Sveučilište Jurja Dobrile

Fakultet informatike u Puli

**Matej Višnjić**

**Benzinska Postaja – MySQL**

Seminarski rad

Sveučilište Jurja Dobrile

Fakultet informatike u Puli

**Matej Višnjić**

**Benzinska Postaja – MySQL**

Seminarski rad

**JMBAG: 0303092423, redoviti student**

**Smjer: Informatika**

**Kolegij: Baze podataka 2**

**Mentor: doc. dr. sc. Goran Oreški**

Sadržaj

[1. UVOD 1](#_Toc78458340)

[2. IZRADA TABLICA 2](#_Toc78458341)

[2.1 Jednostavni EER diagram 2](#_Toc78458342)

[2.2 Opis tablica 3](#_Toc78458343)

[2.3 Prikaz podataka 7](#_Toc78458344)

[2.3.1 Prikaz tablice benzinska 7](#_Toc78458345)

[2.3.2 Prikaz tablice voditelj 8](#_Toc78458346)

[2.3.3 Prikaz tablice blagajnik 8](#_Toc78458347)

[2.3.4 Prikaz tablice artikl 8](#_Toc78458348)

[2.3.5 Prikaz tablice gorivo 9](#_Toc78458349)

[2.3.6 Prikaz tablice pumpa 9](#_Toc78458350)

[2.3.7 Prikaz tablice kupnja i prodani\_proizvodi 10](#_Toc78458351)

[2.3.8 Prikaz tablice račun 10](#_Toc78458352)

[3. POGLEDI 11](#_Toc78458353)

[3.1 Pogled proizvodi 11](#_Toc78458354)

[3.2 Pogled zaposlenici 11](#_Toc78458355)

[3.3 Pogledi artikala i goriva 12](#_Toc78458356)

[4. OKIDAČI, FUNKCIJE I PROCEDURE 13](#_Toc78458357)

[4.1 Okidači 13](#_Toc78458358)

[4.2 Funkcije 13](#_Toc78458359)

[4.2.1 Funkcija za ispis ukupnog broja proizvoda 14](#_Toc78458360)

[4.2.2 Funkcija za prosječnu cijenu određene vrste goriva 14](#_Toc78458361)

[4.2.3 Funkcija za količinu prodanog goriva 15](#_Toc78458362)

[4.2.4 Funkcija za količinu prodanog artikla 15](#_Toc78458363)

[4.2.5 Funkcija za ukupan iznos računa 17](#_Toc78458364)

[4.2.6 Funkcija za vraćanje ukupno istipkanih računa 17](#_Toc78458365)

[4.3 Procedure 18](#_Toc78458366)

[4.3.1 Procedura za uvećati/umanjiti određenu cijenu goriva i artikala 18](#_Toc78458367)

[4.3.2 Procedura za obaviti kupnju 19](#_Toc78458368)

[4.3.3 Procedura za pregled stavki na trenutnoj kupnji 20](#_Toc78458369)

[4.3.4 Procedura za brisanje stavki trenutne kupnje 20](#_Toc78458370)

[4.3.5 Procedura za dodavanje/micanje količine stavki 21](#_Toc78458371)

[4.3.6 Procedura za ispis računa 22](#_Toc78458372)

[4.3.7 Procedure za pregled računa 22](#_Toc78458373)

[4.3.8 Procedure za dodavanje ili brisanje blagajnika 24](#_Toc78458374)

[4.3.9 Procedure za dodavanje ili brisanje artikla i goriva 25](#_Toc78458375)

[4.3.10. Procedura za brisanje proizvoda 26](#_Toc78458376)

[5. KORISNICI 27](#_Toc78458377)

[5.1 Voditelj 27](#_Toc78458378)

[5.2 Blagajnik 27](#_Toc78458379)

[6. ZAKLJUČAK 28](#_Toc78458380)

[7. LITERATURA 29](#_Toc78458381)

# 1. UVOD

Baza podataka je organizirana strukturiranih podataka ili podataka koji se pohranjuju u računalnom sustavu. Mogu pohraniti podatke o osobama, proizvodima, narudžbama i mnogo drugih stvari.

