**DWBI**

Q1. Find the total sales by country\_id and channel\_desc for the US and GB through the Internet and direct sales in September 2000 and October 2000 using ROLL-UP Extension. and country\_iso\_code for each channel\_desc value.

QUERY:

SELECT channels.channel\_desc,calendar\_month\_desc,countries.country\_iso\_code,

TO\_CHAR(SUM(amount\_sold), '9,999,999,999') AS SALES$

FROM sales

JOIN customers ON sales.cust\_id = customers.cust\_id

JOIN times ON sales.time\_id = times.time\_id

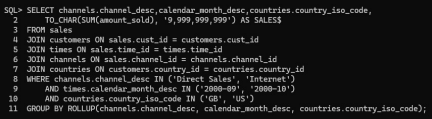
JOIN channels ON sales.channel\_id = channels.channel\_id

JOIN countries ON customers.country\_id = countries.country\_id

WHERE channels.channel\_desc IN ('Direct Sales', 'Internet')

AND times.calendar\_month\_desc IN ('2000-09', '2000-10')

AND countries.country\_iso\_code IN ('GB', 'US')

GROUP BY ROLLUP(channels.channel\_desc, calendar\_month\_desc, countries.country\_iso\_code); 

O/P:



Q2. Find the total sales by country\_iso\_code and channel\_desc for the US and GB through the Internet and direct sales in September 2000 and October 2000 using CUBE aggregation across three dimensions- channel\_desc, calendar\_month\_desc, countries. country\_iso\_code.

QUERY:

SELECT channels.channel\_desc, times.calendar\_month\_desc, countries.country\_iso\_code,

TO\_CHAR(SUM(sales.amount\_sold), '9999999999') AS SALES$

FROM sales JOIN customers ON sales.cust\_id = customers.cust\_id

JOIN times ON sales.time\_id = times.time\_id

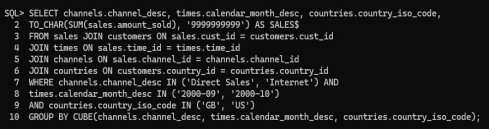
JOIN channels ON sales.channel\_id = channels.channel\_id

JOIN countries ON customers.country\_id = countries.country\_id

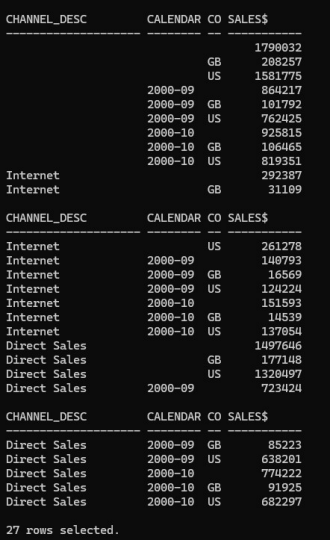
WHERE channels.channel\_desc IN ('Direct Sales', 'Internet') AND

times.calendar\_month\_desc IN ('2000-09', '2000-10')

AND countries.country\_iso\_code IN ('GB', 'US')

GROUP BY CUBE(channels.channel\_desc, times.calendar\_month\_desc, countries.country\_iso\_code); 

O/P:



Q3. Find the total sales by country\_iso and channel\_desc for the US and France through the Internet and direct sales in September 2000

QUERY:

SELECT channels.channel\_desc, countries.country\_iso\_code,

TO\_CHAR(SUM(sales.amount\_sold), '9999999999') AS SALES$ FROM sales

JOIN customers ON sales.cust\_id = customers.cust\_id

JOIN channels ON sales.channel\_id = channels.channel\_id

JOIN countries ON customers.country\_id = countries.country\_id

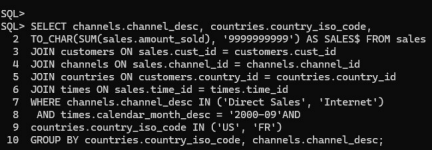
JOIN times ON sales.time\_id = times.time\_id

WHERE channels.channel\_desc IN ('Direct Sales', 'Internet')

AND times.calendar\_month\_desc = '2000-09'AND

countries.country\_iso\_code IN ('US', 'FR')

GROUP BY countries.country\_iso\_code, channels.channel\_desc;



O/P:



Q4. Find the total sales by country\_iso and channel\_desc for the US and GB through the Internet and direct sales in September 2000 and October 2009 using PARTIAL ROLL-UP.

