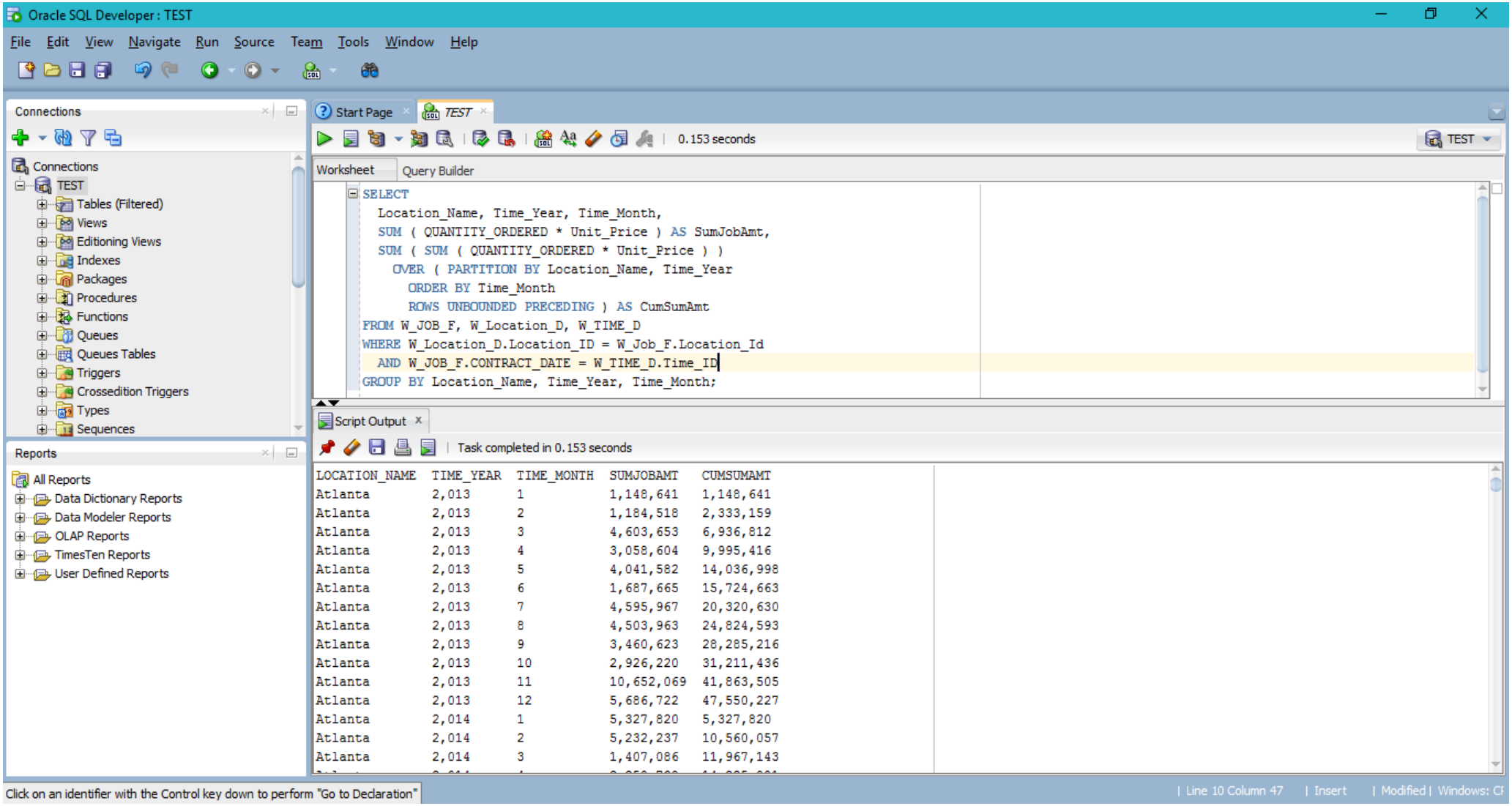
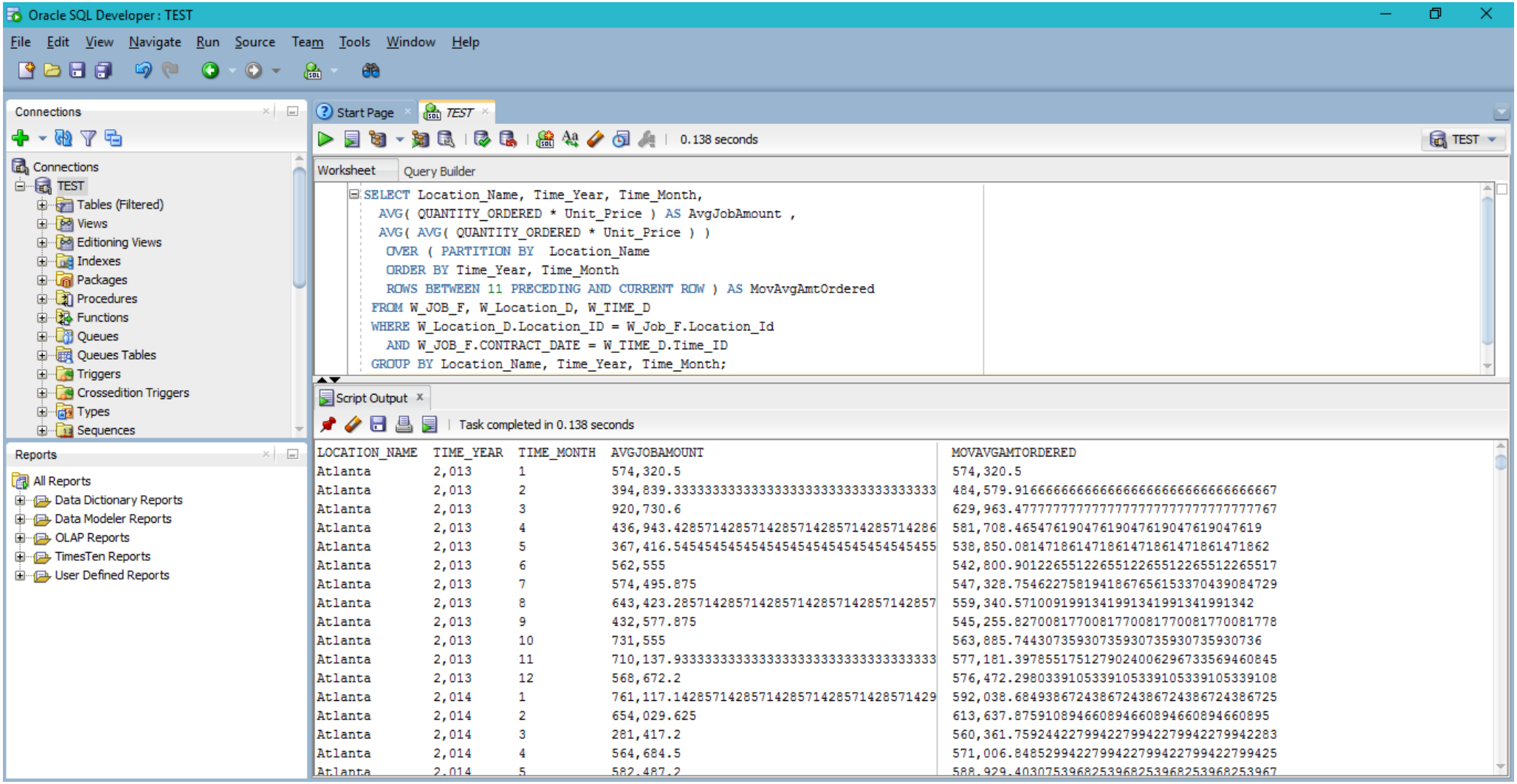
AQ1: Cumulative quantity for locations

```
SELECT
  Location_Name, Time_Year, Time_Month,
  SUM ( QUANTITY_ORDERED * Unit_Price ) AS SumJobAmt,
  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_Location_D.Location_ID = W_Job_F.Location_Id
  AND W_JOB_F.CONTRACT_DATE = W_TIME_D.Time_ID
GROUP BY Location_Name, Time_Year, Time_Month;
```


Oracle SQL Developer : TEST
File Edit View Navigate Run Source Team Tools Window Help
Connections
Connections
TEST
Tables (Filtered)
Views
Editioning Views
Indexes
Packages
Procedures
Functions
Queues
Queues Tables
Triggers
Crossedition Triggers
Types
Sequences
Reports
All Reports
Data Dictionary Reports
Data Modeler Reports
OLAP Reports
TimesTen Reports
User Defined Reports
Start Page
TEST
0.153 seconds
TEST
Worksheet
Query Builder
SELECT
Location_Name, Time_Year, Time_Month,
SUM (QUANTITY_ORDERED * Unit_Price) AS SumJobAmt,
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_Location_D.Location_ID = W_Job_F.Location_Id
AND W_JOB_F.CONTRACT_DATE = W_TIME_D.Time_ID
GROUP BY Location_Name, Time_Year, Time_Month;
Script Output
Task completed in 0.153 seconds
LOCATION_NAME TIME_YEAR TIME_MONTH SUMJOBAMT CUMSUMAMT
Atlanta 2,013 1 1,148,641 1,148,641
Atlanta 2,013 2 1,184,518 2,333,159
Atlanta 2,013 3 4,603,653 6,936,812
Atlanta 2,013 4 3,058,604 9,995,416
Atlanta 2,013 5 4,041,582 14,036,998
Atlanta 2,013 6 1,687,665 15,724,663
Atlanta 2,013 7 4,595,967 20,320,630
Atlanta 2,013 8 4,503,963 24,824,593
Atlanta 2,013 9 3,460,623 28,285,216
Atlanta 2,013 10 2,926,220 31,211,436
Atlanta 2,013 11 10,652,069 41,863,505
Atlanta 2,013 12 5,686,722 47,550,227
Atlanta 2,014 1 5,327,820 5,327,820
Atlanta 2,014 2 5,232,237 10,560,057
Atlanta 2,014 3 1,407,086 11,967,143
Click on an identifier with the Control key down to perform "Go to Declaration"
Line 10 Column 47
Insert
Modified
Windows: CF

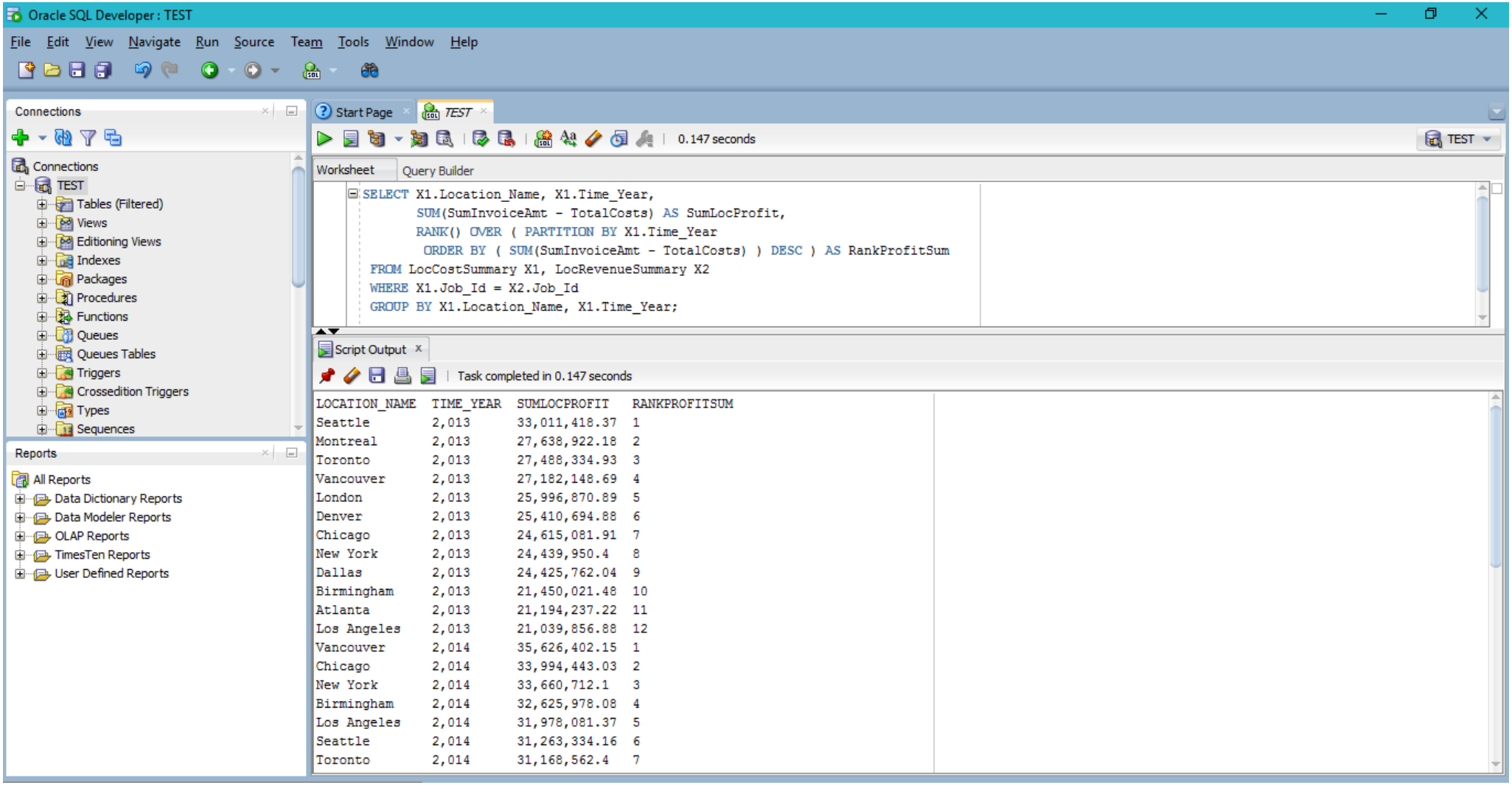
AQ2: Moving average of average amount ordered for locations

```
SELECT Location_Name, Time_Year, Time_Month,
  AVG( QUANTITY_ORDERED * Unit_Price ) AS AvgJobAmount ,
  AVG( AVG( QUANTITY_ORDERED * Unit_Price ) )
   OVER ( PARTITION BY  Location_Name
   ORDER BY Time_Year, Time_Month
   ROWS BETWEEN 11 PRECEDING AND CURRENT ROW ) AS MovAvgAmtOrdered
 FROM W_JOB_F, W_Location_D, W_TIME_D
 WHERE W_Location_D.Location_ID = W_Job_F.Location_Id
   AND W_JOB_F.CONTRACT_DATE = W_TIME_D.Time_ID
 GROUP BY Location_Name, Time_Year, Time_Month;
```


Oracle SQL Developer : TEST
