

Cvetana Krstev

3	Relacioni model podataka.....	2
3.1	Prikaz relacionog modela podataka	2
3.1.1	Prezentacija modela	2
3.1.2	Konstitutivni elementi relacionog modela	2
3.2	Prelazak sa modela entitet-veze na relacioni model	4
3.2.1	Opšta pravila	4
3.2.2	Poseban slučaj veze 1, 1.....	6
3.2.3	Poseban slučaj entiteta koji ima samo atribut koji je i ključ.....	7
3.2.4	Kompletan primer	7
3.3	Normalizacija entiteta i veza.....	8
3.3.1	Minimalnost ključa	8
3.3.2	Prva normalna forma (1NF).....	9
3.3.3	Druga normalna forma (2NF)	9
3.3.4	Treća normalna forma (3NF)	10
3.3.5	Bojs-Kodova normalna forma.....	11
3.3.6	Primer.....	11
3.4	Relaciona algebra.....	14
3.4.1	Selekcija	14
3.4.2	Projekcija	15
3.4.3	Unija.....	15
3.4.4	Presek.....	16
3.4.5	Razlika	16
3.4.6	Dekartov proizvod	17
3.4.7	Spajanje.....	17
3.4.8	Primeri.....	19

3 Relacioni model podataka

3.1 Prikaz relacionog modela podataka

3.1.1 Prezentacija modela

Relacioni model predstavlja bazu podataka kao skup tabela, ne govoreći pri tome ništa o tome kako će informacije biti uskladištene u računar. Tabele predstavljaju, prema tome, *logičku strukturu* relacionog modela. Na fizičkom nivou, sistem za upravljanje bazama podataka može slobodno da koristi bilo koju tehniku skladištenja podataka – sekvencijalne datoteke, indeksne datoteke, haš tabele, nizove pokazivača... – sve dok se ta struktura može povezati na logičkom nivou sa tabelama. Tabele, dakle, predstavljaju samo apstrakciju fizičkog zapisivanja podataka u memoriji računara. Na neformalan način se relacioni model može ovako definisati:

- Podaci su organizovani u obliku dvodimenzionalnih tabela, koje se još zovu i relacije, čije se vrste nazivaju *torke* (od engl. *tuples*);
- Podacima se manipuliše preko operatora relacione algebre;
- Koherentno stanje baze podataka obezbeđuje skup uslova integriteta.

3.1.2 Konstitutivni elementi relacionog modela

Definicija 3.1 – Atribut – Atribut je jedno ime koje opisuje informacije uskladištene u bazi podataka.

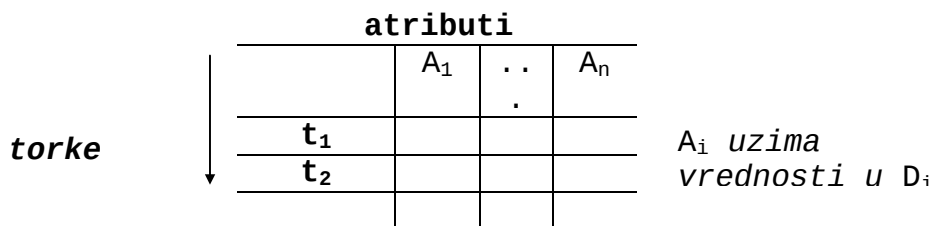
Primeri atributa su: starost osobe, ime osobe, matični broj osobe.

Definicija 3.2 – Domen – Domen atributa je konačan ili beskonačan skup mogućih vrednosti atributa.

Na primer, domen atributa „matični broj osobe“ je skup svih trinaestocifrenih brojeva, dok je domen atributa „ime osobe“ niska slova. U univerzitetskoj praksi za atribut *ocena* je $dom(ocena) = [5,10]$.

Definicija 3.3 – Relacija – Relacija je jedan podskup Dekartovog proizvoda n domena atributa ($n > 0$).

Relacija se predstavlja u obliku dvodimenzionalne tabele u kojoj n atributa odgovara kolonama tabele. Prikazana šematski, tabela pridružena jednoj relaciji „vizuelizuje“ tu relaciju na način predstavljen na Sl. 3-1.



Sl. 3-1 Svaka kolona u tabeli odgovara atributu relacije, a vrsta torki

Definicija 3.4 – *Šema relacije* – Šemom relacije se precizira ime relacije, lista njenih atributa i njihovi domen.

OSOBA		
Matični broj	Ime	Prezime
2204989718136	Suzana	Topalović
1511989517134	Darja	Kovrlija
1212989713132	Valentina	Tasić
2903989515138	Jelena	Korać
2611990719136	Milica	Rosić
0705988915136	Darja	Kovrlija

Sl. 3-2 Primer šeme relacije OSOBA(matični broj; ime: niska; prezime: niska)

Definicija 3.5 – *Stepen relacije* – Stepem relacije je broj njenih atributa.

Definicija 3.6 – *Primerak, n-torka ili torka* – Torka je jedan element skupa koji čini relaciju. Drugim rečima, torka je jedan red u tabeli koja predstavlja relaciju.

Na primer, $\langle 2611990719136, \text{Milica, Rosić} \rangle$ je jedna torka relacije OSOBA koja je prikazana na Sl. 3-2.

Definicija 3.7 – *Kardinalnost* – Kardinalnost relacije je broj torki jedne relacije (broj elemenata skupa).

Kardinalnost relacije OSOBA koja je prikazana na Sl. 3-2 je 6.

Definicija 3.8 – *Kandidat za ključ* – Kandidat za ključ jedne relacije je minimalan skup atributa relacije čije vrednosti identifikuju sve pojedinačne torke relacije.

Prema tome, kandidat za ključ mora da ima različite vrednosti za sve torke jedne relacije. Svaka relacija mora da ima bar jedan kandidat za ključ, a može ih imati i više. Odatle sledi da u jednoj relaciji ne mogu postojati dve identične torke – vrednosti atributa bar tog jednog obaveznog kandidata za ključ se moraju razlikovati.

Svi ključevi kandidati ne moraju biti sastavljeni od istog broja atributa. Takođe, kandidat za ključ može da bude jedan proizvoljno izabrani atribut koji nema nikakvu drugu funkciju osim da posluži kao ključ.

Definicija 3.9 – Primarni ključ – Primarni ključ jedne relacije je jedan odabrani kandidat za ključ. Da bi bilo jasno šta je primarni ključ relacije, njegovi atributi su obično podvučeni.

Definicija 3.10 – Strani ključ – Strani ključ jedne relacije čine jedan ili više atributa koji grade kandidat za ključ neke druge relacije. Pažnja! Strani ključ jedne relacije **ne mora** da bude kandidat za ključ te iste relacije.

Definicija 3.11 – Relaciona šema – Relacionu šemu čini skup šema relacija, sa odabranim primarnim i stranim ključevima.

Definicija 3.12 – Relaciona baza podataka – Relacionu bazu podataka čine skupovi torki relacija opisanih različitim relacionim šemama.

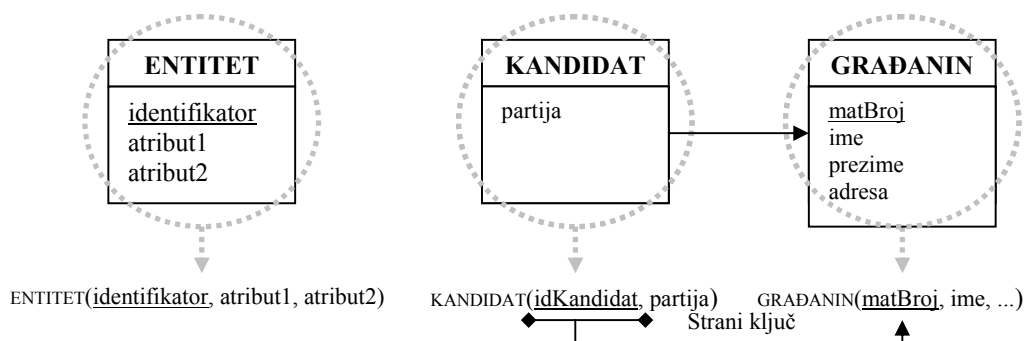
3.2 Prelazak sa modela entitet-veze na relacioni model

3.2.1 Opšta pravila

Da bi se jedan model entitet-vaze preveo u relacioni model treba slediti sledeća pravila.

