The `isodate` package*

Harald Harders
h.harders@tu-bs.de

File Date 2010-01-03, Printed 2010-01-03

Abstract

This package provides commands to switch between different date formats (standard, ISO, numeric, LaTeX package). They are used by the `\today` command, by the `\printdate` and `\printdateTeX` commands that print any date, and by the `\daterange` command that prints a date range. At the moment, this package supports German (old and new orthography, Austrian), British, US, Australian as well as New Zealand English,[1] French, Italian, Danish, Swedish, and Norwegian.

The idea for this package was taken from the akletter class.

Contents

*This file has version v2.30 last revised 2010-01-03.

[1] In order to use Australian or New Zealand, you need a version of babel that supports the used language. It should be available, soon.

Acknowledgements

First of all I have to thank Axel Kielhorn who wrote the package `akletter` which inspired me to write `isodate`. The help of Heiko Oberdiek was necessary to handle characters in substrings which resulted in the package `substr`. David Sanderson found the bug which disabled `isodate` to work without `babel`. He also helped me to improve the documentation and sent me a link to the ISO 8601 norm [1]. Svend Tollak Munkejord has added the Norwegian language, Christian Schlauer has added Swedish, Philip Ratcliffe has added Italian.

Requirements

The package `isodate` needs the package `substr.sty` which can be obtained from `CTAN:macros/latex/contrib/substr/`.

1 Commands

1.1 Switching the date output format

`\isodate` `\numdate` `\shortdate` `\TeXdate` `\origdate` `\shortorigdate` `\Romandate` `\romandate` `\shortRomandate` `\shortromandate`

This package provides five commands to switch the output format of all commands that print dates (described later):

`\isodate`	date format described in ISO 8601 and DIN 5008 [1] (yyyy-mm-dd)
`\numdate`	numeric date format with four digits of the year
`\shortdate`	short numeric date format with two digits of the year
`\TeXdate`	date format used for version description of packages (yyyy/mm/dd)
`\origdate`	original LaTeX format
`\shortorigdate`	original LaTeX format with two instead of four digits of

	the year
`\Romandate`	As `\numdate` but with uppercase Roman numerals for the month
`\romandate`	As `\numdate` but with lowercase Roman numerals for the month
`\shortRomandate`	As `\shortdate` but with uppercase Roman numerals for the month
`\shortromandate`	As `\shortdate` but with lowercase Roman numerals for the month

These commands *do not* print any dates and they don't take an argument. They just switch the format for later usage of the date-printing commands `\today`, `\printdate`, `\printdateTeX`, and `\daterange`.

The numeric and short numeric as well as the Roman numbered formats change their behaviour depending on the current language:

German, nGerman	`dd.\,mm.~yyyy`	resp.	`dd.\,mm.\,yy`
US English	`mm/dd/yyyy`	resp.	`mm/dd/yy`
other languages	`dd/mm/yyyy`	resp.	`dd/mm/yy`

This package supports German (old and new rules, Austrian), US English, French, Danish, Italian, Swedish, and Norwegian. Switching the language by using `\selectlanguage` does *not* switch back to the original date format. So the current date format stays active when changing the language.

The change of the date format works locally. So it is possible to change it locally inside a group; e.g.,

```
\today, {\origdate\today}, \today
```

leads to '2010-01-03, 3rd January 2010, 2010-01-03'.

`\printyearoff` `\printyearon` By default, all formats print the day, month, and year. Sometimes, you may want to print the date without the year. This can be reached by using the command `\printyearoff`. You can switch back with `\printyearon` or by using `\printyearoff` inside a group (e.g., an environment). To switch globally, preceed the command by `\global`. An example:

```
\today, {\printyearoff\today}, \today
```

leads to '3rd January 2010, 3rd January, 3rd January 2010'.

`\printdayoff` `\printdayon` Likewise you can switch on or off printing the day using `\printdayon` and `\printdayoff`. Note that you still have to provide complete dates in the `\printdate` command, described in Section 1.3 below.

1.2 Printing today's date

`\today` As usual, the command `\today` prints the date of today. Its appearance is influenced by the current date format

1.3 Printing any date

`\printdate` The command `\printdate{#1}` prints any date in the current format. The argument may be a date in German, British English, or ISO format, e.g.,

```
\printdate{24.12.2000}
\printdate{24/12/2000}
\printdate{2000-12-24}
```

`\printdateTeX` The command `\printdateTeX{#1}` prints any date in the actual format. The argument must be in the LaTeX format `yyyy/mm/dd`, e.g.,

```
\printdateTeX{2000/12/24}
```

This command is useful for printing version information stored in a macro. For example the version of this package is stored in the macro `\filedate` ('2010/01/03'). To print it with the actual date format you can use the command `\printdateTeX{\filedate}` which leads to e.g., '2010-01-03' or '3rd January 2010'. Another possibility is to switch the input format to `tex` using `\dateinputformat`, described below.

1.4 Printing date ranges

`\daterange` The command `\daterange{#1}{#2}` prints a date range in the current format. The arguments may be a date in German, British English, or ISO format (see above). But there is a limitation: Both arguments must have the same input format.

Depending on the language and date format, this commands leaves out some of the data. The simplest way to understand it is to watch some examples:

```
{\isodate
\daterange{1999-05-03}{1999-05-31} ⟶ 1999-05-03 to 31
\daterange{1999-05-03}{1999-11-03} ⟶ 1999-05-03 to 11-03
\daterange{1999-05-03}{2000-04-07} ⟶ 1999-05-03 to 2000-04-07
}
{\origdate
\daterange{1999-05-03}{1999-05-31} ⟶ 3rd to 31st May 1999
\daterange{1999-05-03}{1999-11-03} ⟶ 0503 to 3rd November 1999
\daterange{1999-05-03}{2000-04-07} ⟶ 3rd May 1999 to 7th April 2000
}
```

1.5 Changing the ISO format

`\isodash` The ISO norm says that the date format is 'yyyy-mm-dd' or 'yyyymmdd' [1]. By default I use the hyphen '-' as separator. You can change this using the `\isodash`[2] command, e.g.,

[2]The name 'isodash' is a little bit confusing and was chosen due to my limited knowledge in English. It should be named 'isoseparator' or 'isosep'. But for compatiblity reasons I will not change it.

```
\printdate{24/12/2000},
\isodash{--}%
\printdate{24/12/2000},
\isodash{}%
\printdate{24/12/2000}
```

leads to '2000-12-24, 2000–12–24, 20001224'. Or for example

```
\isodash{$\cdot$}
\printdate{24/12/2000}
```

leads to '2000·12·24'.

1.6 Changing the original and short original format

\isospacebeforeday
\isospacebeforemonth
\isospacebeforeyear

By default, the original and short original format prints unbreakable spaces between the parts of the dates, e.g., '`19~May~2001`'. If you want to allow breakable spaces or other characters, you can redefine the spaces using `\isospacebeforeday`, `\isospacebeforemonth`, and `\isospacebeforeyear`:

```
\isospacebeforeyear{\ }
```

leads to '`19~May\ 2001`'. Notice that the space is written as `\␣` to ensure that the space is not getting lost under all circumstances.

As the names imply, the spaces before the specified part (day, month, or year) is changed. For most formats, only `\isospacebeforemonth` and `\isospacebeforeyear` are relevant, while for US English, `\isospacebeforeday` and `\isospacebeforeyear` are used.

This only effects the `orig` and `shortorig` formats.

1.7 Changing the short original format

\shortyearsign

The short original format normally prints the year with two digits, e.g., '19th May 01'. Some people want to add an additional sign before the year, e.g., '19th May '01'. This can be achieved by using the command `\shortyearsign`, e.g.,

```
\printdate{24/12/2000},
\shortyearsign{'}%
\printdate{24/12/2000}
```

leads to '24 december 00, 24 december '00' in English.

This only effects the `shortorig` format. The `short` numerical format stays unchanged.

1.8 Changing the German format

The spacings for the numerical formats in the German language (24. 12. 2000 resp. 24. 12. 00) were taken from the Duden [2] and are the defaults when using one of the German derivatives. Some people want to use different spacings. Thus from version 2.03 on it is possible to change them. You can change the spacing between the day and the month using the command `\daymonthsepgerman`. Using the command `\monthyearsepgerman` you can change the spacing between the month and the year for the long and the short format, e.g.,

`\daymonthsepgerman`
`\monthyearsepgerman`
`\monthyearsepnodaygerman`

```
\daymonthsepgerman{\quad}%
\monthyearsepgerman{\qquad}{\quad}%
{\numdate\printdate{24.12.2000}}, {\shortdate\printdate{24.12.2000}}
```

leads to '24. 12. 2000, 24. 12. 00'.

The default values are '`\,`' for the separator between day and month resp. '`\,`' between month and year in the short format and '`~`' in the long format.

Likewise, `\monthyearsepnodaygerman` is used for defining the spacing between the month and the year when printing the day is switched off (using `\printdayoff`). The arguments are the same as for `\monthyearsepgerman`. Default is no space for long and short format.

1.9 Changing the English format

By default, the English date format looks like '24th December 2000'. During the last years, a change has occured in many documents towards '24 December 2000'. This new format is called 'clean look'. Isodate's behaviour can be changed towards it using `\cleanlookdateon` and `\cleanlookdateoff`. This can also be done globally using the `cleanlook` package option.

`\cleanlookdateon`
`\cleanlookdateoff`

At the moment, the 'clean look' functionality only affects British English. If this trend also counts for different languages, please tell it me that I can add support for them.

1.10 User defined month formatting

Internally, the formats using Roman numerals for the month are just links to the `\numdate` and `\shortdate` formats with a changed format for printing the month. Therefore, the command `\Romandate` calls `\numdate` by following sequence:

```
\numdate[Roman]%
\isotwodigitdayfalse
```

This tells `\numdate` to format the month using the `\Roman` command and to typeset the day without a leading zero if it is less than ten.

You may do similar things, e.g.,

```
\numdate[Alph]
```

prints the months with the command `\Alph`, 'A', 'B', ... The day is printed with two digits since every call of `\numdate` or `\shortdate` calls `\isotwodigitdaytrue` which switches printing the day with two digits on. This does not make any sence but may serve as example. If you want to enable days with one digit, append `\isotwodigitdaytrue`:

```
\numdate[Alph]%
\isotwodigitdaytrue
```

You may declare any command that typesets a counter that is given as its mandatory argument (e.g., `\alph`, `\Alph`, `\arabic`, ...) in the optional argument of the `\numdate`, `\shortdate`, `\isodate`, and `\TeXdate` commands, without the leading backslash. You can, of course, define own commands that do it. For instance,

```
{\def\boldnum#1{\textbf{\twodigitarabic{#1}}}%
\numdate[boldnum]%
\printdate{24.3.2000}}
```

`\twodigitarabic` leas to '24/**03**/2000'. Here, the `\twodigitarabic` command has been used that prints a positive number with at least two digits.[3]

If you, for example want a numerical date format with the day and month printed with the 'natural' number of digits rather than with two digits, you may do it as follows:

```
{\numdate[arabic]\isotwodigitdayfalse
\printdate{1.2.2000}}
```

which leads to '1/2/2000'.

Using one of the other date formats reset the numerical format to its standard settings with arabic numerals (with two digits), e.g.,

```
{\numdate[Alph]\printdate{6.12.2000};
\isodate\printdate{6.12.2000};
\numdate\printdate{6.12.2000}}
```

leads to '6/L/2000; 2000-12-06; 06/12/2000'.

1.11 Switching the date input format

`\dateinputformat` As described above, the date can be given in different formats. For the German format `dd.mm.yyyy` and the ISO format `yyyy-mm-dd`, the input format is definite. But when using slashes to seperate the day, month, and year, different formats exist. British people use `dd/mm/yyyy`, American people use `mm/dd/yyyy`, while TEX uses `yyyy/mm/dd` which in fact is an ISO format with slashes instead of dashes.

By default, the British format is used. If the user wants to give the American or TEX format as argument of the `\printdate` or `\daterange` commands, the macro `\dateinputformat` can be used to change the behaviour.

[3] This command is also used for the numerical date formats.

This macro takes the name of the input format as single parameter, e.g., `\dateinputformat{american}`, for switching to American behaviour, e.i., `mm/dd/yyyy`. For example,

```
\numdate
\selectlanguage{UKenglish}%
\dateinputformat{american}%
\printdate{12/31/2004}
```

gives 31/12/2004. In this example, *input* format is American while the *output* format is English.

Valid arguments for the `\dateinputformat` command are `english`, `UKenglish`, `british`, `american`, `USenglish`, `tex`, `latex`, `TeX`, `LaTeX`. The meaning of most possibilities should be clear; `english` means British English.

Beware that the input format may only be changed for the date format using slashes. Thus, you don't have to and are not allowed to specify input formats other than these described above. For example, `\dateinputformat{german}` is not allowed (and not necessary).

