

- **AQ1**

```
SELECT Location_Name,Time_Year,Time_Month,SUM(Quantity_Ordered*Unit_Price) AS AMT,SUM(SUM(Quantity_Ordered*Unit_Price)) OVER (Partition by Location_Name,Time_Year Order By Time_Month ROWS UNBOUNDED PRECEDING) AS CUMSUMAMT
```

```
FROM W_JOB_F,W_LOCATION_D,W_TIME_D
```

```
WHERE W_JOB_F.Location_ID = W_LOCATION_D.Location_ID AND W_TIME_D.Time_id = W_JOB_F.Contract_date
```

```
GROUP BY Location_Name,Time_Year,Time_Month;
```

The screenshot shows the Oracle SQL Developer interface. The left sidebar contains various database objects like TNS, Connections, Tables, Procedures, and Reports. The main workspace has a 'Script Builder' tab with the query code. Below it is a 'Script Output' tab showing the execution completed in 0.342 seconds. Two large tables are displayed below the output. The first table shows raw data with columns LOCATION\_NAME, TIME\_YEAR, TIME\_MONTH, AMT, and CUMSUMAMT. The second table shows the cumulative sum of AMT for each location and time period. Both tables have 312 rows selected.

LOCATION_NAME	TIME_YEAR	TIME_MONTH	AMT	CUMSUMAMT
Vancouver	2013	4	3390405	12362959
Vancouver	2013	5	3500569	16269548
Vancouver	2013	6	4330569	20605227
Vancouver	2013	7	3482838	26094250
Vancouver	2013	8	1654540	37653075
Vancouver	2013	9	1584540	37653075
Vancouver	2013	10	9674488	47367563
Vancouver	2013	11	6992444	54356644
Vancouver	2014	1	5499947	54356644
Vancouver	2014	2	11540980	11540980
Vancouver	2014	3	4952673	24105334
Vancouver	2014	4	9205490	33310824
Vancouver	2014	5	7175510	40464891
Vancouver	2014	6	3778207	56027090
Vancouver	2014	7	6378207	56027090
Vancouver	2014	8	3420514	55777511
Vancouver	2014	9	980514	55777511
Vancouver	2014	10	7714729	64356670

  

LOCATION_NAME	TIME_YEAR	TIME_MONTH	AMT	CUMSUMAMT
Vancouver	2014	11	8074263	72430923
Vancouver	2014	12	5069829	77499922
Vancouver	2015	1	4170869	62008901
Vancouver	2015	2	2030021	62008901

- **AQ2**

```
SELECT X1.Location_Name, X1.Time_Year,
       SUM(SumInvoiceAmt - TotalCosts) AS SumLocProfit,
       RANK() OVER ( PARTITION BY X1.Time_Year
                     ORDER BY ( SUM(SumInvoiceAmt - TotalCosts) ) DESC ) AS RankProfitSum
FROM BQ2 X1, BQ3 X2
WHERE X1.Job_Id = X2.Job_Id
GROUP BY X1.Location_Name, X1.Time_Year;
```

Atividades Notepad++

Applications Places Oracle SQL Developer : con2

File Edit View Navigate Run Source Team Tools Window Help

Connections con1 con2

Welcome Page con2

Query Builder

```
SELECT Location_Name, Time_Year, Time_Month, AVG(Quantity_Ordered*Unit_Price) AS AVGAMT, AVG(AVG(Quantity_Ordered*Unit_Price)) OVER (Partition by Location_Name Order By Time_Year, Time_Month ROWS 11 PRECEDING) AS AVGSUMAMT
FROM W_JOB_F, W_LOCATION_D, W_TIME_D
WHERE W_JOB_F.Location_ID = W_LOCATION_D.Location_ID AND W_TIME_D.Time_id = W_JOB_F.Contract_date
GROUP BY Location_Name, Time_Year, Time_Month;
```

Script Output

Task completed in 0.761 seconds

LOCATION_NAME	TIME_YEAR	TIME_MONTH	AVGAMT	AVGSUMAMT
Toronto	2014	5	519352.375	646233.936
Toronto	2014	6	678000.000	621424.449
Toronto	2014	7	57952.25	618438.886
Toronto	2014	8	602000.000	612333.231
Toronto	2014	9	94054.571	685223.232
Toronto	2014	10	513955.875	663092.34
Toronto	2014	11	1000000.000	650045.465
Toronto	2014	12	492863.429	657737.158
Toronto	2015	1	700000.000	603000.000
Toronto	2015	2	1084787.25	701983.723
Toronto	2015	3	522278	681724.362

