

Python Development

Sintaksa Python-a i
Kontrola toka izvršavanja



POWERED BY



COMTRADE

Podsetnik od prethodnog susreta

- IDLE koristimo za proveru funkcionalnosti Python-a
- Workflow za pisanje Python programa:
 - Pišemo programe u tekstualnom editoru (npr. Sublime ili Notepad++)
 - Čuvamo ih kao **.py** fajlove
 - Pokrećemo ih u command prompt-u: **python <ime-programa>.py**
 - Da bismo mogli da ih pokrenemo na ovaj način moramo imati putanju do Python izvršnih fajlova unutar PATH environment promenljive



POWERED BY



COMTRADE

Podsetnik od prethodnog susreta

- **print** je funkcija koja vrši ispis na standardni izlaz (ekran)
- **input** je funkcija koja učitava podatke unete preko tastature
- Funkcije uzimaju ulazne podatke (*input*) i pretvaraju ih u izlazne podatke (*output*) i/ili akcije
- Razlikujemo ime funkcije ***func*** i poziv funkcije ***func(arg)***



POWERED BY



COMTRADE

Podsetnik od prethodnog susreta

- Primeri za **print**:
 - **print('zdravo svete')**
 - **print("zdravo " + "svete")**
 - **print('zdravo', 'svete')**
 - **print(2 + 2)**
 - **print('2 + 2 = ', 2 + 2)**



POWERED BY



Podsetnik od prethodnog susreta

- Primeri za **input**:
 - **prva_promenljiva = input()**
 - **druga_promenljiva = input('Unesite vrednost: ')**



POWERED BY



COMTRADE

Promenljive

- Dodelom vrednosti **podatak = input()** pravimo novu promenljivu **podatak**
- Deo memorije računara koji čuva podatke
- Možemo više puta da ih koristimo i da ih menjamo
- Ime promenljive mora da počne slovom
- Ime promenljive može da sadrži slova, brojeve i donju crtu (*underscore*)



POWERED BY



COMTRADE

Promenljive

- Dobre prakse programiranja:
 - Python *style guide* - <https://www.python.org/dev/peps/pep-0008/>
 - Ime promenljive bi trebalo da koristi samo mala slova (*lowercase letters*) i donju crtu
 - Ime promenljive bi trebalo da nagoveštava koji podatak ta promenljiva sadrži:
 - ✓ ■ `omiljeni_broj = input('Unesite svoj omiljeni broj: ')`
 - ✗ ■ `a = input()`



POWERED BY



COMTRADE

Rezervisane reči u Python-u

- Postoje posebne reči sa specijalnom namenom koje se ne smeju upotrebljavati kao identifikatori (imena promenljivih, funkcija ili klasa)
- Spisak se može naći na:

https://docs.python.org/3/reference/lexical_analysis.html#keywords

False	await	else	import	pass
None	break	except	in	raise
True	class	finally	is	return
and	continue	for	lambda	try
as	def	from	nonlocal	while
assert	del	global	not	with
async	elif	if	or	yield



POWERED BY



COMTRADE

Rezervisane reči u Python-u

- Postoje posebne reči sa specijalnom namenom koje se ne smeju upotrebljavati kao identifikatori (imena promenljivih, funkcija ili klasa)
- Spisak se može naći na:
https://docs.python.org/3/reference/lexical_analysis.html#keywords
- Iako nije zabranjeno, ne preporučuje se korišćenje već upotrebljenih, standardnih identifikatora, npr. **str**, **int**...
- Ok je upotrebiti rezervisane reči kao deo identifikatora:
 - **moj_print**, **TrueEvalClass**...
 - **print_moj_broj.py**



POWERED BY



COMTRADE

Velika i mala slova u Python-u

- Python pravi razliku između velikih i malih slova (*case-sensitive*)
- *Zadatak: Sa standardnog ulaza učitati vrednost i smestiti u promenljivu moja_promenljiva, potom na standardni izlaz odštampati vrednost koja je smeštena u moja_promenljiva i vrednost koja je smeštena u MOJA_PROMENLJIVA.*
- *Zadatak: Napisati program koji zamenjuje vrednosti dvema promenljivama*