2 Calling the package

The package is called using the `\usepackage` command:
`\usepackage[option]{isodate}`.

The possible package options can be seen in table 2.

Be aware that at least one language option must be set when calling isodate. The last language in the option list is the default language.

The package isodate works well together with `babel.sty`, `german.sty`, or `ngerman.sty`. It does not matter if `isodate` is loaded before or after the used language package.

It is also possible to use isodate without one of the language packages. Then it is not possible to switch between languages using the `\selectlanguage` command.[4] Then the default language is the last one in the option list. If an error occurs when using isodate without one of the packages `babel.sty`, `german.sty`, and `ngerman.sty` please run `tstlang.tex` through latex and send the file `tstlang.log` to the address `h.harders@tu-bs.de`.

If using isodate together with babel it can be useful to put the language options as global options into the optional parameters of the `\documentclass` command. Then automatically the available languages are the same for the text and the dates, and the default language is also the same. For example:

```
\documentclass[english,german]{article}
\usepackage{babel}
\usepackage[num]{isodate}
```

[4] Yes, there is a way to change the date language, but it is a little bit tricky:

```
\makeatletter
\def\iso@languagename{german}%
\dategerman%
\makeatother
```

Table 2: Package options

option	function
`iso`	start with ISO date format
`num`	start with numeric date format with 4 digits of the year
`short`	start with numeric date format with 2 digits of the year
`TeX`	start with LaTeX numeric date format (yyyy/mm/dd)
`orig`	start with normal LaTeX date format (default[a])
`shortorig`	start with short normal LaTeX date format (2 digits)
`Roman`	start with numeric date format (month in uppercase Roman numerals)
`roman`	start with numeric date format (month in lowercase Roman numerals)
`shortRoman`	start with short Roman format
`shortroman`	start with short roman format
`american`	support American English date format
`austrian`	support Austrian date format
`british`	support British English date format
`danish`	support Danish date format
`english`	support British English date format
`french`	support French date format
`german`	support German date format
`naustrian`	support new Austrian date format
`ngerman`	support new German date format
`italian`	support Italian date format
`norsk`	support Norwegian date format
`norwegian`	support Norwegian date format
`swedish`	support Swedish date format
`UKenglish`	support British English date format
`USenglish`	support American English date format
`inputenglish`	English date input format (default)
`inputbritish`	English date input format (default)
`inputUKenglish`	English date input format (default)
`inputamerican`	American date input format
`inputUSenglish`	American date input format
`inputtex`	TeX date input format
`inputTeX`	TeX date input format
`inputlatex`	TeX date input format
`inputLaTeX`	TeX date input format
`cleanlook`	use 'clean look' for English dates
`nocleanlook`	don't use 'clean look' for English dates (default)
`printdayon`	print complete date including the day (default)
`printdayoff`	omit the day in the date

[a]The original format is used as default in order to avoid a different document output by just including the package.

The input format options specify the input format that is used at the begin of the document. You don't have to define multiple options if you want to change the input format in the document using `\dateinputformat`. For example,

```
\documentclass[american,german,british]{article}
\usepackage{babel}
\usepackage[iso,inputamerican]{isodate}
\begin{document}
D \printdate{28.2.2000}\par
ISO \printdate{2000-2-28}\par
US \printdate{2/28/2000}\par
\dateinputformat{british}UK \printdate{28/2/2000}\par
\dateinputformat{tex}\TeX\ \printdate{2000/2/28}
\end{document}
```

works as expected.

Beware that only the mentioned input formats are defined. For example, `inputgerman` does not exist because it is not necessary.

3 Add new languages to the package

The easiest way to add new languages to the package is to copy one of the simple language files `danish.idf` or `french.idf` to the new language name, e.g., `plattdeutsch.idf`, and change it as necessary.

This new file can be used without changing `isodate.sty` if you use its name explicitly in the optional parameter of the `\usepackage` command. If you have added support for a new language please mail me.

A Licence

B Known errors

- The `\printdate` and `\printdateTeX` commands are not very good in checking the argument for correct syntax.
- The language definition files `french.idf` and `german.idf` are not yet commented.
- Isodate and draftcopy do not work together.
- Documentation of the code is partly poor.

C Planned features and changes

- Add other languages. Please help me with this topic. I don't know the date formats in other languages.
- Format short given years to four digits and calculate reasonable first and second digits.

References

[1] International Standard: ISO 8601. `http://www.iso.ch/markete/8601.pdf`, 1988-06-15.

[2] DUDEN Band 1. Die deutsche Rechtschreibung. 21. Auflage, Dudenverlag, Mannheim, Germany, 1996.

D The implementation

D.1 Package file isodate.sty

Heading of the files:

```
1 ⟨isodate⟩\NeedsTeXFormat{LaTeX2e}
2 ⟨isodate⟩\ProvidesPackage{isodate}
3 ⟨danish⟩\ProvidesFile{danish.idf}
4 ⟨english⟩\ProvidesFile{english.idf}
5 ⟨french⟩\ProvidesFile{french.idf}
6 ⟨german⟩\ProvidesFile{german.idf}
7 ⟨italian⟩\ProvidesFile{italian.idf}
8 ⟨norsk⟩\ProvidesFile{norsk.idf}
9 ⟨swedish⟩\ProvidesFile{swedish.idf}
10 ⟨isodate⟩  [2010/01/03  v2.30  Print dates with different formats (HH)]
11 ⟨language⟩  [2010/01/03  v2.30  Language definitions for isodate package (HH)]
```

The package:

```
12 ⟨*isodate⟩
13 \RequirePackage{ifthen}
14 \IfFileExists{substr.sty}{\RequirePackage{substr}%
15  }{\PackageError{isodate.sty}{Package file substr.sty not found}
16    {This version of isodate.sty needs the package substr.sty.^^J%
17      You can download it from
18      CTAN:/macros/latex/contrib/substr/^^J%
19      e.g., one CTAN node is ftp.dante.de.
20      Install substr.sty into your TeX tree.}}
```

Declare the options for the default date format.

```
21 \DeclareOption{iso}{\AtEndOfPackage{\isodate}}
22 \DeclareOption{num}{\AtEndOfPackage{\numdate}}
23 \DeclareOption{short}{\AtEndOfPackage{\shortdate}}
```

```
\DeclareOption{TeX}{\AtEndOfPackage{\TeXdate}}
\DeclareOption{orig}{\AtEndOfPackage{\origdate}}
\DeclareOption{shortorig}{\AtEndOfPackage{\shortorigdate}}
\DeclareOption{Roman}{\AtEndOfPackage{\Romandate}}
\DeclareOption{roman}{\AtEndOfPackage{\romandate}}
\DeclareOption{shortRoman}{\AtEndOfPackage{\shortRomandate}}
\DeclareOption{shortroman}{\AtEndOfPackage{\shortromandate}}
\DeclareOption{cleanlook}{\AtEndOfPackage{\cleanlookdateon}}
\DeclareOption{nocleanlook}{\AtEndOfPackage{\cleanlookdateoff}}
```

Declare the options which decide wheather day is printed.

```
\DeclareOption{printdayoff}{\AtEndOfPackage{\printdayoff}}
\DeclareOption{printdayon}{\AtEndOfPackage{\printdayon}}
```

Declare the options for the default date input format.

```
\DeclareOption{inputenglish}{\AtEndOfPackage{\dateinputformat{english}}}
\DeclareOption{inputbritish}{\AtEndOfPackage{\dateinputformat{english}}}
\DeclareOption{inputUKenglish}{\AtEndOfPackage{\dateinputformat{english}}}
\DeclareOption{inputamerican}{\AtEndOfPackage{\dateinputformat{american}}}
\DeclareOption{inputUSenglish}{\AtEndOfPackage{\dateinputformat{american}}}
\DeclareOption{inputtex}{\AtEndOfPackage{\dateinputformat{tex}}}
\DeclareOption{inputTeX}{\AtEndOfPackage{\dateinputformat{tex}}}
\DeclareOption{inputlatex}{\AtEndOfPackage{\dateinputformat{tex}}}
\DeclareOption{inputLaTeX}{\AtEndOfPackage{\dateinputformat{tex}}}
```

Declare the options for language support.

```
\DeclareOption{american}{\input{english.idf}}
\DeclareOption{australian}{\input{english.idf}}
\DeclareOption{austrian}{\input{german.idf}}
\DeclareOption{danish}{\input{danish.idf}}
\DeclareOption{english}{\input{english.idf}}
\DeclareOption{british}{\input{english.idf}}
\DeclareOption{french}{\input{french.idf}}
\DeclareOption{frenchb}{\input{french.idf}}
\DeclareOption{german}{\input{german.idf}}
\DeclareOption{italian}{\input{italian.idf}}
\DeclareOption{naustrian}{\input{german.idf}}
\DeclareOption{newzealand}{\input{english.idf}}
\DeclareOption{ngerman}{\input{german.idf}}
\DeclareOption{norsk}{\input{norsk.idf}}
\DeclareOption{norwegian}{\input{norsk.idf}}
\DeclareOption{swedish}{\input{swedish.idf}}
\DeclareOption{UKenglish}{\input{english.idf}}
\DeclareOption{USenglish}{\input{english.idf}}
```

Make it possible to load language definition files that are not known by this package.

```
\DeclareOption*{%
  \InputIfFileExists{\CurrentOption.idf}{}{%
    \PackageError{isodate}{%
      Isodate definition file \CurrentOption.idf not found}{%
```

```
      Maybe you misspelled the language option?}}%
  }
```

Set default option to `orig`.

```
\ExecuteOptions{orig,nocleanlook,printdayon}
```

Process the options.

```
\ProcessOptions*
```

Handle the case that no language was given. Throw an error message. Each language definition file `*.idf` must contain a line

```
\let\iso@languageloaded\active
```

that defines the command `\iso@languageloaded`.

```
\ifx\iso@languageloaded\@undefined
  \PackageError{isodate}{%
    You haven't specified a language option}{%
    You need to specify a language, either as a global
    option\MessageBreak
    or as an optional argument to the \string\usepackage\space
    command.\MessageBreak
    If you have used the old isodate package (version <=1.06) you can
    change the\MessageBreak
    usepackage command to \protect\usepackage{isodate}.\MessageBreak
    You shouldn't try to proceed from here, type x to quit.}
\fi
```

`\iso@printday` Prints a day.

```
\newcommand*\iso@printday[1]{%
  \ifisotwodigitday
    \ifthenelse{\number#1<10}{0}{}%
  \fi
  \number#1%
}%
```

`\twodigitarabic` Typesets the given counter with at least two digits. This command is very simple and does only work for positive numbers below 100.

```
\newcommand*\twodigitarabic[1]{%
  \ifthenelse{\number\arabic{#1}<10}{0}{}%
  \arabic{#1}%
}
```

`\iso@printmonth` Prints a month using `\theiso@tmpmonth` as output fourmat.

```
\newcommand*\iso@printmonth[1]{%
  \setcounter{iso@tmpmonth}{#1}%
  \theiso@tmpmonth%
}
```

Define the help counter that prints the month and initialize it to print arabic numbers.

```
\newcounter{iso@tmpmonth}
%\def\theiso@tmpmonth{\arabic{iso@tmpmonth}}
```

`\iso@yeartwo` Prints the argument of the command with two digits.
Example: `\iso@yeartwo{1873}` ⟶ 73.

```
\newcounter{iso@yeartwo}%
\newcommand*\iso@yeartwo[1]{%
  \setcounter{iso@yeartwo}{\number#1}%
  \whiledo{\theiso@yeartwo>99}{%
    \addtocounter{iso@yeartwo}{-100}}{}%
  \ifthenelse{\number\theiso@yeartwo<10}{0}{}\theiso@yeartwo
}
```

`\iso@yearfour` Prints the argument of the command with four digits.

```
\newcommand*\iso@yearfour[1]{%
  \ifthenelse{\number#1<1000}{0}{}%
  \ifthenelse{\number#1<100}{0}{}%
  \ifthenelse{\number#1<10}{0}{}%
  \number#1%
}%
```

`\ifisotwodigitday` Print day with two digits or natural number of digits?

```
\newif\ifisotwodigitday
```

`\iso@dateformat` In this command, the current active date format ist stored. Possible values are: `numeric`, `short`, `iso`, `orig`, `shortorig`, `TeX`.

```
\def\iso@dateformat{numeric}
```

`\iso@inputformat` This macro stores which input format is used for dates given with slashes. Valid formats are `english` (dd/mm/yyyy), `american` (mm/dd/yyyy), and `tex` (yyyy/mm/dd). By default, English is used.

```
\DeclareRobustCommand*\dateinputformat[1]{%
  \ifthenelse{%
    \equal{#1}{english}\OR
    \equal{#1}{british}\OR
    \equal{#1}{UKenglish}}{%
    \def\iso@inputformat{english}%
  }{%
    \ifthenelse{%
      \equal{#1}{american}\OR
      \equal{#1}{USenglish}}{%
      \def\iso@inputformat{american}%
    }{%
      \ifthenelse{%
        \equal{#1}{tex}\OR
        \equal{#1}{TeX}\OR
```

```
        \equal{#1}{latex}\OR
        \equal{#1}{LaTeX}}{%
        \def\iso@inputformat{tex}%
      }{%
        \PackageError{isodate}{Invalid date input format}{%
          Maybe you misspelled the language option (english, american,
          tex)?}%
      }%
    }%
  }%
}
```

