



**ITS** INFORMATION  
TECHNOLOGY  
SCHOOL

---

VISOKA ŠKOLA STRUKOVNIH STUDIJA ZA IT

## **BAZE PODATAKA**

Projekat

**Projektovanje baze podataka za podršku rada bioskopa**

Predmetni nastavnik:

\_\_\_\_\_

Studenti:

ime\_prezime\_1 br. ind.

ime\_prezime\_2 br. ind.

ime\_prezime\_3 br. ind.

Datum predaje \_\_\_\_\_

**Beograd**

**mesec, godina**

## Sadržaj

1.	Rezime .....	3
2.	Poslovni domen - OPIS .....	4
3.	Podmodeli podataka .....	5
4.	Integrisani model podataka (PMOV).....	8
5.	Prevodjenje PMOV u relacioni model .....	9
6.	SQL naredbe za kreiranje tabela .....	11
7.	SQL naredbe za umetanje podataka .....	15
8.	Tabelarni prikaz sadržaja tabela .....	20
9.	Prikaz baznih pogleda, procedura i funkcija .....	24
9.1	Podsistem Radnik .....	24
9.1.1	Pogled .....	24
9.1.2	Poslovna transakcija i bazna procedura .....	24
9.1.2.1	Definicija poslovne transakcije .....	24
9.1.2.2	Bazna procedura .....	25
9.1.3	Bazna funkcija .....	26
9.2	Podsistem Film.....	27
9.2.1	Pogled .....	27
9.2.2	Poslovna transakcija i bazna procedura .....	28
9.2.2.1	Definicija poslovne transakcije .....	28
9.2.2.2	Bazna procedura .....	28
9.2.3	Bazna funkcija .....	30
9.3	Podsistem Dobavljač.....	31
9.3.1	Pogled .....	31
9.3.2	Poslovna transakcija i bazna procedura .....	32
9.3.2.1	Definicija poslovne transakcije .....	32
9.3.2.2	Bazna procedura .....	32
9.3.3	Bazna funkcija .....	34
10.	Literatura .....	36

## 1. Rezime

U ovom projektu je izvršena analiza poslovnog domena "Bioskop", za potrebe udruženja producenata. Kreiran je model podataka za ovaj poslovni domen i izvršena njegova implementacija kroz kreiranje šeme relacione baze podataka. Kroz SQL insert naredbe izvršeno je instanciranje ove šeme baze podataka. U projektu je dat i tabelarni prikaz sadržaja tabela baze podataka. Za svaki podsistem definisane su poslovne transakcije, dati pogledi i bazne procedure pomoću kojih se realizuju poslovne transakcije. Za svaki podsistem kreirane su i bazne funkcije, čijim izvršenjem su pokazane realizacije odgovarajućih poslovnih transakcija.

## 2. Poslovni domen - OPIS

Za potrebe udruženja producenstskih kuća potrebno je isprojektovati bazu podataka sa neophodnim podacima. Poslovni domen je dekomponovan na tri celine (podsistema): Radnik, Film i Dobavljač. U nastavku se daje kratak opis ovih celina.

### I Podsystem – Radnik (student ime\_prezime\_1)

Bioskopi se nalazi u vise gradova, na vise lokacija. Grad ima naziv grada i broj bioskopa koji se u njemu nalaze, a lokacija ima šifru lokacije, naziv i adresu. Grad može imati 0 ili više lokacija, a lokacija se mora nalaziti u samo jednom gradu. Konkretni radnik radi na jednoj lokaciji, dok lokacija može da ima više zaposlenih radnika. Svaki radnik može i ne mora da ima nadređenog. Radnik ima šifru, ime, prezime, pol, JMBG, datum rođenja, pol, platu, premiju. Radnik može biti angažovan na više radnih mesta. Za angažovanje je potrebno voditi evidenciju od kada je radnik angažovan na datom radnom mestu i kada se njegovo angažovanje završava. Za radno mesto treba da se vodi evidencija o šifri i nazivu radnog mesta.

Radnik može raditi na određenoj blagajni. Na blagajni može raditi više radnika. Blagajna ima svoju šifru. Kroz račun blagajne evidentiraju se prodane ulaznice na toj blagajni. Račun se daje klijentu koji kupuje ulaznicu (ulaznice). Podaci o klijentu se ne evidentiraju. Račun blagajne ima šifru računa, cenu, količinu (broj kupljenih ulaznica), naziv i adresu bioskopa. Za blagajnu se svaki dan vodi statistika o prodaji ulaznica. Na statistici prodaje se evidentira datum i broj prodanih ulaznica za taj datum, kao i ukupno raspoloživ broj ulaznica za taj datum.

### II Podsystem Film (student ime\_prezime\_2)

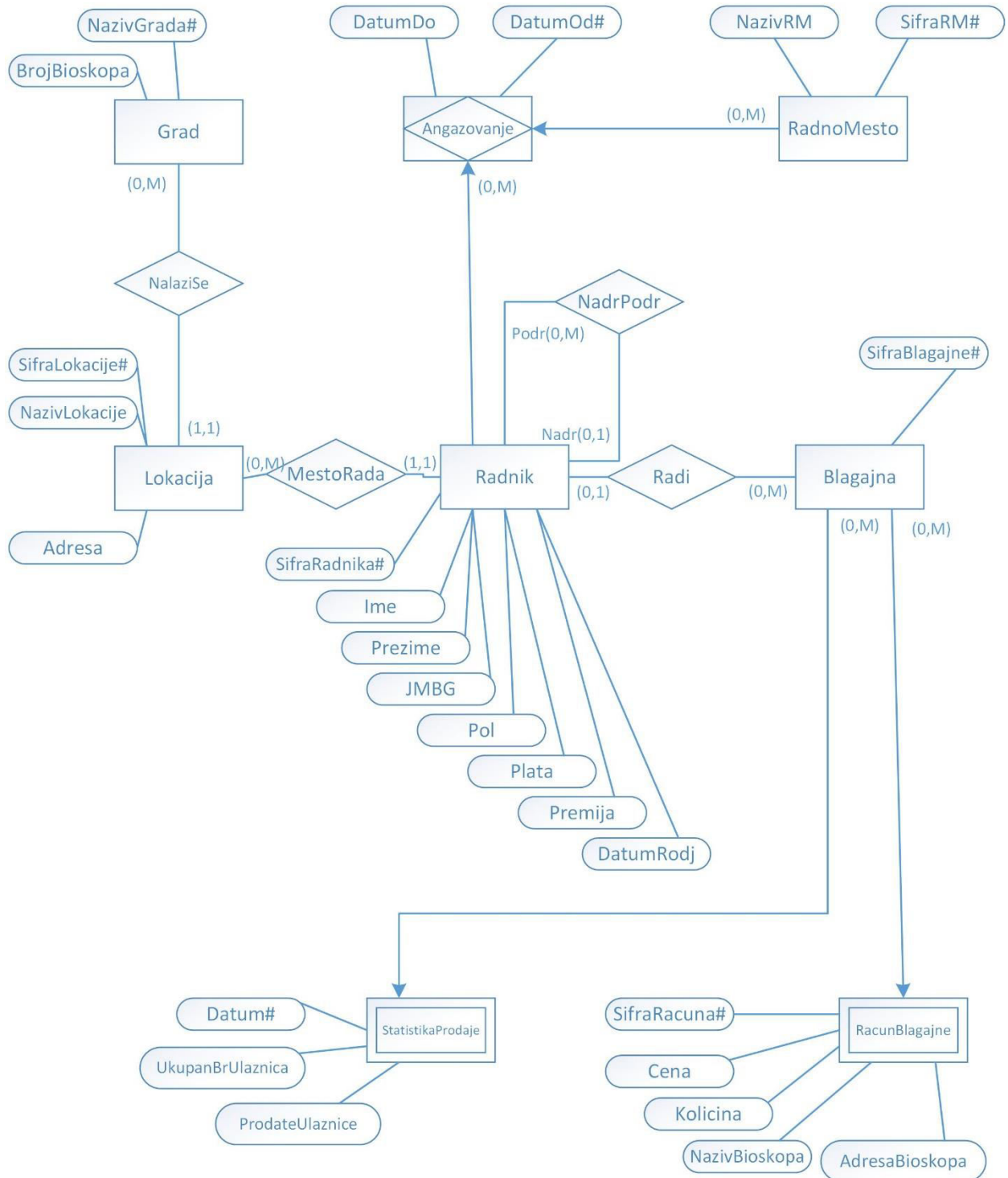
Za svaki film se vode sledeći podaci: šifra filma, naziv, vrsta projekcije, trajanje i godina filma. Za svaki film učestvuje jedan ili više timova. Za tim se vodi evidencija o režiseru, scenaristi, glavnom glumcu, montaži, muzici i produkcijskoj kući. Vodi se repertoar sa filmovima na svim lokacijama u određenom vremenskom periodu. Filmovi se projektuju u više različitih sala. Svaka sala ima svoj broj, svoju šifru, broj redova i broj sedišta. Svaka projekcija ima svoj datum, satnicu i šifru projekcije koja je jednoznačno identifikuje u okviru filma i sale. Na blagajni se prodaju ulaznice za određenu projekciju. Ulaznica ima šifru ulaznice, cenu i oznaku sedišta. Na blagajni se mogu rezervisati ulaznice. Svaka rezervacija sadrži šifru, ime, prezime i telefon osobe koja je rezervisala ulaznicu.

### III Podsystem Dobavljač (student ime\_prezime\_2)

Udruženeje producenstskih kuća evidentira dobavljače od koji se vrši nabavka. Za dobavljača se vode sledeći podaci: šifra dobavljača, naziv i adresa. Roba koja se nabavlja ima šifru, naziv i vrstu robe. Po potrebi, za dobavljača se pravi narudžbenica. Narudžbenica ima broj, datumi i količinu robe koja se naručuje. Kroz jednu narudžbenicu naručuje se samo jedna roba (ali može više komada te robe, što se definiše kroz atribut količina na narudžbenici). Jedna roba se može nalaziti na više narudžbenica. U sistemu se evidentiraju otpremnice dobavljača. Kroz jednu narudžbenicu naručuje se samo jedna roba (ali može više komada te robe, što se definiše kroz atribut količina na narudžbenici). Otpremnica ima broj, datum i količinu robe koja je stigla sa otpremnicom. Na jednoj otpremnici je samo jedna roba (ali može više komada te robe, što se definiše kroz atribut količina na otpremnici). Dobavljač izdaje fakturu po kojoj se vrši isplata. Na fakturi se nalazi broj fakture, iznos bez PDV-a, PDV, iznos sa PDV-om, ukupna vrednost, količina, datum i rok plaćanja. Faktura ima samo jednu robu.

