

SQL over BigData

SQL Basics

Goals

- **extract** data

Goals

- **extract** data
- **filter and order** data

Goals

- **extract** data
- **filter and order** data
- **aggregate** data

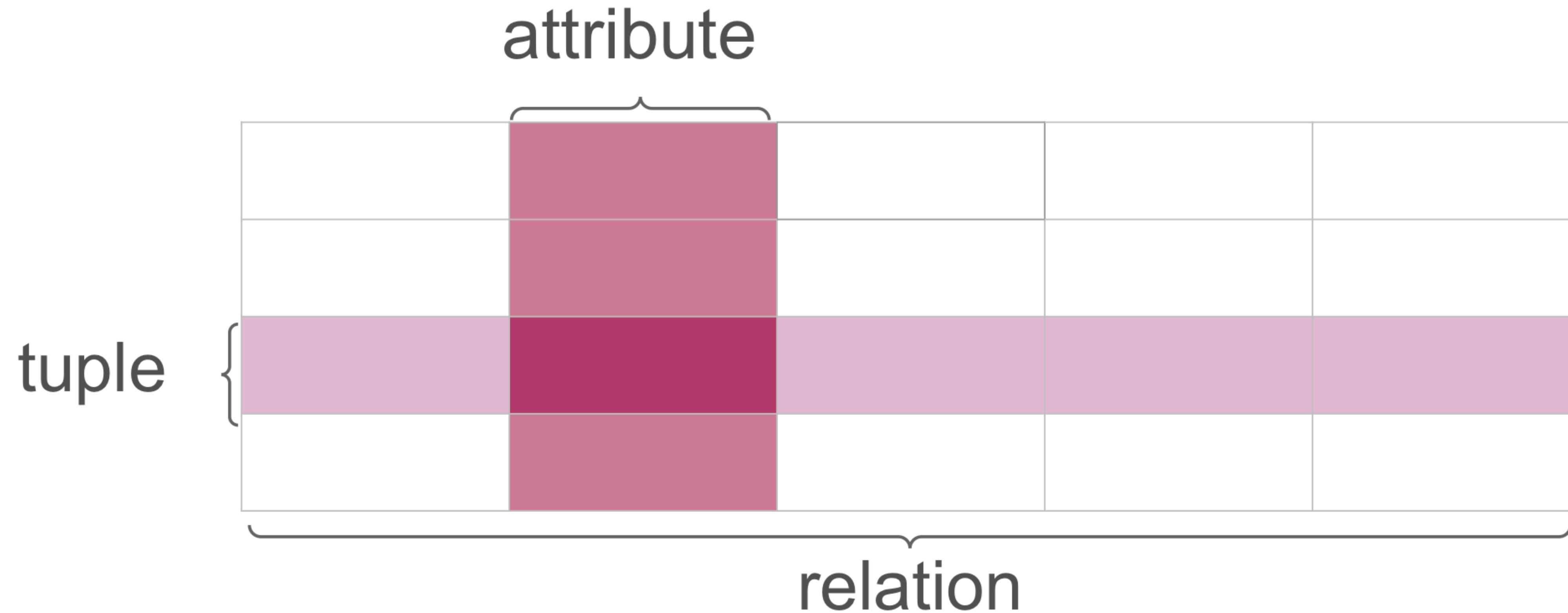
Goals

- **extract** data
- **filter and order** data
- **aggregate** data
- **join** data

Goals

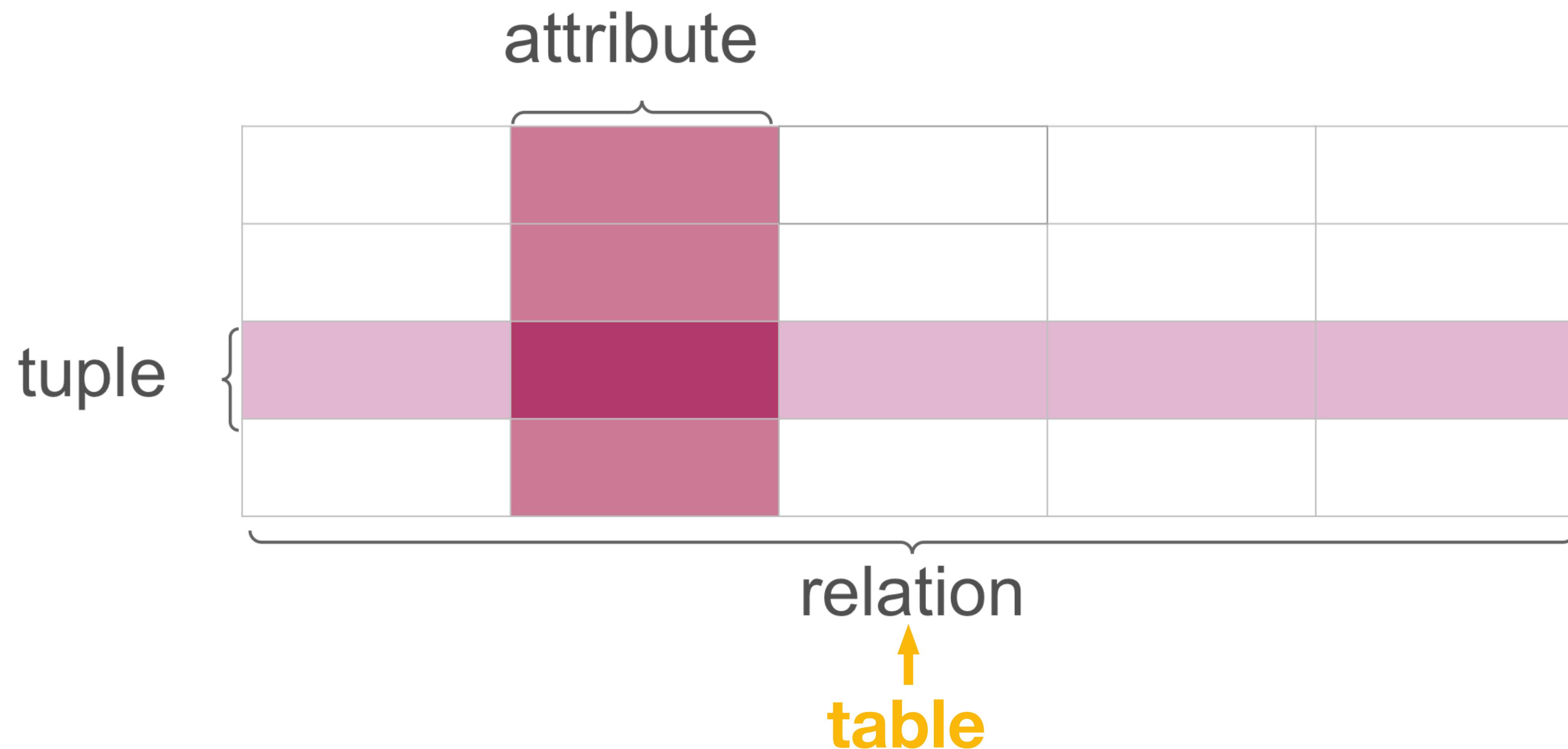
- **extract** data
- **filter and order** data
- **aggregate** data
- **join** data
- **use** subqueries

Relational Algebra



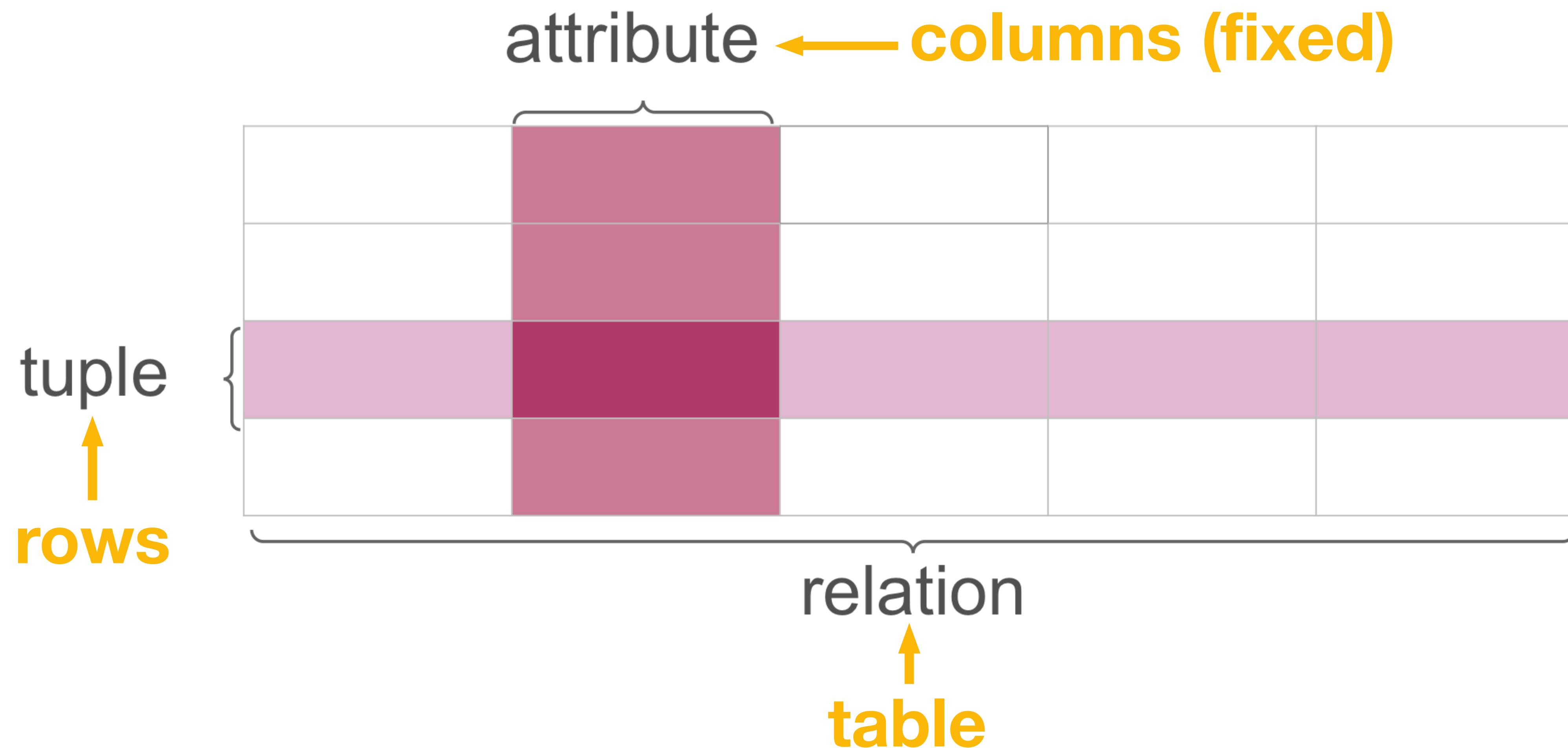
see: https://en.wikipedia.org/wiki/Relational_algebra

