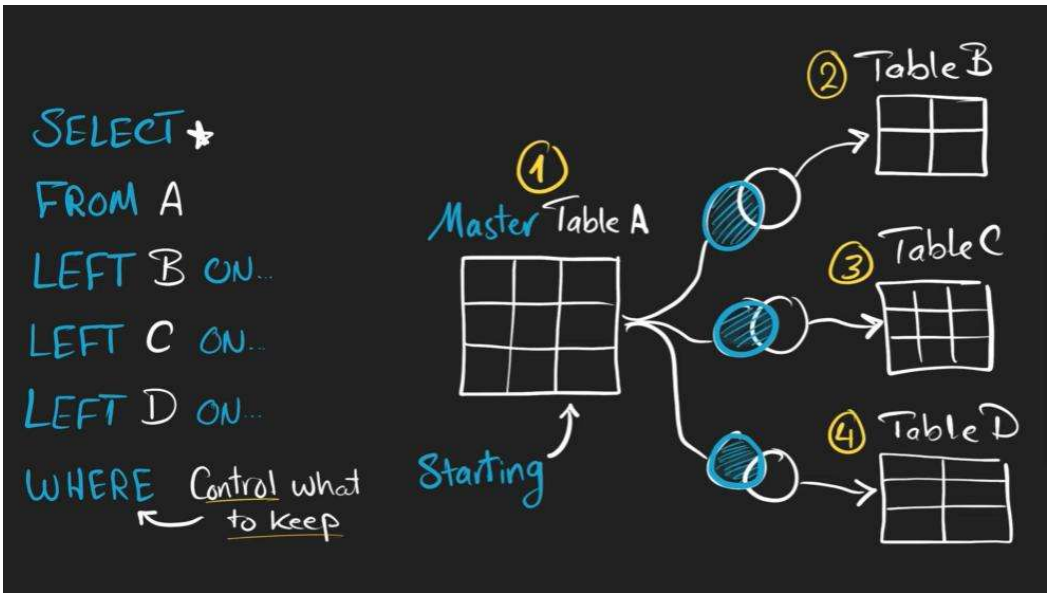
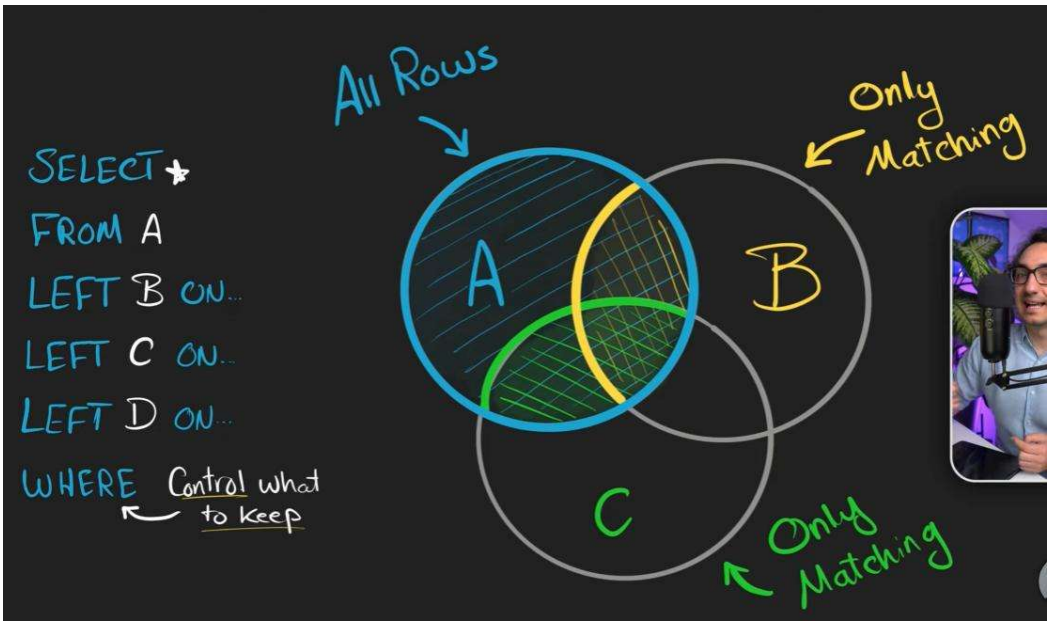
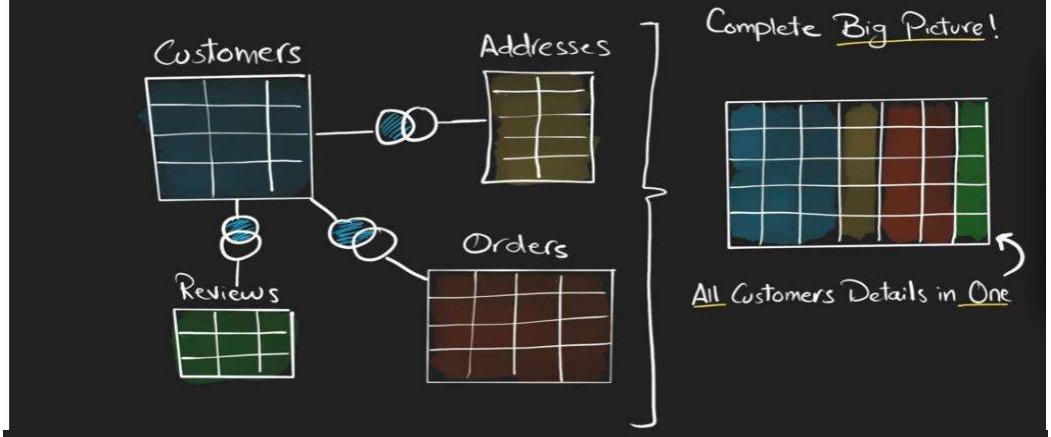
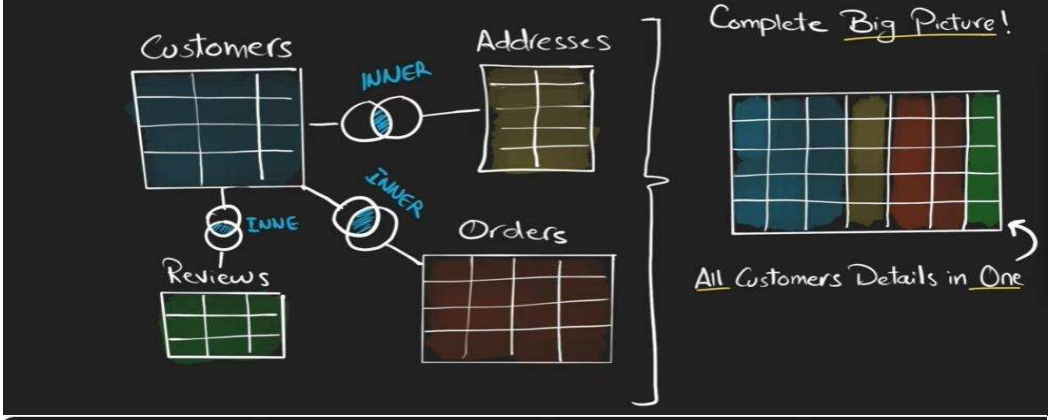

SELECT *
FROM A
LEFT B ON...
LEFT C ON...
LEFT D ON...
WHERE Control what to keep
1
Master Table A
Starting
2 Table B
3 Table C
4 Table D


SELECT *
FROM A
LEFT B ON...
LEFT C ON...
LEFT D ON...
WHERE Control what to keep
All Rows
Only Matching
A
B
C
Only Matching

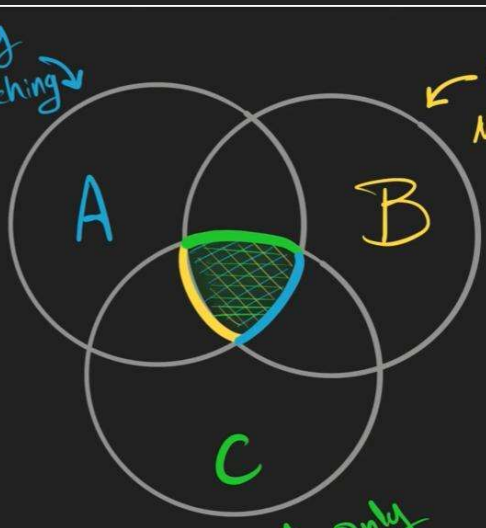
1 Recombine Data

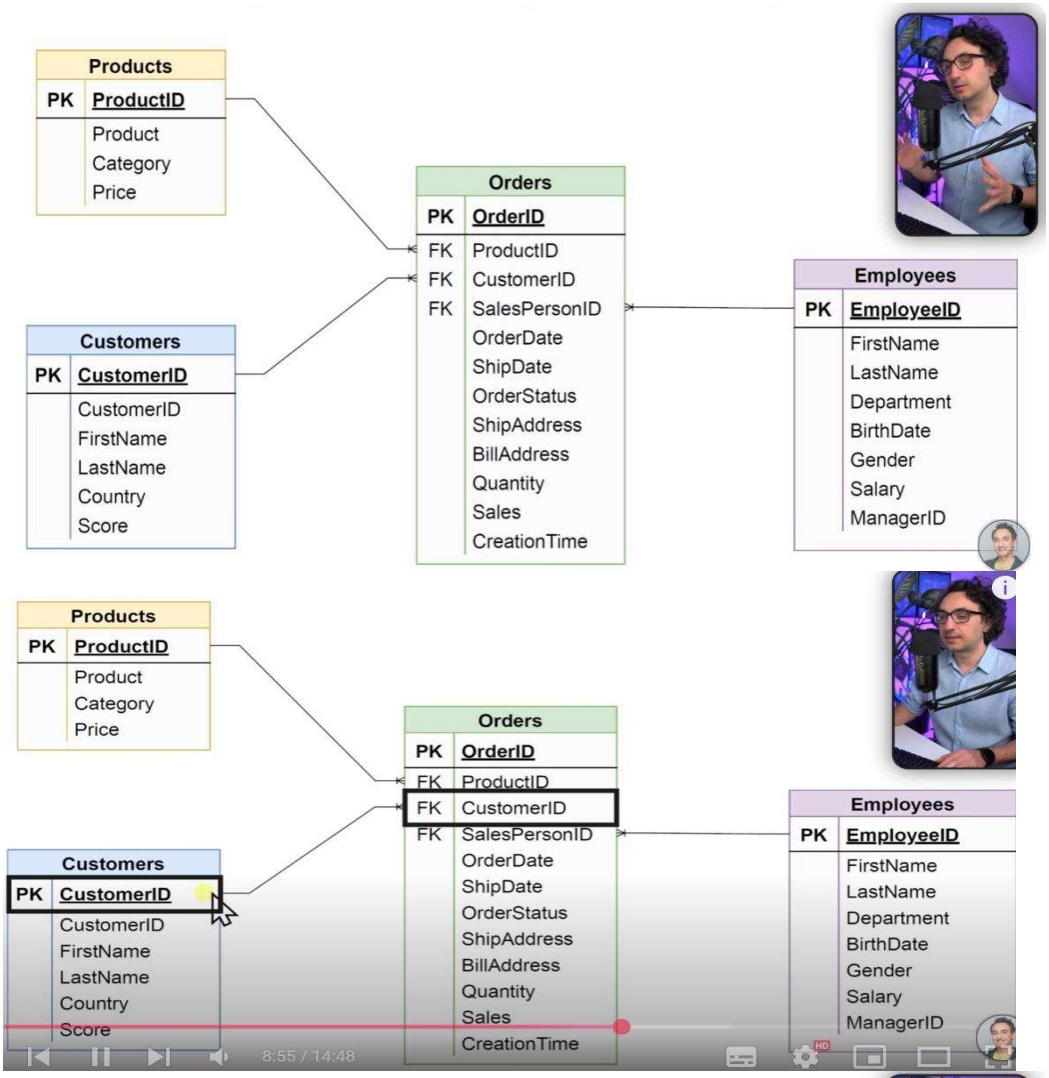


1 Recombine Data




```
SELECT *
FROM A
INNER B ON...
INNER C ON...
  ⋮
```




Products
PK ProductID
Product
Category
Price
Orders
PK OrderID
FK ProductID
FK CustomerID
FK SalesPersonID
OrderDate
ShipDate
OrderStatus
ShipAddress
BillAddress
Quantity
Sales
CreationTime
Customers
PK CustomerID
CustomerID
FirstName
LastName
Country
Score
Employees
PK EmployeeID
FirstName
LastName
Department
BirthDate
Gender
Salary
ManagerID
8:55 / 14:48
11:56 / 14:48

```sql
SELECT
    o.OrderID,
    o.Sales,
    c.FirstName AS CustomerFirstName,
    c.LastName AS CustomerLastName,
    p.Product AS ProductName,
    p.Price,
    e.FirstName AS EmployeeFirstName,
    e.LastName AS EmployeeLastName
FROM Sales.Orders AS o
LEFT JOIN Sales.Customers AS c
ON o.CustomerID = c.CustomerID
LEFT JOIN Sales.Products AS p
ON o.ProductID = p.ProductID
LEFT JOIN Sales.Employees AS e
ON o.SalesPersonID = e.EmployeeID
```

Results Messages

	OrderID	Sales	CustomerFirstName	CustomerLastName	ProductName	Price	EmployeeFirstName	EmployeeLastName
2	2	15	Mary	NULL	Tire	15	Mary	NULL
3	3	20	Jossef	Goldberg	Bottle	10	Carol	Baker
4	4	60	Jossef	Goldberg	Gloves	30	Mary	NULL
5	5	25	Kevin	Brown	Caps	25	Carol	Baker
6	6	50	Mary	NULL	Caps	25	Carol	Baker
7	7	30	Jossef	Goldberg	Tire	15	Frank	Lee
8	8	90	Mark	Schwarz	Bottle	10	Mary	NULL
9	9	20	Kevin	Brown	Bottle	10	Mary	NULL
10	10	60	Mary	NULL	Tire	15	Carol	Baker

SET OPERATORS

Set operations in SQL combine the results of multiple queries into a single result set.

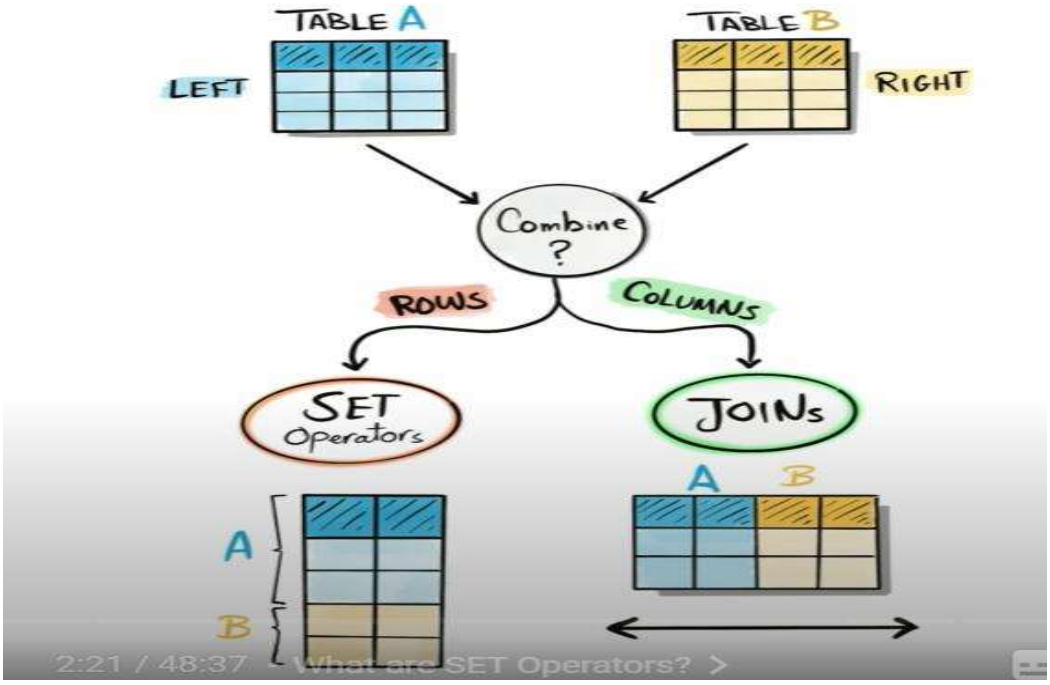


TABLE A
LEFT
TABLE B
RIGHT
Combine ?
ROWS
COLUMNS
SET Operators
JOINs
A
B
A
B
2:21 / 48:37
What are SET Operators? >