`\iso@inputformat` This macro stores which input format is used for dates given with slashes. Valid formats are `english` (dd/mm/yyyy), `american` (mm/dd/yyyy), and `tex` (yyyy/mm/dd). By default, English is used.

```
\dateinputformat{english}
```

`\numdate` Switches to long numerical date format.

```
\DeclareRobustCommand*\numdate[1][twodigitarabic]{%
  \def\iso@dateformat{numeric}%
  \isotwodigitdaytrue
  \def\theiso@tmpmonth{\csname #1\endcsname{iso@tmpmonth}}%
}
```

`\shortdate` Switches to short numerical date format.

```
\DeclareRobustCommand*\shortdate[1][twodigitarabic]{%
  \def\iso@dateformat{short}%
  \isotwodigitdaytrue
  \def\theiso@tmpmonth{\csname #1\endcsname{iso@tmpmonth}}%
}
```

`\isodate` Switches to ISO date format.

```
\DeclareRobustCommand*\isodate[1][twodigitarabic]{%
  \def\iso@dateformat{iso}%
  \isotwodigitdaytrue
  \def\theiso@tmpmonth{\csname #1\endcsname{iso@tmpmonth}}%
}
```

`\origdate` Switches to the original date format.

```
\DeclareRobustCommand*\origdate{%
  \def\iso@dateformat{orig}%
  \isotwodigitdayfalse
  \def\theiso@tmpmonth{\twodigitarabic{iso@tmpmonth}}%
}
```

`\shortorigdate` Switches to the short original date format.

```
\DeclareRobustCommand*\shortorigdate{%
  \def\iso@dateformat{shortorig}%
  \isotwodigitdayfalse
  \def\theiso@tmpmonth{\twodigitarabic{iso@tmpmonth}}%
}
```

q

`\TeXdate` Switches to LaTeX date format.

```
\DeclareRobustCommand*\TeXdate[1][twodigitarabic]{%
  \def\iso@dateformat{TeX}%
  \isotwodigitdaytrue
  \def\theiso@tmpmonth{\csname #1\endcsname{iso@tmpmonth}}%
}
```

`\Romandate` Switches to long numerical date format with month printed in uppercase Roman numerals.

```
\DeclareRobustCommand*\Romandate{%
  \numdate[Roman]%
  \isotwodigitdayfalse
}
```

`\romandate` Switches to long numerical date format with month printed in lowercase Roman numerals.

```
\DeclareRobustCommand*\romandate{%
  \numdate[roman]%
  \isotwodigitdayfalse
}
```

`\shortRomandate` Switches to short numerical date format with month printed in uppercase Roman numerals.

```
\DeclareRobustCommand*\shortRomandate{%
  \shortdate[Roman]%
  \isotwodigitdayfalse
}
```

`\shortromandate` Switches to short numerical date format with month printed in lowercase Roman numerals.

```
\DeclareRobustCommand*\shortromandate{%
  \shortdate[roman]%
  \isotwodigitdayfalse
}
```

`\isodash` Changes the dash in the ISO date format. The default is '-'.

```
\def\iso@isodash{-}%
\DeclareRobustCommand*\isodash[1]{\def\iso@isodash{#1}}%
```

Define the sign that is printed before a two digit year in the short original format. Default is nothing.

\shortyearsign

```
\def\iso@twodigitsign{}
\DeclareRobustCommand*\shortyearsign[1]{\def\iso@twodigitsign{#1}}%
```

\isorangesign Defines the sign or word that is printed between the two dates in a date range. e.g., in English the default is ‘ to ’.

```
\def\iso@rangesign{\csname iso@rangesign@\iso@languagename\endcsname}%
\DeclareRobustCommand*\isorangesign[1]{\def\iso@rangesign{#1}}%
```

\printyearoff \printyearon Switches printing of the year on or off. Default is to print the year.

```
\newif\ifiso@printyear
\DeclareRobustCommand*\printyearon{\iso@printyeartrue}
\DeclareRobustCommand*\printyearoff{\iso@printyearfalse}
\printyearon
```

\printdayoff \printdayon Switch on or off suppressing the day in date output. Default is not print the day.

```
\newif\ifiso@doprintday
\DeclareRobustCommand*\printdayon{\iso@doprintdaytrue}
\DeclareRobustCommand*\printdayoff{\iso@doprintdayfalse}
\printdayon
```

\cleanlookdateoff \cleanlookdateon Switch on or off ‘clean look’ for English dates. Default is not to use ‘clean look’.

```
\newif\ifiso@cleanlook
\DeclareRobustCommand*\cleanlookdateon{\iso@cleanlooktrue}
\DeclareRobustCommand*\cleanlookdateoff{\iso@cleanlookfalse}
\cleanlookdateoff
```

\isospacebeforeday \isospacebeforemonth \isospacebeforeyear Change the spaces in the orig and short orig format. Default is ~ for all of them.

```
\newcommand*\iso@daysep{~}
\newcommand*\iso@monthsep{~}
\newcommand*\iso@yearsep{~}
\DeclareRobustCommand*\isospacebeforeday[1]{\def\iso@daysep{#1}}
\DeclareRobustCommand*\isospacebeforemonth[1]{\def\iso@monthsep{#1}}
\DeclareRobustCommand*\isospacebeforeyear[1]{\def\iso@yearsep{#1}}
```

\iso@printdate Defines the command `iso@printdate` which takes three arguments (year, month, day) and prints the date by using the `\today` command.

```
\newcommand*\iso@printdate[3]{%
  \begingroup%
```

Generate a warning if the active language is not known by `isodate`.

```
    \@ifundefined{iso@printdate@\iso@languagename}{%
      \PackageWarning{isodate}{Language \iso@languagename\space unknown
        to isodate.\MessageBreak
        Using default format}%
    }{}%
```

The counters `\year`, `\month`, and `\day` are preserved as counters instead of changed to macros (as it has been done until version 2.25) to avoid problems with languages that are not defined in `isodate.sty`.

```
    \year=#1 %
    \month=#2 %
    \day=#3 %
    \today%
  \endgroup%
}
```

`\printdate` Prints a date that is given as one argument in one of these formats: `yyyy-mm-dd`, `dd/mm/yyyy`, `dd.mm.yyyy`.

```
\DeclareRobustCommand*\printdate[1]{%
```

Define `\iso@date` command to expand the argument `#1`.

```
  \edef\iso@date{#1}%
```

Count appearances of '/', '-', and '.' in the argument.

```
  \SubStringsToCounter{iso@slash}{/}{\iso@date}%
  \SubStringsToCounter{iso@minus}{-}{\iso@date}%
  \SubStringsToCounter{iso@dot}{.}{\iso@date}%
```

If number of '.' in the argument is equal to 2 then the German format `dd.mm.yyyy` is used.

```
  \ifthenelse{\equal{\theiso@dot}{2}}{%
    \expandafter\iso@input@german\iso@date\@empty}{%
```

If number of '-' in the argument is equal to 2 then the ISO format `yyyy-mm-dd` is used.

```
    \ifthenelse{\equal{\theiso@minus}{2}}{%
      \expandafter\iso@input@iso\iso@date\@empty}{%
```

If number of '/' in the argument is equal to 2 then the British English format `dd/mm/yyyy` is used.

```
      \ifthenelse{\equal{\theiso@slash}{2}}{%
        \expandafter\iso@input@english\iso@date\@empty}{%
```

Else no of the formats above is used an thus an error message is thrown.

```
        ????\iso@isodash ??\iso@isodash ??%
        \PackageError{isodate}{unrecognized date format}{Use one of
          the following formats as macro argument:^^J%
          \space\space dd.mm.yyyy^^J%
          \space\space dd/mm/yyyy^^J%
          \space\space yyyy-mm-dd^^J%
          Don't use any spaces or commands like \protect\, or
          \protect~ inside the argument.}%
        }}}%
}
```

`\iso@input@iso` Converts a string with the format `yyyy-mm-dd` to three arguments `{#1}{#2}{#3}` and calls `\iso@printdate`.

```
\def\iso@input@iso#1-#2-#3\@empty{\iso@printdate{#1}{#2}{#3}}
```

`\iso@input@german` Converts a string with the format `dd.mm.yyyy` to three arguments `{#3}{#2}{#1}` and calls `\iso@printdate`.

```
\def\iso@input@german#1.#2.#3\@empty{\iso@printdate{#3}{#2}{#1}}
```

`\iso@input@english` Converts a string with the format `dd/mm/yyyy` to three arguments `{#3}{#2}{#1}` and calls `\iso@printdate`.

```
\def\iso@input@english#1/#2/#3\@empty{%
  \ifthenelse{\equal{\iso@inputformat}{tex}}{%
    \iso@printdate{#1}{#2}{#3}%
  }{%
    \ifthenelse{\equal{\iso@inputformat}{american}}{%
      \iso@printdate{#3}{#1}{#2}%
    }{%
      \iso@printdate{#3}{#2}{#1}%
    }%
  }%
}
```

`\printdateTeX` Prints a date that is given as one argument in the format `yyyy/mm/dd`.

```
\DeclareRobustCommand*\printdateTeX[1]{%
```

Define `\iso@date` command to expand the argument `#1`.

```
  \edef\iso@date{#1}%
```

Count appearances of '/' in the argument.

```
  \SubStringsToCounter{iso@slash}{/}{\iso@date}%
```

If number of '/' in the argument is equal to 2 then the LaTeX format `yyyy/mm/dd` is used.

```
  \ifthenelse{\equal{\theiso@slash}{2}}{%
    \expandafter\iso@input@TeX\iso@date\@empty}{%
```

Else no of the formats above is used an thus an error message is thrown.

```
    ????\iso@isodash ??\iso@isodash ??%
    \PackageError{isodate}{unrecognized date format}{Use one of
      the following formats as macro argument:^^J%
      \space\space dd.mm.yyyy^^J%
      \space\space dd/mm/yyyy^^J%
      \space\space yyyy-mm-dd^^J%
      Don't use any spaces or commands like \protect\, or
      \protect~ inside the argument.}%
    }}
```

`\iso@input@TeX` Converts a string with the format `yyyy/mm/dd` to three arguments `{#1}{#2}{#3}` and calls `\iso@printdate`.

```
\def\iso@input@TeX#1/#2/#3\@empty{\iso@printdate{#1}{#2}{#3}}
```

`\iso@printmonthday@int` ??????

```
\def\iso@printmonthday@int#1#2{%
  \ifthenelse{\equal{\iso@dateformat}{iso}}{%
```

```
    \iso@printmonth{#1}%
    \ifiso@doprintday
      \iso@isodash\iso@printday{#2}%
    \fi
  }{%
    \ifthenelse{\equal{\iso@dateformat}{TeX}}{%
      \iso@printmonth{#1}%
      \ifiso@doprintday
        /\iso@printday{#2}%
      \fi
    }{%
      \PackageError{isodate.sty}{\csname iso@printmonthday\endcsname:
        Invalid date format `\iso@dateformat'}{Internal error. Please
        report to the package author.}
    }%
  }%
}
```

`\iso@printdate@int` ??????

```
\def\iso@printdate@int#1#2#3{%
  \ifiso@printyear
    \ifthenelse{\equal{\iso@dateformat}{iso}}{%
      \iso@yearfour{\number#1}\iso@isodash%
    }{%
      \ifthenelse{\equal{\iso@dateformat}{TeX}}{%
        \iso@yearfour{\number#1}/%
      }{%
        \PackageError{isodate.sty}{\csname iso@printmonthday\endcsname:
          Invalid date format `\iso@dateformat'}{Internal error. Please
          report to the package author.}
      }%
    }%
  \fi
  \csname iso@printmonthday@int\endcsname{\number#2}{\number#3}%
}
```

`\iso@daterange@int` ??????

```
\def\iso@daterange@int#1#2#3#4#5#6{%
  \ifthenelse{\equal{\iso@dateformat}{iso}\OR
              \equal{\iso@dateformat}{TeX}}{%
    \csname iso@printdate@\iso@languagename\endcsname{#1}{#2}{#3}%
    \iso@rangesign%
    \ifthenelse{\equal{\number#1}{\number#4}}{%
      \ifthenelse{\equal{\number#2}{\number#5}}{%
        \ifiso@doprintday
          \iso@printday{#6}%
        \else
          \iso@printmonthday@int{#5}{#6}%
        \fi
      }{%
```

```
        \iso@printmonthday@int{#5}{#6}%
      }%
    }{%
      \iso@printdate@int{#4}{#5}{#6}%
    }%
  }{%
    \PackageError{isodate.sty}{\csname iso@printmonthday\endcsname:
      Invalid date format `\iso@dateformat'}{Internal error. Please
      report to the package author.}
  }%
}
```