POWERED BY



COMTRADE

Ispisivanje sa formatiranjem

- Vrednost promenljive možemo da ispišemo na više načina
- **format** je funkcija koja nam pruža više mogućnosti za ispis
- Počev od Pythona 3.6.5 moguće je korišćenje prefiksa **f** za lakše formatiranje stringova: *f"Ovo je string sa {promenljivom}"*
- *Zadatak: Napisati program koji štampa vrednost unetu sa standardnog ulaza na što više načina.*



POWERED BY



COMTRADE

Tipovi podataka u Python-u

- Kompletna lista: <https://docs.python.org/3/library/stdtypes.html>
- Danas ćemo govoriti o primitivnim (osnovnim) tipovima podataka

Naziv	Tip	Primer	Komentar
Boolean	bool	svi_prisutni = True	(ne)istinitost tvrdnje
Integer	int	moj_omiljeni_broj = 7	celobrojna vrednost
Float	float	pi = 3.14159	decimalna vrednost
String	str	pozdrav = 'zdravo svete'	tekst, niz karaktera
None	NoneType	naziv_projekta = None	prazna varijabla



POWERED BY



Tipovi podataka u Python-u

- *String*-ovi su primitivni, ali imaju i osobine složenih tipova podataka
- Možemo konvertovati tipove podataka **kastovanjem** (eng. *cast*)
- *Zadatak: Napisati program koji radi sva moguća kastovanja*



POWERED BY



COMTRADE

Aritmetičke operacije u Python-u

Operator	Operacija	Primer	Rezultat
+	sabiranje	$3 + 5$	8
-	oduzimanje	$20 - 5$	15
*	množenje	$3 * 7$	21
/	deljenje	$20 / 7$	2.86
//	celobrojno deljenje	$20 // 7$	2
**	eksponent	$2 ** 3$	8
%	ostatak (kongruencija)	$20 \% 7$	6



POWERED BY



Aritmetičke operacije u Python-u

- Prioritet operacija isti kao u matematici
 - Zgrade
 - Eksponent (**)
 - Množenje (*), deljenje (/), celobrojno deljenje (//), ostatak (%)
 - Sabiranje (+), oduzimanje (-)



POWERED BY



Operatori poređenja u Python-u

Operator	Opis	Primer	Rezultat
==	jednako	10 == 20	False
!=	nije jednako	10 != 20	True
>	veće	10 > 20	False
<	manje	10 < 20	True
>=	veće ili jednako	10 >= 20	False
<=	manje ili jednako	10 <= 20	True



POWERED BY



Boolean operatori u Python-u

iskaz_1	iskaz_2	iskaz_1 and iskaz_2
True	True	True
True	False	False
False	True	False
False	False	False

iskaz_1	iskaz_2	iskaz_1 or iskaz_2
True	True	True
True	False	True
False	True	True
False	False	False

iskaz_1	not iskaz_1
True	False
False	True



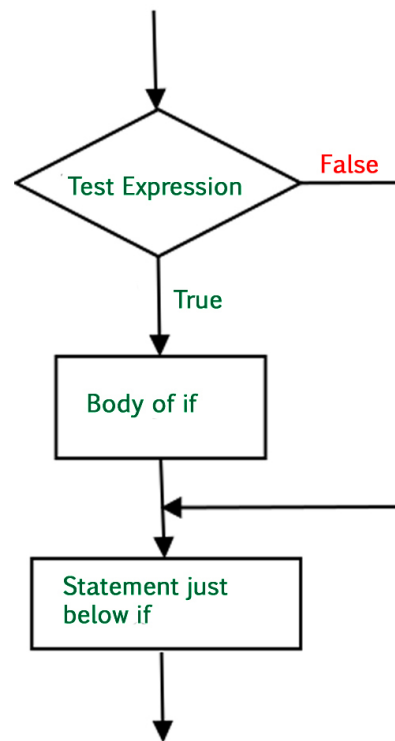
POWERED BY



COMTRADE

Kontrola toka u Python-u

- Izvršavanje koda u zavisnosti od uslova - **if**
- *Zadatak: Uneti broj sa standardnog ulaza i ispisati da li je broj paran*

