# Project: OLAP Operations (using Redshift or PostgreSQL)

**1) Database creation**

```
CREATE DATABASE "Sales Data "
    WITH
    OWNER = postgres
    ENCODING = 'UTF8'
    LC_COLLATE = 'English_United States.1252'
    LC_CTYPE = 'English_United States.1252'
    TABLESPACE = pg_default
    CONNECTION LIMIT = -1
    IS_TEMPLATE = False;\
```

**Table Creation:**

```
CREATE TABLE Sales_sample (Product_Id INT, Region VARCHAR(50), On_date DATE,
 Sales_Amount NUMERIC);
```

**2) Data Creation**

```
INSERT INTO Sales_sample (Product_Id, Region, On_date, Sales_Amount) VALUES
('1', 'East', '2023-10-10', '45000'),
('2', 'West', '2023-09-19', '75000'),
('2', 'East', '2023-10-21', '65000'),
('3', 'North', '2023-09-20', '40000'),
('4', 'North', '2023-08-06', '70000'),
('2', 'South', '2023-08-25', '76000'),
('5', 'North', '2023-11-23', '48000'),
('5', 'West', '2023-11-11', '58000'),
('3', 'East', '2023-09-19', '72000'),
('1', 'West', '2023-09-29', '63000');
```

```
Select Select * from Sales_Sample;
```

| | product_id integer | region character varying (50) | on_date date | sales_amount numeric |
|---|---|---|---|---|
| 1 | 1 | East | 2023-10-10 | 45000 |
| 2 | 2 | West | 2023-09-19 | 75000 |
| 3 | 2 | East | 2023-10-21 | 65000 |
| 4 | 3 | North | 2023-09-20 | 40000 |
| 5 | 4 | North | 2023-08-06 | 70000 |
| 6 | 2 | South | 2023-08-25 | 76000 |
| 7 | 5 | North | 2023-11-23 | 48000 |
| 8 | 5 | West | 2023-11-11 | 58000 |
| 9 | 3 | East | 2023-09-19 | 72000 |
| 10 | 1 | West | 2023-09-29 | 63000 |

**3) OLAP operations**

a) Drill down - Write a query to perform drill down from region to product level to understand sales performance

```
SELECT Region, Product_Id, Sum(Sales_Amount) AS Sales_Amount
FROM Sales_Sample
GROUP BY 1,2
    ORDER BY Region, Product_Id, Sales_Amount;
```

| | region character varying (50) | product_id integer | sales_amount numeric |
|---|---|---|---|
| 1 | East | 1 | 45000 |
| 2 | East | 2 | 65000 |
| 3 | East | 3 | 72000 |
| 4 | North | 3 | 40000 |
| 5 | North | 4 | 70000 |
| 6 | North | 5 | 48000 |
| 7 | South | 2 | 76000 |
| 8 | West | 1 | 63000 |
| 9 | West | 2 | 75000 |
| 10 | West | 5 | 58000 |

b) Roll Up - Write a query to perform roll up from product to region level to view total sales by region.

```
SELECT Region, Product_Id, Sum(Sales_Amount) AS Sales_Amount
FROM Sales_Sample
GROUP BY ROLLUP (1,2)
ORDER BY Region;
```

| | region character varying (50) | product_id integer | sales_amount numeric |
|---|---|---|---|
| 1 | East | 1 | 45000 |
| 2 | East | 2 | 65000 |
| 3 | East | 3 | 72000 |
| 4 | East | [null] | 182000 |
| 5 | North | 3 | 40000 |
| 6 | North | 4 | 70000 |
| 7 | North | 5 | 48000 |
| 8 | North | [null] | 158000 |
| 9 | South | 2 | 76000 |
| 10 | South | [null] | 76000 |
| 11 | West | 1 | 63000 |
| 12 | West | 2 | 75000 |
| 13 | West | 5 | 58000 |
| 14 | West | [null] | 196000 |
| 15 | [null] | [null] | 612000 |

