

Introduction to Database Systems CSE 414

Lecture 4: SQL Joins and Aggregates

CSE 414 - Autumn 2018

1

Announcements

- Homework 2 out now
 - git pull upstream master to get the starter code
 - Due Tuesday Oct. 9 at midnight
- Web quiz 1 due Friday midnight
- Section tomorrow important for HW2

CSE 414 - Autumn 2018

2

Loading Data into SQLite

```
>sqlite3 lecture04.db
sqlite> create table Purchase
(pid int primary key,
 product text,
 price float,
 quantity int,
 month varchar(15));
sqlite> -- download data.txt
sqlite> .mode list
sqlite> .import lec04-data.txt Purchase
```

Specify a filename where the database will be stored

Other DBMSs have other ways of importing data

MUST BE IN SAME MODE AS FILE TYPE
If the data is separated by commas, need to set
sqlite> .mode csv

3

Product(pname, price, category, manufacturer)
Company(cname, country)

Joins in SQL

pname	price	category	manufacturer	cname	country
MultiTouch	199.99	gadget	Canon	GizmoWorks	USA
SingleTouch	49.99	photography	Canon	Canon	Japan
Gizom	50	gadget	GizmoWorks	Hitachi	Japan
SuperGizmo	250.00	gadget	GizmoWorks		

Retrieve all Japanese products that cost < \$150

```
SELECT P.pname, P.price
FROM Product as P, Company as C
WHERE P.manufacturer=C.cname AND
      C.country='Japan' AND C.price < 150
```

Join Predicate

Selection predicates

CSE 414 - Autumn 2018

Product(pname, price, category, manufacturer)
Company(cname, country)

Joins in SQL

pname	price	category	manufacturer	cname	country
MultiTouch	199.99	gadget	Canon	GizmoWorks	USA
SingleTouch	49.99	photography	Canon	Canon	Japan
Gizom	50	gadget	GizmoWorks	Hitachi	Japan
SuperGizmo	250.00	gadget	GizmoWorks		

Retrieve all USA companies that manufacture "gadget" products

CSE 414 - Autumn 2018

5

Product(pname, price, category, manufacturer)
Company(cname, country)

Joins in SQL

pname	price	category	manufacturer	cname	country
MultiTouch	199.99	gadget	Canon	GizmoWorks	USA
SingleTouch	49.99	photography	Canon	Canon	Japan
Gizom	50	gadget	GizmoWorks	Hitachi	Japan
SuperGizmo	250.00	gadget	GizmoWorks		

Retrieve all USA companies that manufacture "gadget" products

```
SELECT DISTINCT C.cname
FROM Product as P, Company as C
WHERE C.country='USA' AND P.category = 'gadget'
      AND P.manufacturer = cname
```

Why DISTINCT?

(Inner) joins

```
Product(pname, price, category, manufacturer)
Company(cname, country)
-- manufacturer is foreign key to Company
```

```
SELECT DISTINCT cname
FROM Product, Company
WHERE country='USA' AND category = 'gadget'
AND manufacturer = cname
```

CSE 414 - Autumn 2018

7

(Inner) joins

```
SELECT DISTINCT cname
FROM Product, Company
WHERE country='USA' AND category = 'gadget'
AND manufacturer = cname
```

Product

pname	category	manufacturer
Gizmo	gadget	GizmoWorks
Camera	Photo	Hitachi
OneClick	Photo	Hitachi

Company

cname	country
GizmoWorks	USA
Canon	Japan
Hitachi	Japan

CSE 414 - Autumn 2018

8

(Inner) joins

```
SELECT DISTINCT cname
FROM Product, Company
WHERE country='USA' AND category = 'gadget'
AND manufacturer = cname
```

Product

pname	category	manufacturer
Gizmo	gadget	GizmoWorks
Camera	Photo	Hitachi
OneClick	Photo	Hitachi

Company

cname	country
GizmoWorks	USA
Canon	Japan
Hitachi	Japan

CSE 414 - Autumn 2018

9

(Inner) joins

```
SELECT DISTINCT cname
FROM Product, Company
WHERE country='USA' AND category = 'gadget'
AND manufacturer = cname
```

