

Python – lekce 8

Seznamy

Existuje proměnná, které Python říká seznam. Jiné programovací jazyky ji nazývají "pole". Python zvolil seznam snad proto, že takový název intuitivněji odpovídá více nějakému seznamu (třeba telefonnímu) než slovíčko pole. Seznamy si děláte, když chodíte nakupovat, když si píšete rozvrh nebo úkoly. Tak přesně o tom je seznam v Pythonu.

- uvádějí se v [], jednotlivé hodnoty se oddělují čárkou
- mohou obsahovat libovolná data
- jednotlivé prvky seznamu jsou indexovány (stejně, jako znaky v řetězci)

```
seznam = []                #prázdný seznam
print(seznam)

seznam = [3, "Python", 3.0]
print(seznam)

print(seznam[0])          #práce s jedním prvkem pomocí indexu

seznam[2]="Vánoce"        #přiřazení
print(seznam)

seznam = seznam + seznam  #sčítání seznamů
print(seznam)

seznam = seznam * 2       #násobení seznamu konstantou
print(seznam)

print(seznam[:4])         #podseznam [0 .. 3]
print(seznam[4:])         #podseznam [4 .. n]
print(seznam [1:4])       #podseznam [1 .. 3]
```

Operace se seznamy

append (hodnota)	- přidá hodnotu na konec seznamu
count (hodnota)	- zjistí počet výskytů dané hodnoty v seznamu
insert (index, hodnota)	- vloží hodnotu na pozici v seznamu
pop(index)	- odebere prvek z dané pozice, není-li index uveden, odebere z konce seznamu
remove (hodnota)	- odebere první výskyt této hodnoty
sort ()	- seřadí seznam podle velikosti
len (seznam)	- kolik je položek v seznamu
reverse()	- otočení (přehození) položek seznamu

```
ovoce=['jablka', 'hrusky', 'tresne']
print(ovoce)

ovoce.append('maliny')      # přidání do seznamu
```

```

ovoce=ovoce+['visne']          # jiné přidání do seznamu
print(ovoce)

ovoce.insert(50,"broskve")      # vloží hod. na konec
print(ovoce)

ovoce.insert(0,"merunky")       # vloží hod. na pozici 0
print(ovoce)

ovoce.pop(1)                    # smazání druhé položky: jablka
print(ovoce)

ovoce.pop()                     # pokud neuvedeme index, odebírá z konce seznamu
print(ovoce)

kolik=ovoce.count("hrušky")     # zjistí počet výskytů dané položky
print(kolik)

ovoce.remove("visne")           # odebere první výskyt dané položky
print(ovoce)

# pokud daná položka v seznamu není, nahlásí nám program chybu, proto je dobré při
# odebírání tuto situaci ošetřit:

if "višně" in seznam:
    ovoce.remove("višně")

ovoce.sort()                    # seřadí seznam
print(ovoce)

ovoce.reverse()                 # obrátí seznam
print(ovoce)

```

Procházení seznamem

```

ovoce=["jablka","hrušky","třešně"]
for jedno in ovoce:
    print ("Moje oblíbené ovoce je", jedno)
    print( "V seznamu je na", ovoce.index(jedno),"místě.\n")
    #případně          str(ovoce.index(jedno)+1)

                                # složitější, ale někdy použitelnější jiné řešení
                                # procházení přes indexy, přes pořadí položek

for poradi in range(len(ovoce)):
    print( "Moje oblíbené ovoce je", ovoce[poradi])
    print("V seznamu je na", poradi,"místě.\n")

```

Vzorový příklad

1. Napište cyklus, který bude načítat z klávesnice 5 čísel a postupně je bude dávat na konec prázdného seznamu. Seznam nakonec vypíše.

```
seznam = []                                #nadefinování prázdného seznamu
for i in range(5):                         #cyklus pro načítání hodnoty
    x = int(input("Zadej číslo do seznamu: ")) #načtení hodnoty z klávesnice
    seznam.append(x)                       #přidání hodnoty do seznamu
print (seznam)                            # výpis čísel v seznamu
```

2. Z tohoto seznamu vytvořte nový, který bude obsahovat dvojnásobné hodnoty načtených čísel

```
seznam2 = []
for i in seznam:                           # cyklus pro procházení seznamem
    seznam2.append (i*2)
print (seznam2)
```

3. Z prvního seznamu vypíše pouze ta čísla, která jsou kladná

```
seznam3 = []
for i in seznam:
    if i > 0:
        seznam3.append(i)
print( seznam3)
```