4.2

SQL: querying the database

- **Kinds of query in relational algrebraic terms**
- **Simple selection queries**
- **Use of basic search conditions and operators**
- **Ordering results**

Queries as relational algebra

- Relational algebra operations work on one or more relations to produce another relation
- Both operands and results are relations, so output from one operation can become input to another operation.
- Operations can be nested (as in arithmetic).

Relational Algebra

- Five basic operations in relational algebra: Selection, Projection, Cartesian product, Union, and Set Difference.

- These perform most of the data retrieval operations needed.

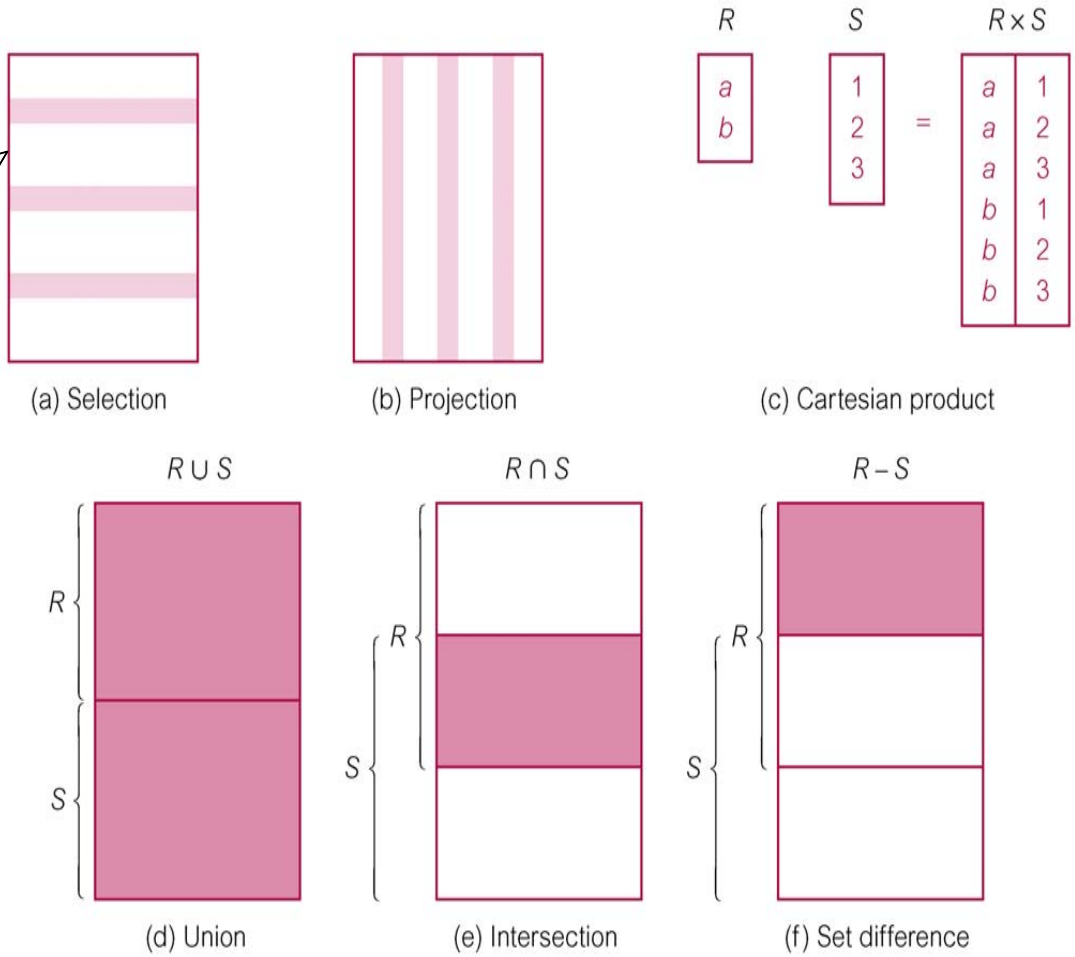
- Also have Join, Intersection, and Division operations, which can be expressed in terms of 5 basic operations.

Relational Algebra Operations

Selection

- defines a relation that contains only those tuples satisfying some condition

e.g., List all staff with a salary greater than 10k



(a) Selection (b) Projection (c) Cartesian product

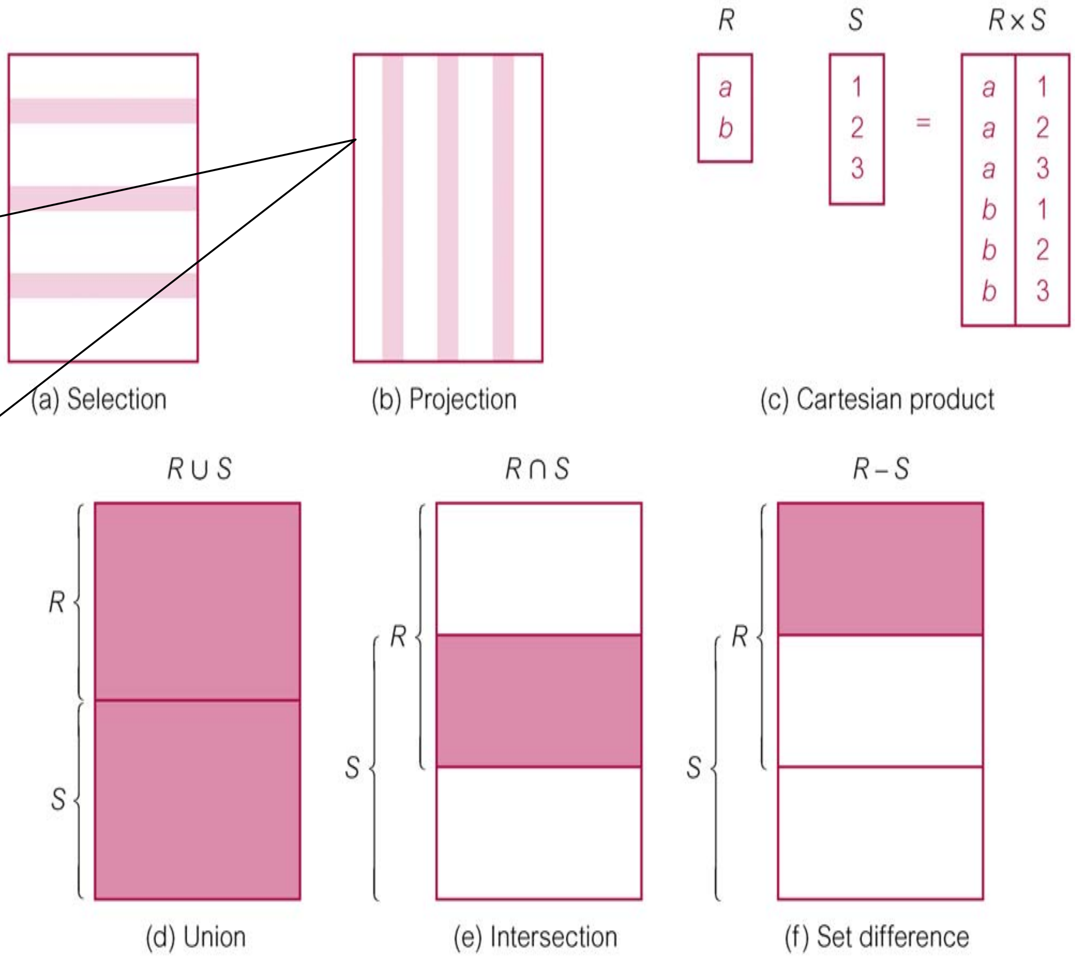
(d) Union (e) Intersection (f) Set difference

