

Look4How

B4J Open Source Application

V0.95

Copyright © Robert Linn, Pinneberg, Germany

30.11.2018

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS for A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

Table of Contents

Overview	4
Getting Started.....	5
Install	5
Create Theme, Category, Memo	7
Main Screen.....	9
Overview	9
Screen Modes	10
Menu Bar.....	12
File menu.....	12
New	12
Open	12
Delete	12
Clear	12
Exit	12
Edit menu	13
Cut	13
Copy	13
Paste	13
Save	13
Open	13
Search menu	14
Quick find.....	15
Find.....	15
Find changed	15
Recent.....	16
Go to item.....	17
Theme menu	18
New	18
Rename.....	18
Clear	18
Delete	18
Select	18
Export.....	18
Category menu	23
New	23
Rename.....	23

Clear	23
Delete	23
Select	23
Export	23
Memo menu.....	24
New	24
New from Clipboard.....	24
Rename.....	24
Delete	24
Move.....	24
Select	24
Export	24
Settings menu	25
Editor	25
Clipboard History	25
Markup	26
Tools menu.....	29
Tasks.....	29
HTML Tag Picker	30
HTML Image Encoder	30
Color Selector.....	31
View menu	32
Always on Top	32
HTML Preview	32
HTML Source	32
Clipboard.....	32
Clipboard History	32
Help menu	34
Help	34
Did you know	34
Author Website	34
About.....	34
Toolbar	35
Item List	36
Editor.....	36
Appendix: List of Files.....	38
Appendix: Export Details.....	39

Placeholder	39
Templates	40
Theme Export	40
Category Export	43
Memo Export.....	44
Appendix: Technical Description.....	46
Development Tool.....	46
Database	47

Overview

Look4How is a Knowledge Management Application to capture information structured by themes with categories and related memos. Theme examples are collections of how-to, code snippets, scribbles, references, documentation and templates.

Look4How has been developed for **personal** use but open for sharing the source code. Further enhancements & improvements are in progress.

Themes can be organized either in a single SQLite database or several SQLite databases:

Single Database with several Themes

Example

Database Look4How with themes like B4X, Python, JavaScript, Snippets etc.

One Database per Theme

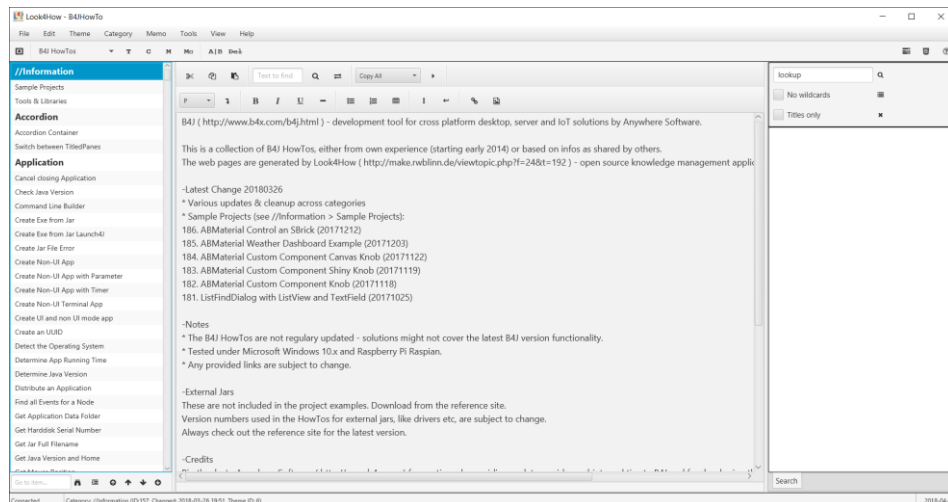
Examples

Database B4X with themes B4J, B4R, B4A, B4i.

Database B4J with themes B4J How-to, B4J Snippets, B4J Templates.

The category or memo content can be kept as plain text or HTML. Beside plain & HTML format, a simple HTML markup line format can also be used with place holders. Content can be exported in text or HTML format.

Screenshot with the B4JHowTo database loaded.



Look4How is written in [B4J](#) (development tool for desktop, server and IoT solutions by [Anywhere Software](#)). Versions used: B4J v6.80 and the Java Development Kit JDK8u192.

Getting Started

Install

Steps to install the Look4How Zip package for Windows (tested with Windows 10).

[Download](#) the Look4How Zip package.

Unzip Look4How.zip to a folder of choice, i.e. Look4How. Following files are extracted:

- Look4How.jar – The core application.
- Look4How.pdf – This help file.
- Look4HowUploader.jar – Tool to upload theme files.

Check if the Java Runtime Environment JRE is installed:

- This is a requirement to run Look4How.
- Recommended is Java 8.
To check the installed Java version, run from the command line:
`java -version.`
Result like:
`java version "1.8.0_192".`

Start Look4How from the command line with command (Java must be on the path)

```
java -jar look4how.jar
```

or

double click on the Windows Explorer with Look4How.jar selected.

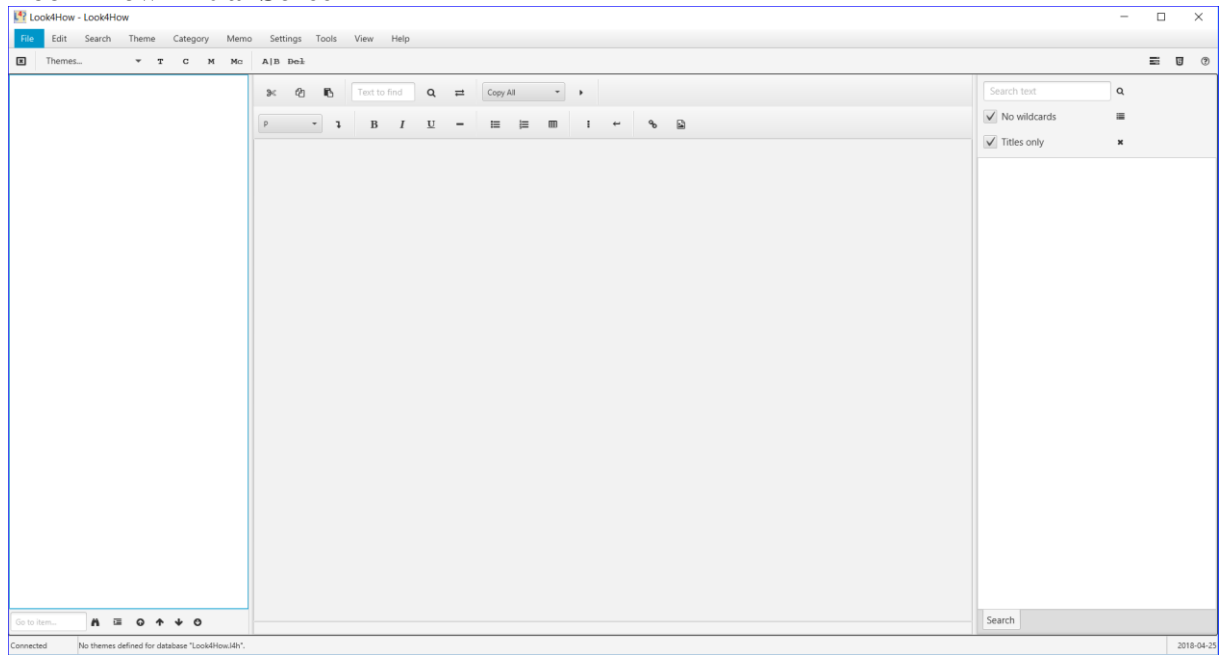
While starting, Look4How creates additional files (see Appendix: List of Files) and the export folder.

The main screen appears, in full screen mode first time, with the status information, that no themes are defined for the default database “Look4How.l4h”. Create a theme with categories and related memos (see Create Theme, Category, Memo).

