SQL - Joins and Aggregate functions

① – Joins

- Inner Joins
- Outer Joins
- Cross join

② Aggregate functions

- GROUP BY
- Built in functions

Gets all students in batch "Sherlock Season 5"

↓

Students

id | name | batchid

C
R – Select
U
D

FROM

Students

WHERE

Batch

---

2 queries

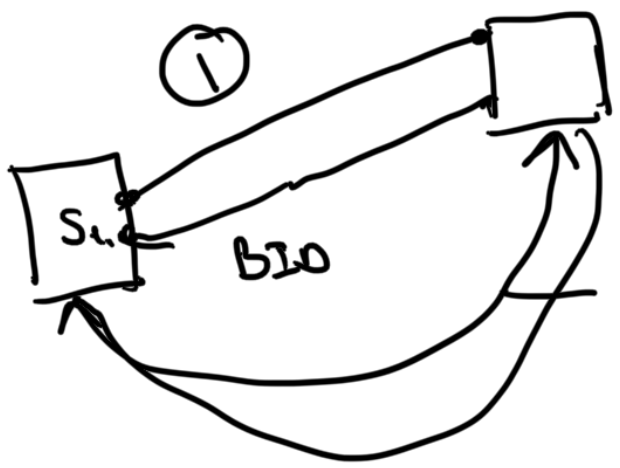
↳ Get all batches with name Sherlock

+ BID

→ Get all Students where BatchId = BID

2 given IDs

↳ Server

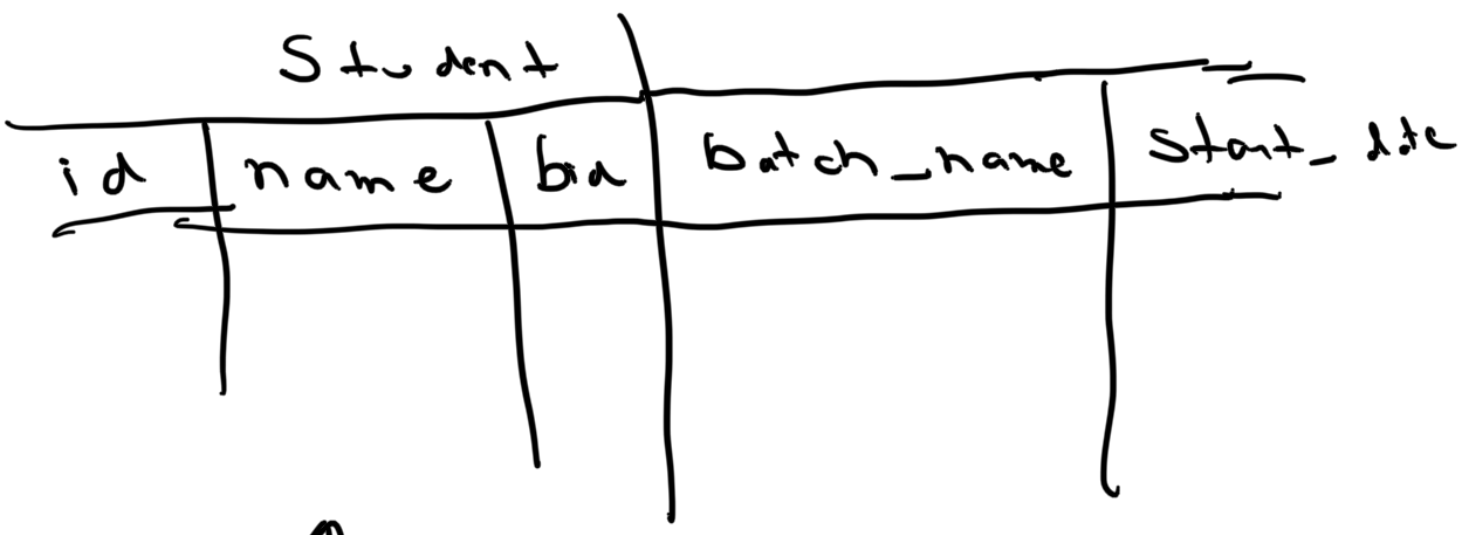


Select student where BID = BID

4 network calls 250 ms x 4

= 1s

Joins



anomalies
↑ normal violation

Denormalising Soo.

Joins

→ whenever I want data from multiple tables

Students id | bid | name

+

Batch b id | name | start_date

S.id	b:id	s.name	b.name	b.start_d.
1	1	John	Sherlock	31/10/

Join s

- Inner Join | Join

Student

id	name	bid
→ 1	John	(1)
→ 2	Thon	(2)

Batch

id	name
(1)	Sherlock
(2)	Love Th underx

S.id	S.name	s.bid	b.id	b.name
→ 1	John	(1)	(1)	Sherlock
→ 2	Thon	(2)	(2)	Love Thun...

