

UVJETNE NAREDBA IF-ELSE—ELIF

Što moramo znati??: naredba za ispis je `print()`, `"="` je znak pridruživanja vrijednosti varijabli. Postoji više načina ispisa (poželjno da znate barem jedan), a aritmetičke operatore ste obavezni znati.

Dosad smo u programima koristili aritmetičke operatore. Međutim, što učiniti ako trebamo u programu uspoređivati više vrijednosti?

Tu nam pomažu operatori uspoređivanja.

Ima ih nekoliko, također ih već znate iz matematike, ali u matematici se oni pišu malo drugačije u odnosu na Python. U nastavku se nalazi prikaz operatora.

Značenje	Oznaka u matematici	Oznaka u Pythonu
Manje od	<	<
Veće od	>	>
Manje ili jednako	≤	<=
Veće ili jednako	≥	>=
Jednako	=	==
Različito	≠	!=

NAPOMENA: Pazite na razliku između `"="` i `"=="`. Znak `"="` je znak pridruživanja koji služi da se neka vrijednost pridruži varijabli, dok znak `"=="` predstavlja operator uspoređivanja koji je istinit ako su uspoređene vrijednosti jednake.

Možemo dalje ?. Naime, u programu mogu postojati uvjeti u kojima se program grana na više mogućih rješenja ovisno o uvjetu. To nazivamo selekcijom. Selekcija koju ćemo naučiti je selekcija tipa `if` (i njezine varijacije).

Primjer_1:

Evo primjera iz stvarnog života; dođemo u dućan sportske opreme i kupujemo sve što nam treba za dječje igralište . Na kraju kupnje dođemo na blagajnu i u slučaju iznosa većeg od 5000 kn imamo pravo na besplatnu dostavu što se ispisuje na računu. Upravo ćemo ovaj zadatak riješiti, ali ćemo ga malo pojednostaviti. Kako ne bismo unosili svaku stavku posebno, unijet ćemo samo konačnu vrijednost kupovine, a program nam treba ispisati: vrijednost računa, pravo na dostavu (samo u slučaju vrijednosti računa većeg od 5000 kn) te prigodnu rečenicu ("Dođite nam opet!").

Sintaksa za if

If uvjet:
blok naredbi

<pre>#Unos vrijednosti računa racun=float(input("Unesite vrijednost računa")) #Selekcija if i ispis prigodne rečenice If racun>5000: print("Imate pravo na besplatnu dostavu") print("Dodajte nam opet")</pre>	<pre># - koristimo za komentare ,oni služe za lakše snalaženje u programu i ne utječu na izvršavanje programa. Ono što je u selekciji if mora biti uvučeno, najbolje koristite tipku tab na tipkovnici. Ovaj dio naredbe nije vezan za selekciju if</pre>
---	---

Kao što vidite na primjeru, **if** uvjet sastoji se od ključne riječi **if**, uvjeta i nakon toga slijedi znak **“:”**. Važno je nakon toga naglasiti da sve što spada pod naredbu **IF se nalazi uvučeno i to će se izvršiti samo ako je uvjet zadovoljen**.

Kao uvlaka se preporučuje tipka **tab** i to uređen kao četiri razmaka.

*Naravno, to je najjednostavnije korištenje **if** uvjeta. Sada zamislimo sljedeći slučaj. Došli smo u školu i trebamo napraviti program koji će nam ispisivati da li smo prošli ili pali ispit. To ćemo učiniti tako da napravimo program koji će u slučaju ocjene 1 ispisivati “Niste položili ispit.”, a za ostale slučajeve ispisivati “Položi ste ispit ! Čestitamo”*

Sintaksa za if – else:

if uvjet:

 blok naredbi

else:

 blok naredbi

<pre># unos ocjene ocjena=int(input("Unesite ocjenu ")) #Selekcija za if – else if ocjena==1: print("Niste položili ispit.") else: print("Položili ste ispit ! Čestitamo")</pre>	<p>Kao što vidite u selekciji if koristimo znak jednakosti ==.</p> <p>Ako je uvjet točan izvršit će se onaj dio programa koji je u selekciji if.</p> <p>Ako uvjet nije točan izvršit će se onaj dio programa koji se nalazi ispod else. Važno je zamijetiti da se naredbe ispod if i else moraju uvući.</p>
--	--

Programski primjeri s rješenjima:

<p>Napiši program koji unosi dva broja te ispisuje poruku jesu li brojevi jednaki ili različiti.</p>	<pre>#program za unošenje i usporedbu dva unesena broja a=int(input("Upiši prvi broj:")) b=int(input("Upiši drugi broj:")) if a==b: print("Brojevi su jednaki") else: print("Brojevi su različiti")</pre>
--	---

Primjer_2:

<p>Napiši program za unos dva broja (input) te ispisuje poruku koji od ta dva broja je veći.</p>	<pre>#usporedba unesenih brojeva a=int(input("Unesi prvi broj:")) b=int(input("Unesi drugi broj:")) if a>b: print("broj",a,"je veći") else: print("broj",b,"je veći")</pre>
--	--

Primjer_3:

<p>Napišite program koji unosi duljine stranica a i b te ispisuje poruku da li se o kvadratu ili pravokutniku.</p> <p>Važno:</p> <p>Što je karakteristično za kvadrat ??</p>	<pre>a=int(input("duljina stranice a:")) b=int(input("duljina stranice b:")) if a==b: print("KVADRAT") else: print("PRAVOKUTNIK")</pre>
---	---

Kao što možete vidjeti, za uspješno rješavanje zadataka sa selekcijom IF – ELSE bitno je poznavati matematičke operatore uspoređivanja. (>,<==,!=..) i aritmetičke operatore (+,-,*,/,//, %....)..

Zadaci za vježbu :

Zadatak_1:

Napiši program koji unosi jedan broj(input)te ispisuje je li taj broj veći ili manji od 100.

