

The lstlinebgrd Package

Martin Scharrer
martin@scharrer-online.de

CTAN: <http://www.ctan.org/pkg/lstlinebgrd>

VC: https://bitbucket.org/martin_scharrer/lstlinebgrd

Version v0.1 – 2012/05/03

Abstract

This add-on package to `listings` provides the option to add background colors to lines of listings. Every line can be given a different color by using a conditional which depends on the line number.

This package was created as response to the question “[Creating a zebra effect using listings](#)” on [TeX Stack Exchange](#).

1 Usage

The following settings are added by `lstlinebgrd` for `listings` and can be used in `\lstset` or any `listings` macro which accepts options.

`linebackgroundcolor=<color commands>`

This options allows the user to provide some \TeX code which sets the background color. The code must use `\color{<color>}` or a different color macro. The use of conditionals is permitted and can be used to color lines depending on the line number (`\value{lstnumber}`) etc., as shown in the example section. The initial setting of this option is `{}`, i.e. no coloring is done.

`linebackgroundsep=<length>`

Sets the distance of the begin of the background color bar and the line. A horizontal skip of this amount to the left from the start of the line is done before drawing the background line. This means that positive amounts will move the background color bar to the left and negative amounts to the right. This does not effect the total width of the color bar.

`linebackgroundwidth=<length>`

Sets the width of the background color bar. The initial default value is `\linewidth`. Note that the background width does not take the actual width of the text into account, because `listing` does not provide this information.

`linebackgroundheight=<length>`

Sets the height of the background color bar, i.e. the amount above the baseline. The initial value is `\ht\strutbox`, i.e. the full height of the line.

`linebackgrounddepth=<length>`

Sets the depth of the background color bar, i.e. the amount below the baseline. The initial value is `\dp\strutbox`, i.e. the full depth of the line.

`linebackgroundcmd=<macro>`

This advanced option sets the macro used to draw the background color bar. It is called at the correct starting position and receives the width, height and depth as three arguments in this order. The initial value is `\color@block` which is an internal macro of the `xcolor` package. This option can be used by advanced users to fine tune the background coloring.

2 Examples

Example 1: Basic example: zebra effect.

Code

```
1 \begin{lstlisting}[language=C,basicstyle=\ttfamily,/  
   linebackgroundcolor={\ifodd\value{lstnumber}\color{green}\fi/  
   }]  
2 /**  
3  * Prints Hello World.  
4  **/  
5 #include <stdio.h>  
6  
7 int main(void) {  
8 printf("Hello World!");  
9 return 0;  
10 }  
11 \end{lstlisting}
```

Result


```
/**  
* Prints Hello World.  
**/  
#include <stdio.h>  
  
int main(void) {  
printf("Hello World!");  
return 0;  
}
```

Example 2: Two-colored zebra effect with extra adjustments.

Code

```
1 \begin{lstlisting}[language=C,basicstyle=\ttfamily\Large,/  
  linebackgroundcolor={\ifodd\value{lstnumber}\color{green}\/  
  else\color{yellow}\fi},numbers=left,linebackgroundsep=1em,/  
  linebackgroundwidth=18em]  
2 /**  
3  * Prints Hello World.  
4  **/  
5 #include <stdio.h>  
6  
7 int main(void) {  
8 printf("Hello World!");  
9 return 0;  
10 }  
11 \end{lstlisting}
```

Result



```
/**  
* Prints Hello World.  
**/  
#include <stdio.h>  
  
int main(void) {  
printf("Hello World!");  
return 0;  
}
```

Example 3: Gradient effect with extra adjustments.

Code

```
1 \begin{lstlisting}[language=C,basicstyle=\ttfamily\tiny,/  
  linebackgroundcolor={\color{blue!\the\numexpr 5*\value{/  
  lstnumber}\relax}},linebackgroundheight=1.7ex,/  
  linebackgrounddepth=.4ex]  
2 /**  
3 * Prints Hello World.  
4 **/  
5 #include <stdio.h>  
6  
7 int main(void) {  
8 printf("Hello World!");  
9 return 0;  
10 }  
11 /**  
12 * Prints Hello World.  
13 **/  
14 #include <stdio.h>  
15  
16 int main(void) {  
17 printf("Hello World!");  
18 return 0;  
19 }  
20 \end{lstlisting}
```

Result

```
/**  
 * Prints Hello World.  
 **/  
#include <stdio.h>  
  
int main(void) {  
printf("Hello World!");  
return 0;  
}  
/**  
 * Prints Hello World.  
 **/  
#include <stdio.h>  
  
int main(void) {  
printf("Hello World!");  
return 0;  
}
```

3 Implementation

```
19 %<!COPYRIGHT>
20 \ProvidesPackage{lstlinebgrd}[%
21 %<!DATE>
22 %<!VERSION>
23 %<*DRIVER>
24     2099/01/01 develop
25 %</DRIVER>
26     Provides 'listings' options to color the /
        background of lines]

27 %
28 \RequirePackage{listings}
29 \RequirePackage{xcolor}
30
31 % Patch line number key to call line background macro
32 \lst@Key{numbers}{none}{%
33     \def\lst@PlaceNumber{\lst@linebgrd}%
34     \lstKV@SwitchCases{#1}%
35     {none&\\%
36         left&\def\lst@PlaceNumber{\llap{\normalfont
37             \lst@numberstyle{\thelstnumber}\kern\%
38                 \lst@numbersep}\lst@linebgrd}\\%
39             right&\def\lst@PlaceNumber{\rlap{\normalfont
40                 \kern\linewidth \kern\lst@numbersep
41                 \lst@numberstyle{\thelstnumber}}\%
42                 \lst@linebgrd}%
43             }{\PackageError{Listings}{Numbers #1 unknown}\%
44                 @ehc}}
45
46 % New keys
47 \lst@Key{linebackgroundcolor}{}{%
48     \def\lst@linebgrdcolor{#1}%
49 }
50 \lst@Key{linebackgroundsep}{0pt}{%
51     \def\lst@linebgrdsep{#1}%
52 }
53 \lst@Key{linebackgroundwidth}{\linewidth}{%
54     \def\lst@linebgrdwidth{#1}%
55 }
56 \lst@Key{linebackgroundheight}{\ht\strutbox}{%
57     \def\lst@linebgrdheight{#1}%
58 }
59 \lst@Key{linebackgrounddepth}{\dp\strutbox}{%
60     \def\lst@linebgrddepth{#1}%
61 }
```

```

62
63
64 % Line Background macro
65 \newcommand{\lst@linebgrd}{%
66   \ifx\lst@linebgrdcolor\empty\else
67   \rlap{%
68     \lst@basicstyle
69     \color{-.}% By default use the opposite ('-')/
70       of the current color ('.') as background
71     \lst@linebgrdcolor{%
72     \kern-\dimexpr\lst@linebgrdsep\relax%
73     \lst@linebgrdcmd{\lst@linebgrdwidth}{\/
74       lst@linebgrdheight}{\lst@linebgrddepth}%
75     }%
76   }%
77   \fi
78 }

```