



[Главная](#) >> [Инструкции](#) >> Цвета терминала Linux

Цвета терминала Linux

Опубликовано: 1 января, 2018 от [romwhite](#) , 11 комментариев, время чтения: 6 минут

Обнаружили ошибку в тексте? Сообщите мне об этом. Выделите текст с ошибкой и нажмите Ctrl+Enter.

Одно из главных преимуществ Linux заключается в том, что при желании в нём можно настроить всё, что угодно. В этом и выражается преимущество проективных систем, над процедурными. Фактически, это конструктор, состоящий из отдельных блоков, каждый из которых можно настроить по своему желанию.

Особенно интересно, если не просто копируешь чужой "рецепт", а пытаешься разобраться, что и как устроено. И даже решая несерьезную задачу, узнаешь что-то новое.

Раскрашиваем цвета вывода ls

Как-то захотелось мне настроить цвета терминала linux в выводе команды ls. Ну не нравился блекло-фиолетовый цвет для каталогов в Linux Mint. Начал разбираться.

Из документации к ls, выяснил, что установка цветов для вывода осуществляется помощи утилиты `dircolors`, которая либо устанавливает параметры цвета из своей собственной базы, либо берет их из файла `~/.dircolors`. Такого файла в моем доме каталоге не было. Также упоминалось, что инициализация использования цвето

Конфиденциальность -
Условия использования

Privacy

происходит через ~/.bashrc, файл с настройками командной оболочки, а сами параметры определяются через переменную LC_COLORS.

Действительно, в .bashrc обнаружился скрипт следующего содержания:

```
# enable color support of ls and also add handy aliases
if [ -x /usr/bin/dircolors ]; then
    test -r ~/.dircolors && eval "$(dircolors -b ~/.dircolors)" || eval "$(dircolors -b)"
    alias ls='ls --color=auto'
    #alias dir='dir --color=auto'
    #alias vdir='vdir --color=auto'

    alias grep='grep --color=auto'
    alias fgrep='fgrep --color=auto'
    alias egrep='egrep --color=auto'
fi
```

Его суть сводится к проверке на наличие утилиты dircolors в системе, присваивания значения переменной и, при удачном выполнении, использовании параметра --color=auto при каждом вызове ls.

Если запустить утилиту dircolors, она выводит значение переменной LC_COLOR.

```
romaha@romaha-linux ~ Привет! $ dircolors
LS_COLORS='rs=0:di=01;34:ln=01;36:mh=00:pi=40;33:so=01;35:do=01;35:bd=40;33;01:cd=40;33;01:or=40;31
=00:su=37;41:sg=30;43:ca=30;41:tw=30;42:ow=34;42:st=37;44:ex=01;32:*.tar=01;31:*.tgz=01;31:*.arc=01
arj=01;31:*.taz=01;31:*.lha=01;31:*.lz4=01;31:*.lzh=01;31:*.lzma=01;31:*.tlz=01;31:*.txz=01;31:*.tz
l:*.t7z=01;31:*.zip=01;31:*.z=01;31:*.Z=01;31:*.dz=01;31:*.gz=01;31:*.lrz=01;31:*.lz=01;31:*.lzo=01
xz=01;31:*.bz2=01;31:*.bz=01;31:*.tbz=01;31:*.tbz2=01;31:*.tz=01;31:*.deb=01;31:*.rpm=01;31:*.jar=0
.war=01;31:*.ear=01;31:*.sar=01;31:*.rar=01;31:*.alz=01;31:*.ace=01;31:*.zoo=01;31:*.cpio=01;31:*.7
1:*.rz=01;31:*.cab=01;31:*.jpg=01;35:*.jpeg=01;35:*.gif=01;35:*.bmp=01;35:*.pbm=01;35:*.pgm=01;35:
1;35:*.tga=01;35:*.xbm=01;35:*.xpm=01;35:*.tif=01;35:*.tiff=01;35:*.png=01;35:*.svg=01;35:*.svgz=01
mng=01;35:*.pcx=01;35:*.mov=01;35:*.mpg=01;35:*.mpeg=01;35:*.m2v=01;35:*.mkv=01;35:*.webm=01;35:*.o
35:*.mp4=01;35:*.m4v=01;35:*.mp4v=01;35:*.vob=01;35:*.qt=01;35:*.nuv=01;35:*.wmv=01;35:*.asf=01;35:
1;35:*.rmvb=01;35:*.flc=01;35:*.avi=01;35:*.fli=01;35:*.flv=01;35:*.gl=01;35:*.dl=01;35:*.xcf=01;35
=01;35:*.yuv=01;35:*.cgm=01;35:*.emf=01;35:*.ogv=01;35:*.ogx=01;35:*.aac=00;36:*.au=00;36:*.flac=00
m4a=00;36:*.mid=00;36:*.midi=00;36:*.mka=00;36:*.mp3=00;36:*.mpc=00;36:*.ogg=00;36:*.ra=00;36:*.wav
:*.oga=00;36:*.opus=00;36:*.spx=00;36:*.xspf=00;36:';
export LS_COLORS
romaha@romaha-linux ~ Привет! $
```

