

Fundamentals of
**DATABASE
SYSTEMS**

FOURTH EDITION

ELMASRI  NAVATHE

Chapter 5

The Relational Data Model and Relational Database Constraints



Chapter Outline

- Relational Model Concepts
- Relational Model Constraints and Relational Database Schemas
- Update Operations and Dealing with Constraint Violations

Relational Model Concepts

- The relational Model of Data is based on the concept of a Relation.
- A Relation is a mathematical concept based on the ideas of sets.
- 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 relational approach in this chapter.

Relational Model Concepts

- The model was first proposed by Dr. E.F. Codd of IBM in 1970 in the following paper:
"A Relational Model for Large Shared Data Banks," Communications of the ACM, June 1970.

The above paper caused a major revolution in the field of Database management and earned Ted Codd the coveted ACM Turing Award.

INFORMAL DEFINITIONS

- **RELATION:** A table of values
 - A relation may be thought of as a **set of rows**.
 - A relation may alternately be thought of as a **set of columns**.
 - Each row represents a fact that corresponds to a real-world **entity** or **relationship**.
 - Each row has a value of an item or set of items that uniquely identifies that row in the table.
 - Sometimes row-ids or sequential numbers are assigned to identify the rows in the table.
 - Each column typically is called by its column name or column header or attribute name.

FORMAL DEFINITIONS

- A **Relation** may be defined in multiple ways.
- The **Schema** of a Relation: $R (A_1, A_2, \dots, A_n)$
Relation schema R is defined over **attributes** A_1, A_2, \dots, A_n
For Example -
CUSTOMER (Cust-id, Cust-name, Address, Phone#)

Here, CUSTOMER is a relation defined over the four attributes Cust-id, Cust-name, Address, Phone#, each of which has a **domain** or a set of valid values. For example, the domain of Cust-id is 6 digit numbers.

FORMAL DEFINITIONS

- A **tuple** is an ordered set of values
- Each value is derived from an appropriate domain.
- Each row in the CUSTOMER table may be referred to as a tuple in the table and would consist of four values.
- $\langle 632895, \text{"John Smith"}, \text{"101 Main St. Atlanta, GA 30332"}, \text{"(404) 894-2000"} \rangle$ is a tuple belonging to the CUSTOMER relation.
- A relation may be regarded as a **set of tuples** (rows).
- Columns in a table are also called attributes of the relation.

FORMAL DEFINITIONS

- A **domain** has a logical definition: e.g., “USA_phone_numbers” are the set of 10 digit phone numbers valid in the U.S.
- A domain may have a data-type or a format defined for it. The USA_phone_numbers may have a format: (ddd)-ddd-dddd where each d is a decimal digit. E.g., Dates have various formats such as monthname, date, year or yyyy-mm-dd, or dd mm,yyyy etc.
- An attribute designates the **role** played by the domain. E.g., the domain Date may be used to define attributes “Invoice-date” and “Payment-date”.

FORMAL DEFINITIONS

- The relation is formed over the cartesian product of the sets; each set has values from a domain; that domain is used in a specific role which is conveyed by the attribute name.
- For example, attribute Cust-name is defined over the domain of strings of 25 characters. The role these strings play in the CUSTOMER relation is that of the name of customers.
- Formally,
Given $R(A_1, A_2, \dots, A_n)$
 $r(R) \subset \text{dom}(A_1) \times \text{dom}(A_2) \times \dots \times \text{dom}(A_n)$
- R: schema of the relation
- r of R: a specific "value" or population of R.
- R is also called the **intension** of a relation
- r is also called the **extension** of a relation

FORMAL DEFINITIONS

- Let $S1 = \{0,1\}$
- Let $S2 = \{a,b,c\}$
- Let $R \subset S1 \times S2$
- Then for example: $r(R) = \{ \langle 0,a \rangle, \langle 0,b \rangle, \langle 1,c \rangle \}$
is one possible “state” or “population” or
“extension” r of the relation R , defined over domains
 $S1$ and $S2$. It has three tuples.

DEFINITION SUMMARY

Informal Terms

Table
Column
Row
Values in a column
Table Definition
Populated Table

Formal Terms

Relation
Attribute/Domain
Tuple
Domain
Schema of a Relation
Extension

Example

Figure 7.1 The attributes and tuples of a relation STUDENT.

