How To: Show Colour Numbers in Unix Terminal

I’m decorating my tmux setup and needed to confirm colour numbers for some elements of the interface. Turns out, it’s simple enough to show all the possible colours with a 1-liner in your favourite Unix shell – bash shell in my case.

Using ESC sequences For Using Colours

I’ll explain how this works in full detail sometime in a separate post, but for now will just give you an example and show how it works:
So, in this example, this is how we achieve colorized text output:

1. echo command uses -e option to support ESC sequences
2. \e[38;5;75m is the ESC sequence specifying color number 75.
3. \e[38;5; is just a special way of telling terminal that we want to use 256-color style

**List 256 Terminal Colours with Bash**

Here’s how we get the colours now: we create a loop from 1 until 255 (0 will be black) and then use the ESC syntax changing colour to $COLOR variable value. We then output the $COLOR value which will be a number:

```bash
for COLOR in {1..255}; do echo -en "\e[38;5;${COLOR}m${COLOR}"; done; echo;
```

Here’s how running this will look in a properly configured 256-color terminal:
Bash Script to Show 256 Terminal Colours

Here’s the same 1-liner converted into proper script for better portability and readability:

#!/bin/bash

for COLOR in {1..255}; do
    echo -en "\e[38;5;${COLOR}m"
    echo -n "${COLOR} "
done

If you save this as `bash-256-colours.sh` and `chmod a+rx bash-256-colours.sh`, you can now run it every time you want to refresh your memory or pick different colours for some use.

See Also

- `tmux`
- `tmux settings`
- `Colorized ls output`