

# HsH-Classes — A set of $\text{\LaTeX}$ classes for use in Hochschule Hannover <sup>\*</sup>

Jan Wille<sup>†</sup>

Created 21.11.2024

Maintained on <a href="https://lab.it.hs-hannover.de/qxx-tul-u1/latex-template-hsh">https://lab.it.hs-hannover.de/qxx-tul-u1/latex-template-hsh</a>
---

## Contents

### Abstract

The following documents a set of  $\text{\LaTeX}$  classes created for the Hochschule Hannover. They are intended to ease the workflow when writing documents by providing a common formatting basis that should work for pretty much everything a student will be expected to write. This can be simple one-paged documents, excercises, lab-reports, papers or bachelors and masters thesises.

The classes provide interfaces to modify commend requiriements, provide commands to get specifics like the logo and provide and pre-configure comonly needed packages. This should get you going imidealty and reduce the setuptime significantly.

---

<sup>\*</sup>This file describes version 3.03.dev0, last revised 2024-08-10.

<sup>†</sup>E-mail: [mail@janwille.de](mailto:mail@janwille.de)

# 1 The different classes

The project classes provided by the Project all carry the HsH- prefix. Here is a list of the available classes and some explanation on when to use which class.

HsH-article	A article-class based on KOMA-Scripts scrartcl. It is designed for quick and compact documents and is useful for writing lab-protocols and alike. It does not have chapters and therefore never breaks to a new page on its own.
HsH-report	A report-class based on KOMA-Scripts scrreprt. This is probably the most useful class, as it can be used for a wide variety of documents (beginning with lab-reports and ending at complete thesis). The line between article and report is somewhat blurry, so use as you see fit.
HsH-standalone	A helper class based based on the standalone class. It is designed only for creating images as separate documents to keep things organized and compiler times low. It is useful for creating graphs, circuit diagrams or other kind of complex sub documents.

# 2 Document options

To configure the behavior and style of documents using this class, options can be passed via the `\documentclass[⟨options⟩]{⟨document-class⟩}` command.

It should be noted that all unknown keys will be passed to the parent class and a log-message issued.

## 2.1 Generic options share by all classes

These Options are available regardless of documentclass and modify common things.

fontfamily	The fontfamily=⟨opt⟩ option configures which font-style is used. For convenience there are also short-forms provided. The available options are:
sans	
roman	
	sans   sans-serif A sans-serif font is used (similar to Arial)
	roman   serif A serif font is used (similar to Times-new-Roman)
language	The language=⟨opt⟩ option set the main language you write in. It ensures texts like auto-generated headings are localised properly. You can pass in any language-name understood by the babel package. German is the default. For convenience there are also short-forms provided.
english	
german	
faculty	The faculty=⟨opt⟩ option configures which faculties logo is used. For convenience there are also short-forms provided. The available options are:
f1	
f2	none   false A gray, non-faculty specific logo is used.
f3	
f4	1   f1 The blue logo of faculty 1 is used.
f5	2   f2 The green logo of faculty 2 is used.
	3   f3 The orange logo of faculty 3 is used.
	4   f4 The red logo of faculty 4 is used.
	5   f5 The purple logo of faculty 5 is used.

## 2.2 Options for modifying the document

The following options are only available for documents (so not utility classes).

linespacing	The linespacing=⟨opt⟩ option configures the spacing in between lines. For convenience there are also short-forms provided. The available options are:
singlespacing	
onehalfspacing	
doublespacing	
	single No additional space is added in between lines.
	onehalf Aproximalty half a line of empty space is added inbetween lines.
	double About a full lineheight is left in between lines.
parskip	The parskip=⟨opt⟩ option configures the spacing in between paragraphs. This is an extending option originally implemented by KOMA-Script.

<code>never</code>	No inter-paragraph spacing will be inserted even if additional vertical spacing is needed for vertical adjustment with <code>\flushbottom</code> .										
<code>never+</code>	No inter-paragraph spacing will be inserted. There must be at least a third of a line of free space at the end of a paragraph.										
<code>never*</code>	No inter-paragraph spacing will be inserted. There must be at least a quarter of a line of free space at the end of a paragraph.										
<code>...</code>	see <a href="#">KOMA-Script manual, Table 3.7</a> for more options.										
<code>headheight</code>	The <code>headheight=&lt;dim&gt;</code> option allows you to set the required size of the header. You may need to modify this if you get a <code>\headheight to low</code> error message. The emssage should tell you what value you need, but you can pass any valid length.										
<code>abstract</code>	The <code>abstract=&lt;opt&gt;</code> option allows you to configure different behaviors of the abstract. The availabel options are described below: <table> <tr> <td><code>keywords</code></td><td>Print the <code>\keywords</code> after the abstract.</td></tr> <tr> <td><code>nokeywords</code></td><td>Do <i>not</i> print the <code>\keywords</code> after the abstract.</td></tr> <tr> <td><code>totoc</code></td><td>The abstract will be listed in the table of contentes.</td></tr> <tr> <td><code>notoc</code></td><td>The abstract will <i>not</i> be listed in the table of contentes.</td></tr> </table>	<code>keywords</code>	Print the <code>\keywords</code> after the abstract.	<code>nokeywords</code>	Do <i>not</i> print the <code>\keywords</code> after the abstract.	<code>totoc</code>	The abstract will be listed in the table of contentes.	<code>notoc</code>	The abstract will <i>not</i> be listed in the table of contentes.		
<code>keywords</code>	Print the <code>\keywords</code> after the abstract.										
<code>nokeywords</code>	Do <i>not</i> print the <code>\keywords</code> after the abstract.										
<code>totoc</code>	The abstract will be listed in the table of contentes.										
<code>notoc</code>	The abstract will <i>not</i> be listed in the table of contentes.										
<code>toc</code>	The <code>toc=&lt;opt&gt;</code> option configures what is listed in the table of contents. <table> <tr> <td><code>totoc</code></td><td>The table of contentes will list itself.</td></tr> <tr> <td><code>notoc</code></td><td>The table of contentes will <i>not</i> list itself.</td></tr> <tr> <td><code>abstract</code></td><td>The abstract will be listed in the table of contentes.</td></tr> <tr> <td><code>noabstract</code></td><td>The abstract will <i>not</i> be listed in the table of contentes.</td></tr> <tr> <td><code>...</code></td><td>see <a href="#">KOMA-Script manual, Table 3.5</a> for more options</td></tr> </table>	<code>totoc</code>	The table of contentes will list itself.	<code>notoc</code>	The table of contentes will <i>not</i> list itself.	<code>abstract</code>	The abstract will be listed in the table of contentes.	<code>noabstract</code>	The abstract will <i>not</i> be listed in the table of contentes.	<code>...</code>	see <a href="#">KOMA-Script manual, Table 3.5</a> for more options
<code>totoc</code>	The table of contentes will list itself.										
<code>notoc</code>	The table of contentes will <i>not</i> list itself.										
<code>abstract</code>	The abstract will be listed in the table of contentes.										
<code>noabstract</code>	The abstract will <i>not</i> be listed in the table of contentes.										
<code>...</code>	see <a href="#">KOMA-Script manual, Table 3.5</a> for more options										
<code>todos</code>	The <code>todos</code> option is a simple switch that activates support for the <code>todonotes</code> package. It enables/disables the package as well as increasing the pagewidth to make space for the notes. You can use the commands from the package to make notes and other anotations (similar to how MS-Words comments work). When you pass <code>off</code>   <code>false</code> all the anotation will disaper from the PDF while still beeing in the source.										

**Note:** setting `todos=off` after having used the option will mostlikly produce compilation errors. These will go away after you remove the temporary files.

### Often relevant KOMA-Script options

The following options are implemented by the parrent classes and only listed here for completeness. For more detils, see [its manual](#).

<code>fontsize</code>	The <code>fontsize=&lt;size&gt;</code> options takes a size in pt. It is usually in therange of 10-12, but other sizes can work as well.
<code>paper</code>	The <code>paper=&lt;size&gt;</code> options accepts a number of options, most ISO formats are supported, but also others like <code>letter</code> or <code>legal</code> .
<code>twoside</code>	The <code>twoside</code> option sets your document up for doublesided printing. The header and footer will take this into account and binding-correction will be applide along the inner edge.
<code>BCOR</code>	The <code>BCOR=&lt;dim&gt;</code> option allows you to define a custom binding-correction. Any valid length can be put here, but to large of a value will shrink the outer margin to a not-desirable level.

