



KATHOLIEKE UNIVERSITEIT
LEUVEN



Databanken

Databanken



Een database is een verzameling gegevens, opgeslagen in de computer, die onderling geïntegreerd zijn en een repeterende structuur vertonen, die op haar beurt past binnen een bepaald datamodel, waardoor elke bewerking van de database (bevraging, toevoeging, verwijdering, enz.) uiteindelijk verricht kan worden met behulp van één of meer basisbewerkingen, die altijd worden uitgevoerd door het database managementsysteem (DBMS).



Enkele begrippen



- **databestand**: een gestructureerde verzameling gegevens die in de computer is opgeslagen;
- **database**: een geïntegreerde verzameling van één of meer databestanden, m.a.w. de verzameling van in een computer (in databestanden of database-files) opgeslagen gegevens;
- **database management systeem** (DBMS): specifieke software die gebruikt wordt om het geheel van gegevens dat de database vormt, te beheren (d.w.z. nieuwe gegevens toevoegen, bestaande wijzigen, zichtbaar maken of verwijderen) en te bevragen;
- **datamodel**: het basisschema dat binnen een bepaald DBMS beschikbaar is voor de structurering van de database; het datamodel zal de toegang bepalen tot de gegevens.
- een **database systeem** ontstaat wanneer database, DBMS en hardware op de juiste manier op elkaar zijn afgestemd.

Relationele database



Een relationeel databasesysteem beschouwt elke manipulatie van gegevens in een database als een functie van tabellen naar tabellen. Tussen de tabellen onderling worden relaties vastgelegd, die de toegang tot de gegevens bepalen. Dit gebeurt aan de hand van *sleutelvelden*. Op die manier komt men tot het relationele datamodel.



Kernbegrippen



- **tabel:** een databestand waarin de gegevens zich tot elkaar verhouden in rijen en kolommen.
- **rij:** bevat de gegevens die bij eenzelfde object horen;
- **kolom:** gegevens die in eenzelfde kolom staan hebben betrekking op eenzelfde eigenschap van de beschreven objecten.
- **veld:** de kruispunten tussen rijen en kolommen noemt men velden, de kleinste informatieëenheden van de database.



Elementen van een database



- Tabellen (*tables, files*)
Daarin zitten de gegevens opgeslagen; samen met hun bijhorende indexbestanden maken de tabellen de kern van de database uit;
- Virtuele tabellen (*views, queries*)
Tabellen die niet als zodanig op schijf bewaard worden, maar alleen een bepaalde manier zijn om de echte tabellen te bekijken;
- Invoerschermen (*forms*)
Vaak gebaseerd op views; forms zorgen voor een gecontroleerde data-acquisitie, en zijn toegesneden op de taken van de gegevensinvoerder.
- Rapporten (*reports*)
Programmacode om gestructureerde rapporten te genereren op basis van de in de tabellen opgeslagen informatie;
- Toepassingen (*applications*)
Een geheel van programma's die samen een afgesloten toegang tot een database vormen. Een toepassing zorgt ervoor dat de tabellen alleen op een geautoriseerde manier kunnen benaderd worden, zodat de integriteit van de gegevens bewaard blijft bij de dagelijkse bewerking.