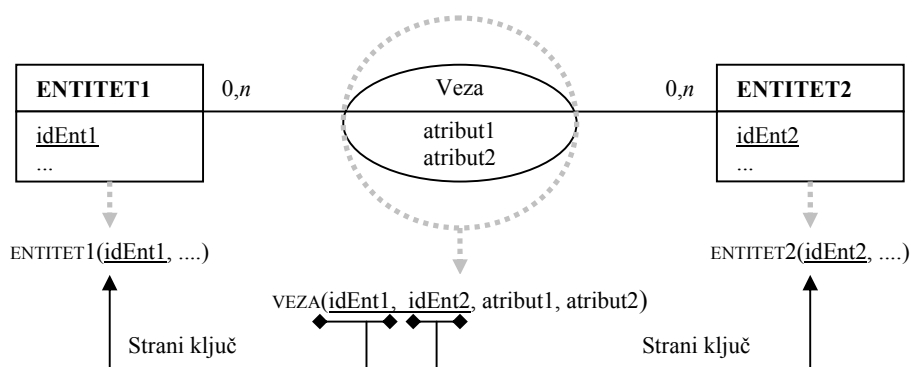
1. Normalizaciju treba uvek obaviti pre prevođenja u relacioni model (postupak normalizacije će biti objašnjen u odeljku 3.3). Ona se u praksi često obavlja *a posteriori* što dovodi do povećanja obima posla u kreiranju baze podataka i proizvodi relacionu šemu koja nije saglasna sa modelom entitet-veze.
2. Svaki entitet postaje jedna šema relacije. Svaki atribut entiteta postaje atribut šeme relacije. Identifikator entiteta se zadržava kao ključ šeme relacije (videti Sl. 3-3a). Treba obratiti pažnju na specifične entitete

(veza specijalizacije) koji dovode do pojave najmanje jednog stranog ključa (videti Sl. 3-3b)



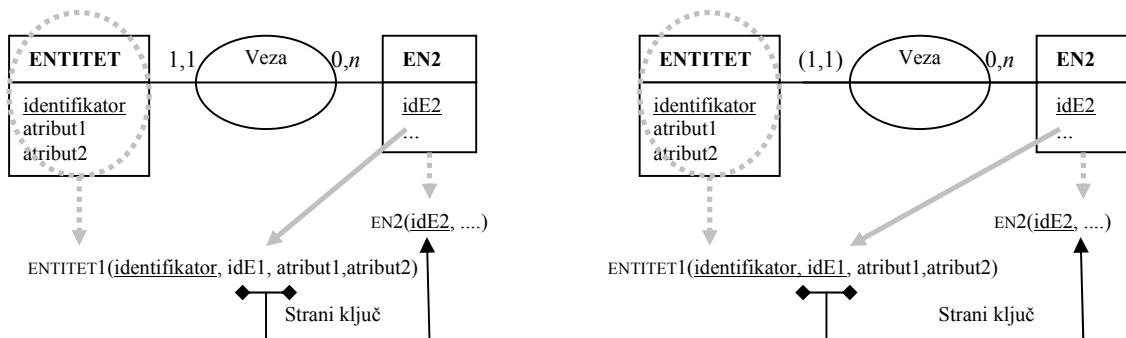
Sl. 3-3 (a) Model sa leve strane: svaki entitet postaje šema relacije; (b) model sa desne strane: identifikator specifičnog entiteta se prevodi u ključ koji je istovremeno i strani ključ.

3. Svaka veza tipa (m,n) (na svim putevima ove veze je maksimalna kardinalnost n) se prevodi u šemu relacije. Svaki atribut veze postaje atribut šeme relacije. Identifikator se dobija od skupa identifikatora entiteta koji učestvuju u vezi. Svaki od ovih identifikatora postaje strani ključ koji se referiše na šemu relacije koja odgovara entitetu iz koga identifikator dolazi. Ovaj slučaj je ilustrovan na Sl. 3-4.



Sl. 3-4 Svaka veza tipa (m,n) postaje jedna šema relacije čiji se primarni ključ gradi od stranih ključeva.

4. Veza na čijem je jednom putu maksimalna kardinalnost 1 (ova veza treba da bude binarna i da nema svoje attribute) ne postaje šema relacije. Ona opisuje zapravo funkcionalnu zavisnost. Šemi relacije koja odgovara entitetu na čijem putu ka vezi je maksimalna kardinalnost 1 se dodaje kao atribut (i prema tome kao strani ključ) identifikator drugog entiteta ove veze. Treba obratiti pažnju da ako put odgovara identifikujućoj vezi, onda atribut koji se dodaje ulazi u ključ šeme relacije. Ovaj korak je ilustrovan na Sl. 3-5.

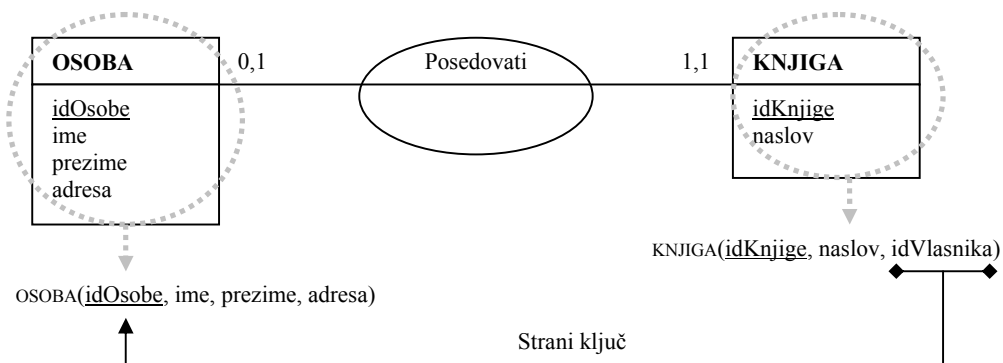


Sl. 3-5 (a) Funkcionalna veza se ne prevodi u novu šemu relacije, već samo u jedan strani ključ (levo); (b) ako put odgovara identifikujućoj vezi, atribut koji se dodaje ulazi u ključ (desno).

3.2.2 Poseban slučaj veze 1, 1

U primeru na Sl. 3-6 maksimalne kardinalnosti na svim putevima veze *Posedovati* su 1. Primena pravila za prevođenje modela entitet-veze u relacioni model koja smo do sada uveli bi dala sledeće:

- $OSOBA(\underline{idOsobe}, ime, prezime, adresa, idKnjige)$ gde je *idKnjige* strani ključ koji se referiše na šemu relacije KNJIGA.
- $KNJIGA(\underline{idKnjige}, naziv, idOsobe)$ gde je *idOsobe* strani ključ koji se referiše na šemu relacije OSOBA.



Sl. 3-6 Veza tipa 1:1 se ne prevodi u šemu relacije

Pošto je veza *Posedovati* tipa 1:1 ona je potpuno realizovana šemom relacije KNJIGA preko stranog ključa *idOsobe*. Prema tome, potpuno je nepotrebno da se ona ponovo realizuje u šemi relacije OSOBA (ili obrnuto). Zbog toga treba izostaviti atribut *idKnjige* u šemi OSOBA, ili obrnuto, atribut *idOsobe* u entitetu KNJIGA. Kardinalnost 0,1 nam govori šta je pravi izbor:

osoba ne mora obavezno da poseduje knjigu, a svaku knjigu mora neko da poseduje, prema tome treba zadržati atribut *idOsobe* u entitetu KNJIGA.

