

Prompt Tips for bash and tcsh, and some stuff for Emacs and L^AT_EX

Luke Abraham

I've made a few changes to my prompts. Here is a list of things that I've added.

I. BASHRC

These lines should be added to your `.bashrc` file.

To have colour highlighting, use the `--color=tty`. The `-F` option puts slashes after directories, asterisks after executables etc. Try `'man ls'` for a more detailed description and other options. Combined use

```
alias ls="ls -F --color=tty"
```

to change

```
directory      executable
```

to

```
directory/     executable*
```

Next, the prompt. I like the simple style of prompt which is

```
[username@machine sdir]$
```

This can be achieved by

```
export PS1="[\u@\h \w]\\$ "
```

Where

`\u` = username

`\h` = machine

`\w` = `sdir` (just gives the directory rather than `/home/username/dir` - this would be `\w`)

I also quite like colour, giving a prompt like

```
[username@machine sdir]$
```

Which is given by

```
export PS1="[\[\e[34m\]\u@\[\e[31m\]\h \w\[\e[0m\]\]\\$ "
```

or, if you want the colours to be slightly brighter you can do it in bold

```
[username@machine sdir]$
```

```
export PS1="[\[\e[34;1m\]\u@\[\e[31;1m\]\h \w\[\e[0m\]\]\\$ "
```

I also have it so that the title text at the top of the terminal window also shows the username and location in the form

```
username@machine:~/dir
```

All these can be incorporated into a little check like the one below. If you're not in an xterm the prompt is

just the simple one described above.

```
if [ "$TERM" = "xterm" ]
then
  #we're not on the console, assume an xterm
  export PS1="\[\[e]2;\u@\H:\w\a[e[34;1m]\]\u@\[e[31;1m]\]\h \w\[\[e[0m]\]\]\$ "
else
  #we're on the system console or maybe TELNETting in
  export PS1="\u@\h \w]\]\$ "
fi
```

You could also go to <http://www-106.ibm.com/developerworks/library/l-tip-prompt/>.

II. CSHRC,TCSHRC

These lines should be added to your `.cshrc` or `.tcshrc` file.

To give directories as `dir/` and executables as `exe*` use

```
alias ls 'ls -F'
```

I haven't found a colour option.

If the machine doesn't have 'less' use

```
alias less 'more'
```

For the simple prompt

```
set prompt="[\n@\m %C]## "
```

For the coloured option

```
set prompt="[\% \033[34m%%n@\% \033[31m%%m %C% \033[38m%]## "
```

or the brighter coloured option

```
set prompt="[\%B% \033[34m%%n@\% \033[31m%%m %C% \033[38m%%b]## "
```

and for the full prompt with title as before

```
switch ($TERM)
case "xterm*":
  set prompt="%\033]0;\n@\m:%~\007%[\%B% \033[34m%%n@\% \033[31m%%m %C% \033[0m%%b]## "
  breaksw
default:
  set prompt="[\n@\m %C]## "
  breaksw
endsw
```

I also recommend looking at <http://www.faqs.org/docs/Linux-mini/Xterm-Title.html>.

III. ZSH

I have recently been introduced to `zsh`, where the above prompt is achieved by putting the line

```
export PS1="[%n@m %c]$ "
```

(like bash) in your `.zshrc` file. To get colours, include the lines

```
autoload colors zsh/terminfo
  if [[ "$terminfo[colors]" -ge 8 ]]; then
    colors
  fi
  for color in RED GREEN YELLOW BLUE MAGENTA CYAN WHITE; do
    eval PR_$color='%{$terminfo[bold]$fg[{$(L)color}]%}'
    eval PR_LIGHT_$color='%{$fg[{$(L)color}]%}'
    (( count = $count + ))1
  done
  PR_NO_COLOUR="%{$terminfo[sgr0]%"
```

with the coloured prompt given by

```
use_color=true
PROMPT='[%{$fg[blue]}%]%n@%{$fg[red]}%}%m% %c%{$fg[default]}%]]%# '
```

You can change your shell using the

```
chsh
```

command.