Relational Algebra



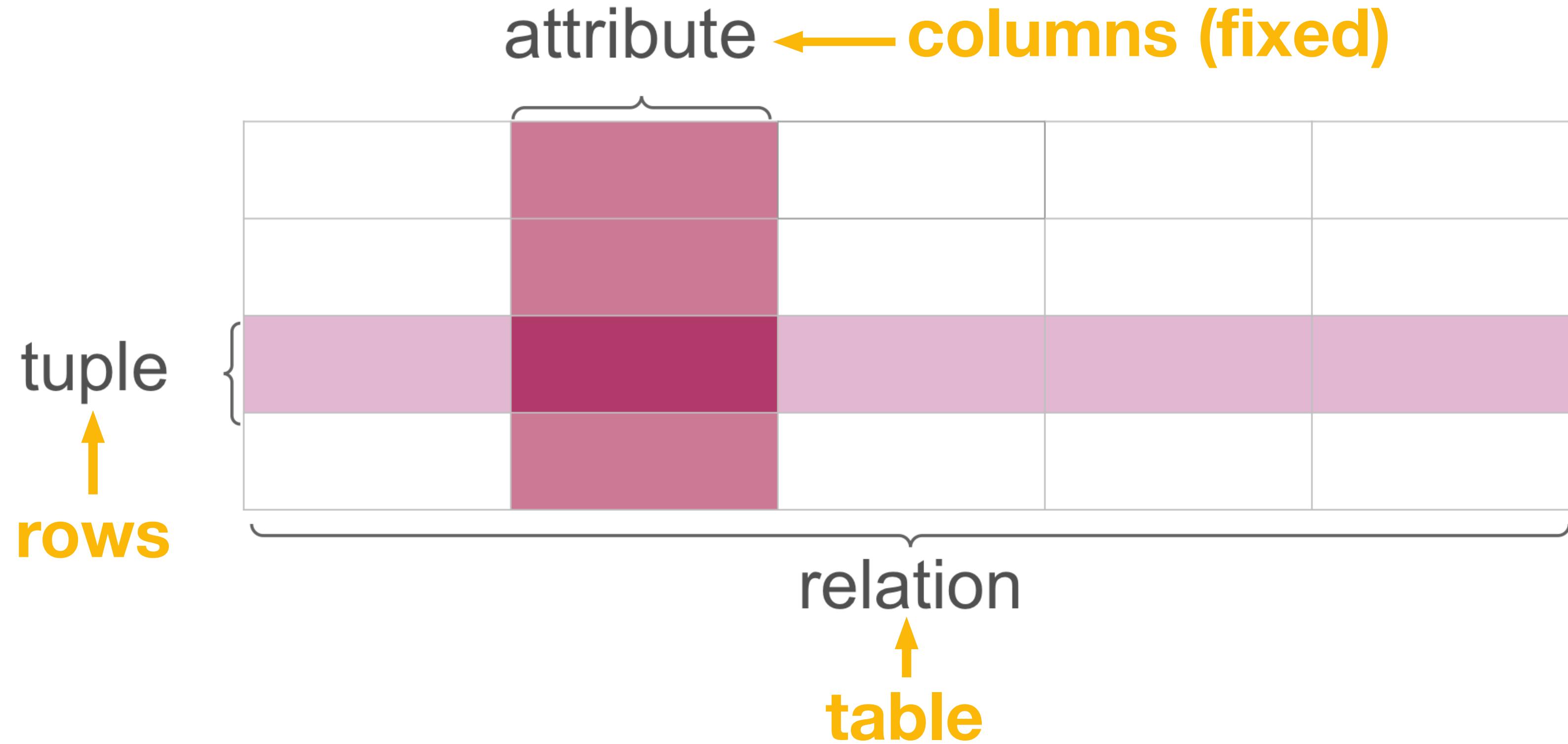
see: https://en.wikipedia.org/wiki/Relational_algebra

Relational Algebra



see: https://en.wikipedia.org/wiki/Relational_algebra

Relational Algebra



Operations:

1. Projection
2. Selection
3. Rename
4. Joins

see: https://en.wikipedia.org/wiki/Relational_algebra

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Andres	Obre Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. De la Constitucion 2222	Mexico D.F.	05021	Mexico
3	Antonio Moreno Taqueria	Antonio Moreno	Mataderos 2312	Mexico D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Honover Sq.	London	WA1 1DP	UK
5	Berglunds snabbkop	Christina Berglund	Berguvsvagen 8	Lulea	S-958 22	Sweeden

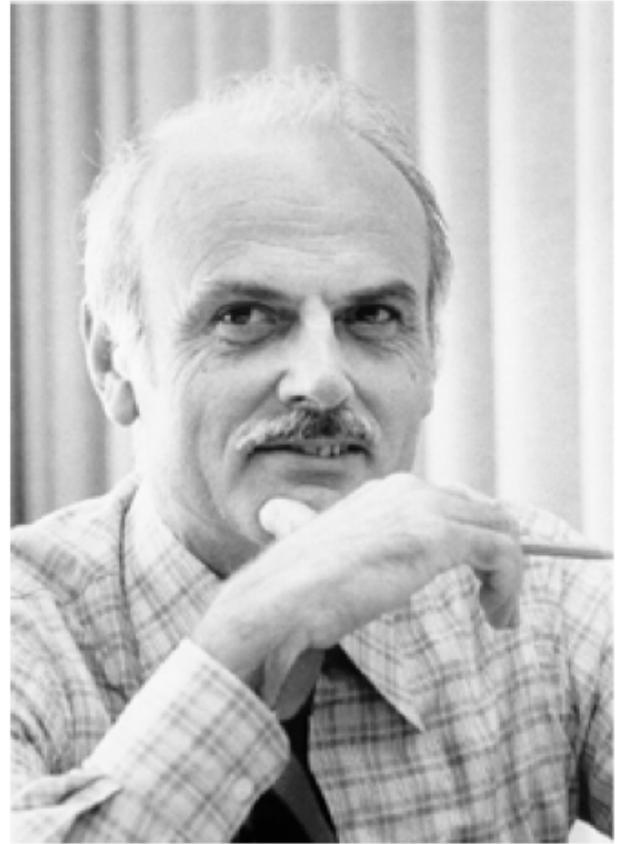
see: https://www.w3schools.com/sql/trysql.asp?filename=trysql_select_all

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Av. de la Constitución 2222	Mexico D.F.	05021	Mexico
3	Antonio Moreno Taqueria	Antonio Moreno	Mataderos 2312	Mexico D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	Lond	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Lulea	S-958 22	Sweden

SELECT CustomerName, City FROM Customers;

CustomerName	City
Alfreds Futterkiste	Berlin
Ana Trujillo Emparedados y helados	Mexico D.F.
Antonio Moreno Taqueria	Mexico D.F.
Around the Horn	London
Berglunds snabbkop	Lulea

