

Испит из Релационих база података, јануар 2019. год. (И смер), I група

Број индекса	Име и презиме

Задаци се раде 240 минута. Максималан број поена је 70. Број поена на испиту се израчунава тако што се саберу освојени поени по задацима, збир подели са два и заокружи. Број поена по задацима је:

Задатак	1	2	3	Збир 1-3		4	5	6	7	Збир 4-7		Укупно
Максимално	12	16	22	50	Збир/2	20	20	30	20	90	Збир/2	
Освојено												

1. Написати упит на SQL-у којим се издвајају идентификатори и називи предмета који задовољавају бар један од услова:

- предмет је обавезан на M смеру и неки студент M смера је тај предмет положио са оценом већом од 7 у последње 3 године;
- предмет има више од 2 условна предмета.

Извештај уредити у опадајућем поретку према називу предмета.

2. Написати упит на SQL-у којим се за студенте који су током школовања више пута прелазили на буџет него на самофинансирање издвајају информације о њиховим првим неуспешним полагањима предмета који нису обавезни на њиховом смеру. Издвојити име и презиме студента, назив полагањих предмета као и број освојених бодова на испиту.

3. Факултет организује путовање за студенте. Пуна цена путовања је 25000 рсд.

- Направити табелу **putovanje** која чува податке о студентима заинтересованим за путовање. Табела има колоне:
 - *indeks* - индекс студента;
 - *cena* - цена путовања за студента;
 - *status_placanja* - статус плаћања.

Дефинисати примарни кључ табеле **putovanje**, као и страни кључ на табелу *dosije*.

- Написати наредбу која у табели **putovanje** поставља ограничење да вредност колоне *status_placanja* може бити 'placeno', 'oslobodjen', 'neplaceno' и да је подразумевана вредност колоне *status_placanja* 'neplaceno'.
- Написати наредбу која у табелу **putovanje** уноси податке за студенте који су у 2012/2013 школској години положили бар половину уписаних ЕСПБ у тој школској години. Напомена: Немају сви студенти оверену школску годину 2012/2013, тј. податак у колони *overeno_bodova* не одговара броју положених ЕСПБ за сваког студента. Унети само индекс студента.
- Написати наредбу која мења табелу **putovanje** тако да садржи пријаве за путовање свих студената који нису дипломирали. Студентима о којима постоје подаци у табели **putovanje** ажурирати податке на следећи начин:
 - ако је студент бар на половини положених испита добио оцену 10 ослобођен је плаћања путовања. Поставити цену на 0 и статус плаћања на 'oslobodjen'
 - осталим студентима израчунати цену према формули: цена путовања се умањује за проценат који одговара проценту испита које је студент положио са оценом 10 у односу на његов број положених испита.

За студенте о којима не постоје подаци у табели **putovanje** унети податке. Унети индекс и цену. Пуну цену путовања умањити за проценат који одговара проценту испита које је студент положио са оценом 10.

- Написати наредбу за брисање табеле **putovanje**.

4. а) Који упитни језици су засновани само на релационом рачуну?
 б) Навести предности релационог модела у односу на хијерархијски модел.
 в) Написати кориснички дефинисану функцију $spisak(indeks, br_predmeta)$ која као аргумент има број индекса и број предмета а као вредност враћа списак бројева индекса свих студената за које важи да им је тренутни просек мањи од просека студента чији је број индекса аргумент уз услов да су положили више предмета од вредности другог аргумента функције.
5. Рангирати све табеле према броју атрибута у примарном кључу које испуњавају следеће услове:
- направљене су пре 289-ог дана 2014. године
 - направио их је корисник који тренутно извршава упит
 - у свом имену не поседују више од два слова 'а'

Извештај уредити уредити у опадајућем редоследу према имену табеле.

6. Нека је дата релациона променљива $R = \{A, B, C, D, E, F, G\}$ и скуп $F \Phi Z$:

- | | |
|------------------------|------------------------|
| 1) $AB \rightarrow C$ | 5) $D \rightarrow FG$ |
| 2) $C \rightarrow A$ | 6) $BF \rightarrow C$ |
| 3) $BC \rightarrow D$ | 7) $CG \rightarrow BD$ |
| 4) $ACD \rightarrow B$ | 8) $CF \rightarrow AG$ |

- а) Одредити све кандидате за кључ релације R.
 б) Одредити нередуцибилни покривач скупа функционалних зависности F.
 в) Испитати да ли је релација R у BCNF и декомпоновати је ако није.
 г) Да ли постоје зависности (и ако постоје које су) које нису очуване у процесу декомпозиције у BCNF?

