

# Zadanie na 1.počítačové cvičenie - skúšobné

April 16, 2018

## 1 Zadanie

Predpokladajme, že súbor `zoznam.txt` má všetky riadky istého tvaru. Tento predpoklad pri načítavaní súboru nemusíte overovať, môžete sa naň spoľahnúť.

Každý riadok je zrefazovaných niekoľko reťazcov tvaru

```
klíč="hodnota"
```

oddelených medzerami.

Dva príklady možných súborov `zoznam.txt`

```
MENO="Bohumel Tomáš" OSC="67720" UID="xbohumel"  
MENO="Brázdilová Veronika" OSC="67674" UID="xbrazdilova"  
MENO="Frolkovič Richard" OSC="69782" UID="xfrolkovicr"  
MENO="Húska Martin" OSC="67644" UID="xhuskam1"  
MENO="Juhász Dávid" OSC="67687" UID="xjuhaszd"  
MENO="Kalina Jakub" OSC="67718" UID="xkalina"  
MENO="Kandová Andrea" OSC="67640" UID="xkandova"
```

```
NUMBER="1" SQUARE="1" CUBE="1"  
NUMBER="2" SQUARE="4" CUBE="8"  
NUMBER="3" SQUARE="9" CUBE="27"  
NUMBER="4" SQUARE="16" CUBE="64"  
NUMBER="5" SQUARE="25" CUBE="125"
```

Predpokladá sa, že každý riadok obsahuje rovnaké kľúče. Tento predpoklad tiež nemusíte overovať pri načítavaní súboru.

Zmeňte váš program tak, aby podporoval tieto nové požiadavky:

KEYS	Pošle klientovi na 1 riadku zoznam všetkých názvov kľúčov vyskytujúcich sa v súbore kdekoľvek. Kľúče sú oddelené čiarkami. Nasledujúci riadok obsahuje status 100 OK.
SELECT <i>zoznam_kľúčov</i>	<i>zoznam_kľúčov</i> je niekoľko kľúčov oddelených čiarkami. Ak <i>zoznam_kľúčov</i> obsahuje neznámy kľúč (ktorý sa v súbore nevyskytuje), vráti status 203 Unknown key a chová sa tak, ako keby tá požiadavka SELECT nebola poslaná. V opačnom prípade táto požiadavka ovplyvňuje nasledujúce požiadavky READ tak, že sa vypisujú z každého riadku len tie kľúče, ktoré boli vybrané príkazom SELECT. Ak klient žiadny príkaz SELECT neposlal, vypisujú pri sa pri príkaze READ vždy všetky kľúče. Ak je všetko v poriadku a v zozname kľúčov sa nevyskytoval žiaden neznámy kľúč, pošle klientovi status 100 OK.
RESET	Vráti pôvodný stav, ako keby klient neposlal žiadny SELECT.

## 2 Vzorové riešenie

Na začiatku cvičenia ho umiestnim na `mpm.svf.stuba.sk`, TCP port 9999.

## 3 Pomôcky pre riešenie

Toto sú moje poznámky vyplývajúce z toho, ako som to robil ja.

- Celý súbor som načítal naraz do pamäti cez `file.readlines`.
- Pre analýzu riadkov som používal funkciu `re.finditer` – vid' dokumentácia Pythonu.
- Súbor som reprezentoval prirodzene ako zoznam slovníkov. To znamená, čo riadok to slovník v zozname.
- Ako som načítaval súbor a postupne prevádzal jeho riadky na slovníky, ukladal som si všetky kľúče na ktoré som narazil do zoznamu kľúčov. Ten sa potom vypisoval v žiadanom tvare ako odpoveď na KEYS.
- SELECT prejde svoj argument (Vid' `str.split`) a po kontrole, či sú všetky kľúče známe ich uloží.

- Spracovanie READ sa zmenilo. Namiesto prostého poslania riadku klientovi sa riadok generuje zo slovníka.

## 4 Úlomky kódu obsahujúce užitočné informácie

### 4.1 Ako sa používa re.finditer

```
>>> import re
>>> line='x=y u=1 t=9 \n'
>>> pattern=r'(.)=(.)'
>>> for m in re.finditer(pattern,line):
...     print "L=%s R=%s" % (m.group(1),m.group(2))
...
L=x R=y
L=u R=1
L=t R=9
```

### 4.2 Ako sa naplňajú slovníky a ako sa cez ne iteruje

```
>>> d={}
>>> d["x"]="y"
>>> d["z"]="woot"
>>> print d["x"]
y
>>> print d["z"]
woot
>>> for k in d:
...     print k,d[k]
...
x y
z woot
```

### 4.3 Ako sa používa str.split

```
>>> s="nejake,veci,oddelene,ciarkami"  
>>> print s.split(",")  
['nejake', 'veci', 'oddelene', 'ciarkami']  
>>> for vec in s.split(","):  
...   print "spracuvam",vec  
...  
spracuvam nejake  
spracuvam veci  
spracuvam oddelene  
spracuvam ciarkami
```