

**BAZE PODATAKA**

Projekat

**Projektovanje baze podataka za firmu koja se bavi**

**termotehnikom**

Predmetni nastavnik:

dr Milosav Majstorović Studenti:

Marko Gojak 131/20

Vladan Tešić 98/20

Uroš Gojak 89/20

Luka Vukoje 118/20

Datum predaje 21.03.2022

**Beograd**

**Mart 2022.**

# Sadržaj

[Poslovni domen - OPIS](#_heading=h.gjdgxs) 4

[PMOV](#_heading=h.30j0zll) 6

[Integrisani model podataka (PMOV)](#_heading=h.1fob9te) 10

[Prevodjenje PMOV u relacioni model](#_heading=h.2et92p0) 11

[SQL naredbe za kreiranje tabela](#_heading=h.3dy6vkm) 14

[SQL naredbe za umetanje podataka](#_heading=h.2s8eyo1) 20

[Tabelarni prikaz sadrzaja tabela](#_heading=h.3rdcrjn) 25

[Prikaz baznih pogleda, procedura i funkcija](#_heading=h.lnxbz9) 32

[Podsistem Radnik](#_heading=h.1ksv4uv) 32

[Pogled](#_heading=h.2jxsxqh) 32

[Bazna procedura](#_heading=h.3j2qqm3) 32

[Bazna funkcija](#_heading=h.4i7ojhp) 33

[Podsistem Projektovanje](#_heading=h.1ci93xb) 34

[Pogled](#_heading=h.2bn6wsx) 34

[Bazna procedura](#_heading=h.qsh70q) 34

[Bazna funkcija](#_heading=h.3as4poj) 35

[Podsistem Izvodjenje](#_heading=h.1pxezwc) 36

[Pogled](#_heading=h.49x2ik5) 36

[Bazna Procedura](#_heading=h.147n2zr) 36

[Bazna funkcija](#_heading=h.23ckvvd) 37

[Podsistem Nabavka](#_heading=h.32hioqz) 38

[Pogled](#_heading=h.41mghml) 38

[Bazna Procedura](#_heading=h.3fwokq0) 39

[Bazna Funkcija](#_heading=h.50wrmr3id20b) 39

[Literatura](#_heading=h.3tbugp1) 40

# Poslovni domen - OPIS

Za potrebe preduzeća koje se bavi instaliranjem centralnog grejanja potrebno je isprojektovati bazu podataka. Poslovni domen je podeljen na 4 celine: Radnik, Projektovanje, Izvodjenje i Nabavka.

1. Podsistem Radnik (Uroš Gojak)

Radnik koje je obeležen sa svojom šifrom, imenom, prezimenom i platom ove atribute prosledjuje svakoj daljoj podeli radnika i zato nisu navedeni. Radnik se deli na majstora i inženjera. Za inženjera se vodi evidencija o random stažu. Za majstora se vodi evidencija o specijalizaciji posla koji radi (za grejanje, klime, kanale…). Majstor može da bude glavni majstor, junior majstor, pomoćnik. Glavni majstor poseduje sertifikate (razni tipovi zavarivanja…), junior majstoru se beleži radni staž kako bi mogao da postane glavni majstor, dok se za pomoćnika beleži na koji način je zapošlen (honorarno ili pod ugovorom na odredjeno vreme). Glavni majstor je nadredjen odredjenom broj majstora i pomoćnika, a svaki junior majstor može da ima najviše jednog pomoćnika. Inženjer može da bude projektant ili izvođač. Svakom od njih se vodi evidencija o statusu (da li je licenciran ili pripravnik). Za oba inženjera se vodi evidencija ko mu je nadredjeni, ako ga ima. Svaki podredjeni inženjer mora da ima jednog nadredjenog, dok svaki nadredjeni može imati više podredjenih. Svaki inženjer projektant je angažovan na PIZ-u. Na svakom PIZ-u radi najmanje jedan projektant. Za svaki PIZ se vodi evidencija o lokaciji, obimu projekta, početku i završetku projekta, šifri projekta. Zadatak inženjera izvođača je da napiše PIO - projekat izvedenog objekta. PIO je projekat stvarno izvedene instalacije. Za PIO se vodi evidencija o datumu predaje, šifri i nazivu.

1. Podsistem Projektovanje (Luka Vukoje)

Svaki inženjer projektant je angažovan na projektu (zove se PIZ - projekat za izvodjenje).

Svaki projekat ima glavnog odgovornog projektanta. Za svaki projekat se vodi evidencija o lokaciji, obimu projekta, početku i završetku projekta i šifri projekta. PIZ ima jednu ili više stavki. Na svakoj stavci PIZ-a radi jedan ili više projektanata. Za stavku se vodi evidencija o identifikatoru stavke, datumu početka realizacije, predvidjenom vremenu i datumu završetka. Svaki inženjer projektant zna da koristii određene softverske pakete. Softverski paket ima šifru, naziv i cenu implementacije. Projektant odlazi na obuku gde usavršava rad nad softverskim paketom. Obuka ima šifru, naziv kursa, vreme trajanja i sertifikat o kvalifikaciji. Nakon završenog PIZ-a preduzeće izdaje izlazni račun. Izlazni račun se knjiži i vodi se evidencija o identifikaciji računa, ceni i datumu.

1. Podsistem Izvodjenje (Marko Gojak)

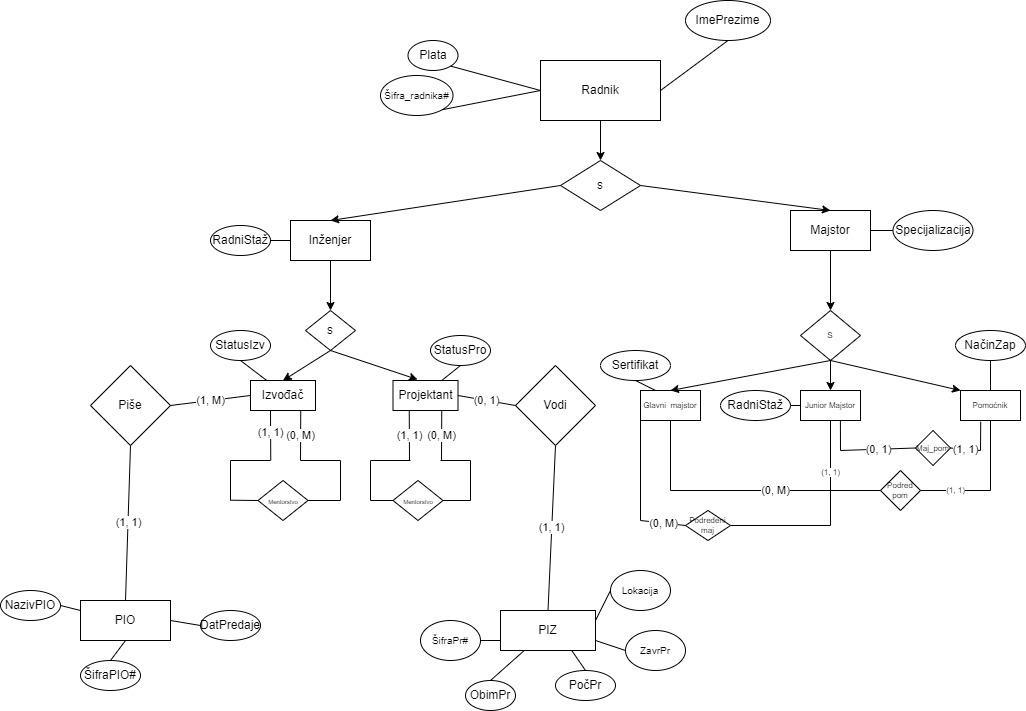
Inženjer izvođač može biti angažovan na više gradilišta za koja se beleži šifra, naziv, lokacija i početak i završetak radova. Svako gradilište mora da ima bar jednog inženjera izvođača. Zadatak inženjera izvođača je da napiše PIO - projekat izvedenog objekta. PIO je projekat stvarno izvedene instalacije (u našem slučaju proizvoda) i u njemu se unose izmene u odnosu na PIZ koji je odradio projektant za gradilište. Za PIO se vodi evidencija o datumu predaje i šifri. PIO ima stavke izmena kojih može biti više. Stavka ima redni broj stavke. Inženjer izvođač je zadužen za nabavku opreme i materijala, koji se obeležava kao proizvod. Inženjer kontaktira dobavljače, koji su opisani identifikatorom i nazivom, i traži ponudu za proizvod koji se nalaze u projektu za određeno gradilište. Za projekat gradilišta se beleže identifikator, naziv i datum kada je prodat, takodje gleda se na koje gradilište se odnosi. Za ponudu se vodi evidencija o datumu izdavanja i šifri. Ponuda ima jednu ili više stavki ponude. Za stavku se vodi evidencija o rednom broju, količini naručenog proizvoda i roku isporuke. Proizvod ima svoj naziv, šifru i cenu. Nakon kupovine proizvoda dobavljač izdaje ulaznu fakturu preduzeću koja se zavodi i ima atribute šifra i datum plaćanja. Izvodjača i ponudu za koju dobija ulaznu fakturu povezuje kupovina za koju se pamti šifra i datum.

