



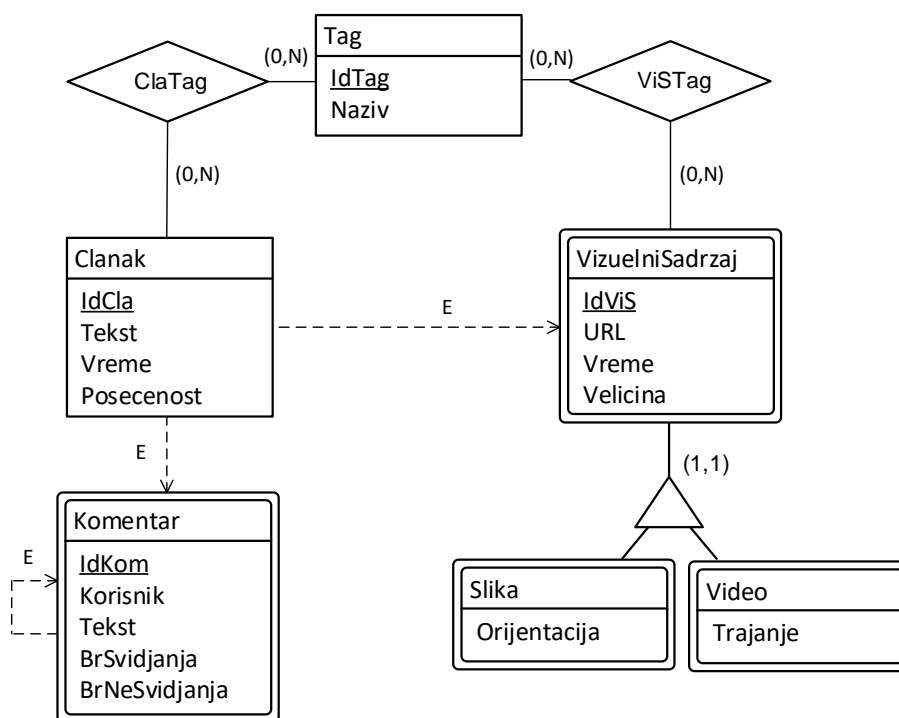
Базе података 1

- августовски испитни рок -

Група А

Посматра се део базе података веб портала за објављивање вести. У овој бази се чувају подаци о ауторима, чланцима и њиховим садржајем, коментарима и таговима. Коментари се дају на неки чланак, али је могуће и да се коментар да на већ неки претходни коментар истог чланка. У табелама је коришћен формат времена у облику гтгг.мес.дд хх:мм:сс, где гтгг представља четвороцифрену годину, мес представља трокарактерни назив месеца (могуће вредности: JAN, FEB, MAR, APR, MAJ, JUN, JUL, AVG, SEP, OKT, NOV, DEC), дд представља двоцифрени дан, хх представља двоцифрени сат, мм представља двоцифрен минут, сс представља двоцифрену секунду.

У наставку је дата релациона шема посматраног дела базе податка.



Clanak (IdCla, Tekst, Vreme, Posecenost)

- | | |
|------------|----------------------------------------------------------------------------------------------------|
| IdCla | - цео број, идентификује чланак, аутоматско додељивање наредног идентификатора |
| Tekst | - низ до 500 знакова, обавезно |
| Vreme | - низ од тачно 20 знакова, обавезно, формат (гтгг.мес.дд хх:мм:сс), представља време објаве чланка |
| Posecenost | - цео број, обавезно, вредност већа од 0 |

Tag (IdTag, Naziv)

IdTag	- цео број, идентификује таг, аутоматско додељивање наредног идентификатора
Naziv	- низ до 50 знакова, јединствено, обавезно

Komentar (IdKom, Korisnik, Tekst, BrSvidjanja, BrNeSvidjanja, IdCla, IdKomNa)

