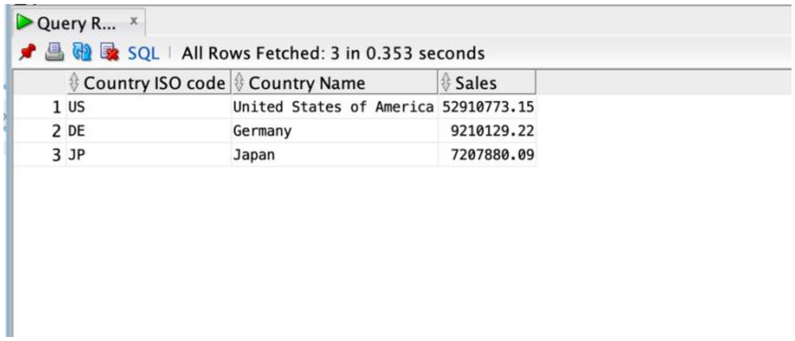
Question 1 Display the top three countries with the highest ‘`total sales`’.

```
SELECT
   c.COUNTRY_ISO_CODE AS "Country ISO code",
   c.COUNTRY_NAME AS "Country Name",
   SUM(s.AMOUNT_SOLD) AS "Sales"
FROM
   SH.COUNTRIES c
JOIN
   SH.CUSTOMERS cu ON c.COUNTRY_ID = cu.COUNTRY_ID
JOIN
   SH.SALES s ON cu.CUST_ID = s.CUST_ID
GROUP BY
   c.COUNTRY_ISO_CODE, c.COUNTRY_NAME
ORDER BY
   SUM(s.AMOUNT_SOLD) DESC
FETCH FIRST 3 ROWS ONLY;
```

OUTPUT:



All Rows Fetched: 3 in 0.353 seconds

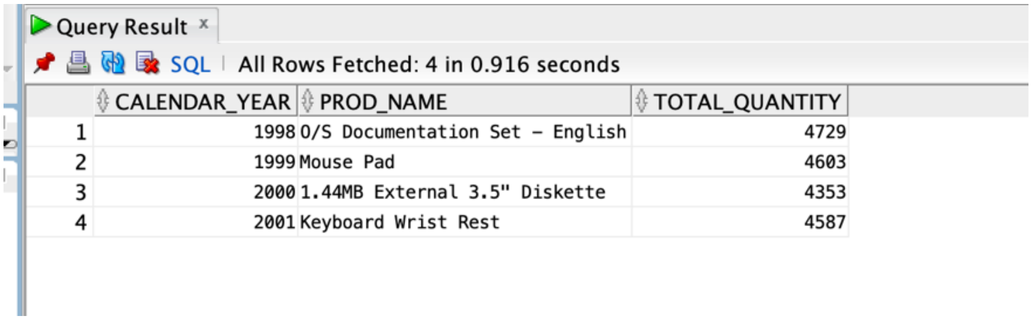
```
    Country ISO code  Country Name              Sales
  1 US                United States of America  52910773.15
  2 DE                Germany                    9210129.22
  3 JP                Japan                      7207880.09
```

Question 2 Find and display the MOST sold products/items (absolute frequency, not %) in the US, in each year 1998 - 2001.

```
WITH US_SALES AS (
   SELECT
      EXTRACT(YEAR FROM T.TIME_ID) AS Calendar_Year,
      P.PROD_NAME AS Prod_name,
      SUM(S.QUANTITY_SOLD) AS Total_Quantity
   FROM
      SH.COUNTRIES C
   JOIN
      SH.CUSTOMERS CS ON C.COUNTRY_ID = CS.COUNTRY_ID
   JOIN
```

```
    SH.SALES S ON CS.CUST_ID = S.CUST_ID
  JOIN
    SH.PRODUCTS P ON S.PROD_ID = P.PROD_ID
  JOIN
    SH.TIMES T ON S.TIME_ID = T.TIME_ID
  WHERE
    C.COUNTRY_ISO_CODE = 'US'
    AND EXTRACT(YEAR FROM T.TIME_ID) BETWEEN 1998 AND 2001
  GROUP BY
    EXTRACT(YEAR FROM T.TIME_ID), P.PROD_NAME
),
RANKED_PRODUCTS AS (
  SELECT
    Calendar_Year,
    Prod_name,
    Total_Quantity,
    ROW_NUMBER() OVER (PARTITION BY Calendar_Year ORDER BY Total_Quantity DESC) AS
Rank
  FROM
    US_SALES
)
SELECT
  Calendar_Year,
  Prod_name,
  Total_Quantity
FROM
  RANKED_PRODUCTS
WHERE
  Rank = 1;
```