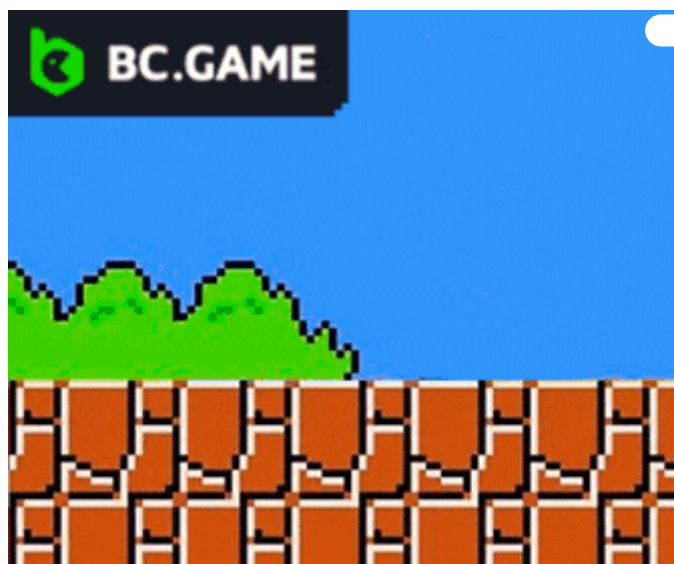
Также возможен вывод значения переменной LC_COLORS в структурированном виде, указанием ключа:

```
$ $ dircolors -p
```

Поскольку вывод получается длинный, приводить его не буду.

Если перенаправить вывод dircolors -p > ~/.dircolors, то будет создан структурированный файл, из которого и будут браться настройки цветов. Реда

данный файл и устанавливаются настройки цвета для вывода `ls`.



В созданном файле `.dircolors` есть подсказка:

```
# Below are the color init strings for the basic file types. A color init
# string consists of one or more of the following numeric codes:
# Attribute codes:
# 00=none 01=bold 04=underscore 05=blink 07=reverse 08=concealed
# Text color codes:
# 30=black 31=red 32=green 33=yellow 34=blue 35=magenta 36=cyan 37=white
# Background color codes:
# 40=black 41=red 42=green 43=yellow 44=blue 45=magenta 46=cyan 47=white
```

Устанавливая последовательность через точку с запятой и получаем нужный нам цвет в консоли `linux`, согласно перечисленным. Цвета терминала `linux` от 30 до 37 для цвета шрифта, а с 40 до 47 для фона. К примеру:

`DIR 01;34` подразумевает, что имена директорий будут выделяться жирным шрифтом и синим цветом.