Notes

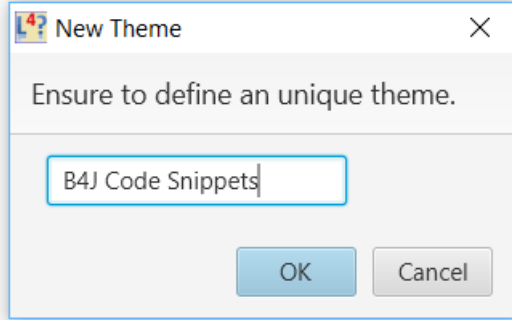
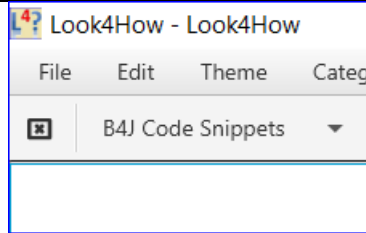
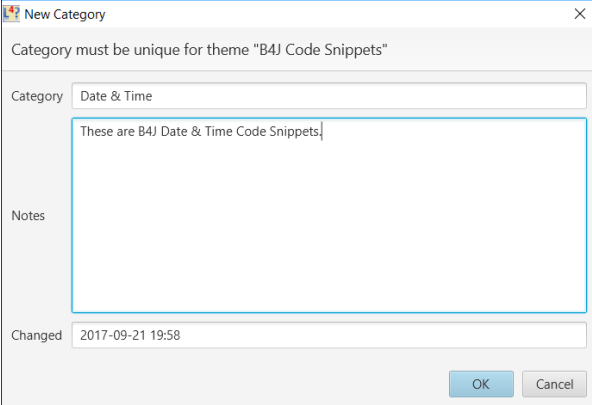
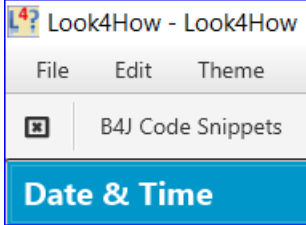
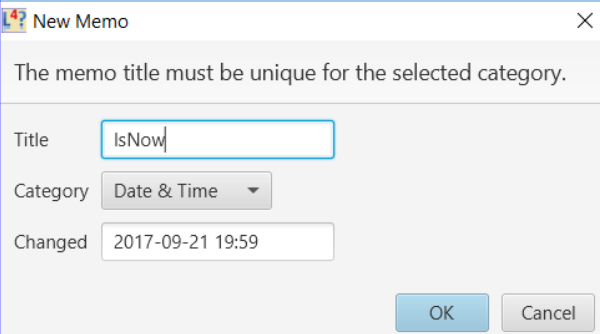
- **Windows 10 & Java9 Desktop Shortcut**
Java9 sets default UI scale to 125%.
If this scale is too big, set the desktop shortcut like:
`<javapath>\java.exe -jar -Dglass.win.uiScale=100%`
`<look4how path>\Look4How.jar`

Look4How Initial Screen



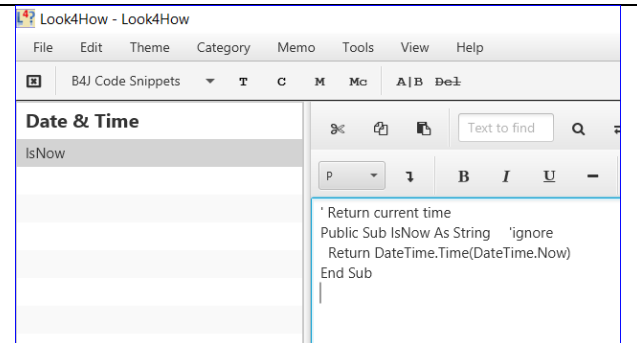
After start, a default database look4how.l4h is created.
Next step is to create a Theme with Categories and related Memos.

Create Theme, Category, Memo

Action	Output
<p>Create a Theme by selecting menu Theme > New Enter a unique Theme. Press OK.</p> <p>Example Theme: B4J Code Snippets.</p>	
<p>The theme “B4J Code Snippets” is listed at the top right combo box. Next step: Create a category for the theme.</p>	
<p>Create a new category by selecting menu Category > New. Enter the Category and the Category Notes. The changed date is set automatically. Press OK.</p> <p>Example category: Date & Time</p>	
<p>The new category “Date & Time” is listed in the list of items.</p>	
<p>Create a new memo for the selected category by selecting menu Memo > New. Enter the title. The category is set automatically (but if more categories then possible to select another category). Press OK.</p> <p>Example Memo Title: IsNow</p>	

Define the memo content in the editor window for the new memo with title “IsNow”.

The memo content is saved automatically.



Main Screen

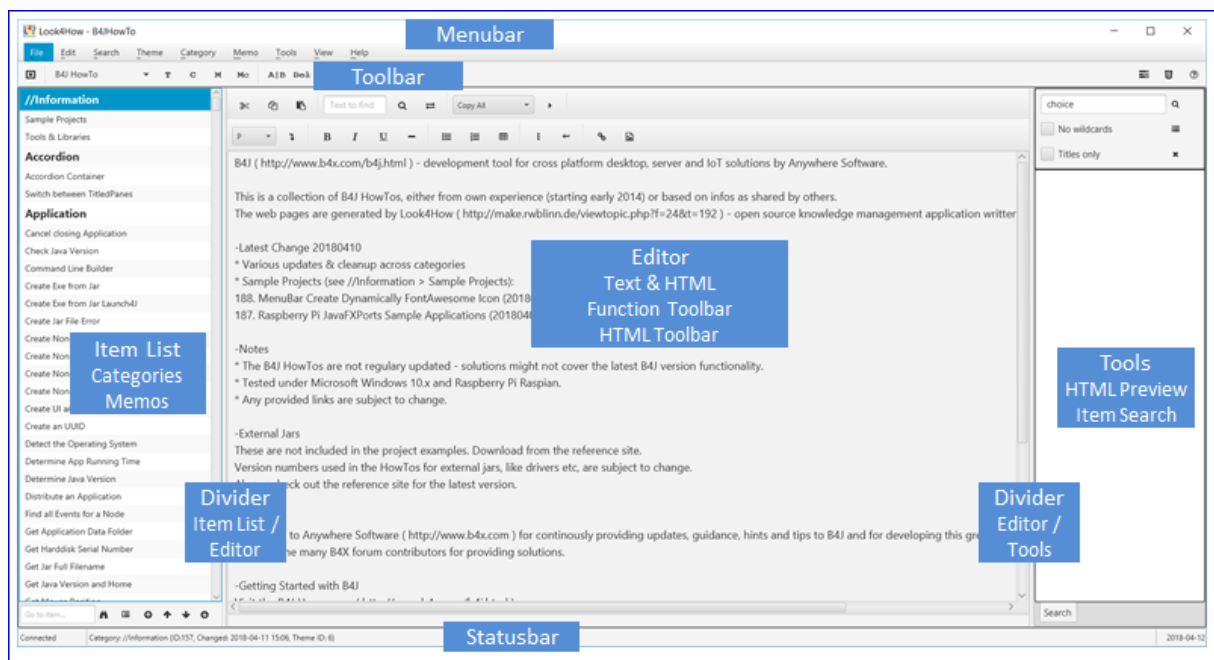
Overview

The main screen has a menu bar at the top with a tool bar underneath.

In the middle there are the 3 panels

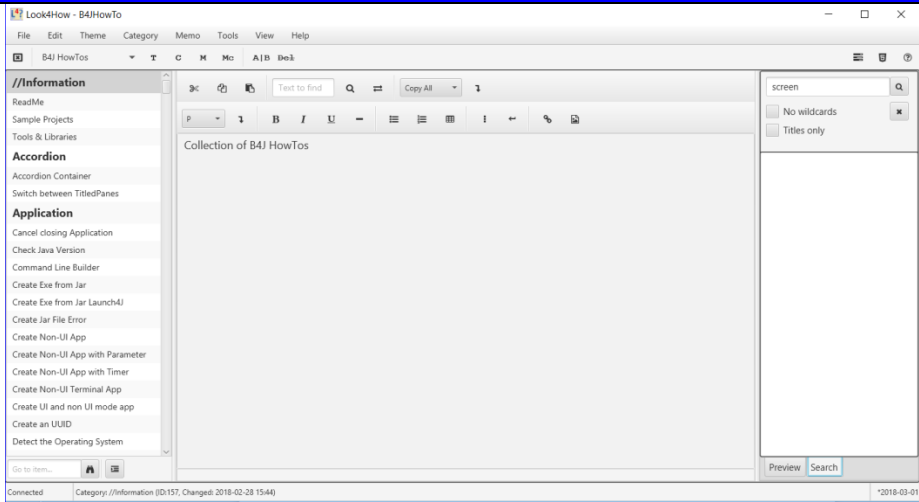
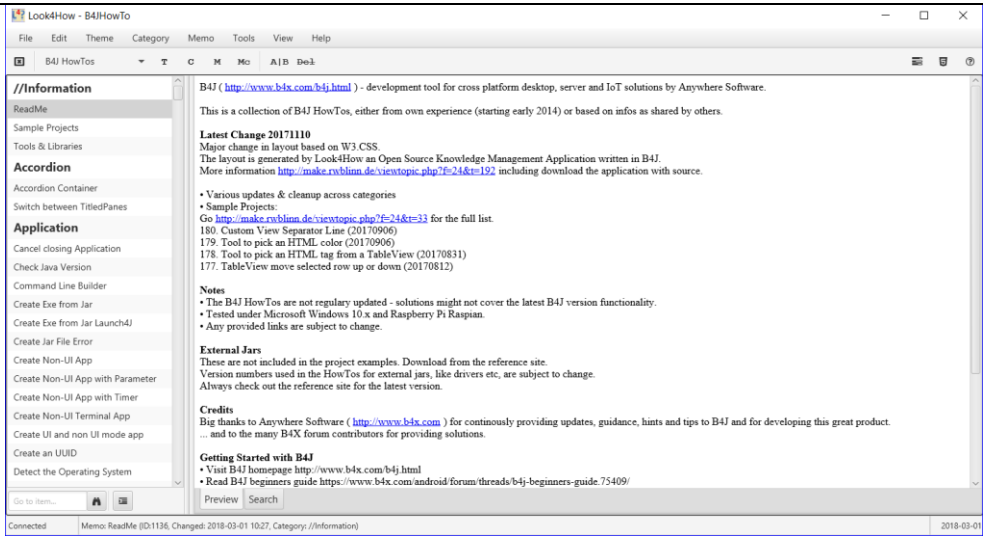
- Item List – listing the Theme Categories with associated Memos
- Editor – for the Category Notes and the Memo Content
- Tools - Sub panels Preview and Search.

