



UNIVERZITET UNOVOM SADU
PRIRODNO-MATEMATIČKI
FAKULTET
DEPARTMAN ZA MATEMATIKU
I INFORMATIKU



Vukan Koprivica, 75/19

Kasa restorana

(simulacija kase u restoranu)

- seminarski rad iz predmeta Skript jezici -

Novi Sad, 2020.

Sadržaj

1. Uvod.....	3
2. Opis programa.....	4
Glavni meni.....	4
Radnici	5
Predjelo,Glavno jelo,Dezerti i Pica	6
Meni	8
3. Zaključak	11

1. Uvod

Kasa restorana predstavlja primitivnu simulaciju rada kase u restoranu na računaru napisana u programskom jeziku Python 3.7.0 u okviru seminarskog rada iz predmeta Seminarski rad A- Skript jezici.

Aplikacija je u osnovi simulacija neke kase u ovom slučaju restorana "Panorama" čiji se podaci koriste u aplikaciji. Zamišljeno je da na što lakši način pomogne radnicima da ukucaju račun i da imaju uvid u sve podatke jela i pića iz restorana.

Aplikacija se sastoji iz 5 modula:

1. Radnici: ovaj modul omogućava da se radnik na samom početku uloguje da ne bi baš svako imao uvid u kasu restorana. Takođe na lak način se dodaje novi radnik u fajl.
2. Predjelo: u ovom modulu je omogućen lak pristup svim predjelima u restoranu takođe se na vrlo lak način dodaje novo predjelo ili uzima samo jedno radi otkucavanja.
3. Glavno jelo: ovaj modul omogućava iste stvari kao i predjelo samo je malo obilnije jer je glavna stvar u restoranu.
4. Dezert: omogućava uzimanje dezerta ili uvid u podatke kao u prethodnim modulima
5. Piće: omogućava kao i u svakom normalnom restoranu pića koja su dostupna svim gostima

Svi moduli sadrže tekstualne fajlove u kojima se čuvaju podaci kao što su šifra, ime, kratak opis i cena.

20|cevapi|350g|580

Listing 1. Linija unosa u okviru datoteke glavnogjela.txt

U daljem tekstu biće detaljnije opisani svi moduli i sve mogućnosti programa.

2. Opis programa

Aplikacija se poziva iz datoteke Menu.py u koju se importuju svi ostali moduli i poziva se funkcija `main()`. Nakon toga se poziva funkcija `login` koja omogućava prijavljivanje radnika da bi se ostatak programa sproveo.

```
16 #radnik se prijavi pa se ispise glavni meni
17 def main():
18     print()
19     print( "Kasa restorana")
20     print( "=====")
21     while not login():
22         print( "\nNiste uneli postojece ime i lozinku!")
```

Listing 2. Prijavljivanje radnika sve dok se ne unesu odgovarajući username i password

```
76 def login():
77     username = input("Korisnicko ime >> ")
78     password = input("Lozinka >> ")
79     return Radnici.login(username, password)
```

Listing 3. Funkcija login

Glavni meni

Nakon uspešnog prijavljivanja radnika ispisuje se glavni meni mogućih radnji u okviru programa. Korisnik unosi neku cifru koja označava neku radnju. Nakon izvršene neke radnje glavni meni se stalno vraća sve dok se ne ukuca x koje označava zatvaranje kase.

```
24     komanda = '0'
25     while komanda != 'X':
26         komanda = menu()
27         if komanda == '1':
28             addPredjela()
29         elif komanda == '2':
30             listPredjela()
31         elif komanda == '3':
32             addGlavnaJela()
33         elif komanda == '4':
34             listGlavnaJela()
35         elif komanda == '5':
36             addDezerti()
37         elif komanda == '6':
38             listDezert()
39         elif komanda == '7':
40             addPica()
41         elif komanda == '8':
42             listPica()
43         elif komanda == '9':
44             Racun()
45         elif komanda == '10':
46             Odnos()
47         elif komanda == '11':
48             addRadnika()
49     print( "Kasa je zatvorena.")
```

