

Computerclub Volwassenen,  
Jeugd en Informatica vzw  
[www.vji.be](http://www.vji.be)

Voorbeelden en oefeningen bij demo SQL



Stefan Cruysberghs

[www.scip.be](http://www.scip.be)

Februari 2003

Voorbeelden .....	3
SELECT .....	3
DISTINCT.....	3
WHERE.....	3
IN & BETWEEN .....	4
SUM, AVG, MAX, MIN, COUNT.....	4
ORDER BY .....	5
GROUP BY .....	5
UNION .....	5
FUNCTIONS .....	6
JOINS .....	6
INNER JOIN .....	6
LEFT JOIN.....	7
SUBSELECTS & EXISTS.....	7
Oefeningen .....	8
Oplossingen oefeningen .....	9

## Voorbeelden

### **SELECT**

```
SELECT *  
FROM animals
```

```
SELECT name, area  
FROM animals
```

```
SELECT a.name, a.area, a.weight  
FROM animals a
```

```
SELECT a.name, (a.weight * 2) as doubleweight  
FROM animals a
```

### **DISTINCT**

```
SELECT DISTINCT a.area  
FROM animals a
```

### **WHERE**

```
SELECT a.*  
FROM animals a  
WHERE a.weight > 5
```

```
SELECT a.*  
FROM animals a  
WHERE a.area = 'South America'
```

```
SELECT a.*  
FROM animals a  
WHERE NOT a.area = 'South America'
```

```
SELECT a.*  
FROM animals a  
WHERE a.area <> 'South America'
```

```
SELECT a.*  
FROM animals a  
WHERE a.area like 'South%'
```

```
SELECT a.*  
FROM animals a  
WHERE a.area like 'South%'  
AND a.weight < 7
```

```
SELECT a.*  
FROM animals a  
WHERE  
a.weight = 5  
OR a.weight = 2
```

```
SELECT a.*
FROM animals a
WHERE
  a.weight <= 5
  AND a.weight >= 2
```

```
SELECT a.*
FROM animals a
WHERE
  a.area <> 'South America'
  AND a.name like '%a%'
  AND a.weight < 5
```

## ***IN & BETWEEN***

```
SELECT a.*
FROM animals a
WHERE
  a.weight in (2,5)
```

```
SELECT a.*
FROM animals a
WHERE
  a.area in ('South America','New Orleans')
```

```
SELECT a.*
FROM animals a
WHERE
  a.weight BETWEEN 2 AND 5
```

## ***SUM, AVG, MAX, MIN, COUNT***

```
SELECT sum(a.weight)
FROM animals a
```

```
SELECT max(a.weight)
FROM animals a
```

```
SELECT avg(a.weight)
FROM animals a
WHERE a.area = 'South America'
```

## **ORDER BY**

```
SELECT a.name, a.area, a.weight
FROM animals a
ORDER BY a.weight
```

```
SELECT a.name, a.area, a.weight
FROM animals a
ORDER BY a.area, a.name
```

```
SELECT a.name, a.area, a.weight
FROM animals a
ORDER BY a.weight DESC
```

```
SELECT a.name, a.area, a.weight
FROM animals a
ORDER BY a.area ASC, a.weight DESC
```

## **GROUP BY**

```
SELECT a.area, COUNT(a.name)
FROM animals a
GROUP BY a.area
```

```
SELECT a.area, AVG(a.weight)
FROM animals a
GROUP BY a.area
```

```
SELECT a.area, AVG(a.weight)
FROM animals a
WHERE a.area <> 'South America'
GROUP BY a.area
ORDER BY a.area DESC
```

```
SELECT a.area, AVG(a.weight), MAX(a.weight), MIN(a.weight), COUNT(a.name)
FROM animals a
GROUP BY a.area
```

