

Испит из Релационих база података, јануар 2020. 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 4-6		Укупно
Максимално	13	17	20	50	Збир/2	20	15	20	35	90	Збир/2	
Освојено												

- Написати упит на SQL-у којим се издвајају семестар и назив за оне предмете из којих је био организован курс 2010. школске године, а никада курс из тог предмета нису уписали студенти из Ниша. Уз семестар и назив предмета издвојити број студената који су га уписали те године и проценат од тада уписаних студената који су у том семестру и школској години и положили уписан предмет. Уколико не постоје неки од тражених бројева за одређени курс, издвојити 0. Колоне назвати, редом, *Upisali* и *Polozili*. Резултат приказати у растућем поретку према називу предмета и броју уписаних студената.
- Написати упит на SQL-у којим се издвајају називи и ознаке смерова са мастер студија (назив нивоа квалификације почиње са *Master*) које је до сада уписало најмање 10 студента. Поред наведених информација издвојити и број студента који су икада уписали тај смер, место рођења, број студената из тог места који су тај смер уписали, а потом и годину у којој је највећи број студената уписао тај смер. За годину уписа узети четвороцифрен број од 4 цифре највеће тежине из индекса студента.
- Написати SQL наредбу којом се прави табела *rok_statistike* која садржи статистике о успешности полагања испита. Табела има атрибуте:
 - tip* - тип испитног рока;
 - id_predmeta* - идентификатор полагањег предмета;
 - prosek* - просечна оцена на положеним испитима из предмета *id_predmeta* у роковима типа *tip*;
 - prolaznost* - пролазност на испитима из предмета *id_predmeta* у роковима типа *tip*. Пролазност се рачуна се као проценат положених испита у односу на број пријављених испита. Подразумевана вредност атрибута је 0.
 - Написати (једну) SQL наредбу којом се у табели *rok_statistike* дефинише
 - примарни кључ;
 - ограничење да вредност колоне *prosek* може бити између 6 и 10;
 - ограничење да вредност колоне *prolaznost* може бити између 0 и 100.
 - Написати SQL наредбу којом се у табелу *rok_statistike* уносе подаци само за предмете који су међу обавезним предметима смерова на основним студијама (може се користити податак да је степен нивоа квалификације за такве смерове *I*). Узети у обзир само испитне рокове чији је период за пријављивање испита био бар 10 дана. Узети у обзир само типове испитних рокова и предмете за које важи да је бар 5 испита из тог предмета пријављено у роковима тог типа. Напомена: сваки ред у табели *ispit* је пријављени испит.
 - Написати SQL наредбу којом се у табели *rok_statistike* за сваки тип испитног рока остављају само статистике о полагањима предмета са највећом пролазношћу у роковима тог типа.
 - Написати SQL наредбу којом се брише табела *rok_statistike*.

