

Baze blic 4. lab

1.

Zadan je ER model s opisima entiteta PREDMET, STUDENT i PREDMETROK:

```
graph LR
    PREDMET -- "rokPredmet" --> PREDMETROK
    PREDMETROK -- "studentRok" --> STUDENT
```

Koja od sljedećih tvrdnji je istinita?

a Entitet PREDMETROK je **egzistencijalno** i **identifikacijski** slabi entitet koji nema vlasnika.

b Entitet PREDMETROK je **egzistencijalno** i **identifikacijski** slabi entitet i njegov vlasnik je entitet PREDMET.

2.

Koja je od sljedećih virtualnih relacija **neizmjenjiva**?

a

```
CREATE VIEW pogled (sifra, prezime, sifra_odjela, placa) AS
SELECT sifra, prezime, sif_odj, placa
FROM zaposlenik
WHERE sif_odj IN
(SELECT sif_odj FROM odjel WHERE naziv LIKE 'Uprava');
```

b

```
CREATE VIEW pogled (sifra, iznos_place) AS
SELECT sifra, placa FROM zaposlenik;
```

c

```
CREATE VIEW pogled AS SELECT * FROM zaposlenik WHERE sif_odj > 1 WITH CHECK OPTION;
```

d

```
CREATE VIEW pogled (prezime, sifOdj) AS
SELECT DISTINCT prezime, sif_odj
FROM zaposlenik;
```

e

```
CREATE VIEW pogled AS
SELECT sifra, placa
FROM zaposlenik
WHERE sifra < 1003;
```

3.

Koja od navedenih izjava je **točna**?

- a** Virtualna relacija se stvara naredbom CREATE TEMP TABLE.
- b** Shema virtualne relacije se nikada trajno ne pohranjuje u sustav.
- c** Sve virtualne relacije su izmjenjive.
- d** Virtualna relacija je drugi naziv za privremenu relaciju.
- e** U nekim SUBPovima virtualne relacije su implementirane tako da se njihov sadržaj fizički pohranjuje te mijenja prilikom promjene sadržaja neke od temeljnih relacija pomoću kojih je virtualna relacija definirana.

4.

Knjige:

SifKnjige	Naslov	IzdavačID
00926299	Baudolino	111
008371298	Snijeg	111
007898798	O ljepoti	221
006123566	Golub	304

Izdavači:

SifIzdavača	Izdavač
111	Meander
112	Ljusak
221	Durieux
304	Mladost

Relacija **Knjige** stvorena je naredbom:

```
CREATE TABLE Knjige(
  SifKnjige VARCHAR(20)
  , Naslov VARCHAR(20)
  , IzdavačID INT(10)
  , FOREIGN KEY (IzdavačID) REFERENCES Izdavači(SifIzdavača) ON DELETE SET NULL
);
```

Ako pokušamo izbrisati n-torku <304, Mladost> iz relacije **Izdavači** naredbom:

```
DELETE FROM Izdavači
WHERE SifIzdavača=304;
```

SUBP će:

- a** Ništa od navedenog.
- b** Izbrišati n-torku <304, Mladost> iz relacije **Izdavači**.
- c** Dozvoliti ovu akciju, ali će još i izbrisati i n-torku <006123566, Golub, 304> iz relacije **Knjige**.
- d** Sustav će izbrisati n-torku <304, Mladost>, a n-torku <006123566, Golub, 304> promijenit će u <NULL, NULL, NULL>.

5.

Izvršena je naredba:

```
CREATE TABLE student(  
  jmbag    INTEGER  
  , ime    VARCHAR(20) NOT NULL  
  , prezime VARCHAR(20)  
  , akGodina INTEGER  
  , CHECK (prezime IS NOT NULL)  
  , CHECK (akGodina>2000));
```

Potom je pokušano izvršenje sljedećih 5 SQL naredbi:

- (1) `INSERT INTO student VALUES(NULL, 'Pero', 'Peric', 2001);`
- (2) `INSERT INTO student VALUES(12345, 'Djelo', 'Hadžiselimovic', 2000);`
- (3) `INSERT INTO student VALUES(12347, 'Mirko', 'Mirkovic', 2002);`
- (4) `INSERT INTO student VALUES(12348, 'Mirko', NULL, 2005);`
- (5) `INSERT INTO student VALUES(12349, NULL, 'Slavko', 2005);`

Koje naredbe su uspješno izvršene?

a Samo naredba (2).

b Naredbe (1) i (3).

c Naredbe (2) i (5).

d Nijedna naredba nije uspješno izvršena.

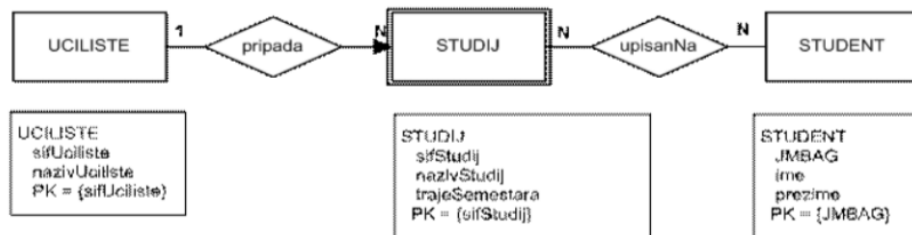
e Naredbe (1), (3) i (5).

