Lekce 4

Kolekce

Kolekce - seznamy, N-tice a slovníky.

C2184 Úvod do programování v Pythonu
podzim 2016

Stanislav Geidl
Národní centrum pro výzkum biomolekul
Masarykova univerzita

Kolekce

- seznam (list)
- N-tice (tuple)
- slovník (dictonary, dict)

List (seznam)

- patří do kolecí, podobně jako N-tice a slovník
- vytváříme pomocí hranatých závorek `[]`
  `["a", "b", "c", "d"]`
- každý prvek má svůj automatický index, který odpovídá pořadí

Práce se seznamy I.

- vytvoření

```
seznam1 = [1, 1, 2, 3, 5, 8, 13]
seznam2 = list(seznam1)
seznam3 = seznam1[:]
seznam4 = seznam1 # nejedna se o novy list,
pouze odkaz na stary!!!
seznam5 = range(2,20,2)
# [2, 4, 6, 8, 10, 12, 14, 16, 18]
```

- přidáváme prvky

```
seznam1.append(21)
# [1, 1, 2, 3, 5, 8, 13, 21]
seznam2.insert(2, 90)
# [1, 1, 90, 2, 3, 5, 8, 13]
seznam3.extend([21, 34])
# [1, 1, 2, 3, 5, 8, 13, 21, 34]
seznam3.append([21, 34])
# [1, 1, 2, 3, 5, 8, 13, [21, 34]]
```


Práce se seznamy II. - přístup k hodnotám

- můžeme přistupovat k jakémukoliv prvku pomocí jeho indexu
  `seznam[x]`, kde nula a kladné číslo n určuje index zleva a zaporné číslo určuje index zprava

```
[1, 2, 3, 4, 5][0] # 1
[1, 2, 3, 4, 5][1] # 2
[1, 2, 3, 4, 5][-1] # 5
```

- přes dvojtečku můžeme nadefinovat rozsah
  `seznam[x:y]`, kde tyto výrazy si odpovídají:

```
[1, 2, 3, 4, 5][:] # [1, 2, 3, 4, 5]
[1, 2, 3, 4, 5][2:] # [3, 4, 5]
[1, 2, 3, 4, 5][:2] # [1, 2]
```

- pozor na číslovaní! v Pythonu začínáme od nuly!

```
[1, 2, 3, 4, 5]
 0. 1. 2. 3. 4.
```

- co bude výsledkem?

```
seznam = [1, 2, 4, 5, 6]
```

```
x[1:4]          x[2:]          x[:2]
x[2:2]          x[-2:]         x[:-2]
```


Práce se seznamy III.

```
seznam1 = ['a', 'b', 'c', 'd', 'e', 'f']
```

- mazání

```
seznam1.remove('c')
# ['a', 'b', 'd', 'e', 'f']
last = seznam1.pop()
# last = 'f'; ['a', 'b', 'd', 'e']
first = seznam1.pop(0)
# last = 'a'; ['b', 'd', 'e', 'f']
```

- přehození směru

```
seznam1.reverse()
# ['f', 'e', 'd', 'b']
```

- vyhledávání

```
seznam1.index('b') # 3 = 4. prvek
seznam1.count('d') # 1 = jedenkrat
```

- seřazení

```
seznam = [1, 4, 3, 6, 2, 5]
seznam.sort() # [1, 2, 3, 4, 5, 6]
seznam.sort(reverse=True) # [6, 5, ...
```


Práce se seznamy IV.

- počítání

```
seznam1 = [1, 1, 2, 3, 5, 8, 13]
len(seznam1) # 7
sum(seznam1) # 33
min(seznam1) # 1
max(seznam1) # 13
```

- procházení

```
for item in [1, 2, 3]:
  print(item)
for item in range(1,4):
  print(item)
```


Tuple (N-tice)

- vytváříme pomocí jednoduchých závorek `()`
- můžeme s nimi pracovat podobně jako se seznamy, jenom je nemužeme měnit, tzn. že funkce `append` a další nejsou dostupné
- můžeme jednoduše převádět na list pomocí `list((1,2))` a podobně zpět `tuple([1,2])`

Dictionary (slovník)

- vytváříme pomocí složených závorek `{}`
  `{1: 3, 2: 4}`
- prvek ve slovníku se skládá z klíče a jeho hodnoty, 1 a 2 jsou klíče, jejich hodnoty jsou 3, resp. 4
- nefungují zde indexy, na hodnoty se dotazujeme pomocí klíče
- každý klíč je unikatní, žádný slovník nemůže obsahovat dva stejné klíče

Práce se slovníky

- vytvoření
```
dict = {'Name': 'Zara', 'Age': 7, 'Class':
'First'}
```
- čtení/získání
```
print(dict['Name']) # Zara
```
- přidání nebo úprava hodnot
```
dict['Age'] = 8 # úprava stávající hodnoty
dict['School'] = "DPS School"# přidání nové
```
- smazání hodnot
```
del dict['Name']
dict.clear() # smaže všechny položky
del dict # smaže celý slovník
```
- procházení hodnot
```
for key in dict:
  print(key)
  print(dict[key])
```


Vícerozměrné kolekce

- kolekce můžeme kombinovat a vytvářet list listů, ...
  `[[1, 2], [2, 3], [4, 5]]`
- můžeme kombinovat i navzájem a vytvářet list N-tic, ...
  `[(1, 2), (2, 3), (4, 5)]`

Příklad

Mějme tento citát:

```
citat = """Hra je jeden z nejefektivnejsich zpusobu,
jak zjednodusit zivot.Presne to jsme delali jako deti,
ale v dospelosti jsme si hrat zapomneli."""
```

1. Převeďte citát na seznam slov (zalomení řádků, '.' a ',' jako součást slov nepovažujte)
2. Počítejte a hledejte:
   a) Kolik citát obsahuje slov?
   b) Kolik je minimální a maximální počet znaků ve slovech?
   c) Jaké je nejdelší slovo?
   d) Jaká je průměrná délka slova?
   e) Jaký je medián délky slova?

Příklad - řešení

```
citat = """Hra je jeden z nejefektivnejsich zpusobu,
jak zjednodusit zivot.Presne to jsme delali jako deti,
ale v dospelosti jsme si hrat zapomneli."""

""" 1 """
citat2 = citat.replace('.','')
citat2 = citat2.replace(',','')
slova = citat2.split()
""" nebo slova = citat.replace('.','').replace('.','').split()"""

""" 2 """
""" a - kolik je slov? """
pocet = len(slova)
print(pocet)
""" b - kolik je minimalni/maximalni pocet znaku ve slovech? """
pocty = []
for slovo in slova:
    pocty.append(len(slovo))
minimum = min(pocty)
maximum = max(pocty)
print("Min: {}\nMax: {}".format(minimum,maximum))
""" c - jake slovo je nejdelsi """
i = pocty.index(maximum)
print(slova[i])
""" d - jaka je prumerna delka slov """
print(sum(pocty)/pocet)
```


Příklad - řešení (pokračování)

```
""" e - jaky je median delky slov """
pocty2 = pocty[:]
pocty2.sort()
i = int(pocet/2)
if pocet \% 2 == 0:
    print((pocty2[i-1]+pocty2[i])/2)
    """ nebo print(sum(pocty2[i-1:i+1])/2) """
else:
    print(pocty2[i])
del pocty2
```