



Листе

Поред појединачних вредности исказаних бројем или ниском карактера, често је потребно забележити већи скуп вредности које су на неки начин повезане, као, на пример, имена у списку путника у неком авиону. Један могући начин би био да уведеш пуно засебних променљивих:

```
1 putnik1 = "Петар Петровић"
2 putnik2 = "Ана Анић"
3 putnik3 = "Мика Микић"
4 putnik4 = "Јелена Јеленковић"
5 putnik5 = "Зорана Зорановић"
6 putnik6 = "Милан Милановић"
```

Међутим, овакво представљање података није zgodно, јер не можеш једноставно да одговориш на разна интересантна питања о путницима (на пример, колико укупно има путника, ко седи на месту чији се број уноси са улаза, на ком месту седи неки дати путник и слично). О томе какве могућности нуди Python када су у питању овакви захтеви говорићемо на овом часу.

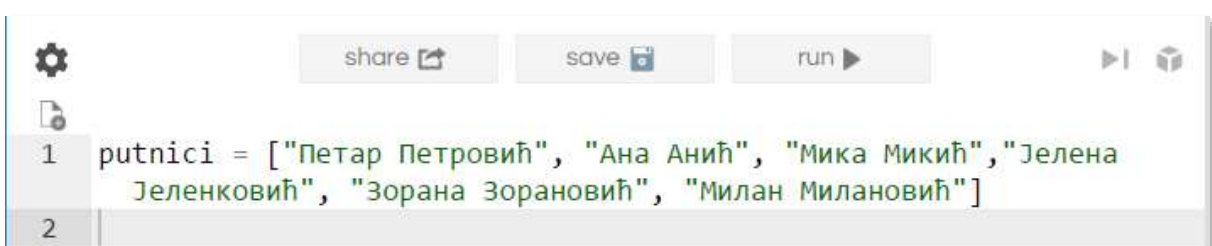
За почетак одгледај следећу видео-лекцију



Python нуди могућност да читав списак буде представљен као посебна врста колекције названа **листа**. Појединачни чланови листе називају се **елементи**. Елементи листе могу бити ниске (као у овом примеру), али и бројеви, друге листе и било шта друго.

Могуће је у исту листу поставити елементе различитог типа.

Листу дефинишеш тако што између угластих заграда (заграда [и]) наводиш елементе раздвојене зарезима. Листу путника сада можеш дефинисати и овако:





Ниске и листе имају доста заједничког, па важи следеће:

- Елементима листе приступа се на основу њихове позиције тј. **индекса**
- Бројање креће од нуле, па је првом елементу листе могуће приступити са `putnici[0]`.
- Негативни индекси упућују на бројање од краја (с десна на лево), па је последњем елементу могуће приступити помоћу `putnici[-1]`.
- Елементима између позиција `a` и `b` укључујући позицију `a`, али искључујући позицију `b` могуће је приступити помоћу `putnici[a:b]`.

Задатак 1.

Познат је списак имена путника у авиону. Седишта су нумерисана од 1 па надаље. Ако стјуардеса унесе број седишта, напиши програм који одређује име путника на том седишту.

Пошто се уносе бројеви од 1 до дужине листе, а индекси се крећу од 0 па до претходника дужине листе, након учитавања броја седишта приступиће се елементу листе чији је индекс за 1 мањи од броја седишта.

Предлог решења

```
1 putnici = ["Петар Петровић", "Ана Анић", "Мика Микић", "Јелена  
2 Јеленковић", "Зорана Зорановић", "Милан Милановић"]  
3 sediste = int(input("Број седишта: "))  
4 print(putnici[sediste - 1])
```

Овако написан програм би могао да да следећи испис при покретању:

```
Python 3.6.1 (default, Dec 2015, 13:05:11)  
[GCC 4.8.2] on linux  
Број седишта: 4  
Јелена Јеленковић
```

**Задатак 2.**

Испиши имена путника са бројевима седишта од 2, 3 и 4, као и име путника који седи на последњем седишту.

Када би се места у авиону бројала од 0, као што се броје индекси листе, потребно би било издвојити елементе у распону [2:5] (то обухвата тачно индексе 2, 3 и 4). Пошто се места броје од 1, потребно је вредности наведене у распону умањити за 1 тј. употребити распон [1:4].