6.

Pronađite netočnu izjavu:

- a** Za zrcaljenje je karakteristična visoka dostupnost.
- b** Poželjno je da su transakcije što veće.
- c** Inkrementalno arhiviranje omogućuje stvaranje arhiva različitih razina.
- d** Zrcaljenje se može obavljati pod kontrolom SUBP-a.
- e** Današnji sustavi omogućuju da se arhiviranje obavlja tijekom rada korisnika.

7.



Koja od sljedećih tvrdnji je istinita:

- a** Entitet STUDIJ je **egzistencijalno** i **identifikacijski** slabi entitet i njegov vlasnik je entitet STUDENT.
- b** Entitet STUDIJ je **egzistencijalno** slabi entitet, ali **NIJE identifikacijski** slabi entitet i njegov vlasnik je entitet UC
- c** Entitet STUDIJ je **egzistencijalno** i **identifikacijski** slabi entitet i njegov vlasnik je entitet UCILISTE.

8.

Pretpostavimo da n-torke tablice `empsalary` izgledaju ovako (`xmin` i `xmax` su "skriveni atributi" koje koristi MVCC mehanizam kontrole istodobnog pristupa):

<code>xmin</code>	<code>xmax</code>	<code>empid</code>	<code>salary</code>	
1978	0	112	5400.00	✓
1978	1980	113	7200.00	✓
1979	0	114	8100.00	✗
1976	1980	115	8200.00	✓
1980	0	115	5300.00	✗
1981	0	116	5800.00	✗

Koji je rezultat upita kojeg izvršava sljedeća transakcija:

```

BEGIN TRANSACTION;
SET TRANSACTION ISOLATION LEVEL SERIALIZABLE;
SELECT * FROM empsalary;
COMMIT TRANSACTION;
  
```

ako vrijedi:

- ID najranije aktivne transakcije je **1977**
- ID koji još nije dodijeljen je **1985**
- aktivne transakcije su **1977, 1979, 1980, 1981**

i ako pretpostavimo da je gornja transakcija dobila identifikator **1984** i da su sve transakcije s nižim identifikatorima (a koje nisu u gornjoj listi aktivnih transakcija) uspješno izvršene?

a

<code>empid</code>	<code>salary</code>
112	5400.00
113	7200.00
115	8200.00

b

<code>empid</code>	<code>salary</code>
112	5400.00
113	7200.00
114	8100.00
115	5300.00
116	5800.00

9.

Izvršena je naredba:

```
CREATE TABLE predmet(  
  sifra      INTEGER  
  , naziv   VARCHAR(20) NOT NULL  
  , ectsbod INTEGER  
  , akGodina INTEGER  
  , CHECK (ectsbod > 5)  
  , CHECK (akGodina > 2000));
```

Potom je pokušano izvršenje sljedećih 5 SQL naredbi:

- (1) INSERT INTO predmet VALUES(NULL, 'Matematika', 6, 2001); ✓
- (2) INSERT INTO predmet VALUES(12345, 'Programiranje', 4, 2003); ✗
- (3) INSERT INTO predmet VALUES(12347, 'Fizika', 6, 2002); ✓
- (4) INSERT INTO predmet VALUES(12348, 'Algoritmi', 10, 2000); ✗
- (5) INSERT INTO predmet VALUES(12349, 'Baze podataka', 8, 1999); ✗

Koje naredbe su uspješno izvršene?

<input type="radio"/> a	Naredbe (1), (3) i (4).
<input type="radio"/> b	Samo naredba (1).
<input type="radio"/> c	Nijedna naredba nije uspješno izvršena.
<input type="radio"/> d	Samo naredba (4).
<input checked="" type="radio"/> e	Naredbe (1) i (3).

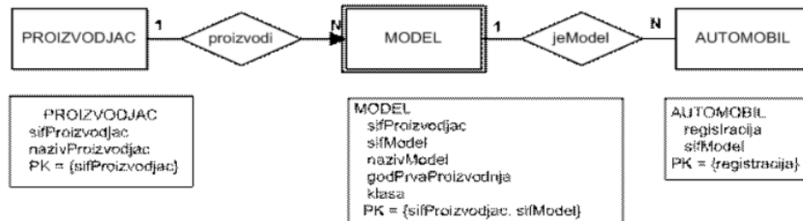
10.

Koja od navedenih ključnih riječi se može koristiti za definiranje ograničenja odnosa među vrijednostima atributa u istoj n-torci?

<input checked="" type="radio"/> a	CHECK
<input type="radio"/> b	NOT NULL
<input checked="" type="radio"/> c	UNIQUE
<input type="radio"/> d	PRIMARY KEY
<input type="radio"/> e	CASE

11.

Zadan je ER model s opisima entiteta PROIZVOJAC, AUTOMOBIL i MODEL:



Koja od sljedećih tvrdnji je istinita?

a

Entitet MODEL je **egzistencijalno i identifikacijski** slabi entitet i njegov vlasnik je entitet PROIZVOJAC.

