USE classicmodels;

/\*1. List the names of the cities in alphabetical order where Classic Models has offices\*/

SELECT city AS 'cityNames' FROM offices

ORDER BY city asc;

| cityNames |
| --- |
| Boston |
| London |
| NYC |
| Paris |
| San Francisco |
| Sydney |
| Tokyo |

/\*2. List the EmployeeNumber, LastName, FirstName, Extension for all employees working out of the Paris office\*/

select employeeNumber, lastName, firstName, extension

from employees

where officeCode=4;

| employeeNumber | lastName | firstName | extension |
| --- | --- | --- | --- |
| 1102 | Bondur | Gerard | x5408 |
| 1337 | Bondur | Loui | x6493 |
| 1370 | Hernandez | Gerard | x2028 |
| 1401 | Castillo | Pamela | x2759 |
| 1702 | Gerard | Martin | x2312 |

/\*3. List the ProductCode, ProductName, ProductVendor, QuantityInStock and ProductLine for all products

with a QuantityInStock between 200 and 1200. \*/

select ProductCode, productName, productVendor, quantityInStock, productLine

from products

where quantityInStock > 200 and quantityInStock < 1200;

| ProductCode | productName | productVendor | quantityInStock | productLine |
| --- | --- | --- | --- | --- |
| S12\_3891 | 1969 Ford Falcon | Second Gear Diecast | 1049 | Classic Cars |
| S18\_2248 | 1911 Ford Town Car | Motor City Art Classics | 540 | Vintage Cars |
| S18\_2581 | P-51-D Mustang | Gearbox Collectibles | 992 | Planes |
| S18\_2795 | 1928 Mercedes-Benz SSK | Gearbox Collectibles | 548 | Vintage Cars |
| S24\_1046 | 1970 Chevy Chevelle SS 454 | Unimax Art Galleries | 1005 | Classic Cars |
| S32\_3522 | 1996 Peterbilt 379 Stake Bed with Outrigger | Red Start Diecast | 814 | Trucks and Buses |
| S50\_1392 | Diamond T620 Semi-Skirted Tanker | Highway 66 Mini Classics | 1016 | Trucks and Buses |
| S50\_4713 | 2002 Yamaha YZR M1 | Autoart Studio Design | 600 | Motorcycles |
| S700\_1938 | The Mayflower | Studio M Art Models | 737 | Ships |
| S700\_3167 | F/A 18 Hornet 1/72 | Motor City Art Classics | 551 | Planes |
| S72\_3212 | Pont Yacht | Unimax Art Galleries | 414 | Ships |

/\*4. (Use a SUBQUERY) List the ProductCode, ProductName, ProductVendor, BuyPrice and MSRP for the

least expensive (lowest MSRP) product sold by ClassicModels. (“MSRP” is the Manufacturer’s Suggested

Retail Price.) \*/

select productCode, productName, productVendor, buyPrice, MSRP

from products

where MSRP=(select min(MSRP) from products);

| productCode | productName | productVendor | buyPrice | MSRP |
| --- | --- | --- | --- | --- |
| S24\_1937 | 1939 Chevrolet Deluxe Coupe | Motor City Art Classics | 22.57 | 33.19 |

/\*5. What is the ProductName and Profit of the product that has the highest profit (profit = MSRP minus

BuyPrice). \*/

select (MSRP-buyPrice) as 'profit', productName from products

where (MSRP-buyPrice)=(select max(MSRP-buyPrice) from products);

| profit | productName | |
| --- | --- | --- |
| 115.72 | 1952 Alpine Renault 1300 | |

/\* 6. List the country and the number of customers from that country for all countries having just two

customers. List the countries sorted in ascending alphabetical order. Title the column heading for the

count of customers as “Customers”. \*/

select country, count(customerName) AS 'Customers'

from customers GROUP BY country

HAVING COUNT(customerName) = 2

order by country asc;

| country | Customers |
| --- | --- |
| Austria | 2 |
| Belgium | 2 |
| Denmark | 2 |
| Ireland | 2 |
| Japan | 2 |
| Portugal | 2 |
| Sweden | 2 |

/\*7. List the ProductCode, ProductName, and number of orders for the products with exactly 25 orders.

