



FREE eBook

LEARNING

typo3

Free unaffiliated eBook created from
Stack Overflow contributors.

#typo3

Table of Contents

About.....	1
Chapter 1: Getting started with typo3.....	2
Remarks.....	2
Versions.....	2
Examples.....	2
Classic installation on a Unix server.....	2
TYPO3 CMS.....	2
Chapter 2: Add custom class for links in RTE.....	4
Introduction.....	4
Examples.....	4
Add custom link classes.....	4
Chapter 3: Basic typoscript for TYPO3 installation.....	5
Introduction.....	5
Examples.....	5
Require Config Typoscript.....	5
get Page content :.....	5
Add css and js file:.....	6
Remove Controller and Action name In URL.....	6
include typoscript:.....	7
Chapter 4: Database Query DataProcessor - Examples.....	8
Examples.....	8
How to get selected records from another table (using TCA group field).....	8
Chapter 5: Extbase Extension Development in TYPO3.....	10
Examples.....	10
Introduction.....	10
Chapter 6: Fluid templating in TYPO3.....	11
Examples.....	11
Typoscript setup.....	11
Example Fluid Template File.....	12
Chapter 7: Image cropping TYPO3 8.7 LTS.....	14

Examples.....	14
Usage and configuration of image cropping.....	14
Chapter 8: Installation & Setup.....	17
Examples.....	17
Installation with composer.....	17
Chapter 9: Links to official documentation Sites.....	18
Examples.....	18
Official documentation.....	18
Main Documentation.....	18
API documentation.....	18
Chapter 10: Realurl with TYPO3.....	19
Examples.....	19
Introduction & Setup.....	19
Chapter 11: TypoScript.....	22
Syntax.....	22
Examples.....	22
Basic TypoScript : Hello World.....	22
Get date in object.....	22
Create link.....	23
Image and Image Resource.....	23
Get current page title.....	25
Get Current URL.....	25
Credits.....	26

About

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [typo3](#)

It is an unofficial and free typo3 ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official typo3.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Chapter 1: Getting started with typo3

Remarks

TYPO3 is a PHP web content management system written, boasting extensions, advanced localization, scalability, and its open source nature. By making a distinction between the frontend and the backend, content visible to site visitors and administrators is innately abstracted. TYPO3 makes heavy use of extensions to supplement the core features with common, compartmentalized enhancements.

Versions

Version	EOM	PHP version	MySQL version	Release Date
6.2 LTS	2017-04	5.3.7-5.6.x	5.1.x-5.6.x	2014-03-25
7 LTS	2018-11	5.5 - 7.1	5.5+	2015-11-10
8 LTS	2020-03	7.0 - 7.1	5.5+	2017-04-04

Examples

Classic installation on a Unix server

Get the Source Package from <http://typo3.org/download/> and upload this package to your web server. Put it one level above the document root.

For this manual, we will use the .tar.gz file. Use the shell to execute the according commands:

```
/var/www/site/htdocs/$ cd ..  
/var/www/site/$ wget get.typo3.org/7.6 -O typo3_src-7.6.x.tar.gz
```

Unpack the typo3_src-7.6.x.tar.gz file on your web server:

```
/var/www/site/$ tar -xzf typo3_src-7.6.x.tar.gz
```

Create these symlinks in your document root:

```
cd htdocs  
ln -s ../typo3_src-7.6.x typo3_src  
ln -s typo3_src/typo3 typo3  
ln -s typo3_src/index.php index.php
```

TYPO3 CMS

TYPO3 is the leading enterprise content management system. It focuses on providing the features that professionals need to build and maintain large and/or complicated websites across multiple devices (mobile and desktop).

Features

- Open Source
- Multilingual
- Frequent security updates - safety first
- Manage as many websites as you want from a single installation
- Highly extendable through a large library of free third-party extensions.
- Expandable through its API based framework
- Professional support by numerous hosting companies, web agencies and freelancers

Read **Getting started with typo3** online: <https://riptutorial.com/typo3/topic/21113/getting-started-with-typo3>

Chapter 2: Add custom class for links in RTE

Introduction

Here is a quick example to see how it is possible to add custom classes for links in TYPO3 RTE.

