

db1 - Database

Mesta : Table

Postanski broj	Naziv mesta	Broj stanovnik
11000	Beograd	2000000
18000	Nis	300000
18205	Novo selo kod Nisa	2000
35320	Novo selo kod Paracina	4000
*	0	0

Record: 1 ▶ ▶ ▶ * of 4

Muske osobe : Select Query

Maticni broj	Ime	Prezime	Pol
210696744432	Gojko	Nikolic	muski
120295460789	Dragan	Stojanovic	muski
051196138921	Dragan	Stojanovic	muski
*			

Record: 1 ▶ ▶ ▶ * of 3

Odakle su osobe : Select Query

Mesta

*	Postanski broj	Naziv mesta	Broj stanovnika
*	11000	Beograd	2000000
*	18000	Nis	300000
*	18205	Novo selo kod Nisa	2000
*	35320	Novo selo kod Paracina	4000

Osobe

Maticni broj	Ime	Prezime	Pol	Mesto
210696744432	Gojko	Nikolic	muski	
120295460789	Dragan	Stojanovic	muski	
051196138921	Dragan	Stojanovic	muski	
*				

**NORMALNE
FORME**

Field: Ime Prezime Mesto Naziv mesta
Table: Osobe Osobe Osobe Mesta
Sort:
Show:
Criteria:

Normalizacija

- Normalizacija je postupak poboljšanja šeme eliminisanjem uzroka koji dovode do anomalija u postupcima ažuriranja baze podataka.
- Procesom normalizacije se relacije, odnosno tabele, dovode u normalne forme koje obezbeđuju minimum redundantnosti i omogućuju lakše održavanje baze.

Normalizacija

- **Normalizacija** je postupak projektovanja logičke strukture baze podataka. Uobičajeno je da se koristi za projektovanje logičke strukture relacionog modela, pa će i ovde normalne forme biti definisane u terminologiji relationalnog modela.
- Međutim, postupak normalizacije ima opštiji značaj i treba ga primenjivati i na druge "klasične" modele baze podataka (mrežni i hijerarhijski), kao i za projektovanje strukture zapisa u obradi podataka zasnovanoj na skupu datoteka. Relaciona terminologija se očigledno i direktno preslikava u terminologiju ovih drugih modela.

Normalizacija

- Najopštije rečeno, dobra je ona struktura baze podataka u kojoj je logička redundansa minimalna. Problemi održavanja redundantne baze podataka i osnovne operacije održavanja su kao što smo već napomenuli:
 - dodavanje nove n-torke u relaciju,
 - izbacivanje neke n-torke iz relacije i
 - izmena vrednosti nekog atributa u relaciji (ažuriranje).

Problemi pri izvođenju ovih operacija (potreba da se pri izmeni jednog atributa, dodavanju ili izbacivanju jedne n-torke sama operacija mora ponavljati više puta, ili čak da se neko logičko dodavanje ne može izvršiti, odnosno da izbacivanje jednog logičkog skupa podataka dovodi do neželjenog izbacivanja drugih podataka) nazivaju se *anomalije u održavanju BP*.

Otklanjanje anomalija

- Jedan od načina rešavanja anomalija je *dekompozicija relacija*, pri čemu se jedna relacija zamenjuje sa više drugih relacija tako da se uklone anomalije. Mogu se prepoznati nekoliko pravila za normalizaciju:
- Polja treba da budu atomična,
- Svaki slog – red u tabeli bi trebalo da ima jedinstveni identifikator – primarni ključ
- Primarni ključ treba da bude jednostavan, stabilan i kratak
- Svako polje bi trebalo da obezbedi neku dodatnu informaciju o slogu u kome primarni ključ služi za identifikaciju
- Informacije u tabelama ne bi trebalo da se pojavljuju na više mesta, već samo ne jednom.

Otklanjanje anomalija

- Ilustrovaćemo probleme održavanja i izveštavanja na primeru jedne nenormalizovane relacije. Nenormalizovana relacija je ranije definisana kao relacija koja poseduje neke vrednosti atributa koje nisu "atomske", odnosno relacija koja poseduje "grupe sa ponavljanjem".

STUDENT

BI	IME	SEM	ŠSMER	IMERUK	ŠPRED	NAZPRED	OCENA
21	ZORAN	5	01	BATA	121	MATEMAT	7
					323	BAZEPOD	8
					056	MARKSIZ	8
77	ANA	7	01	BATA	056	MARKSIZ	10
					121	MATEMAT	5
36	PERA	4	02	MIKA	323	BAZEPOD	8
					456	ELEKTRON	9
					442	FIZIKA	6
					056	MARKSIZ	8

Otklanjanje anomalija

- **Anomalije u dodavanju:** Ako je u novom nastavnom planu definisan novi predmet, ne mogu se ubaciti podaci o tom predmetu dok ga neki student ne položi. Ili, ako se otvori neki novi smer, ne mogu se ubaciti podaci o tom smeru dok ga neki student ne upiše.
- **Anomalije u izbacivanju:** Ako je jedan predmet (FIZIKA) položio samo jedan student (PERA) i ako se on ispiše sa fakulteta, odnosno izbaci odgovarajuća n-torka, gube se i sve informacije o tom predmetu. Ako je taj student bio i jedini student na nekom smeru, gube se i sve informacije o tom smeru.
- **Anomalije u ažuriranju:** Ako se promeni naziv nekog predmeta ili rukovodioc nekog smera, to se mora učiniti na onoliko mesta koliko je studenata položilo taj predmet, odnosno koliko je studenata upisano na dati smer.

Otklanjanje anomalija

■ Problemi u izveštavanju:

- Data struktura relacije je veoma pogodna za izveštaj na zahtev: "Prikaži listu studenata, svih ispita koje je svaki student položio i njegovu prosečnu ocenu" (Uverenje o položenim ispitima). Odgovarajući program bi bio veoma jednostavan i sveo bi se na listanje date relacije (datoteke).
 - Međutim "simetričan" zahtev: "Prikaži listu predmeta, imena svih studenata koji su ga položili i prosečnu ocenu na predmetu", za datu strukturu relacije zahtevao bi znatno složeniji program, ili bi samu relaciju trebalo prestrukturirati i dobiti dve relacije sa istim skupom podataka za dva različita zahteva, čime bi se redundansa podataka, pa samim tim i problemi održavanja baze podataka, umnožili.
-
- *Postupkom normalizacije logička struktura baze podataka se dovodi u takav oblik (ili, drugim rečima, relacije se dovode u normalne forme) u kome se izbegavaju anomalije u održavanju i problemi u izveštavanju.*

Funkcionalne zavisnosti

→ za početak, izlet u svet algebre

- PRIMER 1:
- Pretpostavimo da kupujemo kutiju sa keksom i prodavac vam kaže da svaka kutija košta 40 DIN =>
- Na osnovu toga $VrednostKeksa = BrojKutija \times 4$
- Generalni način da se izrazi odnos između VrednostKeksa i BrojKutija jeste reći da VrednostKeksa zavisi od BrojKutija
- Formalno, VrednostKeksa je funkcionalno zavisna od BrojKutija
- BrojKutija → VrednostKeksa
- BrojKutija determiniše (određuje) VrednostKeksa

Funkcionalne zavisnosti

- PRIMER 2:
- Određivanje iznosa na osnovu cene pojedinačnog artikla i količine:
 - Iznos = JediničnaCena x Količina
 - Sledi: Iznos zavisi od JediničnaCena i Količina
 - Formalno, Iznos je funkcionalno zavistan od JediničnaCena i Količina
 - (JediničnaCena, Količina) → Iznos
 - JediničnaCena i Količina determinišu (određuju) Iznos

Funkcionalne zavisnosti

- PRIMER 3:
- Prepostavimo da znamo da vreća sadrži crvene, plave i žute objekte i da crveni teže 5 dkg, plavi 3 dkg i žuti 7 dkg.
- Ako neko pogleda u vreću i kaže boju objekta, vi odmah znate težinu:
- Sledi: BojaObjekta → Težina (Težina je funkcionalno zavisna od boje)
- Ako je poznato da su crveni objekti lopte, plavi kocke i žuti trouglovi, sledi BojaObjekta → Oblik, tj.
- BojaObjekta → (Težina, Oblik)
- (boja oblika determiniše težinu i oblik)

Funkcionalne zavisnosti

- Navedeno se može predstaviti i na sledeći način:

<u>BojaObjekta</u>	Težina	Oblik
Crvena	5	lopta
Plava	3	kocka
Žuta	7	trougao

- Ovo je relacija s primarnim ključem BojaObjekta
- OBJEKT(#BojaObjekta, Težina, Oblik)

Funkcionalne zavisnosti

■ Kompozitne (složene) grupe:

→ značenje grupe zavisi od toga sa koje strane se pojavljuje atribut

1. BrojNarudžbe → (ŠifraKupca, ŠifraArtikla, Količina)

■ Znači da za datu vrednost BrojNarudžbe, možemo potpuno odrediti vrednost za ŠifraKupca, ŠifraArtikla i Količina, tj. važi:

BrojNarudžbe → ŠifraKupca

BrojNarudžbe → ŠifraArtikla

BrojNarudžbe → Količina

Funkcionalne zavisnosti

- Kompozitne (složene) grupe:
 $(\text{ŠifraKupca}, \text{ŠifraArtikla}, \text{Količina}) \rightarrow \text{Cena}$
- Znači da za date vrednosti ŠifraKupca, ŠifraArtikla i Količina može se odrediti Cena, sa tim da se ovaj determinant-atribut ne može razbiti na delove, odnosno
 - $\text{ŠifraKupca} \rightarrow \text{Cena}$ **!!!! nije istinita tvrdnja**
- Determinanta je atribut od koga je neki atribut potpuno funkcijski zavistan.

Funkcionalne zavisnosti

- Razlika između Primarnog ključa-PK i Determinante:
 - Primarni ključ je UVEK determinant, dok determinant NIJE UVEK primarni ključ
- Primer:
SMESTAJ(#ŠifraStudenta,NazivSmeštaja,Stanarina)
 - Pretpostavka je da svi studenti u istom smeštaju plaćaju istu stanarinu.
 - ŠifraStudenta → (NazivSmeštaja,Stanarina) i PK i Determinanta
 - NazivSmeštaja → Stanarina (samo determinanta)

Funkcionalne zavisnosti

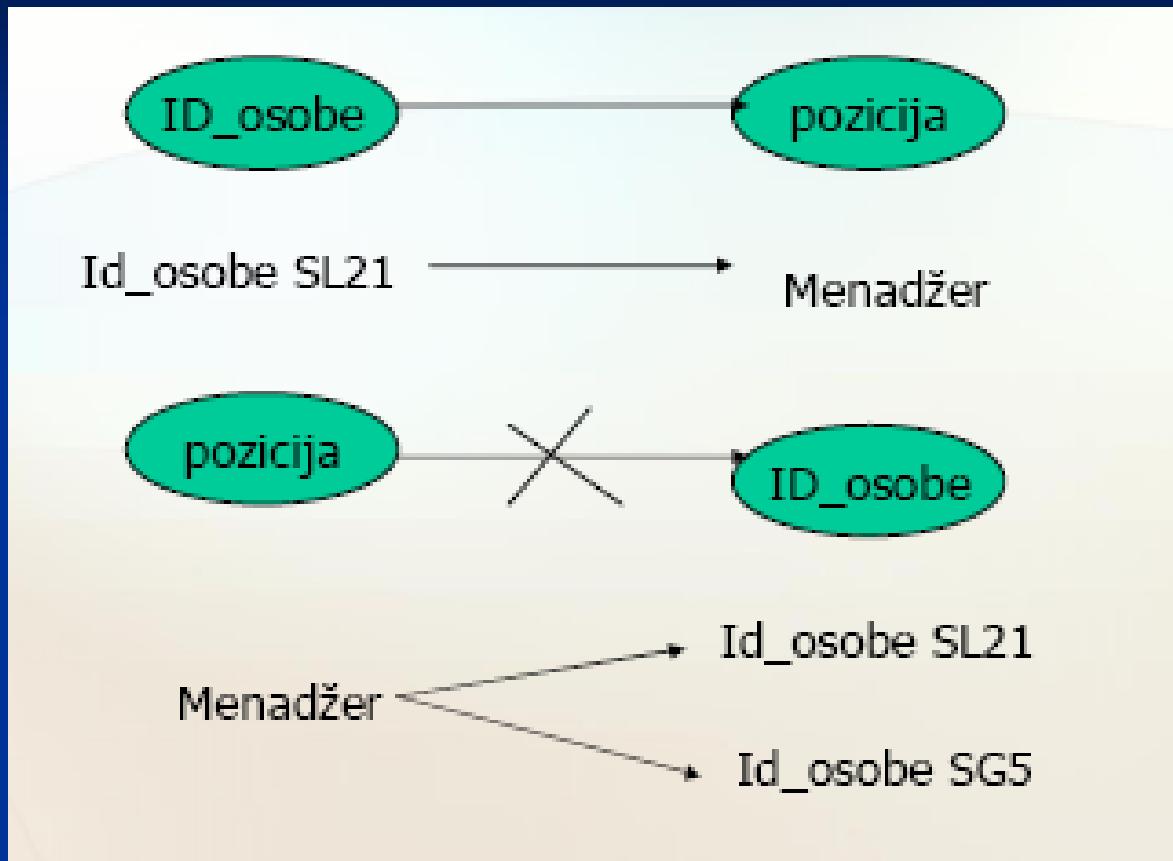
- *Funkcionalna zavisnost atributa:*
- Ako posmatramo tabelu R sa atributima X i Y koji mogu biti kompozitni tj. složeni: za atribut Y tabele R kaže se da je funkcijски zavistan od atributa X iste tabele

$$R.X \rightarrow R.Y$$

ako je svaka pojedina vrednost atributa X povezana sa samo jednom vrijednošću atributa Y.

- *Primer:*
- Za relaciju Proizvod(P#,Naziv,Boja,Težina) važi:
- $P.P\# \rightarrow P.Naziv$, ali ne i $P.Boja \rightarrow P.Težina$

Funkcionalne zavisnosti



Ako u relaciji R vrednost atributa A jednoznačno određuje vrednost atributa B, onda je atribut B funkcionalno zavisан od atributa A ali ne važи obrnuto pravilo.

