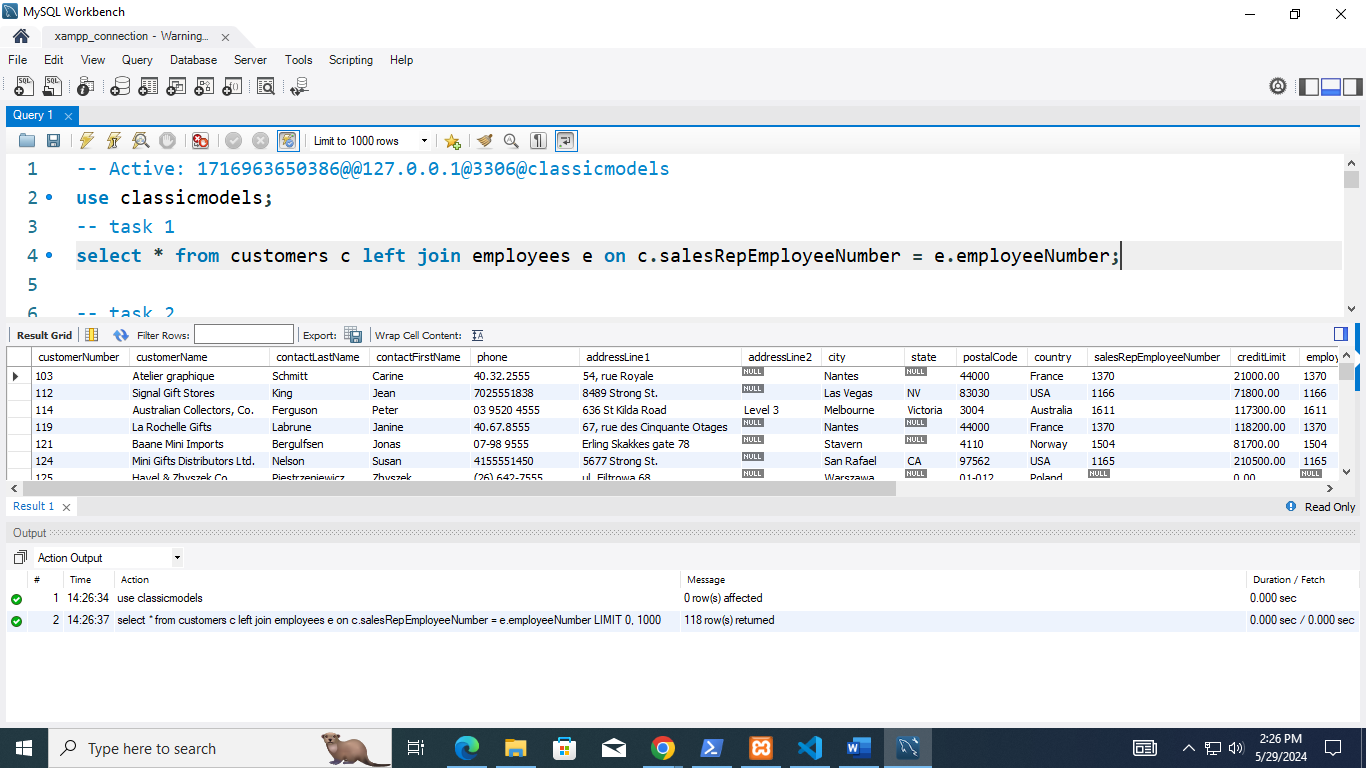
Introduction to Database Systems – Lab

JOINS (LEFT, RIGHT, OUTER) + UNION

Note: Use ‘**Classic Model’** database for following questions.

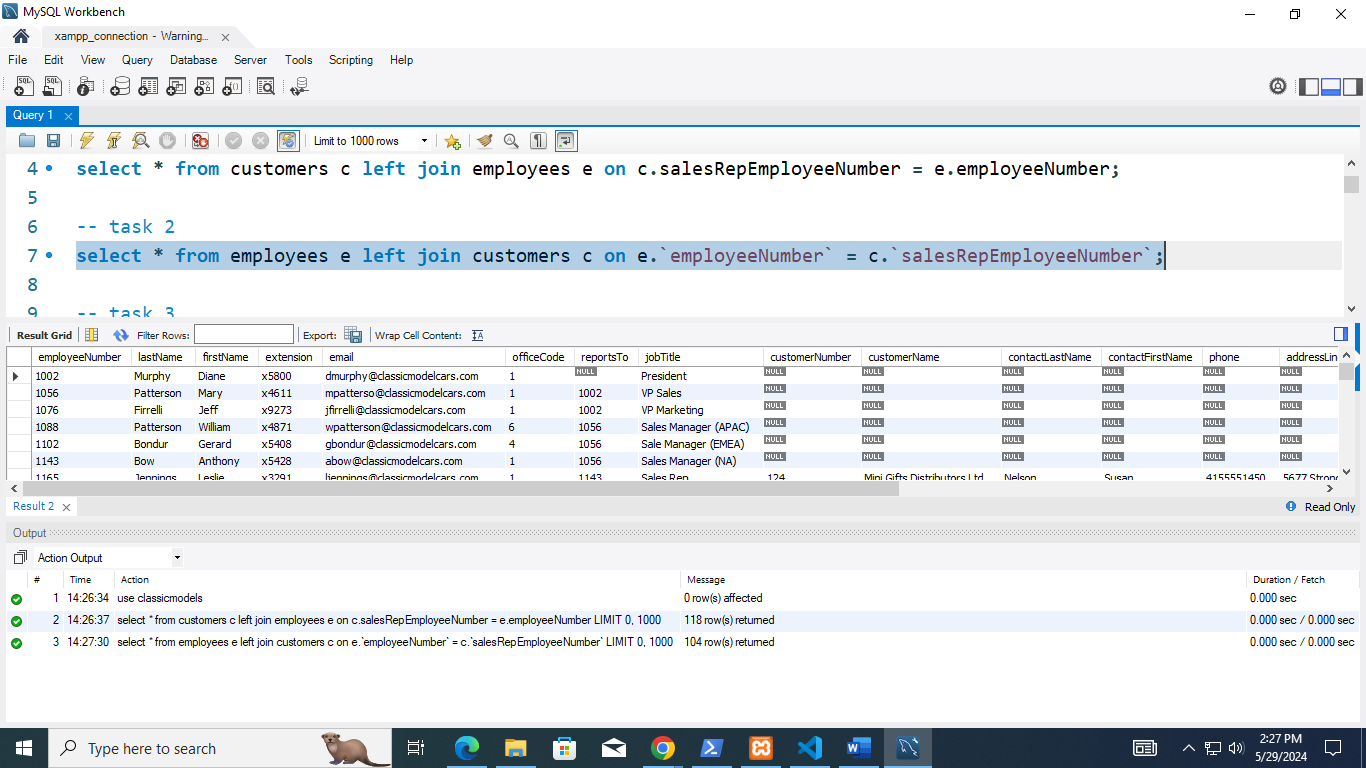
1. List all customers along with their sales representatives (employees who handle their accounts), including customers without a sales representative.