File Edit View Navigate Run Source Team Tools Window Help
Connections
Connections
TEST
Tables (Filtered)
Views
Editioning Views
Indexes
Packages
Procedures
Functions
Queues
Queues Tables
Triggers
Crossedition Triggers
Types
Sequences
Reports
All Reports
Data Dictionary Reports
Data Modeler Reports
OLAP Reports
TimesTen Reports
User Defined Reports
Start Page
TEST
0.138 seconds
TEST
Worksheet
Query Builder
SELECT Location_Name, Time_Year, Time_Month,
AVG(QUANTITY_ORDERED * Unit_Price) AS AvgJobAmount ,
AVG(AVG(QUANTITY_ORDERED * Unit_Price))
OVER (PARTITION BY Location_Name
ORDER BY Time_Year, Time_Month
ROWS BETWEEN 11 PRECEDING AND CURRENT ROW) AS MovAvgAmtOrdered
FROM W_JOB_F, W_Location_D, W_TIME_D
WHERE W_Location_D.Location_ID = W_Job_F.Location_Id
AND W_JOB_F.CONTRACT_DATE = W_TIME_D.Time_ID
GROUP BY Location_Name, Time_Year, Time_Month;
Script Output
Task completed in 0.138 seconds
LOCATION_NAME TIME_YEAR TIME_MONTH AVGJOBAMOUNT MOVAVGAMTORDERED
Atlanta 2,013 1 574,320.5 574,320.5
Atlanta 2,013 2 394,839.33333333333333333333333333333333 484,579.91666666666666666666666666666667
Atlanta 2,013 3 920,730.6 629,963.47777777777777777777777777777767
Atlanta 2,013 4 436,943.42857142857142857142857142857142 581,708.46547619047619047619047619047619
Atlanta 2,013 5 367,416.54545454545454545454545454545455 538,850.08147186147186147186147186147862
Atlanta 2,013 6 562,555 542,800.90122655122655122655122655122655517
Atlanta 2,013 7 574,495.875 547,328.75462275819418676561533704390847 29
Atlanta 2,013 8 643,423.28571428571428571428571428571428 57 559,340.57100919913419913419913419913342
Atlanta 2,013 9 432,577.875 545,255.82700817700817700817700817700817778
Atlanta 2,013 10 731,555 563,885.74430735930735930735930735930736
Atlanta 2,013 11 710,137.93333333333333333333333333333333 577,181.39785517512790240062967335694608 45
Atlanta 2,013 12 568,672.2 576,472.29803391053391053391053391053391 08
Atlanta 2,014 1 761,117.14285714285714285714285714285714 29 592,038.68493867243867243867243867243867 25
Atlanta 2,014 2 654,029.625 613,637.87591089466089466089466089466089 5
Atlanta 2,014 3 281,417.2 560,361.75924422799422799422799422799422 83
Atlanta 2,014 4 564,684.5 571,006.84852994227994227994227994227994 25
Atlanta 2,014 5 582,487.2 588,929.40307539682539682539682539682539 67

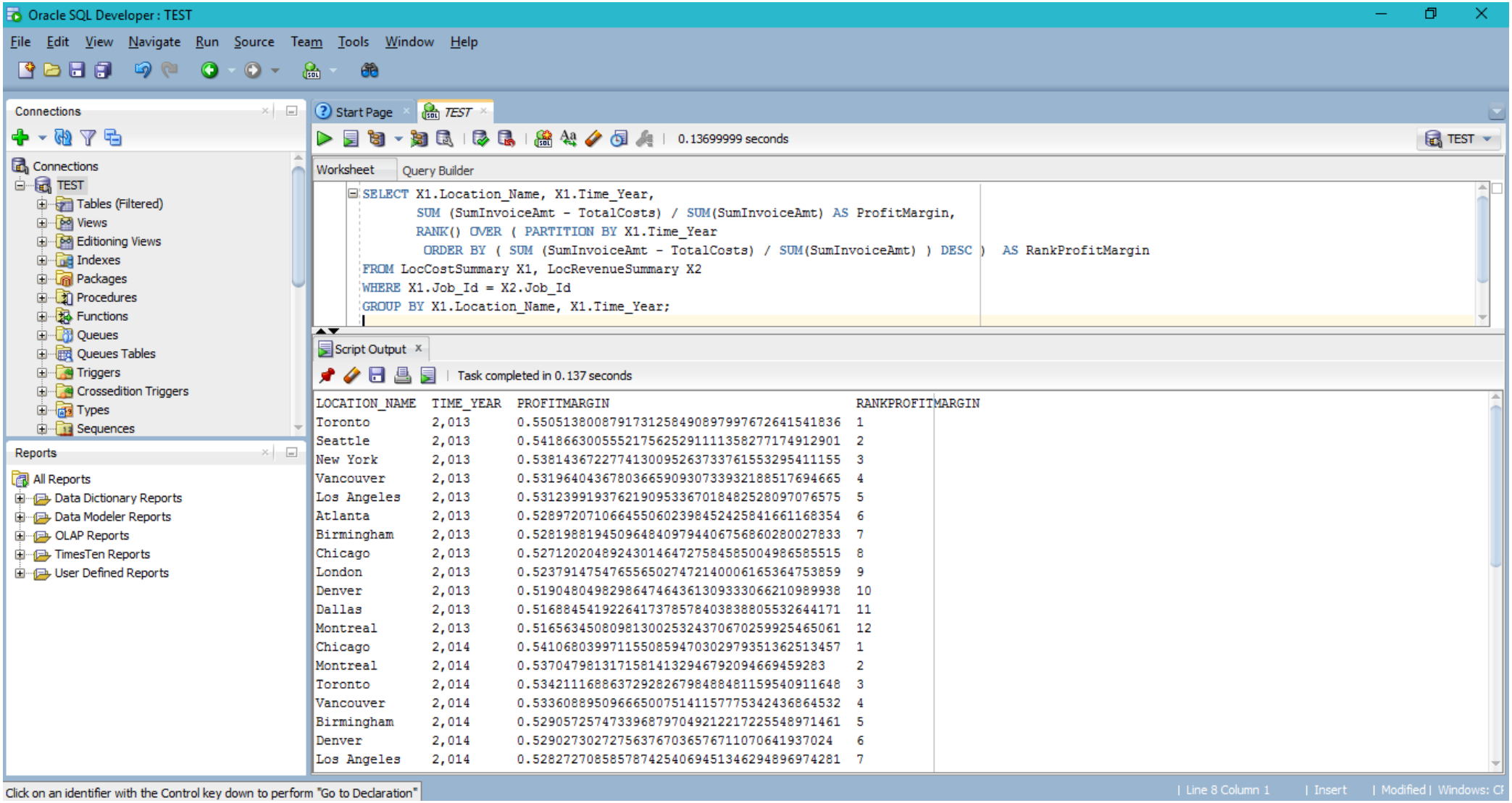
AQ3: Rank locations by descending sum of annual profit

```
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 LocCostSummary X1, LocRevenueSummary X2
WHERE X1.Job_Id = X2.Job_Id
GROUP BY X1.Location_Name, X1.Time_Year;
