

Испит из Релационих база података, јануар 2 2020. год. (И смер), I 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 4-6		Укупно
Максимално	14	16	20	50	Збир/2	20	15	25	30	90	Збир/2	
Освојено												

- Написати SQL упит којим се издвајају информације о свим студентима смерова *Teorijska matematika i primene, Numericka matematika i optimizacija, Astronomija* и *Analiza*, за које важи да је или предмет са највећим бројем ЕСПБ бодовима кога су до сада положили уједно и обавезни предмет са њиховог смера, или да имају просек оцена рачунајући само положене испите већи од 7,5. Издвојити индекс, име и презиме студента у облику *име размак презиме*, као и назив смера. Резултат упита уредити у опадајући редослед по вредности индекса.
- Написати SQL упит којим се издвајају следеће информације за рокове у којима су одржани испити из мање од 25 различитих предмета, укључујући и рокове у којима нико није полагао ни један испит:
  - назив испитног рока;
  - број различитих студената који су изашли на неки испит у том року;
  - број предмета из којих су одржани испити у том року;
  - који проценат од броја под (c) чине предмети на којима у том року нико није добио оцену 10.

Недостајуће вредности заменити са 0 или 0.0.

- Написати SQL наредбу којом се прави табела *diplomirani* која садржи податке о студентима који су дипломирали на факултету. Табела има колоне:
    - indeks* - индекс студента;
    - datum* - датум дипломирања;
    - prosek* - просечна оцена на положеним испитима студента;
    - duzina\_studiranja* - дужина студирања студента (у данима).
 Дефинисати и примарни кључ за табелу *diplomirani*. Примарни кључ се може дефинисати у оквиру наредбе за прављење табеле или као засебна SQL наредба.
  - Написати SQL наредбу која у табелу *diplomirani* уноси податке (у све колоне табеле) за студенте који никада нису поништили испит и који су дипломирали на смеру на коме је предвиђена дужина студирања 4 године. Дужина студирања се рачуна као број дана од датума уписа на факултет до дана када је студент дипломирао.
  - Написати SQL наредбу која мења табелу *diplomirani* тако што студентима чији се подаци налазе у табели и који су студирали смер са називом *Matematika* ажурира просек, док за студенте који су дипломирали али се њихови подаци не налазе у табели уноси индекс, датум дипломирања и просек оцена.
  - Написати SQL наредбу којом се из табеле *diplomirani* бришу подаци о студентима који су факултет уписали пре 2008. године.
  - Написати SQL наредбу која брише табелу *diplomirani*.
- Дати формалну дефиницију и описати релациони оператор сумаризације.
  - Навести предности релационог модела у односу на хијерархијски модел.
  - Наведите објекте у ДБ2 и DDL наредбе које познајете. Над којим објектима може да се примени која DDL наредба?

5. Написати кориснички дефинисану функцију *redosled\_polaganja* чији је аргумент *broj\_indeksa* а вредност листа која садржи име и презиме студента, називе предмета које је студент положио и датуме полагања тих предмета у Европском формату. Датум полагања навести иза предмета раздвојену косом цртом, а појединачне предмете у листи су раздвојити тачком зарезом. Називи предмета у листи су сортирани у растућем редоследу према датумима полагања испита. У случају да студент чији је индекс унет као аргумент није положио ни један испит, вратити поруку "Студент <име и презиме> није положио ни један испит, а ако не постоји студент са унетом вредношћу индекса вратити поруку 'Студент са унетим индексом *broj\_indeksa* не постоји'.

На пример, за студента чији је број индекса 20140021 добија се као резултат *MilosPeric: Analiza 1/20.01.2015; Engleski jezik 1/21.01.2015; Programiranje 1/25.01.2015; Geometrija/03.04.2015*, док се за унети број индекса 123456 добија порука *Студент са унетим индексом 123456 не постоји*.

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

- |                        |                         |
|------------------------|-------------------------|
| 1) $G \rightarrow CF$  | 5) $CG \rightarrow AE$  |
| 2) $BG \rightarrow DA$ | 6) $DG \rightarrow B$   |
| 3) $BF \rightarrow D$  | 7) $AEF \rightarrow D$  |
| 4) $AD \rightarrow GE$ | 8) $BGF \rightarrow EC$ |

(а) Одредити све кандидате за кључ релације  $R$ .

(б) Одредити нередуцибилни покривач скупа функционалних зависности  $F$ .

- б) Дати формалну дефиницију 4 и 5 нормалне форме

- в) Нека је дата релациона променљива  $Z = \{A, B, C, D\}$ . Испитати да ли су следећа два скупа  $\Phi Z$  еквивалентна:

(а)  $X = \{A \rightarrow B, A \rightarrow BC, AB \rightarrow C, AC \rightarrow D, B \rightarrow C\}$ , и

(б)  $Y = \{CA \rightarrow B, BA \rightarrow D, B \rightarrow D, DB \rightarrow C\}$

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

7. а) 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>
    <mesto_rodjenja>"Niš"</mesto_rodjenja>
    <prosek>9.45</prosek>
  </student>
```

Написати упит који враћа број индекса, име, презиме, и просек оцена свих студената чије је име 'Miloš' и који су рођени у Врању.

- б) Шта су и чему служе окидачи?

- в) Која је сврха употребе кориснички дефинисаних типова? Да ли исти ефекат може да се постигне без њихове употребе? Објаснити одговор!

- г) Написати SQL упит који приказује све табеле које је направио корисник који тренутно извршава упит, а које садрже атрибуте типа карактер дуже од 50 бајтова и који могу да садрже ћирилични текст (са кодном страном 1208).

### Упутство:

Направите нови директоријум на Desktop-у чије је име облика **rbpJan2.brojIndeksa.ime.prezime**. Нпр. студент Марко Марковић са бројем индекса 125/2017 треба да направи директоријум *rbpJan2G1.2017125.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 1-3 се решавају над базом података *vstud*, а 4-7 над базом података *mstud*.

Синтаксно неисправна решења за задатке 1-3 носе 0 од предвиђеног броја поена. Грешке у решењима задатака 4-7 које показују непознавање испитне материје повлаче 0 поена на комплетном задатку.

Дозвољена литература се налази у директоријуму `/home/ispit/Desktop/Materijali/rbpKnjige`.