



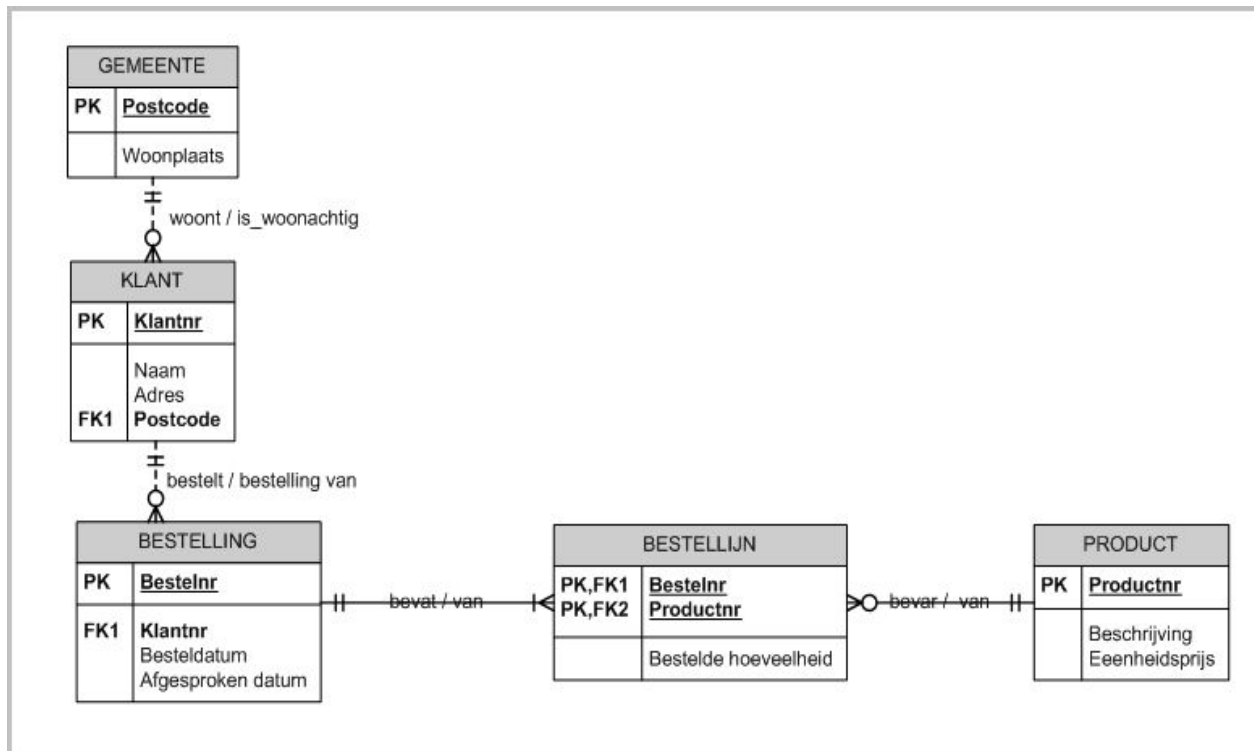
2 Relationeel datamodel - Ontwerp

2.1 Normaliseren

2.2 Entity-relationship diagram. (ERD)

2.1 Normaliseren

- Hoe komen we toe een genormaliseerde dB dat consistent is? Welke gegevens in welke tabel, hoe groeperen....?



2.1 Normaliseren

- Gegevensverzameling verdelen in een aantal **groepen**. Iedere groep voldoet aan de volgende definitie:

Alle attributen van de groep zijn functioneel afhankelijk van de volledige sleutel en tussen attributen onderling zijn geen functionele afhankelijkheden aanwezig.

- → genormaliseerde groepen
- Vb. Naam, adres en postcode is **volledig** afhankelijk van klantr.

KLANT	
PK	<u>Klantr</u>
	Naam Adres Postcode
FK1	

2.1 Normaliseren

- Genormaliseerde groep: bevat vast aantal gegevens in tabel
Alle attributen zijn functioneel afhankelijk van Nummer (sleutel) en tussen attributen (Naam – Woonplaats -Afdeling) zijn onderling geen functionele afhankelijkheden aanwezig.