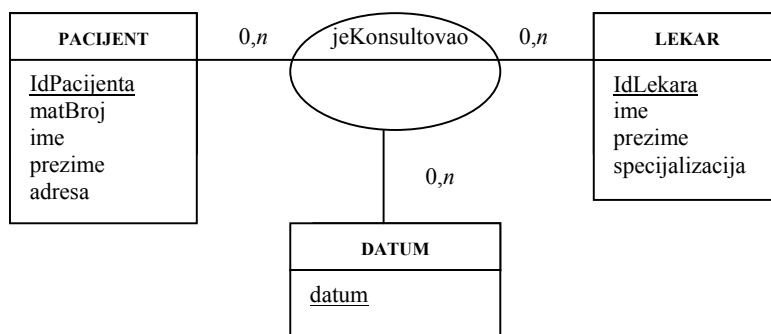
Odgovarajuća relaciona šema je prema tome:

- OSOBA(*idOsobe*, ime, prezime, adresa);
- KNJIGA(*idKnjige*, naziv, *idVlasnika*) gde je *idVlasnika* novo ime za *idOsobe*, strani ključ, koji se referiše na šemu relacije OSOBA.

3.2.3 Poseban slučaj entiteta koji ima samo atribut koji je i ključ

Ako neki entitet ima samo atribut koji je i ključ nije neophodno da se on prevede u šemu relacije. Na primer, entitet DATUM sa Sl. 3-7 ne treba prevoditi u zasebnu šemu relacije jer ona ne bi nosila nikakvu informaciju. Adekvatna relaciona šema koja bi odgovarala modelu entitet-veze sa ove slike bila bi:

- PACIJENT(*idPacijent*, matBroj, ime, prezime, adresa);
- LEKAR(*idLekara*, ime, prezime, specijalizacija);
- JEKONSULTOVAO(*idPacijenta*, *idLekara*, *datum*) gde su *idPacijenta* i *idLekara* strani ključevi koji se referišu na šeme PACIJENT i LEKAR.



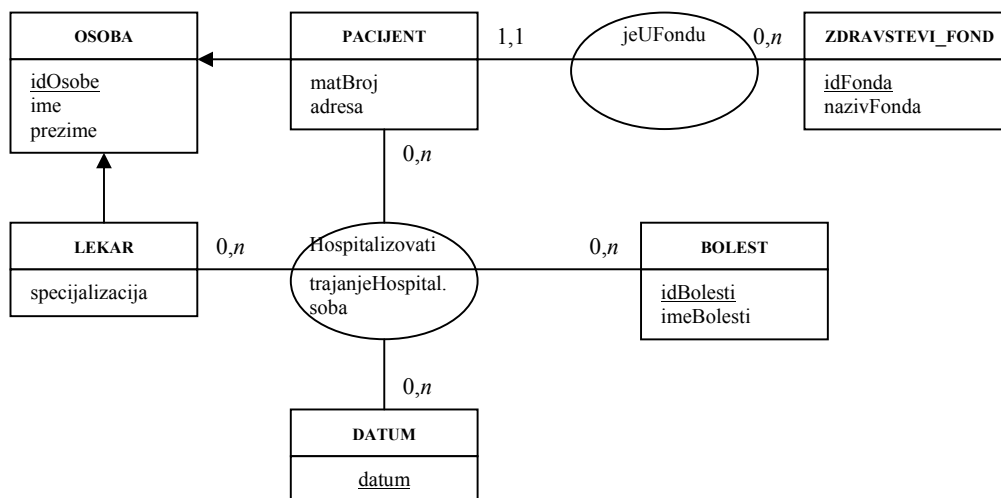
Sl. 3-7 Entitet datum iz ovog modela ne mora da se prevede u zasebnu šemu

3.2.4 Kompletan primer

Sledeća relaciona šema je izvedena iz modela entitet-veze sa Sl. 3-8.

- OSOBA(*idOsoba*, ime, prezime);
- LEKAR(*idLekara*, specijalizacija), gde je atribut *idLekara* strani ključ koji referiše šemu OSOBA;

- PACIJENT(idPacijenta, matBroj, adresa, zdravstveniFond), gde su idPacijenta i zdravstveniFond strani ključevi koji referišu šeme OSOBA i FOND;
- FOND(idFonda, nazivFonda);
- BOLEST(idBolesti, nazivBolesti)
- HOSPITALIZOVATI(idPacijenta, idBolesti, idLekara, datumUlaska, soba, trajanjeHospitalizacije), gde su idPacijenta, idBolesti i idLekara strani ključevi koji referišu šeme PACIJENT, BOLEST i LEKAR.



Sl. 3-8 Primer modela entite-veze - rad bolnice

3.3 Normalizacija entiteta i veza

Normalne forme predstavljaju stepene kvaliteta razvijenih modela koji omogućavaju izbegavanje redundantnosti kao potencijalnog izvora različitih anomalija. Normalizacije se može obaviti i na modelu entitet-veze i na relacionom modelu.

Postoji pet normalnih formi i još dve dodatne. Što je nivo normalizacije viši to je model bolji – u njemu ima manje anomalija. U principu važi da ako je model u n normalnoj formi, on je sigurno i u $n-1$ normalnoj formi.

3.3.1 Minimalnost ključa

Svođenje ključa na minimalan skup atributa predstavlja prvu etapu u normalizaciji:

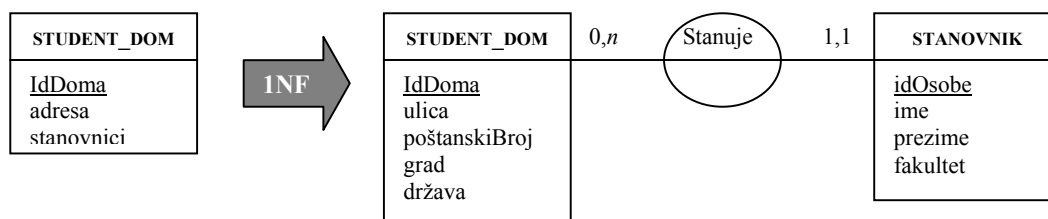
Ključ je minimalan skup atributa koji dozvoljavaju da se svaki primerak (entiteta ili veze) jednoznačno identifikuje.

Svi entiteti i veze imaju bar jedan ključ a mogu ih imati i više. Osim za Bojs-Kod normalnu formu (Boyce-Codd), normalizacije se vrši imajući u vidu neki zadati ključ.

3.3.2 Prva normalna forma (1NF)

Definicija 3.13 – *Prva normalna forma (1NF)* – Entitet ili veza je u prvoj normalnoj formi ako su svi njegovi atributi elementarni (ne mogu se rastaviti).

Da bi entitet ili veza bili u 1NF treba dekomponovati složene attribute – na primer, uvesti attribute *ulica*, *poštanskiBroj*, *grad*, *država* umesto složenog atributa *adresa* – ili treba složeni atribut zameniti novim entitetom, kao u slučaju Sl. 3-9.



Sl. 3-9 Primer normalizacije u prvu normalnu formu

3.3.3 Druga normalna forma (2NF)

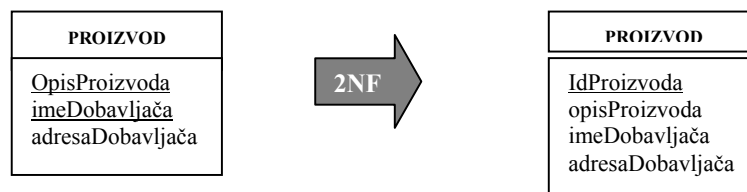
Definicija 3.13 – *Druga normalna forma (2NF)* – Entitet ili veza je u drugoj normalnoj formi ako i samo ako:

- je u prvoj normalnoj formi, i
- ni jedan njegov atribut koji nije deo ključa ne zavisi samo od nekog dela tog ključa.

