



# Can You Debug This?

(I lost 200 inactive users with this query)

```
1  -- Find users who didn't order
2  -- in last 6 months
3  SELECT
4  u.user_id,
5  u.name,
6  o.order_date
7  FROM
8  users u
9  LEFT JOIN
10 orders o
11 ON u.user_id = o.user_id
12 WHERE
13 o.order_date <
14 DATE_SUB(NOW(), INTERVAL 6 MONTH);
```



Most people miss it the first time.

I definitely did... 



- ✓ user1 (old order)
- ~~x user2 (NO orders)~~
- ~~x user3 (NO orders)~~

**200 inactive users LOST**

# THE MISTAKE

Putting the filter in WHERE after a LEFT JOIN

```
1  SELECT
2  u.user_id, u.name, o.order_date
3  FROM users u
4  LEFT JOIN orders o
5  ON u.user_id = o.user_id
6  WHERE o.order_date <
7  DATE_SUB(NOW(), INTERVAL 6 MONTH);
```



- ⚠ WHERE filters AFTER the join
- 💣 Removes rows where o.order\_date = NULL
- 😞 Users who NEVER ordered = GONE!



- ✓ user1 (old order)
- ✓ user2 (NO orders)
- ✓ user3 (NO orders)

**ALL inactive users found!**

# THE FIX

Move filter to JOIN condition

```
1  SELECT
2  u.user_id, u.name, o.order_date
3  FROM users u
4  LEFT JOIN orders o
5  ON u.user_id = o.user_id
6  AND o.order_date <
7  DATE_SUB(NOW(), INTERVAL 6 MONTH);
```

- 👉 AND filters DURING the join
- 🎯 Keeps all users from left table
- 😁 Users who never ordered = included!

