- DEV6B Data & database normalisatie

Quick & Dirty Recap

Wat gaan we bespreken

- Database tabellen
- Data
- Normalisatie (1NF -> BCNF)
- Opdracht



Wat heb je nodig?

Mentale flexibiliteit

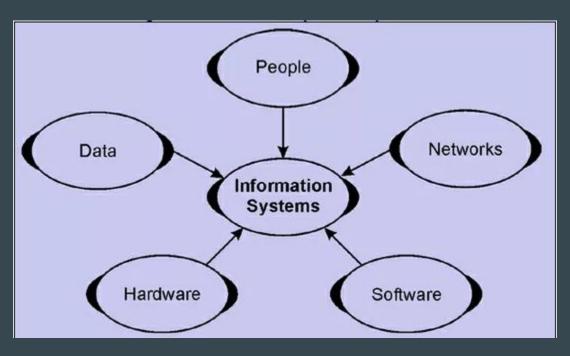
~ 2-2:30 aan tijd

Je weet "in grote lijnen" wat een database is

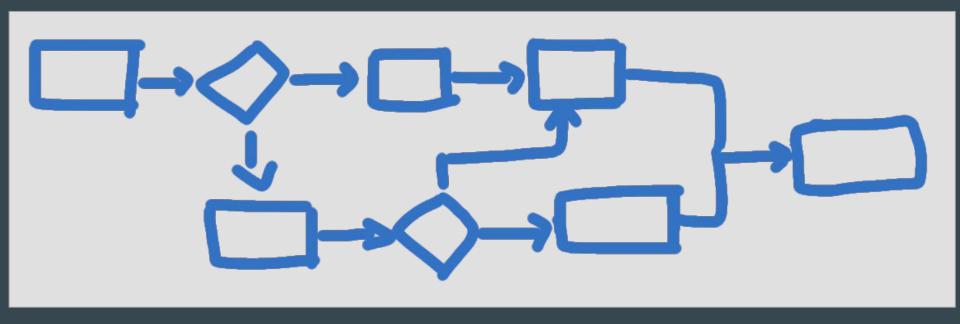
Pen en papier (?!)













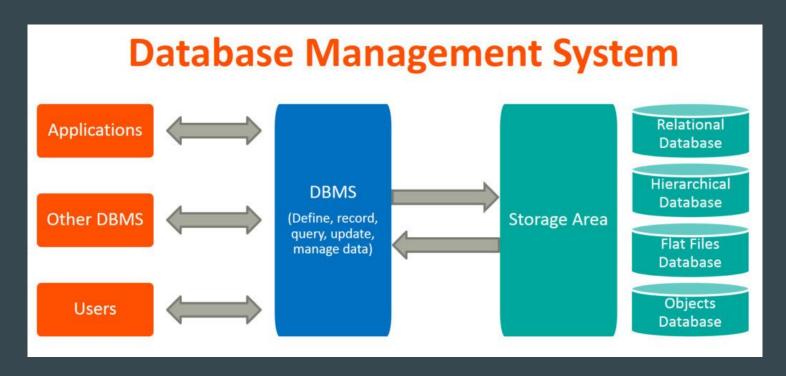
Waarmee kunnen wij u helpen? *	Vul reden van afspraak in			
Op welke manier wilt u uw hypotheekgesprek voeren? *	Geen voorkeur			
Aanhef				
Voorletters Tussenvoegsel Achternaam	J.J. van Rozenhart			
Op welke datum wilt u gebeld worden? *	20-3-2016			
Telefoonnummer *	0612345678			
E-mailadres *	naam@voorbeeld.nl			
Postcode *	1102 BS			

