Week 11

Relational Algebra and SQL
(Aggregation and Grouping Operation)

Pearson Education © 2009

Objectives

- Relational Algebra on aggregation and grouping operation.
- SQL on aggregation and grouping.

Pearson Education © 2009

Aggregate Operations

- $\mathfrak{I}_{AL}(R)$
  - Applies aggregate function list, AL, to R to define a relation over the aggregate list.
  - AL contains one or more (<aggregate_function>, <attribute>) pairs .
- Main aggregate functions are: COUNT, SUM, AVG, MIN, and MAX.

Pearson Education © 2009

Example

Staff

staffNo	fName	lName	position	sex	DOB	salary	branchNo
SL21	John	White	Manager	M	1-Oct-45	30000	B005
SG37	Ann	Beech	Assistant	F	10-Nov-60	12000	B003
SG14	David	Ford	Supervisor	M	24-Mar-58	18000	B003
SA9	Mary	Howe	Assistant	F	19-Feb-70	9000	B007
SG5	Susan	Brand	Manager	F	3-Jun-40	24000	B003
SL41	Julie	Lee	Assistant	F	13-Jun-65	9000	B005

Example – Aggregate Operations

- **How many staff earn more than £10,000?**

$$\rho_R(\text{myCount})\ \Im_{\text{COUNT staffNo}}\ (\sigma_{\text{salary} > 10000}\ (\text{Staff}))$$

myCount
4

Pearson Education © 2009

Example – Aggregate Operations

- **Find the average staff salary**

  $\rho_R(\text{myAverage})\ \Im_{\text{AVERAGE salary}}\ (\text{Staff})$

myAverage
17000

Grouping Operation

- $_{GA}\Im_{AL}(R)$
  - Groups tuples of R by grouping attributes, GA, and then applies aggregate function list, AL, to define a new relation.
  - AL contains one or more (<aggregate_function>, <attribute>) pairs.
  - Resulting relation contains the grouping attributes, GA, along with results of each of the aggregate functions.

Pearson Education © 2009

Example – Grouping Operation

- **Find the number of staff working in each branch and the sum of their salaries.**

$$\rho_R(\text{branchNo, myCount, mySum})$$

$$_{\text{branchNo}}\Im_{\text{COUNT staffNo, SUM salary}}(\text{Staff})$$

branchNo	myCount	mySum
B003	3	54000
B005	2	39000
B007	1	9000

Pearson Education © 2009

SQL: SELECT Statement - Aggregates

- **ISO standard defines five aggregate functions:**

COUNT returns number of values in specified column.

SUM returns sum of values in specified column.

AVG returns average of values in specified column.

MIN returns smallest value in specified column.

MAX returns largest value in specified column.

Pearson Education © 2009

SQL: SELECT Statement - Aggregates

- Each operates on a single column of a table and returns a single value.
- COUNT, MIN, and MAX apply to numeric and non-numeric fields, but SUM and AVG may be used on numeric fields only.
- Apart from COUNT(*), each function eliminates nulls first and operates only on remaining non-null values.

Pearson Education © 2009

SQL: SELECT Statement - Aggregates

- **COUNT(*) counts all rows of a table, regardless of whether nulls or duplicate values occur.**
- **Can use DISTINCT before column name to eliminate duplicates.**
- **DISTINCT has no effect with MIN/MAX, but may have with SUM/AVG.**

Pearson Education © 2009

SQL: SELECT Statement - Aggregates

- **Aggregate functions can be used only in SELECT list and in HAVING clause.**

- **If SELECT list includes an aggregate function and there is no GROUP BY clause, SELECT list cannot reference a column out with an aggregate function. For example, the following is illegal:**

```
SELECT staffNo, COUNT(salary)
FROM Staff;
```


Pearson Education © 2009

Example : Use of COUNT(*)

How many properties cost more than £350 per month to rent?

```
SELECT COUNT(*) AS myCount
FROM PropertyForRent
WHERE rent > 350;
```

myCount
5

Pearson Education © 2009

Example : Use of COUNT(DISTINCT)

How many different properties viewed in May ‘04?

```
SELECT COUNT(DISTINCT propertyNo) AS myCount
FROM Viewing
WHERE viewDate BETWEEN ‘1-May-04’
      AND ‘31-May-04’;
```

myCount
2

Pearson Education © 2009

Example: Use of COUNT and SUM

Find number of Managers and sum of their salaries.

```
SELECT COUNT(staffNo) AS myCount,
       SUM(salary) AS mySum
FROM Staff
WHERE position = 'Manager';
```

myCount	mySum
2	54000.00

Pearson Education © 2009

Example: Use of MIN, MAX, AVG

Find minimum, maximum, and average staff salary.

```
SELECT MIN(salary) AS myMin,
  MAX(salary) AS myMax,
   AVG(salary) AS myAvg
FROM Staff;
```

myMin	myMax	myAvg
9000.00	30000.00	17000.00

Pearson Education © 2009

SQL: SELECT Statement - Grouping

- **Use GROUP BY clause to get sub-totals.**
- **SELECT and GROUP BY closely integrated: each item in SELECT list must be *single-valued per group*, and SELECT clause may only contain:**
  - **column names**
  - **aggregate functions**
  - **constants**
  - **expression involving combinations of the above.**

Pearson Education © 2009

SQL: SELECT Statement - Grouping

- All column names in SELECT list must appear in GROUP BY clause unless name is used only in an aggregate function.
- If WHERE is used with GROUP BY, WHERE is applied first, then groups are formed from remaining rows satisfying predicate.
- ISO considers two nulls to be equal for purposes of GROUP BY.

Pearson Education © 2009

Example: Use of GROUP BY

Find number of staff in each branch and their total salaries.

```
SELECT    branchNo,
          COUNT(staffNo) AS myCount,
          SUM(salary) AS mySum
FROM Staff
GROUP BY branchNo
ORDER BY branchNo;
```


Pearson Education © 2009

Example : Use of GROUP BY

branchNo	myCount	mySum
B003	3	54000.00
B005	2	39000.00
B007	1	9000.00

Pearson Education © 2009

Restricted Groupings – HAVING clause

- HAVING clause is designed for use with GROUP BY to restrict groups that appear in final result table.
- Similar to WHERE, but WHERE filters individual rows whereas HAVING filters groups.
- Column names in HAVING clause must also appear in the GROUP BY list or be contained within an aggregate function.

Pearson Education © 2009

Example : Use of HAVING

For each branch with more than 1 member of staff, find number of staff in each branch and sum of their salaries.

```
SELECT branchNo,
       COUNT(staffNo) AS myCount,
       SUM(salary) AS mySum
FROM Staff
GROUP BY branchNo
HAVING COUNT(staffNo) > 1
ORDER BY branchNo;
```


Pearson Education © 2009

Example: Use of HAVING

branchNo	myCount	mySum
B003	3	54000.00
B005	2	39000.00

Pearson Education © 2009