

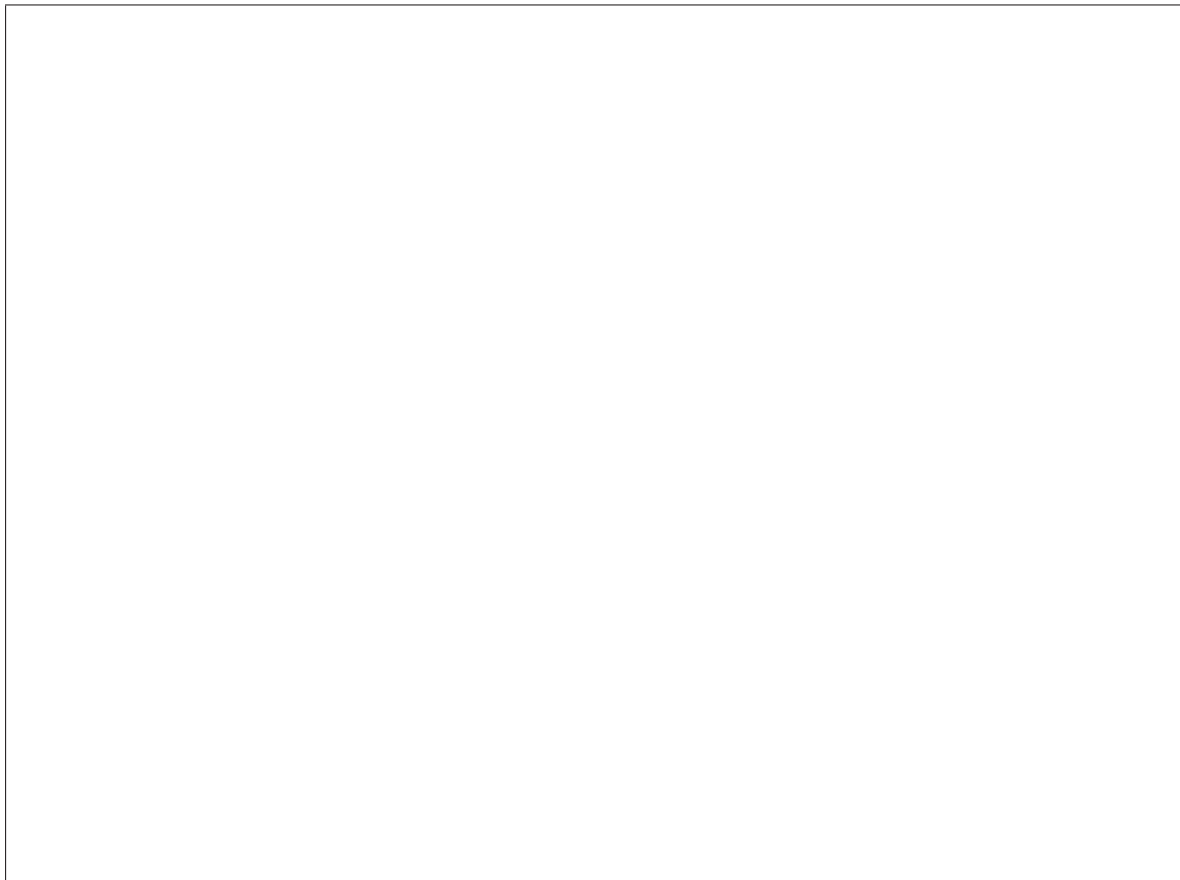
10p.

Opgave 2. De sportvereniging SSS (Sport Schaadt Studie) wil een uitgebreide website hebben voor informatievoorziening (middels documenten). Er is een rechtenstructuur die aangeeft wie wat mag doen met welk document. Hier is een specificatie van de gewenste situatie:

- Van ieder lid is de naam bekend, en de commissies waarin dat lid zitting heeft. Van ieder document is onder andere de creatiedatum en het onderwerp bekend.
- Ieder document heeft precies één lid als eigenaar.
- Er zijn diverse rechten; voorbeelden daarvan zijn: het recht om te schrijven, te lezen, te plaatsen, te verwijderen, et cetera. Niet alle rechten hoeven uitgedeeld te zijn.
- Per lid, document en recht staat vast of dat recht verleend is aan dat lid betreffende dat document, en tot wanneer.
- Per recht is er een lid dat als beheerder optreedt voor dat recht (dat lid mag die rechten uitdelen, terugnemen, et cetera).
- Documenten kunnen naar elkaar verwijzen. Kennis hierover is belangrijk omdat een verwijzing vanuit een document naar een ander mogelijk gevolgen heeft voor de rechten met betrekking tot die documenten.
- Een lid dat eigenaar is van een document, heeft zeker een recht met betrekking tot dat document.

Geef in het antwoordblok een Entity-Relationship diagram dat bovenstaande situatie *zo precies mogelijk* modelleert en gebruik daarbij *zo geschikt mogelijke* constructies. Zowel de ERD-notatie uit het boek als ook de UML notatie (class diagram) is toegestaan, maar een mengeling van beide niet.

Onleesbare tekst wordt fout gerekend.



Het zou kunnen zijn (maar het is niet de bedoeling) dat uw diagram onbedoeld of met opzet *minder* eigenschappen modelleert dan de casus aangeeft. Om dat gebrek enigszins te herstellen is er deze vraag: Geef, in precieze bewoordingen of formules, eigenschappen die *wel* uit de casustekst volgen maar *niet* in uw ER-diagram opgenomen zijn; geef er zoveel mogelijk maar hógstens twee (en dus geen enkele indien zulke eigenschappen niet bestaan):

Onleesbare tekst wordt fout gerekend.