```


Oracle SQL Developer : TEST
File Edit View Navigate Run Source Team Tools Window Help
Connections
Connections
TEST
Tables (Filtered)
Views
Editioning Views
Indexes
Packages
Procedures
Functions
Queues
Queues Tables
Triggers
Crossedition Triggers
Types
Sequences
Reports
All Reports
Data Dictionary Reports
Data Modeler Reports
OLAP Reports
TimesTen Reports
User Defined Reports
Start Page
TEST
0.147 seconds
TEST
Worksheet
Query Builder
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 LocCostSummary X1, LocRevenueSummary X2
WHERE X1.Job_Id = X2.Job_Id
GROUP BY X1.Location_Name, X1.Time_Year;
Script Output
Task completed in 0.147 seconds
LOCATION_NAME TIME_YEAR SUMLOCPROFIT RANKPROFITSUM
Seattle 2,013 33,011,418.37 1
Montreal 2,013 27,638,922.18 2
Toronto 2,013 27,488,334.93 3
Vancouver 2,013 27,182,148.69 4
London 2,013 25,996,870.89 5
Denver 2,013 25,410,694.88 6
Chicago 2,013 24,615,081.91 7
New York 2,013 24,439,950.4 8
Dallas 2,013 24,425,762.04 9
Birmingham 2,013 21,450,021.48 10
Atlanta 2,013 21,194,237.22 11
Los Angeles 2,013 21,039,856.88 12
Vancouver 2,014 35,626,402.15 1
Chicago 2,014 33,994,443.03 2
New York 2,014 33,660,712.1 3
Birmingham 2,014 32,625,978.08 4
Los Angeles 2,014 31,978,081.37 5
Seattle 2,014 31,263,334.16 6
Toronto 2,014 31,168,562.4 7

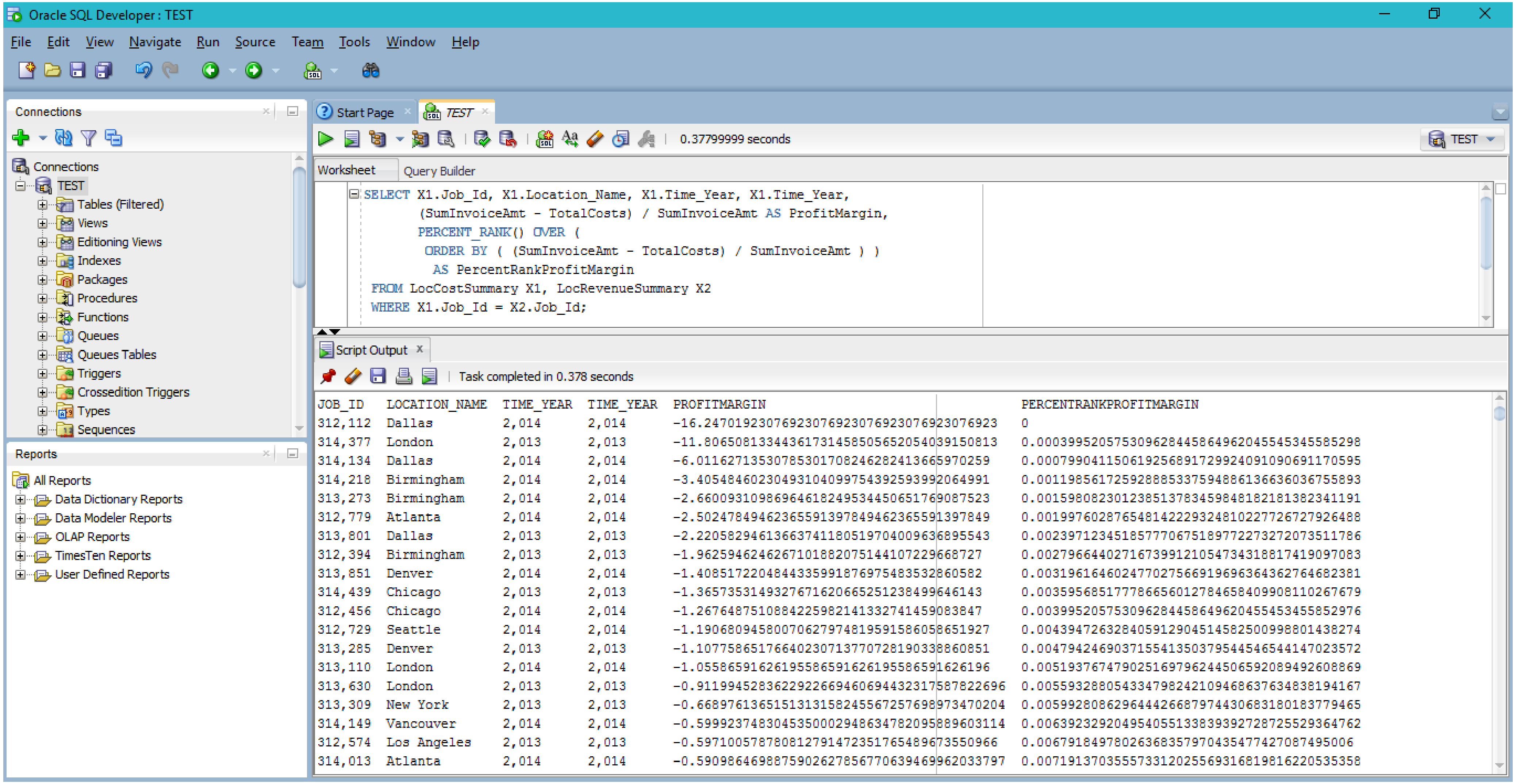
AQ4: Rank locations by descending annual profit margin

```
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 LocCostSummary X1, LocRevenueSummary X2
WHERE X1.Job_Id = X2.Job_Id
GROUP BY X1.Location_Name, X1.Time_Year;
```


Oracle SQL Developer : TEST
File Edit View Navigate Run Source Team Tools Window Help
Connections
Connections
TEST
Tables (Filtered)
Views
Editioning Views
Indexes
Packages
Procedures
Functions
Queues
Queues Tables
Triggers
Crossedition Triggers
Types
Sequences
Reports
All Reports
Data Dictionary Reports
Data Modeler Reports
OLAP Reports
TimesTen Reports
User Defined Reports
Start Page
TEST
0.13699999 seconds
TEST
Worksheet
Query Builder
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 LocCostSummary X1, LocRevenueSummary X2
WHERE X1.Job_Id = X2.Job_Id
GROUP BY X1.Location_Name, X1.Time_Year;
Script Output
Task completed in 0.137 seconds
LOCATION_NAME TIME_YEAR PROFITMARGIN RANKPROFITMARGIN
Toronto 2,013 0.550513800879173125849089799767264154183 6 1
Seattle 2,013 0.541866300555217562529111135827717491290 1 2
New York 2,013 0.538143672277413009526373376155329541115 5 3
Vancouver 2,013 0.531964043678036659093073393218851769466 5 4
Los Angeles 2,013 0.531239919376219095336701848252809707657 5 5