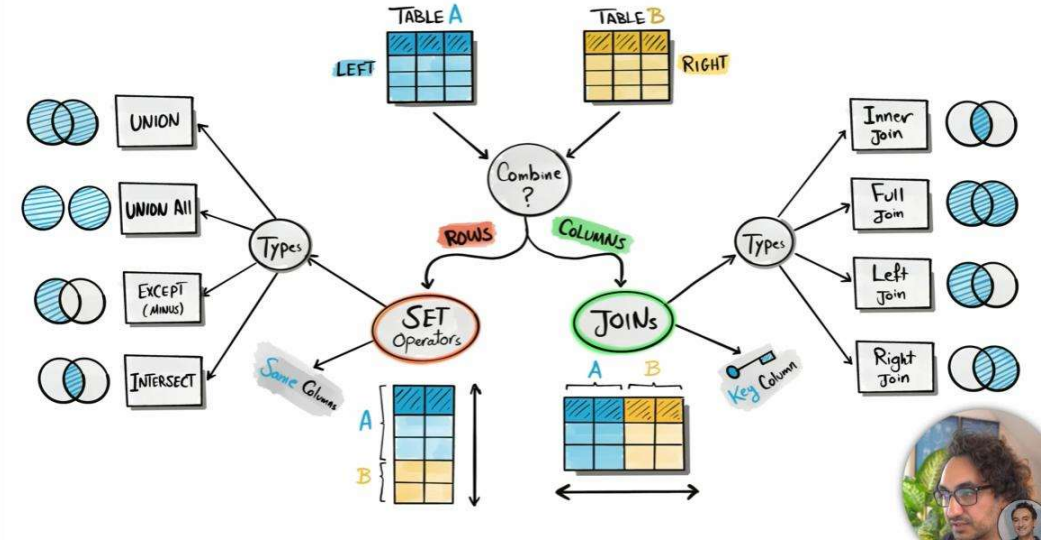
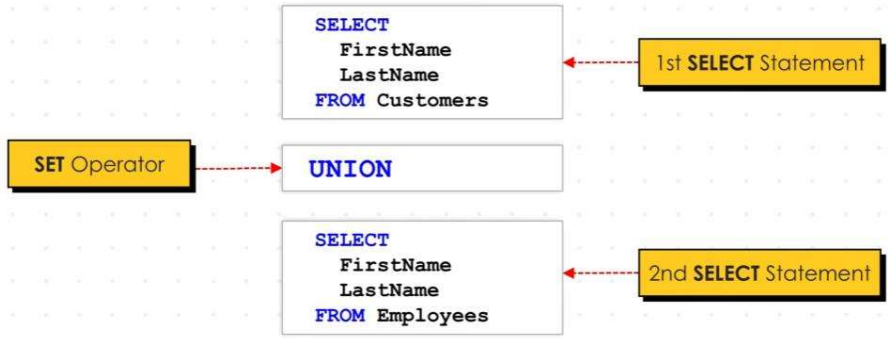


TABLE A
LEFT
TABLE B
RIGHT
Combine ?
ROWS
COLUMNS
Types
UNION
UNION All
EXCEPT (MINUS)
INTERSECT
SET Operators
Same Columns
A
B
JOINs
Types
Inner Join
Full Join
Left Join
Right Join
Key Column
A
B

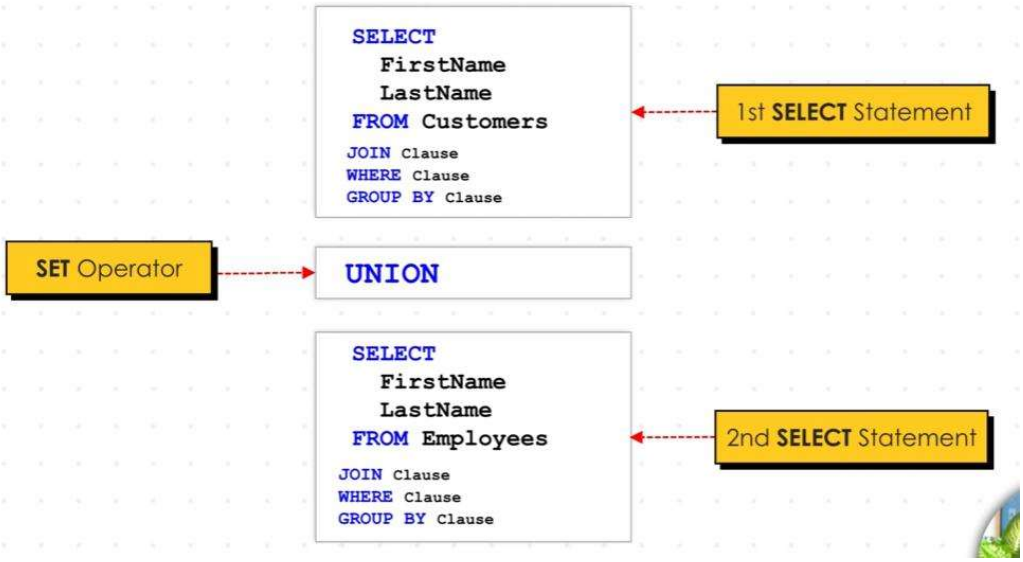


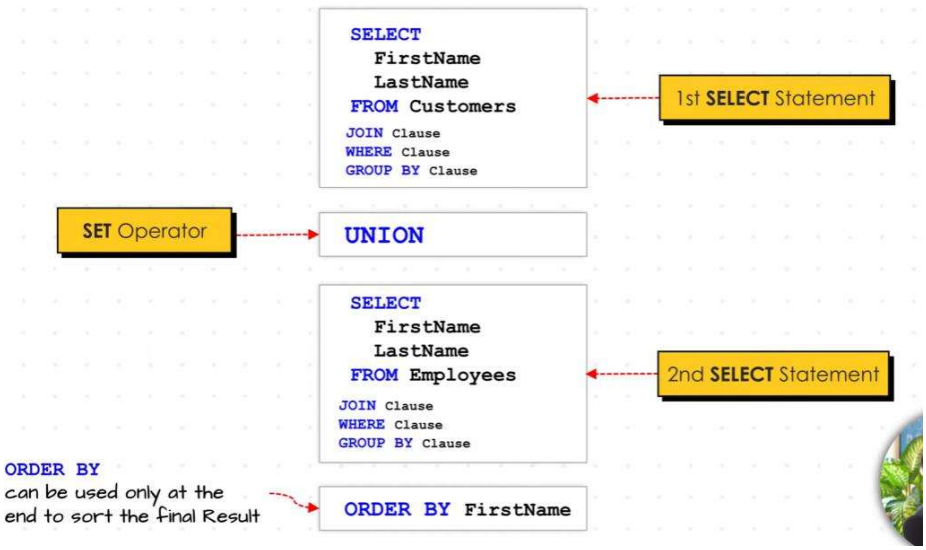
#1 RULE | SQL CLAUSES

- SET Operator can be used almost in all clauses

  WHERE | JOIN | GROUP BY | HAVING

- ORDER BY is allowed only once at the end of query





#2 RULE | NUMBER OF COLUMNS

The number of columns in each query must be the same

Ex-

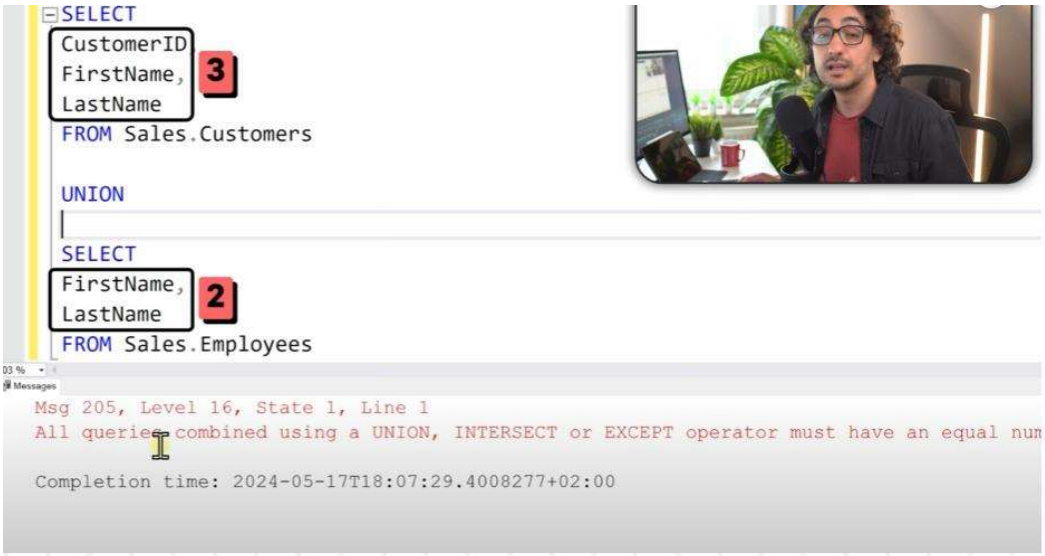
```
SELECT
FirstName,
LastName
FROM Sales.Customers

UNION

SELECT
FirstName,
LastName
FROM Sales.Employees
```

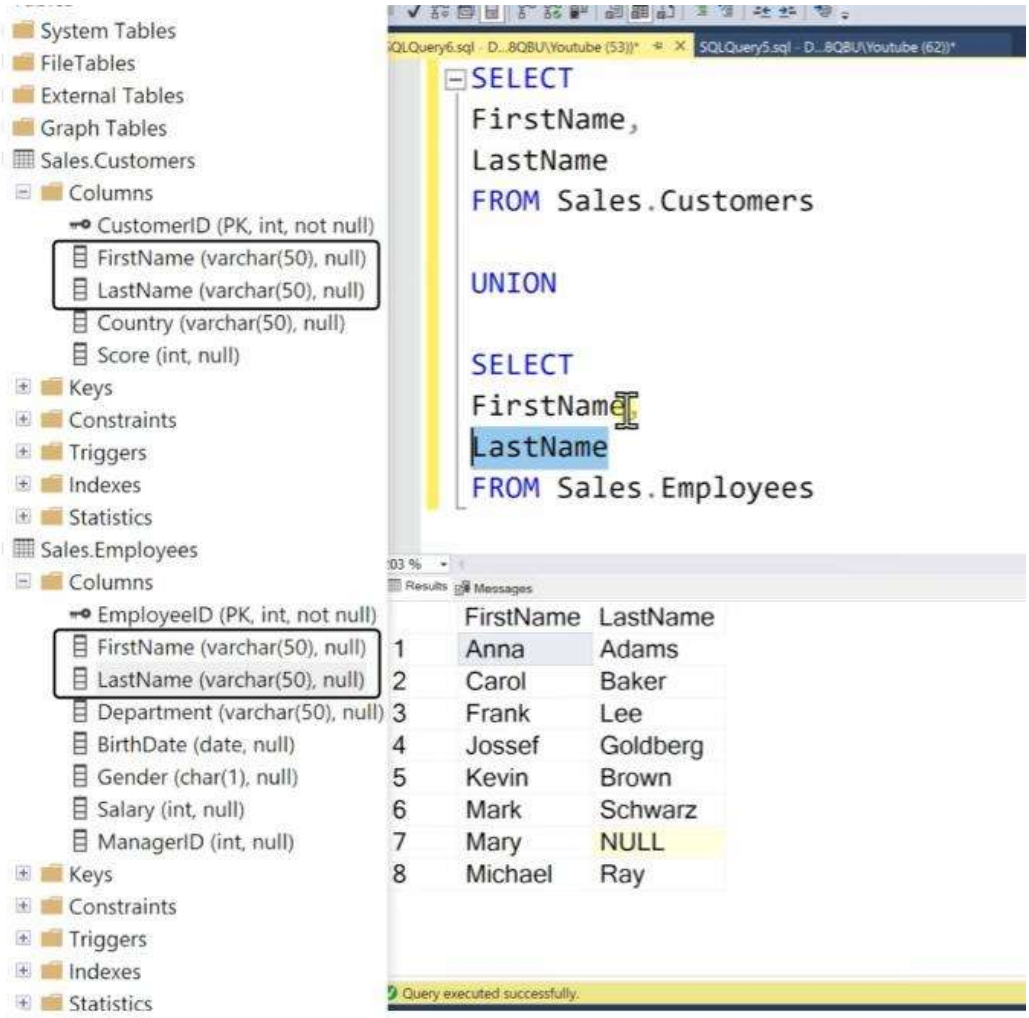
	FirstName	LastName
1	Anna	Adams
2	Carol	Baker
3	Frank	Lee
4	Jossef	Goldberg
5	Kevin	Brown
6	Mark	Schwarz
7	Mary	NULL
8	Michael	Ray

