

SQL - Join.

Recap:

- Sub-queries.
- Joins
- if statement
-

Agenda:

- brush up joins.
- Self join.

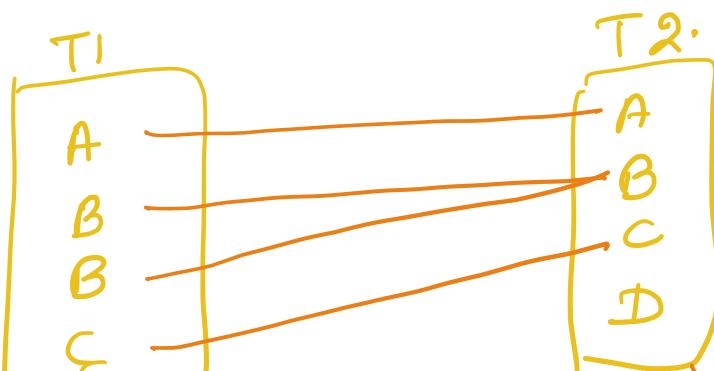
What is Joins?

Trans. table	
Tr-id	Cus-id
T1	A
T2	B

Cus. table	
Cus-id	Name
A	Alan
B	John
C	Mike
D	Donald

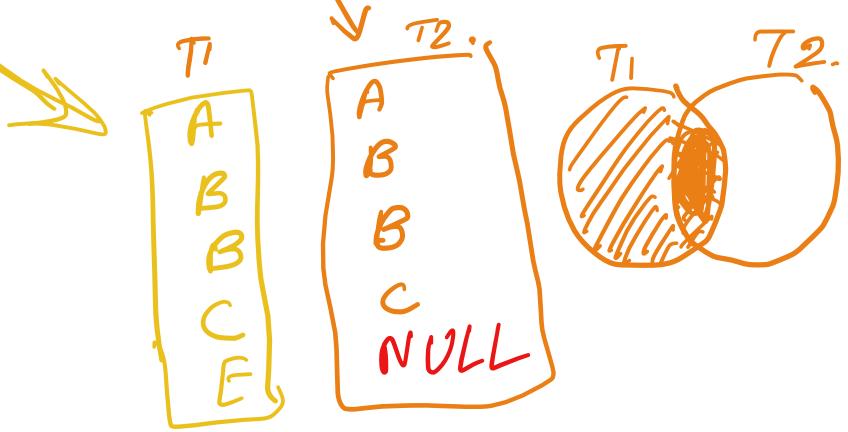
Join.

Tr-id	Cus-id	Name
T1	A	Alan
T2	B	John.



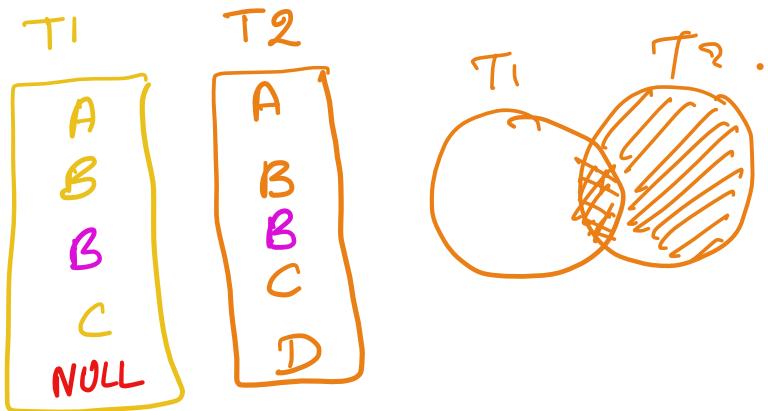
Left Join:

All rows from left table and only matching from right



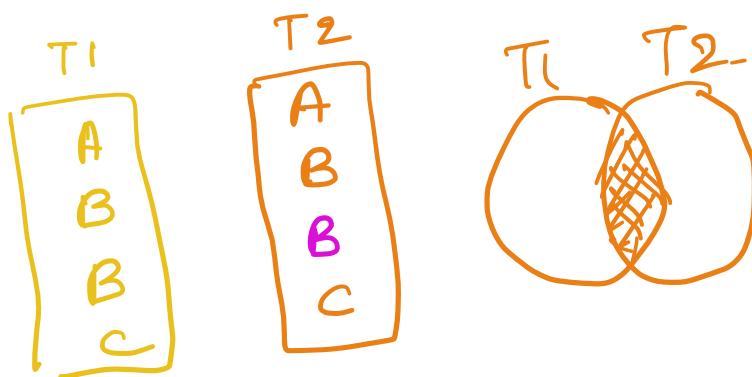
Right Join:

All rows in right tbl and only matching in the left table.



