Assignment 5B Analytical Processing

Total points: 25

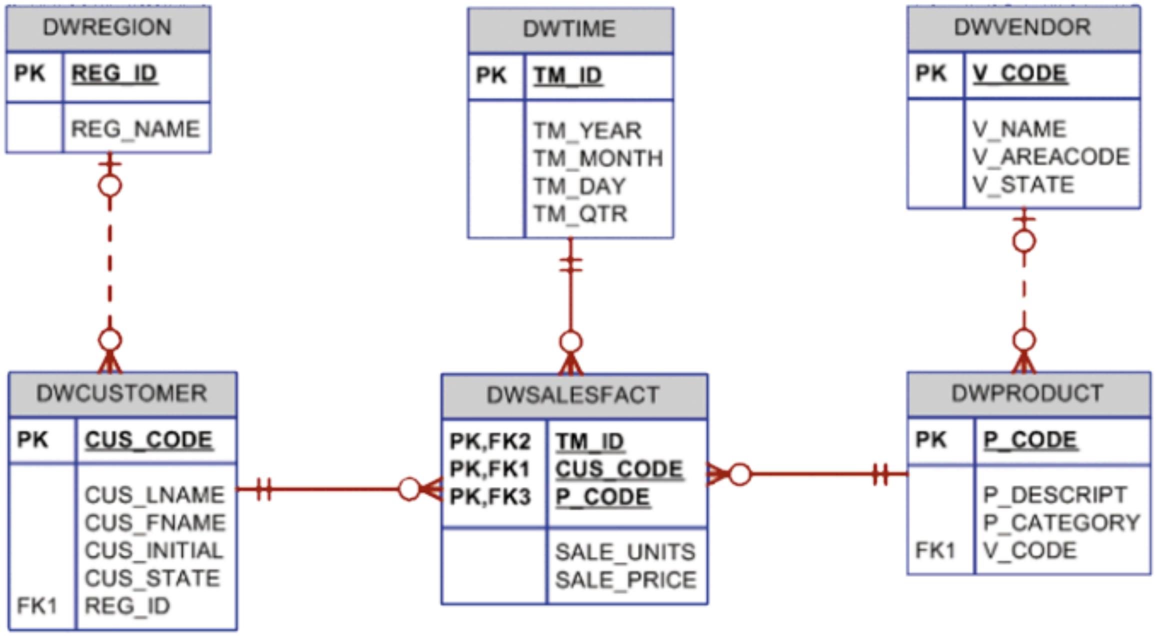
This assignment should be completed individually. For each problem, submit your SQL statement and a screen shot of the SQL results in a single Word document or pdf file. Submit the file via eLearning.

I recommend creating a new user and workspace, log in as that user and load the database script **DWDBINIT.sql** (provided in this week's assignment folder).

Before you attempt to write any SQL queries, familiarize yourself with the database structure and data. I have provided a relational diagram and sample data for this database.

Write queries to address each of the problems below. **Submit both the SQL statements and the screen prints of the outputs from Oracle.**

