Eduardo Martinez

CS443 -- Assignment 3

Write the queries necessary to obtain the required information. Make sure all columns you return have descriptive column headings.

1) Return the Minimum and Maximum sales for all offices.

```
SELECT MIN(sales) AS "MinSales", MAX(sales) AS "MaxSales"
FROM offices;
```

```
  MinSales   MaxSales
---------- ----------
    186042     735042
```

2) Determine how many orders were made in 1989. Return the number of rows that meet this condition.

```
SELECT COUNT(ORDER_NUM) AS "NumOfOrders"
FROM ORDERS
WHERE ORDER_DATE >=('01-JAN-89') AND ORDER_DATE <=('31-DEC-89');
```

```
NumOfOrders
-----------
          9
```

3) List the number of different titles in the sales reps table. Only list each title once and unknown titles should be ignored.

```
SELECT DISTINCT title
FROM salesreps;
```

```
TITLE
----------
Sales Rep
Sales Mgr
VP Sales
```

4) List the average quota for salesreps in office 21.

```
SELECT AVG(quota)
```

```
FROM salesreps
WHERE rep_office = 21;
```

```
AVG(QUOTA)
----------
    350000
```

5) List the average sale amount for all sale reps in each office.

```
SELECT AVG(sales), rep_office
FROM salesreps
GROUP BY rep_office;
```

```
AVG(SALES) REP_OFFICE
---------- ----------
     75985
    367911         13
  346318.5         11
  417957.5         21
    245014         12
```

6) For each salesrep that has made an order, list the minimum, maximum and average order amount for all their orders. Include only those orders made anytime from 1990-1999. Omit from the list any salesrep that has only made 1 order in this time frame. Sort the results by Empl_Num.

```
SELECT REP, MIN(AMOUNT) AS "MIN", MAX(AMOUNT) AS "MAX",
AVG(AMOUNT) AS "AVG"
FROM ORDERS
WHERE ORDER_DATE >= ('01-JAN-1990')AND ORDER_DATE <=('31-
DEC-1999')
GROUP BY REP
HAVING COUNT(REP)>1
ORDER BY REP;
```

```
       REP        MIN        MAX        AVG
---------- ---------- ---------- ----------
       102       2130       3750       2940
       105       3745       4104     3924.5
       107        652      31350 11477.3333
       108        652      45000     9645.5
       109       1480       5625     3552.5
       110        632      22500      11566

5 rows selected.
```

7) Use a sub-query to list the Customer number; Name and Credit Limit of any customers who have exceeded their credit limit (amount > credit limit) on any order.

```
SELECT CUST_NUM, COMPANY, CREDIT_LIMIT
FROM CUSTOMERS
WHERE CREDIT_LIMIT < ANY(SELECT AMOUNT
FROM ORDERS
WHERE CUST_NUM=CUST);
```

CUST_NUM	COMPANY	CREDIT_LIMIT
2109	Chen Associates	25000

8) Use a subquery and using the "all" keyword to find the customer number, Salesrep id, and CreditLimit of every customer whose CreditLimit is larger than the CreditLimit of all of the customers of sales rep number 109.

```
SELECT CUST_NUM, CUST_REP, CREDIT_LIMIT
FROM CUSTOMERS
WHERE CREDIT_LIMIT > ALL ( SELECT CREDIT_LIMIT
FROM CUSTOMERS
WHERE CUST_REP = 109);
```

CUST_NUM	CUST_REP	CREDIT_LIMIT
2118	108	60000
2101	106	65000
2102	101	65000
2106	102	65000

9) Do question 8, still using the subquery but do not use the "all" keyword.

```
SELECT CUST_NUM, CUST_REP, CREDIT_LIMIT
FROM CUSTOMERS
WHERE CREDIT_LIMIT > ( SELECT MAX(CREDIT_LIMIT)
FROM CUSTOMERS
WHERE CUST_REP = 109);
```

CUST_NUM	CUST_REP	CREDIT_LIMIT
2102	101	65000
2101	106	65000
2106	102	65000
2118	108	60000

10) Use sub query and "in" keyword to print the salesreps (ids) who have taken order for the companies 'Zetacorp' or 'JCP Inc.' . Duplicate rows is not allowed

```
SELECT DISTINCT REP
FROM ORDERS
WHERE CUST IN (SELECT CUST_NUM
FROM CUSTOMERS
WHERE COMPANY IN ('Zetacorp','JCP Inc.'));
```

```
       REP
----------
       108
       105
       103
```

11) Use sub query to find the id and the name of every sales rep that represents at least one customer with a credit limit of greater than $5000.

```
SELECT EMPL_NUM, NAME
FROM SALESREPS
WHERE EMPL_NUM IN ( SELECT CUST_REP
FROM CUSTOMERS
WHERE (CREDIT_LIMIT > 50000));
```

```
 EMPL_NUM NAME
---------- ---------------
       101 Dan Roberts
       106 Sam Clark
       109 Mary Jones
       102 Sue Smith
       108 Larry Fitch
```

12) Use sub query and keyword “exists” to list the id and the name of the salesreps in which some customers have orders some products in their hiredate.

```
SELECT empl_num, name
FROM salesreps
WHERE EXISTS ( SELECT *
FROM ORDERS
WHERE ORDER_DATE = hire_date);
```

```
 EMPL_NUM NAME
---------- ---------------
       108 Larry Fitch
       109 Mary Jones
```

13) List all the products (Mfr_ID and Product_ID) that have never been sold. Use the ‘Exists’ clause.

```
SELECT MFR_ID, PRODUCT_ID
FROM PRODUCTS
WHERE NOT EXISTS( SELECT *
```

