

## Ispit iz Baza podataka

7. srpnja 2017.

Zadaci 1 - 5 odnose se na bazu podataka **restoran** koja je prikazana na **slici 1**. Na slici nisu prikazane sve n-torke sadržane u relacijama.

osoba			narudzba				stavka		
sifOsoba	prezime	ime	brNar	sifOsoba	datNar	iznos	brNar	sifJelo	kolicina
1001	Kolar	Marko	451	1003	25.6.2017.	402.00	451	12	2
1002	Ban	Iva	452	1002	26.6.2017.	102.00	451	14	2
1003	Horvat	Ana	453	1001	1.7.2017.	341.00	451	11	4
1004	Novak	Luka	459	1005	1.7.2017.	132.00	452	14	1
							452	16	1

kategorija			jelo				
sifKat	nazKat	sifNadKat	sifJelo	nazJelo	sifKat	cijenaKn	cRazred
101	Meso	NULL	11	Pileći odrezak	204	39.00	NULL
102	Riba	NULL	12	Teleće pečenje	206	51.00	NULL
202	Morska riba	102	13	Skuša na žaru	301	65.00	srednje
204	Perad	101	14	Losos	301	72.00	skupo
301	Plava morska riba	202					

Slika 1.

Baza podataka **restoran** sadrži podatke o narudžbama jela u jednom restoranu. Relacija **osoba** sadrži podatke o osobama koje naručuju dostavu jela iz restorana. Relacija **kategorija** sadrži podatke o kategorijama jela, gdje svaka kategorija može imati nula ili više potkategorija te najviše jednu natkategoriju. Jela (relacija **jelo**) mogu pripadati samo kategoriji koja nema niti jednu potkategoriju. Relacija **narudzba** pohranjuje podatke o izdanim narudžbama, dok relacija **stavka** opisuje stavke pojedine narudžbe. Atribut **sifNadKat** relacije **kategorija** i atribut **cRazred** relacije **jelo** su jedini atributi koji mogu poprimiti NULL vrijednost.

U zadacima 1 - 3 napisati po jednu SQL naredbu kojom će se obaviti sljedeće:

- (5 bodova)** Za jela koja su naručivana u tekućem mjesecu ažurirati cjenovni razred (atribut **cRazred**). Predviđeni razredi su: 'jeftino' – za cijene jela manje od 40 kuna, 'srednje' – za cijene jela između 40 i 70 kuna (uključivo) i 'skupo' – za cijene veće od 70 kuna. Ažuriranje treba biti ispravno bez obzira u kojem mjesecu se naredba pokrene (npr. ako će se pokrenuti u listopadu 2018. godine, uzimat će se u obzir jela naručivana u listopadu 2018. godine).
- (5 bodova)** Za svaku kategoriju ispisati šifru, naziv, broj direktno nadređenih kategorija (nula ili jedna) i broj direktno podređenih kategorija (nula ili više). Kategorije poredati abecedno prema nazivu. **Zadatak riješiti bez podupita.**
- (6 bodova)** Za svaku osobu i kategoriju naručenog jela ispisati šifru osobe, prezime, ime, naziv kategorije jela i najveću količinu jela koju je osoba naručila u toj kategoriji, ali samo ako je ta količina veća od prosječne količine jela naručivanih u toj kategoriji te ako je drugo slovo prezimena osobe (veliko ili malo) slovo 'a'.
- (5 bodova)** Pretpostavite da su u bazi podataka **restoran** kreirane sve relacije te da su definirani svi primarni i strani ključevi. Napisati naredbe kojima će se osigurati da se prilikom unosa novih zapisa u relaciju **stavka** ažuriraju i odgovarajući iznosi u relaciji **narudzba**. Pri tome treba osigurati da ukupni iznos na nekoj narudžbi ne prijeđe iznos od 5000kn. Ako se to dogodi, treba spriječiti operaciju unosa u relaciju **stavka** i dojaviti poruku „Prevelik iznos narudžbe!“.
- (6 bodova)** Korisnik **postgres** kreirao je bazu podataka **restoran** i relacije u njoj sljedećim SQL naredbama:

```
CREATE DATABASE restoran;
CREATE TABLE osoba (...); CREATE TABLE narudzba (...); ...
REVOKE CONNECT ON DATABASE restoran FROM PUBLIC;
REVOKE ALL ON SCHEMA public FROM PUBLIC;
REVOKE ALL ON ALL TABLES IN SCHEMA public FROM PUBLIC;
```

