

# The udiss bundle

TUGINDIA

Version 0.1—October 1, 2024

 <https://ctan.org/pkg/udiss>

 <https://puszcza.gnu.org.ua/bugs/?group=udiss>

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Documentation</b>	<b>2</b>
2.1	User manual . . . . .	2
2.2	Multilingual typesetting . . . . .	7
2.3	Developer’s manual . . . . .	8
2.3.1	Fonts . . . . .	8
2.3.2	Title-page customisation . . . . .	9
2.4	Contributing to our repository . . . . .	10
<b>3</b>	<b>Requests for new templates</b>	<b>10</b>
<b>4</b>	<b>Implementation</b>	<b>11</b>
	<b>Index</b>	<b>58</b>
	<b>GNU Free Documentation License</b>	<b>62</b>

---

The udiss bundle

Copyright © 2024 TUGINDIA

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled ‘GNU Free Documentation License’.

## I Introduction

The `udiss` bundle is a  $\LaTeX$ -class-file developed to assist students in typesetting their university *dissertations*. It is a collection of multiple support files. This is the first experimental version. We need feedback and comments from the community to make it better. Currently, nothing should be considered ‘stable’. Soon after getting comments and reviews from the community, we will move towards a stable version.

Universities often have strict requirements regarding the formatting of the dissertations/theses submitted to them. This bundle pre-supplies a generic style (university-agnostic) for creating dissertations. It can be loaded simply with:

```
\documentclass{udiss}
```

But, apart from the generic style, it also supplies university or department specific files for some selected institutions. Each specialised class file comes with an identifier. E.g., if you want to use the template for the Kerala university, you may use the `ukerala` tag and load its template like in the following:

```
\documentclass{udiss}
\dissertationstyle{ukerala}
```

We are willing to expand. If you want your institution to be supported by us, please have a look at the instructions given in section 3. If you want to develop your own style, please have a look at the developer’s instruction given in section 2.3. If you want to get your class published under this bundle, please have a look at the contribution guidelines given in section 2.4.

This document explains the user interface as well as the implementation of the class. It has three major parts. The first one contains the documentation of the class (section 2), the second one contains the developer’s manual (section 2.3) and the third one contains the implementation of it (section 4).

Before getting started with the documentation, just a brief note about our approach. As of version 0.1, we are planning to have a radically all-modern and backwards-blind approach. This is a Unicode-only package, meaning, it won’t work with the classic  $\text{PDF}\LaTeX$ . We load the modern, well maintained and inclusive implementation of the Latin Modern family, i.e., the New Computer Modern family ([ctan.org/pkg/newcm](https://ctan.org/pkg/newcm)). For using/testing the class, please use the latest  $\text{T}\text{E}\text{X}$  Live (all packages updated, as of 2024-09-27). Both  $\text{X}\text{E}\text{L}\text{A}\text{T}\text{E}\text{X}$  and  $\text{L}\text{u}\text{a}\text{L}\text{A}\text{T}\text{E}\text{X}$  supported, but the latter is recommended as it identifies scripts without markup.

## 2 Documentation

This part of the document describes everything that a normal user of this class should know. It enlists all the features of the class.

### 2.1 User manual

Since this is a class that typesets dissertations, it requires some basic information which is provided using the following macros.

We provide the usual  $\LaTeX$ -like macros for the convenience, but these macros have *key-value*-equivalents as well.

<hr/>	<code>\title</code>	<code>\title{<i>&lt;the-title-of-the-dissertation&gt;</i>}</code>	▲
	<code>\author</code>	<code>\author{<i>&lt;name-of-the-author&gt;</i>}</code>	▲
	<code>\pronouns</code>	<code>\pronouns{<i>&lt;the-pronouns-of-the-author&gt;</i>}</code>	

Like in most standard L<sup>A</sup>T<sub>E</sub>X classes, `\title` command saves the title of the dissertation from its argument. This is one of the commands which we mark as *strict*. In the manual, all such commands are marked with a ▲ symbol at the right end. If such commands (or their parametric equivalents as shown in the following paragraph) are not provided, the compilation is interrupted with an error. The `\author` command doesn't return any error if the class option `blind` is loaded. Since the `blind` option is used for anonymous submissions, the author information is irrelevant anyways and hence, is ignored.

These macros are available as class options too. One just has to use `title={<title>}` in the optional argument of the `\documentclass` command. Similarly, `author` and `pronouns` also are available.

<hr/>	<code>\subtitle</code>	<code>\subtitle{<i>&lt;the-subtitle-of-the-dissertation&gt;</i>}</code>
	<code>\shorttitle</code>	<code>\shorttitle{<i>&lt;a-short-title-for-headers&gt;</i>}</code>

Dissertations often have a subtitle too, but it is not an absolute requirement, so optionally, one may want to use `\subtitle` command. Sometimes the title and the subtitle are too long for the running header. One may provide a short title with `\shorttitle` for this purpose. Depending on the selected template and options, it will be printed.

These commands are also available as class options `subtitle` and `shorttitle`.

<hr/>	<code>\stream</code>	<code>\stream{<i>&lt;the-name-of-the-stream&gt;</i>}</code>	▲
	<code>\discipline</code>	<code>\discipline{<i>&lt;the-name-of-the-discipline&gt;</i>}</code>	▲
	<code>\degree</code>	<code>\degree{<i>&lt;the-name-of-the-degree&gt;</i>}</code>	▲

These details are used at several places in the class. E.g., the declaration, the 'fulfilment' statement printed on the first page. Like most other macros, these too have straight forward *key-value*-equivalents `stream`, `discipline` and `degree`.

<hr/>	<code>\supervisor</code>	<code>\supervisor{<i>&lt;name-of-the-supervisor&gt;</i>}</code>	▲
	<code>\university</code>	<code>\university{<i>&lt;name-of-the-university&gt;</i>}</code>	▲
	<code>\department</code>	<code>\department{<i>&lt;name-of-the-department&gt;</i>}</code>	▲

These commands are not seen in other standard L<sup>A</sup>T<sub>E</sub>X-classes, but `udiss` requires them obligatorily (only except in the `blind` mode).

`supervisor`, `university` and `department` are the *key-value*-equivalents.

<hr/>	<code>\logo</code>	<code>\logo{<i>&lt;file-name&gt;</i>}</code>
	<code>\logo*</code>	<code>\logo*{<i>&lt;TeX-code&gt;</i>}</code>

Some departments mandate printing the university logos in the dissertation. If this command is used, the logo is printed. The location of the logo can be customised as per the requirement. The class provides options for that.

If you are mad about T<sub>E</sub>X and have a code for printing the logo of your institute, use this command with a star and in its argument, paste your code to get it printed at an appropriate place.

We have `logo` and `logo*` as the *key-value*-equivalents.

<code>language</code>	<code>language={\langle babel-language-names \rangle}</code>	
<code>ldfbabel</code>	<code>ldfbabel={\langle babel-ldf-language-names \rangle}</code>	
<code>nofontwarning</code>	<code>nofontwarning={\langle truth-value \rangle}</code>	true   false

The `babel` package has a complex mechanism for loading languages. The complex design has its own advantages, especially with regards to backwards compatibility. For detailed explanation, it's best to go through the user manual of `babel`, or its official online documentation, especially the web-page at: <https://latex3.github.io/babel/guides/which-method-for-which-language.html>.

To help those who are completely at sea, we briefly will have a look at the two mechanisms `babel` provides. One is the original method of loading the language-name(s) as class option(s), so something like:

```
\usepackage[italian,hindi,english]{babel}
```

The last one from the list is understood as the ‘main’ language of the document. These languages are defined using the so-called `.ldf`-files. A lot of support for some languages is available using this mechanism. For some European languages, e.g., French, the difference between the two mechanisms is too sharp and it has been found that a lot of native speakers still prefer the old, stable way of using `.ldf`-files. If you are one of them and you are used to loading language-names as options to package `babel`, you should use the `ldfbabel` option of `udiss`. But we advise against it.

The second approach provided by `babel` for loading languages is as follows:

```
\babelprovide{\langle language-name \rangle}
```

This method follows and uses the Unicode Common Locale Data Repository (CLDR) for creating the resources. This data is saved in `.ini` files. This is a more modern approach, in our opinion. But it may cost us a little bit of inconvenience since it doesn't support the common short-hands which are provided by the `.ldf` files. But short-hands make the input more and more  $\text{\TeX}$ -dependent and thus less productive, so we don't endorse it. That is why the more intuitive name for language-options in this class is for the `.ini`-based approach. We load the `english` language by default if no other is loaded. For this too, we go with the `.ini` file developed for English.

The syntax of both these options is exactly like the one provided by `babel`. One may provide a comma-separated list and the last one of them will be set as the main language of the document. Have a look at section 2.3 for more details regarding this.

We also provide some auxiliary  $\text{\TeX}$  files with the class for setting the font-families as well as the language-specifics as required by `babel`. If a file like that it is not found, we issue a warning. If the users want that warning to be silent, they can use the option `nofontwarning`.

<code>license</code>	<code>license={\langle all-rights-reserved/CC-BY-SA/GFDL \rangle}</code>	<code>all-rights-reserved</code>
<code>copyright-yrs</code>	<code>copyright-yrs={\langle copyrightable-years \rangle}</code>	<code>\the\year</code>
<code>localcopyright</code>	<code>localcopyright={\langle the-word-for-copyright \rangle}</code>	<code>Copyright</code>

This is an option to set a license for the dissertation. We encourage users to use free culture licenses. Currently we support two popular licenses, i.e., ‘CC-BY-SA 4.0+ International’ and ‘GFDL v1.3+’. We don’t set any license by default as we believe making a work ‘free’ (as in freedom) should be the decision of the author and hence ‘all rights are reserved’ by default. If you are not using the `blind` package option, your output will have a copyright notice with your (author’s) name and the current year as the copyrightable year in the metadata of the PDF. If you are wondering why, the `blind` option is for preserving the anonymity of the author and the copyright-notices require a copyright-holder by default, thus with `blind` option, we have disabled choosing licenses. If you want to change the copyrightable year(s), you may use the option `copyright-yrs`. Since the package is supposed to support multilingual typesetting, we have provided an option to change the word ‘Copyright’ to the one in the main language of your document.

<code>placeholders</code>	<code>placeholders</code>
---------------------------	---------------------------

If you want to see how the things so far discussed come together, you may just load this option. It places placeholders for all the options discussed for creating a sample document.

<code>lot</code>	<code>lot={\langle truth-value \rangle}</code>	<code>true   false</code>
<code>lof</code>	<code>lof={\langle truth-value \rangle}</code>	<code>true   false</code>
<code>declaration</code>	<code>declaration={\langle truth-value \rangle}</code>	<code>true   false</code>
<code>declarationtxt</code>	By default, the class only prints the table of contents. If the author requires the list of figures ( <code>lof</code> ) or tables ( <code>lot</code> ), they may use the respective option. These are boolean keys which take either <code>true</code> or <code>false</code> as their values. Not passing any truth-value to a <code>key</code> is equivalent to using <code>key=true</code> . Similarly, many universities require a declaration from the candidate regarding the originality of the work. We have provided a sample draft of the declaration in a renewable macro, but it is not enabled by default. After getting comments from the community, we will decide what should be done with it. Note that the default value for all these keys is <code>false</code> <sup>1</sup> . Declaration text can be edited using <code>declarationtxt</code> option. The local word for declaration is stored in <code>decllocal</code> (defaulting to ‘Declaration’).	
<code>decllocal</code>		

<code>blind</code>	<code>blind={\langle truth-value \rangle}</code>	<code>true   false</code>
--------------------	--	---------------------------

At times, anonymous submissions are required where the name of the author and supervisor is hidden.

<code>framed-title</code>	<code>framed-title={\langle truth-value \rangle}</code>	<code>true   false</code>
---------------------------	---	---------------------------

By default the title-page is very minimal. If a user wants to have a frame around the title-page, they may use this option.

<sup>1</sup>Throughout the documentation, at the right corner, we enlist the possible values to a key and the default one is marked in red.

<code>print</code>	<code>print={\langle truth-value \rangle}</code>	<code>true   false</code>
<code>norefcolors</code>	<code>norefcolors={\langle truth-value \rangle}</code>	<code>true   false</code>
<code>colorprint</code>	<code>udisslinkclr={\langle link-colour \rangle}</code>	<code>udisslink</code>
<code>udisslinkclr</code>	<code>udissurlclr={\langle url-colour \rangle}</code>	<code>udissurl</code>
<code>udissurlclr</code>	<code>udissciteclr={\langle cite-colour \rangle}</code>	<code>udisscite</code>

Many departments require the submissions to be made in the print form as well. For that, `print` is a boolean option. It creates outputs suitable for the print submissions. The class, by default, does not use `print` option. The default output of the class is suitable for digital submissions. The following settings are always turned on by the class in normal circumstances:

- The `hyperref` package is loaded.
- The `xcolor` package is loaded. Some colours for the internal cross-references are pre-selected.

If you don't want colours for the hyperlinks in your digital document, you may use the `norefcolors` option. If you don't like the pre-defined colour-scheme given by us for the digital documents, you may use the class options specifically developed for changing them. They are `udisslinkclr`, `udissurlclr` and `udissciteclr`. The arguments of these options are either the pre-defined colours by package `xcolor` or user-defined colours with the `\colorlet` or `\definecolor` commands. The default colour-scheme used by us is as seen in the macro explanation above.

More importantly though, if a PDF with colours is given for printing and the printer is using the black and white mode, then the coloured words look faded in the print out. To avoid this, we load the `xcolor` package only when the `print` option is *not* loaded (basically always, by default). For performance reasons, we don't load the `hyperref` package when `print` is loaded. Hyper-references make sense only in a digital PDF, so to optimise PDF compilation, in `print` mode, we don't load `hyperref`.

Note that if you are using the `license` option with `CC-BY-SA`, the `doclicense` package `\Required` for printing the license automatically loads the `hyperref` package. We don't have any control over it and hence, you will see the hyperlink boxes around the `CC`-text. You can safely ignore them as the text is still in black colour and your objective is anyways just to *print* the output.

<code>oldstylenumoff</code>	<code>oldstylenumoff={\langle truth-value \rangle}</code>	true   false
<code>olddone</code>	<code>olddone={\langle truth-value \rangle}</code>	true   false

In text, the old style numbers look better. The numbers in the current document are old style. Lining numbers, on the other hand, don't have ascenders and descenders.

If you prefer the lining numbers, you may turn this option on and all the fonts will start showing lining numbers. Note that printing of old style numbers also depends upon whether the font has old style numbers or not. The relevant settings are added to the font automatically, but while selecting the font, first make sure whether the old style table is present or not.

Apart from that, the New Computer Modern font family provides an old-style shape for the number 1 (this exact shape!), but this shape is provided as a character variant and character variants cannot be loaded globally, since various fonts may have various variants at that slot, hence if someone wants this shape, they will have to use the `olddone` option. Without this option the number looks like as seen on the first line and that's the default style of `udiss`.

0123456789	Old style with default 1
0I23456789	Old style with the old 1
0123456789	Lining

<code>fulfilment</code>	<code>fulfilment={\langle the-titlepage-sentence-about-course-requirement \rangle}</code>
-------------------------	---

The title-page prints a sentence which says:

A dissertation submitted in the partial fulfilment of the requirements for the `\udiss@degree` program in `\udiss@discipline`.

In order to change it, one may use this package option. If you want to use the internal macros for univesity and department names, don't forget to use `\makeatletter` and `\makeatother`.

As described in the introduction (section 1), the users can choose to load the generic style or a specialised style. For that, we have the following macro:

<code>\dissertationstyle</code>	<code>\dissertationstyle{\langle style-for-dissertation \rangle}</code>
---------------------------------	---

With this command, you may select the institute or the style. It will set certain defaults.

We provide limited support as of v0.1 (only for university of Kerala). We wish to expand more. Send us support requests for adding your institute to the list.

## 2.2 Multilingual typesetting

The `udiss` class aims at supporting multilingual typesetting by default. We load the `babel` package for linguistic support, `fontspec` and `unicode-math` package for loading fonts. It's the best to not load them again. For loading languages, it's best to use the package internal mechanism. If you find something missing in the mechanism, you may load `babel` or `fontspec` commands directly. Please consider reporting a missing feature so that we can implement it in the future versions.

We will see two examples of loading languages.



## 2.3 Developer's manual

*Only if you sure about what you are doing*

As the section title and the double bend suggests, this area is for advanced customisation. Only enter it if you have sufficient confidence to avoid errors.

### 2.3.1 Fonts

We are a Unicode-only class, so we need to take care of three major font families, i.e., Roman, **Typewriter** and **Sans** (stylistation intentional, for demonstration). The `fontspec` package takes care of Unicode fonts in  $\LaTeX$ , but we, additionally, load package `unicode-math` too, to set math-fonts. Even though, this much is practically enough for font-selection, we go with the `babel`'s mechanism of loading language-specific fonts, so that the fonts don't apply globally. Our class already takes care of most of the requirements regarding font, but if you want to tinker with the defaults, you are advised to read this section. The following parameters are for you.

<hr/> <code>explicittext</code>	<code>explicittext={\langle truth-value \rangle}</code>	<code>true   false</code>
<code>ignorefsoff</code>	<code>ignorefsoff={\langle truth-value \rangle}</code>	<code>true   false</code>

Without this option, one cannot use the `Extension` parameter of package `fontspec/unicode-math`. We have some options later which set the extension for `rm`, `sf` and `tt` families. They are also useless if this key is not used. A word of caution: loading explicit extensions for fonts is restrictive. Suppose you want to use `KpRoman-Regular` as your normal font, but, say, `FreeSerifBold` as your bold font (please don't, this is solely for exemplification :P), loading an explicit extension will cause errors as the former is an OpenType font and the latter is a TrueType one. So if you are 200% sure that all the fonts you need use one particular extension *only*, then use this option. Otherwise, you are better off without it.