Examples

Add custom link classes

```
RTE {
  classes {
    btn-lg {
      name = Button (large)
      requires = btn btn-default
    }

    btn-default {
      name = Button (default)
      requires = btn
    }
  }

  default {
    buttons.link.properties.class.allowedClasses := addToList(btn-lg, btn-default)
    proc.allowedClasses := addToList(btn-lg, btn-default)
  }
}
```

Read Add custom class for links in RTE online: <https://riptutorial.com/typo3/topic/9064/add-custom-class-for-links-in-rte>

Chapter 3: Basic typoscript for TYPO3 installation

Introduction

Here is the list of some basic typoscript for TYPO3 installation.

Examples

Require Config Typoscript.

```
config {
    simulateStaticDocuments = 0
    index_enable = 1
    indexexternals = 1
    linkVars = L
    sys_language_mode = content_fallback
    sys_language_overlay = hideNonTranslated
    doctype = xhtml_trans
    xhtml_cleaning = all
    doctype = html5
    xmlprologue = none
    renderCharset = utf-8
    no_cache = 0
    locale_all = de-DE
    sys_language_uid = 0
    htmlTag_langKey = de-DE
    language = de-DE
    metaCharset = utf-8
}

page.meta.description = {page:description}
page.meta.description.insertData = 1
page.meta.keywords = {page:keywords}
page.meta.keywords.insertData = 1
page.meta.viewport = width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=0;
page.meta.format-detection=telephone=no
page.meta.charset = UTF-8
page.meta.X-UA-Compatible = IE=edge
page = PAGE
page.shortcutIcon = images/favicon.ico
```

get Page content :

```
# CONTENT: Main content
lib.content.main = COA
lib.content.main {
    stdWrap.innerWrap = <!--TYPO3SEARCH_begin-->|<!--TYPO3SEARCH_end-->
    10 < styles.content.get
}
```

```

lib.content.0 < lib.content.main

# CONTENT: Sidebar
lib.content.left = COA
lib.content.left {
    stdWrap.innerWrap = <!--TYPO3SEARCH_begin-->|<!--TYPO3SEARCH_end-->
    10 < styles.content.getLeft
}
lib.content.1 < lib.content.left

# CONTENT: Sidebar
lib.content.right = COA
lib.content.right {
    stdWrap.innerWrap = <!--TYPO3SEARCH_begin-->|<!--TYPO3SEARCH_end-->
    10 < styles.content.getRight
}
lib.content.2 < lib.content.right

# CONTENT: Top Content
lib.content.top = COA
lib.content.top {
    stdWrap.innerWrap = <!--TYPO3SEARCH_begin-->|<!--TYPO3SEARCH_end-->
    10 < styles.content.getBorder
}
lib.content.3 < lib.content.top

```

Add css and js file:

```

page.includeCSS {
    bootstrap = fileadmin/css/bootstrap.min.css
    fonts = fileadmin/css/font-awesome.min.css
    owl = fileadmin/css/owl.carousel.css
    style = fileadmin/css/docs.css
}

page.includeJSFooter{
    bootstrapmin = fileadmin/js/bootstrap.min.js
    lightbox = fileadmin/js/lightbox-plus-jquery.min.js
    owl = fileadmin/js/owl.carousel.min.js
    custom = fileadmin/js/custom.js
}

page.includeJS {
    jqueryMin = EXT:website_template/Resources/Public/js/jquery-1.11.2.min.js
}

```

Remove Controller and Action name In URL

```

plugin.tx_news {
    settings {
        link {
            skipControllerAndAction = 1
        }
    }
}

[globalVar = GP:tx_news_pil|news > 0]

```

```
config.defaultGetVars {
    tx_news_pi1 {
        controller=News
        action=detail
    }
}
[global]
```

include typoscript:

```
<INCLUDE_TYPOSCRIPT: source="FILE:fileadmin/html/mainmenu_typoscript.txt">
<INCLUDE_TYPOSCRIPT: source="DIR:fileadmin/templates/" extensions="ts">
```

Read Basic typoscript for TYPO3 installation online: <https://riptutorial.com/typo3/topic/10551/basic-typoscript-for-typo3-installation>

