

① Window functions

- Aggregate

- Builtin wf

- RANK

- ROW No.

② Stored procedures

③ Query optimisation

- execution lifecycle

- EXPLAIN

- Scans

ORM

↳ WRAPPER

→ clean

db. select ()

• where ()

→ Hibernate | SQL Alchemy | TypeORM

→ JPA

|

)

Window functions

Aggregates | GROUP BY

Select  id

FROM student

Group by batch_id,

id | batch_id

①	1	1	1
②	1	1	1
3	/	/	2
4	/	/	2

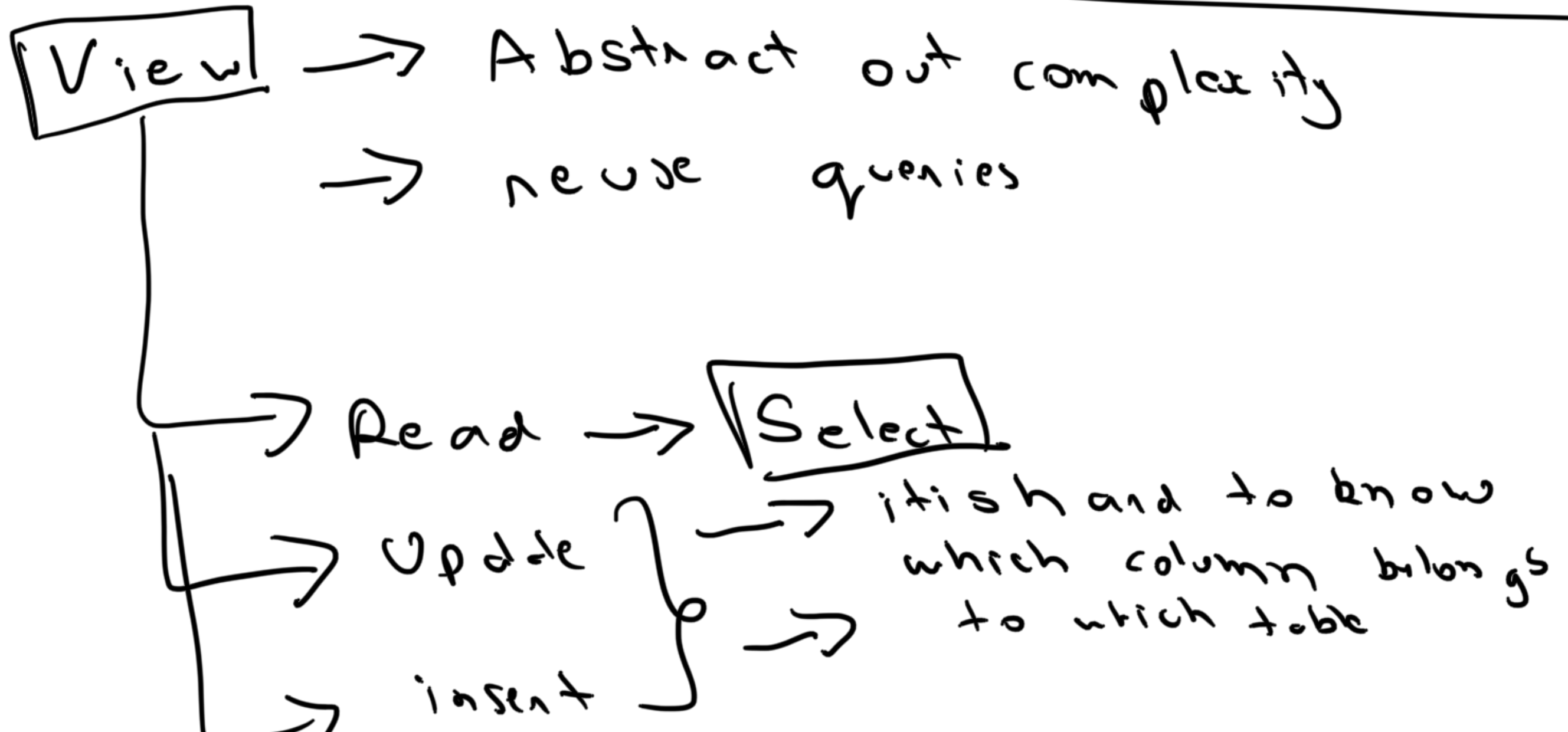
[1, 2]

MySQL 8.0

→ window functions

Print the student names

along with the count
of students in a batch



- Only 1 underlying table
- Mandatory columns view

Abstract out update operations

student

	City
.	Bengaluru
	Bengaluru

→ Bengaluru

→ Bengaluru

Bombay → Mumbai

STORED PROCEDURES

↳ write a set of operations that can be reused

↳ do not return a value

Void

CREATE PROCEDURE 'name' (

Parameters

)