Het zou kunnen zijn (maar het is niet de bedoeling) dat uw diagram ten onrechte of met opzet *méér* eigenschappen modelleert dan de casus aangeeft. Om zo'n "fout" enigszins te herstellen is er deze vraag: Noem en motiveer eigenschappen die *wel* in het diagram staan maar *niet* uit de letterlijke tekst van de casus volgen; geef er zoveel mogelijk maar hógstens twee (en dus geen enkele indien zulke eigenschappen niet bestaan):

Onleesbare tekst wordt fout gerekend.

Zijn er eigenschappen die wel in het ERD staan maar niet in uw databaseschema, of omgekeerd? Zo ja, geef die eigenschappen met een maximum van drie:

Onleesbare tekst wordt fout gerekend.

Aan het ERD wordt nu de volgende eigenschap (“integrity constraint”) toegevoegd:

Voor iedere $(b, d) \in S$ is er een c zodanig dat $(b, c) \in R$ en $(c, d) \in T$.

Geef aan hoe u deze eigenschap in het database schema verwerkt op zodanige manier dat in elke databasevulling die eigenschap zal gelden:

Onleesbare tekst wordt fout gerekend.

10p.



Opgave 4. Beschouw een muzikliefhebber die allerlei gegevens bijhoudt, zoals: liederen, schrijvers, muziekwinkels, opslagmedia, muziekcategorieen, etc. Deze gegevens tezamen vormen een relatie \mathcal{R} die de entiteiten op de volgende manier aan elkaar relateert. Een tuple (L, S, W, P, C, U, M) zit op zeker tijdstip in \mathcal{R} precies wanneer op dat tijdstip het volgende geldt:

1. L is een *Lied*.
2. S is de *Schrijver* van lied L .
3. W is een *Winkel* waar L verkocht wordt.
4. P is de *Prijs* die winkel W vraagt voor lied L op medium M .
5. C is een *Categorie* waarin lied L is ingedeeld.
6. U is de *Uitgever* van lied L .
7. M is een opslagMedium waarop lied L te koop is in winkel W .

Voorts gelden er de volgende eigenschappen op ieder tijdstip:

- a. Een lied heeft precies één schrijver.
- b. Een schrijver werkt voor hooguit één uitgever (dat wil zeggen, van die schrijver worden alle liederen door hooguit één uitgever uitgegeven).
- c. Een uitgever is gespecialiseerd in precies één categorie (dat wil zeggen, die uitgever geeft geen liederen uit die niet tot zijn specialisatiecategorie behoren).

Geef voor ieder van de functionele afhankelijkheden hieronder, met een letter W of O aan of die altijd *Waar* of *Onwaar* is in relatie \mathcal{R} , en motiveer kort uw keuze (de motivatie telt mee in de beoordeling):

Onleesbare tekst wordt fout gerekend.

FD	W/O	Motivatie voor uw keuze
$W, S \rightarrow U$		
$C \rightarrow U$		
$P, M, W \rightarrow L$		
$P, M, L \rightarrow W$		
$L, W \rightarrow M$		
$S \rightarrow C$		
$L \rightarrow C$		

Geef, zonder motivatie, zoveel mogelijk maar hooguit twee, altijd ware multivalued dependencies in \mathcal{R} die niet triviaal of onvermijdelijk zijn en zelf geen functionele afhankelijkheid zijn:

Onleesbare tekst wordt fout gerekend.

10p.

Opgave 5. Beschouw het relatieschema $\mathbf{R} = (\bar{R}, \mathcal{F})$, waarbij de attribootverzameling \bar{R} en de verzameling \mathcal{F} van functionele afhankelijkheden als volgt luiden:

$$\bar{R} = ABCDEF$$

$$\mathcal{F} = \{AB \rightarrow D, \quad BCD \rightarrow F, \quad EF \rightarrow C, \quad E \rightarrow B\}$$

- (1) Geef in het antwoordblok in iedere regel een zo groot mogelijk rechterlid Y zó dat de functionele afhankelijkheid $X \rightarrow Y$ volgt uit de hierboven gegeven verzameling \mathcal{F} (met andere woorden: Y is de closure $X_{\mathcal{F}}^+$). U hoeft de leden van X niet op te nemen in Y .
- (2) *Omcirkel* in het antwoordblok iedere *sleutel* van \mathbf{R} .
- (3) *Onderstreep* in het antwoordblok iedere *supersleutel* van \mathbf{R} .
- (4) Omcirkel in het antwoordblok de *nummers* van de functionele afhankelijkheden die een schending vormen van de BCNF-eis.

Onleesbare tekst wordt fout gerekend.

