Relational Algebra

Key = attribute set that uniquely identifies each row in the relation/table

Customer relation [edit]

Customer ID	Tax ID	Name	Address	[More fields…]
1234567890	555-5512222	Ramesh	323 Southern Avenue	…
2223344556	555-5523232	Adam	1200 Main Street	…
3334445563	555-5533323	Shweta	871 Rani Jhansi Road	…
4232342432	555-5325523	Sarfaraz	123 Maulana Azad Sarani	…

Keys:
Customer ID
Coustomer ID, Tax ID
……

Keys

Superkey

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

Candidate key

- Minimal such set of attributes

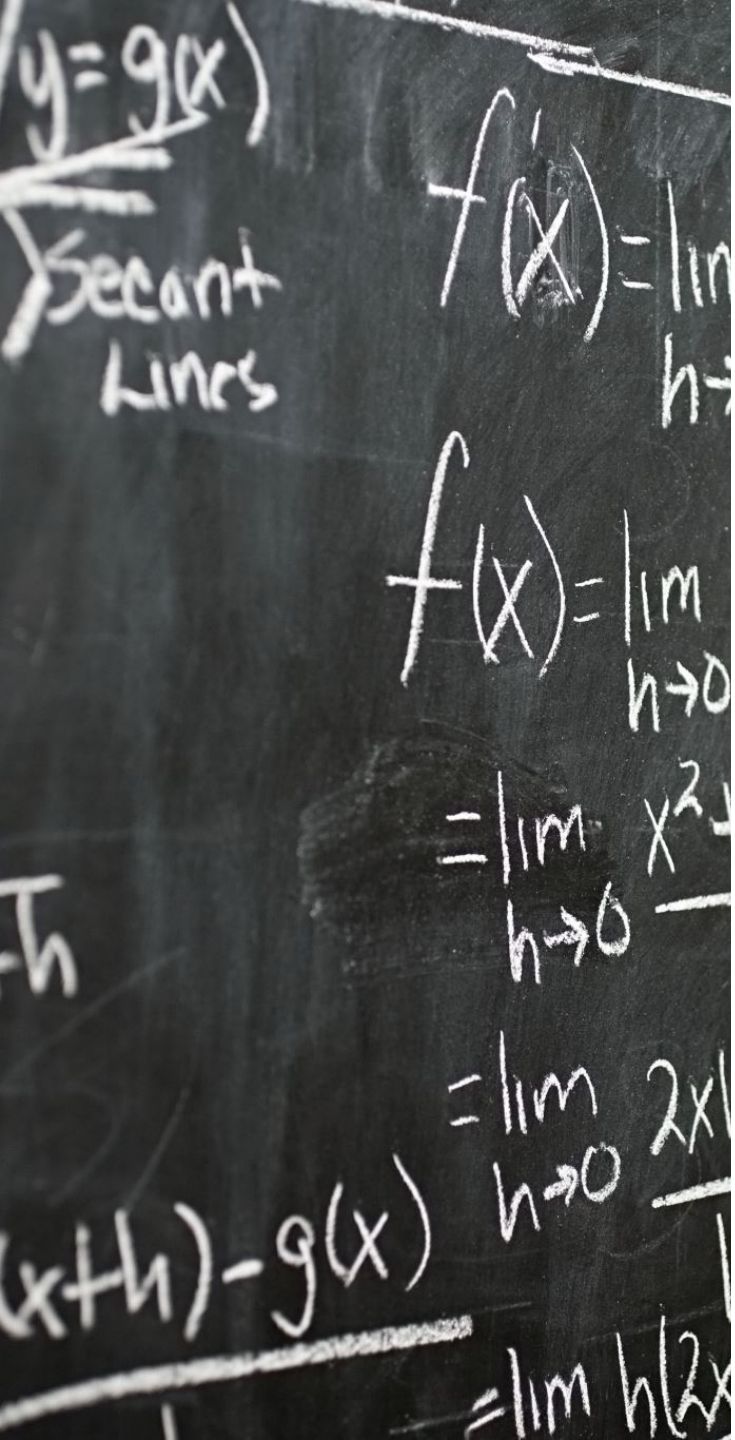
Primary key

- DB Chosen Candidate key
- Plays a very important role
  - E.g. relations typically sorted by this

