

Petlja unutar petlje

PROŠIRIT ĆEMO ZNANJE O PETLJAMA

Što smo dosad naučili na informatici po pitanju petlji?

Naučili smo da imamo `for` i `while` petlje.

Naučili smo da imamo tri oblika `for` petlji. Prvi oblik je kada funkcija `range` ima samo jedan broj u sebi, odnosno unutar zagrada. Drugi oblik je kada funkcija `range` ima dva broja u sebi. Treći oblik je kada funkcija `range` ima tri broja u sebi.

Promotrimo sljedeći primjer kada `for` petlja ima funkciju `range` s jednim brojem u sebi:

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

U ovom primjeru petlja će **5 puta** izvršiti naredbu `print(k)`, odnosno ispisat će brojeve 0, 1, 2, 3 i 4.

Petlja unutar petlje

Kada pišemo programe moguće je stavljati petlje unutar drugih petlji. U prethodnom primjeru `for` petlja je ponavljala naredbu `print(k)`, ali ona može ponavljati bilo koju drugu naredbu. Može ponavljati čak i drugu petlju.

Evo primjera:

```
for k in range(5):  
    for j in range(4):  
        print(j)
```

1. petlja

2. petlja

Petlja `for k in range(5)` će sada 5 puta ponoviti petlju `for j in range(4)`. Odnosno, **1. petlja** će 5 puta ponoviti **2. petlju**.

Primijetite da petlje imaju različite kontrolne varijable! Prva petlja ima varijablu `k`, dok druga petlja ima varijablu `j`. Ne smiju imati iste kontrolne varijable! Upamtite to.

Petlja unutar petlje

Što će se ispisati izvođenjem dviju petlji s prethodne stranice?

Ispisat će se sljedeće:

```
0
1
2
3
0
1
2
3
0
1
2
3
0
1
2
3
0
1
2
3
```

Petlja unutar petlje

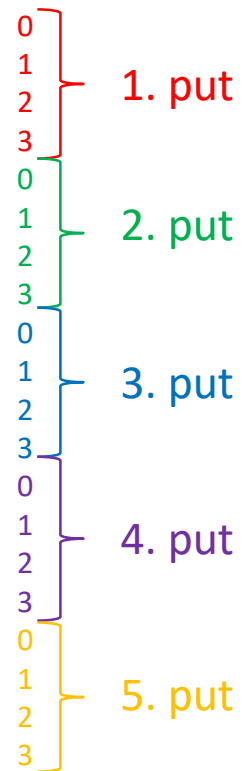
Objašnjenje prethodnog ispisa:

Druga petlja 4 puta izvršava naredbu `print(j)`, odnosno ispisuje brojeve 0, 1, 2 i 3. Pritom je svaki broj u novom redu.

Prva petlja 5 puta izvršava drugu petlju. To zapravo znači da će ispisivanje brojeva 0, 1, 2 i 3 ponoviti 5 puta!

Promotrite sliku na sljedećoj stranici:

Petlja unutar petlje



Petlja unutar petlje

Ako želimo dobiti malo ljepši ispis, odnosno ovakav ispis:

0 1 2 3

0 1 2 3

0 1 2 3

0 1 2 3

0 1 2 3

Onda ćemo to učiniti pomoću sljedećih naredbi:

```
for k in range(5):  
    for j in range(4):  
        print(j, end = ' ')  
    print('\n')
```

Kao što smo učili, `end = ' '` govori naredbi `print` da kad ispiše neku vrijednost ne ide u novi red, nego da stavi razmak. Ovime će se brojevi 0, 1, 2 i 3 ispisati u jednom redu.

Naredba `print('\n')` neće zapravo ništa ispisati, nego će samo prijeći u novi red. Ako ne stavimo `end = ' '`, naredba `print` prelazi u novi red. Ova `print` naredba će se izvršiti tek kad se ispišu brojevi 0 1 2 3.

Zadaci koje trebate riješiti

Riješite sljedeća tri zadatka i pošaljite odgovore na Yammer u obliku privatne poruke.

1. Što će se ispisati izvođenjem sljedećih petlji?

```
for k in range(3):  
    for j in range(2):  
        print(j)
```

2. Što će se ispisati izvođenjem sljedećih petlji?

```
for k in range(4):  
    for j in range(3):  
        print(j, end = ' ')  
    print('\n')
```

3. Napišite petlje koje će ispisati sljedeće brojeve:

```
0 1 2 3 4 5  
0 1 2 3 4 5  
0 1 2 3 4 5  
0 1 2 3 4 5  
0 1 2 3 4 5
```