

ИНФОРМАЦИОНИ СИСТЕМИ И БАЗЕ ПОДАТАКА

У следећим задацима заокружите број испред траженог одговора

374.	<p>Једна од функција система за управљање базом података је ауторизација.</p> <p>Означити шта она регулише:</p> <ol style="list-style-type: none"> 1. Права надокнаде за коришћење базе података. 2. Права приступа појединих корисника појединим објектима (подацима и другим ресурсима) као и права извршења разних операција над тим објектима. 3. Ауторска права над базом података и програмима који користе ту базу података. 	1
375.	<p>Ниво софтвера између корисника и физичких података у бази, који штити кориснике базе од детаља на физичком нивоу, назива се:</p> <ol style="list-style-type: none"> 1. Програмски језик 2. Систем за управљање базама података 3. База података 4. Case алат 	1
376.	<p>Посматрајмо релациони модел БИБЛИОТЕКЕ.</p> <p>Издвојити ком делу релационог модела припада услов да два издавача не могу имати исти назив:</p> <ol style="list-style-type: none"> 1. Структурном делу 2. Манипулативном делу 3. Интегративном делу 4. Софтверском делу 	1
377.	<p>Одредити ентитете који садрже атрибуте који НИСУ одговарајући:</p> <ol style="list-style-type: none"> 1. Ентитет: СТУДЕНТ – Атрибути: име, презиме, смер, број бодова, просек 2. Ентитет: ДРЖАВА – Атрибути: назив, број становника, површина 3. Ентитет: КЊИГА – Атрибути: наслов, аутор, година издања, издавач, адреса издавача, телефон издавача 4. Ентитет: АВИОН – Атрибути: произвођач, марка, година производње, број седишта 	1
378.	<p>Сврха јединственог идентификатора (кључа) је да:</p> <ol style="list-style-type: none"> 1. Јединствено одређује колону те табеле. 2. Јединствено одређује табелу и врсту табеле. 3. Јединствено одређује инстанцу ентитета. 4. Јединствено одређује ентитет у бази података. 	1
379.	<p>Одредити шта релација М:М најчешће крије:</p> <ol style="list-style-type: none"> 1. Још атрибута 2. Још једну релацију 3. Још један ентитет 4. Ништа од наведеног 	1

380.	Одредити шта тачно резервисана реч DISTINCT проузрокује у SELECT наредби:	1
	<ol style="list-style-type: none"> 1. Не приказује NULL вредности. 2. Елиминише све јединствене вредности из табеле. 3. Елиминише све врсте које се више пута јављају у приказу. 4. Елиминише само јединствене врсте у приказу. 	
381.	Одредити оператор који би требало користити у SELECT наредби да би били приказани само они ученици чије презиме почиње словом А:	1
	<ol style="list-style-type: none"> 1. IN 2. LIKE 3. BETWEEN 4. AS LIKE 5. IS NULL 	
382.	Одредити оператор који би требало употребити у SELECT наредби да би биле приказане све врсте табеле UCENICI које садрже null вредности у колони stipendija_id:	1
	<ol style="list-style-type: none"> 1. "=NULL" 2. ISNULL 3. NULL 4. IS NULL 5. LIKE NULL 	
383.	Одредити коју клаузулу је потребно додати упиту SELECT prezime, ime, skola_id, mentor_id FROM takmicari да би такмичари били уређени најпре по школи, а онда по ментору, па по презимену ученика:	1
	<ol style="list-style-type: none"> 1. ORDER BY skola_id, prezime 2. ORDER BY skola_id, mentor_id, prezime, ime 3. ORDER BY skola_id, mentor_id, prezime 4. ORDER BY prezime, mentor_id, skola_id 	
384.	Извршава се упит: SELECT prezime, ime, email FROM ucenici ORDER BY prezime DESC Заокружити исказ који је истинит:	1
	<ol style="list-style-type: none"> 1. Све врсте ће бити сортиране по презимену у абецедном поретку. 2. Све врсте ће бити сортиране по презимену у обрнутом абецедном поретку. 3. Све врсте ће бити сортиране по презимену, а затим по имену. 	
385.	Одредити резервисану реч коју је потребно укључити у ORDER BY клаузулу да би приказали податке из табеле са подацима о такмичарима уређене у опадајућем поретку по броју бодова:	1
	<ol style="list-style-type: none"> 1. DESC 2. ASC 3. SORT 4. CHANGE 5. SORT DESC 	

<p>386. Одабрати клаузулу коју је потребно користити уколико задатак захтева да SELECT упит врати подкуп података:</p> <ol style="list-style-type: none"> 1. ANYWHERE 2. WHICH 3. WHERE 4. EVERY 	<p>1</p>
<p>387. За преглед података из табеле која се налази у оквиру базе података користи се:</p> <ol style="list-style-type: none"> 1. ADD 2. DROP 3. INSERT 4. SELECT 	<p>1</p>
<p>388. За измену структуре у табели која се налази у оквиру базе података - додавање или брисање – уклањање колоне користи се:</p> <ol style="list-style-type: none"> 1. SELECT 2. CREATE 3. INSERT 4. ALTER 	<p>1</p>
<p>389. За додавање нове колоне у табелу, која се налази у оквиру базе података, користи се:</p> <ol style="list-style-type: none"> 1. ADD_COLUMN 2. NEW_COLUMN 3. INSERT 4. ADD 	<p>1</p>

390. Дата је табела KUPCI:

Id	Prezime	Adresa	Mesto	PostBroj	Status
1	Antić	Mokroluška 35	Bečej	21220	Aktivan
2	Kostić	Jug Bogdanova 12	Niš	18000	Pasivan
3	Marić	Masarikova 1	Pirot	18300	Pasivan
4	Perić	Takovska 67	Niš	18000	Pasivan
5	Rakić	Konstantinova 22	Sombor	25000	Aktivan
6	Tomić	Kneza Miloša 13	Bečej	21220	Aktivan

Извршава се упит:

DELETE FROM KUPCI WHERE Mesto = 'Niš';

Колико записа у табели ће остати?

1. 0
2. 2
3. 4
4. 3

1

391. Дата је табела KUPCI:

Id	Prez me	Adresa	Mesto	PostBroj	Status
1	Antić	Mokroluška 35	Bečej	21220	Aktivan
2	Kostić	Jug Bogdanova 12	Niš	18000	Pasivan
3	Marić	Masarikova 1	Pirot	18300	Pasivan
4	Perić	Takovska 67	Niš	18000	Pasivan
5	Rakić	Konstantinova 22	Sombor	25000	Aktivan
6	Tomić	Kneza Miloša 13	Bečej	21220	Aktivan

Извршава се упит:

ALTER TABLE KUPCI ADD (DatUgovora (Date));

Одредити шта ће се десити након извршења упита:

1. Промениће се тип података у колони DatUgovora у табели KUPCI
2. Биће додат нови запис – врста у табелу KUPCI
3. Биће додата нова колона под именом DatUgovora у табелу KUPCI
4. Биће преименована колона у табели KUPCI
5. Биће уклоњена – обрисана колона у табели KUPCI

1

392. Дата је табела KUPCI:

I	Prezime	Adresa	Mesto	PostBroj	Status
1	Antić	Mokroluška 35	Bečej	21220	Aktivan
2	Kostić	Jug Bogdanova 12	Niš	18000	Pasivan
3	Marić	Masarikova 1	Pirot	18300	Pasivan
4	Perić	Takovska 67	Niš	18000	Pasivan
5	Rakić	Konstantinova 22	Sombor	25000	Aktivan
	Tomić	Kneza Miloša 13	Bečej	21220	Aktivan

1

Извршава се упит:

SELECT Mesto, Adresa FROM KUPCI WHERE Prezime like '%tić';

Одредити колико редова ће имати резултантна табела не рачунајући заглавље:

- 0 редова
- 1 ред
- 2 реда**
- 3 реда
- 6 редова

393. За проширење упита за издвајање дела података из табела базе података, као додаток, користи се клаузула:

- ALL
- EVERYONE
- WHEN
- WHERE**

1

394. Дата је табела KUPCI:

Id	Prezime	Adresa	Mesto	Postroj	Status
1	Antić	Mokroluška 35	Bečej	21220	Aktivan
2	Kostić	Jug Bogdanova 12	Niš	18000	Pasivan
3	Marić	Masarikova 1	Pirot	18300	Pasivan
4	Perić	Takovska 67	Niš	18000	Pasivan
5	Rakić	Konstantinova 22	Sombor	25000	Aktivan
6	Tomić	Kneza Miloša 13	Bečej	21220	Aktivan

1

Извршава се упит:

SELECT Mesto, Adresa FROM KUPCI WHERE Prezime like '%mić';

Одредити колико редова ће имати резултантна табела не рачунајући заглавље:

- 0 редова
- 1 ред**
- 2 реда
- 3 реда
- 6 редова

395. Дата је табела KUPCI:

Id	Prezime	Adresa	Mesto	PostBroj	Status
1	Antić	Mokroluška 35	Bečej	21220	Aktivan
2	Kostić	Jug Bogdanova 12	Niš	18000	Pasivan
3	Marić	Masarikova 1	Pirot	18300	Pasivan
4	Perić	Takovska 67	Niš	18000	Pasivan
5	Rakić	Konstantinova 22	Sombor	25000	Aktivan
6	Tomić	Kneza Miloša 13	Bečej	21220	Aktivan

1

Извршава се упит:

SELECT DISTINCT Mesto FROM KUPCI;

Одредити колико редова ће имати резултантна табела не рачунајући заглавље:

1. 5 редова
2. 4 реда
3. 2 реда
4. 3 реда
5. 6 редова

396. Издвојити могући резултат рада следеће SQL наредбе:

```
SELECT ucenik_id
FROM ucenici
WHERE ucenik_id BETWEEN 200 AND 250
OR ucenik_id IN (220, 280, 310)
AND (ucenik_id BETWEEN 250 AND 300)
```

1

1. 100
2. 301, 302, 303, 304, 305, 306, 307, 308, 309
3. 200, 201, 202, 203, 204, 205, 215, 225, 227, 228, 248
4. 120, 180, 310

397. Дата је табела KUPCI:

d	Prezime	Adresa	Mesto	PostBroj	Status
1	Antić	Mokroluška 35	Bečej	21220	Aktivan
2	Kostić	Jug Bogdanova 12	Niš	18000	Pasivan
3	Marić	Masarikova 1	Pirot	18300	Pasivan
4	Perić	Takovska 67	Niš	18000	Pasivan
5	Rakić	Konstantinova 22	Sombor	25000	Aktivan
6	Tomić	Kneza Miloša 13	Bečej	21220	Aktivan

1

Извршава се упит:

SELECT Prezime, Mesto, Adresa FROM KUPCI WHERE LENGTH(Adresa) > 14;

Одредити колико редова ће имати резултантна табела не рачунајући заглавље:

1. 5 редова
2. 4 реда
3. 2 реда
4. 3 реда
5. 0 редова

398. Дата је табела KUPCI:

Id	Prezime	Adresa	Mesto	PostBroj	Status
1	Antić	Mokroluška 35	Bečej	21220	Aktivan
2	Kostić	Jug Bogdanova 12	Niš	18000	Pasivan
3	Marić	Masarikova 1	Pirot	18300	Pasivan
4	Perić	Takovska 67	Niš	18000	Pasivan
5	Rakić	Konstantinova 22	Sombor	2500	Aktivan
6	Tomić	Kneza Miloša 13	Bečej	21220	Aktivan

Извршава се упит:

SELECT Prezime, Mesto, Adresa FROM KUPCI WHERE LENGTH(Adresa) < 10;

Одредити колико редова ће имати резултантна табела не рачунајући заглавље:

1. 5 редова
2. 4 реда
3. 2 реда
4. 1 ред
5. 0 редова

1

399. Дата је табела ARTIKLI:

Id	Naziv	DatProiz	JM	Kol	Cena	Akcija
21	Hleb Sava	15.04.2015	kom	100	35	
32	Hleb Dunav	15.04.2015	kom	50	45	0.05
23	Baget beli	12.04.2015	kom	20	32	0.12
47	Baget crni	12.04.2015	kom	12	35	0.08
25	Tost tamni	10.04.2015	kom	6	75	0.15
36	Tost beli	10.04.2015	kom	8	70	
28	Kifla integralna	15.04.2015	kom	15	18	0
10	Brašno Graham	10.03.2015	kg	24	54	0

Одредити упит који приказује називе за све артикле и за сваки одредити укупну цену за дату количину:

1. **SELECT Naziv, Kol * Cena as Ukupno FROM ARTIKLI;**
2. SELECT Naziv, count(Kol * Cena) FROM ARTIKLI;
3. SELECT Naziv, COUNT (Kol , Cena) as Ukupno FROM ARTIKLI;
4. SELECT Naziv, Kol + Cena FROM ARTIKLI;

1

400. Дата је табела ARTIKLI:

Id	Naziv	DatProiz	JM	Kol	Cena	Akcija
21	Hleb Sava	15.04.15	kom	100	35	
32	Hleb Dunav	15.04.15	kom	50	45	0.05
23	Baget beli	12.04.15	kom	20	32	0.12
47	Baget crni	12.04.15	kom	12	35	0.08
25	Tost tamni	10.04.15	kom	6	75	0.15
36	Tost beli	10.04.15	kom	8	70	
28	Kifla integralna	15.04.15	kom	15	18	0
10	Brašno Graham	10.03.15	kg	24	54	0

Одредити упит који приказује називе за све артикле на акцији и нове цене, умањене за акцијски попуст:

1. SELECT Naziv, Akcija * Cena FROM ARTIKLI where Akcija is not NULL;
2. SELECT Naziv, (1-Akcija) * Cena FROM ARTIKLI where Akcija is NULL;
3. SELECT Naziv, Akcija * Cena FROM ARTIKLI where Akcija <> 0 (NULL);
4. **SELECT Naziv, (1-Akcija) * Cena FROM ARTIKLI where Akcija is not NULL;**

1

401. Дата је табела ARTIKLI:

Id	Naziv	DatProiz	JM	Kol	Cena	Akcija
21	Hleb Sava	15.04.15	kom	100	35	
32	Hleb Dunav	15.04.15	kom	50	45	0.05
23	Baget beli	12.04.15	kom	20	32	0.12
47	Baget crni	12.04.15	kom	12	35	0.08
25	Tost tamni	10.04.15	kom	6	75	0.15
36	Tost beli	10.04.15	kom	8	70	
28	Kifla integralna	15.04.15	kom	15	18	0
10	Brašno Graham	10.03.15	kg	24	54	0

Извршава се упит:

SELECT Naziv, DatProiz, Kol from Artikli where Id < 30 and Cena <= 45;

Одредити колико редова има табела која је резултат рада упита не рачунајући заглавље:

1. 0 редова
2. 1 ред
3. 2 реда
4. **3 реда**

1

402. Дата је табела ARTIKLI:

Id	Naziv	DatProiz	JM	Kol	Cena	Akcija
23	Baget beli	12.04.15	kom	20	32	0.12
47	Baget crni	12.04.15	kom	12	35	0.08
25	Tost tamni	10.04.15	kom	6	75	0.15
36	Tost beli	10.04.15	kom	8	70	
28	Kifla integralna	15.04.15	kom	15	18	0
10	Brašno Graham	10.03.15	kg	24	54	0

Извршава се упит:

SELECT max(Cena), min(Cena) from Artikli;

Заокружити број испред траженог одговора.

Одредити резултат рада упита:

1.	max(Cena)	min(Cena)
	75	32
2.	max(Cena)	min(Cena)
	75	18
3.	min(Cena)	max(Cena)
	6	100
4.	min(Cena)	max(Cena)
	18	54

1

403. Дата је табела ARTIKLI:

Id	Naziv	DatProiz	JM	Kol	Cena	Akcija
21	Hleb Sava	15.04.15	kom	100	35	
32	Hleb Dunav	15.04.15	kom	50	45	0.05
23	Baget beli	12.04.15	kom	20	32	0.12
47	Baget crni	12.04.15	kom	12	35	0.08
25	Tost tamni	10.04.15	kom	6	75	0.15
36	Tost beli	10.04.15	kom	8	70	
28	Kifla integralna	15.04.15	kom	15	18	0
10	Brašno Graham	10.03.15	kg	24	54	0

Извршава се упит:

SELECT Naziv, DatProiz, Kol from Artikli where Id > 30 and Cena >= 45;

Резултат извршења упита је табела са заглављем и

- 0 редова и 2 колоне
- 1 редом и 3 колоне
- 2 реда и 3 колоне
- 3 реда и 2 колоне

1

404. Дата је табела KUPCI:

Id	Prezime	Adresa	Mesto	PostBroj	Status
1	Antić	Mokroluška 35	Bečej	21220	Aktivan
2	Kostić	Jug Bogdanova 12	Niš	18000	Pasivan
3	Marić	Masarikova 1	Pirot	18300	Pasivan
4	Perić	Takovska 6	Niš	18000	Pasivan
5	Rakić	Konstantinova 22	Sombor	25000	Aktivan
6	Tomić	Kneza Miloša 13	Bečej	21220	Aktivan

Извршава се упит:

SELECT Prezime, Mesto, Adresa FROM KUPCI WHERE LENGTH(Adresa) > 10;

Резултат извршења упита је табела са заглављем и

1. 5 редова и 3 колоне
2. 4 реда и 4 колоне
3. 2 реда и 3 колоне
4. 1 редом и 2 колоне
5. 0 редова и 3 колоне

1

405. Дата је табела KUPCI:

Id	Prezime	Adresa	Mesto	PostBroj	Status
1	Antić	Mokroluška 35	Bečej	21220	Aktivan
2	Kostić	Jug Bogdanova 12	Niš	18000	Pasivan
3	Marić	Masarikova 1	Pirot	18300	Pasivan
4	Perić	Takovska 6	Niš	18000	Pasivan
5	Rakić	Konstantinova 22	Sombor	25000	Aktivan
6	Tomić	Kneza Miloša 13	Bečej	21220	Aktivan

Извршава се упит:

SELECT count(*) FROM KUPCI WHERE Status= 'Aktivan';

Одредити резултат упита:

1. 3
2. 4
3. 2
4. 1
5. 0

1

406. Дата је табела KUPCI:

Id	Prezime	Adresa	Mesto	PostBroj	Status
1	Antić	Mokroluška 35	Bečej	21220	Aktivan
2	Kostić	Jug Bogdanova 12	Niš	18000	Pasivan
3	Marić	Masarikova 1	Pirot	18300	Pasivan
4	Perić	Takovska 6	Niš	18000	Pasivan
5	Rakić	Konstantinova 22	Sombor	25000	Aktivan
6	Tomić	Kneza Miloša 13	Bečej	21220	Aktivan

Извршава се упит:

SELECT Prezime, Adresa, Mesto FROM KUPCI order by Prezime;

Записи у резултантној табели ће имати:

1. **поредак исти као у приказаној табели**
2. поредак супротан од поретка у приказаној табели
3. поредак измењен у односу на приказану табелу, али не супротан

1

407. Дата је табела KUPCI:

Id	Prez e	Adresa	Mesto	PostBroj	Status
1	Antić	Mokroluška 35	Bečej	21220	Aktivan
2	Kostić	Jug Bogdanova 12	Niš	18000	Pasivan
3	Marić	Masarikova 1	Pirot	18300	Pasivan
4	Perić	Takovska 6	Niš	18000	Pasivan
5	Rakić	Konstantinova 22	Sombor	25000	Aktivan
6	Tomić	Kneza Miloša 13	Bečej	21220	Aktivan

Извршава се упит:

SELECT Prezime, Adresa, Mesto FROM KUPCI order by Mesto;

Заокружити број испред траженог одговора.

