

# FACTFILE: GCE SOFTWARE SYSTEMS DEVELOPMENT

## (vii) NORMALISATION

### </> Normalisation

#### Learning outcomes

Students should be able to:

- Define, analyse and apply normalisation including:
  - first normal form;
  - second normal form;
  - third normal form;

#### Content

##### Role of Normalisation

The analyst may also use normalisation to confirm the database design as it may be difficult to discover all the entities and attributes using just ER Modelling. Normalisation is a formal bottom up process. It begins by examining the data to be stored by looking at documentation and forms used by an organisation. It involves the application of a series of rules, in steps, to remove undesirable characteristics and problems.

It tends to be more useful for simple databases but can be used to test the correctness of an ER Model. It is complementary. Normalisation aims to minimise data redundancy, storage space, and update anomalies such as insertion, deletion and modification. It also improves the integrity of the data. In some cases full normalisation is not desirable and designers may recombine tables to improve query performance.

The case study, Caroline's Cosmetics, (Entity Relationship (ER) Models) suggested many items that could be stored in the database, and many reports, stickers and labels that could be produced from that information. Normalisation of the PIP (Product Improvement Programme) report shown below will now be performed. Each batch of product has its own PIP report.

<b>Caroline's Cosmetics</b>			
<b>PIP Report</b>		<b>Date: 20/2/2014</b>	
<b>Batch</b>		<b>Product</b>	
Batch No	4	Product Name	Night Night
Date of Manufacture	13/02/2014	Description	Uses Lavender and Chamomile essential oils to calm and relax the mind and body.
No Produced	28		
Average Cost (£)	2.00		
<b>Ingredients</b>			
<b>MaterialID</b>	<b>SupplierID</b>	<b>CommonName</b>	<b>Weight (g)</b>
1	S34	Citric Acid	1000.45
2	S4	Baking Soda	3000.10
3	S1	Almond Oil	310.10
4	S4	Chamomile Essential Oil	350.02
5	S7	Lavender Essential Oil	350.06
		<b>TOTAL WEIGHT (g)</b>	<b>5010.73</b>

Table 7.1 – PIP report for Caroline's Cosmetics



## Unnormalised Form (0NF)

<b>Unnormalised form</b>	<b>BATCH</b> ( <u>BatchID</u> , DateOfManufacture, NoProduced, <u>ProductName</u> , ProductDescription, {MaterialID, SupplierID, CommonName, Weight})
--------------------------	---

ID stands for number or ID. Primary keys are underlined. Foreign keys are represented with \*. Repeating groups are contained within {}. Students may use different notation. Derived attributes such as total weight and average cost should technically not be included here but a design decision may be taken at a later stage to include them if it improves performance.

This data is placed in a tabular format (unnormalised) and data added from another PIP Report to allow further understanding. From this table the key attribute can be identified as BatchID and the repeating group can also be easily identified. This table violates the rules for the relational data model as there is more than one value in each cell, i.e. in the repeating group.

BATCH Table					Repeating Group			
BatchID	Date of Manufacture	No Produced	Product Name	Product Description	MaterialID	SupplierID	CommonName	Weight
4	13/02/2014	28	Night Night	Uses Lavender and Chamomile essential oils to calm and relax the mind and body.	1	S34	Citric Acid	1000.45
					2	S4	Baking Soda	3000.10
					3	S1	Almond Oil	310.10
					4	S4	Chamomile Essential Oil	350.02
					5	S7	Lavender Essential Oil	350.06
5	13/02/2014	30	Butter Smooth	Contains Shea butter and almond oil leaving the skin super smooth.	1	S34	Citric Acid	2500.08
					2	S4	Baking Soda	5010.06
					3	S1	Almond Oil	500.04
					7	S45	Shea Butter	500.06

Table 7.2 – Batch Table in Unnormalised Form (0NF)



## First Normal Form (1NF)

**Definition:** A relation is in first normal form if the domain of each attribute contains only atomic values, and the value of each attribute contains only a single value from that domain, i.e. there are no repeating groups.