select \* from customers c left join employees e on c.salesRepEmployeeNumber = e.employeeNumber;



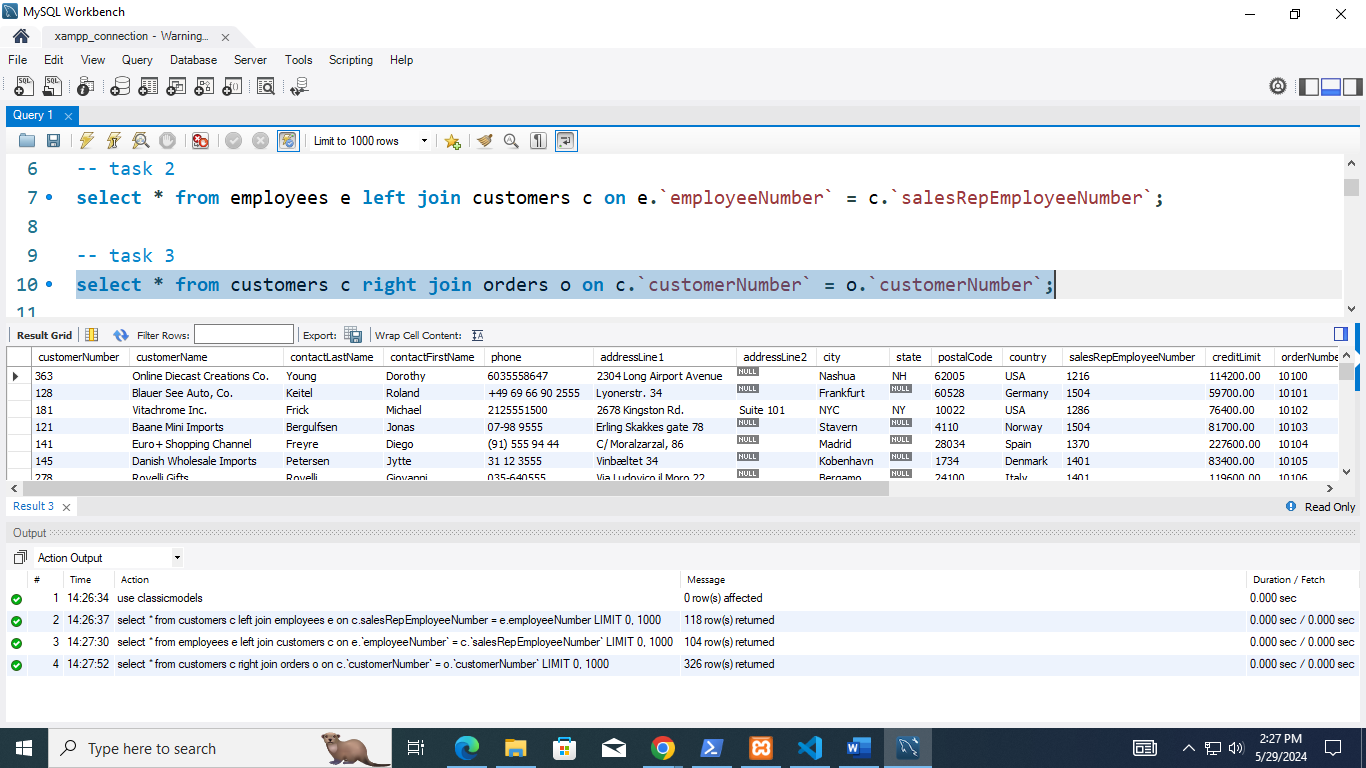
1. List all employees and the customers they handle, including employees who do not have any customers.

select \* from employees e left join customers c on e.`employeeNumber` = c.`salesRepEmployeeNumber`;



