*Mother Nb	Mother LastName	Mother FirstName
16	King	Mary
22	Trudeau	Chantal
30	Ringer	Anne

Cartesian Product

Child LastName	Child FirstName	Mother Nb
King	Majorie	16
Tremblay	Pierre	22
King	John	16
Ford	Mary	16
Trembley	Marc	22
Cramer	Paul	NULL
Brown	Lesly	NULL

Mother Nb	Mother LastName	Mother FirstName	Child LastName	Child FirstName	Mother Nb
16	King	Marie	King	Majorie	16
16	King	Marie	Trembley	Pierre	22
16	King	Marie	King	John	16
16	King	Marie	Ford	MArie	16
16	King	Marie	Trembley	Marc	22
16	King	Marie	Cramer	Paul	NULL
16	King	Marie	Brown	Lesly	NULL
22	Trudeau	Chantal	King	Majorie	16
22	Trudeau	Chantal	Trembley	Pierre	22
22	Trudeau	Chantal	King	John	16
22	Trudeau	Chantal	Ford	Majorie	16
22	Trudeau	Chantal	Trembley	Marc	22
22	Trudeau	Chantal	Cramer	Paul	NULL
22	Trudeau	Chantal	Brown	Lesly	NULL
30	Ringer	Anne	King	Majorie	16
30	Ringer	Anne	Trembley	Pierre	22
30	Ringer	Anne	King	John	16
30	Ringer	Anne	Ford	MArie	16
30	Ringer	Anne	Trembley	Marc	22
30	Ringer	Anne	Cramer	Paul	NULL
30	Ringer	Anne	Brown	Lesly	NULL

```
SELECT *
FROM mother, child
ORDER BY mother.MotherNb
```

Cartesian Product

Mother Nb	Mother LastName	Mother FirstName	Child LastName	Child FirstName	Mother Nb
16	King	Marie	King	Majorie	16
16	King	Marie	Trembley	Pierre	22
16	King	Marie	King	John	16
16	King	Marie	Ford	MArie	16
16	King	Marie	Trembley	Marc	22
16	King	Marie	Cramer	Paul	NULL
16	King	Marie	Brown	Lesly	NULL
22	Trudeau	Chantal	King	Majorie	16
22	Trudeau	Chantal	Trembley	Pierre	22
22	Trudeau	Chantal	King	John	16
22	Trudeau	Chantal	Ford	MArie	16
22	Trudeau	Chantal	Trembley	Marc	22
22	Trudeau	Chantal	Cramer	Paul	NULL
22	Trudeau	Chantal	Brown	Lesly	NULL
30	Ringer	Anne	King	Majorie	16
30	Ringer	Anne	Trembley	Pierre	22
30	Ringer	Anne	King	John	16
30	Ringer	Anne	Ford	MArie	16
30	Ringer	Anne	Trembley	Marc	22
30	Ringer	Anne	Cramer	Paul	NULL
30	Ringer	Anne	Brown	Lesly	NULL

Table joins

The **Join** clause is a relational operation that combines then returns data from two or more tables. When declaring a **Join**, a comparison condition is needed to filter the data being returned, and the tables to be joined must share a common column.

