



Базе података 1

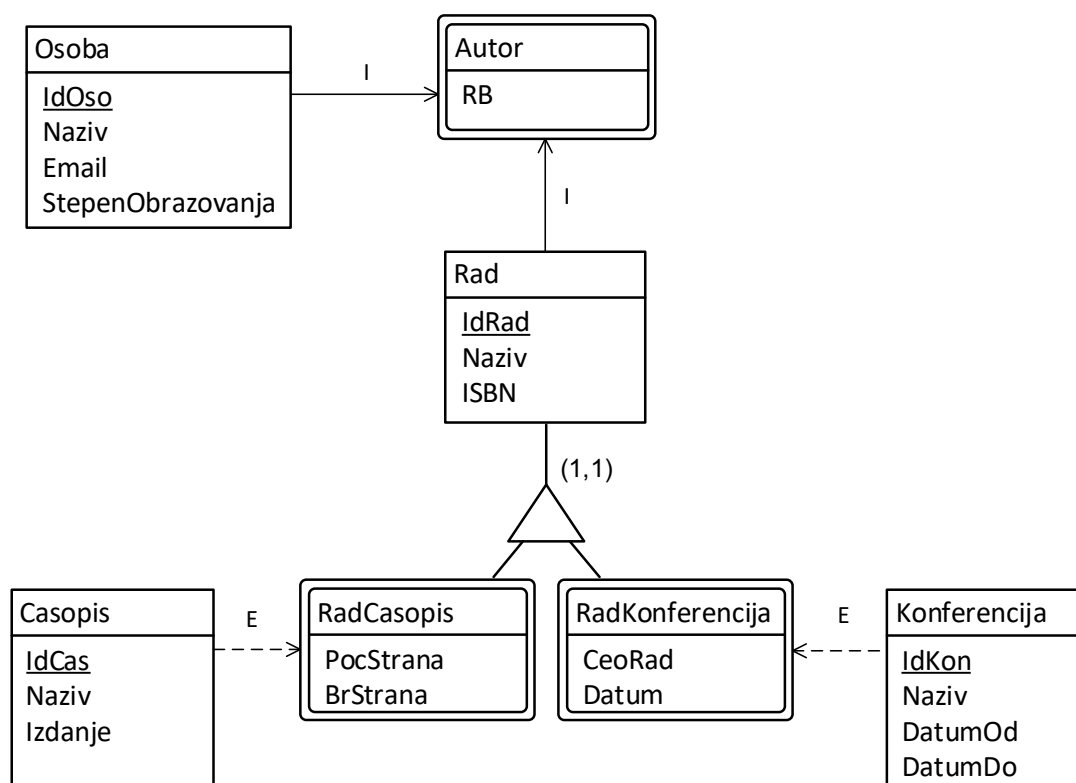
(13C112БП1, 13E113БП1)

- јануарски испитни рок -

Група Б

Посматра се база података за евидентирање научних радова. У овој бази се прате особе (назив, email и степен образовања), часописи (назив и издање), конференције (назив, датум почетка конференције и датум краја конференције) и радови. За сваки рад се води евиденција ко су његови аутори, као и редни бројеви аутора на раду. Сматрати да су редни бројеви аутора исправно попуњени (узимају узастопне вредности од 1 до броја аутора на раду, без прекида и понављања). Радови могу бити објављени или у часописима или на конференцијама. За радове у часописима се прати назив, ISBN, часопис у којем је објављен рад, редни број стране на којој почиње рад и колико страна има рад. За радове на конференцији се прати назив, ISBN, да ли је објављен цео рад или само апстракт и ког датума је рад презентован. У систему се датуми памте као целобројне величине у формату ууууmmdd (уууу – година, mm – месец, dd – дан).

У наставку је дата релациона шема посматране базе података.



Osoba (IdOso, Naziv, Email, StepenObrazovanja)

IdOso	- цео број, идентификује особу, аутоматско додељивање наредног идентификатора
Naziv	- низ до 50 знакова, обавезно
Email	- низ до 100 знакова, обавезно, јединствена вредност
StepenObrazovanja	- цео број, обавезно, вредност између 1 и 5

Напомена:

- значење атрибута StepenObrazovanja: 1 - "osnovne akademske studije", 2 - "master akademske studije", 3 - "doktorske studije", 4 - "postdoktorske studije", 5 - "ostalo".

