

PYTHON

----- PRVI KORACI -----



RADNA VERZIJA 0.824

Dino Isanović

sekcijainfor@gmail.com

NOVU VERZIJU KNJIGE I PROGRAMA ZA
VJEŽBU MOŽETE NAĆI NA OVOM LINKU

[https://github.com/Dinolsanovic/Python-
prvi-koraci](https://github.com/Dinolsanovic/Python-prvi-koraci)

Šta nam je potrebno da bi smo pisali programe u Python programskom jeziku.

1. Potrebno je instalirati programski jezik python na naš računar. [film za instalaciju python programskog jezika i radnog okruženja VS Code](#)

2. Alternativni editor za pisanje python programa je "Pycharm" pisan specijalno za Python programski jezik.

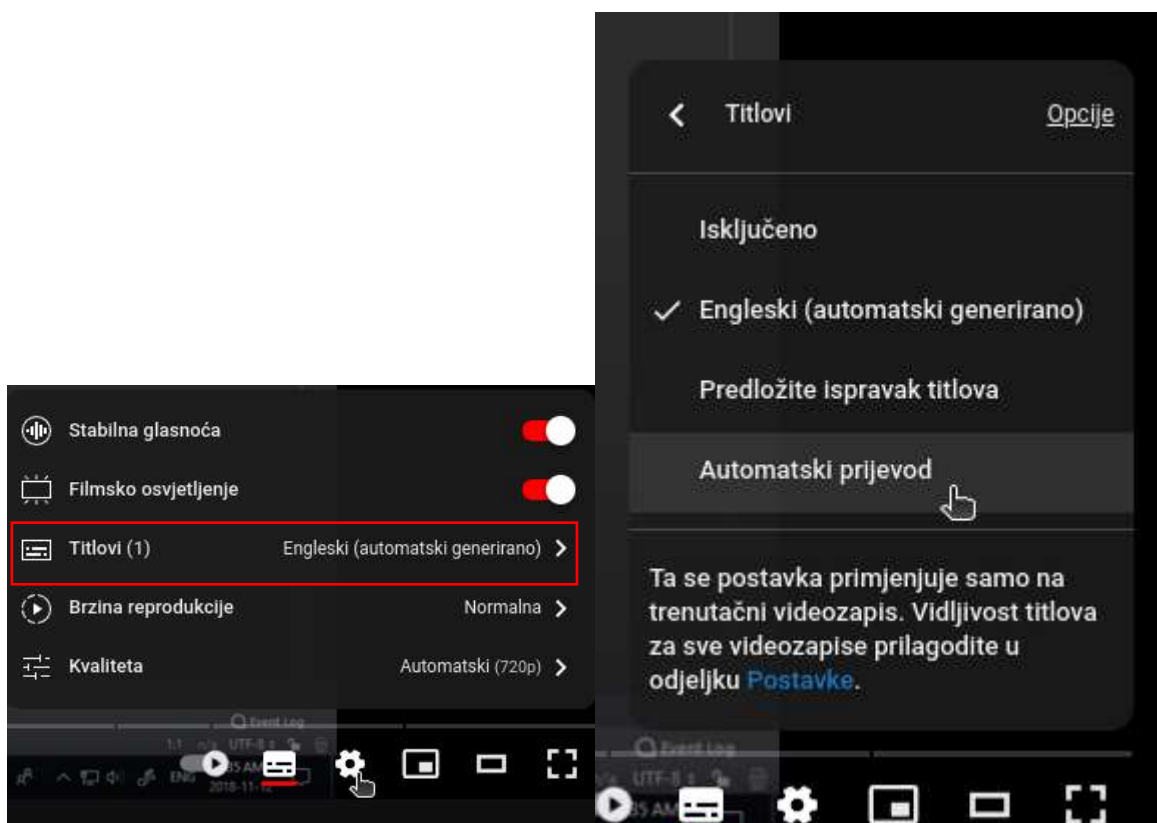
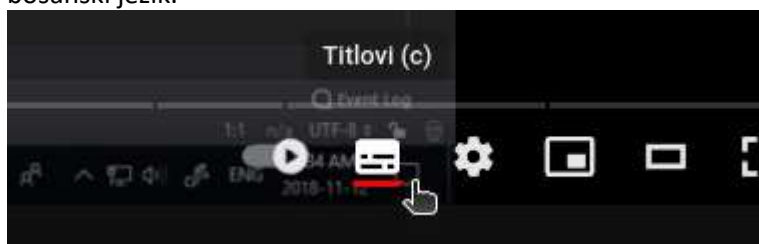
1. [Film za instalaciju Python jezika i Pycharm editora](#)