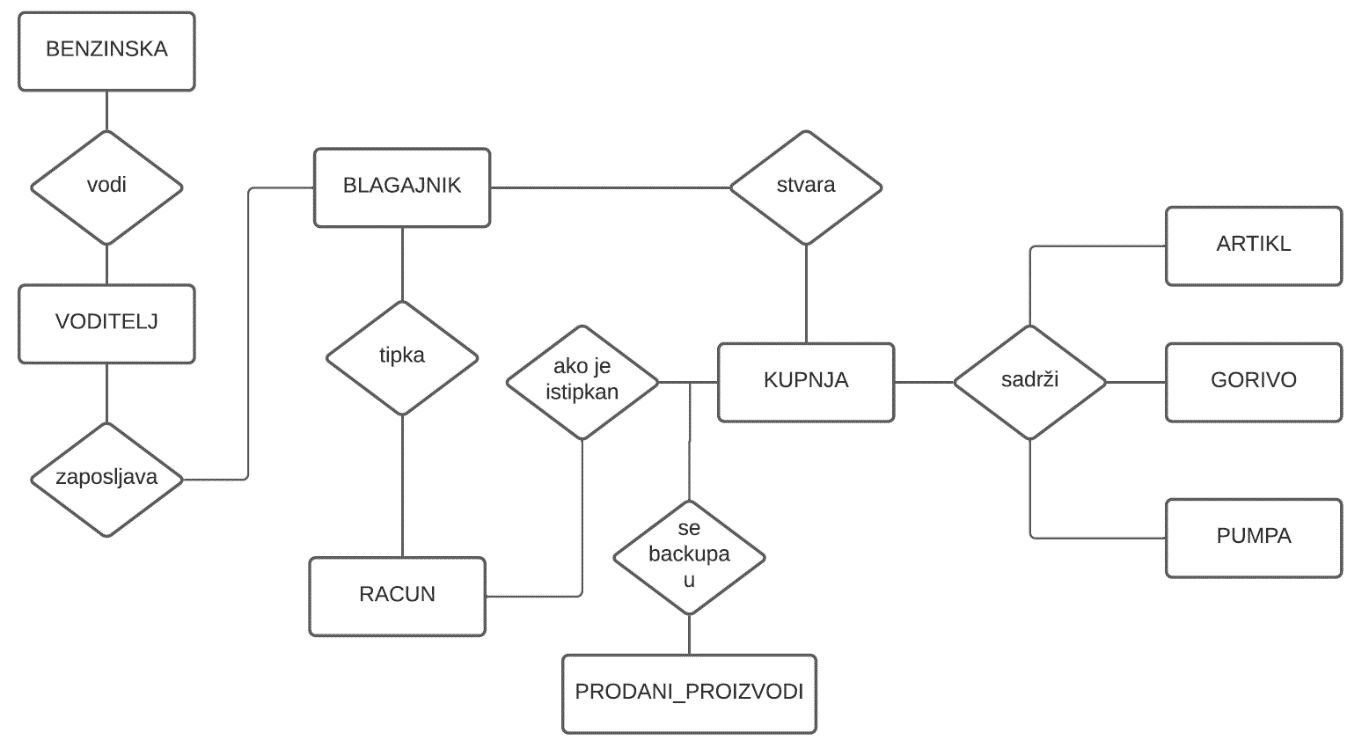
Podaci unutar baza podataka najčešće se modeliraju u retke i stupce u nizu tablica kako bi obrada i upit podataka bili učikoviti. Postoje razno razni programski jezici za komuniciranje sa bazama podataka, kao na primjer SQL[[1]](#footnote-1), MySQL[[2]](#footnote-2), postgreSQL[[3]](#footnote-3) i mnogi drugi. Svaka se razlikuje od drugih, MySQL je open-source sustav relacijske baze podataka.

Tema projekta je baza podataka izmišljene „Benzinske postaje“. Projekt „Benzinska Postaja“ pisan je koristeći MySQL programski jezik i alat MySQL Workbench[[4]](#footnote-4). Projekt je zamišljen tako da prati benzinske postaje, voditelje, blagajnike, artikle, goriva, i pumpu. Isto tako sadržava opcije za kupnju artikala ili goriva, pratnju prodanih proizvoda i svaki istipkani račun.

# 2. IZRADA TABLICA

Baza podataka sadržava 9 tablica; benzinska, voditelj, blagajnik, artikl, gorivo, pumpa, kupnja, prodani\_proizvodi, racun.

## 2.1 Jednostavni EER diagram



Slika 1. jednostavni EER diagram

Na slici 1. prikazan je jednostavan EER diagram bez atributa, koji prikazuje imena tablica i veze među njima. Koncept je osmišljen tako da svaka benzinska ima svoga voditelja, a voditelj zapošljava blagajnika. Blagajnik za svaku kupnju odabere stvari iz tablica artikla i goriva, te odabere određeni broj pumpe(pumpa broj 0 služi ako se kupuju stvari bez da je gorivo natankano). Nakon što je račun istipkan, stvari koji su bili u košarici se brišu iz košarice i dodavaju u prodani\_proizvodi. Prodani proizvodi služe za analizu koliko je stvari prodano.

## 2.2 Opis tablica

Graphical user interface, text, application

Description automatically generated

Slika 2. tablica benzinska

Tablica 'Benzinska' sastoji se od 6 atributa; ID, ime, broj\_telefona, adresa, grad, poštanski\_broj. Služi za osnovne informacije kao što su adresa, broj telefona i slično. id je tipa integer[[5]](#footnote-5), not null[[6]](#footnote-6) i primarni ključ tablice, ime varchar[[7]](#footnote-7) od 25 znakova, broj\_telefona varchar i not null. Atribut adresa je tip varchar od 35 znakova, a atribut grad je tipa char[[8]](#footnote-8) od 25 znakova, poštanski broj je tipa integer i ne smije biti null. Svaka benzinska ima svog voditelja. (o tablici 'voditelj' prikazano na slici 3.)

