Politecnico
di Torino

1859

DBMG

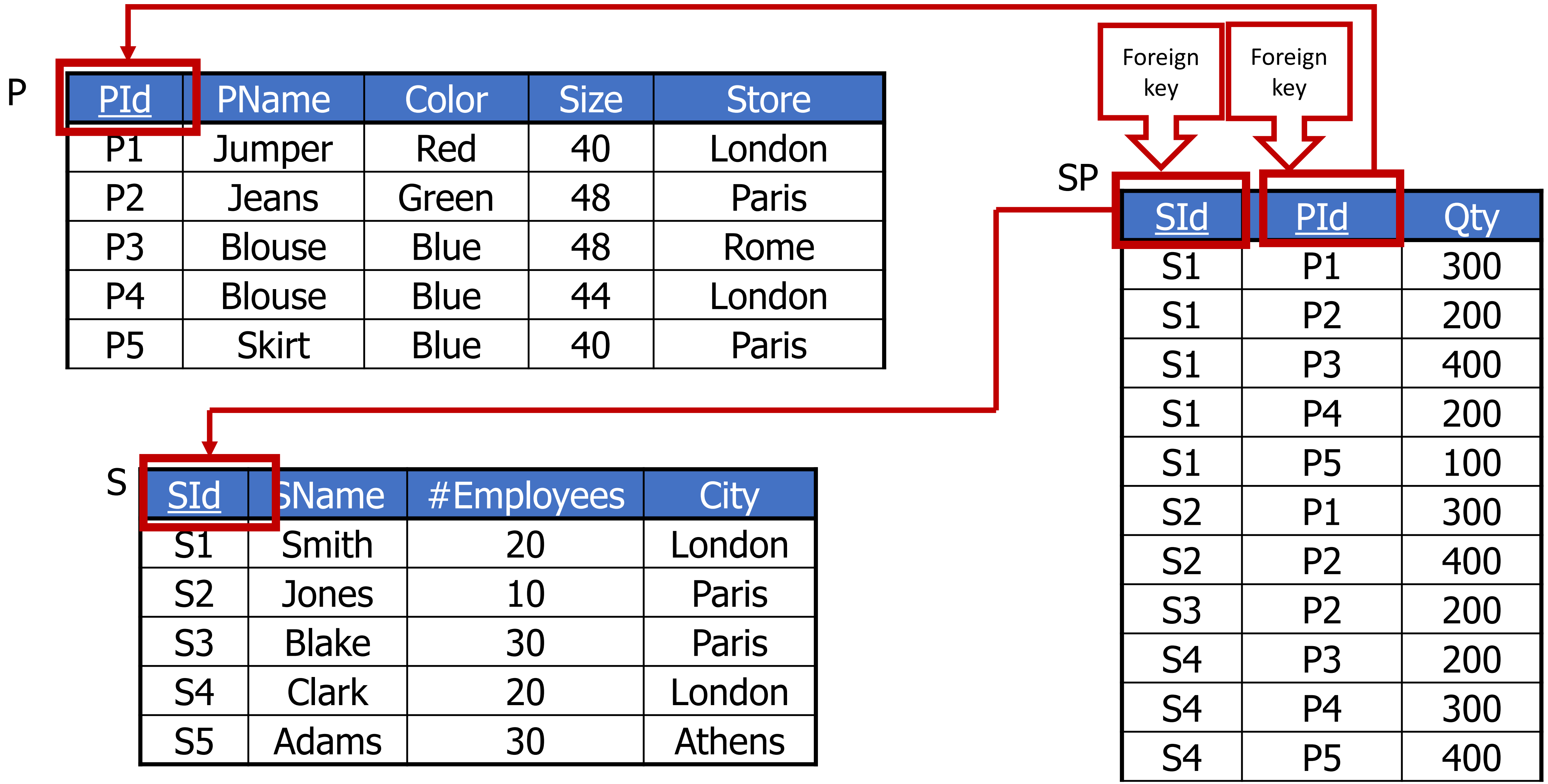
Nested queries

SQL language

Introduction

- A nested query is a SELECT statement contained within another query
  - query nesting allows decomposing a complex problem into simpler subproblems
- SELECT statements may be introduced
  - within a predicate in the WHERE clause
  - within a predicate in the HAVING clause
  - in the FROM clause

Example database: Supply-Product



Foreign key

Foreign key

P

PId	PName	Color	Size	Store
P1	Jumper	Red	40	London
P2	Jeans	Green	48	Paris
P3	Blouse	Blue	48	Rome
P4	Blouse	Blue	44	London
P5	Skirt	Blue	40	Paris