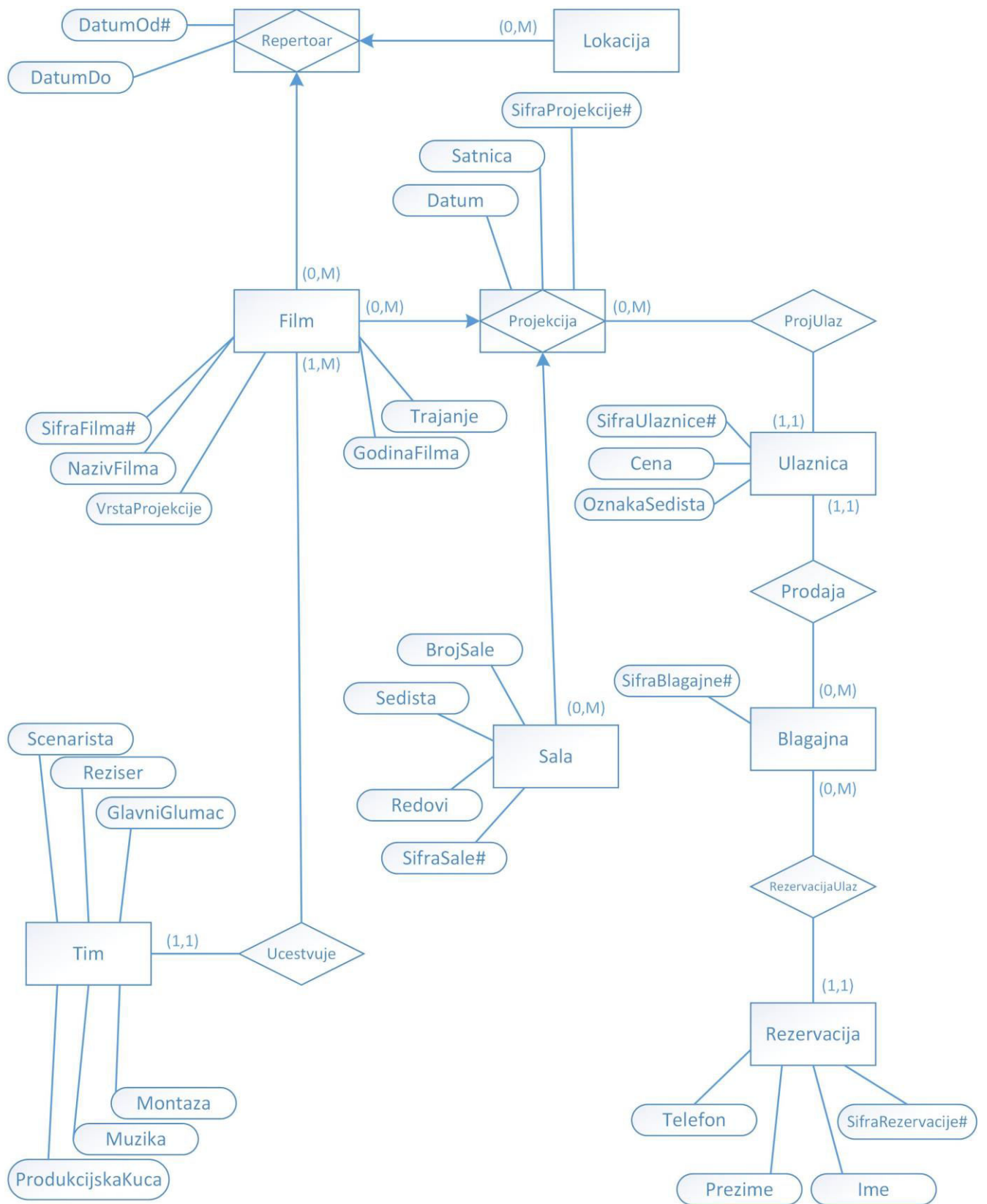
### 3. Podmodeli podataka

U nastavku su prikazani podmodeli podataka, napred opisanih celina, kroz podmodele PMOV.

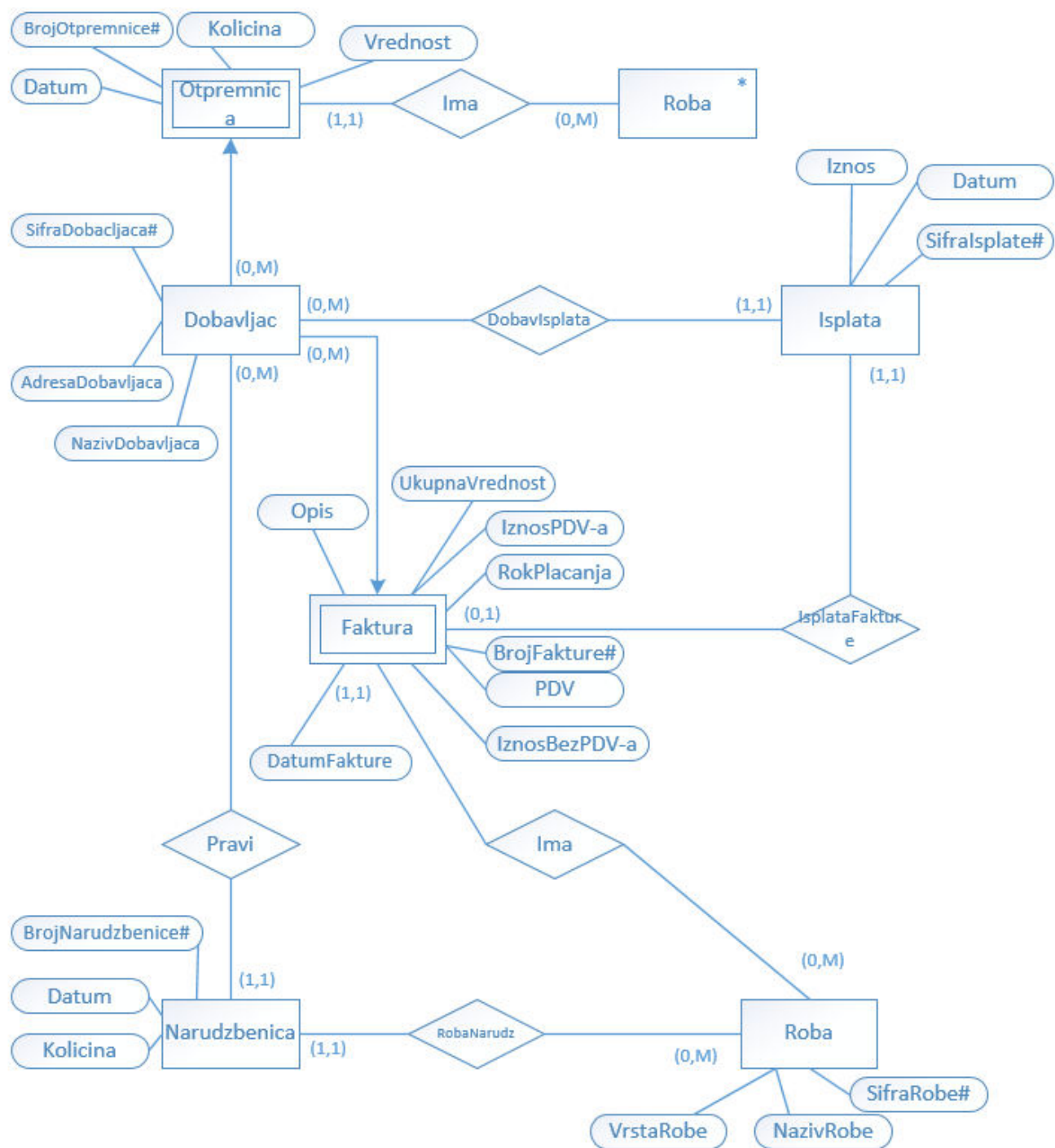


Slika 1. Podmodel PMOV - Radnik

## Projekat Bioskop



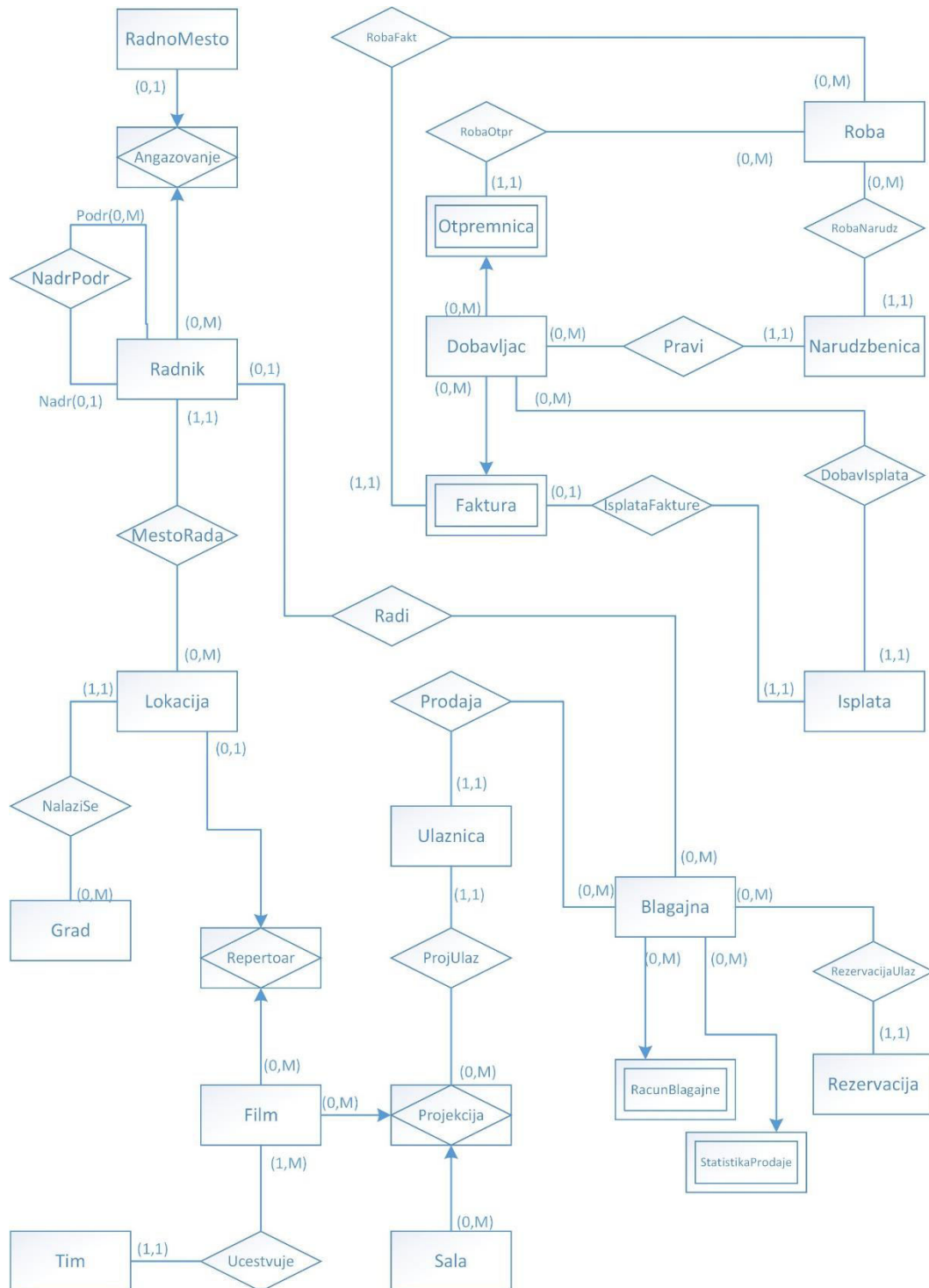
Slika 2. Podmodel PMOV -Film



Slika 3. Podmodel PMOV -Dobavljač

## 4. Integrisani model podataka (PMOV)

Na slici 4 je prikazan integrisani PMOV.