STUDENT	Name	SSN	HomePhone	Address	OfficePhone	Age	GPA
	Benjamin Bayer	305-61-2435	373-1616	2918 Blaubornnet Lane	null	19	3.21
	Katherine Ashly	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
	Charles Cooper	489-22-1100	376-9821	265 Lark Lane	749-6492	28	3.93
	Barbara Berson	533-69-1238	639-8461	7384 Fontana Lane	null	19	3.25

© Addison Wesley Longman, Inc. 2000, Elmasri/Navathe, Fundamentals of Database Systems, Third Edition

CHARACTERISTICS OF RELATIONS

- **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 $R(A_1, A_2, \dots, A_n)$ and the values in $t = \langle v_1, v_2, \dots, v_n \rangle$ to be *ordered*.

(However, a more general *alternative definition* of relation does not require this ordering).

- **Values in a tuple:** All values are considered *atomic* (indivisible). A special **null** value is used to represent values that are unknown or inapplicable to certain tuples.

CHARACTERISTICS OF RELATIONS

● Notation:

- We refer to **component values** of a tuple t by $t[A_i] = v_i$ (the value of attribute A_i for tuple t).

Similarly, $t[A_u, A_v, \dots, A_w]$ refers to the subtuple of t containing the values of attributes A_u, A_v, \dots, A_w , respectively.

CHARACTERISTICS OF RELATIONS

Figure 7.2 The relation STUDENT from Figure 7.1, with a different order of tuples.

STUDENT	Name	SSN	HomePhone	Address	OfficePhone	Age	GPA
	Dick Davidson	422-11-2320	null	3452 Elgin Road	749-1253	25	3.53
	Barbara Benson	533-09-1238	839-8461	7394 Fontana Lane	null	19	3.25
	Charles Cooper	489-22-1100	375-9821	265 Lark Lane	749-6492	28	3.93
	Katherine Ashly	381-62-1245	375-4409	125 Kirby Road	null	18	2.89

Relational Integrity Constraints

- Constraints are *conditions* that must hold on *all* valid relation instances. There are three main types of constraints:
 1. **Key constraints**
 2. **Entity integrity constraints**
 3. **Referential integrity constraints**

Key Constraints

- **Superkey of R:** A set of attributes SK of R such that no two tuples *in any valid relation instance* $r(R)$ will have the same value for SK. That is, for any distinct tuples t_1 and t_2 in $r(R)$, $t_1[SK] \neq t_2[SK]$.
- **Key of R:** A "minimal" superkey; that is, a superkey K such that removal of any attribute from K results in a set of attributes that is not a superkey.

Example: The CAR relation schema:

CAR(State, Reg#, SerialNo, Make, Model, Year)

has two keys $Key_1 = \{State, Reg\# \}$, $Key_2 = \{SerialNo \}$, which are also superkeys. $\{SerialNo, Make \}$ is a superkey but *not* a key.

- If a relation has *several candidate keys*, one is chosen arbitrarily to be the **primary key**. The primary key attributes are *underlined*.

Key Constraints

Figure 5.4 The CAR relation with two candidate keys: LicenseNumber and EngineSerialNumber.

CAR	LicenseNumber	EngineSerialNumber	Make	Model	Year
Texas ABC-739	A69352		Ford	Mustang	96
Florida TVP-347	B43696		Oldsmobile	Cutlass	99
New York MPO-22	X83554		Oldsmobile	Delta	95
California 432-TFY	C43742		Mercedes	190-D	93
California RSK-629	Y82935		Toyota	Camry	98
Texas RSK-629	U028365		Jaguar	XJS	98

© Addison Wesley Longman, Inc. 2000, Elmasri/Navathe, Fundamentals of Database Systems, Third Edition

Entity Integrity

- **Relational Database Schema:** A set S of relation schemas that belong to the same database. S is the *name* of the **database**.

$$S = \{R_1, R_2, \dots, R_n\}$$

- **Entity Integrity:** The *primary key attributes* PK of each relation schema R in S cannot have null values in any tuple of $r(R)$. This is because primary key values are used to *identify* the individual tuples.

$$t[\text{PK}] \neq \text{null for any tuple } t \text{ in } r(R)$$

- **Note:** Other attributes of R may be similarly constrained to disallow null values, even though they are not members of the primary key.

Referential Integrity

- A constraint involving *two* relations (the previous constraints involve a *single* relation).
- Used to specify a *relationship* among tuples in two relations: the **referencing relation** and the **referenced relation**.
- Tuples in the *referencing relation* R_1 have attributes FK (called **foreign key** attributes) that reference the primary key attributes PK of the *referenced relation* R_2 . A tuple t_1 in R_1 is said to **reference** a tuple t_2 in R_2 if $t_1[\text{FK}] = t_2[\text{PK}]$.
- A referential integrity constraint can be displayed in a relational database schema as a directed arc from R_1 .FK to R_2 .