OUTPUT:



All Rows Fetched: 4 in 0.916 seconds

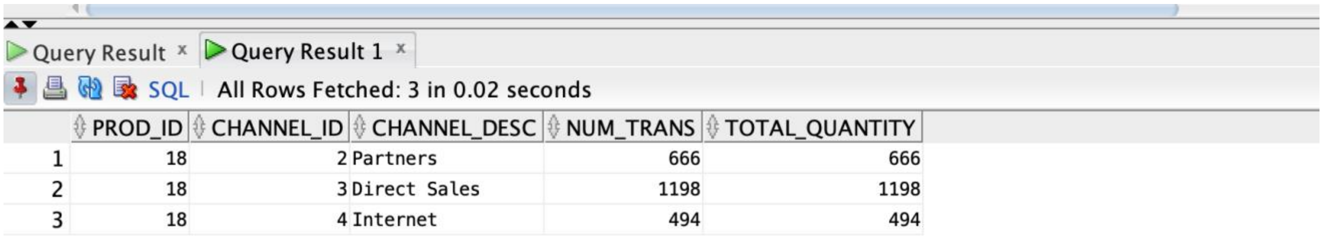
	CALENDAR_YEAR	PROD_NAME	TOTAL_QUANTITY
1	1998	O/S Documentation Set - English	4729
2	1999	Mouse Pad	4603
3	2000	1.44MB External 3.5" Diskette	4353
4	2001	Keyboard Wrist Rest	4587

Question 3 How many sales transactions were recorded for the product with the highest sales revenue in 2001? That is, the item that generated the maximum sales revenue in 2001. Also, identify the: Items sold (fir such product), as well as channel id and description.

```
-------to find the product id------
SELECT
 S.PROD_ID,
  SUM(S.AMOUNT_SOLD) AS Total_Revenue
FROM
  SH.SALES S
JOIN
  SH.TIMES T ON S.TIME_ID = T.TIME_ID
WHERE
  TO_CHAR(T.CALENDAR_YEAR) = '2001'
GROUP BY
  S.PROD_ID
ORDER BY
  Total_Revenue DESC
FETCH FIRST 1 ROW ONLY;

-------putting product id=18 retrieved from above code------
SELECT
  S.PROD_ID,
  S.CHANNEL_ID,
  C.CHANNEL_DESC,
  COUNT(*) AS NUM_TRANS,
  SUM(S.QUANTITY_SOLD) AS TOTAL_QUANTITY
FROM
  SH.SALES S
JOIN
  SH.TIMES T ON S.TIME_ID = T.TIME_ID
JOIN
  SH.CHANNELS C ON S.CHANNEL_ID = C.CHANNEL_ID
WHERE
  TO_CHAR(T.CALENDAR_YEAR) = '2001'
  AND S.PROD_ID = 18
GROUP BY
  S.PROD_ID, S.CHANNEL_ID, C.CHANNEL_DESC;
```

OUTPUT:



All Rows Fetched: 3 in 0.02 seconds

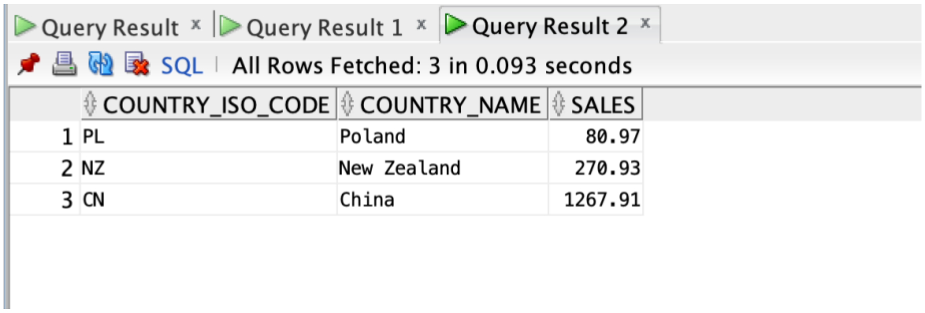
	PROD_ID	CHANNEL_ID	CHANNEL_DESC	NUM_TRANS	TOTAL_QUANTITY
1	18	2	Partners	666	666
2	18	3	Direct Sales	1198	1198
3	18	4	Internet	494	494

Question 4 Display the three countries with the poorest performance sales-wise in 1998.

```
SELECT
```

```
  C.COUNTRY_ISO_CODE,
  C.COUNTRY_NAME,
  COALESCE(SUM(S.AMOUNT_SOLD), 0) AS SALES
FROM
  SH.COUNTRIES C
LEFT JOIN
  SH.CUSTOMERS CU ON C.COUNTRY_ID = CU.COUNTRY_ID
LEFT JOIN
  SH.SALES S ON CU.CUST_ID = S.CUST_ID
LEFT JOIN
  SH.TIMES T ON S.TIME_ID = T.TIME_ID
WHERE
  TO_CHAR(T.CALENDAR_YEAR) = '1998'
GROUP BY
  C.COUNTRY_ISO_CODE, C.COUNTRY_NAME
ORDER BY
  SALES
FETCH FIRST 3 ROWS ONLY;
```

OUTPUT:



All Rows Fetched: 3 in 0.093 seconds

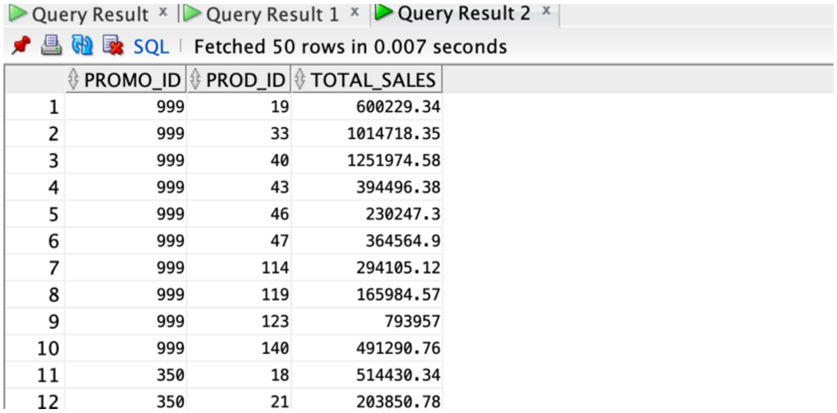
	COUNTRY_ISO_CODE	COUNTRY_NAME	SALES
1	PL	Poland	80.97
2	NZ	New Zealand	270.93
3	CN	China	1267.91

Question 5 Create an aggregated materialised view named "Promotion_Analysis_mv" presenting the sales analysis product-wise for each promotion.

```
CREATE MATERIALIZED VIEW Promotion_Analysis_mv
AS
SELECT
  S.PROMO_ID,
  S.PROD_ID,
  SUM(S.AMOUNT_SOLD) AS TOTAL_SALES
FROM
  SH.SALES S
GROUP BY
  S.PROMO_ID, S.PROD_ID;

Select * from Promotion_Analysis_mv
```

OUTPUT:



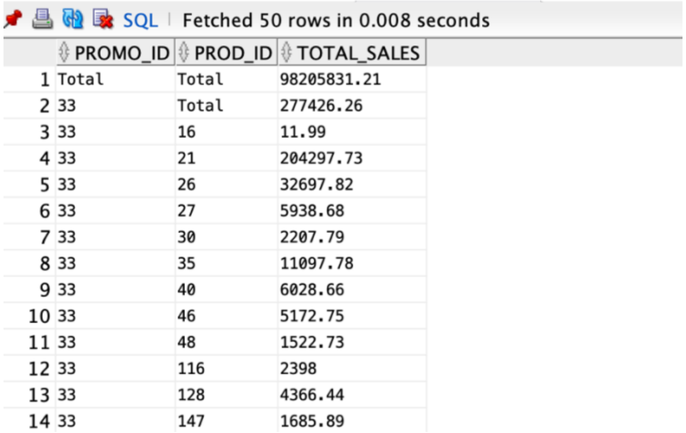
Fetched 50 rows in 0.007 seconds

	PROMO_ID	PROD_ID	TOTAL_SALES
1	999	19	600229.34
2	999	33	1014718.35
3	999	40	1251974.58
4	999	43	394496.38
5	999	46	230247.3
6	999	47	364564.9
7	999	114	294105.12
8	999	119	165984.57
9	999	123	793957
10	999	140	491290.76
11	350	18	514430.34
12	350	21	203850.78

Question 6 Use ROLLUP and CUBE on the materialized view created and provide some useful information of your choice for management.

```
-----using ROLLUP-------
SELECT
  NVL(TO_CHAR(S.PROMO_ID), 'Total') AS PROMO_ID,
  NVL(TO_CHAR(S.PROD_ID), 'Total') AS PROD_ID,
  NVL(TO_CHAR(SUM(S.TOTAL_SALES)), '0') AS TOTAL_SALES
FROM
  Promotion_Analysis_mv S
GROUP BY
  ROLLUP (S.PROMO_ID, S.PROD_ID)
ORDER BY
  S.PROMO_ID NULLS FIRST, S.PROD_ID NULLS FIRST;
```

OUTPUT:



Fetched 50 rows in 0.008 seconds

	PROMO_ID	PROD_ID	TOTAL_SALES
1	Total	Total	98205831.21
2	33	Total	277426.26
3	33	16	11.99
4	33	21	204297.73
5	33	26	32697.82
6	33	27	5938.68
7	33	30	2207.79
8	33	35	11097.78
9	33	40	6028.66
10	33	46	5172.75
11	33	48	1522.73
12	33	116	2398
13	33	128	4366.44
14	33	147	1685.89

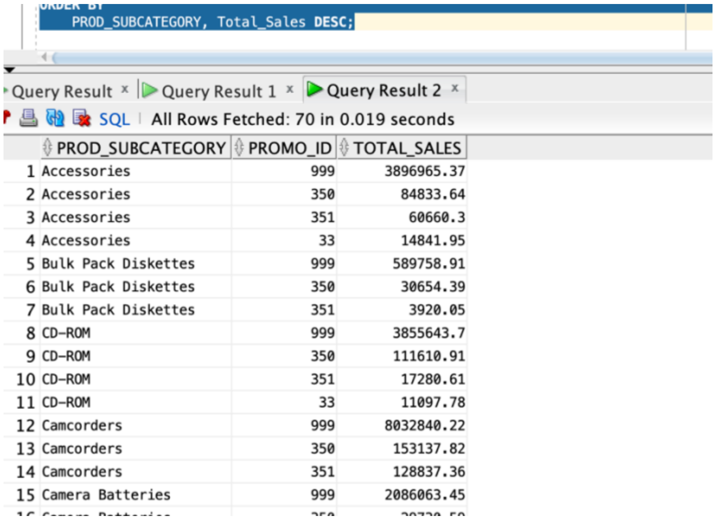
This table can help management identify which promotions are driving the most sales and which products are particularly popular during specific promotions. It can also assist in making decisions about future promotional strategies and product offerings.

Question 7 Use the aggregated table called sh.fweek_pscat_sales_mv from the SH schema. Use this table and other table(s) to provide some useful information of your choice for management.