Chapter 4: Database Query DataProcessor - Examples

Examples

How to get selected records from another table (using TCA group field)

Example of a TCA field configuration where you can select records from a table

```
'my_topfeatures' => array(
    'label' => 'Select Topfeatures',
    'config' => array(
        'type' => 'group',
        'internal_type' => 'db',
        'size' => '4',
        'allowed' => 'tx_topfeatures_items',
        'wizards' => array(
            'suggest' => array(
                'type' => 'suggest',
                'default' => array(
                    'searchWholePhrase' => 1
                ),
            ),
            'pages' => array(
                'searchCondition' => 'doktype = 1'
            )
        )
    )
)
```

Typoscript setup:

```
tt_content.yourctype {
    templateName = FE_Topfeatures
    dataProcessing {
        10 = TYPO3\CMS\Frontend\DataProcessing\DatabaseQueryProcessor
        10 {
            table = tx_topfeatures_items
            pidInList = root
            recursive = 1
            where = deleted = 0 AND hidden = 0
            where.inval = 1
            uidInList = ###selectedtopfeatures###
            orderBy = sorting
            as = tx_topfeatures_items
            markers {
                selectedtopfeatures.field = my_topfeatures
                selectedtopfeatures.commaSeparatedList = 1
            }

            dataProcessing {
                10 = TYPO3\CMS\Frontend\DataProcessing\FilesProcessor
                10 {
```

```
        references.fieldName = downloadfile
        as = downloadfiles
    }
}
}
```

Read Database Query DataProcessor - Examples online:

<https://riptutorial.com/typo3/topic/10105/database-query-dataprocessor---examples>

Chapter 5: Extbase Extension Development in TYPO3

Examples

Introduction

TYPO3 Extbase extension development method came along with the release of TYPO3 4.3.0 in November 2009. Extbase is a PHP-based framework which supports developers in creating clean and easily maintainable TYPO3 extensions. The first real presentation of Extbase happened in April 2009 on the American TYPO3 Conference in Dallas, and a month later on the TYPO3 Developer Days in Elmshorn near Hamburg. After that, a lot of positive feedback and constructive criticism emerged from the community.

Read Extbase Extension Development in TYPO3 online:

<https://riptutorial.com/typo3/topic/2138/extbase-extension-development-in-typo3>

Chapter 6: Fluid templating in TYPO3

Examples

Typoscript setup

If you want to use your own templates (be it HTML, XML, JSON or any other format) for a TYPO3 extension using fluid templating, you can add additional template paths.

An extensions template/partial/layout paths are usually set in TypoScript setup under the properties

```
plugin.tx_<extensionkey>.view.templateRootPaths
plugin.tx_<extensionkey>.view.partialRootPaths
plugin.tx_<extensionkey>.view.layoutRootPaths
```

(<extensionkey> = The extension key in lowercase without underscores. Example: power_blog => powerblog)

It is also possible to set template paths only for a specific plugin of an extension, not for the whole extension. This can be done by setting the paths

```
plugin.tx_<extensionkey>_<pluginname>.view.templateRootPaths
plugin.tx_<extensionkey>_<pluginname>.view.partialRootPaths
plugin.tx_<extensionkey>_<pluginname>.view.layoutRootPaths
```

(<pluginname > = The plugin name in lower case. Example: Pi1 => pi1 , AjaxCall => ajaxcall)

Each of these properties must be an array with numerical keys. The keys are the priority of the path: When searching for a specific template, partial or layout, fluid first looks in the path with the highest key, then in the path with the next lower key and so on, and uses the first matching template it finds.

This prioritization has the benefit that one can override only some of the templates or partials of the extensions default template, and reuse the others.