  

LOCATION_NAME	TIME_YEAR	TIME_MONTH	AVGAMT	AVGSUMAMT
Vancouver	2013	1	546047.5	546047.5
Vancouver	2013	2	501169.6	673808.55
Vancouver	2013	3	611000.000	580000.000
Vancouver	2013	4	423811.875	545711.025
Vancouver	2013	5	500000.000	500000.000
Vancouver	2013	6	574714.909	524702.985
Vancouver	2013	7	386962	565028.558
Vancouver	2013	8	719188.000	565028.558
Vancouver	2013	9	514746.667	529792.358
Vancouver	2013	10	1000000.000	650000.000
Vancouver	2013	11	770880.111	576838.06

  

LOCATION_NAME	TIME_YEAR	TIME_MONTH	AVGAMT	AVGSUMAMT
Vancouver	2014	1	611000.000	611000.000
Vancouver	2014	2	887744.615	6089168.417
Vancouver	2014	3	770880.111	6089168.417
Vancouver	2014	4	250297	617127.015
Vancouver	2014	5	662715.5	637036.25
Vancouver	2014	6	650000.000	650000.000
Vancouver	2014	7	537922.9	655766.932
Vancouver	2014	8	794188.000	655766.932
Vancouver	2014	9	427564.25	665236.597
Vancouver	2014	10	28643.333	646352.986
Vancouver	2014	11	1084787.25	681724.362

  

LOCATION_NAME	TIME_YEAR	TIME_MONTH	AVGAMT	AVGSUMAMT
Vancouver	2015	1	672655.25	642432.250
Vancouver	2015	2	1245000.000	517177.177
Vancouver	2015	3	1463429.889	602721.617
Vancouver	2015	4	676673.667	584927.914

312 rows selected.

Click on an identifier with the Control key down to perform "Go to Declaration"

oracle@localhost:~

\*LER-CONFIG-PASS-HOSTS.txt (~) Data-Warehousing-for-Business-In... wd.txt [Read-Only] (/tmp/mozilla\_... Oracle SQL Developer : con2 Module4\_Analytic\_Queries\_10.pdf 1 / 4

### • AQ3

**SELECT** Location\_Name, Time\_Year, Time\_Month, AVG(Quantity\_Ordered\*Unit\_Price) AS AVGAMT, AVG(AVG(Quantity\_Ordered\*Unit\_Price)) OVER (Partition by Location\_Name Order By Time\_Year, Time\_Month ROWS 11 PRECEDING) AS AVGSUMAMT

**FROM** W\_JOB\_F, W\_LOCATION\_D, W\_TIME\_D

**WHERE** W\_JOB\_F.Location\_ID = W\_LOCATION\_D.Location\_ID AND W\_TIME\_D.Time\_id = W\_JOB\_F.Contract\_date

**GROUP BY** Location\_Name, Time\_Year, Time\_Month;

Atividades LibreOffice Writer

Applications Places Oracle SQL Developer : con2

File Edit View Navigate Run Source Team Tools Window Help

Connections con1 con2

Welcome Page con2

Query Builder

```
SELECT X1.Location_Name, X1.Time_Year,
       SUM(X1.TotalCosts) - SUM(X2.TotalCosts) AS SumLocProfit,
       RANK() OVER (PARTITION BY X1.Location_Name
                    ORDER BY (SUM(SumInvoiceAmt - TotalCosts)) DESC) AS RankProfitSum
  FROM DBO.XP_SumLocProfit
 WHERE X1.Job_Id = X2.Job_Id
 GROUP BY X1.Location_Name, X1.Time_Year;
```

Script Output

Task completed in 0.453 seconds

LOCATION_NAME	TIME_YEAR	SUMLOCPROFIT	RANKPROFITSUM
Montreal	2014	3116955.4	7
Montreal	2014	31153703.4	8
Montreal	2014	25191875.5	9
Montreal	2014	25191875.5	10

  

LOCATION_NAME	TIME_YEAR	SUMLOCPROFIT	RANKPROFITSUM
Atlanta	2014	2749603	11
Atlanta	2014	23324603	12
Atlanta	2014	23324603	13
Atlanta	2015	5790469	1
Dallas	2014	520518.24	3
Dallas	2015	438696.00	4
Seattle	2015	4084578.33	5
Baltimore	2015	3663527.54	6
New York	2015	3663527.54	7
London	2015	3663527.54	8
Denver	2015	261528.03	10
Chicago	2015	3569951.91	11
Atlanta	2015	231953.72	12

39 rows selected.