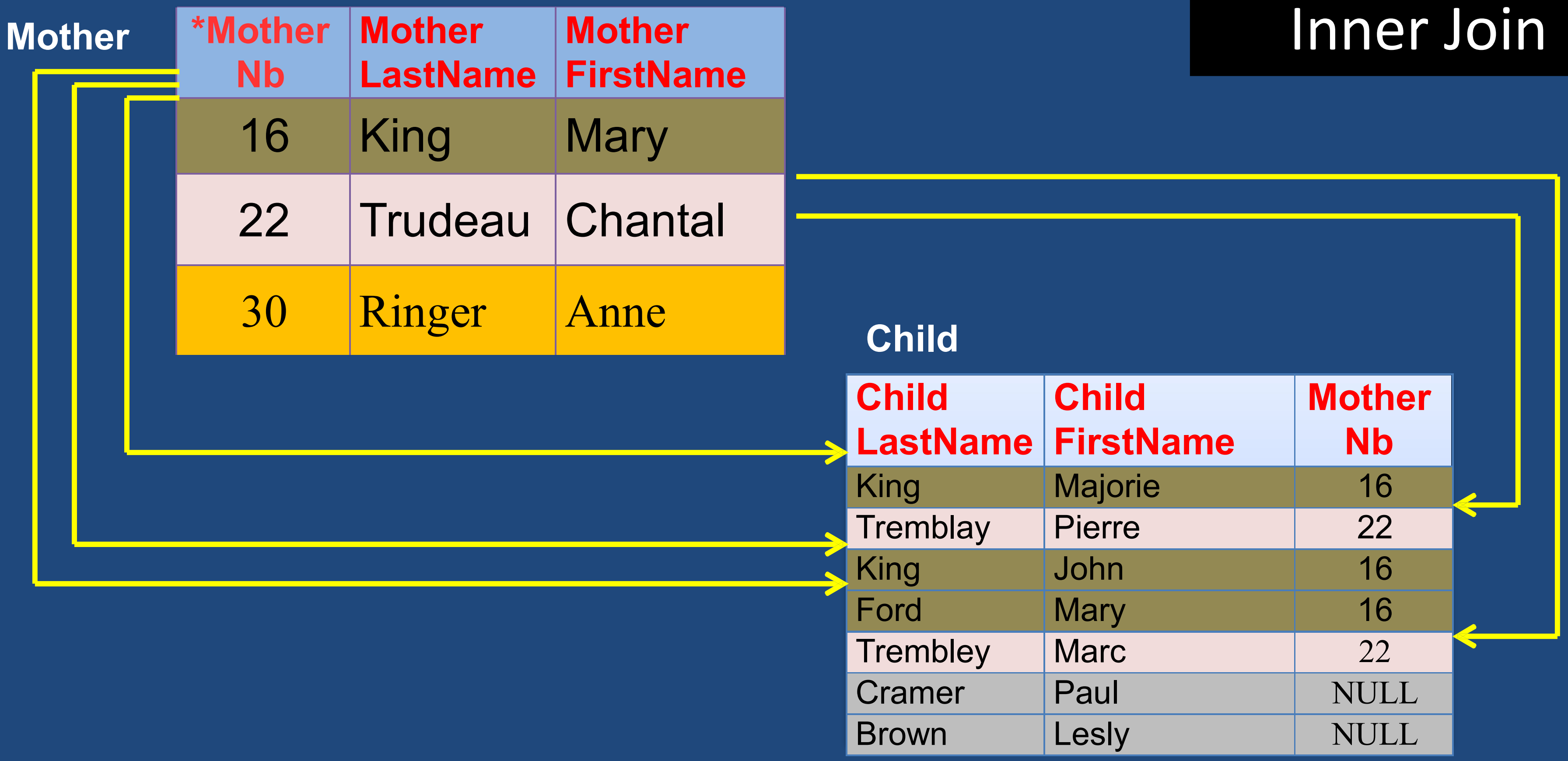
The **Join** clause returns a new table which contains the columns of both tables, and all values that satisfy the column comparison condition.

Inner Joins

(equijoin or natural join)

- **Only returns rows of the Cartesian product of the tables.**
- **The returned rows satisfy the join condition.**

Inner Join



Mother

*Mother Nb	Mother LastName	Mother FirstName
16	King	Mary
22	Trudeau	Chantal
30	Ringer	Anne

Child

Child LastName	Child FirstName	Mother Nb
King	Majorie	16
Tremblay	Pierre	22
King	John	16
Ford	Mary	16
Trembley	Marc	22
Cramer	Paul	NULL
Brown	Lesly	NULL

Equi join of the "Mother" and "Child" tables over the "MotherNb" column

Mother Nb	Mother LastName	Mother FirstName	Child LastName	Child FirstName	Mother Nb
16	King	Mary	King	Majorie	16
22	Trudeau	Chantal	Tremblay	Pierre	22
16	King	Mary	King	John	16
16	King	Mary	Ford	Mary	16
22	Trudeau	Chantal	Trembley	Marc	22

