

Tipovi, promenljive i stringovi

Kako program pamti vrednosti, računa sa njima i slaže tekst

Danas pravimo: karticu lika za igru, sa bojom i imenima fajlova



#FFC857

heroj_007.png

Platno: 80.00 × 49.44 cm

Lik: ZMAJ

Tri pitanja za zagrevanje

1 • prošla tema

```
print("x", end="-")  
print("y")
```

Ispis: x-y

2 • računarski sistemi

Koliko različitih vrednosti
može da zapiše jedan
bajt?

Odgovor: 256

3 • mešano

```
# print(1)  
print(2, 3, sep=" ")
```

Ispis: 23

Pet novih reči za danas

promenljiva

ime pod kojim program pamti neku vrednost

dodela

znak = daje imenu vrednost: sirina = 800

tip

vrsta vrednosti: ceo broj, decimalni broj, tekst...

izraz

deo koda koji daje vrednost, na primer $2 + 3$

string (niska)

tekst u navodnicima, niz znakova

Promenljiva je ime koje pokazuje na vrednost



sirina = 800

Čitamo: „neka ime sirina pokazuje na 800“.

Promenljiva nastaje prvom dodelom.

Ime koje još nije dobilo vrednost ne postoji.

Ime je kao oznaka na fioci: vrednost možemo da zamenimo, a oznaka ostaje ista.

Nova dodela menja vrednost, pa i tip

```
zivoti = 3
print(zivoti, type(zivoti))
zivoti -= 1 # isto što i zivoti = zivoti - 1
print(zivoti)
zivoti = "nema"
print(zivoti, type(zivoti))
```

ISPIS

```
3 <class 'int'>
2
nema <class 'str'>
```

- type(...) kaže kog je tipa vrednost.
- zivoti -= 1 je kraći zapis za zivoti = zivoti - 1.
- Isto ime sada pokazuje na string.

Ime: slova, cifre i podvlaka, ne cifra na početku

Može

`sirina_platna` snake_case: preporuka

`boja2` cifra, ali ne na početku

`Ime` nije isto što i ime

Ne može

`2igrac` počinje cifrom

`moja-boja` crtica je minus

`for` ključna reč

Preporuka: mala slova, reči spojene podvlakom i samo slova engleske abecede, na primer `predjeni_put`. Slova č, ć, š, ž i đ u imenu rade, ali izbegavamo ih.

Više promenljivih dobija vrednosti u jednom redu

```
x, y, z = "trougao", "kvadrat", "kružnica"  
print(x)  
print(y)  
print(z)
```

ISPIS

```
trougao  
kvadrat  
kružnica
```

A ovako?

```
x, y, z = "trougao"
```

PORUKA O GREŠCI

```
ValueError: too many values to unpack (expected 3)
```

Čudovište: NameError

ime?

Kako izgleda

NameError: name 'sirna' is not defined. Python često doda: Did you mean: 'sirina'?

Zašto se pojavljuje

Ime nije dobilo vrednost pre upotrebe, ili je pogrešno otkucano: slovo više, slovo manje, veliko umesto malog.

Kako se pobeđuje

Uporedimo ime iz poruke sa imenom iz dodele, slovo po slovo. Proverimo da dodela stoji iznad upotrebe.

Tipovi i operatori

Koje vrednosti postoje i šta sa njima radimo

Četiri osnovna tipa: int, float, bool i str

```
print(type(7), type(3.5))  
print(type(True), type("7"))  
print(2 ** 100)  
print(0.1 + 0.2)
```

ISPIS

```
<class 'int'> <class 'float'>  
<class 'bool'> <class 'str'>  
1267650600228229401496703205376  
0.30000000000000004
```

int

ceo broj: 7, -3. Nema granicu.

float

decimalni broj, sa tačkom: 3.5

bool

logička vrednost: True ili False

str

string, tekst u navodnicima: "7"

/ uvek daje float, a // odbacuje decimale

```
x, y = 3, 2
print(x + y)
print(x - y)
print(x * y)
print(x / y)
print(x // y)
print(x % y)
print(x ** y)
print(-x, abs(-x))
print(4 / 2)
print(2 + 3 * 4)
print(2 ** 3 ** 2)
```

ISPIS

```
5
1
6
1.5
1
1
9
-3 3
2.0
14
512
```

/

deljenje, rezultat je uvek float: 4 / 2 je 2.0

//

celobrojno deljenje, zaokružuje naniže

%

ostatak pri deljenju

**

stepenovanje; 2 ** 3 ** 2 se računa zdesna

Redosled: zagrade, pa **, pa * / // %, pa + i -. Operator <> ne postoji: različito je !=.

