

Baze podataka

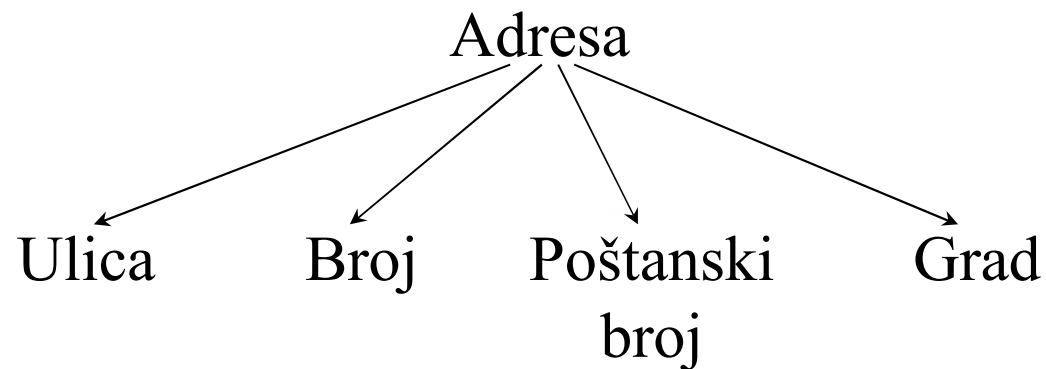
Relacioni model podataka



- Relacioni model podataka predstavlja teorijsku osnovu za baze podataka relacionog tipa
- 1970. prvi put su dati principi i struktura relacionog modela podataka (*E.F.Codd*)
- Prve realizacije na računaru su bile spore (neefikasne)
- Razmatraju se sledeće komponente relacionog modela podataka:
 - Strukturna komponenta – predstava podataka
 - Integritetska komponenta – zaštita podataka
 - Manipulativna komponenta – manipulisanje podacima



- Imenovana vrsta svojstva (osobina entiteta)
- Prost atribut (*simple*) - ne može se rastavljati na delove bez gubitka svakog značenja – atomska vrednost.
- Složen atribut (*composite*) se sastoji od više prostih atributa. Može se rastaviti na jednostavnije.



- Pri projektovanju IS, treba pažljivo birati attribute, u skladu sa potrebama
- Primer:
STUDENT (BrInd, Ime, Prezime, Adresa, Telefon, DatRodjenja, ...)
- **DatumRodjenja** – sa namerom posedovanja podatka o starosti – dobar izbor atributa (informacija se može izračunati)
- **GodineStarosti** – loš izbor atributa – zahteva svakodnevno ažuriranje BP



- Skup svih mogućih vrednosti nekog atributa A_i naziva se domenom tog atributa i označava se sa D_i ili $Dom(A_i)$
- Domen - tip podataka u programiranju
- Jedan domen za više atributa. Obrnuto ne.
- Primeri:
 - Atribut: Visina (cm)
 D_1 : skup celih pozitivnih brojeva
 - Atribut: NazivKnjige
 D_2 : skup svih različitih naslova knjiga
 - Atribut: Boja
 D_3 : {"žut", "crven", "zelen", "plav"}



- Najčešće korišćeni domeni:
 - Skup celih brojeva
 - Skup decimalnih brojeva
 - String – niz alfanumeričkih karaktera
 - Datum
 - Set novčanih vrednosti
 - Logički podatak
- Predefinisani domeni
- Korisnički domeni



- Šema relacije R je konačan skup atributa $\{A_i\}$ i konačan skup ograničenja $\{O\}$ nad vrednostima tih atributa.
- Ograničenja: atributi ne mogu uzimati bilo koje vrednosti
 - Podrazumeva se da kada su zadati atributi, zadati su i njihovi domeni
- Bitne osobine šeme relacije:
 - Nazivi atributa moraju biti različiti - unikatnost
 - Redosled atributa nije bitan
 - Šema relacije mora da sadrži bar jedan atribut



- Šema relacije se zapisuje:
 $R(A_1, A_2, \dots, A_N)$,
 A_i - atributi
 N – broj atributa
- Primer: Šema relacije koja predstavlja jednu klasu (klasa studenata):
STUDENT (BrInd, Ime, Prezime, Adresa)
- Šemom relacije se predstavljaju svojstva klase **objekata** ili **veza** nekog sistema
- Šema relacije može da se tumači i kao **definicija strukture** jedne tabele