Outer Joins

An outer join on two tables returns

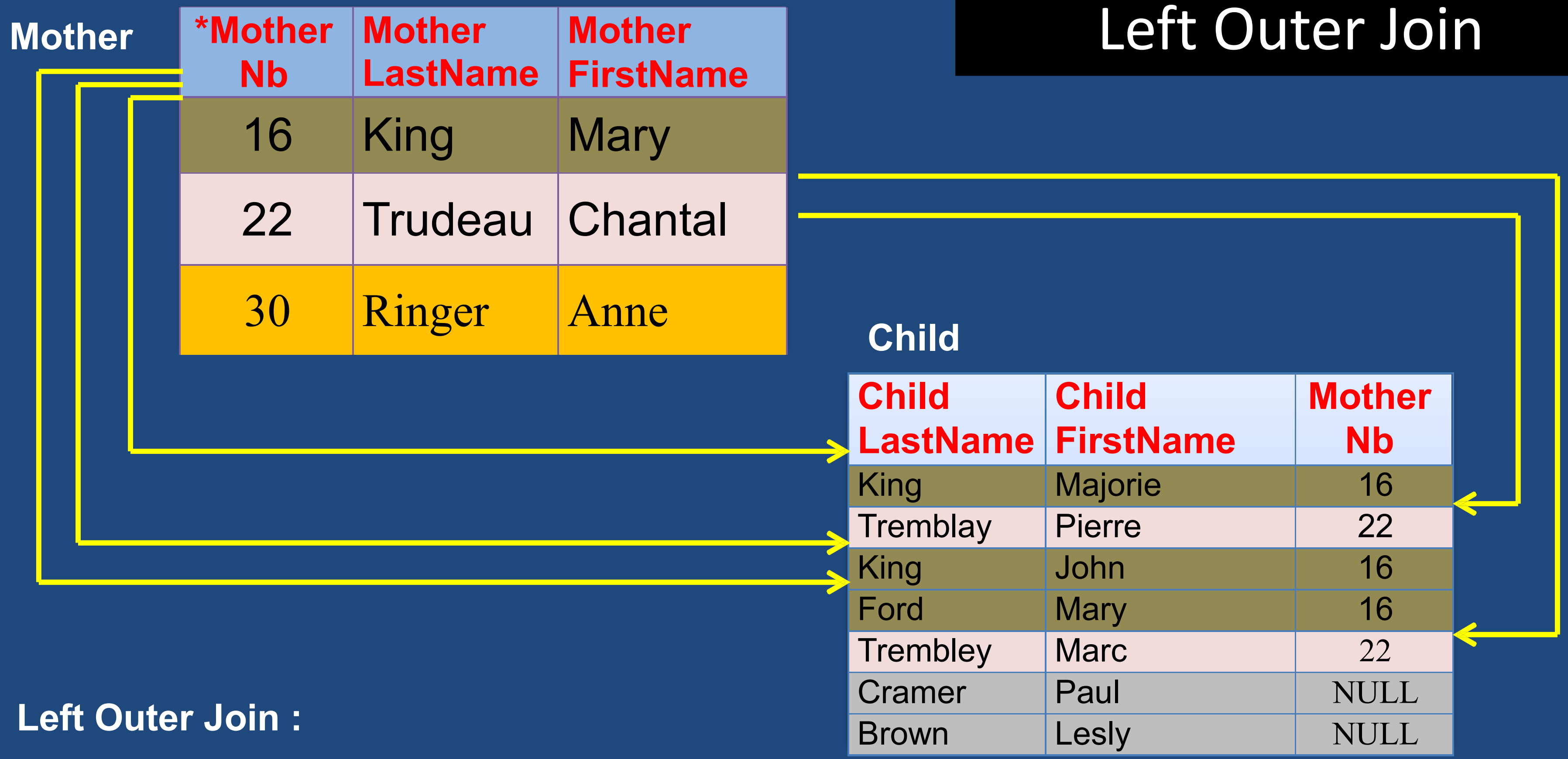
- **the same rows that an inner join would, and**
- **rows from one or both of the tables are excluded.**

Left Outer Join : returns unmatched rows from the left table.

Right Outer Join : returns unmatched rows from the right table.

Full Outer Join : returns unmatched rows from both tables.

