**Informatica – SQL Samenvatting**

**SQL**

*Bekijken*

SELECT Kolom adres;

… DISTINCT Weergave adres; 🡨 verwijder dubbele rijen

… SUM Weergave SUM(kolom) 🡨 alle waardes bij elkaar opgeteld

… AS Weergave SUM(kolom) AS ‘naam’; 🡨 naam tabel

FROM Regel rekeninghouder;

WHERE Voorwaarde plaats = ‘Alkmaar’; 🡨 voorwaarde bij rijen

… AND Voorwaarde rekeninghouder = ‘Gerdien Jansen’;

… OR Voorwaarde leeftijd >= 50;

… LIKE Voorwaarde ‘\_K’; 🡨 mag elke letter of cijfer staan

… LIKE Voorwaarde ‘Ka%’; 🡨 begint met Ka

… NOT Voorwaarde ‘Alkmaar’; 🡨 behalve …

… IN Voorwaarde type = ‘BM’, ‘OV’, ‘PD’; 🡨 OR; OR;

ORDER BY Sorteer adres, huisnummer;

… ASC Sorteer adres ASC; 🡨 oplopend

… DESC Sorteer adres DESC; 🡨 aflopend

GROUP BY Sorteer plaats; 🡨 welke kolom

… HAVING Voorwaarde plaats = ‘Alkmaar’; 🡨 voorwaarde bij groepen

… COUNT Voorwaarde > 3; 🡨 met meer dan drie mensen

*Wijzigen*

INSERT INTO Toevoeging rekeninghouder (rekeninghouder\_id)

… VALUES Toevoeging (‘54’); 🡨 wat word toegevoegd

UPDATE Aanpassing rekeninghouder;

… SET Voorwaarde adres = ‘Kerkstraat 44’; 🡨 welke gegevens

DELETE FROM Verwijdering rekeninghouder;

… WHERE Voorwaarde ID = ‘111’ 🡨 welke rij

DROP TABLE Verwijdering rekeninghouder; 🡨 verwijder tabel

CREATE TABLE Toevoeging Naam (column1, column2, …)

Als er twee kolommen staan aangegeven bij ORDER BY, betekent dat dat hij eerst sorteert op de eerst aangegeven tabel, en als er meerdere gegevens zijn met de zelfde waarde in de tabel, sorteert hij op de tweede aangegeven tabel.

Met een asterisk (\*) selecteer je alle gegevens in een tabel.

= GELIJK AAN

> GROTER DAN

< KLEINER DAN

>= GROTER DAN OF GELIJK AAN

<= KLEINER DAN OF GELIJK AAN

<> NIET GELIJK AAN

… AS ‘’ Naam aan de tabel geven.

… COUNT () Hoeveelheid regels geteld.

… MAX () Grootste waarde in de kolom.

… MIN () Kleinste waarde in de kolom .

… AVG () Gemiddelde waarde in de kolom berekenen.

… SUM () Alle waardes in de kolom bij elkaar opgeteld.

*Data halen uit twee verschillende tabellen:*

WHERE kaart.IDnummer = transacties.IDnummer

*Subqueries:*

SELECT

FROM

WHERE = / IN

(SELECT

FROM

WHERE =

(SELECT

FROM

WHERE));

**NORMALISEREN**

Normaliseren is de dubbele gegevens uit een bron halen en in tabellen zetten.

0e normaalvorm

Alle gegevens noteren.

1e normaalvorm

De hoofdgroep een primaire index sleutel geven en herhalende gegevens samenvoegen in een aparte tabel met een aparte sleutel.

2e normaalvorm

De gegevens die niet afhankelijk zijn van de andere gegevens in de tabel, in een aparte tabel zetten.

**phpMyAdmin**

Tabel aanmaken:

|  |  |
| --- | --- |
| Naam | Naam van de kolom. |
| Type | * INT = getallen * VARCHAR = tekst   Bij lengte 🡪 aantal letters   * DECIMAL = decimaal getal   Bij lengte 🡪 3.2 = 000.00 |
| Lengte | Hoeveelheid tekens. |
| Index | * PRIMARY = hoofd ID van tabel * INDEX = ander ID van tabel |
| A\_I | Automatisch Optellen |

**BEGRIPPEN**

**Data** zijn gegevens. **Meta data** zijn gegevens over de data.

Een **database** is een verzameling gestructureerde gegevens die makkelijk beheerd kan worden.

**DBMS** = Data Base Management System (programma’s om database te beheren)

**RDBMS** = Relationeel Data Base Management System (gegevens in tabellen)

Bij **ongestructureerde data** staan de gegevens ongeordend.

Bij **gestructureerde data** staan de gegevens geordend.

**Redundantie** is als dezelfde gegevens meerdere keren voor komen.