The **zhlipsum** Package: Chinese Dummy Text

Xiangdong Zeng

2020/04/10   v1.2.0*

1   Introduction

The zhlipsum package is used for typesetting dummy text (i.e. "*Lorem ipsum*") as lipsum, kantlipsum, blindtext etc., but for Chinese language. Dummy text will be pretty useful, for example, when testing fonts or page styles.

zhlipsum supports UTF-8, GBK and Big5 encodings. Packages expl3, xparse and l3keys2e in the LATEX3 Project are required. To typeset Chinese properly, zhlipsum should be used with CJK package or CTEX bundle.

2   User's guide

encoding

New: 2017-09-16
Updated: 2018-04-01

encoding = ⟨**utf8**|*gbk*|*big5*⟩

Package option for selecting encoding. Default value is utf8. For Unicode engines as XƎLATEX, LuaLATEX and upLATEX, gbk / big5 encodings are invalid and utf8 will be used forcibly.

If you have loaded CTEX bundle, then the encoding will be selected automatically according to CTEX. Note that in CTEX bundle, the correspoding options are UTF8 and GBK, while the options in zhlipsum are all in *lowercase*.

\zhlipsum

Updated: 2020-04-08

\zhlipsum[⟨*paragraph*⟩][⟨*options*⟩]
\zhlipsum*[⟨*paragraph*⟩][⟨*options*⟩]

Produce dummy text. Both arguments ⟨*paragraph*⟩ and ⟨*options*⟩ are optional. Note that spaces are not allowed between the arguments.

By default, the \zhlipsum command will insert \par after and between dummy text paragraphs, while \zhlipsum* will not give any extra processing. To change the default behavior, you can use the before, after and inter options described below.

The first optional argument ⟨*paragraph*⟩ should be a comma list. It can be specified as the following:

```
──────────────── Example 1 ────────────────
% Suppose the dummy text has 50 paragraphs.
\zhlipsum[2-4]          % Can be specified as a-b
\zhlipsum[4,12,3-8]     % A single number is also acceptable
\zhlipsum[-10,40-]      % Produce paragraphs 1-10 and 40-50
\zhlipsum[-]            % Produce all paragraphs, i.e. 1-50
\zhlipsum               % Default value is 1-3
```

```
\zhlipsum[48-52]        % Numbers larger than 50 will not be considered
                        % i.e. only paragraphs 48-50 are produced
```

The second optional argument ⟨*options*⟩ should be a key-value list. Supported options are the listed below.

name | name = ⟨*name*⟩

New: 2018-03-24

Select the name of the dummy text. There are 6 pre-defined dummy texts described in table 1. The default text is simp when encoding=utf8 or gbk, but trad when encoding=big5.

Table 1   Pre-defined dummy texts

Name	Paragraph numbers	Simplified / traditional	Description	Encodings' support		
				utf8	gbk	big5
simp	50	S	Random dummy text	•	•	
trad	50	T	Random dummy text	•	•	•
nanshanjing	43	T	*Shanhaijing: Nanshanjing*	•		
xiangyu	45	T	*Shiji: Xiang Yu Benji* by Sima Qian	•	•	•
zhufu	110	S	*Zhufu* by Lu Xun	•	•	
aspirin	66	S	Wikipedia: *Aspirin*	•	•	

You can use \newzhlipsum command to define new dummy text as well.

before | before = ⟨*content*⟩
after | after = ⟨*content*⟩
inter | inter = ⟨*content*⟩

New: 2018-03-29

Insert contents before, after or between dummy text paragraphs. Note that the \par command inserted when using \zhlipsum will be overridden by the settings here.

\newzhlipsum | \newzhlipsum{⟨*name*⟩}{⟨*paragraphs list*⟩}

New: 2018-03-29

Declare new dummy text. The ⟨*name*⟩ is case sensitive and the ⟨*paragraphs list*⟩ is a comma list. An example is shown below:

Example 2
```
% Fullwidth comma `，' is used in Chinese language.
% Normal comma `,' is used as separator.
\newzhlipsum{jingyesi}{%
  {床前明月光，}, {疑是地上霜。}, {举头望明月，}, {低头思故乡。}}

\zhlipsum*[-][name=jingyesi]  % Print all the four sentences without `\par'
```
---

3   Programming interface

Usually, the commands provided in section 2 are sufficient for users. For programmers professional users, however, the programming interface is also necessary and provided here. LaTeX3 syntax should be opened when using them.

\g_zhlipsum_seq

A sequence of dummy text names.

\zhlipsum_use:nn

Produce some dummy text paragraphs.
#1: Name
#2: Comma list of aragraph numbers.

`\zhlipsum_if_exist:n`*`TF`*

Test whether the name has been used for dummy text。

#1: Name

`\zhlipsum_new:nn`

Declare dummy text.

#1: Name.

#2: Comma list of texts.

4 Compatibility information

The following option exists in the beta version of zhlipsum package, but has become deprecated after version 1.0.0. It is reserved only for compatibility and may be removed in the future.

`script`

Deprecated option. Now it's the same as `name`.

5 Known issues

Dummy text `nanshanjing` and `xiangyu` have some rarely used characters. To display them correctly, you can use the xeCJK package and set SimSun-ExtB or Hanazono Mincho as the fallback font. Refer to the xeCJK's user guide for specific methods (only for UTF-8 encoding and X$\exists$LATEX engine).

GBK and Big5 encodings do not escape the ASCII range in the second byte, so the second byte of some Chinese characters may have the same encoding as special characters in ASCII like {, }, \ etc., which will lead to compilation failure. The `.def` files in zhlipsum are created with special techniques. Please do not modify them.

If there is no special requirement, UTF-8 encoding and Unicode engines as X$\exists$LATEX and LuaLATEX are always recommended.

In special cases, if you must use GBK or Big5 encoding and need to declare new dummy text, the following method can be taken in order to avoid the problem temporarily.

Example 3

```
% File encoding should be Big5.
% \usepackage[encoding=big5]{zhlipsum}

% Using `\newzhlipsum{big5}{許蓋功, 蓋功許, 功許蓋}' directly will
% lead to an error.
% Use <, >, + to replace {, } and \, then set the original {, } and \
% to be `other' category (i.e. catcode=12).
\begingroup
  \catcode`\<=1
  \catcode`\>=2
  \catcode`\+=0
  \catcode`\{=12
  \catcode`\}=12
  \catcode`\\=12
  +newzhlipsum<big5><許蓋功, 蓋功許, 功許蓋>
+endgroup
\zhlipsum[name=big5]
```