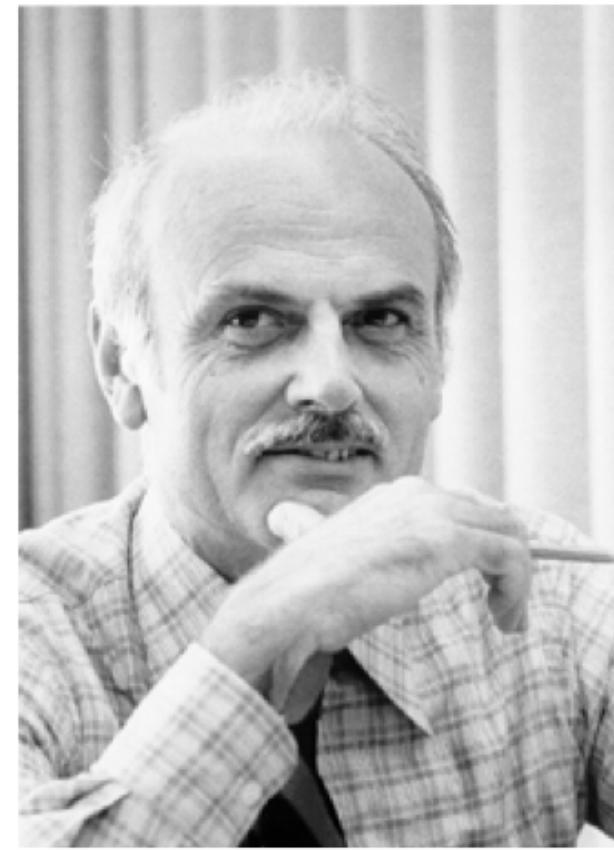
Selection (or restriction) operations



Edgar F. Codd

Codd, E.F. (June 1970). "A Relational Model of Data for Large Shared Data Banks". Communications of the ACM. 13 (6): 377–387. doi:10.1145/362384.362685.

Selection (or restriction) operations



Edgar F. Codd

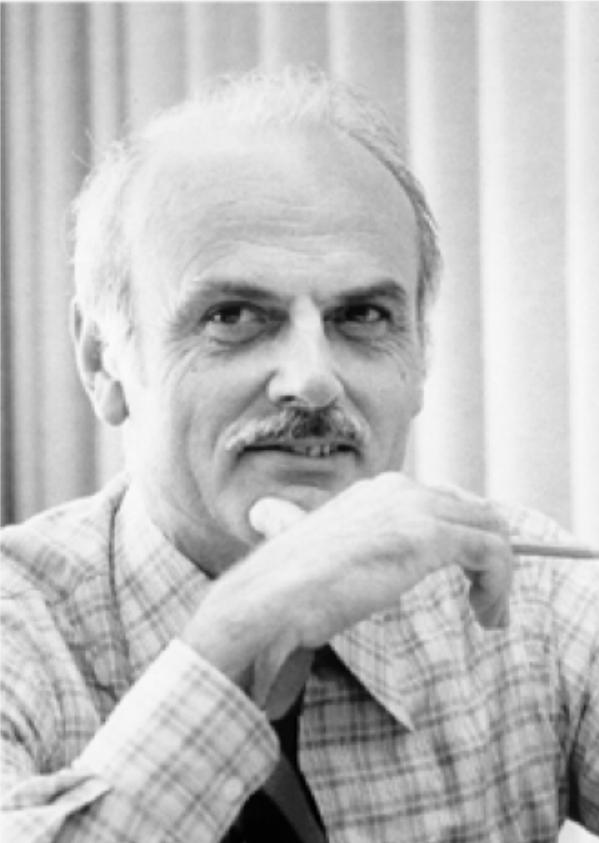
CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Andres	Obre Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. De la Constitucion 2222	Mexico D.F.	05021	Mexico
3	Antonio Moreno Taqueria	Antonio Moreno	Mataderos 2312	Mexico D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Honover Sq.	London	WA1 1DP	UK
5	Berglunds snabbkop	Christina Berglund	Berguvsvage n 8	Lulea	S-958 22	Sweeden

SELECT *
FROM Customers **WHERE** Country='Mexico';

Codd, E.F. (June 1970). "A Relational Model of Data for Large Shared Data Banks".

Communications of the ACM. 13 (6): 377–387. doi:10.1145/362384.362685.

Selection (or restriction) operations



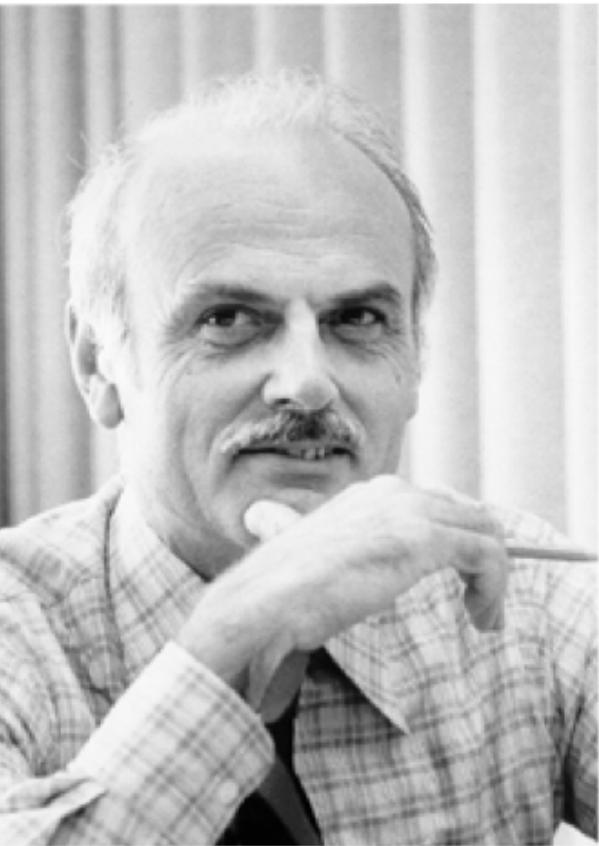
CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
3	Antonio Moreno Taqueria	Antonio Moreno	Mataderos 2312	Mexico D.F.	05023	Mexico

Edgar F. Codd

```
SELECT *
FROM Customers
WHERE Country='Mexico'
AND PostalCode="05023";
```

Codd, E.F. (June 1970). "A Relational Model of Data for Large Shared Data Banks". Communications of the ACM. 13 (6): 377–387. doi:10.1145/362384.362685.

