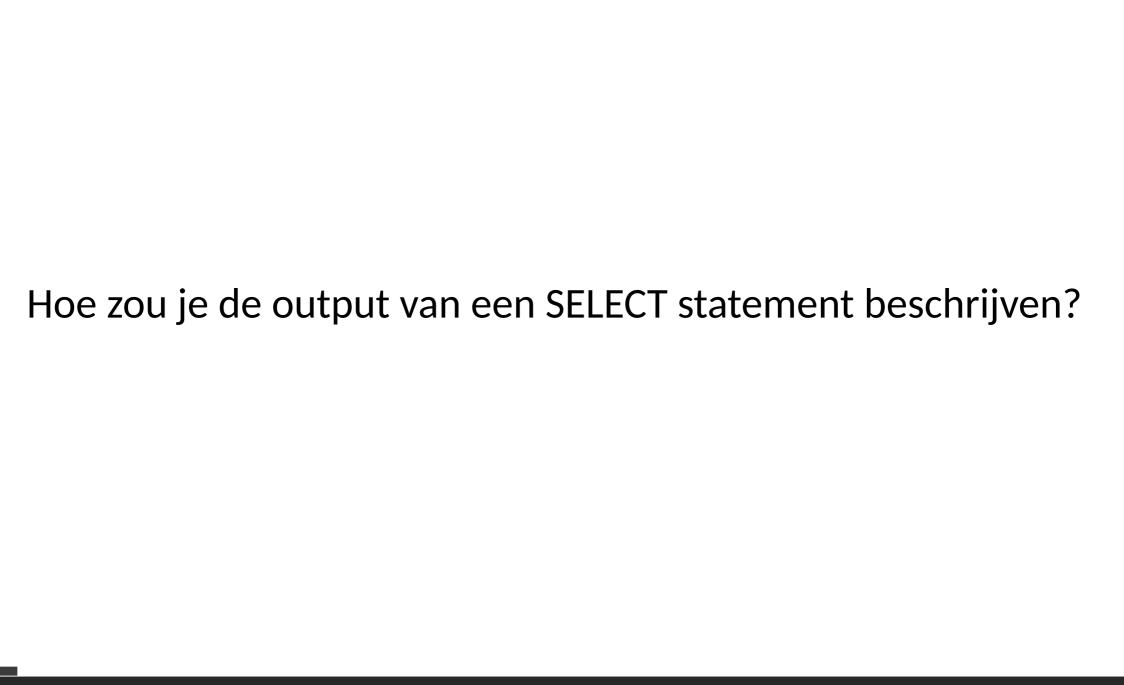
Subqueries 1

Niet-Gecorreleerde Subqueries

wim.bertels@ucll.be

Naamsvermelding-NietCommercieel-GelijkDelen 4.0 Unported Licentie



Wat is een subquery?

- Een tabelexpressie binnen een tabelexpressie
- Resultaat wordt doorgegeven aan aanroepende tabelexpressie
- Subqueries mogen genest zijn

Waarom gebruiken we subquery's?

- Query opsplitsen in deelproblemen die je kan oplossen en de output ervan verder gebruiken
- Zoals bij programmeren: een complexere methode opsplitsen in eenvoudigere (atomaire) taken

Soorten subquery's

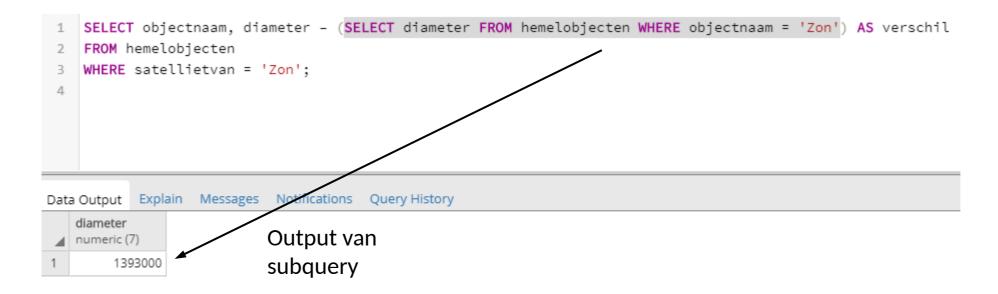
- Scalaire subquery: output = 1 rij, 1 kolom (dus 1 waarde)
- Rij-subquery: output = 1 rij
- Kolom-subquery: output = meerdere rijen met elk 1 waarde
- Tabel-subquery: output = meerdere rijen en kolommen

Voorbeeld:

Geef voor elke planeet hoeveel groter of kleiner deze is dan de zon.

Voorbeeld:

Geef voor elke planeet hoeveel groter of kleiner deze is dan de zon.



```
SELECT objectnaam, diameter –

(SELECT diameter
FROM hemelobjecten
WHERE objectnaam = 'Zon')
AS verschil

FROM hemelobjecten
WHERE satellietvan = 'Zon';
```

Voorbeeld 2:

Geef de hemellichamen met een diameter groter dan Venus.

Voorbeeld 2:

Geef de hemellichamen met een diameter groter dan Venus.

```
SELECT objectnaam, diameter
    FROM hemelobjecten
    WHERE diameter >
     (SELECT diameter
    FROM hemelobjecten
    WHERE objectnaam = 'Venus');
           Explain
                  Messages
                             Notifications
Data Output
   diameter
                   Output van
   numeric (7)
                   subquery
          12104
```

```
SELECT objectnaam, diameter
FROM hemelobjecten
WHERE diameter >
  (SELECT diameter
  FROM hemelobjecten
  WHERE objectnaam = 'Venus');
```

Rij-subquery (1 rij)

Voorbeeld:

Geef alle spelers met hetzelfde geslacht en dezelfde woonplaats als de speler met nummer 7.