2. [Video uputstvo za korištenje Pycharm editora](#)

Izbor programa za pisanje programa u python-u je po slobodnom izboru. Na gore navedenim linkovima se nalaze i kompletni video tutorijali za učenje programiranja u python programskom jeziku. Link za kompletan tutorijal je ispod.

<https://www.youtube.com/watch?v=OFrLs22MDAw&list=PLzMcbGfZo4-mFu00qxl0a67RhjjZj3jXm>

Prilikom gledanja filmova na engleskom jeziku potrebno je uključiti titlove i prevod titlova na bosanski jezik.



Odabrati bosanski jezik u ponuđenoj listi jezika koji će se automatski prevoditi.

Za učenje uz pomoć video tutorijala preporučujem korištenje drugog monitora tako da na jednom monitoru gledate film a na drugom monitoru pišete kod ili koristite neku aplikaciju.

NAREDBE UNOSA I ISPISA

PRINT

Naredba za ispisivanje poruka i vrijednosti varijabli na ekranu

računara. Naredba za ispis aktivira se ključnom riječi **print**. Nakon te riječi otvaraju se i zatvaraju zagrade. Unutar zagrada, moguće je ispisati tekst, ali i vrijednosti varijabli. **Tekst koji želimo ispisati na ekranu uvijek se stavlja unutar navodnika.**

Print („pozdrav“)

Kada računar pokrene ovaj jednostavan program izvršit će samo jednu instrukciju koja će na ekranu računara ispisati poruku koja se nalazi između znakova navodnika. U našem slučaju će se pojaviti riječ pozdrav. Nakon čega će se završiti izvođenje programa koji je imao samo jednu komandu print.

Kako bi smo naredili računaru da ispiše tekst “kraj igre”?



Šta je varijabla u odnosu na obični tekst koji ispisujemo na ekranu. Varijabla je sadržaj memorije računara koji će nestati tek kada se program završi. A tekst „pozdrav“ koji smo ispisali je samo parametar funkcije (komande) *print*. Taj tekst neće biti zapamćen u memoriji računara.

Za izvođenje kompleksnih programa potrebni su nam podaci koje pamte VARIABLE . Zbog toga su varijable jedan od navažnijih dijelova programa. Kada igramo neke igrice u kojima imamo neku životnu energiju “health” ta vrijednost se pamti u nekoj varijabli i kada vrijednost te varijable padne na nulu tada nam igrica to jest program ispiše poruku da je kraj igre (Game over).

Varijabla je rezervisana memorijska lokacija koja pamti podatke i rezultate koji su potrebni za rad programa.

Svaka varijabla ima **svoje ime** i **svoju vrijednost**. Mi u programu strogo koristimo samo ime varijable. A za vrijeme izvršavanja programa računar umjesto imena u svojim operacijama koristi vrijednost varijable. Da bi smo u memoriju računara upisali neku vrijednost moramo rezervisati mjesto pisanjem imena varijable te dodajemo operator dodjeljivanja “=” te upisujemo vrijednost varijable.

$a=10$ Ime varijable je “a” znak dodjeljivanja “=” a vrijednost varijable je 10
U prevodu na naš jezik to bi značilo varijabla “a” dobija vrijednost 10.

znak dodjeljivanja

a = **10**
ime varijable **vrijednost varijable**

Kada se ova instrukcija izvrši računar će rezervisati memorijsku lokaciju pod nazivom “a” a zatim će u tu lokaciju upisati broj 10.