Slika 3. tablica voditelj

Graphical user interface, text, application

Description automatically generated

Tablica 'voditelj' sastoji se od 7 atributa; ID, sifra\_v, OIB, ime, prezime, datum\_zaposlenja, placa. ID je tipa integer,auto\_increment[[9]](#footnote-9), not null, unique[[10]](#footnote-10), i primarni ključ tablice. Sifra je tipa varchar od maksimalno 11 znakova, ne smije biti null i unikatna je. OIB tip varchar od maksimalnih 11 znakova, not null i unique. Ime i prezime su tipa char od maksimalnih 30 znakova. Datum zaposlenja je tip datetime[[11]](#footnote-11), a zadani datum je NOW()[[12]](#footnote-12). Sadrži još CHECK[[13]](#footnote-13) ograničenje koje ograničava da plaća nije manja od 6000 kuna, ako prekršimo ograničenje javit će se greška CHK\_placa\_voditelj.

Graphical user interface, text, application

Description automatically generated

Slika 4. tablica blagajnik

Tablica 'blagajnik' je identična kao i tablica 'voditelj' samo CHECK ograničenje nije postavljeno da je plaća manja od 6000 kuna, nego da nije manja od 4450 kuna.

A picture containing timeline

Description automatically generated

Slika 5. tablica artikl

Tablica 'artikl' sastoji se od 4 atributa; ID koji je tip integer, not null, unikatan i primarni ključ tablice, ima svoju vrstu artikla koja je tip varchar koji prima 20 znakova, naziv varchar koji prima 50 znakova, i cijena koja je numeric[[14]](#footnote-14) tip podataka koji sadrži 10 mjesta, i zakružuje na 2 decimale. Tablica sadrži i dva CHECK ograničenja, koji ograničavaju da cijena ne može biti manja od nule, a drugi da se ne može unijeti vrsta proizvoda, osim onih koje su navedene u CHECK ograničenju.

A picture containing timeline

Description automatically generated

Slika 6. tablica gorivo

Tablica 'gorivo' je identično napravljena kao i tablica 'artikl', kasnije su tablica 'artikl' i 'gorivo' povezani sa view[[15]](#footnote-15)(pogledi) što će te više vidjeti u poglavlju [3.1 Pogled proizvodi](#_3.1_Pogled_proizvodi)

Graphical user interface, text

Description automatically generated

Slika 7. tablica pumpa

Tablica 'pumpa' sastoji se samo od jednog atributa ID koji je tip integer, not null, unikatan i primarni ključ tablice, ima jedno ograničenje koji provjerava da id pumpe nesmije biti manji ili jednak nuli.

Graphical user interface, text, application

Description automatically generated

Slika 8. tablica kupnja

Tablica 'kupnja' sadrži 4 atributa; ID, id\_pumpa, id\_prizvod, kolicina. ID je tip integer, not null, auto\_increment i ujedno i primarni ključ tablice. Id\_pumpa je tip integer, te isto tako strani ključ[[16]](#footnote-16) koji je povezan sa tablicom pumpa. Id\_proizvod je tip integer. Kolicina je tip numeric koji ima 6 brojeva i 2 decimale. Također na slici vidimo i dva ograničenja, prvo ograničenje provjerava da količina nije manja od nule, a drugo ograničenje da broj pumpe nije manji ili jedan nuli.

Slika 9. tablica prodani\_proizvodi

Graphical user interface, text, application

Description automatically generated

Tablica 'prodani\_proizvodi' identični su kao i tablica 'kupnja' a služi samo za praćenje koliko je stvari prodano, kasnije pomoću procedura biti će omogućeno obaviti kupnju, a tablica 'kupnja' nakon svakog otipkanog računa biti će izbrisana, a proizvodi koji su istipkani prebačeni u prodani\_proizvodi.

Graphical user interface, text, application, chat or text message

Description automatically generated

Slika 10. tablica racun

Tablica 'racun' sadrži 5 atributa; ID tipa integer, not null, unikatan, i auto\_increment, id\_blagajnik tipa integer, datum\_izdavanja tipa datetime sa funkcijom NOW(), osnovica koja je numeric ima šest brojeva i dve decimale, PDV i ukupno koji su istog tipa kao i osnovica. Imamo u tablici strani ključ id\_blagajnik koji je povezan sa tablicom 'blagajnik', te ako obrišemo blagajnika, postaviti će se null vrijednost u istipkanim računima(biti će prikazano kasnije). Također, sadrži tri check ograničenja koji 'paze' da osnovica, PDV i ukupno nije manje ili jednako nula.