Here is an example for the extension `news`. In version 4.3.0, the static TypoScript template defines these template paths:

```
plugin.tx_news {
    view {
        templateRootPaths {
            0 = EXT:news/Resources/Private/Templates/
            1 = {$plugin.tx_news.view.templateRootPath}
        }
        partialRootPaths {
            0 = EXT:news/Resources/Private/Partials/
            1 = {$plugin.tx_news.view.partialRootPath}
        }
    }
}
```

```

        layoutRootPaths {
            0 = EXT:news/Resources/Private/Layouts/
            1 = {$plugin.tx_news.view.layoutRootPath}
        }
    }
}

```

In order to use another template, one could add this to the TypoScript setup, in addition to the static template from `news`:

```

plugin.tx_news {
    view {
        templateRootPaths {
            100 = path/to/my/own/templates/
        }
        partialRootPaths {
            100 = path/to/my/own/partials/
        }
        layoutRootPaths {
            100 = path/to/my/own/layouts/
        }
    }
}

```

Then templates would first be searched in `path/to/my/own/templates/`, and if not found there, then in the path defined in the TypoScript constant `{$plugin.tx_news.view.templateRootPath}`, and after that in the folder where the extensions default templates reside, `EXT:news/Resources/Private/Templates/`, which usually resolves to `typo3conf/ext/news/Resources/Private/Templates/`.

If you want to do the same, but only for the plugin `P11` of the extension (its only plugin as of version 4.3.0), the last example would look like this:

```

plugin.tx_news_p11 {
    view {
        templateRootPaths {
            100 = path/to/my/own/templates/
        }
        partialRootPaths {
            100 = path/to/my/own/partials/
        }
        layoutRootPaths {
            100 = path/to/my/own/layouts/
        }
    }
}

```

Example Fluid Template File

Templates/Text.html

```

<html xmlns:f="http://typo3.org/ns/TYPO3/CMS/Fluid/ViewHelpers" data-namespace-typo3-
fluid="true">
<f:layout name="Default" />

```

```
<f:section name="Main">

    <f:format.html>{data.bodytext}</f:format.html>

</f:section>
</html>
```

Layouts/Default.html

```
<html xmlns:f="http://typo3.org/ns/TYPO3/CMS/Fluid/ViewHelpers" data-namespace-typo3-
fluid="true">
<f:spaceless>
    <f:render section="Main" optional="true" />
</f:spaceless>
</html>
```

Read Fluid templating in TYPO3 online: <https://riptutorial.com/typo3/topic/2196/fluid-templating-in-typo3>

Chapter 7: Image cropping TYPO3 8.7 LTS

Examples

Usage and configuration of image cropping

PageTS Settings:

```
## Default Image cropping ##
TCEFORM.sys_file_reference.crop.config.cropVariants {
    default {
        title = Desktop
        selectedRatio = NaN
        allowedAspectRatios {
            NaN {
                title = Free
                value = 0.0
            }
            21:9 {
                title = 21:9
                value = 2.3333333
            }
            16:9 {
                title = 16:9
                value = 1.7777777
            }
            4:3 {
                title = 4:3
                value = 1.3333333
            }
            3:2 {
                title = 3:2
                value = 1.5
            }
            1:1 {
                title = 1:1
                value = 1
            }
        }
    }
}
tablet {
    title = Tablet
    selectedRatio = NaN
    allowedAspectRatios {
        NaN {
            title = Free
            value = 0.0
        }
        21:9 {
            title = 21:9
            value = 2.3333333
        }
        16:9 {
            title = 16:9
            value = 1.7777777
        }
    }
}
```

```

    4:3 {
        title = 4:3
        value = 1.3333333
    }
    3:2 {
        title = 3:2
        value = 1.5
    }
    1:1 {
        title = 1:1
        value = 1
    }
}
}
mobile {
    title = Mobile
    selectedRatio = NaN
    allowedAspectRatios {
        NaN {
            title = Free
            value = 0.0
        }
        21:9 {
            title = 21:9
            value = 2.3333333
        }
        16:9 {
            title = 16:9
            value = 1.7777777
        }
        4:3 {
            title = 4:3
            value = 1.3333333
        }
        3:2 {
            title = 3:2
            value = 1.5
        }
        1:1 {
            title = 1:1
            value = 1
        }
    }
}
}
}
## Default Image cropping - END ##

