Relational Model 2: Relational Algebra

Yufei Tao

Department of Computer Science and Engineering
Chinese University of Hong Kong

The relational model defines:

1. the format by which data should be stored;
2. the operations for querying the data.

We will focus on the second aspect in this lecture.

Relational Algebra

Relational algebra is a language for issuing queries on the data stored in a relation database.

Its core consists of 6 fundamental operations:

- Rename ρ
- Selection σ
- Projection Π
- Set union $\cup$
- Set difference $-$
- Cartesian product $\times$

Rename

Denoted by $\rho_s(T)$

- where T is a table, and s is a string.
- The output of the operation is a table T' that is exactly the same as T, but is named s.

PROF

pid	name	dept	rank	sal
$p1$	Adam	CS	asst	6000
$p2$	Bob	EE	asso	8000
$p3$	Calvin	CS	full	10000
$p4$	Dorothy	EE	asst	5000
$p5$	Emily	EE	asso	8500
$p6$	Frank	CS	full	9000

$\rho_{\text{LECT}}(\text{PROF})$ returns:

LECT

pid	name	dept	rank	sal
$p1$	Adam	CS	asst	6000
$p2$	Bob	EE	asso	8000
$p3$	Calvin	CS	full	10000
$p4$	Dorothy	EE	asst	5000
$p5$	Emily	EE	asso	8500
$p6$	Frank	CS	full	9000

Selection

Denoted by $\sigma_P(T)$

- where T is a table, and P is a predicate on the tuples of T.
- The output is a table T' such that
  - T' has the same schema as T'.
  - T' includes all and only the tuples in T satisfying P.

Each predicate can be

- a comparison using the following operators: $=$, $\neq$, $<$, $\leq$, $>$, $\leq$.
- multiple comparisons connected by $\wedge$ (*and*), $\vee$ (*or*) and $\neg$ (*not*).

PROF

pid	name	dept	rank	sal
$p1$	Adam	CS	asst	6000
$p2$	Bob	EE	asso	8000
$p3$	Calvin	CS	full	10000
$p4$	Dorothy	EE	asst	5000
$p5$	Emily	EE	asso	8500
$p6$	Frank	CS	full	9000

$\sigma_{\text{name}=\text{“Bob”}}(\text{PROF})$ returns:

pid	name	dept	rank	sal
$p2$	Bob	EE	asso	8000

PROF

pid	name	dept	rank	sal
$p1$	Adam	CS	asst	6000
$p2$	Bob	EE	asso	8000
$p3$	Calvin	CS	full	10000
$p4$	Dorothy	EE	asst	5000
$p5$	Emily	EE	asso	8500
$p6$	Frank	CS	full	9000

$\sigma_{\text{dept}=\text{“EE”} \ \wedge \ \text{sal} \ > \ 7000}(\text{PROF})$ returns:

pid	name	dept	rank	sal
$p2$	Bob	EE	asso	8000
$p5$	Emily	EE	asso	8500

Projection

Denoted by $\Pi_A(T)$

- where T is a table, and A is a set of attributes in T.
- The output of the operation is a table T' such that
  - T' has all and only the attributes in A.
  - T' contains all the tuples of T after trimming the attributes not in A.
  - All duplicates (resulting from the trimming) are removed.

PROF

pid	**name**	**dept**	**rank**	**sal**
$p1$	Adam	CS	asst	6000
$p2$	Bob	EE	asso	8000
$p3$	Calvin	CS	full	10000
$p4$	Dorothy	EE	asst	5000
$p5$	Emily	EE	asso	8500
$p6$	Frank	CS	full	9000

$\Pi_{\text{dept}}(\text{PROF})$ returns:

dept
CS
EE

PROF

pid	name	dept	rank	sal
$p1$	Adam	CS	asst	6000
$p2$	Bob	EE	asso	8000
$p3$	Calvin	CS	full	10000
$p4$	Dorothy	EE	asst	5000
$p5$	Emily	EE	asso	8500
$p6$	Frank	CS	full	9000

$\Pi_{\text{dept, rank}}(\text{PROF})$ returns:

dept	rank
CS	asst
EE	asso
CS	full
EE	asst

Union

Denoted by $T_1 \cup T_2$

- where T_1 and T_2 are tables with the same schema.
- The output of the operation is a table T' such that
  - T' has the same schema as T_1 (and hence, T_2).
  - T' contains all the tuples of T_1 and T_2, after removing duplicates.