PK	Medewerkers			
	<i>Nummer</i>	<i>Naam</i>	<i>Woonplaats</i>	<i>Afdeling</i>
	001	Els	Antwerpen	Opleidingen
	002	Pieter	Leuven	Opleidingen
	003	Patricia	Tongeren	Analyse
	004	Rudi	Genk	Analyse
	005	Gert	Hasselt	Programmering

2.1 Normaliseren

- Groep medewerkers: 'Nummer' → sleutel
- Alle attributen: Naam, Woonplaats en afdeling afhankelijk van het Nummer
- Geen functionele afhankelijkheden aanwezig tussen de attributen → **genormaliseerde groep**

Medewerker(Nummer, Naam, Woonplaats, Afdeling)

2.1 Normaliseren



□ Normaliseren vindt plaats in drie stappen:

1. Verwijder de zich herhalende deelverzameling
2. Verwijder de attributen die functioneel afhankelijk zijn van slechts een deel van de sleutel
4. Verwijder de attributen die ook functioneel afhankelijk zijn van ander (niet-sleutel) attributen – transitieve afhankelijkheden

2.1 Normaliseren



Opmerking: VERWIJDEREN

- Het verwijderen uit de oorspronkelijke groep (opsplitsen)
- Maar het tegelijkertijd creëren van een nieuwe groep

Er mag dus niets echt verwijderd worden!!!!!!

2.1 Normaliseren

□ Voorbeeld: projectoverzicht:

Project	Budget	Medewerker	Afdeling	Chef	Uren
001 VDU	1000	003 Patricia	Analyse	Johan	60
		005 Gert	Programmering	Ron	100
002 CRT	800	004 Rudi	Analyse	Johan	200
		005 Gert	Programmering	Ron	50
003 TTV

2.1 Normaliseren



Niet-genormaliseerde gegevensverzameling:

- ☐ Projectnummer
- ☐ Projectomschrijving
- ☐ Budget
- ☐ Medewerkersnummer
- ☐ Naam
- ☐ Afdeling
- ☐ Chef
- ☐ Uren

2.1.1 Eerste normaalvorm

STAP 1: verwijder de zich herhalende deelverzamelingen

1NV: detail:

- 1.1 Inventariseer alle **elementaire** gegevens
- 1.2 Verwijder alle **procesgegevens**
- 1.3 Doe het volgende totdat er geen **nieuwe groepen** meer ontstaan
 - 1.2.1 Geef de **sleutel** van de groep aan
 - 1.5 Geef de deelverzameling aan die een **herhaald** aantal keren voorkomt
 - 1.6 Herhaal de sleutelgegevens van de oorspronkelijke groep samen met de gegevens van de zich herhalende deelverzameling als een nieuwe groep
 - 1.7 **Verwijder** de zich herhalende deelverzameling uit de oorspronkelijk groep
- 1.8 Eind-doe

2.1.1 Eerste normaalvorm



Stap 1.1:

- Alleen elementaire gegevens inventariseren
- Samengestelde gegevens → elementaire gegevens
- Benoemen van een elementair gegeven (naamgeving)

2.1.1 Eerste normaalvorm



Stap 1.2:

- Procesgegevens niet opgenomen in normalisatiestappen (apart noteren)
- Procesgegevens:
 - ▣ Alle voor de berekening benodigde gegevens zijn aanwezig
 - ▣ De voor de berekening benodigde gegevens bevatten op het tijdstip waarop de berekening worden uitgevoerd nog de juiste waarden

2.1.1 Eerste normaalvorm



Stap 1.3

- Zich herhalende deelverzamelingen kunnen genest voorkomen
- 1NV zover doorvoeren totdat alle zich herhalende deelverzamelingen zijn opgedeeld.

2.1.1 Eerste normaalvorm

Stap 1.4

- Kies de meest uitgebreide sleutel
- Na inventarisatie van de elementaire gegevens:

PROJECTOVERZICHT

Projectnummer

Projectomschrijving

Budget

Medewerkersnummer

Naam

Afdeling

Chef

Uren

2.1.1 Eerste normaalvorm

Stap 1.5

- Afhankelijk van de sleutelkeuze in stap 1.4 zullen nul, één of meer zich herhalende deelverzameling aanwezig zijn

PROJECTOVERZICHT

Projectnummer

Projectomschrijving

Budget

Medewerkersnummer

Naam

Afdeling

Chef

Uren



Komt een herhaald aantal keren voor

2.1.1 Eerste normaalvorm

Stap 1.6

- Nu wordt een nieuwe groep gevormd. Deze moet bestaan uit:
de zich **herhalende** deelverzameling
+
de **sleutel** van de oorspronkelijke groep
- Deze laatste opnemen om koppeling met oorspronkelijk groep te behouden.