QUERY:

SELECT channels.channel\_desc, times.calendar\_month\_desc, countries.country\_iso\_code, TO\_CHAR(SUM(sales.amount\_sold), '9999999999') AS SALES$

FROM sales JOIN customers ON sales.cust\_id = customers.cust\_id

JOIN channels ON sales.channel\_id = channels.channel\_id

JOIN countries ON customers.country\_id = countries.country\_id

JOIN times ON sales.time\_id = times.time\_id

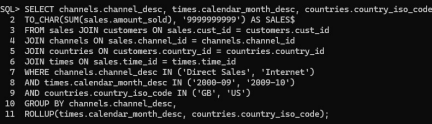
WHERE channels.channel\_desc IN ('Direct Sales', 'Internet')

AND times.calendar\_month\_desc IN ('2000-09', '2009-10')

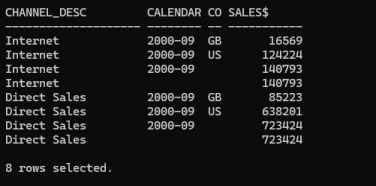
AND countries.country\_iso\_code IN ('GB', 'US')

GROUP BY channels.channel\_desc,

ROLLUP(times.calendar\_month\_desc, countries.country\_iso\_code);



O/P:



Q5. Find the total sales by country\_id and channel\_desc for the US and GB through the Internet and direct sales in September 2000 and October 2009 using PARTIAL CUBE aggregation on month and country code and GROUP BY on channel\_desc.

QUERY:

SELECT channels.channel\_desc, times.calendar\_month\_desc, countries.country\_iso\_code,

TO\_CHAR(SUM(sales.amount\_sold), '9999999999') AS SALES$

FROM sales JOIN customers ON sales.cust\_id = customers.cust\_id

JOIN channels ON sales.channel\_id = channels.channel\_id

JOIN countries ON customers.country\_id = countries.country\_id

JOIN times ON sales.time\_id = times.time\_id

WHERE channels.channel\_desc IN ('Direct Sales', 'Internet')

AND times.calendar\_month\_desc IN ('2000-09', '2009-10')

AND countries.country\_iso\_code IN ('GB', 'US')

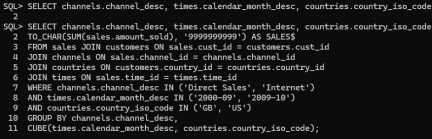
GROUP BY GROUPING SETS(

(channels.channel\_desc, times.calendar\_month\_desc, countries.country\_iso\_code),

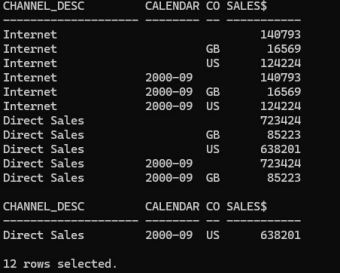
(channels.channel\_desc, countries.country\_iso\_code),

(times.calendar\_month\_desc, countries.country\_iso\_code)

);



O/P:



Q6. Use GROUPING to create a set of mask columns for the result set of Q1. Create grouping on channel\_desc and name it as CH

Create grouping calendar\_month\_desc and name it as MO

Create grouping on country\_iso\_code and name it as CO

QUERY:

SELECT channel\_desc, calendar\_month\_desc, country\_iso\_code,

TO\_CHAR(SUM(amount\_sold), '9,999,999,999') SALES$, GROUPING(channel\_desc) AS

Ch,

GROUPING(calendar\_month\_desc) AS Mo, GROUPING(country\_iso\_code) AS Co

FROM sales, customers, times, channels, countries

WHERE sales.time\_id=times.time\_id

AND sales.cust\_id=customers.cust\_id

AND customers.country\_id = countries.country\_id

AND sales.channel\_id= channels.channel\_id

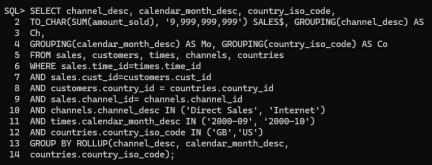
AND channels.channel\_desc IN ('Direct Sales', 'Internet')

