

Funkcije i moduli

Kako da delu koda damo ime, pa ga pozovemo kad god zatreba

Danas pravimo: generator nasumičnih boja i svoj modul paleta.py

```
*****  
* def *  
*****  
~~~~~  
~ return ~  
~~~~~  
#####  
# import #  
#####
```

Tri pitanja za zagrevanje

1 • prošla tema: kolekcije

```
d = {"x": 3}
print(d.get("y", 0))
```

Ispis: 0

2 • liste

```
a = [4, 7, 1]
a.sort()
print(a[0])
```

Ispis: 1

3 • mešano

```
s = set()
for c in "anana":
    s.add(c)
print(len(s))
```

Ispis: 2

Pet novih reči za danas

funkcija deo koda sa imenom, koji se izvršava kad ga pozovemo

parametar ime u zagradama definicije, koje prima vrednost

argument vrednost koju funkciji damo pri pozivu

povratna vrednost rezultat koji funkcija vraća naredbom return

modul fajl .py sa funkcijama koje uvozimo u drugi program

Funkcije već koristimo, a danas pišemo svoje

ugrađene

```
print(len("Cikica"))
```

Postoje u Pythonu: print, len, input, range.

iz modula

```
import math  
print(math.sqrt(16))
```

Uvozimo ih naredbom import: math, random.

naše

```
def pozdrav():  
    print("Zdravo!")
```

Pišemo ih sami, posle reči def.

Pišemo svoju funkciju

def, ime, zagrade i dvotačka

Funkciju prvo definišemo, a zatim pozivamo

```
def naziv_funkcije():  
    print("Zdravo!!!")
```

```
naziv_funkcije()  
naziv_funkcije()
```

ISPIS

```
Zdravo!!!  
Zdravo!!!
```

- def kaže: ovde počinje definicija funkcije.
- Uvučeni red je telo funkcije.
- naziv_funkcije() je poziv: zagrade su obavezne.

Šta ispisuje ovaj program?

```
def crtaj():  
    print("krug")  
  
print("početak")
```

Brojte pozive funkcije crtaj.

A

krug
početak

B

tačno

početak

C

početak
krug

D

NameError

Parametar je u definiciji, argument je u pozivu

```
def pozdrav(ime, prezime):  
    print(f"Ja sam {ime} {prezime}!")
```

```
pozdrav("Pera", "Perić")  
pozdrav("Laza", "Lazić")
```

ISPIS

```
Ja sam Pera Perić!  
Ja sam Laza Lazić!
```

parametri

ime, prezime

imena u zagradama definicije

argumenti

"Pera", "Perić"

vrednosti u zagradama poziva

Pri pozivu ime dobija "Pera", a prezime dobija "Perić", tim redom.

Broj argumenata mora da odgovara broju parametara

```
def pozdrav(ime, prezime):  
    print(f"Ja sam {ime} {prezime}!")  
  
pozdrav("Pera")
```

PORUKA O GREŠCI

```
TypeError: pozdrav() missing 1 required  
positional argument: 'prezime'
```

Funkcija ima dva parametra,
a pozvana je sa jednim
argumentom. Šta će Python reći?

missing 1 required positional
argument: nedostaje jedan obavezan
argument: prezime.

return vraća rezultat onome ko je pozvao funkciju

```
def površina(sirina, visina):  
    return sirina * visina
```

```
p = površina(1920, 1080)  
print(p)  
print(površina(3, 4) + 1)
```

ISPIS

```
2073600  
13
```

Površina platna Full HD: 1920 × 1080 piksela.

Rezultat poziva možemo da sačuvamo u promenljivoj ili da ga odmah koristimo u izrazu, kao `površina(3, 4) + 1`.

Funkcija ispisuje, ali da li vraća?

```
def udvostruci(x):  
    print(x * 2)  
  
y = udvostruci(5)  
print(y)
```

Pogledajte: ima li u funkciji return?

A

10
10

B

tačno

10
None

C

None

D

10

Čudovište: funkcija bez return vraća None

None

Kako izgleda

Umesto rezultata vidimo None ili, kasnije u računu, `TypeError: unsupported operand type(s) for +: 'NoneType' and 'int'`.

Zašto se pojavljuje

Funkcija rezultat ispisuje sa `print`, ali ga ne vraća, ili `return` postoji samo u jednoj grani `if`.

Kako se pobeđuje

Proverimo da li svaki put kroz funkciju završava sa `return` i vrednošću. Ispis i vraćanje su dve različite stvari.

