

JOINS.

Students		
id	name	batch-id
1	A	102
2	B	103
3	C	101
4	D	101
5	E	NULL

batches	
id	name
101	B ₁
102	B ₂
103	B ₃

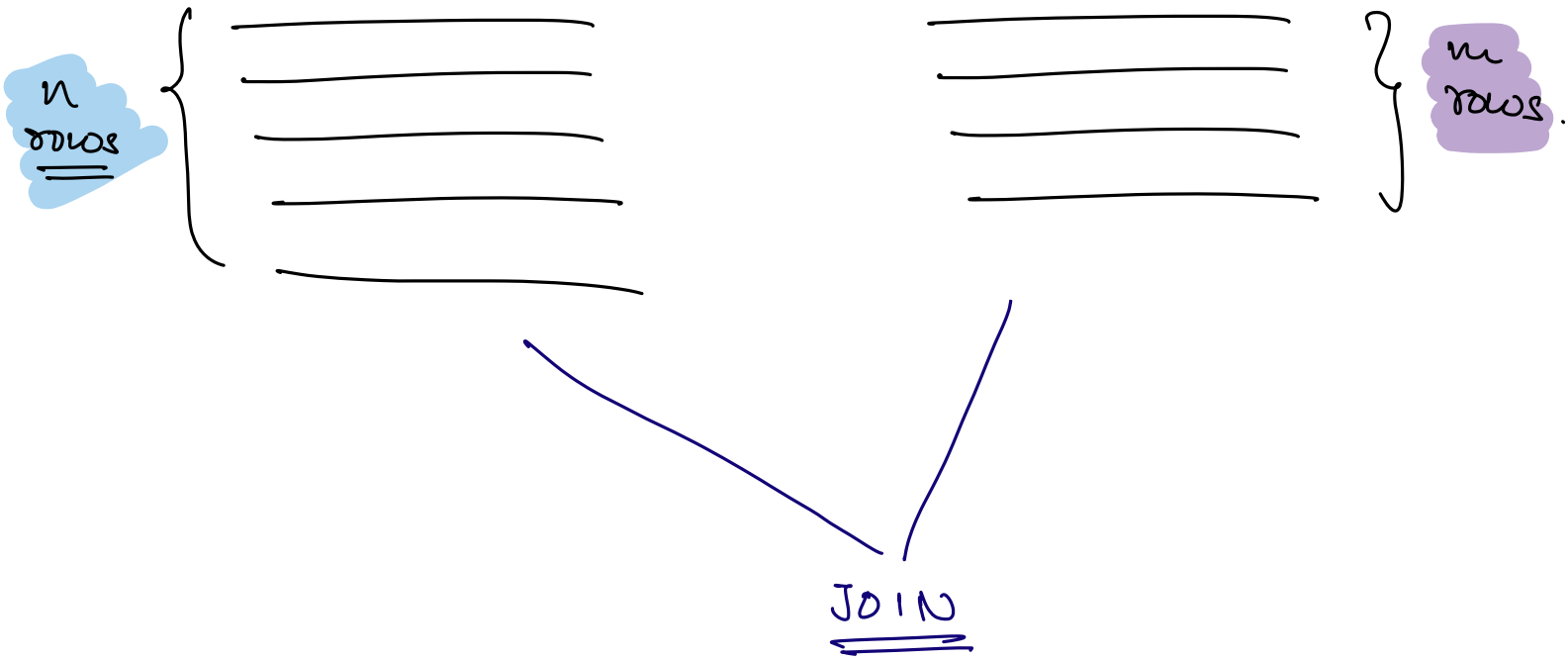
Q: Print the name of students & their batch names.