IdKom	- цео број, идентификује коментар, аутоматско додељивање наредног идентификатора
Korisnik	- низ до 50 знакова, обавезно
Tekst	- низ до 150 знакова, обавезно
BrSvidjanja	- цео број, обавезно, вредност већа или једнака од 0
BrNeSvidjanja	- цео број, обавезно, вредност већа или једнака од 0
IdCla	- страни кључ (табела Clanak), обавезно
IdKomNa	- страни кључ (табела Komentar)

Напомена:

Када коментар има NULL вредност за IdKomNa, то значи да је коментар првог нивоа (коментар дат директно на чланак). Када коментар име различиту вредност од NULL за IdKomNa, то значи да је коментар вишег нивоа (за један већи ниво у односу на коментар са идентификаторим IdKomNa).

VizuelniSadrzaj (IdViS, URL, Vreme, Velicina, IdCla)

IdViS	- цео број, идентификује визуелни садржај, аутоматско додељивање наредног идентификатора
URL	- низ до 150 знакова, обавезно
Vreme	- низ од тачно 20 знакова, формат (гггг.мес.дд хх:мм:сс), представља време настанка визуелног садржаја
Velicina	- цео број, вредност већа од 0, представља величину визуелног садржаја дата у бајтовима
IdCla	- страни кључ (табела Clanak), обавезно

Напомена:

URL представља путању до датотеке са визуелним садржајем. Путања се састоји из ниједног или више директоријума (одвојених специјалним карактером '\'), заједно са називом датотеке са екстензијом (формат URL: \dir1\dir2\...\dirX\file.ext). Сматрати да назив датотеке мора да има бар један карактер и да увек има екстензију.

Slika (IdViS, Orijentacija)

IdViS	- страни кључ (табела VizuelniSadrzaj), обавезно
Orijentacija	- карактер, могуће вредности 'H' и 'V', представља оријентацију слике (H – хоризонтална, V – вертикална)

Video (IdViS, Trajanje)

IdViS	- страни кључ (табела VizuelniSadrzaj), обавезно
Trajanje	- цео број, вредност већа од 0, представља трајање видео записа дато у секундама

ClaTag (IdCla, IdTag)

IdCla	- страни кључ (табела Clanak), обавезно
IdTag	- страни кључ (табела Tag), обавезно

ViSTag (IdViS, IdTag)

IdViS	- страни кључ (табела VizuelniSadrzaj), обавезно
IdTag	- страни кључ (табела Tag), обавезно

Задатак 1 [3 поена]

Потребно је направити SQL упит који исписује све видео записе чија екстензија није *.mp4*, а да се при томе налази на путањи */video/_tmp/<file>* (*<file>* представља назив датотеке са екстензијом). Резултат треба сортирати опадајуће по URL.

Резултат дати у форми: IdViS, URL
У Сactus-у користити таб: Zadatak 1

```
SELECT IdViS, URL
FROM VizuelniSadrzaj NATURAL JOIN Video
WHERE URL LIKE "/video/_tmp/_%._%" ESCAPE '!' AND URL NOT LIKE "%.mp4"
ORDER BY URL DESC
```

Задатак 2 [3 поена]

Потребно је направити SQL упит који исписује све хоризонталне слике из скупштине (имају таг *skupština*) које су веће од 5MB (1MB = 1024*1024B). Резултат треба сортирати опадајуће по величини, а затим растуће по идентификатору.

Резултат дати у форми: IdViS, URL, Vreme, Velicina
У Сactus-у користити таб: Zadatak 2

```
SELECT IdViS, URL, Vreme, Velicina
FROM Tag NATURAL JOIN ViSTag NATURAL JOIN VizuelniSadrzaj NATURAL JOIN Slika
WHERE Tag.Naziv = 'skupština' AND VizuelniSadrzaj.Velicina > 5*1024*1024 AND Slika.Orijentacija = 'H'
ORDER BY VizuelniSadrzaj.Velicina DESC, VizuelniSadrzaj.IdViS
```

Задатак 3 [3 поена]

Потребно је направити SQL упит који исписује статистику везану за визуелни садржај. Потребно је приказати колико има слика и колико има видео записа категоризованих по величини у следеће категорије: “0MB - 5MB”, “5MB - 50MB” и “50MB+”. У случају да се величина визуелног садржаја налази на граници категорија, такав визуелни садржај треба урачунати у вишу категорију. Резултат треба да буде формиран у три реда – категорије. У првом реду треба да буде категорија “0MB - 5MB”, у другом реду “5MB - 50MB”, а у трећем реду “50MB+”.