SaleCo Snowflake schema



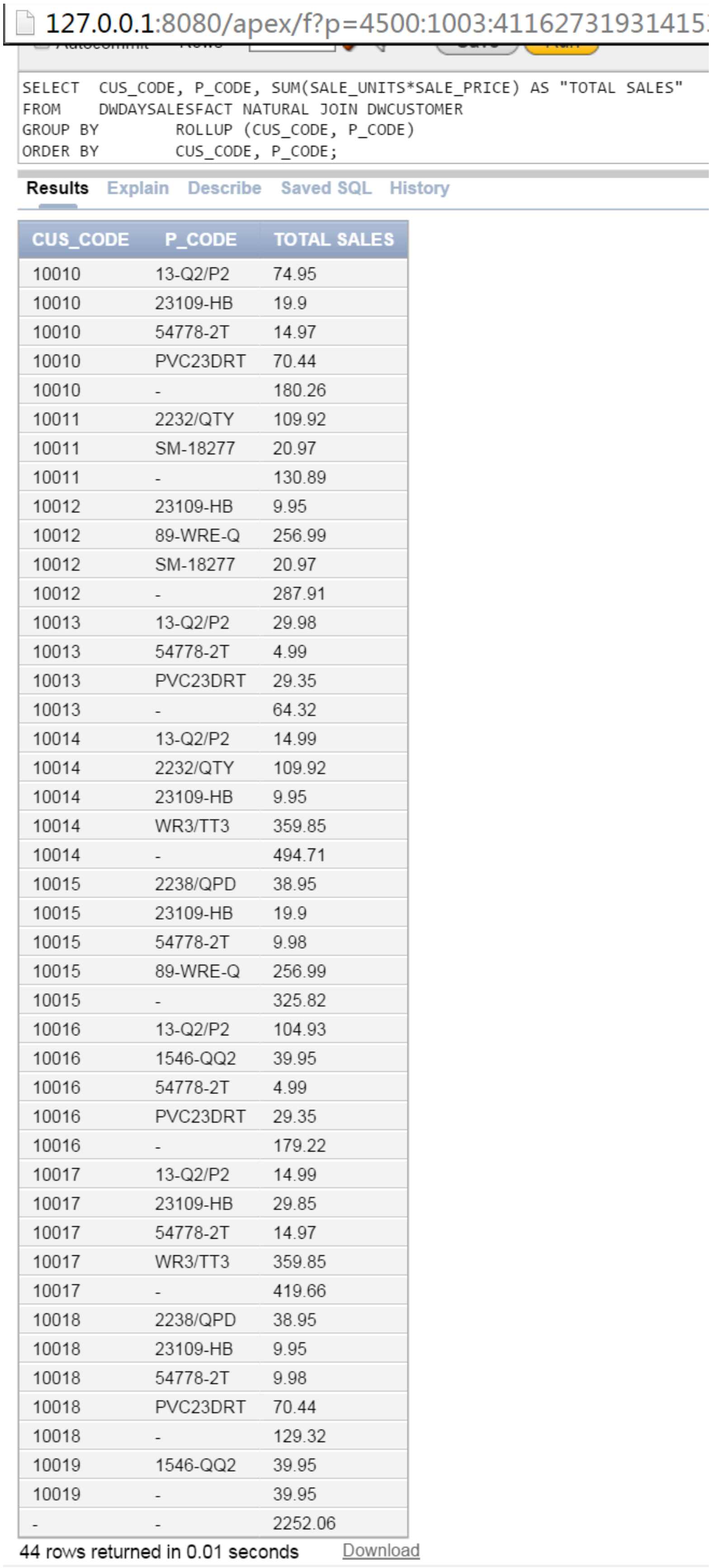
(*Hint*: You will be using the ROLLUP and CUBE commands.)

Because there is no table named DWSALESFACT as discribed in the begining of this assignment. Only a table named DWDAYSALESFACT.

So, I do this assignment go with DWDAYSALESFACT instead of DWSALESFACT.

1. List the total sales by customer and by product, with subtotals by customer and a grand total for all product sales.

   ```
   SELECT      CUS_CODE, P_CODE, SUM(SALE_UNITS*SALE_PRICE) AS "TOTAL SALES"
   FROM        DWDAYSALESFACT NATURAL JOIN DWCUSTOMER
   GROUP BY    ROLLUP (CUS_CODE, P_CODE)
   ORDER BY    CUS_CODE, P_CODE;
   ```