12.

Nad temeljnom relacijom **student** u bazi studAdmin stvara se virtualna relacija **rodjeni1994** koja obuhvaća samo studente rođene 1994. godine. Virtualna relacija stvara se naredbom:

```
CREATE VIEW rodjeni1994 (jmbag, prezime, ime, droid)
AS
SELECT jmbag, prezimeStudent, imeStudent, datumrod
FROM student
WHERE datumrod >= '01.01.1994'
AND datumrod <= '31.12.1994'
WITH CHECK OPTION;
```

Virtualna relacija **rodjeni1994** sadrži ukupno 80 n-torki od kojih je nekoliko prikazano:

jmbag	prezime	ime	droid
0555001482	Lešić	Nikola	03.06.1994
0555001589	Vančina	Igor	04.09.1994
0555001601	Bataljaku	Bojan	22.03.1994
0555001617	Čemeljic	Jan	14.02.1994
...

80, 524
(isto) → pogreška

Temeljna relacija **student** sadrži ukupno 524 n-torke.

Ako se nad virtualnom relacijom izvrši naredba:

```
INSERT INTO rodjeni1994 VALUES ('0036111333', 'Kralj', 'Andrija', '01.05.1993');
```

koliko će se n-torki nakon njena izvršenja nalaziti u temeljnoj relaciji **student**, a koliko u virtualnoj relaciji **rodjeni1994**?

13.

Ako se promatra u okvirima ER modela, što je to **atribut**?

- a Pojam **atribut** nije definiran u okvirima ER modela.
- b **Atribut** je matematička relacija između n entiteta.
- c **Atribut** je bilo što, što ima suštinu ili bit, ima jasnoću kao činjenica ili ideja, posjeduje značajke pomoću kojih se može razlučiti od svoje okoline.
- d **Atribut** je funkcija koja preslikava iz skupa entiteta ili skupa veza u skup vrijednosti.
- e **Atribut** je funkcija koju skup entiteta obavlja u skupu veza.

14.

Nakon stvaranja **studAdmin** baze podataka izvršene su sljedeće naredbe:

```
REVOKE CONNECT ON DATABASE "studAdmin" FROM PUBLIC;  
REVOKE ALL ON SCHEMA public FROM PUBLIC;
```

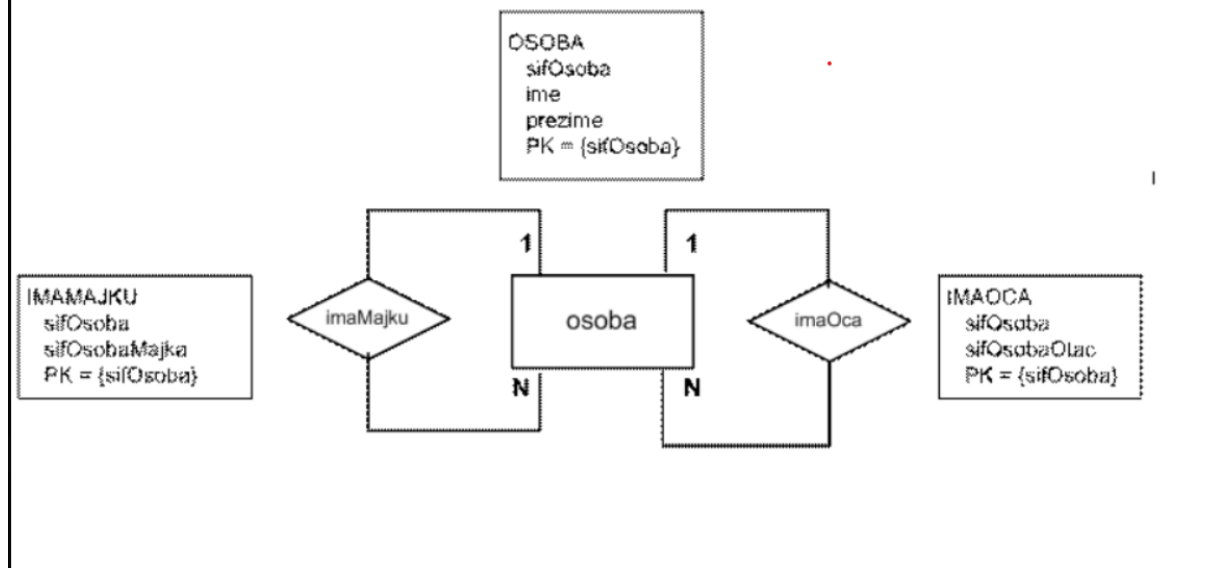
Koje sve dozvole **minimalno** je potrebno dodijeliti **novom** korisniku kako bi mogao pregledavati zapise tablice **student** u **studAdmin** bazi podataka?

connect (na bazu), usage (na shemu), select na student public

(connect na bazu, usage na public shemu, select na student)

15.