Relational Algebra Operations

Projection

- defines a relation containing selected attributes

e.g. list salaries for all staff showing only staffNo, name and salary details



(a) Selection

(b) Projection

(c) Cartesian product

(d) Union

(e) Intersection

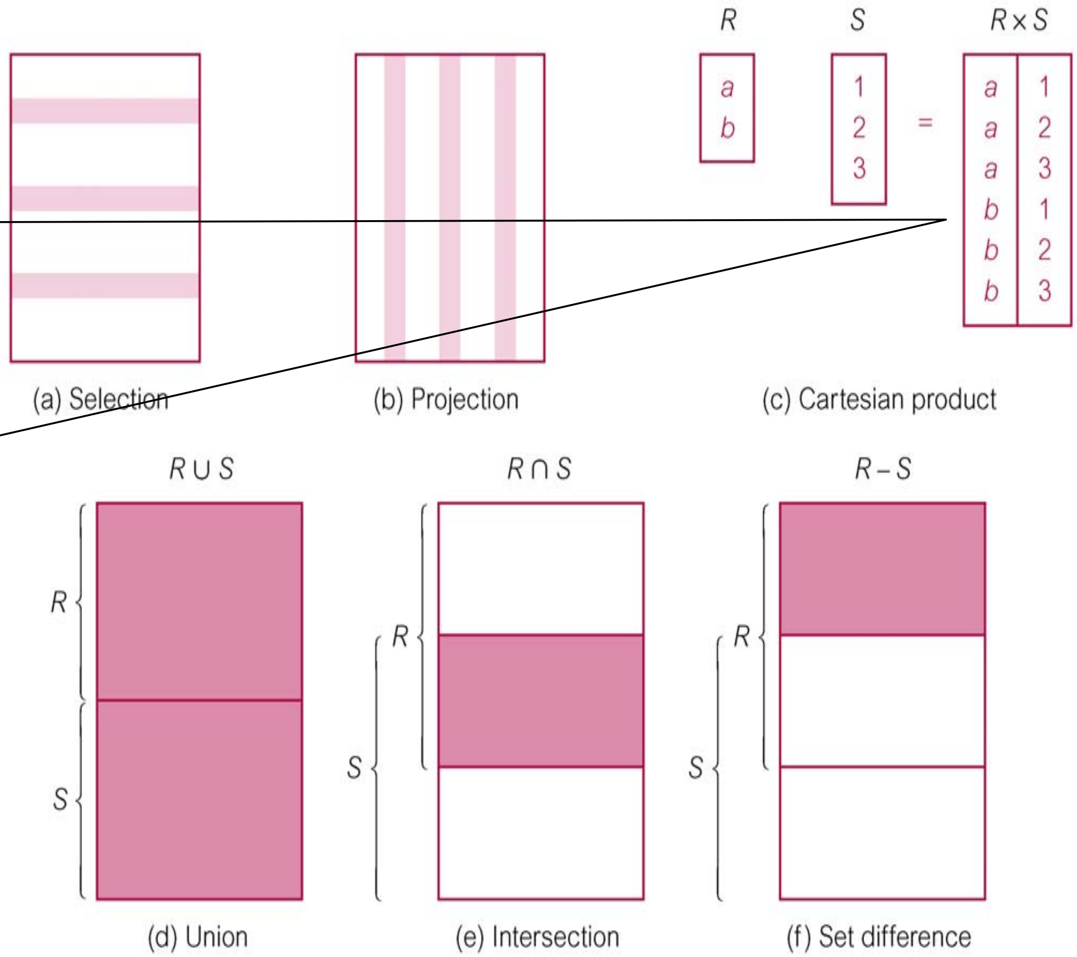
(f) Set difference

Relational Algebra Operations

Cartesian product

- multiplies 2 relations to define all pairs of tuples from the two relations

e.g., List the names and comments of all clients who have viewed a certain property



(a) Selection (b) Projection (c) Cartesian product

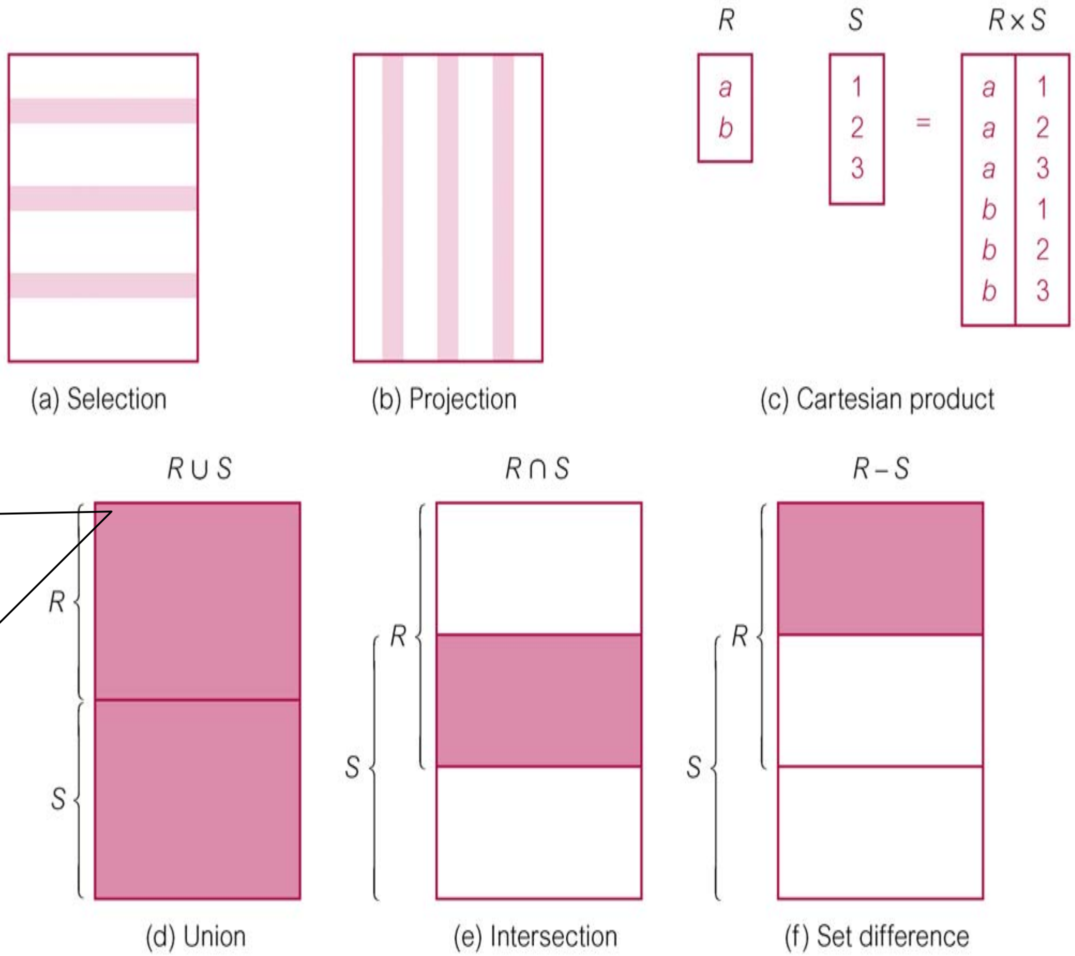
(d) Union (e) Intersection (f) Set difference