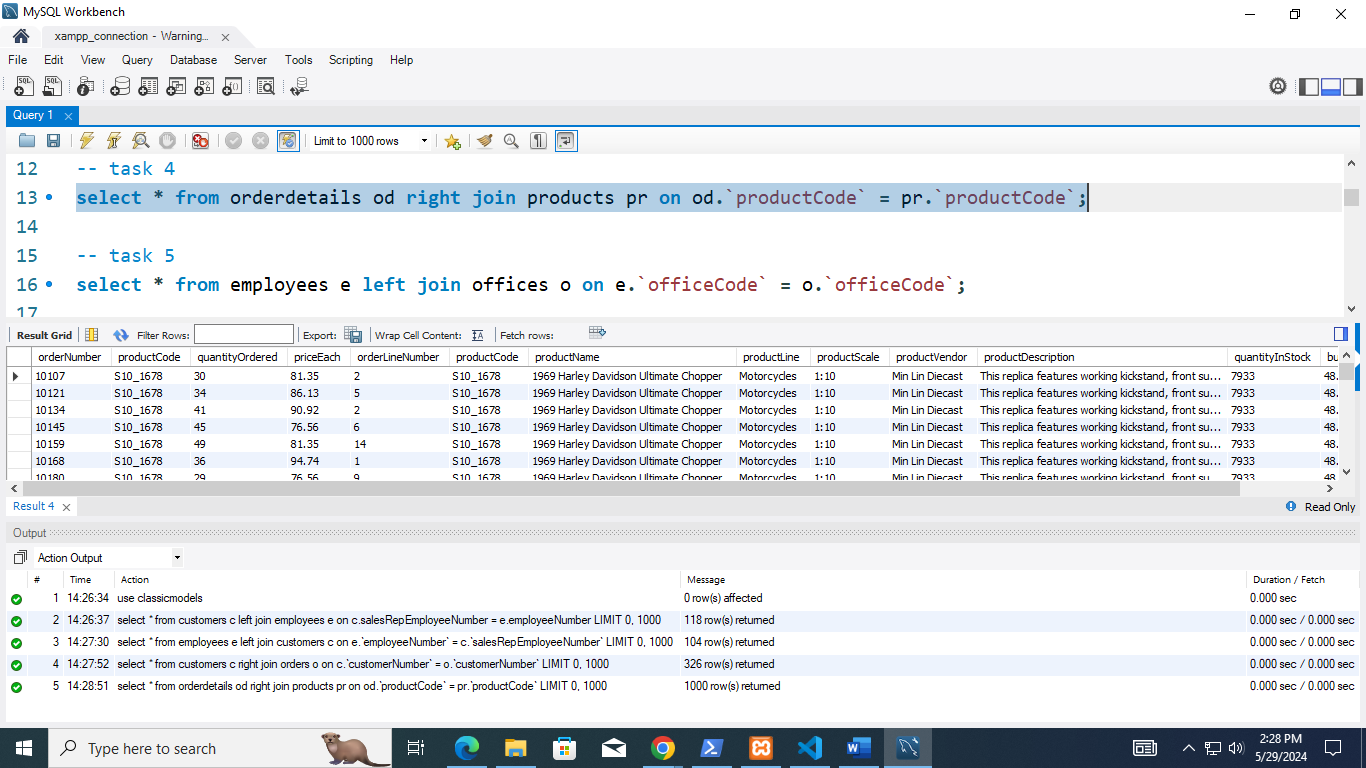
1. Display all orders along with the customer details, including orders that have no associated customer information.

select \* from customers c right join orders o on c.`customerNumber` = o.`customerNumber`;



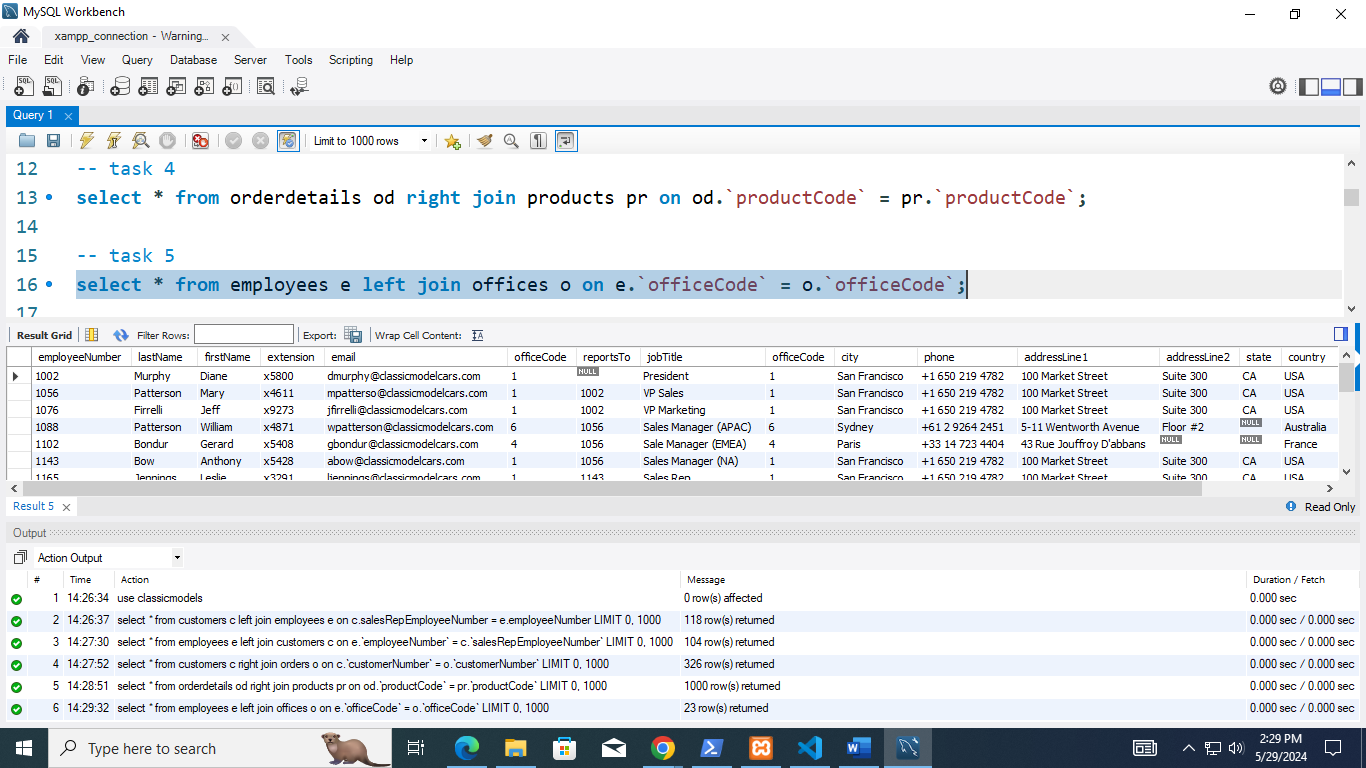
1. List all products along with the order details, including products that have never been ordered.