2.1.1 Eerste normaalvorm

Zo krijgen we volgende twee groepen:

PROJECTOVERZICHT:

Projectnummer
Projectomschrijving
Budget
Medewerkersnummer
Naam
Afdeling
Chef
Uren

} **Herhalend**

PROJECTMEDEWERKER

Projectnummer
Medewerkersnummer
Naam
Afdeling
Chef
Uren

2.1.1 Eerste normaalvorm

Stap 1.7

- Oorspronkelijke groep aanpassen → verwijdering

PROJECTOVERZICHT:

Projectnummer
Projectomschrijving
Budget

PROJECTMEDEWERKER

Projectnummer
Medewerkersnummer
Naam
Afdeling
Chef
Uren

2.1.1 Eerste normaalvorm



Stap 1.8

- Deze stap, het einde van de DOE-loop, zorgt ervoor dat we weer bij stap 1.4 beginnen voor de nieuw gevormde groep

2.1.1 Eerste normaalvorm

Stap 1.4 (2-de doorgang)

- Samengestelde sleutel projectnummer en medewerkersnummer

1 NV

PROJECTOVERZICHT:

Projectnummer
Projectomschrijving
Budget

PROJECTMEDEWERKER

Projectnummer
Medewerkersnummer
Naam
Afdeling
Chef
Uren

2.1.2 Tweede normaalvorm

Stap 2: Verwijder de attributen die functioneel afhankelijk zijn van slechts een gedeelte van de sleutel

- **Alleen groepen met een samengestelde sleutel komen in aanmerking!**
- **Detail 2NV:**
 - 2.1 Geef de attributen aan die niet functioneel afhankelijk zijn van de **volledige** sleutel
 - 2.2 Vorm een aparte groep voor ieder deel van de sleutel waarvan attributen functioneel afhankelijk zijn
 - 2.3 Neem in iedere groep de attributen met het bijhorend sleuteldeel op
 - 2.4 Verwijder deze attributen uit de oorspronkelijke groep

2.1.2 Tweede normaalvorm

- Vraagstelling per attribuut:

- ▣ Welk gegeven of combinatie van gegevens identificeert dit attribuut op een eenduidige wijze?

Attribuut **Naam** is functioneel afhankelijk van sleutel

Medewerkersnummer

Attribuut **Afdeling** is functioneel afhankelijk van sleutel

Medewerkersnummer

Attribuut **Chef** is functioneel afhankelijk van sleutel

Medewerkersnummer

Attribuut **Uren** is functioneel afhankelijk van sleutel

Medewerkersnummer
+
Projectnummer

2.1.2 Tweede normaalvorm



Stap 2.2

In ons voorbeeld ontstaat slechts één nieuwe groep:

MEDEWERKER

2.1.2 Tweede normaalvorm

Stap 2.3

- Geïnterpreteerde groepen moeten gevuld worden
- Iedere nieuwe groep moet voldoen aan de eisen van de tweede normaalvorm
 - ▣ Er mogen dus géén herhalingen aanwezig zijn
 - ▣ En alle attributen moeten functioneel afhankelijk zijn van de **volledige sleutel**

2.1.2 Tweede normaalvorm



Volgende groepen ontstaan:

PROJECTMEDEWERKER

Projectnummer

Medewerkersnummer

Naam

Afdeling

Chef

Uren

MEDEWERKER

Medewerkersnummer

Naam

Afdeling

Chef

2.1.2 Tweede normaalvorm

Stap 2.4

- Verwijdering van de attributen uit de oorspronkelijk groep
- De sleutel van de oorspronkelijk groep mag niet aangetast worden!

2 NV

PROJECT: <u>Projectnummer</u> Projectomschrijving Budget	PROJECTMEDEWERKER <u>Projectnummer</u> <u>Medewerkersnummer</u> Uren	MEDEWERKER <u>Medewerkersnummer</u> Naam Afdeling Chef
---	---	--

2.1.3 Derde normaalvorm

Stap 3: Verwijder attributen die ook functioneel afhankelijk zijn van ander (niet-sleutel) attributen

Bij de stap naar de derde normaalvorm moeten de attributen

- ▣ Die functioneel afhankelijk zijn van de volledige sleutel (2NV),
- ▣ Maar ook nog functioneel afhankelijk zijn van andere attributen,

in aparte groepen worden opgenomen

2.1.3 Derde normaalvorm

Detail 3NV:

- 4.1 Geef de attributen aan die ook functioneel afhankelijk zijn van andere (niet-sleutel) attributen
- 4.2 Vorm een aparte groep voor ieder attribuut of combinatie van attributen, waar andere attributen functioneel van afhankelijk zijn
- 4.3 Neem in iedere nieuwe groep de attributen met hun bijbehorende sleutel op
- 4.4 Verwijder de attributen van de nieuwe groep(en) uit de oorspronkelijke groep

2.1.3 Derde normaalvorm

Stap 4.1

- Attributen die functioneel afhankelijk zijn van andere (niet-sleutel) attributen
- Projectoverzicht: Geen
- Projectmedewerker: Geen
- Medewerker: Chef is functioneel afhankelijk van afdeling

2.1.3 Derde normaalvorm



Stap 4.2

- In ons voorbeeld ontstaat slechts één nieuwe groep:

AFDELING

2.1.3 Derde normaalvorm

Stap 4.3

- De in stap 4.2 geïnventariseerde groepen moeten nu van sleutels en attributen worden voorzien
- De nieuwe groep moet wel aan de definitie van de derde normaalvorm voldoen:
 - ▣ **Zij mogen géén herhalingen bevatten**
 - ▣ **Én alle attributen moeten functioneel afhankelijk zijn van de volledige sleutel**
 - ▣ **Én onderling géén functionele afhankelijkheden bevatten.**
- De nieuwe groep afdeling bestaat uit twee gegevens:

Afdeling en Chef

2.1.3 Derde normaalvorm

- Sleutel: Afdeling of Chef?
- Kunnen zich verschillende situaties voordoen → gebruiker kiest zijn situatie
- Situatie A:
 - ▣ Iedere afdeling heeft één chef en iedere chef is chef van één afdeling

Afdeling	Chef
A	1
B	2
C	3

- ▣ Sleutel: keuze is niet belangrijk

2.1.3 Derde normaalvorm

□ Situatie B:

- Iedere afdeling heeft één chef, maar een chef kan van meerdere afdelingen chef zijn:

Afdeling	Chef
A	1
B	2
C	1

- Sleutel: Afdeling

2.1.3 Derde normaalvorm

□ Situatie C:

- Iedere afdeling heeft meerdere chefs, maar iedere chef is slechts chef van één afdeling

Afdeling	Chef
A	1
A	2
B	3
B	4

- Sleutel: Chef

2.1.3 Derde normaalvorm

□ Situatie D:

- Iedere afdeling heeft meerdere chefs en iedere chef kan chef zijn van meerdere afdelingen.

Afdeling	Chef
A	1
A	2
B	3
B	1

- Sleutel: Combinatie van beide is uniek → samengestelde sleutel

2.1.3 Derde normaalvorm

- Onze keuze: situatie A, maar situatie B wordt niet uitgesloten.
- Medewerker is opgebouwd uit de volgende groepen:

MEDEWERKER

Medewerkersnummer

Naam

Afdeling

Chef

AFDELING

Afdeling

Chef

2.1.3 Derde normaalvorm



Stap 4.4: verwijderen...

Uit de nu nog intact zijnde oorspronkelijke groep moeten de attributen van de nieuwe ontstane groep(en) verwijderd worden

+

De sleutel van de nieuwe groep blijft als attribuut in de oorspronkelijke groep gehandhaafd

2.1.3 Derde normaalvorm

Volgende groepen ontstaan in de 3NV

3 NV

PROJECT:

Projectnummer
Projectomschrijving
Budget

PROJECTMEDEWERKER

Projectnummer
Medewerkersnummer
Uren

MEDEWERKER

Medewerkersnummer
Naam
Afdeling

AFDELING

Afdeling
Chef

2.1.3 Derde normaalvorm

- Alle nieuw gevormde groepen voldoen immers aan de definitie van de derde normaalvorm:
 - ▣ Zij mogen géén herhalingen bevatten
 - ▣ Alle attributen in elke groep op zich moeten functioneel afhankelijk zijn van de volledige sleutel van die groep
 - ▣ Alle attributen in elke groep op zich mogen onderling géén functionele afhankelijkheden bevatten