Click on an identifier with the Control key down to perform "Go to Declaration"

oracle@localhost:~

\*LER-CONFIG-PASS-HOSTS.txt (~) Data-Warehousing-for-Busin... wd.txt [Read-Only] (/tmp/mozilla\_... Oracle SQL Developer : con2 Module4\_base\_queries\_6-2... Module4\_base\_queries\_6-2... 1 / 4

- **AQ4**

```
SELECT BQ2.Location_Name, BQ2.Time_Year, BQ2.Time_Month, SUM(SUMINVAMT - TOTALCOST)/SUM(SUMINVAMT) AS ProfitMargin, RANK() OVER(Partition By BQ2.Time_Year Order by SUM(SUMINVAMT - TOTALCOST)/SUM(SUMINVAMT) DESC) AS Rank
```

**FROM** BQ2, BQ3

**WHERE** BQ2.Job\_id = BQ3.Job\_id

**GROUP BY** BQ2.Location\_Name, BQ2.Time\_Year, BQ2.Time\_Month;

The screenshot shows the Oracle SQL Developer interface with a query window containing the following code:

```
SELECT X1.Location_Name, X1.Time_Year,
       SUM(SumInvoiceAmt - TotalCosts) / Sum(SumInvoiceAmt) AS ProfitMargin,
       RANK() OVER(PARTITION BY X1.Time_Year ORDER BY (SUM(SumInvoiceAmt - TotalCosts) / SUM(SumInvoiceAmt)) DESC) AS RankProfitMargin
  FROM BQ2 X1
 WHERE X1.Job_Id = X2.Job_Id
 GROUP BY X1.Location_Name, X1.Time_Year;
```

The results are displayed in two sections:

LOCATION_NAME	TIME_YEAR	PROFITMARGIN	RANKPROFITMARGIN
Toronto	2013	.505013801	1
New York	2013	.541066201	2
Vancouver	2013	.538143672	3
Los Angeles	2013	.531239919	4
Seattle	2013	.528198819	5
Birmingham	2013	.526177073	6
Paris	2013	.525010095	7
Chicago	2013	.523701475	8
Dallas	2013	.516464005	9
Montreal	2013	.515426424	10
Montreal	2014	.51093451	11
Chicago	2014	.54106804	1
Toronto	2014	.54106804	2
Vancouver	2014	.534211169	3
Birmingham	2014	.52907257	4
Paris	2014	.52827209	5
Los Angeles	2014	.52827209	6
Seattle	2014	.527851013	7
Dallas	2014	.527851013	8
New York	2014	.52716684	9
Montreal	2015	.515426424	10
Seattle	2015	.502051379	11
London	2015	.5853114	12
Atlanta	2015	.5853114	1
Toronto	2015	.549386533	2
Montreal	2015	.549386533	3
Los Angeles	2015	.534070173	4
New York	2015	.534070173	5
London	2015	.521686965	6
Dallas	2015	.521686965	7
Vancouver	2015	.51346401	8
Denver	2015	.51346401	9

36 rows selected.

