

# SQL



## Onderwerpen:

inleiding

basiscommando's

functies

enkel-/meervoudige condities

sorteren

joins

subqueries

aggregeren

groeperen

tijdelijke tabellen

correlated subqueries

samenvoegqueries: union

muteren

## Onderwerpen:

inleiding

basiscommando's

functies

enkel-/meervoudige condities

sorteren

joins

subqueries

aggregeren

**groeperen**

tijdelijke tabellen

correlated subqueries

samenvoegqueries: union

mteren



Met de functies MAX, MIN, SUM, AVG en COUNT kun je gegevens in een selectie samennemen.

Deze aggregeren gegevens tot één regel

**Voorbeeld:**

```
SELECT SUM(korting)
FROM Bestelling
INNER JOIN Klant
ON Bestelling.klant_code=Klant.klant_code
WHERE (YEAR(datum)=2015) AND (naam='Dennis Nijhout');
```

# SQL - Aggregeren

Het bedrag dat Dennis Nijhout in totaal heeft betaald:

```
SELECT SUM(aantal*(basisprijs+
                Formaat.plusprijs+Bodem.plusprijs))
FROM Klant
    INNER JOIN Bestelling
    ON Klant.klant_code=Bestelling.klant_code
    INNER JOIN BesteldePizza
    ON Bestelling.bestel_code=BesteldePizza.bestel_code
    INNER JOIN Pizza
    ON BesteldePizza.pizza_code=Pizza.pizza_code
    INNER JOIN Formaat
    ON BesteldePizza.formaat_code=Formaat.formaat_code
    INNER JOIN Bodem
    ON BesteldePizza.bodem_code=Bodem.bodem_code
WHERE Klant.naam='Dennis Nijhout';
```

# SQL - Aggregeren

Het bedrag dat Dennis Nijhout in totaal heeft betaald:

**Dilemma 1: op welke manier verreken je de korting?**

```
SELECT SUM(aantal * (basisprijs +
                    Formaat.plusprijs + Bodem.plusprijs))
FROM Klant
     INNER JOIN Bestelling
     ON Klant.klant_code = Bestelling.klant_code
     INNER JOIN BesteldePizza
     ON Bestelling.bestel_code = BesteldePizza.bestel_code
     INNER JOIN Pizza
     ON BesteldePizza.pizza_code = Pizza.pizza_code
     INNER JOIN Formaat
     ON BesteldePizza.formaat_code = Formaat.formaat_code
     INNER JOIN Bodem
     ON BesteldePizza.bodem_code = Bodem.bodem_code
WHERE Klant.naam = 'Dennis Nijhout';
```

# SQL - Aggregeren

Het bedrag dat Dennis Nijhout in totaal heeft betaald:

```
SELECT SUM(aantal * (basisprijs +  
                Formaat.plusprijs + Bodem.plusprijs))  
FROM Klant  
    INNER JOIN Bestelling  
    ON Klant.klant_code = Bestelling.klant_code  
    INNER JOIN BesteldePizza  
    ON Bestelling.bestel_code = BesteldePizza.bestel_code  
    INNER JOIN Pizza  
    ON BesteldePizza.pizza_code = Pizza.pizza_code  
    INNER JOIN Formaat  
    ON BesteldePizza.formaat_code = Formaat.formaat_code  
    INNER JOIN Bodem  
    ON BesteldePizza.bodem_code = Bodem.bodem_code  
WHERE Klant.naam = 'Dennis Nijhout';
```

**Dilemma 2: hoe toon je het  
totaalbedrag van iedere klant?**

# SQL - Groeperen

Oplossing:  
Grouperen



# SQL - Groeperen

## GROUP BY

Groepeert data uit SELECT  
maakt per groep één regel met de geaggregeerde waarden

Komt na FROM en WHERE

# SQL - Groeperen

## Voorbeeld 1

Per klant de klantcode en het aantal bestellingen:

klant_code	COUNT(*)
101	17
103	12
104	10
105	10
107	9
109	11
112	22
115	16
118	15
119	7

```
SELECT klant_code, COUNT(*)  
FROM Bestelling  
GROUP BY klant_code;
```

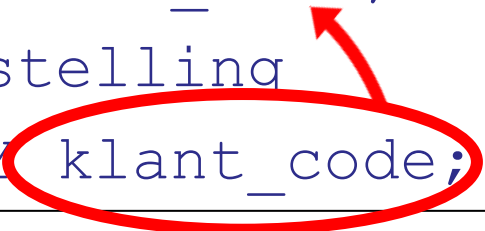
# SQL - Groeperen

## Voorbeeld 1

Per klant de klantcode en het aantal bestellingen:

klant_code	COUNT(*)
101	17
103	12
104	10
105	10
107	9
109	11
112	22
115	16
118	15
119	7

```
SELECT klant_code, COUNT(*)  
FROM Bestelling  
GROUP BY klant_code;
```



"GROUP BY-attribuut"

# SQL - Groeperen

Alleen "GROUP BY-attributen" kunnen worden getoond  
in het SELECT statement.