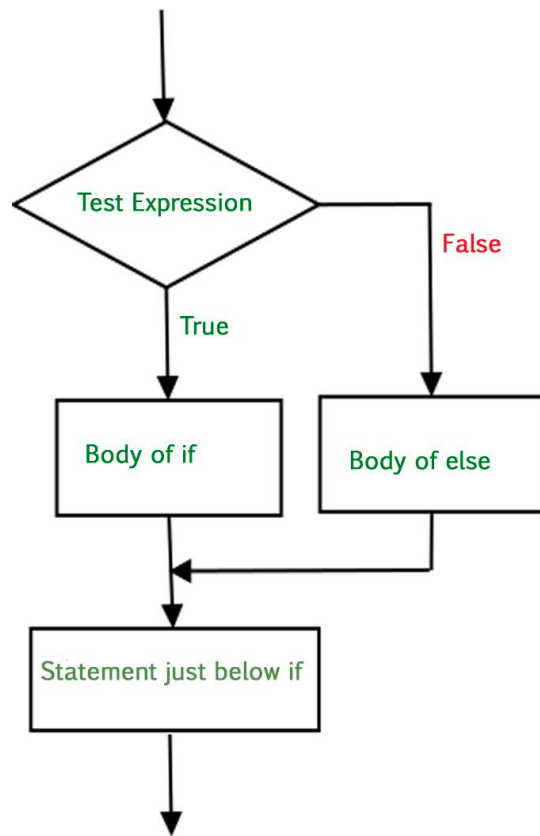


POWERED BY



Kontrola toka u Python-u

- Izvršavanje prvog bloka koda u slučaju da je uslov ispunjen, a drugog bloka koda ako nije - **if-else**
- *Zadatak: Uneti broj sa standardnog ulaza i ispisati da li je broj paran ili neparan*

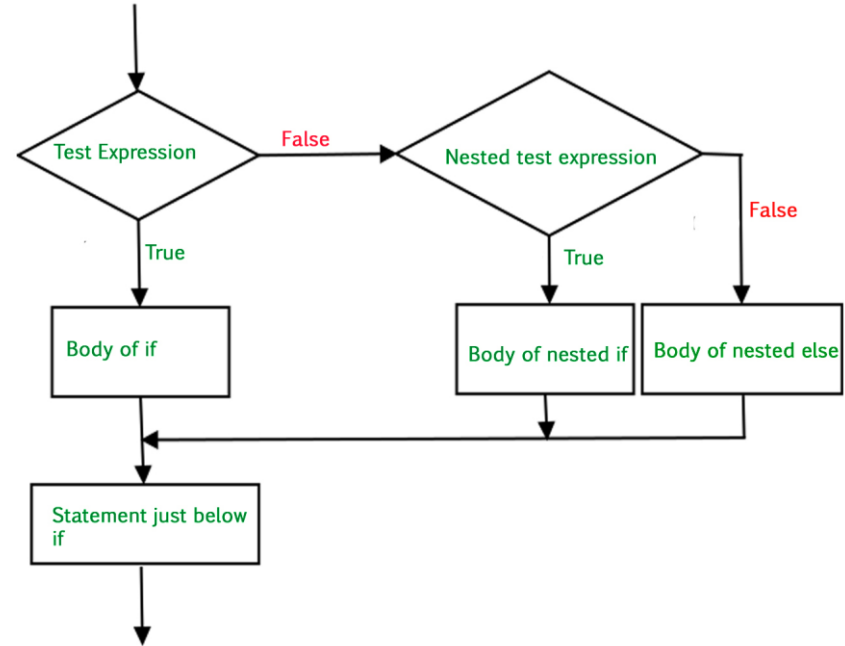


POWERED BY



Kontrola toka u Python-u

- Izvršavanje jednog od blokova koda u zavisnosti od toga koji od uslova je ispunjen - **if-elif-else**
- *Zadatak: Uneti broj sa standardnog ulaza i proveriti da li je uneti broj manji od 100, veći od sto a manji od 200, veći od 200 a manji od 300, i ispisati odgovarajuće poruke*



