



Syllabus

Databases

Mileto Di Marco
Kristien Roels
Bart Soete

Inhoudsopgave

| | |
|--|----------|
| H1 - Inleiding | 3 |
| 1 Algemeen | 3 |
| 1.1 Database | 3 |
| 1.2 Datamodel..... | 4 |
| 1.3 Database Management System (DBMS)..... | 4 |
| 1.4 Databasetaal..... | 4 |
| 2 Het relationele model | 5 |
| 2.1 Tabel, kolom en rij..... | 5 |
| 2.2 Integriteitsregels | 5 |
| 3 SQL..... | 6 |
| 3.1 Wat is SQL?..... | 6 |
| 3.2 De markt van de SQL-producten | 6 |
| 3.3 Informatie..... | 7 |

H1 - Inleiding

1 Algemeen

In deze module leer je hoe informatie (data) gestructureerd kan opgeslagen worden met een computersysteem.

Een databank of database kan je best vergelijken met de vroegere fichebak. Digitaal opslaan van gegevens heeft als voordeel dat we gebruik kunnen maken van de bijzondere rekenkracht van de computer:

- Sneller opzoeken van gegevens
- Sorteren en filteren volgens de gewenste criteria
- Uitvoeren van controles bij toevoegen, verwijderen of wijzigen van data (CRUD)
- Uitvoeren van geautomatiseerde processen bij datamanipulatie
- Efficiënte back-ups maken
- Beveiligen van data tegen ongeoorloofd gebruik

Veel databasesystemen bieden een grafische tool aan voor het bekijken en manipuleren van data. We hebben in deze opleiding als doel om later vanuit eigen applicaties data uit databases te lezen en aan te passen.

Om dit mogelijk te maken moeten we kunnen communiceren met een database vanuit onze eigen software. Dit wordt mogelijk gemaakt met de databasebevragingstalen. De bekendste en meest gebruikte taal hiervoor is **SQL**: Structured Query Language.

SQL is een databasetaal waarin opdrachten geformuleerd kunnen worden. De opdrachten kunnen dan door een Database Management Systeem (DBMS) verwerkt worden.

1.1 Database

Het woord database wordt voor verschillende begrippen gebruikt ¹:

- De opgeslagen gegevens als dusdanig (**database**)
- De wijze waarop de gegevens zijn opgeslagen (**datamodel**)
- De software waarmee databases kunnen worden aangemaakt en benaderd (Database Management Systeem of **DBMS**)

Een database, gegevensbank of databank is een digitaal opgeslagen archief ingericht met het oog op flexibele raadpleging en gebruik.

Databases spelen een belangrijke rol voor het archiveren en actueel houden van gegevens bij onder meer de overheid, financiële instellingen en bedrijven. Maar ook in de wetenschap worden databases veelvuldig gebruikt.

Een database moet aan volgende minimale (**CRUD**) voorwaarden voldoen om als een database gezien te worden:

1. Gegevens moeten eenvoudig kunnen worden opgeslagen (**Create**).
2. Gegevens moeten eenvoudig kunnen worden opgezocht en doorzocht (**Read**).
3. Gegevens moeten gewijzigd kunnen worden (**Update**).
4. Gegevens moeten verwijderd kunnen worden, zonder dat dat de werking van het systeem nadelig beïnvloedt (**Delete**).

¹ Bron Wikipedia

1.2 **Datamodel**

Met een **datamodel** (of gegevensmodel) wordt beschreven hoe de gegevens in een informatiesysteem gestructureerd zijn. We onderscheiden hierbij het conceptueel datamodel en het fysieke datamodel:

- **Conceptueel datamodel** (schema, gegevensmodel) :
Dit definieert welke gegevens vastgelegd kunnen worden, hoe de gegevens gestructureerd zijn en wat de verbanden zijn tussen de gegevens. Een conceptueel datamodel ligt steeds aan de basis van een in werkelijkheid gerealiseerde implementatie en is in feite het hoog-niveauontwerp van een (doorgaans relationele) databank.
- **Fysieke datamodel**
Dit beschrijft de manier waarop gegevens in een database zijn opgeslagen. Elke database heeft zijn eigen datamodel.

1.3 **Database Management System (DBMS)**

Een DBMS zal nooit uit zichzelf de gegevens in een database wijzigen of verwijderen. Iemand of iets (applicatie, programma) zal het DBMS hiertoe opdracht moeten geven.

Het DBMS waar we dit jaar kennis mee zullen maken is **MS SQL Server (Express)**.

