

1. Kada je relacija u trećoj normalnoj formi?

Rezolucija se nalazi u trećoj normalnoj formi, ako i samo ako neključni atributi nisu ni funkcionalno ni tranzitivno zavisni od ključnog atributa, što znači da se svaka relacija koja se nalazi u 3NF nalazi se i u drugoj dok obrnuto ne važi.

PRIMER:

LABORATORIJA < brsobe#, telefon >

LABORANT < sifralab, ime, prezime, brsobe >

2. Operatori pogodni za ažuriranje?

Operatori pogodni za ažuriranje su unija, presek, razlika, proizvod i oni se izvode na bar dve relacije.

Unija dva skupa, dve relacije (tabele) A i B, je tabela koja se sastoji od svih elemenata koji pripadaju tabelama A i B. Da bi unija bila ostvariva dve relacije moraju biti union-kompatibilne tj.

1. Obe relacije moraju imati iste atribute

2. isti atributi moraju biti definisani nad istim domenom.

Presek dve relacije A i B je nova relacija koja sadrži sve n-torke koje su zajedničke za obe relacije.

Razlika A-B dve kompatibilne relacije je nova relacija koja ima iste atribute kao relacije A i B, a telo se sastoji samo od onih n-torki koje se nalaze u relaciji A a ne nalaze u relaciji B.

Proizvod: Kartezijev proizvod 2 skupa A i B definiše se kao skup uređenih parova u kojem prvi element pripada skupu A, a drugi skupu B

U relacionoj algebri zelimo da dobijemo uređen skup n-torki a ne skup parova, onda se definicija Kartezijevog skupa proširuje tako što se skup elemenata zamenjuje skupom n-torki, pri tom je svaka tako novodobijena n-torka rezultat spajanja uređenog para n-torki.

3. Prevođenje veza drugog reda (binarnih veza) na relacioni model

Postupak prevodjenja binarnih veza na relacioni oblik posmatramo na tri primera:

Binarne veze 1:1 prevode se na relacioni jezik tako što se u jednu tabelu koja učestvuje u vezi, uvrsti primarni ključ druge tabele kao atribut. Veza se prema tome iskazuje spoljnjim ključem.

Binarne veze 1:N prevode se na relacionu formu slično kao i veze tipa 1:1. Veza se ponovo iskazuje spoljnjim, stranim ključem ali ne u bilo kojoj od relacija, nego u onoj koja u vezi E-R modela predstavlja objekat tipa povezanosti mnogo.

Binarne veze M:N se prevode na relacionu formu obavezno uvođenjem nove relacije (vezne relacije) u bazu podataka. Ključ te relacije po pravilu je složen i sastoji se od primarnih ključeva objekata koji učestvuju u vezi.

4.Upotreba NULL vrednosti

Null vrednosti su jedno od najvećih novina koje su donele relacione baze podataka, to je mogućnost prikazivanja nepostojećeg podatka čija je vrednost nedefinisana. Primenjuje se:

1) kada u bazi postoje atributi za koje su NULL vrednosti normalne jer to svojstvo nije primenljivo na sve primerke nekog entiteta

2) ako vrednost nekog atributa za neke objekte trenutno nije poznata ili nije dozvoljena (nema broj telefona ili nije raspoređen u neko odeljenje)

3) kada nije nastupio momenat delovanja nekog atributa

5.Navesti i objasniti najmanje 4 Kodova pravila

Osnovno tj. nulto pravilo koje predstavlja uslov bez koga se ne može glasi:

“Svaki sistem za upravljanje bazama podataka, koji se smatra, ili koji jeste relacioni, mora imati mogućnost upravljanja bazom podataka na relacioni način i relacionim metodama.”

1.PRAVILO

Predstavljanje informacije: Sve informacije u relacionoj bazi moraju logički biti predstavljene na isti način-vrednostima podataka u tabelama (relacijama).

2.PRAVILO

Logička dostupnost podacima: svaki podatak mora imati tzv. “atomarnu”

Vrednost koja logički mora biti dostupna korisniku uz pomoć:

- imena relacije u kojoj se nalazi
- vrednosti primarnog ključa
- imena atributa u toj relaciji

3.PRAVILO

Mogućnost prikazivanja nepostojećih informacije:

Relaciona baza podataka mora podržavati koncept nultog podatka tj. nulte vrednosti (NULL vrednosti). To je vrednost nekog atributa koja trenutno nije poznata.