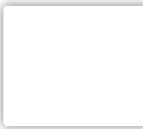
Но таким способом получается всего восемь цветов, а современные эмуляторы терминала способны выводить как минимум 256! Выяснилось, что существуют и другие способы установки цвета. К примеру, значение `EXES 38;5;208;1` - даёт оранжевый толстый шрифт у исполняемых файлов.

Чтобы не перебирать цвета `linux` вручную, есть уже готовая таблица цветов:



	1	2	3	4	5	6	7	8	9
10	11	12	13	14	15		17	18	19
20	21	22	23	24	25	26	27	28	29
30	31	32	33	34	35	36	37	38	39
40	41	42	43	44	45	46	47	48	49
50	51	52	53	54	55	56	57	58	59
60	61	62	63	64	65	66	67	68	69
70	71	72	73	74	75	76	77	78	79
80	81	82	83	84	85	86	87	88	89
90	91	92	93	94	95	96	97	98	99
100	101	102	103	104	105	106	107	108	109
110	111	112	113	114	115	116	117	118	119
120	121	122	123	124	125	126	127	128	129
130	131	132	133	134	135	136	137	138	139
140	141	142	143	144	145	146	147	148	149
150	151	152	153	154	155	156	157	158	159
160	161	162	163	164	165	166	167	168	169
170	171	172	173	174	175	176	177	178	179
180	181	182	183	184	185	186	187	188	189
190	191	192	193	194	195	196	197	198	199
200	201	202	203	204	205	206	207	208	209
210	211	212	213	214	215	216	217	218	219
220	221	222	223	224	225	226	227	228	229
230	231	232	233	234	235	236	237	238	239
240	241	242	243	244	245	246	247	248	249
250	251	252	253	254	255	256			

Подставляя в значение цвета 38;5;x - где x цвет из таблицы, получаем необходимый результат. Получается что-то вроде этого:



```
romaha@romaha-linux ~
Файл Правка Вид Поиск Терминал Вкладки Справка
romaha@romaha-linux ~ x romaha@romaha-linux ~ x + v
# with this program. If not, see <http://www.perlfoundation.
org/artistic_license_1_0>.

# core {{{1
BLK 38;5;68
CAPABILITY 38;5;17
CHR 38;5;113;1
DIR 38;5;183;1
DOOR 38;5;127
EXEC 38;5;208;1
FIFO 38;5;126
FILE 0
LINK target
MULTIHLINK 38;5;222;1
# "NORMAL don't reset the bold attribute -
# https://github.com/trapd00r/LS_COLORS/issues/11
#NORMAL 38;5;254
NORMAL 0
ORPHAN 48;5;196;38;5;232;1
OTHER_WRITABLE 38;5;220;1
SETGID 48;5;3;38;5;0
SETUID 38;5;220;1;3;100;1
SOCK 38;5;197
STICKY 38;5;86;48;5;234
STICKY_OTHER_WRITABLE 48;5;235;38;5;139;3

*LS_COLORS 48;5;89;38;5;197;1;3;4;7 # :-)
# }}}
# documents {{{1
*README 38;5;220;1
*README.rst 38;5;220;1
*LICENSE 38;5;220;1
*COPYING 38;5;220;1
*INSTALL 38;5;220;1
*COPYRIGHT 38;5;220;1
*AUTHORS 38;5;220;1

67,1 4%
```

Но как это работает? Дело в том, что установка цвета вывода информации в эмуляторе терминала осуществляется при помощи ANSI escape-кодов. Это определенная последовательность символов, которая не выводится на экран, однако влияет на вывод. Называется она так, поскольку начинается с \033[- кода "Esc". Поскольку терминалов и эмуляторов терминалов существует множество, эти управляющие коды стандартизированы. Управлением цветом через последовательность от 30 до 37 - это стандарт ECMA-48. Информация о нем содержится в разделе console_codes. А управление через 38;5;* - определено стандартом T.416. Более подробная информация есть, конечно же, в [Википедии](#).

