LU06b - Einführung in Iterationen in Bash

Shell Prog: Kapitel 4.10 - 4.11

Iterationen (Schleifen) in Bash ermöglichen es, wiederholt eine Gruppe von Befehlen auszuführen, bis eine bestimmte Bedingung erfüllt ist. Bash bietet dafür mehrere Arten von Schleifen, die für unterschiedliche Zwecke verwendet werden können.

for-Schleife

Die **for-Schleife** wird verwendet, um durch eine Liste von Werten oder Dateien zu iterieren.

Grundstruktur:

```
for variable in Liste; do
    # Befehle
done
```

Beispiel:

```
#!/bin/bash
for name in Alice Bob Charlie; do
    echo "Hallo, $name!"
done
```

Ausgabe:

```
Hallo, Alice!
Hallo, Bob!
Hallo, Charlie!
```

for-Schleife für Dateien:

```
#!/bin/bash
for file in *.txt; do
    echo "Datei: $file"
done
```

while-Schleife

Die **while-Schleife** führt Befehle aus, solange eine Bedingung wahr ist.

Grundstruktur:

```
while [ Bedingung ]; do
    # Befehle
done
```

Beispiel:

```
#!/bin/bash
zahl=1
while [ $zahl -le 5 ]; do
    echo "Zahl: $zahl"
    zahl=$((zahl + 1))
done
```

Ausgabe:

```
Zahl: 1
Zahl: 2
Zahl: 3
Zahl: 4
Zahl: 5
```

Alle Zeilen einer Datei lesen

Diese Iteration liest eine Datei zeilenweise und stellt die Daten in die Variable `data`. Die Variable mit dem Dateinamen wird in der letzten Zeile mittels Umleitung (<) angegeben.

```
filename=someFile.dat
while read data; do
    # Process the data
done < $filename

===== until-Schleife =====

Die **until-Schleife** ist das Gegenteil der **while-Schleife**: Sie läuft,
bis eine Bedingung wahr wird.

==== Grundstruktur: ====
<code bash>
until [ Bedingung ]; do
```

```
    # Befehle
done
```

Beispiel:

```
#!/bin/bash
zahl=1
until [ $zahl -gt 5 ]; do
    echo "Zahl: $zahl"
    zahl=$((zahl + 1))
done
```

Ausgabe:

```
Zahl: 1
Zahl: 2
Zahl: 3
Zahl: 4
Zahl: 5
```

Schleifensteuerung

Bash bietet Befehle, um Schleifen zu kontrollieren:

- **break**: Beendet die Schleife vorzeitig.
- **continue**: Überspringt den aktuellen Schleifendurchlauf.

Beispiele:

```
# Mit break
for i in {1..10}; do
    if [ $i -eq 5 ]; then
        break
    fi
    echo "Zahl: $i"
done
```

```
# Mit continue
for i in {1..5}; do
    if [ $i -eq 3 ]; then
        continue
    fi
    echo "Zahl: $i"
done
```

Zusammenfassung

Schleife	Beschreibung	Verwendung
for	Iteriert über eine Liste von Werten oder Dateien	Wiederholung bei bekannter Anzahl von Werten
while	Läuft, solange eine Bedingung wahr ist	Dynamische Bedingung
until	Läuft, bis eine Bedingung wahr wird	Dynamische Bedingung, Gegenteil von while
break	Beendet die Schleife	Vorzeitiges Ende
continue	Überspringt den aktuellen Schleifendurchlauf	Fortsetzen der Schleife

M122-LU06

Marcel Suter

From:
https://wiki.bzz.ch/ - **BZZ - Modulwiki**

Permanent link:
https://wiki.bzz.ch/modul/m122/learningunits/lu06/iteration

Last update: **2024/12/04 19:12**