The  $\TeX$ -tree contains many files having the `.fontspec` extension. These files contain predefined settings for fonts which the users may not always want. Thus, we ignore these files by default. If you are very sure that you want to load `.fontspec` files, you may turn our conditional off and the `.fontspec`-files, will be loaded correctly.

<hr/> <code>rmfeatures</code>	<code>rmfeatures={\langle font-features-for-serif-fonts \rangle}</code>
<code>sffeatures</code>	<code>sffeatures={\langle font-features-for-sans-fonts \rangle}</code>
<code>ttfeatures</code>	<code>ttfeatures={\langle font-features-for-mono-fonts \rangle}</code>
<code>mathfeatures</code>	<code>mathfeatures={\langle font-features-for-math-fonts \rangle}</code>

We have `udiss`-options for most commonly used optional parameters of fonts, still there are a lot of them available which we didn't consider important for the purpose of this class, e.g., `Scale`, `Color`. If you want to use these features with your fonts, you may use these four options as per your need. The value of this key is directly injected in the font-loading commands.

<hr/> <code>rmfntext</code>	<code>rmfntext={\langle font-extension-for-serif-fonts \rangle}</code>	<code>ttf   otf</code>
<code>sffntext</code>	<code>sffntext={\langle font-extension-for-sans-fonts \rangle}</code>	<code>ttf   otf</code>
<code>ttfntext</code>	<code>ttfntext={\langle font-extension-for-mono-fonts \rangle}</code>	<code>ttf   otf</code>
<code>mathfntext</code>	<code>mathfntext={\langle font-extension-for-math-fonts \rangle}</code>	<code>ttf   otf</code>

With these four options, one may choose either `ttf` or `otf` extension for a particular font family. As mentioned before, it applies to all the fonts loaded in that family. Note that these options are only used when the `explicittext` option is active. This is why there is no default value set. By default this explicit extension method itself is not used.



<b>rmfont</b>	<b>sffont</b>	<b>ttfont</b>
NewCMR10-Book	NewCMSans10-Book	NewCMMono10-Book
<b>rmbffont</b>	<b>sfbffont</b>	<b>ttbffont</b>
NewCMR10-Bold	NewCMSans10-Bold	NewCMMono10-Bold
<b>rmitfont</b>	<b>sfitfont</b>	<b>ttitfont</b>
NewCMR10-BookItalic	NewCMSans10-BookItalic	NewCMMono10-BookItalic
<b>rmbfitfont</b>	<b>sfbitfont</b>	<b>ttbitfont</b>
NewCMR10-BoldItalic	NewCMSans10-BoldOblique	NewCMMono10-BoldOblique
<b>rmslfont</b>	<b>sfslfont</b>	<b>ttslfont</b>
NewCMR10-Book	NewCMSans10-BookOblique	NewCMMono10-Book
<b>rmbfslfont</b>	<b>sfbfslfont</b>	<b>ttbfslfont</b>
NewCMR10-Bold	NewCMSans10-BoldOblique	NewCMMono10-BoldOblique
<b>mathfont</b>		
NewCMMath-Book		
<b>mathbffont</b>		
NewCMMath-Bold		

Table 1: Default fonts for English language (all ofts)

If you want to know the default fonts we load always with `udiss`, you may refer to table 1.

### 2.3.2 Title-page customisation

The title-page prints several fields and very often authors are very picky about it. Most of the things we offer in it are customisable. One may change the following things for all the major fields on the title-page.

1. The font
2. The font-size
3. The font-color
4. The font-shape (e.g., `\itshape`, `\scshape` etc.)
5. Additional features to the font loaded

All the parameters (along with their default values) are listed in table 2. The table uses ‘-’ when there is no change in the default, only the fields having a special value are marked.

<b>logowidth</b>	<code>logowidth={\langle logo-width-proportion \rangle}</code>	0.3
<b>logoheight</b>	<code>logoheight={\langle logo-height-proportion \rangle}</code>	0.075

By default the width of the logo is  $0.3 \times \text{\linewidth}$  and  $0.075 \times \text{\textheight}$ . If you want to change these default numbers, you may use these parameters.

<i>Title</i>	- titlefont	<code>\bfseries</code> titleshape	huge titlesize	black titlecolor	- titlefeat
<i>Subtitle</i>	- subtitlefont	<code>\normalfont</code> subtitleshape	large subtitlesize	black subtitlecolor	- subtitlefeat
<i>Author</i>	- authorfont	<code>\normalfont</code> authorshape	large authorsize	black authorcolor	- authorfeat
<i>University</i>	- unifont	<code>\normalfont</code> unishape	<code>\normalsize</code> unisize	black unicolor	- unifeat
<i>Department</i>	- deptfont	<code>\normalfont</code> deptshape	small deptsize	black deptcolor	- deptfeat
<i>Fulfilment</i>	- fffont	<code>\normalfont</code> ffshape	small ffsize	black ffcolor	- fffeat

Table 2: Customising title-page



## 2.4 Contributing to the bundle

*Only if you are ready to bear with us :P*

We welcome contributions to our code. Our code is hosted at Puszczka, a free-software-based web-space. But for consistency, we request our supporters to write the code using 60 characters per line limit. Try to adhere to the coding conventions. If there is a slight violation here and there in your patches, please don't mind us editing your code first and then merging it in the main branch.

We release our code with GPLv3+ and the documentation with GFDLv1.3+. Your patches also will have to comply with the conditions of these licenses.

If you wish to document your patches, please avoid gendered language. Use *they/them* pronouns when referring to users.

Apart from university-styles, we also need support for language and fonts. If you know free/libre Unicode fonts for your language, please release them on CTAN or let us know about them, so that we will release using our templates. Provide appropriate translations for the fields provided in class.

The class comes with a test-suite. You are advised to run `l3build check` command before pushing a patch. This way we can be sure that the patch doesn't break any of the already existing features. If you think something was buggy in the original design and you got a fix for that, kindly run `l3build save` on the concerned test file and push the changed `.lvt` and `.tlgs` also.

## 3 Requests for new templates

We are open to your support-requests too. Just let us know the norms and conventions of your institution. Let us know which fields you would like to have pre-filled in the style file. If you require the logo to be printed on the title-page, send it to us in a decent resolution. We will include it in our bundle.

## 4 Implementation

In this section, we explain how this program was developed. We discuss the tools used by us for developing the features offered in this class. The features can be seen in the left margin hyperlinked to their respective place in the documentation section section 2.

We first declare the tag used by the `.ins` file for stripping the source-code of the generic class. We will have several tags for generating the other class-files, but for the convenience of maintenance, we will have all the code in this central `.dtx` file.

```
1 (*class)
```

First we declare the class with some elementary information.

```
2 \ProvidesClass{udiss}[2024/10/01 v0.1
```

```
3 A LaTeX bundle for typesetting dissertations (TUGIndia)]
```

We develop the options for the class using `\ekvdefinekeys` command. This command is provided by the `expkvDEF` package, whereas for creating class/package options, we use the `expkvOPT` package. For performance reasons, we don't use the entire `expkvBUNDLE`.

```
4 \RequirePackage{expkv-def,expkv-opt}
```

`\pronouns` Mostly we have implemented the macros in the same order as they are documented, `pronouns` but the exception is of the first two, i.e., `\title` and `\author`. These commands are developed by the standard classes too and hence, after loading the class (memoir in our case) their definitions are rewritten. Thus these two macros are developed a little late in the code. Rest is still in line with the documentation. The method of saving the values of the macros is an indirect one. We use internal macros (e.g., `\udiss@title`) for saving them.

We start with the `pronouns` field. Since it is optional, we don't need any errors here. At each step, we set some if-conditionals. All of them follow the standard convention of prefixing the class/package name and the `@` symbol for internalising. Have a look at the following two conditionals which we use immediately while defining the `pronouns` field:

```
5 \newif\ifudiss@pronouns
```

```
6 \newif\ifudiss@blind
```

We use the `xparse`-syntax for developing user-side macros. First we check if the class option `blind` is loaded. If it is so, there is no point in saving `pronouns`, hence it is skipped. The following code does that:

```
7 \NewDocumentCommand{ \pronouns }{ m }{%
```

```
8 \ifudiss@blind
```

```
9 \else
```

```
10 \def\udiss@pronouns{#1}%
```

```
11 \fi
```

```
12 }
```

As mentioned before, we use `expkvDEF` and `expkvOPT` packages for developing the *key-values*. The former provides a command `\ekvdefinekeys` that defines all the *keys*. We could have called it just once in the code and developed all the *keys* at one go, but in order to keep the implementation in line with the documentation and for a generally better readable code, we segment it in several pieces and develop the respective options with different calls of `\ekvdefinekeys`. Note that the set used for all of them is still the same, i.e., `udiss`. Storing a value passed to a key in a specific macro for later use is quite straight-forward in `expkvDEF`, but since we need to set some macros conditionally, we

will use the code *type* of `expkv`. The first non-space-token followed by the invocation of this type is the name of the *key*. Its argument is an arbitrary T<sub>E</sub>X-code where we can use the usual T<sub>E</sub>X-argument-call, i.e., #1.

```

13 \ekvdefinekeys{udiss}{%
14   code   pronouns      = {%
15     \ifudiss@blind
16     \else
17       \def\udiss@pronouns{(#1)}%
18     \fi
19     \udiss@pronounstrue
20   }%
21 }

```

(End of definition for `\pronouns` and `pronouns`. These functions are documented on page 3.)

The `\ifudiss@blind` conditional was developed in order to conditionally save (and print) some information which identifies the author. If a user loads the option `blind`, they don't need that information to be printed and since it is not supposed to be printed, why to even save it? Thus we save it only when needed.

**title** Since we don't allow blank arguments for *strict* commands/parameters, we check their existence there and there and generate an error if found blank. Since we need to error twice (once with the macro, once with the parameter, we save the code for generating the error in a handy internal macro. Let's have a look at one developed for the **title** field.

```

22 \newif\ifudiss@title
23 \def\udiss@blank@title@err{%
24   \ClassError{udiss}{%
25     The `title' field cannot be blank%
26   }{%
27     You have provided an empty argument to the `title'
28     field. This is not\MessageBreak permitted by the class.%
29   }%
30 }

```

We use the same method for all the other *strict* commands too.

```

31 \newif\ifudiss@author
32 \def\udiss@blank@author@err{%
33   \ClassError{udiss}{%
34     The `author' field cannot be blank%
35   }{%
36     You have provided an empty argument to the `author'
37     field. This is not\MessageBreak permitted by the class.%
38   }%
39 }

```

Now we start developing the code for the **title** and **author** parameters. In order to check whether a macro is empty or not, we need to use the `\ifx\macro1\macro2` construct. But since we have an argument and not a ready macro for comparison, we first we develop an internal macro with the argument as its value.

```

40 \ekvdefinekeys{udiss}{%
41   code   title          = {%
42     \def\udiss@tmp@title{#1}%

```

The true branch of this conditional, i.e., everything till either `\else` or `\fi` is found, defines the actual macro to be used while typesetting, i.e., `\udiss@title`, but since the user has provided a blank macro, we give a cryptic value to the macro, e.g., `-NoValidTitle-`. Note that this gets printed in the document even though the user gets an error. Since we use this regularly, a user may search for `-NoValid` in the output to find out all the values they missed providing.

```

43     \ifx\udiss@tmp@title\empty
44         \def\udiss@title{-NoValidTitle-}%

```

After this, we print the error for the blank argument. In the `\else` branch (which means the user did *not* pass a blank argument, but provided a legit value), we define the command with the actual user-given-value. This is the most ideal condition.

```

45     \udiss@blank@title@err
46     \else
47         \def\udiss@title{#1}%
48     \fi
49 },%

```

We repeat the same pattern for the *strict* commands now. Here is the code for `author`.

```

50     code    author        = {%
51         \def\udiss@tmp@author{#1}%
52         \ifx\udiss@tmp@author\empty
53             \def\udiss@author{-NoValidAuthor-}%
54             \udiss@blank@author@err
55         \else
56             \def\udiss@author{#1}%
57         \fi
58     }%
59 }

```

(End of definition for `title` and `author`. These functions are documented on page 3.)

`subtitle` Here is the code for the `subtitle` and `shorttitle` fields. As they aren't *strict*, no errors  
`\subtitle` needed for them. We need a conditional though. We set it to true so that it is understood  
`shorttitle` that this field was not forgotten by the author. Only if their value is true, we print them  
`\shorttitle` later.

```

60     \newif\ifudiss@subtitle
61     \newif\ifudiss@shorttitle
62     \ekvdefinekeys{udiss}{%
63         code    subtitle    = {%
64             \IfBlankTF{#1}{%
65                 \def\udiss@subtitle{-NoValidSubtitle-}%
66                 \udiss@subtitltrue
67             }{%
68                 \def\udiss@subtitle{#1}%
69                 \udiss@subtitltrue
70             }%
71     },%
72     code    shorttitle    = {%
73         \IfBlankTF{#1}{%
74             \def\udiss@shorttitle{-NoValidSubtitle-}%
75             \udiss@shorttitltrue

```

```

76     }{%
77     \def\udiss@shorttitle{#1}%
78     \udiss@shorttitletrue
79     }%
80 }%
81 }
82 \NewDocumentCommand{ \subtitle }{ m }{%
83 \def\udiss@subtitle{#1}%
84 \udiss@subtitletrue
85 }
86 \NewDocumentCommand{ \shorttitle }{ m }{%
87 \def\udiss@shorttitle{#1}%
88 \udiss@shorttitletrue
89 }

```

(End of definition for `subtitle` and others. These functions are documented on page 3.)

```

stream Again some compulsory fields, so we develop the errors, then the keys and then the
\stream macros.
discipline
\discipline
degree
\degree
90 \newif\ifudiss@stream
91 \def\udiss@blank@stream@err{%
92 \ClassError{udiss}{%
93 The `stream' field cannot be blank%
94 }{%
95 You have provided an empty argument to the `stream'
96 field. This is not\MessageBreak permitted by the class.%
97 }%
98 }
99 \newif\ifudiss@discipline
100 \def\udiss@blank@discipline@err{%
101 \ClassError{udiss}{%
102 The `discipline' field cannot be blank%
103 }{%
104 You have provided an empty argument to the `discipline'
105 field. This is not\MessageBreak permitted by the class.%
106 }%
107 }
108 \newif\ifudiss@degree
109 \def\udiss@blank@degree@err{%
110 \ClassError{udiss}{%
111 The `degree' field cannot be blank%
112 }{%
113 You have provided an empty argument to the `degree'
114 field. This is not\MessageBreak permitted by the class.%
115 }%
116 }
117 \ekvdefinekeys{udiss}{%
118 code stream = {%
119 \def\udiss@tmp@stream{#1}%
120 \ifx\udiss@tmp@stream\empty
121 \def\udiss@stream{-NoValidStream-}%
122 \udiss@blank@stream@err
123 \else
124 \def\udiss@stream{#1}%

```

```

125     \fi
126   },%
127   code    discipline    = {%
128     \def\udiss@tmp@discipline{#1}%
129     \ifx\udiss@tmp@discipline\empty
130       \def\udiss@discipline{-NoValidDiscipline-}%
131       \udiss@blank@discipline@err
132     \else
133       \def\udiss@discipline{#1}%
134     \fi
135   },%
136   code    degree        = {%
137     \def\udiss@tmp@degree{#1}%
138     \ifx\udiss@tmp@degree\empty
139       \def\udiss@degree{-NoValidDegree-}%
140       \udiss@blank@degree@err
141     \else
142       \def\udiss@degree{#1}%
143     \fi
144   }%
145 }
146 \NewDocumentCommand{ \stream }{ m }{%
147   \IfBlankTF{#1}{%
148     \udiss@blank@stream@err
149     \def\udiss@stream{-NoValidStream-}%
150   }{%
151     \def\udiss@stream{#1}%
152   }%
153   \IfBlankTF{#1}{%
154     \udiss@streamtrue
155   }{%
156     \def\udiss@stream{#1}%
157     \udiss@streamtrue
158   }%
159 }
160 \NewDocumentCommand{ \discipline }{ m }{%
161   \IfBlankTF{#1}{%
162     \udiss@blank@discipline@err
163     \def\udiss@discipline{-NoValidDiscipline-}%
164   }{%
165     \def\udiss@discipline{#1}%
166   }%
167   \IfBlankTF{#1}{%
168     \udiss@disciplinetrue
169   }{%
170     \def\udiss@discipline{#1}%
171     \udiss@disciplinetrue
172   }%
173 }
174 \NewDocumentCommand{ \degree }{ m }{%
175   \IfBlankTF{#1}{%
176     \udiss@blank@degree@err
177     \def\udiss@degree{-NoValidDegree-}%
178   }{%

```

```

179     \def\udiss@degree{#1}%
180   }%
181   \IfBlankTF{#1}{%
182     \udiss@degreetrue
183   }{%
184     \def\udiss@degree{#1}%
185     \udiss@degreetrue
186   }%
187 }

```

(End of definition for *stream* and others. These functions are documented on page 3.)