FROM ORDERS
WHERE PRODUCT= PRODUCT_ID);

```
MFR PRODU
--- -----
ACI 41001
IMM 887P
IMM 887X
QSA XK48
QSA XK48A
BIC 41672
BIC 41089
IMM 887F
```

Updates Questions:

14) Insert the following information into the OFFICES table:
Office: 39 City: Miami Region: Southern Manager: 106 Target: 1000000 Sales: 0

```
INSERT INTO offices(office,city, region,mgr,target,sales)
    VALUES (39,'Miami','Southern',106,1000000,0);
```

```
1 row created.
```

15) Write an insert statement to add Your Name as Empl_Num 772. Use the date the insert is run as the Hire date (sysdate). Sales are zero. Other column remain NULL;

```
INSERT INTO salesreps(empl_num,name,hire_date,sales)
VALUES(722,'Eduardo',SYSDATE,0);
```

```
1 row created.
```

16) Write an insert statement to add 'Tom Sawyer' Empl_Num 814. Use the date the insert is run as the Hire date (sysdate). Sales are zero. Use implicit null values for columns that are not mentioned.

```
INSERT INTO
salesreps(empl_num,name,age,rep_office,title,hire_date,manager,quota,sales)
VALUES(814,'Tom Sawyer',NULL,NULL,NULL,SYSDATE,NULL,NULL,0);
```

```
1 row created.
```

17) Delete all orders for employees 108, 101, 102.

```
DELETE FROM ORDERS
WHERE  REP IN(108,101,102);
```

```
13 rows deleted.
```

18) Delete all sales reps that have no orders and were hired before Jan 1 1987.

```
DELETE FROM salesreps
WHERE empl_num NOT IN (SELECT REP
FROM ORDERS
WHERE REP = empl_num) AND
hire_date < '01-JAN-87';
```

```
0 rows deleted.
```

19) Update your employee record with the following:
Age: 37 Rep_Office:39 Title: Senior VP Manager: NULL Quota: 100000

```
UPDATE salesreps
SET age=37, rep_office=39, title='Senior VP', manager=NULL, quota=100000
WHERE name = 'Eduardo';
```

```
1 row updated.
```

20) Increase customers credit limit by 20% for all customers that have 2 or more orders in which each order is more than 25,000.

```
UPDATE CUSTOMERS
SET CREDIT_LIMIT = CREDIT_LIMIT * 1.25
WHERE 2 <= ( SELECT COUNT(CUST)
FROM ORDERS
WHERE (CUST_NUM = CUST) AND AMOUNT > 25000);
```

```
0 rows updated.
```

21) Increase the credit limit of any customer who has any order that exceeds their credit limit. The new credit limit should be set to their maximum order amount plus $1,000. This must be done in 1 SQL statement.

```
UPDATE CUSTOMERS
SET CREDIT_LIMIT = ( SELECT MAX(AMOUNT)
FROM ORDERS, CUSTOMERS
WHERE CUST = CUST_NUM) + 1000
WHERE CREDIT_LIMIT < ANY ( SELECT AMOUNT
FROM ORDERS
WHERE CUST = CUST_NUM);
```

```
1 row updated.
```

Views and Security Questions

22) Create a view to show the Sales rep Name, and city that the Sales rep works in.

```
CREATE VIEW NameCity AS
SELECT name, city
FROM salesreps, offices
WHERE rep_office = office;
```

```
View created.
```

23) Grant select access of the view created in question 22 to your Database instructors: Ahmad R. Hadaegh (with user id ahadaegh).

```
GRANT SELECT
ON NameCity
TO ahadaegh;
```

```
Grant succeeded.
```

24) Create a view to show the customer name, product, description, quantity ordered and value of parts ordered. The column heading for the customers name should be 'CustName' and the column heading for value of parts ordered should be 'Value'.

```
CREATE VIEW CustInfo AS
SELECT COMPANY AS "CustName", PRODUCT, DESCRIPTION, QTY, AMOUNT
AS "VALUE"
FROM CUSTOMERS, ORDERS, PRODUCTS
WHERE CUST=CUST_NUM AND PRODUCT=PRODUCT_ID;
```

```
View created.
```

25) Grant select access of the view created in question 24 to public

```
GRANT SELECT
ON CustInfo
TO PUBLIC;
```

```
Grant succeeded.
```

26) Revoke access on view created in question 24 from Public.

```
REVOKE SELECT
ON CustInfo
```

FROM PUBLIC;

```
Revoke succeeded.
```

27) Using the view created in question 24 above, list all information for product ‘L14’.

```
SELECT *
FROM CustInfo
WHERE PRODUCT = 'L14';
```

```
no rows selected
```

28) Create a view called TheManagers to list the name of all sales reps that manage some office. Along with the managers name, list the office number and city for each office.

```
CREATE VIEW TheManagers AS
SELECT name, manager, rep_office,city
FROM salesreps, offices
WHERE rep_office = office AND empl_num = ANY(SELECT manager
FROM salesreps);
```

```
View created.
```

29) Grant all privileges on the view created in question 28 to your instructor.

```
GRANT ALL PRIVILEGES
ON TheManagers
TO ahadaegh;
```

```
Grant succeeded.
```

30) Grant Select, Insert and Update on the Offices table to userids ‘jschmidt’ and ‘kmart’.

```
GRANT SELECT, INSERT, UPDATE
ON offices
TO jschmidt, kmart;
```

```
   *
ERROR at line 3:
ORA-01917: user or role 'JSCHMIDT' does not exist
```