Обавезно образложити СВЕ кораке у раду. Навођење само резултата појединих корака неће бити признато као делимично урађен задатак.

7. а) Написати SQL наредбе које атрибуту *jmbg* из табеле *dosije* мењају тип са *char(13)* на *bigint* и *jmbg* постављају га као примарни кључ. Који ниво ауторизације је потребан за сваку од ових наредби да би биле успешно извршене?
 б) На шта се односе појмови *sigurnost* и *integritet* у релационим базама података?
 в) Нека је дата табела `create table dosije_xml (podatak xml not null)` при чему су изглед и структура слога из табеле

```
<?xml version="1.0" encoding="UTF-8" ?>
  <student broj_indeksa="20140021">
    <ime>Miloš</ime>
    <prezime>Perić</prezime>
    <datum_rodjenja>"20.01.1995"</datum_rodjenja>
    <godina_studija>3</godina_studija>
    <prosek>9.45</prosek>
  </student>
```

Написати упит који враћа број индекса, име, презиме, и просек оцена свих студената чије је име 'Milos' и који се налазе на 3 години студија.

Упутство:

Направите нови директоријум на Desktop-у и назовите га у формату **rbpJanG1.brojIndeksa.ime.prezime**. Нпр. студент Марко Марковић са бројем индекса 125/2013 треба да направи директоријум *rbpJanG1.2013125.Marko.Markovic*.

У том директоријуму оставите решења задатака. Решење сваког задатка оставите у посебној датотеци. Датотеке са решењима назовите редним бројем задатка (нпр. *1.sql*, *2.sql* ...). **За сваки задатак оставите тачно једно решење, у супротном задатак неће бити прегледан.**

Задаци се решавају над базом података *vstud*.

Синтаксно неисправна решења за први, други и трећи задатак носе 0 од предвиђеног броја поена. Дозвољена литература се налази у директоријуму `/home/ispit/Desktop/Materijali/rbpKnjige`.

Испит из Релационих база података, јануар 2019. год. (И смер), II група

Број индекса	Име и презиме

Задачи се раде 240 минута. Максималан број поена је 70. Број поена на испиту се израчунава тако што се саберу освојени поени по задацима, збир подели са два и заокружи. Број поена по задацима је:

Задатак	1	2	3	Збир 1-3		4	5	6	7	Збир 4-7		Укупно
Максимално	12	16	22	950	Збир/2	20	20	30	20	90	Збир/2	
Освојено												

1. Написати упит на SQL-у којим се издвајају сви парови студент-предмет за које важи:

- да је предмет за тог студента обавезан предмет из другог семестра;
- да је студент положио све условне предмете за тај предмет;
- да студент није положио тај предмет.

Издвојити пуни потпис студента у формату: име име родитеља презиме. У случају колега издвојити име оца (ако постоји у бази) као име родитеља, док у случају колегиница издвојити име мајке (ако постоји у бази) као име родитеља. Колону назвати *Puni potpis*. Поред пуног потписа, издвојити и назив предмета. Резултат уредити по пуном потпису лексикографски растуће.

2. Написати упит на SQL-у којим се за сваког студенте основних студија који је у току досадашњег школовања уписао више предмета у првом него у другом семестру издвајају подаци о последњем успешном полагању. Издвојити име и презиме студента, назив полагааног предмета као и број бодова који је студент освојио на том испиту.

3. Написати (по једну за сваку ставку) SQL наредбу којом се:

- прави табела **izborni_predmeti** која садржи податке о изборним предметима. Табела има следеће колоне:
 - *id_predmeta* - идентификатор предмета;
 - *id_smera* - идентификатор смера на коме је предмет изборни;
 - *upisalo_ukupno* - број студената тог смера који су тај предмет (икада) уписали као изборни;
 - *polozilo* - број студената тог смера који су тај предмет (икада) положили;
 - *prosek* - просечна оцена на положеним испитима коју су добили студенти тог смера на испиту из тог предмета;
 - *upisalo_prosek_godina* - просечан број студената тог смера који у једној школској години упишу тај предмет као изборни.