Zadatak_2:

Napiši program koji unosi jedan broj (input) te ispisuje poruku je li učitani broj paran ili ne.*Kada je neki broj paran ??, koji ćete aritmetički operator koristiti ??*

Zadatak_3:

Napiši program koji unosi jedan broj te ispisuje poruku je li učitani broj djeljiv s 5 ili ne.

Sintaksa za if – elif – else: višestruki uvjeti:**If uvjet:****blok naredbi****elif uvjet:****blok naredbi****else:****blok naredbi**

Zamislamo situaciju da ovisno o unesenoj ocjeni od 1 do 5, program ispisuje poruku ocjenu koju ste dobili i da li ste ili niste prošli zamišljeni ispit.

<pre># unos ocjene ocjena=int(input("Unesite ocjenu: ")) # selekcija if – elif – else If ocjena==1: print("Ocjena ispita je nedovoljan, niste prošli") elif ocjena==2: print(" Ocjena ispita je dovoljan, prošli ste") elif ocjena==3: print("Ocjena ispita je dobar,prošli ste") elif ocjena==4: print("Ocjena ispita je vrlo dobar,prošli ste") elif ocjena==5: print("Ocjena ispita je odličan, prošli ste") else: print(" Unijeli ste krivu ocjenu")</pre>	<p>elif – kao što možete vidjeti na ovom primjeru koristimo višestruke uvjete za prikaz svake ocjene od 1 do 5.</p> <p>Ako ocjena nije u rasponu od 1 do 5 aktivirat će se događaj pod naredbom else.</p>
--	---

Primjeri programa s rješenjima:

Primjer_1:

<p>Napiši program koji unosi rezultat nogometne utakmice za Španjolsku i Hrvatsku te ispisuje tekst "Španjolska je pobjednik" u slučaju da je Španjolska pobijedila, tekst "Hrvatska je pobjednik" u slučaju da je Hrvatska pobijedila ili tekst "Neriješeno" u slučaju da su obje ekipe osvojile jednak broj bodova.</p>	<pre>h=int(input(" Broj golova hrvatske: ")) s=int(input("Broj golova španjolske: ")) if h>s: print("Hrvatska je pobjednik") elif h<s: print("španjolska je pobjednik") elif h==s: print("Nerješeno")</pre>
---	---

Kao što vidite u ovom primjeru uvjeti mogu završiti i bez naredbe else:

Primjer_2:

Napiši program koji prevodi boje (plava, zelena, crvena) s hrvatskog na engleski jezik. Ukoliko upišemo nešto drugo osim te tri boje, pojavit će se poruka: Ta boja ne postoji u programu.	<pre> b=input("Upisi boju:") if b=="plava": print ("plava – blue") elif b=="zelena": print ("zelena – green") elif b=="crvena": print ("crvena – red") else: print ("Ta boja ne postoji u programu!") </pre>
--	--

Logički operatori:

Sada smo prošli selekciju **if** i njene varijacije. Međutim, u programiranju se znaju javljati zadaci koji zahtijevaju da neki uvjeti budu povezani tako da se zadatak izvrši samo ako su oba uvjeta zadovoljena i sl. Tu na scenu stupaju logički operatori.

Logički operatori su operatori koji ovisno o uvjetu vraćaju **True** (istinu) odnosno logički 1 i **False** (laž) odnosno logičku 0. U Pythonu postoje logički operatori **and** (i), **or** (ili) i **not** (ne).

Logički operator **and** je operator koji vraća vrijednost **True** (istinu) ako i samo ako su svi uvjeti istiniti (zadovoljeni). U suprotnome će vraćena vrijednost biti **False** (laž).

Tablica istinitosti za AND			Tablica istinitosti za OR		
Uvjet 1.	Uvjet 2.	rezultat	Uvjet 1.	Uvjet 2.	rezultat
1	0	0	1	0	1
0	1	0	0	1	1
0	0	0	1	1	1
1	1	1	0	0	0
1 je TRUE	0 je FALSE				

VAŽNO:

Kod logičkog operatora **AND** svi uvjet moraju biti zadovoljeni da bi uvjet bio točan

Kod logičkog operatora **OR** dovoljno je da jedan od uvjeta bude točan.

Primjer zadatka s logičkim operatorom

Napiši program koji unosi dob putnika te ispisuje poruku o cijeni karte prema pravilima: * osobe mlađe od 8 godina voze se besplatno * osobe od 8-18 godina plaćaju kartu 5 kn * osobe starije od 65 godina plaćaju kartu 1 kn * sve ostale osobe plaćaju kartu 8 kn	<pre> a=int(input("Upiši starosnu dob putnika: ")) if a<8: print("karta je besplatna") elif a>=8 and a<18: print("karta je 5kn") elif a>65: print("karta je 1kn") else: print("karta je 8kn") </pre>
--	--

Primjer_2:

Unesite vrijednosti za tri stranice trokuta i usporedite da li je trokut jednakostraničan ili raznostraničan ili jednakokračan	<pre>a=int(input("Vrijednost stranice a: ")) b=int(input("Vrijednost stranice b: ")) c=int(input("Vrijednost stranice c: ")) If a==b and a==c: print("Jednakostraničan") elif a==b or a==c or b==c: print("Jednakooračan") else: print("Raznostraničan")</pre>
---	--

Primjer_3:

Napišimo program koji će nam omogućiti unos godine i ispisati da li je unesena godina prijestupna ili ne.	<pre>god=int(input("Unesi godinu ")) If (g%4==0 and g%100 !=0) or (g % 400==0): print("Prijestupna") else: Print("Nije prijestupna")</pre>
--	--

Primjeri zadataka za provjeru znanja:

Pitanje 1.

Što će biti rezultat izvršavanja sljedećeg programskog koda?

```
semafor = 'plavo'
if (semafor == 'zeleno'):
    print('pređi ulicu')
if (semafor == 'crveno'):
    print('ne možeš preći ulicu')
```

- a) Ispisat će se tekst "ne možeš prijeći ulicu" .
- b) Ispisat će se tekst "prođi ulicu".
- c) Ispisat će se tekst "plavo".
- d) Nijedan od ponuđenih odgovora nije točan.