Relational Algebra Operations

Union

-all tuples of two relations

e.g., list all cities where there is either a branch office or a property to rent



(a) Selection

(b) Projection

(c) Cartesian product

(d) Union

(e) Intersection

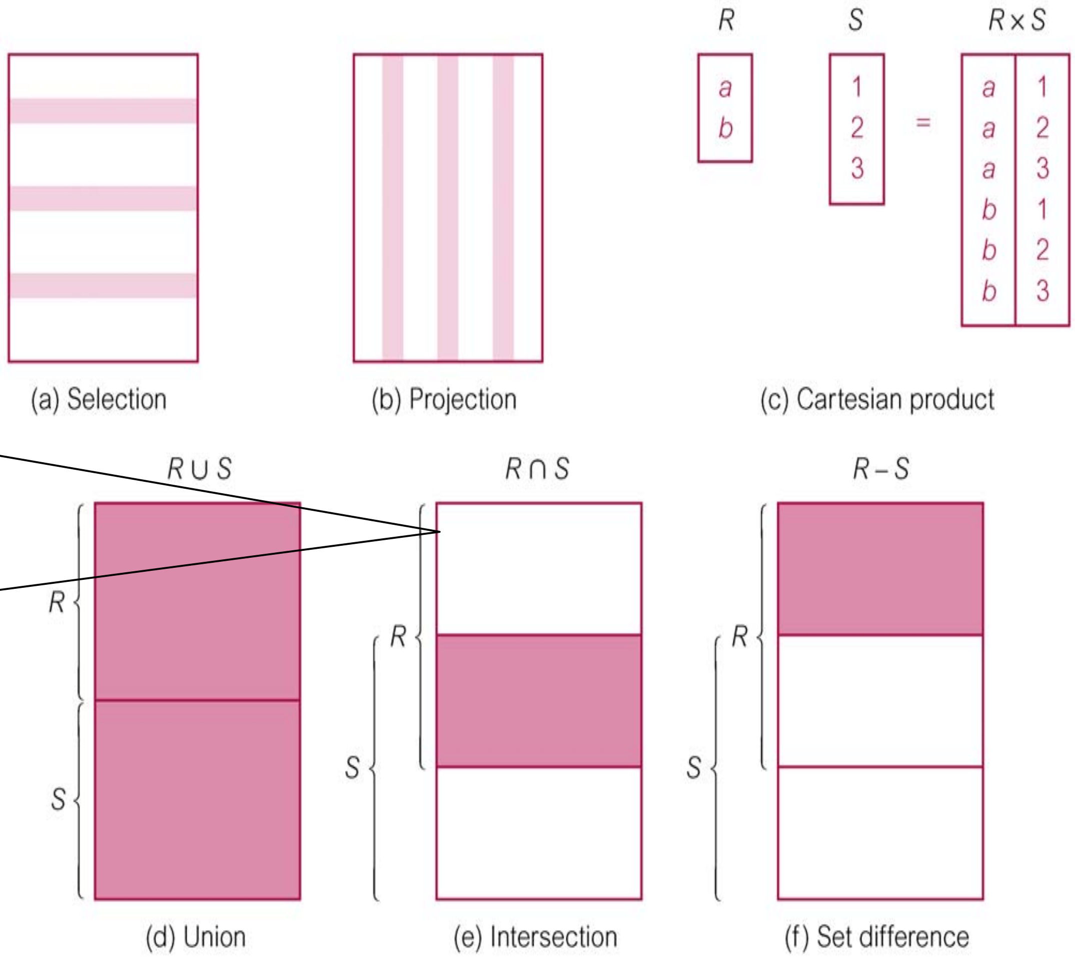
(f) Set difference

Relational Algebra Operations

Intersection

-all tuples existing in two relations

e.g., list all cities where there is both a branch office and at least one property to rent