Раскрашиваем и изменяем строку приглашения:

Разобравшись, как устанавливать цвета linux через esc-коды, не составит труда переделать и раскрасить строку приглашения. За ее вывод отвечает переменная PS1, которая определена в файле ~/.bashrc

В Linux Mint скрипт ее инициализации имеет следующий вид:

```
if [ "$color_prompt" = yes ]; then
  if [[ ${EUID} == 0 ]]; then
    PS1='${debian_chroot:+($debian_chroot)}\[\033[01;31m\]\h\[\033[01;34m\] \w \${\033[00m\} '
  else
    PS1='${debian_chroot:+($debian_chroot)}\[\033[01;32m\]\u@\h\[\033[00m\] \[\033[01;34m\]\w Привет! ${\033[00m\} '
  fi
else
  PS1='${debian_chroot:+($debian_chroot)}\u@\h \w \${\033[00m\} '
fi
```

То-есть сперва определяется, запустил ли скрипт root или обычный пользователь. Поскольку в подавляющем большинстве случаев имеет место именно второй вариант, то и редактировать нужно строку, следующую после первого 'else'. Формат строки расшифрован в документации по баш, в разделе "Prompting". Из него следует, что esc-последовательность должна начинаться с "\[" и заканчиваться "\]". Таким образом:

- `\033[01;32m\]` - устанавливает зеленый цвет;
- `\u` - выводит имя пользователя;
- `\h` - выводит имя хоста;
- `\033[00m\]` - очищает установленные атрибуты;
- `\033[01;34m\]` - устанавливает синий цвет;
- `\w` - выводит имя рабочей директории;
- **Привет!** - пример того, что может быть выведен любой производный текст.

Что ещё можно вывести в строке приглашения:

- `\d` - дату, в формате "Wed Dec 28";
- `\A` - время в 24-часовом формате;
- `\!` - номер текущей команды в истории;
- `\n` - перенос вывода на другую строку;

Введем чтонибудь поинтереснее:

```
$ PS1='${debian_chroot:+($debian_chroot)}\[\033[01;32m\]Matrix has you,\u!\n\[\033[00m\]\n \[\033[01;34m\]Blue pill\[\033[00m\] or \[\033[31m\]red?\n\[\033[00m\]\--->\$ '
```

И строка приглашения примет вид:

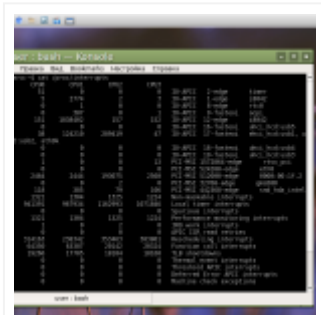
Privacy



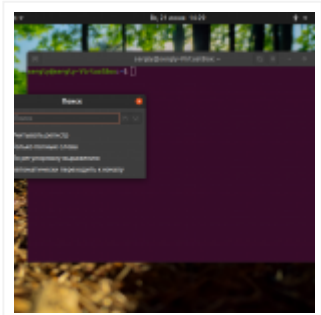
Запуск терминала в Ubuntu



Команды терминала Linux



Лучшие эмуляторы терминала Linux



Горячие клавиши терминала Linux

Оцените статью

★★★★★ (9 оценок, среднее: 4,67 из 5)

📁 [Инструкции](#)

🏷️ [оформление](#), [терминал](#)

Об авторе



ROMWHITE



11 комментариев к “Цвета терминала Linux”



дмитрий

1 января, 2018 в 5:21 пп

с новым годом убунтисты))))

[Ответить](#)



Юрий

1 января, 2018 в 7:13 пп

Вот объясните мне, ну на койэто надо!? Лучше бы дали возможность настройки автоматически запускаемых команд, как в Deepin Terminal или Gnome Terminal! На кой.... мне эти свистоперделки!? Надо функциональность добавлять, а не куету всякую!

