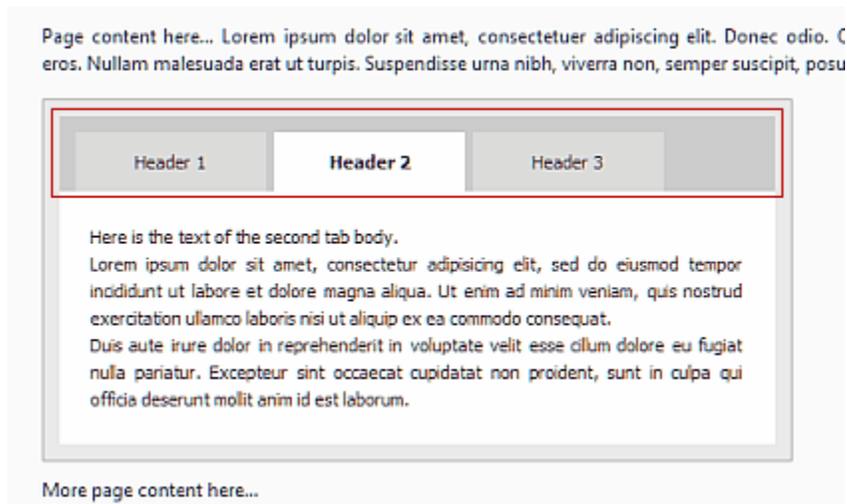
Tab Container

This is the area that contains all tabs (their "headers" and "bodies"):



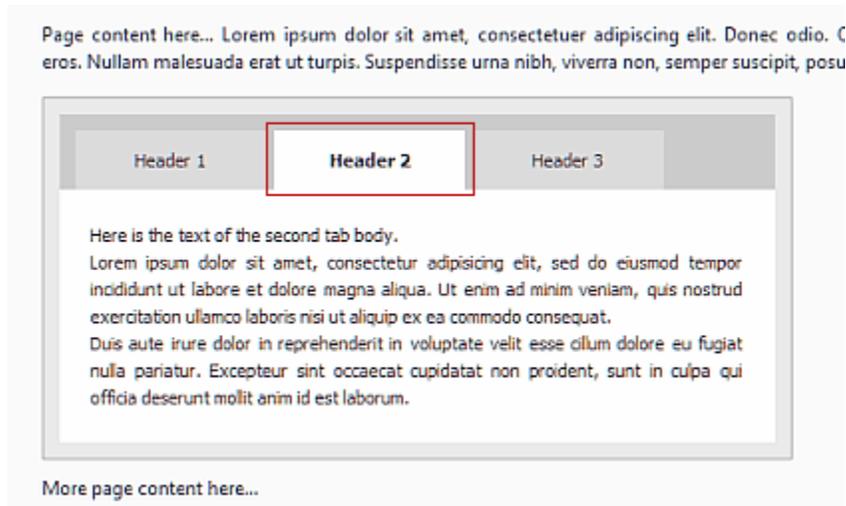
Tab Header Area

This is the area that contains all headers of the tabs:



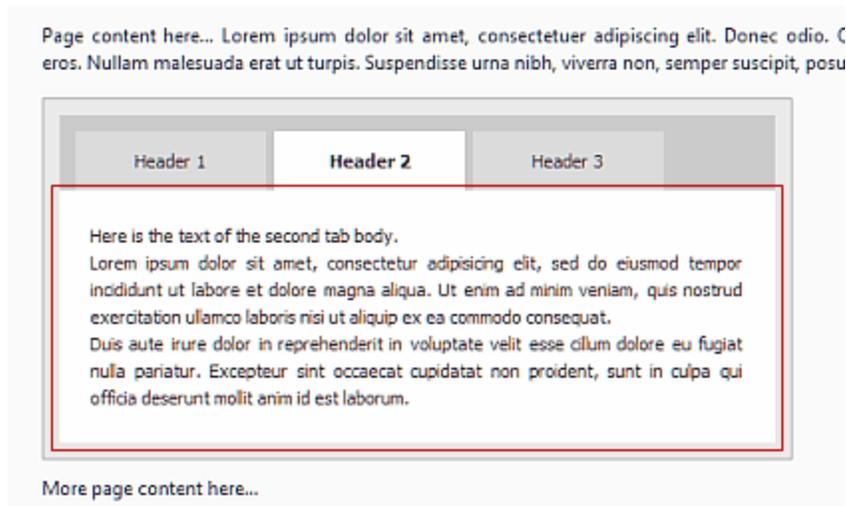
Tab Header

This is the area that shows the "title" of each tab and functions as the selector of the tab:



Tab Body

This is the common area that contains all bodies of the tabs (each "body" shows different content for each "header"):



Tab Content

The Content of the tab is the information that displays. It consists of the "Header" content and the "Body" content. This content can be from a simple text, to rich HTML content, and even a whole new page derived either as an AJAX call or as an external website.

More information [here](#).

Project Style

The style of the tabs is their appearance (colors, background, borders, etc.) and behavior (opening/closing effects, etc.).

1.4 About Likno Software

Likno Software is a software development company dedicated to creating quality software for demanding web authors. Our development team is 100% internet-oriented, focused on creating tools that minimize your web development efforts.

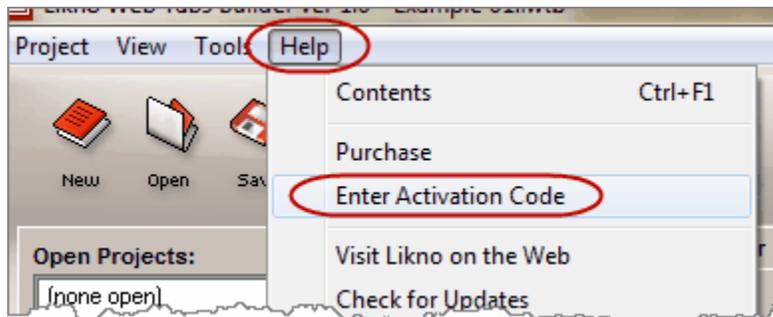
For detailed **technical support** you can access the following URL:

http://www.likno.com/support_page.html

For **sales** questions you may contact us at sales@likno.com.

Our web site is located at <http://www.likno.com>

Note: In case you came here to **find your Installation ID**, it can be found at the "**Help -> Enter Activation Code**" command of the **application menu**, as shown below:



A Tab control in 5 easy steps

Part



2 A Tab control in 5 easy steps

2.1 Step 1: Select the content and properties of the tab control

Step 1

In the **<Project Tabs>** tab you can find all the content properties for each tab individually ("header" and "body" of the tab).

All properties are described and explained in the "[Properties](#)" help section.

Note: If you choose the **UL/LI** or **DIV** population option, don't forget that you should also add the UL/LI elements or DIVs in your actual web page!

2.2 Step 2: Select the Project Style of your tab control

Step 2

In the **<Style Editor>** tab, choose the style properties that will apply to the "container", all "headers" and the "body" of the tab project.

All settings are described and explained in the "[Properties](#)" help section.

2.3 Step 3: Compile and Link the Project

Step 3

Compile:

Click on the "Compile" button and select the location where you want to save your compiled JavaScript files.

Link:

Once you have compiled your project, use the "Link" button to retrieve the necessary code that must be added to your pages in order to link the compiled files to them.

Make sure that the correct path to the source javascript file has been used, otherwise the tabs will not appear.

2.4 Step 4: Upload the files to your server

Step 4

Once you have compiled and linked your compiled javascript project to your pages, you will need to

upload the files to your web server. The files created by Likno Web Tabs Builder during the compilation phase are:

```
./likno-scripts/                (folder)
./likno-scripts/likno-tabs-project.js  *
./likno-scripts/likno-tabs-lib.js
./likno-scripts/jquery.js
./likno-scripts/likno-tabs-project/    * (folder)
(all image files used by the tabs)
```

where "." is the "Site_Root" folder (also specified in *Project Properties*).

(* the highlighted names may be different if you changed the "Compiled Javascript Project" name in *Project Properties*)

All these files and folders have to be uploaded to your web server using your favorite FTP tool. Make sure that the directory structure is kept exactly the same as is displayed here and also pay careful attention in order to keep the same case for the files when they are uploaded. You have also to make sure that the files are placed in the same location relative to your HTML files as they are locally, in order for the directories that were set while linking your compiled project locally to point to the right structure on your web server.

2.5 Step 5: Save your work

Step 5

Your Web Tabs can be saved in a **.lwtb** project file so that you can use/edit it in the future. Just use 'Project->Save' or 'Project->Save As' for this process.

Once your file is modified you will need to [compile it again \(but not link it\)](#) and upload the new **likno-tabs-project.js** file to your web server. The **likno-tabs-lib.js** file needs to be re-uploaded only if you re-compile your project with a newer Likno Web Tabs Builder version (since it may use an updated engine).

User Interface

Part



3 User Interface

3.1 Main window layout

Likno Web Tabs Builder uses several tabs:

Project Tabs tab (F2)

Here you can add tabs ("Headers" and their "Body") to your single tab project and specify their content and positioning.

Style Editor tab (F3)

Here you can modify the style properties of the tab control of the loaded project.

Sample Code tab (F4)

Here you can view sample HTML code that shows the tab control in the Preview Pane, so that you understand what code needs to be added to your own pages.

Preview tab (F5)

Here you can preview the tab control that you have specified in the "Project Tabs" tab.

Project Properties tab (F6)

Here you can specify the compiled files and folders of the project, as well as the domain for which the project should be licensed and functional.

3.2 Keyboard shortcuts

There are several shortcuts defined for quicker access to frequently used functions:

- **F1**
Provides context sensitive help for each property
- **F2**
Opens the Project Tabs tab
- **F3**
Opens the Style Editor tab

- **F4**
Opens the Sample Code tab
- **F5**
Refreshes and opens the Preview tab
- **F6**
Opens the Project Properties tab
- **Ctrl + F12**
Opens the Compile dialog
- **F8**
Compiles the project without any dialogs
- **Shift + Ctrl + F12**
Opens the "Link Compiled Project to Web Pages" dialog
- **Ctrl + N**
Creates a new project
- **Ctrl + O**
Opens a .lwtb project
- **Ctrl + S**
Saves the selected project

Commands

Part



4 Commands

4.1 Command: Project

New

Creates a new **Likno Web Tabs** project.

Open

Opens a Likno Web Tabs project.

The Likno Web Tabs files are found under the **.lwtb** extension.

Save

Saves the selected project into a ***.lwtb** file.

Save As

Saves the current project with a different filename.

Save All Projects

Saves all loaded projects, but not the examples which are read-only.

Close Selected Project

Closes the selected project.

Close All Projects

Closes all loaded projects, but not the examples which are always available.

Export Project

Exports the loaded project as a ZIP file in order to move it to other PCs without having problems with the file paths of the linked images (the "assets" folder).

Exit

Closes the **Likno Web Tabs Builder** application.

4.2 Tools

4.2.1 Command: Compile Project

Compiles the **.lwtb** project into a **.js** file that has to be linked to your web pages.

To use a tab control in your web pages you must take the following four steps:

Step 1. **Compile** the project (note: spaces are not allowed in filename)

Step 2. [Link](#) the compiled project to your web pages

Step 3. **Upload** the necessary files and folders (see a description of these elements below)

Step 4. **Upload** the web pages that contain the project link

(note: steps 2 & 4 are not necessary if you have already linked the project to your web pages and you only want to update your tab control)

How it works:

When compiling a project as *likno-tabs-project.js*, the following elements are created inside the same directory:

1. The core **likno-tabs-project.js** javascript file.
2. The "Compiled Libraries" and "Compiled Images" folders as defined in [Project Properties](#).

It is critical that **you do not forget to upload the *likno-scripts* directory**, where these files are compiled.

See also

[Link Compiled Project to Web Page\(s\)](#)

4.2.2 Command: Link Compiled Project to Web Pages

Links a compiled project (.js file) to a selection of web pages.

At this stage, the application inserts the necessary "Linking Code" into the selected web pages. No DHTML or Javascript knowledge is required since the whole process is performed by the application. The link is a few lines of code and can be easily identified inside the source code by a related comment.

The linking process requires only two pieces of information:

1. The **compiled project file** (eg: likno-tabs-project.js)
2. The **web page(s)** where the link will be added.

Obviously, you need to **upload** the updated web pages on the server so that the tab control can be viewed on-line. However, it is important that you also upload:

1. the compiled project file (eg: *likno-tabs-project.js*)
2. the *likno-scripts* folder
3. the *likno-scripts/likno-tabs-project* folder if you use images. Your folder may be different depending on the name you choose to use for the compiled project file.

Upload the above at the **same** directory structure where they reside locally.

See the [Compile Project](#) command for more info on this issue.

To update the design of an already compiled project you should first make the changes to its respective .lwtb file and then recompile it. Provided that you did not change the name or the directory

of the compiled project, you do not need to re-link the project to the pages that already contain that link.

Notes:

Note about file extensions:

The link can be inserted in any kind of page, regardless of its extension. Just use the All Files (*.*) filter to select web pages that have an extension other than the typical .html or .htm one.

Note for Dreamweaver users:

If you like, you can also link the compiled project to a template (.dwt) or a library (.lbi). Just select the related directory (Templates or Library) and use the All Files (*.*) filter so that you can view the template or library file.

The library (.lbi) is a special Dreamweaver file containing a collection of individual assets or copies of assets that you can place in your web pages. That means the linking code can be added in a library but you should make sure that the library should be placed as the first element after the BODY opening tag (<body>) and that the linking code is the first element, listed in the library file itself.

The project does not have to be compiled inside the Templates directory. On the contrary, we do not advise you to do so as you will have to upload the Templates directory on the server too.

Note for FrontPage users:

Similarly to Dreamweaver, you can link your project into an **include** so that it can be used by all other pages. Just make sure that this include contains only the project and that it is inserted immediately before the </body> tag of the pages.

Similar techniques can be used with other web authoring tools too.

See also

[Compile Project](#)

4.2.3 Command: Project Properties

Allows you to specify the following properties and settings for your loaded project:

[Compile Properties](#)

[Preview Panel Settings](#)

4.2.3.1 Compile Properties

Allows you to specify the following **compile properties and options** for your loaded project:

The **Site_Root folder** is the location where your web site (your main index page) is locally stored. It is also referred as the "Site Root" folder in other applications. It is very important that you specify the correct folder here as it is used in the compile process.