Listing 4. Korisnik unosi broj od 1 do 11 u zavisnosti koju funkciju želi da pozove

Radnici

Modul u kom se nalaze metode za prijavljivanje, formatiranje fajla i dodavanje novog radnika u listu. Nakon otvaranja `radnici.txt` datoteke učitavaju se podaci i to omogućava metodi `login` da proveri da li su uneti podaci korisnika tačni.

```
16 #proveravanje username i password
17 def login(username, password):
18     for rad in radnici:
19         if rad['username'] == username and rad['password'] == password:
20             return True
21     return False
22
23 def loadRadn():
24     for line in open('radnici.txt', 'r').readlines():
25         if len(line) > 1:
26             rad = str2rad(line)
27             radnici.append(rad)
```

Listing 5. Provera se username i password iz učitane datoteke

Takođe moguće je i dodati novog radnika.

```
29 def saveRadnika():
30     file = open('radnici.txt', 'w')
31     for rad in radnici:
32         file.write(rad2str(rad))
33         file.write('\n')
34     file.close()
35
36 #dodaje radnika u listu
37 def addRadnika(rad):
38     radnici.append(rad)
```

Listing 6. Upisivanje novih radnika

Predjelo, Glavno jelo, Dezerti i Pica

U okviru predjela se otvara fajl `predjelo.txt` iz kojeg se izvlace podaci.

```
1 01|govedja supa|govedja supa,rezanca|160
2 02|teleca corba| |230
3 03|pohovana paprika sa sirom|200g|300
4 04|pohovan kackavalj|200g|360
5 05|balkanski tanjir za dve osobe|njeguska,kulen,pecenica,sir,ajvar|800
6 06|njeguska prsut|100g|520
7 07|govedja prsut|100g|490
8 08|svinjska pecenica|100g|390
9 09|kulen|100g|250
10 10|sjenicki sir|150g|250
11 11|kajmak|100g|290
12 12|feta sir|100g|190
13 13|gorgonzola|100g|390
```

Listing 7. Datoteka predjelo.txt

Takođe moguće je dodati novo predjelo u imenik.

```
15 #upisuje novo predjelo u fajl
16
17 def savePredjelo():
18     file = open('predjelo.txt', 'w')
19     for pred in predjela:
20         file.write(predjelo2str(pred))
21         file.write('\n')
22     file.close()
23
24
25 #dodaje predjelo u listu
26
27 def addPredjelo(pred):
28     predjela.append(pred)
29
```

Listing 8. Dodavanje novog predjela

Datoteka predjelo se formatira na određene veličine iznad kojih se postavi header radi urednosti i lakše preglednosti fajla.

```

54 def formatHeader():
55     return \
56         "sifra|Ime                |opis
57
58 #formatira predjelo
59 def formatPredjelo(pred):
60     return u"{0:5}|{1:29}|{2:39}|{3:4}".format(
61         pred['sifra'],
62         pred['ime'],
63         pred['opis'],
64         pred['cena'])
65
66 #za svako predjelo ispisuje po jedan red ispod headera
67
68 def formatPredjela(predList):
69     result = ""
70     for pred in predList:
71         result += formatPredjelo(pred) + '\n'
72     return result
73

```

Listing 9.Formatira se header i za svako predjelo se ispisuje jedan red ispod
Takodje je moguće formatirati sve odjednom.

```

74 #formatira sva predjela
75 def formatAllPredjela():
76     result = ''
77     for pred in predjela:
78         result += u"{0:5}|{1:29}|{2:39}|{3:4}".format(
79             pred['sifra'],
80             pred['ime'],
81             pred['opis'],
82             pred['cena']) + '\n'
83
84     return result