1. Podsistem Nabavka (Vladan Tešić)

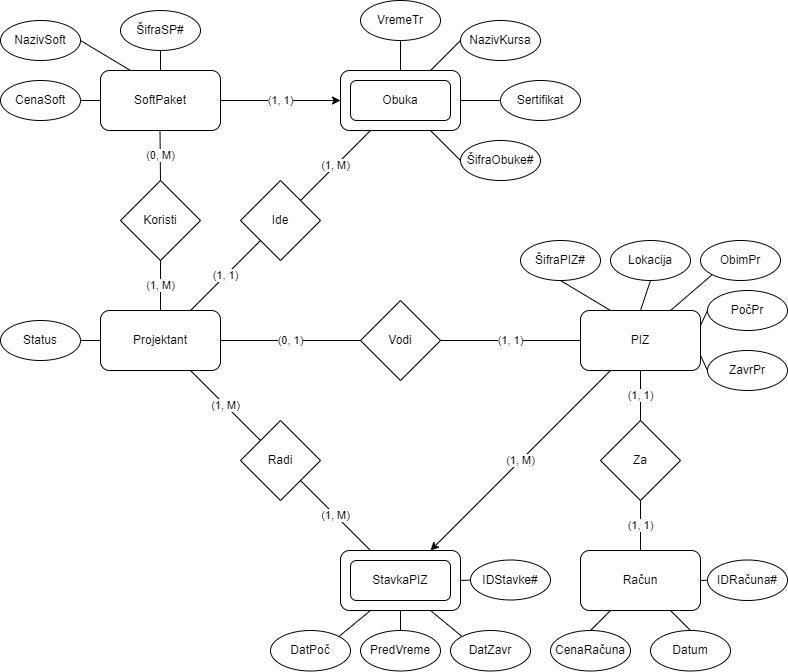
Na osnovu ulazne fakture inženjer izvođač angažuje jednog ili više majstora da obave transport proizvoda od dobavljača do gradilišta. Za transport se vodi evidencija o datumu, utrošenom gorivu, angažovanom vozilu i angažovanom broju ljudi. Za svaki transport majstor piše putni nalog koji se zavodi sa informacijama o relaciji puta, datumu i svojoj jedinstvenoj šifri. Prilikom preuzimanja proizvoda majstor od dobavljača dobija otpremnicu. Otpremnica ima atribute datum izdavanja i svoju jedinstvenu šifru. Otpremnica ima stavke otpremnice. Svaka stavka pokazuje na proizvod koji se prenosi i količinu preuzete robe. Izvodjenje radova na gradilištu, za koje pamtimo ID i naziv, je podeljeno u situacije. Svaka situacija ima šifru, predvidjeni rok početka, završetka, količinu ugrađenog proizvoda i cenu. Nakon svake završene situacije preduzeće pravi izlazni račun koji daje investitoru. Izlazni račun se knjiži i vodi se evidencija o ceni i datumu. Za investitore se pamte šifra i naziv.

# PMOV

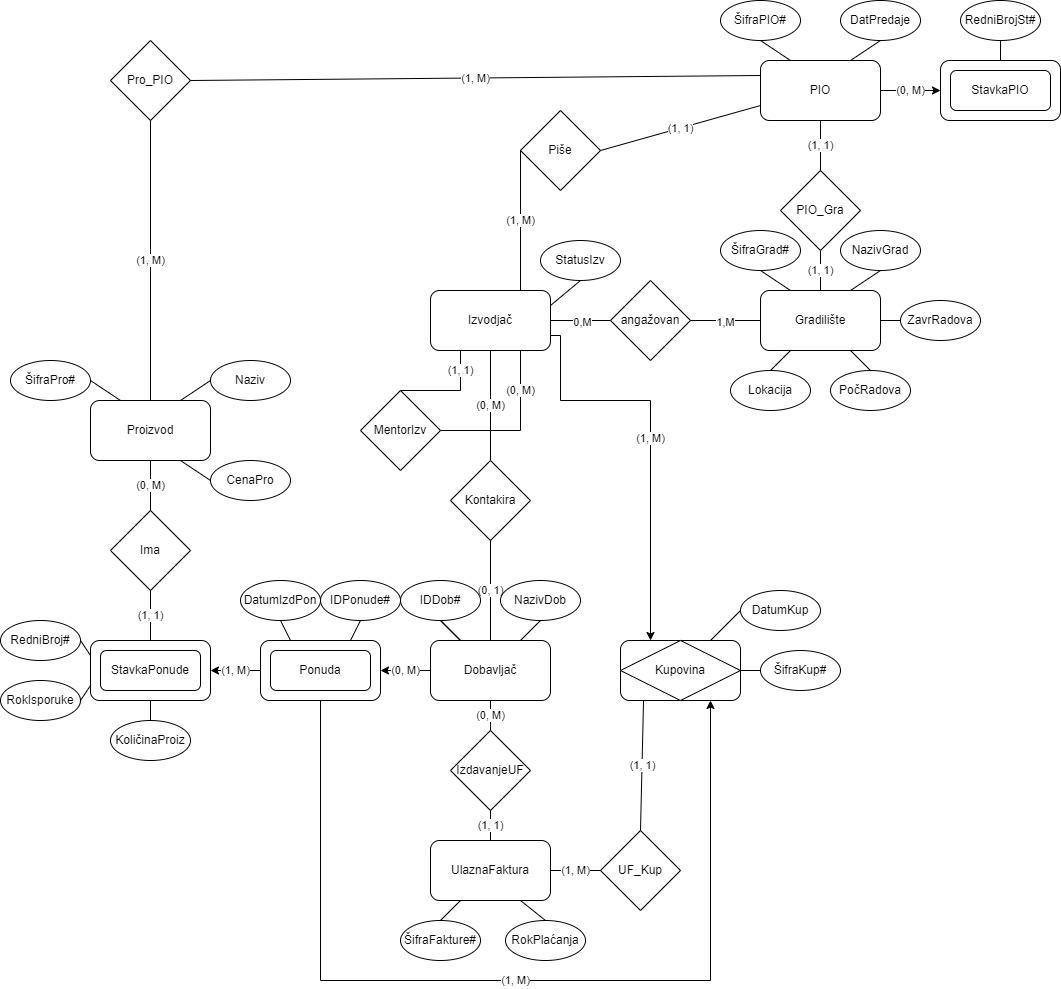
U nastavku su prikazani PMOV, prethodno opisanih celina.



