

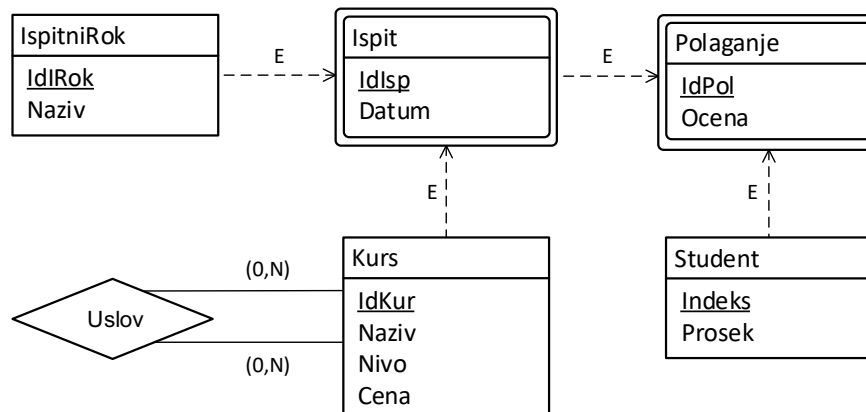


Базе података 1

– јулски испитни рок –

Посматра се део базе података једног факултета. У овој бази прате се студенти, курсеви, испити, испитни рокови, полагања и оцене студената. Студент поседује свој индекс који га идентификује, као и просечну оцену. Сваки курс има свој назив, ниво, као и цену коју студенти плаћају приликом изласка на испит како би имали могућност да добију оцену и да га положи. Курсеве студенти полагају у оквиру испитног рока, у оквиру којег за сваки курс постоји један испит. Дакле један курс могу полагати у више испитних рокова без обзира да ли га претходно нису положили или јесу (када желе бољу оцену). За сваки испит бележи се датум одржавања. Студент не може полагати курс уколико није претходно положио курсеве који су његов предуслов.

У наставку је дата релациона шема посматраног дела базе податка.



Ispit (IdIsp, Datum, IdIRok, IdKur)

- IdIsp - цео број, идентификује испит
- Datum - цео број, обавезно, формат (ggggmmdd)
- IdIRok - страни кључ (табела IspitniRok), обавезно
- IdKur - страни кључ (табела Kurs), обавезно

IspitniRok (IdIRok, Naziv)

- IdIRok - цео број, идентификује испитни рок
- Naziv - низ од 8 знакова, обавезно, формат (mes gggg); mes је наведено малим словима (могуће вредности: jan, feb, mar, apr, maj, jun, jul, avg, sep, okt, nov, dec)

Kurs (IdKur, Naziv, Nivo, Cena)

- IdKur - цео број, идентификује курс
- Naziv - низ од 50 знакова, обавезно
- Nivo - низ од 50 знакова, обавезно
- Cena - цео број, обавезно, вредност већа од 0

Напомена: Број нивоа је променљив и није фиксан осим уколико у задатку није написано другачије.

Polaganje (IdPol, Ocena, Indeks, IdIsp)

IdPol	- ceo број, идентификује полагање, аутоматско додељивање наредног идентификатора
Ocena	- ceo број, обавезно, вредности у опсегу [5, 10]
Indeks	- страни кључ (табела Student), обавезно
IdIsp	- страни кључ (табела Ispit), обавезно

Напомена: Оцена 5 означава да студент није положио испит. Студент приликом сваког полагања плаћа цену курса.

Student (Indeks, Prosek)

Indeks	- низ од 9 знакова, идентификује студента, формат (gggg/bbbb)
Prosek	- разломљени број, није обавезно

Uslov (IdKurUslovljen, IdKurUslov)

IdKurUslovljen	- страни кључ (табела Kurs), обавезно, условљен курс
IdKurUslov	- страни кључ (табела Kurs), обавезно, курс који је услов за условљен

Задатак 1 [3 поена]

Потребно је направити SQL упит који исписује све испите који су одржани 2010. године у свим испитним роковима осим у фебруарском испитном року, тако што исписује идентификатор испита, датум одржавања испита и назив испитног рока. Резултат треба сортирати растуће по идентификатору испита.

Резултат дати у форми: IdIsp, Datum, Ispitni rok
У Сactus-у користити таб: Zadatak 1

Задатак 2 [3 поена]

Потребно је направити SQL упит који исписује који студенти су полагали које испите у испитном року са називом 'jun 2011'. Треба исписати и испите у том испитном року које ниједан студент није полагао. Прво исписати назив испитног рока, идентификатор испита, датум полагања испита и индекс студента који је полагао испит. Резултат треба сортирати опадајуће по идентификатору испита, а затим растуће по индексу студента.