Error because set operators need equal number of columns

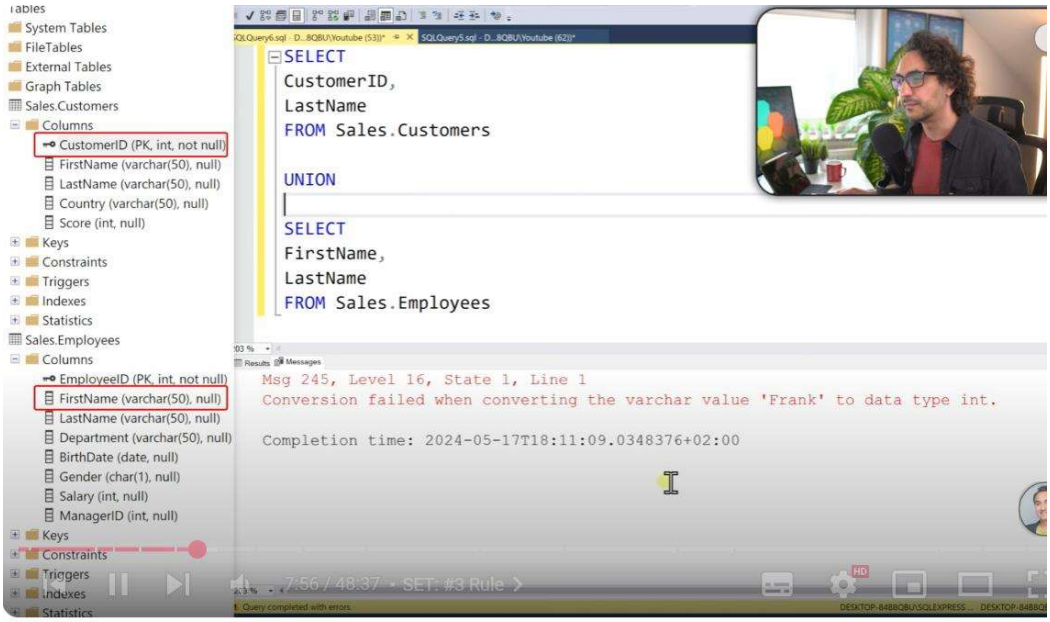


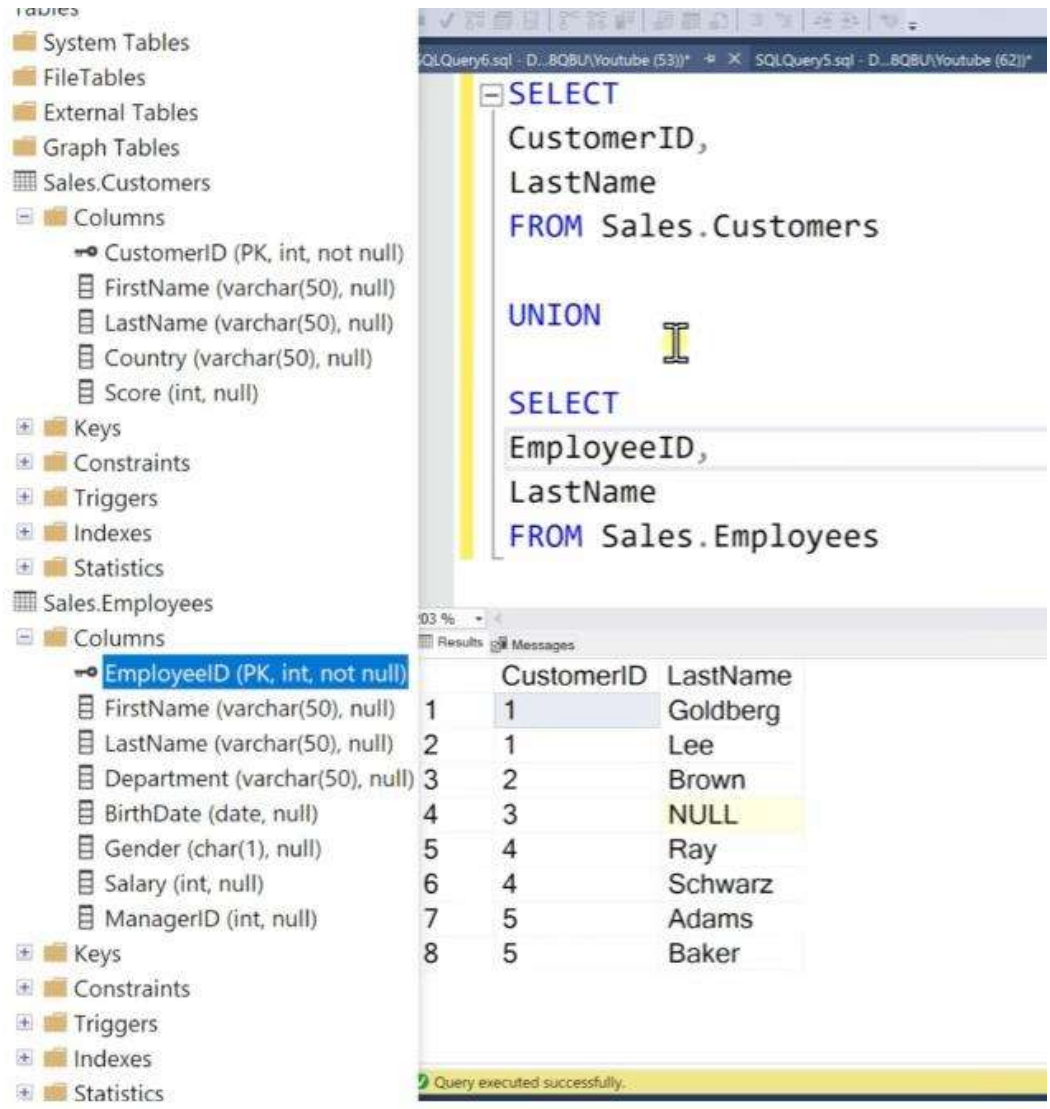
#3 RULE | DATA TYPES

Data types of columns in each query must be compatible



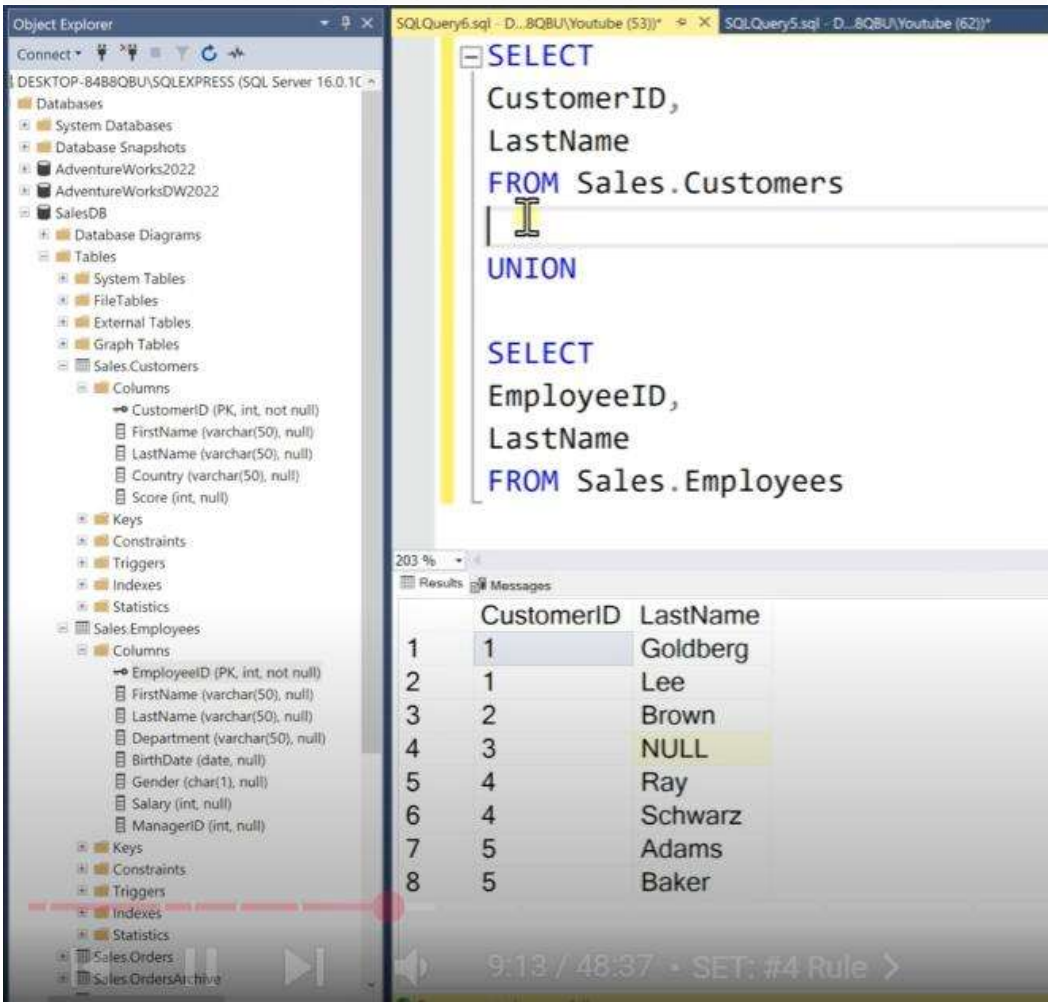
Mismatched Data Type:

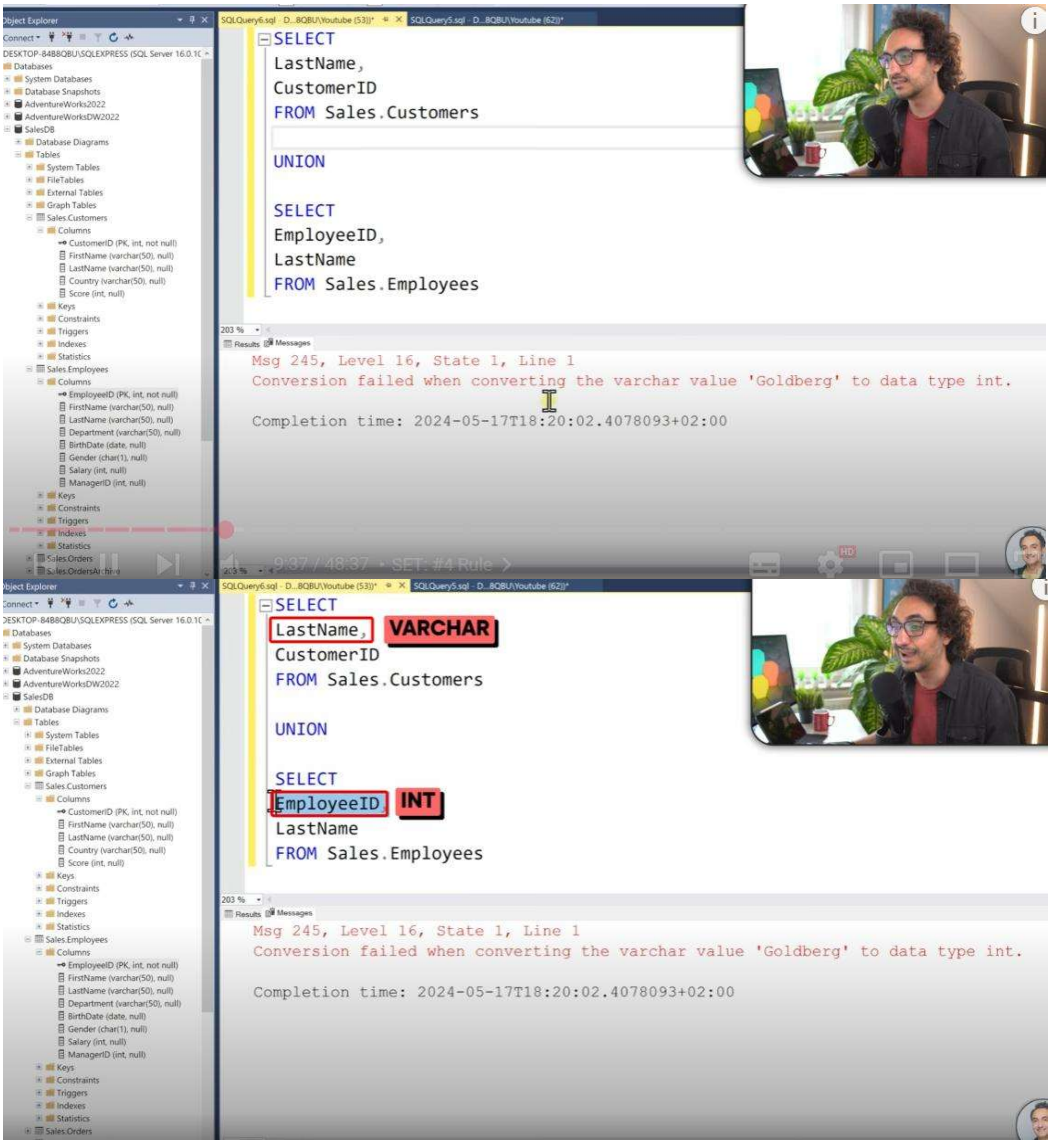

Tables
System Tables
FileTables
External Tables
Graph Tables
Sales.Customers
Columns
CustomerID (PK, int, not null)
FirstName (varchar(50), null)
LastName (varchar(50), null)
Country (varchar(50), null)
Score (int, null)
Keys
Constraints
Triggers
Indexes
Statistics
Sales.Employees
Columns
EmployeeID (PK, int, not null)
FirstName (varchar(50), null)
LastName (varchar(50), null)
Department (varchar(50), null)
BirthDate (date, null)
Gender (char(1), null)
Salary (int, null)
ManagerID (int, null)
Keys
Constraints
Triggers
Indexes
Statistics
SELECT
CustomerID,
LastName
FROM Sales.Customers
UNION
SELECT
FirstName,
LastName
FROM Sales.Employees
Msg 245, Level 16, State 1, Line 1
Conversion failed when converting the varchar value 'Frank' to data type int.
Completion time: 2024-05-17T18:11:09.0348376+02:00


System Tables
FileTables
External Tables
Graph Tables
Sales.Customers
Columns
CustomerID (PK, int, not null)
FirstName (varchar(50), null)
LastName (varchar(50), null)
Country (varchar(50), null)
Score (int, null)
Keys
Constraints
Triggers
Indexes
Statistics
Sales.Employees
Columns
EmployeeID (PK, int, not null)
FirstName (varchar(50), null)
LastName (varchar(50), null)
Department (varchar(50), null)
BirthDate (date, null)
Gender (char(1), null)
Salary (int, null)
ManagerID (int, null)
Keys
Constraints
Triggers
Indexes
Statistics
SELECT
CustomerID,
LastName
FROM Sales.Customers
UNION
SELECT
EmployeeID,
LastName
FROM Sales.Employees
Results
Messages
CustomerID LastName
1 1 Goldberg
2 1 Lee
3 2 Brown
4 3 NULL
5 4 Ray
6 4 Schwarz
7 5 Adams
8 5 Baker
Query executed successfully.

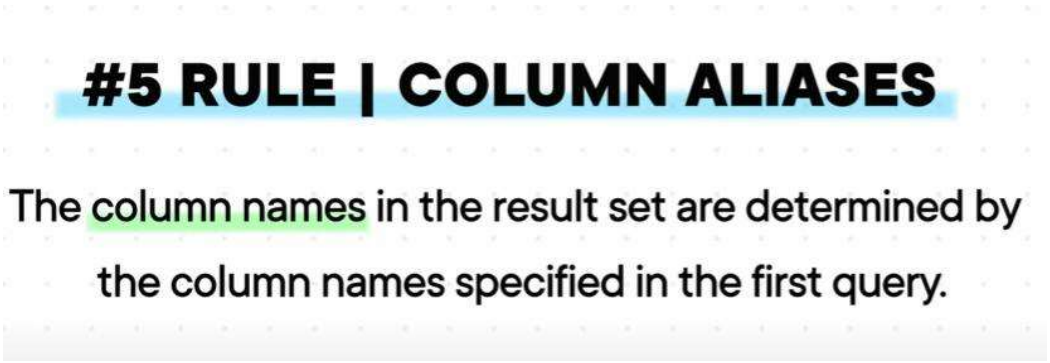
#4 RULE | ORDER OF COLUMNS

The order of the columns in each query must be the same

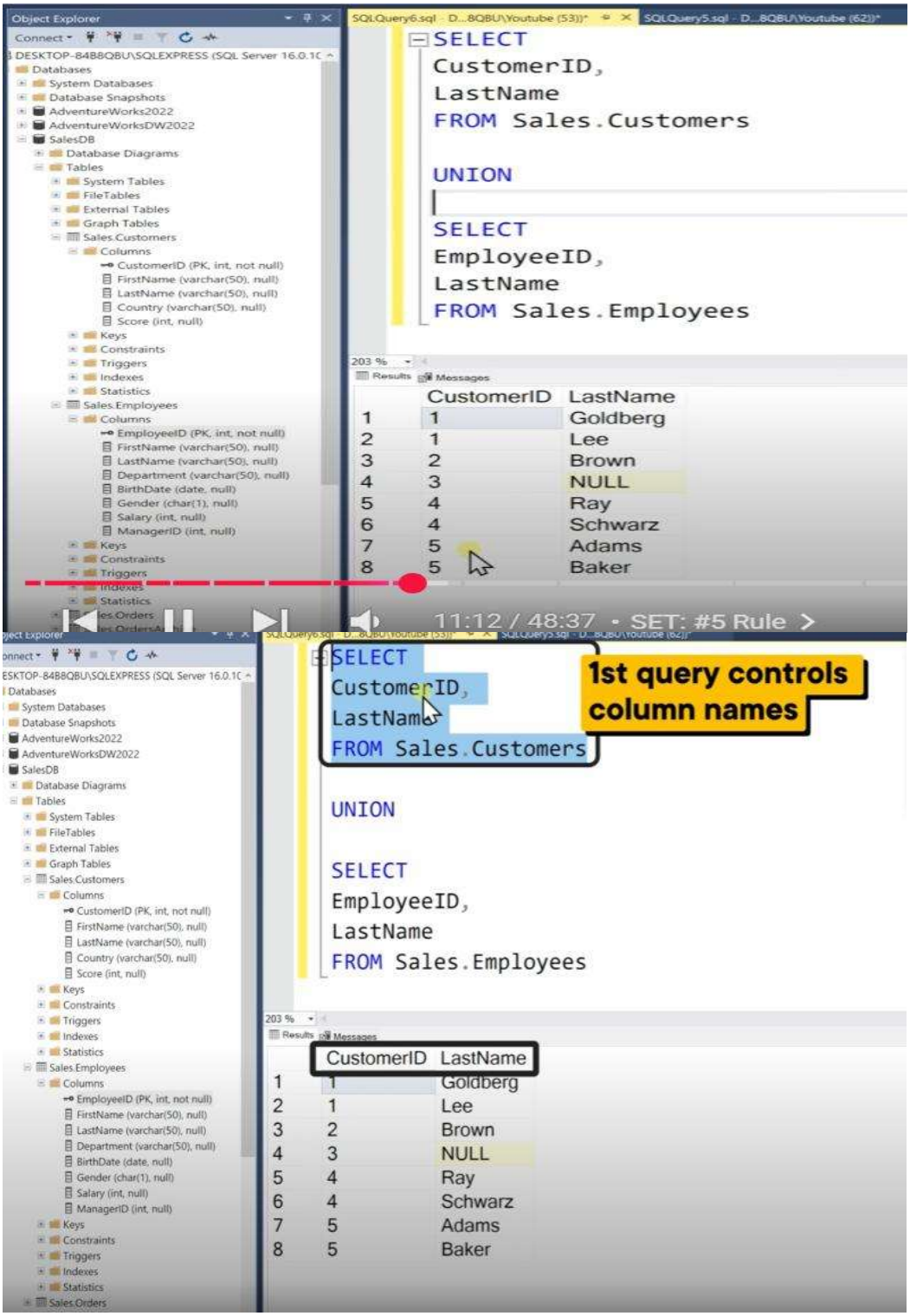


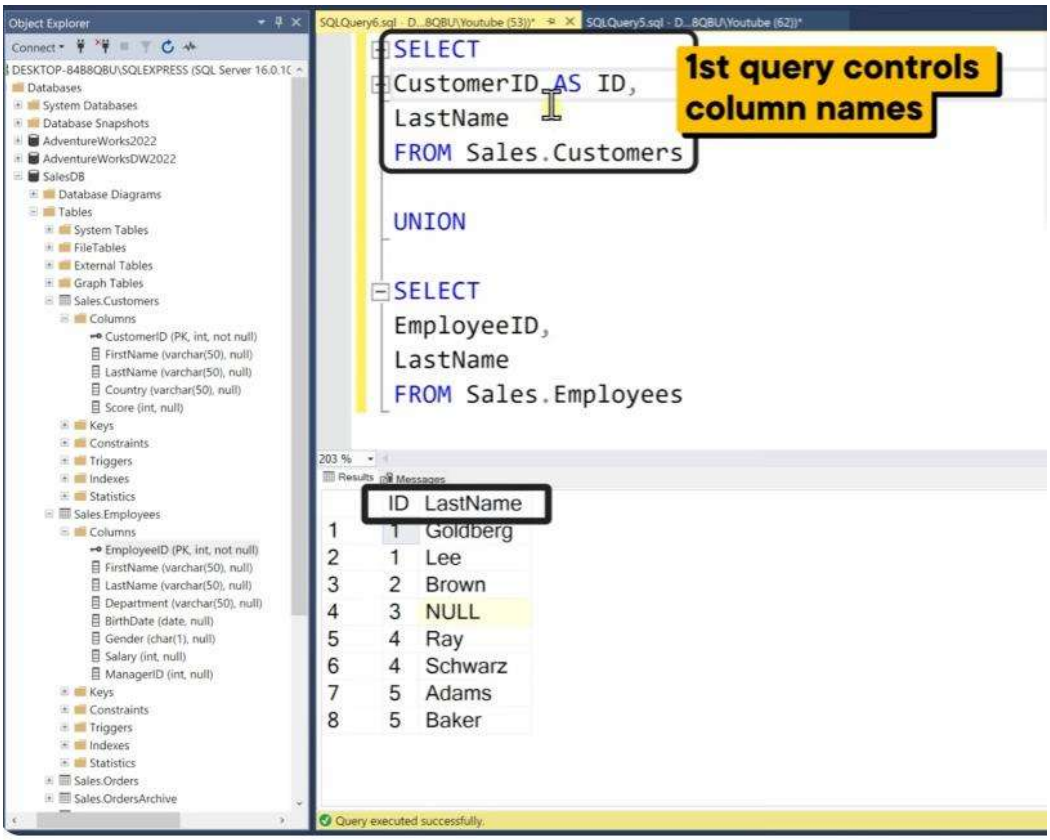


SQL match or map columns with its order.