Below the results, there is a note: "Click on an identifier with the Control key down to perform "Go to Declaration""

- **AQ5**

```
SELECT X1.Job_Id, X1.Location_Name, X1.Time_Year, X1.Time_Year, (SumInvoiceAmt - TotalCosts) / SumInvoiceAmt AS ProfitMargin, PERCENT_RANK() OVER ( ORDER BY ((SumInvoiceAmt - TotalCosts) / SumInvoiceAmt) ) AS PercentRankProfitMargin
```

**FROM** BQ2 X1, BQ3 X2

**WHERE** X1.Job\_Id = X2.Job\_Id;

sab, 00:14 ◊ Oracle DB Developer VM [Executando] - Oracle VM VirtualBox

Fri 23:14

Oracle SQL Developer : con2

```

SELECT X1.Job_Id, X1.Location_Name, X1.Time_Year, X1.Time_Year,
       (SumInvoiceAmt - TotalCosts) / SumInvoiceAmt AS ProfitMargin,
       PERCENT_RANK() OVER (PARTITION BY X1.Location_Name
                             ORDER BY (SumInvoiceAmt - TotalCosts) / SumInvoiceAmt ) AS PercentRankProfitMargin
FROM BQ2 X1, BQ3 X2
WHERE X1.Job_Id = X2.Job_Id;

