

Vak: Databases & Applicaties (DbAp)
 Docent:
 Klas: Deeltijd-semester BIS
 Datum: oktober 2016

Invullen:

Naam: Stud. Nr: Klas:

OPGAVEN NA AFLOOP WEER INLEVEREN!!

Hulpmiddelen: - Boek "Relationele Databases en SQL" (L.Wiegerink e.a.)

- Beschikbare tijd: 90 minuten
- De toets bestaat uit 6 opgaven waarmee totaal 50 punten zijn te verdienen
- Werk de uitwerkingen uit op dit opgavenblad bij de opgaven en lever aan het eind deze opgaven met uitwerkingen en evt. kladpapier weer in!
- Het gebruik van elektronica zoals laptop en mobiele telefoon is NIET toegestaan.
- Het gebruik van het boek "Relationele Databases en SQL" (L.Wiegerink e.a.) is toegestaan.
- De database die bij de toets wordt gebruikt is de database Muziekschool. Een databaseschema met voorbeeldpopulatie zijn gegeven in de bijlagen van deze toets. Alle antwoorden moeten gelden voor een willekeurige vulling van de database: er is een voorbeeld gegeven van een mogelijke vulling!

Vraag 1 (10 punten)

Geef een SELECT-statement voor de volgende informatiebehoefte:

Welke componist heeft twee klassieke stukken gecomponeerd? Geef de componist id, naam van de componist en het aantal stukken dat deze gecomponeerd heeft. Orden van groot naar klein op dit aantal.

Uitwerking:

```
SELECT C.componistId , naam , COUNT(*) AS aantal
FROM Componist C
      INNER JOIN Stuk S ON C.componistId = S.componistId
WHERE S.genrenaam = 'klassiek'
GROUP BY C.componistId , naam
HAVING COUNT(*) = 2
ORDER BY 3 DESC
```

Fout in:

- SELECT: verkeerde kolommen -1 per kolom
 prefix C. bij C.componistid vergeten -1
 andere statistische functie dan COUNT -1
- FROM: LEFT JOIN ipv INNER JOIN ook goed
 Fout in join -3
- WHERE: vergeten of andere conditie -2
 " vergeten -1
- GROUP BY: helemaal vergeten -3
 kolom vergeten -2
- HAVING: helemaal vergeten -3
 verkeerde conditie (op ander kolom) of verkeerde stat functie -2, fout in grenswaarde -1

ORDER BY vergeten -2 , verkeerde kolom -1, DESC vergeten -1

Vraag 2 (10 punten)

Geef een SELECT-statement voor de volgende informatiebehoefte:

Wie is/zijn de oudste componist(en) die verbonden is/zijn aan een muziekschool? Geef van deze componist(en) de naam en de geboortedatum.

Uitwerking:

```
SELECT naam, geboortedatum
FROM Componist
WHERE geboortedatum =
                                (SELECT MIN(geboortedatum)
                                FROM Componist
                                WHERE schoolid IS NOT NULL )
AND schoolid IS NOT NULL
```

Of

```
SELECT naam, geboortedatum
FROM Componist
WHERE geboortedatum <= ALL
                                (SELECT geboortedatum
                                FROM Componist
                                WHERE schoolid IS NOT NULL )
AND schoolid IS NOT NULL
```

Fout in:

SELECT: verkeerde kolommen -1 per kolom

FROM: indien hier muziekschool gejoint dan hoeft extra
 where clause 'AND schoolid IS NOT NULL' niet

WHERE: geboortedatum = MIN(geboortedatum) – 4
 SUBQUERY: SELECT: fout in MIN -2
 (fout in >= ALL -2)

FROM: zie hoofdquery

WHERE: vergeten 'AND schoolid IS NOT NULL' in subquery -2

Geen GROUP BY hier!, wel aanwezig dan -3

AND: vergeten 'AND schoolid IS NOT NULL' in hoofdquery -2

Vraag 3 (10 punten)

Geef een SELECT-statement voor de volgende informatiebehoefte:

Geef een lijst van alle stukken waarin geen piano in voor komt. Toon stuknummer, titel van het stuk en de naam van de componist van het stuk.

