SOFTENG 351: Fundamentals of Database Systems - Guide to Test 1

Michael Kemp

April 10, 2017

Contents

1 Basics of the relational model of data

1.1 The relational model of data

- *relations* are sets of *tuples* often represented as a table

- *attributes* are the column titles of a relation

- for each attribute we assign a *domain* which is a universal set containing all possible values(like a string; $dom(A) = string$

- *tuples* are the rows of a relation and all have the same structure in a relation

- if there is no value for an attribute then the value is *null*

- *relation schema* are a finite set R where attributes are A and each attribute $A \in R$ has a domain $dom(A)$

- relation schema can be written $R = \{A_1, A_2...A_n\}$ or $R(A_1, A_2...A_n)$ or $R(A_1 : dom(A_1)...A_n : dom(A_n)$

- All *R-tuples* (a tuple in a relation schema) are an element t of the Cartesian product of the domains of all the attributes $t \in A_1 \times A_2 \times ...A_n$ because each attribute's value is bound to it's respective domain.

- *R-relations* are a finite set r of R-tuples thus $r \subseteq dom(A_1) \times ...dom(A_n)$

- R-tuples can be written with their values $t = (A_1 : v_1...A_n : v_n)$

- A *database-schema* is a finite set S of relation schemata

- An S-database I consists of one R-relation for $I(R)$ for each relation R in S ($I = \{I(R)|R \in S\}$)

- Having duplicates in a database is normally useless so we have *keys* to ensure a uniqueness over an attribute or a combination of

- *superkey* over a relation schema R

  - finite, non-empty subset $K \subseteq R$
  - is *satisfied* if an R-relation r only has tuples with a unique combination of values for each attribute in the superkey.
  - A *Key* is a superkey if there is no subset which is also a satisfied superkey
  - A *foreign key* is when all of the combination of values of attributes(in the foreign key) is in the set of the table which defines the foreign key($[A_1..A_n] \subseteq S[A_1..A_n]$). Also the same S can not be referenced twice.

1.2 SQL as a data definition and query language

2 Query Languages

2.1 Relational algebra

tell the author to stop procrastinating...

2.2 Relational calculus

tell the author to stop procrastinating...

2.3 SQL

tell the author to stop procrastinating...

3 Database design

3.1 Entity-Relationship modelling

tell the author to stop procrastinating...