# AI Learning Log - Lab 02
**Student:** Maryam
**Lab:** SQL Fundamentals with Real Datasets

---

## AI INTERACTION #1

**Why helpful:** The explanation clarified that NULL is not a value but the absence of a value. The AI explained that nothing equals NULL, not even NULL itself, which is why we must use IS NULL.

### KEY LEARNINGS
- NULL represents missing/unknown data, not a actual value
- Using `= NULL` is silently wrong (returns 0 rows, no error)
- The correct syntax is `IS NULL` or `IS NOT NULL`
- This is a common beginner mistake in SQL

### HOW I VERIFIED
I ran both queries in psql:
```sql
-- Wrong approach - returns 0 rows
SELECT * FROM orders WHERE shipped_date = NULL;

-- Correct approach - returns 7 unshipped orders
SELECT * FROM orders WHERE shipped_date IS NULL;
```
The results confirmed the AI's explanation.

### WHAT I MODIFIED
I added comments in my queries.sql file to remind myself about this:
```sql
-- Query 8: NULL handling
-- IMPORTANT: Never use = NULL, always use IS NULL
WHERE shipped_date IS NULL
```

---

## AI INTERACTION #2

**Why helpful:** The AI explained the SQL execution order (FROM ~~SELECT LIMIT) and showed that WHERE runs before SELECT, so the~~

alias doesn't exist yet.

### KEY LEARNINGS
- SQL clauses execute in a specific order, not the order they're written
- Execution order: FROM SELECT LIMIT
- WHERE runs BEFORE SELECT, so aliases defined in SELECT don't exist yet
- Must repeat the expression in WHERE or use a subquery/CTE

### HOW I VERIFIED
I tested both versions:
```sql
-- Wrong - alias doesn't exist in WHERE
SELECT product_name, price * 0.85 AS discounted_price
FROM products WHERE discounted_price < 3000;

-- Correct - repeat the expression
SELECT product_name, price * 0.85 AS discounted_price
FROM products WHERE price * 0.85 < 3000;
```

### WHAT I MODIFIED
I now understand why my Query 9 works correctly - I compute the values in SELECT only, without trying to filter on them.

---

## AI INTERACTION #3
—
**Why helpful:** Clear explanation of CASE WHEN as SQL's if/else statement. Showed how to add multiple WHEN clauses.

### KEY LEARNINGS
- CASE WHEN is like if/else in programming
- Multiple WHEN clauses are checked in order
- First matching condition wins
- ELSE is the default if nothing matches
- Great for categorizing data (CRITICAL/URGENT/NORMAL)

### HOW I VERIFIED
I extended my Query 10 with three tiers:
```sql
CASE
    WHEN total_amount > 20000 THEN 'CRITICAL'
```

    WHEN total_amount > 10000 THEN 'URGENT'
    ELSE 'NORMAL'
END AS priority
```
The query returned correct priority labels based on order amounts.

### WHAT I MODIFIED
Updated Query 10 in queries.sql to include all three priority tiers (CRITICAL, URGENT, NORMAL) as required by the lab.

---

## Summary
**Total AI Interactions:** 3
**Main Topics Covered:** NULL handling, SQL execution order, CASE WHEN statements
**Overall Learning:** AI tools are helpful for understanding SQL concepts when you ask specific questions and verify the results yourself.
```