Slika 4. Integrisani PMOV



## 5. Prevodjenje PMOV u relacioni model

**Grad** (NazivGrada#, BrojBioskopa)

**Sala** (SifraSale#, BrojSale, Redovi, Sedista)

**Blagajna** (SifraBlagajne#)

**Dobavljac** (SifraDobavljacka#, AdresaDobavljacka, NazivDobavljacka)

**Roba** (SifraRobe#, NazivRobe, VrstaRobe)

**Rezervacija** (SifraRezervacije, Ime, Prezime, Telefon, *SifraBlagajne#fk*)

Rezervacija (SifraBlagajne#fk) referencira Blagajna (SifraBlagajne#)

**Lokacija** (SifraLokacije#, NazivLokacije, Adresa, *NazivGrada#fk*)

Lokacija (NazivGrada#fk) referencira Grad (NazivGrada#)

**RadnoMesto** (SifraRM#, NazivRM)

**Radnik** (SifraRadnika#, Ime, Prezime, JMBG, DatumRodjenja, Pol, Plata, Premija, *SifraNadredjenog#fk*, *SifraLokacije#fk*, *SifraBlagajne#fk*)

Radnik (SifraLokacije#fk) referencira Lokacija (SifraLokacije#)

Radnik (SifraBlagajne#fk) referencira Blagajna (SifraBlagajne#)

Radnik (SifraNadredjenog#fk) referencira Radnik (SifraRadnika#)

**Angazovanje** (*SifraRM#fk*, *SifraRadnika#fk*, DatumOd#, DatumDo)

Angazovanje (SifraRM#fk) referencira RadnoMesto (SifraRM#)

Angazovanje (SifraRadnika#fk) referencira Radnik (SifraRadnika#)

**RacunBlagajne** (SifraRacuna#, Cena, Kolicina, NazivBioskopa, AdresaBioskopa, *SifraBlagajne#fk*)

RacunBlagajne (SifraBlagajne#fk) referencira Blagajna (SifraBlagajne#)

**StatistikaProdaje** (Datum#, UkupanBrojUlaznica, ProdateUlaznice, *SifraBlagajne#fk*)

StatistikaProdaje (SifraBlagajne#fk) referencira Blagajna (SifraBlagajne#)

**Film** (SifraFilma#, NazivFilma, VrstaProjekcije, Trajanje, Godina)

**Tim** (SifraTima#, Reziser, Scenarista, Montaza, GlavniGlumac, ProdukcijaskaKuca, Muzika, *SifraFilma#fk*)

Tim (SifraFilma#fk) referencira Film (SifraFilma#)

**Repertoar** (*SifraFilma#fk*, *SifraLokacije#fk*, DatumOd#, DatumDo)

Repertoar (SifraFilma#fk) referencira Film (SifraFilma#)

## Projekat Bioskop

Repertoar (SifraLokacije#fk) referencira Lokacija (SifraLokacije#)

**Projekcija** (SifraSale#fk, SifraFilma#fk, SifraProjekcije#, Satnica, Datum)

Projekcija (SifraSale#fk) referencira Sala (SifraSale#)

Projekcija (SifraFilma#fk) referencira Film (SifraFilma#)

**Ulaznica** (SifraUlaznice#, Cena, OznakaSedista, SifraSale#fk, SifraFilma#fk, SifraProjekcije#fk, SifraBlagajne#fk)

Ulaznica (SifraSale#fk, SifraFilma#fk, SifraProjekcije#fk) referencira Projekcija (SifraSale#fk, SifraFilma#fk, SifraProjekcije#)

Ulaznica (SifraBlagajne#fk) referencira Blagajna (SifraBlagajne#)

**Narudzbenica** (BrojNarudzbenice#, Datum, Kolicina, SifraDobavljacka#fk, SifraRobe#fk)

Narudzbenica (SifraDobavljacka#fk) referencira Dobavljack (SifraDobavljacka#)

Narudzbenica (SifraRobe#fk) referencira Roba (SifraRobe#)

**Otpremnica** (BrojOtpremnice#, SifraDobavljacka#fk, Kolicina, Datum, Vrednost, SifraRobe#fk)

Otpremnica (SifraDobavljacka#fk) referencira Dobavljack (SifraDobavljacka#)

Otpremnica (SifraRobe#fk) referencira Roba (SifraRobe#)

**Faktura** (BrojFakture#, SifraDobavljacka#fk, SifraRobe#fk, IznosBezPDVa, PDV, IznosPDVa, UkupnaVrednost, Opis, RokPlacanja, Kolicina, DatumFakture)

Faktura (SifraDobavljacka#fk) referencira Dobavljack (SifraDobavljacka#)

Faktura (SifraRobe#fk) referencira Roba (SifraRobe#)

**Isplata** (SifraIsplate#, Iznos, Datum, BrojFakture#fk, SifraDobavljacka#fk)

Isplata (BrojFakture#fk) referencira Faktura (BrojFakture#)

Isplata (SifraDobavljacka#fk) referencira Dobavljack (SifraDobavljacka#)

## 6. SQL naredbe za kreiranje tabela

```
USE master;
GO
--ako postoji neka stara baza podataka sa imenom PrimerProjekta, uništimo je (DROP) i kreiramo
novu bazu sa tim imenom.
IF EXISTS (SELECT name FROM master.dbo.sysdatabases WHERE name = N'PrimerProjekta')
DROP DATABASE PrimerProjekta;
go
CREATE DATABASE PrimerProjekta;
GO
-- korisnik upotrebljava kreiranu bazu .
USE PrimerProjekta;
GO

CREATE TABLE [RadnoMesto](
    [SifraRM#] [int] NOT NULL,
    [NazivRM] [nvarchar](30) NOT NULL,
    PRIMARY KEY ([SifraRM#]) ) ;

CREATE TABLE [Grad](
    [NazivGrada#] [nvarchar](30) NOT NULL,
    [BrojBioskopa] [int] NOT NULL,
    PRIMARY KEY ([NazivGrada#]) ) ;

CREATE TABLE [Film](
    [SifraFilma#] [int] NOT NULL,
    [NazivFilma] [nvarchar](30) NOT NULL,
    [VrstaProjekcije] [nvarchar](30) NOT NULL,
    [Trajanje] [int] NOT NULL,
    [Godina] [numeric](4, 0) NOT NULL,
    PRIMARY KEY ([SifraFilma#])) ;

CREATE TABLE [Tim](
    [SifraTima#] [int] NOT NULL,
    [Reziser] [nvarchar](50) NOT NULL,
    [Scenarista] [nvarchar](50) NOT NULL,
    [Montaza] [nvarchar](50) NOT NULL,
    [GlavniGlumac] [nvarchar](50) NOT NULL,
    [ProdukcijaskaKuca] [nvarchar](50) NOT NULL,
    [Muzika] [nvarchar](50) NOT NULL,
    [SifraFilma#fk] [int] NOT NULL,
    PRIMARY KEY ([SifraTima#]),
    FOREIGN KEY ([SifraFilma#fk]) REFERENCES [Film] ([SifraFilma#])) ;

CREATE TABLE [Lokacija](
    [SifraLokacije#] [int] NOT NULL,
    [NazivLokacije] [nvarchar](30) NOT NULL,
    [Adresa] [nvarchar](30) NOT NULL,
    [NazivGrada#fk] [nvarchar](30) NOT NULL,
    PRIMARY KEY ([SifraLokacije#]),
    FOREIGN KEY ([NazivGrada#fk]) REFERENCES [Grad] ([NazivGrada#]) ) ;

CREATE TABLE [Repertoar](
    [DatumOd] [date] NOT NULL,
    [DatumDo] [date] NOT NULL,
    [SifraFilma#fk] [int] NOT NULL,
```

## Projekat Bioskop

```
[SifraLokacije#fk] [int] NOT NULL,  
PRIMARY KEY ([SifraFilma#fk], [SifraLokacije#fk], [DatumOd]),  
FOREIGN KEY ([SifraFilma#fk]) REFERENCES [Film] ([SifraFilma#]),  
FOREIGN KEY ([SifraLokacije#fk]) REFERENCES [Lokacija] ([SifraLokacije#])) ;  
  
CREATE TABLE [Sala](  
    [SifraSale#] [int] NOT NULL,  
    [BrojSale] [int] NOT NULL,  
    [Redovi] [int] NOT NULL,  
    [Sedista] [nchar](10) NOT NULL,  
PRIMARY KEY ([SifraSale#]) ) ;  
  
CREATE TABLE [Projekcija](  
    [SifraProjekcije#] [int] NOT NULL,  
    [Satnica] [time](7) NOT NULL,  
    [Datum] [date] NOT NULL,  
    [SifraSale#fk] [int] NOT NULL,  
    [SifraFilma#fk] [int] NOT NULL,  
PRIMARY KEY ([SifraSale#fk], [SifraFilma#fk], [SifraProjekcije#]),  
FOREIGN KEY ([SifraSale#fk]) REFERENCES [Sala] ([SifraSale#]),  
FOREIGN KEY ([SifraFilma#fk]) REFERENCES [Film] ([SifraFilma#]) ) ;  
  
CREATE TABLE [Blagajna](  
    [SifraBlagajne#] [int] NOT NULL,  
PRIMARY KEY ([SifraBlagajne#]) ) ;  
  
CREATE TABLE [StatistikaProdaje](  
    [Datum#] [date] NOT NULL,  
    [UkupanBrojUlaznica] [int] NOT NULL,  
    [ProdateUlaznice] [int] NOT NULL,  
    [SifraBlagajne#fk] [int] NOT NULL,  
PRIMARY KEY ([SifraBlagajne#fk], [Datum#]),  
FOREIGN KEY ([SifraBlagajne#fk]) REFERENCES [Blagajna] ([SifraBlagajne#])) ;  
  
CREATE TABLE [Ulaznica](  
    [SifraUlaznice#] [int] NOT NULL,  
    [Cena] [money] NOT NULL,  
    [OznakaSedista] [nvarchar](50) NOT NULL,  
    [SifraProjekcije#fk] [int] NOT NULL,  
    [SifraSale#fk] [int] NOT NULL,  
    [SifraFilma#fk] [int] NOT NULL,  
    [SifraBlagajne#fk] [int] NOT NULL,  
PRIMARY KEY ([SifraUlaznice#]),  
FOREIGN KEY ([SifraBlagajne#fk]) REFERENCES [Blagajna] ([SifraBlagajne#]),  
FOREIGN KEY ([SifraSale#fk], [SifraFilma#fk], [SifraProjekcije#fk]) REFERENCES [Projekcija]  
([SifraSale#fk], [SifraFilma#fk], [SifraProjekcije#])) ;  
  
CREATE TABLE [Dobavljac](  
    [SifraDobavljacka#] [int] NOT NULL,  
    [AdresaDobavljacka] [nvarchar](50) NOT NULL,  
    [NazivDobavljacka] [nvarchar](50) NOT NULL,  
PRIMARY KEY ([SifraDobavljacka#]) ) ;  
  
CREATE TABLE [Roba](  
    [SifraRobe#] [int] NOT NULL,  
    [NazivRobe] [nvarchar](30) NOT NULL,  
    [VrstaRobe] [nvarchar](30) NOT NULL,  
PRIMARY KEY ([SifraRobe#])) ;
```

## Projekat Bioskop

```
CREATE TABLE [Faktura](
    [BrojFakture#] [int] NOT NULL,
    [IznosBezPDVa] [money] NULL,
    [IznosPDV] [money] NULL,
    [PDV] [float] NULL,
    [UkupnaVrednost] [money] NOT NULL,
    [Opis] [nvarchar](30) NOT NULL,
    [RokPlacanja] [int] NOT NULL,
    [Kolicina] [int] NOT NULL,
    [SifraDobavljacka#fk] [int] NOT NULL,
    [SifraRobe#fk] [int] NOT NULL,
    [DatumFakture][date] NOT NULL,
PRIMARY KEY ([BrojFakture#]),
FOREIGN KEY ([Sifradobavljacka#fk]) REFERENCES [Dobavljac] ([Sifradobavljacka#]),
FOREIGN KEY ([SifraRobe#fk]) REFERENCES [Roba] ([SifraRobe#])) ;

CREATE TABLE [Isplata](
    [SifraIsplate#] [int] NOT NULL,
    [Iznos] [money] NOT NULL,
    [Datum] [date] NOT NULL,
    [BrojFakture#fk] [int] NOT NULL,
    [SifraDobavljacka#fk] [int] NOT NULL,
PRIMARY KEY ([SifraIsplate#]),
FOREIGN KEY ([BrojFakture#fk]) REFERENCES [Faktura] ([BrojFakture#]),
FOREIGN KEY ([SifraDobavljacka#fk]) REFERENCES [Dobavljac] ([SifraDobavljacka#]) ) ;

CREATE TABLE [Otpremnica](
    [BrojOtpremnice#] [int] NOT NULL,
    [Kolicina] [int] NULL,
    [Datum] [date] NOT NULL,
    [Vrednost] [money] NOT NULL,
    [SifraDobavljacka#fk] [int] NOT NULL,
    [SifraRobe#fk] [int] NOT NULL,
PRIMARY KEY ([BrojOtpremnice#]),
FOREIGN KEY ([SifraDobavljacka#fk]) REFERENCES [Dobavljac] ([SifraDobavljacka#]),
FOREIGN KEY ([SifraRobe#fk]) REFERENCES [Roba] ([SifraRobe#]) ) ;

CREATE TABLE [Narudzbenica](
    [BrojNarudzbenice#] [int] NOT NULL,
    [Datum] [date] NOT NULL,
    [Kolicina] [int] NOT NULL,
    [SifraDobavljacka#fk] [int] NOT NULL,
    [SifraRobe#fk] [int] NOT NULL,
PRIMARY KEY ([BrojNarudzbenice#]),
FOREIGN KEY ([SifraDobavljacka#fk]) REFERENCES [Dobavljac] ([SifraDobavljacka#]),
FOREIGN KEY ([SifraRobe#fk]) REFERENCES [Roba] ([SifraRobe#]) ) ;

CREATE TABLE [RacunBlagajne](
    [SifraRacuna#] [int] NOT NULL,
    [Kolicina] [int] NOT NULL,
    [NazivBioskopa] [nvarchar](30) NOT NULL,
    [AdresaBioskopa] [nvarchar](30) NOT NULL,
    [Cena] [money] NOT NULL,
    [SifraBlagajne#fk] [int] NOT NULL,
PRIMARY KEY ([SifraRacuna#], [SifraBlagajne#fk]),
FOREIGN KEY ([SifraBlagajne#fk]) REFERENCES [Blagajna] ([SifraBlagajne#]) ) ;
```