Product

pname	category	manufacturer
Gizmo	gadget	GizmoWorks
Camera	Photo	Hitachi
OneClick	Photo	Hitachi

Company

cname	country
GizmoWorks	USA
Canon	Japan
Hitachi	Japan

CSE 414 - Autumn 2018

10

(Inner) joins

```
SELECT DISTINCT cname
FROM Product, Company
WHERE country='USA' AND category = 'gadget'
AND manufacturer = cname
```

Product

pname	category	manufacturer
Gizmo	gadget	GizmoWorks
Camera	Photo	Hitachi
OneClick	Photo	Hitachi

Company

cname	country
GizmoWorks	USA
Canon	Japan
Hitachi	Japan

pname	category	manufacturer	cname	country
Gizmo	gadget	GizmoWorks	GizmoWorks	USA

11

(Inner) joins

```
SELECT DISTINCT cname
FROM Product, Company
WHERE country='USA' AND category = 'gadget'
AND manufacturer = cname
```

Product

pname	category	manufacturer
Gizmo	gadget	GizmoWorks
Camera	Photo	Hitachi
OneClick	Photo	Hitachi

Company

cname	country
GizmoWorks	USA
Canon	Japan
Hitachi	Japan

CSE 414 - Autumn 2018

12

(Inner) joins

```
SELECT DISTINCT cname
FROM Product, Company
WHERE country='USA' AND category = 'gadget'
AND manufacturer = cname
```

Product

pname	category	manufacturer
Gizmo	gadget	GizmoWorks
Camera	Photo	Hitachi
OneClick	Photo	Hitachi

Company

cname	country
GizmoWorks	USA
Canon	Japan
Hitachi	Japan

CSE 414 - Autumn 2018

13

(Inner) joins

```
SELECT DISTINCT cname
FROM Product, Company
WHERE country='USA' AND category = 'gadget'
AND manufacturer = cname
```

```
SELECT DISTINCT cname
FROM Product JOIN Company ON
country = 'USA' AND category = 'gadget'
AND manufacturer = cname
```

CSE 414 - Autumn 2018

14

(Inner) Joins

```
SELECT x1.a1, x2.a2, ... xm.am
FROM R1 as x1, R2 as x2, ... Rm as xm
WHERE Cond
```

for x1 in R1:

for x2 in R2:

...

for xm in Rm:

if Cond(x1, x2...):

output(x1.a1, x2.a2, ... xm.am)

This is called nested loop semantics since we are interpreting what a join means using a nested loop

15

Another example

```
Product(pname, price, category, manufacturer)
Company(cname, country)
-- manufacturer is foreign key to Company
```

Find US companies that manufacture both 'gadgets' and 'photo' products

CSE 414 - Autumn 2018

16

Another example

```
Product(pname, price, category, manufacturer)
Company(cname, country)
-- manufacturer is foreign key to Company
```

Find US companies that manufacture both 'gadgets' and 'photo' products

```
SELECT DISTINCT z.cname
FROM Product x, Company z
WHERE z.country = 'USA'
AND x.manufacturer = z.cname
AND x.category = 'gadget'
AND x.category = 'photography';
```

Does this work?

CSE 414 - Autumn 2018

17

Another example

```
Product(pname, price, category, manufacturer)
Company(cname, country)
-- manufacturer is foreign key to Company
```

Find US companies that manufacture both 'gadgets' and 'photo' products

```
SELECT DISTINCT z.cname
FROM Product x, Company z
WHERE z.country = 'USA'
AND x.manufacturer = z.cname
AND (x.category = 'gadget'
OR x.category = 'photography');
```

What about this?

CSE 414 - Autumn 2018

18

Another example

```
Product(pname, price, category, manufacturer)
Company(cname, country)
-- manufacturer is foreign key to Company
```

Find US companies that manufacture both 'gadgets' and 'photo' products

```
SELECT DISTINCT z.cname
FROM Product x, Product y, Company z
WHERE z.country = 'USA'
AND x.manufacturer = z.cname
AND y.manufacturer = z.cname
AND x.category = 'gadget'
AND y.category = 'photography';
```

Need to include Product twice!