2.1.4 Verder normaliseren

- BCNF: (Boyce-Code of 4.5 NV) werkt alleen anders dan 3NV als er *meerdere overlappende kandidaat-sleutels* zijn en komt heel zeldzaam voor.
- 4NV: Terwijl de tweede , derde en Boyce-Codd normaalvormen zich bezighouden met functionele afhankelijkheden , houdt 4NF zich bezig met een meer algemene afhankelijkheid die bekend staat als een *meerwaardige afhankelijkheid* .
- 5NV: redundantie in relationele databases door gerelateerde *meervoudige relaties* te isoleren

2.2 Entity-relationship diagram (ERD) - ontwerp

- **Wat?** Het entity-relationship model of entity-relationship diagram (ER-diagram) is een visuele weergave van de tabellen, relaties en regels.
- **Gebruik?** ontwerp van een database te schetsen.
- **Wanneer?** Nieuw ontwerp of documenteren van bestaand structuur (bij dit laatste export vanuit dB als csv file automatisch ontwerp)

□

2.2 Entity-relationship diagram (ERD) - ontwerp

□ Begrippen

- ▣ Entiteit: Student, boek (in bib)
- ▣ Entiteitsverzameling: tabel met alle verwante studenten
- ▣ Relatie: koppeling tussen tabellen
- ▣ Attributen/kolom: Studnr-Naam-Adres-Opleiding- ...
- ▣ Attribuutwaarde: 20190258-Annaïck Vandebeek-...

2.2.1 ER-diagram symbolen

Vijf hoofdonderdelen van een ERD

Entiteit



Acties

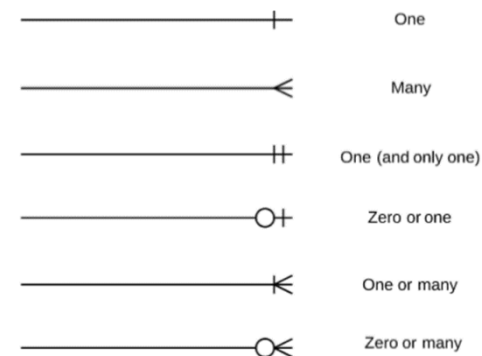
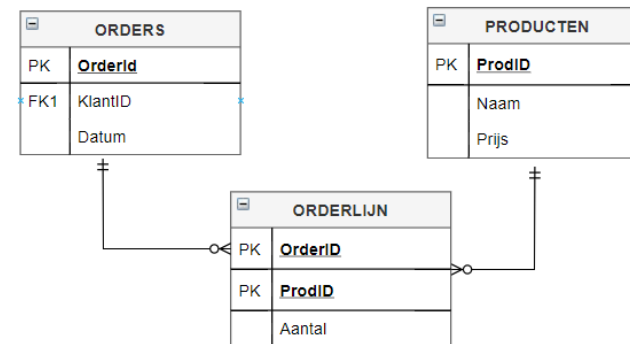