醫事機構代碼	醫事機構名稱	醫事機構地址	醫事機構電話	成人口罩剩餘數	兒童口
145080011	衛生福利部花蓮醫院豐濱	花蓮縣豐濱鄉豐濱村光豐路４１號	(03)8358141	3304	
291010010	連江縣立醫院	連江縣南竿鄉復興村２１７號	(083)623995	0	
2101010013	松山健康服務中心	臺北市松山區八德路４段６９４號１、２樓	(02)27653147	1	
2101020019	大安健康服務中心	臺北市大安區辛亥路３段１５號	(02)27390997	0	
2101090011	大同健康服務中心	臺北市大同區昌吉街５２號１至２樓	(02)25948971	0	
2101100227	中山健康服務中心	臺北市中山區松江路３６７號１樓	(02)25013363	2	
2101110027	內湖健康服務中心	臺北市內湖區民權東路６段９９號	(02)27908387	0	
2101120014	南港健康服務中心	臺北市南港區南港路１段３６０號１樓	(02)27868756	0	
2101150012	士林健康服務中心	臺北市士林區中正路４３９號１樓	(02)28836268	0	
2101161033	北投健康服務中心	臺北市北投區石牌路２段１１１號	(02)28261026	0	
2101170050	信義健康服務中心	臺北市信義區信義路５段１５號	(02)87804152	0	
2101180038	中正健康服務中心	臺北市中正區牯嶺街２４號	(02)23210168	0	
2101191068	萬華健康服務中心	臺北市萬華區東園街１５２號	(02)23395384	0	
2101200017	政大(文山健康服務中心)	臺北市文山區木柵路３段２２０號	(02)22343501	0	
2303240013			(04)22223811	2028	

Example of FaceMask dataset

- Superkey
  - {代碼}, {名稱} ,{電話} , {代碼, 名稱}, {代碼, 電話}, {代碼, 口罩剩餘數}
- Candidate key:
  - {代碼, 電話} is not a miminal set. Since {電話} could be removed from {代碼, 電話} while the remining attribute {代碼} is still a key
  - {代碼}, {名稱}, {電話}, {地址}



Introduction

- Query languages are specialized languages for asking questions or queries, that involve the data in a database.
- Relational algebra : queries in terms of operators
- Every operator in relational algebra accepts (one or two) relation instances as arguments and returns a relation instance as the result.