select \* from orderdetails od right join products pr on od.`productCode` = pr.`productCode`;

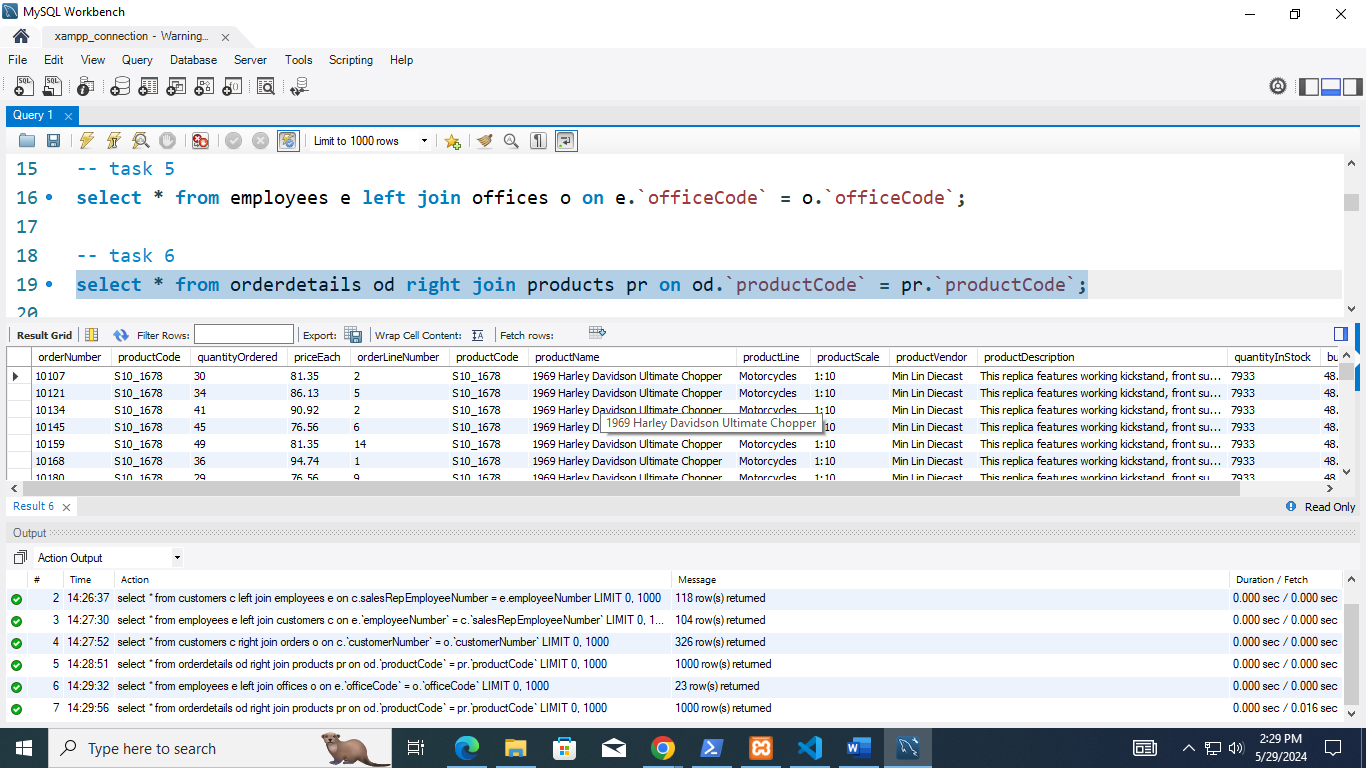


1. Display all employees along with the office location details, including employees who are not assigned to any office.

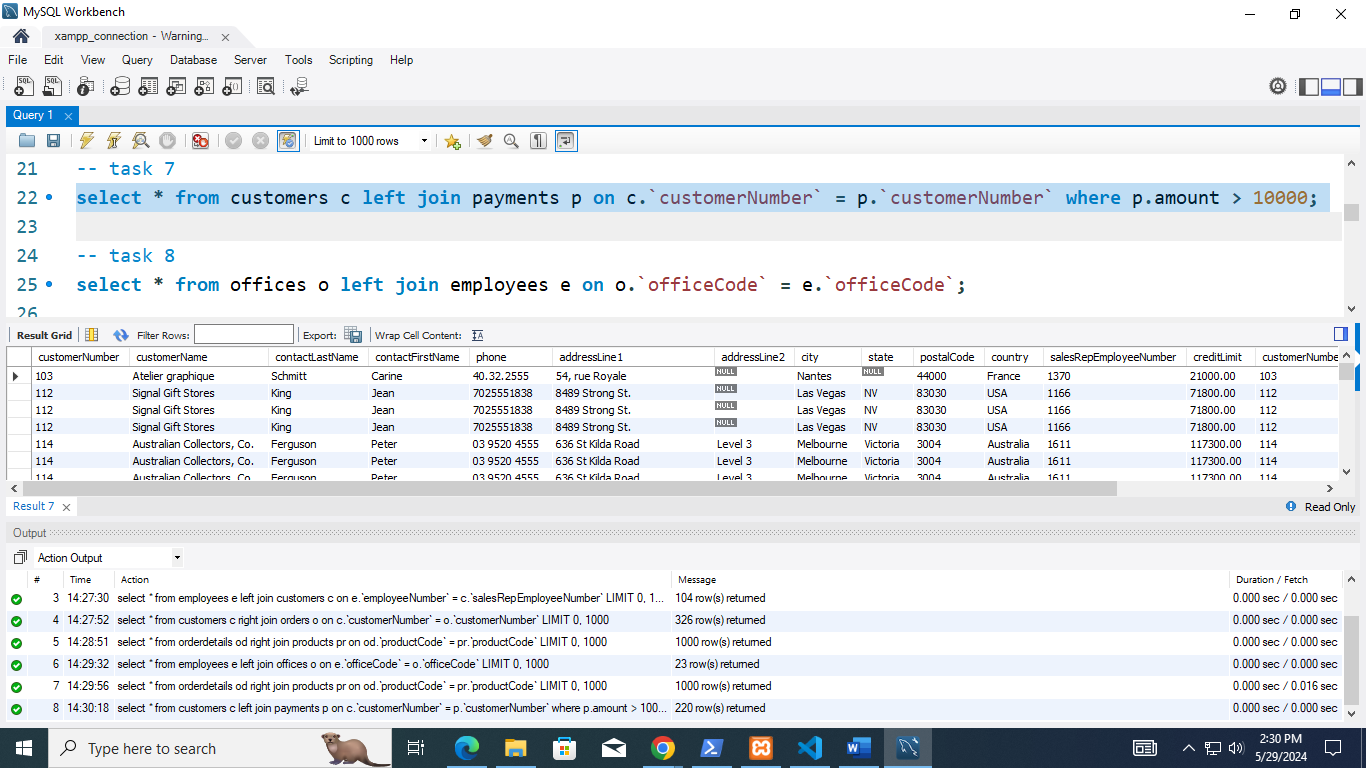
select \* from employees e left join offices o on e.`officeCode` = o.`officeCode`;