We then develop the last set of *strict* macros which are used in the declaration and some on the title-page too. The method is no different.

```

supervisor
\supervisor
university
\university
department
\department
188 \newif\ifudiss@supervisor
189 \newif\ifudiss@university
190 \newif\ifudiss@department
191 \def\udiss@blank@supervisor@err{%
192   \ClassError{udiss}{%
193     The `supervisor' field cannot be blank%
194   }{%
195     You have provided an empty argument to the `supervisor'
196     field. This is not\MessageBreak permitted by the class.%
197   }%
198 }
199 \def\udiss@blank@university@err{%
200   \ClassError{udiss}{%
201     The `university' field cannot be blank%
202   }{%
203     You have provided an empty argument to the `university'
204     field. This is not\MessageBreak permitted by the class.%
205   }%
206 }
207 \def\udiss@blank@department@err{%
208   \ClassError{udiss}{%
209     The `department' field cannot be blank%
210   }{%
211     You have provided an empty argument to the `department'
212     field. This is not\MessageBreak permitted by the class.%
213   }%
214 }
215 \ekvdefinekeys{udiss}{%
216   code    supervisor      = {%
217     \def\udiss@tmp@supervisor{#1}%
218     \ifx\udiss@tmp@supervisor\empty
219       \def\udiss@supervisor{-NoValidSupervisor-}%
220       \udiss@blank@supervisor@err
221     \else
222       \def\udiss@supervisor{#1}%
223     \fi
224   },%
225   code    university      = {%
226     \def\udiss@tmp@university{#1}%

```



```

227 \ifx\udiss@tmp@university\empty
228 \def\udiss@university{-NoValidUniversity-}%
229 \udiss@blank@university@err
230 \else
231 \def\udiss@university{#1}%
232 \fi
233 },%
234 code department = {%
235 \def\udiss@tmp@department{#1}%
236 \ifx\udiss@tmp@department\empty
237 \def\udiss@department{-NoValidDepartment-}%
238 \udiss@blank@department@err
239 \else
240 \def\udiss@department{#1}%
241 \fi
242 }%
243 }
244 \NewDocumentCommand{ \supervisor }{ m }{%
245 \IfBlankTF{#1}{%
246 \udiss@blank@supervisor@err
247 \def\udiss@supervisor{-NoValidSupervisor-}%
248 }{%
249 \def\udiss@supervisor{#1}%
250 }%
251 \IfBlankTF{#1}{%
252 \udiss@supervisortrue
253 }{%
254 \def\udiss@supervisor{#1}%
255 \udiss@supervisortrue
256 }%
257 }
258 \NewDocumentCommand{ \university }{ m }{%
259 \IfBlankTF{#1}{%
260 \udiss@blank@university@err
261 \def\udiss@author{-NoValidUniversity-}%
262 }{%
263 \def\udiss@university{#1}%
264 }%
265 \IfBlankTF{#1}{%
266 \udiss@universitytrue
267 }{%
268 \def\udiss@university{#1}%
269 \udiss@universitytrue
270 }%
271 }
272 \NewDocumentCommand{ \department }{ m }{%
273 \IfBlankTF{#1}{%
274 \udiss@blank@department@err
275 \def\udiss@author{-NoValidDepartment-}%
276 }{%
277 \def\udiss@department{#1}%
278 }%
279 \IfBlankTF{#1}{%
280 \udiss@departmenttrue

```

```

281 }{
282   \def\udiss@department{#1}%
283   \udiss@departmenttrue
284 }%
285 }

```

(End of definition for *supervisor* and others. These functions are documented on page 3.)

**\logo** We develop the `\logo` command with a starred variant using the intuitive `xparse` trickery.  
**\logo\*** Notice the use of two different conditionals `\ifudiss@logo` and `\ifudiss@texlogo`.

```

286 \newif\ifudiss@logo
287 \newif\ifudiss@texlogo
288 \NewDocumentCommand{ \logo }{ s +m }{
289   \IfBooleanTF{#1}{
290     \def\udiss@texlogo{#2}%
291     \udiss@texlogotrue
292   }{
293     \def\udiss@logo{#2}%
294   }%
295   \udiss@logotrue
296 }

```

(End of definition for `\logo` and `\logo*`. These functions are documented on page 3.)

**logo** `expl3` doesn't have a way to have starred variants of the keys, but we can always create a  
**logo\*** different key with an additional star. That's what we do here.

```

297 \ekvdefinekeys{udiss}{
298   code logo = {
299     \def\udiss@logo{#1}%
300     \udiss@logotrue
301   },
302   code logo* = {
303     \def\udiss@texlogo{#1}%
304     \udiss@logotrue
305     \udiss@texlogotrue
306   }
307 }

```

(End of definition for `logo` and `logo*`. These functions are documented on page 3.)

Now we move to the complicated task of developing the mechanism for loading languages. In order to imitate the syntax of `babel`, we need a parser which interprets comma-separated language-names. We develop a new counter called `udiss@langbabel` and use it with `\alph` inside `\csname` in order to produce an ordered sequence of macros that store the different language-names. After parsing, the language macros should look like `\udiss@langbabel@a`, `\udiss@langbabel@a` etc. The following code does that. The `\udiss@lgbb1` macro is the internal parser for the language-names.

```

308 \newcounter{udiss@langbabel}
309 \stepcounter{udiss@langbabel}
310 \def\udiss@int@langbabel#1{
311   \expandafter\edef
312   \csname
313     udiss@langbabel@\alph{udiss@langbabel}%
314   \endcsname{#1}%

```

```

315 \stepcounter{udiss@langbabel}%
316 }
317 \NewDocumentCommand\udiss@lgbb1{ >{ \SplitList{,} } m }{%
318 \ProcessList{#1}{\udiss@int@langbabel}%
319 }

```

**language** We first create the if-conditionals required for this job.  
**ldfbabel**  
**nofontwarning**

```

320 \newif\ifudiss@lg@used
321 \newif\ifudiss@ldfbabel

```

The `language` key takes the comma-separated list as its argument and passes the list to the `\udiss@lgbb1` macro which we just saw. So whenever we pass multiple languages to the option `language`, it is parsed by `\udiss@lgbb1`. We then turn the `\ifudiss@lg@used` conditional on, so that when neither of these options are used by the user, we can do something.

```

322 \ekvdefinekeys{udiss}{%
323   code    language      = {%
324     \udiss@lgbb1{#1}%
325     \udiss@lg@usedtrue
326   },%

```

On the other hand though, the process is quite simple and straight forward. We need to store the list of the `.ldf`-languages in a macro and later pass it to the `babel` package as `babel` already understands how to parse it. We store the list in the `\udiss@ldfbabel@list` macro. With this option, we have to turn an additional conditional on, i.e., `\ifudiss@ldfbabel`. Usually we don't use the `.ldf` mechanism at all as mentioned in section 2.

```

327   code    ldfbabel      = {%
328     \def\udiss@ldfbabel@list{#1}%
329     \udiss@lg@usedtrue
330     \udiss@ldfbabeltrue
331   },%

```

The `nofontwarning` option is developed using the `invbool` *type* which creates (or uses, if present) a conditional and sets it to true. In this case, it is `\ifudiss@font@warning`. Unless the option `nofontwarning` is used, the aforementioned condition is true.

```

332   invbool nofontwarning = {\ifudiss@font@warning}%
333 }

```

*(End of definition for `language`, `ldfbabel`, and `nofontwarning`. These functions are documented on page 4.)*

**license** Now we develop the options for setting a license with the legit copyrightable years. As  
**copyright-yrs** usual, we first develop the required conditionals.  
**localcopyright**

```

334 \newif\ifudiss@free@culture
335 \newif\ifudiss@ccbysa
336 \newif\ifudiss@gfdl

```

We use a new `expkv` *types* for license which is `choice`. It takes a definite list of choices that run some arbitrary code. We define an internal macro `\udiss@license@notice` which holds the notice text of the selected licenses. We set the `\ifudiss@free@culture` conditional to true if the user's choice is `CC-BY-SA` or `GFDL`. Why we do this is because if the choice is `all-rights-reserved`, it need not be explicitly mentioned. It is always assumed. Also, it is seen in PDF's metadata. For this, we enable the conditional. Using

the initial *prefix* of `expkv`, we set the `all-rights-reserved` as the default choice of users.

```

337 \ekvdefinekeys{udiss}{%
338   choice license          = {%
339     all-rights-reserved    = {%
340       \def\udiss@license@notice{%
341         All rights reserved.\textLF
342       }%
343     },%
344     CC-BY-SA              = {%
345       \def\udiss@license@notice{%
346         This work is available under the "Creative Commons
347         Attribution ShareAlike 4.0 International" license.%
348       }%
349       \def\udiss@free@license@url{%
350         https://creativecommons.org/licenses/%
351         by-sa/4.0/legalcode.txt%
352       }%
353       \udiss@free@culturetrue
354       \udiss@ccbysatrue
355     },%
356     GFDL                  = {%
357       \def\udiss@license@notice{%
358         Permission is granted to copy, distribute and/or
359         modify this document under the terms of the GNU Free
360         Documentation License, Version 1.3 or any later
361         version published by the Free Software Foundation;
362         with no Invariant Sections, no Front-Cover Texts,
363         and no Back-Cover Texts. A copy of the license is
364         included in the section entitled "GNU Free
365         Documentation License".%
366       }%
367       \def\udiss@free@license@url{%
368         https://www.gnu.org/licenses/fdl-1.3.txt%
369       }%
370       \udiss@free@culturetrue
371       \udiss@gfdltrue
372     },%
373   },%
374   initial license          = {all-rights-reserved},%

```

Then we use the `store type` which, as the name suggests, stores the value given by the user in a macro. Again, with the *initial prefix*, we set the initial values for the two options.

```

375   store   copyright-yrs    = {\udiss@copyrightable@years},%
376   initial copyright-yrs    = {\the\year},%
377   store   localcopyright   = {\udiss@local@copyright},%
378   initial localcopyright   = {Copyright},%

```

(End of definition for *license*, *copyright-yrs*, and *localcopyright*. These functions are documented on page 5.)

**placeholders** This option pre-fills all the necessary fields with some random (and rather delusional :P) values. We use the *nmeta type* of `expkv` for this. This type uses the already developed keys

with some values. When the key developed with `nmeta` is used by the user, it behaves as if the user has passed the keys inside it with the corresponding values.

```

379 nmeta placeholders = {%
380     title           = {%
381         A groundbreaking dissertation%
382     },%
383     author          = {%
384         Jane Doe%
385     },%
386     pronouns        = {She/her},%
387     subtitle        = {%
388         A milestone study on a longstanding question
389         in a discipline%
390     },%
391     shorttitle      = {A Milestone study},%
392     stream          = {intriguing stream},%
393     discipline       = {rigorous discipline},
394     degree          = {enriching degree},%
395     supervisor      = {A kind supervisor},%
396     university      = {An ideal university},%
397     department      = {A homely department},%
398     license         = {CC-BY-SA},%
399     copyright-yrs   = {2022, 2023, 2024}%
400 }%
401 }
```

(End of definition for `placeholders`. This function is documented on page 5.)

`lot` The list of tables and figures are conditionals developed with the `bool` type of `expkv` which creates (or uses, if already present) a boolean. It is by default set to the false value. The `lof` declaration also is a boolean as it is turned off by default.

```

402 \ekvdefinekeys{udiss}{%
403     bool    lot           = {\ifudiss@lot},%
404     bool    lof          = {\ifudiss@lof},%
405     bool    declaration   = {\ifudiss@declaration},%

```

The `declarationtxt` key contains text which has paragraphs inside it and the default `key-values` of `expkv` are `\def` type (meaning, they don't accept `\par` tokens). So in order to have the effect of `\long\def`, we need to use `long prefix` of `expkv`. We set an initial value for this too. Then we store the value for the local word for declaration.

```

406 long
407 store declarationtxt = {\udiss@declaration},%
408 initial declarationtxt = {%
409     \noindent As required by the university regulations, I
410     wish to state that the work embodied in this thesis
411     titled \enquote{\udiss@title: \udiss@subtitle} forms
412     my own contribution to the research work carried out
413     under the guidance of \udiss@supervisor\ at the
414     \udiss@university, \udiss@department.
415
416     This work has not been submitted for any other degree
417     of this or any other university. Whenever references
418     have been made to previous works of others, it has

```

```

419     been clearly indicated as such and included in the
420     bibliography.
421
422     \bigskip
423     \flushright{%
424       Date: \today
425
426       \vspace{4em}%
427
428       Name: \udiss@author
429     }%
430
431     \bigskip\bigskip
432     \flushleft{%
433       Certified by:%
434
435       \vspace{4em}%
436
437       \udiss@supervisor
438     }%
439   },%
440   store decllocal      = {\udiss@decllocal},%
441   initial decllocal    = {Declaration}%
442 }

```

(End of definition for *lot* and others. These functions are documented on page 5.)

**print** The `print` key is created using the `bool` *type*. The `norefcolors` option, on the other hand, uses the `invbool` *type*. For `udisslinkclr`, `udissurlclr` and `udissciteclr` we use the `store` *type* along with some initial values.

```

443 \ekvdefinekeys{udiss}{%
444   bool    print          = {\ifudiss@print},%
445   invbool norefcolors    = {\ifudiss@colors},%
446   store   udisslinkclr   = {\udiss@int@link@color},%
447   initial udisslinkclr   = {udisslink},%
448   store   udissurlclr    = {\udiss@int@url@color},%
449   initial udissurlclr    = {udissurl},%
450   store   udissciteclr   = {\udiss@int@cite@color},%
451   initial udissciteclr   = {udisscite}%
452 }

```

(End of definition for *print* and others. These functions are documented on page 6.)

**blind** The next boolean is for printing blind documents.

```

453 \ekvdefinekeys{udiss}{%
454   bool    blind          = {\ifudiss@blind}%
455 }

```

(End of definition for *blind*. This function is documented on page 5.)

**framed-title** We provide a conditional for producing a framed title-page. The conditional will be used later while typesetting.

```

456 \ekvdefinekeys{udiss}{%
457   bool    framed-title   = {\ifudiss@ftitle}%
458 }

```

(End of definition for *framed-title*. This function is documented on page 5.)

**oldone** We set the old style numerals as the default for typesetting. We provide an inverse boolean  
**oldstylenumoff** for setting the old style numbers off. The special shape for number one is enabled only  
with the boolean key **oldone**. We develop it here.

```
459 \ekvdefinekeys{udiss}{%
460   invbool oldstylenumoff = {\ifudiss@oldstylenum},%
461   bool    oldone        = {\ifudiss@oldone}%
462 }
```

(End of definition for *oldone* and *oldstylenumoff*. These functions are documented on page 7.)

**explicitext** To turn on the explicit extension mechanism of the font-loading packages, we provide a  
**ignorefsoff** conditional. It is disabled by default and hence is of the **bool** type.

```
463 \ekvdefinekeys{udiss}{%
464   bool    explicitext  = {\ifudiss@explicitext},%
465   invbool ignorefsoff = {\ifudiss@ignorefontspec}%
466 }
```

(End of definition for *explicitext* and *ignorefsoff*. These functions are documented on page 8.)

**fulfilment** In order to store the sentence written on the title-page, we again use the common **store**  
and **initial store** pair.

```
467 \ekvdefinekeys{udiss}{%
468   store   fulfilment   = {\udiss@fulfilment},%
469   initial fulfilment   = {%
470     A dissertation submitted in the partial fulfilment of
471     the requirements for the \udiss@degree\ program in
472     \udiss@discipline.%
473   }%
474 }
```

(End of definition for *fulfilment*. This function is documented on page 7.)

All of the upcoming *keys* are font-selection keys. We set their defaults as explained  
in table I.

**rmfont** The following sets the default text font for the document. We are using NewCM family  
**rmbffont** as the default font-family. But as of now, the NewCM fonts don't have all the variants.  
**rmitfont** E.g., it lacks slanted/oblique shapes in the natural weight, but we provide the regular  
**rmbfitfont** ones in place of them.

```
475 \ekvdefinekeys{udiss}{%
476   store   rmfont      = {\udiss@rmfont},%
477   initial rmfont      = {NewCM10-Book.otf},%
478   store   rmbffont    = {\udiss@rmbffont},%
479   initial rmbffont    = {NewCM10-Bold.otf},%
480   store   rmitfont    = {\udiss@rmitfont},%
481   initial rmitfont    = {NewCM10-BookItalic.otf},%
482   store   rmbfitfont  = {\udiss@rmbfitfont},%
483   initial rmbfitfont  = {NewCM10-BoldItalic.otf},%
484   store   rmslfont    = {\udiss@rmslfont},%
485   initial rmslfont    = {NewCM10-Book.otf},%
486   store   rmbfslfont  = {\udiss@rmbfslfont},%
487   initial rmbfslfont  = {NewCM10-Bold.otf},%
```

(End of definition for *rmfont* and others. These functions are documented on page 9.)

**sffont** Similarly, we store values for the sans fonts too. NewCM lacks certain variants in sans,  
**sfbffont** but again, the options are set font-agnostically and hence they are more in number than  
**sffitfont** the actually available variants. The sans fonts in NewCM, don't have an Italic design.  
**sfbfifont** We have used the oblique version of the fonts there instead. Note that the same designs  
**sfsfont** are used for the slanted fonts too.  
**sfbfslfont**

```
488 store sffont = {\udiss@sffont},%
489 initial sffont = {NewCMSans10-Book.otf},%
490 store sfbffont = {\udiss@sfbffont},%
491 initial sfbffont = {NewCMSans10-Bold.otf},%
492 store sffitfont = {\udiss@sffitfont},%
493 initial sffitfont = {NewCMSans10-BookOblique.otf},%
494 store sfbfifont = {\udiss@sfbfifont},%
495 initial sfbfifont = {NewCMSans10-BoldOblique.otf},%
496 store sfsfont = {\udiss@sfsfont},%
497 initial sfsfont = {NewCMSans10-BookOblique.otf},%
498 store sfbfslfont = {\udiss@sfbfslfont},%
499 initial sfbfslfont = {NewCMSans10-BoldOblique.otf},%
```

(End of definition for *sffont* and others. These functions are documented on page 9.)

**ttfont** The settings for typewriter fonts are loaded in the below code. We have a strange situation  
**ttbffont** here. NewCM provides *BookItalic*, but doesn't have a *BoldItalic*. We instead have  
**ttitfont** *BoldOblique*. Thus we set the *BookItalic* as the *ttitfont* and *BoldOblique* as the  
**ttbfifont** *ttbfifont* font. The same pair for *ttsfont* and *ttbfslfont*.  
**ttsfont**  
**ttbfslfont**

```
500 store ttfont = {\udiss@ttfont},%
501 initial ttfont = {NewCMMono10-Book.otf},%
502 store ttbffont = {\udiss@ttbffont},%
503 initial ttbffont = {NewCMMono10-Bold.otf},%
504 store ttitfont = {\udiss@ttitfont},%
505 initial ttitfont = {NewCMMono10-BookItalic.otf},%
506 store ttbfifont = {\udiss@ttbfifont},%
507 initial ttbfifont = {NewCMMono10-BoldOblique.otf},%
508 store ttsfont = {\udiss@ttsfont},%
509 initial ttsfont = {NewCM10-Book.otf},%
510 store ttbfslfont = {\udiss@ttbfslfont},%
511 initial ttbfslfont = {NewCMMono10-BoldOblique.otf},%
```

(End of definition for *ttfont* and others. These functions are documented on page 9.)