PROF

pid	**name**	**dept**	**rank**	**sal**
*p*1	Adam	CS	asst	6000
*p*2	Bob	EE	asso	8000
*p*3	Calvin	CS	full	10000
*p*4	Dorothy	EE	asst	5000
*p*5	Emily	EE	asso	8500
*p*6	Frank	CS	full	9000

$\sigma_{\text{sal} \leq 5000}(\text{PROF}) \cup \sigma_{\text{sal} \geq 10000}(\text{PROF})$ returns:

PROF

pid	**name**	**dept**	**rank**	**sal**
*p*3	Calvin	CS	full	10000
*p*4	Dorothy	EE	asst	5000

Set difference

Denoted by $T_1 - T_2$

- where T_1 and T_2 are tables with the same schema.
- The output of the operation is a table T' such that
  - T' has the same schema as T_1 (and hence, T_2).
  - T' contains all the tuples that appear in T_1 but not in T_2.

PROF

pid	name	dept	rank	sal
$p1$	Adam	CS	asst	6000
$p2$	Bob	EE	asso	8000
$p3$	Calvin	CS	full	10000
$p4$	Dorothy	EE	asst	5000
$p5$	Emily	EE	asso	8500
$p6$	Frank	CS	full	9000

$\Pi_{\text{rank}}(\sigma_{\text{sal} \geq 8000}(\text{PROF})) - \Pi_{\text{rank}}(\sigma_{\text{sal} \geq 9000}(\text{PROF}))$ returns:

rank
asso

Cartesian product

Denoted by $T_1 \times T_2$

- where T_1 and T_2 are tables.
- The output of the operation is a table T such that
  - The schema of T includes all the attributes in T_1 and T_2 (if an attribute in T_1 has the same name as an attribute in T_2, they are treated as different attributes in T).
  - For every tuple $t_1 \in T_1$ and $t_2 \in T_2$, T contains a tuple t whose values are the same as t_1 (t_2) on the attributes from T_1 (T_2).

PROF

pid	**name**	**dept**	**rank**	**sal**
$p1$	Adam	CS	asst	6000
$p2$	Bob	EE	asso	8000
$p3$	Calvin	CS	full	10000
$p4$	Dorothy	EE	asst	5000
$p5$	Emily	EE	asso	8500

TEACH

pid	**cid**	**year**
$p1$	$c1$	2011
$p2$	$c2$	2012
$p1$	$c2$	2012

PROF $\times$ TEACH returns the table in the next slide.

pid	name	dept	rank	sal	pid	cid	year
$p1$	Adam	CS	asst	6000	p_1	c_1	2011
$p2$	Bob	EE	asso	8000	p_1	c_1	2011
$p3$	Calvin	CS	full	10000	p_1	c_1	2011
$p4$	Dorothy	EE	asst	5000	p_1	c_1	2011
$p5$	Emily	EE	asso	8500	p_1	c_1	2011
$p1$	Adam	CS	asst	6000	p_2	c_2	2012
$p2$	Bob	EE	asso	8000	p_2	c_2	2012
$p3$	Calvin	CS	full	10000	p_2	c_2	2012
$p4$	Dorothy	EE	asst	5000	p_2	c_2	2012
$p5$	Emily	EE	asso	8500	p_2	c_2	2012
$p1$	Adam	CS	asst	6000	p_1	c_2	2012
$p2$	Bob	EE	asso	8000	p_1	c_2	2012
$p3$	Calvin	CS	full	10000	p_1	c_2	2012
$p4$	Dorothy	EE	asst	5000	p_1	c_2	2012
$p5$	Emily	EE	asso	8500	p_1	c_2	2012