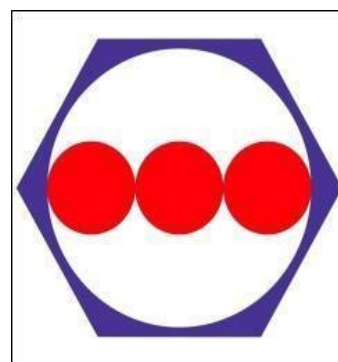




UNIVERZITET U NOVOM SADU
FAKULTET TEHNIČKIH NAUKA
Departman za industrijsko inženjerstvo
i industrijski menadžment



Informacioni sistem za lanac prodavnica
-Projektovanje baza podataka-

Profesor: dr Sonja Ristić
Korisnik: Dajana Narandžić
Student: Daniel Mocko IT54/2015

Novi Sad, Februar 2018

Sadržaj:

1. Opis problema	3
1.1 Cilj i značaj projekta	3
1.2 Opseg projekta	3
1.3 Opis domena	4
1.4 Okruženje projekta	4
1.5 Korisnici sistema	4
1.6 Opis upotrebe Sistema	5
1.7 Funkcionalni zahtevi	6
1.8 Ograničenja, pretpostavke i uslovljenosti	7
2. ER model	8
3. Tabelarni prikaz obelezja	8
4. Šema relacione baze podataka	18
5. DDL	22
6. DML	37
7. Select upiti	50
8. Funkcije	53
9. Trigeri	56
10. Procedure	60
11. Zaključak	65

1.Opis problema

1.1 Cilj i značaj projekta

Cilj projekta je modelovanje baze podataka koja ima zadatak da prikuplja informacije iz realnog sistema i olakša kontrolu i upravljanje poslovanjem lanca prodavnica tehničke opreme. Pregledom i razumevanjem činilaca i njihovih odnosa u realnom sistemu, dolazi se do potrebnih informacija i ideja za kreiranje odgovarajuće baze podataka kako bi učinili podatke lako dostupnim korisnicima sistema. Značaj ovakve automatizacije je u brznoj, sigurnoj i pouzdanoj razmeni podataka među korisnicima. Čuvanjem svih, za sistem relevantnih podataka, dobija se pregled celokupnog poslovanja i funkcionisanja preduzeća što olakšava uočavanje eventualnih slabosti ili prostora za poboljšanje rada sistema.

1.2 Opseg projekta

a. Naručivanje proizvoda

Naručivanje potrebnih proizvoda iz drugih prodavnica ili skladišta.

b. Transport i isporuku proizvoda

Transport naručenih proizvoda od skladišta do odgovarajuće prodavnice ili između prodavnica.

c. Kontrolu akcija

Sprovođenje akcija i odgovarajućih popusta i čuvanje svih informacija o prodajnim cenama proizvoda na akciji.

d. Bonus za prodavce

Prikupljanje i čuvanje svih potrebnih informacija na osnovu kojih se dodeljuje bonus na platu prodavcima.

e. Online prodavnicu

Povezivanje online prodavnice sa prodavnicama, kreiranje online porudžbina od strane kupca i slanje u prodavnicu na dalju obradu.

f. Kontrolu prevoznih sredstava

Pregled informacija o teretnim i civilnim vozilima preduzeća.

g. Planiranje i izveštavanje

Planiranje vrše zaposleni na odgovarajućim radnim pozicijama za obilazak radnji u cilju vršenja kontrole njihovog poslovanja.

1.3 Opis domena

Domen projekta obuhvata:

- a. zaposlene u preduzeću:** pregled pozicija u hijerarhiji, informacije o zaposlenima, njihovim radnim mestima, platama i bonusima, i planove raspodele poslova i izveštaje
- b. objekte preduzeća:** prodavnice, magacine i skladišta u posedu preduzeća
- c. prevozna sredstva:** pregled informacija o teretnim i civilnim vozilima preduzeća
- d. proizvode:** informacije o proizvodima (gde se nalaze na stanju i u kojoj količini) njihovoj dostupnosti u prodavnicama i *online* prodavnici, cenama i eventualnim akcijama i popustima
- e. narudžbine:** iz skladišta, prodavnice i *online* prodavnice
- f. transport:** vozila, vozače, stanje pošiljki u transportu

1.4 Okruženje projekta

Baza podataka koja treba da automatizuje poslovanje lanca prodavnica uz pomoć informacija preuzetih od:

- a. ljudskih resursa**
- b. administracije**

a. Ljudski resursi: podrazumevaju sve zaposlene unutar lanca prodavnica. Pod tim podrazumevamo njegovu funkciju i zaduženja.

b. Administracija: omogućava da se prati u kojoj prodavnici se nalazi proizvod i da li je proizvod u transportu. Pored toga se prati i u kojoj količini se određeni proizvod nalazi u kom magacinu ili prodavnici.

1.5 Korisnici sistema

a. Administrator baze podataka-održava bazu podataka, shodno tome ima pristup svim podacima koji su zapisani u nju

b. Zaposleni:

b.1. Nadređeni-kontrolisu i donose odluke na osnovu podataka koji su smešteni u bazi

b.2. Prodavac-prodaje proizvod, prihvata *online* porudžbine, proverava dostupnost proizvoda, šalje zahtev za porudžbinom, priprema pristiglu porudžbinu

b.3. Skladištar-prihvata i obrađuje porudžbine iz prodavnica

b.4. Vozač-transportuje proizvode po porudžbini do prodavnica

c. Kupci-imaju pregled dostupnih proizvoda u *online* prodavnici

1.6 Opis upotrebe Sistema

a. Administrator baze podataka

Uloga administratora našeg informacionog sistema je da ažurira sistem, prati bilo kakve eventualne promene kao što su npr. promene radnih mesta kod zaposlenih, promene zaposlenih (zapošljavanja/ostavke) i promene cena proizvoda.

b. Zaposleni, u našem informacionom sistemu imaju sledeće pozicije:

b.1. Generalni direktor - samo jedna osoba u ovom celom projektu koja je po hijerarhijskoj skali iznad svih zaposlenih. On je glavni i odgovorni za prodavnice iz lanca.

b.2. Zamenik direktora - osoba koja je postavljena da u slučaju odsustva generalnog direktora iz bilo kakvog razloga ili njegove nemogućnosti obavljanja neke dužnosti, ima jednaka prava i odgovornosti kao direktor.

b.3. Regionalni direktor – nadređen mu je direktor zamenik, a on je odgovoran za više prodavnica unutar jednog regiona. Po potrebi ima ulogu i regionalnog menadžera ako neka prodavnica, u regiji u kojoj on radi, nema istog.

b.4. Regionalni menadžer – nadređen mu je regionalni direktor. Njegova uloga je da piše planove i izveštaje na dnevnom nivou i kontroliše izveštaje administratora i aranžera.

b.5. Administrator – nadređen mu je regionalni menadžer. On je odgovorno lice za sve administrativne poslove unutar prodavnice. Kontroliše računovodstvo i knjigovodstvo na nivou prodavnice.

b.6. Aranžer - nadređen mu je regionalni menadžer. Zadužen je za nadgledanje i organizovanje izlaganja proizvoda u prodavnicama.

c. Prodavac – Lice koje obavlja trgovačke delatnosti. Zaposlen je u tačno određenoj prodavnici. Na nivou prodavnice, jedan prodavac je menadžer prodavnice i ima svog zamenika. Prodavcima se računa bonus na platu u odnosu na količinu prodatih proizvoda u određenom vremenskom periodu.

d. Vozač – lice koje upravlja teretnim vozilom i prevozi proizvode po porudžbini od skladišta do odgovarajuće prodavnice.

e. Skladištar – osoba zaposlena u skladištu, zadužena za pripremu porudžbina za transport.

f. Kupac – fizičko lice koje kupuje proizvode raznih kategorija, unutar prodavnice, po određenim cenama, zarad svojih ličnih potreba.

1.7 Funkcionalni zahtevi

- a.** Preglednost hijerarhije zaposlenih unutar organizacije: sistem omogućava preglednost raspodele odgovornosti poslova u sistemu u smislu pregleda dnevnih i nedeljnih planova i izveštaja na nivou zaposlenih i nadređenih za svaku prodavnicu na nivou prodavnice. Takođe, na nivou svih prodavnica, dostupan je pregled i raspored zaposlenih prodavaca po lokacijama, njihove aktivnosti kao i odgovorni prodavac menadžer i njegov zamenik za svaku prodavnicu.
- b.** Proces planiranja: sistem omogućava svakom zaposlenom na poziciji aranžera, administratora i regionalnog menadžera pisanje individualnih dnevnih izveštaja i nedeljnih planova, koji su u svakom trenutku dostupni na uvid regionalnom direktoru koji im je nadređen..
- c.** Automatizacija nabavke unutar sistema: sistem omogućava slanje porudžbine iz prodavnica u druge prodavnice zahtevom ili porudžbenicama koje se u skladištu dalje obrađuju. Pripremljenu porudžbinu sa fakturom dalje preuzimaju zaposleni u transportu (vozači) i prevoze u odgovarajuću prodavnicu. Podaci o svakoj porudžbini u transportu (status aktivnosti, vozilo, vozač) su dostupni odgovornim licima u hijerarhiji u svakom trenutku.
- d.** Vođenje evidencije prodatih proizvoda: sistem omogućava pregled koji proizvod je kada prodat, na kojoj lokaciji i koji prodavac je prodao proizvod što automatizuje proces praćenja prodaje na nivou sistema, na nivou prodavnice i pruža mogućnost računanja bonusa prodavcu.
- e.** Automatizacija *online* poručivanja: online porudžbine se sistemom direktno prosleđuju u odgovarajuće prodavnice gde se dalje obrađuju.

1.8 Ograničenja, pretpostavke i uslovljenosti

a. Zaposleni

Zaposleni može imati samo jednu funkciju u sistemu. Može biti skladištar, vozač, prodavac ili nadređeni. Nadređeni zaposleni može biti zamenik direktora, regionalni direktor, regionalni menadžer, administrator ili aranžer, i mogu biti zaduženi za više prodavnica. Prodavac može biti menadžer ili zamenik menadžera. Svaki menadžer je zadužen za prodavnicu u kojoj radi. Prodavac može biti zaposlen u samo jednoj prodavnici. Skladištar radi u tačno jednom skladištu. Ukoliko za prodavnicu u regionu regionalnog direktora nema zaduženog regionalnog menadžera, on preuzima ulogu istog.

b. Prodavnica

U jednom gradu može se nalaziti više prodavnica. Svaka prodavnica ima svoj magacin na istoj lokaciji.

c. Plan

Svaki plan važi za jednog zaposlenog: regionalnog menadžera, administratora ili aranžera. Svaki plan se piše za tačno određeni datum. Shodno planu, za svaki plan određenog datuma piše se izveštaj od strane zaposlenog za kog je kreiran plan.

d. Proizvod

Proizvod pripada tačno jednoj vrsti koja pripada tačno jednoj kategoriji. Svaki proizvod mora imati tri cene: osnovna, komercijalna i prodajna cena. Proizvod je predstavljen sa najmanje jednom slikom. Proizvod poseduje *online* cenu samo ukoliko se nalazi u *online* prodavnici. Kada je proizvod na akciji, popust se računa samo na prodajnu cenu.

e. Prevozna sredstva

Prevozna sredstva se dele na transportna i civilna vozila. Transportna vozila su dodeljena vozačima, a civilna nadređenima. Vozačima i nadređenima ne mora u svakom trenutku biti dodeljeno neko od odgovarajućih vozila, ali svako vozilo mora biti raspoređeno odnosno dodeljeno odgovarajućem zaposleom tako da ne postoje neraspoređena vozila.

[illegible]

2. Tabelarni prikaz obeležja

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
SIF_Z	Šifra zaposlenog	Number	8	⊥	$d \geq 0$
IME	Ime	String	30	⊥	Δ
PREZIME	Prezime	String	30	⊥	Δ
JMBG	Jedinstveni matični broj zaposlenog	Number	13	⊥	$d \geq 0$
ADRESA	Adresa	String	30	⊥	Δ
KONTAKT_TELEFON	Kontakt telefon	Number	10	⊥	$d \geq 0$
MAIL	E-mail	String	20	T	Δ
PLATA	Plata	Number	10	⊥	$d \geq 0$
RADNO_MESTO	Stručna sprema	String	30	⊥	Δ
K={SIF_Z}					

Tabela 1 : Zaposleni

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
BR_KARTICE	Menadžer prodavnice	Number	8	T	$d \geq 0$
K={SIF_Z}					

Tabela 2 : Prodavac

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
KATEGORIJA	Položene kategorije	String	20	⊥	Δ
BR_VOZ_DOZVOLE	Broj vozačke dozvole	Number	9	⊥	$d \geq 0$
K = {SIF_Z}					

Tabela 3 : Vozač

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
KATEGORIJA	Kategorija za viljuskar		1	T	Δ
K = {SIF_Z}					

Tabela 4 : Skladištar

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
NADREDJEN	Nadređen	Number	8	\perp	$d \geq 0$
K = {SIF_Z}					

Tabela 5 : Nadređeni

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
SIF_VOZILA	Šifra vozila	Number	8	\perp	$d \geq 0$
MODEL	Model	String	30	\perp	Δ
BR_REG_TAB	Broj registarskih tablica	String	10	\perp	Δ
DATUM_ISTEKA_REGISTRACIJE	Datum isteka registracije	Date		\perp	Δ
DATUM_ZADUZIVANJA	Datum zaduživanja	Date		\perp	Δ
DATUM_RAZDUZIVANJA	Datum razduživanja	Date		\perp	Δ
K = {SIF_VOZILA}					

Tabela 6 : Prevozno sredstvo

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
K = {SIF_VOZILA}					

Tabela 7 : Civilno vozilo

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
Marka	Marka vozila	String	3	⊥	Δ
K = {SIF_VOZILA}					

Tabela 8 : Teretno vozilo

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
SIF_GRADA	Šifra grada	Number	8	⊥	d ≥ 0
NAZIV	Ime grada	String	30	⊥	Δ
K = {SIF_GRADA}					

Tabela 9 : Grad

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
SIF_REGION	Šifra regiona	number	8	⊥	d ≥ 0
NAZIV	naziv	varchar	40	⊥	Δ
K = {SIF_REGION}					

Tabela 6 : Region

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
SIF_PRODAVNICE	Šifra prodavnice	Number	8	⊥	d ≥ 0
ADRESA	Adresa prodavnice	String	30	⊥	Δ
KONTAKT_TELEFON	Kontakt telefon	Number	10	⊥	d ≥ 0
RADNO_VREME_RD	Radno vreme radni dan	String	10	⊥	Δ
RADNO_VREME_SUBOTA	Radno vreme subota	String	10	⊥	Δ
RADNO_VREME_NEDELJA	Radno vreme nedelja	String	10	⊥	Δ
K = {SIF_PRODAVNICE}					

Tabela 11 : Prodavnica

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
SIF_MAGACINA	Šifra magacina	Number	8	⊥	$d \geq 0$
K = {SIF_PRODAVNICE + SIF_MAGACINA}					

Tabela 12 : Magacin

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
KOLICINA	Količina proizvoda u magacinu	Number	5	⊥	$d \geq 0$
K={SIF_PRODAVNICE+ SIF_MAGACINA + SIF_PROIZVODA}					

Tabela 13 : ProizMag

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
KOL_PROIZ	Količina proizvoda	Number	5	⊥	$d \geq 0$
K={SIF_PRODAVNICE + SIF_PROIZVODA}					

Tabela 14 : ProizProd

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
SIF_PROIZVODA	Šifra proizvoda	Number	8	⊥	$d \geq 0$
NAZIV_PROIZVODA	Naziv proizvoda	String	8	⊥	Δ
OPIS	Opis proizvoda	String		T	Δ
OSNOVNA_CENA	Osnovna cena	Number	10	⊥	$d \geq 0$
PRODAJNA_CENA	Prodajna cena	Number	10	⊥	$d \geq 1$
KOMERCIJALNA_CENA	Komercijalna cena	Number	10	⊥	$d \geq 2$
KATEGORIJA	Kategorija proizvoda	String	50	⊥	Δ
VRSTA	Vrsta proizvoda	String	50	⊥	Δ
K = {SIF_PROIZVODA + SIF_VRSTE + SIF_KATEGORIJE}					

Tabela 15 : Proizvod

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
SIF_SLIKE	Šifra slike	Number	8	⊥	$d \geq 0$
PUTANJA_FAJLA	Putanja fajla	String	100	⊥	Δ
K = {SIF_SLIKE }					

Tabela 16 : Slika

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
K = {SIF_SLIKE + SIF_PROIZVODA}					

Tabela 17 : Predstavlja

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
SIF_AKCIJE	Šifra akcije	Number	8	⊥	$d \geq 0$
DATUM_POCETKA	Datum početka akcije	Date		⊥	Δ
DATUM_ZAVRSETKA	Datum završetka akcije	Date		⊥	Δ
PROCENAT	Procenat	Number	3	⊥	$d \geq 0$
K = {SIF_AKCIJE}					

Tabela 18 : Akcija

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
CENA_NA_AKCIJI	Cena proizvoda koji je na akciji	Number	10	⊥	$d \geq 0$
K = {SIF_AKCIJE + SIF_PROIZVODA}					

Tabela 19 : ProizAkc

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
SIF_KUPCA	Šifra online KUPCA	Number	8	⊥	$d \geq 0$
IME	Ime kupca	String	20	⊥	Δ
PREZIME	Prezime kupca	String	20	⊥	Δ
EMAIL	Email	String	100	⊥	Δ
KONTAKT	URL adresa	String	12	⊥	Δ
K = {SIF_KUPCA}					

Tabela 20 : Online kupac

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
SIF_ONLINE_PROIZVODA		Number	8	⊥	$d \geq 0$
DOSTUPNO		Boolean		⊥	
ONLINE_CENA	Online cena	Number	8	⊥	$d \geq 0$
K = {SIF_ONLINE_PROIZVODA}					

Tabela 21 :Online Proizvod

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
SIF_PORUDZBENICE	Šifra porudžbine	Number	8	⊥	$d \geq 0$
KOLICINA	Količina proizvoda	Number	5	⊥	$d \geq 0$
AKTIVNA	Stanje porudžbine	Boolean		⊥	Δ
K = {SIF_PORUDZBENICE}					

Tabela 22 : Porudžbenica

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
SIF_PLANA	Šifra plana	Number	8	⊥	$d \geq 0$
DATUM	Datum za koji se odnosi plan	Date		⊥	Δ
TXT_OPIS	Opis plana	String		⊥	Δ
NEDELJA	Redni broj nedelje u godini	Number	2	⊥	$d \geq 0$
DAN_U_NEDELJI	Redni broj dana u nedelji	Number	1	⊥	$d \geq 0$
K = {SIF_PLANA}					

Tabela 23 : Plan

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
SIF_IZVESTAJA	Šifra izveštaja	Number	8	⊥	$d \geq 0$
NEDELJA	Redni broj nedelje u godini	Number	2	⊥	$d \geq 0$
DAN_U_NEDELJI	Redni broj dana u nedelji	Number	1	⊥	$d \geq 0$
DATUM	Datum	Date		⊥	Δ
TXT_IZVESTAJA	Tekstualni opis izveštaja	String		⊥	Δ
K = {SIF_PLANA + SIF_IZVESTAJA}					