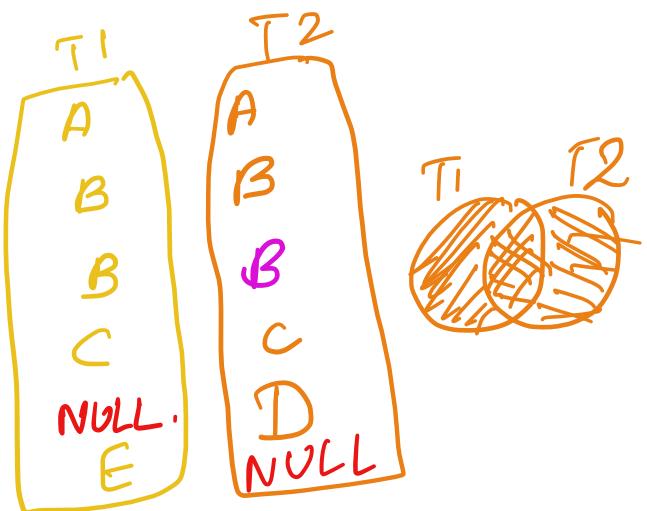
Inner Join

Only matching rows in both table.



Outer Join

- ✓ All rows matching
- ✓ All rows in left
- ✓ All rows in right



Syntax:

Select * from [DB.Tbl 1] → left table

• inner join.

[DB. Tbl 2] → right table

ON [Tbl1. cus-id] = [tbl2. cus-id]

↳ common col between
the 2 tables.

Select [columns to return]

from DB. [left table]

[type of join]

DB. [right table]

ON [lefttbl].[col] = [Righttbl].[col]

Distinct: Used to find the unique rows.

Select distinct column/columns
from DB.Tbl.

To find customers who has not made a purchase

Find customers who have not made purchase.

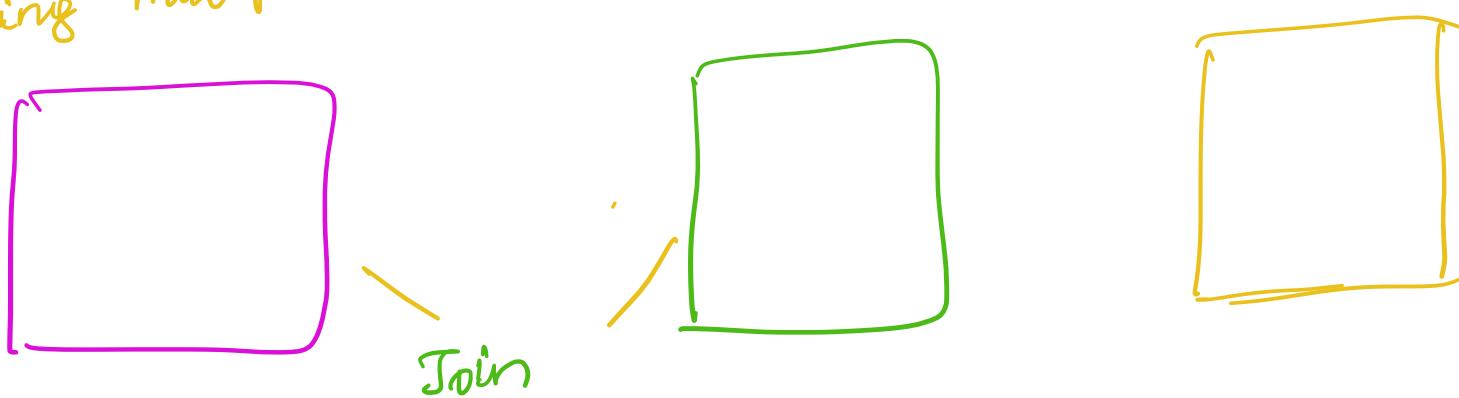
Purchase -		
Trnid	Cusid	Amt
	C10	10
	C14	20
	C8	12