[Ответить](#)



geek

15 января, 2018 в 12:37 пп

Ага давайте ломать глаза об белые буквы на чёрном фоне. И вообще, это же open source, если сильно надо можно пойти и дописать что нужно

[Ответить](#)



Олег

17 января, 2018 в 7:53 пп

Познавательный текст интересный... но по существу действительно - херня. Куда интереснее было бы рассмотрение цветовой разметки на терминале

результата работы скрипта bash ... или кода C/C++, как это делает например, проект: <https://github.com/agauniyal/rang>

[Ответить](#)



proger

[17 января, 2018 в 7:00 дп](#)

Я тоже как-то с этой хренью мучился. Хотел покрасивее консоль сделать. А потом просто открыл для себя zsh и oh-my-zsh. Лучше бы об этом статью написали (хотя там и так все просто в принципе).

[Ответить](#)



Серёга

[3 ноября, 2018 в 12:28 дп](#)

А мне не нравится стандартный вывод даты и времени, поэтому я в своём Mint LMDE 3 так забацал (здесь ещё код ошибки выполнения предыдущей команды выдаётся, если таковая была, например: не выполнилось условие - 1):

```
[code]if [[ ${EUID} == 0 ]] ; then
PS1='$(e=$?; (( $e != 0 )) && echo -e "\033[33mError:
\033[31;1m$e\n\n")\n${debian_chroot:+($debian_chroot)}\[\033[01;31m\]\u\
[\033[00m\]@\[\033[01;31m\]\h \[\033[00m\]$(date -u "+e.%m.%Y %A %T") \
[\033[01;34m\]\w \$\[\033[00m\] '
else
PS1='$(e=$?; (( $e != 0 )) && echo -e "\033[33mError:
\033[31;1m$e\n\n")\n${debian_chroot:+($debian_chroot)}\[\033[01;32m\]\u\
[\033[00m\]@\[\033[01;32m\]\h \[\033[00m\]$(date -u "+e.%m.%Y %A %T") \
[\033[01;34m\]\w \$\[\033[00m\] '
fi[/code]
```

Т.е., по сути - можно любые команды выполнять в PS-переменной 😊

[Ответить](#)



Серёга

[3 ноября, 2018 в 12:34 дп](#)

Privacy

И тут же переделал на более удобное, т.к. терминалом пользуюсь постоянно:

```
if [[ ${EUID} == 0 ]] ; then
PS1='${e=$?; (( $e != 0 )) && echo -e "\033[33mError:
\033[31;1m$e\n\n")}\n${debian_chroot:+($debian_chroot)}\[\033[01;31m\]\u\
[\033[00m\]@\[\033[01;31m\]\h \[\033[00m\]$(date -u "+e.%m.%Y %A %T" | sed
"s/^ //" ) \[\033[01;34m\]\w\n$\[\033[00m\] '
else
PS1='${e=$?; (( $e != 0 )) && echo -e "\033[33mError:
\033[31;1m$e\n\n")}\n${debian_chroot:+($debian_chroot)}\[\033[01;32m\]\u\
[\033[00m\]@\[\033[01;32m\]\h \[\033[00m\]$(date -u "+e.%m.%Y %A %T" | sed
"s/^ //" ) \[\033[01;34m\]\w\n$\[\033[00m\] '
fi
```

[Ответить](#)**Серёга**[3 ноября, 2018 в 12:44 дп](#)

И допеределал уже и под chroot:

```
if [[ ${EUID} == 0 ]] ; then
PS1='${e=$?; (( $e != 0 )) && echo -e "\033[33mError: \033[31;1m$e\n\n")}\n\
[\033[00;34m\]${debian_chroot:+($debian_chroot)}\[\033[01;31m\]\u\
[\033[00m\]@\[\033[01;31m\]\h \[\033[00m\]$(date -u "+e.%m.%Y %A %T" | sed
"s/^ //" ) \[\033[01;34m\]\w\n$\[\033[00m\] '
else
PS1='${e=$?; (( $e != 0 )) && echo -e "\033[33mError: \033[31;1m$e\n\n")}\n\
[\033[00;34m\]${debian_chroot:+($debian_chroot)}\[\033[01;32m\]\u\
[\033[00m\]@\[\033[01;32m\]\h \[\033[00m\]$(date -u "+e.%m.%Y %A %T" | sed
"s/^ //" ) \[\033[01;34m\]\w\n$\[\033[00m\] '
fi
```

