

Baze podataka

Predavanja

13. ER model baze podataka – 3. dio primjeri

Svibanj, 2021.



1. Model baze podataka za razredbeni ispit

Potrebno je evidentirati podatke o kandidatima: JMBG, prezime, ime, završenu srednju školu, mjesto rođenja i mjesto stanovanja. Pretpostavlja se da je kandidat završio samo jednu srednju školu. Za svaku srednju školu treba evidentirati šifru koja ju jedinstveno identificira, naziv, adresu i mjesto u kojem se škola nalazi. Za mjesto treba evidentirati poštanski broj, naziv mjesta i županiju u kojoj se mjesto nalazi. Županija ima svoju šifru i naziv.

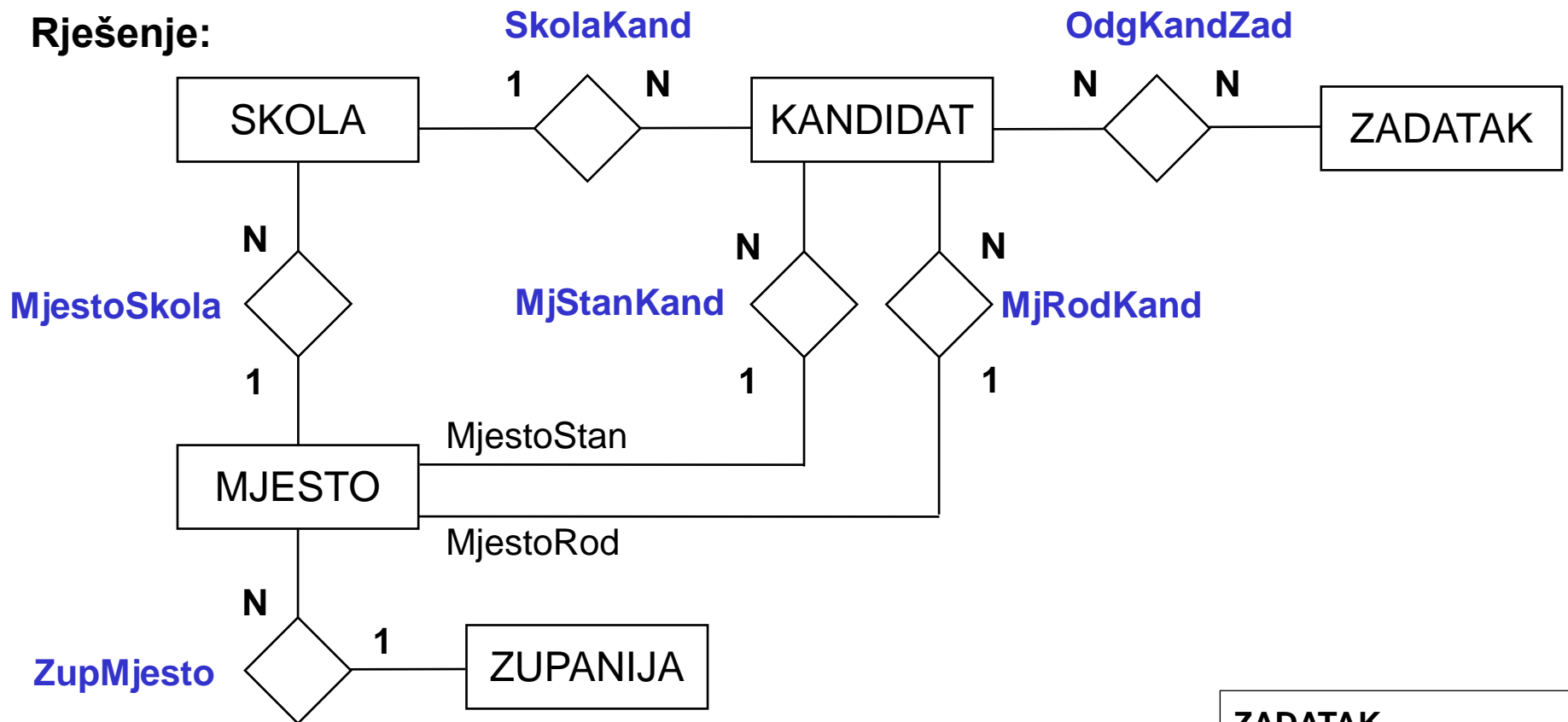
Treba evidentirati podatke o zadacima na testu: redni broj zadatka, tekst zadatka, oznaku točnog odgovora (može biti A, B, C, D ili E).

Za svakog kandidata evidentirati odgovore koje je dao na zadatke (mogući odgovori kandidata su A, B, C, D, E ili ništa).

Nacrtati ER model i opisati entitete i veze. Sve sheme moraju zadovoljavati 3NF.

Opisati relacijski model u obliku SQL naredbi za kreiranje relacija s opisanim integritetskim ograničenjima. Odabrati prikladne tipove podataka.

Rješenje:



KANDIDAT
jmbg
imeKand
prezKand
K = { jmbg }

SKOLA
sifSkola
nazSkola
K = { sifSkola }

MJESTO
pbr
nazMjesto
K = { pbr }

ZUPANIJA
sifZupanija
nazZupanija
K = { sifZupanija }

ZADATAK
rbrZadatak
tekstZadatak
oznTocnogOdg
K = { rbrZadatak }

SkolaKand
jmbg
sifSkola
K = { jmbg }

MjestoSkola
sifSkola
pbr
adresa
K = { sifSkola }

ZupMjesto
pbr
sifZupanija
K = { pbr }

MjStanKand
jmbg
pbr
K = { jmbg }

MjRodKand
jmbg
pbr
K = { jmbg }

OdgKandZad
jmbg
rbrZadatak
odgKand
K = { jmbg, rbrZadatak }

KANDIDAT

jmbg

imeKand

prezKand

 $K = \{ \text{jmbg} \}$

Ako bi neki entitet imao više mogućih ključeva, shema entiteta bi se mogla opisati npr. ovako:

KANDIDAT

jmbg

sifKand

imeKand

prezKand

 $PK = K_1 = \{ \text{jmbg} \}$ **$K_2 = \{ \text{sifKand} \}$**

Relacijski model u obliku SQL naredbi za kreiranje relacija:

```
CREATE TABLE zupanija (  
    sifZupanija    SMALLINT  
    , nazZupanija    CHAR(40)  
    , PRIMARY KEY (sifZupanija));  
  
CREATE TABLE mjesto (  
    pbr            INTEGER  
    , nazMjesto    CHAR(20)  
    , sifZupanija  SMALLINT NOT NULL  
    , PRIMARY KEY (pbr)  
    , FOREIGN KEY (sifZupanija) REFERENCES zupanija(sifZupanija));  
  
CREATE TABLE skola (  
    sifSkola    INTEGER  
    , nazSkola    CHAR(40)  
    , pbr            INTEGER NOT NULL  
    , adresa        CHAR(40)  
    , PRIMARY KEY (sifSkola)  
    , FOREIGN KEY (pbr) REFERENCES mjesto(pbr));  
  
CREATE TABLE zadatak (  
    rbrZadatak    INTEGER  
    , tekstZadatak CHAR(512)  
    , oznTocnogOdg CHAR(1)  
    , PRIMARY KEY (rbrZadatak));
```

Relacijski model u obliku SQL naredbi za kreiranje relacija (nastavak):

```
CREATE TABLE kandidat (  
    jmbg          CHAR(13)  
    , imeKand     CHAR(20)  
    , prezKand    CHAR(20)  
    , pbrRod      INTEGER NOT NULL  
    , pbrStan     INTEGER NOT NULL  
    , sifSkola    INTEGER NOT NULL  
    , PRIMARY KEY (jmbg)  
    , FOREIGN KEY (pbrRod) REFERENCES mjesto (pbr)  
    , FOREIGN KEY (pbrStan) REFERENCES mjesto (pbr)  
    , FOREIGN KEY (sifSkola) REFERENCES skola (sifSkola));
```

```
CREATE TABLE odgKandZad (  
    jmbg          CHAR(13)  
    , rbrZadatak  INTEGER  
    , odgKand     CHAR(1)  
    , PRIMARY KEY (jmbg, rbrZadatak)  
    , FOREIGN KEY (jmbg) REFERENCES kandidat (jmbg)  
    , FOREIGN KEY (rbrZadatak) REFERENCES zadatak (rbrZadatak));
```

2. Model baze podataka za videoteku

Za film se evidentira šifra (identificira film), naslov filma, te osobe i njihove funkcije u filmu.

Funkcije koje osoba može imati u filmu predstavljene su kraticom i nazivom (npr. GL, glumac; RED, redatelj; SC, scenarist, itd.). Za svaku se osobu evidentira šifra osobe (identificira osobu), prezime i ime.

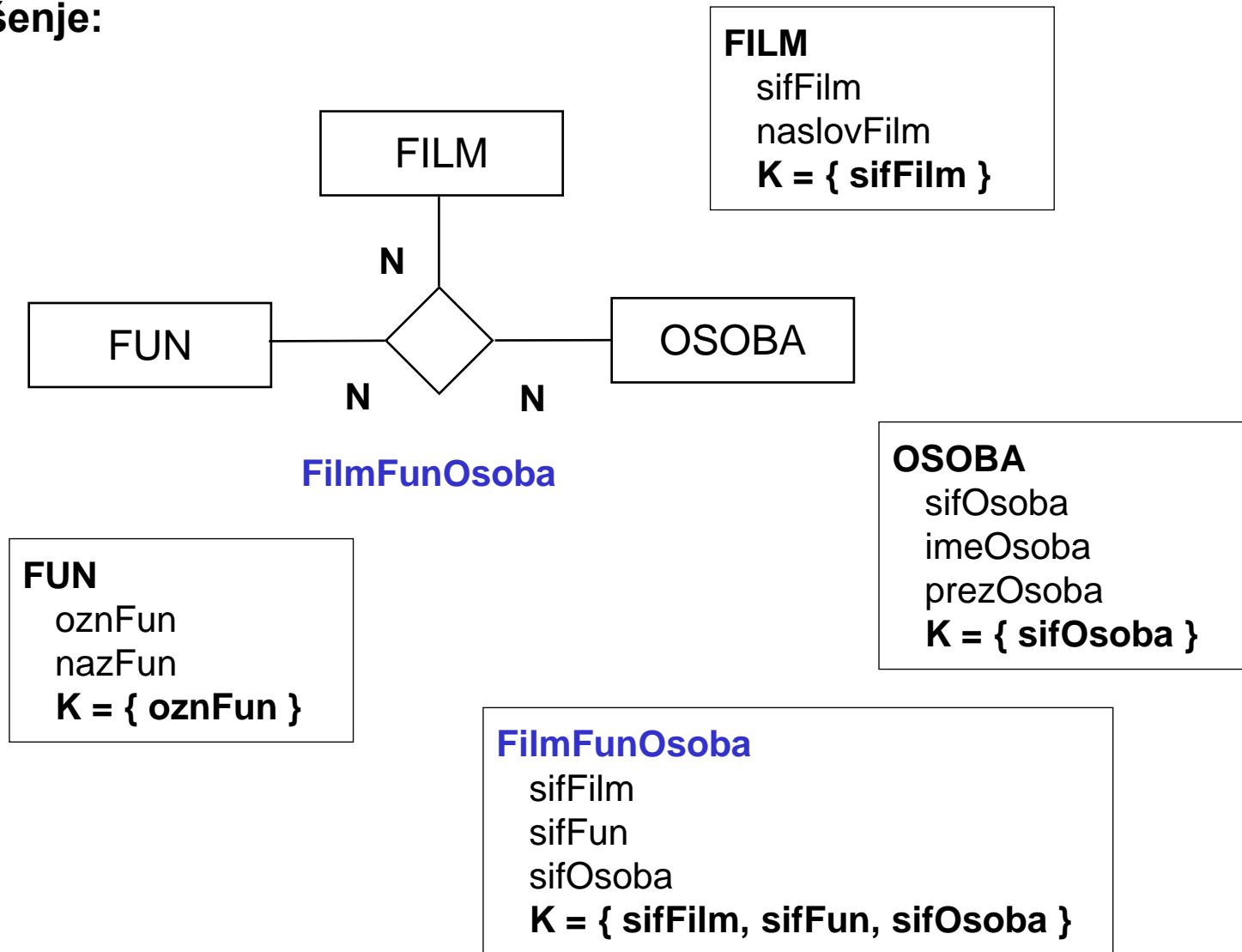
Treba uočiti da ista osoba može u istom filmu imati različite funkcije, npr:

relacija VIDEOTEKA

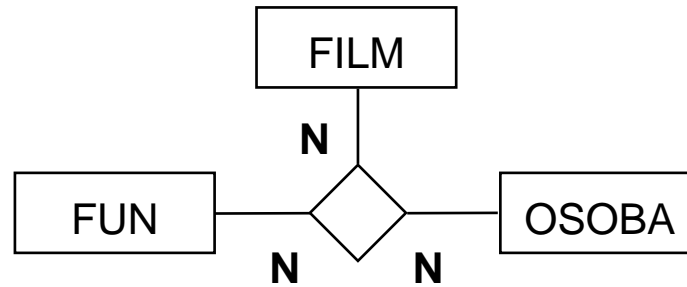
sif Film	naslovFilm	ozn Fun	nazFun	sif Osoba	ime	prezime
1	Nepomirljivi	RED	redatelj	10	Clint	Eastwood
1	Nepomirljivi	GL	glumac	10	Clint	Eastwood
1	Nepomirljivi	GL	glumac	20	Morgan	Freeman
2	Mostovi okruga Madison	RED	redatelj	10	Clint	Eastwood
2	Mostovi okruga Madison	GL	glumac	40	Meryl	Streep
2	Mostovi okruga Madison	GL	glumac	10	Clint	Eastwood
3	Priljavi Harry	RED	redatelj	30	Don	Siegel
3	Priljavi Harry	GL	glumac	10	Clint	Eastwood

Nacrtati ER model i opisati entitete i veze. Sve sheme moraju zadovoljavati 3NF.

Rješenje:



Diskusija:



FilmFunOsoba

Relacije koje
nastaju
transformacijom
ER modela na
slici:

ključevi
relacija su
podcrtani

FILM	
<u><i>sifFilm</i></u>	<i>naslovFilm</i>
1	Nepomirljivi
2	Mostovi okruga M.
3	Prljavi Harry

OSOBA	
<u><i>sifOsoba</i></u>	<i>ime, prezime</i>
10	C. Eastwood
20	M. Freeman
30	D. Siegel
40	M. Streep

FUN	
<u><i>oznFun</i></u>	<i>nazFun</i>
RED	redatelj
GL	glumac

FilmFunOsoba		
<u><i>sifFilm</i></u>	<u><i>oznFun</i></u>	<u><i>sifOsoba</i></u>
1	RED	10
1	GL	10
1	GL	20
2	RED	10
2	GL	40
2	GL	10
3	RED	30
3	GL	10

SQL upit kojim se dohvaćaju osobe i njihove funkcije u filmu "Priljavi Harry"

```
SELECT osoba.*, fun.*
  FROM osoba, film, fun, filmFunOsoba
 WHERE osoba.sifOsoba = filmFunOsoba.sifOsoba
    AND film.sifFilm = filmFunOsoba.sifFilm
    AND fun.oznFun = filmFunOsoba.oznFun
    AND film.naslovFilm = 'Priljavi Harry'
```

sifOsoba	ime	prezime	oznFun	nazFun
10	Clint	Eastwood	GL	glumac
30	Don	Siegel	RED	redatelj

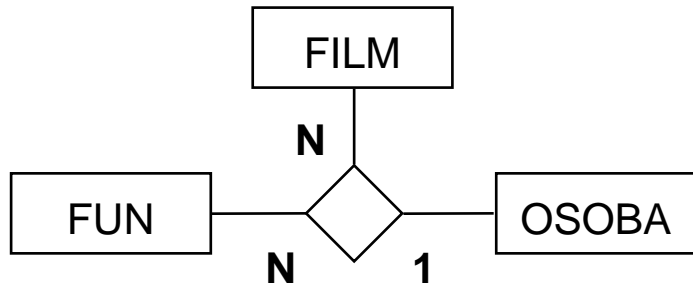
relacija VIDEOTEKA

sif Film	naslovFilm	ozn Fun	nazFun	sif Osoba	ime	prezime
1	Nepomirljivi	RED	redatelj	10	Clint	Eastwood
1	Nepomirljivi	GL	glumac	10	Clint	Eastwood
1	Nepomirljivi	GL	glumac	20	Morgan	Freeman
2	Mostovi okruga Madison	RED	redatelj	10	Clint	Eastwood
2	Mostovi okruga Madison	GL	glumac	40	Meryl	Streep
2	Mostovi okruga Madison	GL	glumac	10	Clint	Eastwood
3	Priljavi Harry	RED	redatelj	30	Don	Siegel
3	Priljavi Harry	GL	glumac	10	Clint	Eastwood

(Teorey): U vezi koja povezuje entitete $E_1, \dots, E_k, \dots, E_m$, spojnost =1 entiteta E_k znači da (...) odnosno, vrijedi funkcijska zavisnost

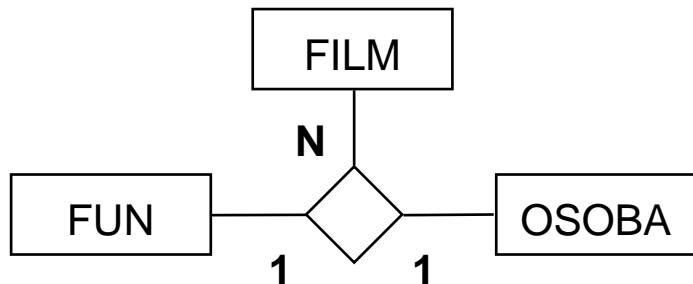
$$\bigcup_{j=1}^m K_j \setminus K_k \rightarrow K_k \quad \text{gdje su skupovi } K_j, (j = 1, \dots, m), \text{ ključevi entiteta } E_1, \dots, E_m$$

ZAŠTO OVI MODELI NISU ISPRAVNI?



$\text{sifFilm, oznFun} \rightarrow \text{sifOsoba}$

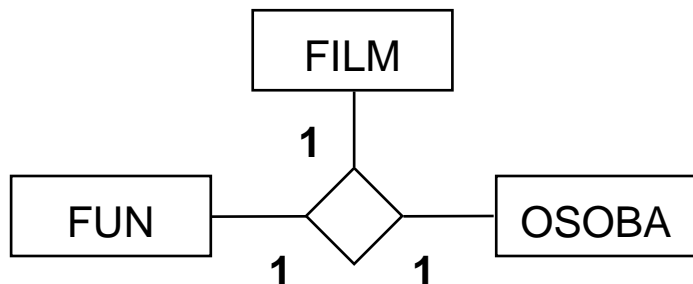
u filmu može glumiti samo jedan glumac, film može režirati samo jedan redatelj, scenarij za film može pisati samo jedan scenarist, ...