verzenden







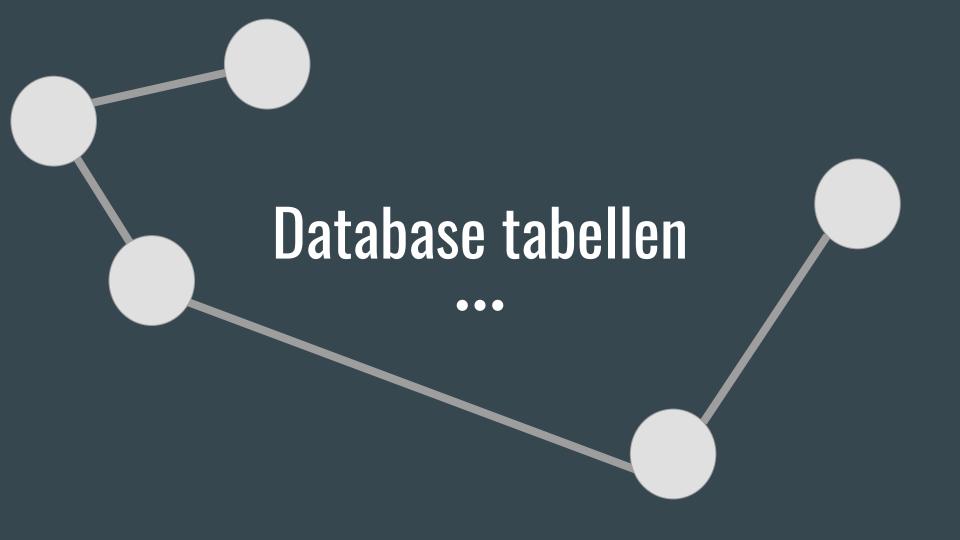








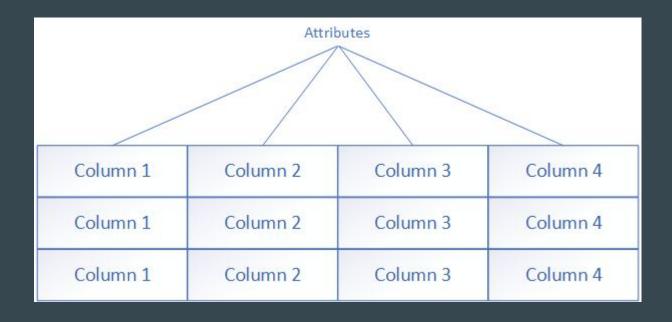
■ Browse Structure SQL Search Search Export Manager M							
	#	Column	Type	Collation	Attributes	Null	Default
	1	<u>ID</u>	bigint(20)		UNSIGNED	No	None
	2	post_author	bigint(20)		UNSIGNED	No	0
	3	post_date	datetime			No	00:00:00:00
	4	post_date_gmt	datetime			No	00:00:00:00
	5	post_content	longtext	utf8_general_ci		No	None
	6	post_title	text	utf8_general_ci		No	None
	7	post_excerpt	text	utf8_general_ci		No	None
	8	post_status	varchar(20)	utf8_general_ci		No	publish
	9	comment_status	varchar(20)	utf8_general_ci		No	open
	10	ping_status	varchar(20)	utf8_general_ci		No	open
	11	post_password	varchar(20)	utf8_general_ci		No	
	12	post_name	varchar(200)	utf8_general_ci		No	
	13	to_ping	text	utf8_general_ci		No	None
	14	pinged	text	utf8_general_ci		No	None





Column 1	Column 2	Column 3	Column 4
Column 1	Column 2	Column 3	Column 4
Column 1	Column 2	Column 3	Column 4





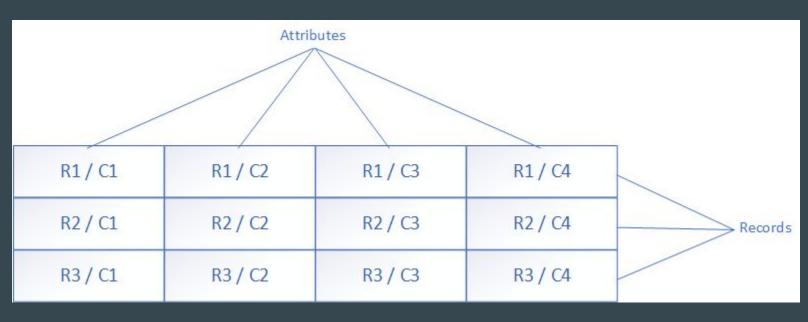


Row 1	Row 1	Row 1	Row 1
Row 2	Row 2	Row 2	Row 2
Row 3	Row 3	Row 3	Row 3

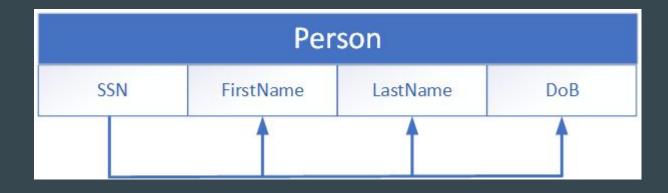


Row 1	Row 1	Row 1	Row 1	
Row 2	Row 2	Row 2	Row 2	Records
Row 3	Row 3	Row 3	Row 3	

















CSAR .

Keys! Probeer ze te onthouden!...

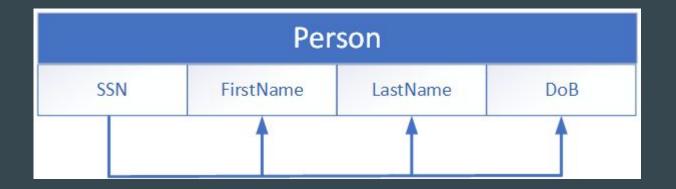




Keys

Candidate key

Atribu(u)t(en) wat de uniekheid garandeert voor een record in een tabel.



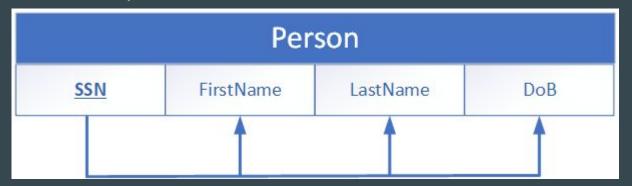


Keys

Candidate key

Atribu(u)t(en) wat de uniekheid garandeert voor een record in een tabel.

SSN is een "candidate key" en, dus, een "Prime attribute".

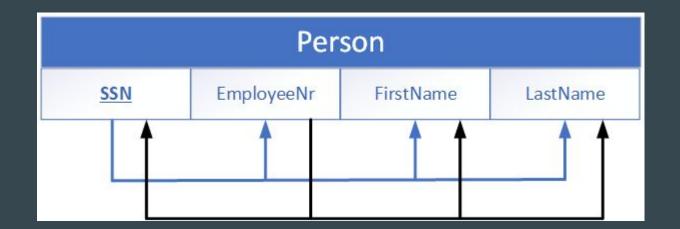




Keys

Candidate key

Atribu(u)t(en) wat de uniekheid garandeert voor een record in een tabel.