c) Cube - Write a query to explore sales data from different perspectives, such as product, region, and date

```
SELECT Region, Product_Id, On_Date, SUM(Sales_Amount) AS Sales_Amount
FROM Sales_Sample
GROUP BY Cube (1,2,3)
ORDER BY Region, Product_Id, On_Date, Sales_Amount;
```

| | region character varying (50) | product_id integer | on_date date | sales_amount numeric |
|---|---|---|---|---|
| 1 | East | 1 | 2023-10-10 | 45000 |
| 2 | East | 1 | [null] | 45000 |
| 3 | East | 2 | 2023-10-21 | 65000 |
| 4 | East | 2 | [null] | 65000 |
| 5 | East | 3 | 2023-09-19 | 72000 |
| 6 | East | 3 | [null] | 72000 |
| 7 | East | [null] | 2023-09-19 | 72000 |
| 8 | East | [null] | 2023-10-10 | 45000 |
| 9 | East | [null] | 2023-10-21 | 65000 |
| 10 | East | [null] | [null] | 182000 |
| 11 | North | 3 | 2023-09-20 | 40000 |
| 12 | North | 3 | [null] | 40000 |
| 13 | North | 4 | 2023-08-06 | 70000 |
| 14 | North | 4 | [null] | 70000 |
| 15 | North | 5 | 2023-11-23 | 48000 |
| 16 | North | 5 | [null] | 48000 |
| 17 | North | [null] | 2023-08-06 | 70000 |
| 18 | North | [null] | 2023-09-20 | 40000 |
| 19 | North | [null] | 2023-11-23 | 48000 |
| 20 | North | [null] | [null] | 158000 |
| 21 | South | 2 | 2023-08-25 | 76000 |
| 22 | South | 2 | [null] | 76000 |
| 23 | South | [null] | 2023-08-25 | 76000 |
| 24 | South | [null] | [null] | 76000 |
| 25 | West | 1 | 2023-09-29 | 63000 |
| 26 | West | 1 | [null] | 63000 |
| 27 | West | 2 | 2023-09-19 | 75000 |

Total rows: 59 Query complete 00:00:00.089

d) Slice - Write a query to slice the data to view sales for a particular region or date range

```
SELECT Region, Product_Id, On_Date, SUM(Sales_Amount) AS Sales_Amount
FROM Sales_Sample
WHERE Region in('North', 'South') OR On_Date BETWEEN To_date('2023-08-20','YYYY-MM-DD') AND
To_Date('2023-10-20','YYYY-MM-DD')
GROUP BY 1,2,3
ORDER BY Region, Product_Id, On_Date, Sales_Amount;
```

| | region character varying (50) | product_id integer | on_date date | sales_amount numeric |
|---|---|---|---|---|
| 1 | East | 1 | 2023-10-10 | 45000 |
| 2 | East | 3 | 2023-09-19 | 72000 |
| 3 | North | 3 | 2023-09-20 | 40000 |
| 4 | North | 4 | 2023-08-06 | 70000 |
| 5 | North | 5 | 2023-11-23 | 48000 |
| 6 | South | 2 | 2023-08-25 | 76000 |
| 7 | West | 1 | 2023-09-29 | 63000 |
| 8 | West | 2 | 2023-09-19 | 75000 |

e) Dice - Write a query to view sales for specific combinations of product, region, and date

```
SELECT Region, Product_Id, On_Date, SUM(Sales_Amount) AS Sales_Amount
FROM Sales_Sample
WHERE Region in ('North', 'South') AND Product_Id IN (1,2) AND On_Date
BETWEEN To_date('2023-08-20','YYYY-MM-DD') And To_Date('2023-10-20','YYYY-MM-DD')
GROUP BY 1,2,3
ORDER BY Region, Product_Id, On_Date, Sales_Amount;
```

| | region character varying (50) | product_id integer | on_date date | sales_amount numeric |
|---|---|---|---|---|
| 1 | South | 2 | 2023-08-25 | 76000 |