

# Task Report v0.5

Geraldo Xexéo

Wednesday 2<sup>nd</sup> June, 2021 - 14:06

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>How to use task-report</b>	<b>2</b>
2.1	Using from Overleaf . . . . .	2
<b>3</b>	<b>Package Options</b>	<b>2</b>
3.1	General behavior . . . . .	3
<b>4</b>	<b>Version</b>	<b>3</b>
<b>5</b>	<b>Usage</b>	<b>3</b>
<b>6</b>	<b>Configuration</b>	<b>6</b>
6.1	Configuring Colors . . . . .	6
6.2	Changing Field Names . . . . .	8
6.3	Modifying the Header . . . . .	9
6.4	Creating New States . . . . .	10
<b>7</b>	<b>Using Labels to Refer to Tasks</b>	<b>11</b>
<b>8</b>	<b>The List of Tasks</b>	<b>11</b>
<b>9</b>	<b>Warnings</b>	<b>11</b>
<b>10</b>	<b>Implementation</b>	<b>12</b>
10.1	Options . . . . .	12
10.2	Required Packages . . . . .	12
10.3	Access to the current version . . . . .	12
10.4	Color Management for States . . . . .	12
10.5	Defining States Through a Simulated List . . . . .	12
10.6	Standard Colors . . . . .	13
10.7	Managing The Fields Names . . . . .	13

10.8 Standard Field Names . . . . .	14
<b>11 List of Tasks</b>	<b>15</b>
11.1 A Counter for Tasks . . . . .	15
11.2 Declaring Tasks . . . . .	15
11.2.1 Helpers . . . . .	15
11.2.2 Main task macro . . . . .	16

## 1 Introduction

This package supports the maintenance and report of a task list..

## 2 How to use task-report

`task-report` is available as open source at <https://github.com/xexeo/task-report>. The stable distribution is in the folder `dist`, while the latest, and unstable, will be in the root folder.

The only file you really need, besides this manual, is `task-report.sty`. This should be in your L<sup>A</sup>T<sub>E</sub>X path, such as in the same folder that your main `.tex` file.

You can make suggestions, complain about bugs, and request features using GitHub’s “Issues” feature, in <https://github.com/xexeo/task-report/issues> (you must be signed in).

### 2.1 Using from Overleaf

The best way to use `task-report` inside Overleaf is to link the distributed style file through its URL. To do that, inside your project, select first “Upload”. An “Add files” window will appear, then select “From External URL” and enter <https://raw.githubusercontent.com/xexeo/task-report/main/dist/task-report.sty> as “URL to fetch the file from” and “task-report” as “File Name in This Project”, as in Figure 1.

## 3 Package Options

`showtasknumbers` show tasks numbers, default is false.

Users should never refer to a task by its number, since it can change from compilation to compilation if any task is inserted in among the existing ones. It is possible to use the `\label` macro to refer to tasks by number, page ou name.

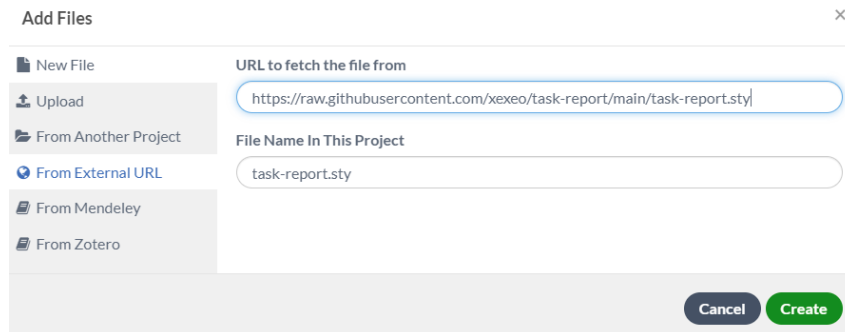


Figure 1: How to link a task-report.sty file in your project folder in Overleaf to the official distribution.

### 3.1 General behavior

## 4 Version

It is possible that the user wants to know the version being used. We provide two commands for it.

`\taskreportversion` Provides the current version  
`\printtaskreportversion` Provides name and version

`\taskreportversion` was used in the title of this article. The current version is v0.5.

If needed, the second macro prints also the name of the package:

`\printtaskreportversion`

will result in:

task-report v0.5

Please state the version when reporting bugs.

