

How to produce a colored GUI in a console application?

Asked 15 years, 10 months ago Modified 1 year, 6 months ago

Viewed 5k times



4



For the following questions, answers may be for C/C++, C#, or Python. I would like the answers to be cross platform if possible but I realize I will probably need

`conio` or `ncurses`



1. How do I output colored text?

2. How would I do a GUI like `top` or `nethack` where certain things are "drawn" to certain spaces in the terminal?

If possible a small oneliner code example would be great.

`c#`

`c++`

`python`

`c`

`console-application`

Share

Improve this question

Follow

edited Aug 13, 2011 at 2:04



Stéphane Gimenez

807 ● 9 ● 21

asked Feb 26, 2009 at 0:47



jimi hendrix

2,259 ● 5 ● 19 ● 18

3 If only it was somehow possible to download the source code to top or nethack. – [bk1e](#) Feb 26, 2009 at 4:42

9 years late, but how about `htop` ? It's a very neat example: github.com/hishamhm/htop – [Madivad](#) Jun 15, 2018 at 17:13

6 Answers

Sorted by:

Highest score (default)



4

Yes, these are VT100 escape codes. The simplest thing is to use some flavor of Curses. Once, you choose a curses flavor it is pretty simple to do both 1 and 2.



Here's a HowTo on ncurses.



<http://web.cs.mun.ca/~rod/ncurses/ncurses.html>



Share Improve this answer

answered Feb 26, 2009 at 0:54

Follow



[BobbyShaftoe](#)

28.5k ● 7 ● 54 ● 74

3 +1: docs.python.org/library/curses.html, amk.ca/python/howto/curses – [S.Lott](#) Feb 26, 2009 at 1:02

The original web link is now broken. – [wonea](#) Aug 17, 2011 at 9:38



1

From this point of view, the console is in many ways just an emulation of a classic terminal device. Curses was created originally to support a way of doing common operations on different terminal types, where the actual



terminal in use could be selected by the user as part of the login sequence. That heritage survives today in ncurses.



The ncurses library provides functions to call to directly position the cursor and emit text, and it is known to work for the Windows Console (where CMD.EXE runs), as well as on various *nix platform equivalents such as XTerms and the like. It probably even works with a true Dec VT100 over a serial line if you had such a thing...

The escape sequences understood by the VT100 and later models became the basis for the ANSI standard terminal. But you really don't want to have to know about that. Use ncurses and you won't have to.

Leaning on conio won't get you cross platform, as that is a DOS/Windows specific API.

Edit: Apparently the [ncurses](#) library itself is not easily built on mingw, at least as observed from a quick attempt to Google it up. However, all is not lost, as ncurses is only one of the descendants of the original [curses](#) library.

Another is [PDCurses](#) which *is* known to compile and run for Windows Consoles, as well as for X11 and a variety of *nix platforms.

(I was just reminded from chasing references at Wikipedia that curses came out of writing the game rogue, which is the ancestor of nethack. Some of its code was "borrowed" from the cursor management module of

the vi editor, as well. So spelunking in the [nethack source kit](#) for ideas may not be a crazy idea at all...)

Share Improve this answer

edited Feb 26, 2009 at 8:59

Follow

answered Feb 26, 2009 at 0:56



RBerteig

43.3k ● 7 ● 90 ● 130

ok thanks...i am using mingw to do C/C++ work, i tried to include ncurses.h, but it said it couldnt find it, do i need to get it from somewhere? – [jimi hendrix](#) Feb 26, 2009 at 1:34

Google implies there may be issues with ncurses and mingw. I'll edit my answer with some detail... – [RBerteig](#) Feb 26, 2009 at 8:44



1

Most terminal windows understand the ANSI escape sequences, which allow coloring, cursor movement etc. You can find a list of them [here](#).



Use of these sequences can seem a bit "old school", but you can use them in cases where curses isn't really applicable. For example, I use the folowing function in my bash scripts to display error messages in red:



```
color_red()  
{  
    echo -e "\033[01;31m$1\033[00m"  
}
```

You can then say things like:

```
color_red "something has gone horribly wrong!"  
exit 1
```

Share Improve this answer

edited Feb 26, 2009 at 9:29

Follow

answered Feb 26, 2009 at 0:53



anon



0

Not cross platform but for Windows / C# colour, see

[Color your Console text \(C#\)](#).



[C++](#)



Share Improve this answer

answered Feb 26, 2009 at 0:55

Follow



[rbrayb](#)

46.7k ● 34 ● 118 ● 179



0

In C#, you can set the text color and the background color via the Console.ForegroundColor and Console.BackgroundColor properties, respectively. For a list of valid colors, see this [MSDN doc](#).



Share Improve this answer

answered Feb 26, 2009 at 0:57

Follow



[Patrick](#)

92.4k ● 11 ● 52 ● 61



You can use a simple library like [termcol](#).

0

Share Improve this answer

answered Jun 14, 2023 at 18:34

Follow



Michael

38 ● 9



While this link may answer the question, it is better to include the essential parts of the answer here and provide the link for reference. Link-only answers can become invalid if the linked page changes. - [From Review](#) – [Chenmunka](#) Jun 17, 2023 at 9:40