### Remove Repeating Group

1<sup>st</sup> Normal Form [Remove Repeating Groups – in {} into a new table along with a copy of the primary key from the original table]

**BATCH**(BatchID, DateOfManufacture, NoProduced, ProductName, ProductDescription)  
**INGREDIENT**(BatchID\*, MaterialID, SupplierID, CommonName, Weight)

The new table is called **INGREDIENT**.

BATCH Table				
BatchID	Date of Manufacture	NoProduced	ProductName	Product Description
4	13/02/2014	28	Night Night	Uses Lavender and Chamomile essential oils to calm and relax the mind and body.
5	13/02/2014	30	Butter Smooth	Contains Shea butter and almond oil leaving the skin super smooth.

INGREDIENT Table				
BatchID	MaterialID	SupplierID	CommonName	Weight
4	1	S34	Citric Acid	1000.45
4	2	S4	Baking Soda	3000.10
4	3	S1	Almond Oil	310.10
4	4	S4	Chamomile Essential Oil	350.02
4	5	S7	Lavender Essential Oil	350.06
5	1	S34	Citric Acid	2500.08
5	2	S4	Baking Soda	5010.06
5	3	S1	Almond Oil	500.04
5	7	S45	Shea Butter	500.06

Link

Table 7.3 – Batch Table and Ingredient Table in First Normal Form (1NF)

The repeating groups have been removed from the BATCH table and placed into a new table called INGREDIENT with a **composite** primary key of both BatchID and MaterialID. Both BatchID and MaterialID are required to uniquely identify a row in the INGREDIENT table. For example if the BatchID is 4 then this would lead to several rows for this table. If the MaterialID was 2 then this would lead to one of 2 rows in the table. Therefore, both BatchID **and** MaterialID are needed, for example 4 and 2, to identify one particular/ unique row in the INGREDIENT table.

Note that a copy of BatchID is retained in the BATCH table. BatchID in the INGREDIENT table is known as a foreign key that links the two tables together.



## Second Normal Form (2NF)

**Definition:** In order to be in Second Normal Form, a relation must first fulfil the requirements to be in First Normal Form. Additionally, each non-key attribute in the relation must be functionally dependent upon the primary key.

We only need to examine the tables that have a composite primary key. Any table that has a primary key consisting of one attribute only is automatically in 2NF.



### Remove Partial Dependency

2<sup>nd</sup> Normal Form [Remove Partial Dependencies].

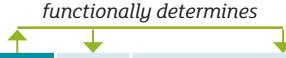
BATCH(BatchID, DateOfManufacture, NoProduced, ProductName, ProductDescription)  
 INGREDIENT(BatchID\*, MaterialID\*, Weight)  
 MATERIAL (MaterialID, SupplierID, CommonName)

The BATCH table was already in second normal form as it did not have a composite primary key. The INGREDIENT table has a composite key and a partial dependency was identified and removed. SupplierID and CommonName are only dependent on MaterialID. BatchID is not involved. SupplierID and CommonName are removed to a new table MATERIAL along with a key field MaterialID.

Looking at the original INGREDIENT table, in first normal form, it can be seen that part of its composite primary key, MaterialID, can be used by itself to determine one unique SupplierID and one unique Common Name. For example the Material with an ID of 2 is always from Supplier S4 and has a common name of Baking Soda. The same applies for the other materials in the table. MaterialID can be said to functionally **determine** SupplierID and CommonName (or SupplierID and CommonName are **functionally dependent** on MaterialID). It does not however lead to one single value for weight. There is a **partial dependency** between MaterialID and SupplierID and CommonName. BatchID in the composite key is not required.

**INGREDIENT Table**

*functionally determines*



BatchID	MaterialID	SupplierID	CommonName	Weight
4	1	S34	Citric Acid	1000.45
4	2	S4	Baking Soda	3000.10
4	3	S1	Almond Oil	310.10
4	4	S4	Chamomile Essential Oil	350.02
4	5	S7	Lavender Essential Oil	350.06
5	1	S34	Citric Acid	2500.08
5	2	S4	Baking Soda	5010.06
5	3	S1	Almond Oil	500.04
5	7	S45	Shea Butter	500.06

Table 7.4 - Ingredient Table showing functional dependencies

To remove this partial dependency the attributes that are determined by the MaterialID from the Ingredient table are removed - SupplierID and CommonName. These are placed in a new table with a *copy* of their **determinant**, MaterialID, which is the primary key. As MaterialID is copied across to the new table it acts as a foreign key in the INGREDIENT table linking the two tables together.