Записи у резултантној табели ће имати:

1. поредак исти као у приказаној табели
2. поредак супротан од приказане табеле
3. **поредак измењен у односу на приказану табелу, али не супротан**

1

408.	После извршења дате наредбе: CREATE TABLE RADNIK (IDBR integer, IME varchar(20) NOT NULL, POS AO varchar(15) NOT NULL, KVALIF varchar(3), RUKOVODILAC integer, DATZAP date, PREMIJA float, PLATA float, BROD integer); добија се: 1. табела RADNIK са девет колона од којих се четири обавезно попуњавају 2. табела RADNIK са шест колона од којих се три обавезно попуњавају 3. табела RADNIK са девет колона од којих се две обавезно попуњавају 4. табела RADNIK са осам колона од којих се једна обавезно попуњава	1
409.	После извршења наредбе: CREATE TABLE RADNIK (IDBR integer, IME varchar(20) NOT NULL, POS AO varchar(15) NOT NULL, KVALIF varchar(3), RUKOVODILAC integer, DATZAP date, PREMIJA float, PLATA float, BROD integer); добија се: 1. табела RADNIK са примарним кључем и страним кључем 2. табела RADNIK са примарним кључем и без страног кључа 3. табела RADNIK без примарног кључа и са страним кључем 4. табела RADNIK без примарног кључа и без страног кључа	1
410.	После извршења наредбе: CREATE INDEX ind_ime ON RADNIK (IME ASC); (поставља) се: 1. у табели RADNIK поставља INDEX над колоном IME 2. у табели INDEX поставља IME над колоном RADNIK 3. у табели RADNIK поставља INDEX над колоном ind_ime 4. у табели RADNIK поставља INDEX над колоном ASC	1
411.	Извршењем дате наредбе: ALTER TABLE PROJEKAT ADD RokKraj date; постиже се: 1. у бази података у табели PROJEKAT уклања се колона RokKraj 2. у бази података у табели PROJEKAT додаје се колона RokKraj 3. у бази података у табели RokKraj додаје се колона PROJEKAT 4. у базу података додаје се табела PROJEKAT	1

412. Дата је табела ARTIKLI:

Id	Naziv	DatProiz	JM	Kol	Cena	Akcija
21	Hleb Sava	15.04.15	kom	100	35	
32	Hleb Dunav	15.04.15	kom	50	45	0.05
23	Baget beli	12.04.15	kom	20	32	0.12
47	Baget crni	12.04.15	kom	12	35	0.08
25	Tost tamni	10.04.15	kom	6	75	0.15
36	Tost beli	10.04.15	kom	8	70	
28	Kifla integralna	15.04.15	kom	15	18	0
10	Brašno Graham	10.03.15	kg	24	54	0

1

Заокружити број испред траженог одговора

Одредити упит који коригује – повећава цене за све артикле линеарно за 8 процената у односу на тренутну цену:

1. UPDATE ARTIKLI SET Cena = Cena*0.8;
2. UPDATE ARTIKLI SET Cena AS Cena*0.8;
3. UPDATE ARTIKLI SET Cena = Cena*(1+0.08);
4. UPDATE ARTIKLI SET Cena ASCena*(1 + 0.08);

413. Дата је табела RADNIK:

IDBR	IME	PREZIME	PLATA	PREMIJA	DATZAP
5874	Todor	Marjanović	1000	1100	19.4.1971
5898	Andrija	Aleksić	1100		20.1.1980
5900	Slobodan	Golubović	900	1300	3.10.1978
5932	Mitar	Gavrilović	2600		25.3.1985
5953	Persida	Kosanović	1100	0	12.1.1979
6234	Marko	Pavlović	1300	3000	17.12.1990
6789	Janko	Nikolić	3900	10	23.12.1999

1

Заокружити број испред упита којим ће се приказати само презиме, име и датум запослења за раднике запослене пре 1. јануара 1980.

1. Select Prezime, Ime, DatZap from Radnik;
2. Select Ime, Prezime, DatZap from Radnik where DatZap < #1.1.1980#;
3. SELECT PREZIME, IME, DATZAPFROM RADNIK where datzap>#1/1/1980#;
4. Select Prezime, Ime, DatZap from Radnik where DatZap<#1/1/1980#;

414. Дата је табела RADNIK:

IDBR	IME	PREZIME	PLATA	PREMIJA	DATZAP
5900	Slobodan	Golubović	3900	1300	3.10.1978
5932	Mitar	Gavrilović	2600		25.3.1985
5953	Persida	Kosanović	1100	0	12.1.1979
6234	Marko	Pavlović	1300	3000	17.12.1990
6789	Janko	Nikolić	3900	10	31.12.1995

Извршава се упит:

SELECT ime, prezime, plata **FROM** radnik
WHERE plata = (SELECT MAX(plata) **FROM** radnik);

Одредити резултат извршења упита:

1.	IME	PREZIME	PLATA
	Slobodan	Golubović	3900
	Mitar	Gavrilović	2600
2.	IME	PREZIME	PLATA
	Slobodan	Golubović	3900
	Janko	Nikolić	3900
3.	IME	PREZIME	PLATA
	Slobodan	Golubović	3900
4.	IME	PREZIME	PLATA
	Janko	Nikolić	3900

1

415. Дата је табела RADNIK:

IDBR	IME	PREZIME	PLATA	PREMIJA	DATZAP
5900	Slobodan	Golubović	3900	1300	3.10.1978
5932	Mitar	Gavrilović	2600		25.3.1985
5953	Persida	Kosanović	1100	0	12.1.1979
6234	Marko	Pavlović	1300	3000	17.12.1990
6789	Janko	Nikolić	3900	10	31.12.1995

Извршава се упит:

SELECT ime, prezime, plata+premija AS zarada **FROM** radnik
WHERE premija is not NULL and premija > 10;

Одредити шта је резултат извршења упита:

1.	IME	PREZIME	ZARADA
	Slobodan	Golubović	5200
	Marko	Pavlović	4300
2.	IME	PREZIME	ZARADA
	Slobodan	Golubović	3900
	Marko	Pavlović	3000
3.	IME	PREZIME	ZARADA
	Mitar	Gavrilović	2600
4.	IME	PREZIME	ZARADA
	Persida	Kosanović	1100
	Janko	Nikolić	3900

1

416. Дата је табела RADNIK:

IDBR	IME	PREZIME	PLATA	PREMIJA	DATZAP
5900	Slobodan	Golubović	3900	1300	3.10.1978
5932	Mitar	Gavrilović	2600		25.3.1985
5953	Persida	Kosanović	1100	0	12.1.1979
6234	Marko	Pavlović	1300	3000	17.12.1990
6789	Janko	Nikolić	3900	10	31.12.1995

Извршава се упит:

SELECT ime, prezime, plata+premija AS zarada **FROM** radnik
WHERE premija is not NULL;

Одредити колико редова има табела која је резултат извршења упита не рачунајући заглавље:

1. 0
2. 2
3. 4
4. 5

1

417. Дата је табела RADNIK и табела ODELJENJE које су међусобно повезане пољем **BROD** (број одељења).

IDBR	IME	PREZIME	PLATA	BROD
5900	Slobodan	Golubović	3900	10
5932	Mitar	Gavrilović	2600	10
5953	Persida	Kosanović	1100	20
6234	Marko	Pavlović	1300	
6789	Janko	Nikolić	3900	10

BROD	Naziv	Mesto
50	Skladišta	Zemun
30	Marketing	Vračar
10	Plasman	Surčin
20	Direkcija	Grocka
40	Nabavka	Barajevo

Извршава се упит:

```
SELECT Naziv
FROM Odeljenje
WHERE brod NOT IN (SELECT brod FROM Radnik);
```

Одредити шта се приказује након извршења упита:

1. Називи свих одељења
2. Називи одељења у којима су распоређени радници
3. Називи одељења у којима нису распоређени радници
4. Назив одељења у ком има највише радника

1

418. Дата је табела KUPCI:

Id	Prezime	Adresa	Mesto	PostBroj	Status
1	Antić	Mokroluška 35	Bečej	21220	Aktivan
2	Kostić	Jug Bogdanova 12	Niš	18000	Pasivan
3	Marić	Masarikova 1	Pirot	18300	Pasivan
4	Perić	Takovska 67	Niš	18000	Pasivan
5	Rakić	Konstantinova 22	Sombor	25000	Aktivan
6	Tomić	Kneza Miloša 13	Bečej	21220	Aktivan

у којој поља **AdresaPostBrojStatus** могу бити неодређена.