DEFINICIJA funkcionalne zavisnosti

- *Potpuna funkcionalna zavisnost atributa:*
- *U tabeli R sa atributima X i Y koji mogu biti kompozitni tj. složeni, Y je potpuno funkcionalno zavistan od X ako vredi da je Y funkcionalno zavistan od X i nije funkcionalno zavistan ni od jednog manjeg podskupa atributa X.*
- *Odnosno, ako vredi $X \rightarrow Y$ tada ne sme postojati ni jedan podskup Z koji sadrži samo deo atributa od kojih se sastoji atribut X, za koji bi vredilo da je $Z \rightarrow Y$.*
- Primer:
- DP(PB#,P#,Količina,Grad), sa sledećim zavisnostima:
 - $DP.PB\# \rightarrow DP.Grad$, gde je Grad potpuno funkcionalno zavistan od PB# jer je za datu vrednost atributa PB# vrednost atributa Grad uvek ista.
 - $DP.(PB\#,P\#) \rightarrow DP.Grad$, gde ne postoji potpuna funkcionalna zavisnost jer je Grad potpuno funkcionalno zavistan samo od PB# koji je podskup složenog atributa (PB#,P#)

Funkcionalne zavisnosti

- *Tranzitivna funkcionalna zavisnost atributa:*
- *Ako vredi $X \rightarrow Y$ i $Y-/ \rightarrow X$ (Y je funkcionalno zavistan od X , a X nije funkcionalno zavistan od Y), i ako $Y \rightarrow A$ (A je funkcionalno zavistan od Y) tada vredi da je A funkcionalno zavistan i od X ($X \rightarrow A$). Ako vredi $A-/ \rightarrow Y$ tada je A potpuno tranzitivno zavistan od X .*

Funkcionalne zavisnosti

Funkcionalne zavisnosti (FZ) su:

- Veze između dva skupa atributa u relaciji
- Postoje kada neki atribut **A** jednoznačno opisuje neki atribut **B**
- Kada je **B** funkcionalno određen sa **A** svaki red podataka sa jednim skupom vrednosti u **A** uvek će imati isti skup vrednosti u **B**
- Funkcionalna zavisnost se tada piše $A \rightarrow B$

Funkcionalne zavisnosti

- Svaki (nehaotični) problemski domen kao deo realnog sveta ili neke apstrakcije opisan je skupom (više ili manje) očiglednih funkcionalnih zavisnosti **F**
- Skup, korišćenjem određenih pravila izvedenih FZ, zajedno sa izvornim skupom FZ čini prošireni skup funkcionalnih zavisnosti **F+**
- Nove FZ izvodimo uz pomoć **Armstrongovih aksioma i pravila**, izvedenih iz tih aksioma

Funkcionalne zavisnosti

Neka su X , Y i Z skupovi atributa. Njihovo međusobno delovanje opisujemo sa **Armstrongovim aksiomima**:

- Refleksivnost (reflexivity)

Ako je $X \supseteq Y$ onda: $X \rightarrow Y$ (trivijalno pravilo)

- Proširenje (augmentation)

Ako vredi $X \rightarrow Y$, onda vredi takođe: $XZ \rightarrow YZ$

- Tranzitivnost (transitivity)

Ako vredi $X \rightarrow Y$ i $Y \rightarrow Z$, onda je: $X \rightarrow Z$

Funkcionalne zavisnosti

Armstrongovi aksiomi mogu biti prošireni su dodatnim pravilima, izvedenim iz aksioma:

- Unija (union)

Ako vredi $X \rightarrow Y$ i $X \rightarrow Z$, onda je: $X \rightarrow YZ$

- Dekompozicija (decomposition)

Ako vredi $X \rightarrow YZ$, onda je: $X \rightarrow Y$ i $X \rightarrow Z$

- Pseudo tranzitivnost (pseudotransitivity)

Ako vredi $X \rightarrow Y$ i $YW \rightarrow Z$, onda: $XW \rightarrow Z$

Funkcionalne zavisnosti

■ Primer:

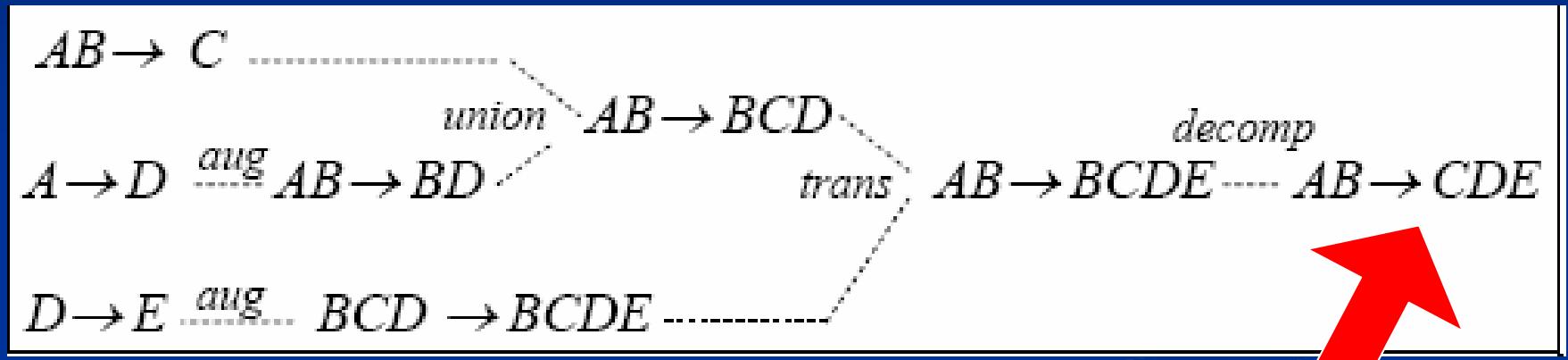
za relaciju $R(ABCDE)$ zadan je skup pravila:

$$F = \{AB \rightarrow C, A \rightarrow D, D \rightarrow E\}$$

koristeći Armstrongove aksiome i pravila izvedena
iz njih potrebno je dokazati da vredi

$$AB \rightarrow CDE$$

Funkcionalne zavisnosti



ZADATAK: Za relaciju R(ABCDEF) zadat je skup pravila:

$$F = \{AB \rightarrow C, A \rightarrow D, D \rightarrow E\}.$$

Koristeći Armstrongove aksiome i pravila koja iz njih proizlaze (osim pseudo tranzitivnosti) potrebno je dokazati da vredi

$$AB \rightarrow CDE.$$

Normalne forme

- Prilikom projektovanja baze podataka je potrebno odrediti:
 - Koje tabele (relacije) će baza imati
 - Koji su atributi svake tabele pojedinačno
 - Kakve su veze između tabela.
- Ponavljanje istog podatka u jednoj bazi podataka se naziva **redundantnost** podataka.
- Tokom korišćenja baze podataka nad njom se vrše osnovne operacije ažuriranja:
 - dodavanje novog zapisa,
 - brisanje postojećeg zapisa, i
 - izmena vrednosti jednog ili više atributa u postojećem zapisu.

Normalne forme

- Normalne forme su rezultat primene prethodnih pravila.
- Razlikuju se pet normalnih formi (NF)
- 1NF, 2NF, 3NF, 4NF i 5NF
- Svaka sledeća normalna forma u redosledu omogućava sve bolju šemu baze podataka od prethodne i manju redundantnost.

Normalne forme

- Kao sto smo napomenuli ranije, teorija normalizacije se ne zaustavlja na trećem pravilu normalizacije. Po pitanju primene teorije, vodi se debata izmedju « teorijskih čistunaca » i profesionalaca gde prvi smatraju da sve tabele moraju biti barem u trećoj normalnoj formi, dok profesionalci zagovaraju *denormalizaciju* u cilju poboljšanja performansi baze.
- Pod *denormalizacijom* oni podrazumevaju redukovanje stepena normalizacije iz treće u drugu normalnu formu.

Normalne forme

- Teorija normalizacije nije ništa drugo nego formalizacija nekih intuitivno prihvatljivih principa o "zdravom oblikovanju šeme baze. Ukoliko već na počeku uočimo sve potrebne entitete, attribute i veze, tada ovakav postupak prirodno produkuje šemu koja zadovoljava sve potrebne norme.
- Sa druge strane ako je polazna šema loše dizajnirana, tada će postupak normalizacije ispraviti sve greške nastale u njoj.

Alati za izradu dijagrama Objekat-Relacija:

- Da bi olakšali posao izrade logičkog modela baze podataka možete u tu svrhu koristiti jedan od postojećih CASE alata za modeliranje :
 - Designer/2000,
 - Oracle Data Designer,
 - Sybase S-Designer ...
 - ERWin
 - BPWin
 - Java Report Designer
- Svi ovi alati su raspoloživi u vezijama Windows 9x ili Windows XT... Ovi alati omogućuju da na laki način, u maniru alata za crtanje, dodajemo objekte u model, stvaramo relacije izmedju njih, menjamo osobine objekata i relacija itd.

Normalizacija baze podataka -preporuke-

Dobro

- Ucinite sve kako bi vaša šema bila u trećoj normalnoj formi (3NF)

Pogrešno

- Predpostaviti da su performance loše bez obavljanja kompletnih i preciznih ispitivanja.
- Izvršite testove performansi na uzorku koji odgovara realnom stanju baze u eksploataciji. Tek posle ovakvih testova možete razmatrati mogućnost denormalizacije strukture baze podataka.

Normalizacija baze podataka -preporuke-

Dobro

- Nastojte rešavati probleme performansi dajući prednost materijalnim poboljšanjima nad kompromisima u koncepciji baze podataka. Planirajući na duži vremenski period, materijal višeg kvaliteta je jeftiniji od održavanja denormalizovane baze podataka.

Pogrešno

- Denormalizovati tabele bez potpunog razumevanja nedostataka primjenjenog tehničkog kompromisa.
- **Primer :** poboljšanje performansi uz povećanje broja nepotrebnih podataka, kompleksnu logiku ažuriranja i dodatne poteškoće u evoluiranju baze tokom njenog životnog ciklusa.

Prva normalna forma (1NF)

- *Definicija: Šema relacije R je u prvoj normalnoj formi (1NF) ako i samo ako domeni relacije R sadrže samo proste ("atomic") vrednosti, i neključni atributi funkcijски зависе од ključа*
- Proste vrednosti su vrednosti koje se dalje ne mogu deliti ili ako u konkretnoj situaciji nisu rastavljene na komponente;
- U 1NF nisu dozvoljeni viševrednosni i kompozitni atributi i njihove kombinacije, tj. nisu dopuštene relacije u relacijama ili relacije kao atributi n -torki.
- 1NF normalizacija
- Ako šema relacije $R(A_1, A_2, \dots, A_n)$ nije u 1NF, onda postoji takva dekompozicija šeme relacije R u skup šema relacija (R_1, \dots, R_k) koje su u 1NF, da se operacijom prirodnog spajanja iz ovih šema relacija može ponovo dobiti šema relacije R.

Prva normalna forma (1NF)

- Za dekompoziciju se koriste sledeća pravila:
- 1. Ako šema relacije $R(A_1, A_2, \dots, A_k, \dots, A_n)$ ima viševrednosni atribut A_i i primarni ključ A_k , tada se šema relacije R razlaže u dve relacije $R_1(A_1, A_2, \dots, A_k, \dots, A_n)$ i $R_2(A_k, A_i)$. Treba uočiti da je R_1 formirana iz R isključivanjem viševrednosnog atributa, a da je R_2 formirana od primarnog ključa A_k i viševrednosnog atributa A_i ;
- 2. Ako relacija $R(A_1, A_2, \dots, A_k, \dots, A_n, Q(B_1, B_2, \dots, B_k, \dots, B_m))$ ima ugnjezdenu relaciju Q , tada se R razlaže u dve relacije $R_1(A_1, A_2, \dots, A_k, \dots, A_n)$ i $Q_1(A_k, B_1, B_2, \dots, B_k, \dots, B_m)$. U slučaju da relacija ima više ugnježdenja, treba ovaj postupak ponoviti rekurzivno.

Prva normalna forma (1NF)

- Primer:
- Šema relacije SEKTOR<SNAZIV, SBR. SRUK, (SLOK)> nije u 1NF, jer je atribut SLOK više vrednosni. Posmatraćemo jednu pojavu ove relacije:
- SEKTOR

SNAZIV	<u>SBR</u>	SRUK	SLOK
N1	S1	R1	(L1,L2,L3)
N2	S2	R2	L1
N3	S3	R3	L1

Prva normalna forma (1NF)

- Da bi se ova relacija prevela u 1NF moguća su dva rešenja.

Prvo rešenje je da se ova relacija razloži u dve relacije

SEKTOR<SNAZIV, SBR, SRUK> i

SEKLOK<SBR, SLOK>.

Za gore navedeni primer, pojave ovih relacija bi imale sledeći oblik:

SNAZIV	SBR	SRUK
N1	S1	R1
N2	S2	R2
N3	S3	R3

SBR	SLOK
S1	L1
S1	L2
S1	L3
S2	L1
S3	L1

Prva normalna forma (1NF)

■ Primer 1:

- Šema relacije RADPROJ (MBR, RIME, (PBR, SATI)) sadrži ugnježdenu relaciju PROJEKTI (PBR, SATI). Atribut PBR je parcijalni primarni ključ ugnježdene relacije, dok je MBR primarni ključ relacije RADPROJ. Posmatraćemo jednu pojavu ove relacije
- RADPROJ

<u>MBR</u>	RIME	PBR	SATI
M1	11	P1	S1
M1	11	P3	S2
M1	11	P5	S3
M2	12	P3	S4
M2	12	P4	S5
M3	13	P1	S6

Prva normalna forma (1NF)

- Da bi se relacija RADPROJ prevela u 1NF, potrebno je da se ugnježdena relacija formira kao nezavisna relacija. Šeme relacije imaju sada sledeći oblik: RAD(MBR, RIME) i PROJ(MBR, PBR, SATI). Za gore zadani primer izgled relacije bi bio:

RAD

MBR	RIME
M1	I1
M2	I2
M3	I3

PROJ

<u>MBR</u>	PBR	SATI
M1	P1	S1
M1	P3	S2
M1	P5	S3
M2	P3	S4
M2	P4	S5
M3	P1	S6

Prva normalna forma (1NF)

- Primer 2:
- Ilustrovaćemo probleme održavanja i izveštavanja na primeru jedne nenormalizovane relacije.
- Nenormalizovana relacija je ranije definisana kao relacija koja poseduje neke vrednosti atributa koje nisu "atomske", odnosno relacija koja poseduje "grupe sa ponavljanjem" i ugnježdene relacije.
- Posmatrajmo sledeću šemu baze:

STUDENT < BI, IME,SEM,ŠSMER,IMERUK,ŠPRED,NAZPRED,OCENA >

Prva normalna forma (1NF)

- Relaciju STUDENT:

STUDENT < BI, IME, SEM, ŠSMER, IMERUK, ŠPRED, NAZPRED, OCENA >

- možemo normalizovati (svesti na 1NF: svi atributi atomarni, domeni atributa u nove relacije, ugnježdene relacije u nove relacije, svi atributi potpuno zavisni od svog ključa u nove relacije) ako u svakoj vrsti prikazane tabele, za svaki ispit koji je neki student položio, ponovimo i sve ostale njegove podatke (BI, IME, SEM, ŠSMER, IMERUK). Relacija bi se širila "na dole".

BI	IME	SEM	ŠSMER	IMERUK	ŠPRED	NAZPRED	OCENA
10	Ana	5	01	Miloš	111	Menadžment	8
					222	Informatika	9
					333	Baze podataka	10
20	Pera	7	01	Miloš	222	Informatika	10
					333	Baze podataka	9
30	Mika	4	02	Maja	111	Menadžment	8
					444	Matematika	6

Prva normalna forma (1NF)

- Međutim, bilo bi dosta ponavljanja podataka, pa da bi se smanjila redundansa do koje bi, očigledno, ovakva normalizacija dovela, možemo ovako dobijenu relaciju, normalizacijom u Prvu normalnu formu:
 - konstatovanjem da su svi atributi atomarni
 - konstatovanjem da nema domena atributa
 - konstatovanjem da POSTOJI ugnježdena relacija sa ključem **ŠPRED**

STUDENT < BI, IME, SEM, ŠSMER, IMERUK, ŠPRED, NAZPRED, OCENA >

- dekomponovati na sledeće dve:
- STUDENT1 (BI, IME, SEM, ŠSMER, IMERUK) (1)
- PRIJAVA (ŠPRED, BI, NAZPRED, OCENA) (2)
- Ključ nove relacije PRIJAVA ostaje ŠPRED ali radi povezivanja sa predhodnom po pravilu mora imati i ključ BI. Prirodno spajanje ove dve relacije po BI rekonstruisalo bi polaznu normalizovanu relaciju.

Prva normalna forma (1NF)

BI	IME	SEM	ŠSMER	IMERUK
10	Ana	5	01	Miloš
20	Pera	7	01	Miloš
30	Mika	4	02	Maja

ŠPRED	NAZPRED	OCENA
111	Menadžment	8
222	Informatika	9
333	Baze podataka	10
222	Informatika	10
333	Baze podataka	9
111	Menadžment	8
444	Matematika	6

Po pravilu primarni ključ prve tabele postaje sekundarni u drugoj.