Šta ispisuje ovaj program?

```
print(9 // 2, 9 % 2, 9 / 3)
```

Tri operatora, tri vrednosti. Pazite na tip svake vrednosti.

A

4 1 3

B

4.5 1 3.0

C

tačno

4 1 3.0

D

4 0 3.0

Ulaz, ispis i konverzije

Kada tekst postaje broj, a broj tekst

input uvek vraća string, čak i kad unesemo broj

```
broj = input("Unesite broj: ")  
print(type(broj))  
print(broj * 2)  
print(int(broj) * 2)
```

ISPIS

```
Unesite broj: 7  
<class 'str'>  
77  
14
```

- Tekst u zagradama je poruka korisniku.
- "7" * 2 ponavlja string: 77.
- int(broj) pravi broj, pa je 7 * 2 = 14.

int, float, str i bool prave vrednost drugog tipa

```
print(int("42") + 1)
print(float("2.5") * 2)
print(str(7) + " života")
print(int(3.9), float(3))
print(bool(0), bool(""), bool("0"))
```

ISPIS

```
43
5.0
7 života
3 3.0
False False True
```

- `int(3.9)` odbacuje decimale: 3, a ne 4.
- Neistiniti su `0`, prazan string i `False`; `"0"` je neprazan string, pa je istinit.

Šta ispisuje ovaj program?

```
print("5" * 3, 5 * 3)
```

Isti znak *, dve različite vrednosti levo od njega.

A

15 15

B

tačno

555 15

C

555 555

D

TypeError

Čudovište: TypeError

"5"+5

Kako izgleda

TypeError: can only concatenate str (not "int") to str. Operacija ne radi sa ovim tipovima.

Zašto se pojavljuje

Spajamo string i broj, najčešće zato što je vrednost došla iz input() i nismo je pretvorili u broj.

Kako se pobeđuje

Odlučimo šta želimo: račun, pa `int("5") + 5` daje 10, ili tekst, pa `"5" + str(5)` daje 55.

Čudovište: ValueError

```
int("abc")
```

Kako izgleda

ValueError: invalid literal for int() with base 10: 'abc'. Poruka navodi vrednost koja smeta.

Zašto se pojavljuje

Tip je dobar, ali vrednost nije: string "abc" ili "7.5" ne može da postane ceo broj.

Kako se pobeđuje

Proverimo šta je korisnik uneo. Za decimalni broj koristimo float, a za ceo broj unosimo samo cifre.

format umeće vrednosti na mesta vitičastih zagrada

```
vreme = int(input("Vreme (h): "))  
brzina = int(input("Brzina (km/h): "))  
predjeni_put = brzina * vreme  
print("Za {} h pređe se {} km.".format(vreme, predjeni_put))  
poruka = "Brzinom od {1} km/h se za {0} h pređe {2} km."  
print(poruka.format(vreme, brzina, predjeni_put))
```

ISPIS

```
Vreme (h): 2  
Brzina (km/h): 60  
Za 2 h pređe se 120 km.  
Brzinom od 60 km/h se za 2 h pređe 120 km.
```

{} popunjava se redom.

{1} je vrednost sa indeksom 1, druga po redu.

f-string: izraz u { } se izračuna i umetne

```
ime = "Mila"  
zivoti = 3  
print(f"{ime} ima {zivoti} života")  
print(f"Dupli životi: {zivoti * 2}")  
print(f"{7 / 3:.2f}")
```

ISPIS

```
Mila ima 3 života  
Dupli životi: 6  
2.33
```

- Slovo f ide ispred navodnika.
- Unutar { } može i račun: {zivoti * 2}.
- :.2f znači dve decimale.