At the bottom the status bar with connection information, last message and date with an editor change indicator (*).



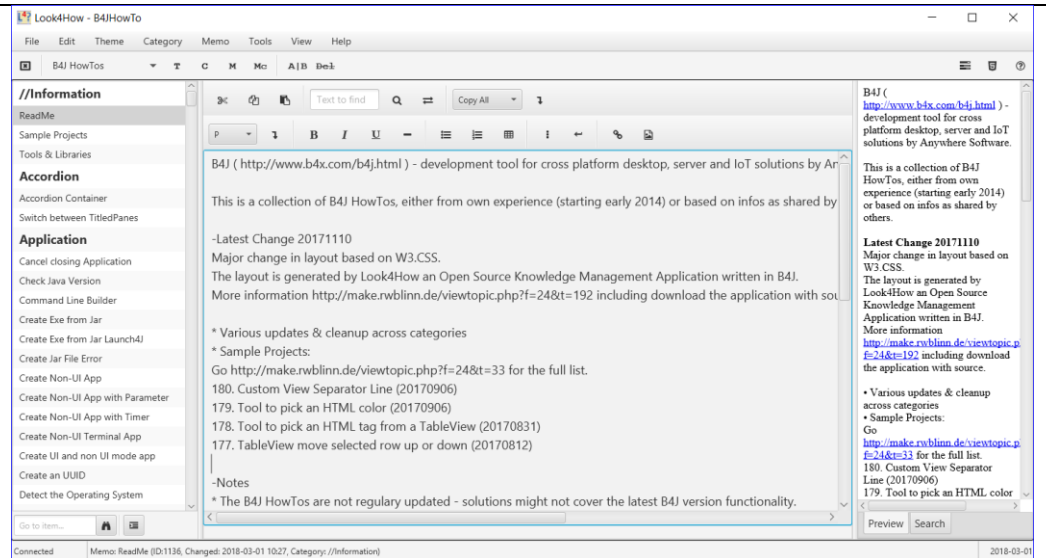
Screen Modes

The dividers can be used to set a mode of preference.

Mode	Screen
Default <p>In the default mode, the three panels are visible. Move the right divider depending editing or view only. In the tools panel, the Preview panel is first shown. Select Search to lookup for items.</p>	
HTML Preview <p>Move the right divider “Editor/Tools” to the left divider “Item List/Editor” and select the Preview panel enables to view the item content in HTML.</p>	

Edit

Move the right divider
“Editor/Tools”
to the far
right. Two
panels remain
visible “Item
List” and
“Editor”



Menu Bar

File menu

The Look4How databases (SQLite) are mandatory located in the application folder. The database default file extension is ".l4h" and should not be changed.

New

Create a new database.

Open

Open an existing database located in the application folder.

Delete

Delete an existing database.

Clear

Clear the content (all themes, categories and memos) of the current database. It is recommended, to make a copy of the database prior clearing.

Exit

Exit the application.
Settings and the last editor changes are saved.
The settings file "Look4How.set" is located in the application folder.

Edit menu

Edit function related to the editor.

Cut

Cut the selected text from the editor.

Copy

Copy the selected text from the editor.

Paste

Paste text content from the clipboard at editor caret position.

Save

Save the editor content to a file.

Open

Load the content of an external file into the editor.

Note: The content of the editor is replaced.

Search menu

The search functions are related to items being a category and/or memo.

The search results are shown in the Search Result List shown in the Search tab of the Tools panel.

The Search tab is always visible.

Search Result List

The list shows the Category title and/or the Memo title with associated category.

By selecting an entry in the list, the category or memo is selected from the list.

Find text field to enter the search text.

Options

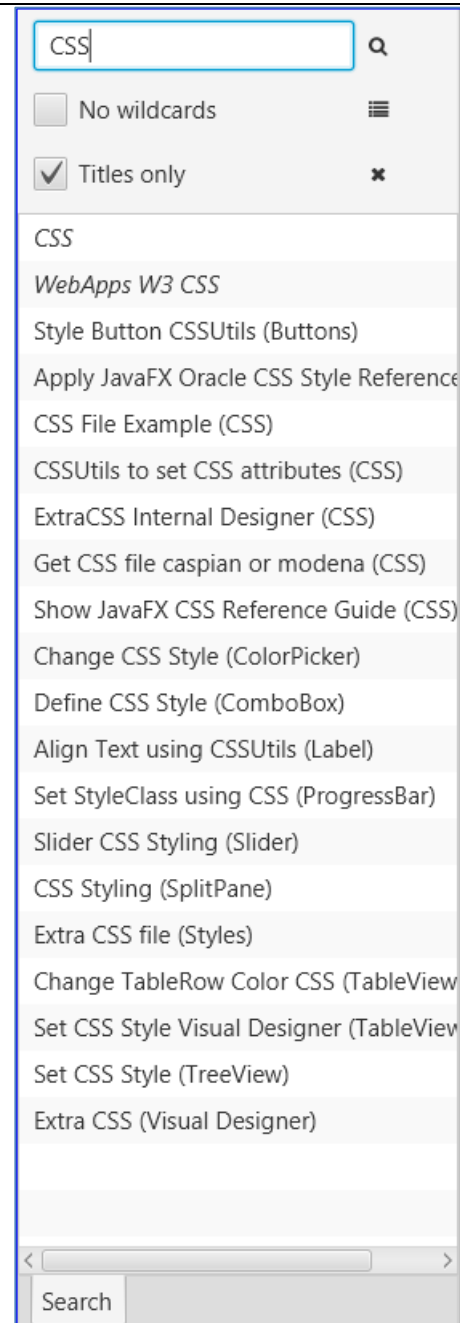
- No wildcards - Do not use wildcards (%text%) to find an exact match.
- Titles only - Search for titles only and not in the category notes and memo content.

Functions

- Search Icon – start searching.
The results are listed in the search result list.
- List Icon - Instead of entering the text, select from a list holding the last 10 find text entries.
- Clear icon: Clear the search result list.

Context Menu

Right click on the search result list, brings up the context menu to copy for all items the titles or content to the clipboard: Copy Titles, Copy Content



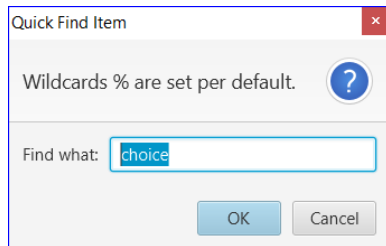
Example search result for find text "CSS" in a B4JHowTo database (B4JhowTo.14h).

Quick find

Quick find items in the list of categories & memos.

An input dialog is shown to add the search text.

The percent sign % wildcard is added to enable matching any sequence of zero or more characters.



Find

Find items, i.e. Categories and Memos, using the Search tab.