Title the column heading for the count of orders as “OrderCount” \*/

select products.productCode, products.productName, count(orderNumber) AS 'OrderCount'

from products JOIN orderDetails

on orderdetails.productCode = products.productCode

group by productCode

having OrderCount = 25;

| productCode | productName | OrderCount |
| --- | --- | --- |
| S18\_1589 | 1965 Aston Martin DB5 | 25 |
| S18\_1749 | 1917 Grand Touring Sedan | 25 |
| S18\_2248 | 1911 Ford Town Car | 25 |
| S18\_2870 | 1999 Indy 500 Monte Carlo SS | 25 |
| S18\_3685 | 1948 Porsche Type 356 Roadster | 25 |
| S18\_4409 | 1932 Alfa Romeo 8C2300 Spider Sport | 25 |
| S24\_1046 | 1970 Chevy Chevelle SS 454 | 25 |
| S24\_1628 | 1966 Shelby Cobra 427 S/C | 25 |
| S24\_2766 | 1949 Jaguar XK 120 | 25 |
| S24\_3191 | 1969 Chevrolet Camaro Z28 | 25 |
| S24\_3432 | 2002 Chevy Corvette | 25 |
| S24\_3969 | 1936 Mercedes Benz 500k Roadster | 25 |

/\*8. List the EmployeeNumber, Firstname + Lastname (concatenated into one column in the answer set,

separated by a blank and referred to as ‘name’) for all the employees reporting to Diane Murphy or

Gerard Bondur \*/

select employeeNumber, concat(firstName,' ',lastName) AS 'Name'

from employees where reportsTo=1002 or reportsTo=1102;

| employeeNumber | Name |
| --- | --- |
| 1056 | Mary Patterson |
| 1076 | Jeff Firrelli |
| 1337 | Loui Bondur |
| 1370 | Gerard Hernandez |
| 1401 | Pamela Castillo |
| 1501 | Larry Bott |
| 1504 | Barry Jones |
| 1702 | Martin Gerard |

/\*9. List the EmployeeNumber, LastName, FirstName of the president of the company (the one employee

with no boss.) \*/

select employeeNumber, lastName, firstName

from employees

where jobTitle = 'President';

| employeeNumber | lastName | firstName |
| --- | --- | --- |
| 1002 | Murphy | Diane |

/\*10. List the ProductName for all products in the “Classic Cars” product line from the 1950’s\*/

select productName from products P join productlines PL

on P.productLine = PL.productLine where P.productLine = 'Classic Cars'

and P.productName like '195\_%';

| productName | |
| --- | --- |
| 1952 Alpine Renault 1300 | |
| 1957 Corvette Convertible | |
| 1957 Ford Thunderbird | |
| 1958 Chevy Corvette Limited Edition | |
| 1952 Citroen-15CV | |
| 1956 Porsche 356A Coupe | |

/\*11. List the month name and the total number of orders for the month in 2004 in which ClassicModels

customers placed the most orders

finish\*/

SELECT MONTHNAME(orderDate) AS 'month', COUNT(orderNumber) AS 'numOrders'

FROM orders WHERE YEAR(orderDate) = 2004

GROUP BY MONTHNAME(orderDate) ORDER BY numOrders DESC LIMIT 1;

| month | numOrders |
| --- | --- |
| November | 33 |

/\*12. List the firstname, lastname of employees who are Sales Reps who have no assigned customers.\*/

SELECT firstname, lastname FROM Employees JOIN Customers

WHERE employees.jobTitle = 'sales rep' AND

NOT(employees.employeeNumber IN (select salesRepEmployeeNumber from customers where salesrepemployeenumber != 'Null'))

GROUP BY firstname, lastname, employeeNumber;

| firstname | lastname |
| --- | --- |
| Yoshimi | Kato |
| Tom | King |

/\*13. List the customername of customers from Switzerland with no orders.\*/

SELECT customerName FROM customers WHERE NOT((customerNumber IN(SELECT customerNumber FROM Payments)))

AND country = 'switzerland';

| customerName |
| --- |
| BG&E Collectables |
| Precious Collectables |

/\*14. List the customername and total quantity of products ordered for customers who have ordered

more than 1650 products across all their orders \*/

SELECT customerName, SUM(quantityOrdered) AS total FROM customers c, orders o, orderdetails od