Prva normalna forma (1NF)

□ 1 NF

BI	IME	SEM	ŠSMER	IMERUK
10	Ana	5	01	Miloš
20	Pera	7	01	Miloš
30	Mika	4	02	Maja

Tabela ISPITI

BI	ŠPRED	NAZPRED	OCENA
10	111	Menadžment	8
10	222	Informatika	9
10	333	Baze podataka	10
20	222	Informatika	10
20	333	Baze podataka	9
30	111	Menadžment	8
30	444	Matematika	6

Prva normalna forma (1NF)

■ Primer 3: Na primer, posmatramo tabelu sa narudžbinama u nekoj prodavnici:

Br. Narudž.	Datum	Šifra kupca	Naziv kupca	Šifra robe	Naziv robe	Naruč. kol.	Isporuč. kol.	Cena
123	18/10/13.	567	Pera Petrović	111	Grfička	1	1	6000
123	18/10/13.	567	Pera Petrović	222	Procesor	1	1	8000
123	18/10/13.	567	Pera Petrović	333	Matična	1	0	6000
123	18/10/13.	567	Pera Petrović	444	HDD	3	3	5000
56	16/10/13.	432	Mika Mikić	555	Tastatura	10	8	700
56	16/10/13.	432	Mika Mikić	444	HDD	10	5	5000

Prva normalna forma (1NF)

- Ono što se ponavlja u više redova tabele, a ima isti sadržaj, potrebno je izdvojiti u posebnu tabelu.

Br. Narudž.	Datum	Šifra kupca	Naziv kupca	Šifra robe	Naziv robe	Naruč. kol.	Isporuč. kol.	Cena
123	18/10/13.	567	Pera Petrović	111	Grfička	1	1	6000
123	18/10/13.	567	Pera Petrović	222	Procesor	1	1	8000
123	18/10/13.	567	Pera Petrović	333	Matična	1	0	6000
123	18/10/13.	567	Pera Petrović	444	HDD	3	3	5000
56	16/10/13.	432	Mika Mikić	555	Tastatura	10	8	700
56	16/10/13.	432	Mika Mikić	444	HDD	10	5	5000

Prva normalna forma (1NF)

- Dobijamo tako dve tabele, kojima moramo dodeliti primarne ključeve da bismo ih na neki način povezali:

Br. Narudž.	Datum	Šifra kupca	Naziv kupca
123	18/10/13.	567	Pera Petrović
56	16/10/13.	432	Mika Mikić

Br. Narudž.	Šifra robe	Naziv robe	Naruč. kol.	Isporuč. kol.	Cena
123	111	Grfička	1	1	6000
123	222	Procesor	1	1	8000
123	333	Matična	1	0	6000
123	444	HDD	3	3	5000
56	555	Tastatura	10	8	700
56	444	HDD	10	5	5000

Prva normalna forma (1NF)

■ Primer 4: NORMALIZACIJA DELA IS NEKOG FORUMA STRUČNJAKA

FORUM			
ID	KORISNIK	ZIVI	ZNANJE
1	Pera Peric	Srbija	Audio
2	Mika Mikic	Srbija	Video
3	Zikica Zikic	Srbija	Audio, Video, Hardware, Windows
4	Mali Perica	Srbija	Audio, Video, Baze, Hardware, Windows, Linux, Programiranje
5	Mali Zikica	BIH	Programiranje, Hardware

Prva normalna forma (1NF)

KORISNIK

KORISNIK_ID	KORISNIK_IME	KORISNIK_ZIVI
1	Pera Peric	Srbija
2	Mika Mikic	Srbija
3	Zikica Zikic	Srbija
4	Mali Perica	Srbija
5	Mali Zikica	BIH

ZNANJE

ZNANJE_ID	KORISNIK_ID	ZNANJE_IME
1	1	Audio
2	2	Video
1	3	Audio
2	3	Video
3	3	Hardware
4	3	Windows
1	4	Audio
2	4	Video
5	4	Baze
3	4	Hardware
4	4	Windows
6	4	Linux
7	4	Programiranje
3	5	Hardware
7	5	Programiranje

Prva normalna forma (1NF)

- Primer 5: NORMALIZACIJA DELA IS BIBLIOTEKE:
- U biblioteci se evidentiraju pozajmice knjiga. Relaciona shema POZ sastoji se od sledećih atributa:

- SIFCLN - šifra člana
- PREZCLN - prezime člana
- IMECLN - ime člana
- DATPOZ - datum pozajmice
- DATVR – datum vraćanja
- PBR - poštanski broj mesta stanovanja člana
- NAZM - naziv mesta stanovanja člana
- ADRCLN - adresa člana
- SIFKNJ - šifra knjige
- NAZKNJ - naziv knjige
- INVBRPRIM – inventarski broj primerka knjige
- SIFIZD - šifra izdavača
- NAZIZD - naziv izdavača

*Prepostavljeni ključ je: K POZ = { SIFCLN }
Normalizujte relaciju POZ na 1NF, ako vrede pravila:*

- Jedan član istoga dana može iznajmiti više knjiga*
- Jedan član istu knjigu može iznajmiti više puta*
- Jedna knjiga ima jednog izdavača*

Prva normalna forma (1NF)

- Šema relacije je:
- $\text{POZ} = \langle \underline{\text{SIFCLN}}, \text{PREZCLN}, \text{IMECLN}, \text{DATPOZ}, \text{DATVR}, \text{PBR}, \underline{\text{NAZM}}, \underline{\text{ADRCLN}}, \text{SIFKNJ}, \text{NAZKNJ}, \text{INVBRPRIM}, \text{SIFIZ}, \text{NAZIZD} \rangle$
- Pretpostavljeni ključ je:
 $K_{POZ} = \{ \underline{\text{SIFCLN}} \}$

Prva normalna forma (1NF)

- NORMALIZACIJA U PRVU NORMALNU FORMU 1NF
- CLAN1 = <SIFCLN, PREZCLN, IMECLN, PBR, NAZM, ADRCLN >
 - K CLAN1 = { SIFCLN }
- POZ1=<SIFCLN, INVBRPRIM, DATPOZ, DATVR, SIFKNJ, NAZKNJ, SIFIZ, NAZIZD >
 - K POZ1 = { SIFCLN, INVBRPRIM, DATPOZ }

Prva normalna forma (1NF)

■ Primer 6: NORMALIZACIJA DELA IS VIDEO KLUB - VIKLU:

Zadana je relaciona shema FILM sa sledećim atributima:

- SifFilm - šifra filma
- NasFilm - naslov filma
- OznZanr - oznaka žanra
- OpiZanr - opis (naziv) žanra
- SifProizv - šifra proizvođača filma
- NazProizv - naziv proizvođača filma
- SifDistrib - šifra distributera filma
- NazDistrib - naziv distributera filma
- SifOso - šifra osobe
- PrezOso - prezime osobe
- ImeOso - ime osobe
- KratFun - kratica funkcije koju osoba ima u filmu
- NazFun - naziv funkcije koju osoba ima u filmu (glumac, režiser,...)

Prva normalna forma (1NF)

- Šema ovakvog dela IS bila bi:
- FILM < **SifFilm**, **NasFilm**, **OznZanr**, **OpiZanr**, **SifProizv**,
NazProizv, **SifDistrib**, **NazDistrib**, **SifOso**, **PrezOso**,
ImeOso, **KratFun**, **NazFun** >
- 1NF
- SifFilm → NasFilm, OznZanr, OpiZanr, SifProizv, NazProizv,
SifDistrib, NazDistrib
- SifFilm NE ODREĐUJE: SifOso, PrezOso, ImeOso, KratFun,
NazFun pa dobijamo dve relacije
- **F1**= SifFilm, NasFilm, OznZanr, OpiZanr, SifProizv, NazProizv,
SifDistrib, NazDistrib
- **F2**= SifFilm, SifOso, KratFun, PrezOso, ImeOso, NazFun

Prva normalna forma (1NF)

Primer 7: Izvršiti normalizaciju relacije Iznajmljivanja 1NF
Iznajmljivanja

KlijentID	Klme	StanID	Adresa	Cena	VlasnikID	Vlme
101	Pera Perić	804	Dušanova 2,Niš	100	40	Laza Lazić
		560	Višegradska 32,Niš	60	93	Jovan Jovanović
124	Mika Mikić	45	Petra Petrovića bb, Beograd	250	40	Laza Lazić
		206	Niška 22, Leskovac	110	93	Jovan Jovanović
		301	Bulevar Oslobođenja 111	150	93	Jovan Jovanović

1NF

KlijentID	Klme	StanID	Adresa	Cena	VlasnikID	Vlme
101	Pera Perić	804	Dušanova 2,Niš	100	40	Laza Lazić
101	Pera Perić	560	Višegradska 32,Niš	60	93	Jovan Jovanović
124	Mika Mikić	45	Petra Petrovića bb, Beograd	250	40	Laza Lazić
124	Mika Mikić	206	Niška 22, Leskovac	110	93	Jovan Jovanović
124	Mika Mikić	301	Bulevar Oslobođenja 111, Leskovac	150	93	Jovan Jovanović

Druga normalna forma

- Definicija druge normalne forme je: Šema relacije R je u drugoj normalnoj formi (2NF) ako i samo ako je u 1NF i ako svaki njen sporedni atribut A potpuno funkcionalno zavisi od primarnog ključa relacije R.
- (2) Šema relacije R je u drugoj normalnoj formi (2NF) ako u 1NF i ako svaki njen sporedni atribut A potpuno funkcionalno zavisi od svakog ključa relacije R.

Druga normalna forma

2NF normalizacija

- Ako šema relacije $R(A_1, A_2, \dots, A_n)$ nije u 2NF, onda postoji takva dekompozicija šeme relacije R u skup šema relacija koje su u 2NF, a operacijom prirodnog spoja se iz ovih može ponovo dobiti šema relacije R .
- Za 2NF normalizaciju se koristi sledeći postupak:
 - 1) U šemi relacije R naći sve potpune funkcionalne zavisnosti sporednih atributa od pojedinih ključnih atributa ili od skupa ključnih atributa.
 - 2) Za svaku potpunu funkcionalnu zavisnost iz predhodnog koraka formirati po jednu šemu relacije. Ključ ovih relacija je odgovarajući ključni atribut ili odgovarajući skup ključnih atributa.

Druga normalna forma

Primer 1: U šemi relacije RADPROJ(MBR, PBR, SATI, RIME, PNAZIV, PLOKACIJA) određujemo da li postoje funkcionalne zavisnosti ostalih atributa od ključnih i sporednih atributa:

- F1:(MBR, PBR) -> SATI
- F2: (MBR, PBR) -> RIME
- F3: (MBR, PBR) -> (PNAZIV, PLOKACIJA)

- F1 je potpuna funkcionalna zavisnost, jer atribut SATI zavisi od složenog ključa a ni MBR -> SATI ni PBR -> SATI ne zadržavaju potpunu zavisnost F1.
- F2 je parcijalna funkcionalna zavisnost jer atribut RIME potpuno zavisi samo od ključa MBR -> RIME.
- F3 je takođe parcijalna funkcionalna zavisnost, jer atributi PNAZIV i PLOKACIJA potpuno zavise samo od ključa PBR -> (PNAZIV, PLOKACIJA). Stoga u relaciji RADPROJ postoje samo tri potpune funkcionalne zavisnosti sporednih ključnih atributa:

- f1: (MBR, PBR) ->SATI
- f2: MBR -> RIME
- f3: PBR -> (PNAZIV, PLOKACIJA)

Druga normalna forma

- Relacija RADPROJ nije u 2NF jer sporedni atribut RIME nije potpuno funkcionalno zavistan od ključa (MBR, PBR) zbog funkcionalne zavisnosti f2. Isto važi za sporedne atribute PNAZIV i PLOKACIJA zbog f3. Stoga se ova relacija razlaže na tri nove relacije:

R1(MBR, PBR, SATI)

R2(MBR, RIME)

R3(PBR, PNAZIV, PLOKACIJA)

Druga normalna forma

- Primer 2: Polazimo naravno od prve normalne forme na jednostavnom primeru popisa radnika u poduzeću i njihovog međusobnog odnosa nadređenosti.
Pretpostavimo da u preduzeću važe sledeći odnosi:

Nadređeni	Podređeni 1	Podređeni 2	Podređeni 3	Podređeni 4
Branko	Jovan	Marija	Stefan	
Marija	Mihajlo	Stanko	Ksenija	Darko
Jovan	Aleksa			

Ako bi u bazi podataka kreirali tabelu tačno prema prethodnom primeru, ubrzo bi se pojavili ogromni problemi. Podaci o radniku Jovanu bili bi upisani u tabelu prilično neracionalno, jer tri reda ostaju prazna. Istovremeno, za Mariju jednostavno nije moguće dodati novog podređenog radnika, jer ne postoji peti red za upis podređenih radnika.

Druga normalna forma

- Ideja koja bi početniku u radu s bazama podataka najverovatnije pala na pamet, bila bi tabela nalik sledećoj tabeli:

Nadređeni	Podređeni
Branko	Jovan, Marija, Stefan
Marija	Mihajlo, Stanko, Ksenija, Darko
Jovan	Aleksa

Nadređeni	Podređeni
Branko	Jovan
Branko	Marija
Branko	Stefan
Marija	Mihajlo
Marija	Stanko
Marija	Ksenija
Marija	Darko
Jovan	Aleksa

- Ovakva tabela je znatno bolje rešenje od prve tabele, jer u tabeli ne postoje nepotrebni redovi, a broj podređenih radnika nije ograničen (osim veličinom polja). Kod ovakve organizacije tabele, međutim, javljaju se novi problemi. Pristup do pojedinog podređenog radnika, kao i njegovo ažuriranje, znatno je složenije nego pre.

- Rešenje koje delimično zadovoljava uslove prve normalne forme prikazano je u tabeli iznad. Budući da za svaku vezu **Nadređeni – Podređeni** postoji poseban red u tabeli, nema problema kod dodavanja novih podređenih, a istovremeno se nepotrebno ne troši prostor u bazi podataka.

Druga normalna forma

- Ipak, preostao je još jedan problem zbog čega tabela još uvek nije u prvoj normalnoj formi – svaki red tabele nije jedinstveno određen ključem. Na primer, ako se u preduzeću pojavi još jedan radnik Branko, kako ćemo znati kojem od dva Branka pripada podređeni radnik.
- Završno rešenje koje garantuje da tabela bude u prvoj normalnoj formi prikazano je u nastavku:

Nadređeni	Podređeni
1	3
1	2
1	4
2	5
2	6
2	7
2	8
3	9

Umesto da u redove Nadređeni i Podređeni upisujemo imena radnika, upisaćemo njihove jedinstvene šifre. Kod dolaska novog radnika u preduzeće dodeljuje mu se nova, jedinstvena šifra, tako da ne može doći do zabune. Sada se tabela zaista nalazi u prvoj normalnoj formi, jer zadovoljava oba postavljena uslova.

Treća normalna forma

- (1) Šema relacije R je u trećoj normalnoj formi (3NF) ako i samo ako je u 2NF i ako kod nje ne postoji tranzitivna zavisnost bilo kog sporednog atributa od primarnog ključa.
- (2) Šema relacije R je u trećoj normalnoj formi (3NF) ako i samo ako je u 2NF i ako kod nje ne postoji tranzitivna zavisnost bilo kog sporednog atributa od bilo kog ključa te relacije.
- (3) Neka je X bilo koji skup atributa šeme relacije R(A_1, A_2, \dots, A_n). Šema relacije R je u trećoj normalnoj formi (3NF) ako uvek kada važi funkcionalna zavisnost $X \rightarrow A_i$ takođe važi da je ili
 - (a) X super ključ relacije R ili da je
 - (b) A_i primarni atribut relacije R.
- Ako je šema relacije R u 2NF, definicije (1) i (2) se mogu koristiti za proveru da li je šema relacije R u 3NF.
- Definicija (3) se može direktno koristiti za proveru da li je šema relacije u 3NF.

Treća normalna forma

3NF normalizacija

- Ako šema relacije R nije u 3NF onda postoji takva dekompozicija relacije R u skup šeme relacija koje su sve u 3NF, a operacijom prirodnog spoja se iz njih može ponovo dobiti R.
- Ako je neka šema relacije R u 2NF, a nije u 3NF tada se za 3NF normalizaciju korist sledeći postupak baziran na definiciji (2):
 - 1) U šemi relacije R naći sve tranzitivne funkcionalne zavisnosti sporednih atributa relacije od ključa relacije.
 - 2) Ako u $R(A)$ važi $X \rightarrow Z$, $z \rightarrow y$, tada se $R(A)$ razlaže na dve relacije $R1(X,Z,W)$ i $R2(Z,Y)$, gde je $W = A - (X \cup Y \cup Z)$.