Tabela 24 : Izveštaj

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
SIF_PRODAJE	Šifra bonusa	Number	8	⊥	$d \geq 0$
KOLICINA	Količina	Number	8	⊥	$d \geq 0$
VREME_PRODAJE	Vreme prodaje	Date		⊥	$d \geq 0$
DATUM_PRODAJE	Datum prodaje	Date		⊥	$d \geq 0$
BONUS_NA_PRODAJU	Iznos bonusa	Number	8	⊥	$d \geq 0$
K = {SIF_PRODAJE + SIF_BONUSA}					

Tabela 25 : Prodaja

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
SIF_SKLADISTA	Šifra skladišta	Number	8	⊥	$d \geq 0$
KONTAKT_TEL	Kontakt telefon	Number	20	⊥	$d \geq 0$
LOKACIJA	Lokacija skladišta	String	100	⊥	Δ
K = {SIF_SKLADISTA}					

Tabela 26 : Skladište

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
KOLICINA	Količina proizvoda u skladištu	Number	5	⊥	$d \geq 0$
K = {SIF_SKLADISTA + SIF_PROIZVODA}					

Tabela 27 : Je na stanju

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
SIF_PORUDZBINE	Šifra porudžbine	Number	6	⊥	$d \geq 0$
DATUM	Date			⊥	$d \geq 0$
AKTIVNA	String	String	8	T	$d \geq 0$
K = {SIF_PORUDZBINE}					

Tabela 28 : Porudžbina

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
VREME_OBRADE	Vreme obrade	Date		⊥	
K = {SIF_PORUDZBINE }					

Tabela 29 : Obradjuje iz skladišta

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
STATUS_ISPORUKE	Status	String	20	⊥	Δ
K = {SIF_PORUDZBINE }					

Tabela 7 : Transportuje iz skladišta

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
SIF_ZAHTEVA	Šif zahteva	Number	5	⊥	$d \geq 0$
AKTIVAN	Status zahteva	String		⊥	Δ
DATUM_KREIRANJA	Datum kreiranja	Date		⊥	
KOLICINA	Količina	Number	5	⊥	$d \geq 0$
K = { SIF_ZAHTEVA}					

Tabela 8 : Zahtev

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
KOLICINA	Količina	Number	5	⊥	Δ
K = {SIF_ZAHTEVA}					

Tabela 9 : Obraduje iz prodavnice

Naziv obeležja	Opis obeležja	Tip podatka	Dužina podataka	Null	Uslov
STATUS_ISPORUKE	Status isporuke	String	20	⊥	d ≥ 0
K = {VOZAC+SKLADISTAR+SIF_PORUDZBINE + SIF_Z + SIF_SKLADISTA + SIF_PROIZVODA}					

Tabela 10 : TransportujeŠema relacione baze podataka

3. Šema relacije baze podataka

ZAPOSLENI ($\{sif_z, ime, prezime, jmbg, adresa, kontakt_telefon, mail, plata, radno_mesto, sef\}, \{sif_z\}$)

$Dom(sef) = Dom(sif_z)$

$Zaposleni[sef] \subseteq Zaposleni[sif_z]$

VOZAC ($\{sif_z, kategorija, br_voz_dozvole\}, \{sif_z\}$)

$VOZAC[sif_z] \subseteq ZAPOSLENI[sif_z]$

SKLADISTAR ($\{sif_z, sif_skladista\}, \{sif_z\}$)

$SKLADISTAR[sif_z] \subseteq ZAPOSLENI[sif_z]$

$SKLADISTAR[sif_skladista] \subseteq SKLADISTE[sif_skladista]$

$NULL(SKADISTAR, sif_skladista) = \perp$

PRODAVAC ($\{sif_z, sif_prodavnice, br_kartice\}, \{sif_z\}$)

$PRODAVAC[sif_z] \subseteq ZAPOSLENI[sif_z]$

$PRODAVAC[sif_prodavnice] \subseteq PRODAVNICA[sif_prodavnice]$

$NULL(PRODAVAC, sif_prodavnice) = T$

NADREDJENI ($\{sif_z, nadredjen\}, \{sif_z\}$)

$NADREDJENI[sif_z] \subseteq ZAPOSLENI[sif_z]$

$NADREDJENI[sif_regiona] \subseteq REGION[sif_regiona]$

$NULL(NADREDJENI, sif_regiona) = T$

$PRODAVAC[sif_z] \cap VOZAC[sif_z] = \emptyset$

$PRODAVAC[sif_z] \cap NADREDJENI[sif_z] = \emptyset$

$PRODAVAC[sif_z] \cap SKLADISTAR[sif_z] = \emptyset$

$VOZAC[sif_z] \cap NADREDJENI[sif_z] = \emptyset$

$VOZAC[sif_z] \cap SKLADISTAR[sif_z] = \emptyset$

$SKLADISTAR[sif_z] \cap NADREDJENI[sif_z] = \emptyset$

$ZAPOSLENI[sif_c] \subseteq PRODAVAC[sif_z] \cup VOZAC[sif_z] \cup SKLADISTAR[sif_z] \cup$

$NADREDJENI[sif_z]$

PREVOZNO_SREDSTVO($\{sif_vozila, model, br_reg_tablice, datum_isteka_registracije, datum_zaduzivanja, datum_razdusivanja\}, \{sif_vozila\}$)

CIVILNO_VOZILO ($\{sif_vozila, dodeljeno\}, \{sif_vozila\}$)

$CIVILNO_VOZILO[sif_vozila] \subseteq PREVOZNO_SREDSTVO[sif_vozila]$

$CIVILNO_VOZILO[dodeljeno] \subseteq NADREDJENI[sif_z]$

$Dom(dodeljeno) = Dom(sif_z)$

$NULL(CIVILNO_VOZILO, dodeljeno) = \perp$

$UNIQUE(CIVILNO_VOZILO, dodeljeno) = T$

TERETNO_VOZILO ($\{sif_vozila, kategorija, vozac\}, \{sif_vozila\}$)
 $TERETNO_VOZILO[sif_vozila] \subseteq PREVOZNO_SREDSTVO[sif_vozila]$
 $TERETNO_VOZILO[vozac] \subseteq VOZAC[sif_z]$
 $Dom(vozac) = Dom(sif_z)$
 $NULL(TERETNO_VOZILO, vozac) = \perp$
 $UNIQUE(TERETNO_VOZILO, vozac) = T$

$CIVILNO_VOZILO[sif_vozila] \cap TERETNO_VOZILO[sif_vozila] = \emptyset$
 $PREVOZNO_SREDSTVO[sif_vozila] \subseteq CIVILNO_VOZILO[sif_vozila]$ U
 $TERETNO_VOZILO[sif_vozila]$

GRAD($\{sif_grada, naziv, postanski_broj, naziv_drzave\}, \{sif_grada\}$)
 $GRAD[sif_regiona] \subseteq REGION[sif_regiona]$
 $NULL(GRAD, sif_regiona) = \perp$

PRODAVNICA($\{sif_prodavnice, sif_grada, adresa, kontakt_telefon, radno_vreme, površina,, administrator, aranzer, menadzer, kontakt_telefon\}, \{sif_prodavnice\}$)
 $PRODAVNICA[sif_grada] \subseteq GRAD[sif_grada]$
 $NULL(PRODAVNICA, sif_grad) = \perp$

$PRODAVNICA[administrator] \subseteq NADREDJENI[sif_z]$
 $Dom(administrator) = Dom(sif_z)$
 $NULL(PRODAVNICA, administrator) = T$

$PRODAVNICA[aranzer] \subseteq NADREDJENI[sif_z]$
 $Dom(aranzer) = Dom(sif_z)$
 $NULL(PRODAVNICA, aranzer) = T$

$PRODAVNICA[MENADZER] \subseteq NADREDJENI[sif_z]$
 $Dom(MENADZER) = Dom(sif_z)$
 $NULL(PRODAVNICA, MENADZER) = T$

MAGACIN ($\{sif_prodavnice, sif_magacina\}, \{sif_prodavnice + sif_magacina\}$)
 $MAGACIN[sif_prodavnice] \subseteq PRODAVNICA[sif_prodavnice]$
 $PRODAVNICA[sif_prodavnice] \subseteq MAGACIN[sif_prodavnice]$
 $NULL(MAGACIN, sif_prodavnice) = \perp$
 $UNIQUE(MAGACIN, sif_prodavnice) = T$

SE_NALAZI_U_MAGACINU($\{sif_prodavnice, sif_magacina, sif_proizvoda, kolicina\}, \{sif_prodavnice + sif_magacina + sif_proizvoda\}$)
 $NA_STANJU_U_MAGACINU[sif_proizvoda] \subseteq PROIZVOD[sif_proizvoda]$
 $NA_STANJU_U_MAGACINU[sif_prodavnice + sif_magacina] \subseteq MAGACIN[sif_prodavnice + sif_magacina]$

PROIZ_PROD ($\{sif_prodavnice, sif_proizvoda, kolicina\}, \{sif_prodavnice + sif_proizvoda\}$)
 $SE_NALAZI_U_PRODAVNICI[sif_proizvoda] \subseteq PROIZVOD[sif_proizvoda]$
 $SE_NALAZI_U_PRODAVNICI[sif_prodavnice] \subseteq PRODAVNICA[sif_prodavnice]$

PROIZVOD ($\{sif_proizvoda, naziv_proizvoda, kategorija, vrsta, opis, osnovna_cena, komercijalna_cena, prodajna_cena\}, \{sif_proizvoda\}$)

SLIKA({sif_slike, putanja_fajla}, {sif_slike})

PREDSTAVLJA({sif_slike,sif_proizvoda},{sif_slike+sif_proizvoda})
 $\text{PREDSTAVLJA}[\text{sif_proizvoda}] \subseteq \text{PROIZVOD}[\text{sif_proizvoda}]$
 $\text{PROIZVOD}[\text{sif_proizvoda}] \subseteq \text{PREDSTAVLJA}[\text{sif_proizvoda}]$
 $\text{PREDSTAVLJA}[\text{sif_slike}] \subseteq \text{SLIKA}[\text{sif_slike}]$
 $\text{SLIKA}[\text{sif_slike}] \subseteq \text{PREDSTAVLJA}[\text{sif_slike}]$

AKCIJA({sif_akcije, datum_pocetka, datum_zavrsetka, procenat}, {sif_akcije})

JE_NA ({sif_akcije, sif_proizvoda, cena_na_akciji}, {sif_akcije+sif_proizvoda})
 $\text{JE_NA}[\text{sif_akcije}] \subseteq \text{AKCIJA}[\text{sif_akcije}]$
 $\text{JE_NA}[\text{sif_proizvoda}] \subseteq \text{PROIZVOD}[\text{sif_proizvoda}]$

ONLINE_PROIZVOD ({sif_online_proizvoda, cena, dostupno}, {sif_online_proizvoda })
 $\text{ONLINE_PROIZVOD}[\text{sif_proizvoda}] \subseteq \text{PROIZVOD}[\text{sif_proizvoda}]$
 $\text{NULL}(\text{ONLINE_PROIZVOD}, \text{sif_prodavnice}) = \perp$

ONLINE_KUPAC({sif_kupca,ime,prezime,email,kontat},{ sif_kupca })

PORUDZBENICA ({sif_porudzbence,KOLICINA,aktivna},{sif_porudzbence})
 $\text{PORUDZBENICA}[\text{sif_prodavnice}] \subseteq \text{PRODAVNICA}[\text{sif_prodavnice}]$
 $\text{NULL}(\text{PORUDZBENICA}, \text{sif_prodavnice}) = \perp$

$\text{PORUDZBENICA}[\text{sif_online_proizvoda}] \subseteq \text{ONLINE_PROIZVOD}[\text{sif_online_proizvoda}]$
 $\text{NULL}(\text{PORUDZBENICA}, \text{sif_online_proizvoda}) = \perp$

$\text{PORUDZBENICA}[\text{sif_kupca}] \subseteq \text{ONLINE_KUPAC}[\text{sif_kupca}]$
 $\text{NULL}(\text{PORUDZBENICA}, \text{sif_kupca}) = \perp$

PLAN({sif_plana, sif_z, datum, nedelja, dan_u_nedelji, txt_opis},{sif_plana})
 $\text{PLAN}[\text{sif_z}] \subseteq \text{NADREDJENI}[\text{sif_x}]$
 $\text{NULL}(\text{PLAN}, \text{sif_z}) = \perp$

IZVESTAJ({sif_plana, datum, nedelja, dan_u_nedelji, sif_izvestaja, txt_izvestaja},
{sif_plana+sif_izvestaja})
 $\text{IZVESTAJ}[\text{sif_plana}] \subseteq \text{PLAN}[\text{sif_plana}]$
 $\text{PLAN}[\text{sif_plana}] \subseteq \text{IZVESTAJ}[\text{sif_plana}]$

PRODAJA({sif_prodaje, kolicina, vreme, datum, bonus}, { sif_prodaje })
 $\text{PRODAJA}[\text{sif_z}] \subseteq \text{PRODAVAC}[\text{sif_z}]$
 $\text{NULL}(\text{PRODAJA}, \text{sif_z}) = \perp$

$\text{PRODAJA}[\text{sif_proizvoda}] \subseteq \text{PROIZVOD}[\text{sif_proizvoda}]$
 $\text{NULL}(\text{PRODAJA}, \text{sif_proizvoda}) = \perp$

SKLADISTE({sif_skladista, kontakt_tel, lokacija}, {sif_skladista})

STANJE_SKLADISTE({sif_skladista,sif_proizvoda,kolicina}, {sif_skladista+sif_proizvoda})
 $\text{JE_NA_STANJU}[\text{sif_skladista}] \subseteq \text{SKLADISTE}[\text{sif_skladista}]$
 $\text{JE_NA_STANJU}[\text{sif_proizvoda}] \subseteq \text{PROIZVOD}[\text{sif_proizvoda}]$

PORUDZBINA($\{sif_porudzbine, datum, aktivna, sif_skladista, sif_proizvoda\}$,
 $\{sif_porudzbine\}$)
 $PORUDZBINA[sif_skladista + sif_proizvoda] \subseteq STANJE_SKLADISTE[sif_skladista + sif_pr$
 $oizvoda]$
 $NULL(PORUDZBINA, sif_skladista + sif_proizvoda) = \perp$

$PORUDZBINA[sif_z] \subseteq PRODAVAC[sif_z]$
 $NULL(PORUDZBINA, sif_z) = \perp$

OBRADJUJE_IZ_SKLADISTA($\{sif_z, sif_porudzbine, datum\}$, $\{sif_porudzbine\}$)
 $OBRADJUJE_IZ_SKLADISTA[sif_z] \subseteq SKLADISTAR[sif_z]$
 $NULL(OBRADJUJE_IZ_SKLADISTA, sif_z) = \perp$

$OBRADJUJE_IZ_SKLADISTA[sif_proudzbine] \subseteq PORUDZBINA[sif_porudzbine]$
 $NULL(OBRADJUJE_IZ_SKLADISTA, sif_proudzbine) = \perp$

TRANSPORTUJE_IZ_SKLADISTA($\{vozac, skladistar, sif_z, sif_porudzbine, sif_skladista,$
 $sif_proizvoda, status, sif_prodavnice\}$, $\{vozac + skladistar + sif_z + sif_porudzbine +$
 $sif_skladista + sif_proizvoda\}$)

$TRANSPORTUJE_IZ_SKLADISTA[vozac] \subseteq VOZAC[sif_z]$
 $dom(vozac) \subseteq dom(VOZAC, sif_z)$

$TRANSPORTUJE_IZ_SKLADISTA[sif_prodavnice] \subseteq PRODAVNICA[sif_prodavnice]$
 $Null(TRANSPORTUJE_IZ_SKLADISTA, sif_prodavnice) = \perp$

$TRANSPORTUJE_IZ_SKLADISTA[sif_porudzbine] \subseteq$
 $OBRADJUJE_IZ_SKLADISTA[sif_porudzbine]$

ZAHTEV($\{sif, zahtev, sif_z, sif_proizvoda, datum, kolicina\}$, $\{sif_zahtev\}$)
 $ZAHTEV[sif_z] \subseteq PRODAVAC[sif_z]$
 $Null(ZAHTEV, sif_z) = \perp$

$ZAHTEV[sif_proizvoda] \subseteq PROIZVOD [sif_proizvoda]$
 $Null(ZAHTEV, sif_proizvoda) = \perp$

OBRADJUJE_IZ_PRODavnICE($\{sif_zahtev, sif_z, sif_proizvoda, obradio\}$, $\{sif_zahteva\}$)
 $OBRADJUJE_IZ_PRODavnICE[sif_z] \subseteq PRODAVAC[sif_z]$
 $Null(OBRADJUJE_IZ_PRODavnICE, sif_z) = \perp$

$OBRADJUJE_IZ_PRODavnICE[sif_zahtev] \subseteq ZAHTEV[sif_zahtev]$
 $Null(OBRADJUJE_IZ_PRODavnICE, sif_zahtev) = \perp$

TRANSPORTUJE($\{status, sif_prodavnice, sif_zahteva\}$, $\{sif_zahtev\}$)
 $TRANSPORTUJE[sif_z] \subseteq VOZAC[sif_z]$
 $Null(TRANSPORTUJE, sif_z) = \perp$

$TRANSPORTUJE[sif_zahtev] \subseteq OBRADJUJE_IZ_PRODavnICE[sif_zahtev]$
 $Null(TRANSPORTUJE, sif_zahteva) = \perp$

$TRANSPORTUJE[sif_prodavnice] \subseteq PRODAVNICA[sif_prodavnice]$
 $Null(TRANSPORTUJE, sif_prodavnice) = \perp$