19

Self-Joins and Tuple Variables

Find US companies that manufacture both 'gadgets' and 'photo' products

- Joining Product with Company is insufficient; need to join Product, with Product, and with Company
- When a relation occurs twice in the FROM clause we call it a self-join; in that case we must use tuple variables (aka table aliases) (why?)

CSE 414 - Autumn 2018

20

Self-joins

```
SELECT DISTINCT z.cname
FROM Product x, Product y, Company z
WHERE z.country = 'USA'
AND x.category = 'gadget'
AND y.category = 'photo'
AND x.manufacturer = z.cname
AND y.manufacturer = z.cname;
```

Product

pname	category	manufacturer
Gizmo	gadget	GizmoWorks
SingleTouch	photo	Hitachi
MultiTouch	Photo	GizmoWorks

Company

cname	country
GizmoWorks	USA
Hitachi	Japan

CSE 414 - Autumn 2018

21

Self-joins

```
SELECT DISTINCT z.cname
FROM Product x, Product y, Company z
WHERE z.country = 'USA'
AND x.category = 'gadget'
AND y.category = 'photo'
AND x.manufacturer = z.cname
AND y.manufacturer = z.cname;
```

Product

pname	category	manufacturer
Gizmo	gadget	GizmoWorks
SingleTouch	photo	Hitachi
MultiTouch	Photo	GizmoWorks

Company

cname	country
GizmoWorks	USA
Hitachi	Japan

CSE 414 - Autumn 2018

22

Self-joins

```
SELECT DISTINCT z.cname
FROM Product x, Product y, Company z
WHERE z.country = 'USA'
AND x.category = 'gadget'
AND y.category = 'photo'
AND x.manufacturer = z.cname
AND y.manufacturer = z.cname;
```

Product

pname	category	manufacturer
Gizmo	gadget	GizmoWorks
SingleTouch	photo	Hitachi
MultiTouch	Photo	GizmoWorks

Company

cname	country
GizmoWorks	USA
Hitachi	Japan

CSE 414 - Autumn 2018

23

Self-joins

```
SELECT DISTINCT z.cname
FROM Product x, Product y, Company z
WHERE z.country = 'USA'
AND x.category = 'gadget'
AND y.category = 'photo'
AND x.manufacturer = z.cname
AND y.manufacturer = z.cname;
```

Product

pname	category	manufacturer
Gizmo	gadget	GizmoWorks
SingleTouch	photo	Hitachi
MultiTouch	Photo	GizmoWorks

Company

cname	country
GizmoWorks	USA
Hitachi	Japan

CSE 414 - Autumn 2018

24

Self-joins

```

SELECT DISTINCT z.cname
FROM Product x, Product y, Company z
WHERE z.country = 'USA'
AND x.category = 'gadget'
AND y.category = 'photo'
AND x.manufacturer = z.cname
AND y.manufacturer = z.cname;

```

Product			
x	pname	category	manufacturer
y	Gizmo	gadget	GizmoWorks
	SingleTouch	photo	Hitachi
	MultiTouch	Photo	GizmoWorks

Company		z
cname	country	
GizmoWorks	USA	
Hitachi	Japan	

CSE 414 - Autumn 2018 25

Self-joins

```

SELECT DISTINCT z.cname
FROM Product x, Product y, Company z
WHERE z.country = 'USA'
AND x.category = 'gadget'
AND y.category = 'photo'
AND x.manufacturer = z.cname
AND y.manufacturer = z.cname;

```

Product			
x	pname	category	manufacturer
y	Gizmo	gadget	GizmoWorks
	SingleTouch	photo	Hitachi
	MultiTouch	Photo	GizmoWorks

Company		z
cname	country	
GizmoWorks	USA	
Hitachi	Japan	

CSE 414 - Autumn 2018 26

Self-joins

```

SELECT DISTINCT z.cname
FROM Product x, Product y, Company z
WHERE z.country = 'USA'
AND x.category = 'gadget'
AND y.category = 'photo'
AND x.manufacturer = z.cname
AND y.manufacturer = z.cname;

```

Product			
x	pname	category	manufacturer
y	Gizmo	gadget	GizmoWorks
	SingleTouch	photo	Hitachi
	MultiTouch	Photo	GizmoWorks

