

Normalization

By Dr. Ahmad Al Shami

Rules of Normalization

1NF Rule: Eliminate repeating groups and make a separate table for each set of related attributes, and give each table a primary key.

2NF Rule: Remove the attribute that depends on only part of a multi-valued key to a separate table.

3NF Rule: Eliminate columns not dependent only on the primary Key and remove them to a separate table.

The First Normal Form (1NF)

A table is said to be in 1NF if it does not contain repeating groups. This is really one of the properties of a relation as discussed in an earlier chapter, a relation must have atomic attributes. Consequently, A table is in 1NF if and only if all underlying domains contain atomic values only.

Consider for example, the following table which is about invoices:

Invoice_number, Customer_number, Customer_name, Customer_address, Invoice_date, Invoice_amount, and for each item the customer has bought, a Product_number, Product_name, Salesman ID, Quantity, Unite_price.

<u>In_No</u>	C_no	C_nm	C_add	In_dt	In_at	P_no	P_nm	Sl_id	Qty	Pirce
100	150	Al	SAU	1/1/99	1000	200	Chair	120	2	50
						300	Bed	120	1	500
						400	Tv	121	1	400
200	180	Ema	SAU	2/2/00	350	200	Chair	120	1	50
						330	Stand	190	3	100

It appears clearly that the above table is not a relation since it has multi values in several columns. For example, P_no, P_nm, Sl_id, Qty, and price have multi values attributes. Hence, the table is not in 1NF.

To transform the invoice table to be in 1NF, we need to remove the repeating attributes by the following decomposition:

Invoice (**Invoice number**, Customer_number, Customer_name, Customer_address, Invoice_date, Invoice_amount)

Invoice relation

<u>In_No</u>	C_no	C_nm	C_add	In_dt	In_at
100	150	Al	SAU	1/1/99	1000
200	180	Ema	SAU	2/2/00	350

Line_Item (**Invoice number**, **Product number**, Product_name, Salesman ID, Quantity, Unite_price)

Line_Item relation

<u>In_No</u>	<u>P_no</u>	P_nm	Sl_id	Qty	Pirce
100	200	Chair	120	2	50
100	300	Bed	120	1	500
100	400	Tv	121	1	400
200	200	Chair	120	1	50
200	330	Stand	190	3	100

The Second Normal Form (2NF)

A relation is in 2NF if it is in 1NF and every non-key attribute is fully dependent on the primary key of the relation. An attribute is called to be a non-key attribute if it is not the primary key and not a part of it.

The concept of 2NF requires that all attributes that are not part of the primary key be fully dependent on the primary key. Consider the relation line_item.

Line_Item (**Invoice number**, **Product number**, Product_name, Salesman ID, Quantity, Unite_price)

Line_Item relation

<u>In_No</u>	<u>P_no</u>	P_nm	Sl_id	Qty	Pirce
100	200	Chair	120	2	50
100	300	Bed	120	1	500
100	400	Tv	121	1	400
200	200	Chair	120	1	50
200	330	Stand	190	3	100

The fields **Invoice_number** and **Product_number** form the primary key of the Line_Item relation. The fields Product_name, Salesman ID, Quantity, and Unite_price are non-key attributes.

The relation is not in 2NF since Product_name and Price are not fully dependent on the primary key (**Invoice_number** and **Product_number**). Product_name and Price fully dependent on Product_number only. Therefore, the above relation suffers from the same anomalies and repetition of information as discussed above since Product_name and Price will be repeated. To resolve these difficulties we could remove those attributes from the relation that are not fully dependent on the primary keys of the relations. Therefore we decompose the relation into the following projections of the original relation:

Product (**Product number**, Product_name, Unite_price)

Product Relation

<u>P_no</u>	<u>P_nm</u>	<u>Pirce</u>
200	Chair	50
300	Bed	500
400	Tv	400
200	Chair	50
330	Stand	100

Line_Item (**Invoice number**, **Product number**, Salesman ID, Quantity)

Line_Item relation

<u>In_No</u>	<u>P_no</u>	<u>Sl_id</u>	<u>Qty</u>
100	200	120	2
100	300	120	1
100	400	121	1
200	200	120	1
200	330	190	3

The table Invoice is in 2NF since all non-key attributes are fully dependent on the primary key (Invoice Number).

Invoice (**Invoice number**, Customer_number, Customer_name, Customer_address, Invoice_date, Invoice_amount)

Invoice relation

<u>In_No</u>	<u>C_no</u>	<u>C_nm</u>	<u>C_add</u>	<u>In_dt</u>	<u>In_at</u>
100	150	Al	SAU	1/1/99	1000
200	180	Ema	SAU	2/2/00	350

The Third Normal Form (3NF)

Consider the following relation

Invoice (**Invoice number**, Customer_number, Customer_name, Customer_address, Invoice_date, Invoice_amount)

Invoice relation

<u>In_No</u>	<u>C_no</u>	<u>C_nm</u>	<u>C_add</u>	<u>In_dt</u>	<u>In_at</u>
100	150	Al	SAU	1/1/99	1000
200	180	Ema	SAU	2/2/00	350

Customer_name is transitively dependent on Invoice number since
(Invoice_number → Customer_number) and
(Customer_number → Customer_name)

similarly,

Customer_address is transitively dependent on Invoice number since
(Invoice_number → Customer_number) and
(Customer_number → Customer_address)

Therefore, the relation is not in 3NF since Customer_address and Customer_name are not directly dependent on Invoice_number. This transitive dependence is an indication that the relation has information about more than one entity and should therefore be decomposed.