## Projekat Bioskop

```
CREATE TABLE [Rezervacija](
    [SifraRezervacije#] [int] NOT NULL,
    [Ime] [nvarchar](15) NOT NULL,
    [Prezime] [nvarchar](15) NOT NULL,
    [SifraBlagajne#fk] [int] NOT NULL,
    PRIMARY KEY ([SifraRezervacije#]),
    FOREIGN KEY ([SifraBlagajne#fk]) REFERENCES [Blagajna] ([SifraBlagajne#]) ) ;

CREATE TABLE [Radnik](
    [SifraRadnika#] [int] NOT NULL,
    [Ime] [nvarchar](30) NOT NULL,
    [Prezime] [nvarchar](30) NOT NULL,
    [JMBG] [numeric](18, 0) NOT NULL,
    [DatumRodjenja] [date] NOT NULL,
    [Pol] [nvarchar](10) NOT NULL,
    [Plata] [money] NOT NULL,
    [Premija] [money] NULL,
    [SifraLokacije#fk] [int] NOT NULL,
    [SifraNadredjenog#fk] [int] NULL,
    [SifraBlagajne#fk] [int] NOT NULL,
    PRIMARY KEY ([SifraRadnika#]),
    FOREIGN KEY ([SifraLokacije#fk]) REFERENCES [Lokacija] ([SifraLokacije#]),
    FOREIGN KEY ([SifraBlagajne#fk]) REFERENCES [Blagajna] ([SifraBlagajne#]),
    FOREIGN KEY ([SifraNadredjenog#fk]) REFERENCES [Radnik] ([SifraRadnika#]) ) ;

CREATE TABLE [Angazovanje](
    [DatOd#] [date] NOT NULL,
    [DatDo] [date] NULL,
    [SifraRM#fk] [int] NOT NULL,
    [SifraRadnika#fk] [int] NOT NULL,
    PRIMARY KEY ([SifraRadnika#fk], [SifraRM#fk], [DatOd#]),
    FOREIGN KEY ([SifraRM#fk]) REFERENCES [RadnoMesto] ([SifraRM#]),
    FOREIGN KEY ([SifraRadnika#fk]) REFERENCES [Radnik] ([SifraRadnika#]) )
```

## 7. SQL naredbe za umetanje podataka

```
--[RadnoMesto]
INSERT [dbo].[RadnoMesto] ([SifraRM#], [NazivRM]) VALUES (1, N'Prodavac')
INSERT [dbo].[RadnoMesto] ([SifraRM#], [NazivRM]) VALUES (2, N'Kinooperater')
INSERT [dbo].[RadnoMesto] ([SifraRM#], [NazivRM]) VALUES (3, N'Pomocnik')
INSERT [dbo].[RadnoMesto] ([SifraRM#], [NazivRM]) VALUES (4, N'Sef')

--[Grad]
INSERT [dbo].[Grad] ([NazivGrada#], [BrojBioskopa]) VALUES (N'Beograd', 2)

INSERT [dbo].[Grad] ([NazivGrada#], [BrojBioskopa]) VALUES (N'Novi Sad', 1)

-- [Film]
INSERT [dbo].[Film] ([SifraFilma#], [NazivFilma], [VrstaProjekcije], [Trajanje], [Godina])
VALUES (10, N'Fight Club', N'2D', 120, CAST(2016 AS Numeric(18, 0)))

INSERT [dbo].[Film] ([SifraFilma#], [NazivFilma], [VrstaProjekcije], [Trajanje], [Godina])
VALUES (11, N'The Dark Knight', N'3D', 140, CAST(2016 AS Numeric(18, 0)))

INSERT [dbo].[Film] ([SifraFilma#], [NazivFilma], [VrstaProjekcije], [Trajanje], [Godina])
VALUES (12, N'The Revenant', N'2D', 130, CAST(2016 AS Numeric(18, 0)))

-- [Tim]
INSERT [dbo].[Tim] ([SifraTima#], [Reziser], [Scenarista], [Montaza], [GlavniGlumac],
[ProdukcijaskaKuca], [Muzika], [SifraFilma#fk]) VALUES (1, N'David Fincher', N'Tom Smith',
N'Kelly Allen', N'Brad Pitt', N'Fox 2000 Pictures', N'Cary Green', 10)

INSERT [dbo].[Tim] ([SifraTima#], [Reziser], [Scenarista], [Montaza], [GlavniGlumac],
[ProdukcijaskaKuca], [Muzika], [SifraFilma#fk]) VALUES (2, N'Chirstopher Nolan', N'Chirstopher
Nolan', N'Joanna Vega', N'Christian Bale', N'Warner Bros', N'Sam Smith', 11)

INSERT [dbo].[Tim] ([SifraTima#], [Reziser], [Scenarista], [Montaza], [GlavniGlumac],
[ProdukcijaskaKuca], [Muzika], [SifraFilma#fk]) VALUES (3, N'Alejandro Gonzales', N'Mark Smith',
N'Janet Morales', N'Leonardo DiCaprio', N'Fox 2000 Pictures', N'Becky Rhodes', 12)

-- [Lokacija]
INSERT [dbo].[Lokacija] ([SifraLokacije#], [NazivLokacije], [Adresa], [NazivGrada#fk]) VALUES
(77, N'Usce Shopping Centre', N'Bulevar Mihajla Pupina 4', N'Beograd')

INSERT [dbo].[Lokacija] ([SifraLokacije#], [NazivLokacije], [Adresa], [NazivGrada#fk]) VALUES
(88, N'Delta City', N'Jurija Gargarina', N'Beograd')

INSERT [dbo].[Lokacija] ([SifraLokacije#], [NazivLokacije], [Adresa], [NazivGrada#fk]) VALUES
(99, N'TC Sad', N'Bulevar Mihajla Pupina 1', N'Novi Sad')

-- [Repertoar]
INSERT [dbo].[Repertoar] ([DatumOd], [DatumDo], [SifraFilma#fk], [SifraLokacije#fk]) VALUES
(CAST(N'2016-02-10' AS Date), CAST(N'2016-02-11' AS Date), 10, 77)

INSERT [dbo].[Repertoar] ([DatumOd], [DatumDo], [SifraFilma#fk], [SifraLokacije#fk]) VALUES
(CAST(N'2016-02-10' AS Date), CAST(N'2016-02-11' AS Date), 11, 88)

INSERT [dbo].[Repertoar] ([DatumOd], [DatumDo], [SifraFilma#fk], [SifraLokacije#fk]) VALUES
(CAST(N'2016-02-10' AS Date), CAST(N'2016-02-11' AS Date), 12, 99)
```

## Projekat Bioskop

```
--[Sala]
INSERT [dbo].[Sala] ([SifraSale#], [BrojSale], [Redovi], [Sedista]) VALUES (1, 10, 15, N'100
')

INSERT [dbo].[Sala] ([SifraSale#], [BrojSale], [Redovi], [Sedista]) VALUES (2, 11, 20, N'150
')

INSERT [dbo].[Sala] ([SifraSale#], [BrojSale], [Redovi], [Sedista]) VALUES (3, 12, 10, N'70
')

--[Projekcija]
INSERT [dbo].[Projekcija] ([SifraProjekcije#], [Satnica], [Datum], [SifraSale#fk],
[SifraFilma#fk]) VALUES (31, CAST(N'17:00:00' AS Time), CAST(N'2016-02-10' AS Date), 1, 10)

INSERT [dbo].[Projekcija] ([SifraProjekcije#], [Satnica], [Datum], [SifraSale#fk],
[SifraFilma#fk]) VALUES (32, CAST(N'18:00:00' AS Time), CAST(N'2016-02-10' AS Date), 2, 11)

INSERT [dbo].[Projekcija] ([SifraProjekcije#], [Satnica], [Datum], [SifraSale#fk],
[SifraFilma#fk]) VALUES (33, CAST(N'19:00:00' AS Time), CAST(N'2016-02-10' AS Date), 3, 12)

--[Blagajna]
INSERT [dbo].[Blagajna] ([SifraBlagajne#]) VALUES (1111)

INSERT [dbo].[Blagajna] ([SifraBlagajne#]) VALUES (2222)

--[StatistikaProdaje]
INSERT [dbo].[StatistikaProdaje] ([Datum#], [UkupanBrojUlaznica], [ProdateUlaznice],
[SifraBlagajne#fk]) VALUES (CAST(N'2016-02-10' AS Date), 100, 50, 1111)

INSERT [dbo].[StatistikaProdaje] ([Datum#], [UkupanBrojUlaznica], [ProdateUlaznice],
[SifraBlagajne#fk]) VALUES (CAST(N'2016-02-11' AS Date), 50, 45, 2222)

--[Ulaznice]
INSERT [dbo].[Ulaznica] ([SifraUlaznice#], [Cena], [OznakaSedista], [SifraProjekcije#fk],
[SifraBlagajne#fk], [SifraSale#fk], [SifraFilma#fk]) VALUES (111, 500.0000, N'R1-S1', 31, 1111,
1, 10)
INSERT [dbo].[Ulaznica] ([SifraUlaznice#], [Cena], [OznakaSedista], [SifraProjekcije#fk],
[SifraBlagajne#fk], [SifraSale#fk], [SifraFilma#fk]) VALUES (222, 400.0000, N'R2-S2', 32, 2222,
2, 11)
INSERT [dbo].[Ulaznica] ([SifraUlaznice#], [Cena], [OznakaSedista], [SifraProjekcije#fk],
[SifraBlagajne#fk], [SifraSale#fk], [SifraFilma#fk]) VALUES (333, 500.0000, N'R3-S3', 33, 2222,
3, 12)

--[Dobavljac]
INSERT [dbo].[Dobavljac] ([SifraDobavljacka#], [AdresaDobavljacka], [NazivDobavljacka]) VALUES
(11, N'Svetogorska 10', N'Marbo')

INSERT [dbo].[Dobavljac] ([SifraDobavljacka#], [AdresaDobavljacka], [NazivDobavljacka]) VALUES
(22, N'Kneza Milosa', N'Next')

INSERT [dbo].[Dobavljac] ([SifraDobavljacka#], [AdresaDobavljacka], [NazivDobavljacka]) VALUES
(33, N'Visegradska 11', N'Rosa')

-- [Roba]
INSERT [dbo].[Roba] ([SifraRobe#], [NazivRobe], [VrstaRobe]) VALUES (666, N'Cips', N'Hrana')

INSERT [dbo].[Roba] ([SifraRobe#], [NazivRobe], [VrstaRobe]) VALUES (777, N'Kokice', N'Hrana')
INSERT [dbo].[Roba] ([SifraRobe#], [NazivRobe], [VrstaRobe]) VALUES (888, N'Jabuka', N'Pice')
```