4.PRAVILO

Dinamički katalog:

Relaciona baza podataka mora biti tako predstavljena da autorizovani korisnici mogu primeniti neki od relacionih jezika na sve podatke u njoj. Pod ovim se podrazumevaju sistemski katalozi, relacije koje su u toku rada za stalno, ili na određeni vremenski period programski kreirane, a koje sadrže neke nove informacije.

6.Vrste ključeva i odgovor sistema:

Primarni ključ je atribut, ponekad skup kombinovanih atributa, čija vrednost jednoznačno određuje samo jednu n-torku, samo jedan slog u tabeli, što znači da jednoznačno izdvaja samo jedan red u njoj. U jednoj tabeli ne smeju postojati dve različite n-torke sa istom vrednošću primarnog ključa.

Ukoliko ne postoji jedan atribut koji zadovoljava uslove za ključni atribut, ključ se mora kreirati kombinacijom dva ili više atributa.

Primarni ključ mora biti: jedinstven (neponovljiv), nepromenljiv i raspoloživ.

Spoljnji ključ tj. Strani ključ je atribut koji u drugoj relaciji ima ulogu primarnog ključa. Uloga spoljašnjih ključeva je prvenstveno u uspostavljanju veza između tabela.

7. Tip povezanosti objekata i grafički simboli koji se koriste

Generalno gledano postoje dva tipa povezanosti i to:

- Tip jedan(1)
- Tip mnogo(N)

Tako da se njihovim kombinovanjem mogu dobiti veze tipa:

- 1:1 (jedan prema jedan)
- 1:N (jedan prema više)
- M:N (više prema više)



8. Objasniti upotrebu tabela indeksa i pojam indeksiranja

Jedna od osnovnih operacija pri manipulisanju velikim brojem podataka je predstavljanje tj. Dobijanje nove informacije na osnovu određenog kriterijuma, tj. Broja prethodno poznatih vrednosti nekog atributa. Ako je baza ogromna i upiti logično složeni, vreme za dobijanje rezultata je nedopustivo dugo, iz tog razloga pribegava se indeksiranju atributa po atributu (ili) atributima po kojima se vrši indeksiranje. Sve tabele su po pravilu indeksirane po primarnom ključu i tada indeksna tabela (uvek je izvedena od osnovne) ima samo dva atributa koje definišu drugačije, po nekoj zakonitosti, složene slogove ili n-torke.

Svaka indeksna tabela ima 2 dela:

1. mesta atributa po kojima se vrši pretraživanje (po kojima se vrši uređivanje indeksa)
2. indeks koji služi kao veza sa osnovnom tabelom

9. navesti naredbe za definisanje podataka u SQL-u i ukratko ih objasniti

CREATE TABLE-kreiranje fizičke tabele baze podataka

ALTER TABLE-menja postojeće definicije tabele, izmena, dodavanje ili uklanjanje kolone (atribute)

DROP TABLE-uklanjanje tabele iz baze podataka

CREATE VIEW-kreiranje virtuelne imenovane tabele, "pogled"

CREATE INDEX-kreira indeks nad jednom ili više tabela, kolona

DROP VIEW-uklanja pogled iz baze podataka

10. Opšti oblik select upita, redosled klauzula i funkcije svake klauzule

SELECT [ALL DISTINCT] lista atributa 1

FROM lista tabela (relacija)

[WHERE lista uslova 1]

[GROUP BY lista atributa 2]

[HAVING lista uslova 2]

[ORDER BY lista atributa 3]

UNION [ALL]

U SELECT-u se definišu (selektuju atributi) čije vrednosti želimo dobiti (odgovara operatoru projekcije).

FROM-izdvaja relacije u kojima se nalaze vrednosti tih atributa (odgovara operatoru spajanja)

(WHERE, HAVING)-definišu uslove koje podaci treba da zadovoljavaju pri izdvajanju, što odgovara operatoru restrikcije (selekcije)

GROUP BY-grupisu podatke po nekom kriterijumu

ORDER BY-sortiraju podatke po rastućem (ASC-ascending) ili opadajućem (DESC-descending) redosledu.

Odredbe SELECT I FROM su obavezne.