`\daterange` Prints a date range.

```
\DeclareRobustCommand*\daterange[2]{%
```

Define `\iso@date` and `\iso@@date` commands to expand the argument `#1` and `#2`. Define `\iso@@@date` which contains both arguments devided by a komma.

```
  \edef\iso@date{#1}%
  \edef\iso@@date{#2}%
  \edef\iso@@@date{\iso@date,\iso@@date}%
```

Count appearances of '/', '-', and '.' in the arguments.

```
  \SubStringsToCounter{iso@slash}{/}{\iso@date}%
  \SubStringsToCounter{iso@minus}{-}{\iso@date}%
  \SubStringsToCounter{iso@dot}{.}{\iso@date}%
  \SubStringsToCounter{iso@@slash}{/}{\iso@@date}%
  \SubStringsToCounter{iso@@minus}{-}{\iso@@date}%
  \SubStringsToCounter{iso@@dot}{.}{\iso@@date}%
```

If number of '.' in both arguments is equal to 2 then the German format `dd.mm.yyyy` is used.

```
  \ifthenelse{\equal{\theiso@dot}{2}\AND\equal{\theiso@@dot}{2}}{%
    \expandafter\iso@range@input@german\iso@@@date\@empty}{%
```

If number of '-' in both arguments is equal to 2 then the ISO format `yyyy-mm-dd` is used.

```
    \ifthenelse{\equal{\theiso@minus}{2}\AND\equal{\theiso@@minus}{2}}{%
      \expandafter\iso@range@input@iso\iso@@@date\@empty}{%
```

If number of '/' in both arguments is equal to 2 then the British English format `dd/mm/yyyy` is used.

```
      \ifthenelse{\equal{\theiso@slash}{2}\AND
                  \equal{\theiso@@slash}{2}}{%
        \expandafter\iso@range@input@english\iso@@@date\@empty}{%
```

Else no of the formats above is used an thus an error message is thrown.

```
        ????\iso@isodash ??\iso@isodash ??%
        \PackageError{isodate}{unrecognized date format}{Use one of
          the following formats as macro argument:^^J%
          \space\space dd.mm.yyyy^^J%
          \space\space dd/mm/yyyy^^J%
          \space\space yyyy-mm-dd^^J%
```

```
354           Don't use any spaces or commands like \protect\, or
355           \protect~ inside the argument.^^J
356           Use the same format for both arguments.}%
357         }}}%
358 }
```

`\iso@range@input@iso` Converts a string with the format `yyyy-mm-dd,yyyy-mm-dd` to six arguments `{#1}{#2}{#3}{#4}{#5}{#6}` and calls `\iso@daterange@`language.

```
359 \def\iso@range@input@iso#1-#2-#3,#4-#5-#6\@empty{%
360   \begingroup
```

Generate a warning if the active language is not known by `isodate`.

```
361     \@ifundefined{iso@daterange@\iso@languagename}{%
362       \PackageWarning{isodate}{Language \iso@languagename\space unknown
363         to isodate.\MessageBreak
364         Using default date range\MessageBreak
365         with range sign --}%
366         \expandafter\def\csname iso@printdate@\iso@languagename\endcsname{}%
```

Print date range in fall-back format.

```
367       \iso@printdate{#1}{#2}{#3}--\iso@printdate{#4}{#5}{#6}%
368     }{%
```

Print date range in the chosen `isodate` format.

```
369       \ifthenelse{\equal{\number#1}{\number#4}}{}{\printyearon}%
370       \csname iso@daterange@\iso@languagename\endcsname{%
371         #1}{#2}{#3}{#4}{#5}{#6}%
372     }%
373   \endgroup
374 }
```

`\iso@range@input@german` Converts a string with the format `dd.mm.yyyy,dd.mm.yyyy` to six arguments `{#3}{#2}{#1}{#6}{#5}{#4}` and calls `\iso@daterange@`language.

```
375 \def\iso@range@input@german#1.#2.#3,#4.#5.#6\@empty{%
376   \begingroup
```

Generate a warning if the active language is not known by `isodate`.

```
377     \@ifundefined{iso@daterange@\iso@languagename}{%
378       \PackageWarning{isodate}{Language \iso@languagename\space unknown
379         to isodate.\MessageBreak
380         Using default date range\MessageBreak
381         with range sign --}%
382         \expandafter\def\csname iso@printdate@\iso@languagename\endcsname{}%
```

Print date range in fall-back format.

```
383       \iso@printdate{#3}{#2}{#1}--\iso@printdate{#6}{#5}{#4}%
384     }{%
```

Print date range in the chosen `isodate` format.

```
385       \ifthenelse{\equal{\number#3}{\number#6}}{}{\printyearon}%
386       \csname iso@daterange@\iso@languagename\endcsname{%
387         #3}{#2}{#1}{#6}{#5}{#4}%
```

```
    }%
  \endgroup
}
```

`\iso@range@input@english` Converts a string with the format `dd/mm/yyyy,dd/mm/yyyy` to six arguments `{#3}{#2}{#1}{#6}{#5}{#4}` and calls `\iso@daterange@`language.

```
\def\iso@range@input@english#1/#2/#3,#4/#5/#6\@empty{%
  \begingroup
```

Generate a warning if the active language is not known by `isodate`.

```
    \@ifundefined{iso@daterange@\iso@languagename}{%
      \PackageWarning{isodate}{Language \iso@languagename\space unknown
        to isodate.\MessageBreak
        Using default date range\MessageBreak
        with range sign --}%
        \expandafter\def\csname iso@printdate@\iso@languagename\endcsname{}%
```

Print date range in fall-back format.

```
        \ifthenelse{\equal{\iso@inputformat}{tex}}{%
          \iso@printdate{#1}{#2}{#3}--\iso@printdate{#4}{#5}{#6}%
        }{%
          \ifthenelse{\equal{\iso@inputformat}{american}}{%
            \iso@printdate{#3}{#1}{#2}--\iso@printdate{#6}{#4}{#5}%
          }{%
            \iso@printdate{#3}{#2}{#1}--\iso@printdate{#6}{#5}{#4}%
          }%
        }%
    }{%
```

Print date range in the chosen `isodate` format.

```
      \ifthenelse{\equal{\number#3}{\number#6}}{}{\printyearon}%
      \ifthenelse{\equal{\iso@inputformat}{tex}}{%
        \csname iso@daterange@\iso@languagename\endcsname{%
          #1}{#2}{#3}{#4}{#5}{#6}%
      }{%
        \ifthenelse{\equal{\iso@inputformat}{american}}{%
          \csname iso@daterange@\iso@languagename\endcsname{%
            #3}{#1}{#2}{#6}{#4}{#5}%
        }{%
          \csname iso@daterange@\iso@languagename\endcsname{%
            #3}{#2}{#1}{#6}{#5}{#4}%
        }%
      }%
    }%
  \endgroup
}
```

Define the counters for conting the appearances of '.', '-', and '/' in the arguments.

```
\newcounter{iso@slash}
\newcounter{iso@minus}
\newcounter{iso@dot}
```

```
\newcounter{iso@@slash}
\newcounter{iso@@minus}
\newcounter{iso@@dot}
```

The command `\iso@languagename` is defined to be able to use this package without loading one of the language packages babel.sty, german.sty, or ngerman.sty.

If neither babel.sty nor german.sty nor ngerman.sty is loaded my computer returns 'nohyphenation' when using `\languagename`. So this is the indication that none of the above packages is loaded.

```
\AtBeginDocument{%
  \@tempswafalse
  \@ifpackageloaded{babel}{%
    \@tempswatrue
    \typeout{isodate: babel.sty has been loaded}%
  }{}%
  \@ifpackageloaded{german}{%
    \@tempswatrue
    \typeout{isodate: german.sty has been loaded}%
  }{}%
  \@ifpackageloaded{ngerman}{%
    \@tempswatrue
    \typeout{isodate: ngerman.sty has been loaded}%
  }{}%
```

The language is not equal 'nohyphenation'. So one of the language packages is loaded. Replace the internal language name `\iso@languagename` by the global language name `\languagename`.

```
  \if@tempswa
    \gdef\iso@languagename{\languagename}%
```

Reload language to surely switch to new date format. The languagename gets first expanded because of errors that would occur otherwise.

```
    \edef\iso@tmplang{\languagename}%
    \expandafter\selectlanguage\expandafter{\iso@tmplang}%
  \else
```

At the end of the preamble still none of the language packages are loaded. So no language switching is possible. Set the date language manually to the last language that was loaded for isodate.

```
    \typeout{isodate: babel.sty, (n)german.sty have not been loaded}%
    \csname date\iso@languagename\endcsname%
  \fi
}
```

⟨/isodate⟩

D.2 Language definition file danish.idf

`\iso@languageloaded` Define the command `\iso@languageloaded` in order to enable `isodate.sty` to determine if at least one language is loaded.

⟨∗danish⟩

```
\let\iso@languageloaded\active
\typeout{Define commands for Danish date format}
```

`\month@danish` Prints the name of today's month in the long form for the original date format.

```
\def\month@danish{\ifcase\month\or
    januar\or februar\or marts\or april\or maj\or juni\or
    juli\or august\or september\or oktober\or november\or december\fi}
```

`\iso@printmonthday@danish` Prints the month and the day given as two arguments (`{mm}{dd}`) in the current date format.

```
\def\iso@printmonthday@danish#1#2{%
  \ifthenelse{\equal{\iso@dateformat}{iso}\OR
              \equal{\iso@dateformat}{TeX}}{%
    \iso@printmonthday@int{#1}{#2}%
  }{%
```

Numeric and short date format: `dd/mm/`

```
    \ifthenelse{\equal{\iso@dateformat}{numeric}\OR
                \equal{\iso@dateformat}{short}}{%
      \ifiso@doprintday
        \iso@printday{#2}/%
      \fi
      \iso@printmonth{#1}%
    }{%
```

Original date format: `d. mmm`

```
      \ifthenelse{\equal{\iso@dateformat}{orig}\OR
                  \equal{\iso@dateformat}{shortorig}}{%
        \ifiso@doprintday
          \iso@printday{#2}.\iso@monthsep
        \fi
        \begingroup
        \edef\lmonth{#1}\def\month{\lmonth}%
        \month@danish%
        \endgroup
      }{}%
    }%
  }%
}
```

`\iso@printdate@danish` Prints the date given as three arguments (`{yyyy}{mm}{dd}`) in the actual date format

```
\def\iso@printdate@danish#1#2#3{%
```

ISO or LaTeXdate format: `yyyy\iso@printmonthday@danish`

```
  \ifthenelse{\equal{\iso@dateformat}{iso}\OR
              \equal{\iso@dateformat}{TeX}}{%
    \iso@printdate@int{#1}{#2}{#3}%
  }{%
    \iso@printmonthday@danish{\number#2}{\number#3}%
    \ifiso@printyear
```

?????

```
      \ifthenelse{\equal{\iso@dateformat}{orig}\OR
                  \equal{\iso@dateformat}{shortorig}}{%
      }{%
        /%
      }%
```

numeric date format: `\iso@printmonthday@danish yyyy`

```
      \ifthenelse{\equal{\iso@dateformat}{numeric}}{%
        \iso@yearfour{\number#1}%
      }{%
```

original date format: `\iso@printmonthday@danish~yyyy`

```
        \ifthenelse{\equal{\iso@dateformat}{orig}}{%
          \iso@yearsep\iso@yearfour{\number#1}%
        }{%
```

short original date format: `\iso@printmonthday@danish~yy`

```
          \ifthenelse{\equal{\iso@dateformat}{shortorig}}{%
            \iso@yearsep\iso@twodigitsign\iso@yeartwo{\number#1}%
          }{%
```

short date format: `\iso@printmonthday@danish yy`

```
            \ifthenelse{\equal{\iso@dateformat}{short}}{%
              \iso@yeartwo{\number#1}%
            }{}%
          }%
        }%
      }%
    \fi
  }%
}
```

`\iso@datedanish` This command redefines the `\today` command to print in the actual date format.

```
\def\iso@datedanish{%
  \def\today{\iso@printdate@danish{\year}{\month}{\day}}}%
```

`\iso@daterange@...` Define date-range commands for dialects.

```
\expandafter\def\csname iso@daterange@\CurrentOption\endcsname{%
  \iso@daterange@danish}%
```

`\iso@daterange@danish` This command takes six arguments (`{yyyy1}{mm1}{dd1}{yyyy2}{mm2}{dd2}`) and prints the corrosponding date range in the actual date format.

```
\def\iso@daterange@danish#1#2#3#4#5#6{%
```

ISO or LaTeX date format.

```
  \ifthenelse{\equal{\iso@dateformat}{iso}\OR
              \equal{\iso@dateformat}{TeX}}{%
```

Call the appropriate international routine.

```
    \iso@daterange@int{#1}{#2}{#3}{#4}{#5}{#6}%
  }{%
```

Numeric, short, or original date format.

