

Functional Dependence Diagrams and Constraints Tutorial

CSU34041
Yvette Graham, TCD
ygraham@tcd.ie

Functional Dependency Diagrams, Normalised Relations, and Constraints

Tutorial

CSU34041
Yvette Graham, TCD
ygraham@tcd.ie

Football League Database

A football league is composed of four divisions (1,2,3,4) each containing many different teams, where each team has a unique name (e.g. Manchester united, Sheffield Wednesday etc.), and the number of points each team has amassed so far in the league competition. Information is also stored concerning each team's captain, and home ground address. The names of all professional footballers (not unique) in the league and the teams for which they play is stored in the database. Also, maintained in the database for each player is a record of their ability in the following positions (goalkeeper, defender, midfielder, striker) in a football team. A player can have the ability to play in more than one position. For example, a footballer named Alan Shearer, position = 'Defender', expertise = 2, however for position = 'Striker', expertise = 11, values > 0).

- (a) Draw a Functional Dependence diagram for the relations in the above database stating any assumptions made in your solution.
- (b) Derive a set of fully normalised relations (i.e. table names and attributes) from the dependence diagram, indicating clearly the primary key (candidate identifier(s)) and foreign keys of each relation.
- (c) Write down any constraints on column values you may deem important.

(A) Functional Dependency Diagram Methodology

1. Underline all of the nouns in the description
2. Create a spreadsheet (or similar on paper) of example data where each column is named by one of the nouns identified in (1)
3. Check if there are any problems with identifying any of the nouns, hint: nouns with non unique values and add an identifier column for it, e.g. *clientID* if the problematic noun was *client*.
4. Check if any *other columns* should correspond with that new identifying column and change the example data where appropriate.
5. Go through each column and ask the question: "Does any other column(s) determine my value? The answer can be
 - a. "no",
 - b. "yes, the ... column determines my value" or
 - c. "yes , the columns ... and ... determine my value"
6. Write the name of the determining column(s) within a single box with an arrow pointing to the name of the other column (boxed on its own)

(B) Normalised Relations Methodology

1. Using the functional dependency diagram from (A), identify all of the boxed attribute names that determine something else in the diagram, e.g. clientID determines clientname
2. Work through each to create a relation or table with the primary key being the boxed attribute, B, e.g. the existence of clientID should indicate the need for a client table with clientID as the primary key.
3. Add all determined from B as attributes in that table.
4. Do the same for all other boxed attributes identified in (1) above, underlining the primary keys.
5. Add a note to each table to say which attributes are foreign keys of others.

(C) Constraints Methodology

1. Read through the question again but paying special attention to the sentences that add a constraint to the data in some way, e.g. if a client belongs to one of 5 user groups, a sensible constraint to add is that the values of Usergroup attribute must be the values, 1-5.
2. Make a list of all constraints that should be applied.