Treća normalna forma

- Primer:
- U šemi relacije RADSEK<MBR,RIME, DATRODJ, ADRESA, SBR, SNAZIV, SRUK> postoje sledeće funkcionalne zavisnosti:
 - f1: MBR -> RIME f2: MBR -> DATRODJ
 - f3: MBR -> ADRESA f4: MBR -> SBR
 - f5: SBR -> SNAZIV f6: SBR -> SRUK
- Koristeći pravilo tranzitivnosti mogu se izvesti sledeće zavisnosti:
 - f7: MBR -> SNAZIV f8: MBR -> SRUK
- Relacija RADSEK je u 2NF, jer svi sporedni atributi potpuno funkcionalno zavise od primarnog ključa MBR. Funkcionalne zavisnosti f4, f5, f6 su tranzitivne preko SBR, a SBR nije podskup ključa relacije RADSEK pa stoga relacija RADSEK nije u 3NF. Relacija RADSEK se razlaže na dve relacije koje su u 3NF:
 - RADSEK1<MBR, RIME, DATRODJ, ADRESA, SBR>
 - RADSEK2<SBR, SNAZIV, SRUK>

NORMALNE FORME-primer

- **Primeri normalizacije**
- Pokažimo proces normalizacije na konkretnom primera iz prakse.
- Primer 1 : Izvor podataka je formular pomoću kojega nabavljač nekog hotela vrši narudžbe robe potrebne za hotelsko poslovanje. Objekat NARUDŽBA, koji treba da se modelira za potrebe obrade podataka, ima kao što se to vidi iz priloženog primerka formulara narudžbe, veći broj atributa.
- **Prva normalna forma** relacije NARUDŽBA treba da obezbedi da se u svakom polju relacije nalazi samo jedan podatak, to jest da svaka n-torka relacije bude sastavljena samo od atomarnih vrednosti. Takva relacija (kreirana na osnovu prezentirane hotelske narudžbe) može da ima sledeći oblik:
 - $NARUDŽBA < brnar\#, datnar, šifkp, nazkp, datisp, prim, šifpr, nazpr, kol, jm, cenpr, vrpr, pop, uknar, ukpop, ukvr >$

NORMALNE FORME

Broj narudzbe	231543/93			Datum	12.03.1993			
Sifra kupca	K-008765							
Naziv i adresa kupca	Hotel "Metropol" Bulevar mira 23 11000 Beograd							
Datum isporuke	17.03.1993							
Primjedba:	Veza telefonski razgovor sa gospodinom Petrovicem							
Proizvod	Naziv	Kol	J.M.	Cijena	Vrijednost	Popust		
P-122	Prasak	300	kg	17.0	5100.00	400.00		
S-001	Sapun	100	kom	3.0	300.00	30.00		
					5400.00	430.00		
Ukupno				4970.00				

Slika 5.15 Formular za narudžbu

NORMALNE FORME

- pri čemu su uvedene skraćenice za atributе relacije NARUDŽBA:

■ <i>brnar</i> -	<i>broj narudžbe,</i>
■ <i>datnar</i> -	<i>datum narudžbe,</i>
■ <i>šifkp</i> -	<i>šifra kupca,</i>
■ <i>nazkp</i> -	<i>naziv i adresa kupca,</i>
■ <i>datisp</i> -	<i>datum isporuke,</i>
■ <i>prim</i> -	<i>primedba,</i>
■ <i>šifpr</i> -	<i>šifra proizvoda,</i>
■ <i>nazpr</i> -	<i>naziv proizvoda,</i>
■ <i>kol</i> -	<i>količina,</i>
■ <i>jm</i> -	<i>jedinica mere,</i>
■ <i>cenpr</i> -	<i>cena proizvoda</i>
■ <i>vrpr</i> -	<i>vrednost proizvoda,</i>
■ <i>pop</i> -	<i>popust,</i>
■ <i>uknar</i> -	<i>ukupna vrednost,</i>
■ <i>ukpop</i> -	<i>ukupan popust,</i>
■ <i>ukvr</i> -	<i>cena koju treba platiti.</i>

NORMALNE FORME

- **Druga normalna forma** treba da eliminiše redundancu podataka što drugim rečima znači da: podatke koji se ponavljaju treba izdvojiti iz relacije, zatim formirati od svake izdvojene grupe koja se ponavlja novu relaciju, i konačno svakoj tako dobijenoj relaciji dodati novi ključ.
- U konkretnom primeru su podaci koji se ponavljaju u formularu narudžbe osenčeni, tako da od relacije NARUDŽBA dekompozicijom dobijamo dve relacije i to:
 - REL 1** (sadrži podatke o kupcu)
REL 1 <*brnar#*, *datnar*, *šifkp*, *nazkp*, *datis*, *prim*, *uknar*, *ukpop*, *ukvr*>
 - Odnosno:
 - REL2** (sadrži podatke o kupljenoj, to jest naručenoj robi)
REL2 <*brnar#*, *šifpr#*, *nazpr*, *kol*, *jm*, *cenpr*, *vrpr*, *pop*>
 - Relacija **REL2** dobija ovom dekompozicijom složeni ključ, i zato treba utvrditi zavisnost svakog atributa od tog složenog ključa. Ukoliko postoje zavisnosti među atributima, relaciju treba prevesti u višu, treću normalnu formu.

NORMALNE FORME

- U relaciji REL2, koja ima složeni ključ, atributi nazpr, jm i cenpr zavise samo od šifpr#, dakle od deela ključa, pa ih zato treba ponovo izdvojiti u posebnu relaciju. Ovako dekomponovani sistem, koji je u 2. normalnoj formi dobija slijedeći izgled:
 -
 - *REL 1 <brnar#, datnar, šifkp, nazkp, datis, prim, uknar, ukpop, ukvr >*
 - *REL21 <brnar#, šifpr#, kol, vrpr, pop >*
 - *REL22 <šifpr#, nazpr, jm, cenpr >*
 -
 - **Treća normalna forma** je sledeći korak kojim se uklanjanju i eventualne tranzitivne zavisnosti. U relaciji REL 1 atributi šifkp i nazkp zavise tranzitivno od ključnog atributa relacije brnar#, pa će se pojaviti redundanca podataka, to jest ponavljanje podataka o šifri, nazivu i adresi kupca prilikom svake narudžbe. Izdvajanjem ova dva atributa u zasebnu relaciju dolazi se do optimalne, 3 normalne forme:
 -
 - *REL 11 <brnar#, datnar, šifkp, datis, prim, uknar, ukpop, ukvr >*
 - *REL 12 <šifkp#, nazkp >*
 - *REL21 <brnar#, šifpr#, kol, vrpr, pop >*
 - *REL22 <šifpr#, nazpr, jm, cenpr >*

Prva normalna forma (1NF)

- Relaciju STUDENT možemo normalizovati (svesti na 1NF) ako u svakoj vrsti prikazane tabele, za svaki ispit koji je neki student položio, ponovimo i sve ostale njegove podatke (BI, IME, SEM, ŠSMER, IMERUK). Međutim, da bi se smanjila redundansa do koje bi, očigledno, ovakva normalizacija dovela, možemo ovako dobijenu relaciju, operacijom projekcije, dekomponovati na sledeće dve:
 - STUDENT1 (BI, IME, SEM, ŠSMER, IMERUK) (1)
 - PRIJAVA (BI, ŠPRED, NAZPRED, OCENA) (2)
- Prirodno spajanje ove dve relacije po BI rekonstruisalo bi polaznu normalizovanu relaciju.

Druga normalna forma (2NF)

- Definicije Druge, Treće i Boyce-Codd-ove normalne forme zasnivaju se na konceptu funkcionalne zavisnosti atributa relacije.
- Na primer u relaciji **STUDENT1** postoje sledeće funkcionalne zavisnosti od primarnog ključa:
 - BI -> IME
 - BI -> SEM
 - BI -> ŠSMER
 - BI -> IMERUK

Druga normalna forma (2NF)

- Primere potpune i nepotpune funkcionalne zavisnosti možemo pokazati na relaciji **PRIJAVA**:
- (a) Pošto važi sledeće:

$$\begin{array}{lll} \text{BI, ŠPRED} & \rightarrow & \text{OCENA} \\ \text{BI} & \rightarrow & \text{OCENA} \\ \text{ŠPRED} & \rightarrow & \text{OCENA} \end{array}$$

Atribut OCENA je potpuno funkcionalno zavisnan od složenog atributa BI, ŠPRED.

- (b) Pošto važi sledeće:

$$\begin{array}{lll} \text{BI, ŠPRED} & \rightarrow & \text{NAZPRED} \\ \text{BI} & \rightarrow & \text{NAZPRED} \\ \text{ŠPRED} & \rightarrow & \text{NAZPRED} \end{array}$$

Atribut NAZPRED je nepotpuno funkcionalno zavisan od složenog atributa BI, PRED, jer je funkcionalno zavisan i od njega i od jednog njegovog dela, od ŠPRED.

Druga normalna forma (2NF)

- Pokazuje se da je nepotpuna funkcionalna zavisnost redundantna, jer se može dedukovati preko proizvoda (kompozicije) zavisnosti:
 - f1: BI, ŠPRED \rightarrowtail ŠPRED
 - f2: ŠPRED \rightarrowtail NAZPRED
- Iz f1 i f2 sledi:
- $f3 = f1 \circ f2: BI, \check{S}PRED \rightarrowtail NAZPRED$, pa je nije neophodno pamtiti u bazi podataka.
- (Sa oznakom \circ se označava proizvod ili kompozicija funkcija. Funkcionalna zavisnost f1 je trivijalna funkcionalna zavisnost, zavisnost koja uvek važi, jer je podskup (ŠPRED) uvek funkcionalno zavisan od skupa (BI, ŠPRED)).

Druga normalna forma (2NF)

- Relacija R je u Drugoj normalnoj formi (2NF) ako i samo ako je u 1NF i svi njeni neključni atributi potpuno i funkcionalno zavise od primarnog ključa.
 - Neključni atributi (atributi koji nisu kandidati za ključ, niti deo kandidata za ključ) relacije prijava su NAZPRED i OCENA, a primarni ključ je složeni atribut BI, ŠPRED. Kako je ranije pokazano atribut OCENA je potpuno funkcionalno zavisao od primarnog ključa, a atribut NAZPRED nije. Zbog toga relacija PRIJAVA nije u 2NF.
 - Svođenje na 2NF vrši se dekompozicijom na taj način što u jednoj projekciji ostavlja primarni ključ i svi atributi koji su potpuno funkcionalno zavisni od njega, a u drugim projekcijama se realizuju one funkcionalne zavisnosti koje su prouzrokovale nepotpune funkcionalne zavisnosti. Za navedeni primer dekompozicija je:

PRIJAVA1(BI, ŠPRED, OCENA) (3)
PREDMET(ŠPRED, NAZPRED) (4)

Treća normalna forma (3NF)

- Relacija R je u Trećoj normalnoj formi (3NF) ako i samo ako je u 2NF i ako svi njeni neključni atributi netranzitivno funkcionalno zavise od primarnog ključa.
 - Zbog postojanja tranzitivne zavisnosti atributa IMERUK od primarnog ključa BI relacija STUDENT1 nije u 3NF.
 - Svođenje na 3NF vrši se dekompozicijom (bez gubljenja informacija) relacije u njene projekcije na taj način što se u jednoj projekciji ostavljaju primarni ključ i svi netranzitivno zavisni atributi, a u drugim projekcijama se realizuju funkcionalne zavisnosti koje su dovele do tranzitivnih zavisnosti.

STUDENT2(BI, IME, SEM, ŠSMER) (5)

SMER(ŠSMER, IMERUK) (6)

Normalizacija tabele Najmovi

Klijent Id	Ime Klijenta	Stan ID	Adresa ID	Pocetak najma	Kraj najma	Najam iznos	Vlasnik Id	Ime vlasnika
CR76	Mate Matić	PG4	Dečanska 8, Niš	01.07.00	31.08.01	350	C040	T.Tadić
		PG16	V.Pelagića 26 Niš	01.09.01	01.09.02	450	C093	E.Spasić
CR56	Petar Perić	PG4	Dečanska 8, Niš	01.09.99	10.06.00	350	C040	T.Tadić
		PG36	K.Miloša 26, Niš	10.10.00	01.12.01	375	C093	E.Spasić
		PG16	V.Pelagića 26 Niš	01.11.02	10.08.03	450	C093	E.Spasić

Prva normalna forma – 1NF

- *Tabela se nalazi u prvoj normalnoj formi ako su svi neključni atributi funkcionalno zavisni od ključa.*
- **Definicija proizlazi iz definicije ključa.** Naime, ako je neka vrednost atributa jedinstvena kroz celu tabelu, trivijalno je da su svi ostali atributi funkcionalno zavisni od tog atributa.
- **Uklanjanje ponavljačih atributa ili grupa atributa**

Prva normalna forma – 1NF

Klijent ID	Ime Klijenta	Stan ID	Adresa ID	Pocetak najma	Kraj najma	Najam iznos	Vlasnik ID	Ime vlasnika
CR76	Mate Matić	PG4	Dečanska 8, Niš	01.07.00	31.08.01	350	C040	T.Tadić
CR76	Mate Matić	PG16	V.Pelagića 26 Niš	01.09.01	01.09.02	450	C093	E.Spasić
CR56	Petar Perić	PG4	Dečanska 8, Niš	01.09.99	10.06.00	350	C040	T.Tadić
CR56	Petar Perić	PG36	K.Miloša 26, Niš	10.10.00	01.12.01	375	C093	E.Spasić
CR56	Petar Perić	PG16	V.Pelagića 26 Niš	01.11.02	10.08.03	450	C093	E.Spasić

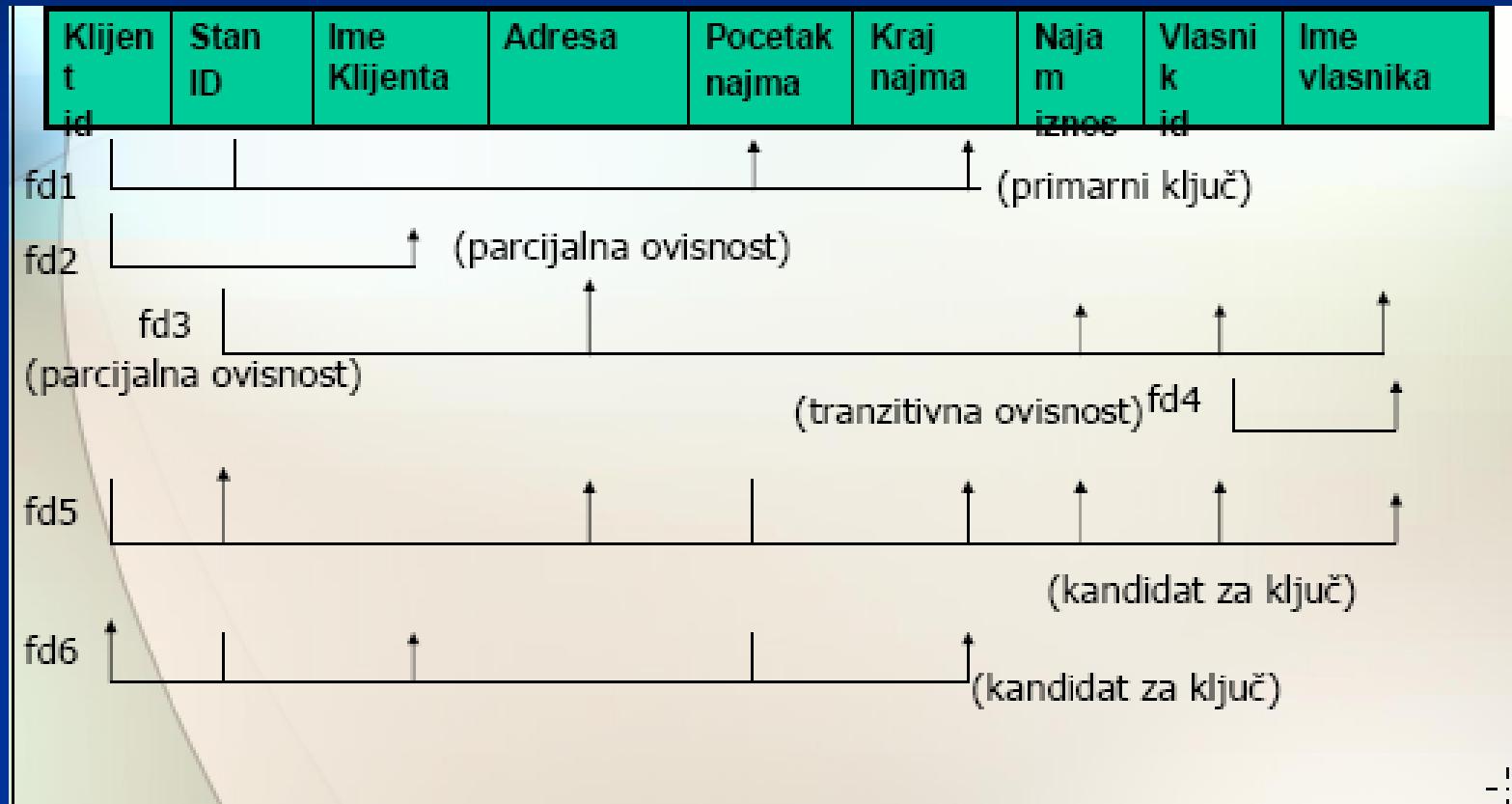
1NF - Problem

- Problem sa relacijom **Najmovi** je u tome što ona sadrži previše informacija, tj. sadrži informacije:
 - i o klijentu,
 - i o stanovima,
 - o najmovima
 - i najmodavcima.
- Jedna od bitnih postavki normalizacije jeste ta, da je odvojene informacije potrebno stavljati u odvojene tabele.