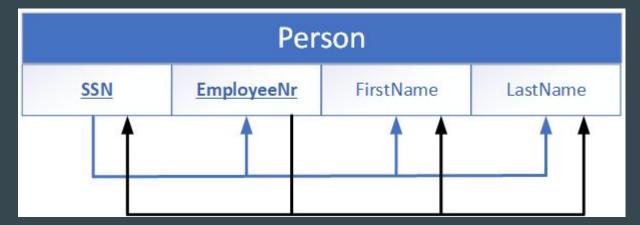


Keys

Candidate key

Atribu(u)t(en) wat de uniekheid garandeert voor een record in een tabel.

SSN & EmployeeNr zijn beide een (aparte) Candidate key





Keys

Candidate key

Atribu(u)t(en) wat de uniekheid garandeert voor een record in een tabel.





Keys

Candidate key

Atribu(u)t(en) wat de uniekheid garandeert voor een record in een tabel.

[PostalCode, HouseNumber] is een candidate key





Keys

Primary key

Één van de "candidate keys" waarin geen null waardes zijn opgenomen.

Voor elke tabel; maximaal 1 Primary key!

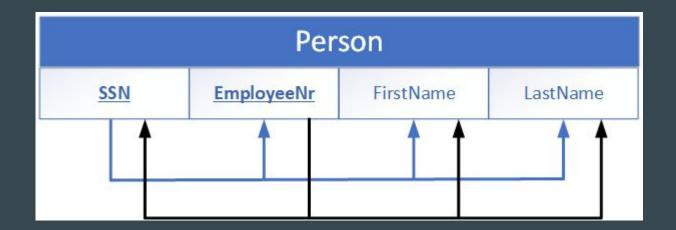
(LET OP: De primary key kan dus uit meerdere attributen bestaan!)



Keys

Primary key

Één van de "candidate keys" waarin geen null waardes zijn opgenomen.





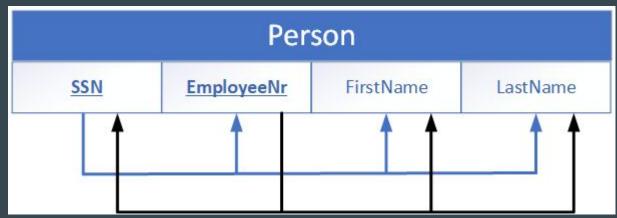
Keys

Primary key

Één van de "candidate keys" waarin geen null waardes zijn opgenomen.

Wanneer er meerdere candidate keys zijn (die voldoen aan de regels) beslissen wij wat

de primary key is!



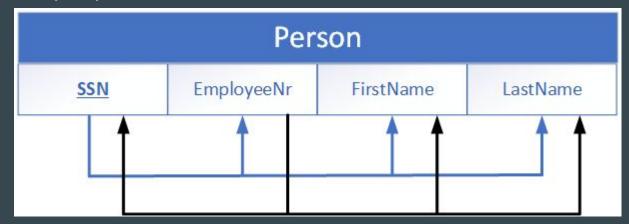


Keys

Primary key

Één van de "candidate keys" waarin geen null waardes zijn opgenomen.

SSN is de primary key

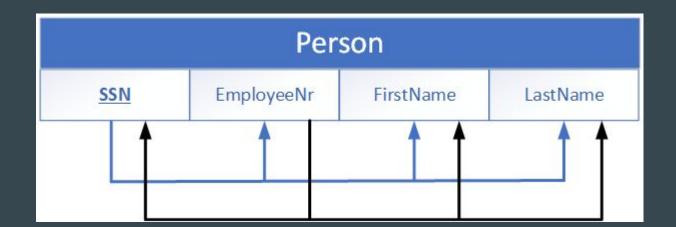




Keys

Alternate key

De "candidate keys" die geen "primary key" zijn.



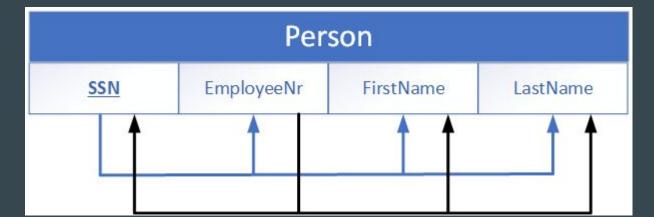


Keys

Alternate key

De "candidate keys" die geen "primary key" zijn.

SSN is de "primary key"; EmployeeNr een "alternate key"





Keys

Super key

Een "candidate key" plus nul, of meer, attributen.

Elke candidate key = = super key

Niet elke super key = = candidate key \rightarrow De kleinste super key is een candidate key



Keys

Foreign key

Een "foreign key" wordt gedefinieerd in een secundaire tabel.

De foreign key verwijst naar de "primary key" van een main tabel.



Data



Denk niet in tabellen; denk in data en afhankelijkheid!



Context == Data & Relaties!





Data						
Attribute A	Attribute B	Attribute C	Attribute D		p.ee.	