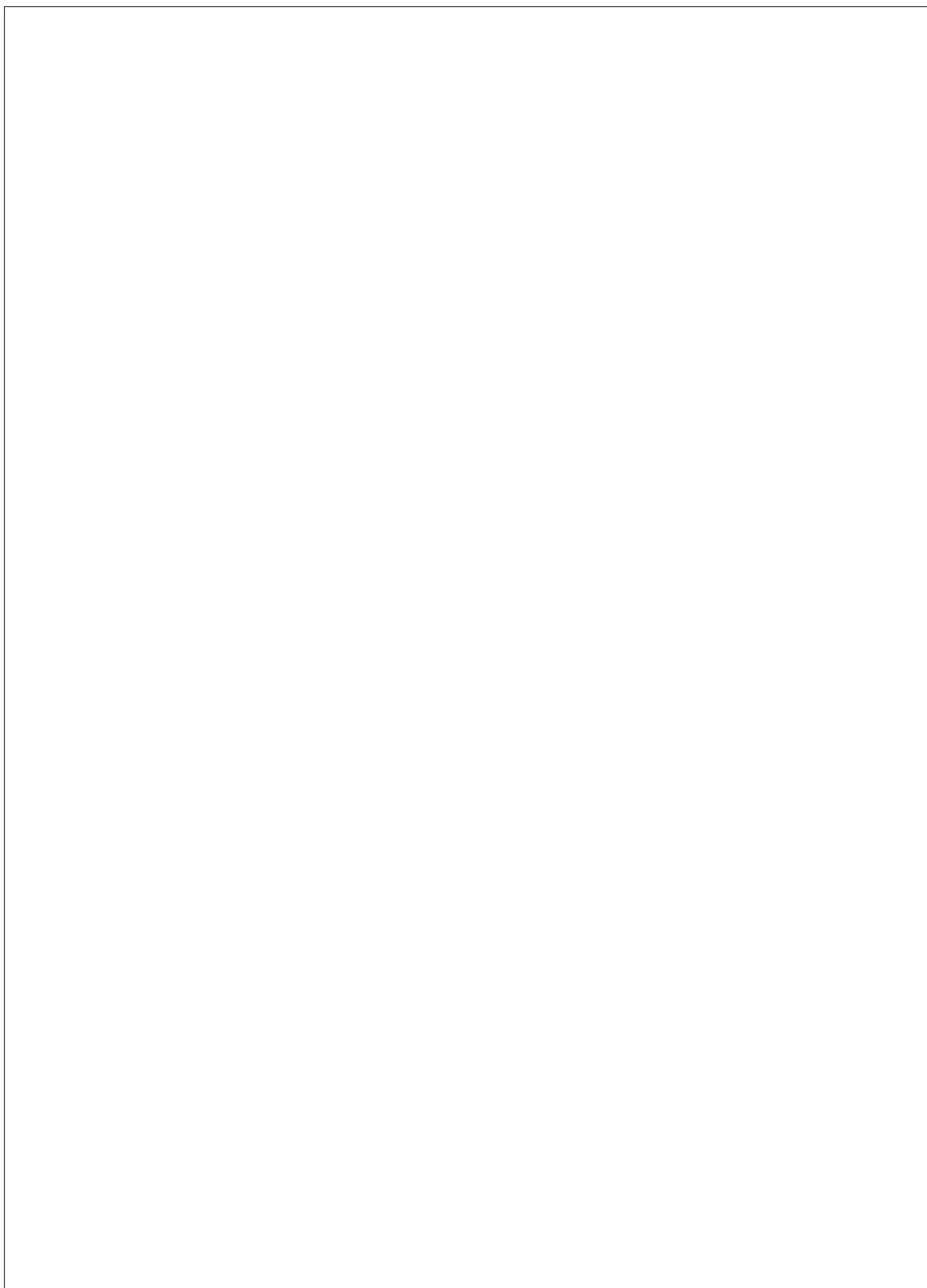
	<u>$X \rightarrow Y$</u>
10	$AD \rightarrow$
11	$AE \rightarrow$
12	$AF \rightarrow$
13	$BC \rightarrow$
14	$BD \rightarrow$
15	$BE \rightarrow$
16	$BF \rightarrow$
17	$CD \rightarrow$
18	$CE \rightarrow$
19	$CF \rightarrow$
20	$DE \rightarrow$
21	$DF \rightarrow$
22	$EF \rightarrow$
23	$ABC \rightarrow$
24	$ABD \rightarrow$
25	$ABE \rightarrow$
26	$ABF \rightarrow$
27	$ACD \rightarrow$
28	$ACE \rightarrow$
29	$ACF \rightarrow$
30	$ADE \rightarrow$
31	$ADF \rightarrow$
32	$AEF \rightarrow$

10p. **Opgave 6.** Beschouw het relatieschema $\mathbf{R} = (ABCDEFGH, \mathcal{F})$, waarbij:

$$\mathcal{F} = \{AB \rightarrow CD, \quad BC \rightarrow E, \quad GH \rightarrow F, \quad F \rightarrow E\}$$

Construeer in het antwoordblok een lossless decompositie van \mathbf{R} tot schema's die ieder in BCNF staan. *Geef ook aan of de functionele afhankelijkheden behouden blijven onder de decompositie.* Geef bij iedere stap een verklaring zodat het voor de corrector heel duidelijk is hoe u te werk gaat.

Onleesbare tekst wordt fout gerekend.



In de volgende opgavenserie wordt het volgende databaseschema gebruikt:

Class (*name*, *type*, *country*, *guns*, *bore*, *displacement*)

Ship (*name*, *classname*, *launched*)

Battle (*name*, *date*)

Outcome (*shipname*, *battlename*, *result*)

De attributen die tot de sleutel behoren zijn onderstreept.

In *Ship* is *classname* een foreign key verwijzend naar *Class* (*name*).

In *Outcome* is *shipname* een foreign key verwijzend naar *Ship* (*name*).

In *Outcome* is *battlename* een foreign key verwijzend naar *Battle* (*name*).

Schepen die volgens eenzelfde ontwerp worden gebouwd vormen samen een klasse (*class*). Klassen komen in twee typen (*type*): *bb* (voor *battleship*) en *bc* (voor *battlecruiser*). De overige attributen van een klasse zijn: het land (*country*), het aantal kanonnen (*guns*), de diameter in centimeters van de kanonsloop (*bore*), en de waterverplaatsing (*displacement*, gemeten in tonnen). Van een schip is, naast de naam (*name*) en de klassenaam (*classname*), ook nog bekend wanneer het te water is gelaten (*launched*). Van een zeeslag (*battle*) is de naam (*name*) en datum (*date*) bekend. Relatie *Outcome* geeft aan hoe schepen de zeeslagen hebben doorstaan: gezonken, beschadigd of okay (*result* = *sunk*, *damaged*, en *ok*, respectievelijk).