**mathfont** Maths fonts have very little variation, so we just have two parameters for selecting the  
**mathbffont** fonts and two of the usual parameters for features and extensions.

```
512 store mathfont = {\udiss@mathfont},%
513 initial mathfont = {NewCMMath-Book.otf},%
514 store mathbffont = {\udiss@mathbffont},%
515 initial mathbffont = {NewCMMath-Bold.otf},%
```

(End of definition for *mathfont* and *mathbffont*. These functions are documented on page 9.)

**rmfeatures** If the user wants to add features to the selected fonts, they are allowed to use these  
**rmfntext** commands and if they use the *explicittext* option, they can choose the font extensions  
**sffeatures**  
**sffntext**  
**ttfeatures**  
**ttfntext**  
**mathfeatures**  
**mathfntext**



too using the *keys* specifically developed for that. We use `store` and `choice types` for this.

```

516 store   rmfeatures    = {\udiss@rmfeatures},%
517 choice  rmfntext      = {%
518         ttf           = {%
519             \ifudiss@explicitext
520             \def\udiss@rmfntext{ttf}%
521         \else
522             \ClassError{udiss}{%
523                 Option `explicitext' is needed for adding
524                 font extensions%
525             }{%
526                 You have loaded font-option `rmfntext' to
527                 add extension to font-names.
528                 This\MessageBreak option is supposed to be
529                 used only with the `explicitext' option.
530                 E.g., for\MessageBreak loading, say,
531                 FreeSerif, you may choose one of the
532                 following two methods:%
533                 \MessageBreak\MessageBreak
534                 1. The recommended method:%
535                 \MessageBreak
536                 \space\space\space
537                 ----8<-----8<-----%
538                 \MessageBreak
539                 \space\space\space
540                 \string\documentclass[\@percentchar
541                 \MessageBreak
542                 \space\space\space\space\space
543                 rmfont = {FreeSerif.ttf}\@percentchar
544                 \MessageBreak
545                 \space\space\space
546                 ]{udiss}\MessageBreak
547                 \space\space\space
548                 ----8<-----8<-----%
549                 \MessageBreak\MessageBreak
550                 2. The alternative:%
551                 \MessageBreak
552                 \space\space\space
553                 ----8<-----8<-----%
554                 \MessageBreak
555                 \space\space\space
556                 \string\documentclass[\@percentchar
557                 \MessageBreak
558                 \space\space\space\space\space
559                 explicitext,\@percentchar
560                 \MessageBreak
561                 \space\space\space\space\space
562                 rmfont = {FreeSerif},\@percentchar
563                 \MessageBreak
564                 \space\space\space\space\space
565                 rmfntext = {ttf}\@percentchar
566                 \MessageBreak
567                 \space\space\space

```

```

568         ]{udiss}%
569         \MessageBreak
570         \space\space\space
571         ----8<-----8<----%
572     }%
573     \fi
574 },%
575 otf         = {%
576     \ifudiss@explicittext
577     \def\udiss@rmfntext{otf}%
578     \else
579     \ClassError{udiss}{%
580     Option `explicittext' is needed for adding
581     font extensions%
582     }{%
583     You have loaded font-option `rmfntext' to
584     add extension to font-names.
585     This\MessageBreak option is supposed to be
586     used only with the `explicittext' option.
587     E.g., for\MessageBreak loading, say,
588     KpRoman-Regular.otf, you may choose one of
589     the following two\MessageBreak
590     methods:%
591     \MessageBreak\MessageBreak
592     1. The recommended method:%
593     \MessageBreak
594     \space\space\space
595     ----8<-----8<----%
596     \MessageBreak
597     \space\space\space
598     \string\documentclass[\@percentchar
599     \MessageBreak
600     \space\space\space\space\space
601     rmfont = {KpRoman-Regular.otf}%
602     \@percentchar
603     \MessageBreak
604     \space\space\space
605     ]{udiss}\MessageBreak
606     \space\space\space
607     ----8<-----8<----%
608     \MessageBreak
609     \MessageBreak
610     2. The alternative:%
611     \MessageBreak
612     \space\space\space
613     ----8<-----8<----%
614     \MessageBreak
615     \space\space\space
616     \string\documentclass[\@percentchar
617     \MessageBreak
618     \space\space\space\space\space
619     explicittext,\@percentchar
620     \MessageBreak
621     \space\space\space\space\space

```

```

622         rfont = {KpRoman-Regular},%
623         \@percentchar
624         \MessageBreak
625         \space\space\space\space\space
626         rfonttext = {otf}\@percentchar
627         \MessageBreak
628         \space\space\space
629     ]{udiss}%
630     \MessageBreak
631     \space\space\space
632     ----8<-----8<-----%
633 }%
634 \fi
635 }%
636 },%
637 store sffeatures = {\udiss@sffeatures},%
638 choice sffntext = {%
639     ttf = {%
640         \ifudiss@explicitext
641         \def\udiss@sffntext{ttf}%
642         \else
643         \ClassError{udiss}{%
644             Option `explicitext' is needed for adding
645             font extensions%
646         }{%
647             You have loaded font-option `sffntext' to
648             add extension to font-names.
649             This\MessageBreak option is supposed to be
650             used only with the `explicitext' option.
651             E.g., for\MessageBreak loading, say,
652             FreeSans, you may choose one of the
653             following two methods:%
654             \MessageBreak\MessageBreak
655             1. The recommended method:%
656             \MessageBreak
657             \space\space\space
658             ----8<-----8<-----%
659             \MessageBreak
660             \space\space\space
661             \string\documentclass[\@percentchar
662             \MessageBreak
663             \space\space\space\space\space
664             sffont = {FreeSans.ttf}\@percentchar
665             \MessageBreak
666             \space\space\space
667             ]{udiss}\MessageBreak
668             \space\space\space
669             ----8<-----8<-----%
670             \MessageBreak\MessageBreak
671             2. The alternative:%
672             \MessageBreak
673             \space\space\space
674             ----8<-----8<-----%
675             \MessageBreak

```

```

676         \space\space\space
677         \string\documentclass[\@percentchar
678         \MessageBreak
679         \space\space\space\space\space
680         explicittext,\@percentchar
681         \MessageBreak
682         \space\space\space\space\space
683         sffont = {FreeSans},\@percentchar
684         \MessageBreak
685         \space\space\space\space\space
686         sffntext = {ttf}\@percentchar
687         \MessageBreak
688         \space\space\space
689         ]{udiss}%
690         \MessageBreak
691         \space\space\space
692         ----8<-----8<-----%
693     }%
694     \fi
695 },%
696 otf          = {%
697     \ifudiss@explicittext
698     \def\udiss@sffntext{otf}%
699     \else
700     \ClassError{udiss}{%
701     Option `explicittext' is needed for adding
702     font extensions%
703     }{%
704     You have loaded font-option `sffntext' to
705     add extension to font-names.
706     This\MessageBreak option is supposed to be
707     used only with the `explicittext' option.
708     E.g., for\MessageBreak loading, say,
709     KpSans-Regular.otf, you may choose one of
710     the following two\MessageBreak
711     methods:%
712     \MessageBreak\MessageBreak
713     1. The recommended method:%
714     \MessageBreak
715     \space\space\space
716     ----8<-----8<-----%
717     \MessageBreak
718     \space\space\space
719     \string\documentclass[\@percentchar
720     \MessageBreak
721     \space\space\space\space\space
722     sffont = {KpSans-Regular.otf}%
723     \@percentchar
724     \MessageBreak
725     \space\space\space
726     ]{udiss}\MessageBreak
727     \space\space\space
728     ----8<-----8<-----%
729     \MessageBreak

```

```

730         \MessageBreak
731         2. The alternative:%
732         \MessageBreak
733         \space\space\space
734         ----8<-----8<-----%
735         \MessageBreak
736         \space\space\space
737         \string\documentclass[\@percentchar
738         \MessageBreak
739         \space\space\space\space\space
740         explicittext,\@percentchar
741         \MessageBreak
742         \space\space\space\space\space
743         sffont = {KpSans-Regular},%
744         \@percentchar
745         \MessageBreak
746         \space\space\space\space\space
747         sffntext = {otf}\@percentchar
748         \MessageBreak
749         \space\space\space
750         ]{udiss}%
751         \MessageBreak
752         \space\space\space
753         ----8<-----8<-----%
754     }%
755     \fi
756 }%
757 },%
758 store ttfeatures = {\udiss@ttfeatures},%
759 choice ttfntext = {%
760     ttf = {%
761         \ifudiss@explicittext
762         \def\udiss@ttfntext{ttf}%
763     \else
764         \ClassError{udiss}{%
765             Option `explicittext' is needed for adding
766             font extensions%
767         }{%
768             You have loaded font-option `ttfntext' to
769             add extension to font-names.
770             This\MessageBreak option is supposed to be
771             used only with the `explicittext' option.
772             E.g., for\MessageBreak loading, say,
773             FreeMono, you may choose one of the
774             following two methods:%
775             \MessageBreak\MessageBreak
776             1. The recommended method:%
777             \MessageBreak
778             \space\space\space
779             ----8<-----8<-----%
780             \MessageBreak
781             \space\space\space
782             \string\documentclass[\@percentchar
783             \MessageBreak

```

```

784         \space\space\space\space\space
785         ttfont = {FreeMono.ttf}\@percentchar
786         \MessageBreak
787         \space\space\space
788     ]{udiss}\MessageBreak
789     \space\space\space
790     ----8<-----8<----%
791     \MessageBreak\MessageBreak
792     2. The alternative:%
793     \MessageBreak
794     \space\space\space
795     ----8<-----8<----%
796     \MessageBreak
797     \space\space\space
798     \string\documentclass[\@percentchar
799         \MessageBreak
800         \space\space\space\space\space
801         explicittext,\@percentchar
802         \MessageBreak
803         \space\space\space\space\space
804         ttfont = {FreeMono},\@percentchar
805         \MessageBreak
806         \space\space\space\space\space
807         ttfonttext = {ttf}\@percentchar
808         \MessageBreak
809         \space\space\space
810     ]{udiss}%
811     \MessageBreak
812     \space\space\space
813     ----8<-----8<----%
814     }%
815     \fi
816 },%
817 otf          = {%
818     \ifudiss@explicittext
819     \def\udiss@ttfonttext{otf}%
820     \else
821     \ClassError{udiss}{%
822         Option `explicittext' is needed for adding
823         font extensions%
824     }{%
825         You have loaded font-option `ttfonttext' to
826         add extension to font-names.
827         This\MessageBreak option is supposed to be
828         used only with the `explicittext' option.
829         E.g., for\MessageBreak loading, say,
830         KpMono-Regular.otf, you may choose one of
831         the following two\MessageBreak
832         methods:%
833         \MessageBreak\MessageBreak
834         1. The recommended method:%
835         \MessageBreak
836         \space\space\space
837         ----8<-----8<----%

```

```

838         \MessageBreak
839         \space\space\space
840         \string\documentclass[\@percentchar
841         \MessageBreak
842         \space\space\space\space\space
843         ttfont = {KpMono-Regular.otf}%
844         \@percentchar
845         \MessageBreak
846         \space\space\space
847     ]{udiss}\MessageBreak
848     \space\space\space
849     ----8<-----8<-----%
850     \MessageBreak
851     \MessageBreak
852     2. The alternative:%
853     \MessageBreak
854     \space\space\space
855     ----8<-----8<-----%
856     \MessageBreak
857     \space\space\space
858     \string\documentclass[\@percentchar
859     \MessageBreak
860     \space\space\space\space\space
861     explicittext,\@percentchar
862     \MessageBreak
863     \space\space\space\space\space
864     ttfont = {KpMono-Regular},%
865     \@percentchar
866     \MessageBreak
867     \space\space\space\space\space
868     ttfntext = {otf}\@percentchar
869     \MessageBreak
870     \space\space\space
871     ]{udiss}%
872     \MessageBreak
873     \space\space\space
874     ----8<-----8<-----%
875     }%
876     \fi
877     }%
878     },%
879     store mathfeatures = {\udiss@mathfeatures},%
880     choice mathfntext = {%
881         ttf = {%
882             \ifudiss@explicittext
883             \def\udiss@mathfntext{ttf}%
884             \else
885             \ClassError{udiss}{%
886                 Option `explicittext' is needed for adding
887                 font extensions%
888             }{%
889                 You have loaded font-option `mathfntext'
890                 to add extension to font-names.
891                 This\MessageBreak option is supposed to be

```

```

892         used only with the `explicittext' option.
893         E.g., for\MessageBreak loading, say,
894         NotoSansMath-Regular, you may choose one
895         of the following two\MessageBreak
896         methods:%
897         \MessageBreak\MessageBreak
898         1. The recommended method:%
899         \MessageBreak
900         \space\space\space
901         ----8<-----8<-----%
902         \MessageBreak
903         \space\space\space
904         \string\documentclass[\@percentchar
905         \MessageBreak
906         \space\space\space\space\space
907         mathfont = {NotoSansMath-Regular.ttf}%
908         \@percentchar
909         \MessageBreak
910         \space\space\space
911         ]{udiss}\MessageBreak
912         \space\space\space
913         ----8<-----8<-----%
914         \MessageBreak\MessageBreak
915         2. The alternative:%
916         \MessageBreak
917         \space\space\space
918         ----8<-----8<-----%
919         \MessageBreak
920         \space\space\space
921         \string\documentclass[\@percentchar
922         \MessageBreak
923         \space\space\space\space\space
924         explicittext,\@percentchar
925         \MessageBreak
926         \space\space\space\space\space
927         mathfont = {NotoSansMath-Regular},%
928         \@percentchar
929         \MessageBreak
930         \space\space\space\space\space
931         mathfntext = {tff}\@percentchar
932         \MessageBreak
933         \space\space\space
934         ]{udiss}%
935         \MessageBreak
936         \space\space\space
937         ----8<-----8<-----%
938     }%
939     \fi
940 },%
941 otf = {%
942 \ifudiss@explicittext
943 \def\udiss@mathfntext{otf}%
944 \else
945 \ClassError{udiss}{%

```



```

946         Option `explicittext' is needed for adding
947         font extensions%
948     }{%
949         You have loaded font-option `mathfntext'
950         to add extension to font-names.
951         This\MessageBreak option is supposed to be
952         used only with the `explicittext' option.
953         E.g., for\MessageBreak loading, say,
954         XITSMath-Regular, you may choose one of
955         the following two\MessageBreak methods:%
956         \MessageBreak\MessageBreak
957         1. The recommended method:%
958         \MessageBreak
959         \space\space\space
960         ----8<-----8<-----%
961         \MessageBreak
962         \space\space\space
963         \string\documentclass[\@percentchar
964         \MessageBreak
965         \space\space\space\space\space
966         mathfont = {XITSMath-Regular.otf}%
967         \@percentchar
968         \MessageBreak
969         \space\space\space
970     ]{udiss}\MessageBreak
971     \space\space\space
972     ----8<-----8<-----%
973     \MessageBreak\MessageBreak
974     2. The alternative:%
975     \MessageBreak
976     \space\space\space
977     ----8<-----8<-----%
978     \MessageBreak
979     \space\space\space
980     \string\documentclass[\@percentchar
981     \MessageBreak
982     \space\space\space\space\space
983     explicittext,\@percentchar
984     \MessageBreak
985     \space\space\space\space\space
986     mathfont = {XITSMath-Regular},%
987     \@percentchar
988     \MessageBreak
989     \space\space\space\space\space
990     mathfntext = {otf}\@percentchar
991     \MessageBreak
992     \space\space\space
993     ]{udiss}%
994     \MessageBreak
995     \space\space\space
996     ----8<-----8<-----%
997     }%
998     \fi
999 }%

```

```

1000     },%
1001 }

```

(End of definition for *rmfeatures* and others. These functions are documented on page 8.)

**titleshape** We store various variables in these keys for customising the layout of the title-page.  
**titlefont** The following are the keys specifically developed for customising the title. The sizing  
**titlefeat** is done with internal macros specifically designed for consistent size-changing. We use  
**titlesize** fontspec-scaling for that.  
**titlecolor**

```

1002 \ekvdefinekeys{udiss}{%
1003   store   titleshape   = {\udiss@title@shape},%
1004   initial titleshape   = {\bfseries},%
1005   store   titlefont    = {\udiss@title@font},%
1006   store   titlesize    = {\udiss@title@size},%
1007   initial titlesize    = {\udiss@int@huge},%
1008   store   titlecolor   = {\udiss@title@color},%
1009   initial titlecolor   = {black},%
1010   store   titlefeat    = {\udiss@title@fontfeats},%
1011   initial titlefeat    = {}%
1012 }

```

(End of definition for *titleshape* and others. These functions are documented on page 10.)

**subtitleshape** The same logic is applied for subtitles and these variables are created.

```

1013 \ekvdefinekeys{udiss}{%
1014   store   subtitleshape = {\udiss@subtitle@shape},%
1015   initial subtitleshape = {\normalfont},%
1016   store   subtitlefont  = {\udiss@subtitle@font},%
1017   store   subtitlesize  = {\udiss@subtitle@size},%
1018   initial subtitlesize  = {\udiss@int@large},%
1019   store   subtitlecolor = {\udiss@subtitle@color},%
1020   initial subtitlecolor = {black},%
1021   store   subtitlefeat  = {\udiss@subtitle@fontfeats},%
1022   initial subtitlefeat  = {}%
1023 }

```

(End of definition for *subtitleshape* and others. These functions are documented on page 10.)