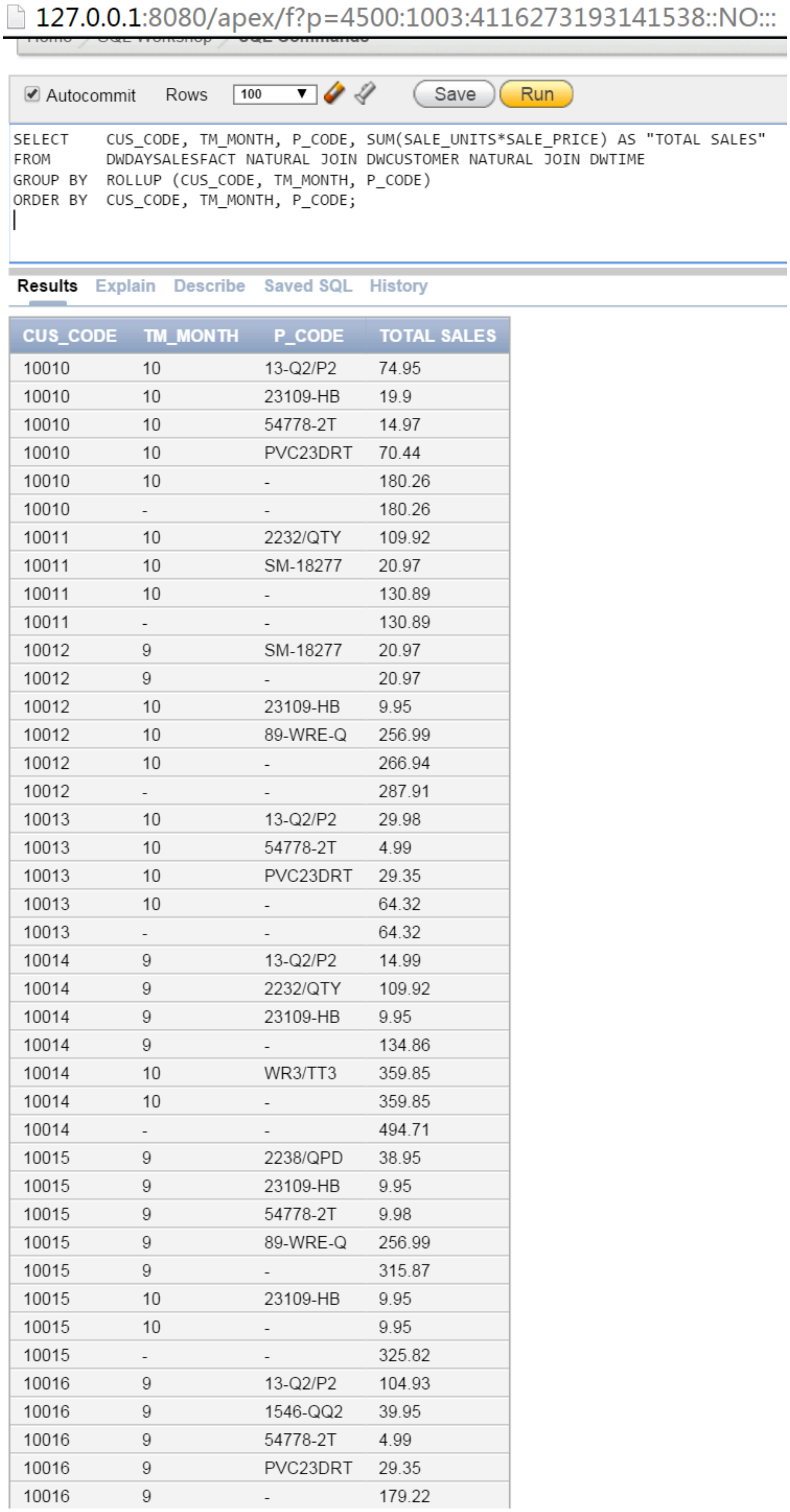
```
SELECT  CUS_CODE, P_CODE, SUM(SALE_UNITS*SALE_PRICE) AS "TOTAL SALES"
FROM    DWDAYSALESFACT NATURAL JOIN DWCUSTOMER
GROUP BY        ROLLUP (CUS_CODE, P_CODE)
ORDER BY        CUS_CODE, P_CODE;
```

