

MIN- en MAX-functie op Alfnumerieke Waarden



Inleiding

In het boek Databases en SQL staat op bladzijde 96 een opsomming van functies die in een query gebruikt kunnen worden onder SELECT en onder HAVING. Het betreft de functies COUNT, SUM, AVG, MAX en MIN.

Onder de uitleg van deze functies staat de tekst: *'voor alle functies geldt dat ze uitgevoerd moeten kunnen worden. Een minimum, maximum, gemiddelde of som kan normaal gesproken alleen van numerieke waarden worden bepaald. Bovendien moeten er waarden bestaan'*

Deze tekst is inhoudelijk correct en richt zich op het doel van de functies. De tekst gaat echter niet in op de mogelijkheid om de functies MIN en MAX ook los te laten op alfanumerieke waarden (iets wat niet kan met de functies AVG en SUM).

Ik zal hierna aangeven hoe we de functies MIN en MAX voor alfanumerieke waarden kunnen gebruiken en ik zal tevens voorbeelden geven van toepassingen.

Toepassen in de SELECT-clausule

voorbeeld met MIN en MAX in SELECT

Zoals gezegd kunnen we de functies MIN en MAX toepassen op alfanumerieke kolommen. Een voorbeeld hiervan is:

```
SELECT MIN(WNAAM), MAX(WNAAM)
FROM WERKNEMER
```

De uitkomst is:

```
-----
MIN(WNAAM) MAX(WNAAM)
-----
AARTS      VOS
-----
```

De uitkomst laat zien dat:

- de functie MIN uitgaat van de *eerste* waarde die je krijgt wanneer je de waarden in de kolom alfabetisch sorteert
- de functie MAX uitgaat van de *laatste* waarde die je krijgt wanneer je de waarden in de kolom alfabetisch sorteert

De uitkomst kan gebruikt worden in vraagstellingen als:

'de namen lopen van tot

Toepassen in combinatie met het gebruik van GROUP BY

voorbeeld MIN en MAX in SELECT in combinatie met GROUP BY

Het gebruiken van de functies MIN of MAX kan ook handig zijn bij de volgende probleemstelling: 'geef de nummers en namen van de werknemers die meer dan één vervanger hebben, alsmede het aantal vervangers die ze hebben'

De SELECT-opdracht is:

```
SELECT W.W#, MIN(WNAAM), COUNT(*)
FROM WERKNEMER W, VERVANGING V
WHERE W.W# = V.VERVANGENE
GROUP BY W.W#
```

De uitkomst is:

```
-----
W.W#      MIN(WNAAM) COUNT(*)
-----
4         CHIN      3
6         EVERS     2
9         JANSEN    2
10        KAPER     2
-----
```

Door het gebruik van de functie MIN kan de naam van een werknemer afgedrukt worden. Het gebruiken van de functie MAX in plaats van de functie MIN geeft dezelfde uitkomst. Dit komt omdat in alle rijen voor een bepaald werknemernummer (w#) dezelfde waarde voor de naam van de werknemer (wnaam) voorkomt.

hetzelfde voorbeeld zonder MIN of MAX

Het opnemen van de kolom wnaam zonder de functie MIN of MAX is in de SELECT-opdracht niet toegestaan. Dit is zo omdat de kolom wnaam niet achter GROUP BY staat.

Uiteraard is een alternatief hiervoor dat je wnaam als kolom achter GROUP BY opneemt:

```
SELECT W.W#, WNAAM, COUNT(*)
FROM WERKNEMER W, VERVANGING V
WHERE W.W# = V.VERVANGENE
GROUP BY W.W#, WNAAM
```

De kop van de uitkomst is dan wel anders, maar dat is slechts een detail. Er staat namelijk geen MIN(...) meer in:

```
-----
W.W#      WNAAM      COUNT(*)
-----
4         CHIN      3
6         EVERS     2
9         JANSEN    2
10        KAPER     2
-----
```

MIN- en MAX-functie op Alfnumerieke Waarden



Toepassen in de HAVING-clausule

voorbeeld MIN en MAX in HAVING

Een andere plaats waar we de functies MIN en MAX op alfanumerieke waarden kunnen toepassen is in de HAVING-clausule. Met HAVING kunnen we bijvoorbeeld bepalen of voor een kolom allemaal dezelfde waarden voorkomen, of juist niet.

Bedenk dat dit ook anders opgelost kan worden.

Het voorbeeld geeft een oplossing voor de vraagstelling:

'Geef de functies die steeds bij dezelfde soort afdeling voorkomen. Geef ook aan hoe vaak de functie voorkomt en welke afdeling dit betreft'

De query:

```
SELECT FNAAM, COUNT(*), MIN(AFDELING)
FROM WERKNEMER
GROUP BY FNAAM
HAVING MIN(AFDELING) = MAX(AFDELING)
```

De uitkomst is:

```
-----
FNAAM      COUNT(*)  MIN(AFDELING)
-----
CONFERENCIER  2      BALIE
KOK          3      KEUKEN
OBER         1      RESTAURANT
SECRETARIS   1      DIRECTIE
-----
```

De gedachte achter de query is dat wanneer bij een bepaalde functie maar één soort afdeling voorkomt de functies MIN en MAX dezelfde waarde moeten hebben.

De vraagstelling kan uiteraard ook opgelost worden met de query:

```
SELECT FNAAM, COUNT(*), MIN(AFDELING)
FROM WERKNEMER
GROUP BY FNAAM
HAVING COUNT(DISTINCT AFDELING) = 1
```

Deze query geeft precies dezelfde uitkomst.

Toepassen in de SELECT-clausule van een subquery

voorbeeld van MIN en MAX als uitkomst van een subquery

Wanneer we de functies MIN en MAX kunnen gebruiken in de SELECT-clausule, dan kunnen we ze ook gebruiken als uitkomst van een subquery. Ik geef hiervan twee voorbeelden. Het eerste betreft de pro-

bleemstelling:

'Geef de gegevens van de vestiging die wat naam betreft alfabetisch het eerste komt'

De query:

```
SELECT *
FROM VESTIGING
WHERE VESNAAM =
      (SELECT MIN(VESNAAM)
       FROM VESTIGING)
```

De uitkomst is:

```
-----
VESNAAM    BRANCHE    PLAATS
-----
FONG        HORECA     ARNHEM
-----
```

Het tweede voorbeeld betreft de probleemstelling:

'Geef de naam, de geboortedatum en woonplaats van de oudste lener'

De query:

```
SELECT LNAAM, GEBDAT, WPLAATS
FROM LENER
WHERE GEBDAT =
      (SELECT MIN(GEBDAT)
       FROM LENER)
```

De uitkomst is:

```
-----
LNAAM      GEBDAT      WPLAATS
-----
HORN        1936-03-21  LOENEN
-----
```

Aan de streepjes tussen jaar, maand en dag in de kolom gebdat (geboortedatum) is te zien dat de geboortedatum is vastgelegd als een alfanumerieke kolom.