```

Listing 10. Formatiranje svih predjela

Da bi se odgovarajuće jelo pronašlo koristi se metod **findPredjelo(sifra)**: koji pronalazi odgovarajuće predjelo na osnovu unete šifre korisnika i kao povratnu informaciju vraća cenu istog. Primenjuje se kod same primene računa jer se cena istog sabira sa ostalim cenama i dobija se konačni račun. Takođe u okviru metode **findPredjela()**: moguće je uzeti u obzir sva predjela radi statistike najskupljih predjela

```

85 #pronalaazi odredjeno predjelo po sifri i vraca njegovu cenu
86 def findPredjelo(sifra):
87     for s in predjela:
88         if s['sifra'] == sifra:
89             print("Izabrali ste :"+s['ime'])
90             return s['cena']
91
92 #pronalaazi sva predjela
93 def findPredjela():
94     rezultat=[]
95     for s in predjela:
96         rezultat.append(s)
97     return rezultat

```

Listing 11. Metode za pronalaženje svih ili određenog predjela

Metode GlavnoJelo,Dezerti i Pica koriste skoro identične metode kao Predjelo i imaju istu primenu.

Meni

Nakon što korisnik unese broj od 1 do 11 neka funkcija u glavnom meniju se izvrši. Naprimer ako stisne 1 funkcija **addPredjela()** :će se pozvati na metod iz modula predjelo i korisniku će biti ponudjeno da unosi podatke o tom predjelu redom kako bi se oni formatirali sa ostalim podacima u fajlu `predjela.txt`. Takođe postoje iste funkcije u okviru ostalih modula i jela klikom na 3,5 ili 7.

```
81 def addRadnika():
82     print( " Upis novog radnika\n")
83     rad = {}
84     rad['ime'] = input("Unesite ime >> ")
85     rad['prezime'] = input("Unesite prezime >> ")
86     rad['username'] = input("Unesite username >> ")
87     rad['password'] = input("Unesite password >> ")
88     Radnici.addRadnika(rad)
89     Radnici.saveRadnika()
```

Listing 12 Funkcija koja na osnovu metoda iz radnika upisuje novog radnika

Ukoliko korisnik želi da pregleda liste jela može to uciniti unošenjem 2,4,6 ili 8 na tastaturi i dobiće brz i lak pristup svim podacima o jelima i piću. Funkcija **listPredjela()** poziva metode za brz ispis podataka.

```
101 def listPredjela():
102     print(Predjelo.formatHeader())
103     print(Predjelo.formatAllPredjela())
```

Listing 13. Funkcija `listPredjela()` koja ispisuje listu sa podacima

sifra	Ime	opis	cena
01	govedja supa	govedja supa, rezanca	160
02	teleca corba		230
03	pohovana paprika sa sirom	200g	300
04	pohovan kackavalj	200g	360
05	balkanski tanjir za dve osobe	njeguska, kulen, pecenica, sir, ajvar	800
06	njeguska prsut	100g	520
07	govedja prsut	100g	490
08	svinjska pecenica	100g	390
09	kulen	100g	250
10	sjenicki sir	150g	250
11	kajmak	100g	290
12	feta sir	100g	190
13	gorgonzola	100g	390

Listing 14. Ispis liste

Glavna funkcija ove aplikacije jeste sam račun. Kada korisnik unese 9 izbacice mu mali pod meni gde opet može da bira koju vrstu jela želi ili mozda piće. Nakon toga unosi odgovarajuću šifru specifičnog jela koje želi da izabere i kao povratnu informaciju dobija njegovu cenu koja automatski ulazi u ukupnu sumu računa. Nakon toga korisnik se opet vraća na odabir jela ili pića sve dok ne izabere sve što mu je potrebno. Nakon toga dobija ispis ukupne vrednosti računa nakon koje treba da unese broj novca koji je primio od gosta i nakon toga mu računar izbacuje kusur. Ukoliko nije uneto dovoljno novca vratiće se opet na unos sve dok se ne unese tačan iznos ili više od toga.

```

147 def menuracun():
148     printMenuracun()
149     command = input(">> ")
150     while command.upper() not in ('1', '2', '3', '4', 'X'):
151         print( "\nUneli ste pogresnu komandu.\n")
152         printMenuracun()
153         command = input(">> ")
154     return command.upper()
155
156 def printMenuracun():
157     print( "\nIzaberite opciju:")
158     print( "  1 - izaberi predjelo")
159     print( "  2 - izaberi glavno jelo")
160     print( "  3 - izaberi dezert")
161     print( "  4 - izaberi pice")
162     print( "  x - naplati ")

