NAME: Abenes, Enrico O. **DATE:** November 26, 2024

SUBJECT & SECTION: CC5 – 2I

LABORATORY ACTIVITY 12: JOINS

Instruction:

- Screenshot your SQL Worksheet code (include your student ID and name as comment at the top of your code)
- Screenshot the Schema page if you are successful.
- You can use **Programiz or any other tool** to run your code.
- Submit your work in either a Word document or a PDF file.

Products Table:

product_id	product_name	category	price
101	Laptop	Electronics	57000
102	Smartphone	Electronics	22000
103	Table	Furniture	200
104	Chair	Furniture	250
105	Coffee Maker	Appliances	3500

Sales Table:

sale_id	product_id	sale_date	quantity
300	102	2024-11-01	2
301	101	2024-11-02	1
302	104	2024-11-03	6
303	106	2024-11-04	1
304	103	2024-11-05	2

Task:

1. **INNER JOIN:** Write a query to retrieve the list of products and their corresponding sales transactions. Only include products that have been sold (i.e., no product without a sale should appear).

```sql
-- Student ID: 19-5357-989
-- Name: Abenes, Enrico O.

SELECT p.product_id, p.product_name, p.category, p.price, s.sale_id, s.sale_date, s.quantity FROM Products p INNER JOIN Sales s ON p.product_id = s.product_id;
```

product_id	product_name	category	price	sale_id	sale_date	quantity
102	Smartphone	Electronics	22000	300	2024-11-01	2
101	Laptop	Electronics	57000	301	2024-11-02	1
104	Chair	Furniture	250	302	2024-11-03	6
103	Table	Furniture	200	304	2024-11-05	2

2. **LEFT JOIN:** Write a query to retrieve the list of all products along with any sales transactions. The result should include all products, even those that have not been sold. For products with no sales, show NULL for the sales details.

```sql
-- Student ID: 19-5357-989
-- Name: Abenes, Enrico O.

SELECT p.product_id, p.product_name, p.category, p.price, s.sale_id, s.sale_date, s.quantity FROM Products p LEFT JOIN Sales s ON p.product_id = s.product_id;
```

product_id	product_name	category	price	sale_id	sale_date	quantity
101	Laptop	Electronics	57000	301	2024-11-02	1
102	Smartphone	Electronics	22000	300	2024-11-01	2
103	Table	Furniture	200	304	2024-11-05	2
104	Chair	Furniture	250	302	2024-11-03	6
105	Coffee Maker	Appliances	3500	NULL	NULL	NULL

3. **RIGHT JOIN:** Write a query to retrieve the list of all sales transactions along with their corresponding product details. The result should include all sales, even those that do not correspond to a valid product (e.g., sales for a non-existent product).

```
1 -- Student ID: 19-5357-989
2 -- Name: Abenes, Enrico O.
3
4 SELECT p.product_id, p.product_name, p.category, p.price, s.sale_id, s.sale_date, s.quantity FROM Sales s RIGHT JOIN Products p ON p.product_id = s.product_id;
```

product_id	product_name	category	price	sale_id	sale_date	quantity
102	Smartphone	Electronics	22000	300	2024-11-01	2
101	Laptop	Electronics	57000	301	2024-11-02	1
104	Chair	Furniture	250	302	2024-11-03	6
103	Table	Furniture	200	304	2024-11-05	2
105	Coffee Maker	Appliances	3500	NULL	NULL	NULL

4. **FULL JOIN:** Write a query to retrieve a list of all products and all sales transactions. This query should include products with no sales and sales that do not correspond to any product.

```
1 -- Student ID: 19-5357-989
2 -- Name: Abenes, Enrico O.
3
4 SELECT p.product_id, p.product_name, p.category, p.price, s.sale_id, s.sale_date, s.quantity FROM Products p FULL JOIN Sales s ON p.product_id = s.product_id;
```

product_id	product_name	category	price	sale_id	sale_date	quantity
101	Laptop	Electronics	57000	301	2024-11-02	1
102	Smartphone	Electronics	22000	300	2024-11-01	2
103	Table	Furniture	200	304	2024-11-05	2
104	Chair	Furniture	250	302	2024-11-03	6
105	Coffee Maker	Appliances	3500	NULL	NULL	NULL
NULL	NULL	NULL	NULL	303	2024-11-04	1