  (1 + 2 = 3) +(1,2) = 3

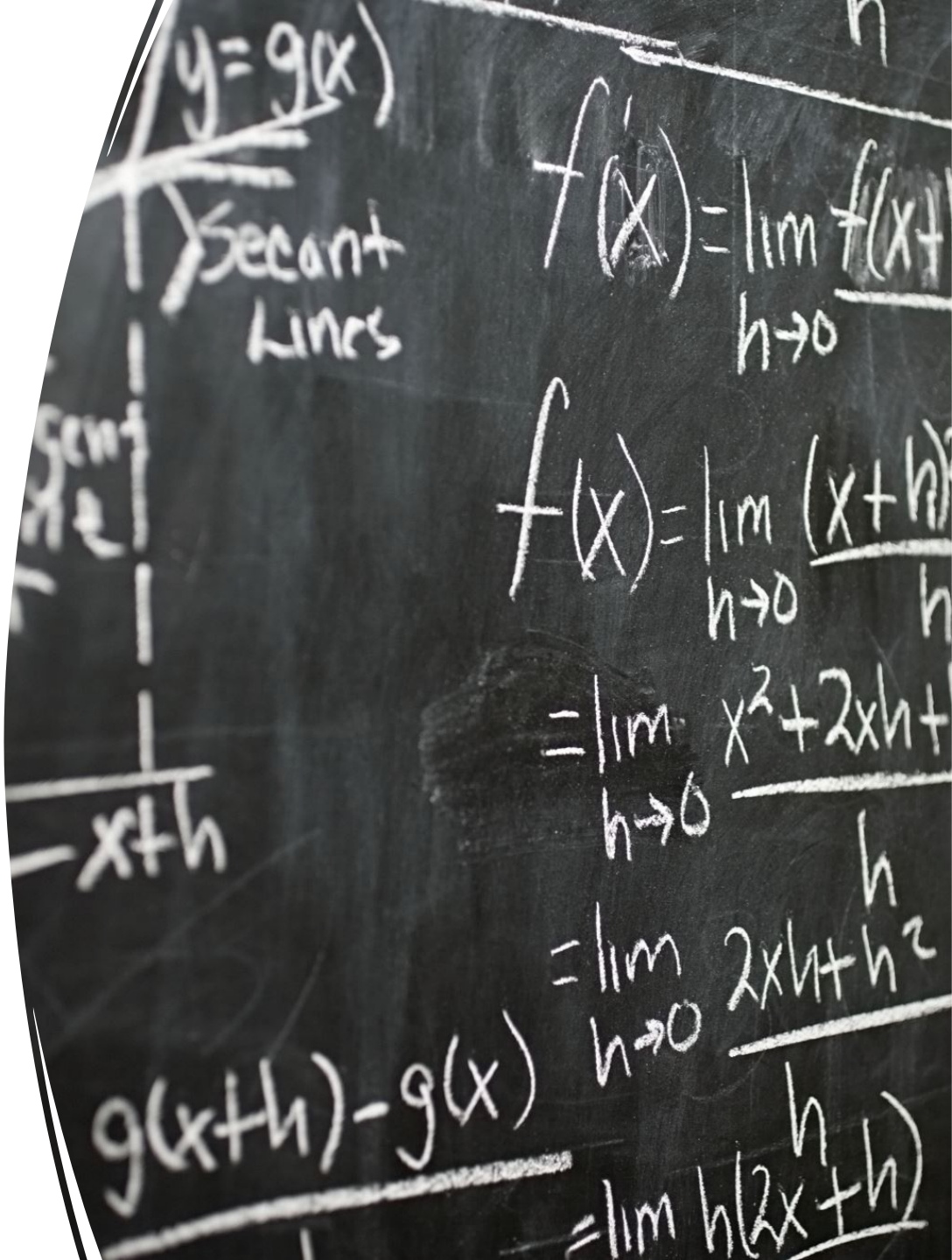
Introduction

(1 + 2 = 3) +(1,2) = 3

- A relational algebra expression is recursively defined to be a relation.
- Relational algebra is a procedural query language
  - Defines a step-by-step procedure for computing the answer.

Basic Operators

- There are six basic operators in relation algebra:
  - select (σ)
  - project (Π)
  - union (U)
  - set different ($-$)
  - Cartesian product (x)
  - rename (ρ)



Selection (σ)

σ

Relation R

a	b	c	d
α	α	1	7
α	β	5	7
β	β	12	3
β	β	23	10

$\sigma_{a=b \wedge d>5}(R)$

a	b	c	d
α	α	1	7
β	β	23	10

Selection (σ)

- The selection operation σ specifies the tuples to retain through a *selection condition*.

- Selection condition is a Boolean combination (an expression using logical connectives ∧ and ∨) of terms that have the form
    - attribute op constant (e.g. A = 3)
    - attribute1 op attribute2 (e.g. X < Y)

  (op is one of the comparison operators
  <, ≤, =, ≠, >, ≥).

Projection (Π)

Π

Relation R

a	b	c
α	10	1
α	20	1
β	30	1
β	40	2

$\Pi_{a,c}(R)$

a	c
α	1
α	1
β	1
β	2

→ Eliminate Duplicated rows

a	c
α	1
β	1
β	2

Projection (Π)

- The projection operator Π allows us to extract columns from a relation.
- The subscript specifies the fields to be retained.
  (The other fields are 'projected out').
- The schema of the result of a projection is determined by the fields that are projected.
- Duplicated row will be eliminated in the final result.
  - This follows from the definition of a relation as a set of tuples.

Employee

f_name	l_name	id	gender	salary	superid	dno
Joseph	Chan	999999	M	29500	654321	4
Victor	Wong	001100	M	30000	888555	5
Carrie	Kwan	898989	F	26000	654321	4
Joyce	Fong	345345	F	12000	777888	4

Find all employees who works in department 4 and whose salary is greater than 25000.

$\sigma_{dno=4 \wedge salary>25000}$ (Employee)

f_name	l_name	id	gender	salary	superid	dno
Joseph	Chan	999999	M	29500	654321	4
Carrie	Kwan	898989	F	26000	654321	4

1. Find the employee names and department number of all employees

2. Find the gender and department number of all employees

$\Pi_{lname,fname,dno}$ (Employee)

f_name	l_name	dno
Joseph	Chan	4
Victor	Wong	5
Carrie	Kwan	4
Joyce	Fong	4

$\Pi_{gender,dno}$ (Employee)

gender	dno
M	4
M	5
F	4

- Compositing operations
- Substituting an expression where a relation is expected
- The result of an relational-algebra expression is always a relation.

$$\Pi_{\text{lname,fname}} (\sigma_{\text{dno=4^salary>25000}} (\text{Employee}))$$

f_name	l_name
Joseph	Chan
Carrie	Kwan

f_name	l_name	id	gender	salary	superid	dno
Joseph	Chan	999999	M	29500	654321	4
Victor	Wong	001100	M	30000	888555	5
Carrie	Kwan	898989	F	26000	654321	4
Joyce	Fong	345345	F	12000	777888	4

Union

R

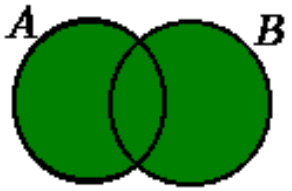
a	b
α	1
α	2
β	1

S

a	b
α	2
β	3

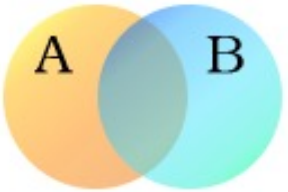
R U S

a	b
α	1
α	2
β	1
β	3



- R ∪ S returns a relation instance containing all tuples that occur in either relation R or relation S (or both).
- The union operation is commutative: R ∪ S = S ∪ R

- Duplicated tuples are eliminated.
- R and S must be *union-compatible*.
  - They have the same number of fields,
  - The corresponding fields have the same domains.
    - The field names are not so important
- The schema of the result is defined to be identical to the schema of R.
  - The fields of R ∪ S inherit names from R.