Company		z
cname	country	
GizmoWorks	USA	
Hitachi	Japan	

CSE 414 - Autumn 2018 27

Self-joins

```

SELECT DISTINCT z.cname
FROM Product x, Product y, Company z
WHERE z.country = 'USA'
AND x.category = 'gadget'
AND y.category = 'photo'
AND x.manufacturer = z.cname
AND y.manufacturer = z.cname;

```

Product			
x	pname	category	manufacturer
y	Gizmo	gadget	GizmoWorks
	SingleTouch	photo	Hitachi
	MultiTouch	Photo	GizmoWorks

Company		z
cname	country	
GizmoWorks	USA	
Hitachi	Japan	

CSE 414 - Autumn 2018 28

Self-joins

```

SELECT DISTINCT z.cname
FROM Product x, Product y, Company z
WHERE z.country = 'USA'
AND x.category = 'gadget'
AND y.category = 'photo'
AND x.manufacturer = z.cname
AND y.manufacturer = z.cname;

```

Product			
x	pname	category	manufacturer
y	Gizmo	gadget	GizmoWorks
	SingleTouch	photo	Hitachi
	MultiTouch	Photo	GizmoWorks

Company		z
cname	country	
GizmoWorks	USA	
Hitachi	Japan	

x.pname	x.category	x.manufacturer	y.pname	y.category	y.manufacturer	z.cname	z.country
Gizmo	gadget	GizmoWorks	MultiTouch	Photo	GizmoWorks	GizmoWorks	²⁹ USA

CSE 414 - Autumn 2018 29

Self-joins

```

SELECT DISTINCT z.cname
FROM Product x, Product y, Company z
WHERE z.country = 'USA'
AND x.category = 'gadget'
AND y.category = 'photo'
AND x.manufacturer = z.cname
AND y.manufacturer = z.cname;

```

Product			
x	pname	category	manufacturer
y	Gizmo	gadget	GizmoWorks
	SingleTouch	photo	Hitachi
	MultiTouch	Photo	GizmoWorks

Company		z
cname	country	
GizmoWorks	USA	
Hitachi	Japan	

x.pname	x.category	x.manufacturer	y.pname	y.category	y.manufacturer	z.cname	z.country
Gizmo	gadget	GizmoWorks	MultiTouch	Photo	GizmoWorks	GizmoWorks	³⁰ USA

CSE 414 - Autumn 2018 30

Joins in SQL

- The join we have just seen is sometimes called an **inner join**
 - Each row in the result **must come from both tables in the join**
- Sometimes we want to include rows from only one of the two table: **outer join**

CSE 414 - Autumn 2018

32

Employee(id, name)
Sales(employeeID, productID)

Inner Join

Employee		Sales	
id	name	employeeID	productID
1	Joe	1	344
2	Jack	1	355
3	Jill	2	544

Retrieve employees and their sales

CSE 414 - Autumn 2018

33

Employee(id, name)
Sales(employeeID, productID)

Inner Join

Employee		Sales	
id	name	employeeID	productID
1	Joe	1	344
2	Jack	1	355
3	Jill	2	544

Retrieve employees and their sales

```
SELECT *
FROM Employee E, Sales S
WHERE E.id = S.employeeID
```

CSE 414 - Autumn 2018

34

Employee(id, name)
Sales(employeeID, productID)

Inner Join

Employee		Sales	
id	name	employeeID	productID
1	Joe	1	344
2	Jack	1	355
3	Jill	2	544

Retrieve employees and their sales

```
SELECT *
FROM Employee E, Sales S
WHERE E.id = S.employeeID
```

id	name	employeeID	productID
1	Joe	1	344
1	Joe	1	355
2	Jack	2	544

CSE 414 - Autumn 2018

35

Employee(id, name)
Sales(employeeID, productID)

Inner Join

Employee		Sales	
id	name	employeeID	productID
1	Joe	1	344
2	Jack	1	355
3	Jill	2	544

Retrieve employees and their sales

```
SELECT *
FROM Employee E, Sales S
WHERE E.id = S.employeeID
```

id	name	employeeID	productID
1	Joe	1	344
1	Joe	1	355
2	Jack	2	544

CSE 414 - Autumn 2018

36

Jill is missing

Employee(id, name)
Sales(employeeID, productID)

