# BRIGHTLEARN TUTORIALS

## DATA ANALYTICS

## EXERCISE 4 : SQL JOINS

### Question 1 - INNER JOIN

Tables : students and grades
Objective : Join students and grades to display only student who have grades

Syntax :

```
SELECT  s.student_id,
        s. student_name,
        g. grade
FROM students AS s
INNER JOIN grades AS g ON s.student_id
 = g. student_id;
```

Output :

| student-id | student-name | grade |
|---|---|---|
| 2 | Bob | B |
| 3 | Charlie | A |

### Question 2 - LEFT JOIN

Objective : Display all employees and departments they belong to. Include employees with no departmen

Tables : employees and departments

Syntax :

```
SELECT  emp_id,
        emp_name,
```

```
        dept_name
FROM employees AS A
LEFT JOIN departments AS B
ON  A.emp_id = B.emp_id;
```

Output :

| emp_id | emp_name | dept_name |
|---|---|---|
| 1 | John | NULL |
| 2 | Lisa | HR |
| 3 | Mike | NULL |

Question 3 - FULL OUTER JOIN

Objective: Display a complete list of products and their quantities sold.

Syntax :

```
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 | product_name | quantity |
|---|---|---|
| 1 | Laptop | NULL |
| 2 | Mouse | 50 |
| 3 | keyboard | NULL |
| 4 | NULL | 30 |

Question 4 - LEFT JOIN + CASE

Objective: Display all orders and indicate whether the customer is 'New' or 'Returning'

Syntax :

```
SELECT  order_id,
        customer_id,
        amount,
        customer_name,
CASE
 WHEN customer_name 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
```

Output :

| order_id | customer_id | amount | customer_name | customer_type |
|---|---|---|---|---|
| 1 | 101 | 500 | Paul | Returning Customer |
| 2 | 102 | 300 | Sara | Returning Customer |
| 3 | 105 | 0 | NULL | New customer |

Question 5 – LEFT JOIN + GROUP BY + SUM

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

Syntax :

```
SELECT region_id,
       region_name
```

```
    SUM(amount) AS total_sales
FROM regions AS A
LEFT JOIN Sales AS B ON A.region_id =
B.region_id
GROUP BY A.region_name;
```

Output :

| region_id | region_name | total_sales |
|---|---|---|
| 1 | North | 2000 |
| 2 | South | 3500 |
| 3 | East | 0 |

Question 6 - LEFT JOIN + CASE

Objective : classify student based on attendance

Syntax :

```
SELECT  student_id,
        name,
        days-present,
CASE
WHEN days_present >= 15 THEN 'Excellent'
WHEN days_present >=10 THEN 'Need Improvement'
ELSE 'Poor attendance'
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 Attendane |
| 3 | Charlie | NULL | Poor atterdance |

Question 7 - INNER JOIN + COUNT + GROUP BY

Objective : show number of tasks per project

Syntax :

```
SELECT project_id,
       name,
   COUNT (task_id) AS task_count
FROM projects AS A
INNER JOIN tasks AS B ON A.project_id
 = B.project_id
GROUP BY A.project_name, Aproject_id ;
```

Output :

| project_id | name | task_count |
|---|---|---|
| 1 | AI Chatbot | 2 |
| 2 | website | 1 |

Question 8 - FULL OUTER JOIN + CASE + WHERE

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

Syntax :

```
SELECT 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 | 250 | NULL | No Return |
| 13 | 180 | NULL | No Return |

Question 9 – LEFT JOIN + COUNT + ORDER BY

Objective : Count how many times each user logged in

Syntax :

```
SELECT 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 user_id, name
ORDER BY login_count DESC;
```

Output :

| user_id | name | login_count |
|---|---|---|
| 2 | Gloria | 2 |
| 3 | Steve | 1 |
| 1 | Nelson | NULL |

Question 10 - LEFT JOIN + CASE + ORDER BY

Objective : show all teachers and the subjects they teach

Syntax :

```
SELECT  teacher_id,
        teacher_name
CASE
 WHEN subject_name IS NOT NULL THEN
  subject_name
 ELSE 'No subject Assigned'
END 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 Dlamini | No subject Assigned |
| 1 | Mr Hlongwane | Math |
| 1 | Mr Hlongwane | Science |
| 2 | Ms Ndaba | No subject Assigned |