$\text{sifFilm, oznFun} \rightarrow \text{sifOsoba}$

$\text{sifFilm, sifOsoba} \rightarrow \text{oznFun}$

dodatno: u jednom filmu osoba može imati samo jednu funkciju



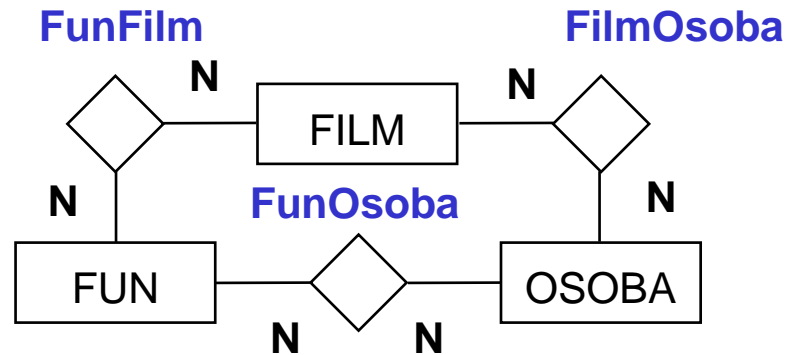
$\text{sifFilm, oznFun} \rightarrow \text{sifOsoba}$

$\text{sifFilm, sifOsoba} \rightarrow \text{oznFun}$

$\text{sifOsoba, oznFun} \rightarrow \text{sifFilm}$

dodatno: ...

ZAŠTO OVAJ MODEL NIJE ISPRAVAN?



Relacije koje
nastaju
transformacijom
ER modela na
slici:

FILM	
<u>sifFilm</u>	<i>naslovFilm</i>
1	Nepomirljivi
2	Mostovi okruga M.
3	Prijavi Harry

ključevi
relacija su
podcrtani

OSOBA	
<u>sifOsoba</u>	<i>ime, prezime</i>
10	C. Eastwood
20	M. Freeman
30	D. Siegel
40	M. Strep

FUN	
<u>oznFun</u>	<i>nazFun</i>
RED	redatelj
GL	glumac

FunFilm	
<u>oznFun</u>	<u>sifFilm</u>
RED	1
RED	2
RED	3
GL	1
GL	2
GL	3

FilmOsoba	
<u>sifFilm</u>	<u>sifOsoba</u>
1	10
1	20
2	10
2	40
3	30
3	10

FunOsoba	
<u>oznFun</u>	<u>sifOsoba</u>
GL	10
GL	20
GL	40
RED	10
RED	30

SQL upit kojim se dohvaćaju osobe i njihove funkcije u filmu "Prljavi Harry"

```
SELECT osoba.*, fun.*
FROM osoba, film, fun, filmOsoba, funOsoba, funFilm
WHERE osoba.sifOsoba = filmOsoba.sifOsoba
      AND filmOsoba.sifFilm = film.sifFilm
      AND film.sifFilm = funFilm.sifFilm
      AND funFilm.oznFun = fun.oznFun
      AND fun.oznFun = funOsoba.oznFun
      AND funOsoba.sifOsoba = osoba.sifOsoba
      AND film.naslovFilm = 'Prljavi Harry'
```

sifOsoba	ime	prezime	oznFun	nazFun
10	Clint	Eastwood	GL	glumac
10	Clint	Eastwood	RED	redatelj
30	Don	Siegel	RED	redatelj

Clint Eastwood nije redatelj filma "Prljavi Harry"!
→ "gubitak informacije"!!!

Dekompozicija relacije VIDEOTEKA nije obavljena bez gubitka informacije. Uvjet za dekompoziciju bez gubitka informacija opisan je u predavanjima.

3. Model baze podataka za poduzeće za održavanje plinskih instalacija

Uređaji koje poduzeće evidentira su brojila, ventili i reduktori. Za svaki pojedini uređaj treba evidentirati vrstu uređaja ('B', 'V' ili 'R'), proizvođača uređaja, tvornički broj i godinu proizvodnje uređaja. Za proizvođače uređaja evidentiraju se njihove šifre i nazivi. Ne postoje dva uređaja istog proizvođača koji imaju jednake tvorničke brojeve.

Dodatno, ovisno o vrsti uređaja, treba evidentirati njima svojstvene, posebne ili specijalističke podatke.

za brojila:	za ventile:	za reduktore:
razred točnosti	promjer navoja	ulazni tlak plina
max. protok plina	način zatvaranja	izlazni tlak plina
		promjer navoja

Potrebno je evidentirati popis lokacija (šifra i naziv) na kojima uređaji mogu biti instalirani. Evidentirati trenutnu lokaciju na kojoj je uređaj instaliran.

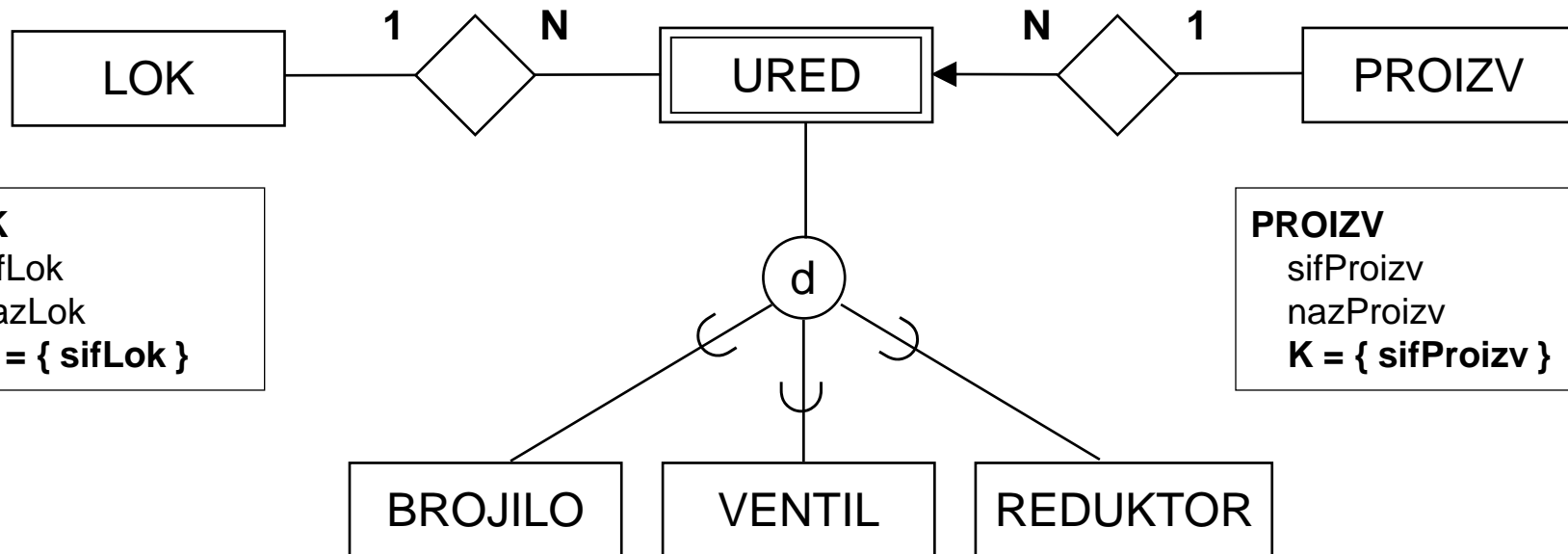
- Nacrtati ER model i opisati entitete i veze. Sve sheme moraju zadovoljavati 3NF. Opisati relacijski model u obliku SQL naredbi za kreiranje relacija s ugrađenim pravilima integriteta.
- Što treba promijeniti u ER modelu iz a) kako bi se omogućilo evidentiranje povijesti premještanja uređaja. Kakve su posljedice na relacijski model?