WHERE c.customerNumber = o.customerNumber

AND o.orderNumber = od.orderNumber

GROUP BY customerName HAVING SUM(quantityOrdered) > 1650;

| customerName | total |
| --- | --- |
| Australian Collectors, Co. | 1926 |
| La Rochelle Gifts | 1832 |
| Mini Gifts Distributors Ltd. | 6366 |
| Euro+ Shopping Channel | 9327 |
| Muscle Machine Inc | 1775 |
| AV Stores, Co. | 1778 |
| Down Under Souveniers, Inc | 1691 |
| The Sharp Gifts Warehouse | 1656 |

/\* 15. Create a NEW table named “TopCustomers” with three columns: CustomerNumber (integer),

ContactDate (DATE) and OrderTotal (a decimal number with 9 digits in total having two decimal places).

None of these columns can be NULL. Include a PRIMARY KEY constraint named “TopCustomer\_PK” on

CustomerNumber. (no answer set)\*/

CREATE TABLE TopCustomers (

CustomerNumber int NOT NULL,

ContactDate DATE NOT NULL,

OrderTotal DECIMAL(9, 2) NOT NULL,

CONSTRAINT TopCustomer\_PK PRIMARY KEY (CustomerNumber)

);

/\* 16. Populate the new table “TopCustomers” with the CustomerNumber, today’s date, and the total

value of all their orders (PriceEach \* quantityOrdered) for those customers whose order total value is

greater than $140,000. (inserted 10 rows, no answer set)\*/

INSERT INTO TopCustomers (customerNumber, contactDate, orderTotal)

SELECT customerNumber, current\_date(), sum(productTotal) AS 'orderTotal'

FROM (SELECT o.orderNumber, customerNumber, (priceEach\*quantityOrdered) AS 'productTotal'

FROM orders o join orderdetails od on o.orderNumber = od.ordernumber) as sumTot

GROUP BY customerNumber HAVING orderTotal > 140000

ORDER BY orderTotal DESC;

/\* 17. List the contents of the TopCustomers table in descending OrderTotal sequence.\*/

SELECT \* FROM TopCustomers

ORDER BY OrderTotal DESC;

| CustomerNumber | ContactDate | OrderTotal |
| --- | --- | --- |
| 141 | 2021-10-10 | 820689.5 |
| 124 | 2021-10-10 | 591827.3 |
| 114 | 2021-10-10 | 180585.1 |
| 151 | 2021-10-10 | 177914 |
| 119 | 2021-10-10 | 158573.1 |
| 148 | 2021-10-10 | 156251 |
| 323 | 2021-10-10 | 154622.1 |
| 131 | 2021-10-10 | 149085.2 |
| 187 | 2021-10-10 | 148410.1 |
| 450 | 2021-10-10 | 143536.3 |

/\* 18. Add a new column to the TopCustomers table called OrderCount (integer)\*/

ALTER TABLE TopCustomers

ADD OrderCount INT;

/\* 19. Update the Top Customers table, setting the OrderCount column to a random number (from 0 to

18). (Should update 10 rows) HINT: use the RAND() and FLOOR() functions\*/

UPDATE TopCustomers

SET OrderCount = (SELECT FLOOR(RAND()\*(18-0+1)+0))

WHERE CustomerNumber > 0;

/\* 20. List the contents of the TopCustomers table in descending OrderCount sequence.\*/

SELECT \* FROM TopCustomers

ORDER BY OrderCount DESC;

| CustomerNumber | ContactDate | OrderTotal | OrderCount |
| --- | --- | --- | --- |
| 450 | 2021-10-10 | 143536.3 | 18 |
| 323 | 2021-10-10 | 154622.1 | 15 |
| 131 | 2021-10-10 | 149085.2 | 14 |
| 148 | 2021-10-10 | 156251 | 13 |
| 119 | 2021-10-10 | 158573.1 | 12 |
| 141 | 2021-10-10 | 820689.5 | 7 |
| 187 | 2021-10-10 | 148410.1 | 6 |
| 151 | 2021-10-10 | 177914 | 5 |
| 124 | 2021-10-10 | 591827.3 | 2 |
| 114 | 2021-10-10 | 180585.1 | 0 |

/\*21. Drop the TopCustomers table\*/

DROP TABLE TopCustomers;