## 2.3 Standalone specific options

The standalone utility class has some special options which are documented here.

<code>margin</code>	The <code>margin=&lt;dim&gt;</code> option controlls how much whicspace is added arround you standalone document. This usually looks better which is why the default is 0.25 cm, but you can supress it by passing 0 cm.
<code>multi</code>	The <code>multi=&lt;opt&gt;</code> option defines which enviroment make up a page. It can be passe more than once.

### 3 Provided commands

The classes define a set of commands which are explained in the following section.

`\HsHClassName` Each class defines the macro `\HsHClassName` to contain its classname. This is mostly usefull so internals can reuse the classname, but you could also check against it if you needed to.

#### 3.1 Title matters

$\text{\LaTeX}$  has a set of default commands which are used to define data for the titlepage, like `\title` or `\author`. The classes define a few additional commands, which are documented her.

`\matrikelnr` The `\matrikelnr{<nr[, ..]>}` macro sets the matrikelnumber of the author(s). It can be a single number or a comma seperated list of numbers. The numbers will be matched to the authors passed into `\author`.

`\professor` You can pass any text to `\professor{<text>}`, it will be printed on the bottom of the titelpage.

`\keywords` This macro can be used to define keywords which are relevant to your document. They will be printed as part of the abstract and be put into the PDF's meta-data.

#### Modifying the Logo

`\HsHlogoPath` The logo is loaded from a file whos name is read from `\HsHlogoPath`. Change it to use a different logo-file.

`\HsHlogoPage` As the PDF file can hae multiple pages, the `\HsHlogoPage` command stores which page to load.

`\includeHsHlogohere` The macro `\includeHsHlogohere[<width>]` is used by `\maketitle` to produce the logo. But if you want it elsewhere, you can call this macro yourself.

#### 3.2 Commands for document writing

The following commands will be usefull to you when writing a document.

`\declarationofauthorship` The command `\declarationofauthorship[<align>]` can be used to print a "declatation of authorship" in the current location, similar to how `tableofcontents` and `friends` work. It will produce a horizontal line, a text block containing the regulatory text and a signature block for every author. The command is localised for both english and german. Using the optional argument, you can define the positioning. Pass `t` for alignment at the top of the page and `b` for bottom alignment (*default*).

The three commands `\frontmatter`, `\mainmatter` and `\backmatter` are provided for report classes. They separete document section and automatically set up `pagenumer` styles.

`\frontmatter` `\frontmatter` set the `pagenumers` to capital roman numerals. This is usually required for everything before the first chapter.

`\mainmatter` `\mainmatter` sets the `pagenumbering` to "normal" arabic numbers. This is usually the style for the document content.

`\backmatter` `\backmatter` can be used for apendixes and alike. It sets the `pagenumbering` to small roman numerals.

#### 3.3 Default $\text{\LaTeX}$ Commands that are modified

Additionally, some of  $\text{\LaTeX}$ s default commands are moddified to better fit this class. This is documented here.

`\title` The `\title[<short-title>]{<title>}` command now takes an additional, optional argument. You can use it for a shorter version of your title, that will be used in the header to save on space.

`\maketitle` The `\maketitle` command if  $\text{\LaTeX}$  default way to create a titlepage. We redefine it to produce a titlepage that matches the sytel typically used on the Hochschule Hannover. This incudes the logo beeing printed, depending on the `f1` to `f5` `documentoptions`. The command now also takes an options alignment-parameter: `\maketitle[<align>]`. You can pass `l` | `c` | `r` to get *left*, *center* or *right* alignment.

### 3.4 Miscellaneous commands

The following commands are mostly for internal use, but are part of the interface and might be useful in some cases.

`\ifsingleauthor` This allows you to check if `\author` holds one or multiple entries. Using this allows the user to display different things for these cases, for example switch words to plural, like this:  
`\ifsingleauthor{one}{multiple} person\ifsingleauthor{}{s}`

## 4 Package loading

The classes load some packages for internal use as well as loading and configuring common use packages. The details are documented in the following section.

### 4.1 Always loaded packages

`fontenc` for output encoding, set to the european characterset  
`babel` for language-specific typesetting  
`bookmark` creates bookmarks in the PDF  
`hyperref` for easy referencing and linking  
`caption` to customize captions and make references point to the beginning of the floats  
`graphicx` for importing and manipulating images  
`amsmath,amssymb,amsfonts` more options when typesetting math  
`lmodern` sets up the Latin-Modern font  
`setspace` used for configuring linespacing  
`enumitem` more powerful item-lists and enumerations

There are also some packages for internal functionality that shouldn't concern the user, but they are listed here for completeness.

`scrbase`  
`pgffor`

### 4.2 Conditionally loaded Packages

A subset of packages is only loaded (or loaded with specific options) depending on options passed to the package

`babel` gets configured depending on `language`  
`csquotes` for language-specific quotations marks  
`ziffer` only loaded for german documents, sets comma as decimal separator  
`todonotes` loaded depending on `todos`

### 4.3 Pre-configured, but not loaded Packages

These packages are configured by the class to work in a cooperative way. The user must load them manually in this preamble via `\usepackage{<pkg-name>}`, as loading them always bears additional, unnecessary overhead.

`biblatex` create a bibliography  
`siunitx` use units in text and equations  
`listings` typeset sourcecode snippets  
`csvsimple` import csv-files into tables  
`pgfplots` / `tikz` A powerful package for drawing and plotting data  
`circuitikz` A TikZ library for drawing circuits

## 5 Implementation

### 5.1 Internal commands

`\HsHClassName` The classname of specific class is stored in the `\HsHClassName` which gets used throughout the code.

```
1 \let\HsHClassName\@currname
```

There is also a second macro `\HsHClassName@ParrentClass` which stores the parent classes name.

```
2 \def\HsHClassName@ParrentClass{%
3 <article> scrartcl%
4 <report> scrreprt%
5 <standalone> standalone%
6 }
```

### 5.2 Option handling

The options are handled using features provided by the KOMA-Script ecosystem. To get access to this the `srcbase` package is loaded.

```
7 \RequirePackage{srcbase}
```

We also require some packages for some of the option, these are loaded next.

```
8 \RequirePackage{setspace}
```

A new family of keys is created and shared by all elements of this project. For that the `\DefineFamily` macro is used. The family name is *HsH*, matching the usual prefixes.

```
9 \DefineFamily{HsH}
```

Additionally each class-file represents a member in the family. This is defined using the `\DefineFamilyMember` macro. It's optional argument is set to the current filename by default, so we do not need to specify it, just the family name to attach it to.

```
10 \DefineFamilyMember{HsH}
```

`\HsH@Options@PassToParrent` To be able to pass options to the parent class where needed easily, a command is defined. It also issues a log-message.

```
11 \newcommand{\HsH@Options@PassToParrent}[1]{%
12   \ClassInfoNoLine{\HsHClassName}{passing option to parent class: #1}%
13   \PassOptionsToClass{#1}{\HsHClassName@ParrentClass}%
14 }
```

`\HsH@Options@DeclareAlias` It is also useful to have short-versions of options. The following commands make it easy to declare these.

```
15 \newcommand{\HsH@Options@DeclareAlias}[3][HsH]{%
16   \DeclareOption{#2}{\FamilyExecuteOptions{#1}{#3}}%
17 }
```

`fontfamily` The first option to be defined is `fontfamily`. It's defined as a *Numerical* key so that it can accept multiple options and map them to a switch case.

```
18 \DefineFamilyKey{HsH}{fontfamily}{
19   \begingroup
20   \FamilySetNumerical{HsH}{parskip}{@tmp}{%
21     {sans}{0}, {sans-serif}{0},%
22     {roman}{1}, {serif}{1},%
23   }{#1}
24   \ifx\FamilyKeyState\FamilyKeyStateProcessed
25     \aftergroup\FamilyKeyStateProcessed
26     \ifcase@tmp% 0
27       \endgroup
28       \renewcommand{\familydefault}{\sfdefault}
29     \if@atdocument\AfterKOMAOptions{\selectfont}\fi
30   \or% 1
31     \endgroup
```

```

32     \renewcommand{\familydefault}{\rmdefault}
33     \if@atdocument\AfterKOMAOptions{\selectfont}\fi
34     \else% should never be
35     \endgroup
36     \fi
37 \else
38     \endgroup
39     \FamilyKeyStateUnknownValue
40 \fi
41 }

