

Допремање већег броја аргумената

Често је корисно писати функције које могу примити више аргумената истовремено.

**063 Допремање више аргумената**

Пример 36: употребити више аргумената приликом позива функције

```
def prikaz_sume(broj1, broj2):
    rezultat = broj1 + broj2
    print(rezultat)

def main():
    print('Suma 12 i 45 je')
    prikaz_sume(12, 45)
```

```
main()
```

Види се да се користе две параметра, broj1 и broj2, унутар хедера функције prikaz\_sume раздвојени зарезом.

Често се сви наведени параметри у загради хедера функције називају листа параметара.

У функцији main() се позива функција prikaz\_sume() са два аргумента 12 и 45.

Ови аргументи се достављају према позицији у одговарајуће параметре у функцији (12 у broj1 а 45 у broj2).

Уместо конкретних вредности, аргументи могу бити промењиве које морају упућивати на конкретне вредности приликом позивања функције.

Промена у параметрима

Када се аргумент достави функцији, параметар промењива функције ће указивати (reference) на вредност аргумента.

Ипак, било каква промена која се изведе над параметар промењивој неће утицати на аргумент.

**064 Измена вредности параметара**

```
def main():
    x = 99
    print('Vrednost je', x)
    promeni_me(x)
    print('U main() funkciji vrednost je i dalje', x)

def promeni_me(y):
    print('Ovde se menja vrednost.')
    y = 0
    print('Sada je vrednost', y)
```

```
main()
```

У коду, промењива x је достављена као аргумент функцији promeni\_me, а то значи да ће параметар у такође указивати на вредност 99.

Промена у мења вредност у 0 и тиме се не мења вредност промене х у функцији main(). Када се нека вредност достави функцији путем аргумента а функција не може променити ту вредност аргумента се назива достављање преко вредности (pass by value). Овај начин комуникације између функција је једносмеран, тј само од прве функције ка позваној функцији.

### Аргументи службене речи

Осим преноса вредности од аргумента према параметру по позицији у позиву функцији, Пајтон омогућава и писање аргумента са службеном речи да би се специфицирало коју вредност аргумент треба да преда параметру:

```
ime_parametra = vrednost
```

Аргумент који је написан у складу са овом синтаксом се назива аргумент службене речи (keyword argument).

### **065 Употреба аргумента службене речи**

```
def main():
    show_interest(korak=0.01, vremenski_period=10, osnovica=10000.0)

def show_interest(osnovica, korak, vremenski_period):
    ulog = osnovica * korak * vremenski_period
    print('Ulog ce biti $', format(ulog, ',.2f'), sep='')

main()
Ulog ce biti $1,000.00
```

Види се да редослед аргумената службених речи не одговара редоследу параметара у хедеру функције.

То значи да се аргументи службене речи достављају не према позицији већ су битне само пренете вредности.

Могуће је промешати позиционе аргументе и аргументе службене речи у позиву функције, али позициони аргументи се морају навести први а аргументи службене речи после њих.

```
show_interest(10000.0, korak=0.01, vremenski_period=10)
```

Овде је први аргумент (10000.0) достављен према својој позицији а друга два су аргументи службене речи.