SQL offers several constructs beyond relational algebra to allow users to write more powerful queries. In this lecture, we will study a collection of constructs designed for statistical analysis.

Syntax of an Aggregate Query

select $A_1, ..., A_t, agg_1(B_1), ..., agg_m(B_m)$
from $T_1, ..., T_n$
where P
group by $C_1, ..., C_g$
having H

where

- $T_1, ..., T_n$ are tables.
- $A_1, ..., A_t, B_1, ..., B_m, C_1, ..., C_g$ are attributes.
- $B_1, ..., B_m$ are called aggregate attributes.
- $C_1, ..., C_g$ are called group-by attributes.
- Each of $A_1, ..., A_t$ must be a group-by attribute (i.e., each A_i is identical to some C_j where $j \in [1, t]$).
- P is a tuple predicate, and H is an group predicate.
- $agg_1, ..., agg_m$ are aggregate functions.

Aggregate Function

$agg(A)$

- agg = count: return the number of the values in A.
- agg = sum: return the sum of the values in A.
- agg = min: return the minimum value in A.
- agg = max: return the maximum value in A.
- agg = avg: return the average of the values in A.

Note

A must be numeric for sum, min, max, and avg.

Aggregate Function

$agg(\text{distinct } A)$

- agg = count: return the number of distinct values in A.
- agg = sum: return the sum of the distinct values in A.
- agg = avg: return the average of the distinct values in A.

Note

"Distinct" has no effect with min and max.

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	8000


```
select count(dept), count(distinct dept), sum(sal), sum(distinct sal)
from PROF
```

Let's try...

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	8000


```
select count(dept), count(distinct dept), sum(sal), sum(distinct sal)
from PROF
```

Result:

5	2	37000	29000

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	6000
$p5$	Emily	EE	asso	8000

select count(*), sum(sal), min(sal), max(sal), avg(sal)
from PROF

Result:

5	37000	5000	10000	7400

Note

count(A) is equivalent to count(*). This is intuitive: count(A) returns the same result no matter which column is used as A. Hence, A can be as well omitted. Note, however, * cannot be used with distinct, which must always be accompanied by a concrete attribute.

Group By

select $A_1, ..., A_t, agg_1(B_1), ..., agg_m(B_m)$
from T
group by $C_1, ..., C_g$

This statement carries out the following steps:

1. Divide T into groups where each group consists of the tuples that are identical on all $C_1, ..., C_g$.
2. Execute the select clause on each group.

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	8000


```
select max(sal) from PROF
group by dept
```

PROF is divided into two groups. The first one includes the tuples with pid = $p1, p3$, while the second one includes the rest. The query returns:

10000
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	8000


```
select dept, rank, count(*) from PROF
group by dept, rank
```

Let's try...

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	8000


```
select dept, rank, count(*) from PROF
group by dept, rank
```

Result:

dept	rank	
CS	asst	1
CS	full	1
EE	asso	2
EE	asst	1

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	8000

select pid, count(*) from PROF
group by dept, rank

Syntax error: Every attribute in the select clause (if not an aggregate attribute) must be a group-by attribute. See the syntax on Slide 3.

Past Muddiest Points - Let's dig deeper!

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	8000

select pid, count(*) from PROF
group by dept, rank

Sort & Grouping...

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

Past Muddiest Points - Let's dig deeper!

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	8000

select pid, count(*) from PROF
group by dept, rank

Sort & Grouping...

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

pid	count(*)
$p1$	1
$p3$	1
$p2$, $p5$	2
$p4$	1

Past Muddiest Points - Let's dig deeper!

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	8000

select pid, count(*) from PROF
group by dept, rank

Sort & Grouping...

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

pid	count(*)
$p1$	1
$p3$	1
$p2$, $p5$	2
$p4$	1

Oracle: Syntax Error! MySQL: return $p2$ or $p5$, depends on mood!