Left Outer Join



Mother

*Mother Nb	Mother LastName	Mother FirstName
16	King	Mary
22	Trudeau	Chantal
30	Ringer	Anne

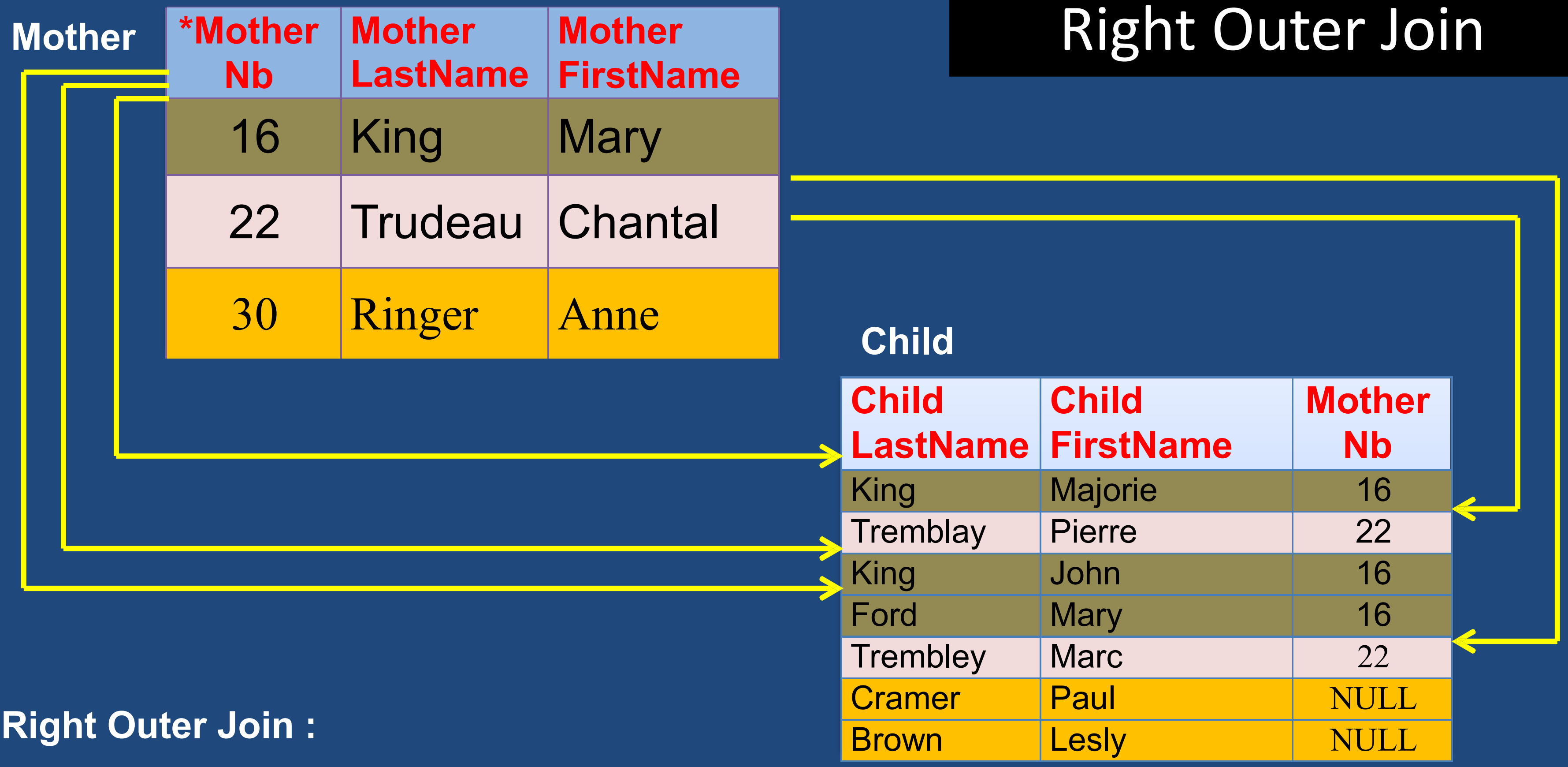
Child

Child LastName	Child FirstName	Mother Nb
King	Majorie	16
Tremblay	Pierre	22
King	John	16
Ford	Mary	16
Trembley	Marc	22
Cramer	Paul	NULL
Brown	Lesly	NULL

Left Outer Join :

Mother Nb	Mother LastName	Mother FirstName	Child LastName	Child FirstName	Mother Nb
16	King	Mary	King	Majorie	16
22	Trudeau	Chantal	Tremblay	Pierre	22
16	King	Mary	King	John	16
16	King	Mary	Ford	Mary	16
22	Trudeau	Chantal	Trembley	Marc	22
30	Ringer	Anne	NULL	NULL	NULL