Резултат дати у форми: Ispitni rok, IdIsp, Datum, Indeks
У Сactus-у користити таб: Zadatak 2

Задатак 3 [3 поена]

Потребно је направити SQL упит који приказује испитне рокове који су се догодили у годинама после 2010. и у којима је полагао бар 3 студента. Резултат треба сортирати растуће по идентификатору испитног рока.

Резултат дати у форми: IdIRok, Naziv
У Сactus-у користити таб: Zadatak 3

Задатак 4 [3 поена]

Потребно је направити SQL упит који за сваки ниво исписује просечну зараду по години. Заради доприноси свако полагање курса када се сматра да је студент платио полагање. Уколико је ниво "I" треба исписати "Osnovni nivo", уколико је "II" тада "Srednji nivo" и уколико је "III" тада "Napredni nivo". Исписати прво назив нивоа, затим просечну годишњу зараду. Зараду исписати за оне нивое за које зарада постоји. Нивое треба исписати од нижих ка вишим.

Резултат дати у форми: Naziv nivoa, Prosečna godišnja zarada.

У Cactus-у користити таб: Zadatak 4

Задатак 5 [4 поена]

Потребно је направити SQL упит који исписује који студент је одређени курс полагао више од једном, као и колико пута га је полагао. Резултат сортирати прво опадајуће по броју излазака, затим растуће по индексу.

Резултат дати у форми: Indeks, IdKur, Број.

У Cactus-у користити таб Zadatak 5.

Задатак 6 [4 поена]

Потребно је направити SQL скрипту која ако постоји табела **Polaganje** брише табелу **Polaganje** из шеме, а затим формира нову табелу **Polaganje** која треба да има одговарајућу структуру и ограничења.

У Sactus-у користити таб Zadatak 6.

Задатак 7 [4 поена]

Потребно је направити SQL упит који исписује нивое курсева у којима ни један курс тог нивоа не условљава неки други курс. У овом задатку сматрати да у бази може бити произвољно много нивоа, дакле тај број није фиксан ни предефинисан. Сортирати нивое у опадајућем поретку.

Резултат дати у форми: Nivo

У Sactus-у користити таб: Zadatak 7

Задатак 8 [4 поена]

Потребно је направити SQL упит који исписује све курсеве који се морају положити како би положио курс са називом „ООР“. Резултат треба сортирати растуће по нивоу, затим растуће по идентификатору курса.

Резултат дати у форми: IdKur, Naziv, Nivo.

У Sactus-у користити таб: Zadatak 8

Није дозвољено коришћење погледа.

Задатак 9 [5 поена]

Потребно је направити SQL скрипту која мења просечну оцену за студента, затим приказује све податке из табеле **Student** сортиране растуће по индексу студента. За студенте који немају ниједан положен испит исписати 0 за просек.

Резултат дати у форми: Indeks, Prosek

У Sactus-у користити таб: Zadatak 9

Није дозвољено коришћење погледа.

Задатак 10 [5 поена]

Потребно је направити SQL исказ који поставља нову цену за курсеве који припадају нивоу чија је просечна цена мања од просечне цене свих нивоа и уколико се у оквиру тог нивоа налази курс са најмањом пролазношћу. Курс са најмањом пролазношћу је онај који има највише оцена 5. Нова цена за курсеве је за 10% већа од просечне цене свих нивоа.

Након ажурирања података, исписати све податке из табеле **Kurs** сортиране растуће по идентификатору курса.

Резултат дати у форми: IdKur, Naziv, Nivo, Cena

У Sactus-у користити таб: Zadatak 10

Није дозвољено коришћење погледа.

Задатак 11 [6 поена]

Потребно је направити SQL упит који исписује колико нивоа је завршио сваки од студената. Студент је завршио одређени ниво уколико је положио све курсеве тог нивоа. Резултат сортирати растуће по индексу студента.

Резултат дати у форми: Indeks, Broj završenih nivoa.

У Sactus-у користити таб: Zadatak 11

Није дозвољено коришћење погледа.

Задатак 12 [6 поена]

Потребно је направити SQL упит који за сваког студента исписује колико најмање испита мора положити и колико најмање новца мора дати тако да заврши последњи ниво, односно положи све испите на последњем нивоу. Последњи ниво је ниво који нема ни један курс који условљава други курс. Претпоставити да постоји само један највиши ниво. Студент не мора положити све испите са претходних нивоа. Није потребно исписивати студенте који су положили све испите последњег нивоа. Резултат сортирати опадајуће по броју курсева, затим опадајуће по новцу, затим по индексу студента растуће.

Резултат дати у форми: Indeks, BrojKurseva, Novca

У Sactus-у користити таб: Zadatak 12

Није дозвољено коришћење погледа.
