

EXT: LESS for TYPO3

Extension Key: t3_less

Language: en

Version: 0.5.2

Keywords: less, less-css, leafo.net/lessphp, lesscss.org, less.js, LESS for TYPO3

Copyright 2006-2013, David Greiner, <hallo@davidgreiner.de>

This document is published under the Open Content License
available from <http://www.opencontent.org/opl.shtml>

The content of this document is related to TYPO3
- a GNU/GPL CMS/Framework available from www.typo3.org

Table of Contents

EXT: LESS for TYPO3.....1	Constants-Editor.....9
Introduction.....3	Import-Directories.....10
Users manual.....4	Hooks.....11
Individual filesettings.....5	Register custom functions.....12
TS-Reference.....7	Known problems.....13
	Thanks to.....14

Introduction

What does it do?

“LESS for TYPO3” makes it possible to use LESS-CSS on a very easy way in TYPO3.

Users manual

Please follow the following steps in order to use this extension.

1. Install the extension using the extension manager
2. Include the static template file
3. choose your favourite compiler using the constants editor
4. define paths to less-folder and css-output folder using the constants editor
5. store your less-files in the defined less-folder
6. clear cache and reload your page
7. If you are using php based compiler, you will find the compiled files in the defined output-folder

Individual filesettings

It is possible to define some settings for each file stored in the less-folder. The following TS is an example for using two less-files. Configuration is splitted for using php-based compiler and for using js-based compiler.

```

plugin.tx_t3less {
    ##Settings only for phpcompiler
    phpcompiler {
        filesettings {
            ## "style" = Filename of your lessfile without ".less"-extension (source file is style.less)
            style {
                media = only screen and (max-device-width: 480px)
                title = beispiel
                compress = 1
                forceOnTop = 1
                allWrap = ""
                excludeFromConcatenation = 0
                sortOrder = 10
            }
            style2 {
                sortOrder = 20
            }
            style3 {
                excludeFromPageRenderer = 1 #this file will not be included via pagerenderer
            }
            ## "iehacks" = Filename of your lessfile without ".less"-extension (source file is iehacks.less)
            iehacks {
                allWrap = <!--[if IE]> | <![endif]-->
            }
        }
    }
    jscompiler {
        filesettings {
            style {
                media = screen
                title = lessdatei
                allWrap = ""
                sortOrder = 10
            }
        }
    }
}

```

```
    }  
    iehacks {  
        media = all  
        title = iehacks  
        allWrap = <!--[if IE]> | <![endif]-->  
        sortOrder = 500  
    }  
}  
}  
}
```

TS-Reference

Options for using **PHP-compiler**

plugin.tx_t3less.**phpcompiler**.filesettings.LESS-FILENAME-WITHOUT-EXTENSION {

Property:	Data type:	Description:	Default:
media	String	The media attribute. Like "screen", "print" or media queries like "only screen and (max-device-width: 480px)"	all
title	String	The title attribute. A title for this stylesheet like "general" or "iehacks" or something else	By default there will be no title
compress	Integer	Options are 1 (activated) or 0 (deactivated). This setting is for usage of page.config.compressCss	1
forceOnTop	Integer	Options are 1 (activated) or 0 (deactivated). With this setting you can force to include this file at the top of your html-markup	0
allWrap	String	If you want to wrap the included file, please use "allWrap". Example: <!--[if IE]> <![endif]-->	By default there will be no wrap
excludeFromConcatenation	Integer	Options are 1 (activated) or 0 (deactivated). This setting is for usage of page.config.concatenateCss	0
excludeFromPageRenderer	Integer	With this setting you can exclude this file from pagerenderer. If activated, the file will not be included to your page.	0
sortOrder	Integer	Lowest input will be included first, highest as latest	By default there is no sortOrder given

Options for using **JS-compiler**

plugin.tx_t3less.**jscompiler**.filesettings.LESS-FILENAME-WITHOUT-EXTENSION {

Property:	Data type:	Description:	Default:
media	String	The media attribute. Like "screen", "print" or media queries like "only screen and (max-device-width: 480px)"	all
title	String	The title attribute. A title for this stylesheet like "general" or "iehacks" or something else	By default there will be no title
allWrap	String	If you want to wrap the included file, please use "allWrap". Example: <!--[if IE]> <![endif]-->	By default there will be no wrap
excludeFromPageRenderer	Integer	With this setting you can exclude this file from pagerenderer. If activated, the file will not be included to your page.	0

Property:	Data type:	Description:	Default:
sortOrder	Integer	Lowest input will be included first, highest as latest	By default there is no sortOrder given

All the other settings are senseless by using js-compiler, so they have the following static values.

compress = FALSE

forceOnTop = FALSE (changed from true to false since v0.5.1, cause sortOrder doesn't work together with it)

excludeFromConcatenation = TRUE

Constants-Editor

There are some further settings in the constants-editor.

