



Курс програмског језика Пајтон ниво 0

07 час – for петља

1. Садржај

Садржај.....	1
Кратак опис	1
For петља	1
функција range	2
увучене петље.....	5
Тест.....	5
Вежбања	8
Задаци	7

2. Кратак опис

Ове структуре се користе када је потребно више пута извршити одређене инструкције. Често се структуре понављања називају и петље (loop). Петље се деле на две категорије, **петље контролисане условом** (condition-controlled) и **петље контролисане бројем понављања** (count-controlled).

3. For петља

Ова петља је петља контролисана бројем понављања. Формат петље:

```
for variable in [value1, value2, etc.]:  
    statement  
    statement  
    etc.
```

У првој линији **variable** је име променљиве. Унутар угласте заграде се појављују вредности одвојене зарезима и оне у Пајтону чине листу. Променљива **variable** добија прву вредност наведену у листи и тада се извршавају блокови наредби у телу петље. По завршетку извршења блокова, променљива **variable** добија следећу вредност из листе и тако редом до краја листе.

Пример 01:

```
print("Ispisujem brojeve od 1 do 5:")  
for broj in [1, 2, 3, 4, 5]:  
    print(broj)  
input("ENTER za kraj...")  
  
Ispisujem brojeve od 1 do 5:  
1  
2  
3  
4  
5  
ENTER za kraj...  
>>> |
```

Пошто у примеру постоји пет вредности у листи, петља ће се реализовати пет пута тј извршиће се пет итерација петље.

Променљиве које се користе у првој линији for структуре се зову **циљне променљиве** (target variable).

Вредности у листи петље не морају бити поређане у редоследу као у примеру 01:

**Пример 02:**

```
print("Ispisujem neparne brojeve od 1 do 9:")
for broj in [1, 3, 5, 7, 9]:
    print(broj)
input("ENTER za kraj...")
```

Ispisujem neparne brojeve od 1 do 9:
1
3
5
7
9
ENTER za kraj...|

Пример 03:

```
print("Ispisujem listu stringova:")
for ime in ['Mica', 'Pera', "Joca"]:
    print(ime)
input("ENTER za kraj...")
```

Ispisujem listu stringova:
Mica
Pera
Joca
ENTER za kraj...
>>> |

2. Range функција

Ова функција олакшава процес писања петљи контролисаних бројем понављања. Функција **range** прави тип објекта познат као итерација (iterable). Овај објекат садржи секвенце вредности које могу бити итериране преко нечега налик петљи:

```
for num in range(5):
    print(num)
```

Сада уместо коришћења листе вредности позива се **range** функција са аргументом 5. Сада функција генерише итерабилну секвенцу целих бројева у опсегу од 0 до 5 (али без укључивања и 5). Овај код ради исто што и: `for num in [0, 1, 2, 3, 4]`

Пример 04:

```
print("Ispisujem poruku 5 puta:")
for ime in range(5):
    print('Zdravo.')
input("ENTER za kraj...")
```

Ispisujem poruku 5 puta:
Zdravo.
Zdravo.
Zdravo.
Zdravo.
Zdravo.
ENTER za kraj...

Аргумент 5 у датом примеру се користи као ограничење секвенце бројева.

Ако се доделе два аргумента **range** функцији, први се користи као почетна вредност секвенце а други као ограничење секвенце вредности. По аутоматизму **range** функција повећава за 1 сваку следећу вредност у секвенци бројева.

Ако се дода и трећи аргумент у **range** функцији, он се користи као корак (step value) по којем се свака следећа вредност у секвенци бројева повећава.

Пример 05:

```
for broj in range(1, 10, 2):
    print(broj)
input("ENTER za kraj...")
```

1
3
5
7
9
ENTER za kraj...



У овом примеру, у for петљи се користи функција range која укључује три аргумента:

- први аргумент је вредност 1, који представља **почетну вредност** која ће се доделити промењивој broj
- други аргумент је вредност 10, који представља **граничну вредност** у листи. А то значи да ће промењива broj моћи да има било коју вредност у опсегу од 1 до 10, не укључујући вредност 10
- трећи аргумент је вредност 2, који представља **корак**. То значи да ће вредност 2 бити додавана свакој следећој вредности у секвенци листе
- петља функционише тако што после доделе почетне вредности промењивој (broj = 1) и одрађивања првог циклуса, на почетку следећег циклуса се врши сабирање претходне вредности у листи (1) са кораком листе (2) и добија се следећа вредност. Врши се провера да ли се та вредност налази у дозвољеном опсегу листе. Ако јесте у дозвољеном опсегу она се додељује промењивој (broj = 3)
- после одрађивања циклуса (итерације) за broj = 9, врши сабирање претходне вредности у листи (9) са кораком листе (2) и добија се следећа вредност. У овом случају добијена вредност је 11 што је изван опсега листе. Програм напушта структуру петље и креће да обрађује следећу линију кода