Set Difference

R

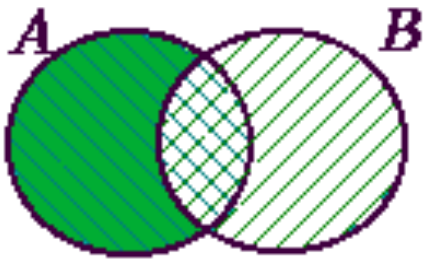
a	b
α	1
α	2
β	1

S

a	b
α	2
β	3

R - S

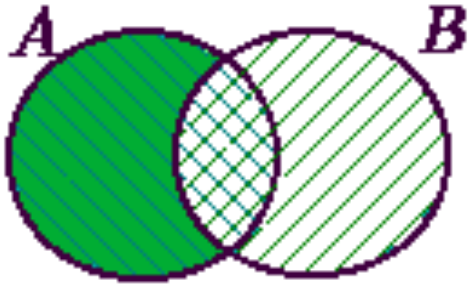
a	b
α	1
β	1



Set Difference

- Set difference $R - S$

  returns a relation instance containing all the tuples that occur in R but not in S.



- Set difference is not commutative:

  in general, $R - S \neq S - R$.

- The relations R and S must be union-compatible.
- The schema of the result is defined to be identical to the schema of R.

Intersection

R

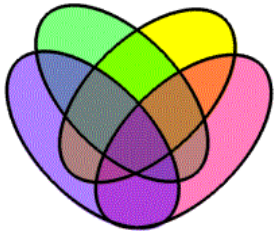
a	b
α	1
α	2
β	1

S

a	b
α	2
β	3

R ∩ S

a	b
α	2

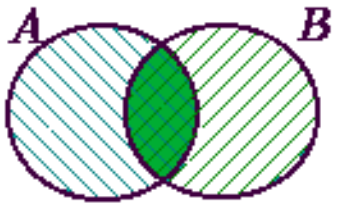


Intersection

- R ∩ S returns a relation instance containing all tuples that occur in both R and S.
- The relations R and S must be union-compatible.
- The schema of the result is defined to be identical to the schema of R.

- Intersection is not considered a basic operation, as it can be derived from the basic operations as shown below:

$$R \cap S = R - (R - S)$$



Cartesian-Product (×)

R

a	b
α	1
β	2

S

c	d	e
α	10	+
β	10	+
β	20	-
γ	10	-

R × S

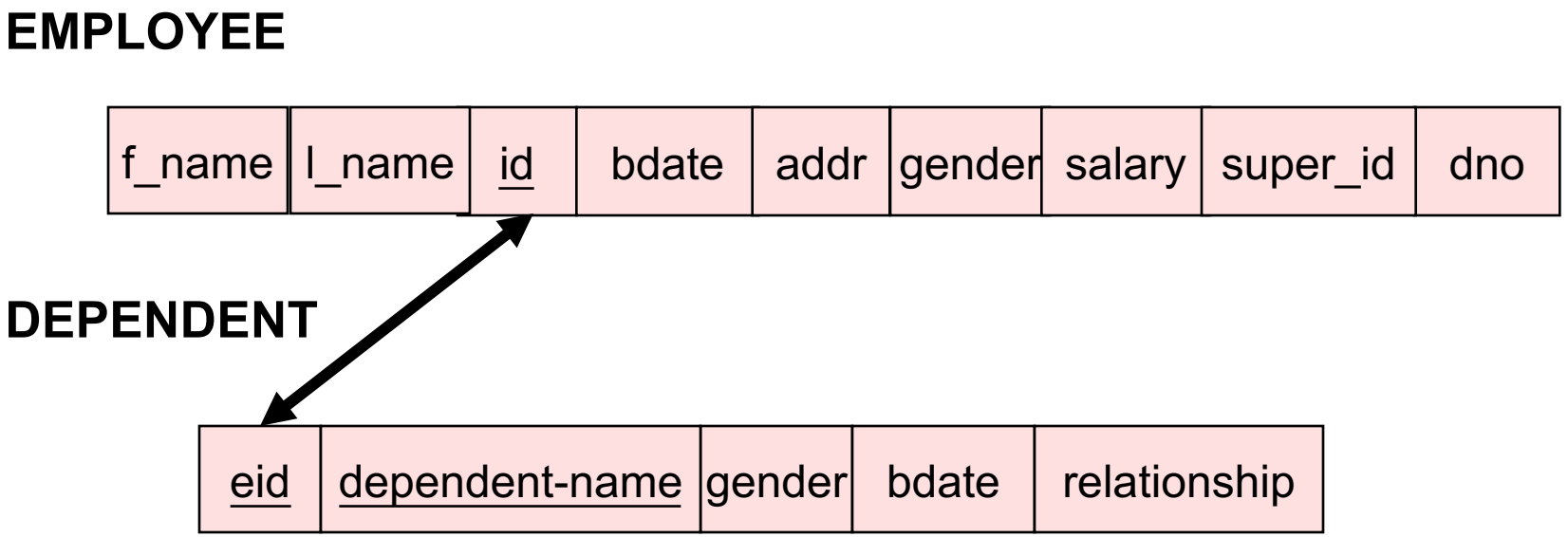
a	b	c	d	e
α	1	α	10	+
α	1	β	10	+
α	1	β	20	-
α	1	γ	10	-
β	2	α	10	+
β	2	β	10	+
β	2	β	20	-
β	2	γ	10	-

Cartesian product (Cross product)

- R $\times$ S returns a relation instance whose schema contains all the fields of R followed by all the fields of S.