Attributen

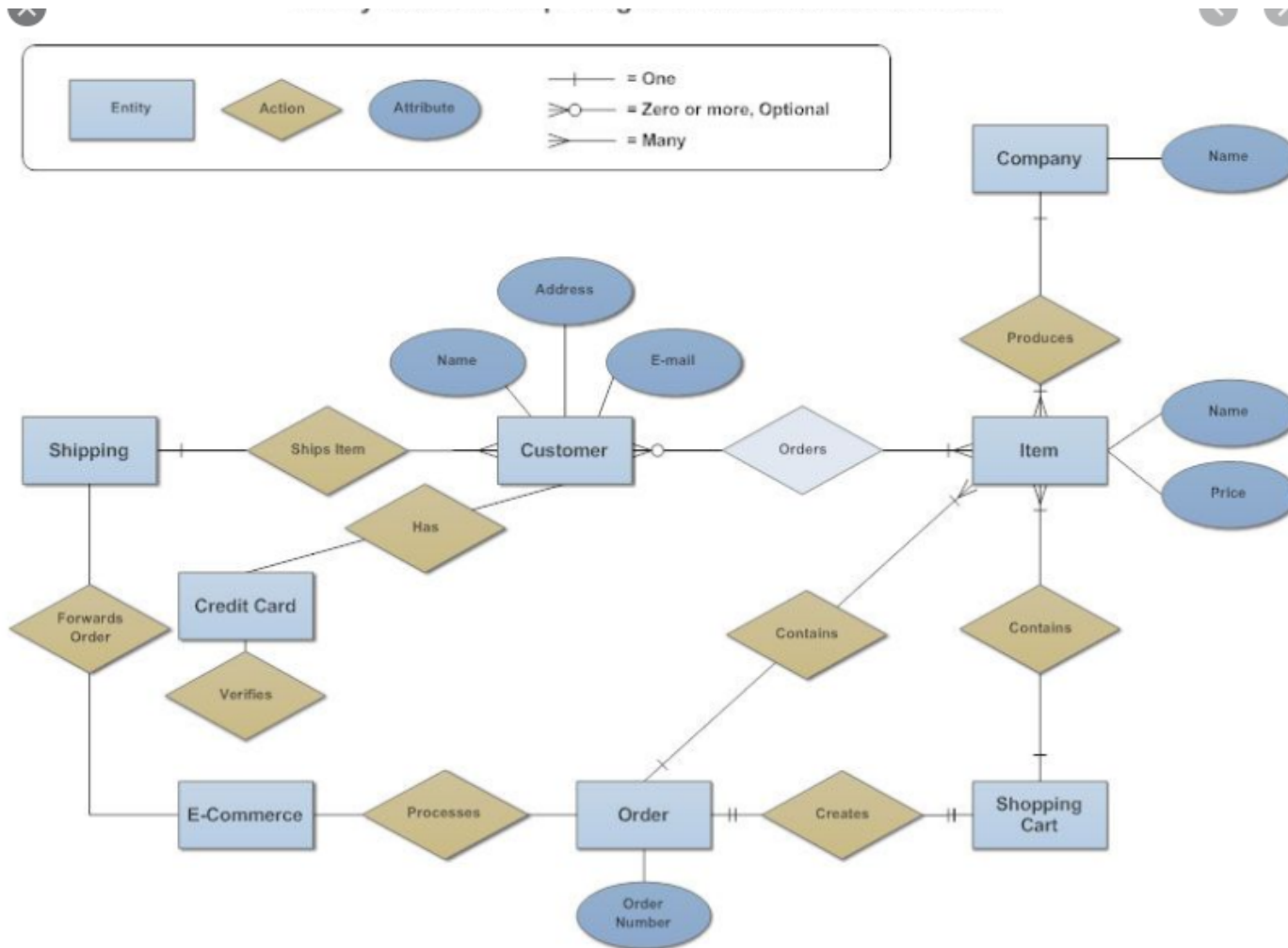


Verbindinglijnen

Kardinaliteit/Ordinaliteit



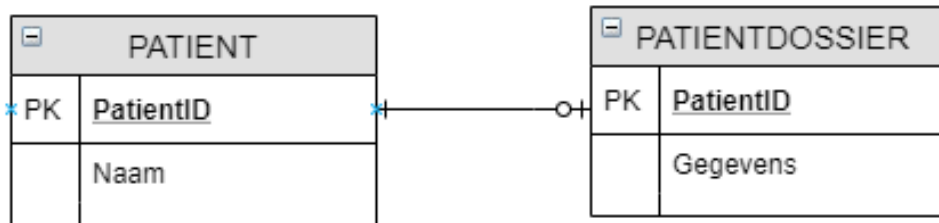
2.2.1 ER-diagram - voorbeeld



2.2.2 ER-diagram - Kardinaliteit in een relatie

□ 1-op-1 relatie

Dit type relatie komt niet zo vaak voor maar voor de volledigheid hoort het er toch bij. Je moet je afvragen of je de beide tabellen in de 1-op-1 relatie niet moeten worden samengevoegd.



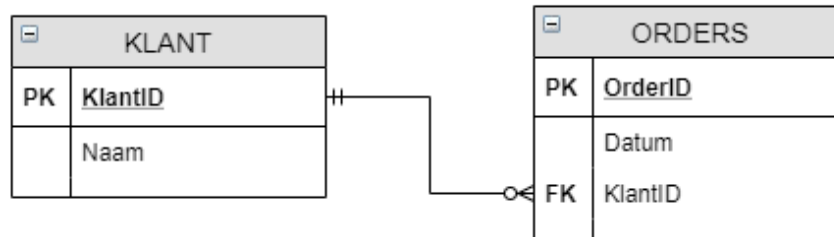
- Kenmerkend voor de 1-op-1 relatie is de primary key van beide tabellen. Deze zijn namelijk gelijk aan elkaar.

De relatie gelezen van rechts naar links: het dossier van een patient hoort bij één patient.

