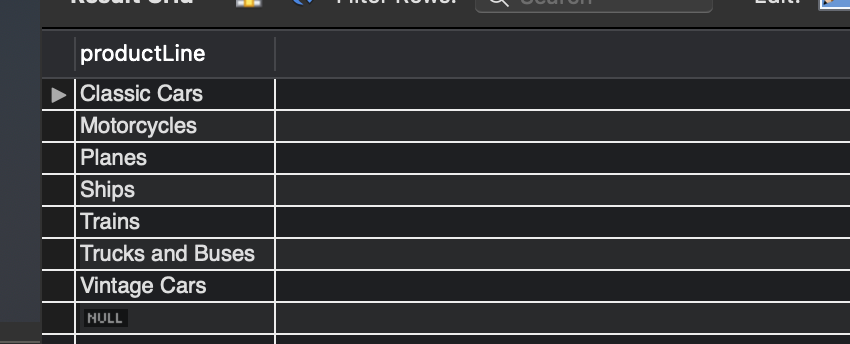
Homework 3

Name: Chakrya Ros

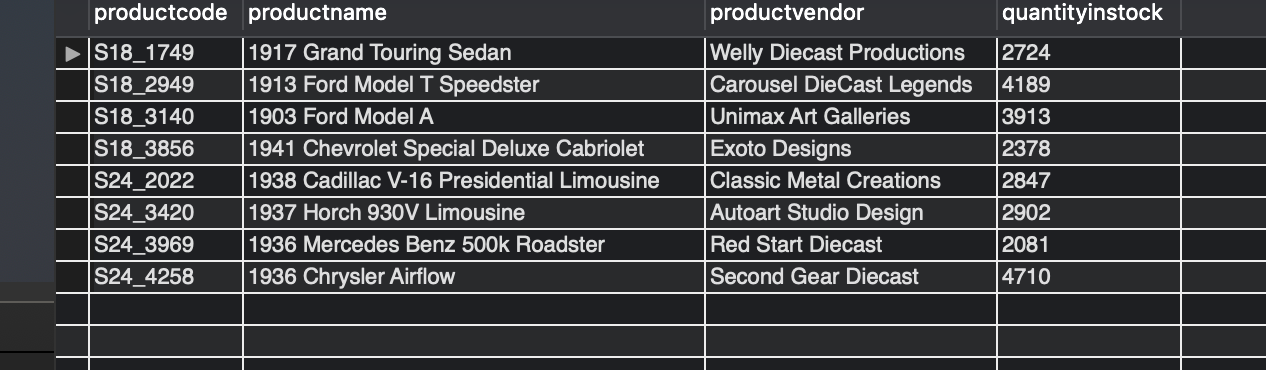
1. select productLine from productlines;



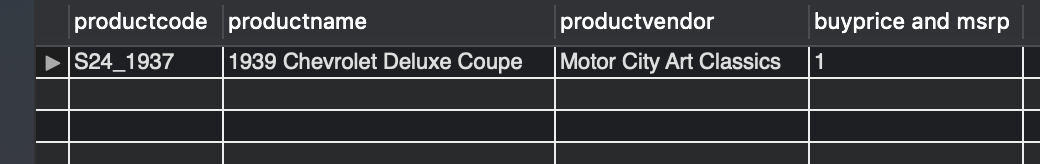
2. select employeeNumber, lastName, firstName, Extension from employees where officeCode = 1;



3. select productcode,productname,productvendor,quantityinstock from products where productline='Vintage Cars' and QuantityInStock between 1000 and 5000;



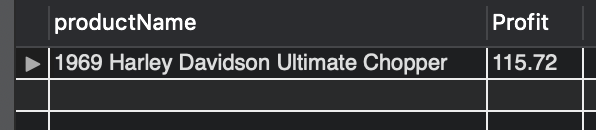
4. select productcode,productname,productvendor,buyprice and msrp from products where MSRP = (select min(msrp) from products);



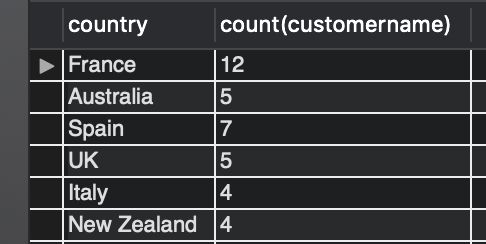
5.

|  |
| --- |
|  |
|  |  |

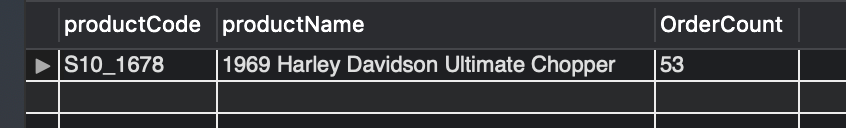
select productName,max((MSRP-buyPrice)) as 'Profit' from products;



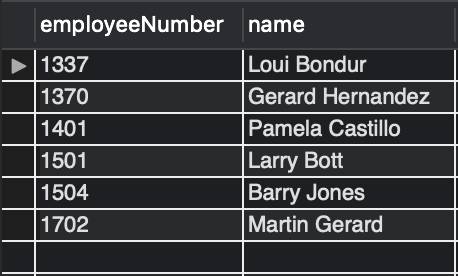
6. select country, count(customername) from customers group by country having count(customername) >= 4 and count(customername) <= 12;



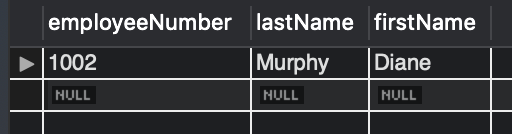
7. select x.productCode, products.productName, max(x.productCount) as OrderCount from (select productCode, count(\*) as productCount from orderdetails group by productCode) as x join products on products.productCode = x.productCode;