- The result contains one tuple *<r,s>* (concatenation of tuples *r* and *s*)
  for each pair of tuples r $\in$ R, s $\in$ S.

Query: for each female employee, retrieve employee's name and a list of the names of her dependents.



$$\text{Female\_emps} \leftarrow \sigma_{\text{gender='F'}} (\text{Employee})$$

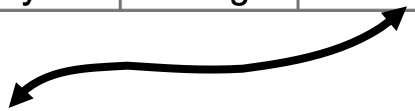
$$\text{Empnames} \leftarrow \Pi_{\text{fname,lname,id}} (\text{Female\_emps})$$

Female_emps

f_name	l_name	id	bdate	address	gender	salary	superid	dno
Alicia	Chan	998877	2-Jul-70	231, Cai Road	F	9500	654321	4
Jennifer	Wong	654321	20-June-60	342, Cheung Road	F	30000	888555	4
Joyce	Fong	345345	19-Dec-80	23, Young Road	F	12000	777888	5

Empnames

f_name	l_name	id
Alicia	Chan	998877
Jennifer	Wong	654321
Joyce	Fong	345345



Dependents

eid	dep_name	gender	bdate	relationship
334455	Alice	F	5-Apr-90	Daughter
334455	Theodore	M	3-Mar-92	Son
654321	Abner	M	29-Feb-94	Son
123456	Alice	F	2-Nov-97	Daughter

Emp_dependents ← Empnames x Dependents

Emp_dependents

f_name	l_name	id	eid	dep_name	gender	bdate	relationship
Alicia	Chan	998877	334455	Alice	F	5-Apr-90	Daughter
Alicia	Chan	998877	334455	Theodore	M	3-Mar-92	Son
Alicia	Chan	998877	654321	Abner	M	29-Feb-94	Son
Alicia	Chan	998877	123456	Alice	F	2-Nov-97	Daughter
Jennifer	Wong	654321	334455	Alice	F	5-Apr-90	Daughter
Jennifer	Wong	654321	334455	Theodore	M	3-Mar-92	Son
Jennifer	Wong	654321	654321	Abner	M	29-Feb-94	Son
Jennifer	Wong	654321	123456	Alice	F	2-Nov-97	Daughter
Joyce	Fong	345345	334455	Alice	F	5-Apr-90	Daughter
Joyce	Fong	345345	334455	Theodore	M	3-Mar-92	Son
Joyce	Fong	345345	654321	Abner	M	29-Feb-94	Son
Joyce	Fong	345345	123456	Alice	F	2-Nov-97	Daughter

Query: for each female employee, retrieve employee's name and a list of the names of her dependents

Actual_dependents $\leftarrow \sigma_{id=eid}$ (Emps_dependents)

Result $\leftarrow \Pi_{fname,lname,dep\_name}$ (Actual_dependents)

Actual_dependents

f_name	l_name	id	eid	dep_name	gender	bdate	relationship
Jennifer	Wong	654321	654321	Abner	M	29-Feb-94	Son

Result

f_name	l_name	dep_name
Jennifer	Wong	Abner

Join ⋈

- Because the sequence of operations, X followed by σ is quite common, a special operations, called the "join" operation ⋈ was created to specify these as a single operation.

- Join can be defined as a cross-product followed by selections and sometimes with projections.

Query: for each female employee, retrieve employee's name and a list of the names of her dependents.

$$\text{Female\_emps} \leftarrow \sigma_{\text{gender='F'}} (\text{Employee})$$

$$\text{Empnames} \leftarrow \pi_{\text{fname,lname,id}} (\text{Female\_emps})$$

$$\text{Actual\_dependents} \leftarrow \text{empnames} \bowtie_{\text{id=eid}} \text{Dependents}$$

$$\text{Result} \leftarrow \pi_{\text{fname,lname,dep\_name}} (\text{Actual\_dependents})$$

Female_emps

f_name	l_name	id	bdate	address	gender	salary	superid	dno
Alicia	Chan	998877	2-Jul-70	231, Cai Road, HK	F	9500	654321	4
Jennifer	Wong	654321	20-June-60	342, Cheung Road, HK	F	30000	888555	4
Joyce	Fong	345345	19-Dec-80	23, Young Road, HK	F	12000	777888	5

Empnames ⋈ Dependents

f_name	l_name	id
Alicia	Chan	998877
Jennifer	Wong	654321
Joyce	Fong	345345

eid	dep_name	gender	bdate	relationship
334455	Alice	F	5-Apr-90	Daughter
334455	Theodore	M	3-Mar-92	Son
654321	Abner	M	29-Feb-94	Son
123456	Alice	F	2-Nov-97	Daughter