Za pisanje jednostavnih programa bit će nam potrebne 2 vrste varijabli.

1. Znakovne varijable (slova, brojevi i specijalni znaci ^_^)
2. Numeričke varijable (brojevi)

Kako bi to izgledalo u programu koji u sebi sadrži 2 varijable, a je znakovna varijabla koja ima vrijednost 10 i b je numerička varijabla koja ima vrijednost 10

```
a="10" #vrijednost znakovnih varijabli se upisuju između navodnika
```

```
b=10 # vrijednost numeričke varijable
```

```
print (a+a) #komanda koja će na ekranu ispisati zbir vrijednosti ovih varijabli
```

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

```
# Kasnije u programu pokušajte sabrati varijable a i b
```

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

```
# komentarisati grešku koja se pojavi
```

Ovaj program proširite sa dodatnim komandama

```
a=b+b
```

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

```
print (a)
```



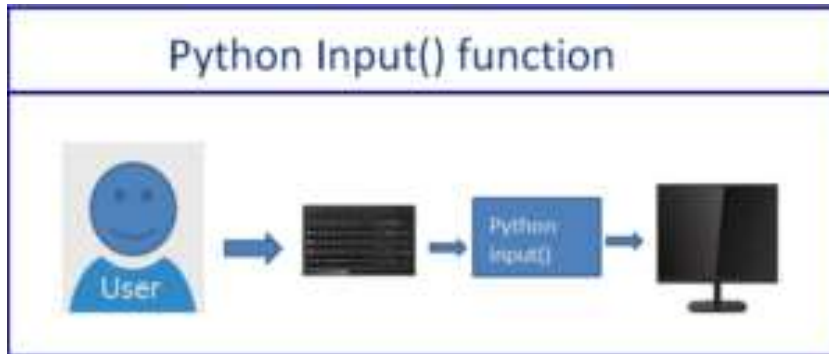
#komentarisati vrijednosti i greške koje ste dobili na ekranu.

Napisati program koji računa zbir brojeva 10 i 7 koristeći varijable

Komanda Print ima više oblika i formata ali za sada je ovo dovoljno.

INPUT

Komandu INPUT koristimo kada želimo da korisnik preko tastature unese vrijednost željene varijable (unos podataka koji su potrebni za rješenje problema).



VARIJABLA = tip podataka (input („tekst“))

Tip podataka označava koje kakve podatke pamti varijabla:
int – cijeli broj
float – decimalni broj
boolean – logička vr.

Prateći tekst obično daje upute korisniku koju podatak da unese.

```
a=tip podataka( input (“prateći tekst”))
```

Ime varijable

Komanda input koju koristimo za unos podataka preko tastature.

Primjeri:

Želimo da korisnik unese svoje ime :
`ime1= input(„unesite svoje ime“)`

Kao što ste vidjeli ovdje nismo naveli tip podataka jer kod unosa običnog teksta to nije potrebno. *lme1* je ime varijable koja će pamtit i ime unešeno preko tastature.

Kada radimo sa brojevima moramo razlikovati 2 vrste brojeva cjelobrojni i decimalni (`int` , `float`) što zahtjeva od nas da unaprijed znamo koje vrste brojeva ćemo koristiti. Ako niste sigurni možete koristiti tip `float`.

Cjelobrojne brojeve označavamo sa `int`

`A= int (input („unesite duzinu kvadrata“))` # kada računar izvrši ovu komandu čekat će da korisnik preko tastature unese vrijednost varijable A

Brojeve sa decimalnim zarezom označavamo sa „`float`“

`D1= float (input („unesite djeljenik“))`

`D2= float (input („unesite djelilac“))`

Kada dođe do izvršavanja funkcije `input` računar staje i čeka da korisnik unese vrijednost označene varijable i kada pritisne tipku ENTER u memoriju računara se zapisuje napisana vrijednost varijable. Koju mi koristimo u programu uz pomoć imena varijable.