SP

SId	PId	Qty
S1	P1	300
S1	P2	200
S1	P3	400
S1	P4	200
S1	P5	100
S2	P1	300
S2	P2	400
S3	P2	200
S4	P3	200
S4	P4	300
S4	P5	400

S

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

Nested queries (no.1)

- Find the codes of the suppliers that are based in the same city as S1

- By using a formulation with nested queries, the problem may be decomposed into two subproblems
  - city of supplier S1
  - codes of the suppliers based in the same city

Nested queries (no.1)

- Find the codes of the suppliers that are based in the same city as S1

IDs of the suppliers based in the same city as S1

```
SELECT SId
FROM S
WHERE City = (SELECT City
              FROM S
              WHERE SId=‘S1’);
```

City of supplier S1

- The ‘=’ operator may be used only if it is known in advance that the inner SELECT statement always returns a single value
- An equivalent formulation may be defined using a join operation

Equivalent formulation

- The equivalent formulation with join is characterized by
  - a FROM clause including all the tables referenced by the FROM clauses of each SELECT statement
  - appropriate join conditions in the WHERE clause
  - if needed selection predicates added in the WHERE clause

Equivalent formulation

- Find the codes of the suppliers that are based in the same city as S1

FROM clause

```
SELECT SId
FROM S                           SX
WHERE City = (SELECT City
              FROM S             SY
              WHERE SId='S1');
```

```
SELECT ...
FROM S AS SX, S AS SY
...
```

JOIN condition

```
SELECT SId
FROM S
WHERE City = (SELECT City
              FROM S
              WHERE SId='S1');
```

```
SELECT ...
FROM S AS SX, S AS SY
WHERE SX.City=SY.City
...
```

SELECTION predicate, SELECT clause

```
SELECT SId
FROM S
WHERE City = (SELECT City
              FROM S
              WHERE SId='S1');
```

```
SELECT SY.SId
FROM S AS SX, S AS SY
WHERE SX.City=SY.City AND
      SX.SId='S1';
```

Equivalent formulation

- Find the codes of the suppliers whose number of employees is smaller than the maximum number of employees

```
SELECT SId
FROM S
WHERE #Employees < (SELECT MAX(#Employees)
                    FROM S);
```

- An equivalent formulation with join is not possible

IN OPERATOR

- It expresses the concept of membership to a set of values

AttributeName IN *(NestedQuery)*

- It allows to write a query by
  - breaking down the problem into subproblems
  - following a "bottom-up" process
- The nested query can be replaced with a list of values
- The equivalent formulation with the join is characterized by
  - FROM clause containing the tables referenced in the FROM of all SELECTs
  - appropriate join conditions in the WHERE clause
  - any selection predicates added in the WHERE clause

The IN operator (no.1)

- Find the names of the suppliers who supply product P2

- Decomposition of the problem into two subproblems
  - codes of the suppliers of product P2
  - names of the suppliers with such codes

The IN operator (no.1)

- Find the names of the suppliers who supply product P2

SP

SId	PId	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

SId
S1
S2
S3

```
(SELECT SId
 FROM SP
 WHERE PId=‘P2’)
```

Codes of the suppliers of P2

The IN operator (no.1)

- Find the names of the suppliers who supply product P2

S

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

Set membership

```
SELECT SName
FROM S
WHERE SId IN (SELECT SId
              FROM SP
              WHERE PId='P2');
```