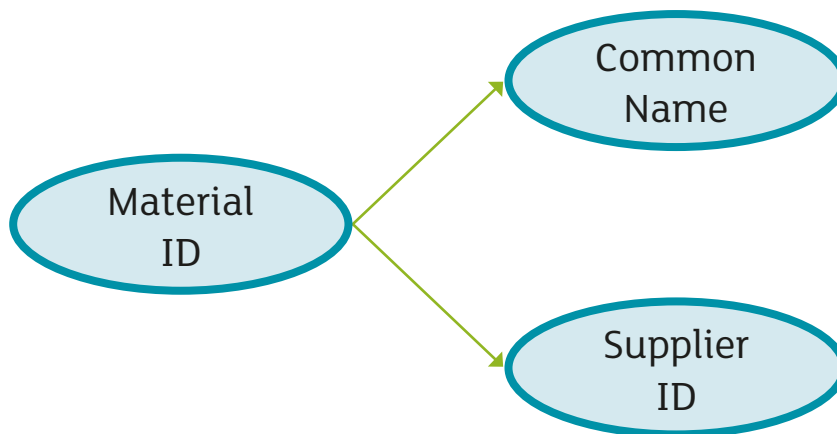


Figure 7.1 – MaterialID determines the SupplierID and the Common Name

**INGREDIENT Table**

BatchID	MaterialID	Weight
4	1	1000.45
4	2	3000.10
4	3	310.10
4	4	350.02
4	5	350.06
5	1	2500.08
5	2	5010.06
5	3	500.04
5	7	500.06

Link

**MATERIAL Table**

MaterialID	SupplierID	CommonName
1	S34	Citric Acid
2	S4	Baking Soda
3	S1	Almond Oil
4	S4	Chamomile Essential Oil
5	S7	Lavender Essential Oil
7	S45	Shea Butter

Table 7.5 - INGREDIENT Table in 2NF showing new MATERIAL Table and the link between them

Data redundancy has been reduced. This reduces both storage space and the problem of update anomalies.

## </> Third Normal Form

For a database to be in third normal form, the following rules have to be met,

- It is already in 2NF.
- There are no non-key attributes that depend on another non-key attribute i.e. remove any transitive dependencies.



### Remove Transitive Dependency

3<sup>rd</sup> Normal Form [Remove Transitive Dependencies].

**BATCH**(BatchID, DateOfManufacture, NoProduced, ProductName\*)  
**PRODUCT**(ProductName, ProductDescription)  
**INGREDIENT**(BatchID\*, MaterialID\*, Weight)  
**MATERIAL**(MaterialLID, SupplierID, CommonName)

A transitive dependency has been removed from the BATCH table creating a new PRODUCT table. Note that there are 4 tables and 3 foreign keys linking these tables.

Each table is examined for transitive dependencies. A transitive dependency is where a field is dependent on another field that is not the primary key. In tables which only have one attribute apart from the primary key there are no transitive dependencies. The only possibilities are therefore the BATCH and MATERIAL Table.

Consideration of the BATCH table shows that all fields relate to the BATCHID except for ProductDescription which is unique to the ProductName. This is a transitive dependency as the primary key is not involved. ProductDescription is removed to a new table, PRODUCT, along with ProductName as the key field.

Consideration of the MATERIAL table, the Supplier ID and Common Name have no dependency on each other. There are therefore no transitive dependencies in the MATERIAL Table. It is already in third normal form.

### BATCH Table

BatchID	Date of Manufacture	NoProduced	ProductName	Product Description
4	13/02/2014	28	Night Night	Uses Lavender and Chamomile essential oils to calm and relax the mind and body.
5	13/02/2014	30	Butter Smooth	Contains Shea butter and almond oil leaving the skin super smooth.