Atlanta 2,013 0.528972071066455060239845242584166116835 4 6
Birmingham 2,013 0.528198819450964840979440675686028002783 3 7
Chicago 2,013 0.527120204892430146472758458500498658551 5 8
London 2,013 0.523791475476556502747214000616536475385 9 9
Denver 2,013 0.519048049829864746436130933306621098993 8 10
Dallas 2,013 0.516884541922641737857840383880553264417 1 11
Montreal 2,013 0.516563450809813002532437067025992546506 1 12
Chicago 2,014 0.541068039971155085947030297935136251345 7 1
Montreal 2,014 0.537047981317158141329467920946694592 83 2
Toronto 2,014 0.534211168863729282679848848115954091164 8 3
Vancouver 2,014 0.533608895096665007514115777534243686453 2 4
Birmingham 2,014 0.529057257473396879704921221722554897146 1 5
Denver 2,014 0.529027302727563767036576711070641937024 6
Los Angeles 2,014 0.528272708585787425406945134629489697428 1 7
Click on an identifier with the Control key down to perform "Go to Declaration"
Line 8 Column 1
Insert
Modified
Windows: Cf

AQ5: Percent rank of job profit margins for locations

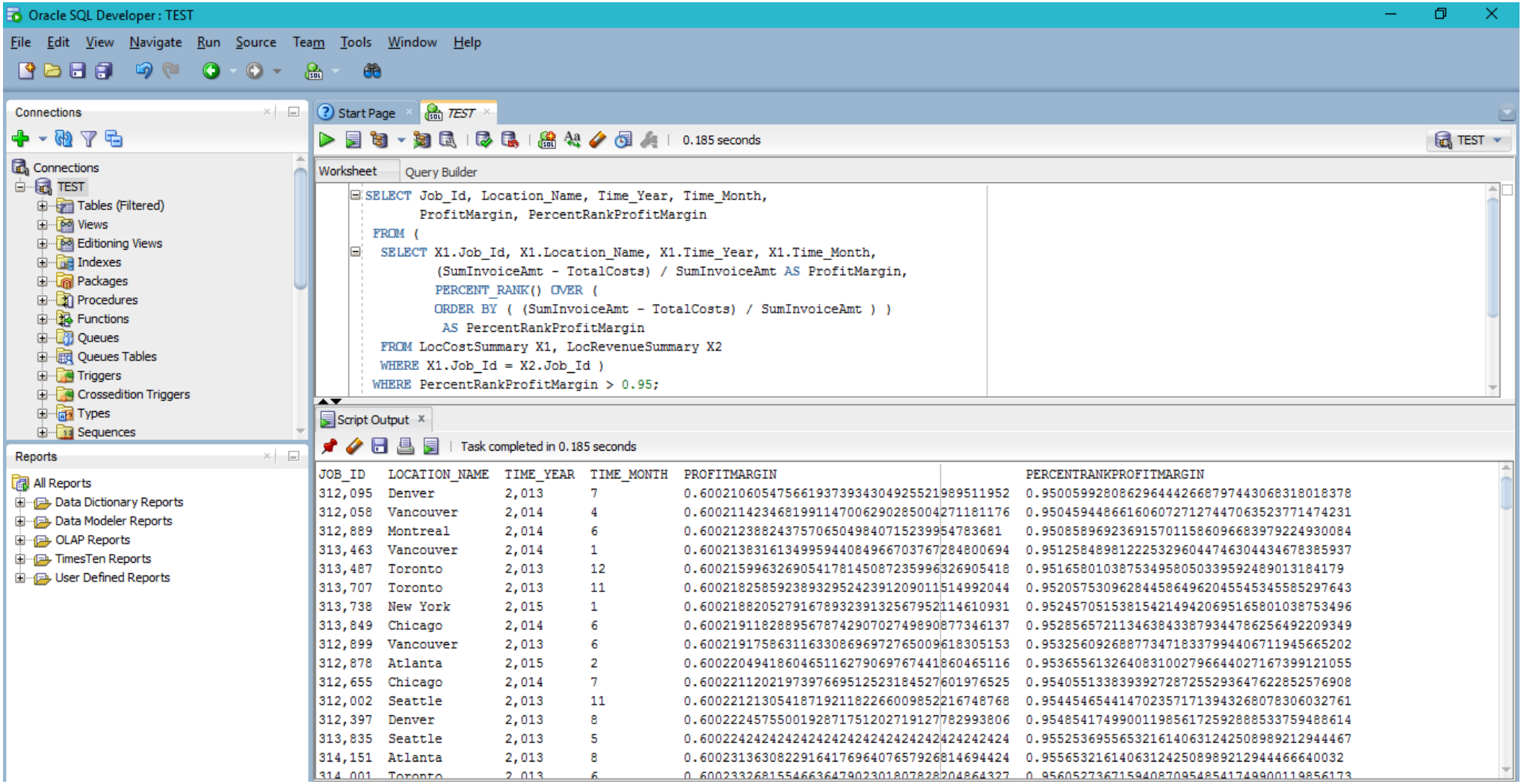
```
SELECT X1.Job_Id, X1.Location_Name, X1.Time_Year, X1.Time_Year,
       (SumInvoiceAmt - TotalCosts) / SumInvoiceAmt AS ProfitMargin,
       PERCENT_RANK() OVER (
        ORDER BY ( (SumInvoiceAmt - TotalCosts) / SumInvoiceAmt ) )
         AS PercentRankProfitMargin
 FROM LocCostSummary X1, LocRevenueSummary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



Oracle SQL Developer : TEST
File Edit View Navigate Run Source Team Tools Window Help
Connections
TEST
Tables (Filtered)
Views
Editioning Views
Indexes
Packages
Procedures
Functions
Queues
Queues Tables
Triggers
Crossedition Triggers
Types
Sequences
Reports
All Reports
Data Dictionary Reports
Data Modeler Reports
OLAP Reports
TimesTen Reports
User Defined Reports
Start Page
TEST
0.37799999 seconds
Worksheet
Query Builder
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 LocCostSummary X1, LocRevenueSummary X2
WHERE X1.Job_Id = X2.Job_Id;
Script Output
Task completed in 0.378 seconds
JOB_ID LOCATION_NAME TIME_YEAR TIME_YEAR PROFITMARGIN PERCENTRANKPROFITMARGIN
312,112 Dallas 2,014 2,014 -16.2470192307692307692307692307692307692307 0
314,377 London 2,013 2,013 -11.806508133443617314585056520540391508130 0.0003995205753096284458649620455453455852980
314,134 Dallas 2,014 2,014 -6.0116271353078530170824628241366597025900 0.0007990411506192568917299240910906911705950
314,218 Birmingham 2,014 2,014 -3.4054846023049310409975439259399206499100 0.0011985617259288853375948861366360367558930