Inner Join

Employee		Sales	
id	name	employeeID	productID
1	Joe	1	344
2	Jack	1	355
3	Jill	2	544

Retrieve employees and their sales

```
SELECT *
FROM Employee E
INNER JOIN
Sales S
ON E.id = S.employeeID
```

id	name	employeeID	productID
1	Joe	1	344
1	Joe	1	355
2	Jack	2	544

CSE 414 - Autumn 2018

37

Alternative syntax

Jill is missing

Employee(id, name)
Sales(employeeID, productID)

Outer Join

Employee	
id	name
1	Joe
2	Jack
3	Jill

Sales	
employeeID	productID
1	344
1	355
2	544

Retrieve employees and their sales

```
SELECT *
FROM Employee E
LEFT OUTER JOIN
Sales S
ON E.id = S.employeeID
```

id	name	employeeID	productID
1	Joe	1	344
1	Joe	1	355
2	Jack	2	544
3	Jill	NULL	NULL

Jill is present

Outer joins

Product(name, category)
Purchase(prodName, store)

-- prodName is foreign key

```
SELECT Product.name, Purchase.store
FROM Product LEFT OUTER JOIN Purchase ON
Product.name = Purchase.prodName
```

CSE 414 - Autumn 2018 39

```
SELECT Product.name, Purchase.store
FROM Product JOIN Purchase ON
Product.name = Purchase.prodName
```

Product	
Name	Category
Gizmo	gadget
Camera	Photo
OneClick	Photo

Purchase	
ProdName	Store
Gizmo	Wiz
Camera	Ritz
Camera	Wiz

CSE 414 - Autumn 2018 40

```
SELECT Product.name, Purchase.store
FROM Product JOIN Purchase ON
Product.name = Purchase.prodName
```

Name	Category
Gizmo	gadget
Camera	Photo
OneClick	Photo

ProdName	Store
Gizmo	Wiz
Camera	Ritz
Camera	Wiz

CSE 414 - Autumn 2018 41

```
SELECT Product.name, Purchase.store
FROM Product JOIN Purchase ON
Product.name = Purchase.prodName
```

Name	Category
Gizmo	gadget
Camera	Photo
OneClick	Photo

ProdName	Store
Gizmo	Wiz
Camera	Ritz
Camera	Wiz

Name	Store
Gizmo	Wiz

Output

CSE 414 - Autumn 2018 42

```
SELECT Product.name, Purchase.store
FROM Product JOIN Purchase ON
Product.name = Purchase.prodName
```

Name	Category
Gizmo	gadget
Camera	Photo
OneClick	Photo

ProdName	Store
Gizmo	Wiz
Camera	Ritz
Camera	Wiz

Name	Store
Gizmo	Wiz

Output

CSE 414 - Autumn 2018 43

```

SELECT Product.name, Purchase.store
FROM Product JOIN Purchase ON
Product.name = Purchase.prodName

```

Product

Name	Category
Gizmo	gadget
Camera	Photo
OneClick	Photo

Purchase

ProdName	Store
Gizmo	Wiz
Camera	Ritz
Camera	Wiz

Output

Name	Store
Gizmo	Wiz

CSE 414 - Autumn 2018 44

```

SELECT Product.name, Purchase.store
FROM Product JOIN Purchase ON
Product.name = Purchase.prodName

```

Product

Name	Category
Gizmo	gadget
Camera	Photo
OneClick	Photo

Purchase

ProdName	Store
Gizmo	Wiz
Camera	Ritz
Camera	Wiz

Output

Name	Store
Gizmo	Wiz

CSE 414 - Autumn 2018 45

```

SELECT Product.name, Purchase.store
FROM Product JOIN Purchase ON
Product.name = Purchase.prodName

```

Product

Name	Category
Gizmo	gadget
Camera	Photo
OneClick	Photo

Purchase

ProdName	Store
Gizmo	Wiz
Camera	Ritz
Camera	Wiz

Output

Name	Store
Gizmo	Wiz
Camera	Ritz

CSE 414 - Autumn 2018 46

```

SELECT Product.name, Purchase.store
FROM Product JOIN Purchase ON
Product.name = Purchase.prodName

