

* Agenda :

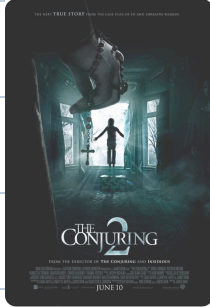
- Like operator
- Working with NULL values
- Problem solving

7th Hand day challenge :

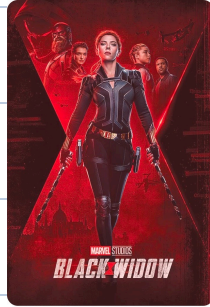
1. Assignments
2. Backlog (Assignments of prev. session)



Like



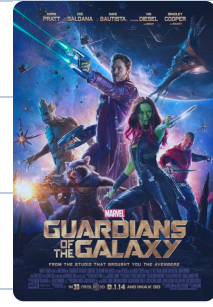
Genre: Horror
Language: English



Genre: Action
Language: English



Genre: Horror
Language: English



Genre: Action
Language: English

- Whenever there is a column storing strings, there comes a requirement to do some kind of pattern matching.

Ex : Assume Scaler's databases where we have following rules to store batch name :

1. Every batch name should have 'Academy' / 'DSML' in them.
2. It should have 'Beg' / 'Inter' / 'Adv' in the naming convention.
3. It should have 'Morn' / 'Eve'.
4. It should have month of the batch.



Batches

b_id	b_name
1	'Acad_Apr_23_Morn_Beg'
2	'June_23_DSML_Inter_Eve'
3	'Mar_23_Acad_Adv_Eve'

Question : Get all the morning beginner batches.

' How to get these batches? '



' We can do pattern matching using LIKE operator ...'





We have two wildcards in **Like** operator :

- Consider them like fill in the blanks :

1. ' **_** ' : Can have exactly one occurrence of a single character.
2. ' **%** ' : Can have any number of character or it can stay empty as

Question : Find all the beginner batches.

How to use wildcards?

1. **% beg %** : Anything can come before beginner and after beginner.

Anywhere in the string

2. **% beg** : Anything can come before beginner. *Beg as suffix.*

3. **beg%** : Anything can come after beginner. *Beg as prefix.*



Let's do some pattern matching.

String

Pattern

Cat

✓ Cat fat, hat, mat, eat

✓ Ca %t elephant, t, at, eat

✓ % anything

✓ % Cat % caterpillar, concatenation

✗ _Cat_

Question : Get data of all the morning batches.

SELECT *

FROM batches

WHERE b_name **like** ' % Morning %' ;



NULL (is null, is not null)

- Do you all remember how we store empties, no value for a particular column for a particular row? We store it as **NULL**, regardless of column's datatype.

- Interestingly working with null's is bit tricky. Let's see how.

Students

s_id	s_name	status
1	A	1
2	B	Null
3	C	0
4	D	1

Question : Get all the students whose status is NULL

Let's try using the following query

```
SELECT * FROM student WHERE status = NULL;
```

- The above query will not return any rows. **WHY?** Because NULL is not equal to NULL. in fact NULL is not equal to anything. Nor is it not equal to anything. It is just NULL.



Try this query and find the output

```
SELECT NULL = NULL
```

- We can't compare **NULL** with anything like we can't compare an Empty Glass with an Empty Brain.



Empty Glass



Empty Brain

- The right operator is **IS NULL**

```
SELECT * FROM student WHERE status IS NULL;
```

- Similarly we don't use not equal to **NULL**, rather we use **IS NOT NULL**, when we want not NULL values.



LIMIT Clause

- LIMIT clause allows us to limit the number of rows returned by a query.
- Suppose we have this query :

```
SELECT * FROM table ;
```

' What if a table has 1-million rows? '



How to get top two rows : `SELECT * FROM table LIMIT 2 ;`

Students Table :

s_id	s_name	name
1	James Jones	1
2	John Miller	2
3	John Martinez	3
4	Michael Garcia	4
5	Jennifer Miller	5

- There's one more thing that we can do : **OFFSET**

OFFSET : 10 row after 100th row