Customer (**Customer number**, Customer_name, Customer_address)

Customer Relation

C_no	C_nm	C_add
150	Al	SAU
180	Ema	SAU

Invoice (**Invoice number**, **Customer number**, Invoice_date, Invoice_amount)

Invoice

In_No	C_no	In_dt	In_at
100	150	1/1/99	1000
200	180	2/2/00	350

The following relations are all in 3NF:

Invoice (**Invoice number**, **Customer number**, Invoice_date, Invoice_amount)

Customer (**Customer number**, Customer_name, Customer_address)

Line_Item (**Invoice number**, **Product number**, Salesman ID, Quantity)

Product (**Product number**, Product_name, Unite_price)

Product relation

P_no	P_nm	Pirce
200	Chair	50
300	Bed	500
400	Tv	400
200	Chair	50
330	Stand	100

Invoice relation

In_No	C_no	In_dt	In_at
100	150	1/1/99	1000
200	180	2/2/00	350

Line_Item relation

In_No	P_no	Sl_id	Qty
100	200	120	2
100	300	120	1
100	400	121	1
200	200	120	1
200	330	190	3

Customer Relation

C_no	C_nm	C_add
150	Al	SAU
180	Ema	SAU

Exercises

1 – Consider the following table about courses. The primary key is Course code and Section Id.

Course Code	Section ID	Room ID	Instructor	course title	Credit Hrs
Bite254	1	D103	Nasser	Database	5
Bite254	2	D102	Mark	Database	5
Bite254	3	D101	John L	Database	5
Bite217	1	D210	John R	Programming	5
Bite217	2	D201	Ahmed	Programming	5

- Give examples of anomalies in the table above
- Is the above table in 1NF? and why ?
- Is the above table in 2NF? and why ?
- Is the above table in 3NF? and why ?

2 - Consider the following Project table where each employee where Emp_no and Start_date is the primary key of the relation. What Normalization rule(s) are violated by the table below? How would you change the table to eliminate those violations?

<u>Emp_No</u>	<u>Start Date</u>	Emp Name	Hours Worked	Emp_Grade	Rate per Hour
100	1-DEC-2001	MD	7	1	12
101	1-DEC-2001	Sam	8	2	12.5
102	1-DEC-2001	Ahmad	8	3	12.8
100	1-NOV-2000	MD	3	1	15.9
102	1-AUG-2000	Ahmad	2	3	12

3 - The following table is about a client information. The primary key is Client_Id. What Normalization rule(s) are violated by the table below? How would you change the table to eliminate these violations?

<u>Client_id</u>	C_name	Credit_lim	Sales_rep	Comm
100	Sali	12,500	MD	0.02
200	Adrian	10,000	MD	0.02
300	Alia	11,000	Jamil	0.03
400	Noor	12,400	Jamal	0.04
500	Emo	6,000	MD	0.02

4 - Consider the following table about Student information. The primary key is the Student ID. What Normalization rule(s) are violated by the table below? How would you change the table to eliminate these violations?

Stud_id	St_name	Club Memb 1	Club Memb 2	Club Memb 3
100	Amani	Ski Club	Drama Club	Tennis Club
200	Salwa	Drama Club		
300	Arwa	Tennis	Drama Club	
400	Salima			

All Shine Company

The following is a monthly sample report for All Shine company showing its projects and the employees who are working on them. You are required to use that report and the Normalization technique to construct the required normalized 3NF tables showing 1NF and 2NF tables in route.

Proj Code	Proj Name	Emp Number	Emp Name	Emp Job	Rate Per hour	Number Hours	Total Amount
1	Painting	101	Sali	Engineer	60	13	780.00
		102	Adam	Painter	30	12	360.00
		103	Al	Manager	100	13	1,300.00
		104	John	Engineer	60	15	900.00
		105	Ema	Painter	30	10	300.00
2	Tiling	106	Sal	Worker	45	10	450.00
		103	Al	Manager	100	11	1,100.00
		107	Noor	Worker	45	12	540.00
		108	Riya	Worker	45	13	585.00
3	Power Supply	104	John	Engineer	60	10	600.00
		101	Sali	Engineer	60	10	600.00
		108	Riya	Worker	45	10	450.00
		106	Sal	Worker	45	10	450.00
		107	Noor	Worker	45	11	495.00
		103	Al	Manager	100	15	1,500.00

Book Request



Request No.: **343345**
Date: **June 18, 1999**

Course Id.	Course Name	Instructor Id.	Instructor Name
------------	-------------	----------------	-----------------

INFO 260 DBMS II

492 Mr. Omar

ISBN	Title	Edition	Author
123 33 121	Normal Databases	1st	Shakespeare

Publisher No.	Publisher Name	No. of Copies
123	Motivate	12

ISBN	Title	Edition	Author
143 44 234	Fly By Database	3rd	Terminator II

Publisher No.	Publisher Name	No. of Copies
123	Motivate	14

ISBN	Title	Edition	Author
------	-------	---------	--------

Publisher No.	Publisher Name	No. of Copies
---------------	----------------	---------------

At the above college, each instructor fills out a text book request form for the courses they teach. Above is an example of a request. Each course has one request filled out by the instructor for that semester. There is room for more than one text book on the form if a course has several required texts. The number of texts ordered may be different from the number of students in the class because some students already have the text.

A course can have many requests because the course runs every semester and there may be a request filled out every semester.

Normalize the data on the Request Form to the third normal form (3NF). Clearly indicate table contents in EACH STEP of the process. Identify the Primary Key by underlining the attribute.