Een belangrijke taak van het DBMS is het handhaven van de **integriteit** van de databasegegevens:

- Het DBMS moet zorgen dat de databasegegevens altijd voldoen aan de voorwaarden die in de werkelijkheid gelden.
 - ✓ Voorbeeld: als 1 werknemer maar voor 1 afdeling tegelijk mag werken, mag nergens in de database geregistreerd kunnen worden dat 1 werknemer in 2 of meer afdelingen werkt.
- Het DBMS moet zorgen dat 2 databasegegevens elkaar niet tegenspreken. Dit wordt ook wel consistentie van gegevens genoemd.

Elk DBMS kent instructies waarmee integriteitsregels gespecificeerd kunnen worden.

1.4 **Databasetaal**

Met behulp van speciale talen worden opdrachten aan een DBMS gegeven. Dit soort talen worden **databasetalen** genoemd.

Opdrachten worden door gebruikers ingevoerd en worden door het DBMS verwerkt. Elk DBMS, van welke fabrikant dan ook, bezit een databasetaal en sommigen hebben zelfs meer dan 1 databasetaal. Tussen al deze talen bestaan verschillen. Ze zijn onder te verdelen in groepen. Eén van deze groepen wordt gevormd door de relationele databasetalen.

De belangrijkste relationele databasetaal is **SQL** (Structured Query Language), deze wordt door nagenoeg alle DBMSen ondersteund.

Naast de standaard SQL breiden DBMS-fabrikanten doorgaans de taal nog uit met specifieke opdrachten (die dan doorgaans niet uitwisselbaar zijn tussen de verschillende DBMSen).

2 Het relationele model

2.1 Tabel, kolom en rij

Gegevens in een relationele database worden opgeslagen in tabellen. Tussen deze tabellen in een database kan je verwijzingen maken met behulp van relaties. Vandaar ook de term relationeel model.

Een tabel heeft 2 speciale eigenschappen:

- De kruising van een rij en een kolom mag maar 1 waarde bevatten.
- De rijen in een tabel hebben geen specifieke volgorde. Men kan dus niet spreken van een 1^{ste} rij, een laatste of een volgende rij. De inhoud van een tabel dient werkelijk beschouwd te worden als een verzameling rijen in de ware (wiskundige) zin van het woord.

2.2 Integriteitsregels

De inhoud van een tabel behoort meestal aan specifieke regels te voldoen, de zogenaamde integriteitsregels. In de voetbalwereld zouden 2 regels kunnen zijn dat het spelersnummer niet negatief mag zijn en er geen 2 spelers zijn die hetzelfde nummer hebben.

Integriteitsregels behoren door een relationeel database management systeem (RDBMS) gecontroleerd te worden. Telkens de inhoud van een tabel wordt aangepast, behoort het RDBMS te controleren of de nieuwe waarden aan de geldende integriteitsregels voldoen. Dit is duidelijk een taak van het RDBMS. De integriteitsregels moeten dan wel eerst gespecificeerd worden, zodat deze bekend zijn.

Integriteitsregels kunnen allerlei vormen hebben. Een aantal komt in de praktijk zo vaak voor dat ze specifieke namen hebben gekregen: primaire sleutel, kandidaatsleutel, alternatieve sleutel en refererende sleutel.

2.2.1 Primaire sleutel

De primaire sleutel van een tabel is een kolom (of een combinatie van een aantal kolommen) uit die tabel die kan gebruikt worden als unieke identificatie voor de rijen in die tabel.

2 verschillende rijen in een tabel mogen nooit dezelfde waarden hebben in de primaire sleutel en in elke rij moet de primaire sleutel altijd een waarde hebben.

De primaire sleutel wordt vaak aangeduid met de afkorting PK (primary key).

2.2.2 Kandidaatsleutel

Sommige tabellen bezitten meer dan 1 kolom (of een combinatie van kolommen) die aangewezen kunnen worden als de primaire sleutel. Ze bezitten allemaal de eigenschappen van een primaire sleutel. Deze kolommen (of combinatie) worden kandidaatsleutels genoemd. Slechts 1 daarvan wordt echter aangewezen als de primaire sleutel. Een tabel heeft dus altijd minimaal één kandidaatsleutel.

2.2.3 Alternatieve sleutel

Een kandidaatsleutel die niet de primaire sleutel van een tabel vormt, wordt een alternatieve sleutel genoemd. Men kan dus stellen dat voor een tabel nul of meer alternatieve sleutels kunnen gedefinieerd kunnen zijn. Het begrip kandidaatsleutel is dus de verzameling van alle primaire en alternatieve sleutels.