[Ответить](#)**Nick**[28 ноября, 2018 в 12:04 пп](#)

Так всё-так как мне из программы вывести в терминал командой printf что-нибудь типа анаогичного HTML:
Это красный и зелёный цвета
?

[Ответить](#)



Serge

[6 января, 2021 в 9:02 дп](#)

Спасибо!!! Чуть ли не единственное нормальное пояснение, на просторах этого интернета, что и как выполнить за 1.5 минуты. (а то всякие индусо-китайцы Вам напишут....."ответ Чемберлену"....)

[Ответить](#)



Евгений

[26 апреля, 2021 в 2:54 пп](#)

Когда у меня открыто несколько окон терминала: локальная машина, тестовый сервер и боевой, удобно различать их по цветам строки приглашения - например, локальный белый или зелёный, тестовый - жёлтый, боевой - красный.

[Ответить](#)

Оставьте комментарий



Privacy

Имя *

Email

☐ Я прочитал и принимаю политику конфиденциальности. Подробнее [Политика конфиденциальности](#) *

Комментировать

Русский

Поиск

ПОИСК ПО КОМАНДАМ

Начните вводить команду

Поиск



Начните изучать
Linux прямо сейчас!

Карта сайта



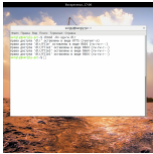
Как пользоваться
редактором Vim

Полезно

Лучшие

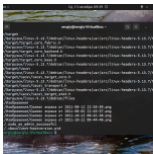
Свежие

Теги



Команда chmod Linux

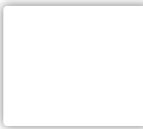
2020-04-13



Команда find в Linux

2021-10-17

Как узнать IP-адрес Linux



Privacy



2023-04-14



Настройка Sion

2021-10-01



Права доступа к файлам в Linux

2020-10-09

РАССЫЛКА

Ваш E-Mail адрес

☐ Я прочитал(а) и принимаю политику конфиденциальности

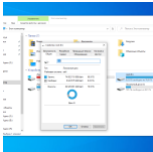
Sign up



Privacy

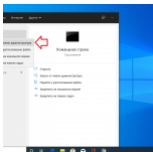
Windows

Списки



Подключение Ext4 в Windows

2023-02-18



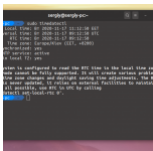
Восстановление Grub после установки Windows 10

2020-08-15



Установка Linux рядом с Windows 10 или 11

2023-02-08



Сбивается время в Ubuntu и Windows

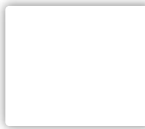
2023-02-18

Смотреть ещё

МЕТА

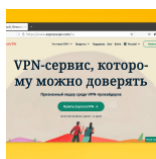
- Регистрация
- Войти
- Лента записей
- Лента комментариев

СЛЕДИТЕ ЗА НАМИ В СОЦИАЛЬНЫХ СЕТЯХ



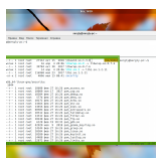


Интересное



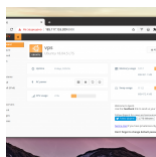
Лучшие VPN сервисы для Linux

2022-10-10



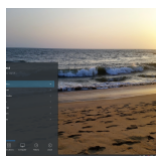
Шпаргалка по tmux

2021-10-01



Лучшие панели управления для Linux

2020-12-08



Самые маленькие дистрибутивы Linux

2020-12-15

©Losst 2024 CC-BY-SA [Политика конфиденциальности](#)

