

- Het examen is volledig open boek. Je mag al het materiaal gebruiken dat voorzien is tijdens de lessen (bv. het boek ‘Principes van databases’ van prof. dr. Guy De Tré, slides, opgaves, (opgeloste) oefeningen. . .) of dat je zelf hebt gebruikt ter verwerking van de leerstof (bv. samenvattingen, notities. . .).
- Alle communicatie met personen (behalve de examenbegeleiders) is strikt verboden en zal bestraft worden met uitsluiting van het examen.
- Het examen dient opgelost te worden op witte bladeren. Gebruik voor elk deel van het examen (sectie 1-4) een apart blad. Vermeld zeker op elk oplossingsblad je studentnummer, naam en richting.
- Indien er vragen of onduidelijkheden zijn, schrijf deze dan nauwgezet op je oplossingsblad, zodat wij hier tijdens het quoteren rekening mee kunnen houden.

1 Open vragen

Beantwoord onderstaande vragen.

1. (Relationeel databankmodel) Waarom gebeurt bij het relationeel databankmodel de bevraging doorgaans niet op de externe laag maar op de logisch laag van het databanksysteem? Leg duidelijk uit.
2. (Beveiliging) Geef een duidelijk voorbeeld van een situatie waarbij poly-instantiatie noodzakelijk is. Welk probleem wordt hierdoor voorkomen?
3. (Herstel na falen) Beschouw de concurrente uitvoering van transacties T_1 , T_2 , T_3 , T_4 en T_5 zoals weergegeven in Tabel 1. Aanpassingen van a , b en c gebeuren in lokale databankbuffers en zijn pas zichtbaar in de databank na flushing van deze databankbuffers. Ga uit van een ‘steal, no force’ flushingstrategie. De gebruikte hersteltechniek is met onmiddellijke aanpassing.
 - a) Veronderstel dat er zich een soft crash voordoet onmiddellijk na tijdstip 23 en dat alle logrecords tot en met dit tijdstip correct naar het logbestand zijn geschreven. Welke transacties moeten dan ongedaan gemaakt worden en welke transacties moeten opnieuw uitgevoerd worden bij het herstel van de inhoud van de databankbuffers?

↓Tijd	T_1	T_2	T_3	T_4	T_5
1		start			
2		lees a			
3		pas a aan			
4		commit			
5	start				
6			start		
7			lees b		
8	-----controlepunt-----				
9			pas b aan		
10			commit		
11	lees a				
12				start	
13					start
14					lees b
15				lees c	
16	pas a aan			pas c aan	
17				rollback	
18					
19	commit				
20					pas b aan
21					lees a
22					pas a aan
23					commit

Tabel 1: Concurrente uitvoering van transacties T_1 , T_2 , T_3 , T_4 en T_5 .

- b) Veronderstel nu dat de soft crash zich voordoet onmiddellijk na tijdstip 18, maar voor tijdstip 19 en dat alle logrecords tot en met dit tijdstip correct naar het logbestand zijn geschreven. Welke transacties moeten nu ongedaan gemaakt worden en welke transacties moeten nu opnieuw uitgevoerd worden bij het herstel van de inhoud van de databankbuffers?
4. (NoSQL) Welk probleem krijg je als je sharding (horizontale schaling) gaat toepassen op (object)relationele databanken? Motiveer je antwoord.

2 Conceptueel ontwerp

Een traiteur wil voor de feestdagen een online systeem opzetten voor het bestellen van feestmaaltijden voor Kerst en oudejaar. Om de data in verband met zijn aanbod en (het afhalen van) alle bestellingen efficiënt te kunnen bijhouden, zal hij gebruik

maken van een databank. Stel een conceptueel ontwerp op voor deze databank, uitgaande van onderstaande vereisten.

Eerst en vooral dient het aanbod dat de traiteur voorziet te worden opgeslagen in de databank. Dit aanbod bestaat uit individuele producten, elk met een overeenkomstige naam. Verder heeft elk product een korte beschrijving, een prijs en behoort het product tot een specifieke categorie (voorgerecht, hoofdgerecht, dessert of drank). Een product wordt uniek geïdentificeerd aan de hand van de combinatie van naam en productcategorie. Tot slot dient voor elk product te worden bijgehouden wat de bijhorende allergenen (bv. noten, schaaldieren...) zijn.