CUS_CODE	P_CODE	TOTAL SALES
10010	13-Q2/P2	74.95
10010	23109-HB	19.9
10010	54778-2T	14.97
10010	PVC23DRT	70.44
10010	-	180.26
10011	2232/QTY	109.92
10011	SM-18277	20.97
10011	-	130.89
10012	23109-HB	9.95
10012	89-WRE-Q	256.99
10012	SM-18277	20.97
10012	-	287.91
10013	13-Q2/P2	29.98
10013	54778-2T	4.99
10013	PVC23DRT	29.35
10013	-	64.32
10014	13-Q2/P2	14.99
10014	2232/QTY	109.92
10014	23109-HB	9.95
10014	WR3/TT3	359.85
10014	-	494.71
10015	2238/QPD	38.95
10015	23109-HB	19.9
10015	54778-2T	9.98
10015	89-WRE-Q	256.99
10015	-	325.82
10016	13-Q2/P2	104.93
10016	1546-QQ2	39.95
10016	54778-2T	4.99
10016	PVC23DRT	29.35
10016	-	179.22
10017	13-Q2/P2	14.99
10017	23109-HB	29.85
10017	54778-2T	14.97
10017	WR3/TT3	359.85
10017	-	419.66
10018	2238/QPD	38.95
10018	23109-HB	9.95
10018	54778-2T	9.98
10018	PVC23DRT	70.44
10018	-	129.32
10019	1546-QQ2	39.95
10019	-	39.95
-	-	2252.06

44 rows returned in 0.01 seconds

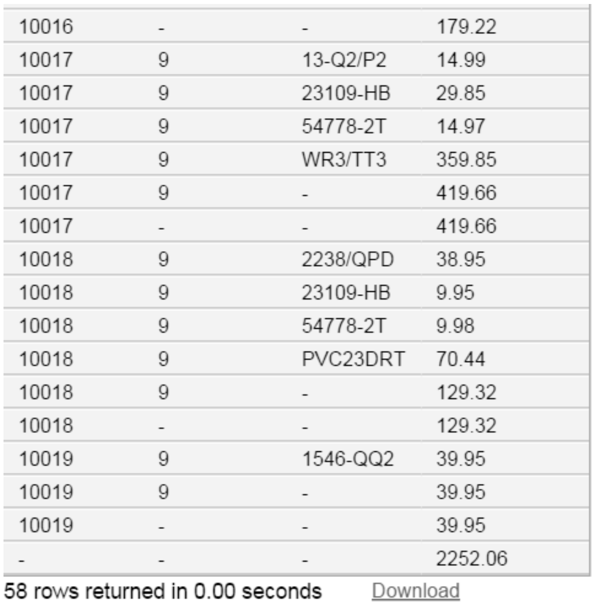
2. List the total sales by customer, month and product, with subtotals by customer and by month and a grand total for all product sales.

```
SELECT     CUS_CODE, TM_MONTH, P_CODE, SUM(SALE_UNITS*SALE_PRICE)
           AS "TOTAL SALES"
FROM       DWDAYSALESFACT NATURAL JOIN DWCUSTOMER NATURAL JOIN DWTIME
GROUP BY   ROLLUP (CUS_CODE, TM_MONTH, P_CODE)
ORDER BY   CUS_CODE, TM_MONTH, P_CODE;
```



```
SELECT    CUS_CODE, TM_MONTH, P_CODE, SUM(SALE_UNITS*SALE_PRICE) AS "TOTAL SALES"
FROM      DWDAYSALESFACT NATURAL JOIN DWCUSTOMER NATURAL JOIN DWTIME
GROUP BY  ROLLUP (CUS_CODE, TM_MONTH, P_CODE)
ORDER BY  CUS_CODE, TM_MONTH, P_CODE;
```

CUS_CODE	TM_MONTH	P_CODE	TOTAL SALES
10010	10	13-Q2/P2	74.95
10010	10	23109-HB	19.9
10010	10	54778-2T	14.97
10010	10	PVC23DRT	70.44
10010	10	-	180.26
10010	-	-	180.26
10011	10	2232/QTY	109.92
10011	10	SM-18277	20.97
10011	10	-	130.89
10011	-	-	130.89
10012	9	SM-18277	20.97
10012	9	-	20.97
10012	10	23109-HB	9.95
10012	10	89-WRE-Q	256.99
10012	10	-	266.94
10012	-	-	287.91
10013	10	13-Q2/P2	29.98
10013	10	54778-2T	4.99
10013	10	PVC23DRT	29.35
10013	10	-	64.32
10013	-	-	64.32
10014	9	13-Q2/P2	14.99
10014	9	2232/QTY	109.92
10014	9	23109-HB	9.95
10014	9	-	134.86
10014	10	WR3/TT3	359.85
10014	10	-	359.85
10014	-	-	494.71
10015	9	2238/QPD	38.95
10015	9	23109-HB	9.95
10015	9	54778-2T	9.98
10015	9	89-WRE-Q	256.99
10015	9	-	315.87
10015	10	23109-HB	9.95
10015	10	-	9.95
10015	-	-	325.82
10016	9	13-Q2/P2	104.93
10016	9	1546-QQ2	39.95
10016	9	54778-2T	4.99
10016	9	PVC23DRT	29.35
10016	9	-	179.22