1. Retrieve all orders along with the products ordered, including orders that do not have any products.

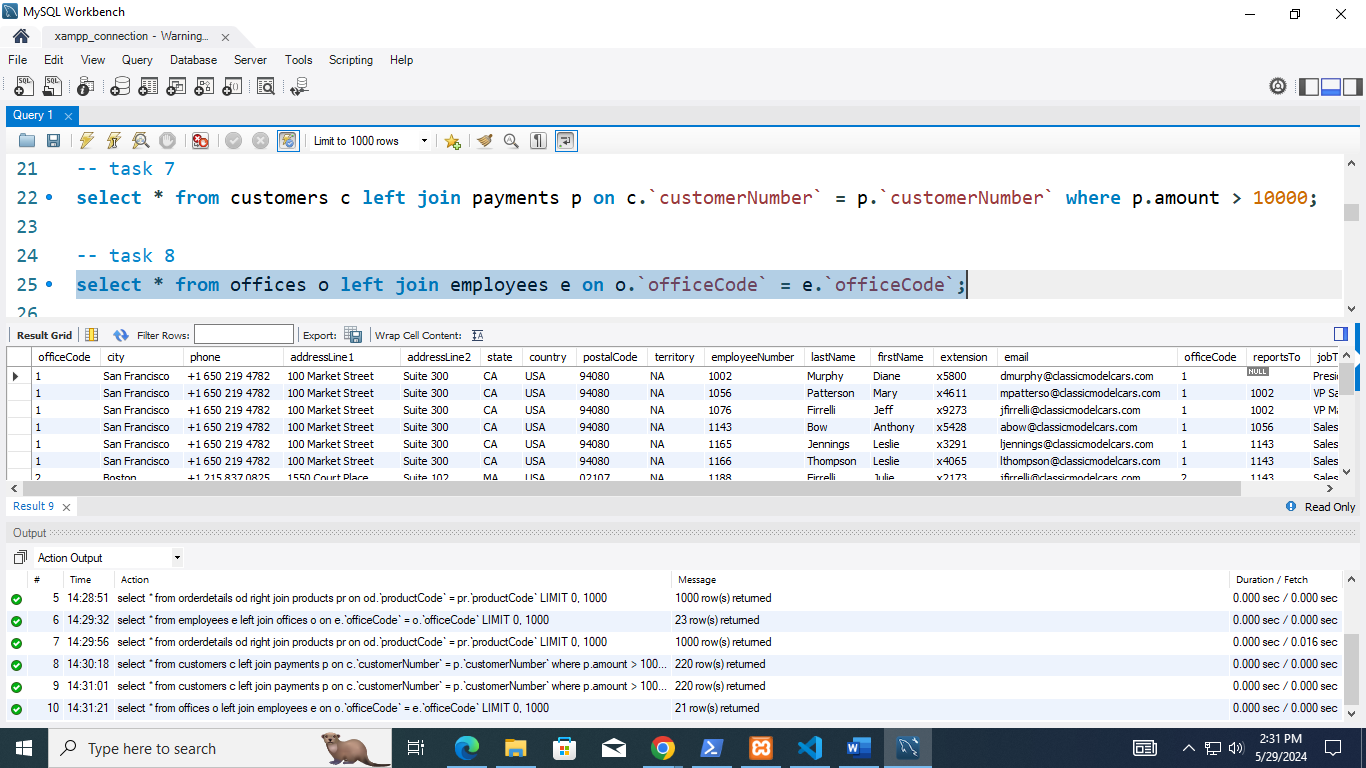
select \* from orderdetails od right join products pr on od.`productCode` = pr.`productCode`;

1. List all customers and their payments, including customers who have not made any payments for payment amount greater than 10000.

select \* from customers c left join payments p on c.`customerNumber` = p.`customerNumber` where p.amount > 10000; 