*functionally determines*

Table 7.6 - Batch Table (in 2NF) showing a transitive dependency exists between ProductName and ProductDescription



To remove the transitive dependency between ProductName and ProductDescription a new table called PRODUCT is created with ProductName and ProductDescription as the attributes. The determinant, ProductName, is selected as the primary key of the new table. A copy of the ProductName is left in the BATCH table as a foreign key and the ProductDescription is removed.

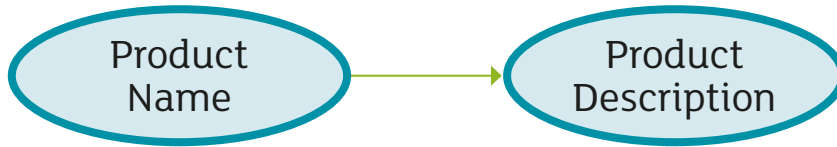


Figure 7.2 – ProductName determines the ProductDescription

**BATCH Table**

BatchID	DateOf Manufacture	NoProduced	ProductName
4	13/02/2014	28	Night Night
5	13/02/2014	30	Butter Smooth

**Link**

**PRODUCT Table**

ProductName	ProductDescription
Night Night	Uses Lavender and Chamomile essential oils to calm and relax the mind and body.
Butter Smooth	Contains Shea butter and almond oil leaving the skin super smooth.

Table 7.7 - BATCH Table split to produce the new PRODUCT Table

Compare the tables obtained using normalisation with that obtained with ER modelling.



## Normalised PIP Report showing tables in 3NF

### BATCH TABLE

BatchID	DateOfManufacture	NoProduced	ProductName
4	13/02/2014	28	Night Night
5	13/02/2014	30	Butter Smooth

### INGREDIENT TABLE

BatchID	MaterialID	Weight
4	1	1000.45
4	2	3000.10
4	3	310.10
4	4	350.02
4	5	350.06
5	1	2500.08
5	2	5010.06
5	3	500.04
5	7	500.06

### MATERIAL TABLE

MaterialID	SupplierID	CommonName
1	S34	Citric Acid
2	S4	Baking Soda
3	S1	Almond Oil
4	S4	Chamomile Essential Oil
5	S7	Lavender Essential Oil
7	S45	Shea Butter

### PRODUCT TABLE

ProductName	ProductDescription
Night Night	Uses Lavender and Chamomile essential oils to calm and relax the mind and body.
Butter Smooth	Contains Shea butter and almond oil leaving the skin super smooth.

Table 7.8: Tables created after normalisation is completed to 3NF



## Questions

- 1** Normalise the following order form. Produce an ER model.  
How do the tables obtained using normalisation compare to those obtained in ER modelling?

Caroline's Cosmetics			
Order Form			
Order No:56 Order Date: 21/04/14			
Customer		Card	
Name	Barbara Babbin	Type	VISA
Address	34 Seaweed Cove	Number	6777 7777 7777
Tel No	028 90566677	Expiry Date	09/17
Email	barbara_babbin@hotmail.co.uk	Card Name	Mrs B Babbin
		CSC	778
Product Name	Quantity	Price (£)	Charge (£)
Skin Soother	1	3.50	3.50
Sleepy Time	2	2.50	5.00
Skin Cleanser	1	3.00	3.00
Romantic Rose	2	2.00	4.00
<b>TOTAL CHARGE (£)</b>			<b>15.50</b>

- 2** Examine each statement below about normalisation and state whether it is true or false.

<b>A</b>	Normalisation is a formal bottom-up data modelling process.	
<b>B</b>	As normalisation proceeds the relations are more vulnerable to update anomalies.	
<b>C</b>	If relations are in first normal form each cell contains more than one value.	
<b>D</b>	A partial dependency can only occur in a relation when the primary key is composite.	
<b>E</b>	A relation that is in 3 <sup>rd</sup> normal form is also in 2 <sup>nd</sup> and 1 <sup>st</sup> normal form.	
<b>F</b>	A design decision may be taken to de-normalise tables in the actual database in order to improve query performance.	