The table sh.fweek_pscat_sales_mv provides weekly sales data by product subcategory. This data can be used to analyze the performance of different product subcategories over time. It allows for comparisons between different channels, promotions, and their impact on sales.

```
---Promotion Impact on Sales:------
SELECT
   PROD_SUBCATEGORY,
   PROMO_ID,
   SUM(DOLLARS) AS Total_Sales
FROM
   SH.FWEEK_PSCAT_SALES_MV
GROUP BY
   PROD_SUBCATEGORY, PROMO_ID
ORDER BY
   PROD_SUBCATEGORY, Total_Sales DESC;
```

OUTPUT:



```
    PROD_SUBCATEGORY, Total_Sales DESC;
```

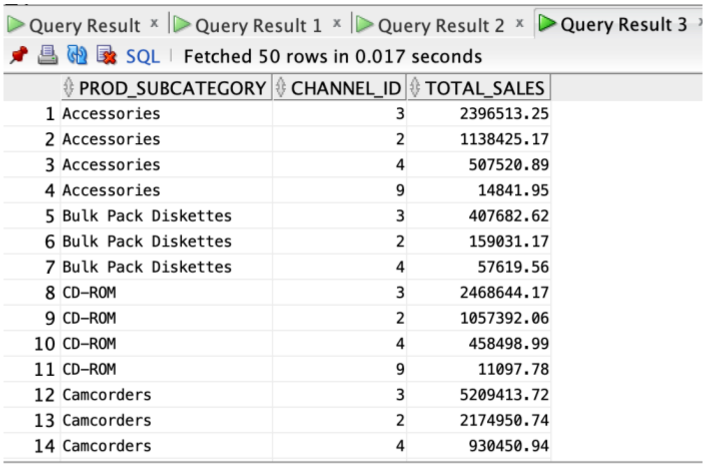
All Rows Fetched: 70 in 0.019 seconds

	PROD_SUBCATEGORY	PROMO_ID	TOTAL_SALES
1	Accessories	999	3896965.37
2	Accessories	350	84833.64
3	Accessories	351	60660.3
4	Accessories	33	14841.95
5	Bulk Pack Diskettes	999	589758.91
6	Bulk Pack Diskettes	350	30654.39
7	Bulk Pack Diskettes	351	3920.05
8	CD-ROM	999	3855643.7
9	CD-ROM	350	111610.91
10	CD-ROM	351	17280.61
11	CD-ROM	33	11097.78
12	Camcorders	999	8032840.22
13	Camcorders	350	153137.82
14	Camcorders	351	128837.36
15	Camera Batteries	999	2086063.45

-------Channel-wise Sales for Product Subcategories:------

```
SELECT
    PROD_SUBCATEGORY,
    CHANNEL_ID,
    SUM(DOLLARS) AS Total_Sales
FROM
    SH.FWEEK_PSCAT_SALES_MV
GROUP BY
    PROD_SUBCATEGORY, CHANNEL_ID
ORDER BY
    PROD_SUBCATEGORY, Total_Sales DESC;
```

OUTPUT:



Fetched 50 rows in 0.017 seconds

	PROD_SUBCATEGORY	CHANNEL_ID	TOTAL_SALES
1	Accessories	3	2396513.25
2	Accessories	2	1138425.17
3	Accessories	4	507520.89
4	Accessories	9	14841.95
5	Bulk Pack Diskettes	3	407682.62
6	Bulk Pack Diskettes	2	159031.17
7	Bulk Pack Diskettes	4	57619.56
8	CD-ROM	3	2468644.17
9	CD-ROM	2	1057392.06
10	CD-ROM	4	458498.99
11	CD-ROM	9	11097.78
12	Camcorders	3	5209413.72
13	Camcorders	2	2174950.74
14	Camcorders	4	930450.94

Management can use this Promotion Impact on Sales table and Channel-wise Sales for Product Subcategories table to make informed decisions about product promotions, inventory management, and marketing strategies. They can also track the effectiveness of specific promotions and channels in driving sales for different product subcategories. This data can ultimately contribute to optimizing sales and marketing efforts for the company.