*Често се промењива којој се додељују вредности у петљи користи за разна израчунавања унутар тела петље:

Пример 06:

```
# Program prikazuje tabelu sa celim brojevima  
# od 1 do 10 i njihovim kvadratima
```

```
print('Broj\tKvadrat')  
print("-----")  
for broj in range(1, 11):  
    kvadrat = broj**2  
    print(broj, '\t', kvadrat)  
input("ENTER za kraj...")
```

Broj	Kvadrat
1	1
2	4
3	9
4	16
5	25
6	36
7	49
8	64
9	81
10	100
ENTER za kraj...	

Пример 07:

```
# Program konvertuje brzine iz  
# kilometara na sat u milje na sat
```

```
start_brzina = 60  
kraj_brzina = 131  
korak = 10  
faktor_konverzije = 0.6214  
  
print('km/h\tm/h')  
print("-----")  
  
for km in range(start_brzina, kraj_brzina, korak):  
    m = km * faktor_konverzije  
    print(km, '\t', m)  
input("ENTER za kraj...")
```



```

km/h      m/h
-----
60         37.284
70         43.498
80         49.711999999999996
90         55.925999999999995
100        62.139999999999999
110        68.354
120        74.568
130        80.782
ENTER za kraj...

```

*Понекад је неопходно допустити кориснику да контролише број итерација петље.

Пример 08:

```

# Program prikazuje brojeve i njihove kvadrate
# korisnik unosi gornju granicu

Uneti gornju granicu: 5

broj      kvadrat
-----
1         1
2         4
3         9
4         16
5         25
ENTER za kraj...

kraj = int(input('Uneti gornju granicu: '))

print()
print('broj\tkvadrat')
print("-----")

for broj in range(1, kraj + 1):
    kvadrat = broj**2
    print(broj, '\t', kvadrat)
input("ENTER za kraj...")

```

Корисник уноси део горње границе опсега и на тај начин директно утиче на број итерација. Наравно, могуће је да корисник утиче на све делове петље својим уносима.

*Коришћење акумулатора

После сваке итерације у петљи долази до промене вредности неке од променљивих које се користе у петљи. Ако су у питању бројчане вредности, често се рачуна сума добијена променом вредности током итерација. Овакав збир се смешта у променљиву која се зове **акумулатор**. Сваки акумулатор се пре стартовања петље мора поставити на 0.

Пример 09:

```

# Program sabira brojeve koje korisnik unese

max = 5
zbir = 0      # inicijalizacija akumulatora

for brojac in range(max):
    broj = int(input("Unesi broj: "))
    zbir = zbir + broj

print('Ukupan zbir unetih brojeva je', zbir)

Unesi broj: 2
Unesi broj: 3
Unesi broj: 4
Unesi broj: 0
Unesi broj: 1
Ukupan zbir unetih brojeva je 10

```



*Увучене петље

Увучена петља се назива петља која се налази унутар тела друге петље.

```
for sati in range(24):  
    for minuti in range(60):  
        for sekunde in range(60):  
            print(sati, ':', minuti, ':', sekunde)|
```

У примеру се најбрже врши итерација за добијање секунда, затим итерација у петљи за минуте па за часове. Види се да унутрашње петље пролазе свим својим итерацијама за време једне итерације спољне петље.

На излазу ће се појавити:

0:0:0

0:0:1

0:0:2

.....

23:59:59

Пример 10:

```
# Program racuna srednje vrednosti rezultata sa svih testova
```

```
broj_ucenika = int(input("Koliko ucenika je radilo testove? "))
```

```
broj_testova = int(input("Koliko testova je uradjeno? "))
```

```
for ucenik in range(broj_ucenika):  
    zbir = 0.0      # inicijalizacija akumulatora  
    print("Ucenik broj", ucenik + 1)  
    print('-----')  
    for test_broj in range(broj_testova):  
        print('Test broj', test_broj + 1, end='')  
        rezultat = float(input(': '))  
        zbir += rezultat  
    srednja_vrednost = zbir / broj_testova  
    print('Srednja vrednost rezultata testova ucenika broj',\  
          ucenik + 1, ' je:', srednja_vrednost)  
    print()
```

```
Koliko ucenika je radilo testove? 3
```

```
Koliko testova je uradjeno? 3
```

```
Ucenik broj 1
```

```
-----
```

```
Test broj 1: 100
```

```
Test broj 2: 95
```

```
Test broj 3: 90
```

```
Srednja vrednost rezultata testova ucenika broj 1 je: 95.0
```

```
Ucenik broj 2
```

```
-----
```

```
Test broj 1: 80
```

```
Test broj 2: 81
```

```
Test broj 3: 82
```

```
Srednja vrednost rezultata testova ucenika broj 2 je: 81.0
```

```
Ucenik broj 3
```

```
-----
```

```
Test broj 1: 75
```

```
Test broj 2: 85
```

```
Test broj 3: 80
```

```
Srednja vrednost rezultata testova ucenika broj 3 je: 80.0
```