Pitanje 2. *

Što će biti rezultat izvršavanja sljedećeg programskog koda?

```
a = -1
b = -1
if (a > b):
    print(a)
else:
    print(b)
```

Odgovor:

Pitanje 3. *

Napišimo program koji će tražiti da unesemo cijene dvije vrste sladoleda ("čokolada", "jagoda") ovisno o unesenoj cijeni ispisat ćemo koji je od njih skuplji, ako je skuplja čokolada ispisat će se "čokolada", ako je skuplja jagoda ispisat će se "jagoda".

```
cokolada=int(input('Unesi cijenu za čokoladu'))
jagoda=int(input('Unesi cijenu za jagodu'))
if (cokolada>jagoda):
    print("čokolada")
else:
    print("jagoda")
```

- a) čokolada
- b) jagoda
- c) Ispisat će se imena oba sladoleda.
- d) Program neće ispisivati nikakvu poruku.

Pitanje 4. *

Što bi ste od ponuđenog uzeli i stavili u uvjet IF da bi ste dobili ispravan rezultat

```
a = int(input('Unesite jedan broj'))
if ( ):
    print('Broj je djeljiv sa 5')
else:
    print('Broj nije djeljiv sa 5')
```

- a) $a \% 5 == 0$
- b) $a / 5 == 0$
- c) $a.5 == 0$
- d) Niti jedan od gore navedenih odgovora nije točan.

Pitanje 5.

Što će se ispisati nakon izvršenja sljedećeg koda?

```
a = 8
print( (a < 6) or (a > -10) )
```

- a) Istina
- b) Netočno

Pitanje 6.

Što će biti rezultat izvršavanja sljedećeg koda?

```
a = 2
b = 62
if ((a >= 10) or (b <= 70)) and (a + b > 50):
    print(a - b)
else:
    print(2 * a - b)
```

Odgovor:

Pitanje 7.

Što će biti rezultat programa nakon izvršenja sljedećeg koda?

```
a = 17
print( (a < 6) and (a > -10) )
```

- a) Istina
- b) Netočno

Pitanje 8.

Što će se ispisati nakon izvršenja sljedećeg koda?

```
a = 8  
print( (a < 6) or (a > -10) )
```

- a) Istina
- b) Netočno

Pitanje 9.

Što će biti rezultat izvršavanja sljedećeg koda?

```
a = 2  
b = 62  
if ((a >= 10) or (b <= 70)) and (a + b > 50):  
    print(a - b)  
else:  
    print(2 * a - b)
```

Odgovor:

Pitanje 10.

Cijena autobusne karte iznosi 65 Kn. Za djecu(stariju od 7 godina i ne stariju od 20 godina) i penzionere (ne mlađe od 65 godina), odobrava se popust od 10 Kn. Napišite program gdje će ovisno u unesenoj godini života ispisati i cijena karte.

Kojim uvjetom bi ste nadopunili programski kod

```
godine = int(input("Unesi koliko imaš godina"))  
cijena = 660  
if (_____):  
    cijena = 660 - 100  
print(cijena)
```

- a) (godina > 7 and godina <= 20) or (godina > 65)
- b) (godina > 7 and godina <20) or (godina > 65)
- c) godina > 7 and godina <= 20 or godina > = 65
- d) godina > 7 and godina <20 or godina > 65

Pitanje 11. *

Što će se ispisati prilikom izvršavanja sljedećeg koda ?

```
bodovi = int(input('Unesi broj bodova postignutih na ispitu'))
if bodovi > 85:
    o = 5
elif bodovi > 70:
    o = 4
elif bodovi > 55:
    o = 3
elif bodovi > 39:
    o = 2
else:
    o = 1
print(o)
```

ako se kao broj bodova unese broj 89 .

Pitanje 12.

Što će se ispisati prilikom izvršavanja sljedećeg koda?

```
a = 2
b = 10
if (a + b > 10):
    print(a * a)
elif (a + b == 10):
    print(a-b)
else:
    print(b)
```

Odgovor:

Pitanje 13.*

Što će se ispisati prilikom izvršavanja sljedećeg koda?

```
a = -10
b = -8
c = -1
if (c > 10):
    print(a * a)
elif (a + b > 10) or (b % 2 == 0):
    print(a - b)
else:
    print(b)
```

Odgovor:

UVOD U Petlje ili iteracije

Prošli smo put radili selekcije, odnosno grananje programa u ovisnosti o nekom uvjetu ili stanju u programu. Danas ćemo obraditi iteracije (petlje, ponavljanje dijela programa), koje su vrlo jednostavne.

Što je uopće petlja i čemu služi?

Petlja predstavlja dio programa koji možemo ponoviti određen broj puta. Na taj si način olakšavamo posao i nemamo potrebe pisati neki dio programa više puta. Ako vam je lakše, možete umjesto riječi iteracija reći i petlja.

Najjednostavniji primjer petlje je onaj školski; napišite 5 rečenica „Volim programiranje!“. Iako je to relativno mali broj ponavljanja, nekima neće biti problem i napisat će pet redova i u svakom ispisati rečenicu s pomoću funkcije `print()`.

Međutim što ako umjesto broja 5 piše broj 50?

Naravno, možete se koristiti copy/paste metodom i brojiti redove, ali čemu se mučiti i filozofirati? Upravo zato upotrebljavamo iteracije.

Postoji više iteracija a mi ćemo naučiti dvije (`for` i `while`) iteraciju, odnosno iteraciju s brojačem (`for`) i iteraciju s uvjetom (`while`).

Prije nego što krenemo rješavati zadatke, hajde da na ovom prethodnom primjeru prikažemo upotrebu iteracija. Prvo da vidimo kako bi to izgledalo s pomoću `for` iteracije:

```
for broj in range(1,6):  
    print(broj)
```

Na početku vidimo ključnu riječ `for`, koja nam objašnjava da se radi o iteraciji s brojačem.