```

`sans` For convinience, there are also two short-versions defined.

```

roman 42 \HsH@Options@DeclareAlias{sans}{fontfamily=sans}
43 \HsH@Options@DeclareAlias{roman}{fontfamily=roman}
44 \<!*standalone>

```

`linespacing` The linespacing options is also a *Numerical* option, mapping to three cases. They execute the appropriate commands of the setspace package.

```

45 \DefineFamilyKey{HsH}{linespacing}{
46   \begingroup
47   \FamilySetNumerical{HsH}{linespacing}{@tempa}{%
48     {single}{0},%
49     {onehalf}{1},%
50     {double}{2},%
51   }{#1}
52   \ifx\FamilyKeyState\FamilyKeyStateProcessed
53     \aftergroup\FamilyKeyStateProcessed
54     \ifcase@tempa% 0
55       \endgroup
56       \if@atdocument\singlespacing\else\AtEndOfClass{\singlespacing}\fi
57     \or% 1
58       \endgroup
59       \if@atdocument\onehalfspacing\else\AtEndOfClass{\onehalfspacing\AfterTOCHead{\singlespac
60     \or% 2
61       \endgroup
62       \if@atdocument\doublespacing\else\AtEndOfClass{\doublespacing\AfterTOCHead{\singlespacin
63
64     \else% should never be
65     \endgroup
66     \fi
67 \else
68     \endgroup
69     \FamilyKeyStateUnknownValue
70 \fi
71 }

```

`singlespacing` For convinience, there are also these short-versions defined.

```

onehalfspacing 72 \HsH@Options@DeclareAlias{singlespacing}{linespacing=single}
doublespacing 73 \HsH@Options@DeclareAlias{onehalfspacing}{linespacing=onehalf}
74 \HsH@Options@DeclareAlias{doublespacing}{linespacing=double}

```

`parskip` The parskip option is special in that it originally a KOMA-Script option that get expanded by this class. Only two new cases are defined here and everything unknown gets passed to the parent class.

It should also be noted that this option can't execute its code ideally, as the commands needed are only defined later when the parent class loads in. So the `\setparsizes` command is pushed into a hook.

```

75 \DefineFamilyKey{HsH}{parskip}{%
76   \begingroup
77   \FamilySetNumerical{HsH}{parskip}{@tempa}{%
78     {never+}{0},%
79     {never*}{1},%

```

```

80 }{#1}
81 \if@atdocument
82 \ClassError{\HsHClassName}{
83   option 'parskip' can only be configured in preamble!
84 }
85 \fi
86 \ifx\FamilyKeyState\FamilyKeyStateProcessed
87   \aftergroup\FamilyKeyStateProcessed
88   \ifcase\@tempa% 0
89     \endgroup
90     \AtEndOfClass{\setparsizes{\z@}{\z@}{.3333\linewidth \@plus 1fil}}
91   \or% 1
92     \endgroup
93     \AtEndOfClass{\setparsizes{\z@}{\z@}{.25\linewidth \@plus 1fil}}
94   \else% should never be
95     \endgroup
96   \fi
97 \else
98   \endgroup
99   \HsH@Options@PassToParrent{parskip=#1}
100   \FamilyKeyStateProcessed
101 \fi
102 }

```

**headheight** The headheight option just set the \headheight to the given value.

```

103 \def\HsH@opt@headheight{
104 \FamilyStringKey{HsH}{headheight}{\HsH@opt@headheight}
105 \AtEndOfClass{
106   \headheight=\HsH@opt@headheight%
107 }

```

**abstract** The abstract option sets multiple different switches and configurations. We first define the needed macros:

```

108 \newif\if\HsH@option@abstract@show@keywords
109 \def\HsH@abstract@heading{
110 <article> \addsec%
111 <report> \addchap%
112 }

```

Now the actual option can be defined to handle all the cases.

```

113 \DefineFamilyKey{HsH}{abstract}{
114   \beginngroup
115   \FamilySetNumerical{HsH}{abstract}{\@tempa}{
116     {keywords}{0},%
117     {nokeywords}{1},%
118     {totoc}{2},{toc}{2},%
119     {notoc}{3},{nottotoc}{3},%
120   }{#1}
121   \ifx\FamilyKeyState\FamilyKeyStateProcessed
122     \aftergroup\FamilyKeyStateProcessed
123     \ifcase\@tempa% 0
124       \endgroup
125       \@HsH@option@abstract@show@keywordstrue
126     \or% 1
127       \endgroup
128       \@HsH@option@abstract@show@keywordsfalse
129     \or% 2
130       \endgroup
131     \def\HsH@abstract@heading{
132 <article> \addsec%
133 <report> \addchap%
134     }
135     \or% 3
136     \endgroup

```



```

137     \def\HsH@abstract@heading{%
138 <article>         \addsec*%
139 <report>          \addchap*%
140     }
141     \else% should never be
142     \endgroup
143     \fi
144 \else
145     \endgroup
146 \fi
147 }

```

toc The toc option is a KOMA-Script options we just extend. Options for the abstract and toc are added.

```

148 \DefineFamilyKey{HsH}{toc}{%
149   \begingroup
150   \FamilySetNumerical{HsH}{toc}{@tempa}{%
151     {totoc}{0},{toc}{0},%
152     {notoc}{1},{nottotoc}{1},%
153     {abstract}{2},%
154     {noabstract}{3},%
155   }{#1}
156   \ifx\FamilyKeyState\FamilyKeyStateProcessed
157     \aftergroup\FamilyKeyStateProcessed
158     \ifcase@tempa% 0
159       \endgroup
160       \AtEndOfClass{\setuptoc{toc}{totoc}}
161     \or% 1
162       \endgroup
163       \AtEndOfClass{\unettoc{toc}{totoc}}
164     \or% 2
165       \endgroup
166       \FamilyExecuteOptions{HsH}{abstract=totoc}
167     \or% 3
168       \endgroup
169       \FamilyExecuteOptions{HsH}{abstract=nottotoc}
170     \else% should never be
171       \endgroup
172     \fi
173   \else
174     \endgroup
175     \HsH@Options@PassToParent{toc=#1}
176     \FamilyKeyStateProcessed
177   \fi
178 }

```

For the twoside option we only redefine the default, everything else is handled by the parent class.

```

179 \DefineFamilyKey{HsH}{twoside}[semi]{%
180   \HsH@Options@PassToParent{twoside=#1,BCOR=1cm}
181   \FamilyKeyStateProcessed
182 }
183 \!/standalone)

```

\HsH@opt@language First, the macro to store the languages name in is created. The default is nil, as babel will see this as no-language.

```
184 \def\HsH@opt@language{nil}
```

language The key is then defined to store its value inside the command. This allows the option to be called multiple times, but only the last set value will be passed on to babel.

```
185 \FamilyStringKey{HsH}{language}{\HsH@opt@language}
```

english For convinience, there are also these short-versions defined.

```

german 186 \HsH@Options@DeclareAlias{english}{language=english}
ngerman 187 \HsH@Options@DeclareAlias{german}{language=ngerman}
188 \HsH@Options@DeclareAlias{ngerman}{language=ngerman}

```

\HsH@opt@faculty As we need a default value that is not zero, the macro needs to be defined an initialised manually.

```
189 \def\HsH@opt@faculty{1}
```

faculty The faculty options is once again a *Numerical* option, mapping the five faculties and storing the selected one in \HsH@opt@faculty.

```

190 \FamilyNumericalKey{HsH}{faculty}{HsH@opt@faculty}{%
191   {0}{1}, {none}{1}, {false}{1}, {off}{1},%
192   {1}{2}, {f1}{2},%
193   {2}{3}, {f2}{3},%
194   {3}{4}, {f3}{4},%
195   {4}{5}, {f4}{5},%
196   {5}{6}, {f5}{6},%
197 }

