ENSF 608: Understanding and Mapping the Relational Data Model

Dr. Emily Marasco

Fall 2021

Textbook: Fundamentals of Database Systems, 7th Ed., Elmasri & Navathe

Lesson Content

- Relational model concepts and constraints
- Relational database schemas
- Update operations and dealing with constraint violations
- ER-to-Relational mapping algorithm
- Mapping EER model constructs to relations

Relational Model Concepts

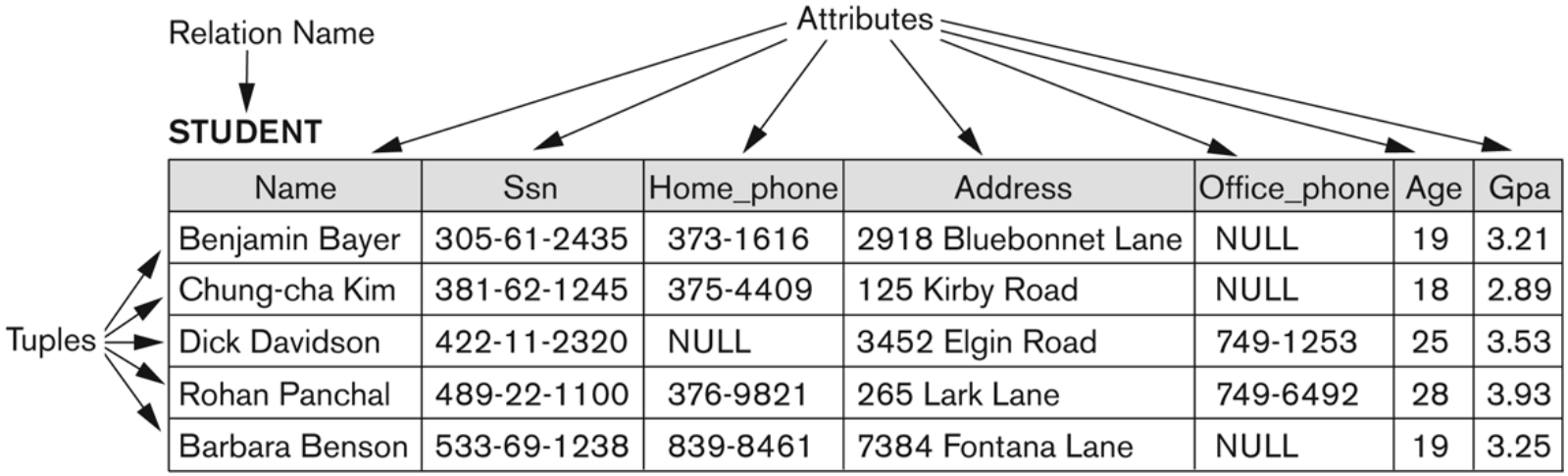
- The model was first proposed by Dr. Codd of IBM Research in 1970: "A Relational Model for Large Shared Data Banks," Communications of the ACM, June 1970
- Based on the concept of a **Relation**
  - The strength of the relational approach to data management comes from the formal foundation provided by the theory of relations
- We review the essentials of the **formal relational model** in this topic
- In **practice**, there is a **standard model** based on SQL – this is described in Chapters 6 and 7 as a language
- **Note:** There are several important differences between the **formal** model and the **practical** model, as we shall see

Informal Definitions (1 of 2)

- Informally, a **relation** looks like a **table** of values.
- A relation typically contains a **set of rows.**
- The data elements in each **row** represent certain facts that correspond to a real-world **entity** or **relationship**
  - In the formal model, rows are called **tuples**
- Each **column** has a column header that gives an indication of the meaning of the data items in that column
  - In the formal model, the column header is called an **attribute name** (or just **attribute**)

Example of a Relation

Figure 5.1 The attributes and tuples of a relation STUDENT.



Name	Ssn	Home_phone	Address	Office_phone	Age	Gpa
Benjamin Bayer	305-61-2435	373-1616	2918 Bluebonnet Lane	NULL	19	3.21
Chung-cha Kim	381-62-1245	375-4409	125 Kirby Road	NULL	18	2.89
Dick Davidson	422-11-2320	NULL	3452 Elgin Road	749-1253	25	3.53
Rohan Panchal	489-22-1100	376-9821	265 Lark Lane	749-6492	28	3.93
Barbara Benson	533-69-1238	839-8461	7384 Fontana Lane	NULL	19	3.25

Informal Definitions (2 of 2)

- Key of a Relation:
  - Each row has a value of a data item (or set of items) that uniquely identifies that row in the table
    - Called the **key**
  - In the STUDENT table, SSN is the key
  - Sometimes row-ids or sequential numbers are assigned as keys to identify the rows in a table
    - Called **artificial key** or **surrogate key**

Formal Definitions - Schema

- The **Schema** (or description) of a Relation:
  - Denoted by $R(A_1, A_2, ...A_n)$
  - *R* is the **name** of the relation
  - The **attributes** of the relation are $A_1, A_2, ..., A_n$
- Example:

  CUSTOMER (Cust-id, Cust-name, Address, Phone #)
  - CUSTOMER is the relation name
  - Defined over the four attributes: Cust-ID, Cust-name, Address, Phone #

CUSTOMER

Cust-ID	Cust-name	Address	Phone #

- Each attribute has a **domain** or a set of valid values.
  - For example, the domain of Cust-id is 6 digit numbers.