The **Compiled Javascript Project** is the javascript file generated when you compile your project. It defaults to *likno-tabs-project.js* but you may as well change it to something else. Note that you can use multiple tab controls in the same website by having different projects compiled using different "Compiled Javascript Project" names.

Once you setup your "Site_Root folder", you are able to edit the "**Compiled subfolders**":

The **Compiled Libraries subfolder** (default: \likno-scripts) which contains the following:

likno-tabs-lib.js The **required** Javascript tab-building Libraries. These are the javascript files containing all the functions that build the tabs and drive their behavior.

likno-tabs-project.js The compiled project file (actual name depends on your settings)

There is no apparent reason to change the libraries folder, so we recommend that you keep the default value.

The **Compiled Images subfolder** (default: \likno-scripts\likno-tabs-project) which contains any image files used by your tabs. If not uploaded, the tabs will show no images. The missing image sign ("x") may also appear in place of those images.

Custom folders may be required when two or more different tab projects are used in the same website and images from both projects use the same names. This is the usual situation that may require you to have a different subfolder for the produced images.

Custom folders may contain **relative** paths too (eg: ../). If you do so, the folder location will be relative to the location of the compiled **likno-tabs-project.js** file.

The **Domain (website) where the compiled project will be used** is very important as it affects whether your tab control is available to your online visitors or just you (for testing purposes).

You can always view (test) your tab control by using a **localhost** or **numeric IP** on the browser but not when using a "**letter-based**" domain (eg: www.fifa.com), **unless** you have **purchased a domains license** and allocated a domain to that license.

For example, "fifa.com" uses the "83.231.250.129" numeric IP.
(find the numeric IP of any domain name here: <http://www.hcidata.info/host2jp.cgi>)

FIFA developers could then test their tab control using any of the following:

c:/portfolio/fifa_site/page.html (local hard drive)

http://localhost/page.html (localhost)

http://83.231.250.129/page.html (their numeric IP)

But their actual visitors could **not view** the tab control when using the respective "letter-based" domain:

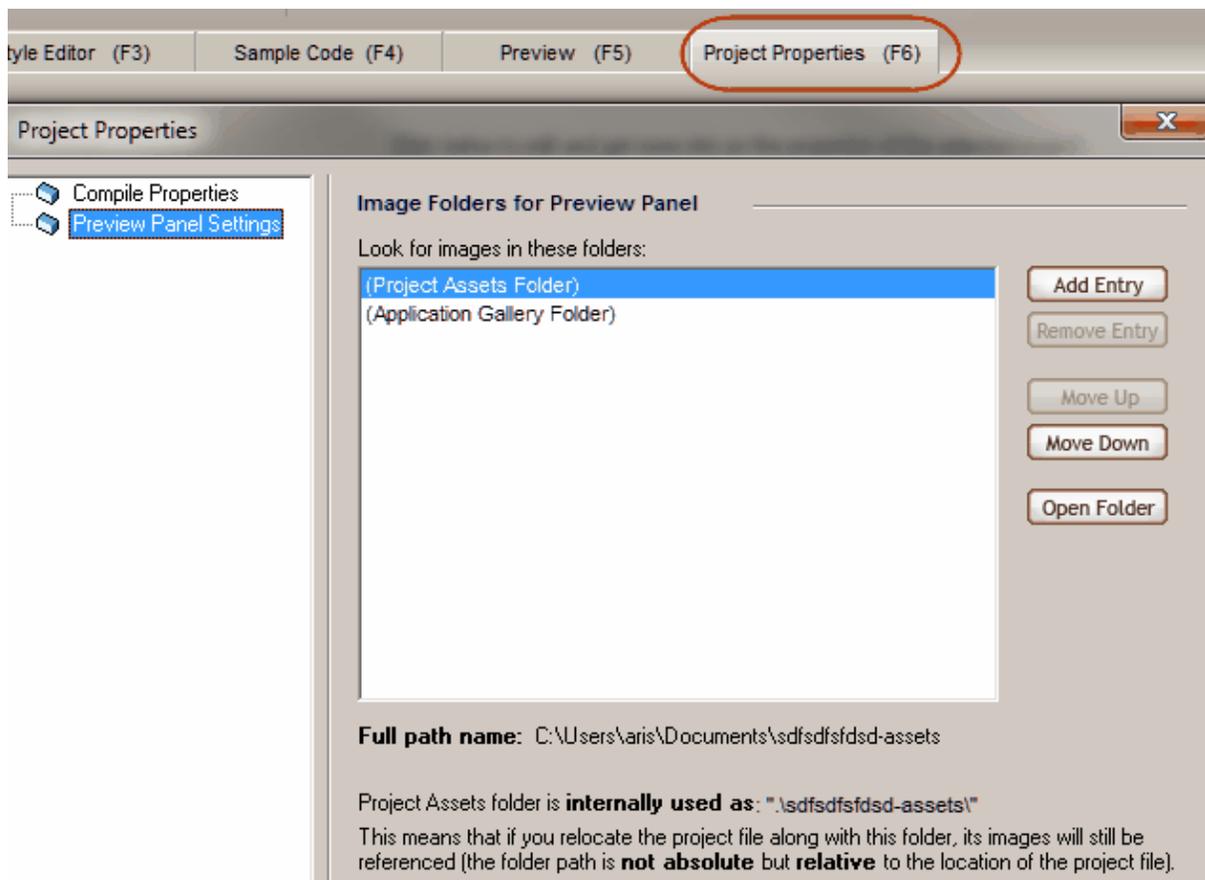
http://www.fifa.com/page.html (domain)

So, what FIFA could do is purchase the "3-Domain" license, **allocate "fifa.com"**, get an Activation Code and choose to compile for the "fifa.com" domain. Or simpler, purchase the **"Unlimited Domains"** license, get an Activation Code and just compile for ANY domain. Any of these 2 options would make the tab control available for the "fifa.com" online visitors.

Note on "subdomains": Do not allocate subdomains, just allocate their major domain, as **10 of its first-level subdomains** will be supported by default (examples: "**subdomain.fifa.com**", "**subdomain2.fifa.com**", but **not "sub.subdomain.fifa.com"**). For more than 10 subdomains (or second-level subdomains) on the same domain, you need to purchase the "Unlimited Domains" License.

4.2.3.2 Preview Panel Settings

Allows you to specify folders so that the Preview Panel can show images (if used) that are **relative to the page or site root**.



While these images will show online (on the actual site), the **local Preview Panel** cannot show them, unless you specify local folders to browse from.

For previewing purposes, the Preview Panel locates graphics on the disk by searching for files in the directories in the **"Image Folders for Preview Panel"** list.

By default the "Project Assets Folder" and the "Application Gallery Folder" are included in the Image Folders list so that any images in those folders will be found automatically. You cannot remove them from the list.

Please note that you must be careful to **avoid duplicate graphics file names in different folders** (the standard folders or yours). If this occurs, the 1st match will "win" depending on how the order of folders is (use the "move up/down" buttons to determine the priority of folders).

4.3 Command: Help

[Contents](#)

Displays the contents of the help file. It also supports both keyword and full-text search.

[Purchase](#)

Transfers you online, to the order page.

[Enter Activation Code](#)

This is where you can enter an Activation Code. You will receive this Activation Code through your Likno User Area account which is created after your purchase of a license for the application.

[Visit Likno on the Web](#)

Visit our web site to contact us, get the latest information on new versions and products.

[Check for Updates](#)

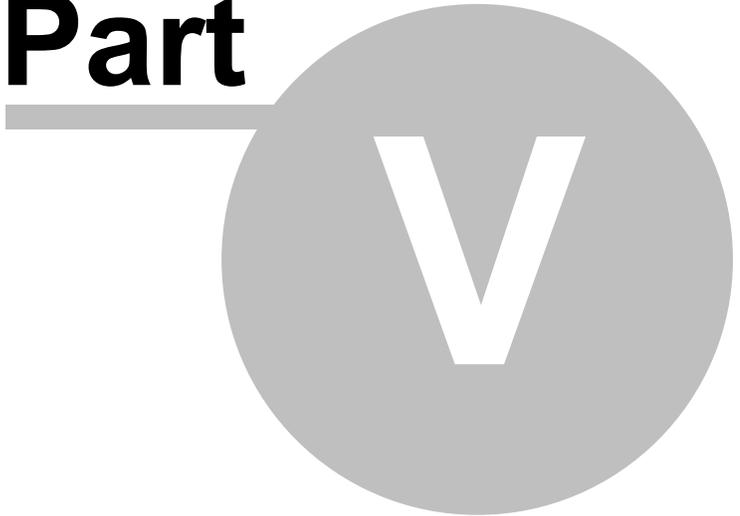
Checks if newer versions have been released since the last time you installed the application. It is recommended that you check for updates frequently as we regularly release improved cross-browser support for your implementations.

[About](#)

Displays the About box of Likno Web Tabs Builder, with your license information.

Properties

Part



5 Properties

5.1 Properties Overview

Properties are classified in the following categories:

The **Project Tabs** properties, which specify the content and positioning of the tab control of the loaded project:

Project Type	("Within" or "Outside" Container)
Tab Control Populated By	(Project, UL/LI structure, DIVs)
Header Source	(for each tab of the tab control)
Body Source	(for each tab of the tab control)
Container Positioning	
Headers Positioning	

The **Project Style** properties, which specify the appearance and behavior of the tab control of the loaded project:

Tabs Show - How
Tabs Show - When
Tabs Hide - How
Tabs Hide - When
Initially Selected Tab
Container Width/Height
Container CSS
Headers CSS
Body CSS

5.2 "Project Style" Properties

5.2.1 Tabs Show - How

Choose a show effect for the tab bodies of your project, along with its duration in milliseconds (ms).

See also

Property: Tabs Hide - How
Property: Tabs Hide - When
Property: Tabs Show - When
Property: Initially Selected Tab
Property: Play Tabs

5.2.2 Tabs Show - When

Choose if the tab bodies show when the mouse is "over" their headers or when you "click" on them.

See also

[Property: Tabs Show - How](#)
[Property: Tabs Hide - How](#)
[Property: Tabs Hide - When](#)
[Property: Initially Selected Tab](#)
[Property: Play Tabs](#)

5.2.3 Tabs Hide - How

Choose a hide effect for the tabs of your project, along with its duration in milliseconds (ms).

See also

[Property: Tabs Show - How](#)
[Property: Tabs Hide - When](#)
[Property: Tabs Show - When](#)
[Property: Initially Selected Tab](#)
[Property: Play Tabs](#)

5.2.4 Tabs Hide - When

Choose when the tab body hides.

By default, when another tab gets selected, the current tab body hides (as the other one shows).

You also have the additional option to "collapse" ("close") the currently selected tab. In that case, the tab control does not the body section but only its headers. If you click on any header, its body will show again.

See also

[Property: Tabs Show - How](#)
[Property: Tabs Hide - How](#)
[Property: Tabs Show - When](#)
[Property: Initially Selected Tab](#)
[Property: Play Tabs](#)

5.2.5 Initially Selected Tab

Choose which tab ('header' and its 'body') shows as initially selected (when the page loads).

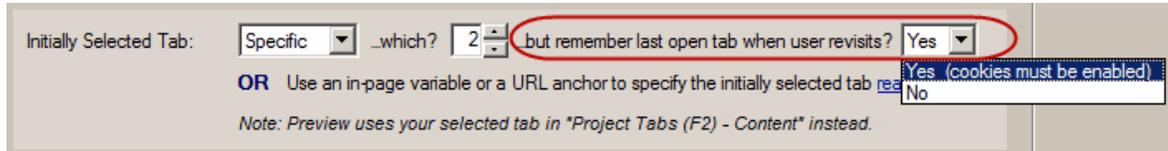
Method 1: Specify this inside the project (through the interface)

Just enter the "order" of the selected tab here:



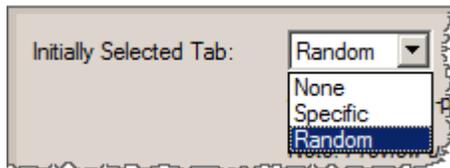
Please note the the application's "Preview" will not necessarily show that tab as selected, as it uses your selected tab in "Project Tabs (F2) - Content" instead. This is done on purpose so that you are facilitated in previewing your currently selected tab when editing it. Obviously, your online page will respect the "Initially Selected Tab" value.

You may also use the following option for those users that revisit the same tab control:



In that case, these users will see the tab control at the state they left it during their previous visit. This means that the "Initially Selected Tab" value will not have any effect if the user had changed the tab selection in the previous visit.

You may also choose a "random" tab to show as initially selected:



Other methods that specify the initially selected tab **externally** (overrides the project value):

Method 2: Use a javascript variable in the page that contains the tab control

In the HTML code of your page add the following code:

```
<script>var projectcompiledname_selected_tab=3;</script>
```

where:

- projectcompiledname is the compiled name of your tab control
- 3 is the order of tab that should be initially selected (3rd tab in this example)

For example, for this project...



...the proper code for showing the 3rd tab is the following:

```
<script>var my_tab_control_selected_tab=3;</script>
```

Important! If you use dashes ('-') in your project name, you need to make them underscores ('_') when using the compiled name in the variable.

For example, if your project name is *likno-tabs-project.js*, the variable should be as follows:

```
<script>var likno_tabs_project_selected_tab=3;</script>
```

Method 3: Use a special anchor in the URL that opens the page

Open the page that contains the tab control using a special "anchor" on its URL:

`http://www.mysite.com/mypage.html#projectcompiledname_3`

For example, for this project...



...the URL for showing the 3rd tab is the following:

`http://www.mysite.com/mypage.html#my_tab_control_3`

Notes on "Method 3" (anchor in URL):

- this works only when the page is opened by a link from **another** page, not from the same page (because most browsers do not refresh the page when the link is from the same page).

- you can also add **multiple anchors** in the same URL! This is useful if you have multiple tab controls in the same page and you want to specify their initially selected tab (for each control separately).

Example:

`http://www.mysite.com/mypage.html#my_tab_control_3#second_tab_control_5#third_tab_control_2`

- you can even add multiple anchors in the same URL to specify initial parameters **for other Likno controls** that exist in the same page, such as accordions, modal windows, etc.!

Example:

`http://www.mysite.com/mypage.html#my_tab_control_3#my_accordion_control_3#my_modal_window_1`

(for more info on their usage read the respective Help section of Likno Web Accordion Builder, Likno Web Modal Windows Builder, etc.)

