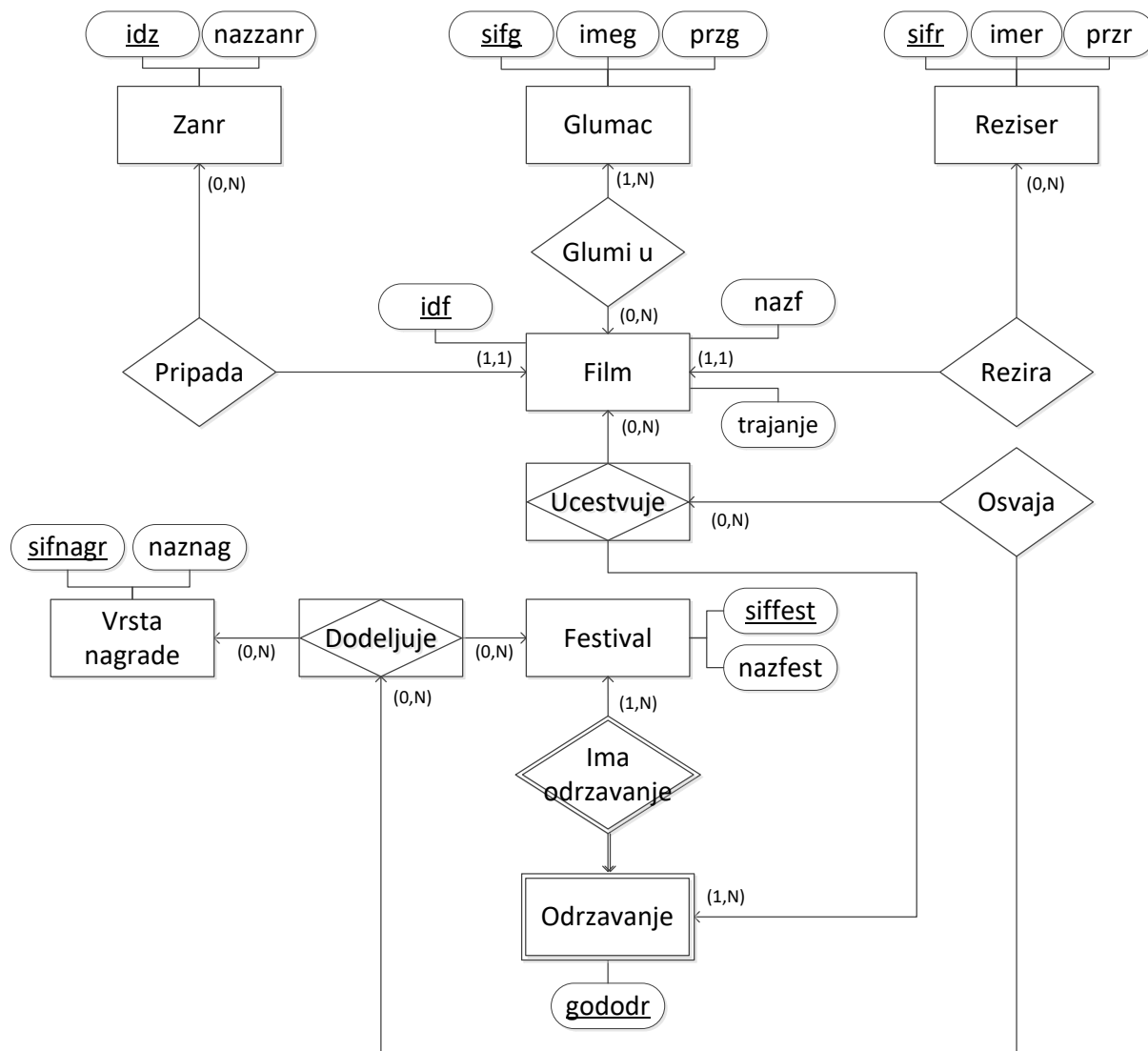


Zadatak 1

Na osnovu datog modela tipova entiteta i poveznika prevesti ER dijagram u relacionu šemu baze podataka. Na osnovu dobijene relacione šeme baze podataka napraviti njen SQL opis.