Uitwerking

```
SELECT stuknr, titel, naam
FROM Stuk S
      INNER JOIN Componist C ON S.componistId = C.componistid
WHERE stuknr NOT IN (SELECT stuknr
                     FROM Bezettingsregel
                     WHERE instrumentnaam = 'piano')

--niet goed is
SELECT stuknr, titel, naam
FROM Stuk S
      INNER JOIN Componist C ON S.componistId = C.componistid
WHERE stuknr IN (SELECT stuknr
                 FROM Bezettingsregel
                 WHERE instrumentnaam <> 'piano')

--niet goed is
SELECT S.stuknr, titel, naam
FROM Stuk S
      INNER JOIN Componist C ON S.componistId = C.componistid
      INNER JOIN Bezettingsregel B ON S.stuknr = B.stuknr
WHERE instrumentnaam <> 'piano'
GROUP BY S.stuknr, titel, naam
- Wel goed bij gebruik van DISTINCT om dubbele te voorkomen
```

Fout in:

SELECT: verkeerde kolommen -1 per kolom

FROM: fout in JOIN -2

WHERE: fout met NOT IN -4

SUBQUERY:

SELECT: geen stuknr hier: -2

WHERE: fout in conditie -2

Vraag 4 (4 punten)

Geef het resultaat van de volgende query, uitgaande van de inhoud van de tabellen **Muziekschool** en **Componist** die in bijlage B is gegeven.

```
SELECT Muziekschool.SchoolId, Muziekschool.Naam, Componist.Naam
FROM   Componist RIGHT OUTER JOIN Muziekschool
      ON Muziekschool.SchoolId = Componist.SchoolId
```

Uitwerking:

MUZIEKSCHOOL

schoolId	naam	naam
1	Muziekschool Amsterdam	Thom Guidi
1	Muziekschool Amsterdam	Jan van Maanen
2	Reijnders' Muziekschool	Sofie Bergeijk
2	Reijnders' Muziekschool	Karl Schumann
3	Het Muziekpakhuis	NULL

Fout in:

Muziekschool 3 vergeten: -2, fout in NULL waarden hier ook -2

Fout in andere records: -1 per fout

Componisten getoond zonder muziekschool: -2 (RIGHT ALS LEFT gelezen)

Vraag 5 (A: 6 punten, B: 2 punten, C: 2 punten)

Schrijf DDL-opdrachten om de volgende twee tabellen toe te voegen aan de Muziekdatabase uit de bijlage. Alle gegeven constraints moeten worden geïmplementeerd. Bij de vreemde sleutelverwijzingen dienen de gegeven update- en delete-regels te worden gespecificeerd. De opdrachten dienen in uitvoerbare volgorde te staan.

Lid

lidnr integer NN PK	voornaam varchar(20) OP	tussenvoegsels varchar(10) OP	achternaam varchar(30) NN	geslacht varchar(5) NN
------------------------------	-------------------------------	-------------------------------------	---------------------------------	------------------------------

FK's: geen

LidSpeelt

lidnr integer NN	instrumentnaam varchar(14) NN	toonhoogte varchar(7) NN	niveaucode char(1) NN
------------------------	-------------------------------------	--------------------------------	-----------------------------

FK's:

LidSpeelt (lidnr) → Lid (lidnr), bij update: cascade, bij delete: cascade

LidSpeelt (instrumentnaam en toonhoogte) → Instrument (instrumentnaam en toonhoogte), bij update: cascade, bij delete: restrict

LidSpeelt (niveaucode) → Niveau (niveaucode), bij update: cascade, bij delete: restrict

- Geef de SQL-instructies om de tabellen **Lid** en **LidSpeelt** aan te maken. Realiseer ook de in het bovenstaande schema aangegeven constraints (o.a de primaire en de vreemde sleutel).
- Geef de SQL-instructies die de volgende constraint implementeert:
“voor de tabel **Lid** moet het geslacht ‘Man’ of ‘Vrouw’ zijn”
- Geef een INSERT-statement voor de tabel **Lid** waarmee uitsluitend de bovenstaande constraint “*geslacht moet ‘Man’ of ‘Vrouw’ zijn*” wordt geschonden

Uitwerking:

A.

