Databases and Information Systems
CS303

Relational Algebra and Relational Calculus
25-08-2023

Relational Algebra

- Basic Relational Operators
- Extended Relational Operators

Extended Relational Operators

- Cannot be expressed by using basic relational operators
  - Generalized projection
  - Aggregation

Extended Relational Operators : Generalized Projection

- $\pi_{F1, F2, \ldots, Fn}(E)$ where F_i s can contain arithmetic operations
- $\pi_{ID,\ name,\ salary\ *1.5}(instructor)$

ID	*name*	*dept_name*	*salary*
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000

instructor

Extended Relational Operators : Aggregation

- $\mathcal{G}_{\text{sum(salary)}}$(instructor)

ID	*name*	*dept_name*	*salary*
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000

instructor

Extended Relational Operators : Aggregation

- $\mathcal{G}_{\text{count-distinct}}(\sigma_{\text{semester} = \text{“Spring”} \wedge \text{year} = 2010}(\text{teaches}))$

ID	*course_id*	*sec_id*	*semester*	*year*
10101	CS-101	1	Fall	2009
10101	CS-315	1	Spring	2010
10101	CS-347	1	Fall	2009
12121	FIN-201	1	Spring	2010
15151	MU-199	1	Spring	2010
22222	PHY-101	1	Fall	2009
32343	HIS-351	1	Spring	2010
45565	CS-101	1	Spring	2010
45565	CS-319	1	Spring	2010
76766	BIO-101	1	Summer	2009
76766	BIO-301	1	Summer	2010
83821	CS-190	1	Spring	2009
83821	CS-190	2	Spring	2009
83821	CS-319	2	Spring	2010
98345	EE-181	1	Spring	2009

teaches

Extended Relational Operators : Aggregation

- dept_name $\mathcal{G}_{\text{dept\_name, sum(salary)}}$(instructor)

ID	*name*	*dept_name*	*salary*
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000

instructor

Extended Relational Operators : Aggregation

- $_{G1,G2...Gn}\mathcal{G}_{F1(A1),\ F2(A2)...\ Fn(An)}(E)$
  - Take the relation given by E and Group by G1, G2... Gn
  - Compute and Display the result of F1(A1), F2(A2)....Fn(An) for each of the small table

Note: Muti-set Relational Algebra

- SQL queries allow repeated results
- Relational algebraic queries eliminate duplicates
- Multi-set Relational algebra is defined to work on multi-sets.

Note: Expressive power of SQL and Relational Algebra

- SELECT A1,A2... An
  FROM r1, r2... rm
  WHERE P
  - $\pi_{A1,A2 \ldots An}(\sigma_P(r1 \times r2 \times \ldots \, rm))$
- SELECT A1,A2, sum(A3)
  FROM r1, r2... rm
  WHERE P
  GROUP BY A1, A2
  - $A1, A2 \; \mathcal{G}_{A1, A2, sum(A3)}(\sigma_P(r1 \times r2 \times \ldots \, rm))$

Note: Expressive power of SQL and Relational Algebra

- SELECT A1,A2, sum(A3)
  FROM r1, r2... rm
  WHERE P
  GROUP BY A1, A2
  HAVING Q
    - Exercise [Assume Q is a condition on A1, A2, sum(A3)]

- Everything that SQL can do has a corresponding relational algebraic operator

Relational Algebra: Uses

- To provide language independent way to represent queries
- Compare and Analyze queries
- Query Optimization using properties of relational operators

Tuple Relational Calculus

Tuple Relational Calculus

- Non-Procedural
  - Specify what is required without specifying how to obtain it

Tuple Relational Calculus: Examples

- Find ID, name dept_name, salary of instructors with salary greater than 80000
  - { t | t ∈ instructor Λ t[salary] > 80000 }

ID	*name*	*dept_name*	*salary*
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000

`instructor`

Tuple Relational Calculus: Examples

- Find ID of instructors with salary greater than 80000
  - $\{ t \mid \exists s \in instructor \ (s[ID] = t[ID] \wedge s[salary] > 80000) \}$