Podrazumevana vrednost parametra

```
def okvir(tekst, znak="*"):  
    sirina = len(tekst) + 4  
    print(znak * sirina)  
    print(znak, tekst, znak)  
    print(znak * sirina)
```

```
okvir("PZU1")  
okvir("Kolekcije", "#")
```

ISPIS

```
*****  
* PZU1 *  
*****  
#####  
# Kolekcije #  
#####
```

znak="*" u definiciji

znači: ako znak ne navedemo, koristi se zvezdica.

Imenovani argumenti mogu da idu bilo kojim redom

```
def pozdrav(ime, prezime, broj_godina=30):  
    print(f"Ja sam {ime} {prezime}. Imam {broj_godina} godina.")
```

```
pozdrav(prezime="Perić", broj_godina=20, ime="Pera")  
pozdrav("Laza", "Lazić")  
pozdrav("Mika", "Mikić", 15)
```

ISPIS

```
Ja sam Pera Perić. Imam 20 godina.  
Ja sam Laza Lazić. Imam 30 godina.  
Ja sam Mika Mikić. Imam 15 godina.
```

ime="Pera" je imenovani argument.

Uz ime parametra redosled u pozivu nije važan. Laza nije naveo godine, pa dobija podrazumevanih 30.

Šta vraća ovaj poziv?

```
def boja(r, g=0, b=0):  
    return (r, g, b)  
  
print(boja(255, b=100))
```

Jedan argument je po redu, a jedan je imenovani.

A

(255, 100, 0)

B

tačno

(255, 0, 100)

C

(255, 100)

D

TypeError

Imenovanim argumentom biramo šta menjamo

```
def zvezda(x, y, velicina=3, boja="bela"):
    print("zvezda", x, y, velicina, boja)
```

```
zvezda(10, 20)
zvezda(10, 20, 5)
zvezda(10, 20, boja="žuta")
```

```
zvezda(10, 20, "žuta")
zvezda(10, 20, boja="plava", velicina=5)
```

Bez imena, "žuta" zauzima mesto parametra velicina. Sa imenima redosled nije važan.

ISPIS

```
zvezda 10 20 3 bela
zvezda 10 20 5 bela
zvezda 10 20 3 žuta
```

ISPIS

```
zvezda 10 20 žuta bela
zvezda 10 20 5 plava
```

Funkcija može da bude argument druge funkcije

```
def racunarska_operacija(operacija, broj):  
    rezultat = operacija(broj)  
    return rezultat  
  
def uvecanje_za_1(broj):  
    return broj + 1  
  
def umanjenje_za_1(broj):  
    return broj - 1  
  
print(racunarska_operacija(uvecanje_za_1, 5))  
print(racunarska_operacija(umanjenje_za_1, 5))
```

ISPIS

6
4

uvecanje_za_1 bez zagrada

je sama funkcija, a ne njen rezultat. Funkcija `racunarska_operacija` je poziva u svom telu.

Pravimo generator nasumičnih boja i tačaka

Cilj: za 15 minuta imamo dve funkcije koje ćemo u PyGame-u koristiti za crtanje.

- 1 `import random`
- 2 `nasumicna_boja()`: vraća torku (r, g, b)
- 3 `nasumicne_koordinate(sirina, visina)`
- 4 Petlja: tri tačke sa bojama
- 5 Dodatak: `zvezdano_nebo(broj=3)`

JEDAN ISPIS

```
(383, 311) (58, 159, 254)
(287, 397) (77, 30, 226)
(794, 114) (1, 38, 83)
```

Brojevi su nasumični: svako pokretanje daje druge tačke i boje.

Ceo program sa dve funkcije

```
import random

def nasumicna_boja():
    return (random.randint(0, 255),
            random.randint(0, 255),
            random.randint(0, 255))

def nasumicne_koordinate(sirina, visina):
    return (random.randint(0, sirina),
            random.randint(0, visina))

for i in range(3):
    tacka = nasumicne_koordinate(800, 600)
    print(tacka, nasumicna_boja())
```

nasumicna_boja()

bez parametara; vraća torku od tri broja

nasumicne_koordinate

dva parametra; vraća torku (x, y)

random.randint(a, b)

ceo broj od a do b, obe granice uključene

Gde živi koje ime

Lokalno, globalno i pravilo LEGB

Promenljiva iz funkcije ne postoji van nje

```
def postavi():  
    boja = "crvena"  
    print("u funkciji:", boja)  
  
postavi()  
print(boja)
```

