

# Seznamy o operace se seznamy

## Seznam

## Operace se seznamy

## Metody seznamů

### Seznam

Seznam v jazyku Python je nejobecnější posloupností. Seznamy mohou obsahovat libovolné datové typy (čísla, řetězce, další seznamy). Seznamy:

- Jsou posloupnostmi objektů – jsou to místa (kolekce), kam můžeme umísťovat další prvky. Jsou to kolekce s definovaným pořadím prvků – posloupnosti.
- Umožňují přístup k prvkům přes pořadí – patří mezi posloupnosti, takže mají definované pořadí prvků.
- Mohou být heterogenní, vnořené a mohou mít proměnlivou délku.
- Jsou nekonstantními posloupnostmi, takže můžeme volně přidávat a odebírat prvky.
- Jsou referencí na objekty.

### Operace se seznamy

Zápis konstanty či operace	Význam
<code>L1 = []</code>	Prázdný seznam
<code>L2 = [0, -1, 3, 5]</code>	Seznam o čtyřech prvcích
<code>L3 = ['abc', 3, [2,4, 'def']]</code>	Vnořený seznam
<code>L3[1]</code>	Výběr prvku
<code>L2[1:2]</code>	Výběr části
<code>L1 + L2, L2 * 3</code>	Skládání, opakování
<code>for x in L2:, 3 in L2</code>	Procházení prvky, je 3 prvkem L2
<code>len(L2)</code>	Počet prvků
<code>del L2[2], del L3[0:2]</code>	Vymazání prvku, části
<code>L2[1] = -1, L2[1:3] = [2, -1]</code>	Přiřazení prvku, části
<code>Range(4)</code>	Vrací seznam, obsahující po sobě jdoucí čísla z daného intervalu
<code>Min(s)</code>	Minimální hodnota v seznamu s
<code>Max(s)</code>	Maximální hodnota v seznamu s

Pozn. Obdobou operace `range(j)` je operace `xrange(j)`, která vrací n-tici po sobě jdoucích čísel z intervalu.

**Příklady (výstup z počítače je psán tučně):**

délka seznamu:

```
len([1, 0, -1, 2])
```

**4**

sloučení dvou seznamů:

```
[1, 2] + [0, 4, -4]
```

**[1, 2, 0, 4, -4]**

opakování::

```
['abc', 2] * 3
```

```
['abc', 2, 'abc', 2, 'abc',  
2]
```

výpis prvků seznamu:

```
for x in [0, 2, -2, 4]:  
print x,
```

**0 2 -2 4**

výpis části seznamu:

```
L1 = [0, 1, 2, -2, 5]
```

```
L1[2:5]
```

```
[2, -2, 5]
výpis prvku seznamu:
L1[-3]
2
změna prvku seznamu:
L1[1] = -2
```

```
L1
[0, -2, 2, -2, 5]
vymazání prvku ze seznamu:
del L1[2]
L1
[-2, -2, 2, 5, 'abc']
```

## Metody seznamů

Metoda je funkce, která se určitým způsobem váže na určitý objekt, při svém zavolání modifikuje pouze tento objekt:

`s` v metodě je seznam.

Metoda	Význam
<code>list(s)</code>	Konvertuje posloupnost <code>s</code> na seznam
<code>s.append(x)</code>	Přidá prvek na konec seznamu. Tato metoda je ekvivalentní zápisu <code>s[len(a):] = [x]</code>
<code>s.extend(L)</code>	Na konec seznamu <code>s</code> přidá všechny prvky seznamu <code>L</code> ; je ekvivalentní zápisu <code>s[len(a):] = L</code>
<code>s.insert(i, x)</code>	Vloží prvek <code>x</code> na pozici <code>i</code> . Argument <code>i</code> znamená index prvku, před který se má nová položka vložit, tudíž <code>s.insert(0, x)</code> vloží prvek na začátek seznamu, zatímco <code>s.insert(len(a), x)</code> na konec (jde o totéž jako <code>s.append(x)</code> )
<code>s.remove(x)</code>	Ze seznamu odstraní daný prvek <code>x</code> , pokud prvků rovných <code>x</code> se v seznamu nachází více, odstraní se první jeho výskyt. Nenajde-li metoda prvek <code>x</code> , dojde k výjimce.
<code>s.pop([i])</code>	Odstraní prvek na pozici <code>i</code> a vrátí jeho hodnotu. Argument <code>i</code> je nepovinný, jeho vynecháním dojde k odstranění posledního prvku seznamu.
<code>s.index(x)</code>	Vrátí index prvního prvku seznamu, jehož hodnota je rovna <code>x</code> . Není-li prvek nalezen, dojde k výjimce.
<code>s.count(x)</code>	Vrátí počet všech výskytů prvků, jejichž hodnota je rovna <code>x</code> . Není-li nalezen žádný prvek, vrátí nulu.
<code>s.sort()</code>	Seřadí prvky seznamu podle velikosti, přičemž modifikuje původní seznam.
<code>s.reverse()</code>	Obrací seznam -- první prvek se stane posledním, druhý předposledním atd. Změna se děje opět na původním seznamu.

Následují **příklady** ukazující použití jednotlivých metod seznamu:

vytvoření seznamu z řetězce:

```
list('avdb')
```

```
['a', 'v', 'd', 'b']
```

*počet prvků v seznamu*

```
a = [66.6, 333, 333, 1, 1234.5]
print a.count(333), a.count(66.6), a.count('x')
2 1 0
```

*počet vložení prvku a připojení prvku s seznamu*

```
a.insert(2, -1)
a.append(333)
a
[66.6, 333, -1, 333, 1, 1234.5, 333]
```

*index prvku*

```
a.index(333)
1
```

*vyhledání a odstranění prvku*

```
a.remove(333)
a
[66.6, -1, 333, 1, 1234.5, 333]
```

*obrácení prvků v seznamu*

```
a.reverse()
a
[333, 1234.5, 1, 333, -1, 66.6]
```

*Setřídění prvků v seznamu*

```
a.sort()
a
[-1, 1, 66.6, 333, 333, 1234.5]
```

### Literatura:

- [1] Rubeš, J.: Nebojte se programovat, ComputerMedia, Bedihošť 2001
- [2] Lutz, M., Ascher, D.: Naučte se Python, Grada, Praha 2003
- [3] Beazley, D. M.: Python, Neocortex, Praha 2002
- [4] Python Reference Manual
- [5] Švec, J.: Létající cirkus, Python tutoriál, 2003
- [6] Švec, J.: Učebnice jazyka Python (aneb Létající cirkus), py.cz, release 2.2, 2002