Right Outer Join



Mother

*Mother Nb	Mother LastName	Mother FirstName
16	King	Mary
22	Trudeau	Chantal
30	Ringer	Anne

Child

Child LastName	Child FirstName	Mother Nb
King	Majorie	16
Tremblay	Pierre	22
King	John	16
Ford	Mary	16
Trembley	Marc	22
Cramer	Paul	NULL
Brown	Lesly	NULL

Right Outer Join :

Mother Nb	Mother LastName	Mother FirstName	Child LastName	Child FirstName	Mother Nb
16	King	Mary	King	Majorie	16
22	Trudeau	Chantal	Tremblay	Pierre	22
16	King	Mary	King	John	16
16	King	Mary	Ford	Mary	16
22	Trudeau	Chantal	Trembley	Marc	22
NULL	NULL	NULL	Cramer	Paul	NULL
NULL	NULL	NULL	Brown	Lesly	NULL

Full Outer Join

Mother

*Mother Nb	Mother LastName	Mother FirstName
16	King	Mary
22	Trudeau	Chantal
30	Ringer	Anne

Child

Child LastName	Child FirstName	Mother Nb
King	Majorie	16
Tremblay	Pierre	22
King	John	16
Ford	Mary	16
Trembley	Marc	22
Cramer	Paul	NULL
Brown	Lesly	NULL

Full Outer Join :

Mother Nb	Mother LastName	Mother FirstName	Child LastName	Child FirstName	Mother Nb
16	King	Mary	King	Majorie	16
22	Trudeau	Chantal	Tremblay	Pierre	22
16	King	Mary	King	John	16
16	King	Mary	Ford	Mary	16
22	Trudeau	Chantal	Trembley	Marc	22
30	Ringer	Anne	NULL	NULL	NULL
NULL	NULL	NULL	Cramer	Paul	NULL
NULL	NULL	NULL	Brown	Lesly	NULL

Inner Joins : 1st syntax

```
SELECT *
FROM mother, child
WHERE mother.MotherNb = child.MotherNb
```

Sorted by mothers numbers :

```
SELECT *
FROM mother, child
WHERE mother.MotherNb = child.MotherNb
ORDER BY mother.MotherNb
```

Using aliases :

```
SELECT *
FROM mother m , child c
WHERE m.MotherNb = c.MotherNb
ORDER BY m.MotherNb
```

Return a table containing the mother numbers, first and last names of the mothers, with the first and last names of their children:

```
SELECT mother.MotherNb, mother.MotherFirstName,
       mother.MotherLastName,  child.ChildFirstName,
       child.ChildLastName
FROM mother, child
WHERE mother.MotherNb = child.MotherNb
ORDER BY mother.MotherNb
```

Using aliases :

```
SELECT      m.MotherNb, m.MotherFirstName,
            m.MotherLastName,  c.ChildFirstName,
            c.ChildLastName
FROM   mother m , child c
WHERE   m.MotherNb = c.MotherNb  ORDER BY m.MotherNb
```

Inner Joins : 2nd Syntax

```
SELECT *
FROM mother INNER JOIN child ON mother.MotherNb = child.MotherNb
```

Sorted by mothers numbers:

```
SELECT *
FROM mother INNER JOIN child ON mother.MotherNb = child.MotherNb
ORDER BY mother.MotherNb
```

Using aliases :

```
SELECT *
FROM mother m INNER JOIN child c ON m.MotherNb = c.MotherNb
ORDER BY m.MotherNb
```

Return a table containing the numbers, first and last names of the mothers, with the first and last names of their children:

```
SELECT   m.MotherNb, m.MotherFirstName, m.MotherLastName,
         c.ChildFirstName, c.ChildLastName
FROM     mother m INNER JOIN child c ON m.MotherNb = c.MotherNb
ORDER BY   m.MotherNb
```

Writing the INNER syntax is optional, as the JOIN clause implicitly assumes INNER by default.

We can implicitly define an inner join like this :

```
SELECT  m.MotherNb, m.MotherFirstName, m.MotherLastName,
        c.ChildFirstName, c.ChildLastName
FROM    mother m JOIN child c ON m.MotherNb = c.MotherNb
ORDER BY   m.MotherNb
```

Return a table containing the first and last name of each employee with the name of the publisher for whom they work. Sort the data by employee name in ascending order.