SELECT

s.id,

s.name,

```
S.id
S.name

FROM
    students S
      INNER JOIN
    batch B
    ON  S.bid
          = B.id
```

Inner join → only get rows where

bid = S.bid

condition matches

N × N

→ for student in students

→ for batch in batch

if student.bid
= batch.bid

then add it to result

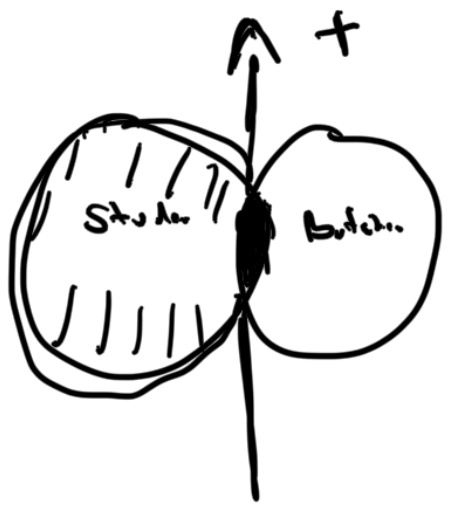
①	
SID	bid
1	3
→ 2	1
3	2
4	4

②	
bid	name
1	Batch1
2	Batch 2
3	Batch 3

1.SID	1.BID	2.BID	2.BNAME
1	3	3	Batch3

2		1		Batch 1
3		2	2	Batch 2
~~4~~				

Outer joins



Inner Join

Students that are batches.

Inner Join

Left outer Join

Sid	bid
1	1
2	2
3	NULL

bid	name
1	B1
2	B2
3	B3

Left table matching right values

SID	BID	BNAME

1	1	B1
2	2	B2
3	NULL	NULL

for student in students:

for batch in batch:

if s.bid == b.id
add to result

if s.bid == NULL
add to result

5:53 - 6:00

10:30 ✓

Students

id	name	buddy_id
1	John	3
2	Thor	NULL
3	Tonia	2

Get all students and their buddy's name

s.id	s.name	buddy name.

→ Self join

↳ Students with Students

Student S1

sid	stu	oid
1	John	3
2	Tom	Null
3	Tenin	2

Students S2

1	John	3
2	Tom	Null
3	Tenin	2

ON

S1.buddyid = S2.sid

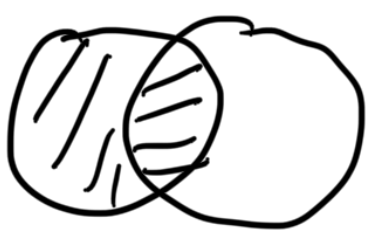
```
SELECT
    S1.first name
    S2.first_name

FROM
    `Students` S1
     JOIN
    ` students` S2

     ON
        S1.buddy_id
```

= S2.id;

SELF - JOIN

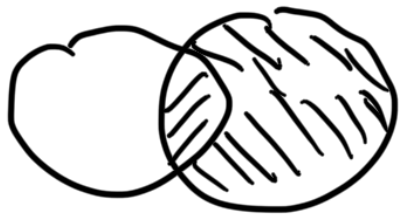
→ Inner Join → only matching rows

→ Left Join → matching + Left



→ Right Join

→ matching + Right



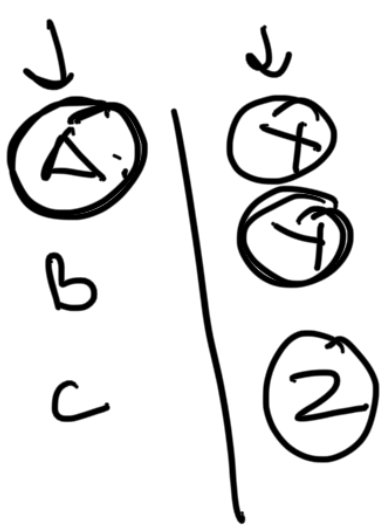
→ Self Join

→ Inner join with same table

Cross join

→ Cartesian product

→ dot product



(A, X), (A, Y), (A, Z)

(B, Y), (B, Z), (B, X)

Table1 × Table2

m × n ⇒ m x n

CROSS JOIN

Select *

FROM

students S

CROSS JOIN

batches b

Where

S.batch_id = b.id;

Inner join

Aggregation

→ Get the student with minimum IQ

Student

id	iq
1	100
2	120
3	90

1

iq
100
120

Select id, iq

FROM students

ORDER BY IQ ASC

LIMIT 1;

id	iq
1	120
2	100
3	60
4	NULL

→ NULLS FIRST | LAST

→ WHERE iq is not null;

Aggregate function

→ rows → single value

Select MIN(iq)
FROM
students;

Aggregate built in function

- MIN
- MAX
- Sum
- COUNT
- AVG