A	B ₂
B	B ₃
C	B ₁
D	B ₁

id	name	batch-id	id	batch-name
1	A	102	101	B ₁
1	A	102	102	B ₂
1	A	102	103	B ₃
2	B	103	101	B ₁
2	B	103	102	B ₂
2	B	103	103	B ₃

Table-1

Table-2



(N rows in left table) \times (M rows in right table)



$N \times M$

Cross Join

Select *

from students, batches;

\Rightarrow Cross Join

id	name	batch-id	id	batch-name
1	A	102	101	B ₁ ✗
1	A	102	102	B ₂ ✓
1	A	102	103	B ₃ ✗
2	B	103	101	B ₁ ✗
2	B	103	102	B ₂ ✗
2	B	103	103	B ₃ ✓

Select *

from students

Join batches

ON students.batch-id = batches.id;

id	name	batch-id	id	batch-name
1	A	102	102	B ₂
2	B	103	103	B ₃
3	C	101	101	B ₁
4	D	101	101	B ₁

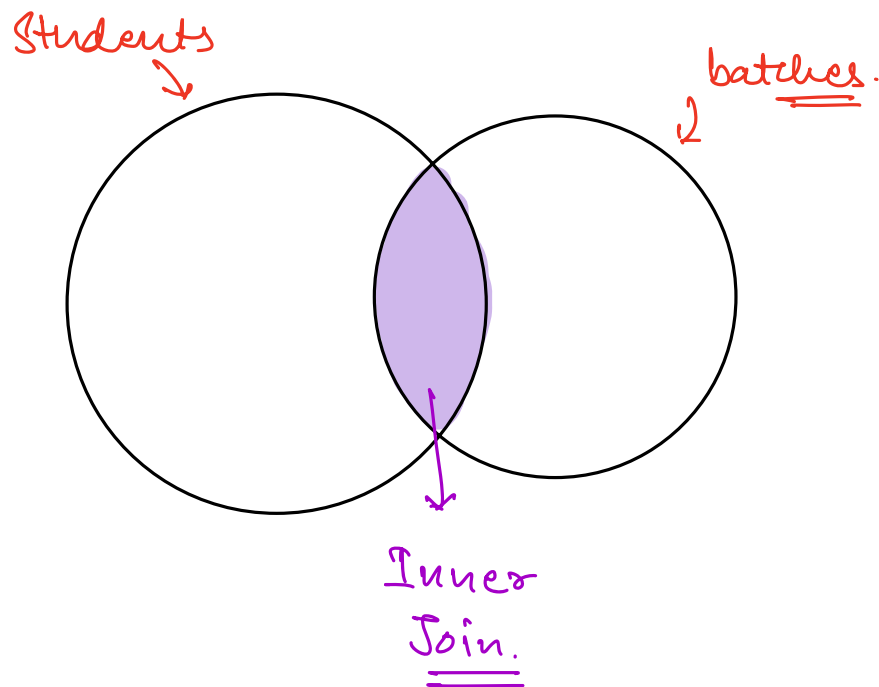
Select *

From Students s

Join batches b

ON s.batch_id = b.id;

INNER JOIN

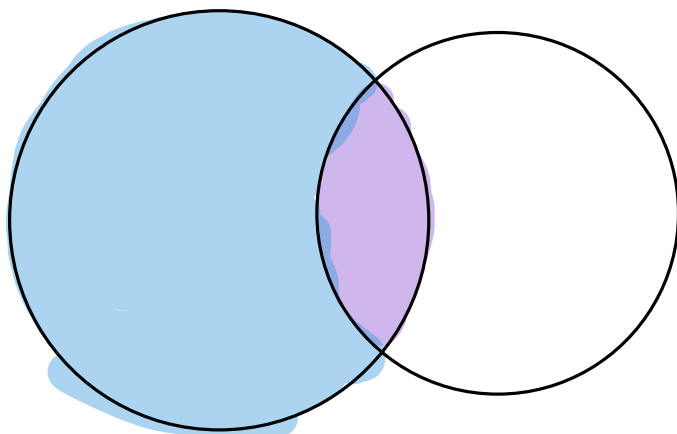


Students		
id	name	batch_id
1	A	102
2	B	103
3	C	101
4	D	101
5	E	NULL

batches	
id	name
101	B ₁
102	B ₂
103	B ₃

OUTER JOIN
 LEFT JOIN
 RIGHT JOIN
 FULL OUTER JOIN (Not in MySQL)