(a) Selection

(b) Projection

(c) Cartesian product

(d) Union

(e) Intersection

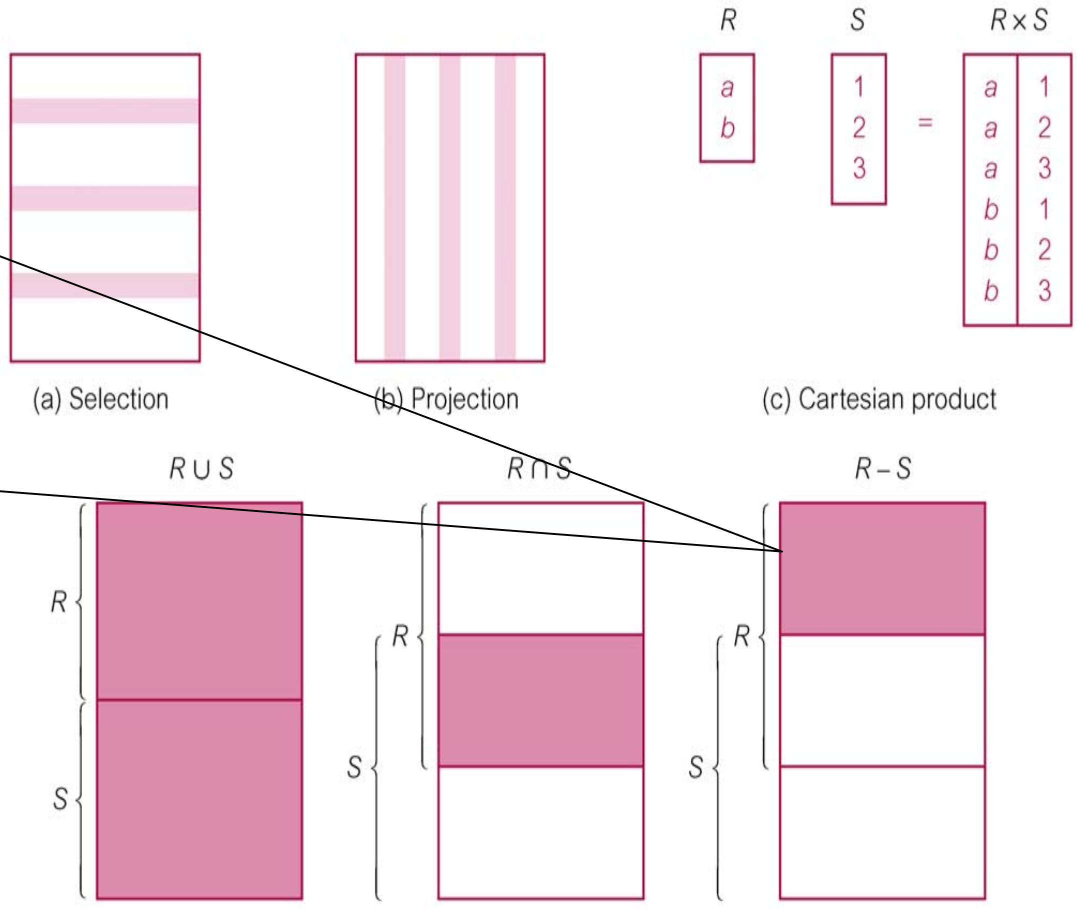
(f) Set difference

Relational Algebra Operations

Set difference

– all the tuples in relation R and not in S

e.g., list all cities where there is a branch office but no properties for rent



(a) Selection

(b) Projection

(c) Cartesian product

(d) Union

(e) Intersection

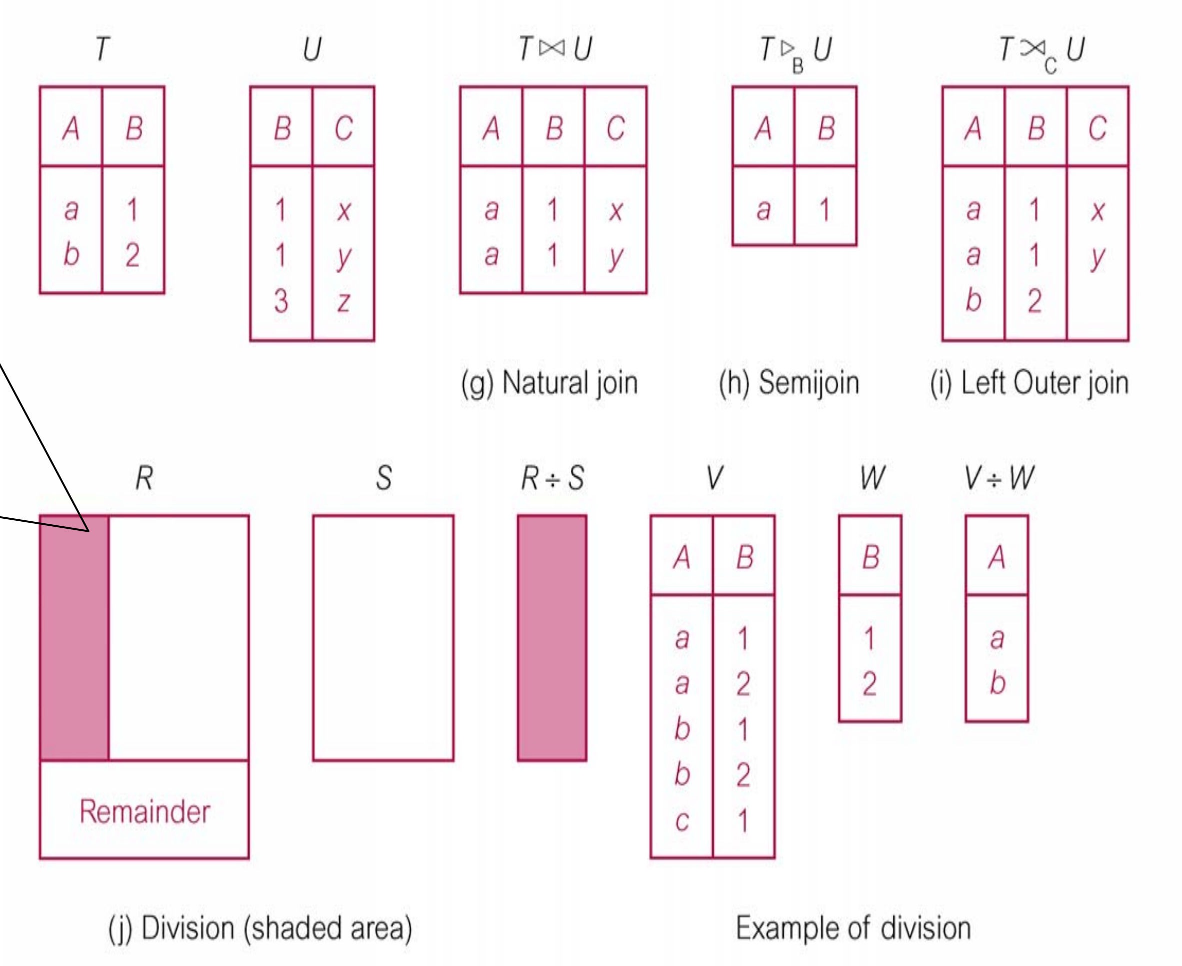
(f) Set difference

Relational Algebra Operations

Division

-the set of tuples in R that match the combination of every tuple in S

- –E.g., Identify all clients who have viewed all properties with 3 rooms



T

A	B
a	1
b	2

U

B	C
1	x
1	y
3	z

$T \bowtie U$

A	B	C
a	1	x
a	1	y

(g) Natural join

$T \triangleright_B U$

A	B
a	1

(h) Semijoin

$T ⟕_C U$

A	B	C
a	1	x
a	1	y
b	2	

(i) Left Outer join

(j) Division (shaded area)

V

A	B
a	1
a	2
b	1
b	2
c	1

W

B
1
2

$V \div W$

A
a
b

Example of division

Queries are expressed as SELECT statements:

- SELECT Specifies which columns are to appear in output.
- FROM Specifies table(s) to be used.
- WHERE Filters rows.
- GROUP BY Forms groups of rows with same column value.
- HAVING Filters groups subject to some condition.
- ORDER BY Specifies the order of the results

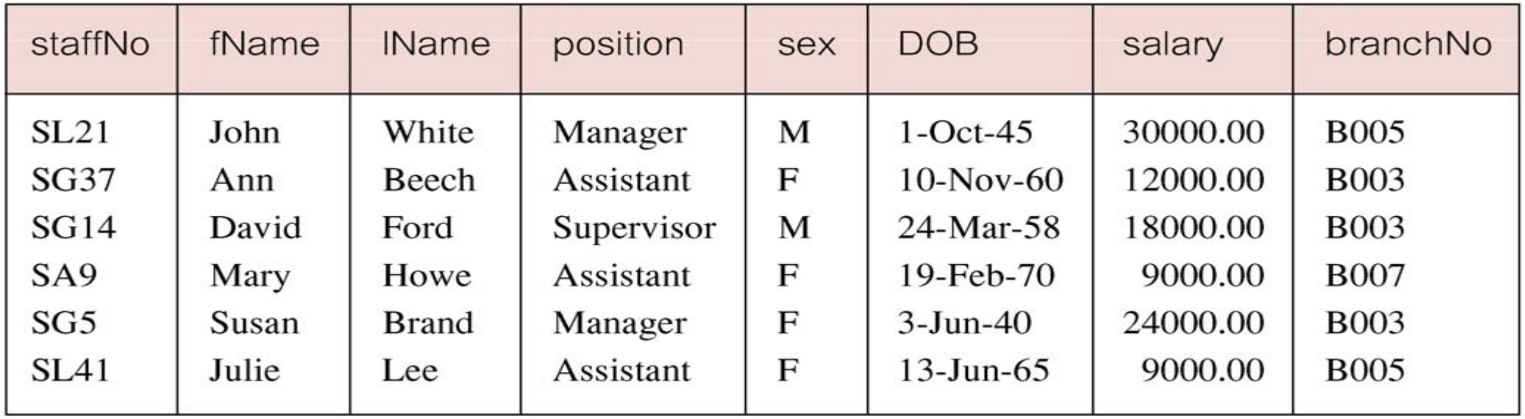
Order of these clauses cannot be changed. Only SELECT and FROM are mandatory.

Simple selection: All Columns, All Rows

- List full details of all staff.

Can use * for all columns: 'SELECT *'

```
SELECT staffNo, fName, lName, address,
       position, sex, DOB, salary, branchNo
FROM Staff;
```

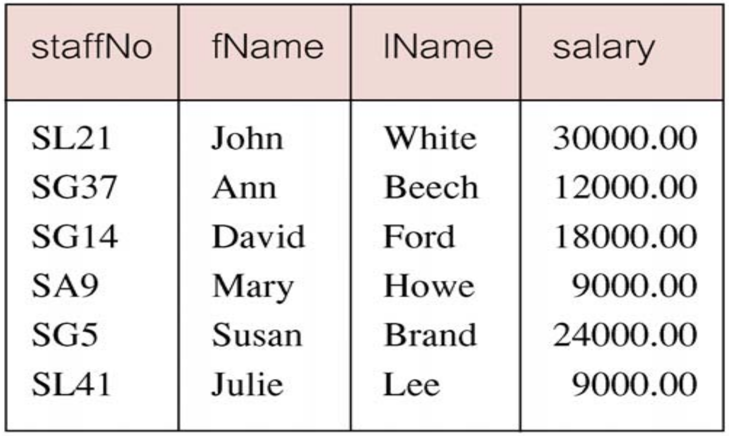


staffNo	fName	lName	position	sex	DOB	salary	branchNo
SL21	John	White	Manager	M	1-Oct-45	30000.00	B005
SG37	Ann	Beech	Assistant	F	10-Nov-60	12000.00	B003
SG14	David	Ford	Supervisor	M	24-Mar-58	18000.00	B003
SA9	Mary	Howe	Assistant	F	19-Feb-70	9000.00	B007
SG5	Susan	Brand	Manager	F	3-Jun-40	24000.00	B003
SL41	Julie	Lee	Assistant	F	13-Jun-65	9000.00	B005

Selecting specific columns, all rows

- Produce a list of salaries for all staff, showing only staff number, first and last names, and salary.

```
SELECT staffNo, fName, lName, salary
FROM Staff;
```



staffNo	fName	lName	salary
SL21	John	White	30000.00
SG37	Ann	Beech	12000.00
SG14	David	Ford	18000.00
SA9	Mary	Howe	9000.00
SG5	Susan	Brand	24000.00
SL41	Julie	Lee	9000.00

Use of DISTINCT

- List the property numbers of all properties that have been viewed.

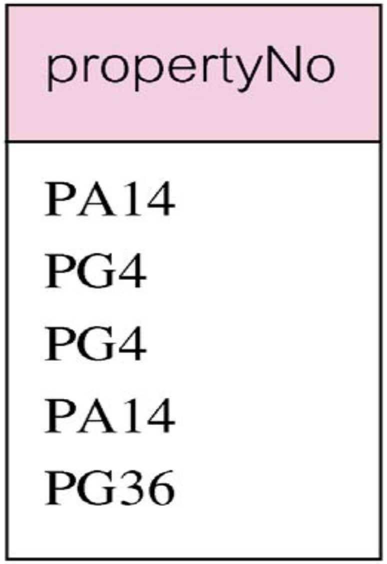
propertyNo	address	viewDate	clientNo	comments

table: 'Viewing'

Use of DISTINCT

- List the property numbers of all properties that have been viewed.

```
SELECT propertyNo
FROM Viewing;
```

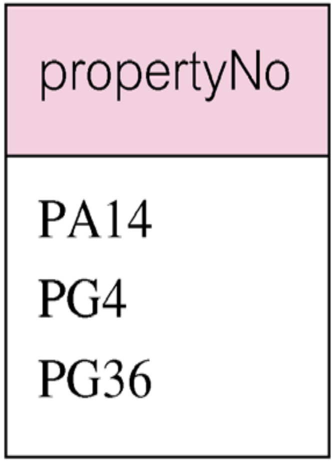


propertyNo
PA14
PG4
PG4
PA14
PG36

Use of DISTINCT

- Use DISTINCT to eliminate duplicates:

```
SELECT    DISTINCT propertyNo
FROM      Viewing;
```