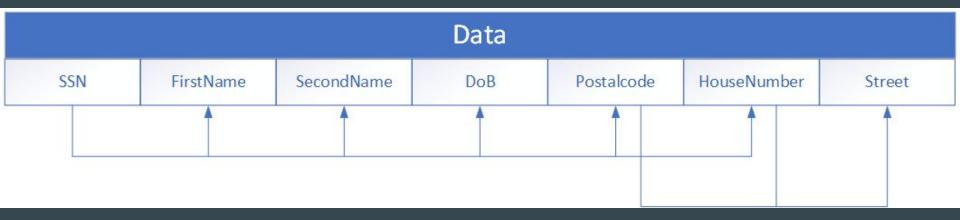


Data						
SSN	FirstName	SecondName	DoB	Postalcode	HouseNumber	Street











De namen van attributen maken niet uit.

De onderlinge relatie tussen data is essentieel!

Data							
А	В	С	D	Е	F	G	
		1					

$$R1 = \{ A, B, C, D, E, F, G \}$$

 $A \rightarrow \{B, C, D, E, F\}$

 $\{E, F\} \rightarrow G$



			Data			
А	В	С	D	Е	F	G
	1	A		1		

$$R1 = \{ A, B, C, D, E, F, G \}$$

$$A \rightarrow \{B, C, D, E, F\}$$

$$\{E, F\} \rightarrow G$$



1. $A = \{A, B, C, \overline{D, E, F}\}$



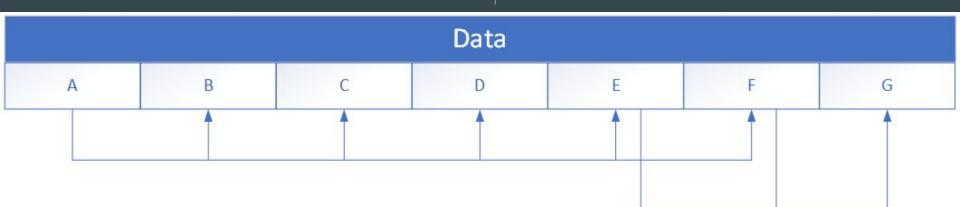
$$R1 = \{ A, B, C, D, E, F, G \}$$

$$A \rightarrow \{B, C, D, E, F\}$$

$$\{E, F\} \rightarrow G \blacktriangleleft$$







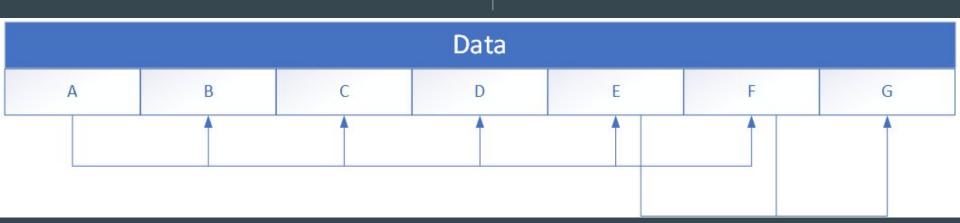
$$R1 = \{ A, B, C, D, E, F, G \}$$

$$A \rightarrow \{B, C, D, E, F\}$$

$$\{E, F\} \rightarrow G$$

1.
$$A = \{A, B, C, D, E, F\}$$

1.1.
$$A = \{A, B, C, D, E, F, G\}$$



$$R1 = \{ A, B, C, D, E, F, G \}$$

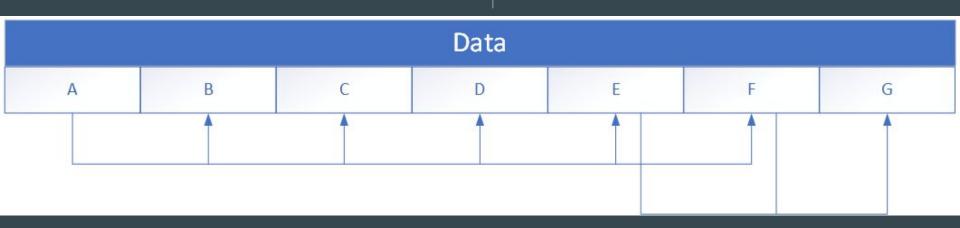
$$A \rightarrow \{B, C, D, E, F\}$$

$$\{E, F\} \rightarrow G$$

1.
$$A = \{A, B, C, \overline{D, E, F}\}$$

1.1.
$$A = \{A, B, C, D, E, F, G\}$$

$$1.1.1. A (Primary key) = R1$$





Normalisatie - Wat bereiken we ermee?



- Geen redundantie in data!
 Door data één enkele keer op te slaan verlagen we de kansen op foutieve data!
- Object-to-data mapping
 Wanneer een database genormaliseerd is, ligt deze conceptueel dichtbij OO.
 (Entity framework (Code first DB(?)))

-1. "Langzame" raportages

Door het normaliseren introduceren we veel/meer "look-up" tabellen,
het doorlopen van deze tabellen kost tijd.

Functional dependency

Transitive dependency





Functional dependency (FD)

"A functional dependency is a constraint that describes the relationship between attributes in a relation"

 $FD: X \rightarrow Y$

CSAR .

