**Opdrachten**

Geef een opdracht in SQL voor het opvragen van de volgende overzichten uit de WERKNEMERS tabel.

* 1. Alle namen van alle werknemers.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) naam FROM `werknemers`

* 1. Alle namen slechts éénmaal genoemd.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) DISTINCT Naam FROM `werknemers`

* 1. Alle combinaties van WOONPLAATS en POSTCODE éénmaal genoemd.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) Woonplaats, Postcode FROM `werknemers`

* 1. Alle salarissen.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) Salaris FROM `werknemers`

* 1. Alle salarissen slechts éénmaal genoemd.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) DISTINCT Salaris FROM `werknemers`

* 1. Alle waarden uit de DATUM\_IN\_D kolom

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) Datum\_in\_d FROM `werknemers`

### 1.3 Rekenen met kolominhouden

Het is ook mogelijk om berekeningen uit te voeren met te selecteren kolommen alvorens ze op het scherm af te drukken. We gebruiken daarvoor (in volgorde van prioriteit) de operatoren

\* en / vermenigvuldigen en delen

+ en - optellen en aftrekken

De prioriteit is uiteraard te beïnvloeden door gebruik te maken van ( ).

We mogen deze operatoren gebruiken om berekeningen uit te voeren met een kolom en een constante, twee kolommen en (hoewel dat niet zo erg nuttig is) twee constanten.

Als we naast het maandsalaris ook het jaarsalaris van de WERKNEMERS

willen zien, kunnen we dat opvragen door :

SELECT naam, salaris, salaris\*12

FROM werknemers

Dat geeft ons het volgende overzicht:

NAAM SALARIS SALARIS\*12

BUCK 4567.00 54804.00

BRASSER 2149.00 25788.00

LIEVENSE 3148.00 37776.00

VLIET 2638.00 31656.00

UMBGROVE 2988.00 35856.00

…..

Als we nog een tweede numerieke kolom in onze tabel zouden hebben (COMMISSIE bijvoorbeeld) zouden we een overzicht kunnen krijgen door

SELECT naam, salaris, commissie, salaris+commissie

FROM werknemers

We moeten hierbij wel waarschuwen voor het feit dat een optelling waar een NULL waarde bij betrokken is ook de waarde NULL oplevert. Immers: als we iets onbekends ergens bij optellen, kunnen we alleen maar aangeven dat dit als uitkomst iets op zal leveren dat ook onbekend is. Met andere woorden: alle WERKNEMERS zonder COMMISSIE krijgen de waarde NULL als uitkomst in de laatste kolom van bovenstaand overzicht (met een functie als NVL( <kolomnaam>,0 ) die iedere NULL waarde omzet in het getal 0, kunnen we dit in sommige RDBMS-en oplossen.)

Vaak leiden rekenkundige bewerkingen tot lelijke en lange kolomkoppen in

onze kolomkoppen overzichten. Dit kunnen we oplossen door in het SELECT gedeelte na de rekenkundige formulering een spatie, gevolgd door een alternatieve kolomkop op te geven.

SELECT NAAM, SALARIS PER\_MAAND, SALARIS \* 12 PER\_JAAR

FROM WERKNEMERS

Dit ziet er meteen al wat fraaier uit dan de formulering zonder alternatieve kolomkoppen.

NAAM PER\_MAAND PER\_JAAR

BUCK 4567.00 54804.00

BRASSER 2149.00 25788.00

LIEVENSE 3148.00 37776.00

VLIET 2638.00 31656.00

UMBGROVE 2988.00 35856.00

….

**Opdrachten**

Geef een opdracht in SQL voor het opvragen van de volgende overzichten uit de WERKNEMERS tabel.

* 1. De naam en het weeksalaris (salaris maal 3 gedeeld door 13) onder de kolomkop WEEKSAL van iedere werknemer.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) naam , salaris \* 3 / 13 weeksalaris FROM werknemers

* 1. De naam en het salaris inclusief een op handen zijnde salarisverhoging van 2% onder de kolomkop NWSAL.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) naam , Salaris \* 1.02 NWSAL FROM werknemers

## Operatoren in het WHERE gedeelte

Tot nu toe hebben we alleen nog maar gebruik gemaakt van het = teken om voorwaarden te stellen aan de te selecteren rijen. Uiteraard hebben we ook andere operatoren tot onze beschikking.

