高亮 TeX 和 LATeX3 代码——使用 texhigh 宏包

雾月*

2025年2月23日

texhigh 宏包¹是专用来高亮 T_{EX} 文件的宏包。基于由 Rust 编写的命令行工具 texhigh²,处理 1.24MB 左右 (37,700 余行)的 expl3-code.tex 只需 0.15s 左右 (操作系统为 Windows, CPU 为 i7-12700H),处理速度约为 minted 宏包使用的 pygmentize(约 2.75s)的 18 倍,texhigh 的增强模式也比它快 10 到 16 倍(约 0.165s)。对于普通大小的 T_{EX} 代码,处理它们所需的时间相比于 T_{EX} 文件本身编译所需的时间,已经可以忽略不记。

texhigh 主要是在 LATEX 中为 texhigh 命令行工具提供交互接口。这要求在编译 TEX 文件时启用 --shell-escape。

texhigh 除了可用于高亮 T_EX 文件,还支持计算文字的布局。基于此特性,texhigh 提供了输出颜文字的功能: $\epsilon(T_T_T)$ 3 ,只需使用 <code>\kaomoji</code> $\{\langle 颜文字 \rangle\}$ 。默认使用系统字体,也可自行设置 ヾ($\geq \nabla \leq *$)o。

 $\Box(\mathfrak{I}^{*'})\mathfrak{I}^{*}$ $\Box(\mathfrak{G}^{\bullet})\mathfrak{I}^{\circ}$ $\Box(\mathfrak{G}^{\bullet})\mathfrak{I}^{\circ}$ $\Box(\mathfrak{G}^{\bullet})\mathfrak{I}^{\circ}$ $\Box(\mathfrak{G}^{\bullet})\mathfrak{I}^{\circ}$ $\Box(\mathfrak{G}^{\bullet})\mathfrak{I}^{\circ}$ $\Box(\mathfrak{G}^{\bullet})\mathfrak{I}^{\circ}$

使用颜文字时可能会遇到字体问题,这时在字符间插入零宽词连接符 U+2060 或可解决。

使用 \kaomoji* 还支持把单行文字输出为图片:

% 这里的 fontsize 影响图片的大小,从而影响清晰度

代码 1



也可以自己封装一下这个命令:

^{*}longaster@163.com

¹https://github.com/Sophanatprime/texhigh

²https://github.com/Sophanatprime/texhigh-rs

```
\makeatletter
% fonts 键用于设置额外的字体。texhigh 会查找系统字体,一般无需另行设置
\NewDocumentCommand\inmoji{ D<>\\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\footnote{\foot
```

texhigh 提供 \texhighverb、\texhighfile、\texhighinput 这几个命令以及一个 texhigh 环境用于高亮 TrX 代码。

\texhighverb 用法和\verb 类似,但没有带星号的版本,它不能作为其它命令的参数; ← \texhightext 用于高亮文字,一般用于高亮已经处理过的文本,和 \texhighverb 相比,它可以作为命令的参数。\texhighfile 用于高亮一个文件,\texhighinput 则用于导入一个已经被处理过的文件。

texhigh 还有很强的可配置性。

为了实现处理 T_EX 源码与输出结果的分离, texhigh 使用"类型"和"类别"来区分不同的记号。字符和控制序列是不同的"类型", 控制序列之间可以有不同的"类别", 例如是原语、LAT_EX3函数等。类型不可改变, 而"类别"可以自由修改。

每个类型都有一些命令用于更改它们的"类别"的显示效果,如,对于一个控制序列,可以使用 \THSetClassCS 改变显示效果。可以为它们设置前景色、背景色,甚至渐变色和底纹等等。实际上普通文字可以显示成什么效果,它们就可以做到同样的效果。具体修改方式可以参考文末 basic 样式的源码。

texhigh 利用 tikz 实现了渐变和底纹效果,同时也可直接集成到 tcolorbox 宏包中。只需要在加载 texhigh 之前加载这几个宏包。

```
\usepackage{tikz}
\usepackage{tcolorbox}
\usepackage{texhigh}
\tcbset{listing engine=texhigh} % 使用这个即可切换至 texhigh
% 若使用 xeCJK, 即在 XeLaTeX 中使用 ctex, 最好设置
\SetKeys[texhigh/high]{
font=\ttfamily\xeCJKsetup{CJKecglue={\hskip} Opt plus 0.08\baselineskip}}}
}
% 这样可避免在显示代码时中英文之间出现不必要的空格。
```