Primjeri:

Unesite ova 2 programa i uočite razliku i komentarišite šta rade ovi programi.

```
a = int(input(„unesite vrijednost a“))
b = int(input(„unesite b: “))
s = a + b
print(s)
```

```
a = input()
b = input()
s = a + b
print(s)
```

***Napisati program koji računa zbir 2 broja

Prije izrade ovog programa moramo razmisliti koji podaci su nam potrebni za izradu ovog programa.

Kada u obzir uzmemo matematičku formulu zbira brojeva $z=a+b$ vidimo da su nam potrebna 2 broja kako bi dobili rezultat koji se od nas traži. Za svaki podatak u programu potrebna nam je varijabla koju ćemo imenovati na svoj način kako bi nas podsjećale na podatke koje one pamte. Pa ćemo nazvati:

prvi sabirak sa „a“

drugi sabirak sa „b“

zbir će biti izračunat i zapamćen pod varijablom „z“.

Svaki program ovog karaktera ima 3 faze

1. Unos podataka
2. Obrada podataka
3. Ispis rezultata



Znak # (hashtag) u programu ima funkciju pisanja komentara i nije obavezan ali kada pišete velike programe bit će vam potrebno da komentarišete pojedine djelove programa kako bi ste kasnije mogli pronaći željeni dio programa.

prilikom pokretanja svaki program bi trebao da se prijavi

šta radi i koja njegova funkcija te kratko uputstvo kako koristiti taj program

to ćemo uraditi korištenjem funkcije print

print ("PROGRAM RAČUNA ZBIR 2 BROJA")

1. KORAK UNOS PODATAKA POTREBNIH ZA DOBIJANJE REZULTATA

U OVOM SLUČAJU TREBAMO UNIJETI 2 BROJA KOJA ĆEMO NAZVATI a i b

UNOS PODATAKA VRŠIMO UZ POMOĆ FUNKCIJE INPUT

a=int(input("unesite prvi broj "))

b=int(input("unesite drugi broj"))

UNOS
PODATAKA

2. KORAK OBRADA PODATAKA

VRLO JEDNOSTAVNO POTREBNO JE UPISATI
ODGOVARAJUĆI MATEMATIČKI IZRAZ
(FORMULU)

c=a+b

OBRADA
PODATAKA

3. KORAK ISPIS REZULTATA U OVOM SLUČAJU VARIJABLA c PAMTI NAŠ REZULTAT

print ("zbir brojeva iznosi ", c)

varijable se nikada ne stavljaju pod znake
navodnika

ISPIS
REZULTATA

**** Program koji računa površinu i obim kvadrata :

predstavljanje programa

Print(„program računa površinu kvadrata“)

unos podataka

a = int(input(„unesite dužinu stranice a “))

obrada podataka

p = a2** *#kvadriranje varijable a može se pisati i p=a*a*

o= 4*a

ispis rezultata

print („površina kvadrata iznosi“)

print(p)

print („obim kvadrata iznosi“, o)

podaci se mogu ispisati i sa jednom komandom print

print („površina kvadrata iznosi „ , p)

kada se startuje program
će ispisati šta radi

Za računanje površine i
obima računaru treba preko
tastature unijeti dužinu
stranice a

Kada se izvrše računске
operacije varijable "p" i "o" će
dobiti neku brojčanu
vrijednost i bit će zapamćene

Ispis rezultata uz pomoć
komande print. Varijable nisu
pod znacima navoda. Što znači
da će biti ispisani brojevi

Zadaci :

- 1) Napisati program koji računa zbir 3 broja sa decimalnim zareзом (**float**).
- 2) Napisati program koji računa površinu pravougaonika.
- 3) Napisati program koji računa površinu i obim pravougaonika.
- 4) Napisati program koji računa jačinu struje u kolu koristeći Ohmov zakon $I=U/R$ (koristiti varijable koje mogu imati decimalni zarez **float**)

Komande za donošenje odluka u programu

Komanda IF (prosti oblik)

Svako inteligentno biće donosi odluke u zavisnosti od situacije u kojoj se nalazi uzimajući u obzir stvari iz svoje okoline. Tako kada pada kiša mi koristimo kišobran. U ovom slučaju da bi smo koristili kišobran treba da pada kiša. Znači „**kiša**“ je uslov za vršenje radnje „**nošenje kišobrana**“.