Functional dependency (FD)

 $FD: X \rightarrow Y$

(De waarde van \mathbf{Y} wordt gedefinieerd aan de hand van \mathbf{X})



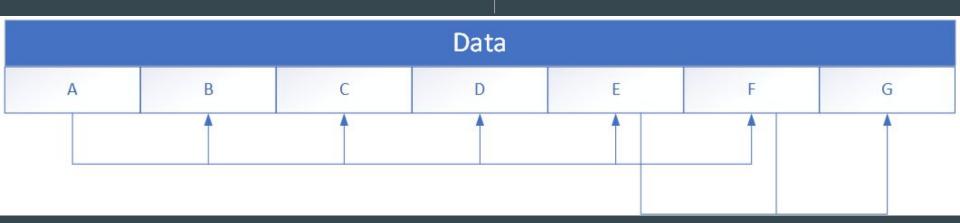
$$R1 = \{ A, B, C, D, E, F, G \}$$

$$A \rightarrow \{B, C, D, E, F\}$$

$$\{E, F\} -> G$$

$$FD: A \rightarrow \{B, C, D, E, F\}$$

$$FD: \{E, F\} \rightarrow G$$





Transitive dependency

A transitive dependency is a functional dependency which holds by virtue of transitivity:

 $FD: X \rightarrow Y$

 $FD: Y \rightarrow Z$

==

 $TD: X \rightarrow Z$



Transitive dependency

 $FD: X \rightarrow Y$

 $FD: Y \rightarrow Z$

 $TD: X \rightarrow Z$

 $FD: A \rightarrow \{..., E, F\}$

 $FD: \{E, F\} \rightarrow G$

 $TD: A \rightarrow G$





Case info

Geen enkel veld mag null waardes bevatten.

Candidate keys zijn automatisch uniek.

<u>Let op:</u> Dit zijn en blijven voorbeelden...

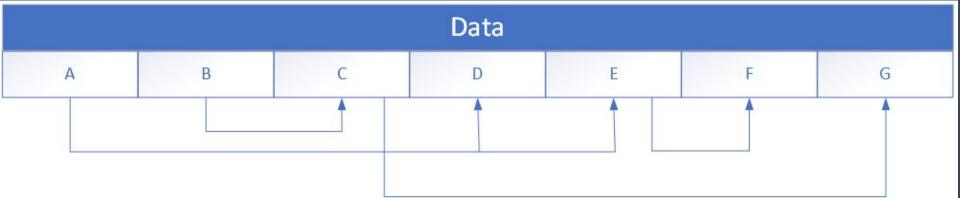
Deze informatie verkrijg je normaal door je opdrachtgever! (context)

R1 = {A, B, C, D, E, F, G}

$$A \rightarrow \{D, E\}$$

 $B \rightarrow C$
 $C \rightarrow G$
 $E \rightarrow F$



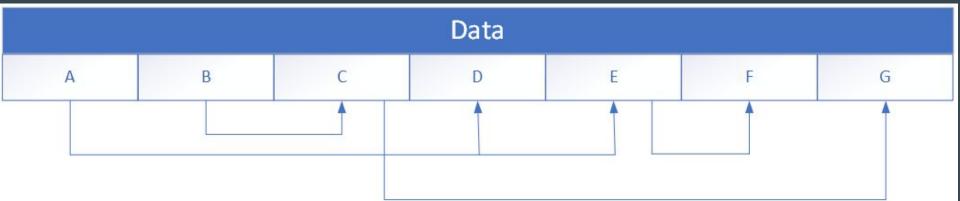


R1 = {A, B, C, D, E, F, G} $A \rightarrow \{D, E\}$ $B \rightarrow C$ $C \rightarrow G$

 $E \rightarrow F$

Candidate key?







```
R1 = {A, B, C, D, E, F, G}

A \rightarrow \{D, E\}

B \rightarrow C

C \rightarrow G

E \rightarrow F
```

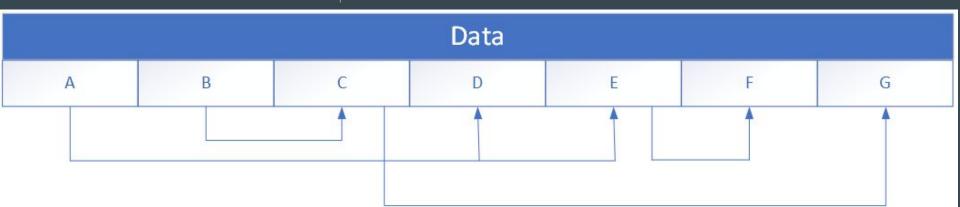
Candidate key?

1. $\{A, B\} \rightarrow \{A, B, C, D, E\}$

1.1. $\{A, B, C, D, E\} \rightarrow \{A, B, C, D, E, F, G\}$ (R1)