Relaciona šema baze podataka:

Zavr({idz, nazzanr}, {idz})

Reziser({sifr, imer, przr}, {sifr})

Film({idf, nazf, trajanje, idz, sifr}, {idf})

Film[idz] \subseteq Zavr[idz], Null(Film, idz) = \perp

Film[sifr] \subseteq Reziser[sifr], Null(Film, sifr) = \perp

Glumac({sifg, imeg, przg}, {sifg})

GlumiU({sifg, idf}, {sifg+idf})

GlumiU[sifg] \subseteq Glumac[sifg],

inverzni referencijalni integritet: Glumac[sifg] \subseteq GlumiU[sifg]

GlumiU[idf] \subseteq Film[idf]

VrstaNagrade({sifnagr, naznag}, {sifnagr})

Festival({siffest, nazfest}, {siffest})

Dodeljuje({sifnagr, sifest}, {sifnarg+sifest})
 Dodeljuje[sifnagr] \subseteq VrstaNagrade[sifnagr]
 Dodeljuje[sifest] \subseteq Festival[sifest]
 Odrzavanje({gododr, sifest}, {gododr+sifest})
 Odrzavanje[sifest] \subseteq Festival[sifest],
 inverzni referencijalni integritet: Festival[sifest] \subseteq Odrzavanje[sifest]
 Ucestvuje({idf, gododr, sifest}, {idf+gododr+sifest})
 Ucestvuje[idf] \subseteq Film[idf]
 Ucestvuje[gododr+sifest] \subseteq Odrzavanje[gododr+sifest],
 inverzni referencijalni integritet: Odrzavanje[gododr+sifest] \subseteq Ucestvuje[gododr+sifest]
 Osvaja({idf, gododr, sifest, sifnagr}, {idf + gododr + sifestodr + sifnagr})
sifest od dodeljuje je preneti ali je izjednačena sa sifest iz Ucestvuje
 Osvaja[idf+gododr+ sifest] \subseteq Ucestvuje[idf+gododr+ sifest]
 Osvaja[sifnagr+ sifest] \subseteq Dodeljuje[sifnagr+ sifest]

SQL opis:

```

CREATE TABLE Zanr (
    idz INTEGER,
    nazzanr VARCHAR(50),
    CONSTRAINT Zanr_PK PRIMARY KEY(idz)
);

CREATE TABLE Reziser (
    sifr INTEGER,
    imer VARCHAR(50),
    przr VARCHAR(50),
    CONSTRAINT Reziser_PK PRIMARY KEY(sifr)
);

CREATE TABLE Film (
    idf INTEGER,
    nazf VARCHAR(50),
    trajanje VARCHAR(50),
    idz INTEGER NOT NULL,
    sifr INTEGER NOT NULL,
    CONSTRAINT Film_PK PRIMARY KEY(idf),
    CONSTRAINT Film_Zanr_FK FOREIGN KEY(idz)
        REFERENCES Zanr(idz),
    CONSTRAINT Film_Reziser_FK FOREIGN KEY(sifr)
        REFERENCES Reziser(sifr)
);

CREATE TABLE Glumac (
    sifg INTEGER,
    imeg VARCHAR(50),
    przg VARCHAR(50),
    CONSTRAINT Glumac_PK PRIMARY KEY(sifg)
);
  