IV. EMACS

A good website on Emacs is <http://cip.physik.uni-wuerzburg.de/virtualmanuals/root-doku/emacs/emacs>. This also has good explanations of how to use most of the Emacs features, such as the spell-checker.

The main thing that I feel Emacs should do automatically but does not is putting the name of the file being edited on the top of the window and in the icon on the status bar. To do this add these lines below into your `.emacs` file which should be in your home directory

```
(setq frame-title-format
  '(("%b (Emacs: " user-login-name "@" system-name ")") )
```

Which gives the same output on both the top of the window and the icon, which is

```
file.name (Emacs: username@Machine)
```

To change the font and the default foreground and background colours use the

```
(set-default-font "font name")
(set-foreground-color "colour name")
(set-background-color "colour name")
```

To see what options there are use the

```
M-x set-default-font RET
```

where RET is the return key. Then type

?

Here `M-x` stands for `Meta-x` where the `Meta` key is the left `Alt` key. `M-x` means to press the left `Alt` key and the `x` key at the same time, just like `C-x`.

To see a full list of colours use the

```
M-x list-colors-display RET
```

Changing your default font will change the size of the window opened when emacs is next loaded, if the font size is larger than previously.

I can also recommend the use of the

```
(setq auto-mode-alist
      (append '(("\\.pstex" . tex-mode)
              ) auto-mode-alist))
```

To add extra file types (such as in the case above, adding `file.pstex` to the `TEX` family). If you use `POV-Ray` (<http://www.povray.org/>) then I can recommend using `pov-mode` (<http://www.acc.umu.se/~woormie/povray/#pov-mod>) which works like `tex-mode` with extra menus and syntax highlighting for `.pov` files.

Also, pressing `Esc` three times while in the mini-buffer will cancel the command it was trying to execute.

Another useful thing to be able to do is column (rather than row) selection. In Emacs this is a bit tricky. In `NEdit` a simple `Ctrl-Mouse 1` drag will select columns instead of rows in the normal way that `Mouse 1` dragging works.

In Emacs the way columns are manipulated is by using the rectangle key bindings (`C-x r-`). To see a full list of options type

```
C-x r C-h
```

You will also need to set the mark (`C-Space` or another way of setting the mark - see the URL above for more information) on the top-left-hand corner of the rectangular column you which to manipulate. Move the cursor (using the arrow keys) to the bottom right hand corner of the space. As well as the commands given (for clear, delete, yank and kill), these for copy and paste are also quite useful: to copy the area type

```
C-x r r r
```

and to paste the area move the cursor to where you want the top-left-hand corner to be, and then type

```
C-x r i r
```

This can be pasted any number of times.

V. CONVERTING DVI FILES TO PDF

When converting `.dvi` files to `.pdf` files (from `LATEX` for instance), I have found that you often lose the top margin, and it is impossible to fix this with commands within the document. I have found a handy website, <http://www.astro.psu.edu/gradinfo/psuthesis/pdf.html> (it should be noted that this site is an American one which is designed for converting files to *letter* format and not *A4*), which gives the correct commands for *A4* as:

```
dvips -t a4 foo.dvi -Ppdf -o foo.ps
```

followed by the line

```
ps2pdf foo.ps foo.pdf
```

The `-t a4` will sort out the margin problem, and the `-Ppdf` is not strictly needed, but it helps with fonts. You must use the two commands, rather than using `dvipdf`, since you cannot specify everything that you need.

VI. L^AT_EX FONTS

Just a quick word about L^AT_EX fonts. I quite like Palatino (which is also the official font of the University). In order to get this for a L^AT_EX document, add change the lines (assuming you have them)

```
\usepackage[T1]{fontenc}  
\usepackage{times}
```

to

```
\usepackage[T1,OT1]{fontenc}  
\usepackage{palatino}
```