```

Script Output X

JOB_ID	LOCATION_NAME	TIME_YEAR	TIME_YEAR	PROFITMARGIN	PERCENTRANK	PROFITMARGIN
312374	Los Angeles	2013	2013	.600494012	.98601670	
314389	Vancouver	2014	2014	.600501838	.98641363	
312387	Toronto	2013	2013	.600501825	.98641363	
312095	Montreal	2014	2014	.600507634	.987215342	
313205	London	2014	2014	.600509474	.987034862	
312389	New York	2014	2014	.600510344	.987034862	
312803	Birmingham	2013	2013	.600527079	.988413903	
312374	Los Angeles	2013	2013	.600526367	.988813424	
314389	Vancouver	2014	2013	.600530659	.989213944	
312387	Toronto	2013	2013	.600530644	.989213944	
312095	Montreal	2014	2014	.600549663	.990013986	
313205	London	2014	2014	.600549559	.990011027	
312389	New York	2014	2014	.600549559	.990011027	
312803	Birmingham	2013	2013	.600551118	.991010068	
313782	Seattle	2013	2013	.600557487	.992009058	
314475	Seattle	2013	2013	.600557487	.992009058	
313442	Dallas	2014	2014	.600562024	.992009058	
313442	Dallas	2014	2014	.600562024	.992009058	
314475	Denver	2014	2014	.600563393	.990004794	
312389	New York	2013	2013	.600563393	.990004794	
312095	Los Angeles	2013	2013	.601080111	.996803835	
314396	Chicago	2014	2014	.60112911	.99720356	
312374	Los Angeles	2013	2013	.60112911	.99720356	
314389	Vancouver	2013	2013	.601226896	.997602877	
312387	Toronto	2014	2014	.601360258	.998401918	
312095	Montreal	2014	2014	.601406053	.998801438	
313205	London	2014	2014	.601406053	.998801438	
312389	New York	2013	2013	.601429959	.998801438	
312803	Birmingham	2013	2013	.601870787	.9996000479	
313782	Seattle	2013	2013	.601870787	.9996000479	
314475	Seattle	2013	2013	.601870787	.9996000479	
313442	Dallas	2014	2014	.601226896	.997602877	
314475	Denver	2013	2013	.601226896	.997602877	
312389	New York	2014	2014	.601226896	.997602877	
312095	Montreal	2014	2014	.601226896	.997602877	
313205	London	2014	2014	.601226896	.997602877	
312389	New York	2013	2013	.601226896	.997602877	
312803	Birmingham	2013	2013	.601226896	.997602877	
313782	Seattle	2013	2013	.601226896	.997602877	
314475	Seattle	2013	2013	.601226896	.997602877	
313442	Dallas	2014	2014	.601226896	.997602877	
314475	Denver	2013	2013	.601226896	.997602877	
312389	New York	2014	2014	.601226896	.997602877	
312095	Montreal	2014	2014	.601226896	.997602877	
313205	London	2014	2014	.601226896	.997602877	
312389	New York	2013	2013	.601226896	.997602877	
312803	Birmingham	2013	2013	.601226896	.997602877	
313782	Seattle	2013	2013	.601226896	.997602877	
314475	Seattle	2013	2013	.601226896	.997602877	
313442	Dallas	2014	2014	.601226896	.997602877	
314475	Denver	2013	2013	.601226896	.997602877	
312389	New York	2014	2014	.601226896	.997602877	
312095	Montreal	2014	2014	.601226896	.997602877	
313205	London	2014	2014	.601226896	.997602877	
312389	New York	2013	2013	.601226896	.997602877	
312803	Birmingham	2013	2013	.601226896	.997602877	
313782	Seattle	2013	2013	.601226896	.997602877	
314475	Seattle	2013	2013	.601226896	.997602877	
313442	Dallas	2014	2014	.601226896	.997602877	
314475	Denver	2013	2013	.601226896	.997602877	
312389	New York	2014	2014	.601226896	.997602877	
312095	Montreal	2014	2014	.601226896	.997602877	
313205	London	2014	2014	.601226896	.997602877	
312389	New York	2013	2013	.601226896	.997602877	
312803	Birmingham	2013	2013	.601226896	.997602877	
313782	Seattle	2013	2013	.601226896	.997602877	
314475	Seattle	2013	2013	.601226896	.997602877	
313442	Dallas	2014	2014	.601226896	.997602877	
314475	Denver	2013	2013	.601226896	.997602877	
312389	New York	2014	2014	.601226896	.997602877	
312095	Montreal	2014	2014	.601226896	.997602877	
313205	London	2014	2014	.601226896	.997602877	
312389	New York	2013	2013	.601226896	.997602877	
312803	Birmingham	2013	2013	.601226896	.997602877	
313782	Seattle	2013	2013	.601226896	.997602877	
314475	Seattle	2013	2013	.601226896	.997602877	
313442	Dallas	2014	2014	.601226896	.997602877	
314475	Denver	2013	2013	.601226896	.997602877	
312389	New York	2014	2014	.601226896	.997602877	
312095	Montreal	2014	2014	.601226896	.997602877	
313205	London	2014	2014	.601226896	.997602877	
312389	New York	2013	2013	.601226896	.997602877	
312803	Birmingham	2013	2013	.601226896	.997602877	
313782	Seattle	2013	2013	.601226896	.997602877	
314475	Seattle	2013	2013	.601226896	.997602877	
313442	Dallas	2014	2014	.601226896	.997602877	
314475	Denver	2013	2013	.601226896	.997602877	
312389	New York	2014	2014	.601226896	.997602877	
312095	Montreal	2014	2014	.601226896	.997602877	
313205	London	2014	2014	.601226896	.997602877	
312389	New York	2013	2013	.601226896	.997602877	
312803	Birmingham	2013	2013	.601226896	.997602877	
313782	Seattle	2013	2013	.601226896	.997602877	
314475	Seattle	2013	2013	.601226896	.997602877	
313442	Dallas	2014	2014	.601226896	.997602877	
314475	Denver	2013	2013	.601226896	.997602877	
312389	New York	2014	2014	.601226896	.997602877	
312095	Montreal	2014	2014	.601226896	.997602877	
313205	London	2014	2014	.601226896	.997602877	
312389	New York	2013	2013	.601226896	.997602877	
312803	Birmingham	2013	2013	.601226896	.997602877	
313782	Seattle	2013	2013	.601226896	.997602877	
314475	Seattle	2013	2013	.601226896	.997602877	
313442	Dallas	2014	2014	.601226896	.997602877	
314475	Denver	2013	2013	.601226896	.997602877	
312389	New York	2014	2014	.601226896	.997602877	
312095	Montreal	2014	2014	.601226896	.997602877	
313205	London	2014	2014	.601226896	.997602877	
312389	New York	2013	2013	.601226896	.997602877	
312803	Birmingham	2013	2013	.601226896	.997602877	
313782	Seattle	2013	2013	.601226896	.997602877	
314475	Seattle	2013	2013	.601226896	.997602877	
313442	Dallas	2014	2014	.601226896	.997602877	
314475	Denver	2013	2013	.601226896	.997602877	
312389	New York	2014	2014	.601226896	.997602877	
312095	Montreal	2014	2014	.601226896	.997602877	
313205	London	2014	2014	.601226896	.997602877	
312389	New York	2013	2013	.601226896	.997602877	
312803	Birmingham	2013	2013	.601226896	.997602877	
313782	Seattle	2013	2013	.601226896	.997602877	
314475	Seattle	2013	2013	.601226896	.997602877	
313442	Dallas	2014	2014	.601226896	.997602877	
314475	Denver	2013	2013	.601226896	.997602877	
312389	New York	2014	2014	.601226896	.997602877	
312095	Montreal	2014	2014	.601226896	.997602877	
313205	London	2014	2014	.601226896	.997602877	
312389	New York	2013	2013	.601226896	.997602877	
312803	Birmingham	2013	2013	.601226896	.997602877	
313782	Seattle	2013	2013	.601226896	.997602877	
314475	Seattle	2013	2013	.601226896	.997602877	
313442	Dallas	2014	2014	.601226896	.997602877	
314475	Denver	2013	2013	.601226896	.997602877	
312389	New York	2014	2014	.601226896	.997602877	
312095	Montreal	2014	2014	.601226896	.997602877	
313205	London	2014	2014	.601226896	.997602877	
312389	New York	2013	2013	.601226896	.997602877	
312803	Birmingham	2013	2013	.601226896	.997602877	
313782	Seattle	2013	2013	.601226896	.997602877	
314475	Seattle	2013	2013	.601226896	.997602877	
313442	Dallas	2014	2014	.601226896	.997602877	
314475	Denver	2013	2013	.601226896	.997602877	
312389	New York	2014	2014	.601226896	.997602877	
312095	Montreal	2014	2014	.601226896	.997602877	
313205	London	2014	2014	.601226896	.997602877	
312389	New York	2013	2013	.601226896	.997602877	
312803	Birmingham	2013	2013	.601226896	.997602877	
313782	Seattle	2013	2013	.601226896	.997602877	
314475	Seattle	2013	2013	.601226896	.997602877	
313442	Dallas	2014	2014	.601226896	.997602877	
314475	Denver	2013	2013	.601226896	.997602877	
312389	New York	2014	2014	.601226896	.997602877	
312095	Montreal	2014	2014	.601226896	.997602877	
313205	London	2014	2014	.601226896	.997602877	
312389	New York	2013	20			

