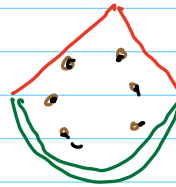


10/05/24

SOL 04 : CRUD-2

AGENDA

- ① LIKE
- ② NULL
- ③ ORDER BY
- ④ LIMIT, OFFSET
- ⑤ UPDATE
- ⑥ DELETE
- ⑦ DELETE, TRUNCATE & DROP.



PATTERN MATCHING

① LIKE (PATTERN MATCHING)

OCT 23

" * ? " " * " ? " (DP)
" * " a * a ?

" % " → Matches 0 or more of any character
- similar to '*'

" _ " (under score) → Match exactly one (any) character.
- similar to '?'

Pattern

'cat'

'cat %'

'%cat'

'%cat %'

'c-t'

'c %t'

- 123 -

%123%

Words

Cat ✓

Catx ✗

xcat ✗

xcatx ✗

Cat ✓

Catx ✓

xcat ✗

xcatx ✗

Cat ✓

Catx ✗

xcat ✓

xcatx ✗

Cat ✓

Catx ✓

xcat ✓

xcatx ✓

Cat ✓

Count ✗

Chat ✗

Ct ✗

Cat ✓

Count ✓

Chat ✓

Ct ✓

'%123%' → ABC123XYZ ✓

'-123-' → ABC123XYZ ✗

↳ 123

MY STOMACH IS EMPTY

≈ =

YOUR STOMACH IS EMPTY

DB empty cells.

-NULL (empty)

NULL = NULL ✗

NULL != NULL ✗

== == == == == ==

ORDER BY (Sort data)

default - Ascending
Descending - DESC.

SELECT * from film?

① ? PK = (asc)

② Sorted by PK - by default.

[1, 2, 2, 5, 9]

DISTINCT, + ORDER BY

SELECT DISTINCT TITLE FROM FILM
ORDER BY RELTR;

name	rel.
PUSHPA	2022
RRR	2023
PUSHPA	2024

PUSHPA 2022
RRR 2023

OR

RRR 2023
PUSHPA 2024

AMBIGUITY

BREAK TILL - 8:14 AM.

LIMIT (count of rows)

SELECT * FROM TABLE
→ LIMIT 1000;

1
2
3, 0
1
2
3

(1001 - 2000)
↑
(2001 - 3000)

OFFSET → Starting point

Start = offset + 1.

fetching (1001 - 2000)

READ!

SELECT * from table
LIMIT 1000 OFFSET 1000 ;

~~== X == X~~ ~~— ≠ X —~~

UPDATE: Update table <table-name>
SET Col = val
WHERE condⁿ(s);

DELETE

DELETE FROM TABLE-NAME
WHERE condⁿ(s);

	DELETE	TRUNCATE	DROP
① Can be recovered	YES	NO	Both data & schema is deleted.
②	Slower.	Faster	<u>NO</u>
③ Can delete specific rows	Can delete specific rows	Cannot delete specific rows.	
④ Auto- inc keys	NOT RESET	<u>RESET!</u>	
⑤ Schema present	YES	YES	<u>NO</u>

logging the info
of each row
that is deleted.

ORDER BY rel-yr, duration, PK

DISTINCT

$\begin{pmatrix} (x, y) \\ -1- \\ (x, y) \end{pmatrix}$



1
2
~~3~~
4
~~5~~
6
7

auto.

count

QUERIES RUN IN CLASS

```
-- SQL 04 CRUD 2  
USE SAKILA;
```

```
SELECT * FROM FILM  
WHERE TITLE LIKE 'LOVE%';
```

```
SELECT * FROM FILM  
WHERE TITLE LIKE '%LOVE';
```

```
SELECT * FROM FILM  
WHERE TITLE LIKE '%LOVE%' AND TITLE LIKE 'ID%';
```

```
SELECT * FROM FILM  
WHERE TITLE LIKE '_USHPA';  
SELECT * FROM FILM  
WHERE TITLE LIKE '_ADAR';
```

```
-- LIST ALL ROWS WHERE EMAIL IS NOT KK@X.COM  
SELECT * FROM STAFF  
WHERE EMAIL != 'KK@X.COM';
```

```
SELECT * FROM STAFF  
WHERE EMAIL NOT IN('KK@X.COM');
```

```
-- IS NULL  
SELECT * FROM STAFF  
WHERE EMAIL != 'KK@X.COM'  
OR EMAIL IS NULL;
```

```
--  
SELECT * FROM STAFF  
WHERE EMAIL IS NOT NULL;
```

```
SELECT * FROM STAFF  
WHERE EMAIL IS NULL;
```

```
SELECT * FROM FILM;
```

```
SELECT * FROM FILM  
WHERE RELEASE_YEAR=2024 ORDER BY RELEASE_YEAR, RENTAL_DURATION;
```

```
SELECT DISTINCT TITLE  
FROM FILM
```

ORDER BY RELEASE_YEAR;

-- Error Code: 3065. Expression #1 of ORDER BY clause is not in
-- SELECT list, references column 'sakila.FILM.release_year'
-- which is not in SELECT list; this is incompatible with DISTINCT

SELECT DISTINCT TITLE, RELEASE_YEAR
FROM FILM
ORDER BY RELEASE_YEAR;

-- FIRST 10 ROWS OF FILM

SELECT * FROM FILM
LIMIT 10;

-- 21-30 ROWS
SELECT * FROM FILM
LIMIT 10 OFFSET 20;

-- LAST 10 ROWS

SELECT * FROM FILM
ORDER BY FILM_ID DESC
LIMIT 10;

-- UPDATE

UPDATE FILM
SET TITLE = 'RRR';

-- UPDATE
UPDATE TABLE_NAME
SET COL=VALUE
WHERE COND(S);

-- DELETE

USE SQL_030524;
INSERT INTO STUDENTS (STUD_NAME, BATCH_ID)
VALUES ('MAYA', 99),
('ARUN', 99);

TRUNCATE TABLE STUDENTS;
DELETE FROM STUDENTS;

DROP TABLE STUDENTS;