313,273 Birmingham 2,014 2,014 -2.6600931098696461824953445065176908752300 0.0015980823012385137834598481821813823411910
312,779 Atlanta 2,014 2,014 -2.5024784946236559139784946236559139784900 0.0019976028765481422293248102277267279264880
313,801 Dallas 2,013 2,013 -2.2205829461366374118051970400963689554300 0.0023971234518577706751897722732720735117860
312,394 Birmingham 2,013 2,013 -1.9625946246267101882075144107229668727 0.0027966440271673991210547343188174190970830
313,851 Denver 2,014 2,014 -1.4085172204844335991876975483532860582 0.0031961646024770275669196963643627646823810
314,439 Chicago 2,013 2,013 -1.3657353149327671620665251238499646143 0.0035956851777866560127846584099081102676790
312,456 Chicago 2,014 2,014 -1.2676487510884225982141332741459083847 0.0039952057530962844586496204554534558529760
312,729 Seattle 2,014 2,014 -1.19068094580070627974819591586058651927 0.0043947263284059129045145825009988014382740
313,285 Denver 2,013 2,013 -1.10775865176640230713770728190338860851 0.0047942469037155413503795445465441470235720
313,110 London 2,014 2,014 -1.05586591626195586591626195586591626196 0.0051937674790251697962445065920894926088690
313,630 London 2,013 2,013 -0.911994528362292266946069443231758782269 0.0055932880543347982421094686376348381941670
313,309 New York 2,013 2,013 -0.668976136515131315824556725769897347020 0.0059928086296444266879744306831801837794650
314,149 Vancouver 2,014 2,014 -0.599923748304535000294863478209588960311 0.0063923292049540551338393927287255293647620
312,574 Los Angeles 2,013 2,013 -0.597100578780812791472351765489673550966 0.0067918497802636835797043547742708749500600
314,013 Atlanta 2,014 2,014 -0.590986469887590262785677063946996203379 0.0071913703555733120255693168198162205353580

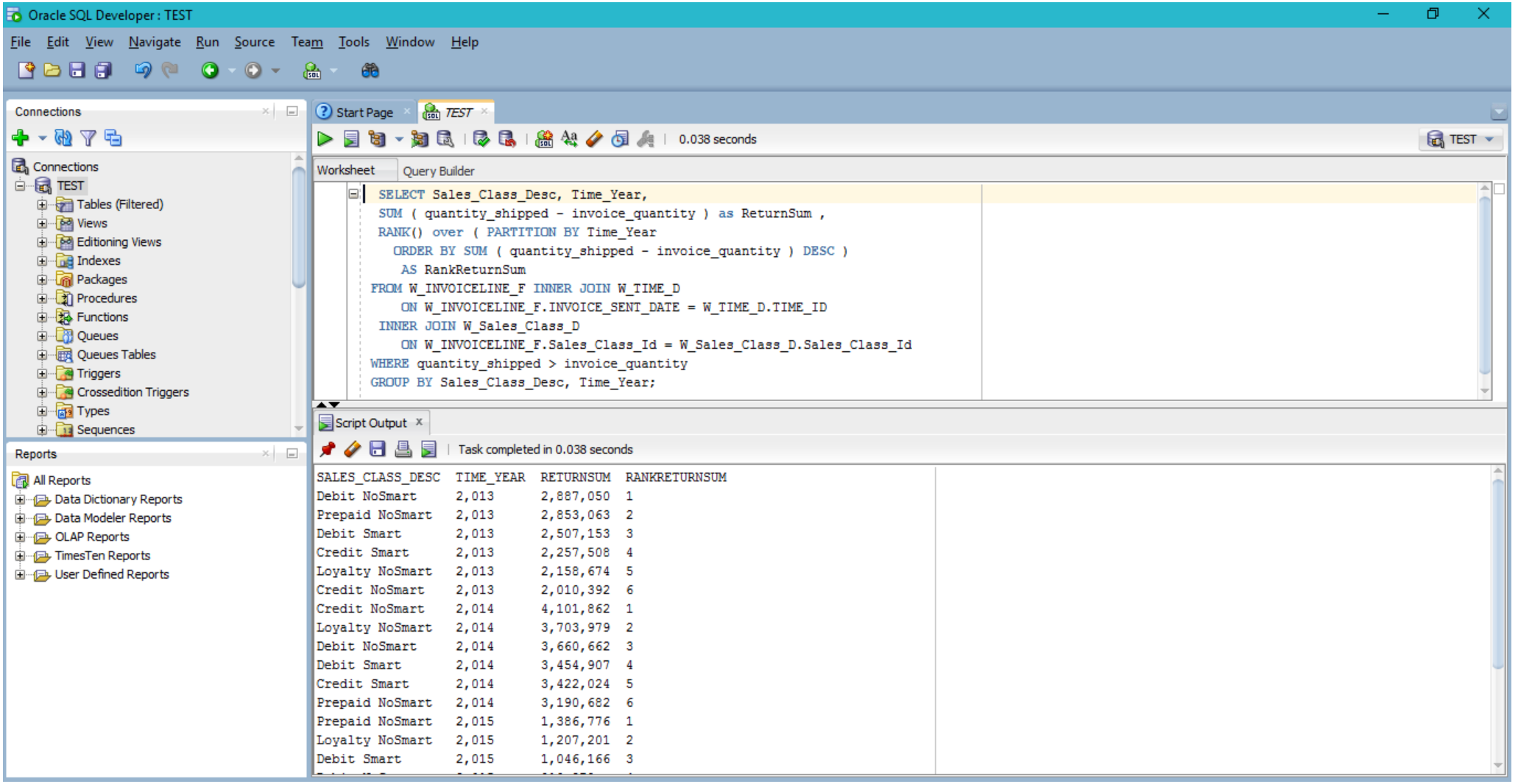
AQ6: Top performers of percent rank of job profit margins for locations

```
SELECT Job_Id, Location_Name, Time_Year, Time_Month,
       ProfitMargin, PercentRankProfitMargin
 FROM (
  SELECT X1.Job_Id, X1.Location_Name, X1.Time_Year, X1.Time_Month,
         (SumInvoiceAmt - TotalCosts) / SumInvoiceAmt AS ProfitMargin,
         PERCENT_RANK() OVER (
         ORDER BY ( (SumInvoiceAmt - TotalCosts) / SumInvoiceAmt ) )
          AS PercentRankProfitMargin
  FROM LocCostSummary X1, LocRevenueSummary X2
  WHERE X1.Job_Id = X2.Job_Id )
 WHERE PercentRankProfitMargin > 0.95;
```


Oracle SQL Developer : TEST
File Edit View Navigate Run Source Team Tools Window Help
Connections
Connections
TEST
Tables (Filtered)
Views
Editioning Views
Indexes
Packages
Procedures
