

Bash script alapok 5. rész

Tömbök



Linux alapok

Varga Tibi 2020

Bash tömbök

A programozás során gyakran szükség van a tömbök használatára, melyekkel egyszerre több értéket is tárolhatunk, amiket utána akár ciklusokban dolgozhatunk fel automatizáltan. Nincs ez másképp a Shell scriptek esetén sem.

A BASH egy dimenziós tömböket kezel, amelyek lehetnek

- **Indexeltek**
- **Asszociatívak**

Indexelt tömb

Egy inicializált tömb létrehozása:

1. Zárójelekkel (ügyeljünk a szóközökre! A végén is!)

tomb1 =("egy" "kettő" "egy" "akármí" 101)

2. Index - érték párral:

tomb[1]=200

vagy szöveges érték esetén idézőjelekkel:

tomb[2]="több szavas érték "

3. Deklarálással: (-a kapcsolóval)

\$ declare -a disztrok=('Debian , ' Ubuntu' 'Kali' 'CentOS')

```
#!/bin/bash
echo indexelt tömb létrehozása
# 1. zárojelekkel
tomb1=("első" "második" 2021 "kétezer")

echo a 1.tömb tartalma: ${tomb1[@]}

# 2. index-érték párokkal
tomb[0]=ez
tomb[1]=meg
tomb[2]=az
tomb[3]=3089

echo a tömb tartalma: ${tomb[@]}

# 3. deklarálással
declare -a tomb3=("Debian" "Ubuntu" "Kali" "CentOS")

echo a 3. tömb tartalma: ${tomb3[@]}
```

```
tibi@tibi-server:~/test$ ./tomb.sh
indexelt tömb létrehozása
a 1.tömb tartalma: első második 2021 kétezer
a tömb tartalma: ez meg az 3089
a 3. tömb tartalma: Debian Ubuntu Kali CentOS
```

Műveletek indexelt tömbbel

- A tömb tartalmának megjelenítése:

echo \${disztrok[@]}

- A tömb elemeinek száma:

echo \${# disztrok[@]}

- Az n. elem hossza:

echo \${#l disztrok[2]}

- A tömb egy elemének (2) visszaadása:

echo \${disztrok[2]}

```
1  #!/bin/bash
2  declare -a disztrok=("Debian" "Ubuntu" "Kali" "CentOS" "SUse")
3
4  echo a 3. tömb tartalma: ${disztrok[@]}
5  # tömb elemeinek meg számolása
6  echo a 3. tömb elemeinek száma: ${#disztrok[@]}
7
8  echo Tömb elemek hossza
9  # index bekérése
10 echo addmeg melyik elem hosszára vagy kíváncs:
11 read i
12 echo a 3. tömb n elem hossza: ${#disztrok[$i]}
13
14 #egy elem kiiratása index alapján
15 echo ez az elem: ${disztrok[$i]}
```

```
tibi@tibi-server:~/test$ ./tomb2.sh
a 3. tömb tartalma: Debian Ubuntu Kali CentOS SUse
a 3. tömb elemeinek száma: 5
Tömb elemek hossza
addmeg melyik elem hosszára vagy kíváncs:
2
a 3. tömb n elem hossza: 4
ez az elem: Kali
```

Műveletek indexelt tömbbel

A tömb egy részének visszaadása:

- A disztrok tömb 1. elemétől 2 elem visszaadása

echo \${disztrok[@]:1:2}

- Hasonló felállítás, de a tömb adott elemére

echo \${disztrok[3]:2:2}

- Új elemek hozzáadása a tömbhöz:

disztrok+=("Manjaro" "Kubuntu") vagy

disztrok=({disztrok[@]} 'Manjaro' 'Kubuntu')

- Törlés a tömbből:

unset disztrok[5]

Műveletek indexelt tömbbel

```
#!/bin/bash
declare -a disztrok=( "Debian" "Ubuntu" "Kali" "CentOS" "SUse" )
# a tömb 1. elemétől 2 elem visszaadása
echo ez az elemek: ${disztrok[@]:1:2}

# A 3. elem 2. tól 2 rész visszaadása
echo a 3. elem részei: ${disztrok[3]:2:2}

# Új elem hozzáadás a tömbhöz
disztrok=(${disztrok[@]} 'Mandrake' 'FreeBSD')
echo elemek a tömbben ${disztrok[@]}

# törlés tömbből
echo "hanyadik elemet töröljem?"
read i
unset disztrok[$i]
echo elemek a tömbben: ${disztrok[@]}
```

```
tibi@tibi-server:~/test$ ./tomb2.sh
ez az elemek: Ubuntu Kali
a 3. elem részei: nt
elemek a tömbben Debian Ubuntu Kali CentOS SUse Mandrake FreeBSD
hanyadik elemet töröljem?
4
elemek a tömbben: Debian Ubuntu Kali CentOS Mandrake FreeBSD
```