Casopis (IdCas, Naziv, Izdanje)

IdCas	- цео број, идентификује часопис, аутоматско додељивање наредног идентификатора
Naziv	- низ до 200 знакова, обавезно, јединствено у комбинацији са пољем Izdanje
Izdanje	- цео број, обавезно, вредност већа од 0, јединствено у комбинацији са пољем Naziv

Konferencija (IdKon, Naziv, DatumOd, DatumDo)

IdKon	- цео број, идентификује конференцију, аутоматско додељивање наредног идентификатора
Naziv	- низ до 200 знакова, обавезно
DatumOd	- цео број, обавезно, у формату (ууууmmdd)
DatumDo	- цео број, обавезно, у формату (ууууmmdd)

Rad (IdRad, Naziv, ISBN)

IdRad	- цео број, идентификује рад, аутоматско додељивање наредног идентификатора
Naziv	- низ до 200 знакова, обавезно
ISBN	- низ од 17 знакова, у формату (ddd-dd-dddd-ddd-d где су d цифре),

RadCasopis (IdRad, PocStrana, BrStrana, IdCas)

IdRad	- цео број, идентификује рад у часопису, страни кључ (табела Rad)
PocStrana	- цео број, вредност 0 или 1
BrStrana	- цео број, обавезно, вредност већа од 0
IdCas	- страни кључ (табела Casopis), обавезно

RadKonferencija (IdRad, CeoRad, Datum, IdKon)

IdRad	- цео број, идентификује рад у часопису, страни кључ (табела Rad)
CeoRad	- цео број, обавезно, вредност већа од 0
Datum	- цео број, у формату (ууууmmdd)
IdKon	- страни кључ (табела Konferencija), обавезно

Напомена:

- значење атрибута CeoRad: 0 – објављен само апстракт, 1 – објављен цео рад

Autor (IdRad, IdOso, RB)

IdRad	- цео број, део примарног кључа, страни кључ (табела Rad), обавезно
IdOso	- цео број, део примарног кључа, страни кључ (табела Osoba), обавезно
RB	- цео број, обавезно, вредност већа од 0

Задатак 1 [4 поена]

Потребно је направити SQL упит који брише из базе све часописе у којима није објављен ниједан рад. Након претходног брисања потребно је исписати све часописе. Сортирати резултат по IdCas опадајуће.

Резултат дати у форми: IdCas, Naziv, Izdanje

У Сactus-у користити таб: Zadatak 1

```
DELETE FROM Casopis
WHERE IdCas NOT IN (SELECT idCas FROM RadCasopis);
```

```
SELECT * FROM Casopis ORDER BY IdCas DESC
```

Задатак 2 [4 поена]

Потребно је написати SQL упит који исписује све радове са конференција које су биле у току 10.1.2020. Сортирати резултат по IdRad растуће.

Резултат дати у форми: IdRad, NazivRada, DatumOd, DatumDo, NazivKonferencije

У Сactus-у користити таб: Zadatak 2

```
SELECT IdRad, Rad.Naziv AS NazivRada, DatumOd, DatumDo,
       Konferencija.Naziv AS NazivKonferencije
FROM Rad JOIN RadKonferencija USING(IdRad) JOIN Konferencija USING(IdKon)
WHERE DatumOd<=20200110 AND DatumDo>=20200110
ORDER BY IdRad
```

Задатак 3 [4 поена]

Потребно је направити SQL упит који за све особе приказује њихов степен образовања у текстуалном облику. Сортирати резултат прво по StepenObrazovanja растуће, па онда по IdOso растуће.

Резултат дати у форми: IdOso, Naziv, Email, Stepen obrazovanja

У Сactus-у користити таб: Zadatak 3

```
SELECT IdOso, Naziv, Email, CASE StepenObrazovanja
    WHEN 1 THEN 'osnovne akademske studije'
    WHEN 2 THEN 'master akademske studije'
    WHEN 3 THEN 'doktorske studije'
    WHEN 4 THEN 'postdoktorske studije'
    ELSE 'ostalo'
END AS "Stepen obrazovanja"
FROM Osoba
ORDER BY StepenObrazovanja, IdOso
```

Задатак 4 [4 поена]

Потребно је направити SQL скрипту која исписује часописе у чијем креирању садржаја је учествовало више од 6 особа. Сматра се да је особа учествовала у креирању садржаја часописа, ако је аутор бар на једном раду у том часопису. Сортирати резултат по називу растуће, а затим по издању растуће.

Резултат дати у форми: IdCas, Naziv, Izdanje
У Cactus-у користити таб: Zadatak 4

```
SELECT IdCas, Casopis.Naziv, Izdanje
FROM Casopis JOIN RadCasopis USING(IdCas) JOIN Autor USING(IdRad)
GROUP BY IdCas
HAVING COUNT(DISTINCT IdOso)>6
ORDER BY Casopis.Naziv, Izdanje
```

Задатак 5 [5 поена]

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

У Cactus-у користити таб: Zadatak 5

```
DROP TABLE IF EXISTS Osoba;
```

```
CREATE TABLE Osoba (
    IdOso  INTEGER PRIMARY KEY AUTOINCREMENT,
    Naziv  VARCHAR(50) NOT NULL,
    Email  VARCHAR(100) NOT NULL UNIQUE,
    StepenObrazovanja INTEGER NOT NULL
        CHECK(StepenObrazovanja>=1 AND StepenObrazovanja<=5)
);
```

Задатак 6 [5 поена]

Потребно је направити SQL скрипту која за све особе приказује на колико радова су били потписани као први аутор. У резултату приказати и особе које ни на једном раду нису биле потписане као први аутор. Сортирати резултат по IdOso растуће.