- Šema relacije je pojam relacionog modela koji služi za predstavljanje neke klase
- Za tako definisanu klasu instance (slogovi, rekordi, n-torke, zapisi) koje ona sadrži predstavljaju se pojmom **relacije**
- Opisna definicija relacije:
Relacija **r** nad šemom relacije **R** je konačan skup n-torki vrednosti atributa (sadržaj tabele)

Formalna definicija relacije:

- Neka je data šema relacije $R(A_1, A_2, \dots, A_N)$ i neka su $D_i = \text{Dom}(A_i)$ odgovarajući domeni atributa.
- Relacija r nad šemom relacije \mathbf{R} je podskup Dekartovog proizvoda domena atributa, odnosno:
$$r \subseteq D_1 \times D_2 \times \dots \times D_N .$$
- Bitne osobine relacije:
 - n -torke moraju biti unikatne; u relaciji ne mogu postojati dve iste n -torke (n -torke sa istom vrednosti svih odgovarajućih atributa)
 - redosled n -torki nije bitan

- Relaciji u praksi odgovara jedna tabela
- Svakoj n-torki odgovara jedan slog (zapis, red) te tabele
- Slogovi u datoteci su zapisani u određenom redosledu, najčešće po redosledu unošenja
- Primer: STUDENT (BrInd,Ime)

student (BrInd Ime) → šema relacije

123/08	J.Jankovic
11/08	P.Petrovic
151/09	J.Jovanovic
III-15/10	M.Markovic

} relacija

Primer iz matematike:

- Relacija definisana na N skupova je podskup Dekartovog proizvoda tih skupova
- Podskup sadrži one n -torke Dekartovog proizvoda koje zadovoljavaju zadatu relaciju
- Primer: Neka su zadati skupovi $A=\{1,2,3,4\}$ i $B=\{4,6,8\}$.

Nad ovim skupovima je definisana relacija $r: A \times B = \{(a,b) \mid a=b/2\}$.

Sledi da je:

$$r = (A, B)$$

2	4
3	6
4	8



- Redosled elemenata u jednoj n-torki je bitan, a redosled n-torki je proizvoljan
- $r \subseteq D_1 \times D_2 \times \dots \times D_N$.
- Skupovi D_1, D_2, \dots, D_N nazivaju se domenima relacije r
- Osobine relacije:
 - Broj domena određuje **stepen relacije** (unarna, binarna, ..., n-arna)
 - Broj n-torki u relaciji naziva se **kardinalnost**



- Primer: Neka su definisani atributi i domeni:
 $BrInd = \{1/09, 2/09, III-56/08, 32/08, 33/07\}$
 $Ime = \{Milan, Zoran, Petar, Dragan\}$
 $Fakultet = \{PFB, FTTHM, FIR\}$
- Može se zadati relacija:
Student \subseteq
 $BrInd \times Ime \times Fakultet$
- Stanje relacije u jednom trenutku može biti:

BrInd	Ime	Fakultet
1/09	Milan	FIR
2/09	Petar	PFB
2/09	Dragan	PFB
III-56/08	Dragan	FIR
32/08	Milan	FTTHM
33/07	Zoran	PFB

Terminologija - RDB	Terminologija - tabele
Relaciona DB	Skup tabela
Relacija	Pojedinačna tabela
Atribut	Naziv kolone u tabeli
Instanca (n-torka)	Red podataka u tabeli
Kardinalnost relacije	Broj redova u tabeli
Stepen relacije	Broj kolona u tabeli
Domen	Skup dozvoljenih vrednosti za podatke u kolonama

- Vrednost NULL – univerzalnog tipa
 - primenjiva za attribute bilo kakvih domena
- U praksi postoje situacije kada u relacije unosimo n-torke za koje su vrednosti nekih atributa nepoznate u tom trenutku. Postoje dva slučaja:
 1. Vrednost postoji, ali nije poznata u trenutku unosa n-torke. Npr. kod upisa studenata unosimo sve podatke izuzev podataka o telefonu, koji se može i naknadno uneti.
STUDENT (BrInd,Ime,Prezime,Telefon)
 - **Posledica trenutnog nepoznavanja vrednosti nekog atributa**, i naknadno se može uneti

2. Ta vrednost je nedefinisana, nema smisla.
Npr. Ako je zadata šema relacije:
STUDENT (BrInd,Ime,Prezime,Telefon,Smer) Ne
može se uneti smer za studente 1. godine, jer se
naknadno opredeljuju za njega.

