

## *Skripty*

Skript = spustitelný soubor obsahující posloupnost příkazů

### **Pár poučení:**

- 1) nepojmenovávej si svůj skript jménem, které je shodné s některým UNIXovským příkazem (např. script, date, time,...)!
- 2) před spuštěním změň přístupová práva (chmod 755 jmeno\_skriptu)
- 3) každý skript začíná 1. řádkem, který spouští shell:  
#!/bin/ksh  
na místě ksh může být jiný název shellu podle aktuálního nastavení. Zjistí se příkazem echo \$SHELL
- 4) řádky začínající # se nevykonávají – jsou to komentáře. Užívejte je hojně, abyste se ve svých skriptech vyznali.

**Př.:** vytvořte skript cas, který vypíše čas.

```
#!/bin/ksh
#####
#Tento skript vypisuje cas a povzbuzuje k praci#
#          vytvorila jej E. Fadrna                #
#          v r. 2007                             #
#####

echo PRESNY CAS
date
echo POKRACUJ V PRACI!
```

## Proměnné

Proměnné = pojmenovaná místa v paměti. Nastavení proměnné:

jmeno=hodnota ↵

### Př.:

```
karel=4716322
... nastavení proměnné karel
Vyzkoušejte si:
echo karel
echo $karel
echo '$karel'
... normální apostrofy
echo ` $karel `
... obrácené apostrofy
echo "$karel"
... uvozovky
```

set ↵

... vypíše všechny dosud definované proměnné, vč. systémových

env ↵

... vypíše proměnné prostředí (nedefinované systémem)

Např. \$HOME, \$PATH, \$TERM, \$USER, \$PS1, ...

export promenna ↵

... export proměnné do shellu potomků

Př.: vytvořte skript mujadresar, který vypíše uživatelské jméno, domovský adresář a jméno terminálu.

```
#!/bin/ksh
#Tento skript vypisuje systemove promenne

echo "Jsem uzivatel: $USER"
echo "Muj domovsky adresar je: $HOME"
echo "Pracuji na terminalu: $TERM"
echo "Tento pocitac se jmenuje: $HOSTNAME"
```

Uložení výsledku příkazu do proměnné: obrácené apostrofy

### Př.:

```
adr=`pwd`
echo $adr
```

... proměnná *adr* nyní obsahuje výsledek příkazu *pwd*

**Př.:**

```
echo "Dnes je: `date`"
```

read

... čte proměnnou v průběhu skriptu

**Př.:** vytvořte skript pozdrav, který vypíše zadaný pozdrav

```
#!/bin/ksh
#Tento skript nacita zadany text do promenne pozdrav
#a pak jej vypise.

echo Zadej text:
read pozdrav
echo Zdravim te: $pozdrav
```

**Př.:** vytvořte skript telefon, který vyhledává v telefonním seznamu (v souboru *seznam.txt*)

```
#!/bin/ksh
#Tento skript hleda zadane jmeno v souboru seznam.txt

#Zadani jmena:
echo Zadej hledane jmeno
read jmeno
#Kontrolni vypis promenne, zda se ulozila spravne
echo Myslis opravdu toto jmeno?
echo $jmeno
#samotne vyhledavani v souboru seznam.txt
grep $jmeno seznam.txt
```

**Př.:** modifikujte předchozí skript tak, aby výsledek příkazu *grep* ukládal do proměnné *cislo* a tu pak vypsál.

**Vnitřní proměnné:**

\$0 ... jméno procedury, tj. skriptu

\$1, \$2 ... poziční parametry...

\$\* ... seznam parametrů kromě jména procedury

\$# ... počet parametrů

**Př.:** skript *test\_parametru* pro výpis vnitřních proměnných a pozičních parametrů.

Spuštění např. takto:

```
test_parametru franta pepa jednicka
```

```
#!/bin/ksh
```

```
#Tento skript vypise sve jmeno a poziční parametry
```

```
echo Moje jmeno, tzn. jmeno skriptu: $0
```

```
echo nacetl jsem $# parametru
```

```
echo jsou to napr. tyto: $1 $2 $3
```

```
echo Druhy parametr je tento: $2
```

**Př.:** napište skript "kopiruj", který zkopíruje soubor, zadaný jako první parametr, do souboru, zadaného jako druhý parametr. Poté ať skript vypíše hlášení: Zkopíroval jsem "soubor1" do "souboru2".

...