**authorshape** Similarly, we have the following keys for author.

```

1024 \ekvdefinekeys{udiss}{%
1025   store   authorshape  = {\udiss@author@shape},%
1026   initial authorshape  = {\normalfont},%
1027   store   authorfont   = {\udiss@author@font},%
1028   store   authorsize   = {\udiss@author@size},%
1029   initial authorsize   = {\udiss@int@large},%
1030   store   authorcolor  = {\udiss@author@color},%
1031   initial authorcolor  = {black},%
1032   store   authorfeat   = {\udiss@author@fontfeats},%
1033   initial authorfeat   = {}%
1034 }

```

(End of definition for *authorshape* and others. These functions are documented on page 10.)

**unishape** The same pattern for printing the university name.  
**unifont**  
**unisize**  
**unicolor**  
**uniffeat**

```

1035 \ekvdefinekeys{udiss}{%
1036   store   unishape   = {\udiss@uni@shape},%
1037   initial unishape   = {\itshape},%
1038   store   unifont    = {\udiss@uni@font},%
1039   store   unisize    = {\udiss@uni@size},%
1040   initial unisize    = {\normalsize},%
1041   store   unicolor   = {\udiss@uni@color},%
1042   initial unicolor   = {black},%
1043   store   uniffeat   = {\udiss@uni@fontfeats},%
1044 }

```

*(End of definition for unishape and others. These functions are documented on page 10.)*

**deptshape** The following keys for the department.  
**deptfont**  
**deptsize**  
**deptcolor**  
**deptfeat**

```

1045 \ekvdefinekeys{udiss}{%
1046   store   deptshape  = {\udiss@dept@shape},%
1047   initial deptshape  = {\normalfont},%
1048   store   deptfont   = {\udiss@dept@font},%
1049   store   deptsize   = {\udiss@dept@size},%
1050   initial deptsize   = {\udiss@int@small},%
1051   store   deptcolor  = {\udiss@dept@color},%
1052   initial deptcolor  = {black},%
1053   store   deptfeat   = {\udiss@dept@fontfeats},%
1054   initial deptfeat   = {}%
1055 }

```

*(End of definition for deptshape and others. These functions are documented on page 10.)*

**ffshape** These are for the fulfilment-text.  
**fffont**  
**ffsize**  
**ffcolor**  
**fffeat**

```

1056 \ekvdefinekeys{udiss}{%
1057   store   ffshape    = {\udiss@fulfilment@shape},%
1058   initial ffshape    = {\normalfont},%
1059   store   fffont     = {\udiss@fulfilment@font},%
1060   store   fffsize    = {\udiss@fulfilment@size},%
1061   initial fffsize    = {\udiss@int@small},%
1062   store   fffcolor   = {\udiss@fulfilment@color},%
1063   initial fffcolor   = {black},%
1064   store   fffeat     = {\udiss@fulfilment@fontfeats},%
1065   initial fffeat     = {}%
1066 }

```

*(End of definition for ffshape and others. These functions are documented on page 10.)*

**logowidth** The following parameters save the values for the logo-height and the logo-width. These  
**logoheight** are the last options after which we \Process our options.

```

1067 \ekvdefinekeys{udiss}{%
1068   store   logowidth  = {\udiss@logo@width},%
1069   initial logowidth  = {0.3},%
1070   store   logoheight = {\udiss@logo@hght},%
1071   initial logoheight = {0.075},%
1072 }
1073 \ekvoProcessGlobalOptions{udiss}

```

(End of definition for `logowidth` and `logoheight`. These functions are documented on page 9.)

`\dissertationstyle` We develop the command for loading custom styles. It just inputs files starting with `udiss-style-` followed by the designated tag, e.g., `ukerala` for the university of Kerala.

```

1074 \NewDocumentCommand{ \dissertationstyle }{ m }{%
1075   \AddToHook{begindocument/before}{%
1076     \input{udiss-style-#1}%
1077   }%
1078 }

```

(End of definition for `\dissertationstyle`. This function is documented on page 7.)

To minimise loading a lot of packages, we use the `memoir` class which provides many features useful for document-creation. But we also need to set print and digital versions of the dissertations. For that we use our conditional.

```

1079 \ifudiss@print
1080 \else
1081   \PassOptionsToClass{oneside}{memoir}%
1082 \fi
1083 \LoadClass{memoir}

```

`\title` We defined most other commands as per the flow of the documentation, but as mentioned before, we had kept `\title` and `\author` aside. As mentioned before, we did so because these commands are defined by the `memoir` class. If we had defined it before loading the `memoir` class (which we just loaded), the commands would have got renewed after loading the class and thus our internal macros would have failed. On the other hand, we could not have loaded the `memoir` class in the beginning, because we use the `\ifudiss@print` conditional, i.e., developed with the help of `expl\VIDEF` package. Even if we had separated the code of `print` key, it would have been necessary to process the keys using `\ekvoProcessGlobalOptions`, but we could not have done that since there were a lot of other options pending. This is why, we develop the `\title` and `\author` commands now.

```

1084 \RenewDocumentCommand{ \title }{ m }{%
1085   \IfBlankTF{#1}{%
1086     \udiss@blank@title@err
1087     \def\udiss@title{-NoValidTitle-}%
1088   }{%
1089     \def\udiss@title{#1}%
1090   }%
1091 }
1092 \RenewDocumentCommand{ \author }{ m }{%
1093   \IfBlankTF{#1}{%
1094     \udiss@blank@author@err
1095     \def\udiss@author{-NoValidAuthor-}%
1096   }{%
1097     \def\udiss@author{#1}%
1098   }%
1099 }

```

(End of definition for `\title` and `\author`. These functions are documented on page 3.)

We are using OpenType fonts for better accessibility. They are incompatible with the classic PDF $\LaTeX$  and require either Lua $\LaTeX$  or Xe $\LaTeX$  for compilation. To ensure

that one of these two is loaded, we load the `iftex` package and mandate a Unicode engine with the `\RequireTUTeX` command.

```

II00 \RequirePackage{iftex}
II01 \RequireTUTeX

```

We used custom `udiss` size commands. They are developed with `fontspec-scaling` as follows:

```

II02 \def\udiss@int@HUGE{%
II03   \addfontfeature{Scale={2.5}}%
II04 }
II05 \def\udiss@int@Huge{%
II06   \addfontfeature{Scale={2}}%
II07 }
II08 \def\udiss@int@huge{%
II09   \addfontfeature{Scale={1.7}}%
II10 }
II11 \def\udiss@int@LARGE{%
II12   \addfontfeature{Scale={1.4}}%
II13 }
II14 \def\udiss@int@Large{%
II15   \addfontfeature{Scale={1.2}}%
II16 }
II17 \def\udiss@int@large{%
II18   \addfontfeature{Scale={1.1}}%
II19 }
II20 \def\udiss@int@small{%
II21   \addfontfeature{Scale={0.9}}%
II22 }
II23 \def\udiss@int@footnotesize{%
II24   \addfontfeature{Scale={0.8}}%
II25 }
II26 \def\udiss@int@scriptsize{%
II27   \addfontfeature{Scale={0.7}}%
II28 }
II29 \def\udiss@int@tiny{%
II30   \addfontfeature{Scale={0.6}}%
II31 }
II32 \def\udiss@int@miniscule{%
II33   \addfontfeature{Scale={0.5}}%
II34 }

```

We require the `babel` package for loading the language settings. We have the following three situations with loading languages in `udiss`:

1. Users don't load any of the language options at all.
2. Users load languages with option `ldfbabel`.
3. Users load languages with option `language`.

If a user has used `ldfbabel`, then the macro which stores the language-list is directly passed to the package using `\PassOptionsToPackage`.

```

II35 \ifudiss@lg@used
II36   \ifudiss@ldfbabel

```

```

II37     \PassOptionsToPackage{%
II38         \udiss@ldfbabel@list
II39     }{babel}%
II40 \fi

```

If user hasn't loaded any language options, we assume that they have no intent to load languages that require .ldf files and hence we mandate loading .ini files for all the languages. In addition to that, we load `english` as the main language. This takes care of the first two situations.

```

II41 \else
II42     \PassOptionsToPackage{english,provide***}{babel}%
II43 \fi
II44 \RequirePackage{babel}

```

We also load package `iflang` for testing language-names.

```

II45 \RequirePackage{iflang}

```

First we check if the user has loaded `ldf` languages. We ignore that branch completely. If it is not loaded and they have loaded exactly one language, then we set that language as the main language. Note that when user uses one language, the value of the counter is 2 as it is increased by the counter each time.

```

II46 \newcounter{udiss@tmp@langcnt}
II47 \ifudiss@ldfbabel
II48 \else
II49     \ifudiss@lg@used
II50         \ifnum\theudiss@langbabel=2\relax
II51             \addtocounter{udiss@langbabel}{-1}%
II52             \ifLuaTeX
II53                 \babelprovide[%
II54                     main,%
II55                     import,%
II56                     onchar           ={\ids fonts}%
II57                 ]{%
II58                     \csname
II59                         udiss@langbabel@\alph{udiss@langbabel}%
II60                     \endcsname
II61                 }%
II62             \else
II63                 \babelprovide[main,import]{%
II64                     \csname
II65                         udiss@langbabel@\alph{udiss@langbabel}%
II66                     \endcsname
II67                 }%
II68             \fi
II69 \fi

```

When users use multiple languages, the situation is a bit more complicated. Because the parser parses and increases the counter each time. Suppose, the user provided 5 languages, the parser will create 5 macros *and* increase the counter for one last time resulting in the value 6. In order to stop processing when the number of actually used languages is reached, we need to store the current number *minus one* because of this. We have to start a loop from value 1 till it reaches the last value. For this, we reset the counter. In order to stop when all the languages are parsed, we need to remember the actual value of the counter somewhere. Hence we create a temporary counter for this and equate it

with the value of our actual counter minus 1. Note that we escape `\ifudiss@ldfbabel` too. Now we start the calculation when the counter's value is 3 or more, basically the author has to use at least two languages in order to activate the loop ahead.

```

1170     \ifnum\theudiss@langbabel>2\relax
1171         \setcounter{udiss@tmp@langcnt}{%
1172             \numexpr\theudiss@langbabel -1\relax
1173         }%

```

We now can safely manipulate our actual counter. Starting from the first language, we load all of them with the `\babelprovide` command. We reset the counter to 1:

```

1174     \setcounter{udiss@langbabel}{1}%

```

Now we start a loop.

```

1175     \loop

```

Only Lua<sub>La</sub>T<sub>E</sub>X supports the `ids fonts` parameter with `\babelprovide` for applying fonts to the input text with respect to their Unicode slots. That is why we start the `\ifLuaTeX` conditional and load the `onchar` option. Loading a conditional inside `\babelprovide` wasn't permitted because of expansion, hence we load the entire command in the else branch. Notice that there is no space after the = sign of the `onchar` option. To know why, you may want to read this wonderful answer by Jonathan P. Spratte: <https://topanswers.xyz/tex?q=8039#a7652>.

```

1176         \ifLuaTeX
1177             \babelprovide[%
1178                 import,%
1179                 onchar           ={ids fonts}%
1180             ]{%

```

Now we pass `\udiss@langbabel@a` as an argument to `\babelprovide` command (the value of the counter is as of now 1).

```

1181             \csname
1182                 udiss@langbabel@\alph{udiss@langbabel}%
1183             \endcsname
1184         }%

```

We start the `\else` branch and use the same thing again, just without the `onchar` option .

```

1185         \else
1186             \babelprovide[import]{%
1187                 \csname
1188                     udiss@langbabel@\alph{udiss@langbabel}%
1189                 \endcsname
1190             }%
1191     \fi

```

Woohoo! We loaded the first language successfully. Now, we increment the counter, but there is another situation which we have to take into account. What if the user just loaded a single language with the `language` option? The loop will run into trouble, that's why we only run the loop if the value of the total number of languages is more than one. We had saved it in a temporary counter.

```

1192         \ifnum\theudiss@tmp@langcnt=1\relax
1193         \else
1194             \addtocounter{udiss@langbabel}{1}%
1195         \fi

```

Now the value of the counter is 2 which means the second language. We need to again load the same settings for it, but we can't keep doing this manually since 1) we are lazy (the most important) 2) we have no idea how many languages the user has loaded, so now we actually activate the `\loop` that we had started with the `\repeat` command. It runs all the commands seen after the `\loop` command (this includes the one used for increasing the counter too) till the value of the counter is less than the value of `udiss@tmp@langcnt`. Remember we had already set it to the value of the original counter (before setting it to 1) *minus one*. Now the next command will stop at one number even lesser than it as we have used the `<` relation while comparing. Also, here we will close the if-conditional that checked if more than one languages were used or not.

```

1196         \ifnum\theudiss@langbabel<\theudiss@tmp@langcnt
1197         \relax
1198         \repeat
1199     \fi

```

Are you wondering why we did this? Because we want to set the *exact* last language as the *main* language! The current value of our counter at this point has that language. But remember we are out of the loop as well as the if-conditional. Thus we have to again consider the situation where user just passed one language. So far what we have done is load all the languages except the last one with `\babelprovide`. This includes the single-language case. Thus we need no further processing in that case. So we only parse the last pending language if the user had passed multiple languages. We start an if-conditional and keep its if-branch empty and write the rest in the `else`-branch.

```

1200     \ifnum\theudiss@langbabel=1\relax

```

We again use `\babelprovide`, but this time with `main` option. Load the last language successfully.

```

1201     \else
1202         \ifLuaTeX
1203         \babelprovide[%
1204             main,%
1205             import,%
1206             onchar           ={ids fonts}%
1207         ]{%
1208             \csname
1209                 udiss@langbabel@\alph{udiss@langbabel}%
1210             \endcsname
1211         }%
1212     \else
1213         \babelprovide[main,import]{%
1214             \csname
1215                 udiss@langbabel@\alph{udiss@langbabel}%
1216             \endcsname
1217         }%
1218     \fi
1219 \fi

```

Now it's the time to load language support files. We need to load all the language support files and here we don't have the constraint to treat the last language specially. Therefore we add 1 to the temporary counter (the one that was decreased before).

```

1220     \setcounter{udiss@tmp@langcnt}{%
1221         \numexpr\theudiss@langbabel + 1\relax
1222     }%

```



We again start from the first language. If the support file exists for that language, we `\input` it.

```

I223     \setcounter{udiss@langbabel}{1}%
I224     \loop
I225       \IfFileExists{%
I226         udiss-fonts-\csname
I227           udiss@langbabel@\alph{udiss@langbabel}%
I228         \endcsname
I229       }{%
I230         \input{%
I231           udiss-fonts-\csname
I232             udiss@langbabel@\alph{udiss@langbabel}%
I233           \endcsname
I234         }%

```

Otherwise, we check if the warning option is true. By default it is always true, but if the user has used `nofontwarning` option to turn them off, this conditional should do its job.

```

I235     }{%
I236       \ifudiss@font@warning
I237       \ClassWarningNoLine{udiss}{%
I238         We don't support \csname
I239           udiss@langbabel@\alph{udiss@langbabel}%
I240         \endcsname\space yet.%
I241       \MessageBreak
I242       It requires a font-setup. You may add the fonts
I243       directly\MessageBreak to your preamble using:%
I244       \MessageBreak\MessageBreak
I245       \space\space\string\babelfont[%
I246         \csname
I247           udiss@langbabel@\alph{udiss@langbabel}%
I248         \endcsname
I249       ]{rm}{main font}%
I250       \MessageBreak
I251       \space\space\string\babelfont[%
I252         \csname
I253           udiss@langbabel@\alph{udiss@langbabel}%
I254         \endcsname
I255       ]{sf}{sans font}%
I256       \MessageBreak
I257       \space\space\string\babelfont[%
I258         \csname
I259           udiss@langbabel@\alph{udiss@langbabel}%
I260         \endcsname
I261       ]{tt}{mono font}%
I262       \MessageBreak\MessageBreak
I263       Read more about this in the documentation of
I264       `udiss' in a\MessageBreak section called
I265       `multilingual typesetting'. Consider%
I266       \MessageBreak contacting us with a support
I267       request and let us know some\MessageBreak
I268       suitable free (libre) fonts for this
I269       language. If you\MessageBreak don't wish to see
I270       this warning again, use the option%

```

```

I271         \MessageBreak `nofontwarning' to suppress it%
I272     }%
I273     \fi
I274 }%

```

Like before, we increment the counter and start the loop till this is repeated for all the languages.

```

I275     \addtocounter{udiss@langbabel}{1}%
I276     \ifnum\theudiss@langbabel<\theudiss@tmp@langcnt\relax
I277     \repeat
I278     \fi
I279 \fi

```

After all this processing, we load the Roman, Typewriter and Sans fonts stored in the variables set by *keys*.

```

I280 \bafont{rm}[%
I281   \ifudiss@ignorefontspec
I282     IgnoreFontspecFile,%
I283   \fi
I284   \ifudiss@explicittext
I285     Extension           = {.\udiss@rmfntext},%
I286   \fi
I287   BoldFont             = {\udiss@rmbffont},%
I288   ItalicFont           = {\udiss@rmitfont},%
I289   BoldItalicFont       = {\udiss@rmbfitfont},%
I290   SlantedFont          = {\udiss@rmslfont},%
I291   BoldSlantedFont      = {\udiss@rmbfslfont},%
I292   \ifudiss@oldstylenum
I293     Numbers             = {OldStyle},%
I294     \ifudiss@olddone
I295       CharacterVariant = {6},%
I296     \fi
I297   \fi
I298   \ifLuaTeX
I299     Renderer            = {HarfBuzz},%
I300   \fi
I301   \udiss@rmfeatures
I302 ]{\udiss@rmfont}
I303 \bafont{sf}[%
I304   \ifudiss@ignorefontspec
I305     IgnoreFontspecFile,%
I306   \fi
I307   \ifudiss@explicittext
I308     Extension           = {.\udiss@sffntext},%
I309   \fi
I310   BoldFont             = {\udiss@sfbffont},%
I311   ItalicFont           = {\udiss@sfitfont},%
I312   BoldItalicFont       = {\udiss@sfbfitfont},%
I313   SlantedFont          = {\udiss@sfsfont},%
I314   BoldSlantedFont      = {\udiss@sfbfslfont},%
I315   \ifudiss@oldstylenum
I316     Numbers             = {OldStyle},%
I317     \ifudiss@olddone
I318       CharacterVariant = {6},%