Selection (or restriction) operations



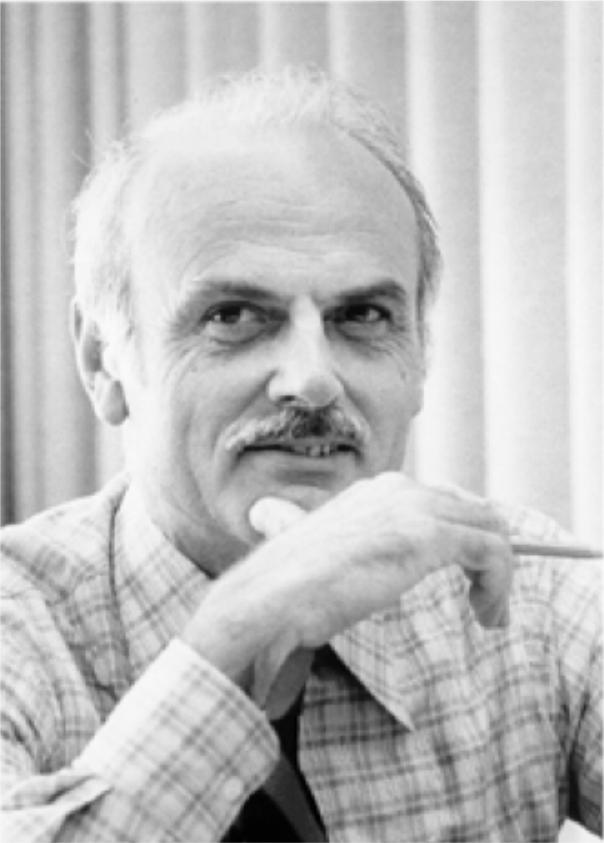
Edgar F. Codd

City	PostalCode	Country
Buenos Aires	1010	Argentina
Buenos Aires	1010	Argentina
Buenos Aires	1010	Argentina
Salzburg	5020	Austria
Bruxelles	B-1180	Belgium
Charleroi	B-6000	Belgium
Sao Paulo	05487-020	Brazil
Rio de Janeiro	05454-876	Brazil

```
SELECT *
FROM Customers
ORDER BY Country;
```

Codd, E.F. (June 1970). "A Relational Model of Data for Large Shared Data Banks". Communications of the ACM. 13 (6): 377–387. doi:10.1145/362384.362685.

Selection (or restriction) operations



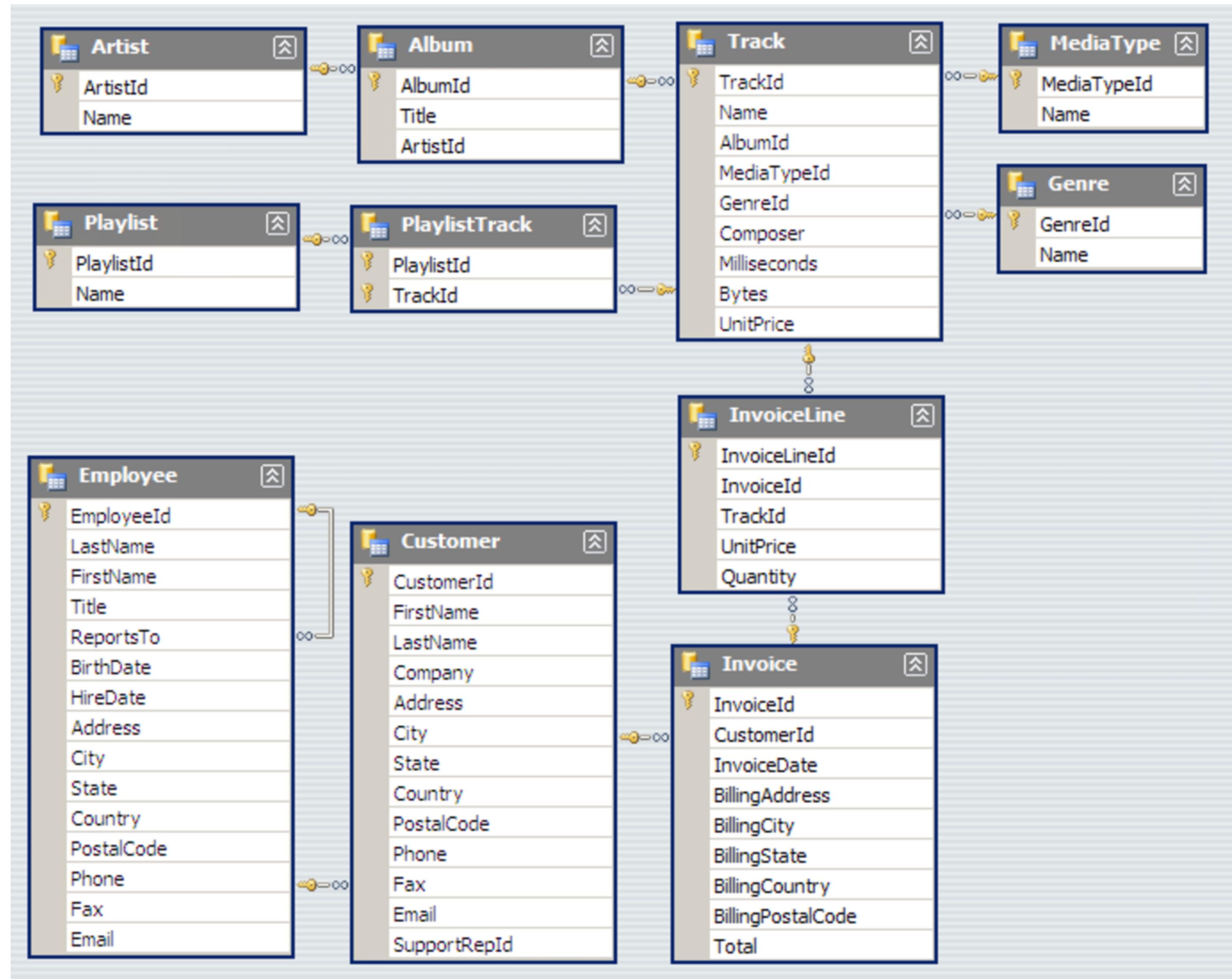
Edgar F. Codd

City	PostalCode	Country
Caracas	1081	Venezuela
San Cristobal	5022	Venezuela
Barquisimeto	3508	Venezuela
I. de Margarita	4980	Venezuela
Eugene	97403	USA
Elgin	97827	USA

```
SELECT *
FROM Customers
ORDER BY Country DESC;
```

Codd, E.F. (June 1970). "A Relational Model of Data for Large Shared Data Banks". Communications of the ACM. 13 (6): 377–387. doi:10.1145/362384.362685.