--Opgave

```

CREATE TABLE Lid
(
    Lidnr            INT    NOT NULL,
    Voornaam         VARCHAR(20) NULL,
    Tussenvoegsels   VARCHAR(10) NULL,
    Achternaam       VARCHAR(30) NOT NULL,
    Geslacht         VARCHAR(5)  NOT NULL,
    CONSTRAINT PK_Lid PRIMARY KEY (Lidnr),
)

CREATE TABLE LidSpeelt
(
    Lidnr            INT    NOT NULL,
    InstrumentNaam   VARCHAR(14) NOT NULL,
    ToonHoogte       VARCHAR(7) NOT NULL,
    NiveauCode       CHAR(1) NOT NULL,
    CONSTRAINT PK_LidSpeelt PRIMARY KEY (Lidnr, InstrumentNaam, ToonHoogte),
    --LidSpeelt (lidnr) -> Lid (lidnr),
    --bij update: cascade, bij delete: cascade
    CONSTRAINT FK_LidSpeelt_Lid
        FOREIGN KEY (Lidnr)
        REFERENCES Lid(Lidnr)
        ON UPDATE CASCADE ON DELETE CASCADE,
    --LidSpeelt (instrumentnaam en toonhoogte)
    -- -> Instrument (instrumentnaam en toonhoogte),
    --bij update: cascade, bij delete: restrict
    CONSTRAINT FK_LidSpeelt_Instrument
        FOREIGN KEY (InstrumentNaam, ToonHoogte)
        REFERENCES Instrument(InstrumentNaam, ToonHoogte)
        ON UPDATE CASCADE ON DELETE NO ACTION
)

--LidSpeelt (niveaucode)
-- -> Niveau (niveaucode),
-- bij update: cascade, bij delete: restrict
--Overige constraints: geen
ALTER TABLE LidSpeelt
ADD CONSTRAINT FK_LidSpeelt_Niveau
FOREIGN KEY (NiveauCode)
REFERENCES Niveau(NiveauCode)
ON UPDATE CASCADE ON DELETE NO ACTION

```

Fout in CREATE TABLE (max 4 fouten in create tables):

- foute kolomnaam, datatype of null/not null -1 per fout
- fout in PK definitie -2
- fout in FK definitie:
 - o fout in tabel, kolom def. per FK-2
 - o fout in ON UPDATE/DELETE -1 (ook als NO ACTION vergeten is, max 1 * fout tellen)

B. ALTER TABLE Lid ADD CONSTRAINT CHK_Geslacht CHECK (Geslacht IN ('Man', 'Vrouw'))

Fout in CHECK: -2

Gebruik van hoofdletters/kleine letters negeren

GESLACHT = 'Man' OR Geslacht = 'Vrouw' ook goed

Ook gecheckt op NOT NULL : -1

C. INSERT INTO Lid(Lidnr, Voornaam, Tussenvoegsels, Achternaam, Geslacht)
VALUES (1, NULL, NULL, 'B', 'iets')

Alles of niets: -2

Vraag 6 (A: 3 punten; B: 3 punten)

*Opmerking: Neem aan dat de inhoud van de tabellen **Stuk** en **Bezettingsregel** is zoals weergegeven in bijlage B.*

Ga ervan uit dat de volgende opdracht is uitgevoerd:

```
ALTER TABLE Bezettingsregel
ADD CONSTRAINT fk_Bezettingsregel_Stuk
FOREIGN KEY (stuknr) REFERENCES Stuk(stuknr)
ON UPDATE CASCADE ON DELETE NO ACTION
```

A. Wat gebeurt er als je vervolgens het volgende UPDATE-statement uitvoert?

```
UPDATE   Stuk
SET      stuknr = 20
WHERE    stuknr = 9
```

Beschrijf voor dit UPDATE-statement de veranderingen in de tabellen **Stuk** en **Bezettingsregel** en motiveer dit met de constraints die wel/niet worden geschonden .

B. Wat gebeurt er als je vervolgens het volgende DELETE-statement uitvoert?

```
DELETE
FROM     Stuk
WHERE    stuknr = 8
```

Beschrijf voor dit UPDATE-statement de veranderingen in de tabellen **Stuk** en **Bezettingsregel** en motiveer dit met de constraints die wel/niet worden geschonden .

