

CST 363
Week 4
Chapter 5 Questions
Aaron Gordon

4. What is denormalization?

Denormalization is the process of intentionally leaving a table in non-normalized form for security, performance or ease of use reasons. Although normalization prevents data update problems, there are times when the performance cost or difficulty of use associated with joining multiple tables outweighs the risk of data issues, particularly when said risk is extremely low, as in the case of data that is effectively immutable.

6. Explain the problems that unnormalized tables have for insert, update, and delete actions.

An unnormalized table will have data duplicated across rows. This means that more data than is strictly necessary is added to the table on insert, that updating data in one record and not in other records where the data appears will result in inconsistent data and that deleting a record may unintentionally delete data from a second entity incorrectly associated with the first.

7. Explain how the representation of weak entities differs from the representation of strong entities.

A weak entity is modeled by including in its relation a foreign key to another entity, possibly as part of the relation's primary key. Often, DELETE rules are established for a weak entity such that all weak entities associated with a strong entity via a foreign key are deleted when the strong entity is deleted.

8. Explain how supertype and subtype entities are transformed into tables.

Subtype attributes are functionally dependent upon the primary key of the supertype, thus the supertype's primary key is not only included in the table that models the subtype but is a candidate key for that table. The natural, and correct, approach to modeling a subtype then is simply to use the primary key of its supertype as both a foreign key to refer to that supertype and as the primary key of the subtype table.

If a supertype has multiple, mutually exclusive subtypes, it may be necessary to include in the supertype's relation an attribute denoting the subtype of a given tuple.

14. Define the terms parent and child as they apply to tables in a database design and give an example of each.

A child table is a table which references a parent table, by means of a foreign key into the parent table. A COMMENTS table in a database used for an online blog might reference an ArticleID field from an ARTICLE table, associating each comment entity with the article it comments on. In this scenario, the COMMENTS table is the child table and the ARTICLE table the parent table.

Parent-child tables are used to model one-to-many relationships and, by extension, used to create the intersection tables used to model many-to-many relationships.

21. Explain the meaning of the term intersection table.

An intersection table is a child table whose records contain the primary keys from two or more parent tables. In this way, each record models an association between two or more other entities via a set of one-to-many relationships and so creates a many-to-many relationship.