Selection (or restriction) operations



Customers

CustomerID	CustomerName	ContactName	Country
1	Alfreds Futterkiste	Maria Andres	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Mexico
3	Antonio Moreno Taqueria	Antonio Moreno	Mexico

Orders

OrderID	CustomerID	OrderDate
10308	2	1996-09-18
10309	37	1996-09-19
10310	77	1996-09-20

Customers

CustomerID	CustomerName	ContactName	Country
1	Alfreds Futterkiste	Maria Andres	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Mexico
3	Antonio Taqueria Moreno	Antonio Moreno	Mexico

Orders

OrderID	CustomerID	OrderDate
10308	2	1996-09-18
10309	37	1996-09-19
10310	77	1996-09-20

see: https://en.wikipedia.org/wiki/Database_normalization

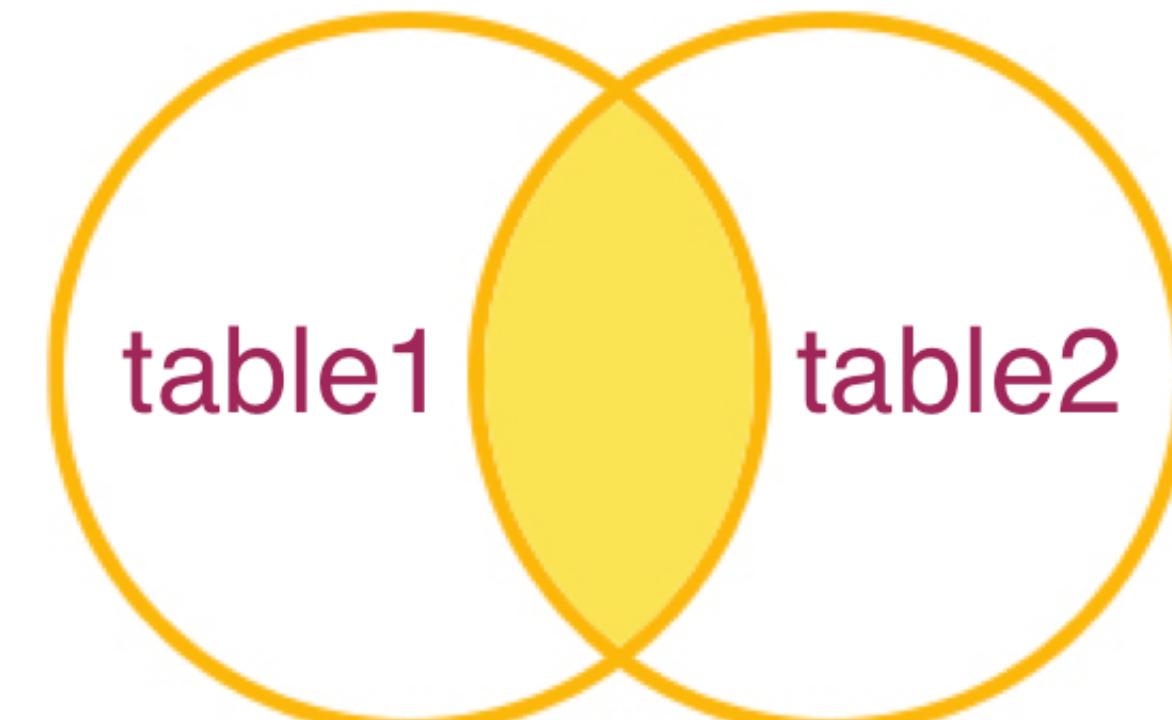
OrderID	CustomerName	OrderDate
10308	Ana Trujillo Emparedados y helados	9/18/1996
10365	Antonio Moreno Taqueria	11/27/1996
10383	Around the Horn	12/16/1996
10355	Around the Horn	11/15/1996
10278	Berglunds snabbkop	8/12/1996

```
SELECT Orders.OrderID, Customers.CustomerName, Orders.OrderDate  
FROM Orders JOIN Customers  
ON Orders.CustomerID=Customers.CustomerID;
```

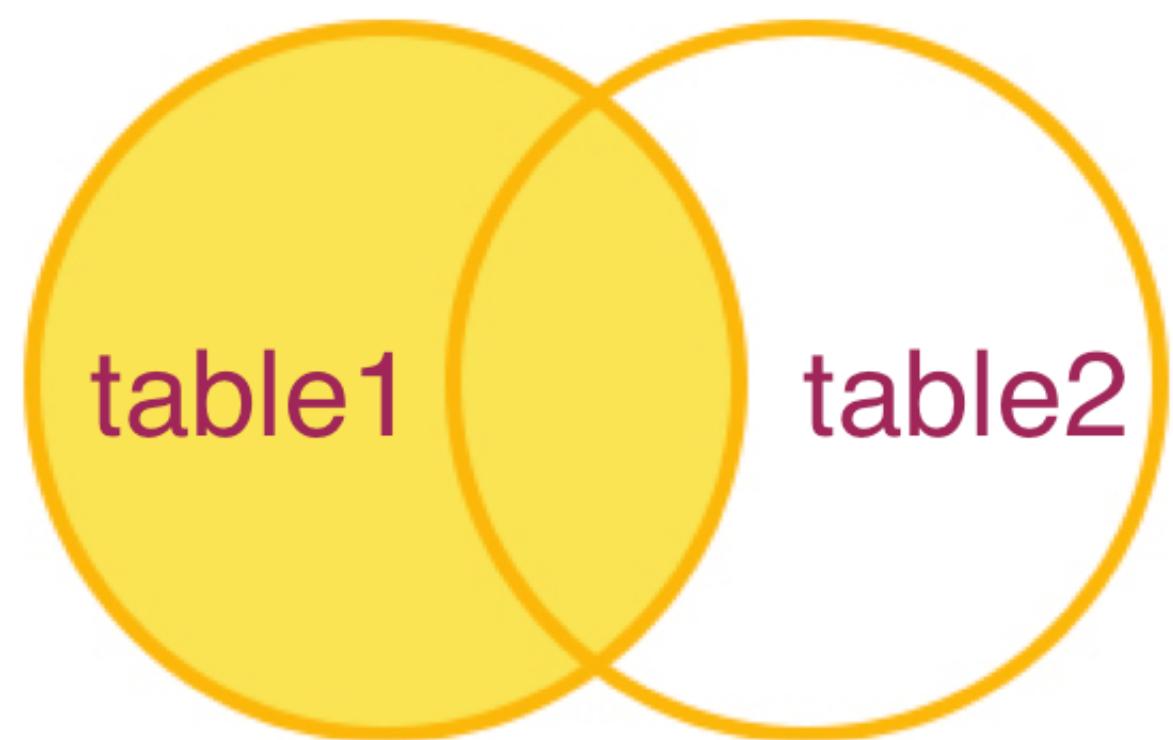
OrderID	CustomerName	OrderDate
10308	Ana Trujillo Emparedados y helados	9/18/1996
10365	Antonio Moreno Taqueria	11/27/1996
10383	Around the Horn	12/16/1996
10355	Around the Horn	11/15/1996
10278	Berglunds snabbkop	8/12/1996

```
SELECT Orders.OrderID, Customers.CustomerName, Orders.OrderDate  
FROM Orders JOIN Customers  
ON Orders.CustomerID=Customers.CustomerID;
```