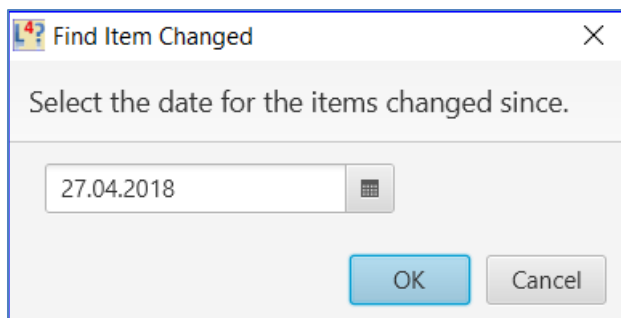
Switches to the Search tab and set as default, the textfield for entering the text.

Find changed

Find items, i.e. categories & memos, which are changed since a certain date.

A date picker is used to select the data. The date picker shows the current date in default format (as used by the device).

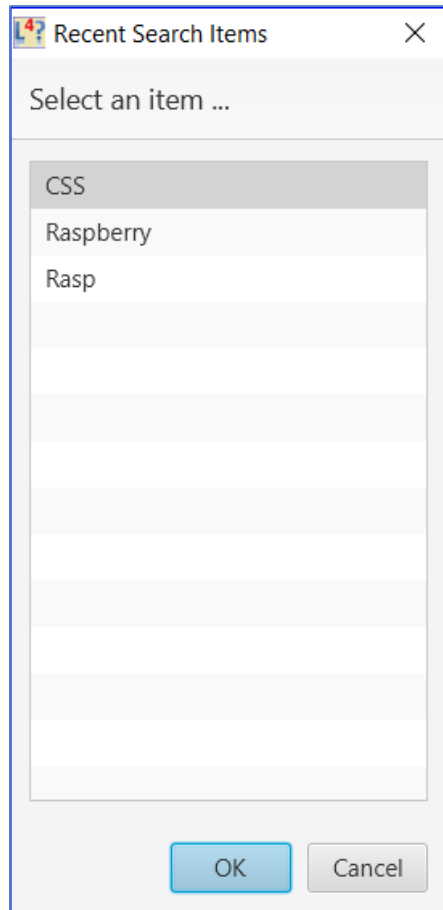
The search result is shown in the **Error! Reference source not found..**



Recent

When searching for an item using the Search Tab, the entered text is stored in the Recent Search Item list.

This menu shows the Recent Search Item list to pick an item from and select in the Item List.

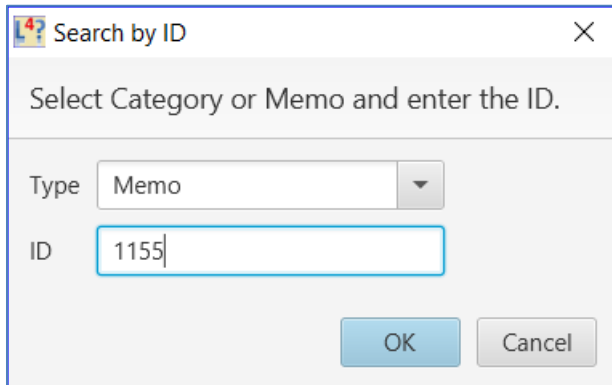


Go to item

Go to an item in the item list.

First, next, previous, last – Jump direct to the item.

ID - A dialog is shown to select the item type Category or Memo and to enter the known ID.

A screenshot of a Windows-style dialog box titled "Search by ID" with a close button (X) in the top right corner. The dialog contains the instruction "Select Category or Memo and enter the ID." Below this, there is a "Type" label followed by a dropdown menu currently showing "Memo". Underneath, there is an "ID" label followed by a text input field containing the number "1155". At the bottom right of the dialog are two buttons: "OK" and "Cancel".

If the ID is found, the item is selected in the Item List.

Theme menu

Manage themes.

New

Create a new theme.

Rename

Rename the theme.

Clear

Clear all categories and associated topics.

Delete

Delete the theme.

Select

Select a theme from the list of themes of the current database.
The list is sorted by theme.

Export

Export the theme.

Clipboard

Copy the content of the theme to the clipboard.

Text

Create text file "theme.txt" in the theme folder and show in default text file viewer.

HTML

Create HTML file "theme.htm" in the theme folder and show in default HTML browser.

HTML Navigation

Export a Theme, to the theme folder, in a *How To* like structure with a menu bar on the left listing the items (the categories & associated memos) and the content on the right (category notes or memo content).

For styling, [w3.css](#) styles are used.

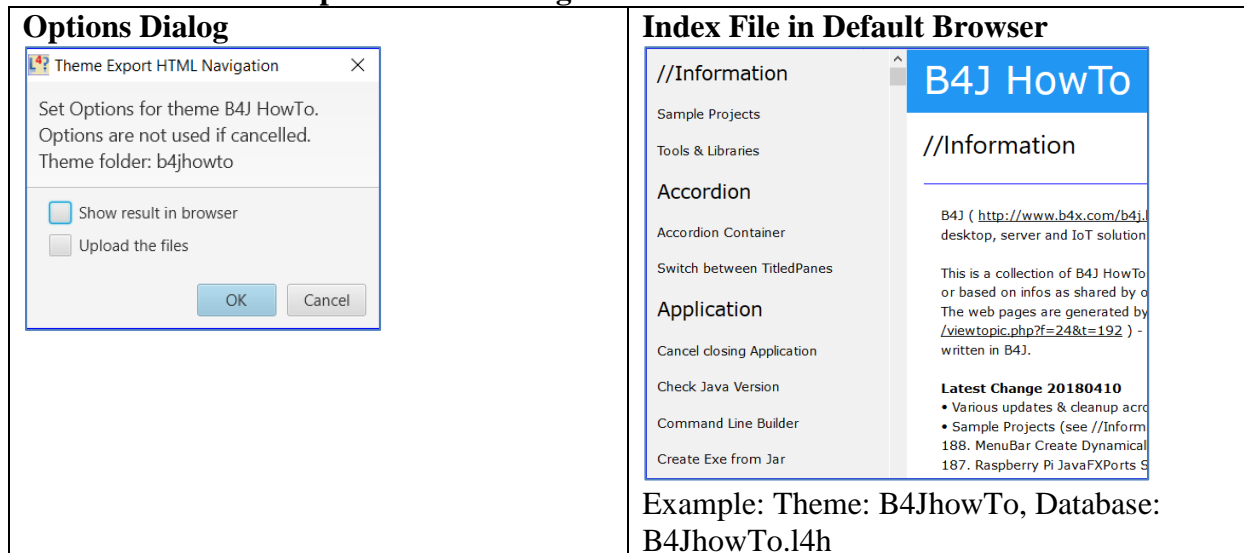
Create Files

A dialog is opened to set options:

- Show result in browser – After the HTML files are created, the default HTML browser is opened with the index file loaded.
- Upload the files – Uploads the created files via FTP to a Remote Server. This is done via external application Look4HowUploader. See **Error! Reference source not found.**

The options are not saved.

Screenshot Theme Export HTML Navigation



HTML Files used & created

- The files are located in a subfolder of the Look4How application. As subfolder name, the theme title is used (in lowercase)
- Each item has its own unique HTML file (which does not change, when updated) using syntax **tTIDcCIDmMID.htm**:
 - T = Theme prefix
 - TID = Theme ID
 - C = Category prefix
 - CID = Category ID
 - M = Memo prefix
 - MID = Memo ID
 - Examples HTML files:
 - Theme with ID 6 and Category with ID 157 = t6c157.htm
 - Theme with ID 6, Category with ID 157 and Memo with ID 1137 = t6c157m1137.htm

- The mandatory styling files w3.css and jquery-1.10.2.min.js are copied from the assets folder.
- The file version.txt holds the date the files have been created.