## Projekat Bioskop

```
INSERT [dbo].[Roba] ([SifraRobe#], [NazivRobe], [VrstaRobe]) VALUES (999, N'Voda', N'Pice')

-- [Faktura]
INSERT [dbo].[Faktura] ([BrojFakture#], [IznosBezPDVa], [IznosPDV], [PDV], [UkupnaVrednost],
[Opis], [RokPlacanja], [Kolicina], [SifraDobavljacka#fk], [SifraRobe#fk], DatumFakture) VALUES
(11111, 20000.0000, 4000.0000, 20, 24000.0000, N'Rosa', 2, 240, 33, 999, CAST(N'2016-02-02' AS
Date))

INSERT [dbo].[Faktura] ([BrojFakture#], [IznosBezPDVa], [IznosPDV], [PDV], [UkupnaVrednost],
[Opis], [RokPlacanja], [Kolicina], [SifraDobavljacka#fk], [SifraRobe#fk], DatumFakture) VALUES
(22222, 10000.0000, 2000.0000, 20, 12000.0000, N'Cips', 3, 220, 11, 666, CAST(N'2016-02-02' AS
Date))

INSERT [dbo].[Faktura] ([BrojFakture#], [IznosBezPDVa], [IznosPDV], [PDV], [UkupnaVrednost],
[Opis], [RokPlacanja], [Kolicina], [SifraDobavljacka#fk], [SifraRobe#fk], DatumFakture) VALUES
(33333, 10000.0000, 1000.0000, 10, 11000.0000, N'Sok jabuka', 1, 110, 22, 888, CAST(N'2016-02-
02' AS Date))

--[Isplata]
INSERT [dbo].[Isplata] ([SifraIsplate#], [Iznos], [Datum], [BrojFakture#fk],
[SifraDobavljacka#fk]) VALUES (30, 24000.0000, CAST(N'2016-02-01' AS Date), 11111, 33)

INSERT [dbo].[Isplata] ([SifraIsplate#], [Iznos], [Datum], [BrojFakture#fk],
[SifraDobavljacka#fk]) VALUES (40, 12000.0000, CAST(N'2016-02-03' AS Date), 22222, 11)

INSERT [dbo].[Isplata] ([SifraIsplate#], [Iznos], [Datum], [BrojFakture#fk],
[SifraDobavljacka#fk]) VALUES (50, 11000.0000, CAST(N'2016-02-10' AS Date), 33333, 22)

--[Narudzbenica]
INSERT [dbo].[Narudzbenica] ([BrojNarudzbenice#], [Datum], [Kolicina], [SifraDobavljacka#fk],
[SifraRobe#fk]) VALUES (5, CAST(N'2016-02-01' AS Date), 240, 33, 999)

INSERT [dbo].[Narudzbenica] ([BrojNarudzbenice#], [Datum], [Kolicina], [SifraDobavljacka#fk],
[SifraRobe#fk]) VALUES (6, CAST(N'2016-02-01' AS Date), 220, 11, 666)

INSERT [dbo].[Narudzbenica] ([BrojNarudzbenice#], [Datum], [Kolicina], [SifraDobavljacka#fk],
[SifraRobe#fk]) VALUES (7, CAST(N'2016-02-01' AS Date), 110, 22, 888)

-- [Otpremnica]
INSERT [dbo].[Otpremnica] ([BrojOtpremnice#], [Kolicina], [Datum], [Vrednost],
[SifraDobavljacka#fk], [SifraRobe#fk]) VALUES (1, 240, CAST(N'2016-02-02' AS Date), 24000.0000,
33, 999)

INSERT [dbo].[Otpremnica] ([BrojOtpremnice#], [Kolicina], [Datum], [Vrednost],
[SifraDobavljacka#fk], [SifraRobe#fk]) VALUES (2, 220, CAST(N'2016-02-02' AS Date), 12000.0000,
11, 666)

INSERT [dbo].[Otpremnica] ([BrojOtpremnice#], [Kolicina], [Datum], [Vrednost],
[SifraDobavljacka#fk], [SifraRobe#fk]) VALUES (3, 110, CAST(N'2016-02-02' AS Date), 11000.0000,
22, 888)

-- [Rezervacija]
INSERT [dbo].[Rezervacija] ([SifraRezervacije#], [Ime], [Prezime], [SifraBlagajne#fk]) VALUES
(1, N'Ivana', N'Nikolic', 1111)
INSERT [dbo].[Rezervacija] ([SifraRezervacije#], [Ime], [Prezime], [SifraBlagajne#fk]) VALUES
(2, N'Milos', N'Radovic', 2222)
INSERT [dbo].[Rezervacija] ([SifraRezervacije#], [Ime], [Prezime], [SifraBlagajne#fk]) VALUES
(3, N'Stefan', N'Markovic', 1111)
```

## Projekat Bioskop

```
-- [Radnik]
INSERT [dbo].[Radnik] ([SifraRadnika#], [Ime], [Prezime], [JMBG], [DatumRodjenja], [Pol],
[Plata], [Premija], [SifraLokacije#fk], [SifraNadredjenog#fk], [SifraBlagajne#fk]) VALUES (1,
N'Marko', N'Markovic', CAST(12344556 AS Numeric(18, 0)), CAST(N'1989-12-11' AS Date), N'Muski',
70000.0000, 5000.0000, 77, NULL, 1111)

INSERT [dbo].[Radnik] ([SifraRadnika#], [Ime], [Prezime], [JMBG], [DatumRodjenja], [Pol],
[Plata], [Premija], [SifraLokacije#fk], [SifraNadredjenog#fk], [SifraBlagajne#fk]) VALUES (2,
N'Petar', N'Peric', CAST(53464988 AS Numeric(18, 0)), CAST(N'1970-01-01' AS Date), N'Muski',
30000.0000, NULL, 77, 1, 2222)

INSERT [dbo].[Radnik] ([SifraRadnika#], [Ime], [Prezime], [JMBG], [DatumRodjenja], [Pol],
[Plata], [Premija], [SifraLokacije#fk], [SifraNadredjenog#fk], [SifraBlagajne#fk]) VALUES (3,
N'Jelena', N'Simic', CAST(23456778 AS Numeric(18, 0)), CAST(N'1976-11-11' AS Date), N'Zenski',
40000.0000, NULL, 88, 1, 1111)

INSERT [dbo].[Radnik] ([SifraRadnika#], [Ime], [Prezime], [JMBG], [DatumRodjenja], [Pol],
[Plata], [Premija], [SifraLokacije#fk], [SifraNadredjenog#fk], [SifraBlagajne#fk]) VALUES (4,
N'Milica', N'Ivanovic', CAST(24456788 AS Numeric(18, 0)), CAST(N'1969-07-12' AS Date),
N'Zenski', 20000.0000, NULL, 99, 1, 1111)

INSERT [dbo].[Radnik] ([SifraRadnika#], [Ime], [Prezime], [JMBG], [DatumRodjenja], [Pol],
[Plata], [Premija], [SifraLokacije#fk], [SifraNadredjenog#fk], [SifraBlagajne#fk]) VALUES (5,
N'Ana', N'Simovic', CAST(46775325 AS Numeric(18, 0)), CAST(N'1986-04-02' AS Date), N'Zenski',
45000.0000, 2000.0000, 99, 1, 2222)

INSERT [dbo].[Radnik] ([SifraRadnika#], [Ime], [Prezime], [JMBG], [DatumRodjenja], [Pol],
[Plata], [Premija], [SifraLokacije#fk], [SifraNadredjenog#fk], [SifraBlagajne#fk]) VALUES (6,
N'Goran', N'Nikolic', CAST(45678955 AS Numeric(18, 0)), CAST(N'1961-05-04' AS Date), N'Muski',
30000.0000, 1000.0000, 88, 1, 1111)

INSERT [dbo].[Radnik] ([SifraRadnika#], [Ime], [Prezime], [JMBG], [DatumRodjenja], [Pol],
[Plata], [Premija], [SifraLokacije#fk], [SifraNadredjenog#fk], [SifraBlagajne#fk]) VALUES (7,
N'Ivan', N'Matic', CAST(89695954 AS Numeric(18, 0)), CAST(N'1988-07-06' AS Date), N'Muski',
32000.0000, NULL, 77, 1, 2222)

-- [Angazovanje]
INSERT [dbo].[Angazovanje] ([DatOd#], [DatDo], [SifraRM#fk], [SifraRadnika#fk]) VALUES
(CAST(N'2015-10-12' AS Date), NULL, 1, 1)

INSERT [dbo].[Angazovanje] ([DatOd#], [DatDo], [SifraRM#fk], [SifraRadnika#fk]) VALUES
(CAST(N'2016-01-01' AS Date), NULL, 1, 2)

INSERT [dbo].[Angazovanje] ([DatOd#], [DatDo], [SifraRM#fk], [SifraRadnika#fk]) VALUES
(CAST(N'2016-02-01' AS Date), NULL, 2, 3)

INSERT [dbo].[Angazovanje] ([DatOd#], [DatDo], [SifraRM#fk], [SifraRadnika#fk]) VALUES
(CAST(N'2015-11-11' AS Date), CAST(N'2015-12-12' AS Date), 3, 4)

INSERT [dbo].[Angazovanje] ([DatOd#], [DatDo], [SifraRM#fk], [SifraRadnika#fk]) VALUES
(CAST(N'2015-08-07' AS Date), CAST(N'2015-12-31' AS Date), 4, 5)

-- [RacunBlagajne]
INSERT [dbo].[RacunBlagajne] ([SifraRacuna#], [Kolicina], [NazivBioskopa], [AdresaBioskopa],
[Cena], [SifraBlagajne#fk]) VALUES (1, 1, N'Cineplexx', N'Bulevar Mihajla Pupina 4', 100.0000,
1111)
```

## Projekat Bioskop

```
INSERT [dbo].[RacunBlagajne] ([SifraRacuna#], [Kolicina], [NazivBioskopa], [AdresaBioskopa],  
[Cena], [SifraBlagajne#fk]) VALUES (2, 2, N'Cineplex', N'Bulevar Mihajla Pupina 4', 500.0000,  
2222)
```

## 8. Tabelarni prikaz sadržaja tabela

DatOd#	DatDo	SifraRM#fk	SifraRadnika#fk
2015-10-12	NULL	1	1
2016-01-01	NULL	1	2
2016-02-01	NULL	2	3
2015-11-11	2015-12-12	3	4
2015-08-07	2015-12-31	4	5

Slika 1. Tabela - Angazovanje

SifraBlagajne#
1111
2222

Slika 2. Tabela - Blagajna

SifraDobavljac...	AdresaDobavlja...	NazivDobavljacka
11	Svetogorska 10	Marbo
22	Kneza Milosa	Next
33	Visegradska 11	Rosa

Slika 3. Tabela - Dobavljac

BrojFakture#	IznosBezPDVa	IznosPDV	PDV	UkupnaVredn	Opis	RokPlacanja	Kolicina	SifraDobavljac...	SifraRobe#fk	DatumFakture
11111	20000.0000	4000.0000	20	24000.0000	Rosa	2	240	33	999	2016-02-02
22222	10000.0000	2000.0000	20	12000.0000	Cips	3	220	11	666	2016-02-02
33333	10000.0000	1000.0000	10	11000.0000	Sok jabuka	1	110	22	888	2016-02-02

Slika 4. Tabela - Faktura

SifraFilma#	NazivFilma	VrstaProjekcije	Trajanje	Godina
10	Fight Club	2D	120	2016
11	The Dark Knight	3D	140	2016
12	The Revenant	2D	130	2016

Slika 5. Tabela - Film

## Projekat Bioskop

	NazivGrada#	BrojBioskopa
►	Beograd	2
	Novi Sad	1

Slika 6. Tabela - Grad

	Sifralplate#	Iznos	Datum	BrojFakture#fk	SifraDobavljac...
►	30	24000.0000	2016-02-01	11111	33
	40	12000.0000	2016-02-03	22222	11
	50	11000.0000	2016-02-10	33333	22

Slika 7. Tabela-Isplata

	SifraLokacije#	NazivLokacije	Adresa	NazivGrada#fk
►	77	Usce Shopping ...	Bulevar Mihajla...	Beograd
	88	Delta City	Jurija Gargarina	Beograd
	99	TC Sad	Bulevar Mihajla...	Novi Sad

Slika 8. Tabela - Lokacija

BrojOtpremnice#	Kolicina	Datum	Vrednost	SifraDobavljacka#fk	SifraRobe#fk
1	240	2016-02-02	24000.00	33	999
2	220	2016-02-02	12000.00	11	666
3	110	2016-02-02	11000.00	22	888

Slika 9. Tabela - Otpremnica

BrojNarudzenice#	Datum	Kolicina	SifraDobavljacka#fk	SifraRobe#fk
5	2016-02-01	240	33	999
6	2016-02-01	220	11	666
7	2016-02-01	110	22	888