If year and month are equal, only print the day of the start date. If only the year is equal, only print month and day of the start date. Otherwise print the whole start date.

```
    \ifthenelse{\equal{\number#1}{\number#4}}{%
      \ifthenelse{\equal{\number#2}{\number#5}}{%
        \ifiso@doprintday
          \ifthenelse{\equal{\iso@dateformat}{orig}\OR
                      \equal{\iso@dateformat}{shortorig}}{%
            \iso@printday{#3}.%
          }{%
            \iso@printday{#3}%
          }%
        \else
          \iso@printmonthday@danish{#2}{#3}%
        \fi
      }{%
        \iso@printmonthday@danish{#2}{#3}%
      }%
    }{%
      \csname iso@printdate@\iso@languagename\endcsname{#1}{#2}{#3}%
    }%
```

Print the end date.

```
    \iso@rangesign
    \csname iso@printdate@\iso@languagename\endcsname{#4}{#5}{#6}%
  }%
}
```

`\iso@rangesign@danish` Sets the word between start and end date in a date range to ‘ til ’.

```
\expandafter\def\csname iso@rangesign@\CurrentOption\endcsname{~til~}
```

Define the language name that will the active language for isodate if none of the packages babel.sty, german.sty, and ngerman.sty is loaded and if this is the last language that is used for isodate. If one of the above packages is used this definition will be overridden by the command `\languagename` that will always return the current used language.

```
\def\iso@languagename{danish}%
```

Redefine the command `\datedanish` that is used by babel to switch to the original Danish date format to enable the use of different date formats. This has to be done after the preamble in order to ensure to overwrite the babel command.

```
\AtBeginDocument{%
  \ifx\undefined\iso@datedanish\else
    \def\datedanish{\iso@datedanish}%
  \fi
}
```

⟨/danish⟩

D.3 Language definition file english.idf

`\iso@languageloaded` Define the command `\iso@languageloaded` in order to enable `isodate.sty` to determine if at least one language is loaded.

```
⟨*english⟩
\let\iso@languageloaded\active
```

`\month@english` Prints the name of today's month in the long form for the original date format.

```
\def\month@english{\ifcase\month\or
    January\or February\or March\or April\or May\or June\or
    July\or August\or September\or October\or November\or December\fi}
```

British and American English dates are very different. So handle them seperately. It might have been easier to put them in different files but I wanted to organize my files analogous to babel.

First handle British English.

```
\ifthenelse{\equal{\CurrentOption}{english}\OR
            \equal{\CurrentOption}{british}\OR
            \equal{\CurrentOption}{UKenglish}}{%
 \typeout{Define commands for English date format}
```

`\day@english` Prints today's day for the original date format.

```
 \def\day@english{%
   \ifiso@cleanlook
     \day
   \else
     \ifcase\day\or
       1st\or 2nd\or 3rd\or 4th\or 5th\or
       6th\or 7th\or 8th\or 9th\or 10th\or
       11th\or 12th\or 13th\or 14th\or 15th\or
       16th\or 17th\or 18th\or 19th\or 20th\or
       21st\or 22nd\or 23rd\or 24th\or 25th\or
       26th\or 27th\or 28th\or 29th\or 30th\or
       31st%
     \fi
   \fi
 }
```

`\iso@printmonthday@english` Prints the month and the day given as two arguments (`{mm}{dd}`) in the current date format.

```
 \def\iso@printmonthday@english#1#2{%
```

Numeric and short date format: `dd/mm/`

```
   \ifthenelse{\equal{\iso@dateformat}{iso}\OR
               \equal{\iso@dateformat}{TeX}}{%
     \iso@printmonthday@int{#1}{#2}%
   }{%
     \ifthenelse{\equal{\iso@dateformat}{numeric}\OR
                 \equal{\iso@dateformat}{short}}{%
```

```
      \ifiso@doprintday
        \iso@printday{#2}/%
      \fi
      \iso@printmonth{#1}%
    }{%
```

Original date format: `ddd mmm`

```
      \ifthenelse{\equal{\iso@dateformat}{orig}\OR
                  \equal{\iso@dateformat}{shortorig}}{%
        \begingroup
        \edef\lday{#2}\def\day{\lday}%
        \edef\lmonth{#1}\def\month{\lmonth}%
        \ifiso@doprintday
          \day@english\iso@monthsep\@empty
        \fi
        \month@english
        \endgroup
      }{}%
    }%
  }%
}
```

`\iso@printdate@english` Prints the date given as three arguments (`{yyyy}{mm}{dd}`) in the actual date format.

```
\def\iso@printdate@english#1#2#3{%
  \ifthenelse{\equal{\iso@dateformat}{iso}\OR
              \equal{\iso@dateformat}{TeX}}{%
    \iso@printdate@int{#1}{#2}{#3}%
  }{%
```

ISO date format: `yyyy-\iso@printmonthday@english`

```
    \iso@printmonthday@english{\number#2}{\number#3}%
```

Numeric date format: `\iso@printmonthday@english yyyy`

```
    \ifiso@printyear
      \ifthenelse{\equal{\iso@dateformat}{orig}\OR
                  \equal{\iso@dateformat}{shortorig}}{%
      }{%
        /%
      }%
      \ifthenelse{\equal{\iso@dateformat}{numeric}}{%
        \iso@yearfour{\number#1}%
      }{%
```

Original date format: `\iso@printmonthday@english~yyyy`

```
        \ifthenelse{\equal{\iso@dateformat}{orig}}{%
          \iso@yearsep\iso@yearfour{\number#1}%
        }{%
```

Short original date format: `\iso@printmonthday@english~yy`

```
          \ifthenelse{\equal{\iso@dateformat}{shortorig}}{%
```

```
624               \iso@yearsep\iso@twodigitsign\iso@yeartwo{\number#1}%
625             }{%
```

Short date format: `\iso@printmonthday@english yy`

```
626               \ifthenelse{\equal{\iso@dateformat}{short}}{%
627                 \iso@yeartwo{\number#1}%
628               }{}%
629             }%
630           }%
631         }%
632       \fi
633     }%
634   }
```

`\iso@printdate@UKenglish` Just a second name for `\iso@printdate@UKenglish`.

```
635   \def\iso@printdate@UKenglish{\iso@printdate@english}
636   \def\iso@printdate@british{\iso@printdate@english}
```

`\iso@dateenglish` This command redefines the `\today` command to print in the actual date format.

```
637   \def\iso@dateenglish{%
638     \def\today{\iso@printdate@english{\year}{\month}{\day}}}%
```

`\iso@daterange@...` Define date-range commands for dialects of English.

```
639   \expandafter\def\csname iso@daterange@\CurrentOption\endcsname{%
640     \iso@daterange@english}%
```

`\iso@daterange@english` This command takes six arguments (`{yyyy1}{mm1}{dd1}{yyyy2}{mm2}{dd2}`) and prints the corrosponding date range in the actual date format.

```
641   \def\iso@daterange@english#1#2#3#4#5#6{%
```

ISO or LaTeX date format.

```
642     \ifthenelse{\equal{\iso@dateformat}{iso}\OR
643                 \equal{\iso@dateformat}{TeX}}{%
```

Print the start date.

```
644       \iso@daterange@int{#1}{#2}{#3}{#4}{#5}{#6}%
645     }{%
```

Numeric, short, or original date format.

If year and month are equal, only print the day of the start date. If only the year is equal, only print month and day of the start date. Otherwise print the whole start date.

```
646       \ifthenelse{\equal{\number#1}{\number#4}}{%
647         \ifthenelse{\equal{\number#2}{\number#5}}{%
648           \ifiso@doprintday
649             \ifthenelse{\equal{\iso@dateformat}{orig}\OR
650                         \equal{\iso@dateformat}{shortorig}}{%
651               \begingroup
652               \edef\lday{#3}\def\day{\lday}%
653               \day@english
```

```
              \endgroup
            }{%
              \iso@printday{#3}%
            }%
          \else
            \csname iso@printmonthday@\iso@languagename\endcsname{#2}{#3}%
          \fi
        }{%
          \csname iso@printmonthday@\iso@languagename\endcsname{#2}{#3}%
        }%
      }{%
        \csname iso@printdate@\iso@languagename\endcsname{#1}{#2}{#3}%
      }%
```

Print the end date.

```
      \iso@rangesign
      \csname iso@printdate@\iso@languagename\endcsname{#4}{#5}{#6}%
    }%
  }
```

Define the language name that will the active language for isodate if none of the packages babel.sty, german.sty, and ngerman.sty is loaded and if this is the last language that is used for isodate. If one of the above packages is used this definition will be overridden by the command `\languagename` that will always return the current used language.

```
  \def\iso@languagename{english}%
```

The end of the British section.

Second handle Australian and New Zealand.

```
}{%
  \ifthenelse{\equal{\CurrentOption}{australian}\OR
              \equal{\CurrentOption}{newzealand}}{%
    \typeout{Define commands for Australian date format}
```

`\iso@printmonthday@australian` Prints the month and the day given as two arguments (`{mm}{dd}`) in the current date format.

```
    \def\iso@printmonthday@australian#1#2{%
      \ifthenelse{\equal{\iso@dateformat}{iso}\OR
                  \equal{\iso@dateformat}{TeX}}{%
        \iso@printmonthday@int{#1}{#2}%
      }{%
```

Numeric and short date format: `dd/mm/`

```
        \ifthenelse{\equal{\iso@dateformat}{numeric}\OR
                    \equal{\iso@dateformat}{short}}{%
          \ifiso@doprintday
            \iso@printday{#2}/%
          \fi
          \iso@printmonth{#1}%
        }{%
```

Original date format: `ddd mmm`

```
        \ifthenelse{\equal{\iso@dateformat}{orig}\OR
                    \equal{\iso@dateformat}{shortorig}}{%
          \begingroup
          \edef\lmonth{#1}\def\month{\lmonth}%
          \ifiso@doprintday
            \iso@printday{#2}\iso@monthsep\@empty
          \fi
          \month@english
          \endgroup
        }{}%
      }%
    }%
  }
```

`\iso@printdate@australian` Prints the date given as three arguments (`{yyyy}{mm}{dd}`) in the actual date format.

```
  \def\iso@printdate@australian#1#2#3{%
    \ifthenelse{\equal{\iso@dateformat}{iso}\OR
                \equal{\iso@dateformat}{TeX}}{%
      \iso@printdate@int{#1}{#2}{#3}%
    }{%
      \iso@printmonthday@australian{\number#2}{\number#3}%
```

Numeric date format: `\iso@printmonthday@australian yyyy`

```
      \ifiso@printyear
        \ifthenelse{\equal{\iso@dateformat}{orig}\OR
                    \equal{\iso@dateformat}{shortorig}}{%
        }{%
          /%
        }%
        \ifthenelse{\equal{\iso@dateformat}{numeric}}{%
          \iso@yearfour{\number#1}%
        }{%
```

Original date format: `\iso@printmonthday@australian~yyyy`

```
          \ifthenelse{\equal{\iso@dateformat}{orig}}{%
            \iso@yearsep\iso@yearfour{\number#1}%
          }{%
```

Short original date format: `\iso@printmonthday@australian~yy`

```
            \ifthenelse{\equal{\iso@dateformat}{shortorig}}{%
              \iso@yearsep\iso@twodigitsign\iso@yeartwo{\number#1}%
            }{%
```

Short date format: `\iso@printmonthday@australian yy`

```
              \ifthenelse{\equal{\iso@dateformat}{short}}{%
                \iso@yeartwo{\number#1}%
              }{}%
            }%
          }%
```

```
727           }%
728         \fi
729       }%
730     }
```

`\iso@printdate@newzealand` Just a second name for `\iso@printdate@UKenglish`.

```
731     \def\iso@printdate@newzealand{\iso@printdate@australian}
```

`\iso@dateaustralian` This command redefines the `\today` command to print in the actual date format.

```
732     \def\iso@dateaustralian{%
733       \def\today{\iso@printdate@australian{\year}{\month}{\day}}}%
```

`\iso@daterange@...` Define date-range commands for dialects of Australian.

```
734     \expandafter\def\csname iso@daterange@\CurrentOption\endcsname{%
735       \iso@daterange@australian}%
```

`\iso@daterange@australian` This command takes six arguments (`{yyyy1}{mm1}{dd1}{yyyy2}{mm2}{dd2}`) and prints the corrosponding date range in the actual date format.

```
736     \def\iso@daterange@australian#1#2#3#4#5#6{%
```

ISO or LaTeX date format.

```
737       \ifthenelse{\equal{\iso@dateformat}{iso}\OR
738                   \equal{\iso@dateformat}{TeX}}{%
```

Print the start date.

```
739         \iso@daterange@int{#1}{#2}{#3}{#4}{#5}{#6}%
740       }{%
```

Numeric, short, or original date format.