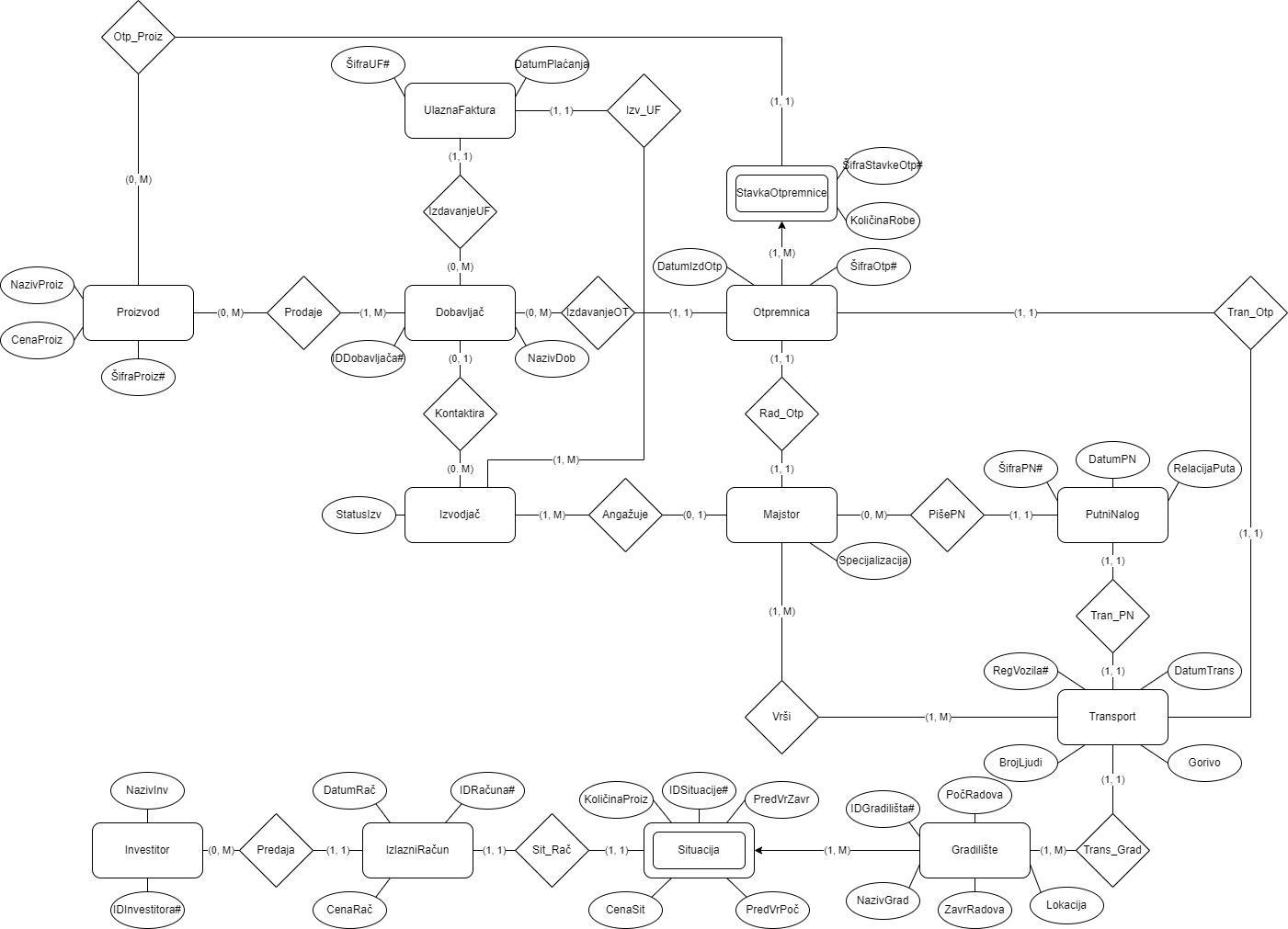
*Slika 1. PMOV - Radnik*



*Slika 2. PMOV - Projektovanje*

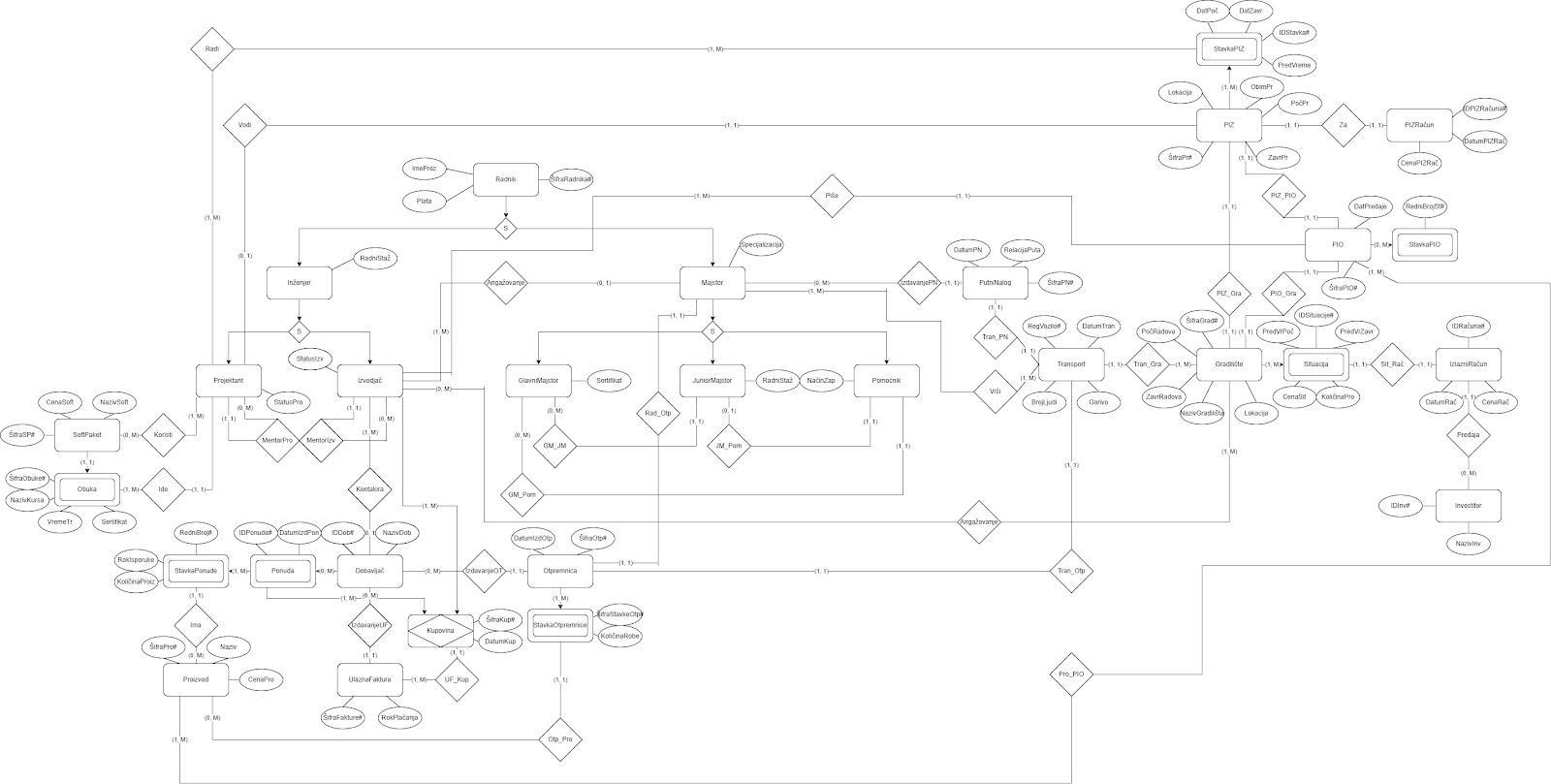


*Slika 3. PMOV - Izvodjenje*



*Slika 4. PMOV - Nabavka*

# Integrisani model podataka (PMOV)



*Slika 5. Integrisani PMOV*

# Prevodjenje PMOV u relacioni model

Radnik(Šifra\_radnika, ImePrezime, Plata)

Proizvod (ŠifraProiz, Naziv, CenaPro)

SoftPaket(ŠifraSP, NazivSoft, CenaSoft)

Obuka(ŠifraObuke, ŠifraSP, VremeTr, NazivKursa, Sertifikat)

Obuka(ŠifraSP) referencira SoftPaket(ŠifraSP)

Investitor (IDInv, NazivInv)

Inženjer(Šifra\_radnika, RadniStaž)

Inženjer(Šifra\_radnika) referencira Radnik(Šifra\_radnika)

Izvodjač(Šifra\_radnika, *Šifra\_mentora*,StatusIzv)

Izvodjač(Šifra\_radnika) referencira Inženjer(Šifra\_radnika)

Izvodjač(Šifra\_mentora) referencira Izvodjač(Šifra\_radnika)

Dobavljač (IDDob, *IdKontakta*, NazivDob)

Dobavljač(IdKontakta) referencira Izvodjač(Šifra\_radnika)

PIZ(ŠifraPIZ, *ŠifraGrad*, *ŠifraPIO*, *IDPIZRačuna*, *ŠifraVodje*, ObimPr, PočetakPr, ZavršetakPr, Lokacija)

PIZ(ŠifraGrad) referencira Gradilište(ŠifraGrad)

PIZ(ŠifraPIO) referencira PIO(ŠifraPIO)

PIZ(IDPIZRačuna) referencira PIZRačun(IDPIZRačuna)

PIZ(ŠifraVodje) referencira Projektant(Šifra\_radnika)

PIZRačun(IDPIZRačuna, *ŠifraPIZ*, CenaPIZRačuna, DatumPIZRačuna)

Račun(ŠifraPIZ) referencira PIZ(ŠifraPIZ)

PIO (ŠifraPIO, *ŠifraPIZ*, *ŠifraGrad, Šifra\_radnika*, DatPredaje)

PIO(ŠifraPIZ) referencira PIZ(ŠifraPIZ)

PIO (ŠifraGrad) referencira Gradilište (ŠifraGrad)

PIO (Šifra\_radnika) referencira Izvođač (Šifra\_radnika)

Gradilište (ŠifraGrad, *ŠifraPIO*, *ŠifraPIZ*, NazivGrada, Lokacija, PočRadova, ZavrRadova)

Gradilište(ŠifraPIO) referencira PIO(ŠifraPIO)

Gradilište(ŠifraPIZ) referencira PIZ(ŠifraPIZ)

StavkaPIZ(IDStavke, ŠifraPIZ, PredVreme, DatPoč, DatZavr)

StavkaPIZ(ŠifraPIZ) referencira PIZ(ŠifraPIZ)

Projektant(Šifra\_radnika, *ŠifraPIZaKojiVodi*, *ŠifraObuke, ŠifraSPObuke, Šifra\_mentora*,StatusPro)

Projektant(Šifra\_radnika) referencira Inženjer(Šifra\_radnika)

Projektant(ŠifraPIZaKojiVodi) referencira PIZ(ŠifraPIZ)

Projektant(ŠifraObuke) referencira Obuka(ŠifraObuke)

Projektant(ŠifraSPObuke) referencira Obuka(ŠifraSP)

Projektant(Šifra\_mentora) referencira Projektant(Šifra\_radnika)

Majstor (Šifra\_radnika, *IDAngažovanje*, *ŠifraOtp*, Specijalizacija)

Majstor(IDAngažovanje) referencira Izvodjač(Šifra\_radnika)

Majstor(ŠifraOtp) referencira Otpremnica(ŠifraOtp)

Otpremnica (ŠifraOtp, *RegVozila*, *IDDob*, *ŠifraMajstora*, DatumIzdOtp)

Otpremnica(RegVozila) referencira Transport(RegVozila)

Otpremnica(IDDob) referencira Dobavljač(IDDob)

Otpremnica(ŠifraMajstora) referencira Majstor(Šifra\_radnika)

Transport (RegVozila, *ŠifraPN*, *ŠifraOtp*, *ŠifraGrad*, BrojLjudi, Gorivo, DatumTrans)

Transport(ŠifraPN) referencira PutniNalog(ŠifraPN)

Transport(ŠifraOtp) referencira Otpremnica(ŠifraOtp)

Transport(ŠifraGrad) referencira Gradilište(ŠifraGrad)

PutniNalog (ŠifraPN, *ŠifraMajstora*, *RegVozila*, RelacijaPuta, DatumPN)

PutniNalog(ŠifraMajstora) referencira Majstor(Šifra\_radnika)