Uitwerking

```
UPDATE      Stuk
SET          stuknr = 20
WHERE       stuknr = 9
```

- A. In de tabel Stuk wordt het stuknummer van Stuk 9 in 20 gewijzigd. Doordat de foreign key `fk_Bezettingsregel_Stuk` als `ON UPDATE CASCADE` is gedefinieerd worden ook de gerelateerde 3 records in tabel `Bezettingsregel` geupdate. Voor deze regels wordt het stuknummer van 9 ook naar 20 gewijzigd.

Fout: fk en on update cascade vergeten te vermelden: -2 (de motivatie ontbreekt dus), vergeten te melden dat de 3 records in `Bezettingsregel` worden geupdate: -2

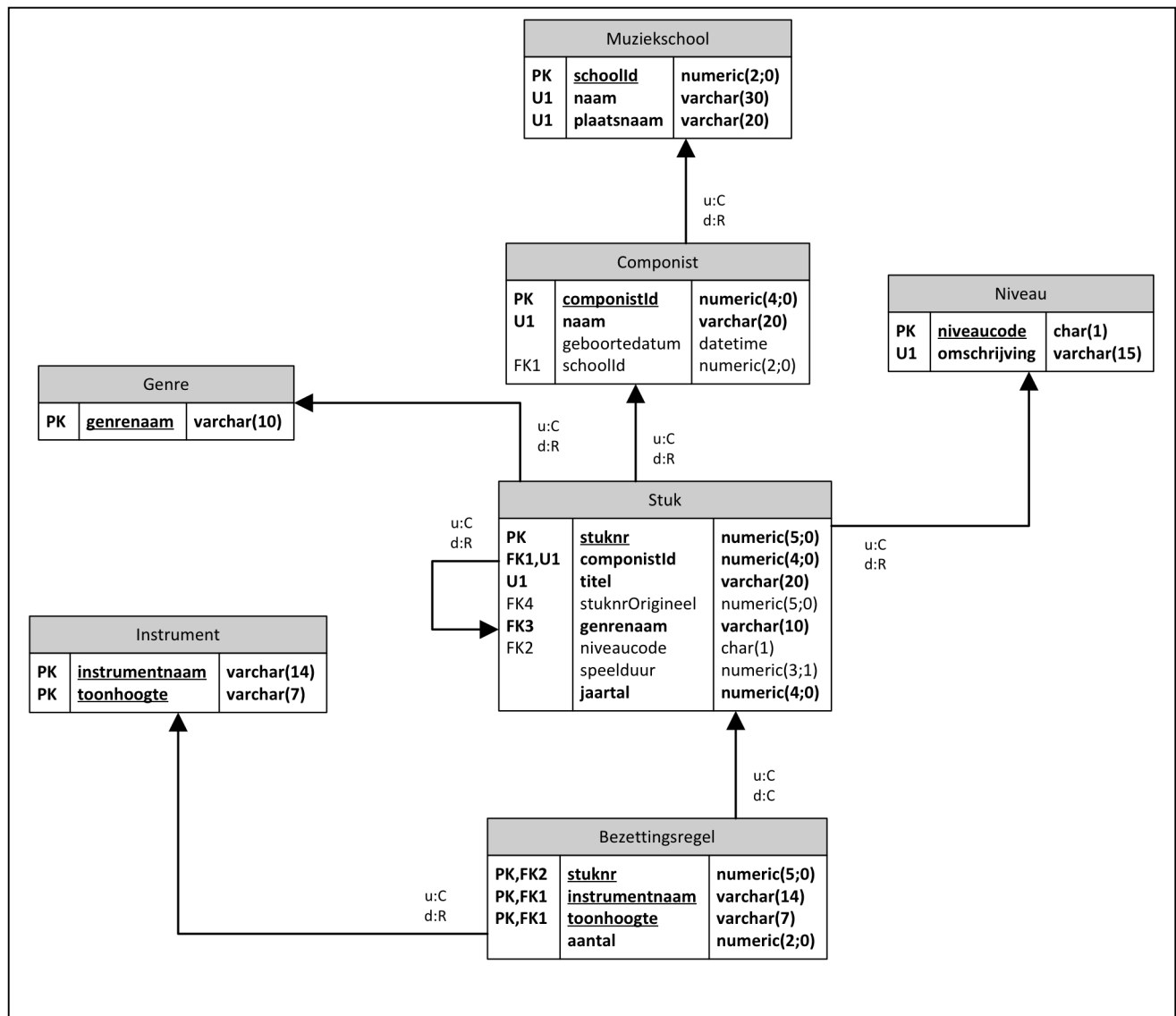
Niets goed: -3

```
DELETE
FROM      Stuk
WHERE     stuknr = 8
```