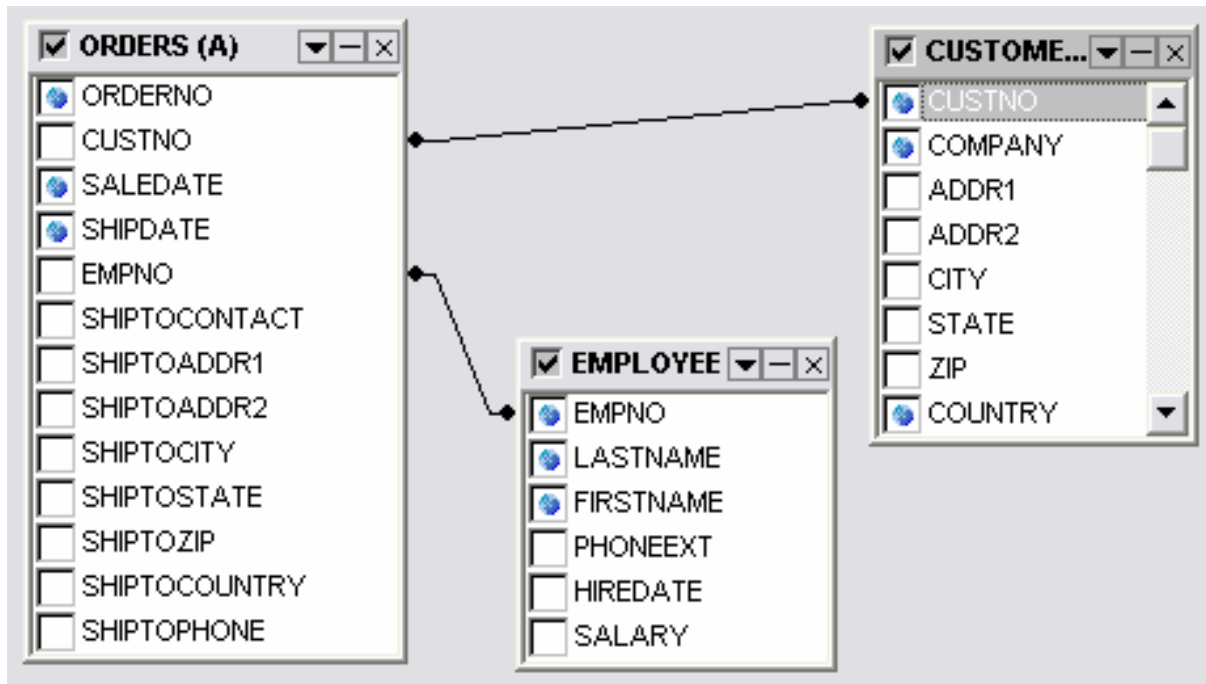
## **UNION**

```
SELECT a.*, 'Y' AS HighWeight
FROM animals a
WHERE a.weight > 10
UNION
SELECT a.*, 'N' AS HighWeight
FROM animals a
WHERE a.weight < 10
```

## FUNCTIONS

```
SELECT UPPER(a.name) AS name
FROM animals a
```

## JOINS



## INNER JOIN

```
SELECT a.orderno, a.saledate, b.custno, b.company, b.country
FROM orders a, customer b
WHERE
    b.custno = a.custno
```

```
SELECT a.orderno, a.saledate, b.custno, b.company, b.country
FROM orders a
    INNER JOIN customer b ON b.custno = a.custno
```

```
SELECT a.orderno, a.saledate, b.custno, b.company, b.country
FROM orders a, customer b
WHERE
    b.custno = a.custno
    AND b.country = 'Canada'
    AND a.saledate > '1/1/1990'
```

```
SELECT a.orderno, a.saledate, b.custno, b.company, b.country, c.empno,
c.lastname
FROM orders a
  INNER JOIN customer b ON b.custno = a.custno
  INNER JOIN employee c ON c.empno = a.empno
```

```
SELECT a.orderno, a.saledate, b.custno, b.company, b.country, c.empno,
c.lastname
FROM orders a, customer b, employee c
WHERE
  b.custno = a.custno
  AND c.empno = a.empno
```

```
SELECT a.orderno, a.saledate, b.custno, b.company, b.country
FROM orders a
  INNER JOIN customer b ON b.custno = a.custno
WHERE
  b.country = 'Canada'
  AND a.saledate > '1/1/1990'
```

```
SELECT a.custno, a.company, a.country, b.orderno
FROM customer a
  INNER JOIN orders b ON b.custno = a.custno
WHERE
  a.country = 'Canada'
ORDER BY a.custno, b.orderno
```

```
SELECT a.custno, a.company, a.country, b.orderno, b.saledate
FROM customer a
  INNER JOIN orders b ON b.custno = a.custno
WHERE
  a.country = 'US'
```

## LEFT JOIN

```
SELECT a.custno, a.company, a.country, b.orderno, b.saledate
FROM customer a
  LEFT JOIN orders b ON b.custno = a.custno
WHERE
  a.country = 'US'
```

```
SELECT a.custno, a.company, a.country, b.orderno, b.saledate
FROM customer a
  LEFT JOIN orders b ON b.custno = a.custno
WHERE
  a.country = 'US'
  AND b.saledate is null
```

## SUBSELECTS & EXISTS

```
SELECT a.custno, a.company, a.country
FROM customer a
WHERE NOT EXISTS
  (SELECT 1 FROM orders b
   WHERE b.custno = a.custno)
  AND a.country = 'US'
```