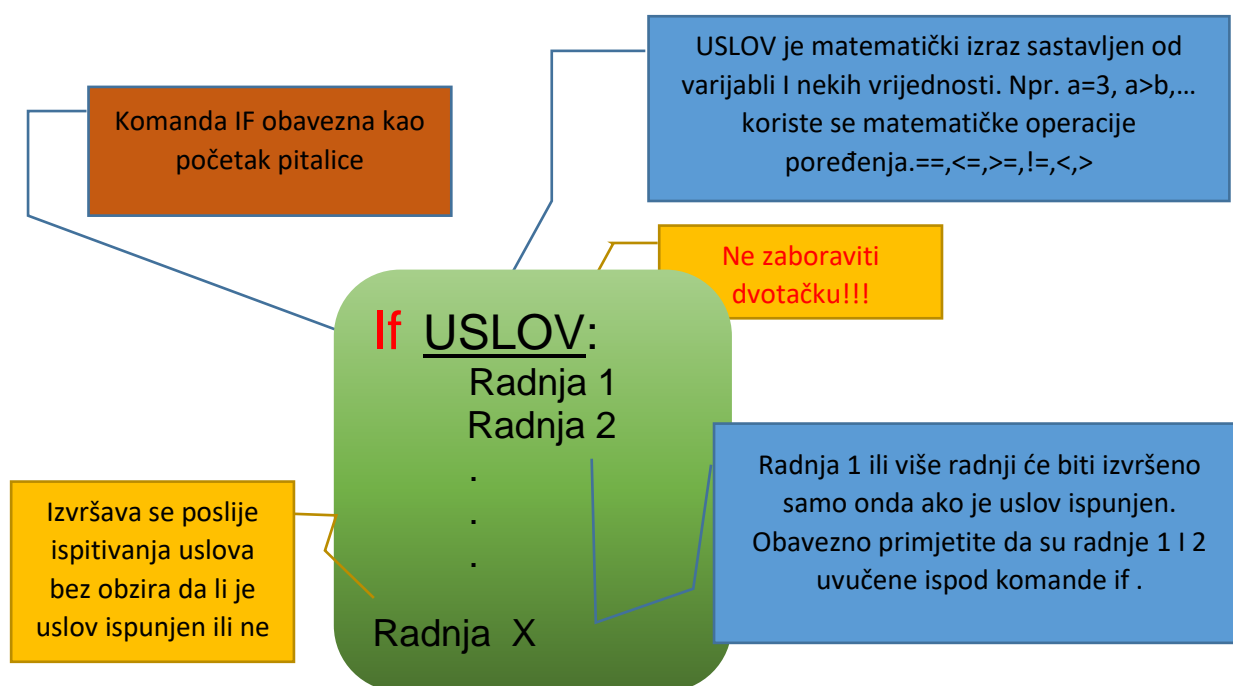
Kod računara situacija je slična postoji uslov i radnja koja se izvršava te naravno i komanda koja sadrži ova dva elementa. Komnada se naziva IF (ako) . Pa često ovu kmonadu nazivamo **pitalica**. Šta ispituje pitalica? Kada pokrenemo program njegova memorija je prazna što znači da moramo u memoriju računara unijeti neke podatke , rekli smo da su **varijable** one koje pamte podatke. Što znači da **pitalica** ispituje vrijednosti **varijabli** i na osnovu tih vrijednosti dešavaju se neke radnje.

Komandi IF koristimo kada želimo da uslovimo vršenje jedne ili više radnji.

Komanda if se sastoji od varijable i matematičkih znakova poređenja.