Codes of the suppliers of P2

SId
S1
S2
S3

Example 1: Equivalent Formulation with Join

- Find the names of the suppliers who supply product P2

IN

```
SELECT SName
FROM S
WHERE SId IN (SELECT SId
              FROM SP
              WHERE PId=‘P2’);
```

JOIN

```
SELECT SName
FROM S, SP
WHERE S.SId=SP.SId
      AND PId=‘P2’;
```

Example 2: IN Operator

- Find the name of suppliers who provide at least one red product

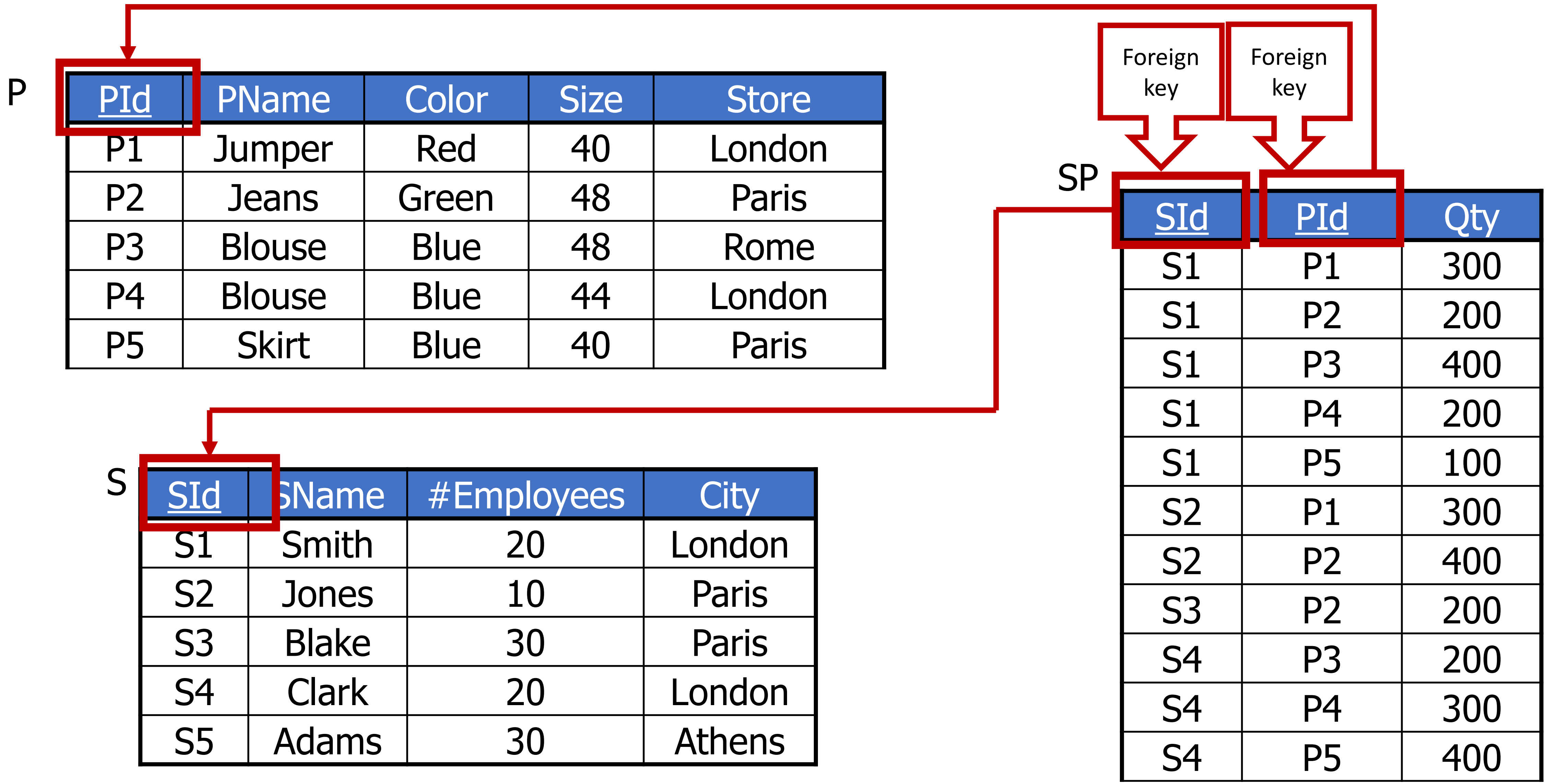
```
SELECT SName
FROM S
WHERE SId IN (SELECT SId
              FROM SP
              WHERE PId IN (SELECT PId
                            FROM P
                            WHERE Color=‘Red’));
```

Red Product Codes

Red Product Supplier Codes

Supplier names with those codes

Example database: Supply-Product



P

PId	PName	Color	Size	Store
P1	Jumper	Red	40	London
P2	Jeans	Green	48	Paris
P3	Blouse	Blue	48	Rome
P4	Blouse	Blue	44	London
P5	Skirt	Blue	40	Paris

S

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

Foreign key

Foreign key

SP

SId	PId	Qty
S1	P1	300
S1	P2	200
S1	P3	400
S1	P4	200
S1	P5	100
S2	P1	300
S2	P2	400
S3	P2	200
S4	P3	200
S4	P4	300
S4	P5	400

Example 2: Equivalent formulation

- Find the name of suppliers who provide at least one red product

IN

```
SELECT SName
FROM S                                 1
WHERE SId IN
(SELECT SId
         FROM SP
         WHERE PId
          IN (SELECT PId
                   FROM P
                   WHERE Color=‘Red’));
```