5. DDL

```
--Sema Radnik
IF SCHEMA_ID('Radnik') IS NOT NULL
    DROP SCHEMA Radnik;
GO
CREATE SCHEMA Radnik
GO;

-- Sekvecna za tabelu Zaposleni
IF OBJECT_ID('Radnik.SeqZaposleni') IS NOT NULL
    DROP SEQUENCE Radnik.SeqZaposleni;

create sequence Radnik.SeqZaposleni as int
start with 1
increment by 1
go

--Tabela Zapoosleni
IF OBJECT_ID('Radnik.Zaposleni') IS NOT NULL
BEGIN
    DROP TABLE Radnik.Zaposleni;
END
GO
create table Radnik.Zaposleni(
    SIF_Z int not null,
    IME varchar(20) not null,
    PREZIME varchar(20) not null,
    JMBG char(13) unique not null,
    ADRESA varchar(100) not null,
    KONTAKT_TELEFON varchar(20) not null,
    MAIL varchar (50),
    PLATA money not null check(PLATA>=0),
    RADNO_MESTO varchar(80) not null,
    SEF int,
    constraint FK_Zapolseni_Sef foreign key (SEF) references Radnik.Zaposleni(SIF_Z),
    constraint PK_SIF_Z primary key (SIF_Z)
);

--Tabela Vozac
IF OBJECT_ID('Radnik.Vozac') is not null
BEGIN
    DROP TABLE Radnik.Vozac;
END
GO

create table Radnik.Vozac(
    SIF_Z int not null,
    kategorija nvarchar(2) not null check(KATEGORIJA like 'C1' or KATEGORIJA like
'D1'),
    BR_VOZ_DOZVOLE numeric(9) not null unique,
    constraint PK_Radnik_Vozac primary key (SIF_Z),
    constraint FK_Vozac_Radnik_Zaposleni foreign key(SIF_Z) references
Radnik.Zaposleni(SIF_Z)
);
```

```

--Tabela Skladistar
IF OBJECT_ID('Radnik.Skladistar') is not null
BEGIN
    DROP TABLE Radnik.Skladistar;
END
GO

create table Radnik.Skladistar(
    SIF_Z int not null,
    KATEGORIJA char check(KATEGORIJA='F'),
    SIF_SKLADISTA int not null,
    constraint PK_Radnik_Skladistar primary key (SIF_Z),
    constraint FK_Skladistar_Radnik_Zaposleni foreign key (SIF_Z) references
Radnik.Zaposleni(SIF_Z)
);

--Tabela Prodavac
IF (OBJECT_ID('Radnik.Prodavac') is not null)
BEGIN
    DROP TABLE Radnik.Prodavac;
END
GO

create table Radnik.Prodavac(
    SIF_Z int not null,
    SIF_PRODAVNICE int not null,
    BR_KARTICE numeric(6,0) unique,
    constraint PK_Radnik_Prodavac primary key(SIF_Z),
    constraint FK_Prodavac_Radnik_Zaposleni foreign key (SIF_Z) references
Radnik.Zaposleni(SIF_Z),
    constraint FK_Prodavac_Radnik_Prodavnica foreign key (SIF_PRODAVNICE) references
Proizvod_u_Prodavnicima.Prodavnica(SIF_PRODAVNICE),
);

--Tabela Nadredjeni
IF OBJECT_ID('Radnik.Nadredjeni') is not null
BEGIN
    DROP TABLE Radnik.Nadredjeni;
END
GO

create table Radnik.Nadredjeni(
    SIF_Z int not null,
    SIF_REGION int,
    KATEGORIJA varchar(10),
    constraint PK_Nadredjeni primary key (SIF_Z),
    constraint FK_Nadredjeni_Radnik_Zaposleni foreign key (SIF_Z) references
Radnik.Zaposleni(SIF_Z),
    constraint FK_Nadredjeni_Radnik_Region foreign key (SIF_REGION) references
Proizvod_u_Prodavnicima.Region(SIF_REGION)
);

--Sema Proizvod_u_Prodavnicima
IF SCHEMA_ID('Proizvod_u_Prodavnicima') IS NOT NULL
    DROP SCHEMA Proizvod_u_Prodavnicima;
GO

CREATE SCHEMA Proizvod_u_Prodavnicima;
GO

```

```

-- sekvecna za tabelu Prodavnica
IF OBJECT_ID('Proizvod_u_Prodavnici.SeqProdavnica') IS NOT NULL
    DROP SEQUENCE Proizvod_u_Prodavnici.SeqProdavnica;

create sequence Proizvod_u_Prodavnici.SeqProdavnica as int
start with 1
increment by 1

go

--Tabela Prodavnica
IF OBJECT_ID('Proizvod_u_Prodavnici.Prodavnica') IS NOT NULL
BEGIN
    DROP TABLE Proizvod_u_Prodavnici.Prodavnica;
END
GO

create table Proizvod_u_Prodavnici.Prodavnica(
    SIF_PRODAVNICE int not null,
    SIF_GRADA int not null,
    ADRESA varchar(100) not null,
    KONTAKT_TELEFON varchar(20),
    POVRSINA numeric(6,2) not null check(POVRSINA>=0),
    RADNOVREME_RD varchar(12),
    RADNOVREME_SUBOTA varchar(12),
    RADNOVREME_NEDELJA varchar(12),
    ARANZER int null,
    ADMINISTRATOR int null,
    MENADZER int null,
    constraint PK_Sif_Prodavnice primary key (SIF_PRODAVNICE),
    constraint FK_Prodavnica_PuP_NadredjeniAranzer foreign key(ARANZER) references
Radnik.Nadredjeni(SIF_Z),
    constraint FK_Prodavnica_PuP_NadredjeniAdministrator foreign key(ADMINISTRATOR)
references Radnik.Nadredjeni(SIF_Z),
    constraint FK_Prodavnica_PuP_NadredjeniMenadzer foreign key(MENADZER) references
Radnik.Nadredjeni(SIF_Z),
    constraint FK_Prodavina_PuP_Grad foreign key (SIF_GRADA) references
Proizvod_u_Prodavnici.Grad(SIF_GRAD)
);

-- sekvecna za tabelu Region
IF OBJECT_ID('Proizvod_u_Prodavnici.SeqRegion') IS NOT NULL
    DROP SEQUENCE Proizvod_u_Prodavnici.SeqRegion;

create sequence Proizvod_u_Prodavnici.SeqRegion as int
start with 1
increment by 1

go

```



```

--Tabela Region
IF OBJECT_ID('Proizvod_u_Prodavnici.Region') IS NOT NULL
BEGIN
    DROP TABLE Proizvod_u_Prodavnici.Region;
END
GO

create table Proizvod_u_Prodavnici.Region(
    SIF_REGION int not null,
    NAZIV varchar(40),
    constraint PK_SIF_REGIONA primary key (SIF_REGION)
);

-- sekvencna za tabelu GRAD
IF OBJECT_ID('Proizvod_u_Prodavnici.SeqGrad') IS NOT NULL
    DROP SEQUENCE Proizvod_u_Prodavnici.SeqGrad;

create sequence Proizvod_u_Prodavnici.SeqGrad as int
    start with 1
    increment by 1
go

--Tabela Grad
IF OBJECT_ID('Proizvod_u_Prodavnici.Grad') IS NOT NULL
BEGIN
    DROP TABLE Proizvod_u_Prodavnici.Grad;
END
GO

create table Proizvod_u_Prodavnici.Grad(
    SIF_GRAD int not null,
    NAZIV varchar(40),
    SIF_REGION int not null,
    constraint PK_SIF_GRAD primary key (SIF_GRAD),
    constraint FK_GRAD_PuP_REGION foreign key (SIF_REGION) references
Proizvod_u_Prodavnici.Region(SIF_REGION)
);

-- sekvencna za tabelu Proizvod
IF OBJECT_ID('Proizvod_u_Prodavnici.SeqProizvod') IS NOT NULL
    DROP SEQUENCE Proizvod_u_Prodavnici.SeqProizvod;

create sequence Proizvod_u_Prodavnici.SeqProizvod as int
    start with 1
    increment by 1
go

```

```

--Tabela Proizvod
IF OBJECT_ID('Proizvod_u_Prodavnici.Proizvod') IS NOT NULL
BEGIN
    DROP TABLE Proizvod_u_Prodavnici.Proizvod;
END
go

create table Proizvod_u_Prodavnici.Proizvod(
    SIF_PROIZVODA int not null,
    NAZIV_PROIZVODA varchar(30) not null,
    OPIS varchar(100),
    OSNOVNA_CENA numeric(10,2) not null check(OSNOVNA_CENA>=0),
    PRODAJNA_CENA numeric(10,2) not null check(PRODAJNA_CENA>=0),
    KOMERCIJALNA_CENA numeric(10,2) not null check(KOMERCIJALNA_CENA>=0),
    KATEGORIJA varchar(50) not null,
    VRSTA varchar(50) not null,
    constraint PK_Proizvod primary key (SIF_PROIZVODA)
);

--Tabela Magacin
IF OBJECT_ID('Proizvod_u_Prodavnici.Magacin') IS NOT NULL
BEGIN
    DROP TABLE Proizvod_u_Prodavnici.Magacin;
END
GO

create table Proizvod_u_Prodavnici.Magacin(
    SIF_MAGACINA int not null,
    SIF_PRODAVNICE int not null unique,
    constraint PK_Magacin primary key (SIF_MAGACINA,SIF_PRODAVNICE),
    constraint FK_Magacin_PuP_Prodavnica foreign key (SIF_PRODAVNICE) references
Proizvod_u_Prodavnici.Prodavnica(SIF_PRODAVNICE)
);

--Tabela nastala izmedju Poizvod i Magacin
IF OBJECT_ID('Proizvod_u_Prodavnici.ProizMag') IS NOT NULL
BEGIN
    DROP TABLE Proizvod_u_Prodavnici.ProizMag;
END
GO

create table Proizvod_u_Prodavnici.ProizMag(
    SIF_MAGACINA int not null,
    SIF_PRODAVNICE int not null,
    SIF_PROIZVODA int not null,
    KOLICINA numeric(6,0) default(0) check(KOLICINA >=0),
    constraint PK_ProdMag primary key (SIF_MAGACINA,SIF_PRODAVNICE,SIF_PROIZVODA),
    constraint FK_ProdMag_PuP_Magacin foreign key (SIF_MAGACINA,SIF_PRODAVNICE)
references Proizvod_u_Prodavnici.Magacin(SIF_MAGACINA,SIF_PRODAVNICE),
    constraint FK_ProdMag_PuP_Proizvod foreign key (SIF_PROIZVODA) references
Proizvod_u_Prodavnici.Proizvod(SIF_PROIZVODA)
);

```

```

--Tabela nastala izmedju Proizvod i Magacin
IF OBJECT_ID('Proizvod_u_Prodavnici.ProizProd') IS NOT NULL
BEGIN
    DROP TABLE Proizvod_u_Prodavnici.ProizProd;
END
GO

create table Proizvod_u_Prodavnici.ProizProd(
    SIF_PROIZVODA int not null,
    SIF_PRODAVNICE int not null,
    KOLICINA numeric(6,0) default(0) check(KOLICINA >=0),
    constraint PK_ProizProd primary key (SIF_PRODAVNICE,SIF_PROIZVODA),
    constraint FK_ProizPro_PuP_Proizvod foreign key (SIF_PROIZVODA) references
Proizvod_u_Prodavnici.Proizvod(SIF_PROIZVODA),
    constraint FK_ProizPro_PuP_Prodavnica foreign key (SIF_PRODAVNICE) references
Proizvod_u_Prodavnici.Prodavnica(SIF_PRODAVNICE)
);

-- sekvecna za tabelu Slika
IF OBJECT_ID('Proizvod_u_Prodavnici.SeqSlika') IS NOT NULL
    DROP SEQUENCE Proizvod_u_Prodavnici.SeqSlika;

create sequence Proizvod_u_Prodavnici.SeqSlika as int
start with 1
increment by 1
go

--Tabela Slika
IF OBJECT_ID('Proizvod_u_Prodavnici.Slika') IS NOT NULL
BEGIN
    DROP TABLE Proizvod_u_Prodavnici.Slika;
END
GO

create table Proizvod_u_Prodavnici.Slika(
    SIF_SLIKA int not null,
    PUTANJA_FAJLA varchar(500) not null,
    constraint PK_Sif_Slike primary key (SIF_SLIKA)
);

--Tabela nastala izmedju Slika i Proizvod
IF OBJECT_ID('Proizvod_u_Prodavnici.Predstavlja') IS NOT NULL
BEGIN
    DROP TABLE Proizvod_u_Prodavnici.Predstavlja;
END
GO

create table Proizvod_u_Prodavnici.Predstavlja(
    SIF_PROIZVODA int not null,
    SIF_SLIKA int not null,
    constraint PK_Predstavlja primary key (SIF_PROIZVODA,SIF_SLIKA),
    constraint FK_Predstavlja_PuP_Proizvod foreign key (SIF_PROIZVODA) references
Proizvod_u_Prodavnici.Proizvod(SIF_PROIZVODA),
    constraint FK_Predstavlja_PuP_Slika foreign key (SIF_SLIKA) references
Proizvod_u_Prodavnici.Slika(SIF_SLIKA)
);

```

```

-- sekvecna za tabelu Akcija
IF OBJECT_ID('Proizvod_u_Prodavnici.SeqAkcija') IS NOT NULL
    DROP SEQUENCE Proizvod_u_Prodavnici.SeqAkcija;

create sequence Proizvod_u_Prodavnici.SeqAkcija as int
start with 1
increment by 1
go

--Tabela Akcija
IF OBJECT_ID('Proizvod_u_Prodavnici.Akcija') IS NOT NULL
BEGIN
    DROP TABLE Proizvod_u_Prodavnici.Akcija;
END
GO

create table Proizvod_u_Prodavnici.Akcija(
    SIF_AKCIJE int not null,
    DATUM_POCKETKA date not null,
    DATUM_ZAVRSETKA date not null,
    PROCENAT numeric(4,2) not null check(PROCENAT > 0),
    constraint PK_Akcija primary key (SIF_AKCIJE)
);

--Tabela nastala izmedju Proizvod i Akcija
IF OBJECT_ID('Proizvod_u_Prodavnici.ProizAkc') IS NOT NULL
BEGIN
    DROP TABLE Proizvod_u_Prodavnici.ProizAkc;
END
GO

create table Proizvod_u_Prodavnici.ProizAkc(
    SIF_AKCIJE int not null,
    SIF_PROIZVODA int not null,
    CENA numeric(10,2) not null check(CENA>=0),
    constraint PK_ProizAkc primary key(SIF_AKCIJE,SIF_PROIZVODA),
    constraint FK_ProizAkc_PuP_Akcija foreign key(SIF_AKCIJE) references
Proizvod_u_Prodavnici.Akcija(SIF_AKCIJE),
    constraint FK_ProizAkc_PuP_Proizvod foreign key (SIF_PROIZVODA) references
Proizvod_u_Prodavnici.Proizvod(SIF_PROIZVODA)
);

--Sema Online_Porudzbina
IF SCHEMA_ID('Online_Porudzbina') IS NOT NULL
    DROP SCHEMA Online_Porudzbina;
GO
CREATE SCHEMA Online_Porudzbina;
GO

```

```

-- sekvecna za tabelu OnlineProizvod
IF OBJECT_ID('Online_Porudzbina.SeqOnlineProizvod') IS NOT NULL
    DROP SEQUENCE Online_Porudzbina.SeqOnlineProizvod;

create sequence Online_Porudzbina.SeqOnlineProizvod as int
start with 1
increment by 1
go

--Tabla OnlineProizvod
IF OBJECT_ID('Online_Porudzbina.OnlineProizvod')IS NOT NULL
    DROP TABLE Online_Porudzbina.OnlineProizvod;
GO
create table Online_Porudzbina.OnlineProizvod(
    SIF_ONLINE_PROIZVODA int not null,
    SIF_PROIZVODA int not null unique,
    DOSTUPNO varchar(2) not null check(DOSTUPNO like 'DA' or DOSTUPNO like 'NE'),
    ONLINE_CENA numeric(10,2) not null check(ONLINE_CENA>=0),
    constraint PK_OnlineProizvod primary key (SIF_ONLINE_PROIZVODA),
    constraint FK_OnlineProizvod_PuP_Proizvod foreign key (SIF_PROIZVODA) references
Proizvod_u_Prodavnici.Proizvod(SIF_PROIZVODA)
);

-- sekvecna za tabelu OnlineKupac
IF OBJECT_ID('Online_Porudzbina.SeqOnlineKupac') IS NOT NULL
    DROP SEQUENCE Online_Porudzbina.SeqOnlineKupac;

create sequence Online_Porudzbina.SeqOnlineKupac as int
start with 1
increment by 1
go

--Tabela OnlineKupac
IF OBJECT_ID('Online_Porudzbina.Online_Kupac')is not null
    drop table Online_Porudzbina.Online_Kupac;
go

create table Online_Porudzbina.Online_Kupac(
    SIF_KUPCA int not null,
    IME varchar(20) not null,
    PREZIME varchar(20) not null,
    EMAIL varchar (100) not null unique,
    KONTAKT varchar(12) not null unique,
    constraint PK_OnlineKupac primary key (SIF_KUPCA)
);

-- sekvecna za tabelu Porudzbjenica
IF OBJECT_ID('Online_Porudzbina.SeqPorudzbjenica') IS NOT NULL
    DROP SEQUENCE Online_Porudzbina.SeqPorudzbjenica;

create sequence Online_Porudzbina.SeqPorudzbjenica as int
start with 1
increment by 1
go

```

```

--Tabela Porudzbena
IF OBJECT_ID('Online_Porudzbina.Porudzbena') IS NOT NULL
    DROP TABLE Online_Porudzbina.Porudzbena;
GO

create table Online_Porudzbina.Porudzbena(
    SIF_PORUDZBENICE int not null,
    SIF_PRODAVNICE int not null,
    SIF_KUPCA int not null,
    SIF_ONLINE_PROIZVODA int not null,
    KOLICINA int not null check(KOLICINA>0),
    AKTIVNA varchar(2) not null check(AKTIVNA like 'DA' or AKTIVNA like 'NE'),
    constraint PK_Porudzbena primary key (SIF_PORUDZBENICE),
    constraint FK_Porudzbena_PuP_Prodavnica foreign key (SIF_PRODAVNICE) references
Proizvod_u_Prodavnici.Prodavnica(SIF_PRODAVNICE),
    constraint FK_Porudzbena_PUP_Kupac foreign key (SIF_KUPCA) references
Online_Porudzbina.Online_Kupac(SIF_KUPCA),
    constraint FK_Porudzbena_PUP_OnlineProizvod foreign key (SIF_ONLINE_PROIZVODA)
references Online_Porudzbina.OnlineProizvod(SIF_ONLINE_PROIZVODA)
);

--Sema Prevoz
IF SCHEMA_ID('Prevoz') IS NOT NULL
    DROP SCHEMA Prevoz;
GO
CREATE SCHEMA Prevoz;
GO

-- sekvecna za tabelu PrevoznoSredstvo
IF OBJECT_ID('Prevoz.SeqPrevoznoSredstvo') IS NOT NULL
    DROP SEQUENCE Prevoz.SeqPrevoznoSredstvo;

create sequence Prevoz.SeqPrevoznoSredstvo as int
start with 1
increment by 1
go

--Tabela Prevozno_Sredstvo
IF OBJECT_ID('Prevoz.Prevozno_Sredstvo') IS NOT NULL
    DROP TABLE Prevoz.Prevozno_Sredstvo;
GO

create table Prevoz.Prevozno_Sredstvo(
    SIF_VOZILA int not null,
    MODEL varchar(30) not null,
    BR_REG_TABLICE varchar(12) not null unique,
    DATUM_ISTEKA_REG date not null,
    DATUM_ZADUZIVANJA date not null,
    DATUM_RAZDUZIVANJA date not null,
    constraint PK_Prevozno_Sredstvo primary key (SIF_VOZILA)
);

