



RJEŠAVANJE PROBLEMA PROGRAMIRANJEM

Uvod u programski jezik
Python

O PYTHONU...

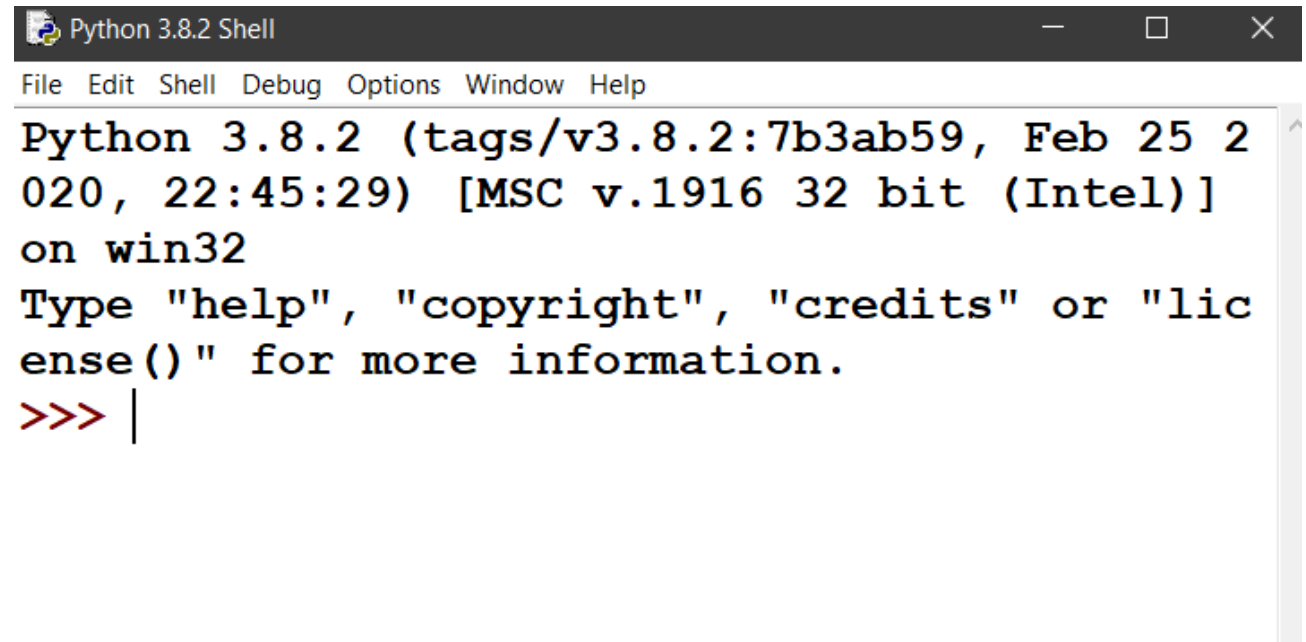
- stvorio ga je Guido van Rossum 1991.
- ime je dobio po kultnoj britanskoj komediji „Monty Python’s Flying Circus” (programiranje može biti zabavno)
- **Python je interpretator koji podržava strukturalno i objektno orijentirano programiranje. Podržava integriranje programskog koda napisanog nekim drugim programskim jezikom. (iz udžbenika)**
- **Python je interpretirani, interaktivni, objektno orijentirani programski jezik visoke razine**
- najveća promjena Python-a bila je pri prelasaku verzije 2.x na verziju 3.x
- trenutna verzija 3.9.7 (<https://www.python.org/>)

ZAŠTO PYTHON?

- besplatan programski jezik
- podržava ga većina operacijskih sustava
- široka rasprostranjenost (među pet najpopularnijih programskih jezika na svijetu)
- koristi se za: razvoj aplikacija (i u igrama), analizi podataka, razvoju interneta (Google, Facebook, Reddit, Dropbox, YouTube, Instagram, Pinterest...)
- jednostavna sintaksa (lako uči, laka čitljivost koda)
- omogućava brz napredak – (programeri provode više vremena razmišljajući o problemu koji rješavaju, a manje vremena razmišljaju o kompleksnosti jezika)
- programi pisani u Pythonu većinom su sporiji

