

## 1. Relational Data Model - Review

- 1.1 Relational Data Model and properties
  - History, Model, Keys, Indices, Constraints
- 1.2 Relational algebra
  - Basic relational operators
  - Joins
  - Division

## 1. Relational Data Model - Review

- **1.1 Relational Data (RD) Model Properties and Keys**
  - History, Model, Keys, Indices, Constraints
- 1.2 Relational algebra
  - Basic relational operators
  - Joins
  - Division

# 1.1 RD Model: Properties and Keys

## Learning Outcomes

- Explain and use basic relational algebra operators
- Explain and use equivalence of expressions
- Explain and use various forms of joins and other relational operators

## • Textbook Readings

- Chap 1 and Chap 2 ... reading
- Chap 3 ... readings (Keys, Integrity constraints)

## • Testing\*

\*Main (but not the only ones) sections of the textbook used for testing are identified in parentheses

- Relational model properties
- Keys and integrity constraints (Relational model properties ( Keys and integrity constraints ( 3.1 – 3.3 ) )

# 1.1 Relational Data Model – Historical Perspective

From <https://www.quickbase.com/articles/timeline-of-database-history>

## Evolution of Relational DB systems

- Prior to 1960s: File systems accessed by (independent) programs
- 1960s: Computerized database started in the 1960s ... two popular data models
  - network model called CODASYL
  - hierarchical model called IMS
- 1970 to 1972: E.F. Codd published a relational database model in which the database's schema (logical organization) was disconnected from physical information storage ... simple yet defined on math (relational algebra) ... business and academia liked it
- Mid - 1970s: Two major relational database system prototypes were created between the years 1974 and 1977:
  - Ingres, which was developed at UBC ... used QUEL language ... MS SQL Server, Sybase, ...
  - System R, created at IBM San Jose ... used SEQUEL language ... E.g., DB2, Oracle, ...
- 1976: P. Chen published Entity-Relationship, or ER ... for Data Modeling
- 1980s: Structured Query Language, or SQL, became the standard query language.

- Relational Data Model – by Codd in 1970
- Was embraced by Industry and Academia (research and development)
  - Simple – widely understood
  - Based on Set Theory – rigorous and hence amenable to research and development in a logical and rigorous manner

EmployeeNumber	FirstName	LastName	Department	Email	Phone
100	Jerry	Johnson	Accounting	JJ@somewhere.com	236-9987
200	Mary	Abernathy	Finance	MA@somewhere.com	444-8898
300	Liz	Smathers	Finance	LS@somewhere.com	777-0098
400	Tom	Caruthers	Accounting	TC@somewhere.com	236-9987
500	Tom	Jackson	Production	TJ@somewhere.com	444-9980
600	Eleanore	Caldera	Legal	EC@somewhere.com	767-0900
700	Richard	Bandalone	Legal	RB@somewhere.com	767-0900

- Relation - definition by Codd:

Relation is a set of tuples  $(d_1, d_2, \dots, d_n)$ , where each element  $d_j$  is a member of  $D_j$ , a data domain ...

- from wiki: [https://en.wikipedia.org/wiki/Relation\\_\(database\)](https://en.wikipedia.org/wiki/Relation_(database))

EmployeeNumber	FirstName	LastName	Department	Email
100	Jerry	Johnson	Accounting	JJ@somewh
200	Mary	Abernathy	Finance	MA@somewh
300	Liz	Smathers	Finance	LS@somewh
400	Tom	Caruthers	Accounting	TC@somewh
500	Tom	Jackson	Production	TJ@somewh
600	Eleanore	Caldera	Legal	EC@somewh
700	Richard	Bandalone	Legal	RB@somewh

- Relation ... tuples ... attributes ... values
- Relational algebra operators
  - Basic (for retrieval):
    - Selection, Projection, Cartesian product, Union, and Set Difference

[https://en.wikipedia.org/wiki/Relational\\_algebra#Introduction](https://en.wikipedia.org/wiki/Relational_algebra#Introduction)