8. select employeeNumber, concat(firstName,' ',lastName) as name from employees where reportsTo = 1102 and 1002;



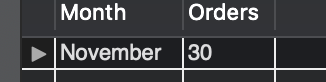
9. select employeeNumber, lastName, firstName from employees where jobTitle = "President";



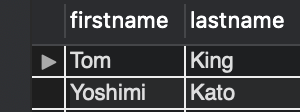
10. select productName from (select productName, substring(productName, 1, 4)as year from products where productLine = "Vintage Cars") as x where year >= '1930' and year < '1940';



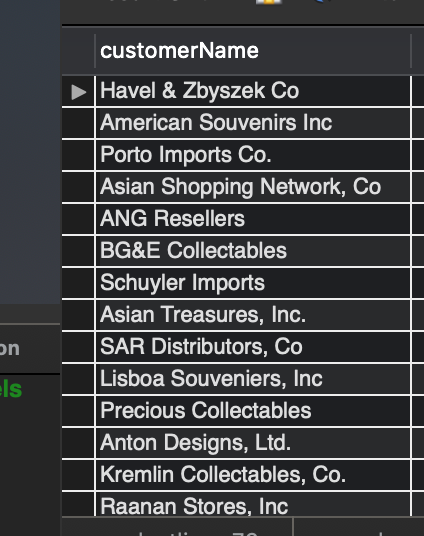
11. select monthname(orderdate) as 'Month', count(ordernumber) as 'Orders' from orders where year(orderdate)=2003 group by monthname(orderdate) order by count(ordernumber) desc limit 1;



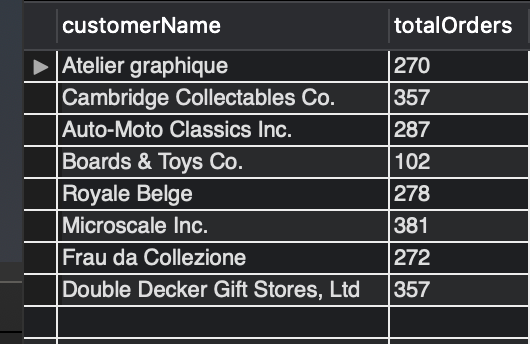
12. select employees.firstname,employees.lastname from employees left join customers on employees.employeenumber=customers.salesrepemployeenumber where employees.jobtitle='Sales Rep' and customers.salesrepemployeenumber is null;



13. select customers.customerName from customers where country != "Germany" and not exists (select \* from orders where customerNumber = customers.customerNumber);



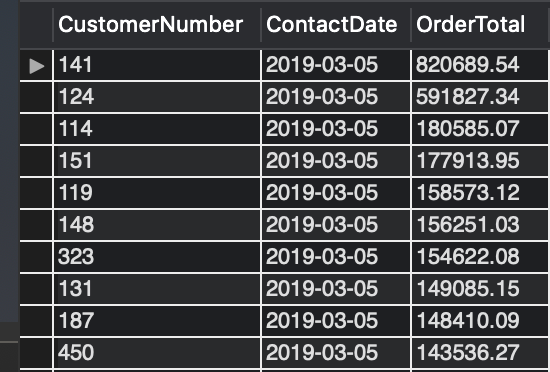
14. select customerName, totalOrders from (select customerNumber, sum(quantityOrdered) as totalOrders from orderdetails join orders on orders.orderNumber = orderdetails.orderNumber group by customerNumber) as x join customers on customers.customerNumber = x.customerNumber where totalOrders < 400;



15. CREATE TABLE IF NOT EXISTS `classicmodels`.`TopCustomers` ( `CustomerNumber` INT NOT NULL, `ContactDate` DATE NOT NULL, `OrderTotal` DECIMAL(9,2) NOT NULL, CONSTRAINT TopCustomer\_PK PRIMARY KEY (`CustomerNumber`) );

16. insert into TopCustomers (customerNumber, ContactDate, OrderTotal) select customerNumber, now() as currentDate, totalValue from (select x.customerNumber, sum(x.totalValue) as totalValue from (select customerNumber, (orderdetails.priceEach \* orderdetails.quantityOrdered) as totalValue from orderdetails join orders on orders.orderNumber = orderdetails.orderNumber) as x group by x.customerNumber) as y where y.totalValue > 140000;

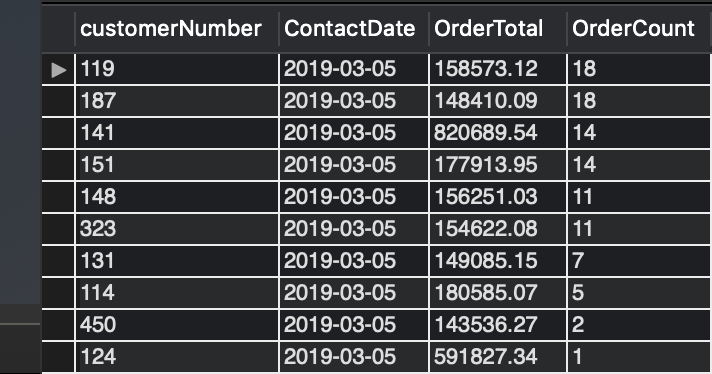
17. select CustomerNumber, ContactDate, OrderTotal from TopCustomers order by OrderTotal desc;



18. alter table TopCustomers add OrderCount int;

19. update TopCustomers set OrderCount = floor(rand() \* (18-0+1) + 0);

20. select customerNumber, ContactDate, OrderTotal, OrderCount from TopCustomers order by OrderCount desc;



21. Drop table TopCustomers;