ID	*name*	*dept_name*	*salary*
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000

`instructor`

Tuple Relational Calculus: Examples

- Find the names of all instructors whose department is in the Watson building.
  - $\{ t \mid \exists s \in instructor (t[name] = s[name] \land \exists u \in department (s[dept\_name] = u[dept\_name] \land u[building] = \text{"Watson"})) \}$

ID	name	dept_name	salary
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000

instructor

dept_name	building	budget
Biology	Watson	90000
Comp. Sci.	Taylor	100000
Elec. Eng.	Taylor	85000
Finance	Painter	120000
History	Painter	50000
Music	Packard	80000
Physics	Watson	70000

department

Tuple Relational Calculus: Examples

- Find the set of all course ids taught in the Fall 2009 semester, the Spring 2010 semester, or both.
- $\{ t \mid \quad \exists s \in section\ (t[course\_id] = s[course\_id] \land s[semester] = \text{“Fall”} \land s[year] = 2009)$

  $\lor$

  $\exists u \in section\ (t[course\_id] = u[course\_id] \land u[semester] = \text{“Summer”} \land u[year] = 2010)$

  $\}$

course_id	sec_id	semester	year	building	room_number	time_slot_id
BIO-101	1	Summer	2009	Painter	514	B
BIO-301	1	Summer	2010	Painter	514	A
CS-101	1	Fall	2009	Packard	101	H
CS-101	1	Spring	2010	Packard	101	F
CS-190	1	Spring	2009	Taylor	3128	E
CS-190	2	Spring	2009	Taylor	3128	A
CS-315	1	Spring	2010	Watson	120	D
CS-319	1	Spring	2010	Watson	100	B
CS-319	2	Spring	2010	Taylor	3128	C
CS-347	1	Fall	2009	Taylor	3128	A
EE-181	1	Spring	2009	Taylor	3128	C
FIN-201	1	Spring	2010	Packard	101	B
HIS-351	1	Spring	2010	Painter	514	C
MU-199	1	Spring	2010	Packard	101	D
PHY-101	1	Fall	2009	Watson	100	A

section

Tuple Relational Calculus: Examples

- Find the set of all course ids taught in the Fall 2009 semester, the Spring 2010 semester, or both.
    - { t | ∃ s ∈ section (t[course_id] = s[course_id] ∧ s[semester] = “Fall” ∧ s[year] = 2009)
      ∨
      ∃ u ∈ section (t[course_id] = u[course_id] ∧ u[semester] = “Summer” ∧ u[year] = 2010)
      }

course_id	*sec_id*	*semester*	*year*	*building*	*room_number*	*time_slot_id*
BIO-101	1	Summer	2009	Painter	514	B
BIO-301	1	Summer	2010	Painter	514	A
CS-101	1	Fall	2009	Packard	101	H
CS-101	1	Spring	2010	Packard	101	F
CS-190	1	Spring	2009	Taylor	3128	E
CS-190	2	Spring	2009	Taylor	3128	A
CS-315	1	Spring	2010	Watson	120	D
CS-319	1	Spring	2010	Watson	100	B
CS-319	2	Spring	2010	Taylor	3128	C
CS-347	1	Fall	2009	Taylor	3128	A
EE-181	1	Spring	2009	Taylor	3128	C
FIN-201	1	Spring	2010	Packard	101	B
HIS-351	1	Spring	2010	Painter	514	C
MU-199	1	Spring	2010	Packard	101	D
PHY-101	1	Fall	2009	Watson	100	A

section

Tuple Relational Calculus: Examples

- Find the set of all course ids taught in the Fall 2009 semester but not in the Spring 2010 semester.
  - { t | ∃ s ∈ section (t[course_id] = s[course_id] ∧ s[semester] = "Fall" ∧ s[year] = 2009)
    ∧
    ¬ ∃ u ∈ section (t[course_id] = u[course_id] ∧ u[semester] = "Summer" ∧ u[year] = 2010) }

