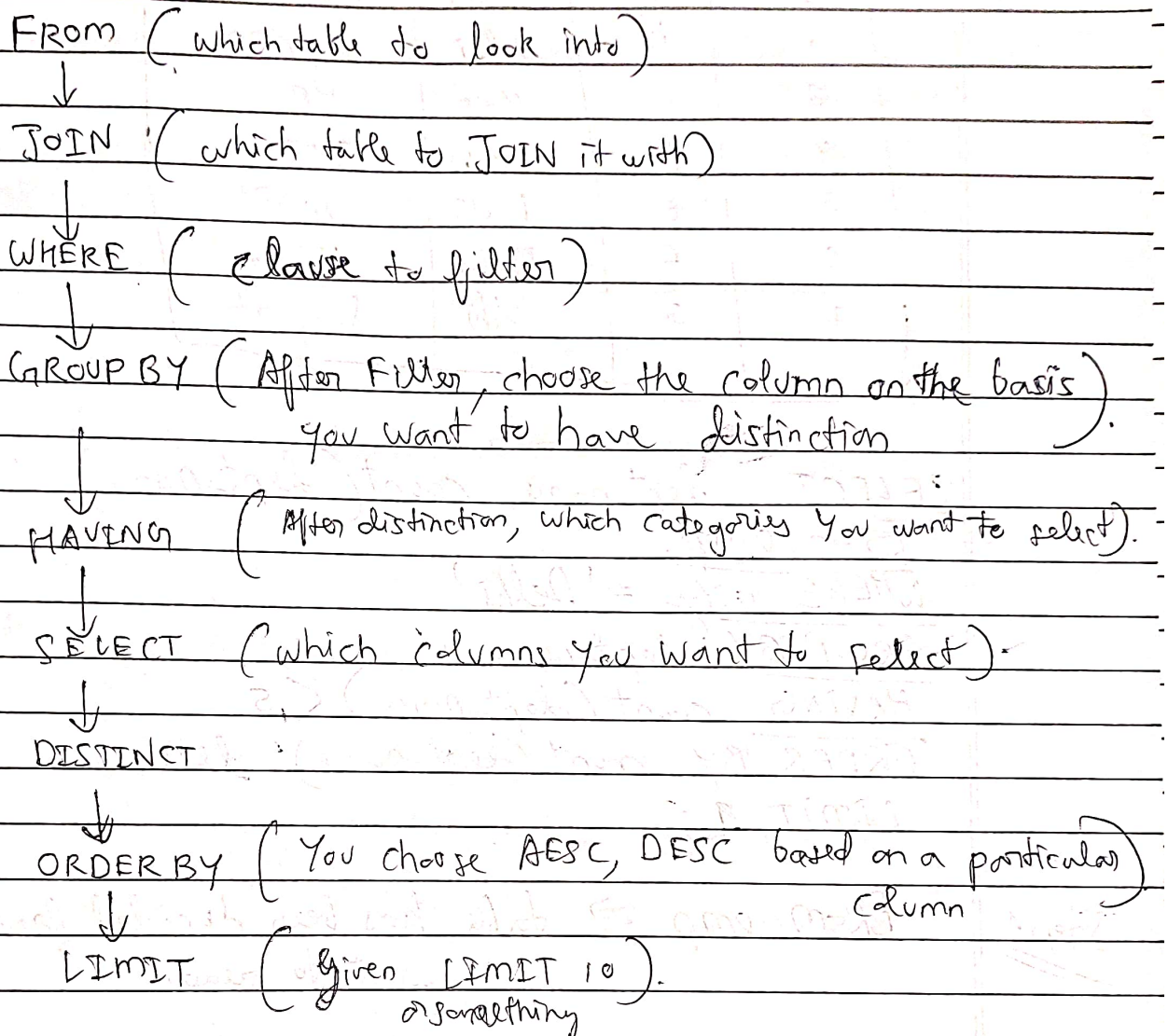
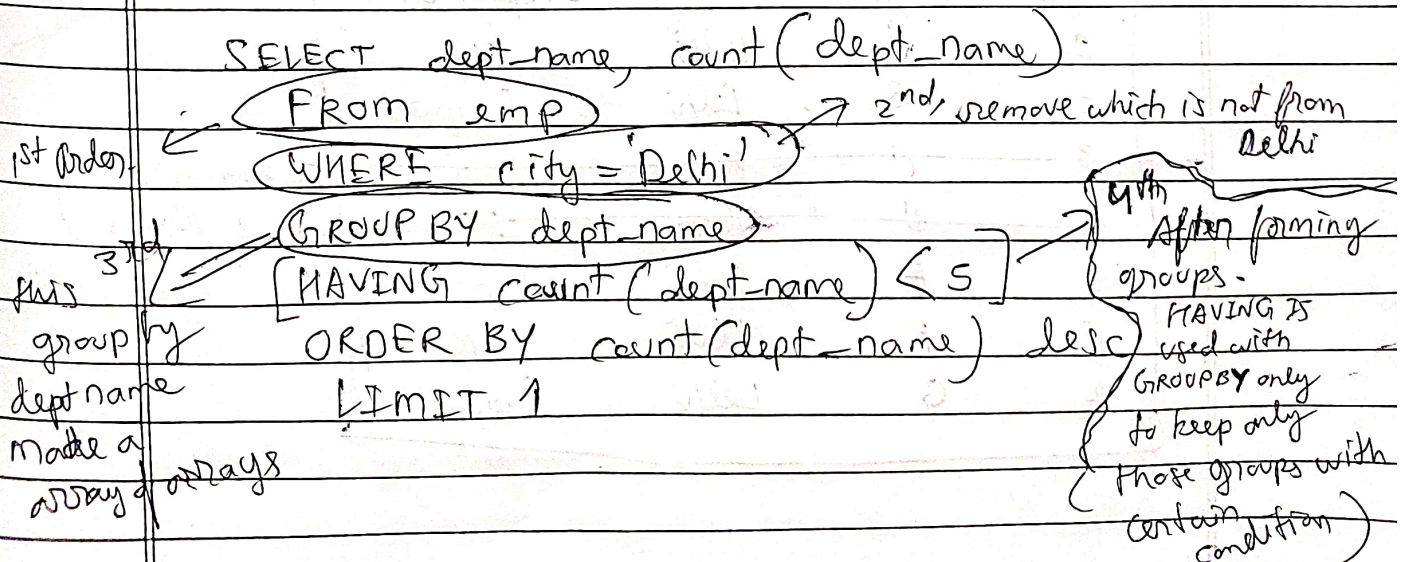