PutniNalog(RegVozila) referencira Transport(RegVozila)

UlaznaFaktura (ŠifraFakture, RokPlaćanja, *IDDob*)

UlaznaFakutra(IDDob) referencira Dobavljač(IDDob)

IzlazniRačun (IDRačuna, *IDSituacije, SituacijaŠifraGrad*, *IDInv*, CenaRač, DatumRač)

IzlazniRačun(IDSituacije) referencira Situacija(IDSituacije)

IzlazniRačun(SituacijaŠifraGrad) referencira Situacija(ŠifraGrad)

IzlazniRačun(IDInv) referencira Investitor(IDInv)

StavkaPIO (RedniBrojSt, ŠifraPIO)

StavkaPIO(ŠifraPIO) referencira PIO(ŠifraPIO)

Ponuda (IDPonude, IDDob, DatumIzdPon)

Ponuda(IDDob) referencira Dobavljač(IDDob)

StavkaPonude (RedniBroj, IDPonude, IDDob, *ŠifraProiz*, KoličinaProiz, RokIsporuke)

StavkaPonude(IDPonude) referencira Ponuda(IDPonude)

StavkaPonude(IDDob) referencira Ponuda(IDDob)

StavkaPonude(ŠifraProiz) referencira Proizvod(ŠifraProiz)

Kupovina (ŠifraKup, Šifra\_radnika, IDPonude, IDDob, *ŠifraFakture*, DatumKup)

Kupovina(IDPonude) referencira Ponuda(IDPonude)

Kupovina(IDDob) referencira Ponuda(IDDob)

Kupovina(Šifra\_radnika) referencira Izvođač(Šifra\_radnika)

Kupovina(ŠifraFakture) referencira UlaznaFakutra(ŠifraFakture)

StavkaOtpremnice (ŠifraStavkeOtp, ŠifraOtp, *ŠifraProiz*, KoličinaRobe)

StavkaOtpremnice(ŠifraOtp) referencira Otrpemnica(ŠifraOtp)

StavkaOtpremnice(ŠifraProiz) referencira Proizvod(ŠifraProiz)

Situacija (IDSituacije, ŠifraGrad, *IDRačuna*, CenaSit, KoličinaPro, PredVrPoč, PredVrZavr)

Situacija(ŠifraGrad) referencira Gradilište(ŠifraGrad)

Situacija(IDRačuna) referencira IzlazniRačun(IDRačuna) GlavniMajstor(Šifra\_radnika, Sertifikat)

GlavniMajstor(Šifra\_radnika) referencira Majstor(Šifra\_radnika)

JuniorMajstor(Šifra\_radnika, *NadredjeniMajstor*, *PodredjeniPom*, RadniStaž)

JuniorMajstor(Šifra\_radnika) referencira Majstor(Šifra\_radnika)

JuniorMajstor(NadredjeniMajstor) referencira GlavniMajstor(Šifra\_radnika)

JuniorMajstor(PodredjeniPom) referencira Pomoćnik(Šifra\_radnika)

Pomoćnik(Šifra\_radnika, *NadredjeniGM*, *NadredjeniJM*, NačinZap)

Pomoćnik(Šifra\_radnika) referencira Majstor(Šifra\_radnika)

Pomoćnik(NadredjeniGM) referencira GlavniMajstor(Šifra\_radnika)

Pomoćnik(NadredjeniJM) referencira JuniorMajstor(Šifra\_radnika)

Angažovan (ŠifraIzv, ŠifraGrad)

Angažovan(ŠifraIzv) referencira Izvodjač(Šifra\_radnika)

Angažovan(ŠifraGrad) referencira Gradilište(ŠifraGrad)

Proiz\_PIO (ŠifraProiz, ŠifraPIO)

Pro\_PIO(ŠifraProiz) referencira Proizvod(ŠifraProiz)

Pro\_PIO(ŠifraPIO) referencira PIO(ŠifraPIO)

IzvršavanjeTransporta(ŠifraMajstora, RegVozila)

IzvršavanjeTransporta(ŠifraMajstora) referencira Majstor(Šifra\_radnika)

IzvršavanjeTransporta(RegVozila) referencira Transport(RegVozila)

KorišćenjeSoftvera(ŠifraSP, Šifra\_radnika)

KorišćenjeSoftvera(ŠifraSP) referencira SoftPaket(ŠifraSP)

KorišćenjeSoftvera(Šifra\_radnika) referencira Projektant(Šifra\_radnika)

RadNaStavci(Šifra\_radnika, IDStavke, ŠifraPIZ)

RadNaStavci(Šifra\_radnika) referencira Projektant(Šifra\_radnika)

RadNaStavci(IDStavke) referencira StavkaPIZ(IDStavke)

RadNaStavci(ŠifraPIZ) referencira StavkaPIZ(ŠifraPIZ)

# SQL naredbe za kreiranje tabela

IF EXISTS (SELECT name FROM master.dbo.sysdatabases WHERE name = N'BazeProjekat') DROP DATABASE BazeProjekat

CREATE DATABASE BazeProjekat

USE BazeProjekat

create table Radnik

(

Sifra\_radnika int primary key,

ImePrezime varchar(30),

Plata numeric(6, 0)

)

create table Proizvod

(

SifraProiz int primary key,

Naziv varchar(50),

CenaPro numeric(7, 0)

)

create table SoftPaket

(

SifraSP int primary key,

NazivSoft varchar(30),

CenaSoft numeric(6, 0)

)

create table Obuka

(

SifraObuke int,

SifraSP int,

VremeTr int,

NazivKursa varchar(30),

Sertifikat varchar(30),

primary key(SifraObuke, SifraSP),

foreign key(SifraSP) references SoftPaket(SifraSP)

)

create table Investitor

(

IDInv int primary key,

NazivInv varchar(30)

)

create table Inzenjer

(

Sifra\_radnika int primary key,

RadniStaz numeric(2, 0),

foreign key(Sifra\_radnika) references Radnik(Sifra\_radnika)

)

create table Izvodjac

(

Sifra\_radnika int primary key,

SifraMentora int,

StatusIzv varchar(15) check(StatusIzv in ('Licenciran', 'Pripravnik')),

foreign key(Sifra\_radnika) references Inzenjer(Sifra\_radnika),

foreign key(SifraMentora) references Izvodjac(Sifra\_radnika)

)

create table Dobavljac

(

IDDob int primary key,

NazivDob varchar(30),

IDKontakta int foreign key references Izvodjac(Sifra\_radnika)

)

create table PIZ

(

SifraPIZ int primary key,

ObimPr varchar(15),

PocetakPr date,

ZavrsetakPr date,

Lokacija varchar(20),

--SifraGrad int foreign key references Gradiliste(SifraGrad),

--SifraPIO int foreign key references PIO(SifraPIO),

--IDPIZRacuna int foreign key references PIZRacun(IDPIZRacuna),

--SifraVodje int foreign key references Projektant(Sifra\_radnika)

)

create table PIZRacun

(

IDPIZRacuna int primary key,

CenaPIZRacuna numeric(7, 0),

DatumPIZRacuna date,

SifraPIZ int foreign key references PIZ(SifraPIZ)

)

create table PIO

(

SifraPIO int primary key,

DatPredaje date,

SifraPIZ int foreign key references PIZ(SifraPIZ),

Sifra\_radnika int foreign key references Izvodjac(Sifra\_radnika),

--SifraGrad int foreign key references Gradiliste(SifraGrad)

)

create table Gradiliste

(

SifraGrad int primary key,

NazivGrad varchar(30),

Lokacija varchar(20),

PocRadova date,

ZavrRadova date,

SifraPIO int foreign key references PIO(SifraPIO),

SifraPIZ int foreign key references PIZ(SifraPIZ)

)

create table StavkaPIZ

(

IDStavke int,

SifraPIZ int,

PredVreme int,

DatPoc date,

DatZavr date,

primary key(IDStavke, SifraPIZ),

foreign key(SifraPIZ) references PIZ(SifraPIZ)

)

create table Projektant

(

Sifra\_radnika int primary key,

StatusPro varchar(15) check(StatusPro in ('Licenciran', 'Pripravnik')),

foreign key(Sifra\_radnika) references Inzenjer(Sifra\_radnika),

SifraPIZaKojiVodi int foreign key references PIZ(SifraPIZ),

SifraObuke int,

SifraSP int,

foreign key(SifraObuke, SifraSP) references Obuka(SifraObuke, SifraSP),

SifraMentora int foreign key references Projektant(Sifra\_radnika)

)

create table Majstor

(

Sifra\_radnika int primary key,

Specijalizacija varchar(30),

IDAngazovanje int foreign key references Izvodjac(Sifra\_radnika),

--SifraOtp int foreign key references Otpremnica(SifraOtp)

)

create table Otpremnica

(

SifraOtp int primary key,

DatumIzdOtp date,

--RegVozila varchar(10) foreign key references Transport(RegVozila),

IDDob int foreign key references Dobavljac(IDDob),

SifraMajstora int foreign key references Majstor(Sifra\_radnika)

)

create table Transport

(

RegVozila varchar(10) primary key,

BrojLjudi int,

Gorivo numeric(4, 1),

DatumTrans date,

--SifraPN int foreign key references PutniNalog(SifraPN),

SifraOtp int foreign key references Otpremnica(SifraOtp),

SifraGrad int foreign key references Gradiliste(SifraGrad)