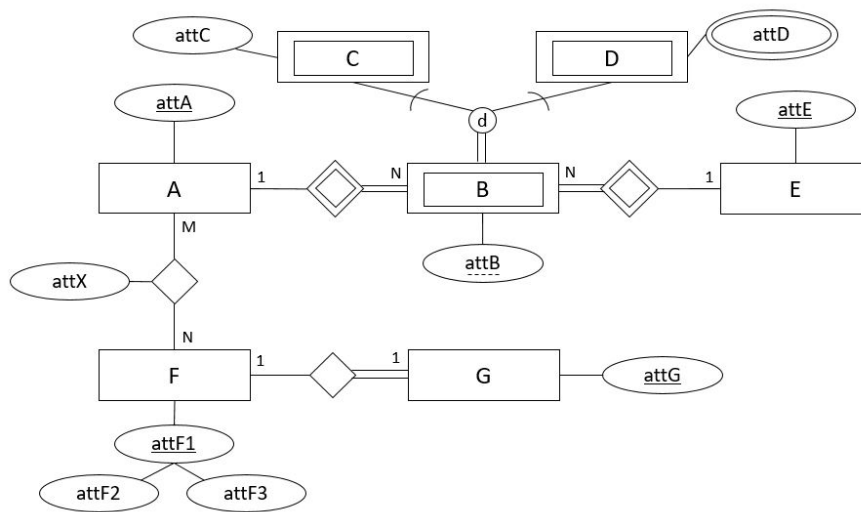
Klanten kunnen een bestelling plaatsen nadat ze een gebruikersprofiel aangemaakt hebben. Ieder profiel wordt uniek geïdentificeerd door een e-mailadres. Daarnaast wordt het bijhorende wachtwoord van de gebruiker (beveiligd) opgeslagen. Naast de producten die de klant heeft besteld, dienen ook het tijdstip waarop de bestelling is geplaatst en (optioneel) bijkomende commentaar van de klant te worden opgeslagen. Een klant kan op eenzelfde tijdstip slechts één bestelling plaatsen. Het is ook van belang om te weten of het gaat om een bestelling voor Kerst of voor oudejaar. In het geval van een bestelling voor oudejaar, biedt de traiteur immers een gratis fles champagne aan en dient de klant aan te geven of hij/zij een brut of een demi-sec wil. Een tweede verschil tussen Kerst en oudejaar wordt gemaakt in het feit dat het aanbod van producten die besteld kunnen worden op Kerst verschilt van het aanbod van producten die besteld kunnen worden op oudejaar. Uiteraard kunnen Kerstproducten enkel opgenomen worden in een bestelling voor Kerst, en kunnen oudejaarsproducten enkel opgenomen worden in een bestelling voor oudejaar. Sommige producten worden op beide feestdagen aangeboden.

Tenslotte dient ook alle data met betrekking tot de regeling voor het afhalen van bestellingen te worden opgeslagen. De traiteur heeft verschillende afhaalpunten, die elk een unieke naam hebben. Deze afhaalpunten hebben daarnaast elk een aantal verschillende openingstijden (bestaande uit een begin- en een eindtijdstip). Openingstijden van eenzelfde afhaalpunt mogen niet overlappen. Bij het bestellen kan de klant opgeven in welk afhaalpunt en tijdens welke openingsperiode hij/zij de bestelling wenst af te halen. Uiteraard kan een bestelling voor Kerst ten laatste op 24 december 2022 en een bestelling voor oudejaar ten laatste op 31 december 2022 worden afgehaald.

3 Logisch ontwerp

Zet het (abstract) EER-diagram gegeven in Figuur 1 om naar een logisch databankschema. Lijst in dit logisch databankschema alle tabellen, attributen, primaire sleutels, vreemde sleutels, NOT NULL-constraints en UNIQUE-constraints op. De datatypes van de attributen mogen achterwege gelaten worden en je mag ervan uitgaan

dat geen enkel attribuut optioneel is. Gebruik de notatie en de omzettingsregels die we hebben aangeleerd tijdens de oefeningenlessen.



Figuur 1: Abstract EER-diagram.

4 Workshops

In Appendix 1 is de DDL-code gegeven voor de aanmaak van een databank waarin data in verband met het uitlenen van boeken en cd's door personen binnen een werkgroep worden bijgehouden. Bekijk deze DDL-code nauwgezet en los daarna onderstaande vragen op.

1. Waar/niet waar. Het verwijderen van een (refererende) rij uit de tabel boek zorgt ervoor dat de gerefereerde rij uit de tabel item automatisch ook verwijderd wordt. Verklaar je antwoord.
2. Op dit moment is het mogelijk dat eenzelfde te ontlenen item wordt toegevoegd in de tabel boek en in de tabel cd. Hoe zou je het fysieke ontwerp aanpassen zodat eenzelfde te ontlenen item enkel kan worden toegevoegd in de tabel boek of in de tabel cd (maar niet in beide tabellen)? Beschrijf nauwgezet in woorden (en dus niet met code).
3. Beschouw de voorbeelddata van de tabellen persoon, item, boek en cd, gegeven in Appendix 2. Beschouw daarnaast onderstaande (genummerde) sequentie van SQL-commando's die inwerken op de databank. Hoe zal de tabel uitleenbeurt eruit zien na uitvoering van het laatste commando? Teken deze

tabel volledig. Je mag veronderstellen dat de tabel uitleenbeurt initieel leeg is en de syntax van de commando's correct is. Ook worden alle statements één voor één uitgevoerd (en dus niet in één keer). Statements die falen mag je dus negeren. We houden geen rekening met de correcte formatering van waarden in de getekende tabel.