```

f1 For convinience, there are also these short-versions defined.

```

f2 198 \HsH@Options@DeclareAlias{f1}{faculty=f1}
f3 199 \HsH@Options@DeclareAlias{f2}{faculty=f2}
f4 200 \HsH@Options@DeclareAlias{f3}{faculty=f3}
f5 201 \HsH@Options@DeclareAlias{f4}{faculty=f4}
202 \HsH@Options@DeclareAlias{f5}{faculty=f5}

```

todos The boolean option todos is simply created using the commands from scrbase. Boolean options allready default to *<true>* if called without and argument, so no need to define an explicit alias.

```
203 \FamilyBoolKey{HsH}{todos}{@todos}
```

For the standalone class the fontsize option is mocked to present a standardised interface. A user might expect this option to be passable to this class and we shouldnt crete an anoying error just for this.

```

204 <standalone>
205 \DefineFamilyKey{HsH}{fontsize}{%
206   \ClassInfoNoLine{\HsHClassName}{The 'fontsize' option is only a mock, its has not effect}
207   \FamilyKeyStateProcessed
208 }
209 </standalone>

```

### 5.2.1 Unknown options

Unknown options will be passed to the parent class. For that a @else@ key is defined on the HsH family, which will be execute for every unknown key-value option. Unknown bare options are handled by the \DeclareOption\* command and will be passed there.

```

210 \DefineFamilyKey{HsH}{@else@}{
211   \HsH@Options@PassToParrent{#1}
212   \FamilyKeyStateProcessed
213 }
214 \DeclareOption*{
215   \HsH@Options@PassToParrent{\CurrentOption}
216 }

```

### 5.2.2 Default options

The different classes all execute a set of default options, which is handled by the following code.

```

217 \FamilyExecuteOptions{HsH}{%
218   fontfamily=sans-serif,

```

```

219 <!*standalone>
220   fontsize=11pt,
221   language=ngerman,
222 <article>   parskip=never+,
223 <report>   parskip=half+,
224   linespacing=single,
225   headheight=2.15\baselineskip,
226 </!*standalone>
227 <*article | report>
228   toc=listof,
229   toc=bibliography,
230   abstract=keywords,
231 </article | report>
232   faculty=none,
233 <*standalone>
234   margin=0.25cm,
235   multi=tikzpicture,
236   multi=circuitikz,
237 </standalone>
238 }

```

Now we can process the options for the HsH family.

```

239 \FamilyProcessOptions{HsH}\relax

```

### 5.2.3 Loading the parent class

```

240 \LoadClass{\HsHClassName@ParrentClass}

```

## 5.3 Package loading

The classes load and configure some common packages to reduce the needed amount of boilerplate code in your preamble.

Additionally there are settings provided for packages that are used more rarely, but will be set up correctly if you decide to load them via `\usepackage{}`.

### 5.3.1 Ensuring German works

With modern LaTeX systems the encoding of inputfiles is UTF-8 by default, so the inputenc package is no longer required. Should the user still use a old setup or use a different encoding, he is responsible for loading inputenc himself.

The font-encoding for the pdf file is also set up to allow for the full european characterset.

```

241 \RequirePackage[T1]{fontenc}
242 \RequirePackage{type1ec}

```

To ensure localised translations of all displayed text automatically dependign on the user-selected language, the babel package is loaded. This also allows for the use of the `\iflanguage` command, which is relevant later.

```

243 \RequirePackage[main=\HsH@opt@language]{babel}

```

Quotationsmarks are also very different between languages, so the following ensures the correct style for the correct language.

```

244 \RequirePackage[autostyle=true]{csquotes}
245 \MakeOuterQuote{"}

```

German uses a comma as the decimal separator, which collides with LaTeX's default english setting of using the comma as a thousands separator and therefore replacing it with some whitespace on printed version. Luckily loading the ziffer package sets this up to match the german standart.

```

246 \iflanguage{ngerman}{\RequirePackage{ziffer}}{}

```

### 5.3.2 Generally usefull packages

We load hyperref for clickable links, a interactive document outline in the PDF and configure it to write meta-data to the PDF.

```

247 <!*standalone>
248 \RequirePackage[hidelinks]{hyperref} % must load before 'bookmarks'
249 \RequirePackage{bookmark}
250 \AtBeginDocument{
251   \hypersetup{
252     pdfinfo={
253       Title={\@title},
254       Author={\@author},
255       Subject={\@subject},
256       Keywords={\@keywords}
257     }
258   }
259 }
260 </!*standalone>

```

The hyperref package has a problem when linking to images, clicking a link to one leads to the *bottom* of the float (where the caption is located), instead of to the *top*. Loading the caption package with the hypercap option fixes that.

```
261 \RequirePackage[hyperc=true]{caption}
```

To display images (mostly through the `\includegraphics` command) we require the graphicx package.

```
262 \RequirePackage{graphicx}
```

The ams family of packages contains a wide collection of math related commands that are generally usefull.

```
263 \RequirePackage{amsmath,amssymb,amsfonts}
```

The enumitem package is very helpful for defining lists of any kind. We use it to provide a checklist implementation.

```
264 \RequirePackage{enumitem}
```

The todonotes package is greate for anotation, but extremly expensive on compiletime. So we load it only if the user requests it. Also its commands are stubed, so that they can be left in the sourcecode and jut not output anything.

```

265 \if@todos
266   \PassOptionsToPackage{svgnames}{xcolor}
267   \AtEndOfClass{\RequirePackage{xcolor}}
268   \RequirePackage[
269     textsize=small,
270     figwidth=.6\textwidth
271   ]{todonotes}
272   \setuptodonotes{
273     backgroundcolor=orange!20!white,
274     linecolor=orange,
275     bordercolor=orange!60!white,
276   }
277 \else
278   \newcommand{\listoftodos}[1]{}
279   \newcommand{\todo}[2] [] {}
280   \newcommand{\missingfigure}[2] [] {}
281 \fi

```

Some more micalanious packages:

```
282 \RequirePackage{xcolor}
```

### 5.3.3 Options for packages that could be loaded by the user

Some package are not always needed and potentially heavy to load in by default. But its still usefull to set default options for these packages.

These differ from the settings provided in `HsH-classes.cfg` in that they are defaults that apply allway and not user-configurable preferences which are user or even project specific.

For the bibtex we ensure the *biber* backend is selcted, which matches the settings in `.latexmkrc`.

```
283 <*article | report>
```

```

284 \PassOptionsToPackage{
285   backend=biber,
286   bibwarn=true,
287   url=true,
288   isbn=false,
289 }{biblatex}
290 \AddToHook{package/biblatex/after}{
291   \renewcommand*{\mkbibacro}[1]{\MakeUppercase{#1}}
292 }
293 </article | report>

```

For bibtex we load the free-standing units, mostly for backwards compatibility. We also ensure german language specific settings are applied.

```

294 \PassOptionsToPackage{free-standing-units}{siunitx}
295 \AddToHook{package/siunitx/after}{
296   \sisetup{
297     tight-spacing=true,
298     per-mode=symbol,
299     exponent-mode=engineering,
300     round-mode=places,
301     round-precision=3,
302   }
303   \iflanguage{ngerman}{
304     \sisetup{output-decimal-marker={,}}
305   }{}
306 }

