$-3 = \frac{15x}{15} = -45 = x$ $\qquad \frac{-40}{32} = -1.25$

Database Ch. 3 - notes

The Relational Model and Normalization

I. Relational Model Terminology

- all relations are tables, but not all tables are relations

Relation: a special case of a table

Key Terms: Relations
relation

Functional Dependency: value of 1 or more attributes determines the ~~of~~ value of another attribute

Determinant: ~~the value of 1~~ attribute that determines another (Ex. A x B) A is determinant

Candidate Key: a determinant that determines all the other columns in a relation

Composite Key: Keys w/ 2 or more columns

Primary Key: column w/ unique values used to id rows in a table

Surrogate Key: key added to a table to serve as a PK (only purpos

Foreign Key: a column or composite of columns that's the PK of a table other than the one in which it appears

Referential Integrity constraint: a statement that limits the values of the foreign key by adding exist constraints

Normal Form: a set of conditions for a given relation

Multivalued Dependency: an anomaly that could arise when a determinant is matched w/ a particular set of values

Characteristics of Relations

- Rows contain data about an entity
- Columns contain data about attributes of the entities
- All entries in a column are of the same kind (if R1,C1 = FN, then R2,C2 FN
- Each column has a unique name
- Cells of the table hold a single value
- The order of the columns is unimportant
- The order of the rows is unimportant
- No two rows may be identical

entity: some identifiable thing (order, person, food, etc.)

domain: grouping of data that meets a specific type condition (Ex. FirstName)

Alternative Terminology

- columns = attributes or field
- rows = tuples or record
- table & relation are pretty much used interchangibly
  - file = table too

- every relation has a defined PK

Functional Dependencies

- heart of db design process

Ex.1. cookies cost $5.00
- how much you pay depends on how many boxes you buy

CookieCost = NumBoxesBought · $5.00
(NumBoxesBought → determinant)

Ex.2 color determines shape

red	→	ball
blue	→	cube

composite FD: more than 1 attribute is needed to determine another attribute

if (A, B) → C, then A nor B will determine C by itself
if C → (A, B), then C → A and C → B (union rule)

★ "Does any column determine the value of another column?" ★

Ex. (OrderNumber, SKU) → (Quantity, Price, ExtPrice)
- given a particular order number and SKU, there is only 1 qty, price, & ext.price

- determinants are unique in a relation only if it determines every other column in the relation

Keys

Key: a combo of 1 or more columns that're used to identify particular rows in a relation

- candidate keys identify a unique row in a relation
  - given a candidate key's value, we can find 1 specific row in the relation that has that value

- surrogate keys are usually hidden from reports b/c they're meaningless to users

- foreign keys express relationships b/w rows of tables

Modification Anomalies

- Deletion: when a row of data is removed, we lose facts about other data
- Insertion: may have to enter 2 pieces of info rather than just 1, depending on the relation

- Update: when an update occurs and it changes other data

Short History of Normal Forms (inventor: Codd)

1NF: a set of conditions for a relation, and has a defined PK

2NF: fixed the modification anomalies from 1NF
- 3NF fixed even more
- Boyce-Codd NF, same thing

- 4NF & 5NF will be discussed later

Normalization Categories

Normalization theory can be divided into:
1) anomalies from functional dependencies (2NF, 3NF, BCNF)
2) " " from multivalued dependencies (4NF)
3) " " from data constraints & odd conditions (esoteric)

- a relation is in 2NF iff it's in 1NF and all non-key attributes are determined by the entire PK
  - single-attribute PK's are automatically 2NF

- relation in 3NF iff in 2NF are no non-key attributes determined by another non-key attribute

Transitive Dependency: non-key attr. determined by another non-key attr.

- relation in BCNF iff in 3NF and every determinant is a candidate key

Eliminating Anomalies from Functional Dependencies w/ BCNF

- fix: design tables so that every determinant is a candidate key
  - "Step-by-Step" method to "Straight-to-BCNF" method

* Figure 3-19

- goal of normalization process is to create relations that are in BCNF. When we normalize to BCNF, anomalies are eliminated

* Study normalization examples in book (pgs 158-164) - 5 ex.

Eliminating Anomalies from Multivalued Dependencies

* occurs when a determinant is matched w/ a particular set of values
* multv. dep. are shown w/ $\Rightarrow A \rightarrow\rightarrow B$

- the determinant of the multiv. dep. can never be the PK
- hard to find

~~5th NF~~

5 NF

- Project-Join NF

- a table can be split apart, but not put back together correctly
- complex conditions

Domain / Key NF

- origin: 1982 paper posted by Ronald Fagin

- no anomalies
- all data constraints should be logical implications of domains & keys
- every determinant of a functional dependency must be a candidate key