- **AQ7**

**SELECT** Sales\_Class\_Desc, Time\_Year, SUM ( quantity\_shipped - invoice\_quantity ) as ReturnSum , RANK() over ( PARTITION BY Time\_Year

**ORDER BY** SUM ( quantity\_shipped - invoice\_quantity ) DESC ) AS RankReturnSum

**FROM** W\_INVOICELINE\_F INNER JOIN W\_TIME\_D  
ON W\_INVOICELINE\_F.INVOICE\_SENT\_DATE = W\_TIME\_D.TIME\_ID

**INNER JOIN** W\_Sales\_Class\_D  
ON W\_INVOICELINE\_F.Sales\_Class\_Id = W\_Sales\_Class\_D.Sales\_Class\_Id

**WHERE** quantity\_shipped > invoice\_quantity

**GROUP BY** Sales\_Class\_Desc, Time\_Year;

```

SELECT Sales_Class_Desc, Time_Year,
       SUM ( quantity_shipped - invoice_quantity ) as ReturnSum ,
       RANK() over ( PARTITION BY Time_Year
                     ORDER BY quantity_shipped - invoice_quantity ) DESC ) AS RankReturnSum
FROM W_INVOICELINE_F INNER JOIN W_TIME_D
ON W_INVOICELINE_F.INVOICE_SENT_DATE = W_TIME_D.TIME_ID
INNER JOIN W_Sales_Class_D
ON W_INVOICELINE_F.Sales_Class_Id = W_Sales_Class_D.Sales_Class_Id
WHERE quantity_shipped > invoice_quantity
GROUP BY Sales_Class_Desc, Time_Year;

```

Script Output:

```

SALES_CLASS_DESC
-----
Debit NoSmart
Prepaid NoSmart
Debit Smart
Credit NoSmart
Loyalty NoSmart
Credit Smart
Debit NoSmart
Debit Smart
Credit Smart
SALES_CLASS_DESC
-----
Debit NoSmart
Prepaid NoSmart
Debit Smart
Credit NoSmart
Loyalty NoSmart
Debit NoSmart
Debit Smart
Credit Smart
18 rows selected.

