Task No:3 Date:06/08/25	Perform Data Cube Operations (OLAP Operations) using SQL Queries Rollup, Rolldown, Slicing, Dicing **Database: MySQL**	

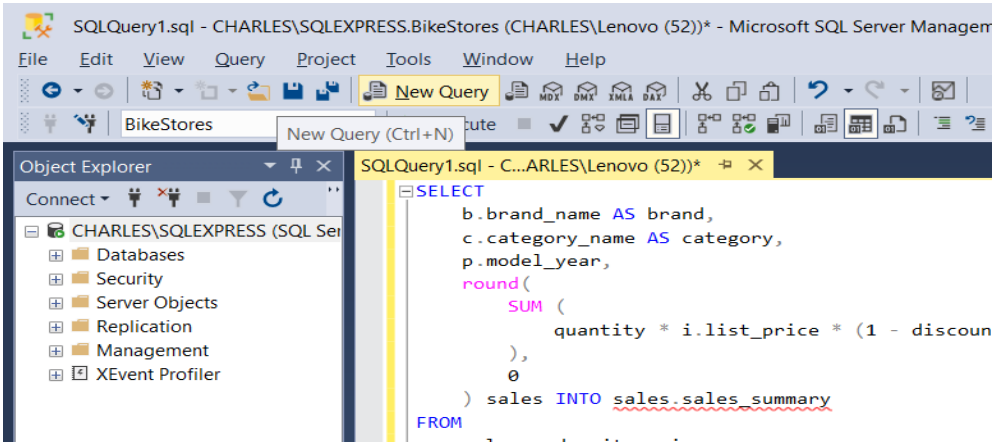
AIM

To Perform Data Cube Operations (OLAP Operations) using SQL Queries Rollup, Roll down, Slicing, Dicing

PROCEDURE

1) create the sales.sales_summary table by using the following query:

```
SELECT
   b.brand_name AS brand,
   c.category_name AS category,
   p.model_year,
   round(
      SUM (
         quantity * i.list_price * (1 - discount)
      ),
      0
   ) sales INTO sales.sales_summary
FROM
   sales.order_items i
INNER JOIN production.products p ON p.product_id = i.product_id
INNER JOIN production.brands b ON b.brand_id = p.brand_id
INNER JOIN production.categories c ON c.category_id = p.category_id
GROUP BY
   b.brand_name,
   c.category_name,
   p.model_year
ORDER BY
   b.brand_name,
   c.category_name,
   p.model_year;
```



2) CUBE:

Refer task 2a:

The CUBE is a subclause of the GROUP BY clause that allows you to generate multiple grouping sets. The following illustrates the general syntax of the CUBE:

```
SELECT
   d1,
   d2,
   d3,
   aggregate_function (c4)
FROM
   table_name
GROUP BY
   d1,
   CUBE (d2, d3);
```

Example:

```
SELECT
   brand,
   category,
   SUM (sales) sales
FROM
   sales.sales_summary
GROUP BY
   CUBE(brand, category);
```

3) ROLLUP

Refer task 2a:

SQL Server ROLLUP syntax

The general syntax of the SQL Server ROLLUP is as follows:

```
SELECT

   d1,

   d2,

   d3,

   aggregate_function(c4)

FROM

   table_name

GROUP BY

   ROLLUP (d1, d2, d3);
```

Upload and configure the sample data warehouse (refer task2a)

Select the New Query, then new query tab will be opened.

ROLLUP Query

```
SELECT
   brand,
   category,
   SUM (sales) sales
FROM
   sales.sales_summary
GROUP BY
   ROLLUP(brand, category);
```

4) DRILLDOWN

This is a reverse of the ROLL UP operation. The data is aggregated from a higher level summary to a lower level summary/detailed data.

```
SELECT
   brand,
   category,
   SUM (sales) sales
FROM
   sales.sales_summary
GROUP BY
   ROLLUP(brand, category)
HAVING SUM (sales) > 20000;
```

5) SLICING:

A slice in a multidimensional array is a column of data corresponding to a single value for one or more members of the dimension. It helps the user to visualize and gather the information specific to a dimension.

```
SELECT
   *
FROM
   sales.customers
WHERE
   state = 'CA';
```

6) DICING:

Dicing is similar to slicing, but it works a little bit differently. Dicing, on the other hand, is more of a zoom feature that selects a subset over all the dimensions, but for specific values of the dimension.

```
SELECT
   city,
   COUNT (*)
FROM
```

```
    sales.customers
WHERE
    state = 'CA'
GROUP BY
    city
HAVING
    COUNT (*) > 10
ORDER BY
    city;
```

7) PIVOT

The steps to make a query a pivot table:

1) First, select a base dataset for pivoting.
2) Second, create a temporary result by using a derived table or common table expression (CTE)
3) Third, apply the PIVOT operator.

- First, select category name and product id from the production.products and production.categories tables as the base data for pivoting:

```
SELECT
    category_name,
    product_id
FROM
    production.products p
    INNER JOIN production.categories c
        ON c.category_id = p.category_id
```

- Second, create a temporary result set using a derived table:

```
SELECT * FROM (
    SELECT
        category_name,
        product_id
    FROM
        production.products p
        INNER JOIN production.categories c
            ON c.category_id = p.category_id
) t
```

- Third, apply the PIVOT operator:

```
    SELECT * FROM
    (
        SELECT
            category_name,
            product_id
        FROM
            production.products p
            INNER JOIN production.categories c
                ON c.category_id = p.category_id
    ) t
```

```
PIVOT(
   COUNT(product_id)
   FOR category_name IN (
      [Children Bicycles],
      [Comfort Bicycles],
      [Cruisers Bicycles],
      [Cyclocross Bicycles],
      [Electric Bikes],
      [Mountain Bikes],
      [Road Bikes])
) AS pivot_table;
```

RESULT:

Thus the Data Cube Operations (OLAP Operations) using SQL Queries Rollup, Roll down, Slicing, Dicing, Pivot was executed using SSMS and SQl Server.