```

```

--Tabela Civilno_Vozilo
IF OBJECT_ID('Prevoz.Civilno_Vozilo') IS NOT NULL
    DROP TABLE Prevoz.Civilno_Vozilo;
GO

create table Prevoz.Civilno_Vozilo(
    SIF_VOZILA int not null,
    SIF_Z int not null unique,
    constraint PK_Civilno_Vozilo primary key (SIF_VOZILA),
    constraint FK_Civilno_Vozilo_Ranik_Nadredjeni foreign key (SIF_Z) references
Radnik.Nadredjeni(SIF_Z),
    constraint FK_Civilno_Vozilo_P_Prevozno_Sredstvo foreign key (SIF_VOZILA)
references Prevoz.Prevozno_Sredstvo(SIF_VOZILA)
);

--Tabela Teretno_Vozilo
IF OBJECT_ID('Prevoz.Teretno_Vozilo') IS NOT NULL
    DROP TABLE Prevoz.Teretno_Vozilo;
GO

create table Prevoz.Teretno_Vozilo(
    SIF_VOZILA int not null,
    MARKA varchar(20) not null,
    VOZAC integer not null unique,
    constraint PK_Teretno_Vozilo primary key (SIF_VOZILA),
    constraint FK_Ter_Voz_Pr_Prevozno_Sredstvo foreign key (SIF_VOZILA) references
Prevoz.Prevozno_Sredstvo(SIF_VOZILA),
    constraint FK_Ter_Voz_Pr_Radnik_Vozac foreign key (VOZAC) references
Radnik.Vozac(SIF_Z)
);

--Sema Izrada_Plana
IF SCHEMA_ID('Izrada_Plana') IS NOT NULL
    DROP SCHEMA Izrada_Plana;
GO
CREATE SCHEMA Izrada_Plana;
GO

-- sekvencna za tabelu Plan
IF OBJECT_ID('Izrada_Plana.SeqPlan') IS NOT NULL
    DROP SEQUENCE Izrada_Plana.SeqPlan;

create sequence Izrada_Plana.SeqPlan as int
start with 1
increment by 1
go

SET QUOTED_IDENTIFIER ON;

```

```

--Tabela Plan
IF OBJECT_ID('Izrada_Plana."Plan"') IS NOT NULL
    DROP TABLE Izrada_Plana."Plan";
GO

create table Izrada_Plana."Plan"(
    SIF_PLANA int not null,
    SIF_KREATORA int not null,
    DATUM date not null,
    NEDELJA int not null check(NEDELJA>0),
    DAN_U_NEDELJI varchar(30),
    OPIS_PLANA varchar(1000) not null,
    constraint PK_Plan primary key(SIF_PLANA),
    constraint FK_Plan_Izrada_Plana_Nadredjeni foreign key (SIF_KREATORA) references
Radnik.Nadredjeni(SIF_Z)
);

-- sekvecna za tabelu Izvestaj
IF OBJECT_ID('Izrada_Plana.SeqIzvestaj') IS NOT NULL
    DROP SEQUENCE Izrada_Plana.SeqIzvestaj;

create sequence Izrada_Plana.SeqIzvestaj as int
start with 1
increment by 1
go

--Tabela Izvestaj
IF OBJECT_ID('Izrada_Plana.Izvestaj') IS NOT NULL
    DROP TABLE Izrada_Plana.Izvestaj;
GO
create table Izrada_Plana.Izvestaj(
    SIF_IZVESTAJ int not null,
    SIF_PLANA int not null,
    DATUM date not null,
    NEDELJA int not null check(NEDELJA>0),
    DAN_U_NEDELJI varchar(30),
    OPIS_IZVESTAJA varchar(1000) not null,
    constraint PK_Izvestaj primary key (SIF_IZVESTAJ),
    constraint FK_Izvestaj_IzradaPlana_Plan foreign key (SIF_PLANA) references
Izrada_Plana."Plan"(SIF_PLANA),
);

--Sema Transpor_Prodavnica_Prodavnica
IF SCHEMA_ID('Transpor_Prodavnica_Prodavnica') IS NOT NULL
    DROP SCHEMA Transpor_Prodavnica_Prodavnica;
GO
CREATE SCHEMA Transpor_Prodavnica_Prodavnica;
GO

```



```

-- sekvecna za tabelu Zahtev
IF OBJECT_ID('Transpor_Prodavnica_Prodavnica.SeqZahtev') IS NOT NULL
    DROP SEQUENCE Transpor_Prodavnica_Prodavnica.SeqZahtev;

create sequence Transpor_Prodavnica_Prodavnica.SeqZahtev as int
start with 1
increment by 1
go

--Tabela Zahtev
IF OBJECT_ID('Transpor_Prodavnica_Prodavnica.Zahtev') IS NOT NULL
    DROP TABLE Transpor_Prodavnica_Prodavnica.Zahtev;
GO
create table Transpor_Prodavnica_Prodavnica.Zahtev(
    SIF_ZAHTEVA int not null,
    SIF_Z int not null,
    SIF_PROIZVODA int not null,
    SIF_PRODAVNICE int not null,
    KOLICINA numeric(10,0) not null check(KOLICINA>0),
    AKTIVAN varchar(2) not null check (AKTIVAN like 'DA' or AKTIVAN like 'NE'),
    DATUM_KREIRANJA date not null,
    constraint PK_Zahtev primary key(SIF_ZAHTEVA),
    constraint FK_Zahtev_TPP_Prodavac foreign key (SIF_Z) references
Radnik.Prodavac(SIF_Z),
    constraint FK_Zahtev_TPP_ProizProd foreign key (SIF_PRODAVNICE,SIF_PROIZVODA)
references Proizvod_u_Prodavnici.ProizProd(SIF_PRODAVNICE,SIF_PROIZVODA)
);

--Tabela nastala izmedju Prodavac i Zahtev
IF OBJECT_ID('Transpor_Prodavnica_Prodavnica.ObradjujeIzProdavnice') IS NOT NULL
    DROP TABLE Transpor_Prodavnica_Prodavnica.ObradjujeIzProdavnice;
GO
create table Transpor_Prodavnica_Prodavnica.ObradjujeIzProdavnice(
    SIF_Z int not null,
    SIF_ZAHTEVA int not null unique,
    DATUM_OBRADE date not null,
    constraint PK_ObradjujeIzProdavnice primary key(SIF_ZAHTEVA),
    constraint FK_ObradjujeIzProdavnice_TPP_Zahtev foreign key (SIF_ZAHTEVA)
references Transpor_Prodavnica_Prodavnica.Zahtev(SIF_ZAHTEVA),
    constraint FK_ObradjujeIzProdavnice_TPP_Prodavac foreign key (SIF_Z) references
Radnik.Prodavac(SIF_Z)
);

```

```

--Tabela nastala izmedju ObradjujeIzProdavnice i Vozac
IF OBJECT_ID('Transpor_Prodavnica_Prodavnica.TransportujeIzProdavnice') IS NOT NULL
    DROP TABLE Transpor_Prodavnica_Prodavnica.TransportujeIzProdavnice;
GO
create table Transpor_Prodavnica_Prodavnica.TransportujeIzProdavnice(
    SIF_Z int not null,
    SIF_ZAHTEVA int not null unique,
    STATUS_ISPORUKE varchar(20) not null check(STATUS_ISPORUKE like 'NEISPORUCEN' or
STATUS_ISPORUKE like 'ISPORUCEN' or STATUS_ISPORUKE like 'U TRANSPORTU'),
    SIF_PRODAVNICE int not null,
    constraint PK_Transportuje primary key (SIF_ZAHTEVA),
    constraint FK_Transportuje_TPP_Vozac foreign key (SIF_Z) references
Radnik.Vozac(SIF_Z),
    constraint FK_Transportuje_TPP_Prodavnica foreign key (SIF_PRODAVNICE) references
Proizvod_u_Prodavnici.Prodavnica(SIF_PRODAVNICE),
    constraint FK_Transportuje_TPP_Zahtev foreign key (SIF_ZAHTEVA) references
Transpor_Prodavnica_Prodavnica.ObradjujeIzProdavnice(SIF_ZAHTEVA)
);

--Sema Transport_Skladiste_Prodavnica
IF SCHEMA_ID('Transport_Skladiste_Prodavnica') IS NOT NULL
begin
    DROP SCHEMA Transport_Skladiste_Prodavnica;
end
GO
CREATE SCHEMA Transport_Skladiste_Prodavnica
GO

-- sekvencna za tabelu Skladiste
IF OBJECT_ID('Transport_Skladiste_Prodavnica.SeqSkladiste') IS NOT NULL
    DROP SEQUENCE Transport_Skladiste_Prodavnica.SeqSkladiste;

create sequence Transport_Skladiste_Prodavnica.SeqSkladiste as int
start with 1
increment by 1
go

--Tabela Skladiste
IF OBJECT_ID('Transport_Skladiste_Prodavnica.Skladiste') IS NOT NULL
    DROP TABLE Transport_Skladiste_Prodavnica.Skladiste;
GO
create table Transport_Skladiste_Prodavnica.Skladiste(
    SIF_SKLADISTA int not null,
    KONTAKT_TELEFON varchar(30) not null unique,
    LOKACIJA varchar(100) not null,
    constraint PK_Skladiste primary key (SIF_SKLADISTA),
);

```

```

--Tabela nastala izmedju Skladiste i Stanje
IF OBJECT_ID('Transport_Skladiste_Prodavnica.SkladisteStanje') IS NOT NULL
    DROP TABLE Transport_Skladiste_Prodavnica.SkladisteStanje;
GO
create table Transport_Skladiste_Prodavnica.SkladisteStanje(
    SIF_SKLADISTA int not null,
    SIF_PROIZVODA int not null,
    KOLICINA numeric(5,0) default(0) check(KOLICINA>=0),
    constraint PK_SkladisteStanje primary key(SIF_SKLADISTA,SIF_PROIZVODA),
    constraint FK_SkladisteStanje_TSP_Skladiste foreign key (SIF_SKLADISTA) references
Transport_Skladiste_Prodavnica.Skladiste(SIF_SKLADISTA),
    constraint FK_SkladisteStanje_TSP_Proizvod foreign key (SIF_PROIZVODA) references
Proizvod_u_Prodavnici.Proizvod(SIF_PROIZVODA)
);

-- sekvecna za tabelu Porudzbina
IF OBJECT_ID('Transport_Skladiste_Prodavnica.SeqPorudzbina') IS NOT NULL
    DROP SEQUENCE Transport_Skladiste_Prodavnica.SeqPorudzbina;

create sequence Transport_Skladiste_Prodavnica.SeqPorudzbina as int
start with 1
increment by 1
go

--Tabela Porudzbina
IF OBJECT_ID('Transport_Skladiste_Prodavnica.Porudzbina') IS NOT NULL
    DROP TABLE Transport_Skladiste_Prodavnica.Porudzbina
GO
create table Transport_Skladiste_Prodavnica.Porudzbina(
    SIF_PORUDZBINE int not null,
    SIF_Z int not null,
    SIF_SKLADISTA int not null,
    SIF_PROIZVODA int not null,
    DATUM date not null,
    AKTIVNA varchar(2) check(AKTIVNA like 'DA' or AKTIVNA like 'NE'),
    constraint PK_Porudzbina primary key (SIF_PORUDZBINE),
    constraint FK_Porudzbina_TSP_Prodavac foreign key (SIF_Z) references
Radnik.Prodavac(SIF_Z),
    constraint FK_Porudzbina_TSP_SkladisteStanje foreign key
(SIF_SKLADISTA,SIF_PROIZVODA) references
Transport_Skladiste_Prodavnica.SkladisteStanje(SIF_SKLADISTA,SIF_PROIZVODA)
);

--Tabela nastala izmedju Porudzbina i Skladistar
IF OBJECT_ID('Transport_Skladiste_Prodavnica.ObradjujeIzSkladista') IS NOT NULL
    DROP TABLE Transport_Skladiste_Prodavnica.ObradjujeIzSkladista;
GO
create table Transport_Skladiste_Prodavnica.ObradjujeIzSkladista(
    SIF_Z int not null,
    SIF_PORUDZBINE int not null,
    VREME_OBRADE date not null,
    constraint PK_ObradjujeIzSkladista primary key (SIF_PORUDZBINE),
    constraint FK_ObradjujeIzSkladista_TSP_Skladistar foreign key (SIF_Z) references
Radnik.Skladistar(SIF_Z),
    constraint FK_ObradjujeIzSkladista_TSP_Porudzbina foreign key (SIF_PORUDZBINE)
references Transport_Skladiste_Prodavnica.Porudzbina(SIF_PORUDZBINE)
);

```

```

--Tabela nastala izmedju ObradjujeIzSkladista i Vozac
IF OBJECT_ID('Transport_Skladiste_Prodavnica.TransportujeIzSkladista') IS NOT NULL
    DROP TABLE Transport_Skladiste_Prodavnica.TransportujeIzSkladista;
GO
create table Transport_Skladiste_Prodavnica.TransportujeIzSkladista(
    SIF_Z int not null,
    SIF_PORUDZBINE int not null,
    SIF_PRODAVNICE int not null,
    STATUS_ISPORUKE varchar(20) not null check(STATUS_ISPORUKE like 'NEISPORUCEN' or
STATUS_ISPORUKE like 'ISPORUCEN' or STATUS_ISPORUKE like 'U TRANSPORTU'),
    constraint PK_TransportujeIzSkladista primary key (SIF_PORUDZBINE),
    constraint FK_TransportujeIzSkladista_TSP_ObradjujeIzSkladista foreign key
(SIF_PORUDZBINE) references
Transport_Skladiste_Prodavnica.ObradjujeIzSkladista(SIF_PORUDZBINE),
    constraint FK_TransportujeIzSkladista_TSP_Vozac foreign key (SIF_Z) references
Radnik.Vozac (SIF_Z),
    constraint FK_TransportujeIzSkladista_TSP_Prodavnica foreign key (SIF_PRODAVNICE)
references Proizvod_u_Prodavnici.Prodavnica(SIF_PRODAVNICE)
);

---Sema Bonus
IF SCHEMA_ID('Bonus') IS NOT NULL
begin
    DROP SCHEMA Bonus;
end
GO
create schema Bonus;
go

-- sekvecna za tabelu Prodaja
IF OBJECT_ID('Bonus.SeqProdaja') IS NOT NULL
    DROP SEQUENCE Bonus.SeqProdaja;

create sequence Bonus.SeqProdaja as int
start with 1
increment by 1
go

--Tabela Prodaja
IF OBJECT_ID('Bonus.Prodaja') IS NOT NULL
    DROP TABLE Bonus.Prodaja;
GO
create table Bonus.Prodaja(
    SIF_PRODAJE int not null,
    KOLICINA int not null check(KOLICINA>0),
    VREME_PRODAJE time not null,
    DATUM_PRODAJE date not null,
    BONUS_NA_PRODAJU numeric(9,2) not null check(BONUS_NA_PRODAJU>0),
    SIF_PROIZVODA int not null,
    SIF_Z int not null,
    constraint PK_Prodao primary key (SIF_PRODAJE),
    constraint FK_Prodao_Bonus_Prodavac foreign key (SIF_Z) references
Radnik.Prodavac(SIF_Z),
    constraint FK_Prodao_Bonus_Proizvod foreign key (SIF_PROIZVODA) references
Proizvod_u_Prodavnici.Proizvod(SIF_PROIZVODA)
);