```

```

CREATE TABLE Glumi_u (
    sifg INTEGER,
    idf INTEGER,
    CONSTRAINT Glumi_u_PK PRIMARY KEY(sifg, idf),
    CONSTRAINT Glumi_u_Glumac_FK FOREIGN KEY(sifg)
        REFERENCES Glumac(sifg),
    CONSTRAINT Glumi_u_Film_FK FOREIGN KEY(idf)
        REFERENCES Film(idf)
);

CREATE TABLE Vrsta_nagrade (
    sifnagr INTEGER,
    naznag VARCHAR(50),
    CONSTRAINT Vrsta_nagrade_PK PRIMARY KEY(sifnagr)
);

CREATE TABLE Festival (
    siffest INTEGER,
    nazfest VARCHAR(50),
    CONSTRAINT Festival_PK PRIMARY KEY(siffest)
);

CREATE TABLE Dodeljuje (
    sifnagr INTEGER,
    siffest INTEGER,
    CONSTRAINT Dodeljuje_PK PRIMARY KEY(sifnagr, siffest),
    CONSTRAINT Dodeljuje_Vrsta_nagrade_FK FOREIGN KEY(sifnagr)
        REFERENCES Vrsta_nagrade(sifnagr),
    CONSTRAINT Dodeljuje_Festival_FK FOREIGN KEY(siffest)
        REFERENCES Festival(siffest)
);

CREATE TABLE Odrzavanje (
    gododr DATE,
    siffest INTEGER,
    CONSTRAINT Odrzavanje_PK PRIMARY KEY(siffest, gododr),
    CONSTRAINT Odrzavanje_Festival_FK FOREIGN KEY(siffest)
        REFERENCES Festival (siffest)
);

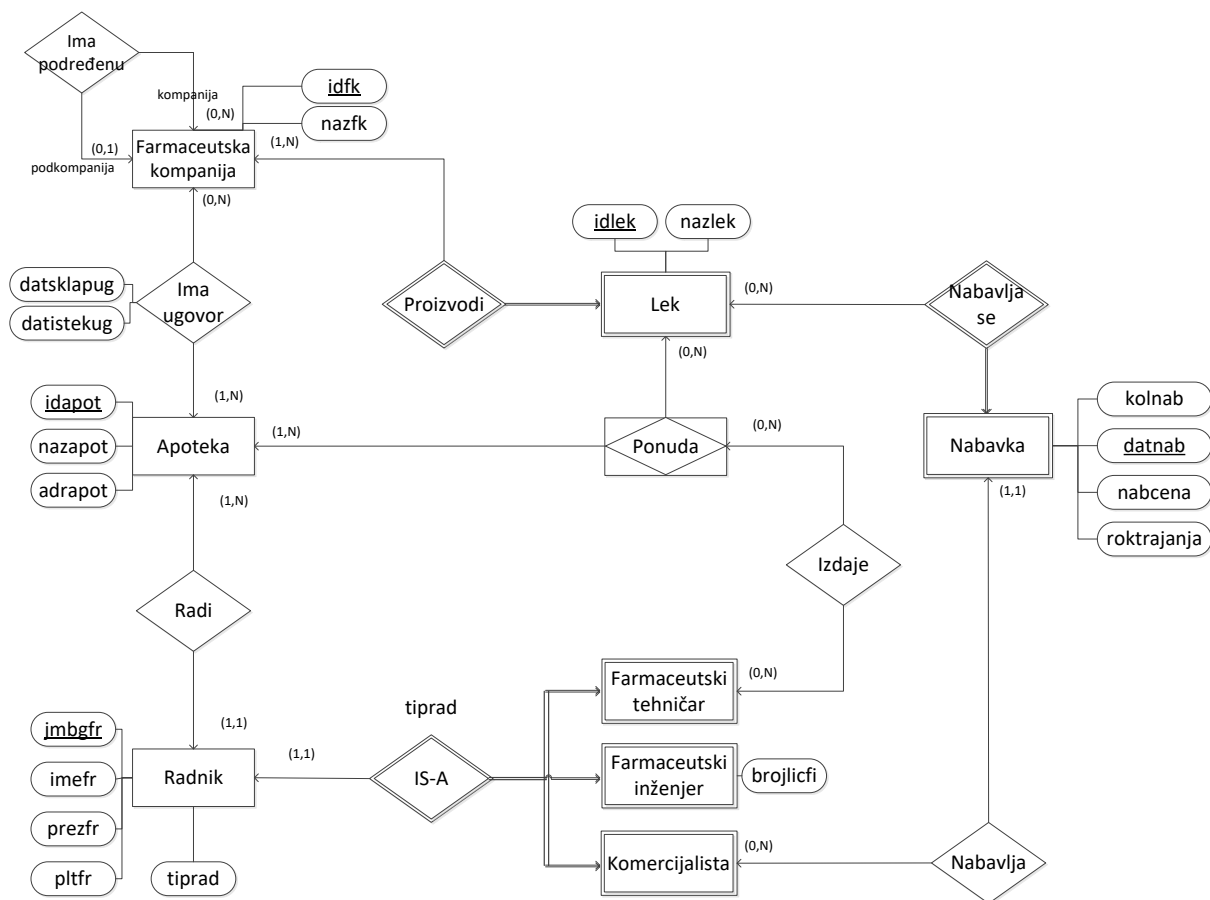
CREATE TABLE Ucestvuje (
    idf INTEGER,
    gododr DATE,
    siffest INTEGER,
    CONSTRAINT Ucestvuje_PK PRIMARY KEY(idf, gododr, siffest),
    CONSTRAINT Ucestvuje_Film_FK FOREIGN KEY(idf)
        REFERENCES Film(idf),
    CONSTRAINT Ucestvuje_Odrzavanje_FK FOREIGN KEY(siffest, gododr)
        REFERENCES Odrzavanje(siffest, gododr)
);