Referential Integrity Constraint

Statement of the constraint

The value in the foreign key column (or columns) FK of the the **referencing relation** R_1 can be either:

- (1) a value of an existing primary key value of the corresponding primary key PK in the **referenced relation** R_2 , or..
- (2) a null.

In case (2), the FK in R_1 should not be a part of its own primary key.

Other Types of Constraints

Semantic Integrity Constraints:

- based on application semantics and cannot be expressed by the model per se
- E.g., “the max. no. of hours per employee for all projects he or she works on is 56 hrs per week”
- A *constraint specification language* may have to be used to express these
- SQL-99 allows triggers and ASSERTIONS to allow for some of these

Figure 5.5 Schema diagram for the COMPANY relational database schema; the primary keys are underlined.

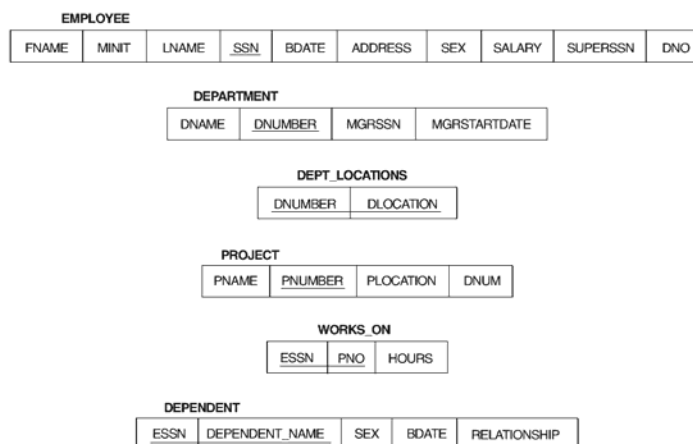


Figure 5.6 One possible relational database state corresponding to the COMPANY schema.

EMPLOYEE	FNAME	MINIT	LNAME	SSN	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DNO
Apple	John		Elmasri	02430039	1965-01-09	2017 Adams, Houston, TX	M	30000	03344000	1
Franklin	Mary		Elmasri	1800-02-08	839 West, Houston, TX	M	40000	00000000	0	3
John	Zelma		Elmasri	1988-01-10	1201 Cook, Spring, TX	F	20000	00-00-00	0	4
James	Madame		Elmasri	1941-05-05	211 Hwy, Dallas, TX	F	40000	00000000	0	4
Thomas	Reagan		Elmasri	1965-08-15	3107 The Oak, Houston, TX	M	30000	03344000	0	3
John	Doyle		Elmasri	1910-01-13	1407 New, Houston, TX	F	20000	03344000	0	3
Arnold	Julius		Elmasri	1960-03-29	1800 Dallas, Houston, TX	M	20000	00-00-00	0	2
James	King		Elmasri	1974-01-10	407 Stone, Houston, TX	M	10000	NULL	NULL	1

DEPARTMENT	DNAME	DNUMBER	MGRSSN	MGRSTARTDATE
Research	Research	1	02430039	1988-01-01
Administration	Administration	4	00-00-00	1989-01-01
Human Resources	Human Resources	1	00000000	1987-06-10