)

create table PutniNalog

(

SifraPN int primary key,

RelacijaPuta varchar(50),

DatumPN date,

SifraMajstora int foreign key references Majstor(Sifra\_radnika),

RegVozila varchar(10) foreign key references Transport(RegVozila)

)

create table UlaznaFaktura

(

SifraFakture int primary key,

RokPlacanja date,

IDDob int foreign key references Dobavljac(IDDob)

)

create table IzlazniRacun

(

IDRacuna int primary key,

CenaRac numeric(7, 0),

DatumRac date,

IDSituacije int,

SituacijaSifraGrad int,

--foreign key(IDSituacije, SituacijaSifraGrad) references Situacija(IDSituacije, SifraGrad),

IDInv int foreign key references Investitor(IDInv)

)

create table StavkaPIO

(

RedniBrojSt int,

SifraPIO int foreign key references PIO(SifraPIO),

primary key(RedniBrojSt, SifraPIO)

)

create table Ponuda

(

IDPonude int,

IDDob int foreign key references Dobavljac(IDDob),

DatumIzdPon date,

primary key(IDPonude, IDDob)

)

create table StavkaPonude

(

RedniBroj int,

IDPonude int,

IDDob int,

SifraProiz int foreign key references Proizvod(SifraProiz),

KolicinaProiz int,

RokIsporuke date,

primary key(RedniBroj, IDPonude, IDDob),

foreign key(IDPonude, IDDob) references Ponuda(IDPonude, IDDob)

)

create table Kupovina

(

SifraKup int,

IDPonude int,

IDDob int,

foreign key(IDPonude, IDDob) references Ponuda(IDPonude, IDDob),

Sifra\_radnika int foreign key references Izvodjac(Sifra\_radnika),

DatumKup date,

SifraFakture int foreign key references UlaznaFaktura(SifraFakture),

primary key(SifraKup, IDPonude, IDDob, Sifra\_radnika)

)

create table StavkaOtpremnice

(

SifraStavkeOtp int,

SifraOtp int foreign key references Otpremnica(SifraOtp),

primary key(SifraStavkeOtp, SifraOtp),

KolicinaRobe int,

SifraProiz int foreign key references Proizvod(SifraProiz)

)

create table Situacija

(

IDSituacije int,

SifraGrad int foreign key references Gradiliste(SifraGrad),

primary key(IDSituacije, SifraGrad),

CenaSit numeric(7, 0),

KolicinaPro numeric(3, 0),

PredVrPoc date,

PredVrZavr date,

IDRacuna int foreign key references IzlazniRacun(IDRacuna)

)

create table GlavniMajstor

(

Sifra\_radnika int primary key foreign key references Majstor(Sifra\_radnika),

Sertifikat varchar(20)

)

create table JuniorMajstor

(

Sifra\_radnika int primary key foreign key references Majstor(Sifra\_radnika),

RadniStaz numeric(2, 0),

NadredjeniMajstor int foreign key references GlavniMajstor(Sifra\_radnika),

--PodredjeniPom int foreign key references Pomocnik(Sifra\_radnika)

)

create table Pomocnik

(

Sifra\_radnika int primary key foreign key references Majstor(Sifra\_radnika),

NacinZap varchar(30) check(NacinZap in('Honorarno', 'Ugovorom')),

NadredjeniGM int foreign key references GlavniMajstor(Sifra\_radnika),

NadredjeniJM int foreign key references JuniorMajstor(Sifra\_radnika)

)

create table Angazovan

(

SifraIzv int foreign key references Izvodjac(Sifra\_radnika),

SifraGrad int foreign key references Gradiliste(SifraGrad),

primary key(SifraIzv, SifraGrad)

)

create table Pro\_PIO

(

SifraProiz int foreign key references Proizvod(SifraProiz),

SifraPIO int foreign key references PIO(SifraPIO),

primary key(SifraProiz, SifraPIO)

)

create table IzvrsavanjeTransporta

(

SifraMajstora int foreign key references Majstor(Sifra\_radnika),

RegVozila varchar(10) foreign key references Transport(RegVozila),

primary key(SifraMajstora, RegVozila)

)

create table KoriscenjeSoftvera

(

SifraSP int foreign key references SoftPaket(SifraSP),

Sifra\_radnika int foreign key references Projektant(Sifra\_radnika),

primary key(SifraSP, Sifra\_radnika)

)

create table RadNaStavci

(

Sifra\_radnika int foreign key references Projektant(Sifra\_radnika),

IDStavke int,

SifraPIZ int,

primary key(Sifra\_radnika, IDStavke, SifraPIZ),

foreign key(IDStavke, SifraPIZ) references StavkaPIZ(IDStavke, SifraPIZ)

)

--DODAVNJE KOMENTARISANIH SPOLJNIH KLJUCEVA

alter table PIZ add

SifraGrad int foreign key references Gradiliste(SifraGrad),

SifraPIO int foreign key references PIO(SifraPIO),

IDPIZRacuna int foreign key references PIZRacun(IDPIZRacuna),

SifraVodje int foreign key references Projektant(Sifra\_radnika)

alter table PIO add

SifraGrad int foreign key references Gradiliste(SifraGrad)

alter table Majstor add

SifraOtp int foreign key references Otpremnica(SifraOtp)

alter table Otpremnica add

RegVozila varchar(10) foreign key references Transport(RegVozila)

alter table Transport add

SifraPN int foreign key references PutniNalog(SifraPN)

alter table IzlazniRacun add

foreign key(IDSituacije, SituacijaSifraGrad) references Situacija(IDSituacije, SifraGrad)

alter table JuniorMajstor add

PodredjeniPom int foreign key references Pomocnik(Sifra\_radnik)

# SQL naredbe za umetanje podataka

insert into Radnik values(1, 'Marko Markovic', 85000)

insert into Radnik values(2, 'Mina Minic', 85000)

insert into Radnik values(3, 'Petar Petrovic', 80000)

insert into Radnik values(4, 'Jovana Jovanovic', 95000)

insert into Radnik values(5, 'Ana Anic', 90000)

insert into Radnik values(6, 'Janko Jankovic', 90000)

insert into Radnik values(7, 'Dragan Dragic', 90000)

insert into Radnik values(8, 'Katarina Katic', 70000)

insert into Radnik values(9, 'Ivana Ivanovic', 70000)

insert into Radnik values(10, 'Luka Lukic', 50000)

insert into Radnik values(11, 'Milan Milanovic', 30000)

insert into Proizvod values(1, 'Sistem za grejanje', 325000)

insert into Proizvod values(2, 'Sistem za klimatizaciju', 650000)

insert into Proizvod values(3, 'Sistem za ventilaciju', 120000)

insert into SoftPaket values(1, 'AutoCad', 18000)

insert into SoftPaket values(2, 'Revit', 24000)

insert into Obuka values(1, 1, 20, 'Pocetni kurs', 'Level 1')

insert into Obuka values(2, 2, 40, 'Napredni kurs', 'Level 2')

insert into Investitor values(1, 'Peteg')

insert into Investitor values(2, 'Seven Invest')

insert into Investitor values(3, 'Hotel Slavija')

insert into Inzenjer values(1, 13)

insert into Inzenjer values(2, 8)

insert into Inzenjer values(3, 3)

insert into Inzenjer values(4, 25)

insert into Inzenjer values(5, 5)

insert into Inzenjer values(6, 1)

insert into Izvodjac(Sifra\_radnika, StatusIzv) values(1, 'Licenciran')

insert into Izvodjac(Sifra\_radnika, StatusIzv) values(2, 'Licenciran')

insert into Izvodjac values(3, 1, 'Pripravnik')

insert into Dobavljac values(1, 'PVF', 1)

insert into Dobavljac values(2, 'Doming', 2)

insert into Dobavljac values(3, 'Jakka Group', 3)

insert into PIZ(SifraPIZ, ObimPr, PocetakPr, ZavrsetakPr, Lokacija) values(1, 'Mali', '2021-8-14', '2022-1-24', 'Zemun')

insert into PIZ(SifraPIZ, ObimPr, PocetakPr, ZavrsetakPr, Lokacija) values(2, 'Srednji', '2021-11-16', '2022-2-3', 'Novi Beograd')

insert into PIZ(SifraPIZ, ObimPr, PocetakPr, ZavrsetakPr, Lokacija) values(3, 'Veliki', '2022-1-9', '2022-3-12', 'Stari Grad')

insert into PIZRacun values(1, 76000, '2022-1-24', 1)

insert into PIZRacun values(2, 260000, '2022-2-3', 2)

insert into PIZRacun values(3, 520000, '2022-3-12', 3)

insert into PIO(SifraPIO, DatPredaje, SifraPIZ, Sifra\_radnika) values(1, '2022-3-16', 1, 3)

insert into PIO(SifraPIO, DatPredaje, SifraPIZ, Sifra\_radnika) values(2, '2022-1-24', 2, 2)

insert into PIO(SifraPIO, DatPredaje, SifraPIZ, Sifra\_radnika) values(3, '2022-2-12', 3, 1)

insert into Gradiliste values(1, 'Zemunske Kapije', 'Zemun', '2021-9-1', '2022-1-18', 1, 1)

insert into Gradiliste values(2, 'Sakura', 'Novi Beograd', '2021-11-28', '2022-1-27', 2, 2)

insert into Gradiliste values(3, 'Levi 9', 'Stari Grad', '2022-1-15', '2022-3-4', 3, 3)