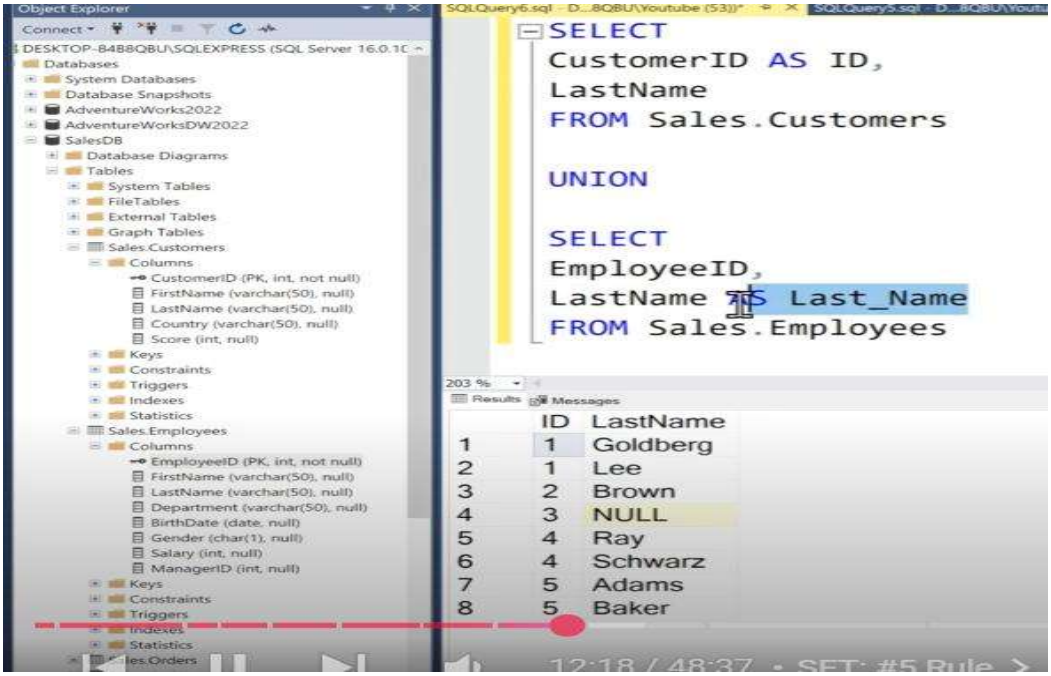


First Query Controls the naming of Output Here naming of second query is totally Ignored


Object Explorer
Connect
DESKTOP-84B8QBU\SQLEXPRESS (SQL Server 16.0.1C
Databases
System Databases
Database Snapshots
AdventureWorks2022
AdventureWorksDW2022
SalesDB
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
Sales.Customers
Columns
CustomerID (PK, int, not null)
FirstName (varchar(50), null)
LastName (varchar(50), null)
Country (varchar(50), null)
Score (int, null)
Keys
Constraints
Triggers
Indexes
Statistics
Sales.Employees
Columns
EmployeeID (PK, int, not null)
FirstName (varchar(50), null)
LastName (varchar(50), null)
Department (varchar(50), null)
BirthDate (date, null)
Gender (char(1), null)
Salary (int, null)
ManagerID (int, null)
Keys
Constraints
Triggers
Indexes
Statistics
SQLQuery6.sql - D...8QBU\Youtube (53))*
SQLQuery5.sql - D...8QBU\Youtube (62))*
SELECT
CustomerID,
LastName
FROM Sales.Customers
UNION
SELECT
EmployeeID,
LastName
FROM Sales.Employees
203 %
Results
Messages
CustomerID LastName
1 1 Goldberg
2 1 Lee
3 2 Brown
4 3 NULL
5 4 Ray
6 4 Schwarz
7 5 Adams
8 5 Baker
11:12 / 48:37 • SET: #5 Rule >
1st query controls column names



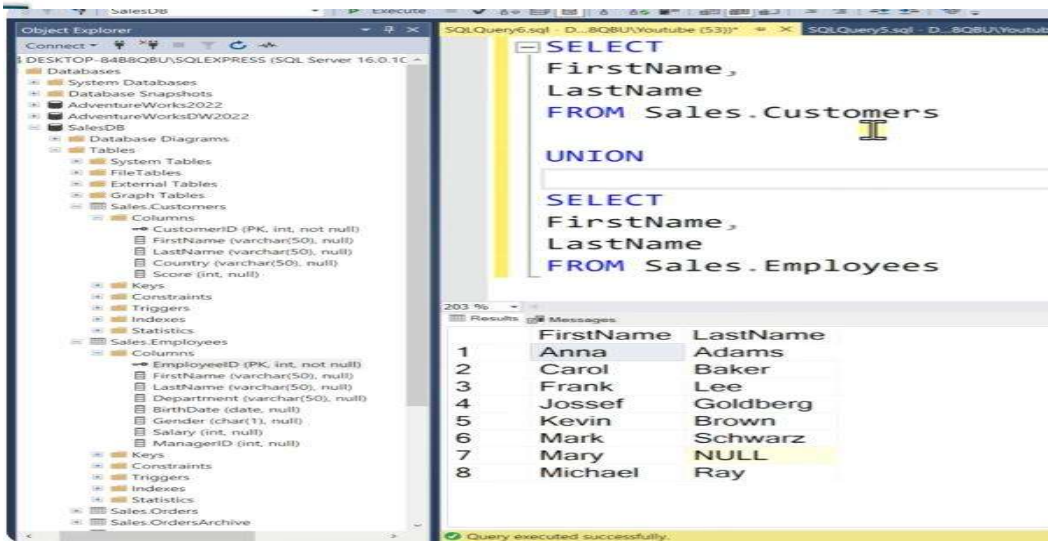
All Aliasis do in the first query not in second.



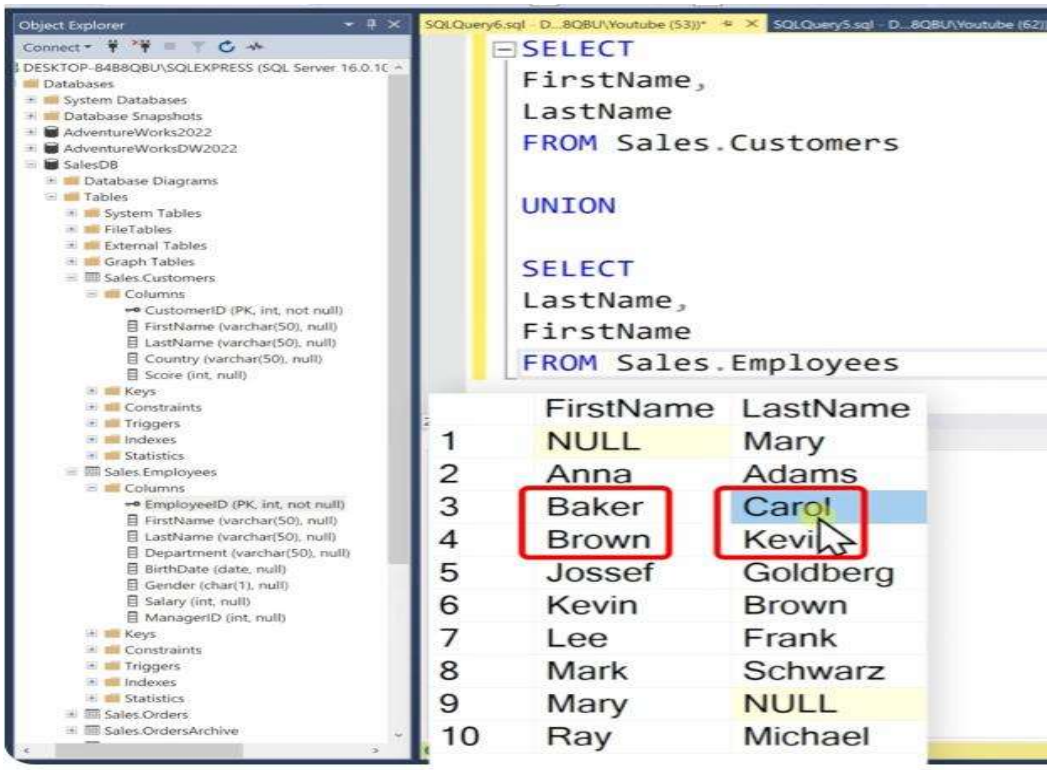
#6 RULE | CORRECT COLUMNS

- Even if all rules are met and SQL shows no errors, the result may be incorrect.

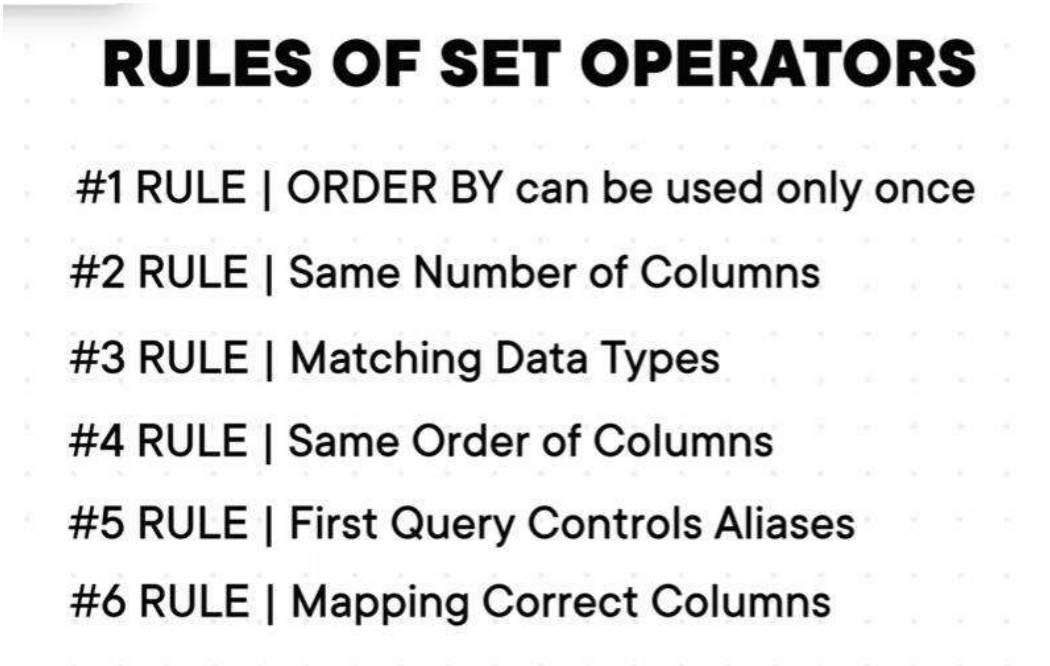
- Incorrect column selection leads to inaccurate results.



In accurate result



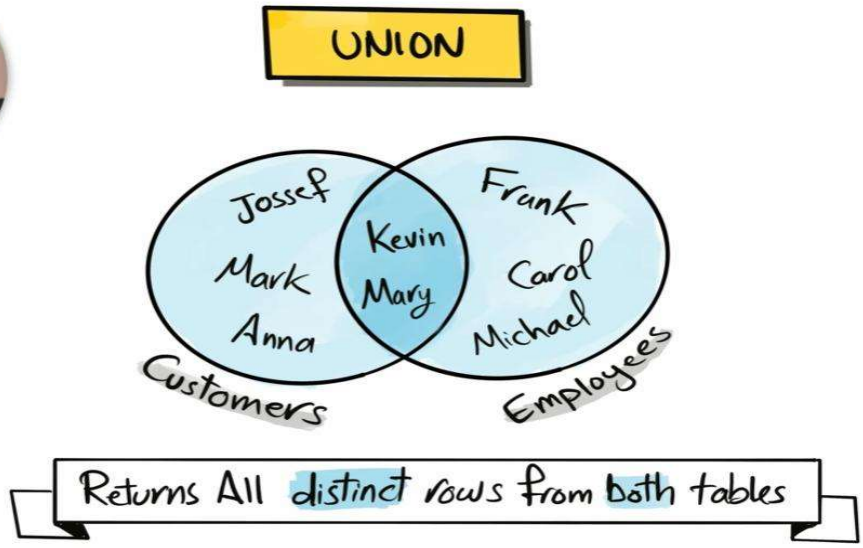
Same information should mapped between two queries that we have to take care other wise we will get inaccurate result.

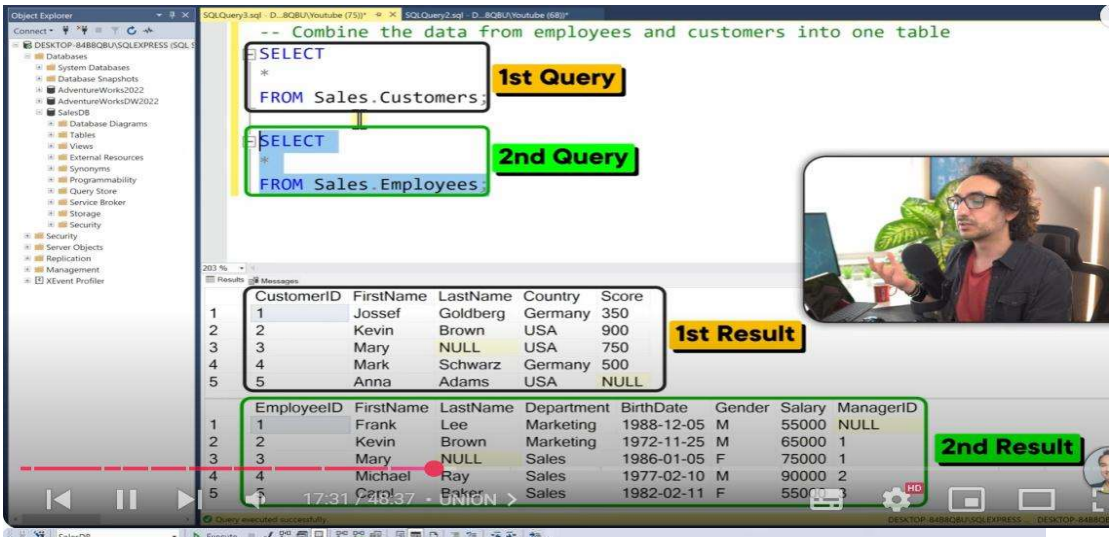


UNION

- Returns all district rows from both queries.

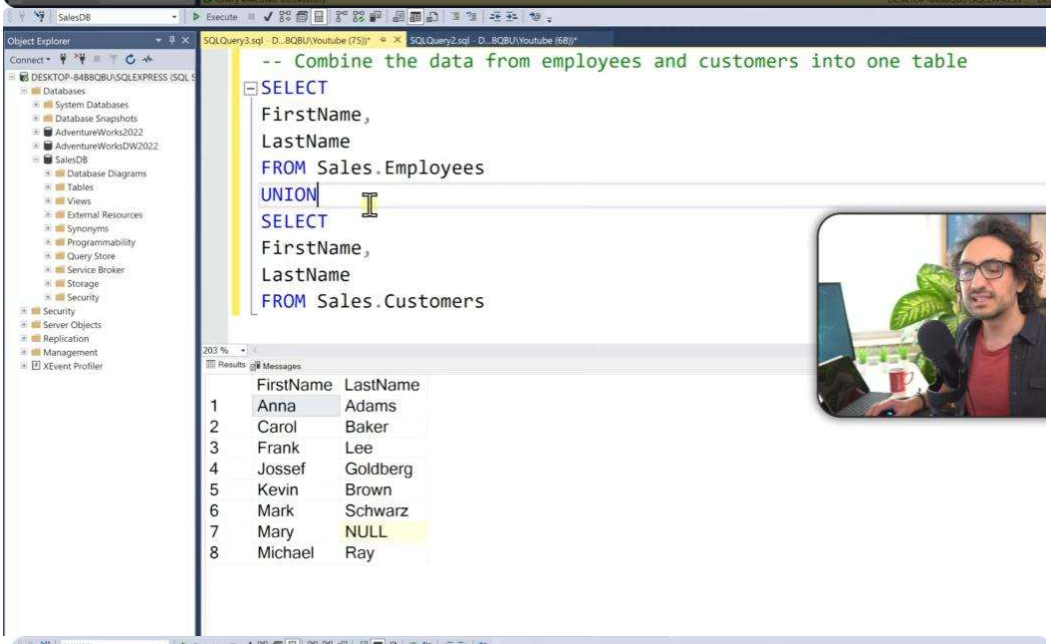
- Removes duplicate rows from the result.