Резултат дати у форми: IdOso, Naziv, Broj
У Cactus-у користити таб: Zadatak 6
Није дозвољено коришћење погледа.

```
SELECT IdOso, Naziv, COUNT(IdRad) AS Broj
FROM Osoba LEFT JOIN (SELECT * FROM Autor WHERE RB=1) A USING (IdOso)
GROUP BY IdOso
ORDER BY IdOso, Naziv
```

Задатак 7 [5 поена]

Потребно је направити SQL упит који исписује све особе које су објавиле више радова у часописима него радова на конференцијама. Сортирати резултат по IdOso опадајуће.

Резултат дати у форми: IdOso, Naziv

У Сactus-у користити таб: Zadatak 7

Није дозвољено коришћење погледа.

```
SELECT IdOso, Naziv
FROM Osoba
WHERE ( SELECT COUNT(*) FROM Autor JOIN RadCasopis USING (IdRad)
        WHERE Autor.IdOso=Osoba.IdOso)
      >( SELECT COUNT(*) FROM Autor JOIN RadKonferencija USING (IdRad)
        WHERE Autor.IdOso=Osoba.IdOso)
ORDER BY IdOso DESC
```

Задатак 8 [6 поена]

Потребно је направити SQL скрипту која мења позиције првог и последњег аутора на раду са идентификатором IdRad=4. Тренутно првопотписаном аутору на раду је потребно доделити редни број тренутно последњег аутора на раду, док тренутно последњем потписаном аутору треба доделити редни број 1. Након претходних измена потребно је исписати све ауторе. Сортирати резултат по IdRad растуће, а затим по RB растуће.

Резултат дати у форми: IdRad, IdOso, RB

У Сactus-у користити таб: Zadatak 8

Није дозвољено коришћење погледа.

```
UPDATE Autor
SET RB = CASE
    WHEN RB=1 THEN (SELECT COUNT(*) FROM Autor WHERE IdRad=4)
    WHEN RB=(SELECT COUNT(*) FROM Autor WHERE IdRad=4) THEN 1
    ELSE RB END
WHERE IdRad=4;

SELECT * FROM Autor
ORDER BY IdRad, RB
```

Задатак 9 [6 поена]

Потребно је направити SQL упит који исписује све особе којима је Ердошев број непознат. Ердошев број представља сарадничку удаљеност одређеног научника од Пола Ердоша (Paul Erdos). Пол Ердош има Ердошев број 0. Ако је научник написао заједнички рад са Полом Ердошем, онда он има Ердошев број 1. Уколико је особа написала заједнички рад са неким научником који има Ердошев број 1 и није написала ниједан заједнички рад са Ердошем, та особа ће имати Ердошев број 2. По истом принципу се додељује Ердошев број и осталим научницима. Уколико је особа написала заједнички рад са неким научником који има Ердошев број X и нема заједнички рад ни са ким ко има Ердошев број мањи од X, та особа ће имати Ердошев број X+1. Ердошев број је непознат особама којима сарадничка удаљеност не може да се израчуна. Сортирати резултат по IdOso растуће.

Резултат дати у форми: IdOso, Naziv

У Sactus-у користити таб: Zadatak 9

Није дозвољено коришћење погледа.

```
WITH RECURSIVE ErdosevBroj(IdOso) AS
(
    SELECT IdOso FROM Osoba WHERE Naziv='Paul Erdos'
    UNION
    SELECT A2.IdOso
    FROM ErdosevBroj JOIN Autor A1 USING(IdOso)
        JOIN Autor A2 USING(IdRad)
)
SELECT IdOso, Naziv
FROM Osoba
WHERE IdOso NOT IN (SELECT IdOso FROM ErdosevBroj)
ORDER BY IdOso
```

Задатак 10 [7 поена]

Потребно је направити SQL упит који исписује све особе којима је Ердошев број једнак 5. Сортирати резултат по IdOso растуће.

Резултат дати у форми: IdOso, Naziv

У Sactus-у користити таб: Zadatak 10

Није дозвољено коришћење погледа.

```
WITH RECURSIVE ErdosNumber(IdOso, Broj) AS (
    SELECT IdOso, 0 FROM Osoba WHERE Naziv='Paul Erdos'
    UNION
    SELECT A2.IdOso, Broj+1
    FROM ErdosNumber JOIN Autor A1 USING(IdOso) JOIN Autor A2 USING(IdRad)
        WHERE Broj<5
)
SELECT IdOso, Naziv
FROM ErdosNumber JOIN Osoba USING(IdOso)
GROUP BY IdOso
HAVING MIN(Broj)=5
ORDER BY IdOso
```