- Kreirati ulogu **novak** s mogućnošću uspostavljanja SQL sjednice i pri tome koristiti autentifikaciju pomoću lozinke. Korisniku za inicijalnu lozinku postaviti ime korisnika ("novak"), omogućiti mu pristup bazi podataka te mu dati mogućnost pregledavanja objekata sheme **public** bez mogućnosti kreiranja novih objekata.
- Kreirati ulogu **ulogaRest** i omogućiti toj ulozi sljedeće:
  - pregled svih podataka o stavkama narudžbi
  - pregled, unos, izmjenu i brisanje podataka o jelima koja su skuplja od 50 kuna i pripadaju kategoriji "Puretina".
- Napisati naredbu kojom se spomenute ovlasti dodjeljuju korisniku **novak**.

- d) Postojećem korisniku (uloži) kolar, koji nema nikakvih ovlasti, omogućiti pregled i izmjenu svih podataka o jelima s mogućnošću dodjeljivanja tih dozvola drugim korisnicima.

6. (5 bodova) Na relacijskoj shemi  $R = \{ A, B, C, D, E, F, G \}$  zadan je skup funkcijskih zavisnosti

$F = \{ A \rightarrow E, C \rightarrow G, EF \rightarrow C, AD \rightarrow F, GF \rightarrow B \}$ .

- Dokažite da vrijedi funkcijska zavisnost  $AD \rightarrow ABCDEFG$ . Navedite pravila korištena uz svaki korak dokaza.
- Je li funkcijska zavisnost  $AD \rightarrow ABCDEF$  potpuna ili nepotpuna? Dokažite.
- Je li skup atributa  $\{AD\}$  dobar izbor za ključ relacije  $R$ ? Zašto?

7. (4 boda) U bazi su pohranjene tablice  $r, s$  i  $t$  koje odgovaraju relacijskim shemama  $R(A, B, C)$ ,  $S(C, D)$  i  $T(B, E)$ . Optimizator upita ne raspolaže informacijama o ključevima tablica, ali raspolaže sljedećim statističkim informacijama vezano uz navedene tablice:

$r(A, B, C)$	$s(C, D)$	$t(B, E)$
$N(r) = 2000$	$N(s) = 1000$	$N(t) = 800$
$V(A, r) = 500$	$V(C, s) = 50$	$V(B, t) = 100$
$V(B, r) = 200$	$V(D, s) = 100$	$V(E, t) = 50$
$V(C, r) = 100$		

Obavlja se operacija  $r \bowtie s \bowtie t$ . Obrazložite koji redosljed spajanja će odabrati optimizator.

8. (4 boda) Prikazan je odsječak sadržaja relacije **boardGame** u trenutku započinjanja naredbe  $\{1\}$ . Odredite rezultate naredbi  $\{5\}$ ,  $\{7\}$ , i  $\{8\}$  ako su u trenutku započinjanja transakcija A, B i C aktivne transakcije s identifikatorima 10 i 12, a zadnja potvrđena transakcija ona s identifikatorom 11. Pretpostavite da do kraja naredbe  $\{10\}$

- osim transakcije A niti jedna druga transakcija ne mijenja sadržaj relacije **boardGame**
- su transakcije 10 i 12 ostale aktivne
- nije aktiviran proces koji "čisti" stare n-torke (garbage collector)

boardGame			
xmin	xmax	bgld	bgName
1	0	1	Clank
2	10	2	Mith
10	0	2	Myth
12	0	3	Sapiens

Transakcija A T <sub>id</sub> =13		Transakcija B T <sub>id</sub> =14		Transakcija C T <sub>id</sub> =15	
{1}	BEGIN TRANSACTION;	{2}	BEGIN TRANSACTION;	{3}	BEGIN TRANSACTION;
{4}	INSERT INTO boardGame VALUES (4, 'Santorini');				SET TRANSACTION ISOLATION LEVEL SERIALIZABLE;
{5}	SELECT xmin, xmax, * FROM boardGame WHERE bgld BETWEEN 1 AND 4;	{7}	SELECT xmin, xmax, * FROM boardGame WHERE bgld BETWEEN 1 AND 4;	{8}	SELECT xmin, xmax, * FROM boardGame WHERE bgld BETWEEN 1 AND 4;
{6}	COMMIT TRANSACTION;	{9}	COMMIT TRANSACTION;	{10}	COMMIT TRANSACTION;