Резултат дати у форми: Grupa, Broj slika, Broj video zapisa
У Сactus-у користити таб: Zadatak 3

```
SELECT '0MB - 5MB' AS Grupa,
(SELECT COUNT (*) FROM VizuelniSadrzaj NATURAL JOIN Slika WHERE Velicina >= 0 AND
Velicina < 5*1024*1024) AS "Broj slika",
(SELECT COUNT (*) FROM VizuelniSadrzaj NATURAL JOIN Video WHERE Velicina >= 0 AND
Velicina < 5*1024*1024) AS "Broj video zapisa"

UNION ALL
```

```
SELECT '5MB - 50MB',  
(SELECT COUNT (*) FROM VizuelniSadrzaj NATURAL JOIN Slika WHERE Velicina >= 5*1024*1024  
AND Velicina < 50*1024*1024),  
(SELECT COUNT (*) FROM VizuelniSadrzaj NATURAL JOIN Video WHERE Velicina >= 5*1024*1024  
AND Velicina < 50*1024*1024)
```

UNION ALL

```
SELECT '50MB+',  
(SELECT COUNT (*) FROM VizuelniSadrzaj NATURAL JOIN Slika WHERE Velicina >=  
50*1024*1024),  
(SELECT COUNT (*) FROM VizuelniSadrzaj NATURAL JOIN Video WHERE Velicina >=  
50*1024*1024)
```

Задатак 4 [3 поена]

Потребно је направити SQL упит који исписује за сваки чланак који има већу посећеност од просека колико има коментара (било ког нивоа) од различитих корисника. Резултат сортирати по пребројаним корисницима опадајуће, а затим по идентификатору чланка растуће.

Резултат дати у форми: IdCla, BrojKorisnika
У Cactus-у користити таб: Zadatak 4

```
SELECT IdCla, COUNT (DISTINCT Korisnik) AS BrojKorisnika  
FROM Clanak LEFT OUTER JOIN Komentar USING (IdCla)  
WHERE Posecenost > (SELECT AVG(Posecenost) FROM Clanak)  
GROUP BY IdCla  
ORDER BY BrojKorisnika DESC, IdCla ASC
```

Задатак 5 [4 поена]

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

У Cactus-у користити таб Zadatak 5.

```
DROP TABLE IF EXISTS Komentar;
```

```
CREATE TABLE Komentar (  
    IdKom INTEGER PRIMARY KEY AUTOINCREMENT,  
    Korisnik VARCHAR(50) NOT NULL,  
    Tekst VARCHAR(150) NOT NULL,  
    BrSvidjanja INTEGER NOT NULL CHECK (BrSvidjanja >= 0),  
    BrNeSvidjanja INTEGER NOT NULL CHECK (BrNeSvidjanja >= 0),  
    IdCla INTEGER NOT NULL REFERENCES Clan(IdCla),  
    IdKomNa INTEGER REFERENCES Komentar(IdKom)  
);
```

Задатак 6 [4 поена]

Потребно је направити SQL упит који исписује све чланке које имају бар 3 видео записа дужа од 4 минута и мање од 5 слика. Резултат треба сортирати по идентификатору чланка растуће.