Druga normalna forma – 2NF

- Tabela je u drugoj normalnoj formi ako i samo ako je u 1NF i ako su svi neključni atributi potpuno funkcionalno zavisni od ključa ili svih ključeva.
- Odnosno, vršimo uklanjanje atributa zavisnih samo od dela jedinstvenog identifikatora – ključa.

Druga normalna forma – 2NF



Druga normalna forma – 2NF

Klijent ID	Ime Klijenta
CR76	Mate Matić
CR56	Petar Perić

Klijent ID	Stan ID	Pocetak najma	Kraj najma
CR76	PG4	01.07.00	31.08.01
CR76	PG16	01.09.01	01.09.02
CR56	PG4	01.09.99	10.06.00
CR56	PG36	10.10.00	01.12.01
CR56	PG16	01.11.02	10.08.03

Druga normalna forma – 2NF

Stan ID	Adresa ID	Najam iznos	Vlasnik ID	Ime vlasnika
PG4	Dečanska 8, Niš	350	C040	T.Tadić
PG16	V.Pelagića 26 Niš	450	C093	E.Spasić
PG36	K.Miloša 26, Niš	375	C093	E.Spasić

Treća normalna forma – 3NF

- Tabela je u trećoj normalnoj formi ako i samo ako je u 2NF i ako ni jedan neključni atribut nije tranzitivno zavistan od ključa.
- Odnosno, vršimo uklanjanje atributa zavisnih od atributa koji nisu deo jedinstvenog identifikatora.

Treća normalna forma – 3NF

Klijent ID	Ime Klijenta
CR76	Mate Matić
CR56	Petar Perić

Klijent ID	Stan ID	Pocetak najma	Kraj najma
CR76	PG4	01.07.00	31.08.01
CR76	PG16	01.09.01	01.09.02
CR56	PG4	01.09.99	10.06.00
CR56	PG36	10.10.00	01.12.01
CR56	PG16	01.11.02	10.08.03

Treća normalna forma – 3NF

Stan ID	Adresa ID	Najam iznos	Vlasnik ID	Ime vlasnika	Vlasnik ID	Ime vlasnika
PG4	Dečanska 8, Niš	350	C040	T.Tadić	C040	T.Tadić
PG16	V.Pelagića 26 Niš	450	C093	E.Spasić	C093	E.Spasić
PG36	K.Miloša 26, Niš	375	C093	E.Spasić	C093	E.Spasić

Boyce-Coddova normalna forma (BCNF)

■ Problemi sa 3NF :

- Ako postoje 2 ili više kandidata za ključ koji imaju slijedeće osobine:
 - kandidati za ključ su složeni (kompozitni)
 - kandidati za ključ se preklapaju tj. imaju barem jedan zajednički atribut

=> Potrebna je “stroža” definicija => BCNF

Boyce-Coddova normalna forma (BCNF)

- Tabela je u Boyce-Coddovoj normalnoj formi ako i samo ako je svaki determinant ujedno i kandidat za ključ
- Determinant je atribut od koga je neki atribut potpuno funkcionalno zavistan.
- Primjer:
- Dob_Prod(#D,Status,Grad,#P,Koli \square ina), -> atributi D, Grad i (D,P) su determinanti.

Boyce-Coddova normalna forma (BCNF)

- Dob_Proiz(#D,Dnaziv,P#, Kol)
- Kandidati za ključ:
 $(D\#, P\#)$ i $(Dnaziv, P\#)$
- Relacija nije u BCNF jer ima dve determinante $D\#$ i $Dnaziv$ koje nisu kandidati za ključeve.
- ($D\#$ i $Dnaziv$ su determinate jer određuju jedan drugog \rightarrow potpuna funkcionalna zavisnost)

Boyce-Coddova normalna forma (BCNF)

- **DOB_GRAD(#D,Grad) -> 3NF, BCNF, binarna**
 - **Dob_Status(#D,Status) -> 3NF, BCNF, binarna**
 - **Dobav(#D,Naziv,Grad,Status) -> BCNF (kandidati za ključ su D i Naziv, a atributi Grad i Status nisu međusobno funkcionalno zavisni)**
- **Primeri tabela koje NISU u BCNF:**
 - **Dob_Prod(#D,Status,Grad,#P,Kolicina) ... od svih determinanti jedino je složeni atribut (D,P) kandidat za ključ**
 - **Dob(#D,Grad,Status) .. Nije u BNCF jer su njene determinante D i Grad, a Grad nije kandidat za ključ.**

Funkcionalne zavisnosti atributa

- Funkcionalna zavisnost atributa relacija je vrsta ograničenja integriteta koja generalizuju koncept ključa.
- Posmatrajmo relaciju R koja ima više atributa. Neki atributi mogu biti složeni što znači da se sastoje od više prostih atributa.
Prepostavimo da u relaciji R postoje atrbute X i Y, koji mogu biti i složeni. Za atribut Y se kaže da je **funkcionalno zavisan** od atributa X, ako i samo ako svakoj vrednosti atributa X odgovara samo jedna vrednost atributa Y. Ova funkcionalna zavisnost se simbolički predstavlja kao:
- $R.X \rightarrow R.Y$
- što se čita kao: Y je funkcionalno zavisno od X, ili X funkcionalno određuje Y.

Tranzitivna zavisnost

- U relaciji R koja ima attribute X, Y, Z, atribut Z je **tranzitivno zavisan** od atributa X ako je:

$$R.X \rightarrow R.Y$$

$$R.Y \rightarrow R.Z$$

$$R.X \rightarrow R.Z$$

$$R.Y \dashrightarrow R.X$$

$$R.Z \dashrightarrow R.X$$

- Drugim rečima atribut Z je tranzitivno funkcionalno zavisan od atributa X ako je funkcionalno zavisan od X i atributa Z koji je takođe funkcionalno zavisan od A.

NORMALIZACIJA

PRIMERI

Normalizacija – primer 1

- Ilustrovaćemo probleme održavanja i izveštavanja na primeru jedne nenormalizovane relacije.
- Nenormalizovana relacija je ranije definisana kao relacija koja poseduje neke vrednosti atributa koje nisu "atomske", odnosno relacija koja poseduje "grupe sa ponavljanjem" i ugnježdene relacije.
- Posmatrajmo sledeću šemu baze:
STUDENT < BI, IME, SEM, ŠSMER, IMERUK, ŠPRED, NAZPRED, OCENA >

Normalizacija – primer 1

Prva normalna forma (1NF)

- Relaciju STUDENT:

STUDENT < BI, IME, SEM, ŠSMER, IMERUK, ŠPRED, NAZPRED, OCENA >

- možemo normalizovati (svesti na 1NF: svi atributi atomarni, domeni atributa u nove relacije, ugnježdene relacije u nove relacije, svi atributi potpuno zavisni od svog ključa u nove relacije) ako u svakoj vrsti prikazane tabele, za svaki ispit koji je neki student položio, ponovimo i sve ostale njegove podatke (BI, IME, SEM, ŠSMER, IMERUK). Relacija bi se širila “na dole ”.
- Međutim, bilo bi dosta ponavljanja podataka, pa da bi se smanjila redundansa do koje bi, očigledno, ovakva normalizacija dovela, možemo ovako dobijenu relaciju, normalizacijom u Prvu normalnu formu:
 - konstatovanjem da su svi atributi atomarni
 - konstatovanjem da nema domena atributa
 - konstatovanjem da POSTOJI ugnježdena relacija sa ključem ŠPRED
- dekomponovati na sledeće dve:

Normalizacija – primer 1

Prva normalna forma (1NF)

STUDENT < BI, IME, SEM, ŠSMER, IMERUK, ŠPRED, NAZPRED, OCENA >

- STUDENT1 (BI, IME, SEM, ŠSMER, IMERUK) (1)

- PRIJAVA (ŠPRED, BI, NAZPRED, OCENA) (2)
- Ključ nove relacije PRIJAVA ostaje ŠPRED ali radi povezivanja sa predhodnom mora imati i ključ BI. Prirodno spajanje ove dve relacije po BI rekonstruisalo bi polaznu normalizovanu relaciju.

Normalizacija – primer 1

Druga normalna forma (2NF)

STUDENT < BI, IME, SEM, ŠSMER, IMERUK, ŠPRED, NAZPRED, OCENA >

Potražićemo postojanje funkcionalnih zavisnosti:

- Na primer u relaciji (1):

STUDENT1 (BI, IME, SEM, ŠSMER, IMERUK)

postoje sledeće funkcionalne zavisnosti:

- $BI \rightarrow IME$
- $BI \rightarrow SEM$
- $BI \rightarrow \underline{\text{ŠSMER}}$
- $BI \rightarrow IMERUK$

pa svi atributi potpuno funkcionalno zavise od ključa BI

Normalizacija – primer 1

Druga normalna forma (2NF)

STUDENT < BI, IME,SEM,ŠSMER,IMERUK,ŠPRED,NAZPRED,OCENA >

- Primećujemo potpune i nepotpune funkcionalne zavisnosti u odnosu na složeni ključ na relaciji **PRIJAVA**:

PRIJAVA (ŠPRED, BI, NAZPRED, OCENA)

(a) Pošto važi sledeće:

- BI, ŠPRED → OCENA
- BI →! OCENA
- ŠPRED →! OCENA

atribut OCENA je potpuno funkcionalno zavisnan od složenog atributa BI, ŠPRED.

b) Pošto važi sledeće:

- BI, ŠPRED → NAZPRED
- BI →! NAZPRED
- ŠPRED → NAZPRED

atribut NAZPRED je nepotpuno funkcionalno zavisan od složenog atributa BI, ŠPRED, jer je funkcionalno zavisan i od njega a i od jednog njegovog dela, od ŠPRED.

Normalizacija – primer 1

Druga normalna forma (2NF)

- Neključni atributi (atributi koji nisu kandidati za ključ, niti deo kandidata za ključ) relacije prijava su **NAZPRED** i **OCENA**, a primarni ključ je složeni atribut **BI**, **ŠPRED**.
- Kako je ranije pokazano atribut **OCENA** je potpuno funkcionalno zavisan od primarnog ključa, a atribut **NAZPRED** nije. Zbog toga relacija **PRIJAVA** nije u 2NF.
- Svođenje na 2NF vrši se dekompozicijom na taj način što u jednoj projekciji ostavlja primarni ključ i svi atributi koji su potpuno funkcionalno zavisni od njega, a u drugim projekcijama se realizuju one funkcionalne zavisnosti koje su prouzrokovale nepotpune funkcionalne zavisnosti, definisane na predthodnom slajdu. Za navedeni primer dekompozicija je:

PRIJAVA (ŠPRED, BI, NAZPRED, OCENA)

- **PRIJAVA1(BI, ŠPRED, OCENA).....(3)**
- **PREDMET(~~ŠPRED~~, NAZPRED)(4)**

Normalizacija – primer 1

Treća normalna forma (3NF)

STUDENT < BI, IME, SEM, ŠSMER, IMERUK, ŠPRED, NAZPRED, OCENA >

- Posmatrajmo relaciju

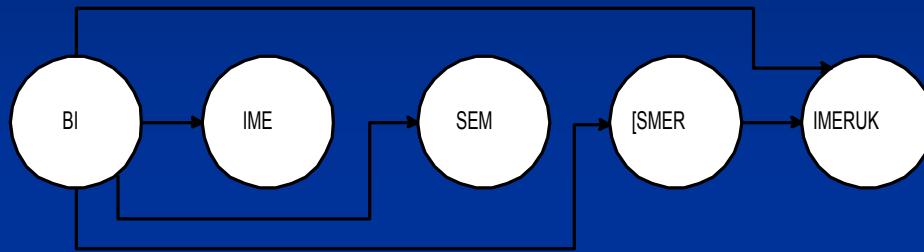
STUDENT1(BI, IME, SEM, ŠSMER, IMERUK).

- Očigledno je da u ovoj relaciji postoji redundansa podataka (isto **IMERUK** se pojavljuje uz svakog studenta upisanog na odgovarajući smer) i da postoje anomalije u održavanju:
 - Ne mogu se dodati informacije o novom smeru dok ga neki student ne upiše.
 - Ako se iz relacije **STUDENT1** izbaci student koji je jedini bio upisan na neki smer, gube se informacije i o tom smeru.
 - Ako se promeni rukovodioc smera atribut **IMERUK** treba promeniti na onoliko mesta koliko je studenata upisano na taj smer.

Razlog za redundansu i anomalije u održavanju je postojanje tranzitivne funkcionalne zavisnosti (za koju smo pokazali da je redundantna) atributa **IMERUK** od atributa **BI**.

Normalizacija – primer 1

Treća normalna forma (3NF)



- Tranzitivna **FZ.** se javlja kod atributa **IMERUK** koji je tranzitivno funkcionalno zavisan, preko ključa **ŠSMERA** od ključa **BI**.

- f1: $BI \rightarrow \check{SSMER}$
- f2: $\check{SSMER} \rightarrow IMERUK$

Normalizacija – primer 1

Treća normalna forma (3NF)

- Zbog postojanja tranzitivne zavisnosti atributa **IMERUK** od primarnog ključa **BI** relacija **STUDENT1**

STUDENT1(BI, IME, SEM, ŠSMER, IMERUK).

nije u **3NF**.

- Svođenje na **3NF** vrši se dekompozicijom (bez gubljenja informacija) relacije u njene projekcije na taj način što se u jednoj projekciji ostavljaju primarni ključ i svi netranzitivno zavisni atributi, a u drugim projekcijama se realizuju funkcionalne zavisnosti koje su dovele do tranzitivnih zavisnosti.

STUDENT2(BI, IME, SEM, ŠSMER).....(5)

SMER(ŠSMER, IMERUK).....(6)

Normalizacija – primer 1

Treća normalna forma (3NF)

- Ovako definisan sistem se jasno prepoznaje u modelu BP sa relacijama u 3NF:

(**STUDENT2, PREDMET, SMER i PRIJAVA1**).