## 5 Usage

`\trtask` This is the main macro for the package. `\trtask[<status>]{<name>}{<due date>}{<when-assigned>}{<who-assigned>}{<to-who-assigned>}{<why>}{<when-done>}{<extra-text>}`

Although it has 8 arguments parameters, all of them can be empty. It has 9 arguments because this is the standard L<sup>A</sup>T<sub>E</sub>X maximum.

There are 5 possible standard states that are associated with a color (*de-*

*fault color*): todo (red), doing(yellow), tocheck (cyan), checking (blue), and done(green). There is a pseudo-state other (black) that will be the color of any other word used of a state. Colors belong to the `xcolor` package and are configurable, as explained in section 6.1.

The example:

```
%\trtask[todo]{Make coffe}{Soon}{\today}{Geraldo Xexéo}{}{}{}{Please do the coffe strong}
%\trtask[doing]{Drink a coffe}{In 5 days}{June, 1st 2021}{The Boss}{The Worker}{}{}{}
%\trtask[tocheck]{Roast the beef}{}{June, 1st 2021}{The Boss}{The Worker}{}{\today}{Rare}
%\trtask[checking]{Drink another beer}{}{June, 1st 2021}{The Boss}{The Worker}{}{\today}
%\trtask[done]{Drink a beer}{}{\today}{The Boss}{The Worker}{Because I want}{\today}{}
%\trtask{}{}{}{}{}{}{}{}
%\trtask[done]{}{}{}{}{}{}{}{}
%\trtask[arg 1]{arg 2}{arg 3}{arg 4}{arg 5}{arg 6}{arg 7}{arg 8}{arg 9}
%
```

generates:

**Make coffe**

**Status:** todo  
**Due date:** Soon  
**Assigned in:** Wednesday 2<sup>nd</sup> June, 2021  
**From:** Geraldo Xexéo

---

Please do the coffe strong

**Drink a coffe**

**Status:** doing  
**Due date:** In 5 days  
**Assigned in:** June, 1st 2021  
**From:** The Boss  
**To:** The Worker

### Roast the beef

**Status:** tocheck  
**Assigned in:** June, 1st 2021  
**From:** The Boss  
**To:** The Worker  
**Done in:** Wednesday 2<sup>nd</sup> June, 2021

---

Rare, please

### Drink another beer

**Status:** checking  
**Assigned in:** June, 1st 2021  
**From:** The Boss  
**To:** The Worker  
**Done in:** Wednesday 2<sup>nd</sup> June, 2021

---

Please check if you have age to drink in your country. If not, drink orange-juice.

### Drink a beer

**Status:** done  
**Assigned in:** Wednesday 2<sup>nd</sup> June, 2021  
**From:** The Boss  
**To:** The Worker  
**Rationale:** Because I want  
**Done in:** Wednesday 2<sup>nd</sup> June, 2021

### Task without a name

### Task without a name

**Status:** done

arg 2
<b>Status:</b> arg 1 <b>Due date:</b> arg 3 <b>Assigned in:</b> arg 4 <b>From:</b> arg 5 <b>To:</b> arg 6 <b>Rationale:</b> arg 7 <b>Done in:</b> arg 8
-----
arg 9

## 6 Configuration

### 6.1 Configuring Colors

`\trsetboxcolors` Set the colors for the 6 types of standard states Black otherwise.

```
\trsetboxcolors{\todo-color}{\doing-color}{\tocheck-color}{\checking-color}
{\done-color}{\other-color}
```

The example:

```
%\trsetboxcolors{magenta}{purple}{orange}{pink}{lime}{pink}
%
%\trtask[todo]{Drink a coffe}{next day}{\today}{The Boss}{The Worker}%
%{To keep awake}{\today}{Coffe must be strong and add 1 spoon of sugar}
%\trtask[doing]{Drink a coffe}{next day}{\today}{The Boss}{The Worker}%
%{To keep awake}{\today}{Coffe must be strong and add 1 spoon of sugar}
%\trtask[tocheck]{Drink a coffe}{next day}{\today}{The Boss}{The Worker}%
%{To keep awake}{\today}{Coffe must be strong and add 1 spoon of sugar}
%\trtask[checking]{Drink a coffe}{next day}{\today}{The Boss}{The Worker}%
%{To keep awake}{\today}{Coffe must be strong and add 1 spoon of sugar}
%\trtask[done]{Drink a coffe}{next day}{\today}{The Boss}{The Worker}%
%{To keep awake}{\today}{Coffe must be strong and add 1 spoon of sugar}
%\trtask{Drink a coffe}{\today}{next day}{The Boss}{The Worker}%
%{To keep awake}{\today}{Coffe must be strong and add 1 spoon of sugar}
%
```

generates:

### Drink a coffe

**Status:** todo  
**Due date:** next day  
**Assigned in:** Wednesday 2<sup>nd</sup> June, 2021  
**From:** The Boss  
**To:** The Worker  
**Rationale:** To keep awake  
**Done in:** Wednesday 2<sup>nd</sup> June, 2021

---

Coffe must be strong and add 1 spoon of sugar

### Drink a coffe

**Status:** doing  
**Due date:** next day  
**Assigned in:** Wednesday 2<sup>nd</sup> June, 2021  
**From:** The Boss  
**To:** The Worker  
**Rationale:** To keep awake  
**Done in:** Wednesday 2<sup>nd</sup> June, 2021

---

Coffe must be strong and add 1 spoon of sugar

### Drink a coffe

**Status:** tocheck  
**Due date:** next day  
**Assigned in:** Wednesday 2<sup>nd</sup> June, 2021  
**From:** The Boss  
**To:** The Worker  
**Rationale:** To keep awake  
**Done in:** Wednesday 2<sup>nd</sup> June, 2021

---

Coffe must be strong and add 1 spoon of sugar

### Drink a coffe

**Status:** checking  
**Due date:** next day  
**Assigned in:** Wednesday 2<sup>nd</sup> June, 2021  
**From:** The Boss  
**To:** The Worker  
**Rationale:** To keep awake  
**Done in:** Wednesday 2<sup>nd</sup> June, 2021

---

Coffe must be strong and add 1 spoon of sugar

### Drink a coffe

**Status:** done  
**Due date:** next day  
**Assigned in:** Wednesday 2<sup>nd</sup> June, 2021  
**From:** The Boss  
**To:** The Worker  
**Rationale:** To keep awake  
**Done in:** Wednesday 2<sup>nd</sup> June, 2021

---

Coffe must be strong and add 1 spoon of sugar

### Drink a coffe

**Due date:** Wednesday 2<sup>nd</sup> June, 2021  
**Assigned in:** next day  
**From:** The Boss  
**To:** The Worker  
**Rationale:** To keep awake  
**Done in:** Wednesday 2<sup>nd</sup> June, 2021

---

Coffe must be strong and add 1 spoon of sugar

## 6.2 Changing Field Names

`\trsetfieldsnames` You can change all field names with command `\trsetfieldsnames`. This command can be used to internationalization or to make these box used by any other reason.

`\trsetfieldsnames{\langle status \rangle}{\langle due date \rangle}{\langle assigned \rangle}{\langle from \rangle}{\langle to \rangle}{\langle done \rangle}{\langle why \rangle}`



For example:

```
%\trsetfieldsnames{Stage}{Must be ready in}{Demand date}{Who asked for}{To who}{Ready in}
%\trtask[done]{Drink a coffe}{5 days}{\today}{The Boss}{The Worker}%
%{To keep awake}{\today}{Coffe must be strong and add 1 spoon of sugar}
%
```

generates:

**Drink a coffe**  
**Stage:** done  
**Must be ready in:** 5 days  
**Demand date:** Wednesday 2<sup>nd</sup> June, 2021  
**Who asked for:** The Boss  
**To who:** The Worker  
**Reasons:** To keep awake  
**Ready in:** Wednesday 2<sup>nd</sup> June, 2021  

---

**Coffe must be strong and add 1 spoon of sugar**

### 6.3 Modifying the Header

```
\trsettitlelabel        Sets the label for the header.
                      \trsettitlelabel{\<label-for-header>}

% \trsettitlelabel{Task}
%
%\trtask[done]{Drink a coffe}{5 days}{\today}{The Boss}{The Worker}
%{To keep awake}{\today}{Coffe must be strong and add 1 spoon of sugar}
%
%
```

#### Task: Drink a coffe

**Status:** done  
**Due date:** 5 days  
**Assigned in:** Wednesday 2<sup>nd</sup> June, 2021  
**From:** The Boss  
**To:** The Worker  
**Rationale:** To keep awake  
**Done in:** Wednesday 2<sup>nd</sup> June, 2021