11. Sta su to vezni objekti i koje osobine ih karakterisu?

Vezni objekti ili veze iskazuju način međusobne povezanosti objekata, a rezultat su najčešće zakonskih propisa, dogovora, ugovora, statuta... Veze po prirodi mogu biti I objekti (entiteti), I onda moraju imati svoj identifikator, a mogu, ali ne moraju imati I svoje atribute.

Svaka veza se karakterise sa 3 osobine:

1. red veze
2. način uspostavljanja veze
3. tip veze

Red veze određuje broj objekata koji čine vezu. Najčešće se javljaju

- unarne veze
- binarne veze
- trojne veze

a moguće je i da se javljaju veze višeg reda tzv. n-tarne veze koje se zbog nepreglednosti E-R modela u praksi izbegavaju kad god je to moguće.

Unarne veze su relativno retke, a uspostavljaju se unutar jedne tabele, jednog objekta.

PRIMER:

RADNIK<IDBR#, IME, POSAO, KVALIF, RUKOVODILAC....>

Gde su IDBR radnika i rukovodilac u unarnoj vezi.

Binarna veza je veza između 2 različita objekta, i najčešće se susreće u praksi.

Trojna veza nastaje obično daljim proširenjem binarne veze.

12. Nabrojati i kratko opisati aritmetičke funkcije?

POWER(BROJ, e)-stepen nekog broja

ROUND(broj, d)-zaokruživanje na d decimala

TRUNC(broj, d)-odseca na d decimala

ABS(broj)-absolutna vrednost

SIGN(broj)-znak broja (1 za >0, 0 za 0, -1 za broj <0)

SQRT(broj)-kvadratni koren broja

MOD(broj1, broj2)-izračunava broj1 moduo broj2

SQL naredbe mogu imati aritmetičke izraze sastavljene od imena kolona, imena funkcije i konstanti povezanih operatorima(+, *, -, /)

13. Navesti i objasniti operacije pogodne za izvestavanje

U ove operacije spadaju:

- selekcija
- projekcija
- spajanje
- deljenje

Selekcija (ogranicenje, restrikcija) izdvaja iz relacije samo neke n-torke koje zadovoljavaju zadani uslov, koji je logički definisan. N-torke kojima je taj logički uslov zadovoljen definiše novi relaciju. Upit kojem se vrši selekcija mora uvek biti logički i izvodljiv, u suprotnom se selekcije ne može prevesti.

Projekcija neke relacije daje novu relaciju koja se sastoji samo od određenih (samo jednog) atributa zadane relacije. Rezultat operacije projekcije je podskup izabranih atributa zadane relacije sa svim njenim n-torkama, ali mora paziti po kom atributu se vrši projekcija posto zbog nesmotrenosti može doći do gubitka podataka.

Spajanje

Operacija spajanja ima više vrsta od kojih su dve najvažnije:

1. prirodno spajanje
2. spajanje pod nekim uslovom

Prirodno spajanje relacije (A i B) daje relaciju koja ima attribute relacije A, i one attribute relacije B koje nema relacija A.

A < X1.....X20 >

B < X10.....X30 >

AB < X1.....X30 >

Ako relacije koje se spajaju nemaju ni jedan zajednički atribut onda operacija prirodnog spajanja prelazi u običan **Kartezijev** proizvod.

Spajanje pod nekim uslovom izvodi se nad relacijama samo onda kada one nemaju nijedan isti atribut. Rezultat je običan Kartezijev proizvod tih relacija koji sadrži samo one n-torke koje zadovoljavaju logički uslov definisan logičkim izrazom.

Deljenje se ne može izvesti uvek i sa proizvoljnim tabelama.

Da bi operacija A podeljeno sa B (A/B) bila izvodljiva, potrebno je da se svi atributi relacije B nalaze u relaciji A.

A		B	D
X	Y	Y	X
012	A1	A3	032
022	A2		042
032	A3		062
042	A3		
052	A2		
062	A3		
072	A2		

14.Sta je entitet i sta po svojoj prirodi moze biti?

Entitet ili objekat je deo realnog sveta opisan ogranicenim brojem atributa.

Po svojoj prirodi moze biti:

- deo okruzenja(clan kolektiva,aparat...)
- apstraktni pojam(neka mera,necije zvanje,boja...)
- dogadjaj(saobracajni udes,upis studenata)
- asocijacija(predmet-nastavnik,polaznik-kurs...)

15.Prikupljanje informacija i koji su izvori?