 $1.1.1. \{A, B\} == \{A, B, C, D, E, F, G\} (R1)$

1.1.1.1. {A, B} == Primary key

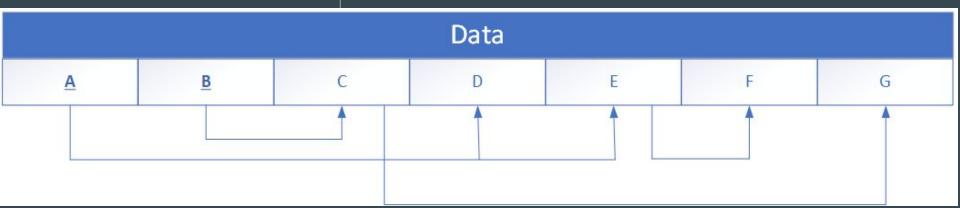


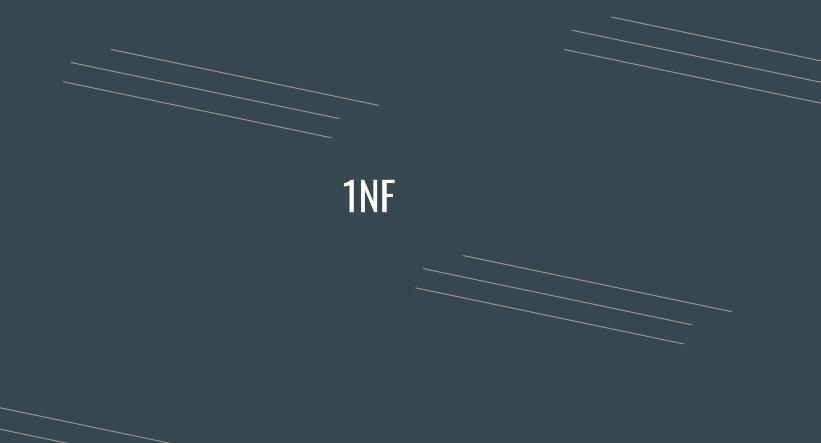
R1 = {A, B, C, D, E, F, G} $A \rightarrow \{D, E\}$ $B \rightarrow C$ $C \rightarrow G$

 $E \rightarrow F$

Primary key: {A, B}







Normalisatie - 1st normal form

Regels

Elke attribuut bevat atomische data! *Attributen mogen alleen één enkele waarde bevatten.*



Normalisatie - 1st normal form



Regels

Elke attribuut (kolom) bevat atomische data! *Attributen mogen alleen één enkele waarde bevatten.*

Data					
PostalCode	HouseNumber	Country, City, Street			

Dit is fout!

Normalisatie - 1st normal form



Dit is fout!

Data						
PostalCode	HouseNumber	Country, City, Street				

Data						
PostalCode	HouseNumber	Country	City	Street		

Dit is goed!





Regels

1st normal form

Primary key \rightarrow Non-prime FD's zijn afhankelijk van de **gehele** primary key.

Let op: Non-prime \rightarrow Non-prime FD's worden voor nu "genegeerd". (tot 3NF)

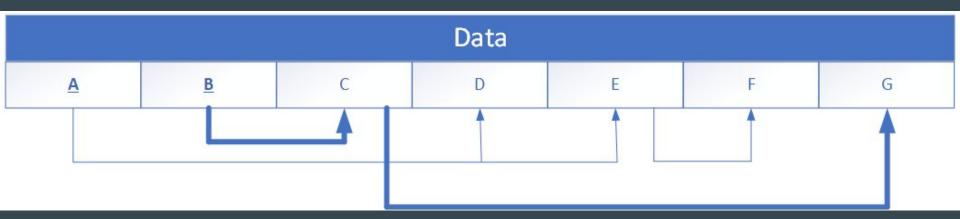


Regels

1st normal form

Primary key \rightarrow Non-prime FD's zijn afhankelijk van de **gehele** primary key.

Let op: Non-prime \rightarrow Non-prime FD's worden voor nu "genegeerd". (tot 3NF)



R1 = {A, B, C, D, E, F, G} A \rightarrow {D, E}

 $\mathbf{B} \to \mathbf{C}$

 $C \rightarrow G$

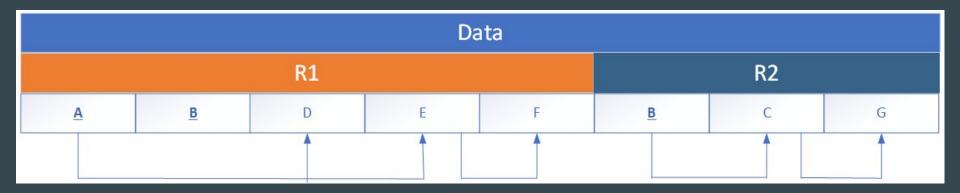
 $E \rightarrow F$

CSAR

Data							
<u>A</u>	<u>B</u>	С	D	Е	F	G	

Primary key: {A, B}





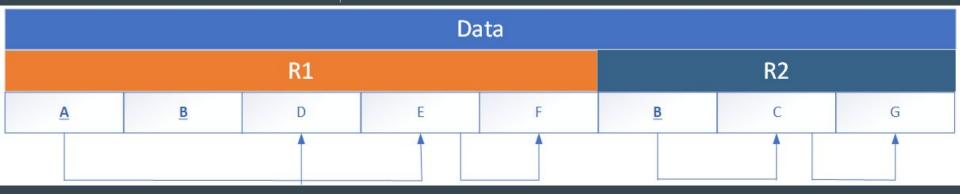
R1 = {A, B, D, E, F} R2 = {B, C, G} $A \rightarrow \{D, E\}$ $B \rightarrow C$ $C \rightarrow G$

 $E \rightarrow F$

Primary key: {A, B}

Primary key: B



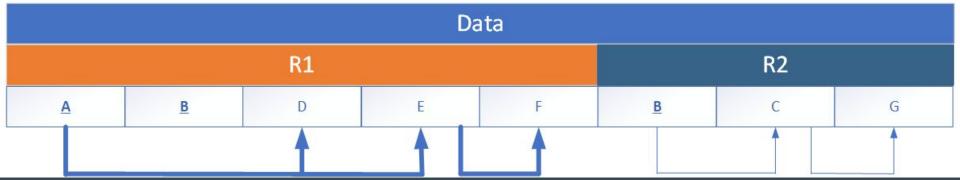


R1 = {A, B, D, E, F} R2 = {B, C, G} $A \rightarrow \{D, E\}$ $B \rightarrow C$ $C \rightarrow G$