Дефинисати примарни кључ над табелом **izborni_predmeti**.

- дефинише да колоне *upisalo_ukupno*, *polozilo* и *upisalo_prosek_godina* имају подразумевану вредност 0.
- у табелу **izborni_predmeti** уносе за све смерове подаци о свим изборним предметима икада уписаним. Унети све податке осим просека. Просечан број студената који су изборни предмет уписали у једној години заокружити на најмањи цео број који је једнак или већи од резултата.
- за изборне предмете на смеровима докторских студијама ажурира просечна положена оцена у табели **izborni_predmeti**.
- брише табела **izborni_predmeti**.

4. а) Шта су матични језици а шта језици података? Наведите матичне језике и језике податка које познајте.
 б) Набројите додатне (не-Кодове) операторе релационе алгебре.
 в) Нека табела *dosije* садржи атрибут *prosek*. Написати окидач *Azur* који ажурира садржај атрибута просек сваки пут када студент положи неки испит.
5. Рангирати (помоћу *rank* према просечној дужини њихових атрибута све табеле које испуњавају следеће услове

- табела није алиас, материјализована табела или поглед
- садржи више од 4 атрибута
- направио их је корисник који тренутно извршава упит
- табела не садржи текстуални атрибут који у свом имену има велика слова

Претпоставити да табела не садржи атрибуте који имају кориснички дефинисани тип. Извештај уредити уредити у опадајућем редоследу према имену табеле.

6. Нека је дата релациона променљива $R = \{A, B, C, D, E, F, G\}$ и скуп $F \Phi Z$:

- | | |
|------------------------|------------------------|
| 1) $AB \rightarrow C$ | 5) $D \rightarrow FG$ |
| 2) $C \rightarrow A$ | 6) $BF \rightarrow C$ |
| 3) $BC \rightarrow D$ | 7) $CG \rightarrow BD$ |
| 4) $ACD \rightarrow B$ | 8) $CF \rightarrow AG$ |

- а) Одредити све кандидате за кључ релације R .
 б) Одредити нередуцибилни покривач скупа функционалних зависности F .
 в) Испитати да ли је релација R у BCNF и декомпоновати је ако није.

Обавезно образложити СВЕ кораке у раду. Навођење само резултата појединих корака неће бити признато као делимично урађен задатак.

7. а) Написати SQL наредбе којима се свим корисницима допушта ажурирање атрибута *jmbg* у табели досије и забрањује приступ табели *ispit*. Који су могући нивои ауторизације потребни за сваку од ових наредби да би биле успешно извршене?
 б) Дати *формалне* дефиниције друге, треће и Бојс-Кодове нормалне форме.
 в) Нека је дата табела `create table dosije_xml (podatak xml not null)` при чему су изглед и структура слога из табеле

```
<?xml version="1.0" encoding="UTF-8" ?>
  <student broj_indeksa="20140122">
    <ime>Petar</ime>
    <prezime>Petrović</prezime>
    <datum_rodjenja>"28.02.1995"</datum_rodjenja>
    <broj_polozenih_ispita>15</godina_studija>
    <zbir_dobijenih_ocena>105</zbir_dobijenih_ocena>
  </student>
```

Написати упит који враћа број индекса, име, презиме, датум рођења и просек оцена свих студената чије је име 'Marko' и који имају просек оцена већи од 8.1.

Упутство:

Направите нови директоријум на Desktop-у и назовите га у формату **rbpJanG2.brojIndeksa.ime.prezime**. Нпр. студент Марко Марковић са бројем индекса 125/2013 треба да направи директоријум *rbpJanG2.2013125.Marko.Markovic*.

У том директоријуму оставите решења задатака. Решење сваког задатка оставите у посебној датотеци. Датотеке са решењима назовите редним бројем задатка (нпр. *1.sql, 2.sql ...*). **За сваки задатак оставите тачно једно решење, у супротном задатак неће бити прегледан.**

Задаци се решавају над базом података *vstud*.

Синтаксно неисправна решења за први, други и трећи задатак носе 0 од предвиђеног броја поена. Дозвољена литература се налази у директоријуму `/home/ispit/Desktop/Materijali/rbpKnjige`.