See also

[Property: Tabs Show - How](#)
[Property: Tabs Show - When](#)
[Property: Tabs Hide - How](#)

[Property: Tabs Hide - When](#)
[Property: Initially Selected Tab](#)
[Property: Play Tabs](#)

5.2.6 Play Tabs

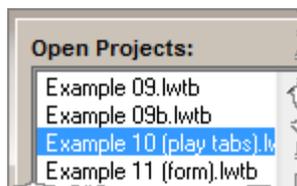
Choose if the tab control behaves as a "slideshow", meaning that its tabs show automatically (one at a time) after some seconds.

The first tab that shows is whichever you choose at the [Initially Selected Tab](#) property. If "0" is used there (which means that no tab is initially "selected"), then the "slideshow" selects the 1st tab in the tab control and continues from there.

Some options you can use:

- When the user puts the mouse over the tab control area the "slideshow" can pause.
- By increasing the "Delay" you can make your tabs show longer before they get deselected for their next tab to become selected.
- Through the "Direction" property you can make your tabs show in "random" order or even in "reverse" order (going backwards).
- Through the "Loop" property you can make the "slideshow" start over when the final tab is reached (or vice versa if "Direction" is "Reverse"). Obviously this option has no effect if the "Direction" is set to "Random".

The "**Example 10 (play tabs)**" project uses this feature, so you can check it there:



The "Play Tabs" feature is also enabled in the "**Example 03 (image)**" and "**Example 03b (image)**" projects.

"Play/pause/navigate" through external page buttons option:

This option allows you to add some specially named images (buttons) in your web page, so that your visitors can use them to "play", "pause" or navigate the tabs at will.

View a full implementation of this feature at "**Example 03 (image)**" (Preview and Sample Code):

likno_tabs_generic_BUTTONID_play_over.png
likno_tabs_generic_BUTTONID_pause.png
likno_tabs_generic_BUTTONID_pause_over.png

- likno_tabs_generic_BUTTONID_previous.png

This image functions as the "previous" button. Note that for this function you need to put 1 more image in the same online folder, but NOT in the page:

likno_tabs_generic_BUTTONID_previous_over.png

- likno_tabs_generic_BUTTONID_next.png

This image functions as the "next" button. Note that for this function you need to put 1 more image in the same online folder, but NOT in the page:

likno_tabs_generic_BUTTONID_next_over.png

- likno_tabs_generic_BUTTONID_first.png

This image functions as the "first" button. Note that for this function you need to put 1 more image in the same online folder, but NOT in the page:

likno_tabs_generic_BUTTONID_first_over.png

- likno_tabs_generic_BUTTONID_last.png

This image functions as the "last" button. Note that for this function you need to put 1 more image in the same online folder, but NOT in the page:

likno_tabs_generic_BUTTONID_last_over.png

Some important rules:

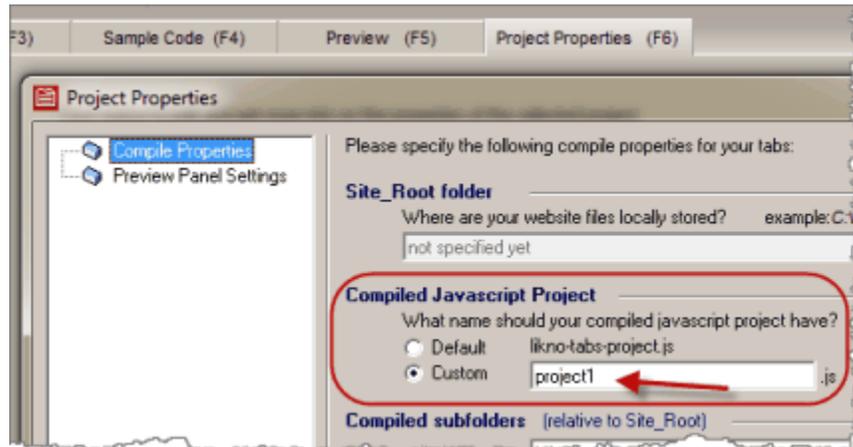
- The "_BUTTONIND_" part in the filenames can be anything you like (or nothing), the rest should remain as is, otherwise the tab control cannot map these images to the related functions ("play", "previous" etc.).

- All images above are **optional**. You can use only the ones that are related to the functions that you want. For example, you may want to use only the "play/pause" images and not the "previous", "next" ones.

- All images should be in "png", "gif" or "jpg" format.

- If you have **more than one tab controls** using this option in the same page, then you must change the "_generic_" part in the filenames to the "compiled javascript project" name of the

related tab control. For example, for this project...

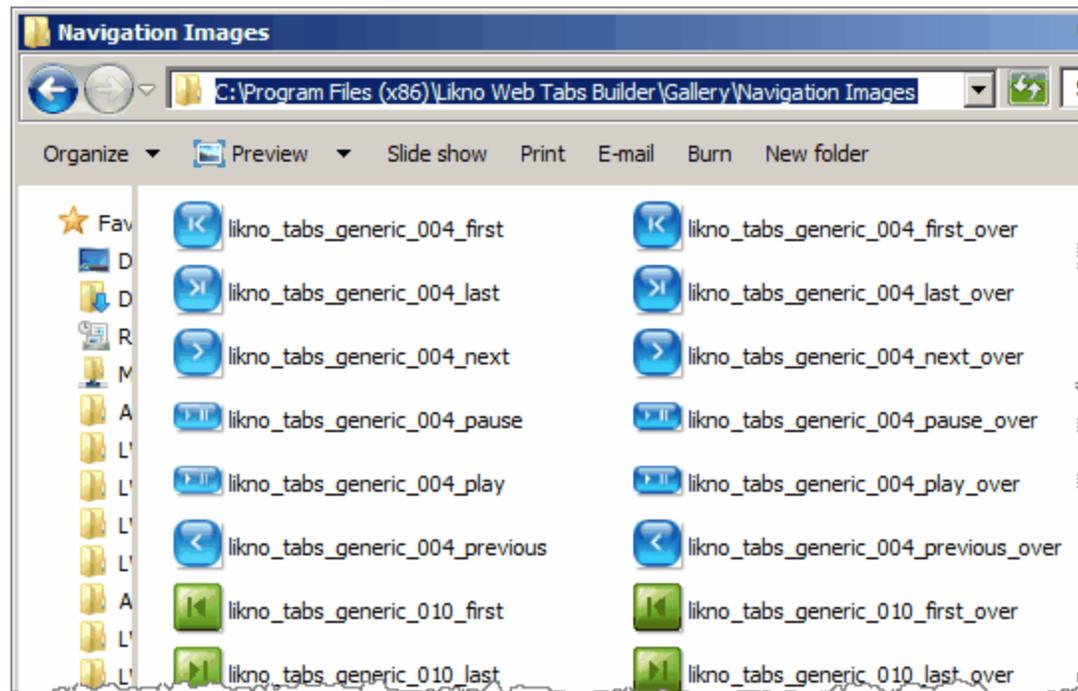


...you should use the following names:

likno_tabs_project1_BUTTONID_play.png
 likno_tabs_project1_BUTTONID_play_over.png
 likno_tabs_project1_BUTTONID_pause.png
 likno_tabs_project1_BUTTONID_pause_over.png
 ...etc.

You should do this only if you have multiple tab controls in the same page, otherwise don't bother as the "generic" name will work for any tab control no matter its name.

You will find buttons that meet all above specifications at the following installation folder:



Just grab them from there and put them online **OR you can create your own buttons according**

to the specifications explained above.

See also

[Property: Tabs Hide - How](#)
[Property: Tabs Hide - When](#)
[Property: Tabs Show - How](#)
[Property: Tabs Show - When](#)
[Property: Initially Selected Tab](#)

5.2.7 Container Width/Height

Specify the size of the container (i.e. the tab control).

If you use **"Auto" for Width**, the container will use all available width within its parent element. For example the page width, or a table cell width, or a div width, depending on who is its parent element.

If you use **"Auto" for Height**, the container will use the height of the content of its bodies (different for each tab OR the same based on the highest one).

See also

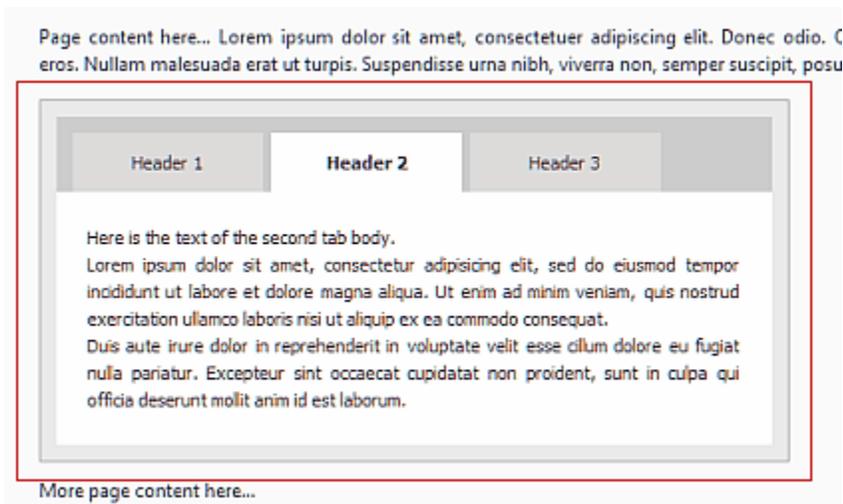
[Property: Container CSS](#)

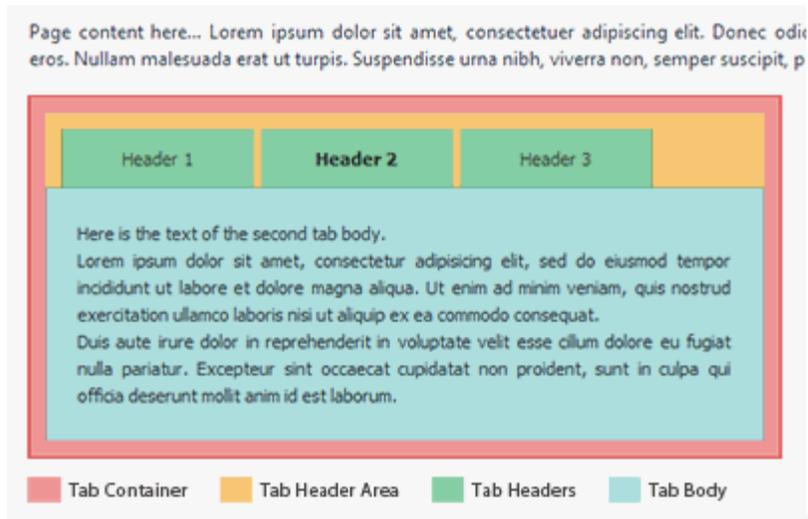
5.2.8 Container CSS

Specify the style of the container (i.e. the tab control), using pure CSS code.

For more information about CSS styling, visit <http://www.likno.com/cssref>

This property affects the appearance of the entire tab control (container), as shown below:





Note about width: Contrary to the rest of the CSS properties, do not use the 'width' parameter inside the Container CSS as there are [separate Width/Height properties](#) for this purpose.

See also

[Property: Container Width/Height](#)

[Headers CSS](#)

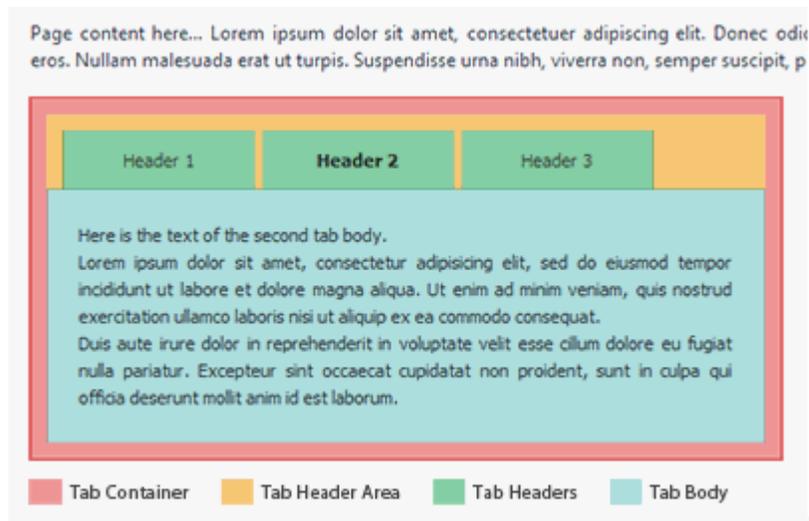
[Body CSS](#)

5.2.9 Headers CSS

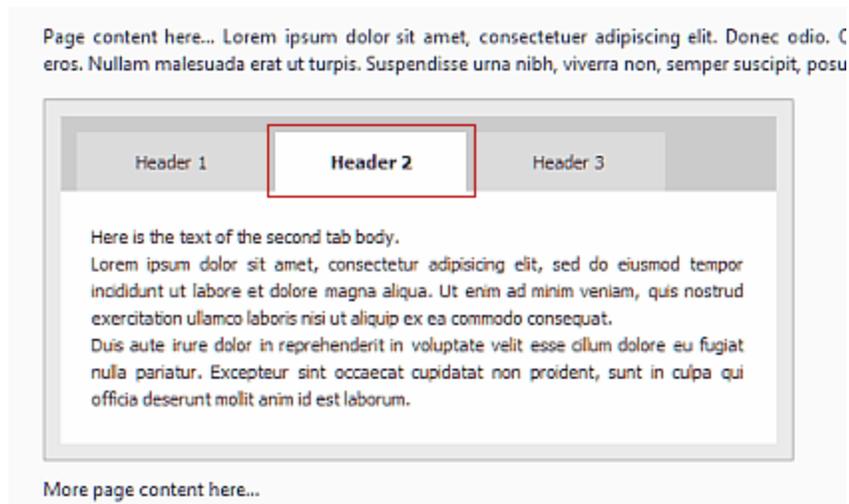
Specify the style of Headers, using pure CSS code.