MATEMATIČKA OZNAKA	OZNAKA U PROGRAMU	ZNAČENJE
=	==	JEDNAKO (IDENTIČNO)
<	<	MANJE
>	>	VEĆE
>=	>=	VEĆE ILI JEDNAKO
<=	<=	MANJE ILI JEDNAKO
≠	!=	RAZLIČITO

OPŠTI OBLIK KOMANDE IF:



*** Primjer:

Napisati program koji će od korisnika tražiti da unese 2 broja i ispisati koji je od tih brojeva veći.

```
Print(„program upoređuje 2
broja“)
Print(„*****“)
#unos brojeva
a=int(input(„unesite 1. broj:“))
b= int(input(„unesite 2. broj:“))
# provjera unešenih brojeva
if a<b:
    print(„drugi broj je veći od
    prvog“)
    print(„*****“)
if a>b:
    print(„prvi broj je veći od
    drugog“)
    print(„*****“)
if a==b:
    print(„brojevi su jednaki“)
    print(„*****“)
print („kraj programa“)
```

Unos brojeva preko tastature.
Unešene brojeve će pamtit
varijable a i b

Ako je $a < b$ onda ispiši poruku „drugi
broj je veći od prvog“ i ispiši
zvjezdice. Ako ne nastavi dalje

Slijedi sljedeća provjera da li je $a > b$ i
ako jeste ispiši poruku koja je
uvučena a ako nije idi na sljedeću
komandu

Sljedeća komanda je ispitivanje da li
su varijable a i b identične. Ako jesu
ispiši poruku a ako nisu uradi
sljedeću komandu. Obratite pažnju
na komandu print koja nije
uvučena .

PITANJA:

Koliko imamo varijabli ?
Kako se zovu ?
Šta pamte ?
Koliko imamo uslova
(pitalica) ?
Kako su napisani ?
Gdje se određuje gdje će biti
ispisana poruka da je prvi broj
veći od drugog ?
Koji je uslov da se ispiše
poruka **kraj programa** ?

*** Napisati program koji će od korisnika tražiti da unese šifru te ispisuje kada je šifra tačno unešena. (Tačna šifra je 123).

Da razmislimo prije rada, šta bi trebalo da se desi :

1. Tražiti od korisnika da unese šifru
2. Provjeriti da li je unešena šifra identična sa tačnom šifrom
3. Ispisati odgovarajuću poruku
4. Završiti program

uvodni dio

Print („program provjerava tačnost unešene šifre“)

unos podataka

x=int(input(„unesite šifru“))

provjera šifre

#ako je vrijednost varijable x identična 123 to jest pravoj šifri tada ispiši poruku

If x==123:

Print(„unešena je prava šifra“)

Print („kraj programa“)



* Napomena obratite pažnju na zadnje 2 print komande i gdje se nalaze u programu. Naravno ovaj program se može napisati na ljepši način, jer očigledno fali i poruka da je unešena pogrešna šifra. Dovršiti program sa odgovarajućom if pitalicom.

*** Napisati program koji će provjeriti da li je unešeni broj paran.

print ("program provjerava da li je unešeni broj paran")

unos broja

a=int(input("Unesite broj: "))

provjeriti da li je ostatak dijeljenja

sa brojem 2 jednak nuli a%2 modul koji računa ostatak djeljenja broja a sa 2

if a%2==0:

print ("broj je paran")

provjera da li je ostatak dijeljenja

sa brojem 2 nije jednak (!=) nuli

if a%2!=0:

print("broj nije paran")

print ("kraj programa")

*** jednostavniji i brži način sa korištenjem IF ELSE

print ('program provjerava dali je broj paran')

a=int(input('Upisi broj: '))

if a%2==0: _____
print ('Broj je paran')

else:
print ('Broj je neparan')