⇒
LEFT
JOIN



Students

id	name	batch-id
1	A	102
2	B	103
3	C	101
4	D	101
5	E	NULL

batches

id	name
101	B ₁
102	B ₂
103	B ₃

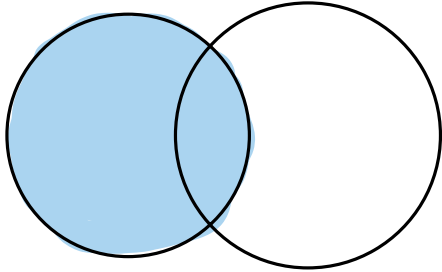
id	name	batch-id	id	batch-name
1	A	102	102	B ₂
2	B	103	103	B ₃
3	C	101	101	B ₁
4	D	101	101	B ₁
5	E	NULL	NULL	NULL

Select *

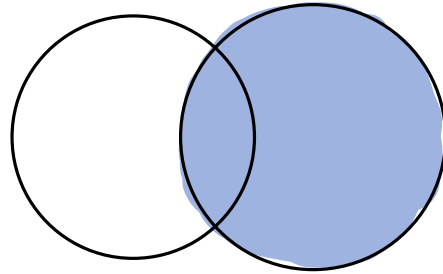
From Students s

LEFT JOIN batches b

ON s.batch_id = b.id;



LEFT JOIN |
LEFT OUTER
JOIN
==



RIGHT JOIN |
RIGHT OUTER
JOIN
==

RIGHT JOIN

Students

id	name	batch-id
1	A	102
2	B	103
3	C	101
4	D	101

batches

id	name
101	B ₁
102	B ₂
103	B ₃
104	B ₄

id	name	batch-id	id	batch-name
1	A	102	102	B ₂
2	B	103	103	B ₃
3	C	101	101	B ₁
4	D	101	101	B ₁
NULL	NULL	NULL	104	B ₄

Select *

from Students s

RIGHT JOIN batches b

ON s.batch-id = b.id;

⇒

Students

Student id

SELF
JOIN

id	name	buddy id
1	Anjum	5
2	Mukta	4
3	Hardik	1
4	Chetan	6
5	Aditya	2
6	Saurabh	3

Q: Write a query to get student name & their
buddy name.

Anjum → Aditya
Mukta → Chetan
Hardik → Anjum

Students → S1

id	name	buddy_id
1	Anjum	5
2	Mukta	4
3	Hardik	1
4	Chetan	6
5	Aditya	2
6	Saurabh	3

Students → S2

id	name	buddy_id
1	Anjum	5
2	Mukta	4
3	Hardik	1
4	Chetan	6
5	Aditya	2
6	Saurabh	3

Select S1.name as student, S2.name as buddy
from Students S1
join Students S2
ON S1.buddy_id = S2.id

employee

emp-id	name	manager_id
101	Neeraj	200
102	Saurabh	190
190	Anjum	500
200	Harsh	300
300	Mukta	700

Q: Get the employee name & their manager's name.

employee e1

emp-id	name	manager-id
101	Neeraj	200
102	Saurabh	190
190	Anjum	500
200	Harsh	300
300	Mukta	700

employee e2

emp-id	name	manager-id
101	Neeraj	200
102	Saurabh	190
190	Anjum	500
200	Harsh	300
300	Mukta	700

select e1.name as emp-name, e2.name as manager-name
from employee e1
join employee e2
ON e1.manager-id = e2.emp-id

use sakila;

select * from film;

select * from language;

-- foreign key in film table -- language (language_id → language_id)

-- Q: Get the language of all the films;

select f.title, l.name

from film f

join language l

ON f.language_id = l.language_id;

-- Q: Display the first_name, last_name, email of customers who have rented a film.

-- JOIN customer and rental table on customer_id

select c.first_name, c.last_name, c.email

from customer c

join rental r

ON c.customer_id = r.customer_id;

-- Get the count of customers who have rented a film;

select COUNT(*) -- count the number of rows.

from customer c

join rental r

ON c.customer_id = r.customer_id;

-- Get all the unique customers who have a rented movie

select DISTINCT c.first_name, c.last_name, c.email

from customer c

join rental r

ON c.customer_id = r.customer_id;

select * from customer;

select * from payment;

-- Q: List all the customers and their associated payment amounts (if any);

-- Customer | Payment

select c.customer_id, c.first_name, c.last_name, p.amount

from customer c

left join payment p

on c.customer_id = p.customer_id;