course_id	*sec_id*	*semester*	*year*	*building*	*room_number*	*time_slot_id*
BIO-101	1	Summer	2009	Painter	514	B
BIO-301	1	Summer	2010	Painter	514	A
CS-101	1	Fall	2009	Packard	101	H
CS-101	1	Spring	2010	Packard	101	F
CS-190	1	Spring	2009	Taylor	3128	E
CS-190	2	Spring	2009	Taylor	3128	A
CS-315	1	Spring	2010	Watson	120	D
CS-319	1	Spring	2010	Watson	100	B
CS-319	2	Spring	2010	Taylor	3128	C
CS-347	1	Fall	2009	Taylor	3128	A
EE-181	1	Spring	2009	Taylor	3128	C
FIN-201	1	Spring	2010	Packard	101	B
HIS-351	1	Spring	2010	Painter	514	C
MU-199	1	Spring	2010	Packard	101	D
PHY-101	1	Fall	2009	Watson	100	A

section

Tuple Relational Calculus: Examples

- Find all student ids who have taken all courses offered in the Biology department.
    - { t | ∃ s ∈ student (t[ID] = s[student_id] ∧
      ∀ u ∈ course (u[dept_name] = “Biology” ⇒ ∃ r ∈ takes r[ID] = s[ID] ∧
      r[course_id] = u[course_id]
)
      }

course_id	title	dept_name	credits
BIO-101	Intro. to Biology	Biology	4
BIO-301	Genetics	Biology	4
BIO-399	Computational Biology	Biology	3
CS-101	Intro. to Computer Science	Comp. Sci.	4
CS-190	Game Design	Comp. Sci.	4
CS-315	Robotics	Comp. Sci.	3
CS-319	Image Processing	Comp. Sci.	3
CS-347	Database System Concepts	Comp. Sci.	3
EE-181	Intro. to Digital Systems	Elec. Eng.	3
FIN-201	Investment Banking	Finance	3
HIS-351	World History	History	3
MU-199	Music Video Production	Music	3
PHY-101	Physical Principles	Physics	4

course

ID	name	dept_name	tot_cred
00128	Zhang	Comp. Sci.	102
12345	Shankar	Comp. Sci.	32
19991	Brandt	History	80
23121	Chavez	Finance	110
44553	Peltier	Physics	56
45678	Levy	Physics	46
54321	Williams	Comp. Sci.	54
55739	Sanchez	Music	38
70557	Snow	Physics	0
76543	Brown	Comp. Sci.	58
76653	Aoi	Elec. Eng.	60
98765	Bourikas	Elec. Eng.	98
98988	Tanaka	Biology	120

student

ID	course_id	sec_id	semester	year	grade
00128	CS-101	1	Fall	2009	A
00128	CS-347	1	Fall	2009	A-
12345	CS-101	1	Fall	2009	C
12345	CS-190	2	Spring	2009	A
12345	CS-315	1	Spring	2010	A
12345	CS-347	1	Fall	2009	A
19991	HIS-351	1	Spring	2010	B
23121	FIN-201	1	Spring	2010	C+
44553	PHY-101	1	Fall	2009	B-
45678	CS-101	1	Fall	2009	F
45678	CS-101	1	Spring	2010	B+
45678	CS-319	1	Spring	2010	B
54321	CS-101	1	Fall	2009	A-
54321	CS-190	2	Spring	2009	B+
55739	MU-199	1	Spring	2010	A-
76543	CS-101	1	Fall	2009	A
76543	CS-319	2	Spring	2010	A
76653	EE-181	1	Spring	2009	C
98765	CS-101	1	Fall	2009	C-
98765	CS-315	1	Spring	2010	B
98988	BIO-101	1	Summer	2009	A
98988	BIO-301	1	Summer	2010	*null*

takes

Tuple Relational Calculus: Formal Definition

- { t | P(t) } where P is a formula.