 - Navedeni problem je **posledica odabrane strukture šeme relacije**



- Pri projektovanju IS pažljivo birati šeme relacija – da bi se izbegao unos prevelikog broja vrednosti NULL (racionalnost)
- Primer: Šema relacije Službenik i uvođenje atributa medalja
Službenik (JMBG,Ime,Medalja)
- Za većinu službenika na tom mestu bi ostalo prazno mesto sa vrednošću NULL
- Rešenje problema: Podela na dve šeme relacija
Službenik (JMBG,Ime)
Odlikovanja (JMBG,Medalja)
- Nova relacija bi bila potpuno popunjena, a preko atributa JMBG bila bi “povezana” sa šemom relacije
Službenik



- Osnovni pojmovi relacionog modela podataka: atribut, domen, šema relacije i relacija
- Šema relacione BP – prvi izvedeni pojam relacionog modela podataka
- **Šema relacione BP** je konačan skup šema relacija $\{R_i\}$ i konačan skup $\{U_i\}$ ograničenja koja važe između njih
- Skup ograničenja $\{U_i\}$ uključuje samo ograničenja koja važe između pojedinih šema relacija. Ograničenja O_i su uključena kroz relacije (ograničenja nad atributima)

- Relaciona BP je drugi izvedeni pojam u okviru relacionog modela podataka.
- **Definicija:** Relaciona baza podataka BP je konačan skup relacija $\{r_i\}$ nad šemom relacione BP $\{R_i\}$
- Relaciona BP je skup svih tabela (sadržaja) jedne BP



- Relaciona BP je konačan skup relacija
 - Svaka relacija ima svoju šemu
 - Svaka relacija ima svoje instance (n-torke)
tj. telo
- Nazivi šema relacija moraju biti različiti
- U jednoj šemi relacije imena atributa moraju biti različita



- Služi za predstavljanje ograničenja koja važe nad podacima, odnosno nad vrednostima pojedinih atributa.
- Ta ograničenja se po prirodi mogu podeliti u tri grupe:
 - **Identifikacioni (egzistencijalni) integritet**
 - **Ograničenja vrednosti atributa**
 - **Referencijalni integritet**



1. Ograničenja koja proizilaze iz zahteva unikatnosti n -torki u relacijama. Nazivaju se **identifikacionim ili egzistencijalnim integritetom**.
 - U jednoj relaciji ne mogu da postoje dve iste n -torke



2. Ograničenja koja se eksplicitno zadaju preko skupova ograničenja O_i u okviru šema relacija R_i .
 - Takva ograničenja su posledica ograničenja koja važe nad svojstvima atributa u realnom svetu
 - *column-integrity rule*



3. Ograničenja koja uključuju attribute koji se nalaze u različitim relacijama i koja se zadaju preko skupa ograničenja U u okviru šeme relacione BP
 - Takva ograničenja se nazivaju **referencijalnim integritetom**



- Prema definiciji relacije, u njoj se nikada ne mogu javiti dve identične n-torke
- Preciznije: svakoj vrednosti skupa svih atributa A_1, A_2, \dots, A_N šeme relacije $R = \{A_i\}$, odgovara jedna n-torka u relaciji r , odnosno skup vrednosti a_1, a_2, \dots, a_N – jedinstvenost
- Pošto su sve n-torke u jednoj R različite
 - **super ključ uvek postoji**
- Može se uočiti da i pojedini podskupovi atributa ili njihovih kombinacija jedinstveno određuju samo jednu n-torku u relaciji



- **Definicija:** Super ključ neke šeme relacije R je svaki podskup atributa X te šeme koji ima osobinu da jednoznačno određuje n-torke u relaciji.
- Podseća na pojam funkcije $y=f(x)$.
- Stoga se kaže da R funkcijski zavisi od X ili da X funkcijski uslovljava R, ili formalno

$$X \rightarrow R$$

- Šema relacije može da ima više super ključeva. Skup svih super ključeva SK se definiše sa:

$$SK = \{X_i \mid X_i \subseteq R \wedge X_i \rightarrow R\}$$

Kandidat ključ (*candidate key*)

- Pitanje: koji od super ključeva su minimalni, tako da njihovi sastavni delovi više nisu super-ključevi? (neredudantnost)
- Primer: u relaciji **ispit** super ključ može biti **BrInd, SifP, Ocena** ili **BrInd, SifP**
Pojedinačno ne mogu biti BrInd, SifP
- Zašto je bitan kandidat ključ?
Može da posluži za direktan pristup n-torki.
- Vrednosti kandidat-ključeva se zapisuju u pristupnoj (indeksnoj) datoteci – princip minimalnosti.