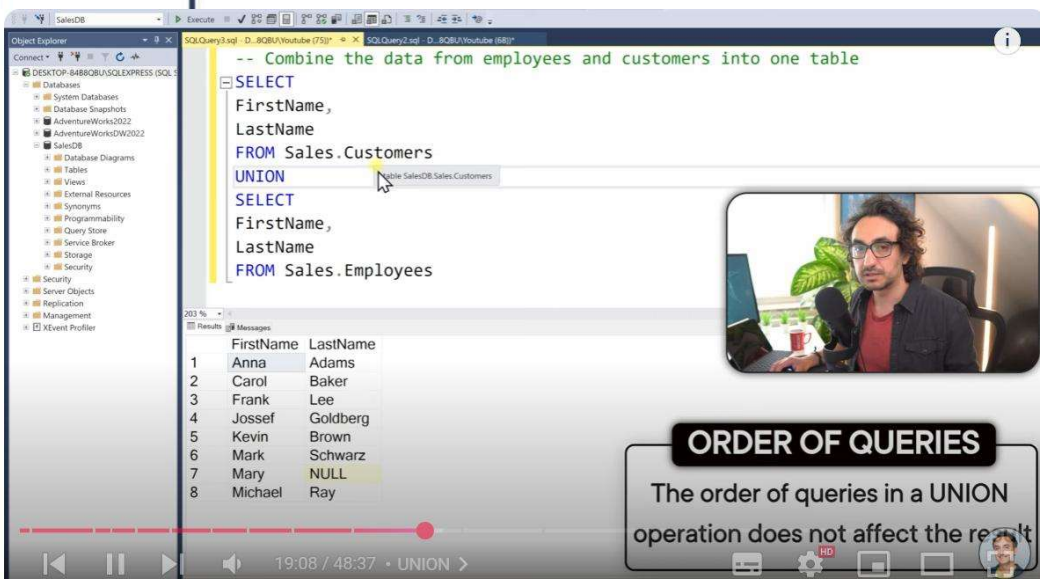


	CustomerID	FirstName	LastName	Country	Score
1	1	Jossef	Goldberg	Germany	350
2	2	Kevin	Brown	USA	900
3	3	Mary	NULL	USA	750
4	4	Mark	Schwarz	Germany	500
5	5	Anna	Adams	USA	NULL

	EmployeeID	FirstName	LastName	Department	BirthDate	Gender	Salary	ManagerID
1	1	Frank	Lee	Marketing	1988-12-05	M	55000	NULL
2	2	Kevin	Brown	Marketing	1972-11-25	M	65000	1
3	3	Mary	NULL	Sales	1986-01-05	F	75000	1
4	4	Michael	Ray	Sales	1977-02-10	M	90000	2
5	5	Carol	Baker	Sales	1982-02-11	F	55000	3

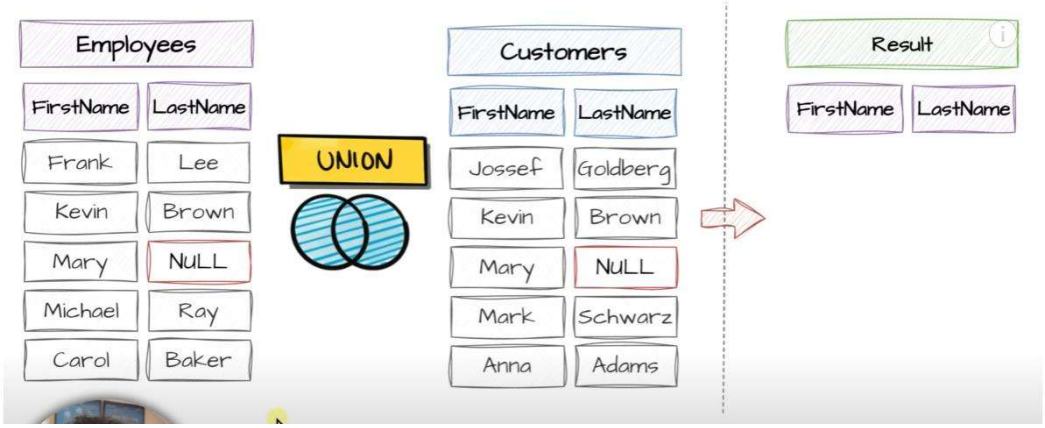


	FirstName	LastName
1	Anna	Adams
2	Carol	Baker
3	Frank	Lee
4	Jossef	Goldberg
5	Kevin	Brown
6	Mark	Schwarz
7	Mary	NULL
8	Michael	Ray

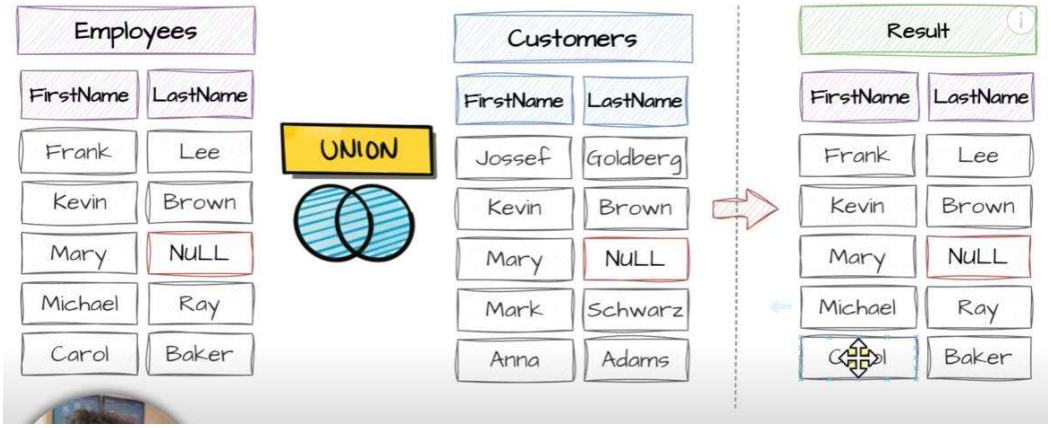


	FirstName	LastName
1	Anna	Adams
2	Carol	Baker
3	Frank	Lee
4	Jossef	Goldberg
5	Kevin	Brown
6	Mark	Schwarz
7	Mary	NULL
8	Michael	Ray

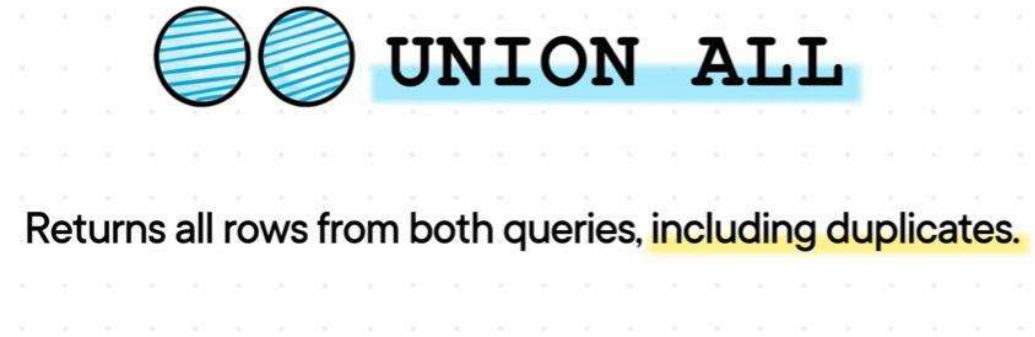
How Union Works - Execution of union



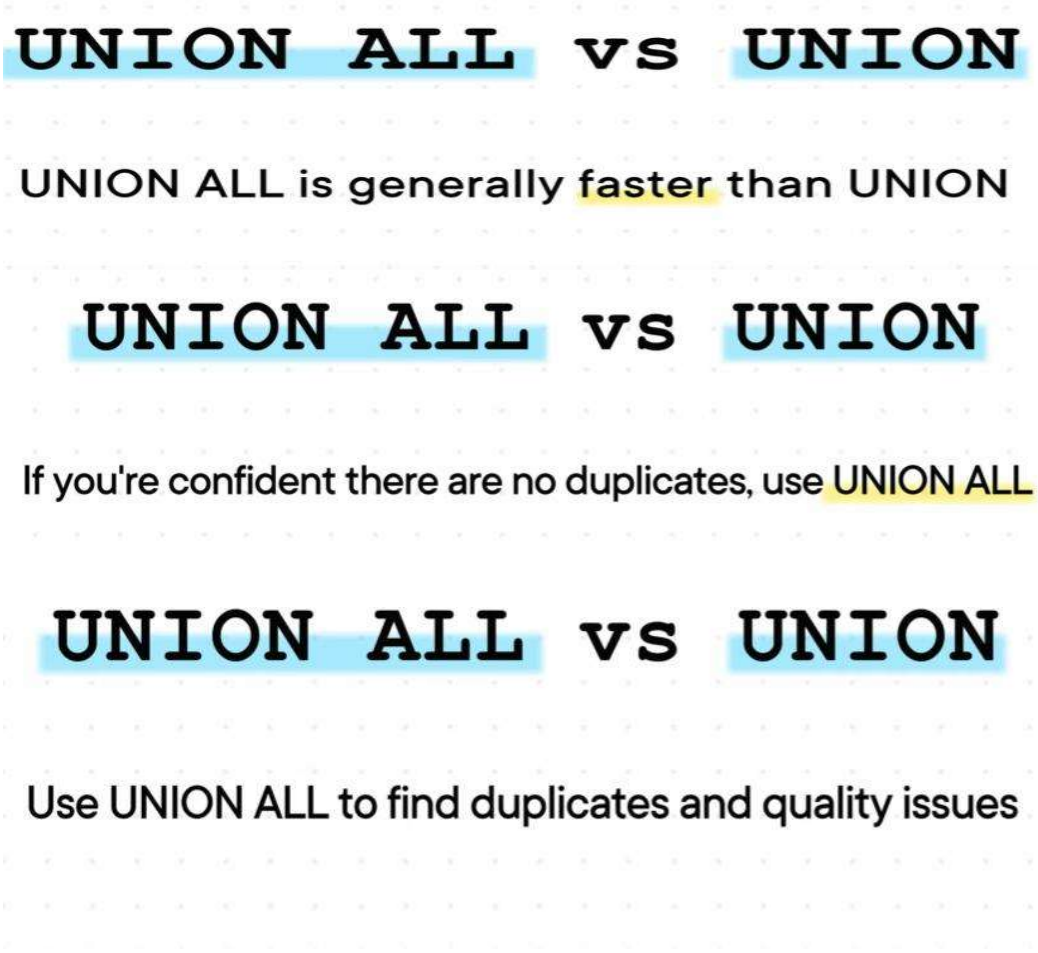
All all data from first table with no duplicates values

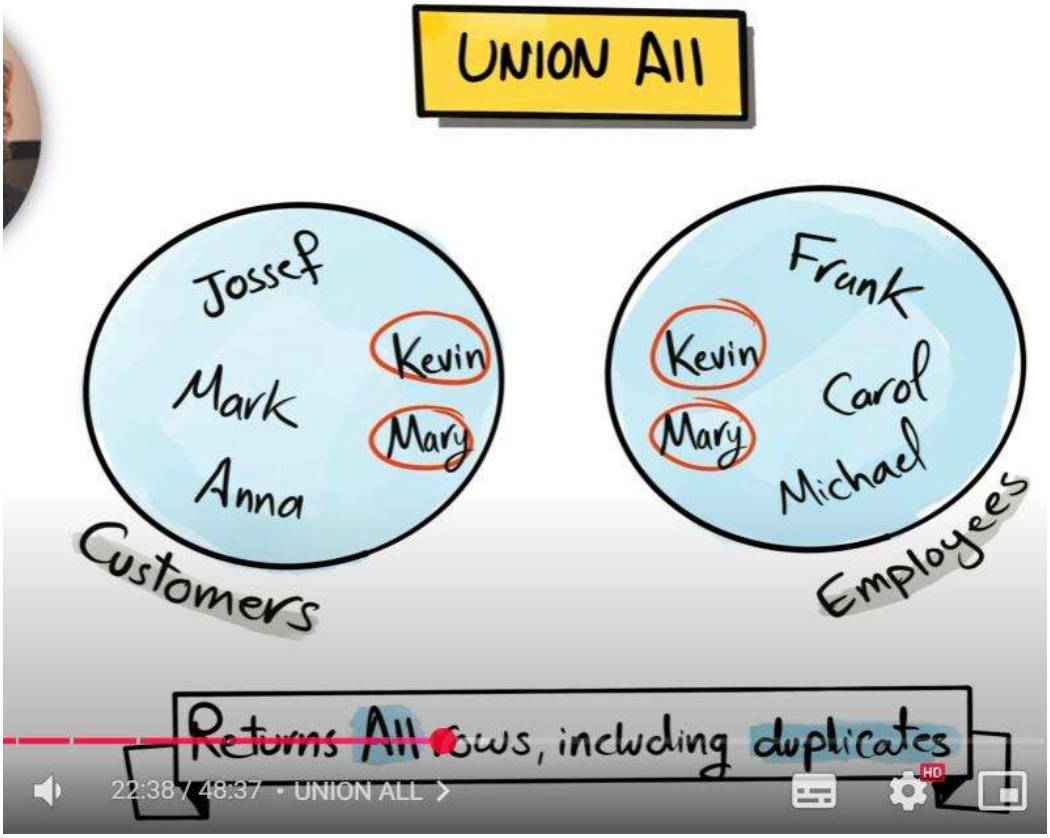


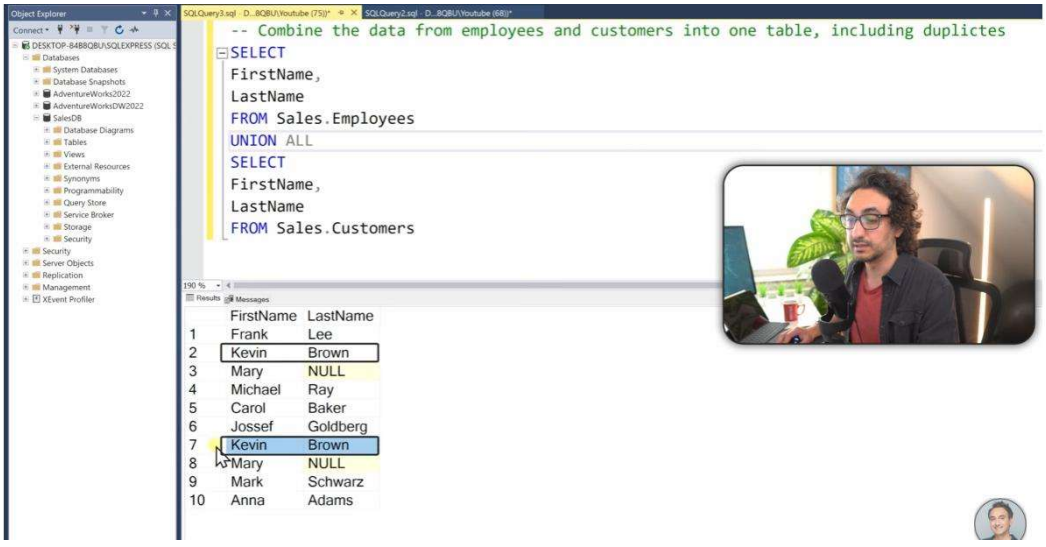
Combines all data from second table by checking it become duplicate from first table or not if it same in first table then SQL ignores that and check for next one

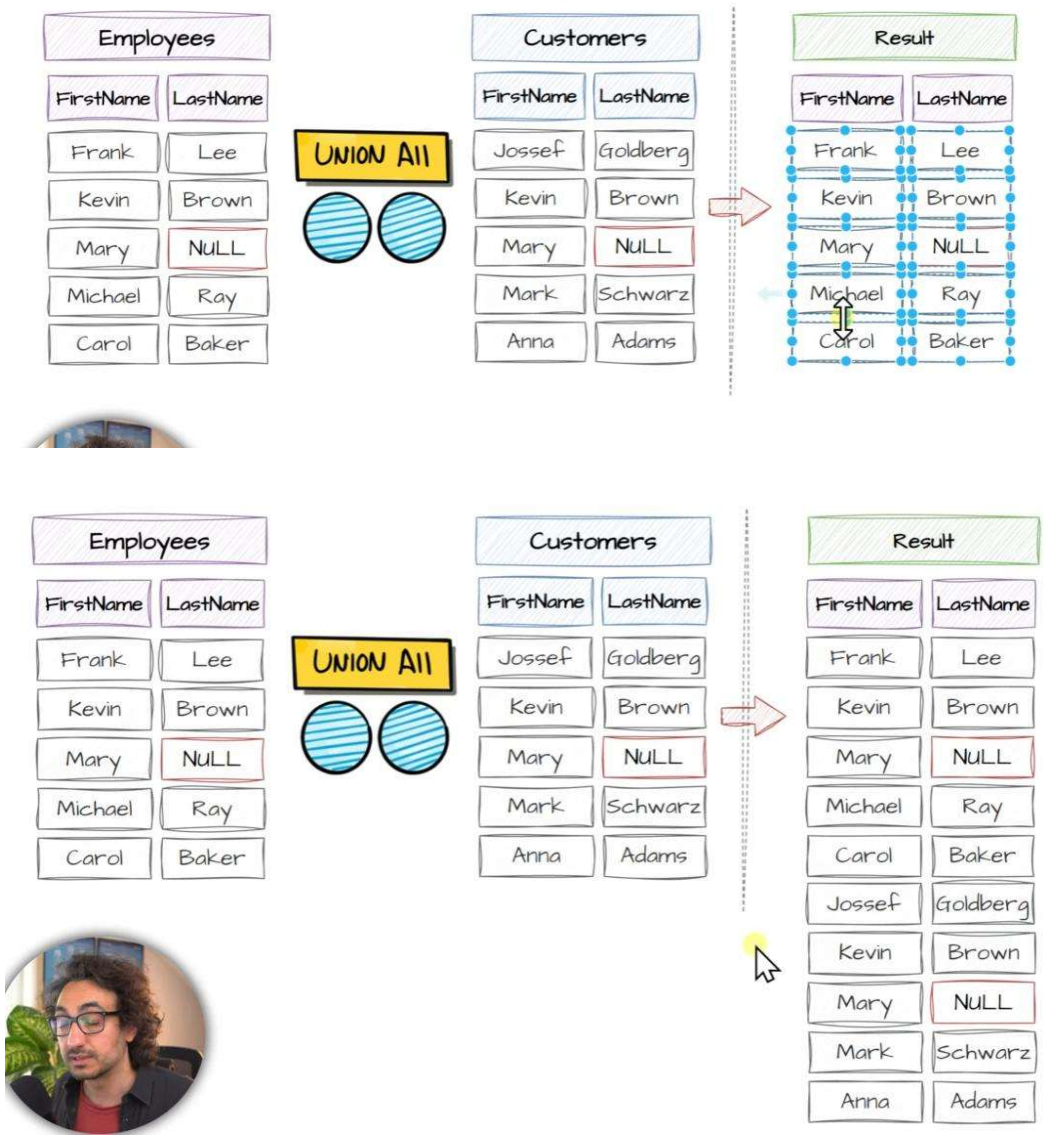


It combine the data from both query into one table along with duplicated it keep whole data
Not removed duplicated