Drugim rečima, atributi treba da zavisi od celog skupa atributa koji učestvuju u ključu. Prema tome, ako je ključ sveden na samo jedan atribut ili ako sadrži sve attribute, entitet ili veza su obavezno u 2NF. Entitet ili veza mogu da budu u drugoj normalnoj formi u odnosu na jedan kandidat za ključ, a da ne budu u odnosu na neke druge kandidate.

Sl. 3-10 prikazuje entitet PROIZVOD koji opisuje proizvode koje isporučuju razni dobavljači. Pretpostavljamo da jedna dobavljač može da

isporuči više proizvoda i da jedan proizvod može da isporučuje više dobavljača. U tom slučaju ni *opisProizvoda* ni *imeDobavljača* ne mogu sami da budu ključ entiteta PROIZVOD. Nasuprot tome, ključ (*opisProizvoda*, *imeDobavljača*) jeste identifikator entiteta PROIZVOD. Međutim, atribut *adresaDobavljača* zavisi samo od dela ključa – *imeDobavljača*. Jedno rešenje (neadekvatno) je da se uvede novi atribut *idProizvoda* koji će biti ključ, čime entitet prelazi u 2NF. Videćemo u sledećem odeljku bolje rešenje.



Sl. 3-10 Primer prevođenja u drugu normalnu formu. Ovde pretpostavljamo da jedan dobavljač isporučuje više proizvoda i da jedan proizvod može da isporuči više dobavljača.

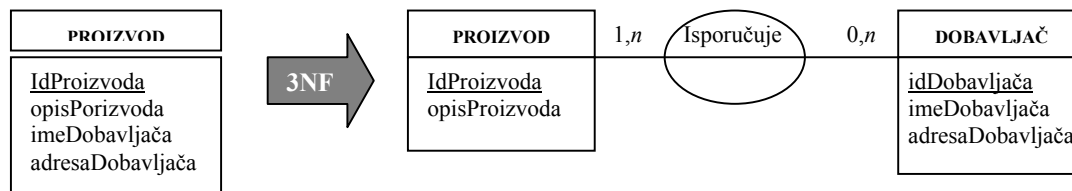
3.3.4 Treća normalna forma (3NF)

Definicija 3.14 – *Treća normalna forma (3NF)* – Entitet ili veza je u trećoj normalnoj formi ako i samo ako:

- je u drugoj normalnoj formi, i
- ni jedan njegov atribut koji nije deo ključa ne zavisi od nekog skupa atributa koji nisu u ključu.

Prema tome, ako je entitet ili veza u drugoj normalnoj formi i ima samo jedan atribut koji nije u ključu on je obavezno u 3NF. Entitet ili veza mogu da budu u trećoj normalnoj formi u odnosu na jedan kandidat za ključ, a da ne budu u odnosu na neke druge kandidate.

Tokom prevođenja u 3NF mogu se otkriti i formulisati neki skriveni entiteti kao što prikazuje Sl. 3-11.



Sl. 3-11 Primer prevođenja u treću normalnu formu. U ovom primeru atribut *adresaDobavljača* zavisi od *imenaDobavljača* (zapravo *idDobavljača*)

3.3.5 Bojs-Kodova normalna forma

Definicija 3.15 – *Bojs-Kodova normalna forma (BKNF)* – Entitet ili veza je u Bojs-Kodovoj normalnoj formi ako i samo ako postoji samo takva međuzavisnost atributa u kojoj ključ određuje atribut koji nije ključ.

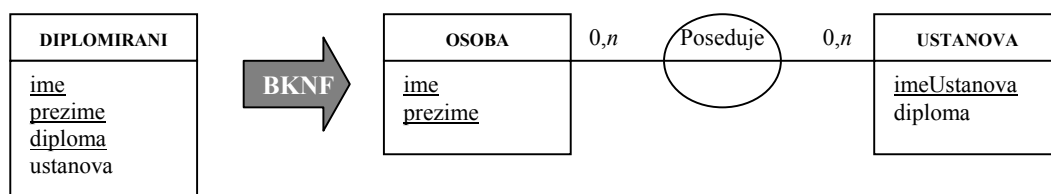
Entitet ili veza koji je u Bojs-Kodovoj normalnoj formi za jedan kandidat za ključ je i za sve ostale kandidate.

Pogledajmo primer entiteta DIPLOMIRANI koji modelira osobu (*ime* i *prezime*) koja je stekla neku diplomu (*diploma*) kod neke obrazovne ustanove (*ustanova*). Pretpostavimo da nije moguća situacija u kojoj je jedna osoba stekla dve iste diplome, ali da može da ima više različitih diploma. Jedna ustanova izdaje samo jednu vrstu diploma, ali istu diplomu može da izdaje više obrazovanih ustanova. Entitet DIPLOMIRANI ima dva kandidata za ključ:

- (*ime, prezime, ustanova*)
- (*ime, prezime, diploma*).

Entitet DIPLOMIRANI nije u 2NF u odnosu na ključ (*ime, prezime, ustanova*) jer *diploma* zavisi od *ustanove*. Nasuprot tome, entitet DIPLOMIRANI jeste u 3NF u odnosu na ključ (*ime, prezime, diploma*) – zato što je u 2NF (atribut *ustanova* ne zavisi od dela ključa – od čega uopšte zavisi?) i ima samo jedan atribut izvan ključa. Pa ipak, entitet DIPLOMIRANI nije u BKNF jer postoji zavisnost od atributa koji nije u ključu – *diploma* zavisi od *ustanove*.

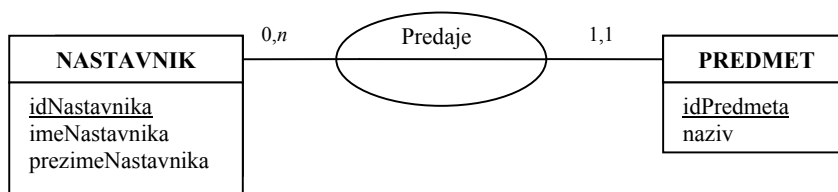
Model koji je u Bojs-Kodovoj normalnoj formi se smatra da je dovoljno dobar da može da se otpočne za implementacijom baze.



Sl. 3-12 Primer prevođenja u Bojs-Kodovu normalnu formu

3.3.6 Primer

Zadatak 1. Sastaviti relacionu šemu za mali model entitet-veze dat na slici, a zatim dati i jedan mali primer relacione baze podataka koja odgovara toj šemi:



NASTAVNIK (*idNastavnika*, *imeNastavnika*, *prezimeNastavnika*)
 PREDMET(*idPredmeta*, *naziv*, *predajeNastavnik*), gde je *predajeNastavnik* strani ključ koji referiše relaciju NASTAVNIK.

U ovoj relacionoj šemi ne postoji posebna relacija za vezu *Predaje* jer je maksimalna kardinalnost na jednom putu 1, pa se u relaciju koji odgovara entitetu na čijem kraju puta je maksimalna kardinalnost 1 – a to je PREDMET – uključuje i identifikator drugog entiteta.

