The Turkish style for babel

Mustafa Burc, Johannes Braams, Javier Bezos

July 5, 2019

1 The Turkish language

The file `turkish.dtx`[1] defines all the language definition macros for the Turkish language[2].

Turkish typographic rules specify that a little 'white space' should be added before the characters '`:`', '`!`' and '`=`'. In order to insert this white space automatically these characters are made 'active'. Also `\frenhspacing` is set.

Typical usage with pdfTEX is:

```
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc} % also latin5
\usepackage[turkish]{babel}
```

The = shorthand is potentially dangerous. You can deactivate with the babel option `shorthands` or with `\shorthandoff`:

```
\usepackage[turkish,shorthands=:!]{babel}
```

This style doesn't handle the fi ligature (yet). You can break it by hand with `f{}i` or `f{\kern0pt}i`, but this can be done automatically, too. With pdfTEX and monolingual documents, use microtype, as for example:

```
\usepackage{microtype}
\DisableLigatures[f]{encoding = *, family = *}
```

With XeTEX, ligatures are handled internally by the font, provided the corresponding feature has been implemented (not all fonts do); e. g.:

```
\usepackage{fontspec}
\setmainfont[Language=Turkish]{Iwona}
```

With LuaTEX you can use either method (remember with microtype you have also to set `Renderer=Basic`, at least at the time of this writing). Alternative approachs with LuaTEX are the setnolig package or a `fea` file (not provided here).

[1] The file described in this section has version number v1.4 and was last revised on 2019/07/05.

[2] Mustafa Burc, `z6001@rziris01.rrz.uni-hamburg.de` provided the code for this file. It is based on the work by Pierre Mackay; Turgut Uyar, `uyar@cs.itu.edu.tr` supplied additional translations in version 1.2j and later. Version 1.3 was prepared by Javier Bezos.

The code

The macro `\LdfInit` takes care of preventing that this file is loaded more than once, checking the category code of the `@` sign, etc.

```
1 ⟨*code⟩
2 \LdfInit{turkish}\captionsturkish
```

When this file is read as an option, i.e. by the `\usepackage` command, `turkish` could be an 'unknown' language in which case we have to make it known. So we check for the existence of `\l@turkish` to see whether we have to do something here.

```
3 \ifx\l@turkish\@undefined
4   \@nopatterns{Turkish}
5   \adddialect\l@turkish0\fi
```

The next step consists of defining commands to switch to (and from) the Turkish language.

`\captionsturkish` The macro `\captionsturkish` defines all strings used in the four standard documentclasses provided with LaTeX.

```
6 \addto\captionsturkish{%
7   \def\prefacename{\"Ons\"oz}%
8   \def\refname{Kaynaklar}%
9   \def\abstractname{\"Ozet}%
10  \def\bibname{Kaynak\c ca}%
11  \def\chaptername{B\"ol\"um}%
12  \def\appendixname{Ek}%
13  \def\contentsname{\.I\c cindekiler}%
14  \def\listfigurename{\c Sekil Listesi}%
15  \def\listtablename{Tablo Listesi}%
16  \def\indexname{Dizin}%
17  \def\figurename{\c Sekil}%
18  \def\tablename{Tablo}%
19  \def\partname{K\i s\i m}%
20  \def\enclname{\.Ili\c sik}%
21  \def\ccname{Di\u ger Al\i c\i lar}%
22  \def\headtoname{Al\i c\i}%
23  \def\pagename{Sayfa}%
24  \def\subjectname{\.Ilgili}%
25  \def\seename{bkz.}%
26  \def\alsoname{ayr\i ca bkz.}%
27  \def\proofname{Kan\i t}%
28  \def\glossaryname{L\"ugat\c ce}% <-- Tentative
29 }%
```

`\dateturkish` The macro `\dateturkish` redefines the command `\today` to produce Turkish dates.

```
30 \def\dateturkish{%
31   \def\today{\number\day~\ifcase\month\or
32     Ocak\or \c Subat\or Mart\or Nisan\or May\i s\or Haziran\or
```

```
    Temmuz\or A\u gustos\or Eyl\"ul\or Ekim\or Kas\i m\or
    Aral\i k\fi
    \space\number\year}}
```

The following code is taken into account only with babel 3.9g and later. Defines case and hyphen mapping, as well as UTF-8 strings. First the Unicode branch.

```
\ifx\BabelLower\@undefined\else
\StartBabelCommands*{turkish}{captions}
  [unicode, charset=utf8, fontenc=EU1 EU2 TU]
  \SetString\prefacename{Önsöz}
  \SetString\refname{Kaynaklar}
  \SetString\abstractname{Özet}
  \SetString\bibname{Kaynakça}
  \SetString\chaptername{Bölüm}
  \SetString\appendixname{Ek}
  \SetString\contentsname{çindekiler}
  \SetString\listfigurename{ekil Listesi}
  \SetString\listtablename{Tablo Listesi}
  \SetString\indexname{Dizin}
  \SetString\figurename{ekil}
  \SetString\tablename{Tablo}
  \SetString\partname{Ksm}
  \SetString\enclname{liik}
  \SetString\ccname{Dier Alclar}
  \SetString\headtoname{Alc}
  \SetString\pagename{Sayfa}
  \SetString\subjectname{lgili}
  \SetString\seename{bkz.}
  \SetString\alsoname{ayrca bkz.}
  \SetString\proofname{Kant}
  \SetString\glossaryname{Lügatçe}% <-- Tentative
  \SetCase
    {\uccode`i=`\relax
     \uccode`=`I\relax}
    {\lccode`=`i\relax
     \lccode`I=`\relax}
  \SetHyphenMap{%
    \BabelLower{`}{`i}%
    \BabelLower{`I}{`}}
\StartBabelCommands*{turkish}{date}
  [unicode, charset=utf8, fontenc=EU1 EU2 TU]
  \SetStringLoop{month#1name}{%
    Ocak,ubat,Mart,Nisan,Mays,Haziran,%
    Temmuz,Austos,Eylül,Ekim,Kasm,Aralk}
```