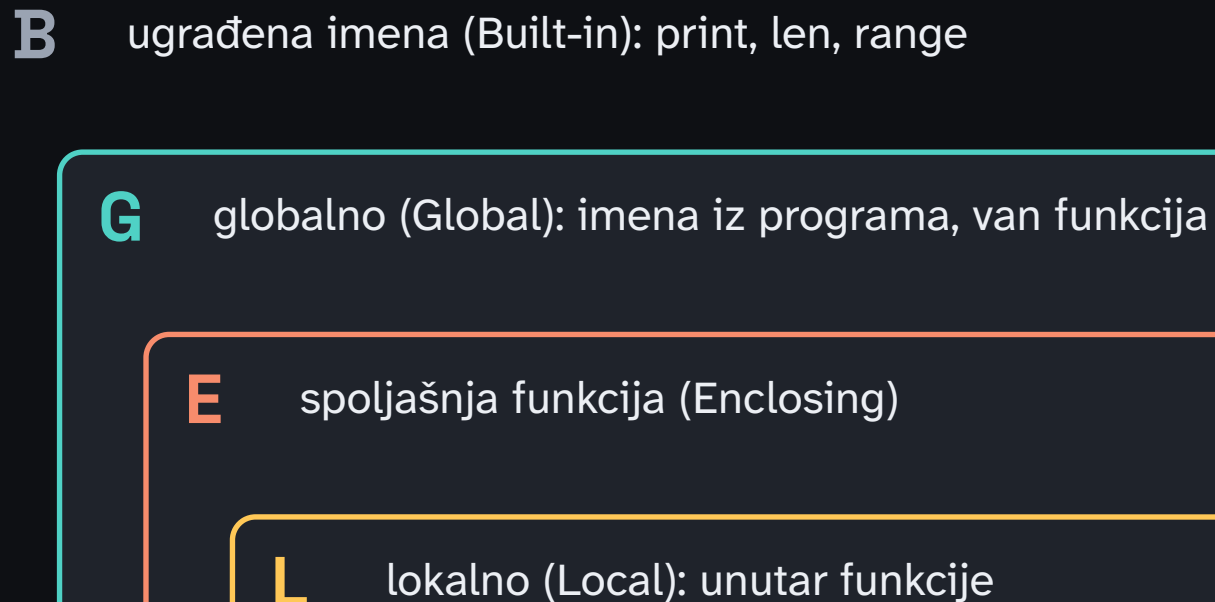
ISPIS

```
u funkciji: crvena  
NameError: name 'boja'  
is not defined
```

boja je lokalna promenljiva.

Nastaje kad se funkcija pozove, a nestaje kad se funkcija završi. Van funkcije to ime ne postoji.

Pravilo LEGB: ime se traži iznutra ka spolja



↑
4. **B**
3. **G**
2. **E**
1. **L**

Python uzima prvo ime koje nađe. Ako ga nema ni u B, javlja `NameError`.

Dodela pravi novo lokalno ime, a čitanje traži spolja

Unutra je dodela: novo ime (L)

```
def nova_funkcija():  
    promenljiva = 10  
    def untrasnja_funkcija():  
        promenljiva = 20  
        print("unutra:", promenljiva)  
    untrasnja_funkcija()  
    print("spolja:", promenljiva)
```

```
nova_funkcija()
```

ISPIS

```
unutra: 20  
spolja: 10
```

Unutra nema dodele: ime iz spoljašnje (E)

```
def nova_funkcija():  
    promenljiva = 10  
    def untrasnja_funkcija():  
        print("unutra:", promenljiva)  
    untrasnja_funkcija()  
    print("spolja:", promenljiva)
```

```
nova_funkcija()
```

ISPIS

```
unutra: 10  
spolja: 10
```

Koje x ispisuje funkcija unutra?

```
x = "globalno"

def spolja():
    x = "spolja"
    def unutra():
        print(x)
    unutra()

spolja()
```

Pratite pravilo LEGB.

A

globalno

B

tačno

spolja

C

NameError

D

x

global treba samo kad funkcija menja globalno ime

Čitanje: radi

```
zivot = 3

def prikazi():
    print("život", zivot)
prikazi()
```

ISPIS

život 3

Dodela bez global: greška

```
zivot = 3

def udarac():
    zivot -= 1
udarac()
print(zivot)
```

PORUKA O GREŠCI

```
UnboundLocalError: cannot
access local variable
'zivot' where it is not
associated with a value
```

Dodela sa global: radi

```
zivot = 3

def udarac():
    global zivot
    zivot -= 1
udarac()
print(zivot)
```

ISPIS

2

Moduli

Funkcije iz drugog fajla

Tri načina da uvezemo modul

import math

ime modula, tačka, funkcija

```
import math  
print(math.sqrt(100))
```

10.0

import math as m

kraće ime za modul

```
import math as m  
print(m.floor(2.7))
```

2

from math import pi

samo jedno ime, bez tačke

```
from math import pi  
print(round(pi, 2))
```

3.14

Standardna biblioteka: math i random su već tu

```
import random

print(random.randint(1, 6))
print(random.choice(["crvena", "plava", "žuta"]))
print(random.random())
```