If year and month are equal, only print the day of the start date. If only the year is equal, only print month and day of the start date. Otherwise print the whole start date.

```
741         \ifthenelse{\equal{\number#1}{\number#4}}{%
742           \ifthenelse{\equal{\number#2}{\number#5}}{%
743             \ifiso@doprintday
744               \iso@printday{#3}%
745             \else
746               \csname iso@printmonthday@\iso@languagename\endcsname{#2}{#3}%
747             \fi
748           }{%
749             \iso@printmonthday@australian{#2}{#3}%
750           }%
751         }{%
752           \csname iso@printdate@\iso@languagename\endcsname{#1}{#2}{#3}%
753         }%
```

Print the end date.

```
754         \iso@rangesign
755         \csname iso@printdate@\iso@languagename\endcsname{#4}{#5}{#6}%
756       }%
757     }
```

Define the language name that will the active language for isodate if none of the packages babel.sty, german.sty, and ngerman.sty is loaded and if this is the last language that is used for isodate. If one of the above packages is used this definition will be overridden by the command `\languagename` that will always return the current used language.

```
\def\iso@languagename{australian}%
```

The end of the Australian section.

Third, handle American.

```
}{%
  \typeout{Define commands for American date format}
```

`\iso@printmonthday@american` Prints the month and the day given as two arguments (`{mm}{dd}`) in the current date format.

```
\def\iso@printmonthday@american#1#2{%
  \ifthenelse{\equal{\iso@dateformat}{iso}\OR
              \equal{\iso@dateformat}{TeX}}{%
    \iso@printmonthday@int{#1}{#2}%
  }{%
```

Numeric and short date format: `mm/dd/`

```
    \ifthenelse{\equal{\iso@dateformat}{numeric}\OR
                \equal{\iso@dateformat}{short}}{%
      \iso@printmonth{#1}%
      \ifiso@doprintday
        /\iso@printday{#2}%
      \fi
    }{%
```

Original date format: `mmm d`

```
      \ifthenelse{\equal{\iso@dateformat}{orig}\OR
                  \equal{\iso@dateformat}{shortorig}}{%
        \begingroup%
        \edef\lmonth{#1}%
        \def\month{\lmonth}%
        \month@english%
        \endgroup
        \ifiso@doprintday
          \iso@daysep\iso@printday{#2}%
        \fi
      }{}%
    }%
  }%
}
```

`\iso@printdate@american` Prints the date given as three arguments (`{yyyy}{mm}{dd}`) in the actual date format.

```
\def\iso@printdate@american#1#2#3{%
  \ifthenelse{\equal{\iso@dateformat}{iso}\OR
              \equal{\iso@dateformat}{TeX}}{%
```

```
      \iso@printdate@int{#1}{#2}{#3}%
    }{%
      \iso@printmonthday@american{\number#2}{\number#3}%
```
Numeric date format: `\iso@printmonthday@american yyyy`
```
      \ifiso@printyear
        \ifthenelse{\equal{\iso@dateformat}{orig}\OR
                    \equal{\iso@dateformat}{shortorig}}{%
        }{%
          /%
        }%
        \ifthenelse{\equal{\iso@dateformat}{numeric}}{%
          \iso@yearfour{\number#1}%
        }{%
```
Original date format: `\iso@printmonthday@american,~yyyy`
```
          \ifthenelse{\equal{\iso@dateformat}{orig}}{%
            \ifiso@doprintday,\fi
            \iso@yearsep\iso@yearfour{\number#1}%
          }{%
```
Short original date format: `\iso@printmonthday@american,~yyyy`
```
            \ifthenelse{\equal{\iso@dateformat}{shortorig}}{%
              \ifiso@doprintday,\fi
              \iso@yearsep\iso@twodigitsign\iso@yeartwo{\number#1}%
            }{%
```
Short date format: `\iso@printmonthday@american yy`
```
              \ifthenelse{\equal{\iso@dateformat}{short}}{%
                \iso@yeartwo{\number#1}%
              }{}%
            }%
          }%
        }%
      \fi
    }%
  }
```

`\iso@printdate@USenglish` Just a second name for `\iso@printdate@UKamerican`.
```
  \def\iso@printdate@USenglish{\iso@printdate@american}
```

`\iso@dateamerican` This command redefines the `\today` command to print in the actual date format.
```
  \def\iso@dateamerican{%
    \def\today{\iso@printdate@american{\year}{\month}{\day}}}%
```

`\iso@daterange@...` Define date-range commands for dialects of American.
```
  \expandafter\def\csname iso@daterange@\CurrentOption\endcsname{%
    \iso@daterange@american}%
```

`\iso@daterange@american` This command takes six arguments (`{yyyy1}{mm1}{dd1}{yyyy2}{mm2}{dd2}`) and prints the corrosponding date range in the actual date format.
```
  \def\iso@daterange@american#1#2#3#4#5#6{%
```

ISO or LaTeX date format.

```
      \ifthenelse{\equal{\iso@dateformat}{iso}\OR
                  \equal{\iso@dateformat}{TeX}}{%
```

Print the start date.

```
        \iso@daterange@int{#1}{#2}{#3}{#4}{#5}{#6}%
      }{%
```

Original date format.

If year and month are equal, print `mmm d1 to d2, yyyy`. If only the year is equal, print `mmm1 d1 to mmm2 d2, yyyy`. Otherwise print the whole start and end date.

```
        \ifthenelse{\equal{\iso@dateformat}{orig}\OR
                    \equal{\iso@dateformat}{shortorig}}{%
          \ifthenelse{\equal{\number#1}{\number#4}}{%
            \ifthenelse{\equal{\number#2}{\number#5}}{%
              \iso@printmonthday@american{#2}{#3}%
              \iso@rangesign
              \ifiso@doprintday
                \iso@printday{#6},\iso@yearsep\@empty
              \else
                \iso@printmonthday@american{#5}{#6}\iso@yearsep\@empty
              \fi
              \ifthenelse{\equal{\iso@dateformat}{orig}}{%
                \iso@yearfour{\number#4}%
              }{%
                \iso@twodigitsign\iso@yeartwo{\number#4}%
              }%
            }{%
              \iso@printmonthday@american{#2}{#3}%
              \iso@rangesign
              \csname iso@printdate@\iso@languagename\endcsname{%
                #4}{#5}{#6}%
            }%
          }{%
            \csname iso@printdate@\iso@languagename\endcsname{#1}{#2}{#3}%
            \iso@rangesign%
            \csname iso@printdate@\iso@languagename\endcsname{#4}{#5}{#6}%
          }%
        }{%
```

Numeric or short date format.

If year and month are equal, only print the day of the end date. Otherwise print the whole end date.

```
          \ifthenelse{\equal{\number#1}{\number#4}}{%
            \iso@printmonthday@american{#2}{#3}%
          }{%
            \csname iso@printdate@\iso@languagename\endcsname{#1}{#2}{#3}%
          }%
```

Print the end date.

```
862           \iso@rangesign
863           \csname iso@printdate@\iso@languagename\endcsname{#4}{#5}{#6}%
864         }%
865       }%
866     }
```

Define the language name that will the active language for isodate if none of the packages babel.sty, german.sty, and ngerman.sty is loaded and if this is the last language that is used for isodate. If one of the above packages is used this definition will be overridden by the command `\languagename` that will always return the current used language.

```
867     \def\iso@languagename{american}%
```

The end of the American section.

```
868   }
869 }
```

`\iso@rangesign@...` Sets the word between start and end date in a date range to ‘ to’.

```
870 \expandafter\def\csname iso@rangesign@\CurrentOption\endcsname{~to~}
```

Redefine the command `date`language that is used by babel.sty, german.sty, and ngerman.sty to switch to the original English/American date format to enable the use of different date formats. This has to be done after the preamble in order to ensure to overwrite the babel command.

Do this only if `\iso@date`language is defined.

```
871 \AtBeginDocument{%
872   \ifx\undefined\iso@dateenglish\else
873     \def\dateenglish{\iso@dateenglish}%
874     \def\datebritish{\iso@dateenglish}%
875     \def\dateUKenglish{\iso@dateenglish}%
876   \fi
877   \ifx\undefined\iso@dateaustralian\else
878     \def\dateaustralian{\iso@dateaustralian}%
879     \def\datenewzealand{\iso@dateaustralian}%
880   \fi
881   \ifx\undefined\iso@dateamerican\else
882     \def\dateamerican{\iso@dateamerican}%
883     \def\dateUSenglish{\iso@dateamerican}%
884   \fi
885 }
886 ⟨/english⟩
```

D.4 Language definition file french.idf

`\iso@languageloaded` Define the command `\iso@languageloaded` in order to enable `isodate.sty` to determine if at least one language is loaded.

```
887 ⟨*french⟩
888 \let\iso@languageloaded\active
889 \typeout{Define commands for French date format}
```

```
\def\month@french{\ifcase\month\or
  janvier\or f\'evrier\or mars\or avril\or mai\or juin\or
  juillet\or ao\^ut\or septembre\or octobre\or novembre\or
  d\'ecembre\fi}
\def\iso@printmonthday@french#1#2{%
  \ifthenelse{\equal{\iso@dateformat}{iso}\OR
              \equal{\iso@dateformat}{TeX}}{%
    \iso@printmonthday@int{#1}{#2}%
  }{%
    \ifthenelse{\equal{\iso@dateformat}{numeric}\OR
                \equal{\iso@dateformat}{short}}{%
      \ifiso@doprintday
        \iso@printday{#2}/%
      \fi
      \iso@printmonth{#1}%
    }{%
      \ifthenelse{\equal{\iso@dateformat}{orig}\OR
                  \equal{\iso@dateformat}{shortorig}}{%
        \begingroup
        \edef\lday{#2}\edef\day{\lday}%
        \edef\lmonth{#1}\def\month{\lmonth}%
        \ifiso@doprintday
          \number\day\ifnum1=\day \noexpand\ier\fi\iso@monthsep
        \fi
        \month@french
        \endgroup
      }{}%
    }%
  }%
}
\def\iso@printdate@french#1#2#3{%
  \ifthenelse{\equal{\iso@dateformat}{iso}\OR
              \equal{\iso@dateformat}{TeX}}{%
    \iso@printdate@int{#1}{#2}{#3}%
  }{%
    \iso@printmonthday@french{\number#2}{\number#3}%
    \ifiso@printyear
      \ifthenelse{\equal{\iso@dateformat}{orig}\OR
                  \equal{\iso@dateformat}{shortorig}}{%
      }{%
        /%
      }%
      \ifthenelse{\equal{\iso@dateformat}{numeric}}{%
        \iso@yearfour{\number#1}%
      }{%
        \ifthenelse{\equal{\iso@dateformat}{orig}}{%
          \iso@yearsep\iso@yearfour{\number#1}%
        }{%
          \ifthenelse{\equal{\iso@dateformat}{shortorig}}{%
```

```
            \iso@yearsep\iso@twodigitsign\iso@yeartwo{\number#1}%
          }{%
            \ifthenelse{\equal{\iso@dateformat}{short}}{%
              \iso@yeartwo{\number#1}%
            }{}%
          }%
        }%
      }%
    \fi
  }%
}
\def\iso@datefrench{%
  \def\today{\iso@printdate@french{\year}{\month}{\day}}}%
```

`\iso@daterange@...` Define date-range commands for dialects.

```
\expandafter\def\csname iso@daterange@\CurrentOption\endcsname{%
  \iso@daterange@french}%
\def\iso@daterange@french#1#2#3#4#5#6{%
  \ifthenelse{\equal{\iso@dateformat}{iso}\OR
              \equal{\iso@dateformat}{TeX}}{%
    \iso@daterange@int{#1}{#2}{#3}{#4}{#5}{#6}%
  }{%
    \ifthenelse{\equal{\number#1}{\number#4}}{%
      \ifthenelse{\equal{\number#2}{\number#5}}{%
        \ifiso@doprintday
          \ifthenelse{\equal{\iso@dateformat}{orig}}{%
            \begingroup
            \edef\lday{#3}\edef\day{\lday}%
            \number\day\ifnum1=\day \noexpand\ier\fi
            \endgroup
          }{%
            \iso@printday{#3}%
          }%
        \else
          \csname iso@printmonthday@\iso@languagename\endcsname{#2}{#3}%
        \fi
      }{%
        \iso@printmonthday@french{#2}{#3}%
      }%
    }{%
      \csname iso@printdate@\iso@languagename\endcsname{#1}{#2}{#3}%
    }%
    \iso@rangesign
    \csname iso@printdate@\iso@languagename\endcsname{#4}{#5}{#6}%
  }%
}
\expandafter\def\csname iso@rangesign@\CurrentOption\endcsname{~au~}
```