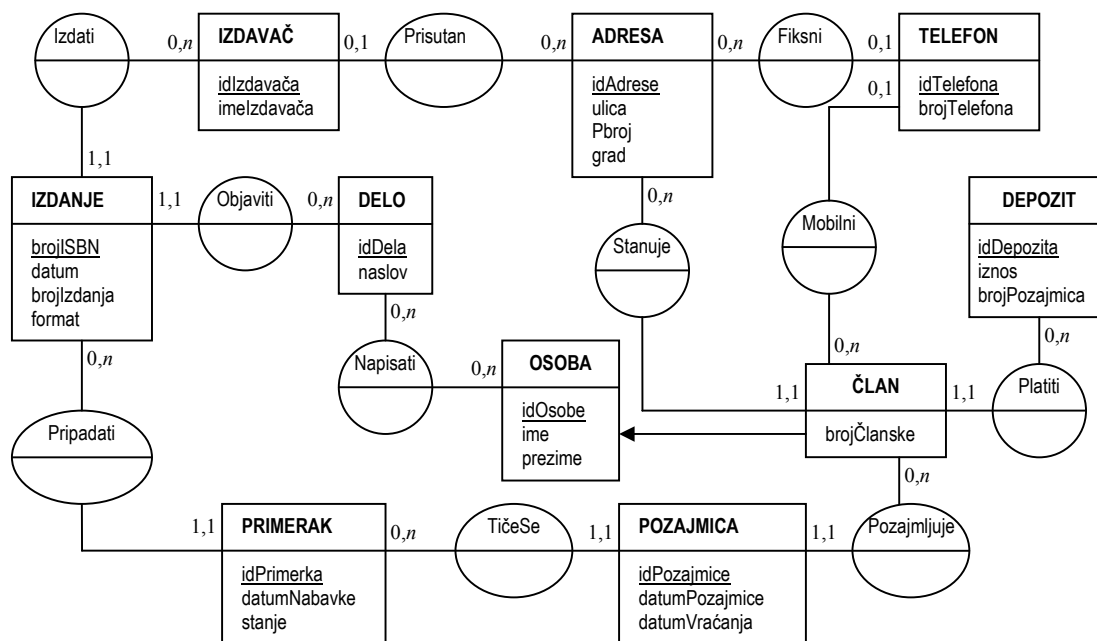
NASTAVNIK		
idNastavnika	ime	prezime
BI02	Cvetana	Krstev
BI01	Aleksandra	Vraneš
BI04	Bojan	Đorđević

PREDMET		
idPred.	naziv	predajeN
0927058	Multimedijalni dokumenti	BI02
0927031	Bibliotekarstvo 1	BI01
0927014	Uvod u arhivistiku	BI04

Zadatak 2. Slučaj jedne biblioteke (drugi deo).

Zahvaljujući automatizaciji član biblioteke će moći da pronade knjigu ako zna kako se ona zove. Moći će da sazna šta je sve pisao jedan autor, listu autora jedne knjige i njenog izdavača. Svaka knjiga je nabavljena u jednom ili u više primeraka. Neke knjige su više puta izdavane od različitih izdavača. Da bi se učlanio, čovek treba da ostavi depozit i svoje podatke – adresu i broj telefona. U zavisnosti od depozita, član može da pozajmi od dve do deset knjiga. Knjiga mora da se vrati za najviše 15 dana. Uz pomoć sistema za upravljanje pozajmicama biće moguće da se u svakom trenutku sazna koje su sve knjige kod nekog člana biblioteke, a takođe i kod koga je neka knjiga koja nije na polici. Sistem za upravljanje pozajmicama omogućić će da se za svaki primerak knjige sazna kada je nabavljena, njeno stanje i da li je na raspolaganju.

U gornjem tekstu treba identifikovati reči koje se konkretizuju entitetima, vezama i atributima. Predložiti identifikatore za sve entitete i veze. Sačiniti dobro formirani model entitet-veze za ovu biblioteku.



SI. 3-13 Model entitet-veze jedne biblioteke (nastavak 2)

Relacioni model:

IZDAVAČ(idIzdavača, imeIzdavača, idAdrese) (– veza *Izdati* je funkcija)

IZDANJE(brojISBN, datum, brojIzdanja, format, idIzdavača, idDela) (– veza *Objaviti* je funkcija; veza *Izdati* je funkcija)

DELO(idDela, naslov)

PRIMERAK(idPrimerka, datumNabavke, stanje, brojISBN) (– veza *Pripadati* je funkcija)

POZAJMICA(idPozajmice, datumPozajmice, datumVraćanja, idPrimerka, idČlana) (– veza *TičeSe* je funkcija; veza *Pozajmljuje* je funkcija)

OSOBA(idOsobe, ime, prezime)

NAPISATI(idDela, idOsobe) (– veza *Napisati* je tipa (m,n))

ČLAN(idČlana, brojČlanske, idDepozita, idAdrese) (– identifikator člana preuzet od osobe; veza *Platiti* je funkcija; veza *Stanuje* je funkcija)

DEPOZIT(idDepozita, iznos, pozajmica)

ADRESA(idAdrese, ulica, pBroj, grad)

TELEFON(idTelefona, brojTelefona, idAdresaFiksni, idČlanMobilni) – veza *Fiksni* je funkcija; veza *Mobilni* je funkcija).

3.4 Relaciona algebra

Relaciona algebra je koherentan matematički mehanizam na koga se oslanja relacioni model. Relaciona algebra uvodi skup elementarnih formalnih operacija nad relacijama s ciljem da se proizvedu nove relacije. Ove operacije omogućavaju da se predstave upiti nad bazama podataka čiji rezultat će biti relacija (predstavljena u obliku tabele). Ovaj formalizam je u jezgru upitnog jezika SQL.

Razlikujemo tri tipa relacionih operacija:

- *Unarne operacije* – (selekcija i projekcija) koje su i najjednostavnije jer omogućavaju proizvodnju novih tabela na osnovu jedne postojeće tabele.
- *Binarne skupovne operacije* – (unija, presek i razlika) omogućavaju da se proizvede nove tabele na osnovu dve relacije koje su istog stepena i domena.
- *Binarne (ili n -arne) operacije* – (Dekartov proizvod, spajanje) – omogućavaju da se proizvede nova relacija na osnovu dve ili više postojećih.

3.4.1 Selekcija

Definicija 3.16 – *Selekcija (restrikcija)* – Restrikcija jedne relacije R generiše novu relaciju u kojoj su samo one torke polazne relacije koje zadovoljavaju zadati logički uslov E . Piše se: $\sigma_{(E)}R$.

Prema tome, selekcija omogućava da se iz tabele odaberu samo neke njene vrste. Rezultat je nova relacija koja ima iste attribute kao i polazna relacija R .

OSOBA		
idOsobe	ime	Prezime
5	Milica	Rosić
1	Ana	Polić
12	Milica	Spasić
3	Svetlana	Veličković

$\sigma_{(ime="Milica")}$ OSOBA		
idOsobe	ime	Prezime
5	Milica	Rosić
12	Milica	Spasić

Tab. 3-1 (a) Primer relacije OSOBA; (b) Selekcija torki iz relacije OSOBA za koje je vrednost atributa *ime* jednaka "Milica"

3.4.2 Projekcija

Definicija 3.16 – *Projekcija* – Projekcija jedne relacije R sastoji se u ukidanju svih atributa osim atributa $A_1, A_2, \dots, A_n$ i u brisanju svih dupliranih torki koje bi time nastale iz nove relacije. Piše se: $\Pi_{(A_1, A_2, \dots, A_n)} R$.

Drugim rečima, selekcija omogućava da se iz tabele odaberu samo neke njene kolone.

OSOBA			$\Pi_{(ime, prezime)} OSOBA$	
idOsobe	ime	Prezime	ime	Prezime
5	Milica	Rosić	Milica	Rosić
1	Ana	Polić	Ana	Polić
12	Milica	Spasić	Milica	Spasić
3	Svetlana	Veličković	Svetlana	Veličković

Tab. 3-2 (a) Primer relacije OSOBA; (b) Restrikcija relacije OSOBA na attribute *ime* i *prezime*

3.4.3 Unija

Definicija 3.17 – *Unija* – Unija dve relacije R_1 i R_2 koje imaju iste šeme je treća relacija koja se sastoji od svih torki koje pripadaju ili relaciji R_1 ili relaciji R_2 , bez dupliranja. Piše se: $R_1 \cup R_2$.

Rezultat unije dve relacije je nova relacija R koja ima istu šemu kao i relacije od kojih je nastala. Ako dve polazne relacije imaju iste torke one neće biti ponovljene u novoj relaciji. Na primer, unija relacija SLIKE i ZVUK iz Tab. 3-3 sadrži samo jednu torku <Milica, Ilić> iako se ona javlja u obe relacije. Smisao nove relacija $SLIKE \cup ZVUK$ je da ona sadrži sve studente koji su se prijavili za grupe za obradu slike ili zvuka za multimedijalni dokument.