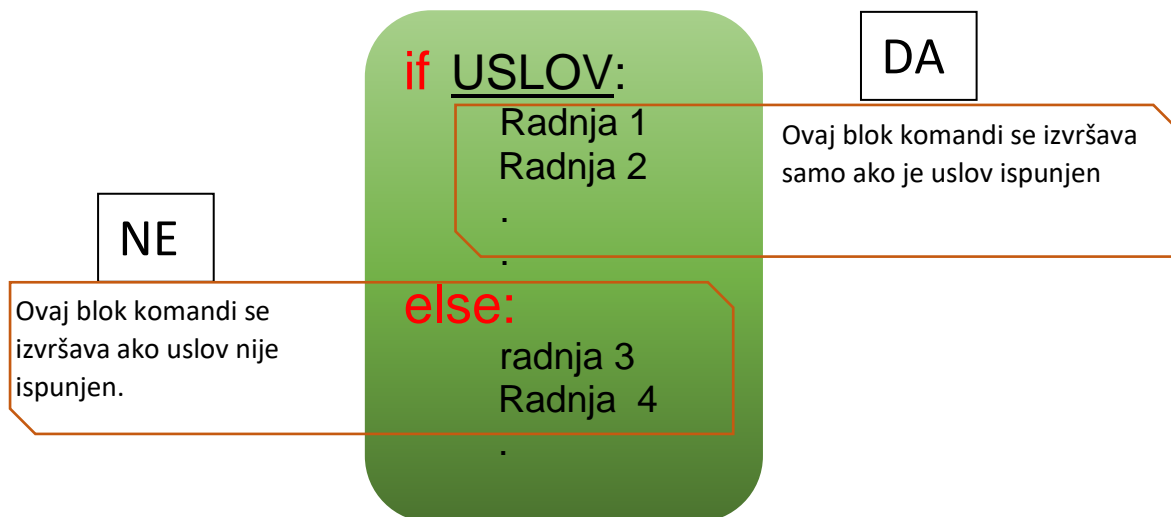
a%2 ova funkcija daje ostatak dijeljenja broja a sa 2. Ako je ostatak djeljenja varijable a sa 2 jednak 0 .

IF – ELSE

PITALICA

Prevedeno na jezik ljudi, **AKO** je uslov ispunjen radi 1. radnju **INAČE** ako nije ispunjen uslov radi radnju 3. Ovdje smo u mogućnosti da sa 1 pitalicom donesemo odluke u 2 ili više mogućih situacija. Sa ovim oblikom se treba pažljivo postupati i koristiti samo u situacijama gdje imamo izbor **TAČNO** ili **NETAČNO**. Inače može dovesti do pogrešnih radnji u programu.

OPŠTI OBLIK KOMANDE IF - ELSE



*** Napisati program koji će od korisnika tražiti da unese šifru te ispisuje kada je šifra tačno unešena. (Tačna šifra je 123).

Da razmislimo prije rada, šta bi trebalo da se desi :

5. Tražiti od korisnika da unese šifru
6. Provjeriti da li je unešena šifra identična sa tačnom šifrom
7. Ispisati odgovarajuću poruku
8. Završiti program

uvodni dio

```
Print („program provjerava tačnost unešene šifre“)
```

unos podataka

```
x=int(input(„unesite šifru“))
```

provjera šifre

#ako je vrijednost varijable x identična 123 to jest pravoj šifri tada ispiši poruku

```
If x==123:
```

```
    print(„unešena je prava šifra“)
```

```
else:
```

```
    print(„unijeli ste neispravnu šifru“)
```

```
Print („kraj programa“)
```


**** Napisati program koji će od korisnika tražiti da unese šifru te ako šifra nije ispravna program će ugasi računar a ako je šifra ispravna pokrenut će facebook stranicu.