- **Definicija:** Kandidat ključ neke šeme relacije R je svaki podskup atributa X te šeme koji ima osobinu da jednoznačno određuje n-torke relacije, a da ni jedan njegov pravi podskup nema tu osobinu.
 - Kandidat ključ neke šeme relacije R je svaki njen super-ključ koji ima osobinu minimalnosti.
 - Svaka šema relacije ima barem jedan kandidat ključ.
- Formalno, matematički, skup svih kandidat ključeva se može definisati kao:

$$KK = \{X_i \mid X_i \subseteq R \wedge X_i \rightarrow R \wedge \neg \exists Y (Y \subset X_i \wedge Y \rightarrow R)\}$$

Primarni ključ (*primary key*)

- Ako neka šema relacije ima više KK, za potrebe direktnog pristupa bira se jedan od njih
- **Definicija:** Primarni ključ neke šeme relacije R je jedan izabrani kandidat ključ ili jedini kandidat ključ ako nema izbora.
 - **Svaka šema relacije ima jedan i samo jedan primarni ključ.**
- Formalna definicija primarnog ključa je $PK \in KK$
- Atributi koji čine primarni ključ zovu se primarni atributi

- Ako neka R ima više kandidat ključeva, izbor primarnog ključa se vrši po kriterijumima:
 - Prednost imaju ključevi “po prirodi stvari” – posledica su nekog prirodnog “suštinskog” ograničenja. Suštinsko ograničenje se nikada ne može promeniti.
 - Ako po prethodnom kriterijumu ima više ravnopravnih, bira se onaj koji zauzima manje memorijskog prostora
- Ukoliko u jednoj relaciji ne možemo definisati prirodan primarni ključ, uvodi se šifra, ID itd.
 - Najjednostavnije rešenje je redni broj unosa

Spoljašnji ključ (foreign key)

- Dosadašnji ključevi su definisani unutar jedne R. Za spoljašnji ključ neophodne su dve R
- Spoljašnji ključ u šemi relacije R je svaki njen podskup atributa za koji važi ograničenje vrednosti u relaciji **r** na sledeće dve vrednosti:
 - Vrednost primarnog ključa u nekoj relaciji (tzv. ciljnoj relaciji)
 - Vrednost NULL

- Primer:

– NASLOV(SIFN, NAZIV, SIFO)

OBLAST(SIFO, NAZIV)

spoljašnji ključ



- Student (BrInd, Ime)
 - Knjiga(SifK, *SifN*)
 - Naslov(SifN, Naziv, *SifO*)
 - Oblast(SifO, Naziv)
 - Autor(SifA, Ime)
 - Pozajmica(SifP, *BrInd*, Dana)
 - Rezervacija(SifN, *BrInd*, Datum)
 - Drzi(SifK, *BrInd*, Datum)
 - Je_autor(SifA, *SifN*, Koji)
-
- Jedna šema relacije može da sadrži više spoljašnjih ključeva
 - Spoljašnji ključ može biti u sastavu primarnog ključa
 - Spoljašnji ključ može istovremeno biti i primarni ključ u celini

Primer 1:

- Radnik(SifR, Ime, Adresa, SifNad)
- Pretpostavka: svaki radnik ima samo jednog nadređenog
- Sve se dešava unutar jedne šeme relacije koja sadrži oba učesnika u vezi “spoljašnjii ključ – primarni ključ”

Primer 2:

Osoba(JMBG, Ime, Adresa,...)

Brak(JMBG1, JMBG2, Datum_venčanja)



Identifikacioni integritet

- Identifikacioni integritet proizilazi iz osobina unikatnosti n -torki u relaciji i svodi se na formulaciju odgovarajućeg uslova
- Uslov identifikacionog integriteta:
Ni jedan atribut šeme relacije R koji je u sastavu primarnog ključa nikada ne sme imati NULL vrednost u relaciji r .
- Ukoliko ovo ne bi bilo ispunjeno, može nastupiti situacija da dve ili više n -torki u relaciji postanu identične

- Primer:

Ispit (BrInd SifP Ocena)

.....