Actual_dependents

f_name	l_name	id	eid	dep_name	gender	bdate	relationship
Jennifer	Wong	654321	654321	Abner	M	29-Feb-94	Son

Result

f_name	l_name	dep_name
Jennifer	Wong	Abner

Conditional Join

- The most general version of join operation accepts a join condition c.
- The join condition is identical to a selection condition in form.
- The operation is defined as follows:

$$R \bowtie_C S = \sigma_C(R \times S)$$

S

sid	sname	rating	age
22	Dustin	7	45.0
31	Lubber	8	55.5
58	Rusty	10	35.5

R

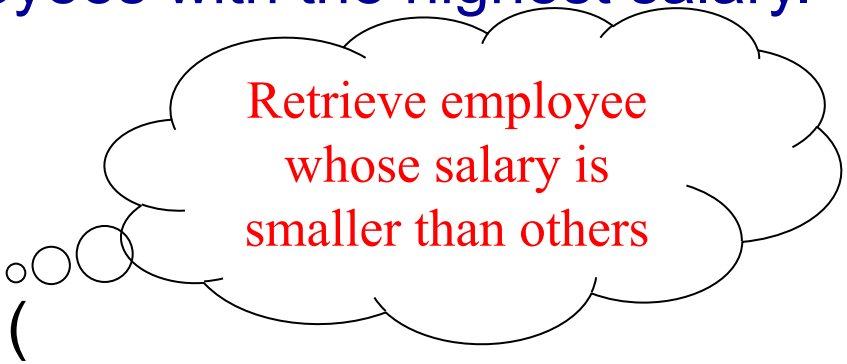
sid	bid	day
22	101	10/10/96
58	103	11/12/96

$$S \bowtie_{S.sid<R.sid} R$$

(sid)	sname	rating	age	(sid)	bid	day
22	Dustin	7	45.0	58	103	11/12/96
31	Lubber	8	55.5	58	103	11/12/96

Example

Query: Find the names of employees with the highest salary.



$$\Pi_{\text{lname,fname}} (\text{Employee}) - \Pi_{\text{Employee.lname, Employee.fname}} (\text{Employee} \bowtie_{\text{Employee.salary<F.salary}} \rho (\text{F, Employee}))$$

$$\text{Employee} \bowtie_{\text{Employee.salary>F.salary}} \rho (\text{F, Employee})$$

Cannot work ? Why ?

Employee

eid	lname	salary
22	Dustin	7
31	Lubber	8
58	Rusty	10

Employee as F

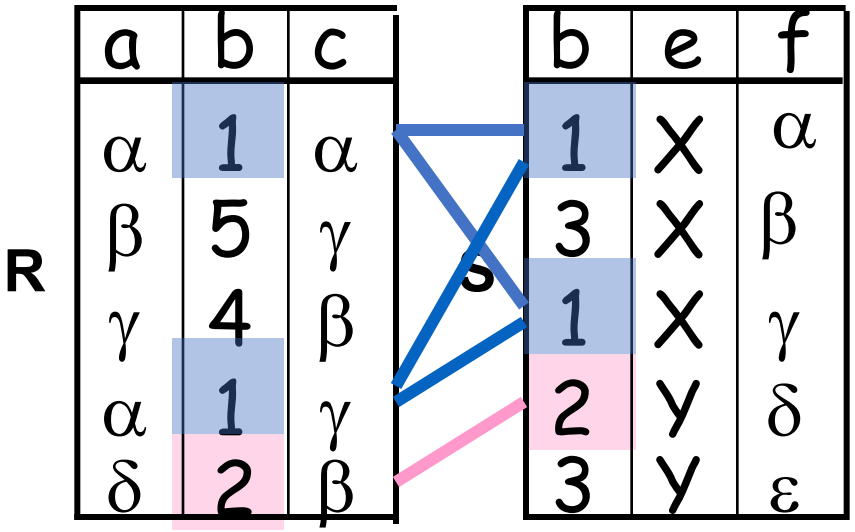
eid	lname	salary
22	Dustin	7
31	Lubber	8
58	Rusty	10

$$\text{Employee} \bowtie_{\text{Employee.salary<F.salary}} \text{F}$$

eid	lname	salary	(eid)	lname	salary
22	Dustin	7	31	Lubber	8
31	Dustin	7	58	Rusty	10
31	Lubber	8	58	Rusty	10

Equi-Join

A condition join where the condition contains only equalities.
The conditions of the join have the form R.name1 = S.name2, where we want to join R with S.
In the resulting relation, S.name2 will be dropped.



$R \bowtie_{R.b=S.b} S$

a	b	c	e	f
α	1	α	X	α
α	1	α	X	γ
α	1	γ	X	α
α	1	γ	X	γ
δ	2	β	Y	δ

Natural Join ⋈

An Equijoin on all common fields.

