

Понављање наредби

Често је потребно да се наредба у програму понови више пута.

Коришћење наредби понављања омогућава краће записивање програма.

Наредбе које омогућавају да се једна наредба или неколико наредби понове више пута називају се **петље**.

Петље омогућавају краћи и јаснији запис програмског кода у којем треба да се више пута понови једна или неколико наредби.

У програмском језику Пајтон постоје две врсте петљи: **for** и **while**.

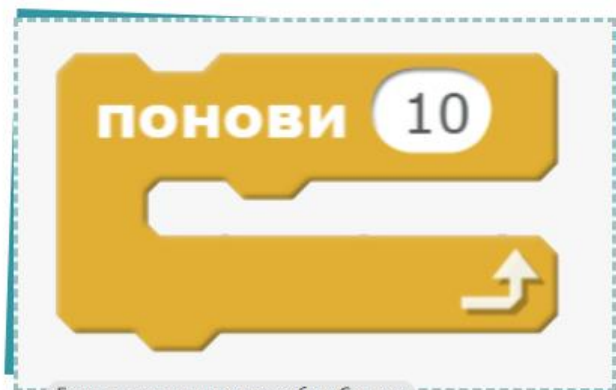
Петља *for*

Некада је унапред познато колико пута треба да се понове наредбе.

У Скречу постоји блок у који се стављају блокови чије извршавање треба поновити одређени број пута.

У програмском језику Пајтон за понављање наредби тачно одређени број пута користи се петља **for**.

На крају првог реда петље **for** обавезно треба ставити знак **:** и након њега се, увучене у односу на остале наредбе у програмском коду, пише једна или више наредби које треба да се понављају у петљи.



Блок за понављање наредби у Скречу

```
for i in range (broj) :
    naredba
```

Општи облик петље *for*

Функција `range` одређује колико пута ће се поновити наредбе у петљи (на пример, за цртање квадрата наредбе ће се поновити четири пута).

Током извршавања петље бројач узима вредности из скупа бројева који је одређен функцијом `range`.

Функције `range` се може позвати на три различита начина.

Пример позивања функције <code>range</code>	Објашњење	Скуп вредности за бројач
<pre>for i in range(4): print(i)</pre>	Функција <code>range</code> даје вредности из интервала од 0 до 3.	0, 1, 2, 3
<pre>for i in range(3, 10): print(i)</pre>	Функција <code>range</code> даје низ бројева од првог до другог задатог броја, али други број није укључен у низ.	3, 4, 5, 6, 7, 8, 9
<pre>for i in range(1, 11, 2): print(i)</pre>	Трећи број представља корак бројања.	1, 3, 5, 7, 9

Петља `for` се може користити за испис садржаја листе.

Петља `for` ће прво исписати елемент листе на позицији 0 ($i=0$), затим други елемент ($i=1$), трећи елемент ($i=2$) и тако редом за све елементе листе.

```
radni_dani=['ponedeljak', 'utorak',  
            'sreda', 'cetvrtak', 'petak']  
  
for dan in radni_dani:  
    print(dan)
```

Програм за исписивање елемената листе

```
ponedeljak  
utorak  
sreda  
cetvrtak  
petak  
>>>
```

Резултат рада програма

Петља *while*

Понекад није унапред познато колико пута треба поновити неку наредбу.

Потребно да се она понавља све док је испуњен неки услов.

За понављање наредби док је неки услов тачан користи се петља *while*.

Наредбе које треба да се понављају у петљи увлаче се за један табулатор (Tab) или четири размака удесно у односу на остатак програмског кода.

```
while uslov:  
    naredba
```

Општи облик петље *while*

Променљива **uslov** је логичка променљива и може имати две вредности: тачно и нетачно.

Наредбе у петљи понављаће се све док је услов тачан.

За прављење услова најчешће се користе релацијски оператори за упоређивање две вредности.



Оператор

Опис

>	веће
>=	веће или једнако
<	мање
<=	мање или једнако
==	једнако
!=	различито

Цифре неког броја могу се издвојити помоћу операција целобројно дељење и остатак при целобројном дељењу.

На пример, цифре двоцифреног броја 57 могу се издвојити на следећи начин: $57\%10=7$ и $57//10=5$.

Када није познато да ли је број двоцифрен односно колико цифара има број, треба употребити петљу **while**.



```
n=int(input('Unesi ceo broj:'))
while n>0:
    print(n%10)
    n=n//10
```

Програм који испишује цифре броја почевши од цифре јединица

```
Unesi ceo broj:123
3
2
1
>>>
```