123 01 8

123 02 8

- Ako bi se dozvolilo da npr. SifP uzme vrednost NULL

Ispit (BrInd SifP Ocena)

.....

123 NULL 8

123 NULL 8

- Dobile bi se dve identične n-torke što je nedozvoljeno u relacionom modelu.

- Suština referencijalnog integriteta je u ograničavanju vrednosti spoljašnjeg ključa.
- *Uslov referencijalnog integriteta:*
Svaki podskup atributa šeme relacije R koji predstavlja spoljašnji ključ može u relaciji r imati:
 - ili vrednost primarnog ključa u ciljnoj relaciji
 - ili vrednost NULL.



- **Sa stanovišta izmena (ažuriranja) u relaciji koja sadrži spoljašnji ključ** to podrazumeva da važe sledeća ograničenja:
 - Ne može se uneti n-torka sa vrednošću spoljašnjeg ključa koja nije jednaka nekoj vrednosti primarnog ključa u ciljnoj relaciji ili NULL vrednosti
 - Ne može se izmeniti n-torka tako da vrednost spoljašnjeg ključa ne bude jednaka nekoj vrednosti primarnog ključa u ciljnoj relaciji ili NULL vrednosti

- **Sa stanovišta izmena (ažuriranja) u ciljnoj relaciji:**
 - Dodavanje nove n-torke (u ciljnoj relaciji) ne narušava ref. int. - nastaje samo nova vrednost primarnog ključa – novi zapis
 - Uklanjanjem n-torke (a izmena ponekad) dovodi do nestanka jedne vrednosti primarnog ključa. Ako bi se ta operacija izvršavala bezuslovno to bi moglo da naruši referencijalni integritet



- Primer:
Za šemu relacije Radnik:
Radnik(SifR, Ime, Adresa, SifNad)
 - Postojala bi sledeća specifikacija:
 $\text{Radnik[SifNad]} \subseteq \text{Radnik[SifR]} \cup \text{NULL}$



Dinamička specifikacija referenc. integriteta

- Neka je r ciljna relacija nad R sa primarnim ključem X , i neka je s relacija nad S koja sadrži spoljašnji ključ Y
- Tada se za sledeće operacije u r :
 - DELETE: uklanjanje n -torke,
 - UPDATE: ažuriranje vrednosti primarnog K , navodi jedna od sledeće tri akcije za relaciju s

RESTRICTED, CASCADES, NULLS

Dinamička specifikacija referenc. integriteta

- **RESTRICTED:** operacija se ne izvršava ako u relaciji **s** postoji vrednost spoljašnjeg ključa koja odgovara vrednosti primarnog ključa u **r**
- **CASCADES:** operacija se uvek izvršava. Ako je uklonjena **n**-torka iz ciljne relacije **r**, u **s** se uklanjaju sve **n**-torke sa datim ključem. Ako je došlo do izmene, menjaju se sve vrednosti **n**-torki u relaciji **s** sa novim spoljašnjim ključem
- **NULLS:** operacija se uvek izvršava. U **s** se u svim **n**-torkama koje imaju dati promenjeni spoljašnji ključ, menja njegova vrednost u **NULL**.
 - **NULLS** ne može ako je spoljašnji ključ u sastavu primarnog ključa!!!

Dinamička specifikacija referenc. integriteta

- Dinamička specifikacija ref.int. se navodi za svaki spoljašnji ključ sveke šeme relacije posebno
- Opšta forma takve jedne specifikacije je:
operacija OF *CiljnaSema* **akcija** *SpoljašnjiKljuč*
- Operacije: DELETE, UPDATE
Akcije: RESTRICTED, CASCADES, NULLS
- Naznaka *SpoljašnjiKljuč* se može izostaviti
 - Referencijalni integritet se može zadati za svaki pojedinačni spoljašnji ključ
 - Referencijalni integritet se može zadati za celu šemu relacije (obuhvata sve spoljašnje ključeve)



Dinamička specifikacija referenc. integriteta

- Primer:
Par šema relacija za mrežu železničkih pruga
Stanica(SifS, Naziv)
Relacija(SifSPoc, SifSZad, Km)
- Za šemu relacije **Relacija**, za oba spoljašnja ključa važe iste dinamičke specifikacije referencijalnog integriteta:
DELETE OF Stanica RESTRICTED
UPDATE OF Stanica CASCADES
- Ako se ništa ne navede za akciju se podrazumeva najoštrija - RESTRICTED



Pitanja ?