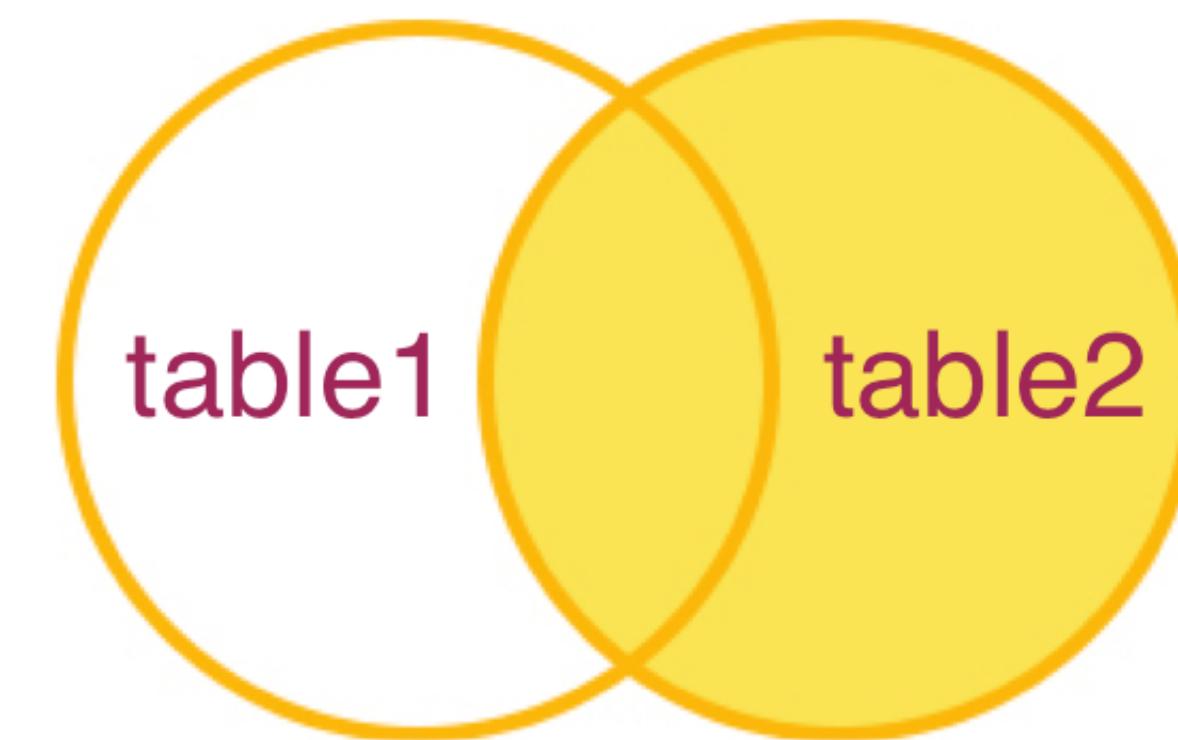
INNER JOIN



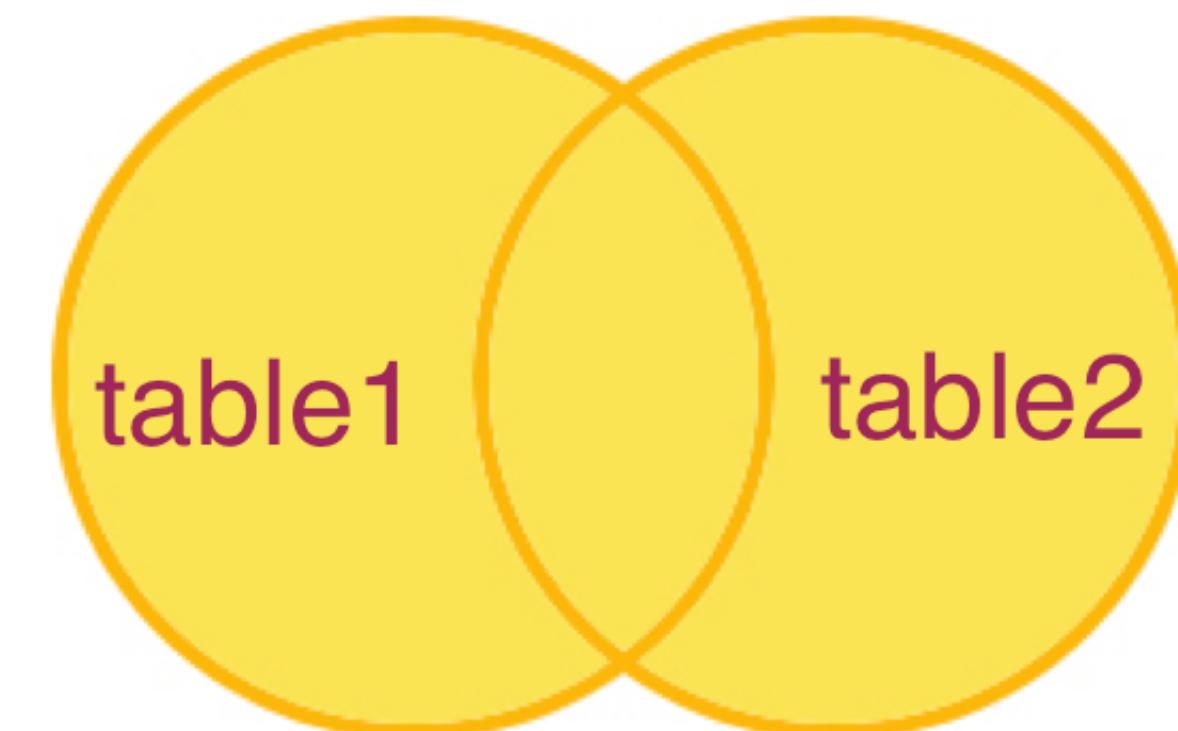
LEFT JOIN



RIGHT JOIN



SQL Joins



FULL OUTER JOIN

INNER JOIN

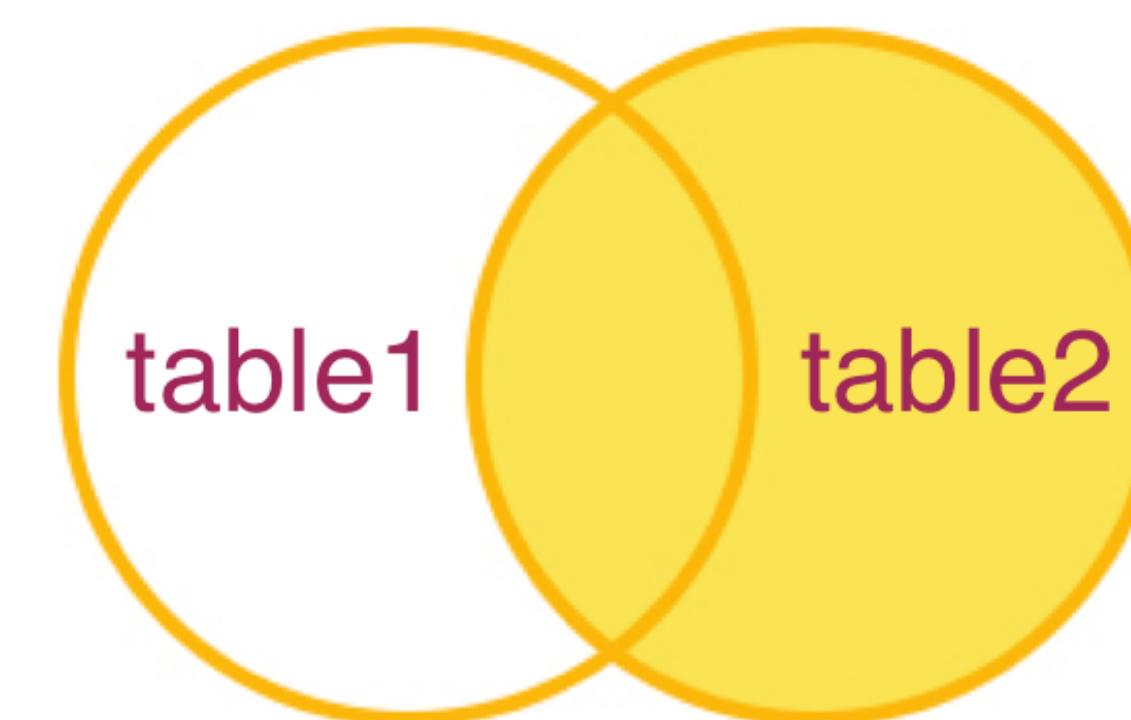
a	b
3	3
4	4

LEFT JOIN

a	b
1	null
2	null
3	3
4	4

A	B
1	3
2	4
3	5
4	6

RIGHT JOIN



a	b
1	null
2	null
3	3
4	4
null	6
null	5

FULL OUTER JOIN

Data Aggregation

```
SELECT column_name, aggregate_function(column_name)  
FROM table_name  
GROUP BY column_name
```

Data Aggregation

```
SELECT Country, COUNT(*)  
FROM Customers  
GROUP BY Country
```

```
SELECT Country, State, COUNT(*)  
FROM Customers  
GROUP BY Country, State
```

Data Aggregation

```
SELECT OrderDate, MIN(OrderID), MAX(OrderID)  
FROM Orders  
GROUP BY OrderDate;
```

OrderDate	MIN(OrderID)	MAX(OrderID)
1996-07-04	10248	10248
1996-07-05	10249	10249
1996-07-08	10250	10251

Number of records: 160

Where vs Having

```
SELECT column_name,  
aggregate_function(column_name) FROM table_name  
    WHERE column_name operator value  
GROUP BY column_name  
    HAVING aggregate_function(column_name) operator value;
```

Where vs Having

```
SELECT column_name,  
aggregate_function(column_name) FROM table_name  
  WHERE column_name operator value  
GROUP BY column_name  
  HAVING aggregate_function(column_name) operator value;
```

→ Waaaaay more efficient =)

Where vs Having

```
SELECT column_name,  
aggregate_function(column_name) FROM table_name  