```

```

I319 \fi
I320 \fi
I321 \ifLuaTeX
I322   Renderer          = {HarfBuzz},%
I323 \fi
I324 \udiss@sffeatures
I325 ]{\udiss@sffont}
I326 \bafont{tt}[%
I327   \ifudiss@ignorefontspec
I328     IgnoreFontspecFile,%
I329 \fi
I330 \ifudiss@explicittext
I331   Extension          = {.\udiss@ttfntext},%
I332 \fi
I333 BoldFont             = {\udiss@ttbfont},%
I334 ItalicFont           = {\udiss@ttitfont},%
I335 BoldItalicFont       = {\udiss@ttbfitfont},%
I336 SlantedFont          = {\udiss@ttslfont},%
I337 BoldSlantedFont     = {\udiss@ttbfsfont},%
I338 \ifLuaTeX
I339   Renderer          = {HarfBuzz},%
I340 \fi
I341 \udiss@ttfeatures
I342 ]{\udiss@ttfont}

```

We load the unicode-math here for loading the math fonts. After that, inside `\expanded`, we load the `\setmathfont` command.

```

I343 \RequirePackage{unicode-math}
I344 \expanded{%
I345   \setmathfont[%
I346     \ifudiss@ignorefontspec%
I347       IgnoreFontspecFile,%
I348   \fi%
I349   \ifudiss@explicittext
I350     Extension          = {.\udiss@mathfntext},%
I351   \fi
I352   BoldFont             = {\udiss@mathbfont},%
I353   SlantedFont          = {\udiss@mathfont},%
I354   BoldSlantedFont     = {\udiss@mathbfont},%
I355   \udiss@mathfeatures%
I356 ]{\udiss@mathfont}%
I357 }

```

Now we load a few last packages.

```

I358 \RequirePackage{graphicx}
I359 \RequirePackage{xcolor}
I360 \RequirePackage{csquotes}

```

For Creative Commons license, we use the `doclicense` package with appropriate settings.

```

I361 \ifudiss@ccbysa
I362   \RequirePackage[%
I363     hyperxmp          = {false},%
I364     type              = {CC},%

```

```

I365     modifier          = {by-sa},%
I366     version           = {4.0}%
I367   ]{doclicense}
I368 \fi

```

We set the `udiss-specific-colours` with the following commands, but we set them conditionally. If the colours are disabled, we set the same colours to `black` value.

```

I369 \ifudiss@colors
I370   \colorlet{udisslink}{red!60!black}
I371   \colorlet{udissurl}{blue!60!black}
I372   \colorlet{udisscite}{green!60!black}
I373 \else
I374   \colorlet{udisslink}{black}
I375   \colorlet{udissurl}{black}
I376   \colorlet{udisscite}{black}
I377 \fi

```

We start the hook `begindocument/before` and check if author did not provide any of the *strict* fields and if they have done so, we generate errors.

```

I378 \AddToHook{begindocument/before}{%
I379   \ifdefined\udiss@title
I380   \else
I381     \def\udiss@title{\textbf{-NoValidTitle-}}%
I382     \ClassError{udiss}{%
I383       Title not given%
I384     }{%
I385       `title' is a compulsory field. Provide it in the
I386       preamble like in the following:%
I387       \MessageBreak
I388       \MessageBreak
I389       ----8<-----8<-----%
I390       \MessageBreak\MessageBreak
I391       \space\space\string\title{title-of-the-dissertation}%
I392       \MessageBreak\MessageBreak
I393       ----8<-----8<-----%
I394       \MessageBreak
I395     }%
I396   \fi
I397   \ifdefined\udiss@author
I398   \else
I399     \def\udiss@author{\textbf{-NoValidAuthor-}}%
I400     \ClassError{udiss}{%
I401       No author given in a normal (non-blind) document%
I402     }{%
I403       Do you want a blind document? If yes, then use the
I404       package option `blind' like\MessageBreak in the
I405       following:%
I406       \MessageBreak
I407       ----8<-----8<-----%
I408       \MessageBreak\MessageBreak
I409       \space\space\string\documentclass[blind]{udiss}%
I410       \MessageBreak\MessageBreak
I411       ----8<-----8<-----%
I412       \MessageBreak

```

```

I413      Otherwise, `author' is a compulsory field. Provide it
I414      in the preamble like in\MessageBreak the following:%
I415      \MessageBreak
I416      ----8<-----8<-----%
I417      \MessageBreak\MessageBreak
I418      \space\space\string\author{author-name}%
I419      \MessageBreak\MessageBreak
I420      ----8<-----8<-----%
I421      \MessageBreak
I422      }%
I423      \fi
I424      \ifdefined\udiss@stream
I425      \else
I426      \def\udiss@stream{\textbf{-NoValidStream-}}%
I427      \ClassError{udiss}{%
I428      Stream not given%
I429      }{%
I430      `stream' is a compulsory field. Provide it in the
I431      preamble like in the following:%
I432      \MessageBreak
I433      \MessageBreak
I434      ----8<-----8<-----%
I435      \MessageBreak\MessageBreak
I436      \space\space\string\stream{stream-of-the-candidate}%
I437      \MessageBreak\MessageBreak
I438      ----8<-----8<-----%
I439      \MessageBreak
I440      }%
I441      \fi
I442      \ifdefined\udiss@discipline
I443      \else
I444      \def\udiss@discipline{\textbf{-NoValidDiscipline-}}%
I445      \ClassError{udiss}{%
I446      Discipline not given%
I447      }{%
I448      `discipline' is a compulsory field. Provide it in the
I449      preamble like in the following:%
I450      \MessageBreak
I451      \MessageBreak
I452      ----8<-----8<-----%
I453      \MessageBreak\MessageBreak
I454      \space\space\string\discipline%
I455      {discipline-of-the-candidate}%
I456      \MessageBreak\MessageBreak
I457      ----8<-----8<-----%
I458      \MessageBreak
I459      }%
I460      \fi
I461      \ifdefined\udiss@degree
I462      \else
I463      \def\udiss@degree{\textbf{-NoValidDegree-}}%
I464      \ClassError{udiss}{%
I465      Degree not given%
I466      }{%

```

```

1467     `degree' is a compulsory field. Provide it in the
1468     preamble like in the following:%
1469     \MessageBreak
1470     \MessageBreak
1471     ----8<-----8<-----%
1472     \MessageBreak\MessageBreak
1473     \space\space\string\degree{degree-of-the-candidate}%
1474     \MessageBreak\MessageBreak
1475     ----8<-----8<-----%
1476     \MessageBreak
1477     }%
1478     \fi
1479     \ifdefined\udiss@supervisor
1480     \else
1481     \def\udiss@title{\textbf{-NoValidTitle-}}%
1482     \ClassError{udiss}{%
1483     No supervisor given in a normal (non-blind) document%
1484     }{%
1485     Do you want a blind document? If yes, then use the
1486     package option `blind' like\MessageBreak in the
1487     following:%
1488     \MessageBreak
1489     ----8<-----8<-----%
1490     \MessageBreak\MessageBreak
1491     \space\space\string\documentclass[blind]{udiss}%
1492     \MessageBreak\MessageBreak
1493     ----8<-----8<-----%
1494     \MessageBreak
1495     Otherwise, `supervisor' is a compulsory field. Provide
1496     it in the preamble like\MessageBreak in the
1497     following:%
1498     \MessageBreak
1499     ----8<-----8<-----%
1500     \MessageBreak\MessageBreak
1501     \space\space\string\supervisor{supervisor's-name}%
1502     \MessageBreak\MessageBreak
1503     ----8<-----8<-----%
1504     \MessageBreak
1505     }%
1506     \fi
1507     \ifdefined\udiss@university
1508     \else
1509     \def\udiss@university{\textbf{-NoValidUniversity-}}%
1510     \ClassError{udiss}{%
1511     No university given in a normal (non-blind) document%
1512     }{%
1513     Do you want a blind document? If yes, then use the
1514     package option `blind' like\MessageBreak in the
1515     following:%
1516     \MessageBreak
1517     ----8<-----8<-----%
1518     \MessageBreak\MessageBreak
1519     \space\space\string\documentclass[blind]{udiss}%
1520     \MessageBreak\MessageBreak

```

```

I521     ----8<-----8<----%
I522     \MessageBreak
I523     Otherwise, `university' is a compulsory field. Provide
I524     it in the preamble like\MessageBreak in the
I525     following:%
I526     \MessageBreak
I527     ----8<-----8<----%
I528     \MessageBreak\MessageBreak
I529     \space\space\string\university{university-name}%
I530     \MessageBreak\MessageBreak
I531     ----8<-----8<----%
I532     \MessageBreak
I533     }%
I534     \fi
I535     \ifdefined\udiss@department
I536     \else
I537     \def\udiss@department{\textbf{-NoValidDepartment-}}%
I538     \ClassError{udiss}{%
I539     No department given in a normal (non-blind) document%
I540     }{%
I541     Do you want a blind document? If yes, then use the
I542     package option `blind' like\MessageBreak in the
I543     following:%
I544     \MessageBreak
I545     ----8<-----8<----%
I546     \MessageBreak\MessageBreak
I547     \space\space\string\documentclass[blind]{udiss}%
I548     \MessageBreak\MessageBreak
I549     ----8<-----8<----%
I550     \MessageBreak
I551     Otherwise, `department'\space is a compulsory field.
I552     Provide it in the preamble like in\MessageBreak the
I553     following:%
I554     \MessageBreak
I555     ----8<-----8<----%
I556     \MessageBreak\MessageBreak
I557     \space\space\string\department{department-name}%
I558     \MessageBreak\MessageBreak
I559     ----8<-----8<----%
I560     \MessageBreak
I561     }%
I562     \fi

```

For printing, hyperref and hyperxmp are irrelevant, so we load them only when the documents are digital.

```

I563     \ifudiss@print
I564     \else
I565     \RequirePackage{hyperref}%
I566     \RequirePackage{hyperxmp}%

```

We set the metadata here using \hypersetup command.

```

I567     \hypersetup{%
I568     unicode,%
I569     colorlinks,%

```

```

1570     pdftitle           = {\udiss@title}%
1571   }%
1572   \ifudiss@subtitle
1573     \hypersetup{%
1574       pdfsubject       = {\udiss@subtitle}%
1575     }%
1576   \fi

```

We don't set author's name if the user has used the `blind` option as it defeats the whole purpose. If it is not used, we set the copyright information and the license-url. All-rights-reserved doesn't have a specific license-url, so it is used only if the user uses a free culture license.

```

1577   \ifudiss@blind
1578   \else
1579     \hypersetup{%
1580       pdfauthor         = {\udiss@author},%
1581       pdfcopyright      = {%
1582         \udiss@title\textLF
1583         \udiss@local@copyright\
1584         ©
1585         \udiss@copyrightable@years\
1586         \udiss@author\textLF
1587         \udiss@license@notice
1588       },%
1589     }%
1590   \ifudiss@free@culture
1591     \hypersetup{%
1592       pdflicenseurl     = {%
1593         \udiss@free@license@url
1594       },%
1595     }%
1596   \fi
1597 \fi

```

If the `color` option is not disabled, we load the color variables with `\hypersetup`.

```

1598   \ifudiss@colors
1599     \hypersetup{%
1600       linkcolor         = {\udiss@int@link@color},%
1601       urlcolor          = {\udiss@int@url@color},%
1602       citecolor         = {\udiss@int@cite@color}%
1603     }%

```

Otherwise, we use the `hidelinks` option of `hyperref` which creates links, but without colors.

```

1604   \else
1605     \hypersetup{%
1606       hidelinks%
1607     }%
1608   \fi
1609 \fi
1610 }

```

Now, after setting everything, we typeset the front matter including title-page, table of contents, optionally list of figures/tables and declaration with the following code.



```

r611 \AddToHook{begindocument/end}{%
r612   \begingroup
r613   \thispagestyle{titlingpage}%
r614   \ifudiss@ftitle
r615     \setlength{\fboxsep}{25pt}%
r616     \noindent\fbox{%
r617       \begin{minipage}%
r618         [c][\dimexpr\textheight-2\fboxsep-2\fboxrule]%
r619         [c]{\dimexpr\linewidth-2\fboxsep-2\fboxrule}%
r620       \fi
r621         \hspace{0pt}%
r622         \vspace{3em}%
r623         \begin{center}
r624           \begingroup
r625             \udiss@title@shape
r626             \ifx\udiss@title@font\empty
r627             \else
r628               \fontspec[\udiss@title@fontfeats]%
r629                 {\udiss@title@font}%
r630             \fi
r631             \udiss@title@size
r632             \color{\udiss@title@color}%
r633             \udiss@title
r634           \endgroup
r635
r636           \medskip
r637
r638           \ifudiss@subtitle
r639             \begingroup
r640               \udiss@subtitle@shape
r641               \ifx\udiss@subtitle@font\empty
r642               \else
r643                 \fontspec[\udiss@subtitle@fontfeats]%
r644                   {\udiss@subtitle@font}%
r645               \fi
r646               \udiss@subtitle@size
r647               \color{\udiss@subtitle@color}%
r648               \udiss@subtitle
r649             \endgroup
r650           \fi
r651
r652           \vfill
r653
r654           \ifudiss@blind
r655           \else
r656             \begingroup
r657               \udiss@author@shape
r658               \ifx\udiss@author@font\empty
r659               \else
r660                 \fontspec[\udiss@author@fontfeats]%
r661                   {\udiss@author@font}%
r662               \fi
r663               \udiss@author@size
r664               \color{\udiss@author@color}%

```

```

I665         \udiss@author
I666         \ifudiss@pronouns
I667         \begingroup
I668         \space
I669         \addfontfeature{Scale={0.6}}%
I670         \udiss@pronouns
I671         \endgroup
I672         \fi
I673         \endgroup
I674
I675         \vspace{2em}%
I676         \fi
I677
I678         \begingroup
I679         \udiss@fulfilment@shape
I680         \ifx\udiss@fulfilment@font\empty
I681         \else
I682         \fontspec[\udiss@fulfilment@fontfeats]%
I683             {\udiss@fulfilment@font}%
I684         \fi
I685         \udiss@fulfilment@size
I686         \color{\udiss@fulfilment@color}%
I687         \udiss@fulfilment
I688         \endgroup
I689
I690         \vfill
I691
I692         \begingroup
I693         \udiss@uni@shape
I694         \ifx\udiss@uni@font\empty
I695         \else
I696         \fontspec[\udiss@university@fontfeats]%
I697             {\udiss@uni@font}%
I698         \fi
I699         \udiss@uni@size
I700         \color{\udiss@uni@color}%
I701         \udiss@university
I702         \endgroup
I703
I704         \vspace{0.5em}%
I705
I706         \begingroup
I707         \udiss@dept@shape
I708         \ifx\udiss@dept@font\empty
I709         \else
I710         \fontspec[\udiss@dept@fontfeats]%
I711             {\udiss@dept@font}%
I712         \fi
I713         \udiss@dept@size
I714         \color{\udiss@dept@color}%
I715         \udiss@department
I716         \endgroup
I717
I718         \ifudiss@logo

```

```

I719         \bigskip
I720         \ifudiss@texlogo
I721           \udiss@texlogo
I722         \fi
I723         \includegraphics[%
I724           width      = {%
I725             \udiss@logo@width\linewidth
I726           },%
I727           height     = {%
I728             \udiss@logo@hght\textheight
I729           }%
I730         ]{%
I731           \udiss@logo
I732         }%
I733         \fi
I734       \end{center}
I735 \ifudiss@ftitle
I736   \end{minipage}%
I737 }%
I738 \fi
I739
I740 \newpage
I741
I742 \ifudiss@blind
I743 \else
I744   \ifudiss@free@culture
I745     \thispagestyle{titlingpage}%
I746     \vspace*{\stretch{1}}
I747     \begin{quote}
I748       \setlength{\parindent}{0pt}%
I749       \begin{group}
I750         \begin{minipage}{\linewidth}
I751           \bfseries
I752           \udiss@title
I753
I754           \udiss@local@copyright\
I755           ©
I756           \udiss@copyrightable@years\
I757           \udiss@author
I758
I759           \medskip
I760           \noindent\hrule \linewidth 0.4pt\relax
I761         \end{minipage}
I762       \end{group}
I763
I764   \ifudiss@gfdl
I765     Permission is granted to copy, distribute and/or
I766     modify this document under the terms of the GNU
I767     Free Documentation License, Version 1.3 or any
I768     later version published by the Free Software
I769     Foundation; with no Invariant Sections, no
I770     Front-Cover Texts, and no Back-Cover Texts. A copy
I771     of the license is included in the section entitled
I772     \enquote{GNU Free Documentation License}.\%

```

```

1773     \fi
1774     \ifudiss@ccbysa
1775         \begin{minipage}{\linewidth}
1776             \doclicenseThis
1777         \end{minipage}
1778     \fi
1779     \vspace{\stretch{3}}
1780 \end{quote}
1781 \newpage
1782 \fi
1783 \fi
1784
1785 \tableofcontents*
1786
1787 \newpage
1788
1789 \ifudiss@blind
1790 \else
1791     \ifudiss@declaration
1792         \thispagestyle{empty}%
1793         \begin{center}%
1794             {%
1795                 \LARGE\bfseries
1796                 \udiss@decllocal
1797             }%
1798         \end{center}%
1799
1800     \udiss@declaration
1801 \fi
1802 \fi
1803
1804 \newpage
1805
1806 \ifudiss@lot
1807     \listoftables
1808 \fi
1809
1810 \newpage
1811
1812 \ifudiss@lof
1813     \listoffigures
1814 \fi
1815 \endgroup
1816 }%

```

The following code adds the GFDL v1.3+ notice at the end of the document if that option was used.

```

1817 \ifudiss@blind
1818 \else
1819     \ifudiss@gfdl
1820         \AddToHook{enddocument}{%
1821             \newpage
1822             \begin{group
1823                 \pagestyle{plain}%

