In this thread I would like to discuss database normalization. I will define what normalization is. I will describe why database normalization is important. I will talk about the normal forms of database normalization. I will also discuss 1 to many relationships and how they are represented in Mongo DB. So, what is database normalization?

Database normalization is the process of organizing data into tables in such a way that the relationships between the data are clear, concise, and easy to understand. Normalization also eliminates duplicate data and ensures that data is consistent across the database. Normalized databases are more efficient and easier to maintain than non-normalized databases. The main reasons for normalizing a data base are to reduce but not eliminate redundancy yet doing so does eliminate anomalies for data insertion, updating data, and deletion anomalies. The latter reasons for normalizing a database are why it is so very important to do so. (1NF, 2NF, 3NF and BCNF in Database Normalization | Studytonight, n.d.) There are rules for database normalization. Let’s look at them below:

Normalization rules are divided into the following normal forms:

* First Normal Form
* Second Normal Form
* Third Normal Form
* BCNF
* Fourth Normal Form

For a table to be in the First Normal Form, it should follow the following 4 rules:

* It should only have single(atomic) valued attributes/columns.
* Values stored in a column should be of the same domain
* All the columns in a table should have unique names.
* And the order in which data is stored, does not matter.

For a table to be in the Second Normal Form:

* It should be in the First Normal form.
* And, it should not have Partial Dependency.

A table is said to be in the Third Normal Form when,

* It is in the Second Normal form.
* And, it doesn't have Transitive Dependency.

(1NF, 2NF, 3NF and BCNF in Database Normalization | Studytonight, n.d.)

A table is said to be in Boyce-Codd Normal Form when,

* BCNF is really an extension of 3rd Normal Form (3NF). For this reason, it is frequently termed 3.5NF. 3NF states that all data in a table must depend only on that table’s primary key, and not on any other field in the table. At first glance BCNF and 3NF are the same thing. However, in some rare cases it does happen that a 3NF table is not BCNF-compliant. This may happen in tables with two or more overlapping composite candidate keys.

(Techopedia, 2017)

A table is said to be in the Fourth Normal Form when,

* It is in the Boyce-Codd Normal Form.
* And, it doesn't have Multi-Valued Dependency.

(1NF, 2NF, 3NF and BCNF in Database Normalization | Studytonight, n.d.)

“One to many relationships are represented in Mongo DB by embedding connected data in a single document as to reduce the number of read operations required to obtain data. In general, you should structure your schema, so your application receives all of its required information in a single read operation. The advantage of embedding over referencing if you need to view many data entities in context of another.”

(Model One-to-Many Relationships With Embedded Documents — MongoDB Manual, n.d.)

References:

*1NF, 2NF, 3NF and BCNF in Database Normalization | Studytonight*. (n.d.). Retrieved November 1, 2022, from <https://www.studytonight.com/dbms/database-normalization.php>

Techopedia. (2017, June 20). *Boyce-Codd Normal Form (BCNF)*. Techopedia.com. Retrieved November 1, 2022, from <https://www.techopedia.com/definition/5642/boyce-codd-normal-form-bcnf>

*Model One-to-Many Relationships with Embedded Documents — MongoDB Manual*. (n.d.). Retrieved November 1, 2022, [from https://www.mongodb.com/docs/manual/tutorial/model-embedded-one-to-many-relationships-between-documents/](from%20https:/www.mongodb.com/docs/manual/tutorial/model-embedded-one-to-many-relationships-between-documents/)