Functions
Queues
Queues Tables
Triggers
Crossedition Triggers
Types
Sequences
Reports
All Reports
Data Dictionary Reports
Data Modeler Reports
OLAP Reports
TimesTen Reports
User Defined Reports
Start Page
TEST
0.185 seconds
TEST
Worksheet
Query Builder
SELECT Job_Id, Location_Name, Time_Year, Time_Month,
ProfitMargin, PercentRankProfitMargin
FROM (
SELECT X1.Job_Id, X1.Location_Name, X1.Time_Year, X1.Time_Month,
(SumInvoiceAmt - TotalCosts) / SumInvoiceAmt AS ProfitMargin,
PERCENT_RANK() OVER (
ORDER BY ((SumInvoiceAmt - TotalCosts) / SumInvoiceAmt))
AS PercentRankProfitMargin
FROM LocCostSummary X1, LocRevenueSummary X2
WHERE X1.Job_Id = X2.Job_Id)
WHERE PercentRankProfitMargin > 0.95;
Script Output
Task completed in 0.185 seconds
JOB_ID LOCATION_NAME TIME_YEAR TIME_MONTH PROFITMARGIN PERCENTRANKPROFITMARGIN
312,095 Denver 2,013 7 0.600210605475661937393430492552198951195 2 0.950059928086296444266879744306831801837 8
312,058 Vancouver 2,014 4 0.600211423468199114700629028500427118117 6 0.950459448661606072712744706352377147423 1
312,889 Montreal 2,014 6 0.600212388243757065049840715239954783681 0.950858969236915701158609668397922493008 4
313,463 Vancouver 2,014 1 0.600213831613499594408496670376728480069 4 0.951258489812225329604474630443467838593 7
313,487 Toronto 2,013 12 0.600215996326905417814508723599632690541 8 0.951658010387534958050339592489013184179
313,707 Toronto 2,013 11 0.600218258592389329524239120901151499204 4 0.952057530962844586496204554534558529764 3
313,738 New York 2,015 1 0.600218820527916789323913256795211461093 1 0.952457051538154214942069516580103875349 6
313,849 Chicago 2,014 6 0.600219118288956787429070274989087734613 7 0.952856572113463843387934478625649220934 9
312,899 Vancouver 2,013 6 0.600219175863116330869697276500961830515 3 0.953256092688773471833799440671194566520 2
312,878 Atlanta 2,015 2 0.600220494186046511627906976744186046511 6 0.953655613264083100279664402716739912105 5
312,655 Chicago 2,014 7 0.600221120219739766951252318452760197652 5 0.954055133839392728725529364762285257690 8
312,002 Seattle 2,013 11 0.600221213054187192118226600985221674876 8 0.954454654414702357171394326807830603276 1
312,397 Denver 2,013 8 0.600222457550019287175120271912778299380 6 0.954854174990011985617259288853375948861 4
313,835 Seattle 2,013 5 0.600224242424242424242424242424242424 0.955253695565321614063124250898921294446 7
314,151 Atlanta 2,013 8 0.600231363082291641769640765792681469442 4 0.955653216140631242508989212944466640032
314,001 Toronto 2,013 6 0.600233268155466364790230180782820486432 7 0.956052736715940870954854174990011985617 3