insert into StavkaPIZ values(1, 1, 2, '2021-8-14', '2021-9-16')

insert into StavkaPIZ values(2, 1, 3, '2021-9-17', '2022-1-24')

insert into StavkaPIZ values(3, 2, 2, '2021-11-16', '2022-1-13')

insert into StavkaPIZ values(4, 2, 1, '2022-1-14', '2022-2-3')

insert into StavkaPIZ values(5, 3, 1, '2022-1-9', '2022-2-7')

insert into StavkaPIZ values(6, 3, 1, '2022-2-8', '2022-3-12')

insert into Projektant(Sifra\_radnika, StatusPro, SifraPIZaKojiVodi, SifraObuke, SifraSP) values(4, 'Licenciran', 3, 2, 2)

insert into Projektant values(5, 'Pripravnik', 2, 1, 1, 4)

insert into Projektant values(6, 'Pripravnik', 1, 1, 1, 4)

insert into Majstor(Sifra\_radnika, Specijalizacija) values(7, 'Grejac')

insert into Majstor(Sifra\_radnika, Specijalizacija, IDAngazovanje) values(8, 'Limar', 1)

insert into Majstor(Sifra\_radnika, Specijalizacija, IDAngazovanje) values(9, 'Klimatizer', 2)

insert into Majstor(Sifra\_radnika, IDAngazovanje) values(10, 3)

insert into Majstor(Sifra\_radnika) values(11)

insert into Otpremnica(SifraOtp, DatumIzdOtp, IDDob, SifraMajstora) values(1, '2021-8-28', 3, 8)

insert into Otpremnica(SifraOtp, DatumIzdOtp, IDDob, SifraMajstora) values(2, '2021-11-23', 2, 9)

insert into Otpremnica(SifraOtp, DatumIzdOtp, IDDob, SifraMajstora) values(3, '2022-1-13', 1, 10)

insert into Transport(RegVozila, BrojLjudi, Gorivo, DatumTrans, SifraOtp, SifraGrad) values('BG-423-GS', 3, 43.5, '2021-8-30', 1, 1)

insert into Transport(RegVozila, BrojLjudi, Gorivo, DatumTrans, SifraOtp, SifraGrad) values('BG-678-AD', 4, 50.3, '2021-11-26', 2, 2)

insert into Transport(RegVozila, BrojLjudi, Gorivo, DatumTrans, SifraOtp, SifraGrad) values('BG-5694-DB', 2, 47.5, '2022-1-14', 3, 3)

insert into PutniNalog values(1, 'Banovo Brdo - Zemun', '2021-8-30', 8, 'BG-423-GS')

insert into PutniNalog values(2, 'Dorćol - Novi Beograd', '2021-11-26', 9, 'BG-678-AD')

insert into PutniNalog values(3, 'Voždovac - Stari Grad', '2022-1-14', 10, 'BG-5694-DB')

insert into UlaznaFaktura values(1, '2022-2-14', 1)

insert into UlaznaFaktura values(2, '2022-3-7', 2)

insert into UlaznaFaktura values(3, '2022-4-17', 3)

insert into IzlazniRacun(IDRacuna, CenaRac, DatumRac, IDInv) values(1, 110120, '2021-10-25', 1)

insert into IzlazniRacun(IDRacuna, CenaRac, DatumRac, IDInv) values(2, 120530, '2021-12-14', 2)

insert into IzlazniRacun(IDRacuna, CenaRac, DatumRac, IDInv) values(3, 140230, '2021-2-18', 1)

insert into IzlazniRacun(IDRacuna, CenaRac, DatumRac, IDInv) values(4, 100570, '2021-10-25', 3)

insert into IzlazniRacun(IDRacuna, CenaRac, DatumRac, IDInv) values(5, 90410, '2021-12-14', 2)

insert into IzlazniRacun(IDRacuna, CenaRac, DatumRac, IDInv) values(6, 150760, '2021-2-18', 3)

insert into StavkaPIO values(1, 1)

insert into StavkaPIO values(2, 1)

insert into StavkaPIO values(3, 2)

insert into StavkaPIO values(4, 2)

insert into StavkaPIO values(5, 3)

insert into StavkaPIO values(6, 3)

insert into Ponuda values(1, 1, '2021-8-20')

insert into Ponuda values(2, 2, '2021-11-17')

insert into Ponuda values(3, 3, '2021-1-7')

insert into StavkaPonude values(1, 1, 1, 1, 3, '2021-8-30')

insert into StavkaPonude values(2, 1, 1, 2, 2, '2021-8-31')

insert into StavkaPonude values(3, 2, 2, 1, 1, '2021-11-25')

insert into StavkaPonude values(4, 2, 2, 3, 5, '2021-11-26')

insert into StavkaPonude values(5, 3, 3, 2, 4, '2022-1-12')

insert into StavkaPonude values(6, 3, 3, 3, 6, '2022-1-14')

insert into Kupovina values(1, 1, 1, 3, '2021-8-24', 1)

insert into Kupovina values(2, 2, 2, 2, '2021-11-20', 2)

insert into Kupovina values(3, 3, 3, 1, '2022-1-10', 3)

insert into StavkaOtpremnice values(1, 1, 10, 1)

insert into StavkaOtpremnice values(2, 1, 20, 1)

insert into StavkaOtpremnice values(3, 2, 10, 2)

insert into StavkaOtpremnice values(4, 2, 30, 2)

insert into StavkaOtpremnice values(5, 3, 20, 3)

insert into StavkaOtpremnice values(6, 3, 30, 3)

insert into Situacija values(1, 1, 110120, 30, '2021-9-1', '2021-11-12', 1)

insert into Situacija values(2, 1, 120530, 70, '2021-11-12', '2022-1-18', 2)

insert into Situacija values(3, 2, 140230, 60, '2021-11-28', '2021-12-31', 3)

insert into Situacija values(4, 2, 100570, 40, '2021-12-31', '2022-1-27', 4)

insert into Situacija values(5, 3, 90410, 55, '2022-1-15', '2022-2-4', 5)

insert into Situacija values(6, 3, 150760, 45, '2022-2-4', '2022-3-4', 6)

insert into GlavniMajstor values(7, 'TIG zavarivanje')

insert into JuniorMajstor(Sifra\_radnika, RadniStaz, NadredjeniMajstor) values(8, 7, 7)

insert into JuniorMajstor(Sifra\_radnika, RadniStaz, NadredjeniMajstor) values(9, 10, 7)

insert into Pomocnik values(10, 'Ugovorom', 7, 8)

insert into Pomocnik values(11, 'Honorarno', 7, 9)

insert into Angazovan values(3, 1)

insert into Angazovan values(2, 2)

insert into Angazovan values(1, 3)

insert into Pro\_PIO values(1, 1)

insert into Pro\_PIO values(2, 2)

insert into Pro\_PIO values(3, 3)

insert into IzvrsavanjeTransporta values(8, 'BG-423-GS')

insert into IzvrsavanjeTransporta values(9, 'BG-678-AD')

insert into IzvrsavanjeTransporta values(10, 'BG-5694-DB')

insert into KoriscenjeSoftvera values(2, 4)

insert into KoriscenjeSoftvera values(1, 5)

insert into KoriscenjeSoftvera values(1, 6)

insert into RadNaStavci values(4, 5, 3)

insert into RadNaStavci values(5, 3, 2)

insert into RadNaStavci values(6, 1, 1)

--UBACIVANJE DODATNIH SPOLJINIH KLJUCEVA

update PIZ

set SifraGrad = 1, SifraPIO = 1, IDPIZRacuna = 1, SifraVodje = 6

where SifraPIZ = 1

update PIZ

set SifraGrad = 2, SifraPIO = 2, IDPIZRacuna = 2, SifraVodje = 5

where SifraPIZ = 2

update PIZ

set SifraGrad = 3, SifraPIO = 3, IDPIZRacuna = 3, SifraVodje = 4

where SifraPIZ = 3

update PIO

set SifraGrad = 1

where SifraPIO = 1

update PIO

set SifraGrad = 2

where SifraPIO = 2

update PIO

set SifraGrad = 3

where SifraPIO = 3

update Majstor

set SifraOtp = 1

where Sifra\_radnika = 8

update Majstor

set SifraOtp = 2

where Sifra\_radnika = 9

update Majstor

set SifraOtp = 3

where Sifra\_radnika = 10

update Otpremnica

set RegVozila = 'BG-423-GS'

where SifraOtp = 1

update Otpremnica

set RegVozila = 'BG-678-AD'

where SifraOtp = 2

update Otpremnica

set RegVozila = 'BG-5694-DB'

where SifraOtp = 3

update Transport

set SifraPN = 1

where RegVozila = 'BG-423-GS'

update Transport

set SifraPN = 2

where RegVozila = 'BG-678-AD'

update Transport

set SifraPN = 3

where RegVozila = 'BG-5694-DB'

