

# Introduction to KornShell

Soha Péter

2018. november 26.

## Kivonat

Csak saját felelősségre! KSH szakértők **NE** olvassák!

## 1. Részletes dolgok

**\$0**: az éppen végrehajtott script relatív útvonalát tartalmazza az aktuális könyvtárhoz képest.

**\${sh.file}**: KornShell specifikus cucc, majdnem ugyanazt tudja, mint a **\$0**, csak ez abszolút path-t ad vissza

**variable=\$(script)**: végrehajtja az argumentumában lévő parancsokat és a printoutját behelyettesíti a *variable* változóba.

**pwd**: **P**rint **W**orking **D**irectory. Az aktuális könyvtár *abszolút* elérési útját adja vissza (alapértelmezetten printeli).

**source** vagy **.** *fájlnev*: "Bemásolja" futásidőben a mögötte lévő fájl tartalmát az aktuális script "elejére". Gyakorlatilag, mint a *java import* vagy *c++ include*.

**Ez az sor**: "`[[ feltetel ]]&& utasitas1 || utasitas2`" ezt jelenti:

```
if [feltetel]; then
utasitas1
else
utasitas2
fi
```

*fontos!* Az utasitas1 mindenképp igazzal kell visszatérjen, mert ellenkező esetben a logikai vagy miatt az utasitas2 is végrehajtná ( mivel a logikai és mindkét operadusának igaznak kell lennie, vagy az eredmény hamis).

**read**: beletesz egy *printout*-ot egy változóba. Szintaxisa: `dirname $0 | read valtozonev`. Ez ekvivalens azzal, hogy: `valtozo=$(dirname $)`

**stream redirect:** *echo* "valami" [*> &t1 / > &t2*]: "valami kiírása a standard outputra / error streamre"

**Stringekről:** Stringeket úgy lehet definiálni, hogy:

- `"$xxx"` (Ebben az esetben minden, ami kiértékelhető, ki lesz értékelve)
- `'$xxx'` (Ebben az esetben minden **SZÓ SZERINT** kerül a stringbe, semmi kiértékelés)
- `'alma'` Ez azzal ekvivalens, hogy `$()`

## 2. Okosságok

- korn shebang: `#!/bin/ksh` (utána kötelező üres sor!)
- A változók nem tartalmaznak perjelet!
- printout berakása változóba: `w=$(cd $(dirname $0));pwd`
- függvény szintaxis 1: **function** name { }
- függvény szintaxis 2: **name()**{ } (nem használjuk!!!)
- if szintaxis 1: **if**[ *feltétel* ]; then *command* **fi**
- if szintaxis 2: `[[ "$valami" ]]` (impliciten -n flaget ért alatta a ksh)
- -z/-n a feltételben: nulla/nemnulla a hossz
- a `[[ $VALTOZO ]]`: \$Valtozo hossza nem nulla (ugyanaz, mint: `[[ -n $Valtozo ]]`)
- \$IFS/\$OFS: input és output field separator. Default whitespace, átállítható, de nem ajánlott
- a **read**-del több változóba is be lehet tölteni cuccot!
- &&: logikai és művelet
- ||: logikai vagy művelet
- **file descriptor** 1: standard out
- **file descriptor** 2: standard error
- a **return** utasítás csak numerikus értéket tud visszaadni 0-256-ig
- **std error átirányítása az std outra**: `2 > &1`