Wanneer we spreken van het *type* van een schip, dan bedoelen we het *type* van de klasse van dat schip; net zo voor de attributen *country*, *guns*, *bore*, *displacement*. Dus alle schepen van een klasse komen uit één land: het land dat in de klasse genoemd staat.

U mag identifiers tot hun eerste letter afkorten. Het schema luidt dan:

C (*n*, *t*, *c*, *g*, *b*, *d*)

S (*n*, *c*, *l*)

B (*n*, *d*)

O (*s*, *b*, *r*)

Geef ook een formulering van de vraag in Relationele Algebra op zodanige manier dat het Cartesisch product \times daarin niet voorkomt:

Onleesbare tekst wordt fout gerekend.

<i>Class</i> (<u><i>name</i></u> , <i>type</i> , <i>country</i> , <i>guns</i> , <i>bore</i> , <i>displacement</i>)	<i>C</i> (<u><i>n</i></u> , <i>t</i> , <i>c</i> , <i>g</i> , <i>b</i> , <i>d</i>)
<i>Ship</i> (<u><i>name</i></u> , <i>classname</i> , <i>launched</i>)	<i>S</i> (<u><i>n</i></u> , <i>c</i> , <i>l</i>)
<i>Battle</i> (<u><i>name</i></u> , <i>date</i>)	<i>B</i> (<u><i>n</i></u> , <i>d</i>)
<i>Outcome</i> (<u><i>shipname</i></u> , <u><i>battlename</i></u> , <i>result</i>)	<i>O</i> (<u><i>s</i></u> , <u><i>b</i></u> , <i>r</i>)

10p. **Opgave 8.** Formuleer in SQL met een group-by query:

Geef van iedere klasse waarvan alle schepen tussen 1700 en 1800 te water zijn gelaten, het aantal schepen dat in een of andere zeeslag gezonken is, mits dat aantal minstens drie is.

Veronderstel dat datums (zoals het *launched* attribuut van een schip) vergeleken kunnen worden met $<$ ('kleiner dan'), etc, en dus ook *min* en *max* daarop van toepassing zijn.

Onleesbare tekst wordt fout gerekend.

10p. **Opgave 10.** Beschouw het volgende databaseschema:



Person (*id*, *name*)
Student(*id*, *university*)

De bedoeling is dat iedere student ook een persoon is. Daarom willen we dat voortdurend de volgende acties worden uitgevoerd:

1. Wanneer een rij wordt verwijderd uit tabel *Person*, dan wordt de rij met hetzelfde *id* ook verwijderd uit tabel *Student*.
2. Wanneer er gepoogd wordt een rij toe te voegen aan tabel *Student*, dan wordt getest of er een rij met hetzelfde *id* bestaat in *Person*; is dat niet het geval, dan vindt de toevoeging niet plaats.

Voeg key constraints toe aan het schema waardoor de gewenste acties automatisch door het databasesysteem worden uitgevoerd. Ter herinnering, key constraints zien er als volgt uit:

- primary key (x, x', \dots)
- foreign key (x, x', \dots) references $T(y, y', \dots)$ on *event* action on *event'* action' ...

Hierbij staat x voor een attribuutnaam, T voor een tabelnaam, *event* voor een gebeurtenis (*delete*, *update*) en *action* voor een actie (*set null*, *set default*, *no action*, *cascade*).

Onleesbare tekst wordt fout gerekend.

10p.

Opgave 11. Zet in onderstaande tabel een merkteken ‘×’ bij iedere combinatie van isolatieniveaus voor transacties T_1 en T_2 waarbij T_2 geen phantom kan ervaren.

Onleesbare tekst wordt fout gerekend.

Op de met ‘×’ gemerkte niveaus kan T_2 geen phantom ervaren:

Isolatieniveau voor T_1 : ↓	Isolatieniveau voor T_2 :			
	read uncommitted	read committed	repeatable read	serializable
read uncommitted				
read committed				
repeatable read				
serializable				

Leg uit wat er wordt verstaan onder ‘transactie T ervaart geen phantoms’:

Onleesbare tekst wordt fout gerekend.

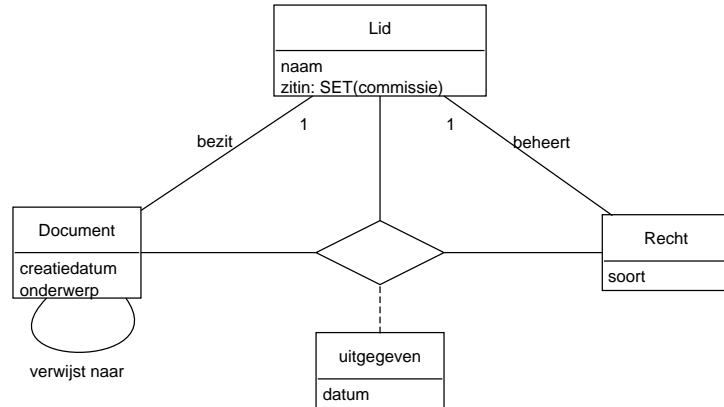
Antwoorden

Opgave 1

Opgave 2

Antwoord:

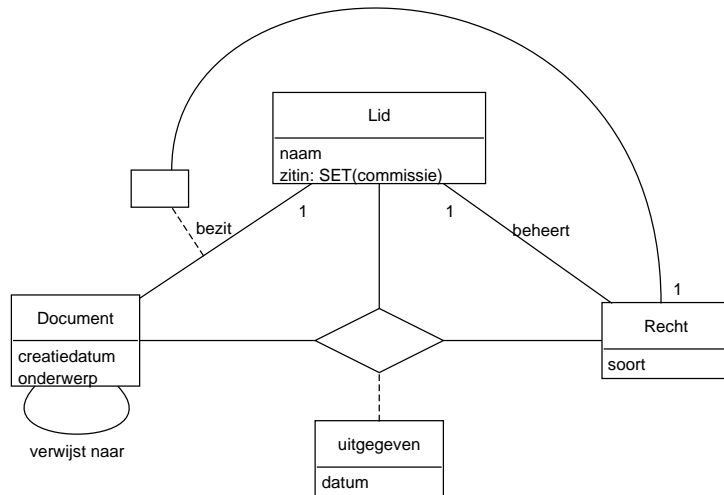
(Niet-vermelde multipliciteiten leggen geen beperkingen op: 0..*.)