Van de overige attributen kunnen alleen  
geaggregeerde waarden worden getoond.

# SQL - Groeperen

## Voorbeeld 2

De code en de naam van iedere pizza met het aantal keer dat deze pizza is besteld:

```
SELECT Pizza.pizza_code, naam, SUM(aantal)
FROM Pizza
INNER JOIN BesteldePizza
ON Pizza.pizza_code=BesteldePizza.pizza_code
GROUP BY Pizza.pizza_code, naam;
```

# SQL - Groeperen

## Voorbeeld 3

De naam van iedere bezorger met het aantal klanten uit Soest waar deze bezorger ooit een bestelling heeft afgegeven:

```
SELECT Bezorger.naam, COUNT(DISTINCT Klant.klant_code)
FROM Bezorger
INNER JOIN Bestelling
ON Bezorger.bezorger_code=Bestelling.bezorger_code
INNER JOIN Klant
ON Bestelling.klant_code=Klant.klant_code
WHERE plaats='Soest'
GROUP BY Bezorger.naam;
```

# SQL - Groeperen

## Voorbeeld 3

De naam van iedere bezorger met het aantal klanten uit Soest waar deze bezorger ooit een bestelling heeft afgegeven:

```
SELECT Bezorger.naam, COUNT(DISTINCT Klant.klant_code)
FROM Bezorger
INNER JOIN Bestelling
ON Bezorger.besorger_code=Bestelling.besorger_code
INNER JOIN Klant
ON Bestelling.klant_code=Klant.klant_code
WHERE plaats='Soest'
GROUP BY Bezorger.naam;
```

Wat nou als er 2 bezorgers zijn met dezelfde naam?

# SQL - Groeperen

## Voorbeeld 3

De naam van iedere bezorger met het aantal klanten uit Soest waar deze bezorger ooit een bestelling heeft afgegeven:

```
SELECT Bezorger.naam, COUNT(DISTINCT Klant.klant_code)
FROM Bezorger
INNER JOIN Bestelling
ON Bezorger.besorger_code=Bestelling.besorger_code
INNER JOIN Klant
ON Bestelling.klant_code=Klant.klant_code
WHERE plaats='Soest'
GROUP BY Bezorger.naam, Bezorger.besorger_code;
```



# SQL - Groeperen

## Voorbeeld 3

De naam van iedere bezorger met het aantal klanten uit Soest waar deze bezorger ooit een bestelling heeft afgegeven:

```
SELECT Bezorger.naam, COUNT(DISTINCT Klant.klant_code)
FROM Bezorger
INNER JOIN Bestelling
ON Bezorger.besorger_code=Bestelling.besorger_code
INNER JOIN Klant
ON Bestelling.klant_code=Klant.klant_code
WHERE plaats='Soest'
GROUP BY Bezorger.naam, Bezorger.besorger_code
```

En de bezorgers die nog nooit iets hebben afgegeven in Soest?

# SQL - Groeperen

## Voorbeeld 3

De naam van iedere bezorger met het aantal klanten uit Soest waar deze bezorger ooit een bestelling heeft afgegeven:

```
SELECT Bezorger.naam, COUNT(DISTINCT Klant.klant_code)
FROM Bezorger
LEFT OUTER JOIN Bestelling
ON Bezorger.besorger_code=Bestelling.besorger_code
INNER JOIN Klant
ON Bestelling.klant_code=Klant.klant_code
WHERE plaats='Soest'
GROUP BY Bezorger.naam, Bezorger.besorger_code;
```

# SQL - Groeperen

## Voorbeeld 3

De naam van iedere bezorger met het aantal klanten uit Soest waar deze bezorger ooit een bestelling heeft afgegeven:

```
SELECT Bezorger.naam, COUNT(DISTINCT Klant.klant_code)
FROM Bezorger
LEFT OUTER JOIN Bestelling
ON Bezorger.bezorger_code=Bestelling.bezorger_code
INNER JOIN Klant
ON Bestelling.klant_code=Klant.klant_code
WHERE plaats='Soest'
GROUP BY Bezorger.naam, Bezorger.bezorger_code;
```

## Oefenopgaven SQL - Groeperen



Maak een query waarmee wordt getoond:

37. Het aantal bestellingen per dag (per datum)
38. De naam van iedere bezorger met het aantal bestelling in 2015
39. De bestelcode en de totaalprijs van iedere bestelling
40. De pizzacode van iedere pizza en het aantal (verschillende) klanten dat deze pizza heeft besteld