JOIN

```
SELECT SName
FROM S,SP                  1
WHERE S.SId=SP.SId
```

Example 2: Equivalent formulation

- Find the name of suppliers who provide at least one red product

IN

```
SELECT SName
FROM S
WHERE SId IN
(SELECT SId
        FROM SP
        WHERE PId
        IN (SELECT PId
                FROM P
                WHERE Color=‘Red’));
```

2

JOIN

```
SELECT SName
FROM S, SP, P
WHERE S.SId=SP.SId AND
      P.SId= SP.SId
```

2

Example 2: Equivalent formulation

- Find the name of suppliers who provide at least one red product

IN

```
SELECT SName
FROM S
WHERE SId IN
(SELECT SId
          FROM SP
          WHERE PId
          IN (SELECT PId
                    FROM P
                    WHERE Color=‘Red’));
```

3

JOIN

```
SELECT SName
FROM S, SP, P
WHERE S.SId=SP.SId AND
          P.SId= SP.Sid AND
          Color = ‘Red’
```

3

NOT IN OPERATOR

- It expresses the concept of exclusion from a set of values

AttributeName NOT IN *(NestedQuery)*

- It requires the identification of an appropriate *set to be excluded* defined by
  - a nested query
  - a list of values
- There is no equivalent formulation with join

Example 1: Concept of exclusion

- Find the names of the suppliers who *do not* supply product P2
  - it is not possible to express the query with a join operation

```
SELECT SName
FROM S, SP
WHERE S.SId = SP.SId
      AND PId <>'P2';
```

Wrong solution

- The query matches the request:
  - Find the name of suppliers who provide at least one product other than P2

Wrong solution (no.1)

- Find the names of the suppliers who *do not* supply product P2

S

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

SP

SId	PId	Qty
S1	P1	300
S1	P2	200
S1	P3	400
S1	P4	200
S1	P5	100
S2	P1	300
S2	P2	400
S3	P2	200
S4	P3	200
S4	P4	300
S4	P5	400

R

SName
Smith
Jones
Clark

NOT IN operator (no.1)

- Find the names of the suppliers who *do not* supply product P2
- Set to be excluded
  - suppliers of product P2

```
SELECT SName
FROM S
WHERE SId NOT IN (SELECT SId
                  FROM SP
                  WHERE PId=‘P2’);
```

Codes of the suppliers who supply P2

does not belong to

NOT IN operator (no.2)

- Find the names of the suppliers who only supply product P2

⬇

Find the names of the suppliers of P2 who have never supplied products other than P2

- Set to be excluded
  - suppliers of products other than P2

NOT IN operator (no.2)

- Find the names of the suppliers who only supply product P2

```
SELECT SName
FROM S
WHERE S.SId NOT IN (SELECT SId
                    FROM SP
                    WHERE PId<>‘P2’)

        AND S.SId IN (SELECT SId
                    FROM SP);
```

Codes of the suppliers who supply at least one product other than P2

Codes of the suppliers who supply at least one product

Alternative solution (no.2)

- Find the names of the suppliers who only supply product P2

```
SELECT SName
FROM S, SP
WHERE S.SId NOT IN (SELECT SId
               FROM SP
               WHERE PId<>‘P2’)
      AND S.SId=SP.SId;
```

Codes of the suppliers who supply at least one product other than P2

NOT IN operator (no.3)

- Find the names of the suppliers who *do not* supply any red products
- Set to be excluded:
  - suppliers of red products, identified by their codes

```
SELECT SName
FROM S
WHERE SId NOT IN
                    (SELECT SId
                    FROM SP
                    WHERE PId IN  (SELECT PId
                                   FROM P
                                   WHERE Color=‘Red’));
```

Codes of the suppliers of at least one red product

Wrong alternative (no.3)

- Find the names of the suppliers who *do not* supply any red products

```
SELECT SName
FROM S
WHERE SId IN
        (SELECT SId
         FROM SP
         WHERE PId NOT IN (SELECT PId
                           FROM P
                           WHERE Color='Red'));
```

Codes of the suppliers of non-red products

- The set of elements to be excluded is incorrect

Wrong alternative (no.3)

- Find the names of the suppliers who *do not* supply any red products

P

PId	PName	Color	Size	Store
P1	Jumper	Red	40	London
P2	Jeans	Green	48	Paris
P3	Blouse	Blue	48	Rome
P4	Blouse	Blue	44	London
P5	Skirt	Blue	40	Paris

S

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

SP

SId	PId	Qty
S1	P1	300
S1	P2	200
S1	P3	400
S1	P4	200
S1	P5	100
S2	P1	300
S2	P2	400
S3	P2	200
S4	P3	200
S4	P4	300
S4	P5	400