Резултат дати у форми: IdCla, Tekst

У Сactus-у користити таб: Zadatak 6

Није дозвољено коришћење погледа.

```
WITH Clanci3VideoVece4Min (IdCla) AS (  
    SELECT IdCla  
    FROM VizuelniSadrzaj NATURAL JOIN Video  
    WHERE Trajanje>4*60  
    GROUP BY IdCla  
    HAVING COUNT(*) >= 3  
)  
ClanciVise4Slika (IdCla) AS (  
    SELECT IdCla  
    FROM VizuelniSadrzaj NATURAL JOIN Slika  
    GROUP BY IdCla  
    HAVING COUNT(*) >=4  
)  
SELECT IdCla, Tekst  
FROM Clanak  
WHERE IdCla IN (SELECT IdCla FROM Clanci3VideoVece4Min)  
AND IdCla NOT IN (SELECT IdCla FROM ClanciVise4Slika)  
ORDER BY IdCla
```

Задатак 7 [4 поена]

Потребно је направити SQL упит који исписује све чланке из 2019. године заједно са бројем коментара првог нивоа (коментари који су директно везани за чланак). Резултат сортирати по месецу растуће (прво чланци из јануара, па фебруара, па марта итд.), па по броју директних коментара опадајуће, а затим по идентификатору чланка растуће.

Резултат дати у форми: IdCla, Tekst, Vreme, BrKomentara

У Сactus-у користити таб: Zadatak 7

Није дозвољено коришћење погледа.

```
WITH ClanakMesec2019 (IdCla, Tekst, Vreme, Mesec) AS (  
    SELECT IdCla, Tekst, Vreme,  
        CASE WHEN Vreme LIKE '2019.JAN.____:__:__' THEN 1  
        WHEN Vreme LIKE '2019.FEB.____:__:__' THEN 2  
        WHEN Vreme LIKE '2019.MAR.____:__:__' THEN 3  
        WHEN Vreme LIKE '2019.APR.____:__:__' THEN 4  
        WHEN Vreme LIKE '2019.MAJ.____:__:__' THEN 5  
        WHEN Vreme LIKE '2019.JUN.____:__:__' THEN 6  
        WHEN Vreme LIKE '2019.JUL.____:__:__' THEN 7  
        WHEN Vreme LIKE '2019.AVG.____:__:__' THEN 8
```

```

        WHEN Vreme LIKE '2019.SEP.____:__:__' THEN 9
        WHEN Vreme LIKE '2019.OKT.____:__:__' THEN 10
        WHEN Vreme LIKE '2019.NOV.____:__:__' THEN 11
        WHEN Vreme LIKE '2019.DEC.____:__:__' THEN 12
        ELSE -1 END
    FROM Clanak
    WHERE Vreme LIKE '2019.____.____:__:__'
)
SELECT cm2019.IdCla, cm2019.Tekst, cm2019.Vreme,
       (SELECT COUNT(*) FROM Komentar WHERE Komentar.IdCla=cm2019.IdCla AND
        Komentar.IdKomNa IS NULL) AS BrKomentara
FROM ClanakMesec2019 cm2019
ORDER BY cm2019.Mesec, BrKomentara DESC, IdCla

```

Задатак 8 [4 поена]

Потребно је направити SQL упит који исписује све парове чланака који имају међусобно идентичан скуп тагова. Игнорисати оне чланке који немају тагове. При исписивању парова чланака потребно је приказати само њихове идентификаторе (уређени пар – (IdCla1, IdCla2)) тако да се парови не понављају и где је IdCla1 мањи од IdCla2. Резултат сортирати растуће прво по IdCla1, а затим по IdCla2

Резултат дати у форми: IdCla1, IdCla2

У Sactus-у користити таб: Zadatak 8

Није дозвољено коришћење погледа.

```

SELECT ct1.IdCla AS IdCla1, ct2.IdCla AS IdCla2
FROM ClaTag ct1 JOIN ClaTag ct2 USING (IdTag)
WHERE ct1.IdCla<ct2.IdCla
GROUP BY ct1.IdCla, ct2.IdCla
HAVING      COUNT (*) = (SELECT COUNT(*) FROM ClaTag WHERE ClaTag.IdCla = ct1.IdCla)
          AND COUNT(*) = (SELECT COUNT(*) FROM ClaTag WHERE ClaTag.IdCla = ct2.IdCla)