UPDATE

END;

CALL Function Name ()

Compilation → once per session

↓
[Cache]

6:27 - 6:30

11:00

OS

CN

→ Concurrency

→ Synchronization

→ Lifecycle of a packet

→ Structure of networks

- OSI model

→ Threads

→ Locking

→ HTTP / HTTPS

→ TCP

→ Socket - I } Python

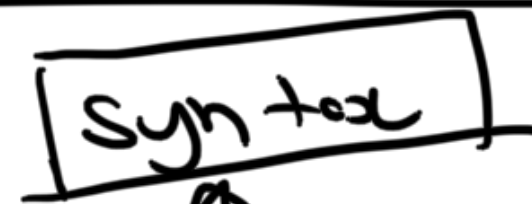
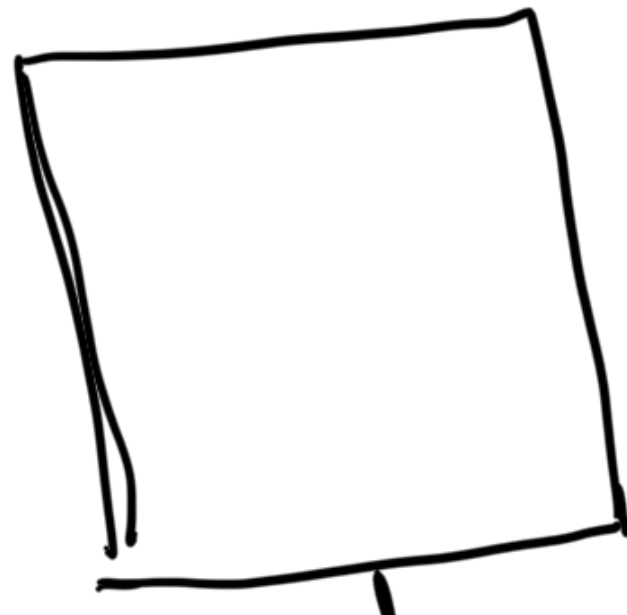
→ Socket - II

