Question1:

SELECT distinct city

FROM customers

where country = 'spain'

order by city;

Graphical user interface, application, table

Description automatically generated with medium confidence

Question2:

SELECT employeeNumber, firstName, firstName

FROM employees

where officeCode in (

select officeCode

from offices

where city = 'Paris' )

Table

Description automatically generated

Question3:

SELECT ProductCode, ProductName, ProductScale, ProductVendor, buyPrice

FROM products

where productline ='Motorcycles'

having buyPrice<80 AND buyPrice >50

Table

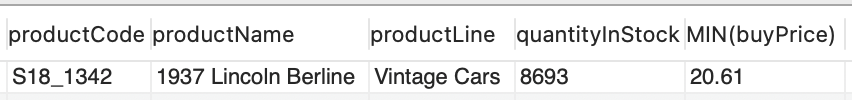
Description automatically generated

Question4:

SELECT productCode, productName, productLine, quantityInStock, MIN(buyPrice)

FROM products

where productLine='Vintage Cars'



Question5:

SELECT productName, productVendor, quantityinStock, quantityOrdered \* priceEach as 'total cost'

FROM orderdetails o, products p

where o.productCode=p.productCode

order by quantityOrdered \* priceEach desc limit 5;

Table

Description automatically generated

Question6:

SELECT customerNumber, customerName, phone, country, state, salesRepEmployeeNumber, creditLimit

FROM customers

where creditLimit>130000

order by creditLimit

Table

Description automatically generated

Question7:

use classicmodels;

SELECT p.productCode, p.productName, quantityOrdered as 'OrderCount'

FROM orderdetails o, products p

where o.productCode=p.productCode AND p.productVendor='Welly Diecast Productions'

order by quantityOrdered ASC limit 1

Graphical user interface, text

Description automatically generated

Question8:

use classicmodels;

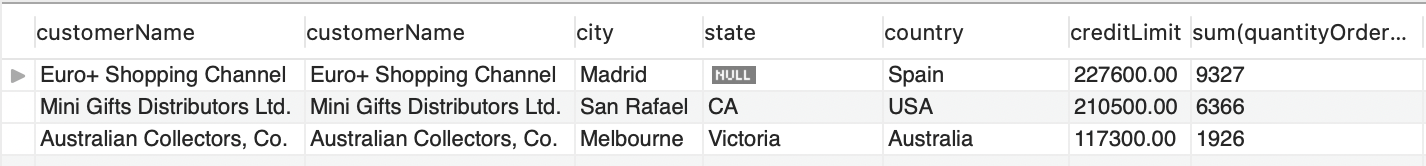
SELECT c.customerName, c.customerName, c.city, c.state, c.country, c.creditLimit, sum(quantityOrdered)

FROM customers c, orders o, orderdetails d

where c.customerNumber=o.customerNumber AND d.orderNumber=o.orderNumber

group by c.customerNumber

order by sum(quantityOrdered) desc limit 3



Question9:

use classicmodels;

SELECT OfficeCode, city, state, country

FROM offices

where NOT country= 'USA' AND addressLine2 is null

Table

Description automatically generated

Question10:

use classicmodels;

SELECT productName, productLine

FROM products

WHERE productName like '193%' AND productLine='Vintage Cars'

Table

Description automatically generated

Question11.

use classicmodels;

SELECT orderNumber, orderDate, requiredDate, shippedDate, Extract(Month from shippedDate) as 'ship month', requiredDate-shippedDate as 'date difference'

FROM orders

where requiredDate-shippedDate<3 AND YEAR( shippedDate)=2005;

Table

Description automatically generated

Question12:

use classicmodels;

SELECT c.customerNumber, c.customerName, c.city, c.country, count(o.orderNumber) as 'Orders'

FROM customers c, orders o

where c.customerNumber=o.customerNumber and c.customerNumber<150

group by c.customerNumber

order by count(o.orderNumber) desc

Text, table

Description automatically generated

Question13:

use classicmodels;

SELECT c.customerNumber, c.customerName

FROM customers c

where c.customerNumber not in(

select customerNumber

from payments )AND c.country='Switzerland '

Table

Description automatically generated

Question14:

use classicmodels;

use classicmodels;

SELECT sum(o.quantityOrdered), p.productLine

FROM orderdetails o, products p

where o.productCode=p.productCode

group by p.productLine

having sum(o.quantityOrdered)>12000

Table

Description automatically generated

Question 15:

CREATE TABLE if not exists TopCustomers (

CustomerNumber int not null,

OrderCount int not null,

OrderTotal decimal(9,2) not null,

ContactDate date not null,

primary key (CustomerNumber)

);

Question16:

INSERT INTO TopCustomers (CustomerNumber, ContactDate, OrderCount, OrderTotal)

select\* from( select customerNumber ,curdate(),count(\*), sum(orderdetails.quantityOrdered\* orderdetails.priceEach) as total from orders join orderdetails

on orders.orderNumber=orderdetails.orderNumber group by customerNumber)as n

where n.total>130000;

Question17:

select \*

from TopCustomers

order by OrderTotal limit 5

Table

Description automatically generated

Question18:

ALTER TABLE TopCustomers2

ADD COLUMN CustomerRatings INT DEFAULT 0 AFTER ContactDate ;

Question19:

UPDATE TopCustomers2

SET CustomerRatings=RAND()\*(11-1)+1

WHERE CustomerNumber IS NOT NULL;



Question20：

SELECT \*

FROM TopCustomers2

ORDER BY CustomerRatings DESC;

Table

Description automatically generated

QUESTION21：

ALTER TABLE TopCustomers2

DROP COLUMN CustomerRatings;