PYTHON SHELL

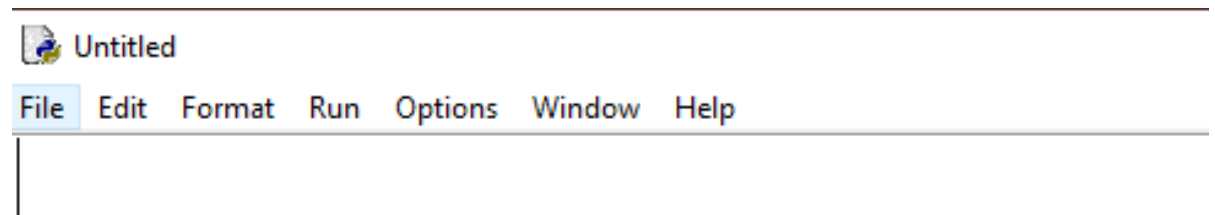
- interaktivni način rada
- **upisane naredbe se interpretiraju se u realnom vremenu i odmah vidimo njihov ishod**
- koristi se za isprobavanje korištenja naredbi ili nekih ideja
- može poslužiti kao kalkulator



```
Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)]
on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> |
```

TRAJNO POHRANJIVANJE PROGRAMA

- skriptni način rada
- **pisanje, mijenjanje i trajno pohranjivanje programa**
- File → New File
- **programi se spremaju u datoteke s nastavkom .py**, a izvršavaju se s naredbom Run Module u izborniku Run



ARITMETIČKI OPERATORI

Operator	Primjer	Opis
+	$x = a + b$	Zbrajanje dvaju brojeva, $a = 5$, $b = 3$, $x = 5 + 3 = 8$
-	$x = a - b$	Oduzimanje dvaju brojeva, $a = 5$, $b = 3$, $x = 5 - 3 = 2$
**	$x = a ** b$	Potenciranje dvaju brojeva, $a = 5$, $b = 3$, $x = 5 ** 3 = 125$
*	$x = a * b$	Množenje dvaju brojeva, $a = 5$, $b = 3$, $x = 5 * 3 = 15$
%	$x = a \% b$	Ostatak pri djeljenju dvaju brojeva, $a = 5$, $b = 3$, $x = 5 \% 3 = 2$
/	$x = a / b$	Djeljenje dvaju brojeva, $a = 5$, $b = 3$, $x = 5 / 3 = 1.6666...$
//	$x = a // b$	Cjelobrojno djeljenje dvaju brojeva, $a = 5$, $b = 3$, $x = 5 // 3 = 1$

REDOSLIJED IZVRŠAVANJA

Prioritet	Primjer	Opis
1.	()	od unutarnjih zagrada prema vanjskim zagradama
2.	**	
3.	// * / %	s lijeva na desno
4.	+ -	s lijeva na desno

PRIMJERI U SHELLU:

```
Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)]
on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 9/2
4.5
>>> 9//2
4
>>> 5**3
125
>>> 17%8
1
>>> 17%5
2
>>> 4+7**2//5%4
5
>>>
```


PRIMJERI:

1. Izraz $2^{**} 4^{**} 2$ ekvivalentan je kojemu izrazu?
 - a) $2^{**} (4^{**} 2)$
 - b) $(2^{**} 4)^{**} 2$
2. Kolika je vrijednost a ako je $a = 2^{**} 2^{**} 3$?
3. Kolika je vrijednost izraza a ako je $a = 2 + 2 * 2$?
4. Kolika je vrijednost izraza a ako je $a = 17 / 2 \% 2 * 3^{**} 3$?
5. Kolika je vrijednost izraza a ako je $a = 9 * 4 \% 5$?
6. Kolika je vrijednost izraza a ako je $a = 170 \% 19 // 3$?
7. Kolika je vrijednost izraza a ako je $a = 170 // 17 \% 4$?
8. Kolika je vrijednost izraza a ako je $a = 9 * 6 \% 15 // 2$?
9. Kolika je vrijednost izraza a ako je $a = -9 \% 5$?
10. Kolika je vrijednost izraza a ako je $a = 12 \% - 10$?

