Normalising Databases

Class Notes

Normalisation is a formal process to structure a database in such a way that optimises the database and reduces data redundancy (that is, repeated data). Normalised databases are generally more efficient to run than those which are not - a very important consideration when dealing with large databases.

The process may involve adding fields or even whole tables, and comes in three steps.

First Normal Form (1NF)

A database is considered 1NF if each field contains a single value rather than a list of values. So, arrays or anything like arrays are not permitted. This includes having multiple fields that store the same type of information such as “FirstParent” and “SecondParent”, or “FirstCarNumberPlate” and “SecondCarNumberPlate”

Second Normal Form (2NF)

A database is considered 2NF if it’s 1NF *and* all the columns depend on the primary key. This means that all the fields in the database should describe the thing that the primary key identifies. If you have a table of cars and a field for the owner name, then that is not 2NF since the name of the owner does not describe the car.

Foreign keys are exempt from this rule. So, if you have a car table and there is a driver’s license number of the owner which allows you to look him or her up in another table, then that’s fine.

Third Normal Form (3NF)

A database is consider 3NF is it is 2NF *and* all the columns except the primary key are independent of each other.

Consider the car example again. If the car table stored both model and manufacturer, then it is not 3NF because the model is dependent on the manufacturer. Ford doesn’t have a Corolla - only Toyota does. What model you have is partially determined by the manufacturer of the car

As another example, imagine a table of patient data that stores height, weight, body mass index and a boolean for whether they are overweight or not. Whether someone is overweight or not is determined from the BMI, so the boolean is dependent on the BMI value, making the table not 3NF.

Furthermore, the BMI is calculated from the weight and height, making it dependent as well.

They’re Only Guidelines

Although it is ideal, it is not always possible or practical to have a fully 3NF database. Having weight, height and BMI in the same health data table *does* make sense, after all. However, if you find yourself with a table that does not abide by some of the rules above, make sure you have a good reason not to fix it.

It should also be pointed out that the higher the normal form, the more acceptable it is not to abide by it. There are a good few databases that aren’t 3NF, very few that aren’t 2NF and databases that are not 1NF should basically never happen.

Summary

A table is…

- First normal form if there are no arrays and no fields that store the same type of information as another field.
- Second normal form if it is first normal form and every field directly describes the thing represented by the primary key (not counting foreign keys).
- Third normal form if it is second normal form and no field depends on or is derived from any other field except the primary key.