STUDENT2(BI, IME, SEM, ŠSMER)

PRIJAVA1(BI, ŠPRED, OCENA)

PREDMET(ŠPRED, NAZPRED)

SMER(ŠSMER, IMERUK)

Normalizacija – primer 2

- Pogledajmo primer namerno loše oblikovane relacije **VTSNIS** u koju želimo zabeležiti podatke o **ST_PROGRAM_u**, nastavniku koji ga drži, kabinetu gde nastavnik radi, studentima koji su upisali **ST_PROGRAM**, ali i ocenama koje su studenti dobili na ispitu. Dakle, u tu relaciju želimo natrpati sve šta znamo i predmetima, ispitima, nastavnicima a to je, naravno, vrlo loše.
- Odredićemo prvo sve moguće primarne i sekundarne ključeve.
- **VTSNIS <INDEKS#, ST_PROGRAM##, NAZIV_ST_PROGRAMA, NASTAVNIK, KABINET, OCENA>**
- Prepostavimo da relacija zadovoljava osnovne elemente normalizacije u Prvu normalnu formu!

Normalizacija - primer 2

- **VTSNIS <INDEKS#, ST_PROGRAM##, NAZIV_ST_PROGRAMA, NASTAVNIK, KABINET, OCENA>**
- Zatim, u toj relaciji tražimo pojavu potpunih i nepotpunih funkcionalnih zavisnosti, a posle toga i moguće postojanje tranzitivnih funkcionalnih zavisnosti, naravno prema ostalim atributima u šemi baze :
- **(INDEKS#, ST_PROGRAM##) –? OCENA**
- Atribut **OCENA** je potpuno funkcionalno zavisан od složenog ključa, jer ako izbacimo bilo koji atribut iz skupa **(INDEKS#, ST_PROGRAM##)**, funkcionalna zavisnost se gubi.
- **ST_PROGRAM## ↔? NAZIV_ST_PROGRAMA**
- **ST_PROGRAM## ↔? NASTAVNIK**
- **ST_PROGRAM## ↔? KABINET**
- Sa druge strane, **NAZIV_ST_PROGRAMA, NASTAVNIK, KABINET** su parcijalno zavisni od složenog ključa, jer u stvari, samo zavise od ključa **ST_PROGRAM##**, ali ne i od ključa **INDEKS#**.

Normalizacija - primer 2

- Šta se dešava sa atributima **ST_PROGRAM## ↔? NASTAVNIK ↔? KABINET**
 - **NASTAVNIK ↔? KABINET**
 - Za **KABINET** se kaže da je tranzitivno zavisan od **ST_PROGRAM##**, jer je u stvari zavisan od **NASTAVNIK**, a **NASTAVNIK** je zavisan od **ST_PROGRAM##**:
 - **ST_PROGRAM## ↔? NASTAVNIK ↔? KABINET**
- 
- Parcijalne i tranzitivne zavisnosti mogu uzrokovati probleme kod manipulisanja sa podacima, pa ih je poželjno ukloniti. Tu nam pomaže postupak normalizacije.

Normalizacija - primer 2

- **Druga normalna forma**
- Relacija je u **drugoj normalnoj formi** (2NF) ako je u 1NF i ako je svaki neprimarni atribut potpuno funkcionalno zavisan od primarnog ključa.
- Malo pre definisana relacija VTSNIS jasno nije u 2NF. To čini probleme, kao npr.
 1. Ako se želi dodati podaci o novom ST_PROGRAMu, to se ne može učiniti dok ga ni jedan student nije upisao. Jer je INDEKS# deo primarnog ključa.
 2. Ako želimo uneti podatke o novom nastavniku i njegovom kabinetu, to se ne može uraditi dok ga ne zadužimo sa bar jednim ST_PROGRAMom koji predaje, odnosno, dok taj ST_PROGRAM nije upisao bar jedan student.
 3. Ako se želi promeniti naslov ST_PROGRAMa, ah to se mora učiniti na velikom broju mesta, u svim n-torkama gde se on pojavljuje (prema broju studenata koji ga slušaju).
 4. A šta ako svi studenti odustanu od navedenog ST_PROGRAMa. Tada će se morati pobrisati sve n-torce koje odgovaraju studentima koji ST_PROGRAM slušaju, i kao rezultat nestati će podaci o ST_PROGRAMu iz baze podataka.

Normalizacija - primer 2

- Druga normalna forma relacije
- **VTSNIS <INDEKS#, ST_PROGRAM##, NAZIV_ST_PROGRAMA, NASTAVNIK, KABINET, OCENA>**
- Ako se baza podataka dovede u 2NF ovi problemi nestaju. To je moguće postići razbijanjem relacije **VTSNIS** u dve relacije prema definiciji:
- **VTSNIS(INDEKS#, ST_PROGRAM##, OCENA)**
- **ST_PROGRAM (ST_PROGRAM##, NAZIV_ST_PROGRAMA, NASTAVNIK, KABINET)**
- Obe relacije su sada u 2NF.
- Ipak relacija **ST_PROGRAM** će tražiti dalju obradu jer još uvek postoji tranzitivna zavisnost:
- **ST_PROGRAM## ↔? NASTAVNIK ↔? KABINET**

Normalizacija - primer 2

- Treća normalna forma
- Relacija je u trećoj normalnoj formi (3NF) ako je u 2NF i ako ne sadrži tranzitivne zavisnosti.
- *Alternativna definicija je: Svaki neključni atribut mora zavisiti od ključa, i to od celog ključa, i ni o čemu drugom nego ključa.*
- Malo pre definisana relacija **ST_PROGRAM** nije u 3NF zbog tranzitivne zavisnosti.
- To može dovesti do poteškoća u radu:
 1. Ne mogu se upisati podaci o novom nastavniku i njegovom kabinetu dok još nije zadužen za nekim ST_PROGRAM.
 2. Ako se menja broj kabineta nastavnika, onda je to potrebno učiniti na više mesta, na svim n-torkama prema broju ST_PROGRAMa koje drži.
 3. Ako nastavnik privremeno ne predaje ni jedan ST_PROGRAM podaci o njemu nastaju iz baze podataka.

Normalizacija - primer 2

- Treća normalna forma relacije

VTSNIS <INDEKS#, ST_PROGRAM##, NAZIV_ST_PROGRAMA, NASTAVNIK, KABINET, OCENA>

- Stoga je relaciju **ST_PROGRAM**

ST_PROGRAM (ST_PROGRAM##, NAZIV_ST_PROGRAMA, NASTAVNIK, KABINET)

potrebno razbiti u dve relacije, i tako prekinuti tranzitivnu zavisnost. Konačno bi početna relacija **VTSNIS** bila razbijena na 3 relacije:

- **VTSNIS(INDEKS#, ST_PROGRAM##, OCENA)**
- **ST_PROGRAM (ST_PROGRAM##, NAZIV_ST_PROGRAMA, NASTAVNIK)**
- **NASTAVNIK (NASTAVNIK##, KABINET)**

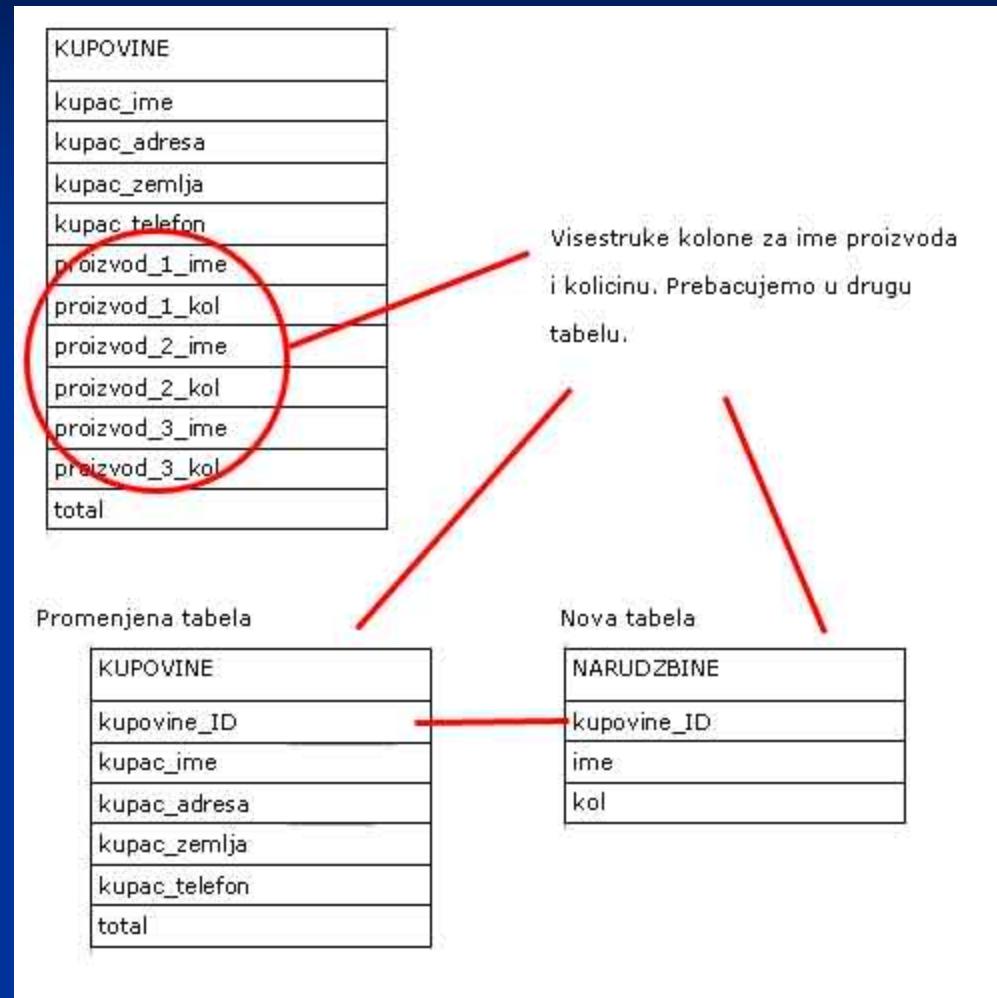
koje su do kraja normalizovane. U stvari, ovu bi šemu dobili i odmah, direktno iz ER-sheme da smo u početnoj fazi pravilno identifikovali entitete.

Normalizacija – primer 3

■ Prva normalna forma

Pravilo glasi: ako se neka kolona više puta ponavlja u nekoj tabeli, onda ona treba da se izbriše i postavi u posebnu tabelu.

Postoje više kolona za ime proizvoda i količinu što nije samo bespotrebno trošenje prostora na disku, već je kupovina ograničena na kupovinu tri proizvoda od jedanput. Potrebno je izbrisati višestruke kolone, ubaciti ih u novu tabelu, a staru tabelu povezati sa novonastalom.



Normalizacija – primer 3

- Druga normalna forma
- Pravilo glasi: da pored prve normalne forme, svi podaci koji nisu povezani sa ključem u tabeli, treba da se prebace u drugu tabelu.



Primer koji ilustruje problem je recimo da imate tabelu koja opisuje proizvode u prodavnici. Svaki zapis sadrži informaciju o proizvodu (ime, cena), postoji takođe i opis kategorije (miš, matična ploča...).

Pošto u nekoj kategoriji može biti više proizvoda, ime kategorije se ponavlja više puta to nije dobro, što zbog prostora, što zbog kasnijeg menjanja i izmena. Primer, ako smo inicijalno imali MSI Matične Ploče, a kasnije smo uveli novu kategoriju MSI Matične Ploče Green, ako ažuriramo samo neki od opisa postoji opasnost da dođe do dvomislenosti, a to sigurno niko ne želi.

Normalizacija – primer 3

■ Treća normalna forma

Ovo pravilo se odnosi na eliminisanje zavisnosti unutar tabele. I glasi: pored toga što treba da bude u skladu sa drugom normalnom formom, svi podaci koji nisu ključevi, a zvise od drugih podataka koji nisu ključevi, treba da se prebace u posebnu tabelu (tranzitivnost).



Normalizacija – primer 4

- Pogledajmo primer namerno loše oblikovane relacije **IZVEŠTAJ** da bi se te loše strane istakle:

**IZVEŠTAJ (INDEKS#, STUD_PROGRAM#, NAZIV_STUD_PROGRAM,
NASTAVNIK, SOBA_NAST.#, OCENA)**

- u koju želimo beležiti podatke o studijskom programu, nastavniku koji ga drži, sobu gde nastavnik radi, studente koji su upisali studijski program, ali i ocene koje su dobili na ispitu. Dakle, u tu relaciju želimo natrpati sve što znamo i predmetima, ispitima, nastavnicima i to je, naravno, loše. Tu toj relaciji nalazimo nekoliko funkcionalnih zavisnosti:

- **(INDEKS#, STUD_PROGRAM #) –! OCENA**
- **STUD_PROGRAM # –! NASLOV_STUD_PROGRAM**
- **STUD_PROGRAM # –! NASTAVNIK**
- **STUD_PROGRAM # –! SOBA_NAST.#**
- **NASTAVNIK –! SOBA_NAST.#**

- Atribut **OCENA** je potpuno funkcionalno zavisan od ključa, jer ako izbacimo bilo koji atribut iz skupa **(INDEKS#, STUD_PROGRAM #)**, funkcionalna zavisnost se gubi.
- Sa druge strane **NASLOV_STUD_PROGRAM**, **NASTAVNIK**, **SOBA_NAST.#** su parcijalno zavisni od ključa, jer u stvari, samo zavise od **STUD_PROGRAM #**, ali ne i od **INDEKS#**.

Normalizacija – primer 4

- **Druga normalna forma**
- Relacija je u **drugoj normalnoj formi** (2NF) ako je u 1NF i ako je svaki neprimarni atribut potpuno funkcionalno zavisan od primarnog ključa.
- Malo pre definisana relacija **IZVEŠTAJ** jasno nije u 2NF. To čini probleme, kao npr.
 - 1. Ako se žele dodati podaci o novom studijskom programu, to se ne može učiniti dok ga ni jedan student nije upisao. Jer je **INDEKS#** deo primarnog ključa.
 - 2. Ako želimo uneti podatke o novom nastavniku i njegovoj sobi, to se ne može uraditi dok ga ne zadužimo sa bar jednim studijskim programom koji predaje, odnosno, dok taj studijski program nije upisao bar jedan student.
 - 3. Ako se želi promeniti naslov studijskog programa, a to se mora učniti na velikom broju mesta, u svim n-torkama gde se on pojavljuje (prema broju studenata koji ga slušaju).
 - 4. A ako svi studenti odustanu od navedenog studijskog programa? Tada će se morati pobrisati sve n-torke koje odgovaraju studentima koji slušaju studijski program, i kao rezultat nestatće podaci o studijskom programu iz baze podataka.

Normalizacija – primer 4

- Druga normalna forma
- Ako se baza podataka dovede u 2NF ovi problemi nestaju. To je moguće postići razbijanjem relacije **IZVEŠTAJ** u dve relacije prema definiciji:
 - **IZVEŠTAJ** (**INDEKS#**, **STUD_PROGRAM #**, **OCENA**)
 - **STUD_PROGRAM** (**STUD_PROGRAM #**, **NASLOV_STUD_PROGRAM**, **NASTAVNIK**, **SOBA_NAST.#**)
- Obe relacije su sada u 2N. Ipak relacija **STUD_PROGRAM** će tražiti dalju obradu jer još uvek postoji tranzitivna zavisnost:
- **STUD_PROGRAM #** –! **NASTAVNIK** –! **SOBA_NAST.#**

Normalizacija – primer 4

- Treća normalna forma
- Relacija je u **trećoj normalnoj formi** (3NF) ako je u 2NF i ako ne sadrži tranzitivne zavisnosti.
- Alternativna definicija je: Svaki neključni atribut mora zavisiti od ključa, i to od celog ključa, i ni o čemu drugom nego od ključa.
- Malo pre definirana relacija **STUD_PROGRAM** nije u 3NF zbog tranzitivne zavisnosti.
- To može dovesti do poteškoća u radu:
 - 1. Ne mogu se upisati podaci o novom nastavniku i njegovoj sobi dok još nije zadužen za neki studijski program.
 - 2. Ako se menja broj sobe nastavnika, onda je to potrebno učiniti na više mesta, na svim n-torkama prema broju studijskih programa koje drži.
 - 3. Ako nastavnik privremeno ne predaje ni jedan studijski program podaci o njemu nastaju iz baze podataka.

