



aantal pagina's.: ga tot einde	lector: Wim Bertels, Serhat Erdogan opleidingsonderdeel (OPO): Data Management  code OPO: B-UCLL-MBI11H
Datum: 20/06/24 groep: nvt	naam student: ..... voornaam: ..... studentennummer: r0956428.....
Hulpmiddelen: infra	maximumduur: <b>2u00</b>  Punten 10/20

### Prekondities

Lees eerst dit alvorens aan het examen te beginnen.

Het is een **examen via pc zonder algemene internettoegang**.

De maximumduur van het examen is 2 uur 00 minuten. Het examen zelf is voorzien om opgelost te kunnen worden op 1 uur 30 minuten. **Er wordt geen rekening gehouden met antwoorden die te laat of in de verkeerde vorm worden ingediend.**

Enkel antwoorden die op de gevraagde manier worden ingediend tellen; er wordt geen rekening gehouden met andere vormen van indiening. Je dient dus op de juiste manier in te dienen.

Dit examen is **persoonlijk per student**, het gedrag of de behandeling omtrent het naleven van deze regels van andere studenten -die zich mogelijk niet aan deze regels houden- geven u niet het recht de hier gegeven regels te overtreden. U werkt strikt individueel Het examenreglement is van toepassing. Overtredingen van de hier en in het OER gegeven regels worden beschouwd als fraude.

Bij de minste onduidelijkheid van deze regels brengt u onmiddellijk de beschikbare lector hiervan op de hoogte.

Het is jouw taak om te zorgen dat benodigde hardware, software en internet werken; net als de andere vereisten, zie hiervoor Toledo. Gelieve deze ruim op voorhand door te nemen en in orde te brengen (voor de laatste les die door de lectoren zelf gegeven wordt).

Hierbij stemt u in met deze regels en geeft u te kennen deze zonder misverstanden begrepen te hebben.

### Postkondities (voorbehouden)

--

**Veel Succes !**

groep: .....	pg. 2
naam student: .....	B-UCLL-MBI11H
voornaam: .....	20/06/24
studentennummer: r0956428.....	

Er zijn 3 onderdelen.

---

Zoals op een productiedatabank is het **aantal connecties beperkt**, hou dit in het achterhoofd.

Indien u de **databank 'stuk'** maakt, dan is dit **uw verantwoordelijkheid**. Ook voor het verdere verloop van het examen.

Je dient je script in via je **persoonlijke sftp ruimte in de map juni\_dataman2. Scripts, die de gegeven conventies niet volgen of fouten geven, resulteren in een nul voor dit onderdeel.**

Redundante info: dit is bv

hoofdletter gevoelig, juiste bestandsformaat en extensie, juiste naamgeving, voert uit zonder fouten wanneer het wordt geladen als sql script, ..

Controle scripts op het einde: heb je al de scripts ingediend in de map juni\_dataman2 in je persoonlijke sftp ruimte

( <sftp://r0956428@r0956428.webontwerp.ucll.be:22/> )?

Het gaat om de scripts: **local\_r0956428\_backup.sql**, **local\_r0956428\_backup.sql**, **local\_r0956428\_perf.sql** en **r0956428\_pl.sql**

(het is jouw taak om dit na te kijken!)

---

Te vermijden fouten, tips voor scripts:

- test of de scripts die je indient ook effectief uitvoeren, het is bv niet omdat je ergens iets werkend hebt gekregen, dat het script dat je indient ook uitvoert zonder fouten.
- test of hetgeen je aangemaakt hebt op de databank (zonder fouten), ook effectief werkt (bij uitvoering/gebruik).
- vergeet het schema niet (afhankelijk van de context) in elk script, en ook niet in je functie zelf (in de code en/of als parameter). Set search\_path is sessie afhankelijk, een opgeroepen functie moet altijd uitvoeren, onafhankelijk van de sessie instellingen waarin ze wordt opgeroepen.
- een select instructie verandert de structuur niet, het is immers een DML instructie.

Tip: maak het erd schema van je databank, zodat je de structuur visueel kan zien.

groep: .....	pg. 3
naam student: .....	B-UCLL-MBI11H
voornaam: .....	20/06/24
studentennummer: r0956428.....	

**De databank local\_r0956428 staat klaar op de server fuji.ucll.be, poort 52324 en is bereikbaar voor de gebruiker local\_r0956428.**

1. **Neem een reserve kopie van enkel de data, niet de structuur.**

Er worden dus bv geen create instructies verwacht, zorg dat er enkel ISO sql in je script staat. Zorg dat enkel de originele data aanwezig is (in het geval van aanpassingen verder in het examen).

**Doe dit enkel voor de tabellen onder het schema mammoet.**

Dien het **local\_r0956428\_backup.sql bestand** in op je repository in de map **juni\_dataman2**. Hou je aan deze naamconventies voor de evaluatie.

Dit r0956428\_backup.sql script dient zonder fouten uit voeren op elke lege databank met dezelfde structuur met **hetzelfde schema** en tabellen  
Er wordt dus een plat tekst bestand gevraagd (.sql).

Tip: heb je je local\_user niet?