```

6. DML

```
--Zaposleni
insert into
Radnik.Zaposleni(SIF_Z, IME, PREZIME, JMBG, ADRESA, KONTAKT_TELEFON, MAIL, PLATA, RADNO_MESTO, SEF
)
values
    (NEXT VALUE FOR Radnik.SeqZaposleni, 'Daniel', 'Mocko', '1309996800118', 'Vodna
26/22', '069/617-105', 'danielmocko50@gmail.com', 200000, 'Direktor', null),

    (NEXT VALUE FOR Radnik.SeqZaposleni, 'Lidija', 'Tadic', '1234567891002', 'Masarikova
164', '065/333-333', 'ldijatadic@gmail.com', 45000, 'Regionalni Direktor', 1),
    (NEXT VALUE FOR Radnik.SeqZaposleni, 'Natasa', 'Vukic', '1234567891003', 'Kralja
petra 16', '065/444-444', 'natasa12@gmail.com', 45000, 'Regionalni Direktor', 1),

    (NEXT VALUE FOR Radnik.SeqZaposleni, 'Marko', 'Petrovic', '1234567891001', 'Jovana
Ducica 5', '065/222-222', 'markopetrovic123@gmail.com', 45000, 'Regionalni Menadzer', 2),
    (NEXT VALUE FOR Radnik.SeqZaposleni, 'Jovana', 'Milanovic', '1234567890223',
'Jovanina 2', '937900987', 'jox@gmail.com', 65000, 'Regionalni Menadzer', 2),
    (NEXT VALUE FOR Radnik.SeqZaposleni, 'Aleksandra', 'Kitanovic', '1234567890304',
'Sandrina 33', '047263985', 'sakia33@gmail.com', 76000, 'Regionalni Menadzer', 3),
    (NEXT VALUE FOR Radnik.SeqZaposleni, 'Kradisa', 'Kradic', '1234567890407',
'kradisina 2', '487654', 'kradi@gmail.com', 39000, 'Regionalni Menadzer', 3),

    (NEXT VALUE FOR Radnik.SeqZaposleni, 'Petar', 'Petrovic', '1234567890258',
'Petroviceva 34', '065/453-445', 'p@gmail.com', 37000, 'Administrator', 4),
    (NEXT VALUE FOR Radnik.SeqZaposleni, 'Nenad', 'Antonic', '1234567890281', 'Stanka
Opsenice 16', '062/4897-654', 'nedzada@krstarica.com', 56000, 'Administrator', 4),
    (NEXT VALUE FOR Radnik.SeqZaposleni, 'Nesa', 'Bridzis', '1234567890275', 'Hajduk
Veljkova 17', '062/7697-657', 'bridzis@krstarica.com', 48000, 'Administrator', 5),
    (NEXT VALUE FOR Radnik.SeqZaposleni, 'Sasa', 'Kovacevic', '1234567899275', 'Vuk
Brankovica 37', '062/5897-659', 'kovac@krstarica.com', 46000, 'Administrator', 5),
    (NEXT VALUE FOR Radnik.SeqZaposleni, 'Nikolina', 'Djuric', '1234567892275',
'Zeleznicka 66', '065/3297-645', 'nilina456@krstarica.com', 50000, 'Administrator', 6),

    (NEXT VALUE FOR Radnik.SeqZaposleni, 'Igor', 'Stokic', '1234567890410', 'Boska
Buhe 3', '065/534-534', 'themasternasty@gmail.com', 36000, 'Aranzer', 7),
    (NEXT VALUE FOR Radnik.SeqZaposleni, 'Mitar', 'Rosandic', '1234567890464', 'Neka
ulica 1', '062/234-4234', 'mrosandic@gmail.com', 46000, 'Aranzer', 3),
    (NEXT VALUE FOR Radnik.SeqZaposleni, 'Radisa', 'Kostic', '1234567890418',
'Kosticeva 2', '063/423-423', 'lakosta@gmail.com', 36000, 'Aranzer', 7),
    (NEXT VALUE FOR Radnik.SeqZaposleni, 'Predrag', 'Savic', '1234567865464',
'Zanatlijska 55', '064/236-4884', 'predragkostic@gmail.com', 46000, 'Aranzer', 6),
    (NEXT VALUE FOR Radnik.SeqZaposleni, 'Mita', 'Pantic', '1234567820478', 'Mitino
Sokace 26', '063/993-243', 'mitalegenda@gmail.com', 36000, 'Aranzer', 3),

    (NEXT VALUE FOR Radnik.SeqZaposleni, 'Dejan', 'Vuckovic', '2806995840288',
'Krajiska 48', '061/2523-099', 'dejan@gmail.com', 50000, 'Menadzer Prodavac', 4),
    (NEXT VALUE FOR Radnik.SeqZaposleni, 'Ozren', 'Kolaic', '1234567890163',
'Beogradski Kej 25', '061/5345-435', 'ozrensky@gmail.com', 16000, 'Menadzer Prodavac', 4),
    (NEXT VALUE FOR Radnik.SeqZaposleni, 'Marko', 'Krivokapic', '1234567890033',
'Sutjeska 34', '064/4324-344', 'marko@gmail.com', 43000, 'Menadzer Prodavac', 5),

    (NEXT VALUE FOR Radnik.SeqZaposleni, 'Igor', 'Sember', '1234567890081', 'Boska
Buhe 3', '062/534-534', 'themasternasty@gmail.com', 36000, 'Zamenik Menadzera
Prodavca', 18),
```

```

(NEXT VALUE FOR Radnik.SeqZaposleni, 'Miroslav', 'Djuricin', '1234567890250',
'Vuka Karadzica 34', '061/0238-943', 'mikica@gmail.com', 70000, 'Zamenik Menadzera
Prodavca',19),
(NEXT VALUE FOR Radnik.SeqZaposleni, 'Zivko', 'Petlovic', '1234567890401',
'Bosanska 69', '064/534-543', 'oOoOo0djesba@gmail.com', 30000, 'Zamenik Menadzera
Prodavca',null),
(NEXT VALUE FOR Radnik.SeqZaposleni, 'Jusuf', 'Jusufovic', '1234567890079',
'Visegradska 33', '061/434-56', 'ju-suf-o@gmail.com', 30000, 'Zamenik Menadzera
Prodavca',20),

(NEXT VALUE FOR Radnik.SeqZaposleni,
'Marko', 'Markovic', '1234567891000', 'Zanatlijska 32', '065/111-
111', 'markomarkovic@gmail.com', 45000, 'Prodavac',18),
(NEXT VALUE FOR Radnik.SeqZaposleni, 'Petar', 'Petrovic', '1234567891004', 'Cara
Dusana 62', '065/555-333', 'petarperi@gmail.com', 45000, 'Prodavac',18),
(NEXT VALUE FOR Radnik.SeqZaposleni, 'Sasa', 'Djokovic', '1234567891005', 'Jovana
Cvijica 32', '065/666-555', 'nekimejl@gmail.com', 45000, 'Prodavac',19),
(NEXT VALUE FOR Radnik.SeqZaposleni,
'Andjelka', 'Danilovic', '1234567891006', 'Brace ribnikar 27', '065/777-
333', 'ankelaiandrija@gmail.com', 45000, 'Prodavac',18),
(NEXT VALUE FOR Radnik.SeqZaposleni, 'Ana', 'Pop', '1234567891007', 'Zeleznicka
54', '065/888-111', 'gospodipomiluj@gmail.com', 45000, 'Prodavac',19),
(NEXT VALUE FOR Radnik.SeqZaposleni,
'Ljubica', 'Markovic', '1234567891008', 'Proleteska 76', '065/999-
111', 'markomarkovic@gmail.com', 45000, 'Prodavac',19),
(NEXT VALUE FOR Radnik.SeqZaposleni, 'Goran', 'Kostic', '1234567890351',
'Kosticeva 5', '456785643', 'goki@gmail.com', 31000, 'Prodavac',null),
(NEXT VALUE FOR Radnik.SeqZaposleni, 'Radisa', 'Markovic', '1234567890241',
'Slepa ulica 1', '64534543', 'kdaopsd@gmail.com', 23000, 'Prodavac',20),
(NEXT VALUE FOR Radnik.SeqZaposleni, 'Milana', 'Milanovic', '1234567890023',
'Milanina 2', '765486456', 'milana@gmail.com', 23000, 'Prodavac',20),
(NEXT VALUE FOR Radnik.SeqZaposleni, 'Aleksandra', 'Aleksandrovic',
'1234567890336', 'Sandrina 2', '7654876', 'sakia@gmail.com', 20000, 'Prodavac',20),
(NEXT VALUE FOR Radnik.SeqZaposleni, 'Petar', 'Zamurovic', '1234567890247',
'Petrova 2', '1987654', 'peki@gmail.com', 27000, 'Prodavac',18),

(NEXT VALUE FOR Radnik.SeqZaposleni, 'Goran', 'Majkic', '1234567890038',
'Goranova 1', '3876543', 'gok324i@gmail.com', 90000, 'Vozac',null),
(NEXT VALUE FOR Radnik.SeqZaposleni, 'Ranko', 'Perovic', '1234567890372', 'Slepa
ulica 2', '064589472384', 'rankobre@gmail.com', 87000, 'Vozac',null),

(NEXT VALUE FOR Radnik.SeqZaposleni, 'Marijana', 'Kitic', '1234567890425',
'Marijanina 33', '3396345', 'makija3@gmail.com', 34000, 'Skladistar',null),
(NEXT VALUE FOR Radnik.SeqZaposleni, 'Dejan', 'Dejanovic', '1234567890436',
'Dejanova 33', '2222837', 'dex33@gmail.com', 29000, 'Skladistar',null),
(NEXT VALUE FOR Radnik.SeqZaposleni, 'Bojan', 'Bojanic', '1234567890453',
'Bokijeva 33', '555565654', 'box33@gmail.com', 31000, 'Skladistar',null);

--Vozac
insert into Radnik.Vozac(SIF_Z,kategorija,BR_VOZ_DOZVOLE)
values
(36, 'C1',000047856),
(37, 'D1',000078456);

```

```

--Skladiste
insert into
Transport_Skladiste_Prodavnica.Skladiste(SIF_SKLADISTA,KONTAKT_TELEFON,LOKACIJA)
values
    (NEXT VALUE FOR Transport_Skladiste_Prodavnica.SeqSkladiste,'021/682-
456','Sentandrejski put 54, Novi sad'),
    (NEXT VALUE FOR Transport_Skladiste_Prodavnica.SeqSkladiste,'021/682-
457','Konstantina Velikog bb, Beograd'),
    (NEXT VALUE FOR Transport_Skladiste_Prodavnica.SeqSkladiste,'021/682-
458','Partizanska ulica bb, Nis');

--Skladistar
insert into Radnik.Skladistar(SIF_Z,KATEGORIJA,SIF_SKLADISTA)
values
    (38,'F',1),
    (39,'F',2),
    (40,'F',3);

--Region
insert into Proizvod_u_Prodavnici.Region(SIF_REGION,NAZIV)
values
    (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqRegion,'Sremski okrug'),
    (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqRegion,'Banatski okrug'),
    (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqRegion,'Backi okrug'),
    (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqRegion,'Beogradski okrug'),
    (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqRegion,'Nišavski okrug');

--Grad
insert into Proizvod_u_Prodavnici.Grad(SIF_GRAD,NAZIV,SIF_REGION)
values
    (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqGrad,'Novi Sad',3),
    (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqGrad,'Apatin',3),
    (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqGrad,'Sremska Mitrovica',1),
    (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqGrad,'Ruma',1),
    (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqGrad,'Nis',5),
    (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqGrad,'Beograd',4),
    (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqGrad,'Zrenjanin',2);

--Nadredjeni
insert into Radnik.Nadredjeni(SIF_Z,SIF_REGION,KATEGORIJA)
values
    (1,null,'B'),
    (2,1,'B'),
    (3,null,'B'),
    (4,2,'B'),
    (5,5,'B'),
    (6,5,'B'),
    (7,4,'B'),
    (8,4,'B'),
    (9,null,'B'),
    (10,3,'B'),
    (11,1,'B'),
    (12,5,'B'),
    (13,3,'B'),
    (14,1,'B'),
    (15,null,'B'),
    (16,5,'B'),
    (17,4,'B');

```

```

--Prodavnica
insert into Proizvod_u_Prodavnici.Prodavnica
(SIF_PRODAVNICE,ADRESA,KONTAKT_TELEFON,POVRSINA,RADNOVREME_RD,RADNOVREME_SUBOTA,RADNOVREM
E_NEDELJA,SIF_GRADA,ARANZER,ADMINISTRATOR,MENADZER)
values
  (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqProdavnica,'Mise Dimitrijevic
66','022/122-001',50.2,'8:00-22:00','8:00-20:00','08:00-15:00',1,13,10,6),
  (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqProdavnica,'Kralja petra 6','022/122-
002',47,'8:00-22:00','8:00-20:00','08:00-15:00',3,14,11,2),
  (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqProdavnica,'Zeleznicka 84','022/122-
003',34,'8:00-22:00','8:00-20:00','08:00-15:00',1,13,10,6),
  (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqProdavnica,'Ive Andrica 66','022/122-
004',83,'8:00-22:00','8:00-20:00','08:00-15:00',5,16,12,5),
  (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqProdavnica,'Desanka Maksimovic
39','022/122-005',61,'8:00-22:00','8:00-20:00','08:00-15:00',5,16,12,5),
  (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqProdavnica,'Ive Andrica 66','022/122-
004',83,'8:00-22:00','8:00-20:00','08:00-15:00',6,8,17,7);

--Prodavac
insert into Radnik.Prodavac (SIF_Z,SIF_PRODAVNICE,BR_KARTICE)
values
  (18,1,110010),
  (19,2,110011),
  (20,4,110012),

  (21,1,110013),
  (22,2,110014),
  (23,3,110015),
  (24,4,110016),

  (25,1,110017),
  (26,1,110018),
  (27,2,110019),
  (29,2,110020),
  (30,3,110021),
  (31,3,110022),
  (32,4,110023),
  (33,4,110024),
  (34,5,110025),
  (35,5,110026);

--Proizvod
insert into
Proizvod_u_Prodavnici.Proizvod(SIF_PROIZVODA,NAZIV_PROIZVODA,OPIS,OSNOVNA_CENA,PRODAJNA_C
ENA,KOMERCIJALNA_CENA,KATEGORIJA,VRSTA)
values
  (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqProizvod, 'LED Monitor', 'Jos malo pa
nestalo', 13000, 16800, 20000, 'Tehnika', 'Monitor'),
  (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqProizvod, 'GTX 1080ti', 'Najjaca graficka
karta na trzistu', 99000, 110000, 125000, 'Tehnika', 'Graficke karte'),
  (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqProizvod, 'GTX 1070ti', 'Gamerska
graficka', 69000, 80000, 95000, 'Tehnika', 'Graficke karte'),
  (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqProizvod, 'GTX 950m', 'Solidna graficka
karta', 6500, 8100, 8900, 'Tehnika', 'Graficke karte'),
  (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqProizvod, 'Intel i5 6300hq', 'Solidan
procesor', 16000, 17800, 19700, 'Tehnika', 'Procesori'),
  (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqProizvod, 'Intel i7 7700hq', 'Solidan
procesor dosta', 17000, 18800, 19700, 'Tehnika', 'Procesori'),

```



```

        (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqProizvod, 'AMD Ryzen 7', 'Dobar gamerski
AMD procesor', 78000, 83000, 88000, 'Tehnika', 'Procesori'),
        (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqProizvod, 'Frizider XZIY2', 'Veliki
frizider', 6500, 7800, 8000, 'Bela tehnika', 'Frizideri'),
        (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqProizvod, 'Zamrzivac', 'Ides u djavola',
12000, 13400, 14300, 'Bela tehnika', 'Zamrzivaci'),
        (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqProizvod, 'ASUS ROG MATICNA', 'wow',
16500, 17400, 19300, 'Tehnika', 'Maticne Ploce'),
        (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqProizvod, 'Lenovo Legion', 'Gamerski
laptop', 120000, 123000, 126000, 'Tehnika', 'Laptopovi'),
        (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqProizvod, 'ASUS ROG', 'Asusov gamerski
laptop', 134000, 139000, 143000, 'Tehnika', 'Laptopovi'),
        (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqProizvod, 'HP Omen', 'wow', 100000,
103000, 104340, 'Tehnika', 'Laptopovi'),
        (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqProizvod, 'LCD Monitor', 'Dobar neki jako
ono', 6500, 7800, 8000, 'Tehnika', 'Monitor');

```

--Akcija

insert into

Proizvod_u_Prodavnici.Akcija(SIF_AKCIJE,DATUM_POCETKA,DATUM_ZAVRSETKA,PROCENAT)

values

```

        (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqAkcija, '21-JAN-18', '22-JAN-18', 7),
        (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqAkcija, '1-DEC-18', '21-JAN-19', 30),
        (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqAkcija, '2-JUN-18', '21-NOV-18', 21),
        (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqAkcija, '21-JAN-18', '21-JAN-18', 9),
        (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqAkcija, '28-JAN-18', '21-FEB-18', 50),
        (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqAkcija, '9-SEP-18', '9-SEP-18', 50);

```

--ProizAkc

insert into Proizvod_u_Prodavnici.ProizAkc(SIF_AKCIJE,SIF_PROIZVODA,CENA)

values

```

        (1, 1, 5000),
        (2, 3, 9000),
        (3, 5, 80000),
        (4, 10, 4000),
        (5, 12, 8000);

```

--Magacin

insert into Proizvod_u_Prodavnici.Magacin(SIF_MAGACINA,SIF_PRODAVNICE)

values

```

        (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqMagacin,1),
        (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqMagacin,2),
        (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqMagacin,3),
        (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqMagacin,4),
        (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqMagacin,5),
        (NEXT VALUE FOR Proizvod_u_Prodavnici.SeqMagacin,6);

```

--ProizMag

insert into

Proizvod_u_Prodavnici.ProizMag(SIF_MAGACINA,SIF_PRODAVNICE,SIF_PROIZVODA,KOLICINA)

values

```

        (1,1,1,5),
        (1,1,2,15),
        (1,1,4,7),
        (1,1,6,3),
        (1,1,8,1),
        (1,1,9,5),
        (1,1,11,8),
        (2,2,3,0),

```

```

(2,2,4,6),
(2,2,5,2),
(2,2,6,12),
(2,2,7,15),
(2,2,8,7),
(2,2,9,3),
(2,2,10,2),
(3,3,1,5),
(3,3,2,4),
(3,3,8,6),
(3,3,7,8),
(3,3,12,4),
(3,3,13,2),
(4,4,4,3),
(4,4,7,1),
(4,4,12,2),
(4,4,11,6),
(5,5,11,6),
(5,5,6,3),
(5,5,1,7),
(5,5,12,4),
(6,6,1,4);

```

--ProizProd

```

insert into Proizvod_u_Prodavnici.ProizProd(SIF_PRODAVNICE,SIF_PROIZVODA,KOLICINA)
values

```

```

(1,1,1),
(1,4,1),
(1,5,1),
(1,6,2),
(1,8,1),
(1,10,1),
(2,1,1),
(2,4,1),
(2,8,1),
(2,13,1),
(3,1,1),
(3,7,1),
(3,11,1),
(3,12,1),
(3,13,1),
(4,3,1),
(4,4,1),
(4,5,1),
(4,6,1),
(4,8,1),
(5,1,1),
(5,2,1),
(5,3,1),
(5,9,1),
(5,12,1),
(6,3,1),
(6,4,1),
(6,5,1),
(6,6,1);

```

```

--Slika
insert into Proizvod_u_Prodavnici.Slika(SIF_SLIKA,PUTANJA_FAJLA)
values
(NEXT VALUE FOR Proizvod_u_Prodavnici.SeqSlika, 'http://hhffhsdifhsdf8489f98hrfh8h9sdf'),
(NEXT VALUE FOR Proizvod_u_Prodavnici.SeqSlika, 'http://dostajebilof'),
(NEXT VALUE FOR Proizvod_u_Prodavnici.SeqSlika, 'http://mnogoslikaima'),
(NEXT VALUE FOR Proizvod_u_Prodavnici.SeqSlika, 'http://sjuuu'),
(NEXT VALUE FOR Proizvod_u_Prodavnici.SeqSlika, 'http://ualajao'),
(NEXT VALUE FOR Proizvod_u_Prodavnici.SeqSlika, 'http://dosadnojeovobas'),
(NEXT VALUE FOR Proizvod_u_Prodavnici.SeqSlika, 'http://smoriosamsevec'),
(NEXT VALUE FOR Proizvod_u_Prodavnici.SeqSlika, 'http://heheslika'),
(NEXT VALUE FOR Proizvod_u_Prodavnici.SeqSlika, 'http://evoneznambogami'),
(NEXT VALUE FOR Proizvod_u_Prodavnici.SeqSlika, 'http://nekatamoslikaproizvoda'),
(NEXT VALUE FOR Proizvod_u_Prodavnici.SeqSlika, 'http://slikaslicica');

--Predstavlja
insert into Proizvod_u_Prodavnici.Predstavlja(SIF_SLIKA, SIF_PROIZVODA)
values
(1, 1),
(2, 1),
(3, 2),
(4, 2),
(5, 3),
(6, 7),
(7, 6),
(8, 4),
(9, 11),
(10, 12),
(11, 12);

--Prevozno_Sredstvo
insert into
Prevoz.Prevozno_Sredstvo(SIF_VOZILA,MODEL,BR_REG_TABLICE,DATUM_ISTEKA_REG,DATUM_ZADUZIVAN
JA,DATUM_RAZDUZIVANJA)
values
(NEXT VALUE FOR Prevoz.SeqPrevoznoSredstvo, 'Limuzina', 'NS-145-32-1', '31-DEC-18', '23-
JAN-17', '30-JUN-19'),
(NEXT VALUE FOR Prevoz.SeqPrevoznoSredstvo, 'Kamion', 'NS-423-54-6', '1-JUN-18', '2-JAN-
16', '1-NOV-19'),
(NEXT VALUE FOR Prevoz.SeqPrevoznoSredstvo, 'Kombi', 'NS-432-51-3', '3-FEB-18', '3-OCT-
17', '21-MAR-19'),
(NEXT VALUE FOR Prevoz.SeqPrevoznoSredstvo, 'Kupe', 'KI-427-54-7', '22-JUN-19', '2-MAY-
16', '1-NOV-19'),
(NEXT VALUE FOR Prevoz.SeqPrevoznoSredstvo, 'Hecbek', 'KI-413-56-6', '4-NOV-18', '2-JUN-
18', '1-DEC-19'),
(NEXT VALUE FOR Prevoz.SeqPrevoznoSredstvo, 'Kupe', 'KI-424-54-7', '22-JUN-19', '2-MAY-
16', '1-NOV-19'),
(NEXT VALUE FOR Prevoz.SeqPrevoznoSredstvo, 'Hecbek', 'KI-413-54-6', '4-NOV-18', '2-JUN-
18', '1-DEC-19'),
(NEXT VALUE FOR Prevoz.SeqPrevoznoSredstvo, 'Kupe', 'KI-424-34-7', '22-JUN-19', '2-MAY-
16', '1-NOV-19'),
(NEXT VALUE FOR Prevoz.SeqPrevoznoSredstvo, 'Hecbek', 'KI-413-36-6', '4-NOV-18', '2-JUN-
18', '1-DEC-19');