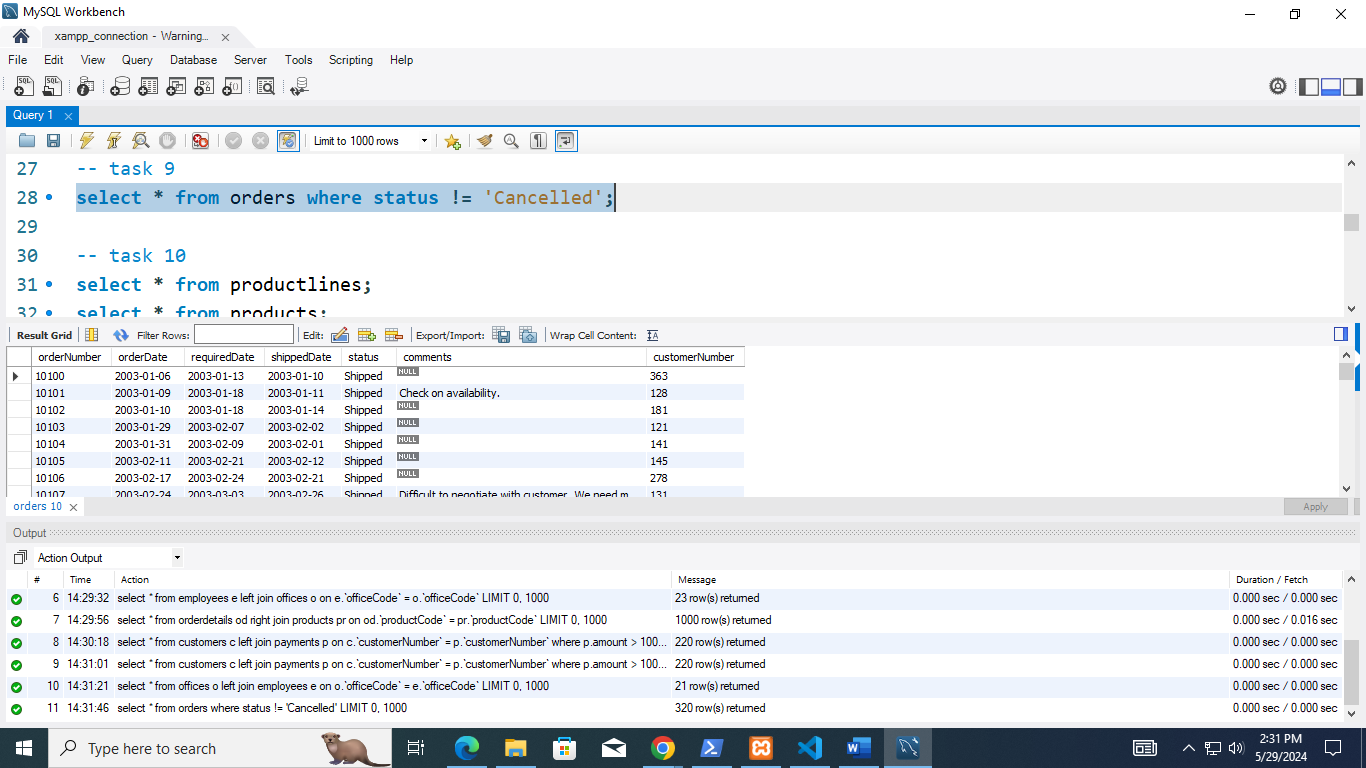
1. Display the list all offices and their employees, including offices with no employees.

select \* from offices o left join employees e on o.`officeCode` = e.`officeCode`;

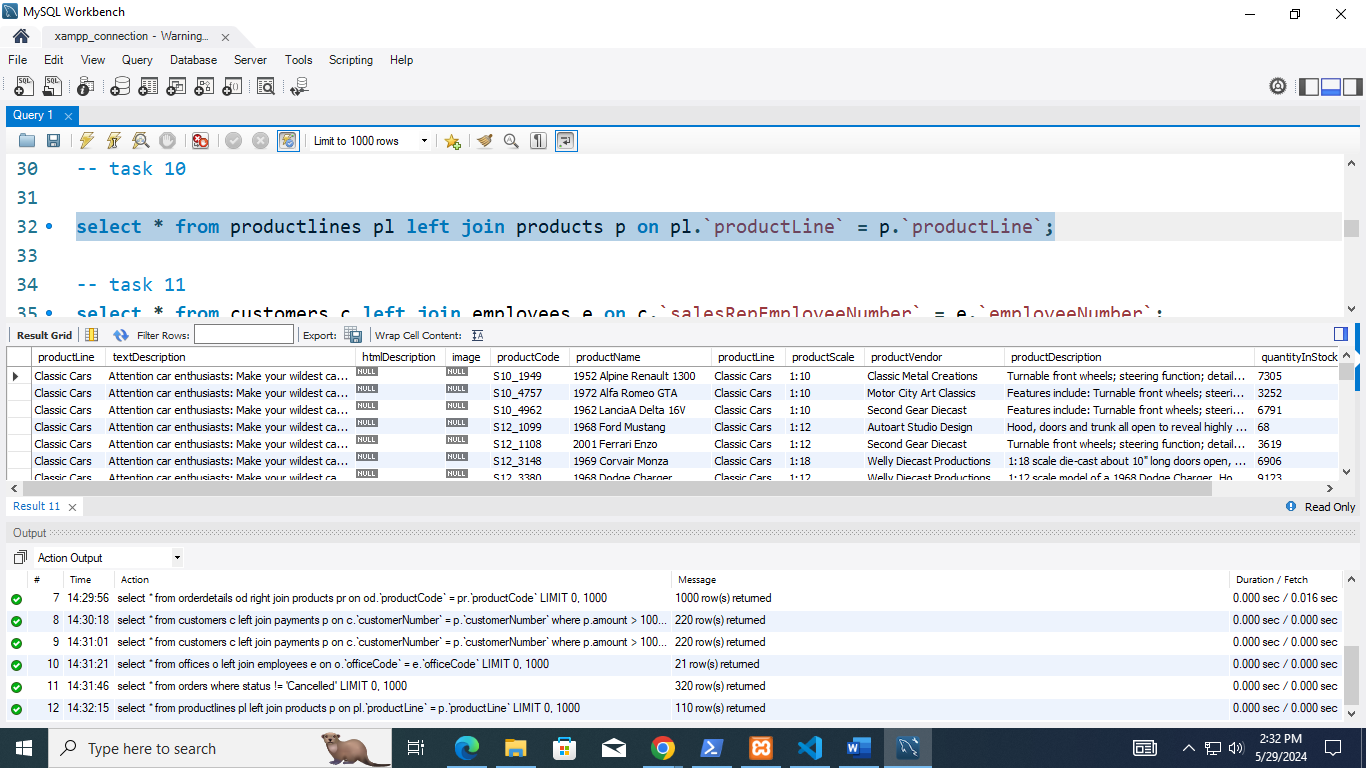


1. List all orders and the status of their shipments, including orders that have not been shipped.

select \* from orders where status != 'Cancelled';