Rješenje:

a)

LokUred

ProizvUred



LOK

sifLok
nazLok
K = { sifLok }

PROIZV

sifProizv
nazProizv
K = { sifProizv }

URED

tvBroj
sifProizv
godProizv
vrUred
K = { tvBroj, sifProizv }

ProizvUred

tvBroj
sifProizv
K = { tvBroj, sifProizv }

LokUred

tvBroj
sifProizv
sifLok
K = { tvBroj, sifProizv }

BROJILO

tvBroj
sifProizv
razrToen
maxProtok
K = { tvBroj, sifProizv }

REDUKTOR

tvBroj
sifProizv
ulTlak
izlTlak
promjNavoja
K = { tvBroj, sifProizv }

VENTIL

tvBroj
sifProizv
promjNavoja
nacZatvar
K = { tvBroj, sifProizv }

Rješenje: Relacijski model

```
CREATE TABLE proizvod (
    sifProizv    INTEGER
, nazProizv    CHAR(20)
, PRIMARY KEY (sifProizv));
```

```
CREATE TABLE lok (
    sifLok      INTEGER
, nazLok      CHAR(40)
, PRIMARY KEY (sifLok));
```

```
CREATE TABLE ured (
    tvBroj      CHAR(20)
, sifProizv    INTEGER
, godProizv    SMALLINT
, vrUred      CHAR(1)
, sifLok      INTEGER    NOT NULL
, PRIMARY KEY (tvBroj, sifProizv)
, FOREIGN KEY (sifProizv) REFERENCES proizvod (sifProizv)
, FOREIGN KEY (sifLok) REFERENCES lok (sifLok));
```

```
CREATE TABLE brojilo (
    tvBroj      CHAR(20)
, sifProizv    INTEGER
, razrTocn    DECIMAL(3,1)
, maxProtok    DECIMAL(5,4)
, PRIMARY KEY (tvBroj, sifProizv)
, FOREIGN KEY (tvBroj, sifProizv)
    REFERENCES ured (tvBroj, sifProizv));
```

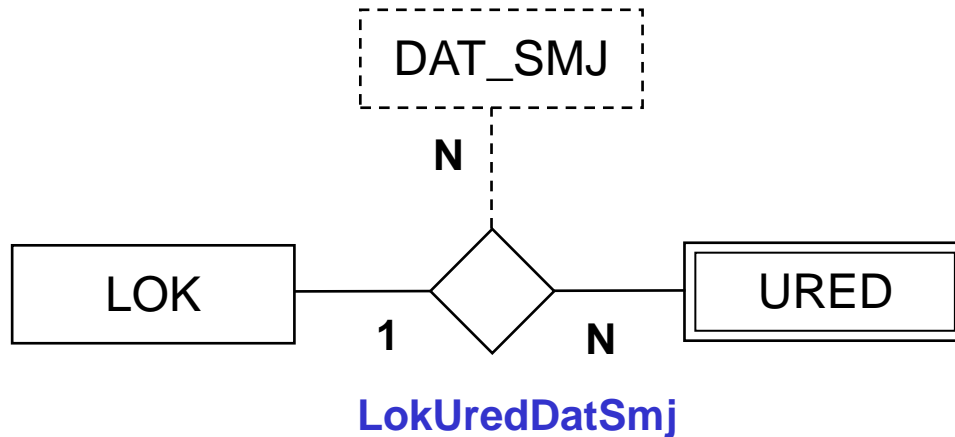

Rješenje: Relacijski model (nastavak)

```
CREATE TABLE ventil (  
    tvBroj          CHAR(20)  
    , sifProizv     INTEGER  
    , promjNavoja   DECIMAL(3,1)  
    , nacZatvar     DECIMAL(5,4)  
    , PRIMARY KEY (tvBroj, sifProizv)  
    , FOREIGN KEY (tvBroj, sifProizv)  
      REFERENCES ured (tvBroj, sifProizv));
```

```
CREATE TABLE reduktor (  
    tvBroj          CHAR(20)  
    , sifProizv     INTEGER  
    , ulTlak        DECIMAL(6,2)  
    , izlTlak       DECIMAL(6,2)  
    , promjNavoja   DECIMAL(3,1)  
    , PRIMARY KEY (tvBroj, sifProizv)  
    , FOREIGN KEY (tvBroj, sifProizv)  
      REFERENCES ured (tvBroj, sifProizv));
```

Rješenje:

- b) Pretpostavi li se da jedan uređaj ne može biti premješten više nego jedan puta na dan, segment ER modela će se promijeniti na sljedeći način:



U odnosu na rješenje pod a), u novom relacijskom modelu potrebno je izbaciti atribut sifLok iz relacije ured, te dodati novu relaciju lokUredDatSmj

```
CREATE TABLE lokUredDatSmj (  
    tvBroj          CHAR(20)  
    , sifProizv     INTEGER  
    , datSmjestaj   DATE  
    , sifLok        INTEGER NOT NULL  
    , PRIMARY KEY (tvBroj, sifProizv, datSmjestaj)  
    , FOREIGN KEY (tvBroj, sifProizv)  
        REFERENCES ured (tvBroj, sifProizv)  
    , FOREIGN KEY (sifLok)  
        REFERENCES lok(sifLok));
```

4. Model baze podataka automehaničarske radionice

Evidentirati podatke o automobilima. Automobil je identificiran tvorničkim brojem (ne postoje dva automobila s istim tvorničkim brojem). Za automobil treba evidentirati godinu proizvodnje i model automobila. Modeli automobila identificirani su proizvođačem i nazivom modela (međusobno različiti proizvođači mogu svoje modele nazivati istim imenom - npr. Renault može imati svoj model naziva Europa, a Opel može imati sasvim drugi model koji se također naziva Europa). Za model automobila evidentira se godina u kojoj je model prvi puta proizveden. Proizvođač ima naziv, a identificiran je svojom šifrom.

U radionici je napravljen popis vrsta poslova koji se mogu obavljati na automobilima. Vrste poslova su šifrirane, a osim šifre i opisa vrste posla (npr. "Izmjena ulja", "Podešavanje ventila", itd.), za svaku vrstu posla se evidentira normativom zadano trajanje izraženo u minutama (koliko bi vremena mehaničar trebao utrošiti obavljajući posao te vrste). Vrste poslova su nezavisne od modela automobila - npr. "Izmjena ulja" je uvijek jednak posao neovisno od modela automobila na kojem se obavlja.

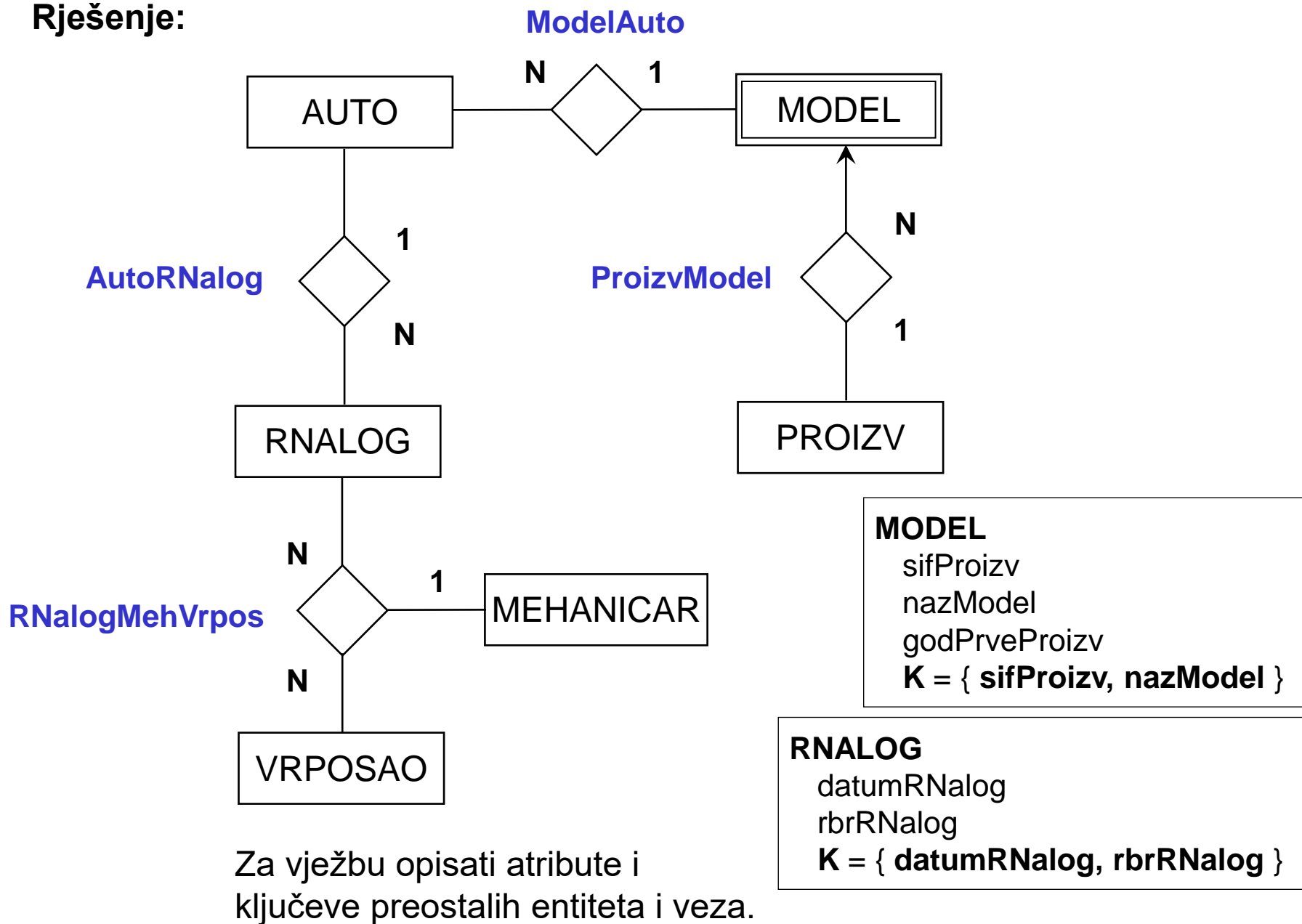
Za mehaničare zaposlene u radionici evidentira se jmbg, prezime i ime.

Za svaki dolazak automobila u radionicu otvara se jedan Radni nalog na kojem se evidentira automobil i datum dolaska automobila u radionicu. Isti automobil može biti primljen u radionicu više puta (čak i istog dana), ali se svaki put otvara novi Radni nalog. Radni nalog nema šifru. Radni nalog pri otvaranju dobiva svoj redni broj, pri čemu svakog dana redni brojevi naloga započinju ponovo s brojem jedan. Za isti datum ne postoje dva Radna naloga s istim brojem.

Uz Radni nalog se evidentira koji mehaničari će obaviti koje vrste poslova na automobilu. Poslove koji su zadani na Radnom nalogu može obaviti jedan ili nekoliko mehaničara, ali jedan zadani posao će jedan mehaničar obaviti sam od početka do kraja. Mehaničari na raznim Radnim nalogima mogu obavljati različite vrste poslova. Mehaničar odmah po obavljenom poslu na nekom automobilu evidentira koliko je vremena u minutama zaista utrošio na obavljanje tog posla (to se vrijeme može razlikovati od normativom zadanog vremena).