```

For better compatibility with the listings package we load the scrhack package. We also pass some configurations to it if it gets loaded.

```

307 \RequirePackage{scrhack}
308 \AddToHook{package/listings/before}{\RequirePackage{xcolor}}
309 \AddToHook{package/listings/after}{
310   \lstset{
311     commentstyle=\color{gray},
312     keywordstyle=\color{FireBrick}\bfseries,
313     stringstyle=\color{DodgerBlue},
314     showstringspaces=false,
315     basicstyle=\ttfamily,
316     tabsize=4,
317     escapeinside={(*@){@*}},
318   }
319   \RequirePackage{lstautogobble}\lstset{autogobble=true}
320   \iflanguage{ngerman}{
321     \lstset{literatex={\0}{\1}{\2}{\3}{\4}{\5}{\6}{\7}{\8}{\9}{\10}{\11}{\12}{\13}{\14}{\15}{\16}{\17}{\18}{\19}{\20}{\21}{\22}{\23}{\24}{\25}{\26}{\27}{\28}{\29}{\30}{\31}{\32}{\33}{\34}{\35}{\36}{\37}{\38}{\39}{\40}{\41}{\42}{\43}{\44}{\45}{\46}{\47}{\48}{\49}{\50}{\51}{\52}{\53}{\54}{\55}{\56}{\57}{\58}{\59}{\60}{\61}{\62}{\63}{\64}{\65}{\66}{\67}{\68}{\69}{\70}{\71}{\72}{\73}{\74}{\75}{\76}{\77}{\78}{\79}{\80}{\81}{\82}{\83}{\84}{\85}{\86}{\87}{\88}{\89}{\90}{\91}{\92}{\93}{\94}{\95}{\96}{\97}{\98}{\99}}
322   }{}
323 }

```

The csvsimple is helpful when wanting to import CSV-files and display them as tables.

```

324 \PassOptionsToPackage{13}{csvsimple}
325 \AddToHook{package/csvsimple/after}{
326   \csvstyle{every csv}{separator=semicolon}
327 }

```

pgfplots is a powerful package for plotting data.

```

328 \AddToHook{package/pgfplots/after}{
329   \usepgfplotslibrary{fillbetween}
330   \usetikzlibrary{patterns}
331   \pgfplotsset{
332     compat=1.16,
333     /pgf/number format/read comma as period,
334     tick label style={font=\tiny},
335     legend style={font=\footnotesize},
336     MyPlots/.style = { % Style for all plots
337       width=.6\textwidth,
338       grid=major,

```

```

339     grid style={loosely dotted,gray!50},
340     legend pos=outer north east,
341     FM1/.style = {blue, thick, mark=x, mark size=1.5,samples=100},    % Format 1
342     FM2/.style = {orange, thick, mark=*, mark size=1,samples=100},    % Format 2
343     FM3/.style = {green, thick, mark=square*, mark size=1,samples=100}, % Format 3
344     FM4/.style = {red, thick, mark=diamond*, mark size=1,samples=100}, % Format 4
345     FM5/.style = {pink, thick, mark=triangel*, mark size=1,samples=100},% Format 5
346     errBars/.style = {error bars/.cd, y dir=both, y explicit, % ermöglicht Fehlerbalken
347         error mark options={mark size=1pt,rotate=90}},
348 }
349 }
350 \iflanguage{ngerman}{
351     \pgfplotsset{/pgf/number format/use comma}
352 }{}
353 }
354 \AddToHook{package/pgfplotstable/after}{
355     \pgfplotstableset{
356         col sep=semicolon, % global separator for csv files
357     }
358 }

```

The circuitikz needs to be configured so it matches typical European styles.

```

359 \PassOptionsToPackage{
360     european,
361     EFvoltages,
362     straightvoltages,
363     betterproportions
364 }{circuitikz}
365 \AddToHook{package/circuitikz/after}{
366     \ctikzset{
367         font=\footnotesize,
368         bipole annotation style/.style={font=\tiny ,inner sep=1pt},
369     }
370 }

```

For other packages we provide the settings more as a recommendation of what is useful.

As the user might want to change these, we outsource this to a separate file and input it `\AtBeginDocument`. That way the user can just replace the file with his custom version.

```

371 \AtBeginDocument{
372     \makeatletter
373     \InputIfFileExists{HsH-classes.cfg}{
374         \ClassInfo{\HsHClassName}{Local config file HsH-classes.cfg used.}
375     }{
376         \ClassInfo{\HsHClassName}{No HsH-classes.cfg!! I hope you configured it yourself.}
377     }
378     \makeatother
379 }

```

## 5.4 Custom commands

### 5.4.1 Document separation commands

The following commands are only defined for book type classes by default. But they are also useful for the report class, so we define them in that case.

```
380 \report
```

`\if@mainmatter` We define a switch which stores if the document is currently at a mainmatter section. It defaults to true as the user needs to explicitly set the state to something different.

```
381 \newif\if@mainmatter\@mainmattertrue
```

As a page number change requires a fresh page, this is ensured first. We also need to make sure that on a two-sided document, the first page is always on the left.

`\frontmatter` The pagenumbering is set to capital roman numerals.

```
382 \newcommand{\frontmatter}{
383   \if@twoside\cleardoubleoddpagel\else\clearpagel\fi
384   \@mainmatterfalse\pagenumbering{Roman}
385 }
```

`\mainmatter` The pagenumbering is set to arabic numerals.

```
386 \newcommand{\mainmatter}{
387   \if@twoside\cleardoubleoddpagel\else\clearpagel\fi
388   \@mainmattertrue\pagenumbering{arabic}
389 }
```

`\backmatter` The pagenumbering is set to arabic numerals.

```
390 \newcommand{\backmatter}{
391   \if@openright\cleardoubleoddpagel\else\clearpagel\fi
392   \@mainmatterfalse\pagenumbering{roman}
393 }
```

```
394 \</report>
```

### 5.4.2 The Logo for Hochschule Hannover

The following macros are responsible for creating the logo. They load a specific page of a PDF file and display it.

`\HsHlogoPath` This macro contains the path to load the PDF from. It defaults to `HSH-Logo.pdf`, which is provided by this project inside the `scr/` folder.

```
395 \newcommand{\HsHlogoPath}{HSH-Logo.pdf}
```

`\HsHlogoPage` This macro stores the page to use from the PDF. It will be set via the `documentoption` `faculty`.

```
396 \newcommand{\HsHlogoPage}{\HsH@opt@faculty}
```

`\includeHsHlogohere` Calling this macro produces the logo in-place. You can specify the width as an optional argument. The default is 5 cm.

If the file provided via `\HsHlogoPath` doesn't exist, the command will produce an error.

```
397 \newcommand{\includeHsHlogohere}[1][5cm]{
398   \IfFileExists{\HsHlogoPath}{
399     \includegraphics[width=#1,page=\HsHlogoPage]{\HsHlogoPath}
400   }{
401     \ClassError{\HsHClassName}{\HsHlogoPath}space not found!}{
402       The HsH Logo is necessary for the titlepage! Try putting it next to your source file or u
403     }
404   }
405 }
```

### 5.4.3 Title matters

```
406 < *article | report >
```

The following commands relate to the creation of the titlepage. They implement how the user can define the different data fields.

`\title` We redefine the `\title` command to take an optional argument. This is stored in the additional `\@shorttitle` macro.

```
407 \renewcommand{\title}[2][]{
408   \gdef\@title{#2}
409   \gdef\@shorttitle{#1}
410 }
```

`\@shorttitle` This new macro stores a short version of the title. This will be used in places where the full title might overflow the available space.

```
411 \def\@shorttitle{\@empty}
```

`\gdef@with@and@as@comma` First we define a helpermacro that will fill a given macro while treating the `\and` command as a comma. This is needed for `\author` and `\matrikelnr` where we later treat them as comma-separated lists when the titlepage gets constructed.

```
412 \def\gdef@with@and@as@comma#1#2{{\def\and{,}\xdef#1{#2}}}
```

`\author` We redefine the `\author` command so that it sets `\@author`, but allowing for both comma and `\and` to be used as a separator.

```
413 \renewcommand{\author}[1]{\gdef@with@and@as@comma\@author{#1}}
```

The `\@author` macro is also set to `\@empty`, this makes it easier to handle it later.

```
414 \let\@author\@empty
```

`\matrikelnr` These macros set and store the matrikel-number (or set of numbers), which will be printed `\@matrikelnr` on the titlepage.

```
415 \newcommand{\matrikelnr}[1]{\gdef@with@and@as@comma\@matrikelnr{#1}}
```

```
416 \def\@matrikelnr{\@empty}
```