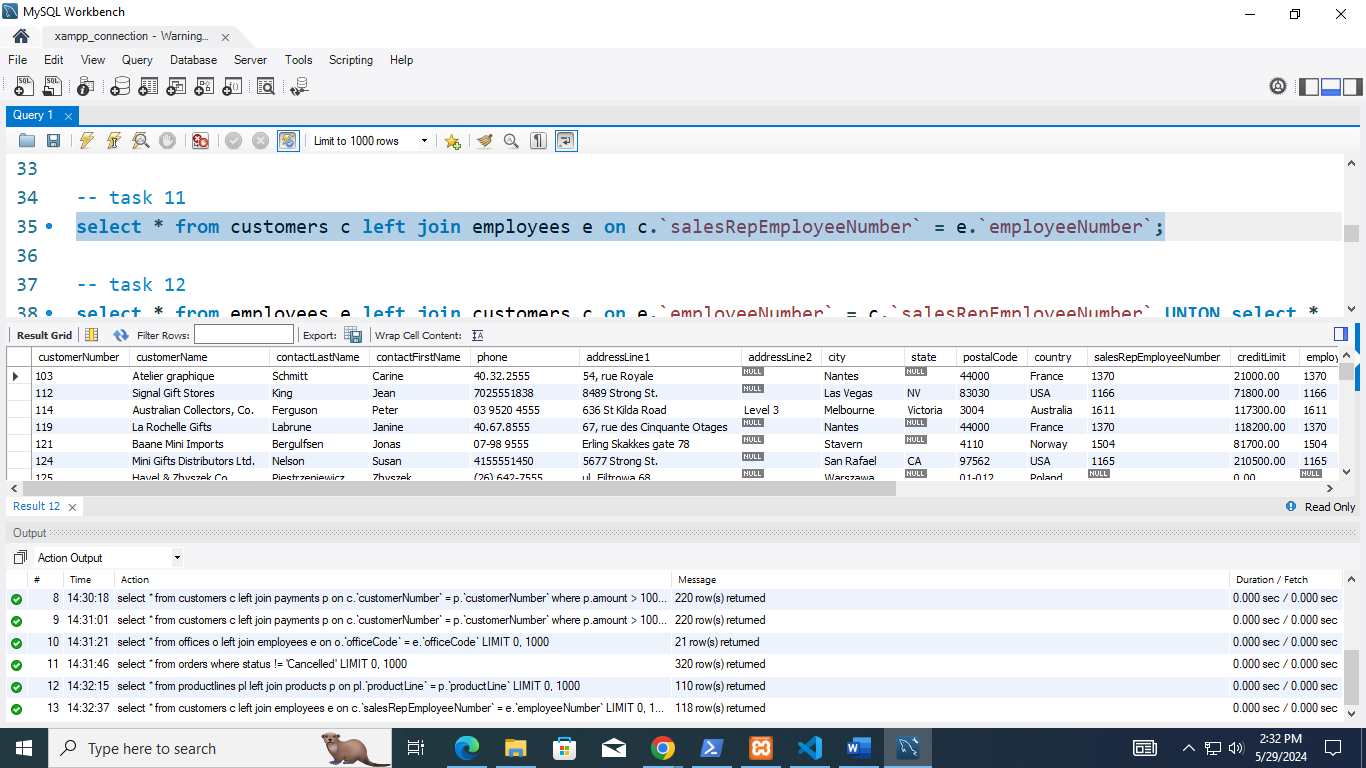


1. List all product lines and the products in each line, including product lines with no products

select \* from productlines pl left join products p on pl.`productLine` = p.`productLine`;

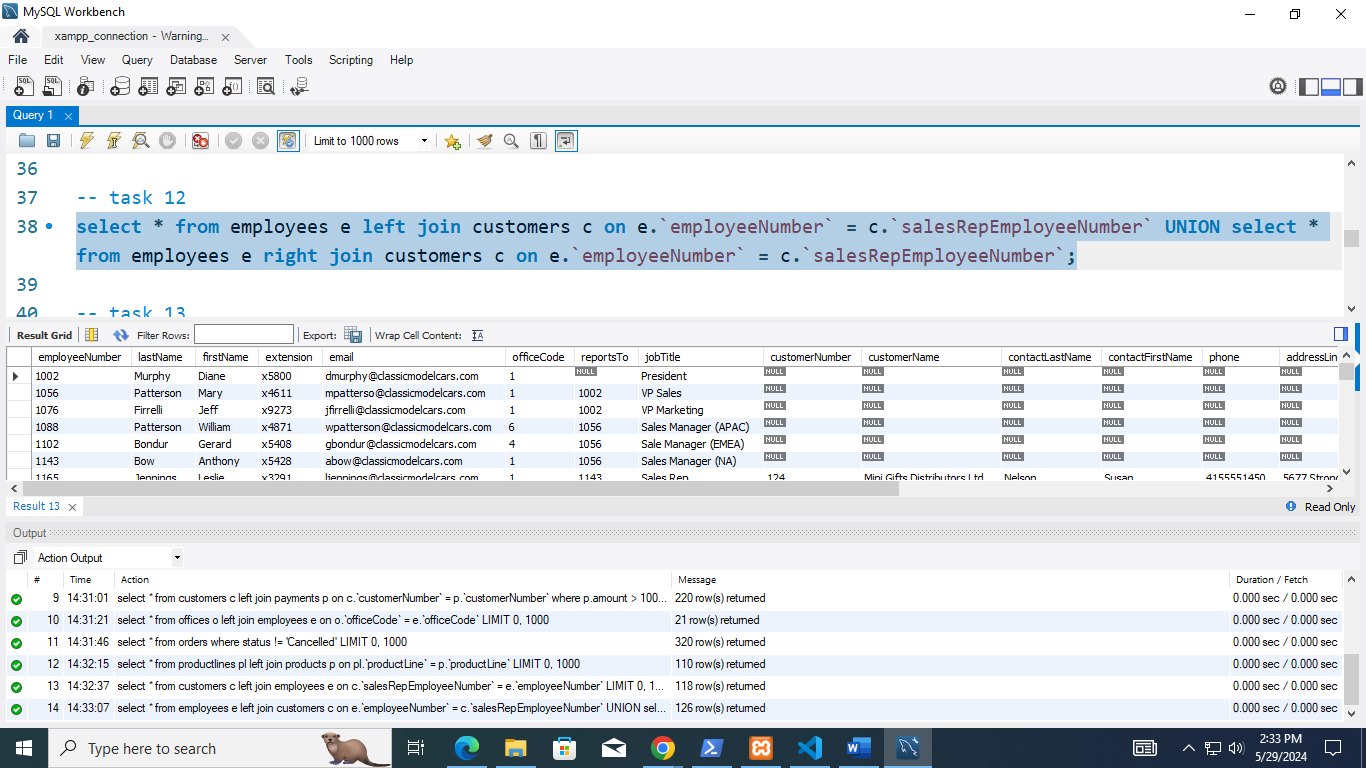
1. Write a query to display all employees along with the customer they handle, make sure to display all of the customer regardless of their sales representative.