1. INSERT INTO uitleenbeurt VALUES
 (1, '10/12/2022', null, 'jan.pieters@gmail.com', null);
2. INSERT INTO uitleenbeurt VALUES
 (2, '10/12/2022', '12/12/2022', 'jan.pieters@gmail.com', 3);
3. INSERT INTO uitleenbeurt VALUES
 (5, '10/12/2022', '12/12/2022', 'jan.pieters@gmail.com', 2);
4. INSERT INTO uitleenbeurt VALUES
 (1, '13/12/2022', '13/12/2022', 'sofie.pieters@gmail.com', 2);
5. INSERT INTO uitleenbeurt VALUES
 (2, '11/12/2022', '12/12/2022', 'sofie.pieters@gmail.com', 2);
6. INSERT INTO uitleenbeurt VALUES
 (1, '10/12/2022', '12/12/2022', 'marie.janssens@gmail.com', null);
7. INSERT INTO uitleenbeurt VALUES
 (4, '13/12/2022', '14/12/2022', 'jan.pieters@gmail.com', 4);
8. INSERT INTO uitleenbeurt VALUES
 (2, '12/12/2022', null, 'jan.pieters@gmail.com', null);
9. INSERT INTO uitleenbeurt VALUES
 (3, '18/12/2022', null, 'jan.pieters@gmail.com', null);
10. INSERT INTO uitleenbeurt VALUES
 (3, '19/12/2022', '28/12/2022', 'sofie.pieters@gmail.com', 5);

Appendix 1: DDL-code uitleen databank

```
CREATE TABLE persoon (  
    naam VARCHAR UNIQUE NOT NULL,  
    email VARCHAR PRIMARY KEY  
);  
  
CREATE TABLE item (  
    id INTEGER PRIMARY KEY,  
    titel VARCHAR NOT NULL,  
    auteur VARCHAR  
);  
  
CREATE TABLE boek (  
    isbn VARCHAR PRIMARY KEY,  
    id INTEGER UNIQUE NOT NULL,  
    uitgeverij VARCHAR NOT NULL,  
    CONSTRAINT boek_item_fk FOREIGN KEY (id) REFERENCES item (id)  
        ON DELETE CASCADE ON UPDATE CASCADE  
);  
  
CREATE TABLE cd (  
    ean VARCHAR PRIMARY KEY,  
    id INTEGER UNIQUE NOT NULL,  
    speelduur INTEGER NOT NULL CHECK (speelduur > 0),  
    CONSTRAINT cd_item_fk FOREIGN KEY (id) REFERENCES item (id)  
        ON DELETE CASCADE ON UPDATE CASCADE  
);  
  
CREATE TABLE uitleenbeurt (  
    id INTEGER,  
    begindatum DATE,  
    einddatum DATE,  
    email VARCHAR NOT NULL,  
    score INTEGER CHECK (score BETWEEN 0 AND 4),  
    CONSTRAINT uitleenbeurt_pk PRIMARY KEY (id, begindatum),  
    CONSTRAINT uitleenbeurt_unique UNIQUE (id, einddatum),  
    CONSTRAINT uitleenbeurt_item_fk FOREIGN KEY (id) REFERENCES item (id),  
    CONSTRAINT uitleenbeurt_persoon_fk FOREIGN KEY (email) REFERENCES persoon (email),  
    CONSTRAINT uitleenbeurt_data_check CHECK (begindatum < einddatum)  
);
```

Appendix 1: DDL-code uitleen databank (vervolg)

```
CREATE FUNCTION function1()
    RETURNS TRIGGER AS
$BODY$
BEGIN
    IF EXISTS(
        SELECT 1 FROM uitleenbeurt
            WHERE email = NEW.email AND einddatum IS NULL
        GROUP BY email
        HAVING COUNT(1) >= 2
    ) THEN
        RAISE EXCEPTION 'foutmelding!';
    END IF;
    RETURN NEW;
END;
$BODY$
LANGUAGE plpgsql;

CREATE TRIGGER trigger1
    BEFORE INSERT ON uitleenbeurt
    FOR EACH ROW EXECUTE PROCEDURE function1();
```

Appendix 2: Voorbeelddata uitleen databank

naam	email
jan	jan.pieters@gmail.com
marie	marie.janssens@gmail.com
sofie	sofie.pieters@gmail.com

Tabel 2: Voorbeelddata persoon tabel.

id	titel	auteur
1	Principes van databases	Guy De Tré
2	Computernetwerken	Kurose & Ross
3	Praktisch UML	Warmer & Kleppe
4	Engelse intensieve taalcursus	NULL

Tabel 3: Voorbeelddata item tabel.

isbn	id	uitgeverij
9789043035804	1	Pearson
9789043021203	2	Pearson
9789043039215	3	Pearson

Tabel 4: Voorbeelddata boek tabel.

ean	id	speelduur
9783632988465	4	240

Tabel 5: Voorbeelddata cd tabel.