## Order of Execution in SQL



### SQL Example



Table

E-Id	name	city	dept-name
1	A	Delhi	HR
2	B	Delhi	Support
3	C	Kota	HR
4	D	Agra	IT
5	E	Delhi	Support
6	F	Delhi	HR
7	G	Delhi	Support

```

SELECT dept-name, count(dept-name)
FROM emp.
WHERE city = 'Delhi'
GROUP BY dept-name
HAVING count(dept-name) < 5
ORDER BY count(dept-name) desc
LIMIT 4;

```

Step 1 From emp  $\Rightarrow$  table has been decided from which you want to read.

Step 2 where city = 'Delhi': (Now table changes)

EId	name	city	dept-name
1	A	Delhi	HR
2	B	Delhi	Support
3	C	Kota	HR
4	D	Agra	IT
5	E	Delhi	Support
6	F	Delhi	HR
7	G	Delhi	Support

(Removed  
2 Rows)



Step 3)

GROUP BY dept\_name | This works like  
Key  $\rightarrow$  List Map ]

HR  $\rightarrow$  [ (1, A, Delhi, HR)  
(6, F, Delhi, HR) ]

Support  $\rightarrow$  [ (2, B, Delhi, Support)  
(5, E, Delhi, Support)  
(7, G, Delhi, Support) ]

Step 4

Having count (dept\_name) < 5

(HR  $\rightarrow$  2, Support  $\rightarrow$  3) Both remain.

Step 5

SELECT dept\_name, count (dept\_name)

(HR, 2)  
(Support, 3)

Step 6

Order By count (dept\_name) desc

(Support, 3)  
(HR, 2)

Step 7)

LIMIT 1

$\rightarrow$  (Support, 3)

$\rightarrow$  this is the order

Leetcode problem

Person table:

id	email
1	a@b.com
2	c@d.com
3	a@b.com

I have to print  
duplicate emails.

→ a@b.com

I can Group By emails. and only have those  
whose count(email) > 1. That's all

```
SELECT p.email FROM PERSON p
GROUP BY (p.email)
HAVING COUNT(p.email) > 1
```

⊗ (Order of Execution)

1) GROUP BY (p.email).

a@b.com →  $\left[ \begin{array}{l} (1, a@b.com) \\ (3, a@b.com) \end{array} \right]$

c@d.com →  $\left[ (2, c@d.com) \right]$

2) HAVING COUNT(p.email) > 1

$\left[ \begin{array}{l} a@b.com \rightarrow 2 \\ c@d.com \rightarrow 1 \end{array} \right] \rightarrow \text{SELECT.}$