识别行内数学公式:

```
代码 4
<mark>\texhighverb</mark>!公式 $ \int a^b x^2 dx = \frac{1}{3} x^3 | a^b $!。
渐变:
                                                                代码 5
\ttexhighverb\t[style=tikz.gradient, use-ctab=latex3, \leftarrow
   config-file=config.cfg] | \sys_get_shell:nnNTF |
\sys_get_shell:nnNTF
 底纹:
                                                                代码 6
\makeatletter
% #1: tikz options, #2: text
\\def\\myshadetext#1#2{\\texhigh@shadetext\{#1}{\\bfseries #2}}
\makeatother
{\LARGE
% 在加载 texhigh 之前加载 tikz 宏包!
% 使用 grass.png 作为文字底纹, 依赖 tikz 的 fill.image 库, 会自动加载这个库。
\texhighverb [use-ctab=latex3, this-cs=\myshadetext{fill stretch ←

    image=grass.png
}]
|\sys get shell:nnNTF|
\sys\_get\_shell:nnNTF
  中文命令识别 (TFX 原语带有下划线):
                                                                代码 7
\begin{texhigh} [output=\jobname.texhigh, use-ctab=cjk]
 \def\好好好{中文 Good}
 \好好好\relax
\end{texhigh}
 \def\好好好{中文 Good}
 \好好好\relax
  类别码混合使用:
                                                                代码 8
% 自动检测 \makeatletter 和 \ExplSyntaxOn 块,
% \makeatletter 和 \ExplSyntaxOn 必须在行首,前面可以有空格
\begin{texhigh}[lexer-catcode={atletter, explon}]
\def\foo@#1{[#1]} \foo@{FOO} \@kernel
\makeatletter
```

```
\def\foo@:#1{[#1]} \foo@:#1{FOO} \@kernel \scan stop:
\ExplSyntaxOn
\cs set:Npn \foo@: #1 { [#1] }
\foo@: {FOO} \@kernel \scan_stop:
\ExplSyntaxOff
\@kernel \scan stop:
\makeatother
\@kernel \scan stop:
\end{texhigh}
\def\foo@#1{[#1]} \foo@{FOO} \@kernel
\makeatletter
\def\foo@:#1{[#1]} \foo@:#1{FOO} \@kernel \scan stop:
\ExplSyntaxOn
\cs set:Npn \foo@: #1 { [#1] }
\foo@: {FOO} \@kernel \scan stop:
\ExplSyntaxOff
\@kernel \scan stop:
\makeatother
\@kernel \scan stop:
```

也可手动调整类别码:

```
代码 9
\begin{texhigh}[
  lexer-catcode*={3}{9}{@=11, ?=11}, % [3, 9) 行, @ 和 ? 的类别码为 11
  % lexer-catcode*={3}{9}{ @?=11 }, % <- 可以合并
 lexer-catcode*={5}{7}{explon}, % [5, 7) 行, 启用 expl3 的类别码
\def\foo@#1{[#1]} \foo@{FOO} \@kernel
\makeatletter
\def\foo@:#1{[#1]} \foo@:#1{F00} \@kerne? \scan_stop:
\ExplSyntaxOn
\cs set:Npn \foo@: #1 { [#1] }
\foo@: {FOO} \@kernel \scan stop:
\ExplSyntaxOff
\@kernel \scan stop:
\makeatother
\@kernel \scan stop:
\end{texhigh}
\def\foo@#1{[#1]} \foo@{F00} \@kernel
\makeatletter
\def\foo@:#1{[#1]} \foo@:#1{FOO} \@kerne? \scan stop:
\ExplSyntaxOn
```

```
\cs_set:Npn \foo@: #1 { [#1] }
\foo@: {F00} \@kernel \scan_stop:
\ExplSyntaxOff
\@kernel \scan_stop:
\makeatother
\@kernel \scan_stop:
```

除了用 use-ctab 设置主类别码表以外,还可以添加额外的类别码表,优先选择靠后的类别码表:

还能进一步细化到列数:

```
      % 1 相当于 [1, 0],
      (1, 9] 表示第 1 行第 9 个字符

      % 这里就是从第一行的开始,直到第一行第 9 个字符 (不含),空格的类别码为 0
      设置类别码时,特殊字符必须转义,其它字符可转义也可不转义

      \texhighverb[lexer-catcode*={1}{[1,9]}{\text{1,9]}}{\text{abd def #1{\space }}}

      \texhighverb{abd def #1"\space"}

      abd def #1{\space }

      abd def #1"\space"
```

lexer-catcode* 的前两个参数还支持更加复杂的模式,如纯文本正则表达式和 T_{EX} 正则表达式。

例如检测 \verb 命令:

```
      \texhighverb
      [lexer-catcode*={\c{verb}\*?\|}{\|}{str}]
      (在码 12)

      {a \verb|\macro in \verb| command \verb*|\macro in \verb| \macro
      a \verb|\macro in \verb| \macro
```

纯文本正则表达式就是针对纯文本的正则表达式,日常见到的正则表达式都是这一类, texhigh 支持的纯文本正则表达式的完整语法见 https://docs.rs/regex/latest/regex/#syntax。

T_EX 正则表达式是针对 T_EX token 的正则表达式, I^AT_EX3 的 I3regex 就是这类, texhigh 支持的 T_EX 正则表达式语法和 I3regex 基本一致,但暂不支持 \b \B \G \u 这几个转义序列,以及 \c 转义序列的否定形式(即暂不支持 [^\c{end}] 这类用法)。

```
代码 13
% \I 放在开头表示这是纯文本正则表达式,匹配源码
\begin{texhigh}[gobble=2, % 每行删除前两个字符
  lexer-catcode*=
   {\I\\catcode`\\!=11[\s\\]}
   {\I\\endgroup\s}
                                             \\def\!mark#1{MARK1} \!mark
   {!=11},
                                             \begingroup
                                               \catcode`\!=11
]
                                               \def\!mark#1{MARK2} \!mark
  \def\!mark#1{MARK1} \!mark
  \begingroup
                                               \endgroupp \!mark
   \catcode`\!=11
                                             \endgroup
   \def\!mark#1{MARK2} \!mark
                                             \!mark
   \endgroupp \!mark
  \endgroup
  \!mark
\end{texhigh}
                                                                 代码 14
%\T 放在开头表示这是 TeX 正则表达式, 匹配 token
% \T 大多数时候可以省略,但当模式以数字或 [ 开头
时,\T 不可省略,否则被当作行号
\begin{texhigh}[gobble=auto, % 检测空格并删除
  lexer-catcode*=
   {\T\c{catcode}\`\c{!}\=11} % 注意字符转义
                                             \def\!mark#1{MARK1} \!mark
   {\c{endgroup}}
                                             \begingroup
                                               \catcode`\!=11
   {!=11},
]
                                               \def\!mark#1{MARK2} \!mark
  \def\!mark#1{MARK1} \!mark
                                               \endgroupp \!mark
  \begingroup
                                             \endgroup
   \catcode`\!=11
                                             \!mark
   \def\!mark#1{MARK2} \!mark
   \endgroupp \!mark
  \endgroup
  \!mark
\end{texhigh}
```

lexer 也可混合使用正则表达式和行数:

```
\begin{texhigh}[gobble=auto,
    lexer-catcode*={(^|\cJ.|\n)\c{ExplSyntaxOn}}{4}{\:\_=11},

    \_ExplSyntaxOn
    \_cs_set:Npn \expl_off: { \ExplSyntaxOff } \expl_off:
    \_ExplSyntaxOff
    \_expl_off:
    \_explSyntaxOn
    \_cs_set:Npn \expl_off: { \ExplSyntaxOff } \expl_off:
    \_ExplSyntaxOff
    \_explSyntaxOff
```