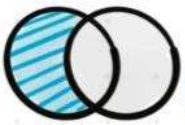



UNION All
Jossef
Mark
Anna
Kevin
Mary
Customers
Frank
Kevin
Mary
Carol
Michael
Employees
Returns All rows, including duplicates
22:38 / 48:37 • UNION ALL


-- Combine the data from employees and customers into one table, including duplictes
SELECT
FirstName,
LastName
FROM Sales.Employees
UNION ALL
SELECT
FirstName,
LastName
FROM Sales.Customers
FirstName LastName
1 Frank Lee
2 Kevin Brown
3 Mary NULL
4 Michael Ray
5 Carol Baker
6 Jossef Goldberg
7 Kevin Brown
8 Mary NULL
9 Mark Schwarz
10 Anna Adams



Except - Minus in Other databases but in sql we call it except



EXCEPT

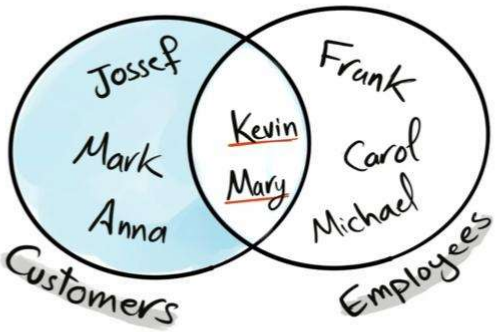
- Returns all distinct rows from the first query that are not found in the second query.

- It is the only one where the order of queries affects the final result.

it's like the others it's going to go I remove the duplicates from the result

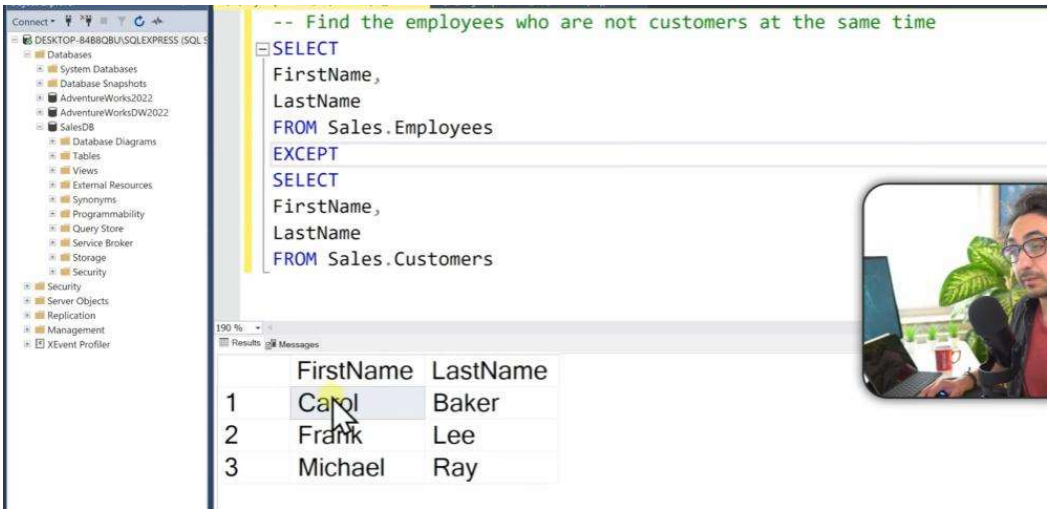
EXCEPT

(MINUS)

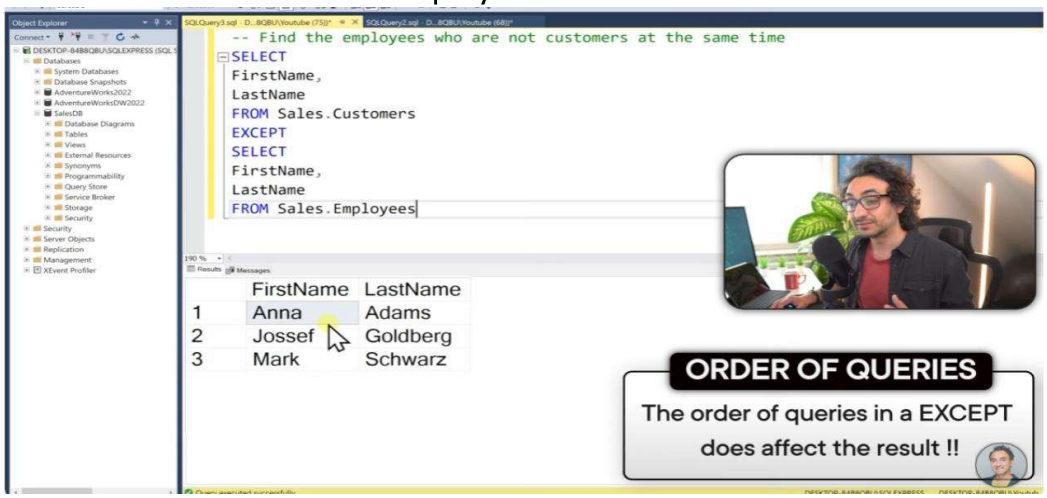


Returns unique rows in 1st Table that are not in 2nd Table

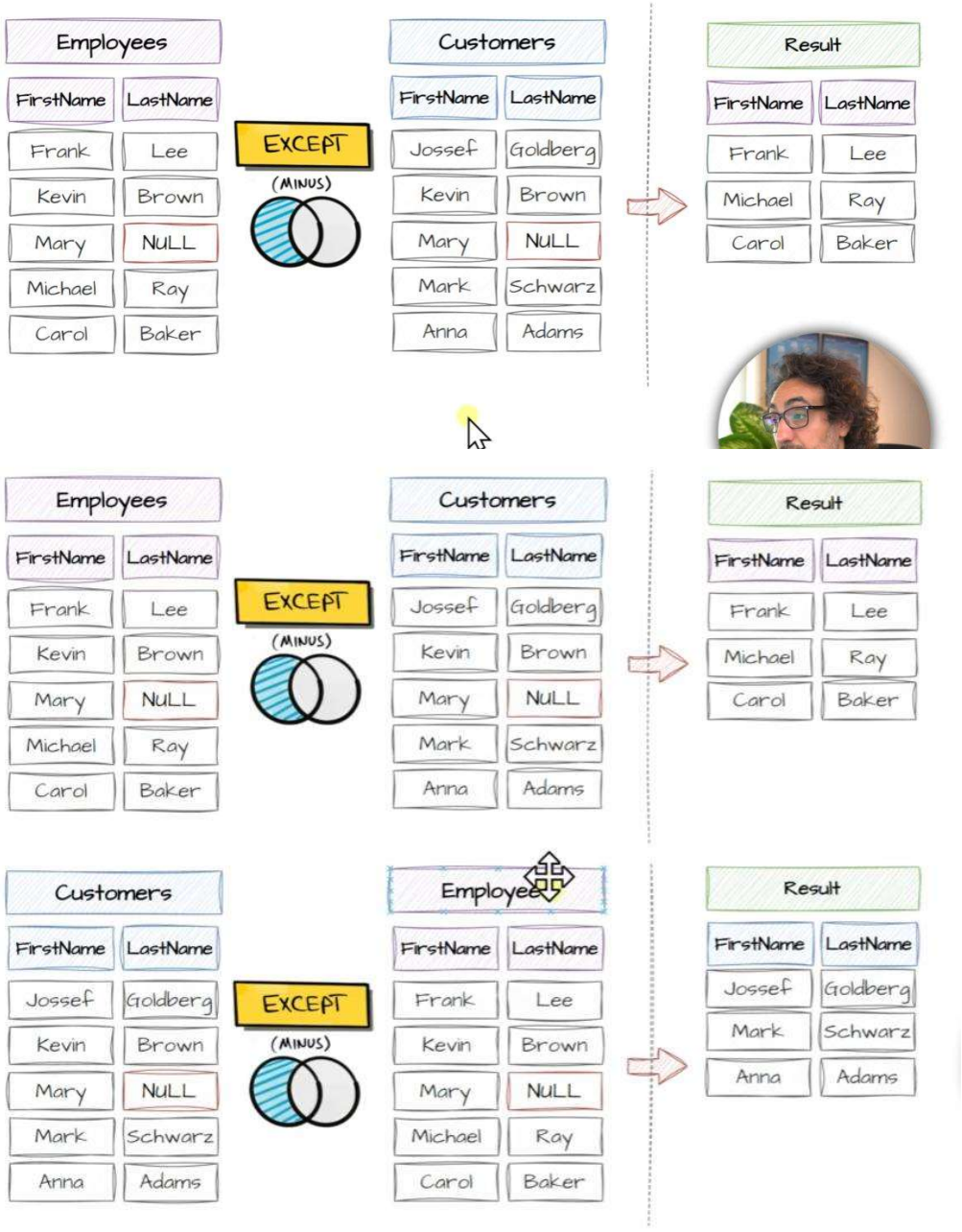
Employees who are not customers



Customers --- who are not employees



Data of one table check into another wheather employee exist in customer table or not if exist then it would not added in result if not exist it add into final result.
Order of query is very important

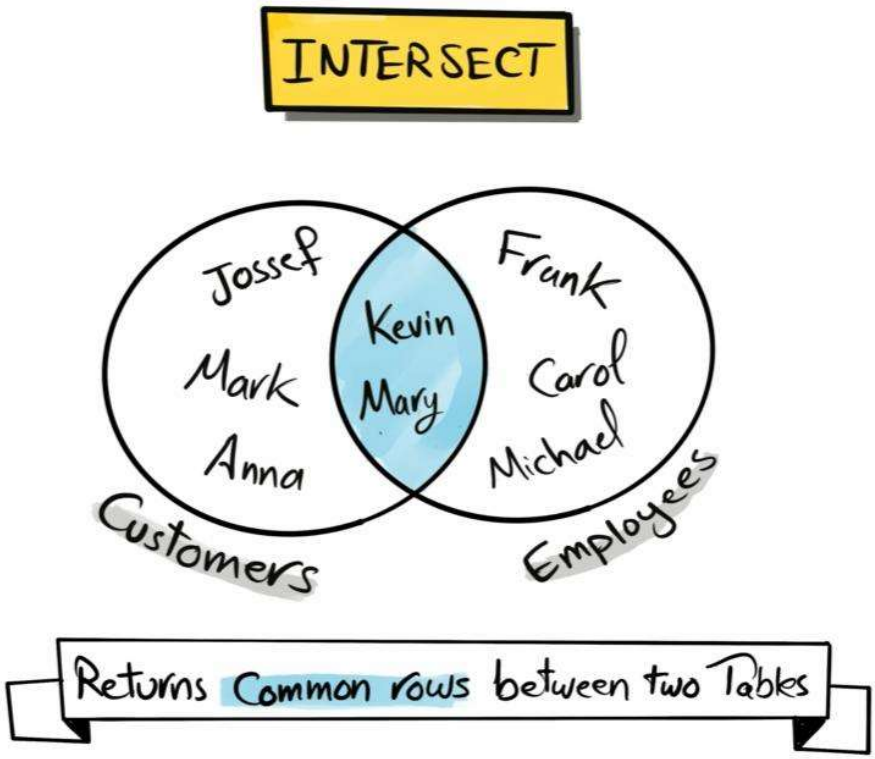


Customer is main and empoyee table is for check in second sceneriao as we changed order

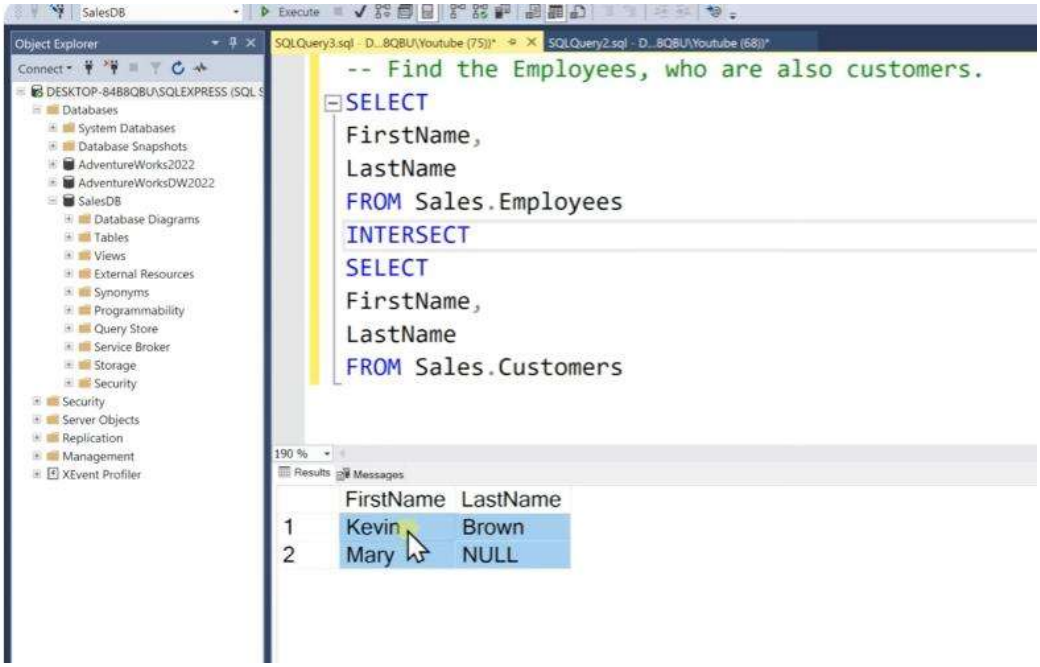
Intersect returns the duplicates

INTERSECT

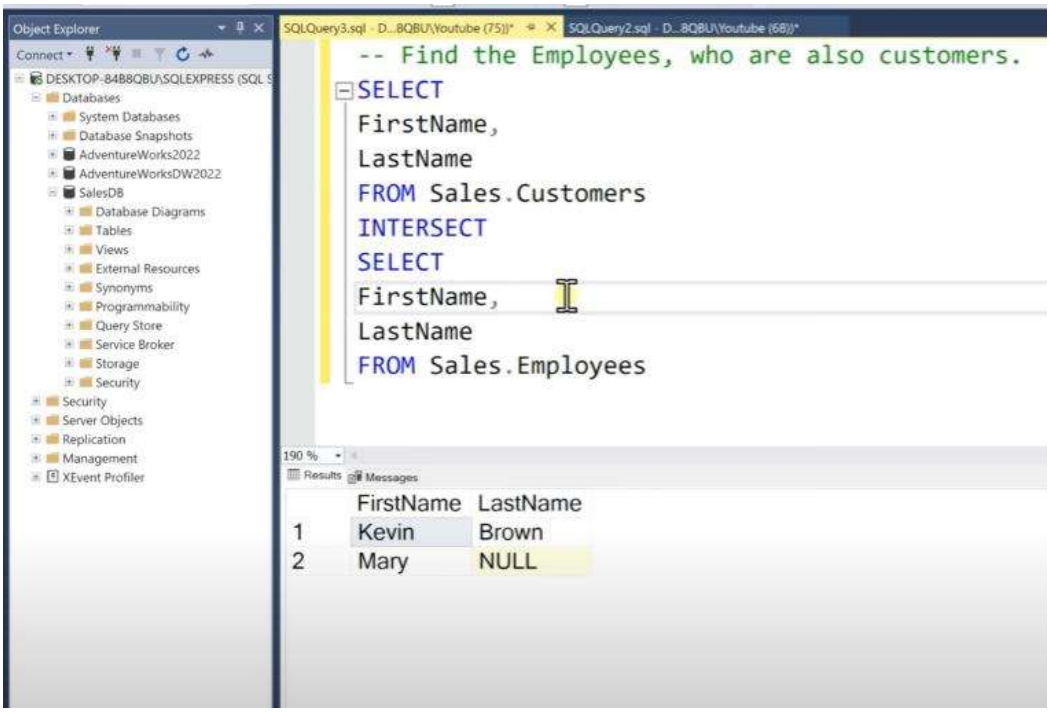
Returns only the rows that are common in both queries