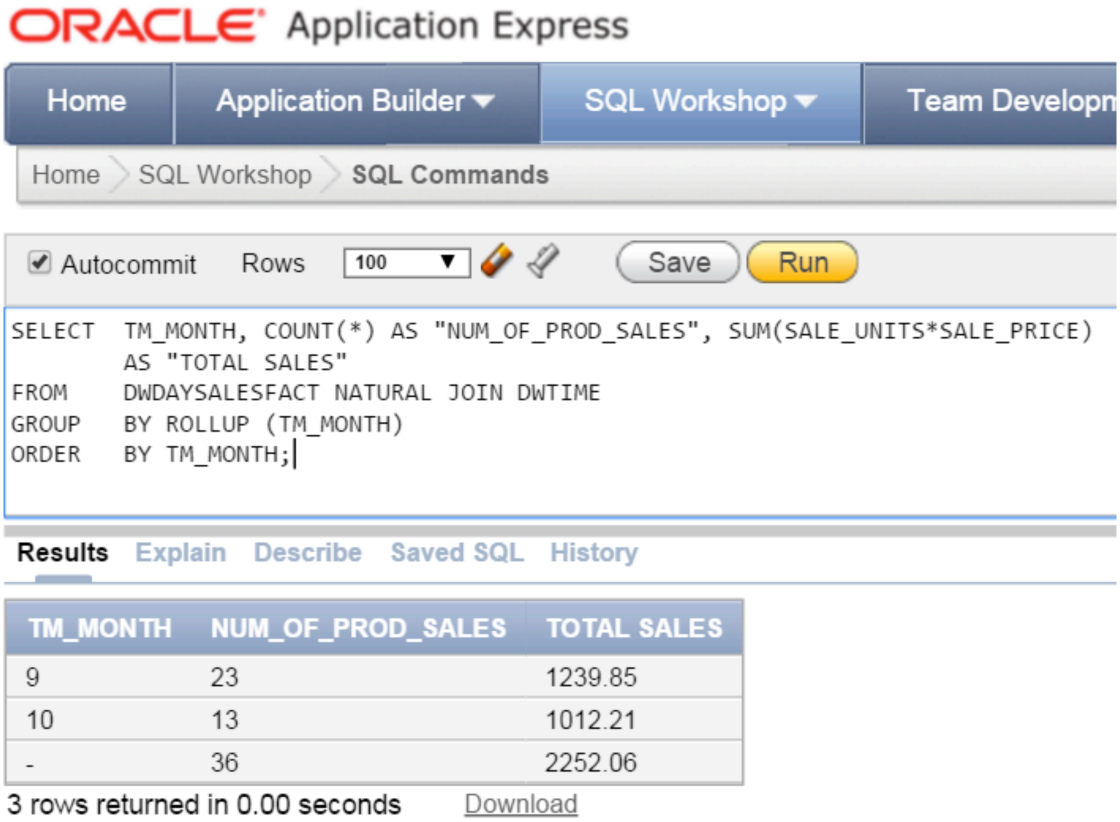


10016	-	-	179.22
10017	9	13-Q2/P2	14.99
10017	9	23109-HB	29.85
10017	9	54778-2T	14.97
10017	9	WR3/TT3	359.85
10017	9	-	419.66
10017	-	-	419.66
10018	9	2238/QPD	38.95
10018	9	23109-HB	9.95
10018	9	54778-2T	9.98
10018	9	PVC23DRT	70.44
10018	9	-	129.32
10018	-	-	129.32
10019	9	1546-QQ2	39.95
10019	9	-	39.95
10019	-	-	39.95
-	-	-	2252.06

