

2.7.A Skripte/Funktionen – Arbeitsblatt

- Bearbeitungsdauer: 20 min
- Ergebnissicherung: Auf dem Server, Dateinamen siehe Aufgaben
- Hilfsmittel: Lerntext

Aufgabe 1: Verbesserte help-Funktion

Die Hilfe für Programme (`ls`, `mkdir`, `oneko`, ...) erhält man mit dem Programm `man`. Die Hilfe für in die Shell eingebaute Funktionen (`cd`, `set`, `read`, ...) und einige der Shell-Schlüsselwörter (`for`, `if`, `case`, ...) erhält man mit der in der Shell eingebauten Funktion `help`. Was für die Programme `man` ist, das ist für Shell-Elemente `help`.

Aber anders als der Befehl `man` ruft die eingebaute Shell-Funktion `help` nicht den Pager `less` auf. Das heißt, die gesamte Hilfe wird platt ausgegeben. Bei längeren Hilfeseiten ist also das meiste nach oben durchgerutscht, bevor man es gesehen hat. Abhilfe besteht darin, die Ausgabe von `help` durch eine Befehls-Pipeline an `less` weiterzugeben:

```
schueler@debian964:~$ help -m set | less
```

Einfacher wäre es allerdings, wenn es eine Funktion gäbe (z. B. mit dem Namen `h()`), die automatisch die Ausgabe von `help` an `less` weitergibt:

```
schueler@debian964:~$ h -m set
```

- Schreiben Sie eine Funktion `h()`, die die gewünschte Hilfe-Seite aufruft und an `less` weitergibt! Alle Optionen für `help` sollen dabei an `h()` weitergegeben werden.
Ergebnis: `helpless.sh`
- Ergänzen Sie die Datei `.bashrc` in Ihrem persönlichen Verzeichnis um die Definition von `h()`, so dass Sie `h()` nach jedem Start einer interaktiven Shell zur Verfügung haben!

Aufgabe 2: Verzeichniswechsel mehrere Ebenen nach oben

Mit der Befehlszeile `cd ..` kann man genau ein Verzeichnis nach oben, also in Richtung Stammverzeichnis, wechseln. Schön wäre es, wenn man in einem Schritt mehrmals `cd ..` ausführen könnte.

- Schreiben Sie eine Funktion mit dem Namen `...` (2 Punkte nacheinander), mit der man einmal `cd ..` aufruft!
Ergebnis: `doppelpunkt1.sh`
- Erweitern Sie die Funktion `...`, so dass man mit dem ersten Parameter angeben kann, wie oft `cd ..` aufgerufen wird. Falls der Parameter fehlt, soll `cd ..` nur einmal aufgerufen werden (Default-Wert).
Ergebnis: `doppelpunkt2.sh`

Aufgabe 3: Modularisierung des Start-Skriptes

Funktionen eignen sich auch dazu, große Skripte in handliche und verständliche Abschnitte zu zerteilen.

- Zerteilen Sie das Start-Skript `mstartB` in mehrere Funktionen!
Ergebnis: `mstartC`