```

- **AQ8**

**SELECT** Time\_Year, Sales\_Class\_Desc, SUM ( quantity\_shipped - invoice\_quantity ) as SumReturnQty, Ratio\_To\_Report(SUM ( quantity\_shipped - invoice\_quantity ))  
OVER ( PARTITION BY Time\_Year ) AS RatioReturnSum

**FROM** W\_INVOICELINE\_F INNER JOIN W\_TIME\_D  
ON W\_INVOICELINE\_F.INVOICE\_SENT\_DATE = W\_TIME\_D.TIME\_ID

**INNER JOIN** W\_Sales\_Class\_D  
ON W\_INVOICELINE\_F.Sales\_Class\_Id = W\_Sales\_Class\_D.Sales\_Class\_Id

**WHERE** quantity\_shipped > invoice\_quantity

**GROUP BY** Sales\_Class\_Desc, Time\_Year

**ORDER BY** Time\_Year, SUM( quantity\_shipped - invoice\_quantity );

```

SELECT Time_Year, Sales_Class_Desc,
       Ratio_ReportSum ( quantity_shipped / invoice.quantity ) AS RatioReturnSum
  FROM W_INVOICELINE_F INNER JOIN W_INVOICECHI_F
    ON W_INVOICELINE_F.INVOICE_ID = W_INVOICECHI_F.INVOICE_ID
   WHERE Sales_Class_Id = W_Sales_Class.D.Sales_Class_Id
  GROUP BY Sales_Class_Desc, Time_Year
  ORDER BY Time_Year, SUM( quantity_shipped - invoice.quantity );

```

Script Output: Task completed in 0.161 seconds

TIME_YEAR	SALES_CLASS_DESC	RatioReportSum
2013	Credit NoSmart	
2013	Loyalty NoSmart	
2013	Debit Smart	
2013	Debit NoSmart	
2013	Prepaid Smart	
2013	Prepaid NoSmart	
2014	Credit Smart	
2014	Credit NoSmart	
2014	Debit Smart	
2014	Debit NoSmart	
2014	Loyalty NoSmart	

18 rows selected.

### • AQ9

**SELECT Location\_Name, W\_Time\_D.Time\_Year, SUM(BusDaysDiff) as SumDelayDays,  
RANK() OVER ( PARTITION BY W\_Time\_D.Time\_Year  
                  ORDER BY SUM(BusDaysDiff) DESC) AS RankSumDelayDays, DENSE\_RANK()  
                  OVER ( PARTITION BY W\_Time\_D.Time\_Year ORDER BY SUM(BusDaysDiff) DESC)  
AS RankSumDelayDays**

**FROM BQ6, W\_Time\_D**

**WHERE W\_Time\_D.Time\_Id = BQ6.Date\_Ship\_By**

**GROUP BY Location\_Name, W\_Time\_D.Time\_Year;**

```

SELECT Location_Name, W_Time_D.Time_Year,
       SUM(BusDaysDiff) AS SumDelayDays,
       RANK() OVER ( PARTITION BY W_Time_D.Time_Year
                     ORDER BY SUM(BusDaysDiff) DESC) AS RankSumDelayDays,
       DENSE_RANK() OVER ( PARTITION BY W_Time_D.Time_Year
                     ORDER BY SUM(BusDaysDiff) DESC) AS DenseRankSumDelayDays
  FROM BQ6, W_Time_D
 WHERE W_Time_D.Time_Id = BQ6.Date_Ship_By
 GROUP BY Location_Name, W_Time_D.Time_Year;

```

LOCATION_NAME	TIME_YEAR	SUMDELAYDAYS	RANKSUMDELAYDAYS	DENSE_RANKSUMDELAYDAYS
Seattle	2013	24	1	1
Montreal	2013	20	2	2
Toronto	2013	15	3	3
Chicago	2013	12	4	4
Dallas	2013	9	5	5
Los Angeles	2013	8	6	6
Boston	2013	7	7	7
Toronto	2013	4	8	8
New York	2013	2	9	9
Baltimore	2014	29	1	1
Birmingham	2014	26	2	2

LOCATION_NAME	TIME_YEAR	SUMDELAYDAYS	RANKSUMDELAYDAYS	DENSE_RANKSUMDELAYDAYS
Los Angeles	2013	19	3	1
Toronto	2014	17	4	4
New York	2014	10	6	5
Dallas	2014	9	7	6
Montreal	2014	7	8	7
Atlanta	2014	4	9	8
Chicago	2014	2	10	9
Denver	2014	2	10	9
Montevideo	2015	8	1	1
Los Angeles	2015	6	2	2

LOCATION_NAME	TIME_YEAR	SUMDELAYDAYS	RANKSUMDELAYDAYS	DENSE_RANKSUMDELAYDAYS
Dallas	2015	5	3	3
Montreal	2015	5	5	4
Atlanta	2015	3	5	4
Baltimore	2015	2	5	4
New York	2015	2	5	4

27 rows selected.

