

Database 1

1 Wat is een database?

Leerdoelen

- Je weet wat een database is en waarvoor die gebruikt wordt.
- Je kunt gegevens uit de echte wereld omzetten naar tabellen en kolommen.
- Je begrijpt het verschil tussen ruwe gegevens en een gestructureerd model.

Uitleg

In het dagelijks leven slaan organisaties gegevens op: over klanten, producten, studenten, boeken, etc. Een **database** is een digitale plek waar zulke gegevens netjes georganiseerd worden bewaard. Je verdeelt de informatie over verschillende tabellen, waarbij elke tabel over één onderwerp gaat, zoals:

- **Studenten:** naam, klas, geboortedatum
- **Docenten:** naam, vak, afdeling
- **Opleidingen:** naam, niveau, duur

Een database lijkt een beetje op een Excel-bestand, maar is veel krachtiger en beter gestructureerd. Je wilt geen dubbele gegevens en alles moet logisch met elkaar verbonden zijn.

Voorbeeld: Dit is géén goede database:

Naam	Klas	Opleiding
-----	-----	-----
Fatima	SD1A	Software Developer
Ali	SD1A	Software Developer
Robin	SD2B	Software Developer
Jente	SD1A	Software Developer

→ Hier zie je dat de opleiding meerdere keren herhaald wordt. Dat is zonde en foutgevoelig.

☐ Wat zou beter zijn?

Je zou in plaats daarvan een aparte tabel 'Studenten' maken en een aparte tabel 'Opleidingen'. Studenten krijgen dan een *verwijzing* naar hun opleiding (dat komt in de volgende lessen aan bod).

☐ Opdracht 1 – Gegevens analyseren

1. Bekijk onderstaande lijst met gegevens:

Naam: Esra
Klas: SD1A
Opleiding: Software Developer
Docent: De Jong
Vak: Webontwikkeling
Lesdag: Woensdag

2. Welke verschillende **onderwerpen** zie je hierin? Probeer per onderwerp de eigenschappen op te schrijven.
 - **Student:** naam, klas
 - ... (jij vult aan)
3. Zet je antwoorden in een tabelvorm: welke tabellen zou je nodig hebben? Welke kolommen zouden erin staan?

☐ Reflectie

- Waarom is het niet handig om alle informatie in één grote tabel te zetten?
- Wat denk je dat het voordeel is van losse tabellen met verbindingen?

☐ Inleveren

- Lever je tabellenindeling in (.txt, .pdf of screenshot).

- Voeg je antwoorden toe op de reflectievragen.

2 Entiteiten en Attributen

Leerdoelen

- Je weet wat een entiteit is en wat een attribuut is.
- Je kunt entiteiten en hun attributen herkennen in een realistisch scenario.
- Je kunt een eerste versie van een ERD tekenen met entiteiten en attributen.

Uitleg

Entiteit

Een **entiteit** is een "*ding*", "*persoon*" of "*gebeurtenis*" in de echte wereld waarover je gegevens wilt opslaan. Bijvoorbeeld:

- Student
- Cursus
- Docent
- Opleiding

Attribuut

Een **attribuut** is een eigenschap van een entiteit. Bijvoorbeeld:

- De entiteit Student heeft de attributen: studentnummer, voornaam, achternaam, telefoonnummer.
- De entiteit Cursus heeft de attributen: naam, duur, startdatum.

Voorbeeld

image.png and or type unknown

☐☐ In een ERD (Entity Relationship Diagram) teken je entiteiten als rechthoeken en attributen als ovale of gelabelde velden ernaast.

In een ERD heb je **drie kolommen** de eerste voor de **keys**, de tweede voor de **attribuutnamen** en de derde voor de **datatypes**.

☐☐ Opdracht 1 – Entiteiten en attributen

1. Je werkt voor een evenementenbureau. Zij willen bijhouden:
 - Welke klanten boekingen doen
 - Welke evenementen er zijn
 - Welke locaties beschikbaar zijn
2. Maak een lijstje van minstens 3 entiteiten uit deze situatie. Bijvoorbeeld:
 - Klant
 - ...
 - ...
3. Geef per entiteit minstens 3 bijpassende attributen. Bijv.:
 - Klant → voornaam, e-mailadres, ...
 - Locatie → naam,
4. Teken jouw eerste ERD in [Lucidchart](#).

Lucichart

Registreer je voor Lucichart en maak een gratis account.

Kies toevoegen library

image.png
Image not found or type unknown

Zoek naar ERD en selecteer "Entity Relationship"

image.png
Image not found or type unknown

Gebruik vervolgens dit figuur om een entiteit te maken.

image.png
Image not found or type unknown

☐☐ Reflectie

- Hoe wist je of iets een entiteit was of gewoon een attribuut?
- Heb je misschien dingen dubbel in verschillende entiteiten? Kun je iets beter loskoppelen?

☐☐ Inleveren

- Lever je ERD in als PDF, afbeelding of link (als je Lucidchart gebruikt).
- Lever ook je antwoorden op de reflectievragen in (PDF, TXT).