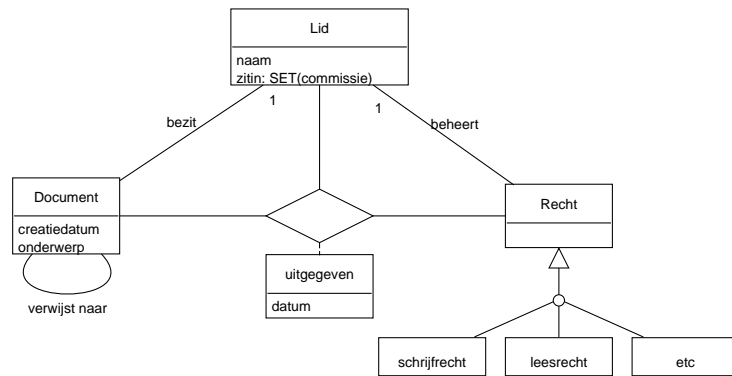
Toelichting.

Er is géén afzonderlijke entiteit *Commissie* (met een associatie naar *Lid*) omdat er volgens de casus van commissies geen informatie wordt bijgehouden.

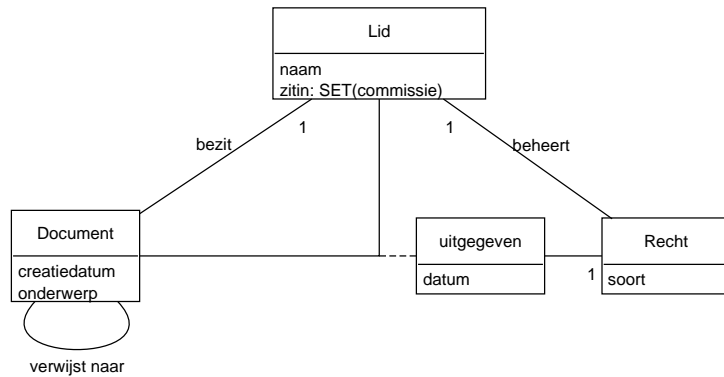
Een variant waarin ook het laatste punt (bullet) uit de casus wordt vastgelegd:



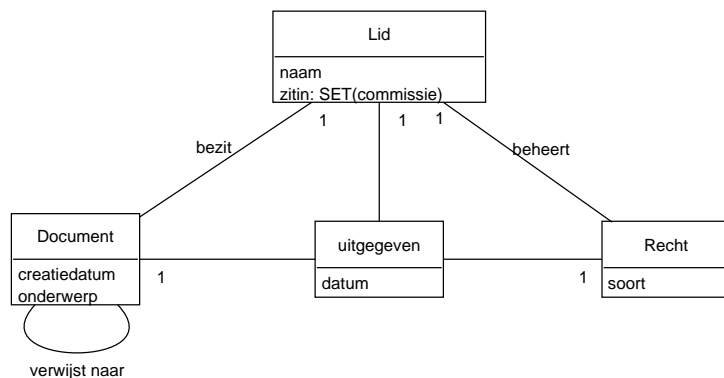
Een variant waarin de soorten rechten als afzonderlijke entiteiten worden geformaliseerd:



Een foute formalisering. In het volgende ERD is het niet zo dat “Per (*drietal*) lid, document en recht staat vast of dat recht verleend is aan dat lid betreffende dat document, en tot wanneer” maar in plaats daarvan wel dat “Per (*tweetal*) lid en document staat vast of aan dat lid betreffende dat document een recht verleend is en tot wanneer”:



Nog een fout. Volgens het volgende diagram kan er meermalen “per lid, document en recht” een datum vastgesteld zijn:



Antwoord:

In het eerste ERD is de eigenschap van bullet 7 niet geformaliseerd (Een lid dat eigenaar is van een document, heeft zeker een recht met betrekking tot dat document).

Antwoord:

Geen

Opgave 3

Antwoord:

$A(a1, a2, \text{primary key } (a1))$
 $B(a1, b1, b2, \text{primary key } (a1), \text{foreign key } (a1) \text{ references } A(a1))$
 $C(c1, c2, a1, \text{primary key } c1, \text{foreign key } (a1) \text{ references } B(a1))$
 $D(d1, d2, \text{primary key } d1)$
 $S(a1, d1, \text{primary key } (a1, d1),$
 foreign key $(a1)$ references $B(a1)$,
 foreign key $(d1)$ references $D(d1)$)
 $T(c1, d1, \text{primary key } (c1, d1),$
 foreign key $(c1)$ references $C(c1)$,
 foreign key $(d1)$ references $D(d1)$)

Toelichting.

Entiteiten A en B zijn niet samengenomen in één tabel $A(a1, a2, b1, b2)$ omdat dan NULLs nodig zijn om een A -instantie te representeren die geen B -instantie is.

Entiteit R is niet als aparte tabel nodig, omdat R -instanties gerepresenteerd kunnen worden in een attribuut (namelijk het $a1$ -attribuut) van C (vanwege de eigenschap dat er bij iedere C -instantie precies één B -instantie is waarmee die in relatie R zit).