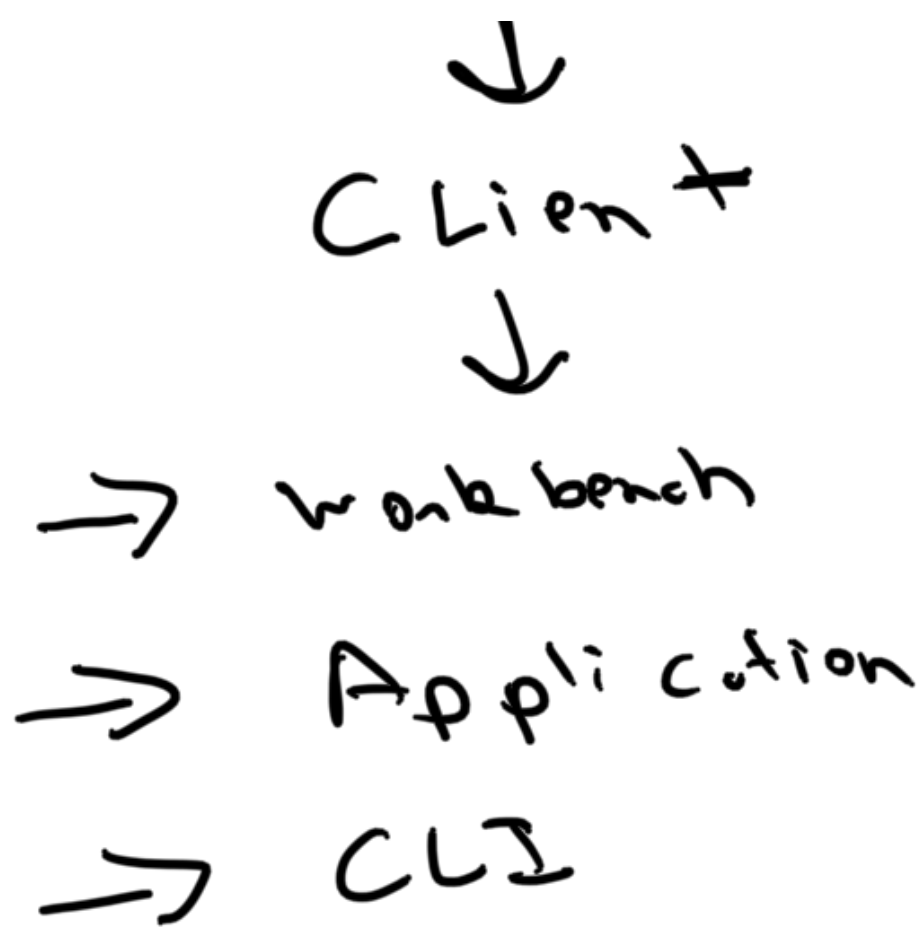
5

5

10

INSERT
ADD INS

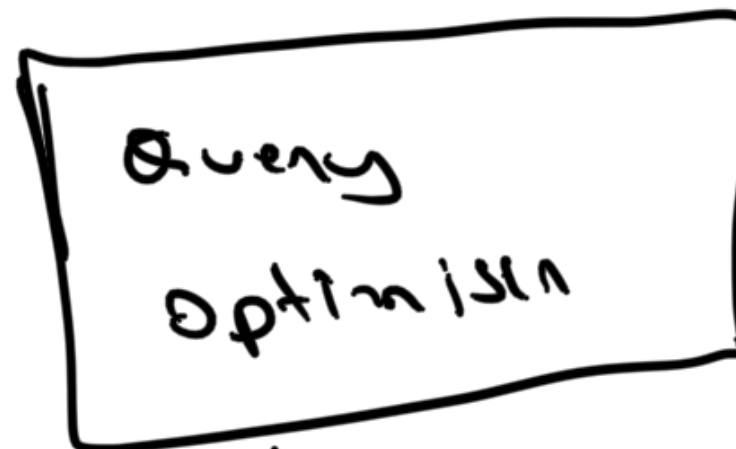
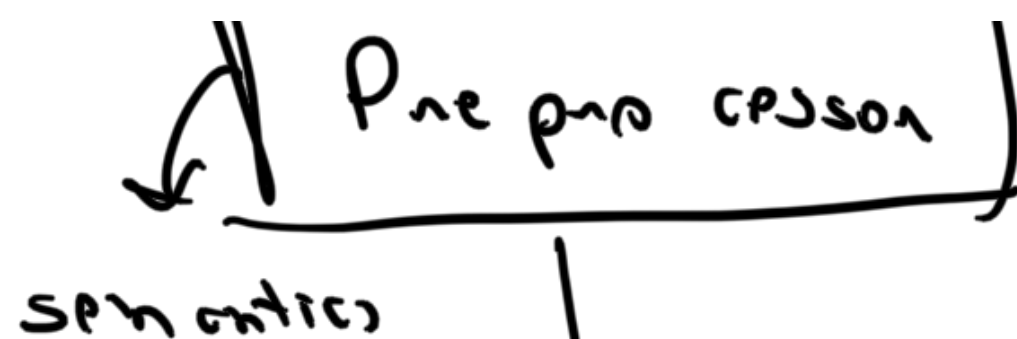




FROM



WHERE



Execution engine

Database engine

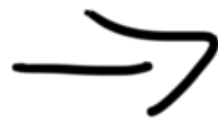
a = ? on b = ? }
→ InnoDB }
→ MySQL }

Execution plan → Cost

→ Cost based optimizer

Optimization Select

A



cost

1.0

time ?

10ms

12ms

B \rightarrow 1.5 || 20 ms

① \rightarrow cost x

Added on index

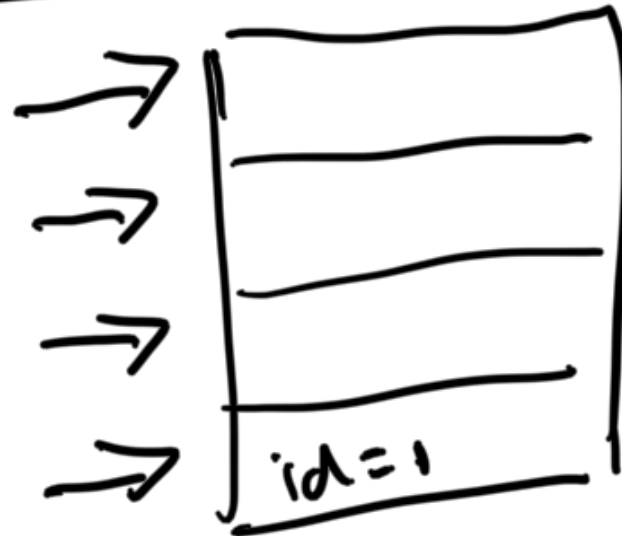
② \rightarrow cost y

y < x

EXPLAIN

ANALYZE

Full table Scan



sequential

WORST

Index

Full Index Scan

Index seek | lookup