可用使用 lines 设置源文件需要保留的行数, gobble=auto 检测空格时只会检测保留下来的代码行:

```
代码 16
\begin{texhigh}[gobble=auto, % 检测空格并删除
  lines={2,8}, % 只保留 [2,8) 行
  lexer-catcode*=
    {\T\c{catcode}\`\c{!}\=11} % 注意字符转义
    {\c{endgroup}}
                                                \def\!mark#1{MARK1} \!mark
    {!=11},
]
                                                \begingroup
                                                  \catcode \!=11
      \kill this line
  \def\!mark#1{MARK1} \!mark
                                                  \def\!mark#1{MARK2} \!mark
  \begingroup
                                                  \endgroupp \!mark
                                                \endgroup
    \catcode`\!=11
    \def\!mark#1{MARK2} \!mark
    \endgroupp \!mark
  \endgroup
  \!mark
\end{texhigh}
```

```
% \THSetCharReplacement{\} {\textvisiblespace} % texhigh 已设置
\THSetCharReplacement{\$}{\S} % 把 $ 替换为 \S
\texhighverb[char-replacements={\ \$}, % 设置哪些字符要替换
char-category*={symbol}{[\ \$]}{\mbox{\color{red}#1}}, % 修改颜色
] {\def\a #1{$#1$} \a {\ b}}
\def\a_u#1{$#1$}_u\a_u{\ub}
```

控制序列的名称里的字符也会被替换:

char-category 也可以用来替换字符,但不会替换控制序列名称里的字符:

```
代码 19
\ExplSyntaxOn
 \cs new protected: Npn \chartouni #1 { \fbox{ \int to Hex:n { `#1 }} }
\ExplSyntaxOff
\texhighverb[
 % 这里使用正则表达式查找字符的类别,下面的正则表达式匹配是 Emoji 但不
是 ASCII 的字符
 char-category*={emoji}{[\p{Emoji}--\p{ASCII}]}{\chartouni{#1}\\ }
Emoji: | 1F400 |
           1F403
                 1F405
                       1F407
                             1F409
                                   1F40D
                                         1F40E
                                               1F410
                                                     1F412
                                                           1F413
1F415
     1F416
```