Извршава се упит:

```
INSERT INTO KUPCI (Id, Prezime, Mesto)
VALUES (9,'Andrejić','Ada');
```

Одредити резултат рада овог упита:

1. Добићемо поруку о грешци
2. У табелу ће бити додат један запис
3. Табела се неће променити
4. Додате вредности нису правилне

1

419. Дата је табела KUPCI:

Id	Prezime	Adresa	Mesto	PostBroj	Status
1	Antić	Mokroluška 35	Bečej	21220	Aktivan
2	Kostić	Jug Bogdanova 12	Niš	18000	Pasivan
3	Marić	Masarikova 1	Pirot	18300	Pasivan
4	Perić	Takovska 67	Niš	18000	Pasivan
5	Rakić	Konstantinova 22	Sombor	25000	Aktivan
6	Tomić	Kneza Miloša 13	Bečej	21220	Aktivan

Извршава се упит:

DELETE FROM KUPCI WHERE Mesto Like “*?o?*”;

Одредити вредности **Id** оних записа који ће бити уклоњени из табеле након извршења датог упита:

- 1,2
- 2,3
- 3,5**
- 1,6

1

420. Дата је табела KUPCI:

Id	Prezime	Adresa	Mesto	PostBroj	Status
1	Antić	Mokroluška 35	Bečej	21220	Aktivan
2	Kostić	Jug Bogdanova 12	Niš	18000	Pasivan
3	Marić	Masarikova 1	Pirot	18300	Pasivan
4	Perić	Takovska 67	Niš	18000	Pasivan
5	Rakić	Konstantinova 22	Sombor	25000	Aktivan
6	Tomić	Kneza Miloša 13	Bečej	21220	Aktivan

Извршава се упит:

CREATE TABLE NoviKupci

(Id number PrimaryKey, Prezime text(50), Adresa Text(50), Mesto Text(20), PostBroj Number, Status Text(8) is Null);

Одредити резултат извршења упита:

- Креира нову табелу NoviKupci**
- У нову табелу **NoviKupci** преписује све записе из табеле **KUPCI**
- Уклања табелу **KUPCI** и замењује је табелом **NoviKupci**
- Не креира нову табелу **NoviKupci**, јер већ постоји табела **KUPCI**

1

421.	<p>Упит: CREATE TABLE NoviKupci (Id number PrimaryKey, Prezime text(50), Adresa Text(50), Mesto Text(20), PostBroj Number, Status Text(8) is Null);</p> <p>спада у:</p> <ol style="list-style-type: none"> 1. Параметарске упите 2. Акционе упите 3. Селекционе упите 4. Агрегатне упите 	1
422.	<p>Дате су табеле: KUPCI, NoviKupci следеће (једнаке) структуре: (Id number PrimaryKey, Prezime text(50), Adresa Text(50), Mesto Text(20), PostBroj Number, Status Text(8) is Null); :</p> <p>Извршава се упит: INSERT INTO NoviKupci SELECT *FROM KUPCI WHERE Status = 'Aktivan';</p> <p>Одредити резултат извршења упита:</p> <ol style="list-style-type: none"> 1. У табелу NoviKupci уписује све записе из табеле KUPCI 2. У табелу NoviKupci уписује само записе из табеле KUPCI са задатим статусом 3. У табелу KUPCI уписује све записе из табеле NoviKupci 4. У табелу KUPCI уписује само записе из табеле NoviKupci са задатим статусом 	1
423.	<p>Дат је упит:</p> <p>SELECT * FROM ucenici WHERE odeljenje=4 OR odeljenje=7 OR odeljenje=10</p> <p>Изабрати оператор који би требало користити у датом упиту да би избегли вишеструко коришћење оператора OR:</p> <ol style="list-style-type: none"> 1. LIKE 2. BETWEEN 3. AND 4. IN 	1
424.	<p>Заокружити број испред понуђеног одговора који представља број вредности које може да врати подупит уколико је испред њега постављен оператор једнакости (=), а да при том упит у коме се налази тај подупит функционише.</p> <ol style="list-style-type: none"> 1. Неограничено 2. Само једну 3. До две 4. До пет 	1
425.	<p>Изабрати исказ о подупитима који је тачан:</p> <ol style="list-style-type: none"> 1. Подупит се пише под наводницима. 2. Подупит не сме да садржи групне функције. 3. Подупит се често користи у WHERE клаузули да врати вредности које испуњавају одређени услов. 4. Подупит се најчешће извршава последњи, након што се изврши главни (спољни) упит. 	1

426. Табела **Zaposleni** садржи поља: `zaposleni_id`, `ime`, `prezime`, `plata`, `odsek_id`.

Дат је упит:

```
SELECT zaposleni_id, ime
FROM zaposleni
WHERE plata=(SELECT MAX(plata) FROM zaposleni GROUP BY odsek_id)
```

Одредити разлог због кога дати упит не ради:

1. Подупит се пише под наводницима.
2. Подупит не сме користити групне функције.
3. Подупит враћа више од једне врсте, а коришћен је оператор за поређење са једном вредношћу.
4. Подупит може користити само колоне које се налазе у спољњем упиту иза речи SELECT.

1

427. Дате су табеле: **KUPCI**, **NoviKupci** следеће (једнаке) структуре:
(`Id number PrimaryKey`, `Prezime text(50)`, `Adresa Text(50)`, `Mesto Text(20)`, `PostBroj Number`, `Status Text(8) is Null`); :

Извршава се упит:

```
INSERT INTO NoviKupci
SELECT *FROM KUPCI WHERE Status <> 'Aktivan';
```

Одредити шта је резултат извршења датог упита:

1. У табелу **NoviKupci** уписује само записе из табеле **KUPCI** са статусом који је једнак **Aktivan**
2. У табелу **NoviKupci** уписује само записе из табеле **KUPCI** са статусом који није једнак **Aktivan**
3. У табелу **KUPCI** уписује све записе из табеле **NoviKupci**
4. У табелу **KUPCI** уписује само записе из табеле **NoviKupci** са статусом који није једнак **Aktivan**

1

428. Табела **Zaposleni** садржи поља: `zaposleni_id`, `ime`, `prezime`, `plata`, `odsek_id`.

Потребно је приказати ид одсека, најмању и највећу плату у сваком одсеку.

Заокружити број испред упита који одговара претходном захтеву:

1. SELECT `odsek_id`, MIN(`plata`), MAX(`plata`)
FROM `zaposleni` HAVING `odsek_id`
2. SELECT `odsek_id`, MIN(`plata`), MAX(`plata`)
FROM `zaposleni` GROUP BY `odsek_id`
3. SELECT `odsek_id`, MIN(`plata`), MAX(`plata`)
FROM `zaposleni`
WHERE `odsek=odsek_id`
4. SELECT MIN(`plata`), MAX(`plata`)
FROM `zaposleni` GROUP BY `odsek_id`

1

429.	<p>Заокружити број испред назива споја који је потребно користити да би приказали сви редови из табела Zaposleni и Istorija_zaposlenih.</p> <ol style="list-style-type: none"> 1. Унутрашњи спој 2. Десни спољашњи спој 3. Леви спољашњи спој 4. Потпуни спољашњи спој 	1
430.	<p>Заокруживањем редног броја испред понуђеног одговора, изабрати исказ који најбоље описује природно спајање табела.</p> <ol style="list-style-type: none"> 1. Спој између две табеле које имају колоне са истим именом, типом података и дужином. 2. Спој између две табеле који као резултат даје Декартов производ. 3. Спој између две табеле код којих не постоје поља која имају исто име, тип и дужину. 4. Спој који користи само једну табелу. 	1
431.	<p>Одабрати клаузулу коју је потребно користити уколико листа иза резервисане речи SELECT садржи колону и групну функцију:</p> <ol style="list-style-type: none"> 1. HAVING клаузулу 2. GROUP BY клаузулу 3. JOIN клаузулу 4. ORDER BY клаузулу 	1
432.	<p>Табела Zaposleni садржи поља: <i>Zaposleni_Id</i>, <i>Ime</i>, <i>Prezime</i>, <i>Plata</i>, <i>Odsek_Id</i> и незавршени упит:</p> <pre>SELECT COUNT(<i>Odsek_Id</i>), <i>Ime</i>, <i>Prezime</i> FROM <i>Zaposleni</i> GROUP BY?????</pre> <p>Одредити чиме је потребно заменити ????? да би упит био исправан. Заокружити број испред тачног одговора:</p> <ol style="list-style-type: none"> 1. <i>Odsek_Id</i> 2. <i>Odsek_Id</i>, <i>Ime</i> 3. <i>Odsek_Id</i>, <i>Ime</i>, <i>Prezime</i> 4. <i>Ime</i>, <i>Prezime</i> 	1
433.	<p>Извршава се следећа SELECT наредба:</p> <pre>SELECT MAX(<i>plata</i>), <i>odsek_id</i> FROM <i>zaposleni</i> GROUP BY <i>odsek_id</i></pre> <p>Означити које ће вредности бити приказане:</p> <ol style="list-style-type: none"> 1. Највећа плата коју има запослени у фирми. 2. Највећа плата у сваком одсеку. 3. Запослени који има највећу плату. 4. Запослени који имају највећу плату у сваком одсеку. 	1