Cus tbl.	
Cusid	Name
C1	Thanish
C10	Alan
C14	John
C8	Murphy

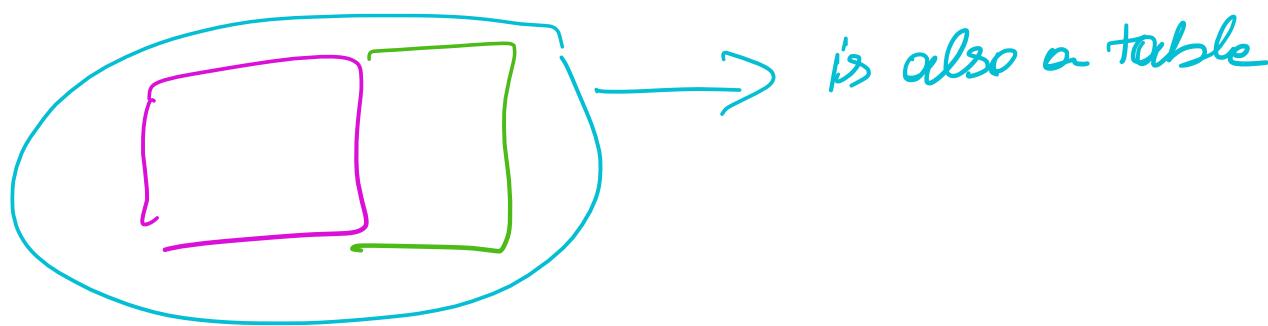
Cusid	Name	Amount
C1	Thanish	NULL
C10	Alan	10
C14	John	20
C8	Murphy	12

this is the customer who has not made a purchase at all

Joining multiple tables.

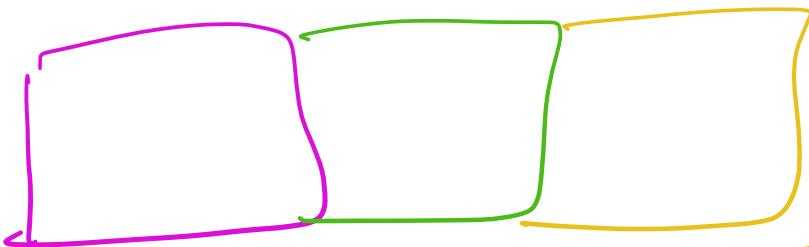


When we join 2 tables what is the output? O/p is another table.



is also a table to the output

So we can join more tables ..



Home work

T1

id	...
1	
1	
2	
2	
NULL	

T2.

id
1
2
3
NULL

- innerjoin
- outer
- left
- Right

Self Join:

Give the name of the manager for each employee

Emp-id	Name	Manager-id
1	John	2
2	Wick	4
3	Bryan	4
4	Jessy	NULL

↓ output

Emp-id	Name	Manager-id	Manager name
1	John	2	Wick.
2	Wick	4	Jessy.
3	Bryan	4	Jessy.
4	NULL	NULL	NULL.



How to get this O/P?

employee table : T1

Emp-id	Name	Manager-id
1	John	2
2	Wick	4
3	Bryan	4
4	Jessy	NULL.

employee Table: T2 .

Emp-id	Name	Manager-id
1	John	2
2	Wick	4
3	Bryan	4
4	Jessy	NULL.

T1 & T2 are same tables.

employee table : T1

Emp-id	Name	Manager-id
1	John	2
2	Wick	4
3	Bryan	4
4	Jessy	NULL.

employee Table: T2 .

Emp-id	Name	Manager-id
1	John	2
2	Wick	4
3	Bryan	4
4	Jessy	NULL.

Select
 → T1.emp-id
 → T1.Name as emp-name
 → T1.manager-id.
 T2.Name as manager-name

from DB.employee_tbl as T1
 inner join.

