FirstName	LastName	Language
Tom	LaPorta	Spanish
Jim	Sequeira	Spanish

The use of the two tables `Interpreter` and `Languages` gives all the functionality that one would get from the non-1NF tables `Interpreter` and `WhoInterprets`. And it does not suffer from the problem of redundancy and potential inconsistency.

In summary, ensuring that a table is in 1NF requires taking repeating groups, if any, putting them in a separate table, and linking them with the original table using a key column from the original table.

SECOND NORMAL FORM

Tables in the second normal form (2NF) satisfy the property

> *Must be in 1NF and, in addition, all columns not part of any candidate key must be dependent upon the whole key.*[15]

The 2NF is relevant in the context of tables that have multi-column keys. A 1NF table that does not have multi-column keys is automatically in 2NF. Ensuring that a table is in 2NF avoids redundancy in the context of multi-column keys. If some columns depend only on a part of a multi-column key, then to avoid redundancy these columns should be moved to a separate table along with a copy of the relevant column(s) in the key.

As an example illustrating the elimination of redundancy using the 2NF, consider the following table `Parts`:

PARTS

Part	ManufacturerId	ManufacturerName
Wheel	1	Zenith
Tire	2	GoodYear
Seat	3	PusaGate
HandleBar	1	Zenith
Spoke	1	Zenith
Brakes	4	Honeywell

[15] Specifying column B as being dependent (functionally dependent to be precise) upon column A is equivalent to saying that the values in column A determine the corresponding values in column B.

The key for table `Parts` is a two-column key consisting of the columns `Part` and `ManufacturerId`. Table `Parts` is not in 2NF because column `ManufacturerName` does not depend upon the whole key – it depends only upon the column `ManufacturerId`.[16] As a result, there is redundancy resulting from multiple occurrences of manufacturer names, for example, Zenith occurs multiple times. If Zenith is acquired, say by Alcatel-Lucent, then the name Zenith will have to be changed to Alcatel-Lucent in multiple places – not making all the changes will result in inconsistency.

To eliminate this redundancy, table `Parts` is split into the following two 2NF tables:

PARTMANUFACTURER

Part	ManufacturerId
Wheel	1
Tire	2
Seat	3
HandleBar	1
Spoke	1
Brakes	4

MANUFACTURER

ManufacturerId	ManufacturerName
1	Zenith
2	GoodYear
3	PusaGate
4	Honeywell

THIRD NORMAL FORM

Tables in the third normal form (3NF) satisfy the property

> *Must be in 2NF and, in addition, no non-key column must depend upon another non-key column (all non-key columns must directly depend only upon the key or every key if there are multiple keys).*

[16] Each manufacturer is assigned a unique id to avoid issues such as two companies having the same name.

Consider table Books, which we saw earlier, and which is in 3NF:

BOOKS

ISBN	Title	Price	Authors	Pages	PubDate	Qty
0929306279	Bell Labs	29.95	Gehani	269	2003	121
0929306260	Java	49.95	Sahni & Kumar	465	2003	35
0670031844	White Mughals	34.95	Dalrymple	459	2003	78
0439357624	Born Confused	16.95	Hidier	432	2002	11

Column ISBN is the key for the Books table and all other columns are non-key columns. Books is in 2NF because it is in 1NF (all column types are atomic) and because all non-key values in each row depend upon the single column key ISBN. Moreover, Books is in 3NF because no non-key value in a row depends upon another non-key value in that row.

Let us now consider table Books2, which is similar to Books, but has an extra column, the non-key column #BooksWritten specifies the number of books written independently by an author or jointly by a combination of authors:

BOOKS2

ISBN	Title	Price	Authors	#BooksWritten	…
0929306279	Bell Labs	29.95	Gehani	10	…
0929306260	Java	49.95	Sahni & Kumar	1	…
0670031844	White Mughals	34.95	Dalrymple	4	…
0439357624	Born Confused	16.95	Hidier	12	…
0929306155	Ada	33.95	Gehani	10	…
0929306171	C	39.95	Gehani	10	…

Table Books2 is in 2NF but not in 3NF because the non-key column #BooksWritten depends upon the non-key column Authors and not directly upon the primary key column ISBN. Consequently, table Books2 is not in 3NF. Note the redundancy with respect to the number of books written. The number of books written by Gehani appears three times. Any change in the number of books written by Gehani would require multiple (three at this time) updates. If all updates are not done, then this will result data inconsistency.

To bring table Books2 to 3NF, column #BooksWritten must be taken out and put, for example, in a new table BooksWritten along with a copy of the column named Authors.

BOOKSWRITTEN

Authors	#BooksWritten
Gehani	10
Sahni & Kumar	1
Dalrymple	4
Hidier	12

4. EXERCISES

1. What is the ER Model? How does it work?
2. Give definitions for the
   a. First normal form.
   b. Second normal form.
   c. Third normal form.
3. What are the advantages and disadvantages of using a normalized database design?
4. Write a relational algebra expression that produces a list of interpreters who can speak both Spanish and French using the two tables `Interpreter` (third version, page 85) and `Languages` (page 86).
5. The total invoice amount is not stored in table `Orders`. What are the advantages of storing it? What are the disadvantages of not storing it?
6. Why is the unit price stored in the `OrderInfo` table? Why can this price not be picked up from the `Books` table?
7. As mentioned in the discussion about entity set `Books` (page 76), it would be more appropriate to list each author individually (with the first and last names listed separately). This will facilitate searches and report generation. How will you do this? Justify your answer.
8. What are the pros and cons of the modification mentioned in Exercise 7 in terms of writing queries and their efficiency.
9. What is the impact of taking all the data in the four tables `Customers`, `Books`, `Orders`, and `OrderInfo` and storing them in one big table. Discuss your answers based on topics such as
   - storage space,
   - access time,
   - concurrency, and
   - redundancy of data.