# Experiment 5

**Student Name: Umang Kaushik** **UID: 23BCS10712**
**Branch: BE CSE** **Section/Group: KRG-1B**
**Semester: 4th** **Date of Performance: 16th Sep**
**Subject Name: DBMS** **Subject Code: 23CSP-333**

## 1. Questions and Code:

### Question 1 -

1. Create a large dataset: - Create a table names transaction_data (id , value) with 1 million records. - take id 1 and 2, and for each id, generate 1 million records in value column - Use Generate_series () and random() to populate the data.
2. Create a normal view and materialized view to for sales_summary, which includes total_quantity_sold, total_sales, and total_orders with aggregation.
3. Compare the performance and execution time of both.

### Code -

```
CREATE TABLE transaction_data (
   id INT,
   value NUMERIC
);

INSERT INTO transaction_data (id, value)
SELECT 1 AS id, random() * 1000 AS value
FROM generate_series(1, 1000000)
UNION ALL
```

```
SELECT 2 AS id, random() * 1000 AS value
FROM generate_series(1, 1000000);

CREATE VIEW sales_summary_view AS
SELECT
   id,
   COUNT(*) AS total_orders,
   SUM(value) AS total_sales
FROM
   transaction_data
GROUP BY
   id;

CREATE MATERIALIZED VIEW sales_summary_materialized_view
AS
SELECT
   id,
   COUNT(*) AS total_orders,
   SUM(value) AS total_sales
FROM
   transaction_data
GROUP BY
   id;

EXPLAIN ANALYZE SELECT * FROM sales_summary_view;

EXPLAIN ANALYZE SELECT * FROM
sales_summary_materialized_view;

REFRESH MATERIALIZED VIEW sales_summary_materialized_view;
```

**Question 2 –**

The company TechMart Solutions stores all sales transactions in a central database.
A new reporting team has been formed to analyze sales but they should not have direct access to the base tables for security reasons. The database administrator has decided to: Create restricted views to display only summarized, non-sensitive data. 2. Assign access to these views to specific users using DCL commands (GRANT, REVOKE).

**Code –**

```
CREATE TABLE sales_transactions (
    transaction_id SERIAL PRIMARY KEY,
    product_name VARCHAR(100),
    customer_id INT, -- Sensitive column
    sale_amount NUMERIC,
    transaction_date DATE
);

INSERT INTO sales_transactions (product_name, customer_id, sale_amount, transaction_date) VALUES
('Laptop', 101, 1200.00, '2025-01-15'),
('Mouse', 102, 25.00, '2025-01-16'),
('Laptop', 103, 1250.00, '2025-02-10'),
('Keyboard', 101, 75.00, '2025-02-12'),
```

```
('Mouse', 103, 30.00, '2025-02-20');

CREATE ROLE reporting_team;

CREATE VIEW monthly_product_sales AS
SELECT
    to_char(transaction_date, 'YYYY-MM') AS sales_month,
    product_name,
    COUNT(*) AS units_sold,
    SUM(sale_amount) AS total_revenue
FROM
    sales_transactions
GROUP BY
    sales_month,
    product_name;

GRANT SELECT ON monthly_product_sales TO reporting_team;
REVOKE SELECT ON monthly_product_sales FROM reporting_team;
```