

Усмени испит из Релационих база података, септембар 2016. год. (Исмер)
 Усмени испит из Увода у релационе базе података, септембар 2016. год. (Р, Н смер)

Број индекса	Име и презиме

Задаци се раде 180 минута. **ПИСАТИ ЧИТКО - НЕЧИТКИ ЗАДАЦИ НЕЋЕ БИТИ ПРЕГЛЕДАНИ!**
 Максималан број поена је 35. 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:

Задатака	1	2	3	4	5	6	7	Збир	Укупно
Максимално	10	9	9	9	10	15	8	70	Збир/2
Освојено									

- Показати да помоћу оператора из минималног скупа Кодових оператора могу да се изведу оператор дељења и оператор ексклузивне уније.
 - Доказати да за операторе релационе алгебре важи да је пројекција није дистрибутивна у односу на разлику.
 - Навести ограничења базе која могу да се јаве над базом *VSTUD*.
 - Набројати послове које обавља администратор базе података.
- Детаљно описати референцијалне акције које могу да се јаве при брисању или уношењу података у табелу у систему DB2.
 - Написати SQL упит којим се приказују сви алијаси на табеле које је направио корисник који извршава упит, као и табеле на које се ти алијаси односе, при чему излаз треба да буде сортиран у опадајућем редоследу према времену када су алијаси формирани. Имена табела наводити са пуним квалификованим именом (схема.табела)
- Написати MQT упит којим се издвајају имена, презимена, место рођења и максимум просечне оцене добијене на положеним испитима у неком од испитних рокова.
 - Написати упит којим се формира и попуњава табела *dosije_jmbg* која има исте атрибуте и садржај као и табела *dosije* и чији је примарни кључ атрибут *jmbg*. Табелу формирати тако да се забрањује њено брисање.
 - Написати SQL упит помоћу којих се омогућава кориснику KOR1 да уноси податке у табелу *dosije_jmbg*, али не и да чита податке из те табеле.
- Написати кориснички дефинисану функцију *rang(broj_indeksa, id_smera)* чији је аргумент број индекса студента, а вредност редослед тог студента у листи студената на смеру *id_smera* сортираној према просеку студента.
 - Написати SQL упит којим се дуплира вредност атрибута *indeks* у табели досије.
 - Написати SQL упит којим се издвајају име, презиме, број индекса студента и позиција четвртог појављивања ниске 'Нови' од краја у месту рођења студента или 0 ако таква ниска не постоји. На пример, ако је место рођења студента 'АбвгНови абвг Нови Сад НовиНовиНови' тада је тражена вредност 15, док је за место рођења студента 'Нови Сад абвг НовиНови Град' тада је тражена вредност 0.

6. Нека је дата релациона променљива $R = \{A, B, C, D, E, F, G\}$ и скуп $F \Phi 3$:

- 1) $AB \rightarrow C$
- 2) $C \rightarrow A$
- 3) $BC \rightarrow D$
- 4) $ACD \rightarrow B$
- 5) $D \rightarrow EG$
- 6) $BE \rightarrow C$
- 7) $CG \rightarrow BD$
- 8) $CE \rightarrow AG$

- a) Одредити минимални покривач скупа функционалних зависности
- b) Одредити све кандидате за кључ релације R .
- в) Нека је релација $R_1 = \{C, D, F\}$ пројекција релације R . Одредити скуп функционалних зависности које су важеће у R_1 .

Обавезно образложити СВЕ кораке у раду. Навођење само резултата појединих корака (нпр. затворења скупа атрибута без објашњења како се до њега дошло) неће бити признато као делимично урађен задатак.

7. а) Дати дефиниције II, III, Бојс-Кодове и IV нормалне форме.
- б) Проверити да ли је релација R из задатка 6. у III нормалној форми.

Упутство: Задаци могу да се раде у вежбанци, на рачунару или комбиновано. Уколико се неки од задатака ради на рачунару направите нови директоријум и **ОБАВЕЗНО** га назовите **Ime.Prezime.Jun.brojIndeksa**. Нпр. ако студент Пера Перић има број индекса 125/2013, он треба да направи директоријум *Pera.Peric.Sep.2013.125*. У направљеном директоријуму оставите решења задатака. Решење сваког задатка оставите у посебној датотеци. Датотеке са решењима назовите према редном броју задатка (нпр. *задатак 1*, *задатак 2*, ...). За сваки задатак оставите тачно једно решење, у супротном задатак неће бити прегледан и биће оцењен са 0 (нула) поена.