```

Fluid Template Example:

```

<f:for each="{images}" as="image">
    <picture>
        <source srcset="{f:uri.image(image: image, maxWidth: settings.maxImgWidth,
cropVariant: 'default')}}" media="(min-width: 1200px)">
        <source srcset="{f:uri.image(image: image, maxWidth: '992', cropVariant: 'default')},
{f:uri.image(image: image, maxWidth: '1984', cropVariant: 'default')} 2x" media="(min-width:
992px)">
        <source srcset="{f:uri.image(image: image, maxWidth: '768', cropVariant: 'tablet')},
{f:uri.image(image: image, maxWidth: '1536', cropVariant: 'tablet')} 2x" media="(min-width:
768px)">
        <source srcset="{f:uri.image(image: image, maxWidth: '768', cropVariant: 'mobile')},

```

```
{f:uri.image(image: image, maxWidth: '1536', cropVariant: 'mobile')} 2x" media="(max-width: 767px)">
  <!--Fallback-->
  
</picture>
</f:for>
```

Read Image cropping TYPO3 8.7 LTS online: <https://riptutorial.com/typo3/topic/9820/image-cropping-typo3-8-7-lts>

Chapter 8: Installation & Setup

Examples

Installation with composer

TYPO3 can be solely installed with the PHP dependency manager composer. Composer has to be available on the server, then a TYPO3 project can be started by using the base distribution.

```
composer create-project typo3/cms-base-distribution .
```

This will pull the TYPO3 core from the git repository, download it to `vendor/typo3/cms/` and create a folder called `web/` that will be the document root of the project.

The base distribution contains nothing more than a `composer.json` and a `.gitignore` file to get a project started. These files could also be created by hand. The `composer.json` could look like this:

```
{
  "repositories": [
    { "type": "composer", "url": "https://composer.typo3.org/" }
  ],
  "name": "typo3/cms-base-distribution",
  "description": "TYPO3 CMS Base Distribution",
  "license": "GPL-2.0+",
  "require": {
    "typo3/cms": "^7.6"
  },
  "extra": {
    "typo3/cms": {
      "cms-package-dir": "${vendor-dir}/typo3/cms",
      "web-dir": "web"
    }
  }
}
```

When this file is created by hand the command `composer update` will also fetch all files needed for a TYPO3 project.

Further information can be found here:

- composer.typo3.org
- usetypo3.com/typo3-and-composer
- [Composer in the TYPO3 wiki](#)

Read Installation & Setup online: <https://riptutorial.com/typo3/topic/3238/installation---setup>

Chapter 9: Links to official documentation Sites

Examples

Official documentation

TYPO3 has extensive documentation. This documentation is linked here, so people can find stuff that is not documented here.

Main Documentation

The documentation of TYPO3 CMS is collected at docs.typo3.org, there is a list of all [documentation for the core](#), and [documentation for extensions](#).

Prominent documentation areas are

- [TypoScript reference](#) (tsref). A detailed documentation of TypoScript.
- [TCA reference](#). A detailed documentation of the TCA.

API documentation

The [API documentation](#) for different versions of TYPO3 CMS is very useful when [contributing to the core](#) or [creating an extension](#).

Read [Links to official documentation Sites online](#): <https://riptutorial.com/typo3/topic/3237/links-to-official-documentation-sites>

Chapter 10: Realurl with TYPO3

Examples

Introduction & Setup

The extension 'realurl' provides complete transformation of URLs with GET parameter in the browser, like "index.php?id=123&type=0&L=1" into a virtual path, a so called "Speaking URL" like "home/about-us/index.html" and back again. The objective is that URLs shall be as human readable as possible.

The extension is very flexible and can provide from simple translation of page IDs to encoding of almost any possible combination of GET parameters.

To install this extension, four steps must be taken:

1. Install it in the Extension Manager
2. Configure Apache / .htaccess
3. Modify your TypoScript template records with configuration for RealURL
4. Configure the extension in typo3conf/localconf.php

Configure Apache

RealURLs work by providing 'virtual paths' to 'virtual files'. These don't actually exist on the file-system, so you must tell Apache to let a PHP-script handle the request if it can't find the file. This way, all URLs to pages (like `www.server.com/products/product1/left.html`) will be 'redirected' to `/index.php`, which will handle the translation of the URL into GET parameters. Real files (like images, the TYPO3 backend, static html-files, etc.) will still be handled by Apache itself though.

You should put the supplied sample .htaccess file (called `_.htaccess`) in the root of your TYPO3-installation.