WORKS_ON	ESSN	PNO	HOURS
02430039	1	30.0	
02430039	2	7.5	
00000000	3	40.0	
03344000	1	30.0	
03344000	2	30.0	
03344000	3	10.0	
03344000	4	10.0	
03344000	5	10.0	
03344000	6	10.0	
00000000	10	10.0	
00000000	11	10.0	
00000000	12	10.0	
00000000	13	10.0	
00000000	14	10.0	
00000000	15	10.0	
00000000	16	10.0	
00000000	17	10.0	
00000000	18	10.0	
00000000	19	10.0	
00000000	20	10.0	
00000000	21	10.0	
00000000	22	10.0	
00000000	23	10.0	
00000000	24	10.0	
00000000	25	10.0	
00000000	26	10.0	
00000000	27	10.0	
00000000	28	10.0	
00000000	29	10.0	
00000000	30	10.0	
00000000	31	10.0	
00000000	32	10.0	
00000000	33	10.0	
00000000	34	10.0	
00000000	35	10.0	
00000000	36	10.0	
00000000	37	10.0	
00000000	38	10.0	
00000000	39	10.0	
00000000	40	10.0	
00000000	41	10.0	
00000000	42	10.0	
00000000	43	10.0	
00000000	44	10.0	
00000000	45	10.0	
00000000	46	10.0	
00000000	47	10.0	
00000000	48	10.0	
00000000	49	10.0	
00000000	50	10.0	
00000000	51	10.0	
00000000	52	10.0	
00000000	53	10.0	
00000000	54	10.0	
00000000	55	10.0	
00000000	56	10.0	
00000000	57	10.0	
00000000	58	10.0	
00000000	59	10.0	
00000000	60	10.0	
00000000	61	10.0	
00000000	62	10.0	
00000000	63	10.0	
00000000	64	10.0	
00000000	65	10.0	
00000000	66	10.0	
00000000	67	10.0	
00000000	68	10.0	
00000000	69	10.0	
00000000	70	10.0	
00000000	71	10.0	
00000000	72	10.0	
00000000	73	10.0	
00000000	74	10.0	
00000000	75	10.0	
00000000	76	10.0	
00000000	77	10.0	
00000000	78	10.0	
00000000	79	10.0	
00000000	80	10.0	
00000000	81	10.0	
00000000	82	10.0	
00000000	83	10.0	
00000000	84	10.0	
00000000	85	10.0	
00000000	86	10.0	
00000000	87	10.0	
00000000	88	10.0	
00000000	89	10.0	
00000000	90	10.0	
00000000	91	10.0	
00000000	92	10.0	
00000000	93	10.0	
00000000	94	10.0	
00000000	95	10.0	
00000000	96	10.0	
00000000	97	10.0	
00000000	98	10.0	
00000000	99	10.0	
00000000	100	10.0	

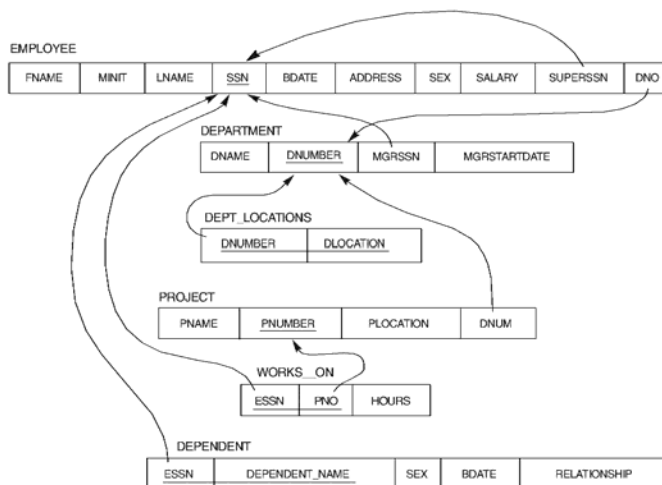
PROJECT	PNAME	PNUMBER	PLOCATION	DNUM
Project1	Project1	1	Houston	3
Project2	Project2	2	Houston	3
Project3	Project3	3	Houston	3
Project4	Project4	4	Houston	3
Project5	Project5	5	Houston	3
Project6	Project6	6	Houston	3
Project7	Project7	7	Houston	3
Project8	Project8	8	Houston	3
Project9	Project9	9	Houston	3
Project10	Project10	10	Houston	3
Project11	Project11	11	Houston	3
Project12	Project12	12	Houston	3
Project13	Project13	13	Houston	3
Project14	Project14	14	Houston	3
Project15	Project15	15	Houston	3
Project16	Project16	16	Houston	3
Project17	Project17	17	Houston	3
Project18	Project18	18	Houston	3
Project19	Project19	19	Houston	3
Project20	Project20	20	Houston	3
Project21	Project21	21	Houston	3
Project22	Project22	22	Houston	3
Project23	Project23	23	Houston	3
Project24	Project24	24	Houston	3
Project25	Project25	25	Houston	3
Project26	Project26	26	Houston	3
Project27	Project27	27	Houston	3
Project28	Project28	28	Houston	3
Project29	Project29	29	Houston	3
Project30	Project30	30	Houston	3
Project31	Project31	31	Houston	3
Project32	Project32	32	Houston	3
Project33	Project33	33	Houston	3
Project34	Project34	34	Houston	3
Project35	Project35	35	Houston	3
Project36	Project36	36	Houston	3
Project37	Project37	37	Houston	3
Project38	Project38	38	Houston	3
Project39	Project39	39	Houston	3
Project40	Project40	40	Houston	3
Project41	Project41	41	Houston	3
Project42	Project42	42	Houston	3
Project43	Project43	43	Houston	3
Project44	Project44	44	Houston	3
Project45	Project45	45	Houston	3
Project46	Project46	46	Houston	3
Project47	Project47	47	Houston	3
Project48	Project48	48	Houston	3
Project49	Project49	49	Houston	3
Project50	Project50	50	Houston	3
Project51	Project51	51	Houston	3
Project52	Project52	52	Houston	3
Project53	Project53	53	Houston	3
Project54	Project54	54	Houston	3
Project55	Project55	55	Houston	3
Project56	Project56	56	Houston	3
Project57	Project57	57	Houston	3
Project58	Project58	58	Houston	3
Project59	Project59	59	Houston	3
Project60	Project60	60	Houston	3
Project61	Project61	61	Houston	3
Project62	Project62	62	Houston	3
Project63	Project63	63	Houston	3
Project64	Project64	64	Houston	3
Project65	Project65	65	Houston	3
Project66	Project66	66	Houston	3
Project67	Project67	67	Houston	3
Project68	Project68	68	Houston	3
Project69	Project69	69	Houston	3
Project70	Project70	70	Houston	3
Project71	Project71	71	Houston	3
Project72	Project72	72	Houston	3
Project73	Project73	73	Houston	3
Project74	Project74	74	Houston	3
Project75	Project75	75	Houston	3
Project76	Project76	76	Houston	3
Project77	Project77	77	Houston	3
Project78	Project78	78	Houston	3
Project79	Project79	79	Houston	3
Project80	Project80	80	Houston	3
Project81	Project81	81	Houston	3
Project82	Project82	82	Houston	3
Project83	Project83	83	Houston	3
Project84	Project84	84	Houston	3
Project85	Project85	85	Houston	3
Project86	Project86	86	Houston	3
Project87	Project87	87	Houston	3
Project88	Project88	88	Houston	3
Project89	Project89	89	Houston	3
Project90	Project90	90	Houston	3
Project91	Project91	91	Houston	3
Project92	Project92	92	Houston	3
Project93	Project93	93	Houston	3
Project94	Project94	94	Houston	3
Project95	Project95	95	Houston	3
Project96	Project96	96	Houston	3
Project97	Project97	97	Houston	3
Project98	Project98	98	Houston	3
Project99	Project99	99	Houston	3
Project100	Project100	100	Houston	3