4. а) Навести формалне дефиниције бар 4 додатна оператора релационе алгебре.
 б) Описати својства трансакција у релационим базама.
 в) Навести формалне дефиниције примарног и спољашњег кључа. Шта је суперкључ а шта кандидат за кључ? Зашто (по теорији) свака релација има примарни кључ?
5. Написати кориснички дефинисану функцију *istaslova* чији је аргумент *niska* карактер променљива максималне дужине 200 која садржи самогласнике и специјалне знаке празнину и узвичник. Функција враћа резултат *Isto slovo slovo* ако аргумент *niska* садржи само слово *slovo* и специјалне знаке. У случају да је *niska* празна, приказати поруку *Prazna niska*, ако *niska* садржи само специјалне знаке приказати поруку *Ne sadrži slova* и *Nije isto slovo* у осталим случајевима. На пример, позив *istaslova('aaaaa aaaa')* као резултат треба да врати *Isto slovo a*, позив *istaslova('ae aaaa')* враћа резултат *Nije isto slovo*, док позив *istaslova('! ! !')* враћа *Ne sadrži slova*.
6. Нека је дата релациона променљива $R = \{A, B, C, D, E, F, G\}$ и скуп $F \Phi Z$:
- 1) $BG \rightarrow CD$
 - 2) $G \rightarrow F$
 - 3) $CD \rightarrow GE$
 - 4) $C \rightarrow FG$
 - 5) $F \rightarrow D$
- а) Одредити све кандидате за кључ релације R .
 б) Одредити нередуцибилни покривач скупа функционалних зависности F .
 в) Испитати да ли је релација R у 2, 3 и BCNF и декомпоновати је ако није.
 г) Нека је релација $R_1 = \{C, D, G\}$ пројекција релације R . Одредити скуп функционалних зависности које су важеће у R_1 .
- Обавезно образложити СВЕ кораке у раду. Навођење само резултата појединих корака неће бити признато као делимично урађен задатак.
7. а) Детаљано опишите аспекте проблема сигурности које познајете.
 б) Које услове мора да испуни поглед да би могао да се ажурира у систему DB2? Шта означава опција WITH CHECK приликом дефинисања погледа?
 в) Написати наредбе којима се формира алијас *Studenti* за табелу *Dosije*, и поставља атрибут *jmbg* као примарни кључ. Који ниво ауторизација је потребан за извршавање ових наредби?
 г) Написати SQL упит који за све табеле које је направио корисник који тренутно извршава упит, а које нису коришћене у последњих годину дана у односу на тренутак када се извршава упит, приказује назив табеле, број атрибута и број слогова у њој, у формату *ime-tabele sadrži xx atributa i yy slogova*. На пример, за табелу *dosije* упит би приказао *DOSIJE sadrži 6 atributa i 7 slogova*

Упутство:

Направите нови директоријум на Desktop-у чије је име облика **rbpJan1.brojIndeksa.ime.prezime**. Нпр. студент Марко Марковић са бројем индекса 125/2017 треба да направи директоријум *rbpsep.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`.

Испит из Релационих база података, јануар 2020. год. (И смер), II 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 4-6		Укупно
Максимално	13	17	20	50	Збир/2	20	10	25	35	90	Збир/2	
Освојено												

- Написати упит на SQL-у којим се издвајају информације о студирању оних студената који су положили макар један обавезан предмет са свог смера, а који није положио ни један други студент из њиховог места рођења. Издвојити информације само за студенте са не-информатичких смерова (који нису са смерова *Рачунарство и информатика* и *Информатика*), који имају просек оцена већи од 8,5 и чије је место рођења познато. Издвојити индекс, име и презиме студента, назив смера, просечну оцену (као реалан број у фиксном зарезу са 2 децимална места), као и број положених испита свакога од њих. Испит уредити у опадајућем поретку по просеку студента и у оквиру истог просека у растућем поретку по имену студента.
- Написати упит на SQL-у којим се за све студенте мастер студија (назив нивоа квалификације почиње са *Master*) чији је укупан просек оцена строго већи од њиховог просека оцена на обавезним, а строго мањи од њиховог просека оцена на необавезним предметима, издвајају број индекса, просечна оцена на положеним обавезним предметима (колону назвати *Obavezni prosek*), као и о број различитих испитних рокова у којима су положили макар један предмет (колону назвати *Broj srećnih rokova*). Просечну оцену приказати као реалан број у фиксном зарезу са 4 децимална места. Извештај уредити у опадајућим редоследима по броју срећних рокова, и у оквиру истог броја срећних рокова у опадајућем поретку по просеку студента. Уколико за неког од студента не постоји просек, на том месту у извештају навести 0.
- (a) Написати SQL наредбу којом се прави табела *izborni_predmeti_statistike* која садржи статистике о успешности полагања испита из изборних предмета. Табела има атрибуте:
 - id_smera* - идентификатор смера;
 - id_predmeta* - идентификатор изборног предмета;
 - prosek* - просечна оцена на положеним испитима из предмета *id_predmeta* студената који студирају смер *id_smera*;
 - prolaznost* - пролазност на испитима из предмета *id_predmeta* студената који студирају смер *id_smera*. Пролазност се рачуна као проценат положених испита у односу на број пријављених испита. Подразумевана вредност атрибута је 0.
- (b) Написати (једну) SQL наредбу којом се у табели *izborni_predmeti_statistike* дефинише
 - примарни кључ;
 - страни кључ на табелу *smer*;
 - страни кључ на табелу *predmet*;
 - ограничење да вредност колоне *prolaznost* мора бити између 0 и 100.
- (c) Написати SQL наредбу којом се у табелу *izborni_predmeti_statistike* уносе подаци за смерове и њихове изборне предмете за које важи да су бар 4 испита из тог изборног предмета положили студенти тог смера.
Напомена: сваки ред у табели *ispit* је пријављени испит.
- (d) Написати SQL наредбу којом се из табеле *izborni_predmeti_statistike* бришу подаци за изборне предмете чија је просечна оцена већа од просека просечних оцена изборних предмета истог смера.
- (e) Написати SQL наредбу којом се брише табела *izborni_predmeti_statistike*.

