

Задатак 085: Креирати дугме за приказ целог садржаја базе података и функције за приказ свих записа на конзоли.

```

from tkinter import *
import sqlite3

class baza_podataka:
    def __init__(self, prozor1):
        self.link = sqlite3.connect('adresar.db')
        self.kursor = self.link.cursor()
        #self.kreiranje_tabele()
        self.postavljanje_lejbela_u_prozor1(prozor1)
        self.postavljanje_tekst_boksa_u_prozor1(prozor1)
        self.kreiranje_dugmeta_za_unos_podataka_u_bp(prozor1)
        self.kreiranje_dugmeta_za_prikaz_zapisa_iz_baze(prozor1)
        self.link.commit()
        self.link.close()

    def kreiranje_tabele(self):
        self.kursor.execute('''CREATE TABLE adrese(ime text, mesto text, godina integer)''')

    def unos(self):
        self.link = sqlite3.connect('adresar.db')
        self.kursor = self.link.cursor()
        self.kursor.execute("INSERT INTO adrese VALUES(:ime, :mesto, :godina)",
        {
            'ime': self.ime.get(),
            'mesto': self.mesto.get(),
            'godina': self.godina.get()
        })

        self.ime.delete(0, END)
        self.mesto.delete(0, END)
        self.godina.delete(0, END)

        self.link.commit()
        self.link.close()

    def upit_cela_baza(self):
        self.link = sqlite3.connect("adresar.db")
        self.kursor = self.link.cursor()
        self.kursor.execute("SELECT *, oid FROM adrese")
        self.zapisi = self.kursor.fetchall()
        print(self.zapisi)
        self.link.commit()
        self.link.close()

    def postavljanje_lejbela_u_prozor1(self, proz1):
        self.opis_lejbel = Label(proz1, text = "Pojedinačni podaci koji cine jedan zapis u bazi
podataka:")
        self.opis_lejbel.grid(row = 0, column = 0, columnspan = 2, sticky = W)
        self.ime_lejbel = Label(proz1, text = "Ime osobe")
        self.ime_lejbel.grid(row = 1, column = 0, sticky = W)
        self.mesto_lejbel = Label(proz1, text = "Mesto u kojem zivi osoba")
        self.mesto_lejbel.grid(row = 2, column = 0, sticky = W)
        self.godina_lejbel = Label(proz1, text = "Godina rodjenja osobe")
        self.godina_lejbel.grid(row = 3, column = 0, sticky = W)

    def postavljanje_tekst_boksa_u_prozor1(self, proz1):
        self.ime = Entry(proz1, width = 30)

```

```

self.ime.grid(row = 1, column = 1, padx = 20)
self.mesto = Entry(proz1, width = 30)
self.mesto.grid(row = 2, column = 1)
self.godina = Entry(proz1, width = 30)
self.godina.grid(row = 3, column = 1)

def kreiranje_dugmeta_za_unos_podataka_u_bp(self, proz1):
    self.dugme_unos = Button(proz1, text = "Unos zapisa u bazu podataka", command =
self.unos)
    self.dugme_unos.grid(row = 4, column = 0, columnspan = 2, pady = 10, padx = 10, ipadx =
100)

def kreiranje_dugmeta_za_prikaz_zapisa_iz_baze(self, proz1):
    self.dugme_upit_cela_baza = Button(proz1, text = "Prikaz svih zapisa u bazi podataka",
command = self.upit_cela_baza)
    self.dugme_upit_cela_baza.grid(row = 5, column = 0, columnspan = 2, pady = 10, padx =
10, ipadx = 88)

def main():
    osnovni_prozor = Tk()
    osnovni_prozor.title("Tkinter sa klasama")
    osnovni_prozor.iconbitmap("py_slika.ico")
    osnovni_prozor.geometry("400x400")
    bp = baza_podataka(osnovni_prozor)
    osnovni_prozor.mainloop()

main()

```

Задатак 086: Модификовати функцију `upit_cela_baza()` из задатка 085 тако да се прикаже садржај свих записа из базе података и на основном екрану корисничког интерфејса.

```

def upit_cela_baza(self):
    self.link = sqlite3.connect("adresar.db")
    self.kursor = self.link.cursor()
    self.kursor.execute("SELECT *, oid FROM adrese")
    self.zapisi = self.kursor.fetchall()
    print(self.zapisi)

    ispis_zapisa = ''
    for zapis in self.zapisi:
        ispis_zapisa += str(zapis) + "\n"
    self.upit_lejbel = Label(text = ispis_zapisa)
    self.upit_lejbel.grid(row = 6, column = 0, columnspan = 2)

    self.link.commit()
    self.link.close()