Formal Definitions - Tuple

- A **tuple** is an ordered set of values (enclosed in angled brackets '<…>')
- Each value is derived from an appropriate **domain**.
- A row in the CUSTOMER relation is a 4-tuple and would consist of four values, for example:
  - <632895, "John Smith", "101 Main St. Atlanta, GA 30332", "(404) 894-2000">
  - This is called a 4-tuple as it has 4 values
  - A tuple (row) in the CUSTOMER relation.
- A relation is a **set** of such tuples (rows)

Formal Definitions - Domain

- A **domain** has a logical definition:
  - Example: “Alberta_phone_numbers” are the set of 10 digit phone numbers valid in Alberta.
- A domain also has a data-type or a format defined for it.
  - The Alberta_phone_numbers may have a format: (ddd)ddd-dddd where each d is a decimal digit.
  - Dates have various formats such as year, month, date formatted as yyyy-mm-dd, or as dd mm,yyyy etc.
- The attribute name designates the role played by a domain in a relation:
  - Used to interpret the meaning of the data elements corresponding to that attribute
  - Example: The domain Date may be used to define two attributes named “Invoice-date” and “Payment-date” with different meanings

Formal Definitions - State

- The **relation state** is a subset of the Cartesian product of the domains of its attributes
  - each domain contains the set of all possible values the attribute can take.
- Example: attribute Cust-name is defined over the domain of character strings of maximum length 25
  - dom(Cust-name) is varchar(25)
- The role these strings play in the CUSTOMER relation is that of the **name of a customer**.

Formal Definitions - Summary

- Formally,
  - Given $R(A_1, A_2, \ldots\ldots\ldots, A_n)$
  - $r(R) \subset \text{dom}(A_1) \times \text{dom}(A_2) \times \ldots\ldots \times \text{dom}(A_n)$
- $R(A_1, A_2, \ldots, A_n)$ is the **schema** of the relation
- R is the **name** of the relation
- $A_1, A_2, \ldots, A_n$ are the **attributes** of the relation
- $r(R)$: a specific **state** (or "value" or “population”) of relation R – this is a **set of tuples** (rows)
  - $r(R) = \{t_1, t_2, \ldots, t_n\}$ where each ti is an *n*-tuple
  - $t_i = < v_1, v_2, \ldots, v_n >$ where each v_j **element-of** dom(A_j)

Formal Definitions - Example

- Let $R(A_1, A_2)$ be a relation schema:
  - Let $\text{dom}(A_1) = \{0,1\}$
  - Let $\text{dom}(A_2) = \{a,b,c\}$
- Then: $\text{dom}(A_1) \times \text{dom}(A_2)$ is all possible combinations:

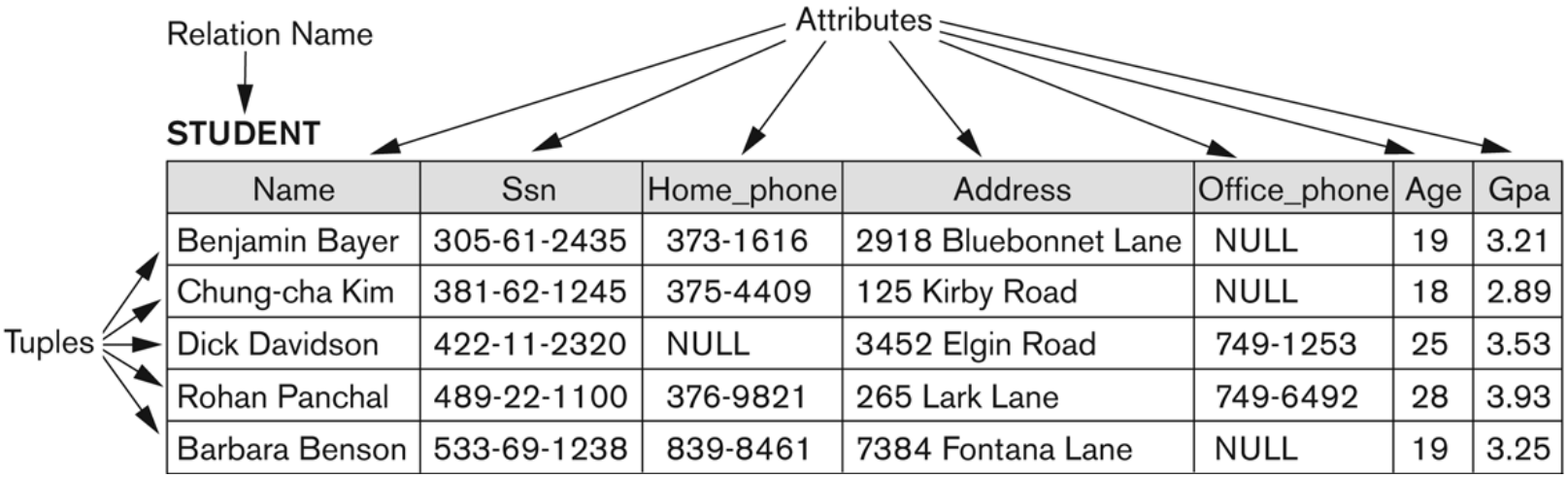
$$\{<0,a>,<0,b>,<0,c>,<1,a>,<1,b>,<1,c>\}$$