Platno po zlatnom preseku, na dve decimale

```
sirina = float(input("Širina platna (cm): "))  
fi = (1 + 5 ** 0.5) / 2  
visina = sirina / fi  
print(f"Visina: {visina:.2f} cm")  
print(f"Platno: {sirina:.2f} × {visina:.2f} cm")
```

ISPIS

```
Širina platna (cm): 80  
Visina: 49.44 cm  
Platno: 80.00 × 49.44 cm
```



širina : visina = $fi \approx 1.618$

Format posle dvotačke: boja i ime fajla

Boja iz RGB u heksadecimalni zapis

```
r, g, b = 255, 200, 87  
print(f"#{r:02X}{g:02X}{b:02X}")
```

ISPIS

#FFC857



:02X = dve heksadecimalne cifre, velika slova, nula ispred

Ime fajla za kadar sprajta

```
kadar = 7  
print(f"heroj_{kadar:03}.png")  
kadar = 12  
print(f"heroj_{kadar:03}.png")
```

ISPIS

heroj_007.png
heroj_012.png

:03 = najmanje tri cifre, nule ispred

Stringovi

Tekst kao niz znakova

Svaki znak ima indeks; prvi je 0, poslednji -1

indeks

0

1

2

3

4

5



od kraja

-6

-5

-4

-3

-2

-1

```
s = "PIKSEL"  
print(s[0], s[3], s[-1])
```

ISPIS

P S L

Isecanje: s[od:do], a gornja granica se ne uključuje

```
s = "Programiranje za umetnike 1"  
print(s[0])  
print(s[1:3])  
print(s[-3:-1])  
print(s[-3:])  
print(s[:13])  
print(len(s))
```

ISPIS

```
P  
r  
o  
e  
e 1  
Programiranje  
27
```

- s[1:3] daje znakove sa indeksima 1 i 2.
- Bez prve granice kreće od početka, bez druge ide do kraja.
- Granice se razdvajaju dvotačkom, ne zarezom.

Šta ispisuje ovaj program?

```
rec = "animacija"  
print(rec[2:5], rec[-3:])
```

Pomoć: napišite indekse ispod slova reči animacija.

A tačno
ima ija

B
imac ija

C
nim ija

D
ima ja

Metode stringa prave nov string, a stari ostaje isti

```
y = "    Ovo je primer    \t\n "  
print(y.strip())  
print("Ovo JE pRimER!".lower())  
print("Ovo JE pRimER!".upper())  
print("Ovo JE pRimER".replace("JE", "i nije neki"))  
print(len("Ovo je primer..."))  
print("JE" in "Ovo JE pRimER")
```

ISPIS

```
Ovo je primer  
ovo je primer!  
OVO JE PRIMER!  
Ovo i nije neki pRimER  
16  
True
```

Metoda je funkcija koja se poziva nad vrednošću, sa tačkom: tekst.upper(). Funkcija len se piše ispred: len(tekst).

+ spaja stringove bez razmaka, a * ponavlja string

```
naslov = "Programiranje za umetnike"  
print(naslov + " " + "1")  
print("na" + "primer")  
print("na" + " " + "primer")  
print("-" * 20)  
print("ha" * 3 + "!")
```

ISPIS

```
Programiranje za umetnike 1  
naprimer  
na primer  
-----  
hahaha!
```

- "na" + "primer" daje naprimer: razmak dodajemo sami.
- Posebni znakovi: \n novi red, \t tabulator, \\ kosa crta.

Pravimo karticu lika za igru

Cilj: za 15 minuta program pita za lika i ispisuje njegovu boju i imena fajlova za animaciju.

- 1 Unos imena; strip uklanja višak razmaka
- 2 Unos tri broja za boju i broja kadrova: `int(input())`
- 3 Ime velikim slovima: `upper()`
- 4 Boja: `f"#{r:02X}{g:02X}{b:02X}"`
- 5 Prvi i poslednji kadar: `lower()` i `:03`

KAKO IZGLEDA

```
Ime lika:    Zmaj
Crvena (0-255): 120
Zelena (0-255): 200
Plava (0-255): 80
Broj kadrova: 12
=====
Lik: ZMAJ
Boja: #78C850
Prvi kadar: zmaj_001.png
Poslednji:  zmaj_012.png
=====
```

Ceo program „Kartica lika“

```
ime = input("Ime lika: ").strip()
r = int(input("Crvena (0-255): "))
g = int(input("Zelena (0-255): "))
b = int(input("Plava (0-255): "))
kadrova = int(input("Broj kadrova: "))
print("=" * 28)
print(f"Lik: {ime.upper()}")
print(f"Boja: #{r:02X}{g:02X}{b:02X}")
print(f"Prvi kadar: {ime.lower()}_{1:03}.png")
print(f"Poslednji: {ime.lower()}_{kadrova:03}.png")
print("=" * 28)
```

.strip()

„ Zmaj “ postaje „Zmaj“

:02X

120 postaje 78

:03

1 postaje 001

Boja: #78C850

Pronađi grešku

```
rec = "sprajt"  
print(rec[1,3])
```

Želimo da ispišemo pr, prva dva slova posle s.
Kako glasi ispravan drugi red?