58 rows returned in 0.00 seconds Download

3. List the number of product sales (number of rows) and total sales by month, with subtotals by month and a grand total for all sales.

```
SELECT    TM_MONTH, COUNT(*) AS "NUM_OF_PROD_SALES",
          SUM(SALE_UNITS*SALE_PRICE)  AS "TOTAL SALES"
FROM      DWDAYSALESFACT NATURAL JOIN DWTIME
GROUP  BY  ROLLUP (TM_MONTH)
ORDER  BY  TM_MONTH;
```



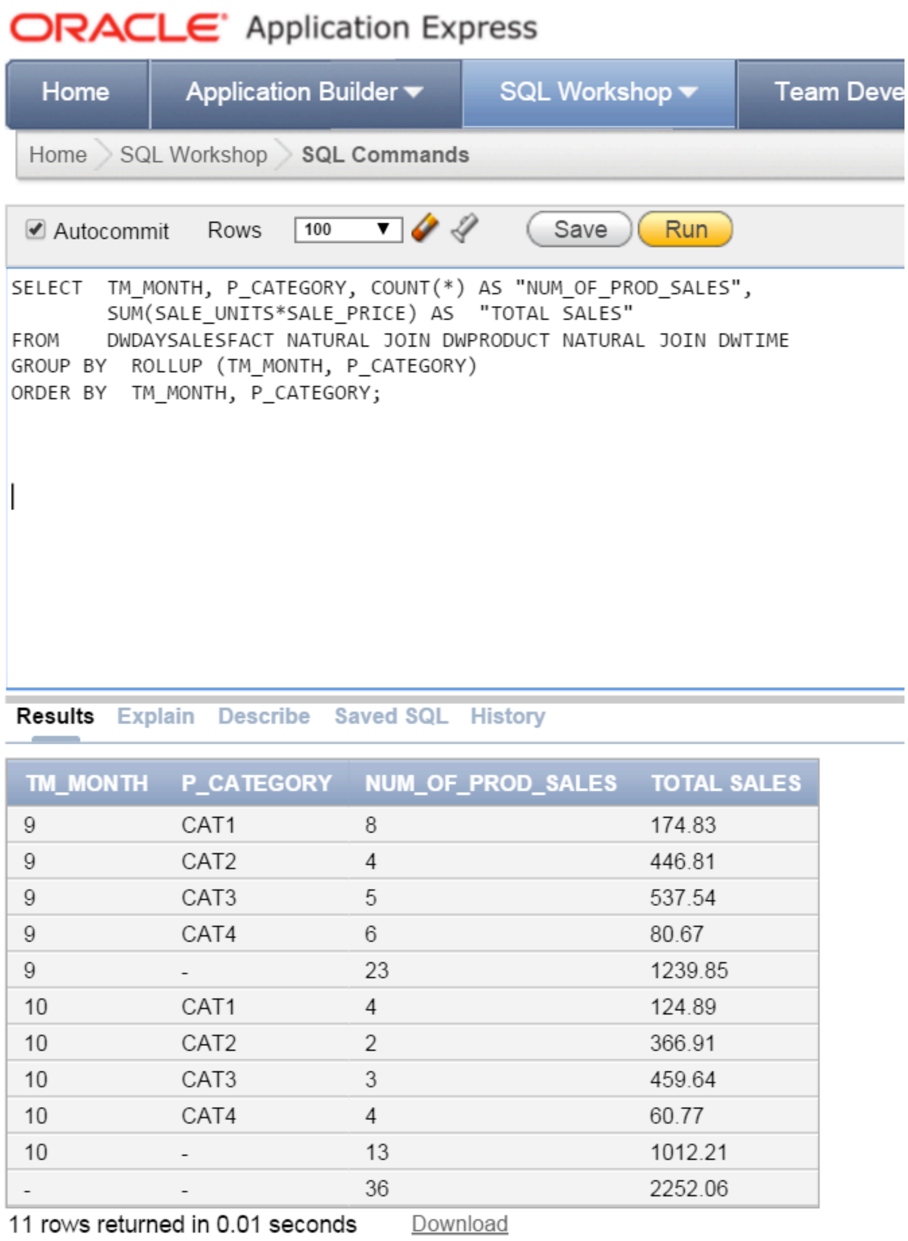
```
SELECT  TM_MONTH, COUNT(*) AS "NUM_OF_PROD_SALES", SUM(SALE_UNITS*SALE_PRICE)
        AS "TOTAL SALES"
FROM    DWDAYSALESFACT NATURAL JOIN DWTIME
GROUP   BY ROLLUP (TM_MONTH)
ORDER   BY TM_MONTH;
```