Napomena za pisanje ovog programa možemo koristiti kod iz prethodnog programa (copy/paste) te ga jednostavno proširiti sa određenim funkcijama. Kako u samom programskom jeziku ne postoji funkcija za gašenje računara niti funkcija za otvaranje web stranice morat će mo koristiti programske biblioteke koje omogućuju željene funkcije. Python je programski jezik koji ima preko 137 000 programskih biblioteka koje omogućuju mnoštvo dodatnih funkcija iz raznih oblasti života. Prije upotrebe u programu biblioteke se moraju prethodno instalirati na računar sa komandom "pip install" u konzoli ili direktno iz editora kao je što je Pycharm koji omogućava direktnu instalaciju biblioteka bez korištenja pip install. Za gašenje računara koristit će mo biblioteku "os" a za otvaranje stranice koristit će mo biblioteku "webbrowser"

```
#učitavanje potrebnih biblioteka
import webbrowser
import os
# tražimo od korisnika da unese šifru
# tačna šifra je "a123"
a=input("Unesite šifru: ")
# provjeravamo šifru to jest vrijednost varijable a
if a=="a123":
    webbrowser.open("facebook.com")
    # nova komanda iz biblioteke webbrowser (open)
else:
    os.system('shutdown /p /f')
    # nova komanda iz biblioteke os gasi windows OS
```

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

```
a=int(input('Upisi broj: '))

if a%5==0:

    print ('Broj je djeljiv s 5')

else:

    print ('Broj nije djeljiv s 5')
```

*** Napisati program koji će postaviti 2 pitanja i provjeriti da li je odgovor tačan ili ne (kviz).

```
# kviz
print("program postavlja pitanja i provjerava odgovore")
print("*****")

#postavljanje 1. pitanja i provjera odgovora
#pitanje ispisujemo sa komandom print
print("Koji je glavni grad BiH")
a=input("unesite odgovor: ")

#provjera unešenog odgovora
if a=="Sarajevo":
    print("vaš odgovor je tačan")
else:
    print("vaš odgovor je netačan")

#postavljanje i provjera 2. pitanja
b=input("kako se zove najviši vrh BiH: ")
if b=="Maglić":
    print("vaš odgovor je tačan")
else:
    print("vaš odgovor je netačan")
print("kraj kviza")
```

napisati isti ovaj program koji će na kraju ispisati broj tačnih odgovora te procenat tačnih odgovora. Kasnije može se dati i ocjena za određeni postotak tačnih odgovora.



U principu program će biti isti kao i gore napisani program samo što ćemo uvesti nešto novo a to su brojači. Brojači imaju veliku primjenu u programiranju stoga su veoma važni. Brojači se obično pišu u obliku $b=b+1$... povećavamo stanje brojača za 1. $b=b+1$ u kraćem obliku se piše $b+=1$.

U našem slučaju bit će 2 brojača

Brojač tačnih odgovora (bt)

Brojač netačnih odgovora (bn)

Tako da ćemo za svaki dat tačan odgovor brojač bt povećati za jedan $bt+=1$ a u suprotnom slučaju brojač netačnih odgovora se povećava za $bn+=1$.

Kada saberemo broj tačnih i broj netačnih odgovora dobijamo ukupan broj pitanja (bp) . $bp=bn+bt$ to će nam kasnije biti potrebno da izračunamo procenat (p) tačnih odgovora po formuli

$p=bt/bp*100$.

```
# kviz
print("program postavlja pitanja i provjerava odgovore")
print("*****")

#postavljanje 1. pitanja i provjera odgovora
#pitanje ispisujemo sa komandom print
print("Koji je glavni grad BiH")
a=input("unesite odgovor: ")
```

```
#provjera unešenog odgovora
if a=="Sarajevo":
    print("vaš odgovor je tačan")
    bt+=1
else:
    print("vaš odgovor je netačan")
    bn+=1
#postavljanje i provjera 2. pitanja
b=input("kako se zove najviši vrh BiH: ")
if b=="Maglić":
    print("vaš odgovor je tačan")
    # dopunite nedostajuće komande
else:
    print("vaš odgovor je netačan")
    # dopunite nedostajuće komande

# računanje ukupnog broja pitanja bp
# ovaj dio programa ide na kraj kviza
bp=bn+bt
# računajte procenta tačnih odgovora
p=bt/bp*100

print („imate „ , p , “ tačnih odgovora“)
print ("kraj kviza")
```

Svaki put moramo povećati brojač.