We kunnen gebruik maken van het < of het > teken om andersoortige voorwaarden te stellen. Bijvoorbeeld alle namen van werknemers met een salaris van meer dan 3000 Euro:

SELECT naam

FROM werknemers

WHERE salaris > 3000

Het < of > teken kunnen we ook gebruiken om twee kolommen met elkaar te vergelijken in plaats van het vergelijken van een kolom met een constante zoals in het voorgaande voorbeeld. Hoewel we niet kunnen zeggen dat dit nu zo'n voor de hand liggende vraag is, zouden we kunnen vragen om de namen van alle werknemers met een code die groter is dan de code van hun chef

SELECT naam

FROM werknemers

WHERE code > chef

Ondanks het feit dat deze opdracht in dit geval het juiste antwoord oplevert moeten we toch waarschuwen voor het gebruik van het < en het > teken bij alfanumerieke (CHAR) kolommen. Omdat de ASCII tabel (gelukkig) onderscheid maakt tussen hoofdletters en kleine letters, kan dit nog wel eens tot problemen leiden.

Bovendien wordt er nu lexicografisch vergeleken; dat wil zeggen dat eerst gekeken zal worden naar het eerste teken, als dit gelijk is wordt gekeken naar het volgende teken enzovoort. Dat is een prima uitgangspunt voor het indelen van bijvoorbeeld het telefoonboek maar het strookt niet met ons numerieke positiestelsel omdat daarbij de waarde 80 kleiner zal zijn dan 100 terwijl bij een lexicografische vergelijking we bij het eerste teken al constateren dat de 8 in de ASCII tabel een hogere waarde heeft dan de 1 en dat daarmee de waarde 80 dus groter is dan de waarde 100! Bij numerieke (NUMERIC) kolommen leveren deze operatoren vanzelfsprekend geen enkel probleem op.

Als we het < of > teken willen combineren met het = teken, doen we dit eenvoudigweg door ze achter elkaar te plaatsen

SELECT naam

FROM Werknemers

WHERE Salaris >= 3000

De volgorde van deze twee tekens staat vast, met andere woorden het is niet toegestaan om => te gebruiken in plaats van >= .

De 'ongelijk aan' operator wordt in de verschillende RDBMS-en op verschillende wijzen voorgesteld. De meest voorkomende vormen zijn < >, ! = en ^ = . In deze reader zullen we het ! = teken gebruiken. Zo kunnen we dus alle namen van werknemers die niet in ROTTERDAM wonen opvragen door

SELECT naam

FROM werknemers

WHERE woonplaats != 'ROTTERDAM'

**Opdrachten**

Geef een opdracht in SQL voor het opvragen van de volgende overzichten uit de WERKNEMERS tabel.

* 1. Alle namen en telefoonnummers van werknemers uit AMSTERDAM .

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) naam, Telefoon FROM werknemers WHERE woonplaats = 'AMSTERDAM'

* 1. Alle namen, adressen en woonplaatsen van werknemers die minder verdienen dan

2600 Euro.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) naam, Telefoon, Salaris FROM werknemers WHERE salaris >= 2600

* 1. Alle namen van werknemers die een naam hebben die in het telefoonboek na de

naam JANSEN komt.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) Naam, Telefoon FROM werknemers WHERE Naam > 'Jansen'

* 1. De woonplaatsen van alle werknemers die niet voor chef 40 werken.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) Naam, Woonplaats FROM werknemers WHERE Naam != Chef

* 1. De salarissen van alle werknemers die uit MAASTRICHT komen.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) Naam, Woonplaats, Salaris FROM werknemers WHERE Woonplaats =

"MAASTRICHT"

* 1. De namen en adressen van alle werknemers die na 1 januari 1988 in dienst zijn

gekomen.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) Naam, Woonplaats, Datum\_in\_d FROM werknemers WHERE Datum\_in\_d >=

"1 januari 1988"

* 1. De namen en 10% van het salaris (onder de kolomkop SAL\_DEEL) van alle

werknemers die niet in ROTTERDAM wonen.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) naam, Woonplaats, Salaris \* 1.10 SAL\_DEEL FROM werknemers