Предлог решења

```
1 putnici = ["Петар Петровић", "Ана Анић", "Мика Микић", "Јелена  
2 Јеленковић", "Зорана Зорановић", "Милан Милановић"]  
3 print(putnici[1:4])  
4 print(putnici[-1])
```

Претрага листе

Python нуди могућност одређивања индекса (позиције) у листи на којој се налази тражени елемент. Један начин да то урадиш јесте да употребиш функцију **index** која враћа индекс првог појављивања траженог елемента. Уколико траженог елемента у листи нема, добићеш обавештење о грешци када се програм буде извршавао.

Задатак 3.

На ком броју седишта седи Мика Микић?

Предлог решења

```
1 putnici = ["Петар Петровић", "Ана Анић", "Мика Микић", "Јелена  
2 Јеленковић", "Зорана Зорановић", "Милан Милановић"]  
3 print(putnici.index("Мика Микић"))
```



Функције за рад са листама

За рад са листама Python нуди функције које ти могу бити од велике помоћи. Неке од њих су:

- Функција **len** израчунава дужину листе,
- Функција **sum** израчунава збир елемената листе,
- Функција **min** одређује вредност најмањег елемента у листи,
- Функција **max** одређује вредност највећег елемента у листи.

Задатак 4.

Ако су дате Маркове оцене из неколико предмета, напиши програм који ће израчунати његову просечну оцену.

За одређивање просечне оцене су ти потребне сума њихова сума и број. За те вредности употреби функције `sum` и `len`.

Предлог решења

```
1 ocene = [5, 4, 5, 3, 5]
2 prosek = sum(ocene) / len(ocene)
3 print(prosek)
4
```

Задатак 5.

Време се често мења и дешава се да се у једној недељи смењују и хладни и топли дани. Ако су дате су температуре у подне у данима током једне недеље, одреди колики је распон температура тј. разлика између највише и најниже подневне температуре.

Предлог решења

```
1 temperature = [17, 23, 12, 15, 19, 21, 25]
2 print(max(temperature) - min(temperature))
3
```



Сортирање листе

У раду са листама је често потребно елементе листе уредити, тј. **сортирати**. У језику Python уређивање листе од најмањег до највећег елемента, а и у обрнутом редоследу, је врло једноставно. Сортирање листе се изводи употребом функције **sorted**.

```
1 lista = [17, 23, 12, 15, 19, 21, 25]
2 sortirana_lista = sorted(lista)
3 print(sortirana_lista)
4
```

Функција се може применити и на листу која садржи ниске, при чему се ниске уређују на начин на који се уређују у речницима, према алфабетском распореду (упоредиви су и специјални карактери, али о томе ћеш учити касније).

Задатак 6.

Наставница треба да у дневник унесе имена ученика, међутим, од педагога је добила списак ученика који није сортиран. Напиши програм који помаже наставници да добије ученике сортиране по абecedном реду.

Предлог решења

```
1 ucenici = ["Ljubić Milenko", "Stojković Milica", "Vilimonović
2           Aleksandar", \
3           "Jokić Đurađ", "Filipović Kalina", "Zlatković Jasmina"]
4 print(sorted(ucenici))
```

Примети да је листа овом решењу била дуга, па је било једноставније и прегледније да буде записан кроз више редова. Да би нагласио да се нека наредба наставља и у следећој линији на крај линије постави симбол \.

Задатак 7.

Дата је листа цена производа. Колико коштају три најјефтинија, а колико три најскупља производа?

Предлог решења

```
1 cene = [58.00, 104.95, 117.50, 11.95, 10.4, 37.95, 85.5]
2 sortirane_cene = sorted(cene)
3 print(sum(sortirane_cene[0:3]))
4 print(sum(sortirane_cene[-3:]))
5
```



Задатак 8.

Дате су цене три производа. Ко купи сва три, најјефтинији ће добити за динар. Колико ће платити?

Покушај да решиш овај задатак. Задатак можеш пронаћи и проверити сопствено решење у окружењу [интерактивног уџбеника](#).