У примеру се користе следеће промењиве:

broj_ucenika – број ученика који су радили све тестове

broj_testova – број тестова које су ученици радили

ucenik – редни број ученика чији се резултати тренутно уносе

zbir – збир бодова са свих тестова које је добио ученик под редним бројем **ucenik**

test_broj – редни број теста чији се резултат за ученика под редним бројем **ucenik** тренутно уноси

rezultat – резултат теста под редним бројем **test_broj** ученика под редним бројем **ucenik**

srednja_vrednost – средња вредност резултат тестова ученика под редним бројем **ucenik**

У примеру се користи увучена петља која врши итерације по редном броју теста. Спољна петља врши итерације по редном броју ученика. По изласку из увучене петље добија се збир бодова на тестовима које је урадио ученик под редним бројем **ucenik** и може се наћи средња вредност тих тестова.

НАПОМЕНА

У линији кода: `print('Test broj', test_broj + 1, end = '')`

се користи једна од инструкција за форматирање стрингова на излазу, `end = ''`. Ова инструкција омогућава да се по завршетку исписивања стринга не прелази у следећи ред већ да се настави са исписивањем у истом реду.

*Употреба for петље за цртање карактерима

Пример 11:

```
redovi = int(input('Koliko redova? '))
kolone = int(input('Koliko kolona? '))
for x in range(redovi):
    for y in range(kolone):
        print('*', end = '')
    print()
```

```
Koliko redova? 5
Koliko kolona? 10
*****
*****
*****
*****
*****
```

Пример 12:

```
osnovica = 8
for x in range(osnovica):
    for y in range(x + 1):
        print('*', end = '')
    print()
```

```
*
**
***
****
*****
*****
*****
*****
```

Пример 13:

```
koraci = 6
for x in range(koraci):
    for y in range(x):
        print(' ', end = '')
    print('#')
```

```
#
 #
  #
   #
    #
     .
```



Тест

1. Која од следећих је петља контролисана бројањем понављања?
 - a) for
 - б) foreach
 - в) while
 - г) do-while
2. Каква заграда се користи и шта се у њој налази код for-in петље ?
3. Постоји ли ограничење у броју вредности код for-in петље ?
4. Колико итерација ће бити у for петљи са листом од 4 елемента ?
5. Која од промењивих у наведеном коду је циљна промењива ?

```
a = 1
print(a)
for x in [10, 7, 2]:
    print(a + x)
```

 - a) x
 - б) a + x
 - в) in
 - г) a
6. Оператор -= је пример _____ оператора.
 - a) релационог
 - б) аугументована додела
 - в) комплексна додела
 - г) повратна додела
7. Чему служи функција range ?
8. Шта значи када се испише range(2) унутар for петље?
9. У претходном задатку 2 је:
 - a) функција
 - б) услов
 - в) аргумент
 - г) промењива
10. Ако хоћу да дефинишем step value унутар функције range, колико вредности се мора унети као аргумент у функцију ?
 - a) 1
 - б) 0
 - в) 2
 - г) 3
11. У примеру for x in range(1, 5, 2) шта представља број 5 ?
 - a) корак
 - б) број итерација
 - в) граничну вредност
 - г) почетну вредност
12. Колико ће се извршити итерација у примеру 11 ?
13. Да ли треба онемогућити да корисник директно утиче на број итерација у петљи?



14. Како се зове промењива у коју се смешта збир вредности које се добијају током итерација у петљи ?
15. Која је најчешћа почетна вредност акумулатора ?
16. Шта су то увучене петље ?

Вежбања

1. Напиши for петљу која исписује следећи низ бројева:
0, 10, 20, 30, 40, 50...1000
2. Напиши петљу која тражи од корисника да унесе број. Петља треба да итерира 10 пута и да садржи тотални збир свих унетих бројева почевши од унетог броја
3. Користећи for in петљу у низу 3, 5, 15, 21, 36, 111 пронаћи бројеве дељиве са 3, са 7 и са 9
4. Проверити да ли се унето име налази у списку имена станара зграде
5. Напиши програм који прима број унет од корисника и проверава ако је паран или непаран. Ако је паран, исписује низ целих бројева до 4 коришћењем range() а ако је непаран исписује низ до 7
6. Исписати сваки пети паран број у опсегу целих бројева од а до b
7. Сваки пут када се пронађе одређени број у низу нека се чује звук
8. Сабрати збир бацање два коцке па изабрати највећи збир из три бацања
9. Две петље итерирају по бројевима. Приказати оне бројеве који се јављају у обе петље
10. Нацртати ASCII круг коришћењем for петље

Задаци

1. Сакупљач буба

Сакупљач буба сакупља бубе сваки дан у пет дана. Написати програм који садржи укупан број буба сакупљених током тих пет дана. Петља треба да пита за број сакупљених буба сваког дана и када се заврши петља, програм треба да прикаже укупан број сакупљених буба.