Zadaci za vežbanje (1)

- Napisati program koji pomaže kasirki da obračuna kusur koji treba da vrati kupcu. Za unetu cenu artikla, količinu artikla i iznos koji je kupac dao, program treba da ispiše vrednost kusura. Sve ulazne vrednosti su pozitivni celi brojevi. *Napomena: Pretpostaviti da je unos ispravan.*

Test primeri:

INTERAKCIJA SA PROGRAMOM:

Unesite cenu, kolicinu i iznos:

132 2 500

Kusur: 236 dinara

INTERAKCIJA SA PROGRAMOM:

Unesite cenu, kolicinu i iznos:

59 6 2000

Kusur: 1646 dinara

Zadaci za vežbanje (2)

- Napisati program koji za uneti redni broj dana u nedelji ispisuje ime odgovarajućeg dana. U slučaju neispravnog unosa, ispisati odgovarajuću poruku o grešci.

Test primeri:

```
INTERAKCIJA SA PROGRAMOM:
```

```
Unesite broj: 4
```

```
U pitanju je: cetvrtak
```

```
INTERAKCIJA SA PROGRAMOM:
```

```
Unesite broj: 8
```

```
Greska: neispravan unos  
dana.
```

```
INTERAKCIJA SA PROGRAMOM:
```

```
Unesite broj: 7
```

```
U pitanju je: nedelja
```

Zadaci za vežbanje (3)

- Napisati program koji za unetu godinu i mesec ispisuje naziv meseca kao i koliko dana ima u tom mesecu te godine. U slučaju neispravnog unosa ispisati odgovarajuću poruku o grešci. *Napomena: Godina je prestupna kada je deljiva sa 400 ili kada je deljiva sa 4 a nije deljiva sa 100.*

Test primeri:

INTERAKCIJA SA PROGRAMOM:

Unesite godinu: 2018
Unesite mesec: 13
Greska: neispravan unos
meseca.

INTERAKCIJA SA PROGRAMOM:

Unesite godinu: 2000
Unesite mesec: 2
Februar, 29 dana

INTERAKCIJA SA PROGRAMOM:

Unesite godinu: 2018
Unesite mesec: 1
Januar, 31 dan

Zadaci za vežbanje (4)

- Napisati program koji za uneti pozitivan trocifreni broj ispisuje njegove cifre jedinica, desetica i stotina. *Napomena: Pretpostaviti da je unos ispravan, tj. da je uneti broj zaista trocifren.*

Test primeri:

INTERAKCIJA SA PROGRAMOM:

Unesite trocifreni broj: 697

Cifra jedinica: 7

Cifra desetica: 9

Cifra stotina: 6

INTERAKCIJA SA PROGRAMOM:

Unesite trocifreni broj: 504

Cifra jedinica: 4

Cifra desetica: 0

Cifra stotina: 5

Zadaci za vežbanje (5)

- Napisati program koji za uneti pozitivan četvorocifreni broj računa njegove cifre jedinica, desetica i stotina a potom ispisati najveću cifru. Program takođe treba ispisati grešku ukoliko uneti broj nije četvorocifren.

Test primeri:

INTERAKCIJA SA PROGRAMOM:

Unesite četvorocifreni broj: 6835

Najveća cifra je: 8

INTERAKCIJA SA PROGRAMOM:

Unesite četvorocifreni broj: 7777

Najveća cifra je: 7

INTERAKCIJA SA PROGRAMOM:

Unesite četvorocifreni broj: 238

Greska: niste uneli četvorocifreni broj.

INTERAKCIJA SA PROGRAMOM:

Unesite četvorocifreni broj: -2002

Najveća cifra je: 2