Constant:	Data type:	Description:	Default:
plugin.tx_t3less.enable.mode	String (select)	Choose which compiler do you want to use. PHP-Compiler / JS-Compiler	PHP-Compiler
plugin.tx_t3less.files.pathToLessFiles	String	Path to an folder which contains your less-files	fileadmin/t3_less/lessfiles
Following settings are only important if you use the php-compiler			
plugin.tx_t3less.files.outputFolder	String	Path to an folder which will contain your compiled files if you use php-compiler	fileadmin/t3_less/cssfiles
plugin.tx_t3less.other.activateCompiler	Boolean	It is recommended to deactivate compiler if you don't have some new less-files to save serverpower	1
plugin.tx_t3less.other.unlinkCssFilesWithNoSourceFile	Boolean	If you deactivate this settings, files will not longer be deleted if their source file change (this is why compiled files are getting a md5-hash in their filenames)	1
plugin.tx_t3less.other.compressed	Boolean	CSS file will be compressed if you check this (recommended)	1
plugin.tx_t3less.other.forceMode	Boolean	ALWAYS generate new file (not recommended for production)	0
plugin.tx_t3less.other.importDirs	String	Comma seperated list of pathes to folders which should be used for imports.	
plugin.tx_t3less.other.lessJsScriptPath	String	Path to JS compiler script	EXT:t3_less/Resources/Public/Js/less-1.3.3.min.js

Import-Directories

It is possible to define one or more directories in which the phppcompiler will search for files which are included by using `@import "lessfile.less"`; in your less-files.

For example:

You have defined

```
"plugin.tx_t3less.other.importDirs = fileadmin/less/variables/, fileadmin/less/colors/"
```

```
"plugin.tx_t3less.files.pathToLessFiles = fileadmin/less/"
```

 "in the constants editor

And in your less file "fileadmin/less/main.less" you have defined

```
@import "variables.less";
```

Now, the phppcompiler is searching for "variables.less" in both defined folders. This has one big advantage:

If you have a file variables.less which contains only variables and which is not directly in the "pathToLessFiles" - folder, this file will not be compiled to a 0-byte file, but the variables will be available in all your compiled less files.

Please read http://leafo.net/lessphp/docs/#import_directory for more informations and a better explanation than mine ;)

Hooks

There is a hook to pass less files from other extensions. It is very easy to use. Just add following line to ext_localconf.php from your extension.

```
$GLOBALS['TYPO3_CONF_VARS']['EXTCONF']['t3less']['addForeignLessFiles'][] =  
'path/to/a/further/folder/with/less/files/';
```

All *.less-files in the folder defined this way will be included too and can be configured by the known way.

Register custom functions

There is the possibility to register custom functions. Please take a look at http://leafo.net/lessphp/docs/#custom_functions

Using this in t3_less is very easy. There is an example class in EXT:t3_less/Classes/UserFunction/class.user_exampleClass.php which is configured via TypoScript:

```
plugin.tx_t3less {
    ##Settings only for phpcompiler
    phpcompiler {
        registerFunctions {
            #Example user function
            double =
EXT:t3_less/Classes/UserFunction/class.user_exampleClass.php:user_exampleClass-
>exampleFunction
        }
    }
}
```

Known problems

There are no known problems at the moment.

If you find a bug or something else, please let me know and contact me:

`hallo@davidgreiner.de`

or post it on forge:

http://forge.typo3.org/projects/extension-t3_less

Thanks to

Thanks to leafo.net for developing this awesome less-php-compiler.

<http://leafo.net/lessphp/>

Thanks to lesscss.org for developing this amazing css-framework.

<http://lesscss.org/>