AND times.calendar\_month\_desc IN ('2000-09', '2000-10')

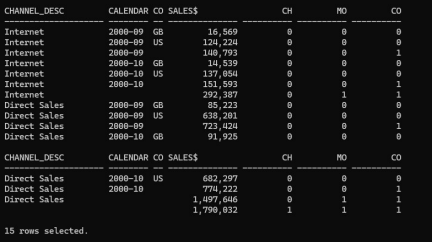
AND countries.country\_iso\_code IN ('GB','US')

GROUP BY ROLLUP(channel\_desc, calendar\_month\_desc,

countries.country\_iso\_code);



O/P:



Q7. Find the total sales by country\_id and channel\_desc for the US and GB through the Internet and direct sales in September 2000 and October 2009 using GROUPING SETS. Calculate aggregates over three groupings:

(channel\_desc, calendar\_month\_desc, country\_iso\_code)

(channel\_desc, country\_iso\_code)

(calendar\_month\_desc, country\_iso\_code)

QUERY:

SELECT channels.channel\_desc, times.calendar\_month\_desc, countries.country\_iso\_code, TO\_CHAR(SUM(sales.amount\_sold), '9999999999') AS SALES$

FROM sales JOIN customers ON sales.cust\_id = customers.cust\_id

JOIN channels ON sales.channel\_id = channels.channel\_id

JOIN countries ON customers.country\_id = countries.country\_id

JOIN times ON sales.time\_id = times.time\_id

WHERE channels.channel\_desc IN ('Direct Sales', 'Internet')

AND times.calendar\_month\_desc IN ('2000-09', '2009-10')

AND countries.country\_iso\_code IN ('GB', 'US')

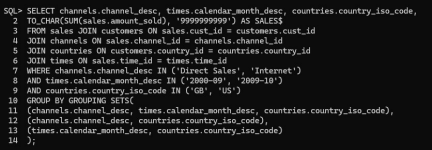
GROUP BY GROUPING SETS(

(channels.channel\_desc, times.calendar\_month\_desc, countries.country\_iso\_code),

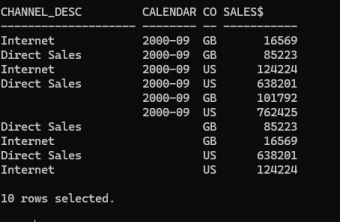
(channels.channel\_desc, countries.country\_iso\_code),

(times.calendar\_month\_desc, countries.country\_iso\_code)

);



O/P:



Q8. Consider the following Query and make conclusion from the result obtained. (scott Schema)

QUERY:

SELECT deptno,job,SUM(sal)

FROM emp

GROUP BY CUBE(deptno,job);



O/P:



Q9. Calculate the salary for each department present in different cities of hr schema using rollup. QUERY:

SELECT departments.department\_id, locations.city, SUM(employees.salary) AS total\_salary FROM employees

JOIN departments ON employees.department\_id = departments.department\_id

JOIN locations ON departments.location\_id = locations.location\_id

GROUP BY ROLLUP (departments.department\_id, locations.city);

O/P:



Q10. Calculate the salary for each department present in different cities of hr schema using cube.

QUERY:

SELECT departments.department\_id, locations.city, SUM(employees.salary) AS total\_salary

FROM employees

JOIN departments ON employees.department\_id = departments.department\_id

JOIN locations ON departments.location\_id = locations.location\_id

GROUP BY CUBE (departments.department\_id, locations.city);

O/P:



CONCLUSION:

In this practical, we explored the use of SQL aggregation techniques like ROLLUP, CUBE, and GROUPING SETS. These tools help in summarizing data efficiently across multiple dimensions, which is crucial for data warehousing. By applying these methods, we can generate detailed and high-level summaries with minimal effort, enhancing the performance and scalability of data analysis tasks.