CMPT 308L-111: Lab 2

Professor Labouseur

Graham Burek

February 2, 2016

Types of Keys

Explain the distinctions among the terms primary key, candidate key, and superkey.

Superkey: Any column or set of columns that uniquely identify every row.
Candidate Key: A minimal superkey. It uses the fewest number of columns needed to uniquely identify every row.
Primary Key: A candidate key declared primary.
Foreign Key: The value(s) in one table that must match the primary key of another table.
Primary keys and candidate keys are minimized superkeys, and foreign keys create relationships between different tables.

Data Types

Write a short essay on data types. Select a topic for which you might create a table. Name the table and list its fields (columns). For each field, give its data type and whether or not it is nullable.

In SQL database systems, data types are declared for all the columns in a given table. Data types can represent all kinds of data, such as words (`TEXT`, `CHAR`), currency (`DECIMAL`, `NUMERIC`), times (`TIME`, `TIMESTAMP`) dates (`DATE`, `DATETIME`). A column that is declared to be a certain data type must have values whose format conforms to that of the data type. Data types are useful because they standardize (restrict) the kind of data that can be entered into any fields in a database. If data is standardized, it becomes much easier to analyze it and present it to users. Additionally, data types help prevent ambiguity in database fields. Further constraints can also be placed on data types for a field. For example, a field with a data type of `TEXT` could be declared `NOT NULL` in order to prevent any values in that field from being left empty.

A topic that I might create a table for is a database of video games. One of the tables would be named `Titles` and would given information related to different video game titles. The fields would be:
1. Title - `TEXT, NOT NULL`
2. Platform - `TEXT, NOT NULL`
3. MinimumAge - `INT, NOT NULL`
4. Genre - `TEXT, NOT NULL`

Relational Rules

Explain the following relational rules with examples and reasons why they are important.
a. The first normal form rule
b. The access rows by content only rule
c. The all rows must be unique rule

Relational Rules continued on next page. . .

The first normal form rule: The intersection of any row or column can only be one instance of a data type (atomic). For example, a video game title in the `Titles` table above has more than one platform it runs on. Instead of inserting two values into a single cell, a table that conforms to the ”first normal form” rule would create a separate table that deals with the relationships between a particular title and the platforms it runs on. The tables could then be linked using a foreign key.

The access rows by content only rule: Because the order of tuples in a database is not guaranteed, is it important to access tuples based on what they contain, not where they are in a database. For example, if the third row in the `Titles` table contained a record about Super Mario, then it would be better to access the record by querying for the title (”Super Mario”) of the record then by querying for the row number (3).

The all rows must be unique rule: This means that all the rows in the database must be unique. Having duplicate rows would only create the possibility for data inconsistency. For example, a game with two different platforms in the `Titles` table should follow the first normal form rules instead of creating two records in the same table.