Now the OT1 branch, only partially, because this encoding is not suited for Turkish (no dotted I).

```
\StartBabelCommands{turkish}{}[ot1enc, fontenc=OT1]
  \SetCase
    {\uccode"10=`I\relax}
    {\lccode`I="10\relax}
```

And finally, the generic branch, using the LICR and assuming T1.

```
\StartBabelCommands*{turkish}{captions}
  \SetString\prefacename{\"Ons\"oz}
  \SetString\refname{Kaynaklar}
  \SetString\abstractname{\"Ozet}
  \SetString\bibname{Kaynak\c ca}
  \SetString\chaptername{B\"ol\"um}
  \SetString\appendixname{Ek}
  \SetString\contentsname{\.I\c cindekiler}
  \SetString\listfigurename{\c Sekil Listesi}
  \SetString\listtablename{Tablo Listesi}
  \SetString\indexname{Dizin}
  \SetString\figurename{\c Sekil}
  \SetString\tablename{Tablo}
  \SetString\partname{K\i s\i m}
  \SetString\enclname{\.Ili\c sik}
  \SetString\ccname{Di\u ger Al\i c\i lar}
  \SetString\headtoname{Al\i c\i}
  \SetString\pagename{Sayfa}
  \SetString\subjectname{\.Ilgili}
  \SetString\seename{bkz.}
  \SetString\alsoname{ayr\i ca bkz.}
  \SetString\proofname{Kan\i t}
  \SetString\glossaryname{L\"ugat\c ce}% <-- Tentative
  \SetCase
    {\uccode`i="9D\relax
     \uccode"19=`I\relax}
    {\lccode"9D=`i\relax
     \lccode`I="19\relax}
  \SetHyphenMap{%
    \BabelLower{"9D}{`i}%
    \BabelLower{`I}{"19}}
\StartBabelCommands*{turkish}{date}
  \SetStringLoop{month#1name}{%
    Ocak,\c Subat,Mart,Nisan,May\i s,Haziran,%
    Temmuz,A\u gustos,Eyl\"ul,Ekim,Kas\i m,Aral\i k}
  \SetString\today{%
    \number\day~\@nameuse{month\romannumeral\month name}%
    \space\number\year}
\EndBabelCommands
\fi
```

`\extrasturkish` `\noextrasturkish`

The macro `\extrasturkish` will perform all the extra definitions needed for the Turkish language. The macro `\noextrasturkish` is used to cancel the actions of `\extrasturkish`.

Turkish typographic rules specify that a little 'white space' should be added before the characters '`:`', '`!`' and '`=`'. In order to insert this white space automatically these characters are made `\active`, so they have to be treated in a special way.

```
\initiate@active@char{:}
\initiate@active@char{!}
```

We specify that the turkish group of shorthands should be used. These characters are 'turned on' once, later their definition may vary.

```
\addto\extrasturkish{%
  \languageshorthands{turkish}%
  \bbl@activate{:}%
  \bbl@activate{!}%
  \bbl@activate{=}%
  \bbl@frenchspacing}
```

For Turkish texts `\frenchspacing` should be in effect. We make sure this is the case and reset it if necessary.

```
\addto\noextrasturkish{\bbl@nonfrenchspacing}
```

`\turkish@sh@!@` `\turkish@sh@=@` `\turkish@sh@:@`

The definitions for the three active characters were made using intermediate macros. These are defined now. The insertion of extra 'white space' should only happen outside math mode, hence the check `\ifmmode` in the macros.

```
\declare@shorthand{turkish}{:}{%
  \ifmmode
    \string:%
  \else\relax
    \ifhmode
      \ifdim\lastskip>\z@
        \unskip\penalty\@M\thinspace
      \fi
    \fi
    \string:%
  \fi}
\declare@shorthand{turkish}{!}{%
  \ifmmode
    \string!%
  \else\relax
    \ifhmode
      \ifdim\lastskip>\z@
        \unskip\penalty\@M\thinspace
      \fi
    \fi
    \string!%
  \fi}
\initiate@active@char{=}
\declare@shorthand{turkish}{=}{%
  \ifmmode
    \string=%
  \else\relax
    \ifhmode
      \ifdim\lastskip>\z@
        \unskip\kern\fontdimen2\font
        \kern-1.4\fontdimen3\font
```

```
      \fi
    \fi
    \string=%
  \fi}
```

The macro `\ldf@finish` takes care of looking for a configuration file, setting the main language to be switched on at `\begin{document}` and resetting the category code of `@` to its original value.

```
\ldf@finish{turkish}
```

⟨/code⟩