- Atom:
  - s ∈ relation
  - s[attribute1] # u[attribute2] s[attribute] # c
    # can be = < <= >= >
    s[attribute1] and u[attribute2] should have same domain
    s[attribute] and constant c should have same attribute

- Formulas:
  - An Atom is a formula
  - If P_1 is a formula and P_2 is a formula then:
    - $P_1 \wedge P_2$ $P_1 \vee P_2$ $P_1 \Rightarrow P_2$ are formulas
  - If P is a formula then
    - $\exists s\, P$ and $\forall s\, P$ are formulas
  - If P is a formula then $\neg P$ is a formula

Tuple Relational Calculus: Safety Expression

- $\{ t \mid \neg (t \in \text{relation}) \}$
- Generates infinite tuples (including those that are not in the database)
- We need a notion of (Finite) Domain / Universe to talk about negation

Tuple Relational Calculus: Expressive power

- Tuple Relational Calculus (restricted to safe expressions) has same expressive power as Basic Relational Algebra ($\cup$ - X σ π ρ)

- Tuple Relational Calculus (restricted to safe expressions) cannot express generalized relational operators

Domain Relational Calculus

Domain Relational Calculus

- Uses domain variables instead of tuple variables
- Closely related to Tuple Relational Calculus

Domain Relational Calculus : Formal definition

- $\{ < x_1 x_2 x_3 \ldots x_n > \mid P(x_1 x_2 x_3 \ldots x_n) \}$ where $x_1 x_2 x_3 \ldots x_n$ are domain variables
  P is a formula

- P can be of the form:
  - $< x_1 x_2 x_3 \ldots x_m > \in r$ where r is a relation with m attributes
  - x # y x # c where c is a constant of same domain as x
    # can be = < <= >= >
  - If P_1 is a formula, then $\neg P_1$ and (P_1) are formulas
  - If P_1 and P_2 are formulas then
    - $P_1 \wedge P_2$ $P_1 \vee P_2$ $P_1 \Rightarrow P_2$ are formulas
  - If P is a formula that contains x as domain variable then
    - $\exists s\, P$ and $\forall s\, P$ are formulas

Domain Relational Calculus : Examples

- Find the instructor ID, name, dept name, and salary for instructors whose salary is greater than 80000:
    - $\{ < i, n, d, s > \mid < i, n, d, s > \in instructor \wedge s > 80000 \}$

ID	*name*	*dept_name*	*salary*
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000

`instructor`

Domain Relational Calculus : Examples

- Find the instructor ID, name for instructors whose salary is greater than 80000:
  - $\{ < i, n > \mid \exists\ d, s\ \ < i, n, d, s > \in instructor\ \wedge\ s > 80000 \}$

ID	*name*	*dept_name*	*salary*
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000

`instructor`

Domain Relational Calculus : Safety Expressions

- $\{ < i, n, d, s > \mid \neg(< i, n, d, s > \in instructor) \}$
- $\{ < x > \mid \exists\ y (< x, y > \in r) \wedge \exists\ z (\neg(< x, z > \in r) \wedge P(x, z)) \}$
- An expression of the form P is safe if:
    - All values that appear in tuples of the expression are values from dom(P).
    - For every "there exists" subformula of the form $\exists\ x\ (P1\ (x))$, the subformula is true if and only if there is a value x in dom(P1) such that P1 (x) is true.
    - For every "for all" subformula of the form $\forall x\ (P1\ (x))$, the subformula is true if and only if P1 (x) is true for all values x from dom(P1).

Domain Relational Calculus : Expressive Power

- Safe domain relational calculus have same expressive power as same tuple relational calculus

- All the following have same expressive power:
  - Basic operators of Relational Algebra
  - Tuple Relational Calculus restricted to safe expressions
  - Domain Relational Calculus restricted to safe expressions

Exercise

- Find the names of all instructors in the Physics department together with the course id of all courses they teach.