Nakon toga vidimo **VARIJABLU** `broj` koja će poprimati vrijednosti iz funkcije `range()`. Unutar funkcije `range()` vidimo vrijednosti 1 i 6 odvojene zarezom. Ovdje treba napomenuti da je prva vrijednost početna, a druga završna, s tim da se ponavljanje izvršava do završne, a nju ne uključuje.

U ovom bi primjeru varijabla `broj` u prvom koraku imala vrijednost 1 i zatim bi se s pomoću funkcije `print()` ispisao broj 1. Nakon toga se opet vraćamo na početak iteracije i varijabla ima vrijednost 2 te se opet to ispisuje. I tako sve do vrijednosti 6, gdje brojač dobiva upravo tu vrijednost. Budući da prema funkciji `range()` taj broj ne ulazi u interval, ne ulazi se u iteraciju, nego se izlazi van. I na taj način smo ispisali brojeve od 1 do 5.

Kod iteracija s uvjetom, odnosno iteracije `while`, obavezni smo sami napraviti brojač jer se u zaglavlju iteracije zapravo ispisuje uvjet, to jest, u ovom slučaju, je li naš brojač manji od 6, a to izgleda ovako:

```
broj = 1  
while broj < 6:  
    print(broj)  
    broj = broj + 1
```

Za početak, prije same iteracije, moramo napraviti varijablu `broj` u kojoj se nalazi naša početna vrijednost, u ovom slučaju to je broj 1. Nakon toga slijedi ulazak u iteraciju. Vidimo ključnu riječ `while`, koja označuje početak `while` iteracije, a nakon toga slijedi uvjet gdje se u slučaju ako je uvjet zadovoljen ulazi u iteraciju, a u suprotnome izlazi iz nje.

Pri prvom koraku iteracije vrijednost brojača je 1 i to je manje od 6, pa se ulazi u iteraciju i ispisuje vrijednost brojača. Nakon toga se brojaču povećava vrijednost za 1.

Ovo je vrlo važno jer bi u protivnom vrijednost brojača ostala 1, tj. bila bi konstanta i u tom slučaju bi uvjet uvijek bio zadovoljen i petlja bi se stalno izvodila što nazivamo beskonačnom petljom. Ipak, ovdje smo povećali vrijednost varijable brojača koja se izvršava sve dok je brojač manji od 6. Kada vrijednost brojača dođe do broja 6, dolazimo do zaglavlja iteracije, ali uvjet nije ispunjen pa se zbog toga izlazi iz petlje.

PETLJA FOR

Kao što smo do sada naučili to je konačna petlja koja ima svoj početak i kraj.

Petlja **for** ima više oblika i prikazat ćemo sve mogućnosti.

U petlji **for** brojevi i rezultati obrade uvijek se ispisuju u stupcu tj. jedan ispod drugoga

Oblik_1: Imamo definiranu početnu i završnu vrijednost

for a in range(1,6): print(a)	a – predstavlja brojač koji nam poprima vrijednost za jedan više svaki puta kada prolazi kroz petlju do konačno broja 6. koji je naveden u funkciji range(1,6) . Naredbe koje nalaze u petlji obavezno moraju biti uvučene, to smo učili kod selekcija
--	---

Oblik_2: Definiramo početnu vrijednost, završnu vrijednost i korak uvećanja

for a in range(1,6,2): print(a)	(1,6,2) – Prvim prolaskom kroz petlju ispisat će se broj 1, drugim broj 3, sljedećim prolaskom broj 5. Ispisat ćemo neparne brojeve od 1 do 6. Što ako želimo ispisati parne brojeve ? Stavit ćemo samo range(0,6,2)
--	--

Oblik_3: Ispis brojeva od većeg prema manjem

for a in range(6,1,-1): print(a)	(6,1,-1) – plavi -1 – predstavlja da idemo u natrag od većeg prema manjem (5,4,3,2,1) Razmislite !! Što bi ste promijenili da ispišete neparne brojeve od 6 do 1, a što da ispišete parne brojeve od 6 do 1??
---	---

Oblik_4: Ispis brojeva u retku

<pre>for a in range(1,6): print(a,end=" ")</pre>	<p><code>end=""</code> – omogućuje ispis rezultata obrade u retku.</p> <p>Važno: ako stavite razmak između navodnika, dobit ćete i lijepi razmak između brojeva u retku .</p> <p>Stavite li između navodnika (- ili ,) brojevi će biti međusobno odijeljeni tim znakom.</p>
--	---

Oblik_5: Petlja for i zamjena naredbe `range` sa `in` i `not in`

<pre>a="sunce" for i in a: print(i)</pre>	<p>Umjesto <code>range</code> , koristimo ključnu riječ <code>in</code>.</p> <p><code>In</code> – provjerava da li se ono što tražimo nalazi u stringu.(znakovnoj varijabli).</p>
---	---

Primjeri s rješenjima **petlja for** i **petlja while**

Primjer_1:

Napiši program koji 5 puta ispisuje tvoje ime pomoću petlje for	for i in range(5): print ("Petar")
	a=0 while a<5: print("Petar") a=a+1

Primjer_2:

Napisi program koji 4 puta ispisuje tvoje ime pomoću petlje for i zatim jednom tvoje prezime ispod svih imena.	for i in range(4): print ("Petar") print ("Petrović") Pojašnjenje: Ime želimo ispisati četiri puta i ono mora biti u petlji , zato je uvučeno. Prezime je van petlje i zato nije uvučeno.
	a=0 while a<5: print("Petar") a=a+1 print("Petrović")

Primjer_3:

Napiši program koji pomoću petlje for ispisuje sve parne brojeve od 40 do 60 jedan pored drugoga.	for i in range(40,61): if i%2==0: print(i, end = " ") Važno: Obratite pažnju na uvlake , if je u petlji i zato je uvučen, a ispis je u okviru selekcije if i uvučen je u okviru nje. tj. dvostruko je uvučen u odnosu na petlju for.
	a=40 while (a>39) and (a<=60): if a%2==0: print(a,end=" ") a=a+1

Primjer_4:

Napiši program koji pomoću petlje for ispisuje brojeve djeljive s pet od 100-150 jedan pored drugoga.	for i in range(100,151): if i%5==0: print(i, end = "")
	a=100 while (a>99) and (a<=150): if a%5==0: print(a,end=" ") a=a+1

Primjer_5:

Napiši program za ispis brojeva od 1 do 10, ali bez broja 6.	for i in range(1,11): if i!=6: print(i, end = "")
	a=0 while a<11: if a!=6: print(a,end=" ") a=a+1

Primjer_6:

Napiši program koji ispisuje slovo po slovo, jedno ispod drugoga, pomoću naredbe for riječ 'sunce'	a="sunce" for i in a: print(i)
--	--------------------------------------

Primjer_6a:

Napiši program koji za upisanu riječ ispisuje samo samoglasnike.	a=input("Upiši riječ:") for i in a: if i in "aeiouAEIOU": print (i)
--	--

Primjer_7:

Proširi prethodni program tako da umjesto samoglasnika program prebroji i ispiše broj samoglasnika u rijeci.	a=input("Upiši riječ:") sam=0 for i in a: if i in "aeiouAEIOU": sam=sam+1 print ("Broj samoglasnika:", sam)
Važno: U ovom primjeru za korištenje brojača moramo definirati početnu vrijednost sam=0 i uvećanje te vrijednosti za 1 sam=sam+1	

Primjer_8:

Zbrajanje u petlji FOR <i>Napišite program koji ispisuje zbroj parnih brojeva od 1 do 20.</i>	<pre>zbroj = 0 for i in range (0, 21, 2): zbroj = zbroj + i print ('Nakon broja', i, 'zbroj je', zbroj)</pre>
	<pre>a=0 zbroj=0 while a<21: if a%2==0: zbroj=zbroj+a print(" Zbroj brojeva je",zbroj) a=a+1</pre>

VAŽNO:

U petlji WHILE , brojač može ići i unatrag Primjer: Zadatak_2: Za ispis brojeva od 8 do 1	<pre>a=8 while a>=1: print(a) a=a-1</pre>
--	--

Zadaci za vježbu: Pokušajte zadatke riješiti pomoću *petlje FOR* i pomoću *petlje WHILE*

Zadatak_1:

Ispiši samo parne brojeve do broja deset, ali i broj deset.

Zadatak_2:

Ispiši prvih osam brojeva u obrnutom nizu (od 8 do 1)

- Preuredi program pa ispiši samo parne brojeve u obrnutom nizu od 8 do 1
- Preuredi program pa ispiši samo neparne brojeve u obrnutom nizu od 8 do 1

Zadatak_3:

- Ispišite prvih 10 brojeva , ali bez broja 6
- Ispišite prvih 20 brojeva ,ali bez broja 8 i broja 13 (sjetite se logičkih operatora)

Zadatak_4:

Napišite program koji ispisuje zbroj parnih brojeva od 1 do 20.

- Napišite program koji ispisuje zbroj neparnih brojeva od 20 do 1

Zadatak_5:

Unesi neki broj preko tipkovnice(input) i ispiši sve brojeve do tog broja, ali i taj broj

Zadatak_6:

Unesi neki broj (input). Napravite ispis svih parnih brojeva do broja koji ste unijeli

a) Ispi zadatak samo za neparne brojeve

Zadatak_7:

Unesite neki broj(input) i napravite ispis svih brojeva do tog broja u obrnutom redoslijedu.

PETLJE – provjera znanja

Pitanje 1.

Koji od ponuđenih naredbi treba dodati na označeno mjesto da bi program tri puta ispisao riječi "Dobar dan" .

```
for _____:  
    print("Dobar dan")
```

- A. i in range(3)
- B. i inrange(3)
- C. i in 3
- D. i = 3

Pitanje 2.

S kojom od ponuđenih linija koda , program NEĆE pet puta ispisati riječi "Dobar dan".

```
for _____:  
    print("Dobar dan")
```

Odaberite točan odgovor:

- A. i in range(1, 5)
- B. i in range (5)
- C. i in range(14, 19)

Pitanje 3. **

Što će biti rezultat sljedećeg programa, ako mu se na oba unosa unese broj 2 ?

```
for i in range(8):  
    o = int(input("unesi broj opravdanih"))  
    n = int(input("unesi broj neopravdanih"))  
    u = o + n  
    print(u)
```

- A. 28 puta bit će ispisan broj 4.
- B. 56 puta će se ispisati broj 2
- C. 28 puta će biti ispisan broj 2.
- D. Bit će ispisan broj 4

Pitanje 4.

Što će biti rezultat izvršenja sljedećeg programskog koda ?

```
for broj in range(1,5):  
    print(broj)
```

- A. Brojevi 1, 2, 3, 4 će biti ispisani.
- B. Ispisat će se broj 5.
- C. Brojevi 1, 2, 3, 4, 5 će biti ispisani.
- D. Nijedan od ponuđenih odgovora nije točan.

Pitanje 5. *

Što će biti rezultat izvršenja sljedećeg programskog koda ?

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

- A. Brojevi 1, 2, 3, 4 će biti ispisani.
- B. Brojevi 0, 1, 2, 3, 4, 5 će biti ispisani.
- C. Broj 5 će biti ispisan.
- D. Nijedan od ponuđenih odgovora nije točan.

