

Придодавање података у постојећи фајл

Ретки су програми који користе фајлове за смештање малих количина података, обично су то огромне колекције података.

За рад са великим количинама података у фајловима, најчешће се користе петље.

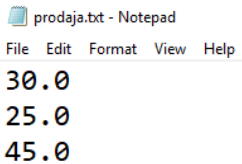
У следећем примеру се добија количина продаје у три дана од стране корисника и уписују те вредности у фајл prodaja.txt.

```
def main():
    broj_dana = int(input('Koliko dana je trajala prodaja? '))
    prodaja_fajl = open('prodaja.txt', 'w')

    for i in range(1, broj_dana + 1):
        prodaja = float(input('Uneti prodaju za dan #' + str(i) + ': '))
        prodaja_fajl.write(str(prodaja) + '\n')

    prodaja_fajl.close()
    print('Podaci su upisani u prodaja.txt.')
```

```
main()
Koliko dana je trajala prodaja? 3
Uneti prodaju za dan #1: 30
Uneti prodaju za dan #2: 25
Uneti prodaju za dan #3: 45
Podaci su upisani u prodaja.txt.
```



```
prodaja.txt - Notepad
File Edit Format View Help
30.0
25.0
45.0
```

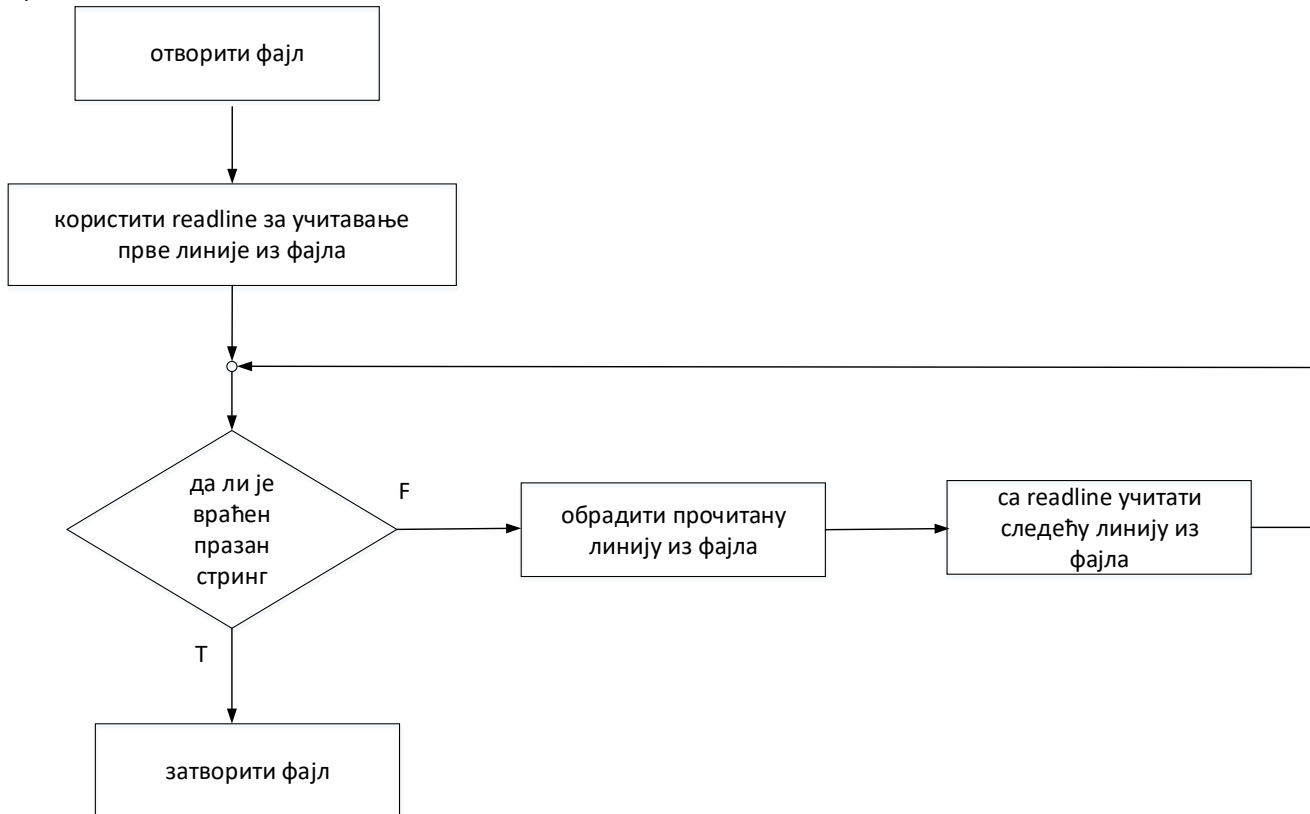
Читање фајла помоћу петље и детекција краја фајла

Често је потребно читати садржај фајла а да није познато колики је број података смештен у фајл.

Овде се поставља проблем детекције када се са читањем података дошло до краја фајла.

У Пајтону, метода `readline` враћа празан стринг (' ') када се покуша прочитати податак после последњег податка у фајлу.

Зато је могуће овде користити `while` петљу са испитивањем услова да ли је вредност враћена са `readline` празан стринг.



```

def main():
    prodaja_fajl = open('prodaja.txt', 'r')
    linija = prodaja_fajl.readline()
    while linija != '':
        kolicina = float(linija)
        print(format(kolicina, '.2f'))
        linija = prodaja_fajl.readline()

    prodaja_fajl.close()
  
```

```

main()
Даје:
30.00
25.00
45.00
  
```