Alternatively, you could include the following lines in your `httpd.conf`, probably in the `VirtualHost` -section. Here is an example:

```
<VirtualHost 127.0.0.1>
    DocumentRoot /var/www/typo3/dev/testsite-3/
    ServerName www.test1.intra

    RewriteEngine On
    RewriteRule ^/typo3$ - [L]
    RewriteRule ^/typo3/.*$ - [L]

    RewriteCond %{REQUEST_FILENAME} !-f
    RewriteCond %{REQUEST_FILENAME} !-d
    RewriteCond %{REQUEST_FILENAME} !-l
    RewriteRule .* /index.php
</VirtualHost>
```

If you put it into a `.htaccess` file it has to look slightly different, basically stripping the leading slashes (“/”):

```
RewriteEngine On
RewriteRule ^typo3$ - [L]
RewriteRule ^typo3/.*$ - [L]

RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
RewriteCond %{REQUEST_FILENAME} !-l
RewriteRule .* index.php
```

This will tell Apache that it should rewrite every URL that's not a filename, directory or symlink. It leaves everything starting with /typo3/ alone too.

NGINX Configuration (optional)

In case NGINX is used as http server instead of Apache, rewrite rule also needs to be applied for it and placed into `server` section of website configuration from `sites-available`:

```
location / {
    try_files $uri $uri/ /index.php$is_args$args;
}
```

TypoScript configuration

Place these three lines in the main TypoScript template record of your website:

```
config.simulateStaticDocuments = 0
config.baseURL = http://mydomain.com/
config.tx_realurl_enable = 1
```

Line 1 simply disables “simulateStaticDocuments” - “RealURL” is incompatible with simulateStaticDocuments and will simply not work if it has been enabled. This line should remind you of this fact.

Line 2 makes the frontend output a `<base>` tag in the header of the pages. This is required because relative references to images, stylesheets etc. will break when the virtual paths are used unless this has been set. Please see below for a detail discussion of why this is needed. Do not forget to write real name of your domain! And note slash in the end – it is required!

Line 3 enables the encoding of URLs as the virtual paths, the “Speaking URLs”.

If you use `config.typolinkEnableLinksAcrossDomains`, make sure you check “Is site root?” on all pages that are site roots.

And finally you have to generate a `realurl.conf.php` file. Either you can create it automatically by permitting the extension to do it or you can do it manually.

Will be covered in next topic

Courtesy :

Dmitry Dulepov

<https://docs.typo3.org/typo3cms/extensions/realurl/Realurl/Index.html>

Read Realurl with TYPO3 online: <https://riptutorial.com/typo3/topic/2199/realurl-with-typo3>

Chapter 11: TypeScript

Syntax

- #one line comment
- /* multiple line comments */
- parameter = value
- parameter.property = value2
- parameter.property > # delete property
- parameter.property2 < parameter.property # copy (deep) properties
- parameter.property := addToList(35) # Add value to comma separated list
- parameter.property\\.with\\.dots = value # Escape dots to allow properties with dots

Examples

Basic TypeScript : Hello World

```
page = PAGE
page.10 = TEXT
page.10.value = HELLO WORLD
```

Usually this TypeScript snippets are added to Web >> Template >> Info/Modify >> setup

This snippet opens a new `PAGE` object. Inside the `PAGE` object, the 10th entry is set to be a `TEXT` object. The value of this `TEXT` object is set to `HELLO WORLD`. It is convention to only use the increments of 10 to define content objects in order to allow more entries to be added at a later time.

Get date in object

Get current date and time

```
lib.date = TEXT
lib.date {
  data = date:U
  strftime = %d.%m.%Y %H:%M:%S
  wrap = Today is |
}
```

Get last login time and date from fe_users

```
lib.date = TEXT
lib.date {
  data = TSFE:fe_user|user|lastlogin
  strftime = %d.%m.%Y %H:%M:%S
  wrap = Last login is at |
}
```

Consider url to be <http://test.com/page1/?tstamp=1469683852>

```
lib.date = TEXT
lib.date {
  data = GP : tstamp
  strftime = %d.%m.%Y %H:%M:%S
  wrap = Current tstamp is |
}
```