Screenshot Theme Export HTML Navigation

Files created for theme B4JhowTo in folder <appfolder>/b4jhowto

Name	Ext	Size	Date
⋈ [.]		<DIR>	11.04.2018 16:37
index	htm	134.032	13.04.2018 10:46
jquery-1.10.2.min	js	93.113	11.04.2018 16:37
t6c157	htm	4.320	13.04.2018 10:46
t6c157m1137	htm	50.183	13.04.2018 10:46
t6c157m1138	htm	5.144	13.04.2018 10:46
t6c158	htm	320	13.04.2018 10:46
t6c158m1139	htm	3.851	13.04.2018 10:46
t6c158m1140	htm	2.677	13.04.2018 10:46
t6c159	htm	322	13.04.2018 10:46

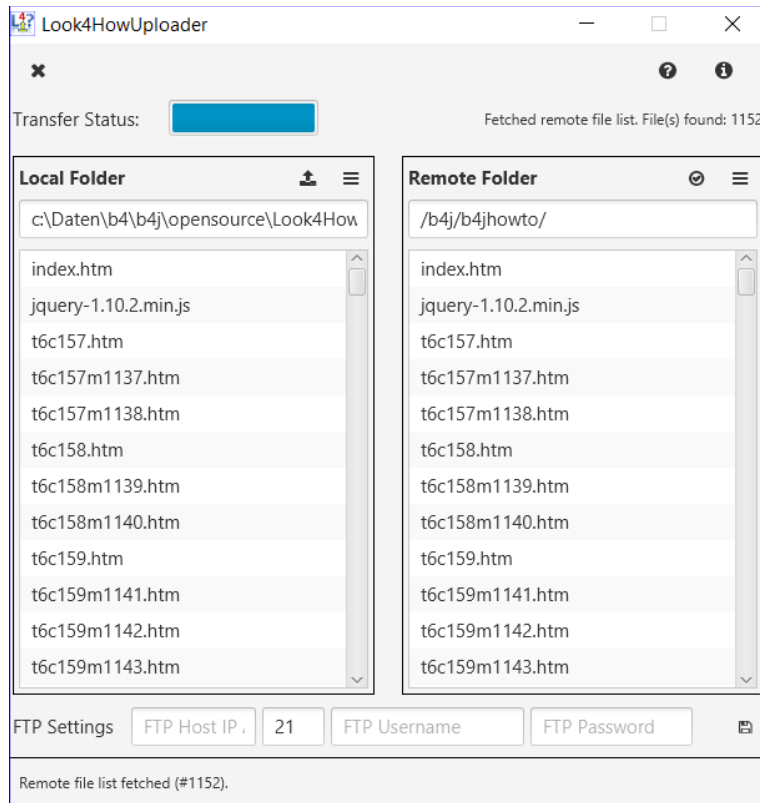
HTML Templates

For the theme, category and memo files following templates, as described under Templates, are used: export-html-theme-navigation.tpl, export-html-theme-navigation-category.tpl, export-html-theme-navigation-memo.tpl.

Upload Files (FTP)

The HTML Navigation files for a Theme can be uploaded from the local device to a remote server via FTP.

The external application Look4HowUploader is used to transfer the files from the selected folder to the remote folder.



To be able to upload, FTP parameters are required. These can be saved and are set when the application is started. It is recommended not to save the FTP password as stored as plain text in the settings file (Look4HowUploader.set).

The Local Folder must be set and the files can be retrieved using the List icon. The location is saved in the settings and reloaded when the application starts.

The Remote Folder must be set and the folder should already exist on the server.

The list of files can be retrieved using the List icon. The response depends on the remote connection speed – while waiting for the result, the Transfer Status progress bar is moving.

To transfer the file from local to the remote folder, click on the Upload icon in the Local Folder pane. During transfer, the Transfer Status progress bar is updated and text is displayed which file is being uploaded.

After uploading the Remote Folder list is updated.

To compare the content of the local folder with the remote folder, click on the Ok icon in the Remote Folder pane. After comparing, a message shows either no differences found or lists the different files found with the option to delete.

If files are deleted, the folders are compared again and should have the same number of files.

Database

Export the selected theme to a database (in the theme folder) with one table.

If the database already exists, it will be deleted.

The database and the single table have the same name, which is created from the theme; blanks are stripped out.

The table structure:

[ID] INTEGER PRIMARY KEY

[MemoTitle] TEXT

[MemoContent] TEXT

[Category] TEXT

[CategoryNotes] TEXT

[Theme] TEXT

Category menu

Manage categories.

New

Create a new category.

Rename

Rename a category.

Clear

Clear all associated memos.

Delete

Delete a category with associated memos.

Select

Select a category from the list of categories of the current theme.
The list is sorted by category.

Export

Export the selected category to text or HTML file(s) in the theme folder.

Clipboard

Copy the content of the selected category to the clipboard.

Text

Create text file "theme-category.txt" and show in default text file viewer.

HTML

Create HTML file "theme-category.htm" and show in default HTML browser.

HTML

Create HTML file "theme-category.htm" and show in default HTML browser.

Replace Field

Replace all occurrences, regardless its case, for the field Category and Notes.

Memo menu

Manage memos.

New

Create a new memo with a unique title for the associated category.

New from Clipboard

Create a new memo from the clipboard content (must be text format).

Rename

Rename a memo title.

Delete

Delete a memo.

Move

Move a memo to another category.

Select

Select a memo from the list of memos of the current theme.
The list is sorted by memo title.

Export

Export the selected memo to text or HTML file in the theme folder.

Clipboard

Copy the content of the selected memo to the clipboard.

Text

Create text file "theme-category-memotitle.txt" and show in the default text file viewer.

HTML

Create HTML file "theme-category-memotitle.htm" and show in default HTML browser.

Replace Field

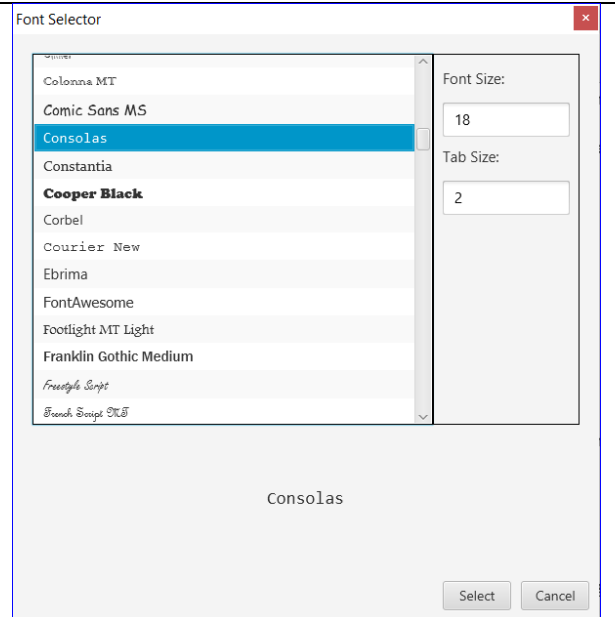
Replace all occurrences, regardless its case, for the field Title and Content.

Settings menu

Setting options which are stored and reloaded from the Look4How settings file (Look4How.set).

Editor

Set the editor font family, font size and tab size.

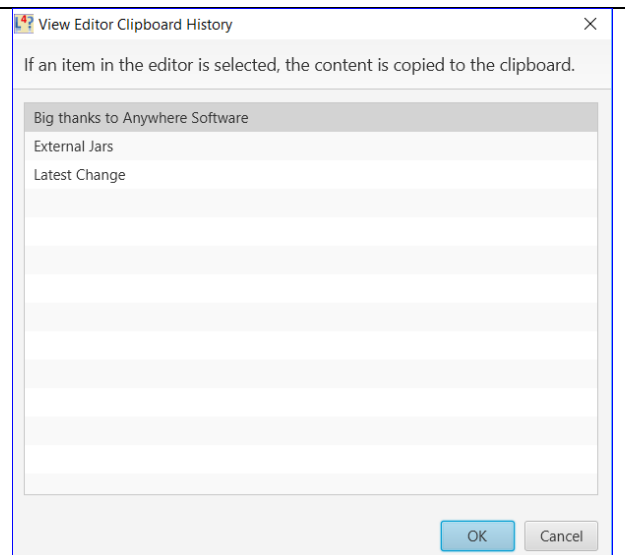


Clipboard History

Text, which is copied from the editor to the clipboard, can be added to the clipboard history list. This list contains only text copied from the editor and not from any other field.

This option enabled to set the clipboard history.

The clipboard history can be viewed under Menu View > Clipboard History.



Markup

Introduction

Markup is a feature to HTML format a page by using Markup tags to format a page. The markups are converted to HTML code.

Markups are set at line start.

The markups are defined in the text file “Markup.map” in the application folder.

The markup file contains a map (Key=Value) with the Markup (Key) and Conversion (Value).

If a **Markup is recognized at the beginning of a line**, it is converted accordingly.

Note

The markup feature is under development. For future releases of *Look4How*, Wiki like markups and a markup manager are planned.