3 Primary Keys

☐☐ Leerdoelen

- Je weet wat een **primary key** is.
- Je begrijpt waarom een primary key verplicht is in elke tabel.
- Je kunt per entiteit een geschikte primary key kiezen.

☐☐ Uitleg

☐☐ Wat is een Primary Key (PK)?

Een primary key is een uniek gegeven waarmee je één rij uit een tabel kunt identificeren. Elke tabel in een database **moet** een primary key hebben.

Regel: Elke entiteit heeft precies één PK. Heb je meerdere opties? Kies er één. Heb je geen goede kandidaat? Gebruik dan een kunstmatige sleutel, zoals een `id`.

Voorbeelden:

- `Student` → `student_id`
- `Evenement` → `evenement_id`

- Locatie → locatie_id

Een primary key moet:

- Uniek zijn
- Nooit leeg zijn
- Vast blijven (mag niet wijzigen)

Als je later tabellen met elkaar verbindt, gebruik je de PK van de ene tabel als verwijzing in de andere – dat wordt dan een **foreign key**.

▣ Voorbeeld van een ERD met PK

image.png
image not found or type unknown

▣ Opdracht 1 – Kies je primary keys

1. Gebruik het ERD van jouw scenario (bijv. evenement, klant, locatie).
2. Voeg aan elke entiteit een geschikte primary key toe. Noteer die als `entiteit_id` of kies iets wat uniek is.
3. Zet in de eerste kolom van je ERD de letters PK om aan te geven dat deze regel de PK bevat.
4. Controleer: is jouw PK echt uniek en onveranderlijk?

▣ Reflectie

- Welke van je gekozen PK's is natuurlijk (bestaand gegeven), en welke kunstmatig (gegenereerd ID)?
- Wat zou er misgaan als je geen PK kiest?

▣ Inleveren

- Lever je ERD in met onderstreepte primary keys (PDF, foto of Lucidchart link).
- Voeg een korte toelichting toe bij je keuze(s) voor PK.

4 1:N-relaties en Foreign Keys

Leerdoelen

- Je weet wat een 1:N-relatie is in een database.
- Je begrijpt wat een foreign key (FK) is en waarvoor die dient.
- Je kunt zelf een 1:N-relatie modelleren met een foreign key.

Uitleg

Wat is een 1:N-relatie?

Bij een **één-op-veel-relatie** (1:N) hoort bij één rij in de ene tabel, meerdere rijen in de andere tabel.

Bijvoorbeeld:

- 1 Klant → veel Boeking(en)
- 1 Docent → veel Cursussen

De **“meer”-kant** krijgt de foreign key (FK). De FK is een kopie van de primary key (PK) van de andere tabel.

De regel **Meer = harkje = FK**

- Teken het harkje aan de kant waar “veel” is
- In die tabel voeg je de foreign key toe
- Voorbeeld: `boekingen` krijgt `klant_id` als FK

Voorbeeld

image.png and or type unknown

Het `klantnummer` in `boeking` is een *foreign key* die verwijst naar `klant`.

Opdracht 1 – Relaties en foreign keys

1. Bekijk jouw ERD van de vorige opdrachten.
2. Bepaal minstens twee 1:N-relaties. Bijv.:
 - 1 Locatie → Veel Evenementen
 - ...
3. Voeg aan de ‘meer’-entiteit een foreign key toe (bijv. `klant_id` in `boeking`).
4. Teken deze relaties met een harkje en schrijf de FK er duidelijk bij.

Reflectie

- Waarom voeg je de FK toe aan de “meer”-kant?
- Wat zou er gebeuren als je die FK vergeet?

Inleveren

- Lever je bijgewerkte ERD in (PDF, foto of Lucidchart-link), inclusief foreign keys.
- Beantwoord de reflectievragen.

5 Datatypes en Validatie

Leerdoelen

- Je kent de meest gebruikte datatypes in een database.
- Je kunt passende datatypes kiezen voor attributen in je ERD.
- Je weet waarom validatie belangrijk is bij het kiezen van datatypes.

Uitleg

Wat is een datatype?

Een datatype bepaalt welk soort informatie je in een kolom/tabel opslaat. Het zorgt ervoor dat je database weet hoe de data eruitziet en wat ermee mag gebeuren.

Veelgebruikte datatypes:

Datatype	Gebruik	Voorbeeld
INT	Voor hele getallen (bijv. ID's, aantallen)	5, 142
VARCHAR(50)	Voor tekst tot X karakters	'Jan', 'email@example.com'
DATE	Voor datums zonder tijd	'2025-06-06'
DATETIME	Voor datum én tijd	'2025-06-06 14:30:00'
DECIMAL(6,2)	Voor getallen met komma (bijv. prijs)	12.99, 9999.00

Voorbeeld

image.png
image not found or type unknown

Tips voor goede keuzes:

- Kies niet zomaar `TEXT` als alternatief voor `varchar()`; `TEXT` maakt je database trager.
- Let op lengte bij `VARCHAR`: hoe langer, hoe trager.
- Gebruik `DECIMAL` bij geldbedragen, niet `FLOAT`.
- Een telefoonnummer is een `VARCHAR()` en geen `INT`, waarom?