2.2.4 Refererende sleutel

Een refererende sleutel is een kolom (of een combinatie van kolommen) in een tabel waarvan de populatie een deelverzameling is van de populatie van de primaire sleutel van een (andere) tabel. Andere termen hiervoor zijn verwijzende sleutel, vreemde sleutel, foreign key (FK) of referential key.

Met de refererende sleutel kunnen we 'koppelingen' gaan maken tussen 2 of meerdere tabellen.

3 SQL

3.1 *Wat is SQL?*

SQL is een relationele databasetaal. De taal bevat onder andere instructies voor het invoeren, wijzigen, verwijderen, raadplegen en beveiligen van gegevens. We noemen SQL een relationele databasetaal, omdat het met gegevens omgaat volgens de regels gedefinieerd in het relationele model.

Omdat SQL een relationele databasetaal is, wordt zij geschaard onder de niet-procedurele databasetalen. Met niet-procedureel wordt bedoeld dat gebruikers met behulp van instructies slechts hoeven aan te geven met welke gegevens zij willen werken. Zij hoeven niet aan te geven hoe de gegevens één voor één dienen te worden benaderd.

SQL is op 2 manieren te gebruiken:

- Ten eerste kan het interactief worden gebruikt: een SQL-instructie wordt (in het correcte programma) ingetypt en direct verwerkt. Het resultaat is ook onmiddellijk zichtbaar. We zullen deze manier gebruiken om de basisprincipes van SQL aan te leren. Interactieve SQL is bedoeld voor de ontwikkelaars van applicaties en voor eindgebruikers die zelf databases willen benaderen.
- De tweede gebruikswijze wordt embedded SQL genoemd. SQL-instructies zijn dan opgenomen (embedded) in een programma dat in een andere programmeertaal geschreven is. Resultaten van SQL-instructies zijn in dit geval niet direct zichtbaar voor de gebruiker, maar worden door het omhullende programma bewerkt. Embedded SQL wordt vooral gebruikt in programma's die voor eindgebruikers ontwikkeld zijn. Deze eindgebruikers hoeven dan geen SQL-instructies te leren, maar krijgen eenvoudige menu's en schermen te zien. In de lessen C# en ASP.net worden van deze embedded mogelijkheden gebruik gemaakt.

3.2 *De markt van de SQL-producten*

De taal SQL is in zeer vele producten op één of andere wijze geïmplementeerd. SQL-producten zijn verkrijgbaar voor alle besturingssystemen en op alle soorten machines, van de kleinste PC tot het grootste mainframe.

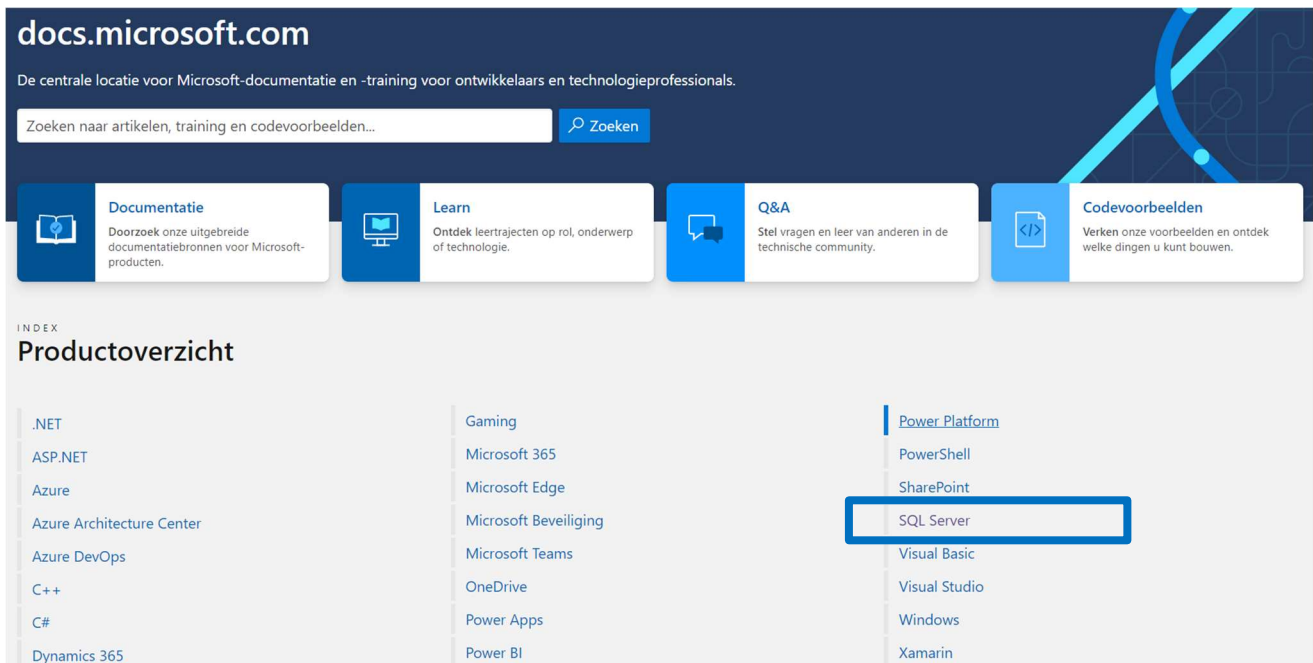
Hieronder een kort overzicht van de meest gebruikte SQL-producten :