Koji od ponuđenih odgovora (relacijske sheme) odgovara zadanom ER modelu?



a

OSOBA = sifOsoba, ime, prezime

OSOBARODITELJ = sifOsoba, sifOsobaMajka, sifOsobaOtac

b

OSOBA = sifOsoba, ime, prezime

RODITELJ = sifOsoba, ime, prezime, sifDijete

c

OSOBA = sifOsoba, ime, prezime

OSOBAROTAC = sifOsoba, sifOsobaOtac

OSOBAMAJKA = sifOsoba, sifOsobaMajka

d

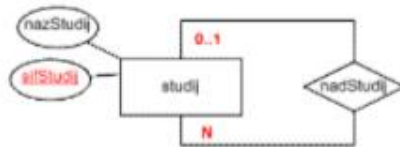
OSOBA = sifOsoba, ime, prezime, sifOsoba, sifOsoba

e

OSOBA = sifOsoba, ime, prezime, sifOsobaMajka, sifOsobaOtac

16.

Zadan je ER model s entitetom **studij** te vezom **nadStudij**. Potcrtan (i crveno obojen) atribut je ključ entiteta.



Prikazani ER model je potrebno preslikati u ekvivalentan **relacijski model** navođenjem SQL naredbi za kreiranje relacija s ugrađenim općim pravilima integriteta. Koje od ponuđenih rješenja je ispravno?

(ništ od ovoga)

a `sql CREATE TABLE studij (sifStudij INT PRIMARY KEY , nazStudij VARCHAR(50) NOT NULL);
CREATE TABLE nadStudij (sifStudij INT REFERENCES studij(sifStudij) , sifNadStudij INT REFERENCES studij(sifStudij) , PRIMARY KEY(sifNadStudij));`

b `CREATE TABLE studij (
sifStudij INT PRIMARY KEY
, nazStudij VARCHAR(50) NOT NULL
, sifNadStudij INT REFERENCES studij(sifStudij));

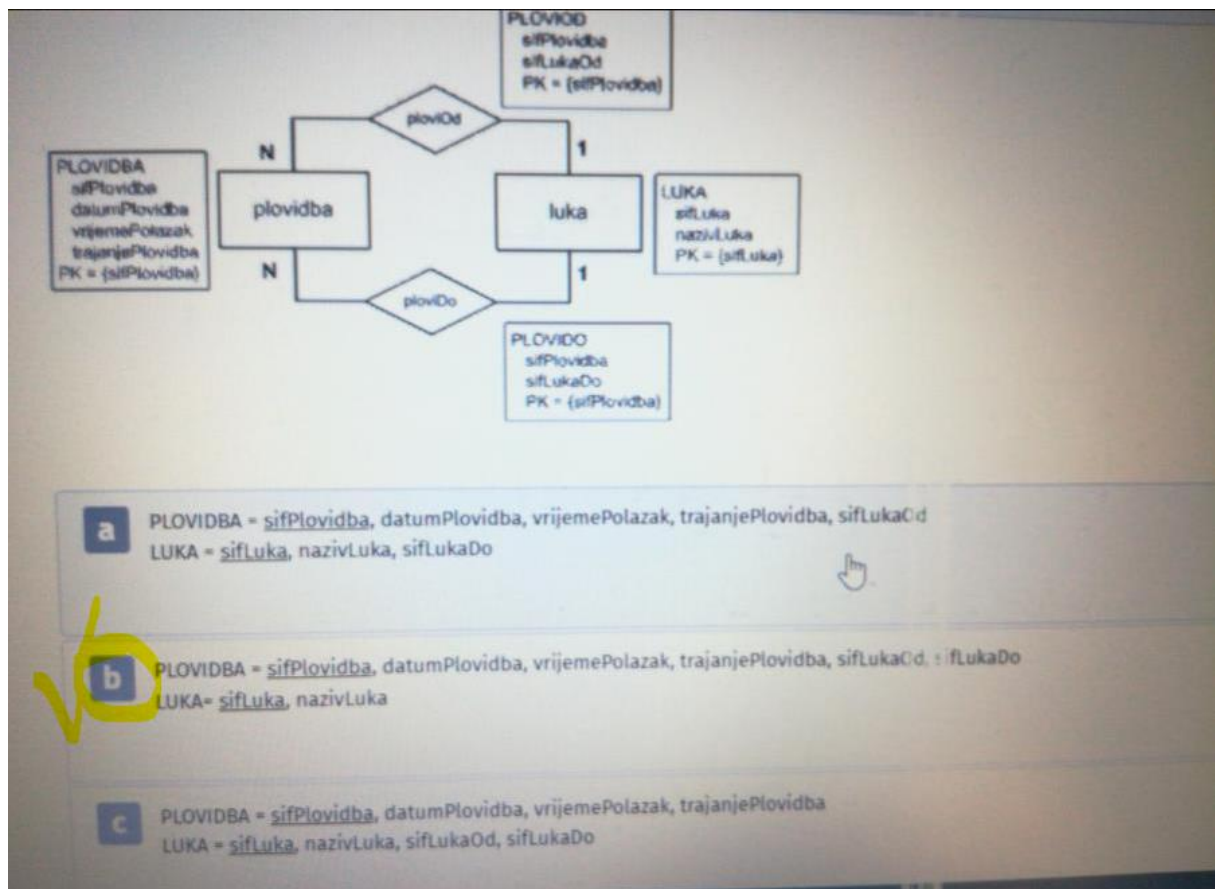
CREATE TABLE nadStudij (
sifStudij INT REFERENCES studij(sifStudij)
, sifNadStudij INT REFERENCES studij(sifStudij));`

c `sql CREATE TABLE studij (sifStudij INT PRIMARY KEY , nazStudij VARCHAR(50) NOT NULL);
CREATE TABLE nadStudij (sifStudij INT REFERENCES studij(sifStudij) , sifNadStudij INT REFERENCES studij(sifStudij) , PRIMARY KEY(sifStudij));`