PRIMJERI:

11. Kolika je vrijednost izraza a ako je $a = 8 + 6 * 6 / 2 * 2 - 1$?

12. Koji su od sljedećih aritmetičkih izraza sintaksno ispravno napisani?

a) $(9 - 3) / (3 ** 2)$

b) $19 + * 4$

c) $92 - (22 - (14 * 23))$

d) $(85 + (11 + (27 * 14) - 13))$

e) $57 / +4$

13. Ako je $n=987.6$, koji će od sljedećih izraza imati vrijednost 8.0?

a) $((n - n \% 10) \% 100) / 10$

b) $(n / 10) \% 10$

c) $(n // 10) \% 10$

d) $(n \% 10) // 10$

PRIMJERI:

14. Odredi što će ispisati sljedeći dio koda:

- $a = 4, b = 3$
- $x = a + b ** 3 \% 5 // 2$

15. Odredi što će ispisati sljedeći dio koda:

- $a = 6, b = 2$
- $x = a - 2 + b * 4 - a ** (b + 1) * 2 \% 4$

16. Odredi što će ispisati sljedeći dio koda:

- $a = 3, b = 2$
- $x = b ** (b * 5 // a + (2 - a + 10)) \% 5$

FUNKCIJE U PYTHON-U

- matematika
 - uređena trojka oblika (D, K, f) koja svakom elementu skupa D (domena) pridružuje točno jedan element iz skupa K (kodomena) koristeći pravilo f .
 - npr. $f(x) = 2x + 7$ (primjer linearne funkcije čiji je graf pravac)
- **informatika**
 - **za zadane početne vrijednosti nam vraćaju neku novu vrijednost**
 - **koristimo ih bez da znamo kako su osmišljene i kako dolaze do rješenja**
- unaprijed definirane funkcije – zapisane u osnove programskog jezika
 - **abs(x)** – apsolutna vrijednost
 - **round(n, d)** – zaokružuje broj n na d decimala
 - **pow(x, y, mod)** – potencija broja x na y , a ako dodamo još mod računa ostatak pri dijeljenju s tim brojem
 - `print()`
 - ...

PRIMJERI U SHELLU:

```
Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29)
) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more
information.
>>> abs(2.5)
2.5
>>> abs(-54.7)
54.7
>>> round(2.8985)
3
>>> round(2.8984, 1)
2.9
>>> round(2.8984, 2)
2.9
>>> pow(2, 5)
32
>>> pow(3, 2)
9
>>> pow(3, 2, 5)
4
>>> pow(3, 2, 4)
1
>>>
```



VARIJABLE

VARIJABLE U PYTHONU

- memorijska lokacija simboličnog imena
- ne treba se (za razliku od drugih programskih jezika) unaprijed deklarirati i za nju rezervirati određenu memoriju
- imena varijabli:
 - **mogu sadržavati brojeve, velika i mala slova te znak "_"**
 - **ne smiju početi brojem**
 - **python razlikuje velika i mala slova**
 - nazivi važnih varijabli bi trebali biti opisani imenom (ne r, nego rezultat)
 - **duljina neka je do 15 znakova**
 - **uvijek počinju malim slovom ne znakom "_"**
- svaki puta kad mijenjamo vrijednost varijable mijenja se i njezina memorijska lokacija

PRIMJER

```
>>> a=5
>>> id(a)
1922365424
>>> a=7
>>> id(a)
1922365456
>>> a=a+2
>>> a
9
>>> id(a)
1922365488
>>> id(a)
1922365488
>>> |
```

- naredba **id()** nam vraća adresu memorijske lokacije neke varijable

PRIDRUŽIVANJE VRIJEDNOSTI

- znakom = možemo varijabli pridružiti neku vrijednost
- **umjesto $a = a + 2$ možemo pisati $a += 2$ (skraćeni zapis) i to vrijedi za sve aritmetičke operatore**
- možemo istovremeno pridružiti različite ili iste vrijednosti većem broju varijabli

```
>>> a = 4
>>> a
4
>>> a = a + 2
>>> a
6
>>> a += 2
>>> a
8
>>> a %= 5
>>> a
3
>>> a = b = c = 4
>>> a
4
>>> b
4
>>> a, b, c = -2, 5.3, 3
>>> a
-2
>>> c
3
>>> |
```

