

Базе података 1

– други колоквијум –

Број индекса (гггг/бббб), име и презиме	Потпис дежурног	Број поена	

Напомена: Није дозвољена употреба литературе. Колоквијум траје **90** минута.

1. (5)Дат је садржај дела базе података са ски аранжманима једне агенције. У табелама *Skijaliste* и *Termin* се прате подаци о скијалиштима која се налазе у понуди и организованим терминима .

Skijaliste			
IdSki	Naziv	BrStaza	Osnezenje
1	Kopaonik	25	D
2	Zlatibor	4	N
3	Les 2 Alpes	96	D
4	Paradiski	253	D
5	Chatel	49	NULL
6	Bansko	18	D

Termin				
IdTer	Od	Do	IdSki	Cena
1	20181229	20190105	1	350
2	20190105	20190112	1	230
3	20190126	20190202	6	165
4	20190125	20190203	3	396
5	20190201	20190210	3	410
6	20190308	20190317	3	379

а) Написати *SQL* упит за прављење табеле *Skijaliste*. *IdSki* је целобројна величина која идентификује скијалиште, *Naziv* представља низ до 45 карактера и обавезно је, *BrStaza* је целобројна величина, *Osnezenje* је карактер ('D' - има освежење; 'N' - нема освежење) . Остала ограничења треба игнорисати.

б) Одлучено је да се укину сви термини који су скупљи од 400. Написати *SQL* упит које реализују брисање описаних термина.

в) Уведено је вештачко освежење на Златибору. Написати *SQL* упит за промену информације о освежењу за *IdSki* једнако 2.

г) Одлучено је да се уведе нови термин за Срећење на Копаоник. Датум поласка је предвиђен за 13. фебруар 2019., датум повратка за 17. фебруар 2019., *idTer* једнако 7 и *Cenom* од 170. Написати *SQL* упит који додаје наведени термин у одговарајућу табелу.

д) Написати *SQL* упит за приказ скијалишта којима није позната информација о освежењу (*NULL* вредност). Приказ резултата треба да буде у формату: *IdSki, Naziv Skijalista* (називи колоне треба да имају више речи).

ђ) Написати *SQL* упит за приказ просечног броја стаза на свим скијалиштима која поседују вештачко освежење и имају више од 10 стаза. Приказ резултата треба да буде у формату: *Prosek*.

е) Потребно је написати *SQL* упит која дохвати све термине који се односе на скијалишта са више од 15 стаза. Резултат треба сортирати прво растуће по датуму поласка, а после опадајући по броју стаза. Приказ резултата треба да буде у формату: *Naziv, BrStaza, Od, Do, Cena*

ж) Потребно је написати *SQL* скрипту која прави поглед (*VIEW*) *TerminUMartu* који као приказ даје оне термине који су у потпуности у марту 2019. Искористи поглед *TerminUMartu* за приказ свих јефтиних термина у марту. Термин је јефтин ако је његова цена нижа од 200. Приказ резултата треба да буде у формату: *IdTer, Od, Do, Cena*

з) Потребно је написати *SQL* упит која дохвата све датуме када је било 2 или више поласка. Приказ резултата треба да буде у формату: *Datum*

2.(10) Дати су шема релације $R(A, B, C, D, E, F, G, H)$ и скуп функцијских зависности $F = \{ AFG \rightarrow H, E \rightarrow DF, FG \rightarrow AG, C \rightarrow AF, CD \rightarrow EFG \}$. Потребно је:

а) Одредити скуп кандидат кључева КК дате шеме.

Одговор:

б) Испитати редом да ли је дата шема у BC, 3. и 2. нормалној форми и сваки пут у табели назначити да ли посматрана зависност нарушава посматрану нормалну форму.

	$AFG \rightarrow H$	$E \rightarrow DF$	$FG \rightarrow AG$	$C \rightarrow AF$	$CD \rightarrow EFG$
BCNF					
3NF					
2NF					

в) Спровести нормализацију дате шеме у 3. нормалну форму алгоритмом који гарантује очување функцијских зависности.

Одговор:

г) Спровести нормализацију дате шеме у BC нормалну форму, издвајајући зависности редоследом са десна на лево.

Одговор:

д) Испитати да ли је при поступку у оквиру тачке г) дошло до суштинских губитка функцијских зависности и којих?

Одговор: