Normalization Overview

by Sophia

WHAT'S COVERED

In this lesson, you will explore the importance of normalizing an ERD, in three parts. Specifically, this lesson covers:

1. Why Normalization Is Important
2. Reviewing the Normal Forms
3. Movie Ratings Example

1. Why Normalization Is Important

In the previous lesson, you learned why modeling a database prior to its actual construction is important. Normalization is an important step in that modeling.

Normalization is usually performed between the conceptual phase and the logical data modeling phase in a database development project. It's important to do it before getting into the physical creation of the database because normalization changes the structures and relationships among the entities, and those cannot be easily changed once the database is under construction—or worse yet, in actual use.

Normalization offers a number of significant benefits to a database, including the following:

- Minimizing redundancy: Normalization minimizes redundancy by preventing the need to store duplicate data. As you learned earlier, redundancy not only makes the database take up more storage space, but it can lead to data entry inconsistencies. Reducing the opportunities for inconsistencies to be introduced also improves data integrity.
- Facilitating updates: Normalization makes it easier to update and modify data without introducing anomalies. **Anomalies** are inconsistencies that may occur when there are redundancies or dependencies between non-key attributes. You will learn about those issues in upcoming lessons.
- Simplifying queries: Normalized databases often result in simpler and more efficient queries because the structure of the tables enables easier data retrieval and reduces the need for complex joins.
- Supporting scalability: A well-normalized database is generally more scalable. As the data grows, a normalized structure provides a solid foundation for extending the database schema without significant

modifications.

- Enhancing maintainability: Database maintenance is more straightforward in a normalized structure. Changes to the schema can be made more easily without affecting the entire system.
- Improving performance: Normalized structures often lead to smaller, more focused indexes that can speed up search operations.

For all of these reasons, normalization is crucial for designing efficient, maintainable, and scalable databases. It contributes to data integrity, reduces redundancy, and provides a solid foundation for effectively managing information.

TERM TO KNOW

Anomalies

Inconsistencies that may occur when there are redundancies or dependencies between non-key attributes.

2. Reviewing the Normal Forms

Now that you understand the importance of normalization, let's briefly review what you learned in the previous lesson about 1NF through 3NF, and also introduce two additional forms: 4NF and 5NF. The first three **normal forms** are first normal form (1NF), second normal form (2NF), and third normal form (3NF). From a table structure perspective, the second normal form (2NF) improves upon the first normal form (1NF), and the third normal form (3NF) improves upon the second normal form (2NF). Although there are higher normal forms, the third normal form (3NF) is typically the highest level necessary for most applications. You will learn a lot more about 1NF through 3NF in upcoming lessons.

There are some use cases for higher normal forms (such as 4NF, 5NF, and beyond), but they are not used as often in practice. High normal forms require more complex queries and may require trade-offs in terms of query performance and database maintenance. There are many cases in which the advantages of going beyond 3NF aren't worth the complexity added.

Certain scenarios, however, require higher normal forms, particularly in large and complex databases where data integrity and consistency are of utmost importance. Certain types of complex data structures can use 4NF and 5NF to deal with multivalued dependencies and join dependencies, respectively.

4NF deals with situations where there are multiple, independent sets of facts within a table. To achieve 4NF, you separate these sets of facts into their own tables. For example, suppose a table contains the columns CourseID, Instructor, and Textbook. The composite key is the combination of CourseID and Instructor. There is no way to enter multiple textbooks for a certain CourseID/Instructor combo without violating the need for the composite key to be unique in each entry. To achieve 4NF, you would split it into two tables: one with CourseID and Instructor, and the other with CourseID and Textbook.

A fifth normal form (5NF) is typically used when complex join dependencies exist, often involving more than one multivalued dependency. A classic example is a database of academic courses and prerequisites. A course

might have multiple prerequisites, and a single prerequisite may apply to multiple courses. There may be join dependencies between different sets of attributes in this case. By further decomposing the data into separate tables, join dependencies can be eliminated. Every set of attributes would be dependent only on the table's primary key.

As you work through the remaining lessons in this challenge, you will expand your knowledge of each of these normal forms by studying examples of each one and learning how to change a non-normalized database into a normalized one at each level of normal form. But before getting into that, let's consider why normalization is such an important topic.

TERM TO KNOW

Normal Forms

The series of stages or steps that the normalization process works through.

3. Movie Ratings Example

Let's go back to our movie rating database. If it had been set up in a spreadsheet, it would look something like this:

User	Movie Title	NumericRating	Textual Rating	Actor	Genre	ReleaseDate
Andy Joe	Toy Story	5	Such a classic!	Tom Hanks, Tim Allen	Animation, Adventure, Comedy	1995
Andy Joe	Titanic	4	Great movie!	Leonardo DiCaprio, Kate Winslet, Billy Zane, Kathy Bates	Drama	1997
Sophia McKenzie	Terminator 2: Judgment Day	5	Loved this movie, definitely a must watch!	Arnold Schwarzenegger, Linda Hamilton, Edward Furlong	Action, Science Fiction	1991
James Wang	Titanic	5	Best movie!	Leonardo DiCaprio, Kate Winslet, Billy Zane, Kathy Bates	Romance	1997
Barton Raftor	Titanic	3	It was OK, could be shorter.	Leonardo DiCaprio, Kate Winslet, Billy Zane, Kathy Bates	Drama, Romance	1997
Barton Raftor	Toy Story	4.5	Good movie! Watched a few times.	Tom Hanks, Tim Allen, Don Rickles, Jim Varney, Annie Potts, John Morris	Animation, Adventure, Comedy	1995

This spreadsheet has some issues that normalization can help correct. The actors listed for the movie *Toy Story* seem to be different, since different users entered them. The genre has the same issue. Notice that the movie *Titanic* is listed as a drama by one user, as a romance by another user, and as a drama and a romance by a third user. We will walk through the normalization process throughout the upcoming lessons to resolve these issues and define the final normalized database structure.

SUMMARY

In this lesson, you learned that database normalization is a process that organizes and structures data efficiently in a relational database. **Normalization is important** because it minimizes data redundancy and improves data integrity by preventing anomalies that may arise from denormalized or redundant storage. This is achieved by breaking down a database into multiple tables and applying a set of rules known as normal forms. This is to ensure that data is stored in a standardized and logical manner.

You **reviewed the most commonly used normal forms**, which are the first normal form (1NF), the second

normal form (2NF), and the third normal form (3NF). In 1NF, each column in a table contains atomic values, while each row is unique, eliminating repeating groups. With 2NF, all non-key attributes are fully functionally dependent on the entire primary key, eliminating partial dependencies. A 3NF ensures that there are no transitive dependencies between non-key attributes. It is often sufficient for most databases to achieve at least 3NF, although higher normal forms, such as 4NF or 5NF, are available for cases that are more complex. You learned in the **movie ratings example** that as a result of normalization, databases can be managed, accessed, and queried more easily, and inconsistencies or anomalies are reduced.

Source: THIS TUTORIAL WAS AUTHORED BY DR. VINCENT TRAN, PHD (2020) AND Faithe Wempen (2024) FOR SOPHIA LEARNING. PLEASE SEE OUR TERMS OF USE.

TERMS TO KNOW

Anomalies

Inconsistencies that may occur when there are redundancies or dependencies between non-key attributes.

Normal Forms

The series of stages or steps that the normalization process works through.