select \* from customers c left join employees e on c.`salesRepEmployeeNumber` = e.`employeeNumber`;

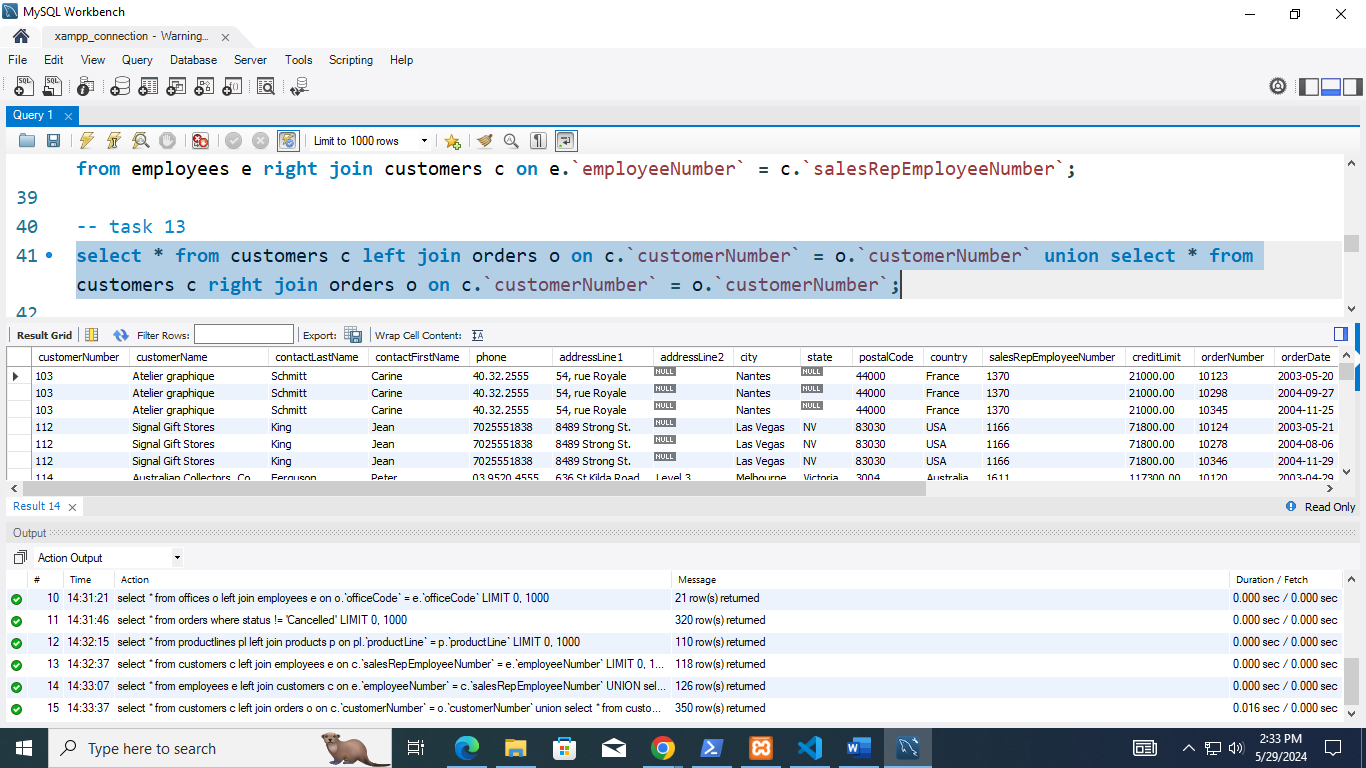


1. Display the list of all employees and all customers in order of their association.

select \* from employees e left join customers c on e.`employeeNumber` = c.`salesRepEmployeeNumber` UNION select \* from employees e right join customers c on e.`employeeNumber` = c.`salesRepEmployeeNumber`;



1. List all customers and their orders, including customers with no orders and orders with no corresponding customers.

select \* from customers c left join orders o on c.`customerNumber` = o.`customerNumber` union select \* from customers c right join orders o on c.`customerNumber` = o.`customerNumber`;

1. List all product lines and their products, including product lines with no products and products not assigned to any product line.
2. List the name of all employees and all customers that has ‘a’ in their name.
3. Retrieve the distinct list of all product codes whose either quantity ordered is greater than 5 or quantity in stock in greater than 5.
4. List all unique country name where either employee office is located or where customer resides.
5. Retrieve the product names along with their respective product line names.
6. List all employees and the offices they work in, including office city and country where there is ‘a’ or ‘s’ in their name.
7. List all orders along with the customer names who placed those orders.

P.S: In order to achieve outer join you can take left join union with right join.