9. (7 bodova) Tvrtka za iznajmljivanje automobila evidentira iznajmljivanje automobila pravnim ili fizičkim osobama. Obje vrste osoba identificiraju se OIB-om. Pored toga, za pravnu osobu bilježimo naziv i matični broj, a za fizičku osobu ime i prezime osobe. Automobil je identificiran tvorničkim brojem (ne postoje dva automobila s istim tvorničkim brojem). Za automobil treba evidentirati godinu proizvodnje i model automobila. Modeli automobila identificirani su proizvođačem i nazivom modela (međusobno različiti proizvođači mogu svoje modele nazivati istim imenom - npr. Fiat može imati svoj model naziva Sport, a Toyota može imati sasvim drugi model koji se također naziva Sport). Za model automobila evidentira se godina u kojoj je model prvi put proizveden te mogući rezervni dijelovi. Svaki rezervni dio identificira se šifrom dijela, a dodatno se bilježe naziv i opis rezervnog dijela te informacija kojim sve modelima automobila taj dio odgovara. Proizvođač automobila ima naziv, a identificiran je svojom šifrom. Osoba može određenog datuma unajmiti jedan ili više automobila. Isti automobil može u istom danu iznajmiti jedna ili više osoba. Osoba može isti automobil unajmiti više puta, ali ne istoga dana. Nakon što osoba vrati automobil, bilježi se broj sati koliko je bio iznajmljen te broj prijeđenih kilometara.

Nacrtati ER model baze podataka, navesti sheme entiteta i sheme veza (označiti ključeve). Svaki entitet (osim slabih entiteta) opisuje se isključivo vlastitim atributima. Nužno je da sve sheme zadovoljavaju 3NF.

10. (3 boda) Zadana je relacijska shema  $R$  i skupovi atributa  $X \subseteq R$ ,  $Y \subseteq R$ ,  $Z \subseteq R$ .

- Nadopunite: Skup atributa  $Z$  je tranzitivno ovisan o  $X$  ako vrijedi:  $X \rightarrow Y$ , \_\_\_\_\_ i \_\_\_\_\_.
- Kada je relacijska shema u 3. normalnoj formi?

# RJEŠENJA

## 1. UPDATE jelo

```
SET cRazred =
CASE
    WHEN cijenaKn <40 THEN 'jeftino'
    WHEN cijenaKn BETWEEN 40 and 70 THEN 'srednje'
    ELSE 'skupo'
END
WHERE sifJelo IN (
    select sifJelo
    from narudzba join stavka
    on narudzba.brNar = stavka.brNar
    where EXTRACT (month from datNar) = EXTRACT (month from CURRENT_DATE)
    AND EXTRACT (year from datNar) = EXTRACT (year from CURRENT_DATE)
)
```

## 2. Rješenje 1:

```
SELECT kat.sifKat, kat.nazKat, COUNT(DISTINCT nad.sifKat) brojNadredjenihKat,
        COUNT(pod.sifKat) brojPodredjenihKat
FROM   kategorija AS kat LEFT JOIN kategorija AS nad ON kat.sifNadKat=nad.sifKat
        LEFT JOIN kategorija AS pod ON pod.sifNadKat=kat.sifKat
GROUP BY kat.sifKat, kat.nazKat
ORDER BY kat.nazKat;
```

## Rješenje 2:

```
SELECT kat.sifKat, kat.nazKat,
        CASE
            WHEN kat.sifNadKat IS NULL THEN 0
            ELSE 1
        END as brojnadredjenihkat,
        COUNT(pod.sifKat) brojPodredjenihKat
FROM   kategorija AS kat
        LEFT JOIN kategorija AS pod ON pod.sifNadKat=kat.sifKat
GROUP BY kat.sifKat, kat.nazKat
ORDER BY kat.nazKat;
```