Slika 10. Tabela - Narudzenica

	SifraProjekcije#	Satnica	Datum	SifraSale#fk	SifraFilma#fk
►	31	17:00:00	2016-02-10	1	10
	32	18:00:00	2016-02-10	2	11
	33	19:00:00	2016-02-10	3	12

Slika 11. Tabela - Projekcija

## Projekat Bioskop

	SifraRacuna#	Kolicina	NazivBioskopa	AdresaBioskopa	Cena	SifraBlagajne#fk
►	1	1	Cineplexx	Bulevar Mihajla...	100.0000	1111
	2	2	Cineplex	Bulevar Mihajla...	500.0000	2222

Slika 13. Tabela - RacunBlagajne

SifraRadnika#	Ime	Prezime	JMBG	DatumRodjenja	Pol	Plata	Premija	SifraLokacije#fk	SifraNadredjenog#fk	SifraBlagajne#fk
1	Marko	Markovic	12344556	1989-12-11	Muski	70000.00	5000.00	77	NULL	1111
2	Petar	Peric	53464988	1970-01-01	Muski	30000.00	NULL	77	1	2222
3	Jelena	Simic	23456778	1976-11-11	Zenski	40000.00	NULL	88	1	1111
4	Milica	Ivanovic	24456788	1969-07-12	Zenski	20000.00	NULL	99	1	1111
5	Ana	Simovic	46775325	1986-04-02	Zenski	45000.00	2000.00	99	1	2222
6	Goran	Nikolic	45678955	1961-05-04	Muski	30000.00	1000.00	88	1	1111
7	Ivan	Matic	89695954	1988-07-06	Muski	32000.00	NULL	77	1	2222

Slika 14. Tabela - Radnik

	SifraRM#	NazivRM
►	1	Prodavac
	2	Kinooperater
	3	Pomocnik
	4	Sef

Slika 15. Tabela - RadnoMesto

	DatumOd	DatumDo	SifraFilma#fk	SifraLokacije#fk
►	2016-02-10	2016-02-11	10	77
	2016-02-10	2016-02-11	11	88
	2016-02-10	2016-02-11	12	99

Slika 16. Tabela - Repertoar

	SifraRezervacij...	Ime	Prezime	SifraBlagajne#fk
►	1	Ivana	Nikolic	1111
	2	Milos	Radovic	2222
	3	Stefan	Markovic	1111

Slika 17. Tabela - Rezervacija

	SifraRobe#	NazivRobe	VrstaRobe
▶	666	Cips	Hrana
	777	Kokice	Hrana
	888	Jabuka	Pice
	999	Voda	Pice

Slika 18. Tabela - Roba

	SifraSale#	BrojSale	Redovi	Sedista
▶	1	10	15	100
	2	11	20	150
	3	12	10	70

Slika 19. Tabela - Sala

	Datum#	UkupanBrojUl...	ProdateUlaznice	SifraBlagajne#fk
▶	2016-02-10	100	50	1111
	2016-02-11	50	45	2222

Slika 20. Tabela - StatistikaProdaje

	Reziser	Scenarista	Montaza	GlavniGlumac	ProdukcijaskaK...	Muzika	SifraFilma#
▶	David Fincher	Tom Smith	Kelly Allen	Brad Pitt	Fox 2000 Pictures	Cary Green	10
	Christopher No...	Christopher No...	Joanna Vega	Christian Bale	Warner Bros	Sam Smith	11
	Alejandro Gonz...	Mark Smith	Janet Morales	Leonardo DiCa...	Fox 2000 Pictures	Becky Rhodes	12

Slika 21. Tabela - Tim

SifraUlaznice#	Cena	OznakaSedista	SifraProjekcije#fk	SifraSale#fk	SifraFilma#fk	SifraBlagajne#fk
111	500.00	R1-S1	31	1	10	1111
222	400.00	R2-S2	32	2	11	2222
333	500.00	R3-S3	33	3	12	2222

Slika 22. Tabela - Ulaznica

## 9. Prikaz baznih pogleda, procedura i funkcija

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

### 9.1 Podsistem Radnik

#### 9.1.1 Pogled

Potrebno je kreirati pogled pRadnik\_Lokacija\_Grad\_Angazovanje (prefiks p za pogled) kroz koji će se moći videti podaci o radniku, nazivu lokacije gde radnik živi, kao i nazivu grada lokacije, kao i podatke o angažovanjima radnika. Od podataka radnika, progled treba da prikazuje sledeće: šifru radnika, ime, prezime, ukupnu zaradu (plata i premija).

```
USE PrimerProjekta;
GO
```

```
CREATE VIEW pRadnik_Lokacija_Grad_Angazovanje (SifraRadnika#, Ime, Prezime, Zarada,
    NazivLokacije, NazivGrada#, sifraRM#fk, datOd#, datDo )
AS
SELECT r.SifraRadnika#, r.Ime, r.Prezime, r.Plata + isNull(r.Premija, 0) as Zarada,
l.NazivLokacije, g.NazivGrada#, a.SifraRM#fk, a.DatOd#, a.DatDo
FROM Radnik r JOIN Lokacija l ON r.SifraLokacije#fk = l.SifraLokacije# JOIN Grad g ON
l.NazivGrada#fk = g.NazivGrada#
left JOIN Angazovanje a on a.SifraRadnika#fk = r.SifraRadnika#;
```

```
-- Upit nad pogledom
```

```
SELECT *
FROM pRadnik_Lokacija_Grad_Angazovanje;
```

Sadržaj koji upit vraća dat je u sledećoj tabeli.

SifraRadnika#	Ime	Prezime	Zarada	NazivLokacije	NazivGrada#	sifraRM#fk	datOd#	datDo
1	Marko	Markovic	75000.00	Usce Shopping Centre	Beograd	1	2015-10-12	NULL
2	Petar	Peric	30000.00	Usce Shopping Centre	Beograd	1	2016-01-01	NULL
3	Jelena	Simic	40000.00	Delta City	Beograd	2	2016-02-01	NULL
4	Milica	Ivanovic	20000.00	TC Sad	Novi Sad	3	2015-11-11	2015-12-12
5	Ana	Simovic	47000.00	TC Sad	Novi Sad	4	2015-08-07	2015-12-31
6	Goran	Nikolic	31000.00	Delta City	Beograd	NULL	NULL	NULL
7	Ivan	Matic	32000.00	Usce Shopping Centre	Beograd	NULL	NULL	NULL

#### 9.1.2 Poslovna transakcija i bazna procedura

##### 9.1.2.1 Definicija poslovne transakcije

Udruženje producenstskih kuća ima potrebu da povremno, u skladu sa poslovnom politikom, otvori novo radno mesto i da u određenom gradu rasporedi radnika/radnike (čija lokacija rada pripada tom gradu) na novo otvoreno radno mesto. Na novo radno mesto se raspoređuje radnik/radnici, na sledeći način:



- Svi radnici bez tekućeg angazovanja (datDo je prazno) iz tog grada dobijaju angažovanje na novom radnom mestu. Početak angažovanja je tekući datum.
- Ako nema radnika bez tekućeg angažovanja u tom gradu, onda se radnik (iz tog grada) koji je najduže u tekućem angažovanju premešta na novo radno mesto (ako ih ima više sa istim početkom tekućeg angažovanja, proizvoljno izabrati jednog). Postojeće angažovanje treba zatvoriti sa tekućim datumom, a novo angažovanje otpočeti sa danom posle tekućeg datuma.

#### 9.1.2.2 Bazna procedura

Napraviti baznu proceduru `kdpOtvaranjeRM` (prefiks `kdp` - korisnički definisana procedura), koja će kao argument prihvatiti naziv novog radnog mesta i naziv grada u kojem treba rasporediti radnika/radnike čija lokacija rada je upravo u tom gradu. Šifra novog radnog mesta treba da je za jedan veća od do tada najveće šifre radnog mesta. Formirati novo angažovanje (ili angažovanja, ako se raspoređuje više radnika) u skladu sa definicijom poslovne transakcije datom ranije, u ovom odeljku.

U nastavku sledi programski kod za kreiranje bazne procedure.

USE PrimerProjekta;

GO

```
CREATE PROCEDURE kdpOtvaranjeRM
    @pNazivRM nvarchar(30), @pNazivGrada nvarchar(30)
AS
DECLARE @lSifraNovoRM as int, @lBrojUbacenihAngazovanja as int = 0, @lSifraRadnika as int,
        @lSifraRM as int, @lDatOd as date
BEGIN TRY
    SET XACT_ABORT ON -- U slučaju pojave greške automatsko sprečavanje daljeg izvršavanja niza naredbi
    SELECT @lSifraNovoRM = isNull(max(SifraRM#), 0) + 1
    FROM RadnoMesto;
    BEGIN TRANSACTION
    -- Ubacivanje novog radnog mesta
    INSERT INTO RadnoMesto(SifraRM#, NazivRM)
        VALUES (@lSifraNovoRM, @pNazivRM);

    -- Ubacivanje angazovanja na novo radno mesto, trenutno neangažovanih radnika
    INSERT INTO Angazovanje (SifraRadnika#fk, SifraRM#fk, DatOd#)
    SELECT DISTINCT p.SifraRadnika#, @lSifraNovoRM, getDate()
    FROM pRadnik_Lokacija_Grad_Angazovanje p
    WHERE (p.sifraRM#fk is Null OR p.datDo is Not null) AND
        UPPER(p.NazivGrada#) = UPPER(@pNazivGrada);

    -- Određivanje broja novo ubačenih angazovanja
    SELECT @lBrojUbacenihAngazovanja = count(*)
    FROM Angazovanje
    WHERE SifraRM#fk = @lSifraNovoRM ;

    --Testiranje da li je došlo do ubacivanja angazovanja
    IF @lBrojUbacenihAngazovanja = 0
    BEGIN
        -- Traži se angažovanje koje treba zatvoriti
        SELECT top(1) @lSifraRadnika = p.SifraRadnika#, @lSifraRM = p.sifraRM#fk,
            @lDatOd = p.datOd#
        FROM pRadnik_Lokacija_Grad_Angazovanje p
```

```

WHERE (p.datDo is null AND p.datOd# is not Null)
AND UPPER(p.NazivGrada#) = UPPER(@pNazivGrada)
ORDER BY p.datOd#;
-- Testiranje da li se našlo angažovanje koje treba zatvoriti
IF @lsifraRadnika is not null
BEGIN
    -- Zatvara se nađeno angažovanje
    UPDATE Angazovanje
    SET DatDo = getDate()
    WHERE SifraRadnika#fk = @lsifraRadnika AND SifraRM#fk = @lsifraRM
    AND DatOd# = @lDatOd;

    -- Ubacuje se angažovanje na novo radno mesto
    INSERT INTO Angazovanje (SifraRadnika#fk, SifraRM#fk, DatOd#)
    VALUES (@lsifraRadnika, @lsifraNovoRM, getDate()+1);

    SET @lBrojUbacenihAngazovanja = 1
END;
END;
IF @lBrojUbacenihAngazovanja = 0
BEGIN
    PRINT 'Nije uspelo ubacivanje angažovanja.
    Proveriti postoje li radnici u željenom gradu.';
    PRINT 'Transakcija je poništila promene.'
    ROLLBACK TRANSACTION;
END
ELSE
BEGIN
    PRINT 'Ubačeno je ' + str(@lBrojUbacenihAngazovanja) + ' angažovanj(e/a).'
    COMMIT TRANSACTION;
END;
END TRY
BEGIN CATCH
    PRINT 'DOŠLO JE DO POJAVE GREŠKE!'
    PRINT '----- Proverite ulazne parametre-----'
    ROLLBACK TRANSACTION;
    PRINT 'Transakcija je poništila promene.'
END CATCH;

-- Prvo pozivanje bazne procedure
EXEC kdpOtvaranjeRM @pNazivRM = 'Blagajnik', @pNazivGrada = 'Beograd';

-- Drugo pozivanje bazne procedure
EXEC kdpOtvaranjeRM @pNazivRM = 'Blagajnik', @pNazivGrada = 'Niš';

```

### 9.1.3 Bazna funkcija

Potrebno je kreirati baznu funkciju kdfRadnikAngazovanje (prefiks kdf - korisnički definisana funkcija) koja će kao argument prihvatiti naziv grada a vratiti u tabeli šifru, ime i prezime radnika čija lokacija rada je u tom gradu, stim da imaju, ili su imali angažovanje. Funkcija u tabeli treba da vrati i naziv grada, naziv lokacije, datum početka i datum završetka angažovanja, kao i naziv radnog mesta na kojem je radnik angažovan.