 $E \rightarrow F$

Primary key: {A, B} Primary key: B





Normalisatie - 2nd normal form



Data										
R	R1		R2			R3				
<u>A</u>	<u>B</u>	<u>B</u>	С	G	A	D	E	F		
						^				

Normalisatie - 2nd normal form

R1 = {A, B} R2 = {B, C, G} R3 = {A, D, E, F} $A \rightarrow \{D, E\}$ $B \rightarrow C$ $C \rightarrow G$ $E \rightarrow F$ Primary key: {A, B} Primary key: B Primary key: A







Normalisatie - 3rd normal form



Regels

2nd normal form

Alle attributen in een relatie worden gedefinieerd door de primary key van de relatie.

Normalisatie - 3rd normal form



Regels

2nd normal form

Alle attributen in een relatie worden gedefinieerd door de primary key van de relatie.



CSAR

R1 = {A, B} R2 = {B, C, G} R3 = {A, D, E, F} $A \rightarrow \{D, E\}$ $B \rightarrow C$ $\mathbf{C} \rightarrow \mathbf{G}$ $E \rightarrow F$ Primary key: {A, B} Primary key: B Primary key: A



 $R1 = \{A, B\}$

 $R2 = \{B, C\}$

 $R3 = \{A, D, E, F\}$

 $R4 = \{C, G\}$

 $A \longrightarrow \{D, E\}$

 $B \rightarrow C$

 $C \rightarrow G$

 $E \rightarrow F$

Primary key: {A, B}

Primary key: B

Primary key: A

Primary key: C



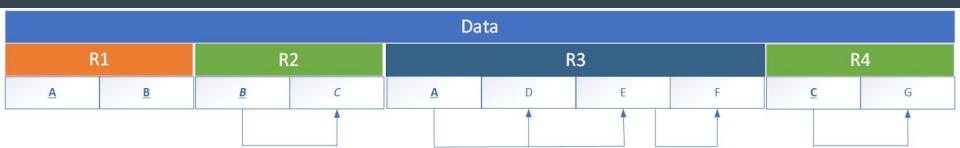




Regels

2nd normal form

Alle attributen in een relatie worden gedefinieerd door de primary key van de relatie.



 $R1 = \{A, B\}$

 $R2 = \{B, C, G\}$

 $R3 = \{A, D, E, F\}$

 $R4 = \{C, G\}$

 $A \rightarrow \{D, E\}$

 $B \rightarrow C$

 $C \rightarrow G$

 $E \rightarrow F$

Primary key: {A, B}

Primary key: B

Primary key: A

Primary key: C



Data											
R	R1		R2		R3				R4		
A	<u>B</u>	<u>B</u>	С	<u>A</u>	D	E	F	<u>c</u>	G		
					^						



```
R1 = {A, B}

R2 = {B, C, G}

R3 = {A, D, E}

R4 = {C, G}

R5 = {E, F}

A \rightarrow {D, E} \mid B \rightarrow C

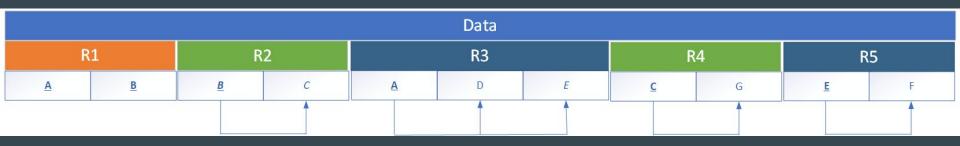
C \rightarrow G \mid E \rightarrow F
```

Primary key: {A, B} Primary key: B

Primary key: A

Primary key: C

Primary key: E



BCNF

Normalisatie - BC normal form



Regels

3rd normal form

Voor elke relatie (R#) is een van de volgende regels waar <u>waar</u>:

 $X \rightarrow Y$;

Waarbij Y een subset van X is. $(X = \{a,b,c\} \mid Y = c)$

 $X \rightarrow Y$;

Waarbij X een superkey is van de relatie (R#).

