

Assign commands to key

A lot of programs (especially Bash) use the GNU Readline library for command line input. Readline allows to define each key with a custom value. This definition is stored in the file `.inputrc` in the users home directory.

If you want to assign an often used command, e.g. `less /var/log/message` to the key F1, you'll need to add the following line to your `.inputrc`:

```
"e[11~": "less /var/log/messages^M"
```

The last char in `e[11~` is a tilde char. All characters have to be entered exactly as given, with only one exception: the `^M` at the end. The `^M` means Enter and causes a direct execution of the command. Without the `^M`, the

command would be simply copied into the command line and which would wait for the input of enter or a different char.

The way you enter the `^M` depends on the editor you use. In vi you have to press `ctrl-V` and Enter.

It has to be one single char (ASCII-code 13).

The character sequence `e` will be interpreted as an Escape char from Readline. The given sequence works in X11. In the console the sequence differs, but both definitions can be in the same `.inputrc` file.

The change in `.inputrc` applies with a fresh started Shell. You can refresh the setting with the key sequence `ctrl-X` and `ctrl-R` in the running shell. F1 should work now.

How to use the trick with other keys? First you have to know the code that the key generates. You can get the code when you press `ctrl-V` and the desired key.

You will get an Escape-sequence (the first 2 chars are `^[` which is equal to `ctrl-[` and equal to Escape)

If you replace `^[` with `e`, you got the sequence for `.inputrc`.

More details of the usage of readline can be found in the bash manpage (`man bash`).

Translated from

<http://www.pro-linux.de/news/2002/0040.html>