

## SQL 02 - Filtering and Subqueries.

### Recap:

- Strings.
- Order by
- limit
- offset
- Substr, initcap
- Concat.
- round, ceil, floor.
- inline calc.

### Agenda:

- Filters. (where)
- Subqueries.

Where: It is used to filter rows.

```
Select *  
from DB-tbl  
where  
  Cus-name = "Raja"
```

Give me all columns  
⇒ and only rows  
where the customer  
name is Raja.

Cus-Name	Age	
Raja	10	T

Select \* from  
DB-tbl where

Raja	12	= Raja	F
Harri	16	= Raja	F
Mae	15	= Raja	T
Raja	18	= Raja	F



Cus_Name	Age
Raja	10
Raja	15

cus-name = "Raja"  
 ↓  
 exact string match.  
 case sensitive

Position of Where clause?

- Select, order by, limit, offset, from.

Sequence  
of  
writing

Select, columns  
 from  
where  
 order by  
 limit  
 offset

give me prod-id between

1 and 4 (exclude) and also 6

Prod-id	> 1	< 4	AND	= 6	OR.
1	F	T	F	F	F
2	T	T	T	F	T
3	T	T	T	F	T
4	F	F	F	F	F
5	F	F	F	F	F
6	F	F	F	T	F

2	T	F	F	F	F
4	T	F	F	F	F
5	T	F	F	F	F
6	T	F	F	T	T

$T \text{ or } T = T$ $T \text{ or } F = T$ $F \text{ or } T = T$ $F \text{ or } F = F$	$1 \text{ or } 1 = 1$ $1 \text{ or } 0 = 1$ $0 \text{ or } 1 = 1$ $0 \text{ or } 0 = 0$	$\text{True And True} = T$ $\text{True And False} = F$ $F \text{ And } T = F$ $F \text{ And } F = F$	$1 \text{ and } 1 = 1$ $1 \text{ and } 0 = 0$ $0 \text{ and } 1 = 0$ $0 \text{ and } 0 = 0$
--	--	---	--

Prod-id > 3 → give all above 3

Prod-id >= 3 → give all equal to 3 and also above 3

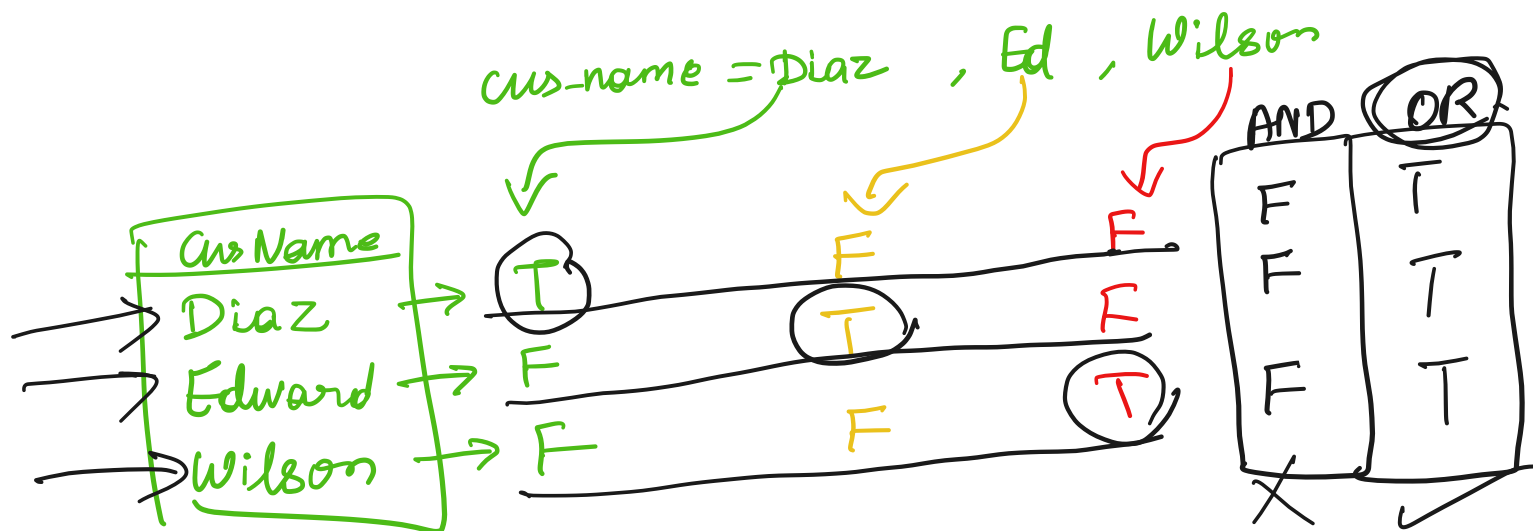
When selecting with ranges: "Between"

select \* from DB.tbl  
where product-id between 3 and 8

Same as:

select \* from DB.tbl  
where product-id >= 3 and  
product-id <= 8.

Select customers "Diaz", "Edward", "Wilson"  
 - Should it be "AND" or "OR"?



IN clause

Select \* from DB.tbl  
 where customername  
 IN ('Diaz', 'Edward', 'Wilson')

is  
same  
as

Select \* from DB.tbl  
 where  
 customername = "Diaz"  
 or  
 customername = "Edward"  
 or  
 customername = "Wilson"

Terry,  
 Jeremiah  
 Jerry

Get all records of a  
 possible customer names  
 starting with Jer

X in ("Jerry", "Jeremiah", "Jerry")

## Wildcard:

It searches based on partial match.

"Jer %" → First 3 chars should be "Jer" but the overall length can be anything also can be any char.

"% Jer" → should end with "Jer" but can start with anything and any length

"% Jer %" → it should contain "Jer" but can start and end with anything and any length

⇒ Jerome.

"Jer   " → Means should be only 4 character and 4<sup>th</sup> one can be anything.

"Jer      " → strictly 5 chars and the last 2 can be anything.

"Jer    %" → Should start with "Jer", Should be minimum 4 character 4<sup>th</sup> can be anything.

"A % O" → Start with 'A', end with 'O' with anything, any length btw.

To use wildcard we use the Clause LIKE

NULL - means absence of data.

" " , ""  $\rightarrow$  empty  $\Rightarrow$  no value, i.e. could be space

When you want to filter Null

↑ is NULL

$\text{trim}(\text{"__Thanish__"}) = \text{"Thanish"}$

$\text{trim}(\text{"_ "}) = \text{" "}$

$\text{trim}(\text{"_ _ _ _ _"}) = \text{" "}$

$\text{trim}(\text{"Thanish_ "}) = \text{"Thanish"}$

Case statement:

Pur Amt	cus bucket
3\$	0-5\$
8\$	6-10\$
19\$	16-20\$

It creates a new column with values based on specified condition.

Case when amount  $< 5$   
then "0-5\$"

when amount between 6 and 10

then "6 - 10\$" ←  
 when amount between 11 and 20  
 then "15 - 20\$" ←  
 Else "above 20\$"  
 End.

Doubt session

trim("Tharish Batcho")

NO, this will not be removed.

Country  
 India  
 USA

substr(country, 1, 2) → IN. 1  
 US.      India  
                  ↑↑  
                  USA  
                  ↑

substr(Tharish, 5, 2).      -5, -2  
                  ↑  
                  Start      AN's.

tharish → thaNish  
                  ↑