## 2.3 Prikaz podataka

U ovome poglavnju prikazati ćemo neke od podataka koje sadržavaju tablice, i opisati ukratko za što služe.

## 2.3.1 Prikaz tablice benzinska

Graphical user interface, application, Teams

Description automatically generated

Slika 11. prikaz podataka benzinska

Na slici 11. prikazan je podatak koji sadrži tablica 'benzinska'. Ova tablica sadrži samo jedan zapis i prikazuje nam informacije o benzinskoj postaji koja se trenutno koristi.

## 2.3.2 Prikaz tablice voditelj

Graphical user interface, application, Teams

Description automatically generated

Slika 12. prikaz podataka voditelj

Svaka benzinska postaja ima svog voditelja, koji zapošljava ostale blagajnike. Voditelj ima svoj šifru kojom se prijavljuje u sustav, i za razliku od blagajnika može vidjeti više informacija, kao što su ukupni broj proizvoda koji su prodani, najprodavanije artikle i još mnoge druge stvari.

## 2.3.3 Prikaz tablice blagajnik

Graphical user interface, application, table

Description automatically generated

Slika 13. prikaz podataka blagajnik

Trenutno je zaposleno tri blagajnika, svaki se razlikuje po svom ID i šifri za se prijaviti u sustav. Kao blagajnik, nema previše mogućnosti, osim da može dodavati stavke za kupovinu, ispisivati račun i slično. Ostale stvari kao što su analiza prodaje, podizanje/spuštanje cijene goriva je blagajnicima onemogućena.

## 2.3.4 Prikaz tablice artikl

Table

Description automatically generated

Slika 14. prikaz podataka artikala

Svaki artikl ima svoj jedinstveni ID, vrstu artikla, naziv i cijenu. U tablici 'artikl' prikazani su svi artikli. Također, svi artikli su prikazani u pogledima(*en.view*).

Imamo 7 različitih vrsta artikala, koji su razvrstani u pogledima. Tako možemo lakše pronaći proizvod koji želimo staviti u kupnju. *\*Na slici prikazano je prvih 20 artikala, sveukupno ih je 51\*.*

## 2.3.5 Prikaz tablice gorivo

Table

Description automatically generated

Slika 15. prikaz podataka gorivo

Kao što je prikazano na slici, tablica „gorivo“ je identična kao i tablica „artikl“. Ovdje su prikazana sva goriva koja nudi benzinska postaja. Artikli i gorivo su povezani u pogled „proizvodi“. Pogled „proizvodi“ prikazuje sve artikle i goriva skupa. Također, imamo procedure koje mogu promjeniti cijene goriva, koje radi voditelj poslovnice, to će biti prikazano u poglavlju [4.3.1 Procedura za uvećati/umanjiti određenu cijenu goriva i artikala](#_4.3.1_Procedura_za).

## 2.3.6 Prikaz tablice pumpa

Graphical user interface, application

Description automatically generated

Slika 16. prikaz podataka pumpa

Benzinska postaja ima 6 pumpi goriva. Broj nula nam služi kada netko nije kupio gorivo, nego samo neke od artikala.

## 2.3.7 Prikaz tablice kupnja i prodani\_proizvodi

Tablice „kupnja“ služi za unos artikala ili goriva koje blagajnik unosi kad neki kupac dođe na benzinsku. Kada ispisujemo račun tablica kupnja se briše, to jest truncate[[17]](#footnote-17). Više o tome prikazano je u poglavlju [4.3.6 Procedura za ispis računa](#_4.3.6_Procedura_za). Tablica „Prodani proizvodi“ služi za pratiti koliko je artikala ili goriva prodano. Točnije, kada ispisujemo račun i tablica kupnja se obriše, stvari koji su bili u tablici kupnja, kopiraju se u tablicu „prodani proizvodi“.

## 2.3.8 Prikaz tablice račun

Slika 17. prikaz podataka račun

Text

Description automatically generated

Tablica „racun“ ima svoj ID koji predstavlja broj računa, ima ID blagajnika koji je istipkao račun, datum izdavanja, osnovicu, PDV i ukupni iznos računa. Također, postoji procedura koja prikazuje ime i prezime blagajnika koji je kucao račun, više o tome prikazano je u poglavlju [4.3.7 Procedure za pregled računa](#_4.3.7_Procedure_za).

# 3. POGLEDI

Korištenje pogleda olakšavamo da spojimo dve ili više tablica u jednu virtualnu. U pogledima ne možemo dodavati nove stavke, niti brisati. Ja sam napravio 13 pogleda. Svi pogledi su napravljeni na jednostavan način i nisu prekompleksni.

## 3.1 Pogled proizvodi

Proizvodi su pogled gdje su spojene dvije tablice „artikl“ i „gorivo“. Sintaksa nije kompleksna, korišten je select[[18]](#footnote-18) upit i union[[19]](#footnote-19).

