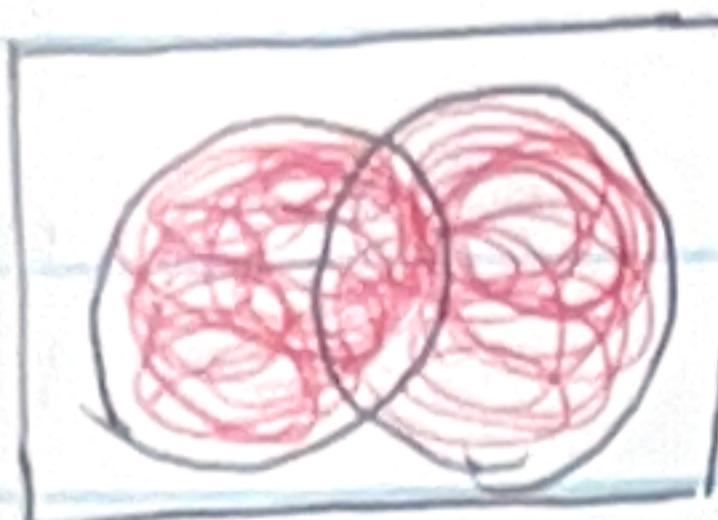


Q3 Full Outer Join -



SELECT

SELECT product_id,

product_name,

quantity

FROM products AS A

FULL OUTER JOIN Sales AS B

ON A.product_id = B.product_id;

Output

product_id	Productname	Quantity
1	Laptop	Null
2	Mouse	50
3	Keyboard	Null
4	Null	30

Exercise 4

Question 1

INNER JOIN

Q1 SELECT Student_id,

Student_name,

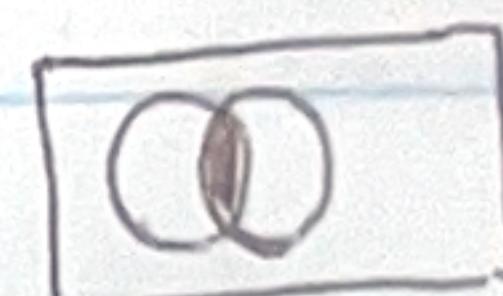
grade

FROM Students AS A

INNER JOIN grades AS B

ON A.Student_id = B.Student_id;

Output



-INNER JOIN

Studentsid	Studentname	grade.
2	Bob	B
3	Charlie	A

Left JOIN + CASE

Q4

SELECT order_id,

A.customer_id,

amount,

Customer_name,

CASE

when B.customer_id IS NOT NULL

Then 'Returning Customer'

ELSE 'New Customer'

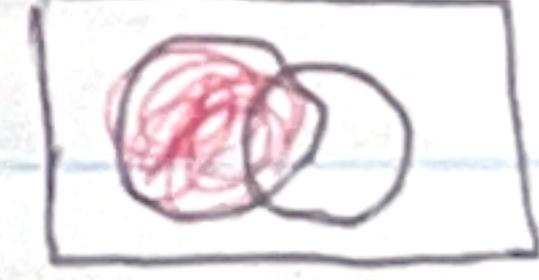
END AS Customer_type

FROM Orders AS A

LEFT JOIN Customers AS B

ON A.customer_id = B.customer_id;

Q2 LEFT JOIN



SELECT emp_id,

emp_name,

dept_name

FROM employees AS A

LEFT JOIN departments AS B

ON A.employees

ON A.emp_id = B.emp_id;

Output

Order_id	Customer_id	amount	Customer_name	Customer-type
1	101	500	Paul	Returning Customer
2	102	300	Sarah	Returning Customer
3	105	0	Null	New Customer

Output

emp_id	emp_name	dep_name
1	John	null
2	Lisa	HR
3	Mike	null

Q7

```

SELECT A.product_id, A.name,
       Count(B.task_id) AS task_count
FROM projects AS A
INNER JOIN tasks AS B
ON A.product_id = B.product_id
GROUP BY A.product_id,
         A.name;
    
```

Output	project_id	name	taskCount
	1	AI Chatbot	2
	2	Website	1

Q8

```

SELECT a.region_id,
       b.region_name,
       SUM(amount) AS total_sales
FROM region AS a
LEFT JOIN sales AS b
ON a.region_id = b.region_id
GROUP BY a.region_id,
         b.region_name;
    
```

Output	region_id	region-name	total_sales
	1	North	2000
	2	South	3500
	3	East	NULL

Q8

```

SELECT COALESCE (A.cust_id, B.cust_id) AS cust_id,
       Order_total,
       return_total
    
```

CASE
WHEN return_total IS NOT NULL THEN 'Returned'
ELSE 'No Return'
END AS return_Status
FROM orders AS A
FULL OUTER JOIN returns AS B
ON A.cust_id = B.cust_id
WHERE order_total > 100;

Output	Cust_id	order_total	return_total	return_Status
	11	120	20	Returned
	12	200	NULL	No Return
	13	180	NULL	No Return
	14	NULL	100	Returned

Q6

```

SELECT A.Student_id, name, days_present,
CASE
    WHEN days_present >= 15 THEN 'Excellent'
    WHEN days_present BETWEEN 6 AND 14
        THEN 'Needs Improvement'
    WHEN days_present <= 5 Then 'Poor Attendance'
    ELSE 'No Records'
END AS attendance_Status
FROM Students AS A
LEFT JOIN attendance AS B
ON A.Student_id = B.Student_id;
    
```

Output	Student_id	name	days_present	attendance_Status
	1	Alice	18	Excellent
	2	Bob	5	Poor Attendance
	3	Charlie	NULL	No Record

Q9 `SELECT A.user_id, name,
 COUNT(login_date) AS login-count
 FROM users AS A
 LEFT JOIN logins AS B
 ON A.user_id = B.user_id
 GROUP BY A.user_id, name
 ORDER BY login-count DESC;`

Output

user_id	name	login-count
2	Gloria	2
3	Steve	1
1	Nelson	0

Q10

`SELECT A.teacher_id, teacher-name,
 COALESCE(Subject-name, 'No Subject Assigned'
 AS Subject-name
 FROM teachers AS A
 LEFT JOIN subjects AS B
 ON A.teacher_id = B.teacher_id
 ORDER BY teacher-name ASC;`

Output

teacher_id	teacher-name	Subject-name
3	Mr. Blamiki	No Subject Assigned
1	Mr. Hlongwane	Math
1	Mr. Hlongwane	Science
2	Ms. Mdabu	No Subject Assigned