texhigh 默认的定义以及部分命令的用法可参考 texhigh.prelude.ths 文件。

```
%%%---- File: texhigh.prelude.ths ----%%%
\ProcessKeyOptions[texhigh/prelude]
\THSetCharReplacement {\ }{\textvisiblespace}
% \THSetCharReplacement*{\ }{\ifincsname\space\else\textvisiblespace\fi}
\THSetCharReplacement {\^^1} {\mbox{\THcolor {gray8}$\rightarrow$}}
\texhighsetclassfallback{ch}{group.0}{group}
\texhighsetclassfallback{ch}{group.1}{group}
\texhighsetclassfallback{ch}{group.2}{group}
\texhighsetclassfallback{ch}{group.3}{group}
\texhighsetclassfallback{ch}{group.4}{group.1, group}
\texhighsetclassfallback {ch} {group.5} {group.2, group}
\texhighsetclassfallback {ch} {group.6} {group.3, group}
\texhighsetclassfallback{ch}{group.7}{group.1, group}
\texhighsetclassfallback{ch}{group.8}{group.2, group}
\texhighsetclassfallback{ch}{group.9}{group.3, group}
\texhighsetclassfallback{ch}{group.-1}{group.miss}
\texhighsetclassfallback{ch}{group.-2}{group.miss}
\texhighsetclassfallback{ch}{group.-3}{group.miss}
\texhighsetclassfallback{ch}{group.-4}{group.miss}
\texhighsetclassfallback{ch}{group.-5}{group.miss}
```

```
\texhighsetclassfallback {cs} {primitive.luatex} {primitive, tex, latex}
\texhighsetclassfallback {cs} {primitive.xetex} {primitive, tex, latex}
\texhighsetclassfallback {cs}{primitive.uptex}{primitive, tex, latex}
\texhighsetclassfallback {cs} {primitive.pdftex} {primitive, tex, latex}
\texhighsetclassfallback {cs}{primitive.etex}{primitive, tex, latex}
\texhighsetclassfallback{cs}{primitive.knuthtex}{primitive, tex, latex}
\texhighsetclassfallback{cs}{primitive.widely}{primitive, tex, latex}
\texhighsetclassfallback{cs}{primitive.sometex}{primitive, tex, latex}
\texhighsetclassfallback{cs}{primitive.luametatex}{context}
\texhighsetclassfallback \{cs\{plaintex\}\{tex\}
\texhighsetclassfallback {cs} {latex.document} {latex}
\texhighsetclassfallback \{cs\{latex.programming\{latex\}
\texhighsetclassfallback \{cs\}\latex.internal\}\laternal, latex\}
\texhighsetclassfallback {cs} {latex.public} {latex}
\texhighsetclassfallback{cs}{latex3.primitive}{latex3.function.kernel, latex3.function, \( --
    latex3, primitive}
\texhighsetclassfallback{cs}{latex3.variable}{latex3, latex}
\texhighsetclassfallback{cs}{latex3.function}{latex3, latex}
\time \text{texhighsetclassfallback} \{cs\} \{latex3.variable.kernel} \{latex3.kernel, latex3.variable, \leftarrow
    latex3}
\texhighsetclassfallback {cs} {latex3.function.kernel} {latex3.kernel, latex3.function, \( ---
    latex3}
\texhighsetclassfallback{cs}{latex3.variable.internal}{internal, latex3.variable, latex3}
\texhighsetclassfallback{cs}{latex3.variable.public}{latex3.variable, latex3}
\texhighsetclassfallback {cs} {latex3.function.internal} {internal, latex3.function, latex3}
\texhighsetclassfallback \{cs\{latex3.function.public\}\{latex3.function, latex3\}
\texhighsetclassfallback \{cs\\ math\\ tex, latex\}
\texhighsetclassfallback \{cs\{symbol\}\{tex, latex\}
\texhighsetclassfallback{rs}{math.inline}{math}
\texhighsetclassfallback{re}{math.inline}{math}
\@ifpackageloaded{xcolor}{}{\RequirePackage{xcolor}}
\RequirePackage{ninecolors}
\THSaveStyle {plain} {\THSetPlainStyle {*}}
\THSaveStyle {plain0} { \THSetPlainStyle {bp,cs,ch,st,es,ee,pn,color} } %
  \THSetClassRS\{comment\}\\degingroup\THColorStatus\{1\\THcolor\gray\}\{0.5\}\%
  \THSetClassRE { comment } { \ \ endgroup \} }
\THSaveStyle {basic} {\%
  \THSetClassBP {?}
    {\discretionary
      {\hbox{{\THcolor{gray8}\_$\hookleftarrow$}}}
      {\hbox{{\THcolor{gray8}$\hookrightarrow$\_}}}
      {}}%
  \THSetClassCS\{texhigh\}{\\fboxsep\z0\% for texhigh package itself
```

```
\cline{Theolor}{\vertex}{\clinespread{1}\bfseries\strut} Theolor{black}#1#2}}% \leftarrow
\hookrightarrow #1=escape char, #2=cs name
  \THSetClassCS{\latex3.primitive}{\mbox{\THcolor{red4}\bfseries#1#2}}%
  \THSetClassCS{latex3.kernel}{\mbox{\THcolor{red4}\bfseries#1#2}}%
  \THSetClassCS{latex3.variable}{\mbox{\THcolor{azure6}#1#2}}%
  \THSetClassCS{latex3.function}{\mbox{\THcolor{green5}#1#2}}%