```

Задатак 087: Модификовати функцију `upit_cela_baza()` из задатка 086 тако да се прикаже прво редни број записа а затим име особе.

```

for zapis in self.zapisi:
    ispis_zapisa += str(zapis[3]) + " " + str(zapis[0]) + "\n"

```

Задатак 088: Модификовати код из задатка 087 тако да се омогући брисање одређеног записа преко уноса id особе.

```

from tkinter import *
import sqlite3

class baza_podataka:
    def __init__(self, prozor1):
        self.link = sqlite3.connect('adresar.db')
        self.kursor = self.link.cursor()
        #self.kreiranje_tabele()
        self.postavljanje_lejbela_u_prozor1(prozor1)
        self.postavljanje_tekst_boksa_u_prozor1(prozor1)
        self.kreiranje_dugmeta_za_unos_podataka_u_bp(prozor1)
        self.kreiranje_dugmeta_za_prikaz_zapisa_iz_baze(prozor1)
        self.kreiranje_dugmeta_za_brisanje_zapisa(prozor1)
        self.link.commit()
        self.link.close()

    def kreiranje_tabele(self):
        self.kursor.execute('''CREATE TABLE adrese(ime text, mesto text, godina integer)''')

    def brisanje(self):
        self.link = sqlite3.connect('adresar.db')
        self.kursor = self.link.cursor()
        self.kursor.execute("DELETE from adrese WHERE oid = " + self.box_za_brisanje.get())
        self.box_za_brisanje.delete(0, END)
        self.link.commit()
        self.link.close()

    def unos(self):
        self.link = sqlite3.connect('adresar.db')
        self.kursor = self.link.cursor()
        self.kursor.execute("INSERT INTO adrese VALUES(:ime, :mesto, :godina)",
        {
            'ime': self.ime.get(),
            'mesto': self.mesto.get(),
            'godina': self.godina.get()
        })

        self.ime.delete(0, END)
        self.mesto.delete(0, END)
        self.godina.delete(0, END)

        self.link.commit()
        self.link.close()

    def upit_cela_baza(self):
        self.link = sqlite3.connect("adresar.db")
        self.kursor = self.link.cursor()
        self.kursor.execute("SELECT *, oid FROM adrese")
        self.zapisi = self.kursor.fetchall()
        print(self.zapisi)

        ispis_zapisa = ''
        for zapis in self.zapisi:
            ispis_zapisa += str(zapis[3]) + " " + str(zapis[0]) + "\n"
        self.upit_lejbel = Label(text = ispis_zapisa)
        self.upit_lejbel.grid(row = 6, column = 0, columnspan = 2)

```

```

self.link.commit()
self.link.close()

def postavljanje_lejbel_a_u_prozor1(self, proz1):
    self.opis_lejbel = Label(proz1, text = "Pojedinačni podaci koji čine jedan zapis u bazi
podataka:")
    self.opis_lejbel.grid(row = 0, column = 0, columnspan = 2, sticky = W)
    self.ime_lejbel = Label(proz1, text = "Ime osobe")
    self.ime_lejbel.grid(row = 1, column = 0, pady = (10, 1))
    self.mesto_lejbel = Label(proz1, text = "Mesto u kojem živi osoba")
    self.mesto_lejbel.grid(row = 2, column = 0, sticky = W)
    self.godina_lejbel = Label(proz1, text = "Godina rođenja osobe")
    self.godina_lejbel.grid(row = 3, column = 0, sticky = W)
    self.box_za_brisanje_lejbel = Label(proz1, text = "Brisanje ID")
    self.box_za_brisanje_lejbel.grid(row = 7, column = 0, pady = 5)

def postavljanje_tekst_boksa_u_prozor1(self, proz1):
    self.ime = Entry(proz1, width = 30)
    self.ime.grid(row = 1, column = 1, padx = 20, pady = (10, 1))
    self.mesto = Entry(proz1, width = 30)
    self.mesto.grid(row = 2, column = 1)
    self.godina = Entry(proz1, width = 30)
    self.godina.grid(row = 3, column = 1)
    self.box_za_brisanje = Entry(proz1, width = 30)
    self.box_za_brisanje.grid(row = 7, column = 1, pady = 5)

def kreiranje_dugmeta_za_unos_podataka_u_bp(self, proz1):
    self.dugme_unos = Button(proz1, text = "Unos zapisa u bazu podataka", command =
self.unos)
    self.dugme_unos.grid(row = 4, column = 0, columnspan = 2, pady = 10, padx = 10, ipadx =
100)

def kreiranje_dugmeta_za_prikaz_zapisa_iz_baze(self, proz1):
    self.dugme_upit_cela_baza = Button(proz1, text = "Prikaz svih zapisa u bazi podataka",
command = self.upit_cela_baza)
    self.dugme_upit_cela_baza.grid(row = 5, column = 0, columnspan = 2, pady = 10, padx =
10, ipadx = 88)

def kreiranje_dugmeta_za_brisanje_zapisa(self, proz1):
    self.dugme_brisanje_zapisa = Button(proz1, text = "Brisanje upisanog broja zapisa",
command = self.brisanje)
    self.dugme_brisanje_zapisa.grid(row = 8, column = 0, columnspan = 2, pady = 10, padx =
10, ipadx = 88)

def main():
    osnovni_prozor = Tk()
    osnovni_prozor.title("Tkinter sa klasama")
    osnovni_prozor.iconbitmap("py_slika.ico")
    osnovni_prozor.geometry("400x400")
    bp = baza_podataka(osnovni_prozor)
    osnovni_prozor.mainloop()

main()