PORUKA O GREŠCI

```
Traceback (most recent call last):  
  File "sprajt.py", line 2, in <module>  
    print(rec[1,3])  
            ~~~^
```

TypeError: string indices must be integers, not 'tuple'

line 2

red u kom je greška

TypeError

pogrešan tip unutar []

'tuple'

1,3 je par brojeva, a ne isečak

Izaberite nivo i počnite

LEVEL 1

Platno

Unesite širinu i visinu platna u centimetrima. Ispišite površinu i obim na dve decimale.

Pomoć: `float(input(...))` i `:.2f`

LEVEL 2

Inicijali

Unesite ime i prezime, malim slovima i sa razmacima. Ispišite inicijale, na primer M. J.

Pomoć: `strip()`, `[0]` i `upper()`

LEVEL 3

Ime fajla

Unesite lika, akciju i broj kadra. Ispišite ime fajla bez velikih slova i razmaka: `crni_zmaj_trči_004.png`

Pomoć: `:03`, `lower()` i `replace(" ", "_")`

Sledeći put: program bira šta će uraditi

```
zivoti = 0
if zivoti == 0:
    print("Kraj igre")
else:
    print(f"Još {zivoti} života")
```

ISPIS

Kraj igre

U temi 07 učimo uslovna grananja: if, elif i else. Dvotačka i uvlačenje su isti kao u temi 05.

Rezime

Promenljive i tipovi

```
sirina = 800  
type(3.5)    # float
```

Promenljiva nastaje dodelom.

Operatori

```
7 / 2    # 3.5  
7 // 2   # 3      7 % 2   # 1
```

/ uvek daje float.

Ulaz i konverzije

```
n = int(input("Broj: "))
```

input uvek vraća string.

Stringovi

```
s[0]    s[1:3]    s.upper()  
f"{x:.2f}"    f"{k:03}"
```

Gornja granica se ne uključuje.

Pitanje pre odlaska

```
x = "3"  
print(x * 2 + "1")
```

Šta tačno ispisuje ovaj program?

Odgovor pošaljite pre nego što izađete. Jedan red je dovoljan.



Literatura

- Python dokumentacija, Tutorial, poglavlje „An Informal Introduction to Python“ (brojevi i stringovi): docs.python.org/3/tutorial
- Python dokumentacija, „Format Specification Mini-Language“ (zapisi `:.2f`, `:03`, `:02X`): docs.python.org/3/library/string.html
- Wentworth, P., Elkner, J., Downey, A. B., Meyers, C. How to Think Like a Computer Scientist: Learning with Python 3 (besplatna knjiga)
- Sweigart, A. Invent Your Own Computer Games with Python, 4. izdanje, No Starch Press (besplatno na inventwithpython.com)

Dodatna literatura i zanimljivi sajtovi

Petlja: Priručnik za Python

petlja.org · Priručnik za Python

Besplatan priručnik na srpskom, sa zadacima po nivoima težine.

Zvanični Python tutorijal

docs.python.org/3/tutorial

Tutorijal iz zvanične dokumentacije, na engleskom.

W3Schools: Python

w3schools.com/python

Kratke lekcije sa primerima koje menjate i pokrećete u pregledaču.

Python Tutor

pythontutor.com

Crta korak po korak šta program radi i šta je u kojoj promenljivoj.

Hvala!

Pitanja su uvek dobrodošla: na času, u četvrtu ili međuvremeno.

Slajdovi, primeri i kartica za podsećanje su na sajtu predmeta.

PREDAVANJA

dr Staša Vujičić Stanković

stasa.vujicic.stankovic@math.rs

www.matf.bg.ac.rs/~stasa.vujicic.stankovic

VEŽBE

Nevena Ćirić

poincare.matf.bg.ac.rs/~nevena.ciric/p1uu.html