For more information about CSS styling, visit <http://www.likno.com/cssref>

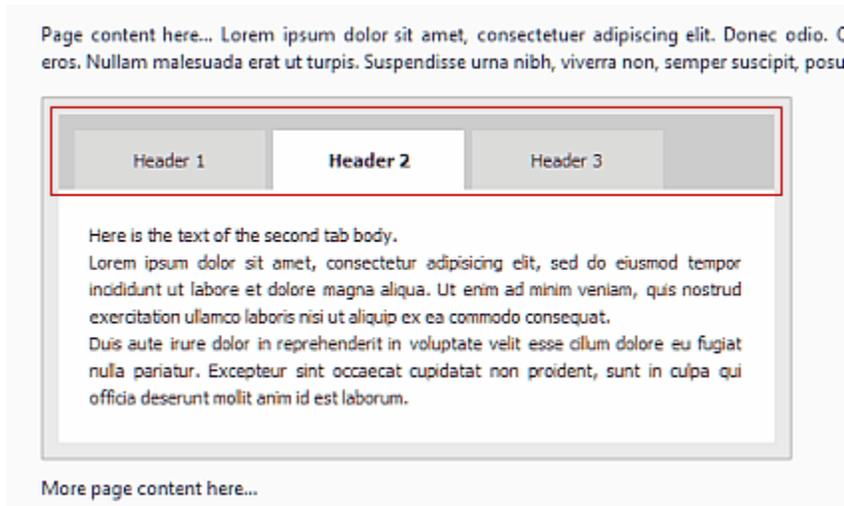
These properties affect the appearance of the "Tab Headers" and the "Tab Header Area", as shown below:



Tab Header:
(the "title" that functions as the selector of its tab)



Tab Header Area:
(the area that contains all headers of the tabs)



Tab Headers Common CSS

Through the Tab Headers Common CSS, you can specify most settings for the appearance of all your Headers.

For example, "Example 01" uses the following "Common CSS":

```
font-family: Segoe UI, Tahoma, Verdana, Arial, Helvetica, sans-serif;
font-size: 11px;
text-align: center;
padding: 10px;
cursor: pointer;
margin-right: 3px;
width: 100px;
font-weight: normal;
```

Some notes:

- Use **width: auto;** (or skip this parameter) for automatic width based on header's content.
- The same applies for height (**height: auto;** or skip this parameter).
- The **padding: 10px;** provides a "minimum" space around the header's content and its borders (shows as green below).



- The **margin-right: 3px;** provides the space between a header and its next header.



- These values are **combined** with the "Extra CSS" values for each different state (see below).

Tab Headers Extra CSS

These are **extra** CSS values that **supplement** or **override** the "Tab Headers Common CSS" values, so that headers can differentiate their appearance on their 4 different states:

Tab Deselected state:



Tab Deselected - Mouse Over state:



Tab Selected state:



Tab Selected - Mouse Over state:



For example, "Example 01" uses the following "Extra CSS" values:

Tab Deselected state:

```
color: #3f3f3f;
background: #efefef;
border-top: 1px solid #b3b3b3;
border-left: 1px solid #b3b3b3;
border-right: 1px solid #b3b3b3;
border-bottom: 1px solid #909090;
```

Tab Deselected - Mouse Over state:

```
color: #000000;
background: #f7f7f7; (differs from "Deselected")
font-weight: bold; (differs from "Deselected")
border-top: 1px solid #b3b3b3;
border-left: 1px solid #b3b3b3;
border-right: 1px solid #b3b3b3;
border-bottom: 1px solid #909090;
```

Tab Selected state:*(most values differ from "Deselected")*

```

color: #3f3f3f;
background: #FFFFFF;
font-weight: bold;
border-top: 1px solid #909090;
border-left: 1px solid #909090;
border-right: 1px solid #909090;
border-bottom: 1px solid #FFFFFF;

```

Tab Selected - Mouse Over state:

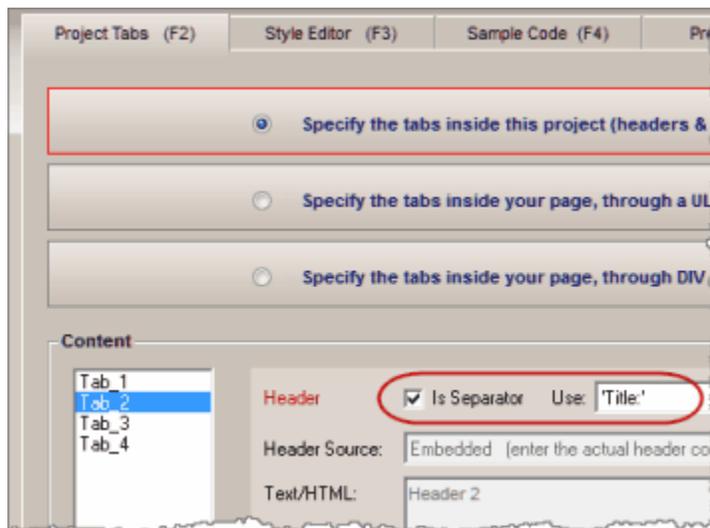
```

color: #000000; (differs from "Selected")
background: #FFFFFF;
font-weight: bold;
border-top: 1px solid #909090;
border-left: 1px solid #909090;
border-right: 1px solid #909090;
border-bottom: 1px solid #FFFFFF;

```

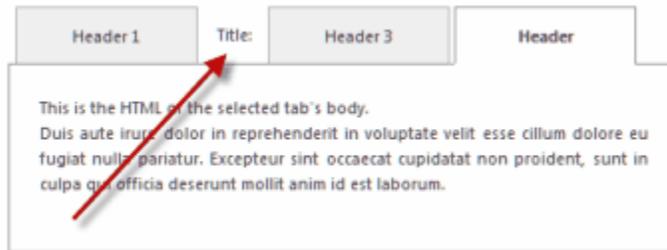
Tab Headers Separator CSS

A header can act as a separator between 2 other headers or groups of headers, when the following option is used in "Project Tabs" for its tab:



A separator does not have any "mouse over/click" effects, neither "body content", as it is there to provide separation or "title" support.

For example in "Example 01", if you make "Tab_2" a separator you will get something like this:

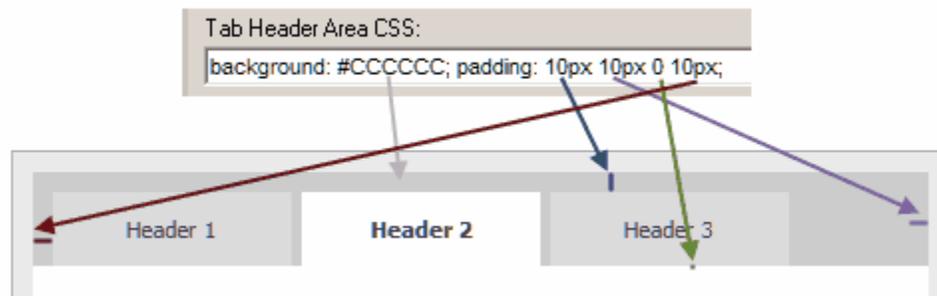


Through the Tab Headers Separator CSS, your "separator" headers can **differentiate** their appearance from "normal" headers.

Tab Header Area CSS:

Through the Tab Header Area CSS, you can adjust the space around the entire "area" of headers ("padding"), the area background color ("background") or background image ("background-image"), the area border ("border"), etc.

For example, "Example 02" uses the following values:



Note that space between headers is adjusted through the **Tab Headers Common CSS** (see above).

About the 'width' parameter:

Contrary to the Container CSS property, you need to use the 'width' parameter inside the Header CSS properties as there are no [separate Width/Height properties](#) for this purpose. If you want the width to be determined automatically and match the Header's content width, just use **width:auto**; or do not use the 'width' parameter at all.

About the "Ignore Headers CSS" option:

This option appears only if the [Project Type](#) is "Headers Outside Container".

In that case, if you want your headers (which are page DIVs) to use your website CSS instead of

the CSS values specified within the project, you can check the **"Ignore header-related CSS"** option in Style Editor:



Read more on the [Ignore header-related CSS](#) topic.

See also

[Container CSS](#)

[Body CSS](#)

[Ignore header-related CSS](#)

5.2.10 Body CSS

Specify the style of the bodies, using pure CSS code.

For more information about CSS styling, visit <http://www.likno.com/cssref>

Note about width: Contrary to the Container CSS property, you need to use the 'width' parameter inside the Body CSS properties as there are no [separate Width/Height properties](#) for this purpose. If you want the width to be determined automatically and match the container's width, just use **width:auto**; or do not use the 'width' parameter at all.

See also

[Container CSS](#)

[Headers CSS](#)

5.2.11 Ignore header-related CSS

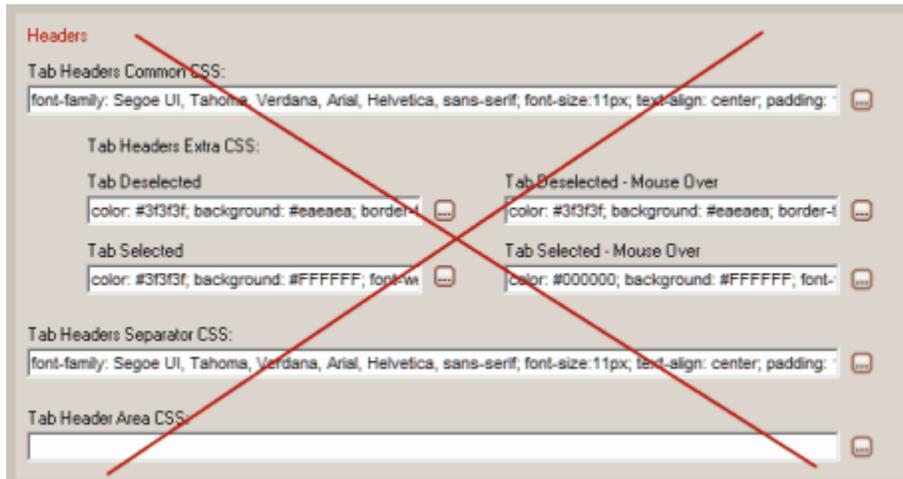
This option appears only if the [Project Type](#) is "Headers Outside Container".

If you want your headers (which are page DIVs) to use your website CSS instead of the CSS values specified within the project, you can check the **"Ignore header-related CSS"** option:



If you do so, the page DIVs that play the role of "Headers" (those using `class="liknotabs_header"`)

ignore the project's header-related CSS properties in Style Editor:



Instead, they use the **CSS of your page** for their appearance.

Example 15 demonstrates this feature: changes on the project's Headers CSS are ignored, as the headers styling is controlled by the website CSS instead.

5.3 "Project Tabs" Properties

5.3.1 Tab Control Populated By

This is where these options are located:



1. Tab Control is populated by the **project itself**.

This means that you add/edit all tabs, along with their header content and body content inside this application. The compiled project contains all information: the number of tabs and their content, as well as the style and positioning preferences.

2. Tab Control is populated by a **UL/LI structure**, located inside your page.

This means that you add/edit all tabs, along with their header content and body content NOT inside this application but inside your online page where the tab control is to be located. The compiled project contains only the style and positioning preferences but the actual data on the number of tabs and their content is retrieved by a UL/LI structure that you add and edit inside your page.

For example:

```
<ul style="display: none;">
  <li>
    Header 1
  </li>
  <li>
    <p>Here is the text of the 1st tab body.</p>
  </li>
  <li>
    Header 2
  </li>
  <li>
    <p>Here is the text of the 2nd tab body.</p>
  </li>
  <li>
    Header 3
  </li>
  <li>
    <p>Here is the text of the 3rd tab body.</p>
  </li>
</ul>
```

Note that you may also add a separator between two tabs, through an additional tag that uses special syntax:

```
<ul style="display: none;">
  <li>
    Header 1
  </li>
  <li>
    <p>Here is the text of the 1st tab body.</p>
  </li>
  <li> [SEPARATOR]
  </li>
  <li>
    Header 2
  </li>
```

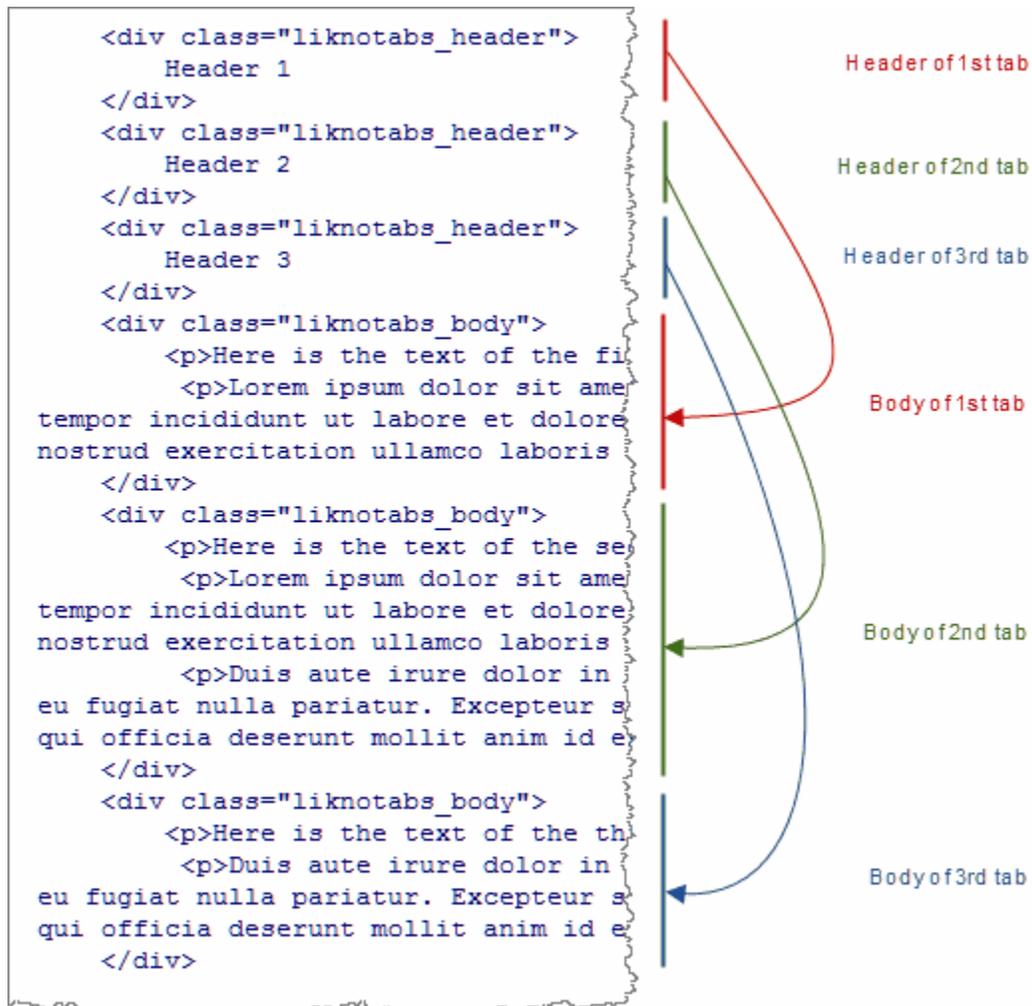
```
<li>  
  <p>Here is the text of the 2nd tab body.</p>  
</li>  
  
<li>  
  Header 3  
</li>  
  
<li>  
  <p>Here is the text of the 3rd tab body.</p>  
</li>  
  
</ul>
```

In the above example, the separator is specified by the [SEPARATOR] string which is followed by the actual separator characters (any characters you like):



3. Tab Control is populated by **DIV elements**, located inside your page.

This works exactly like the UL/LI option above, with the difference that you use DIV elements to specify the tabs and their content on your page, instead of the UL/LI structure.