  <mark>\THSetClassCS</mark>{internal}{\mbox{<mark>\THcolor</mark>{brown3}#1#2}}%
  <mark>\THSetClassCS</mark>{latex.document}{\mbox{<mark>\THcolor</mark>{magenta4}\bfseries#1#2}}%
  \THSetClassCS{latex.programming}{\mbox{\THcolor{yellow7}#1#2}}%
  <mark>\THSetClassCS</mark>{latex}{\mbox{<mark>\THcolor</mark>{yellow8}#1#2}}%
  \THSetClassCS{primitive}{\texhigh@underline{\THcolor{purple5}\bfseries#1#2}}%
  <mark>\THSetClassCS</mark>{?}{\mbox{<mark>\THcolor</mark>{magenta5}#1#2}}%
  \THSetClassCH{?}{#1}% char
  \THSetClassPN {?}{#1}% punctuation
  <mark>\THSetClassCH</mark>{group}{\mbox{<mark>\THcolor</mark>{purple3}#1}}%
  \THSetClassCH{group.1}{\mbox{\THcolor[HTML]{179FFF}#1}}
  \THSetClassCH {group.2} {\mbox{\THcolor [HTML] {DA6ED6}#1}}
  \THSetClassCH{group.3}{\mbox{\THcolor[HTML]{F8BA16}#1}}
  \THSetClassCH {group.miss} {\mbox{\THcolor {red} #1}} %
  \THSetClassCH{digit}{\mbox{\THcolor{azure8}#1}}%
  \THSetClassRS{comment}{\begingroup\THcolor[gray]{0.5}\THSetPlainStyle{cs,color}}%
  \THSetClassRE{comment}{\endgroup}%
  \THSetClassRS{parameter}{\begingroup\THcolor{magenta2}}%
  \THSetClassRE{parameter}{\endgroup}%
  \THSetClassRE \{ math \} \{ \ \ endgroup \} \%
\THUseSavedStyle{basic}
\long\def\\texhigh@underline\#1\{\leavevmode\\setbox\\z@\\hbox\{\#1\}\%
  \hb@xt@\wd\z@{\rlap{\unhcopy\z@}%
    \kern.05em\vrule height-.25ex depth.4ex width\dimexpr\wd\z@-.1em\relax \kern.05em}}
\if@texhighload@color
  \relax
\@ifpackageloaded{tikz}{<mark>\@texhighload@tikztrue</mark>}{}
\newbox\texhigh@picturebox
\if@texhighload@tikz
  \RequirePackage{tikz}
  \usetikzlibrary{shadings}
  \usetikzlibrary{fill.image}
  \protected\def\\texhigh@shadetext\#1\#2\{\%
    \setbox\texhigh@picturebox=\hbox{{\texhigh@pdfliteral}{7 Tr }#2}}%
    \tikz[baseline=0,line width=0pt]\path\pgfextra{\rlap{\copy\texhigh@picturebox}}
      [#1] (0,-\dp\texhigh@picturebox) rectangle (\wd\texhigh@picturebox,\ht \efficies
```

```
\tikzset{texhigh/.is family,
    texhigh/gradient primitive/.style={left color=blue,right color=cyan},
    texhigh/gradient ?/.style={left color=red,right color=blue},
    texhigh/gradient-style/.style={texhigh/gradient #1}}
  \THSaveStyle {tikz.gradient} {\%
    \THSetClassCS{latex}{\texhigh@underline{\THcolor{purple}\bfseries#1#2}}
    \THSetClassCS {primitive}
      {\texhigh@shadetext{texhigh/gradient-style=primitive}{\bfseries #1#2}}%
    \THSetClassCS \{?} {\\texhigh@shadetext \{\texhigh/gradient-style=?} \{\#1\#2\}\%
\fi
\@ifpackageloaded{tcolorbox}{
  \tcbuselibrary{listings@core}
  \def\tcb@texhigh@file#1#2{%
    {\edef\tcb@temp{\texhighfile [{\unexpanded\expandafter{#1}}]}\tcb@temp{#2}}}
  \def\tcb@texhigh@uselistinglisting{\tcb@texhigh@file\kvtcb@texhighoptions \

    \kvtcb@listingfile}

  \def\tcb@texhigh@usetemplisting{\tcb@texhigh@file\kvtcb@texhighoptions\kvtcb@tempfile}
  \def\tcb@texhigh@doc@usetemplisting{\tcb@texhigh@file\kvtcb@doctexhighoptions \
→ \kvtcb@tempfile}
  \tcbset{
    texhigh options/.code=\edef\kvtcb@texhighoptions{\unexpanded{#1}},
    texhigh options=,
    texhigh options pre/.code=\edef\kvtcb@texhighoptions{\unexpanded{#1},\unexpanded \
→ \expandafter{\kvtcb@texhighoptions}},
    texhigh options app/.code=\edef\kvtcb@texhighoptions{\unexpanded\expandafter{ \( \lefta \)

   \kvtcb@texhighoptions,#1}},
    texhigh gobble/.style={texhigh options app={gobble=#1}},
    texhigh gobble/.default=auto,
    texhigh config file/.style={texhigh options app={config-file={#1}}},
    texhigh ctab file/.style={texhigh options app={ctab-file={#1}}},
    texhigh use ctab/.style={texhigh options app={use-ctab={#1}}},
    texhigh style/.style={texhigh options app={style={#1}}},
    texhigh detect catcodes/.style={texhigh options app={lexer-catcode={#1}}},
    listing engine/texhigh/.code={\let\tcbuselistinglisting\tcb@texhigh@uselistinglisting
      \let\tcbusetemplisting\tcb@texhigh@usetemplisting
      \let\tcb@doc@usetemplisting\tcb@texhigh@doc@usetemplisting},
}{}
```