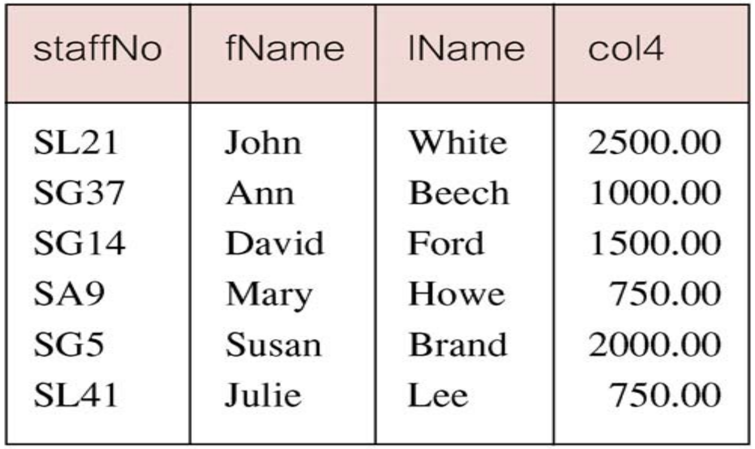


propertyNo
PA14
PG4
PG36

Calculated Fields

- Produce a list of monthly salaries for all staff, showing staff number, first and last names, and salary details.

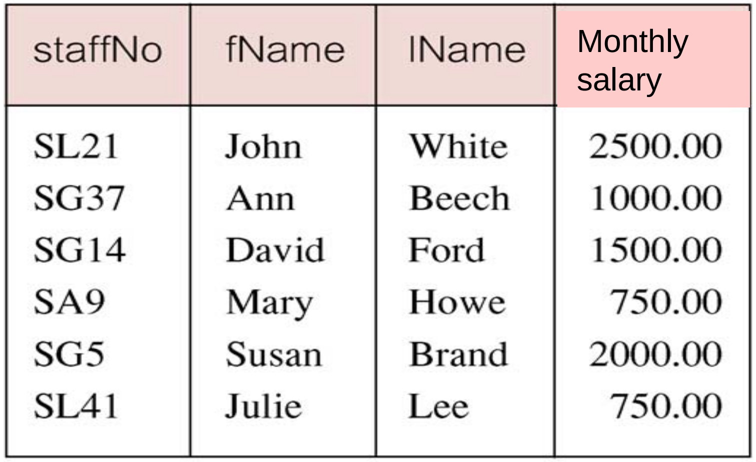
```
SELECT     staffNo, fName, lName, salary/12
FROM       Staff;
```



staffNo	fName	lName	col4
SL21	John	White	2500.00
SG37	Ann	Beech	1000.00
SG14	David	Ford	1500.00
SA9	Mary	Howe	750.00
SG5	Susan	Brand	2000.00
SL41	Julie	Lee	750.00

Giving titles to results lists: 'AS'

```
SELECT     staffNo, fName, lName, salary/12
AS monthlySalary
FROM       Staff;
```

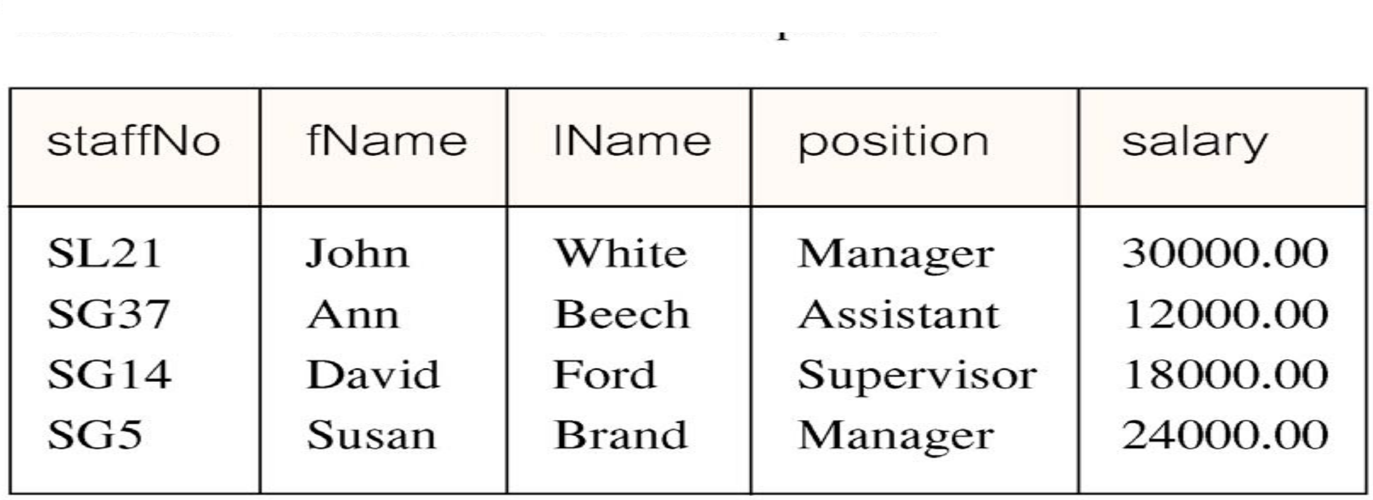


staffNo	fName	lName	Monthly salary
SL21	John	White	2500.00
SG37	Ann	Beech	1000.00
SG14	David	Ford	1500.00
SA9	Mary	Howe	750.00
SG5	Susan	Brand	2000.00
SL41	Julie	Lee	750.00

Comparison Search

- List all staff with a salary greater than 10,000.

```
SELECT staffNo, fName, lName, position, salary
FROM       Staff
WHERE salary > 10000;
```

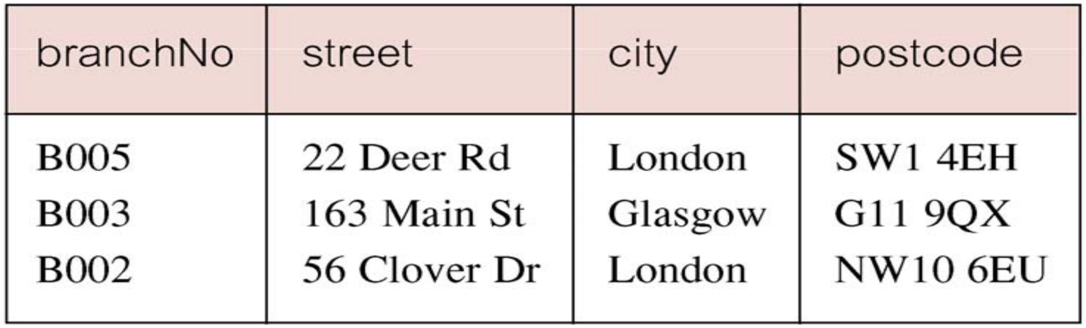


staffNo	fName	lName	position	salary
SL21	John	White	Manager	30000.00
SG37	Ann	Beech	Assistant	12000.00
SG14	David	Ford	Supervisor	18000.00
SG5	Susan	Brand	Manager	24000.00