Note about "image-based headers" when using the "UL/LI structure" or "DIV elements" population method:

You can also use 4 images for the 4 different states of a single image header, if you just put their tags at the beginning of the element and next to each other:

```

<li>
  
  
  
  
</li>

```

OR

```

<div class="liknotabs_header">
  
  
  

```

```

</div>
```

OR (both image and text)

```
<div class="liknotabs_header">
  
  
  
  
  Header Text here...
</div>
```

5.3.2 Project Type: Headers Within/Outside Container

This specifies where the tab headers are positioned relative to the container (and their bodies).

There are 2 types of projects (as shown in the examples):

- Projects that use the ["Headers Within Container" Project Type](#).

This is the typical and easiest type, where all headers are "glued" to their bodies, inside the tab control container.

- Projects that use the ["Headers Outside Container" Project Type](#) (advanced).

This is an advanced type, where headers are **not "glued"** to their bodies but are rather "free" elements (DIVs) of the page, **outside** the tab control container.

This allows for advanced tab controls as it provides non-traditional tab designs for your pages. It is for **advanced users** only, that understand HTML and CSS well.

In this type, your project can use the **"DIV elements"** [population option](#) only.

Note: All Examples (and your projects that originate from them) are usually fitted for a **single Project Type**. You can still switch to the other type however, by creating a "clone" of this project with its Project Type changed. Just note that this change usually "breaks" the styling of the project notably, requiring you to also adjust its CSS properties accordingly.

See also

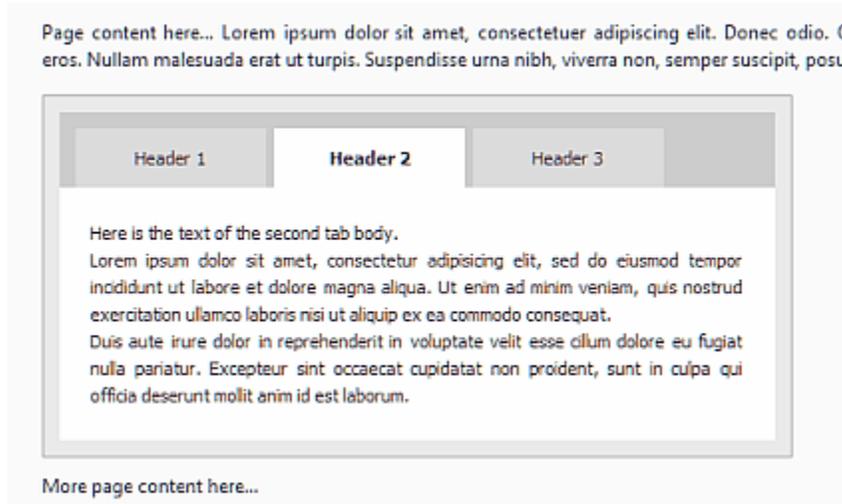
[Project Type 1: Headers Within Container](#)

[Project Type 2: Headers Outside Container](#)

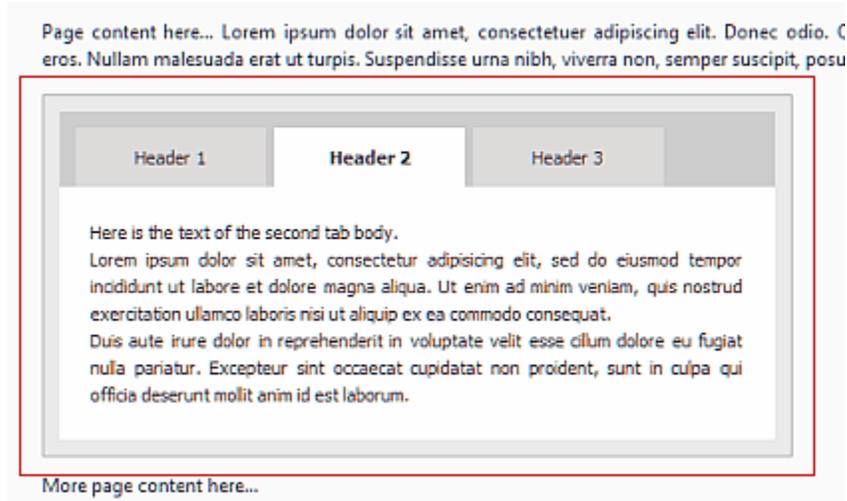
5.3.2.1 Headers Within Container (Project Type 1 of 2)

This is the typical project type, where all headers are "glued" to their bodies, inside the "Container" area of the tab control.

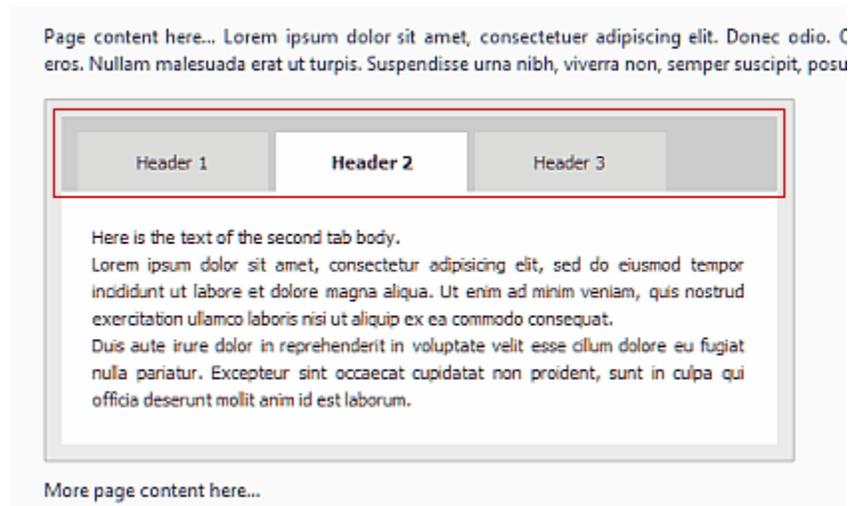
For example, for this tab control...



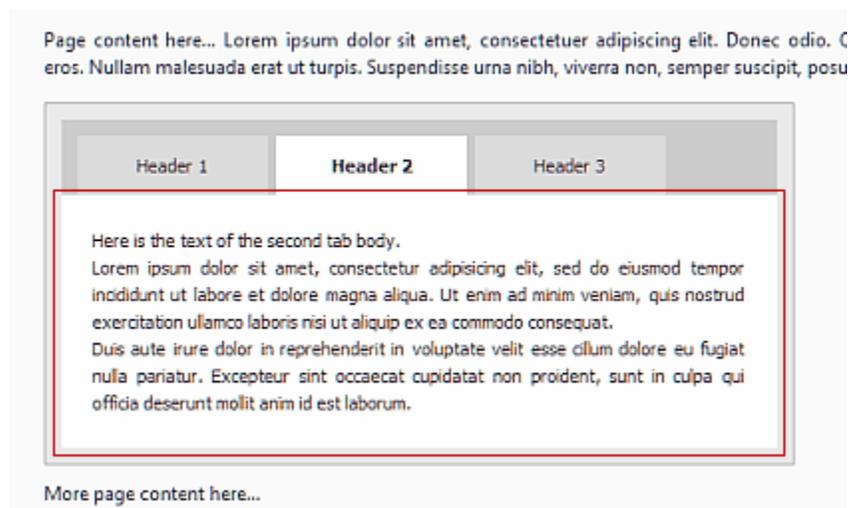
...this is the **"Container"**...



...which contains the **"Headers"**...



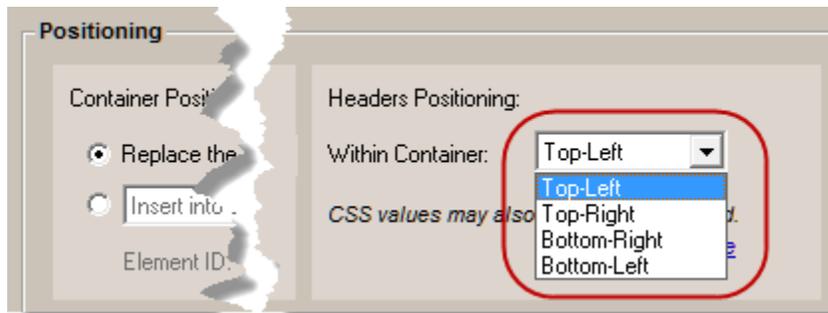
... which are glued to their **"Bodies"** below them:



(unlike "Headers", "Bodies" are always within the "Container")

Changing the headers alignment and CSS considerations:

You can also change the **alignment** of your tab headers here:



In most cases, changing the original positioning value will make your tab headers show "broken" in design. This is because you need to **make some adjustments to the CSS values** in the Style Editor.

For example, in "**Example 04**" (which is a "**Top**" headers positioning example), the "Tab Headers Deselected CSS" is as follows:

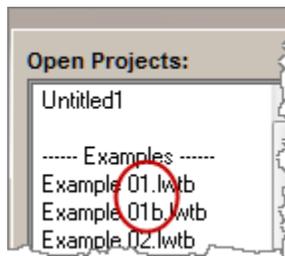


```
color: #3f3f3f;
background: #CCCCCC;
border-top: 1px solid #b3b3b3;
border-left: 1px solid #b3b3b3;
border-right: 1px solid #b3b3b3;
border-bottom: 1px solid #909090;
background-image: url(example04headerbg.jpg);
background-repeat: repeat-x;
background-position: top;
```

In the similar "**Example 04b**" (which is its "**Bottom**" respective example), the "Tab Headers Deselected CSS" is:

```
color: #3f3f3f;
background: #CCCCCC;
border-top: 1px solid #909090; (differs from "Top")
border-left: 1px solid #b3b3b3;
border-right: 1px solid #b3b3b3;
border-bottom: 1px solid #b3b3b3; (differs from "Top")
background-image: url(example04bheaderbg.jpg);
background-repeat: repeat-x;
background-position: top;
```

Concerning the "Top/Bottom" differentiation, most "Top" examples already come with their respective "Bottom" project, as shown below:



So it is strongly recommended that you start your project from an example that already matches your header positioning needs, at least in terms of "Top" or "Bottom" positioning.

When changing the alignment from "Left" to "Right", in most cases you will have to adjust properties such as:

- padding-left,
- padding-right,
- margin-left,
- margin-right,
- border-left,
- border-right,
- etc.

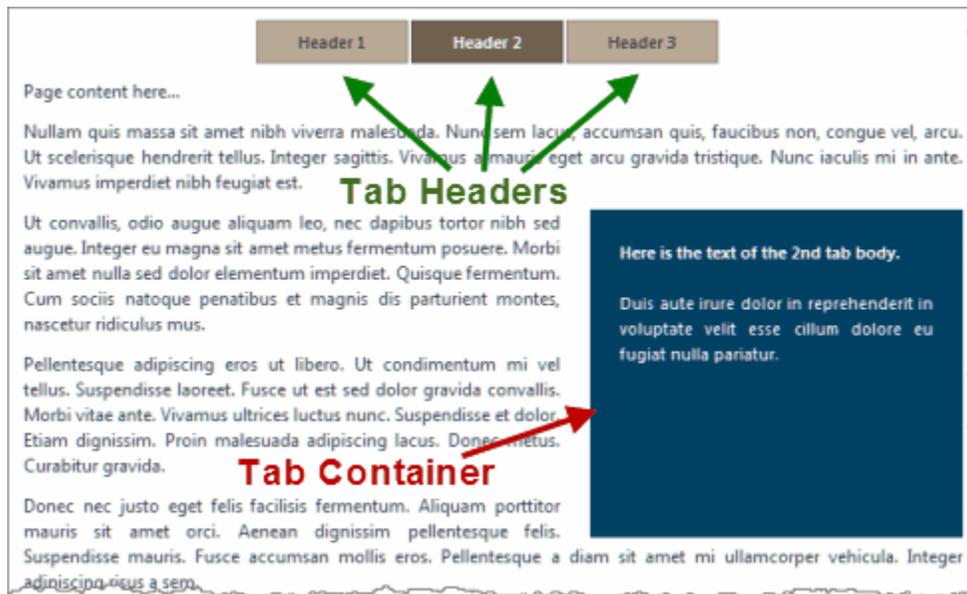
See also

[The 2 Project Types for "Headers Positioning"](#)
[Project Type 2 of 2: Headers Outside Container](#)

5.3.2.2 Headers Outside Container (Project Type 2 of 2)

This is an **advanced** project type, where headers are **not "glued"** to their bodies but are rather "free" elements (DIVs) of the page, **outside** the tab control container.

Example:

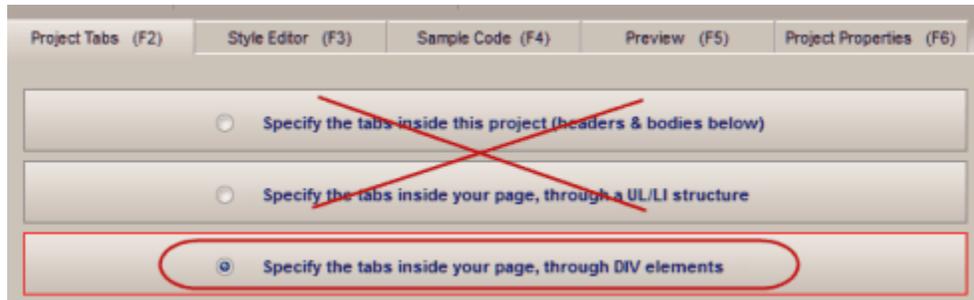


OR:



This allows for advanced tab controls as it provides non-traditional tab designs for your pages. It is for **advanced users** only, that understand HTML and CSS well.

If you start your project from an example that uses this type, your project will use the **"DIV elements" population option** only.



