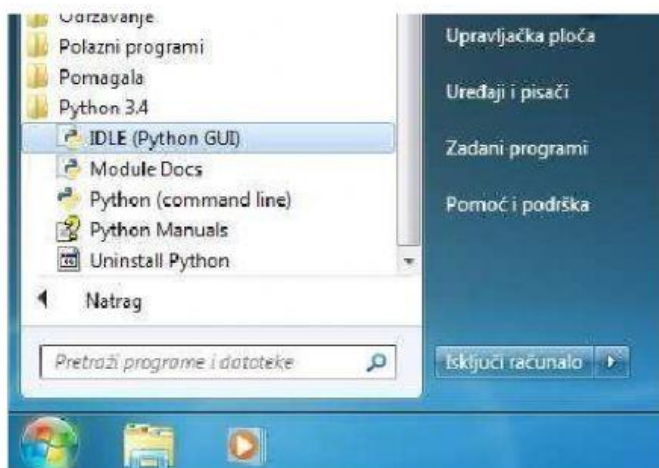


Programski jezik Python

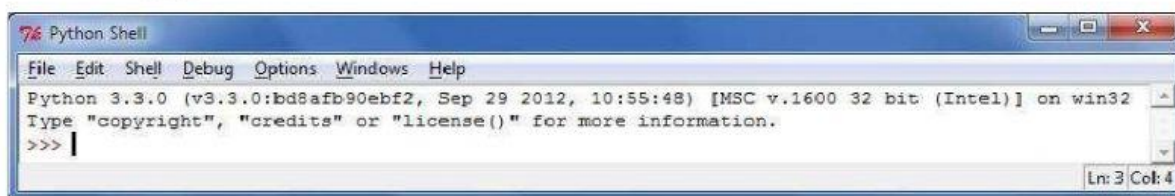
pokrećemo otvaranjem razvojnog okruženja za Python nazvanog IDLE (izvedenica iz Integrated DeveLopment Environment) na više načina:

1. Start > Svi programi (*All Programs*) > Python 3.4 > (Python GUI)
2. Dvostrukim klikom na ikonu prečice na radnoj površini



Kad se IDLE pokrene otvori se prozor koji čini
Shell

sučelje s Pythonom i naziva se



Na početku novog retka pojavljuju se strelice i to mjesto nazivamo znak
(engl. *prompt*)

```
>>>
```

gdje unosimo neki podatak ili naredbu.

Osnovni tipovi podataka

Osnovni tipovi podataka u Pythonu su:

- Cijeli
- Decimalni brojevi
- niz ili string

Cijeli brojevi

Kad se nakon znaka upita utipka neka vrijednost, Python ispisuje tu istu vrijednost:

```
>>> 150
```

Python ispisuje vrijednost
Pythonovog ispisa.

bojom tako da se razlikuje ono što smo utipkali od

Naziv za podatak u Pythonu je **int**.

Broj znamenki cijelog broja nije ograničen:

```
>>> 13
13
>>> -40

>>> 1234567890123456789012345
1234567890123456789012345
```

Decimalni brojevi

Prilikom upisivanja decimalnog broja koristimo prepoznaje decimalni broj, tj. da je tip **float**.

točku. Po toj točki Python

Nekoliko primjera:

```
>>> 2.6
2.6
>>> 26.

>>> 0.26
0.26
>>> .26

>>> .026
0.026
```

Pretvaranje decimalnih brojeva u cijele i obratno

Python ima ugrađene funkcije **int()** i **float()** pomoću kojih možemo pretvoriti decimalni broj u cijeli i obratno.

Za pretvaranje broja u cijeli koristimo funkciju **int()**:

```
>>> int(2.3)
2
>>> int(2.8)
2
```

Pretvaranje se sastoji u zaokruživanju decimalnog broja u prvi cijeli broj.

Za pretvaranje broja u decimalni koristi se funkcija **float()**:

```
>>> float(4)
4.0
```

Pretvaranje se sastoji u dodavanju decimalne točke i prikazu sa decimalnom nulom.