Antwoord:

Nee

Antwoord:

Neem de volgende check in het schema van S op:

exists (select * from C c , T t where $a1 = c.a1$ and $c.c1 = t.c1$ and $t.d1 = d1$)

Of –minder efficiënt– voeg een assertion toe aan het database schema met de volgende check:
not exists (

 select * from S s where not exists (
 select * from C c , T t where $a1 = c.a1$ and $c.c1 = t.c1$ and $t.d1 = d1$)

Of –met enige redundantie– voeg een attribuut $c1$ toe aan S en ook de volgende checks:

check (($a1, c1$) in (select $a1, c1$ from C))

check (($c1, d1$) in (select $c1, d1$ from T))

De laatste check hierboven kan als een foreign key constraint geformuleerd worden:

foreign key $(c1, d1)$ references $T(c1, d1)$

Opgave 4

Antwoord:

FD	W/O	Motivatie voor uw keuze
$W, S \rightarrow U$	W	(b)
$C \rightarrow U$	O	twee uitgevers kunnen in dezelfde categorie gespecialiseerd zijn
$P, M, W \rightarrow L$	O	twee liederen kunnen te koop zijn bij dezelfde P , M en W
$P, M, L \rightarrow W$	O	twee winkels kunnen hetzelfde lied verkopen met dezelfde P en M
$L, W \rightarrow M$	O	een L kan in W op verscheidene Media te koop zijn
$S \rightarrow C$	W	wegens $S \rightarrow U$ (b) en $U \rightarrow C$ (c)
$L \rightarrow C$	W	wegens $L \rightarrow S$ (2,a) en $S \rightarrow U$ (b) en $U \rightarrow C$ (c) (“wegens 5” is fout!)

Antwoord:

$LSWPCUM = LSCU \bowtie LWPM$ (L is key in lhs) (maar dit is equivalent met $L \rightarrow SCU$).

$LSWPCUM = SCU \bowtie LSWPM$ (S is key in lhs) (maar dit is equivalent met $S \rightarrow CU$).

Antwoord:

	X	\rightarrow	Y
10	AD	\rightarrow	
11	AE	\rightarrow	BD
12	AF	\rightarrow	
13	BC	\rightarrow	
14	BD	\rightarrow	
15	BE	\rightarrow	
16	BF	\rightarrow	
17	CD	\rightarrow	
18	CE	\rightarrow	B
19	CF	\rightarrow	
20	DE	\rightarrow	B
21	DF	\rightarrow	
22	EF	\rightarrow	BC
23	ABC	\rightarrow	DF
24	ABD	\rightarrow	
25	ABE	\rightarrow	D
26	ABF	\rightarrow	D
27	ACD	\rightarrow	
28	ACE	\rightarrow	BDF
29	ACF	\rightarrow	
30	ADE	\rightarrow	B
31	ADF	\rightarrow	
32	AEF	\rightarrow	BCD

Toelichting.

Let op: iedere key is ook een superkey, omgekeerd is dit niet het geval.

Antwoord:

Let op: $AB \rightarrow E$ volgt uit \mathcal{F} , en zal dus (bij een correcte redenering) een FD worden van een component met attributen $ABE\dots$ (ook al zit C daar niet bij). Net zo: $GH \rightarrow E$ volgt uit \mathcal{F} en zal dus een FD worden van een component met attributen $GHE\dots$ (ook al zit F daar niet bij).

We passen het BCNF-algoritme toe.

- We bekijken $\mathbf{R} = (ABCDEFGH, \mathcal{F})$.

Van de gegeven \mathcal{F} zijn alle leden een schending van de BCNF-eis voor \mathbf{R} ; bijvoorbeeld voor $AB \rightarrow CD$: de FD is niet-triviaal en het linkerlid is geen sleutel: $AB_{\mathcal{F}}^+ = ABCDE \neq ABCDEFGH$. We kiezen (zomaar) de eerste, $AB \rightarrow CD$, ter eliminatie. Dus splitsen we \bar{R} in $\bar{R}_1 = ABCD$ en $\bar{R}_2 = AB\cancel{C}\cancel{D}EFGH = ABEFGH$. Dit levert schema's $\mathbf{R}_i = (\bar{R}_i, \mathcal{F}_i)$, waarbij \mathcal{F}_i (een basis voor) de inperking is van \mathcal{F}^+ tot \bar{R}_i . Dus, $\mathbf{R}_1 = (ABCD, \{AB \rightarrow CD\})$ en $\mathbf{R}_2 = (ABEFGH, \{AB \rightarrow E, GH \rightarrow F, F \rightarrow E\})$. Let op: de $AB \rightarrow E$ zit weliswaar niet in \mathcal{F} , maar toch in \mathcal{F}_2 zoals aan het begin is uitgelegd. De afhankelijkheid $BC \rightarrow E$ is verloren gegaan.

- We bekijken nu $\mathbf{R}_1 = (ABCD, \{AB \rightarrow CD\})$.

In \mathbf{R}_1 is $AB \rightarrow CD$ geen schending van de BCNF-conditie, en dus staat \mathbf{R}_1 in BCNF.

- We bekijken nu $\mathbf{R}_2 = (ABEFGH, \{AB \rightarrow E, GH \rightarrow F, F \rightarrow E\})$.