```

```

--Teretno_Vozilo
insert into Prevoz.Teretno_Vozilo(SIF_VOZILA,MARKA,VOZAC)
values
(3,'Volkswagen',36),
(2,'Volvo',37);

--Civilno_Vozilo
insert into Prevoz.Civilno_Vozilo(SIF_VOZILA,SIF_Z)
values
(1,1),
(4,2),
(5,3),
(6,4),
(7,5),
(8,8),
(9,9);

--OnlineProizvod
insert into
Online_Porudzbina.OnlineProizvod(SIF_ONLINE_PROIZVODA,SIF_PROIZVODA,DOSTUPNO,ONLINE_CENA)
values
(NEXT VALUE FOR Online_Porudzbina.SeqOnlineProizvod, 1, 'DA', 4600),
(NEXT VALUE FOR Online_Porudzbina.SeqOnlineProizvod, 2, 'NE', 9600),
(NEXT VALUE FOR Online_Porudzbina.SeqOnlineProizvod, 3, 'DA', 90000),
(NEXT VALUE FOR Online_Porudzbina.SeqOnlineProizvod, 4, 'NE', 4600),
(NEXT VALUE FOR Online_Porudzbina.SeqOnlineProizvod, 12, 'DA', 8900),
(NEXT VALUE FOR Online_Porudzbina.SeqOnlineProizvod, 6, 'DA', 14100),
(NEXT VALUE FOR Online_Porudzbina.SeqOnlineProizvod, 7, 'NE', 4600),
(NEXT VALUE FOR Online_Porudzbina.SeqOnlineProizvod, 8, 'DA', 9600),
(NEXT VALUE FOR Online_Porudzbina.SeqOnlineProizvod, 9, 'DA', 90000),
(NEXT VALUE FOR Online_Porudzbina.SeqOnlineProizvod, 13, 'DA', 14100);

--Online_Kupac
insert into Online_Porudzbina.Online_Kupac(SIF_KUPCA,IME,PREZIME,KONTAKT,EMAIL)
values
(NEXT VALUE FOR Online_Porudzbina.SeqOnlineKupac,'Petar','Petrovic','069/152-
3756','petarp@gmail.com'),
(NEXT VALUE FOR Online_Porudzbina.SeqOnlineKupac,'Marko','Markovic','069/162-
3756','markomark34@gmail.com'),
(NEXT VALUE FOR Online_Porudzbina.SeqOnlineKupac,'Natasa','Colic','069/112-
3756','natasica99@gmail.com'),
(NEXT VALUE FOR Online_Porudzbina.SeqOnlineKupac,'Milica','Milic','069/172-
3756','micabrat74@gmail.com'),
(NEXT VALUE FOR Online_Porudzbina.SeqOnlineKupac,'Dejana','Sasisc','069/152-
376','dekicarmozda@gmail.com'),
(NEXT VALUE FOR Online_Porudzbina.SeqOnlineKupac,'Dajman','Vasic','069/179-
3756','gdemijezenenko@gmail.com');

--Porudzbjenica
insert into
Online_Porudzbina.Porudzbjenica(SIF_PORUDZBENICE,SIF_ONLINE_PROIZVODA,KOLICINA,AKTIVNA,SIF
_PRODAVNICE,SIF_KUPCA)
values
(NEXT VALUE FOR Online_Porudzbina.SeqPorudzbjenica,2,1,'DA',2,1),
(NEXT VALUE FOR Online_Porudzbina.SeqPorudzbjenica,4,1,'DA',3,2),
(NEXT VALUE FOR Online_Porudzbina.SeqPorudzbjenica,2,2,'DA',5,3),
(NEXT VALUE FOR Online_Porudzbina.SeqPorudzbjenica,3,1,'NE',2,4),
(NEXT VALUE FOR Online_Porudzbina.SeqPorudzbjenica,1,1,'NE',2,5),
(NEXT VALUE FOR Online_Porudzbina.SeqPorudzbjenica,1,1,'DA',1,6);

```

```

--Plan
insert into
Izrada_Plana."Plan"(SIF_PLANA,SIF_KREATORA,DATUM,NEDELJA,DAN_U_NEDELJI,OPIS_PLANA)
values
    (NEXT VALUE FOR Izrada_Plana.SeqPlan, 2, '2-JAN-17', DATEPART(WEEK, '2-JAN-17'), DATENAME(WEEKDAY, '2-JAN-17'), 'Otrezniti se'),
    (NEXT VALUE FOR Izrada_Plana.SeqPlan, 4, '2-JAN-17', DATEPART(WEEK, '2-JAN-17'), DATENAME(WEEKDAY, '2-JAN-17'), 'Otrezniti se'),
    (NEXT VALUE FOR Izrada_Plana.SeqPlan, 5, '2-JAN-17', DATEPART(WEEK, '2-JAN-17'), DATENAME(WEEKDAY, '2-JAN-17'), 'Otrezniti se'),
    (NEXT VALUE FOR Izrada_Plana.SeqPlan, 9, '2-JAN-17', DATEPART(WEEK, '2-JAN-17'), DATENAME(WEEKDAY, '2-JAN-17'), 'Otrezniti se'),
    (NEXT VALUE FOR Izrada_Plana.SeqPlan, 10, '2-JAN-17', DATEPART(WEEK, '2-JAN-17'), DATENAME(WEEKDAY, '2-JAN-17'), 'Otrezniti se'),
    (NEXT VALUE FOR Izrada_Plana.SeqPlan, 14, '2-JAN-17', DATEPART(WEEK, '2-JAN-17'), DATENAME(WEEKDAY, '2-JAN-17'), 'Otrezniti se'),
    (NEXT VALUE FOR Izrada_Plana.SeqPlan, 15, '2-JAN-17', DATEPART(WEEK, '2-JAN-17'), DATENAME(WEEKDAY, '2-JAN-17'), 'Otrezniti se'),

    (NEXT VALUE FOR Izrada_Plana.SeqPlan, 2, '3-JAN-17', DATEPART(WEEK, '3-JAN-17'), DATENAME(WEEKDAY, '3-JAN-17'), 'Otrezniti se od sinoc'),
    (NEXT VALUE FOR Izrada_Plana.SeqPlan, 4, '3-JAN-17', DATEPART(WEEK, '3-JAN-17'), DATENAME(WEEKDAY, '3-JAN-17'), 'Otrezniti se od sinoc'),
    (NEXT VALUE FOR Izrada_Plana.SeqPlan, 5, '3-JAN-17', DATEPART(WEEK, '3-JAN-17'), DATENAME(WEEKDAY, '3-JAN-17'), 'Otrezniti se od sinoc'),
    (NEXT VALUE FOR Izrada_Plana.SeqPlan, 9, '3-JAN-17', DATEPART(WEEK, '3-JAN-17'), DATENAME(WEEKDAY, '3-JAN-17'), 'Otrezniti se od sinoc'),
    (NEXT VALUE FOR Izrada_Plana.SeqPlan, 10, '3-JAN-17', DATEPART(WEEK, '3-JAN-17'), DATENAME(WEEKDAY, '3-JAN-17'), 'Otrezniti se od sinoc'),
    (NEXT VALUE FOR Izrada_Plana.SeqPlan, 14, '3-JAN-17', DATEPART(WEEK, '3-JAN-17'), DATENAME(WEEKDAY, '3-JAN-17'), 'Otrezniti se od sinoc'),
    (NEXT VALUE FOR Izrada_Plana.SeqPlan, 15, '3-JAN-17', DATEPART(WEEK, '3-JAN-17'), DATENAME(WEEKDAY, '3-JAN-17'), 'Otrezniti se od sinoc'),

    (NEXT VALUE FOR Izrada_Plana.SeqPlan, 2, '4-JAN-17', DATEPART(WEEK, '4-JAN-17'), DATENAME(WEEKDAY, '4-JAN-17'), 'Raditi nesto'),
    (NEXT VALUE FOR Izrada_Plana.SeqPlan, 4, '4-JAN-17', DATEPART(WEEK, '4-JAN-17'), DATENAME(WEEKDAY, '4-JAN-17'), 'Raditi nesto'),
    (NEXT VALUE FOR Izrada_Plana.SeqPlan, 5, '4-JAN-17', DATEPART(WEEK, '4-JAN-17'), DATENAME(WEEKDAY, '4-JAN-17'), 'Raditi nesto'),
    (NEXT VALUE FOR Izrada_Plana.SeqPlan, 9, '4-JAN-17', DATEPART(WEEK, '4-JAN-17'), DATENAME(WEEKDAY, '4-JAN-17'), 'Raditi nesto'),
    (NEXT VALUE FOR Izrada_Plana.SeqPlan, 10, '4-JAN-17', DATEPART(WEEK, '4-JAN-17'), DATENAME(WEEKDAY, '4-JAN-17'), 'Raditi nesto'),
    (NEXT VALUE FOR Izrada_Plana.SeqPlan, 14, '4-JAN-17', DATEPART(WEEK, '4-JAN-17'), DATENAME(WEEKDAY, '4-JAN-17'), 'Raditi nesto'),
    (NEXT VALUE FOR Izrada_Plana.SeqPlan, 15, '4-JAN-17', DATEPART(WEEK, '4-JAN-17'), DATENAME(WEEKDAY, '4-JAN-17'), 'Raditi nesto'),

    (NEXT VALUE FOR Izrada_Plana.SeqPlan, 2, '5-JAN-17', DATEPART(WEEK, '5-JAN-17'), DATENAME(WEEKDAY, '5-JAN-17'), 'Priprema pred badnji dan'),
    (NEXT VALUE FOR Izrada_Plana.SeqPlan, 4, '5-JAN-17', DATEPART(WEEK, '5-JAN-17'), DATENAME(WEEKDAY, '5-JAN-17'), 'Priprema pred badnji dan'),
    (NEXT VALUE FOR Izrada_Plana.SeqPlan, 5, '5-JAN-17', DATEPART(WEEK, '5-JAN-17'), DATENAME(WEEKDAY, '5-JAN-17'), 'Priprema pred badnji dan'),
    (NEXT VALUE FOR Izrada_Plana.SeqPlan, 9, '5-JAN-17', DATEPART(WEEK, '5-JAN-17'), DATENAME(WEEKDAY, '5-JAN-17'), 'Priprema pred badnji dan'),
    (NEXT VALUE FOR Izrada_Plana.SeqPlan, 10, '5-JAN-17', DATEPART(WEEK, '5-JAN-17'), DATENAME(WEEKDAY, '5-JAN-17'), 'Priprema pred badnji dan'),

```

```

        (NEXT VALUE FOR Izrada_Plana.SeqPlan,14,'5-JAN-17',DATEPART(WEEK,'5-JAN-17'),DATENAME(WEEKDAY,'5-JAN-17'),'Priprema pred badnji dan'),
        (NEXT VALUE FOR Izrada_Plana.SeqPlan,15,'5-JAN-17',DATEPART(WEEK,'5-JAN-17'),DATENAME(WEEKDAY,'5-JAN-17'),'Priprema pred badnji dan');

--Izvestaj
insert into
Izrada_Plana.Izvestaj(SIF_IZVESTAJ,SIF_PLANA,DATUM,NEDELJA,DAN_U_NEDELJI,OPIS_IZVESTAJA)
values
    (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,1,'2-JAN-17',DATEPART(WEEK,'2-JAN-17'),DATENAME(WEEKDAY,'2-JAN-17'),'Svi smo trezni :)'),
    (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,2,'2-JAN-17',DATEPART(WEEK,'2-JAN-17'),DATENAME(WEEKDAY,'2-JAN-17'),'Svi smo trezni :)'),
    (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,3,'2-JAN-17',DATEPART(WEEK,'2-JAN-17'),DATENAME(WEEKDAY,'2-JAN-17'),'Svi smo trezni :)'),
    (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,4,'2-JAN-17',DATEPART(WEEK,'2-JAN-17'),DATENAME(WEEKDAY,'2-JAN-17'),'Svi smo trezni :)'),
    (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,5,'2-JAN-17',DATEPART(WEEK,'2-JAN-17'),DATENAME(WEEKDAY,'2-JAN-17'),'Svi smo trezni :)'),
    (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,6,'2-JAN-17',DATEPART(WEEK,'2-JAN-17'),DATENAME(WEEKDAY,'2-JAN-17'),'Svi smo trezni :)'),
    (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,7,'2-JAN-17',DATEPART(WEEK,'2-JAN-17'),DATENAME(WEEKDAY,'2-JAN-17'),'Svi smo trezni :)'),

    (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,8,'3-JAN-17',DATEPART(WEEK,'3-JAN-17'),DATENAME(WEEKDAY,'3-JAN-17'),'E sad smo stvarno trezni :D'),
    (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,9,'3-JAN-17',DATEPART(WEEK,'3-JAN-17'),DATENAME(WEEKDAY,'3-JAN-17'),'E sad smo stvarno trezni :D'),
    (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,10,'3-JAN-17',DATEPART(WEEK,'3-JAN-17'),DATENAME(WEEKDAY,'3-JAN-17'),'E sad smo stvarno trezni :D'),
    (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,11,'3-JAN-17',DATEPART(WEEK,'3-JAN-17'),DATENAME(WEEKDAY,'3-JAN-17'),'E sad smo stvarno trezni :D'),
    (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,12,'3-JAN-17',DATEPART(WEEK,'3-JAN-17'),DATENAME(WEEKDAY,'3-JAN-17'),'E sad smo stvarno trezni :D'),
    (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,13,'3-JAN-17',DATEPART(WEEK,'3-JAN-17'),DATENAME(WEEKDAY,'3-JAN-17'),'E sad smo stvarno trezni :D'),
    (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,14,'3-JAN-17',DATEPART(WEEK,'3-JAN-17'),DATENAME(WEEKDAY,'3-JAN-17'),'E sad smo stvarno trezni :D'),

    (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,15,'4-JAN-17',DATEPART(WEEK,'4-JAN-17'),DATENAME(WEEKDAY,'4-JAN-17'),'Uradili nista'),
    (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,16,'4-JAN-17',DATEPART(WEEK,'4-JAN-17'),DATENAME(WEEKDAY,'4-JAN-17'),'Uradili nista'),
    (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,17,'4-JAN-17',DATEPART(WEEK,'4-JAN-17'),DATENAME(WEEKDAY,'4-JAN-17'),'Uradili nista'),
    (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,18,'4-JAN-17',DATEPART(WEEK,'4-JAN-17'),DATENAME(WEEKDAY,'4-JAN-17'),'Uradili nista'),
    (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,19,'4-JAN-17',DATEPART(WEEK,'4-JAN-17'),DATENAME(WEEKDAY,'4-JAN-17'),'Uradili nista'),
    (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,20,'4-JAN-17',DATEPART(WEEK,'4-JAN-17'),DATENAME(WEEKDAY,'4-JAN-17'),'Uradili nista'),
    (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,21,'4-JAN-17',DATEPART(WEEK,'4-JAN-17'),DATENAME(WEEKDAY,'4-JAN-17'),'Uradili nista'),

    (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,22,'5-JAN-17',DATEPART(WEEK,'5-JAN-17'),DATENAME(WEEKDAY,'5-JAN-17'),'Dogovorili se kad cemo po badnjak, u 5:00 ustajanje'),
    (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,23,'5-JAN-17',DATEPART(WEEK,'5-JAN-17'),DATENAME(WEEKDAY,'5-JAN-17'),'Dogovorili se kad cemo po badnjak, u 5:00 ustajanje'),

```

```

        (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,24,'5-JAN-17',DATEPART(WEEK,'5-JAN-17'),DATENAME(WEEKDAY,'5-JAN-17'),'Dogovorili se kad cemo po badnjak, u 5:00 ustajanje'),
        (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,25,'5-JAN-17',DATEPART(WEEK,'5-JAN-17'),DATENAME(WEEKDAY,'5-JAN-17'),'Dogovorili se kad cemo po badnjak, u 5:00 ustajanje'),
        (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,26,'5-JAN-17',DATEPART(WEEK,'5-JAN-17'),DATENAME(WEEKDAY,'5-JAN-17'),'Dogovorili se kad cemo po badnjak, u 5:00 ustajanje'),
        (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,27,'5-JAN-17',DATEPART(WEEK,'5-JAN-17'),DATENAME(WEEKDAY,'5-JAN-17'),'Dogovorili se kad cemo po badnjak, u 5:00 ustajanje'),
        (NEXT VALUE FOR Izrada_Plana.SeqIzvestaj,28,'5-JAN-17',DATEPART(WEEK,'5-JAN-17'),DATENAME(WEEKDAY,'5-JAN-17'),'Dogovorili se kad cemo po badnjak, u 5:00 ustajanje');

```

--Zahtev

insert into

Transpor_Prodavnica_Prodavnica.Zahtev(SIF_ZAHTEVA,SIF_Z,SIF_PRODAVNICE,SIF_PROIZVODA,KOLICINA,DATUM_KREIRANJA,AKTIVAN)

values

```

        (NEXT VALUE FOR Transpor_Prodavnica_Prodavnica.SeqZahtev,27,3,11,1,'3-MAY-17','DA'),
        (NEXT VALUE FOR Transpor_Prodavnica_Prodavnica.SeqZahtev,27,4,3,1,'3-MAY-17','DA'),
        (NEXT VALUE FOR Transpor_Prodavnica_Prodavnica.SeqZahtev,27,5,12,1,'3-MAY-17','DA'),
        (NEXT VALUE FOR Transpor_Prodavnica_Prodavnica.SeqZahtev,29,1,8,1,'14-MAY-17','DA'),
        (NEXT VALUE FOR Transpor_Prodavnica_Prodavnica.SeqZahtev,29,1,1,4,'14-MAY-17','NE'),
        (NEXT VALUE FOR Transpor_Prodavnica_Prodavnica.SeqZahtev,29,5,9,1,'14-MAY-17','DA'),
        (NEXT VALUE FOR Transpor_Prodavnica_Prodavnica.SeqZahtev,26,2,4,3,'14-MAY-17','NE'),
        (NEXT VALUE FOR Transpor_Prodavnica_Prodavnica.SeqZahtev,32,2,4,8,'22-MAY-17','NE'),
        (NEXT VALUE FOR Transpor_Prodavnica_Prodavnica.SeqZahtev,32,3,7,1,'22-MAY-17','DA'),
        (NEXT VALUE FOR Transpor_Prodavnica_Prodavnica.SeqZahtev,32,1,10,2,'22-MAY-17','NE');

```

--ObradjujeIzProdavnice

insert into

Transpor_Prodavnica_Prodavnica.ObradjujeIzProdavnice(SIF_ZAHTEVA,SIF_Z,DATUM_OBRADE)

values

```

        (5,26,'15-MAY-17'),
        (7,27,'15-MAY-17'),
        (8,29,'23-MAY-17'),
        (10,21,'23-MAY-17');

```

--TransportujeIzProdavnice

insert into

Transpor_Prodavnica_Prodavnica.TransportujeIzProdavnice(SIF_ZAHTEVA,SIF_Z,SIF_PRODAVNICE,STATUS_ISPORUKE)

values

```

        (5,36,1,'ISPORUCEN'),
        (7,36,2,'ISPORUCEN'),
        (8,37,2,'U TRANSPORTU'),
        (10,37,1,'U TRANSPORTU');