DEPENDENT	ESSN	DEPENDENT_NAME	SEX	BDATE	RELATIONSHIP
02430039	Apple	John	M	1965-01-09	DAUGHTER
03344000	Franklin	Mary	F	1980-02-08	SUN
03344000	John	Zelma	F	1988-01-10	SISTER
00000000	James	Madame	F	1941-05-05	SISTER
03344000	Thomas	Reagan	M	1965-08-15	SUN
03344000	John	Doyle	F	1910-01-13	SUN
03344000	James	King	M	1974-01-10	SUN

© Addison Wesley Longman, Inc. 2000, Elmasri/Navathe, Fundamentals of Database Systems, Third Edition

Copyright © 2004 Ramez Elmasri and Shamkant Navathe

Figure 5.7 Referential integrity constraints displayed on the COMPANY relational database schema diagram.



© Addison Wesley Longman, Inc. 2000, Elmasri/Navathe, Fundamentals of Database Systems, Third Edition

Copyright © 2004 Ramez Elmasri and Shamkant Navathe

Update Operations on Relations

- INSERT a tuple.
- DELETE a tuple.
- MODIFY a tuple.

- Integrity constraints should not be violated by the update operations.
- Several update operations may have to be grouped together.
- Updates may *propagate* to cause other updates automatically. This may be necessary to maintain integrity constraints.

Update Operations on Relations

- In case of integrity violation, several actions can be taken:
 - Cancel the operation that causes the violation (REJECT option)
 - Perform the operation but inform the user of the violation
 - Trigger additional updates so the violation is corrected (CASCADE option, SET NULL option)
 - Execute a user-specified error-correction routine

In-Class Exercise

(Taken from Exercise 5.15)

Consider the following relations for a database that keeps track of student enrollment in courses and the books adopted for each course:

STUDENT(SSN, Name, Major, Bdate)

COURSE(Course#, Cname, Dept)

ENROLL(SSN, Course#, Quarter, Grade)

BOOK_ADOPTION(Course#, Quarter, Book_ISBN)

TEXT(Book_ISBN, Book_Title, Publisher, Author)

Draw a relational schema diagram specifying the foreign keys for this schema.