Markup Conversion Table

Markup	Converted to ...	Comment
\ =	<HR>	Horizontal Rule. Do not use the minus sign (–), to define a horizontal rule as the minus sign is used to set a line to bold.
\ =	<HR style="border-top: 3px double #8c8b8b;">	Horizontal Rule with double lines. Do not use the minus sign (–=, to define a horizontal rule as the minus sign is used to set a line to bold.
-	.. 	Bold with line break. Double dots (..) replaced by full line content.
#	<I>.. </I> 	Italic with line break. The special character # is escaped. Double dots (..) replaced by full line content.
	<H1>.. </H1>	Heading Level 1. Double dots (..) replaced by full line content.
img\://		Image file located in the theme folder. Place holder #LOCALPATH# can be used to obtain full path to the local folder. Example: img://#LOCALPATH#myimage.png Image data base64, i.e. img:///data:image/png;base64,DATASTRING
file\://	file\:=<a href\="\"#LOCALPATH##LINE#\">#LINE#	File located in the theme folder. Place holder #LOCALPATH# can be used to obtain full path to the local folder.
*	=• #LINE#	List item.
[CODE]	<CODE>#RB#	Start of a Code block.
[/CODE]	</CODE>	End of a Code block.

Special Actions

- Blanks in Keys must be escaped with a backslash \
- Do not escape blanks in values
- All links containing http, https and mailto are replaced by <a href...> (except images using the key img://).
- If a line starts with "<" and ends with ">", these characters are replaced by the HTML special charcters < and >. This enables to define HTML examples.
- Replacing images with base64 encoded data, can make the HTML preview or export slow as a line can be very long (20000 characters or more).
- Place holder #LINE# to show the current line
- Place holder #RB# to replace blank
- Place holder #LINERB# show current line with blanks replaced
- Place holder #LOCALPATH# can be used to obtain full path to the local folder
- Place holder [CODE]/[CODE] converted into HTML tags CODE and /CODE

The keys and values can be changed using a text editor. Do not remove the first line.

Example Markup

Markup File	Memo Content	Memo Preview
<pre>#Mon Sep 04 13:58:38 CEST 2017 \=\= =<HR> \=\=\= =<HR style="border-top: 3px double #8c8b8b;"> ~=...
 \#=<I>...</I>
 =<H1>...</H1> img\:= file\:=<a href\="#LOCALPATH##LINE#">#LINE# *\ =&bull; #LINE# [CODE]=<CODE>#RB# [/CODE]=</CODE>
=#CR#</pre>	<pre>Code Example [Code] Sub MySub(a as Int, b as Int) as Int Dim Result as Int = a + b Return Result End Sub [/code] This is normal text .</pre>	<pre>Code Example Sub MySub(a as Int, b as Int) as Int Dim Result as Int = a + b Return Result End Sub This is normal text .</pre>

Tools menu

Supplemental tools.

Tasks

Task manager to handle tasks (create, update, delete, search) with fields Task (one liner), priority (High, Normal, Low, On Hold), completed (checkbox) and Status. Option export to HTML or the clipboard.

The tasks are stored in the SQLite database tasks.db with structure:

Table: tasks

Fields:

- [ID] INTEGER PRIMARY KEY
- [Task] TEXT NOT NULL
- [Priority] TEXT
- [Completed] INTEGER
- [Status] TEXT

The screenshot shows a window titled "Tasks" with a search bar and a list of tasks. The tasks are displayed in a table with columns: ID, Task, Priority, and Completed. The table contains 10 rows of tasks, with the first row (ID 29) highlighted. The status section shows "Seeking solution." and a filter for "Normal" priority. The bottom of the window has a "Hide Completed" checkbox and a version number "v20180423".

ID	Task	Priority	Completed
29	NEW: L4H - Menu Search Item Changed DatePi...	Normal	<input type="checkbox"/>
28	NEW: L4H - Upload as a background task the H...	Normal	<input checked="" type="checkbox"/>
27	NEW: L4H - Export Search Result (using the sea...	Normal	<input checked="" type="checkbox"/>
26	NEW: L4H - Replace JavaFX TextArea by RichTe...	On Hold	<input checked="" type="checkbox"/>
25	NEW: L4H - Progressbar showing progress whe...	Normal	<input checked="" type="checkbox"/>
24	NEW: L4H - HTML Color Picker 1) Change Color...	Normal	<input checked="" type="checkbox"/>
23	NEW: L4H - Workaround for Java 9 WARNING: ...	On Hold	<input checked="" type="checkbox"/>
22	NEW: L4H - Option export full name /theme-cat...	Low	<input type="checkbox"/>

HTML Tag Picker

Manage HTML Tags with option to insert the selected tag into the editor at caret position.

The HTML tags to insert are defined as a map in the file HTMLTagPicker.map.

The syntax of the map is:

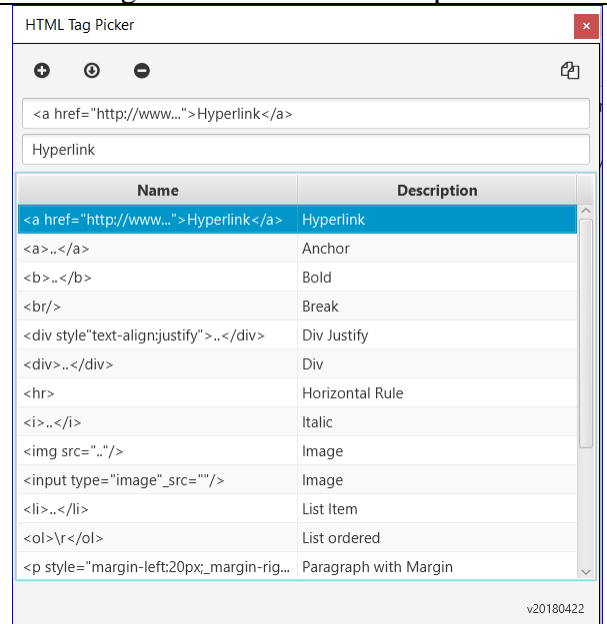
HTML Tag=Brief Description

Examples:

=Break, ..

<i>..

- Two dots (..), between the HTML start and end tag, are used to insert selected text between the start and end tag.
- HTML tag spaces are replaced by underscores, i.e. img_src
- HTML tag list sorted by tag.
- List can be modified using the HTML Tag Picker or with an external editor (file HTMLTagPicker.map, do not change the first line).



HTML Image Encoder

Paste or select an image, encode and insert the HTML IMG tag into the editor at caret position.

Paste an image in PNG format from the clipboard or select from a folder, convert to base64 string and insert into the editor.

The generated HTML tag for the PNG image:

.

Note

that the generated HTML code can become rather large (20K or more).



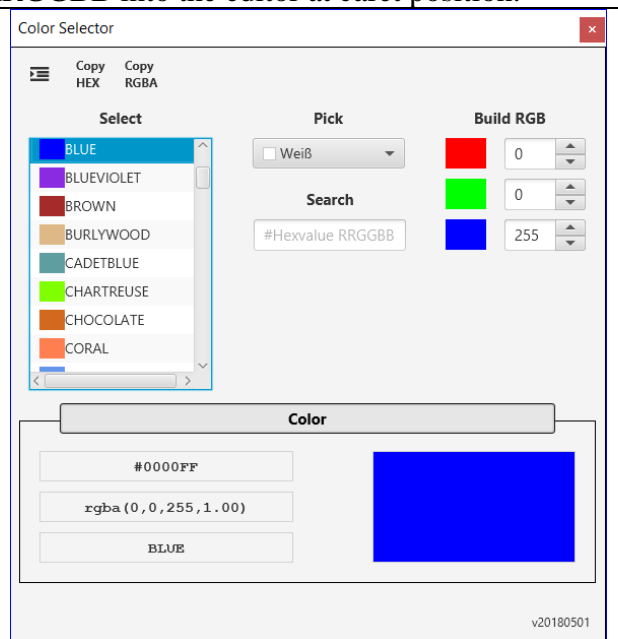
Color Selector

Select a color and insert the color hex value #RRGGBB into the editor at caret position.

Select a color and insert its HEX code in the editor.

There are 4 ways to get a color:

- By Color Name – these are picked from the JavaFX standard list of colors
- By Color Picker – this is the default color picker as provided by windows
- By Color Builder – using Red, Green, Blue sliders with value between 0 – 255
- By Color Search – enter a HEX value in format #RRGGBB



View menu

Always on Top

Keep Look4How always on top.

HTML Preview

Show the content of the editor in a HTML preview window (webview) in the Tools panel.

Clicking on an external link opens the default browser (and reloads the preview content).

If HTML Markup is checked, the preview shows the converted HTML code.