WHERE Woonplaats != "ROTTERDAM"

* 1. Alle woonplaatsen (zonder dubbelen) van werknemers die voor 31 december 1987

in dienst zijn gekomen.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) Woonplaats, Datum\_in\_d, [COUNT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/group-by-functions.html#function_count)(\*) as CNT FROM werknemers GROUP BY

Woonplaats, Datum\_in\_d ORDER BY `CNT` ASC

* 1. Alle namen en telefoonnummers van werknemers met een postcode boven 3088.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) Naam, Telefoon, Postcode FROM werknemers WHERE Postcode >= 3088

* 1. Alle namen van werknemers die niet meer dan 3500 Euro verdienen.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) Naam, Salaris FROM werknemers WHERE Salaris <= 3500

## Het combineren van operatoren met AND en OR

Vaak zullen we niet kunnen volstaan met het stellen van één voorwaarde maar zullen we meerdere voorwaarden willen combineren. In SQL kunnen we dan gebruik maken van AND en OR.

Zo zouden we ons af kunnen vragen welke werknemers uit ROTTERDAM aangenomen zijn voor 1 januari 1987.

SELECT naam

FROM werknemers

WHERE woonplaats = 'ROTTERDAM' AND

datum\_in\_d < '1987-01-01'

Dat is een geheel andere vraag dan de vraag welke werknemers uit ROTTERDAM komen OF voor 1 januari 1987 zijn aangenomen

SELECT naam

FROM werknemers

WHERE woonplaats = 'ROTTERDAM' OR

datum\_in\_d < '1987-01-01'

Bij het gecombineerd gebruiken van de AND en OR operatoren moeten we enige voorzichtigheid in acht nemen.

Stel dat we een overzicht willen van alle werknemers die voor 1 januari 1987 in dienst zijn gekomen en uit ROTTERDAM of uit UTRECHT komen, dan levert de opdracht

SELECT naam

FROM werknemers

WHERE datum\_in\_d < '1987-01-01' AND

woonplaats = 'ROTTERDAM' OR

woonplaats = 'UTRECHT'

ons niet het juiste antwoord op. Zo geformuleerd vragen we immers om de namen van alle werknemers die voor 1 januari 1987 in dienst zijn gekomen en in ROTTERDAM wonen, aangevuld met de namen van werknemers die in UTRECHT wonen.

We kunnen dit probleem op twee manieren oplossen

SELECT naam

FROM werknemers

WHERE datum\_in\_d < ‘1987-01-01' AND

woonplaats = 'ROTTERDAM' OR

datum\_in\_d < ‘1987-01-01' AND

woonplaats = 'UTRECHT'

of

SELECT naam

FROM werknemers

WHERE datum\_in\_d < ‘1987-01-01' AND

(woonplaats = 'ROTTERDAM' OR

woonplaats = 'UTRECHT')

In veel gevallen zullen we, vanwege de kortere formulering, voor de tweede oplossing kiezen: het gebruik van haakjes.

**Opdrachten**

Geef een opdracht in SQL voor het opvragen van de volgende overzichten uit de WERKNEMERS tabel.

* 1. De namen van alle werknemers die in UTRECHT wonen en een salaris hebben van

meer dan 3000 Euro.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) naam, Woonplaats, Salaris FROM werknemers WHERE woonplaats = "UTRECHT"

[AND](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/logical-operators.html#operator_and) Salaris >= 3000

* 1. De namen van alle werknemers met een postcode tussen

(inclusief de grens waarden) 3050 en 3100.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) naam, Woonplaats, Postcode FROM werknemers WHERE Postcode >= 3050

[AND](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/logical-operators.html#operator_and) Postcode <= 3100

* 1. De namen van alle werknemers met een salaris beneden de 2500 Euro of een salaris

boven de 4000 Euro.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) naam, Woonplaats, Salaris FROM werknemers

WHERE Salaris <= 2500 [OR](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/logical-operators.html#operator_or) Salaris >= 4000

* 1. De namen van alle werknemers met een salaris tussen de 2500 en 4000 Euro.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) naam, Woonplaats, Salaris FROM werknemers WHERE Salaris >= 2500

[AND](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/logical-operators.html#operator_and) Salaris <= 4000