update IzlazniRacun

set IDSituacije = 1, SituacijaSifraGrad = 1

where IDRacuna = 1

update IzlazniRacun

set IDSituacije = 2, SituacijaSifraGrad = 1

where IDRacuna = 2

update IzlazniRacun

set IDSituacije = 3, SituacijaSifraGrad = 2

where IDRacuna = 3

update IzlazniRacun

set IDSituacije = 4, SituacijaSifraGrad = 2

where IDRacuna = 4

update IzlazniRacun

set IDSituacije = 5, SituacijaSifraGrad = 3

where IDRacuna = 5

update IzlazniRacun

set IDSituacije = 6, SituacijaSifraGrad = 3

where IDRacuna = 6

update JuniorMajstor

set PodredjeniPom = 10

where Sifra\_radnika = 8

update JuniorMajstor

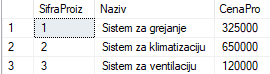
set PodredjeniPom = 11

where Sifra\_radnika = 9

# Tabelarni prikaz sadrzaja tabela



Slika 1. Tabela - Radnik



Slika 2. Tabela - Proizvod



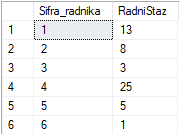
Slika 3. Tabela – Softverski Paket



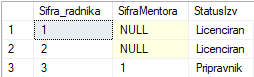
Slika 4. Tabela - Obuka



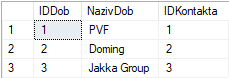
Slika 5. Tabela - Investitor



Slika 6. Tabela - Inzenjer



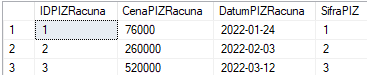
Slika 7. Izvodjac



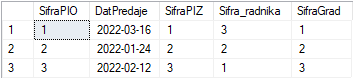
Slika 8. Tabela - Dobavljac



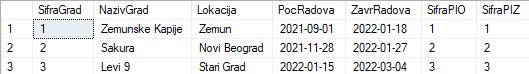
Slika 9. Tabela - PIZ



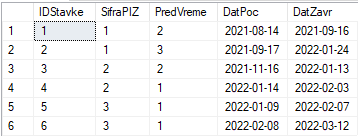
Slika 10. Tabela – Racun za PIZ



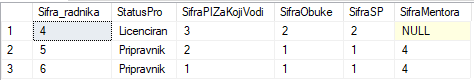
Slika 11. Tabela - PIO



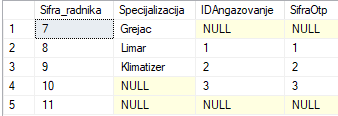
Slika 13. Tabela - Gradiliste



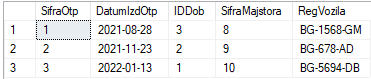
Slika 14. Tabela – Stavka PIZ



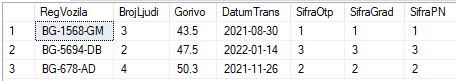
Slika 15. Tabela - Projektant



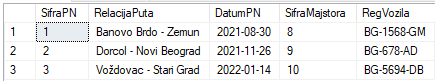
Slika 16. Tabela - Majstor



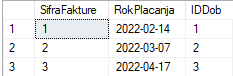
Slika 17. Tabela - Otpremica



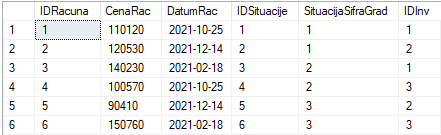
Slika 18. Tabela - Transport



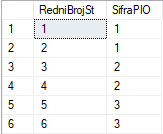
Slika 19. Tabela - PutniNalog



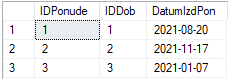
Slika 20. Tabela - UlaznaFaktura



Slika 21. Tabela – Izlazni racun



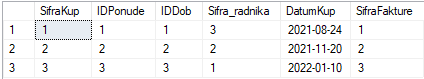
Slika 22. Tabela – Stavka PIO



Slika 23. Tabela – Ponuda



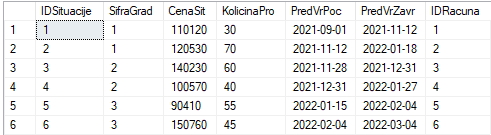
Slika 24. Tabela – Stavka ponude



Slika 25. Tabela – Kupovina



Slika 26. Tabela – Stavka Otpremice



Slika 27. Tabela – Situacija



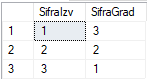
Slika 28. Tabela – Glavni Majstor



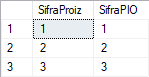
Slika 29. Tabela – Junior Majstor



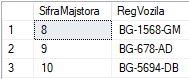
Slika 30. Tabela – Pomocnik



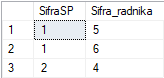
Slika 31. Tabela – Angazovan



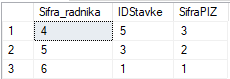
Slika 31. Tabela – Proizvod\_PIO



Slika 31. Tabela – Izvrsavanje Transporta



Slika 32. Tabela – Koriscenje Softvera



Slika 33. Tabela – Rad na stavci

# Prikaz baznih pogleda, procedura i funkcija

U ovom poglavlju, po podsistemima, se daju prikazi pogleda, baznih procedura i funkcija.

## Podsistem Radnik

### Pogled

create view Radnici\_i\_Njihove\_Pozicije

as

select r.Sifra\_radnika, r.ImePrezime, r.Plata, i.RadniStaz 'Radni staz inzenjera', m.Specijalizacija, iz.SifraMentora as 'Izvodjac mentor', iz.StatusIzv as 'Status izvodjaca',

p.SifraMentora as 'Projektant mentor', p.StatusPro as 'Status Projektanta', gm.Sertifikat, jm.RadniStaz as 'Radni staz juniora', jm.NadredjeniMajstor, jm.PodredjeniPom,

po.NacinZap, po.NadredjeniGM, po.NadredjeniJM

from Radnik r full outer join Inzenjer i on r.Sifra\_radnika = i.Sifra\_radnika full outer join Majstor m on r.Sifra\_radnika = m.Sifra\_radnika full outer join Izvodjac iz on iz.Sifra\_radnika = i.Sifra\_radnika

full outer join Projektant p on p.Sifra\_radnika = i.Sifra\_radnika full outer join GlavniMajstor gm on gm.Sifra\_radnika = m.Sifra\_radnika full outer join JuniorMajstor jm on jm.Sifra\_radnika = m.Sifra\_radnika

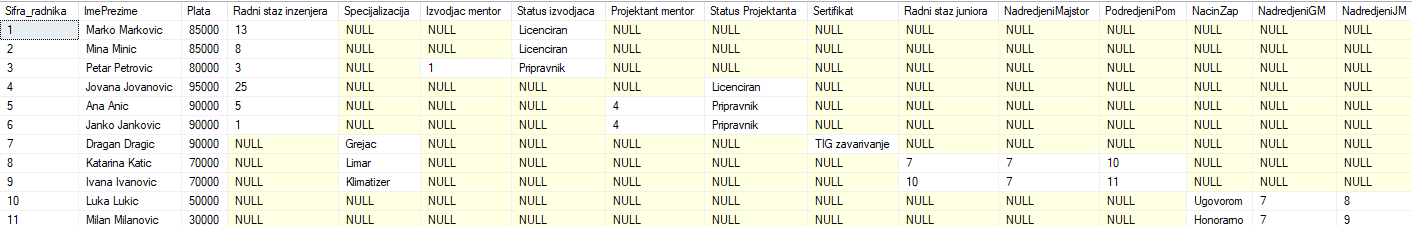
full outer join Pomocnik po on po.Sifra\_radnika = m.Sifra\_radnika

--POZIV POGLEDA:

go

select \* from Radnici\_i\_Njihove\_Pozicije

Sadržaj koji upit vraća dat je u sledećoj tabeli.



### Bazna procedura

-Procedura vezana za celinu “Radnik” služi za zapošljavanje novog izvođača i time unošenje datih podataka u određene tabele:

go

create procedure Zaposljavanje\_Novog\_Izvodjaca

@ID int, @ImePrezime varchar(30), @Plata numeric(6, 0), @RadniStaz int, @Mentor int, @Status varchar(15) as

begin

if @ID in (select Sifra\_radnika from Radnik) or @Mentor not in (select SifraMentora from Izvodjac) or @Status not in ('Licenciran', 'Pripravnik')

begin

print 'Niste pravilno uneli vrednosti novog Izvodjaca'

return -1

end

insert into Radnik values(@ID, @ImePrezime, @Plata)

insert into Inzenjer values(@ID, @RadniStaz)

insert into Izvodjac values(@ID, @Mentor, @Status)

end

--POZIV PROCEDURE:

go

declare @NoviRadnik int

set @NoviRadnik = (select max(Sifra\_radnika) from Radnik) + 1

exec Zaposljavanje\_Novog\_Izvodjaca @ID = @NoviRadnik, @ImePrezime = 'Toma Tomic', @Plata = 82000, @RadniStaz = 3, @Mentor = 1, @Status = 'Pripravnik'

### Bazna funkcija

create function Prikaz\_Promena\_Radnik() returns table as

return

select \* from Radnici\_i\_Njihove\_Pozicije

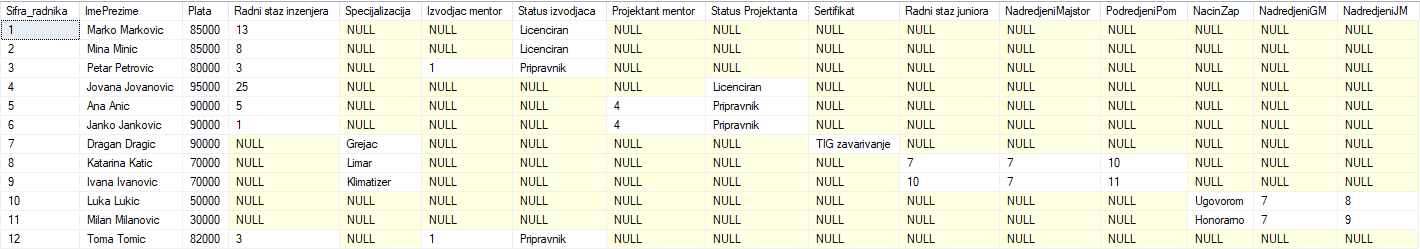
--POZIV FUNKCIJE:

go

select \* from Prikaz\_Promena\_Radnik()

order by Sifra\_radnika

Sadržaj koji upit vraća dat je u sledećoj tabeli.