Pitanje 6. *

Što će biti rezultat izvršenja sljedećeg programa, ako mu se pri pokretanju za vrijednost **a** unese broj 9 , a za vrijednost **b** broj 15?

```
a = int(input("Unesi a"))  
b = int(input("Unesi b"))  
for i in range(a,b+1,8):  
    print(i)
```

Pitanje 7. *

Što od ponuđenog bi ste unijeli u programski kod da se ispišu svi dvoznamenkasti brojevi koji su djeljivi sa 3,

```
for i in range(____):  
    print(i)
```

- A. 10, 100, 3
- B. 12, 100, 3
- C. 12, 99, 3
- D. 10, 99, 3

Pitanje 8. *

U igri skrivača brojimo do 200 ali po pet. Koju od ponuđenih linija koda bi ste iskoristili u programu za ispis brojeva do 200, uključujući i 200 s korakom od po 5.

```
for i in range(_____):  
    print(i)
```

- A. 0, 200, 5
- B. 0, 201, 5
- C. 5, 201, 5
- D. 5, 200, 5

Pitanje 9. **

Što će biti rezultat izvršenja sljedećeg programskog koda ?

```
s = 0  
for i in range(6):  
    s = s + i  
print(s)
```

Odgovor:

Pitanje 10. **

Što će biti rezultat izvršenja sljedećeg programskog koda?

```
p = 0  
for i in range(4):  
    p = p * i  
print(p)
```

Odgovor:

UVOD U GRAFIČKI MODUL PYTHON-a

Python je veliki i u posljednje vrijeme vrlo korišten za potrebe programiranja u raznim područjima života.

Da bi ste mogli koristiti pojedine module morate ih pozvati.

Što je to modul ?, naziv za skup naredbenih funkcija koje služe u razvoju programa.

Prvi modul koji ćemo upoznati je grafički modul, a do osmog razreda i modul math.

Krenimo!

Prije pisanja grafičkog programskog koda moramo ga pozvati i to je ujedno i naša prva linija programa.

Pozivamo ga sa: **from turtle import***

Naredbe koje ćemo koristiti su vrlo jednostavne i grafički modul ćemo povezati sa petljama kako bi vidjeli sve prednosti koje imamo koristeći petlje u procesu programiranja.

Naredbe:

fd(dužina linije)

rt(kut koji ide u smjeru kazaljke na satu- pojednostavljeno prema podu)

lt(kut koji ide u suprotnom smjeru od kazaljke sata – pojednostavljeno prema plafonu).

Vrijednost dužine koju možemo koristiti u okviru naredbe fd je 300 točkica.

Grafički ekran ima na raspolaganju ukupno 600 točkica po dužini i 600 točkica po visini. U ovom dijelu učili bi smo koordinativni sustav što je gradivo viših razreda pa da se ne bunimo. Bit će nam dovoljno 300 točkica.

Do sada ste iz matematike učili kuteve, njihove vrijednosti , crtali jednakokračne trokute, kvadrate, pravokutnike itd.

Da li ste zamijetili ???

Puni krug iznosi 360^0 , crtamo li jednakokračni trokut od 120^0 , kako smo došli do 120^0 ?. Jednostavno svaki geometrijski lik tj. vrijednost njegovih kuteva dobivamo tako da 360 podijelimo sa brojem kuteva.

Trokut $360/3=120$

Kvadrat $360/4=90$

Peterokut $360/5=72$

Itd.

Prvi naš zadatak:

Nacrtajmo kvadrat dužine stranica 100. Kut pretpostavljam da znate?

<pre>from turtle import* fd(100) rt(90) fd(100) rt(90) fd(100) rt(90) fd(100)</pre> <p>Programski kod je vrlo dugačak, pojednostavimo ga s petljama koje smo naučili.</p>	<pre>from turtle import* for a in range(5): fd(100) rt(90)</pre> <p>Kao što vidite petlja nam omogućuje znatno smanjenje programskog koda.</p>	<pre>from turtle import* a=0 while a<5: fd(100) rt(90) a=a+1</pre>
---	--	---

Zamislite da trebate nacrtati šesterokut , bez korištenja petlji. Kakav bi bio kod ?? . pogledajte s petljama !.

<p>Pogledajte kako jednostavno!!</p>	<pre>from turtle import* for a in range(7): fd(100) rt(360/6)</pre> <p>Kao što vidite petlja nam omogućuje znatno smanjenje programskog koda.</p>	<pre>from turtle import* a=0 while a<7: fd(100) rt(360/6) a=a+1</pre>
---	---	--

Pravo i vidljivo značenje petlji u postupku stvaranja programa.

Pogledajmo dvije nove naredbe pensize() i pencolor() , samo ime nam govori čemu služe . Debljina olovke , boja olovke.

Primjeri programa:

Primjer_1:

<p>Nacrtajmo šesterokut debljine linije 6. i plave boje.</p>	<pre>from turtle import* pensize(6) pencolor("blue") for a in range(7): fd(100) rt(360/6)</pre> <p>Kao što vidite petlja nam omogućuje znatno smanjenje programskog koda.</p>	<pre>from turtle import* pensize(6) pencolor("blue") a=0 while a<7: fd(100) rt(360/6) a=a+1</pre>
---	---	--

Primjer_2:

<p>Nacrtajmo šesterokut tako da svaka linija za jedan bude deblja od prethodne.</p>	<pre>from turtle import* pencolor("blue") for a in range(7): pensize(a) fd(100) rt(360/6)</pre> <p>Kao što vidite, pensize() je u petlji. Zašto ??</p>	<pre>from turtle import* pencolor("blue") a=0 while a<7: pensize(a) fd(100) rt(360/6) a=a+1</pre>
--	---	--

Primjer_3:

<p>Nacrtajmo šesterokut debljine linije 6 i neka bude ispunjen crvenom bojom.</p> <p>Za ovaj primjer moramo koristiti blok naredbi</p> <p>fillcolor("red") - boja ispune begin_fill() – označavamo početak dijela za bojanje end_fill() kraj dijela za bojanje</p>	<pre>from turtle import* pensize(6) fillcolor("red") begin_fill() for a in range(7): fd(100) rt(360/6) end_fill()</pre>	<pre>from turtle import* pensize(6) a=0 fillcolor("red") begin_fill() while a<7: fd(100) rt(360/6) a=a+1 end_fill()</pre>
--	---	--

Zadatak_1:

**Nacrtajte peterokut tako da svaka linija bude za jedan tanja od prethodne.
peterokut ispunite zelenom bojom.**

Nadam se da ste uživali i naučili pravo značenje petlji i njihovoj primjeni u programiranju,
nadam se da ste svjesni koliko nam olakšavaju rad !!!