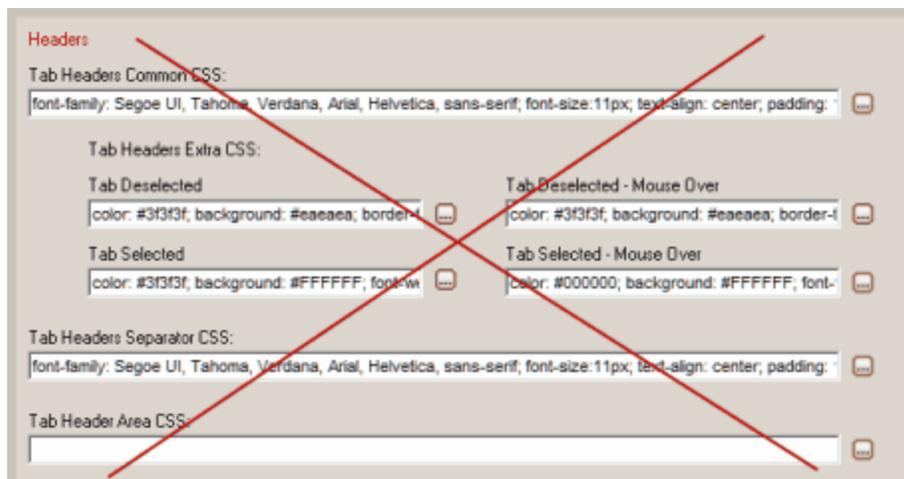
Some of the Examples that use this method are the following: **Example 12, 13, 14, 15, 17, 19, 20, 21**

About the "Ignore Headers CSS" option:

If you want your headers (which are page DIVs) to use your website CSS instead of the CSS values specified within the project, you can check the **"Ignore header-related CSS"** option in Style Editor:



If you do so, the page DIVs that play the role of "Headers" (those using `class="liknotabs_header"`) **ignore the project's header-related CSS properties** in Style Editor:



Instead, they use the **CSS of your page** for their appearance.

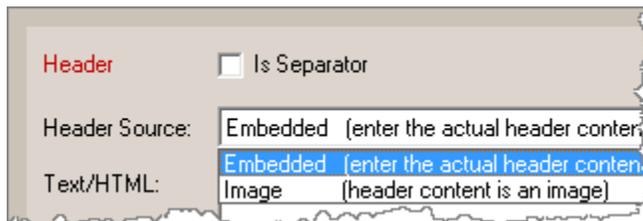
Example 15 demonstrates this feature: changes on the project's Headers CSS are ignored, as the headers styling is controlled by the website CSS instead.

See also

[The 2 Project Types for "Headers Positioning"](#)
[Project Type 1 of 2: Headers Within Container](#)

5.3.3 Header Source options

This is where these options are located:

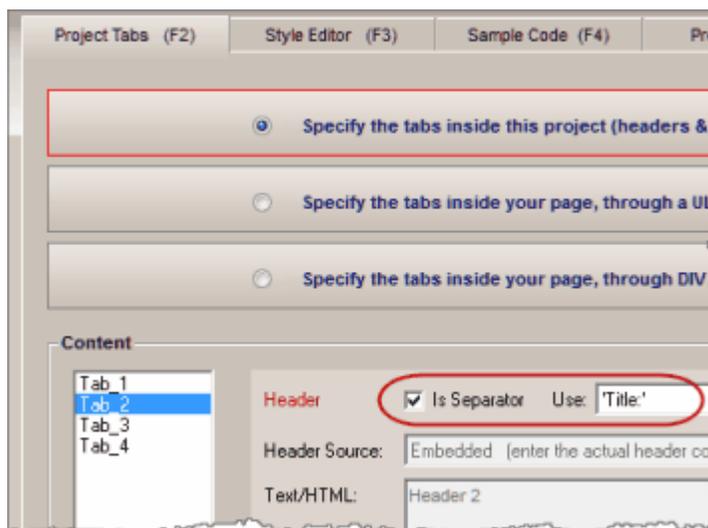


Read more for these options on how to specify the content of a tab header:

[Is Separator](#)
[Embedded](#)
[Image](#)

5.3.3.1 Is Separator

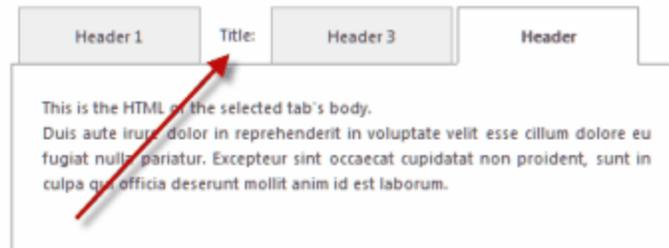
A header can act as a separator between 2 other headers or groups of headers, when the following option is used in "Project Tabs" for its tab:



A separator does not have any "mouse over/click" effects, neither "body content", as it is there to

provide separation or "title" support.

For example in "Example 01", if you make "Tab_2" a separator you will get something like this:



The combo box next to the "Is Separator" option provides some standard content to show.

The most important option is the last one: <Custom: Use Header values below>

This means that you can show anything you like by specifying it in the Header [Embedded](#) or [Image](#) properties.

Note: Through the [Tab Headers Separator CSS](#), your "separator" headers can differentiate their appearance from "normal" headers.

See also

[Header Source options](#)

[Tab Headers Separator CSS](#)

5.3.3.2 Embedded (for header)

Add the content of the tab header directly into the "Text/HTML" property.

You can either add simple text to be displayed, or more complex HTML code.

Notes:

- The paths of any files (images, etc.) used inside the HTML should be relative to the location of the web page that shows the tab control. [read more](#)

- For the Preview Panel to be able to show images using Site_Root-relative paths (example: `"/image1.gif"`) or Page-relative paths (example: `"image1.gif"` or `"pics/image1.gif"`), you need to add them to the "Project Assets" folder or to any other local folder that you specify through the **Image Folders for Preview Panel** list in [Preview Panel Settings](#).

See also

[Header Source options](#)

5.3.3.3 Image (for header)

The content of the tab header is an image.

You can also specify a different image to show on the 4 different states:

Tab Deselected state

Tab Deselected - Mouse Over state

Tab Selected state

Tab Selected - Mouse Over state

The image can be sourced either from the project's "Assets" folder, the application "Gallery" folder or a web location (relative or absolute to the loaded page).

Note when using the "UL/LI structure" or "DIV elements" population method:

You can also use 4 images for the 4 different states of a single image header, if you just put their tags at the beginning of the element and next to each other:

```
<li>
  
  
  
  
</li>
```

OR

```
<div class="liknotabs_header">
  
  
  
  
</div>
```

OR (both image and text)

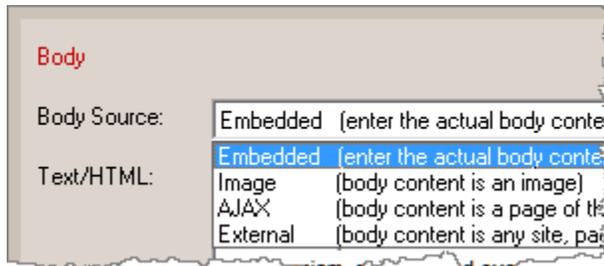
```
<div class="liknotabs_header">
  
  
  
  
  Header Text here...
</div>
```

See also

[Header Source options](#)

5.3.4 Body Source options

This is where these options are located:



Read more for these options on how to specify the content of a tab body:

[Embedded](#)
[Image](#)
[AJAX](#)
[External Site](#)

5.3.4.1 Embedded

Add the content of the tab body directly into the "Text/HTML" property.

You can either add simple text to be displayed, or more complex HTML code.

Notes:

- The paths of any files (images, etc.) used inside the HTML should be relative to the location of the web page that shows the tab control. [read more](#)

- For the Preview Panel to be able to show images using Site_Root-relative paths (example: `"/image1.gif"`) or Page-relative paths (example: `"image1.gif"` or `"pics/image1.gif"`), you need to add them to the "Project Assets" folder or to any other local folder that you specify through the **Image Folders for Preview Panel** list in [Preview Panel Settings](#).

See also

[How do I display a YouTube video in a Tab?](#)
[Body Source options](#)

5.3.4.2 Image

The content of the tab body is an image.

You can also specify a different image to show on the 4 different states:

Tab Deselected state

Tab Deselected - Mouse Over state

Tab Selected state

Tab Selected - Mouse Over state

The image can be sourced either from the project's "Assets" folder, the application "Gallery" folder or a web location (relative or absolute to the loaded page).

See also

[Body Source options](#)

5.3.4.3 AJAX

The content of the tab body is contained in a page that is located in your domain and does NOT have JavaScript code. AJAX does not allow calling content (HTML) from a page outside your domain. Add the path/name of the file in the "Page Link" property.

Examples: ajax1.php
 ajax2.htm
 /ajaxpages/ajax1.php

Notes:

- The path of the page should be **relative** to the location of the web page that shows the tab control.
- The entered page should be present on your website, the **Preview cannot show it!**
- You **cannot use absolute URLs**, such as http:// or https://, choose the "[External](#)" option for this.

See also

[Body Source options](#)

5.3.4.4 External

The content of the tab body is an external site (e.g. Google) or page.

Add the URL of the site you want to be displayed at the "Page Link" property.

Example: http://www.google.com

See also

[Body Source options](#)

5.3.5 Container Positioning

Specify where the tab control (i.e. its container) will be placed inside your page.

You can either insert it **inside a page element** (usually a DIV) or have the tab control **replace this page element** (this means that you only use this element as a placeholder).

You can also use an **image** as a placeholder (sometimes this is easier, like in blogs or CMS platforms where you may not be able to add DIVs in your page HTML).

How Do I...

Part



6 How Do I...

6.1 Find my INSTALLATION ID

In order to activate your license you need to login to your Likno User Area and provide us with the unique **Installation ID** of your current installation.

Your Installation ID can be found at the "**Help -> Enter Activation Code**" option of the application menu.

6.2 Purchase/Activate

Step 1. Purchase

- Access the online registration page located at <http://www.likno.com/jquery-tabs/purchase.php>
 - Click on your preferred method of payment and enter the necessary info in the order form
- You will soon receive an e-mail containing instructions on how to get the Activation Code to use it in your installation.

Step 2. Activate your Installation

- Choose the "**Help -> Enter Activation Code**" menu option in the application
- Enter your Activation Code

You will need to have "Full Admin rights", [read how here](#).

Following these simple steps you will be a **registered user** and you will be able to take **full advantage** of the Likno Web Tabs Builder features.

Methods of payment

There are many flexible payment options:

- Credit Card Orders (all major credit cards accepted)
- Phone Orders (toll-free)
- Mail Orders (pay by check)
- Fax Orders
- Purchase Orders

All on-line transactions are protected by an extremely high level of encryption, the highest allowed by law.

6.3 Add effects when a tab body shows or hides

Change the [Show - How](#) and the [Hide - How](#) properties under the "**Style Editor**" tab.

See also

[Property: Show - How](#)

[Property: Hide - How](#)

6.4 Change the position where the tab control is located on page

Change the [Container Positioning](#) property under the "Style Editor" tab.

See also

[Container Positioning](#)

6.5 Load with AJAX the content of a tab body

You might need to load the content of one of your tab bodies from another file in your server at the time you display that tab. In that case you need to use the **AJAX** technique in order to load a remote HTML file and inject its contents into the DOM.

This technique is included into the application, and the only thing you need to do is to select the "AJAX" value in the [<Body Source>](#) property under the "Project Tabs (F2)" panel, and then define the path and name of the file that you want to display, through the [<Page Link>](#) property below.

Note! Due to security issues you are not allowed to enter an *absolute path* as the value of the call property. You can only call a file that resides within your site by using a *relative path*.

See also

[Body Source options](#)

[Body Source: AJAX](#)

6.6 Load an external site as the content of a tab body

You might need to load a full site as the content of one of your tab bodies.

In that case you have to select the "External Site" value in the [<Body Source>](#) property under the "Project Tabs (F2)" tab, and then define the URL of the site that you want to display in the [<Page Link>](#) property below.

See also

[Body Source options](#)

[Body Source: External](#)

6.7 Use multiple Tab projects in the same page

There is no restriction in the number of Tab projects you can use in the same page.

You only need to specify a different name for the "Compiled JavaScript Project" (default: *likno-tabs-project.js*) inside the [Project Properties](#) pane and make sure that you [link](#) all different projects to your page.

Note! It is very important that different tab projects are using different page elements for their positioning. You cannot associate the same page element with two different tab projects!

Also read *Troubleshooting on common errors* about multiple projects.

6.8 Properly use paths in the "Website Folder" option

Contrary to the "Project Assets" option, this property is **free text** as you may need to enter both the **path** and the **filename** of the website image.

You can use **paths** to the image file in 3 different ways:

"Absolute" paths:

- <http://www.likno.com/jquery-tabs/assets-app/close.gif>
- <http://www.mysite.com/pics/image1.gif>
- <https://www.google.com/image2.gif>

"Document-relative" paths:

*(image folder is relative to the folder of the **loaded page** that shows the tab control)*

- close.gif
- pics/close.gif
- ../close2.gif

"Site_Root-relative" paths:

(image folder is relative to the Site Root folder)

- /close.gif
- /pics/close.gif
- /close2.gif

Note: For the Preview Panel to be able to show images using Site_Root-relative paths (example: *"/image1.gif"*) or Page-relative paths (example: *"image1.gif"* or *"pics/image1.gif"*), you need to add them to the "Project Assets" folder or to any other local folder that you specify through the **Image Folders for Preview Panel** list in [Preview Panel Settings](#).

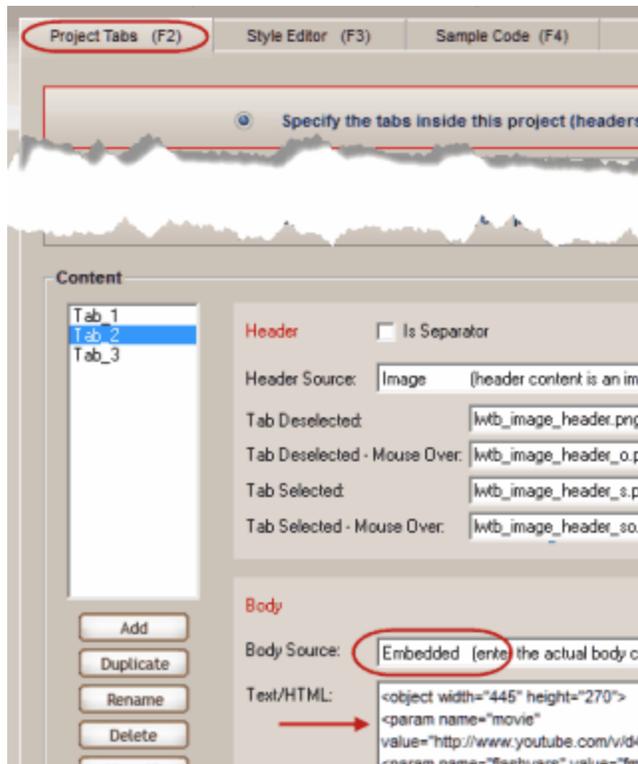
6.9 Display a video/movie in a Tab?