SLIKE		ZVUK		SLIKE $\cup$ ZVUK	
Ime	Prezime	Ime	Prezime	Ime	Prezime
Milica	Ilić	Milica	Kevrešan	Milica	Ilić
Marija	Stojadinović	Milica	Spasić	Marija	Stojadinović
Marina	Milošević	Milica	Ilić	Marina	Milošević
				Milica	Kevrešan
				Milica	Spasić

Tab. 3-3 Primer unije dve relacije

3.4.4 Presek

Definicija 3.18 – *Presek* – Presek dve relacije R_1 i R_2 koje imaju iste šeme je treća relacija koja se sastoji od svih torki koje pripadaju i relaciji R_1 i relaciji R_2 . Piše se: $R_1 \cap R_2$.

Rezultat preseka dve relacije je nova relacija R koja ima istu šemu kao i relacije od kojih je nastala. Ako dve polazne relacije imaju iste torke samo one će se ponoviti u novoj relaciji. Na primer, presek relacija SLIKE i ZVUK iz Tab. 3-4 sadrži samo jednu torku <Milica, Ilić>. iako se ona javlja u obe relacije. Smisao nove relacija $SLIKE \cap ZVUK$ je da ona sadrži sve studente koji su se prijavili i za grupu za obradu slike i za grupu za obradu zvuka za multimedijalni dokument.

SLIKE		ZVUK		SLIKE $\cap$ ZVUK	
Ime	Prezime	Ime	Prezime	Ime	Prezime
Milica	Ilić	Milica	Kevrešan	Milica	Ilić
Marija	Stojadinović	Milica	Spasić		
Marina	Milošević	Milica	Ilić		

Tab. 3-4 Primer preseka dve relacije

3.4.5 Razlika

Definicija 3.19 – *Razlika* – Razlika dve relacije R_1 i R_2 koje imaju iste šeme je treća relacija koja se sastoji od svih torki koje pripadaju relaciji R_1 ali ne pripadaju relaciji R_2 . Piše se: $R_1 - R_2$.

Rezultat razlike dve relacije je nova relacija R koja ima istu šemu kao i relacije od kojih je nastala. Ako dve polazne relacije imaju iste torke one će biti isključene iz relacije R_1 . Na primer, razlika relacija SLIKE i ZVUK iz Tab. 3-4 sadrži sve torke iz relacije R_1 osim <Milica, Ilić> jer se ona javlja i u relaciji R_2 . Smisao nove relacija $SLIKE - ZVUK$ je da ona sadrži sve studente koji su se prijavili za grupu za obradu slike ali ne i za grupu za obradu zvuka za multimedijalni dokument.

SLIKE		ZVUK		SLIKE - ZVUK	
Ime	Prezime	Ime	Prezime	Ime	Prezime
Milica	Ilić	Milica	Kevrešan	Marija	Stojadinović
Marija	Stojadinović	Milica	Spasić	Marina	Milošević
Marina	Milošević	Milica	Ilić		

Tab. 3-5 Primer razlike dve relacije

3.4.6 Dekartov proizvod

Definicija 3.20 – *Dekartov proizvod* – Dekartov proizvod dve relacije R_1 i R_2 je treća relacija koja se sastoji od torki koje su dobijene kombinovanjem torki koje pripadaju relaciji R_1 i torki koje pripadaju relaciji R_2 . Piše se: $R_1 \times R_2$.

Rezultat Dekartovog proizvoda dve relacije je nova relacija R koja ima sve atribute relacije R_1 i sve atribute relacije R_2 . Broj torki nove relacije je proizvod broja torki relacije R_1 i broja torki relacije R_2 . Na primer, Dekartov proizvod relacija OSOBA i POKLON iz Tabele 3-6 je nova relacija koja ima atribute *ime*, *prezime* (iz relacije OSOBA), *proizvod* i *cena* (iz relacije OSOBA).

OSOBA	
Ime	Prezime
Milica	Spasić
Katarina	Bekavac

POKLON	
proizvod	cena
knjiga	200
šolja	150
torta	500

OSOBA $\times$ POKLON			
Ime	Prezime	Proizvod	cena
Milica	Spasić	Knjiga	200
Milica	Spasić	Šolja	150
Milica	Spasić	Torta	500
Katarina	Bekavac	Knjiga	200
Katarina	Bekavac	Šolja	150
Katarina	Bekavac	Torta	500

Tab. 3-6 Primer Dekartovog proizvoda koji omogućava da se generišu sve moguće kombinacije osoba i poklona koje mogu da dobiju

3.4.7 Spajanje

Definicija 3.21 – *Spajanje* – Spajanjem dve relacije R_1 i R_2 dobija se treća relacija koja se sastoji od odabira onih torki dobijenih kombinovanjem torki koje pripadaju relaciji R_1 i torki koje pripadaju relaciji R_2 koje zadovoljavaju logički uslov E . Piše se: $R_1 \bowtie_E R_2$.

U stvari, spajanje nije ništa drugo nego relacija koja se dobija primenom restrikcije (selekcije) na Dekartov proizvod dve relacije R_1 i R_2 : $R_1 \bowtie_E R_2 = \sigma_{(E)}(R_1 \times R_2)$. Primer operacije spajanja dat je u Tab. 3-7.

OSOBA		
ime	prezime	uzrast
Milica	Lukić	6
Ana	Čirić	42
Milica	Katić	16
Iva	Matić	6

POKLON		
Uzrast	proizvod	cena
99	knjiga	500
30	CD	600
10	pernica	400

OSOBA $\times_{((uzrast \leq uzrastP) \wedge (cena \leq 500))}$ POKLON					
Ime	prezime	uzrast	uzrastP	proizvod	cena
Milica	Lukić	6	99	knjiga	500
Milica	Lukić	6	10	pernica	400
Ana	Čirić	42	99	knjiga	500
Milica	Katić	16	99	knjiga	500
Iva	Matić	6	99	knjiga	500
Iva	Matić	6	10	pernica	400

Tab. 3-7 Primer operacije spajanja kojom se dobijaju svi mogući pokloni koje osobe mogu da dobiju, a da pokloni odgovaraju njihovom uzrastu ($uzrast \leq uzrastP$) i da su odgovarajući po ceni ($cena \leq 500$)

Definicija 3.22 – Prirodno spajanje – Prirodno spajanje dve relacije R_1 i R_2 je spajanje u kome logički izraz E predstavlja test jednakosti između atributa koji imaju ista imena (i domene) u relacijama R_1 i R_2 . U dobijenoj relaciji ovi atributi se ne dupliraju već čine jednu kolonu. Piše se: $R_1 \times R_2$.

Ako se spajanje obavlja samo nad podskupom zajedničkih atributa onda te attribute treba tačno navesti: $R_1 \times_{A_1, \dots, A_n} R_2$. Primer operacije prirodnog spajanja dat je u Tab. 3-7.