Python kao kalkulator

Kako bi dobili neku novu vrijednost koristimo se matematičkim izrazima. U matematičkim izrazima rabimo matematičke operatore ovisno što želimo dobiti. U Pythonu postoje sljedeći operatori:

Operator	Opis djelovanja
+	Zbrajanje
-	
*	Množenje
/	Dijeljenje
//	Cjelobrojno
%	dijeljenja

Neki jednostavniji izrazi:

```
>>> 6+3
9

>>> 6-3
3

>>> 6*3
18

>>> 6/3
2.0
```

Vidimo da su rezultati uvijek cjelobrojni osim kod dijeljenja iako rezultat može biti cijeli broj.

Operator (/) uvijek daje decimalni rezultat (**float**).

Operator za cjelobrojno (//) ispisuje rezultat dijeljenja kao cijeli broj bez ostatka:

```
>>> 17//4
4
```

Kod običnog djeljenja bi izgledalo ovako:

```
>>> 17/4
4.25
```

Operator za **dijeljenja** (%) ispisuje ostatak od cjelobrojnog dijeljenja:

```
>>> 17%4
1
```

Znači, pokažimo to kao u matematici,

$17 : 4 = 4$ i ostatak 1

Taj **dijeljenja** (**1**) ispisuje Python.

Kao u matematici operatori su svrstani u red prvenstva:

```
>>> 2+3*4
```

Prvo množenje ($3*4=12$) pa zbrajanje ($12+2=14$)

```
>>> 15/3-2
3.0
```

Kod pisanja složenijih izraza mogu se koristiti zagrade:

```
>>> (2+3) * 4
20
```

Znakovni niz – string

Osnovni tip podataka za prikaz je znakovni niz ili string.

U Pythonu se taj tip zove str, a vrijednost znakovnog niza obilježava se jednostrukim ili navodnim znacima na početku i na kraju niza.

U pravilu, preporučljivo je koristiti jednostruke

```
>>> 'Open source'
'Open source'
```

Ako se unutar teksta ponovo javljaju navodnici, onda se koriste navodnici.

```
>>> 'Linux je "Open source" sustav'
'Linux je "Open source" sustav'
```

Treba naglasiti da i za jednostruki i za dvostruki navodnik postoje zasebne tipke na tipkovnici. Dvostruki navodnik se smije pisati kao dva uzastopno napisana jednostruka navodnika.

Varijable i znak pridruživanja

Problem pamćenja vrijednosti riješen je uvođenjem varijabli.

```
>>> a=15
```

Varijable su imena kojima pojedine vrijednosti. Primjerice:

Varijabli **a** smo pridružili vrijednost 15. Ako napišemo:

```
>>> a
```

Python će ispisati vrijednost **a**.

Isto tako:

```
>>> b=5
>>> b
5
>>> a+b

>>> a-b
10
>>> a*b

>>> a/b
3.0
>>> a%b
0
```

Možemo uočiti da operator dijeljenja (/) uvijek daje decimalni rezultat (**float**).

Tekst se pridružuje na način da se stave jednostruki ili dvostruki navodnici:

```
>>> a='Jedan'
>>> a
```

Na sličan način:

```
>>> b='Dva'
>>> a+b
'JedanDva'
```

```
>>>a+b+'Tri'
'JedanDvaTri'
```

```
>>> program='Word'+ 'Pad'
>>> program
```


Za prikaz riječi sa razmakom upisujemo navodnike s razmakom:

```
>>>a+' '+b  
'Jedan Dva'
```

```
>>> a*4
```

```
>>> (a+b)*3  
'JedanDvaJedanDvaJedanDva'
```

```
>>> (a+b+' ')*3  
'JedanDva JedanDva JedanDva'
```

Višestruko pridruživanje

Python omogućuje pridruživanje tj. jednu vrijednost istodobno pridružuje na više varijabli.

```
>>> x=y=z=20  
>>> x  
20  
>>> y  
  
>>> z  
20
```