JEDAN ISPIS

```
5
žuta
0.8150524343095853
```

math.sqrt(x)

kvadratni koren

math.pibroj π **random.randint(a, b)**

ceo broj od a do b, uključujući b

random.choice(lista)

jedan element liste

random.random()

realan broj od 0 do 1

Svoj modul: paleta.py uvozimo kao math

paleta.py

```
# paleta.py: naš modul sa bojama
CRNA = (0, 0, 0)
BELA = (255, 255, 255)
NARANDZASTA = (255, 140, 0)

def posvetli(boja, koliko):
    r, g, b = boja
    return (min(r + koliko, 255),
            min(g + koliko, 255),
            min(b + koliko, 255))
```

min(a, b) vraća manji broj, da boja ne pređe 255.

crtez.py (u istom folderu)

```
import paleta
from paleta import posvetli

print(paleta.NARANDZASTA)
print(posvetli(paleta.CRNA, 60))
print(posvetli(paleta.NARANDZASTA, 200))
```

ISPIS

```
(255, 140, 0)
(60, 60, 60)
(255, 255, 200)
```

Paket je folder sa modulima

STRUKTURA PROJEKTA

```
igra/  
    __init__.py    nije obavezan  
    grafika.py  
    zvuk.py  
glavni.py
```

U glavni.py: `from igra import grafika`

TERMINAL

```
$ pip install pygame-ce
```

- modul: jedan .py fajl
- paket: folder sa modulima
- biblioteka: gotov kod za neku oblast, npr. PyGame

pip preuzima pakete sa interneta (sajt PyPI) i instalira ih, da bismo mogli da pišemo `import pygame`.

Pronađi grešku

```
nacrtaj_zvezdu(5)

def nacrtaj_zvezdu(n):
    print("*" * n)
```

Program treba da nacрта pet zvezdica. Šta treba premestiti?

PORUKA O GREŠCI

```
Traceback (most recent call last):
  File "zvezda.py", line 1, in <module>
    nacrtaj_zvezdu(5)
    ^^^^^^^^^^^^^^^^^
NameError: name 'nacrtaj_zvezdu'
is not defined
```

line 1

greška je u prvom redu, u pozivu

NameError

ime još ne postoji

is not defined

def se još nije izvršio

Izaberite nivo i počnite

LEVEL 1

Obim platna

Napišite funkciju `obim(sirina, visina)` koja vraća obim pravougaonog platna. Pozovite je za dva platna.

Pomoć: `return 2 * (...)`

LEVEL 2

Siva boja

Funkcija `sivo(boja)` prima torku `(r, g, b)` i vraća sivu boju: sva tri broja su prosek.

Pomoć: `r, g, b = boja` i `//`

LEVEL 3

Mešanje boja

Funkcija `mesaj(boja1, boja2, udeo=0.5)` meša dve boje uvezene iz modula `paleta.py`.

Pomoć: `from paleta import CRNA, BELA`

Iste dve funkcije uskoro boje 200 tačaka

```
for i in range(200):  
    boja = nasumicna_boja()  
    tacka = nasumicne_koordinate(SIRINA, VISINA)  
    pg.draw.circle(ekran, boja, tacka, 4)
```

U sledećoj temi, o PyGame-u, `nasumicna_boja()` i `nasumicne_koordinate()` daju boju i mesto svakog kruga u prozoru. Slika desno je nacrtana baš tim funkcijama.



Rezime

def i return

```
def površina(s, v):  
    return s * v
```

Bez return funkcija vraća None.

argumenti

```
okvir("PZU1", znak="#")  
zvezda(10, 20, boja="žuta")
```

Podrazumevane vrednosti i imenovani argumenti.

prostor imena

```
L → E → G → B  
global zivot
```

global samo kad funkcija dodeljuje globalnom imenu.

moduli

```
import math as m  
from paleta import BELA
```

Modul je .py fajl, paket je folder sa modulima.

Pitanje pre odlaska

```
def f(x):  
    x = x + 1  
  
x = 5  
f(x)  
print(x)
```

Koji broj ispisuje program?

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



Literatura

- Python dokumentacija, Tutorial, poglavlja 4.8 „Defining Functions“ i 6 „Modules“:
docs.python.org/3/tutorial
- Barry, P. Head First Python, 2. izdanje, O'Reilly, 2016 (postoji prevod na srpski u izdanju CET-a),
prvo poglavlje
- 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.

Python Tutor

pythontutor.com

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

Zvanični Python tutorijal

docs.python.org/3/tutorial

Tutorijal iz zvanične dokumentacije, na engleskom.

Exercism: Python

exercism.org/tracks/python

Besplatni zadaci za vežbu, poređani po temama.

Hvala!

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

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