If the content is (pure) HTML, then it is recommended not to use the HTML Markup option (uncheck the option) as this makes the HTML preview slow.

Each change in the editor is displayed in the preview.

The preview window is not shown if this option is not checked.

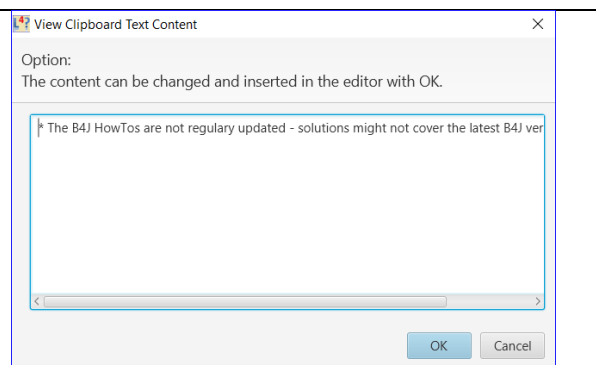
HTML Source

View the HTML source code for the selected item.

This menu item is enabled if the HTML Preview is enabled.

Clipboard

View the content of the clipboard (must be text format).

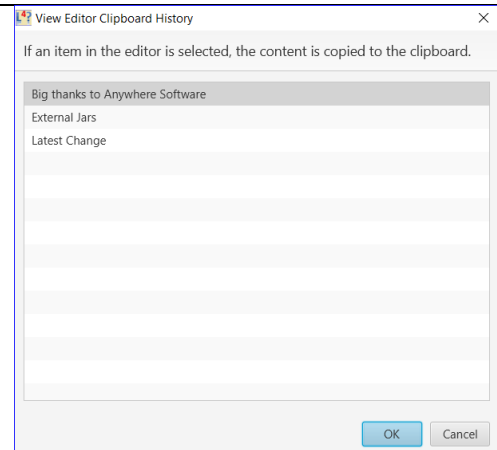


Clipboard History

Text, which is copied from the editor to the clipboard, can be added to the clipboard history list. This list contains only text copied from the editor and not from any other field.

This option enabled to set the clipboard history.

The clipboard history option is set under Menu Settings > Clipboard History.



Help menu

Help

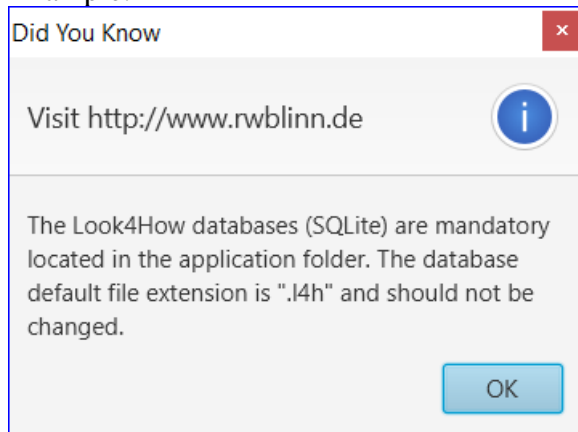
The Look4How documentation (PDF format) shown in the default PDF viewer.

Did you know

Information Dialog showing, randomly, a Look4How hint.

The hints are taken from the Look4How documentation.

Example:



Author Website

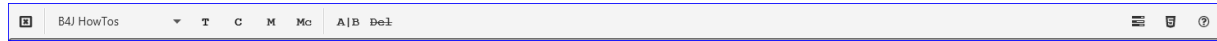
Visit the author's website.

About

Show information about the application Look4How.

Toolbar

The toolbar enabled to quick access functions using buttons or dropdown combo box.



- Close – exit the application
- Combo Box Theme Select showing the selected theme
- New Theme
- New Category
- New Memo
- New Memo from clipboard content
- Rename selected item
- Delete selected item
- Tasks
- HTML Tag Picker
- Help (PDF format)

Item List

The item list contains for the selected theme, the categories and associated memos. The item list is sorted alphabetically by category and memo title.

Key Short Cuts

- Delete - delete the selected item
- F2 - rename the selected item
- Insert - insert a new item
- Double Click – copy the editor content to the clipboard

Go To Toolbar



The GoTo toolbar at the bottom of the item list enables to go to the

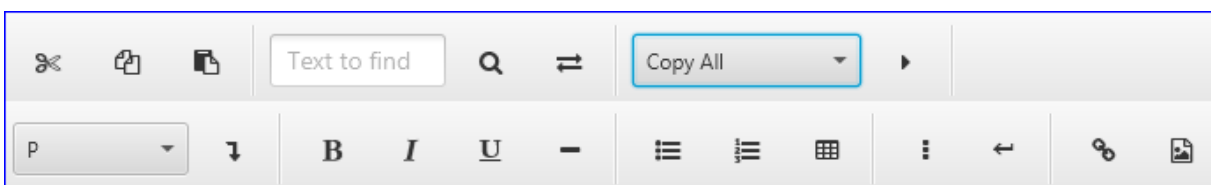
- Title as entered in the text search field,
- Category (selected from a list),
If a title is found and pressing enter or click on the go to icon, the next item is searched for and set accordingly,
- Move to first, previous, next or last list item.

It is not possible to move categories within the list, only memos can be moved to another category.

Editor

The editor is a plain text editor (JavaFX Text input component) with additional functions. There are is a Function toolbar and HTML Tag toolbar at the top, the editor in the middle and a status bar at the bottom.

Toolbars



Function Toolbar (Top)

cut, copy, paste, find & replace text.

Drop down list with "Copy All", "Text lowercase", "Text UPPERCASE", "Insert Date Time", "Insert Date", "Insert Time", "Insert File", "Open URL".

HTML Tag Toolbar (Bottom)

Insert from selection of HTML tags, set bold, italic, underline, unordered or ordered list, table. Add line break to current line or all lines.

Drag & Drop

The editor accepts text to be dropped from another application.

Appendix: List of Files

The files listed in the application folder.

Filename	Description
Look4How.jar	The application. Requires the Java JRE environment to be installed. If Java is not on the file path, start the application with: <pathtojava>java.exe -jar Look4How.jar.
Look4How.pdf	Help file.
Look4How.l4h	Default database containing themes, categories and memos. This database is created if Look4How is started first time. Look4How uses SQLite as the underlying database.
Look4How.set	Holds the settings which are saved and loaded. The settings are using a map (settings key=value). This file can be modified with a text editor. Do not change the first line.
<name>.l4h	Look4How SQLite database for a dedicated theme. Examples b4jhowto.l4h, nodered.l4h.
Export<name>.tpl	Templates are used for exporting a theme, category or memo (see Appendix: Export) The template files are located in the application folder. The templates can be modified using a text editor.
HTMLTagPicker.map	HTML tags used by the tool HTML Tag Picker. The HTML tags are defined using a map. This file can be modified with a text editor. Do not change the first line. Special characters needs to be escaped using the backslash character \, i.e # as \#, space as \ .
Markup.map	Place holders in the plain text are converted to HTML tags and shown in the preview or HTML export (see Markup). The place holders are defined using a map. This file can be modified with a text editor. <i>Do not change the first line.</i> Examples: -:..<, \#:<I>..<</I> Special characters needs to be escaped using the backslash character \, i.e # as \#, space as \ .
Tasks.db	Tasks SQLite database used by the tool Tasks.

Appendix: Export Details

Theme, category or memo can be exported in text or HTML format.

Depending on what item to export, a file template (“export...tpl”) with place holder is used.

The place holders are replaced by the respective content from the database or an action is taken, like populating a list of memos.

Placeholder

Placeholders are defined by the tag #Placeholder.

Important: The place holders are case sensitive.

Placeholder	Description (Example)
Database	
#Database	Database opened as displayed in the title (B4X).
Theme	
#ID	Theme ID (3)
#Theme	Theme selected (Scribbles).
#Categories	Categories with notes and memo(s) content for the selected theme.
#CategoryList	List of categories ordered by category for the selected theme.
#MemoList	List of memos ordered by memo title for the selected theme.
#CategoryMemoList	List of categories with memos ordered by category and memo title for the selected theme.
#CategorySelect	Drop down list of categories ordered by category for the selected theme.
#CategoryMemoSelect	Drop down list of categories with memos ordered by category and memo title for the selected theme.
Category	
#ID	Category ID (12)
#Category	Category selected (Python).
#Notes	Category notes (Python).
#Changed	Date category changed (2017-09-15 11:11).
#Memos	List of memos with content for the category.
#MemoList	List of memos ordered memo title for the selected category.
#MemoSelect	Drop down list of memos ordered by memo title for the selected category.