## 3.

```
SELECT osoba.sifOsoba, prezime, ime, nazKat, MAX(kolicina)
FROM   osoba NATURAL JOIN narudzba
        NATURAL JOIN stavka s1
        NATURAL JOIN jelo j1
        NATURAL JOIN kategorija
WHERE UPPER(SUBSTRING(prezime FROM 2 FOR 1)) = 'A'
GROUP BY osoba.sifOsoba, prezime, ime, nazKat, j1.sifKat
HAVING MAX(kolicina)>= ( SELECT AVG(kolicina)
                        FROM stavka s2 NATURAL JOIN jelo j2
                        WHERE j2.sifKat = j1.sifKat          -- korelacija s vanjskim
                        );
```

```

4. CREATE FUNCTION chkInsStavka()
    RETURNS TRIGGER AS $$
DECLARE
    p_cijena DECIMAL(8,2);
    p_iznos DECIMAL(8,2);
BEGIN
    SELECT cijenaKn INTO p_cijena
        FROM jelo
        WHERE sifJelo = NEW.sifJelo;

    SELECT iznos INTO p_iznos
        FROM narudzba
        WHERE brNar = NEW.brNar;

    IF (p_iznos + p_cijena * NEW.kolicina > 5000) THEN
        RAISE EXCEPTION 'Prevelik iznos narudžbe!';
    ELSE
        UPDATE narudzba
            SET iznos = iznos + p_cijena * NEW.kolicina
            WHERE brNar = NEW.brNar;
    END IF;

    RETURN NEW;
END;
$$ LANGUAGE plpgsql;

CREATE TRIGGER stavkaIns
    BEFORE INSERT ON stavka
    FOR EACH ROW
        EXECUTE PROCEDURE chkInsStavka();

```

## 5.

a) --postgres:

```

CREATE ROLE novak LOGIN PASSWORD 'novak';
GRANT CONNECT ON DATABASE restoran TO novak;
GRANT USAGE ON SCHEMA PUBLIC TO novak;

```

b)

```

CREATE ROLE ulogaRest;
GRANT USAGE ON SCHEMA public TO ulogaRest;

```

```

GRANT SELECT ON stavka TO ulogaRest;

```

```

CREATE VIEW skupljaPuretina AS
SELECT *
FROM jelo
WHERE cijenaKn > 50
    AND sifKat IN
    (
        SELECT sifKat
        FROM kategorija
        WHERE nazKat = 'Puretina'
    )
WITH CHECK OPTION;

```

```

GRANT SELECT, INSERT, UPDATE, DELETE ON skupljaPuretina TO ulogaRest;

```

c)

GRANT ulogaRest TO novak;

d)

GRANT CONNECT ON DATABASE restoran TO kolar;

GRANT USAGE ON SCHEMA PUBLIC TO kolar;

GRANT SELECT, UPDATE ON jelo TO kolar WITH GRANT OPTION;

6. a)

$AD \rightarrow AD$       refleksivnost

$AD \rightarrow AD \wedge AD \rightarrow F \Rightarrow AD \rightarrow ADF$       akumulacija

$AD \rightarrow ADF \wedge A \rightarrow E \Rightarrow AD \rightarrow ADEF$       akumulacija

$AD \rightarrow ADEF \wedge EF \rightarrow C \Rightarrow AD \rightarrow ACDEF$       akumulacija

$AD \rightarrow ACDEF \wedge C \rightarrow G \Rightarrow AD \rightarrow ACDEFG$       akumulacija

$AD \rightarrow ACDEFG \wedge GF \rightarrow B \Rightarrow AD \rightarrow ABCDEFG$       akumulacija

b) Vrijedi li funkcijska zavisnost  $A \rightarrow ABCDEF$ ?

$A \rightarrow A$       refleksivnost

$A \rightarrow A \wedge A \rightarrow E \Rightarrow A \rightarrow AE$       akumulacija

FZ ne vrijedi.

Vrijedi li funkcijska zavisnost  $D \rightarrow ABCDEF$ ?

$D \rightarrow D$       refleksivnost

FZ ne vrijedi.

FZ je potpuna jer ne postoji pravi podskup od AD koji funkcijski određuje ABCDEFG.

c) Da, jer je to minimalan skup atributa koji funkcijski određuje attribute relacijske sheme R.

7.

Postoje tri moguća rasporeda spajanja:  $(r \bowtie s) \bowtie t$ ,  $(r \bowtie t) \bowtie s$  i  $(s \bowtie t) \bowtie r$ .