Define the language name that will the active language for isodate if none of the packages babel.sty, german.sty, and ngerman.sty is loaded and if this is the last language that is used for isodate. If one of the above packages is used this definition will be overridden by the command `\languagename` that will always return the current used language.

```
\def\iso@languagename{french}%
```

`\datefrenchb` has to be defined additionally because babel starts with language frenchb instead of french.

```
\AtBeginDocument{%
  \ifx\undefined\iso@datefrench\else
    \def\datefrench{\iso@datefrench}%
    \def\datefrenchb{\iso@datefrench}%
  \fi
}
⟨/french⟩
```

D.5 Language definition file german.idf

`\iso@languageloaded` Define the command `\iso@languageloaded` in order to enable `isodate.sty` to determine if at least one language is loaded.

```
⟨*german⟩
\let\iso@languageloaded\active
\typeout{Define commands for German date format (\CurrentOption)}
```

Define spaces between day and month resp. month and year. `dm` stands for day-month and `my` for month-year. The defaults are taken from the Duden [2].

```
\def\iso@dmsepgerman{\,}%
\def\iso@mylongsepgerman{~}%
\def\iso@myshortsepgerman{\,}%
\def\iso@mylongsepnodaygerman{}%
\def\iso@myshortsepnodaygerman{}%
```

`\daymonthsepgerman` Change space between day and month in numeric date formats for the German language. The only parameter is the new spacing.

```
\DeclareRobustCommand*\daymonthsepgerman[1]{\def\iso@dmsepgerman{#1}}
%    \begin{macrocode}
% \end{macro}
% \begin{macro}{\monthyearsepgerman}
% Change space between month and year in numeric date formats for the
% German language. The first parameter is the new spacing for the long
% format and the second for the short format.
%    \begin{macrocode}
\DeclareRobustCommand*\monthyearsepgerman[2]{%
  \def\iso@mylongsepgerman{#1}%
  \def\iso@myshortsepgerman{#2}}
\DeclareRobustCommand*\monthyearsepnodaygerman[2]{%
  \def\iso@mylongsepnodaygerman{#1}%
  \def\iso@myshortsepnodaygerman{#2}}
```

```
\def\month@german{\ifcase\month\or
  Januar\or Februar\or M\"arz\or April\or Mai\or Juni\or
  Juli\or August\or September\or Oktober\or November\or Dezember\fi}
\def\month@ngerman{\month@german}
\def\month@austrian{\ifnum1=\month
  J\"anner\else \month@german\fi}
\def\month@naustrian{\month@austrian}

\@namedef{iso@printmonthday@\CurrentOption}#1#2{%
  \ifthenelse{\equal{\iso@dateformat}{iso}\OR
              \equal{\iso@dateformat}{TeX}}{%
    \iso@printmonthday@int{#1}{#2}%
  }{%
    \ifthenelse{\equal{\iso@dateformat}{numeric}\OR
                \equal{\iso@dateformat}{short}}{%
      \ifiso@doprintday
        \iso@printday{#2}.\iso@dmsepgerman
      \fi
      \iso@printmonth{#1}%
    }{%
      \ifthenelse{\equal{\iso@dateformat}{orig}\OR
                  \equal{\iso@dateformat}{shortorig}}{%
        \ifiso@doprintday
          \iso@printday{#2}.\iso@monthsep\@empty
        \fi
        \begingroup
        \edef\lmonth{#1}%
        \def\month{\lmonth}\csname month@\iso@languagename\endcsname%
        \endgroup
      }{}%
    }%
  }%
}

\@namedef{iso@printdate@\CurrentOption}#1#2#3{%
  \ifthenelse{\equal{\iso@dateformat}{iso}\OR
              \equal{\iso@dateformat}{TeX}}{%
    \iso@printdate@int{#1}{#2}{#3}%
  }{%
    \csname iso@printmonthday@\iso@languagename\endcsname{%
      \number#2}{\number#3}%
    \ifiso@printyear
      \ifthenelse{\equal{\iso@dateformat}{orig}\OR
                  \equal{\iso@dateformat}{shortorig}}{%
      }{%
        \ifiso@doprintday.\else/\fi
      }%
      \ifthenelse{\equal{\iso@dateformat}{numeric}}{%
        \ifiso@doprintday
          \iso@mylongsepgerman\@empty
        \else
```

```
          \iso@mylongsepnodaygerman\@empty
        \fi
        \iso@yearfour{\number#1}%
      }{%
        \ifthenelse{\equal{\iso@dateformat}{orig}}{%
          \iso@yearsep\iso@yearfour{\number#1}%
        }{%
          \ifthenelse{\equal{\iso@dateformat}{shortorig}}{%
            \iso@yearsep\iso@twodigitsign\iso@yeartwo{\number#1}%
          }{%
            \ifthenelse{\equal{\iso@dateformat}{short}}{%
              \ifiso@doprintday
                \iso@myshortsepgerman\@empty
              \else
                \iso@myshortsepnodaygerman\@empty
              \fi
              \iso@yeartwo{\number#1}%
            }{}%
          }%
        }%
      }%
    \fi
  }%
}
\@namedef{iso@daterange@\CurrentOption}#1#2#3#4#5#6{%
  \ifthenelse{\equal{\iso@dateformat}{iso}\OR
              \equal{\iso@dateformat}{TeX}}{%
    \iso@daterange@int{#1}{#2}{#3}{#4}{#5}{#6}%
  }{%
    \ifthenelse{\equal{\number#1}{\number#4}}{%
      \ifthenelse{\equal{\number#2}{\number#5}}{%
        \ifiso@doprintday
          \iso@printday{#3}.%
        \else
          \csname iso@printmonthday@\iso@languagename\endcsname{#2}{#3}%
        \fi
      }{%
          \csname iso@printmonthday@\iso@languagename\endcsname{#2}{#3}%
        }%
    }{%
        \csname iso@printdate@\iso@languagename\endcsname{#1}{#2}{#3}%
      }%
    \iso@rangesign
    \csname iso@printdate@\iso@languagename\endcsname{#4}{#5}{#6}%
  }%
}
\expandafter\def\csname iso@rangesign@\CurrentOption\endcsname{~bis~}
\ifthenelse{\equal{\CurrentOption}{german}}{%
```

```
    \def\iso@dategerman{%
      \def\today{\iso@printdate@german{\year}{\month}{\day}}}%
```

Define the language name that will the active language for isodate if none of the packages babel.sty, german.sty, and ngerman.sty is loaded and if this is the last language that is used for isodate. If one of the above packages is used this definition will be overridden by the command `\languagename` that will always return the current used language.

```
    \def\iso@languagename{german}%
  }{%
  \ifthenelse{\equal{\CurrentOption}{ngerman}}{%
    \def\iso@datengerman{%
      \def\today{\iso@printdate@ngerman{\year}{\month}{\day}}}%
```

Define the language name that will the active language for isodate if none of the packages babel.sty, german.sty, and ngerman.sty is loaded and if this is the last language that is used for isodate. If one of the above packages is used this definition will be overridden by the command `\languagename` that will always return the current used language.

```
    \def\iso@languagename{ngerman}%
  }{%
  \ifthenelse{\equal{\CurrentOption}{austrian}}{%
    \def\iso@dateaustrian{%
      \def\today{\iso@printdate@austrian{\year}{\month}{\day}}}%
```

Define the language name that will the active language for isodate if none of the packages babel.sty, german.sty, and ngerman.sty is loaded and if this is the last language that is used for isodate. If one of the above packages is used this definition will be overridden by the command `\languagename` that will always return the current used language.

```
    \def\iso@languagename{austrian}%
  }{%
  \ifthenelse{\equal{\CurrentOption}{naustrian}}{%
    \def\iso@datenaustrian{%
      \def\today{\iso@printdate@naustrian{\year}{\month}{\day}}}%
```

Define the language name that will the active language for isodate if none of the packages babel.sty, german.sty, and ngerman.sty is loaded and if this is the last language that is used for isodate. If one of the above packages is used this definition will be overridden by the command `\languagename` that will always return the current used language.

```
    \def\iso@languagename{naustrian}%
  }{%
  }}}}
```

Redefine the command `date`language that is used by babel.sty, german.sty, and ngerman.sty to switch to the original German date format to enable the use of different date formats. This has to be done after the preamble in order to ensure to overwrite the babel command.

Do this only if \iso@datelanguage is defined.

```
\AtBeginDocument{%
  \ifx\undefined\iso@dategerman\else
    \def\dategerman{\iso@dategerman}%
  \fi
  \ifx\undefined\iso@datengerman\else
    \def\datengerman{\iso@datengerman}%
  \fi
  \ifx\undefined\iso@dateaustrian\else
    \def\dateaustrian{\iso@dateaustrian}%
  \fi
  \ifx\undefined\iso@datenaustrian\else
    \def\datenaustrian{\iso@datenaustrian}%
  \fi
}
```

⟨/german⟩

D.6 Language definition file italian.idf

\iso@languageloaded Define the command \iso@languageloaded in order to enable isodate.sty to determine if at least one language is loaded.

⟨∗italian⟩

```
\let\iso@languageloaded\active
\typeout{Define commands for Italian date format}

\def\month@italian{\ifcase\month\or
  gennaio\or febbraio\or marzo\or aprile\or maggio\or giugno\or
  luglio\or agosto\or settembre\or ottobre\or novembre\or
  dicembre\fi}

\def\iso@printmonthday@italian#1#2{%
  \ifthenelse{\equal{\iso@dateformat}{iso}\OR
              \equal{\iso@dateformat}{TeX}}{%
    \iso@printmonthday@int{#1}{#2}%
  }{%
    \ifthenelse{\equal{\iso@dateformat}{numeric}\OR
                \equal{\iso@dateformat}{short}}{%
      \ifiso@doprintday
        \iso@printday{#2}/%
      \fi
      \iso@printmonth{#1}%
    }{%
      \ifthenelse{\equal{\iso@dateformat}{orig}\OR
                  \equal{\iso@dateformat}{shortorig}}{%
        \begingroup
        \edef\lday{#2}\edef\day{\lday}%
        \edef\lmonth{#1}\def\month{\lmonth}%
        \ifiso@doprintday
          \number\day\ifnum1=\day \noexpand\textordmasculine\fi
          \iso@monthsep
```

```
        \fi
        \month@italian
        \endgroup
      }{}%
    }%
  }%
}
\def\iso@printdate@italian#1#2#3{%
  \ifthenelse{\equal{\iso@dateformat}{iso}\OR
    \equal{\iso@dateformat}{TeX}}{%
    \iso@printdate@int{#1}{#2}{#3}%
  }{%
    \iso@printmonthday@italian{\number#2}{\number#3}%
    \ifiso@printyear
      \ifthenelse{\equal{\iso@dateformat}{orig}\OR
                  \equal{\iso@dateformat}{shortorig}}{%
      }{%
        /%
      }%
      \ifthenelse{\equal{\iso@dateformat}{numeric}}{%
        \iso@yearfour{\number#1}%
      }{%
        \ifthenelse{\equal{\iso@dateformat}{orig}}{%
          \iso@yearsep\iso@yearfour{\number#1}%
        }{%
          \ifthenelse{\equal{\iso@dateformat}{shortorig}}{%
            \iso@yearsep\iso@twodigitsign\iso@yeartwo{\number#1}%
          }{%
            \ifthenelse{\equal{\iso@dateformat}{short}}{%
              \iso@yeartwo{\number#1}%
            }{}%
          }%
        }%
      }%
    \fi
  }%
}
\def\iso@dateitalian{%
  \def\today{\iso@printdate@italian{\year}{\month}{\day}}}%
```

`\iso@daterange@...` Define date-range commands for dialects.

```
\expandafter\def\csname iso@daterange@\CurrentOption\endcsname{%
  \iso@daterange@italian}%

\def\iso@daterange@italian#1#2#3#4#5#6{%
  \ifthenelse{\equal{\iso@dateformat}{iso}\OR
              \equal{\iso@dateformat}{TeX}}{%
    \iso@daterange@int{#1}{#2}{#3}{#4}{#5}{#6}%
  }{%
```

```
    \ifthenelse{\equal{\number#1}{\number#4}}{%
      \ifthenelse{\equal{\number#2}{\number#5}}{%
        \ifiso@doprintday
          \ifthenelse{\equal{\iso@dateformat}{orig}}{%
            \begingroup
            \edef\lday{#3}\edef\day{\lday}%
            \number\day\ifnum1=\day \noexpand\textordmasculine\fi
            \endgroup
          }{%
            \iso@printday{#3}%
          }%
        \else
          \iso@printmonthday@italian{#2}{#3}%
        \fi
      }{%
        \iso@printmonthday@italian{#2}{#3}%
      }%
    }{%
      \csname iso@printdate@\iso@languagename\endcsname{#1}{#2}{#3}%
    }%
    \iso@rangesign
    \csname iso@printdate@\iso@languagename\endcsname{#4}{#5}{#6}%
  }%
}
\expandafter\def\csname iso@rangesign@\CurrentOption\endcsname{~al~}
```

Define the language name that will the active language for isodate if none of the packages babel.sty, german.sty, and ngerman.sty is loaded and if this is the last language that is used for isodate. If one of the above packages is used this definition will be overridden by the command `\languagename` that will always return the current used language.

```
\def\iso@languagename{italian}%
\AtBeginDocument{%
  \ifx\undefined\iso@dateitalian\else
    \def\dateitalian{\iso@dateitalian}%
  \fi
}
```

⟨/italian⟩

D.7 Language definition file norsk.idf

This file was provided by Svend Tollak Munkejord (svend.t.munkejord@energy.sintef.no).

