



## SQL language: basics

The **SELECT** statement:  
Aggregate Functions, **GROUP BY**

# The SELECT statement: basics

- Aggregate functions
- GROUP BY operator
- Having Clause





# The SELECT statement: basics

## Aggregate functions

# Aggregate functions

- An aggregate function
  - operates on a set of values
  - produces a single (aggregate) value as a result

# Aggregate functions

- Aggregate functions available in SQL-2
  - COUNT: count of elements in a given attribute
  - SUM: sum of values for a given attribute
  - AVG: average of values for a given attribute
  - MAX: maximum value of a given attribute
  - MIN: minimum value of a given attribute

# Aggregate functions

- An aggregate function
  - operates on a set of values
  - produces a single (aggregate) value as a result
  - is specified in the `SELECT` clause

# Structure of the SELECT statement

```
SELECT ListOfAggregateFunctionsToDisplay  
FROM ListOfTablesToUse  
[WHERE TupleConditions ]  
[ORDER BY ListOfOrderingAttributes];
```



# Aggregate functions

## ➤ An aggregate function

- operates on a set of values
- produces a single (aggregate) value as a result
- is specified in the **SELECT** clause
  - non-aggregate attributes may not be specified at the same time
  - multiple aggregate functions may be specified simultaneously



# The COUNT function

- Counts the number of elements in a set
  - rows in a table
  - (possibly distinct) values for one or more attributes

COUNT (<\* | [DISTINCT | ALL] *ListOfAttributes* >)

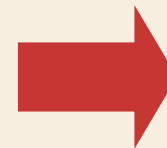
# The COUNT function (no.1)

➤ Find the number of suppliers

```
SELECT COUNT(*)  
FROM S;
```

S

<u>SId</u>	SName	#Employees	City
S1	Smith	20	London
S2	Jones	10	Paris
S3	Blake	30	Paris
S4	Clark	20	London
S5	Adams	30	Athens



R

5

## The COUNT function (no.2)


➤ Find the number of suppliers that supply at least one product

SP

<u>SId</u>	<u>PId</u>	Qty
S1	P1	300
S1	P2	200
S1	P3	400
S1	P4	200
S1	P5	100
S1	P6	100
S2	P1	300
S2	P2	400
S3	P2	200
S4	P3	200
S4	P4	300
S4	P5	400

```
SELECT COUNT(*)  
FROM SP;
```

R



12

➤ It counts the number of supplied products, not the suppliers

## The COUNT function (no.2)

➤ Find the number of suppliers that supply at least one product

SP

<u>SId</u>	<u>PId</u>	Qty
S1	P1	300
S1	P2	200
S1	P3	400
S1	P4	200
S1	P5	100
S1	P6	100
S2	P1	300
S2	P2	400
S3	P2	200
S4	P3	200
S4	P4	300
S4	P5	400

```
SELECT COUNT(SId)  
FROM SP;
```



R

12

➤ It still counts the number of supplied products, not the suppliers



## The COUNT function (no.2)

➤ Find the number of suppliers that supply at least one product

SP

<u>SId</u>	<u>PId</u>	Qty
S1	P1	300
S1	P2	200
S1	P3	400
S1	P4	200
S1	P5	100
S1	P6	100
S2	P1	300
S2	P2	400
S3	P2	200
S4	P3	200
S4	P4	300
S4	P5	400

```
SELECT COUNT(DISTINCT SId)  
FROM SP;
```



R

4

➤ It counts the number of distinct suppliers

# The COUNT function

- Counts the number of elements in a set
  - rows in a table
  - (possibly distinct) values for one or more attributes

COUNT (<\* | [DISTINCT | ALL] *ListOfAttributes* >)

- If the function argument is preceded by **DISTINCT**, it counts the number of distinct values of the argument


# Aggregate functions and WHERE

➤ Find the number of suppliers providing product P2


SP

<u>SId</u>	<u>PId</u>	Qty
S1	P1	300
S1	P2	200
S1	P3	400
S1	P4	200
S1	P5	100
S1	P6	100
S2	P1	300
S2	P2	400
S3	P2	200
S4	P3	200
S4	P4	300
S4	P5	400

```
SELECT COUNT(*)  
FROM SP  
WHERE PId='P2';
```