```

```
CREATE TABLE Osvaja (  
    idf INTEGER,  
    gododr DATE,  
    siffest INTEGER,  
    sifnagr INTEGER,  
    CONSTRAINT Osvaja_PK PRIMARY KEY(idf, gododr, siffest, sifnagr),  
    CONSTRAINT Osvaja_Dodeljuje_FK FOREIGN KEY(sifnagr,siffest)  
        REFERENCES Dodeljuje(sifnagr,siffest),  
    CONSTRAINT Osvaja_Ucestvuje_FK FOREIGN KEY(idf,gododr,siffest)  
        REFERENCES Ucestvuje(idf,gododr,siffest)  
);
```

Zadatak 2

Na osnovu datog modela tipova entiteta i poveznika prevesti ER dijagram u relacionu šemu baze podataka. Na osnovu dobijene relacione šeme baze podataka napraviti njen SQL opis.



FarmaceutskaKompanija(idfk, nazfk, idnadfk), {idfk})
 FarmaceutskaKompanija[idnadfk] \subseteq FarmaceutskaKompanija[idfk]
 Null(FarmaceutskaKompanija, idnadfk) = T
 dom(idnadfk) \subseteq dom(idfk)
 Lek(idfk, idlek, nazlek), {idfk+idlek})
 Lek[idfk] \subseteq FarmaceutskaKompanija[idfk]
 inverzni referencijalni integritet: FarmaceutskaKompanija[idfk] \subseteq Lek[idfk]
 Apoteka(idapot, nazapot, adrapot), {idapot})
 ImaUgovor(idfk, idapot, datsklapug, datistekug), {idfk+idapot})
 ImaUgovor[idfk] \subseteq FarmaceutskaKompanija[idfk]
 ImaUgovor[idapot] \subseteq Apoteka[idapot]
 inverzni referencijalni integritet: Apoteka[idapot] \subseteq ImaUgovor[idapot]
 Radnik(jmbgfr, imefr, prezfr, pltfr, tiprad, idapot), {jmbgfr})
 Radnik[idapot] \subseteq Apoteka[idapot], Null(Radnik, idapot) = \perp
 inverzni referencijalni integritet: Apoteka[idapot] \subseteq Radnik[idapot]
 FarmaceutskiTehničar(jmbgfr), {jmbgfr})
 FarmaceutskiTehničar[jmbgfr] \subseteq Radnik[jmbgfr]
 FarmaceutskiInženjer(jmbgfr, brojlicfi), {jmbgfr})
 FarmaceutskiInženjer[jmbgfr] \subseteq Radnik[jmbgfr]
 Komercijalista(jmbgfr), {jmbgfr})

Komercijalista[jmbgfr] \subseteq Radnik[jmbgfr]

Totalna IS-A hijerarhija:

Radnik[jmbgfr] \subseteq FarmaceutskiTehničar[jmbgfr] \cup FarmaceutskiInženjer[jmbgfr] \cup

Komercijalista[jmbgfr]

Nepresečna IS-A hijerarhija:

FarmaceutskiTehničar[jmbgfr] \cap FarmaceutskiInženjer[jmbgfr] = \emptyset

FarmaceutskiTehničar[jmbgfr] \cap Komercijalista[jmbgfr] = \emptyset

FarmaceutskiInženjer[jmbgfr] \cap Komercijalista[jmbgfr] = \emptyset

Nabavka({idfk, idlek, datnab, kolnab, nabcena, roktrajanja, jmbgfr}, {idfk+idlek+datnab})

Nabavka[jmbgfr] \subseteq Komercijalista[jmbgfr], Null(Nabavka, jmbgfr) = \perp

Nabavka[idfk+idlek] \subseteq Lek[idfk+idlek]

Ponuda({idapot, idfk, idlek,}, {idapot+ idfk+idlek})

Ponuda[idapot] \subseteq Apoteka[idapot]

inverzni referencijalni integritet: Apoteka[idapot] \subseteq Ponuda[idapot]

Ponuda[idlek+idfk] \subseteq Lek[idlek+idfk]