## Podsistem Projektovanje

### Pogled

create view Projektanti\_na\_Obuci

as

select p.Sifra\_radnika, r.ImePrezime, p.SifraObuke, p.SifraSP, o.VremeTr as 'Vreme trajanja', CenaSoft as 'Cena paketa'

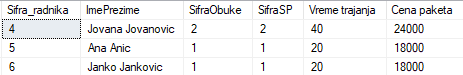
from Projektant p full outer join Obuka o on o.SifraObuke = p.SifraObuke full outer join SoftPaket sp on o.SifraSP = sp.SifraSP join Radnik r on p.Sifra\_radnika = r.Sifra\_radnika

--POZIV POGLEDA:

go

select \* from Projektanti\_na\_Obuci

Sadržaj koji upit vraća dat je u sledećoj tabeli.



### Bazna procedura

-Procedura treba da prebaci projektante na sledeći nivo obuke, projektante na početnom kursu prebaciti na napredni, a oni već na naprednom završavaju sve obuke. Naknadno treba povećati vreme trajanja obuka za deset dana zbog velikog update-a nad softverskim paketima koji im je i povećao cene:

create procedure Napredovanje\_Obuke as

begin

update Projektant

set SifraObuke = null, SifraSP = null

where SifraObuke = 2

update Projektant

set SifraObuke = 2, SifraSP = 2

where SifraObuke = 1

update Obuka

set VremeTr = VremeTr + 10

update SoftPaket

set CenaSoft = CenaSoft + 1000

end

--POZIV PROCEDURE:

go

exec Napredovanje\_Obuke

### Bazna funkcija

create function Pregled\_Promena() returns table as

return

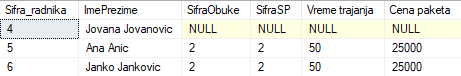
select \* from Projektanti\_na\_Obuci

--POZIV FUNKCIJE:

go

select \* from Pregled\_Promena()

Sadržaj koji upit vraća dat je u sledećoj tabeli.



## Podsistem Izvodjenje

### Pogled

create view Pregled\_Ponuda

as

select k.SifraKup, r.ImePrezime, d.NazivDob as 'Dobavljac', k.IDPonude, sp.RedniBroj as 'Stavka Ponude', sp.SifraProiz as 'Proizvod', pr.Naziv as 'Naziv proizvoda', sp.KolicinaProiz,

pr.CenaPro \* sp.KolicinaProiz as 'Cena', uf.SifraFakture

from Ponuda p join StavkaPonude sp on sp.IDPonude = p.IDPonude join Kupovina k on k.IDPonude = p.IDPonude join UlaznaFaktura uf on uf.SifraFakture = k.SifraFakture

join Dobavljac d on d.IDDob = p.IDDob join Proizvod pr on pr.SifraProiz = sp.SifraProiz join Radnik r on r.Sifra\_radnika = k.Sifra\_radnika

--POZIV POGLEDA:

go

select \* from Pregled\_Ponuda

Sadržaj koji upit vraća dat je u sledećoj tabeli.



### Bazna Procedura

-Procedura izvršava proces pravljenja nove kupovine, Izvođač kontaktira Dobavljača, Dobavljač izdaje ponudu sa svojim stavkama u kojima se nalaze proizvodi, i nakon obavljene kupovine izdaje se Ulazna Faktura:

create procedure Nova\_Kupovina

@BrojStavki int, @Dobavljac int, @Izvodjac int as

begin

declare @MaxPonuda int

set @MaxPonuda = (select max(IDPonude) from Ponuda)

insert into Ponuda values(@MaxPonuda + 1, @Dobavljac, (select cast(getdate() as date)))

declare @MaxStavkaPon int

set @MaxPonuda = (select max(IDPonude) from Ponuda)

declare @KolicinaRobe int

declare @Proizvod int

declare @MaxFaktura int

declare @MaxKupovina int

declare @x int = @BrojStavki

while (@x > 0)

begin

set @x = @x - 1

set @MaxStavkaPon = (select max(RedniBroj) from StavkaPonude)

set @KolicinaRobe = (select cast(floor(rand()\*(5-1+1)+1) as int))

set @Proizvod = (select cast(floor(rand()\*(3-1+1)+1) as int))

insert into StavkaPonude values(@MaxStavkaPon + 1, @MaxPonuda, @Dobavljac, @Proizvod, @KolicinaRobe, (select cast(getdate() + 10 as date)))

end

set @MaxFaktura = (select max(SifraFakture) from UlaznaFaktura)

insert into UlaznaFaktura values(@MaxFaktura + 1, (select cast(getdate() + 30 as date)), @Dobavljac)

set @MaxKupovina = (select max(SifraKup) from Kupovina)

set @MaxPonuda = (select max(IDPonude) from Ponuda)

set @MaxFaktura = (select max(SifraFakture) from UlaznaFaktura)

insert into Kupovina values(@MaxKupovina + 1, @MaxPonuda, @Dobavljac, @Izvodjac, (select cast(getdate() + 20 as date)), @MaxFaktura)

end

--POZIV PROCEDURE:

exec Nova\_Kupovina @BrojStavki = 3, @Dobavljac = 2, @Izvodjac = 2

### Bazna funkcija

create function Pregled\_Promena\_Izvodjenje() returns table as

return

select \* from Pregled\_Ponuda

--POZIV FUNKCIJE:

go

select \* from Pregled\_Promena\_Izvodjenje()

Sadržaj koji upit vraća dat je u sledećoj tabeli.



## Podsistem Nabavka

### Pogled

create view Detalji\_o\_Transportu

as

select izt.RegVozila, izt.SifraMajstora, r.ImePrezime as 'Izvrsilac transporta', t.BrojLjudi, t.Gorivo, t.DatumTrans, pn.RelacijaPuta, pn.SifraPN, o.SifraOtp

from Transport t join PutniNalog pn on pn.RegVozila = t.RegVozila join Otpremnica o on o.RegVozila = t.RegVozila join IzvrsavanjeTransporta izt on izt.RegVozila = t.RegVozila

join Radnik r on r.Sifra\_radnika = pn.SifraMajstora

--POZIV POGLEDA:

go

### select \* from Detalji\_o\_Transportu

Sadržaj koji upit vraća dat je u sledećoj tabeli.

### 

### Bazna Procedura

### -Zbog neplaniranog kvara na jednom od vozila transporta potrebno je napraviti proceduru koja ce angazovati novo vozilo, time azurirati i sve potrebne tabele koje su direktno povezane sa tim transportom:

create procedure Promena\_Transporta

@PokvarenoVozilo varchar(10), @NovoVozilo varchar(10) as

begin

update PutniNalog

set RegVozila = NULL

where RegVozila = @PokvarenoVozilo

update Otpremnica

set RegVozila = NULL

where RegVozila = @PokvarenoVozilo

declare @Majstor int = (select SifraMajstora from IzvrsavanjeTransporta where RegVozila = @PokvarenoVozilo)

delete from IzvrsavanjeTransporta

where RegVozila = @PokvarenoVozilo

declare @BrojLjudi int = (select BrojLjudi from Transport where RegVozila = @PokvarenoVozilo)

declare @Gorivo numeric(4, 1) = (select Gorivo from Transport where RegVozila = @PokvarenoVozilo)

declare @Datum date = (select DatumTrans from Transport where RegVozila = @PokvarenoVozilo)

declare @PutniNalog int = (select SifraPN from PutniNalog where RegVozila is null)

declare @Otpremnica int = (select SifraOtp from Otpremnica where RegVozila is null)

declare @Gradiliste int = (select SifraGrad from Transport where RegVozila = @PokvarenoVozilo)

delete from Transport

where RegVozila = @PokvarenoVozilo

insert into Transport values(@NovoVozilo, @BrojLjudi, @Gorivo, @Datum, @Otpremnica, @Gradiliste, @PutniNalog)

update PutniNalog

set RegVozila = @NovoVozilo

where RegVozila is null

update Otpremnica

set RegVozila = @NovoVozilo

where RegVozila is null

insert into IzvrsavanjeTransporta values(@Majstor, @NovoVozilo)

end

--POZIV PROCEDURE:

### exec Promena\_Transporta @PokvarenoVozilo = 'BG-423-GS', @NovoVozilo = 'BG-1568-GM'

### Bazna Funkcija

create function Promene\_Nabavka() returns table as

return

select \* from Detalji\_o\_Transportu

--POZIV FUNKCIJE:

go

### select \* from Promene\_Nabavka()

Sadržaj koji upit vraća dat je u sledećoj tabeli.

### 

# Literatura

1. Majstorović M.: Baze podataka, predavanja i vežbe za predmet u ppt formi, školska

2021/2022. godina (na studentskom portalu).