You can load and display a video/movie inside a tab body.

Open "Example 08" inside the application to see this functionality.

How is this setup?

Copy the video's "embed code" from YouTube (for example) and paste it into the "Text/HTML" property of the Body [Embedded](#) option, inside the "Project Tabs" tab.



You can also change the [Container Width/Height](#) properties in "Style Editor" tab, to fit the width and height of the video and avoid scrollbars (or just use "auto" there).

To show your own video, just replace the "<object>" section with its respective embed code:

```
<object width="560" height="340">
<param name="movie"
value="http://www.youtube.com/v/d41Y9MqXrgE&hl=en&fs=1"></param>
<param name="flashvars"
value="fmt_url_map=34%7Chttp%3A%2F%2Fv12.lscache5.c.youtube.com%2Fvideo
playback"></param>
<param name="allowFullScreen" value="true"></param>
<param name="allowscriptaccess" value="always"></param>
<embed src="http://www.youtube.com/v/d41Y9MqXrgE&hl=en&fs=1"
type="application/x-shockwave-flash" allowscriptaccess="always"
allowfullscreen="true" width="560" height="340"></embed>
</object>
```

Exactly the same would apply for other movie types, such as Flash movies.

Note: If you want to support IE6 (which is dropped by YouTube), you need to add the following parameter inside the embed code that you take from YouTube:

```
<param name="flashvars"
value="fmt_url_map=34%7Chttp%3A%2F%2Fv12.lscache5.c.youtube.com%2Fvideo
playback"></param>
```

6.10 Use the "background-image" value in CSS properties

The following is an example of CSS that applies a background image to the tab through the **background-image** value:

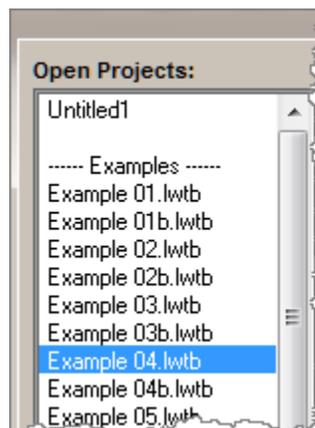
```
font-family: Segoe UI, Tahoma, Verdana, Arial, Helvetica, sans-serif; font-size: 11px; text-align:
justify; padding: 20px; color: #3f3f3f; border: 1px solid #909090; background-color: #eaeaea;
background-image: url(example04bodybg.jpg); background-repeat: repeat-x; background-position:
top;
```

1. Location of the image

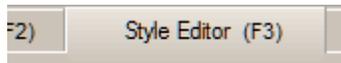
The project needs to know **where to find this image**, otherwise it will not be displayed in the Preview Panel and will not be compiled either.

You should put this image inside the project's "Assets" folder, which is a subfolder at the folder where the project is located.

To demonstrate this, just open the "Example 04.lwtb"...



... go to the "Style Editor tab..."



...and edit the Body CSS value:



There you will find the CSS mentioned in the beginning, that uses the `background-image: url(example04bodybg.jpg);` value.

Now, open the "assets" folder of this example, which should be at:

C:\Program Files\Likno Web Tabs Builder\Examples\Example 04-assets
(the installation path may be different in your computer)

There you will find the image file **example04bodybg.jpg**.

If that image wasn't there, the background could not appear.

So, if you ever use a background image in your CSS, make sure that you copy this image at the project asset's folder first, otherwise the image cannot be used by the compiled project.

2. Adjusting the width/height of the Container and "background-image" considerations

When a tab control uses a "background-image" on its Container CSS (or other CSS properties), this image can either **"repeat"** itself (**tiles** within the background) or **"no-repeat"** (shows as is, once).

This is determined by the **"background-repeat"** CSS value:

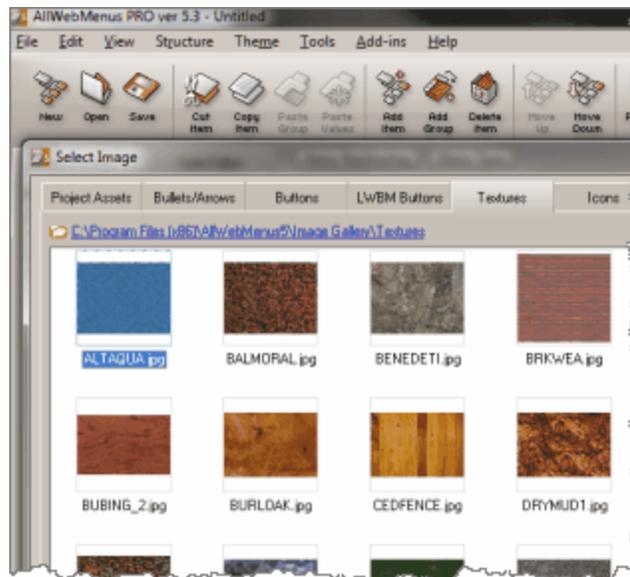
```
background-repeat: no-repeat;  
OR  
background-repeat: repeat;  
OR  
no existence of this property, which defaults to "background-repeat: repeat;"
```

Why is this important?

1. When the **"no-repeat"** value is used, the control may not show properly when you change the size ("Width" or "Height") of the Container. To show the control properly when making the Container bigger or smaller, you need to **also resize the background image with an image editor**. See "Example 07" or "Example 10", which use a "no-repeat" background image on Container.
2. This is not the case when the **"repeat"** value is used (or no value is used at all). Since the image tiles and acts as a "texture", **no image resizing is required** when changing the Container size.

Tip on "texture" images:

[AllWebMenus](#) has plenty of textures you can copy and use as background images:



What is...

Part



7 What is...

7.1 The HTML Linking Code

This is the code added in the HTML of the page during the linking process in order to link the necessary files to your page.

This section looks like the following one (yours may be slightly different):

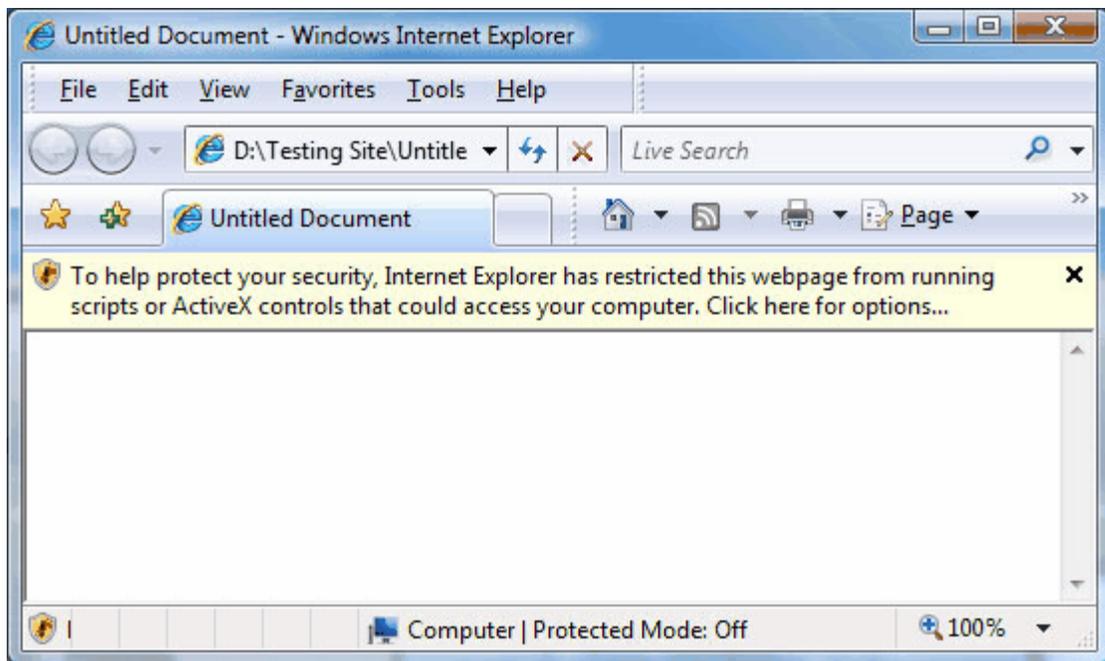
```
<!-- ***** BEGIN LIKNO WEB TABS CODE FOR likno-tabs-project ***** -->
<script type="text/javascript">var lwtbLinkedBy="LiknoWebTabs
[1]",lwtbName="likno-tabs-project",lwtbBN="112";</script><script charset="UTF-8" src="
likno-scripts/likno-tabs-project.js" type="text/javascript"></script>
<!-- ***** END LIKNO WEB TABS CODE FOR likno-tabs-project ***** -->
```

Note: You may need to adjust the "src" path (highlighted), so that your page links to the "js" file correctly.

7.2 The Internet Explorer Information Bar Warning

When you preview a **local** web page in Internet Explorer (IE6 XP SP2 +, IE7) that contains **javascript** (like our scripts do), you will get the following message:

[To help protect your security, Internet Explorer has restricted this webpage from running scripts or ActiveX controls that could access your computer. Click here for options...]



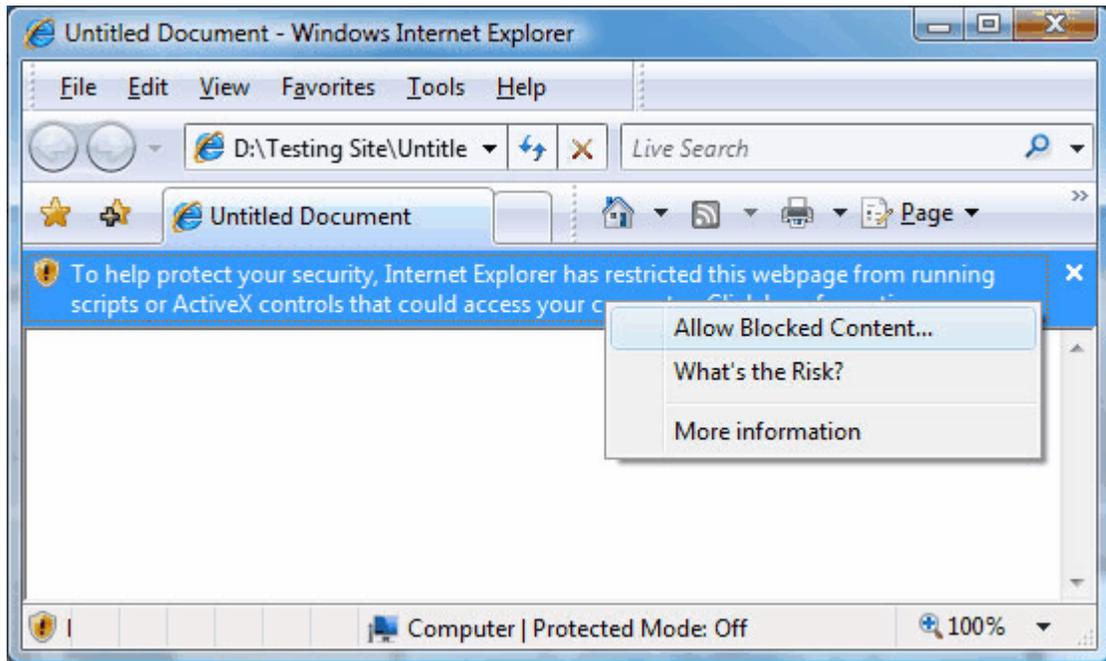
This is **not a problem**, as the "security" warning **never appears online on your actual pages**. It is not only related to Likno Software products but to all javascripts running on your pages **locally**.

This default behavior of Internet Explorer only happens when viewing pages saved to the **local**

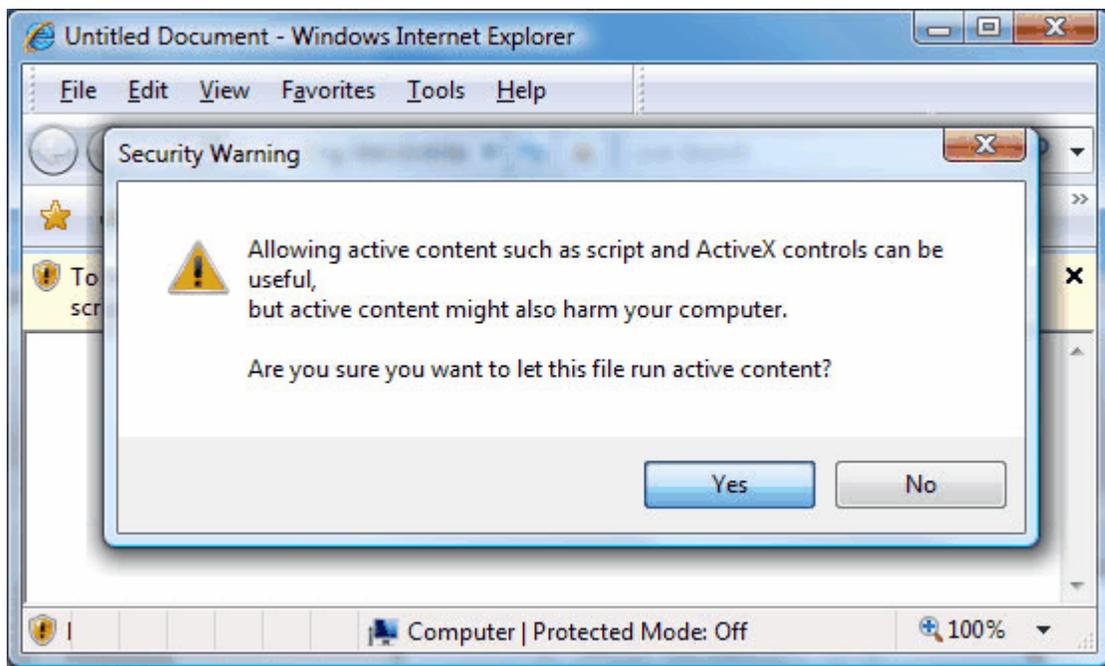
computer and only in Internet Explorer.