- **BQ10**

```
SELECT Location_Name, W_Time_D.Time_Year, COUNT(*) AS NumJobs, SUM(BusDaysDiff)
as SumDelayDays, SUM(Quantity_Ordered - SumDelayShipQty) / SUM(Quantity_Ordered) AS
PromisedDelayRate, RANK() OVER ( PARTITION BY W_Time_D.Time_Year
```

```
ORDER BY SUM(Quantity_Ordered - SumDelayShipQty)/SUM(Quantity_Ordered) DESC) AS
RankDelayRate
```

**FROM** BQ5, W\_Time\_D

**WHERE** W\_Time\_D.Time\_Id = BQ5.Date\_Promised

**GROUP BY** Location\_Name, W\_Time\_D.Time\_Year;

The screenshot shows the Oracle SQL Developer interface with a query window open. The query is:

```
SELECT Location_Name, W_Time_D.Time_Year,
       COUNT(*) AS NumJobs,
       SUM(BusDaysDiff) AS SumDelayDays,
       SUM(Quantity_Ordered - SumDelayShipQty) / SUM(Quantity_Ordered) AS
PromisedDelayRate,
       RANK() OVER ( PARTITION BY W_Time_D.Time_Year
                     ORDER BY SUM(Quantity_Ordered - SumDelayShipQty) /
                           SUM(Quantity_Ordered) DESC) AS
RankDelayRate
  FROM BQ5, W_Time_D
 WHERE W_Time_D.Time_Id = BQ5.Date_Promised
 GROUP BY Location_Name, W_Time_D.Time_Year;
```

The results are displayed in three tables:

LOCATION_NAME	TIME_YEAR	NUMJOBS	SUMDELAYDAYS	PROMISEDELAYRATE	RANKDELAYRATE
Chicago	2013	2	7	.68319378	1
New York	2013	1	3	.650158479	2
Los Angeles	2013	3	15	.37404646	4
Montreal	2013	2	4	.317172659	5
Toronto	2013	6	36	.303047551	6
Austin	2013	1	18	.14205022	7
Atlanta	2013	1	4	.117450466	8
Dallas	2013	1	9	.062469206	9
Montgomery	2013	1	3	.89495783	1
Vancouver	2014	5	22	.374317233	8
Seattle	2014	4	25	.454927727	6
Birmingham	2014	1	6	.49321022	7
Philadelphia	2014	5	29	.279954516	9
Tampa	2014	4	22	.28920522	10
Chicago	2014	1	4	.13130793	11
Seattle	2015	1	5	.88407177	1
Dallas	2015	2	7	.760125039	2

  

LOCATION_NAME	TIME_YEAR	NUMJOBS	SUMDELAYDAYS	PROMISEDELAYRATE	RANKDELAYRATE
New York	2014	2	3	.570137113	3
Seattle	2014	11	41	.530241016	4
Montreal	2014	2	4	.202010202	5
Birmingham	2014	4	25	.454927727	6
Philadelphia	2014	1	6	.49321022	7
Vancouver	2014	5	22	.374317233	8
Seattle	2014	5	29	.279954516	9
Tampa	2014	4	22	.28920522	10
Chicago	2014	1	4	.13130793	11
Montgomery	2015	1	5	.88407177	1
Dallas	2015	2	7	.760125039	2

  

LOCATION_NAME	TIME_YEAR	NUMJOBS	SUMDELAYDAYS	PROMISEDELAYRATE	RANKDELAYRATE
New York	2015	1	1	.659947242	3
Atlanta	2015	1	4	.354695149	4
Vancouver	2015	2	16	.224699	5
Montreal	2015	1	7	.0261056511	6
Los Angeles	2015	1	4	.000320815303	7

27 rows selected.