```

Задатак 089: Модификовати код из задатка 088 тако да се приликом избора измене унетих података из базе података отвара нов прозор у којем се врше све измене података. Кликом на дугме на том прозору се чувају унете измене и затвара се прозор.

```

from tkinter import *
import sqlite3

class baza_podataka:
    def __init__(self, prozor1):
        self.link = sqlite3.connect('adresar.db')
        self.kursor = self.link.cursor()
        #self.kreiranje_tabele()
        self.postavljanje_lejbela_u_prozor1(prozor1)
        self.postavljanje_tekst_boksa_u_prozor1(prozor1)
        self.kreiranje_dugmeta_za_unos_podataka_u_bp(prozor1)
        self.kreiranje_dugmeta_za_prikaz_zapisa_iz_baze(prozor1)
        self.kreiranje_dugmeta_za_brisanje_zapisa(prozor1)
        self.kreiranje_dugmeta_za_nadogradnju(prozor1)
        self.link.commit()
        self.link.close()

    def kreiranje_tabele(self):
        self.kursor.execute('''CREATE TABLE adrese(ime text, mesto text, godina integer)''')

    def nadogradnja(self):
        self.link = sqlite3.connect('adresar.db')
        self.kursor = self.link.cursor()
        self.zapis_id = self.box_za_brisanje.get()
        self.kursor.execute('''UPDATE adrese SET ime = :ime, mesto = :mesto, godina = :godina
WHERE oid = :oid''',
                                {
                                    'ime' : self.ime_prozor2.get(),
                                    'mesto' : self.mesto_prozor2.get(),
                                    'godina' : self.godina_prozor2.get(),
                                    'oid' : self.zapis_id})
        self.link.commit()
        self.link.close()
        prozor2.destroy()

    def izmena(self):
        global prozor2
        prozor2 = Tk()
        prozor2.title("Nadogradnja zapisa")
        prozor2.iconbitmap("py_slika.ico")
        prozor2.geometry("400x200")
        self.link = sqlite3.connect('adresar.db')
        self.kursor = self.link.cursor()
        self.zapis_id = self.box_za_brisanje.get()
        self.kursor.execute("SELECT * FROM adrese WHERE oid = " + self.zapis_id)
        self.zapisi = self.kursor.fetchall()

        self.ime_prozor2 = Entry(prozor2, width = 30)
        self.ime_prozor2.grid(row = 1, column = 1, padx = 20, pady = (10,1))
        self.mesto_prozor2 = Entry(prozor2, width = 30)
        self.mesto_prozor2.grid(row = 2, column = 1)
        self.godina_prozor2 = Entry(prozor2, width = 30)
        self.godina_prozor2.grid(row = 3, column = 1)

```

```

self.opis_lejbel_prozor2 = Label(prozor2, text = "Pojedinačni podaci koji čine jedan
zapis u bazi podataka:")
self.opis_lejbel_prozor2.grid(row = 0, column = 0, columnspan = 2, sticky = W)
self.ime_lejbel_prozor2 = Label(prozor2, text = "Ime osobe")
self.ime_lejbel_prozor2.grid(row = 1, column = 0, pady = (10, 1))
self.mesto_lejbel_prozor2 = Label(prozor2, text = "Mesto u kojem živi osoba")
self.mesto_lejbel_prozor2.grid(row = 2, column = 0, sticky = W)
self.godina_lejbel_prozor2 = Label(prozor2, text = "Godina rođenja osobe")
self.godina_lejbel_prozor2.grid(row = 3, column = 0, sticky = W)

for zapis in self.zapisi:
    self.ime_prozor2.insert(0, zapis[0])
    self.mesto_prozor2.insert(0, zapis[1])
    self.godina_prozor2.insert(0, zapis[2])

self.dugme_nadogradnja = Button(prozor2, text = "Memorisanje zapisa", command =
self.nadogradnja)
self.dugme_nadogradnja.grid(row = 6, column = 0, columnspan = 2, pady = 10, padx = 10,
ipadx = 88)

self.link.commit()
self.link.close()

def brisanje(self):
    self.link = sqlite3.connect('adresar.db')
    self.kursor = self.link.cursor()
    self.kursor.execute("DELETE from adrese WHERE oid = " + self.box_za_brisanje.get())
    self.box_za_brisanje.delete(0, END)
    self.link.commit()
    self.link.close()

def unos(self):
    self.link = sqlite3.connect('adresar.db')
    self.kursor = self.link.cursor()
    self.kursor.execute("INSERT INTO adrese VALUES(:ime, :mesto, :godina)",
    {
        'ime': self.ime.get(),
        'mesto': self.mesto.get(),
        'godina': self.godina.get()
    })

    self.ime.delete(0, END)
    self.mesto.delete(0, END)
    self.godina.delete(0, END)

    self.link.commit()
    self.link.close()

def upit_cela_baza(self):
    self.link = sqlite3.connect("adresar.db")
    self.kursor = self.link.cursor()
    self.kursor.execute("SELECT *, oid FROM adrese")
    self.zapisi = self.kursor.fetchall()
    print(self.zapisi)

ispis_zapisa = ''
for zapis in self.zapisi:
    ispis_zapisa += str(zapis[3]) + " " + str(zapis[0]) + "\n"