  WHERE column_name operator value  
GROUP BY column_name  
  — HAVING aggregate_function(column_name) operator value;  
  
→ Huge resource consumption
```

Where vs Having

```
SELECT Country  
FROM Customers  
GROUP BY Country  
HAVING COUNT(*) > 10;
```

Country
France
Germany
USA

Number of records: 160

SQL Subqueries

```
SELECT *
FROM Customers
WHERE Country IN (
    SELECT Country
    FROM Customers
    GROUP BY Country HAVING COUNT(*) > 10
);
```

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Andres	Obre Str. 57	Berlin	12209	Germany
6	Brauer See Delikatessen	Hanna Moos	Forstrerstr. 57	Mannheim	68306	Germany
7	Blondel pere et fils	Frederique Citeaux	24, place Kleber	Strasbourg	67000	France

Summary

Summary

You now know:

- how to **extract, filter and order** data

Summary

You now know:

- how to **extract, filter and order** data
- how to **aggregate** data and **use** subqueries

Summary

You now know:

- how to **extract, filter and order** data
- how to **aggregate** data and **use** subqueries
- how to (inner / left / right / full outer) **join** data

Summary

You now know:

- how to **extract, filter and order** data
- how to **aggregate** data and **use** subqueries
- how to (inner / left / right / full outer) **join** data

see: <https://www.w3schools.com/sql/default.asp>

W3Schools SQL Quiz

Result:

25 of 25

100%

Perfect!!!

Time Spent

4:43

Check your answers

Try again