Opdracht 1 – Pas je datatypes aan

1. Bekijk jouw ERD's (minstens 2).
2. Kies voor elk attribuut een passend datatype uit de tabel hierboven.
3. Schrijf de datatypes in de derde kolom van de ERD's..
4. Gebruik minimaal 3 verschillende soorten datatypes; je mag er zelf attributen bij bedenken.

Inleveren

- Lever je bijgewerkte ERD in met daarin duidelijk per attribuut het datatype genoteerd en de FK.

6 Case – Modelleer een realistisch scenario

Leerdoelen

- Je kunt zelfstandig entiteiten en relaties herkennen in een realistisch scenario.
- Je past de 5 stappen toe om een ERD te maken.
- Je maakt een logisch en technisch correct ERD met PK's, FK's en datatypes.

Uitleg

Tot nu toe heb je geleerd wat entiteiten, attributen, PK's, FK's, datatypes en 1:N-relaties zijn. Nu pas je alles toe op een echte situatie.

Scenario: Cursusadministratie

Een opleidingscentrum biedt cursussen aan. Klanten kunnen zich inschrijven op een cursus. Elke cursus heeft een vaste docent. Klanten mogen zich inschrijven op meerdere cursussen. Een klant kan meerdere inschrijvingen doen.

Beschikbare info:

- **Klant:** naam, e-mail, geboortedatum
- **Cursus:** titel, startdatum, prijs
- **Docent:** naam, specialisatie
- **Inschrijving:** datum van inschrijving

Wat verwacht je in je ERD?

- Entiteiten: klant, cursus, docent, inschrijving

- Relaties:
 - 1 klant ↔ veel inschrijvingen
 - 1 cursus ↔ veel inschrijvingen
 - 1 docent ↔ veel cursussen

☐☐ Opdracht – Maak het ERD

1. Teken een ERD in Lucidchart of op papier.
2. Volg de 5 stappen:
 1. Bepaal de entiteiten en hun attributen
 2. Bepaal de primary keys
 3. Bepaal per attribuut een datatype
 4. Leg de relaties vast met 1:N en harkjes
 5. Voeg de juiste foreign keys toe
3. Check: heb je alle FK's benoemd?
4. Check: zijn alle attributen voorzien van een datatype?

☐☐ Reflectie

- Welke keuzes heb je gemaakt bij het modelleren?
- Welke relaties waren makkelijk/moeilijk te bepalen?
- Wat zou je kunnen verbeteren aan je ERD?

☐☐ Inleveren

- Lever je ERD in als PDF of als afbeelding (scan/foto van papier of export uit Lucidchart).
- Beantwoord de reflectievragen in een los tekstbestand (.txt of .docx).

7 Meerdere relaties en N:N

Leerdoelen

- Je begrijpt wat een N:N-relatie is en wanneer die voorkomt.
- Je kunt een N:N-relatie correct omzetten naar aparte tabellen met FK's.
- Je kunt meerdere relaties per entiteit modelleren in een ERD.

Uitleg

Wat is een N:N-relatie?

Een **veel-op-veel** (N:N) relatie komt voor wanneer meerdere records uit entiteit A gekoppeld kunnen zijn aan meerdere records uit entiteit B.

Voorbeeld: Studenten kunnen zich inschrijven voor meerdere vakken. Elk vak kan meerdere studenten hebben.

Hoe modelleer je dit?

Je maakt een extra tabel tussen de twee entiteiten. Deze bevat alleen de foreign keys van beide kanten.

Student ← FK → Inschrijving ← FK → Vak

De tussentabel bevat meestal ook extra informatie, zoals de inschrijfdatum of het cijfer.

Meerdere relaties op één entiteit?

Soms is een entiteit aan meerdere andere entiteiten verbonden.

Voorbeeld: Een **medewerker** werkt in een afdeling, maar ook aan meerdere projecten.

- 1 medewerker ↔ 1 afdeling → 1:N
- 1 medewerker ↔ meerdere projecten ↔ N:N

Gebruik verschillende relaties als het logisch is dat een entiteit meerdere rollen vervult.

Opdracht – N:N-model

1. Maak een ERD voor dit scenario:
Een muziekschool organiseert **lessen**. **Leerlingen** kunnen zich inschrijven op meerdere lessen. Elke **les** wordt gevolgd door meerdere leerlingen.
2. Modelleer de juiste entiteiten, attributen, PK's, FK's en datatypes.
3. Voeg een tussentabel toe om de N:N-relatie correct te verwerken.

☐☐ Reflectie

- Waarom is een tussentabel nodig bij N:N?
- Welke extra informatie kun je in de tussentabel kwijt?
- Waar moet je op letten bij het toevoegen van meerdere relaties in je ERD?

☐☐ Inleveren

- Lever je ERD met N:N-relatie in (PDF of afbeelding).
- Beantwoord de reflectievragen in een tekstbestand.

--

Revision #6

Created 6 June 2025 13:55:57 by Max

Updated 6 June 2025 20:55:19 by Max