



## Наредбе понављања

Како би на основу онога што си до сада научио написао програм у којем се десет пута извршава наредба `print("Zdravo")`. То би могло да изгледа овако

```
1 print("Zdravo")
2 print("Zdravo")
3 print("Zdravo")
4 print("Zdravo")
5 print("Zdravo")
6 print("Zdravo")
7 print("Zdravo")
8 print("Zdravo")
9 print("Zdravo")
10 print("Zdravo")
11
```

Овај програм било напорно откуцати (чак иако смо користили технику копирања и лепљења). Ако сада замислимо да желимо да корисник прво унесе број колико пута жели да се испише поздрав, тада је јасно да тај програм не можеш испрограмирати на овај начин. Решење долази у облику **петљи**, тј. **наредби које омогућавају да се нека наредба или неки блок наредби понове више пута**. Тема овог часа су управо петље. На почетку одгледај следећу видео-лекцију



### Python – понављање

Као што си могао да видиш у видео-лекцији, начин да у програму нешто (на пример, команду `print("Zdravo")`) поновиш дати број пута (на пример, 10) јесте да употребиш наредни облик програма:

```
1 for i in range(10):      # ponovi 10 puta:
2     print("Zdravo")      #   ispiši tekst "Zdravo"
3 |
```

Обрати пажњу, наредба која се понавља је, слично као и код команде `if`, увучена чиме се наглашава да је она написана у склопу `for` команде и да ће бити извршена дати број пута.



Дакле, команда која се користи за поновљено извршавање команде `print` јесте наредба **for**. Облик команде је следећи

```
for i in range(10):
```

Komanda

где се **i** назива **бројачем** који узима вредности из интервала од 0 до, у овом случају 9, који је дефинисан командом `range(10)`. Да би се Komanda поновила 50 пута потребно је у функцији `range` уместо 10 навести 50.

Наредба `for` заједно са командама које се у њеном склопу понављају чине такозвану **петљу**. Један пролаз кроз петљу чини једно извршавање понављајућих команди.

### Задатак 1.

Напиши програм који израчунава обиме пет једнакостраничних троуглова чије дужине страница корисник уноси.

Да би се израчунао и исписао обим једног троугла потребно је у програму навести команду за учитавање величине странице, затим израчунати обим и на крају исписати добијени обим. Како су све три потребне да би обавио задатак за један троугао, све три се морају наћи унутар наредбе `for`. Дакле, програм би могао да изгледа овако:

```
1 for i in range(5):                                # ponovi 5 puta:
2     a = int(input("Unesi dužinu stranice:"))        # učitaju dužinu stranice
3     o = 3 * a                                        # izračunaj obim
4     print(o)                                        # ispiši obim
5
```

**Задатак 2.**

Напиши програм који учитава неколико имена ученика и сваког појединачно поздравља.

Новина у овом задатку није само другачији скуп команди већ и то што се не зна унапред колико ће се пута команде понављати, већ потребно број понављања учитати.

**Предлог решења**

```
1 n = int(input("Unesi broj učenika: "))      # učitaj broj učenika n
2 for i in range(n):                          # ponovi n puta:
3     ime = input("Kako se zoveš: ")          # unesi ime učenika
4     print("Zdravo, ti se zoveš:", ime)      # pozdravi učenika
```

**Бројачке променљиве**

Откуцај следећи програм и покрени га.

```
1 for i in range(10):
2     print(i)
```

Резултат извршавања овог програма јесте испис бројева од 0 до 9. Разлог томе јесте начин на који се for команда извршава. Наиме,

`i in range(10)`

значи да ће при сваком, од 10, пролаза кроз петљу, бројач `i` узима вредности из интервала од 0 до 9, при чему у првом пролазу узима вредност 0, другом 1, трећем 2 итд.

Команда `range()` може да има и мало другачији. Наиме, уколико би написали овакву команду

`for i in range(4,10):`  
 `print(i)`

ће резултирати исписом броје почевши од 4, па до 9.

**Задатак 2.**

Напиши програм који исписује све бројеве почевши од вредности првог уčitаног броја до вредности другог уčitаног броја.