Datatypes

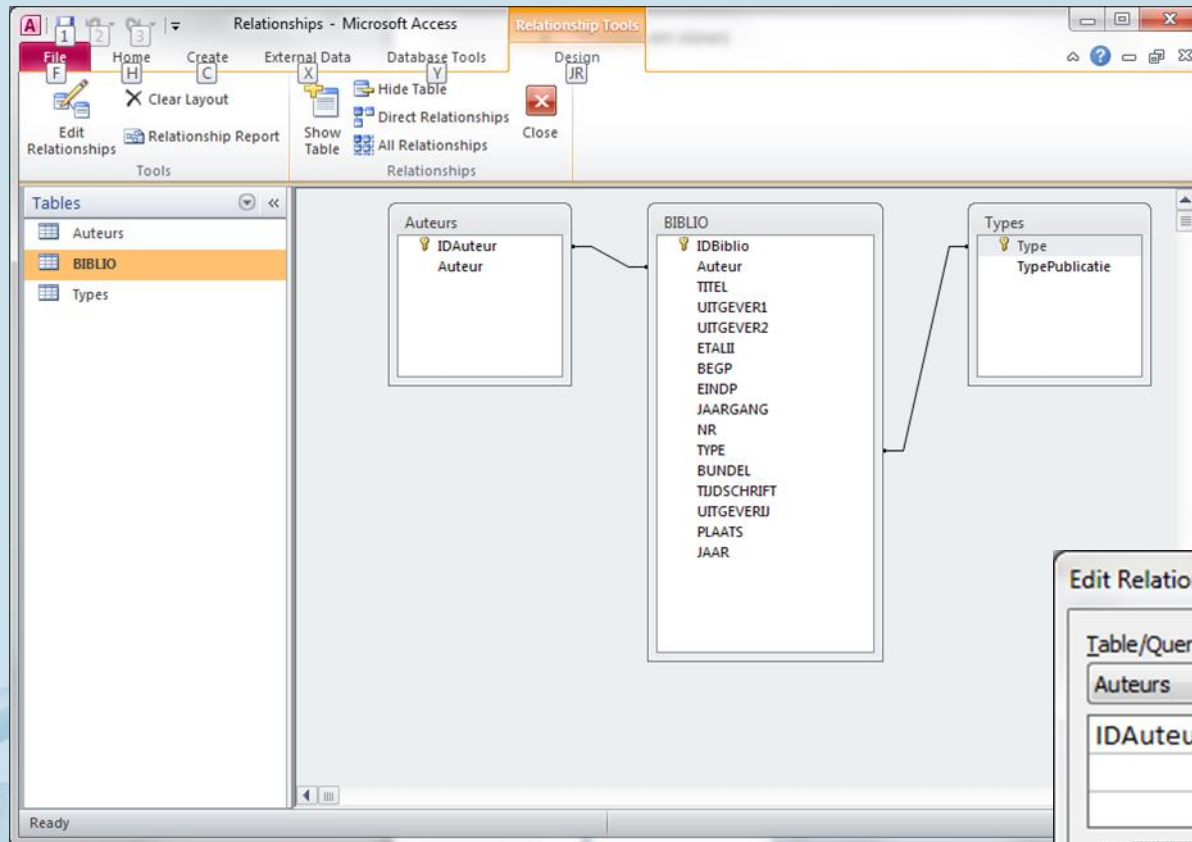


- Korte Tekst
- Lange Tekst
- Numeriek
 - Float
 - Integer
 - Autonumber
- Datum/Tijd
- Currency
- Logical (Yes/No)
- Binary

- Elke tabel moet minstens met één andere tabel verbonden zijn
- Elke tabel moet een unieke sleutel hebben, de *primary key*
- Sleutels van andere tabellen opgenomen in een tabel noemt men *secondary keys*
- Tussen de *primary key* in de moedertabel en de *secondary keys* in de doeltabellen bestaat een *one-to-many* relatie



One-to-many relations



The Edit Relationships dialog box shows the relationship between Auteurs and BIBLIO. The primary key is IDAuteur in Auteurs, and the related field is Auteur in BIBLIO. The relationship type is One-To-Many. The Enforce Referential Integrity checkbox is checked.

Table/Query:	Related Table/Query:
Auteurs	BIBLIO
IDAuteur	Auteur

☒ Enforce Referential Integrity
☐ Cascade Update Related Fields
☐ Cascade Delete Related Records

Relationship Type: One-To-Many

Buttons: OK, Cancel, Join Type..., Create New...