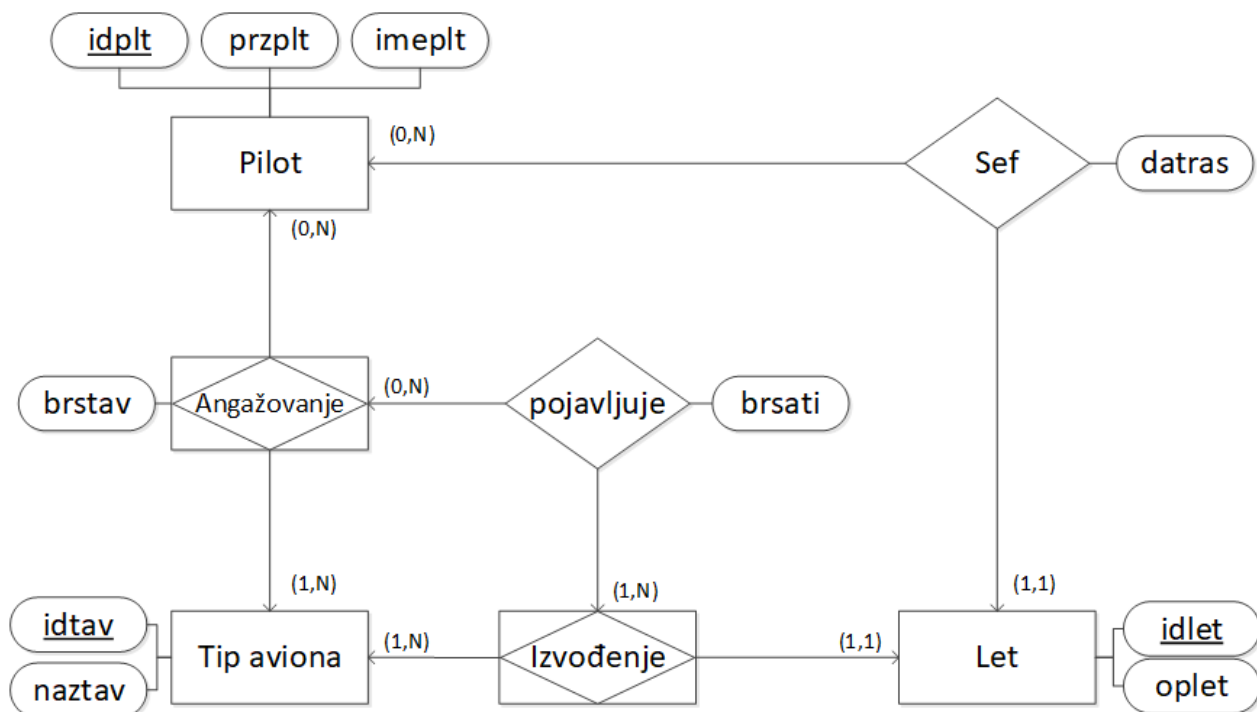
Izdaje({idfk, idlek, idapot, jmbgfr}, {idfk+idlek+idapot+jmbgfr})

Izdaje[jmbgfr] \subseteq FarmaceutskiTehničar[jmbgfr]

Izdaje[idfk+idlek+idapot] \subseteq Ponuda[idfk+idlek+idapot]

Zadatak 3

Na osnovu datog modela tipova entiteta i poveznika prevesti ER dijagram u relacionu šemu baze podataka. Na osnovu dobijene relacione šeme baze podataka napraviti njen SQL opis.



Pilot({idplt, przplt, imeplt}, {idplt})

TipAviona({idtav, naztav}, {idtav})

Angažovanje({brstav, idplt, idtav}, {idplt+idtav})

Angažovanje[idplt] ⊆ Pilot[idplt]

Angažovanje[idtav] ⊆ TipAviona[idtav]

inverzni referencijalni integritet: TipAviona[idtav] ⊆ Angažovanje[idtav]

Let({idlet, oplet, datras, idpltsef, idtav}, {idlet})

Let[idpltsef] ⊆ Pilot[idplt]

Null(Let, idpltsef) = ⊥

dom(Let, idpltsef) ⊆ dom(Pilot, idplt)

Let[idtav] ⊆ TipAviona[idtav]

Null(Let, idtav) = ⊥

inverzni referencijalni integritet: TipAviona[idtav] ⊆ Let[idtav]

Pojavljuje({brsati, idplt, idtav, idlet}, {idplt+idtav+idlet})

Pojavljuje[idplt+idtav] ⊆ Angažovanje[idplt+idtav]

Pojavljuje[idlet] ⊆ Let[idlet]

inverzni referencijalni integritet: Let[idlet] ⊆ Pojavljuje[idlet]