<u>SId</u>	<u>PId</u>	Qty
S1	P2	200
S2	P2	400
S3	P2	200



R

3
---

# Aggregate functions and WHERE

- Aggregate functions are only evaluated once all predicates in the **WHERE** clause have been applied



# The SUM, MAX, MIN, AVG functions

## ➤ SUM, MAX, MIN and AVG

- they allow an attribute or an expression as argument

## ➤ SUM and AVG

- they only allow numeric type or time interval attributes

## ➤ MAX and MIN

- they require an expression that can be ordered
  - may also be applied to character strings and time instants

# The SUM function

➤ Find the overall quantity of supplied pieces for product P2

SP

<u>SId</u>	<u>PId</u>	Qty
S1	P1	300
S1	P2	200
S1	P3	400
S1	P4	200
S1	P5	100
S1	P6	100
S2	P1	300
S2	P2	400
S3	P2	200
S4	P3	200
S4	P4	300
S4	P5	400

```
SELECT SUM(Qty)
FROM SP
WHERE PId='P2';
```

<u>SId</u>	<u>PId</u>	Qty
S1	P2	200
S2	P2	400
S3	P2	200

R

800



## The SELECT statement: basics

The GROUP BY operator

# Grouping

➤ *For each product*, find the overall quantity of supplied pieces

SP

<u>SId</u>	<u>PId</u>	Qty
S1	P1	300
S1	P2	200
S1	P3	400
S1	P4	200
S1	P5	100
S1	P6	100
S2	P1	300
S2	P2	400
S3	P2	200
S4	P3	200
S4	P4	300
S4	P5	400

SP

<u>SId</u>	<u>PId</u>	Qty
S1	P1	300
S2	P1	300
S1	P2	200
S2	P2	400
S3	P2	200
S1	P3	400
S4	P3	200
S1	P4	200
S4	P4	300
S1	P5	100
S4	P5	400
S1	P6	100

R

<u>PId</u>	
P1	600
P2	800
P3	600
P4	500
P5	500
P6	100



# Grouping

➤ *For each product*, find the overall quantity of supplied pieces

```
SELECT PId, SUM(Qty)
FROM SP
GROUP BY PId;
```

# GROUP BY

➤ Grouping clause

**GROUP BY** *ListOfGroupingAttributes*

- the order of grouping attributes is irrelevant

➤ Only

- attributes specified in the GROUP BY clause
- aggregate functions

are allowed to appear in the **SELECT** statement

# GROUP BY and WHERE

- For each product, find the overall quantity of pieces supplied by suppliers based in Paris

S

<u>SId</u>	SName	#Employees	City
S1	Smith	20	London
S2	Jones	10	Paris
S3	Blake	30	Paris
S4	Clark	20	London
S5	Adams	30	Athens

SP

<u>SId</u>	<u>PId</u>	Qty
S1	P1	300
S1	P2	200
S1	P3	400
S1	P4	200
S1	P5	100
S1	P6	100
S2	P1	300
S2	P2	400
S3	P2	200
S4	P3	200
S4	P4	300
S4	P5	400

## GROUP BY and WHERE

- For each product, find the overall quantity of pieces supplied by suppliers based in Paris

```
SELECT ...  
FROM SP, S  
WHERE SP.SId=S.SId AND City='Paris'  
...
```



# GROUP BY and WHERE

- For each product, find the overall quantity of pieces supplied by suppliers based in Paris

S.SId	S.SName	S.#Empl	S.City	SP.SId	SP.PId	SP.Qty
S1	Smith	20	London	S1	P1	300
S1	Smith	20	London	S1	P2	200
S1	Smith	20	London	S1	P3	400
S1	Smith	20	London	S1	P4	200
S1	Smith	20	London	S1	P5	100
S1	Smith	20	London	S1	P6	100
S2	Jones	10	Paris	S2	P1	300
S2	Jones	10	Paris	S2	P2	400
S3	Blake	30	Paris	S3	P2	200
S4	Clark	20	London	S4	P3	200
S4	Clark	20	London	S4	P4	300
S4	Clark	20	London	S4	P5	400