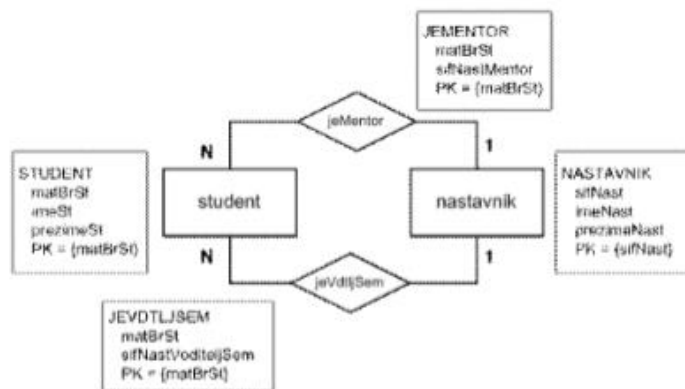
d `CREATE TABLE studij (
sifStudij INT PRIMARY KEY
, nazStudij VARCHAR(50) NOT NULL);

CREATE TABLE nadStudij (
sifStudij INT REFERENCES studij(sifStudij)
, sifNadStudij INT
PRIMARY KEY(sifNadStudij));`

17.



18.



STUDENT = matBrSt, imeSt, prezimeSt, sifNastMentor, sifNastVoditeljSem
 NASTAVNIK = sifNast, imeNast, prezimeNast



STUDENT = matBrSt, imeSt, prezimeSt
 STUDNAST = matBrSt, sifNastMentor, sifNastVoditeljSem
 NASTAVNIK = sifNast, imeNast, prezimeNast



STUDENT = matBrSt, imeSt, prezimeSt, sifNastMentor
 NASTAVNIK = sifNast, imeNast, prezimeNast, sifNastVoditeljSem



STUDENT = matBrSt, imeSt, prezimeSt
 NASTAVNIK = sifNast, imeNast, prezimeNast, sifNastMentor, sifNastVoditeljSem



STUDENT = matBrSt, imeSt, prezimeSt, sifNast, sifNast
 NASTAVNIK = sifNast, imeNast, prezimeNast

19.

Korisnik *kolar* stvorio je shemu **poslovanje** i u njoj tablicu **oprema**. Kada korisnik *kolar* pokušava obaviti sljedeće SQL upite, koji će upit vratiti grešku:

a	<code>DROP SCHEMA poslovanje CASCADE;</code>
b	<code>DROP TABLE poslovanje.oprema;</code>
c	<code>SELECT * FROM poslovanje.oprema;</code>
d	<code>SELECT * FROM oprema;</code>
e	<code>DROP SCHEMA poslovanje;</code>

20.

Privremena relacija...

a	...stvara se naredbom CREATE TABLE.
b	...se ne može ukloniti naredbom DROP TABLE.
c	...vidljiva je samo u sklopu SQL sjednice u kojoj je stvorena.
d	...stvara se naredbom CREATE VIEW.
e	...dinamički mijenja svoj sadržaj nakon upisa novog retka u temeljnu relaciju na osnovu koje je stvorena.

21.

Koji entiteti posjeduju i druge atribute osim svojih vlastitih atributa?

a Entiteti ne mogu posjedovati druge atribute osim svojih vlastitih atributa.

b Entiteti u ternarnoj vezi.

c Identifikacijski slabi entiteti.

d Regularni entiteti.

e Egzistencijalno, ali ne i identifikacijski slabi entiteti.

22.

Asistentima na fakultetu se pomoću virtualne relacije **predmetiAsistenata** omogućava održavanje (čitanje i izmjene) podataka o predmetima koji se održavaju u trenutnoj akademskoj godini. Virtualna relacija **predmetiAsistenata** stvorena je na sljedeći način:

```
CREATE VIEW predmetiAsistenata AS
SELECT *
FROM predmet
WHERE predmet.sifpredmet IN
(SELECT predmetgrupa.sifpredmet
FROM predmetgrupa
WHERE predmetgrupa.akgodina =
CASE
WHEN EXTRACT(MONTH FROM CURRENT_DATE) > 9
THEN EXTRACT(YEAR FROM CURRENT_DATE)
ELSE EXTRACT(YEAR FROM CURRENT_DATE) - 1
END)
WITH CHECK OPTION;
```

Koje je sve naredbe potrebno izvršiti kako bi stvorili ulogu **asistent**, te kako bi korisnik novak postao **asistent**?

a CREATE ROLE asistent;
GRANT ALL ON predmetiAsistenata TO asistent;
GRANT asistent TO novak;

b CREATE ROLE asistent;
GRANT ALL ON predmetiAsistenata TO asistent WITH CHECK OPTION;
GRANT asistent TO novak;

c CREATE ROLE asistent;
GRANT ALL ON predmetiAsistenata TO novak;

d CREATE ROLE asistent;
GRANT SELECT, UPDATE ON predmetiAsistenata TO novak;

e CREATE ROLE asistent;
GRANT SELECT, UPDATE ON predmetiAsistenata TO asistent;
GRANT asistent TO novak;

23.