Table	Column	Row
Relation	Attribute	Tuple
File	Field	Record

Table 3.1	Characteristics of a Relational Table
1	A table is perceived as a two-dimensional structure composed of rows and columns.
2	Each table row (tuple) represents a single entity occurrence within the entity set.
3	Each table column represents an attribute, and each column has a distinct name.
4	Each intersection of a row and column represents a single data value.
5	All values in a column must conform to the same data format (all values must be from a domain set)
6	Each column has a specific range of values known as the attribute domain.
7	The order of the rows and columns is immaterial to the DBMS.
8	Each table must have an attribute or combination of attributes that uniquely identifies each row (each table must have a primary key).

EmployeeNumber	FirstName	LastName	Department	
100	Jerry	Johnson	Accounting	JJ@s
200	Mary	Abernathy	Finance	MA@
300	Liz	Smathers	Finance	LS@
400	Tom	Caruthers	Accounting	TC@
500	Tom	Jackson	Production	TJ@
600	Eleanore	Caldera	Legal	EC@
700	Richard	Bandalone	Legal	RB@



**FIGURE 3.2 AN EXAMPLE OF A SIMPLE RELATIONAL DATABASE**

**Table name: PRODUCT**

**Primary key: PROD\_CODE**

**Foreign key: VEND\_CODE**

**Database name: Ch03\_SaleCo**

PROD_CODE	PROD_DESCRIPT	PROD_PRICE	PROD_ON_HAND	VEND_CODE
001278-AB	Claw hammer	12.95	23	232
123-21UUY	Houselite chain saw, 16-in. bar	189.99	4	235
QER-34256	Sledge hammer, 16-lb. head	18.63	6	231
SRE-657UG	Rat-tail file	2.99	15	232
ZZX/3245Q	Steel tape, 12-ft. length	6.79	8	235

link

**Table name: VENDOR**

**Primary key: VEND\_CODE**

**Foreign key: none**

VEND_CODE	VEND_CONTACT	VEND_AREACODE	VEND_PHONE
230	Shelly K. Smithson	608	555-1234
231	James Johnson	615	123-4536
232	Annelise Crystall	608	224-2134
233	Candice Wallace	904	342-6567
234	Arthur Jones	615	123-3324
235	Henry Ortozo	615	899-3425



- Keys – Keys – and more Keys
  - Designer determines based on application semantics
    - Semantics are derived by talking to users
  - Composite key vs single-column key
  - Key attribute
  - Superkey (or simply a key)
    - Candidate key
      - *irreducible attributes* to ensure uniqueness (cannot remove any attr)
    - Primary key ... DB chosen candidate key
  - Null: absence of any data value
  - Entity Integrity constraint (unique, no null)
  - Foreign key constraint
  - Secondary key ... DB concept only
    - not in Relational Data Model as secondary keys may not be unique (e.g., Name used for secondary key for efficient search)

EMP

ENO	ENAME	TITLE
E1	J. Doe	Elect. Eng
E2	M. Smith	Syst. Anal.
E3	A. Lee	Mech. Eng.
E4	J. Miller	Programmer
E5	B. Casey	Syst. Anal.
E6	L. Chu	Elect. Eng.
E7	R. Davis	Mech. Eng.
E8	J. Jones	Syst. Anal.

ASG

ENO	PNO	RESP	DUR
E1	P1	Manager	12
E2	P1	Analyst	24
E2	P2	Analyst	6
E3	P3	Consultant	10
E3	P4	Engineer	48
E4	P2	Programmer	18
E5	P2	Manager	24
E6	P4	Manager	48
E7	P3	Engineer	36
E8	P3	Manager	40

PROJ

PNO	PNAME	BUDGET	LOC
P1	Instrumentation	150000	Montreal
P2	Database Develop.	135000	New York
P3	CAD/CAM	250000	New York
P4	Maintenance	310000	Paris