Normalisatie - Boyce codd normal form

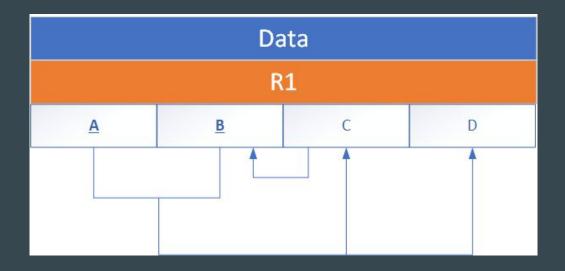


R1 = {A, B, C, D}

$${A, B} \rightarrow {C, D}$$

 ${C} \rightarrow {B}$

Primary key: {A, B}



Normalisatie - 2nd normal form

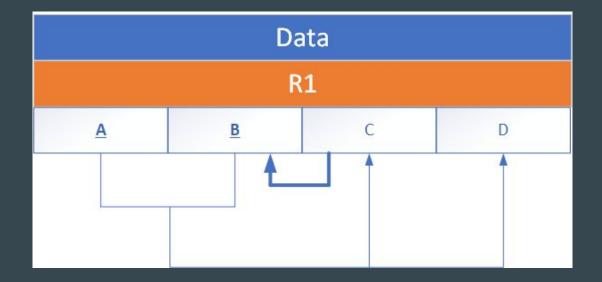


R1 = {A, B, C, D}

$${A, B} \rightarrow {C, D}$$

 ${C} \rightarrow {B}$

Primary key: {A, B}

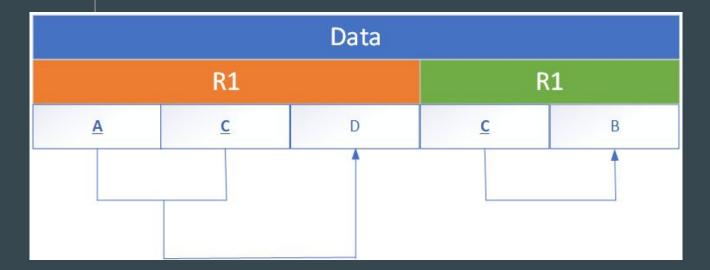


Normalisatie - 2nd normal form

```
CSAR .
```

R1 = {A, C, D}
R2 = {C, B}
{A, B}
$$\rightarrow$$
 {C, D}
 $C \rightarrow B$

Primary key: {A, C} Primary key: C

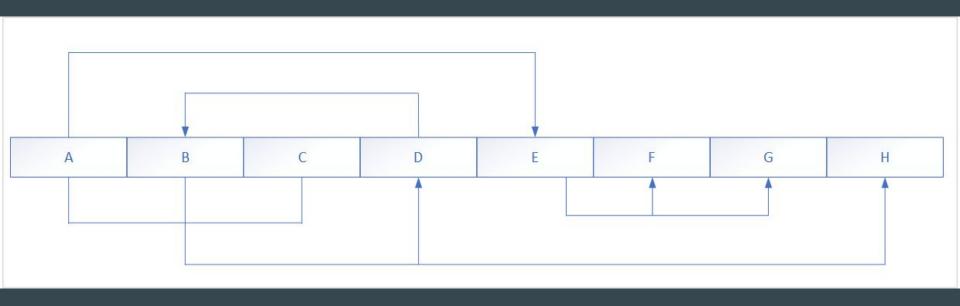


Opdracht

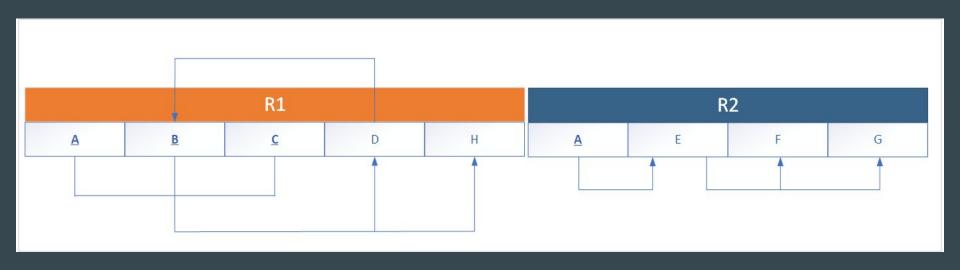
•••

```
0101100 \delta_{1011000} \\ 0101100 \delta_{1011000} \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000 \\ 01011000
```

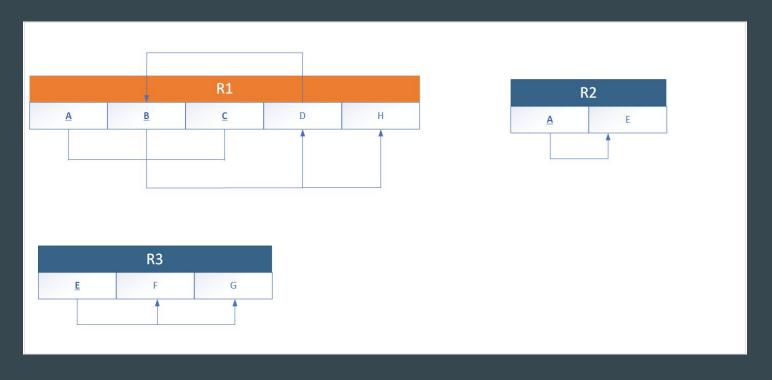




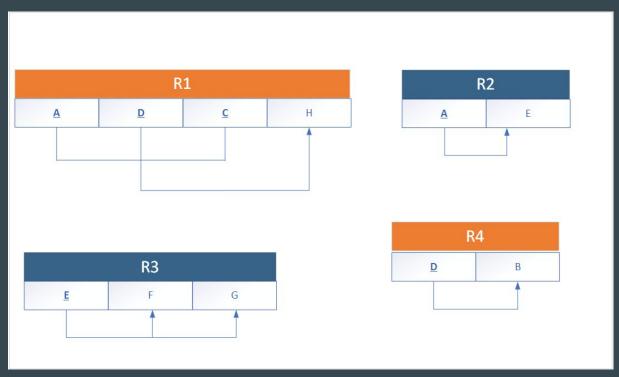












Vragen?



Feedback?