Verbind je gebruiker r0956428 met de administration databank op poort 52324, server fuji.ucll.be. En voer uit (als in in reeks 7 zit, anders analoog aanpassen):

```
SELECT user_administration.set_session_svg('dataman','2T17');
```

```
SELECT user_administration.new_local_user('dataman');
```

-- deze geeft het wachtwoord (, dit verandert steeds bij uitvoering van de functie)

groep: .....	pg. 4
naam student: .....	B-UCLL-MBI11H
voornaam: .....	20/06/24
studentennummer: r0956428.....	

2. Optimaliseer volgende query (tabellen onder het schema mammoet):

```
select *
from toegangspunt
where gebouw similar to '%((77)|(99))';
```

Schrijf een script **local\_r0956428\_perf.sql** waarin je toont dat je query geoptimaliseerd is. Maak gebruik van een **transactie die vastgelegd wordt**. Bij uitvoering van je script zou duidelijk uit de uitvoer moeten blijken dat je de query verbeterd hebt. Gebruik korte, duidelijke commentaar in je script.

De uitvoer van de query moet hetzelfde blijven ongeacht de omstandigheden.

Een aanzet van het script zou het onderstaande kunnen bevatten (alleen ter illustratie, bevat mogelijk fouten, jouw script kan anders zijn):

```
-- commentaar, onvolledig, enkel om de structuur van het script aan te geven.
-- er is sprake van een transactie ..
INSTRUKTIE;
-- er is een vertrekpunt ..
INSTRUKTIE;
-- commentaar ..
INSTRUKTIE;
-- commentaar
..
INSTRUKTIE;
-- commentaar
```

groep: .....	pg. 5
naam student: .....	B-UCLL-MBI11H
voornaam: .....	20/06/24
studentennummer: r0956428.....	

Optionele tip: (de getallen kunnen variëren)

```
select *
from toegangspunt
where gebouw similar to '%((77)|(99))';
```

#### QUERY PLAN

Seq Scan on toegangspunt (cost=0.00..38.00 rows=93 width=16)  
 Filter: (gebouw ~ '^((?:.\*(?:77)|(?:99)))\$')::text)

-- en !

#### QUERY PLAN

-----  
 Bitmap Heap Scan on toegangspunt (cost=8.71..22.42 rows=20 width=16)  
 Recheck Cond: ("right"(gebouw, 2) = ANY ('{77,99}'::text[]))  
 -> Bitmap Index Scan on x (cost=0.00..8.71 rows=20 width=0)  
 Index Cond: ("right"(gebouw, 2) = ANY ('{77,99}'::text[]))

groep: .....  
naam student: .....  
voornaam: .....  
studentennummer: r0956428.....

pg. 6  
B-UCLL-MBI11H  
20/06/24

3. Dien dit script in en pas het ook toe **op je persoonlijk databank op fuji.ucll.be.**

Je dient dit script in zoals de andere scripts, noem je **script studentnr\_pl.sql** (waarbij je studentnr vervangt door je eigen studenten nummer). De wijzigingen van dit script worden **vastgelegd in je persoonlijke databank** op fuji. Dit is een vraag over procedurele sql.

Scripts, die de gegeven conventies niet volgen of fouten geven, resulteren in een nul voor dit onderdeel. Redundante info: dit is bv hoofdletter gevoelig, juiste bestandsformaat en extensie, juiste naamgeving, voert uit zonder fouten wanneer het wordt geladen als sql script, ..

a. Schrijf **een functie die op basis van de tabel abonnementen een cumulatief overzicht van de aansluitingskosten geeft per dagtarief gesorteerd volgens naam.**

De aansluitingskosten wordt er dus voor elke rij in het resultaat cumulatief bijgeteld.

Geef ook het cumulatieve relatieve percentage weer binnen elke groep.

Noem je **functie cum\_perc\_aansluitingskosten.**

De gevraagde **kolommen** zijn:

**dagtarief, aansluitingskosten, c\_aansluitingskosten, p\_aansluitingskosten, naam**

Hierbij staat c\_aansluitingskosten voor de cumulatieve aansluitingskosten en p\_aansluitingskosten voor de relatieve cumulatieve aansluitingskosten (het relatief cumulatief percentage)

**Zorg dat deze functie ook effectief werkt.**

Voorbeeld van een mogelijke uitvoer

---	-----	-----	-----	---
t1	250.00	250.00	0.35714285714285714286	1
t1	150.00	400.00	0.57142857142857142857	1
t1	300.00	700.00	1.00000000000000000000	1
t2	2000.00	2000.00	0.50000000000000000000	1
t2	2000.00	4000.00	1.00000000000000000000	2

t1 is hier een dagtarief; 250, 150 en 300 de aansluitingskosten opeenvolgend; met als uiteindelijk de totale c\_aansluitingskosten 700; en deze 700 is 1.000 (of 100%) van 700 (p\_aansluitingskosten); de volgorde voor de berekening.

(Je mag eventueel extra data toevoegen in je script om uitgebreider te testen)

Bv je script zou het onderstaande kunnen bevatten (alleen ter illustratie, bevat mogelijk fouten, jouw script zal anders zijn)

```
BEGIN;  
-- commentaar, onvolledig, enkel om de structuur van het script aan te geven.  
CREATE ..  
-- test  
SELECT cum_perc_aansluitingskosten();  
-- test nagekeken en ok, vastleggen  
-- de wijzigingen zijn definitief  
COMMIT;
```