Informacije se prikupljaju iz vise izvora,a pre svega od:

1. korisnika
2. postojece dokumentacije
3. spoljnjih izvora

Postupak prikupljanja informacija izvodi se:

- prvo,u fazi analize postojeceg informacionog sistema
- drugo,u fazi utvrdjivanja ciljeva koji se zele postici

Izvori informacija:

Osnovni izvor informacija su korisnici sistema.Informacije se od korisnika prikupljaju pomocu intervjua,upitnika I posmatranjem.

Intervjui se koriste za sakupljanje podataka u direktnom razgovoru sa korisnicima,postavljenjem odredjenih pitanja od strane analiticara.

Intervju se moze obavljati grupno ili pojednacno.

Upitnik(anketa) je tehnika koja omogucuje da analiticar kontaktira veliki broj ljudi I da uporedi njihove odgovore na isto pitanje.Anketa moze biti anonimna,sto ima svojih prednosti,ali zato ima I svoje mane posto upitnik obicno popuni 30-40% ljudi kojima je upitnik upucen.u upitniku se obicno postavljaju pitanja sa ponudjenim odgovorima,ali I oni koji zahtevaju opsirne odgovore.

16.Prednosti i mane pravljenja aplikativnog programa

Prednosti:

Korisnik ne mora da poznaje upitni jezik

Korisnik ne mora da poznaje konfiguraciju tabele

Mogucnost slucajnih gresaka je svedena na minimum

Bezbednost podataka je povecana

Olaksan je prenos programa(korisnik ne mora da vodi racuna o kompatibilnosti)

Mane:

Korisnik moze da uradi samo ono sto je unapred predvidjeno tj.sto mu aplikacija dozvoli.

17.Domen

Domen je skup vrednosti koje atribut moze da poprими.

18.Integritet entiteta

Nijedan atribut koji je primarni ključ ili deo primarnog ključa neke bazne relacije ne moze da uzme null vrednost.

19.Referencijalni integritet

Skup vrednosti spoljnjeg ključa relacije R1 mora biti podskup skupa vrednosti primarnog ključa relacije R2 sa kojom se povezuje relacija R1.Relacije R1 I R2 ne moraju biti različite.

20.Sta je to distribuirana baza podataka?

Pod distribuiranom bazom podataka smatra se danas svaki informacioni sistem koji radi u racunarskoj mrezi,I koji ima najmanje dva dislocirana racunara,sa dislociranim tabelama u kojima se nalaze podaci.

21.Objasniti upotrebu WHERE klauzule

Where klauzula nam omogućuje:

- izdvajanje(selekciju) redova koji zadovoljavaju neki uslov
- izdvajanje redova koji zadovoljavaju vise uslova(AND)
- izdvajanje redova koji zadovoljavaju bar jedan od uslova(OR)
- izdvajanje redova koji zadovoljavaju slozene uslove(AND I OR)
- izdvajanje redova cija je vrednost unutar nekih granica(BETWEEN)
- izdvajanje redova cija vrednost pripada nekoj listi vrednosti(IN)
- izdvajanje redova koji ne zadovoljavaju neke uslove(NOT,IS NOT)
- izdvajanje redova ako neka vrednost postoji(EXISTS)...

Sintaksa pisanja uslova (WHERE) pomocu izraza uporedjivanja,u opstem slucaju je sledeca:

argument1 operator uporedjivanja argument2

a argumenti mogu biti

- jedan atribut
- skup atributa(u zagradama odvojeni zapetom)
- naredba SELECT

koja vraca najvise jednu n-torku,dok operator uporedjivanja moze biti bilo koji od operatora:

- veci(>)
- manji(<)
- veci ili jednak (>=)
- manji ili jednak (<=)
- jednak(=)
- razlicit(<>)

prilikom postavljanja uslova mogu se koristiti sledece opcije:

1. BETWEEN
2. IN
3. EXISTS

22.Prevođenje E-R modela na relacioni oblik

Tehnika prevodjenja E-R modela na relacioni model izvodi se na taj nacin sto:

- Svaki **objekat** iz E-R modela postaje **relacija**
- Svaka **veza** moze,ali ne mora,da postane **vezna relacija**
- **Ime objekta** postaje **ime relacije**
- **Karakteristike objekta postaju njegovi atributi**
- **Identifikatori objekata postaju ključevi relacija**

23. Prevođenje unarnih veza na relacioni oblik

Predstavimo na tri primera:

unarne veza tipa 1:1

unarne veze postoje de facto medju n-torkama jedne tabele a prevode se na relacioni oblik uvodjenjem sifre jedne n-torke(koja ucestvuje u vezi) kao spoljnjeg ključa u drugu,koja je u vezi sa njom.

unarne veza tipa 1:N prevode se na relacioni oblik identicno kao I uslucaju 1:1 uvodjenjem spoljnjeg,preimenovanog ključa.

unarne veza tipa M:N prevodi se na relacionu formu uvodjenjem nove vezne tabele,vezne relacije,ciji je ključ slozen-sastavljen od dva atributa.

