

# BrightLight Data Analytics — SQL JOIN Exercises

## Instructions

- For each question, study the table structures and write SQL queries using LEFT JOIN, INNER JOIN, or FULL OUTER JOIN.
  - Some questions require additional logic like CASE, WHERE, ORDER BY, or aggregate functions like SUM, COUNT.
  - Draw the final output table.
  - Submit your handwritten answers to [rofhiwa@brightlighttutorials.co.za](mailto:rofhiwa@brightlighttutorials.co.za)
- 

## Basic Join Practice (Joins Only)

---

### Question 1 — INNER JOIN

**Objective:** Join `students` and `grades` to display only students who have grades.

**Table:** `students`

student_id	student_name
1	Alice
2	Bob
3	Charlie

**Table:** `grades`

student_id	grade
2	B
3	A
4	C

**Expected Output:**

- `student_id`
  - `student_name`
  - `grade`
- 

### Question 2 — LEFT JOIN

**Objective:** Display all employees and the departments they belong to. Include employees with no department.

**Table:** `employees`

<code>emp_id</code>	<code>emp_name</code>
1	John
2	Lisa
3	Mike

**Table:** `departments`

<code>emp_id</code>	<code>dept_name</code>
2	HR
4	IT

#### Expected Output:

- `emp_id`
  - `emp_name`
  - `dept_name`
- 

### Question 3 — FULL OUTER JOIN

**Objective:** Display a complete list of products and their quantities sold. Include products with no sales and sales for unknown products.

**Table:** `products`

<code>product_id</code>	<code>product_name</code>
1	Laptop
2	Mouse
3	Keyboard

**Table:** `sales`

<code>product_id</code>	<code>quantity</code>
2	50
4	30

#### Expected Output:

- `product_id`
- `product_name`

- quantity
- 

## Advanced Join Practice (Logic + Filtering + Aggregates)

---

### Question 4 — LEFT JOIN + CASE

**Objective:** Display all orders and indicate whether the customer is “New” or “Returning”.

**Table:** `orders`

order_id	customer_id	amount
1	101	500
2	102	300
3	105	0

**Table:** `customers`

customer_id	customer_name
101	Paul
102	Sarah
104	Emma

### Expected Output:

- order\_id
  - customer\_id
  - amount
  - customer\_name
  - customer\_type (*New Customer / Returning Customer*)
- 

### Question 5 — LEFT JOIN + GROUP BY + SUM

**Objective:** Show total sales per region and include regions with no sales.

**Table:** `sales`

sale_id	region_id	amount
1	1	2000
2	2	3500

3	4	1000
---	---	------

**Table:** `regions`

region_id	region_name
1	North
2	South
3	East

### Expected Output:

- `region_id`
- `region_name`
- `total_sales`

## Question 6 — LEFT JOIN + CASE

**Objective:** Classify students based on attendance.

**Table:** `students`

student_id	name
1	Alice
2	Bob
3	Charlie

**Table:** `attendance`

student_id	days_present
1	18
2	5
4	20

### Expected Output:

- `student_id`
- `name`
- `days_present`
- `attendance_status` (*Excellent / Needs Improvement / Poor Attendance*)

## Question 7 — INNER JOIN + COUNT + GROUP BY

**Objective:** Show number of tasks per project. Only include projects that have tasks.

**Table: projects**

project_id	name
1	AI Chatbot
2	Website
3	Data Report

**Table: tasks**

task_id	project_id	status
10	1	Complete
11	1	Pending
12	2	Complete

**Expected Output:**

- project\_id
  - name
  - task\_count
- 

### Question 8 — FULL OUTER JOIN + CASE + WHERE

**Objective:** Classify customers based on whether they returned anything and filter by high order total.

**Table: orders**

order_id	cust_id	order_total
1	11	120
2	12	250
3	13	180

**Table: returns**

- |             |           |                |
|-------------|-----------|----------------|
| • return_id | • cust_id | • return_total |
| • 101       | • 11      | • 20           |
| • 102       | • 14      | • 100          |

**Expected Output:**

- cust\_id
- order\_total
- return\_total

- `return_status` (*Returned / No Return*)

Filter to include only customers with `order_total > 100`.

---

## Question 9 — LEFT JOIN + COUNT + ORDER BY

**Objective:** Count how many times each user logged in.

**Table:** `users`

<code>user_id</code>	<code>name</code>
1	Nelson
2	Gloria
3	Steve

**Table:** `logins`

<code>user_id</code>	<code>login_date</code>
2	2024-06-01
2	2024-06-02
3	2024-06-03

**Expected Output:**

- `user_id`
- `name`
- `login_count`

Order by `login_count DESC`.

---

## Question 10 — LEFT JOIN + CASE + ORDER BY

**Objective:** Show all teachers and the subjects they teach. If no subject, label appropriately.

**Table:** `teachers`

<code>teacher_id</code>	<code>teacher_name</code>
1	Mr. Hlongwane
2	Ms. Ndaba
3	Mr. Dlamini

**Table: subjects**

subject_id	teacher_id	subject_name
1	1	Math
2	1	Science
3	4	History

**Expected Output:**

- teacher\_id
- teacher\_name
- subject\_name (or "No Subject Assigned")

Order by teacher\_name ASC.