Compound Comparison Search

- List addresses of all branch offices in London or Glasgow.

```
SELECT    *
FROM      Branch
WHERE     city = ‘London’ OR city =
          ‘Glasgow’;
```

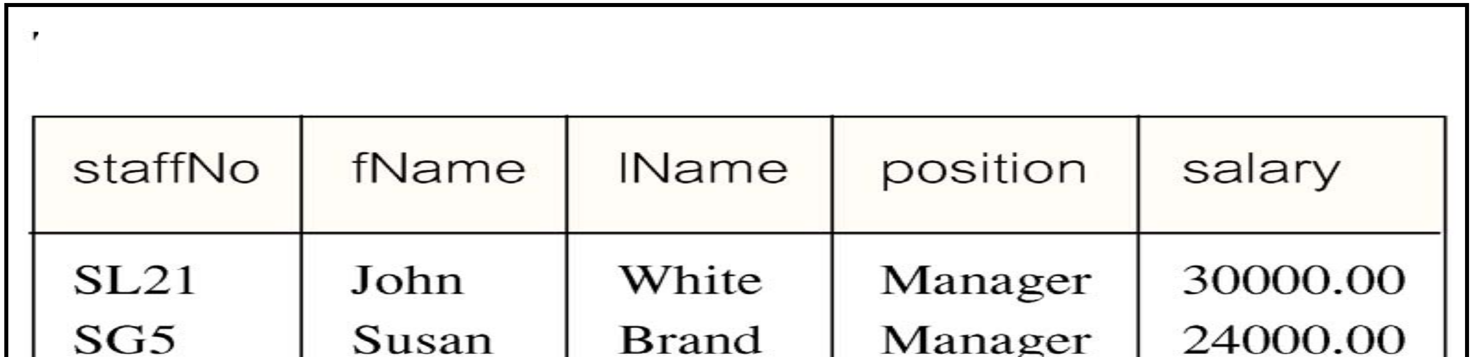


branchNo	street	city	postcode
B005	22 Deer Rd	London	SW1 4EH
B003	163 Main St	Glasgow	G11 9QX
B002	56 Clover Dr	London	NW10 6EU

Range Search Condition: 'BETWEEN'

- List all staff with a salary between 20,000 and 30,000.

```
 SELECT staffNo, fName, lName, position,
salary
 FROM   Staff
 WHERE salary BETWEEN 20000 AND
   30000;
```



staffNo	fName	lName	position	salary
SL21	John	White	Manager	30000.00
SG5	Susan	Brand	Manager	24000.00

Range Search Condition

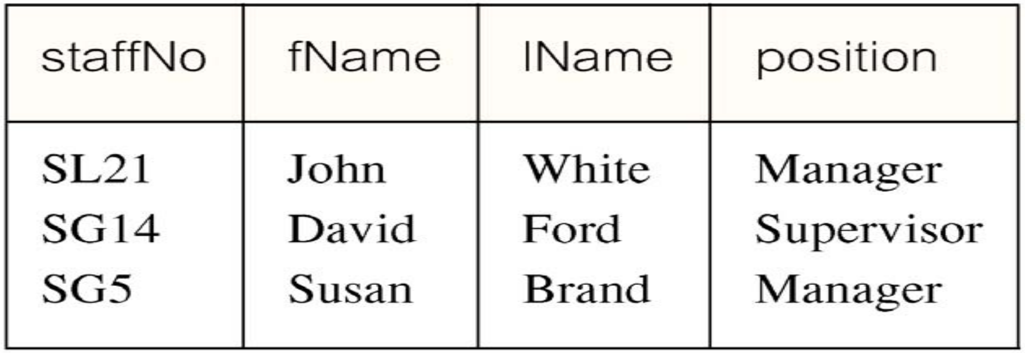
- also a negated version: NOT BETWEEN.
- an alternative to using GT and LT

```
  SELECT staffNo, fName, lName, position,
   salary
 FROM   Staff WHERE salary>=20000
   AND salary <= 30000;
```

Set membership search: use of ‘IN’

- List all managers and supervisors.

```
SELECT staffNo, fName, lName, position
FROM Staff
WHERE position IN (‘Manager’, ‘Supervisor’);
```



staffNo	fName	lName	position
SL21	John	White	Manager
SG14	David	Ford	Supervisor
SG5	Susan	Brand	Manager

Set Membership – use of ‘IN’

negated version: ‘NOT IN’

alternatively, use

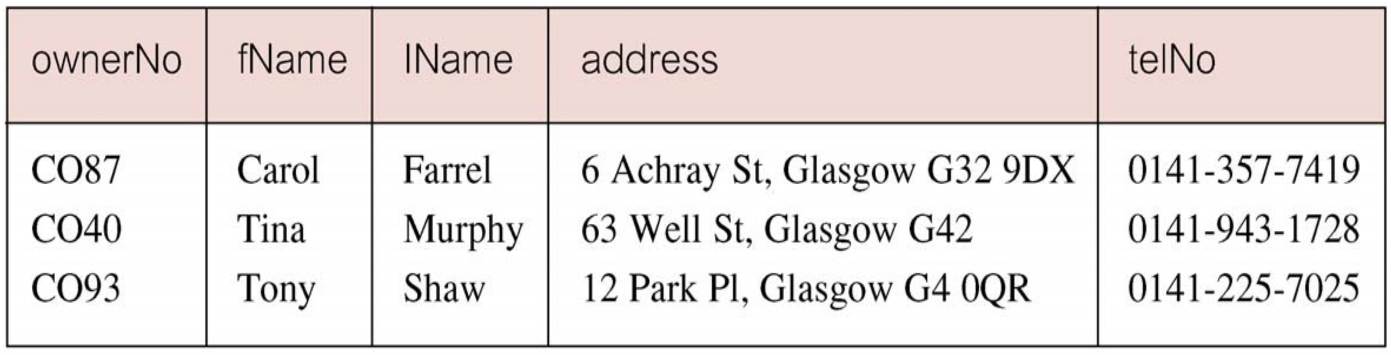
```
SELECT staffNo, fName, lName, position
FROM Staff
  WHERE position=‘Manager’ OR
   position=‘Supervisor’;
```

IN is more efficient when set contains many values.