Having

select $A_1, ..., A_t, agg_1(B_1), ..., agg_n(B_m)$
from T
group by $C_1, ..., C_g$
having H

This statement carries out the following steps:

1. Divide T into groups where each group consists of the tuples that are identical on all $C_1, ..., C_g$.
2. Eliminate the groups that do not satisfy H (i.e. the group predicate).
3. Execute the select clause on each of the remaining groups.

Group Predicate

H is a set of aggregate comparisons connected by logic operators: AND, OR, and NOT, where an aggregate comparison has the form

$$agg(A)\ op\ v$$

where

- agg is an aggregate function.
- op can be $=, <>, <, <=, >=, >$.
- A is an attribute.

PROF

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

select rank, max(sal) from PROF
group by rank
having count(*) >= 2

Let's try...

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	8000


```
select rank, max(sal) from PROF
group by rank
having count(*) >= 2
```


Grouping...

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

PROF

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


```
select rank, max(sal) from PROF
group by rank
having count(*) >= 2
```


Result:

rank	
asst	6000
asso	8000

Note that the group of rank = full has been eliminated by the having clause.

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	8000


```
select rank, count(*) from PROF
group by rank
having count(*) >= 2 and max(sal) >= 7000
```


Result:

rank	
asso	2

Where

select $A_1, \ldots, A_t, agg_1(B_1), \ldots, agg_m(B_m)$
from T
where P
group by $C_1, \ldots, C_g$
having H

This statement carries out the following steps:

1. Perform a selection on T using P.
2. Execute group-by, having and select on the result of the selection.

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	8000


```
select dept, min(sal) from PROF
where sal >= 8000
group by dept
having count(*) >= 2
```


Let's try...

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	8000


```
select dept, min(sal) from PROF
where sal >= 8000
group by dept
having count(*) >= 2
```


Grouping...

pid	name	dept	rank	sal
$p3$	Calvin	CS	full	10000
$p2$	Bob	EE	asso	8000
$p5$	Emily	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	8000


```
select dept, min(sal) from PROF
where sal >= 8000
group by dept
having count(*) >= 2
```


Result:

dept	
EE	8000

The group dept = CS is eliminated because it has only 1 tuple.

Tuple Predicate P vs. Group Predicate H

- A comparison in P has the form $A\ op\ v$, while a comparison in H has the form $agg(A)\ op\ v$.
- P filters tuples before group by, while H filters groups after group by.

Everything Together

select $A_1, ..., A_t, agg_1(B_1), ..., agg_m(B_m)$
from $T_1, ..., T_n$
where P
group by $C_1, ..., C_g$
having H

This statement carries out the following steps:

1. Perform the cartesian product $T_1 \times ... \times T_n$.
2. Execute where, group-by, having and select on the cartesian product.

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	8000

TEACH

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


```
select PROF.pid, count(*)
from PROF, TEACH
where PROF.pid = TEACH.pid
group by PROF.pid
having count(*) >= 2
```

Let's try...

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	8000

TEACH

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


```
select PROF.pid, count(*)
from PROF, TEACH
where PROF.pid = TEACH.pid
group by PROF.pid
having count(*) >= 2
```


Grouping...

pid	name	dept	rank	sal	pid	cid	year
$p1$	Adam	CS	asst	6000	$p1$	$c1$	2011
$p1$	Adam	CS	asst	6000	$p1$	$c2$	2012
$p2$	Bob	EE	asso	8000	$p2$	$c2$	2012

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	8000

TEACH

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


```
select PROF.pid, count(*)
from PROF, TEACH
where PROF.pid = TEACH.pid
group by PROF.pid
having count(*) >= 2
```

Result:

pid	
$p1$	2

Funny stuffs

```
SQL> select count(*) from developer_brain;
select count(*) from developer_brain
                     *
ERROR at line 1:
ORA-00942: table or view does not exist
```


http://www.orafaq.com/wiki/Fun_stuff