For R ⋈ S, the resulting schema contains the attributes of R followed by the attributes in S that are not in R.

If two relations have no attributes in common, the result is simply the cross-product.

R

a	b	c	d
α	1	α	X
β	2	γ	X
γ	4	β	Y
α	1	γ	Y
δ	2	β	Y

S

b	d	e
1	X	α
3	X	β
1	X	γ
2	Y	δ
3	Y	ε

R ⋈ S

a	b	c	d	e
α	1	α	X	α
α	1	α	X	γ
δ	2	β	Y	δ

Division

A

x	y
α	1
α	2
α	3
β	1
γ	1
δ	1
δ	3
δ	4
ε	1
ε	2

B

y
1
2

A / B

x
α
ε

The division operation A/B is the set of all x values
(in the form of unary tuples)

such that for every y value in a tuple of B,
there is a tuple <x,y> in A.

A

x	y
α	1
α	2
α	3
β	1
γ	1
δ	1
δ	3
δ	4
ε	1
ε	2

B

y
1
2

A / B

x
α
ε

Another way to understand division:

For each x value in A, consider the set of y values that appear in tuples of A with that x value.

If this set contains all y values in B, the x value is in the result of A/B.

Examples of Division A/B

sno	pno
s1	p1
s1	p2
s1	p3
s1	p4
s2	p1
s2	p2
s3	p2
s4	p2
s4	p4

A

pno
p2

B1

sno
s1
s2
s3
s4

A/B1

pno
p2
p4

B2

sno
s1
s4

A/B2

pno
p1
p2
p4

B3

sno
s1

A/B3

Employee	fname	lname	id	bdate	address	salary	sid	dno

Works_on	id	pno

Query: Retrieve the names of employees who work on all the projects that `Chris Peng' works on.

PENG $\leftarrow \sigma_{\text{fname="Chris" ^ lname="Peng"}}$ (Employee)

PENG_pnos $\leftarrow \Pi_{\text{pno}}$ (Works_on $\bowtie$ PENG)

fname	lname	id	bdate	address	salary	sid	dno	pno

pno

Result_id $\leftarrow$ Works_on / PENG_pnos

Result $\leftarrow \Pi_{\text{fname,lname}}$ (Result_id $\bowtie$ Employee)

User (ID, PID, Name)
Inbound (PID, Origin, flight_no, date)
Airport (IATA code, city, country)

Query 1: Find PID of users who traveled UK and were back Taiwan on 2020/03/16

- Solution 1

Π_{PID} ($\sigma_{country=UK}$ (Airport) $\bowtie$ $\sigma_{date=2020/03/16}$ (Inbound))

- Solution 2:

Π_{PID}($\sigma_{country=UK \text{ and } date=2020/03/16}$ (Airport $\bowtie$ Inbound))

Query 2: Find the ID of users who traveled to UK and were back Taiwan on 2020/03/16

- Solution 1:

$$\Pi_{ID}((\sigma_{country=UK}(Airport) \bowtie \sigma_{date=2020/03/16}(Inbound)) \bowtie User)$$

- Solution 2:

$$\Pi_{ID}(\Pi_{PID}((\Pi_{LATA\ code}(\sigma_{country=UK}\ Airport)) \bowtie \sigma_{date=2020/03/16}(Inbound)) \bowtie User)$$

Query 3: Find the ID of users who were back from UK or Japan.

- Solution 1:

ρ (TempAirport, (σ $_{country=UK\ V\ country=Japan}$ Airport))

Π $_{ID}$ (TempAirport $\bowtie$ Inbound $\bowtie$ User)

- Solution 2

Π $_{ID}$ (((σ $_{country=UK}$ Airport) $\bowtie$ Inbound) $\bowtie$ User) U

Π $_{ID}$ (((σ $_{country=Japan}$ Airport) $\bowtie$ Inbound) $\bowtie$ User)

Query 4: Find the PID of users who were in the same plant of one confirmed case with his PID as TW0001 on 2020/02/28