Create link

A link to any text by using typolink object

```
lib.link = TEXT
lib.link {
  value = Here is link text

  typolink {

    #You can give page uid or any external url here
    parameter = http://www.example.com/

    #Target of link
    extTarget = _blank

    #Additional parameters bound to link
    ATagParams = class="linkclass" title="Here is a link"
  }
}
```

Following will create link with javascript popup

```
lib.link = TEXT
lib.link {
  value = Open a popup window.

  stdWrap.typolink {
    # The first parameter is the page ID of the target page,
    # second parameter is the size of the popup window.
    parameter = 10 500x400

    # The title attribute of the link.
    title = Click here to open a popup window.

    # The parameters of the popup window.
    JSwindow_params = menubar=0, scrollbars=0, toolbar=0, resizable=1
  }
}
```

Image and Image Resource

Get image from file

```
lib.myImage = IMAGE
```

```
lib.myImage.file = fileadmin/My-Image.png
lib.myImage.file.width = 100
lib.myImage.file.height = 100
```

Get image from page properties

```
lib.pageImage = IMAGE
lib.pageImage {
    file.import.data = levelmedia: -1, "slide"
    file.import = uploads/media/
    file.import.listNum = 0
    file.import.override.field = media

    border = 0
    altText = xy
    titleText = xy
}
```

get images from page resources (FAL)

```
lib.pageResources = FILES
lib.pageResources {
    references {
        table = pages
        uid.data = uid
        fieldName = media
    }
    renderObj = IMAGE
    renderObj {
        file {
            import.data = file:current:uid
            treatIdAsReference = 1
            width = 150c
            height = 150c
        }
        altText.data = file:current:alternative
        titleText.data = file:current:title
    }
    maxItems = 3
}
```

Get image resource

```
lib.myImage = IMAGE
lib.myImage.file = fileadmin/My-Image.png
lib.myImage.file.width = 100
lib.myImage.file.height = 100
lib.myImage.stdWrap.wrap (
    <div style="background-image:url(|); width:100px; height:100px;">Headline</div>
)
```

Get images from content elements

```
lib.ceImages = FILES
lib.ceImages {
    stdWrap.wrap = <div class="pic">|</div>
```

```

references {
    table = tt_content
    // current CE (needs context)
    #uid.data = uid
    // for CSC
    fieldName = image
    // for FSC
    fieldname = assets
}
renderObj = IMAGE
renderObj {
    file {
        import.data = file:current:uid
        treatIdAsReference = 1
        width = 150c
        height = 150c
    }
    altText.data = file:current:alternative
    titleText.data = file:current:title
    stdWrap.typolink.parameter.data = file:current:link
}
maxItems = 5
}

```

Get current page title

```

lib.pagetitle = TEXT
lib.pagetitle.data = page : title

```

Get Current URL

```

lib.currentURL= TEXT
lib.currentURL.data = getIndpEnv:TYPO3_REQUEST_URL

```

Read TypoScript online: <https://riptutorial.com/typo3/topic/2127/typoscript>

Credits

S. No	Chapters	Contributors
1	Getting started with typo3	4444 , Anu Bhuvanendran Nair , bschauer , Community , Daniel , Isak Combrinck
2	Add custom class for links in RTE	StatiX
3	Basic typoscript for TYPO3 installation	TYPO3 Developer
4	Database Query DataProcessor - Examples	bschauer
5	Extbase Extension Development in TYPO3	Ricky Mathew , StephenKing
6	Fluid templating in TYPO3	András Ottó , Aswathy S , bschauer , Daniel , Jost , Ricky Mathew , StephenKing
7	Image cropping TYPO3 8.7 LTS	bschauer , Mateng
8	Installation & Setup	Community , Daniel
9	Links to official documentation Sites	Bernd Wilke πφ , Daniel , Jost
10	Realurl with TYPO3	Anu Bhuvanendran Nair , Dirk D. , Viktor Livakivskyi
11	TypoScript	Anu Bhuvanendran Nair , Bernd Wilke πφ , Daniel , Grigori Prokhorov , Mihir Bhatt , pgampe , ravisachaniya , TYPO3 Developer , undko