---

Coffe must be strong and add 1 spoon of sugar

Back to normal:

```
% \trsettitlelabel{}  
%  
%\trtask[done]{Drink a coffe}{5 days}{\today}{The Boss}{The Worker}  
%{To keep awake}{\today}{Coffe must be strong and add 1 spoon of sugar}  
%
```

#### Drink a coffe

**Status:** done  
**Due date:** 5 days  
**Assigned in:** Wednesday 2<sup>nd</sup> June, 2021  
**From:** The Boss  
**To:** The Worker  
**Rationale:** To keep awake  
**Done in:** Wednesday 2<sup>nd</sup> June, 2021

---

Coffe must be strong and add 1 spoon of sugar

## 6.4 Creating New States

`\traddstatecolor` Users can create their own state using  
`\traddstatecolor{<state>}{<color>}`

For example:

```
%\traddstatecolor{waiting}{teal}  
%\trtask[waiting]{Drink a coffe}{\today}{The Boss}{The Worker}%  
%{To keep awake}{\today}{Coffe must be strong and add 1 spoon of sugar}  
%
```

generates:

**Drink a coffe**  
**Status:** waiting  
**Due date:** ASAP  
**Assigned in:** Wednesday 2<sup>nd</sup> June, 2021  
**From:** The Boss  
**To:** The Worker  
**Rationale:** To keep awake  
**Done in:** Wednesday 2<sup>nd</sup> June, 2021  

---

Coffe must be strong and add 1 spoon of sugar

## 7 Using Labels to Refer to Tasks

You can use labels, such as `\label{task:kanbam}` in the last argument<sup>1</sup> and refer to it with commands `\ref{task:kanbam}` (19), `\pageref{task:kanbam}` (16) ou `\nameref{task:kanbam}` (Make a L<sup>A</sup>T<sub>E</sub>X Kanbam document).

## 8 The List of Tasks

`\listoftasks` A list of tasks can be generated with the command:

`\listoftasks`

Its title can be changed with the command:

`\trsettasklisttitle{<new-title>}`

The command:

`\setlength{\cfttasksnumwidth}{25pt}`

from package `tocloft` can be used to set the space used by the section indicator.

## 9 Warnings

This package uses another package that changes L<sup>A</sup>T<sub>E</sub>X's standard behavior for summary and lists. When you use it, you must explicitly change pages with `\newpage` before `\tableofcontents` or similar commands.

---

<sup>1</sup>Actually, they will probably break only if used in the second argument, that is used to name them

## 10 Implementation

### 10.1 Options

`showtasknumbers`

Since task numbers can change between compilations this default is set to `false`.

```
1 \newif\if@showtasknumbers\@showtasknumbersfalse%
2 %
3 \DeclareOption{showtasknumbers}{\@showtasknumberstrue}%
4 %
5 \ProcessOptions\relax
```

### 10.2 Required Packages

- `mdframed` makes the frame around tasks
- `etoolbox` command `ifstringempty`
- `tocloft` create the list of tasks

```
6 \RequirePackage{tcolorbox}%
7 \RequirePackage{etoolbox}%
8 \RequirePackage{tocloft}%
9 \RequirePackage{hyperref}%
10
```

### 10.3 Access to the current version

We provide some macros for the user to know the version being used.

`\taskreportversion` Provides current version of coop-writing

```
11 \newcommand{\taskreportversion}{\tr@version}%
```

`\printtaskreportversion` Provides package name and version

```
12 \newcommand{\printtaskreportversion}{task-report \taskreportversion}%
```

### 10.4 Color Management for States

### 10.5 Defining States Through a Simulated List

`\traddstatecolor` Inserts states and their colors in a simulated list and uses them  
`\usestatecolor`

```

13 \newcommand{\traddstatecolor}[2]{%
14     \@ifundefined{\string\color@#2}%
15     {%
16         \message{#2 not defined}%
17         \expandafter\ERROR%
18     }%
19     {%
20         \expandafter\gdef\csname tr@astate@\detokenize{#1}\endcsname{#2}%
21     }%
22 }%
23 %
24 \def\tr@usestatecolor#1{%
25     \ifcsname tr@astate@\detokenize{#1}\endcsname%
26     \csname tr@astate@\detokenize{#1}\expandafter\endcsname%
27     \else%
28 \tr@astate@other%
29     \fi%
30 }%