TM_MONTH	NUM_OF_PROD_SALES	TOTAL SALES
9	23	1239.85
10	13	1012.21
-	36	2252.06

3 rows returned in 0.00 seconds

4. List the number of product sales (number of rows) and total sales by month and product category with subtotals by month and product category and a grand total for all sales.

```
SELECT    TM_MONTH, P_CATEGORY, COUNT(*) AS "NUM_OF_PROD_SALES",
    SUM(SALE_UNITS*SALE_PRICE) AS  "TOTAL SALES"
FROM      DWDAYSALESFACT NATURAL JOIN DWPRODUCT NATURAL JOIN DWTIME
GROUP BY  ROLLUP (TM_MONTH, P_CATEGORY)
ORDER BY  TM_MONTH, P_CATEGORY;
```



☑ Autocommit Rows 100

```
SELECT  TM_MONTH, P_CATEGORY, COUNT(*) AS "NUM_OF_PROD_SALES",
        SUM(SALE_UNITS*SALE_PRICE) AS  "TOTAL SALES"
FROM    DWDAYSALESFACT NATURAL JOIN DWPRODUCT NATURAL JOIN DWTIME
GROUP BY  ROLLUP (TM_MONTH, P_CATEGORY)
ORDER BY  TM_MONTH, P_CATEGORY;
```