**Предлог решења**

```
1 a = int(input("Odakle se broji: "))
2 b = int(input("Dokle se broji: "))
3 for i in range(a, b+1):
4     print(i)
```

**Пролазак кроз листу**

Команду `for` се можеш користити и за пролазак кроз листу. Уместо проласка кроз лист целих бројеве креирану командом `range()`, можеш се користити листу произвољних чланова. Понављања команди у петљи ће бити онлико колико има чланова листе. На пример, петља

```
for i in [3, 5, 9]:
    print(i)
```

ће имати три пролаза. На излазу ће бити исписани бројеви 3, 5 и 9, јер су то вредности које ће и узети у првом, другом и трећем пролазу кроз петљу.

**Задатак 3.**

Дата је листа у којој се налази списак неколико оперативних система. Исписи све елементе ове листе (сваки у посебној линији).

**Предлог решења**

```
1 operativni_sistemi = ["windows", "linux", "android", "ios"]
2 for sistem in operativni_sistemi:
3     print(sistem)
```

Овај задатак можеш решити и на други начин, тако шт ћеш употребити бројачку променљиву и елементима листе приступати на основу индекса. На тај начин уз сваки систем можеш исписати и његов редни број. Бројачка променљива узима редом вредности од 0, па све до броја који је за један мањи од дужине листе. Ако редни бројеви иду од 1 па до дужине листе, редни број је у сваком кораку за један већи од вредности бројачке променљиве.



Решењи би у том случају могл да изгледа овако:

```
1 operativni_sistemi = ["windows", "linux", "android", "ios"]
2 for i in range(len(operativni_sistemi)):
3     print(i + 1, operativni_sistemi[i])
```

## Пролазак кроз листу парова

У петљи можеш проћи и кроз све елементе листе парова. Покушај да решиш наредни задатак.

### Задатак 4.

Ако је дата листа која садржи парове имена и презимена неколико научника, напиши програм који прави њихов списак.

### Предлог решења

Један начин решавања овог задатка јесте да прођеш кроз листу парова као кроз било коју другу листу тј. да уведемо променљиву која ће имати редом вредности једног по једног уређеног пара. Појединачним елементима пара (имену и презимену), онда можеш да приступити на основу индекса у оквиру пара.

```
1 naucnici = [("Nils", "Bor"), ("Čarls", "Darvin"), ("Isak", "Njutn"),
2             ("Marija", "Kiri")]
3 for naucnik in naucnici:
4     print(naucnik[0], naucnik[1])
```

Слично, можеш употребити и бројачку променљиву која ће пролазити кроз индексе листе и затим сваком пару приступати на основу његовог индекса у листи, а онда сваком елементу пара (имену и презимену) приступати на основу његовог индекса у пару.

```
1 naucnici = [("Nils", "Bor"), ("Čarls", "Darvin"), ("Isak", "Njutn"),
2             ("Marija", "Kiri")]
3 for i in range(len(naucnici)):
4     naucnik = naucnici[i]
5     print(naucnik[0], naucnik[1])
```

Постоји још један начин проласка кроз листу парова, који у појединим ситуацијама може бити од велике помоћи.



```
1 naucnici = [("Nils", "Bor"), ("Čarls", "Darvin"), ("Isak", "Njutn"),  
2             ("Marija", "Kiri")]  
3 for (ime, prezime) in naucnici:  
4     print(ime, prezime)
```

У овом сличају променљиве `ime` и `prezime` редом узимају имена и презимена научника.

## Пролазак кроз речник

У петљи можеш проћи и кроз речник. листе парова. Нека Покушај да решиш наредни задатак

### Задатак 5.

Нека је дат речник који садржи називе неколико земаља и њихове површине. Испиши називе и површине свих земаља.

### Предлог решења

```
1 površine = {"Srbija": 88361,  
2             "Hrvatska": 56594,  
3             "Crna Gora": 13812,  
4             "Bosna i Hercegovina": 51197,  
5             "Slovenija": 20273,  
6             "Makedonija": 25713}  
7 for zemlja in površine:  
8     print("Naziv: ", zemlja, "Površina: ", površine[zemlja])  
9
```

У овом решењу променљива `zemlja` редом узима вредности кључева у датом речнику, тј. назива земаља.

Још један користан начин да се прође кроз елементе речника је да се употреби метода `items()` кроз чији се резултат може проћи на исти начин којим се пролази кроз све елементе листе парова.

```
1 površine = {"Srbija": 88361,  
2             "Hrvatska": 56594,  
3             "Crna Gora": 13812,  
4             "Bosna i Hercegovina": 51197,  
5             "Slovenija": 20273,  
6             "Makedonija": 25713}  
7 for (zemlja, površina) in površine.items():  
8     print("Naziv: ", zemlja, "Površina: ", površina)  
9
```