Koja od sljedećih izjava **NIJE** istinita?

- a** Ključ veze **uvijek** sadrži ključeve **svih** entiteta koje veza povezuje.
- b** Ključ veze **može** sadržavati vlastite atribute veze.
- c** Ključ veze ovisi o ključevima entiteta koje veza povezuje i o njihovim spojnostima u vezi.
- d** Shema veze **može** sadržavati vlastite atribute.
- e** Shema veze **uvijek** sadrži ključeve **svih** entiteta koje povezuje.


24.

Kad završava SQL sjednica?

- a** kada se korisniku oduzme pravo prijave na sustav za upravljanje bazama podataka
- b** kada sustav za upravljanje bazama podataka javi grešku u izvođenju SQL naredbe
- c** kada korisnik prekine vezu prema sustavu za upravljanje bazama podataka
- d** nakon izvođenja svake SQL naredbe – svaka pojedina SQL naredba otvara novu SQL sjednicu
- e** kada korisnik izvede posljednju SQL naredbu

25.

Privremena relacija...

- a ...dinamički mijenja svoj sadržaj nakon upisa novog retka u temeljnu relaciju na osnovu koje je stvorena.
- b ...stvora se naredbom CREATE VIEW.
- cse ne može ukloniti naredbom DROP TABLE. 
- d ...stvora se naredbom CREATE TABLE.
- e ...vidljiva je samo u sklopu SQL sjednice u kojoj je stvorena.

26.

Asistentima na fakultetu se pomoću virtualne relacije **predmetiAsistenata** omogućava održavanje (Čitanje i izmjene) podataka o predmetima koji se održavaju u trenutnoj akademskoj godini. Virtualna relacija **predmetiAsistenata** stvorena je na sljedeći način:

```
CREATE VIEW predmetiAsistenata AS
SELECT *
FROM predmet
WHERE predmet.sifpredmet IN
  (SELECT predmetgrupa.sifpredmet
   FROM predmetgrupa
   WHERE predmetgrupa.akgodina =
     CASE
       WHEN EXTRACT(MONTH FROM CURRENT_DATE) > 9
       THEN EXTRACT(YEAR FROM CURRENT_DATE)
       ELSE EXTRACT(YEAR FROM CURRENT_DATE) - 1
     END)
WITH CHECK OPTION;
```

Koje je sve naredbe potrebno izvršiti kako bi stvorili ulogu **asistent**, te kako bi korisnik novog postao **asistent**?

- a CREATE ROLE asistent;
GRANT ALL ON predmetiAsistenata TO asistent;
GRANT asistent TO novak;
- b CREATE ROLE asistent;
GRANT SELECT, UPDATE ON predmetiAsistenata TO novak;
- c CREATE ROLE asistent;
GRANT ALL ON predmetiAsistenata TO novak;
- d CREATE ROLE asistent;
GRANT SELECT, UPDATE ON predmetiAsistenata TO asistent;
GRANT asistent TO novak;
- e CREATE ROLE asistent;

27.

Pretpostavimo da n-torka tablice *empsalary* koji opisuje plaću zaposlenika sa šifrom 112 izgleda ovako (*xmin* i *xmax* su "skriveni atributi" koje koristi MVCC mehanizam kontrole istodobnog pristupa):

<i>xmin</i>	<i>xmax</i>	<i>empid</i>	<i>salary</i>
1978	0	112	5400.00

Pretpostavimo da želimo izvršiti sljedeću transakciju:

```
BEGIN TRANSACTION;
UPDATE empsalary
  SET salary = 5500.00 WHERE empid = 112;
COMMIT TRANSACTION;
```


Kako izgledaju n-torke koje se odnose na plaću zaposlenika sa šifrom 112 nakon provođenja ove transakcije, ako znamo da je transakcija od sustava dobila identifikator 1985 i da proces koji "čisti" stare n-torke još nije aktiviran?

a

<i>xmin</i>	<i>xmax</i>	<i>empid</i>	<i>salary</i>
1978	1985	112	5500.00

b

<i>xmin</i>	<i>*xmax*</i>	<i>empid</i>	<i>salary</i>
1978	1985	112	5400.00
1985	0	112	5500.00



28.

Što zovemo migrirajućim n-torkama?

- a** n-torke iz jedne virtualne relacije koje nisu vidljive iz druge virtualne relacije
- b** n-torke koje premještamo iz privremene relacije u virtualnu
- c** n-torke unesene u trajnu relaciju koje nisu vidljive u virtualnoj relaciji
- d** n-torke unesene u privremenu relaciju koje nestaju nakon isteka SQL sjednice
- e** n-torke unesene u virtualnu relaciju koje se kasnije ne pojavljuju u toj relaciji

29.