## GROUP BY and WHERE

- For each product, find the overall quantity of pieces supplied by suppliers based in Paris

```
SELECT PId, SUM(Qty)
FROM SP, S
WHERE SP.SId=S.SId AND City='Paris'
GROUP BY PId;
```

- Products that are not supplied by any supplier are not included in the result

# GROUP BY and WHERE

- For each product, find the overall quantity of pieces supplied by suppliers based in Paris

SP.PId	SP.Qty
P1	300
P2	400
P2	200



R

SP.PId	
P1	300
P2	600

# GROUP BY and SELECT

- For each product, find the code, the name and the overall supplied quantity

```
SELECT P.PId, PName, SUM(Qty)
FROM P, SP
WHERE P.PId=SP.PId
GROUP BY P.PId, PName
```

- Syntactic device

- attributes that are unambiguously determined by other attributes already present in the GROUP BY clause may be added *without altering the result*

# Structure of the SELECT statement

```
SELECT [DISTINCT] ListOfAttributesToDisplay  
FROM ListOfTablesToUse  
[WHERE TupleConditions ]  
[GROUP BY ListOfGroupingAttributes ]  
[ORDER BY ListOfOrderingAttributes ];
```



## Group selection condition

- Find the overall quantity of supplied pieces for the products for which at least 600 pieces are supplied *overall*
  - the condition is defined on *aggregate values*
- The **WHERE** clause may not be used for this purpose

## Group selection condition (no.1)

- Find the overall quantity of supplied pieces for the products for which at least 600 pieces are supplied *overall*

SP

<u>SId</u>	<u>PId</u>	Qty
S1	P1	300
S1	P2	200
S1	P3	400
S1	P4	200
S1	P5	100
S1	P6	100
S2	P1	300
S2	P2	400
S3	P2	200
S4	P3	200
S4	P4	300
S4	P5	400

SP

<u>SId</u>	<u>PId</u>	Qty
S1	P1	300
S2	P1	300
S1	P2	200
S2	P2	400
S3	P2	200
S1	P3	400
S4	P3	200
S1	P4	200
S4	P4	300
S1	P5	100
S4	P5	400
S1	P6	100

R

<u>PId</u>	
P1	600
P2	800
P3	600

## Group selection condition (no.1)

- Find the overall quantity of supplied pieces for the products for which at least 600 pieces are supplied *overall*

```
SELECT PId, SUM(Qty)
FROM SP
GROUP BY PId
HAVING SUM(Qty) >= 600;
```

- The **HAVING** clause allows the specification of conditions on the aggregate functions

## Group selection condition (no.2)

➤ Find the codes of the red products supplied by more than one supplier

P

<u>PId</u>	PName	Color	Size	Store
P1	Jumper	Red	40	London
P2	Jeans	Green	48	Paris
P3	Blouse	Blue	48	Rome
P4	Blouse	Red	44	London
P5	Skirt	Blue	40	Paris
P6	Shorts	Red	42	London

SP

<u>SId</u>	<u>PId</u>	Qty
S1	P1	300
S1	P2	200
S1	P3	400
S1	P4	200
S1	P5	100
S1	P6	100
S2	P1	300
S2	P2	400
S3	P2	200
S4	P3	200
S4	P4	300
S4	P5	400

## Group selection condition (no.2)

- Find the codes of the red products supplied by more than one supplier

```
SELECT SP.PId
FROM SP, P
WHERE SP.PId=P.PId AND Color='Red'
GROUP BY SP.PId
HAVING COUNT(*)>1;
```

## Group selection condition (no.2)

➤ Find the codes of the red products supplied by more than one supplier

S.SId	S.PId	S.Qty	P.PId	P.PName	P.Color	P.Size	P.Store
S1	P1	300	P1	Jumper	Red	40	London
S2	P1	300	P1	Jumper	Red	40	London
S1	P6	100	P6	Shorts	Red	42	London



R

PId
P1



# Structure of the SELECT statement

```
SELECT [DISTINCT] ListOfAttributesToDisplay
FROM ListOfTablesToUse
[WHERE TupleConditions ]
[GROUP BY ListOfGroupingAttributes ]
[HAVING AggregateConditions ]
[ORDER BY ListOfOrderingAttributes ];
```