- **Eerste normaalvorm**
- In deze eerste normalisatiestap zullen we de *repeating groups* verwijderen. Dat doen we eenvoudig door van de repeating groups nieuwe objecten te maken. In dat nieuwe object kopiëren we de sleutel van het hoofdobject, de tijdschriftfiche, en geven we het object ook een eigen sleutel.
- Het voorbeeld is relatief complex, omdat er meerdere repeating groups te onderscheiden zijn. Om dat te verduidelijken zetten we één en ander in kolommen naast elkaar:

Artikel in Tijdschrift



TIJDSCHRIFT	TIJDSCHRIFTUITGAVE	ARTIKEL
titel	uitgavenr	artikeltitel
uitgeverij	jaargang	artikelauteur
plaats	aantalpaginas	beginpagina
ISSN		eindpagina

Eén tijdschrift kan dus meerdere uitgaven bevatten en elke uitgave kan meerdere artikels bevatten.

TIJDSCHRIFTUITGAVE (uitgaveID, tijdschriftID, uitgavenr, jaargang, aantalpaginas, RG(artikeltitel, artikelauteur, beginpagina, eindpagina))

De repeating group wordt dan uit de tijdschriftfiche geschrapt:

TIJDSCHRIFT (tijdschriftID, titel, uitgeverij, plaats, ISSN)

Tweede normaalvorm



- In deze tweede stap interesseren we ons in de objecten met samengestelde sleutel, m.a.w. de objecten die we in de eerste normaalvorm hebben bijgemaakt. Hier gaan we na of er eigenschappen zijn die in groep *covariëren* met elkaar, maar niet met de gehele sleutel. Dat zijn eigenschappen die van elkaar afhankelijk zijn. In dit geval is het duidelijk dat we de artikels als nieuw object kunnen afzonderen.
- ARTIKEL (artikelID, uitgaveID, artikeltitel, artikelauteur, beginpagina, eindpagina)
- De repeating group wordt dan uit de uitgavefiche geschrapt:
- TIJDSCHRIFTUITGAVE (uitgaveID, tijdschriftID, uitgavenr, jaargang, aantalpaginas)

Derde Normaalvorm



- Voor de derde normaalvorm bekijken we niet-sleutel gegevens. Daar zien we b.v. dat artikelauteur een mogelijke Repeating Group is. Bovendien kan het wenselijk zijn auteurs éénduidig te identificeren. Om dat te bereiken zullen we een bijkomende tabel maken met auteursgegevens:
- ARTIKELAUTEUR (artikelauteurID, artikelID, artikelauteur)
- ARTIKEL (artikelID, uitgaveID, artikeltitel, beginpagina, eindpagina)





Structured Query Language



- Een standaard taal om databanken op een uniforme manier te benaderen
 - Data Definition Language
 - Om databanken te definiëren
 - Data Manipulation Language
 - Om databanken te gebruiken



SQL Voorbeeld



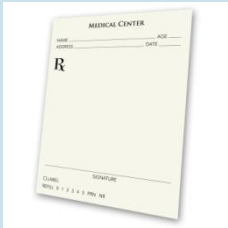
- SELECT QUERY
 - SELECT ID, FirstName, LastName
FROM Subscribers
WHERE Year = 2009 AND SubscriptionPaid= Yes
ORDER BY LastName, FirstName;



- INSERT STATEMENT
 - INSERT INTO Subscribers (FirstName, LastName) VALUES ("Jan", "Janssens");
- DELETE STATEMENT
 - DELETE FROM Subscribers WHERE ID = 314;
- UPDATE STATEMENT
 - UPDATE Productcatalog
Set Price = Price*1,05
WHERE ProductType = "DVD";

Database opzet

WebForm POST statement



cookie



webserver

SQL Query



database
server

Retrieved records



Dynamic web page
= template + database fields

session

Persoonlijke databases



- Generic
 - Microsoft Access
 - Filemaker
 - OpenOffice
- Specific databases:
 - Photo
 - Flickr, Picasa, Mediadex, ...
 - Music, video etc...
 - iTunes, Youtube, ...
 - Bibliography
 - EndNote, Reference Manager, Zotero, Refworks, ...
 - Books
 - The Library Thing
 - Readerware
 - BookCat