

CSCI1270

Introduction to Database Systems

Relations

Relational Data Model

Introduced by Ted Codd (late 60's – early 70's)

- Before = “Network Data Model” (Cobol as DDL, DML)
- Very contentious: Database Wars

Relational data model contributes:

1. Separation of logical and physical data models (data independence)
2. Declarative query languages
3. Formal semantics
4. Query optimization (key to commercial success)

First prototypes:

- Ingres (UC Berkeley)
- System R (IBM) → DB2

Relations

Account =

bname	acct_no	balance
Downtown	A-101	500
Brighton	A-201	900
Brighton	A-217	500

Table name

Attribute names

Terms:

- **Tables** → Relations
- **Columns** → Attributes
- **Rows** → Tuples
- **Schema** (e.g.: Acct_Schema = (bname, acct_no, balance))
- **Domain** -> set of all possible values for an attribute.
(e.g., domain(acct_no)= { A101, A201, A217, **A230**})

Why Are They Called Relations?

Mathematical relations

Given sets: $R = \{1, 2, 3\}$, $S = \{3, 4\}$

- $R \times S = \{ (1, 3), (1, 4), (2, 3), (2, 4), (3, 3), (3, 4) \}$
- A **relation** on R, S is any subset (\subseteq) of $R \times S$
 - (e.g: $\{ (1, 4), (3, 4) \}$)

Database relations

Given attribute domains

Branches = $\{ \text{Downtown, Brighton, ...} \}$

Accounts = $\{ \text{A-101, A-201, A-217, ...} \}$

Balances = $R = \text{real numbers}$

Account \subseteq *Branches* \times *Accounts* \times *Balances* (no nesting = 1NF)

$\{ (\text{Downtown}, \text{A-101}, 500),$
 $(\text{Brighton}, \text{A-201}, 900),$
 $(\text{Brighton}, \text{A-217}, 500) \}$

Relations

Account =

bname	acct_no	balance
Downtown	A-101	500
Brighton	A-201	900
Brighton	A-217	500

Considered equivalent to...

{ (Downtown, A-101, 500),
(Brighton, A-201, 900),
(Brighton, A-217, 500) }

Relational database semantics are defined in terms of mathematical relations (i.e., sets)

Keys and Relations

Kinds of keys

1. Superkeys

- set of attributes of table for which every row has distinct set of values

2. Candidate keys

- “minimal” superkeys

3. Primary keys

- DBA-chosen candidate key

Act as Integrity Constraints

i.e., guard against illegal/invalid instance of given schema

e.g., Branch = (bname, bcity, assets) \Rightarrow

bname	bcity	assets
Brighton	Brooklyn	5M
Brighton	Boston	3M



Invalid!!

Integrity Constraints in Create Table

- **not null**
- **primary key** (A_1, \dots, A_n)

Example: Declare *ID* as the primary key for *instructor*

.

```
create table instructor (  
    ID          char(5),  
    name        varchar(20) not null,  
    dept_name   varchar(20),  
    salary     numeric(8,2),  
    primary key (ID)
```

primary key declaration on an attribute automatically ensures
not null

And a Few More Relation Definitions

- **create table** *student* (
 student_ID **varchar**(5),
 name **varchar**(20) not null,
 dept_name **varchar**(20),
 total_credit **numeric**(3,0),
 primary key (*student-ID*));
 - **create table** *takes* (
 student_ID **varchar**(5),
 course_id **varchar**(8),
 sec_id **varchar**(8),
 semester **varchar**(6),
 year **numeric**(4,0),
 grade **varchar**(2),
 primary key (*student_ID*, *course_id*, *sec_id*, *semester*, *year*));
- w.o. *sec_id* in key:
- (*stan*, *cs127*, 1, *fall*, 2017, A)
(*stan*, *cs127*, 2, *fall*, 2017, B)
- is disallowed
- Note: *sec_id* should be dropped from primary key above.
 - Must ensure a student cannot be registered for two sections of the same course in the same semester

Example of Using Keys

<i>student_ID</i>	<i>course_id</i>	<i>sec_id</i>	<i>semester</i>	<i>year</i>	<i>grade</i>
S1	CS127	1	fall	2017	A
S1	CS127	2	fall	2017	B

With *sec_id* in the key, these 2 tuples are unique and thus are allowed

Without *sec_id* in the key, these 2 tuples are identical and thus are disallowed

Bank Database

(Schema)

Account		
bname	<u>acct_no</u>	balance

Branch		
<u>bname</u>	bcity	assets

Depositor	
cname	acct_no

Borrower	
cname	lno

Customer		
<u>cname</u>	cstreet	ccity

Loan		
bname	lno	amt

Bank Database

Account		
bname	<u>acct_no</u>	balance
Downtown	A-101	500
Mianus	A-215	700
Perry	A-102	400
R.H.	A-305	350
Brighton	A-201	900
Redwood	A-222	700
Brighton	A-217	750

Depositor	
cname	acct_no
Johnson	A-101
Smith	A-215
Hayes	A-102
Turner	A-305
Johnson	A-201
Jones	A-217
Lindsay	A-222

Customer		
<u>cname</u>	cstreet	ccity
Jones	Main	Harrison
Smith	North	Rye
Hayes	Main	Harrison
Curry	North	Rye
Lindsay	Park	Pittsfield
Turner	Putnam	Stanford
Williams	Nassau	Princeton
Adams	Spring	Pittsfield
Johnson	Alma	Palo Alto
Glenn	Sand Hill	Woodside
Brooks	Senator	Brooklyn
Green	Walnut	Stanford

Branch		
<u>bname</u>	bcity	assets
Downtown	Brooklyn	9M
Redwood	Palo Alto	2.1M
Perry	Horseneck	1.7M
Mianus	Horseneck	0.4M
R.H.	Horseneck	8M
Pownel	Bennington	0.3M
N. Town	Rye	3.7M
Brighton	Brooklyn	7.1M

Borrower	
cname	lno
Jones	L-17
Smith	L-23
Hayes	L-15
Jackson	L-14
Curry	L-93
Smith	L-11
Williams	L-17
Adams	L-16

Loan		
bname	lno	amt
Downtown	L-17	1000
Redwood	L-23	2000
Perry	L-15	1500
Downtown	L-14	1500
Mianus	L-93	500
R.H.	L-11	900
Perry	L-16	1300