Next question (can't go back)

Nakon stvaranja **studAdmin** baze podataka izvršene su sljedeće naredbe:

```
REVOKE CONNECT ON DATABASE "studAdmin" FROM PUBLIC;
REVOKE ALL ON SCHEMA public FROM PUBLIC;
```

Koje sve dozvole **minimalno** je potrebno dodijeliti **novom** korisniku kako bi mogao pregledavati zapise tablice **student** u **studAdmin** bazi podataka?

- a GRANT USAGE ON SCHEMA public TO korisnik;
GRANT SELECT ON student TO korisnik;
- b GRANT CONNECT ON DATABASE studAdmin TO korisnik;
GRANT CREATE ON DATABASE studAdmin TO korisnik;
GRANT SELECT ON student TO korisnik;
- c GRANT CONNECT ON DATABASE studAdmin TO korisnik;
GRANT SELECT ON student TO korisnik;
- d GRANT CONNECT ON DATABASE studAdmin TO korisnik;
GRANT USAGE ON SCHEMA public TO korisnik;
GRANT SELECT ON student TO korisnik;

30.

Koja od navedenih izjava je **točna**?

- a Sadržaj privremene relacije mijenja se prilikom promjene sadržaja temeljnih relacija koje se koriste u njezinoj definiciji.
- b Privremenu relaciju je moguće ukloniti iz baze podataka obavljanjem naredbe DROP TABLE.
- c Za nekog korisnika je privremena relacija "vidljiva" u okviru svih SQL-sjednica koje je taj korisnik ostvario sa instancom SUBPa.
- d Prilikom kreiranja privremene relacije moguće je definirati ograničenja referencijskog integriteta.
- e Problem "zastarijevanja" podataka u virtualnim relacijama može se izbjeći uporabom privremenih relacija.

31.

Koju anomaliju će prouzročiti sljedno izvršavanje sljedećih naredbi (razina izolacije za obje transakcije je READ UNCOMMITTED):

NAPOMENA: pretpostavite da RDBMS kojeg koristite implementira razinu izolacije READ UNCOMMITTED na način kako je ista definirana SQL standardom

Transakcija A

```
BEGIN WORK;
UPDATE artikli SET iznos = iznos+1 WHERE sifra = 1482;
```

Transakcija B

```
BEGIN WORK;
SELECT iznos FROM artikli WHERE sifra = 1482;
```

Transakcija A

```
SELECT iznos FROM artikli WHERE sifra = 1482;
ROLLBACK WORK;
```

Transakcija B

```
COMMIT WORK;
```

priglasno čitanje?

32.

Što je to identifikacijski slabi entitet?

a Identifikacijski slabi entitet **može** postojati samostalno, ali za identifikaciju mu nisu dovoljni vlastiti atributi već se koriste i ključni atributi nekog drugog entiteta.

b Identifikacijski slabi entitet **ne može** postojati ukoliko ne postoji i njegov entitet vlasnik i za identifikaciju mu **nisu** dovoljni vlastiti atributi već se koriste i ključni atributi entiteta vlasnika.

c Identifikacijski slabi entitet **ne može** postojati ukoliko ne postoji i njegov entitet vlasnik i za identifikaciju mu nisu dovoljni vlastiti atributi

33.

Nad temeljnom relacijom **student** u bazi studAdmin stvara se virtualna relacija **slovoA** koja obuhvaća samo studente čije prezime počinje slovom A. Virtualna relacija stvara se naredbom:

```
CREATE VIEW slovoA (jmbag, prezime, ime)
AS
SELECT jmbag, prezimeStudent, imeStudent
FROM student
WHERE prezimeStudent like 'A%';
WITH CHECK OPTION;
```

Virtualna relacija **slovoA** sadrži ukupno 5 n-torki:

jmbag	prezime	ime
0555004185	Asić	Igor
0555004965	Arbanas	Ana
0555004986	Adamović	Marijana
0555005002	Abramović	Gordana
0555005198	Anić	Ana

5, 525 (isto)

Temeljna relacija **student** sadrži ukupno 524 n-torki.

Ako se nad virtualnom relacijom izvrši naredba

```
INSERT INTO slovoA VALUES ('0836111333', 'Kralj', 'Andrija');
```

koliko će se n-torki nakon njena izvršenja nalaziti u temeljnoj relaciji **student**, a koliko u virtualnoj relaciji **slovoA**?

34.

Zadana je relacija:

Knjiga

sif knjige	naslov	cijena	valuta	jezik
1	Fahrenheit 451	15	USD	ENG
2	Na česli	12	USD	ENG
3	Suite Française	95	HRK	FR
4	Madame Bovary	150	HRK	FR
5	O ljepoti	150	HRK	HR
6	Snijeg	220	HRK	HR

Izveden je sljedeći SQL upit:

```
CREATE VIEW cijena_kunama(Naslov, Cijena_u_kunama) AS
SELECT naslov, CASE
    WHEN valuta='USD' THEN cijena*5
    ELSE cijena
END AS prihod
FROM knjiga
WHERE cijena>13
```

Nakon izvođenja naredbe:

```
SELECT * FROM cijena_kunama;
```

ispisuje se tablica:

a	Naslov	Cijena_u_kunama
	Fahrenheit 451	75
	Suite Française	95
	Madame Bovary	150
	O ljepoti	150
	Snijeg	220

35.