```

`\trsetboxcolors` Set the colors for the 6 types of standard states Black otherwise.

```

\trsetboxcolors{\<todo-color>}{\<doing-color>}{\<tocheck-color>}{\<checking-color>}
{\<done-color>}{\<other-color>}

```

```

31 \newcommand{\trsetboxcolors}[6]{%
32 \traddstatecolor{todo}{#1}
33 \traddstatecolor{doing}{#2}
34 \traddstatecolor{tocheck}{#3}
35 \traddstatecolor{checking}{#4}
36 \traddstatecolor{done}{#5}
37 \traddstatecolor{other}{#6}
38 }%

```

## 10.6 Standard Colors

Sets the standard colors for the standard states

```

39 \trsetboxcolors{red}{yellow}{cyan}{blue}{green}{black}

```

## 10.7 Managing The Fields Names

`\traddfieldname` Define field names - does not create a new field  
`\tr@usesfieldname`

```

40 \newcommand{\traddfieldname}[2]{%
41     \expandafter\gdef\csname tr@fieldnames@\detokenize{#1}\endcsname{#2}%
42 }%
43 %
44 \def\tr@usesfieldname#1{%

```

```

45 \ifcsname tr@fieldnames@\detokenize{#1}\endcsname%
46 \csname tr@fieldnames@\detokenize{#1}\expandafter\endcsname%
47 \else%
48 Missing Field Name%
49 \fi%
50 }%

```

## 10.8 Standard Field Names

Sets the standard text for the fields

```

\trsetfieldsnames Set the names for the field names and for the titles.
\trsettitlelabel  \trsetfieldsnames{\<status>}{\<duedate>}{\<assigned>}{\<from>}{\<to>}{\<done>}{\<why>}
                  \trsettitlelabel{\<title-label>}

51 \newcommand{\trsetfieldsnames}[7]{%
52   \traddfieldname{status}{#1}%
53   \traddfieldname{duedate}{#2}%
54   \traddfieldname{assigned}{#3}%
55   \traddfieldname{from}{#4}%
56   \traddfieldname{to}{#5}%
57   \traddfieldname{done}{#6}%
58   \traddfieldname{why}{#7}%
59 }%
60 %
61 \traddfieldname{defaulttitle}{Task without a name}%
62 %
63 \def\tr@defaulttitlelabel{}%
64 \newcommand{\trsettitlelabel}[1]{%
65   \def\tr@defaulttitlelabel{#1}%
66 }%
67 %
68 \trsetfieldsnames{Status}{Due date}{Assigned in}{From}{To}{Done in}{Rationale}
69 %

\trsetcolon

70 \newcommand{\trsetcolon}[1]{%
71   \traddfieldname{colonmark}{#1}
72 }
73 \trsetcolon{:}

\tr@currentboxcolor The dirt trick of using a global variable to store a value

74 \def\tr@currentboxcolor{\tr@usestatecolor{other}}%

\trstatuscolor Select the color based on the state

```

```

\trstatuscolor{\state}

75 \newcommand{\trstatuscolor}[1]{%
76 {%
77 \ifblank{#1}%
78 {%
79 \def\tr@currentboxcolor{\tr@usestatecolor{other}}%
80 }%
81 {%
82 \def\tr@currentboxcolor{\tr@usestatecolor{#1}}%
83 }%
84 }%

```

## 11 List of Tasks

```

85 \newcommand{\trsettasklisttitle}[1]{%
86   \def\trlisttasks{#1}%
87 }%
88 \trsettasklisttitle{List of Tasks}%
89 %% cria a lista, depende do pacote tcloft
90 \@ifundefined{chapter}%
91 {\newlistof[section]{tasks}{trt}{\trlisttasks}}%
92 {\newlistof[chapter]{tasks}{trt}{\trlisttasks}}%

```

### 11.1 A Counter for Tasks

### 11.2 Declaring Tasks

#### 11.2.1 Helpers

`\tr@field` Helpers to build the task `\trtitle` should be improved to check if `\tr@defaulttitlelabel` is not empty and use it

```

93 \newcommand{\tr@title}[2]{%
94 \ifx\tr@defaulttitlelabel\empty%
95   \if@showtasknumbers%
96     \textbf{#2\tr@usesfieldname{colonmark} #1}%
97   \else%
98     \textbf{#1}%
99   \fi%
100 \else%
101   \if@showtasknumbers%
102     \textbf{\tr@defaulttitlelabel\ #2\tr@usesfieldname{colonmark}\ #1}%
103   \else%
104     \textbf{\tr@defaulttitlelabel\tr@usesfieldname{colonmark}\ #1}%
105   \fi%
106 \fi%