Asszociatív tömb

Asszociatív tömb Létrehozása:

- Csak deklarációval (figyeljünk a -A kapcsolóra)
- **[kulcs] – " érték "** párok formájában:

```
declare -A szavak=([egy]="one" [ketto]="two" [harom]="three")
```

Egy elem kiírás:

```
echo ${szavak[egy]}
```

one

Indexek kiírása:

```
echo ${!szavak[@]}
```


Asszociatív tömb

Asszociatív tömb létrehozása majd értékadása:

declare -A proba

Ezután már létrejön a tömbünk.

Most már adhatunk is hozzá értékeket többféleképpen is:

proba[elem1]="Első elem"

proba["elem2"]="Második elem"

proba[kecske]=káposzta

Asszociatív tömb létrehozása és kiolvasása ciklussal

```
declare -A tomb2=( [első]="1." [második]="2." [harmadik]="3." )
```

```
echo "tomb2 asszociatív tömb kilistázása:"
```

```
    for i in ${!tomb2[@]} ; do
```

```
        echo "$i: ${tomb2[$i]}"
```

```
    done
```

Asszociatív tömb kiolvasása ciklussal

```
#!/bin/bash
declare -A tomb2=( [elso]="Debian"
                  [masodik]="Ubuntu" [harmadik]="Kali"
                  [negyedik]="CentOS" [otodik]="SUSE" )
# kiírás ciklussal
echo Tömb elemek kiírása ciklussal

for i in ${!tomb2[@]} ; do
    echo Kulcs: $i Érték: ${tomb2[$i]}
done
```

```
tibi@tibi-server:~/test$ ./tombcikl3.sh
Tömb elemek kiírása ciklussal
Kulcs: elso érték: Debian
Kulcs: negyedik érték: CentOS
Kulcs: otodik érték: SUSE
Kulcs: masodik érték: Ubuntu
Kulcs: harmadik érték: Kali
```

Tömb vizsgálata elágazással

Logikai vizsgálat (létezik-e a kulcs)

```
#!/bin/bash
declare -A tomb2=( [első]="Debian"
                  [második]="Ubuntu" [harmadik]="Kali"
                  [negyedik]="CentOS" [ötödik]="SUSE" )
# kiírás ciklussal
echo "Melyik kulcsot vizsgáljuk?"
read i

if [[ ${tomb2[$i]} ]]; then
    echo Tömb $i kulcsú eleme létezik
    echo Érték: ${tomb2[$i]}
else
    echo nincs ilyen kulcs a tömbben
fi
```

```
tibi@tibi-server:~/test$ ./tombcikl4.sh
Melyik kulcsot vizsgáljuk?
ötödik
Tömb ötödik kulcsú eleme létezik
Érték: SUSE
tibi@tibi-server:~/test$ ./tombcikl4.sh
Melyik kulcsot vizsgáljuk?
hatodik
nincs ilyen kulcs a tömbben
```

Tömb vizsgálata elágazással

Érték vizsgálat (A megadott kulcs és hozzátartozó érték vizsgálata)

```
#!/bin/bash
declare -A tomb2=( [első]="Debian"
                  [második]="Ubuntu" [harmadik]="Kali"
                  [negyedik]="CentOS" [ötödik]="SUSE" )
# kiírás ciklussal
echo "Melyik kulcsot vizsgáljuk?"
read i

if [[ "${tomb2[$i]}" = "Kali" ]]; then
    echo Az érték helyes
    echo Kulcs $i Érték: ${tomb2[$i]}
else
    echo Az érték nem egyezik
fi
```

```
tibi@tibi-server:~/test$ ./tombcikl5.sh
Melyik kulcsot vizsgáljuk?
második
Az érték nem egyezik
tibi@tibi-server:~/test$ ./tombcikl5.sh
Melyik kulcsot vizsgáljuk?
harmadik
Az érték helyes
Kulcs harmadik Érték: Kali
tibi@tibi-server:~/test$
```

Tömbök használata függvényekben

```
#!/bin/bash
```

```
declare -A tomb=(  
    [elem1]="Első elem"  
    [elem2]="Második elem"  
    [elem3]="Harmadik elem"  
    [több szavas]=10)
```

```
function fuggveny {  
    tomb2=("$@")           # Összes paraméter begyűjtése egy másik tömbbe  
    for i in "${!tomb2[@]}"; do  
        ertek=${tomb2[$i]}  
        echo "$i: $ertek"  
    done  
}  
fuggveny "${tomb[@]},,    # Függvény hívása a kibontott asszociatív tömbbel
```