24. Sta je to normalizacija?

Normalizacija je ustvari postupak izmene(najcesce dekompozicije,to jest rastavljanja na vise relacija)prvokoncipiranih tabela(relacija)s ciljem postizanja dobrih osobina modela.

25. Sta je to redudanca podataka?

Ponavljanje podataka(istih informacija) u jednoj ili vise tabela.

26. Kada se relacija nalazi u 1NF?

Relacija se nalazi u prvoj normalnoj formi(1NF) ako,I samo ako,je njen domen(podaci u tabeli) skup atomarnih podataka.Buduci da je ovo uslov da bi neka tabela uopste bila relacija,sledi da se svaka tabela nalazi u prvoj normalnoj formi.

27. Kada se relacija nalazi u drugoj normalnoj formi(2NF)?

Relacija se nalazi u drugoj normalnoj formi ako svaki atribut koji nije ključni zavisi potpuno(a ne delimicno)od ključnog atributa.

PRIMER:

R21 <sifart#,kvalitet,cena>

R22 <sifkup#,sifart#,kolicina>

28. Problem gubitka informacija

Relacija se dekomponuje bez opasnosti gubitaka funkcionalnih zavisnosti samo ako se dekompozicija vrsi prema funkcionalnoj zavisnosti koja ne ide od kandidata ključa(kao sto je uradjeno sa relacijama koje povezuju laboratorije I laborante).

29. Faze projektovanja informacionog sistema?

- Definisane izvora informacija
- Izbor metoda I projektovanja sistema
- Izbor modela sistema
- Izbor nacina izrade dokumentacije
- Izbor nacina odrzavanja dokumantacije
- Definisane nacina vodjenja projekta

- Vrednovanje kvaliteta projekta

30. Sta su to bazne(tabele)relacije, a sta izvedene relacije?

Bazne tabele se jos nazivaju I fizicke tabele, jer one postoje na memorijskom medijumu racunara-disku, a **izvedene** relacije se mogu dobiti iz baznih relacija operacijama koje se definisu nad baznim tabelama.

31. navesti naredbe za rukovanje podacima u SQL-u i ukratko ih objasniti

- SELECT –pristup podacima
- INSERT-unosenje podataka, dodavanje redova u tabelu
- DELETE-brisanje podataka, izbacivanje redova iz tabele
- UPDATE-ažuriranje, izmena vrednosti podataka u koloni

32. Spajanje dve ili vise relacija JOIN: koristi se za prikazivanje podataka iz vise tabela koristeći zajedničke atribute.

33. Spajanje po jednakosti EQUIJOIN: koristi se za prikazivanje podataka iz vise tabela koristeći atribute sa istim imenom sa jednakim vrednostima.

34. Spajanje po nejednakosti THETA JOIN: koristi se za prikazivanje podataka iz vise tabela koristeći atribute koji su definisani nad istim domenima, ali se vrednosti povezuju po nejednakosti (<>, >, >=, <, <=).

35. Spoljnje spajanje OUTER JOIN: omogućuje da se u rezultat spajanja uključe i one n-torke koje ne zadovoljavaju uslov spajanja jer u nekoj od tabela nema ni jedne odgovarajuće vrednosti, ali zadovoljavaju uslov iz WHERE klauzule. U WHERE klauzuli naznaci se iz koje tabele se uzimaju vrednosti leve ili desne (LEFT ili RIGHT JOIN).