Employee	Publisher
Anabela Domingues	Binnet & Hardley
Ann Devon	Scootney Books
Annette Roulet	Lucerne Publishing
Aria Cruz	Algodata Infosystems
Carine Schmitt	Algodata Infosystems
Carlos Hernadez	Lucerne Publishing
Daniel Tonini	Binnet & Hardley
Diego Roel	Algodata Infosystems
Elizabeth Lincoln	Binnet & Hardley
Francisco Chang	Scootney Books
Gary Thomas	New Moon Books
Helen Bennett	Binnet & Hardley
Helvetius Nagy	Lucerne Publishing
Howard Snyder	New Moon Books
Janine Labrune	GGG&G
Karin Josephs	New Moon Books
Karla Jablonski	Lucerne Publishing
Laurence Lebihan	New Moon Books
…	…

Return a table containing the first and last name of each employee with the name of the publisher for whom they work.

```
SELECT    e.fname + ' ' + e.lname AS 'Employee',
          p.pub_name AS 'Publisher'
FROM employee e JOIN publishers p ON e.pub_id = p.pub_id
ORDER BY e.fname
```

Return a table containing each store name, with their publisher and state they are located in.

Publisher	Store	State
Algodata Infosystems	Barnum's	CA
Algodata Infosystems	News & Brews	CA
Algodata Infosystems	Fricative Bookshop	CA

```
SELECT  p.pub_name AS Publisher,
        s.stor_name AS Store,
        p.state AS State
FROM publishers p  JOIN stores s ON p.state=s.state
```

Return the table of all books that were sold in Washington. Show the name of each book, the city they were sold in, and the quantity of that book sold in that city.

Book	Quantity	City	State
The Busy Executive's Database Guide	5	Seattle	WA
Is Anger the Enemy?	3	Seattle	WA
Is Anger the Enemy?	20	Remulade	WA
The Gourmet Microwave	25	Remulade	WA
Computer Phobic AND Non-Phobic Individuals: Behavior Variations	20	Remulade	WA
Life Without Fear	25	Remulade	WA
Prolonged Data Deprivation: Four Case Studies	15	Remulade	WA
Emotional Security: A New Algorithm	25	Remulade	WA

```
SELECT        T.title AS Book,          Sum(S.qty) AS Quantity
              ST.city AS City,          ST.state AS state
FROM titles T INNER JOIN sales S        ON T.title_id = S.title_id
              INNER  JOIN stores ST     ON S.stor_id = ST.stor_id
WHERE ST.state = 'WA'
GROUP BY T.title_id, T.title, ST.city, ST.state
```

Left Inner Join

Return a table containing the numbers, first and last names of the mothers, with the first and last names of their children. <u>Also display mothers with no children.</u>

Left Inner Join :

Mother Nb	MotherLast Name	MotherFirst Name	Child LastName	ChildFirst Name	Mother Nb
16	King	Mary	King	Majorie	16
22	Trudeau	Chantal	Tremblay	Pierre	22
16	King	Mary	King	John	16
16	King	Mary	Ford	Mary	16
22	Trudeau	Chantal	Trembley	Marc	22
30	Ringer	Anne	NULL	NULL	NULL

```
SELECT      m.MotherNb, m.MotherFirstName, m.MotherLastName,
            c.ChildFirstName, c.ChildLastName
FROM mother m LEFT JOIN child c ON m.MotherNb = c.MotherNb
```

Right Outer Join

Return a table containing the numbers, first and last names of the mothers, with the first and last names of their children. <u>Also display children with no mothers.</u>

Right Outer Join :

Mother Nb	Mother LastName	Mother FirstName	Child LastName	Child FirstName	Mother Nb
16	King	Mary	King	Majorie	16
22	Trudeau	Chantal	Tremblay	Pierre	22
16	King	Mary	King	John	16
16	King	Mary	Ford	Mary	16
22	Trudeau	Chantal	Trembley	Marc	22
NULL	NULL	NULL	Cramer	Paul	NULL
NULL	NULL	NULL	Brown	Lesly	NULL

```
SELECT     m.MotherNb, m.MotherFirstName, m.MotherLastName,
           c.ChildFirstName, c.ChildLastName
FROM mother m RIGHT JOIN child c ON m.MotherNb = c.MotherNb
```

Full Outer Join

Return a table containing the numbers, first and last names of the mothers, with the first and last names of their children. Also display the children with no mothers and the mothers with no children.