```

```

--SkladisteStanje
insert into
Transport_Skladiste_Prodavnica.SkladisteStanje(SIF_SKLADISTA,SIF_PROIZVODA,KOLICINA)
values
    (1,1,4),
    (1,2,19),
    (1,3,6),
    (1,4,4),
    (1,5,0),
    (1,6,8),
    (1,7,4),
    (1,8,23),
    (1,9,4),
    (1,10,6),
    (1,11,9),
    (1,12,7),
    (1,13,15),

    (2,1,4),
    (2,2,19),
    (2,3,6),
    (2,4,4),
    (2,5,0),
    (2,6,8),
    (2,7,4),
    (2,8,23),
    (2,9,4),
    (2,10,6),
    (2,11,9),
    (2,12,7),
    (2,13,15),

    (3,1,4),
    (3,2,19),
    (3,3,6),
    (3,4,4),
    (3,5,0),
    (3,6,8),
    (3,7,4),
    (3,8,23),
    (3,9,4),
    (3,10,6),
    (3,11,9),
    (3,12,7),
    (3,13,15);

--Porudzbina
insert into
Transport_Skladiste_Prodavnica.Porudzbina(SIF_PORUDZBINE,SIF_SKLADISTA,SIF_PROIZVODA,SIF_
Z,DATUM,AKTIVNA)
values
    (NEXT VALUE FOR Transport_Skladiste_Prodavnica.SeqPorudzbina,1,5,30,'24-JUN-
17','DA'),
    (NEXT VALUE FOR Transport_Skladiste_Prodavnica.SeqPorudzbina,1,9,33,'14-JUN-
17','DA'),
    (NEXT VALUE FOR Transport_Skladiste_Prodavnica.SeqPorudzbina,1,11,32,'14-JUN-
17','NE'),
    (NEXT VALUE FOR Transport_Skladiste_Prodavnica.SeqPorudzbina,1,13,34,'24-JUN-
17','NE'),

```



```

        (NEXT VALUE FOR Transport_Skladiste_Prodavnica.SeqPorudzbina,2,2,26,'24-JUN-17','DA'),
        (NEXT VALUE FOR Transport_Skladiste_Prodavnica.SeqPorudzbina,2,8,31,'16-JUN-17','DA'),
        (NEXT VALUE FOR Transport_Skladiste_Prodavnica.SeqPorudzbina,2,12,29,'19-JUN-17','NE'),
        (NEXT VALUE FOR Transport_Skladiste_Prodavnica.SeqPorudzbina,2,13,33,'23-JUN-17','NE'),

        (NEXT VALUE FOR Transport_Skladiste_Prodavnica.SeqPorudzbina,3,3,26,'24-JUN-17','DA'),
        (NEXT VALUE FOR Transport_Skladiste_Prodavnica.SeqPorudzbina,3,9,25,'19-JUN-17','DA'),
        (NEXT VALUE FOR Transport_Skladiste_Prodavnica.SeqPorudzbina,3,1,34,'18-JUN-17','NE'),
        (NEXT VALUE FOR Transport_Skladiste_Prodavnica.SeqPorudzbina,3,13,30,'24-JUN-17','NE');

--ObradjujeIzSkladista
insert into
Transport_Skladiste_Prodavnica.ObradjujeIzSkladista(SIF_PORUDZBINE,SIF_Z,VREME_OBRADE)
values
    (3,38,'25-JUN-17'),
    (4,38,'15-JUN-17'),
    (7,39,'25-JUN-17'),
    (8,39,'17-JUN-17'),
    (11,40,'25-JUN-17'),
    (12,40,'20-JUN-17');

--TransportujeIzSkladista
insert into
Transport_Skladiste_Prodavnica.TransportujeIzSkladista(SIF_PORUDZBINE,SIF_Z,SIF_PRODAVNICE,STATUS_ISPORUKE)
values
    (3,36,3,'ISPORUCEN'),
    (4,36,4,'ISPORUCEN'),
    (7,36,2,'ISPORUCEN'),
    (8,37,4,'ISPORUCEN'),
    (11,37,5,'U TRANSPORTU'),
    (12,37,3,'U TRANSPORTU');

--Prodaja
insert into
Bonus.Prodaja(SIF_PRODAJE,KOLICINA,VREME_PRODAJE,DATUM_PRODAJE,SIF_PROIZVODA,SIF_Z,BONUS_NA_PRODAJU)
values
    (NEXT VALUE FOR Bonus.SeqProdaja,2,cast('07:35' as time),'14-JUL-17',5,34,200),
    (NEXT VALUE FOR Bonus.SeqProdaja,1,cast('08:35' as time),'14-JUL-17',13,32,220),
    (NEXT VALUE FOR Bonus.SeqProdaja,1,cast('08:55' as time),'15-JUL-17',3,32,180),
    (NEXT VALUE FOR Bonus.SeqProdaja,1,cast('09:35' as time),'15-JUL-17',10,26,160),
    (NEXT VALUE FOR Bonus.SeqProdaja,1,cast('17:24' as time),'16-JUL-17',4,29,150),
    (NEXT VALUE FOR Bonus.SeqProdaja,1,cast('17:32' as time),'16-JUL-17',9,27,200),
    (NEXT VALUE FOR Bonus.SeqProdaja,2,cast('17:47' as time),'17-JUL-17',6,22,180),
    (NEXT VALUE FOR Bonus.SeqProdaja,1,cast('19:35' as time),'17-JUL-17',5,26,200),
    (NEXT VALUE FOR Bonus.SeqProdaja,2,cast('14:35' as time),'18-JUL-17',9,30,300),
    (NEXT VALUE FOR Bonus.SeqProdaja,2,cast('15:35' as time),'18-JUL-17',8,35,320),
    (NEXT VALUE FOR Bonus.SeqProdaja,2,cast('16:35' as time),'19-JUL-17',6,30,100);

```

7. Select upiti

/***** UPIT 1 *****/

Izlistati podatke o zaposlenom kao i broj proizvoda koji su prodali:
Sifra zaposlenog, Ime, Prezime, Plata, Broj karticem, Broj prodatih proizvoda

*****/

```
select prodaja.SIF_Z, IME, PREZIME, PLATA, BR_KARTICE, count(SIF_PRODAJE) as 'Broj prodatih  
proizvoda'  
from Radnik.Zaposleni zaposleni  
left join Radnik.Prodavac prodavac on (zaposleni.SIF_Z=prodavac.SIF_Z)  
inner join Bonus.Prodaja prodaja on (prodavac.SIF_Z= prodaja.SIF_Z)  
group by prodaja.SIF_Z, IME, PREZIME, PLATA, BR_KARTICE  
order by count(SIF_PRODAJE) desc, SIF_Z asc
```

/***** UPIT 2 *****/

Upit koji izlistava podake o prodavnicima na sledeci nacin:
Sifra prodavnice, adresa prodavnice, ime i prezime Aranzera, ime i prezime
Administradora, ime i prezime Menadzera prodavnice

*****/

```
select SIF_PRODAVNICE, prodavnica.ADRESA,  
zaposleniAranzer.IME+ ' '+zaposleniAranzer.PREZIME as 'Aranzer',  
zaposleniAdministrator.IME+ ' '+zaposleniAdministrator.PREZIME as  
'Administrator',  
zaposleniMenadzer.IME+ ' '+zaposleniMenadzer.PREZIME as 'Menadzer',  
grad.NAZIV as 'Grad',  
region.NAZIV as 'REGION'  
from Proizvod_u_Prodavnicima.Prodavnica prodavnica  
inner join Proizvod_u_Prodavnicima.Grad grad on prodavnica.SIF_GRADA=grad.SIF_GRAD  
inner join Proizvod_u_Prodavnicima.Region region on  
grad.SIF_REGION=region.SIF_REGION  
inner join Radnik.Nadredjeni administrator on  
prodavnica.ADMINISTRATOR=administrator.SIF_Z  
inner join Radnik.Nadredjeni aranzera on prodavnica.ARANZER=aranzer.SIF_Z  
inner join Radnik.Nadredjeni menadzer on prodavnica.MENADZER=menadzer.SIF_Z  
inner join Radnik.Zaposleni zaposleniAdministrator on  
administrator.SIF_Z=zaposleniAdministrator.SIF_Z  
inner join Radnik.Zaposleni zaposleniAranzer on aranzera.SIF_Z=  
zaposleniAranzer.SIF_Z  
inner join Radnik.Zaposleni zaposleniMenadzer on  
menadzer.SIF_Z=zaposleniMenadzer.SIF_Z
```

/***** UPIT 3 *****/

Upit koji izlistava podatke o proizvodu koji se trenutno nalazi na akciji i u kojoj je prodavnici on dostupan:

Sifra proizvoda, naziv proizvoda, komercijalna cena , cena na akciji, pocetak akcije, kraj akcije, sifra prodavnice, adresa prodavnice, kolicina proizvoda u prodavnici, adresa prodavnice, grad prodavnice

*****/

```
select proizvod.SIF_PROIZVODA, NAZIV_PROIZVODA, KOMERCIJALNA_CENA, proizAkc.CENA as 'Cena
akcije',
    akcija.DATUM_POCKETKA, akcija.DATUM_ZAVRSETKA, prodavnica.SIF_PRODAVNICE, proizProd.KO
LICINA,
    prodavnica.ADRESA as 'Adresa prodavnice', grad.NAZIV as 'Grad'
from Proizvod_u_Prodavnici.Akcija akcija
    inner join Proizvod_u_Prodavnici.ProizAkc proizAkc on akcija.SIF_AKCIJE=
proizAkc.SIF_AKCIJE
    inner join Proizvod_u_Prodavnici.Proizvod proizvod on proizAkc.SIF_PROIZVODA =
proizvod.SIF_PROIZVODA
    inner join Proizvod_u_Prodavnici.ProizProd proizProd on proizvod.SIF_PROIZVODA =
proizProd.SIF_PROIZVODA
    inner join Proizvod_u_Prodavnici.Prodavnica prodavnica on
proizProd.SIF_PRODAVNICE=prodavnica.SIF_PRODAVNICE
    left join Proizvod_u_Prodavnici.Grad grad on prodavnica.SIF_GRADA=grad.SIF_GRAD
where GETDATE()>=akcija.DATUM_POCKETKA and GETDATE()<=akcija.DATUM_ZAVRSETKA
```

/***** UPIT 4 *****/

Upit koji prikazuje podatke o proizvodima koji se trenutno nalaze u traspotru prema prodavnici:

Sifra prodavnice, Adresa, Grad, Naziv proizvoda, Kolicina proizvoda, ime i prezime vozaca koji trenutno transportuje proizvod, ime i prezime prodavca koji je obradio zahtev za proizvodom, ime i prezime prodavca koji je porucio odgovarajuci proizvod, Status isporuke

*****/

```
select transport.SIF_PRODAVNICE,
    prodavnicaTrans.ADRESA,
    grad.NAZIV as 'Grad',
    proizvod.NAZIV_PROIZVODA,
    zahtev.KOLICINA,
    zapsolenVozac.IME+ ' '+zapsolenVozac.PREZIME as 'Vozac',
    zaposlenObradio.IME+ ' '+zaposlenObradio.PREZIME as 'Obradio',
    zaposleniZahtevao.IME+ ' '+zaposleniZahtevao.PREZIME as 'Porucio',
    transport.STATUS_ISPORUKE

from Transpor_Prodavnica_Prodavnica.TransportujeIzProdavnice transport
    inner join Radnik.Vozac vozac on transport.SIF_Z=vozac.SIF_Z
    inner join Proizvod_u_Prodavnici.Prodavnica prodavnicaTrans on
transport.SIF_PRODAVNICE= prodavnicaTrans.SIF_PRODAVNICE
    inner join Proizvod_u_Prodavnici.Grad grad on
prodavnicaTrans.SIF_GRADA=grad.SIF_GRAD
    inner join Radnik.Zaposleni zapsolenVozac on vozac.SIF_Z=zapsolenVozac.SIF_Z
    inner join Transpor_Prodavnica_Prodavnica.ObradjujeIzProdavnice obrada on
transport.SIF_ZAHEVA=obrada.SIF_ZAHEVA
    inner join Radnik.Prodavac obradio on obrada.SIF_Z=obradio.SIF_Z
    inner join Radnik.Zaposleni zaposlenObradio on obradio.SIF_Z=zaposlenObradio.SIF_Z
```

```

        inner join Transpor_Prodavnica_Prodavnica.Zahtev zahtev on obrada.SIF_ZAHTEVA=
zahtev.SIF_ZAHTEVA
        inner join Radnik.Prodavac zahtevao on zahtev.SIF_Z=zahtevao.SIF_Z
        inner join Radnik.Zaposleni zaposleniZahtevao on
zahtevao.SIF_Z=zaposleniZahtevao.SIF_Z
        inner join Proizvod_u_Prodavnicima.Proizvod proizvod on zahtev.SIF_PROIZVODA=
proizvod.SIF_PROIZVODA
where STATUS_ISPORUKE like 'U TRANSPORTU'
order by SIF_PRODAVNICE

```

/****** UPIT 5 *****/

Upitkoji izlistava podatke o zaposlenima koji su aranzeri u bar 2 prodavnice:
Sifra zaposlenog, Ime i prezime Aranзера, broj prodavnica za koje je zaduzen

*****/

```

select zaposlen.SIF_Z, IME + ' '+PREZIME as 'Aranzer',count(SIF_PRODAVNICE) as 'Broj
prodavnica'
from Proizvod_u_Prodavnicima.Prodavnica prodavnica
        inner join Radnik.Nadredjeni aranzer on prodavnica.ARANZER=aranzer.SIF_Z
        inner join Radnik.Zaposleni zaposlen on zaposlen.SIF_Z=aranzer.SIF_Z
group by zaposlen.SIF_Z, IME,PREZIME
having count(SIF_PRODAVNICE)>1

```

8. Funkcije

```

/*****      FUNKCIJA 1      *****/

      Funkcija koja za prosledjenu sifru Aranzera,Administratora ili Menadzera ispisuje
      broj prodavnica za koji je zaduzen.

*****/

if OBJECT_ID('Funkcija_BrojProdavnicaNadredjenog','FN')is not null
    drop function Funkcija_BrojProdavnicaNadredjenog
go

create function Funkcija_BrojProdavnicaNadredjenog
(
    @sif_nadredjenog as int
)
returns int
as
begin
    declare @radnoMesto as varchar(50);
    set @radnoMesto=(select RADNO_MESTO from Radnik.Zaposleni where
SIF_Z=@sif_nadredjenog)

    declare @brojProdavnica as int;

    if(@radnoMesto like 'Aranzer')
    begin
        set @brojProdavnica=(select count(SIF_PRODAVNICE) from
Proizvod_u_Prodavnicima.Prodavnica where ARANZER=@sif_nadredjenog)
    end
    else if (@radnoMesto like 'Administrator')
    begin
        set @brojProdavnica=(select count(SIF_PRODAVNICE) from
Proizvod_u_Prodavnicima.Prodavnica where ADMINISTRATOR=@sif_nadredjenog)
    end
    else if (@radnoMesto like 'Regionalni Menadzer' or @radnoMesto like 'Regionalni
Direktor')
    begin
        set @brojProdavnica=(select count(SIF_PRODAVNICE) from
Proizvod_u_Prodavnicima.Prodavnica where MENADZER=@sif_nadredjenog)
    end

    return @brojProdavnica;
end
go
```

```

/***** TEST *****/

--Regionalni Menadzer kao Menadzer prodavnice
select dbo.Funkcija_BrojProdavnicaNadredjenog(5) as 'Broj prodavnica'

--Regionalni Direktor kao Menadzer prodavnice
select dbo.Funkcija_BrojProdavnicaNadredjenog(2) as 'Broj prodavnica'

--Aranzer
select dbo.Funkcija_BrojProdavnicaNadredjenog(2) as 'Broj prodavnica'

--Administrator
select dbo.Funkcija_BrojProdavnicaNadredjenog(14) as 'Broj prodavnica'

/***** Funkcija 2 *****/

Funkcija koja za prosledjnu sifru akcije vraca stanje akcije:
1. Akcija je prosla
2. Akcija trenutno traje
3. Akcija treba da pocne
4. Akcija ne postoji

*****/
if OBJECT_ID('Funkcija_AktivnostAkcije','FN') is not null
    drop function Funkcija_AktivnostAkcije
go

create function Funkcija_AktivnostAkcije
(
    @sifraAkcije as int
)
returns varchar(100)
as
begin
    declare @StanjeAkcije as varchar(50);
    declare @PocetakAkcije as date
    declare @KrajAkcije as date

    set @PocetakAkcije=(select DATUM_POCKETKA from Proizvod_u_Prodavnici.Akcija where
SIF_AKCIJE=@sifraAkcije)
    set @KrajAkcije=(select DATUM_ZAVRSETKA from Proizvod_u_Prodavnici.Akcija where
SIF_AKCIJE=@sifraAkcije)
    if exists (select * from Proizvod_u_Prodavnici.Akcija where
SIF_AKCIJE=@sifraAkcije)
        begin
            if (
                (DATEPART(YEAR,@KrajAkcije)<DATEPART(YEAR,GETDATE())) or
                (DATEPART(YEAR,@KrajAkcije)=DATEPART(YEAR,GETDATE()) and
DATEPART(MONTH,@KrajAkcije)<DATEPART(MONTH,GETDATE())) or
                (DATEPART(YEAR,@KrajAkcije)=DATEPART(YEAR,GETDATE()) and
DATEPART(MONTH,@KrajAkcije)=DATEPART(MONTH,GETDATE()) and
DATEPART(DAY,@KrajAkcije)<DATEPART(DAY,GETDATE()))
            )
                begin
                    set @StanjeAkcije='Akcija je prosla';
                end
            else if(
                (DATEPART(YEAR,@PocetakAkcije)>DATEPART(YEAR,GETDATE())) or

```

```

        (DATEPART(YEAR,@PocetakAkcije)=DATEPART(YEAR,GETDATE())) and
DATEPART(MONTH,@PocetakAkcije)>DATEPART(MONTH,GETDATE())) or
        (DATEPART(YEAR,@PocetakAkcije)=DATEPART(YEAR,GETDATE())) and
DATEPART(MONTH,@PocetakAkcije)=DATEPART(MONTH,GETDATE()) and
DATEPART(DAY,@PocetakAkcije)>DATEPART(DAY,GETDATE()))
    )
    begin
        set @StanjeAkcije='Akcija treba da pocne';
    end
    else if(
        (DATEPART(YEAR,@PocetakAkcije)<DATEPART(YEAR,GETDATE())) or
        (DATEPART(YEAR,@PocetakAkcije)=DATEPART(YEAR,GETDATE())) and
DATEPART(MONTH,@PocetakAkcije)<DATEPART(MONTH,GETDATE())) or
        (DATEPART(YEAR,@PocetakAkcije)=DATEPART(YEAR,GETDATE())) and
DATEPART(MONTH,@PocetakAkcije)=DATEPART(MONTH,GETDATE()) and
DATEPART(DAY,@PocetakAkcije)>=DATEPART(DAY,GETDATE()))
        and
        (DATEPART(YEAR,@KrajAkcije)>DATEPART(YEAR,GETDATE())) or
        (DATEPART(YEAR,@KrajAkcije)=DATEPART(YEAR,GETDATE())) and
DATEPART(MONTH,@KrajAkcije)>DATEPART(MONTH,GETDATE())) or
        (DATEPART(YEAR,@KrajAkcije)=DATEPART(YEAR,GETDATE())) and
DATEPART(MONTH,@KrajAkcije)=DATEPART(MONTH,GETDATE()) and
DATEPART(DAY,@KrajAkcije)>=DATEPART(DAY,GETDATE()))
    )
    begin
        set @StanjeAkcije='Akcija trenutno traje';
    end
end
else
    set @StanjeAkcije='Akcija ne postoji';
return @StanjeAkcije;
end
go

/***** TEST *****/

--Akcija je prosla
select dbo.Funkcija_AktivnostAkcije(1)

--Akcija treba da pocne
select dbo.Funkcija_AktivnostAkcije(2)

--Akcija trenutno traje
select dbo.Funkcija_AktivnostAkcije(3)

--Akcija ne postoji
select dbo.Funkcija_AktivnostAkcije(10)