36. Agregatne funkcije: AVG-prosek vrednosti u koloni, COUNT, SUM, MIN, MAX...

37. Nivoi organizovanja podataka

Fizicka organizacija podataka

Logicka organizacija podataka

- Opsta logicka organizacija-globalna sema
- Programska logicka organizacija-programerska podrška
- Strukture podataka za dijalog krajnjeg korisnika za racunarom

38. Nabrojati i kratko objasniti funkcije za rad sa karakterima

Ime se završava slovom **a**

WHERE ime **LIKE** "*"a"

Trece slovo imena je **v**

WHERE ime **LIKE** "?? v*"

U imenu nema slovo **N**

WHERE ime **NOT LIKE** "*n*"

Ime je duzine od 6 slova

WHERE ime **LIKE** "??????"

Ime nije duzine od 5 slova

WHERE ime **NOT LIKE** "?????"

U imenu je slovo **I** ispred slova **N**

WHERE ime **LIKE** "*I*N*"

Broj ima dve cifre crticu cifru

WHERE tel **LIKE** "[0-9][0-9][0-9]"

Operator LIKE je pogodan za pretraživanje tekstualnih podataka. Znak * je takozvani dzoker, znak koji zamenjuje proizvoljan broj znakova. "M*" znaci da prvo slovo treba da bude M, a znak * znak zamenjuje proizvoljan niz karaktera, tako da ce u rezultatu upita biti prikazani svi zaposleni cije ime pocinje slovom M, bez obzira na broj slova u imenu.

39. Nabrojati i kratko objasniti sisteme za obradu podataka

Sistemi za obradu podataka se dele na dve vrste:

1. operacioni
2. informacioni

Operacioni sistem podržava organizaciju rada u racunarskom sistemu za direktnu obradu podataka kod koga su svi tipovi i nacini njihovog koriscenja potpuno definisani pri projektovanju baze podataka.

Osobine operacionih sistema su:

- programi obrade su napisani unapred
- podržavaju rad sa terminalima
- podržava rutinske (svakodnevne) obrade
- ažuriranje podataka u realnom vremenu
- pretraživanje podataka samo po primarnom ključu
- koriste se jezici za poslovnu obradu (Cobol, C)

Informacioni sistemi podatke koriste na nacine koji nisu unapred odredjeni. Sredstva za nalazjenje podataka traže se tek kad se pojavi korisnicki zahtev.

Osobine informacionih sistema

- programi obrade nisu napisani unapred jer su zahtevi spontani
- podržavaju hitne zahteve za nekim informacijama
- ažuriranje podataka vrši se posle nekog vremena (poludirektno on-line)
- pretraživanje podataka po primarnom i sekundarnim ključevima
- koriste se specijalni upitni jezici (SQL, QBE...)

40. Sta je datoteka?

Skup zapisa koji odgovaraju jednom prostoru objekata čini datoteku, I prema tome mora važiti sledeće:

- jednom prostoru objekata odgovara jedna datoteka
- broj zapisa u datoteci jednak je broju objekata u prostoru
- broj elementarnih obeležja objekta jednak je broju polja u zapisu

41. Pojam zapisa-SLOG?

Skup svih vrednosti elementarnih obeležja koja se odnosi na jedan objekat predstavlja zapis ili logički slog, I pri tome se svaki elementarni podatak (pojedinačna vrednost) smesta u jedno polje zapisa.

Atributi: brrad, ime, pol, odeljenje, radnistaz

Zapis: 51325 Ana Z prodaja 3god2mes17dan

42. Navedi modele podataka

Danas postoje četiri modela u upotrebi, ali najvažnija su dva poslednja

1. HIJERARHIJSKI model
2. MREZNI model
3. REALACIONI model
4. OBJEKTNI model

43. Navedi vrste objekata

Objekte u ovom smislu delimo na:

ČVRSTE objekte-objekti u punom smislu te reci (pod cvrstim objektom smatra se onaj entitet koji se može potpuno definisati jednim primarnim ključem i nizom atributa)

SLABE objekte-koji na neki način (egzistencijalno ili identifikacijski) zavise od jednog ili više cvrstih objekata.

44. Sta je kardinalnost veze?

Kardinalnost veze (preslikavanja) se odnosi na brojnost elemenata partitivnog skupa u koji se preslikava jedan element skupa originala. Za svako preslikavanje definiše se minimalna i maksimalna kardinalnost.

Minimalna i maksimalna kardinalnost jednog preslikavanja određuje se identifikacijom minimalne i maksimalne vrednosti podskupa u koji se može preslikati neki, bilo koji element skupa originala.