UNOŠENJE VRIJEDNOSTI

- naredba `input()` omogućava nam unošenje vrijednosti varijable
- kod unosa podataka moramo paziti na tip podatka koji želimo spremiti (cijeli broj, realni, znakovi...)
- ne možemo lako unositi više varijabli istovremeno

```
>>> a = input()
5
>>> a
'5'
>>> a+2
Traceback (most recent call last):
  File "<pyshell#2>", line 1, in <module>
    a+2
TypeError: can only concatenate str (not "int") to str
>>> b = input()
7
>>> a+b
'57'
>>> int(a)+int(b)
12
>>> |
```

ODABIR TIPA PODATAKA

Funkcija	Opis
int()	vraća cjelobrojnu vrijednost neke varijable
float()	vraća realnu vrijednost neke varijable
str()	vraća znakovnu vrijednost neke varijable

ODABIR TIPa PODATAKA

```
>>> a = "5"
>>> a
'5'
>>> a+2
Traceback (most recent call last):
  File "<pyshell#2>", line 1, in <module>
    a+2
TypeError: can only concatenate str (not "int") to str
>>> int(a)+2
7
>>> float(a)+2
7.0
>>> a+str(2)
'52'
>>> |
```


ODABIR TIPA PODATAKA

```
>>> a = input()
5
>>> a
'5'
>>> a = int(input())
5
>>> a
5
>>> a+2
7
>>> b = float(input())
5.7
>>> b
5.7
>>> c = int(input("Unesi cijeli broj:"))
Unesi cijeli broj:7
>>> c
7
>>> |
```

- tip podataka možemo zadati tako da naredbu za pretvaranje u određeni tip dodamo ispred input naredbe
- možemo dodati i string radi lakšeg unosa podataka

UNOŠENJE VRIJEDNOSTI VIŠE VARIJABLI

```
>>> a, b, c = 5, 5.2, "prog"
>>> c
'prog'
>>> a, b, c = int(input())
5 2 3
Traceback (most recent call last):
  File "<pyshell#2>", line 1, in <module>
    a, b, c = int(input())
ValueError: invalid literal for int() with base 10: '5 2 3 '
>>> a, b, c = eval(input())
5, 2, 3
>>> c
3
>>> a, b, c = map(int, input().split())
4 9 7
>>> c
7
>>> b
9
>>> a, b, c = map(int, input().split(", "))
7, 8, 9
>>> c
9
>>> a
7
```

- funkcija **eval()** omogućava unošenje više vrijednosti iako nije namijenjena tome
- funkcija **map(f, slijed)** omogućuje djelovanje funkcije f na svaki element iz zadanog slijeda

NAREDBA ISPISA

- naredba `print()` omogućuje ispis zadanih vrijednosti na zaslon monitora
- `print()` dodaje razmak između ispisanih vrijednosti, ali mi možemo argumentom `sep` dodati neki drugi znak
- argumentom `end` možemo promijeniti da nam kursor dođe u novi red (da ispis završi nekim drugim znakom)

```
>>> a = 7
>>> print("Danas je sunčan dan")
Danas je sunčan dan
>>> print(a+2)
9
>>> print("Broj a jednak je", a)
Broj a jednak je 7
>>> print("Broj a jednak je", a, sep = ":")
Broj a jednak je:7
>>> print("Broj a jednak je", a, sep = "--")
Broj a jednak je--7
>>> print("Broj a jednak je", a, sep = "--", end = "!")
Broj a jednak je--7!
>>> |
```

NAREDBA ISPISA

Posebni znakovi kod ispisa (escape sekvence)	Značenje
\\	ispisuje lijevo ukošenu crtu (backslash, \)
\'	ispisuje jednostruki navodnik
\"	ispisuje dvostruki navodnik
\n	prijelaz u novi red
\t	horizontalni tab

```
>>> print("Backslash, \\")
Backslash, \
>>> print("U navodnicima \"dvostruki\" i \'jednostruki\'")
U navodnicima "dvostruki" i 'jednostruki'
>>> print("prvi red \ndrugi red \ntreći red")
prvi red
drugi red
treći red
>>> print("tab \ttab \ttab")
tab      tab      tab
>>> |
```