434.	<p>Извршава се следећа SELECT наредба: SELECT MIN(datum_zaposlenja), odsek_id FROM zaposleni GROUP BY odsek_id</p> <p>Одредити које ће вредности бити приказане:</p> <ol style="list-style-type: none"> 1. Најранији датум запослења за сваки одсек предузећа. 2. Најранији датум запослења у целој табели ZAPOSLENI. 3. Датум запослења последњег запосленог радника у целом предузећу. 4. Датум запослења најстаријег запосленог радника у целом предузећу. 5. Датум запослења најстаријег запосленог радника у сваком одсеку предузећа. 	1
435.	<p>Означити функцију која враћа датум и време на рачунару који је сервер базе података:</p> <ol style="list-style-type: none"> 1. DATE 2. DATETIME 3. SYSDATE 4. CURRENTDATE 	1
436.	<p>Означити оператор поређења који захтева листу вредности:</p> <ol style="list-style-type: none"> 1. IN 2. LIKE 3. BETWEEN ... IN ... 4. IS NULL 	1
437.	<p>Означити шта се дешава када се креира Декартов производ:</p> <ol style="list-style-type: none"> 1. Табела се спаја са собом, свака колона са сваком колоном. 2. Табела се спаја са другом једнаком табелом. 3. Све врсте једне табеле се спајају са свим врстама друге табеле. 4. Све врсте које се не слажу у WHERE клаузули се приказују. 	1
438.	<p>Одредити реченицу која је тачна о оператору ANY који се примењује са подупитом који враћа више вредности:</p> <ol style="list-style-type: none"> 1. Оператор ANY може да се користи са резервисаном речју DISTINCT. 2. Оператор ANY је синоним за оператор ALL. 3. Оператор ANY упоређујес са свим вредностима које враћа подупит. 4. Оператор ANY може да се користи са операторима LIKE и IN. 	1

439. Дата је табела **ARTIKLI**:

Id	Naziv	DatProiz	JM	Kol	Cena	Akcija
21	Hleb Sava	15.04.2015	kom	100	35	
32	Hleb Dunav	15.04.2015	kom	50	45	0.05
23	Baget beli	12.04.2015	kom	20	32	0.12
47	Baget crni	12.04.2015	kom	12	35	0.08
25	Tost tamni	10.04.2015	kom	6	75	0.15
36	Tost beli	10.04.2015	kom	8	70	
28	Kifla integralna	15.04.2015	kom	15	18	0
10	Brašno Graham	10.03.2015	kg	24	54	0

Потребно је приказати називе, цене и количине за све артикле чији назив **не почиње** са Hleb, Baget или Tost.

Заокружити број испред упита који одговара постављеном захтеву.

1. SELECT Naziv, Kol, Cena FROM ARTIKLI;
2. SELECT Naziv, Kol, Cena FROM ARTIKLI where Naziv not in ('Hleb*', 'Baget*', 'Tost*');
3. SELECT Naziv, Kol, Cena FROM ARTIKLI where Naziv not like 'Hleb*' and Naziv not like 'Tost*' and Naziv not like 'Baget*';
4. SELECT Naziv, Kol, Cena FROM ARTIKLI where Naziv not like ('Hleb*', 'Baget*', 'Tost*');

1

440. Дата је табела **ARTIKLI** и упит:

Id	Naziv	DatProiz	JM	Kol	Cena	Akcija
21	Hleb Sava	15.04.2015	kom	100	35	
32	Hleb Dunav	15.04.2015	kom	50	45	0.05
23	Baget beli	12.04.2015	kom	20	32	0.12
47	Baget crni	12.04.2015	kom	12	35	0.08
36	Tost beli	10.04.2015	kom	8	70	
28	Kifla integralna	15.04.2015	kom	15	18	0
10	Brašno Graham	10.03.2015	kg	24	54	0

SELECT Naziv, Kol, Cena, DateDiff("d", DatProiz, "2015-04-18") FROM ARTIKLI;

Одредити шта ће бити приказано након извршења упита:

1. Називи, цене и количине за све артикле
2. Називи, цене и количине за све артикле који су произведени пре 18.04.2015.
3. Називи, цене, количине и број дана од датума производње до 18.04.2015. за све артикле
4. Називи, цене, количине и број дана од датума производње до дана истека рока за све артикле

1

441. Дата је табела **RADNIK** и упит:

IDBR	IME	PREZIME	PLATA	PREMIJA	DATZAP
6234	Marko	Pavlović	1300	3000	17.12.1990
6789	Janko	Nikolić	3900	10	23.12.1999

SELECT ime, prezime, DateDiff("yyyy", datzap, now()) as god from radnik;

Одредити шта је резултат упита:

1. Табела са подацима о именима и презименима радника
2. Табела са подацима о именима, презименима и броју година које су протекле од датума запослења радника до краја века
3. Табела са подацима о именима, презименима и датумима запослења радника
4. Табела са подацима о именима, презименима и броју година које су протекле од датума запослења радника до тренутног датума

1

442. Дата је табела **RADNIK (IDBR, prezime, ime, plata, datZap, BROD)** и упит:

**SELECT prezime, ime, posao, plata
FROM Radnik WHERE ime LIKE @name**

Одредити групу којој припада дати упит:

1. Параметарски упити
2. Групни упити
3. Акциони упити
4. Агрегатни упити

1

443. Одредити оператор поређења који се може користити у упиту који има подупит који враћа више врста:

1. IN, ANY и ALL
2. LIKE
3. BETWEEN... AND ...
4. =, < и >

1

444. Извршава се SELECT упит који у себи има подупит:

**SELECT prezime, ime
FROM ucenici
WHERE odeljenje_id IN
(SELECT odeljenje_id
FROM podela
WHERE nastavnik_id=20)**

Одредити реченицу о датом подупиту која је тачна:

1. Спољни упит се извршава пре угњеженог подупита.
2. Резултати унутрашњег упита се враћају спољном упиту.
3. Јавља се грешка ако и унутрашњи и спољни упит не враћају вредности.
4. Оба упита (и спољашњи и унутрашњи) морају вратити вредност да не би дошло до грешке.

1

445.	<p>Одредити реченицу која најбоље описује значење ANY оператора:</p> <ol style="list-style-type: none"> 1. Једнак било ком члану листе. 2. Упоредиће вредност са сваком вредношћу коју враћа подупит. 3. Упоредиће вредност са првом вредности коју враћа подупит. 4. Једнак свакој вредности у листи. 	1
446.	<p>Потребно је креирати извештај који приказује имена свих производа чија је цена већа од просечне цене свих производа:</p> <p>Заокружити број испред упита који одговара постављеном задатку:</p> <ol style="list-style-type: none"> 1. SELECT naziv FROM proizvod WHERE cena > (SELECT AVG(cena) FROM proizvod) 2. SELECT naziv FROM proizvod WHERE cena > AVG(cena) 3. SELECT AVG(cena), naziv FROM proizvod WHERE cena > AVG(cena) GROUP BY naziv 4. SELECT naziv FROM (SELECT AVG(cena) FROM proizvod) WHERE cena > AVG(cena) 	1
447.	<p>Табела ARTIKLI садржи следеће колоне: artikl_id, naziv, kategorija, cena, kolicina. Потребно је да се прикаже категорија и минимална цена артикла у свакој категорији.</p> <p>Изабрати упит који даје тражени извештај:</p> <ol style="list-style-type: none"> 1. SELECT kategorija, MIN(cena) FROM artikli GROUP BY cena 2. SELECT kategorija, MIN(cena) FROM artikli GROUP BY kategorija 3. SELECT MIN(cena), kategorija FROM artikli GROUP BY MIN(cena), kategorija 4. SELECT cena, MIN(kategorija) FROM artikli GROUP BY kategorija 	1
448.	<p>Уколико подупит формираног упита враћа једну врсту, заокружити број испред оператора који се може користити у таквом упиту</p> <ol style="list-style-type: none"> 1. LIKE 2. ANY 3. ALL 4. IN 	1

449.	<p>Табела RADIONICA садржи следеће колоне: radinica_id, naziv, zanat, lokacija_id. Потребно је да се прикаже број радионица са јединственим занатом на свакој локацији.</p> <p>Изабрати упит који даје тражени извештај:</p> <ol style="list-style-type: none"> 1. SELECT location_id, COUNT(DISTINCT zanat) FROM radionica 2. SELECT location_id, COUNT(zanat) FROM radionica GROUP BY lokacija_id 3. SELECT location_id, COUNT(DISTINCT zanat) FROM radionica GROUP BY lokacija_id 4. SELECT location_id, COUNT(DISTINCT zanat) FROM radionica GROUP BY zanat 	1
450.	<p>Одредити број спојева (join) који се морају применити за спајање пет табела:</p> <ol style="list-style-type: none"> 1. 3 2. 4 3. 5 4. Један више од броја табела 	1
451.	<p>Изабрати назив могућности SQL наредбе да враћа податке из две или више табела :</p> <ol style="list-style-type: none"> 1. Селекција 2. Пројекција 3. Спајање 4. Разврставање 	1
452.	<p>Изабрати у ком делу SQL наредбе се могу користити аритметичке операције:</p> <ol style="list-style-type: none"> 1. SELECT 2. FROM 3. WHERE 4. ORDER BY 	1

453. Табела **RADNIK** садржи следеће колоне: radnik_id, ime, prezime, odsek_id, plata. Потребно је да се прикаже просечна зарада запослених у сваком одсеку.