- The relation state $r(R) \subset \text{dom}(A_1) \times \text{dom}(A_2)$
- For example: $r(R)$ could be $\{<0,a>,<0,b>,<1,c>\}$
  - this is one possible state (or "population" or "extension") r of the relation R, defined over A_1 and A_2.
  - It has three 2-tuples: $<0,a>,<0,b>,<1,c>$

Definition Summary

Informal Terms	Formal Terms
Table	Relation
Column Header	Attribute
All possible Column Values	Domain
Row	Tuple
Table Definition	Schema of a Relation
Populated Table	State of the Relation

Example – A Relation STUDENT



Name	Ssn	Home_phone	Address	Office_phone	Age	Gpa
Benjamin Bayer	305-61-2435	373-1616	2918 Bluebonnet Lane	NULL	19	3.21
Chung-cha Kim	381-62-1245	375-4409	125 Kirby Road	NULL	18	2.89
Dick Davidson	422-11-2320	NULL	3452 Elgin Road	749-1253	25	3.53
Rohan Panchal	489-22-1100	376-9821	265 Lark Lane	749-6492	28	3.93
Barbara Benson	533-69-1238	839-8461	7384 Fontana Lane	NULL	19	3.25

$R(A_1, A_2, A_3, A_4, A_5, A_6, A_7)$ = STUDENT(Name, Ssn, Home_phone, Address, Office_phone, Age, Gpa)

STUDENT(Name: string, Ssn: string, Home_phone: string, Address: string, Office_phone: string, Age: integer, Gpa: real)

Example domain: dom(Gpa) = Grade_point_averages
Grade_point_averages are the possible values of computed grade point averages; each must be a real (floating-point) number between 0 and 4.

Example tuple: t = <'Barbara Benson', '533-68-1238', '839-8461', '7384 Fontana Lane', NULL, 19, 3.25>

Characteristics of Relations (1 of 3)

- Ordering of tuples in a relation $r(R)$:
  - The tuples are **not considered to be ordered**, even though they appear to be in the tabular form.
- Ordering of attributes in a relation schema R (and of values within each tuple):
  - We will consider the attributes in $R(A_1, A_2, ..., A_n)$ and the values in $t = <v_1, v_2, ..., v_n>$ to be ordered.
    - (However, a more general alternative definition of relation does not require this ordering. It includes both the name and the value for each of the attributes).
    - Example: t = { <name, “John” >, <SSN, 123456789> }
    - This representation may be called as “self-describing”.

Same State as Previous Figure (but with Different Order of Tuples)

Figure 5.2 The relation STUDENT from Figure 5.1 with a different order of tuples.

STUDENT

Name	Ssn	Home_phone	Address	Office_phone	Age	Gpa
Dick Davidson	422-11-2320	NULL	3452 Elgin Road	749-1253	25	3.53
Barbara Benson	533-69-1238	839-8461	7384 Fontana Lane	NULL	19	3.25
Rohan Panchal	489-22-1100	376-9821	265 Lark Lane	749-6492	28	3.93
Chung-cha Kim	381-62-1245	375-4409	125 Kirby Road	NULL	18	2.89
Benjamin Bayer	305-61-2435	373-1616	2918 Bluebonnet Lane	NULL	19	3.21

Characteristics of Relations (2 of 3)

- Values in a tuple:
  - All values are considered atomic (indivisible).
  - Each value in a tuple must be from the domain of the attribute for that column
    - If tuple $t = <v_1, v_2, \ldots, v_n>$ is a tuple (row) in the relation state r of $R(A_1, A_2, \ldots, A_n)$
    - Then each v_i must be a value from **dom** (A_i)
  - A special **null** value is used to represent values that are unknown or not available or inapplicable in certain tuples.

Characteristics of Relations (3 of 3)

- Notation:
  - We refer to **component values** of a tuple *t* by:
    - $t[A_i]$ or $t.A_i$
    - This is the value v_i of attribute A_i for tuple t
  - Similarly, $t[A_u, A_v, ..., A_w]$ refers to the subtuple of *t* containing the values of attributes $A_u, A_v, ..., A_w$, respectively in *t*