```

Listing 15. Ispis mini menija u okviru računa

```

def Racun():
    suma = 0
    komandar = '0'
    while komandar != 'X':
        komandar = menuracun()
        if komandar == '1':
            suma=suma + int(findPredjelo())
            print("Vas trenutni racun je:",suma)
        elif komandar == '2':
            suma=suma + int(findGlavnoJelo())
            print("Vas trenutni racun je:",suma)
        elif komandar == '3':
            suma=suma + int(findDezert())
            print("Vas trenutni racun je:",suma)
        elif komandar == '4':
            suma=suma + int(findPice())
            print("Vas trenutni racun je:",suma)
        print("Vas racun je:", suma)
        placeno = 0
        placeno += int(input("Unesite sumu: "))
        while suma>int(placeno):
            print("Niste uneli dovoljno novca,unesite jos!")
            placeno += int(input("Unesite sumu: "))
        print("Vas kusur je ",int(placeno)-suma)

```

Listing 16. Funkcija račun koja je objašnjena u tekstu iznad

Za kraj se ukoliko korisnik unese 10 biće mu ponuđen jos jedan mini meni u okviru kog bira predjela, glavna jela, dezerte ili pića i dobija odnos cena svih jela i pića tog odabira u stubičastom grafikonu.

```

227 def printMenuodnos():
228     print( "\nIzaberite odnos cena:")
229     print( "  1 - izaberite predjelo")
230     print( "  2 - izaberite glavno jelo")
231     print( "  3 - izaberite desert")
232     print( "  4 - izaberite pice")
233     print( "  X - vrati se u glavni meni ")
234
235 def menuodnos():
236     printMenuodnos()
237     command = input(">> ")
238     while command.upper() not in ('1', '2', '3', '4', 'X'):
239         print( "\nUneli ste pogresnu komandu.\n")
240         printMenuodnos()
241         command = input(">> ")
242     return command.upper()

```

Listing 17. Ispis malog menija u okviru odnosa

```

244 def odnosCenaP():
245     predjela = Predjelo.findPredjela()
246     predjela.sort(key = lambda x: x['cena'])
247     cena = []
248     ime = []
249     for i in predjela:
250         cena.append(int(i['cena']))
251         ime.append(i['ime'])
252     plt.xlabel('ime')
253     plt.ylabel('cena')
254     plt.ylim(ymin=150, ymax=900)
255     plt.xticks(rotation=90)
256     plt.bar(ime, cena)
257     plt.show()

```

Listing 18. Primer koda za ispis odnosa cena svih pića

Nakon svake obavljene funkcije aplikacija se vraća na prvobitni meni sve dok na kraju korisnik ne unese x i aplikacija kasa se zatvori.

```

>> x
Kasa je zatvorena.

```

Listing 19. Zatvaranje aplikacije

3. Zaključak

U okviru ove aplikacije urađena je simulacija kase u restoranu "Panorama" gde je omogućen pristup podacima kao i unos racuna gostiju i uvid u odnos jela i cena. Program nije usavršen moglo bi se još dosta na njemu raditi i dodavati raznih funkcija kao i čuvanje drugih podataka pomoću kojih bi se imao uvid u potrošnji, gužvi i mesečnih prihoda samog restorana. Može se primenjivati u raznim restoranima uz izmenu podataka.