Изабрати упит који даје тражени извештај:

1.

```
SELECT plata(AVG)
FROM radnik
GROUP BY odsek_id
```
2.

```
SELECT AVG plata
FROM radnik
GROUP BY odsek_id
```
3.

```
SELECT AVG(plata)
FROM radnik
BY odsek_id
```
4.

```
SELECT AVG(plata)
FROM radnik
GROUP BY odsek_id
```

1

454. Табела **PROIZVOD** садржи следеће колоне: proizvod_id, opis, cena, lokacija_id. Потребно је приказати оне производе који задовољавају следеће услове:

- **cena** сваког приказаног производа приказује се увећана за 10%.
- **lokacija_id** мора бити 2000, 2001 или 2002.
- 10% оригиналне цене мора бити мање од 100 динара.
-

Проценити који упит даје тражене податке:

1.

```
SELECT cena, proizvod_id, cena*1.10
FROM proizvod
WHERE cena*0.10<100,00
AND lokacija_id IN (2000, 2001, 2002)
```
2.

```
SELECT cena, proizvod_id, cena*0.10
FROM proizvod
WHERE cena*1.10>100,00
AND (lokacija_id IN (2000, 2001, 2002))
```
3.

```
SELECT cena, proizvod_id, cena*1.10
FROM proizvod
WHERE cena*1.10<100,00
AND lokacija_id = (2000, 2001, 2002)
```
4.

```
SELECT cena, proizvod_id, cena*1.10
FROM proizvod
WHERE cena*0.10>100,00
AND (lokacija_id IN (2000, 2001, 2002))
```

1

455. Одредити шта ће се десити ако се користи DELETE наредба без WHERE клаузуле:

1. Све врсте у табели ће бити обрисане.
2. Појавит ће се грешка.
3. Ни једна врста неће бити обрисана.
4. Само ће једна врста бити обрисана.

1

456.	Извршава се наредба:	1
	DELETE * FROM ucenik.	
	Одредити шта ће се десити са табелом UCENIK:	
	<ol style="list-style-type: none"> 1. Биће обрисана табела UCENIK. 2. Биће обрисана колона UCENIK. 3. Све врсте у табели UCENIK ће бити обрисане. 4. Само ће прва врста у табели UCENIK бити обрисана. 	
457.	Табела ZAPOSLENI има следећа поља (zaposeleni_id, ime, prezime, plata, odsek_id, bonus). Потребно је доделити бонус свим запосленим у сектору 10, Вредност бонуса поставити на исти бонус који је додељен запосленом чији је id 1221:	1
	<ol style="list-style-type: none"> 1. UPDATE zaposleni SET bonus = SELECT bonus FROM zaposleni WHERE zaposeleni_id = 1221 	
	<ol style="list-style-type: none"> 2. UPDATE zaposleni SET bonus = (SELECT bonus FROM zaposleni WHERE zaposeleni_id = 1221) 	
	<ol style="list-style-type: none"> 3. UPDATE zaposleni SET bonus = (SELECT bonus FROM zaposleni WHERE zaposeleni_id = 1221) WHERE odsek_id = 10 	
	<ol style="list-style-type: none"> 4. UPDATE zaposleni SET bonus = (SELECT bonus FROM zaposleni WHERE zaposeleni_id = 1221 WHERE odsek_id = 10) 	
458.	Извршава се наредба:	1
	SELECT COUNT(*) FROM ucenik.	
	Одредити шта ради ова наредба:	
	<ol style="list-style-type: none"> 1. Јавит ће се грешка, јер нема WHERE клаузуле. 2. Биће приказан број врста у табели UCENIK. 3. Биће приказани сви подаци о ученицима из табеле, а након тога приказат ће се и њихов број. 4. Биће приказани сви подаци о ученицима. 	

459.	<p>Извршава се упит:</p> <pre>SELECT prezime, ime, email FROM ucenik ORDER BY prezime WHERE prosek >= 4.50</pre> <p>Наредба се неће извршити.</p> <p>Заокружити због чега се наредба неће извршити:</p> <ol style="list-style-type: none"> 1. Наредба се неће извршити једино ако нема ни једног одличног ученика. 2. Услов није добро написан. 3. Потребно је убацити HAVING клаузулу. 4. Потребно је променити редослед клаузула. 	1
460.	<p>Изабрати где се пише ORDER BY клаузула:</p> <ol style="list-style-type: none"> 1. Увек на крају упита. 2. У листи иза SELECT наредбе. 3. Испред WHERE клаузуле. 	1
461.	<p>Извршава се следећа наредба:</p> <pre>SELECT ucenik_id, ime, prosek FROM ucenik ORDER BY ime, prosek</pre> <p>Изабрати шта се дешава када се наредба изврши:</p> <ol style="list-style-type: none"> 1. Списак ученика се сортира нумерички. 2. Списак ученика се сортира алфаветски. 3. Списак ученика се сортира најпре нумерички, а онда алфаветски. 4. Списак ученика се сортира најпре алфаветски, а онда нумерички. 	1
462.	<p>Изабрати која реченица о властитом споју (self join) је истинита:</p> <ol style="list-style-type: none"> 1. Алијаси табела се не могу користити за одређивање имена табеле. 2. Властити спој (self join) се може користити само уз помоћ погледа (view). 3. Алијаси табела се морају користити да би одредили имена табеле. 	1
463.	<p>Изабрати која реченица о клаузули GROUP BY је истинита:</p> <ol style="list-style-type: none"> 1. Груписање се најпре врши по првој колони у листи која се налази иза резервисаних речи GROUP BY. 2. Груписање се најпре врши по последњој колони у листи која се налази иза резервисаних речи GROUP BY. 3. GROUP BY клаузула се не може користити са ORDER BY клаузулом. 	1
464.	<p>Уколико је потребно табелу спојити саму са собом, тада је потребно користити:</p> <ol style="list-style-type: none"> 1. Алијасе табела. 2. Алијасе колона. 3. HAVING клаузулу. 4. GROUP BY клаузулу. 	1

465.	Изабрати наредбу која се користи да се уклони поглед (view):	
	<ol style="list-style-type: none"> 1. DROP 2. DELETE 3. DROP VIEW 4. REMOVE VIEW 	1
466.	Уколико се кроз поглед (view) жели да се промене подаци он не сме садржати:	
	<ol style="list-style-type: none"> 1. WHERE клаузулу 2. Спој 3. алијас колоне 4. GROUP BY клаузулу 	1
467.	Означити како се назива спајање свих врста једне табеле, са свим врстама друге табеле:	
	<ol style="list-style-type: none"> 1. упаривање 2. производ 3. Декартов производ 4. Декартово спајање 	1
468.	Означити тачан исказ:	
	<ol style="list-style-type: none"> 1. Окидачи (тригери) се углавном користе за контролу извршавања наредби SQL-а. 2. Окидачи (тригери) се углавном користе за чување базе података од вируса. 3. Окидачи (тригери) се углавном користе за очување интегритета података у бази података. 	1
469.	Изабрати тврђење које је истинито у вези ADO.NET-а:	
	<ol style="list-style-type: none"> 1. ADO.NET технологија представља модел састављен од скупа класа за приступ и манипулацију подацима који су смештени у различитим базама података. 2. ADO.NET је језик намењен приступу подацима који су смештени у различитим базама података. 3. ADO.NET технологија је скуп функција намењених коришћењу података на Интернету који су смештени у различитим базама података. 	1

470. Дата је табела **RADNIK**, табела **ODELJENJE** и упит:

IDBR	IME	PREZIME	PLATA	BROD
5900	Slobodan	Golubović	900	10
5932	Mitar	Gavrilović	600	10
5953	Persida	Kosanović	1100	20
6234	Marko	Pavlović	1300	30
6789	Janko	Nikolić	800	10

BROD	Naziv	Mesto
50	Skladišta	Zemun
30	Marketing	Vračar
10	Plasman	Surčin
20	Direkcija	Grocka
40	Nabavka	Barajevo

2

**SELECT imeod, avg(plata) AS ProsekPlata FROM Radnik, Odeljenje
WHERE Odeljenje.brod=Radnik.brod GROUP BY imeod HAVING avg(plata)>1000;**

Одредити резултат извршавања датог упита:

1. Приказују називи свих одељења и висина просечне плате у њима
2. Групишу по одељењима радници са платом већом од просечне плате
3. Приказују називи одељења и висина просечне плате у њима само за одељења у којима је просечна плата већа од 1000
4. Приказују бројеви свих одељења и висина просечне плате у њима

471. Дата је табела **RADNIK**, табела **ODELJENJE**:

IDBR	IME	PREZIME	PLATA	BROD
5900	Slobodan	Golubović	3900	10
5932	Mitar	Gavrilović	2600	10
5953	Persida	Kosanović	1100	20
6234	Marko	Pavlović	1300	
6789	Janko	Nikolić	3900	10

BROD	Naziv	Mesto
50	Skladišta	Zemun
30	Marketing	Vračar
10	Plasman	Surčin
20	Direkcija	Grocka
40	Nabavka	Barajevo

2

Извршава се упит:

**UPDATE Radnik SET radnik.brod = 40
WHERE radnik.brod is null;**

Одредите шта се остварује извршењем овог упита:

1. Распоређивање свих радника у одељење чији је **brod** 40
2. Брисање нераспоређених радника
3. Уклањање одељења чији је **brod** 40
4. Распоређивање нераспоређених радника у одељење чији је **brod** 40