Pattern Matching: use of 'LIKE'

- Find all owners with the string 'Glasgow' in their address.

```
SELECT    clientNo, fName, lName, address,
  telNo
FROM      PrivateOwner
WHERE     address LIKE '%Glasgow%';
```



ownerNo	fName	lName	address	telNo
CO87	Carol	Farrel	6 Achray St, Glasgow G32 9DX	0141-357-7419
CO40	Tina	Murphy	63 Well St, Glasgow G42	0141-943-1728
CO93	Tony	Shaw	12 Park Pl, Glasgow G4 0QR	0141-225-7025

Pattern Matching

SQL has two special pattern matching symbols:

%: sequence of zero or more characters;

_ (underscore): any single character.

LIKE '%Glasgow%' means a sequence of characters of any length containing 'Glasgow'

propertyNo	address	viewDate	clientNo	comments

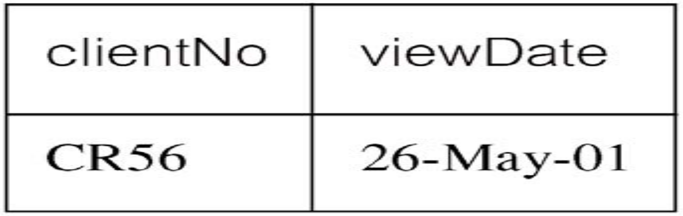
table: 'Viewing'

‘IS NULL’ Search Condition

- List details of all viewings on property PG4 where a comment has not been supplied.

```
SELECTclientNo, viewDate
FROM   Viewing
WHERE propertyNo = ‘PG4’ AND comment
  IS NULL;
```

◆Negated version (IS NOT NULL)

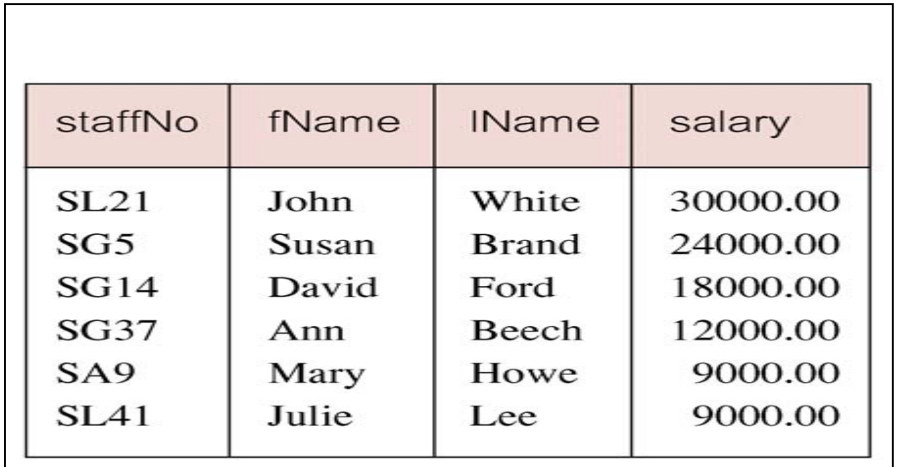


clientNo	viewDate
CR56	26-May-01

Ordering results lists: ‘ORDER BY’

- List salaries for all staff, arranged in descending order of salary.

```
SELECT staffNo, fName, lName, salary
 FROM   Staff
 ORDER BY salary DESC;
```



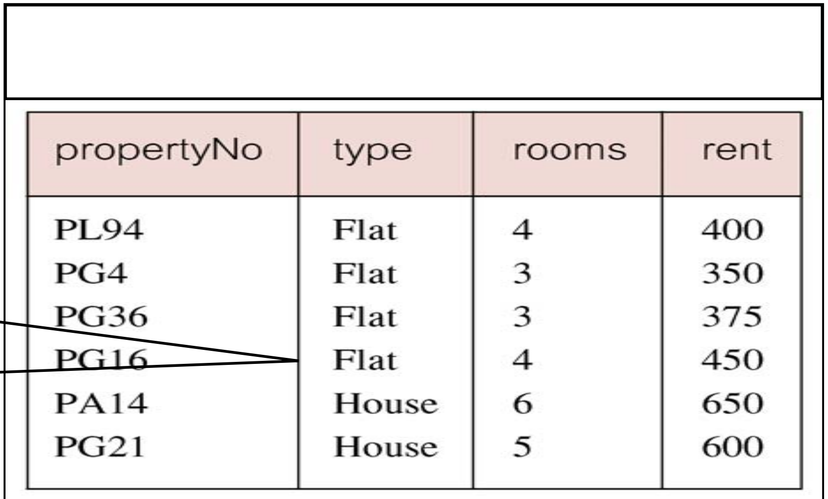
staffNo	fName	lName	salary
SL21	John	White	30000.00
SG5	Susan	Brand	24000.00
SG14	David	Ford	18000.00
SG37	Ann	Beech	12000.00
SA9	Mary	Howe	9000.00
SL41	Julie	Lee	9000.00

Multiple Column Ordering of results lists

- Produce an abbreviated list of properties in order of property type.

```
SELECT     propertyNo, type, rooms, rent
FROM       PropertyForRent
ORDER      BY type;
```

Four flats in this list - as no minor sort key specified, system arranges these rows in any order it chooses.

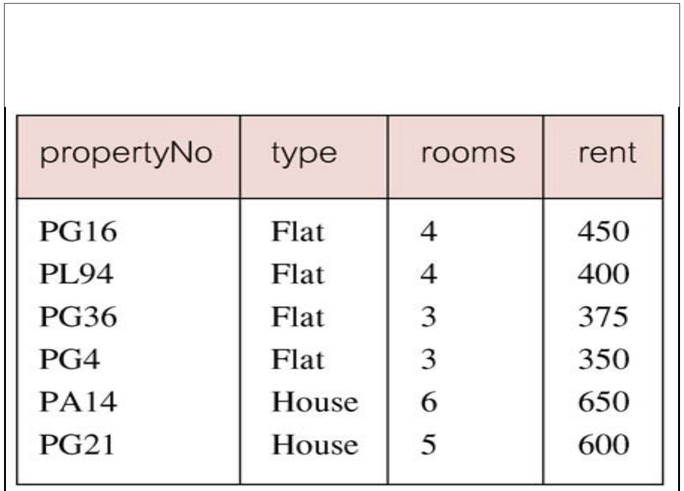


propertyNo	type	rooms	rent
PL94	Flat	4	400
PG4	Flat	3	350
PG36	Flat	3	375
PG16	Flat	4	450
PA14	House	6	650
PG21	House	5	600

Multiple Column Ordering

- Produce the same list, arranged in order of rent

```
SELECT propertyNo, type, rooms, rent
 FROM   PropertyForRent
 ORDER BY type, rent DESC;
```



propertyNo	type	rooms	rent
PG16	Flat	4	450
PL94	Flat	4	400
PG36	Flat	3	375
PG4	Flat	3	350
PA14	House	6	650
PG21	House	5	600