`\iso@languageloaded` Define the command `\iso@languageloaded` in order to enable `isodate.sty` to determine if at least one language is loaded.

```
⟨*norsk⟩
\let\iso@languageloaded\active
\typeout{Define commands for Norwegian date format}
```

`\month@norsk` Prints the name of today's month in the long form for the original date format.

```
\def\month@norsk{\ifcase\month\or
    januar\or februar\or mars\or april\or mai\or juni\or
    juli\or august\or september\or oktober\or november\or desember\fi}
```

`\iso@printmonthday@norsk` Prints the month and the day given as two arguments (`{mm}{dd}`) in the current date format.

```
\def\iso@printmonthday@norsk#1#2{%
  \ifthenelse{\equal{\iso@dateformat}{iso}\OR
              \equal{\iso@dateformat}{TeX}}{%
    \iso@printmonthday@int{#1}{#2}%
  }{%
```

Numeric and short date format: `dd/mm/`

```
    \ifthenelse{\equal{\iso@dateformat}{numeric}\OR
                \equal{\iso@dateformat}{short}}{%
      \ifiso@doprintday
        \iso@printday{#2}/%
      \fi
      \iso@printmonth{#1}%
    }{%
```

Original date format: `d. mmm`

```
      \ifthenelse{\equal{\iso@dateformat}{orig}\OR
                  \equal{\iso@dateformat}{shortorig}}{%
        \ifiso@doprintday
          \iso@printday{#2}.\iso@monthsep
        \fi
        \begingroup
        \edef\lmonth{#1}\def\month{\lmonth}%
        \month@norsk%
        \endgroup
      }{}%
    }%
  }%
}
```

`\iso@printdate@norsk` Prints the date given as three arguments (`{yyyy}{mm}{dd}`) in the actual date format

```
\def\iso@printdate@norsk#1#2#3{%
```

ISO or LaTeXdate format: `yyyy\iso@printmonthday@norsk`

```
  \ifthenelse{\equal{\iso@dateformat}{iso}\OR
              \equal{\iso@dateformat}{TeX}}{%
    \iso@printdate@int{#1}{#2}{#3}%
  }{%
    \iso@printmonthday@norsk{\number#2}{\number#3}%
```

numeric date format: `\iso@printmonthday@norsk yyyy`

```
    \ifiso@printyear
       \ifthenelse{\equal{\iso@dateformat}{orig}\OR
```

```
                  \equal{\iso@dateformat}{shortorig}}{%
      }{%
        /%
      }%
      \ifthenelse{\equal{\iso@dateformat}{numeric}}{%
        \iso@yearfour{\number#1}%
      }{%
```

original date format: `\iso@printmonthday@norsk~yyyy`

```
        \ifthenelse{\equal{\iso@dateformat}{orig}}{%
          \iso@yearsep\iso@yearfour{\number#1}%
        }{%
```

short original date format: `\iso@printmonthday@norsk~yyyy`

```
          \ifthenelse{\equal{\iso@dateformat}{shortorig}}{%
            \iso@yearsep\iso@twodigitsign\iso@yeartwo{\number#1}%
          }{%
```

short date format: `\iso@printmonthday@norsk yy`

```
            \ifthenelse{\equal{\iso@dateformat}{short}}{%
              \iso@yeartwo{\number#1}%
            }{}%
          }%
        }%
      }%
    \fi
  }%
}
```

`\iso@datenorsk` This command redefines the `\today` command to print in the actual date format.

```
\def\iso@datenorsk{%
  \def\today{\iso@printdate@norsk{\year}{\month}{\day}}}%
```

`\iso@daterange@...` Define date-range commands for dialects.

```
\expandafter\def\csname iso@daterange@\CurrentOption\endcsname{%
  \iso@daterange@norsk}%
```

`\iso@daterange@norsk` This command takes six arguments (`{yyyy1}{mm1}{dd1}{yyyy2}{mm2}{dd2}`) and prints the corrosponding date range in the actual date format.

```
\def\iso@daterange@norsk#1#2#3#4#5#6{%
```

ISO or LaTeX date format.

```
  \ifthenelse{\equal{\iso@dateformat}{iso}\OR
              \equal{\iso@dateformat}{TeX}}{%
    \iso@daterange@int{#1}{#2}{#3}{#4}{#5}{#6}%
  }{%
```

Numeric, short, or original date format.

If year and month are equal, only print the day of the start date. If only the year is equal, only print month and day of the start date. Otherwise print the whole start date.

```
    \ifthenelse{\equal{\number#1}{\number#4}}{%
      \ifthenelse{\equal{\number#2}{\number#5}}{%
        \ifiso@doprintday
          \ifthenelse{\equal{\iso@dateformat}{orig}\OR
                      \equal{\iso@dateformat}{shortorig}}{%
            \iso@printday{#3}.%
          }{%
            \iso@printday{#3}%
          }%
        \else
          \iso@printmonthday@norsk{#2}{#3}%
        \fi
      }{%
        \iso@printmonthday@norsk{#2}{#3}%
      }%
    }{%
      \csname iso@printdate@\iso@languagename\endcsname{#1}{#2}{#3}%
    }%
```

Print the end date.

```
    \iso@rangesign
    \csname iso@printdate@\iso@languagename\endcsname{#4}{#5}{#6}%
  }%
}
```

`\iso@rangesign@norsk` Sets the word between start and end date in a date range to ‘ til ’.

```
\expandafter\def\csname iso@rangesign@\CurrentOption\endcsname{~til~}
```

Define the language name that will the active language for isodate if none of the packages babel.sty, german.sty, and ngerman.sty is loaded and if this is the last language that is used for isodate. If one of the above packages is used this definition will be overridden by the command `\languagename` that will always return the current used language.

```
\def\iso@languagename{norsk}%
```

Redefine the command `\datenorsk` that is used by babel to switch to the original Norsk date format to enable the use of different date formats. This has to be done after the preamble in order to ensure to overwrite the babel command.

```
\AtBeginDocument{%
  \ifx\undefined\iso@datenorsk\else
    \def\datenorsk{\iso@datenorsk}%
  \fi
}
```

⟨/norsk⟩

D.8 Language definition file swedish.idf

This file was provided by Christian Schlauer (christian.schlauer@web.de).

`\iso@languageloaded` Define the command `\iso@languageloaded` in order to enable `isodate.sty` to determine if at least one language is loaded.

```
⟨*swedish⟩
\let\iso@languageloaded\active
\typeout{Define commands for Swedish date format}
```

`\month@swedish` Prints the name of today's month in the long form for the original date format.

```
\def\month@swedish{\ifcase\month\or
    januari\or februari\or mars\or april\or maj\or juni\or
    juli\or augusti\or september\or oktober\or november\or december\fi}
```

`\iso@printmonthday@swedish` Prints the month and the day given as two arguments (`{mm}{dd}`) in the current date format.

```
\def\iso@printmonthday@swedish#1#2{%
  \ifthenelse{\equal{\iso@dateformat}{iso}\OR
              \equal{\iso@dateformat}{TeX}}{%
    \iso@printmonthday@int{#1}{#2}%
  }{%
```

Numeric and short date format: `dd/mm/`

```
    \ifthenelse{\equal{\iso@dateformat}{numeric}\OR
                \equal{\iso@dateformat}{short}}{%
      \ifiso@doprintday
        \iso@printday{#2}/%
      \fi
      \iso@printmonth{#1}%
    }{%
```

Original date format: `d. mmm`

```
      \ifthenelse{\equal{\iso@dateformat}{orig}\OR
                  \equal{\iso@dateformat}{shortorig}}{%
        \ifiso@doprintday
          \iso@printday{#2}.\iso@monthsep
        \fi
        \begingroup
        \edef\lmonth{#1}\def\month{\lmonth}%
        \month@swedish%
        \endgroup
      }{}%
    }%
  }%
}
```

`\iso@printdate@swedish` Prints the date given as three arguments (`{yyyy}{mm}{dd}`) in the actual date format

```
\def\iso@printdate@swedish#1#2#3{%
```

ISO or LaTeXdate format: `yyyy\iso@printmonthday@swedish`

```
  \ifthenelse{\equal{\iso@dateformat}{iso}\OR
              \equal{\iso@dateformat}{TeX}}{%
```

```
    \iso@printdate@int{#1}{#2}{#3}%
  }{%
    \iso@printmonthday@swedish{\number#2}{\number#3}%
```

numeric date format: `\iso@printmonthday@swedish yyyy`

```
    \ifiso@printyear
      \ifthenelse{\equal{\iso@dateformat}{orig}\OR
                  \equal{\iso@dateformat}{shortorig}}{%
      }{%
        /%
      }%
      \ifthenelse{\equal{\iso@dateformat}{numeric}}{%
        \iso@yearfour{\number#1}%
      }{%
```

original date format: `\iso@printmonthday@swedish~yyyy`

```
        \ifthenelse{\equal{\iso@dateformat}{orig}}{%
          \iso@yearsep\iso@yearfour{\number#1}%
        }{%
```

short original date format: `\iso@printmonthday@swedish~yy`

```
          \ifthenelse{\equal{\iso@dateformat}{shortorig}}{%
            \iso@yearsep\iso@twodigitsign\iso@yeartwo{\number#1}%
          }{%
```

short date format: `\iso@printmonthday@swedish yy`

```
            \ifthenelse{\equal{\iso@dateformat}{short}}{%
              \iso@yeartwo{\number#1}%
            }{}%
          }%
        }%
      }%
    \fi
  }%
}
```

`\iso@dateswedish` This command redefines the `\today` command to print in the actual date format.

```
\def\iso@dateswedish{%
  \def\today{\iso@printdate@swedish{\year}{\month}{\day}}}%
```

`\iso@daterange@...` Define date-range commands for dialects.

```
\expandafter\def\csname iso@daterange@\CurrentOption\endcsname{%
  \iso@daterange@swedish}%
```

`\iso@daterange@swedish` This command takes six arguments (`{yyyy1}{mm1}{dd1}{yyyy2}{mm2}{dd2}`) and prints the corrosponding date range in the actual date format.

```
\def\iso@daterange@swedish#1#2#3#4#5#6{%
```

ISO or LaTeX date format.

```
  \ifthenelse{\equal{\iso@dateformat}{iso}\OR
              \equal{\iso@dateformat}{TeX}}{%
```

```
    \iso@daterange@int{#1}{#2}{#3}{#4}{#5}{#6}%
  }{%
```

Numeric, short, or original date format.

If year and month are equal, only print the day of the start date. If only the year is equal, only print month and day of the start date. Otherwise print the whole start date.

```
    \ifthenelse{\equal{\number#1}{\number#4}}{%
      \ifthenelse{\equal{\number#2}{\number#5}}{%
        \ifiso@doprintday
          \ifthenelse{\equal{\iso@dateformat}{orig}\OR
                      \equal{\iso@dateformat}{shortorig}}{%
            \iso@printday{#3}.%
          }{%
            \iso@printday{#3}%
          }%
        \else
          \iso@printmonthday@swedish{#2}{#3}%
        \fi
      }{%
        \iso@printmonthday@swedish{#2}{#3}%
      }%
    }{%
      \csname iso@printdate@\iso@languagename\endcsname{#1}{#2}{#3}%
    }%
```

Print the end date.

```
    \iso@rangesign
    \csname iso@printdate@\iso@languagename\endcsname{#4}{#5}{#6}%
  }%
}
```

`\iso@rangesign@swedish` Sets the word between start and end date in a date range to ‘ till ’.

```
\expandafter\def\csname iso@rangesign@\CurrentOption\endcsname{~till~}
```

Define the language name that will the active language for isodate if none of the packages babel.sty, german.sty, and ngerman.sty is loaded and if this is the last language that is used for isodate. If one of the above packages is used this definition will be overridden by the command `\languagename` that will always return the current used language.

```
\def\iso@languagename{swedish}%
```

Redefine the command `\dateswedish` that is used by babel to switch to the original Swedish date format to enable the use of different date formats. This has to be done after the preamble in order to ensure to overwrite the babel command.

```
\AtBeginDocument{%
  \ifx\undefined\iso@dateswedish\else
    \def\dateswedish{\iso@dateswedish}%
  \fi
}
```

⟨/swedish⟩

Change History

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.

L