OSOBA		
ime	prezime	Uzrast
Milica	Lukić	6
Ana	Čirić	40
Milica	Katić	20
Iva	Matić	6

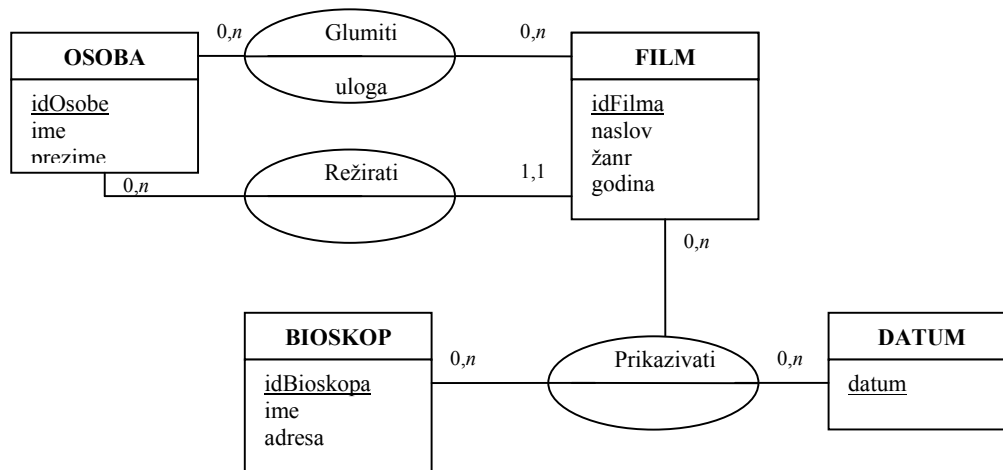
POKLON		
Uzrast	proizvod	cena
40	knjiga	500
6	lutka	1000
20	sat	3000

OSOBA $\times$ POKLON				
Ime	prezime	uzrast	proizvod	cena
Milica	Lukić	6	lutka	1000
Ana	Čirić	40	knjiga	500
Milica	Katić	20	sat	3000
Iva	Matić	6	lutka	1000

Tab. 3-8 Primer prirodnog spajanja ($OSOBA \times POKLON$) - operacija može da se zapiše i kao $OSOBA \times_{uzrast} POKLON$

3.4.8 Primeri

Zadatak 1. (a) Sastaviti relacionu šemu za mali model entitet-veze dat na slici:



OSOBA (idOsobe, ime, prezime)

FILM(idFilma, naslov, žanr, godina, idRežiser), gde je idRežiser strani ključ koji referiše relaciju OSOBA.

GLUMITI(idGlumca, idFilma, uloga), gde je idGlumca strani ključ koji referiše relaciju OSOBA, a idFilma strani ključ koji referiše relaciju FILM.

BIOSKOP(idBioskopa, ime, adresa)

PROJEKCIJA(idBioskopa, idFilma, datum), gde je idBioskopa strani ključ koji referiše relaciju BIOSKOP, a idFilma strani ključ koji referiše relaciju FILM.

U ovoj relacionoj šemi ne postoji posebna relacija za vezu *Režirati* jer je maksimalna kardinalnost na jednom putu 1, pa se u relaciju koji odgovara entitetu na čijem kraju puta je maksimalna kardinalnost 1 – a to je FILM – uključuje i identifikator drugog entiteta.

(b) Neka je jedan primer relacione baze koja koristi relacionu šemu razvijenu pod (a) dat sledećim tabelama:

OSOBA		
idOsobe	ime	prezime
01	Odri	Hepbern
02	Gregori	Pek
03	Keri	Grant
04	Piter	O'Tul
05	Eva Marija	Sent
06	Ingrid	Bergman
07	Hemfri	Bogart
16	Bili	Vajlder

17	Majkl	Kertiz
18	Alfred	Hičkok
19	Frenk	Kapra
20	Stenli	Dana
21	Robert	Maligan
22	Vilijam	Vajlder
23	Ketrin	Hepbern
24	Entoni	Harvi

GLUMITI		
idGlumac	idFilm	Uloga
01	01	princeza
02	01	novinar
01	02	udovica
03	02	službenik
01	03	ćerka
04	03	Naučnik
03	05	unuk
03	06	Reklamni agent
05	06	Ljubavnica

03	07	Vladin agent
06	07	Ćerka naci-zločinca
02	08	doktor
06	08	Doktorka
06	09	Žena francuskog vode otpora
07	09	Vlasnik kafane
01	10	Neugledna devojka
07	10	milioner
04	11	Kralj Henri II
23	11	Kraljica Eleonor

FILM				
idF.	naziv	žanr	idRež.	godina
01	Praznik u Rimu	komedija	22	1953
02	Šarada	komedija	20	1963
03	Kako ukrasti milion dolara	komedija	22	1966
04	Ubiti pticu rugalicu	drama	21	1962
05	Arsenik i stare čipke	komedija	19	1946
06	Sever-severozapad	triler	18	1959
07	Ozloglašena	triler	18	1946
08	Opčinjena	triler	18	1945
09	Kazablanka	ljubavni	17	1942
10	Sabrina	Ljubavni	16	1954
11	Zima jednog lava	istorijski	24	1968

BIOSKOP		
idBioskopa	ime	adresa
01	Kinoteka	Kosovska
02	Dom omladine	Makedonska
03	Kulturni centar	Kolarčeva
04	Studentski grad	Novi Beograd

01	04	02-08-2009
04	03	08-04-2007
03	06	02-12-2007
02	02	08-12-2007
03	03	05-11-2008
04	03	06-11-2007
01	06	05-07-2007
02	04	02-09-2009
04	06	01-08-2010
02	10	03-10-2010
02	09	02-12-2011
01	08	01-02-2010
01	03	02-03-2009
04	11	20-12-2013

PRIKAZIVATI		
idBioskopa	idFilm	Datum
02	05	01-05-2010
02	05	02-05-2010
02	05	03-05-2010
02	04	02-12-2009
01	01	07-05-2009
02	07	09-05-2008

Šta se traži sledećim upitima formulisanim koristeći operacije relacione algebre i kakve odgovore će oni dati na primeru gornje relacione baze?

1. $\sigma_{(godina < "1960")}$ FILM

Sve torke iz relacije FILM kod kojih je vrednost atributa *godina* manja od 1960 – svi filmovi snimljeni pre 1960.

idF.	Naziv	žanr	idRež.	Godina
01	Praznik u Rimu	komedija	22	1953
05	Arsenik i stare čipke	komedija	19	1946
06	Sever-severozapad	triler	18	1959
07	Ozloglašena	triler	18	1946
08	Opčinjena	triler	18	1945
09	Kazablanka	ljubavni	17	1942
10	Sabrina	Ljubavni	16	1954

2. $\sigma_{(godina < "1950" \wedge \text{žanr} = \text{"triler"})}$ FILM

Sve torke iz relacije FILM kod kojih je vrednost atributa *godina* manja od 1960 i vrednost atributa *žanr* jednaka „triler“ – svi triler filmovi snimljeni pre 1960.

idF.	Naziv	žanr	idRež.	Godina
07	Ozloglašena	triler	18	1946
08	Opčinjena	triler	18	1945

3. $\sigma_{(\neg(godina < "1950"))} \sigma_{(\text{žanr} = \text{"komedija"})}$ FILM

Prva restrikcija daje sve torke iz relacije FILM kod kojih je vrednost atributa *žanr* jednaka „komedija“, a druga restrikcija daje sve torke iz tako proizvedene relacije kod kojih vrednost atributa *godina* nije manja od 1950 – sve komedije snimljene posle 1950.

I korak:

idF.	Naziv	Žanr	idRež.	Godina
01	Praznik u Rimu	Komedija	22	1953
02	Šarada	komedija	20	1963
03	Kako ukrasti milion dolara	komedija	22	1966
05	Arsenik i stare čipke	komedija	19	1946

II korak:

idF.	Naziv	žanr	idRež.	Godina
01	Praznik u Rimu	komedija	22	1953
02	Šarada	komedija	20	1963
03	Kako ukrasti milion dolara	komedija	22	1966

4. $\Pi_{(\text{naziv}, \text{žanr}, \text{godina})}$ FILM

Sve torke iz relacije FILM sa odabranim atributima – *naziv*, *žanr*, *godina*.

naziv	žanr	godina
Praznik u Rimu	komedija	1953
Šarada	komedija	1963
Kako ukrasti milion dolara	komedija	1966
Ubiti pticu rugalicu	drama	1962
Arsenik i stare čipke	komedija	1946
Sever-severozapad	triler	1959
Ozloglašena	triler	1946
Opčinjena	triler	1945
Kazablanka	ljubavni	1942
Sabrina	Ljubavni	1954

4. $\Pi_{(\text{naziv}, \text{žanr}, \text{godina})} \sigma_{(godina < "1950")}$ FILM

Prva restrikcija daje sve torke iz relacije FILM kod kojih je vrednost atributa *godina* manja od 1950, a druga projekcija bira attribute *naziv*, *žanr*, *godina* – nazivi, žanr i godina snimanja svih filmova snimljenih pre 1950. godine.