STRINGOVI

Rad sa stringovima

Python poznaje nekoliko vrsta tipova podataka kao što su npr:

- **string** (niz znakova, piše se uvijek unutar navodnika)
- **int** (cijeli broj)
- **float** (decimalni broj)

Do sada smo uspješno radili sa tipovima podataka int i float

String tj. niz znakova definiramo **unutar navodnika**.

```
a="Pero"
```

```
b="123"
```

Stringove možemo **spajati**.

```
a="Pero"
```

```
b="123"
```

```
print(a+b)
```

Rješenje: Pero123

```
a="123"
```

```
b="456"
```

```
print(a+b)
```

Rješenje: 123456

String ne možemo spojiti s drugim tipom podataka npr. int.

```
a="123"
```

```
b=456
```

```
print(a+b)
```

Rješenje:

TypeError: Can't convert "int" object to str implicitly

Ako želimo izvršiti istu operaciju nad dvije različite vrste podataka, potrebno ih je pretvoriti u istu vrstu podataka. String pretvaramo u cijeli broj(int) pomoću naredbe **int()**.

```
a="123"  
b=456  
  
print int(a)+b
```

Rješenje:

579

Svaki znak u stringu ima svoje mjesto koje nazivamo **indeks**. Mjesto tj. indeks definiramo pomoću **[]** **zagrade**.

```
a="Pero"  
  
print a[0]
```

Rješenje:

P

Objašnjenje: Python mjesta broji od 0. U riječi Pero P ima indeks 0, e ima indeks 1, r ima indeks 2, o ima indeks 3.

P	E	R	O
[0]	[1]	[2]	[3]
[-4]	[-3]	[-2]	[-1]

Indeks možemo pregledavati i "unazad", u tom slučaju zadnje slovo ima indeks **-1**, predzadnje -2 itd.

```
a="Pero"  
  
print a[-1]
```

Rješenje: o

Neke osnovne naredbe za rad sa stringovima

Naredba	Objašnjenje	Primjer	Rješenje
len()	Duljina stringa	a="Pero" print len(a)	4
max()	Vraća najveći element stringa	a="56392" print max(a)	9
min()	Vraća najmanji element stringa	a="56392" print min(a)	2
a.replace()	Promijeni slova	a="Informatika" b=a.replace("a","o") print b	informatiko
a.count()	Prebroji određeno slovo	a="informatika" b=a.count("i") print b	2
a.endswith()	Provjeri s kojim znakom završava riječ		

Primjena gotovih funkcija u postupku programiranja

Primjer_1:

Prosječna ocjena

U listu su unesene ocjene iz nekoliko predmeta. Izračunajte prosječnu ocjenu.

```
ocjene = [5, 4, 5, 3, 5, 4, 4, 5]
```

```
prosjek = sum(ocjene) / len(ocjene)
```

```
print(prosjek)
```

Primjer_2:

Najniža ocjena

Ako znamo da su na listi tri posljednje ocjene iz prirodnih znanosti. Izračunajte najnižu ocjenu iz prirodnih znanosti.

```
ocjene = [5, 4, 5, 3, 5, 4, 4, 5]
ocjene_iz_prirodnih_znanosti = ocjene[-3:]
print(min(ocjene_iz_prirodnih_znanosti))
```

Razlika temperature

Primjer_3:

Vrijeme se često mijenja i u tijeku jednog tjedna izmjenjuju se topli i hladni dani.

Ako imamo unesene temperature za jedan tjedan, utvrdite razliku između najviše i najniže temperature.

```
temperatura = [17, 23, 12, 15, 19, 21, 25]
print(max(temperatura) - min(temperatura))
```

Primjer_4:

Visine djevojčica i dječaka u razredu

Poznate su visine djevojčica i dječaka u jednom razredu.

Napravite ispis , koliko ste visina unijeli tj. Koliko ima u tom razredu djevojčica i dječaka i izračunajte prosječnu visinu svih učenika.

```
visine_djevojica = [165, 153, 155, 155, 157]
```

```
visine_djecaka = [170, 168, 173, 156, 172]
```

```
visine = visine_djevojica + visine_djecaka
```

```
print(len(visine))
```

```
print(sum(visine) / len(visine))
```

Primjer_5.

Navedene su cijene tri proizvoda. Ako kupite sva tri proizvoda jedan od tih proizvod dobivate za 1 Kn. Koliko ćete platiti sve proizvode ?

```
cijene = [1420, 1799, 1569]
```

```
cijene_po_redu = sort(cijene)
```

```
cijene_po_redu[0] = 1
```

```
print(sum(cijene_po_redu))
```

Priimjer_6.