DODATNO OBLIKOVANJE ISPISA

- unutar dvostrukih navodnika navodimo poseban tekst (varijablu) koji želimo ispisati, a izvan navodnika vrijednost varijable ispred koje ide znak %

Kod	Opis
%s	znakovni niz
%c	znak
%d	dekadski cijeli broj
%i	cijeli broj
%u	cijeli broj bez predznaka
%o	oktalni broj
%x	heksadekadski broj
%X	heksadekadski broj zapisan velikim slovima
%e	eksponencijalni zapis broja s pomičnom točkom
%E	eksponencijalni zapis broja s pomičnom točkom, veliko E
%f	realni broj
%.xf	ispisuje decimalni dio realnog broja na x decimala

PRIMJERI

```
>>> a=5
>>> b=3
>>> c=3.78956
>>> print("Prvi broj %d Drugi broj %10d" %(a, b))
Prvi broj 5 Drugi broj          3
>>> print("Prvi broj %i Drugi broj %10i" %(a, b))
Prvi broj 5 Drugi broj          3
>>> print("Prvi broj %x Drugi broj %5f" %(a, c))
Prvi broj 5 Drugi broj 3.789560
>>> print("Treći broj %e" %(c))
Treći broj 3.789560e+00
>>> d = 3547.4567
>>> print("Četvrti broj %e" %(d))
Četvrti broj 3.547457e+03
>>> print("Treći broj %.3f" %(c))
Treći broj 3.790
```




PRVI PROGRAMI

PRIMJER 1.

Napiši program koji učitava dva cijela broja a i b i ispisuje njihov zbroj i količnik. (Pretpostavlja se da je b različito od nule)

PR 1. - RJEŠENJE

```
a = int(input())  
b = int(input())  
print("Zbroj je:", a+b, "Količnik je", a/b)
```

PRIMJER 2.

Napiši program koji unosi četiri cijela broja i računa njihovu prosječnu vrijednost.

Ulaz

$$a = 5$$

$$b = 4$$

$$c = 7$$

$$d = 8$$

Izlaz

6.0

PR. 2. – RJEŠENJE

```
a = int(input())  
b = int(input())  
c = int(input())  
d = int(input())  
prosjek = (a+b+c+d)/4  
print("Prosjek je", prosjek)
```

PRIMJER 3.

Napiši program koji učitava duljinu radijusa kruga (realan broj) i ispisuje njegov opseg. Zadatak riješite bez konstante pi, odnosno $\pi = 3.14$.

Ulaz

5

Izlaz

62.83185

Ulaz

7

Izlaz

125.6637

PR. 3. – RJEŠENJE (BEZ KONSTANTE)

```
r = float(input("Unesi radijus"))  
o = 2 * r * 3.14  
print("Opseg je: ", o)
```

PRIMJER 4.

Napiši program koji će učitati godinu rođenja osobe i izračunati koliko godina osoba ima sada.

Ulaz

1988

Izlaz

33

PR. 4. – RJEŠENJE

```
g = int(input("Godina rođenja:"))  
print(2021-g)
```

PRIMJER 5.

Napiši program koji će učitati prirodan dvoznamenkasti broj i ispisati posebno njegove znamenke.

Ulaz

57

Izlaz

5

7

PR. 5. – RJEŠENJE

```
a = int(input("Unesi dvoznamenkasti broj:"))
jed = a % 10
des = a // 10
print(jed, "\n", des, sep = "")
```

PRIMJER 6.

Napiši program koji će učitati prirodan troznamenkasti broj i ispisati zbroj njegovih znamenki.

Ulaz

167

Izlaz

14

PR. 6. – RJEŠENJE

```
a = int(input("Unesi troznamenkasti broj:"))
jed = a % 10
des = a // 10 % 10
sto = a // 100
print(jed+des+sto)
```


ZADACI ZA VJEŽBU

- **Str. 56. – 57.**

DZ

- 1. a, b, c, d
- 2.
- 6. znamenke
- 7. znamenke
- 11.
- 12.

- **Str. 57. – 59.**

- 8.
- 9.
- 10.
- 14.
- 15.
- 16.
- 17.
- 18.