```

9. Triggeri

```
/*
    Trigger koji za porudzbina koja se obradi u tabli [ObradjujeIzSkladista] setuje
    njeno obelezje [AKTIVNA]
    sa stanja ['DA'] na stanje ['NE'].
*/

if OBJECT_ID('Transport_Skladiste_Prodavnica.Trigger_ObradjujeIzSkladista','TR') is not
null
    drop trigger Transport_Skladiste_Prodavnica.Trigger_ObradjujeIzSkladista
go

create trigger Transport_Skladiste_Prodavnica.Trigger_ObradjujeIzSkladista on
Transport_Skladiste_Prodavnica.ObradjujeIzSkladista
after insert
as
begin
    declare @Sif_Porudzbine as int
    set @Sif_Porudzbine=(select SIF_PORUDZBINE from inserted)

    update Transport_Skladiste_Prodavnica.Porudzbina
    set AKTIVNA='NE'
    where SIF_PORUDZBINE=@Sif_Porudzbine
end
go

/***** TEST *****/
select * from Transport_Skladiste_Prodavnica.Porudzbina where SIF_PORUDZBINE=2

insert into
Transport_Skladiste_Prodavnica.ObradjujeIzSkladista(SIF_PORUDZBINE,SIF_Z,VREME_OBRADE)
values (2,39,'8-SEP-17')

select * from Transport_Skladiste_Prodavnica.Porudzbina where SIF_PORUDZBINE=2

/*
    Trigger koji na izmenu radnog mesta regulise platu zaposlenima ali samo onima koji
    pripadaju grupi Prodavaca a to su:
    Prodavac, Zamenik Menadzera Prodavca i Menadzer Prodavac.
    Kriterijum za regulaciju plate su :
    1. Ako je radno mesto [Prodavac] i zaposleni postane [Zamenik Menadzera
    Prodavca] potrebno je UVECATI platu od 5%
    2. Ako je radno mesto [Prodavac] i zaposleni postane [Menadzera Prodavac]
    potrebno je UVECATI platu od 10%
    3. Ako je radno mesto [Zamenik Menadzera Prodavca] i zaposleni postane
    [Prodavac] potrebno je SMANJITI platu od 8%
    4. Ako je radno mesto [Zamenik Menadzera Prodavca] i zaposleni postane
    [Menadzera Prodavac] potrebno je UVECATI platu od 5%
    5. Ako je radno mesto [Menadzera Prodavac] i zaposleni postane [Prodavac]
    potrebno je SMANJITI platu od 12%
    6. Ako je radno mesto [Menadzera Prodavac] i zaposleni postane [Zamenik
    Menadzera Prodavca] potrebno je SMANJITI platu od 6%
*/
if object_id('Radnik.Trigger_Zaposleni','TR') is not null
```



```

drop trigger Radnik.Trigger_Zaposleni
go

create trigger Radnik.Trigger_Zaposleni on Radnik.Zaposleni
after update
as
begin
    declare @Radnik as int;
    set @Radnik=(select SIF_Z from inserted)

    declare @Radno_Mesto_staro as varchar(50);
    set @Radno_Mesto_staro = (select RADNO_MESTO from deleted);

    declare @Radno_Mesto_novo as varchar(50);
    set @Radno_Mesto_novo = (select RADNO_MESTO from inserted);

    if(@Radno_Mesto_staro like 'Prodavac')
    begin
        if(@Radno_Mesto_novo like 'Zamenik Menadzera Prodavca')
        begin
            update Radnik.Zaposleni
            set PLATA=PLATA+PLATA*0.05
            where SIF_Z=@Radnik
        end
        if(@Radno_Mesto_novo like 'Menadzer Prodavac')
        begin
            update Radnik.Zaposleni
            set PLATA=PLATA+PLATA*0.1
            where SIF_Z=@Radnik
        end
    end

    if(@Radno_Mesto_staro like 'Zamenik Menadzera Prodavca')
    begin
        if(@Radno_Mesto_novo like 'Prodavac')
        begin
            update Radnik.Zaposleni
            set PLATA=PLATA-PLATA*0.08
            where SIF_Z=@Radnik
        end
        if(@Radno_Mesto_novo like 'Menadzer Prodavac')
        begin
            update Radnik.Zaposleni
            set PLATA=PLATA+PLATA*0.05
            where SIF_Z=@Radnik
        end
    end

    if(@Radno_Mesto_staro like 'Menadzer Prodavac')
    begin
        if(@Radno_Mesto_novo like 'Prodavac')
        begin
            update Radnik.Zaposleni
            set PLATA=PLATA-PLATA*0.12
            where SIF_Z=@Radnik
        end
        if(@Radno_Mesto_novo like 'Zamenik Menadzera Prodavca')
        begin
            update Radnik.Zaposleni

```

```

                set PLATA=PLATA-PLATA*0.06
                where SIF_Z=@Radnik
            end
        end
    end
go

/***** TESTE *****/

--TEST 1
--update Prodavac na Zamenik Menadzera Prodavca
select RADNO_MESTO,PLATA from Radnik.Zaposleni where SIF_Z=32

update Radnik.Zaposleni
set RADNO_MESTO='Zamenik Menadzera Prodavca'
where SIF_Z=32

select RADNO_MESTO,PLATA from Radnik.Zaposleni where SIF_Z=32

--update Prodavac na Menadzer Prodavac
select RADNO_MESTO,PLATA from Radnik.Zaposleni where SIF_Z=33

update Radnik.Zaposleni
set RADNO_MESTO='Menadzer Prodavac'
where SIF_Z=33

select RADNO_MESTO,PLATA from Radnik.Zaposleni where SIF_Z=33

--TEST 2
--update Zamenik Menadzera Prodavca na Menadzer Prodavac
select RADNO_MESTO,PLATA from Radnik.Zaposleni where SIF_Z=32

update Radnik.Zaposleni
set RADNO_MESTO='Menadzer Prodavac'
where SIF_Z=32

select RADNO_MESTO,PLATA from Radnik.Zaposleni where SIF_Z=32

--update Menadzer Prodavac u Zamenik Menadzer Prodavca
select RADNO_MESTO,PLATA from Radnik.Zaposleni where SIF_Z=33

update Radnik.Zaposleni
set RADNO_MESTO='Zamenik Menadzera Prodavca'
where SIF_Z=33

select RADNO_MESTO,PLATA from Radnik.Zaposleni where SIF_Z=33

```

```

--TEST 3
--update Menadzer Prodavac na Prodavac
select RADNO_MESTO,PLATA from Radnik.Zaposleni where SIF_Z=32

update Radnik.Zaposleni
set RADNO_MESTO='Prodavac'
where SIF_Z=32

select RADNO_MESTO,PLATA from Radnik.Zaposleni where SIF_Z=32

--update Zamenik Menadzera Prodavca na Prodavac
select RADNO_MESTO,PLATA from Radnik.Zaposleni where SIF_Z=33

update Radnik.Zaposleni
set RADNO_MESTO='Prodavac'
where SIF_Z=33

select RADNO_MESTO,PLATA from Radnik.Zaposleni where SIF_Z=33

```

10. Procedure

```

/***** PROCEDURA 1 *****/

Procedura koja ne dozvoljava da prilikom unosa torke u prodavnicu nadredjeni bude
zaduzen za
prodavnicu koja pripada regionionu za koje on nije zaduzen

*****/

if OBJECT_ID('Proizvod_u_Prodavnici.NadredjeniProdavnica','P') is not null
    drop procedure Proizvod_u_Prodavnici.NadredjeniProdavnica
go
create procedure Proizvod_u_Prodavnici.NadredjeniProdavnica

    @SIF_PRODAVNICE as int,
    @ADRESA as varchar(100),
    @KONTAKT_TELEFON as varchar(20),
    @POVRSINA as numeric(6,2),
    @RADNOVREME_RD as varchar(12),
    @RADNOVREME_SUBOTA as varchar(12),
    @RADNOVREME_NEDELJA as varchar(12),
    @SIF_GRADA as int,
    @ARANZER as int,
    @ADMINISTRATOR as int,
    @MENADZER as int

as
begin
    declare @Radno_Mesto as varchar(80);
    set @Radno_Mesto=(select RADNO_MESTO from Radnik.Zaposleni where SIF_Z=@ARANZER);

    declare @msg varchar(400);
    --provera da li odgovara pozicija radnom mestu, npr. aranzjer u prodavnici moze
    biti samo aranzjer, ne neko drugi
    if(@Radno_Mesto not like 'Aranzer' and @Radno_Mesto is not null)
    begin
        set @msg=('Radno mesto ovog radnika je '+@Radno_Mesto+', a na ovom radnom
mestu se trazi Aranzjer');
        THROW 50000, @msg, 1;
    end

    set @Radno_Mesto=(select RADNO_MESTO from Radnik.Zaposleni where
SIF_Z=@ADMINISTRATOR);
    if(@Radno_Mesto not like 'Administrator' and @Radno_Mesto is not null)
    begin
        set @msg=('Radno mesto ovog radnika je '+@Radno_Mesto+',a na ovom radnom
mestu se trazi Administrator');
        THROW 50000,@msg,1;
    end

    set @Radno_Mesto=(select RADNO_MESTO from Radnik.Zaposleni where
SIF_Z=@MENADZER);
    if(@Radno_Mesto not like 'Regionalni Menadzer' and @Radno_Mesto not like
'Regionalni Direktor' and @Radno_Mesto is not null)
    begin
        set @msg=('Radno mesto ovog radnika je '+@Radno_Mesto+',a na ovom radnom
mestu se trazi Regionalni Direktor ili Regionalni Menadzer');
    end
end

```

```

        THROW 50000, @msg ,1;
    end

    declare @region_Grada as int;
    set @region_Grada=(select SIF_REGION from Proizvod_u_Prodavnici.Grad where
SIF_GRAD=@SIF_GRADA)

    declare @sif_Region as int;
    set @sif_Region =(select SIF_REGION from Radnik.Nadredjeni where SIF_Z=@ARANZER);
    if((@sif_Region <> @region_Grada) and @sif_Region is not null)
    begin
        set @msg =('Aranzer nije zaduzen za region u kojem se nalazi prodavnica,
prodavnica se nalazi u regionu '+cast(@region_Grada as varchar(10)));
        THROW 50000, @msg ,1;
    end

    set @sif_Region =(select SIF_REGION from Radnik.Nadredjeni where
SIF_Z=@ADMINISTRATOR);
    if((@sif_Region <> @region_Grada) and @sif_Region is not null)
    begin
        set @msg =('Administrator nije zaduzen za region u kojem se nalazi
prodavnica, prodavnica se nalazi u regionu '+cast(@region_Grada as varchar(10)));
        THROW 50000, @msg ,1;
    end

    set @sif_Region =(select SIF_REGION from Radnik.Nadredjeni where SIF_Z=@MENADZER);
    if((@sif_Region <> @region_Grada) and @sif_Region is not null)
    begin
        set @msg =('Menadzer nije zaduzen za region u kojem se nalazi prodavnica,
prodavnica se nalazi u regionu '+cast(@region_Grada as varchar(10)));
        THROW 50000, @msg ,1;
    end

    insert into Proizvod_u_Prodavnici.Prodavnica
(SIF_PRODAVNICE,ADRESA,KONTAKT_TELEFON,POVRSINA,RADNOVREME_RD,RADNOVREME_SUBOTA,RADNOVREM
E_NEDELJA,SIF_GRADA,ARANZER,ADMINISTRATOR,MENADZER)
    values
(@SIF_PRODAVNICE,@ADRESA,@KONTAKT_TELEFON,@POVRSINA,@RADNOVREME_RD,@RADNOVREME_SUBOTA,@RA
DNOVREME_NEDELJA,@SIF_GRADA,@ARANZER,@ADMINISTRATOR,@MENADZER)

end
go

/*****          TEST          *****/

--Radno mesto nije aranzer
exec Proizvod_u_Prodavnici.NadredjeniProdavnica 10,'Arsenije Carnojevica 45','022/ 223-
626',5,'8:00-22:00','8:00-20:00','08:00-15:00',4,24,11,2

--Radno mesto nije administrator
exec Proizvod_u_Prodavnici.NadredjeniProdavnica 10,'Arsenije Carnojevica 45','022/ 223-
626',5,'8:00-22:00','8:00-20:00','08:00-15:00',4,14,21,2

--Radno mesto nije Regionalni Direktor ili Regionalni Menadzer
exec Proizvod_u_Prodavnici.NadredjeniProdavnica 10,'Arsenije Carnojevica 45','022/ 223-
626',5,'8:00-22:00','8:00-20:00','08:00-15:00',4,14,11,30

```

```

--Aranzer nije zaduzen za odgovarajuci region
exec Proizvod_u_Prodavnici.NadredjeniProdavnica 10,'Arsenije Carnojevetica 45','022/ 223-626',5,'8:00-22:00','8:00-20:00','08:00-15:00',4,13,11,2

--Administrator nije zaduzen za odgovarajuci region
exec Proizvod_u_Prodavnici.NadredjeniProdavnica 10,'Arsenije Carnojevetica 45','022/ 223-626',5,'8:00-22:00','8:00-20:00','08:00-15:00',4,14,9,6

--Administrator nije zaduzen za odgovarajuci region
exec Proizvod_u_Prodavnici.NadredjeniProdavnica 10,'Arsenije Carnojevetica 45','022/ 223-626',5,'8:00-22:00','8:00-20:00','08:00-15:00',4,14,11,6

--unos i torku
exec Proizvod_u_Prodavnici.NadredjeniProdavnica 10,'Arsenije Carnojevetica 45','022/ 223-626',5,'8:00-22:00','8:00-20:00','08:00-15:00',4,14,11,2

--Brisanje unete troke torke
DELETE FROM Proizvod_u_Prodavnici.Prodavnica
where SIF_PRODavnICE=10

--unos i torku ako su aranzer,administrator i menadzer null
exec Proizvod_u_Prodavnici.NadredjeniProdavnica 10,'Arsenije Carnojevetica 45','022/ 223-626',5,'8:00-22:00','8:00-20:00','08:00-15:00',4,null,null,null

```

```

/***** PROCEDURA 2 *****/
Procedura koja za prosledjenu sifru skladišta ispisuje proizvode na sledeći način:
U skladištu sa sifrom [SIF_SKLADISTA] na lokaciji [LOKACIJA] se nalaze sledeći
proizvodi:

[REDNI BROJ], [SIFRA PROIZVODA],[NAZIV],[KOLICINA],[KOMERCIJALNA
CENA],[KATEGORIJA],[VRSTA]

Kao i poruka o ukupnom broj proizvoda

Ukoliko u skladištu nema proizvoda ispisuje se poruka 'U skladištu ne nalazi ni
jedan proizvod!'

Ukoliko ne postoji skladište ispisuje se poruka 'Ne postoji skladište sa sifrom
[SIF_SKLADISTA]'

*****/

if OBJECT_ID('Transport_Skladiste_Prodavnica.PregledProizvodaSkladista','P')is not null
drop procedure Transport_Skladiste_Prodavnica.PregledProizvodaSkladista
go
create procedure Transport_Skladiste_Prodavnica.PregledProizvodaSkladista
@sif_SKLADISTA as int
as
begin

declare @rbr as int,
@sif_proizvoda as int,
@naziv as varchar(60),
@kolicina as numeric(4,0),
@kategorija as varchar(60),
@cena as money,
@vrsta as varchar(60),
@akcija as date,
@lokacija as varchar(100),
@brojProizvoda as int;

if exists(select* from Transport_Skladiste_Prodavnica.Skladiste where
SIF_SKLADISTA=@SIF_SKLADISTA)
begin

set @brojProizvoda= (select count(*) from
Transport_Skladiste_Prodavnica.SkladisteStanje where SIF_SKLADISTA=@SIF_SKLADISTA)

if(@brojProizvoda>0)
begin
set @rbr=1;
set @lokacija=(select LOKACIJA from
Transport_Skladiste_Prodavnica.Skladiste where SIF_SKLADISTA=@SIF_SKLADISTA)
print('U skladištu sa sifrom '+cast(@SIF_SKLADISTA as varchar(10))+
na lokaciji '+@lokacija+' se nalaze sledeći proizvodi:');

declare cursor_skladiste cursor fast_forward for
select
ss.SIF_PROIZVODA,NAZIV_PROIZVODA,KOLICINA,KOMERCIJALNA_CENA,KATEGORIJA,VRSTA
from Transport_Skladiste_Prodavnica.SkladisteStanje ss

```

```

                                inner join Proizvod_u_Prodavnici.Proizvod p on
(ss.SIF_PROIZVODA=p.SIF_PROIZVODA)

                                where ss.SIF_SKLADISTA=@SIF_SKLADISTA

                                open cursor_skladiste;

                                fetch next from cursor_skladiste into
@sif_proizvoda,@naziv,@kolicina,@cena,@kategorija,@vrsta;

                                while @@FETCH_STATUS=0
                                begin
                                        print(cast(@rbr as varchar(10))+', sifra:
'+cast(@sif_proizvoda as varchar(10))+', naziv: '+@naziv+', kolicina: '+cast(@kolicina as
varchar(10))+
                                        ', prodajna cena: '+cast(@cena as varchar(10))+', kategorija:
'+@kategorija +', vrsta: '+@vrsta);

                                        fetch next from cursor_skladiste into
@sif_proizvoda,@naziv,@kolicina,@cena,@kategorija,@vrsta;
                                        set @rbr=@rbr+1;
                                end

                                close cursor_skladiste

                                deallocate cursor_skladiste;

                                print('Ukupno proizvoda: ' + cast(@brojProizvoda as varchar(10)))

                                end
                                else
                                        print('U skladistu ne nalazi ni jedan proizvod!')

                                end
                                else
                                        print('Ne postoji skladiste sa sifrom '+cast(@SIF_SKLADISTA as
varchar(10)));

                                end
                                go

/***** TEST *****/

--nepostojece skladiste
exec Transport_Skladiste_Prodavnica.PregledProizvodaSkladista 4

--postojee skladiste
exec Transport_Skladiste_Prodavnica.PregledProizvodaSkladista 2

```


11. Zaključak

U ovom sistemu praćene su aktivnosti jednog lanca prodavnice, kao što su nabavka, isporuka, transport proizvoda, *online* poručivanje, dodeljivanje bonusa radnicima itd. Zabeleženi su svi mogući podaci koji su neophodni za adekvatno kontrolisanje i upravljanje sistemom. Naš tim se potrudio da ti podaci budu zabeleženi, prikazani i pohranjeni u bazi podataka na najefikasniji način, u nadi da se dovede do najmanje pojave redundanse kao i da se pouzdanost podataka dovede na najviši mogući nivo.

Važno je naglasiti da je šema rađena sa ciljem da se konstantno može nadograđivati.