U nastavku sledi programski kod za kreiranje bazne funkcije. Kao što se vidi, funkcija koristi prethodno definisani pogled Radnik\_Lokacija\_Grad\_Angazovanje.

## Projekat Bioskop

```
USE PrimerProjekta;  
GO
```

```
CREATE FUNCTION kdfRadnikAngazovanje (@pNazivGrada nVarChar(30))  
RETURNS Table  
AS  
RETURN SELECT p.SifraRadnika#, p.Ime, p.Prezime, p.NazivGrada#, p.NazivLokacije, rm.NazivRM,  
p.datOd#, p.datDo  
FROM Radnik_Lokacija_Grad_Angazovanje p JOIN RadnoMesto rm  
ON p.sifraRM#fk = rm.SifraRM#  
WHERE UPPER(p.NazivGrada#) = UPPER(@pNazivGrada);
```

Primer pozivanja funkcije sledi u nastavku.

```
-- Pozivanje funkcije  
SELECT *  
FROM kdfRadnikAngazovanje ('Beograd');
```

Funkcija vraća vrednost u tabeli, i ta tabela je prikazana u nastavku. Kao što se vidi, u Beogradu, na radnom mestu blagajnika trenutno su angažovani radnici sa šifrom 6 i 7. Ranije je viđeno, u upitu nad pogledom, da ovi radnici nisu imali angažovanje. Sadašnje njihovo angažovanje je rezultat pozivanja bazne procedure `kdpOtvaranjeRM`. Ovim je pokazano da je bazna procedura izvršila željenu poslovnu transakciju (ubačeno je novo radno mesto, i raspoređeni su odgovarajući radnici nanjega).

SifraRadnika#	Ime	Prezime	NazivGrada#	NazivLokacije	NazivRM	datOd#	datDo
1	Marko	Markovic	Beograd	Usce Shopping Centre	Prodavac	2015-10-12	NULL
2	Petar	Peric	Beograd	Usce Shopping Centre	Prodavac	2016-01-01	NULL
3	Jelena	Simic	Beograd	Delta City	Kinooperater	2016-02-01	NULL
6	Goran	Nikolic	Beograd	Delta City	Blagajnik	2019-10-04	NULL
7	Ivan	Matic	Beograd	Usce Shopping Centre	Blagajnik	2019-10-04	NULL

## 9.2 Podsystem Film

### 9.2.1 Pogled

Potrebno je kreirati pogled `pFilm_Sala_Projekcija_Ulaznica` (prefiks `p` za pogled) kroz koji će se moći videti podaci o filmu (šifra filma, naziv filma), projekcijama filma (šifri sale, datum, satnica), kao i samim ulaznicama (cena, oznaka sedišta).

```
USE PrimerProjekta;  
GO
```

```
CREATE VIEW pFilm_Sala_Projekcija_Ulaznica(SifraFilma#, NazivFilma, SifraSale#, Datum, Satnica,  
Cena, OznakaSedista)  
AS  
SELECT f.SifraFilma#, f.NazivFilma, p.SifraSale#fk, p.Datum, p.Satnica, u.Cena, u.OznakaSedista  
FROM Film f JOIN Projekcija p ON f.SifraFilma# = p.SifraFilma#fk  
JOIN Ulaznica u ON u.SifraFilma#fk = p.SifraFilma#fk AND u.SifraProjekcije#fk =  
p.SifraProjekcije# AND u.SifraSale#fk = p.SifraSale#fk;
```

## Projekat Bioskop

```
-- Upit nad pogledom
SELECT *
FROM pFilm_Sala_Projekcija_Ulaznica
```

Sadržaj koji upit vraća dat je u sledećoj tabeli.

	SifraFilma#	NazivFilma	SifraSale#	Datum	Satnica	Cena	OznakaSedista
1	10	Fight Club	1	2016-02-10	17:00:00.0000000	500.00	R1-S1
2	11	The Dark Knight	2	2016-02-10	18:00:00.0000000	400.00	R2-S2
3	10	Fight Club	1	2016-02-10	19:00:00.0000000	500.00	R3-S3

### 9.2.2 Poslovna transakcija i bazna procedura

#### 9.2.2.1 Definicija poslovne transakcije

Određena blagajna ima potrebu da za zadati naziv filma i šifru sale, za zadati datum i vreme početka projekcije, izvrši prodaju traženog broja ulaznica. Maksimalan broj ulaznica koji se prodaju u jednoj poslovnoj transakciji može biti 5. Pri ovome, poznat je i red u kojem će biti sedišta za ulaznice. Ako za zadatu projekciju nema traženi broj slobodnih mesta, od prodaje ulaznica se odustaje.

#### 9.2.2.2 Bazna procedura

Napraviti baznu proceduru `kdpProdajaUlaznica` (prefiks `kdp` - korisnički definisana procedura), koja će kao argument prihvatiti šifru filma, šifru sale, datum i vreme početka projekcije kao i cenu ulaznice. Pored toga potrebno je navesti i broj ulaznica koji se kupuje, red kao i početno sedište u navedenoj Sali. U nastavku sledi programski kod za kreiranje bazne procedure.

```
USE PrimerProjekta
GO
```

```
CREATE PROCEDURE kdpProdajaUlaznica
```

```
-- potrebni parametri
@pCena money,
@pSifraSale#fk int,
@pSifraFilma#fk int,
@pSifraBlagajne#fk int,
```

```
@pSatnica time(7),
@pDatum date,
@pRed int,
@pSediste int,
@pBrojUlaznica int
```

```
AS
DECLARE
--podaci za ulaznicu
```

## Projekat Bioskop

```
@lSifraUlaznice# as int, @lOznakaSedista as nvarchar(50), @lBrojacSedista as int,  
    @lSediste as int, @lSifraProjekcije as int,  
    @lMaxRed as int, @lMaxSedista as int, @lBrojSala as int, @lBrojFilmova as int  
  
BEGIN TRY  
    SET XACT_ABORT ON -- U slučaju pojave greške automatsko sprečavanje daljeg izvršavanja  
niza naredbi  
    SET @lBrojSala = 0; -- Promenljiva za testiranje postojanja sale  
    SET @lBrojFilmova = 0; -- Promenljiva za testiranje postojanja filma  
    SELECT @lSifraUlaznice# = max(SifraUlaznice#)+1  
    FROM Ulaznica;  
  
    SELECT @lSifraProjekcije = SifraProjekcije#  
    FROM Projekcija  
    WHERE Datum = @pDatum AND Satnica = @pSatnica AND SifraSale#fk = @pSifraSale#fk  
        AND SifraFilma#fk = @pSifraFilma#fk;  
  
    SELECT @lMaxRed = Redovi + 1, @lMaxSedista = Sedista + 1  
    FROM Sala  
    WHERE SifraSale# = @pSifraSale#fk  
  
    SELECT @lBrojSala = count(*)  
    FROM Sala  
    WHERE SifraSale# = @pSifraSale#fk;  
  
    SELECT @lBrojFilmova = count(*)  
    FROM Film  
    WHERE SifraFilma# = @pSifraFilma#fk;  
  
    SET @lBrojacSedista = 0;  
  
    SET @lSediste = @pSediste;  
  
    IF @lBrojSala = 0 OR @lBrojFilmova = 0  
    BEGIN  
        PRINT 'Proverite ulazne parametre!'  
    END  
    ELSE  
  
        IF @pBrojUlaznica > 5  
        BEGIN  
            PRINT 'Broj ulaznica ne moze biti veci od 5!'  
        END  
        ELSE  
  
            IF @pRed > @lMaxRed  
            BEGIN  
                PRINT 'Red koji ste uneli je veci od broja redova koje postoje za navedenu salu!'  
            END  
            ELSE  
  
                IF @pSediste + @pBrojUlaznica > @lMaxSedista  
                BEGIN  
                    PRINT 'Nije moguće uzeti karte od navedenog sedista!'  
                END  
                ELSE  
  
                    WHILE (@lBrojacSedista < @pBrojUlaznica)
```

```

BEGIN
BEGIN TRANSACTION
    SET @lOznakaSedista = 'R'+
    CAST(@pRed AS varchar(10)) + '-S' + CAST(@lSediste AS varchar(10));

    -- Ubacivanje ulaznice
    INSERT INTO Ulaznica(SifraUlaznice#, Cena,
SifraProjekcije#fk, SifraSale#fk, SifraFilma#fk, SifraBlagajne#fk, OznakaSedista)
    VALUES(@lSifraUlaznice#, @pCena, @lSifraProjekcije,
@pSifraSale#fk, @pSifraFilma#fk, @pSifraBlagajne#fk, @lOznakaSedista );

    SET @lBrojacSedista = @lBrojacSedista + 1
    SET @lSediste = @lSediste + 1;
    SET @lSifraUlaznice# = @lSifraUlaznice# + 1;

    PRINT 'Ubačena je nova ulaznica broj: ' +
    CAST(@lBrojacSedista AS varchar(10));

COMMIT TRANSACTION;
END

END TRY

BEGIN CATCH
    PRINT 'DOŠLO JE DO POJAVE GREŠKE!';
    PRINT '----- Proverite ulazne parametre-----';
    PRINT ERROR_MESSAGE();

    ROLLBACK TRANSACTION;
    PRINT 'Procedura je poništila promene.';
END CATCH;

-- Pozivanje bazne procedure
EXEC kdpProdajaUlaznica @pCena = 700.00, @pSifraSale#fk = 1, @pSifraFilma#fk = 10,
@pSifraBlagajne#fk = 1111, @pSatnica = '17:00:00.0000000', @pDatum = '2016-02-10', @pRed = 5,
@pSediste = 10, @pBrojUlaznica = 5;

```

### 9.2.3 Bazna funkcija

Potrebno je kreirati baznu funkciju kdfPrikazUlaznica (prefiks kdf – korisnički definisana funkcija) koja će kao argument prihvatiti šifru filma, a vratiti tabelu gde se nalazi šifra filma, naziv filma, šifra sale, datum, satnica, cena karte i oznaka sedišta za navedenu šifru filma.

U nastavku sledi programski kod za kreiranje bazne funkcije. Kao što se može primetiti, funkcija koristi predhodno definisani pogled pFilm\_Sala\_Projekcija\_Ulaznica.

```

CREATE FUNCTION kdfPrikazUlaznica(@pSifraFilma int)
RETURNS TABLE
AS
RETURN SELECT p.SifraFilma#, p.NazivFilma, p.SifraSale#, p.Datum, p.Satnica, p.Cena,
p.OznakaSedista
FROM pFilm_Sala_Projekcija_Ulaznica p
WHERE p.SifraFilma# = @pSifraFilma;

-- Pozivanje bazne funkcije
SELECT *
FROM kdfPrikazUlaznica (10);

```

	SifraFilma#	NazivFilma	SifraSale#	Datum	Satnica	Cena	OznakaSedista
1	10	Fight Club	1	2016-02-10	17:00:00.0000000	500.00	R1-S1
2	10	Fight Club	1	2016-02-10	17:00:00.0000000	700.00	R5-S10
3	10	Fight Club	1	2016-02-10	17:00:00.0000000	700.00	R5-S11
4	10	Fight Club	1	2016-02-10	17:00:00.0000000	700.00	R5-S12
5	10	Fight Club	1	2016-02-10	17:00:00.0000000	700.00	R5-S13
6	10	Fight Club	1	2016-02-10	17:00:00.0000000	700.00	R5-S14

Kao što se vidi, rezultat izvršenja bazne funkcije pokazuje rad bazne procedura, koja je realizovala jedan primer zahtevane poslovne transakcije.