`\professor` These three macros-groups give options to the user to print peoples names on the titlepage, `\firstexaminer` who are relevant to the document, but not the author.

```
\secondexaminer 417 \newcommand{\professor}[1]{\gdef\@professor{#1}}
```

```
\@professor 418 \def\@professor{\@empty}
```

```
\@firstexaminer 419 \newcommand{\firstexaminer}[1]{\gdef\@firstexaminer{#1}}
```

```
\@secondexaminer 420 \def\@firstexaminer{\@empty}
```

```
421 \newcommand{\secondexaminer}[1]{\gdef\@secondexaminer{#1}}
```

```
422 \def\@secondexaminer{\@empty}
```

`\keywords` The macro-group defines and holds keywords which describe the document. They are used `\@keywords` when printing the abstract as well as in the PDF's meta-data.

```
423 \newcommand{\keywords}[1]{\gdef\@keywords{#1}}
```

```
424 \def\@keywords{\@empty}
```

```
425 \end{article} | report
```

#### 5.4.4 Commands for document writing

`\declarationofauthorship` The declaration of authorship is not relevant for the standalone variant.

```
426 \if*!standalone
```

The `pgffor` package is required to handle the loop over the list of authors.

```
427 \RequirePackage{pgffor}
```

Now the command is defined. It takes a optional argument which defaults to `b`.

```
428 \newcommand{\declarationofauthorship}[1][b]{
```

First the argument is passed and an error raised for invalid arguments. Passing in `b` will push the declaration to the bottom of the page and add a horizontal line. Passing `t` simply adds some space.

```
429 \if#1b
```

```
430 \vspace*{\fill}
```

```
431 \hrule
```

```
432 \else\if#1t
```

```
433 \vspace*{2em}
```

```
434 \else
```

```
435 \ClassError{\HsHClassName}{Wrong Parameter for ‘\declarationofauthorship’}{
```

```
436 ‘\string\declarationofauthorship’ only accepts ‘t’ and ‘b’ as parameters.
```

```
437 }
```

```
438 \fi\fi
```

Now the actual declaration can be constructed. It uses the text from `\decofauthname` and `\decofauthtext`.

```
439 \vskip 3em
```

```
440 {\centering\bfseries\usekomafont{section}{\decofauthname}\par}
```

```
441 \vskip 3em
```

```
442 \decofauthtext\par
```



The last step is to loop over all authors by reading \@author and creating a signature box for each one. \thanks also needs to be cleared, as a footnote wouldn't make sense here.

```

443 \beginngroup
444   \renewcommand{\thanks}{\sbox0}
445   \raggedleft
446   \foreach \tmp@ in \@author {
447     \if\tmp@empty\else
448       \hskip 1em \parbox{4cm}{
449         \vskip 4em
450         \hrule\vskip 4pt
451         \raggedleft\footnotesize\tmp@
452       }%
453     \fi
454   }\par
455 \endgroup
456 }

```

\if@singleauthor To ensure \decofauthtext is properly spelled for one or multiple authors, we define a conditional that holds this information. Additionally we check the number of authors \@AtBeginDocument and store it.

```

457 \newif\if@singleauthor
458 \AtBeginDocument{
459   \beginngroup
460     \newcount\count@
461     \count@=\z@
462     \@for\tmp@:=\@author\do{\advance\count@\@ne}
463     \ifnum\count@>\@ne\global\@singleauthorfalse\else\global\@singleauthortrue\fi
464   \endgroup
465 }

```

\ifsingleauthor A macro to simply act on this condition is also provided. It will be used in text-definitions later on.

```

466 \newcommand{\ifsingleauthor}[2]{\if@singleauthor#1\else#2\fi}

```

\declarationAuthorship

```

467 \def\declarationAuthorship{%
468   \ClassWarning{HsHClassName}{%
469     Command \string\declarationAuthorship\space is deprecate.\MessageBreak
470     Replace it with \string\declarationofauthorship.
471   }%
472   \declarationofauthorship%
473 }
474 \!/stadnalone)

```

checklist (env.) Using the enumitem package, we define a helpful environment for creation checklist.

```

475 \newlist{checklist}{itemize}{2}
476 \setlist[checklist]{label=$\square$}

```

### 5.4.5 Miscellaneous commands

#### Utility commands

For writing absolute values, we provide the  $\abs{\langle equ \rangle}$  command, which puts groable, vertical bars on both sides of the equation inside.

```

477 \newcommand{\abs}[1]{\ensuremath{\left\vert\right\vert#1\right\vert}}

```

#### Configuring mathmode-indices

The only hard requirements for documents writing on Hochschule Hannover is, that the indices in mathematic formulas must be typeset in an upright ("steil") font, not the default kursive font. We configure this by first defining a macro to replace the default \sb macro. We can then assign this to `_`. For that to work we need to change its catcode to make it modifiable.

**Note:** You can allways use `\sb` to use the original behaviour for special cases.

```
478 \def\@subinrm#1{\sb{\mathrm{#1}}}  
479 {\catcode'\_ =13 \global\let\_ =\@subinrm}
```

`\upsubscripts` Now we can define a command to activate this new behavior by changing the catcode of `_` to 13, which makes it a normal macro.

```
480 \newcommand\upsubscripts{\catcode'\_ =12}
```

`\normalsubscripts` To switch back we simply need to reset the catcode of `_` back to the original, which makes it a builidin operator with the default behavior.

```
481 \newcommand\normalsubscripts{\catcode'\_ =8}
```

## 5.5 Document setup

The following sets up the look and feel of the documents using this classe. All configuration and stylin is done here.

### 5.5.1 Fonts and text styling

```
482 \RequirePackage{lmodern}
```

### 5.5.2 Page layout

```
483 \<!*standalone>  
484 \areaset[current]{0.75\paperwidth}{0.8\paperheight}  
485 \if@todos  
486 \addtolength\paperwidth{5cm}  
487 \addtolength\marginparwidth{5cm}  
488 \fi  
489 \</!*standalone>
```

### 5.5.3 Styling L<sup>A</sup>T<sub>E</sub>X default constucts

#### Floats

```
490 \<!*standalone>
```

Floats should alwas prefer the *here* placement, than the *top* of the following page.

```
491 \renewcommand{\fps@figure}{h!t}  
492 \renewcommand{\fps@table}{h!t}
```

Floats should be centered by default and the width of the caption box is limited.

```
493 \g@addto@macro\floatboxreset\centering  
494 \setcapwidth{0.8\textwidth}
```

The names of floating enviroments are redefined to show abreviations only.

```
495 \defcaptionname{english}\figurename{Fig.}  
496 \defcaptionname{german,ngerman}\figurename{Abb.}  
497 \defcaptionname{english}\tablename{Tab.}  
498 \defcaptionname{german,ngerman}\tablename{Tab.}
```

For subfigures we need to define a name used in autoreferences.

```
499 \AtBeginDocument{  
500 \makeatletter  
501 \@ifpackageloaded{subfigure}{  
502 \let\subfigureautorefname\figureautorefname  
503 }{}%  
504 \makeatother  
505 }  
506 \</!*standalone>
```

#### Lists

For unordert liste the markers are redefined to look a littel nicer.

```
507 \renewcommand{\labelitemi}{\raisebox{.3ex}{\scalebox{0.7}{\bullet$}}}  
508 \renewcommand{\labelitemii}{\raisebox{.3ex}{\scalebox{0.7}{\circ$}}}  
509 \renewcommand{\labelitemiii}{\raisebox{-.1ex}{\scalebox{1.3}{\cdot$}}}  
510 \renewcommand{\labelitemiv}{\raisebox{.1ex}{-}}
```

## Abstract

We define some custom behavior for the abstract. Its typeset as a quote, so its a little smaller than the rest of the text. We also indent the heading to match that.

```
511 <!*standalone>
512 \renewenvironment{abstract}{
513   \begingroup
514   <article>      \edef\raggedsection{\raggedsection%
515   <report>      \edef\raggedchapter{\raggedchapter%
516     \hskip\leftmargini}
517     \HsH@abstract@heading{\abstractname}
518   \endgroup
519   \quotation
520   \setparsizes{\z@}{\z@}{.25\linewidth \@plus 1fil}\selectfont
521 }{%
522   \ifx\@keywords\@empty\else\if@HsH@option@abstract@show@keywords
523     \par\bigskip
524     \noindent\textbf{\keywordsname}\hskip 2em\@keywords
525   \fi\fi\par
526   \endquotation
527 }
528 </!standalone>
```