I korak

idF.	naziv	žanr	idRež.	Godina
05	Arsenik i stare čipke	komedija	19	1946
07	Ozloglašena	triler	18	1946
08	Opčinjena	triler	18	1945
09	Kazablanka	ljubavni	17	1942

II korak

naziv	žanr	godina
Arsenik i stare čipke	komedija	1946
Ozloglašena	triler	1946
Opčinjena	triler	1945
Kazablanka	ljubavni	1942

5. $(\Pi_{(ime, prezime)} \sigma_{(prezime="Hepbern")} OSOBA) \cup (\Pi_{(ime, prezime)} \sigma_{(prezime="Vajlder")} OSOBA)$
Prvi član unije daje iz relacije OSOBA novu relaciju samo sa atributima *ime* i *prezime* i onim torkama kod kojih je vrednost atributa *prezime*="Hepbern".
Drugi član unije daje iz relacije OSOBA novu relaciju samo sa atributima *ime* i *prezime* i onim torkama kod kojih je vrednost atributa *prezime*="Vajlder".
Interesuju nas osobe iz sveta filma koje se prezivaju Vajlder ili se prezivaju Hepbern.

ime	prezime
Odri	Hepbern
Bili	Vajlder
Vilijam	Vajlder

6. $(\Pi_{(naslov, žanr)} \sigma_{(godina > "1960")} FILM) \times (\Pi_{(ime)} BIOSKOP)$
Prvi član Dekartovog proizvoda daje iz relacije FILM novu relaciju samo sa atributima *naslov* i *žanr* i onim torkama kod kojih je vrednost atributa *godina* veća od 1960. Drugi član unije daje iz relacije BIOSKOP novu relaciju samo sa atributom *ime*.

I korak

naziv	žanr
Šarada	komedija
Kako ukrasti milion dolara	komedija
Ubiti pticu rugalicu	drama

II korak

Ime
Kinoteka
Dom omladine
Kulturni centar
Studentski grad

III korak

naziv	žanr	Ime
Šarada	komedija	Kinoteka
Šarada	komedija	Doma omladine
Šarada	komedija	Kulturni centar
Šarada	komedija	Studentski grad
Kako ukrasti milion dolara	komedija	Kinoteka
Kako ukrasti milion dolara	komedija	Doma omladine
Kako ukrasti milion dolara	komedija	Kulturni centar
Kako ukrasti milion dolara	komedija	Studentski grad
Ubiti pticu rugalicu	drama	Kinoteka
Ubiti pticu rugalicu	drama	Doma omladine
Ubiti pticu rugalicu	drama	Kulturni centar
Ubiti pticu rugalicu	drama	Studentski grad

7. $\Pi_{(naslov, ime, prezime)} (FILM \bowtie_{idRežisera=idOsobe} OSOBA)$

U prvom koraku se obavlja prirodno spajanje relacija FILM i OSOBA po atributima *idRežiser* i *idOsoba*. U drugom koraku se iz nove tabele biraju atributim *naslov*, *ime*, *prezime*.

I korak

idF.	naslov	žanr	idRež.	godina	idO	ime	prezime
01	Praznik u Rimu	komedija	22	1953	22	Vilijam	Vajlder
02	Šarada	komedija	20	1963	20	Stenli	Dana
03	Kako ukrasti milion dolara	komedija	22	1966	22	Vilijam	Vajlder
04	Ubiti pticu rugalicu	drama	21	1962	21	Robert	Maligan
05	Arsenik i stare čipke	komedija	19	1946	19	Frenk	Kapra
06	Sever-severozapad	triler	18	1959	18	Alfred	Hičkok
07	Ozloglašena	triler	18	1946	18	Alfred	Hičkok
08	Opčinjena	triler	18	1945	18	Alfred	Hičkok
09	Kazablanka	ljubavni	17	1942	17	Majkl	Kertiz
10	Sabrina	Ljubavni	16	1954	16	Bili	Vajlder

II korak

naslov	ime	prezime
Praznik u Rimu	Vilijam	Vajlder
Šarada	Stenli	Dana
Kako ukrasti milion dolara	Vilijam	Vajlder
Ubiti pticu rugalicu	Robert	Maligan
Arsenik i stare čipke	Frenk	Kapra
Sever-severozapad	Alfred	Hičkok
Ozloglašena	Alfred	Hičkok
Opčinjena	Alfred	Hičkok
Kazablanka	Majkl	Kertiz
Sabrina	Bili	Vajlder

2. Zadati upite koristeći operacije relacione algebre da bi se dobili odgovori na postavljena pitanja. Pretpostavimo da imamo relacionu šemu iz prethodnog zadatka.

(a) Koji su nazivi filmova čiji je žanr „ljubavni“?

$\Pi_{naslov} \sigma_{(žanr='ljubavni')} FILM$

(b) Koja su imena i prezimena režisera?

$\Pi_{(ime, prezime)} (OSOBA \bowtie_{idOsobe=idRežisera} FILM)$

(c) Koja su imena i prezimena glumaca?

$\Pi_{(ime, prezime)} (OSOBA \bowtie_{idOsobe=idGlumca} GLUMITI)$

(d) Koja su imena i prezimena glumaca koji su i režiseri?

$R_1 = \Pi_{(ime, prezime)} (OSOBA \bowtie_{idOsobe=idRežisera} FILM)$

$R_2 = \Pi_{(ime, prezime)} (OSOBA \bowtie_{idOsobe=idGlumca} GLUMITI)$

$R = R_1 \cap R_2$

(e) Koji su filmovi (identifikatori) prikazani 2007. godine?

$\Pi_{idFilma} \sigma_{(datum \geq 01-01-2007 \wedge datum \leq 31-12-2007)} PRIKAZATI$

(f) Koji su naslovi filmova prikazanih 2007. godine?

$\Pi_{naslov} (FILM \Join_{idFilma} \sigma_{(datum \geq 01-01-2007 \wedge datum \leq 31-12-2007)} PRIKAZATI)$

(g) Izlistati naslove filmova koje je režirao Alfred Hičkok.

$\Pi_{naslov} (\sigma_{(ime = \text{"Alfred"} \wedge prezime = \text{"Hičkok"})} OSOBA \Join_{idOsobe = idRežisera} FILM)$

(h) Koji režiseri su režirali i komedije i ljubavne filmove?

$R_1 = \Pi_{idRežisera} \sigma_{(žanr = \text{"ljubavni"})} FILM$

$R_2 = \Pi_{idRežisera} \sigma_{(žanr = \text{"komedija"})} FILM$

$R = OSOBA \Join_{idOsobe = idRežisera} (R_1 \cap R_2)$

(i) Koji su naslovi filmova u kojima je glumila Odri Hepbern a koji su prikazani u bioskopu Kinoteka?

Odri Hepbern glumica:

$R_1 = \sigma_{(ime = \text{"Odri"} \wedge prezime = \text{"Hepbern"})} OSOBA \Join_{idOsobe = idGlumca} GLUMITI$

Bioskop Kinoteka:

$R_2 = \sigma_{(ime = \text{"Kinoteka"})} BIOSKOP$

Prikazivanje filmova Odri Hepbern u Kinoteci:

$R_3 = R_1 \Join_{idFilma} PRIKAZIVATI \Join_{idBioskopa} R_2$

Naslovi filmova Odri Hepbern prikazanih u Kinoteci:

$R_3 = \Pi_{(naslov)} R_3$

Napomena:

Ovaj tekst se zasniva na knjizi:

Laurent Audibert, *Base de données de la modélisation au SQL*, Ellipses Édition, 2009,

ISBN 978-2-7298-5120-0

3 Modèle de données relationnel, 75-104.