- B. In de tabel Stuk wordt het record van stuknummer 8 NIET verwijderd. Stuknummer 8 bestaat niet in de tabel `Bezettingregel`, de cascading rules van die foreign key `Bezettingsregel(Stuknr) -> Stuk(Stuknr)` hoeven dus niet verder geanalyseerd te worden. Echter, er bestaat ook een selfreference van `Stuk(StuknrOrigineel) -> Stuk(Stuknr)`, deze foreign key is gedefinieerd als `ON DELETE NO ACTION` (oftewel restricted) en aangezien stuknr 8 wel als `stuknrorigineel` voorkomt (bij stuknr 10) gaat deze delete dus niet door.

Fout: niet vermeld dat er gerelateerde records zijn: -2 (de motivatie waarom het niet doorgaat ontbreekt dus)

Niets goed: -3

Bijlage A: Logisch Relationeel Schema (overzicht)

Bij alle vreemde sleutel-verwijzingen geldt bij update: cascade, bij delete: restrict, behalve bij de vreemde sleutel-verwijzing van Bezettingsregel naar Stuk, daarvoor geldt bij update cascade, bij delete cascade.

Bijlage B: Tabellen met voorbeeldpopulatie

MUZIEKSCHOOL

schoolld	naam	plaatsnaam
1	Muziekschool Amsterdam	Amsterdam
2	Reijnders' Muziekschool	Nijmegen
3	Het Muziekpakhuis	Amsterdam

GENRE

genrenaam
jazz
klassiek
pop
techno

COMPONIST

componistld	naam	geboortedatum	schoolld
1	Charlie Parker	12-12-1904	NULL
2	Thom Guidi	5-1-1946	1
4	Rudolf Escher	8-1-1912	NULL
5	Sofie Bergeijk	12-7-1960	2
8	W.A. Mozart	27-01-1756	NULL
9	Karl Schumann	10-10-1935	2
10	Jan van Maanen	8-9-1965	1

NIVEAU

niveaucode	omschrijving
A	beginners
B	gevorderden
C	vergeevorderden

STUK

stuknr	componistld	titel	stuknrOrigineel	genrenaam	niveaucode	speelduur	jaartal
1	1	Blue bird	NULL	jazz	NULL	4.5	1954
2	2	Blue bird	1	jazz	B	4.0	1988
3	4	Air pur charmer	2	klassiek	B	4.5	1953
5	5	Lina	NULL	klassiek	B	5.0	1979
8	8	Berceuse	NULL	klassiek	NULL	4.0	1786
9	2	Cradle song	8	klassiek	B	3.5	1990
10	8	Non piu andrai	NULL	klassiek	NULL	NULL	1791
12	9	I'll never go	10	pop	A	6.0	1996
13	10	Swinging Lina	5	jazz	B	8.0	1997
14	5	Little Lina	5	klassiek	A	4.3	1998
15	10	Blue sky	1	jazz	A	4.0	1998

BEZETTINGSREGEL

stuknr	instrumentnaam	toonhoogte	aantal
2	drums		1
2	piano		1
2	saxofoon	alt	2
2	saxofoon	tenor	1
3	fluit		1
5	fluit		3
9	fluit		1
9	fluit	alt	1
9	piano		1
12	fluit		2
12	piano		1
13	drums		1
13	fluit		2
13	saxofoon	alt	1
13	saxofoon	tenor	1
14	fluit		1
14	piano		1
15	fluit	alt	2
15	piano		1
15	saxofoon	alt	2

INSTRUMENT

instrumentnaam	toonhoogte
drums	
fluit	
fluit	alt
gitaar	
piano	
saxofoon	alt
saxofoon	sopraan
saxofoon	tenor
viool	
viool	alt