1)  $(r \bowtie s) \bowtie t$

$$N(r \bowtie s) = N(r) * N(s) / \max(V(C,r), V(C,s)) = 2000 * 1000 / \max(100, 50) = 2,000,000 / 100 = \mathbf{20,000}$$

2)  $(r \bowtie t) \bowtie s$

$$N(r \bowtie t) = N(r) * N(t) / \max(V(B,r), V(B,t)) = 2000 * 800 / \max(200, 100) = 1,600,000 / 200 = \mathbf{8,000}$$

3)  $(s \bowtie t) \bowtie r$

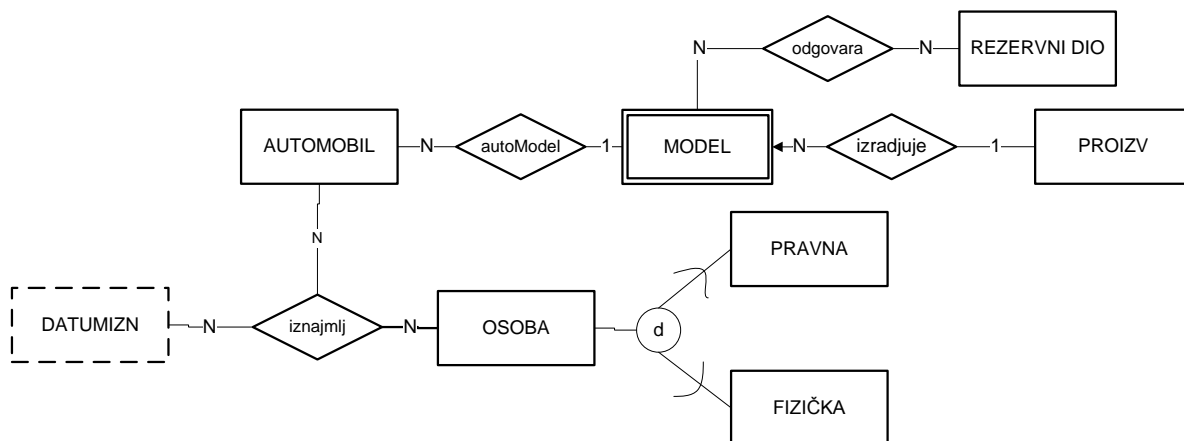
$$N(s \bowtie t) = 1000 * 800 = \mathbf{800,000}$$

**Slučaj 2) je optimalan** jer je međurezultat najmanji.       $(r \bowtie t) \bowtie s$

8.

{5}				{7}				{8}			
<i>xmin</i>	<i>xmax</i>	<i>bgld</i>	<i>bgName</i>	<i>xmin</i>	<i>xmax</i>	<i>bgld</i>	<i>bgName</i>	<i>xmin</i>	<i>xmax</i>	<i>bgld</i>	<i>bgName</i>
1	0	1	Clank	1	0	1	Clank	1	0	1	Clank
2	10	2	Mith	2	10	2	Mith	2	10	2	Mith
13	0	4	Santorini	13	0	4	Santorini				

9.



<b>AUTOMOBIL</b> <u>tvBroj</u> godinaProizv	<b>PROIZV</b> <u>sifProizv</u> nazivProizv	<b>MODEL</b> <u>sifProizv</u> <u>nazModel</u> godPrveProizv	<b>REZERVNI DIO</b> <u>sifDijela</u> nazDijela opisDijela	<b>OSOBA</b> <u>OIB</u>	<b>PRAVNA</b> <u>OIB</u> naziv matBroj K2={matBroj}
<b>iznajm</b> <u>datumlzn</u> <u>tvBroj</u> <u>OIB</u> brSati brKm	<b>autoModel</b> <u>tvBroj</u> sifProizv nazModel	<b>izradjuje</b> <u>sifProizv</u> <u>nazModel</u>	<b>odgovara</b> <u>sifProizv</u> <u>nazModel</u> <u>sifDijela</u>		<b>FIZIČKA</b> <u>OIB</u> ime prezime

10.

a) Skup atributa Z je tranzitivno ovisan o X ako vrijedi:  $X \rightarrow Y$ ,  $Y$  ne određuje X i  $Y \rightarrow Z$ .

b) Relacijska shema je u 3NF ako je u 1NF i ako:

niti jedan atribut iz zavisnog dijela nije tranzitivno funkcijski ovisan o bilo kojem ključu relacije.