Text

Description automatically generated with low confidence

Slika 18. prikaz koda za pogled proizvodi

## 3.2 Pogled zaposlenici

Text

Description automatically generated

Slika 19. prikaz koda za pogled zaposlenici

Pogled zaposlenici napravljen je slično kao i pogled „proizvodi“ samo nije select upitom odabrano sve, nego samo stupci koje smo naveli. Sadrži uniju kao i prethodni pogled. Slika 20. prikazuje podatke iz pogleda.

Graphical user interface, application, table

Description automatically generated

Slika 20. prikaz pogleda zaposlenici

## 3.3 Pogledi artikala i goriva

Text

Description automatically generated with low confidenceA picture containing Word

Description automatically generated

Slika 21. kod pogleda dizel i kava

Pogledi za artikle i gorivo poredani su prema vrsti artikla/goriva. Ostali pogledi za određenu vrstu artikala i goriva napravljen su na identičan način. Na slici 22. prikazani su upiti za ova dva pogleda.

Graphical user interface, text, application

Description automatically generatedGraphical user interface, text

Description automatically generated

Slika 22. prikaz pogleda „dizel“ i „kava“

# 4. OKIDAČI, FUNKCIJE I PROCEDURE

U ovome poglavlju proći ćemo kroz sve funkcije, procedure i okidače[[20]](#footnote-20)(*en.trigger*) koje su stvorene u bazi podataka za benzinsku postaju.

## 4.1 Okidači

Imamo samo dva okidača koji služe da ako unosimo novi artikl ili gorivo, da automatski prebacimo vrstu artikla/goriva u velika slova u slučaju da osoba koja je unosila nije pisala velikim slovima. Na slici 23. i 24. prikazana su ta dva okidača.

Graphical user interface, text, application

Description automatically generatedGraphical user interface, text, application

Description automatically generated

Slika 24. okidač za gorivo

Slika 23. okidač za artikle

## 4.2 Funkcije

Funkcije su skup SQL izraza koji izvršavaju neki zadatak ili operaciju i vraćaju jednu vrijednost. Funkcijski parametar prima samo ulazni parametar za razliku od procedura. U našoj bazi podataka ukupno imamo 6 različitih funkcija.

## 4.2.1 Funkcija za ispis ukupnog broja proizvoda

Graphical user interface, text, application

Description automatically generated

Slika 25. kod funkcije za ukupan broj proizvoda

Funkcija vraća ukupan broj proizvoda(artikli+gorivo), što iznosi 61.

## 4.2.2 Funkcija za prosječnu cijenu određene vrste goriva

Graphical user interface, text, application, email

Description automatically generated

Slika 26. kod funkcije za prosječnu cijenu goriva

Funkcija za izračun prosječne cijene goriva osmišljena je tako da kao parametar funkcije upisujemo vrstu goriva i onda funkcija vraća prosječnu cijenu određenog goriva.

## 4.2.3 Funkcija za količinu prodanog goriva

Graphical user interface, text, application

Description automatically generated

Slika 27. kod funkcije za količinu prodanog goriva

Funkcija vraća sveukupnu količinu prodanog goriva, ako gorivo nikad nije prodanog funkcija vraća null vrijednost. Kao parametar koristimo ID od određenog goriva. Primjer kada imamo neku prodanu količinu možete pogledati na slici 28.



Slika 27. primjer funkcije za količinu prodanog goriva

## 4.2.4 Funkcija za količinu prodanog artikla

Funkcija za vraćanje količine određenog prodanog artikla napravljena je na sličan način kao i funkcija za količinu prodanog goriva. Ako niti jedan artikl nije prodan vraća vrijednost null, u suprotnom vraća broj prodanih artikala. Kao parametar isto koristimo ID određenog artikla. Kod funkcije možete pogledati na slici 28., a primjer funkcije na slici 29.

A picture containing graphical user interface

Description automatically generated

Slika 28. kod funkcije za količinu prodanog artikla



Slika 29. primjer funkcije za količinu prodanog artikla

## 4.2.5 Funkcija za ukupan iznos računa

Funkcija za ukupan iznos računa stvorena je isključivo za kasnije korištenje u proceduri za ispis račun, što možete vidjeti na [4.3.6 Procedura za ispis računa](#_4.3.6_Procedura_za). Funkcija množi cijenu i količinu sa trenutne stavke računa, te vraća iznos računa. Kod možete pogledati na slici 30.

Graphical user interface, text, application

Description automatically generated

Slika 30. kod funkcije za ukupan iznos računa

## 4.2.6 Funkcija za vraćanje ukupno istipkanih računa

Posljednja funkcija napravljena je da znamo podatak koliko je sveukupno računa istipkano od početka poslovanja. Funkcija ne prima niti jedan parametar, nije kompleksna ima jedan jednostavni select upit koji sa funkcijom count[[21]](#footnote-21) zbraja sve račune te vraća broj svih tih računa. Kod možete vidjeti na slici 31.