```

```

107 }%
108 %
109 \newcommand{\tr@field}[2]{%
110 \textbf{#1}\tr@usesfieldname{colonmark} #2%
111 }%
112 %

```

### 11.2.2 Main task macro

`\trtask` Describes a task

```

\trtask[<status>]{<name>}{<due-date>}{<when-assigned>}{<who-assigned>}
{<to-who-assigned>}{<why>}{<when-done>}{<extra-text>}

```

For example:

```

%\trtask[doing]{Make a \LaTeX\ Kanbam document}{soon}
% {2021-05-31}{Xexéo}{Geraldo Xexéo}{Help control tasks}
% {not yet}{Use colors\label{task:kanbam}}
%

```

generates:

**Make a  $\text{\LaTeX}$  Kanbam document**

**Status:** doing  
**Due date:** soon  
**Assigned in:** 2021-05-31  
**From:** Xexéo  
**To:** Geraldo Xexéo  
**Rationale:** Help control tasks  
**Done in:** not yet

---

Use colors

It is possible to reference the task (as Task 19 in page 16, named Make a  $\text{\LaTeX}$  Kanbam document).

```

113 % [status] , what, due-date when-assigned, from-who, to-who, why, when-done, extra
114 \newcounter{taskcounter}
115 %
116 \newcommand{\trtask}[9][ ]{%
117   \par%
118   \ifblank{#2}{
119     \def\trtask@temp{\tr@usesfieldname{defaulttitle}}%
120   }%
121   {%

```



```

122     \def\trtask@temp{#2}%
123     }%
124     \refstepcounter{tasks}%
125 % \refstepcounter updates \@currentlabel (\LaTeX magic!)
126 % therefore the next command must appear before #9, where label must be.
127     \refstepcounter{taskcounter}%
128     \protected@edef\@currentlabelname{\trtask@temp}% trying to make it work
129     \addcontentsline{trt}{tasks}{\protect\numberline{\thetasks}{\trtask@temp}}%
130     \trstatuscolor{#1}%
131     \begin{tcolorbox}[title=\tr@title{\trtask@temp}{\thetaskcounter},%
132 colback=\tr@currentboxcolor!5!white,%
133 colframe=\tr@currentboxcolor!75!black%
134 ]%
135         \ifblank{#1}{}%
136         {%
137             \tr@field{\tr@usesfieldname{status}}{#1}\newline%
138         }%
139         \ifblank{#3}{}%
140     {%
141         \tr@field{\tr@usesfieldname{duedate}}{#3}\newline%
142     }%
143     \ifblank{#4}{}%
144     {%
145         \tr@field{\tr@usesfieldname{assigned}}{#4}\newline%
146     }%
147     \ifblank{#5}{}%
148     {%
149         \tr@field{\tr@usesfieldname{from}}{#5}\newline%
150     }%
151     \ifblank{#6}{}%
152     {%
153         \tr@field{\tr@usesfieldname{to}}{#6}\newline%
154     }%
155     \ifblank{#7}{}%
156     {%
157         \tr@field{\tr@usesfieldname{why}}{#7}\newline%
158     }%
159     \ifblank{#8}{}%
160     {%
161         \tr@field{\tr@usesfieldname{done}}{#8}\newline%
162     }%
163         \ifblank{#9}{}%
164         {\tcblower%
165             #9%
166         }%
167     \end{tcolorbox}%
168 }%

```

End of Code

## Change History

v0.1		v0.4
General: First version . . . . .	1	General: Allows redefining field names . . . . . 1
v0.2		v0.5
General: User can create new states	1	General: Allows redefining header and colon . . . . . 1
v0.3		Numbers in tasks and references 1
General: List of Tasks . . . . .	1	

# Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.