PAY

TITLE	SAL
Elect. Eng.	40000
Syst. Anal.	34000
Mech. Eng.	27000
Programmer	24000

- Several different types of keys are used in the relational data model
  - **Composite key**: key that is composed of more than one attribute
  - **Key attribute**: attribute that is a part of a key
  - **Superkey, or simply a key**: key that can uniquely identify any row in the table
  - **Null**: absence of any data value
    - Unknown attribute value, known but missing attribute value, or inapplicable condition
  - **Foreign key**: primary key of one table that has been placed into another table to create a common attribute/link
  - **Key and Secondary key**: keys (e.g., for indices) that are used for data retrieval purposes by the DBMS itself – DB Administrator use only

- **Super Key** - consist of one or more attributes that determine other attributes that ensure that each row in a table is uniquely identifiable.
- **Candidate Key** – is a **minimal (irreducible)** superkey – **minimal**: If we remove any attribute from it (if it is composite) then it will no longer be a superkey – it is defined/determined by the DB administrator
  - A table can have a number of candidate keys
    - Examples
      - Employee table .... Employee ID, Social Insurance Number (SIN), Credit card number
      - Vehicle ... License Plate number, Serial number
- **Primary key**: A candidate key chosen as a primary key when a table is created
- **Foreign key**:
  - A **foreign key** is a column (or collection of columns) in one table that refers to the **primary key** in another table. It is used to link two tables together.

## • Constraints

- **Null**: absence of any data value
  - Unknown attribute value, known but missing attribute value, or inapplicable condition
  - **NOT NULL constraint**: placed on a column to ensure that every row in the table has a value for that column
- **UNIQUE** constraint: restriction placed on a column to ensure that no duplicate values exist for that column => allows NULL value
- **Foreign key**: primary key of one table that has been placed into another table to create a common attribute
- **Secondary key**:
  - Used strictly for data retrieval purposes by the DBMS itself – DB Administrator use only
  - Secondary key concept does not exist in a Relational Data Model
  - Secondary key might not be unique (used for efficient retrieval only)
- **Entity integrity**: condition in which each row in the table has its own unique identity
  - All of the values in the primary key must be unique
  - No key attribute in the primary key can contain a null
- **Referential integrity**: every reference to an entity instance by another entity instance is valid

Table 3.3	Relational Database Keys
Key Type	Definition
Superkey	An attribute or combination of attributes that uniquely identifies each row in a table.
Candidate key	A minimal (irreducible) superkey; a superkey that does not contain a subset of attributes that is itself a superkey.
Primary key	A candidate key selected to uniquely identify all other attribute values in any given row; cannot contain null entries.
Foreign key	An attribute or combination of attributes in one table whose values must either match the primary key in another table or be null.
Secondary key	An attribute or combination of attributes used strictly for data retrieval purposes.

# Keys and Integrity Constraints ... continued

**FIGURE 3.3 AN ILLUSTRATION OF INTEGRITY RULES**

**Table name: CUSTOMER**

**Database name: Ch03\_InsureCo**

**Primary key: CUS\_CODE**

**Foreign key: AGENT\_CODE**

CUS_CODE	CUS_LNAME	CUS_FNAME	CUS_INITIAL	CUS_RENEW_DATE	AGENT_CODE
10010	Ramas	Alfred	A	05-Apr-2018	502
10011	Dunne	Leona	K	16-Jun-2018	501
10012	Smith	Kathy	W	29-Jan-2019	502
10013	Olowski	Paul	F	14-Oct-2018	
10014	Orlando	Myron		28-Dec-2018	501
10015	O'Brian	Amy	B	22-Sep-2018	503
10016	Brown	James	G	25-Mar-2019	502
10017	Williams	George		17-Jul-2018	503
10018	Farriss	Anne	G	03-Dec-2018	501
10019	Smith	Olette	K	14-Mar-2019	503

**Table name: AGENT (only five selected fields are shown)**

**Primary key: AGENT\_CODE**

**Foreign key: none**

AGENT_CODE	AGENT_AREACODE	AGENT_PHONE	AGENT_LNAME	AGENT_YTD_SLS
501	713	228-1249	Alby	132735.75
502	615	882-1244	Hahn	138967.35
503	615	123-5589	Okon	127093.45

# Questions / Answers