## Misc

We activate one of the commands defined above to make math-indices upright by default.

```
529 \upsubscripts
```

We want a ragged bottom instead of spreading the paragraphs over the page.

```
530 \raggedbottom
```

## 5.5.4 Header and footer

```
531 <*article | report>
```

The header and footer are styled using the low-level commands provided by the KOMA-Script package `scrlayer-scrpage`. We also disable the ruler shown in the heading of draft mode.

```
532 \RequirePackage{scrlayer-scrpage}
533 \FamilyOptions{KOMA}{headsepline,singlespacing=true}
534 \let\layercontentsmeasure\relax
```

First we define the new pagestyle `HsHheadings`.

```
535 \newpagestyle{HsHheadings}{
536   {
537     \parbox[b]{\sls@headwidth}{
538       \LaTeXraggedright
539       \ifx\@shorttitle\@empty\@title\else\@shorttitle\fi
540     }%
541   }
542   {
543     \parbox[b]{\sls@headwidth}{
544       \LaTeXraggedleft
545       \leftmark
546     }%
547   }
548   {
549     \parbox[b]{.45\sls@headwidth}{
550       \LaTeXraggedright
551       \ifx\@shorttitle\@empty\@title\else\@shorttitle\fi
552     }%
553     \hfill
554     \parbox[b]{.45\sls@headwidth}{
555       \LaTeXraggedleft
556       \headmark
557     }%
558   }
559 }
```

```

558 }
559 (\textwidth,.1pt)
560 }{
561 {\pagemark}
562 {\hfill\pagemark}
563 {\hfill\pagemark}
564 }

Than all generic settings are applied:
565 \clearpaairofpagestyles
566 \ofoot*{\pagemark}
567 \pagestyle{HsHheadings}
568 \article\automark{section}
569 \report\automark{chapter}
570 \report\renewcommand*{\chapterpagestyle}{HsHheadings}
571 \</article | report>

```

### 5.5.5 Titlepage

`\maketitle` `\maketitle` is a wrapper for `\@maketitle` that just checks the parameter and issues an error message if it is incorrect. It also handles the different default for article or report.

```

572 \<article | report>
573 \article\renewcommand\maketitle[1][c]{
574 \report\renewcommand\maketitle[1][l]{
575 \if#1l\@maketitle{#1}\else%
576 \if#1c\@maketitle{#1}\else%
577 \if#1r\@maketitle{#1}\else%
578 \ClassWarning{HsHClassName}{unkown option for \maketitle}
579 \thispagestyle{empty}\null\clearpage
580 \fi\fi\fi
581 }

```

`\@maketitle` The definition of `\@maketitle` is mostly taken from the source-code of the KOMA-Script parentclass, but was modified to create the desired style.

```

582 \newtoks\@tabtoks
583 \newcommand\addtabtoks[1]{\global\@tabtoks\expandafter{\the\@tabtoks#1}}
584 \newcommand\eadddtabtoks[1]{\edef\mytmp{#1}\expandafter\addtabtoks\expandafter{\mytmp}}
585 % %%\newcommand*\resettabtoks{\global\@tabtoks{}}
586 \newcommand*\printtabtoks{\the\@tabtoks}
587 \addtokomafont{publishers}{\normalsize}
588 \g@addto@macro\titlepage{\singlespacing}
589 %
590 \renewcommand*\@maketitle[1]{
591 \expandafter\ifnum \curname <scr@v@3.12\endcsname>\scr@compatibility\relax
592 \else
593 \def\and{%
594 \end{tabular}
595 \hspace{1em} \@plus.17fil
596 \begin{tabular}[t]{c}%
597 }
598 \fi
599 \<article>
600 \par
601 \ifx\@uppertitleback\@empty\else
602 \ClassWarning{KOMAClassName}{%
603 non empty \string\uppertitleback\space ignored
604 by \string\maketitle\MessageBreak
605 in 'titlepage=false' mode%
606 }
607 \fi
608 \ifx\@lowertitleback\@empty\else
609 \ClassWarning{KOMAClassName}{%
610 non empty \string\lowertitleback\space ignored
611 by \string\maketitle\MessageBreak

```

```

612      in 'titlepage=false' mode%
613    }
614  \fi
615 </article>
616 <report> \begin{titlepage}
617 <article> \begin{group
618 <*report>
619   \if@titlepageiscoverpage
620     \edef\titlepage@restore{
621       \noexpand\endgroup
622       \noexpand\global\noexpand\@colht\the\@colht
623       \noexpand\global\noexpand\@colroom\the\@colroom
624       \noexpand\global\vsizethe\vsizethe
625       \noexpand\global\noexpand\@titlepageiscoverpagefalse
626       \noexpand\let\noexpand\titlepage@restore\noexpand\relax
627     }
628     \begin{group
629       \topmargin=\dimexpr \coverpagetopmargin-1in\relax
630       \oddsidemargin=\dimexpr \coverpageleftmargin-1in\relax
631       \evensidemargin=\dimexpr \coverpageleftmargin-1in\relax
632       \textwidth=\dimexpr
633       \paperwidth-\coverpageleftmargin-\coverpagerightmargin\relax
634       \textheight=\dimexpr
635       \paperheight-\coverpagetopmargin-\coverpagebottommargin\relax
636       \headheight=0pt
637       \headsep=0pt
638       \footskip=\baselineskip
639       \@colht=\textheight
640       \@colroom=\textheight
641       \vsizethe\textheight
642       \columnwidth=\textwidth
643       \hsize=\columnwidth
644       \linewidth=\hsize
645     \else
646       \let\titlepage@restore\relax
647     \fi
648     \let\footnotesize\small
649     \let\footnoterule\relax
650     \let\footnote\thanks
651 </report>
652 <article> \let\titlepage@restore\relax
653   \renewcommand*\thefootnote{\@fnsymbol\c@footnote}%
654   \let\@oldmakefnmark\@makefnmark
655   \renewcommand*\@makefnmark{\rlap{\@oldmakefnmark}}%
656 <article> \next@tdpage
657   \ifx\@extratitle\@empty
658 <article> \ifx\@frontispiece\@empty\else \mbox{}\fi
659 <*report>
660   \ifx\@frontispiece\@empty\else
661     \if@twoside\mbox{}\next@tdpage\fi
662     \noindent\@frontispiece\next@tdpage
663   \fi
664 </report>
665   \else
666 <article> \@makeextratitle
667 <*report>
668   \noindent\@extratitle
669   \ifx\@frontispiece\@empty
670   \else
671     \next@tdpage
672     \noindent\@frontispiece
673   \fi
674   \next@tdpage

```

```

675 </report>
676 \fi
677 <*article>
678 \ifx\@frontispiece\@empty
679 \ifx\@extratitle\@empty\else\next@tdpage\fi
680 \else
681 \next@tpage
682 \@makefrontispiece
683 \next@tdpage
684 \fi
685 \global\@topnum=\z@
686 </article>
687 \setparsizes{\z@}{\z@}{\z@\@plus 1fil}\par@updaterelative
688 \vspace*{1cm}
689 \begin{minipage}[t]{\textwidth}%
690 \ifx\@titlehead\@empty \else
691 \usekomafont{titlehead}{\@titlehead}%
692 \fi
693 \hfill
694 % image with referencepoint in lower left corner:
695 \raisebox{0pt}[\ht\strutbox][\dp\strutbox]{\includeHsHlogohere}
696 \end{minipage}
697 \raisebox{10pt}{\rule{\textwidth}{0.5pt}}
698 \null
699 <article> \vskip 2em
700 <report> \vfill
701 \begin{group}
702 \if#1c\centering\fi
703 \if#1r\raggedleft\fi
704 \ifx\@subject\@empty\else
705 {\usekomafont{subject}{\@subject}\par}
706 <article> \vskip 1.5em
707 <report> \vskip 3em
708 \fi
709 {\usekomafont{title}{\huge\@title}\par}
710 <article> \vskip .5em
711 <report> \vskip 1em
712 {\ifx\@subtitle\@empty\else\usekomafont{subtitle}{\@subtitle}\par\fi}
713 <article|report> \vskip 4em
714 {\ifx\@matrikelnr\@empty
715 \if\@author\@empty\else\usekomafont{author}{
716 \parbox{\dimexpr\linewidth}{
717 \if#1c\centering\fi
718 \if#1r\raggedleft\fi
719 \@author
720 }
721 }\fi
722 \else
723 \if\@author\@empty\else
724 % sneaky comma needed after \@matrikelnr to deal with single item lists
725 \foreach \x [count=\i,evaluate=\i as \y using {\@matrikelnr,\i-1}] in \@author {
726 \usekomafont{author}{
727 \def\arraystretch{1.2}
728 \if#1l\begin{tabular}{@{}l l}\fi
729 \if#1c\begin{tabular}{l l}\fi
730 \if#1r\begin{tabular}{r r@{}}\fi
731 \printtabtoks
732 \end{tabular}
733 }%
734 \fi
735 \fi}
736 <article> \vskip 1em
737 <report> \vskip 1.5em