```

```

self.upit_lejbel = Label(text = ispis_zapisa)
self.upit_lejbel.grid(row = 6, column = 0, columnspan = 2)

self.link.commit()
self.link.close()

def postavljanje_lejbela_u_prozor1(self, proz1):
    self.opis_lejbel = Label(proz1, text = "Pojedinačni podaci koji čine jedan zapis u bazi
podataka:")
    self.opis_lejbel.grid(row = 0, column = 0, columnspan = 2, sticky = W)
    self.ime_lejbel = Label(proz1, text = "Ime osobe")
    self.ime_lejbel.grid(row = 1, column = 0, pady = (10, 1))
    self.mesto_lejbel = Label(proz1, text = "Mesto u kojem živi osoba")
    self.mesto_lejbel.grid(row = 2, column = 0, sticky = W)
    self.godina_lejbel = Label(proz1, text = "Godina rođenja osobe")
    self.godina_lejbel.grid(row = 3, column = 0, sticky = W)
    self.box_za_brisanje_lejbel = Label(proz1, text = "Izabrani ID")
    self.box_za_brisanje_lejbel.grid(row = 7, column = 0, pady = 5)

def postavljanje_tekst_boksa_u_prozor1(self, proz1):
    self.ime = Entry(proz1, width = 30)
    self.ime.grid(row = 1, column = 1, padx = 20, pady = (10, 1))
    self.mesto = Entry(proz1, width = 30)
    self.mesto.grid(row = 2, column = 1)
    self.godina = Entry(proz1, width = 30)
    self.godina.grid(row = 3, column = 1)
    self.box_za_brisanje = Entry(proz1, width = 30)
    self.box_za_brisanje.grid(row = 7, column = 1, pady = 5)

def kreiranje_dugmeta_za_unos_podataka_u_bp(self, proz1):
    self.dugme_unos = Button(proz1, text = "Unos zapisa u bazu podataka", command =
self.unos)
    self.dugme_unos.grid(row = 4, column = 0, columnspan = 2, pady = 10, padx = 10, ipadx =
100)

def kreiranje_dugmeta_za_prikaz_zapisa_iz_baze(self, proz1):
    self.dugme_upit_cela_baza = Button(proz1, text = "Prikaz svih zapisa u bazi podataka",
command = self.upit_cela_baza)
    self.dugme_upit_cela_baza.grid(row = 5, column = 0, columnspan = 2, pady = 10, padx =
10, ipadx = 88)

def kreiranje_dugmeta_za_brisanje_zapisa(self, proz1):
    self.dugme_brisanje_zapisa = Button(proz1, text = "Brisanje upisanog broja zapisa",
command = self.brisanje)
    self.dugme_brisanje_zapisa.grid(row = 8, column = 0, columnspan = 2, pady = 10, padx =
10, ipadx = 88)

def kreiranje_dugmeta_za_nadogradnju(self, proz1):
    self.dugme_nadogradnja = Button(proz1, text = "Izmena zapisa", command = self.izmena)
    self.dugme_nadogradnja.grid(row = 9, column = 0, columnspan = 2, pady = 10, padx = 10,
ipadx = 88)

def main():
    osnovni_prozor = Tk()
    osnovni_prozor.title("Tkinter sa klasama")
    osnovni_prozor.iconbitmap("py_slika.ico")
    osnovni_prozor.geometry("400x400")
    bp = baza_podataka(osnovni_prozor)

```

```
osnovni_prozor.mainloop()  
main()
```