<b>Symbols</b>	<code>\ifblank</code> 77, 118, 135,	<code>\thetaskcounter</code> ... 131
<code>\@currentlabel</code> .... 125	139, 143, 147,	<code>\thetasks</code> ..... 129
<code>\@currentlabelname</code> . 128	151, 155, 159, 163	<code>\tr@astate@other</code> .. 28
<code>\ifundefined</code> ... 14, 90	<code>\ifcsname</code> ..... 25, 45	<code>\tr@currentboxcolor</code>
<code>\showtasknumbersfalse</code>	<code>\ifx</code> ..... 94	<u>74</u> , 79, 82, 132, 133
..... 1		<code>\tr@defaulttitlelabel</code>
<code>\@showtasknumberstrue</code>	<b>L</b>	63, 65, 94, 102, 104
..... 3	<code>\LaTeX</code> ..... 125	<code>\tr@field</code> ..... <u>93</u> ,
		137, 141, 145,
	<b>N</b>	149, 153, 157, 161
<code>\_</code> ..... 102, 104	<code>\newif</code> ..... 1	<code>\tr@title</code> ..... 93, 131
<b>A</b>	<code>\newline</code> 137, 141, 145,	<code>\tr@usesfieldname</code> .
<code>\addcontentsline</code> .. 129	149, 153, 157, 161	... <u>40</u> , 96, 102,
<b>C</b>	<code>\newlistof</code> ..... 91, 92	104, 110, 119,
<code>\color@</code> ..... 14	<code>\numberline</code> ..... 129	137, 141, 145,
<code>\csname</code> .. 20, 26, 41, 46	<b>P</b>	149, 153, 157, 161
<b>D</b>	<code>\par</code> ..... 117	<code>\tr@usestatecolor</code> .
<code>\DeclareOption</code> ..... 3	<code>\printtaskreportversion</code>	.... 24, 74, 79, 82
<code>\detokenize</code> .... 20,	..... <u>12</u>	<code>\tr@version</code> ..... 11
25, 26, 41, 45, 46	<code>\ProcessOptions</code> .... 5	<code>\traddfieldname</code> ...
<b>E</b>	<code>\protect</code> ..... 129	. <u>40</u> , 52–58, 61, 71
<code>\else</code> 27, 47, 97, 100, 103	<code>\protected@edef</code> ... 128	<code>\traddstatecolor</code> ..
<code>\empty</code> ..... 94	<b>R</b>	..... <u>13</u> , 32–37
<code>\end</code> ..... 167	<code>\refstepcounter</code> ...	<code>\trlisttasks</code> . 86, 91, 92
<code>\endcsname</code> .... 20,	.... 124, 125, 127	<code>\trsetboxcolors</code> . <u>31</u> , 39
25, 26, 41, 45, 46	<code>\relax</code> ..... 5	<code>\trsetcolon</code> ..... <u>70</u>
<code>\ERROR</code> ..... 17	<code>\RequirePackage</code> ... 6–9	<code>\trsetfieldsnames</code> . <u>51</u>
<code>\expandafter</code> .....	<b>S</b>	<code>\trsettasklisttitle</code>
. 17, 20, 26, 41, 46	<code>\string</code> ..... 14	..... 85, 88
<b>F</b>	<b>T</b>	<code>\trsettitlelabel</code> .. <u>51</u>
<code>\fi</code> . 29, 49, 99, 105, 106	<code>\taskreportversion</code> .	<code>\trstatuscolor</code> . <u>75</u> , 130
<b>I</b>	..... <u>11</u> , 12	<code>\trtask</code> ..... <u>113</u>
<code>\if@showtasknumbers</code>	<code>\tcblower</code> ..... 164	<code>\trtask@temp</code> .. 119,
..... 1, 95, 101	<code>\textbf</code> .....	122, 128, 129, 131
	96, 98, 102, 104, 110	<b>U</b>
		<code>\usestatecolor</code> .... <u>13</u>

# List of Tasks

5.1	Make coffe . . . . .	4
5.2	Drink a coffe . . . . .	4
5.3	Roast the beef . . . . .	4
5.4	Drink another beer . . . . .	5
5.5	Drink a beer . . . . .	5
5.6	Task without a name . . . . .	5
5.7	Task without a name . . . . .	5
5.8	arg 2 . . . . .	5
6.1	Drink a coffe . . . . .	6
6.2	Drink a coffe . . . . .	7
6.3	Drink a coffe . . . . .	7
6.4	Drink a coffe . . . . .	7
6.5	Drink a coffe . . . . .	8
6.6	Drink a coffe . . . . .	8
6.7	Drink a coffe . . . . .	9
6.8	Drink a coffe . . . . .	9
6.9	Drink a coffe . . . . .	10
6.10	Drink a coffe . . . . .	11
11.1	Make a L <sup>A</sup> T <sub>E</sub> X Kanbam document . . . . .	16