Koja od navedenih izjava je **točna**?

a Sve virtualne relacije su izmjenjive.

b Virtualna relacija se stvara naredbom CREATE TEMP TABLE.

c U nekim SUBPovima virtualne relacije su implementirane tako da se njihov sadržaj fizički pohranjuje te mijenja prilikom promjene sadržaja neke od temeljnih relacija pomoću kojih je virtualna relacija definirana.

d Virtualna relacija je drugi naziv za privremenu relaciju.

e Shema virtualne relacije se nikada trajno ne pohranjuje u sustav.

36.

Nakon stvaranja **studAdmin** baze podataka izvršene su sljedeće naredbe:

```
REVOKE CONNECT ON DATABASE "studAdmin" FROM PUBLIC;  
REVOKE ALL ON SCHEMA public FROM PUBLIC;
```

Koje sve dozvole **minimalno** je potrebno dodijeliti **novom** korisniku kako bi mogao pregledavati zapise tablice **student** u **studAdmin** bazi podataka?

a

```
GRANT CONNECT ON DATABASE studAdmin TO korisnik;  
GRANT USAGE ON SCHEMA public TO korisnik;  
GRANT SELECT ON student TO korisnik;
```

b

```
GRANT SELECT ON student TO korisnik;
```

c

```
GRANT CONNECT ON DATABASE studAdmin TO korisnik;  
GRANT SELECT ON student TO korisnik;
```

37.

Tablice **proizvod** i **proizvodSkladiste** baze podataka **skladiste** kreirane su u shemi **public**:

proizvodSkladiste					proizvod		
sif_proizvod	sif_skladiste	kolicina	polozaj	zadnji_dat_zaprimanje	sif_proizvod	tezina	opis
101	1	4	10AA	15.6.2007.	101	1	krastavci
102	1	1	12GA	8.6.2007.	102	0.5	napolitanke
101	2	1	11BB	9.6.2007.	103	180	sapun

Podatke evidentiraju djelatnici s korisničkim imenima: horvat i novak. Njihove ovlasti su:

horvat:

- pregledavanje svih podataka u tablici **proizvod**
- pregledavanje, izmjena, dodavanje i brisanje svih podataka u tablici **proizvodSkladiste**

novak:

- pregledavanje svih podataka u tablici **proizvod**
- pregledavanje svih podataka u tablici **proizvodSkladiste** i izmjena vrijednosti atributa **polozaj**

Korisnik **PUBLIC** ima predefiniране ovlasti u bazi podataka **skladiste**. Administrator je obavio sljedeći niz naredbi:

Administrator je izvršio sljedeći niz naredbi:

```
CREATE LOGIN horvat WITH PASSWORD 'horvatPwd';  
CREATE LOGIN novak WITH PASSWORD 'novakPwd';  
GRANT CONNECT ON DATABASE skladiste TO horvat;  
GRANT CONNECT ON DATABASE skladiste TO novak;  
  
CREATE ROLE uloga1;  
GRANT SELECT, INSERT, UPDATE, DELETE ON proizvodSkladiste TO uloga1;  
  
CREATE ROLE uloga2;  
GRANT SELECT ON proizvodSkladiste TO uloga2;  
GRANT SELECT ON proizvod TO uloga2;  
  
CREATE ROLE uloga3;  
GRANT UPDATE (polozaj) ON proizvodSkladiste TO uloga3;  
  
GRANT uloga1 to horvat;  
GRANT uloga2 to horvat;  
GRANT uloga3 to novak;
```

Koja od sljedećih izjava je istinita?

a

Korisniku **horvat** su dodijeljene sve potrebne ovlasti

b

Niti jednom korisniku nisu dodijeljene sve potrebne ovlasti

c

Korisnici nemaju dozvolu korištenja sheme **public** u kojoj su kreirane gornje tablice

Koja od sljedećih izjava je istinita?

a

Korisniku **horvat** su dodijeljene sve potrebne ovlasti

b

Niti jednom korisniku nisu dodijeljene sve potrebne ovlasti

c

Korisniku **horvat** i korisniku **novak** su dodijeljene sve potrebne ovlasti

d

Korisnici nemaju dozvolu korištenja sheme **public** u kojoj su kreirane gornje tablice

e

Korisniku **novak** su dodijeljene sve potrebne ovlasti

38.

Administrator baze podataka stvorio je uloge sljedećim naredbama:

```
CREATE ROLE horvat LOGIN INHERIT;  
CREATE ROLE admin INHERIT;  
CREATE ROLE korisnik NOINHERIT;  
CREATE ROLE student NOINHERIT;  
  
GRANT admin TO horvat;  
GRANT korisnik TO admin;  
GRANT student TO korisnik;
```

Kada korisnik *horvat* otvori SQL sjednicu prema bazi podataka ima prava koja su dodijeljena ulogama:

- a** horvat i admin
- b** horvat, admin i student
- c** horvat, admin i korisnik
- d** horvat i korisnik
- e** horvat i student