In \mathbf{R}_2 zijn $AB \rightarrow E$ en $GH \rightarrow F$ en $F \rightarrow E$ alledrie een schending van de BCNF-conditie; voor $GH \rightarrow F$ is de reden dat die niet-triviaal is en GH geen sleutel is in \mathbf{R}_2 (want $GH_{\mathcal{F}_2}^+ = EFGH \neq ABEFGH$). We kiezen (zomaar) $GH \rightarrow F$ ter eliminatie. Dus splitsen we \bar{R}_2 in $\bar{R}_{2a} = GHF$ en $\bar{R}_{2b} = ABE\cancel{F}GH = ABEGH$. Dit levert schema's $\mathbf{R}_{2j} = (\bar{R}_{2j}, \mathcal{F}_{2j})$, waarbij \mathcal{F}_{2j} (een basis voor) de inperking is van \mathcal{F}^+ (of \mathcal{F}_2^+) tot \bar{R}_{2j} . Dus $\mathbf{R}_{2a} = (GHF, \{GH \rightarrow F\})$ en $\mathbf{R}_{2b} = (ABEGH, \{AB \rightarrow E, GH \rightarrow E\})$. Let op: de $GH \rightarrow E$ zit weliswaar niet in \mathcal{F}_2 , maar toch in \mathcal{F}_{2b} zoals aan het begin is uitgelegd.

- We bekijken nu $\mathbf{R}_{2a} = (FGH, \{GH \rightarrow F\})$. Deze staat in BCNF.

- We bekijken nu $\mathbf{R}_{2b} = (ABEGH, \{AB \rightarrow E, GH \rightarrow E\})$. Zowel $AB \rightarrow E$ als ook $GH \rightarrow E$ vormen een schending van de BCNF-conditie in \mathbf{R}_{2b} ; bijvoorbeeld voor $AB \rightarrow E$: die FD is niet triviaal en het linkerlid is geen sleutel (want $AB_{\mathcal{F}_{2b}}^+ = ABE \neq ABEGH$). We kiezen (zomaar) $AB \rightarrow E$ ter eliminatie. Dus splitsen we \bar{R}_{2b} in $\bar{R}_{2b1} = ABE$ en $\bar{R}_{2b2} = AB\cancel{E}GH = ABGH$. Dit levert schema's $\mathbf{R}_{2bx} = (\bar{R}_{2bx}, \mathcal{F}_{2bx})$, waarbij \mathcal{F}_{2bx} (een basis voor) de inperking is van \mathcal{F}^+ (of \mathcal{F}_{2b}^+) tot \bar{R}_{2bx} . Dus $\mathbf{R}_{2b1} = (ABE, \{AB \rightarrow E\})$ en $\mathbf{R}_{2b2} = (ABGH, \{\})$. Beide staan in BCNF.

- Dus $\{\mathbf{R}_1, \mathbf{R}_{2a}, \mathbf{R}_{2b1}, \mathbf{R}_{2b2}\}$ is een decompositie van \mathbf{R} waarvan alle componenten in BCNF staan. De functionele afhankelijkheden zijn niet allemaal behouden; met name is $BC \rightarrow E$ verloren gegaan.

- NB. Omdat dit een lossless decompositie is (een eigenschap van het toegepaste BCNF-algoritme), schrijven we ook wel:

$$"ABCDEF GH = ABCD \bowtie FGH \bowtie ABE \bowtie ABGH \text{ geldt in } \mathbf{R}."$$

Wanneer je met een andere schending begint kun je mogelijk een andere decompositie bereiken.

Opgave 7

Antwoord:

$$\begin{aligned}
& \text{"(geef) ieder land waarvan een schip voor 1600 te water ging"} \\
&= \{c\text{entry:txt} \mid \text{"c\text{entry} is een land"} \wedge \text{"een schip uit c\text{entry} voor 1600 te water"} \bullet c\text{entry}\} \\
&= \{c\text{entry:txt} \mid (\exists c:C \bullet c.c=c\text{entry}) \wedge (\exists s:S \mid \text{"s is uit c\text{entry}}" \bullet \text{"s voor 1600 te water"}) \bullet c\text{entry}\} \\
&= \{c\text{entry:txt}; c:C \mid c.c=c\text{entry} \quad \wedge (\exists s:S \mid \text{"s is uit c\text{entry}}" \bullet \text{"s voor 1600 te water"}) \bullet c\text{entry}\} \\
&= \{c\text{entry:txt}; c:C \mid c.c=c\text{entry} \quad \wedge (\exists s:S \mid \text{"s is uit c.c"} \bullet \text{"s voor 1600 te water"}) \bullet c.c\} \\
&= \{c:C \mid (\exists c\text{entry:txt} \bullet c.c=c\text{entry}) \wedge (\exists s:S \mid \text{"s is uit c.c"} \bullet \text{"s voor 1600 te water"}) \bullet c.c\} \\
&= \{c:C \mid c.c \in \text{txt} \quad \wedge (\exists s:S \mid \text{"s is uit c.c"} \bullet \text{"s voor 1600 te water"}) \bullet c.c\} \\
&= \{c:C \mid (\exists s:S \mid \text{"s is uit c.c"} \bullet \text{"s voor 1600 te water"}) \bullet c.c\} \\
&= \{c:C; s:S \mid \text{"s is uit c.c"} \wedge \text{"s voor 1600 te water"} \bullet c.c\} \\
&= \{c:C; s:S \mid (\exists c':C \mid s.c=c'.n \bullet c'.c=c.c) \wedge s.l < 1600 \bullet c.c\} \\
&= \{c, c' : C; s : S \mid s.c=c'.n \wedge c'.c=c.c \wedge s.l < 1600 \bullet c.c\} \\
&= \{c' : C; s : S \mid s.c=c'.n \wedge (\exists c : C \bullet c'.c=c.c) \wedge s.l < 1600 \bullet c'.c\} \\
&= \{c' : C; s : S \mid s.c=c'.n \wedge s.l < 1600 \bullet c'.c\}
\end{aligned}$$

Antwoord:

select distinct c1.c from C c1, S s where s.c = c1.n and s.l < 1600

Antwoord:

$\pi_{country} (\sigma_{launched < 1600} (C \bowtie_{name=classname} S))$
 $= \pi_c (\sigma_{sl < 1600} (C' \bowtie S'))$

Hierbij zijn met renaming nieuwe relaties gedefinieerd:

$S' = S[\dots, cn, sl]$ en $C' = C[cn, c, \dots]$.