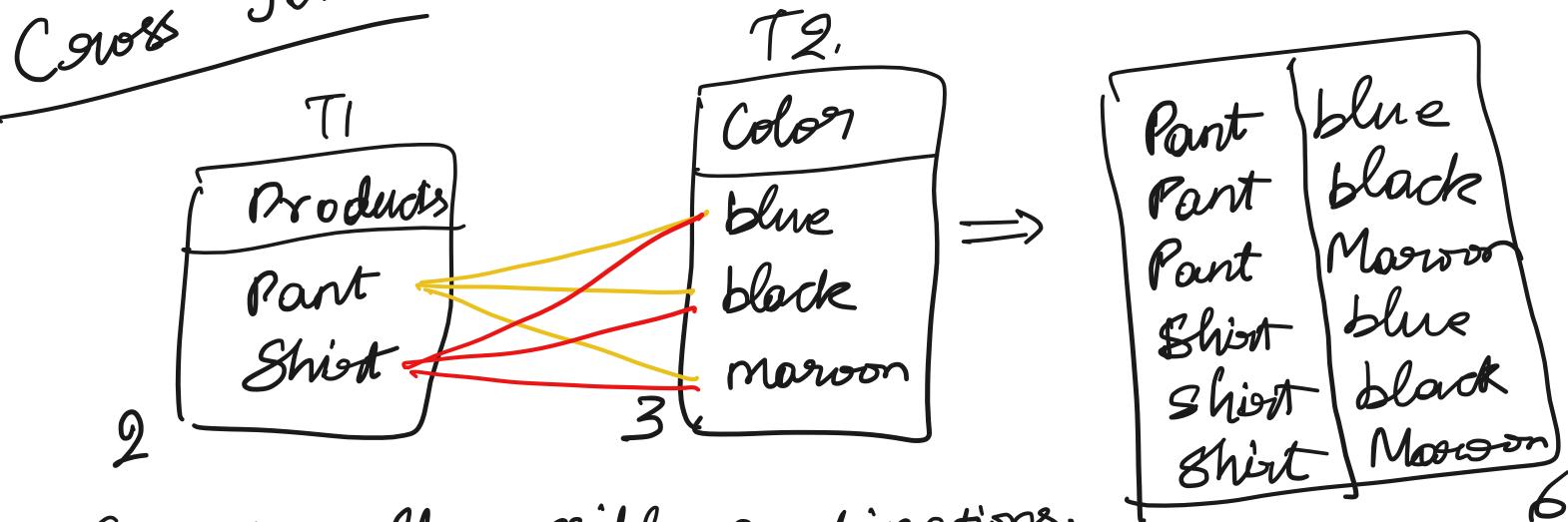
DB.employee_tbl as T2.

ON T1.Manager_id

= T2.emp-id

man_id = emp_id

Cross Join: also called Cartesian joins.



Create all possible combinations.

table 1 table 2 $O(P)$
8 row 3 \Rightarrow 24 rows.

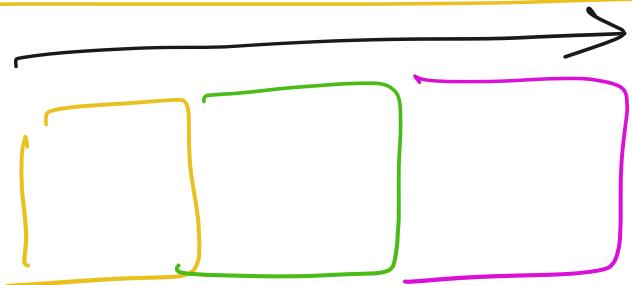
Select $t1.* , t2.*$

from T1

cross join

T2.

Joins:



horizontal
growth.

India Sales:

Prod	Sales
iphone14	20
iphon18	80

US sales

Prod	sales.
Mac	100
iphonels	80
iphonel4	8.

Union

India Sales:

Prod	Sales
iphone14	20
iphon18	80

?

Mac
iphonels
iphonel4

100
80
8.

Union → vertical growth.

- Make sure the columns are logically matching.
- they should same data type

Prod	Sales	Date

Prod	Country	Sales

X This is wrong.

Syntax :

(Select col1, col2, col3
from DB.Tbl 1)

Union - distinct

(Select col1, col2, col3
from DB.Tbl 2)

Union-distinct : Duplicates
will be removed.

union-all : Duplicates
will not be removed.

Joining based on many columns.

Syntax

Select * from

DB.table1 T1 left join

DB.table2 T2.

on

T1.col1 = T1.col1 and
T1.col2 = T2.col2

col1 = vendor-id
col2 = Market
date