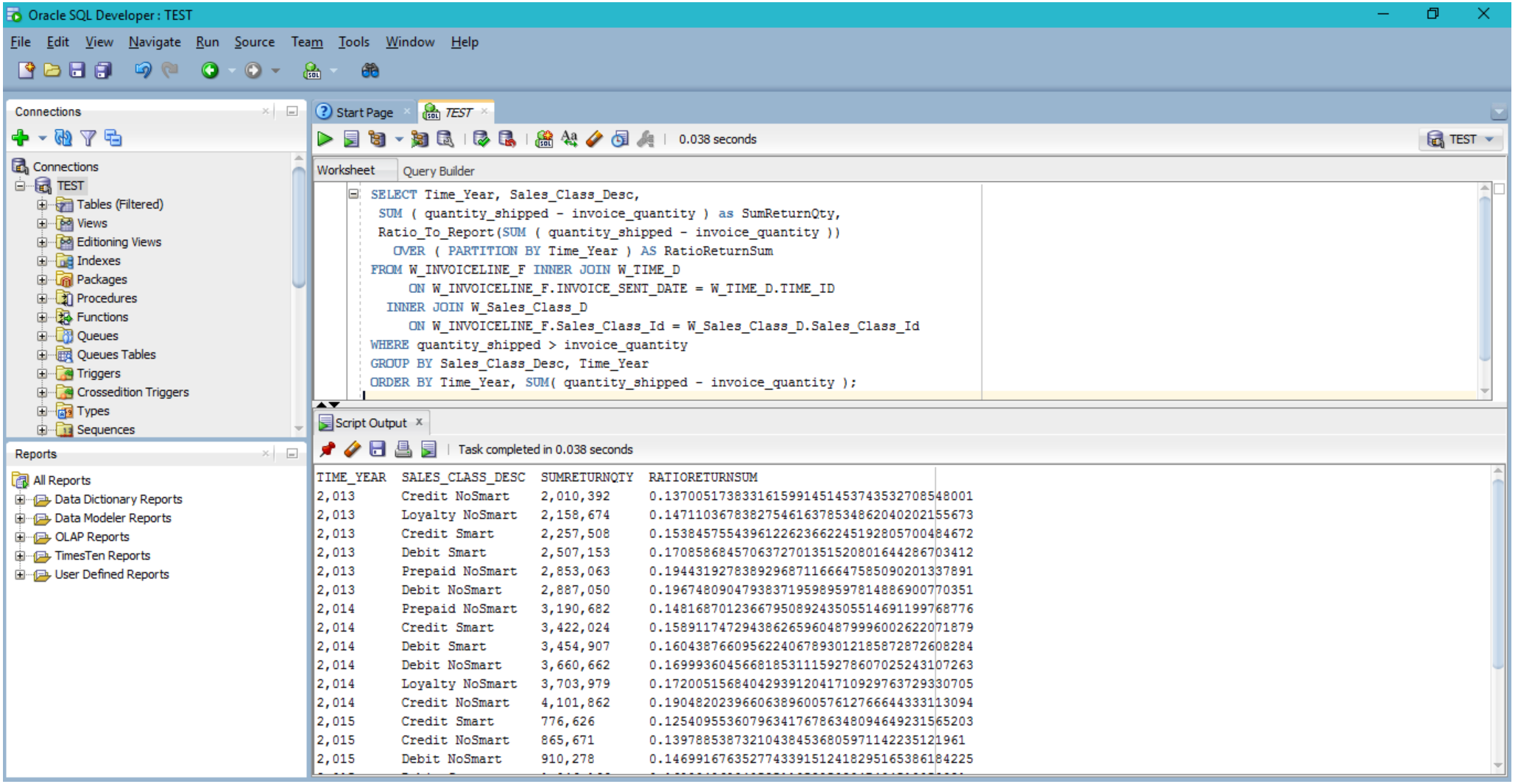
AQ7: Rank sales class by return quantities for each year

```
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;
```


Oracle SQL Developer : TEST
File Edit View Navigate Run Source Team Tools Window Help
Connections
Connections
TEST
Tables (Filtered)
Views
Editioning Views
Indexes
Packages
Procedures
Functions
Queues
Queues Tables
Triggers
Crossedition Triggers
Types
Sequences
Reports
All Reports
Data Dictionary Reports
Data Modeler Reports
OLAP Reports
TimesTen Reports
User Defined Reports
Start Page
TEST
0.038 seconds
TEST
Worksheet
Query Builder
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;
Script Output
Task completed in 0.038 seconds
SALES_CLASS_DESC TIME_YEAR RETURNSUM RANKRETURNSUM
Debit NoSmart 2,013 2,887,050 1
Prepaid NoSmart 2,013 2,853,063 2
Debit Smart 2,013 2,507,153 3
Credit Smart 2,013 2,257,508 4
Loyalty NoSmart 2,013 2,158,674 5
Credit NoSmart 2,013 2,010,392 6
Credit NoSmart 2,014 4,101,862 1
Loyalty NoSmart 2,014 3,703,979 2
Debit NoSmart 2,014 3,660,662 3
Debit Smart 2,014 3,454,907 4
Credit Smart 2,014 3,422,024 5
Prepaid NoSmart 2,014 3,190,682 6
Prepaid NoSmart 2,015 1,386,776 1
Loyalty NoSmart 2,015 1,207,201 2
Debit Smart 2,015 1,046,166 3

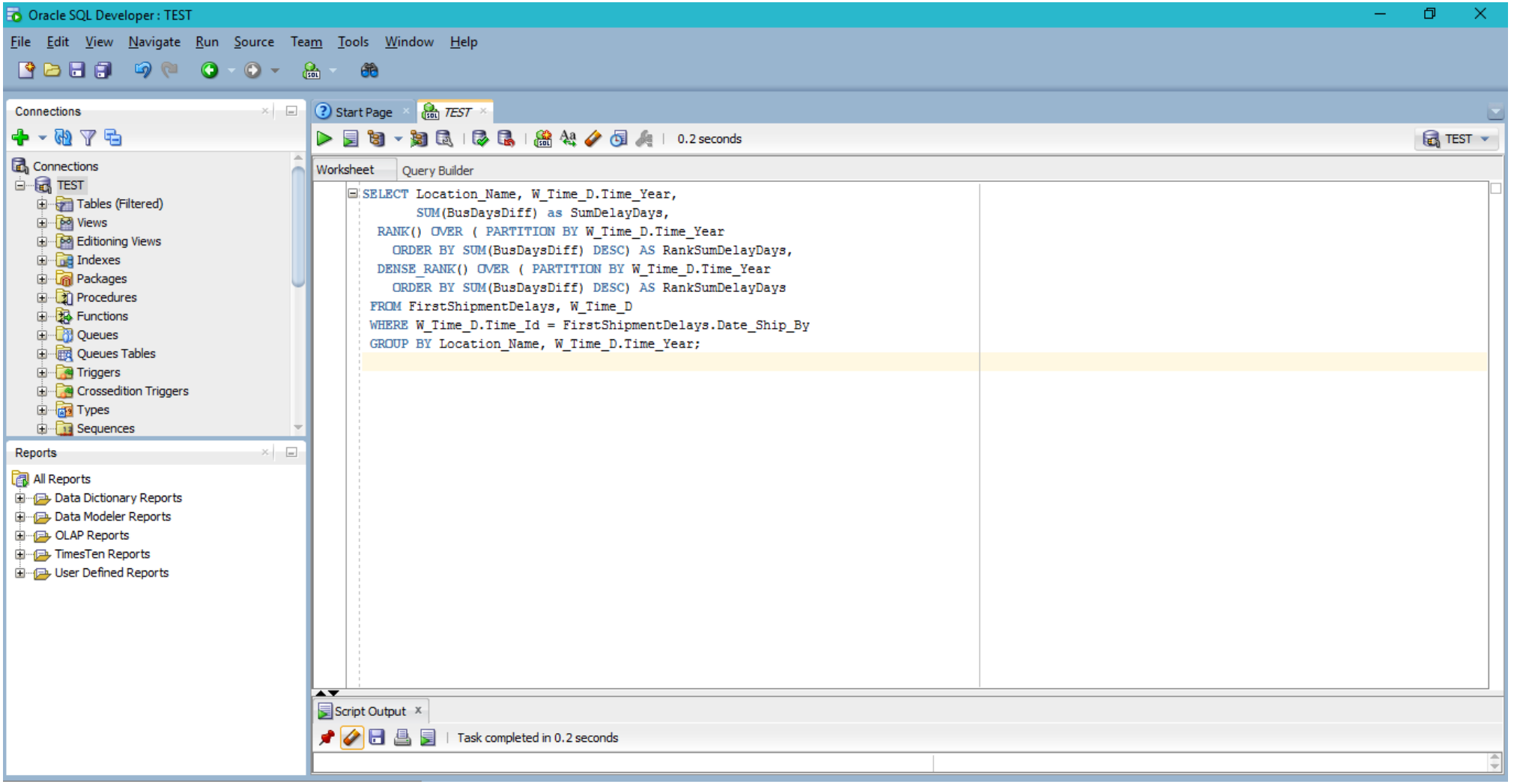
AQ8: Ratio to report of return quantities for sales classes by year

```
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 );
```


Oracle SQL Developer : TEST
File Edit View Navigate Run Source Team Tools Window Help
Connections
Connections
TEST
Tables (Filtered)
Views
Editioning Views
Indexes
Packages
Procedures
Functions
Queues
Queues Tables
Triggers
Crossedition Triggers
Types
Sequences
Reports
All Reports
Data Dictionary Reports
Data Modeler Reports
OLAP Reports
TimesTen Reports
User Defined Reports
Start Page
TEST
0.038 seconds
TEST
Worksheet
Query Builder
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);
Script Output
Task completed in 0.038 seconds
TIME_YEAR SALES_CLASS_DESC SUMRETURNQTY RATIORETURNSUM
2,013 Credit NoSmart 2,010,392 0.13700517383316159914514537435327085480 01
2,013 Loyalty NoSmart 2,158,674 0.14711036783827546163785348620402021556 73
2,013 Credit Smart 2,257,508 0.15384575543961226236622451928057004846 72
2,013 Debit Smart 2,507,153 0.17085868457063727013515208016442867034 12
2,013 Prepaid NoSmart 2,853,063 0.19443192783892968711666475850902013378 91
2,013 Debit NoSmart 2,887,050 0.19674809047938371959895978148869007703 51
2,014 Prepaid NoSmart 3,190,682 0.14816870123667950892435055146911997687 76