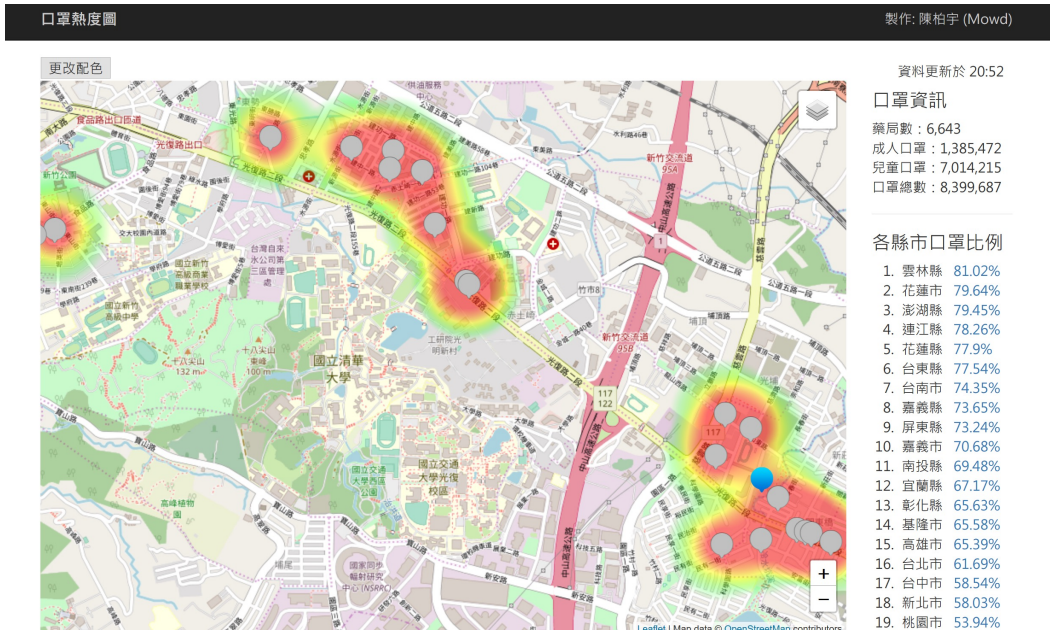
-Solution 1

$\Pi_{\text{PID, flight\_no}} (\sigma_{\text{Date=2020/02/28}} \text{Inbound}) / \Pi_{\text{flight\_no}} (\sigma_{\text{PID=TW001 and Date=2020/02/28}} \text{Inbound})$

-Any other solution ?

Covid 2019 HackMD

Open Data Taiwan FB Group



Maskdata (ID, name, address, Tel, adult_no, child _no, time)

Additional Relational Operations

- AGGREGATE FUNCTIONS
  - Functions such as *SUM*, *COUNT*, *AVERAGE*, *MIN*, *MAX* are often applied to sets of values or sets of tuples in database applications
  - <grouping attributes> *F* <function list> (R)
    - The grouping attributes are optional
    - e.g. Retrieve the average salary of all employees (no grouping needed):

    R(AVGSAL) ← $F_{AVERAGE\ SALARY}$ (EMPLOYEE)

Query 1: Find the average number of adult face mask in each clinic .

$$R(ID, ave\_adult) = {}_{ID}\mathcal{F}_{avg(adult\_no)}(Maskdata)$$

Query 2: Find the phone number of the clinic ID that has the maximal number of adult face mask

Step 1: $R(ID, MAX\_ID\_adult) =_{ID} F_{MAX(adult\_no)}(Maskdata)$

ID	MAX_ID_adult

Step 2: $R(ID, Final\_MAX) =_{ID} F_{MAX(MAX\_ID\_adult)}(Maskdata)$

ID	MAX_ID_adult

→

ID	Final_MAX

Step 3: $\Pi_{tel}(R(ID, Final\_MAX) \bowtie Maskdata)$

Query 3: Given one location (longitude, latitude), find the nearest clinic ID that has face mask

	field1	field2	field4	fi
	過濾	過濾	過濾	過濾
1	2312010014	新竹市東區衛生所	03)5236158	0
2	2312041028	新竹市北區衛生所	03)5353969	600
3	2312050018	新竹市香山衛生所	03)5388109	600
4	5912010046	中美藥局	03)5283828	597
5	5912010206	康寧藥局	03)5321797	0
6	5912010215	李藥局	03)5779177	600
7	5912010313	金松大藥局	03)5719019	0
8	5912010322	南門藥局	03)5222665	0
9	5912010331	大家藥局	035)234035	600
10	5912010484	聖心藥局	03)5610697	597
11	5912010724	倫洋藥局	03)5749482	0
12	5912010779	誠真藥局	03)5278233	4

新竹市
新竹市東區民族路４０之２號
新竹市北區國華街六十九號一樓
新竹市香山區牛埔里育德街１８８號２樓
新竹市東區南大里西大路一ｏ二巷三號一樓
新竹市東區親仁里東大路１段１７９號
新竹市東區新莊里長春街３７號
新竹市東區立功里光復路二段３１０號
新竹市東區關帝里南門街８２號
新竹市東區南門里武昌街６號１樓
新竹市東區南大里食品路２６９號
新竹市東區建中路３８號１樓

Geocoding API

Google
Openstreet

ID	Name	(X, Y)	Tel
2312010014	新竹市東區衛生所	(122.1, 34.123)	
2312041028	新竹市北區衛生所	(122.1, 35.12131)	

Spatial Query in DB will rescue you !

MySQL 5.7 Reference Manual / Functions and Operators / Spatial Analysis Functions

12.16 Spatial Analysis Functions

Summary

- The relational model has rigorously defined query languages that are simple and powerful.
- **Relational algebra** is operational; useful as an internal representation for query evaluation steps.
- Typically there are several ways of expressing a given query; a query optimizer should choose an efficient version.
- Reading assignment: Chapter X
-