Rij-subquery (1 rij)

Voorbeeld:

Geef alle spelers met hetzelfde geslacht en dezelfde woonplaats als de speler met nummer 7.

```
SELECT spelersnr, naam, plaats
    FROM spelers
    WHERE (plaats, geslacht) =
     (SELECT plaats, geslacht
    FROM spelers
    WHERE spelersnr = 7);
               Output van
                              Notifications
                                            Query Histor
Data Output
               subquervages
                          geslacht
    plaats
    character varying (30)
                          character (1)
   Den Haag
                          M
```

Rij-subquery (1 rij)

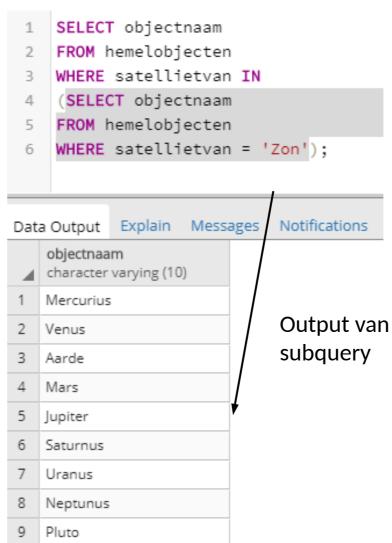
```
SELECT spelersnr, naam, plaats
FROM spelers
WHERE (plaats, geslacht) =
   (SELECT plaats, geslacht
   FROM spelers
   WHERE spelersnr = 7);
```

Kolom-subquery (meerdere rijen, elk 1 waarde)

Voorbeeld: Geef alle manen. Kolom-subquery (meerdere rijen, elk 1

waarde)

Voorbeeld: Geef alle manen.



Kolom-subquery (meerdere rijen, elk 1 waarde)

```
SELECT objectnaam
FROM hemelobjecten
WHERE satellietvan IN
(SELECT objectnaam
FROM hemelobjecten
WHERE satellietvan = 'Zon');
```

- Geeft een tijdelijk resultaat
- Subquery moet een pseudoniem krijgen als de subquery in de from staat van de originele query

Voorbeeld:

Geef de reizen die een hemelobject bezoeken dat over alle reizen heen minstens 5 keer bezocht wordt.

Voorbeeld:

Geef de reizen die een hemelobject bezoeken dat over alle reizen heen minstens 5 keer bezocht wordt.

Tussenstap:

SELECT objectnaam

FROM

bezoeken

GROUP BY objectnaam

HAVING COUNT(*) >= 5;

Voorbeeld:

Geef de reizen die een hemelobject bezoeken dat over alle reizen heen

minstens 5 keer bezocht wordt.

```
SELECT reizen.reisnr, reizen.vertrekdatum
    FROM reizen
    INNER JOIN bezoeken b using (reisnr)
    INNER JOIN (
     SELECT objectnaam
 6 FROM bezoeken
7 GROUP BY objectnaam
8 HAVING COUNT(*) >= 5
9 ) AS veelbez ON b.objectnaam = veelbez.objectnaam
    GROUP BY reizen.reisnr, reizen.vertrekdatum;
                           Notifications Query History
Data Output Explain Messages
   objectnaam
   character varying (10)
                            Output van
   Maan
                            subquery
```

SELECT reizen.reisnr, reizen.vertrekdatum

FROM reizen INNER JOIN bezoeken b USING (reisnr)

INNER JOIN

(SELECT objectnaam

FROM bezoeken

GROUP BY objectnaam

HAVING COUNT(*) >= 5

AS veelbez

ON b.objectnaam = veelbez.objectnaam GROUP BY reizen.reisnr, reizen.vertrekdatum;

Nog iets klein

Geef alle reizen die geen bezoek hebben gebracht aan de maan.

Nog iets klein

Geef alle reizen die geen bezoek hebben gebracht aan de maan.

SELECT reizen.reisnr

FROM reizen INNER JOIN bezoeken USING (reisnr)

WHERE bezoeken.objectnaam <> 'Maan'

GROUP BY reizen.reisnr;

?

Nog iets klein

Geef alle reizen die geen bezoek hebben gebracht aan de maan.

SELECT reizen.reisnr FROM reizen INNER JOIN bezoeken USING (reisnr) WHERE bezoeken.objectnaam <> 'Maan' GROUP BY reizen.reisnr;

Waarom fout?

Check: Alle reizen

SELECT reisnr FROM reizen;

Oplossing (tussenstap)

Alle reizen die WEL de maan hebben bezocht:

SELECT reizen.reisnr

FROM reizen INNER JOIN bezoeken USING (reisnr)

WHERE bezoeken.objectnaam = 'Maan'

GROUP BY reizen.reisnr;

Oplossing

Alle reizen die GEEN bezoek hebben gebracht aan de maan.

```
SELECT reisnr

FROM reizen

WHERE reisnr NOT IN

(SELECT reisnr

FROM bezoeken

WHERE objectnaam = 'Maan');
```

Uitdaging

Probeer deze zonder subquery te schrijven (op een regenachtige dag.., als je echt teveel tijd hebt..):

```
SELECT avg(totaal)
FROM
(SELECT spelersnr, sum(bedrag) as total
FROM boetes
GROUP BY spelersnr) as totalen
```

Wat toont deze query?

Wim Bertels (CC)BY-SA-NC

Referenties:

Slides subqueries deel 1 sql 2012-13, K. Beheydt

SQL Leerboek, R. Van der lans