Nacrtati ER model i opisati entitete i veze. Sve sheme moraju zadovoljavati 3NF.

Rješenje:



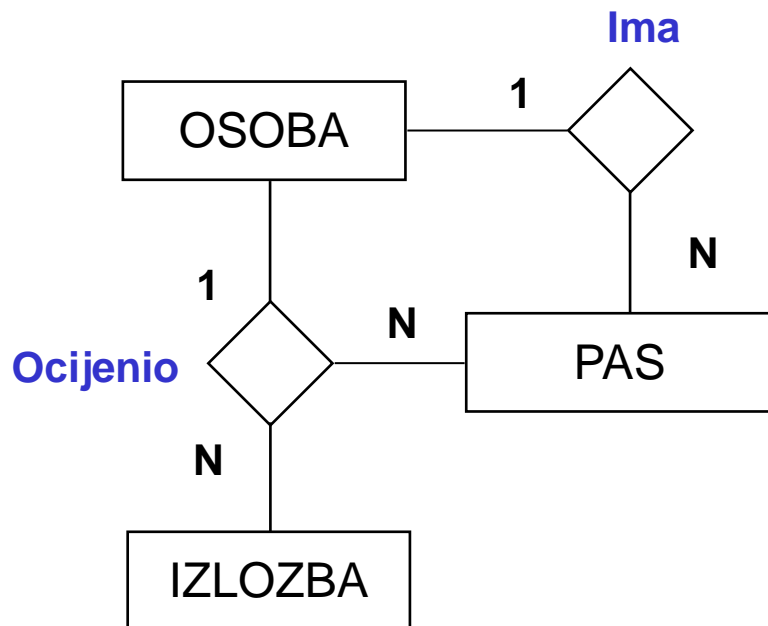
5. Model baze podataka za izložbe pasa

Za svaku se osobu evidentira jmbg, prezime i ime. Za psa se evidentira broj markice koja identificira psa, ime psa, datum okota i osoba koja je vlasnik tog psa. Pretpostavlja se da jedna osoba može imati više pasa, a pas pripada samo jednoj osobi.

Neki vlasnici vode svoje pse na izložbe pasa. Za izložbu se evidentira šifra izložbe koja ju jedinstveno identificira i datum izložbe. Za jednog psa na jednoj izložbi treba evidentirati samo jednu ocjenu i osobu koja ga je ocjenjivala. Ista osoba na jednoj izložbi može ocijeniti više pasa. Ista osoba može ocjenjivati istog psa na više različitih izložbi. Za osobe koje ocjenjuju pse također se evidentiraju jmbg, prezime i ime. Te osobe mogu istovremeno biti i vlasnici pasa.

Nacrtati ER model i opisati entitete i veze. Sve sheme moraju zadovoljavati 3NF.

Rješenje:



OSOBA

jmbg
imeOso
prezOso
K = { jmbg }

PAS

brMarkice
imePas
datOkota
K = { brMarkice }

IZLOZBA

siflzlozba
datlzlozba
K = { siflzlozba }

Ima

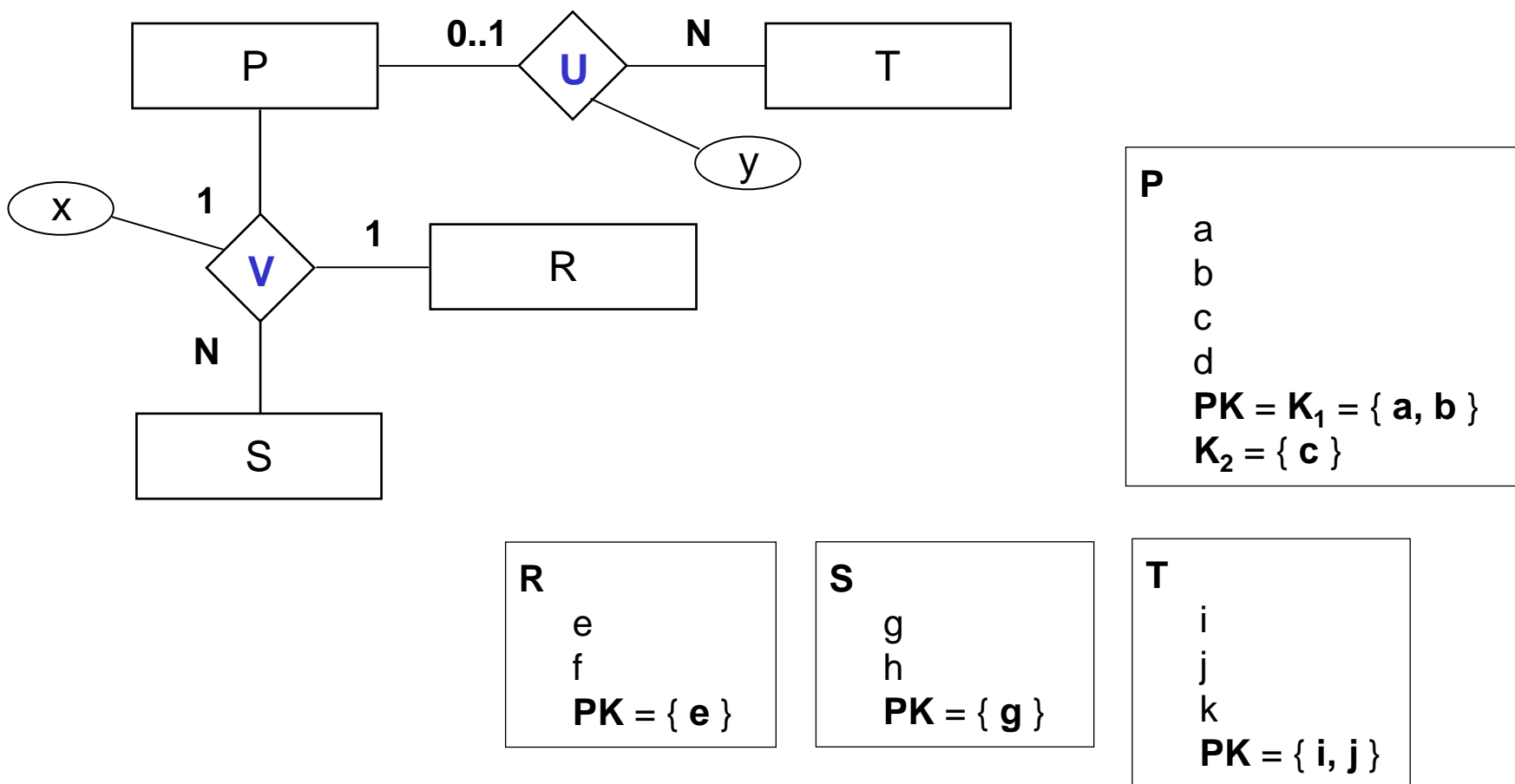
brMarkice
jmbg
K = { brMarkice }

Ocijenio

brMarkice
siflzlozba
jmbg
ocjena
K = { siflzlozba, brMarkice }

6. Zadan je ER model i pripadne sheme entiteta. Na slici su prikazani samo vlastiti atributi veza.

Definirati sheme veza. Napisati SQL naredbe za kreiranje relacija relacijskog modela. Tipove podataka ne treba navoditi. Naredbe moraju sadržavati definicije integritetskih ograničenja.



Rješenje:

Sheme veza

V

a

b

e

g

x

$PK = K_1 = \{ a, b, g \}$

$K_2 = \{ e, g \}$

U

i

j

a

b

y

$PK = \{ i, j \}$

Relacijski model

```
CREATE TABLE p (  
  a ...  
  , b ...  
  , c ...  
  , d ...  
  , PRIMARY KEY (a, b)  
  , UNIQUE (c));
```

```
CREATE TABLE r (  
  e ...  
  , f ...  
  , PRIMARY KEY (e));
```

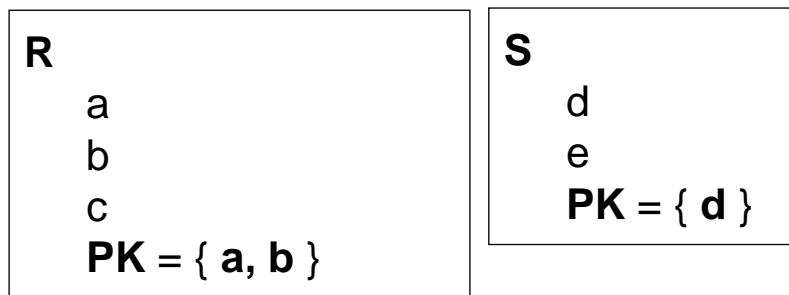
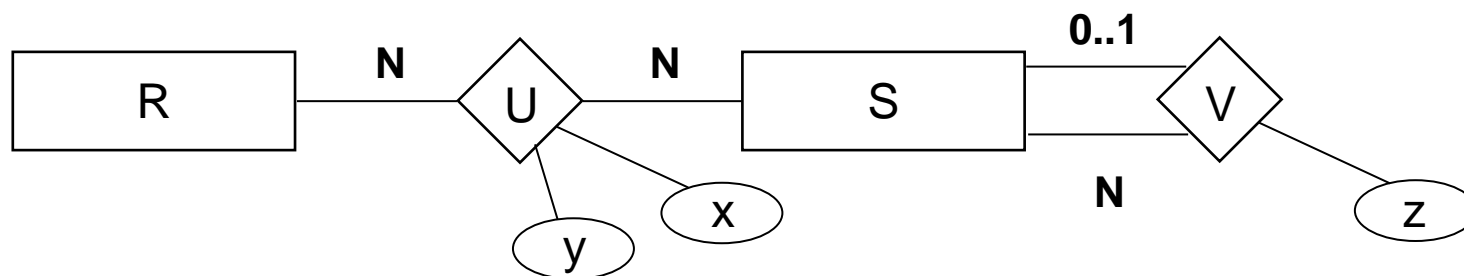
```
CREATE TABLE s (  
  g ...  
  , h ...  
  , PRIMARY KEY (g));
```

```
CREATE TABLE t (  
  i ...  
  , j ...  
  , k ...  
  , y ...  
  , a ...  
  , b ...  
  , PRIMARY KEY (i, j)  
  , FOREIGN KEY (a, b)  
    REFERENCES p (a, b));
```

```
CREATE TABLE v (  
  a ...  
  , b ...  
  , e ...  
  , g ...  
  , x ...  
  , PRIMARY KEY (a, b, g)  
  , UNIQUE (e, g)  
  , FOREIGN KEY (a, b)  
    REFERENCES p (a, b)  
  , FOREIGN KEY (e)  
    REFERENCES r (e)  
  , FOREIGN KEY (g)  
    REFERENCES s (g));
```

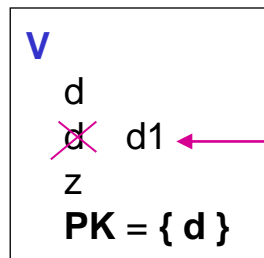
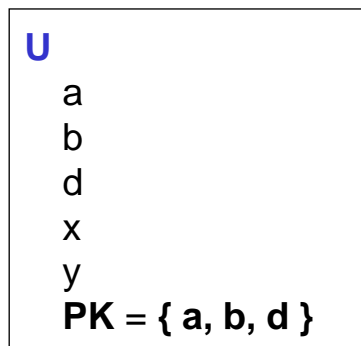
7. Zadan je ER model i pripadne sheme entiteta. Na slici su prikazani samo vlastiti atributi veza.

Definirati sheme veza. Napisati SQL naredbe za kreiranje relacija relacijskog modela. Tipove podataka ne treba navoditi. Naredbe moraju sadržavati definicije integritetskih ograničenja.



Rješenje:

Sheme veza



Preimenovati jedan
od atributa

Relacijski model

```
CREATE TABLE r (  
    a    ...  
    , b    ...  
    , c    ...  
    , PRIMARY KEY (a, b));
```

```
CREATE TABLE u (  
    a    ...  
    , b    ...  
    , d    ...  
    , x    ...  
    , y    ...  
    , PRIMARY KEY (a, b, d)  
    , FOREIGN KEY (a, b) REFERENCES r (a, b)  
    , FOREIGN KEY (d) REFERENCES s (d));
```

```
CREATE TABLE s (  
    d    ...  
    , d1    ...  
    , e    ...  
    , z    ...  
    , PRIMARY KEY (d)  
    , FOREIGN KEY (d1) REFERENCES s (d));
```