2,014 Credit Smart 3,422,024 0.15891174729438626596048799960026220718 79
2,014 Debit Smart 3,454,907 0.16043876609562240678930121858728726082 84
2,014 Debit NoSmart 3,660,662 0.16999360456681853111592786070252431072 63
2,014 Loyalty NoSmart 3,703,979 0.17200515684042939120417109297637293307 05
2,014 Credit NoSmart 4,101,862 0.19048202396606389600576127666443331130 94
2,015 Credit Smart 776,626 0.12540955360796341767863480946492315652 03
2,015 Credit NoSmart 865,671 0.13978853873210438453680597114223512196 1
2,015 Debit NoSmart 910,278 0.14699167635277433915124182951653861842 25

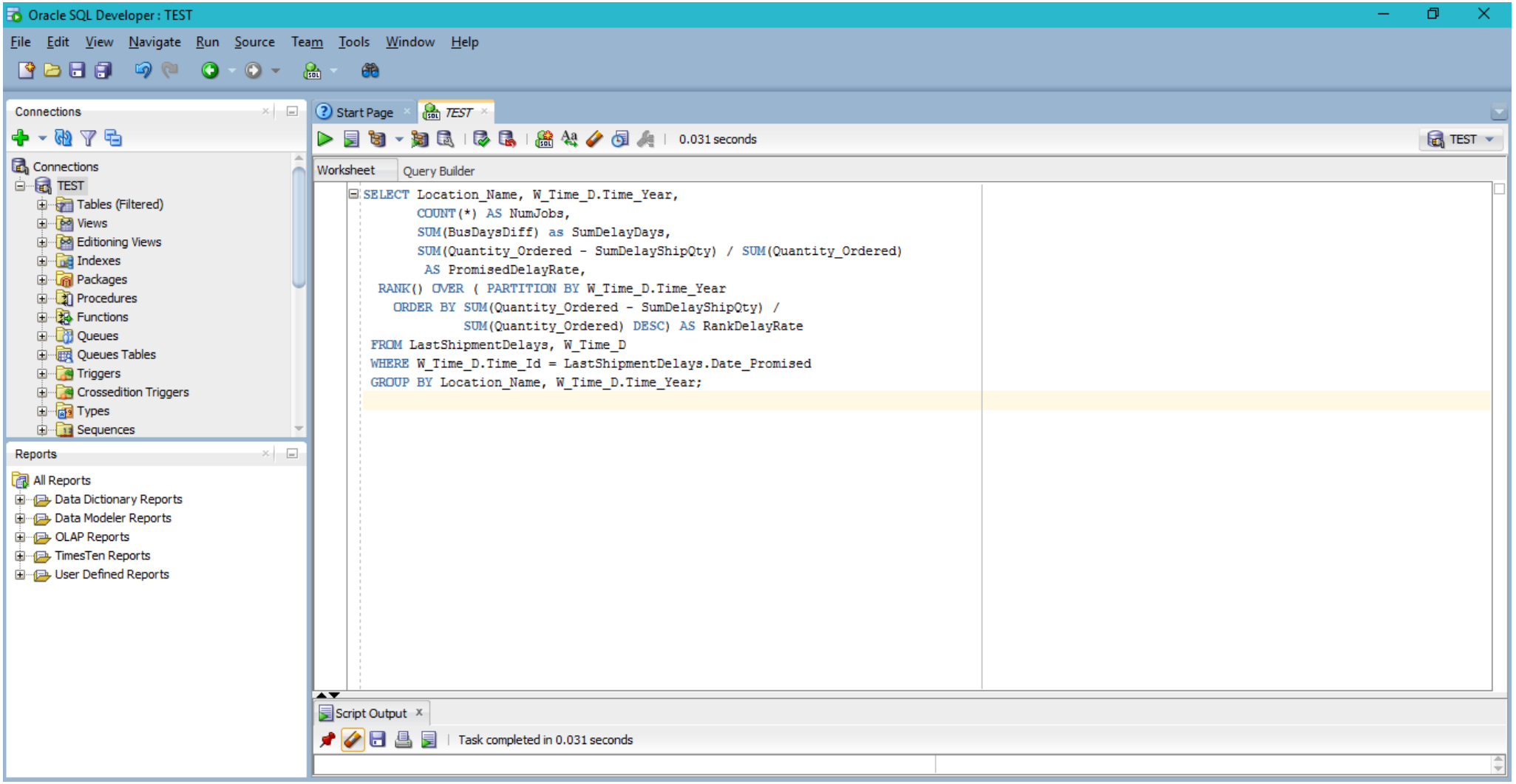
AQ9: Rank locations by sum of business days delayed for the job shipped by date (first shipment)

```
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 FirstShipmentDelays, W_Time_D
 WHERE W_Time_D.Time_Id = FirstShipmentDelays.Date_Ship_By
 GROUP BY Location_Name, W_Time_D.Time_Year;
```


Oracle SQL Developer : TEST
File Edit View Navigate Run Source Team Tools Window Help
Connections
Connections
TEST
Tables (Filtered)
Views
Editioning Views
Indexes
Packages
Procedures
Functions
Queues
Queues Tables
Triggers
Crossedition Triggers
Types
Sequences
Reports
All Reports
Data Dictionary Reports
Data Modeler Reports
OLAP Reports
TimesTen Reports
User Defined Reports
Start Page
TEST
0.2 seconds
TEST
Worksheet
Query Builder
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 FirstShipmentDelays, W_Time_D
WHERE W_Time_D.Time_Id = FirstShipmentDelays.Date_Ship_By
GROUP BY Location_Name, W_Time_D.Time_Year;
Script Output
Task completed in 0.2 seconds

AQ10: Rank locations by delay rate for jobs delayed on the last shipment date

```
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 LastShipmentDelays, W_Time_D
 WHERE W_Time_D.Time_Id = LastShipmentDelays.Date_Promised
 GROUP BY Location_Name, W_Time_D.Time_Year;
```


Oracle SQL Developer : TEST
File Edit View Navigate Run Source Team Tools Window Help
Connections
Connections
TEST
Tables (Filtered)
Views
Editioning Views
Indexes
Packages
Procedures
Functions
Queues
Queues Tables
Triggers
Crossedition Triggers
Types
Sequences
Reports
All Reports
Data Dictionary Reports
Data Modeler Reports
OLAP Reports
TimesTen Reports
User Defined Reports
Start Page
TEST
0.031 seconds
TEST
Worksheet
Query Builder
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 LastShipmentDelays, W_Time_D
WHERE W_Time_D.Time_Id = LastShipmentDelays.Date_Promised
GROUP BY Location_Name, W_Time_D.Time_Year;
Script Output
Task completed in 0.031 seconds