Memo	
#ID	Memo ID (426)
#Title	Memo title (Readme).
#Content	Memo content (This is the readme content).
#Changed	Date memo changed (2017-09-15 11:11).
Special	
Planned next version.	

Templates

Templates with placeholders are used to export a theme, category or memo.
The default templates ...

- Are located in the application folder.
- Can be changed according requirements.
- Are rather simple without sophisticated format or HTML CSS styling.

Theme Export

Template Filename	Content Example
TEXT FORMAT	The file export-text-theme.tpl has a place holder #Categories, which uses the file export-text-theme-category.tpl. In export-text-theme-category.tpl a place holder #Memos is defined which uses export-text-theme-memo.tpl
export-text-theme.tpl	Theme: #Theme Database: #Database Categories: #Categories
export-text-theme-category.tpl	#Category Notes: #Notes Changed: #Changed ID: #ID Memos: #Memos
export-text-theme-memo.tpl	Title: #Title Content: #Content Changed: #Changed ID: #ID Category:

	#Category
HTML FORMAT	Same principle as text format. Note the anchors used for the Categories () and Memos ().
export-html-theme.tpl	<pre> <!DOCTYPE html> <HTML> <!-- Look4How Category Theme --> <TITLE>#Theme</TITLE> <HEAD> <STYLE> table {width: 100%;border: 1px solid #cef;text-align: left; } th { font-weight: bold;background-color: #acf;border-bottom: 1px solid #cef; } td,th {padding: 4px 5px; } .odd {background-color: #def; } .odd td {border-bottom: 1px solid #cef; } a { text-decoration:none; color: #000;} h1 {color: green;} h2 {color: red;} h3 {color: blue;} ul {color: red} #author {font-size:0.8em;color:blue } </STYLE> </HEAD> <BODY> <DIV ALIGN="RIGHT">Database: #Database</DIV> <H1>Theme: #Theme</H1> <HR> <FORM name="formselect"> Select Category or Memo: <SELECT onchange="if (this.value) window.location.href=this.value"> #CategoryMemoSelect </SELECT> </FORM> <HR> #Categories <P id="author">Look4How by Robert W.B. Linn</P> </BODY> </HTML> </pre>
export-html-theme-category.tpl	<pre> <!-- CATEGORYBEGIN --> <H2>#Category</H2> <TABLE> <TR><TD>#Notes</TD></TR> <TR><TD>Changed: #Changed, ID: #ID</TD></TR> <TR><TD>Top</TD></TR> <TABLE> <HR> #Memos <!-- CATEGORYEND --> </pre>
export-html-theme-memo.tpl	<pre> <!-- MEMOBEGIN --> <H3>#Title</H3> <TABLE> <TR><TD>#Content</TD></TR> <TR><TD>Changed: #Changed, Category: #Category, ID: #ID</TD></TR> <TR><TD>Top</TD></TR> <TABLE> </pre>

	<!-- MEMOEND -->
--	------------------

Category Export

Same applies as for Theme export.

Template Filename	Content Example
TEXT	
export-text-category.tpl	Category: #Category Theme: #Theme Database: #Database #Notes ID: #ID Changed: #Changed Memos: #Memos
export-text-category-memo.tpl	Title: #Title Content: #Content ID: #ID Changed: #Changed
HTML	
export-html-category.tpl	<pre> <!DOCTYPE html> <HTML> <TITLE>#Category</TITLE> <HEAD> <STYLE> table {font-family: arial, sans-serif; border-collapse: collapse; width: 100%; } td, th {border: 1px solid #dddddd; text-align: left; padding: 4px; } tr:nth-child(even) {background-color: #f2f2f2;} th {background-color: #FF0000; color: white; } h2 {color: red;} h3 {color: blue;} #author {font-size:0.8em;color:blue } </STYLE> </HEAD> <BODY> <DIV ALIGN="RIGHT">Theme: #Theme, Database: #Database</BR></DIV> <HR> <H2>#Category</H2> <TABLE> <TR><TD>#Notes</TD></TR> <TR><TD>ID: #ID, Changed: #Changed</TD></TR> <TABLE> <HR> #MemoList </pre>

	<pre> <HR> <FORM name="formselect"> <SELECT onchange="if (this.value) window.location.href=this.value"> #MemoSelect </SELECT> </FORM> <HR> #Memos <P id="author">Look4How by Robert W.B. Linn</P> </BODY> </HTML> </pre>
export-html-category-memo.tpl	<pre> <!-- MEMOBEGIN --> <H3>#Title</H3> <TABLE> <TR><TD>#Content</TD></TR> <TR><TD>ID: #ID, Changed: #Changed</TD></TR> <TABLE> <!-- MEMOEND --> </pre>

Memo Export

Same applies as for Theme export.

Template Filename	Content Example
TEXT	
export-text-memo.tpl	<pre> Title: #Title Content: #Content ID: #ID Changed: #Changed, Category: #Category </pre>
HTML	
export-html-memo.tpl	<pre> <!DOCTYPE html> <HTML> <TITLE>#Title</TITLE> <HEAD> <STYLE> table {font-family: arial, sans-serif; border-collapse: collapse; width: 100%; } td, th {border: 1px solid #dddddd; text-align: left; padding: 4px; } tr:nth-child(even) {background-color: #f2f2f2;} th {background-color: #FF0000; color: white; } code { font-family: monospace; } #author {font-size:0.8em;color:blue } </STYLE> </HEAD> <BODY> <DIV ALIGN="RIGHT">Theme: #Theme, Database: #Database</BR></DIV> <HR> <H3>#Title</H3> </pre>

```
<TABLE>
<TR><TD>#Content</TD></TR>
<TR><TD>ID: #ID, Changed: #Changed, Category: #Category</TD></TR>
<TABLE>
<HR>
<P id="author">Look4How by Robert W.B. Linn</P>
</BODY>
</HTML>
```

Appendix: Technical Description

This chapter provides technical information about the application.

Development Tool

Look4Now is Open Source developed with [B4J](#) from [Anywhere Software](#).
Versions used: B4J v6.80, Java Development Kit v8u192.

The goal is to use as much of the core libraries as possible and develop missing functionality within Look4How except for

- dialogs (variety of dialogs used provided by the additional library jRLDialogsx) ,
- handling of key events (managed by the additional library jReflection).

Libraries used

Core: jcore, jfx, cssutils, javaobject, jstringutils, jsq1,json

Additional: jreflection , jrldialogsx

Static Code Modules

- Constants – global constants

Classes

- DBManager – All database operations are handled by the DBManager with some help of DBUtils
- DBUtils – additional database operations – planned to integrate in the DbManager
- Utils – generic utility subs structured by area, i.e. File, Form, Date& Time, Views.
- HTMLTagPicker – Tool to pick and insert a HTML tag in the editor.
- ColorSelector – Tool to pick and insert a HTML color value in the editor.
- HTMLImageEncoder – Tool encode images toBase64 Uri and insert in the editor.
- HTMLTextEditor – Custom View Text & HTML editor
- SeparatorLine – Custom View Toolbar Separator line
- Tasks – Tool to manage tasks (Fields: Task, Priority, Completed, Status)

Database

The themes, categories and memos are stored in an SQLite database, which file is (mandatory) located a subfolder of the application folder and has the same name as the theme folder.

Example

Look4How Folder: c:\Daten\look4how

B4J HowTo Subfolder: c:\Daten\look4how\B4JHowTo\

B4J HowTo SQLite Database: c:\Daten\look4how\B4JHowTo\B4JHowTo

The database scheme (SQL create statement) for the tables **themes**, **categories**, **memos**:

- CREATE TABLE If Not EXISTS [themes] ([ID] INTEGER PRIMARY KEY, [Theme] TEXT)
- CREATE TABLE If Not EXISTS [categories] ([ID] INTEGER PRIMARY KEY, [Category] TEXT, [Notes] TEXT, [Theme] INTEGER)
- CREATE TABLE If Not EXISTS [memos] ([ID] INTEGER PRIMARY KEY, [Title] TEXT, [Changed] TEXT, [Content] TEXT, [Category] INTEGER)

Table Relationships

- Memo is related to a category which is related to a theme.
- Theme can have 1:n categories - category assigned to 1 theme.
- Category must be unique per theme.
- Category can have 1:n memos - memo assigned to 1 category.
- Memo title must be unique per category.