2. Сагореле калорије

Трчањем по плану сагориш 4.2 калорије по минути. Напиши програм који користи петљу за приказ броја калорија сагорелих после 10, 15, 20, 25 и 30 минута.

3. Анализа буџета

Напиши програм који тражи од корисника да унесе укупан буџет за једну годину. Петља треба онда да тражи од корисника да унесе све трошкове за сваки месец и да прати тоталан збир. Ако је буџет прекорачен током године, послати поруку. Када се петља заврши, програм треба да прикаже укупну суму са којом је корисник изнад или испод буџета.

4. Дужина пута

Даљина коју прелази возило се може израчунати као: $daljina = brzina \times vreme$. На пример, ако воз путује 40 километара на сат за три сата, пређени пут је 120 km. Написати програм који пита корисника за брзину возила (у km/h) и за број сати које је путовао. Треба онда да користи петљу за приказ растојања које је возило прешло за сваки сат тог периода. Пример:



која је брзина возила у km/h ?	40	Enter
колико сати је трајао пут ?	3	Enter
сат	пређено растојање	
1	40	
2	80	
3	120	

5. Дневна температура

Написати програм који користи увучене петље за сакупљање података о променама температуре током 3 дана. Спољна петља ће итерирати за дане. Унутрашња петља ће итерирати за 6, 12, 18 и 24 часа. Свака итерација унутрашње петље ће тражити од корисника температуру у степенима. Програм ће приказати просечну дневну температуру.

6. Табела за целзијусе у фаренхајт

Напиши програм који приказује табелу целизијусових степени од 0 до 20 и њихових еквивалената у фаренхајтовим степенима. Формула за конверзију је $F = (9/5) * C + 32$, где је F температура у фаренхајтовим степенима а C температура у целзијусовим степенима. Програм мора користити петљу за приказ табеле.

7. Пени као плата

Напиши програм који израчунава суму новца које ће човек зарадити током периода времена ако је његова плата један пени првог дана, два пенија другог дана а за даље се удвостручава сваког дана. Програм треба да пита корисника за број дана. Приказати табелу која показује каква је плата сваког дана а онда прикаже тоталну суму плате на крају периода. Излаз треба да је приказан у суми долара а не броја пенија. (1 долар = 100 пенија)

8. Нивои океана

Претпостављајући да се нивои океана повећавају за 1.6 милиметара годишње, направити апликацију која приказује број милиметара за које ће се океан подићи сваке године за следећих 25 година.

9. Повећање уписнине

У једном колеџу, уписнина за редовног студента је \$8,000 по семестру. Објављено је да ће уписнина се повећати за 3% сваке године у следећих 5 година. Написати програм са петљом који приказује пројектоване уписнине за семестре за следећих 5 година.

10. Израчунавање факторијела броја

У математици, нотација $n!$ представља факторијел позитивног целог броја n . Факторијел од n је производ свих ненегативних целих бројева од 1 до n . Пример:

$$7! = 1 \times 2 \times 3 \times 4 \times 5 \times 6 \times 7 = 5,040$$

$$4! = 1 \times 2 \times 3 \times 4 = 24$$

Написати програм којим корисник уноси ненегативан цео број и користи петљу за израчунавање факторијела тог броја. Приказати факторијел.



11. Организми

Написати програм који предвиђа апроксимативни број организама. Апликација треба да користи текст бокс да би омогућила кориснику да унесе почетни број организама, просечно дневно повећање популације (у процентима) и број дана који ће се оставити организмима за размножавање. На пример:

Почетни број организама: 2	
Дневно пораст у процентима: 30	
Дана за размножавање: 10	
Дана	Број организама
1	2
2	2.6
3	3.38
4	4.394
5	5.7122
6	7.42586
7	9.653619
8	12.5497
9	16.31462
10	21.209

12. Напиши програм који користи увучене петље за цртање следеће шеме:

```
*****
*****
*****
****
***
**
*
```

13. Напиши програм који користи увучене петље за цртање следеће шеме:

```
##
# #
#  #
#   #
#    #
#     #
#      #
```