Normalizacija – primer 4

- Treća normalna forma
- Stoga je relaciju **STUD_PROGRAM** potrebno razbiti u dve relacije, te tako prekinuti tranzitivnu zavisnost. Konačno bi početna relacija **IZVEŠTAJ** bila razbijena na 3 relacije:
- **IZVEŠTAJ** (INDEKS#, STUD_PROGRAM #, OCENA)
- **STUD_PROGRAM** (STUD_PROGRAM #, NASLOV_STUD_PROGRAM, NASTAVNIK)
- **NASTAVNIK** (NASTAVNIK#, SOBA_NAST.)
- koje su do kraja normalizovane. U stvari ovu bi šemu dobili i odmah, direktno iz ER-sheme da smo u početnoj fazi pravilno identifikovali entitete.

Normalizacija – primer 4

IZVEŠTAJ		
INDEKS#	STUD_PROGRAM #	OCENA

STUD_PROGRAM		
STUD_PROGRAM #	NASLOV_STUD_PROGRAM	NASTAVNIK

NASTAVNIK	
NASTAVNIK #	SOBA_NAST

Normalizacija – primer 5

- Nogomenta »sezona« 2014/2015 (izmišljeni podaci)

datum	domaći	gostujući	stadion	rezultat	sudija	adr_sud	vreme
05.01.04.	C.Zvezda	Partizan	C.Zvezda	2:1	Pera	ad_Pera	kiša
05.01.11.	Partizan	C.Zvezda	Partizan	2:2	Laza	ad_Laza	oblacčo
05.01.18.	Radnički Niš	Vojvodina	Čair	5:0	Mika	ad_Mika	sunce
05.01.25.	Vojvodina	Radnički Niš	Karađorđe	0:2	Joca	ad_Joca	sunce
05.02.04.	Radnički	Čukarički	Čika Dača	3:1	Dule	ad_Dule	oblačno
05.02.11.	Čukarički	Radnički	Čukarički	1:2	Mile	ad_Mile	sunce

Normalizacija – primer 5

■ Šema baze:

UTAKMICA (datum, domaći, gostujući, stadion, rezultat,
sudija, adr_sud, vreme)

- Primarni atributi (primarni ključ): (*datum, domaćin*)
- Postoje i drugi kandidati za ključ: (*datum, stadion*) ili (*datum, gostujući*) ili (*datum, sudija*)

Normalizacija – primer 5

- Imamo ove funkcionalne zavisnosti:

- $domaćin \rightarrow stadion$
- $sudija \rightarrow adr\ sud.$

- potpune funkcionalne ovisnosti od ključeva:

- $(datum, domaćin) \rightarrow gostujući$
- $(datum, domaćin) \rightarrow sudija$
- $(datum, domaćin) \rightarrow rezultat$

- parcijalne funkcionalne zavisnosti od ključeva:

- $(datum, domaćin) \rightarrow stadion$

- **tranzitivne** funkcionalne ovisnosti od ključeva:

- $(datum, domaćin) \rightarrow adr\ sud.$

jer: $(datum, domaćin) \rightarrow sudija \rightarrow adr\ sud.$

Normalizacija – primer 5

- Parcijalne i tranzitivne zavisnosti mogu uzrokovati probleme kod manipulisanja sa podacima, pa ih je poželjno ukloniti. Tu nam pomaže normalizacija.
- Npr. u slučaju "nogometne sezone":
 - 1. Imamo novog sudiju koji stanuje u Nišu, ali to nije moguće evidentirati u navedenoj relaciji jer sudija još nije sudio ni jednu utakmicu. Nije moguće upisati red ako su primarni atributi "prazni".
 - 2. Ako se sudija preselio u drugi grad, njegovu novu adresu moramo mijenjati na puno mesta.
 - Ovi problemi uobičajeno se nazivaju **anomalije**: anomalija kod upisa podataka, anomalija kod izmene podataka i anomalija kod brisanja podataka.

Normalizacija – primer 5

datum	domaći	gostujući	rezultat	sudija	adr_sud	vreme
05.01.04.	C.Zvezda	Partizan	2:1	Pera	ad_Pera	kiša
05.01.11.	Partizan	C.Zvezda	2:2	Laza	ad_Laza	oblacčo
05.01.18.	Radnički Niš	Vojvodina	5:0	Mika	ad_Mika	sunce
05.01.25.	Vojvodina	Radnički Niš	0:2	Joca	ad_Joca	sunce
05.02.04.	Radnički	Čukarički	3:1	Dule	ad_Dule	oblačno
05.02.11.	Čukarički	Radnički	1:2	Mile	ad_Mile	sunce

klub	stadion
C.Zvezda	C.Zvezda
Partizan	Partizan
Radnički Niš	Čair
Vojvodina	Karađorđe
Radnički	Čika Dača
Čukarički	Čukarički

U ovim relacijama nema parcijalnih zavisnosti pa je sad šema baze:

UTAKMICA (datum, domaćin, gostujući, rezultat, sudija, adr sud., vreme)

KLUB (klub, stadion)

Normalizacija – primer 5

datum	domaći	gostujući	rezultat	sudija	vreme
05.01.04.	C.Zvezda	Partizan	2:1	Pera	kiša
05.01.11.	Partizan	C.Zvezda	2:2	Laza	oblacčo
05.01.18.	Radnički Niš	Vojvodina	5:0	Mika	sunce
05.01.25.	Vojvodina	Radnički Niš	0:2	Joca	sunce
05.02.04.	Radnički	Čukarički	3:1	Dule	oblačno
05.02.11.	Čukarički	Radnički	1:2	Mile	sunce

klub	stadion
C.Zvezda	C.Zvezda
Partizan	Partizan
Radnički Niš	Čair
Vojvodina	Karađorđe
Radnički	Čika Dača
Čukarički	Čukarički

sudija	adr_sud
Pera	ad_Pera
Laza	ad_Laza
Mika	ad_Mika
Joca	ad_Joca
Dule	ad_Dule
Mile	ad_Mile

Šema baze sada izgleda:

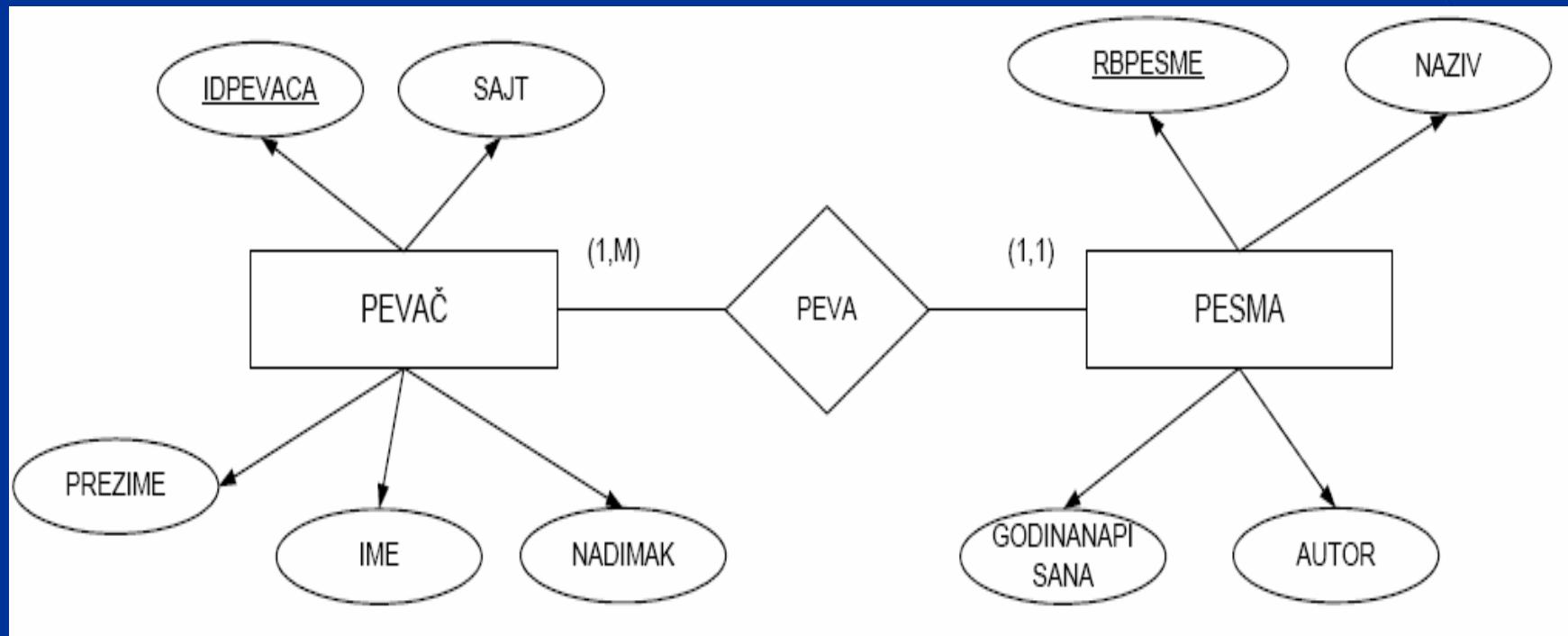
UTAKMICA (datum, domaćin, gostujući, rezultat, sudija, vreme)

KLUB (klub, stadion)

SUDIJA (sudija, adr sud.)

Normalizacija – primer 6

- Kreiranje baze podataka pod nazivom PESME, sa strukturuom tabela koja odgovara relacionom modelu podataka koji se dobija prevodenjem datog ER dijagrama u relacioni model. Ograničenja spoljnih ključeva formirati naknadno, dodavanjem ograničenja na nivou polja i tom prilikom treba definisati akcije referencijalnog integriteta u slučaju izmene i brisanja podataka u povezanim tabelama restriktivno – izvršiti normalizaciju.



Normalizacija – primer 6

- Domeni atributa su:

Naziv atributa	Tip podatka	Ograničenje	Obavezna vrednost
idprevaca	Ceo broj	Veceg opsega	da
ime	String	Max. 30 karaktera	da
prezime	String	Max. 30 karaktera	da
nadimak	String	Max. 20 karaktera	ne
rbpesme	Automatski brojač	1, 2, 3, ..., n	da
naziv	String	Max. 50 karaktera	da
autor	String	Max. 50 karaktera	ne
sajt	String	Max. 200 karaktera	ne
godinanapisana	Ceo broj	Vrednost mora biti veca od 1900 Inicijalna vrednost je 2011	da

Normalizacija – primer 6

■ ŠEMA RELACIJE

PESMA (idpesme, idpevaca, nazivpesme, datumizvodjenja, prezimepevaca, imepevaca, kompozitor, godinanastanka)

■ SE NE NALAZI U 1.NF, POŠTO ATRIBUT **KOMPOZITOR** NE UZIMA VREDNOSTI IZ SKUPA ČIJI SU ELEMENTI ATOMARNE VREDNOSTI.

1.NF:

■ **PESMA** (idpesme, idpevaca, nazivpesme, prezimepevaca, imepevaca, kompozitorime, kompozitorprezime, godinanastanka)

- Funkcionalne zavisnosti:
 - $\text{idpesme} \rightarrow \text{nazivpesme}, \text{kompozitor}, \text{godinanastanka}$
 - $\text{idpevaca} \rightarrow \text{prezimepevaca}, \text{imepevaca}$
 - $\text{idpevaca}, \text{idpesme} \rightarrow \text{datumizvodjenja}$

3.NF:

- **PESMA** (idpesme, nazivpesme, kompozitorime, kompozitorprezime, godinanastanka)
- **PEVAC** (idpevaca, prezimepevaca, imepevaca)
- **IZVODI** (idpevaca, idpesme, datumizvodjenja)

Normalizacija – primer 7

■ Podaci o radu Fizičkog odseka

indeks	student	kid	naslov	nastavnik	ured	ocena	datum
F-6543	Darko Milovanović	2141	Opšta fizika 1	M. Furić	306	4	10.09.02.
F-5432	Dejan Popović	2141	Opšta fizika 1	M. Furić	306	3	10.09.02.
F-9876	Vlatka Antolković	2471	Fizika lasera	A. Dulčić	217	3	15.09.02.
F-7654	Marijan Marciuš	2471	Fizika lasera	A. Dulčić	217	5	15.09.02.
F-7654	Marijan Marciuš	2477	Energetika	Đ. Miljanić	IRB	2	25.09.02.
F-8765	Monika Lukavečki	2477	Energetika	Đ. Miljanić	IRB	4	25.09.02.
F-4321	Saša Sambolek	2303	Elektrodinamika	S. Brant	415	1	20.09.02.
F-9876	Vlatka Antolković	2303	Elektrodinamika	S. Brant	415	4	20.09.02.
F-4321	Saša Sambolek	2303	Elektrodinamika	S. Brant	415	4	12.12.02.

Šema baze je:

KOLEGIJUM (datum,indeks,kid,student,naslov,nastavnik,ured,ocena)

Primarni atributi - (primarni ključ) : (datum,indeks,kid)

Postoje i drugi kandidati za ključ: (datum,indeks,naslov) ili (datum,student,kid)

Normalizacija – primer 7

- Imamo ove funkcionalne zavisnosti:

$(\text{datum}, \text{indeks}, \text{kid}) \rightarrow \text{ocena}$

$\text{kid} \rightarrow \text{naslov}$

$\text{kid} \rightarrow \text{nastavnik}$

$\text{kid} \rightarrow \text{ured}$

$\text{nastavnik} \rightarrow \text{ured}$

- potpune funkcionalne zavisnosti:

$(\text{datum}, \text{indeks}, \text{kid}) \rightarrow \text{ocena}$

- parcijalne funkcionalne zavisnosti:

$(\text{datum}, \text{indeks}, \text{kid}) \rightarrow \text{naslov}$

$(\text{datum}, \text{indeks}, \text{kid}) \rightarrow \text{nastavnik}$

$(\text{datum}, \text{indeks}, \text{kid}) \rightarrow \text{ured}$

$(\text{datum}, \text{indeks}, \text{kid}) \rightarrow \text{student}$

- tranzitivne funkcionalne ovisnosti:

$\text{kid} \rightarrow \text{ured}$ (jer tranzitivno važi: $\text{kid} \rightarrow \text{nastavnik} \rightarrow \text{ured}$)

Normalizacija – primer 7

indeks	kid	ocena	datum
F-6543	2141	4	10.09.02.
F-5432	2141	3	10.09.02.
F-9876	2471	3	15.09.02.
F-7654	2471	5	15.09.02.
F-7654	2477	2	25.09.02.
F-8765	2477	4	25.09.02.
F-4321	2303	1	20.09.02.
F-9876	2303	4	20.09.02.
F-4321	2303	4	12.12.02.

indeks	student
F-4321	Saša Sambolek
F-5432	Dejan Popović
F-6543	Darko Milodanović
F-7654	Marijan Marciuš
F-8765	Monika Lukavečki
F-9876	Vlatka Antolković

kid	naslov	nastavnik	ured
2141	Opšta fizika 1	M. Furč	306
2303	Elektrodinamika	S. Brant	415
2471	Fizika lasera	A. Dulčić	217
2477	Energetika	Đ. Miljanić	IRB

Šema:

ISPIT (datum, indeks, kid, ocena)

STUDENT (indeks, student)

KOLEGIJUM (kid, naslov, nastavnik, ured)

- Druga normalna forma na osnovu funkcionalnih zavisnosti.
- Primećujemo da nema parcijalnih zavisnosti, ali još uvek postoje tranzitivne zavisnosti!

