

.bashrc

Aufgabe 1: History

Erweitere die history list so, dass sie Platz für 2000 Befehle hat und 200000 Befehle in einer Datei bewahrt.

Was ändert sich mit der Variable
`HISTTIMEFORMAT='%F_%T'`

Aufgabe 2: Prompt

Lies das Kapitel „Prompting“ in der „man“-Page für „bash“. Erstelle einen Prompt-String welcher die folgenden Daten beinhaltet:

- Loginname
- Hostname des Rechners
- die History-Nummer des aktuellen Kommandos, in Klammern „z. B. (10)“
- das aktuelle Verzeichnis
- Prompt-Zeichen für normalen Benutzer oder „root“

Baue das neue Prompt in das Profile-Startscript für die Bash für deinen aktuellen Benutzer ein.

profile oder bashrc?

umask und exportierte Variablen gehören besser in die `.profile`, weil beides an die Kindprozesse vererbt wird.

Eine Ersetzung in der Form `$PATH="$PATH:$HOME/bin"` würde z. B. mit jeder weiteren Subshell ausgeführt und jedes Mal neu evaluiert werden, so dass dieselbe Ergänzung jedes Mal wieder stattfindet.

Beispiel Umgebungsvariable setzen

`~/profile`

```
...  
VAR=wert  
export VAR
```

Abmelden: Grafisch oder

```
loginctl list-sessions
```

```
loginctl terminate-session 5
```

Wieder anmelden.

```
echo $VAR
```

(Neustart `gnome-terminal` reicht nicht)

Häufige Kommandofolgen abkürzen

`ls -l` abkürzen:

```
alias ll='ls -l'
```

Aliase anzeigen:

```
alias
```

Dateiinhalt ohne Kommentare ausgeben:

```
alias ccat='egrep -v "^\s*(#|$)"'
```

oder

```
function ccat {  
    egrep -v '^\s*(#|$)' "$1"  
}
```

Dateiinhalt ohne Kommentare durchsuchen:

```
function cgrep {  
    egrep -v '^\s*(#|$)' "$2" | grep --colour=auto "$1"  
}
```

Funktionen und Variablen anzeigen:

```
set
```

Funktion löschen

```
unset -f ccat
```

Dokumentation

```
man bash
```

/INVOCATION

help function

From:

<https://wiki.lab.linuxhotel.de/> - **Linuxhotel Wiki**

Permanent link:

https://wiki.lab.linuxhotel.de/doku.php/lpi1:shell_configdateien

Last update: **2022/05/09 15:14**

