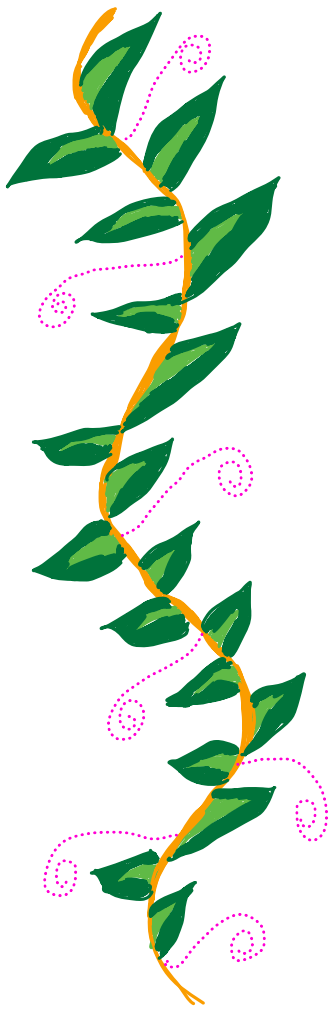
CRUD 2



* Agenda :

  # Read Continued ...

  1. AND , OR , NOT
  2. IN
  3. BETWEEN
  4. LIKE
  5. IS NULL , IS NOT NULL
  6. ORDER BY
  7. LIMIT

Challenge : i) Solve all assignment questions

ii) Clear backlog (session , Assign. , HW)

AND / OR / NOT

↳ To specify a condtn we use WHERE clause.

↳ We have operators in SQL as well.

Ques: Give movies which were released in 2006 & were having a rating of 'PG-13'

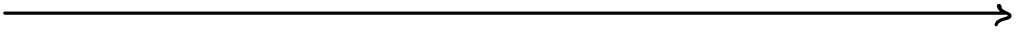
Ques: Give movies released in 2006 & rating not equal to 'PG-13'

Not equal operator

i) !=

ii) not

iii) <>



IN Operator :

→ Give me all the students with batch_id either 5, 2, 1, 7, 10.

```
Select *
from students
where student_id = 5
    or student_id = 2
    or student_id = 1
    or student_id = 4 ;
```

When we want to compare a column with a set of values, use IN operator

---

Between

→ Get all the movies where release_year ≥ 2005 and release_year ≤ 2016

```
Between 2005 and 2016
```

LIKE :

b_id	name
1	Oct_2023_Acad_morn_int
2	Nov_2022_DSML_adv_eve
3	Acad_NOV_2023_adv_morn

← Batches

1. Every Academy batch should have academy in their name.
2. It should have 'Beg', 'intr', 'adv'
3. It should contain 'Morn', 'Eve'

-> We have an operator → Like

We have 2 wildcard symbols in Like :

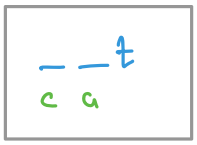
i) '_' : exactly one occurance of any character

ii) '%' : any no. of occurance of any no. of character/s.
Or No character as well.

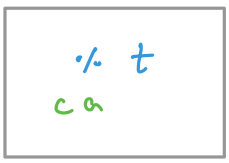
Given String

Pattern

cat



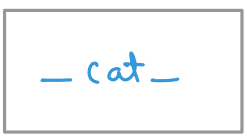
✓



✓✓

✓✓

✓✓



✗

```
Select  *
from    batches
where   name  like  '% Mor %'
```

→ % mom _ acad % ✗

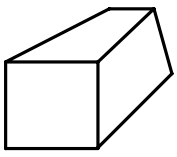
→ % mom% % Acad % ✗

→ name like '% mom %' and
name like '% Acad %'

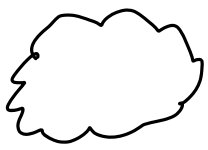
* **Null** (IS Null , IS NOT NULL)

S_id	name	Status
		Null

⇒ Null = Null ✗



empty



empty

S_id	name	Status
1	Rahul	Null
2	Tarun	1
3	Mahyesh	2

```
select *
from students
where status != 2 or
      status is Null;
```

Order By :

→ Here data is sorted in ascending order by default.

→
```
select *
from film
order by title ;
```

PK

s_id	Year	name
1	1998	D
2	1999	D
3	1998	C

Students

By default tie breaker is PK.

```
order by year, name ;
```

3 1998 C
1 1998 D
2 1999 D

Limit :

100,000

```
select *
from table_name
```

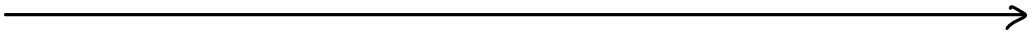
Limit is used to limit the number of rows I want from my query.

To get bottom N rows use order by in desc order

* **syntax :**

```
select *
from table_name
limit N;          # Get top N rows
```

```
select *
from table_name
limit M, N;       # Get N rows after Mth row.
```



Order by pseudo code

```
table = [ ], [ ]
ans = [ ]
for row in table:
    if row.matches (where condtn)
        ans.add (row)

ans = Sort (according to col given in order by)
```

→ Iterate & print the fields we want.