Graphical user interface, text, application

Description automatically generated

Slika 31. kod funkcije za ukupan broj izdanih računa

## 4.3 Procedure

Procedure ili pohranjena procedura je podprogram koji je pohranjen u bazi podataka. Procedura ima svoje ime, parametre koji mogu biti ulazni, izlazni i ulazno-izlazni i SQL naredbe. Poziv procedure ne vrši se sa select naredbom kao kod funkcija, nego sa call naredbom. U našoj bazi imamo sveukupno 14 procedura. Sve procedure većinom su rađene u transakcijama[[22]](#footnote-22).

## 4.3.1 Procedura za uvećati/umanjiti određenu cijenu goriva i artikala

Procedura je zamišljena tako da prima parametar uvecaj/umanji, vrstu goriva, i iznos za koji uvecajemo/umanjujemo cijenu. Ako krivo napišemo neke od parametara, procedura nam vraća grešku i cijena ne bude promjenjena sve dok se točno ne upisuju svi parametri.

Slika 32. kod procedure za uvećati/umanjiti cijenu goriva

A picture containing graphical user interface

Description automatically generated

Za cijenu artikala napravljena je još jedna potpuno identična procedura, samo se razlikuje u imenu procedure „cijena\_artikala“, i u upitima ne koristimo tablicu „gorivo“ nego tablicu „artikl“. Procedura javlja grešku ako unesemo nepoznatu vrstu goriva ili cijenu manju od nule ili ako koristimo krivu ključnu riječ. Primjer korištenja procedure artikla možete vidjeti na slici 33.



Slika 33. primjer procedure za uvećati/umanjiti cijenu artikala

## 4.3.2 Procedura za obaviti kupnju

A picture containing timeline

Description automatically generated

Slika 34. kod procedure za kupnju

Procedura za obaviti kupnju osmišljena je na način da prima 3 parametra; ID pumpe, ID proizvoda, količinu proizvoda. Ako neki od parametra nije točan, na primjer krivi ID proizvoda, procedura javlja grešku i kupnja neće biti uspješna, u suprotnom dodavanje stavki u kupnju je uspješna. Ako želimo dodati neki proizvod na drugu pumpu javiti će nam grešku da, ili obrišemo cijelu narudžbu ili da prvo isprintamo račun. Također, postoje opcije za micanje ili dodavanje količina proizvoda s trenutne kupnje ili za brisanje proizvoda s trenutne kupnje.

## 4.3.3 Procedura za pregled stavki na trenutnoj kupnji

A screenshot of a computer

Description automatically generated with medium confidence

Slika 35. kod procedure za pregled stavki trenutne kupnje

Procedura za pregled stavki jednostavna je i prima jedan parametar, ID pumpe i prikazuje nam artikle koje ćemo kupiti. Prikazuje 6 stupaca; ID pumpe, šifru artikla, naziv, količinu, cijenu, iznos. Primjer možete vidjeti na slici 36.



Slika 36. primjer procedure za pregled stavki trenutne kupnje

## 4.3.4 Procedura za brisanje stavki trenutne kupnje

Graphical user interface, text

Description automatically generated

Slika 36. kod procedure za brisanje stavki trenutne kupnje

Procedura briše stavke iz košarice. Potrebno je kao parametar unjeti ID proizvoda(sifra\_artikla) i ako se taj ID nalazi u košarici obrisati će se određeni artikl pod tom šifrom(IDem). U suprotnom vratit će „Ne postoji taj ID proizvoda“. Primjer možete vidjeti na slici broj 37.



Slika 37. primjer procedure za brisanje stavki

## 4.3.5 Procedura za dodavanje/micanje količine stavki

A picture containing text

Description automatically generated

Slika 38. kod procedure za dodavanje/micanje količine

Procedura prima 3 parametra; ID proizvod, količina, parametar p koji može biti samo „dodaj“ ili „oduzmi“ u suprotnom procedura javlja grešku. Također, javlja grešku i ako upišemo krivi ID proizvoda. Kada upišemo točan ID proizvoda koji se nalazi u košarici onda će procedura se izvršiti uspješno. Prijer uspješno uvećane količine možete pogledati na slici broj 39.



Slika 39. primjer procedure za dodavanje količine

## 4.3.6 Procedura za ispis računa

A picture containing graphical user interface

Description automatically generated

Slika 40. kod procedure za ispis računa

Procedura za ispisivanje računa prima dva parametra; ID blagajnika i ID pumpe. Ako je neki parametar krivi procedura vraća grešku, isto tako ako u košarici nema stavki, vratiti će grešku. Nakon što je račun uspješno istipkan, podaci iz košarice prebacuju se u tablicu „prodani\_proizvodi“, a tablica „kupnja“ se truncate(brisanje svih podataka iz tablice). Kada su točni svi parametri procedura vraća da je račun uspješno istipkan.