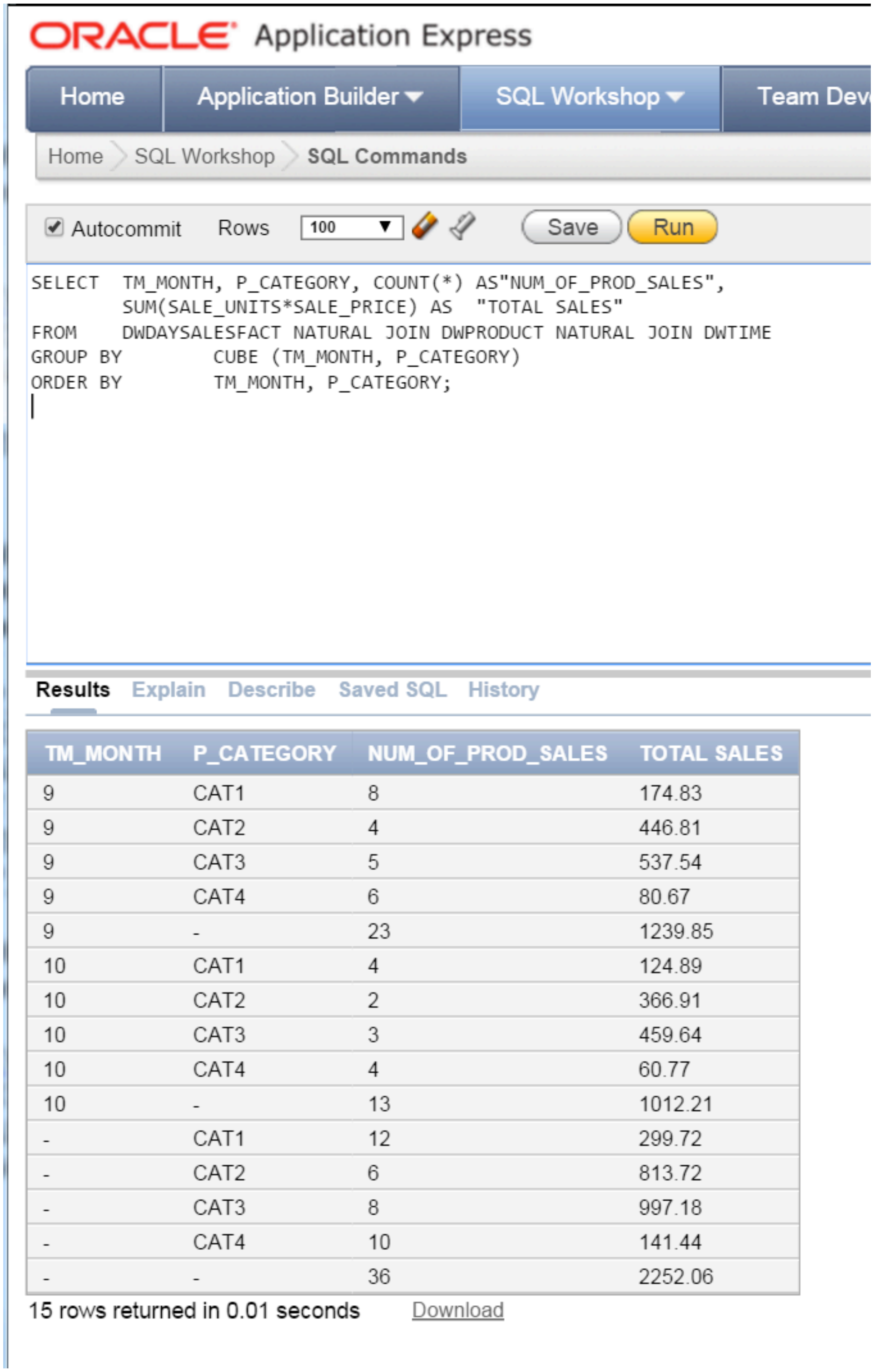
TM_MONTH	P_CATEGORY	NUM_OF_PROD_SALES	TOTAL SALES
9	CAT1	8	174.83
9	CAT2	4	446.81
9	CAT3	5	537.54
9	CAT4	6	80.67
9	-	23	1239.85
10	CAT1	4	124.89
10	CAT2	2	366.91
10	CAT3	3	459.64
10	CAT4	4	60.77
10	-	13	1012.21
-	-	36	2252.06

11 rows returned in 0.01 seconds

5. Using the answer to problem 4 as your base, what command would you need to generate the same output but with subtotals in all columns?

Use the CUBE command

```
SELECT      TM_MONTH, P_CATEGORY, COUNT(*) AS"NUM_OF_PROD_SALES",
            SUM(SALE_UNITS*SALE_PRICE) AS  "TOTAL SALES"
FROM        DWDAYSALESFACT  NATURAL  JOIN  DWPRODUCT  NATURAL  JOIN
            DWTIME
GROUP BY    CUBE (TM_MONTH, P_CATEGORY)
ORDER BY    TM_MONTH, P_CATEGORY;
```



☑ Autocommit   Rows 100

```
SELECT  TM_MONTH, P_CATEGORY, COUNT(*) AS"NUM_OF_PROD_SALES",
        SUM(SALE_UNITS*SALE_PRICE) AS  "TOTAL SALES"
FROM    DWDAYSALESFACT NATURAL JOIN DWPRODUCT NATURAL JOIN DWTIME
GROUP BY        CUBE (TM_MONTH, P_CATEGORY)
ORDER BY        TM_MONTH, P_CATEGORY;
```

TM_MONTH	P_CATEGORY	NUM_OF_PROD_SALES	TOTAL SALES
9	CAT1	8	174.83
9	CAT2	4	446.81
9	CAT3	5	537.54
9	CAT4	6	80.67
9	-	23	1239.85
10	CAT1	4	124.89
10	CAT2	2	366.91
10	CAT3	3	459.64
10	CAT4	4	60.77
10	-	13	1012.21
-	CAT1	12	299.72
-	CAT2	6	813.72
-	CAT3	8	997.18
-	CAT4	10	141.44
-	-	36	2252.06

15 rows returned in 0.01 seconds   Download