U listi imamo unesene cijene nekoliko proizvoda. Izračunajte sumu tri najjeftinija i tri najskuplja proizvoda.

```
cijene = [58.00, 104.95, 117.50, 11.95, 10.4, 37.95, 85.5]
```

```
sortirane_cijene = sort(cijene)
```

```
print("najjeftiniji",sum(sortirane_cijene[0:3]))
```

```
print("najskuplji",sum(sortirane_cijene[-3:]))
```

Primjeri zadataka s rješenjima:

1. Napiši program koji unosi ime i ispisuje poruku je li upisano ime muško ili žensko. Ako ime završava na a, ispisat će se poruka "Žensko ime", inače će se ispisati poruka "Muško ime."

```
a= input ("Upisi ime:")
if a[-1]=="a":
    print ("Žensko ime")
else:
    print ("Musko ime")
```

2. Napiši program koji dopušta unos dvije riječi i ispisuje poruku koja riječ je duža.

```
a= input ("Upisi prvu riječ: ")
b= input ("Upisi drugi riječ: ")
if len(a)>len(b):
    print ("Riječ",a,"je dulja.")
else:
    print ("Rijec",b,"je dulja.")
```

3. Napiši program koji unosi rečenicu i ispisuje istu tu rečenicu u kojoj je svako slovo a promijenjeno u slovo o.

```
a= input ("Upisi recenicu: ")
b=a.replace("a","o")
print (b)
```

4. Napiši program koji unosi rečenicu u 3.licu jednine prezenta engleskog jezika i ispisuje ju u prošlom jeziku.

```
a= input ("Upisi recenicu: ")
b=a.replace("is","was")
print (b)
```

5. Napiši program koji unosi riječ i ispisuje poruku "Kalodont!" ako riječ završava na "ka". U protivnom će se ispisati poruka "Dalje".

```
a= input ("Upisi rijec: ")
if a.endswith("ka"):
    print( "Kalodont!")
else:
    print ("Dalje")
```

6. Napiši program koji unosi riječ i ispisuje poruku je li upisana riječ muškog, ženskog ili srednjeg roda. Ako riječ završava na o, ispisat će se poruka "srednji rod", ako riječ završava na "a" ispisat će se poruka "ženski rod". U svim ostalim slučajevima ispisat će se poruka "muški rod".

```
a= input (" Upisi riječ: ")
if a[-1]=="o":
    print( "Srednji rod" )
elif a[-1]=="a"
    print( "Ženski rod" )
else:
    print ("Muški rod")
```

Zadaci za vježbu :

Zadatak_1:

Napiši program koji provjerava duljinu lozinke. Ako upisana lozinka ima manje od 8 znakova, ispisat će se poruka: Slaba lozinka. Ako lozinka ima 8 ili više znakova ispisat će se poruka: Jaka lozinka.

Zadatak_2:

Napiši program koji unosi korisničko ime te provjerava je li korisničko ime upisano ili ne. Ukoliko je ispisat će pozdravnu poruku, a ako nije ispisat će poruku: "Niste upisali korisničko ime."

*Podsjetnik kod definiranja unosa stringova (Znakova) koristimo samo naredbu input bez int ili float
Da li nešto je ili nije dovoljni je u funkciju if staviti prazno mjesto if== "", prazna mjesta su dva navodnika.*

Zadatak_3:

Napiši program koji unosi tvoju najdražu riječ. Ispiši koliko slova ima ta riječ te poruku je li riječ duga ili kratka. Riječ je duga ako ima 7 ili više slova.

PRIMJERI ZADATAKA:

Pitanje 1.

Koji uvjet bi ste stavili u naredbu IF da bi ste dobili rješenje programa, ako je broj u listi ispisat će se da je u listi, ako nije ispisat će se da nije u listi

```
lista = [2,27,35,17]
a = int(input("Unesi jedan broj"))
if ( ):
    print("Uneseni broj se nalazi u listi")
else:
    print("Uneseni broj se ne nalazi u listi")
```

- A. a in lista
- B. len(a)>2
- C. a exist in lista

Pitanje 2.*

Što će biti rezultat izvršenja sljedećeg koda?

```
lista = ["Edita", "Ana", "Petra"]
for ime in lista:
    print(ime)
```

- A. Ispisat će se imena Edita, Ana i Petra.
- B. Bit će ispisana riječ lista.
- C. Ispisuju se brojevi 0,1 i 2.
- D. Tri puta će se ispisati riječ "ime"

Pitanje 3. *

Što će biti rezultat izvršenja sljedećeg programskog koda?

```
ljubimci = ["pas", "mačka", "papagaj", "zec", "kornjača"]
for i in [1,3]:
    print(ljubimci[i])
```

- A. Ispisat će se riječi mačka i zec.
- B. Napisat će se riječi "pas" i "papagaj".
- C. Napisat će se riječi "pas, mačka i papagaj".

Pitanje 4. *

Što će bit rezultat sljedećeg programskog koda?

```
ljubimci = ["pas", "macka", "papagaj", "zec", "kornjaca"]  
for i in range(3):  
    print(ljubimci[i])
```

- A. Pisat će se riječi pas, mačka i papagaj.
- B. Napisat će se riječi pas, mačka, papagaj i zec.
- C. Papagaj će se ispisati.
- D. Ispisat će se riječ zec.
- E. Nijedan od ponuđenih odgovora nije točan.

Pitanje 5.

Što bi ste od ponuđenih naredbi umetnuli u naredbu for , da bi ispisali sve članove liste ljubimci.

```
for i in range(_____):  
    print(ljubimci[i])
```

- A. 1, len (kućni ljubimci)
- B. len (kućni ljubimci) - 1
- C. ljubavnici
- D. Nijedan od ponuđenih odgovora nije točan.

Pitanje 6.

Što će biti rezultat izvršenja sljedećeg programskog koda?

```
lista = [-3, 5, 16, 5, -2]  
s = 0  
for x in lista:  
    s = s + x  
print(s)
```

Pitanje 7.

Što će biti rezultat izvršenja sljedećeg programskog koda?

```
lista = [22, 20, 21, 24, 15, 19]  
s = 0  
for x in lista:  
    if x % 2 == 0:  
        s = s + x  
print(s)
```

Odgovor:

Pitanje 8. **

Koja od navedenih tvrdnji je točna ?

```
r = []
```

```
slova = "abcdefghijklmnoprstuvz"
```

```
for x in slova:
```

```
    r[x] = 0
```

- A. Stvorena je lista s malim slovima abecede.
- B. Stvorena je lista s malim slovima abecede kojima je pridružena vrijednost 0
- C. Program je pogrešno napisan i javit će pogrešku