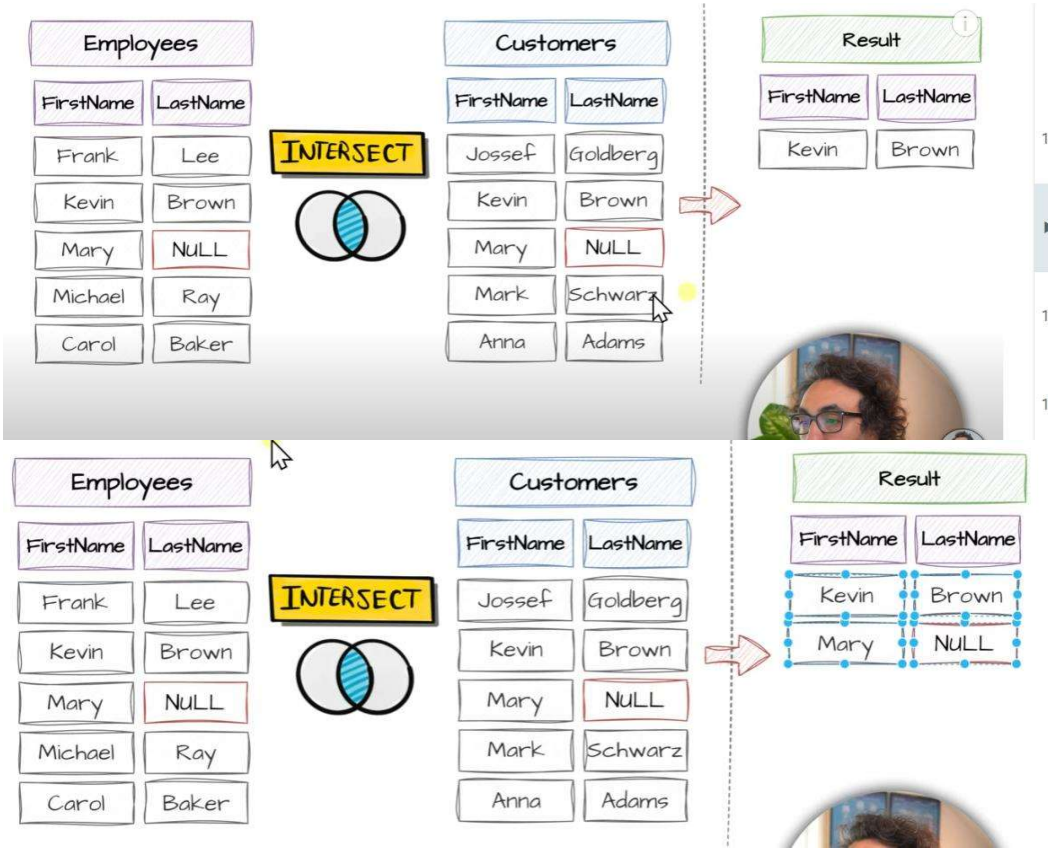
Returns Common rows between two Tables



Order not matters here



Row by row check going on from main table to lookup table



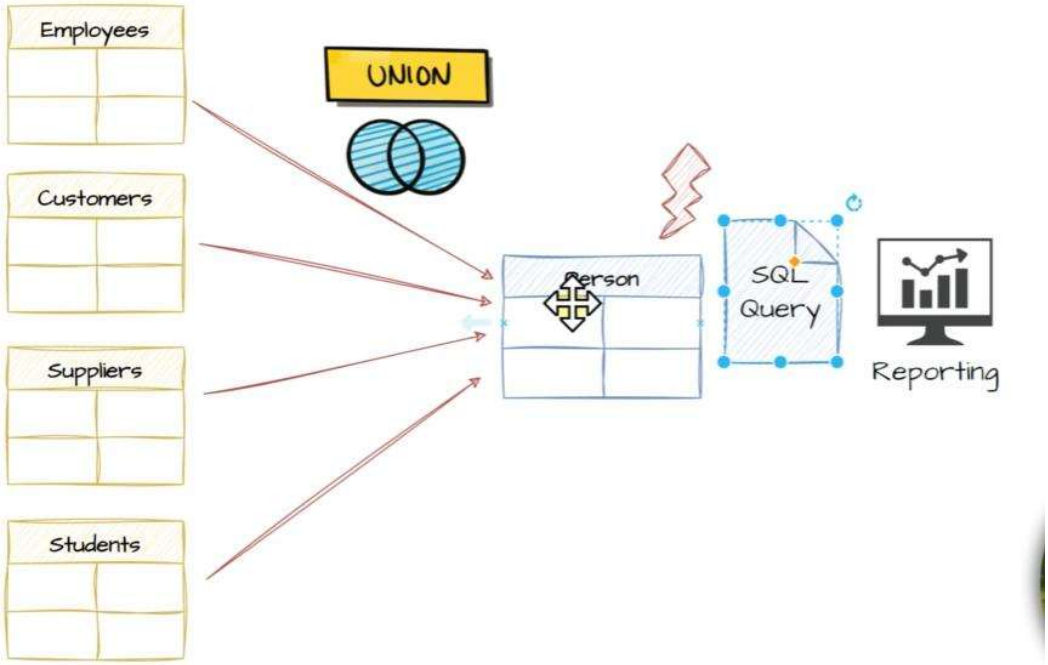
How to used sql for report generation

We have four tables contains info about people now we need to generate report of those
So we will write sql query for each and generate report
What if we forget logic what we write in one query although all queries are of same patterns logic because all tables contains persons info. So this things becomes very hectic and if we write separate logic for each query then there may be a inconsistency in report

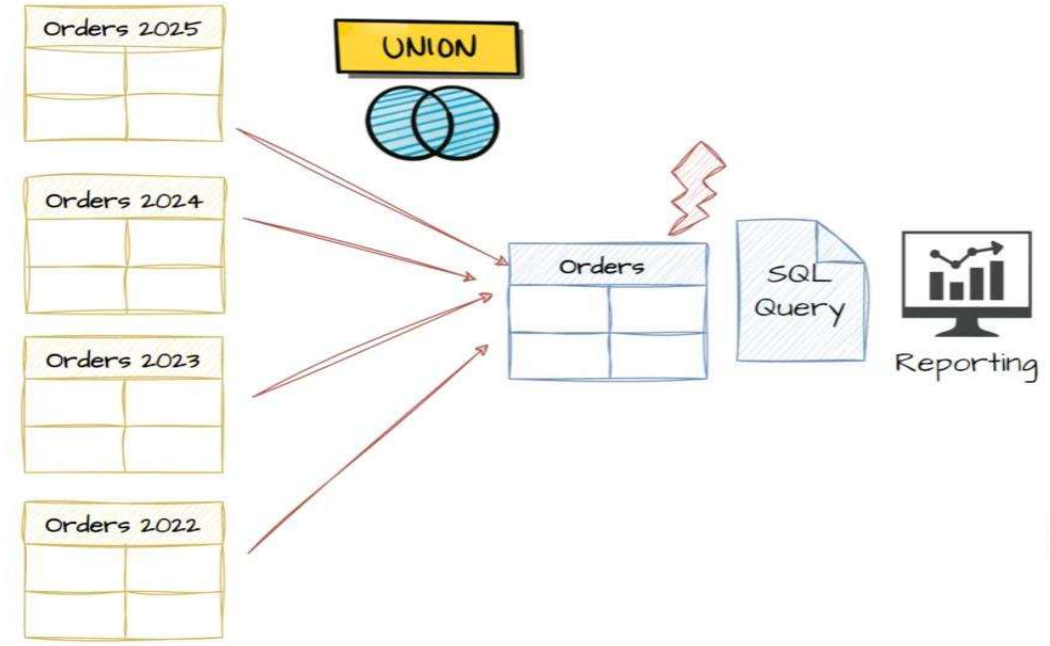
Generated Report would not be a clear.

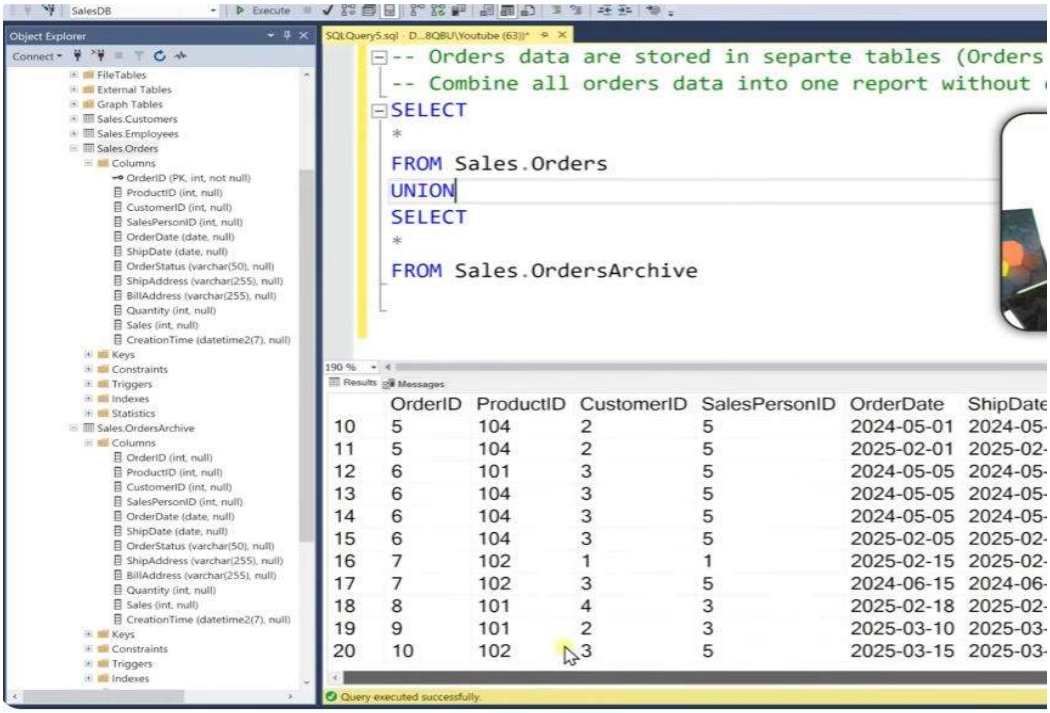
Now we will used set operators inorder to combine all tables into one table