```

```

1824     \addcontentsline{toc}%
1825                 {chapter}%
1826                 {GNU Free Documentation License}%
1827     \input{gfdl-tex-1p3}%
1828     \clearpage
1829     \endgroup
1830 }%
1831 \fi
1832 \fi
1833 </class>

```

Our class ended here. Now we start creating the language-support files. The first is the Malayalam language. We use fonts provided at: <https://ctan.org/pkg/rit-fonts>. Notice that this is done conditionally. If the language is loaded as the main language, the fonts are set as the default fonts using class options, else `malayalam` is passed as the optional argument to `\babelfont`.

```

1834 (*babel-ml)
1835 \IfLanguageName{malayalam}{%
1836     \ekvset{udiss}{%
1837         rfont           = {RIT-Rachana-Regular.ttf},%
1838         rmitfont        = {RIT-Rachana-Italic.ttf},%
1839         rmbffont        = {RIT-Rachana-Bold.ttf},%
1840         rmbfitfont      = {RIT-Rachana-BoldItalic.ttf},%
1841         rmfeatures     = {%
1842             Script      = {Malayalam}%
1843         },%
1844         sffont          = {RIT-MeeraNew.ttf},%
1845         sfitfont        = {RIT-MeeraNew.ttf},%
1846         sfbffont        = {RIT-tjoy-bold.ttf},%
1847         sfbfitfont      = {RIT-tjoy-bold.ttf},%
1848         sffeatures     = {%
1849             Script      = {Malayalam}%
1850         },%
1851         ttfont          = {RIT-MeeraNew.ttf},%
1852         ttitfont        = {RIT-MeeraNew.ttf},%
1853         ttbfont         = {RIT-tjoy-bold.ttf},%
1854         ttbfitfont      = {RIT-tjoy-bold.ttf},%
1855         sffeatures     = {%
1856             Script      = {Malayalam}%
1857         },%
1858     }%
1859 }{%
1860     \babelfont[malayalam]{rm}{%
1861         IgnoreFontspecFile,%
1862         Script          = {Malayalam},%
1863         ItalicFont      = {RIT-Rachana-Italic.ttf},%
1864         BoldFont        = {RIT-Rachana-Bold.ttf},%
1865         BoldItalicFont  = {RIT-Rachana-BoldItalic.ttf},%
1866         \ifLuaTeX
1867             Renderer    = {HarfBuzz}%
1868         \fi
1869     ]{RIT-Rachana-Regular.ttf}
1870     \babelfont[malayalam]{sf}{%
1871         IgnoreFontspecFile,%

```

```

1872 Script = {Malayalam},%
1873 ItalicFont = {RIT-tnjoy-regular.ttf},%
1874 BoldFont = {RIT-tnjoy-bold.ttf},%
1875 BoldItalicFont = {RIT-tnjoy-bold.ttf},%
1876 \ifLuaTeX
1877 Renderer = {HarfBuzz}%
1878 \fi
1879 ]{RIT-tnjoy-regular.ttf}
1880 \babelfont[malayalam]{tt}[%
1881 IgnoreFontspecFile,%
1882 Script = {Malayalam},%
1883 ItalicFont = {RIT-tnjoy-regular.ttf},%
1884 BoldFont = {RIT-tnjoy-bold.ttf},%
1885 BoldItalicFont = {RIT-tnjoy-bold.ttf},%
1886 \ifLuaTeX
1887 Renderer = {HarfBuzz}%
1888 \fi
1889 ]{RIT-tnjoy-regular.ttf}%
1890 }
1891 </babel-ml>

```

Marathi requires local numerals and local counters. We provide that here and set Mukta as the default font provided in: <https://ctan.org/pkg/EkType-Tanka>.

```

1892 (*babel-mr)
1893 \babelprovide[%
1894 mapdigits,%
1895 counters/स्वर = अ आ इ ई उ ऊ ए ऐ ओ औ अं अः अँ ऋ लृ ऑं,%
1896 counters/अंक = एक दोन तीन चार पाच सहा सात आठ नऊ दहा अकरा
1897 बारा तेरा चौदा पंधरा सोळा सतरा अठरा एकोणीस वीस एकवीस बावीस
1898 तेवीस चोवीस पंचवीस सव्वीस सत्तावीस अठ्ठावीस एकोणतीस तीस एकतीस
1899 बत्तीस तेहतीस चौतीस पस्तीस छत्तीस सदतीस अडतीस एकोणचाळीस चाळीस
1900 एकेचाळीस बेचाळीस त्रेचाळीस चव्वेचाळीस पंचेचाळीस शेहेचाळीस सत्तेचाळीस
1901 अठ्ठेचाळीस एकोणपन्नास पन्नास एकावन्न बावन्न त्रेपन्न चौपन्न पंचावन्न
1902 छप्पन्न सत्तावन्न अठ्ठावन्न एकोणसाठ साठ एकसष्ट बासष्ट त्रेसष्ट चौसष्ट
1903 पासष्ट सहासष्ट सदुष्ट अडुसष्ट एकोणसत्तर सत्तर एकाहत्तर बाहत्तर
1904 त्र्याहत्तर चौऱ्याहत्तर पंचाहत्तर शाहत्तर सत्याहत्तर अठ्ठ्याहत्तर
1905 एकोणऐंशी ऐंशी एक्याऐंशी ब्याऐंशी त्र्याऐंशी चौऱ्याऐंशी पंच्याऐंशी श्याऐंशी
1906 सत्याऐंशी अठ्ठ्याऐंशी एकोणनव्वद नव्वद एक्याण्णव ब्याण्णव त्र्याण्णव
1907 चौऱ्याण्णव पंचाण्णव शहाण्णव सत्याण्णव अठ्ठ्याण्णव नव्याण्णव शंभर,%
1908 counters/व्यंजन = क ख ग घ ङ%
1909 च छ ज झ ञ%
1910 ट ठ ड ढ ण%
1911 त थ द ध न%
1912 प फ ब भ म%
1913 य र ल व श ष स ह ळ,%
1914 Alph = अंक,%
1915 alph = स्वर%
1916 ]{marathi}
1917 \def\theenumiii{\localecounter{व्यंजन}{enumiii}}
1918 \def\theenumiv{\localecounter{अंक}{enumiii}}
1919 \IfLanguageName{marathi}{%
1920 \ekvset{udiss}{%
1921 rmfont = {Mukta-Regular.ttf},%
1922 rmitfont = {Mukta-Regular.ttf},%

```

```

1923     rmbffont           = {Mukta-Bold.ttf},%
1924     rmbfitfont        = {Mukta-Bold.ttf},%
1925     rmfeatures        = {%
1926         Script         = {Devanagari}%
1927     },%
1928     sffont             = {Mukta-Regular.ttf},%
1929     sffitfont         = {Mukta-Regular.ttf},%
1930     sfbffont          = {Mukta-Bold.ttf},%
1931     sfbfitfont        = {Mukta-Bold.ttf},%
1932     sffeatures        = {%
1933         Script         = {Devanagari}%
1934     },%
1935     ttfont            = {Mukta-Regular.ttf},%
1936     ttitfont          = {Mukta-Regular.ttf},%
1937     ttbfont           = {Mukta-Bold.ttf},%
1938     ttbfitfont        = {Mukta-Bold.ttf},%
1939     sffeatures        = {%
1940         Script         = {Devanagari}%
1941     }%
1942 }%
1943 }-%
1944 \babelfont[marathi]{rm}[%
1945     IgnoreFontspecFile,%
1946     ItalicFont         = {Mukta-Regular.ttf},%
1947     BoldFont           = {Mukta-Bold.ttf},%
1948     BoldItalicFont     = {Mukta-Bold.ttf},%
1949     Language           = {Marathi},%
1950     Script             = {Devanagari},%
1951     Numbers            = {OldStyle},%
1952     \ifLuaTeX
1953     Renderer           = {HarfBuzz}%
1954     \fi
1955 ]{Mukta-Regular.ttf}
1956 \babelfont[marathi]{sf}[%
1957     IgnoreFontspecFile,%
1958     ItalicFont         = {Mukta-Regular.ttf},%
1959     BoldFont           = {Mukta-Bold.ttf},%
1960     BoldItalicFont     = {Mukta-Bold.ttf},%
1961     Language           = {Marathi},%
1962     Script             = {Devanagari},%
1963     Numbers            = {OldStyle},%
1964     \ifLuaTeX
1965     Renderer           = {HarfBuzz}%
1966     \fi
1967 ]{Mukta-Regular.ttf}
1968 \babelfont[marathi]{tt}[%
1969     IgnoreFontspecFile,%
1970     ItalicFont         = {Mukta-Regular.ttf},%
1971     BoldFont           = {Mukta-Bold.ttf},%
1972     BoldItalicFont     = {Mukta-Bold.ttf},%
1973     Language           = {Marathi},%
1974     Script             = {Devanagari},%
1975     Numbers            = {OldStyle},%
1976     \ifLuaTeX

```

```

1977     Renderer                = {HarfBuzz}%
1978     \fi
1979   ]{Mukta-Regular.ttf}%
1980 }
1981 </babel-mr>
1982 <*babel-en>
1983 \bafont[english]{rm}[%
1984   IgnoreFontspecFile,%
1985   Extension                = {.otf},%
1986   ItalicFont                = {NewCM10-BookItalic},%
1987   BoldFont                  = {NewCM10-Bold},%
1988   BoldItalicFont            = {NewCM10-BoldItalic},%
1989   SlantedFont               = {NewCM10-Book},%
1990   BoldSlantedFont           = {NewCM10-Bold},%
1991   Numbers                   = {OldStyle},%
1992   CharacterVariant          = {6},%
1993   \ifLuaTeX
1994     Renderer                = {HarfBuzz}%
1995     \fi
1996 ]{NewCM10-Book}
1997 \bafont[english]{sf}[%
1998   IgnoreFontspecFile,%
1999   Extension                = {.otf},%
2000   ItalicFont                = {NewCMSans10-BookOblique},%
2001   BoldFont                  = {NewCMSans10-Bold},%
2002   BoldItalicFont            = {NewCMSans10-BoldOblique},%
2003   SlantedFont               = {NewCMSans10-Book},%
2004   BoldSlantedFont           = {NewCMSans10-Bold},%
2005   Numbers                   = {OldStyle},%
2006   CharacterVariant          = {6},%
2007   \ifLuaTeX
2008     Renderer                = {HarfBuzz}%
2009     \fi
2010 ]{NewCMSans10-Book}
2011 \bafont[english]{tt}[%
2012   IgnoreFontspecFile,%
2013   Extension                = {.otf},%
2014   BoldFont                  = {NewCMMono10-Bold},%
2015   ItalicFont                = {NewCMMono10-Italic},%
2016   \ifLuaTeX
2017     Renderer                = {HarfBuzz}%
2018     \fi
2019 ]{NewCMMono10-Book}
2020 </babel-en>

```

Hindi and Marathi share the script, but the language variants Marathi needs are not required by Hindi. So we use the same font, but with a little different settings for Hindi.

```

2021 <*babel-hi>
2022 \IfLanguageName{hindi}{%
2023   \ekvset{udiss}{%
2024     rmfont                = {Mukta-Regular.ttf},%
2025     rmitfont              = {Mukta-Regular.ttf},%
2026     rmbffont              = {Mukta-Bold.ttf},%
2027     rmbfitfont            = {Mukta-Bold.ttf},%

```



```

2028     rmfeatures           = {%
2029     Script               = {Devanagari}%
2030     },%
2031     sffont               = {Mukta-Regular.ttf},%
2032     sffitfont            = {Mukta-Regular.ttf},%
2033     sfbffont             = {Mukta-Bold.ttf},%
2034     sfbffitfont         = {Mukta-Bold.ttf},%
2035     sffeatures           = {%
2036     Script               = {Devanagari}%
2037     },%
2038     ttfont               = {Mukta-Regular.ttf},%
2039     ttitfont             = {Mukta-Regular.ttf},%
2040     ttbfont              = {Mukta-Bold.ttf},%
2041     ttbffitfont         = {Mukta-Bold.ttf},%
2042     sffeatures           = {%
2043     Script               = {Devanagari}%
2044     }%
2045 }%
2046 }{%
2047 \belfont[hindi]{rm}[%
2048   IgnoreFontspecFile,%
2049   ItalicFont             = {Mukta-Regular.ttf},%
2050   BoldFont               = {Mukta-Bold.ttf},%
2051   BoldItalicFont         = {Mukta-Bold.ttf},%
2052   Language                = {Hindi},%
2053   Script                  = {Devanagari},%
2054   Numbers                 = {OldStyle},%
2055   \ifLuaTeX
2056     Renderer              = {HarfBuzz}%
2057   \fi
2058 ]{Mukta-Regular.ttf}
2059 \belfont[hindi]{sf}[%
2060   IgnoreFontspecFile,%
2061   ItalicFont             = {Mukta-Regular.ttf},%
2062   BoldFont               = {Mukta-Bold.ttf},%
2063   BoldItalicFont         = {Mukta-Bold.ttf},%
2064   Language                = {Hindi},%
2065   Script                  = {Devanagari},%
2066   Numbers                 = {OldStyle},%
2067   \ifLuaTeX
2068     Renderer              = {HarfBuzz}%
2069   \fi
2070 ]{Mukta-Regular.ttf}
2071 \belfont[hindi]{tt}[%
2072   IgnoreFontspecFile,%
2073   ItalicFont             = {Mukta-Regular.ttf},%
2074   BoldFont               = {Mukta-Bold.ttf},%
2075   BoldItalicFont         = {Mukta-Bold.ttf},%
2076   Language                = {Hindi},%
2077   Script                  = {Devanagari},%
2078   Numbers                 = {OldStyle},%
2079   \ifLuaTeX
2080     Renderer              = {HarfBuzz}%
2081   \fi

```

```

2082   ]{Mukta-Regular.ttf}
2083   }
2084 </babel-hi)

```

In this tag, we set three main variables for the `ukerala` dissertation-style. This is just a sample. These style files can contain much more than this. In the experimental release, we start with a small sample. We will be expanding this more.

```

2085 <*ukerala)
2086 \ekvset{udiss}{%
2087   university           = {University of Kerala},%
2088   logo                 = {udiss-logo-ukerala.png},%
2089   logoheight          = {0.11}%
2090 }
2091 </ukerala)

```

## Index

The italic numbers denote the pages where the corresponding entry is described, numbers underlined point to the definition, all others indicate the places where it is used.

	<b>Symbols</b>	
<code>\l</code> .....	<i>413, 471, 1583, 1585, 1754, 1756</i>	<code>deptshape</code> .....
	<b>A</b>	<i>10, 1045</i>
<code>\author</code> .....	<i>3, 1084, 1418</i>	<code>deptsize</code> .....
<code>author</code> .....	<i>22</i>	<i>10, 1045</i>
<code>authorcolor</code> .....	<i>10, 1024</i>	<code>\discipline</code> .....
<code>authorffeat</code> .....	<i>10, 1024</i>	<i>3, 90, 1454</i>
<code>authorfont</code> .....	<i>10, 1024</i>	<code>discipline</code> .....
<code>authorshape</code> .....	<i>10, 1024</i>	<i>90</i>
<code>authorsize</code> .....	<i>10, 1024</i>	<code>\dissertationstyle</code> .....
	<b>B</b>	<i>7, 1074</i>
<code>blind</code> .....	<i>5, 453</i>	<code>\doclicenseThis</code> .....
	<b>C</b>	<i>1776</i>
<code>colorprint</code> .....	<i>6</i>	<b>E</b>
<code>copyright-yrs</code> .....	<i>5, 334</i>	<code>explicittext</code> .....
	<b>D</b>	<i>8, 463</i>
<code>declaration</code> .....	<i>5, 402</i>	<b>F</b>
<code>declarationtxt</code> .....	<i>5, 402</i>	<code>ffcolor</code> .....
<code>decllocal</code> .....	<i>5, 402</i>	<i>10, 1056</i>
<code>\degree</code> .....	<i>3, 90, 1473</i>	<code>fffeat</code> .....
<code>degree</code> .....	<i>90</i>	<i>10</i>
<code>\department</code> .....	<i>3, 188, 1557</i>	<code>ffffeat</code> .....
<code>department</code> .....	<i>188</i>	<i>1056</i>
<code>deptcolor</code> .....	<i>10, 1045</i>	<code>fffont</code> .....
<code>deptffeat</code> .....	<i>10, 1045</i>	<i>10, 1056</i>
<code>deptfont</code> .....	<i>10, 1045</i>	<code>ffshape</code> .....
		<i>10, 1056</i>
		<code>ffsize</code> .....
		<i>10, 1056</i>
		<code>framed-title</code> .....
		<i>5, 456</i>
		<code>fulfilment</code> .....
		<i>7, 467</i>
		<b>I</b>
		<code>\ifdefined</code> .....
		<i>1379,</i>
		<i>1397, 1424, 1442, 1461, 1479, 1507, 1535</i>
		<code>ignorefsoff</code> .....
		<i>8, 463</i>
		<b>L</b>
		<code>language</code> .....
		<i>4, 320</i>
		<code>ldfbabel</code> .....
		<i>4, 320</i>
		<code>license</code> .....
		<i>5, 334</i>
		<code>localcopyright</code> .....
		<i>5, 334</i>

