

WHILE PETLJA

While petlja je programska struktura koja ima istu namjenu kao i for petlja. To znači da while petlja isto omogućava ponavljanje određenog bloka naredbi. Naravno razlika ove petlje i for petlje je u tome što while petlja ne sadržava **brojač** kojeg sadržava for petlja. Brojač for petlji omogućava određivanje određenog broja ponavljanja nekog bloka naredbi. While petlja nema brojač, ali kao i for petlja ima uvjet. Taj uvjet jo omogućava da se izvršavaju naredbe. **Tako dugo dok je uvjet ispunjen ponavljaaj blok naredbi.** Ako uvjet nije ispunjen, nemoj ponavljati taj blok naredbi. Tako bismo mogli prevesti funkcionalnost while petlje.

While uvjet:

- Blok naredbi 1
- Blok naredbi 2
-

Python deklaracija

Dok je uvjet ispunjen:

- Izvrši naredbu jedan
- Izvrši naredbu dva.....

Govorni jezik

While petlja je vrlo jednostavna petlja koja samo omogućava ponavljanje kada je određeni uvjet ispunjen. Uvjet u while petlji može biti određen operatorima: **usporedbe, logičkim operatorima i aritmetičkim operatorima.**

ZADACI

Zadatak1:

Napisati program koji će korisniku dati mogućnost pogađanja lozinke. Mogućnost pogađanja lozinke izvršavati će se toliko puta dok korisnik ne pogodi lozinku. Kada korisnik pogodi lozinku, ispisati **upisali ste točnu lozinku** u suprotnom javiti vijest o pogrešno upisanoj lozinci i ponoviti unos.

Primjer ispisa:

```
Unesite lozinku: 'tajno'  
Upisali ste točnu lozinku
```

Rješenje:

```
password = ""  
while password != "tajno":  
    password = input("Unesite lozinku: ")  
    if password == "tajno":  
        print("Upisali ste točnu lozinku")  
    else:  
        print("upisali ste pogresnu lozinku-pokusajte ponovno")
```

U rješenju je na mjestu while uvjeta korišten operator usporedbe !=. Program je ponavljao **unesite ozinku** tako dugo dok uvjet u while petlji nije bio ispunjen točnije dok **password=="tajno"** odnosno dok **password** više nije bio !=**"tajno"**

Zadatak2:

Napraviti program koji će korisniku omogućiti pogađanje brojeva. Ako korisnik upiše bilo koji broj, izvršavati će se blok naredbi ispod while petlje. Ako odabere 0, dogodi se prekid programa. Prekid programa omogućiti sa naredbom **break**. Ako korisnik upiše točan broj, ispiše se poruka o pogodnom broju i program se dalje izvršava. Ako korisnik napiše prevelik ili premali broj od traženog, ispisati prigodnu poruku korisniku i dalje izvršavati program, sve dok korisnik sma ne odabere opciju 0.

Primjer ispisa:

```
Igra pogađanja brojeva
Za prekid igre unesite 0
Unesite broj: 17
Pogodili ste zadani broj cestitamo!
Unesite broj: |
```

Rješenje:

```
print ("Igra pogađanja brojeva")
print ("Za prekid igre unesite 0")

pogodi=17

while 1:
    broj=int(input("Unesite broj: "))
    if broj==pogodi:
        print ("Pogodili ste zadani broj cestitamo!")
        broj=0
    elif broj==0:
        break
    elif broj > pogodi:
        print ("Broj je veci od zadanog, pokusajte ponovo")
    elif broj < pogodi:
        print ("Broj je manji od zadanog, pokusajte ponovo")
|
```

Dakle kao fiksni broj koji je trebalo pogoditi, stavio se broj 17. U while petlji uvjet se stavio na 1. U svijetu logičkih operatora to znači **true** odnosno stanje u kojem postoji vrijednost. Taj 1 u uvjetu znači ako je unesen bilo koji broj koji nije nula, izvršavaj naredbe ispod petlje while. Kada korisnik odabere 0 to je kao da je odabrao **false** odnosno stanje bez vrijednosti; program se prekida.

Zadatak3:

Tražiti od korisnika unos broja u rasponu od 10-20. Ukoliko korisnik pogodi raspon; ispisati: **cestitamo-unijeli ste broj u rasponu** i ispisati broj kojeg je korisnik unio. Ako korisnik ne pogodi broj, ispisati: **broj nije u rasponu od 10-20; Pokušajte ponovno**. Petlja se izvršava tako dugo dok je uvjet na TRUE odnosno dok korisnik unosi brojeve veće od nula. Ako korisnik unese 0 odnosno FALSE, prekida se izvođenje programa.

Primjer ispisa:

```
unesite broj u rasponu od 10 do 20: 12
('Cestitamo_unijeli ste broj u rasponu ', 12)
```

```
unesite broj u rasponu od 10 do 20: 0
>>> |
```

Rješenje:

```
while True:
    broj = int(input("unesite broj u rasponu od 10 do 20: "))
    if broj >= 10 and broj <= 20:
        print("Cestitamo_unijeli ste broj u rasponu ",broj)

    elif broj==False:
        break
    else:
        print("Broj nije u rasponu od 10 do 20")
        print("Pokusajte ponovno")
```

Zadatak4:

Program treba omogućiti korisniku odabir jedne od četiri aritmetičke operacije (+,-,*,/) Svakoj operaciji je potrebno dodijeliti neki broj (npr. 1. Zbrajanje, 2. Oduzimanje, 3. Množenje, 4. Dijeljenje, 5. Izlaz iz programa) Kada korisnik odabere jednu od navedenih operacija, od njega se traži da unese dva broja. Ispisati rezultat odabrane operacije.

Rješenje:

```
izbor = 0
while 1:
    print "Dobrodošli u kalkulator.py!"
    print "Izaberi racunsku operaciju:"
    print " "
    print "1) Zbrajanje"
    print "2) Oduzimanje"
    print "3) Mnozenje"
    print "4) Dijeljenje"
    print "5) Izlaz iz kalkulatora"
    print " "
    izbor = input("Unesi broj ispred zeljene operacije:")

    if izbor == 1:
        prib1 = input("Zbroji ovaj broj: ")
        prib2 = input("s ovim brojem: ")
        print prib1, "+", prib2, "=", prib1 + prib2

    elif izbor == 2:
        oduz1 = input("Oduzmi od ovog broja: ")
        oduz2 = input("ovaj broj: ")
        print oduz1, "-", oduz2, "=", oduz1 - oduz2

    elif izbor == 3:
        mnoz1 = input("Pomnozi ovaj broj: ")
        mnoz2 = input("sa ovim brojem: ")
        print mnoz1, "*", mnoz2, "=", mnoz1 * mnoz2

    elif izbor == 4:
        dij1 = input("Podijeli ovaj broj: ")
        dij2 = input("sa ovim brojem: ")
        print dij1, "/", dij2, "=", dij1 / dij2

    elif izbor == 5:
        break

print "Hvala sto ste koristili kalkulator.py!"
```