ID	name	dept_name	salary
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000

instructor

ID	course_id	sec_id	semester	year
10101	CS-101	1	Fall	2009
10101	CS-315	1	Spring	2010
10101	CS-347	1	Fall	2009
12121	FIN-201	1	Spring	2010
15151	MU-199	1	Spring	2010
22222	PHY-101	1	Fall	2009
32343	HIS-351	1	Spring	2010
45565	CS-101	1	Spring	2010
45565	CS-319	1	Spring	2010
76766	BIO-101	1	Summer	2009
76766	BIO-301	1	Summer	2010
83821	CS-190	1	Spring	2009
83821	CS-190	2	Spring	2009
83821	CS-319	2	Spring	2010
98345	EE-181	1	Spring	2009

teaches

Exercise

- Find the set of all course IDs taught in the Fall 2009 semester, the Spring 2010 semester, or both:

course_id	*sec_id*	*semester*	*year*	*building*	*room_number*	*time_slot_id*
BIO-101	1	Summer	2009	Painter	514	B
BIO-301	1	Summer	2010	Painter	514	A
CS-101	1	Fall	2009	Packard	101	H
CS-101	1	Spring	2010	Packard	101	F
CS-190	1	Spring	2009	Taylor	3128	E
CS-190	2	Spring	2009	Taylor	3128	A
CS-315	1	Spring	2010	Watson	120	D
CS-319	1	Spring	2010	Watson	100	B
CS-319	2	Spring	2010	Taylor	3128	C
CS-347	1	Fall	2009	Taylor	3128	A
EE-181	1	Spring	2009	Taylor	3128	C
FIN-201	1	Spring	2010	Packard	101	B
HIS-351	1	Spring	2010	Painter	514	C
MU-199	1	Spring	2010	Packard	101	D
PHY-101	1	Fall	2009	Watson	100	A

section

Exercise

- Find all students who have taken all courses offered in the Biology department

ID	name	dept_name	tot_cred
00128	Zhang	Comp. Sci.	102
12345	Shankar	Comp. Sci.	32
19991	Brandt	History	80
23121	Chavez	Finance	110
44553	Peltier	Physics	56
45678	Levy	Physics	46
54321	Williams	Comp. Sci.	54
55739	Sanchez	Music	38
70557	Snow	Physics	0
76543	Brown	Comp. Sci.	58
76653	Aoi	Elec. Eng.	60
98765	Bourikas	Elec. Eng.	98
98988	Tanaka	Biology	120

student

course_id	title	dept_name	credits
BIO-101	Intro. to Biology	Biology	4
BIO-301	Genetics	Biology	4
BIO-399	Computational Biology	Biology	3
CS-101	Intro. to Computer Science	Comp. Sci.	4
CS-190	Game Design	Comp. Sci.	4
CS-315	Robotics	Comp. Sci.	3
CS-319	Image Processing	Comp. Sci.	3
CS-347	Database System Concepts	Comp. Sci.	3
EE-181	Intro. to Digital Systems	Elec. Eng.	3
FIN-201	Investment Banking	Finance	3
HIS-351	World History	History	3
MU-199	Music Video Production	Music	3
PHY-101	Physical Principles	Physics	4

course

ID	course_id	sec_id	semester	year	grade
00128	CS-101	1	Fall	2009	A
00128	CS-347	1	Fall	2009	A-
12345	CS-101	1	Fall	2009	C
12345	CS-190	2	Spring	2009	A
12345	CS-315	1	Spring	2010	A
12345	CS-347	1	Fall	2009	A
19991	HIS-351	1	Spring	2010	B
23121	FIN-201	1	Spring	2010	C+
44553	PHY-101	1	Fall	2009	B-
45678	CS-101	1	Fall	2009	F
45678	CS-101	1	Spring	2010	B+
45678	CS-319	1	Spring	2010	B
54321	CS-101	1	Fall	2009	A-
54321	CS-190	2	Spring	2009	B+
55739	MU-199	1	Spring	2010	A-
76543	CS-101	1	Fall	2009	A
76543	CS-319	2	Spring	2010	A
76653	EE-181	1	Spring	2009	C
98765	CS-101	1	Fall	2009	C-
98765	CS-315	1	Spring	2010	B
98988	BIO-101	1	Summer	2009	A
98988	BIO-301	1	Summer	2010	*null*

takes

Summary

- Other representations: Datalog, Conjunctive queries ....
- Relational Calculus has close connections with Mathematical Logic
- The Field of Finite Model Theory was motivated by Databases

Reference:

Database System Concepts by Silberschatz, Korth and Sudarshan

Chapter 6