45. Prevođenje veza reda većeg od 2 na relacioni model

n-tarne veze mogu biti sledećih tipova:

- Povezanost svih objekata je tipa jedan
- Povezanost jednog objekta je tipa mnogo, a preostalih tipa jedan
- Povezanost dva objekta je tipa mnogo, a preostalih tipa jedan
- Povezanost svih objekata je tipa mnogo

N-tarne veze prevode se po pravilu na relacioni oblik uvek uvođenjem dodatnih veznih relacija, koje uključuju identifikatore (primarne ključeve) objekata koji tvore vezu.

46. Objasniti razliku u realizaciji i upotrebi ugnjezdenih upita i prirodnog spajanja

Naredba: SELECT * FROM A,B; daje Kartezijev proizvod relacija **A i B**. Ako relacija A ima 1000 n-torki i relacija B 1000 n-torki onda nova relacija dobijena spajanjem ovih dveju relacija ima 1000 * 1000, tj. 1 000 000 n-torki. Dakle nova relacija ima vrlo veliki broj zapisa. Sada sledi pretraživanje tog skupa i izdvajanje samo onih n-torki koje zadovoljavaju uslov iz klauzule WHERE. Ovo je vrlo dugotrajan postupak, za razliku od tehnike ugnjezdenih upita gde se najpre izvrši selekcija n-torki koje zadovoljavaju uslov, a tek onda tako redukovani skup se proverava u sledecem upitu. Iako se ugnjezdeni upiti uvek mogu realizovati preko operacije prirodnog spajanja to se ne preporučuje zbog drastičnog povećanja vremena obrade upita.

47. Nabrojati i ukratko objasniti upravljačke naredbe

- **GRANT**-dodela prava korišćenja tabele drugim korisnicima od strane vlasnika tabele
- **REVOKE**-oduzimanje prava korišćenja tabele drugim korisnicima
- **BEGIN TRANSAKCIJA**-početak transakcije koji se može završiti jednom od dveju navedenih naredbi
- **COMMIT WORK**-prenos dejstva transakcije na bazu podataka
- **ROLLBACK WORK**-poništanje dejstva transakcije na bazu podataka

48.Šta je podatak?

Podatak je iskaz definisan jednom prostom izjavnom rečenicom

49.Šta je informacija?

Informacija je novi podatak koji poseduje neku relevantnu novinu,neko novo saznanje,a rezultat je obrade poznatih podataka.

50.Šta je atribut?

Atribut mora biti tako izabran da se može i mora opisati samo jednim elementarnim(atomarnim podatkom)

51.Opisati(ukratko) mrežni model baze podataka

Mrežni model informacionog sistema nastao je kao proširenje hijerarhijskog.Osnovna razlika,nazovimo to poboljšanje,u odnosu na hijerarhijski model,leži u činjenici da se dozvolilo da "naslednik" ima više "više roditelja",što znači da ovakav model prihvata i veze tipa **M:N**.

52.Opisati(ukratko) relacioni model baze podataka

Relacioni model je danas najpopularniji model baze podataka i to zahvaljujući pre svega sledećim osobinama:

- Struktura modela je veoma jednostavna,baza podataka predstavlja skup tabela
- Moguća je formalno-matematička interpretacija tabela

Kao što mu i samo ime govori,ovaj model se zasniva na tabelama i relacijama među njima.Kako se i relacija(veza) između tabela može u relacionom modelu opet prikazati kao tabela,u relacionom modelu sve je definisano samo tabelama.

53.Šta rade i kako se koriste klauzule GROUP BY i HAVING?

Klauzula **GROUP BY**,koju treba da sledi lista atributa,koristi se za grupisanje n-torki na osnovu nekog kriterijuma.Naime,naredba **SELECT** kao rezultat daje opet relaciju,pa n-torke nisu složene ni pokakvom redu,jer to p definiciji relacije nije ni potrebno.Naredbom **GROUP BY** n-torke u relaciji bivaju presložene na način da sve n-torke unutar grupe imaju jednake vrednosti atributa po kome se grupišu.Ako se navede više atributa po kojima treba vršiti grupisanje,prvo se ređaju n-torke sa jednakom vrednošću prvog atributa,zatim se unutar tih grupa preslažu n-torke prema vrednostima drugog atributa itd...

Grupisanje se može vršiti po više kolona,i tada svaka različita kombinacija kolona predstavlja jednu grupu.U okviru dobijenih grupa mogu se uvoditi dodatni uslovi za selekciju primenom klauzule **HAVING**(koji imaju)