De relatie gelezen van links naar rechts: Een patient heeft nul of één patientdossier.

2.2.2 ER-diagram - Kardinaliteit in een relatie

□ 1-op-veel relatie (meest voorkomende relatie)



- Bij de tabel ORDERS verwijst de foreign key KlantID naar KlantID in de tabel KLANT.

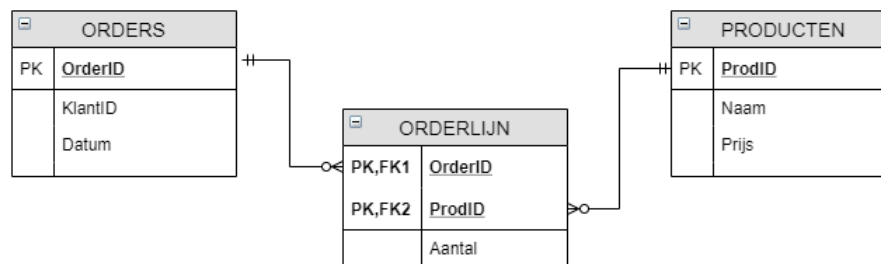
De relatie gelezen van links naar rechts: een klant heeft nul of meerdere orders.

De relatie gelezen van rechts naar links: een order wordt geplaatst door één klant.

2.2.2 ER-diagram - Kardinaliteit in een relatie

□ Veel-op-veel relatie (met koppeltabel)

Een veel-op-veel relatie heeft altijd een zogenaamde koppeltabel.



- De tabel ORDERLIJN (koppeltabel) verzorgt de veel-op-veel relatie tussen ORDERS en PRODUCTEN.

De relatie gelezen van links naar rechts: Een order bevat nul of meerdere producten.

De relatie gelezen van rechts naar links: Een product hoort bij nul of meerdere orders.

2.2.3 ER-diagram - Tips bij ontwerp in draw.io

□ STAP 1 Identificeer de entiteiten

Een entiteit is een rechthoek met alle attributen beschreven.

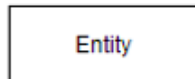


Table	
PK,FK1	<u>row 1</u>
PK,FK2	<u>row 2</u>
	row 1

Table	
Row 1	
Row 2	
Row 3	

Table	
PK	<u>uniqueId</u>
	row 1
	row 2
	row 3

draw.io installatie:

<https://github.com/jgraph/drawio-desktop/releases/tag/v14.4.3>

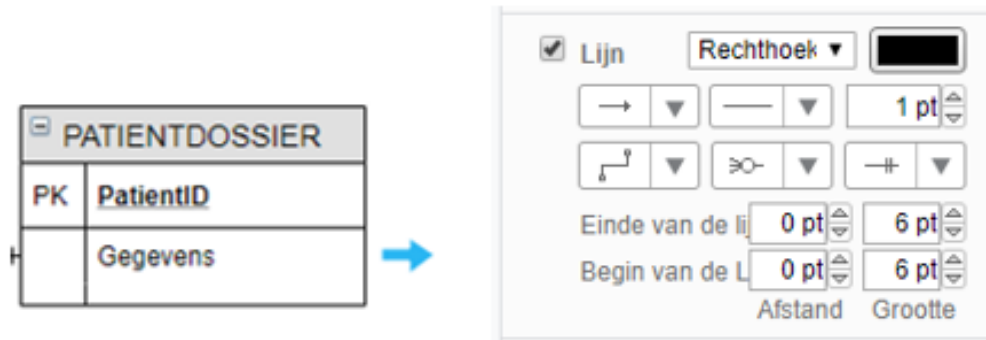
rechtstreekse downloadlink:

<https://github.com/jgraph/drawio-desktop/releases/download/v14.4.3/draw.io-14.4.3-windows-installer.exe>

2.2.3 ER-diagram - Tips bij ontwerp in draw.io

□ **STAP 2 Identificeer de relaties**

Sleep de blauwe pijl vanuit tabel met de vreemde sleutel naar tabel met de primaire sleutel. Selecteer de pijl en pas eventueel het begin- of eindpunt aan om de kardinaliteiten/ordinaliteiten te specificeren.



- **STAP Voltooi het diagram.** Zorg dat alle entiteiten verbonden zijn en de relaties (pijlen) tussen de entiteiten correct zijn. Voeg eventueel kleuren, nota's, titels, tekst, links,.... toe.