472. Дата је табела **RADNIK**, табела **ODELJENJE**:

IDBR	IME	PREZIME	PLATA	BROD
5900	Slobodan	Golubović	3900	10
5932	Mitar	Gavrilović	2600	10
5953	Persida	Kosanović	1100	20
6234	Marko	Pavlović	1300	
6789	Janko	Nikolić	3900	10

BROD	Naziv	Mesto
50	Skladišta	Zemun
30	Marketing	Vračar
10	Plasman	Surčin
20	Direkcija	Grocka
40	Nabavka	Barajevo

2

Одредити ком типу упита припада упит којим се радници из одељења **Plasman**, прераспоређују у одељење **Nabavka**:

1. Параметарски упит
2. Агрегатни упит
3. Угњеждени упит
4. **Акциони упит**

473. Дата је табела **RADNIK**, табела **ODELJENJE**:

IDBR	IME	PREZIME	PLATA	BROD
5900	Slobodan	Golubović	3900	10
5932	Mitar	Gavrilović	2600	10
5953	Persida	Kosanović	1100	20
6234	Marko	Pavlović	1300	
6789	Janko	Nikolić	3900	10

BROD	Naziv	Mesto
50	Skladišta	Zemun
30	Marketing	Vračar
10	Plasman	Surčin
20	Direkcija	Grocka
40	Nabavka	Barajevo

2

Одредити упит којим се радници из одељења **Plasman**, прераспоређују у одељење **Nabavka**:

1. UPDATE Radnik SET brod = 10 WHERE brod=40;
2. **UPDATE Radnik SET brod = 40 WHERE brod=10;**
3. ALTER TABLE Radnik SET brod = 40 WHERE brod=10;
4. DROP TABLE Radnik SET brod = 10 WHERE brod=40;

474. Дата је табела **RADNIK**, табела **ODELJENJE** и упит:

IDBR	IME	PREZIME	PLATA	BROD
5900	Slobodan	Golubović	3900	10
5932	Mitar	Gavrilović	2600	10
5953	Persida	Kosanović	1100	20
6234	Marko	Pavlović	1300	
6789	Janko	Nikolić	3900	10

BROD	Naziv	Mesto
50	Skladišta	Zemun
30	Marketing	Vračar
10	Plasman	Surčin
20	Direkcija	Grocka
40	Nabavka	Barajevo

2

SELECT odeljenje.imeod, radnik.ime
from odeljenje left join radnik on radnik.brod = odeljenje.brod ;

Одредити шта се види као резултат датог упита. Приказ чине:

1. Називи свих одељења и имена радника у њима
2. Називи само одељења у којима нема радника
3. Називи само одељења у којима има радника са именима радника у њима
4. **Називи одељења у којима има радника са именима радника у њима, и називи одељења у којима нема радника**

<p>475. Одредити оператор поређења који се не може користити у упиту који има подупит који враћа више вредности:</p> <ol style="list-style-type: none"> 1. ALL 2. ANY 3. = 4. > 	2
<p>476. Табела ZAPOSLENI има следећа поља (zaposleni_id, ime, prezime, plata, odsek_id). Потребно је повећати плату запосленима у IT сектору 12.5%. IT сектор има као шифру одсека – 15. Изабрати коју наредбу је потребно употребити да би се урадила тражена измена у табели:</p> <ol style="list-style-type: none"> 1. UPDATE zaposleni(plata) SET plata=plata*1,125 2. UPDATE zaposleni SET plata=plata*0.125 WHERE odsek_id=15 3. UPDATE zaposleni SET plata=plata*1.125 WHERE odsek_id=15 4. UPDATE zaposleni (plata) VALUES (plata*0.125) WHERE odsek_id=15 	2
<p>477. Дат је упит за креирање погледа и наведени искази који се односе на дати упит. Заокружити број испред тачног исказа:</p> <p>CREATE VIEW Pregled_Proseka AS SELECT UcenikID, Ime, Prezime, AVG(Ocena) AS Prosek FROM Testovi WHERE OdeljenjeID IN (1, 2, 3, 4) GROUP BY UcenikID, Ime, Prezime;</p> <ol style="list-style-type: none"> 1. Подаци у табели Testovi се могу модификовати коришћењем погледа Pregled_Proseka 2. Коришћењем датог погледа, подаци се могу само у додавати у табелу Testovi, али не и мењати 3. Овако дат упит изазива грешку при извршењу 4. Коришћењем датог погледа, подаци из табеле Testovi се могу само прегледавати, али не и додавати или мењати 	2

У следећим задацима заокружите бројеве испред тражених одговора

478.	Одредити својство података које база података не може да обезбеди:	
	<ol style="list-style-type: none"> 1. Исправност 2. Истинитост 3. Потпуност 4. Заштићеност 	1
479.	Одабрати све могуће инстанце ентитета ОСОБА:	
	<ol style="list-style-type: none"> 1. Новак Ђоковић 2. име и презиме 3. Никола Тесла 4. женско 5. датум рођења 	1
480.	За измену – ажурирање структуре табеле која се налази у оквиру базе података користи се:	
	<ol style="list-style-type: none"> 1. ALTER – ADD – за додавање колоне у табелу 2. ALTER – DROP – за брисање – уклањање колоне из табеле 3. ALTER – INSERT – за додавање колоне у табелу 4. ALTER – DELETE – за брисање – уклањање колоне из табеле 	1
481.	За упите са специфицираним редоследом приказа врста у резултујућој табели користи се клаузула ORDER BY после које се наводи назив колоне:	
	<ol style="list-style-type: none"> 1. и службена реч ASCENDING за растући редослед 2. и службена реч DESCENDING за опадајући редослед 3. и службена реч ASCENDING за опадајући редослед 4. и службена реч DESCENDING за растући редослед 5. службена реч се може изоставити, добија се растући поредак 6. службена реч се може изоставити, добија се опадајући поредак 	1,5
482.	Одредити компоненте информатичких система поред харвера, софтвера и рачунарских мрежа:	
	<ol style="list-style-type: none"> 1. подаци 2. језик за обраду података 3. базна компонента 4. мрежна компонента 	2
483.	Делови манипулативне компоненте релационог система су:	
	<ol style="list-style-type: none"> 1. Релациона алгебра 2. Релационе функције 3. Релациони рачун 4. Релациони типови 	2

484. Одредити ентитете који садрже одговарајуће атрибуте:

1. Ентитет: КЊИГА – Атрибути: наслов, аутор, издавач, година издања
2. Ентитет: АУТОМОБИЛ – Атрибути: марка, година производње, боја, власник, година рођења власника, регистарски број
3. Ентитет: УЧЕНИК – Атрибути: име, презиме, разред, одељење, број оправданих, број неоправданих, просек
4. Ентитет: ДРЖАВА – Атрибути: назив, број становника, површина, главни град, број становника главног града, име градоначелника главног града

2

485. Одредити ентитете који садрже атрибуте, који НИСУ одговарајући:

1. Ентитет: КЊИГА – Атрибути: наслов, аутор, издавач, година издања
2. Ентитет: АУТОМОБИЛ – Атрибути: марка, година производње, боја, власник, година рођења власника, регистарски број
3. Ентитет: УЧЕНИК – Атрибути: име, презиме, разред, одељење, број оправданих, број неоправданих, просек
4. Ентитет: ДРЖАВА – Атрибути: назив, број становника, површина, главни град, број становника главног града, име градоначелника главног града
5. ЕНТИТЕТ: САЈАМ – Атрибути: назив, датум почетка, датум завршетка, број излагача, број посетилаца

2

486. Поребно је у табелу под именом RADNIK додати примарни кључ - прогласити примарним кључем колону IDBR.

1. ALTER TABLE RADNIK
ADD CONSTRAINT PK_RADNIK PRIMARY KEY (IDBR);
2. ALTER TABLERADNIK
ADD PRIMARY KEY (IDBR);
3. CREATE TABLERADNIK
ADD PK_RADNIK PRIMARY KEY (IDBR);
4. DROPTABLERADNIK
ADD PK_RADNIK PRIMARY KEY (IDBR);

2

487. Дата је табела RADNIK:

IDBR	IME	PREZIME	PLATA	PREMIJA	DATZAP
5900	Slobodan	Golubović	900	1300	3.10.1978
5932	Mitar	Gavrilović	2600		25.3.1985
5953	Persida	Kosanović	1100	0	12.1.1979
6234	Marko	Pavlović	1300	3000	17.12.1990
6789	Janko	Nikolić	3900	10	31.12.1995

2

Заокружити бројеве испред упита којим ће се приказати само презиме и име за раднике запослене у периоду од 1. јануара 1980. до 31. децембра 1995.