## 4.3.7 Procedure za pregled računa

Postoji dvije slične procedure za pregled svih istipkanih računa, i za pregled svih istipkanih računa određenog blagajnika. U proceduri za pregled svih računa iskorišten je select upit koji je left joinom[[23]](#footnote-23) povezan sa tablicom blagajnik, procedura ne prima niti jedan parametar.

Procedura određenih računa blagajnika napravljena je isto sa select upitom ali je korišten inner join[[24]](#footnote-24), i where[[25]](#footnote-25) gdje je dan uvjet da je parametar jednak ID-u blagajnika.

A screenshot of a computer

Description automatically generated with medium confidence

Slika 41. kod procedure za pregled svih računa

A screenshot of a computer

Description automatically generated with medium confidence

Slika 42. kod procedure za pregled računa određenog blagajnika

Graphical user interface, text, application

Description automatically generated

Slika 43. primjer procedure za pregled svih računa



Slika 44. primjer procedure za pregled računa određenog blagajnika

## 4.3.8 Procedure za dodavanje ili brisanje blagajnika

Graphical user interface, text

Description automatically generated

Slika 45. kod procedure za dodavanje blagajnika

Procedura za dodavanje blagajnika prima 5 parametara; sifra, OIB, ime, prezime i placu. Procedura je napravljena jednostavno, samo insert u transakciji. Zamišljena je tako da samo voditelj ju smije koristiti za dodavanje novih blagajnika.

Graphical user interface

Description automatically generated with medium confidence

Slika 46. kod procedure za brisanje blagajnika

Procedura za brisanje blagajnika malo je kompleksnija od procedura za dodavanje blagajnika. Prvo provjerava da li postoji ID određenog zaposlenika kojeg želimo brisati, ako ne postoji onda vrati „Ne postoji taj zaposlenik!“, u suprotnom vrati nam da je „zaposlenik uspješno obrisan“. Ne preporučuje se nikad brisanje zaposlenika radi praćenja analize računa i slično.

## 4.3.9 Procedure za dodavanje ili brisanje artikla i goriva

Graphical user interface, text, application

Description automatically generated

Slika 47. kod procedure za dodavanje artikla

Procedura za dodavanje vrlo jednostavan insert u transakciji. Prima 4 parametra; id, vrsta, naziv, cijena. Ako je artikl dodan uspješno vrati nam poruku „Uspješno dodan artikl“.

Text

Description automatically generated

Slika 48. kod procedure za dodavanje goriva

Identična procedura kao i na slici 47. samo što se razlikuje što dodavamo u tablicu „gorivo“ a ne „artikl“ kao na proceduri iznad.

## 4.3.10. Procedura za brisanje proizvoda

Text

Description automatically generated with medium confidence

Slika 49. kod procedure za brisanje proizvoda

Procedura za brisanje proizvoda prima jedan jedini parametar, naziv, koji upisujemo za brisanje proizvoda. Ako nije dobar naziv proizvoda procedura vraća „Ne postoji taj proizvod“. Ako je naziv točan onda se proizvod uspješno obriše. Ovakva procedura je moguća zato što imamo pogled proizvodi, gdje se nalaze svi proizvodi.

Ovo je ujedno i zadnja procedura koju ima naša benzinska postaja.

# 5. KORISNICI

Korisnici (*en. users*) u MySQL je zapis u USER tablici poslužitelja koji sadrži podatke za prijavu (korisničko ime i lozinka), privilegije računa i podatke o hostu za MySQL račun. Kada stvaramo korisnika imamo mu pravo davati privilegije što smije a što ne smije dirati. U našoj benzinskoj, voditelj će imati pristup svim procedurama, funkcijama i slično, dok blagajnici će biti ograničeni na samo nekoliko procedura koje su potrebne za obavljati kupnju, tipkati račun i slično.

## 5.1 Voditelj



Slika 49. kod korisnika „voditelj“

Korisnik „voditelj“ vrlo je lagano stvoren, doslovno dve linije koda. Druga linija prikazuje da smo mu komandom „grant all privileges on“ dali pravo na korištenje svih procedura, funkcija, pogleda i tablica.

## 5.2 Blagajnik

Slika 50. kod korisnika „blagajnik“

Text

Description automatically generated with medium confidence

Korisnik „blagajnik“ razlikuje se od korisnika „voditelj“ jer nema pravo na sve kao voditelj, nego na određene poglede, funkcije, procedure. Kao blagajnik ne može dodavati ili brisati nove korisnike, artikle i goriva nego to radi samo voditelj. Od blagajnika mnogo procedura i funkcija je sakriveno.