### 9.3 Podsystem Dobavljač

#### 9.3.1 Pogled

Potrebno je kreirati pogled pDobavljac\_Otpremnica\_Faktura (prefiks p za pogled) kroz koji će se moći videti podaci o dobavljačima (SifraDobavljacka#, AdresaDobavljacka, NazivDobavljacka), njihovim otpremnicama (BrojOtpremnice#, Kolicina, Datum, Vrednost) i fakturama (BrojFakture#, PDV, UkupnaVrednost, RokPlacanja) koje su za istu robu kao i otpremnica i imaju ukupnu vrednost jednaku vrednosti sa otpremnice.

```
USE PrimerProjekta;
GO
```

```
CREATE VIEW pDobavljac_Otpremnica_Faktura (SifraDobavljacka#, AdresaDobavljacka,
NazivDobavljacka, BrojOtpremnice#, Kolicina, Datum, Vrednost, BrojFakture#, DatumFakture,
PDV, UkupnaVrednost, RokPlacanja)
AS
SELECT d.SifraDobavljacka#, d.AdresaDobavljacka, d.NazivDobavljacka, o.BrojOtpremnice#,
o.Kolicina, o.Datum, o.Vrednost, f.BrojFakture#, f.DatumFakture, f.PDV, f.UkupnaVrednost,
f.RokPlacanja
FROM Dobavljac d JOIN Otpremnica o ON d.SifraDobavljacka# = o.SifraDobavljacka#fk
LEFT JOIN Faktura f ON d.SifraDobavljacka# = f.SifraDobavljacka#fk
AND f.SifraRobe#fk = o.SifraRobe#fk AND f.UkupnaVrednost = o.Vrednost
```

```
-- Upit nad pogledom
```

```
SELECT *
FROM pDobavljac_Otpremnica_Faktura;
```

Sadržaj koji upit vraća dat je u sledećoj tabeli.

	SifraDobavljacka#	AdresaDobavljacka	NazivDobavljacka	BrojOtpremnice#	Kolicina	Datum	Vrednost	BrojFakture#	DatumFakture	PDV	UkupnaVrednost	RokPlacanja
1	33	Visegradska 11	Rosa	1	240	2016-02-02	24000.00	11111	2016-02-02	20	24000.00	2
2	11	Svetogorska 10	Marbo	2	220	2016-02-02	12000.00	22222	2016-02-02	20	12000.00	3
3	22	Kneza Milosa	Next	3	110	2016-02-02	11000.00	33333	2016-02-02	10	11000.00	1

### 9.3.2 Poslovna transakcija i bazna procedura

#### 9.3.2.1 Definicija poslovne transakcije

Za željenog dobavljača treba formirati otpremnicu za datu robu, u poznatoj količini i vrednost dotične robe. Datum otpremnice je 10 dana pre tekućeg datuma. Posle formiranja otpremnice treba formirati fakturu koja odgovara datoj otpremnici. Datum fakture je isti kao i datum odgovarajuće otpremnice. Rok plaćanja fakture i PDV na fakturi su takođe poznati poslovnoj transakciji u vreme njene realizacije. Posle formiranja fakture treba formirati odgovarajuću isplatu. Datum na isplati je datum na odgovarajućoj fakturi, uvećan za rok plaćanja sa fakture.

#### 9.3.2.2 Bazna procedura

Napraviti baznu proceduru `kdpFormiranjeOtpremniceFaktureIsplate` (prefiks `kdp` – korisnički definisana procedura), koja treba da formira otpremnicu dobavljača u sistemu (na osnovu podataka sa otpremnice koju je dobavljač poslao). Procedura takođe treba da formira fakturu za dotičnu otpremnicu (na osnovu podataka sa fakture koju je dobavljač poslao). Posle formiranja fakture, procedura treba da formira odgovarajuću isplatu.

Da bi navedene dokumente formirala, procedura kao ulazne parametre treba da ima: šifru dobavljača, šifru robe, količinu i vrednost robe na otpremnici, pdv, kao i rok plaćanja na fakturi.

Kod formiranja otpremnice, broj nove otpremnice treba da bude za 1 veći od do tada najvećeg broja otpremnice. Broj nove fakture treba da bude za 1 veći od do tada najvećeg broja fakture. Ukupna vrednost na fakturi jednaka je vrednosti sa otpremnice. Datum fakture jednak je datumu otpremnice. Iznos PDV-a na fakturi je procenat poreza primenjen na ukupan iznos.

Šifra isplate treba da je za 1 veća od do tada najveće šifre isplate. Iznos isplate jednak je ukupnoj vrednosti sa fakture. Datum isplate je datum fakture uvećan za rok plaćanja sa fakture (rok plaćanja je izražen u danima).

Programski kod za kreiranje procedure sledi u nastavku.

USE PrimerProjekta

GO

```
CREATE PROCEDURE kdpFormiranjeOtpremniceFaktureIsplate
-- podaci za otpremnicu
@pKolicina int,
@pVrednost money,
@pSifraDobavljacka#fk int,
@pSifraRobe#fk int,

-- podaci za fakturu
@pPDV int,
@pRokPlacanja int

AS
DECLARE
--podaci za otpremnicu
@lDatumOtpremnice as date, @lBrojOtpremnice# as int, @lBrojUbacenihOtpremnica as int = 0,

--podaci za fakturu
@lBrojFakture# as int, @lUkupnaVrednost as money, @lIznosPDV as money, @lIznosBezPDV as money,
@lOpis as nvarchar(10),
```



## Projekat Bioskop

```
--podaci za isplatu
@lSifraIsplate# as int, @lDatum as date

BEGIN TRY
    SET XACT_ABORT ON -- U slučaju pojave greške automatsko sprečavanje daljeg izvršavanja
    niza naredbi

    BEGIN TRANSACTION
    --otpremnicka
    SELECT @lBrojOtpremnice# = max(BrojOtpremnice#)+1,
    @lDatumOtpremnice = GETDATE()-10
    FROM Otpremnica;

    --faktura
    SELECT @lBrojFakture# = max(BrojFakture#)+1
    FROM Faktura;

    SET @lUkupnaVrednost = @pVrednost;
    SET @lIznosPDV = (@lUkupnaVrednost / 100) * @pPDV;
    SET @lIznosBezPDV = @lUkupnaVrednost - @lIznosPDV;

    SELECT @lOpis = NazivRobe
    FROM Roba
    WHERE SifraRobe# = @pSifraRobe#fk;

    --isplata

    SELECT @lSifraIsplate# = max(SifraIsplate#)+1
    FROM Isplata;

    SET @lDatum = DATEADD(DAY, @pRokPlacanja, @lDatumOtpremnice);

    -- Ubacivanje otpremnice
    INSERT INTO Otpremnica(BrojOtpremnice#, Kolicina, Datum, Vrednost,
SifraDobavljacka#fk, SifraRobe#fk)
    VALUES(@lBrojOtpremnice#, @pKolicina, @lDatumOtpremnice, @pVrednost,
@pSifraDobavljacka#fk, @pSifraRobe#fk);

    --Ubacivanje fakture
    INSERT INTO Faktura(BrojFakture#, IznosBezPDVa, IznosPDV, PDV, UkupnaVrednost,
Opis, RokPlacanja, Kolicina, SifraDobavljacka#fk, SifraRobe#fk, DatumFakture)
    VALUES(@lBrojFakture#, @lIznosBezPDV, @lIznosPDV, @pPDV, @lUkupnaVrednost,
@lOpis, @pRokPlacanja, @pKolicina, @pSifraDobavljacka#fk, @pSifraRobe#fk, @lDatumOtpremnice);

    -- Ubacivanje isplate
    INSERT INTO Isplata(SifraIsplate#, Iznos, Datum, BrojFakture#fk,
SifraDobavljacka#fk)
    VALUES(@lSifraIsplate#, @lUkupnaVrednost, @lDatum, @lBrojFakture#,
@pSifraDobavljacka#fk);

    PRINT 'Ubačena je otpremnica se brojem otpremnice:' + str(@lBrojOtpremnice#) ;
    PRINT 'Ubačena je faktura se brojem fakture:' + str(@lBrojFakture#) ;
    PRINT 'Ubačena je isplata se šifrom isplate:' + str(@lSifraIsplate#) ;
    COMMIT TRANSACTION;

END TRY
```

## Projekat Bioskop

```
BEGIN CATCH
    PRINT 'DOŠLO JE DO POJAVE GREŠKE!' ;
    PRINT '----- Proverite ulazne parametre-----' ;
    PRINT ERROR_MESSAGE();

    ROLLBACK TRANSACTION;
    PRINT 'Procedura je poništila promene.';
END CATCH;

-- Prvo pozivanje bazne procedure

EXEC kdpFormiranjeOtpremniceFaktureIsplate @pKolicina = 150, @pVrednost = 15000.00,
@pSifraDobavljacka#fk = 22, @pSifraRobe#fk = 999, @pPDV = 10, @pRokPlacanja = 5;

-- Drugo pozivanje bazne procedure

EXEC kdpFormiranjeOtpremniceFaktureIsplate @pKolicina = 150, @pVrednost = 15000.00,
@pSifraDobavljacka#fk = 22, @pSifraRobe#fk = 1, @pPDV = 10, @pRokPlacanja = 5;
```

U drugom pozivu procedure prosleđena je vrednost 1 za parametar @pSifraRobe#fk. Pošto takve robe nema u bazi (u vreme izvršenja procedure) pri pokušaju ubacivanja otpremnice došlo je do narušavanja ograničenja spoljnog ključa, pa je procedura izbacila poruke o grešci, kao što je prikazano u nastavku.

```
(0 rows affected)
DOŠLO JE DO POJAVE GREŠKE!
----- Proverite ulazne parametre-----
The INSERT statement conflicted with the FOREIGN KEY constraint
"FK_Otpremnic_Sifra_03F0984C". The conflict occurred in database "PrimerProjekta", table
"dbo.Roba", column 'SifraRobe#'.
Procedura je poništila promene.
```

### 9.3.3 Bazna funkcija

Potrebno je kreirati baznu funkciju kdfOtpremnicaFakturaIsplata (prefiks kdf – korisnički definisana funkcija) koja će kao argument prihvatiti broj otpremnice, a vratiti tabelu gde se nalazi broj otpremnice, vrednost otpremnice, datum otpremnice, naziv dobavljača, broj fakture koja je za istu robu kao i otpremnica i ima ukupnu vrednost jednaku vrednosti sa otpremnice (ako takva fakture postoji), kao i datum i iznos isplate za dotičnu fakturu (ako takva isplata postoji).

U nastavku sledi programski kod za kreiranje bazne funkcije. Kao što se može primetiti, funkcija koristi predhodno definisani pogled pDobavljac\_Otpremnica\_Faktura.

```
CREATE FUNCTION kdfOtpremnicaFakturaIsplata (@pBrojOtpremnice int)
RETURNS TABLE
AS
RETURN SELECT p.BrojOtpremnice#, p.Vrednost as VrednostOtpremnice, p.Datum as
datumOtpremnice, p.NazivDobavljacka, p.BrojFakture#,
i.datum as datumIsplate, i.Iznos as iznosIsplate
FROM pDobavljac_Otpremnica_Faktura p LEFT JOIN Isplata i
ON i.BrojFakture#fk = p.BrojFakture#
WHERE p.BrojOtpremnice# = @pBrojOtpremnice;

-- Pozivanje bazne funkcije
SELECT *
FROM kdfOtpremnicaFakturaIsplata (4);
```

## Projekat Bioskop

BrojOtpremnice#	VrednostOtpremnice	datumOtpremnice	NazivDobavljacka	BrojFakture#	datumIsplate	iznosIsplate
4	15000.00	2019-10-17	Next	33334	2019-10-22	15000.00

## 10. Literatura

1. Majstorović M.: Baze podataka, skripta, Visoka škola strukovnih studija za informacione tehnologije, Beograd, 2013.
2. M. Majstorović, Uvod u baze podataka, Visoka škola strukovnih studija za informacione tehnologije, Beograd, 2015, ISBN 978-86-89007-03-9.
3. Majstorović M.: Napredne baze podataka, ITS, Beograd, 2017, ISBN 978-86-89007-19-0.
4. Majstorović M.: Baze podataka, predavanja i vežbe za predmet u ppt formi, školska 2019/2020. (na studentskom portalu).