```

```

738     {\usekomafont{date}{\@date\par}}
739 <article>      \vskip 1em
740 <report>      \vskip \z@ \@plus3fill
741     \usekomafont{publishers}{
742         \def\arraystretch{1.2}
743         \if#1l\begin{tabular}{@{}l l}\fi
744         \if#1c\begin{tabular}{l l}\fi
745         \if#1r\begin{tabular}{r r@{}}\fi
746         \if\@professor\@empty\else\textbf{\professorname:}&\@professor\\\fi
747         \if\@firstexaminer\@empty\else\textbf{\firstexaminername:}&\@firstexaminer\\\fi
748         \if\@secondexaminer\@empty\else\textbf{\secondexaminername:}&\@secondexaminer\\\fi
749     \end{tabular}
750 }
751 <*article>
752     \ifx\@dedication\@empty\else
753         \vskip 2em
754         {\usekomafont{dedication}{\@dedication \par}}%
755     \fi
756 </article>
757     \par
758     \endgroup
759 <article>      \vskip 2em
760 <report>      \vskip 3em
761 <article>      \ifx\titlepagestyle\@empty\else\thispagestyle{\titlepagestyle}\fi
762     \@thanks\global\let\@thanks\@empty
763 <*report>
764     \vfill\null
765     \if@twoside
766         \@tempswatrue
767         \expandafter\ifnum \@nameuse{scr@v@3.12}>\scr@compatibility\relax
768         \else
769             \ifx\@uppertitleback\@empty\ifx\@lowertitleback\@empty
770                 \@tempswafalse
771             \fi\fi
772         \fi
773         \if@tempswa
774             \next@tpage
775             \begin{minipage}[t]{\textwidth}
776                 \@uppertitleback
777             \end{minipage}\par
778             \vfill
779             \begin{minipage}[b]{\textwidth}
780                 \@lowertitleback
781             \end{minipage}\par
782             \@thanks\global\let\@thanks\@empty
783         \fi
784     \else
785         \ifx\@uppertitleback\@empty\else
786             \ClassWarning{\KOMAClassName}{%
787                 non empty \string\uppertitleback\space ignored
788                 by \string\maketitle\MessageBreak
789                 in 'twoside=false' mode%
790             }
791         \fi
792         \ifx\@lowertitleback\@empty\else
793             \ClassWarning{\KOMAClassName}{%
794                 non empty \string\lowertitleback\space ignored
795                 by \string\maketitle\MessageBreak
796                 in 'twoside=false' mode%
797             }
798         \fi
799     \fi
800     \ifx\@dedication\@empty

```

```

801 \else
802 \next@tdpage\null\vfill
803 {\centering\usekomafont{dedication}{\@dedication \par}}%
804 \vskip \z@ \@plus3fill
805 \@thanks\global\let\@thanks\@empty
806 \cleardoubleemptypage
807 \fi
808 \ifx\titlepage@restore\relax\else\clearpage\titlepage@restore\fi
809 \end{report}
810 \end{article} \endgroup
811 \end{report} \end{titlepage}
812 \setcounter{footnote}{0}
813 \expandafter\ifnum \csname scr@v@3.12\endcsname>\scr@compatibility\relax
814 \let\@thanks\relax
815 \let\maketitle\relax
816 \let\@maketitle\relax
817 \global\let\@thanks\@empty
818 \global\let\@author\@empty
819 \global\let\@date\@empty
820 \global\let\@title\@empty
821 \global\let\@subtitle\@empty
822 \global\let\@extratitle\@empty
823 \global\let\@frontispiece\@empty
824 \global\let\@titlehead\@empty
825 \global\let\@subject\@empty
826 \global\let\@publishers\@empty
827 \global\let\@uppertitleback\@empty
828 \global\let\@lowertitleback\@empty
829 \global\let\@dedication\@empty
830 \global\let\@matrikelnr\@empty
831 \global\let\@professor\@empty
832 \global\let\author\relax
833 \global\let\title\relax
834 \global\let\extratitle\relax
835 \global\let\titlehead\relax
836 \global\let\subject\relax
837 \global\let\publishers\relax
838 \global\let\uppertitleback\relax
839 \global\let\lowertitleback\relax
840 \global\let\dedication\relax
841 \global\let\date\relax
842 \global\let\matrikelnr\relax
843 \global\let\professor\relax
844 \fi
845 \global\let\and\relax
846 }
847 \end{article} | report)

```

## 5.6 Localisation

The following section contains language specific definitions of text used by the classes.

`\abstractname` The English word "Abstract" is preferred over the German translation.

```
848 \defcaptionname{german,ngerman}\abstractname{Abstract}
```

`\professorname` Define the commands content for the different supported languages.

```

\firstexaminername 849 \newcaptionname{english}\professorname{Professor}
\secondexaminername 850 \newcaptionname{german,ngerman}\professorname{Professor(in)/Lehrbeauftragte(r)}
851 \newcaptionname{english}\firstexaminername{First examiner}
852 \newcaptionname{german,ngerman}\firstexaminername{Erstpr{"u}fer(in)}
853 \newcaptionname{english}\secondexaminername{Second examiner}
854 \newcaptionname{german,ngerman}\secondexaminername{Zweitpr{"u}fer(in)}

```



`\decofauthname` Define the german translations for the command.

```
855 \newcaptionname{english}\decofauthname{Declaration of Authorship}
856 \newcaptionname{german,ngerman}\decofauthname{Selbstst{\a}ndigkeitserkl{\a}rung}
```

`\decofauthtext` Define the german translations for the command.

```
857 \newcaptionname{english}\decofauthtext{%
858   \ifsingleauthor{I}{We} hereby certify that the work \ifsingleauthor{I}{we}
859   \ifsingleauthor{am}{are} submitting is entirely of \ifsingleauthor{my}{our}
860   own making except where otherwise indicated. \ifsingleauthor{I}{We}
861   \ifsingleauthor{am}{are} aware of regulations concerning plagiarism,
862   including disciplinary actions that may result from it. Any use of the
863   works of any other author, in any form, is properly acknowledged at
864   their point of use.
865 }
866 \newcaptionname{german,ngerman}\decofauthtext{%
867   Hiermit best{\a}tige\ifsingleauthor{}{n} \ifsingleauthor{ich}{wir},
868   dass die folgende Arbeit eigenst{\a}ndig von \ifsingleauthor{mir}{uns}
869   allein erstellt und unter Ber{\u}cksichtigung der zur Verf{\u}gung
870   gestellten Aufgabenstellung sowie dem Arbeitsmaterial unter Angabe aller
871   verwendeten Quellen erarbeitet wurde. Die Regelungen und Konsequenzen
872   eines Plagiats, inklusive disziplinarischer Ma{\ss}nahmen, sind
873   \ifsingleauthor{mir}{uns} bewusst. Insbesondere wurden alle Zitate und
874   gedanklichen {\u}bernahmen als solche kenntlich gemacht.
875 }
```

`\keywordsname` Define the german translations for the command.

```
876 \newcaptionname{english}\keywordsname{Keywords}
877 \newcaptionname{german,ngerman}\keywordsname{Schl{\u}sselw{\o}rter}
```