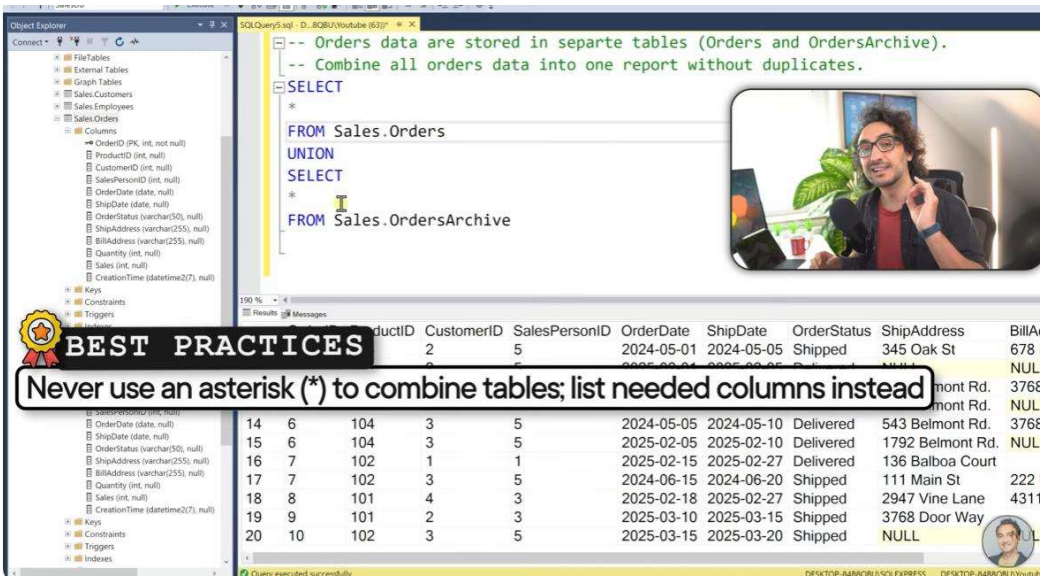
Example for data Analyst-

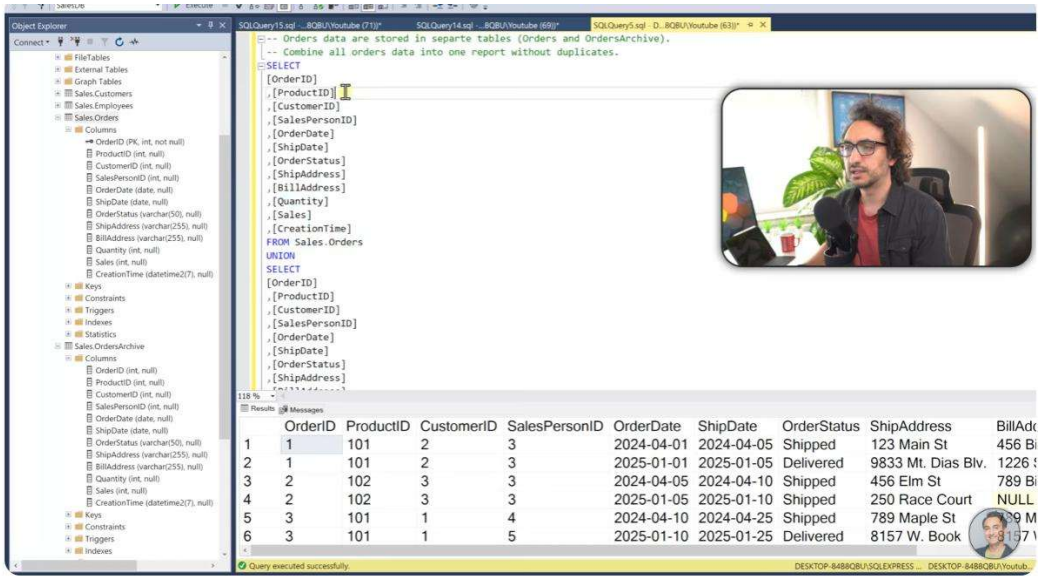


One query so report would be consistent we can chage logic at any time

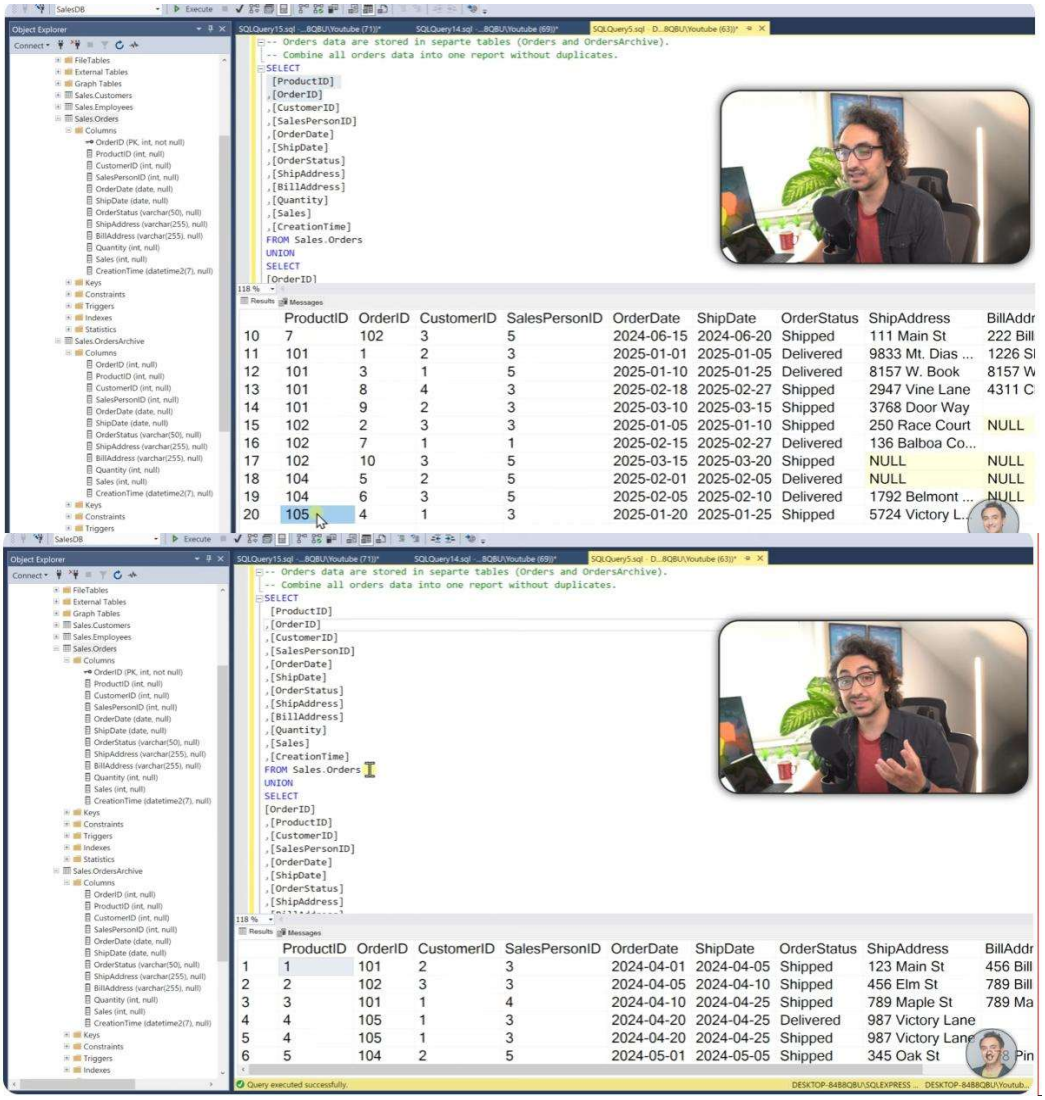

Orders 2025
UNION
Orders 2024
Orders
SQL Query
Reporting
Orders 2023
Orders 2022


Object Explorer
Connect
FileTables
External Tables
Graph Tables
Sales.Customers
Sales.Employees
Sales.Orders
Columns
OrderID (PK, int, not null)
ProductID (int, null)
CustomerID (int, null)
SalesPersonID (int, null)
OrderDate (date, null)
ShipDate (date, null)
OrderStatus (varchar(50), null)
ShipAddress (varchar(255), null)
BillAddress (varchar(255), null)
Quantity (int, null)
Sales (int, null)
CreationTime (datetime2(7), null)
Keys
Constraints
Triggers
Indexes
Statistics
Sales.OrdersArchive
Columns
OrderID (int, null)
ProductID (int, null)
CustomerID (int, null)
SalesPersonID (int, null)
OrderDate (date, null)
ShipDate (date, null)
OrderStatus (varchar(50), null)
ShipAddress (varchar(255), null)
BillAddress (varchar(255), null)
Quantity (int, null)
Sales (int, null)
CreationTime (datetime2(7), null)
Keys
Constraints
Triggers
Indexes
-- Orders data are stored in separte tables (Orders
-- Combine all orders data into one report without
SELECT
*
FROM Sales.Orders
UNION
SELECT
*
FROM Sales.OrdersArchive
Results
Messages
OrderID ProductID CustomerID SalesPersonID OrderDate ShipDate
10 5 104 2 5 2024-05-01 2024-05-
11 5 104 2 5 2025-02-01 2025-02-
12 6 101 3 5 2024-05-05 2024-05-
13 6 104 3 5 2024-05-05 2024-05-
14 6 104 3 5 2024-05-05 2024-05-
15 6 104 3 5 2025-02-05 2025-02-
16 7 102 1 1 2025-02-15 2025-02-
17 7 102 3 5 2024-06-15 2024-06-
18 8 101 4 3 2025-02-18 2025-02-
19 9 101 2 3 2025-03-10 2025-03-
20 10 102 3 5 2025-03-15 2025-03-
Query executed successfully.

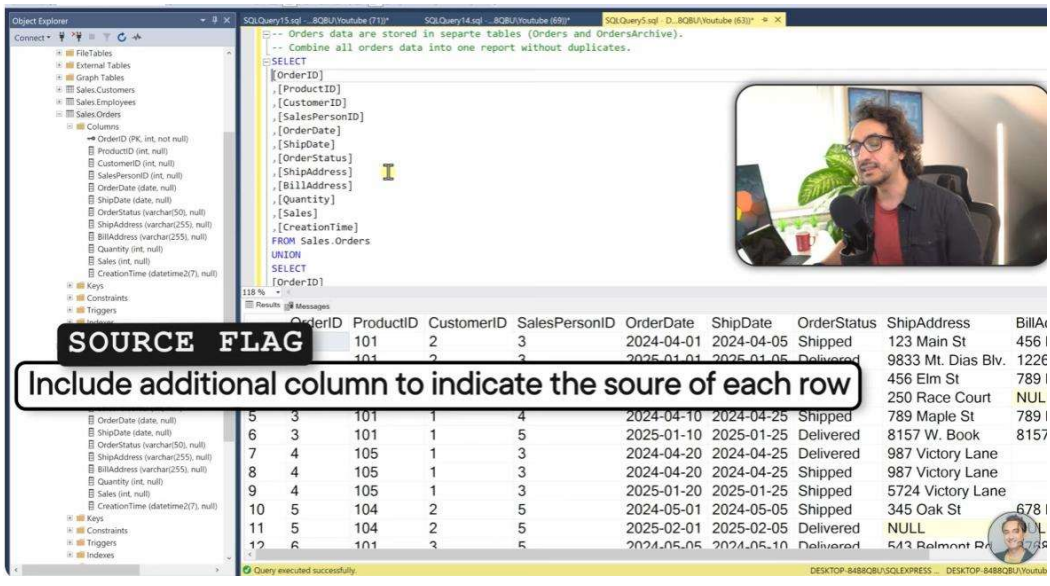


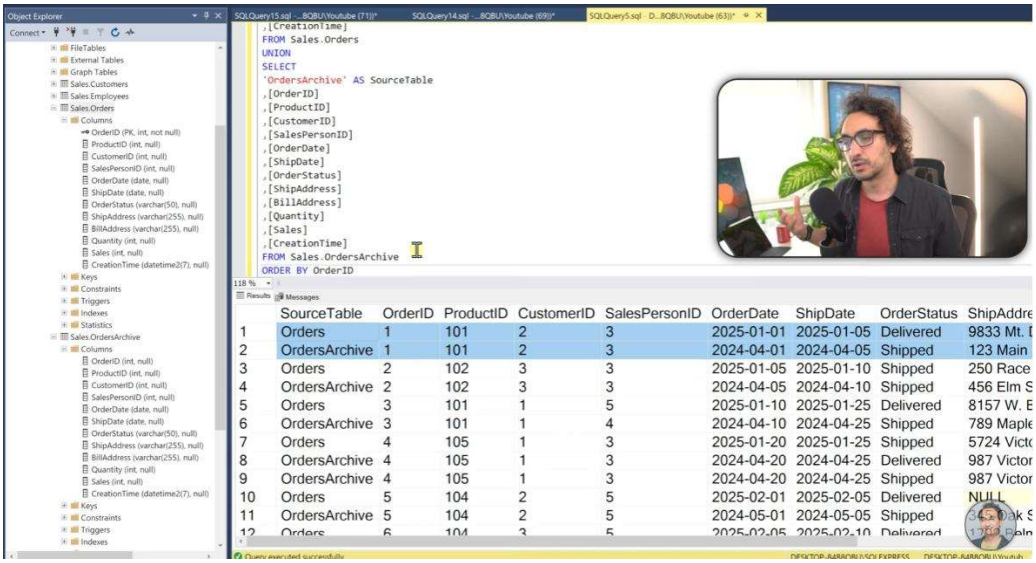


Incorrect data by changing orders of columns so column order matters and we have to write output column name manually its’s a good practice


-- Orders data are stored in separte tables (Orders and OrdersArchive).
-- Combine all orders data into one report without duplicates.
SELECT
[ProductID]
,[OrderID]
,[CustomerID]
,[SalesPersonID]
,[OrderDate]
,[ShipDate]
,[OrderStatus]
,[ShipAddress]
,[BillAddress]
,[Quantity]
,[Sales]
,[CreationTime]
FROM Sales.Orders
UNION
SELECT
[OrderID]
,[ProductID]
,[CustomerID]
,[SalesPersonID]
,[OrderDate]
,[ShipDate]
,[OrderStatus]
,[ShipAddress]
,[BillAddress]
,[Quantity]
,[Sales]
,[CreationTime]
FROM Sales.Orders
UNION
SELECT
[OrderID]
,[ProductID]
,[CustomerID]
,[SalesPersonID]
,[OrderDate]
,[ShipDate]
,[OrderStatus]
,[ShipAddress]
ProductID OrderID CustomerID SalesPersonID OrderDate ShipDate OrderStatus ShipAddress
10 7 102 3 5 2024-06-15 2024-06-20 Shipped 111 Main St
11 101 1 2 3 2025-01-01 2025-01-05 Delivered 9833 Mt. Dias ...
12 101 3 1 5 2025-01-10 2025-01-25 Delivered 8157 W. Book
13 101 8 4 3 2025-02-18 2025-02-27 Shipped 2947 Vine Lane
14 101 9 2 3 2025-03-10 2025-03-15 Shipped 3768 Door Way
15 102 2 3 3 2025-01-05 2025-01-10 Shipped 250 Race Court NULL
16 102 7 1 1 2025-02-15 2025-02-27 Delivered 136 Balboa Co...
17 102 10 3 5 2025-03-15 2025-03-20 Shipped NULL NULL
18 104 5 2 5 2025-02-01 2025-02-05 Delivered NULL NULL
19 104 6 3 5 2025-02-05 2025-02-10 Delivered 1792 Belmont ... NULL
20 105 4 1 3 2025-01-20 2025-01-25 Shipped 5724 Victory L...
1 1 101 2 3 2024-04-01 2024-04-05 Shipped 123 Main St
2 2 102 3 3 2024-04-05 2024-04-10 Shipped 456 Elm St
3 3 101 1 4 2024-04-10 2024-04-25 Shipped 789 Maple St
4 4 105 1 3 2024-04-20 2024-04-25 Delivered 987 Victory Lane
5 4 105 1 3 2024-04-20 2024-04-25 Shipped 987 Victory Lane
6 5 104 2 5 2024-05-01 2024-05-05 Shipped 345 Oak St

Another way

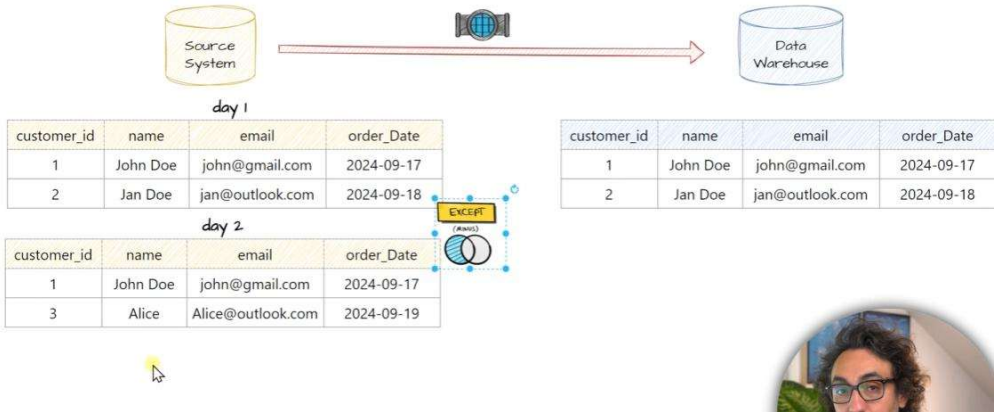

-- Orders data are stored in separte tables (Orders and OrdersArchive).
-- Combine all orders data into one report without duplicates.
SELECT
[OrderID]
,[ProductID]
,[CustomerID]
,[SalesPersonID]
,[OrderDate]
,[ShipDate]
,[OrderStatus]
,[ShipAddress]
,[BillAddress]
,[Quantity]
,[Sales]
,[CreationTime]
FROM Sales.Orders
UNION
SELECT
[OrderID]
SOURCE FLAG
Include additional column to indicate the soure of each row
OrderID ProductID CustomerID SalesPersonID OrderDate ShipDate OrderStatus ShipAddress
1 101 2 3 2024-04-01 2024-04-05 Shipped 123 Main St
3 101 1 4 2024-04-10 2024-04-25 Shipped 789 Maple St
6 3 101 1 5 2025-01-10 2025-01-25 Delivered 8157 W. Book
7 4 105 1 3 2024-04-20 2024-04-25 Delivered 987 Victory Lane
8 4 105 1 3 2024-04-20 2024-04-25 Shipped 987 Victory Lane
9 4 105 1 3 2025-01-20 2025-01-25 Shipped 5724 Victory Lane
10 5 104 2 5 2024-05-01 2024-05-05 Shipped 345 Oak St
11 5 104 2 5 2025-02-01 2025-02-05 Delivered NULL

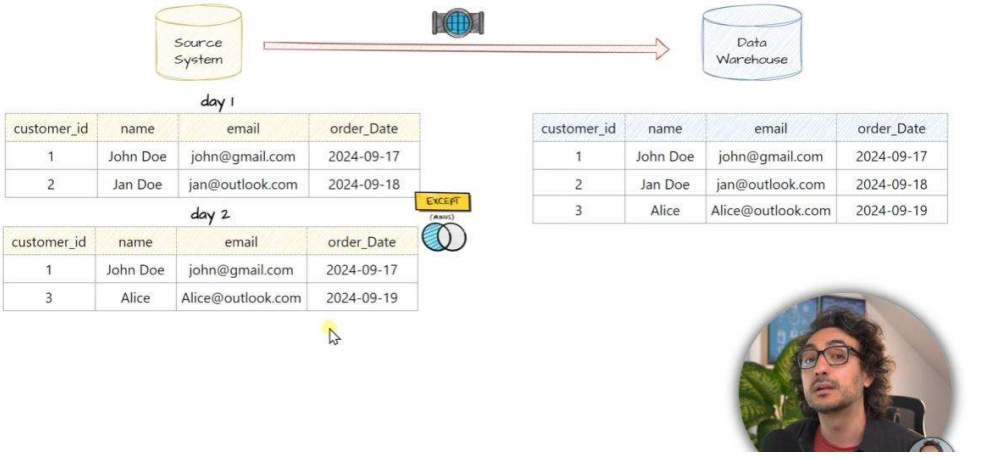


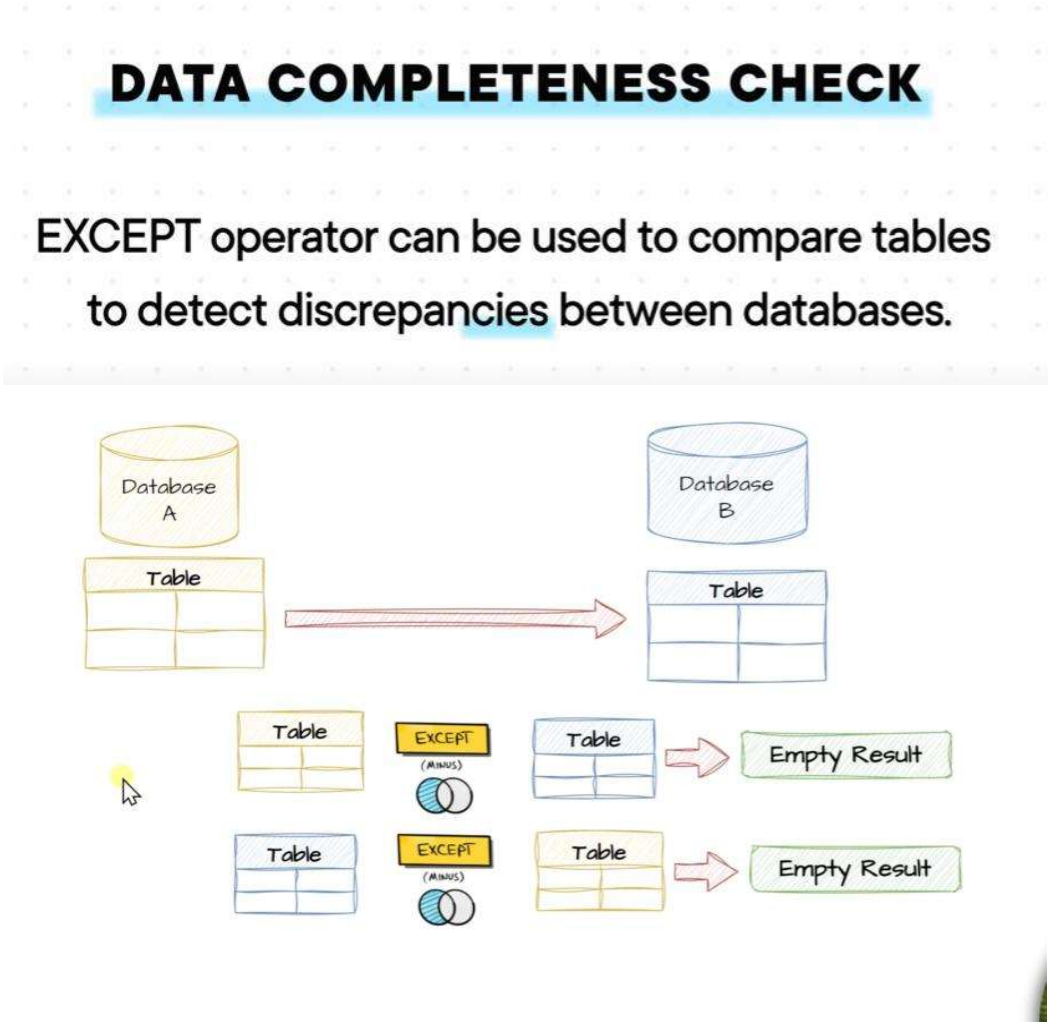
Example for data engineer -

DELTA DETECTION

Identifying the differences or changes (delta) between two batches of data.







Improve data Migration quality