This does not happen when pages have been uploaded to a web server and are then viewed online.

To overpass this Internet Explorer warning, you can click on the message bar and select "**Allow Blocked Content**":



Then, a warning dialog will appear and you can just press "**Yes**" for the web page to properly load all the javascript related files:



(do not worry about the warning, our scripts are 100% safe)

You can read more about the IE Information bar at:

<http://support.microsoft.com/kb/843017> (Windows XP)

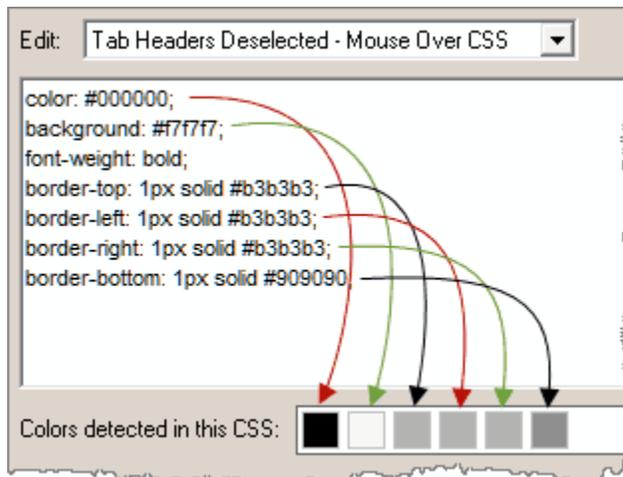
<http://windowshelp.microsoft.com/Windows/en-US/Help/91b1038d-3f53-43a3-8bd8-9551e7c0b7c61033.mspx> (Windows Vista)

7.3 The "Colors detected in this CSS" feature

This feature allows you to change a color in a CSS property, by using an advanced Color Picker control instead of typing its HEX code.

How does this work?

When you edit any CSS property, the application tries to detect the colors that the CSS text is using, by detecting their HEX code:



Note that only the 3-digit or 6-digit HEX syntax (starting with a #) can be detected by the Color Picker control, such as:

color: #0000FF;

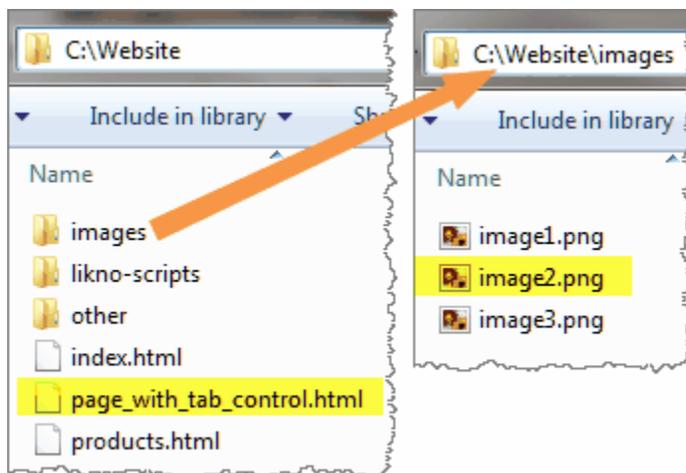
The following (valid) color syntaxes are not detected for easy editing (but are still functional as CSS values):

color: blue;
 color: rgb(0,0,255);
 color: rgb(0%,0%,100%);

7.4 The "Relative paths detected in this HTML" feature

This feature detects all "relative" (not absolute) paths used by resources (images, files, etc.) inside the "Embedded" Header or Body HTML, so that you check if these point to the correct location relative to the online page that contains the tab control.

For example, let's assume that you have the following website structure...



...and that the "page_with_tab_control.html" is the page where you have your tab control added.

If you want to show "image2.png" in a tab, this is how its "Embedded HTML" should look like:

```
<div style='padding: 21px 0 20px 17px; line-height: 14px;'>  
  
  
  
Text here, next to the image...  
  
</div>
```

OR `src="/images/image2.png"` but NOT `src="image2.png"`.

Why?

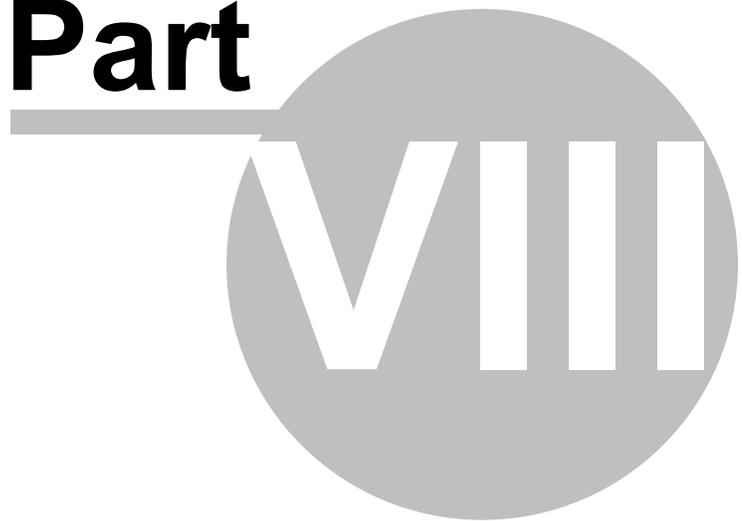
Because, the compiled tab control seeks "relative" resources (images, files, etc. that use "relative" paths in the "Embedded HTML") based on the page that shows it. And in this example the image is located in a subfolder of where the page is located.

As a "worry-free" alternative you can use "absolute" paths, like `http://www.likno.com/images/image2.png`.

Note for the Preview Panel only: For the Preview Panel to be able to show images using Site_Root-relative paths (example: `"/image1.gif"`) or Page-relative paths (example: `"image1.gif"` or `"pics/image1.gif"`), you need to add them to the "Project Assets" folder or to any other local folder that you specify through the **Image Folders for Preview Panel** list in [Preview Panel Settings](#).

FAQ and Tips

Part



8 FAQ and Tips

8.1 Troubleshooting

This is a collection of quick tips to avoid trouble.

Note: An [Online Knowledge Base](#) is also available with more information and up-to-date content. Please check this knowledge base also.

I added the code in my page but nothing happens.

- Make sure that you have linked the project correctly.
- Make sure that you selected the correct domain before compiling (Project Properties).
- Make sure that you uploaded the **likno-scripts** directory after you compiled your project.

I added a second Tab project to my web page and now nothing works properly.

Make sure that the two projects and their options use different names.

Filenames:

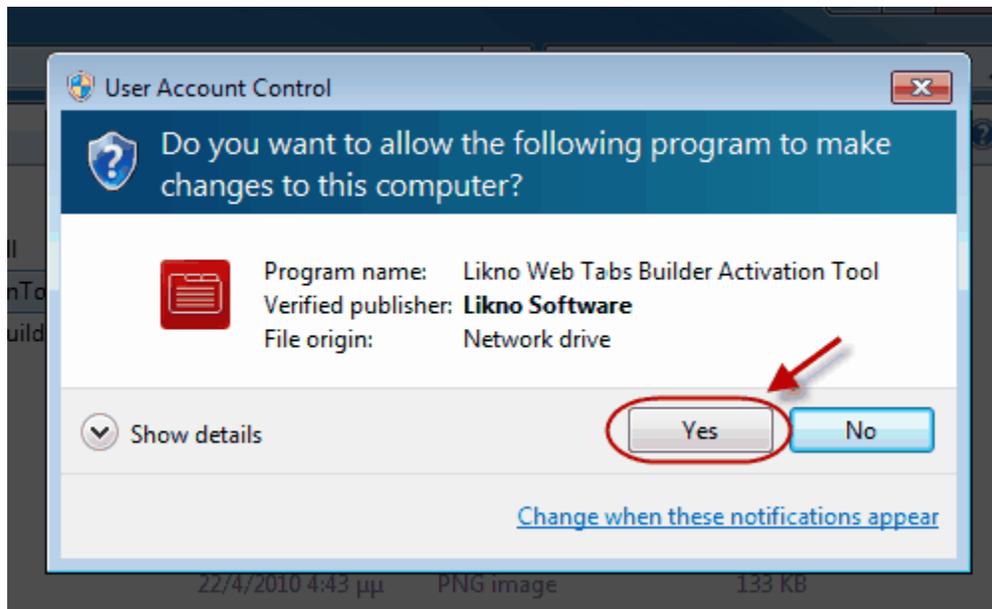
- **Spaces in the compiled project name are not allowed** as they are not parsed properly by some browsers. As an example, use "main_project.js" or "mainproject.js" instead of "main project.js".
- Avoid using filenames that contain special characters, like quotes ("), apostrophes ('), etc. In most cases these characters will produce an error.

8.2 "Full Admin rights" and the activation process

In order to activate Likno Web Tabs Builder you need to be an "administrator" on that PC. Otherwise, the activation process fails.

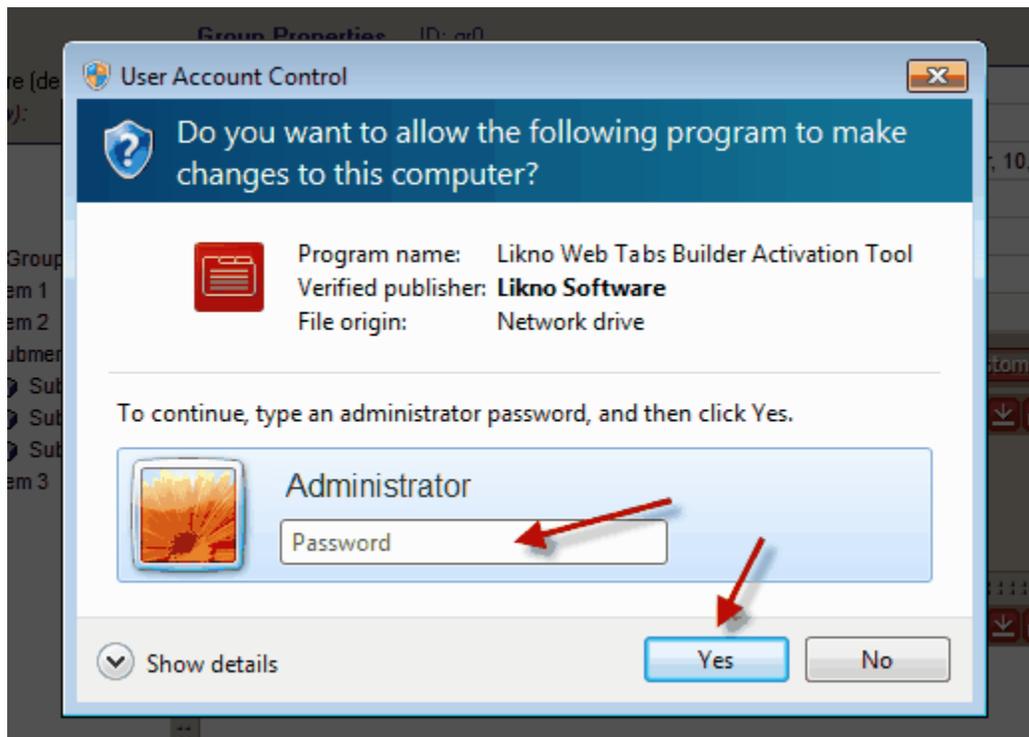
There are some cases (Windows Vista or 7 mostly) where you might think that you are administrator but you miss certain rights that are required by the activation process (you may be a "protected administrator" instead).

To get "full" administrator rights for the activation process, choose **"Yes"** on the following popup window:



(this window appears only for the activation process, you will not see that again after you activate)

If an administrator password is requested and you do not have this, then please contact your administrator to activate Likno Web Tabs Builder through the administrator account.



Index

- A -

About Likno Software 6
Activate 56
Activation 70
Add to page 8
Admin 70
Administrator 70
AJAX 53, 57

- B -

Body CSS 37
Body Source 52, 53

- C -

Close Selected Project 14
Compile 8
Compile Project 14
Compile Properties 16
Container 53
Container CSS 30
Container Height 30
Container Positioning 53
Container Width 30
Content 8, 49, 50, 51, 52, 53
Create Tabs 8

- D -

Delay 26
Direction 26
Dynamic 57

- E -

Edit 9
Effect 22
Embedded 50, 52
Exit 9, 14
External site 53

- F -

Flash 59
Ftp 8

- G -

Generate files 8

- H -

Header Source 49, 50, 51
Headers 42
Headers CSS 31
Headers Outside Container 42, 46
Headers Positioning 42
Headers Within Container 42, 43
Help 19
Hide How 23
Hide When 23
How Do I
 Add effects when a tab body shows or hides 56
 Change the position where the tab control is located on page 57
 Display a video/movie in a Tab? 59
 Find my INSTALLATION ID 56
 Load an external site as the content of a tab body 57
 Load with AJAX the content of a tab body 57
 Properly use paths in the "Website Folder" option 58
 Purchase/Activate 56
 Use multiple Tab projects in the same page 57
 Use the "background-image" value in CSS properties 60

- I -

Ignore header-related CSS 37
Image 51, 52
Installation ID 56

- K -

Keyboard shortcuts 11

- L -

Layout 11
 Library 8
 Likno Web Tabs Builder information 2
 Link 8
 Link Compiled Project to Web Pages 15
 Loop 26

- M -

Main window layout 11

- N -

navigation buttons 26
 New 14

- O -

Open 14
 Options 8

- P -

Pause 26
 Play Tabs 26
 Positioning 53
 Preview Panel Settings 18
 Project 14
 Project Properties 16
 Project Style 8
 Project Type 42, 43, 46
 Properties Overview 22
 Publish 8
 Purchase 56

- R -

Register 56

- S -

Save 9, 14

Save As 14
 Selected Tab 23
 Separator 49
 Settings 8
 Shortcuts 11
 Show How 22
 Show When 22
 Style 8

- T -

Tab Control Populated By 38
 Tab Deselected - Mouse Over state 31
 Tab Deselected state 31
 Tab Selected - Mouse Over state 31
 Tab Selected state 31
 Terminology 3
 Title 49
 Troubleshooting 70

- U -

UAC 70
 Upload 8

- V -

Video 59

- W -

Web Server 57
 What is
 The "Colors detected in this CSS" feature 66
 The "Relative paths detected in this HTML" feature 67
 The HTML menu-linking code 64
 The Internet Explorer Information Bar Warning 64
 What's new! 3

- Y -

YouTube video 59



<http://www.likno.com>