# 6. ZAKLJUČAK

U današnjem modernom dobu, to jest „internet doba“, baze podataka susrećemo svakodnevno, na razno raznim portalima, aplikacijama i slično. To bi značilo da poznavanje rada bazi podataka su vrlo bitne i vrlo lako ih je primjenjivati na razne sustave.

Ovim projektom stvorili smo bazu podataka za jednu zamišljenu benzinsku postaju. Ima razno raznih procedura, funkcija i tablica za obavljanje kupnje, tipkanje računa, dodavanje artikla, brisanje artikala i mnoge druge stvari. Postoje dvije vrste korisnika voditelj i blaganik. Sa grafičkim sučeljem poboljšali bi rad ovog projekta, i bilo bi lakše koristiti sve te procedure.

# 7. LITERATURA

1. <https://www.mysql.com/>

2. <https://hr.wikipedia.org/wiki/MySQL>

3. <https://www.w3schools.com/>

4. [www.youtube.com](http://www.youtube.com)

5. <www.google.com>

6. <https://www.postgresql.org/>

7. <https://www.mysql.com/products/workbench/>

8. <https://support.microsoft.com/hr-hr/office/osnove-baza-podataka-a849ac16-07c7-4a31-9948-3c8c94a7c204>

1. SQL – Strukturirani jezik upita, služi za komuniciranje sa bazom podataka [↑](#footnote-ref-1)
2. <https://www.mysql.com/> [↑](#footnote-ref-2)
3. <https://www.postgresql.org/> [↑](#footnote-ref-3)
4. <https://www.mysql.com/products/workbench/> [↑](#footnote-ref-4)
5. INTEGER – tip podataka koji ima 4 byta, i sprema vrijednosti od -2147483647 do 2147483647 [↑](#footnote-ref-5)
6. NOT NULL – ograničenje koje osigurava da vrijednosti pohranjene u stupcu nisu NULL. [↑](#footnote-ref-6)
7. VARCHAR – tip podataka koji sprema vrijednosti od 0 do 65535 byta, te sprema slova i brojeve i posebne znakove [↑](#footnote-ref-7)
8. CHAR – slično kao i varchar, samo sprema slova. Duljina može bit od 0 do 255 [↑](#footnote-ref-8)
9. AUTO\_INCREMENT – koristi se za automatsko dodjeljivanje brojeva kada novi zapis dodajemo u tablicu [↑](#footnote-ref-9)
10. UNIQUE – sve što je spremljeno u stupac je unikatno, ne može se ponavljati 2 iste vrijednosti. [↑](#footnote-ref-10)
11. DATETIME – tip podataka koji služi za spremanje datuma i vremena [↑](#footnote-ref-11)
12. NOW() – funkcija koja vraća trenutni datum i vrijeme. [↑](#footnote-ref-12)
13. CHECK – ograničenje kad želimo nešto ograničiti. (npr. da plaća nije manja od 3800 kuna.) [↑](#footnote-ref-13)
14. NUMERIC – tip podataka koji služi za brojeve sa decimalama. (lijevi broj u zagradi je ukupno brojeva, a desni znači na koliko decimala zaokružujemo) [↑](#footnote-ref-14)
15. Pogled – *eng. view* - su virtualne tablice koje ne pohranjuju vlastite podatke već prikazuju podatke drugih tablica [↑](#footnote-ref-15)
16. Strani ključ – *eng. foreign key* vrsta ograničenja koja se mogu koristiti za održavanje itegriteta između tablica. [↑](#footnote-ref-16)
17. TRUNCATE – briše sve podatke iz tablice, auto\_increment brojčanik se resetira. [↑](#footnote-ref-17)
18. SELECT – koristi se za dohvaćanje redaka iz jedne ili više tablica [↑](#footnote-ref-18)
19. UNION – operator koji kombinira rezultat iz dvaju ili više tablica. Kad koristimo union moramo imati isti broj stupaca. Isto tako stupci moraju imati slične tipove podataka. [↑](#footnote-ref-19)
20. OKIDAČ – *en.trigger* - pohranjeni program koji se automatski poziva kad se umetne, ažurira ili obriše zapis u tablici. [↑](#footnote-ref-20)
21. COUNT – funkcija koja vraća broj redaka u tablici. [↑](#footnote-ref-21)
22. TRANSAKCIJA – skupina izjava, upita, operacija koje se prije umetanja, ažuriranja ili brisanja mogu predati(commit) ili povratiti prijašnji podaci. [↑](#footnote-ref-22)
23. LEFT JOIN – vraća podatke iz tablice 1 i tablice 2. NULL rezultat se javlja ako tablica 2 nema podudaranja sa tablicom 1. [↑](#footnote-ref-23)
24. INNER JOIN – vraća sve zapise koje imaju podudarne vrijednosti [↑](#footnote-ref-24)
25. WHERE – omogućuje da odredimo uvjet pretraživanja tablica [↑](#footnote-ref-25)