1. Select Prezime, Ime from Radnik where DatZap > #1.1.1980#; and < #1.1.1995#;
2. Select Prezime, Ime from Radnik where DatZap <= #1.1.1980# and DatZap <= #31.12.1995#;
3. SELECT PREZIME, IME, DATZAPFROM RADNIK where datzap between #1/1/1980# and #31/12/1995#;
4. Select Prezime, Ime from Radnik where datzap between #1/1/1980# and #1/1/1995#;

488. Дата је табела **RADNIK**, табела **ODELJENJE**:

IDBR	IME	PREZIME	PLATA	BROD
5900	Slobodan	Golubović	3900	10
5932	Mitar	Gavrilović	2600	10
5953	Persida	Kosanović	1100	20
6234	Marko	Pavlović	1300	
6789	Janko	Nikolić	3900	10

BROD	Naziv	Mesto
50	Skladišta	Zemun
30	Marketing	Vračar
10	Plasman	Surčin
20	Direkcija	Grocka
40	Nabavka	Barajevo

Дат је упит:

UPDATE Radnik SET radnik.brod = 50

WHERE radnik.brod=(Select brod from odeljenje where odeljenje.imeod = 'Plasman');

Одредити којим типовима упита припада дати упит:

1. Параметарски упит
2. Агрегатни упит
3. Угњеждени упит
4. **Акциони упит**

2

489. Изабрати које две команде се могу користити за измену постојећих података у бази:

1. DELETE
2. **MERGE**
3. SELECT
4. **UPDATE**

2

490. Одредити које кључне речи се могу користити када се креира поглед (view):

1. **HAVING**
2. **WHERE**
3. OPTION
4. INTO

2

491. Изабрати главне компоненте ADO.NET технологије:

1. Access
2. **.NET Framework data providers**
3. **DataSet**
4. Office

2

Допуните следеће реченице и табеле

492. Систем базе података у ужем смислу састоји се из базе података и _____ система за управљање базом података.

1

493. Навести средство којим корисник остварује комуникацију са релационом базом података: _____ релациони упитни језик.

1

494. Системи за управљање базом података данас имају углавном _____ клијент-сервер архитектуру.

1

495. Релациони модел података представља базу података као скуп _____ табела(или међусобно повезаних релацијама).

1

496.	Уколико су сви атрибути релације елементарни и не могу се раставити, кажемо да је релација у <u>prvoj</u> нормалној форми.	1																																																
497.	Уколико сви атрибути релације који нису део кључа, зависе од сваког атрибута који је део кључа, кажемо да је релација у <u>drugoј</u> нормалној форми.	1																																																
498.	Уколико сви атрибути релације који нису део кључа, не зависе од неког атрибута који није део кључа, кажемо да је релација у <u>trecoј</u> нормалној форми.	1																																																
499.	Уколико је ентитет егзистенцијално зависан од неког ентитета са којим је у вези, онда њега називамо <u>slabi</u> ентитет.	1																																																
500.	<p>Дата је табела RADNIK, табела ODELJENJE:</p> <table><tr><th>IDBR</th><th>IME</th><th>PREZIME</th><th>PLATA</th><th>BROD</th></tr><tr><td>5900</td><td>Slobodan</td><td>Golubović</td><td>3900</td><td>10</td></tr><tr><td>5932</td><td>Mitar</td><td>Gavrilović</td><td>2600</td><td>10</td></tr><tr><td>5953</td><td>Persida</td><td>Kosanović</td><td>1100</td><td>20</td></tr><tr><td>6234</td><td>Marko</td><td>Pavlović</td><td>1300</td><td></td></tr><tr><td>6789</td><td>Janko</td><td>Nikolić</td><td>3900</td><td>10</td></tr></table> <table><tr><th>BROD</th><th>Naziv</th><th>Mesto</th></tr><tr><td>50</td><td>Skladišta</td><td>Zemun</td></tr><tr><td>30</td><td>Marketing</td><td>Vračar</td></tr><tr><td>10</td><td>Plasman</td><td>Surčin</td></tr><tr><td>20</td><td>Direkcija</td><td>Grocka</td></tr><tr><td>40</td><td>Nabavka</td><td>Barajevo</td></tr></table> <p>Извршава се упит: UPDATE Radnik SET radnik.brod = 50 WHERE radnik.brod=(Select brod from odeljenje where odeljenje.imeod = 'Plasman');</p> <p>Допуните реченицу:</p> <p>Извршењем упита ажурирају се подаци у записима за <u>3</u> радника. 3 zapisa se azuriraju</p>	IDBR	IME	PREZIME	PLATA	BROD	5900	Slobodan	Golubović	3900	10	5932	Mitar	Gavrilović	2600	10	5953	Persida	Kosanović	1100	20	6234	Marko	Pavlović	1300		6789	Janko	Nikolić	3900	10	BROD	Naziv	Mesto	50	Skladišta	Zemun	30	Marketing	Vračar	10	Plasman	Surčin	20	Direkcija	Grocka	40	Nabavka	Barajevo	1
IDBR	IME	PREZIME	PLATA	BROD																																														
5900	Slobodan	Golubović	3900	10																																														
5932	Mitar	Gavrilović	2600	10																																														
5953	Persida	Kosanović	1100	20																																														
6234	Marko	Pavlović	1300																																															
6789	Janko	Nikolić	3900	10																																														
BROD	Naziv	Mesto																																																
50	Skladišta	Zemun																																																
30	Marketing	Vračar																																																
10	Plasman	Surčin																																																
20	Direkcija	Grocka																																																
40	Nabavka	Barajevo																																																
501.	Поглед је <u>virtuelna</u> (изведена) табела, чији се садржај генерише на захтев <u>upita</u> .	2																																																
502.	Тригери (окидачи) се могу стартовати непосредно <u>pre</u> или <u>posle</u> SQL наредбе.	2																																																
503.	Навести компоненте релационог модела: <u>strukturna</u> , <u>manipulativna</u> и <u>integritetna</u> .	3																																																

У следећим задацима уредите и повежите појмове према захтеву

504.	Повезати резервисану реч са редним бројем који одговара појму на који се односи у терминологији база података.																
	<table> <tr> <td>1. Табела</td><td><u>3</u></td><td>VIEW</td></tr> <tr> <td>2. Страни кључ</td><td><u>5</u></td><td>CONSTRAINT</td></tr> <tr> <td>3. Поглед</td><td><u>1</u></td><td>TABLE</td></tr> <tr> <td>4. Окидач</td><td><u>2</u></td><td>FOREIGN KEY</td></tr> <tr> <td>5. Ограничење</td><td><u>4</u></td><td>TRIGGER</td></tr> </table>	1. Табела	<u>3</u>	VIEW	2. Страни кључ	<u>5</u>	CONSTRAINT	3. Поглед	<u>1</u>	TABLE	4. Окидач	<u>2</u>	FOREIGN KEY	5. Ограничење	<u>4</u>	TRIGGER	2,5
1. Табела	<u>3</u>	VIEW															
2. Страни кључ	<u>5</u>	CONSTRAINT															
3. Поглед	<u>1</u>	TABLE															
4. Окидач	<u>2</u>	FOREIGN KEY															
5. Ограничење	<u>4</u>	TRIGGER															

505.	Уписати редни број почев од 1 на линију испред резервисане речи тако да одговара редоследу навођења.			2,5
За формирање упита за издвајање дела података из табеле која се налази у оквиру базе података користе се у следећем редоследу:				
<u>4</u>	GROUP BY			
<u>3</u>	WHERE			
<u>1</u>	SELECT			
<u>5</u>	ORDER BY			
<u>2</u>	FROM			
<hr/>				
506.	На левој страни су дате кардиналности везе, а на десној страни релације. На линији испред релација уписати одговарајући редни број кардиналности везе.			3
1. 1 : 1	<u>2</u>	ВЛАСНИК – БРОЈ ТЕЛЕФОНА		
2. 1 : M	<u>3</u>	НАСТАВНИК – ПРЕДМЕТ		
3. M : M	<u>1</u>	ОСОБА – ПАСОШ		
<hr/>				
507.	Написати на цртама испред логичких операција редне бројеве њихових приоритета:			3
1. највиши приоритет	<u>3</u>	OR		
2. средњи приоритет	<u>1</u>	NOT		
3. најнижи приоритет	<u>2</u>	AND		

508. Дата је табела **RADNIK** и табела **ODELJENJE**:

IDBR	IME	PREZIME	PLATA	BROD
5900	Slobodan	Golubović	3900	10
5932	Mitar	Gavrilović	2600	10
5953	Persida	Kosanović	1100	20
6234	Marko	Pavlović	1300	
6789	Janko	Nikolić	3900	10

BROD	Naziv	Mesto
50	Skladišta	Zemun
30	Marketing	Vračar
10	Plasman	Surčin
20	Direkcija	Grocka
40	Nabavka	Barajevo

Повезати упите и њихова значења уписом броја упита на одговарајућу линију:

- | | | |
|--|----------|--|
| 1. SELECT odeljenje.imeod, radnik.prezime
from odeljenje inner join radnik on
radnik.brod = odeljenje.brod ; | <u>5</u> | Приказује само раднике који нису
распоређени у одељења |
| 2. SELECT odeljenje.imeod, radnik.prezime
from odeljenje left join radnik on
radnik.brod = odeljenje.brod ; | <u>3</u> | Приказује све раднике, и оне који
нису распоређени у одељења, и
само она одељења у којима има
радника |
| 3. SELECT odeljenje.imeod, radnik.prezime
from odeljenje right join radnik on
radnik.brod = odeljenje.brod ; | <u>1</u> | Приказује само одељења у којима
има радника, и само раднике
распоређене у одељењима |
| 4. SELECT odeljenje.imeod, radnik.prezime
from odeljenje left join radnik on
radnik.brod = odeljenje.brod
where radnik.brod is null ; | <u>2</u> | Приказује сва одељења, и она у
којима нема радника, и само оне
раднике који су распоређени у
одељења |
| 5. SELECT odeljenje.imeod, radnik.prezime
from odeljenje right join radnik on
radnik.brod = odeljenje.brod
where odeljenje.brod is null ; | <u>4</u> | Приказује само одељења у којима
нема радника |

3