## Oefeningen

1. Toon een lijst van werknemers die na 1 januari 1992 zijn aangenomen en die ondertussen meer dan 35 000 dollar verdienen. (12 records)
2. Hoeveel loonkosten zijn er voor alle werknemers (1386202.259 dollar)
3. Toon een lijst van alle klanten uit Orlando en Florida. Sorteer de lijst op de firmanaam. (9 records)
4. Toon een lijst van klanten waarbij de naam van de stad de letter 'a' bevat. Sorteer de lijst op de naam van de stad en de naam van de staat. (37 records)
5. Toon alle orders van klanten uit de Verenigde Staten. (75 records)
6. Toon een alfabetische lijst van de klanten die iets gekocht hebben. Toon in een nieuw aantal veld hoeveel orders ze geplaatst hebben. (48 records, 1 klant 11 orders, 2 klanten 10 orders, ...)
7. Toon alle werknemers, gesorteerd op achternaam en het totaal van de orders die ze afgehandeld hebben. De orders moeten betaald zijn met Visa. (21 records)
8. Toon een lijst van alle klanten uit de Verenigde Staten met het totaal bedrag van de orders. Ook als er geen aankopen zijn moeten de klanten getoond worden. (27 records waarbij 4 klanten zonder aankopen)
9. Toon een lijst van alle orders voor klanten uit de Verenigde Staten die afgehandeld zijn door werknemers die meer dan 45 000 dollar verdienen. Sorteer de lijst zodat de grootste bedragen van orders bovenaan staan. (9 records)
10. Toon een lijst van alle klanten (55). Maak een nieuw resultaat veld OldCustomer. Dit heeft de waarde 'Y' als die klant na 1/1/1994 niets meer gekocht heeft. Bij de andere klanten zet je 'N'. (25 oude klanten)



## Oplossingen oefeningen

1. Toon een lijst van werknemers die na 1 januari 1992 zijn aangenomen en die ondertussen meer dan 35 000 dollar verdienen. (12 records)

```
SELECT a.*
FROM employee a
WHERE a.hiredate > '1/1/1992'
      AND a.salary > 35000
```

2. Hoeveel loonkosten zijn er voor alle werknemers (1386202.259 dollar)

```
SELECT sum(a.salary)
FROM employee a
```

3. Toon een lijst van alle klanten uit Orlando en Florida. Sorteer de lijst op de firmanaam. (9 records)

```
SELECT a.*
FROM customer a
WHERE a.country = 'US'
      AND a.state in ('OR', 'FL')
ORDER BY a.company
```

4. Toon een lijst van klanten waarbij de naam van de stad de letter 'a' bevat. Sorteer de lijst op de naam van de stad en de naam van de staat. (37 records)

```
SELECT a.*
FROM customer a
WHERE a.city like '%a%'
ORDER BY a.city, a.state
```

5. Toon alle orders van klanten uit de Verenigde Staten. (75 records)

```
SELECT a.custno, a.company, a.country, b.orderno, b.saledate
FROM customer a
      INNER JOIN orders b ON b.custno = a.custno
WHERE a.country = 'US'
```

6. Toon een alfabetische lijst van de klanten die iets gekocht hebben. Toon in een nieuw aantal veld hoeveel orders ze geplaatst hebben. (48 records, 1 klant 11 orders, 2 klanten 10 orders, ...)

```
SELECT a.custno, b.company, count(a.orderno)
FROM orders a
      INNER JOIN customer b ON b.custno = a.custno
GROUP BY a.custno, b.company
ORDER BY b.company
```

7. Toon alle werknemers, gesorteerd op achternaam en het totaal van de orders die ze afgehandeld hebben. De orders moeten betaald zijn met Visa. (21 records)

```
SELECT b.lastname, b.firstname, sum(a.amountpaid)
FROM orders a
     INNER JOIN employee b ON b.empno = a.empno
WHERE
     a.paymentmethod = 'Visa'
GROUP BY b.lastname, b.firstname
ORDER BY b.lastname, b.firstname
```

8. Toon een lijst van alle klanten uit de Verenigde Staten met het totaal bedrag van de orders. Ook als er geen aankopen zijn moeten de klanten getoond worden. (27 records waarbij 4 klanten zonder aankopen)

```
SELECT a.custno, a.company, sum(b.amountpaid)
FROM customer a
     LEFT JOIN orders b ON b.custno = a.custno
WHERE
     a.country = 'US'
GROUP BY a.custno, a.company
```

9. Toon een lijst van alle orders voor klanten uit de Verenigde Staten die afgehandeld zijn door werknemers die meer dan 45 000 dollar verdienen. Sorteer de lijst zodat de grootste bedragen van orders bovenaan staan. (9 records)

```
SELECT a.orderno, a.saledate, a.amountpaid, a.custno, a.empno, c.salary
FROM orders a
     INNER JOIN customer b ON b.custno = a.custno
     INNER JOIN employee c ON c.empno = a.empno
WHERE
     c.salary > 45000
     AND b.country = 'US'
ORDER BY a.amountpaid DESC
```

10. Toon een lijst van alle klanten (55). Maak een nieuw resultaat veld OldCustomer. Dit heeft de waarde 'Y' als die klant na 1/1/1994 niets meer gekocht heeft. Bij de andere klanten zet je 'N'. (25 oude klanten)

```
SELECT 'Y' AS OldCustomer, a.custno, a.company
FROM customer a
WHERE NOT EXISTS
     (SELECT 1 FROM orders b
      WHERE b.custno = a.custno
           AND b.saledate > '1/1/1994')
UNION
SELECT 'N' AS OldCustomer, b.custno, b.company
FROM customer b
     INNER JOIN orders c ON c.custno = b.custno
WHERE
     c.saledate > '1/1/1994'
```