Opgave 8

Antwoord:

```
select s.classname, count (distinct s.name)
from ship s, outcome o
where s.name = o.shipname and o.result = sunk and s.classname not in
      (select s.classname from ship s where not (s.launched between 1700 and 1800))
group by s.classname
having count (distinct s.name) > 2
```

(Wanneer een schip maar éénmaal kan zinken, is ‘distinct’ overbodig, en kan zelfs ‘count (distinct s.name)’ vervangen worden door ‘count (*)’.)

Toelichting.

Een alternatief met meer group by clauses:

```
select s.classname, count (distinct s.name)
from ship s, outcome o
where s.name = o.shipname and o.result = sunk and s.classname in
      (select s'.classname from ship s'
       group by s'.classname
       having 1700 < min (s'.launched) and max (s'.launched) < 1800)
group by s.classname
having count (distinct s.name) > 2
```

Opgave 9

Antwoord:

Korthedshalve laten we de typering achterwege (en korten dus $b : B$ af tot b , et cetera).

$$\{b \mid (\forall c \mid \text{“}c \text{ neemt deel aan } b\text{”} \bullet \text{“er is een andere klasse met gelijk kanonmentaal”}) \bullet b.n\}$$

$$= \{b \mid (\forall c \mid (\exists s \mid \text{“}s \text{ behoort tot } c\text{”} \bullet \text{“}s \text{ neemt deel aan } b\text{”}) \bullet (\exists c' \mid c' \neq c \bullet c'.g = c.g)) \bullet b.n\}$$

$$= \{b \mid (\forall c \mid (\exists s \mid s.c = c.n \bullet (\exists o \bullet s.n = o.s \wedge o.b = b.n)) \bullet (\exists c' \mid c' \neq c \bullet c'.g = c.g)) \bullet b.n\}$$

$$= \{b \mid (\forall c \mid (\exists s; o \bullet s.c = c.n \wedge s.n = o.s \wedge o.b = b.n) \bullet (\exists c' \mid c' \neq c \bullet c'.g = c.g)) \bullet b.n\}$$

$$= \{b \mid (\forall c; s; o \mid s.c = c.n \wedge s.n = o.s \wedge o.b = b.n \bullet (\exists c' \mid c' \neq c \bullet c'.g = c.g)) \bullet b.n\}$$

$$= \{b \mid \neg (\exists c; s; o \mid s.c = c.n \wedge s.n = o.s \wedge o.b = b.n \bullet \neg (\exists c' \mid c' \neq c \bullet c'.g = c.g)) \bullet b.n\}$$

$= \text{select } b.n \text{ from } B \text{ } b \text{ where not exists (}$
 $\text{select } * \text{ from } C \text{ } c, S \text{ } s, O \text{ } o \text{ where}$
 $s.c = c.n \text{ and } s.n = o.s \text{ and } o.b = b.n \text{ and}$
 $\text{not exists (select } * \text{ from } C \text{ } c' \text{ where } c'.n \neq c.n \text{ and } c'.g = c.g))$

Opgave 10

Antwoord:

Person (*id*, *name*, primary key *id*)

Student (*id*, *name*, foreign key (*id*) references *Person*(*id*) on delete cascade)

Het deel ‘on delete cascade’ zorgt voor actie 1,

het deel ‘foreign key(*id*) references *Person(id)*’ zorgt voor actie 2.

Er werd niet gevraagd om in *Student* de *id* tot primary key te verklaren, en ook niet om de actie die opgeroepen wordt door ‘on update cascade’ toe te voegen aan de foreign key constraint.

Antwoord:

```
create trigger OnPersonDelete
  after delete on Person
  referencing old as DeletedPerson
  for each row
  delete from Student s where s.id = DeletedPerson.id;
create trigger OnStudentInsert
  before insert on Student
  referencing new as NewStudent
  for each row
  when (NewStudent.id not in (select id from Person))
  abort;
```

Varianten met statement level granularity (‘for each statement’, de default) zijn ook mogelijk.
Toelichting.

Varianten met statement level granularity (de default):

```
create trigger OnPersonDelete'
  after delete on Person
  referencing old table as DeletedPersons
  for each statement
  delete from Student s where s.id in (select id from DeletedPersons);
create trigger OnStudentInsert'
  before insert on Student
  referencing new table as NewStudents
  for each statement
  when (exists ((select id from NewStudents) except (select id from Person)))
  abort;
```

Opgave 11

Antwoord:

Op de met ‘×’ gemerkte niveaus kan T_2 geen phantom ervaren:

Isolatie-niveau voor T_1 : ↓	Isolatie-niveau voor T_2 :			
	read uncommitted	read committed	repeatable read	serializable
read uncommitted				×
read committed				×
repeatable read				×
serializable				×

Het niveau voor T_1 doet niet terzake!

Antwoord:

‘Transactie T ervaart geen phantoms’ precies wanneer het volgende *niet* gebeurt:

T selecteert rijen van een tabel met een selectie criterium P en gedurende de executie van T wordt die tabel door een andere transactie uitgebreid met een rij die aan selectie criterium P voldoet — waarna T opnieuw rijen van die tabel volgens hetzelfde criterium P selecteert (en dus “een nieuwe rij”, de phantom, krijgt opgeleverd).