4. а) Опишите аспекте релационог модела података.
 б) Дати формалну дефиницију и описати релациони оператор сумаризације
 в) Навести предности релационог модела у односу на хијерархијски модел.
5. Написати кориснички дефинисану функцију *slova* чији је аргумент *niska* карактер променљива максималне дужине 200, а резултат разлика броја малих и великих слова у нисци. Резултат је позитиван ако ниска има више малих и негативан уколико ниска има више великих слова. У случају да је ниска празна, приказати поруку *Prazna niska*, а ако је број малих и великих слова једнак вратити поруку *Jednak broj velikih i malih slova*. На пример, *slova('Ovo je niska sa velikim i MALIM slovima KOJA se Ponaumljaju')* као резултат треба да врати 26 (дужина ниске је 58, садржи 11 великих слова и 10 специјалних знакова (овде празнина)).
6. Нека је дата релациона променљива $R = \{A, B, C, D, E, F, G\}$ и скуп $F \Phi Z$:
- 1) $BG \rightarrow CD$
 - 2) $G \rightarrow F$
 - 3) $CD \rightarrow GE$
 - 4) $C \rightarrow FG$
 - 5) $F \rightarrow D$
- а) Одредити све кандидате за кључ релације R .
 б) Одредити нередуцибилни покривач скупа функционалних зависности F .
 в) Испитати да ли је релација R у 2, 3 и BCNF и декомпоновати је ако није.
 г) Нека је релација $R_1 = \{C, D, G\}$ пројекција релације R . Одредити скуп функционалних зависности које су важеће у R_1 .
- Обавезно образложити СВЕ кораке у раду. Навођење само резултата појединих корака неће бити признато као делимично урађен задатак.
7. а) Детаљано опишите аспекте проблема сигурности које познајете.
 б) Које услове мора да испуни поглед да би могао да се ажурира у систему DB2? Шта означава опција *WITH CHECK* приликом дефинисања погледа?
 в) Написати наредбе којима се формира синоним *Studenti* за табелу *Dosije*, и поставља атрибут *Prezime* као примарни кључ. Који ниво ауторизација је потребан за извршавање ових наредби?
 г) Написати SQL упит који за све табеле које нису алијаси а направио их је корисник који тренутно извршава упит у последњих годину дана у односу на тренутак када се извршава упит, приказује за сваки тип број табела тог типа и укупну дужину њихових атрибута.

Упутство:

Направите нови директоријум на Desktop-у чије је име облика **rbpJan1.brojIndeksa.ime.prezime**. Нпр. студент Марко Марковић са бројем индекса 125/2017 треба да направи директоријум *rbpsep.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`.