```

Задатак 9 [5 поена]

Потребно је направити SQL скрипту која додаје редом у базу тагове са називима 'video' и 'slika' (сматрати да наведени тагови не постоје у бази). При додавању тагова у базу идентификатор треба да буде за један већи од претходног (сматрати да аутоматско додељивање идентификатора није активирано). Затим је потребно свакој слици у бази доделити таг 'slika' и сваком видео запису у бази треба доделити таг 'video'. Написати упит који враћа за сваки визуелни садржај заједно са свим таговима тог визуелног садржаја. Сортирати резултат прво по идентификатору визуелног садржаја растуће, а затим растуће по идентификатору тага.

Резултат дати у форми: IdVis, IdTag, Naziv

У Sactus-у користити таб: Zadatak 9

Није дозвољено коришћење погледа.

```
INSERT INTO Tag (IdTag, Naziv) VALUES ( (SELECT MAX(IdTag)+1 FROM Tag), 'video');
INSERT INTO Tag (IdTag, Naziv) VALUES ( (SELECT MAX(IdTag)+1 FROM Tag), 'slika');
```

```
INSERT INTO ViSTag (IdViS,IdTag)
SELECT IdViS, (SELECT MAX(IdTag) FROM Tag WHERE Naziv='video')
FROM Video;
```

```
INSERT INTO ViSTag (IdViS,IdTag)
SELECT IdViS, (SELECT MAX(IdTag) FROM Tag WHERE Naziv='slika')
FROM Slika;
```

```
SELECT IdViS, IdTag, Naziv
FROM VisTag NATURAL JOIN Tag
ORDER BY IdVis, IdTag
```

Задатак 10 [5 поена]

Потребно је направити SQL скрипту која приказује све коментаре за чланак са највећим идентификатором, укључујући и коментаре на коментаре тог чланка. Коментаре треба сортирати тако да буду међусобно угњеждени (по дубини нивоа), а унутар нивоа треба да буду сортирани по броју свиђања опадајуће, броју не свиђања растуће, а након тога по идентификатору коментара растуће. При испису потребно је исписати и ниво у коме се налази коментар.

Резултат дати у форми: Nivo, IdKom, Korisnik, Tekst, BrSvidjanja, BrNeSvidjanja

У Sactus-у користити таб: Zadatak 10

Није дозвољено коришћење погледа

```
WITH RECURSIVE KomentariIspis (Nivo, IdKom, Korisnik, Tekst, BrSvidjanja, BrNeSvidjanja) AS
(
    SELECT 1, IdKom, Korisnik, Tekst, BrSvidjanja, BrNeSvidjanja
    FROM Komentar
    WHERE IdCla = (SELECT MAX(IdCla) FROM Clanak) AND IdKomNa IS NULL
    UNION ALL
    SELECT ki.Nivo + 1 AS Nivo, k.IdKom, k.Korisnik, k.Tekst, k.BrSvidjanja, k.BrNeSvidjanja
    FROM KomentariIspis ki JOIN Komentar k ON (ki.IdKom = k.IdKomNa)
    ORDER BY Nivo DESC, BrSvidjanja DESC, BrNeSvidjanja ASC, k.IdKom ASC
)
SELECT * FROM KomentariIspis
```

Задатак 11 [6 поена]

Потребно је направити SQL упит који исписује до 5 чланака који су највише повезани са чланком који има идентификатор 1 (у тих 5 не спада чланак са идентификатором 1). Коефицијент на основу које се рачуна повезаност чланака се добија на основу тагова које поседују, као и тагови које поседују њихови визуелни садржаји. За сваки исти таг који чланци међусобно имају коефицијент се увећава за 3. За сваки заједнички таг њихових визуелних садржаја коефицијент се увећава за 1 (на основу једног заједничког тага визуелног садржаја могуће је увећати коефицијент максимално за 1). Резултат треба сортирати по коефицијенту у опадајућем поретку, а затим по идентификатору чланка растуће.