Full outer join:

Mother Nb	Mother LastName	Mother FirstName	Child LastName	Child FirstName	Mother Nb
16	King	Mary	King	Majorie	16
22	Trudeau	Chantal	Tremblay	Pierre	22
16	King	Mary	King	John	16
16	King	Mary	Ford	Mary	16
22	Trudeau	Chantal	Trembley	Marc	22
30	Ringer	Anne	NULL	NULL	NULL
NULL	NULL	NULL	Cramer	Paul	NULL
NULL	NULL	NULL	Brown	Lesly	NULL

```
SELECT    m.MotherNb, m.MotherFirstName, m.MotherLastName,
          c.ChildFirstName, c.ChildLastName
FROM mere m FULL JOIN child c ON m.MotherNb = c.MotherNb
```

Write the SQL queries which returns the following lists:

1. The list of pairs (author, publisher) living in the same city.
2. The list of pairs (author, publisher) residing in the same city, including authors who do not meet this criterion.
3. The list of pairs (author, publisher) residing in the same city, including publishers who do not meet this criterion.
4. The list of pairs (author, publisher) living in the same city, including authors and publishers who do not meet this criterion.

Self-Join

A **self-join** is a join of a table with itself. The expression used to denote a self join always requires the use of an alias to refer to tables.

Example:

Return a table containing authors with the same name (au_lname) and different ids

```
SELECT  a.au_id, a.au_lname, a.au_fname, b.au_id, b.au_lname, b.au_fname
FROM    authors a JOIN authors b ON a.au_lname=b.au_lname
WHERE   a.au_id!=b.au_id
```

Self-Join

Return a table containing the list of pairs (author, author) living in the same city.

```
SELECT a.au_id, b.au_id, a.au_lname, a.state, a.city, b.au_lname, b.state,b.city
FROM authors a JOIN authors b ON a.city=b.city
WHERE a.au_id!=b.au_id
ORDER BY a.au_id
```

5. Return a table containing each advance made to each author. The table should also show the book the advance way paid for and the publisher paying the advance.

Book	Author	Publisher	Advance
The Busy Executive's Database Guide	Green	Algodata Infosystems	5000.00
The Busy Executive's Database Guide	Bennet	Algodata Infosystems	5000.00
Cooking with Computers: Surreptitious Balance Sheets	O'Leary	Algodata Infosystems	5000.00
Cooking with Computers: Surreptitious Balance Sheets	MacFeather	Algodata Infosystems	5000.00
Straight Talk About Computers	Straight	Algodata Infosystems	5000.00
But Is It User Friendly?	Carson	Algodata Infosystems	7000.00
Secrets of Silicon Valley	Dull	Algodata Infosystems	8000.00
Secrets of Silicon Valley	Hunter	Algodata Infosystems	8000.00
Net Etiquette	Locksley	Algodata Infosystems	NULL
Silicon Valley Gastronomic Treats	del Castillo	Binnet & Hardley	0.00
The Gourmet Microwave	DeFrance	Binnet & Hardley	15000.00
The Gourmet Microwave	Ringer	Binnet & Hardley	15000.00
Computer Phobic AND Non-Phobic Individuals: Behavior Variations	MacFeather	Binnet & Hardley	7000.00
Computer Phobic AND Non-Phobic Individuals: Behavior Variations	Karsen	Binnet & Hardley	7000.00
Onions, Leeks, and Garlic: Cooking Secrets of the Mediterranean	Panteley	Binnet & Hardley	7000.00
Fifty Years in Buckingham Palace Kitchens	Blotchet-Halls	Binnet & Hardley	4000.00
Sushi, Anyone?	O'Leary	Binnet & Hardley	8000.00
Sushi, Anyone?	Gringlesby	Binnet & Hardley	8000.00
Sushi, Anyone?	Yokomoto	Binnet & Hardley	8000.00
You Can Combat Computer Stress!	Green	New Moon Books	10125.00
Is Anger the Enemy?	Ringer	New Moon Books	2275.00
Is Anger the Enemy?	Ringer	New Moon Books	2275.00
Life Without Fear	Ringer	New Moon Books	6000.00
Prolonged Data Deprivation: Four Case Studies	White	New Moon Books	2000.00
Emotional Security: A New Algorithm	Locksley	New Moon Books	4000.00