- **Microsoft SQL Server**
- MySQL (open source)
- Oracle (Oracle)
- Cache (Intersystems)
- DB2 (IBM)
- PostgreSQL (open source)
- ...

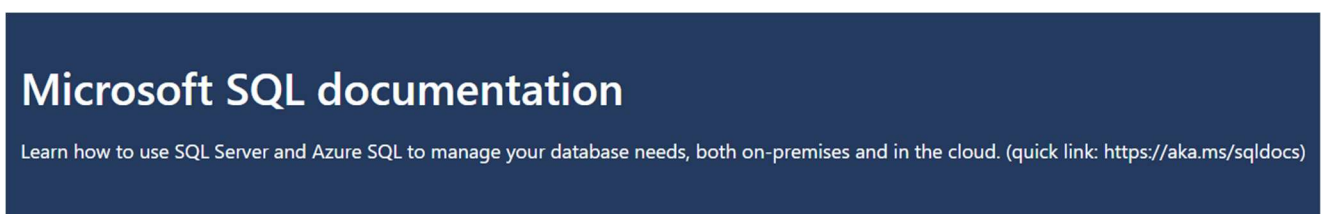
Bij zowat alle moderne programmeertalen wordt de mogelijkheid geboden SQL-instructies op te nemen in de programma's. De systemen werken transparant d.w.z. onafhankelijk van de database die gebruikt wordt. Als je bijvoorbeeld een C# programma maakt, dan kan je SQL-instructies lanceren vanuit dit programma op databases van om het even welk type (MDF, DBF, ODBC, ...). Dit heeft als voornaamste voordeel dat, wanneer het databasesysteem vervangen wordt, er aan de code van de programma's niets meer hoeft gewijzigd te worden.

3.3 Informatie

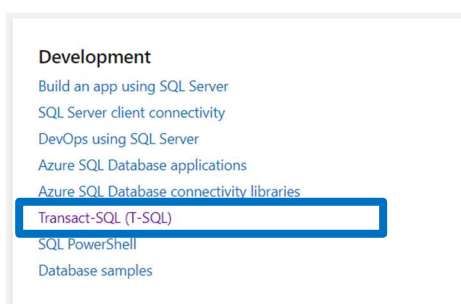
- Microsoft Docs is de belangrijkste informatiebron: <https://docs.microsoft.com/nl-nl/>



Na het kiezen van SQL Server: <https://docs.microsoft.com/nl-be/sql/?view=sql-server-ver15>



Kies hier bijvoorbeeld voor Development – Transact-SQL (T-SQL): <https://docs.microsoft.com/nl-nl/sql/t-sql/language-reference?view=sql-server-ver15>



Tip: als de cursor op een SQL-keyword in de Query Editor van de Management Studio staat, druk dan de **F1**-toets en help over het betreffende keyword verschijnt online in de Microsoft Docs.

- W3schools bevat een SQL Tutorial en een SQL Reference: <https://www.w3schools.com/>



- Naast de bovenstaande documentatie is er nog veel online cursusmateriaal te vinden. O.a.:
 - SQL Server tutorial for beginners (149 video's):
<https://www.youtube.com/playlist?list=PL08903FB7ACA1C2FB>
 - SQL Server Programming (18 video's):
<https://www.youtube.com/playlist?list=PLNI5-AWhQzcleQWADpUgriRxebMkMmi4H>
- Wikipedia-pagina over SQL: <http://en.wikipedia.org/wiki/SQL>