Резултат дати у форми: IdCla, Tekst, Koeficijent

У Sactus-у користити таб: Zadatak 11

Није дозвољено коришћење погледа.

```
WITH ClanakTagoviKoef (IdCla, Koeficijent) AS
(
    SELECT ct2.IdCla, COUNT(*) * 3
    FROM ClaTag ct1, ClaTag ct2
    WHERE ct1.IdCla = 1 AND ct1.IdTag=ct2.IdTag
    GROUP BY ct2.IdCla
),
VisTagoviKoef (IdCla, Koeficijent) AS
(
    SELECT vs.IdCla, COUNT(DISTINCT vt.IdTag) * 1
    FROM VizuelniSadrzaj vs, ViSTag vt
    WHERE vs.IdCla != 1 AND vs.IdVis=vt.IdVis AND vt.IdTag IN (
        SELECT DISTINCT vt1.IdTag
        FROM VizuelniSadrzaj vs1, ViSTag vt1
        WHERE vs1.IdCla = 1 AND vs1.IdVis=vt1.IdVis)
    GROUP BY vs.IdCla
)
SELECT IdCla, Tekst, COALESCE( ClanakTagoviKoef.Koeficijent, 0 ) +
COALESCE( VisTagoviKoef.Koeficijent, 0 ) AS Koeficijent
FROM Clanak LEFT OUTER JOIN ClanakTagoviKoef USING (IdCla)
LEFT OUTER JOIN VisTagoviKoef USING (IdCla)
WHERE Clanak.IdCla!=1
ORDER BY Koeficijent DESC, IdCla
LIMIT 5
```

Задатак 12 [6 поена]

Расправа о чланку представља листу свих коментара почев од коментара првог нивоа (директан коментар на чланак) па све до последњег (највишег нивоа) коментара на коментар. Добра расправа је расправа код које је укупан број свиђања у коментарима расправе већи него укупан број не свиђања у коментарима те расправе. Један чланак може да има више добрих расправа (које могу чак да деле један део коментара). Лоша расправа је она која није добра. Потребно је обрисати све коментаре који припадају лошим расправама и не припадају добрим расправама, а затим исписати преостали садржај табеле **Komentar**. Сортирати резултат по идентификатору коментара растуће.

Резултат дати у форми: IdKom, Korisnik, Tekst, BrSvidjanja, BrNeSvidjanja, IdCla, IdKomNa
У Cactus-у користити таб: Zadatak 12

```
WITH RECURSIVE Rasprava (IdKom, BrSvidjanja, BrNeSvidjanja, PoslednjiKomentar) AS (  
SELECT IdKom, BrSvidjanja, BrNeSvidjanja, (CASE WHEN (SELECT COUNT(*) FROM Komentar  
WHERE Komentar.IdKomNa=k.IdKom)>0 THEN 0 ELSE 1 END )  
FROM Komentar k WHERE IdKomNa IS NULL  
  
UNION ALL  
  
SELECT k.IdKom, k.BrSvidjanja+r.BrSvidjanja, k.BrNeSvidjanja+r.BrNeSvidjanja, (CASE      WHEN  
(SELECT COUNT(*) FROM Komentar WHERE Komentar.IdKomNa=k.IdKom)>0      THEN 0 ELSE 1  
END )  
FROM Komentar k, Rasprava r WHERE k.IdKomNa = r.IdKom  
)  
,  
DobraRasprava (IdKom) AS (  
SELECT IdKom FROM Rasprava WHERE PoslednjiKomentar=1 AND BrSvidjanja>BrNeSvidjanja  
UNION  
SELECT Komentar.IdKomNa FROM Komentar NATURAL JOIN DobraRasprava  
WHERE Komentar.IdKomNa IS NOT NULL  
)  
  
DELETE FROM Komentar  
WHERE IdKom NOT IN (SELECT IdKom FROM DobraRasprava);  
  
SELECT * FROM Komentar ORDER BY IdKom
```