```

Product

Name	Category
Gizmo	gadget
Camera	Photo
OneClick	Photo

Purchase

ProdName	Store
Gizmo	Wiz
Camera	Ritz
Camera	Wiz

Output

Name	Store
Gizmo	Wiz
Camera	Ritz
Camera	Wiz

CSE 414 - Autumn 2018 47

```

SELECT Product.name, Purchase.store
FROM Product JOIN Purchase ON
Product.name = Purchase.prodName

```

Product

Name	Category
Gizmo	gadget
Camera	Photo
OneClick	Photo

Purchase

ProdName	Store
Gizmo	Wiz
Camera	Ritz
Camera	Wiz

Output

Name	Store
Gizmo	Wiz
Camera	Ritz
Camera	Wiz

CSE 414 - Autumn 2018 48

```

SELECT Product.name, Purchase.store
FROM Product LEFT OUTER JOIN Purchase ON
Product.name = Purchase.prodName

```

Product

Name	Category
Gizmo	gadget
Camera	Photo
OneClick	Photo

Purchase

ProdName	Store
Gizmo	Wiz
Camera	Ritz
Camera	Wiz

Output

Name	Store
Gizmo	Wiz
Camera	Ritz
Camera	Wiz

CSE 414 - Autumn 2018 49


```
SELECT Product.name, Purchase.store
FROM Product LEFT OUTER JOIN Purchase ON
Product.name = Purchase.prodName
```

Product		Purchase	
Name	Category	ProdName	Store
Gizmo	gadget	Gizmo	Wiz
Camera	Photo	Camera	Ritz
OneClick	Photo	Camera	Wiz

Output	
Name	Store
Gizmo	Wiz
Camera	Ritz
Camera	Wiz
OneClick	NULL

CSE 414 - Autumn 2018 50

```
SELECT Product.name, Purchase.store
FROM Product FULL OUTER JOIN Purchase ON
Product.name = Purchase.prodName
```

Product		Purchase	
Name	Category	ProdName	Store
Gizmo	gadget	Gizmo	Wiz
Camera	Photo	Camera	Ritz
OneClick	Photo	Camera	Wiz
		Phone	Foo

Output	
Name	Store
Gizmo	Wiz
Camera	Ritz
Camera	Wiz
OneClick	NULL
NULL	Foo

CSE 414 - Autumn 2018 51

Outer Joins

```
tableA (LEFT/RIGHT/FULL) OUTER JOIN tableB ON p
```

- Left outer join:
 - Include tuples from tableA even if no match
- Right outer join:
 - Include tuples from tableB even if no match
- Full outer join:
 - Include tuples from both even if no match
- In all cases:
 - Patch tuples without matches using NULL

CSE 414 - Autumn 2018 52

Aggregates in SQL

CSE 414 - Autumn 2018 54

Simple Aggregations

Five basic aggregate operations in SQL

```
select count(*) from Purchase
select sum(quantity) from Purchase
select avg(price) from Purchase
select max(quantity) from Purchase
select min(quantity) from Purchase
```

Except count, all aggregations apply to a single attribute

CSE 414 - Autumn 2018 55

Aggregates and NULL Values

Null values are not used in aggregates

```
insert into Purchase
values(12, 'gadget', NULL, NULL, 'april')
```

Try the following

```
select count(*) from Purchase
select count(quantity) from Purchase

select sum(quantity) from Purchase

select count(*)
from Purchase
where quantity is not null;
```

CSE 414 - Autumn 2018 56

Counting Duplicates

COUNT applies to duplicates, unless otherwise stated:

```
SELECT count(product)
FROM Purchase
WHERE price > 4.99
```

same as count(*) if no nulls

We probably want:

```
SELECT count(DISTINCT product)
FROM Purchase
WHERE price > 4.99
```

CSE 414 - Autumn 2018

57

More Examples

```
SELECT Sum(P.price * P.quantity)
FROM Purchase as P
```

```
SELECT Sum(P.price * P.quantity)
FROM Purchase as P
WHERE P.product = 'bagel'
```

What do they mean?

CSE 414 - Autumn 2018

58