lof	5, <a href="#">402</a>	\stream	3, <a href="#">90</a> , <a href="#">1436</a>
\logo	3, <a href="#">286</a>	stream	<a href="#">90</a>
logo	<a href="#">297</a>	\subtitle	3, <a href="#">60</a>
\logo*	3, <a href="#">286</a>	subtitle	<a href="#">60</a>
logo*	<a href="#">297</a>	subtitlecolor	10, <a href="#">1013</a>
logoheight	9, <a href="#">1067</a>	subtitledfont	10, <a href="#">1013</a>
logowidth	9, <a href="#">1067</a>	subtitledfont	10, <a href="#">1013</a>
lot	5, <a href="#">402</a>	subtitledshape	10, <a href="#">1013</a>
<b>M</b>			
mathbffont	9, <a href="#">512</a>	subtitledsize	10, <a href="#">1013</a>
mathfeatures	8, <a href="#">516</a>	\supervisor	3, <a href="#">188</a> , <a href="#">1501</a>
mathfntext	8, <a href="#">516</a>	supervisor	<a href="#">188</a>
mathfont	9, <a href="#">512</a>	<b>T</b>	
<b>N</b>			
nofontwarning	4, <a href="#">320</a>	TeX and L <sup>A</sup> T <sub>E</sub> X <sub>2<math>\epsilon</math></sub> commands:	
norefcolors	6, <a href="#">443</a>	\ifudiss@author	31
<b>O</b>			
oldone	7, <a href="#">459</a>	\ifudiss@blind	6,
oldstylenumoff	7, <a href="#">459</a>	8, 15, 454, 1577, 1654, 1742, 1789, 1817	
<b>P</b>			
placeholders	5, <a href="#">379</a>	\ifudiss@ccbysa	335, 1361, 1774
print	6, <a href="#">443</a>	\ifudiss@colors	445, 1369, 1598
\pronouns	3, <a href="#">5</a>	\ifudiss@declaration	405, 1791
pronouns	<a href="#">5</a>	\ifudiss@degree	108
<b>R</b>			
rmbffont	9, <a href="#">475</a>	\ifudiss@department	190
rmbfitfont	9, <a href="#">475</a>	\ifudiss@discipline	99
rmbfslfont	9, <a href="#">475</a>	\ifudiss@explicittext	
rmfeatures	8, <a href="#">516</a>	.... 464, 519, 576, 640, 697, 761,	
rmfntext	8, <a href="#">516</a>	818, 882, 942, 1284, 1307, 1330, 1349	
rmfont	9, <a href="#">475</a>	\ifudiss@font@warning	332, 1236
rmitfont	9, <a href="#">475</a>	\ifudiss@free@culture	334, 1590, 1744
rmslfont	9, <a href="#">475</a>	\ifudiss@ftitle	457, 1614, 1735
<b>S</b>			
\setcounter	1171, 1174, 1220, 1223	\ifudiss@gfdl	336, 1764, 1819
sfbffont	9, <a href="#">488</a>	\ifudiss@ignorefontspec	
sfbfitfont	9, <a href="#">488</a>	.... 465, 1281, 1304, 1327, 1346	
sfbfslfont	9, <a href="#">488</a>	\ifudiss@ldfbabel	321, 1136, 1147
sffeatures	8, <a href="#">516</a>	\ifudiss@lg@used	320, 1135, 1149
sffntext	8, <a href="#">516</a>	\ifudiss@lof	404, 1812
sffont	9, <a href="#">488</a>	\ifudiss@logo	286, 1718
sfitfont	9, <a href="#">488</a>	\ifudiss@lot	403, 1806
sfsfont	9, <a href="#">488</a>	\ifudiss@oldone	461, 1294, 1317
\shorttitle	3, <a href="#">60</a>	\ifudiss@oldstylenum	460, 1292, 1315
shorttitle	<a href="#">60</a>	\ifudiss@print	444, 1079, 1563
		\ifudiss@pronouns	5, 1666
		\ifudiss@shorttitle	61
		\ifudiss@stream	90
		\ifudiss@subtitle	60, 1572, 1638
		\ifudiss@supervisor	188
		\ifudiss@texlogo	287, 1720
		\ifudiss@title	22
		\ifudiss@university	189
		\theudiss@langbabel	
		1150, 1170, 1172, 1196, 1200, 1221, 1276	
		\theudiss@tmp@langcnt	1192, 1196, 1276

<code>\udiss@author</code> .....	III
..... 53, 56, 261, 275, 428, 1095, 1097, 1397, 1399, 1580, 1586, 1665, 1757	
<code>\udiss@author@color</code> .....	III4
..... 1030, 1664	
<code>\udiss@author@font</code> ..	III7
..... 1027, 1658, 1661	
<code>\udiss@author@fontfeats</code> ..	III2
..... 1032, 1660	
<code>\udiss@author@shape</code> .....	III6
..... 1025, 1657	
<code>\udiss@author@size</code> .....	III29
..... 1028, 1663	
<code>\udiss@blank@author@err</code> .	III29
..... 32, 54, 1094	
<code>\udiss@blank@degree@err</code> .	III8
..... 109, 140, 176	
<code>\udiss@blank@department@err</code> .....	III30
..... 207, 238, 274	
<code>\udiss@blank@discipline@err</code> .....	III30
..... 100, 131, 162	
<code>\udiss@blank@stream@err</code> .	III30
..... 91, 122, 148	
<code>\udiss@blank@supervisor@err</code> .....	III30
..... 191, 220, 246	
<code>\udiss@blank@title@err</code> ..	III28
..... 23, 45, 1086	
<code>\udiss@blank@university@err</code> .....	III25
..... 199, 229, 260	
<code>\udiss@ccbysattrue</code> .....	III25
..... 354	
<code>\udiss@copyrightable@years</code> .....	III25
..... 375, 1585, 1756	
<code>\udiss@declaration</code> .....	III25
..... 407, 1800	
<code>\udiss@decllocal</code> .....	III25
..... 440, 1796	
<code>\udiss@degree</code> .....	III25
..... 139, 142, 177, 179, 184, 471, 1461, 1463	
<code>\udiss@degreetrue</code> .....	III25
..... 182, 185	
<code>\udiss@department</code> .....	III25
..... 237, 240, 277, 282, 414, 1535, 1537, 1715	
<code>\udiss@departmenttrue</code> .....	III25
..... 280, 283	
<code>\udiss@dept@color</code> .....	III25
..... 1051, 1714	
<code>\udiss@dept@font</code> ....	III25
..... 1048, 1708, 1711	
<code>\udiss@dept@fontfeats</code> ....	III25
..... 1053, 1710	
<code>\udiss@dept@shape</code> .....	III25
..... 1046, 1707	
<code>\udiss@dept@size</code> .....	III25
..... 1049, 1713	
<code>\udiss@discipline</code> .....	III25
..... 130, 133, 163, 165, 170, 472, 1442, 1444	
<code>\udiss@disciplinetrue</code> .....	III25
..... 168, 171	
<code>\udiss@free@culturetrue</code> ...	III25
..... 353, 370	
<code>\udiss@free@license@url</code> .	III25
..... 349, 367, 1593	
<code>\udiss@fulfilment</code> .....	III25
..... 468, 1687	
<code>\udiss@fulfilment@color</code> ..	III25
..... 1062, 1686	
<code>\udiss@fulfilment@font</code> .	III25
..... 1059, 1680, 1683	
<code>\udiss@fulfilment@fontfeats</code> .	III25
..... 1064, 1682	
<code>\udiss@fulfilment@shape</code> ..	III25
..... 1057, 1679	
<code>\udiss@fulfilment@size</code> ...	III25
..... 1060, 1685	
<code>\udiss@gfdltrue</code> .....	III25
..... 371	
<code>\udiss@int@cite@color</code> .....	III25
..... 450, 1602	
<code>\udiss@int@footnotesize</code> .....	III25
..... 1123	
<code>\udiss@int@HUGE</code> .....	III25
..... 1102	
<code>\udiss@int@Huge</code> .....	III25
..... 1105	
<code>\udiss@int@huge</code> .....	III25
..... 1007, 1108	
<code>\udiss@int@langbabel</code> .....	III25
..... 310, 318	
<code>\udiss@int@LARGE</code> .....	III25
..... 1111	
<code>\udiss@int@Large</code> .....	III25
..... 1114	
<code>\udiss@int@large</code> ....	III25
..... 1018, 1029, 1117	
<code>\udiss@int@link@color</code> .....	III25
..... 446, 1600	
<code>\udiss@int@miniscule</code> .....	III25
..... 1132	
<code>\udiss@int@scriptsize</code> .....	III25
..... 1126	
<code>\udiss@int@small</code> ....	III25
..... 1050, 1061, 1120	
<code>\udiss@int@tiny</code> .....	III25
..... 1129	
<code>\udiss@int@url@color</code> .....	III25
..... 448, 1601	
<code>\udiss@ldfbabel@list</code> .....	III25
..... 328, 1138	
<code>\udiss@ldfbabeltrue</code> .....	III25
..... 330	
<code>\udiss@lg@usedtrue</code> .....	III25
..... 325, 329	
<code>\udiss@license@notice</code> .....	III25
..... 340, 345, 357, 1587	
<code>\udiss@local@copyright</code> .	III25
..... 377, 1583, 1754	
<code>\udiss@logo</code> .....	III25
..... 293, 299, 1731	
<code>\udiss@logo@hght</code> .....	III25
..... 1070, 1728	
<code>\udiss@logo@width</code> .....	III25
..... 1068, 1725	
<code>\udiss@logotrue</code> .....	III25
..... 295, 300, 304	
<code>\udiss@mathbfont</code> ....	III25
..... 514, 1352, 1354	
<code>\udiss@mathfeatures</code> .....	III25
..... 879, 1355	
<code>\udiss@mathfntext</code> ....	III25
..... 883, 943, 1350	
<code>\udiss@mathfont</code> .....	III25
..... 512, 1353, 1356	
<code>\udiss@pronouns</code> .....	III25
..... 10, 17, 1670	
<code>\udiss@pronounstrue</code> .....	III25
..... 19	
<code>\udiss@rmbfont</code> .....	III25
..... 478, 1287	
<code>\udiss@rmbfitfont</code> .....	III25
..... 482, 1289	
<code>\udiss@rmbfslfont</code> .....	III25
..... 486, 1291	
<code>\udiss@rmfeatures</code> .....	III25
..... 516, 1301	
<code>\udiss@rmfntext</code> .....	III25
..... 520, 577, 1285	
<code>\udiss@rmfont</code> .....	III25
..... 476, 1302	
<code>\udiss@rmitfont</code> .....	III25
..... 480, 1288	
<code>\udiss@rmslfont</code> .....	III25
..... 484, 1290	
<code>\udiss@sfbfont</code> .....	III25
..... 490, 1310	
<code>\udiss@sfbfitfont</code> .....	III25
..... 494, 1312	
<code>\udiss@sfbfslfont</code> .....	III25
..... 498, 1314	
<code>\udiss@sffeatures</code> .....	III25
..... 637, 1324	
<code>\udiss@sffntext</code> .....	III25
..... 641, 698, 1308	
<code>\udiss@sffont</code> .....	III25
..... 488, 1325	
<code>\udiss@sffitfont</code> .....	III25
..... 492, 1311	
<code>\udiss@sfsfont</code> .....	III25
..... 496, 1313	
<code>\udiss@shorttitle</code> .....	III25
..... 74, 77, 87	
<code>\udiss@shorttitletrue</code> ....	III25
..... 75, 78, 88	
<code>\udiss@stream</code> .....	III25
..... 121, 124, 149, 151, 156, 1424, 1426	
<code>\udiss@streamtrue</code> .....	III25
..... 154, 157	
<code>\udiss@subtitle</code> .....	III25
..... 65, 68, 83, 411, 1574, 1648	
<code>\udiss@subtitle@color</code> ....	III25
..... 1019, 1647	
<code>\udiss@subtitle@font</code> .	III25
..... 1016, 1641, 1644	
<code>\udiss@subtitle@fontfeats</code> .	III25
..... 1021, 1643	
<code>\udiss@subtitle@shape</code> ....	III25
..... 1014, 1640	
<code>\udiss@subtitle@size</code> .....	III25
..... 1017, 1646	
<code>\udiss@subtitletrue</code> .....	III25
..... 66, 69, 84	

<code>\udiss@supervisor</code> .....	<code>\udiss@universitytrue</code> .....	266, 269
219, 222, 247, 249, 254, 413, 437, 1479	<code>\textbf</code> .....	1381,
<code>\udiss@supervisortrue</code> .....	1399, 1426, 1444, 1463, 1481, 1509, 1537	
252, 255	<code>\title</code> .....	3, 1084, 1391
<code>\udiss@texlogo</code> .....	<code>title</code> .....	22
290, 303, 1721	<code>titlecolor</code> .....	10, 1002
<code>\udiss@texlogotrue</code> .....	<code>titleffeat</code> .....	10, 1002
291, 305	<code>titlefont</code> .....	10, 1002
<code>\udiss@title</code> ...	<code>titleshape</code> .....	10, 1002
44, 47, 1087, 1089,	<code>titlesize</code> .....	10, 1002
1379, 1381, 1481, 1570, 1582, 1633, 1752	<code>\today</code> .....	424
<code>\udiss@title@color</code> .....	<code>ttbfont</code> .....	9, 500
1008, 1632	<code>ttbfitfont</code> .....	9, 500
<code>\udiss@title@font</code> ..	<code>ttbfsfont</code> .....	9, 500
1005, 1626, 1629	<code>ttfeatures</code> .....	8, 516
<code>\udiss@title@fontfeats</code> ...	<code>ttfntext</code> .....	8, 516
1010, 1628	<code>ttfont</code> .....	9, 500
<code>\udiss@title@shape</code> .....	<code>ttitfont</code> .....	9, 500
1003, 1625	<code>ttslfont</code> .....	9, 500
<code>\udiss@title@size</code> .....		
1006, 1631		
<code>\udiss@tmp@author</code> .....		
51, 52		
<code>\udiss@tmp@degree</code> .....		
137, 138		
<code>\udiss@tmp@department</code> .....		
235, 236		
<code>\udiss@tmp@discipline</code> .....		
128, 129		
<code>\udiss@tmp@stream</code> .....		
119, 120		
<code>\udiss@tmp@supervisor</code> .....		
217, 218		
<code>\udiss@tmp@title</code> .....		
42, 43		
<code>\udiss@tmp@university</code> .....		
226, 227		
<code>\udiss@ttbfont</code> .....		
502, 1333		
<code>\udiss@ttbfitfont</code> .....		
506, 1335		
<code>\udiss@ttbfsfont</code> .....		
510, 1337		
<code>\udiss@ttfeatures</code> .....		
758, 1341		
<code>\udiss@ttfntext</code> .....		
762, 819, 1331		
<code>\udiss@ttfont</code> .....		
500, 1342		
<code>\udiss@ttitfont</code> .....		
504, 1334		
<code>\udiss@ttslfont</code> .....		
508, 1336		
<code>\udiss@uni@color</code> .....		
1041, 1700		
<code>\udiss@uni@font</code> .....		
1038, 1694, 1697		
<code>\udiss@uni@fontfeats</code> .....		
1043		
<code>\udiss@uni@shape</code> .....		
1036, 1693		
<code>\udiss@uni@size</code> .....		
1039, 1699		
<code>\udiss@university</code> .....		
228,		
231, 263, 268, 414, 1507, 1509, 1701		
<code>\udiss@university@fontfeats</code> ...		
1696		

## U

`udiss@title` commands:

<code>\udiss@title:</code> .....	411
<code>udissciteclr</code> .....	6, 443
<code>udisslinkclr</code> .....	6, 443
<code>udissurlclr</code> .....	6, 443
<code>unicolor</code> .....	10, 1035
<code>uniffeat</code> .....	10, 1035
<code>unifont</code> .....	10, 1035
<code>unishape</code> .....	10, 1035
<code>unysize</code> .....	10, 1035
<code>\university</code> .....	3, 188, 1529
<code>university</code> .....	188

# GNU Free Documentation License

Version 1.3, 3 November 2008

Copyright © 2000, 2001, 2002, 2007, 2008 Free Software Foundation, Inc.

<https://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

## Preamble

The purpose of this License is to make a manual, textbook, or other functional and useful document ‘free’ in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of ‘copyleft’, which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

## I. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The ‘**Document**’, below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as ‘**you**’. You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A ‘**Modified Version**’ of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A ‘**Secondary Section**’ is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document’s overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The ‘**Invariant Sections**’ are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is

not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The ‘**Cover Texts**’ are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A ‘**Transparent**’ copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not ‘Transparent’ is called ‘**Opaque**’.

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format,  $\LaTeX$  input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The ‘**Title Page**’ means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, ‘Title Page’ means the text near the most prominent appearance of the work’s title, preceding the beginning of the body of the text.

The ‘**publisher**’ means any person or entity that distributes copies of the Document to the public.

A section ‘**Entitled XYZ**’ means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as ‘**Acknowledgements**’, ‘**Dedications**’, ‘**Endorsements**’, or ‘**History**’.) To ‘**Preserve the Title**’ of such a section when you modify the Document means that it remains a section ‘**Entitled XYZ**’ according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

## 2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical

measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

### 3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

### 4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.



- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section Entitled 'History', Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled 'History' in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the 'History' section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. For any section Entitled 'Acknowledgements' or 'Dedications', Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section Entitled 'Endorsements'. Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section to be Entitled 'Endorsements' or to conflict in title with any Invariant Section.
- O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled 'Endorsements', provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of

peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

## **5. COMBINING DOCUMENTS**

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled ‘History’ in the various original documents, forming one section Entitled ‘History’; likewise combine any sections Entitled ‘Acknowledgements’, and any sections Entitled ‘Dedications’. You must delete all sections Entitled ‘Endorsements’.

## **6. COLLECTIONS OF DOCUMENTS**

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

## **7. AGGREGATION WITH INDEPENDENT WORKS**

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an ‘aggregate’ if the copyright resulting from the compilation is not used to limit the legal

rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

## 8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled 'Acknowledgements', 'Dedications', or 'History', the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

## 9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

## 10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to

the present version, but may differ in detail to address new problems or concerns. See <https://www.gnu.org/licenses/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License ‘or any later version’ applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy’s public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

## II. RELICENSING

‘Massive Multiauthor Collaboration Site’ (or ‘MMC Site’) means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A ‘Massive Multiauthor Collaboration’ (or ‘MMC’) contained in the site means any set of copyrightable works thus published on the MMC site.

‘CC-BY-SA’ means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

‘Incorporate’ means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is ‘eligible for relicensing’ if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

## ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright © YEAR YOUR NAME.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled ‘GNU Free Documentation License’.

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the ‘with ... Texts.’ line with this:

with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.