Normalizacija – primer 7

indeks	kid	ocena	datum
F-6543	2141	4	10.09.02.
F-5432	2141	3	10.09.02.
F-9876	2471	3	15.09.02.
F-7654	2471	5	15.09.02.
F-7654	2477	2	25.09.02.
F-8765	2477	4	25.09.02.
F-4321	2303	1	20.09.02.
F-9876	2303	4	20.09.02.
F-4321	2303	4	12.12.02.

indeks	student
F-4321	Saša Sambolek
F-5432	Dejan Popović
F-6543	Darko Milodanović
F-7654	Marijan Marciuš
F-8765	Monika Lukavečki
F-9876	Vlatka Antolković

kid	naslov	nastavnik
2141	Opšta fizika 1	M. Furč
2303	Elektrodinamika	S. Brant
2471	Fizika lasera	A. Dulčić
2477	Energetika	Đ. Miljanić

nastavnik	ured
M. Furić	306
S. Brant	415
A. Dulčić	217
Đ. Miljanić	IRB

Šema:

ISPIT (datum, indeks, kid, ocena)

STUDENT (indeks, student)

KOLEGIJUM (kid, naslov, nastavnik)

URED (nastavnik, ured)

NORMALIZACIJA RELACIJE STUDENTI:

1NF

Studenti

BrojIndeksa	SifraPredmeta	Ime	Prezime	Semestar	Ocena	Predmet	Profesor	SifraSmera	Smer
199	MAT	Ana	Jeftić	2	7	Matematika	Ristić	01	IT
199	IT101				8	Osnove IT	Gogić	01	IT
454	IT101	Andrija	Jovanović	1	8	Osnove IT	Gogić	02	SE
124	IT101	Zoran	Kostić	1	10	Osnove IT	Gogić	01	IT
124	ENG100				7	Engleski I	Mišić	01	IT
124	MAT				6	Matematika	Ristić	01	IT
215	ENG100	Slavko	Stojanović	2	9	Engleski I	Mišić	02	SE
215	PR200				10	Programiranje	Bojić	02	SE

1NF

Studenti

BrojIndeksa	SifraPredmeta	Ime	Prezime	Semestar	Ocena	Predmet	Profesor	SifraSmera	Smer
199	MAT	Ana	Jeftić	2	7	Matematika	Ristić	01	IT
199	IT101	Ana	Jeftić	2	8	Osnove IT	Gogić	01	IT
454	IT101	Andrija	Jovanović	1	8	Osnove IT	Gogić	02	SE
124	IT101	Zoran	Kostić	1	10	Osnove IT	Gogić	01	IT
124	ENG100	Zoran	Kostić	1	7	Engleski I	Mišić	01	IT
124	MAT	Zoran	Kostić	1	6	Matematika	Ristić	01	IT
215	ENG100	Slavko	Stojanović	2	9	Engleski I	Mišić	02	SE
215	PR200	Slavko	Stojanović	2	10	Programiranje	Bojić	02	SE

Formalni opis relacije

- Studenti (BrojIndeksa, SifraPredmeta, Ime, Prezime, Semestar, Ocena, Predmet, Profesor, SifraSmera, Smer)
- **Primarni ključ** relacije Studenti je složeni ključ koji se sastoji od atributa BrojIndeksa, SifraPredmeta,
- **Funkcionalne zavisnosti:**
- BrojIndeksa, SifraPredmeta → Ime, Prezime, Semestar, Ocena, Predmet, Profesor, SifraSmera, Smer
- BrojIndeksa → Ime, Prezime, Semestar, SifraSmera, NazivSmera
- SifraPredmeta → Predmet, Profesor
- SifraSmera → Smer

2NF

- Relacija je u **drugoј normalnoј formi** ako je relacija u prvoј normalnoј formi i svi atributi potpuno funkcionalno zavise od primarnog ključa, a ne od nekog njegovog dela.
- 2NF je uvek ispunjena kada je primarni ključ prost atribut, odnosno kada se primarni ključ sastoji samo od jednog atributa.

2NF

Iz funkcionalne zavisnosti relacije Studenti

BrojIndeksa, SifraPredmeta → Ime, Prezime,
Semestar, Ocena, Predmet, Profesor, SifraSmera,
Smer

se vidi:

- Primarni ključ ove relacije je složeni ključ BrojIndeksa, SifraPredmeta
- Svi ne-ključni atributi osim atributa Ocena su nepotpuno funkcionalno zavisni od primarnog ključa.

2NF

- Relacija Studenti može da se prevede u 2NF dekompozicijom relacije Studenti na tri sledeće relacije:
- Studenti1 (BrojIndeksa, Ime, Prezime, Semestar, SifraSmera, Smer)
- Prijava (BrojIndeksa, SifraPredmeta, Ocena)
- Predmeti (SifraPredmeta, Predmet, Profesor)

2NF

Studenti1					
BrojIndeksa	Ime	Prezime	Semestar	SifraSmera	Smer
199	Ana	Jeftić	2	01	IT
199	Ana	Jeftić	2	01	IT
454	Andrija	Jovanović	1	02	SE
124	Zoran	Kostić	1	01	IT
124	Zoran	Kostić	1	01	IT
124	Zoran	Kostić	1	01	IT
215	Slavko	Stojanović	2	02	SE
215	Slavko	Stojanović	2	02	SE

Prijava		
BrojIndeksa	SifraPredmeta	Ocena
199	MAT	7
199	IT101	8
454	IT101	8
124	IT101	10
124	ENG100	7
124	MAT	6
215	ENG100	9
215	PR200	10

Predmeti		
SifraPredmeta	Predmet	Profesor
MAT	Matematika	Ristić
IT101	Osnove IT	Gogić
ENG100	Engleski I	Mišić
PR200	Programiranje	Bojić

3NF

- Kaže se da je relacija **u trećoj normalnoj formi** ako je u prvoj i drugoj normalnoj formi i ako svaki atribut koji nije deo primarnog ključa zavisi samo od primarnog ključa, odnosno ne postoji zavisnost između ne – ključnih atributa.
- Može se reći da je relacija u 3NF ako je u 2NF i ako svi njeni ne-ključni atributi zavise od primarnog ključa.
- Za 3NF se zahteva da svaka tabela sadrži podatke o samo jednom tipu entiteta.

3NF

- Relacija Studenti1 ne zadovoljava uslove 3NF jer postoji zavisnost između ne-ključnih atributa SifraSmera i Smer.
- Ova dva atributa su tranzitivno zavisna od primarnog ključa BrojIndeksa.
- Dekompozicija
- Studenti2 (BrojIndeksa, Ime, Prezime, Semestar)
- Smerovi (SifraSmera, Smer)

Studenti2			
BrojIndeksa	Ime	Prezime	Semestar
199	Ana	Jeftić	2
454	Andrija	Jovanović	1
124	Zoran	Kostić	1
215	Slavko	Stojanović	2

Smerovi	
SifraSmera	Smer
01	IT
02	SE

OSTALE FORME

- Postoji više od 3 forme normalizacije:
 - Boyce-Codd Normal Form,
 - Fourth Normal Form,
 - Fifth Normal Form/Join-Projection Normal Form.
- Retko se vrši normalizacija po ovim formama zato što su poboljšanja mala u odnosu na potreban rad i stepen iskorišćenja baze podataka.

Normalizacija tabele RADNIK

- RADNIK=Matbr, Prez, Ime, Datrod, Šifra odeljenja> Šema baze

<u>Matbr</u>	Prez	Ime	Datrod	Šifra odeljenja
1111	Novak	Ivan	28.12.60.	115
1121	Kostić	Ivana	16.10.55.	89
1133	Hristić	Krsta	19.03.48.	115

$K_{RADNIK} = \{ \underline{\text{Matbr}} \}$ ključ relacije

Domeni sadrže atomarne-jednostavne vrednosti

Postoje funkcionalne zavisnosti:

$\text{Matbr} \rightarrow \text{Prezime}$

$\text{Matbr} \rightarrow \text{Datrod}$

Relaciona shema RADNIK je u 1 NF

$\text{Matbr} \rightarrow \text{Ime}$

$\text{Matbr} \rightarrow \text{Šifra odeljenja}$

Normalizacija tabele RADNIK1

- Preduzeće evidentira podatke o radnicima i njihovoј deci - korisnicima zdravstvenog osiguranja u tabeli RADNIK1:

<u>Matbr</u>	Prez	Ime	Ime D	DatrodD
1111	Novak	Ivan	Jasna	21.01.83.
1111	Novak	Ivan	Vedran	13.12.85.
1121	Kostić	Ivana	Ivan	23.03.82.
1133	Hristić	Krsta	Petar	22.02.81.
1133	Hristić	Krsta	Ana	19.09.84.
1133	Hristić	Krsta	Ivan	05.11.87.

Normalizacija tabele RADNIK1

- K_{RADNIK1} = { Matbr }
- Domeni sadrže jednostavne vrednosti i
- postoji FZ:

Matbr → Prezime

Matbr → Ime

- ne postoji FZ:

Matbr → ImeD

Matbr → DatRodD

- Relaciona shema RADNIK1 nije u 1 NF

Normalizacija tabele RADNIK1

DETE

<u>Matbr</u>	<u>Ime D</u>	DatrodđD
1111	Jasna	21.01.83.
1111	Vedran	13.12.85.
1121	Ivan	23.03.82.
1133	Petar	22.02.81.
1133	Ana	19.09.84.
1133	Ivan	05.11.87.

$$K_{DETE} = \{ \underline{\text{Matbr}}, \underline{\text{ImeDj}} \}$$

RADNIK2

<u>Matbr</u>	Prez	Ime
1111	Novak	Ivan
1121	Kostić	Ivana
1133	Hristić	Krsta

$$K_{RADNIK2} = \{ \underline{\text{Matbr}} \}$$

- Normalizacija na 1NF
 - Izdvajanjem atributa koji nisu funkcionalno zavisni od ključa
 - U posebnu relaciju izdvaja se skup atributa koji se ponavlja, zajedno sa ključem originalne relacije

Normalizacija tabele RADNIK

- Normalizacija na 1NF promenom ključa u tabelu:
RADNIK3

<u>Matbr</u>	<u>Prez</u>	<u>Ime</u>	<u>Ime D</u>	<u>DatrodD</u>
1111	Novak	Ivan	Jasna	21.01.83.
1111	Novak	Ivan	Vedran	13.12.85.
1121	Kostić	Ivana	Ivan	23.03.82.
1133	Hristić	Krsta	Petar	22.02.81.
1133	Hristić	Krsta	Ana	19.09.84.
1133	Hristić	Krsta	Ivan	05.11.87.

$$K_{RADNIK3} = \{ \underline{\text{Matbr}}, \underline{\text{ImeD}} \}$$

Drugia normalna forma - 2 NF

- Relaciona shema RADNIK3 zadovoljava 1 NF
postoje FZ:

Matbr, ImeD → Prezime i

Matbr, ImeD → Ime

ali postoje i :

Matbr → Prezime

Matbr → Ime

Matbr, ImeD → Prezime i

Matbr, ImeD → Ime

} su nepotpune!!!!!!!

- Relaciona shema RADNIK3 ne zadovoljava 2 NF.

Druga normalna forma - 2 NF

- Normalizacijom na 2 NF nastaju:
 - Relaciona shema koja sadrži skup atributa koji su bili nepotpuno funkcionalno zavisni od ključa i dela ključa od koga su potpuno funkcionalno zavisni
 - Relaciona shema koja sadrži ključ originalne relacije i skup atributa koji su potpuno funkcionalno zavisni od ključa

Druga normalna forma - 2 NF

RADNIK4

<u>Matbr</u>	<u>Prez</u>	<u>Ime</u>
1111	Novak	Ivan
1121	Kostić	Ivana
1133	Hristić	Krsta

$$K_{RADNIK4} = \{ \underline{\text{Matbr}} \}$$

DETE

<u>Matbr</u>	<u>Ime D</u>	<u>DatrodD</u>
1111	Jasna	21.01.83.
1111	Vedran	13.12.85.
1121	Ivan	23.03.82.
1133	Petar	22.02.81.
1133	Ana	19.09.84.
1133	Ivan	05.11.87.

$$K_{DETE} = \{ \underline{\text{Matbr}}, \underline{\text{ImeD}} \}$$

Treća normalna forma - 3 NF

- Definicija:
- Relacija (relaciona shema) je u 3 NF ako je u 2 NF i ako: ni jedan atribut iz zavisnog dela nije tranzitivno zavistan od ključa ili ključeva

PRIMER POZ :

1. U biblioteci se evidentiraju pozajmice knjiga. Relaciona shema POZ sastoji se od sledećih atributa:

- SIFCLN - šifra člana
- PREZCLN - prezime člana
- IMECLN - ime člana
- DATPOZ - datum pozajmice
- DATVR – datum vraćanja
- PBR - poštanski broj mesta stanovanja člana
- NAZM - naziv mesta stanovanja člana
- ADRCLN - adresa člana
- SIFKNJ - šifra knjige
- NAZKNJ - naziv knjige
- INVBRPRIM – inventarski broj primerka knjige
- SIFIZD - šifra izdavača
- NAZIZD - naziv izdavača

Prepostavljeni ključ je: $K_{POZ} = \{ \underline{SIFCLN} \}$

Normalizujte relaciju POZ na 1NF, 2NF, 3NF ako vrijede pravila:

- Jedan član istoga dana može iznajmiti više knjiga
- Jedan član istu knjigu može iznajmiti više puta
- Jedna knjiga ima jednog izdavača

Šema relacije POZ<SIFCLN, PREZCLN, IMECLN, DATPOZ, DATVR, PBR, NAZM, ADRCLN, SIFKNJ, NAZKNJ, INVBRPRIM, SIFIZD, NAZIZD>

PRIMER POZ :

- POZ=<SIFCLN,PREZCLN,IMECLN,DATPOZ,DATVR,PBR,NAZM,ADRCLN,SIFKNJ,NAZKNJ, INVBRPRIM, SIFIZ, NAZIZD >
- Prepostavljeni ključ je:

$$K_{POZ} = \{ \underline{SIFCLN} \}$$

1NF

- CLAN1 = <SIFCLN,PREZCLN,IMECLN, PBR,NAZM,ADRCLN >

$$K_{CLAN1} = \{ \underline{SIFCLN} \}$$

- POZ1=<SIFCLN, INVBRPRIM, DATPOZ,DATVR, SIFKNJ,NAZKNJ, SIFIZ,NAZIZD >

$$K_{POZ1} = \{ \underline{SIFCLN}, \underline{INVBRPRIM}, \underline{DATPOZ} \}$$

2NF

- CLAN1 zadovoljava 2NF
- POZ1=<SIFCLN,INVBRPRIM,DATPOZ,DATVR,NAZKNJ,SIFKNJ,
SIFIZD,NAZIZD >
 $K_{POZ1} = \{ \underline{SIFCLN}, \underline{INVBRPRIM}, \underline{DATPOZ} \}$
- PRIM1=<INVBRPRIM, SIFKNJ, NAZKNJ,SIFIZD,NAZIZD >
 $K_{PRIM1} = \{ INVBRPRIM \}$
- POZAJMICA=<SIFCLN, INVBRPRIM,DATPOZ,DATVR >
 $K_{POZAJMICA} = \{ \underline{SIFCLN}, \underline{INVBRPRIM}, \underline{DATPOZ} \}$

3NF

- CLAN1 = <SIFCLN,PREZCLN,IMECLN, PBR,NAZMJ,ADRCLN >
 $K_{CLAN1} = \{ SIFCLN \}$
- CLAN = <SIFCLN,PREZCLN,IMECLN, PBR, ADRCLN >
 $K_{CLAN} = \{ SIFCLN \}$
- MESTO = <PBR,NAZMJ >
 $K_{MESTO} = \{ PBR \}$
- PRIM1= < INVBRPRIM, SIFKNJ, NAZKNJ, SIFIZD,NAZIZD >
 $K_{PRIM1} = \{INVBRPRIM\}$
- PRIMERAK=<INVBRPRIM, SIFKNJ >
 $K_{PRIMERAK} = \{INVBRPRIM\}$
- KNJ1=<SIFKNJ, NAZKNJ, SIFIZD,NAZIZD >
 $K_{KNJ1} = \{SIFKNJ\}$
- KNJIGA=<SIFKNJ, NAZKNJ, SIFIZD >
 $K_{KNJIGA} = \{SIFKNJ\}$
- IZDAVAC=SIFIZD,NAZIZD >
 $K_{IZDAVAC} = \{SIFIZD\}$
- POSUDBA=<SIFCLN, INVBRPRIM,DATPOZ,DATVR >
 $K_{POSUDBA} = \{ SIFCLN, INVBRPRIM, DATPOZ \}$

zadovoljava 3NF